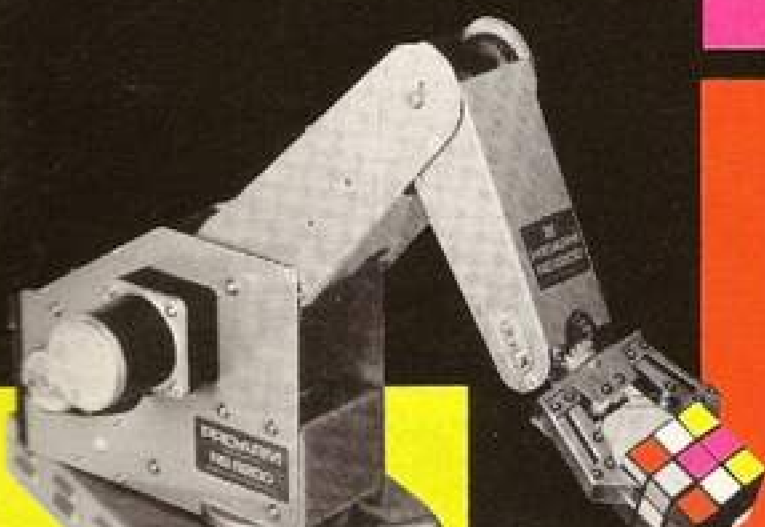
PRO-ARM ROBOTICS RS-2200



MARCRAFT

Model RS-2200
Pro-Arm

KNOWLEDGE TRANSFER PROJECT
RS-2200 P/A PRO-ARM

Advanced technology has provided us with an array of timesaving, entertaining, and useful products. These new products, however, require more and more technical information and skill to design, manufacture, and service. But, technology, no matter how far it advances, will never be able to communicate for us.

Therefore, the purpose of this RS-2200 P/A PRO-ARM project is to act as a catalyst for expanding your understanding of basic industrial robot theory and operation. It also provides an exciting and LIVE vehicle for developing and demonstrating a working robotics system, using the Pro-Arm and a computer. Simulated industrial robot operation, during controlled laboratory research and classroom training exercises, will aid in the development of important job-related skills.

The Model RS-2200 project is divided into several active-learning instruction SECTIONS. The first section contains preliminary information, such as the equipment required, the knowledge transfer objectives of the product, a knowledge transfer guide (to chart progress), a parts list, a parts identification photograph, a schematic diagram of the RS-2200's circuitry, a list of disk-handling instructions, and a list of safety precautions.The second section contains information concerning the proper installation of the Pro-Arm. A systematic approach to the safe handling of the RS-2200 precedes the operating instructions and the running of the two built-in test programs. The third section involves a brief introduction to robotics, followed by a method (for those students with no prior programming experience) of writing, editing, and running robotics programs on the Pro-Arm, using the MARCRAFT PK TEACH software with a computer keyboard or a mouse. For the more advanced students, Section IV provides a thorough explanation of the workings of the Pro-Arm and the identification of its functional parts. It also introduces the BASIC programming language as a more direct method of writing programs for the RS-2200 Pro-Arm. BASIC programming commands allow the student to take advantage of the full range of features offered by the Pro-Arm. Section V contains the Knowledge Transfer Review, permitting the instructor to gauge the progress of each student. Section VI includes additional information which may be useful when either troubleshooting a malfunction or seeking technical assistance. It also contains a glossary of terms used throughout this manual. These instruction sections incorporate a full range of industrial robotics explorations and job-skill experiences.

The Model RS-2200 Knowledge Transfer Project will help you shorten the distance between information reception and actual understanding.

NOTE TO INSTRUCTORS: Page 162 contains a detailed photograph of the Pro-Arm's wired printed-circuit board. Answers to the test explorations of Sections II, III, and IV are contained on pages 167 - 171. These pages may be removed at your option.

As with all the MARCRAFT Knowledge Transfer Projects, the Model RS-2200 project will lend itself to a wide variety of classroom applications. The instructor is free to incorporate the materials presented here in any manner he or she feels is best suited to the individual classroom.

TABLE OF CONTENTS

SECTION I PROJECT INFORMATION

SECTION II INTRODUCTION TO THE PRO-ARM

SECTION III ROBOTICS WITH PK TEACH

SECTION IV PROGRAMMING THE RS-2200 PRO-ARM

SECTION I

PROJECT INFORMATION

EQUIPMENT REQUIREMENTS

To perform the tests, measurements and alignment of your Pro-Arm robot, the following items are required.

1. Robot, Pro-Arm.
2. Power supply, regulated, Pro-Arm.
3. Power cord, ac, Pro-Arm.
4. Experimentation blocks (4 per set).
5. Grid sheet.
6. Computer system (IBM PC, PC/XT, PC AT or compatible).
7. Interface cable (IBM parallel port).
8. Program disk (IBM or MS DOS compatible).
9. PK TEACH Disk.
10. Knowledge Transfer Manual.
11. Mouse.
12. Protractor.
13. Masking tape.
14. Felt pen (water soluble).
15. Cardboard backing.
16. Voltmeter.

KNOWLEDGE TRANSFER OBJECTIVES

EQUIPMENT PRECAUTIONS
Equipment inspection
Damage control
Disk handling
IC handling
Safety considerations

RECORDKEEPING
Inventorying parts
Recording test results in appropriate charts
Completing "Knowledge Transfer Guide"

ROBOTICS COMPREHENSION
Learning the history of robotics
Defining robotic terms
Understanding stepper motors
Classifying robots
Identifying Pro-Arm characteristics

EQUIPMENT INSTALLATION
Placing the grid sheet
Centering the Pro-Arm
Interfacing with a computer
Supplying the power
Expanding the Pro-Arm's memory
Interfacing with a mouse
Interfacing external devices

EQUIPMENT TESTING
Performing the power-up procedure
Running the internal self-test programs
Running the computer demo program
Calibrating the demo program
Checking the external interfaces

COMPUTER SKILLS
Booting up the computer
Loading the PK TEACH software
Using a mouse/keyboard to run the Pro-Arm
Programming the Pro-Arm
Learning to program in BASIC
Giving the Pro-Arm specific commands
Saving new programs to disk
Loading old programs from disk
Editing existing programs
Deleting unwanted programs
Learning ASCII characters

EQUIPMENT AND CIRCUIT COMPREHENSION
Block diagram
Schematic diagram
PC-board wiring photograph

TROUBLESHOOTING
Reading service flowcharts
Isolating defective components
Reading a schematic diagram
Removing and replacing defective ICs
Adjusting the gripper limit switch
Adjusting the drive chains

KNOWLEDGE TRANSFER GUIDE

The purpose of this Knowledge Transfer Guide is to allow you to keep an accurate record of your progress in the operation and programming of the Pro-Arm. To achieve optimum knowledge transfer or educational benefit from this project, be sure to follow the sequence listed below. Do not proceed to the next LEARNING EXPERIENCE until the previous one has been completed and fully understood.

At the completion of each LEARNING EXPERIENCE, obtain your instructor's evaluation and initials. The FINAL EVALUATION is provided so your instructor can indicate your degree of achievement on the entire project.

Good luck, and enjoy this KNOWLEDGE TRANSFER PROJECT!

YOUR NAME ____

LEARNING EXPERIENCE	COMPLETION DATE	INSTRUCTOR	
		EVALUATION	INITIALS
SECTION I PROJECT INFORMATION			
PARTS INVENTORY			
DISK-HANDLING INSTRUCTIONS			
SAFETY PRECAUTIONS			
SECTION II INTRODUCTION TO THE PRO-ARM			
NOMENCLATURE			
Lab Exploration 1			
Lab Exploration 2			
THE PRO-ARM INSTALLATION			
Lab Exploration 3			
Lab Exploration 4			
OPERATIONAL TESTING			
Lab Exploration 5			
Lab Exploration 6			
Lab Exploration 7			
Lab Exploration 8			
SECTION III ROBOTICS WITH PK TEACH			
INTRODUCTION TO ROBOTICS			
Lab Exploration 9			
Lab Exploration 10			
PK TEACH SOFTWARE			
Lab Exploration 11			
Lab Exploration 12			
Lab Exploration 13			
Lab Exploration 14			
Lab Exploration 15			
Lab Exploration 16			
Lab Exploration 17			
Lab Exploration 18			

LEARNING EXPERIENCE	COMPLETION DATE	INSTRUCTOR EVALUATION	INSTRUCTOR INITIALS
SECTION IV PROGRAMMING THE RS-2200 PRO-ARM			
PARTS AND PERFORMANCE			
Lab Exploration 19			
Lab Exploration 20			
Lab Exploration 21			
Lab Exploration 22			
Lab Exploration 23			
Lab Exploration 24			
Lab Exploration 25			
OTHER INTERFACING FEATURES			
Lab Exploration 26			
A BASIC DEMO			
Lab Exploration 27			
Lab Exploration 28			
PRO-ARM COMMANDS			
Lab Exploration 29			
Lab Exploration 30			
Lab Exploration 31			
Lab Exploration 32			
Lab Exploration 33			
Lab Exploration 34			
Lab Exploration 35			
WORK ENVELOPE			
Lab Exploration 36			
Lab Exploration 37			
REPEATABILITY			
Lab Exploration 38			
PROGRAMMING INTERFACE FEATURES			
Lab Exploration 39			
Lab Exploration 40			
Lab Exploration 41			
YOUR FIRST BASIC PROGRAM			
Lab Exploration 42			
SECTION V TESTING			
KNOWLEDGE TRANSFER REVIEW			
FINAL EVALUATION			

PARTS INVENTORY

CAREFULLY unpack the Pro-Arm robot, following the instructions included. Check (√) each item against the following **PARTS LIST**. For assistance in proper identification of these items, refer to the illustrations in **PARTS IDENTIFICATION** on the next page.

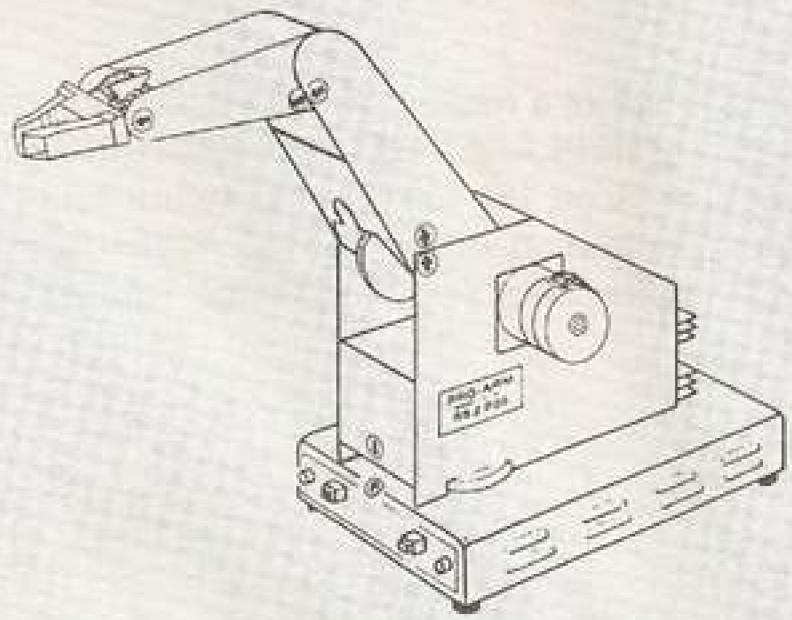
In the event of a missing, incorrect, or damaged part, refer to the **REPLACEMENT and EXCHANGE PARTS** heading on **Page 165** for ordering instructions.

Upon **completion** of the **PARTS INVENTORY**, have your **instructor initial your** Knowledge Transfer Guide.

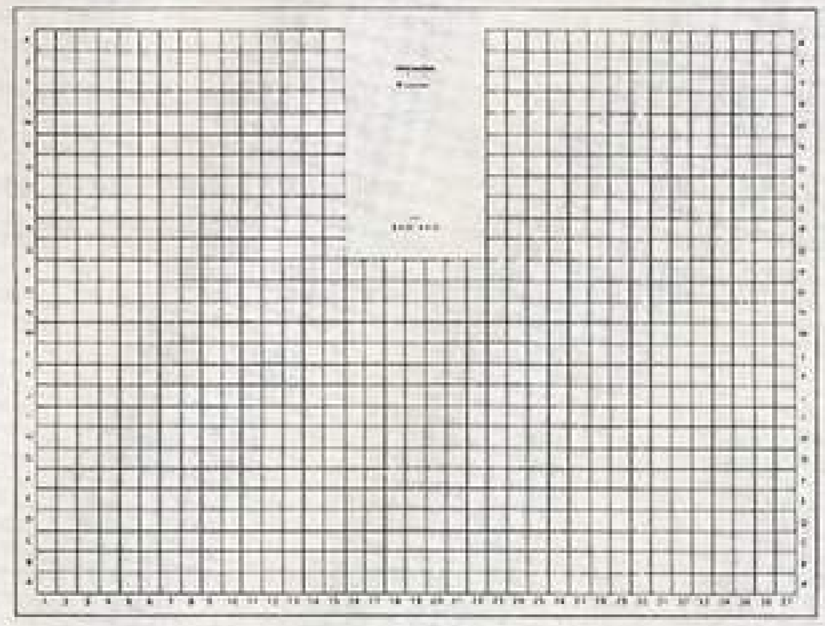
PARTS LIST

√	DESCRIPTION	PART#	QTY
	Pro-Arm Robot	RA-2200	1
	Power supply, regulated	RA-201	1
	Power cord, ac	RA-203	1
	Interface cable, IBM PC	RA-205	1
	Connector, 6-pin (wired)	RA-206	1
	Connector, 2-pin (wired)	RA-207	2
	Connector, 10-pin (wired)	RA-209W	1
	Cylinder/bottle gripper kit	RA-215	1
	Experimentation blocks	RA-216	4
	Grid sheet	RA-217	1
	Program disk, PK TEACH	RA-218	1
	Program disk, MARCRAFT	RA-219	1
	Knowledge Transfer Manual	RA-220	2
	Mouse, serial	RA-275	1
	9 to 25-pin D-shell adapter	RA-276	1

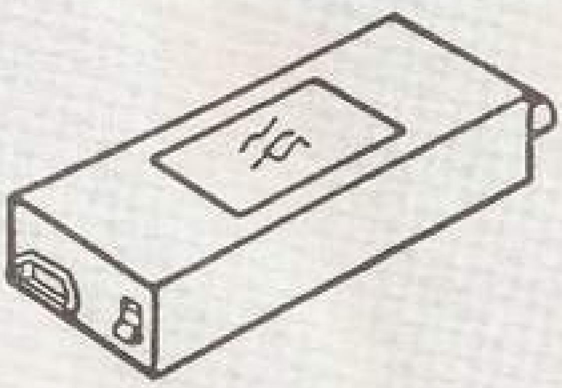
PARTS IDENTIFICATION



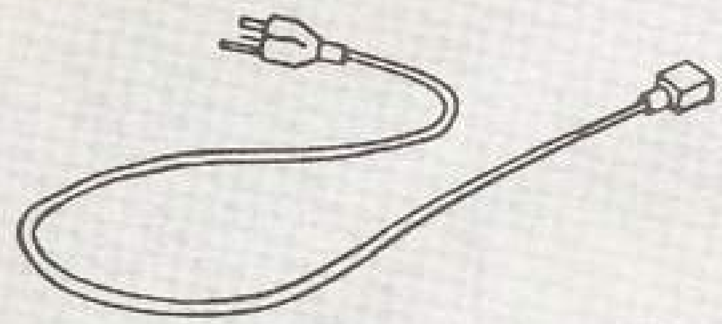
PRO-ARM ROBOT



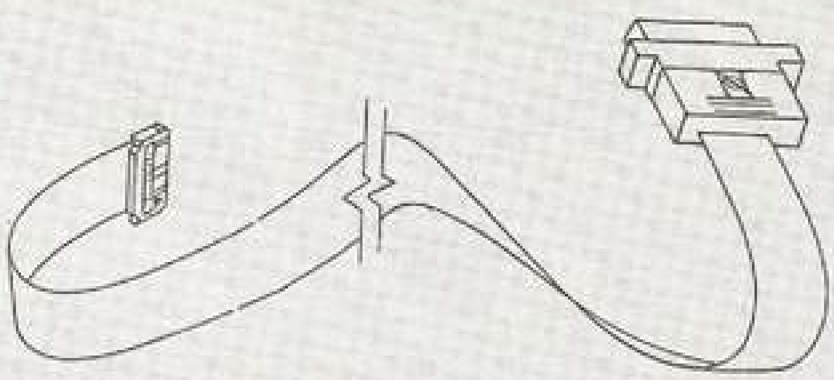
GRID SHEET



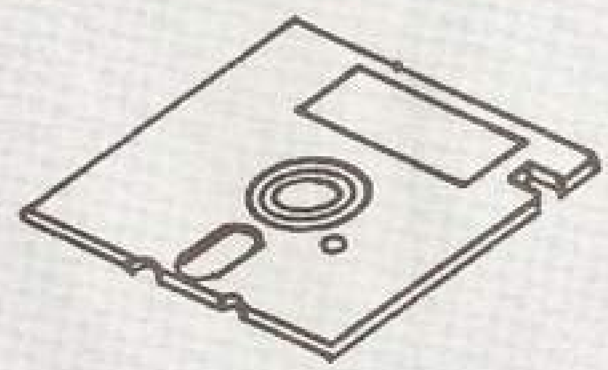
POWER SUPPLY



AC POWER CORD

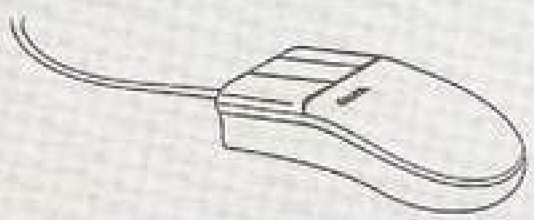


INTERFACE CABLE

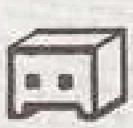


PROGRAM DISK

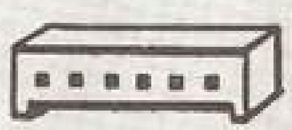
EXPERIMENTAL BLOCKS



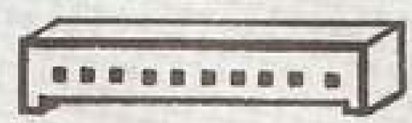
MOUSE



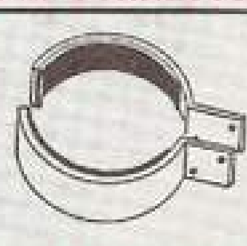
2-PIN CONNECTOR



6-PIN CONNECTOR

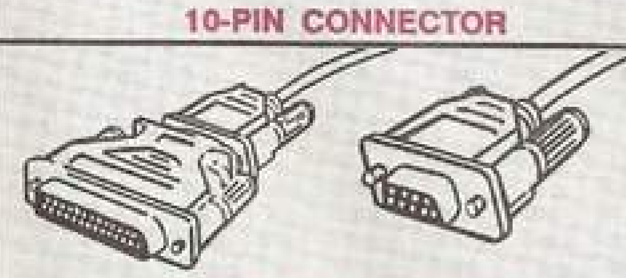


10-PIN CONNECTOR



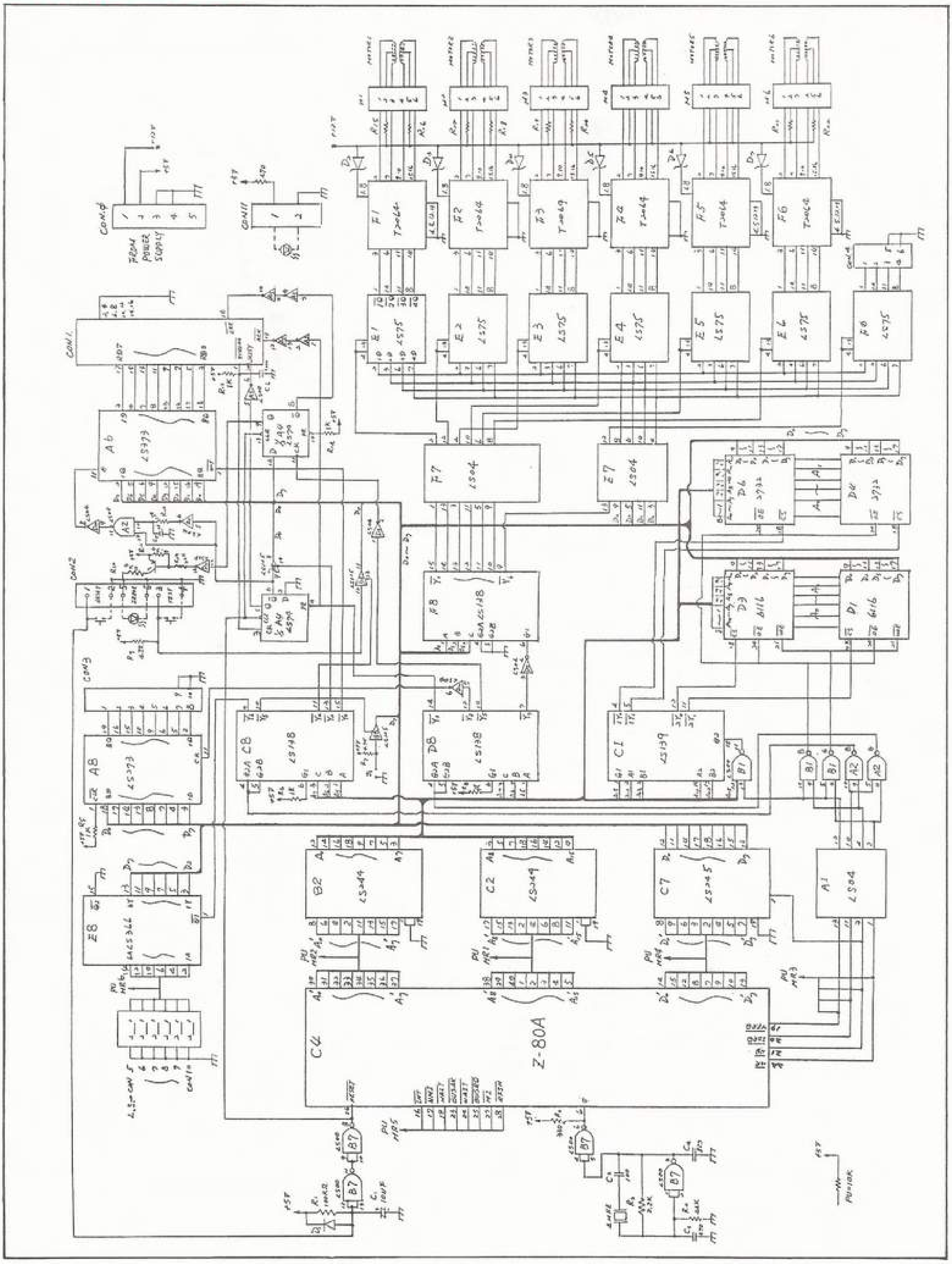
CYLINDER GRIPPER KIT

MANUAL



9 TO 25-PIN ADAPTER

SCHEMATIC DIAGRAM



DISK HANDLING INSTRUCTIONS

WARNING: Read and follow the next series of instructions and understand them thoroughly before inserting, removing, or storing any of the disks used with the computer in the Pro-Arm system, in order to avoid possible damage to them. They are made of extremely delicate materials and are susceptible to destruction from a variety of sources.

As you read each instruction, place a check mark in the space provided.

___1. **Always keep disks in their protective sleeves** when they are not actually inserted in the computer's disk drive. If dust is allowed to collect on the recording surfaces, degraded performance and, eventually, total failure is a sure result. The disk drives themselves will also be damaged.

___2. **Avoid bending the disks,** especially when inserting them into the drive door. One crease is all it takes to render a disk useless.

___3. **Insert disks very carefully** through the drive door. Sometimes a drive will offer a small amount of resistance as a disk is being inserted. A smooth, gradual insertion will prevent data-destroying scratches from being deposited on the recording surface.

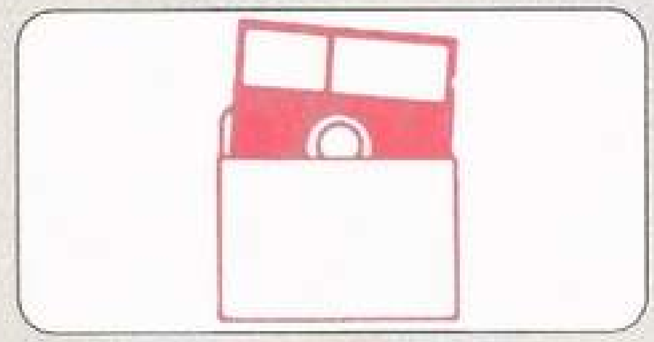
___4. **Never touch the surface** of an exposed disk surface. Just the smudge of a greasy fingerprint will send a disk on a quick trip to the graveyard.

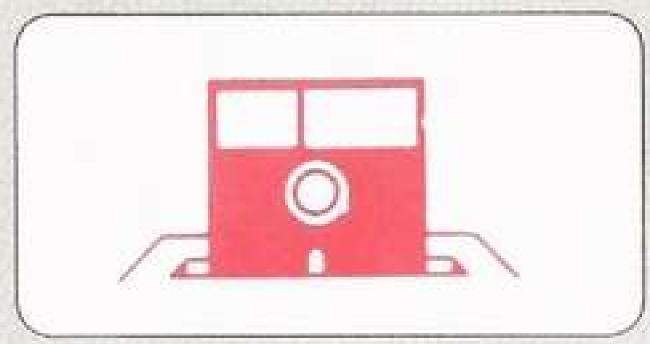
___5. **Keep disks stored upright** in a cool, dry place, out of direct sunlight and away from strong electromagnetic fields, such as those generated by television sets or stereo speakers.

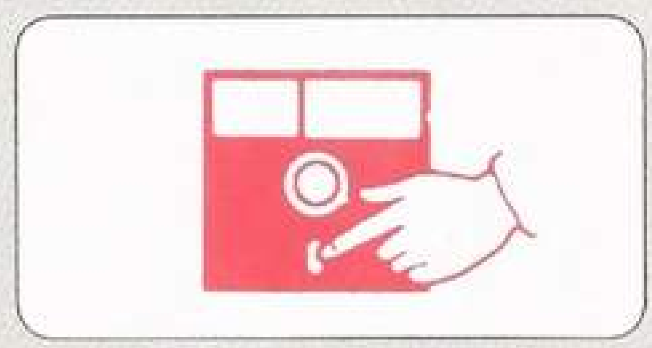
___6. **Use only a felt-tipped pen** when writing on labels already attached to a disk. Do not use pencils or hard-tipped pens, as these will scratch the disk surface.

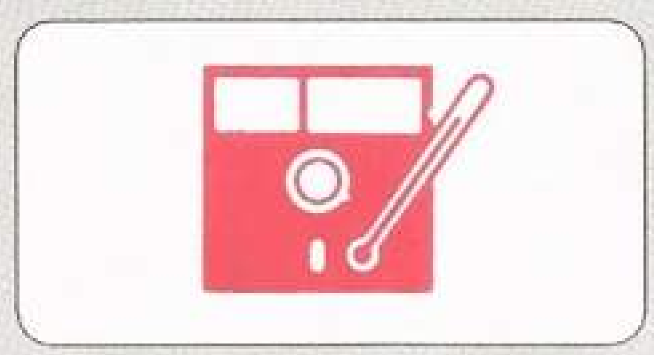
___7. **Always make a backup copy** of an important disk, just in case something happens to destroy your original.

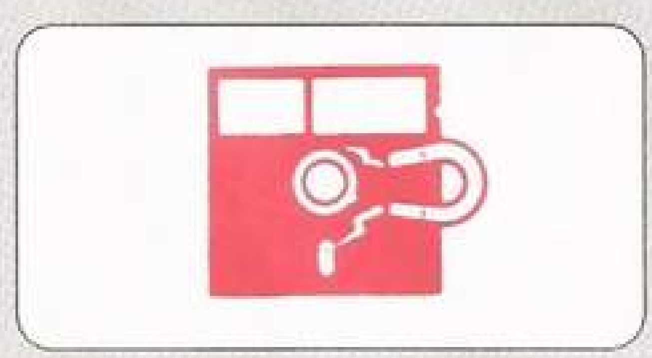
Be sure you thoroughly understand the preceding rules before inserting, removing, or storing any Program or Data disks used in the Pro-Arm system computer. After reading and checking off each of the instructions above, have your instructor initial your Knowledge Transfer Guide.







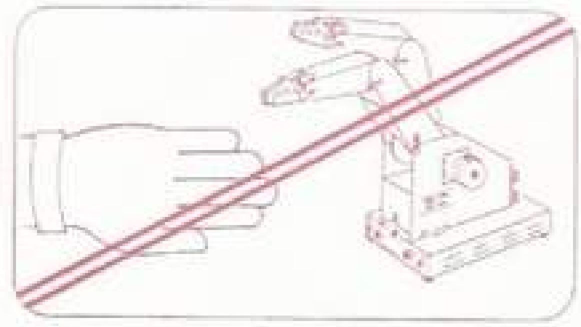




SAFETY PRECAUTIONS

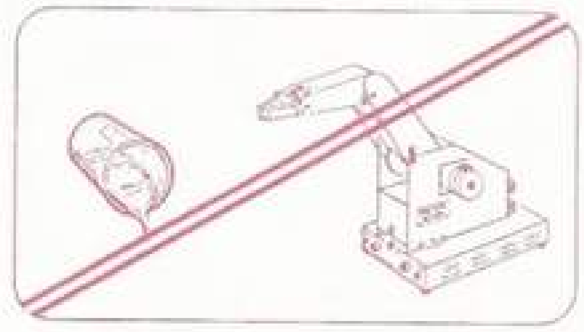
Before conducting any activities with the Pro-Arm robot, the following safety precautions should be thoroughly read and understood.

To prevent damage to the Pro-Arm robot, and injury to yourself and to others, carefully follow these precautions whenever you are engaged in robotic activity. As you read each precaution, place a check mark (√) in the space provided to assure that each one has been reviewed and understood.



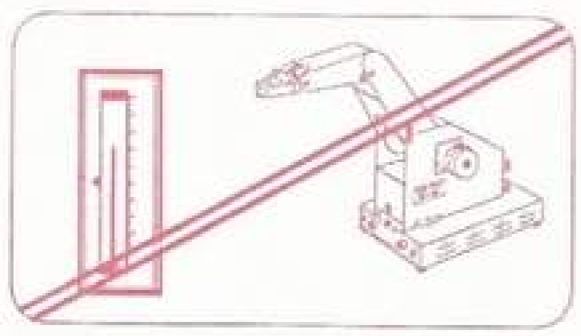
___1. **Use** a power source that has **a stable ac line voltage of 105–125-V ac, 50–60Hz** (or 200–264-V ac, if alternately switched).

___2. **Do not move the axes manually** (whether or not the robot is in motion).



___3. While the Pro-Arm is in motion, **do not block its movement with any object or with your body.** Blocking the robot's movement will force the programmed sequence to become out of sync, possibly causing mechanical damage.

___4. To avoid possible injury, **do not touch any movable part** during operation or while the Pro-Arm is turned on.

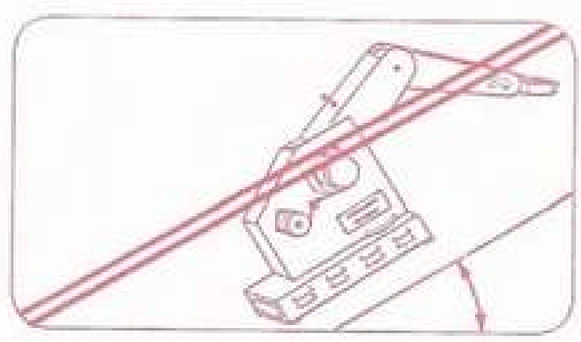


___5. **Do not operate** the Pro-Arm **with any protective cover removed,** unless specifically instructed to do so during an exploration.

___6. **Keep** loose-fitting clothes and other articles **away from moving parts** of the Pro-Arm.

___7. **Do not dismantle or pull on the interface cable.**

___8. **Always operate** the Pro-Arm with it positioned **on a flat, stable, and horizontal surface.**



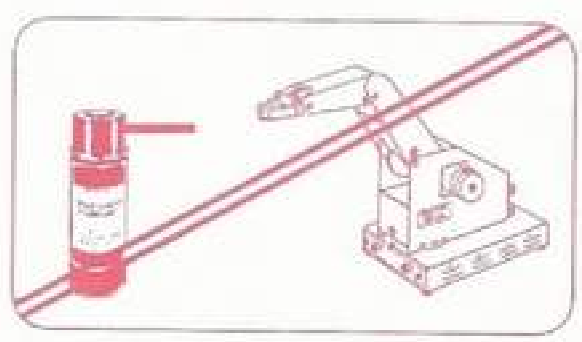
___9. **Avoid** operating the Pro-Arm in **direct sunlight,** near **heat** registers, or in **wet locations.**

___10. **Keep liquids away** from the Pro-Arm at all times. Never use the gripper to lift drinks or containers of fluids.

___11. **Do not operate the Pro-Arm outdoors.**

___12. **Avoid** placing or using the Pro-Arm in **locations having severe vibration, dust, humidity, moisture, or excessive magnetic fields.**

___13. **Do not block the ventilation holes** on the sides or bottom of the Pro-Arm's base.



___14. **Do not lubricate the Pro-Arm.**

___15. **Do not** program the Pro-Arm to perform any action which would **exceed an axis limit of travel.**

Note: This robot is not designed for industrial use.

Periodically review these SAFETY PRECAUTIONS for the Pro-Arm, and keep them in mind during all of the Lab Explorations. This completes SECTION I of your Knowledge Transfer Project. If you feel you thoroughly understand the safety precautions, have your instructor initial your Knowledge Transfer Guide.

SECTION II

INTRODUCTION TO THE PRO-ARM

DISCUSSION

In this section, you will learn some of the terms used to describe various parts and aspects of the Pro-Arm system, learn to install it for normal operation, and finally, learn to perform two operational tests to check its electrical and mechanical fitness.

At the end of this manual is a glossary with the definitions of various robotic terms. When words appear in the text which do not have sufficient explanation or description, please turn to the glossary for a further clarification.

These actions will be accomplished in the following sequence:

1. Nomenclature
2. The PRO-ARM Installation
3. Operational Testing

Be sure to follow the steps of each of these activities in the exact sequence in which they are presented. This section is designed to begin your familiarization with the Pro-Arm. More detailed information about the Pro-Arm will be presented in Section III.

NOMENCLATURE

DISCUSSION

Before we can begin to install the Pro-Arm system, there is an important matter to consider. ***Nomenclature.***

Nomenclature is simply a system of names that we give to specific items pertaining to the particular science we happen to be engaged in. In the science of robotics in general, and concerning the RS-2200 Pro-Arm system in particular, we need to be familiar with the language commonly used to discuss the subject. In other words, we need to know what we are talking about.

Since this is still only an introduction to the Pro-Arm, the nomenclature contained herein will be limited to only the names of the parts we need to know in order to get the Pro-Arm system operating. Additional terminology will be presented later, when we need to understand specific details about robotics systems. The material will be presented in the following sequence:

1. Pro-Arm Robot
2. IBM-Type Computer

The answers and typical test results for the following exploration are shown in the **ANSWERS** section on Pages 167–171. (If these pages have been removed, have your instructor approve your test results.)

LAB EXPLORATION 1

In the appropriate space provided in Fig. 1, write the name of the part described in each of the following steps. Notice that the part lettering in Fig. 1 corresponds to each step letter. As you identify each part, also study its purpose.

___1. NAMES OF PARTS **(Pro-Arm)**

Purpose: This exploration is designed to give the student practice in identifying and naming 21 of the parts associated with the MARCRAFT RS-2200. It will familiarize the student with the basic purpose of each of the Pro-Arm parts which are named and identified.

Equipment: Knowledge Transfer Manual, Pencil

As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

___a. The **POWER CORD** connects the Pro-Arm's power supply unit to a 110-volt ac (alternating current) receptacle.

___b. The **POWER SWITCH** turns the Pro-Arm's power-supply unit on and off.

___c. The **POWER SUPPLY** is used to change the high ac voltage from the wall receptacle to the low dc (direct current) voltage which is required for proper operation of the Pro-Arm.

___d. The **POWER CABLE** connects the dc voltage from the power supply to the Pro-Arm.

___e. The **ERROR LAMP** does exactly what the name implies. It detects some type of error. The error lamp may be turned off by depressing the reset button.

___f. The **TEST BUTTON** is used to perform the self-tests of the Pro-Arm.

___g. The **RESET BUTTON** is used to reset the Pro-Arm, to receive a new set of commands, and to clear its RAM (Random-Access Memory).

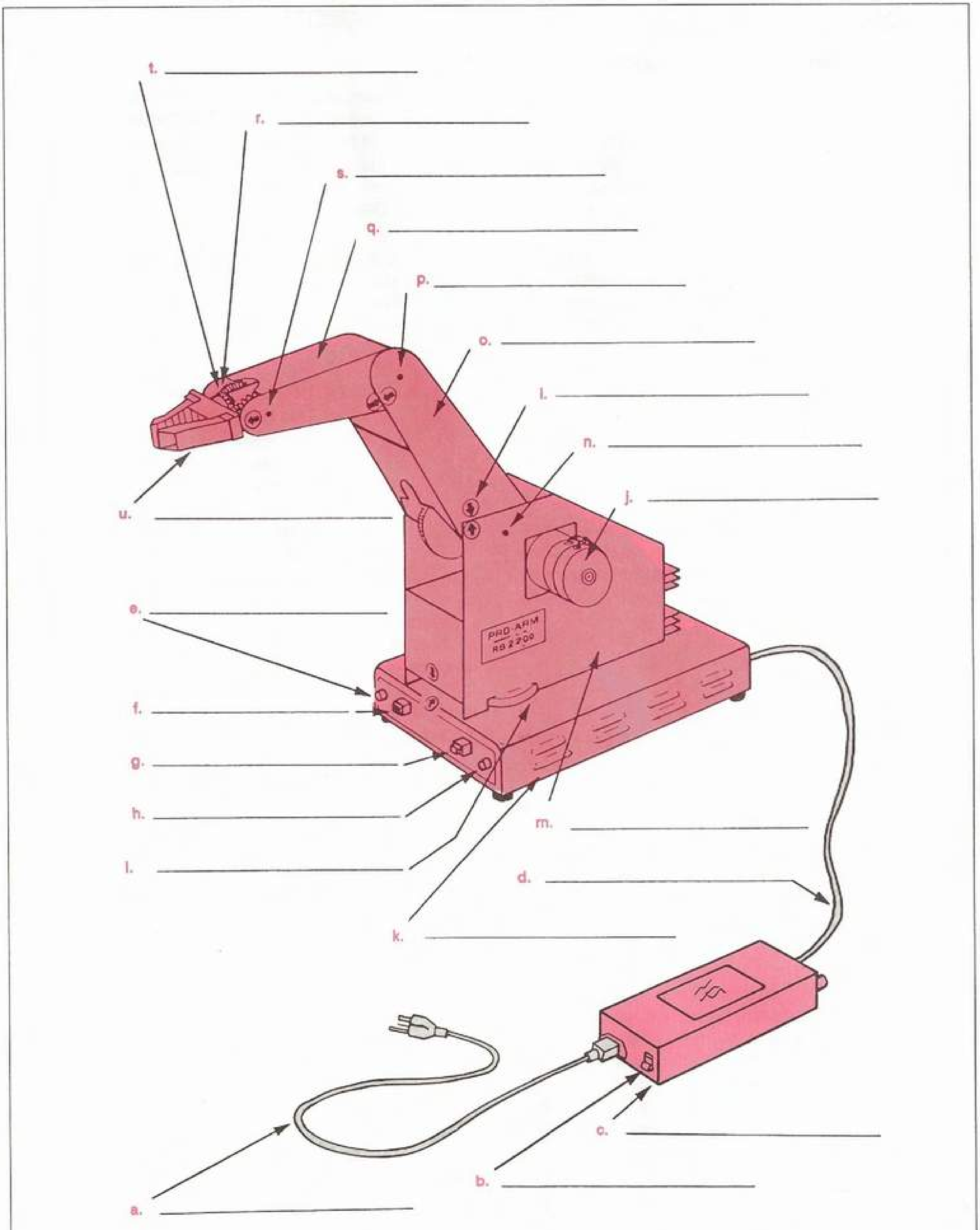


Fig. 1: Parts of the Pro-Arm

____h. The **POWER LAMP**, when lit, indicates that the power cord is plugged into a 110-volt ac receptacle and that the power-supply switch is turned on.

____i. The **ALIGNMENT ARROWS** are used for placing the arm joints in their zero position.

____j. The **STEPPING MOTOR** is used to move an axis (or joint).

____k. The **BASE** is used to provide a foundation for the Pro-Arm. It is also used to house a PCB (printed-circuit board).

____l. The **BASE AXIS** is used to rotate the body of the Pro-Arm.

____m. The **BODY** moves in a horizontal plane and also supports the arm.

____n. The **SHOULDER AXIS** is used to move the upper arm.

____o. The **UPPER ARM** is connected to the body and provides vertical arm movement.

____p. The **ELBOW AXIS** is used to move the lower arm.

____q. The **LOWER ARM** is connected to the upper arm and provides arm extension.

____r. The **WRIST ROLL AXIS** is used to rotate the wrist on the Pro-Arm.

____s. The **WRIST PITCH AXIS** is used to move the wrist up or down.

____t. The **WRIST** may be rotated to provide "roll" or moved up and down to provide "pitch."

____u. The **GRIPPER** or HAND is called an "end effector" and is used for holding purposes.

____v. This completes Lab Exploration 1, "Names of Parts (Pro-Arm)." Have your instructor initial your Knowledge Transfer Guide.

The answers and typical test results for the following exploration are shown in the **ANSWERS** section on Pages 167–171. (If these pages have been removed, have your instructor approve your test results.)

LAB EXPLORATION 2

In the appropriate space provided in Fig. 2, write the name of the part described in each of the following steps. Notice that the part lettering in Fig. 2 corresponds to each step letter. As you identify each part, also study its purpose.

____2. NAMES OF PARTS **(Computer System)**

Purpose: This exploration is designed to give the student practice in identifying and naming 11 of the parts associated with an IBM-compatible computer. It will familiarize the student with the basic purpose of each of the computer parts which are identified.

Equipment: Knowledge Transfer Manual, Pencil

As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____a. The **SYSTEM UNIT** contains the brains (microprocessor and memory chips), PCBs (printed-circuit boards) and all of the associated circuitry of the computer. It also houses the power supply.

____b. The **KEYBOARD** allows the inputting of data (alphanumeric characters), as well as various commands to the computer. It plugs into a special socket on the system unit. Older computer models have this socket located in the rear panel, while newer models place it more conveniently on the front panel.

____c. The **MONITOR** allows the visual display of data and graphics (pictures) during computer operation.

____d. The **SYSTEM POWER CORD** connects the computer's internal power-supply unit to a 110-volt ac (alternating current) receptacle.

____e. The **SYSTEM POWER SWITCH** (or button) turns the computer's power-supply unit on and off.

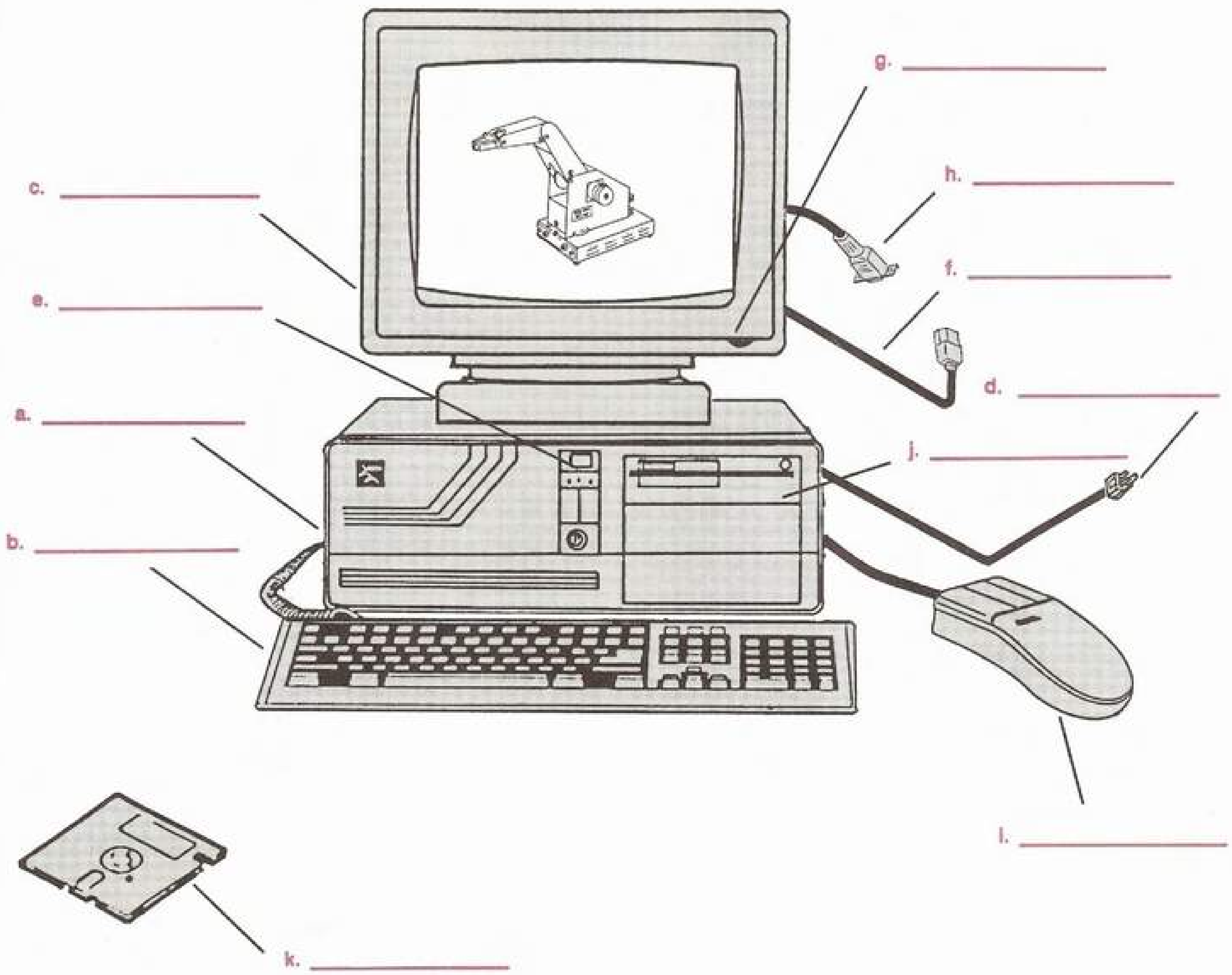


Fig. 2: Parts of the Computer System

____f. The **MONITOR POWER CORD** supplies power to the monitor from the computer. It connects to a special plug-in receptacle located on the rear panel of the system unit.

____g. The **MONITOR POWER SWITCH** (or button) turns the monitor on or off. In most cases, this switch is kept in the "**ON**" position so that the monitor is automatically turned on or off along with the system unit.

____h. The **VIDEO SIGNAL CABLE** connects the video output signals from the system unit to the monitor. The D-Shell connector on the end of the cable plugs into a matching socket on a video adapter card through a slot in the rear panel of the system unit.

____i. The **MOUSE** is another type of input device. It allows some forms of program operation to be done much more quickly and easily than can be done with the keyboard. Its cable plugs into a special serial port connector socket through one of the slots in the back panel of the system unit.

____j. The **FLOPPY DISK DRIVE** is a memory device, an input device, and an output device all in one unit. It is primarily used to load (input) programs from a floppy disk into the computer's internal memory, and to save (output) programs from the computer's internal memory to a floppy disk. Because the programs are stored on the floppy disk, the floppy drive is known as an external memory device.

____k. The **FLOPPY DISK** stores the programs and data (software) that tell the computer what to do and how to do it. The disk is made of delicate material and must be handled carefully at all times.

____l. This completes Lab Exploration 2, "Names of Parts (Computer System)," and the NOMENCLATURE activities. Have your instructor initial your Knowledge Transfer Guide.

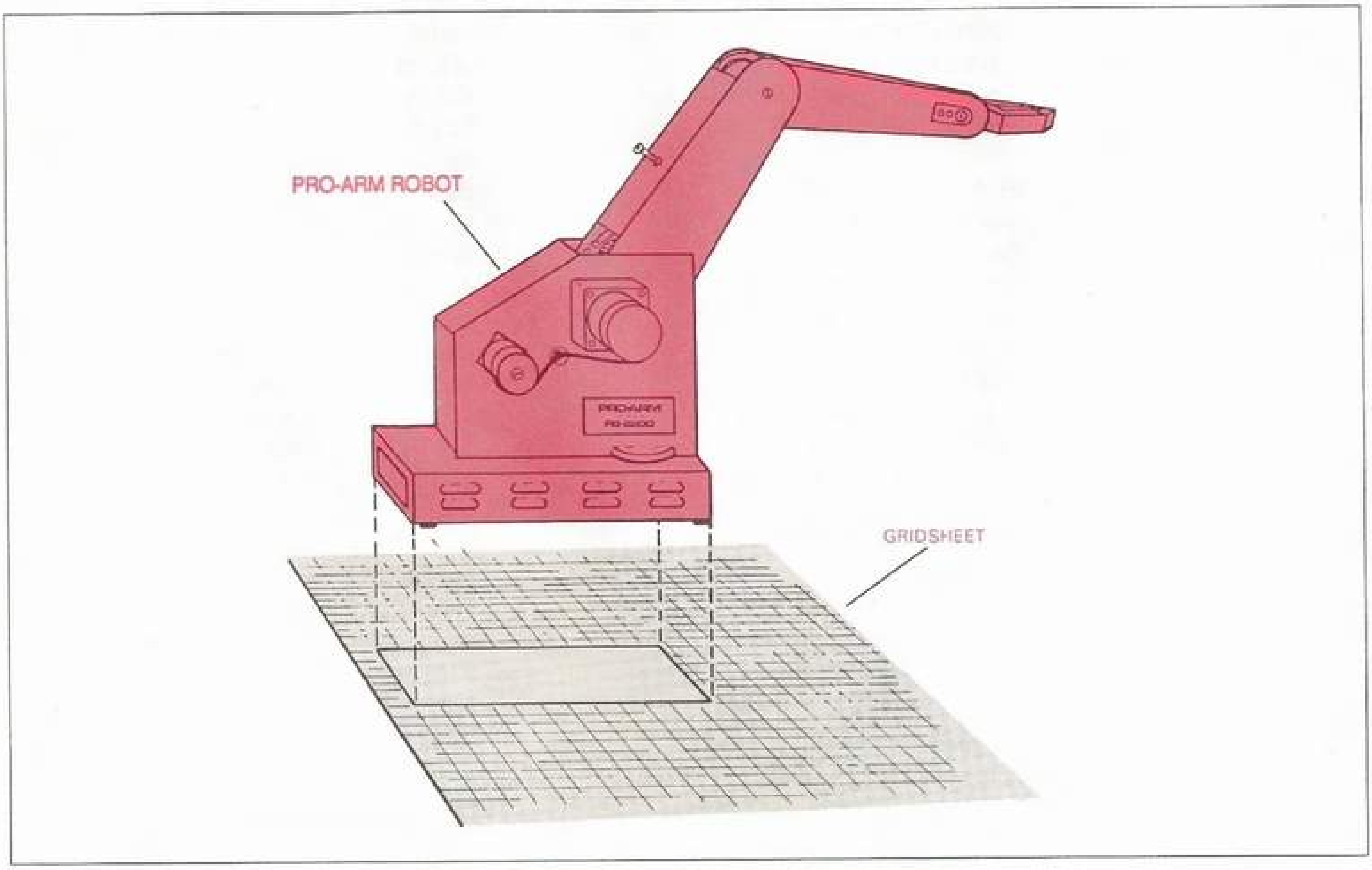


Fig. 3: Positioning the Pro-Arm on the Grid Sheet

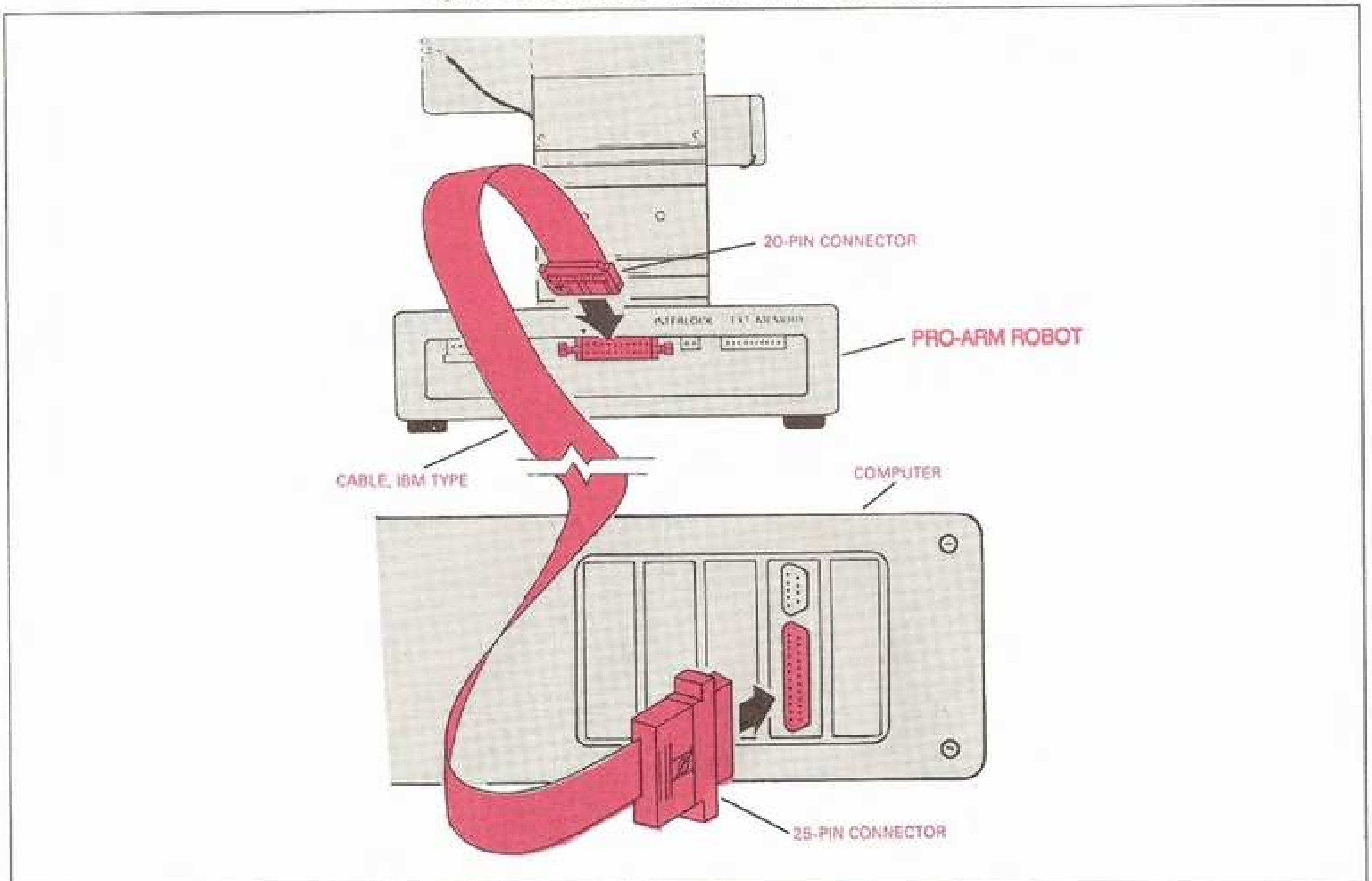


Fig. 4: Connecting the IBM-type Interface Cable

PRO-ARM INSTALLATION

DISCUSSION

Adhere to the following setup procedures, in order to get the most knowledge transfer possible from the RS-2200 P/A Pro-Arm robotics system.

Even though the Pro-Arm can be activated and tested while not connected to the computer, the following material will cover the entire installation procedure required when activating the RS-2200 system. This procedure will be presented in the sequence below:

1. PRO-ARM Hardware
2. Program Disks and DOS

PRO-ARM HARDWARE

Commit the following procedure to memory as soon as possible. You will repeat it many times during the course of this project.

LAB EXPLORATION 3

____3. HARDWARE

Purpose: This procedure will properly install the complete Pro-Arm system for normal operation and will ensure that its working envelope is free from any obstruction.

Equipment: Grid Sheet, Pro-Arm Robot, Table or Desk, IBM-Compatible Computer System, Pro-Arm Interface Cable, Pro-Arm Power Supply, Pro-Arm Power Cord, Serial Mouse, and the 9 to 25-Pin Adapter, if required.

As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

____a. Unroll the grid sheet that is supplied with your robot.

Note: The grid sheet is provided so that most Pro-Arm activities and learning experiences will be conducted from known points of reference. This ability helps to determine how well the Pro-Arm repeats its assigned tasks.

____b. Position the grid sheet on the flat surface of a sturdy bench or table, having a minimum work area of 30 in x 40 in (76cm x 102cm), that is free from any obstructions.

____c. Place the Pro-Arm robot on the grid sheet in the position indicated. (Refer to Fig. 3.)

____d. Plug the 25-pin (D-shell) cable connector into the parallel-printer port connector on the rear of the computer. (Refer to Fig. 4.)

Note: The connector can be prevented from working loose from the computer by positioning and tightening the two small screws on the cable connector into their proper holes on the port connector.

____e. Plug the 20-pin connector into the input connector marked "**COMPUTER**", on the back of the Pro-Arm's base, matching the arrow on the connector with the arrow marking on the robot. The connector can be inserted in only one direction. Again, refer to Fig. 4.

Note: To disconnect the cable from the Pro-Arm, push the two levers located on each side of the connector to the side, as shown in Fig. 5. The connector should slide out.

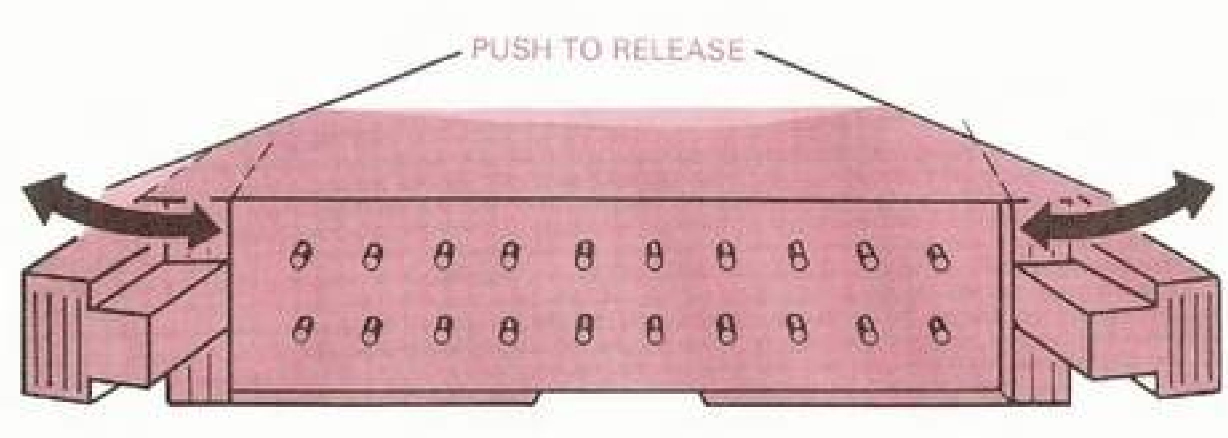


Fig. 5: Disconnecting the IBM-type Interface Cable

Note: The IBM-type cable has a 25-pin (DB25) connector on one end, which is compatible with the parallel-printer port on the IBM PC, PC/XT or PC AT. If you are interfacing this robot with a computer other than these models, it may be necessary to change the cable connector, pin configuration, or both. At this time, compare the pin configuration of your computer's printer port with Table 1.

Caution: If any discrepancies are found, you will be required to take whatever action is necessary in order to insure that the signals which are listed for the Pro-Arm connector, that is, the end of the interface cable which plugs into the Pro-Arm, are available at the specific pins indicated in Table 1.

Note: A solid-state power supply is included with the Pro-Arm system. (Refer to Fig. 6.) It provides outputs of 5-V dc for the operation of the logic IC circuits of the Pro-Arm and 12-V dc for the operation of the stepper motors.

____f. Plug the Power-Supply Output Cable into the power connector on the rear of the Pro-Arm, matching the arrow marking on the cable with the arrow marking on the robot. Refer to Fig. 6.

Caution: Be sure that the power supply is not plugged into the ac receptacle at this time. Also, check carefully to avoid shifting the Power-Supply Output Cable to the left or right as you match the pins on the Pro-Arm with the sockets of the cable connector.

SIGNAL	ROBOT CONNECTOR (INPUT)	IBM PC TYPE CABLE PORT (OUTPUT)
STROBE	1	1
GROUND	2,4,6,8,10,12,14,16	19,20,21
DATA 0	3	2
DATA 1	5	3
DATA 2	7	4
DATA 3	9	5
DATA 4	11	6
DATA 5	13	7
DATA 6	15	8
DATA 7	17	9
ERROR	18	15
ACKNOWLEDGE	19	10
BUSY	20	11

TABLE 1: Pin Configuration of 8-Bit IBM-type Cable Port

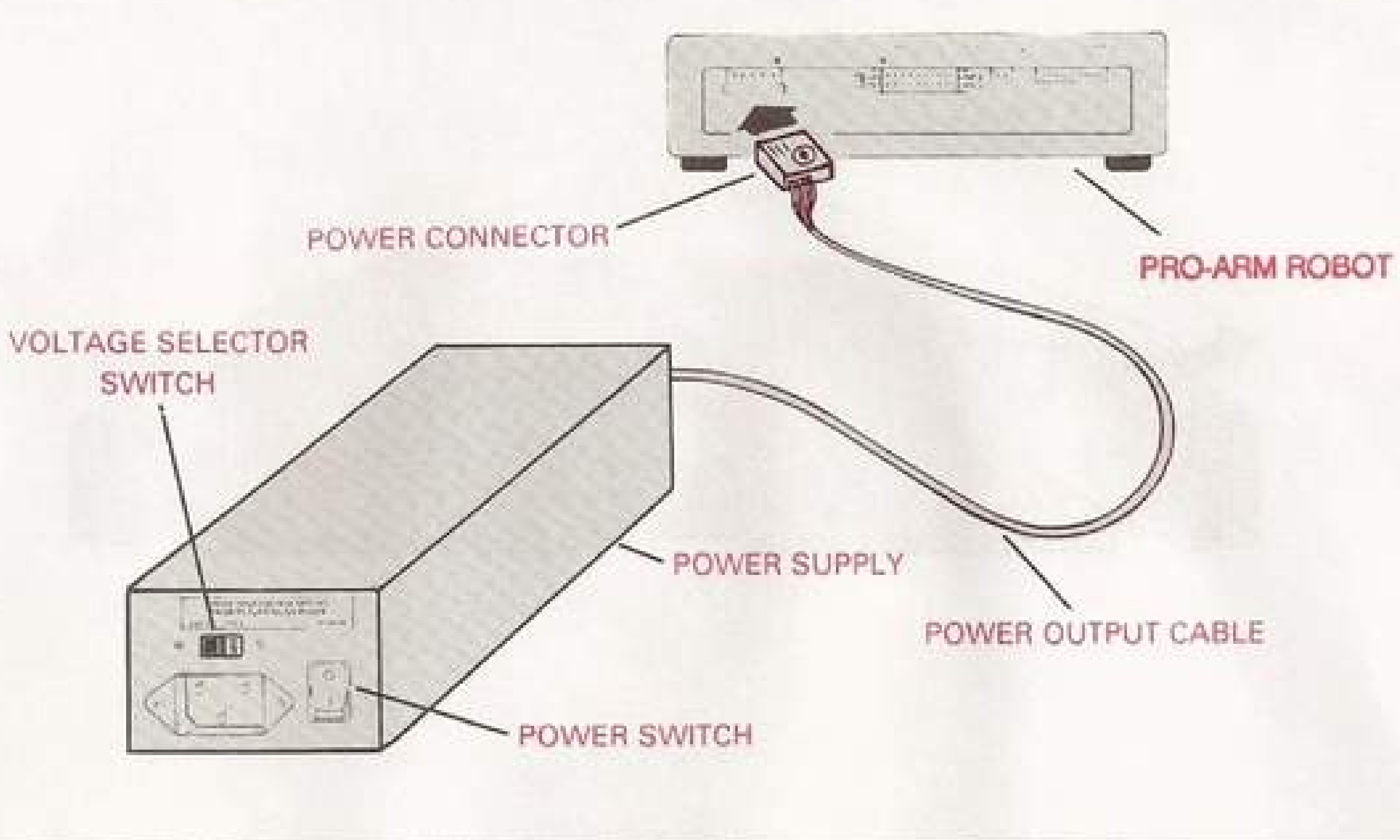


Fig. 6: Connecting the Power Supply Output Cable

____g. The power-supply input voltage is factory set at 110-V ac. If you are using the Pro-Arm on a 220-V ac power source, change the Voltage Selector Switch on the power supply to the "220 V" setting. (Refer to Fig. 7.)

____h. Insert the female end of the power cord into the connector on the power supply.

____i. Plug the male end of the power cord into a 110-V ac, 50/60 Hz wall receptacle (or 220-V ac, if alternately switched).

____j. Referring to Fig. 8, locate the Serial Port Mouse Connection on the Input/Output (I/O) adapter card installed in the computer.

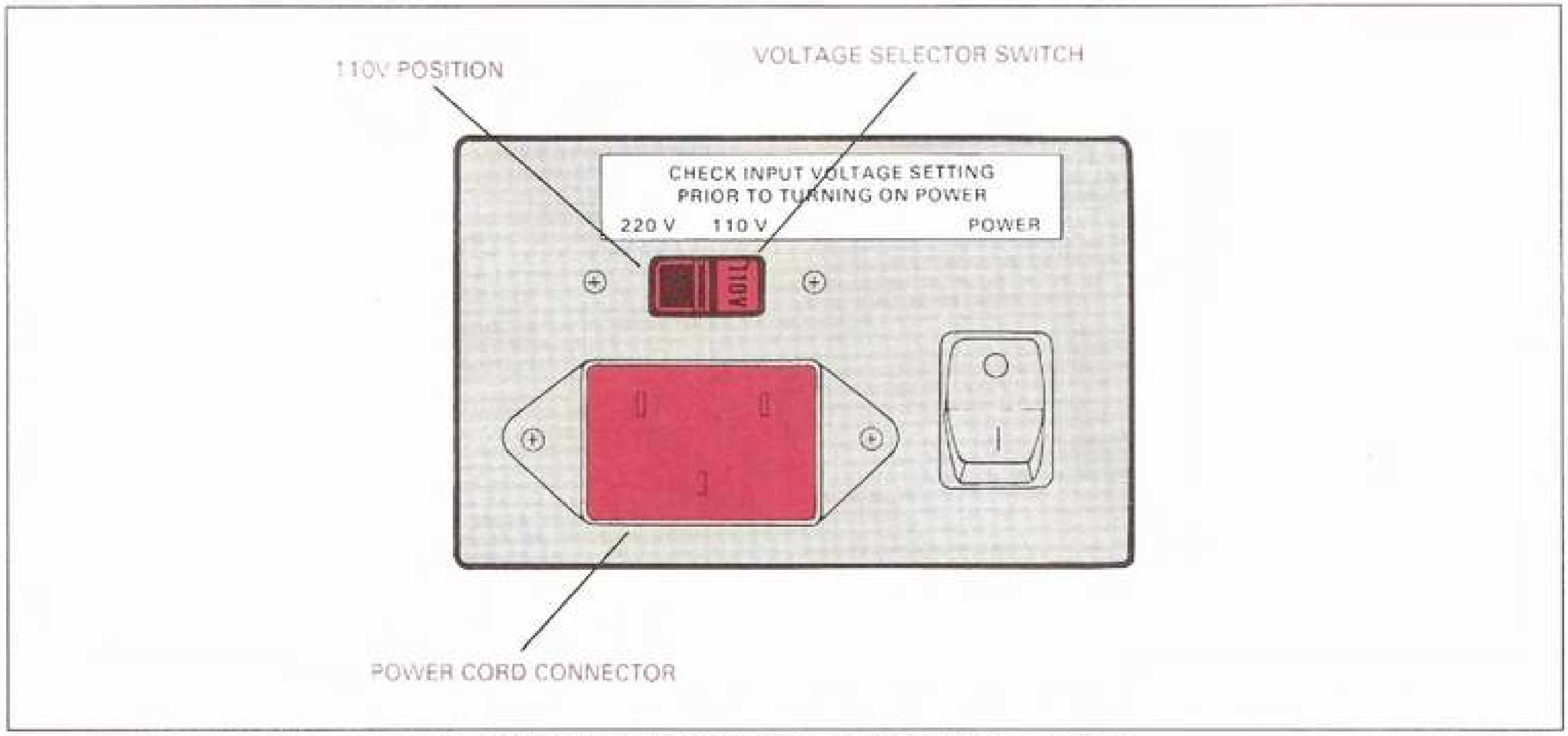


Fig. 7: Voltage Selector Switch and Power Cord Connector

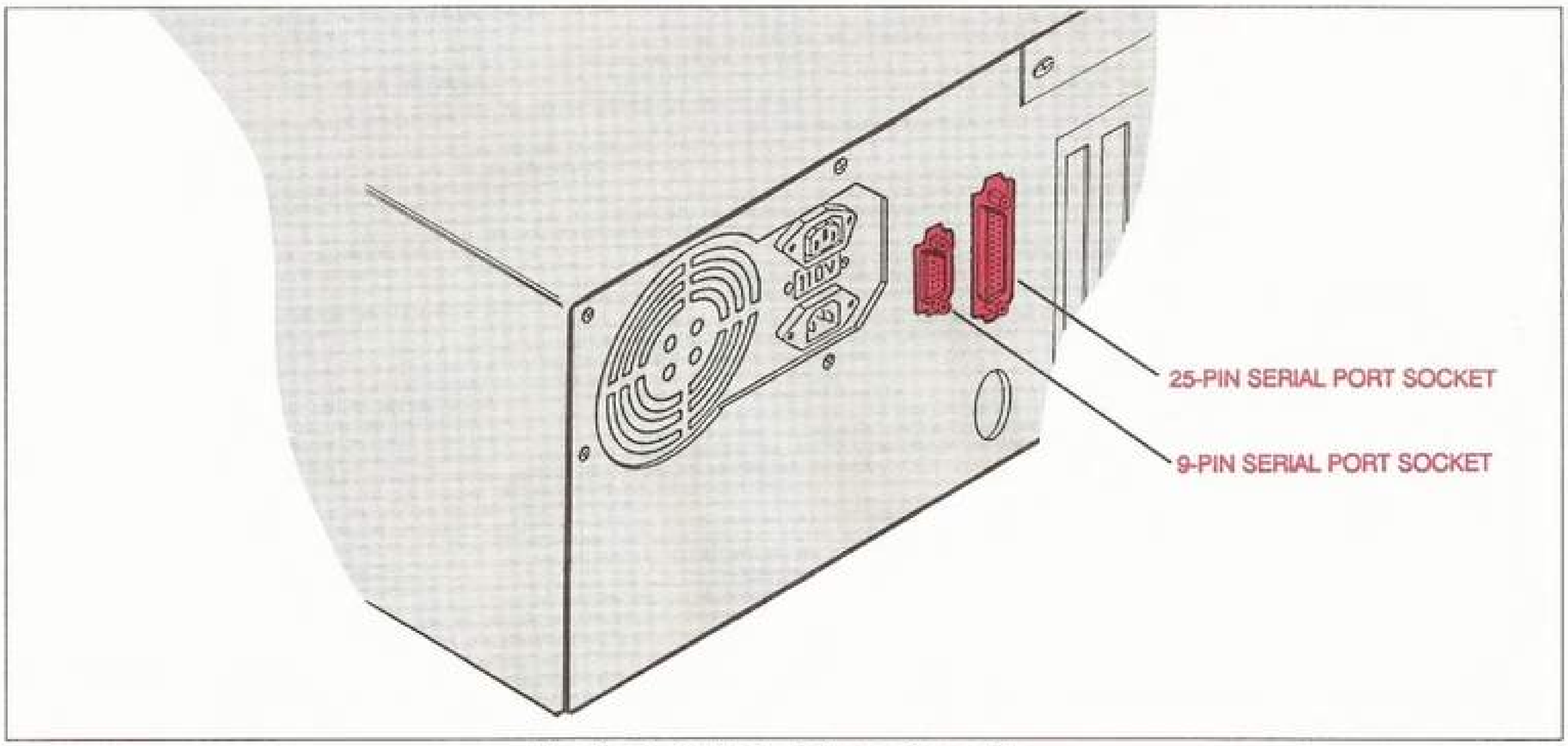


Fig. 8: The Serial Port Mouse Connection

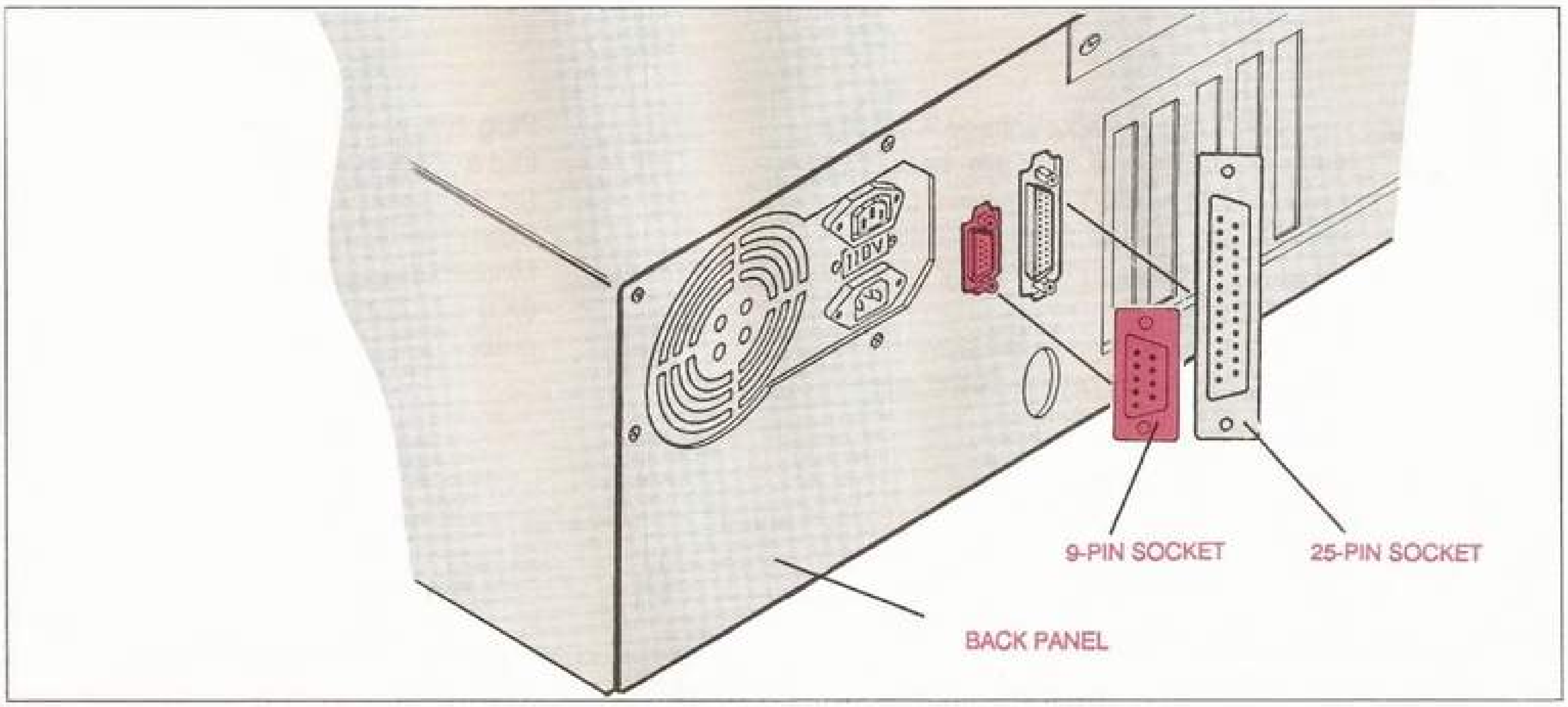


Fig. 9: The Serial Mouse Connector Socket

Note:You will have to look through the slots on the computer's back panel to find it. On most IBM-compatible computers, it is a female 9-pin D-Shell connector socket, but on some models it is a female 25-pin D-Shell connector socket which is mounted in an adjacent opening in the computer's back panel. Both types are shown in Figs. 8 and 9.

Note: The pointing device which comes with the RS-2200 P/A Pro-Arm robot is a three-button, serial mouse. It will make the operation of the Pro-Arm/computer interface (the hardware connection between the Pro-Arm robot and the computer) a much easier exercise than it would be if the student had to learn a programming language before being able to supply instructions to the system. Installing the mouse is fairly simple, and we will go ahead and do it now, even though the mouse is not required to conduct the two self-tests.

____k. Plug the 9-pin, male, D-Shell connector plug (mounted at the end of the mouse cable) through the proper slot in the rear panel of the computer and into the 9-pin, female, D-Shell connector socket on the adapter card.

Note: If your computer is using a 25-pin, D-Shell connector socket instead of the 9-pin type, you will need to use the 9 to 25-pin, D-Shell adapter supplied with the mouse to make this connection. (See Fig. 10.)

____l. Tighten the screws on the male, D-Shell connector plug so that the connection between the mouse and the computer cannot disengage.

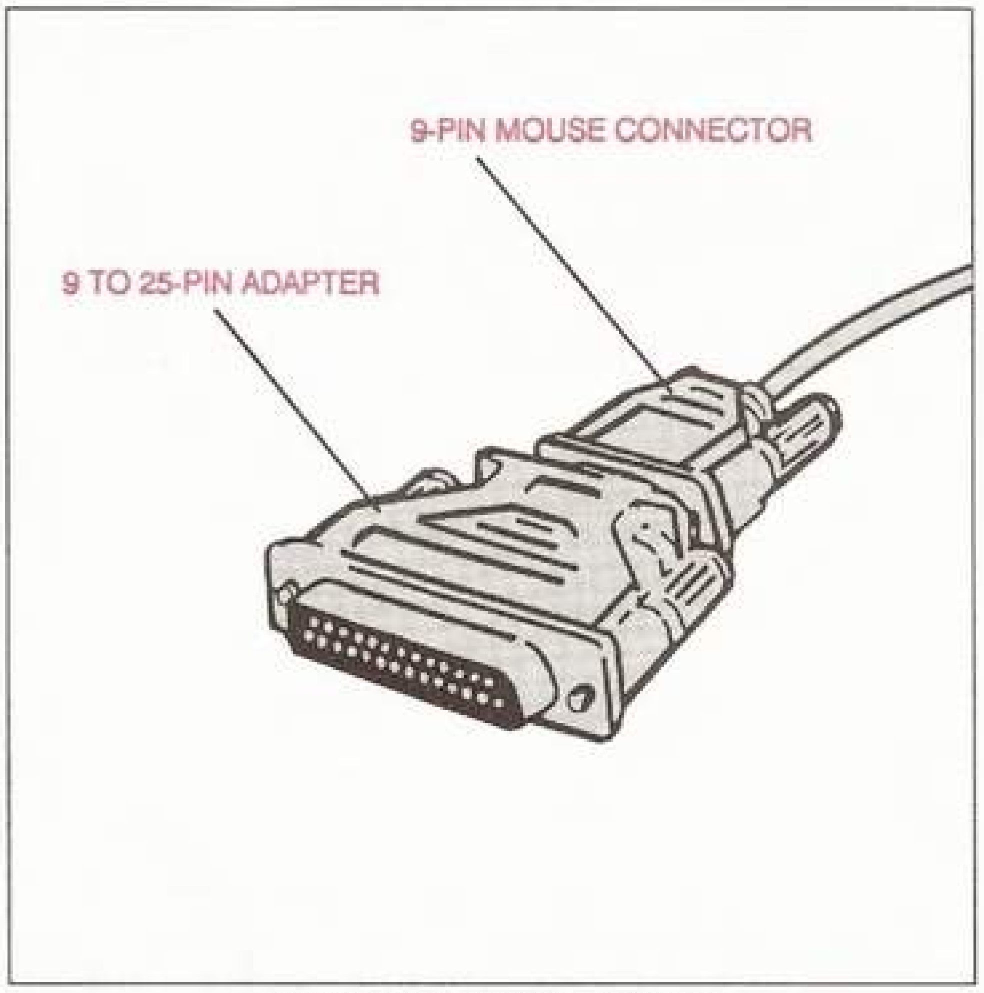


Fig. 10: The D-Shell Adapter

Note: The mouse pointer operates by the use of a free-spinning, rubber ball which is located underneath the mouse's plastic housing. The ball itself is covered with a rubber material that depends on friction for its movement. For best results, it would be wise to avoid using a slick desk surface upon which to manipulate the mouse. The purchase of a mouse pad, composed of sponge rubber and nylon material, will result in optimum mouse performance, as well as protection of both the desk top and the mouse from scratches and dirt.

___m. Place the mouse on the pad (or other suitable surface) where its operation will feel natural and comfortable, as shown in Fig. 11.

___n. This completes Lab Exploration 3, "Hardware." Please have your instructor initial your Knowledge Transfer Guide.

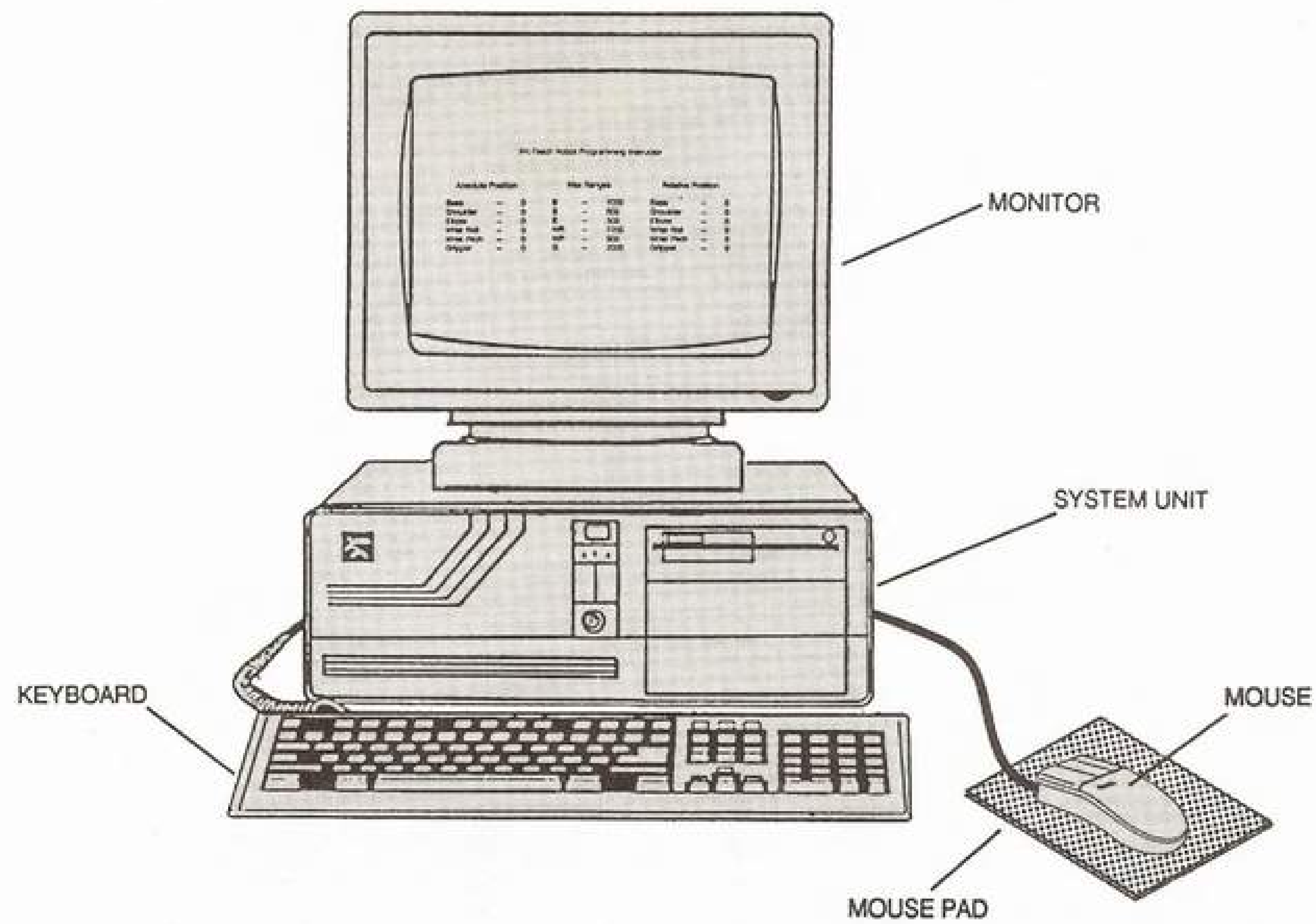


Fig. 11: The Mouse Installed

PROGRAM DISKS AND DOS

Apart from conducting the self-tests, all of the Pro-Arm activities will require the use of one of the Program Disks (software). Not only does the Pro-Arm require programs to tell it what tasks to perform, but so does the computer. The software needed to operate the RS-2200 P/A Pro-Arm Robotics system is located on the Program Disks. The exceptions are the ROM self-tests in the Pro-Arm and the ROM BIOS ***(Basic Input/Output System)*** programs in the computer itself, which tell the computer how to interpret and execute the various commands throughout the system.

One of the Program Disks, the Pro-Arm Teach Program, contains the MARCRAFT PK TEACH software package, which allows the use of the mouse in writing, editing, and running programs for the Pro-Arm. The mouse can also be used as a Teach Pendent. That is, a control program can be developed, which uses the mouse to first move the Pro-Arm to a specific position, records that position in memory, and then moves to a new position, records that position, and continues this process until the entire program has been saved.

The other Program Disk included in the RS-2200 P/A Pro-Arm system is the Pro-Arm Program Disk written in IBM-PC format. It contains several ***BASIC*** language programs that are used in conjunction with the programming aspects of this Knowledge Transfer Project. Some rudimentary knowledge of ***BASIC*** is helpful when using this disk.

Depending on the computer being used with your RS-2200 Pro-Arm system, a version of DOS ***(Disk Operating System)*** will have to be loaded prior to using either one of the Program Disks just described. The act of loading DOS into a IBM-type computer is called ***"Booting Up"*** If the computer is not yet powered, we refer to the boot-up procedure as a ***"Cold Boot."*** If power has already been applied to the computer, we refer to the boot-up procedure as a ***"Warm Boot,"*** or in some cases, a ***"System Reset."*** Since DOS is actually a special set of programs (software), it also is contained on a disk. The procedure which follows will serve equally well when handling any floppy disk, but for the sake of simplicity, we will assume that the computer is turned off to begin with, and that the system being used contains only one floppy drive (the ***A*** drive).

After reviewing the Disk-Handling Instructions in Section I, follow this procedure to properly load DOS (boot up the system), and then insert the Pro-Arm Program Disk of your choice.

LAB EXPLORATION 4

As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

____4. DOS AND PROGRAM DISK USE

Purpose: This exploration will familiarize the student with the proper boot-up procedure for an IBM-compatible computer and the preparation of the system for use with the Pro-Arm software.

Equipment: IBM-Compatible Computer System, DOS System Disk, Pro-Arm BASIC Program Disk, and the MARCRAFT PK TEACH Disk.

____a. Check all the connections to the computer, including the power cord, to be sure that everything is ready.

____b. Make sure that the original DOS System Disk and both of the RS-2200 Pro-arm disks have been backed up, and that you are using the working copies.

____c. Locate the working copy of the DOS System Disk, and remove it from its protective sleeve. Grasp the disk, as shown in Fig. 12, by the label on the disk.

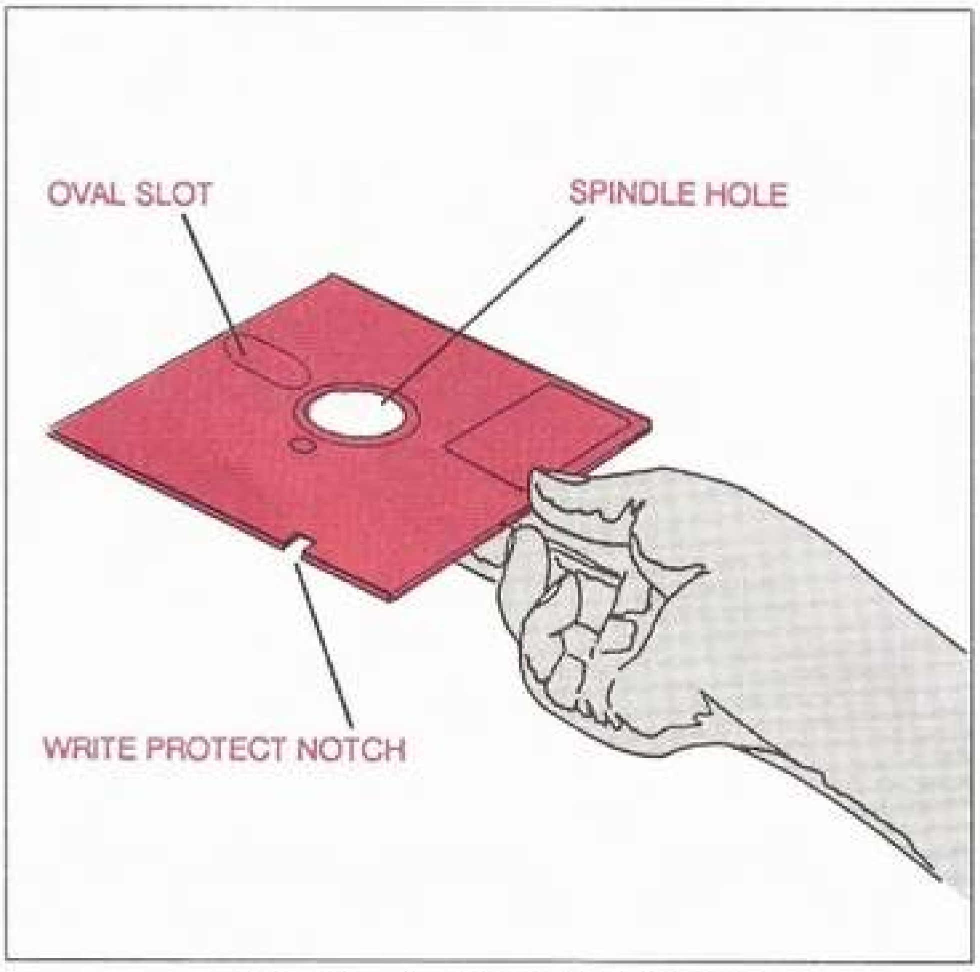


Fig. 12: Grasping a Floppy Disk

____d. Carefully insert the floppy disk into the slot of the drive door, as shown in Fig. 13. The edge closest to the exposed disk surfaces goes in first. There are two small notches which flank the oval openings in the disk envelope. *The Read/Write **(R/W)** heads of the disk drive touch the surface of the floppy disk through these openings.*

____e. Slide the disk in gently and smoothly until it clears the spindle lock lever. *Caution: It is at this step that many disks are damaged due to carelessness! Forcing the disk roughly against any resistance may bend or scratch the disk, making it totally unusable.*

____f. Once the disk has cleared the spindle lock lever, push the lever into its locked position, as depicted in Fig. 14. If there is any hint of jamming, do not force it down. If the disk is positioned properly, the lever will lock easily. *Note: The spindle lock lever lowers the drive-motor spindle into the spindle hole in the center of the disk to hold it firmly during rotation.*

____g. Turn the power to the computer on. Most older models have the power switch located on the right rear of the system unit, while newer computers have push-button power switches located on the front panel.

*Note: The computer should give audible and visual indications of activity. The disk drive should come to life and the monitor should provide information as the system checks itself out. Finally, the familiar "**A>**" prompt should appear on the screen, indicating that the system is ready for use and waiting for your response.*

*Caution: If the proper indication does not appear on the monitor, **TURN OFF THE COMPUTER IMMEDIATELY**. Confirm your equipment connections, and then refer to the **TROUBLESHOOTING** section in the back of this manual.*

____h. Disengage the spindle lock lever and remove the DOS System Disk, grasping it as described in **Step c**, above, and place it back into its protective sleeve.

____i. Determine which of the RS-2200 Program Disks you wish to use and follow **Steps d through** f. Type in the commands needed to access the disk and run your program.

____j. After all work is done, remove any disk still in the disk drive and place it back into its protective sleeve.

____k. Turn off the power to the computer.

____l. This completes Lab Exploration 4, "DOS and Program Disk Use," and the PRO-ARM INSTALLATION activities. Have your instructor initial your Knowledge Transfer Guide.

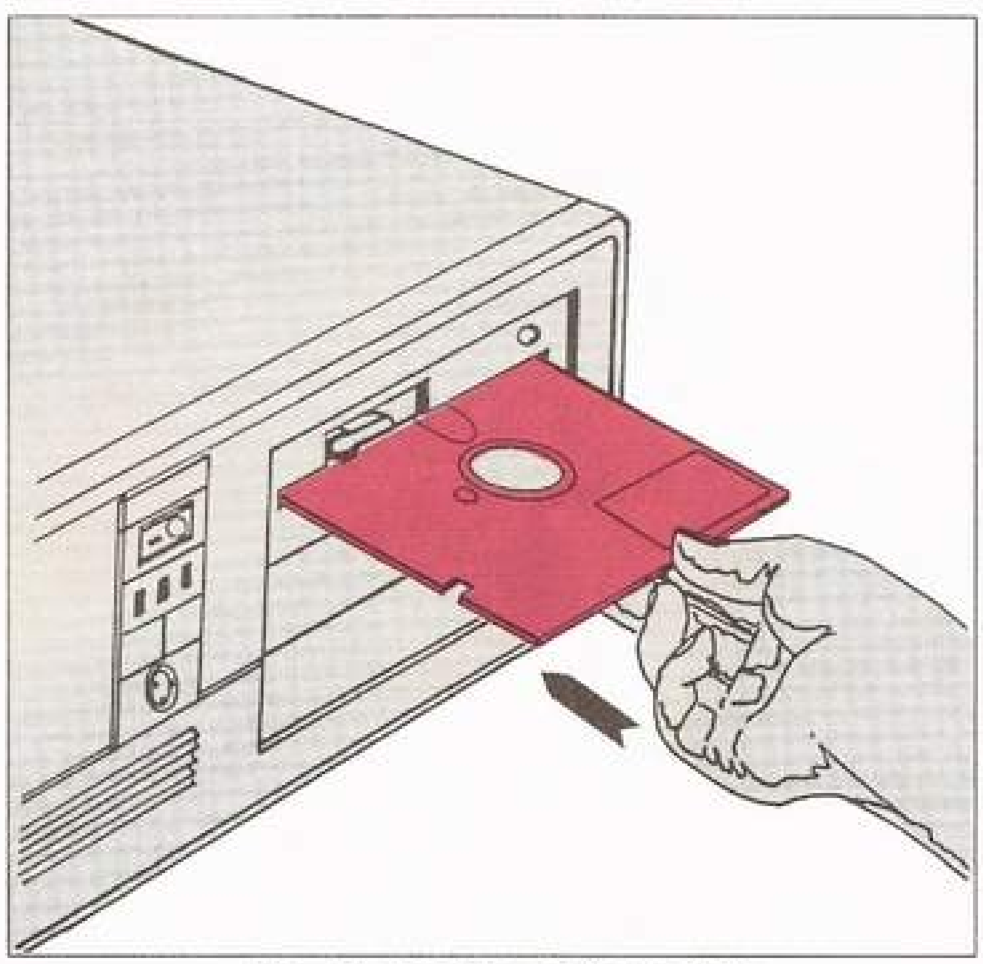


Fig. 13: Inserting a Floppy Disk

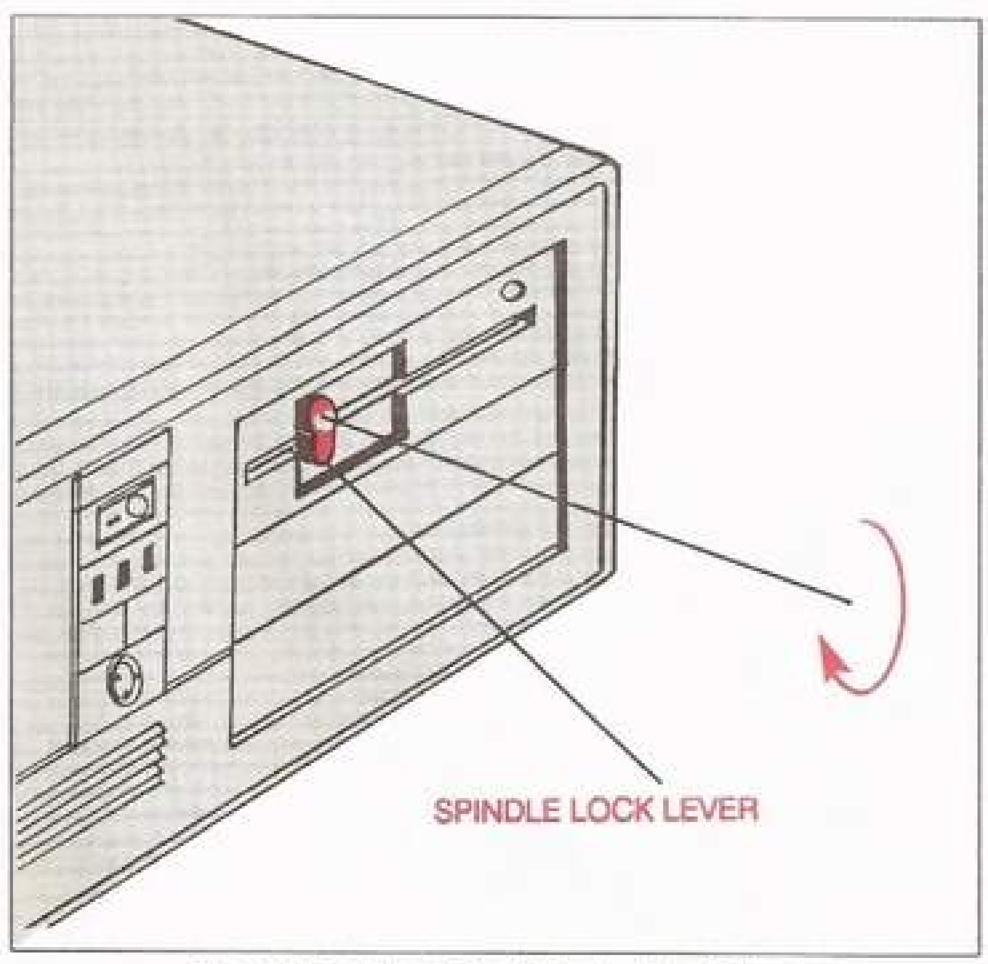


Fig. 14: Locking the Spindle Lock Lever

OPERATIONAL TESTING

DISCUSSION

Although it's not absolutely necessary to do it each time that the Pro-Arm system is going to be used, it's a good idea to give The system a preliminary check first. This will ensure that its electronic circuits and mechanical parts are all functioning properly before any teaching sessions begin.

In order to make this task as easy as possible, there are two test programs which have been built into the ROM ***(Read-Only Memory)*** IC that is plugged into socket "*D6*" on the Pro-Arm's PCB ***(Printed-Circuit Board)***. (Refer to the Printed Circuit Board Wiring Photograph on Page 162.)

The two self-test programs contain the instructions which are sent to the Pro-Arm, in a predetermined sequence, to tell it how to perform the necessary movements for testing. If the Pro-Arm does everything correctly, it will have demonstrated its fitness for further use.

The material on OPERATIONAL TESTING will be presented in the following sequence:

1. Control Panel
2. Pro-Arm Power-Up Procedure
3. ROM Self-Tests

CONTROL PANEL

Before we can instruct the Pro-Arm to execute the two test programs, we need to look more closely at the Pro-Arm's front control panel. As shown in Fig. 15, the Front Control Panel features two LED ***(Light-Emitting Diode)*** indicator lamps, one red and one green, plus two push-button switches, one red and one yellow.

The green LED is the ***Power Indicator***. It should be glowing brightly whenever the power supply is switched on and the power output cable is properly connected, as previously discussed.

The red LED is the ***Error Indicator***. It should glow whenever an incorrect command is sent to the Pro-Arm from the computer. It is also activated whenever the Pro-Arm exceeds its operating range (with the "***limit control***" function in operation).

The red pushbutton is the ***Test Switch***. It is used to activate the two internal self-test programs.

The yellow pushbutton is the ***Reset Switch***. It is used to reset the entire RS-2200 system and to erase any program currently in the Pro-Arm's RAM ***(Random-Access Memory)*** IC (located in socket "*D3*" of the Pro-Arm's PCB, shown in Fig. 100 on Page 162). The two test programs stored in the ROM IC that is plugged into socket "*D6*" are permanent, and cannot be erased.

We also need to become familiar with the proper steps in applying power to the Pro-Arm. After refreshing your memory by reviewing the Safety Precautions previously covered in Section I, follow the next sequence of steps each time you apply power.

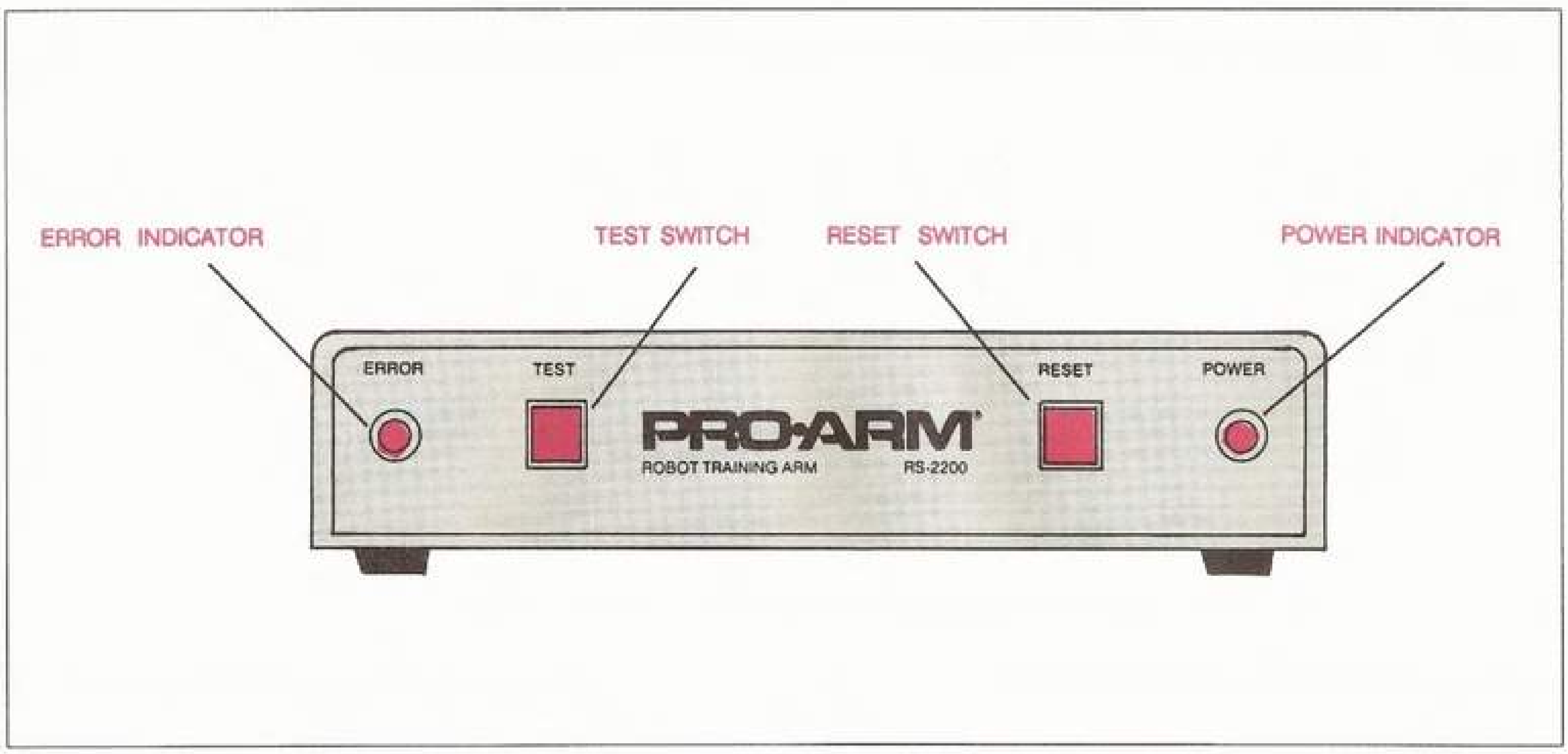


Fig. 15: The Front Control Panel

PRO-ARM POWER-UP PROCEDURE

Commit the following procedure to memory as soon as possible. You will repeat it many times during the course of this project.

As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed. Refer to Fig. 16.

LAB EXPLORATION 5

____5. POWER UP

Purpose: This procedure will familiarize the student with the proper steps that should be followed whenever applying power to the Pro-Arm.

Equipment: Pro-Arm robot, Pro-Arm Power Supply, Pro-Arm Power Cord, and the Pro-Arm Grid Sheet.

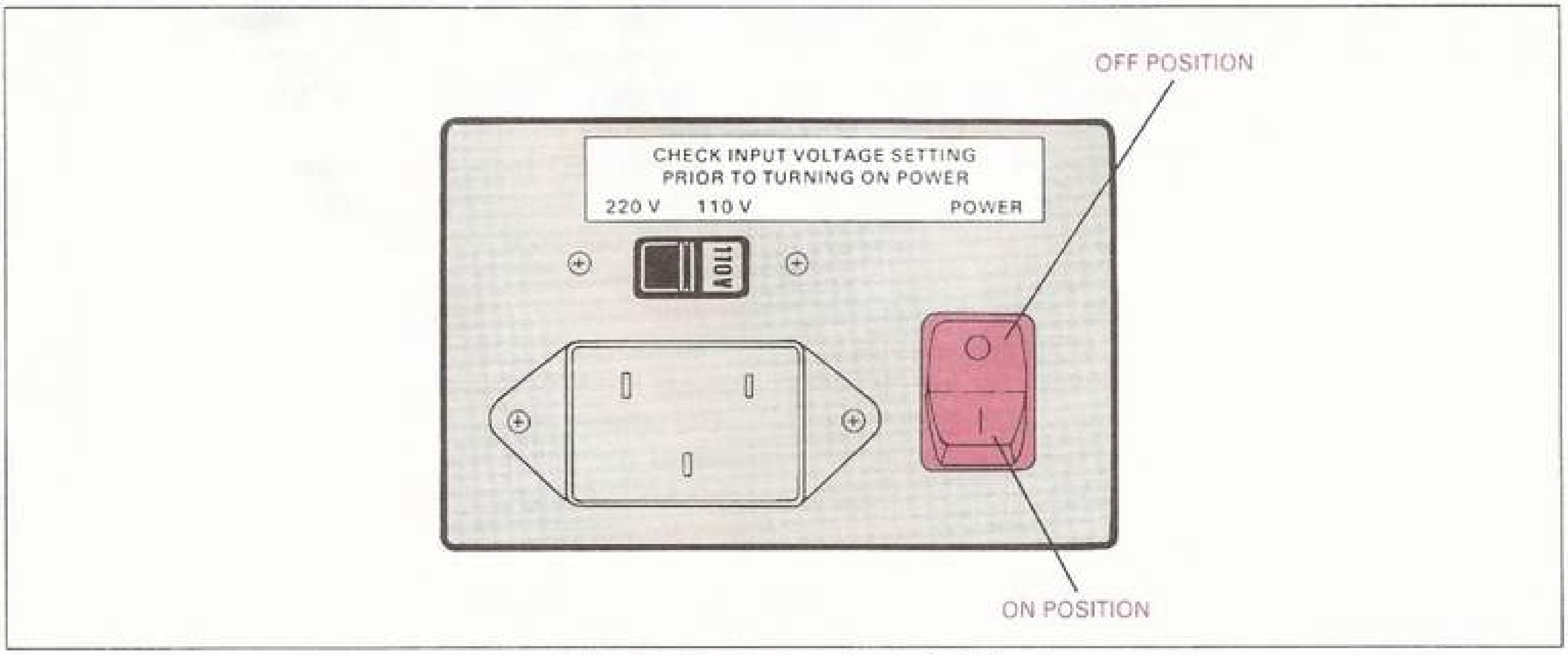


Fig. 16: The Pro-Arm Power Switch

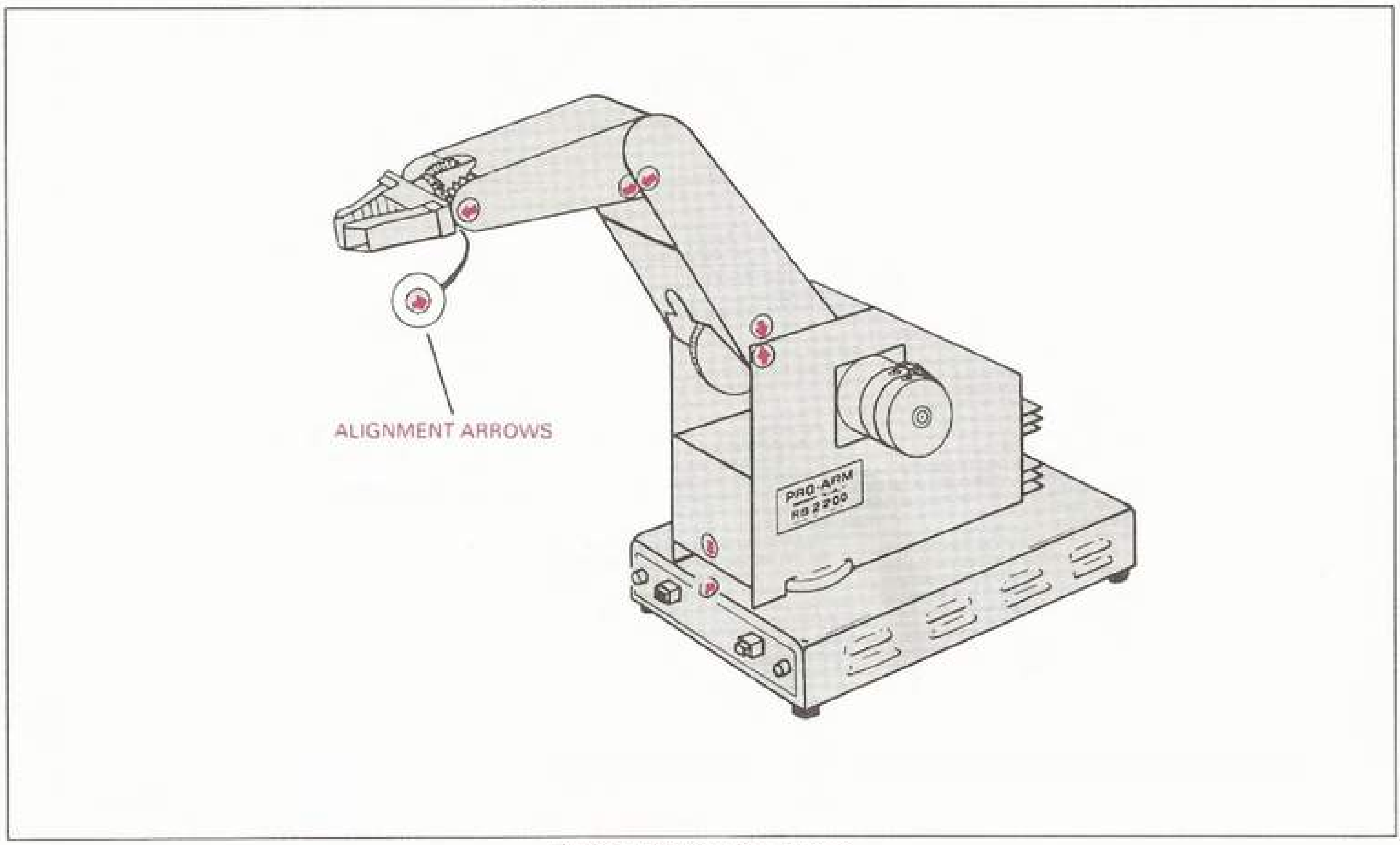


Fig. 17: Aligning the Arrows

____a. Make sure the Pro-Arm system has been properly installed as demonstrated in the previous Lab Explorations.

____b. Place the Power Switch on the power supply to the **ON** ("I") position.

Caution: If you hear a loud buzzing or humming sound coming from the power supply, turn it off immediately! This sound indicates an overload condition or an improper setting of the Voltage Selector Switch. Refer to the ***TROUBLESHOOTING*** *portion of Section VI.*

____c. Check to be sure that the green LED Power Indicator is glowing brightly.

Note: If the gripper is open when power is first applied, it will automatically close and actuate its limit switch.

Caution: If an open gripper does not close, or the green LED Power Indicator is glowing faintly or not at all, turn off the power supply immediately. Again, refer to the ***TROUBLESHOOTING*** *portion of Section VI.*

____d. After all activities have been completed, or when the Pro-Arm is not being used for a certain period of time, turn the power-supply switch to the **OFF** ("O") position."

____e. This completes Lab Exploration 5, "Power Up." Have your instructor initial your Knowledge Transfer Guide.

ROM SELF-TESTS

Each time before using the Pro-Arm system, you must be sure that the arm is in its *"HOME"* or *"NEST"* position. This means that each axis must be aligned to its *"ZERO"* position. When the arrows on each side of a joint are aligned, as shown in Fig. 17, the axis is in its zero position. However, before you can learn how to zero position the axes, you must first learn how to move each axis.

In the Pro-Arm, as in most robots, the energy developed by an actuator ***(stepping motor)*** is transferred by mechanical means to an arm axis where work is to be done. This mechanical transfer of energy is accomplished through the use of levers, chains, cables, pulleys, belts, cams, and gears.

The answers and typical test results for the following explorations are shown in the **ANSWERS** section on Pages 167–171. (If these pages have been removed, have your instructor approve your test results.) If you do not obtain similar results, ***TURN OFF THE PRO-ARM POWER SUPPLY IMMEDIATELY;*** then confirm your connections.

LAB EXPLORATION 6

In the following Lab Exploration, you will learn how to use the ***RESET*** and ***TEST*** buttons on the Pro-Arm to sequentially move each axis. You will also have the opportunity to identify the stepping motor and the linkage associated with each axis.

As you proceed through the following steps, refer to Fig. 18 to identify the various motors, gears, and linkage. Be sure to complete the blank boxes in Table 2.

____6. ARM AND AXIS FAMILIARIZATION

Purpose: In this exploration the student will use the Pro-Arm Test and Reset buttons to move an axis any desired distance and direction—within limits, name the devices used to provide linkage between a power source and an axis, identify the corresponding stepping motor, axis, and linkage used to move any part of the Pro-Arm, and explain the mechanics of the wrist *"roll"* and *"pitch"* movement and the use of a cable and tension springs to control the gripper.

Equipment: Pro-Arm Robot, Pro-Arm Power Supply, Pro-Arm Power Cord, Pro-Arm Grid Sheet, Knowledge Transfer Manual, Pencil, and a Measurement Scale.

As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

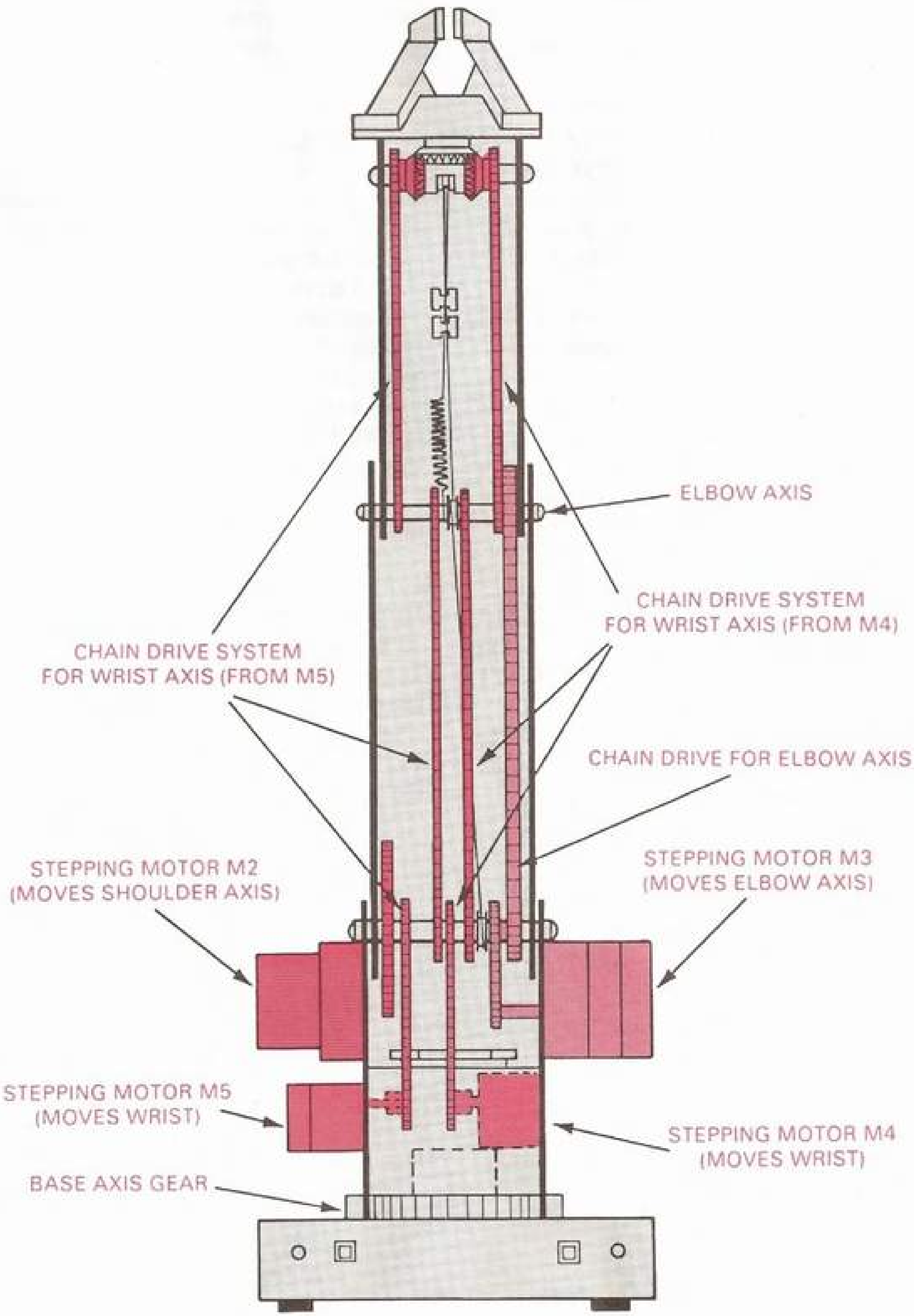


Fig. 18: The Pro-Arm's Motors, Gears, and Linkage

To check the results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____a. Perform the Pro-Arm Power-Up Procedure.

____b. Be sure that the Pro-Arm is free of any obstruction for at least 16 inches on each side and across the front of the grid sheet.

____c. Compare the positioning of each axis with respect to Fig. 17. Although it is not necessary that the zero-position arrows be perfectly aligned, they should be relatively close to alignment. If not, be sure to check with your instructor before proceeding.

____d. Depress the **RESET** button and then depress the **TEST** button, keeping both buttons pressed simultaneously. Then, first release the **RESET** button and then the **TEST** button. Summary:

(1) Depress **RESET** button.
(2) Depress **TEST** button.
(3) Release **RESET** button.
(4) Release **TEST** button.

____e. Look down through the opening between the top of the forearm and the body, as shown in Fig. 19. Stepping motor **M1** should be visible. This motor is responsible for moving the Pro-Arm along its **BASE** axis, but because of its location within the body, its action cannot be directly observed.

____f. Observe that one of the **BASE** axis gears for **M1** is located on top of the base unit and may be seen from the front of the robot, as illustrated in Fig. 18. A small gear on the end of the motor shaft is driven against this large gear to provide body movement (arm swing). The small gear is well hidden within the body and cannot readily be seen.

____g. Depress the **TEST** button for a second or two. This should cause the body to rotate along the **BASE** axis. In which direction (clockwise or counterclockwise) did it move?

____

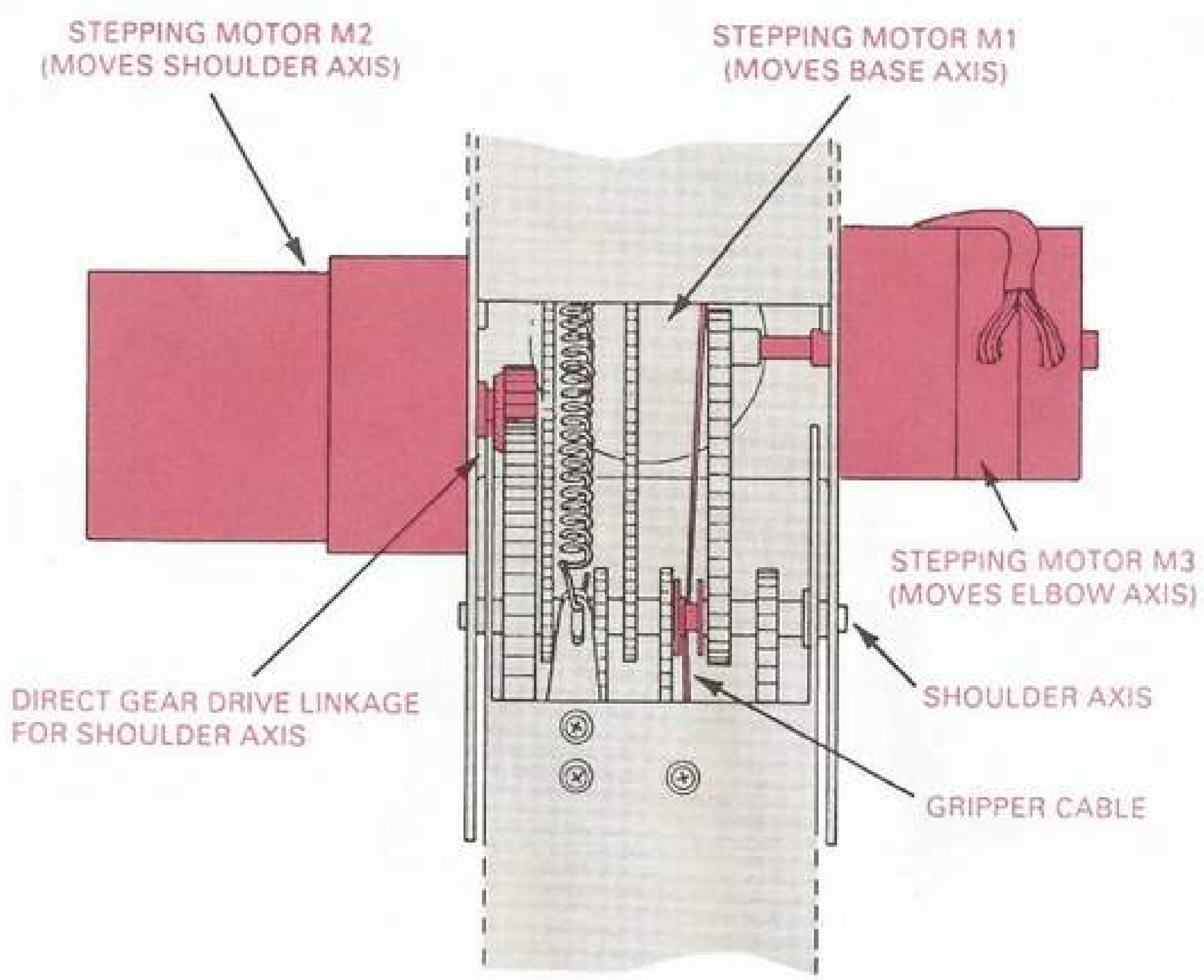


Fig. 19: View from Top of Forearm

___h. Once more, depress the **TEST** button for a second or two. Again, the body should move along the **BASE** axis. In which direction did it move this time?

____

___i. Locate the large stepping motor on the left side of the Pro-Arm. This is motor **M2** and it is used to move the forearm around the **SHOULDER** axis. Its linkage to the forearm can be observed by again looking down through the opening between the top of the forearm and the body (Fig. 19). This is also direct gear-drive linkage, as in motor **M1,** since the small gear on the end of the motor shaft drives the large gear fastened to the side of the forearm.

___j. Depress the **TEST** button for a second or two for the third and fourth times. Notice that the shoulder now moves. In which direction does the shoulder first move?

____

In which direction does the shoulder move next?

____

___k. Observe that the stepping motor located on the outside right of the Pro-Arm's body is used to move the upper arm.

*Note: This is motor **M3**, and by looking through the openings on the underside of the arm, you can see the chain drive system used to move the forearm. While completing the next step, observe the movement of the M3 drive gear and the chain linkage.*

___l. Depress the **TEST** button for a second or two for the fifth and sixth times. Which axis moves as motor **M3** is activated?

____

In which direction did the forearm first move?

____

In which direction did the forearm move the second time?

____

___m. Locate the two stepping motors required to move the **WRIST** in a **"wrist roll"** or **"wrist pitch"** manner. These are the **M4** and **M5** motors. Notice that **M4** is mounted on the right, inside of the body, while **M5** is mounted on the left, outside of the body. Notice also how **M4** and **M5** are aligned with each other. While completing the next step, observe the movement of the pair of chains driven by **M4** and **M5**.

___n. Depress the **TEST** button for a second or two for the seventh and eighth times to move the wrist. Did the wrist **"roll"** or **"pitch?"**

____

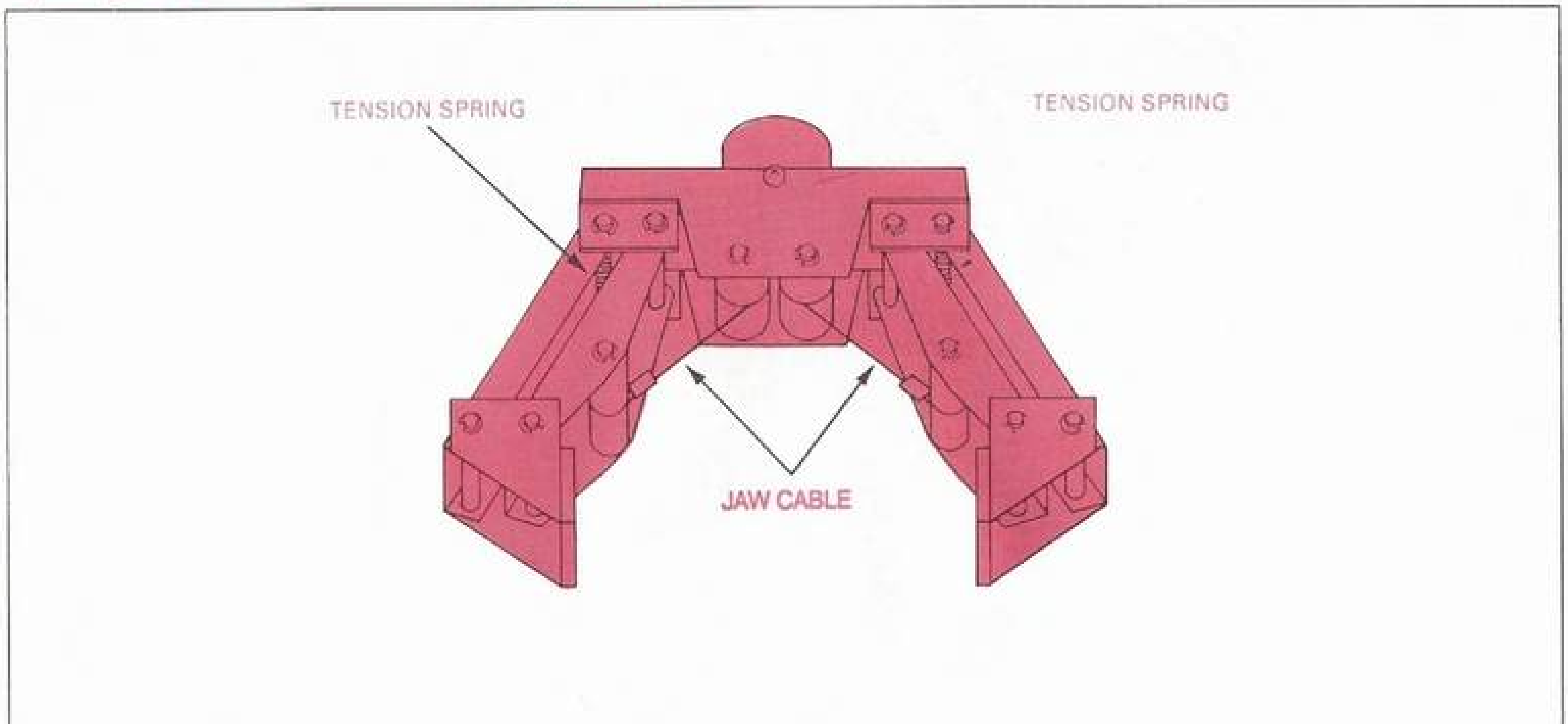


Fig. 20: The Pro-Arm's Gripper Assembly

In which direction did the wrist first move?

____

In which direction did the wrist move the second time?

____

Did both chains move in the same or opposite directions?

____

____o. Depress the **TEST** button for a second or two for the ninth and tenth times to move the wrist again. This time did the wrist "roll' or "pitch?"

____

In which direction did the wrist first move?

____

In which direction did the wrist move the second time?

____

Did both chains move in the same or opposite direction?

____

____p. Notice that the **GRIPPER** is controlled by stepping motor **M6**, which is mounted inside the body behind **M4**, and hidden from view. The gripper is connected to **M6** by the use of a small, steel cable, which is allowed to relax when **M6** turns in one direction, opening the gripper. When **M6** turns in the opposite direction, the cable is tightened and the gripper is pulled closed. See Fig. 19 and Fig. 20.

____q. Depress the **TEST** button for the eleventh time and continue holding it down until the gripper is fully open. There may be a slight delay before the gripper begins to open. You should be able to hear when **M6** stops turning, indicating that the gripper has fully opened. Use a scale to measure and record the maximum opening of the gripper.

Maximum gripper opening = ________________

*Note: If your answer is significantly less than the value given in the **ANSWERS** section, the Gripper Motor Limit Switch may need to be recalibrated. See the Calibration Procedure in the **TROUBLESHOOTING** section.*

TEST BUTTON DEPRESSIONS	MOTOR NO.	ENTITY	DIRECTIONS OF MOVEMENT	AXIS ACTUATED
1	M1	BODY		
2	M1	BODY		
3	M2	UPPER BODY		
4	M2	UPPER BODY		
5	M3	FOREARM		
6	M3	FOREARM		
7	M4,5	WRIST ROLL		
8	M4,5	WRIST ROLL		
9	M4,5	WRIST PITCH		
10	M4,5	WRIST PITCH		
11	M6	GRIPPER		
12	M6	GRIPPER		

Table 2: Axis Movement Sequence

____r. Depress the **TEST** button for the twelfth time and continue holding it down until the gripper is fully closed. Again, there may be a slight delay until the gripper begins to close. Listen for the audible **"click"** of the Gripper Motor Limit Switch.

____s. Turn the Pro-Arm's power supply **"OFF."**

____t. Complete Table 2 using the answers you recorded above.

____u. This completes Lab Exploration 6, "Arm and Axis Familiarization." Have your instructor initial your Knowledge Transfer Guide.

LAB EXPLORATION 7

You will now learn how to properly position each axis of the Pro-Arm robot so that the arm is in tł ***"HOME"*** position. This position is also known as the ***"ZERO"*** or ***"NEST"*** position. The ***HOME*** position should be the beginning and ending position of the Pro-Arm robot for all movement routines. It is necessary for the arm to be in this position before beginning any of the learning activities in the manual.

Actually, any arm position may be defined by the user as the ***HOME*** position. However, throughout this manual, ***HOME*** position refers to the arm position in which all axes are in their ***ZERO*** position with their corresponding arrows aligned, as shown in Fig. 21.

____7. ZERO POSITIONING

Purpose: This exploration will give the student practice in adjusting all of the Pro-Arm's axes to their zero positions.

Equipment: Pro-Arm Robot, Pro-Arm Power Supply, Pro-Arm Power Cord, Pro-Arm Grid Sheet, and the Knowledge Transfer Manual.

As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

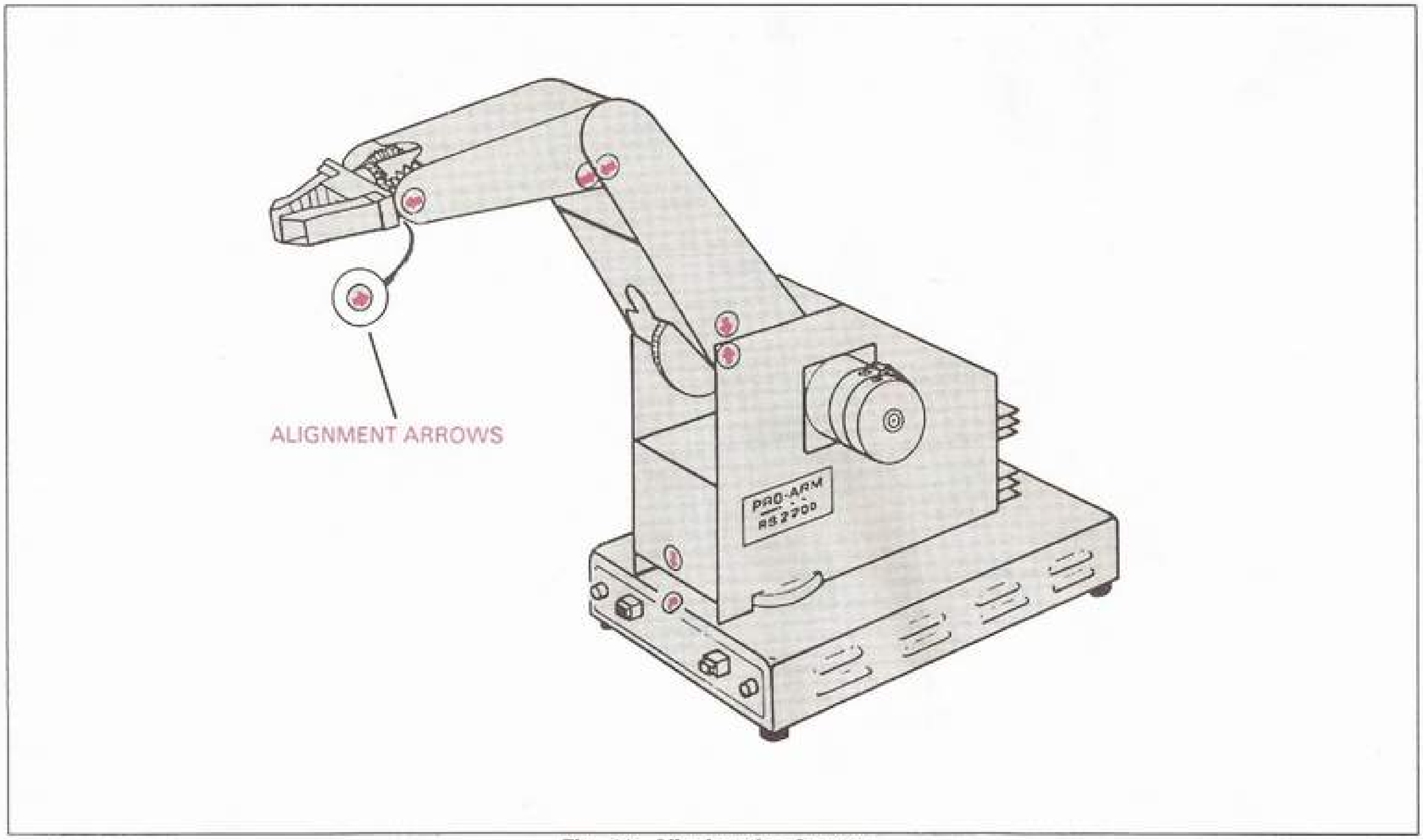


Fig. 21: Aligning the Arrows

____a. Perform the Pro-Arm Power-Up Procedure.

____b. Be sure that the Pro-Arm is free of any obstruction for at least 16 inches on each side and across the front of the grid sheet.

____c. Depress the **RESET** button and then depress the **TEST** button, keeping both buttons pressed simultaneously. Then, first release the **RESET** button and then the **TEST** button. Summary:

(1) Depress **RESET** button.
(2) Depress **TEST** button.
(3) Release **RESET** button.
(4) Release **TEST** button.

____d. Zero-position the **BASE** axis by depressing the **TEST** button to move the Pro-Arm in a counterclockwise direction until the alignment arrow on the body is slightly to the right of the arrow on the base. Again depress the **TEST** button, moving the Pro-Arm clockwise until both arrows are aligned.

Note: If you move the body too far, it will be necessary to start over. This is done by quickly depressing the test button a number of times until the body again moves in a counterclockwise direction.

____e. Zero-position the **SHOULDER** axis by depressing the **TEST** button to move the upper arm downward until the point of the arrow on the upper arm is below the point of the arrow on the body. Push the **TEST** button again, raising the upper arm until the points of both arrows are aligned.

____f. Now zero-position the **ELBOW** axis by depressing the **TEST** button until the arrow on the forearm is below the arrow on the upper arm. Push the **TEST** button again, moving the forearm upward until the points of both arrows are aligned.

____g. Zero-position the **WRIST ROLL** axis by depressing the **TEST** button to move the wrist in a clockwise direction until the arrow on the left side of the wrist is above the forearm arrow and the arrow on the right side of the wrist is below the arrow on the forearm. Again, push the **TEST** button, moving the wrist in a counterclockwise direction until the wrist is in a level position. In this position, both arrows on the wrist (one on each side) should be:

____(1) Aligned with the corresponding arrows on the forearm.
____(2) An equal distance above the corresponding arrows on the forearm.
____(3) An equal distance below the corresponding arrows on the forearm.

____h. Zero-position the **WRIST PITCH** axis by pressing the **TEST** button to move the wrist downward until both of the arrows on the wrist are below both arrows on the forearm. Press the **TEST** button again, moving the wrist upward until the arrows are aligned. If the points on the arrows on both sides of the wrist are not aligned at this time, start over with **Step c**, quickly moving through the steps to **Step h**.

____i. It should not be necessary to adjust the **GRIPPER** since it should have automatically closed when the power switch was first turned on. However, if you wish to observe the gripper in motion, simply depress the **TEST** button to open it. Then, press the **TEST** button again to close it.

Note: Listen for the audible **"click"**, *indicating that the Gripper Motor Limit Switch has been activated.*

____j. Make a final check to be sure that all of the zero-position arrows are properly aligned.

Note: It is possible that the alignment arrows may have been moved since your Pro-Arm left the factory. If you notice or suspect any obvious misalignment of these arrows, then you should remedy this problem immediately. Use your own stickers if the original arrow markers are worn out or damaged.

____k. Depress the **RESET** button to break out of the positioning procedure.

Note: You may break out of the zero-positioning procedure at any time by pushing the ***RESET*** *button.*

____l. Turn the Pro-Arm power supply "**OFF.**"

____m. This completes Lab Exploration 7, "Zero Positioning." Have your instructor initial your Knowledge Transfer Guide.

Once the Pro-Arm has been zero-positioned, it's time to move on to the second self-test contained in the internal ROM memory. This test will demonstrate the axis movement of the Pro-Arm at three different speeds.

____8. DEMO

Purpose: This exploration will allow the student to test the Pro-Arm for verification that each axis (except for the ***GRIPPER***) and its corresponding stepping motor is working properly at varying speeds.

Equipment: Pro-Arm Robot, Pro-Arm Power Supply, Pro-Arm Power Cord, Pro-Arm Grid Sheet, and the Knowledge Transfer Manual.

As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

____a. Perform the Pro-Arm Power-Up Procedure.

____b. Perform the Zero-Positioning Procedure.

____c. Be sure that the Pro-Arm is free of any obstruction for at least 16 inches on each side and across the front of the grid sheet.

____d. Depress the **RESET** button and while holding it down, also depress the **TEST** button. Then, release the **RESET** button and continue to hold down the **TEST** button until the Pro-Arm starts to move (about 5 seconds).

Note: This demonstration routine will run at three different speeds and, then, will repeat itself.

____e. After you have observed the Pro-Arm complete all the movements which are programmed in the **ROM** chip, depress the **TEST** button. The arm will stop once the program is completed. *Note: You may also stop it at any time by depressing the **RESET** button (preferably near home position).*

____f. Turn the Pro-Arm's power supply **"OFF."**

____g. This completes Lab Exploration 8, "Demo," and the OPERATIONAL TESTING activities. This also completes SECTION II of your Knowledge Transfer Project. Have your instructor initial your Knowledge Transfer Guide.

SECTION III

ROBOTICS WITH PK TEACH

DISCUSSION

This section first introduces the student to the basic principles of robotics, and then presents the MARCRAFT PK TEACH software.

Using the PK TEACH software supplied with your system, you can easily **"teach"** the Pro-Arm robot to perform a task directly from commands generated at your computer keyboard or mouse (also supplied), without having to learn a computer programming language first. Axes positions will be simultaneously displayed on your monitor and recorded in permanent memory for later recall.

At the end of this manual is a glossary containing definitions of robotics terms. When words appear in the text unaccompanied with sufficient explanations or descriptions, please turn to the glossary for further clarification.

The material will be presented in the sequence below:

1. Introduction to Robotics
2. PK TEACH Software

INTRODUCTION TO ROBOTICS

DISCUSSION

The subject of robotics is both fascinating and complicated. It would take many books to cover all the latest developments in the field, and this manual will not attempt to teach any mechanical, electronic, or microprocessor engineering. Indeed, much of the technological understanding required to appreciate the most recent advances in robotics is beyond the scope of this Knowledge Transfer Project.

In the twentieth century, man's interest in robotics has grown substantially. Evidence of this increased interest is contained in many of today's movies, books, and television programs. Many moviegoers remember the Disney classic, ***Forbidden Planet***, which introduced as its main character, ***"Robbie the Robot."*** And almost everyone is familiar with the two human-like androids, ***"R2D2"*** and ***"C3PO,"*** the main nonhuman characters around which the movie, ***Star Wars***, revolved.

Be that as it may, rather than thinking of the robot as an anthropomorphic ***(humanoid-type with intelligence, mobility, and manipulation skills)***, man-serving machine of the future, it would be more realistic to view the robot as an intelligent, man-serving device for today. While fictional concepts of robots are fine in the world of entertainment, the industrial robot of today in no way resembles a walking, talking, beeping, creeping, metal bucket from a science-fiction writer's imagination.

Modern industrial robots are a product of recent technological breakthroughs in both the mechanical and electronic sciences. In order to fully appreciate the level of achievement and advancement that is incorporated in today's industrial robotics systems, the student must be knowledgeable in mechanics, electronics, and computer science, since it is these three disciplines that have been skillfully combined to produce the modern field of robotics. It is hoped that the introduction to robotics which the student will receive from the MARCRAFT RS-2200 P/A Pro-Arm Robotics System will be the catalyst for a determined effort in all these fields of study.

EVOLUTION

What is a robot anyway? The word ***"robot"*** comes from the Czech word ***"robota"*** which means ***"serf"*** or ***"worker."*** The American Robot Industry Association ***(RIA)*** defines a robot as a "manipulator designed to move material, parts, tools, or specialized devices, through variable programmed motions for the performance of a variety of tasks."

Although much of the technology used in modern robotics systems has been recently developed (from the late 1970s), man's fascination with robots or man-like machines is historic. Even the ancient classic writing by Homer, ***The Iliad***, contains a description of two solid-gold statues which followed and helped the Greek God of the mechanical arts, Hephaestus.

In 1801, ***Joseph Jacquard*** invented a textile machine ***(The Jacquard Loom)*** which was operated by using punch cards. Many believe that his invention marked the beginnings of modern robotics. Fig. 22 pictures a Jacquard loom.

As the punch cards moved over the loom, rods, which were attached to specific yarns, fit through the holes in the cards while others remained down. In this way, the yarns making up each individual weave cycle were selected.

Then, in 1893, ***George Moore*** built what was called a "***walking locomotive.***" It was an armor-plated man-shaped figure that could walk at up to 9 mph. It used a 0.5-horsepower boiler that utilized gas for fuel. As shown in Fig. 23, the only things that kept the figure from being a totally independent unit were the gas lines attached to it. The steam vent was built into its cigar.

It wasn't until 1954 that the first programmable robot was designed by ***George Devol***. Five years later, in 1959, the ***Planet Corporation*** marketed the first commercially available robot. Then, in 1962, ***General Motors*** installed the first industrial robot on a production line. Once robots had shown their effectiveness in industrial production during that first decade (1960s), their use began to increase, with limited growth. These first robots were not mobile, and appeared to be no more than mere extensions of other types of existing automated machines.

Then, with the advent of the ***microprocessor*** and the ***microcomputer***, the growth rate in both the numbers of robots being utilized, and the various uses to which they were being put, began. Their rise was phenomenal. Recent breakthroughs in microprocessor technology have joined together with older, well-known, electronic and mechanical concepts, and has resulted in the creation of several new classes of industrial robots, operating under microprocessor control.

The industrial robots of today evolved from equipment that was originally designed to quickly accomplish specific, repetitive tasks on the production line. These types of devices, called "***hard automation,***" were basically made up of simple drive mechanisms. They were controlled not by microprocessors, but by timers, switches, cams, and various mechanical or electrical stops. But, only mechanical adjustments were possible with such devices. This meant that no deviation from one specific process could be attained using hard-automation equipment.

As more flexibility became possible (permitted by the use of speed controllers, sequencers, adjustable stops, and timers), the devices gained the ability to vary their motions. They became the forerunners of today's inductrial robots.

The addition of servo controls allowed the robotic equipment to determine the proper positioning of the end-effector by comparing known position values with actual position values ***(feedback)***. The result was a robot with greatly increased accuracy and repeatability.

The greatest advancement yet most recently came: the addition of computer control. With movements now being controlled by a microprocessor, many capabilities that weren't absolutely necessary in every industrial application were incorporated into the most expensive industrial robots. Soon, specialized robots were being manufactured, which could be tailored to do a required range of functions, at a reasonable cost.

As more powerful microcomputers are developed, the capabilities of the robots they control will increase as well. The future seems to promise continuous growth in this already explosive field.

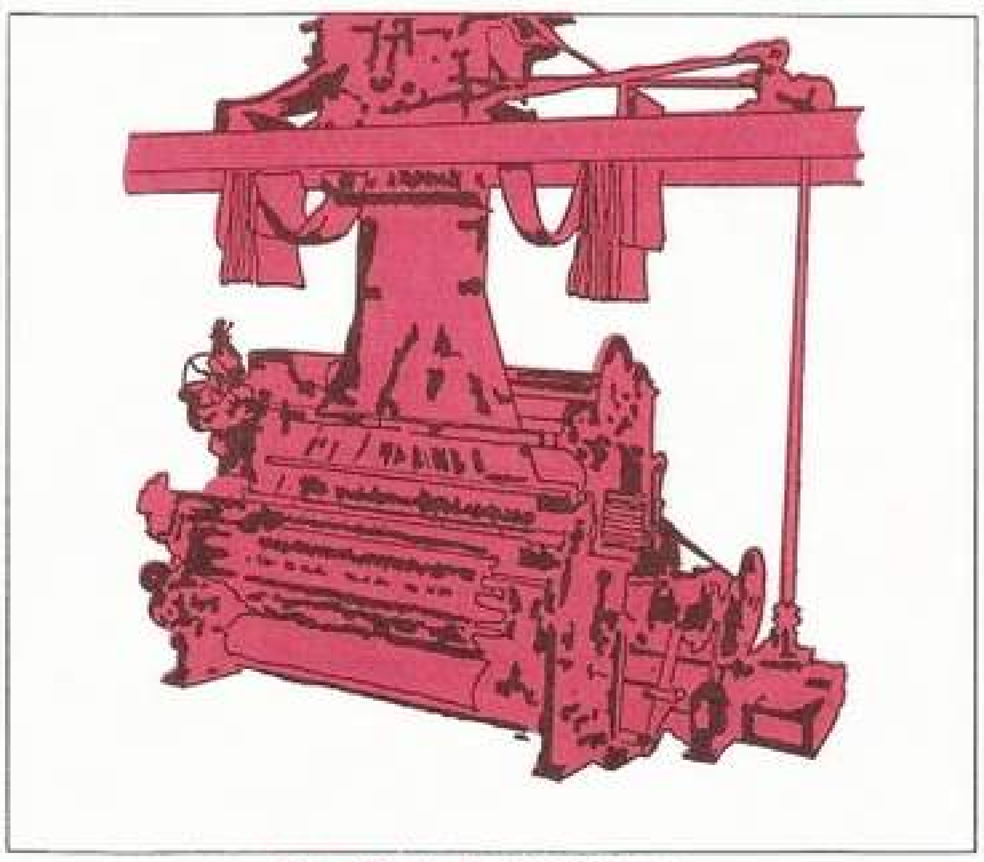


Fig. 22: A Jacquard Loom

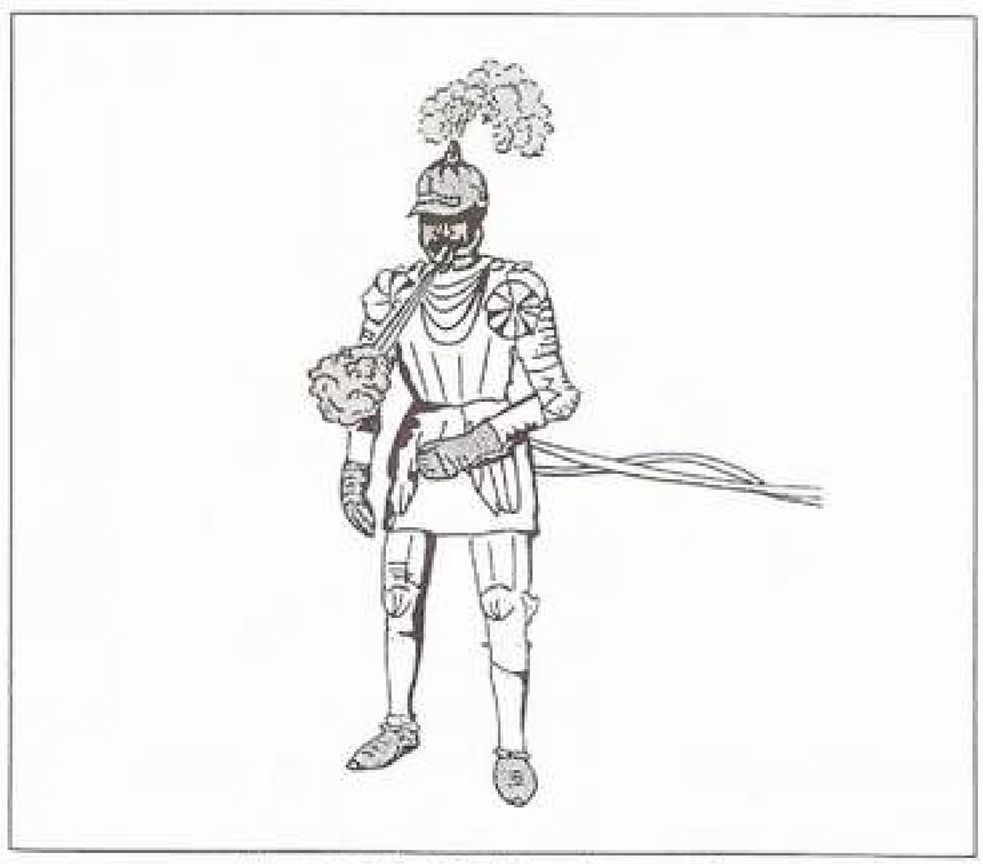


Fig. 23: The Walking Locomotive

The answers and typical test results for the following exploration are shown in the **ANSWERS** section on pages 167–171. (If these pages have been removed, have your instructor approve your test results.)

____9. ROBOT DEVELOPMENT

Purpose: This exploration will provide the student with the opportunity to review major steps in the history of robotics.

Equipment: Knowledge Transfer Manual, Pencil

In the appropriate space provided write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____a. An anthropomorphic robot would be a humanoid-type of machine with ________________, ________________, and ________________ ________.

____b. Equipment designed to perform only one type of repetitive function is called ________ ____________.

____c. The beginnings of modern robotics can be traced to a textile machine built by ____________ ____________ in ________.

____d. The Czech word, **"robota,"** means ____________ or ____________.

____e. The first industrial robot ever installed on a production line occurred in ____________.

What company was responsible for this being done?

____________ ____________.

____f. Three disciplines involved in the development of industrial robotics are ________________, ________________, and ____________ ____________.

____g. The **"walking locomotive"** was built by ____________ ____________ in ____________.

____h. The installation of servo controls allowed the use of ________________ to improve a robot's accuracy and repeatability.

____i. The most important breakthrough so far, in the evolution of industrial robotics, has been the development of both the ________________ and the ________________.

____j. Only ____________ adjustments were possible on hard-automation equipment.

____k. This completes Lab Exploration 9, "Robot Development." Have your instructor initial your Knowledge Transfer Guide.

CATEGORIES

Most all industrial robots can be classified into three general categories: ***low-technology***, ***medium-technology***, and ***high-technology***. As the robot's design technology goes higher, the list of jobs which that robot can perform gets longer. Higher-technology robots can do everything that the lower-technology types can do, and generally with larger payloads.

LOW-TECHNOLOGY ROBOTS

Two terms used to describe low-technology robots are ***"limited-sequence,"*** and ***"pick-and-place"*** (Fig. 24). As you might expect, these types of robots are the simplest in design and are classed as ***"nonservo"*** types.

By making small adjustments in the end stops, some variations in the amount of travel along the robot's axes can be achieved. Also, the speed of travel can be altered. It is even possible to change the movement pattern sequence by making slight mechanical modifications. Therefore, as limited as these simple types of robots are, they are certainly more sophisticated than hard automation.

Some distinguishing characteristics of low-technology industrial robots are:

1. Two to four nonservo-type degrees of freedom.
2. Each axis can assume only two positions.
3. Maximum distance of travel for each axis is fairly restricted.
4. Maximum payload varies between 5.3 ounces to 30 pounds.
5. Faster cycle time than any other class of robot.
6. A very high degree of accuracy and repeatability between +/-0.025 mm and +/-0.2 mm typical, with some units capable of +/-0.01 mm.
7. Low-cost pneumatic actuation normally is used since low-technology robots manipulate light payloads at high speed.
8. Controllers are simple because of the limited-sequence nature of most low-technology robots. They mainly consist of the mechanical, air-logic, solid-state programmable or electronic programmable types.
9. Typical tasks include simple assembly, loading or unloading, material handling, press operations, and injection molding.

MEDIUM-TECHNOLOGY ROBOTS

Until recently, medium-technology robots were considered to be high-technology units. With the advent of evermore sophisticated microprocessor controllers, a newer class of robots has been created (high-technology) that is capable of performing a much wider range of tasks than ever before. Therefore, we now have a category of industrial robot (medium-technology) that fits somewhere between the very simplest and the most sophisticated.

Another term used to describe a medium-technology robot is a ***"limited-application"*** device. See Fig. 25.

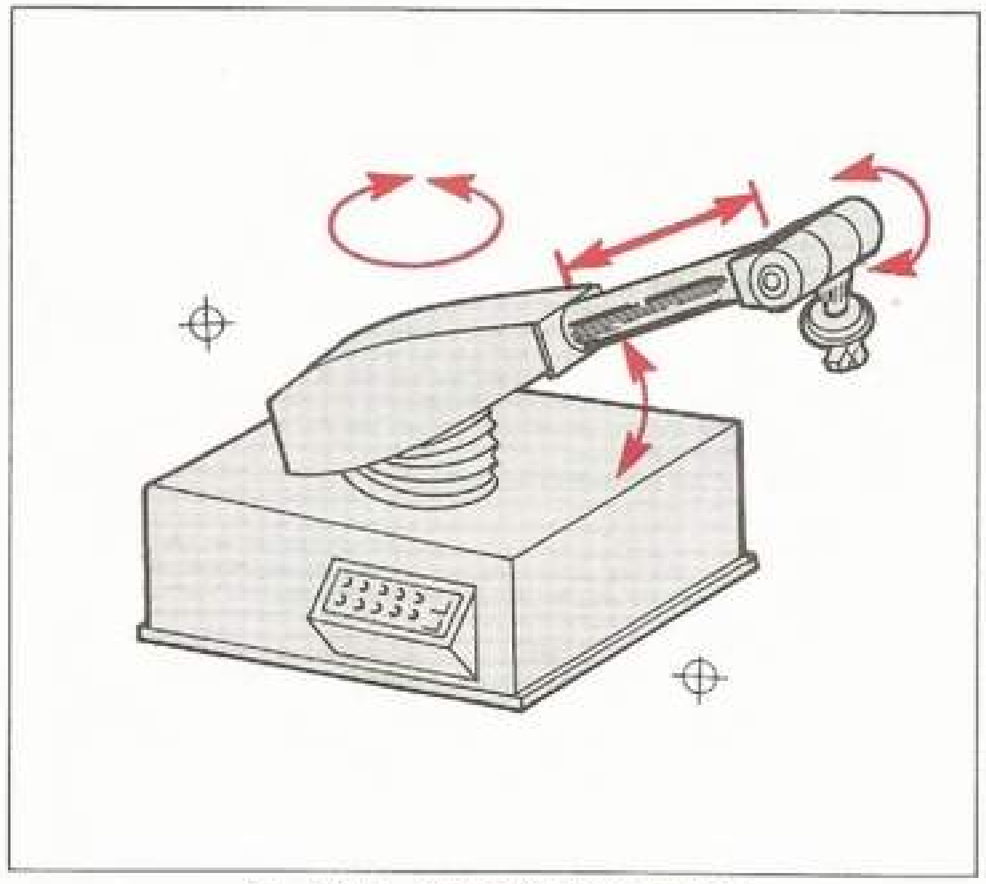


Fig. 24: A Pick-and-Place Device

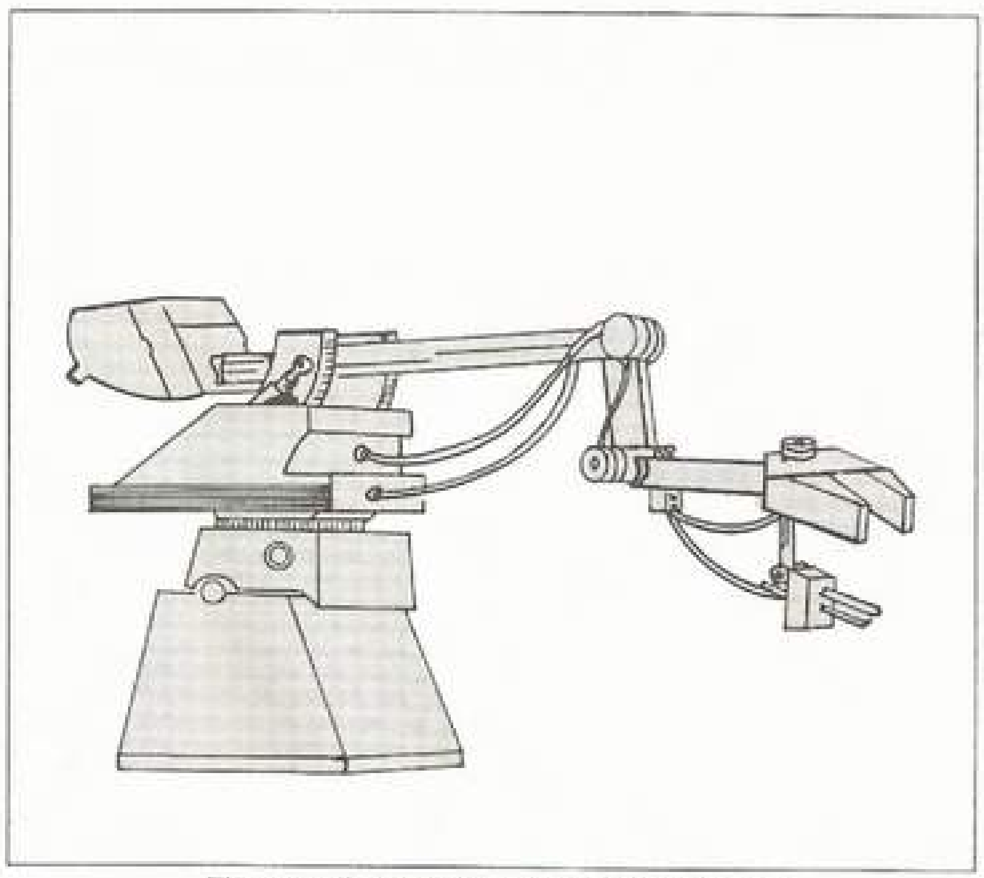


Fig. 25: A Limited-Application Device

A comparison of the following characteristics with those given for the low-technology robots will clearly reveal several important advantages which medium-technology robots enjoy over their earlier ancestors.

The distinguishing characteristics of medium-technology industrial robots are:

1. Up to six degrees of freedom possible, including a shoulder axis for elevation.
2. The horizontal and vertical axes have much greater freedom of travel, as well as the manipulator, which uses wrist bend, wrist yaw, and wrist rotation.
3. A greater distance of travel along each axis is possible.
4. Greater payloads are handled effectively (up to 150 pounds).
5. Speed is comparable to low-technology robots but since larger payloads are frequently handled, the cycle time is greater.
6. Accuracy is not as good as that of low-technology robots, ranging between +/–0.2 mm to +/–1.27 mm.
7. Axis actuation is accomplished chiefly by hydraulic or electric hydraulics, due mainly to the larger payloads involved.
8. Medium-technology robots operate with controllers that are reprogrammable, giving them the ability to learn. They utilize solid-state and microprocessor-based controllers.
9. Medium-technology robots can do everything that low-technology robots can do, and they can handle bigger payloads while doing it. This allows them to work effectively at die casting.

HIGH-TECHNOLOGY ROBOTS

High-technology industrial robots are obviously the top-of-the-line in the robot family. They incorporate the latest state-of-the-art designs and functions, using highly flexible manipulators and advanced programmable controllers. Another name for a high-technology robot is a "***universal***" device. See Fig. 26.

The distinguishing characteristics of high-technology industrial robots are:

1. They can have up to ten degrees of freedom, although no more than six are required.
2. A separate, microprocessor-controlled DC servo is used to actuate each axis, allowing the robot to be positioned at any given point in its work envelope.
3. Some have built-in diagnostic routines.
4. Payloads vary from 2.2 pounds to 1 ton.
5. Speed of operation is called straight-line velocity, and approaches 40 inches per second on the faster models.
6. Repeatability has been measured at +/–0.05 mm to +/–0.1 mm.
7. Pneumatic actuation is used to control the manipulator and the gripper.
8. High-technology robots use microprocessor-based computers as controllers. Some utilize proprietary system software and computer language which features the capability to add, delete, or replace data during on-line operation.
9. Among some of the sophisticated tasks executed by high-technology industrial robots are PC board assembly, spraypainting, arc welding, material inspection, palletizing, packaging, and machine tool loading.

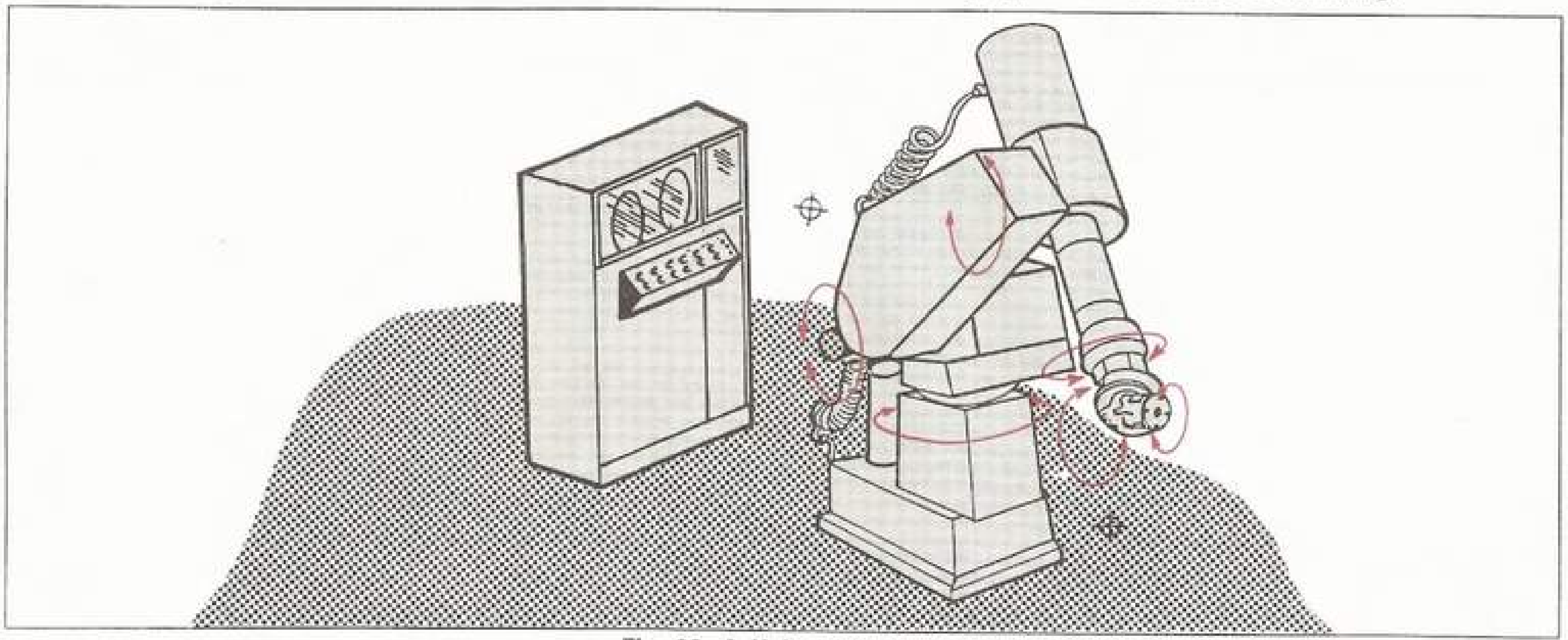


Fig. 26: A Universal Device

LAB EXPLORATION 10

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____10. ROBOT CLASSIFICATIONS

Purpose: This exploration will provide the student with the opportunity to review the major differences between the three categories of industrial robots.

Equipment: Knowledge Transfer Manual, Pencil

____a. Low-technology robots are also called ______________________ or ______________________ devices.

____b. Since low-technology robots do not contain any sensors, they are also called ______________________ devices. This means that they do not utilize ______________________ while in operation.

____c. Low-technology robots manipulate light ______________________ at high ______________________. Therefore, their actuation is achieved by ______________________ means.

____d. A medium-technology robot is called a ______________________ device.

____e. Payloads of up to ______________________ are possible when using medium-technology robots.

____f. If a robot has the capability to learn many different routines, its controller must be ______________________.

____g. When payloads become lighter, cycle time ______________________.

____h. Industrial robots which have built-in diagnostic capabilities are classified as ______________________.

____i. Medium-technology robots can have up to ____________ degrees of freedom.

____j. If an industrial robot is being used for spray-painting purposes, chances are it is a ______________________ type.

____k. A high-technology industrial robot is also known by another name. What is it? ______________________.

____l. High-technology industrial robots can have up to ______________ degrees of freedom, and can lift payloads of up to ______________________.

____m. This completes Lab Exploration 10, "Robot Classifications," and the INTRODUCTION TO ROBOTICS activities. Have your instructor initial your Knowledge Transfer Guide.

DISCUSSION

Since the use of high-technology robots depends a great deal on the work of skillful programming engineers, the software aspect of robotics is one of the first roadblocks encountered by those who are eager to enter the field. Indeed, it's certain that many would-be students have been intimidated or turned away from the study of robotics because they didn't already possess enough of the programming skills necessary to write meaningful robot programs.

If programming skills and experience are desired, Section IV explores this subject in detail. However, MARCRAFT International Corporation believes that the lack of programming skills and experience should not be allowed to hinder a student's introduction to robotics. For this reason, the MARCRAFT RS-2200 P/A Pro-Arm Robotics System now comes with our new PK TEACH software package. The package is designed to assist the student in the writing, editing, and testing of meaningful programs for the Pro-Arm.

No programming knowledge or experience is required to begin using the PK TEACH software package immediately. However, since the Pro-Arm interfaces with an IBM PC or compatible computer, it is assumed that the student has become familiarized with the nomenclature and the interfacing connections which were covered in the previous sections. Also, quicker progress will be realized by students who possess a working knowledge of the IBM PC or compatible computer. DOS commands which help in the creation of files and the manipulation of directories should be understood before beginning. The DOS reference tutorial contains all the necessary information needed for the student to master these commands.

The PK TEACH package runs on an IBM PC or compatible, with 640K or more of memory. Although PK TEACH may work with smaller amounts of memory, the size of your programs will probably be restricted. If a mouse operation is desired, then at least one serial port should be available.

Since MARCRAFT's PK TEACH program is designed to work effectively with a ***THREE-BUTTON*** **mouse** (see Fig. 27), be sure that the proper ***MOUSE.COM*** or ***MOUSE.SYS*** driver program has been properly installed first, as prescribed by the mouse manufacturer's enclosed instructions. When using the mouse, direct the diamond to the desired selection and press the ***LEFT*** mouse button. If you want to abort the operation being displayed, press the ***RIGHT*** mouse button.

If you want to exit, press the ***CENTER*** mouse button, except when you are in the movement section of the program, where the ***CENTER*** button is used to change axes. In general, pressing the ***LEFT*** button is like telling the program ***YES***, while pressing the ***RIGHT*** button is like telling the program ***NO***.

If the keyboard is the preferred input device, PK TEACH is versatile enough to allow the user to select options by either moving the reverse-highlight bar over the desired option or by pressing the highlighted letter of each option. Pressing ***ENTER*** then calls the selected option up.

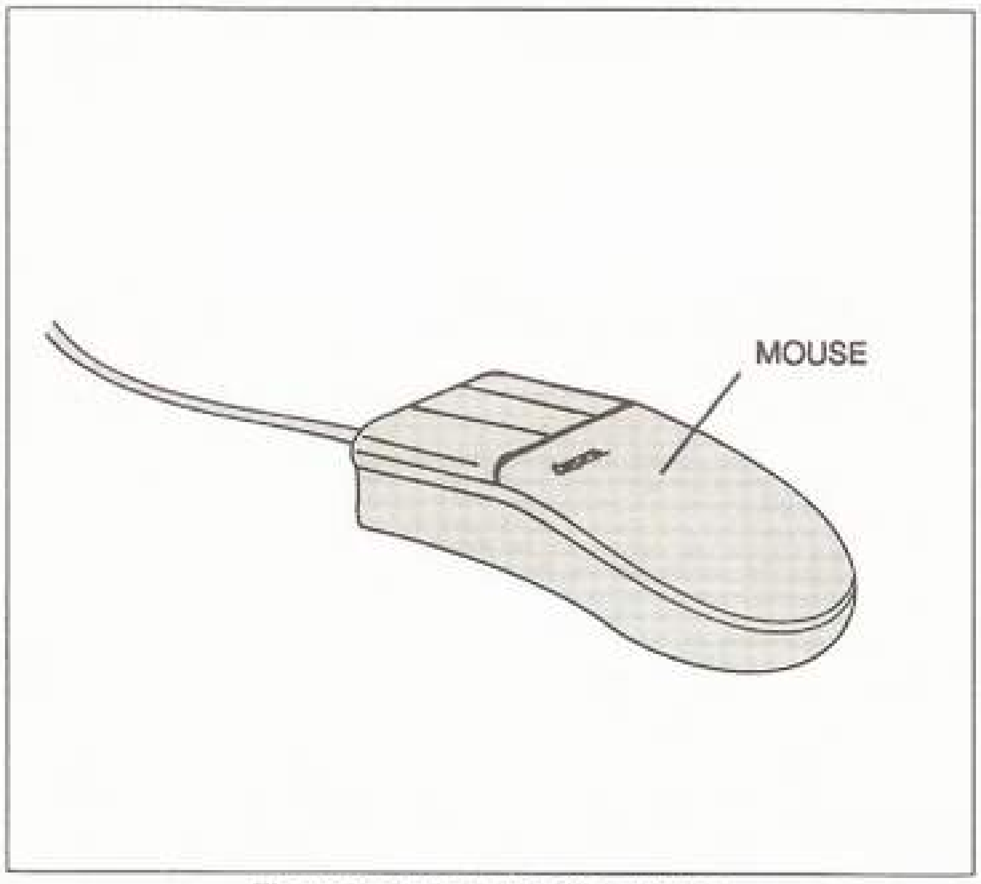


Fig. 27: A Three-Button Mouse

Each time PK TEACH is run, the first screen to be displayed is the MARCRAFT PK TEACH introductory screen. To get to the Main Menu, press any key. The Main Menu will give the user seven options from which to choose. If you do not have a mouse connected to the system, the Main Menu appears as in Fig. 28. If you have a mouse attached to the system, and the mouse driver program has been loaded, then the Main Menu appears as in Fig. 29.

*Note: When running PK TEACH, a solid center screen may appear with no writing and no action. If this happens, press the **RESET** button on the Pro-Arm to begin the PK TEACH session.*

Five of the Main Menu options will be presented in the following sequence:

1. Moving the Pro-Arm
2. Writing a New Program
3. Editing a Program
4. Running a Program
5. Deleting a Program

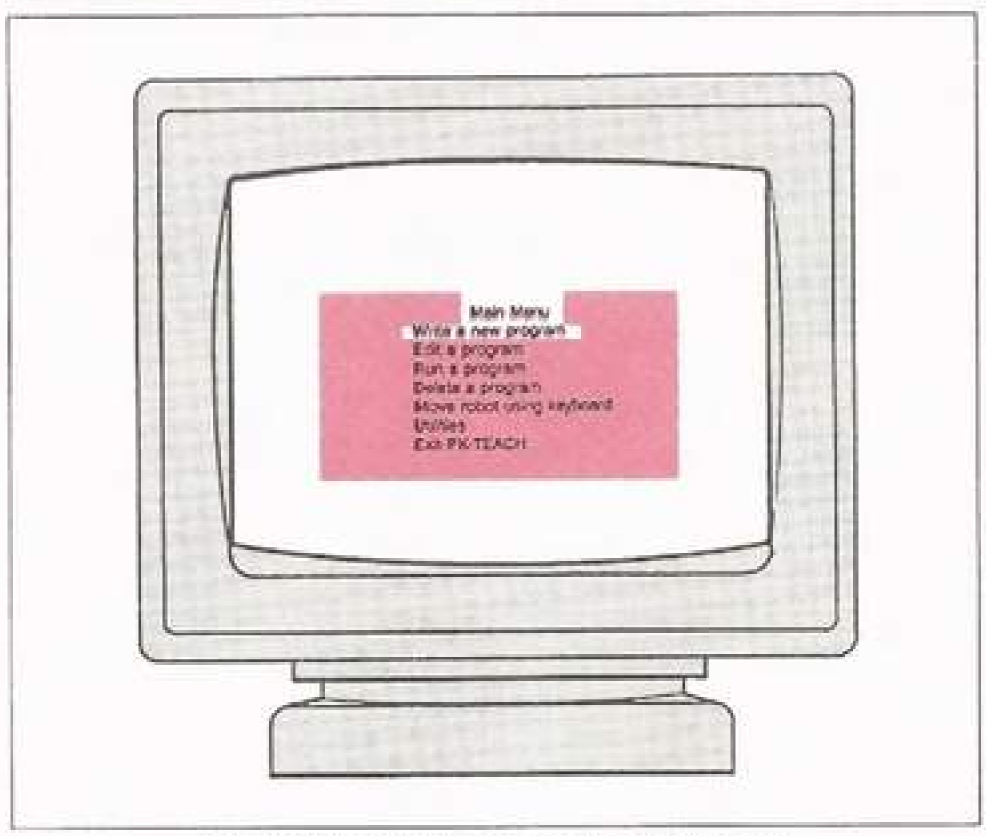
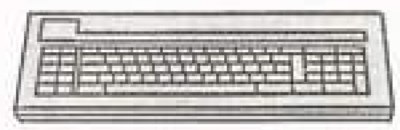
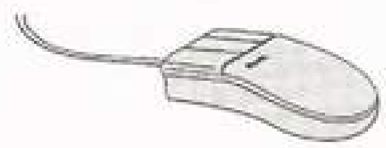


Fig. 28: Main Menu for the Keyboard

As the first several options are presented, operations will be separated and marked by icons for both the keyboard and mouse. Follow the steps for whichever input device ***(keyboard or mouse)*** you are using for your pendent.



To make a selection from the main program menu use either the cursor movement keys on the keypad, or press the intensified letter until the desired option is highlighted and then press ***ENTER***.



To make a selection, move the diamond over the selection you wish to make and press the ***LEFT*** button. You can still use the keyboard to make a selection even if your mouse has been activated.

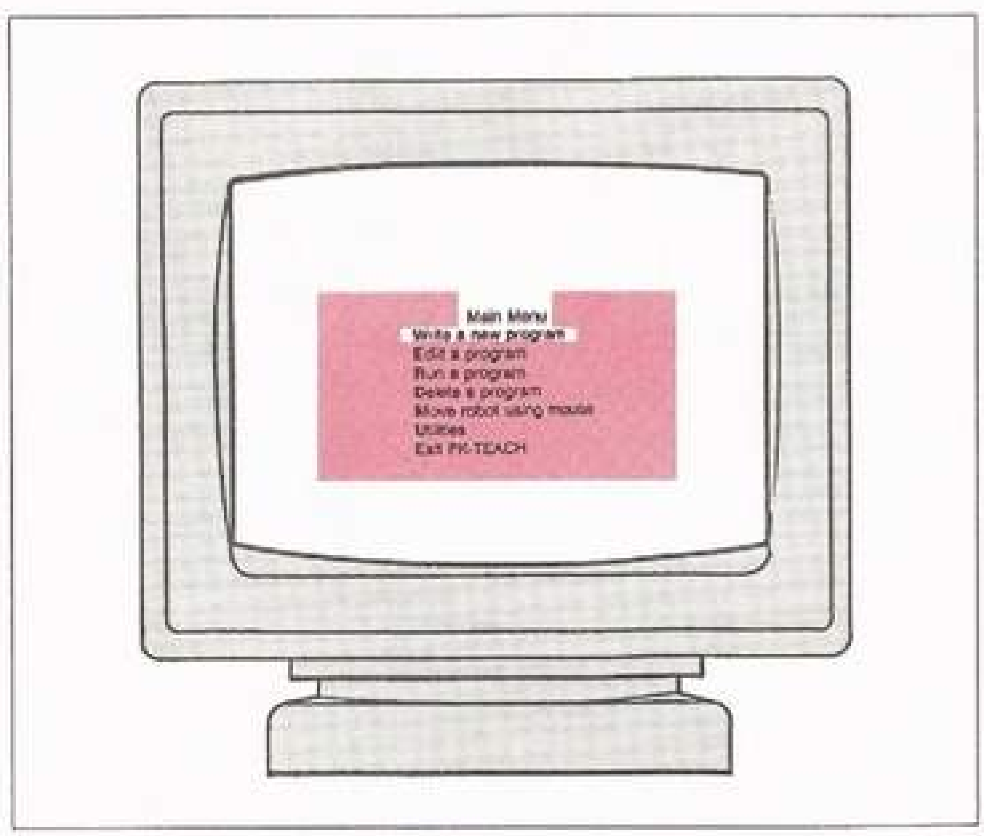


Fig. 29: Main Menu for the Mouse

MOVING THE PRO-ARM

Because moving the Pro-Arm robot properly is an integral part of writing, editing, and running a program in PK TEACH, this will be the first option selected from the menu.

LAB EXPLORATION 11

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

___11. MOVE IT

Purpose: This exploration will give the student practice in moving the Pro-Arm with either the mouse or the keyboard using the PK TEACH software disk.

Equipment: IBM-compatible computer system, Serial mouse, 9 to 25-pin adapter (if required), Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Pro-Arm Grid Sheet, IBM-type Parallel Interface Cable, Knowledge Transfer Manual, and the PK TEACH Program Disk.

___a. Assemble and install the required equipment on an uncluttered table or desk. Refer to the previous Pro-Arm labs.

___b. Perform the Pro-Arm Power-Up procedure.

___c. Be sure that the Pro-Arm is free of any obstruction for at least 16 inches on each side and across the front of the grid sheet.

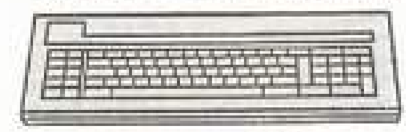
___d. Perform the Pro-Arm Zero-Positioning procedure.

___e. Perform the DOS Boot-Up procedure.

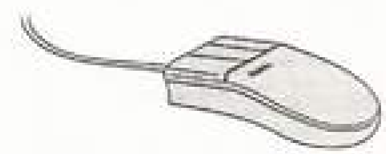
___f. Remove the DOS System Disk and insert the Pro-Arm TEACH Disk.

___g. Type **PKTEACH** and then press **ENTER**. *Note: The MARCRAFT PK TEACH introductory screen appears. Press any key.*

___h. At the PK TEACH main menu:



Using the cursor movement keys, select **"Move robot using keyboard."** The selection will be highlighted. Then, press **ENTER.** You may alternately make this selection by pressing the highlighted letter "M" before pressing **ENTER.**



Using the mouse, position the diamond over **"Move robot using mouse,"** and click the **LEFT** mouse button.

*Note: If the **NUM LOCK** key is active, a message window will appear informing you that you will not be able to use the cursor movement keys until you press **NUM LOCK** once to deactivate it. If you are using an AT-style keyboard, the second set of cursor-movement keys will work.*

Five windows appear on the screen. The three windows across the top supply information about the position of each axis of the Pro-Arm. Refer to Fig. 30.

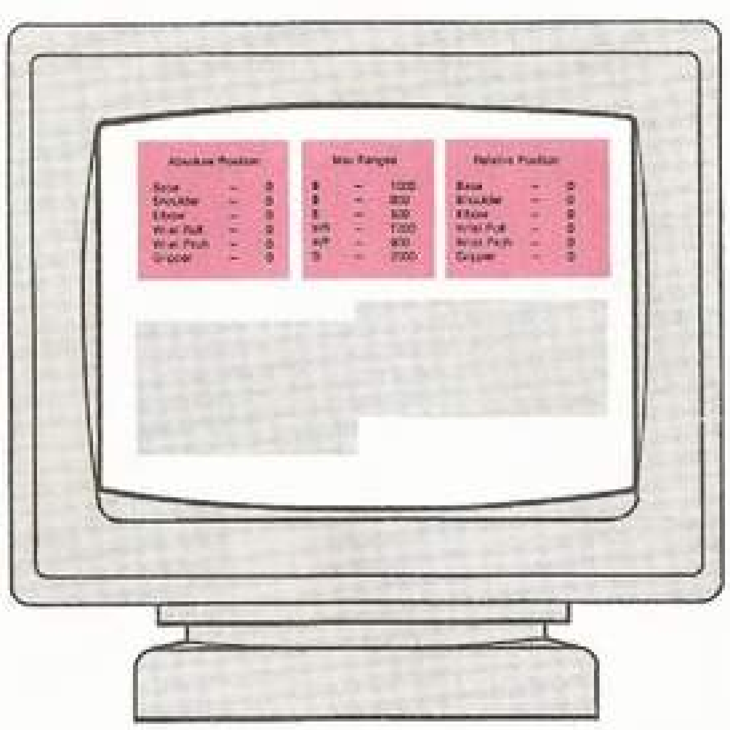


Fig. 30: The Three Top Windows

At the upper left is the "**Absolute Position**" window. Its function is to display the latest positional information available on all six motors, using the "**ZERO**" or "**HOME**" position as a reference point.

At the upper right is the "**Relative Position**" window. Its function is to display the latest positional information available on all six motors, using the last position which was saved in the current program as a reference point. As soon as a positional point is stored (during a **WRITE** session), the relative position indicators will return to zero. As soon as the Pro-Arm moves to a new position, that position, relative to the position which was just saved, will be displayed.

Note: Since you are now merely moving the Pro-Arm around, rather than storing positions on a program, the ***"Relative Position"*** *window's information will be identical with the* ***"Absolute Position"*** *window's information for the remainder of this exploration.*

The window at the upper middle is the "**Max Range**" window. It contains information about the maximum number of steps that each of the six motors are allowed to travel. These ranges never change and reflect the physical limitations of the Pro-Arm, relating to its absolute position rather than to its relative position. For example, the "**Max Range**" window states that the shoulder motor's maximum range of movement is 600 steps above or below its home position. If an attempt is made to move the shoulder move than +/– 600 steps from the home position, the Pro-Arm will not move and a window will pop up with an error message that reads, "**Robot is unable to do that...Robot Shoulder out of range...Press any Key or Mouse Button to continue.**" The same message will appear if any other commands are received which would move Pro-Arm beyond the maximum ranges specified. The message would be adjusted for whichever motor would exceed its range limit.

The window in the lower right area of the screen is the "**Robot Command**" window. This window displays the latest command that was sent to the Pro-Arm. Right now it is blank, since no commands have yet been sent (by moving one of the Pro-Arm's motors). It won't show any reading until then. Although programming is not discussed in this section, you can check this window, after moving the Pro-Arm, to see which motor is referenced on the command line. The command line will be discussed thoroughly in Section IV.

The window in the lower left of the screen has no name. Its display varies depending upon whether or not you are using the mouse. At this point, if the mouse is connected to the system, this window should appear as in Fig. 31.

Note: If you are not using the mouse, the window appears as in Fig. 32. Although it has no name, we will refer to this window as the ***"Movement Control"*** *window, when referring to keyboard operations.*

____i. At the "**Movement Control**" window:

Take no action. The window already contains a menu of movement parameters from which to select, as shown in Fig. 32. To make a selection, use the up/down cursor-movement keys, or press the highlighted letter indicated. When the selection is made, it becomes a highlighted bar.

Click the **LEFT** mouse button. The window should now contain a menu of movement parameters from which to select (see Fig. 33) by clicking the combination of mouse buttons indicated to the right of each parameter. This window is now labeled the "**Pendent control panel.**"

Note: If you clicked the ***RIGHT*** *mouse button instead of the* ***LEFT****, the window will contain the same display as described for the keyboard. You can still use the mouse to make your selections by moving the diamond over the chosen parameter and clicking the* ***LEFT*** *button (point-and-shoot). Some find that they prefer this method. In the Lab Explorations of this section, all the mouse instructions will be given from the pendant mode.*

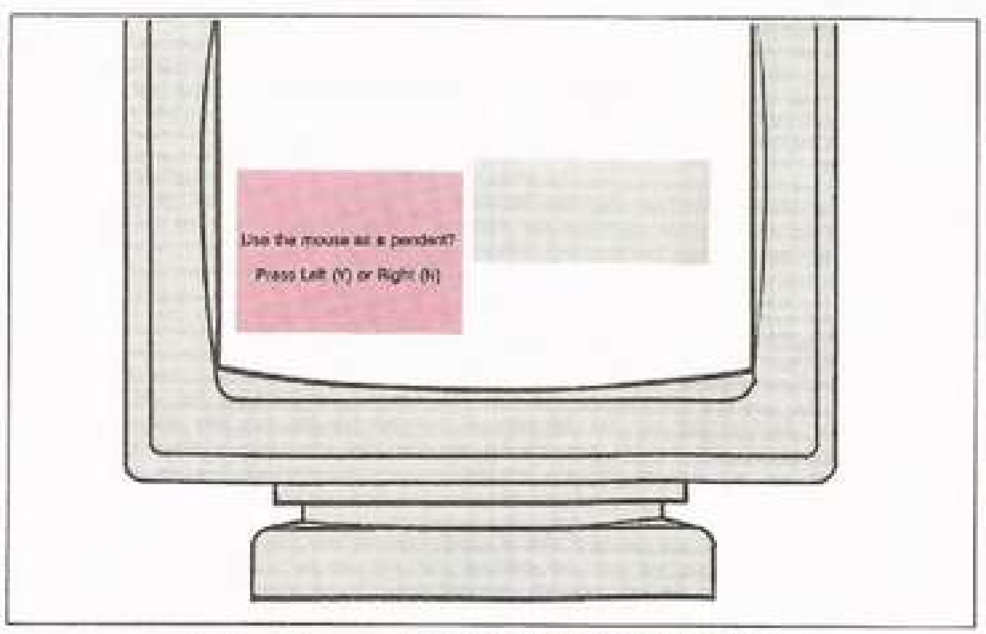


Fig. 31: The Mouse Pendent Window

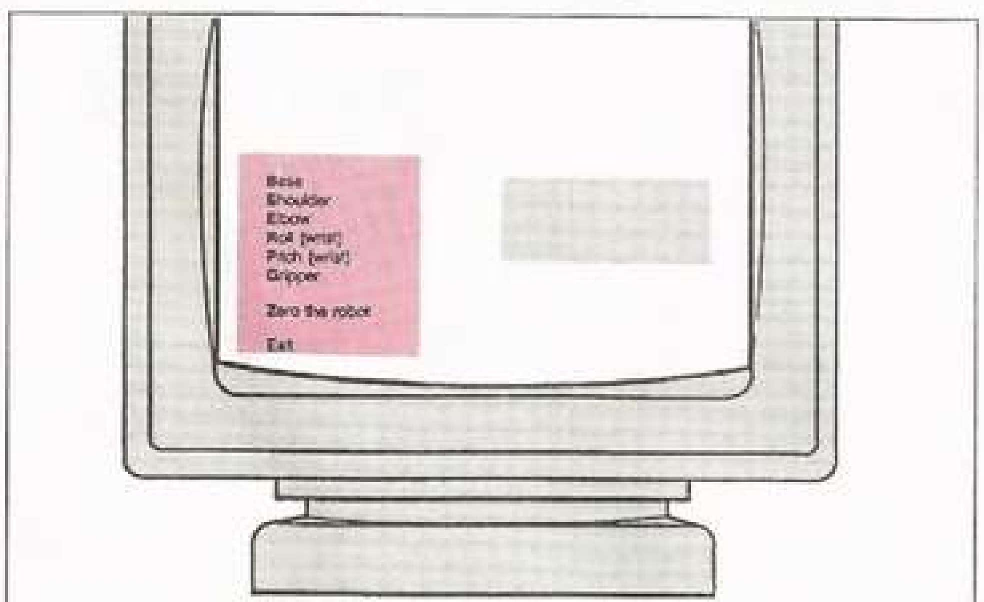


Fig. 32: The Movement Control Window for the Keyboard

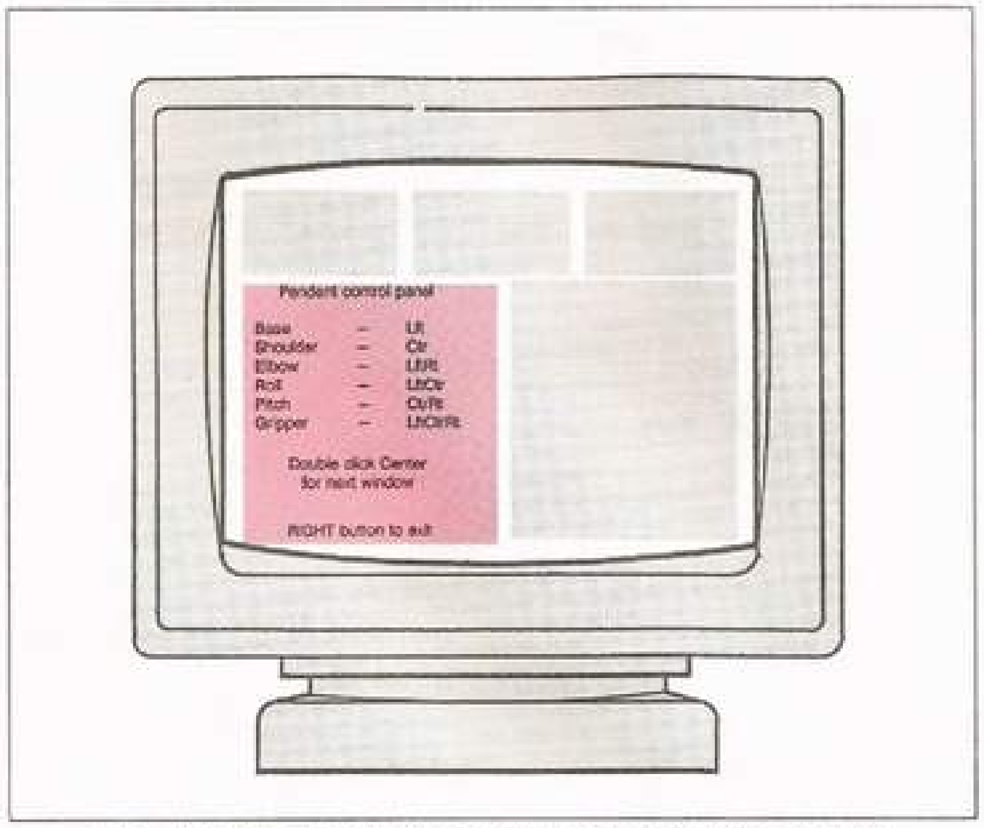
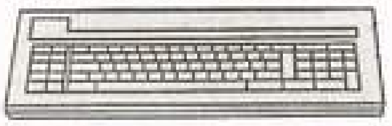


Fig. 33: The Pendent Control Panel for the Mouse

____j. At the "**Movement Control**" window:



Select the **BASE** parameter and press **ENTER**. *Note: At this point, the "**Movement Control**" window becomes the "**Move Base**" window, as shown in Fig. 34.*

Select the **BASE** parameter by clicking the **LEFT** mouse button. *Note: At this point, the "**Pendent Control**" panel becomes the "**Move Base**" window, as shown in Fig. 35.*

____k. Examine the first line of text in the "**Move Base**" window. The first line lists the "**Current increment setting**" for the base. At this time the current increment setting is ________________.

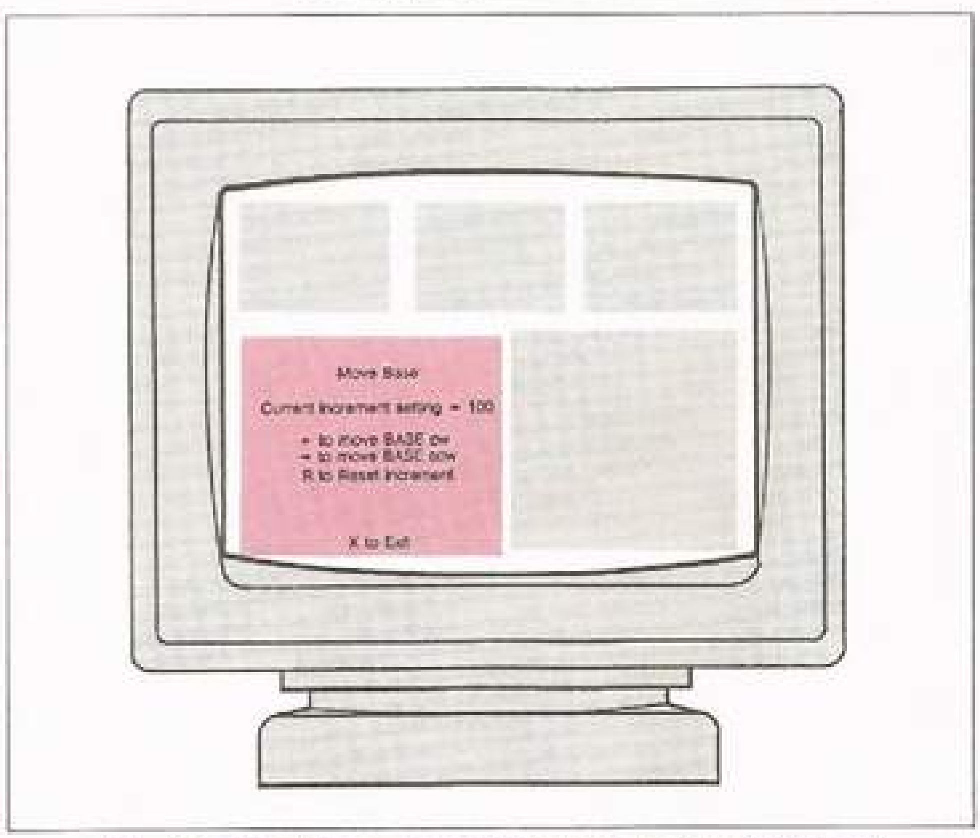
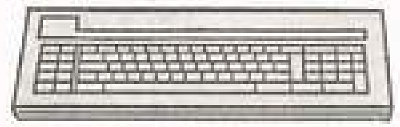


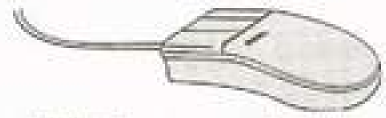
Fig. 34: The Move Base Window for the Keyboard

Note: This number shows the minimum number of steps that the base motor will move, in either direction, for each move command it receives.

____l. The next line indicates that clicking the **LEFT** mouse button will move the base clockwise. If you are using a keyboard, pressing the **LEFT** cursor-arrow key will accomplish the same thing. Perform the following movement:



Press and release the **LEFT** cursor-arrow key.



Click the **LEFT** mouse button.

Did the base move?

____

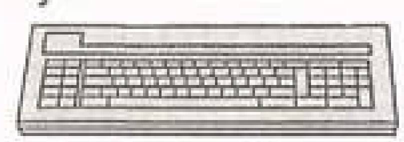
In what direction did the Pro-Arm's base move?

____

How many steps does the "**Absolute Position**" window indicate that the base moved?

____

____m. To move the base counterclockwise, the next line instructs the user to click the **RIGHT** mouse button. Keyboard users will want to press the **RIGHT** cursor-arrow key. Perform the following movement:



Move the Pro-Arm 400 steps counterclockwise by pressing the **RIGHT** cursor arrow key.

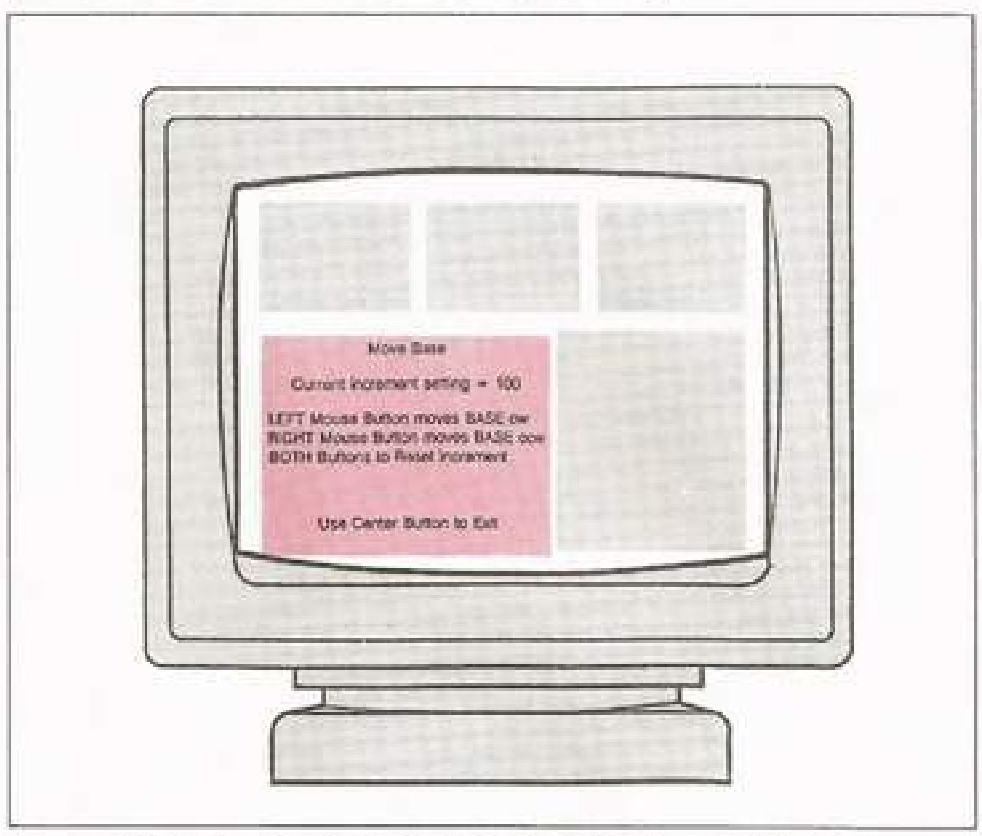
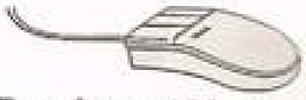


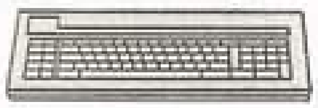
Fig. 35: The Move Base Window for the Mouse



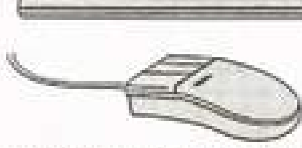
Move the Pro-Arm 400 steps counterclockwise by clicking the **RIGHT** mouse button the required number of clicks.

*Note: If you have tried both the keyboard and the mouse, you may have already noticed a difference in the way the keyboard and the mouse work when sending a **MOVE** command to the Pro-Arm. The mouse button must be released, and the user must wait until the current movement is completed, before sending another move command to the Pro-Arm. However, the keyboard can send movement commands to the Pro-Arm continuously, by simply holding a movement key down, without waiting for the previous command to be executed.*

____n. Certain movements require a larger or smaller increment in order to get the Pro-Arm into the exact position desired. Perform the following:



Press the letter "**R**".

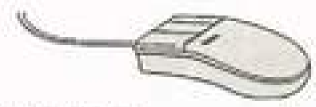


Click the **LEFT** and **RIGHT** mouse buttons together.

*Note: The "**Move Base**" window has become the "**Reset Increment**" window, and now displays the valid increment range for the Pro-Arm's base, along with the current increment setting. Once you are in the "**Reset Increment**" window, you cannot exit without entering an increment value. If you don't want to change anything, type in the same increment value as the current setting. The only exit from the "**Reset Increment**" window is by entering a valid increment setting and then pressing the keyboard **ENTER** key. Try a few different increment settings to make sure they work as advertised. Then, exit the "**Reset Increment**" window.*

____o. Return the base to its home position and exit the "Move Base" window in the following manner:

Press the "**X**" key.

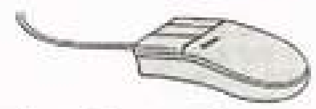


Click the **CENTER** mouse button.

____p. At the "**Movement Control**" window:

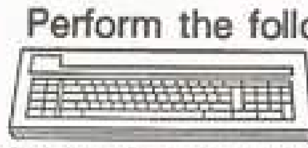
[KEYBOARD ICON]

Select the **SHOULDER** parameter and press **ENTER**. *Note: The "**Movement Control**" window becomes the "**Move Shoulder**" window as shown in Fig. 36.*

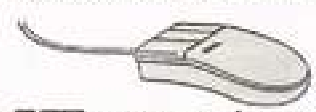


At the "**Pendent Control**" panel, select the **SHOULDER** parameter by clicking the **CENTER** mouse button. *Note: The "**Pendent Control**" panel becomes the "**Move Shoulder**" window as shown in Fig. 37.*

____q. In the "**Move Shoulder**" window, the current increment setting is _________. Perform the following:



Press and release the **UP** cursor-arrow key.



Click the **LEFT** mouse button.

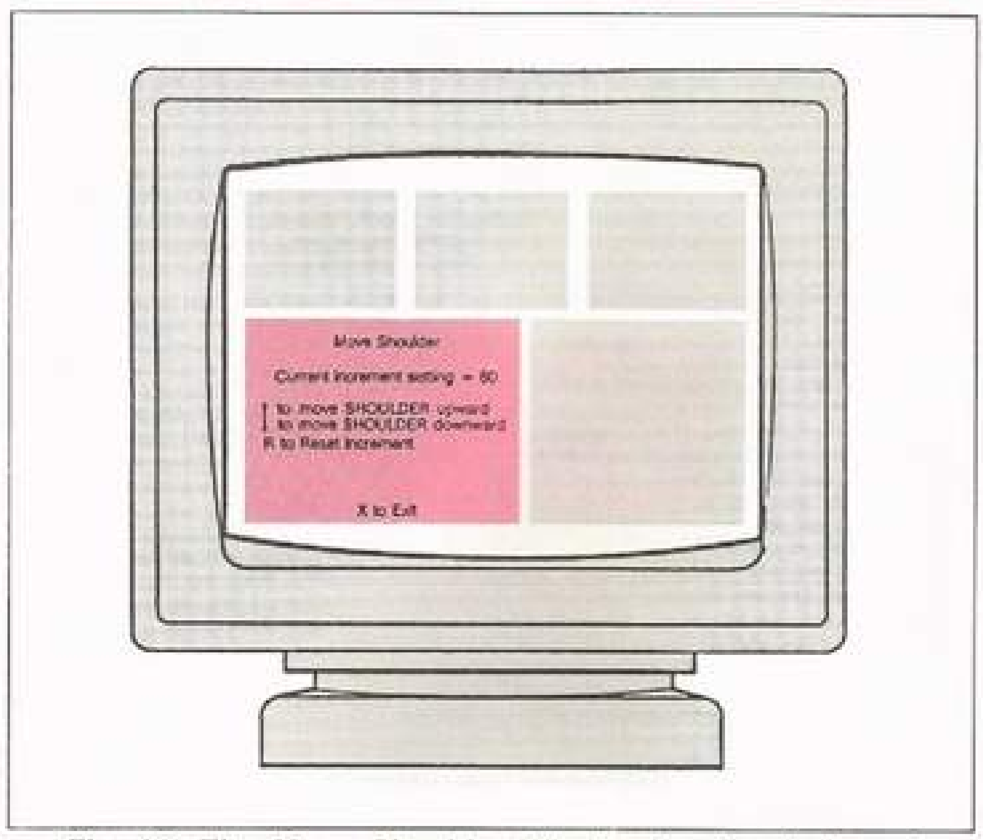


Fig. 36: The Move Shoulder Window for the Keyboard

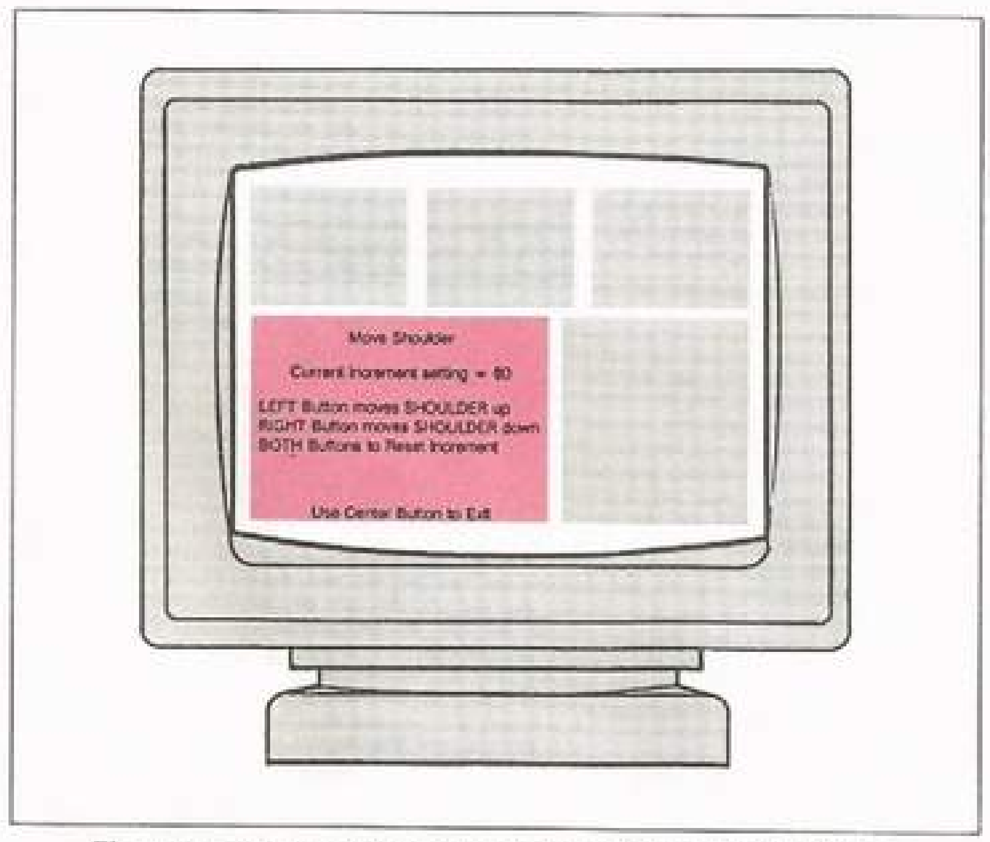


Fig. 37: The Move Shoulder Window for the Mouse

____r. The arm moved ____________________.

The "**Absolute Position**" window indicates that the Pro-Arm moved how many steps?

____

Note: There will be a minus (–) sign preceding the step number in the window if the Pro-Arm moved ***DOWN****.*

Are there any other positional changes indicated in the "**Absolute Position**" window?

____

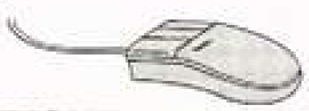
If so, what were they?

____

Note: It may sometimes be necessary to make sure that the movement of one axis doesn't carry another axis out of its maximum range.

____s. Move the shoulder down for 180 steps in the following manner:

Press and release the **DOWN** cursor-arrow key three times slowly.



Click the **RIGHT** mouse button, and wait for the Pro-Arm to stop moving. Click the **RIGHT** mouse button again, wait, and click the **RIGHT** mouse button once more.

Did the arm move down as expected?

______

The step position for the shoulder now reads what?

______

Note: You should now be able to conduct movements of the Pro-Arm, using either the keyboard or the mouse, without step by step instructions.

____t. Practice making movements with the Pro-Arm using the **elbow** (Fig. 38 and Fig. 39), **wrist roll** (Fig. 40 and Fig. 41), **wrist pitch** (Fig. 42 and Fig. 43), and **gripper** (Fig. 44 and Fig. 45) parameters. After getting the feel of the PK TEACH movement parameters, align the arrows of each axis to the "**HOME**" or "**ZERO**" position. Then, exit either to the "**Movement Control**" window, or the "**Pendent Control**" panel.

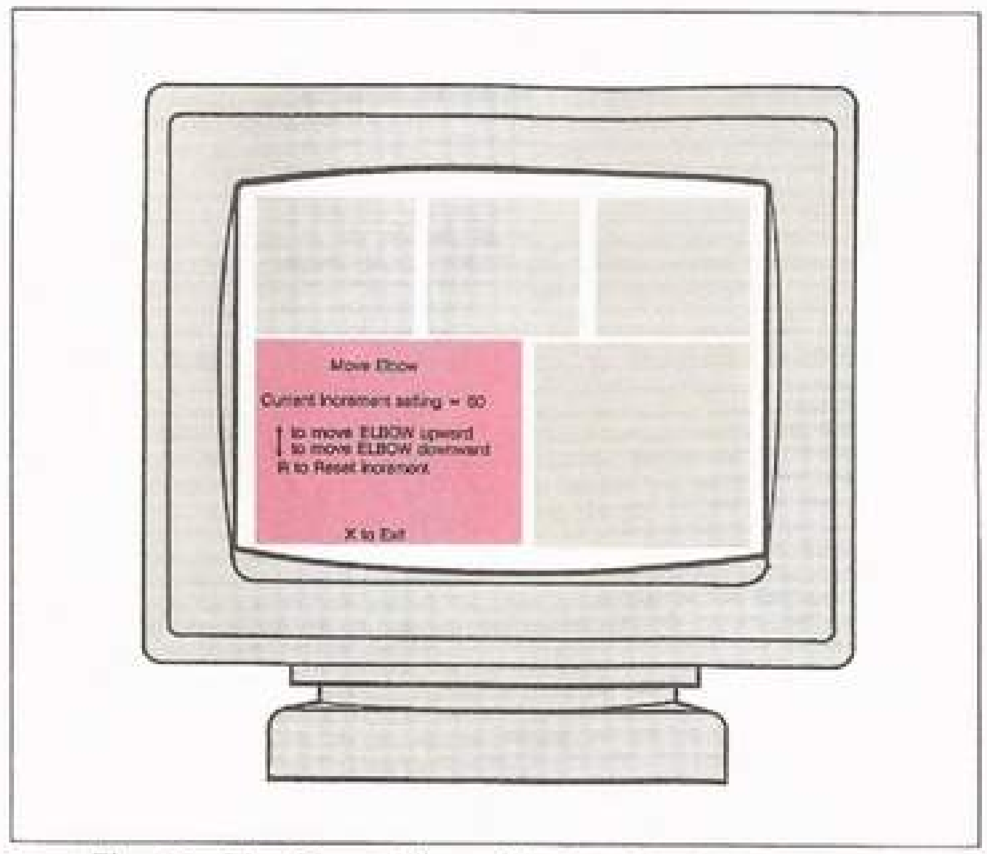


Fig. 38: The Move Elbow Window for the Keyboard

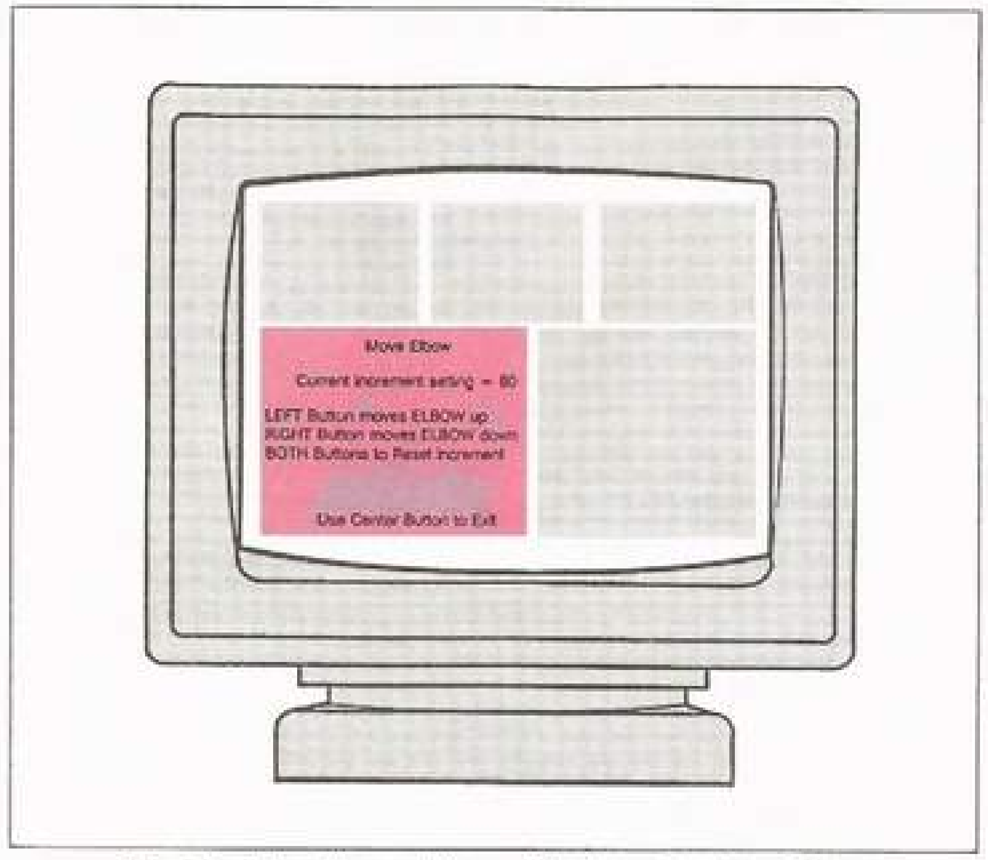


Fig. 39: The Move Elbow Window for the Mouse

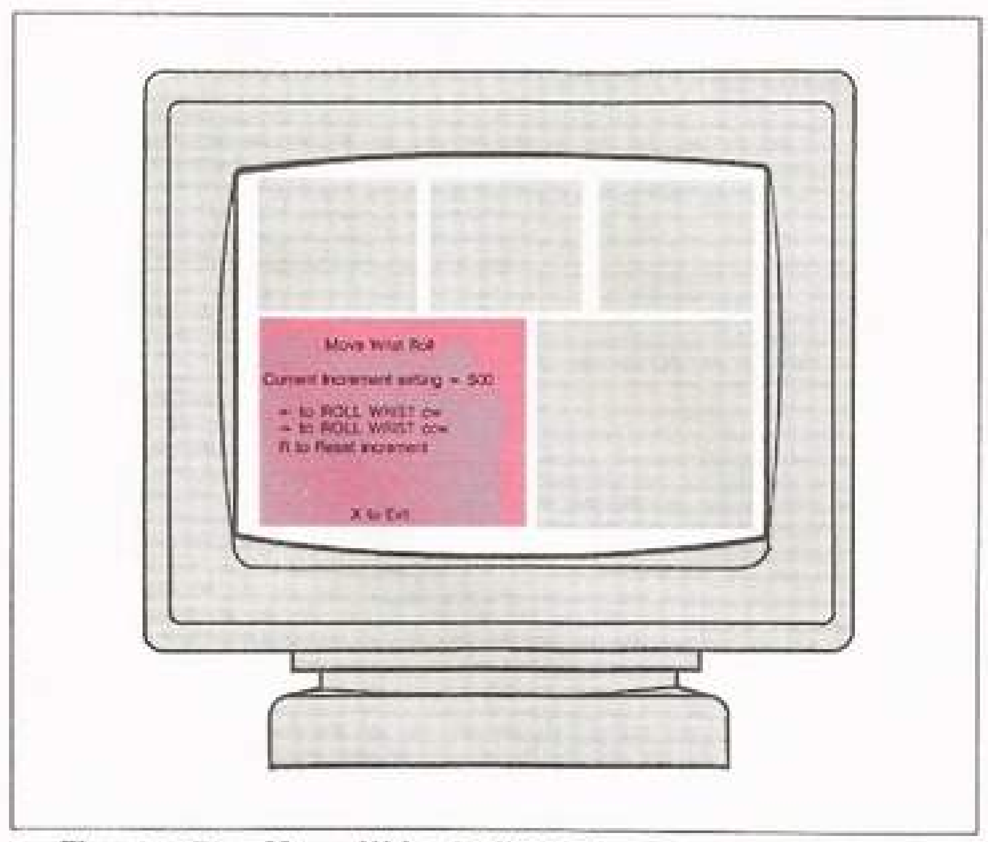


Fig. 40: The Move Wrist Roll Window for the Keyboard

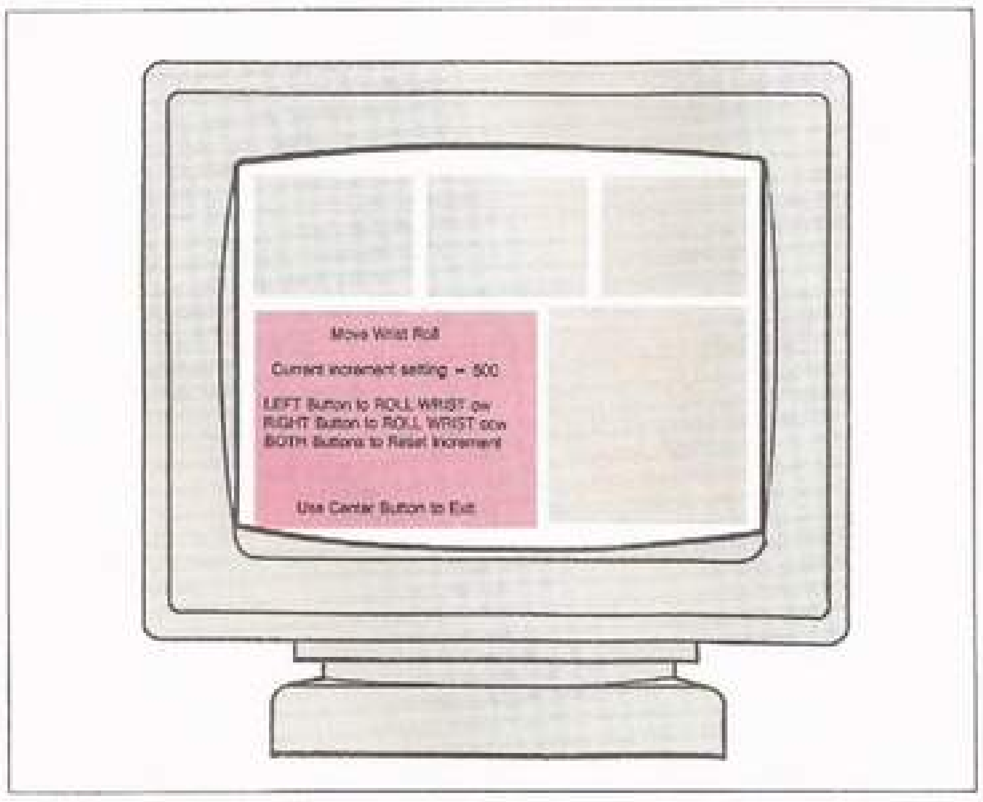


Fig. 41: The Move Wrist Roll Window for the Mouse

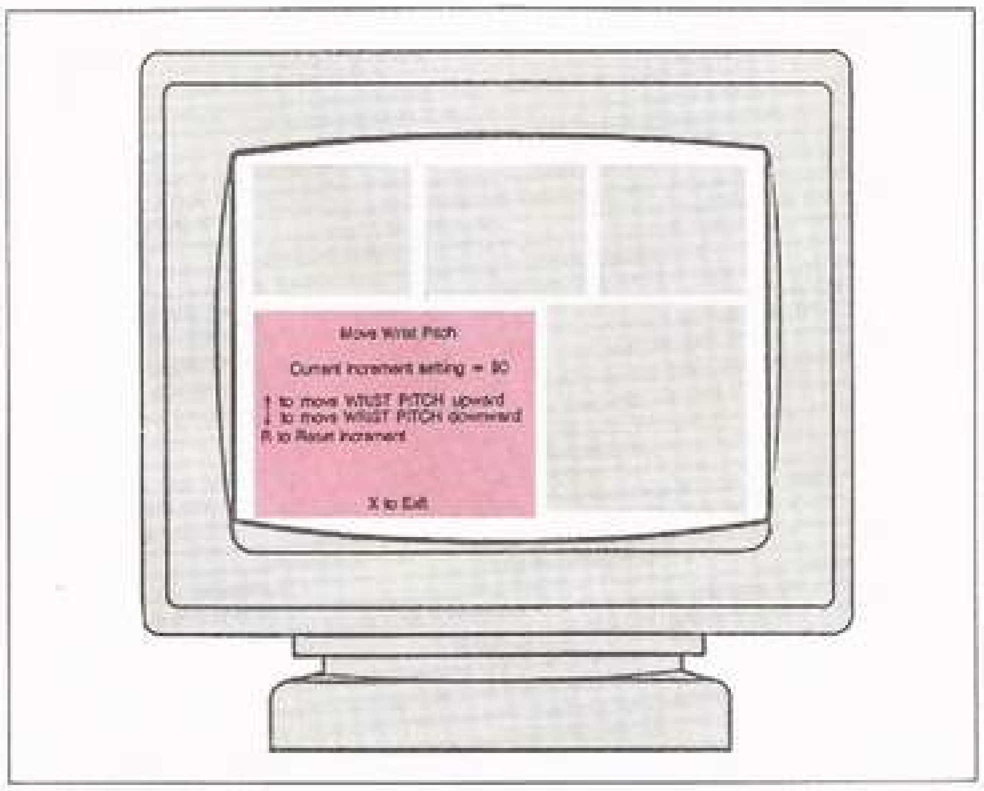


Fig. 42: The Move Wrist Pitch Window for the Keyboard

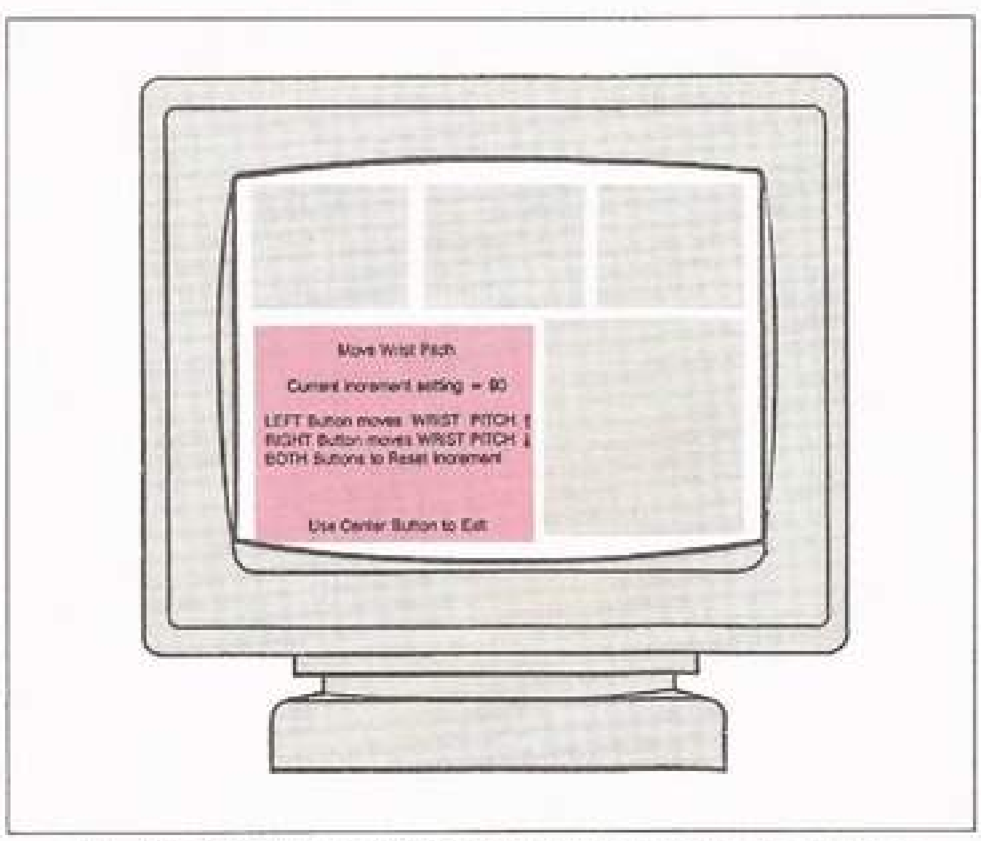


Fig. 43: The Move Wrist Pitch Window for the Mouse

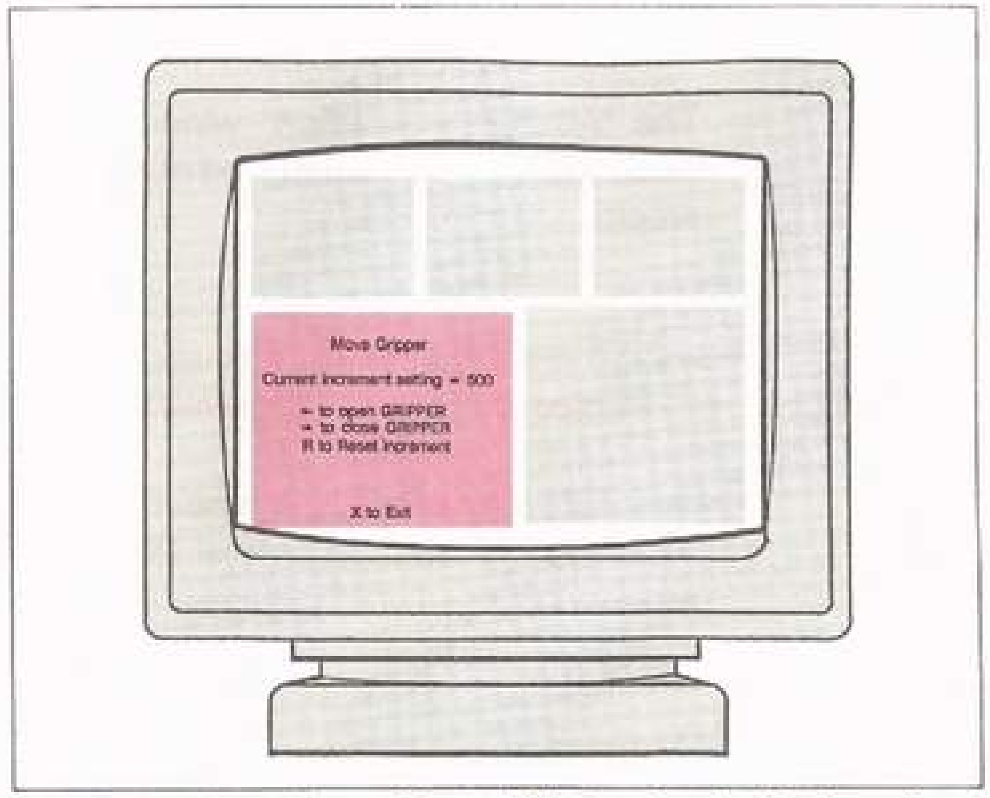


Fig. 44: The Move Gripper Window for the Keyboard

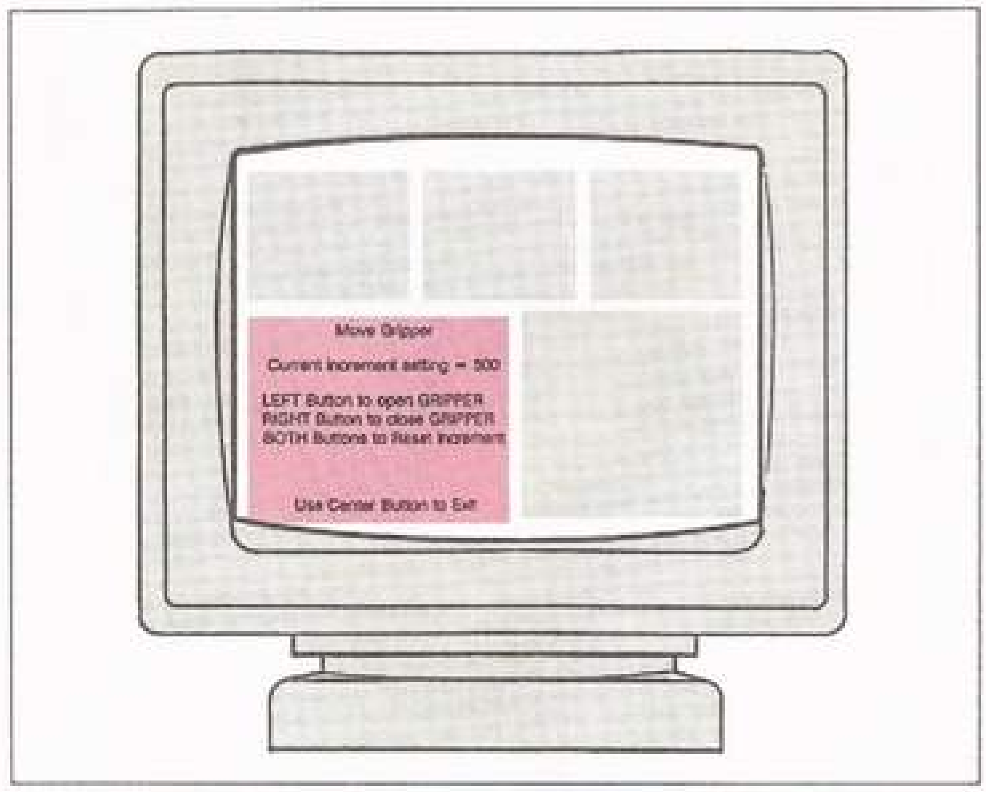


Fig. 45: The Move Gripper Window for the Mouse

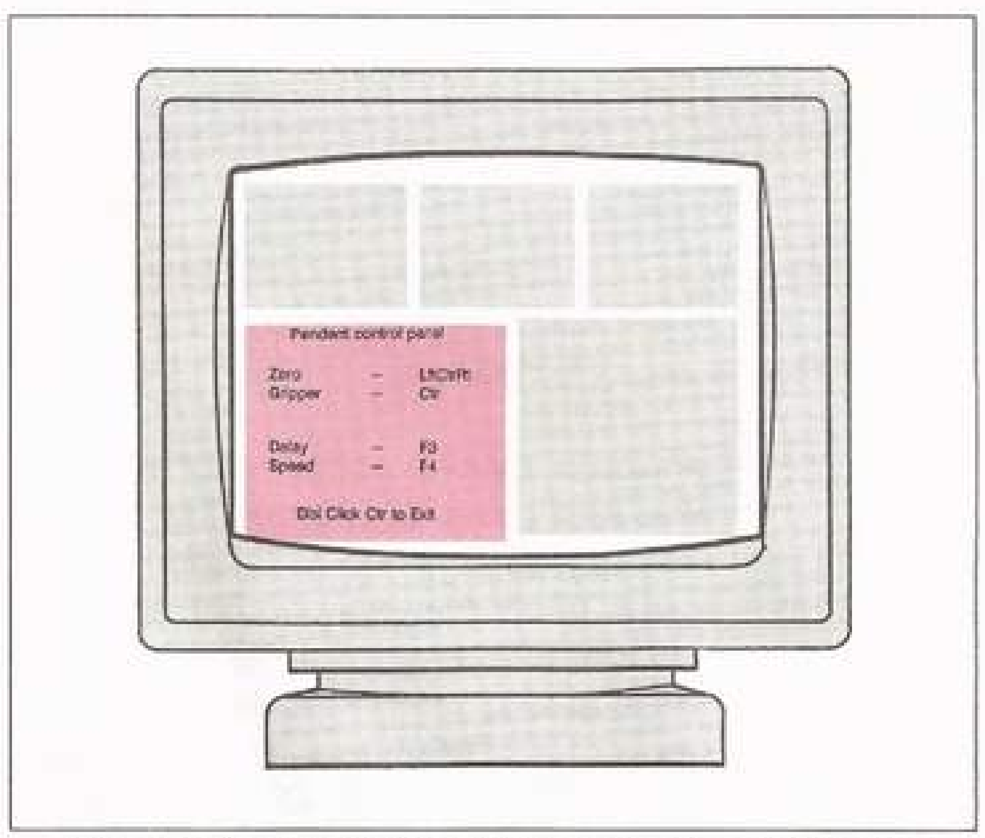
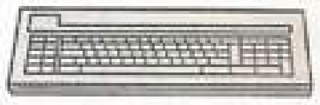
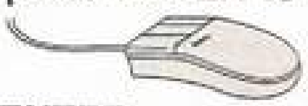


Fig. 46: The Pendent Control Panel's Second Window

____u. Perform the following:



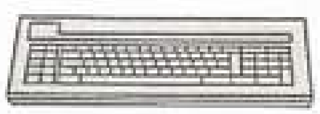
Select "**Zero the robot**" from the "**Movement Control**" window and press **ENTER** to call it up.



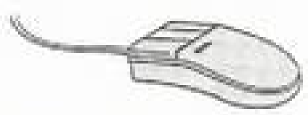
Click the **CENTER** mouse button twice, quickly. Notice that a second window for the "**Pendent Control**" panel appears, containing several other parameters, as shown in Fig. 46. Click all three mouse buttons together to call the "**Zero**" parameter up.

*Note: The "**Zero Position**" window has popped up on the screen with the message, "**NOTE: This option DOES NOT tell the robot to go back to the zero position. This option tells the robot to consider its current position to be the zero position. Do you wish to have the current position be the zero position?**"*

If you zeroed the Pro-Arm as instructed in the previous step, then:



Press the letter "**Y**" and proceed to **Step v.**



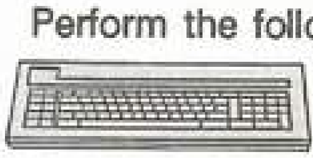
Click the **LEFT** mouse button and proceed to **Step v.**

If you have not yet aligned the axes of the Pro-Arm to their home positions:

Press the letter "**N**" and repeat **Step t.**

Click the **RIGHT** mouse button and repeat **Step t.**

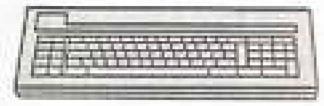
____v. Perform the following:



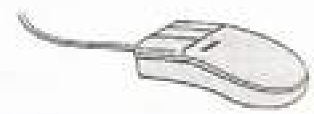
Using several of the various movement parameters, move the Pro-Arm to a position other than "**ZERO**". Then, return to the the "**Movement Control**" window.

Double-click the **CENTER** mouse button to return to the first "**Pendent Control**" panel's window. Then, move the Pro-Arm to a position other than "**ZERO**", using several of the various movement parameters.

____w. Perform the following:



Select "**Exit**" from the "**Movement Control**" window menu and press **ENTER.**



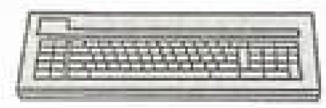
Click the **RIGHT** mouse button.

What happened?

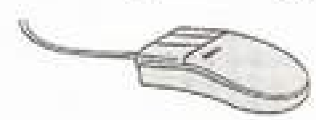
____

____

*Note: The "**Exit MOVE**" verification window appears*



Verify your decision to exit from the "**Movement Control**" window by pressing the letter "**Y.**"



Verify your decision to exit from the "**Pendent Control**" panel by clicking the **RIGHT** mouse button.

Did the Main Menu reappear on the screen?

____

What else happened?

____

____

____x. Select "**Exit**" from the Main Menu, and verify your decision to leave PK TEACH.

____y. Turn the Pro-Arm power supply off, and remove the PK TEACH disk from the disk drive, placing it in its protective sleeve. Then, turn the computer system off, but leave the system set up with all the interface connections intact.

____z. This completes Lab Exploration 11, "Move It." Please have your instructor initial your Knowledge Transfer Guide.

Now that you have become familiar with how to use PK TEACH in moving the Pro-Arm, it's time to explore some of the other options available with this software package. One of those options allows the user to write a program for the Pro-Arm, without having any previous programming knowledge or experience.

How about writing a program of your own? Your first program doesn't have to be anything fancy. Just have some fun, while you learn how an original program is put together, tested, and corrected.

With the RS-2200 P/A Pro-Arm System set up as in the previous exploration, let's begin.

LAB EXPLORATION 12

In the appropriate space provided, write the word that best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

____12. WRITE IT

Purpose: This exploration will give the student practice in writing programs for the Pro-Arm robot using the PK TEACH software disk.

Equipment: IBM-compatible computer system, Serial mouse, 9 to 25-pin adapter (if required), Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Grid Sheet, IBM-type Pro-Arm Parallel Interface Cable, Knowledge Transfer Manual, and the PK TEACH Disk.

____a. Check the system installation and interfacing to be sure that it is ready to go.

____b. Boot the computer system and apply power to the Pro-Arm.

____c. At the prompt **(A>_)**, remove the **DOS** disk, insert the PK TEACH disk, type **PKTEACH**, and press **ENTER**.

____d. Press any key or button and at the Main Menu, select **"Write a new program,"** and call it up **(click or ENTER)**. Check to be sure that the **NUM LOCK** key is deactivated. *Note: The zero-position "Verification" window appears, as shown in Fig. 47.*

____e. At the **"Verification"** window, answer **"Y"**. *Note: The "Filename" window appears, as shown in Fig. 48.*

Note: This is where you can name the program you are about to write. The program name can be eight characters or less.

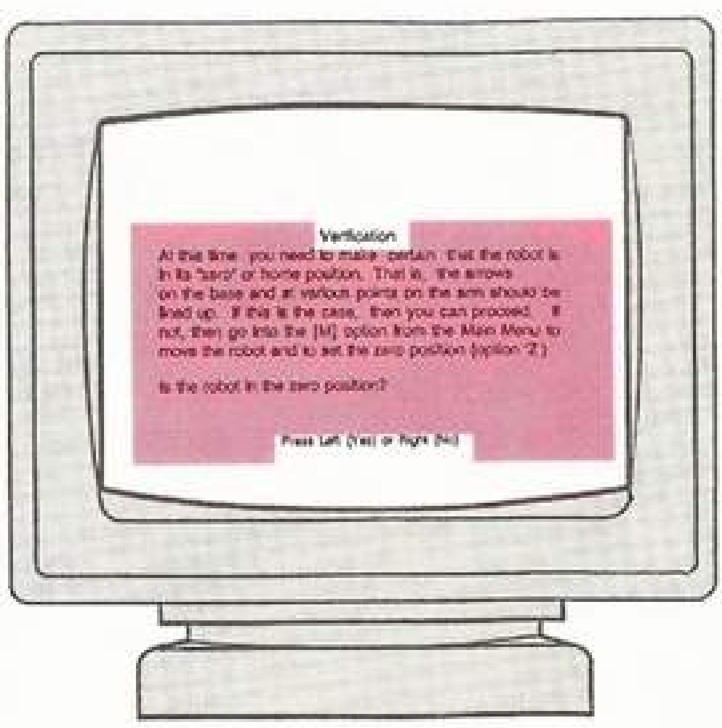


Fig. 47: The Zero-Position Verification Window

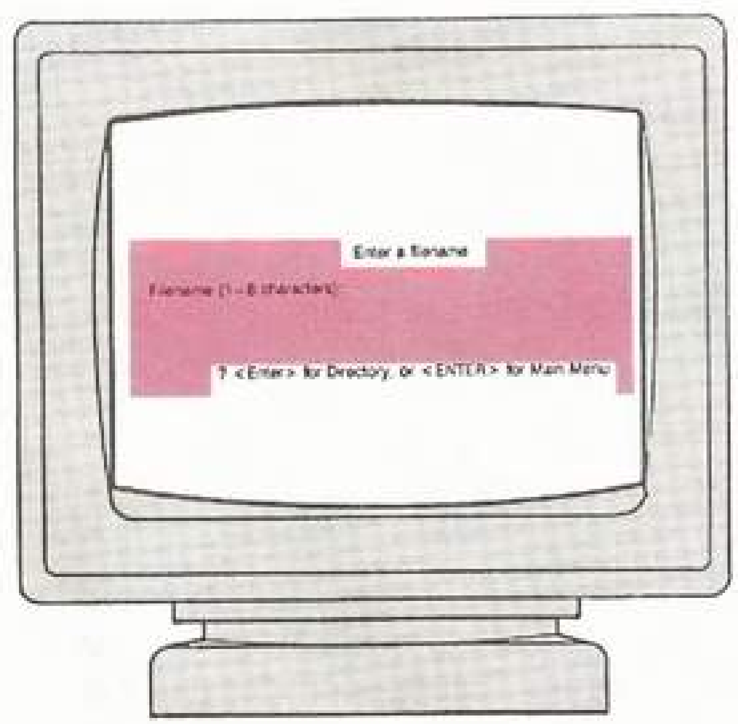


Fig. 48: The Filename Window

*Note: At this point, if needed, you should already be familiar with how to zero the Pro-Arm using PK TEACH. Another way is simply to run the Pro-Arm's self-test program, using the **TEST** and **RESET** buttons on its Front Panel.*

Caution: If you have never created a DOS filename before, you might want to check an MS-DOS reference manual before going on, because there are several characters that are invalid for use in filenames. Generally, if you only use letters or numbers in the filename, any eight-character (or less) combination will work.

*Note: Most filenames contain a three-character grouping called a **"Filename Extension"** to help both the system and the user to determine what type of file has been saved. The filename extension is normally supplied by the user, and is separated from the filename by a period. However, PK TEACH adds the extension **".PGR"** (which stands for program) automatically to any program that the user saves while working with it. Therefore, it is not necessary to include an extension when naming your own programs in PK TEACH.*

____f. Type in the filename you have selected and press **ENTER**. Note: Before choosing a filename, you can check the Directory for a list of filenames already on the disk by typing "**?**" and then pressing **ENTER**. Or, you can press **ENTER** by itself, in which case, the Main Menu appears. If you enter an invalid filename, the "**Whoops!**" window in Fig. 49a appears. If you call up the Directory with no PGR-type programs saved on the disk, the "**Whoops!**" window in Fig. 49b appears. If you enter a filename that already exists, the "**Whoops**" window in Fig. 49c appears.

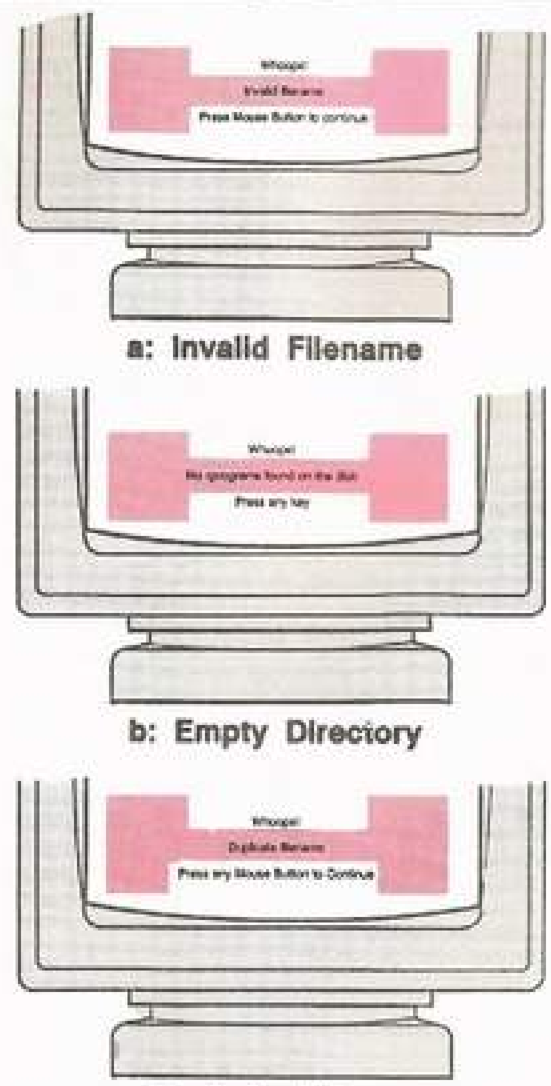
*Note: Press any key or mouse button to get back to the **"Filename"** window, and begin again. If you continually arrive at the invalid filename window, check the PK TEACH disk to see if it has a tape over its write-protect notch. If it does, you will have to remove it to allow your filename to be saved.*

____g. Press **ENTER** after deciding on the name of your file.

____h. If you have been using the keyboard, the familiar five-window arrangement is displayed. If you are using the mouse, what question appears on the screen?

____

____

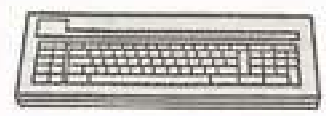


a: Invalid Filename

b: Empty Directory

c: Duplicate Filename

Fig. 49: The "Whoops!" Windows



As shown in Fig. 50, the "**Movement Control**" window has become the "**Writing**" menu, at the top of which is a line which indicates that the program named by the user is now being written. Notice that PK TEACH has added the extension "**.PRG**" to the filename you supplied. Below the program line are six command choices, including the familiar Exit. What are the other five command choices?

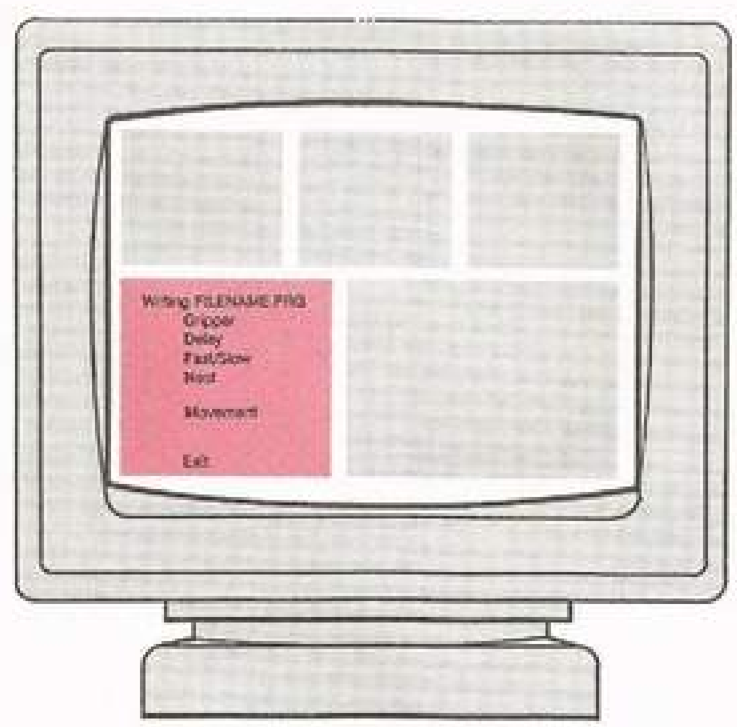


Fig. 50: The Writing Menu for the Keyboard

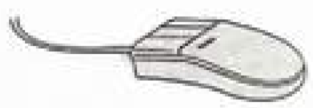
____

____

____

____

____



Click the **LEFT** button and notice the differences in the choices given in the parameter list. The layout at first appears to be identical to the **"Pendent Control"** panel we used in the last exploration (see Fig. 33). However, if you double-click the **CENTER** mouse button, the **"Pendent Control"** panel will appear as shown in Fig. 51. Which new command choices now appear?

____

____

One of the commands which was listed in the second window of the **"Pendent Control"** panel for the **"Move robot with mouse"** parameter (see Fig. 46) is not displayed. Which command is this?

____

____ i. Review the following:

The ______________ command gives you the choice of either ______________ or ______________ the end effector.

It also provides a "________________" window to double-check your decision, as shown in Fig 52.

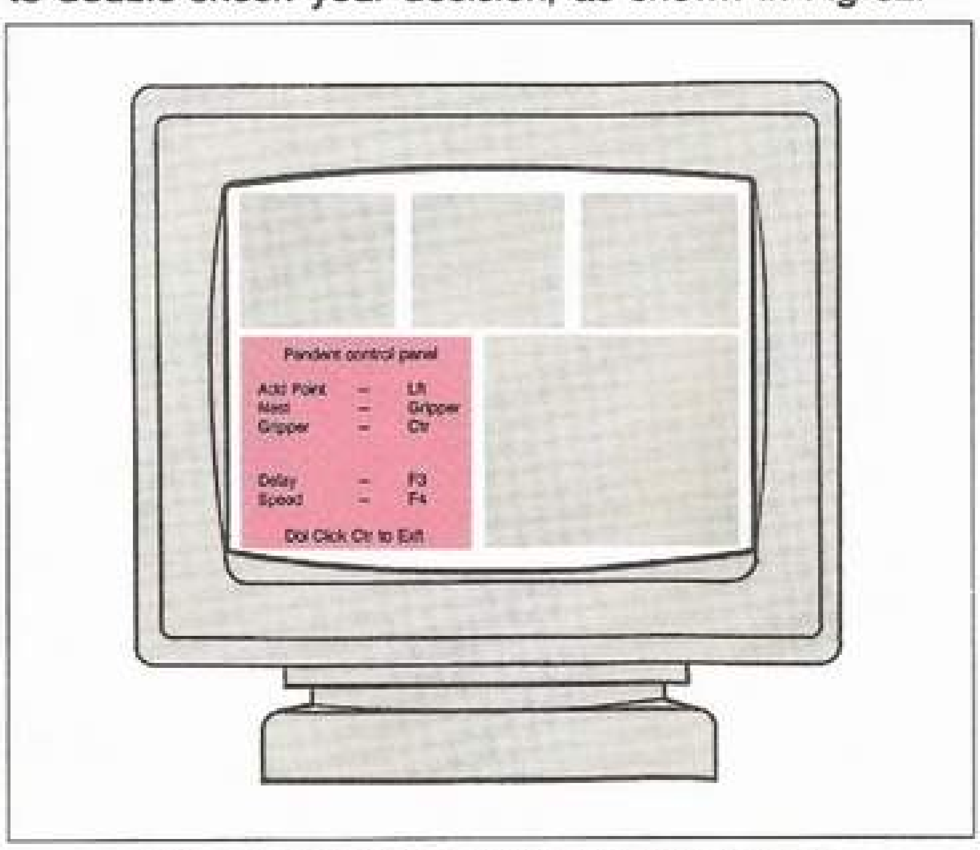


Fig. 51: The Writing Pendent Control Panel

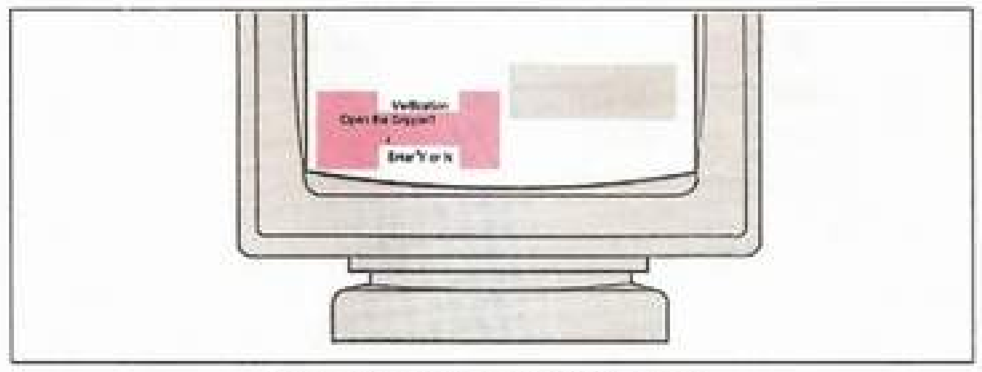


Fig. 52: Gripper Verification

The ________________ command allows you to stop the Pro-Arm from doing anything for a time period, which varies between __________ and __________ seconds, as shown in Fig. 53.

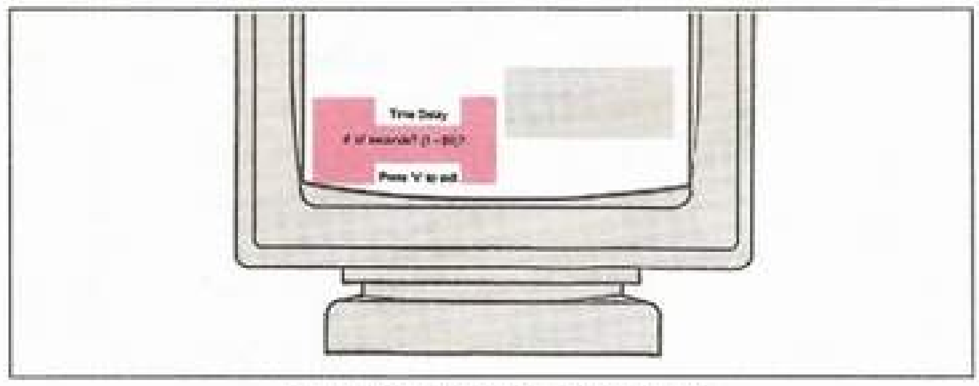


Fig. 53: Time Delay Setting

The ______________ command allows you to vary the ________________ of the Pro-Arm. There are __________ ______________ to choose from, as shown in Fig. 54.

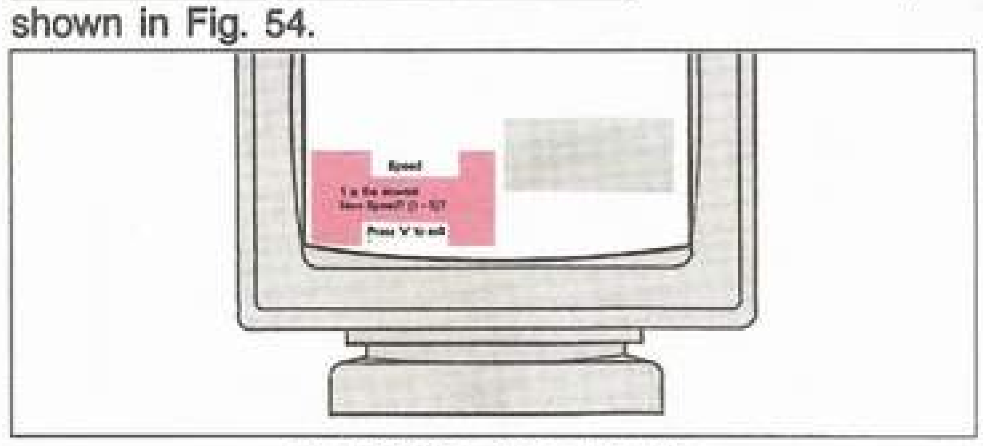


Fig. 54: The Speed Setting

The ________________ command will move the Pro-Arm to its ______________ position. It also gives you a chance to change your mind before entering the command, as shown in Fig. 55.

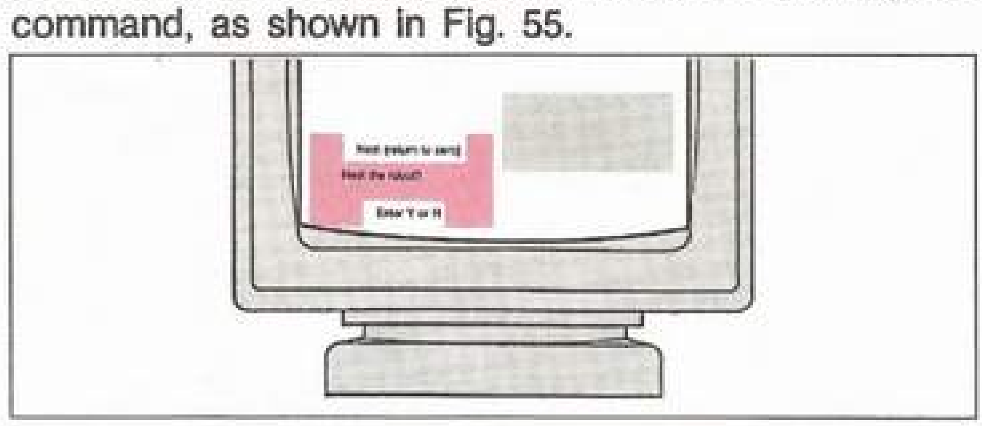


Fig. 55: The Nest Verification

In order to select a new position for the Pro-Arm, the ________________ command must be selected from the menu, after which the movement parameter list is displayed, as shown in Fig. 56. It now contains a new command called ______________ ________ __________.

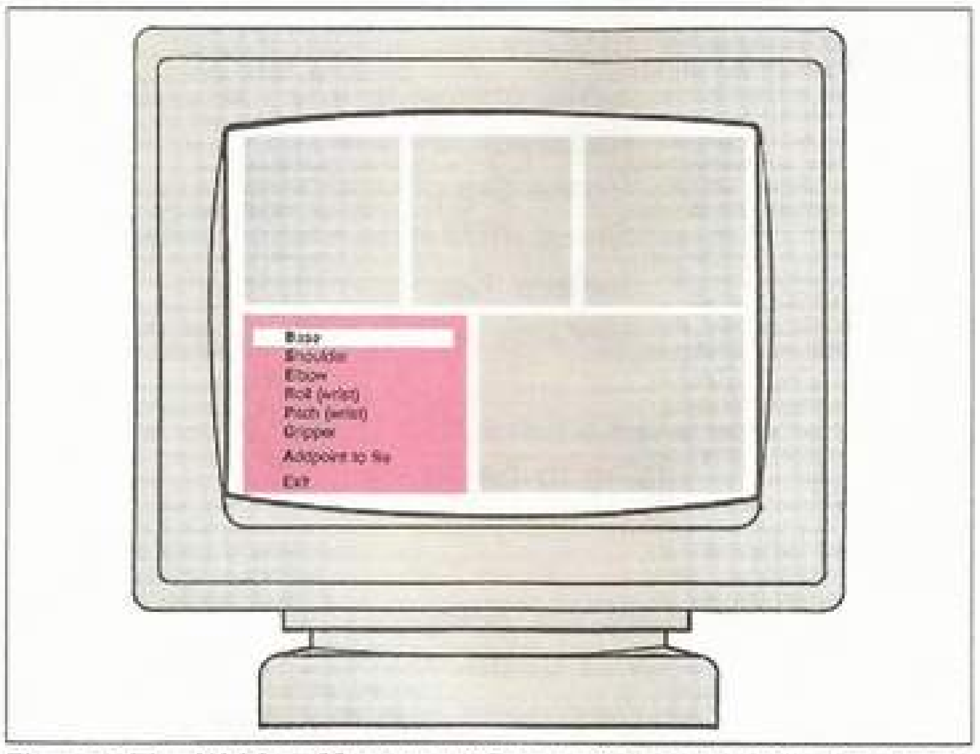


Fig. 56: The Writing Movement Parameter List for the Keyboard

When the Pro-Arm has been moved to a position which must be remembered, the ______________ ______

______________ command will save this point in memory.

*Note: If you try to save an identical position twice without having moved the Pro-Arm, the **"No net change in position"** message is displayed.*

After moving the Pro-Arm to a position you want to save, you must double-click the ____________ mouse

button to display the part of the menu with the __________________ command. Refer to Fig. 51. To

make the Pro-Arm move slower or faster, select the __________________ command by pressing _______.

Refer to Fig. 54. To make the Pro-Arm assume the **HOME** position at various points in a program, you must use the _________________ command. Refer to

Fig. 55. To make the Pro-Arm stop all movement for a specified length of time, you must use the _________

command by pressing ________. Refer to Fig. 53.

*Note: When using the keyboard, you must **"Press 'x' to Exit"** the selection windows. When using the mouse, you must click the **"RIGHT button to exit."***

____j. At this point, if you haven't already done so, it would be a good idea to think about what you would like the Pro-Arm to do in this, your first program.

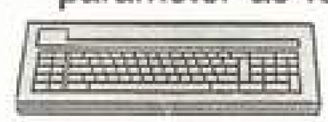
*Note: For example, picking up one of the test blocks from grid **"D19"** and placing it on grid **"H7"** would be a good starter program. Try to picture in your mind's eye exactly what movements you want the Pro-Arm to perform. This will be your program, so think it through.*

____k. **Write** your program for the Pro-Arm now, moving the arm to various points and then adding them to the sequence.

Note: Remember that you can change the increment setting for any of the movement parameters, if more precise positioning is required. The Pro-Arm will carry out the movements as you request them. You do not have to save a point position unless it is the one you want. You can always change the position before adding the point to the program. Mistakes can be corrected during the editing process.

____l. After saving the last positional point in your program, add the **"Nest"** command in order to finish your program with the Pro-Arm in its **"Home"** position.

____m. Exit the **"Write a new program"** menu parameter as follows:



First, select **"Exit"** from the **"Movement"** menu and press **ENTER**. Then, select **"Exit"** from the first **"Write a new program"** menu and press **ENTER** again. You will see the **"Verification"** window with the question, **"Exit WRITE?"** Press **"Y."** You will be asked if you want to save the program you just wrote. If you are happy with the way it turned out, press **"Y."** Otherwise, press **"N"**, which will discard the program. You should now be back to the Main Menu.

First, after saving your final positional point, you must double-click the **CENTER** mouse button to return to the first **"Pendent Control"** panel. Then, click the **RIGHT** mouse button. You will be asked if you want to save the program you just wrote. If you do, click the **LEFT** mouse button; if you don't, click the **RIGHT** mouse button. You should now be back to the Main Menu.

____n. Select **"Exit"** from the Main Menu and verify your decision to leave PK TEACH.

____o. Turn the Pro-Arm power supply off, and remove the PK TEACH disk from the disk drive, placing it in its protective sleeve.

____p. Turn the computer system off, but leave the system set up with all the interface connections intact.

____q. This completes Lab Exploration 12, "Write It." Have your instructor initial your Knowledge Transfer Guide.

EDITING A PROGRAM

Now that you have written and saved a Pro-Arm program of your own, it's time to learn the next step in the program development process.

Rarely does a new program behave exactly as it is expected to. There are always little mistakes ***(bugs)*** which must be removed and corrections which need to be made before the program is ready for an operational test. The easiest way to do this is to run the program through the ***editor***, with the Pro-Arm attached, of course, to examine and evaluate the execution of each command. Any mistakes which show up can be corrected.

As with the other options in PK TEACH, the ***NUM LOCK*** key will be checked to make sure it's deactivated, and the ***ZERO*** position of the Pro-Arm will be questioned as well.

LAB EXPLORATION 13

As you begin each editing session, you will be given a list of valid user files in the PK TEACH directory from which to choose. When you see the program you want to edit, move the cursor over it and select it. If the program you are looking for is not on the list, then it may be in another directory. You will need to exit PK TEACH and use your DOS commands to locate the file you need, and move it into the PK TEACH directory. Then, you can restart PK TEACH and begin editing.

Once you have selected the program you want to edit, you will be placed in the same screen of windows that you have already become familiar with, although several more commands will be available which were not on the menu during the ***"Write a new program"*** session.

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____13. EDIT IT

Purpose: This exploration will give the student practice in editing previously written programs for the Pro-Arm using the PK TEACH software disk.

Equipment: IBM-compatible computer system, Serial mouse, 9 to 25-pin adapter (if required), Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Grid Sheet, IBM-type Pro-Arm Parallel Interface Cable, Knowledge Transfer Manual, and the PK TEACH Disk

____a. Check the system installation and interfacing to be sure that it is ready to go. *Note: Everything should still be set up, as it was in the previous exploration.*

____b. Boot the computer system and apply power to the Pro-Arm.

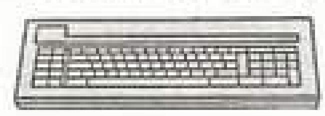
____c. At the prompt **(A>_)**, remove the _____ disk, insert the PK TEACH disk, type **PKTEACH,** and press __________.

____d. Press any key and at the Main Menu, select **"Edit a program,"** and call it up **(click or ENTER)**.

____e. After verifying that the Pro-Arm is in its **"HOME"** position, select the program you wish to edit. *Note: At this time, unless you have been writing other programs, there should only be one user-supplied program available to edit.*

____f. Make your pendent selection if using the mouse.

____g. Complete the following:



In the "______________ ______________" window is the "______________" menu, similar to the **"Writing"** menu we saw in the **"Write a new program"** session.

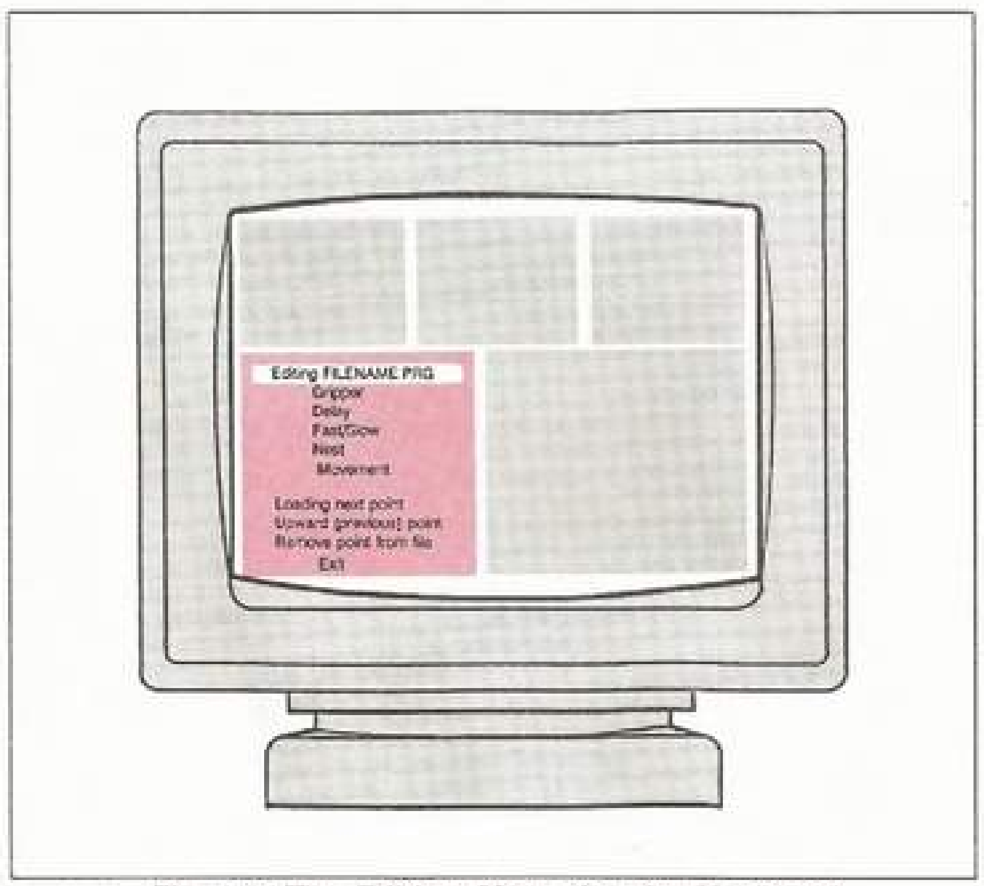
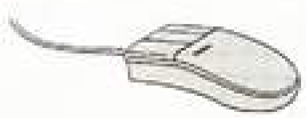


Fig. 57: The Editing Menu for the Keyboard

The top line of the menu now indicates that a/an ______________ session is taking place on the specific file listed.

There are three new commands in the menu, shown in Fig. 57. One of them, the __________ __________ __________ command, allows the user to single-step the Pro-Arm through each movement. In this way, any flaws in the Pro-Arm's actions can be detected easily.

To move one step backwards in the program, the user must use the __________ __________ ________ command, and to get rid of an unwanted command, the __________ __________ __________ __________ command must be used.



To see the edit commands in the "______________ ________________" panel, as depicted in Fig. 58, the user must ________________-_______________ the ____________ mouse button.

To single-step to the next programmed Pro-Arm position, the user chooses the __________ __________ command by clicking the __________ and __________ mouse buttons together.

The __________ __________ command will step the Pro-Arm back one step in the sequence. Click the __________ and __________ mouse buttons together to select it.

The __________ __________ command will remove an unwanted point from the program when the user clicks the __________ and __________ mouse buttons together.

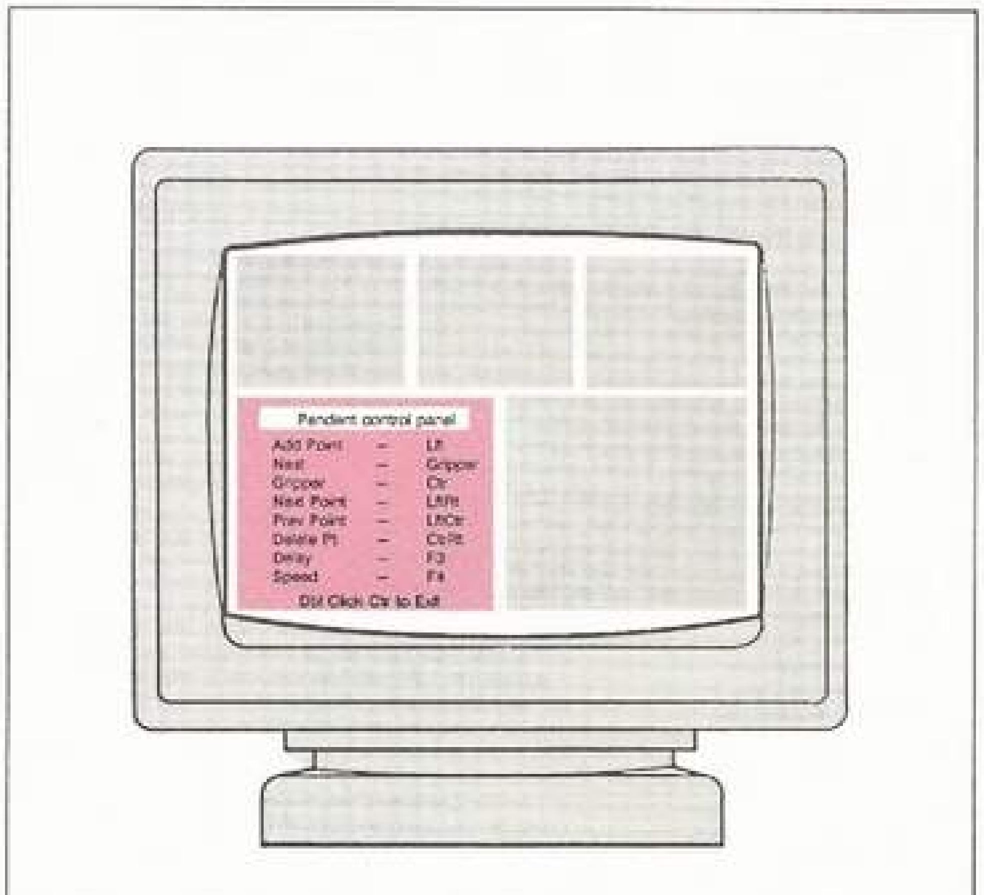


Fig. 58: The Editing Pendent Control Panel for the Mouse

____h. Single-step the Pro-Arm through the program you wrote, using the editing commands as required. If you need to remove a point from the program, be sure to write down what happened in the space provided below.

____

____

Note: When correcting a bad command in the program, always delete the bad command first, before inserting the corrected position with the "Movement" command. The Pro-Arm needs to be in the position previous to the deleted one, before you can correctly edit the movement sequence.

____i. Keep working until you are satisfied with the results of your editing session. You can make the program more complex if you wish.

____j. When you are satisfied with your results, select "Exit," and verify.

____k. Decide whether or not to save your changes and select.

____l. Back at the Main Menu, select "Exit," and verify.

____n. Turn the Pro-Arm power supply off, and remove the PK TEACH disk from the disk drive, placing it in its protective sleeve.

____o. Turn the computer system off, but leave the system set up with all the interface connections intact.

____p. This completes Lab Exploration 13, "Edit It." Have your instructor initial your Knowledge Transfer Guide.

RUNNING A PROGRAM

Once a program has been written, saved, and edited, it can be run using the PK TEACH *"Run a program"* option. This option does not allow for any single-stepping or alteration of the program commands. By the time this option is employed, the user program should have been tested and approved. This would require the editing and removal of any movements which would tend to damage the Pro-Arm or other nearby equipment.

As in the *"Edit a program"* option, the user is given a list of filenames to choose from. Once a program is selected for execution, PK TEACH will load the selection and immediately begin executing it.

The screen layout is similar to the one presented in the previous options already discussed, in that the three position windows are at the top half of the screen. The difference is in the bottom half of the screen, which is completely dominated by the *"Running"* window, as shown in Fig. 59. The *"Running"* window displays the name of the program which is executing, as well as the individual commands as they are being executed by the Pro-Arm. An example line of code is shown in the figure.

When you feel you have a program thoroughly debugged and ready to *"show off,"* run the following exploration for your instructor.

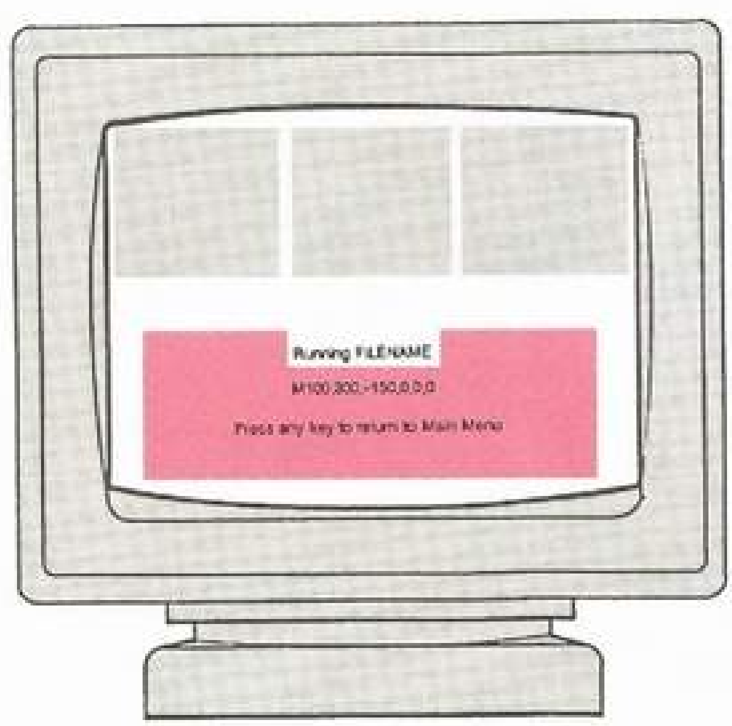


Fig. 59: The Running Window

LAB EXPLORATION 14

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____14. RUN IT

Purpose: This exploration will give the student the opportunity to demonstrate a previously written program for the Pro-Arm robot using the PK TEACH software disk.

Equipment: IBM-compatible computer system, Serial mouse, 9 to 25-pin adapter (if required), Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Grid Sheet, IBM-type Pro-Arm Parallel Interface Cable, Knowledge Transfer Manual, and the PK TEACH Disk.

____a. Check the system installation and interfacing to be sure that it is ready to go. *Note: Everything should still be set up, as it was in the previous exploration.*

____b. Boot the computer system and apply power to the Pro-Arm.

____c. At the prompt **(A>_)**, remove the ______disk, insert the ____________ disk, type ________________, and press________________.

____d. Press any key and at the Main Menu, select **"Run a program,"** and call it up **(click or ENTER)**.

____e. After verifying that the Pro-Arm is in its "__________" position, select the program you wish to run.

____f. Does the program begin to execute immediately?

____

Does the program run properly, as expected?

____

Caution: If not, you may have to repeat Lab Explorations 12 and 13.

____g. Back at the Main Menu, select **"Exit,"** and verify.

____h. Turn the Pro-Arm power supply off, and remove the PK TEACH disk from the disk drive, placing it in its protective sleeve.

____i. Turn the computer system off, but leave the system set up with all the interface connections intact.

____j. This completes Lab Exploration 14, "Run It." Have your instructor initial your Knowledge Transfer Guide.

DELETING A PROGRAM

Every now and then, a user program will creep into the directory, which either doesn't belong there or no longer performs the useful function it once did. It then becomes a candidate for removal or deletion. Once a program has been deleted, it is gone forever. For this reason, be absolutely sure that the program being deleted is backed up on another disk, or is simply not wanted any longer, period.

LAB EXPLORATION 15

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____15. DELETE IT

Purpose: This exploration will give the student the opportunity to remove a previously written Pro-Arm program, using the PK TEACH software disk.

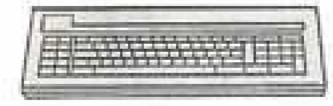
Equipment: IBM-compatible computer system, Serial mouse, 9 to 25-pin adapter (if required), Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Grid Sheet, IBM-type Pro-Arm Parallel Interface Cable, Knowledge Transfer Manual, and the PK TEACH Disk.

____a. Check the system installation and interfacing to be sure that it is ready to go. *Note: Everything should still be set up, as it was in the previous exploration.*

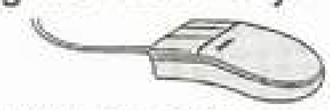
____b. Boot the computer system and apply power to the Pro-Arm.

____c. At the prompt (________), remove the ______ disk, insert the ______________ disk, type ______________, and press ______________.

____d. Press any ______ and at the ______ ______, select **"Delete a program"**, and call it up (**click or ENTER**). Once you have called up the **"Delete a program"** option, you can still change your mind about deleting anything by:



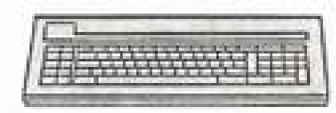
Pressing the **"ESC"** key.



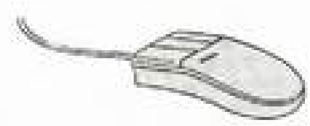
Clicking the **RIGHT** mouse button.

Caution: Do not delete a file you may want to keep, unless you have backed it up on another disk or directory.you have backed it up on another disk or directory.

____e. After checking the file list carefully, to be sure that there is a file in the list that can safely be deleted while doing this exploration, perform the following:



Select a user file to be deleted, and press **ENTER**.



Move the diamond over the file to be deleted, and click the **LEFT** mouse button.

____f. Even after you have picked a program to delete, the **"Verification"** window will check with you once again before finally erasing it. This is to help prevent an accidental erasure. If you are sure about deleting the program you have selected, choose **"Y"** at the **"Verification"** window, and kiss your program good-bye.

____g. Back at the _______ _______, select __________, and ____________.

____h. Turn the Pro-Arm power supply _____, and remove the ___________ disk from the disk drive, placing it in its __________________ ___________.

____i. Turn the computer system ______, but leave the system set up with all the _______________ connections intact.

____j. This completes Lab Exploration 15, "Delete It." Please have your instructor initial your Knowledge Transfer Guide.

UTILITIES

The *"Utilities"* option is actually just a submenu which provides three additional functions. These functions are:

1. Translation of PK TEACH to ASCII.
2. Directory of Programs.
3. Exit to DOS.

TRANSLATION OF PK TEACH TO ASCII

The *"Translation"* option allows you to convert a program written for the Pro-Arm, by PK TEACH, into a standard ASCII or text file. As a result, the program can then be run, using other programming languages, apart from the PK TEACH software package. The commands can also be directly edited with a word-processing program.

When this option is selected, the user is given a list of files from which to choose. The selected file will then be converted into ASCII by PK TEACH, while the program lines are counted. At the conclusion of the translation process, the user is prompted to press any key to return to the Main Menu, as shown in Fig. 60. By this time, a converted version of the file has been added to the current directory, complete with an *.ASC* extension.

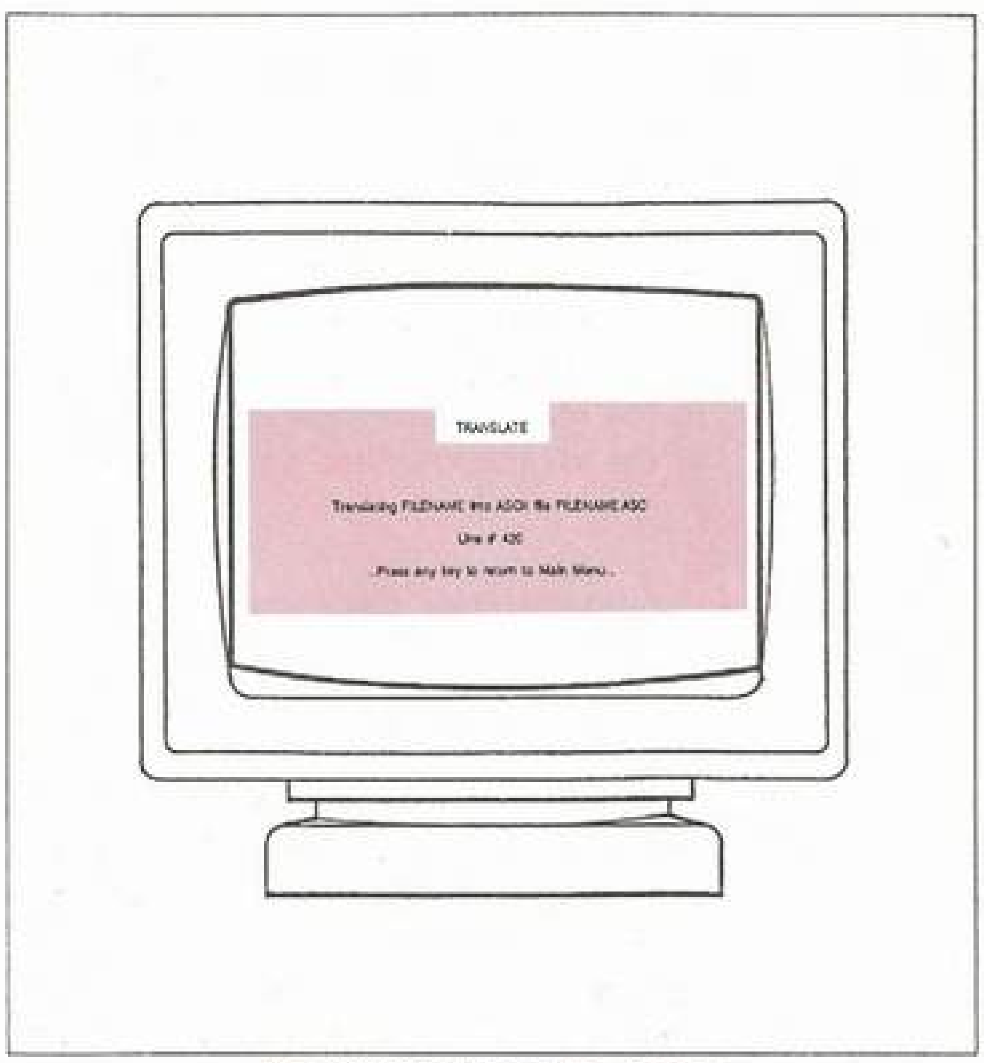


Fig. 60: The Translate Screen

LAB EXPLORATION 16

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____16 TRANSLATE

Purpose: This exploration will give the student the opportunity to translate a previously written program for the Pro-Arm, using the PK TEACH software disk, into a standard ***ASCII*** or text file.

Equipment: IBM-compatible computer system, Serial mouse, 9 to 25-pin adapter (if required), Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Grid Sheet, IBM-type Pro-Arm Parallel Interface Cable, Knowledge Transfer Manual, and the PK TEACH Disk.

____a. Check the system ______________ and ______________ to be sure that it is ready to go. *Note: Everything should still be set up, as it was in the previous exploration.*

____b. Boot the computer system and apply power to the Pro-Arm.

____c. At the prompt (______), remove the ______ disk, insert the ____________ disk, type ____________, and press ____________.

____d. Press any ______ and at the ______ ______, select **"Utilities,"** and **press or click.**

____e. At the submenu, select **"Translate PK TEACH file to ASCII,"** and **press or click.**

____f. Choose the file to be translated, and **press or click.** Does the window display the action?

____

Are the number of lines counted?

____

Note: Fig. 60 shows a sample number (420). This number flashes rapidly as the program is translated.

____g. Press any key and return to the Main Menu.

____h. Back at the ________ ________, select ________, and __________.

____i. At the DOS command line, type **DIR** to list the files on the PK TEACH disk.

____j. Is there a file listed with the .ASC (ASCII) extension?

____

Does it have the same filename as the filename with the .PRG extension which you created previously by using PK TEACH?

____

____k. Turn the ____________ ____________ ____________ __________, and remove the __________ __________ from the ________ ________, placing it in its ______________ __________.

____l. Turn the __________ ________ ______, but leave the __________ set up with all the ______________ connections intact.

____m. This completes Lab Exploration 16, "Translate." Have your instructor initial your Knowledge Transfer Guide.

DIRECTORY OF PROGRAMS

The "**Directory**" option simply gives the user a display of all the PK TEACH user files in the current directory. Any key or mouse button returns the user to the Main Menu.

LAB EXPLORATION 17

____17. DIRECTORY

Purpose: This exploration will give the student practice in using the directory utility of the PK TEACH software disk.

Equipment: IBM-compatible computer system, Serial mouse, 9 to 25-pin adapter (if required), Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Grid Sheet, IBM-type Pro-Arm Parallel Interface Cable, Knowledge Transfer Manual, and the PK TEACH Disk.

____a. Check the system ____________ and ______________.

____b. Boot the ______________ system and ________ ________ to the Pro-Arm.

____c. At the prompt (______), remove the ______ disk, insert the ____________ disk, type ____________, and press _________. Press any ______.

____d. At the ______ ______ select "**Utilities**", and **press or click**.

____e. At the submenu, select "**Directory of Programs,**" and **press or click**.

____f. After reviewing the user files, press any key or click any mouse button to return to the Main Menu. Were any other files in the directory other than user files?

____

Note: Only user-created files are listed.

____g. Back at the Main Menu, select "Exit," and verify.

____h. Turn the Pro-Arm power supply off, remove the PK TEACH disk from the disk drive, and place it in its protective sleeve.

____i. Turn the computer system off.

____j. This completes Lab Exploration 17, "Directory." Have your instructor initial your Knowledge Transfer Guide.

EXIT TO DOS

This feature allows the user to go back to the DOS command line, perform any necessary function, and then return directly to PK TEACH by typing the word *"EXIT"* at the prompt.

LAB EXPLORATION 18

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____18. SHELL

Purpose: This exploration will give the student practice in using the ***DOS*** shell utility of the PK TEACH software disk.

Equipment: IBM-compatible computer system, Serial mouse, 9 to 25-pin adapter (if required), Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Grid Sheet, IBM-type Pro-Arm Parallel Interface Cable, Knowledge Transfer Manual, and the PK TEACH Disk.

____a. Check the system ____________ and ______________ to be sure that it is ready to go.

____b. Boot the _____________ system and _______ _______ to the Pro-Arm.

____c. At the prompt (_______), remove the ______ disk, insert the ____________ disk, type _____________, and press __________. Press any _____.

____d. At the ______ ______, select **"Utilities,"** and **press or click.**

____e. At the Submenu, select **"Exit to DOS,"** and **press or click.**

____f. The screen is now back to the DOS command line. What message is displayed directly above the DOS prompt?

____

____

____g. Type **EXIT** and press **ENTER**. The _____________ ____________ appears.

Note: Although the utilities submenu is superimposed over part of the Main Menu, the Main Menu is still active.

____h. Exit from the Main Menu and verify.

____i. Turn the Pro-Arm power supply off, remove the PK TEACH disk from the disk drive, and place it in its protective sleeve.

____j. Turn the computer system off.

____k. This completes Lab Exploration 18, "Shell," and the PK TEACH SOFTWARE activities. This also completes SECTION III of your Knowledge Transfer Project. Have your instructor initial your Knowledge Transfer Guide.

SECTION IV

PROGRAMMING THE RS-2200 PRO-ARM

DISCUSSION

As you already know the MARCRAFT RS-2200 P/A Pro-Arm system comes with the PK Teach software, with which programs for the Pro-Arm may be easily written, edited, saved, and executed. All of this can be done without any knowledge of programming whatsoever, using either the keyboard or a serial mouse (included). There are, however, several features built into the Pro-Arm which can only be utilized by those writing your own programs.

This section is included for those students who have an interest in learning more about programming industrial robots, using the MARCRAFT RS-2200 Pro-Arm.

As we mentioned earlier, in Section III, one of the main advantages of high-technology industrial robots over hard-automation equipment, and even low-technology robots, is their ability to learn new functions and routines. They can do this because of the fact that they are controlled by microprocessors. The microprocessors, in turn, are controlled by the specific programs ***(software)*** written for them by programming engineers. It takes patience and skill to write successful programs for industrial robotics applications. Many programmers are computer science graduates, and others have degrees in several fields. Programmers are highly paid professionals, and the future looks very promising for those who choose this kind of work.

The field of advanced robotics programming makes use of various forms of high-level mathematics. Among these are Homogeneous Transformations, Kinematic Equations, Laplace Transformation, and Lagrangian Dynamics. These mathematical studies basically concern themselves with precisely defining the joint movements of an industrial robot. This is necessary so that the end-effector can be moved to a specific, and minutely measured, point within the robot's work envelope. These mathematical studies are beyond the scope of this project, and are not required to achieve rewarding results with the RS-2200 Pro-Arm.

While the Pro-Arm can be programmed using many languages, we recommend ***BASIC,*** for those who wish to pursue this subject for the first time. The ***BASIC (Beginner's All-Purpose Symbolic Instruction Code)*** programming language is universally understood, and it has been chosen for use with the RS-2200 P/A Pro-Arm Knowledge Transfer Project.

Most all IBM-compatible computer operating systems include a version of ***BASIC***. Remember to load the particular version of ***BASIC*** that came with your computer system, when performing the explorations in this section.

We have already discussed the subject of robotics in general, and have reviewed the three basic categories of robots used in industry. Now, it's time to look more closely at the MARCRAFT Pro-Arm. Its full range of capabilities should be thoroughly understood before attempting to program it. While the PK TEACH software makes it easy to write a program for the Pro-Arm, it doesn't give us much information about how the Pro-Arm really works, nor does it utilize some of its more unique features, which can only be accessed by directly programming the Pro-Arm.

Note: At the end of this manual is a glossary with definitions of robotics terms. When words appear in the text which are unaccompanied with sufficient explanations or descriptions, please turn to the glossary for further clarification.

When comparing the capabilities of the MARCRAFT Pro-Arm with those listed for the three general categories of industrial robots, it may seem that the Pro-Arm doesn't really fit completely into any one of them. This is because the Pro-Arm was designed to be a small version of the very large robots presently used in industry for general-purpose material handling and manufacturing applications. It makes an ideal robot for many educational applications because of its low cost, large working envelope, comparatively large payload capacity, and small floor space requirements (see the **SPECIFICATIONS** in Section VI). Since it can simulate industrial robot operation, the Pro-Arm is most often used in laboratory and classroom training, as well as research.

The MARCRAFT Pro-Arm is a general-purpose, point-to-point, five-axis, jointed-arm, stepping-motor-controlled, industrial-type robot. It is called a joint-coordinate type of robot because each of its five axis and gripper imitate the movements and actions of a corresponding human joint. The joint-coordinate type of robot is most commonly found in industrial robotics applications.

The power source, usually called an ***Actuator,*** is some type of device used to drive the robot. The three basic types of actuators used today are: (1) ***hydraulic,*** (2) ***electric,*** and (3) ***pneumatic.***

The mechanical arm mechanism of a jointed-arm robot is called the ***Manipulator***, and usually consists of a base, an arm having a series of axes or joints, and a power source. The manipulator is used to move a device, called an ***End-Effector***, in any number of directions.

The end-effector most frequently used is a ***hand*** or ***Gripper***. It effectively moves materials or holds various parts and tools. End-effectors may have many different sizes, shapes, and forms depending upon the specific chore which the robot must perform. They may be a simple gripping device, a vacuum-type holding apparatus, or even a magnet.

The number of joints in a robotics arm is often referred to as the number of ***"degrees of freedom."*** The degrees of freedom are actually the number of different ways in which the end-effector can be moved by the use of the arm's axes. Here is a further elaboration on the Pro-Arm's axes which were introduced in Lab Exploration 1.

The first axis ***(Base Axis)*** corresponds to the human waist and rotates the ***"body"*** about the base. This axis provides ***"arm sweep"*** and only moves in a ***horizontal*** plane. The second axis ***(Shoulder Axis)*** corresponds to the human shoulder and provides ***"shoulder swivel."*** This joint permits movement only in a ***vertical*** plane.

The third axis ***(Elbow Axis)*** is similar to the human elbow and provides ***"elbow extension,"*** which allows the end-effector to be positioned at different distances from the body. The fourth axis ***(Wrist Roll Axis)*** is similar to a ***rotating*** human wrist. The fifth axis ***(Wrist Pitch Axis)*** corresponds to the ***bending*** of a human wrist.

The MARCRAFT RS-2200 P/A Pro-Arm robotics system uses an IBM or IBM-compatible computer (with either keyboard or mouse) for a ***"teach device"*** when operating the Pro-Arm, a built-in (on-board) microprocessor (a Z80 mounted in the base unit) for the ***"controller,"*** electric stepping motors (powered by an external power supply) for ***"actuators,"*** disks (either MARCRAFT-supplied software or user-created programs) for ***"permanent storage,"*** and a gripper for an ***"end-effector."*** See the block diagram shown in Fig. 61.

In order to take advantage of all the features incorporated into the Pro-Arm system, you may wish to use a programming language to write your own routines. For those who wish to program the Pro-Arm directly, a brief description of the ***BASIC*** language commands required to operate the Pro-Arm are included, along with lab explorations to demonstrate their use. Of course, advanced programmers may use any language with which they are familiar.

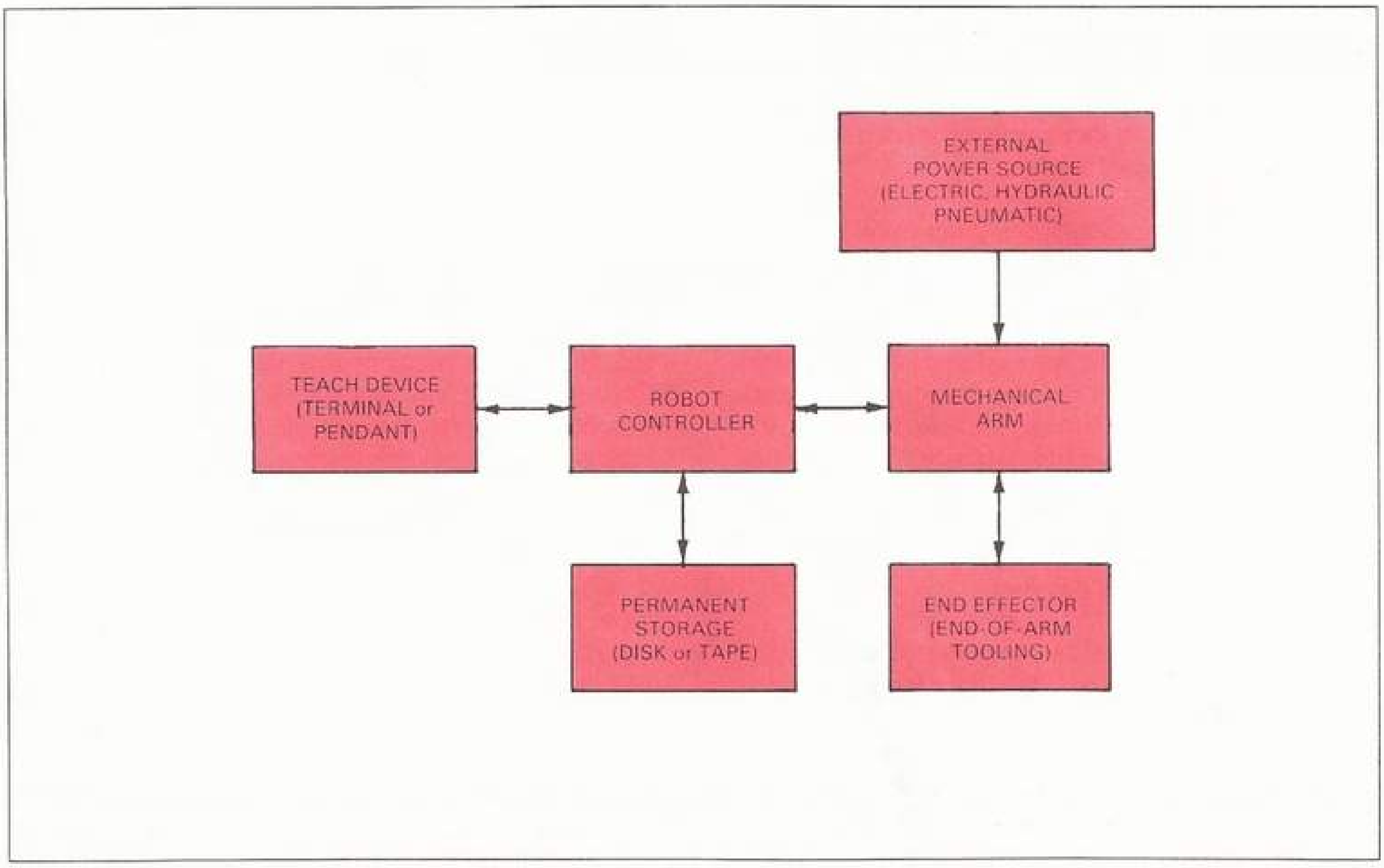


Fig. 61: Marcraft Pro-Arm Block Diagram

The computer commands are sent to the robot controller from the computer (IBM-compatible) through any parallel-printer port that transmits ***ASCII*** character codes. These commands manipulate the axes and gripper, provide time delays, select the speed of operation, control the output ports and limit switches, and store data in memory.

External devices, such as conveyor belts, motors, and sensors, can be software controlled by the Pro-Arm using four external input switches and the 4-bit output port. A software-controlled external interlock switch can also be connected to the robot. An 8-bit output port is provided for connecting an external storage device, such as a disk drive or a memory expansion board.

Although writing programs in ***BASIC*** does not require a full knowledge of how the Pro-Arm robot works, the use of some other programming languages definitely would. Because of this, a complete explanation of how the Pro-Arm works follows.

This explanation, and the information on the ***BASIC*** programming language, will be presented in the following sequence:

1. Parts and Performance
2. Other Interfacing Features
3. BASIC: A Short History
4. A BASIC Demo
5. Command Line Structure
6. Pro-Arm Commands
7. Work Envelope
8. Repeatability
9. Programming Interface Features
10. Your First BASIC Program

PARTS AND PERFORMANCE

DISCUSSION

Fig. 62 shows a block diagram of the electronic portion of the Pro-Arm. The major components or subsystems are as follows:

1. CPU Microprocessor
2. RAM Memory
3. ROM Memory
4. Motor Controllers
5. Stepping Motors
6. Parallel Interface
7. Power Supply

All of the components, with the exception of the power supply, are located within the robot.

CPU MICROPROCESSOR

The CPU ***(Central Processing Unit)*** is the ***"brain"*** of the Pro-Arm. The CPU directs and controls all Pro-Arm functions, including memory Read and Write, data input and output, and motor control. A ***Z80*** (manufactured by ***Zilog, Inc.***) microprocessor CPU chip, commonly found in 8-bit microcomputers, is used.

Data, in the form of 8-bit Pro-Arm commands ***(bytes)***, is transferred through a ***"data bus"*** between the various components, including the parallel interface, the CPU, the motor controllers, the input/output devices, and memory. The data bus is a two-way avenue of information; that is, data can flow to and from the CPU and memory, or between memory and the motor controllers, etc.

The Z80 microprocessor has 8 data lines on its data bus and 16 address lines on its ***"address bus."*** Notice that the address bus is only a one-way avenue to the various components from the CPU. This is because the CPU doesn't need to receive any address information from the various other components. It alone decides what addresses will be written to or read from. Early computer systems used 4-bit buses to transfer data between different components. Those 4-bit chunks of data were called ***"nibbles."*** It is easy to see that it takes two nibbles to equal one ***byte***. The address information is also in the form of bytes, just as is the data information. The only difference is that the address information is 16-bits wide ***(two bytes or one word)***. These bytes or words are simply made up of groups of voltage levels (either ***0 volts*** or ***5 volts***). The groups of voltage levels are interpreted as binary numeric values (either ***1 or 0***) or control codes, which have specific meanings.

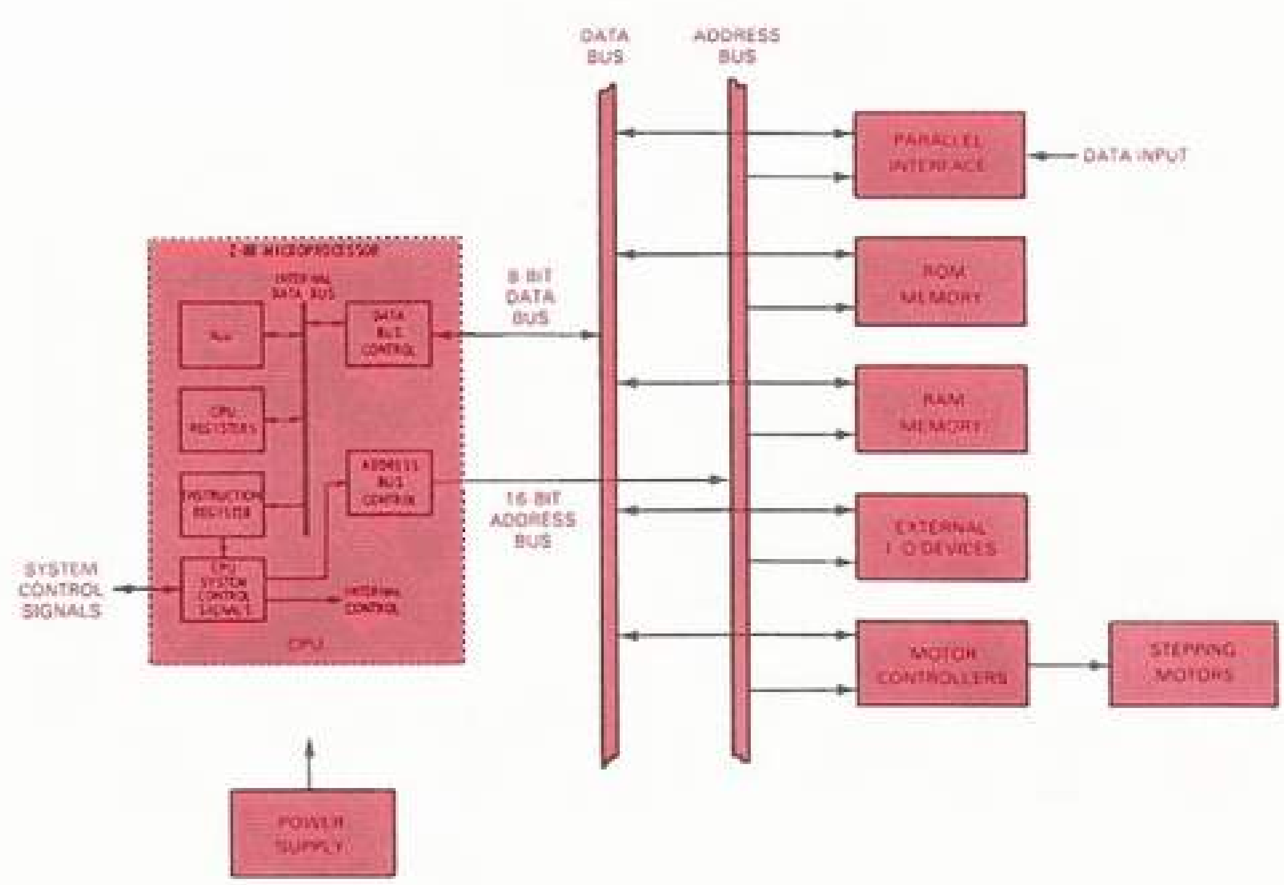


Fig. 62: RS-2200 Electronic Block Diagram

In the case of data, the 8-bit bytes are interpreted as depicted in the ***ASCII*** chart on Page 159. Each byte is present on the data bus for only a short period of time, just long enough for the intended component (the component of the system which was addressed or signaled by the information on the address bus) to receive and store it, or send it. In some cases, two devices must be addressed before any data is transferred. One device must be told to send data, while another must be instructed to receive it.

For address information, a 16-bit word is simply interpreted as a knock on the door by the component whose house number matches the 16-bit code. Again, the information remains on the address bus just long enough for the addressed component to open its door.

In many computer systems, there are enough control signals being used to require a special bus ***(the control bus)*** just for them. The RS-2200 P/A Pro-Arm system uses several control signal lines, but a separate bus has not been designated for them.

LAB EXPLORATION 19

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____19. PRO-ARM'S CPU

Purpose: To review the major points discussed about the Pro-Arm's microprocessor.

Equipment: Knowledge Transfer Manual, Pencil

____a. The CPU used by the Pro-Arm is the ____________, which is manufactured by ________________.

____b. A two-way avenue by which information is transferred between the various components of the system is called a/an ____________________.

____c. An 8-bit chunk of data is called a ____________.

____d. A one-way avenue by which information is transferred from the CPU to the various components of the system is called a/an ____________________.

____e. A 16-bit chunk of data is called a ____________.

____f. If it takes ____________ bits of data to equal one nibble, then two nibbles will equal one ______________, and four nibbles will equal one ____________.

____g. The __________ decides which addres ses will be written to or read from.

____h. The __________ directs and controls all of the Pro-Arm's functions.

____i. This completes Lab Exploration 19, "Pro-Arm's CPU." Have your instructor initial your Knowledge Transfer Guide.

RAM MEMORY

RAM ***(Random-Access Memory)*** is actually divided into several different types. The two most commonly used are SRAM ***(Static Random-Access Memory)*** and DRAM ***(Dynamic Random-Access Memory).*** The use of DRAM requires extra circuitry to keep the data from being lost during microprocessor operation. For this reason the Pro-Arm uses SRAM, which is easier to implement. RAM memory is lost whenever the power is turned off or the system is ***RESET***.

In the Pro-Arm, RAM is contained in a 24-pin DIP ***(Dual In-line Package)*** chip inserted into socket ***"D3"*** on the printed-circuit board (see Fig. 63). This chip holds 2048 8-bit locations ***(2K)*** for data. Since memory space is needed for specific programming instructions, the amount of memory space reserved for positional data will hold 100 axes-position commands.

An additional RAM chip may be added in order to expand the memory capacity of the Pro-Arm up to 270 axes positions, doubling the Pro-Arm's existing RAM capabilities from 2048 8-bit locations to 4096 8-bit locations ***(4K)***. Before going any further, find empty IC socket ***"D4"***, located on the PC board underneath the baseplate, as shown in Fig. 63. The following lab exploration can be used when RAM memory expansion becomes desirable.

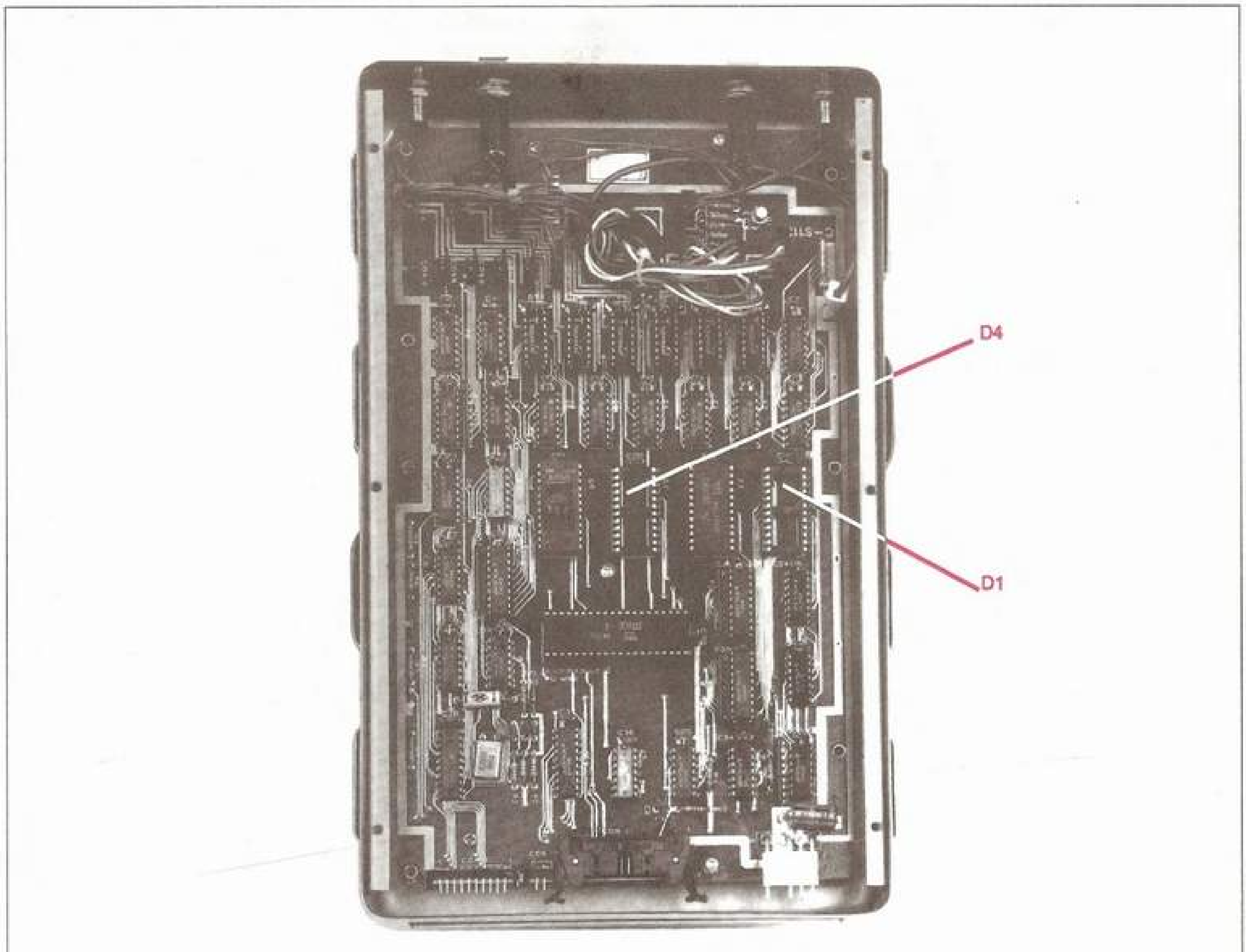


Fig. 63: Memory Area of the PCB

LAB EXPLORATION 20

As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

___20. RAM MEMORY EXPANSION

Purpose: This exploration serves as a guide in the expansion of the Pro-Arm's RAM capabilities.

Equipment: Phillips screwdriver, Spare 4016 or 6116 RAM IC, and the Pro-Arm Robot.

___a. Set the Pro-Arm on its side with the baseplate facing you.

___b. Use the Phillips-head screwdriver to remove the six screws and lockwashers from the baseplate.

___c. Insert a 4016 or a 6116 RAM IC into the "**D1**" socket. (Refer to Fig. 63). *Note: An IC just like it should already be inserted into socket* ***D3***. *Caution: Make sure that Pin #1 on the chip is aligned with the Pin #1 marking on the socket.*

___d. Now use the Phillips-head screwdriver to replace the baseplate with the six screws and their lockwashers. *Caution: Please check the Pro-Arm to be sure it sits level once the baseplate has been reinstalled. If there is room for it to wobble, reorient the baseplate until any wobble is gone.*

___e. This completes Lab Exploration 20, "RAM Memory Expansion." Have your instructor initial your Knowledge Transfer Guide.

ROM MEMORY

There is another type of memory found in the Pro-Arm. This type of memory is called ROM ***(Read-Only Memory).*** The difference between this type of memory and RAM is that ROM memory cannot be written to during normal microprocessor operations. Programs are permanently written on the ROM IC chips before they are ever installed in the Pro-Arm. ROM IC chips come in several different types also. The less expensive variety are programmed for specific purposes at the factory. Their programs can never be erased or altered. They are the standard ROM chips.

Another type called a PROM ***(Programmable Read-Only Memory)*** can be programmed by the user, using a special piece of equipment called a ***"PROM Burner."*** If a mistake is made during the programming process, the chip cannot be used and must be discarded. A variation of the PROM is the EPROM ***(Erasable Programmable Read-Only Memory)***, which can be completely erased and reprogrammed if necessary, thereby allowing a programming error to be corrected without having to discard the chip.

Another variation is called an EEPROM ***(Electrically Erasable Programmable Read-Only Memory).*** This chip can be selectively erased and reprogrammed using specialized programming equipment. Because of their versatility, EPROMs and EEPROMs are the more expensive types of ROM.

The two test programs, which you performed during the Operational Testing of Section II, are located inside of a 2732 EPROM IC which is inserted into socket *"D6"*. This IC contains 4096 8-bit locations. Another 2732 EPROM, or equivalent IC, can be inserted into the *"D4"* socket, to provide for additional internally contained programs for the Pro-Arm. These programs can be specified by the user. Since the Pro-Arm uses a Z80 microprocessor, the user should either use a known, valid program which has been written for the Z80, or have some personal programming experience with its command set, when planning to install additional ROM programs.

For addressing information, check the memory map of Fig. 64. It contains a listing of the valid memory addresses being used by the Pro-Arm.

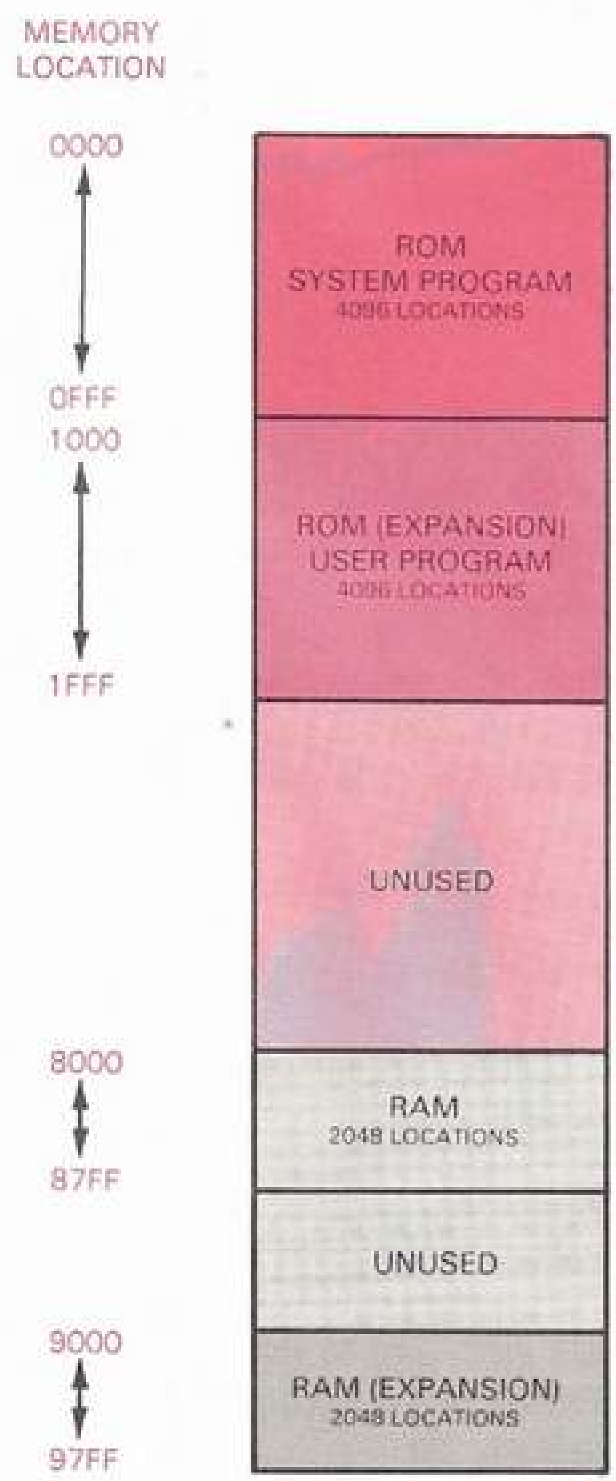


Fig. 64: RAM and ROM Memory Mapping

LAB EXPLORATION 21

If you have a program you wish to install internally on the Pro-Arm, perform the steps in the following lab exploration. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

____21. ROM MEMORY EXPANSION

Purpose: This exploration serves as a guide in the expansion of the Pro-Arm's ROM capabilities.

Equipment: Phillips screwdriver, 2732 EPROM or equivalent IC with program installed, and the Pro-Arm Robot.

____a. Set the Pro-Arm on its side with the baseplate facing you.

____b. Use the Phillips-head screwdriver to remove the six screws and lockwashers from the baseplate.

____c. Insert the 2732 EPROM IC containing the program you wish to internalize into the "D4" socket. Refer to Fig. 63. *Note: An IC just like it, containing the two self-test programs, should already be inserted into socket "D6". Information on the proper programming of the 2732 EPROM is not a part of this project. You will need to familiarize yourself with the Z-80 Microprocessor command set and have some previous programming experience to successfully accomplish this task. You may consult the documentation included with your EPROM burner for further details.*

____d. Use the Phillips-head screwdriver to replace the baseplate with the six screws and lockwashers. *Caution: Check the Pro-Arm to be sure that it sits level once the baseplate has been reinstalled. If there is room for it to wobble, reorient the baseplate until any wobble is gone.*

____e. This completes Lab Exploration 21, "ROM Memory Expansion." Have your instructor initial your Knowledge Transfer Guide.

Each of the six stepping motors is directly controlled by a ***Motor Controller*** circuit consisting of two ICs (a 74LS75 TTL 4-bit Bistable Latch and a 2064 Darlington Array Motor Driver). Each circuit is controlled by 8-bit commands from the data bus. Refer to the schematic diagram and notice each two-chip combination.

For example, ***M1*** (the ***Base Motor)*** is controlled by the ***E1/F1*** combination, while ***M2*** (the ***Shoulder Motor)*** is controlled by the ***E2/F2*** combination, etc. Also note that the 2064 driver chips (or ECG 2085s) are connected to the 12-V dc power supply through two connections at Pins 1 and 8. This is because two separate circuits and motor windings are being used in order to provide the required polarities for each step that the motor takes.

The acronym TTL ***(Transistor-Transistor Logic)*** tells us that the 74LS75 Bistable Latch belongs to the family of digital IC circuits that uses a 5-V dc voltage supply. Since this fact is understood by most technicians, there is no need to show a connection to the 5-V dc supply (and ground) for all the chips in the TTL family.

Since the motors which actuate the base, shoulder, and elbow axes draw more current than the smaller wrist and gripper motors, it isn't surprising to find that the wires connected to the 12-V dc supply for ***M1, M2,*** and ***M3*** are larger than those for ***M4, M5,*** and ***M6.***

The Motor Driver ICs have been known to overheat and fail when the Pro-Arm is being improperly operated, or when a circuit becomes shorted somewhere else on the PCB. This especially holds true for the ***M1, M2,*** and ***M3 Motor Drivers.***

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____22. DRIVING THE MOTORS

Purpose: To review the major points discussed about the Pro-Arm's motor drivers.
Equipment: Knowledge Transfer Manual, Pencil

____a. There are _________ motors installed in the Pro-Arm.

____b. Each motor is controlled by a ______________ combination circuit.

____c. The IC combination _____________ controls M5.

____d. A 74LS75 IC is called a __________ __________ __________, and uses a voltage of ________. It belongs to a family of digital logic IC circuits called _________.

____e. Since the stepping motors require a voltage of _______, the _____________ ICs are wired to that power source.

____f. When comparing the wrist and the base motors, the ________________ motor requires more current to operate.

____g. When the Pro-Arm is being improperly operated, it is likely that the motor driver ICs may __________________.

____h. This completes Lab Exploration 22, "Driving the Motors." Have your instructor initial your Knowledge Transfer Guide.

Six 2-phase bifilar-type ***(unipolar)*** stepping motors are used in the Pro-Arm to manipulate the five axes and the gripper. Since bifilar means "fitted with or involving the use of two wires," it follows that these stepping motors require two power-supply connections and two separate stator windings in order to be able to provide the high degree of rotational accuracy required. Since the current being used is flowing in only one direction ***(dc),*** the construction of the motor is unipolar. That is, each stator winding is ***center-tapped,*** allowing a simpler wiring design to achieve bidirectional operation. The stepping voltages are coded in 4-bit combinations, so that precise movements can be obtained.

A stepping motor has the capability of rotating in either direction, starting or stopping at various positions, and moving its rotor in precise angular increments for each input excitation change or digital step pulse. The precise angular movement is repeated for each input step command, which allows the motor to accurately move its rotor to a known, repeatable position. Refer to Fig. 65 for the specifications of each of the Pro-Arm's axes. An example of the program command is also provided.

Fig. 66 shows the internal construction of the bifilar type of stepping motor. As shown in the schematic, each stator winding is center-tapped to obtain the alternate magnetic polarities.

Unlike the first stepping motors used with the Pro-Arm, recent versions of the wrist and gripper motors are constructed so that the gear train can be separated from the motor by twisting the two halves about a quarter of a turn. This simplifies motor replacement procedures in the event that either the gear train or the motor is bad, but not both.

Four wires from each one of the Pro-Arm's stepping motors are connected to the PCB by a 4-wire plug-in connector. Two wires from each stepping motor are connected ***(soldered)*** to a larger wire, which is plugged onto a post protruding from the Pro-Arm's PCB. This post is wired to the 12-V dc supply line. When disconnecting a motor wiring-harness plug-in connector from the PCB for testing or for motor replacement, always be sure to label each harness connector and make a note of the wire orientation for identification purposes. The motor wiring-harness connections are pictured in Fig. 67.

AXIS	DEG/STEP	DIRECTION	MOTOR	+/-	MAX STEPS	EXAMPLE
Base	0.12	cw	M1	+	1000	M1000,0,0,0,0,0
		ccw		-	1000	M −1000,0,0,0,0,0
Shoulder	0.12	up	M2	+	600	M0,600,0,0,0,0
		down		-	600	M0, −600,0,0,0,0
Elbow	0.08	up	M3	+	600	M0,0,600,0,0,0
		down		-	600	M0,0, −600,0,0,0
Wrist Roll	0.10	cw	M4	+	7200	M0,0,0,7200,7200,0
			M5	+		
		ccw	M4	-	7200	M0,0,0, −7200, −7200,0
			M5	-		
Wrist Pitch	0.10	down	M4	+	900	M0,0,0,900, −900,0
			M5	-		
		up	M4	-	900	M0,0,0, −900,900,0
			M5	+		
Gripper	0.10	close	M6	+	2000	M0,0,0,0,0,1800
		open		-	2000	M0,0,0,0,0, −1800

Fig. 65: Axes Movement Specifications

Gear trains become stripped and motors can sometimes burn out when operated under conditions not in accordance with the manufacturer's specifications. This is because a dc motor tends to try to overcome any resistance to its natural rotation.

It does this by drawing more current from the power supply. Soon the heat will cause damage, if not in the motor, then in the power supply, motor-controller circuits, or somewhere else on the PCB. Given the ruggedness which is inherent in these stepper motors, something else usually gives out first.

The MARCRAFT Pro-Arm will provide reliable service for many years, if operated within the guidelines given in the Knowledge Transfer Manual.

The design of the gripper is a subject of special interest when considering the Pro-Arm's stepper motors. For this reason, special precautions have been taken in interfacing the gripper hardware with the gripper motor.

Another look at Fig. 65 will reveal that the gripper motor has been assigned a 2000-step range of motion on either side of true zero positioning.

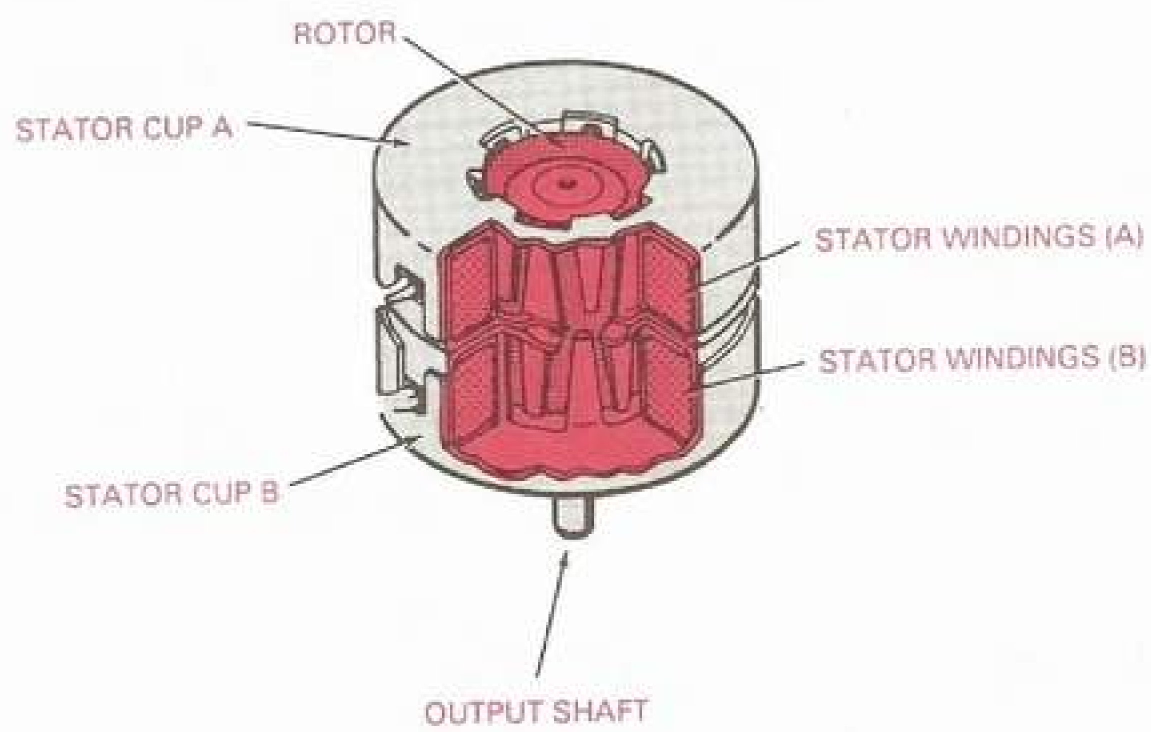


Fig. 66: A 2-Phase Bifilar Stepping Motor

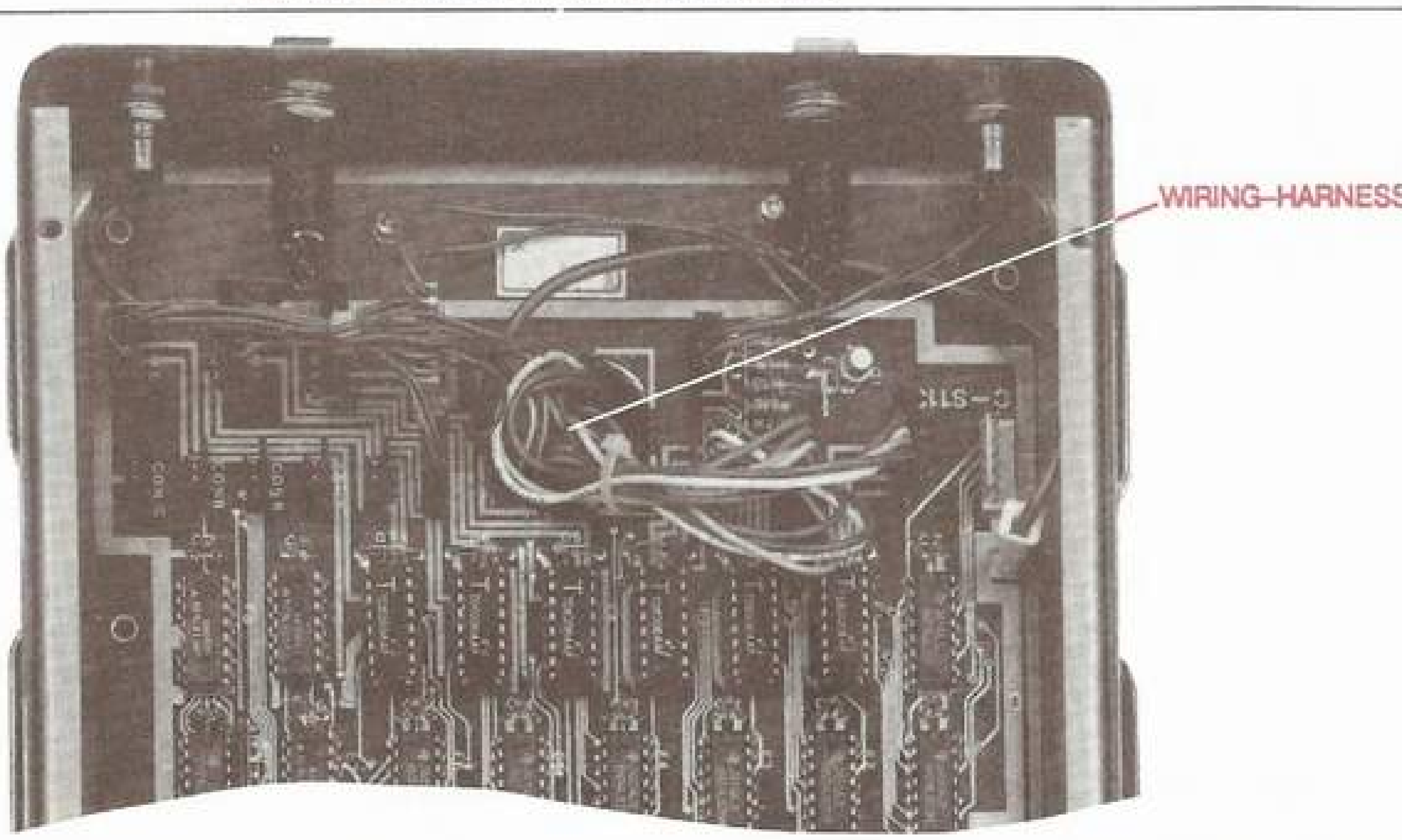


Fig. 67: Motor Wiring-Harness Connections

Since the zero position is defined with the Pro-Arm's gripper in the closed position, something must be done to prevent the gripper from continuing to turn past the closed position for an additional 2000 steps before stopping. Obviously, if the gripper motor continued to pull on the gripper cable for 2000 additional step pulses after the gripper closed, the gripper motor and its driver circuits would soon be damaged.

For this reason, a limit switch has been installed in the Pro-Arm near *M6,* right below the travel path of the gripper cable, as shown in Fig. 68 (shown with the ***body backplate*** removed). Its two wires run through the harness and plug into **Connector 5** on the Pro-Arm's PCB.

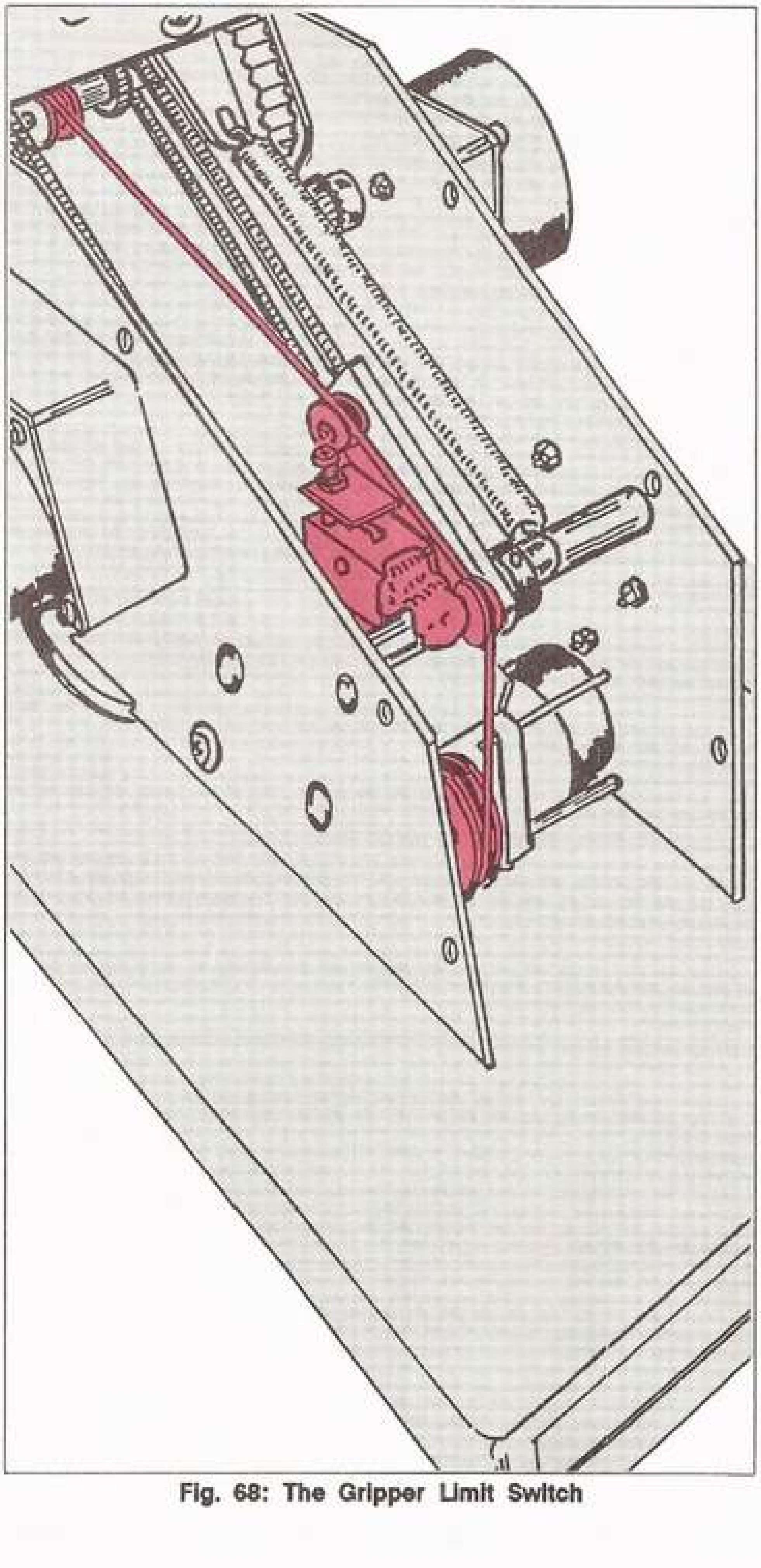


Fig. 68: The Gripper Limit Switch

As soon as the gripper closes, the tension increases enough to pull the limit switch closed, shutting off the gripper motor ***(M6).*** This provides a reliable zero-reference point for the gripper, and prevents damage to the gripper motor and other components. This is why the gripper always closes automatically (if it is open) when the Pro-Arm is powered up.

The gripper limit switch must be set properly, however. If the cable is wound too far before the limit switch is closed, the gripper may not have enough time to open widely enough during normal operations. If the cable is not wound far enough before the limit switch is closed, the gripper may not be able to close completely.

LAB EXPLORATION 23

___23. MOTOR STEPPING

Purpose: To review the major points discussed about the Pro-Arm's stepping motors.

Equipment: Knowledge Transfer Manual, Pencil

___a. If something involves the use of two wires, it is said to be ___________.

___b. __________ power wires are required for each of the Pro-Arm's stepping motors.

___c. __________ control wires are used in the driver circuits of each motor.

___d. A stepper motor repeats a __________ ____________ __________ for each ________ _______ ____________.

___e. The stepper motor assigned to the elbow joint moves ____________ degrees per step.

___f. The stepper motor which is assigned to the gripper can move a maximum of ___________ steps, in either direction, from its true zero position.

___g. A __________ __________ _________ is installed in the Pro-Arm to prevent damage to the __________ _________ and to provide a reliable __________ __________ _________ for the gripper.

___h. This completes Lab Exploration 23, "Motor Stepping." Have your instructor initial your Knowledge Transfer Guide.

PARALLEL INTERFACE

The computer's parallel port sends data, byte by byte, to the Pro-Arm's parallel port by way of the parallel interface cable. Refer to Fig. 69. You installed this cable during Lab Exploration 3, "Hardware." One byte is composed of 8-bits, as shown in Fig. 70. Notice that the byte shown in Fig. 70 is made up of only *"zeros"* or *"ones."* This is because the binary numbering system only uses those two digits. Each *"one"* is worth twice the value of the *"one"* to its immediate right. That means that the value shown in the Fig. 70 represents the number 128 *(DB7 or Data Bit 7)* in normal decimal notation.

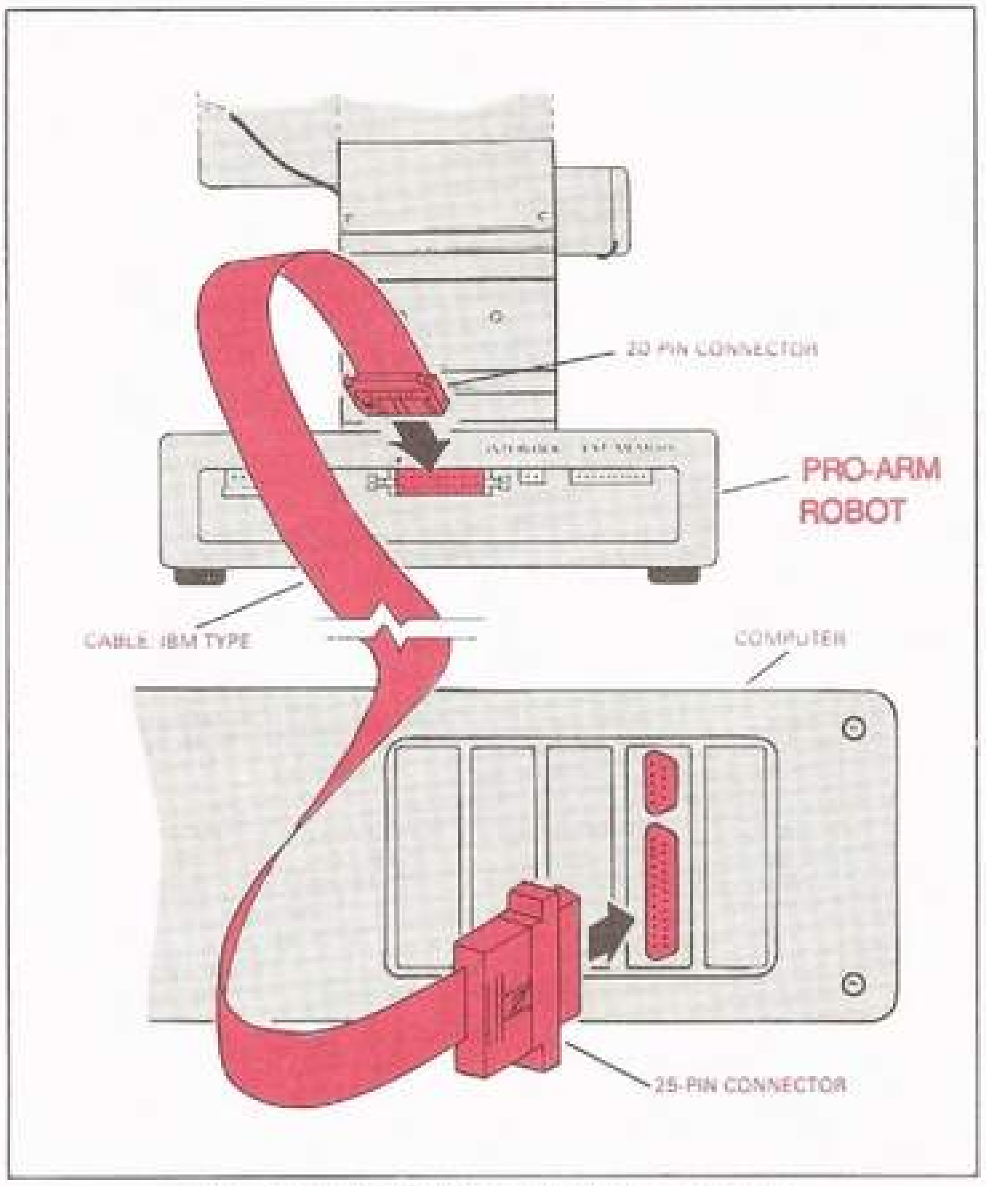


Fig. 69: The IBM-type Interface Cable

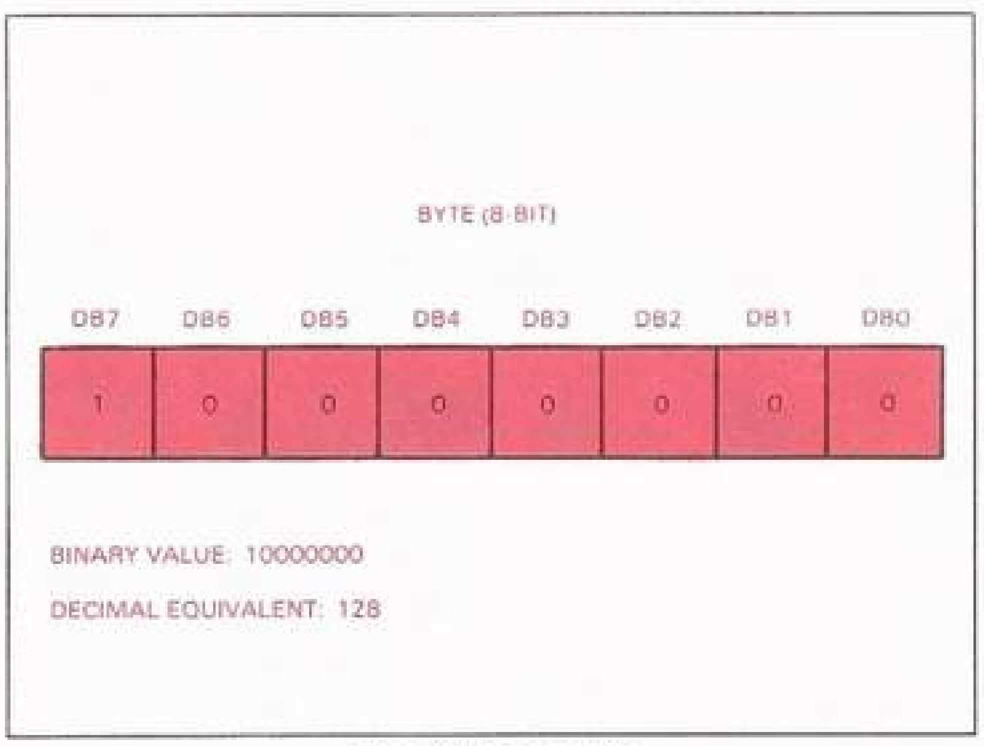


Fig. 70: One Byte

In order to figure out the decimal equivalent of an 8-bit binary value *(byte)*, you must add the individual values of any position that has a *"one"* in it. *"Zeros"* receive no value. The value of the position immediately to the right of DB7 *(position DB6)* is 64. If that position had a one (1) in it also, the value of the byte in Fig. 70 would be 192. If we follow this logic to its inevitable conclusion, the highest numeric value that could be represented with 8 bits of data would be 255. In order to count higher we would need more bits, and a bigger data bus. Larger systems use both. If we needed a bigger number on the Z80's data bus, we could send the number in two or more separate bytes, end to end, with the rightmost digits *(lower byte)* first.

This 8-bit data structure *(or byte)* can also be used to code numbers or letters *(alphanumeric characters)* which can then be interpreted as commands by the Z80 CPU. These are called *ASCII* codes and are shown in the **SPECIFICATIONS** portion of Section VI.

ASCII is an acronym for the *American Standard Code for Information Interchange*. It was created so that the various manufacturers of digital electronic communications equipment could ensure that their products could talk with and listen to each other.

The Pro-Arm and the computer talk with and listen to each other using the IBM-type interface cable supplied with the system. Here are some of the signals being transferred between them. Refer to the **SPECIFICATIONS** section for the pin numbers associated with each signal.

SIGNAL	SOURCE	DESCRIPTION
Data Strobe	Computer	A LOW pulse clocks data from the computer to the Pro-Arm. It signals the Pro-Arm that another byte is ready for transfer.
Data (0–7)	Computer	Input data levels: HIGH = binary 1, LOW = binary 0.
Acknowledge	Pro-Arm	A LOW digital pulse informs the computer that the last character has been received and that the Pro-Arm is ready to accept another character.

SIGNAL	SOURCE	DESCRIPTION
Busy	Pro-Arm	A HIGH digital signal tells the computer that the Pro-Arm is not yet ready to receive the next byte of data (character).

LAB EXPLORATION 24

____24. PRO-ARM AND COMPUTER INTERFACING

Purpose: To review the major points discussed about the Pro-Arm's parallel interface.

Equipment: Knowledge Transfer Manual, Pencil

____a. Data moves out of the computer to the Pro-Arm's interface cable through a/an __________ ______ on the computer.

____b. Data moves into the Pro-Arm from the interface cable through a/an _________ ________ on the Pro-Arm.

____c. Data moves from the computer to the Pro-Arm in a ______ ______ ______ format.

____d. ASCII is an acronym for the _____________ ___________ _______ ______ _____________ ____________.

____e. Numbers and letters are sometimes referred to as ____________________ __________________.

____f. The signal from the Pro-Arm telling the computer that a byte of data has been received, and that it is ready to receive another byte of data, is called a/an __________________.

____g. If the data byte 00101101 was a binary numerical value, it would be equal to _____________ in decimal notation.

____h. If the data byte 01011100 was an ASCII code, it would stand for the symbol ______________.

*Note: See the **SPECIFICATIONS** portion of Section VI.*

____j. This completes Lab Exploration 24, "Pro-Arm and Computer Interfacing." Have your instructor initial your Knowledge Transfer Guide.

POWER SUPPLY

The regulated power supply that comes with the RS-2200 P/A Pro-Arm system provides both ***5-V dc*** and ***12-V dc*** outputs to the Pro-Arm. The digital electronic circuitry receive ***5-V dc at 1.2A***, and the stepping motors receive ***12-V dc at 6A***.

The power-supply input voltage is switch-selected to operate at either ***110—120-V ac*** or ***220—240-V ac, 50/60 Hz*** line voltage. The input circuit is fused. Refer to Lab Exploration 3, "Hardware," and Lab Exploration 5, "Power Up," for further information.

Because of the dangerously high voltages present inside the power-supply case, we recommend that power-supply problems be referred to qualified repair personnel. Regardless of the original warranty (active or expired), return the defective unit to MARCRAFT as outlined under the **TECHNICAL ASSISTANCE and FACTORY SERVICE** heading (Section VI) on page 163.

LAB EXPLORATION 25

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____25. PRO-ARM POWER

Purpose: To review the major points in the discussion about the Pro-Arm's power supply.

Equipment: Knowledge Transfer Manual, Pencil

____a. The voltage rating of the power supply for the Pro-Arm's digital electronic circuitry is _________ at ________.

____b. The voltage rating of the power supply for the Pro-Arm's stepper-motor circuitry is _________ at _________.

____c. The power supply's input voltage is switch-selected to operate at ____________ or ___________ line voltage, at a frequency of _________.

____d. Defective power supplies should be referred to qualified repair personnel due to the presence of ____________ ______ _________ inside the case cover.

____e. This completes Lab Exploration 25, "Pro-Arm Power," and the PARTS AND PERFORMANCE activities. Have your instructor initial your Knowledge Transfer Guide.

OTHER INTERFACING FEATURES

DISCUSSION

Several features exist on the Pro-Arm which have yet to be discussed in any detail. This is because they have to do with interfacing the Pro-Arm with various external devices other than the computer. These features require some programming before they can be used effectively. Because of this, the programming aspects of each feature will not be discussed or demonstrated until the latter parts of this section.

These features will be presented in the following sequence:

1. Interlock
2. 4-Bit Output Port
3. 8-Bit Output Port
4. Input Switches

INTERLOCK

The Pro-Arm is equipped with an *"INTERLOCK"* connector mounted on its PCB *(Connector 12)*, as shown in Fig. 71.

A 2-wire external switch or sensor may be plugged into the interlock connector, allowing the Pro-Arm to check the status of some external device. When the interlock is activated by a programmed command, the Pro-Arm will stop all movement until the external switch of the sensor is in a closed-circuit state. Once the circuit closes, the interlock is deactivated.

Naturally, if the external switch or sensor is already in a closed-circuit state, the program being executed proceeds with no delay.

Activating the interlock when no external device is connected to the Pro-Arm will simply cause the Pro-Arm to stop all movement indefinitely until a closed-circuit condition is detected by the interlock (by plugging a shorting jumper over the two interlock pins), or until the *RESET* button is pushed.

4-BIT OUTPUT PORT

The Pro-Arm contains a 4-bit parallel output port with which it can send commands to an external device, such as an additional stepping motor or a sensor. The external device is plugged onto the 6-pin jack *(Connector 4)* on the PCB, as shown in Fig. 72. The main requirement for any such external device is that its driver circuit must be TTL-compatible. Its input lines must be capable of receiving TTL voltage levels *(0-V dc and 5-V dc)*.

When a proper command is sent to the Pro-Arm. along with a decimal number between 0 *(0000 binary)* and 15 *(1111 binary)*, the binary form of the number is sent along the data bus on lines *D0—D3* to chip *E7* (see the schematic), which is a 74LS04 hex inverter. The 4-bit number is inverted *(complemented)* as it passes through chip *E7*, and then is sent to chip *F0*, which is a 4-bit bistable latch with complementary outputs.

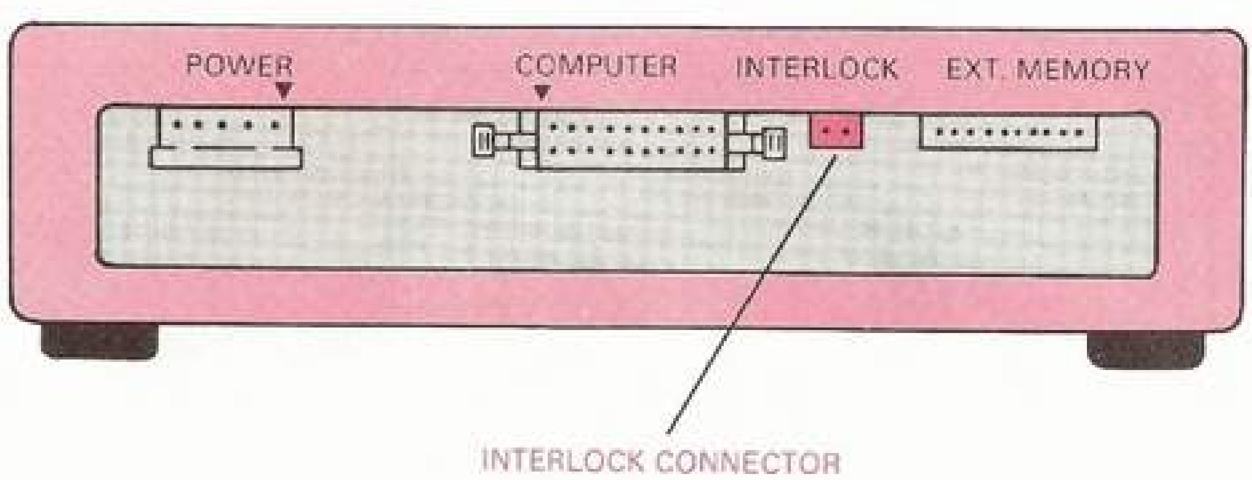


Fig. 71: The Interlock Connector

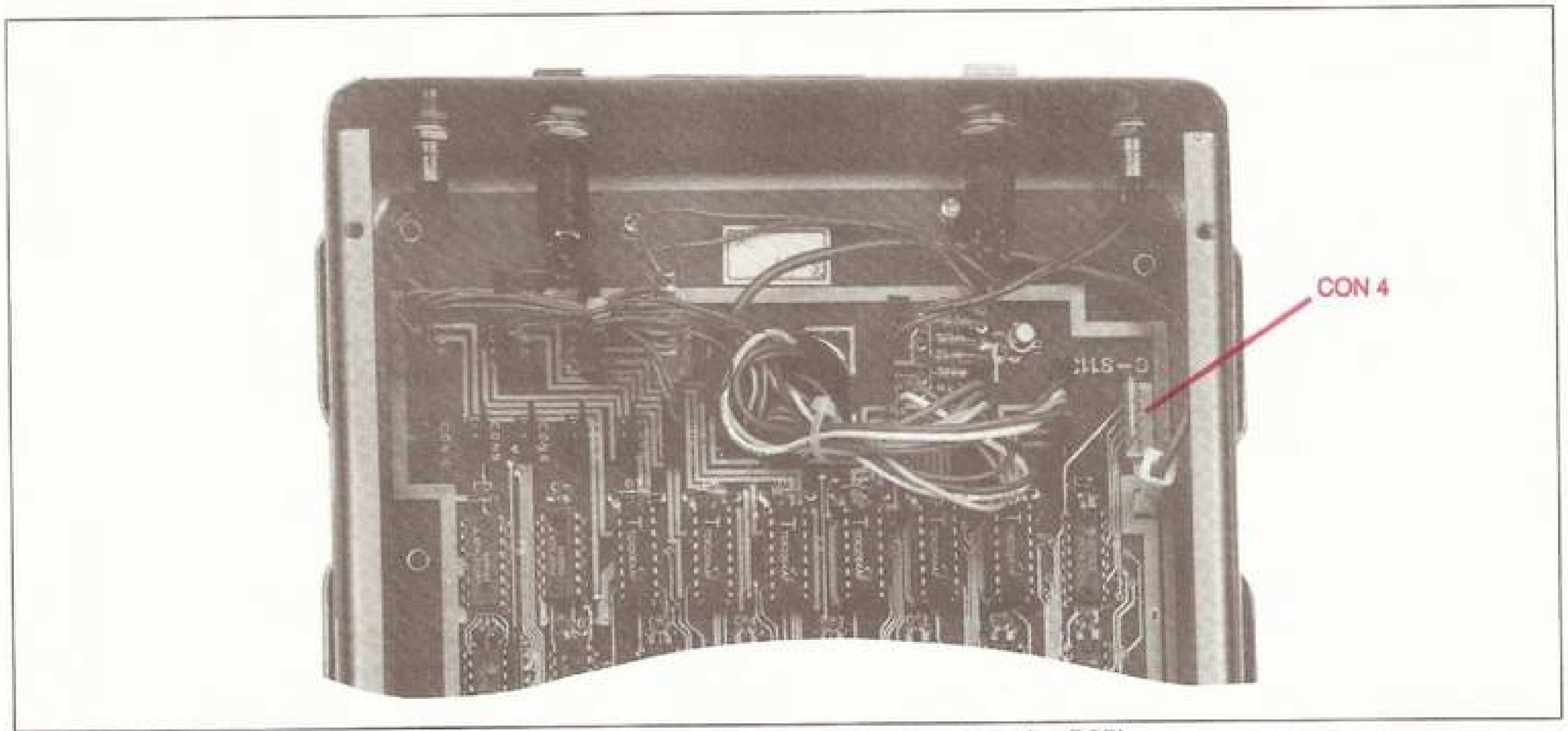


Fig. 72: The 4-bit Output Port (Connector 4 at the PCB)

Notice that the 4-bit number is inverted once again (the complementary outputs are selected) as it passes through the latch. This is because we want the number which was specified in the command to be same number applied to the four pins of ***Connector 4*** on the PCB. Table 3 shows the pin assignment for the 4-bit output port. As explained, the inputs on the last stage (the 74LS75 4-bit bistable latch) have been inverted prior to reaching the output connector so that they will match the values specified in the command.

Let's suppose that we told the Pro-Arm to send the number 2 to ***Connector 4***, in order to tell an external device to perform some specific task. What would happen if the noninverted outputs ***(Pins 16, 15, 10, and 9)*** of chip ***F0*** were sent to ***Connector 4***?

PIN NO.	OUTPUT SIGNAL
1	DB3
2	DB2
3	DB1
4	DB0
5	GND
6	GND

TABLE 3: The 4-Bit Output Port Pin Assignment

Decimal number 2 ***(binary 0010)*** would become decimal number 13 ***(binary 1101)*** as it left the hex inverter chip ***E7***. Sending the noninverted outputs of the latch to ***Connector 4*** would send decimal number 13, not decimal number 2. Therefore, the complementary output pins of chip ***F0 (1, 14, 11, and 8)*** are selected because decimal number 13 is inverted back to the original decimal number 2.

Fig. 73 depicts the wiring between chip ***F0*** and the 6-pin male ***Connector 4***. A wired, 6-pin, female plug connector is supplied with the MARCRAFT RS-2200 P/A Pro-Arm system in order to make the interfacing with this port easier.

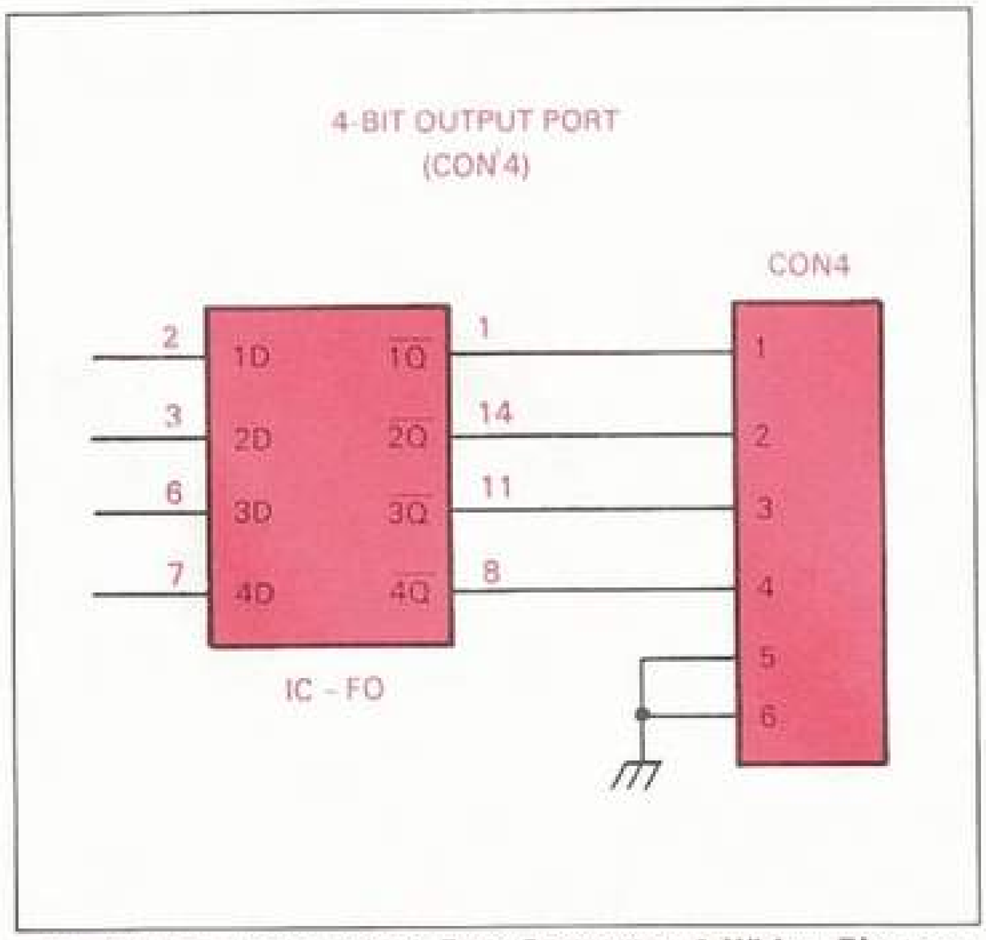


Fig. 73: The 4-Bit Output-Port Connector 4 Wiring Diagram

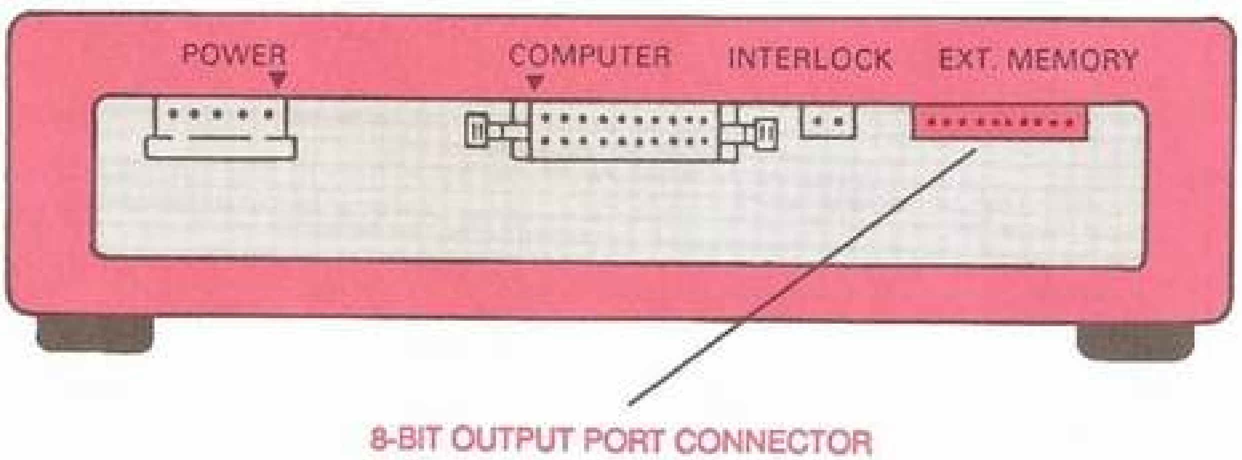
8-BIT OUTPUT PORT

To make the interface, simply wire the external TTL-compatible device to the supplied 6-pin, female plug connector, and then plug it into the matching male connector marked **"CON 4"** on the Pro-Arm's PCB.

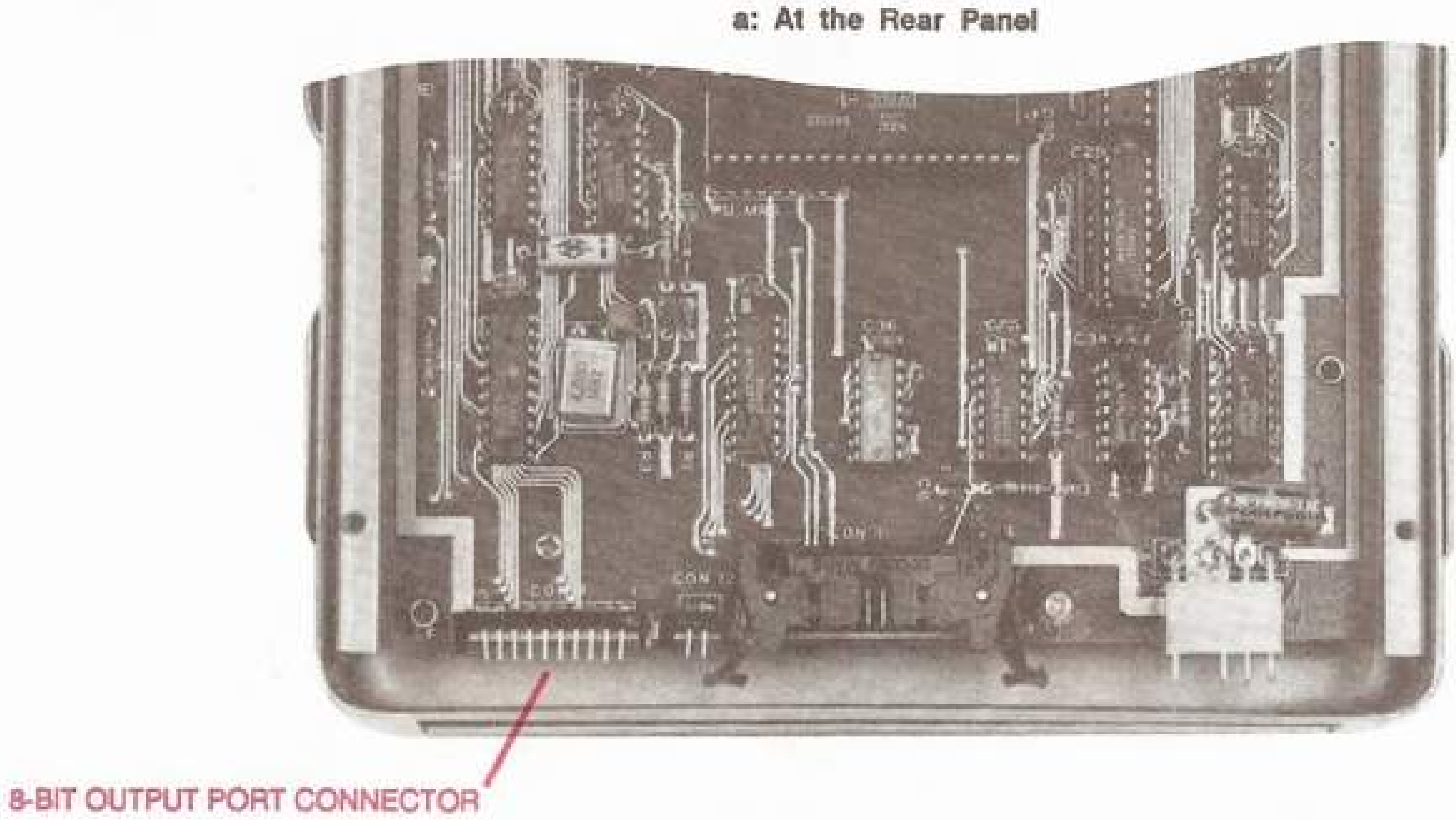
*Caution: Do not connect any live external devices to **Connector 4** when running the self-tests. Externally attached devices may exhibit erratic behavior during any self-test program execution.*

The Pro-Arm also contains an 8-bit parallel output port which has been designed to allow the transfer of data from specific addresses in the Pro-Arm's memory map (see Fig. 64) to an external device.

Fig. 74a shows its location on the Pro-Arm's rear panel, marked **"EXT MEMORY."** On the Pro-Arm's PCB, it is marked **"CON 3,"** as shown in Fig. 74b. A wired, 10-pin, female plug connector is supplied with the MARCRAFT RS-2200 P/A Pro-Arm system also—again, for ease of interfacing.



a: At the Rear Panel



b: At the PCB

Fig. 74: The 8-Bit Output-Port Connector

Unless this data transfer feature is being utilized (by specific program command), all of the output port signals are in a *"LOW"* state *(0-V dc)*. Table 4 shows the pin assignment for this port.

PIN NO.	OUTPUT SIGNAL
1	DB0
2	DB1
3	DB2
4	DB3
5	DB4
6	DB5
7	DB6
8	DB7
9	GND
10	GND

TABLE 4: The 8-Bit Output-Port Pin Assignment

If we check the schematic diagram we find that chip ***A8***, which is a 74LS273 octal flip-flop, is wired so that data is clocked straight through, on command, without any inversion taking place.

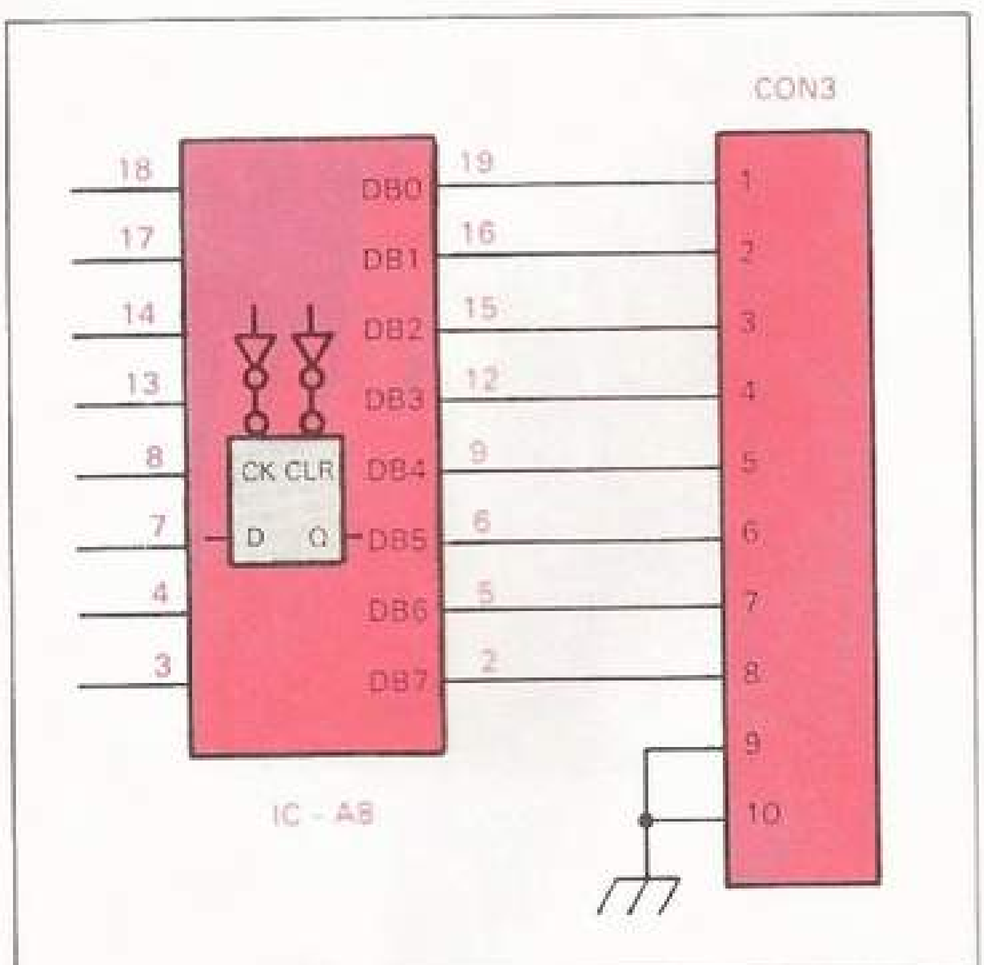


Fig. 75: The 8-Bit Output-Port Connector 3 Wiring Diagram

Fig. 75 depicts the wiring between chip ***A8*** and the 10-pin male ***Connector 3***. To make the interface, simply wire the external TTL- compatible device to the supplied 10-pin female plug connector, and then plug it into the matching male connector marked ***"CON 3"*** on the Pro-Arm's PCB ***(EXT MEMORY*** on the rear panel***)***.

INPUT SWITCHES

The Pro-Arm is equipped with four input-switch connections designed for use with the limit, sensor, or interlock switches. When used in conjunction with the proper programming commands, these switches can be used to remotely control the 4-bit output port previously described. In other words, external devices can be used to control other external devices through the Pro-Arm's circuitry. This allows the Pro-Arm to work in harmony with other devices in the ***"work cell."***

The program command which functions with these input switches in the remote operation of the 4-bit output port is independent from the program command used to control ***Connector 4*** directly.

Each of the switches controls a corresponding signal from the 4-bit output port. Table 5 shows the input-switch connector and the output-port pin designations.

INPUT SWITCH	CONNECTOR 4 OUTPUT PORT BIT/PIN
CON7	DB3/1
CON8	DB2/2
CON9	DB1/3
CON10	DB0/4

TABLE 5: Input-Switch Pin Assignment

The four switches can be programmed to operate either in a ***"normally open"*** ***(HIGH*** TTL level***)*** or a ***"normally closed"*** ***(LOW*** TTL level***)*** configuration. Each switch can be individually programmed to either be enabled or disabled, but until the programming actually occurs, all switches are initially disabled and all output signals are ***LOW***.

NORMALLY OPEN

Using the proper programming command, along with a string of four switch-state parameters ***(0 for disabled or 1 for enabled)***, will allow each enabled switch to send its logic level, whether ***LOW*** or ***HIGH***, to its assigned output-port pin at ***Connector 4***. On the other hand, each disabled switch will only be able to send a ***LOW*** to its assigned output-port pin at ***Connector 4***. See Table 6.

For example, suppose the switch-state parameter string read 1,1,1,0. The outputs at port pins ***DB3, DB2, and DB1*** would be identical to the inputs at switches ***CON7, CON8, and CON9***, respectively. The output at port pin ***DB0*** would remain ***LOW*** no matter what the level of the input at ***CON10*** happened to be.

When a switch-state parameter = "1"

If input switch = ***HIGH***, then its ***Connector 4*** output pin = ***HIGH***.

If input switch = ***LOW***, then its ***Connector 4*** output pin = ***LOW***.

When a switch-state parameter = "0"

If input switch = ***HIGH***, then its ***Connector 4*** output pin = ***LOW***.

If input switch = ***LOW***, then its ***Connector 4*** output pin = ***LOW***.

INPUT SWITCH				COMMAND PARAMETER STRING	OUTPUT PORT			
CON7	CON8	CON9	CON10		DB3	DB2	DB1	DB0
H	H	H	H	U0,0,0,0	L	L	L	L
H	H	H	H	U0,0,0,1	L	L	L	H
H	H	H	H	U0,1,0,0	L	H	L	L
H	H	H	H	U1,1,1,1	H	H	H	H
L	L	L	L	U0,0,0,0	L	L	L	L
L	L	L	L	U1,0,0,0	L	L	L	L
L	L	L	L	U0,1,1,0	L	L	L	L
L	L	L	L	U1,1,1,1	L	L	L	L
L	H	H	H	U0,0,0,0	L	L	L	L
L	H	H	H	U1,0,0,0	L	L	L	L
H	H	L	H	U0,1,0,1	L	H	L	H
L	H	L	H	U1,1,1,1	L	H	L	H

TABLE 6: Truth Table Examples (Normally Open Switches)

Using the proper programming command, along with a string of four switch-state parameters *(0 for disabled or 1 for enabled)*, will allow each enabled switch to send the complement of its logic level, whether *LOW* or *HIGH*, to its assigned output-port pin at *Connector 4*. Each disabled switch will still only be able to send a *LOW* to its assigned output-port pin at *Connector 4*. See Table 7.

When a switch-state parameter = "1"

If input switch = ***HIGH***, then its ***Connector 4*** output pin = ***LOW***.

If input switch = ***LOW***, then its ***Connector 4*** output pin = ***HIGH***.

When a switch-state parameter = "0":

If input switch = ***HIGH***, then its ***Connector 4*** output pin = ***LOW***.

If input switch = ***LOW***, then its ***Connector 4*** output pin = ***LOW***.

For example, suppose the switch-state parameter string read 0,1,1,0. The outputs at port pins ***DB2 and DB1*** would be opposite to the inputs at switches ***CON8 and CON9***, respectively. The outputs at port pins ***DB3 and DB0*** would remain ***LOW*** no matter what the level of the inputs at ***CON7 and CON10*** happened to be.

To connect an external switch or sensor, simply wire it to a 2-pin female plug connector, and then plug the 2-pin connector onto one of the 2-pin male connectors marked ***CON7, CON8, CON9 or CON10*** located on the Pro Arm's PCB.

INPUT SWITCH				COMMAND PARAMETER STRING	OUTPUT PORT			
CON7	CON8	CON9	CON10		DB3	DB2	DB1	DB0
H	H	H	H	V0,0,0,0	L	L	L	L
H	H	H	H	V0,0,0,1	L	L	L	L
H	H	H	H	V0,1,0,0	L	L	L	L
H	H	H	H	V1,1,1,1	L	L	L	L
L	L	L	L	V0,0,0,0	L	L	L	L
L	L	L	L	V1,0,0,0	H	L	L	L
L	L	L	L	V0,1,1,0	L	H	H	L
L	L	L	L	V1,1,1,1	H	H	H	H
L	H	H	H	V0,0,0,0	L	L	L	L
L	H	H	H	V1,0,0,0	H	L	L	L
H	H	L	H	V0,1,0,1	L	L	L	L
L	H	L	H	V1,1,1,1	H	L	H	L

TABLE 7: Truth Table Examples (Normally Closed Switches)

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____26. PRO-ARM INTERFACING

Purpose: To review the major points in the discussion about the Pro-Arm's external interfacing capabilities.

Equipment: Knowledge Transfer Manual, Pencil

____a. The Pro-Arm will stop all movement indefinitely if its ________________ is activated when no external device is wired to Connector ____________.

____b. If an external device's switch or sensor is in a closed state when the Pro-Arm activates its ____________________, then the program being executed ___________ with no ___________.

____c. To use the Pro-Arm's 4-bit output port, an external device must plug onto Connector ____________, which is a __________ _______ ___________, mounted on the _______________.

____d. If a binary number has been inverted, we say it has been _____________.

____e. In digital electronics, a "**LOW**" signal state indicates a voltage of _________, while a "**HIGH**" signal state indicates a voltage of __________.

____f. A disabled input switch can only send a logic _________ to its corresponding output pin on Connector __________.

____g. The decimal value for the 4-bit binary number 1010 is ______________.

____h. An input sensor attached to **CON 9** would control output pin _________ at Connector _____________.

____i. An enabled input switch, whose signal state is inverted at its corresponding output-port pin, is being operated in the _______________ _______________ configuration.

____j. When the Pro-Arm works in harmony with other devices, it is part of a _______ ________.

____k. This completes Lab Exploration 26, "Pro-Arm Interfacing," and the OTHER INTERFACING FEATURES activities. Have your instructor initial your Knowledge Transfer Guide.

BASIC: A SHORT HISTORY

DISCUSSION

Back in the mid-1960s, computer science was not the most sought-after major in higher education. Classes were tough, and many students lost interest in what they were learning long before any real progress took place in their minds. The microcomputer had not yet been heard of, and programmers were learning their skills on big, mainframe computers. In those days, everyone sat at a work station which was connected to the mainframe. The mainframe serviced each work station on a time-share basis. If a student wasn't a computer major, there was little chance of he or she ever learning how to do any programming.

It was at this time that ***John Kemeny*** and ***Thomas Kurtz*** developed the ***BASIC*** programming language, at Dartmouth College in New Hampshire. The idea was to create a programming language that was easy to use, even by those students who were not majoring in computer science. Although it began very humbly, ***BASIC*** has since grown in power. Today, it has been improved upon to the point where it can handle almost any programming task.

*Note: The **BASIC** instruction contained here concerns itself only with the commands needed to run the Pro-Arm, and is not intended to be a complete discourse on the subject of **BASIC** programming. For a detailed examination of **BASIC** and its commands, please consult an appropriate textbook or other source of instruction.*

A BASIC DEMO

DISCUSSION

Supplied on the Pro-Arm Program Disk is a demonstration program that allows you to test and demonstrate the Pro-Arm and the computer as a working system. This demo is written in ***BASIC*** and is called ***DEMO.BAS***. To demonstrate how ***DEMO.BAS*** works with the Pro-Arm, you will first have to load your own version of ***BASIC*** into the computer.

LAB EXPLORATION 27

____27. RUN THE DEMO

Purpose: This exploration will give the student practice in using the version of ***BASIC*** that was included with the computer he or she is using in the classroom. After loading ***BASIC***, the student will load ***DEMO.BAS*** from the MARCRAFT Program Disk and run it on the RS-2200 P/A Pro-Arm system.

Equipment: IBM-compatible Computer System, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Pro-Arm Grid Sheet, IBM-type Parallel Interface Cable, Knowledge Transfer Manual, and the MARCRAFT Program Disk.

____a. Check the system installation and interfacing to be sure that it is ready to go.

____b. Boot the computer system and apply power to the Pro-Arm.

____c. At the prompt (**A>_**) remove the DOS disk from floppy drive **A** and return it to its protective sleeve. Insert the disk containing the **BASIC** software that was included with the computer in your classroom. *Note: If you are using an IBM system, leave the system disk in the floppy drive for the next step.*

____d. Boot your **BASIC** software up by typing **GWBASIC** and pressing **ENTER**. *Note: If you are using an IBM system, type either **BASIC** or **BASICA**, and press **ENTER**.*

____e. The introductory screen appears, along with the IBM-style **BASIC** prompt, which is a/an _________.

____f. Remove the **BASIC** system disk from floppy drive **A** and return it to its protective sleeve. Insert the MARCRAFT Program Disk into floppy drive **A** and type **FILES**. Press **ENTER**.

____g. The display reveals that the MARCRAFT Program Disk contains four **BASIC** format files. They Are ____________,

____________, ____________,

and ____________.

____h. The _______ file extension shows that

they are all meant to be run from within the **BASIC** interpreter.

____i. At the **BASIC** command line, type LOAD "**DEMO** and press **ENTER**. *Note: An alternate form of this command would be to press* ***F3****, type* ***DEMO****, and press* ***ENTER****.*

____j. The floppy drive begins to spin and the **DEMO.BAS** program is loaded into memory. To check if it has been properly loaded, type **LIST** and press **ENTER**. What happened?

____

____

____k. At this point, zero-position the Pro-Arm using the **TEST** and **RESET** buttons on its front panel. Then, type **RUN** and press **ENTER**. *Note: An alternate method of entering the* ***RUN*** *command would be to press* ***F2****.*

____l. Press the **SPACE BAR** to pass the opening screen. The program now asks you to confirm four things. They are:

1. ______________________________

2. ______________________________

3. ______________________________

4. ______________________________

____m. After making the requested confirmations, press the **SPACE BAR** and follow the next two instructions. The Pro-Arm should be positioned on the _______

_________. _____ of the round blocks

should be stacked at grid location _____.

____n. Press the **SPACE BAR** again. What decision must now be made?

____

____o. Answer the question asked and press **ENTER**.

Note: A message now appears explaining that the ***DEMO.BAS*** *program may require some slight adjustment to allow for variations in the operating characteristics among different Pro-Arm robots. However, the only way to tell if* ***DEMO.BAS*** *needs to be adjusted is to begin running it on your Pro-Arm.*

Caution: If, when the program begins to execute, the Pro-Arm obviously goes too far down so that it begins digging into the grid sheet, press its ***RESET*** *button* ***IMMEDIATELY*** *to stop any further Pro-Arm movement. Then, refer to the Lab Exploration 28, "Demo Adjust", before making any further attempt at running* ***DEMO.BAS****.*

____p. Press the **SPACE BAR**. The Pro-Arm ____________ itself to its ______

____________.

____q. As the Pro-Arm moves the test blocks around, the ________________ are

displayed on the monitor.

____r. What message is displayed when the demo is run more than once?

____

____s. When the demo has been completely executed, type **SYSTEM** to return to the DOS prompt.

Note: ***DEMO.BAS*** *gives the Pro-Arm a good test of all its individual components. If the experimental blocks are moved properly, this will show that the Pro-Arm is working well. If any noticeable problems are observed however, refer to* ***TROUBLESHOOTING****, in Section VI.*

____t. Turn off the Pro-Arm's power supply and remove the MARCRAFT Program Disk, placing it in its protective sleeve.

____u. Turn off the computer system, but leave the installation and interface connections as they are.

____v. This completes Lab Exploration 27, "Run the Demo." Have your instructor initial your Knowledge Transfer Guide.

Note: If no calibration problems were encountered during the previous exploration, then no changes need to be made to line ***420*** *in the* ***DEMO.BAS*** *program. When it becomes apparent that the Pro-Arm needs to be recalibrated with the demonstration program, perform the following exploration.*

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

___28. DEMO ADJUST

Purpose: This exploration will show the student how to adjust the ***DEMO.BAS*** demonstration program in order to match it to the calibration of the Pro-Arm that is being used in the classroom. All equipment is still set up as it was in the previous Lab Exploration and the ***DEMO.BAS*** program has already been loaded from the MARCRAFT Program Disk. During execution, it has indicated that an adjustment is required.

Equipment: IBM-compatible Computer System, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Pro-Arm Grid Sheet, IBM-type Pro-Arm Parallel Interface Cable, Knowledge Transfer Manual, and the MARCRAFT Program Disk.

___a. During the execution of **DEMO.BAS**, the Pro-Arm has shown the need for having the program adjusted. When the Pro-Arm's **RESET** button was pushed, the gripper closed. The remaining program lines were dumped to the screen.

___b. If you are not running this Lab Exploration from **Step o** of the previous exploration, you need to load **BASIC** and the **DEMO.BAS** program now. Use The **TEST** and **RESET** buttons to place the Pro-Arm in its **HOME** position.

___c. Since you are at the **BASIC** command line and the **DEMO.BAS** program is loaded, press **F1** and type **420**. *Note: You have just instructed the computer to display line **420** in the **DEMO.BAS** program. It should read as follows:*

```
420 PRINT "M0,615,0,0,0,0" : LPRINT "M0,615,0,0,0,0"
```

*Note: This program line is executed during the zero-positioning of the Pro-Arm, prior to when the arm reaches down to pick up the top block. Notice the second number **(615)**. That three-digit number stands for the number of downward steps taken by the shoulder from its maximum upward position limit.*

___d. Recall which symptom was observed. Did the arm lower far enough so that the gripper grabbed the bottom block? Or worse yet, did the gripper dig into the grid sheet? If the answer to either of these questions is yes, go to **Step f**. Did the arm fail to reach the top block with the very first movement? If so, go to **Step e**.

___e. Since the arm is not reaching down far enough to pick up the top block, the number of steps in this parameter must be increased. Here's how to do it!

___(1) Use the cursor-movement arrows to position the cursor somewhere under the number "**615**" (or the existing number) on line **420**. We want to change the number slightly to allow the shoulder motor to go down a few more steps.

Caution: We only want to make a slight change now, and observe how the arm reacts to this new value.

___(2) Position the cursor directly under the digit "**1**" (or existing 2nd digit) and type the digit "**2**" (or one digit higher). Notice that nothing else on the line has moved or changed. The number of steps should now be "**625**" (or ten more than before). This means that the shoulder motor will move ten steps lower than it did the last time we ran **DEMO.BAS**.

___(3) Press **ENTER** to change line **420** in memory. *Note: No changes to a line can be remembered by the computer unless we press **ENTER** after making the change.*

___(4) Run **DEMO.BAS** again by pressing **F2**. If the arm is still not reaching the top experimental block, press the **RESET** button on the Pro-Arm's front panel and wait for the rest of the program lines to be dumped to the screen. If the arm picks up the block properly, allow the program to execute completely and then proceed to **Step g**.

____(5) Press **F1** and type **420** again.

____(6) Repeat the procedures in **Steps e.(1)** through **e.(4)** using the new values. Increase the step count by no more than ten steps each time.

____f. Since the arm is reaching down too far, perhaps even diving into the grid sheet, we don't want to take any more chances running the demonstration program under these conditions, since the Pro-Arm could be damaged by doing so. The number of downward steps which the shoulder is taking must be decreased. Here's how to do it!

____(1) Use the cursor-movement arrows to position the cursor somewhere under the number "**615**" (or the existing number) on line **420**. We want to lower the number enough to ensure that the shoulder motor doesn't have another opportunity to go down so far again.

Caution: We want to make a large change first, and then make smaller downward adjustments later, if necessary. This will prevent the Pro-Arm from making another dive into the work surface.

____(2) For the number "**615**" (or the existing number), position the cursor directly under the digit "**6**" (or third digit) and type the digit "**5**" (or one less than the current number). Then, position the cursor directly under the "**1**" (or second digit) and type the digit "**6**" (or five more than the current number). Notice that nothing else on the line has been moved or changed. The number of steps should now be "**565**" (or fifty steps less than what it was).

Note: This means that the shoulder motor will stop fifty steps higher than it did the last time we ran ***DEMO.BAS****.*

____(3) Press **ENTER** to change line **420** in memory. *Note: No changes to a line can be remembered by the computer unless we press* ***ENTER*** *after making the change.*

____(4) Run **DEMO.BAS** again by pressing **F2**. If you find that the arm is now not reaching down far enough, press the Pro-Arm's **RESET** button and go to **Step e.(5)**. If the arm is still reaching too low or touching the work surface, press the **RESET** button on the Pro-Arm's front panel and wait for the rest of the program lines to be dumped to the screen. If the arm picks up the top block properly, allow the program to finish before proceeding to **Step g**.

____(5) Press **F1** and type **420** again.

____(6) Repeat the procedures in **Steps f.(1)** through **f.(4)** using the new values. If you find that the arm is now not reaching down far enough, press the **RESET** button on the Pro-Arm and go to **Step e.(5)**.

____g. Run the demonstration program several times with the edited line number **420**, and make sure it works properly. Repeat the editing process until you are satisfied. **DO NOT TURN THE SYSTEM OFF!** If you turn the power off now, you will lose your corrections.

____h. When it becomes evident that line **420** is operating correctly, it's time to save the corrected version of **DEMO.BAS** to the disk. Make sure you are at the **BASIC** command line.

____i. Press **F4** and type **DEMO**. The corrected version of **DEMO.BAS** should now be saved to your MARCRAFT Program Disk.

____j. If you performed this exploration because of problems in Lab Exploration 27, "Run The Demo," return to that exploration now, starting at **Step k**.

____k. Turn off the Pro-Arm's power supply and remove the MARCRAFT Program Disk, placing it in its protective sleeve.

____l. Turn off the computer system, but leave the installation and interface connections as they are.

____m. This completes Lab Exploration 28, "Demo Adjust," and the activities to A BASIC DEMO. Have your instructor initial your Knowledge Transfer Guide.

COMMAND-LINE STRUCTURE

DISCUSSION

BASIC is constructed using a command language that is made up chiefly of everyday English words, with a mixture of abbreviations and other familiar characters. The Pro-Arm's microprocessor recognizes these words and abbreviations ***(characters)*** as various *ASCII* command codes (check **SPECIFICATIONS**, Section VI again). The microprocessor receives these codes from the computer, and translates them into control instructions for operating the Pro-Arm. The computer sends the command codes to the Pro-Arm through its parallel-printer port.

That means that the programs written for the Pro-Arm are identical in format to any other program that instructs the computer to send data. That data takes the form of literal letters and numbers ***(alphanumeric characters)***, and the data flows from the computer to the Pro-Arm through the printer port.

The only difference is that the Pro-Arm is there, not a printer. And the Pro-Arm will only recognize and accept certain characters and strings ***(groups of characters)***, which must be in a specific order. Data that does not meet these requirements is ignored.

A Pro-Arm command consists of an uppercase letter that is usually (but not always) followed by a parameter list, consisting of numbers. Some versions of *BASIC* are case-sensitive, and, therefore, it has become standard practice, when writing *BASIC* programs, to use all capital letters on the command line.

Since the Pro-Arm is connected directly to the computer's parallel-printer port, every *BASIC* command line that is written for it will contain the *LPRINT* statement. This is one of the most powerful statements in *BASIC*. For Pro-Arm purposes, it will be used to send data directly out of the printer port. We suggest you also use the closely related *PRINT* statement to display the command on the monitor as it is being executed by the Pro-Arm.

Here is an example of a *BASIC* language command line for use with the Pro-Arm system. Let's look at it closely.

```
210 PRINT "M0,60,0,0,0,0" : LPRINT "M0,60,0,0,0,0"
```

Notice that the line begins at the extreme left with a number, known as the ***"line or statement number."*** This number helps to identify a specific line of code in the program, especially for editing purposes. If a *BASIC* program was loaded into memory and a change was necessary on this line, it could be called up on the screen by simply typing the *LIST* command, followed by the line number ***(210)***, and then pressing *ENTER*.

Next comes the *PRINT* statement, a quotation mark, followed by a command for the Pro-Arm (the *"M"* or *M(ove)* command), followed by a parameter list, and, finally, another quotation mark. Even though there is more data on this program line, let's just talk about what's happening before the colon, for now.

Here, the *PRINT* statement is a *BASIC* language command directed at the computer, telling it to display everything that is within the quotation marks on the system monitor. That's all. This part of the command line is sending the data within the quotes to the monitor only, not to the Pro-Arm.

Notice the colon. It is used to separate commands, when more than one is contained on a single command line, as in this case. *Note: When a line of code takes more that one line of text in this manual, we will break the two commands into separate program lines. The line containing the **LPRINT** statement will follow the line containing the **PRINT** statement. When writing a **BASIC** program for the Pro-Arm robot, always include the **PRINT** statement and the code within the quotation marks to ensure that the command will be displayed on the computer monitor as it is executed.*

Next comes the *LPRINT* statement, followed by the identical opening quotation mark, the *"M"* command, the parameter list, and, finally, the closing quotation mark.

Everything after the colon is a separate command from everything before the colon. Although both commands look almost identical, they are doing completely different things. While the *PRINT* statement sends the data within the quotes to the screen, the *LPRINT* statement sends the data within the quotes to the Pro-Arm, through the parallel printer port. So, altogether, we have a command which is first shown on the screen for the user's benefit, and then sent to the Pro-Arm through the parallel printer port, for execution.

Once the data within the quotes arrives at the Pro-Arm, through the interface cable, the microprocessor ***(Z80)*** inside the Pro-Arm must decide what the data means.

What does the data inside the quotes mean? Look at the data again. This is the only data that reaches the Pro-Arm from program line *210*.

```
M0,60,0,0,0,0
```

Even if the meaning of this data is still unclear, the statement should look familiar to anyone who has completed Section III on PK TEACH. If the *"Robot Command"* window was carefully observed during the PK TEACH sessions, it should be possible to tell which motor is being referenced in the command above.

Every time a *M(ove)* command was sent to the Pro-Arm during the PK TEACH sessions, it took a similar form of the above command. The *"M"* is the *M(ove)* command itself, while the string of numbers which follow it tell the Pro-Arm which motor should move and how many steps it should take. Even the direction of movement is indicated in the parameter string, with a minus sign used to indicate steps below the *"ZERO"* or *"HOME"* position. The absence of a minus sign indicates that the number of steps taken are above the *"ZERO"* position.

The motor positions are listed left to right, beginning with the base motor *(M1)*, and separated from each other by commas. Therefore, the command shown above instructs the Pro-Arm to move its shoulder motor *(M2)* 60 steps above its *"ZERO"* position. Refer to the Axes Parameter List given below.

Parameter	Axis
m1	Base
m2	Shoulder
m3	Elbow
m4	Wrist
m5	Wrist
m6	Gripper

Since there are six motors in the Pro-Arm, there are six number positions *(parameters)* on the command line, with each one separated from the next by a comma.

Let's look at another command line and imagine that it follows right after line *210* in a *BASIC* program for the Pro-Arm.

```
220 PRINT "C" : LPRINT "C"
```

Notice that even though this line would follow line *210* in the program, it's assigned line number is *"220"* instead of *"211."* This is done on purpose, because of the fact that most programs are edited after first being written. It would be much easier to insert additional program commands between lines *210* and *220* than between lines *210* and *211*. In fact, no more command lines can be inserted between *210* and *211*. If more code was required, it would have to be tacked onto line *210*, using colons to separate the additional commands. This would create a sloppy listing, and make the program much more difficult to edit or debug. Instead, when more code is needed at this particular location in the program, new command lines can be assigned any number from *211* to *219*, and can still fit properly. This is because the commands are sent to the Pro-Arm in the numerical sequence dictated by the line numbers.

Look at the rest of program line *220*. The *PRINT* statement again sends the Pro-Arm command to the screen for the user to see. This time it's the *"C" (Close Gripper)* command. After the command is displayed on the screen, the *LPRINT* statement sends the *"C"* command to the Pro-Arm, which, then, closes its gripper. Whenever data is sent somewhere, whether it be to the screen or to the Pro-Arm, it must always be enclosed in *quotation marks* when using *BASIC*.

PRO-ARM COMMANDS

DISCUSSION

The Pro-Arm's commands are explained in the following listing, with several *IBM DOS BASIC* language statements provided as examples. These examples output the command itself to the monitor prior to execution. The explorations of commands which are used to program external devices are covered in the **PROGRAMMING INTERFACE FEATURES** portion of this section.

Keep in mind that if the Pro-Arm receives an improper or unknown command, the ***ERROR*** lamp will turn on.

Note: Throughout this manual, when reference is made to the "axes," unless otherwise specified, this includes the gripper as well, even though technically the gripper is not considered to be an axis.

C(lose)

Instructs the gripper to close to the maximum limit. The command *"C"* is used alone.

Input parameter: C

Input example: **150 PRINT "C" : LPRINT "C"**

Example results: Gripper closes to maximum position.

D(elay)

Provides a timed stop or time delay in the running program. The command *"D"* is followed by the selected delay (in seconds) to a maximum of 99 seconds (+/–1%).

Input parameter: Dn where n = the selected delay.

Input example: **170 PRINT "D15" : LPRINT "D15"**

Example results: After executing the previous command, the program stops for 15 seconds before continuing with the next instruction or command.

G(o-To)

Recalls the axes movement parameters that were stored in memory with *P(osition)* and *H(ere)* commands. The command *"G"* is followed by the previously designated memory location number.

Input parameter: Gn where n = the designated memory location.

Input example: **190 PRINT "G2" : LPRINT "G2"**

Example results: Axes parameters stored in memory location 2 are recalled and the axes move according to the stored instructions.

H(ere)

Stores the current position of the axes in memory. A maximum of 100 positions (270 with expanded RAM memory) can be stored. The command *"H"* is followed by a designated memory-location number.

Input parameter: Hn where n = the designated memory location.

Input example: **215 PRINT "H3" : LPRINT "H3"**

Example results: Current axes positions are stored in memory location 3.

L(imit)

Limits movement of axes to their maximum travel positions. When command *"L1"* is given, movement of the axes will be restricted to the maximum steps (from zero position) shown in **SPECIFICATIONS**, Section VI, regardless of the input parameters. If an axis limit is exceeded, the Pro-Arm will stop, the ***ERROR*** lamp will turn on, and, thereafter, only the *N(est)* command will be recognized. Command *"L0"* will release the travel limitation.

Input parameter: L0 or L1

Input example: **230 PRINT "L1" : LPRINT "L1"**

Example results: Movement of axes will be restricted to the maximum steps shown in **SPECIFICATIONS**, Section VI .

M(ove)

Instructs each axis to move a specified amount. The command *"M"* is followed by a parameter list comprised of the number of steps each motor must move. (Refer to the Axes Parameter List, given on Page 98, for axes parameter designations.) Be sure to designate a value for each motor, even if it is *"0."*

Input parameter: Mm1,m2,m3,m4,m5,m6

Input example: **245 PRINT"M200,0,150,0,0,0"**
255 LPRINT"M200,0,150,0,0,0"

Example results: Base will move clockwise 200 steps; shoulder moves 0 steps; elbow moves upwards 150 steps; wrist-roll, wrist-pitch, and gripper move 0 steps.

N(est)

Returns axes to the *"HOME"* or zero position, which was designated by the previously used *Z(ero)* command. It also closes the gripper. The command *"N"* is used alone.

Input parameter: N

Input example: **260 PRINT "N" : LPRINT "N"**

Example results: Axes return to zero position.

O(pen)

Instructs the gripper to open to the maximum limit. The command *"O"* is used alone.

Input parameter: O

Input example: **265 PRINT "O" : LPRINT "O"**

Example results: Gripper opens to maximum.

P(osition)

Stores axes-position instructions in memory. The parameters are similar to those of the *M(ove)* command, except that with this command, the axes instructions are sent to memory rather than to the motors. Also, the parameters are based on the movement from zero position, rather than movement from the current position. A maximum of 100 positions (270 with expanded RAM memory) can be stored. Command *"P"* is followed by a user-designated memory-location number of 0 through 99, and a parameter list.

Input parameter: Pn,m1,m2,m3,m4,m5,m6

Input example: **280 PRINT "P8,-800,500,150,0,0,0"**
290 LPRINT "P8,-800,500,150,0,0,0"

Example results: Axes parameters will be stored in memory location 8. When this memory location is later recalled, the base will move counterclockwise 800 steps; shoulder moves upward 500 steps; elbow moves upward 150 steps; wrist-roll, wrist-pitch, and gripper move 0 steps.

Q(uad-Out)

Sends a 4-bit logic signal to the 4-bit output port, Connector 4, and is used to control external devices, such as an additional stepping motor or a sensor. (Refer to the **OTHER INTERFACING FEATURES** section discussed earlier.) The driver circuit for the external device must be TTL compatible. The command is Qn and consists of the character *"Q"* followed by a decimal number of 0 through 15, which represents the equivalent 4-bit binary number.

Input parameter: Qn

Input example: **325 PRINT "Q2" : LPRINT "Q2"**

Example results: Decimal 2 is equivalent to binary number 0010, which makes output *DB1* (pin 3) signal *HIGH* and ***DB0, DB2, and DB3 LOW***. *Note: When n = 0 (**"Q0"**), all output pins are **LOW**.*

R(ead)

Initiates the transfer of 8-bit data bytes from specific addresses in the Pro-Arm's memory map to an external device, such as a printer. (Refer to the **OTHER INTERFACING FEATURES** section discussed earlier.) The command is Rh and consists of the character *"R"* followed by the memory address in hexadecimal.

Input parameter: Rh

Input example: **385 PRINT "R87FF" : LPRINT "R87FF"**

Example results: Contents of memory location 87FF is transmitted to the 8-bit output port.

*Note: When this command is not used, all output port signals are in a **LOW** state.*

S(peed)

Designates speed of all axes motors, with a choice of 5 speeds. (See **SPECIFICATIONS** in Section VI for motor speed detail.) All motors run at the same speed. If the *S(peed)* command is not used, the motors default to speed 3. The command *"S"* is followed by the selected speed number (1–5).

Input parameter: Sn where n = the selected speed number.

Input example: **295 PRINT "S5" : LPRINT "S5"**

Example results: Axes move at speed 5 (the fastest speed) until a new speed command is given.

U(nclose)

Designates the ***"normally open"*** configuration for the four external input switches that are used to remotely control the Pro-Arm's 4-bit output port, independent of the *"Q"* command. The command is Ud3,d2,d1,d0 and consists of the character *"U"* followed by a parameter list representing the four switch states, with either a *"0"* for disabled or a *"1"* for enabled. (Refer to the **OTHER INTERFACING FEATURES** section discussed earlier.) If a switch is enabled (parameter *"1"*), then the output will be the same level as the switch level. For example, if a switch parameter is *"1"* and its level is ***HIGH (open)***, then the corresponding output port level will also be ***HIGH***. Likewise, if the switch level is ***LOW (closed)***, then the output level will also be ***LOW***. If a switch is disabled (parameter *"0"*), the corresponding output level will always be ***LOW***, regardless of the switch level. Refer to the following parameters:

Parameter = "1":
If input = HIGH, then output = HIGH
If input = LOW, then output = LOW
Parameter = "0":
If input = HIGH, then output = LOW
If input = LOW, the output = LOW

Input parameter: Ud3,d2,d1,d0

Input example: **410 PRINT "U1,1,1,0"**
420 LPRINT "U1,1,1,0"

Example results: Output ***DB3, DB2, and DB1*** are the same level as the switches connected to ***CON7*** through ***CON9***. ***DB0*** remains ***LOW***, regardless of the level of the switch connected to ***CON 10***. *Note: Until programmed, all switches are initially disabled and all output signals are **LOW**.*

V(ice-Versa)

Designates the ***"normally closed"*** configuration for the four external input switches that are used to remotely control the Pro-Arm's 4-bit output port, independent of the ***"Q"*** command. The command is Vd3,d2,d1,d0 and consists of the character ***"V"*** followed by a parameter list representing the four switch states, with either a ***"0"*** for disabled or a ***"1"*** for enabled. (Refer to the **OTHER INTERFACING FEATURES** section discussed earlier.) If a switch is enabled (parameter ***"1"***), then the output will be the opposite level as the switch level. For example, if a switch parameter is ***"1"*** and its level is ***LOW (closed)***, then the corresponding output port level will be ***HIGH***. Likewise, if the switch level is ***HIGH (open)***, then the output level will be ***LOW***. If a switch is disabled (parameter ***"0"***), the corresponding output level will always be ***LOW***, regardless of the switch level. Refer to the following:

Parameter = "1":
- ***If input = HIGH, then output = LOW***
- ***If input = LOW, then output = HIGH***

Parameter = "0":
- ***If input = HIGH, then output = LOW***
- ***If input = LOW, the output = LOW***

Input parameter: Vd3,d2,d1,d0

Input example: **420 PRINT "U1,0,1,1"**
430 LPRINT "U1,0,1,1"

Example results: Output ***DB3, DB1, and DB0*** are the opposite level of the switches connected to ***CON7, CON9, and CON 10. DB2*** remains ***LOW***, regardless of the level of the switch connected to ***CON 8***. *Note: Until programmed, all switches are initially disabled and all output signals are **LOW**.*

W(ait)

Checks the status of an optional external interlock switch. If the switch is open (or if no switch is connected), the Pro-Arm stops all movement until the switch is closed. (Refer to the **OTHER INTERFACING FEATURES** section discussed earlier.) The command ***"W"*** is used alone. Once the switch is placed in the closed position, this command is deactivated. *Note: If this command is used without a switch connected, depress the **RESET** button to reactivate the Pro-Arm.*

Input parameter: W

Input example: **360 PRINT "W": LPRINT "W"**

Example results: Activates the interlock circuit and checks the status of the Pro-Arm's interlock switch or sensor.

Z(ero)

Sets the home or zero position of the axes. Each program should begin with this command, otherwise the zero position will be the axes positions when power is first applied, and your program will probably run incorrectly. The command ***"Z"*** is used alone.

Input parameter: Z

Input example: **110 PRINT "Z" : LPRINT "Z"**

Example results: The current positions of the axes are set as home or zero position.

The following explorations are designed to familiarize the student with the commands listed above for the purpose of writing ***BASIC*** user programs for the Pro-Arm.

MOVE COMMAND

As already mentioned, the Pro-Arm uses six stepping motors, five to move the axes and one to operate the gripper. Therefore, each movement command must include instructions for each motor. These instructions tell each motor how many steps to turn and in which direction (see the **COMMAND LINE STRUCTURE** heading discussed earlier).

Fig. 76 shows the maximum distance, in degrees and steps, that each axis may move from its zero position. You will find this information very useful when writing your own programs to control the Pro-Arm. *Note: For wrist-roll and wrist-pitch, motors **M4** and **M5** must both be programmed so that they move simultaneously.*

To review the format of the *M(ove)* command when programming in ***IBM BASIC***, examine the following lines.

```
statement#  PRINT "Mm1,m2,m3,m4,m5,m6"
statement#  LPRINT "Mm1,m2,m3,m4,m5,m6"
```

Because of line length restrictions, we have divided the code into two program statements, the first of which contains the command ***PRINT*** (to output to the screen), and its parameter list, consisting of the direction and number of steps each motor is to turn. The next command, ***LPRINT*** (to output to the Pro-Arm), is followed by the identical parameter list. Notice that no colon is required, since each of the two commands has its own line number.

Notice that the *M(ove)* command must be enclosed within the quotes and each motor movement must be separated by a comma.

AXIS	DEG/STEP	DIRECTION	MOTOR	+/-	MAX STEPS	EXAMPLE
Base	0.12	cw	M1	+	1000	M1000,0,0,0,0,0
		ccw		-	1000	M−1000,0,0,0,0,0
Shoulder	0.12	up	M2	+	600	M0,600,0,0,0,0
		down		-	600	M0,−600,0,0,0,0
Elbow	0.10	up	M3	+	500	M0,0,500,0,0,0
		down		-	500	M0,0−500,0,0,0
Wrist Roll	0.10	cw	M4	+	7200	M0,0,0,7200,7200,0
			M5	+		
		ccw	M4	-	7200	M0,0,0,−7200,−7200,0
			M5	-		
Wrist Pitch	0.10	down	M4	+	900	M0,0,0,900,−900,0
			M5	-		
		up	M4	-	900	M0,0,0,−900,900,0
			M5	+		
Gripper	0.10	close	M6	+	2000	M0,0,0,0,0,2000
		open		-	2000	M0,0,0,0,0,−2000

Fig. 76: Axes Movement Specifications

STATEMENT NUMBER	MOTOR NUMBER	AXIS	DIRECTION	STEPS MOVED	DEGREES MOVED
30	1				
40	1				

TABLE 8: First Movement Log

An example as to how this command should be entered or typed when using an IBM PC, PC/XT or PC-compatible computer would be:

```
60 PRINT "M1000,200,-75,0,0,-900"
70 LPRINT "M1000,200,-75,0,0,-900"
```

In this example, the *M* tells the Pro-Arm that this is a *M(ove)* command. The 1000 means that the body is to move 1000 steps clockwise, the 200 means that the shoulder is to move 200 steps upward, the minus 75 means that the elbow is to move 75 steps downward, the next two zeros mean that the wrist is not to move, and the minus 900 means that the gripper is to open 900 steps.

*Caution: If you do not include a movement for each of the stepping motors, the command will not work. A common error is to separate the **M** and the first number with a comma, or to leave empty spaces between the quotes. It is also a common error to forget to place a quote mark at the end of the command.*

Note: These two commands could both be contained on the same program line, provided a colon is inserted between the two commands.

Be sure to carefully study Fig. 76 to determine how motors *M4* and *M5* work together to move the wrist.

LAB EXPLORATION 29

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

The answers and typical test results for this exploration are shown in the **ANSWERS** section on Pages 167–171. (If these pages have been removed, have your instructor approve your test results.)

____29. MOVE

Purpose: This exploration will provide the student with practice using the *M(ove)* command within a ***BASIC*** program setting.

Equipment: IBM-compatible Computer System, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Pro-Arm Grid Sheet, IBM-type Parallel Interface Cable, Knowledge Transfer Manual, and the MARCRAFT Program Disk.

____a. Check the system installation and interfacing to be sure that it is ready to go.

____b. Boot the computer system and apply power to the Pro-Arm.

STATEMENT NUMBER	MOTOR NUMBER	AXIS	DIRECTION	STEPS MOVED	DEGREES MOVED
30	1				
30	2				
30	3				
30	4				
30	5				
30	6				

TABLE 9: Second Movement Log

____c. At the prompt (A>_), remove the **DOS** disk from floppy drive **A**, and return it to its protective sleeve. Insert the disk containing the **BASIC** software (included with the computer in your classroom). *Note: If you are using an IBM system, leave the system disk in the floppy drive for the next step.*

____d. Boot your **BASIC** software up by typing **GWBASIC** and pressing **ENTER**. *Note: If you are using an IBM system, type in either **BASIC** or **BASICA**, and press **ENTER**.*

____e. Use the **TEST** and **RESET** buttons on the Pro-Arm's front panel to place it in its "**ZERO**" position.

____f. At the **BASIC** prompt, type in the following program. (Don't forget to press **ENTER** at the end of each line to place it in memory.)

```
20 PRINT "Z" : LPRINT "Z"
30 PRINT "M500,0,0,0,0,0" : LPRINT "M500,0,0,0,0,0"
40 PRINT "M-500,0,0,0,0,0" : LPRINT "M-500,0,0,0,0,0"
50 PRINT "END" : END
```

*Note: The **LPRINT "Z"** command sets the Pro-Arm's current position as the "**HOME**" position, and is required at the beginning of all programs. Also, although **IBM BASIC** can tell when the end of a program occurs, many systems using **BASIC** cannot. Therefore, in order to make your programs universally compatible in form, it's a good programming practice to use the **END** command as the last line of any **BASIC** program.*

____g. Momentarily press the **RESET** button to be sure the Pro-Arm is ready to receive commands from the computer.

____h. Type **RUN** and press **ENTER** (or press **F2**). Observe the action of the arm and then describe its movement by completing Table 8. Also, notice that the program commands appear on the screen as each one is executed by the Pro-Arm.

____i. Substitute the following lines for Statements 30 and 40:

```
25 PRINT "M500,300,-200,0,0,-1000"
30 LPRINT "M500,300,-200,0,0,-1000"
35 PRINT "M-500,-300,200,0,0,1000"
40 LPRINT "M-500,-300,200,0,0,1000"
```

____j. Type **RUN** and press **ENTER** (or press **F2**). Again, observe the action of the arm and complete Table 9 for the arm movement of Statement 30. Use the information in the Axes Movement Specifications (Fig. 76) to calculate the number of degrees moved.

____k. Now try writing a few program lines to move the arm. Remember not to exceed the allowable number of steps for any axis in either direction from the zero position. First, write your program lines in the space provided at the bottom of the page, and then type them into the computer.

____l. After you have typed the program lines into your computer, momentarily touch the **RESET** button on the Pro-Arm. Then execute the program lines by typing **RUN** and pressing **ENTER**. Did the arm move as desired?

____

If not, then rewrite the program lines and try executing them again.

____m. Turn off the Pro-Arm's power supply, remove the **BASIC** Software Disk, and place it in its protective sleeve.

____n. Turn off the computer system, but leave the installation and interface connections as they are.

____o. This completes Lab Exploration 29, "Move." Please have your instructor initial your Knowledge Transfer Guide.

ZERO AND NEST COMMANDS

We've already discussed how the Pro-Arm can store up to 100 positions in its RAM memory. One of the positions that the Pro-Arm is often required to remember is its *"HOME"* position. This is done using the *Z(ero)* command. In fact, the Pro-Arm may follow commands (including arm positions) directly from a computer, or it may actually follow computer commands using arm positions previously stored in RAM (as in the ***DEMO.BAS*** program).

This exploration is concerned with learning how to use the *Z(ero)* (zero-position) and *N(est)* commands.

LAB EXPLORATION 30

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____30. ZERO AND NEST

Purpose: This exploration will provide the student with practice using the *Z(ero)* and *N(est)* commands within a ***BASIC*** program setting.

Equipment: IBM-compatible Computer System, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Pro-Arm Grid Sheet, IBM-type Parallel Interface Cable, Knowledge Transfer Manual, and the MARCRAFT Program Disk.

____a. Check the system installation and interfacing to be sure that it is ready to go.

____b. Boot the computer system and apply power to the Pro-Arm.

____c. At the prompt (**A>_**) remove the **DOS** disk from floppy drive **A** and return it to its protective sleeve. Insert the disk containing the **BASIC** software (included with the computer in your classroom). *Note: If you are using an IBM system, leave the system disk in the floppy drive for the next step.*

____d. Boot your **BASIC** software up by typing **GWBASIC** and pressing **ENTER**. *Note: If you are using an IBM system, type in either **BASIC** or **BASICA**, and press **ENTER**.*

____e. Use the **TEST** and **RESET** buttons on the Pro-Arm's front panel to place it in its **"ZERO"** position.

____f. At the **BASIC** prompt, type the following program. *Note: Don't forget to press **ENTER** at the end of each line to place the line in memory.*

```
20 PRINT "Z" : LPRINT "Z"
30 PRINT "M500,0,0,0,0,0" : LPRINT "M500,0,0,0,0,0"
40 PRINT "M-500,0,0,0,0,0" : LPRINT "M-500,0,0,0,0,0"
50 PRINT "END" : END
```

____g. Momentarily press the Pro-Arm's **RESET** button to be sure that the commands from the computer will get through.

____h. Type **RUN** and press **ENTER** (or press **F2**).

____i. Before the **RUN** command was executed, the arm was in its ________ position.

____j. Statement 30 caused the arm to move ________ steps in a ____________ direction.

____k. Statement 40 returned the arm to its ____________ position by moving it __________ steps in a ____________ direction.

____l. Rewrite the above program by replacing Statement 40 so that the new program is as follows:

```
20 PRINT "Z" : LPRINT "Z"
30 PRINT "M500,0,0,0,0,0" : LPRINT "M500,0,0,0,0,0"
40 PRINT "N" : LPRINT "N"
50 PRINT "END" : END
```

____m. Before proceeding, be sure the Pro-Arm is once again in its "**HOME**" position.

____n. Type **RUN** and press **ENTER** (or press **F2**).

____o. Did the Pro-Arm repeat the exact same motion as it did when executing the program in **Step h**?

____

It should have!

____p. Statement 20 instructs the Pro-Arm to memorize the current position as the "**HOME**" position. What command gave this instruction?

____

Would it still memorize the current position as its "**HOME**" position, even if initially the axes were not properly aligned at their zero position?

____

____q. Statement 30 instructs the Pro-Arm to move its body by rotating the base axis 500 steps clockwise. Did it?

____

____r. Statement 40 instructs the Pro-Arm to **NEST** (return to "**HOME**" position). Did it?

____

What command gave this instruction?

____

In which statement did it memorize the "**HOME**" position?

____

____s. Now type **NEW** and press **ENTER**. This will remove the above program from the computer's memory so that you may write another program.

____t. Next, clear the screen by typing **CLS** and then pressing **ENTER**.

____u. In the space provided below, write a few program lines which make the arm move to two different positions and then return to its "**HOME**" position using the **N**(est) command. Remember to first identify the "**HOME**" position by using the **Z**(ero) command.

____v. Type your program lines into the computer and momentarily press the **RESET** button on the Pro-Arm.

____w. Now, type **RUN** and press **ENTER** (or press **F2**). Did the Pro-Arm return to its "**HOME**" position when executing the **N**(est) command?

____

If not, then rewrite or correct your program lines so that the arm will **NEST**.

____x. Turn off the Pro-Arm's power supply and remove the **BASIC** Software Disk, placing it in its protective sleeve.

____y. Turn off the computer system, but leave the installation and interface connections as they are.

____z. This completes Lab Exploration 30, "Zero and Nest." Have your instructor initial your Knowledge Transfer Guide.

SPEED COMMAND

As you know, the stepping motors are used to move the axes in the Pro-Arm. The motors may be controlled to move the arm in any one of five different speeds. This is important since you may wish to move the arm to a specific position at a rapid speed and, then, perform an operation which requires a slow rate of movement.

The time it takes the Pro-Arm to move from one location to another is often called the ***"cycle time,"*** and it is affected by two factors: ***(1) payload*** (weight of the part to be moved), and ***(2) extension*** of the manipulator's arm (distance from the shoulder axis to the end-effector). This speed of movement is measured in either inches per second or millimeters per second.

The formula for calculating speed is:

SPEED = DISTANCE / TIME

For example, if an industrial robot took 20 seconds to move a payload 100 inches, then the speed would equal:

SPEED= 100 inches / 20 seconds
= 5 inches / 1 second

The Pro-Arm is designed to move at 5 different speeds as described in the following Table.

SPEED NO.	DESCRIPTION	MOVEMENT SPEED (Approximate)
1	Very slow	2.5 inch/sec (64 mm/sec)
2	Slow	4 inch/sec (102 mm/sec)
3	Medium	5 inch/sec (127 mm/sec)
4	Fast	7 inch/sec (178 mm/sec)
5	Very fast	8 inch/sec (203 mm/sec)

TABLE 10: The Five Pro-Arm Motor Speeds

LAB EXPLORATION 31

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____31. SPEED

Purpose: This exploration will provide the student with practice using the ***S(peed)*** command within a ***BASIC*** program setting.

Equipment: IBM-compatible Computer System, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Pro-Arm Grid Sheet, IBM-type Parallel Interface Cable, Knowledge Transfer Manual, MARCRAFT Program Disk, and a Stop Watch.

____a. Check the system installation and interfacing to be sure that it is ready to go.

____b. Boot the computer system and apply power to the Pro-Arm.

____c. At the prompt **(A>_)** remove the **DOS** disk from floppy drive **A** and return it to its protective sleeve. Insert the disk containing the **BASIC** software (included with the computer in your classroom). *Note: If you are using an IBM system, leave the system disk in the floppy drive for the next step.*

____d. Boot your **BASIC** software up by typing **GWBASIC** and pressing **ENTER.** *Note: If you are using an IBM system, type in either **BASIC** or **BASICA**, and press **ENTER**.*

____e. Use the **TEST** and **RESET** buttons on the Pro-Arm's front panel to place it in its "ZERO" position.

____f. At the **BASIC** prompt, type the following program. *Note: Don't forget to press **ENTER** at the end of each line to place the line in memory.*

```
20 PRINT "Z" : LPRINT "Z"
30 PRINT "S1" : LPRINT "S1"
35 PRINT "M1000,0,0,0,0,0"
40 LPRINT "M1000,0,0,0,0,0"
45 PRINT "M-1000,0,0,0,0,0"
50 LPRINT "M-1000,0,0,0,0,0"
60 PRINT "END" : END
```

Note: Statement 30 tells the Pro-Arm to move at speed number 1. This is the ***slowest speed*** *possible for the Pro-Arm. Statement 40 tells the Pro-Arm to move the base axis 1000 steps in a clockwise direction. Statement 50 tells the Pro-Arm to move the base axis 1000 steps in a counterclockwise direction.*

____g. Momentarily press the Pro-Arm's **RESET** button to be sure that the computer commands are received.

____h. Type **RUN** and press **ENTER** (or press **F2**). Notice how slowly the arm moves as the program commands appear on the screen.

____i. Type **RUN** and press **ENTER** again. This time, use a stop watch, or other device, to measure and record the time required for the arm to complete the 2000-step cycle.

________________ ***seconds at speed number 1***

____j. During the 2000-step cycle, the gripper traveled 70 inches (1780 mm). Using the equation given above, and the time recorded in **Step i**, calculate and record the speed of this cycle. Be sure to include the units (e.g. in./sec., mm/sec., etc.) with the calculated value. Use the space at the bottom of this page to make your calculation.

Speed using S1 command = ________________

____k. Now retype Statement 30 in your computer program as follows:

```
30 PRINT "S5" : LPRINT "S5"
```

Note: Statement 30 now tells the Pro-Arm to move at speed number 5. This is the ***fastest speed*** *possible for the Pro-Arm.*

____l. Type **RUN** and press **ENTER** (or press **F2**). Again measure and record the time required for the arm to complete the 2000-step cycle.

________________ ***seconds at speed number 5***

____m. Again calculate the speed for the 2000-step cycle using the time recorded in **Step l.** Use the space at the bottom of the page to make your calculations.

Speed using S5 command = ________________

____n. Turn off the Pro-Arm's power supply, remove the **BASIC** Software Disk, and place it in its protective sleeve.

____o. Turn off the computer system, but leave the installation and interface connections as they are.

____p. This completes Lab Exploration 31, "Speed." Please have your instructor initial your Knowledge Transfer Guide.

POSITION, HERE, AND GO-TO COMMANDS

The Pro-Arm may be required to move to a specific location that has been previously specified, or may need to return to a particular position that has been memorized. Therefore the Pro-Arm must be able to memorize a position before first moving to it, as well as memorizing a current position; so that it may return upon command. This is done by using the *P(ostion)* command to memorize a specified position, and the *H(ere)* command to memorize a current position. The *G(o-To)* command is used to recall a position that has been memorized using either the *P(osition)* or *H(ere)* command, and then used to instruct the Pro-Arm to move to the position.

The format of the *P(osition)* command is:

```
statement# PRINT "Pn,m1,m2,m3,m4,m5,m6"
statement# LPRINT "Pn,m1,m2,m3,m4,m5,m6"
```

The command begins with a statement number followed by the *PRINT* command (for output to the screen), and then a parameter list. Next comes another statement number, the *LPRINT* command (for output to the Pro-Arm), and, then, an identical parameter list. The parameter list must begin with the capital letter *P*, followed by any integer from 0 through 99. This means that you may identify up to 100 different positions (a number may only be used once). This number is used in conjunction with the *G(o-To)* command to recall a specific position.

A special characteristic of the *P(osition)* command is that it may be placed anywhere within the program prior to recalling it with the *G(o-To)* command. However, a common practice is to place all of the *P(osition)* commands together, near the top of the program. An example of a correctly written *P(osition)* command is:

```
35 PRINT "P3,200,0,100,0,0,0"
40 LPRINT "P3,200,0,100,0,0,0"
```

The format of the *H(ere)* command is:

```
statement#  PRINT "Hn" : LPRINT "Hn"
```

Following the statement number and *PRINT* command is the quoted parameter, which consists of the capital letter *H* and any integer from 0 through 99 that has not previously been used in either a *P(osition)* command or another *H(ere)* command. After a separating colon is the *LPRINT* command, followed by the parameter in quotes. An example of a correctly written *H(ere)* command is:

```
75 PRINT "M0,0,200,50,50,0"
80 LPRINT "M0,0,200,50,50,0"
83 PRINT "M500,100,0,0,0,200"
85 LPRINT "M500,100,0,0,0,200"
90 PRINT "H7" : LPRINT "H7"
```

Statements 80 and 85 will cause the arm to move to a specific position and, then, Statement 90 will memorize it as position number 7. The *H(ere)* command does not memorize the *M(ove)* command on Statement 85, but rather the arm position after executing the statement.

By using the *G(o-To)* command as illustrated in the following statements, the arm will return to the memorized position specified by the Statement 35 example, and, then, to the position memorized following Statement 85 above.

```
165 PRINT "N" : LPRINT "N"
170 PRINT "G3" : LPRINT "G3"
175 PRINT "G7" : LPRINT "G7"
```

Remember that a *H(ere)* command does not memorize a *M(ove)* command, but rather the current arm position.

LAB EXPLORATION 32

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____32. POSITION, HERE, AND GO-TO

Purpose: This exploration will provide the student with practice using the *P(osition), H(ere),* and *G(o-To)* commands within a *BASIC* program setting.

Equipment: IBM-compatible Computer System, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Pro-Arm Grid Sheet, IBM-type Parallel Interface Cable, Knowledge Transfer Manual, and the Marcraft Program Disk.

____a. Check the system installation and interfacing to be sure that it is ready to go.

____b. Boot the computer system and apply power to the Pro-Arm.

____c. At the prompt **(A>_)** remove the **DOS** disk from floppy drive **A** and return it to its protective sleeve. Insert the disk containing the **BASIC** software (included with the computer in your classroom).

Note: If you are using an IBM system, leave the system disk in the floppy drive for the next step.

____d. Boot your **BASIC** software up by typing **GWBASIC** and pressing **ENTER**. *Note: If you are using an IBM system, type either **BASIC** or **BASICA**, and press **ENTER**.*

____e. Use the **TEST** and **RESET** buttons on the Pro-Arm's front panel to place it in its "**ZERO**" position.

____f. At the **BASIC** prompt, type in the following program. *Note: Don't forget to press **ENTER** at the end of each line to place the line in memory.*

```
20 PRINT "Z":LPRINT "Z"
30 PRINT "P1,500,0,0,0,0,0":LPRINT "P1,500,0,0,0,0,0"
40 PRINT "P2,0,500,0,0,0,0":LPRINT "P2,0,500,0,0,0,0"
50 PRINT "G1":LPRINT "G1"
60 PRINT "N":LPRINT "N"
70 PRINT "G2":LPRINT "G2"
80 PRINT "N":LPRINT "N"
90 PRINT "END":END
```

*Note: There must be a comma between the **P1** and the **500** in Statement 30 and between the **P2** and the **0** in Statement 40. Also, note that each **P**(osition) command is identified by a different number following the **P** (e.g., **P1**, **P2**, etc.).*

____g. Momentarily press the Pro-Arm's **RESET** button to make sure that the computer commands are received.

____h. Press **F2**.

____i. Did the arm move directly to the position identified in Statement 30, and then directly to the position indicated in Statement 40?

____

Why?

____

____

____j. What statement caused the arm to move to the position given in Statement 30?

____

____k. What statement caused the arm to move to the position given in Statement 40?

____

____l. Type **NEW** and press **ENTER** to erase the current program in memory, so that you may write another program.

____m. Clear the screen by typing **CLS** and pressing **ENTER**.

____n. Now type the following program, pressing **ENTER** after each line.

```
20 PRINT "Z":LPRINT "Z"
30 PRINT "M500,0,0,0,0,0":LPRINT "M500,0,0,0,0,0"
40 PRINT "H1":LPRINT "H1"
50 PRINT "M0,300,0,0,0,0":LPRINT "M0,300,0,0,0,0"
60 PRINT "H2":LPRINT "H2"
70 PRINT "N":LPRINT "N"
80 PRINT "G1":LPRINT "G1"
90 PRINT "G2":LPRINT "G2"
100 PRINT "N":LPRINT "N"
110 PRINT "END":END
```

____o. Momentarily press the Pro-Arm's **RESET** button to make sure that the computer commands are received.

____p. Press **F2**.

____q. Did Statement 30 cause the Pro-Arm to move?

____

Why?

____

____

____r. Did Statement 40 cause the Pro-Arm to move?

____

Why?

____

____

____s. Which statement gave the Pro-Arm instructions to return to the position it was in after executing Statement 50?

____

____t. From "**HOME**" position, what **M**(ove) command would you use to move the arm to the position specified by "**G2**"?

____

Why won't LPRINT "M0,300,0,0,0,0" work?

____

____

____

____u. Turn off the Pro-Arm's power supply, remove the **BASIC** Software Disk, and place it in its protective sleeve.

____v. Turn off the computer system, but leave the installation and interface connections as they are.

____w. This completes Lab Exploration 32, "Position, Here, and Go-To." Have your instructor initial your Knowledge Transfer Guide.

THE DELAY COMMAND

It may be desirable to have the Pro-Arm temporarily stop its movement at a particular position. This could be done for any number of reasons. The Pro-Arm uses the ***D(elay)*** command to halt the arm movement for a specified length of time.

An example of a properly written delay command is:

```
210 PRINT "D13" : LPRINT "D13"
```

The "***D***" in the above statement indicates that this is a delay command and the "***13***" means that it is to delay the Pro-Arm's movement for 13 seconds. The delay time must be an integer from 1 through 99. *Caution: **DO NOT** use fractions of a second.*

LAB EXPLORATION 33

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____33. DELAY

Purpose: This exploration will provide the student with practice using the ***D(elay)*** command within a ***BASIC*** program setting.

Equipment: IBM-compatible computer system, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Pro-Arm Grid Sheet, IBM-type Parallel Interface Cable, Knowledge Transfer Manual, MARCRAFT Program Disk, and a Stop Watch.

____a. Check the system installation and interfacing to be sure that it is ready to go.

____b. Boot the computer system and apply power to the Pro-Arm.

____c. At the prompt (**A>_**) remove the **DOS** disk from floppy drive **A** and return it to its protective sleeve. Insert the disk containing the **BASIC** software (included with the computer in your classroom). *Note: If you are using an IBM system, leave the system disk in the floppy drive for the next step.*

____d. Boot your **BASIC** software up by typing **GWBASIC** and pressing **ENTER.** *Note: If you are using an IBM system, type in either **BASIC** or **BASICA**, and press **ENTER**.*

____e. Use the **TEST** and **RESET** buttons on the Pro-Arm's front panel to place it in its "**ZERO**" position.

____f. At the **BASIC** prompt, type the following program. *Note: Don't forget to press **ENTER** at the end of each line to place the line in memory.*

```
20 PRINT "Z":LPRINT "Z"
30 PRINT "M500,0,0,0,0,0":LPRINT "M500,0,0,0,0,0"
50 PRINT "N":LPRINT "N"
60 PRINT "END":END
```

____g. Momentarily press the Pro-Arm's **RESET** button.

____h. Press **F2**. For what length of time did the arm wait at the position given in Statement 30 before returning to its "**HOME**" position?

____

____i. Now, add the following statement to the above program.

```
40 PRINT "D5":LPRINT "D5"
```

____j. Momentarily press the Pro-Arm's **RESET** button.

____k. Press **F2.** This time did the Pro-Arm pause for a few seconds at the position given in Statement 30?

____

____l. Press **F2 again.** This time, use the stop watch to measure and record the delay time following the **M**(ove) given in Statement 30.

Delay time = ________ seconds

____m. What is the delay time specified in Statement 40?

____

____n. Did the arm in fact pause for the specified time?

____

____o. Type **NEW**, and then press **ENTER**, to clear **(erase)** the above program from the computer's memory.

____p. Type **CLS**, and then press **ENTER** to clear the screen.

____q. In the space at the bottom of the page, write a few program lines that will move the body 700 steps clockwise, pause for 5 seconds, move the elbow upward 300 steps, pause for 8 seconds, and then return "**HOME**", using the **N**(est) command.

____r. Now type your program lines into the computer.

____s. Press the Pro-Arm's **RESET** button.

____t. Press **F2**. Did your program lines move the arm as specified in **Step q**?

____

If not, correct your program lines and try executing them again.

____u. If the arm followed the correct routine, **RUN** the program again so that, using a stop watch or a watch's second hand, you can measure the delay times. Record the delay times in the space below.

First delay = ________ seconds

Second delay = ________ seconds

____v. Turn off the Pro-Arm's power supply and remove the **BASIC** Software Disk, placing it in its protective sleeve.

____w. Turn off the computer system, but leave the installation and interface connections as they are.

____x. This completes Lab Exploration 33, "Delay." Please have your instructor initial your Knowledge Transfer Guide.

The gripper *(hand)* may be opened as part of a *M(ove)* command using any number of negative steps up to minus 2000. It may also be closed using the *M(ove)* command with the same number of steps that was used to open it, by giving a positive number. However, if it is to be fully opened or fully closed, all that is necessary is to use the *O(pen)* or *C(lose)* command.

LAB EXPLORATION 34

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____34. OPEN OR CLOSE

Purpose: This exploration will provide the student with practice using both the *O(pen)* and *C(lose)* commands within a ***BASIC*** program setting.

Equipment: IBM-compatible Computer System, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Pro-Arm Grid Sheet, IBM-type Parallel Interface Cable, Knowledge Transfer Manual, MARCRAFT Program Disk, and a Ruler (inches or millimeters).

____a. Check the system installation and interfacing to be sure that it is ready to go.

____b. Boot the computer system and apply power to the Pro-Arm.

____c. At the prompt **(A>_)** remove the **DOS** disk from floppy drive **A** and return it to its protective sleeve. Insert the disk containing the **BASIC** software (included with the computer in your classroom). *Note: If you are using an IBM system, leave the system disk in the floppy drive for the next step.*

____d. Boot your **BASIC** software up by typing **GWBASIC** and pressing **ENTER**. *Note: If you are using an IBM system, type in either **BASIC** or **BASICA**, and press **ENTER**.*

____e. Use the **TEST** and **RESET** buttons on the Pro-Arm's front panel to place it in its "**ZERO**" position.

____f. At the **BASIC** prompt, type the following program. *Note: Don't forget to press **ENTER** at the end of each line to place the line in memory.*

```
20 PRINT "Z":LPRINT "Z"
25 PRINT "M0,0,0,0,0,-2000"
30 LPRINT "M0,0,0,0,0,-2000"
40 PRINT "D10":LPRINT "D10"
50 PRINT "M0,0,0,0,0,2000":LPRINT "M0,0,0,0,0,2000"
60 PRINT "END":END
```

____g. Momentarily press the Pro-Arm's **RESET** button.

____h. Press **F2**. The gripper should fully open and then close. Did it?

____

____i. Press **F2** again. This time, during the 10-second delay caused by Statement 40, use a scale to measure the maximum gripper opening. Record your measurement below.

Maximum gripper opening = ________________

____j. The number of steps required to fully open the gripper is ________.

____k. The number of steps required to fully close the gripper is ________.

____l. Is a positive or negative number of steps required to open the gripper?

____

____m. Erase Statement 25 and retype Statements 30 and 50 as follows. Don't forget to press **ENTER** after typing in each statement.

```
30 PRINT "M0,0,0,0,0,-900":LPRINT "M0,0,0,0,0,-900"
50 PRINT "M0,0,0,0,0,900":LPRINT "M0,0,0,0,0,900"
```

____n. Press **F2** and measure the gripper opening again. Record your measurement below.

Gripper opening for minus 900 steps = __________

Note: The gripper can be programmed to open to any position between fully opened and fully closed by using the appropriate number of steps.

____o. Type **NEW**, and then press **ENTER**. This will erase the program currently in memory so that a new program may be entered. *Note: Failure to use the **NEW** command before typing in a new program can cause erroneous program execution when the program lines from two different programs are mixed together.*

____p. Type **CLS**, and then press **ENTER** to clear the screen.

____q. Now type in the following program. Be sure to type a capital letter **"O"** (**"oh"**) and not the integer (**"zero"**) in Statement 30.

```
20 PRINT "Z":LPRINT "Z"
30 PRINT "O":LPRINT "O"
40 PRINT "D10":LPRINT "D10"
50 PRINT "C":LPRINT "C"
60 PRINT "END":END
```

____r. Momentarily press the Pro-Arm's **RESET** button to be sure the gripper is fully closed and that the Pro-Arm is ready to receive a new set of commands from the computer.

____s. Press **F2**. The gripper should fully open and then close. Did it?

____

____t. Press **F2** again. This time measure and record the maximum gripper opening.

Maximum gripper opening = ________________

____u. Is the maximum gripper opening, using the **O**(pen) command, the same as that when using the **M**(ove) command with minus 2000 steps specified?

____

It should be!

____v. Again, execute the program by pressing **F2.** However, this time, place a small object, such as one of the experimental wood blocks (supplied), between the jaws of the gripper as they begin to close. Listen carefully and you will hear a **"click"** sound from the limit switch when the gripper is fully closed on the object.

____w. Once again, press **F2.** This will release the object as the gripper will again fully open and then close.

____x. Turn off the Pro-Arm's power supply and remove the **BASIC** Software Disk, and place it in its protective sleeve.

____y. Turn off the computer system, but leave the installation and interface connections as they are.

____z. This completes Lab Exploration 34, "Open or Close." Have your instructor initial your Knowledge Transfer Guide.

THE LIMIT COMMAND

It is possible to give the Pro-Arm a *M(ove)* command that exceeds its limits. When this happens, the arm will try to move past the axis limit, even though it is physically impossible. You will know when this happens because the arm will start to *"chatter."* Once a movement has attempted to exceed an axis limit, the axis will no longer be synchronized, and will not return to its zero position.

The Pro-Arm uses the *L(imit)* command to stop the arm from trying to move past its maximum travel position. This is done by the Pro-Arm's internal Z80 microprocessor, by comparing the parameters in any *M(ove)* or *P(osition)* command against the maximum allowable steps information, which is stored in the Pro-Arm's ROM memory chip. The format of this command is Ln where n is either a 1 or 0. The *1* will turn *ON (activate)* the limit operation, and a *0* will *(deactivate)* turn it *OFF*.

The following is an example of how this command is written.

```
50 PRINT "L1":LPRINT "L1"
80 PRINT "L0":LPRINT "L0"
```

LAB EXPLORATION 35

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____35. LIMIT

Purpose: This exploration will provide the student with practice in using the *L(imit)* command within a *BASIC* program setting.

Equipment: IBM-compatible Computer System, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Pro-Arm Grid Sheet, IBM-type Parallel Interface Cable, Knowledge Transfer Manual, and the MARCRAFT Program Disk.

____a. Check the system installation and interfacing to be sure that it is ready to go.

____b. Boot the computer system and apply power to the Pro-Arm.

____c. At the prompt **(A>_)** remove the **DOS** disk from floppy drive **A** and return it to its protective sleeve. Insert the disk containing the **BASIC** software (included with the computer in your classroom). *Note: If you are using an IBM system, leave the system disk in the floppy drive for the next step.*

____d. Boot your **BASIC** software up by typing **GWBASIC** and pressing **ENTER**. *Note: If you are using an IBM system, type in either **BASIC** or **BASICA**, and press **ENTER**.*

____e. Use the **TEST** and **RESET** buttons on the Pro-Arm's front panel to place it in its **"ZERO"** position.

____f. At the **BASIC** prompt, type the following program. *Note: Don't forget to press **ENTER** at the end of each line to place the line in memory.*

```
20 PRINT "Z":LPRINT "Z"
40 PRINT "M0,800,0,0,0,0":LPRINT "M0,800,0,0,0,0"
60 PRINT "M0,-800,0,0,0,0":LPRINT "M0,-800,0,0,0,0"
90 PRINT "END":END
```

____g. Statement 40 instructs the Pro-Arm to move its forearm **(shoulder axis)** upward. How many steps will it attempt to move the forearm?

____

What is the number of steps required to move the forearm to its maximum travel position?

____

____h. Momentarily press the Pro-Arm's **RESET** button and then press **F2**.

____i. The arm should have moved upward to its maximum travel position and then back downward. Describe what happened when it reached its maximum position..

____

____

Statement 40 instructed the Pro-Arm to move its forearm upward 800 steps. How many steps upward did it actually move?

____

Statement 60 instructed the Pro-Arm to move its forearm downward 800 steps. How many steps did it move?

____

Did the forearm return to its **ZERO** position?

____

If not, how many steps past the **ZERO** position did it move?

____

Did the **RED** error lamp come on?

____

If not, how would you know that there was, in fact, a programming error?

____

____

____

____

____j. Again, adjust the Pro-Arm to its **"HOME"** position by zero-positioning all axes.

____k. Add the following statement to the above program.

```
30 PRINT "L1":LPRINT "L1"
```

____l. Momentarily press the Pro-Arm's **RESET** button to be sure it is again ready to receive commands from the computer. Then, press **F2**.

____m. With Statement 30 inserted into the program, did the forearm return to its zero position?

____

Describe the arm movement.

____

____

____n. Did the **RED** error lamp come on this time?

____

If it did, you may turn it **OFF** by pressing the **RESET** button. *Note: Since the L(imit) command was activated by Statement 30, the arm was not allowed to exceed its step count beyond the maximum limit of travel. However, it will be necessary for you to again zero-position the shoulder axis, since the arm remained at its maximum, upward, shoulder position.*

____o. Add Statements 50, 70, and 80 to the program, as shown below.

```
20 PRINT "Z":LPRINT "Z"
30 PRINT "L1":LPRINT "L1"
40 PRINT "M0,800,0,0,0,0":LPRINT "M0,800,0,0,0,0"
50 PRINT "N":LPRINT "N"
60 PRINT "M0,-800,0,0,0,0":LPRINT "M0,-800,0,0,0,0"
70 PRINT "M0,200,0,0,0,0":LPRINT "M0,200,0,0,0,0"
80 PRINT "N":LPRINT "N"
90 PRINT "END":END
```

*Note: Without knowing the maximum limit positions of the robot, it may appear that Statement 40 would move the forearm **(shoulder axis)** 800 steps upward and that Statement 50 would return it to its **"HOME"** position. Statement 60 would then move the forearm 800 steps downward, Statement 70 would move the arm 200 steps upward, and Statement 80 would again return the arm to its **"HOME"** position.*

____p. Momentarily press the Pro-Arm's **RESET** button and then press **F2.** Did the forearm move upward as instructed in Statement 40?

____

Did the arm return to its "**HOME**" position as instructed in Statement 50?

____

Did the **RED** error lamp come on?

____

____q. Did the forearm move downward as instructed in Statement 60?

____

If not, momentarily press the Pro-Arm's **RESET** button. Describe the arm movement.

____

____

____

Since program Statement 60 instructed the forearm to move downward 200 steps past its maximum limit (minus 600), did the **RED** error lamp come **ON** and did the **L**(imit) command halt the arm movement?

____

*Note: When you depressed the **RESET** button in Step p, the Pro-Arm returned to its command default values. The default value of the **L**(imit) command is **0 (OFF)**.*

____r. Turn off the Pro-Arm's power supply, remove the **BASIC** Software Disk, and place it in its protective sleeve.

____s. Turn off the computer system, but leave the installation and interface connections as they are.

____t. This completes Lab Exploration 35, "Limit," and the PRO-ARM COMMANDS activities. Have your instructor initial your Knowledge Transfer Guide.

This page intentionally left blank

This page intentionally left blank

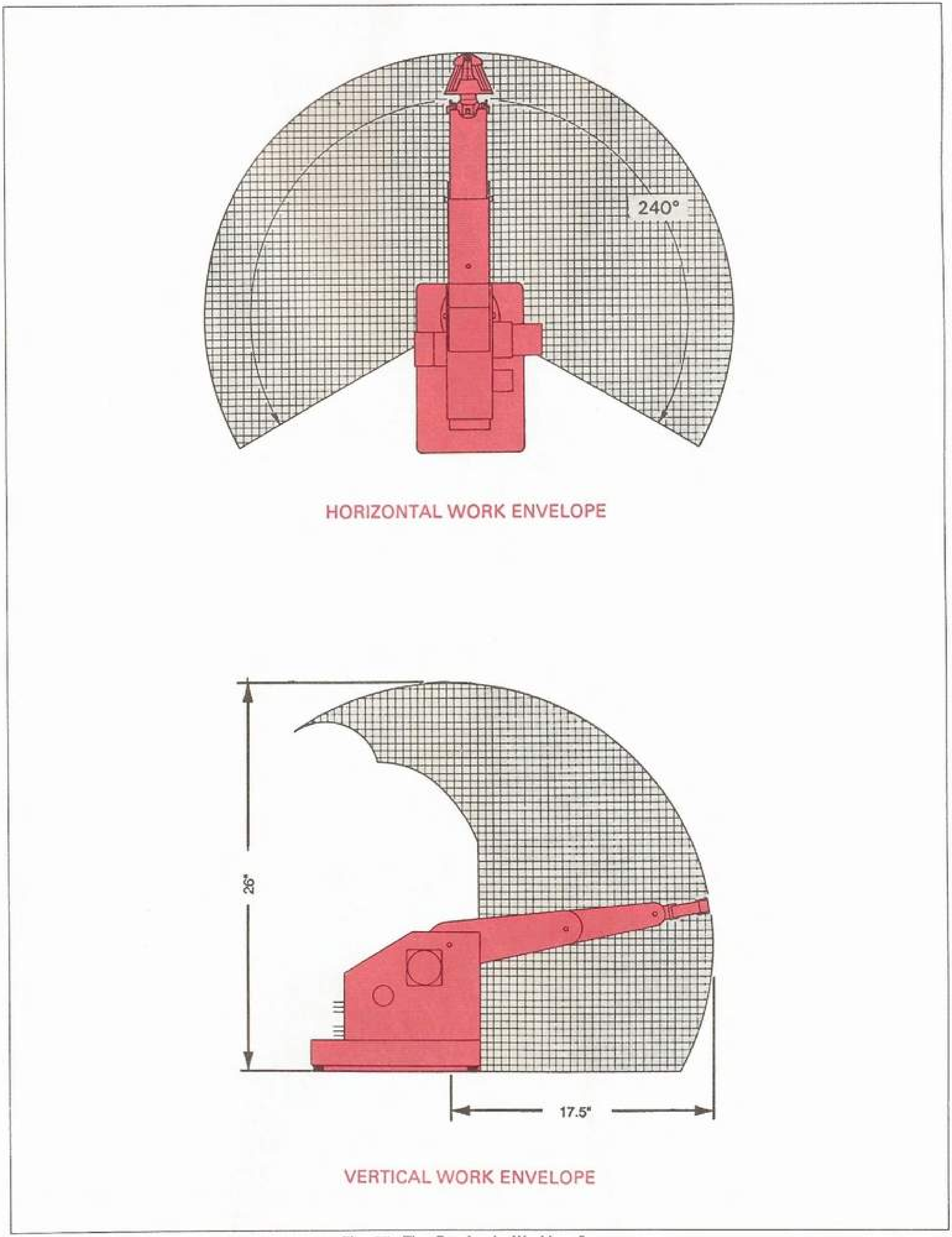


Fig. 77: The Pro-Arm's Working Space

DISCUSSION

The working space, or work envelope, of an industrial robot is the volume of space in which the arm can move. It is always important to know the working space of a robot since this information determines the area in which a task is to be performed. It also has special importance in that it is used to establish a danger zone in which individuals are not permitted at any time when the robot is in operation. This safety rule is strictly enforced by industry.

The Pro-Arm's working space is determined by both the ***Horizontal Work Envelope*** and the ***Vertical Work Envelope***, as shown in Fig. 77.

In the following explorations, you will determine both the horizontal and the vertical work envelopes for the Pro-Arm.

HORIZONTAL WORK ENVELOPE

The horizontal work envelope is determined by the arm ***"reach"*** and ***"swing."*** The ***"reach"*** is the maximum distance from the center line of the Pro-Arm ***(base axis)*** to the end of its arm ***(gripper)***, with the arm fully extended in a horizontal position. The ***"swing"*** is the rotation about the center line of the Pro-Arm ***(base axis).***

LAB EXPLORATION 36

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____36. HORIZONTAL

Purpose: This exploration will provide the student with a sample ***BASIC*** program for determining the horizontal work envelope of the Pro-Arm.

Equipment: IBM-compatible Computer System, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Pro-Arm Grid Sheet, IBM-type Parallel Interface Cable, Knowledge Transfer Manual, MARCRAFT Program Disk, Felt-tip Pen (water soluble), Protractor, Soft Pencil, and a Damp Cloth.

____a. Check the system installation and interfacing to be sure that it is ready to go, and ensure that the Pro-Arm is properly placed onto the grid sheet in the space provided.

____b. Boot the computer system and apply power to the Pro-Arm.

____c. At the prompt **(A>_)** remove the **DOS** disk from floppy drive **A** and return it to its protective sleeve. Insert the disk containing the **BASIC** software (included with the computer in your classroom). *Note: If you are using an IBM system, leave the system disk in the floppy drive for the next step.*

____d. Boot up your **BASIC** software by typing **GWBASIC** and pressing **ENTER.** *Note: If you are using an IBM system, type either* ***BASIC*** *or* ***BASICA****, and press* ***ENTER.***

____e. Use the **TEST** and **RESET** buttons on the Pro-Arm's front panel to place it in its "**ZERO**" position.

____f. At the **BASIC** prompt, type in the following program. *Note: Don't forget to press **ENTER** at the end of each line to place the line in memory.*

```
20 PRINT "Z":LPRINT "Z"
25 PRINT "M0,-450,300,0,0,0"
30 LPRINT "M0,-450,300,0,0,0"
35 PRINT "M-1000,0,0,0,0,0"
40 LPRINT "M-1000,0,0,0,0,0"
50 PRINT "D5":LPRINT "D5"
60 PRINT "FOR X=1 TO 8":FOR X=1 TO 8
70 PRINT "M250,0,0,0,0,0":LPRINT "M250,0,0,0,0,0"
80 PRINT "D5":LPRINT "D5"
90 PRINT "NEXT X":NEXT X
100 PRINT "N":LPRINT "N"
110 PRINT "END":END
```

*Note: Statement 30 will fully extend the arm to a horizontal position. Statement 40 will swing the arm to its maximum counterclockwise position. Statements 60 through 90 are using the **FOR NEXT** command to perform a loop of repeated movements. Every line between the words **FOR** and **NEXT** will be repeated by the number of times dictated by the parameter following the word **FOR**. In this case, the arm will swing in a clockwise direction, stopping 8 times, at increments of 250 steps. Statements 50 and 80 will cause the arm movement to delay for 5 seconds at various positions as it swings through its **Horizontal Envelope**. Notice that Statement 80 falls between Statements 60 and 90, and, therefore, becomes part of the **FOR NEXT** loop.*

____g. Momentarily press the Pro-Arm's **RESET** button.

____h. Press **F2**.

____i. Each time the arm pauses, use the felt-tip pen to make a mark on the grid sheet directly below the tip end of the gripper. If necessary, **RUN** the program several times.

____j. Turn off the Pro-Arm's power supply and remove the **BASIC** Software Disk, placing it in its protective sleeve.

____k. Turn off the computer system, but leave the installation and interface connections as they are.

____l. Carefully remove the grid sheet from under the Pro-Arm.

____m. Using the felt-tip pen, draw a smooth arc through the marks you made on the grid sheet in **Step i**. Refer to Fig. 78.

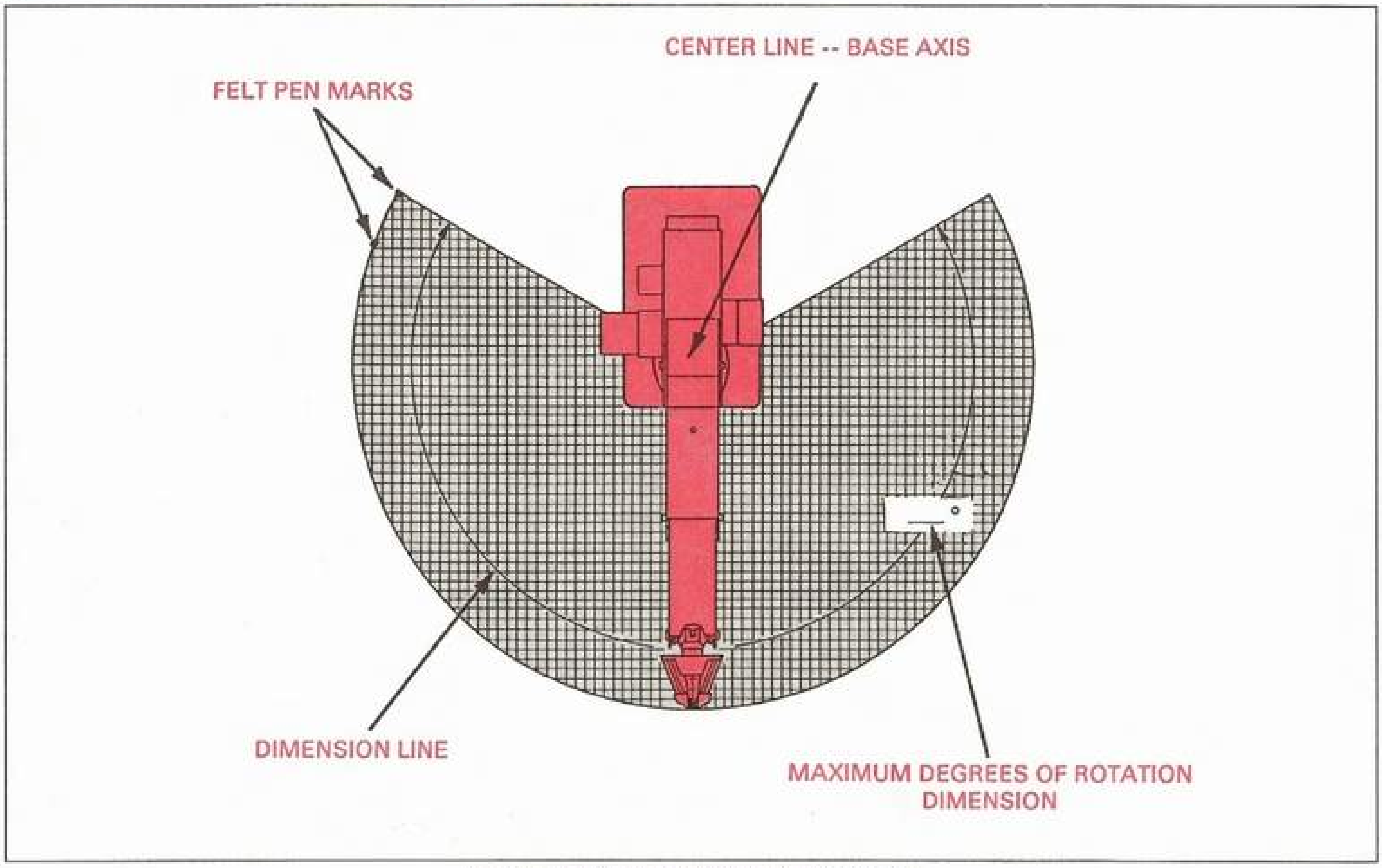


Fig. 78: Horizontal Envelope Nomenclature

____n. Next, draw a line connecting each end of the arc to the center line of the base axis (marked "**BASE AXIS**" on the grid sheet). Refer again to Fig. 78.

____o. Using the protractor, measure the maximum degrees-of-rotation of the arm about the base axis. Record this measurement below.

Maximum degrees-of-rotation = ____________

____p. Copy the Horizontal Work Envelope from the grid sheet onto the appropriate portion of the Working Space Chart in Fig. 79.

____q. Draw a dimension line on the chart indicating the rotational swing, and then add the degrees-of-rotation measurement from **Step o.** Refer to the example in Fig. 78.

____r. Using a soft pencil, lightly shade-in the work envelope, again referring to Fig. 78.

____s. Using a damp cloth, clean off the felt-tip pen marks from the grid sheet.

____t. This completes Lab Exploration 36, "Horizontal." Have your instructor initial your Knowledge Transfer Guide.

VERTICAL WORK ENVELOPE

In this activity, you will determine the Pro-Arm's Vertical Work Envelope by first placing the grid sheet next to the side of the Pro-Arm, and, then, by moving the arm through the vertical envelope. As the arm moves through its vertical work envelope, it will pause momentarily at selected intervals. Using a felt-tip pen, you will make a series of mark on the grid sheet indicating the end points on the gripper. You will then connect the marks together to display the vertical work envelope of the Pro-Arm.

Since the Pro-Arm is an articulate coordinate (jointed-arm) type of robot, the vertical work envelope will resemble the characteristic "**tear**" shape.

LAB EXPLORATION 37

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

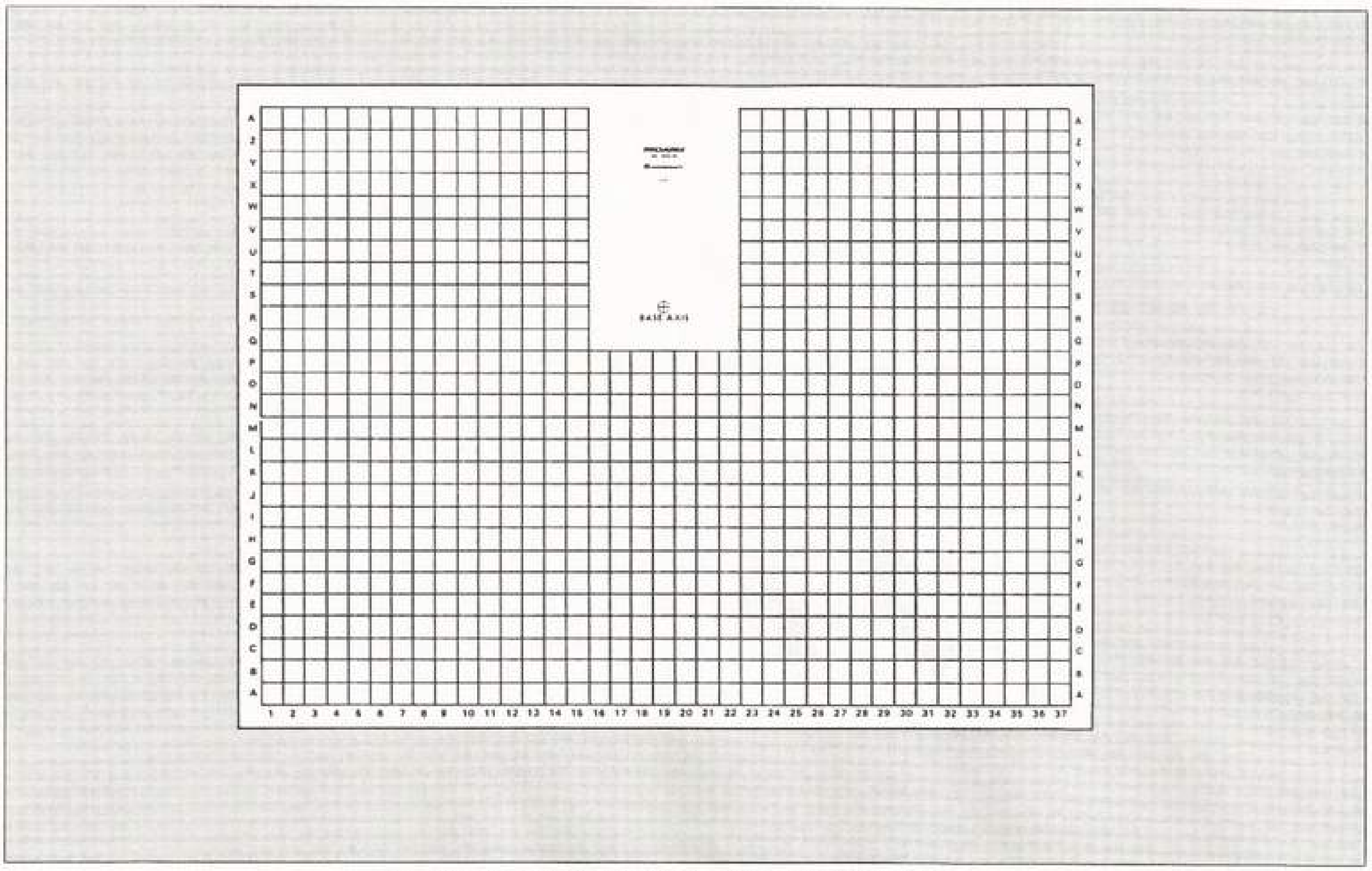


Fig. 79: Working Space Chart—Horizontal Work Envelope

____37. VERTICAL

Purpose: This exploration will provide the student with a sample ***BASIC*** program for determining the vertical work envelope of the Pro-Arm.

Equipment: IBM-compatible Computer System, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Pro-Arm Grid Sheet, IBM-type Parallel Interface Cable, Knowledge Transfer Manual, MARCRAFT Program Disk, Felt-tip Pen (water soluble), Cardboard backing, Soft Pencil, Damp Cloth, and Masking Tape.

____a. Check the system installation and interfacing to be sure that it is ready to go.

____b. Cut a piece of cardboard to the identical dimensions of the grid sheet and use several pieces of masking tape to mount the grid sheet to the cardboard backing.

____c. Boot the computer system and apply power to the Pro-Arm.

____d. At the prompt (**A>-**) remove the **DOS** disk from floppy drive **A** and return it to its protective sleeve. Insert the disk containing the **BASIC** software (included with the computer in your classroom). *Note: If you are using an IBM system, leave the system disk in the disk drive.*

____e. Boot up your **BASIC** software by typing in **GWBASIC** and pressing **ENTER**. *Note: If you are using an IBM system, type either **BASIC** or **BASICA**, and press **ENTER**.*

____f. Use the **TEST** and **RESET** buttons on the Pro-Arm's front panel to place it in its "**ZERO**" position.

____g. Place the grid sheet behind and parallel to the Pro-Arm, and align the vertical grid line between numbers **15** and **16** with the center line of the base axis, as shown in Fig. 80.

____h. Press the Pro-Arm's **RESET** button.

____i. At the **BASIC** prompt, load and run the Vertical Envelope program **VERTEN.BAS** by typing **RUN "VERTEN**, and then pressing **ENTER**.

Note: When executing this program, the arm will first move to a position near the table top and the Pro-Arm's body. It will then move outward until it is fully extended, remaining about one inch (25 mm) above the table top. Even though it would be possible for the arm to skim along the surface of the table, as a safety precaution, it should remain well above it.

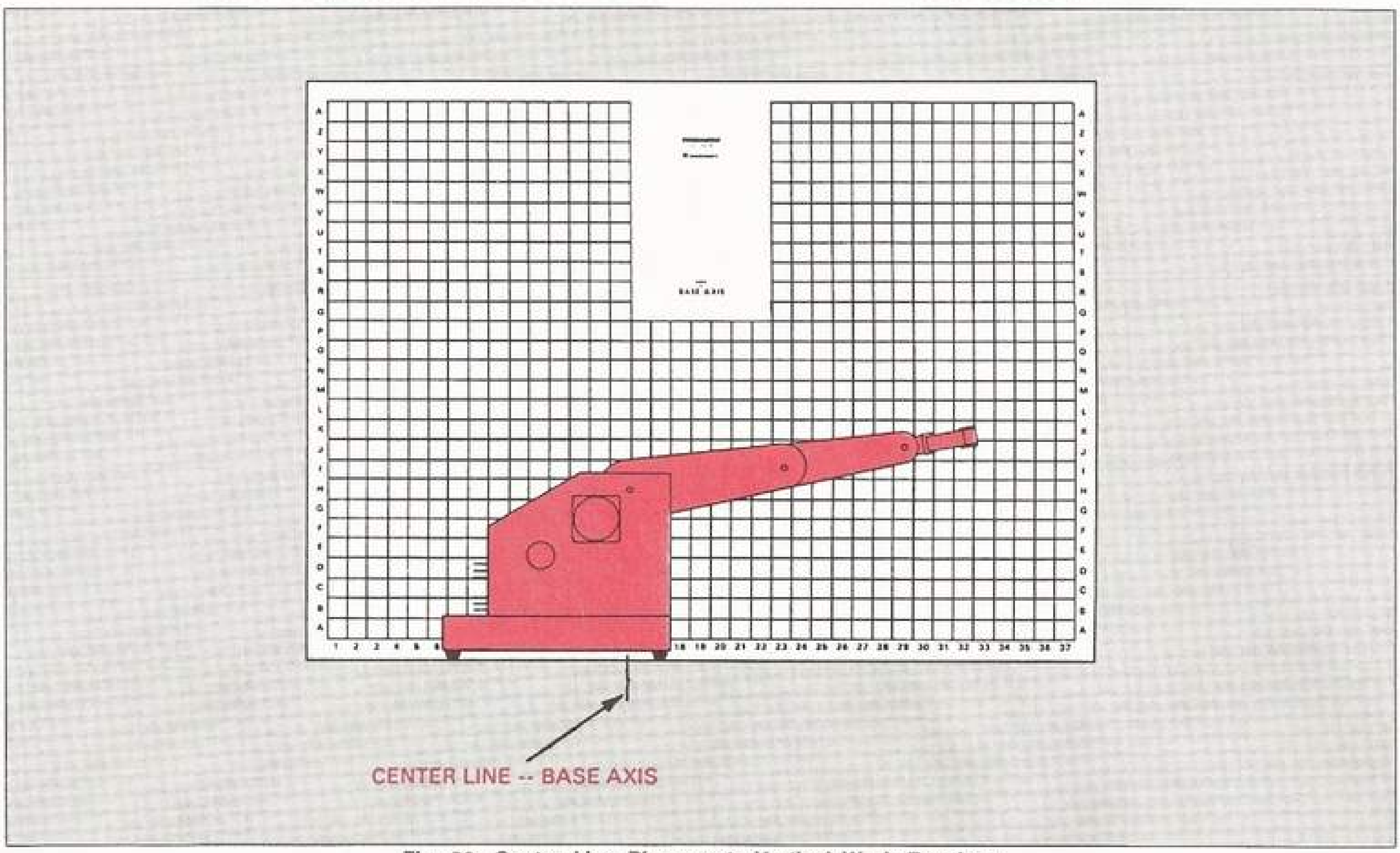


Fig. 80: Center Line Placement—Vertical Work Envelope

____j. Using the felt-tip pen (water soluble), make a mark on the grid sheet at the end of the gripper each time the gripper pauses for a few seconds. If necessary, **RUN** the program several times.

____k. Turn off the Pro-Arm's power supply and remove the **BASIC** Software Disk, placing it in its protective sleeve.

____l. Turn off the computer system, but leave the installation and interface connections as they are.

____m. Lay the grid sheet down and carefully draw a straight horizontal line through the first four marks, as shown in Fig. 81. Then, continue with this line by making a series of smooth, curved lines through the markings, until you reach the last mark.

____n. The last mark you made should be nearly in line with the imaginary front of the body of the Pro-Arm. Draw a vertical line from this mark down to the horizontal line that indicates the top of the work surface. Your completed work envelope should be similar to that shown in Fig. 81.

____o. Copy the vertical work envelope from the grid sheet onto the Working Space Chart in Fig. 82.

____p. Measure the horizontal distance, on the Working Space Chart, from the center line of the base axis to the point on the work envelope that is the greatest distance from the base axis, as depicted in Fig. 81. Record this measurement below. *Note: Each line represents a distance of one inch (25 mm).*

Horizontal distance = ____________________

____q. Draw a dimension line on the Working Space Chart indicating the horizontal distance, and then add the measurement from **Step p**. Refer to Fig. 81, again.

____r. Measure the vertical distance of this work envelope from the work surface, and record this measurement in the space provided.

Vertical distance = ____________________

____s. Draw a dimension line on the Working Space Chart indicating the vertical distance, and then add the measurement from **Step r**. Refer to Fig. 81, if necessary.

____t. Using a damp cloth, clean the felt-tip pen marks off the grid sheet and wipe it dry.

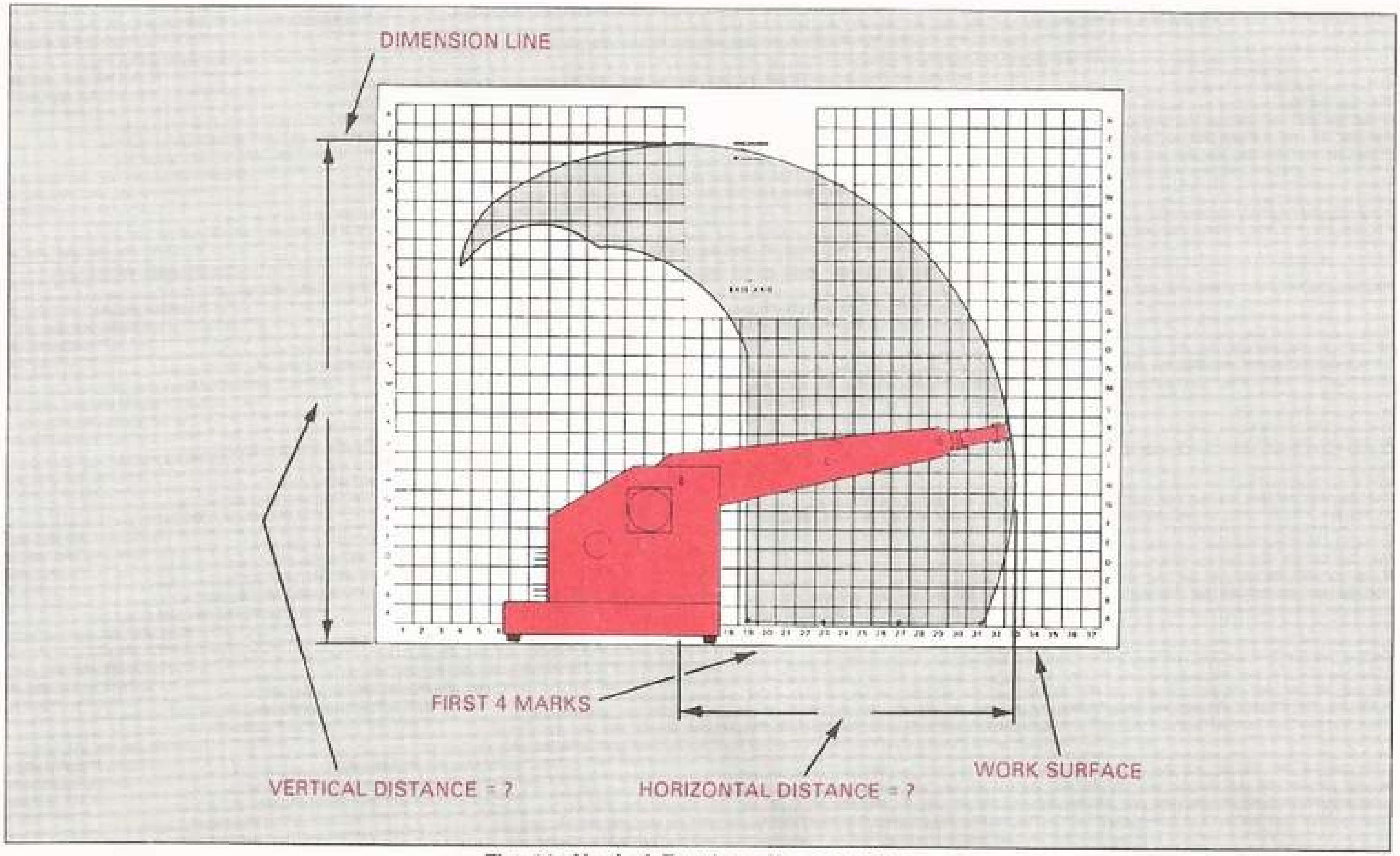


Fig. 81: Vertical Envelope Nomenclature

____u. Remove the Pro-Arm grid sheet from the cardboard backing and reposition it under the Pro-Arm in the space provided.

____v. List several commands used in the **VERTEN.BAS** program that were new.

(1) ______________________________

(2) ______________________________

(3) ______________________________

____w. The **REM** command merely allows the programmer to document the program, to make it easier for the user to understand. Nothing that follows the word **REM** on a program line is executed.

____x. The **WIDTH** command indicates how many text characters the screen should display per line. For most high-resolution monitors, the width is automatically displayed at 80 characters per line, but some medium-resolution monitors have a habit of kicking text back at 40 characters per line. The **WIDTH 80** statement in the **VERTEN.BAS** program will prevent this from happening.

____y. **KEY OFF** is an interesting command you can use whenever you want to get rid of the Function Key Bar at the bottom of the **BASIC** screen layout. The Function Key Bar is there to remind the user of the meanings of the function keys. By simply pressing the function key that has been assigned to the indicated command, you can avoid having to type in the entire command. You have been using **F2** to **RUN** programs frequently, rather than typing out the **RUN** command, and then pressing **ENTER**. To clear the Function Key Bar off the screen, simply type **KEY OFF**, and press **ENTER**. The function keys still work as before. To restore the Function Key Bar, simply type **KEY ON**, and press **ENTER**.

____z. This completes Lab Exploration 37, "Vertical," and the WORK ENVELOPE activities. Please have your instructor initial your Knowledge Transfer Guide.

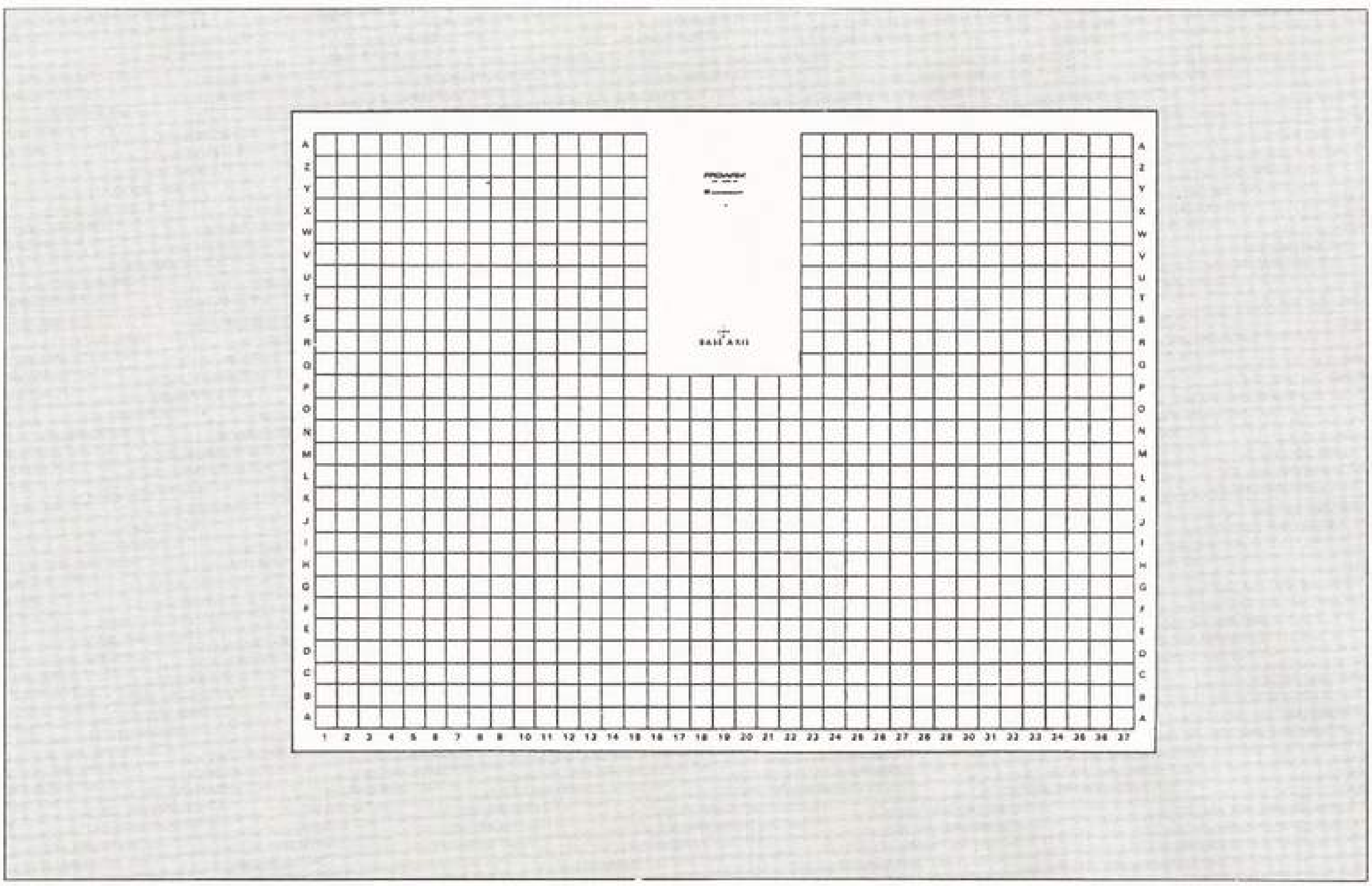


Fig. 82: Working Space Chart—Vertical Work Envelope

This page intentionally left blank

This page intentionally left blank

This page intentionally left blank

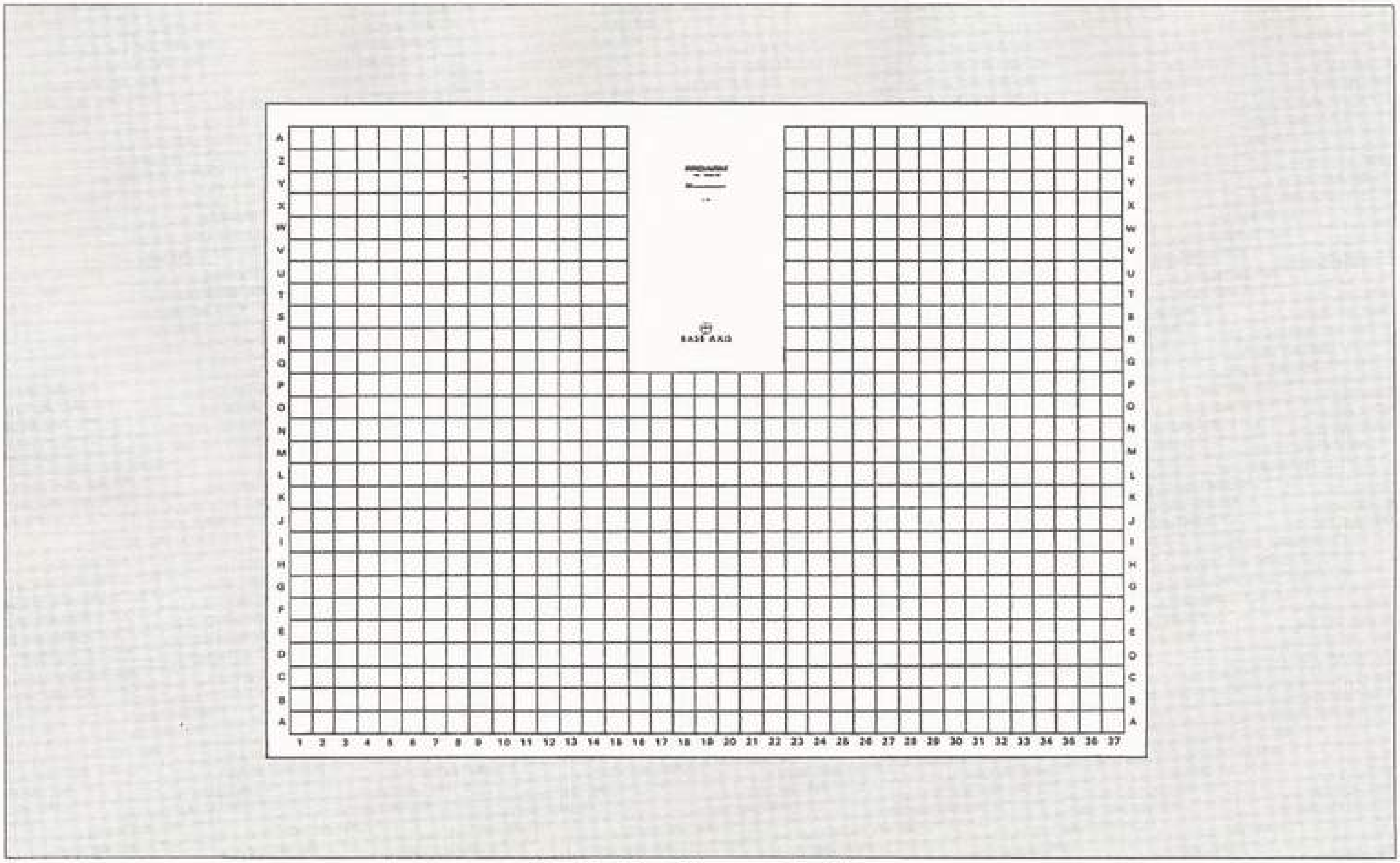


Fig. 83: First Placement Position

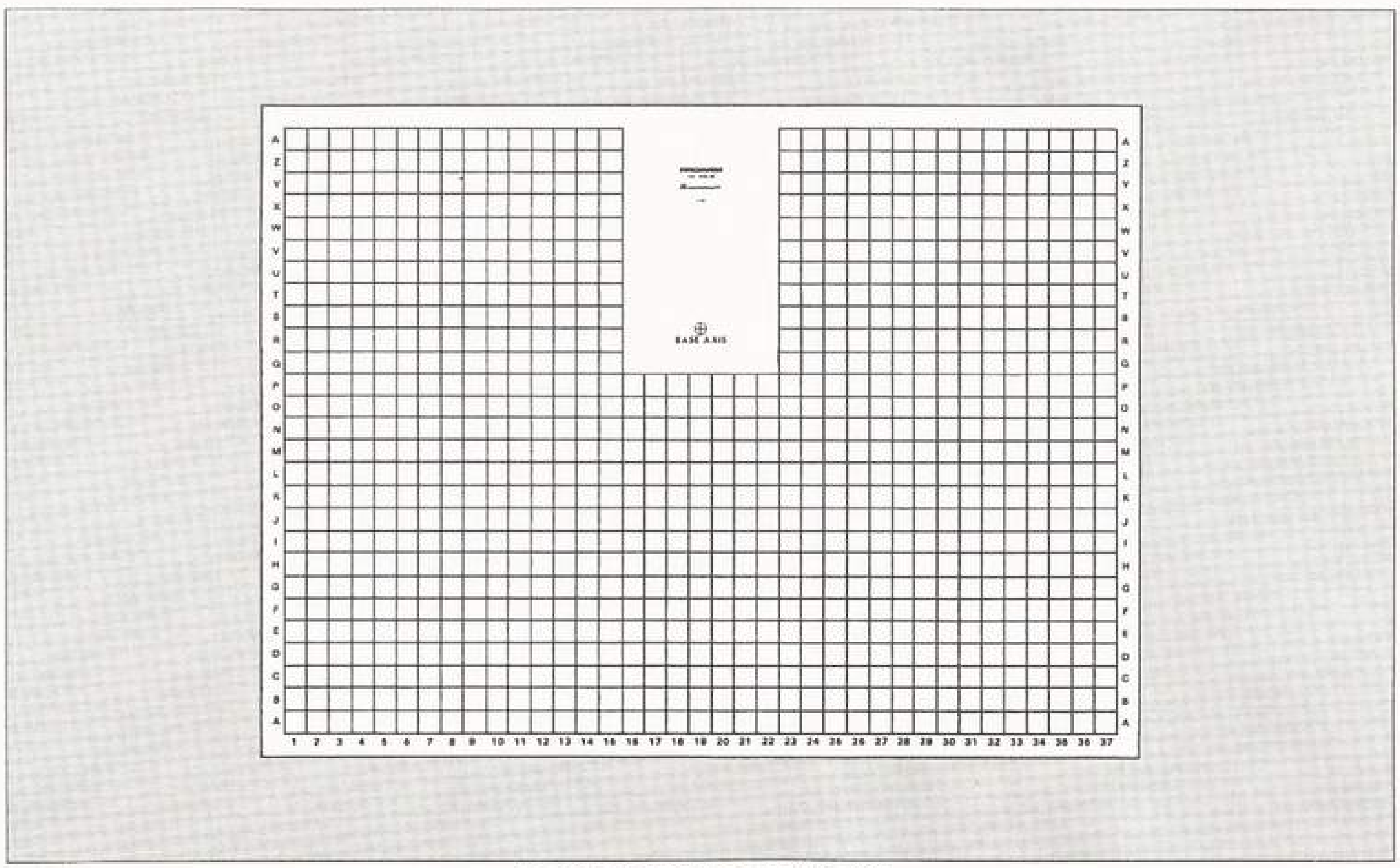


Fig. 84: Second Placement Position

REPEATABILITY

DISCUSSION

Repeatability is the degree to which a manipulator arm is able to repeat its moves and return to a specific position each time a program is executed. Normally, the fewer axes an industrial robot has, the better its repeatability. This is due to the cumulative error that is developed when moving several axes at the same time.

In the following exploration, you will run a program that moves a small object from one side of the grid sheet to the other, and then back again. The program repeats this cycle five times, so that you may observe the repeatability of the Pro-Arm. Each time the arm places the object, it will return to its *"HOME"* position before picking up the object again and moving it to the other side of the grid sheet. This will give you time to observe the placement position.

LAB EXPLORATION 38

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

___38: REPEAT

Purpose: This exploration will provide the student with a sample ***BASIC*** program for determining the repeatability of the Pro-Arm.

Equipment: IBM-compatible Computer System, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, Pro-Arm Grid Sheet, IBM-type Parallel Interface Cable, Knowledge Transfer Manual, MARCRAFT Program Disk, and a Soft Pencil.

___a. Check the system installation and interfacing to be sure that it is ready to go.

___b. Boot the computer system and apply power to the Pro-Arm.

___c. At the prompt **(A>_)** remove the **DOS** disk from floppy drive **A** and return it to its protective sleeve. Insert the disk containing the **BASIC** software (included with the computer in your classroom).

Note: If you are using an IBM system, leave the system disk in the floppy drive for the next step.

___d. Boot up your **BASIC** software by typing in **GWBASIC** and pressing **ENTER**. *Note: If you are using an IBM system, type either **BASIC** or **BASICA**, and press **ENTER**.*

___e. Use the **TEST** and **RESET** buttons on the Pro-Arm's front panel to place it in its "**ZERO**" position.

___f. Be sure the Pro-Arm is placed on the grid sheet in its proper position and that the sides of its base are very closely aligned with the grid lines.

___g. Press the Pro-Arm's **RESET** button to be sure that the data from the computer will get through.

___h. Place one of the small, round blocks provided with the Pro-Arm (or, any small object that is about 1 inch in height) in the center of grid square **D19.**

___i. At the **BASIC** prompt, load and run the Repeat Control program **REPEAT.BAS** by typing **RUN "REPEAT**, and then pressing **ENTER**.

___j. Observe the arm movement. The arm will first move ________________,

pick up the block, move in a ____________________ direction,

and then place the block on the grid sheet. The arm will then momentarily return to its ____________ position.

___k. Observe and record the exact location of this first placement position, by drawing a circle on the corresponding position of the grid shown in Fig. 83.

___l. The arm will next return, pick up the block, and place it on the other side of the grid sheet. Observe and record the exact location of this second placement position of the block, by drawing a circle on the corresponding portion of the grid shown in Fig. 84.

___m. During the fourth or fifth cycle, again observe the first and second placement positions of the block, and record them in Figs. 83 and 84, respectively.

____n. After the arm has completed the five pick-and-place cycles, it should return the experimental block to its original position in grid square **D19**. Did it?

____

____o. Measure the difference, if any, between the first and last placements, as recorded in Fig. 83.

Placement difference (Fig. 83) = _________ inches

____p. Measure the difference, if any, between the first and last placements, as recorded in Fig. 84.

Placement difference (Fig. 84) = _________ inches

____q. Turn off the Pro-Arm's power supply and remove the **BASIC** Software Disk, placing it in its protective sleeve.

____r. Turn off the computer system, but leave the installation and interface connections as they are.

____s. This completes Lab Exploration 38, "Repeat," and the REPEATABILITY activities. Have your instructor initial your Knowledge Transfer Guide.

PROGRAMMING INTERFACE FEATURES

DISCUSSION

The programming aspects of the Pro-Arm's interfacing capabilities are the subject of the following explorations. The physical aspects of the interfacing features were discussed, in detail, earlier in this section. Review those discussions, if necessary.

PRO-ARM'S INTERLOCK

The following exploration will demonstrate how the Pro-Arm's interlock feature can be tested.

LAB EXPLORATION 39

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

____39. INTERLOCK

Purpose: This exploration will provide the student with a simple test procedure for determining the working status of the Pro-Arm's interlock.

Equipment: IBM-compatible Computer System, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, IBM-type Parallel Interface Cable, Knowledge Transfer Manual, and a 2-Pin Connector (wired).

____a. While checking the system installation and interfacing, plug one of the wired, 2-pin connectors (supplied) into the Pro-Arm's Interlock Connector shown in Fig. 85. Make sure that the two wires are not touching each other.

____b. Boot the computer system and apply power to the Pro-Arm.

____c. At the prompt **(A>_)** remove the **DOS** disk from floppy drive **A** and return it to its protective sleeve. Insert the disk containing the **BASIC** software (included with the computer in your classroom). *Note: If you are using an IBM system, leave the system disk in the floppy drive for the next step.*

____d. Boot up your BASIC software by typing in **GWBASIC** and pressing **ENTER**. *Note: If you are using an IBM system, type either* ***BASIC*** *or* ***BASICA****, and press* ***ENTER.***

____e. Use the **TEST** and **RESET** buttons on the Pro-Arm's front panel to place it in its "**ZERO**" position.

____f. At the **BASIC** prompt, type in the following program lines. Remember to press **ENTER** after each line is completed.

```
20 PRINT "Z":LPRINT "Z"
30 PRINT "M150,0,0,0,0,0":LPRINT "M150,0,0,0,0,0"
40 PRINT "D3":LPRINT "D3"
50 PRINT "M0,-300,0,0,0,0":LPRINT "M0,-300,0,0,0,0"
60 PRINT "D3":LPRINT "D3"
70 PRINT "W":LPRINT "W"
80 PRINT "N":LPRINT "N"
90 PRINT "END":END
```

____g. Press the Pro-Arm's **RESET** button to be sure that the data from the computer will get through.

____h. Press **F2**.

____i. Observe the first movement of the arm. It moves along the ____________ axis in a __________________ direction for _________ steps. It then waits for how long?

____

____j. Observe the second movement of the arm. It moves along the __________axis in a ________________ direction for _________ steps. It then waits for how long?

____

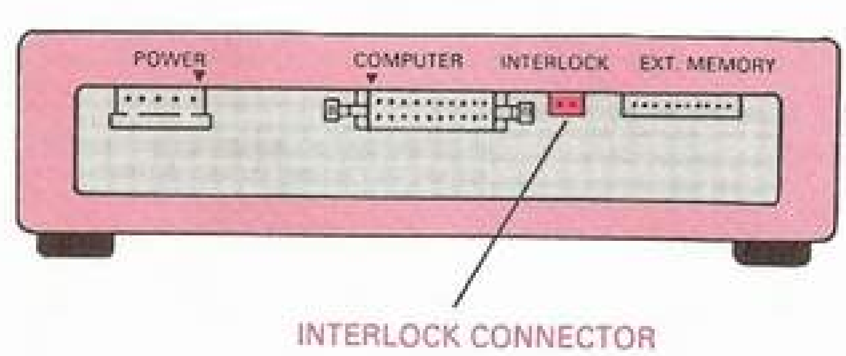


Fig. 85: The Pro-Arm's Interlock Connector

____k. Does the third movement of the arm begin after the second delay?

____

If not, why?

____

____

____

____l. Simulate a closed-circuit state, at the interlock connector, by twisting the bare ends of the wires from the 2-pin connector together. Describe any movement of the arm that occurs.

____

____m. Press **F2** to run the program again. Are the first and second delays now equal in duration?

____

Note: Since the bare ends of the wires from the 2-pin connector are still twisted together, the Pro-Arm sees a closed-circuit state when the "W" command is executed. Therefore, no reason exists for the Pro-Arm to halt its normal movement.

_____n. Unplug the 2-pin connector, and press **F2** again. When the interlock is activated, press the Pro-Arm's **RESET** button. Is the result the same as in **Step l**?

____

Why?

____

____

*Note: When the **RESET** button was pressed, the data inside the Pro-Arm's memory was erased, including the zero-positioning data. After the interlock was deactivated, the "N(est)" command was sent to the Pro-Arm, but had no effect. This was because the Pro-Arm assumed its new position (the position it had when its **RESET** button was pressed) to be the zero position.*

____o. Turn off the Pro-Arm's power supply and remove the **BASIC** or **DOS** System Disk, placing it in its protective sleeve.

____p. Turn off the computer system.

____q. This completes Lab Exploration 39, "Interlock." Have your instructor initial your Knowledge Transfer Guide.

4-BIT OUTPUT PORT

The following exploration will demonstrate how the Pro-Arm's 4-bit output port can be tested. See Fig. 86 for the pin assignments.

PIN NO.	OUTPUT SIGNAL
1	DB3
2	DB2
3	DB1
4	DB0
5	GND
6	GND

Fig. 86: The 4-Bit Output-Port Pin Assignment

LAB EXPLORATION 40

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____40. 4-BIT OUT

Purpose: This exploration will provide the student with a simple test procedure for determining the working status of the Pro-Arm's 4-bit output port.

Equipment: IBM-compatible Computer System, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, IBM Parallel Interface Cable, Knowledge Transfer Manual, 6-Pin Connector (wired), Phillips Screwdriver, and a Voltmeter, with clip leads.

____a. With the power cord and the interface cable still connected to the Pro-Arm, carefully set the Pro-Arm on its back, as shown in Fig. 87.

*Caution: **DO NOT** pull or crimp the power or interface cables. We want these connections to remain operable during the remainder of this exploration.*

____b. Next, using the Phillips-head screwdriver, remove the six machine screws holding the base plate to the body. Remove the body baseplate, so that the Pro-Arm's PCB is open to view.

____c. Locate **Connector 4** on the PCB (see Fig. 88) and insert the 6-pin connector plug (wired) onto it, making sure none of the six wires are touching each other.

____d. Connect the dc voltmeter, as shown in Fig. 89. The **BLACK** lead should be clipped to the **NEGATIVE** lead of the power input capacitor, **C0**. The **RED** lead should be clipped to the wire which is plugged to **Pin 4 (DB0)** of **Connector 4.**

Caution: Make sure none of the other wires at Connector 4 are touching the RED voltmeter lead.

____e. Turn the voltmeter on, and set it for a voltage reading of **5-V dc**.

____f. Boot the computer system and apply power to the Pro-Arm.

____g. At the prompt **(A>_)** remove the **DOS** disk from floppy drive **A** and return it to its protective sleeve. Insert the disk containing the **BASIC** software (included with the computer in your classroom). *Note: If you are using an IBM system, leave the system disk in the floppy drive for the next step.*

____h. Boot up your **BASIC** software by typing in **GWBASIC**, and pressing **ENTER**.

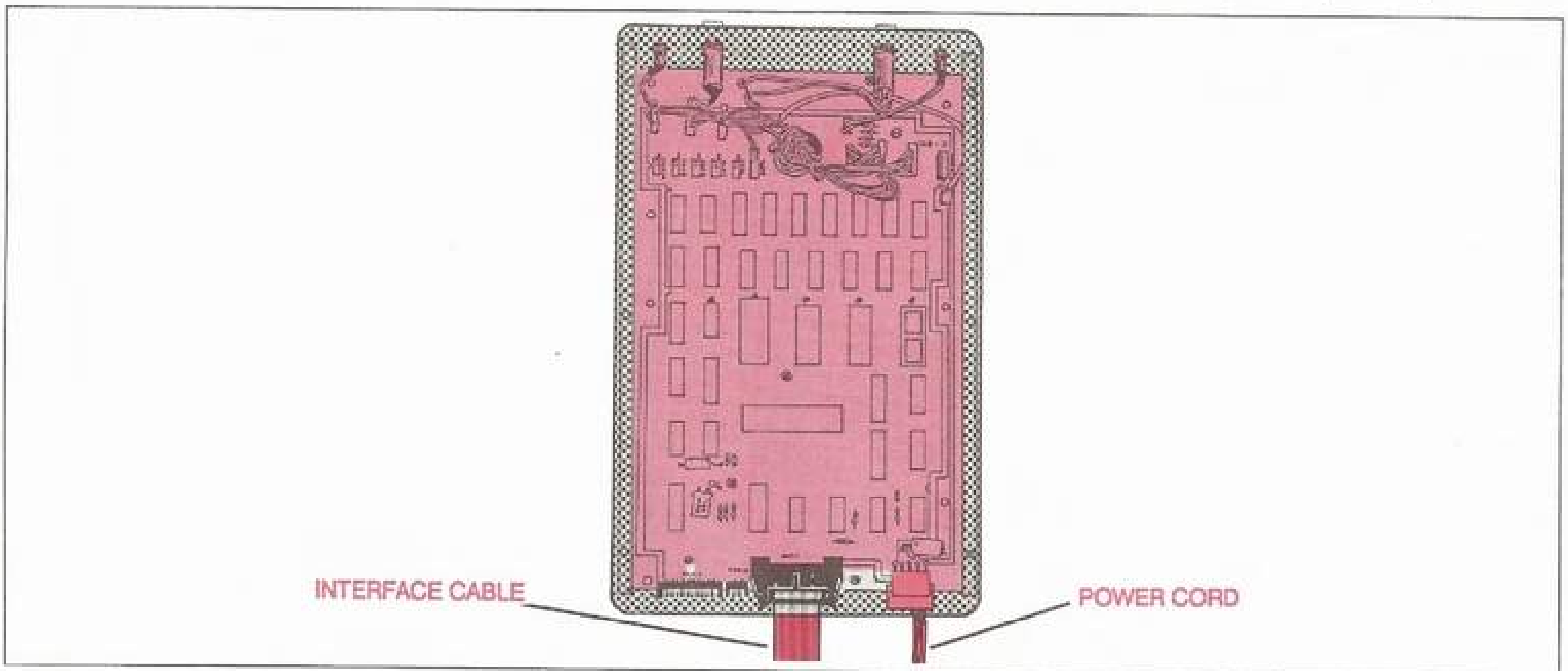


Fig. 87: Pro-Arm with Base Up

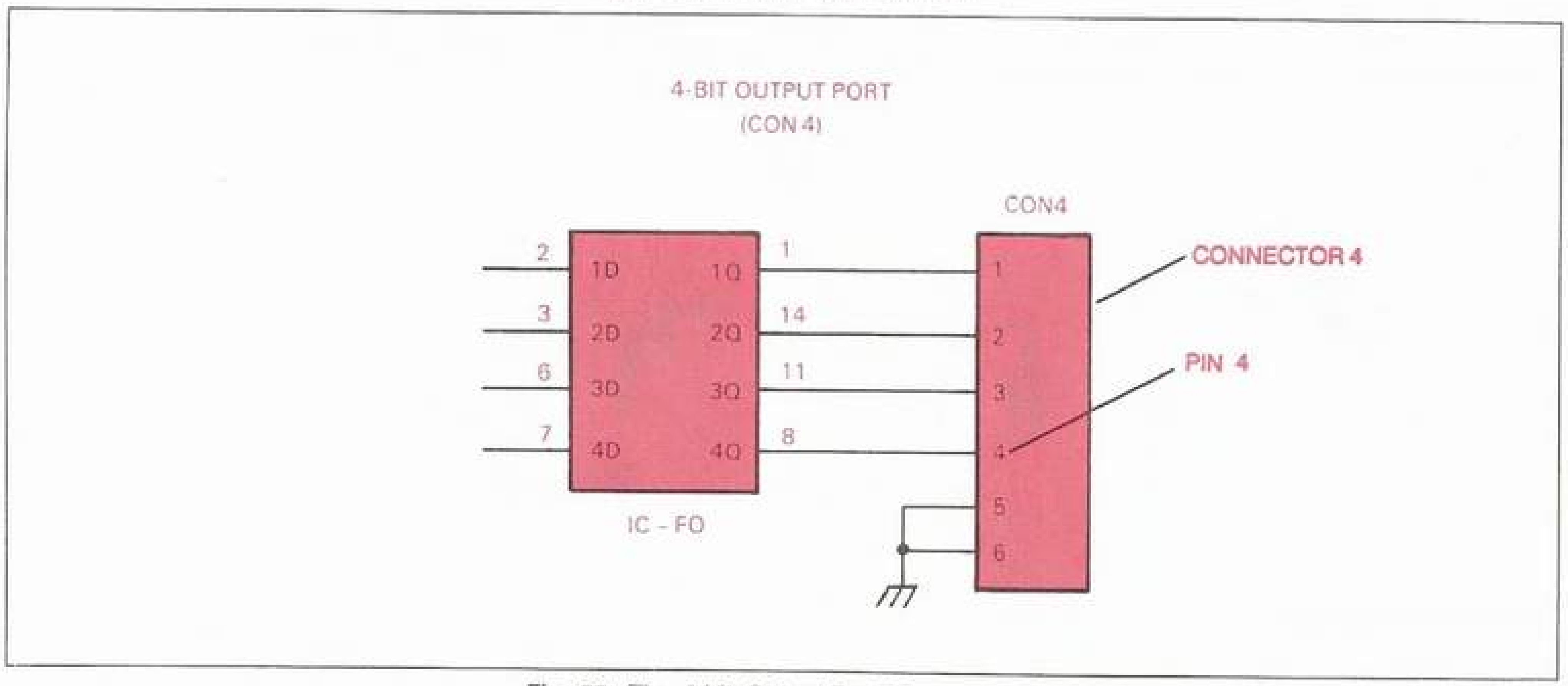


Fig. 88: The 4-bit Output-Port Connector 4

*Note: If you are using an IBM system, type in either **BASIC** or **BASICA**, and press **ENTER**.*

____i. At the **BASIC** prompt, type in the following program lines. *Note: Remember to press **ENTER** after each line is completed.*

```
20 PRINT "Z":LPRINT "Z"
30 PRINT "D5":LPRINT "D5"
40 PRINT "Q1":LPRINT "Q1"
50 PRINT "D5":LPRINT "D5"
60 PRINT "Q0":LPRINT "Q0"
70 PRINT "END":END
```

____j. Press the Pro-Arm's **RESET** button to be sure that the data from the computer will get through.

____k. Press **F2** and watch the voltmeter for a reading. *Note: No movement of the arm should be taking place.*

____l. What happens when Statement 40 appears on the screen?

____

____

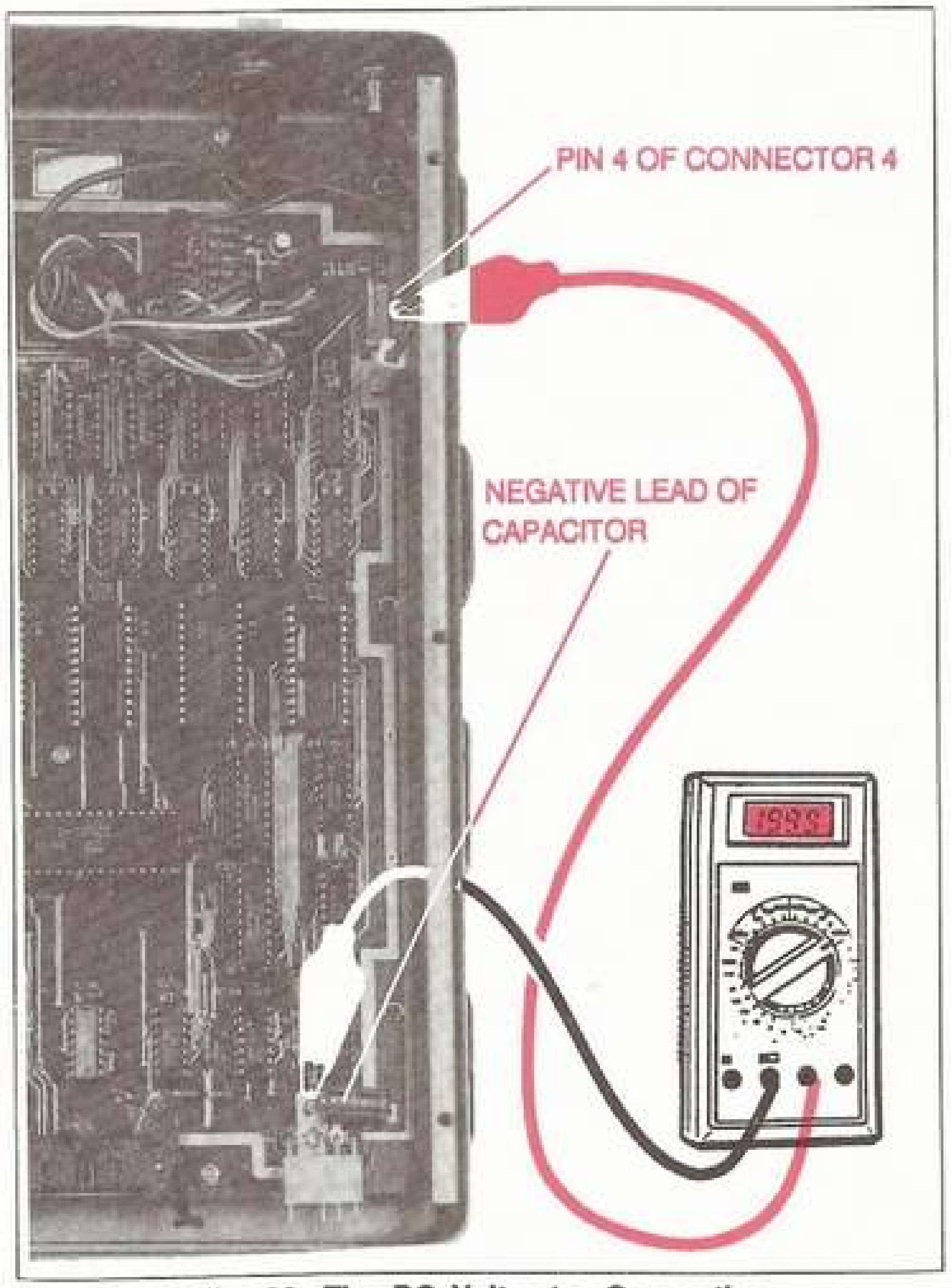


Fig. 89: The DC Voltmeter Connection

____m. What happens when Statement 60 appears on the screen?

____

____

*Note: The meter is connected to **DB0 (Pin 4)** of **Connector 4**. As Statement 40 is executed, **DB0** goes **HIGH** and the meter shows a reading of near **5-V dc** for five seconds. When Statement 60 is executed, **DB0** goes **LOW** and the meter changes to a reading of **0-V dc.***

____n. Change line 40 of the program to read:

```
40 PRINT "Q2":LPRINT "Q2"
```

____m. Unclip the **RED** voltmeter lead from the wire plugged onto **Pin 4** of **Connector 4**, and clip it to the wire plugged onto **Pin 3**, instead.

____n. Press the Pro-Arm's **RESET** button and then press **F2** to run the program again.

____o. Does the meter show the same results as in **Steps** l and **m**?

____

It should.

*Note: Since a binary two **(0010)** is being sent to the output port when Statement 40 is executed, only **DB1 (Pin 3)** goes **HIGH**.*

____p. Try running the program, as it is, several more times, checking the other three wires (**Pins 1, 2, and 4**) for an output. *Note: **Pins 5 and 6** are **ground**.*

____q. Do any other wires except the one plugged to **Pin 3** indicate an output?

____

____r. What would happen if you changed Statement 40 to read as shown below?

```
40 PRINT "Q15":LPRINT "Q15"
```

____

____

____

____s. How many pins would be outputting voltage after Statement 40 was executed?

____

____t. Change Statement 40 to read, as shown above, and check your answer to **Step s.**

____u. Were you correct?

____

If not, review the **OTHER INTERFACING FEATURES** discussed earlier in this section.

____v. Turn off the Pro-Arm's power supply and remove the **BASIC** or **DOS** System Disk, placing it in its protective sleeve.

____w. Turn off the computer system and the voltmeter.

____x. Unplug the wired 6-pin plug connector from **Connector 4** and disconnect the voltmeter leads.

____y. Replace the Pro-Arm's base plate and secure it firmly with the six machine screws. Carefully set the Pro-Arm upright with its power and interface connections intact.

____z. This completes Lab Exploration 40, "4-Bit Out." Please have your instructor initial your Knowledge Transfer Guide.

The following exploration will demonstrate how the Pro-Arm's input switches can be tested. See the **SPECIFICATIONS** heading in Section VI for the Input-Switch Pin Assignments.

LAB EXPLORATION 41

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

To check the test results of this exploration, be sure to refer to the **ANSWERS** section in the back of this Knowledge Transfer Manual.

____41. INPUTS OUT

Purpose: This exploration will provide the student with a simple test procedure for determining the working status of the Pro-Arm's input switches.

Equipment: IBM-compatible Computer System, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, IBM-type Parallel Interface Cable, Knowledge Transfer Manual, 6-Pin Connector (wired), Phillips Screwdriver, Voltmeter, with clip leads, and 4 Switch Jumpers.

____a. With the power cord and the interface cable still connected to the Pro-Arm, carefully set the Pro-Arm on its back, as shown in Fig. 87. *Caution: **DO NOT** pull or crimp the power or interface cables. They must continue to operate during the remainder of this exploration.*

____b. Use the Phillips screwdriver to remove the six machine screws holding the baseplate to the body. Next, remove the baseplate, so that the Pro-Arm's PCB is open to view.

____c. Locate **Connector 4** on the PCB (see Fig. 88) and insert the 6-pin connector plug (wired) onto it, making sure that none of the six wires are touching each other.

____d. Connect the dc voltmeter as shown in Fig. 89. The **BLACK** lead is clipped to the **NEGATIVE** lead of the power input capacitor, **C0**. The **RED** lead is clipped to the wire plugged to **Pin 4 (DB0)** of **Connector 4**.

*Caution: Make sure none of the other wires at **Connector 4** are touching the **RED** voltmeter lead.*

___e. Turn the voltmeter on, and set it for a voltage reading of **5-V dc**.

___f. Boot the computer system and apply power to the Pro-Arm.

___g. At the prompt **(A>_)** remove the **DOS** disk from floppy drive **A** and return it to its protective sleeve. Insert the disk containing the **BASIC** software (included with the computer in your classroom). *Note: If you are using an IBM system, leave the system disk in the floppy drive for the next step.*

___h. Boot up your **BASIC** software by typing in **GWBASIC** and pressing **ENTER**. *Note: If you are using an IBM system, type in either **BASIC** or **BASIC**, and press **ENTER**.*

___i. At the **BASIC** prompt, type in the following six program lines. Remember to press **ENTER** after each line is completed.

```
20 PRINT "Z":LPRINT "Z"
30 PRINT "D5":LPRINT "D5"
40 PRINT "U1,1,1,1":LPRINT "U1,1,1,1"
50 PRINT "D5":LPRINT "D5"
60 PRINT "U0,0,0,0":LPRINT "U0,0,0,0"
70 PRINT "END":END
```

___j. Press the Pro-Arm's **RESET** button to be sure that the data from the computer will get through.

___k. Press **F2** and watch the voltmeter for a reading. *Note: No movement of the arm should be taking place, but each program line should appear on the screen as it is executed.*

___l. Does a reading appear on the voltmeter as Statement 40 is executed?

____

Is this the expected response?

____

Which configuration is being used for the input switches by Statements 40 and 60?

____

___m. Refer to Table 11, and check to see if it agrees with the reading from **Step l**. Does it?

____

INPUT SWITCH				COMMAND PARAMETER STRING	OUTPUT PORT			
CON7	CON8	CON9	CON10		DB3	DB2	DB1	DB0
H	H	H	H	U0,0,0,0	L	L	L	L
H	H	H	H	U0,0,0,1	L	L	L	H
H	H	H	H	U0,1,0,0	L	H	L	L
H	H	H	H	U1,1,1,1	H	H	H	H
L	L	L	L	U0,0,0,0	L	L	L	L
L	L	L	L	U1,0,0,0	L	L	L	L
L	L	L	L	U0,1,1,0	L	L	L	L
L	L	L	L	U1,1,1,1	L	L	L	L
L	H	H	H	U0,0,0,0	L	L	L	L
L	H	H	H	U1,0,0,0	L	L	L	L
H	H	L	H	U0,1,0,1	L	H	L	H
L	H	L	H	U1,1,1,1	L	H	L	H

TABLE 11: Truth Table Examples—Normally Open Switches

*Note: Since the "U" command is being used in Statements 40 and 60, the normally open configuration is being used for the input switches in this program. Since the parameter string of Statement 40 indicates that all four switches are enabled, then the output level at **Pin 4** of **Connector 4** should be the same as the input level at **Connector 10**. We are only able to check one level at a time with the voltmeter, and right now, it is connected to output port **DB0**.*

____n. Place a jumper over the switch at **Connector 10**. Refer to Fig. 90 for its location.

____o. Press the Pro-Arm's **RESET** button, and press **F2** to rerun the program. Observe the voltmeter. Is the indication any different?

____

Why?

____

____

*Note: Placing the jumper on **Connector 10** brought the input level on that switch **LOW**. Since the output follows the input on an enabled switch being operated in the normally open configuration, the expected output at port **DB0** would be **LOW**.*

____p. Rewrite Statements 40 and 60 in the program as follows:

```
40 PRINT "V1,1,1,1":LPRINT "V1,1,1,1"
60 PRINT "V0,0,0,0":LPRINT "V0,0,0,0"
```

____q. With the jumper still on input switch **Connector 10**, and the voltmeter still clipped to output port **DB0**, press the Pro-Arm's **RESET** button and press **F2** to run the edited program.

____r. What does the voltmeter show?

____

Is this the expected output?

____

INPUT SWITCH				COMMAND PARAMETER STRING	OUTPUT PORT			
CON7	CON8	CON9	CON10		DB3	DB2	DB1	DB0
H	H	H	H	V0,0,0,0	L	L	L	L
H	H	H	H	V0,0,0,1	L	L	L	L
H	H	H	H	V0,1,0,0	L	L	L	L
H	H	H	H	V1,1,1,1	L	L	L	L
L	L	L	L	V0,0,0,0	L	L	L	L
L	L	L	L	V1,0,0,0	H	L	L	L
L	L	L	L	V0,1,1,0	L	H	H	L
L	L	L	L	V1,1,1,1	H	H	H	H
L	H	H	H	V0,0,0,0	L	L	L	L
L	H	H	H	V1,0,0,0	H	L	L	L
H	H	L	H	V0,1,0,1	L	L	L	L
L	H	L	H	V1,1,1,1	H	L	H	L

TABLE 12: Truth Table Examples—Normally Closed Switches

Which configuration is now being used for the input switches by Statements 40 and 60?

____

____s. Do the results in **Step** r agree with the information in Table 12?

____

Why?

____

____

*Note: Keeping the jumper on **Connector 10** kept the input level on that switch **LOW**. Since the output is opposite of the input on an enabled switch being operated in the normally closed configuration, the expected output at port **DB0** would be **HIGH**.*

____t. You may experiment with various switch settings and command parameters, using both configurations, at this time. Take readings at each of the four output ports, if desired, to confirm your understanding. Check your results against the Truth Table Examples in Tables 11 and 12.

____u. Turn off the Pro-Arm's power supply and remove the **BASIC** or **DOS** System Disk, placing it in its protective sleeve.

____v. Turn off the computer system and the voltmeter.

____w. Unplug the wired 6-pin plug connector from **Connector 4** and disconnect the voltmeter leads.

____x. Remove any of the switch jumpers still in place on **Connectors 7 through 10**.

____y. Replace the Pro-Arm's base plate and secure it firmly with the six machine screws. Carefully set the Pro-Arm upright with its power and interface connections intact.

____z. This completes Lab Exploration 41, "Inputs Out," and the PROGRAMMING INTERFACE FEATURES activities. Have your instructor initial your Knowledge Transfer Guide.

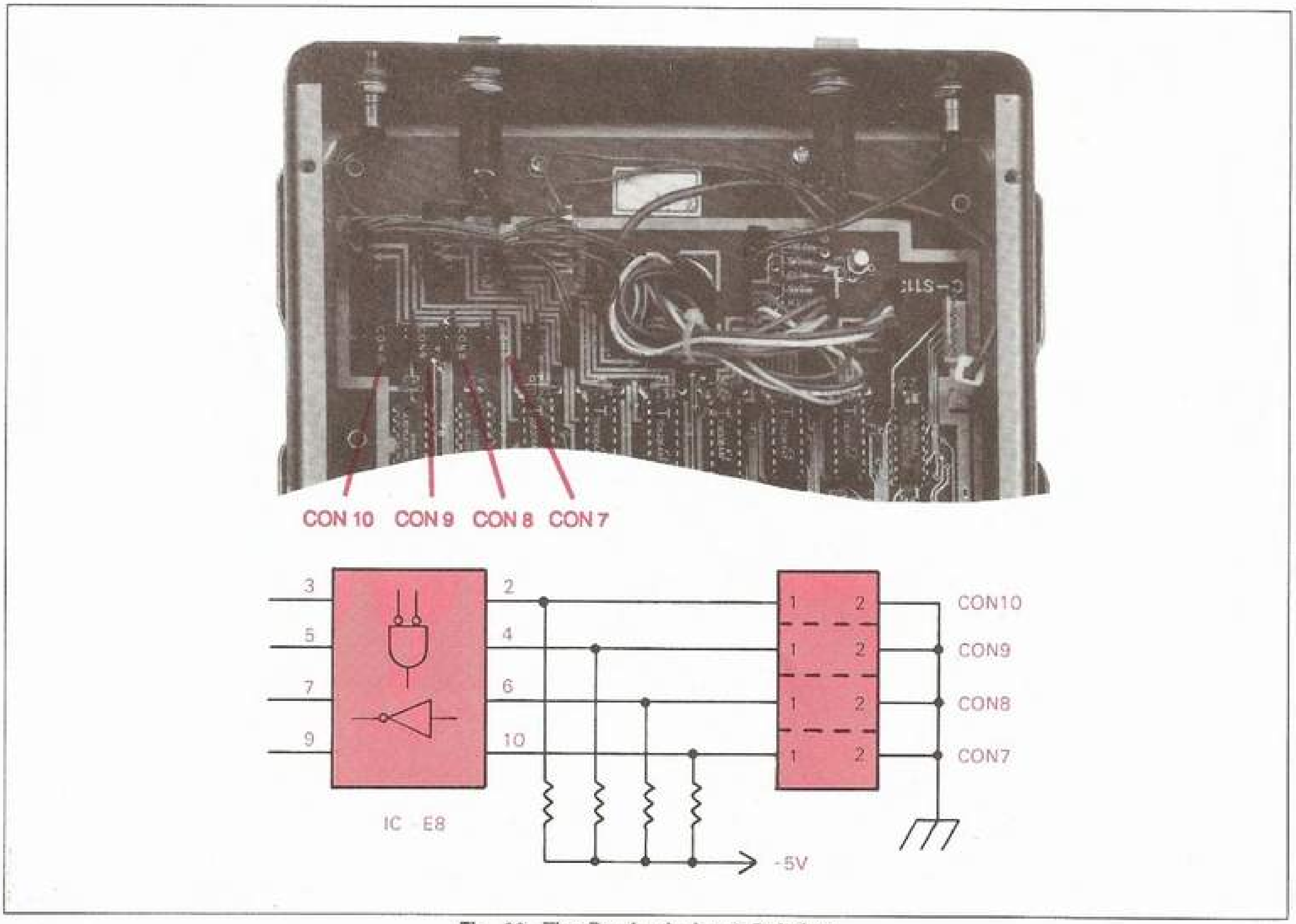


Fig. 90: The Pro-Arm's Input Switches

YOUR FIRST BASIC PROGRAM

DISCUSSION

Here is your turn to write a ***BASIC*** program for the MARCRAFT RS-2200 Pro-Arm. The following exploration is merely a series of blanks, waiting for you to fill them in.

LAB EXPLORATION 42

In the appropriate space provided, write the word or phrase which best fits the meaning of each sentence. As you complete each of the following steps, place a check mark (√) in the space provided to assure that all steps are performed.

____42. YOUR PROGRAM

Purpose: This exploration will provide the student with the opportunity to write an original ***BASIC*** program for the Pro-Arm.

Equipment: IBM-compatible Computer System, Pro-Arm Robot, Pro-Arm Power Supply and Power Cord, IBM-type Parallel Interface Cable, Knowledge Transfer Manual, Grid Sheet, Experimental Blocks, and a Pencil.

____a. Determine at least four specific things the Pro-Arm will do in your program and write them below.

(1) ____________________

(2) ____________________

(3) ____________________

(4) ____________________

____b. On the other side of this page, write your program in the space provided. If more space is needed, use the extra blank pages at the end of this section..

____c. Check the system installation and interfacing to be sure that it is ready to go.

____d. Boot the computer system and apply power to the Pro-Arm.

____e. At the prompt **(A>_)**, remove the **DOS** disk from floppy drive **A** and return it to its protective sleeve. Insert the disk containing the **BASIC** software (included with the computer in your classroom).

Note: If you are using an IBM system, leave the system disk in the floppy drive for the next step.

____f. Boot your **BASIC** software up by typing in **GWBASIC** and pressing **ENTER**.

*Note: If you are using an IBM system, type in either **BASIC** or **BASICA**, and press **ENTER**.*

____g. Use the **TEST** and **RESET** buttons on the Pro-Arm's front panel to place it in its **"ZERO"** position.

____h. At the **BASIC** prompt, type in your **BASIC** program from the space above. Don't forget to press **ENTER** at the end of each line to placethe line in memory.

____i. Momentarily press the Pro-Arm's **RESET** button.

____j. Press **F2.** Observe how closely your program follows the requirements you wrote in **Step a.** Does the Pro-Arm perform as indicated?

____

____k. If the answer in **Step j** is **YES**, then go on to the next step. If the answer is **NO**, then edit your program and begin again at **Step i**. Do not quit until your program performs correctly.

____l. Remove the **BASIC** System Disk and return it to its protective sleeve. Insert the Pro-Arm Program Disk in the drive and close the drive door.

____m. Create a name for your program (using eight letters or less) and write it here.

____

*Note: **BASIC** will add the **.BAS** extension for you automatically when you save your mprogram to disk.*

____n. Press **F4** and type the name from **Step m.**

____o. Press **ENTER** to save your program to the Pro-Arm Program Disk.

____p. Turn off the Pro-Arm's power supply and remove the Pro-Arm Program Disk, placing it in its protective sleeve.

____q. Turn off the computer system, but leave the installation and interface connections as they are.

____r. This completes Lab Exploration 42, "Your Program," and YOUR FIRST BASIC PROGRAM activities. This also completes SECTION IV of your Knowledge Transfer Project. Have your instructor initial your Knowledge Transfer Guide.

Notes:

Notes:

Notes:

SECTION V

KNOWLEDGE TRANSFER REVIEW

This Knowledge Transfer Review will test your understanding of the material covered in the Discussions and Explorations of the previous Sections. When you have completed the 25 questions, have your instructor grade this exam and enter your score in the Knowledge Transfer Guide.

Multiple Choice: Complete the following statements by circling the correct letter.

1. Discharge built-up body static prior to handling an IC by:
   a. Touching anyone within reach.
   b. Connecting a grounding strap around your wrist.
   c. Wearing rubber gloves.
   d. Grabbing the soldering iron.

2. When writing on labels already attached to a disk:
   a. Be sure to press firmly with lead pencils.
   b. Only use ball-point pens.
   c. Chisel the information in.
   d. Only use felt-tip pens.

3. While the Pro-Arm is in motion:
   a. Do not block its movement with any object or your body.
   b. Do not keep liquids away from its circuits.
   c. Do not prepare adequate space for its working envelope.
   d. Do not worry about exceeding its operating parameters.

4. To prepare the Pro-Arm to receive a new set of commands and to clear its RAM:
   a. Press the Pro-Arm's TEST button.
   b. Press ENTER.
   c. Press the Pro-Arm's RESET button.
   d. Press F2.

5. The Pro-Arm's SHOULDER AXIS is used:
   a. To rotate the body.
   b. To move the upper arm.
   c. To house the Pro-Arm's PCB.
   d. To move the lower arm.

6. The GRIPPER is used for:
   a. Holding or manipulating objects.
   b. Directing a large orchestra.
   c. Opening locks or combination safes.
   d. Pouring water, milk or orange juice.

7. The microprocessor, memory chips, PCBs, and the power supply are all housed in the:
   a. Monitor.
   b. Keyboard.
   c. Pro-Arm.
   d. System Unit.

8. If power has already been applied to the computer, we refer to the boot-up procedure as:
   a. A cold boot.
   b. A hot boot.
   c. A warm boot.
   d. A new boot.

9. The first self-test contained in the Pro-Arm's ROM memory:
   a. Is used to test the Pro-Arm's operation at three different speeds.
   b. Is used to align each axis to its "ZERO" position.
   c. Is used to test the Pro-Arm's ability to repeat the placement of an object accurately.
   d. Is used to demonstrate the Pro-Arm's vertical envelope.

10. In low-technology robots:
    a. Each axis can assume only two positions.
    b. The cycle time is slower than any other class of robot.
    c. Maximum payload varies between 30 to 150 lbs.
    d. End effectors manipulate heavy payloads at slow speed.

11. Another name for a medium-technology robot is:
    a. A pick-and-place device.
    b. A universal device.
    c. A limited-sequence device.
    d. A limited-application device.

12. In high-technology robots:
    a. Payloads can vary in weight from 2.2 ounces to 1 pound.
    b. Some have built-in diagnostic routines.
    c. There are two to four nonservo-type degrees of freedom.
    d. Low-cost pneumatic actuation normally is used.

13. The data bus used by the Zilog Z80 microprocessor is:
    a. 4 bits wide.
    b. 6 bits wide.
    c. 8 bits wide.
    d. 16 bits wide.

14. An 8-bit chunk of data is called a:
a. Bit.
b. Nibble.
c. Byte.
d. Word.

15. The limit switch is used:
a. To prevent the gripper from turning past its closed position.
b. To provide a reliable zero-reference point for the gripper.
c. To prevent damage to the gripper motor and other components.
d. All of the above.

16. ASCII stands for:
a. The American Society of Computer Information Ignorance.
b. The American Standard Code for Information Interchange.
c. The American Symposium for Communication Interface Independence.
d. The All-purpose Symbolic Code of Immediate Interfacing.

17. The BUSY interface signal originates from the:
a. Computer, and tells the Pro-Arm to get busy.
b. Computer, and tells the Pro-Arm to wait.
c. Pro-Arm, and tells the computer to wait.
d. Pro-Arm, and tells the computer to get busy.

18. The Pro-Arm's input switches allow external devices to:
a. Interrupt the Pro-Arm's execution process.
b. Send signals to other external devices.
c. Read data from specific Pro-Arm addresses.
d. Run the self-test programs.

19. All BASIC command lines must begin with:
a. A line number.
b. A PRINT command.
c. A quotation mark.
d. A capital letter.

20. The BASIC command to move the Pro-Arm is:
a. The "Z" command.
b. The "W" command.
c. The "V" command.
d. The "M" command.

21. The D(elay) command will halt the Pro-Arm's movement:
a. For a maximum of 25 seconds.
b. For a maximum of 15 seconds.
c. For a maximum of 99 seconds.
d. For any amount of time desired.

22. The horizontal work envelope is determined by the arm's:
a. Pitch and Roll.
b. Reach and Swing.
c. Tilt and Sway.
d. Yaw and Tilt.

23. The Pro-Arm's vertical work envelope is shaped like a:
a. Tear.
b. Triangle.
c. Cylinder.
d. First-class letter.

24. The Pro-Arm will halt all movement if:
a. The "W" command is executed during a closed interlock condition.
b. Any liquid is detected in the work envelope.
c. The "N" command is executed at any time.
d. None of the above.

25. Since the command "V1,1,0,0" has just been executed:
a. Output port DB0 will be HIGH, when CON10 is jumpered.
b. Output port DB1 will be HIGH, when CON9 is open.
c. Output port DB2 will be LOW, when CON8 is open.
d. Output port DB3 will be LOW, when CON7 is jumpered.

SECTION VI

TROUBLESHOOTING AND SPECIFICATIONS

TROUBLESHOOTING

DISCUSSION

This section provides additional technical information for those occasions when repair or adjustment of the Pro-Arm is required, or when factory service is the logical course of action.

Your Pro-Arm robot has been engineered to provide long and trouble-free service. Consequently, your Pro-Arm should not require lubrication. However, it may be necessary from time to time to make a chain adjustment, calibrate the gripper cutoff switch, change an IC, or troubleshoot some other malfunction. The Service Flowcharts on Page 150 should help you in locating any problem with your Pro-Arm. For chain adjustments, refer to the Adjustment Procedures on Page 152. For cutoff-switch calibration, refer to the Calibration Procedure on Page 153.

Refer to Fig. 92 for an illustration of the Pro-Arm's mechanical drives and to Page 13 for the schematic diagram.

If, upon initial use, your Pro-Arm fails to operate from your computer, it is usually due to a cabling problem between the computer and the Pro-Arm. Be sure you are using the proper cable configuration, and that your computer is either an IBM PC, PC/XT, or PC AT or a compatible model. Refer to the **PRO-ARM INSTALLATION** portion of Section II.

For replacement motors and individual PC board components, see the **TECHNICAL ASSISTANCE AND FACTORY SERVICE** heading in this section. In the event that there is a malfunction on the PC board other than an IC, you may choose to replace the entire board. In this case, refer to the **REPLACEMENT and EXCHANGE PARTS** heading.

Caution: Follow the Safety Precautions listed in Section I while testing and troubleshooting the Pro-Arm.

TOOL REQUIREMENTS

The following hand tools may be required when attempting to repair the Pro-Arm.

1. Screwdriver, Phillips-head (medium).
2. Lineman's pliers.
3. Crimper, barrel.
4. Wire stripper.
5. Soldering pencil, 20- to 30-watt range, with small pointed tip.
6. Retaining-ring pliers.
7. IC insertion/extraction tools.

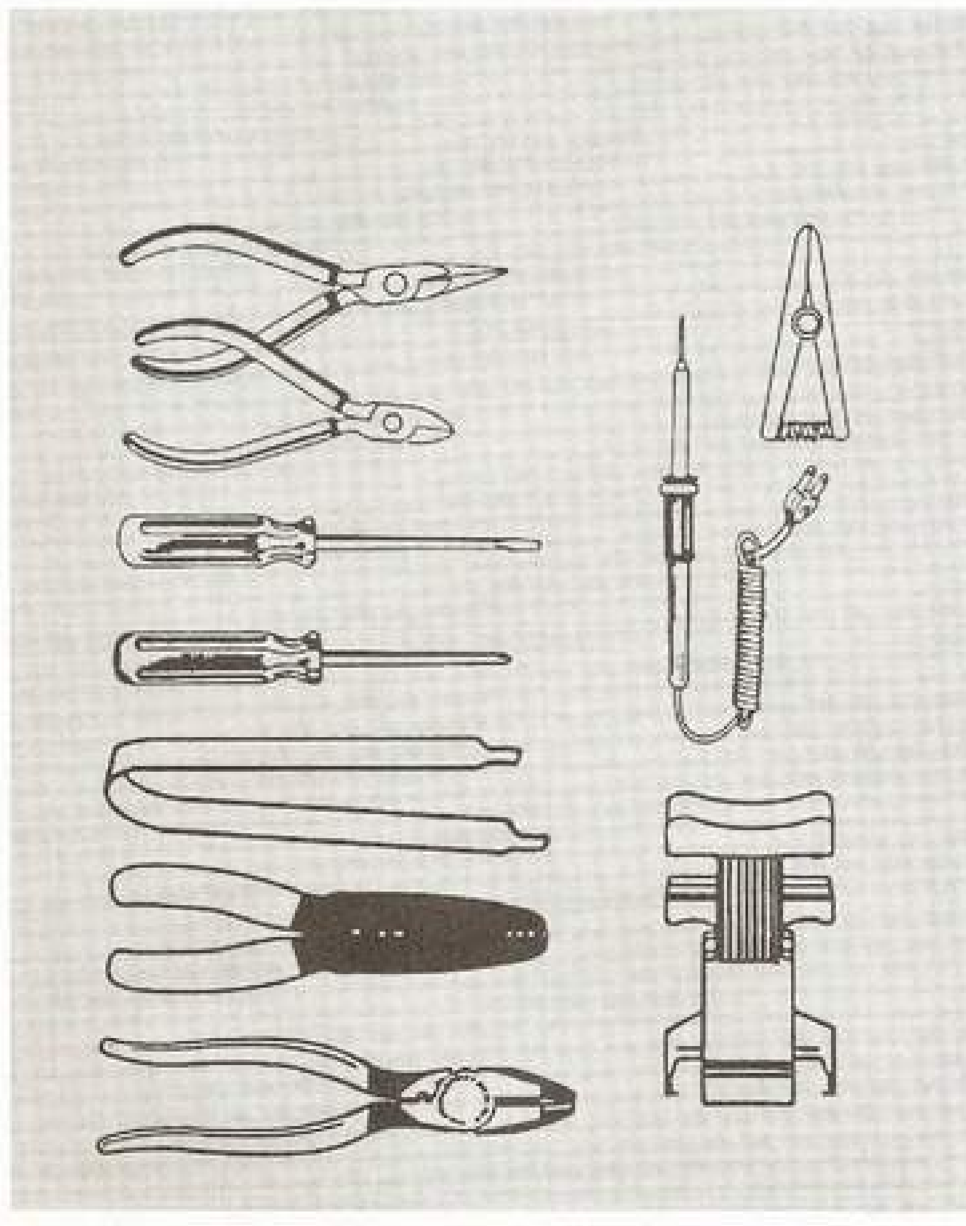


Fig. 91: Troubleshooting Tools Required

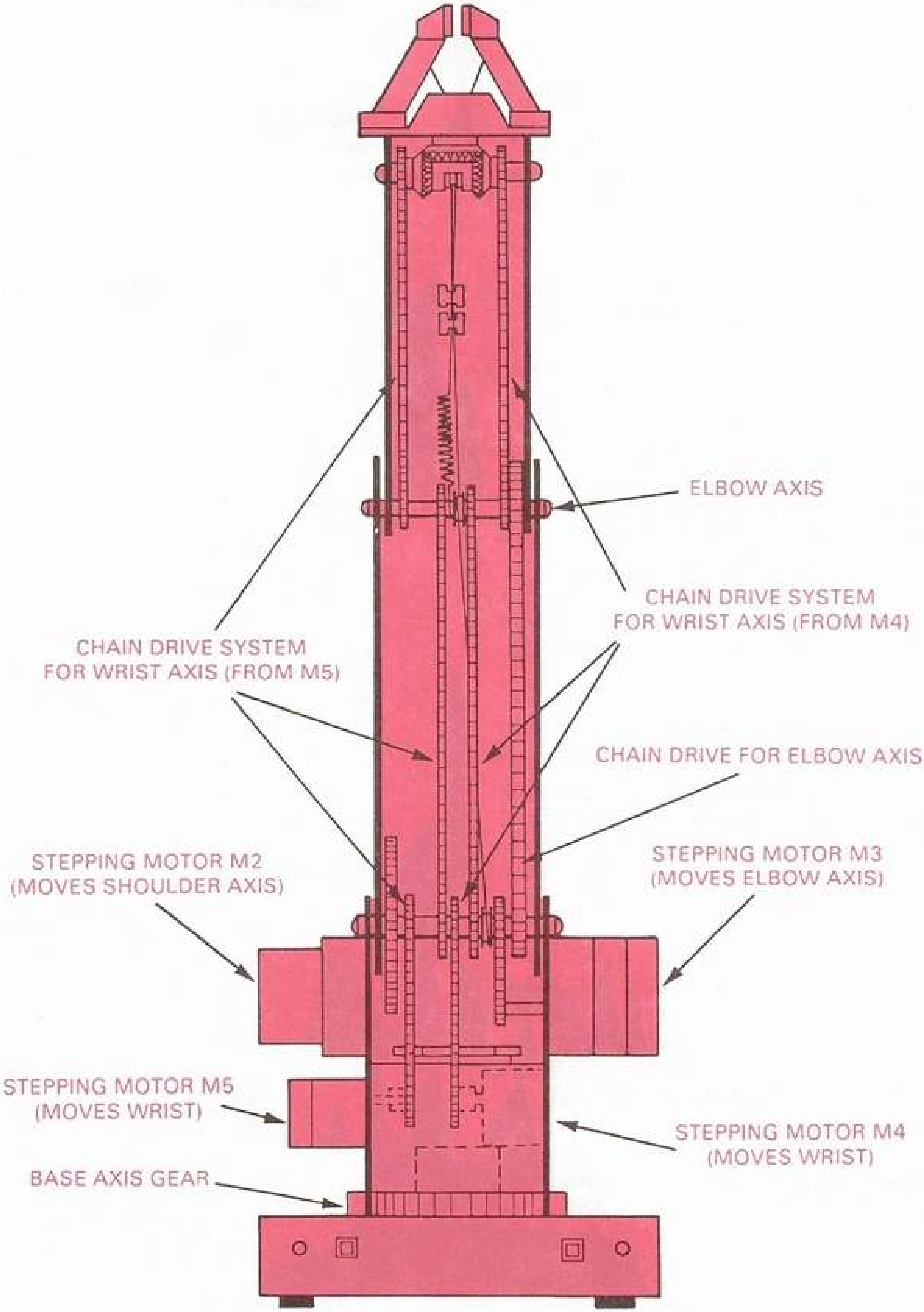


Fig. 92: The Mechanical Drives

IC HANDLING INSTRUCTIONS

WARNING: Follow these instructions and understand them thoroughly before removing and replacing any ICs from the Pro-Arm, in order to avoid possible damage to them.

The ICs used in the Pro-Arm may be static-sensitive devices that can be destroyed by electrostatic discharges during handling or insertion. In order to minimize the chances of this happening, carefully follow these instructions whenever you are engaged in removing or replacing ICs in the Pro-Arm's PCB. As you read each precaution, place a check mark (√) in the space provided to assure that each one has been reviewed and understood.

___1. **Do not unpack ICs** until you are ready to replace a bad one. Keep them in their protective container until you are ready to use them. Replace only one IC at a time.

___2. **Do not touch** the IC's pins. At no time should the pins of the IC ever be touched. Hold the IC by the ends of its package.

___3. **Minimize handling** of the IC. Handle the ICs as little as possible.

___4. **Keep foreign objects away** from the ICs. Avoid allowing tools, metal objects, plastic bags, styrofoam, and other foreign matter from coming into contact with the pins of the IC.

___5. **Do not slide the ICs** over any surfaces. This action will generate static buildup.

___6. **Discharge built-up body static** prior to handling an IC. Connect a grounding strap around your wrist prior to handling or inserting an IC.

___7. **Use an IC insertion tool** if there is one available. This will be your best protection. Follow the manufacturer's instructions for usage of the tool.

___8. **Use care** once the IC has been inserted. Avoid allowing tools and other objects to touch the foil side of the printed circuit (PC) board.

___9. **Use extreme caution** when soldering on or around circuits containing ICs or other semiconductor devices. These devices are extremely sensitive to excessive temperature conditions.

Be sure you thoroughly understand these rules before touching or replacing any ICs in the Pro-Arm.

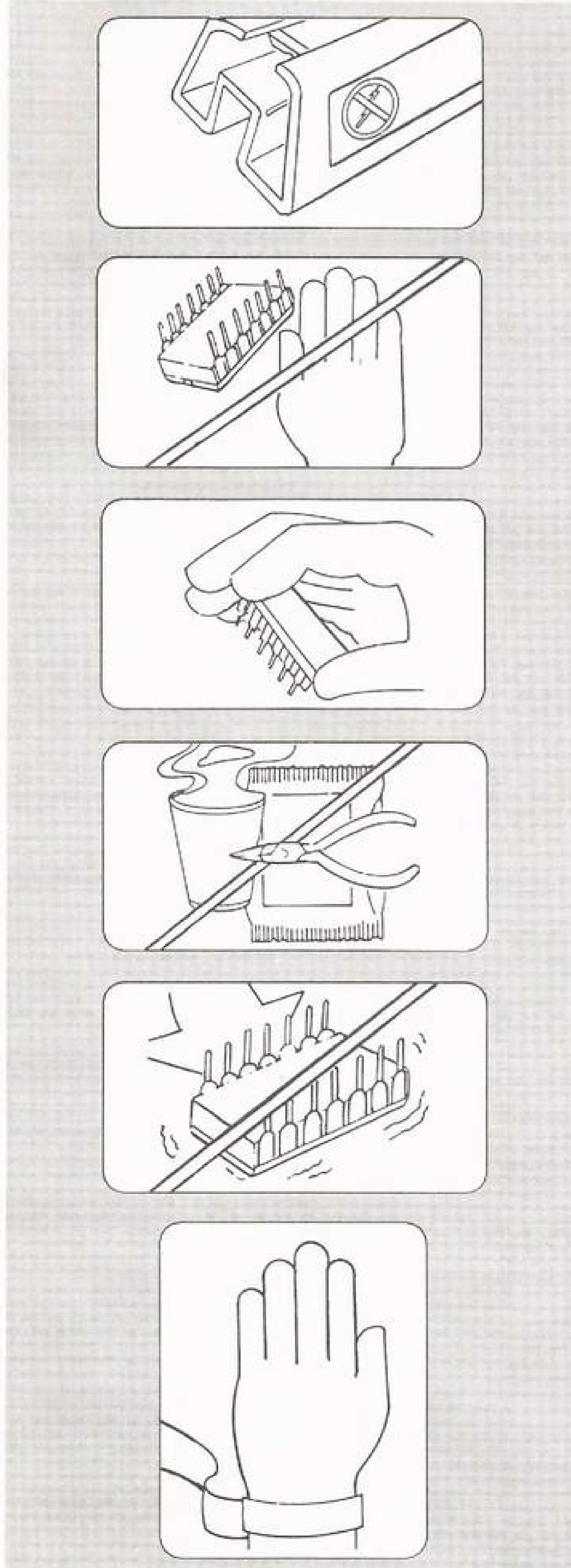
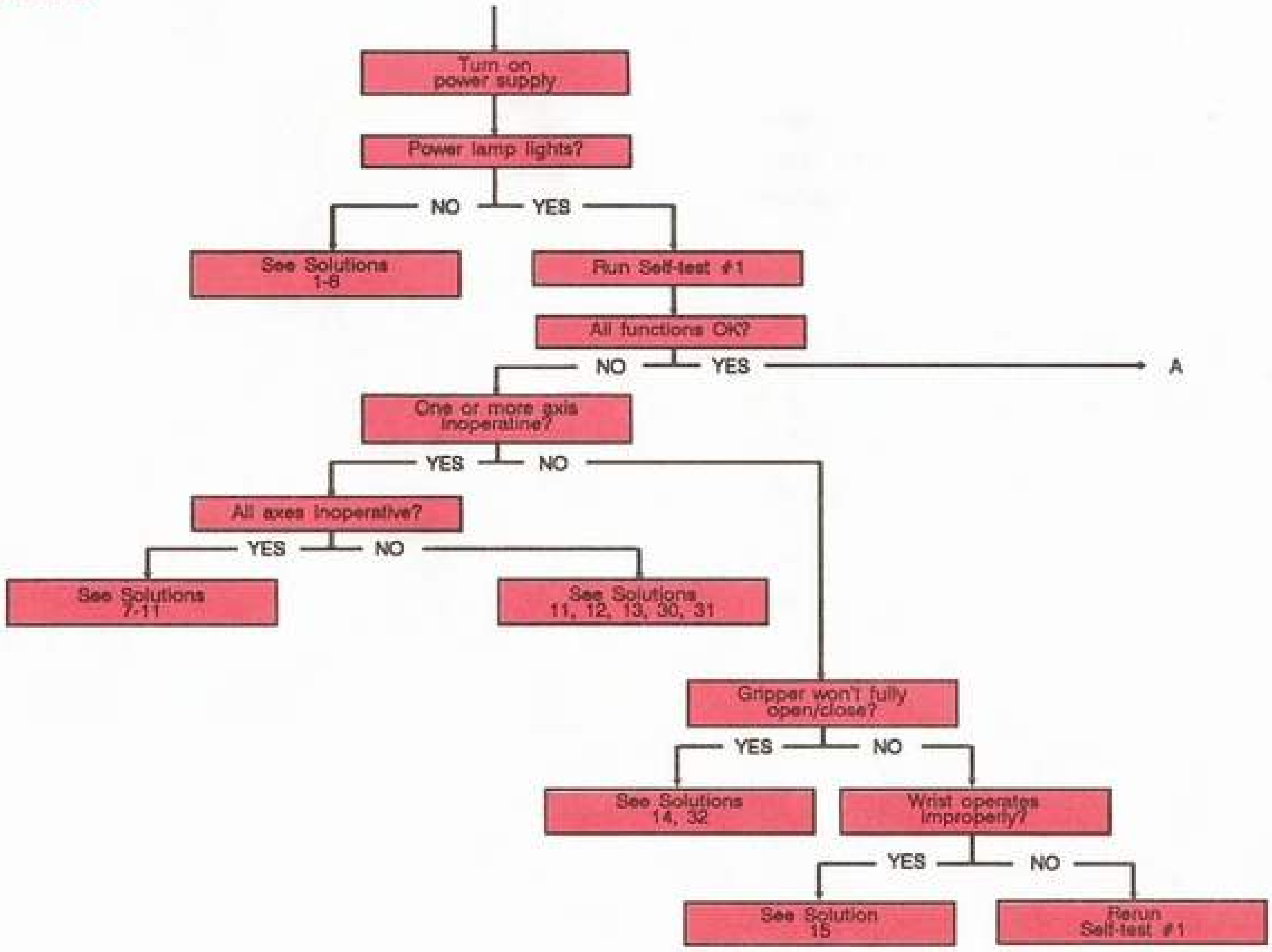
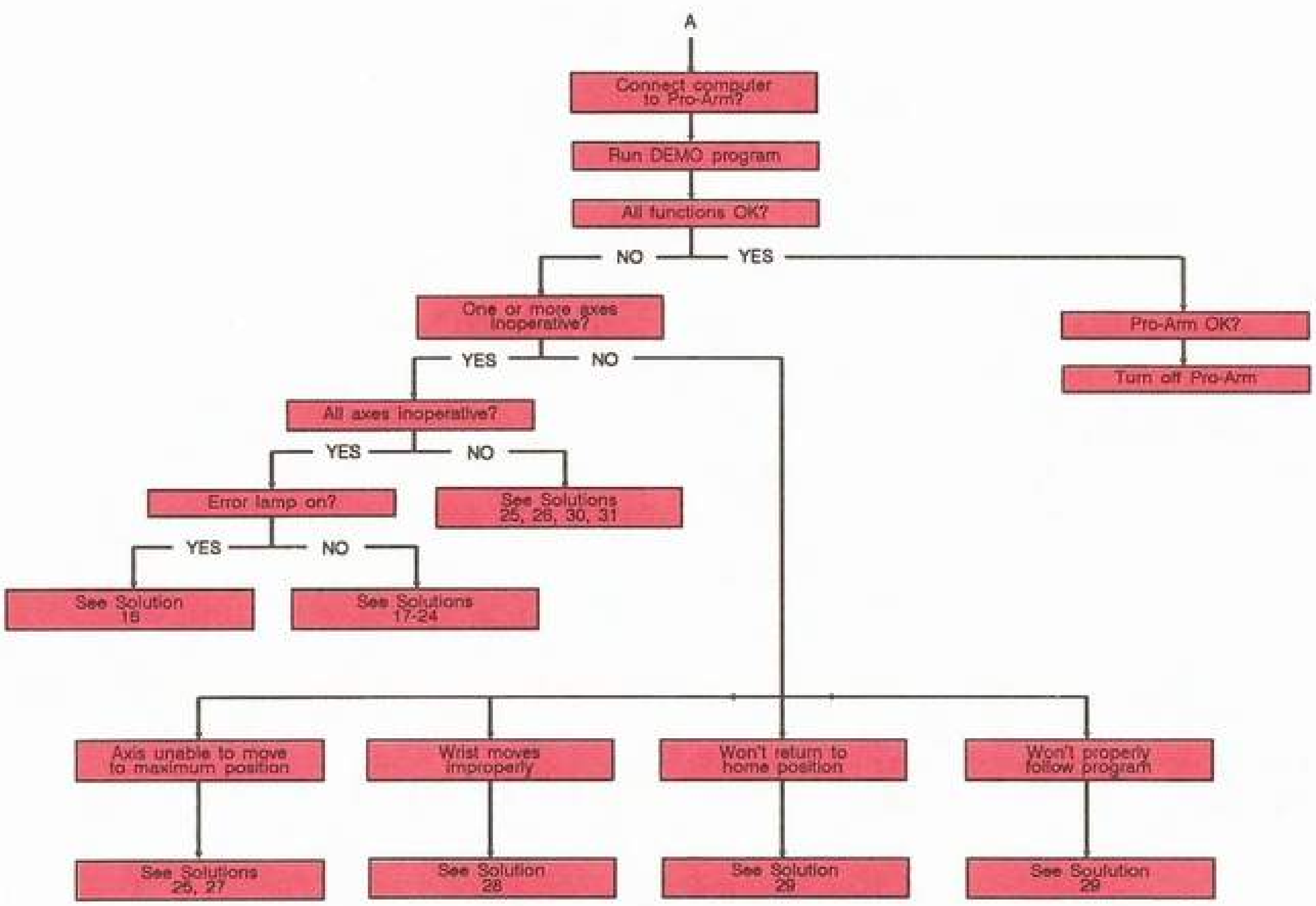


Fig. 93: Proper IC Handling Techniques

SERVICE FLOWCHARTS





SOLUTIONS

1. Power supply not plugged into wall receptacle. Confirm.

2. Improper voltages at wall receptacle. Check voltage value.

3. Voltage selector switch on power supply set improperly. Confirm setting.

4. Power supply not connected to Pro-Arm. Confirm connection.

5. Power supply improperly connected to Pro-Arm. Confirm proper connection.

6. Power supply defective. Replace.

7. Test improperly initiated. Rerun test.

8. Gripper won't fully close. Check limit-switch operation and calibrate if necessary.

9. No voltage from power supply. Check Pro-Arm's power lamp.

10. **RESET** or **TEST** switch is inoperative. Check continuity.

11. Circuit malfunction. Replace PC board.

12. Mechanical-linkage problem on inoperative axis. Check for broken, loose, or slipped chains, excessively tight chains, or loose gears. Check for foreign matter between chains, pulleys, and gears.

13. Motor driver IC blown. Replace with new IC.

14. Improperly set limit switch. Perform calibration procedure.

15. Mechanical linkage problem on wrist axes. Check for excessively tight chains, broken, loose, or slipped chains, or loose gears. Check for foreign matter between chains and gears.

16. Improper command sent. Check the programming instructions.

17. Pro-Arm is in a **"busy state."** Depress **RESET** button.

18. Computer improperly connected to the Pro-Arm. Refer to installation instructions.

19. Computer's printer port not operating properly, or not designated correctly by software. Check buffer ICs and program.

20. Data commands not reaching the Pro-Arm's microprocessor. Replace IC **A6**.

21. Interface cable malfunction. Check or replace cable.

22. Incorrect pin configuration (if using computer other than an IBM-type). Check pin configuration (refer to Table 1 or Table 15).

23. Computer malfunction. Check, using different unit.

24. Circuit malfunction on PC board. Replace PC board.

25. Mechanical linkage problem on inoperative axis. Check for excessively tight chains, broken, loose, or slipped chains, or loose gears. Check for foreign matter between chains and gears.

26. Motor driver IC blown. Replace with new IC.

27. Zero-position of axis improperly set. Reposition as needed.

28. Mechanical-linkage problem on wrist axes. Check for broken, loose, or slipped chains, excessively tight chains, or loose gears. Check for foreign matter between chains and gears.

29. **Z**(ero) command not sent at beginning of program. Add command to program.

30. Electrical continuity problem on inoperative axis. Check for loose or damaged wires from plug connector.

31. Switch motor plug with adjacent connector and check operation with self-test #1. If bad axis now runs, check driver circuit.

32. Gripper limit-switch information not being transferred. Replace IC **E8**.

ADJUSTMENT PROCEDURES

If an axis chain has excessive slack, it will cause the axis to move erratically. If this should occur, adjust the chain tension using the procedure described below. *Caution: Do not overly tighten the chains or the axes will bind and become inoperative.*

Refer to Fig. 92 for the chain reference numbers.

CHAIN 41 or 51

____a. Loosen the two small Phillips-head screws located on the side of the elbow, as shown in Fig. 94.

____b. Using a large screwdriver, push the jaw sprocket assembly upwards until the chain is taut.

____c. Maintaining the tension achieved in **Step b**, retighten the small screws.

____d. Release the tension on the jaw sprocket assembly. *Caution: It is easy to make this adjustment too tight. Release some of the tension if it appears to be more than desired.*

CHAIN 31, 42, or 52

____a. Loosen the large screws located on both sides of the elbow-shoulder joint. Refer to Fig. 94.

____b. Pull the elbow upwards until all three chains are taut, or until as much slack has been removed from each chain as possible.

____c. While maintaining the tension achieved in **Step b**, retighten the screws.

____d. If chains 42 or 52 are still not taut, loosen the nuts on the two adjustment screws located at the top of the shoulder (see Fig. 94). Next, adjust the screws until the chains are taut, and then retighten the nuts.

____e. If chain 31 is still not taut, loosen the large adjuster nut located on the right side of the shoulder (see Fig. 94). Next, tighten the chain by moving the adjuster sprocket, and then retighten the nut.

CHAIN 43

____a. Loosen the four screws securing motor **M4**.

____b. Push the motor back, towards the rear of the Pro-Arm, until the chain is taut.

____c. Retighten the four screws.

CHAIN 53

____a. Loosen the four screws securing motor **M5**.

____b. Push the motor back, toward the rear of the Pro-Arm, until the chain is taut.

____c. Retighten the four screws.

CHAIN 32

____a. Loosen the four screws securing motor **M3**.

____b. Push the motor back, toward the rear of the robot, until the chain is taut.

____c. Retighten the four screws.

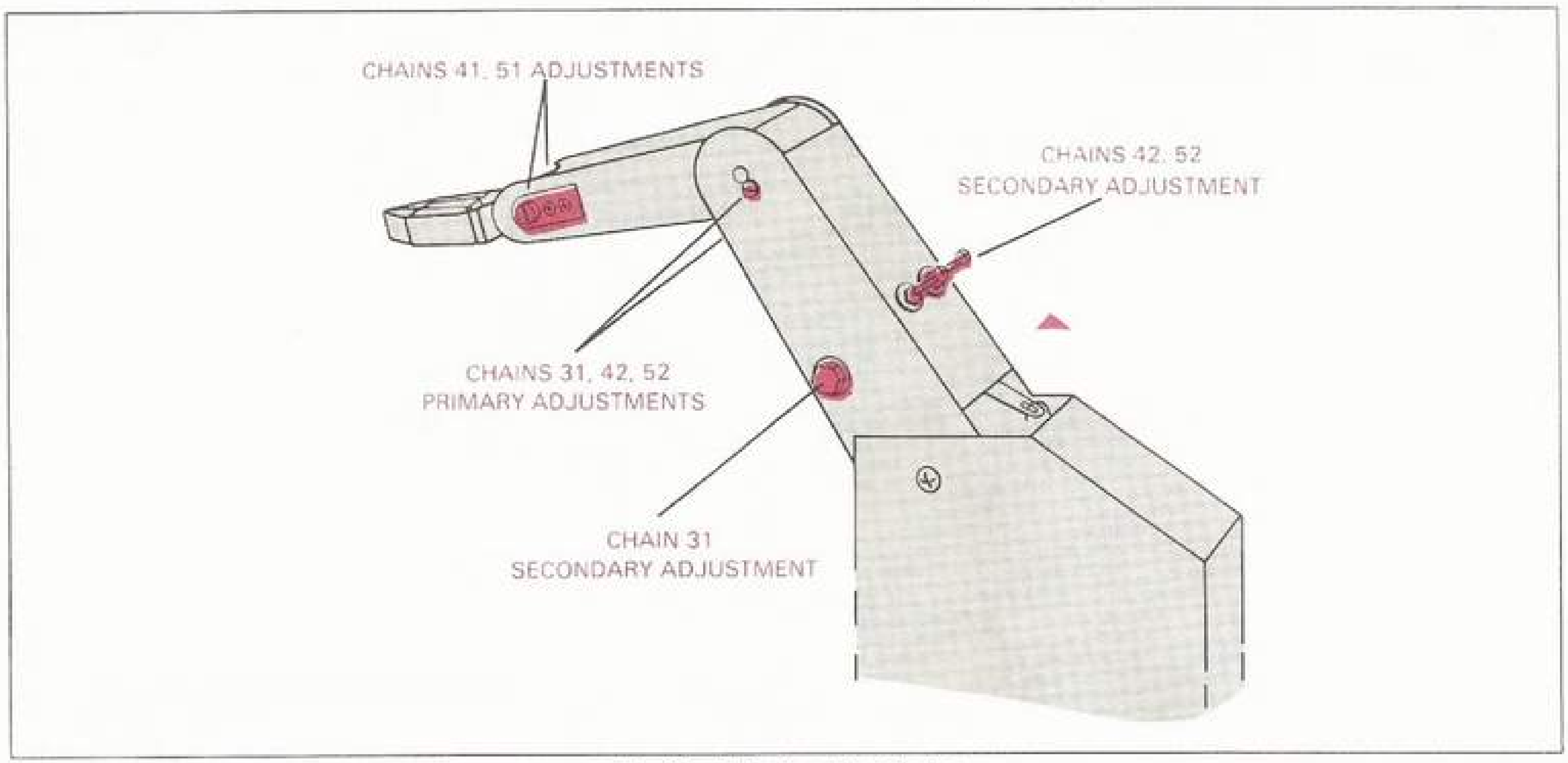


Fig. 94: Adjusting the Chains

CALIBRATION PROCEDURE

This procedure should be performed whenever it is suspected that the gripper limit *(cutoff)* switch is not precisely set. The gripper cable must activate this switch before any commands will be accepted by the Pro-Arm.

____a. Using the Phillips-head screwdriver, remove the six screws (three per side) holding the Pro-Arm's **body backplate** (containing the unit's serial number), as shown in Fig. 95.

____b. Set the **body backplate** aside and install the Pro-Arm system for the operation of self-test #1.

____c. Observe the gripper cable and pulley arrangement, and the gripper limit switch, as shown in Fig. 96. Check to see if the cable is binding between the flywheel and the gripper motor **M6**, or if it has slipped off any of its pulleys.

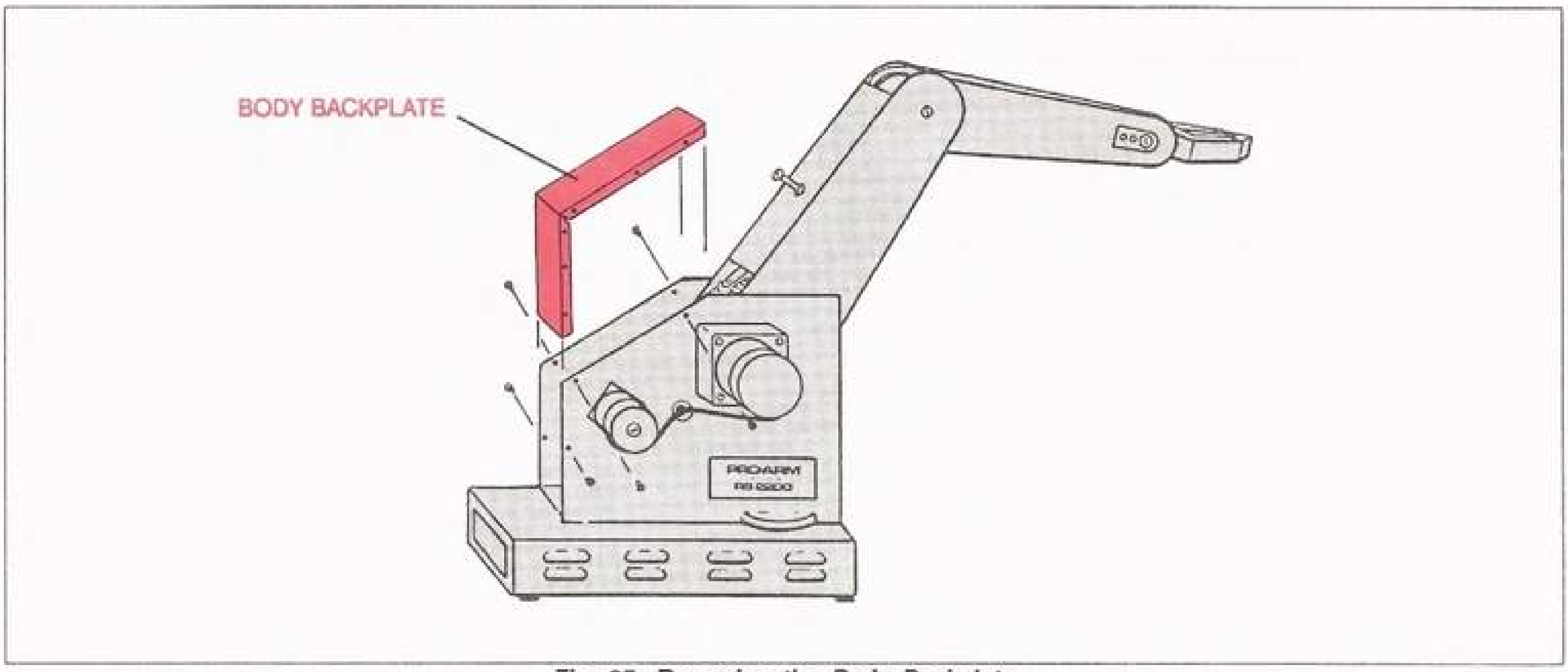


Fig. 95: Removing the Body Backplate

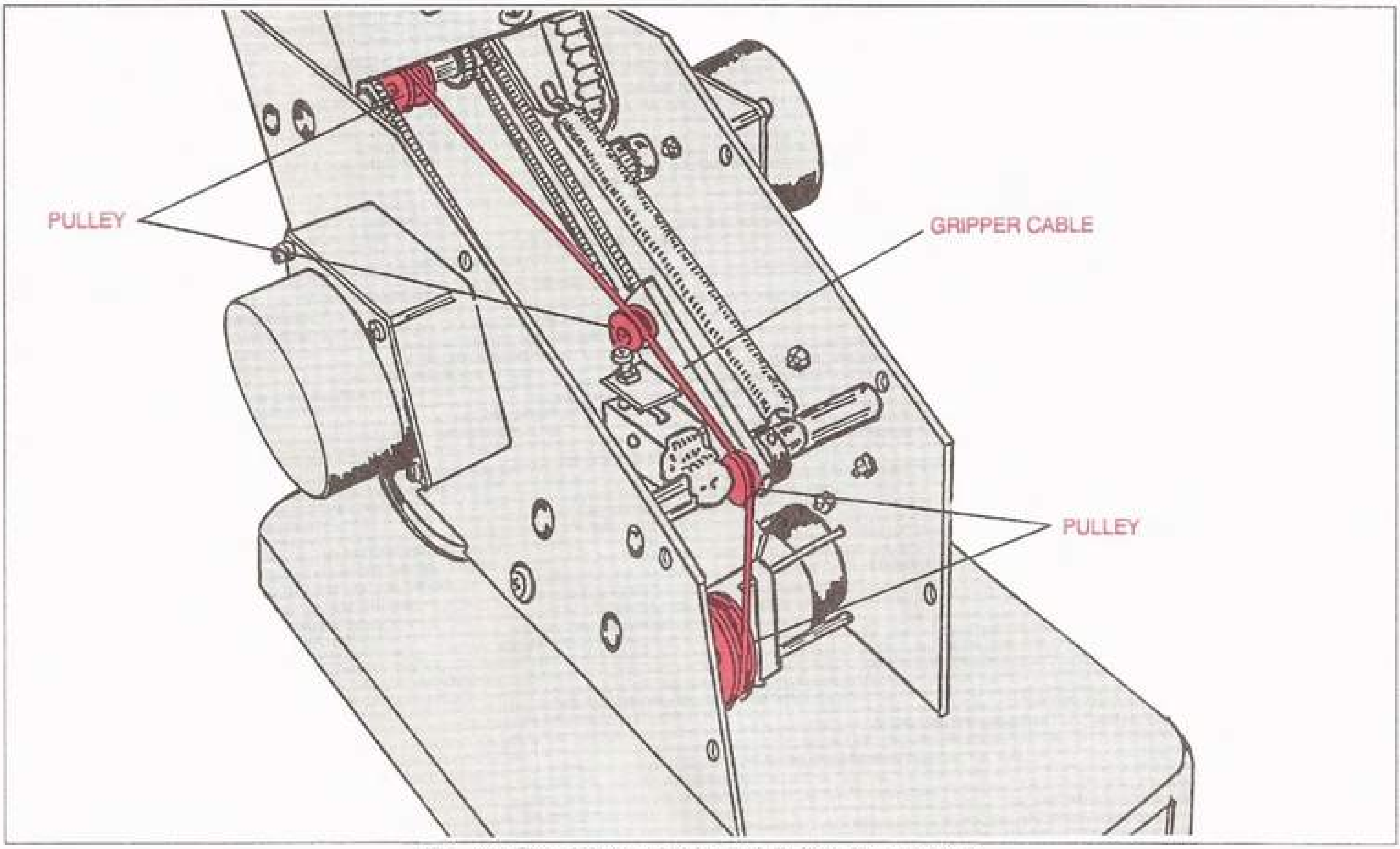


Fig. 96: The Gripper Cable and Pulley Arrangement

____d. Examine the gripper pulley and cable arrangement for excessive tightness or slackness. Check to be sure that the gear on the gripper motor is tight **(not movable without motor action)**, and that the motor shaft itself is not bent.

____e. Make a note of any abnormal conditions observed in **Steps c** and **d**, but take no action yet. *Note: In most cases, the first course of action to take is to attempt to* ***OPEN*** *the gripper, using the zero-positioning self-test program. Whatever the problem is, it's best to first observe that problem in action.*

____f. Apply power to the Pro-Arm. If the gripper or its cutoff switch is **OPEN**, the gripper motor will run momentarily. When the cable tension becomes great enough (as the gripper closes), the switch will click closed and the gripper motor will stop running. *Caution: If the gripper motor continues running, even though the gripper is closed, manually press the gripper cutoff switch to the* ***OFF*** *position.* You will hear the click of the switch as the motor stops running. Once the gripper motor stops running, it is all right to stop pressing the cutoff switch.

____g. Start the self-test procedure, and at the eleventh press of the **TEST** button, hold it down long enough to **OPEN** the gripper as wide as possible, before the gripper motor stops running. *Note: The gripper motor is attached to a timer circuit which will cut it off after a certain period.* At this point, the gripper cable should have enough slack in it to allow reorienting any misguided cable path.

____h. The conditions noted in **Step c** and **d** should have been verified by careful observation while the gripper was opening. Now is the time to make sure that the cable rigging is correct, and that the motor gear is tightly fastened to the motor shaft. *Note: If the motor shaft is bent, you have a* ***PROBLEM!*** *The shaft must be carefully straightened, or the motor itself will have to be replaced.*

____i. Press the **RESET** button and carefully observe the switch action as the gripper closes. If the gripper motor stops before the gripper closes, go to **Step p**. *Note: The ideal situation is to hear the cutoff switch click just as soon as the gripper closes (not before and not too long afterwards). Usually, whenever any problems occur, it's because the switch doesn't activate soon enough. This causes the cable to tighten excessively and strains the gripper motor along with its associated circuitry.*

____j. If the gripper motor continues to run after the gripper has closed, manually click the cutoff switch. Loosen the nut at the base of the cutoff-switch adjustment screw, as depicted in Fig. 97. *Note: You may have to break the lock-seal to do it.*

____k. Using a small Phillips-head screwdriver, tighten the cutoff-switch adjustment screw so that it protrudes lower into the arm of the gripper cutoff switch. *Note: This should cause the switch to cutoff sooner.*

____l. Tighten the nut at the base of the cutoff switch adjustment screw. *Note:* ***Do not*** *use any lock-seal at this time. This adjustment must be tested first.*

____m. Run the self-test again, opening the gripper widely on the eleventh press of the **TEST** button.

____n. Press the **RESET** button and check for the proper operation of the cutoff switch as described in **Step l**.

____o. Repeat **Steps j** through **n** until the switch operation is satisfactory. Check it several times and then go to **Step v**.

____p. Loosen the nut at the base of the cutoff-switch adjustment screw, as shown in Fig. 97. *Note: You may have to break the lock-seal to do it.*

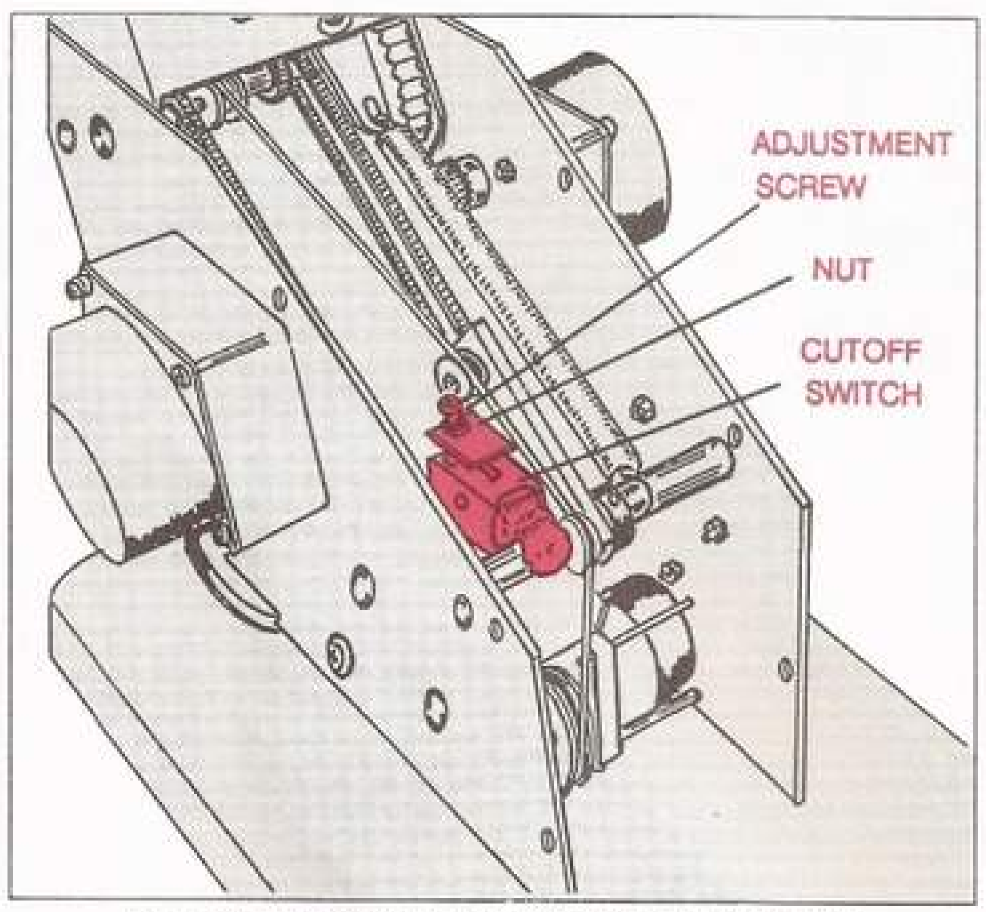


Fig. 97: The Gripper Cutoff-Switch Adjustment

____q. Using a small Phillips-head screwdriver, loosen the cutoff-switch adjustment screw so that it doesn't protrude as far into the arm of the switch itself. *Note: This should cause the switch to cutoff later.*

____r. Tighten the nut at the base of the cutoff-switch adjustment screw. *Note: **Do not** use any lock-seal at this time. This adjustment must be tested first.*

____s. Run the self-test again, opening the gripper widely on the eleventh press of the **TEST** button.

____t. Press the **RESET** button and check for the proper operation of the cutoff switch as described in **Step l**.

____u. Repeat **Steps p** through **t** until the switch operation is satisfactory. Check it several times.

____v. Remove power from the Pro-Arm.

____w. Apply some lock-seal to the nut at the base of the cutoff adjustment screw, to prevent the Pro-Arm's operational vibrations from changing the screw setting.

____x. Using the Phillips-head screwdriver, replace the Pro-Arm's body backplate using the six screws (three per side) you removed in **Step a**.

SPECIFICATIONS

GENERAL

Application:	Education, research
Positioning Accuracy:	+/–0.9mm (0.035")
Load Capacity:	500g (1.1 lb)
Reach:	445mm (17.5")
Gripper Opening:	100mm (4")
Cylinder Gripper Opening:	180mm (7")
Gripping Force:	1.4 kg (3 lbs or 13n) maximum
Movement Speed:	5 software-controlled speeds, 75–200mm/sec (3–8"/sec)
Weight (body):	8 kg (17.6 lbs)
Degrees of freedom:	5 axes + gripper
Operational Range	
Base:	240°
Shoulder:	140°
Elbow:	100°
Wrist Roll:	360°
Wrist Pitch:	180°
Drive System:	6 stepping motors, open loop control
Speed Control:	Square wave
Controller:	Z80 Microprocessor with 4K bytes of EPROM and 1K bytes of RAM.
Memory Capacity (RAM):	100 positions (2K bytes)
Memory Expansion (RAM):	270 positions total (4K bytes)
Interface:	8-bit Centronics parallel port
External Device Ports:	One 4-bit parallel output, one 8-bit parallel output, one interlock switch, four input switches (all software controllable).
Program Languages:	All
Power Supply	
Input:	105–125-V ac, 200–264-V ac (switchable), 50/60 Hz, 60 W max;
Output:	5-V dc @ 1.2A, 12-V dc @ 6A.

AXES MOVEMENTS

AXIS	DEG/STEP	DIRECTION	MOTOR	+/-	MAX STEPS	EXAMPLE
Base	0.12	cw	M1	+	1000	M1000,0,0,0,0,0
		ccw		-	1000	M −1000,0,0,0,0,0
Shoulder	0.12	up	M2	+	600	M0,600,0,0,0,0
		down		-	600	M0, −600,0,0,0,0
Elbow	0.10	up	M3	+	500	M0,0,500,0,0,0
		down		-	500	M0,0 − 500,0,0,0
Wrist Roll	0.10	cw	M4	+	7200	M0,0,0,7200,7200,0
			M5	+		
		ccw	M4	-	7200	M0,0,0, −7200, −7200,0
			M5	-		
Wrist Pitch	0.10	down	M4	+	900	M0,0,0,900, −900,0
			M5	-		
		up	M4	-	900	M0,0,0, −900,900,0
			M5	+		
Gripper	0.10	close	M6	+	2000	M0,0,0,0,0,2000
		open		-	2000	M0,0,0,0,0,−2000

Fig. 98: Axes Movement Specifications

MOTOR SPEEDS

SPEED NO	DESCRIPTION	MOVEMENT SPEED (Approximate)
1	Very slow	2.5 inch/sec (64mm/sec)
2	Slow	4 inch/sec (102mm/sec)
3	Medium	5 inch/sec (127mm/sec)
4	Fast	7 inch/sec (178mm/sec)
5	Very fast	8 inch/sec (203mm/sec)

TABLE 13: The Five Pro-Arm Motor Speeds

Character		Binary Bit 7 to Bit 0	Octal	Decimal	Hexadecimal
NUL	(null)	00000000	000	000	00
SOH	(start of heading)	00000001	001	001	01
STX	(start text)	00000010	002	002	02
ETX	(end text)	00000011	003	003	03
EOT	(end transmission)	00000100	004	004	04
ENQ	(enquiry)	00000101	005	005	05
ACK	(acknowledge)	00000110	006	006	06
BEL	(beep)	00000111	007	007	07
BS	(backspace)	00001000	010	008	08
HT	(horizontal tab)	00001001	011	009	09
LF	(line feed)	00001010	012	010	0A
VT	(vertical tab)	00001011	013	011	0B
FF	(form feed)	00001100	014	012	0C
CR	(carriage return)	00001101	015	013	0D
SO	(shift out)	00001110	016	014	0E
SI	(shift in)	00001111	017	015	0F
DLE	(data link escape)	00010000	020	016	10
DC1	(data control 1)	00010001	021	017	11
DC2	(data control 2)	00010010	022	018	12
DC3	(data control 3)	00010011	023	019	13
DC4	(data control 4)	00010100	024	020	14
NAK	(not acknowledge)	00010101	025	021	15
SYN	(synchronous idle)	00010110	026	022	16
ETB	(end transmit block)	00010111	027	023	17
CAN	(cancel)	00011000	030	024	18
EM	(end of medium)	00011001	031	025	19
SUB	(substitute)	00011010	032	026	1A
ESC	(escape)	00011011	033	027	1B
FS	(form separator)	00011100	034	028	1C
GS	(group separator)	00011101	035	029	1D
RS	(record separator)	00011110	036	030	1E
US	(unit separator)	00011111	037	031	1F
SP	(space)	00100000	040	032	20
!		00100001	041	033	21
"		00100010	042	034	22
#		00100011	043	035	23
$		00100100	044	036	24
%		00100101	045	037	25
&		00100110	046	038	26
'		00100111	047	039	27
(		00101000	050	040	28
)		00101001	051	041	29
*		00101010	052	042	2A
+		00101011	053	043	2B
,		00101100	054	044	2C
-		00101101	055	045	2D
.		00101110	056	046	2E
/		00101111	057	047	2F

Character	Binary Bit 7 to Bit 0	Octal	Decimal	Hexadecimal
0	00110000	060	048	30
1	00110001	061	049	31
2	00110010	062	050	32
3	00110011	063	051	33
4	00110100	064	052	34
5	00110101	065	053	35
6	00110110	066	054	36
7	00110111	067	055	37
8	00111000	070	056	38
9	00111001	071	057	39
:	00111010	072	058	3A
;	00111011	073	059	3B
<	00111100	074	060	3C
=	00111101	075	061	3D
>	00111110	076	062	3E
?	00111111	077	063	3F
@	01000000	100	064	40
A	01000001	101	065	41
B	01000010	102	066	42
C	01000011	103	067	43
D	01000100	104	068	44
E	01000101	105	069	45
F	01000110	106	070	46
G	01000111	107	071	47
H	01001000	110	072	48
I	01001001	111	073	49
J	01001010	112	074	4A
K	01001011	113	075	4B
L	01001100	114	076	4C
M	01001101	115	077	4D
N	01001110	116	078	4E
O	01001111	117	079	4F
P	01010000	120	080	50
Q	01010001	121	081	51
R	01010010	122	082	52
S	01010011	123	083	53
T	01010100	124	084	54
U	01010101	125	085	55
V	01010110	126	086	56
W	01010111	127	087	57
X	01011000	130	088	58
Y	01011001	131	089	59
Z	01011010	132	090	5A
[	01011011	133	091	5B
\	01011100	134	092	5C
]	01011101	135	093	5D
^	01011110	136	094	5E
_	01011111	137	095	5F

Character	Binary Bit 7 to Bit 0	Octal	Decimal	Hexadecimal
`	01100000	140	096	60
a	01100001	141	097	61
b	01100010	142	098	62
c	01100011	143	099	63
d	01100100	144	100	64
e	01100101	145	101	65
f	01100110	146	102	66
g	01100111	147	103	67
h	01101000	150	104	68
i	01101001	151	105	69
j	01101010	152	106	6A
k	01101011	153	107	6B
l	01101100	154	108	6C
m	01101101	155	109	6D
n	01101110	156	110	6E
o	01101111	157	111	6F
p	01110000	160	112	70
q	01110001	161	113	71
r	01110010	162	114	72
s	01110011	163	115	73
t	01110100	164	116	74
u	01110101	165	117	75
v	01110110	166	118	76
w	01110111	167	119	77
x	01111000	170	120	78
y	01111001	171	121	79
z	01111010	172	122	7A
{	01111011	173	123	7B
|	01111100	174	124	7C
}	01111101	175	125	7D
~	01111110	176	126	7E
DEL (delete)	01111111	177	127	7F

TABLE 14: The ASCII Code

*Note: To type **ASCII** characters not shown on the keyboard, hold the **ALT** key down while typing the decimal number (from the keypad) corresponding to the character desired. Release the **ALT** key and the character will appear at the cursor.*

There is an extended ***IBM ASCII*** character set which contains various line-draw characters and Greek symbols. These characters are not included in Table 14. The decimal reference numbers to this extended character set range from 128 to 255. By following the instructions above, these characters can also be typed.

MEMORY MAP

The **RAM** and **EPROM** memories may be accessed and dumped using the following addresses.

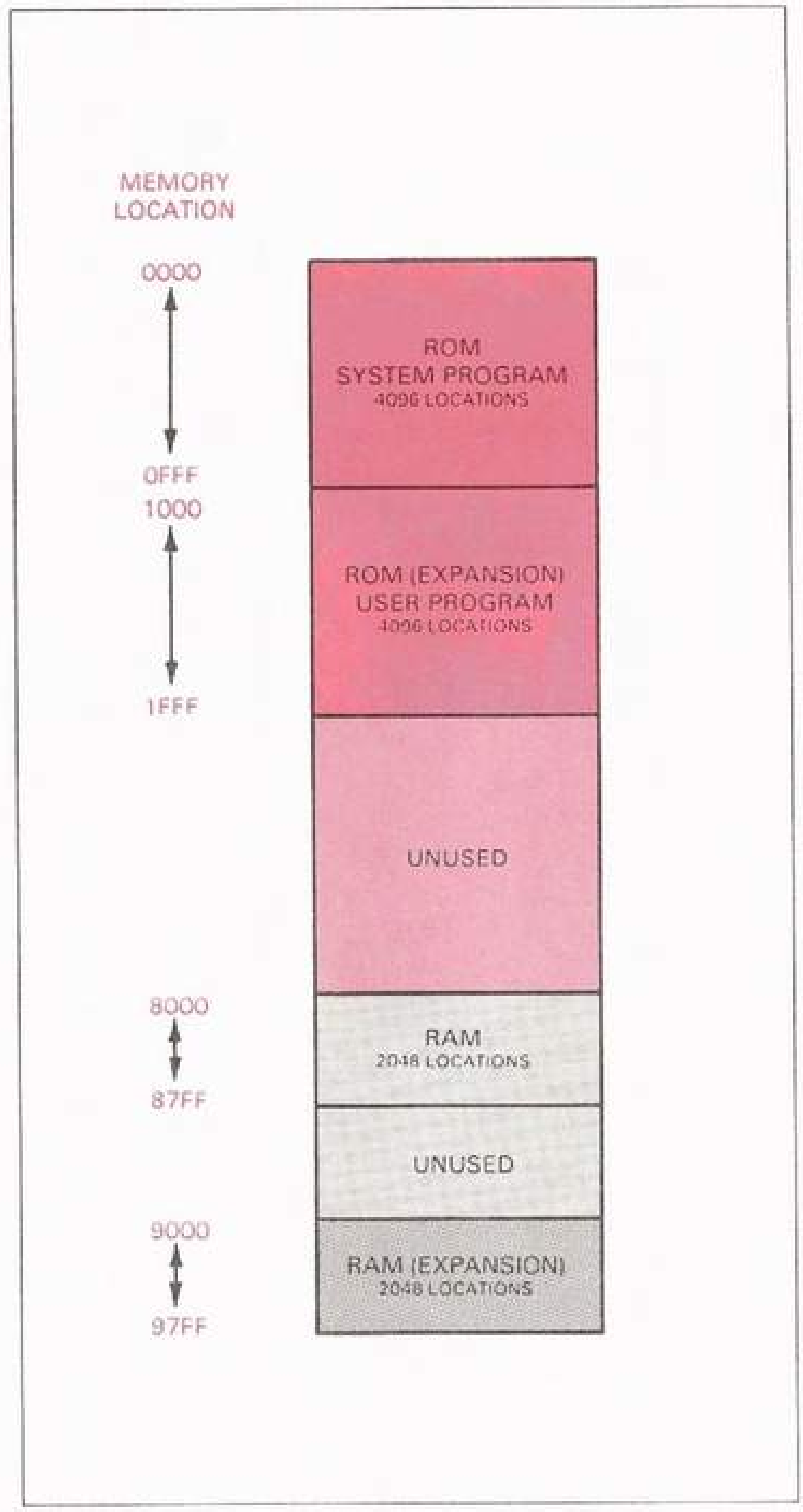


Fig. 99: RAM and ROM Memory Mapping

SIGNAL	ROBOT CONNECTOR (INPUT)	IBM PC TYPE CABLE (PORT OUTPUT)
STROBE	1	1
GROUND	2,4,6,8,10,12,14,16	19,20,21
DATA 0	3	2
DATA 1	5	3
DATA 2	7	4
DATA 3	9	5
DATA 4	11	6
DATA 5	13	7
DATA 6	15	8
DATA 7	17	9
ERROR	18	15
ACKNOWLEDGE	19	10
BUSY	20	11

TABLE 15: Pin Configuration of 8-Bit Port IBM-Type Cable

INPUT SWITCH	CONNECTOR 4 OUTPUT PORT BIT/PIN
CON7	DB3/1
CON8	DB2/2
CON9	DB1/3
CON10	DB0/4

TABLE 16: Input-Switch Pin Assignment

PIN NO.	OUTPUT SIGNAL
1	DB3
2	DB2
3	DB1
4	DB0
5	GND
6	GND

TABLE 17: The 4-Bit Output-Port Pin Assignment

PIN NO.	OUTPUT SIGNAL
1	DB0
2	DB1
3	DB2
4	DB3
5	DB4
6	DB5
7	DB6
8	DB7
9	GND
10	GND

TABLE 18: The 8-Bit Output-Port Pin Assignment

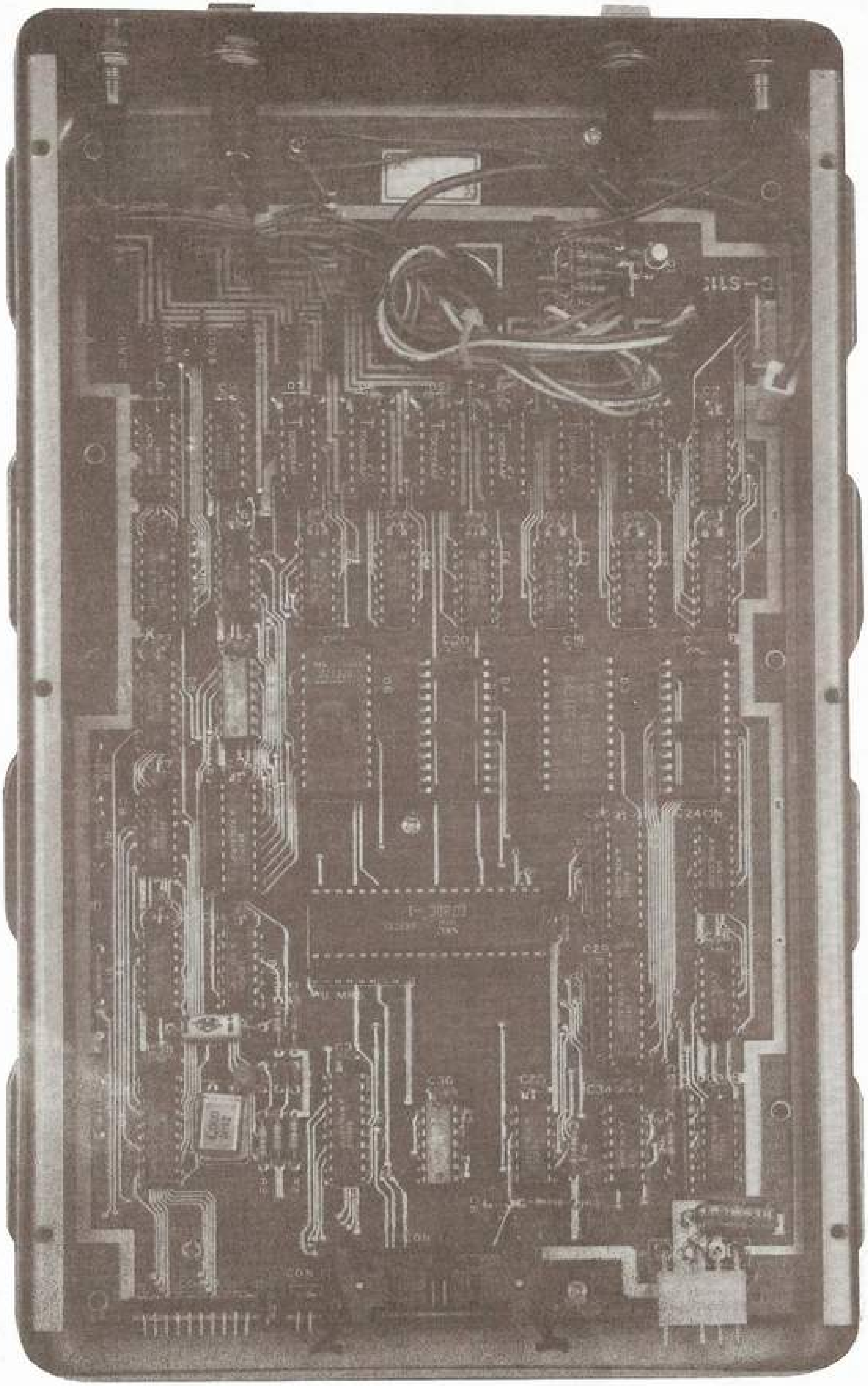


Fig. 100: The Pro-Arm's Printed-Circuit Board

TECHNICAL ASSISTANCE and FACTORY SERVICE

TECHNICAL ASSISTANCE

If the Pro-Arm malfunctions and you are unable to isolate the problem using the troubleshooting information in this manual, you may write the Factory Service consultant for technical assistance (see address below). There is no charge for this service. However, before you write, be sure to **(1)** check the Pro-Arm system's hardware thoroughly, and **(2)** try to isolate the source of the problem using the troubleshooting flowcharts.

When you write, please include the following information:

1. The serial number (from the body backplate).
2. The date of purchase and authorized dealer's name
3. The purchase invoice number.
4. A full description of the problem as well as you can describe it.
5. The sections of the manual you have successfully completed.
6. Your mailing address and telephone number.

You may also contact a MARCRAFT service technician by telephone. This service is also free of charge. Please have the above requested information available when you place the call.

FACTORY SERVICE

Prior to returning an inoperative Pro-Arm (or accessory) for repair, it is suggested that you first contact a MARCRAFT service technician. The technician may be able to determine the problem without the need for returning the Pro-Arm. Refer to the telephone number given below.

(509) 783-7359

MARCRAFT warrants that each product manufactured by it will be free from defects in material and workmanship under conditions of normal use and service for a period of ninety (**90**) days from date of purchase (motors are warranted for one (**1**) year) from an authorized MARCRAFT distributor.

This warranty extends only to the original purchaser.

This warranty shall not apply to any product or parts which have been subject to misuse, neglect, accident, or abnormal conditions of operation.

In the event of failure of a product covered by this warranty, MARCRAFT will repair a product returned to an authorized Service Facility within the warranty period, provided the warrantor's examination discloses to its satisfaction that the product was defective. The warrantor may, at its option, replace the product in lieu of repair.

With regard to any product returned within the warranty period, said repairs or replacement will be made without charge. If the failure has been caused by misuse, neglect, accident, or abnormal conditions of operation, repairs will be billed at a nominal cost. In such cases, an estimate will be submitted before work is started, if requested.

THE FOREGOING WARRANTY IS IN LIEU OF ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS, OR ADEQUACY FOR ANY PARTICULAR PURPOSE OR USE. MARCRAFT SHALL NOT BE LIABLE FOR ANY SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN CONTRACT, TORT, OR OTHERWISE. THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS, AND YOU MAY ALSO HAVE OTHER RIGHTS WHICH VARY FROM STATE TO STATE.

WARRANTY SERVICE

In the event of failure of a part of the system covered by the warranty, the manufacturer will replace the part returned to an authorized Service Facility within the warranty period, provided that the warrantor's examination discloses to their satisfaction that the part was defective. During this period, the warrantor may, at their option, replace the entire system in lieu of replacing a component. With regard to any part returned within the warranty period, said replacements will be made without charge. The motors will be replaced at no charge for up to one (**1**) year from the date of purchase. If the failure has been caused by misuse, neglect, accident, or abnormal conditions of operation, repairs will be billed at nominal cost. In such cases, an estimate will be submitted before work is started, if requested.

DO NOT RETURN YOUR PRO-ARM, OR ANY OF ITS COMPONENTS, WITHOUT PRIOR AUTHORIZATION FROM MARCRAFT.. This authorization will take the form of a Return Merchandise Authorization (**RMA**) number issued by an authorized MARCRAFT representative.

When returning products to MARCRAFT, write the **RMA** number clearly on the outside of the shipping carton, in a conspicuous place. If possible, ship the Pro-Arm in the original shipping carton, less cables and other accessories (unless requested). Ship prepaid and insured to the address shown below.

MARCRAFT International Corp.
100 N. Morain St.
Kennewick, WA 99336

OUT-OF-WARRANTY SERVICE

After the stated warranty period has expired, you may obtain reconditioned components (excluding motors) for your system through the MARCRAFT Pro-Arm Component Exchange Program for a nominal fee. The exchange fee is payable in advance. **DO NOT RETURN ANY COMPONENTS TO MARCRAFT WITHOUT PRIOR AUTHORIZATION FROM MARCRAFT OR AN AUTHORIZED MARCRAFT REPRESENTATIVE.**

REPLACEMENT and EXCHANGE PARTS

If an actual problem occurs with the Pro-Arm system, you may obtain replacement parts through the MARCRAFT Component Exchange Program. Isolate the source of the problem using the troubleshooting flow-charts in this manual, and localize the problem to a particular component.

If the problem occurs during the warranty period, the Exchange is free of charge. However, if the warranty period has expired, replacement parts can be purchased through MARCRAFT at a nominal fee. To arrange this service, call the toll-free MARCRAFT service line at 1-800-441-6006.

When you call to participate in the exchange program, please have the following information available:

1. The serial number of your Pro-Arm.
2. The name of the failing component.
3. Date of original purchase and the authorized dealer's name.
4. The purchase invoice number.
5. Intended method of payment (if out-of-warranty).
6. Your mailing address and telephone number.
7. Type of mailing method preferred.

Under the MARCRAFT exchange program, you have **(21)** days to return the defective equipment. After that time, you will be billed for the unreturned product. Returned merchandise must have all parts included and be in repairable condition. You will be billed for any missing parts.

ANSWERS

1. b
2. d
3. a
4. c
5. b
6. a
7. d
8. c
9. b
10. a
11. d
12. b
13. c
14. c
15. d
16. b
17. c
18. b
19. a
20. d
21. c
22. b
23. a
24. d
25. c

LAB EXPLORATION 1

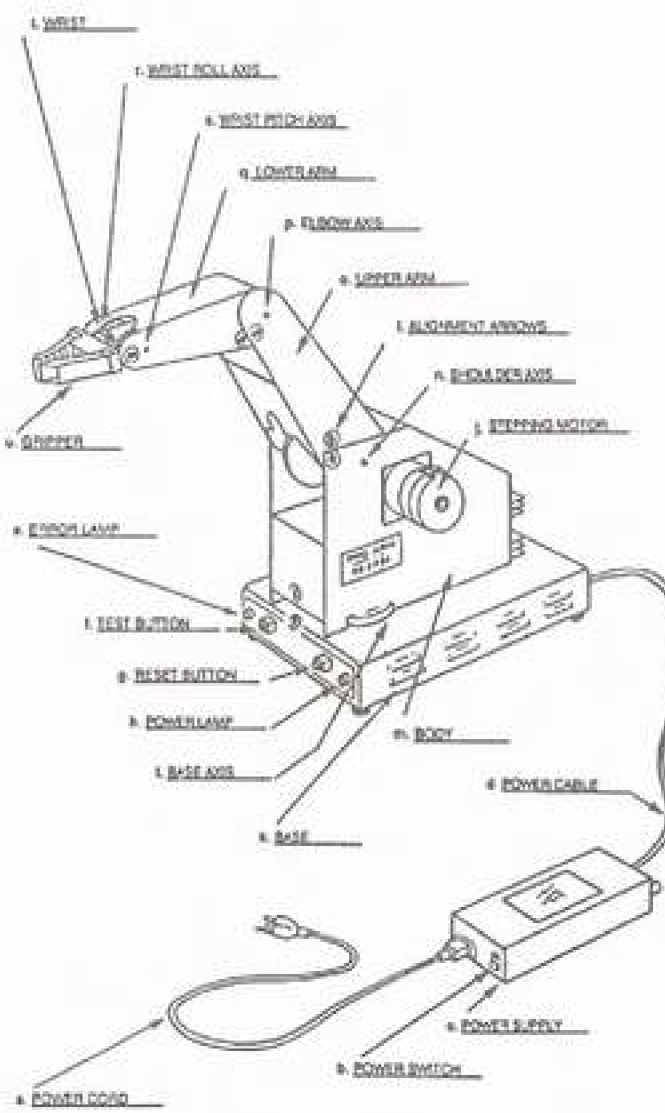


Fig. 1

LAB EXPLORATION 2

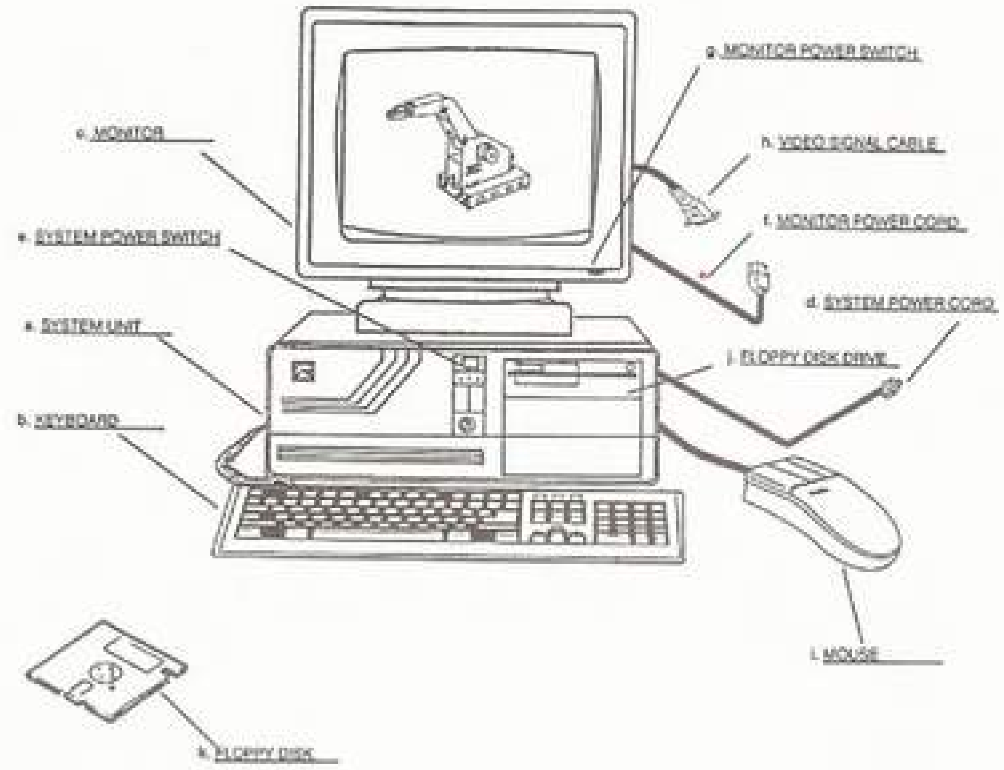


Fig. 2

LAB EXPLORATION 6

g. counterclockwise
h. clockwise
j. downward, upward
l. elbow axis, downward, upward
n. it rolled, clockwise, counterclockwise, opposite
o. it pitched, downward, upward, same
q. 4.5 inch (114 mm)
t.

TEST BUTTON DEPRESSIONS	MOTOR #	ENTITY	DIRECTION OF MOVEMENT	AXIS ACTUATED
1	M1	BODY	CCW	Base
2	M1	BODY	CW	Base
3	M2	UPPER ARM	Down	Shoulder
4	M2	UPPER ARM	Up	Shoulder
5	M3	FOREARM	Down	Elbow
6	M3	FOREARM	Up	Elbow
7	M4,M5	WRIST	CW	Wrist Roll
8	M4,M5	WRIST	CCW	Wrist Roll
9	M4.M5	WRIST	Down	Wrist Pitch
10	M4,M5	WRIST	Up	Wrist Pitch
11	M6	GRIPPER	Open	Gripper
12	M6	GRIPPER	Close	Gripper

Table 2

LAB EXPLORATION 9

a. Intelligence, mobility, manipulation skills
b. Hard automation
c. Joseph Jacquard, 1801
d. Serf, worker
e. 1962, General Motors
f. Electronics, mechanics, computer science
g. George Moore, 1893
h. Feedback
i. Microprocessor, microcomputer
j. Mechanical

LAB EXPLORATION 10

a. Pick-and-place, limited-sequence
b. Nonservo, feedback
c. Payloads, speeds, pneumatic
d. Limited-application
e. 150 lbs
f. Programmable
g. Decreases
h. High-technology
i. Six
j. High-technology
k. A universal device
l. Ten, 1 ton

LAB EXPLORATION 11

l. Yes, clockwise, 100
q. 60
r. Upward, 60, yes, elbow = minus 36 and wrist pitch = minus 30
s. Yes, minus 120
w. The "Verification" window for exiting appears, yes, the Pro-Arm moved to its "Home" position, yes

LAB EXPLORATION 12

h. Use the mouse as a pendant?, (Gripper, Delay, Fast/Slow, Nest, Movement), (Add point, Nest), Zero
i. Gripper, opening, closing, verification, Delay, 1 to 99, Fast/Slow, speed, five speeds, Nest, ZERO, Movement, Addpoint to file, Addpoint to file, CENTER, Add point, Speed, F4, Nest, Delay, F3

LAB EXPLORATION 13

c. DOS, ENTER
g. Movement control, Editing, editing, Load next point, Upward (previous) point, Remove point from file, Pendent control, double click, center, Next point, LEFT, RIGHT, Prev point, LEFT, CENTER, Delete Pt, CENTER, RIGHT
h. The Pro-Arm returned to its position prior to the point which was deleted

LAB EXPLORATION 14

c. DOS, PK TEACH, PKTEACH, ENTER
e. ZERO
f. Yes, yes

LAB EXPLORATION 15

c. A>_, DOS, PK TEACH, PKTEACH, ENTER
d. Key, Main Menu
g. Main Menu, Exit, verify
h. Off, PK TEACH, protective sleeve
i. Off, interface

LAB EXPLORATION 16

a. Installation, interfacing
c. A>_, DOS, PK TEACH, PKTEACH, ENTER
f. Yes, yes
d. Key, Main Menu
f. Yes, yes
h. Main Menu, Exit, verify
j. Yes, yes
k. Pro-Arm power supply off, PK TEACH disk, disk drive, protective sleeve
l. Computer system off, system, interface

LAB EXPLORATION 17

a. Installation, interfacing
b. Computer, apply power
c. A>_, DOS, PK TEACH, PKTEACH, ENTER, key
d. Main Menu
f. No

LAB EXPLORATION 18

a. Installation, interfacing
b. Computer, apply power
c. A>_, DOS, PK TEACH, PKTEACH, ENTER, key
d. Main Menu
f. Enter "EXIT" to return to PK TEACH
g. Utilities submenu

LAB EXPLORATION 19

a. Z80, Zilog Inc.
b. Data bus
c. Byte
d. Address bus
e. Word
f. Four, byte, word
g. CPU
h. CPU

LAB EXPLORATION 22

a. Six
b. Motor controller
c. E5/F5
d. 4-bit bistable latch, 5-V dc, TTL
e. 12-V dc, driver
f. Base
g. Overheat

LAB EXPLORATION 23

a. Bifilar
b. Two
c. Four
d. Precise angular movement, input step command
e. 0.10
f. 2000
g. Gripper limit switch, gripper motor, zero-reference point

LAB EXPLORATION 24

a. Parallel port
b. Parallel port
c. Byte-by-byte
d. American Standard Code for Information Interchange
e. Alphanumeric characters
f. Acknowledge
g. 45°
h. \

LAB EXPLORATION 25

a. 5-V dc, 1.2A
b. 12-V dc, 6A
c. 105 – 125-V ac, 200 – 264-V ac, 50/60 Hz
d. Dangerously high voltages

LAB EXPLORATION 26

a. Interlock, 12
b. Interlock, continues, delay
c. 4, 6-Pin male jack, PCB
d. Complemented
e. 0-V dc, 5-V dc
f. Zero, 4
g. 10
h. DB1, 4
i. Normally closed
j. Work cell

LAB EXPLORATION 27

e. OK
g. ALIGN.BAS, DEMO.BAS, REPEAT.BAS, VERTEN.BAS
h. .BAS
j. Program lines to DEMO.BAS are displayed.
l.
(1) The Pro-Arm is properly connected to the computer
(2) The Pro-Arm's power supply is turned on
(3) The Pro-Arm is aligned in its zero or home position
(4) The Pro-Arm's RESET button has been pressed
m. Grid sheet, Two, D19
n. How many times do you wish to execute this demo program?
p. Aligns, zero position
q. Commands
r. The message "REPEATED SEQUENCE #? BEGINS NOW" is displayed

LAB EXPLORATION 29

h.

STATEMENT NUMBER	MOTOR NUMBER	AXIS	DIRECTION	STEPS MOVED	DEGREES MOVED
30	1	Base	CW	500	60
40	1	Base	CCW	–500	60

Table 8

j.

STATEMENT NUMBER	MOTOR NUMBER	AXIS	DIRECTION	STEPS MOVED	DEGREES MOVED
30	1	Base	CW	500	60
30	2	Shoulder	Up	300	36
30	3	Elbow	Down	–200	20
30	4	Wrist R.	N/A	0	0
30	5	Wrist P.	N/A	0	0
30	6	Gripper	Open	–1000	N/A

Table 9

l. Yes

LAB EXPLORATION 30

i. ZERO
j. 500, clockwise
k. ZERO, 500, counterclockwise
o. Yes
p. LPRINT "Z", yes
q. Yes
r. Yes, LPRINT "N", 20
w. Yes

LAB EXPLORATION 31

i. 28 (approximately)
j. 2.5 in/sec or 63.57 mm/sec (approximately)
l. 9 (approximately)
m. 7.77 in/sec or 197.8 mm/sec (approximately)

LAB EXPLORATION 32

i. No, the N(est) command in Statement 60 instructs the arm to return HOME before moving to position 2
j. 50
k. 70
q. Yes, it is a M(ove) command
r. No, it is a H(ere) command and only stores in memory the arm position after Statement 30
s. 90
t. LPRINT "M500,300,0,0,0,0", the M(ove) command moves the axes a specified number of steps from the previous position, whereas the G(o-To) command moves the arm to a specified position regardless of the previous position

LAB EXPLORATION 33

h. Momentarily, while changing directions
k. Yes
l. 5
m. 5 seconds
n. Yes
t. Yes
u. 5, 8

LAB EXPLORATION 34

h. Yes
i. 3.5 inch or 90 mm (approximately)
j. minus 2000
k. 2000
l. Negative
n. 1.625 inch or 41 mm (approximately)
s. Yes
t. 3.5 inch or 90 mm (approximately)
u. Yes

LAB EXPLORATION 35

g. 800, 600
i. The motor chattered as it tried to move past its maximum limit of travel, 600, 800, no, 200, no, by the chatter in the motor
m. No, it moved upward 600 steps to its maximum limit of travel and stopped
n. Yes
p. Yes, yes, yes
q. No, The arm moved downward 600 steps and then chattered for 200 more steps trying to move past its 600-step limit. It then moved upward 200 steps and then to a position 200 steps above HOME, no

LAB EXPLORATION 36

o. 240
p.

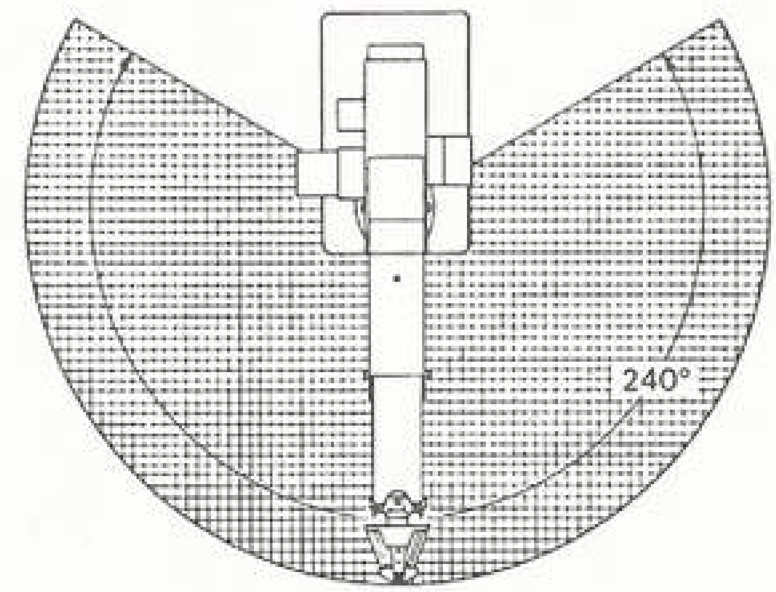


Fig. 79: Working Space Chart—Horizontal Work Envelope

LAB EXPLORATION 37

p. 17.5 inch or 445 mm (approximately)
q. See **step s** below.
r. 26 inch or 660 mm (approximately)
s.

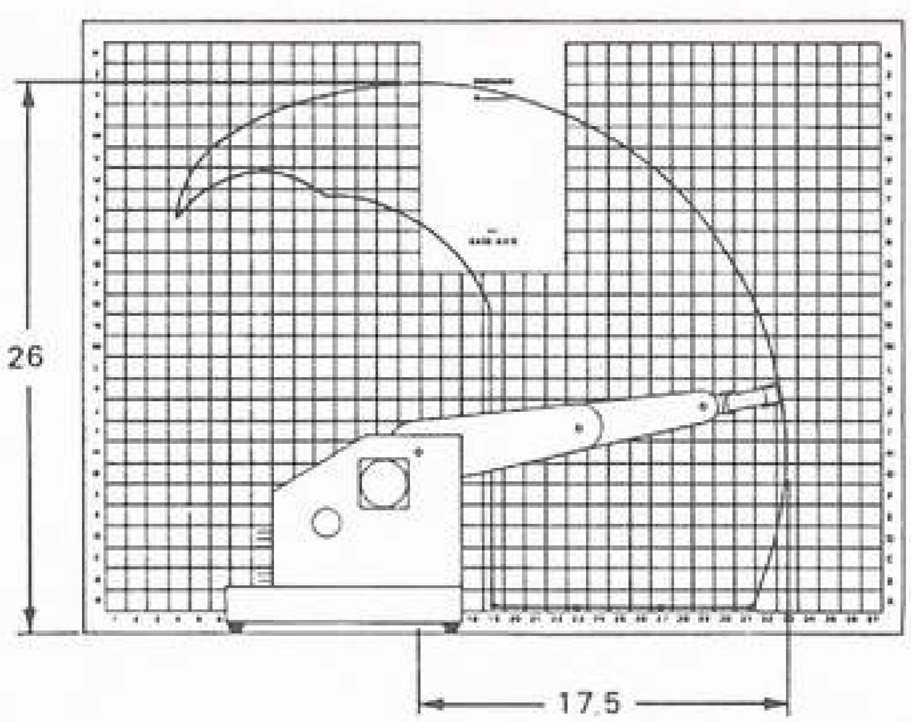


Fig. 82: Working Space Chart—Vertical Work Envelope

v.
(1) REM
(2) WIDTH
(3) KEY OFF

LAB EXPLORATION 38

j. Downward, clockwise, HOME

k.

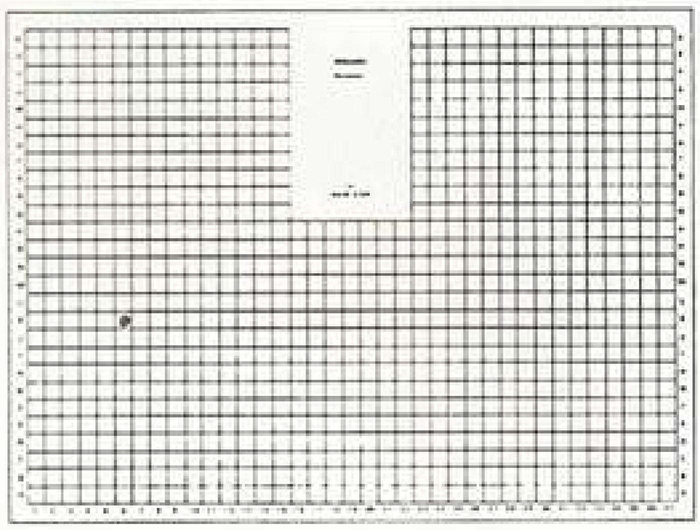


Fig. 83: First Placement Position

l.

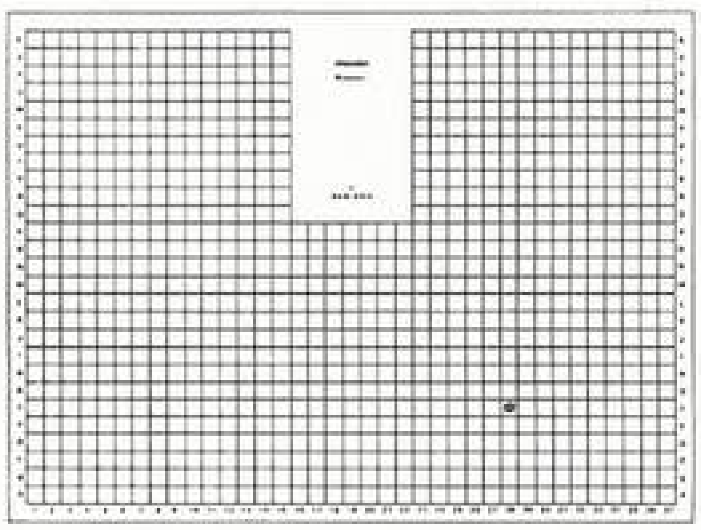


Fig. 84: Second Placement Position

n. Yes

o. 0 (approximately)

p. 0 (approximately)

LAB EXPLORATION 39

i. Base, clockwise, 150, 3 seconds

j. Shoulder, downward, 300, 3 seconds

k. No, The W(ait) command checks the interlock and finds an open-state condition which halts all arm movement

l. The arm returns to its HOME position immediately

m. Yes

n. No, By pressing the Pro-Arm's RESET button all zero-positioning data is erased and the arm's present position is assumed to be HOME

LAB EXPLORATION 40

l. The meter reads approximately 5-V dc

m. The meter reads 0-V dc

o. Yes

q. No

r. When Statement 40 is executed, all output pins at Connector 4 would go "HIGH"

s. Four

u. Yes

LAB EXPLORATION 41

l. Yes, yes, normally open

m. Yes

o. Yes, The jumper brings the input level LOW.

r. 5-V dc (approximately), yes, normally closed

s. Yes, The jumper makes the input LOW. Since the input is enabled using the normally closed configuration, the output will be the opposite of the input.

GLOSSARY

The following list includes many commonly used robotics terms.

ACCURACY: The placement repeatability of a robot. It is determined by how close the manipulator can be returned to the exact same point when repeatedly cycled. Placement and repeatability from between +/-0 .01mm and +/-0.2mm is common.

ACTUATOR: A device that converts electrical, hydraulic, or pneumatic energy to mechanical power in order to cause motion and positioning of the robot.

ADAPTABLE: The ability to make self-directed correction, often by the use of visual, force, or tactile sensors.

ARM: An interconnected series of links and powered joints that support and move an end-effector through space.

ARTICULATE COORDINATE SYSTEM: Also known as a jointed coordinate system, and is one of the most popular robots with industrial applications. It resembles the human arm and has two axes (shoulder and elbow) that have the capability of bending. It can also rotate in the horizontal plane and be fitted for a variety of end-effectors. Its work envelope forms the major portion of a sphere.

ARTIFICIAL INTELLIGENCE: The ability of a machine to perform functions that are normally associated with human intelligence, such as learning, judgement, pattern recognition, and classification.

ASCII: An acronym for the American Standard Code for Information Interchange. A binary-based standard for defining codes for information exchange between equipment produced by different manufacturers.

AXIS (plural **AXES**): Rotational joints that determine the number of intricate motions a robot can perform. Axes are also referred to as "degrees of freedom."

BASIC: An acronym for the Beginners All-purpose Symbolic Instruction Code. A common programming language which groups many different steps into easy-to-use command words. A nearly universal programming language.

BETA (β) AXIS: An axis that allows the arm to bend up and down.

CARTESIAN COORDINATES: A three-number coordinate system that uses three perpendicular axes (X,Y,Z), or planes, to identify a point in space.

CELL: See Work Cell.

CHIP: An integrated circuit.

CLOSED-LOOP SYSTEM: A system that uses feedback for control purposes. It is often referred to as a servo system.

CONTINUOUS PATH: A servo-control system that specifies all points along a desired path of motion. Servo-controlled robots allow the axes to trace out a smooth curved path.

CONTROLLER: The robot "brain" that directs the motion of the manipulator.

CURSOR: The blinking dash on the computer monitor, which indicates the position of the next character to be entered at the keyboard of the computer. It also indicates the position where any keyboard action to be executed will commence.

CYCLE TIME: The period of time required to complete a sequence of operations that are repeated on a regular basis.

CYLINDRICAL COORDINATE ROBOT: A robot whose manipulator arm has three degrees of freedom (rotation, up-and-down, and extension and retraction). Its work envelope traces out a cylinder or portion thereof.

DEAD ZONE: The area at the rear of a robot's work envelope in which the end-effector may not reach.

DEGREES OF FREEDOM: The number of independent ways in which an end-effector can move. Each axis or joint on the arm is considered a degree of freedom.

DIGITAL LOGIC: A voltage-based binary logic system with only two distinct possible states, HIGH (logic 1) or LOW (logic 0). The combination of logic states allows the representation of numbers for high-speed arithmetic operations, or of codes which represent alphanumeric characters for high-speed communication.

DRAM: An abbreviation of Dynamic Random-Access Memory. A form of RAM which requires periodic rewriting of its data (refreshing) to prevent data loss.

DRIFT: The tendency of a system's performance to slowly move away from the desired response.

DUAL CYCLE: The amount of time that a device or system is active or at full power.

ELBOW: The joint that connects together a robot's upper arm and forearm.

END-EFFECTOR: A tool (gripper, magnet, driver, etc.) attached to the wrist of a manipulator in order to perform the robot's task.

END-OF-ARM TOOLING: Same as End-Effector.

END STOP: Used on each axis of a nonservo type robot to limit the amount of travel.

FEEDBACK: The use of tactile and visual sensors to allow a controller to adjust a robot's positioning, with respect to variations affecting repeatability.

FLEXIBILITY: The ability of a robot to perform a number of different tasks.

GRIPPER: A hand-type device used by a robot to grasp and hold objects (either a work tool or the material being worked on). It is sometimes referred to as a type of end-effector. Can be actuated hydraulically, pneumatically, or electrically.

HARD AUTOMATION: Equipment designed to perform specific and repetitive functions on an industrial production line at a fast pace. A family of devices from which robots evolved.

HYDRAULIC ACTUATOR: A device for converting fluid pressure into mechanical motion.

IC: An abbreviation of Integrated Circuit. A microminiature silicon chip upon which hundreds, thousands, and even millions of electronic components and their associated circuitry have been photographically etched. Various packaging concepts are added to facilitate the incorporation of the IC into printed-circuit boards, etc.

INTERFACE: A shared boundary between independent systems that act on or communicate with each other. Often, a mechanical or electrical connection.

INTELLIGENT ROBOT: A robot that is programmable to make performance choices based on sensory inputs.

JOINT: A rotational degree of freedom in a manipulator.

JOINTED COORDINATE SYSTEM: See Articulate Coordinate System.

KEYPAD: The set of alternate numbers on the keyboard that are arranged like a calculator and activated by the NUM LOCK key. Unless activated, these keys manipulate the cursor and screen text.

KINEMATICS: The study of the motion of a rigid body without considering the forces acting upon that body.

LOAD CAPACITY: See Payload.

MANIPULATOR: The mechanical mechanism (robot arm) consisting of a series of jointed or sliding segments used to move an object through space. Often referred to as the robot "arm."

NONSERVO ROBOT: A robot which is limited to three or four degrees of freedom. Travel is restricted to the end stops, resulting in only two possible positions for each axis.

NORMALLY CLOSED: A switch configuration in which the circuit is inactive when the switch is closed and activated when the switch is opened.

NORMALLY OPEN: A switch configuration in which the circuit is inactive when the switch is open and activated when the switch is closed.

OFF-LINE PROGRAMMING: Programming a robot by first writing a set of instructions using an independent computer, and then using the software to control the robot at a later time.

ON-LINE PROGRAMMING: Programming a robot using a computer that directly controls the robot in a real-time mode.

OPEN-LOOP ROBOT SYSTEM: A system in which there are no signals coming from the axes to tell the controller the current position or velocity of the manipulator. Axes will only stop moving when the maximum travel limits are reached or when the controller removes the drive signal after a predetermined amount of time or number of steps moved. Also referred to as a nonservo system.

PAYLOAD: The maximum weight (measured at the gripper) that a robot can move under program control and still maintain specified repeatability and reliability standards (at maximum speed and reach).

PCB: An abbreviation of Printed Circuit Board. A board for the mounting of electronic components on which most connections are made by printed circuitry composed of a conducting metal, such as copper, arranged in a predetermined pattern on an insulating substrate.

PENDANT: Usually a hand-held device having programming buttons that command the controller to move and record in memory the current position of a robot's axes as a control program is developed. Often called a "Teach Pendant."

PICK-AND-PLACE ROBOT: A robot that operates by moving its axes between two end points with little or no trajectory control.

PITCH: The up-and-down rotational movement of a wrist.

PNEUMATIC ACTUATOR: A device for converting air (gas) pressure into mechanical motion.

POINT-TO-POINT: A servo-control system whereby motion is made between end stops. A limited number of points are specified by the controller and the path of motion is from point-to-point rather than in a continuous, smooth path.

POLAR COORDINATE SYSTEM: See Spherical Coordinate System.

POWER SUPPLY: Any device which provides and regulates the energy used by a robot's actuators, whether hydraulic, pneumatic, electric, or some combination.

PROGRAMMABLE: Capable of receiving and processing specific instructions to perform a sequence of steps.

PROXIMITY DETECTOR: A device that senses an object and can measure its distance from the object.

R (REACH) AXIS: An in-and-out axis.

RAM: An abbreviation of Random-Access Memory. A form of volatile computer memory in which data can be either read from or written to without regard to data sequence.

REACH: The maximum distance from the center line of a robot to the end of its tool mounting plate.

RELIABILITY: The percent of time for which normal operation may be expected. Also referred to as "up time."

REPEATABILITY: The degree to which a robot system is able to return to a specific point.

ROBOT: A reprogrammable multifunctional manipulator designed to move material, parts, tools, or specialized devices through variable programmed motions for the performance of a variety of tasks. (Definition accepted at the 11th International Symposium of Industrial Robots.)

ROLL: Rotational movement around the center line of a wrist. Sometimes referred to as "twist."

SENSOR: A transducer or feedback device that can interact with an object to provide a quantitative measure of a characteristic of the object.

SERVO: A device (motor) that provides a robot with positional control over the entire range of travel about its axis.

SHOULDER: The portion of a manipulator arm that is attached to the base.

SPEED: The rate at which the end of the manipulator arm can move a load under program control. Usually expressed in inches per second or millimeters per second.

SPHERICAL COORDINATE ROBOT: Also known as the polar coordinate system. It has three axes of movement (extension and retraction, vertical pivot, and horizontal rotation) and traces out a sphere, or portion thereof, as its work envelope.

SRAM: An abbreviation of Static Random-Access Memory. A form of RAM which require no refreshing or rewriting of data.

STEPPER MOTOR: A motor which uses electric pulses to move the armature a given number of degrees.

STRAIGHT-LINE VELOCITY: A term used to describe the speed rating of high-technology industrial robots. It measures how quickly an end-effector can move between two distinct points.

SWING: Rotation about the center line of the robot.

TACTILE SENSOR: A sensor that can detect the presence of an object or measure torque or force through contact with the object.

TEACH: The process of programming a manipulator arm to perform a desired sequence of tasks.

THETA (θ) AXIS: A rotational axis.

TORQUE: The turning effort that is required to rotate a mass through a radius.

TRANSDUCER: A device that converts one form of energy into another. Many times, physical quantities are converted into electrical signals.

TWIST: See Roll.

U AXIS: The elbow axis which moves the forearm.

UPPER ARM: The portion of a manipulator arm that is attached to the shoulder.

VOLATILE: In computer language, a form of memory which is lost as soon as power is removed.

W AXIS: The shoulder axis that moves the upper arm of a manipulator.

WALKTHROUGH PROGRAMMING: A method of programming a robot whereby the manipulator arm is physically moved through a complete operating cycle.

WORK CELL: The total environment (including other robots) in which the robot performs its task.

WORK ENVELOPE: The three-dimensional space that includes all of the points that can be reached by an end-effector.

WORKING SPACE: Same as Work Envelope.

WRIST: A set of rotary joints that allows an end-effector to be oriented to its workpiece.

YAW: A wrist's angular movement from side-to-side.

Z AXIS: An up-and-down axis.