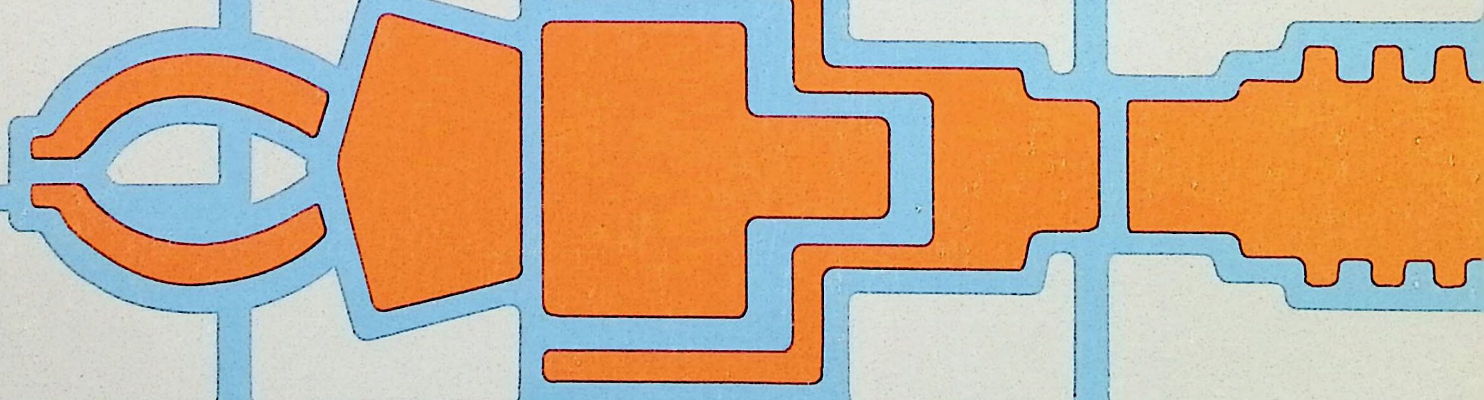
ET-18 ROBOT

TECHNICAL MANUAL



Heathkit
Zenith Educational Systems

HEATH COMPANY PHONE DIRECTORY

The following telephone numbers are direct lines to the departments listed:

Kit orders and delivery information	(616) 982-3411
Credit	(616) 982-3561
Replacement Parts	(616) 982-3571

Technical Assistance Phone Numbers
8:00 A.M. to 4:30 P.M., EST, Weekdays Only

Audio,	(616) 982-3310
Amateur Radio	(616) 982-3296
Test Equipment, Weather Instruments and Home Clocks	(616) 982-3315
Television	(616) 982-3307
Aircraft, Marine, Security, Scanners, Automotive, Appliances and General Products	(616) 982-3496
Computers — Hardware	(616) 982-3309
Computers — Software: Operating Systems, Languages, Utilities	(616) 982-3860
Computers — Software: Application Programs	(616) 982-3884

YOUR HEATHKIT 90-DAY LIMITED WARRANTY

Consumer Protection Plan for Heathkit Consumer Products

Welcome to the Heath family. We believe you will enjoy assembling your kit and will be pleased with its performance. Please read this Consumer Protection Plan carefully. It is a "LIMITED WARRANTY" as defined in the U.S. Consumer Product Warranty and Federal Trade Commission Improvement Act. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Heath's Responsibility

PARTS — Replacements for factory defective parts will be supplied free for 90 days from date of purchase. Replacement parts are warranted for the remaining portion of the original warranty period. You can obtain warranty parts direct from Heath Company by writing or telephoning us at (616) 982-3571. And we will pay shipping charges to get those parts to you . . . anywhere in the world.

SERVICE LABOR — For a period of 90 days from the date of purchase, any malfunction caused by defective parts or error in design will be corrected at no charge to you. You must deliver the unit at your expense to the Heath factory, any Heath/Zenith Computers and Electronics center (units of Veritechnology Electronics Corporation), or any of our authorized overseas distributors.

TECHNICAL CONSULTATION — You will receive free consultation on any problem you might encounter in the assembly or use of your Heathkit product. Just drop us a line or give us a call. Sorry, we cannot accept collect calls.

NOT COVERED — The correction of assembly errors, adjustments, calibration, and damage due to misuse, abuse, or negligence are not covered by the warranty. Use of corrosive solder and/or the unauthorized modification of the product or of any furnished component will void this warranty in its entirety. This warranty does not include reimbursement for inconvenience, loss of use, customer assembly, set-up time, or unauthorized service.

This warranty covers only Heath products and is not extended to other equipment or components that a customer uses in conjunction with our products.

SUCH REPAIR AND REPLACMENT SHALL BE THE SOLE REMEDY OF THE CUSTOMER AND THERE SHALL BE NO LIABILITY ON THE PART OF HEATH FOR ANY SPECIAL, INDIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES, INCLUDING BUT NOT LIMITED TO ANY LOSS OF BUSINESS OR PROFITS, WHETHER OR NOT FORESEEABLE.

Some states do not allow the exclusion or limitation of incidental or consequential damages, so the above limitation or exclusion may not apply to you.

Owner's Responsibility

EFFECTIVE WARRANTY DATE — Warranty begins on the date of first consumer purchase. You must supply a copy of your proof of purchase when you request warranty service or parts.

ASSEMBLY — Before seeking warranty service, you should complete the assembly by carefully following the manual instructions. Heathkit service agencies cannot complete assembly and adjustments that are customer's responsibility.

ACCESSORY EQUIPMENT — Performance malfunctions involving other non-Heath accessory equipment, (antennas, audio components, computer peripherals and software, etc.) are not covered by this warranty and are the owner's responsibility.

SHIPPING UNITS — Follow the packing instructions published in the assembly manuals. Damage due to inadequate packing cannot be repaired under warranty.

If you are not satisfied with our service (warranty or otherwise) or our products, write directly to our Director of Customer Service, Heath Company, Benton Harbor MI 49022. He will make certain your problems receive immediate, personal attention.

HERO ROBOT

Model ET-18A

Technical Manual

595-3071-03

HEATH COMPANY
BENTON HARBOR, MICHIGAN 49022

WARNING

This equipment has been verified to comply with the limits for a Class B computing device, pursuant to Subpart J of part 15 of FCC Rules.

This equipment generates and uses radio frequency energy for its operation and if not installed and used properly, that is, in strict accordance with the instruction manual, may cause interference to radio and television reception. It has been type tested and found to comply with the RF emission limits for a Class B computing device which is intended to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Move the equipment away from the receiver being interfered with.
- Relocate the equipment with respect to the receiver.
- Reorient the receiving antenna.

If you need additional help, consult your dealer or ask for assistance from the manufacturer. Customer service information may be found on the inside back cover of this Manual or on an insert sheet supplied with this equipment. You may also find the following booklet helpful: "How to Identify and Resolve Radio-TV Interference Problems." This booklet is available from the U.S. Government Printing Office, Washington, D.C. 20402. — Stock No. 004-000-00345-4.

TABLE OF CONTENTS

FCC Statement . . . . Page 2
Table of Contents . . . . Page 3
Introduction . . . . Page 5
Specifications . . . . Page 7
Programming Techniques . . . . Page 9
    Keyboard Operation . . . . Page 10
    ROM Subroutines . . . . Page 20
    ROBOT Interpreter . . . . Page 23
    "Program External" Data and
        Interrupt Locations . . . . Page 32
Disassembly and Component Locations . . . . Page 39
    Panel Removal . . . . Page 39
    Fuse and Battery Locations . . . . Page 39
    Main Component Locations . . . . Page 40
    Plugs and Connectors . . . . Page 41
    Mechanical Assemblies . . . . Page 42
Adjustments . . . . Page 43
    Arm and Shoulder Rotation . . . . Page 44
    Power Supply . . . . Page 47
    Sonar Transmit Frequency . . . . Page 48
    Speed Control . . . . Page 48
    Sense . . . . Page 49
    Motion Detector . . . . Page 49
    Speech . . . . Page 50
    Time . . . . Page 50
    Steering . . . . Page 51
    Drive Wheel Sensor . . . . Page 51
Troubleshooting . . . . Page 53
    Operation Check . . . . Page 54
    Initial Tests . . . . Page 55
    Troubleshooting Charts . . . . Page 62
Theory of Operation . . . . Page 71
    Block/Interconnect Diagram . . . . Page 71
    Circuit Description . . . . Page 72
Circuit Board X-Ray Views . . . . Page 83
Semiconductor Identification Charts . . . . . . Page 85
Replacement Parts List . . . . Page 99
Appendix . . . . Page 113
    Programmer's Information Sheet . . . . . . Page 115
    6808 Instruction Set . . . . Page 116
    ROM Options and
        CPU Jumper Placement . . . . Page 118
    Port Map . . . . (fold-out from) Page 119
    Programs List . . . . Page 120
Index . . . . Page 121
Schematics . . . . (fold-in)
Customer Service . . . . (inside back cover)

INTRODUCTION

As the Table of Contents shows, this Hero Robot Technical Manual contains valuable information for the advanced user, for the technically-minded person who wants to know how it works, and for the technician who might have reason to service the Robot.

A good portion of this Manual is devoted to "Programming Techniques," where Robot Language is explained, as well as ROM subroutines and speech programs. This section will help you achieve the most use and application of your Hero Robot and its many unique functions.

Although there are very few adjustments, and they will seldom require readjustment, complete instructions are presented to help you maintain peak operating condition of Hero. Many illustrations are included to show you how to remove the outer panels, locate the components and adjustments, and for the replacement of mechanical assemblies.

Should your Hero ever fail to operate properly, the Troubleshooting section will help you to trace a problem to a particular area or circuit board. Then, if you are an experienced technician, you can narrow the problem to a given circuit or component and repair it. If you do not have training or experience in servicing digital electronic equipment, we strongly recommend that you limit your troubleshooting to the locating of a faulty circuit board. Then send that circuit board to the Heath Company for repair. (See "Service Information" inside the rear cover of this Manual.)

A "Theory of Operation" section features a "Block Diagram and Interconnect" where the various circuits and interconnecting lines are discussed in general nontechnical terms. The block diagram discussion will provide a nontechnical person with a basic understanding of how it works. This is followed by a "Circuit Description" of the individual circuits in a more technical language, related to electronic components that are shown on Schematic Diagrams.

The "Schematic Diagrams," "X-Ray Views" of the circuit boards, "Semiconductor Identification Charts," and "Replacement Parts Lists," will prove useful in troubleshooting, locating specific parts on the circuit boards, and if you need to order a replacement part. (See "Replacement Parts" on the inside rear cover.)

Finally, the "Appendix" section presents the 6800 Instruction Set and a Robot Language instruction set with memory map that will aid the advanced user in programming Hero. And a discussion of the Memory Options and jumper placement will allow you to expand the use of Hero for many additional sophisticated applications.

In general, every effort has been made to make this Technical Manual complete and comprehensive for the advanced user, for the service technician, and for the student of electronics and robotics.

SPECIFICATIONS

SENSES

Sound: Detects and quantifies ambient sound levels over the frequency range of 200 to 5000 Hz. Resolution is 1 part in 256. The sound sensor is approximately omnidirectional.

Light: Detects and quantifies ambient light levels over the visible spectrum. Resolution is 1 part in 256. Sensor reception angle is approximately 30 degrees with the head panel installed.

Ultrasonic Ranging: Pulsed ultrasonic system. Maximum range is 8 feet (2.4 meters). Range resolution is .4 inch (1 centimeter). Horizontal and vertical beam width are each approximately 30 degrees. Transmit frequency is approximately 32 kHz.

Motion: Detects motion about it using a continuous wave ultrasonic system. Sensitivity depends on the size and relative movement of the object detected. Will detect an average-sized adult walking toward the sensor at a distance of 15 feet. If the sensor is pointed toward a wall, motion detection is approximately omnidirectional.

Speech: Synthesized phoneme-based system. Generates 64 phonemes which can be concatenated in any combination to simulate human speech or various sound effects. Reference pitch is hardware selectable, and instantaneously variable over four levels of inflection using software.

Time: Hardware four-year calendar clock. Counts seconds, minutes, hours, day of week, day of month, month of year. Accuracy after initial set-up is + or – 120 seconds per year.

HEAD

A stepper motor rotates the head 350 degrees. The head contains sensors for the motion detector, light detector, ultrasonic ranging system, and sound detector. It also contains the breadboarding area (Experimental Board), the LED display and hexadecimal keyboard, the Abort button, and provides a mounting point for the arm.

The breadboarding area (Experimental Board) provides direct access to a user I/O port, a user defined interrupt, CPU control lines, and +12 and +5 volts.

The hexadecimal keyboard and display provide for entry and display of data to and from the CPU.

ARM

The arm rotates 350 degrees in the horizontal plane with the head. The shoulder motor raises and lowers the arm 150 degrees in the vertical plane. An extender motor extends and retracts the gripper 5 inches (12.7 centimeters.) A wrist pivot motor pivots the gripper 180 degrees total (90 degrees above or below axis of arm). The wrist rotate motor rotates the gripper through 350 degrees. The gripper motor opens the gripper 3-1/2" (9 centimeters). The wrist assembly may rotate 90 degrees at the extreme end of arm extension (user option).

The arm maximum payload (with arm horizontal and fully retracted) is 16 oz (450 grams); and with the arm fully extended is 8 oz (225 grams). The gripper force at the tip of the gripper is 5 oz (140 grams) minimum.

TORSO

Provides mounting surfaces for the main circuit boards. Access to the main wiring harness and CPU board is through a hinged door at the back of the torso assembly.

BASE

Contains the mechanical components of the main drive and steering mechanisms. Houses three of the batteries and supports the torso and head assemblies.

BATTERIES

Four 4-ampere-hour rechargeable gel cells, connected in two electrically independent 12-volt systems (logic system and drive system). The batteries are protected against total discharge by an automatic low-voltage sensor, which shuts down most Robot functions (subject to user override) if either system falls to 10 volts.

CHARGER

Power Requirements	120/240 volts AC, 50/60 Hz, 60 watts (max).
Output Voltage	27 volts (max.) unregulated.
Output Current	1.75 amps (max.) into fully discharged batteries.
Recharge Time	10 hours (max.) with Robot off. Longer with Robot on, depending on Robot functions being used.

GENERAL

Operating Temperature	0°C to +40°C (32°F to 104°F) ambient.
Weight	39 Lbs. (17.6 kg), with accessories.
Dimensions	20" high × 18" diameter (max). (50.8 × 45.7 cm).
Minimum Turning Radius	12" (30.5 cm).

The Heath Company reserves the right to discontinue products and to change specifications at any time without incurring any obligation to incorporate new features in products previously sold.

PROGRAMMING TECHNIQUES

This portion of the Technical Manual is written for people who have some working knowledge of machine language programming for the 6800 series microprocessor. Since most of the commands that the Robot recognizes are in 6800 language, you need to master the vocabulary and concepts before you can progress very far.

If you are not knowledgeable in the 6800 machine language, you need to study some source material before you go further in this section. You can find the information in a number of Heathkit/Zenith Educational products, such as the EE-1800 Robotics course, the EE-3401 Microprocessors course, or the EC-6800 Introduction to Microprocessors course. With one of these, you will master the concepts and vocabulary necessary to understand the following material.

The following will cover the Robot's machine language subroutines and the commands available in the special Robot interpreter (Robot Language). In addition, there will be information about special locations in memory that you can use to accomplish various programming feats more easily.

Throughout this portion of the Manual, we will use the hexadecimal numbering system whenever we talk about memory addresses, numbers on the display, or keyboard entries.

KEYBOARD OPERATION

This section of the Manual deals with the keyboard and tells you how to use it to enter, debug, and modify your programs. A sample program is included.

KEYBOARD

NOTE: For your convenience the keys, except RESET, are shown below as you would actually see them, ACCA 1, for example.

The keyboard allows you to quickly enter commands and data to the microprocessor. After you press the RESET key, the display will show "HEro 1.X." Then press either ACCA 1 or AUTO A to enter the Machine Language Program Mode or the Robot Language Repeat Mode. The next keyboard entry will be interpreted as a command. The following paragraphs discuss the various commands available in either mode.

Display Accumulator A

ACCA 1 Press this key and the contents of accumulator A will be displayed. The first four digits and decimal point will identify the display, and the next two digits show the contents of the accumulator.

In the following example, the contents of accumulator A is $4A_{16}$* (or binary 01001010).

Example: Acca.4A

Now you may change the contents of accumulator A if you wish. To do this, press the CHAN C key. The display will now be:

Acca. _ _

With two key strokes, enter the new hexadecimal number you want in accumulator A.

Display Accumulator B

ACCB 2 Press this key and the contents of accumulator B will be displayed. A typical display is:

Accb.5F

*NOTE: In this section of the Manual, the subscript sixteen will be used to indicate hexadecimal numbers.

In this example, accumulator B contains $5F_{16}$ (binary 01011111).

The contents of accumulator B can be changed in the same way that accumulator A is changed.

Display Program Counter

PC 3 Press this key and the contents of the microprocessor's program counter will be displayed. The first two digits and decimal point identify the display, and the next four digits show the contents of the program counter.

Example: Pc.0617

In this example, the program counter may contain any number. You may change the program counter by pressing the CHAN C key and then entering the new hexadecimal number.

Display Index Register

INDEX 4 Press this key and the contents of the index register will be displayed.

Example: In.FDF4.

You can change the register by pressing the CHAN C key and then entering a new hexadecimal number.

Display Condition Codes Register

CC 5 Press this key and the contents of the condition codes register (1's and 0's) will be displayed. The display letters (H, I, N, Z, V, and C)* correspond to the letters assigned to the six condition codes.

Example: 001001

This register **cannot** be changed by pressing the CHAN C key.

*NOTE: You can refer to the six character locations on the display by using the letters "HINZVC," with the left-most location being the "H" and the right-most being the "C." This identification comes from the Condition Code Register bits that they represent: H= Half-carry, I= Interrupt mask, N= Negative, Z= Zero, V= oVerflow, and C= (full)-Carry.

Display Stack Pointer Register

SP 6 Press this key and the contents of the stack pointer register will be displayed.

Example: SP.0F9E

This register **cannot** be changed by pressing the CHAN C key.

Resume User's Program

RTI 7 Press this key and your program will start at the location contained in the program counter. This key is used to return to normal user program operation from breakpoints or single stepping.

Single Step User's Program

SS 8 Press this key and the microprocessor will perform only one step of your program. The instruction to be performed is taken from the address contained in the program counter. After the step, the next instruction and its address are displayed. The displayed instruction may be changed by pressing the CHAN C key and then entering the new data. Also at this time, you may examine registers, memory, or use any of the other monitor functions.

Set Breakpoint

BR 9 Press this key and you can then make an entry into the monitor breakpoint table. A breakpoint is a point where you want to stop the program to examine the microprocessor registers, memory, etc.

The display is _ _ _ _ br.

Enter the four digits of a hexadecimal address for the breakpoint. The address must be the address of an operational code in your program and that code must be in RAM. No breakpoints are possible in ROM. You may have up to four breakpoints in your program at any one time. Beyond that, a "FULL" message will display.

Do not press the RESET key. This clears all the breakpoints.

If you make an incorrect entry, and the entry is still displayed, press the CHA C key as many times as necessary for the display to return to _ _ _ _ br. Then enter the correct address.

Auto Load Of Memory

AUTO A Press this key and _ _ _ _ Ad will be displayed.

Enter the address you want to start at. Example: Enter 0, 6, 0, and 0. The display is now:

0600 _ _.

Enter the 2-digit hexadecimal value you want entered at that address.

The display will now advance to the next address. You can continue changing memory data until you press the RESET or ABORT keys.

Display Previous Address

BACK B Press this key when an address and its data are displayed (you are examining memory with the E function, your program has come to a breakpoint, or you are single stepping your program), and the previous address and its data will be displayed. You may change this data by pressing the CHAN C key and then entering the new data.

Change Displayed Value

CHAN C Press this key when an address and its data are displayed, and the data will be replaced with "_ _". Then enter the new hexadecimal value you want at this address. You may use this function to correct a value you entered by mistake.

DO User Program

DO D Press this key and the display will become:

_ _ _ _ do.

Enter the beginning address of your program. Your program will now start at the new address instead of where the program counter was pointing. The display will become blank and the program will run until a display is called for, until it comes to a breakpoint, or until you press the RESET or ABORT keys, or return to the executive mode.

This key function combines several other functions. You could get the same result by displaying and changing the program counter and then pressing the RTI 7 key.

Examine Memory

EXAM E Press this key and the display will become:

----Ad.

Enter a new address. The display will now indicate the data at this new address. You may now change the displayed value by using the CHAN C key or you can step backwards or forwards through memory using the BACK B and FWD F keys.

Display Next Address

FWD F Press this key when an address and its data are displayed, and the next address and its data will be displayed. You may change this data by pressing the CHAN C key and then entering the new data.

ENTERING PROGRAMS

The entry below shows the first two instructions of the Sample Program (Page 19) and indicates the various information they contain. This information is further described in the following paragraphs.

Instruction Address: Usually called the Program Counter. In order to perform an instruction, the Program Counter must contain the address that is in this column. RTI and SS require the Program Counter to contain the address that is in this column for proper execution. The address entered after DO is pressed must be an instruction address. Breakpoints are not recognized except at instruction addresses.

Instruction: Normally one, two, or three bytes of data as required by the addressing mode used.

Opcode: A byte of information referred to as machine code. It indicates in hexadecimal the operation to be performed.

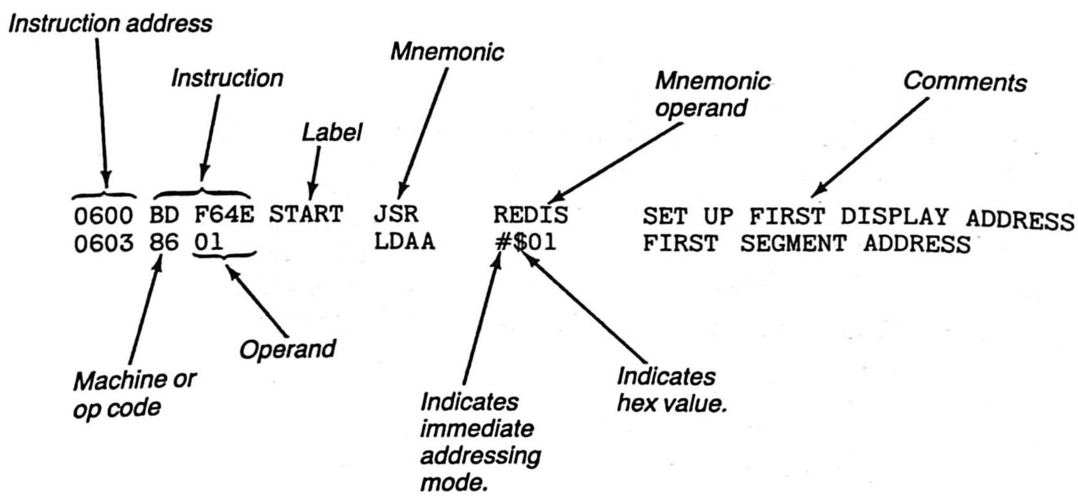
Operand: This is additional hexadecimal information required to perform the operation. It may be zero, one, or two bytes as determined by the addressing mode.

Label: Usually a name applied to a subroutine in the program used more than once. In the sample programs, the address to be entered to begin execution is labeled "Start."

Mnemonic: A three-letter indication of the source instruction. A fourth letter, A or B, is added to indicate which of two accumulators if the instruction applied to either one.

Mnemonic operand: Again, this is additional information that is required for the operation. It may be a label, address, or data. The $ sign indicates the information is a hex value. The # sign indicates the immediate addressing opcode is to be used.

Comments: A brief description of what is happening in the program.



When you load a program into Hero, only the one, two, or three* bytes of each instruction are entered. You may use either of two methods to enter the instructions: "Auto Load", or the more laborious "Examine and Change." Forward, Back, and Change are valid commands in the Examine, and Change method and may be used to correct entry errors. However, they are not valid for the Auto Load method. If you make an error in the Auto Load method, press the RESET, the ACCA 1 or AUTO A, and AUTO A keys. Then enter the address where the error was made and continue from there; or, remember where the error was made and then examine and change that memory location after you finish entering the entire program.

The following charts show the sequence of events you would use to enter the first two instructions of the Sample Program. The first chart shows the Examine and Change method while the second chart shows the Auto Load method.

Examine and Change

Press ACCA 1 or AUTO A to enter the machine language or Robot language programming mode.

Press the EXAM key and then enter the first instruction address, 0600. Then check the display and continue to enter the program as shown below.

Display is	→ Press	→ Enter	→ Display is	→ Press
0600XX	CHAN	BD	0600BD	FWD
0601XX	CHAN	F6	0601F6	FWD
0602XX	CHAN	4E	06024E	FWD
0603XX	CHAN	86	060386	FWD
0604XX	CHAN	01	060401	FWD
0605XX	CHAN	etc.		

Auto

Press ACCA 1 or AUTO A.

Press the AUTO key and then enter the first instruction address, 0600.

Display is	Enter
0600--	BD
0601--	F6
0602--	4E
0603--	86
0604--	01
0605--	etc.

Press RESET after last entry to exit.

*Up to 8 bytes may be used in the special case of the "Move All Motors" command, Page 24.

If you use Examine and Change, the last entry in the Sample Program (Page 19) results in the display 0629DF, and this display remains until you enter a new command through the keyboard.

If you use Auto, the last display will be the address of the next continuous memory location, which is the last program instruction address plus the number of bytes in the instruction. In this program, 0627 plus two; 0629 _ _. The dash (or "prompt") characters are displayed in the data locations.

After you enter a program, by either method, check the ending address to be sure that you have not omitted or double entered data.

Enter the Sample Program (on Page 19) into your Robot. Use either of the two entry methods.

If you used the Examine and Change method to enter the program, you can run the program by pressing DO and entering the address of the instruction labeled "Start," 0600. If you used the Auto method, first press the RESET key. Press 1 to select the (machine language) Program Mode. Then press DO and enter the address of the instruction labeled "Start," 0600.

USING BREAKPOINTS

We will now use the Sample Program to show how programs can be inadvertently changed and even "crash" when breakpoints are inserted at improper locations (at addresses other than the instruction address).

Press RESET, press 1, and insert breakpoints at 0604 and 0605.

Press BR 0604.

Press BR 0605.

Start the program by pressing DO 0600.

Notice that the CPU has ignored the breakpoint at 0604 and stopped at 0605.

Examine 0603 and 0604 by pressing EXAM, 0603, and FWD. The instructions thêre are correct (86 and 01).

Examine accumulator A by pressing the ACCA key.

Accumulator A has been loaded with the software interrupt instruction 3F that was temporarily placed at 0604 by the breakpoint at that address.

Watch the "H"* display and press RTI. The 3F in accumulator A caused the first display to be incorrect. The program will stop next at 0605.

Insert a breakpoint at 0602 and then press RTI to resume execution. The program will run until it comes to 0602 which is changed by the breakpoint to 3F. The program will "crash" because of the wrong instruction and the "C" display will have its b and c segments lit, showing a "1." Press RESET to return control to the monitor program.

Press 1, then EXAM, and enter 0600. Use FWD and CHAN to examine and correct errors introduced when the program crashed. You will always find the data at the breakpoint addresses has been changed. More often than not, the data at the breakpoint addresses will become 3F, although this may also change because the program crashed. Before you proceed, run the program to be sure all errors have been corrected.

In order to properly execute SS or RTI, the program counter must contain the instruction address where you wish to start. If single step begins at an incorrect address, the single step routine will not execute an invalid instruction and the display will not change. If the instruction at the PC address is a valid opcode, SS will execute the instruction using the following bytes as necessary and will continue unless it comes to an invalid instruction. RTI will try to execute the instruction in the same manner; except that in the case of an invalid instruction, the program will probably crash. We will use SS to illustrate what happens.

Push RESET, push 1 to select the Program Mode. Examine the Program Counter by pushing PC. Then change it to 0618 by pushing CHAN and 0618.

Press SS. The instruction at 0618 is not a valid instruction. In the single step mode, the machine will reject the instruction, 0618 FD will continue to be displayed, and nothing will happen. If you press RTI, the program will crash as it would when an invalid instruction is encountered. Probably only the first instruction will be changed, if any, in this particular circumstance. If you press RTI to see what happens, examine the program afterwards and correct any errors introduced; then run the program to be sure it is correct before proceeding.

Examine and change the Program Counter to 0611 by pressing PC, CHAN, and 0611. Press SS. In this case F7 is a valid instruction, STAB extended, and B is stored at nonexistent memory location C8CE.

Press SS. Here again, 2F is a valid instruction (a conditional branch BLE). A branch may occur to 0617 or the program may fall through to 0616. In either case, two incorrect instructions have been performed in place of two or three correct instructions, introducing error in the program. This is of no great consequence in this program, but may be in another. Since an invalid instruction was not encountered, placing the Program Counter at 0611 and pressing RTI would do exactly the same thing.

Now sample program 1 will be used to illustrate a procedure using breakpoints and single step to go through a program.

There are two important considerations pertaining to reserved memory bytes to keep in mind. First; DIGADD is used by all monitor routines. (Refer to Page 21 for a discussion of the two memory locations called DIGADD.) If you examine these memory locations, 0FF2 and 0FF3, you will always find C12F (the "V" display address*) there because the Examine command puts it there before it outputs the data. Secondly; DIGADD is always loaded with C16F (the "H" display address*) when DO or RTI are used.

Single step uses several temporary locations in common with many of the monitor routines. Single step will replace information stored at these locations by the monitor routines. As a result, the routine may return with incorrect information or it may not be able to return at all and the program will crash.

When the program stops, at a breakpoint or after a single step, the address of the next instruction (contained in the program counter) and the instruction will be displayed. You may examine and make changes to any register (except the condition code register and the stack pointer) or address provided you DO NOT change the program counter. The instruction displayed when the program stopped will be the next one executed when SS or RTI is pressed, regardless of what is being displayed. If the counter is changed, execution will continue at the address pointed to by the program counter.

The following procedure gives instructions. The six characters on the right, on the same line, indicate what the display should be after you perform the instruction. You will be instructed to examine registers affected by the instruction that has been executed.

*NOTE: The six character locations on the display can be referred to using the letters "HINZVC," with the left-most location being the "H" and the right-most being the "C." This identification comes from the Condition Code Register bits that they represent: H= Half-carry, I= Interrupt mask, N= Negative, Z= Zero, V= oVerflow, and C= (full)-Carry.

SAMPLE OPERATION

You may examine any other registers or memory locations if you wish. The comment after an instruction is explanatory information.

INSTRUCTION	DISPLAY
Press RESET	HEro1.X
Press 1	1._____
Press PC, CHAN, and enter 0600. The program counter now contains the start address.	Pc 0600
Press SP. This is the next location available on the stack. The JSR instruction should store the address for return from the REDIS subroutine (0603) at this location.	SP.0F9E
Press SS, Jump to REDIS.	F64E FF
Press EXAM and 0F9D	0F9d 06
Press FWD. Return address is on stack.	0F9E 03
Press BR and enter 0603 to get past monitor routine.	0603 br
Press RTI. You might normally use examine to check the result of the routine. In this case, DIGADD is loaded with C16F. Examine will just change what is there, so do not examine DIGADD.	0603 86
Press SS.	0605. 20
Press ACCA. (A) is loaded with the correct value.	Acca.01
Press SS. Branch to OUT-offset correct.	0610 bd
Press SS. Jump to OUTCH.	F7C8 FF
Press BR and enter 0613 to get past monitor routine. You could check the stack here if desired.	0613 br
Press RTI. Exit OUTCH address of next display in DIGADD; Do not check.	0613 CE
Press SS.	0616 09
Press INDEX. Is (X) loaded?	In.2F00
Press SS.	0617 26
Press INDEX. Is (X) decremented?	In.2EFF

Heathkit®

INSTRUCTION	DISPLAY
Press CC. Z bit is clear if (X) is not 0000 yet.	XXX0 XX (X = don't care)
Press SS. Branches to WAIT if Z was clear.	0616 09
Press INDEX, CHAN, and enter 0001.	In.0001
Press SS.	0617 26
Press CC. (X) decremented to 0000 sets Z bit. Should drop through branch now.	XXX1 XX
Press SS. It did.	0619 16
Press ACCA.	Acca.01
Press SS.	061A 5d
Press ACCB. What was in (A) should be in (B).	Accb.01
Press SS.	061b 26
Press CC. Z bit is clear if (B) is not 00.	XXX0 XX
Press SS. Branches to SAME if Z is clear.	0607 F6
Press SS.	060A Cb
Press ACCB. When the program runs normally, (B) at this point would be 5F because exit from OUTCH would be with the next display address, C15F, in DIGADD. Single step has caused DIGADD to be C10F.	Accb.0F
Press SS.	060C F7
Press ACCB. Hex 10 has been added to (B).	Accb.1F
Press SS. (B) has been stored at DIGADD. There is no reason to examine 0FF3, since EXAM and SS will change what is there anyway.	060F 48
Press ACCA.	Acca.01
Press SS.	0610 bd
Press ACCA. ACCA was 0000 0001 binary (01 hex). It has been shifted left and is now 0000 0010 binary (02 hex). The program is back to OUT again. The program has proven good to that point.	Acca.02
Press RESET. This clears the previous breakpoints.	HEro1.X
Press 1 to select the Program Mode	1.

INSTRUCTION	DISPLAY
Press BR and enter 061A	061A br.
Press DO and enter 0600. You may have noticed the program ran up to the breakpoint and the counter segment in "H" was mementarily lit. Now you are in another loop. You could press RTI eight times and go back through the loop until (B) is 00. Again, since the branch is operating properly, it is easier to change (B) to 00 and continue.	061A 5d
Press ACCB, CHAN, and enter 00.	Accb.00
Press SS	061b 26
Press CC. The Z bit is set and the program should fall through the branch.	XXX1 XX
Press SS. It did.	061d 86
Press SS.	061F FE
Press ACCA. (A) is loaded correctly.	Acca.01
Press SS.	0622 8C
Press Index. This is DIGADD again. Although the program has just finished with the "H" display, single step has placed C10F in DIGADD. This happens to be the address that will be in DIGADD after DP goes out in the "C" display and should result in a branch back to START.	In.C10F
Press SS. Same conditional BRANCH.	0625 26
Press CC. Z is set and the program should fall through.	XXX1 XX
Press SS. It did.	0627 20
Press SS. Every instruction in the program has been run except for the conditional branch at 0625.	0600 bd
Press RESET. Clears the breakpoint at 061A.	HEro1.X
Press 1.	1._____
Press BR and enter 0622.	0622 br
Press DO and enter 0600.	0622 8C
Press Index. This time the program runs straight through until after (X) is loaded from DIGADD (at 001D) without an intervening single step or breakpoint. All segments were turned on and off in the "H" display. "I" display address C15F is in the index register as it should be.	In.C15F

INSTRUCTION	DISPLAY
Press SS. Conditional branch.	0625 26
Press CC. Z is clear and a branch to OUT should take place.	XXX0 XX
Press SS. It did.	0610 bd

The entire program has now been run.

SAMPLE PROGRAM

This sample program will give you practice entering programs and show the use of Monitor subroutines.

PROGRAM
Turn on and off each segment in sequence beginning at H display.
Uses monitor subroutines REDIS and OUTCH.
NOTE: One DP in each display is active.

```
0600   BD F64E   START   JSR     REDIS       Set up first display address
0603   86 01             LDA A   #$01        First segment code
0605   20 09             BRA     OUT
0607   F6 0FF3   SAME    LDA B   DIGADD+1    Fix display address
060A   CB 10             ADD B   #$10        For next segment
060C   F7 0FF3           STA B   DIGADD+1
060F   48                ASL A               Next segment code
0610   BD F7C8   OUT     JSR     OUTCH       Output segment
0613   CE 2F00           LDX     #$2F00      Time to wait
0616   09        WAIT    DEX
0617   26 FD             BNE     WAIT        Time out yet?
0619   16                TAB
061A   5D                TST B               Last segment this display?
061B   26 EA             BNE     SAME        Next segment
061d   86 01             LDA A   #$01        Reset segment code
061F   FE 0FF2           LDX     DIGADD      Next display
0622   8C C10F           CPX     #$C10F      Last display yet?
0625   26 E9             BNE     OUT
0627   20 d7             BRA     START       Do again
```

ROM SUBROUTINES

There are a number of subroutines in ROM which will save you time and memory when you write programs. These subroutines are described here and used in sample programs.

Note that it is best to run all subroutines in Machine Language; in Robot Language, the CPU can take 6 to 8 seconds to process one keypress using the INCH subroutine. If you commonly program in Robot Language, employ the Change to Machine Language 83 command before the BD XX XX Jump to Subroutine, and the Return to Robot Language (or SWI) 3F after the return from subroutine.

Since this "Programming Techniques" portion of the Manual presumes that you are familiar with 6800 machine language programming, the example programs will not have as much detailed explanation as was given in previous sections. We chose the memory locations for the programs with the intention of avoiding the destruction of one program by entering another on top of it.

INCH SUBROUTINE (F777)

INCH is a routine that processes the signals from the keyboard to filter out keybounce. It does this by going through a loop 32 times (20 hex) when it senses either a key close (press) or a key open (not pressed). The routine must receive the same key signal 32 times (about 9 ms, or about 5 seconds in Robot Language) before it exits the loop and accepts the input.

INCH is automatically used when the CPU is in the Executive, Program, and Repeat modes, and when needed in the Utility Mode. You must call INCH for programs where you wish to use the keyboard as an input source during the time that the program is running.

Program: This program positions the Robot's head in one of three places, depending on whether you press key 1, 2, or 3. This program will only run in Machine Language (Program Mode) since the ninth command "3F" will not be accepted if the CPU is in the Robot Language. The program goes immediately to INCH, and returns only when it accepts a keypress. Then it processes the number represented by the keypress. The program goes back and forth between Machine Language and Robot Language for rapid execution and programming flexibility. You may begin entering.

ENTER	COMMENTS
1	
A	
0300	
BD F777	JSR INCH
81 03	CMPA 03
27 0A	BEQ (move to 50)
81 01	CMPA 01
27 0D	BEQ (move to 75)
81 02	CMPA 02
27 10	BEQ (move to 62)
20 EF	BRA To Start
3F	SWI (Robot Language)
C3 D4 50	Move Head 50
83	Change to Machine Language
20 E8	BRA To Start
3F	SWI (Robot Language)
C3 D4 75	Move Head to 75
83	Change to Machine Language
20 E1	BRA To Start
3F	SWI (Robot Language)
C3 D4 62	Move Head to 62
83	Change to Machine Language
20 DA	BRA To Start
*	

*To run a program after you enter it, press RESET, A or 1 (to select either Robot or Machine Language), DO, and the beginning address.

OUTCH SUBROUTINE (F7C8)

OUTCH takes the value of the number stored in accumulator A and uses the binary representation of the number to designate which segments of a display location (character) to light. The eight-bit binary number can represent all eight segments of the display location, including the decimal point. Then OUTCH uses the number to actually light the specified segments of one of the display locations. The location is stored at DIGADD (0FF2 & 0FF3) in memory. OUTCH reads DIGADD, stores the "character" there, and then updates DIGADD so that the next character will go in the next display location. NOTE: Since you can control every segment individually, you can produce a very wide variety of characters and shapes.

The representation of a single display location below shows how the binary number determines what segments will light. Each bit location controls one segment of the display. If the bit is "1" the segment is on, if the bit is "0" the segment is off.

From Figure 1, you can see that a capital "H" would have bits 1, 2, 3, 5, and 6 turned on (0011 0111), which is 37 in hex. Lower case "h" would be 0001 0111, or 17 in hex.

OUTCH is automatically used in the Modes of operation that show information on the display. You may use OUTCH if you wish to show numbers on the display during a program.

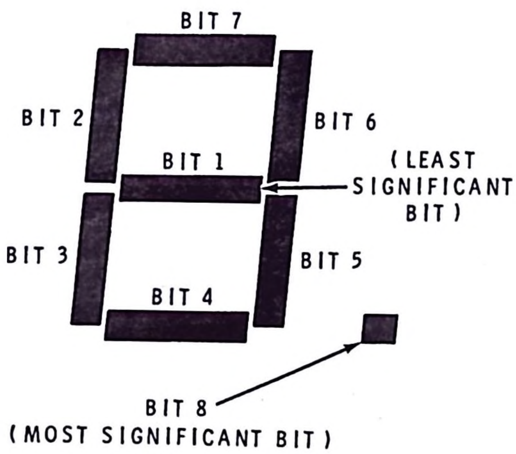


FIGURE 1

Program: This program uses the number "37" (second byte of the program) to create a capital H. After running the program, change the data to produce some other figure of your own choice. This program can run in either Robot or Machine Language.

ENTER	COMMENTS
1	
A	
0326	
86 37	LDAA with Hex 37
BD F7C8	JSR OUTCH
20 FE	BRA Nowhere

(See *, Page 20)

OUTHEX SUBROUTINE (F7B5)

OUTHEX also sends a character to the display. However, this character is one of the recognized hexadecimal numbers, 0 through F. The number sent is the number that is in the four least significant bits of accumulator A, and it is sent to the display location designated by DIGADD. DIGADD is then set to place the next character.

Program: This program is virtually the same as the OUTCH routine, except that it calls OUTHEX. Enter this at memory location 032D, or revise the OUTCH subroutine to get this one. Run it and note that it displays the least significant number "7" of the 37. Change the 37 to 57 and nothing appears to change. NOTE: Change the final command to 20 F9 (jump back to the beginning of the program), and the program will repeat the single character, filling the display as DIGADD changes. If you run this program in Robot Language, it will go more slowly and you can see the characters enter one at a time.

ENTER	COMMENTS
1	
A	
032D	
8637	LDAA With 37
BD F7B5	JSR OUTHEX
20 FE	BRA Nowhere

(See *, Page 20)

OUTBYT SUBROUTINE (F7AD)

OUTBYT is similar to OUTHEX, except that OUTBYT sends both numbers from accumulator A to be displayed in two adjacent display characters. Revise the OUTHEX subroutine by changing the **F7B5** to **F7AD** (at memory address 0331), and you will see both characters show up on the display.

CLRDIS SUBROUTINE (F65B)

CLRDIS outputs blanks to all of the display locations, and resets DIGADD to the left-most display location, ready for some new data to be displayed. CLRDIS is normally required before any of the OUT subroutines.

Program: See the program for OUTSTR.

OUTSTR SUBROUTINE (F7E5)

OUTSTR outputs a string of characters to the display, beginning with the left-most display location. The location for the first character is determined by the number in the index register when the subroutine is called (the index register is loaded into DIGADD to specify the first character's location). The character shape is "constructed" using binary, the same as for the OUTCH subroutine. The subroutine recognizes a character as the end of the string if that character has its decimal point lit. (Add hex 80 to a character to give it a decimal point. For example, 37="H" and B7="H.").

Program: This program clears the display and then outputs the string "HELLO!" The period in the exclamation point marks the end of the string for the subroutine. If you run this program in the Robot Language, you can see each individual letter created. If you change the last command from **20 FE** to **20 F2** (at memory address 0342) the program will repeatedly clear and write on the display; this happens so fast in Machine Language that you can't see it, so run it in Robot Language.

ENTER	COMMENTS
1	
A	
0335	
BD F65B	JSR CLRDIS
BD F7E5	JSR OUTSTR
37	Letter H
4F	Letter E
0E	Letter L
0E	Letter L
7E	Letter O
A0	Exclamation Point
20 FE	BRA Nowhere
*	

DSPLAY SUBROUTINE (F6F9)

DSPLAY is a subroutine that calls OUTBYT to output one, two, or three bytes from a specified location to the display. Normally, you would use OUTBYT without calling DSPLAY if you had only one byte, but you could use DSPLAY. The number of bytes is stored in accumulator B, the address of the first byte is loaded into the index register, and the data bytes are stored beginning at the given address.

Program: This program supplies its own data at the end of the program, but you could create a program to use data from the Robot's own sensors or memory. In such an application, you might want to call one, two, or three bytes of data to display, depending upon what you were sampling at each instance. This program calls REDIS to make sure that each display operation begins at the left-most character on the display.

ENTER	COMMENTS
1	
A	
0345	
BD F64E	JSR REDIS
CE 0352	LDX from memory 0352
C6 03	LDAB 3
BD F6F9	JSR DSPLAY
20 FE	BRA Nowhere
DE	Data
BB	Data
1E	Data
*	

*To run a program after you enter it, press RESET, A or 1 (to select either Robot or Machine Language), DO, and the beginning address.

IHB SUBROUTINE (F796)

IHB uses INCH and OUTHEX to monitor the keyboard and accept the input from the first two key strokes it identifies. It does two things with this input. First, it shows the two numbers on the display. Second, it stores those same numbers in accumulator A. This is the subroutine that allows you to enter a program and check your entry at the same time.

Program: This program is useful as well as instructive. It uses IHB twice to let you enter any address in memory (and shows the address on the display). Then the program goes to that location and speaks whatever it finds there until it reaches a BF or an FF. When it reaches an FF, the program returns to the beginning and waits for another entry. This program is useful if you wish to be able to call up a number of speeches (from either ROM or RAM). It is also intriguing to hear how regular (non-speech) programs sound.

ENTER	COMMENTS
1	
A	
0360	
BD F65B	JSR CLRDIS
BD F796	JSR IHB
B7 0371	STAA 0371
BD F796	JSR IHB
B7 0372	STAA in 0372
3F	SWI (Robot Language)
72 0101	SPEAK
83	Change to Machine Language
20 EA	BRA Back to Start

(See *, Page 22)

REDIS SUBROUTINE (F64E)

REDIS is a short routine to reset the address at DIGADD to the left-most display position. It is useful when some previous output operation to the display may have left DIGADD with the wrong address. REDIS is used in the program for DSPLAY.

This section of the Manual has dealt with programming aids (subroutines) available in the 6800 Machine Language. Continue to the next section for the Robot Language commands available through the Robot's interpreter.

ROBOT INTERPRETER

Since the CPU is a dedicated part of the Robot, it could work more efficiently if it had a special set of instructions that helped tailor its inputs and outputs to the Robot's logical and mechanical needs. This could allow special commands that could provide direct control for motors, sensors, speech, display, conditional branching, and other operations that are unique to the Robot. The internal "program" that provides such special tailored commands is called the "interpreter," and the language you can use with it is called "Robot Language."

You can enter the Robot Language (Repeat Mode) from the Executive ("ready") Mode by pressing the "A" (or Repeat) key. If you are already operating, but under the Machine Language (Program Mode), you must use the opcode **3F, software interrupt**, to change to Robot Language. If you do not, the program will "crash" since the CPU cannot handle Robot Language without the interpreter. However, the interpreter will handle all of the Machine Language commands, as well as the Robot Language commands. But this takes much longer than it would in Machine Language, since it must determine if each command is Machine or Robot Language before implementing it.

INTERPRETER TABLE

The following table lists the interpreter commands and gives both the hexadecimal command itself and the form for entering that command.

COMMAND	FORM*	TITLE
02	02	Abort Drive Motor
03	03	Abort Steering Motor
04	04	Abort Arm Motors
05	05	Abort Speech
1C	1C OO	Branch if Base Busy
1D	1D OO	Branch if Steering Busy
1E	1E OO	Branch if Arm Busy
1F	1F OO	Branch if Speech Busy
21	21	Zero
3A	3A	Return to Executive ("ready")
41	41	Enable Light Detector
42	42	Enable Sound Detector
45	45	Enable Ultrasonic Ranging
4B	4B	Enable Motion Detector
4E	4E	Enable Display
51	51	Disable Light Detector
52	52	Disable Sound Detector
55	55	Disable Motion Detector
5B	5B	Disable Ultrasonic Ranging
5E	5E	Disable Display
61	61 OO	Speak, Continue (index)
62	62 OO	Speak, Wait (index)
71	71 MM MM	Speak, Continue (extended)
72	72 MM MM	Speak, Wait (extended)
83	83	Change to Machine Language (3F changes back to robot)
87	87 XX XX	Sleep (immediate)
8F	8F XX XX	Pause (immediate)
BF	BF MM MM	Jump when Speaking (extended)
C3	C3 SS XX	Motor Move, wait abs (immediate)
CC	CC SS XX	Motor Move, continue abs (immediate)
D3	D3 SS XX	Motor Move, wait rel (immediate)
DC	DC SS XX	Motor Move, continue rel (immediate)
E3	E3 OO	Motor Move, wait abs (index)
EC	EC OO	Motor Move, continue abs (index)
F3	F3 MM MM	Motor Move, wait abs (extended)
FC	FC MM MM	Motor Move, continue abs (extended)
FD	FD **	Motors, Move All abs (immediate)

*The entire command word consists of these hex numbers. The second and third bytes provide data about the command. The different letters indicate what type of data is required.

XX= distance, position, time, etc.
SS= select motor, speed, direction
MM= memory address
OO= offset number.

**Seven motor position bytes, giving absolute position for these motors, in order: extend, shoulder, rotate, pivot, gripper, head, and steering (e.g., FD 98 86 93 A5 75 C2 93 runs all motors to end).

MOTOR CONTROL COMMANDS

The basic motor control word consists of the opcode followed by two bytes that specify the motor instructions. Those two bytes are formed as follows:

Byte 2	Byte 3
hex no. = S S	hex no. = X X
binary = $\overbrace{\text{mmms}}\,\overbrace{\text{sDdd}}$	binary = $\overbrace{\text{dddd}}\,\overbrace{\text{dddd}}$

In the binary numbers:

mmm are the motor select bits.
- 000 selects the drive motor.
- 001 selects the extend motor.
- 010 selects the shoulder motor.
- 011 selects the wrist rotate motor.
- 100 selects the wrist pivot motor.
- 101 selects the gripper motor.
- 110 selects the head motor.
- 111 selects the steering motor.

ss are the speed select bits:
- 01 = slow.
- 10 = medium.
- 11 = fast.

D is the bit that selects which way the motors run (except as noted). The position of each motor is stored as a hex number in a specified memory location. If D equals:

- 0 = The motor runs in the direction that causes the position number to increase.
- 1 = The motor runs in the direction that causes the position number to decrease.

dd dddd dddd bits determine how far (for relative commands), or to what position (for absolute commands), the motor turns.

NOTES:

1. Some motors (such as the head motor) may have trouble starting up in the "fast" speed. To overcome this, start the motor in slow for three units of travel. Then run in medium for three more units before going to fast for the rest of the distance.

2. The interpreter will not allow you to drive more than one arm motor at any one time. If you try, the interpreter will drive one motor at a time, while others wait for their turn.

3. The "move relative" commands only tell a motor what direction to go and how far to go. This means that a command could tell a motor to go beyond its normal limit. If the motor position number (see "Data In Memory," Page 32) goes to its low limit, the motor would reach the limit, reverse, and go all of the way to the other end. If the motor position number reaches its high limit, the motor will simply stop.

Example: The last two bytes for three commands are shown below. The three commands "ramp" the head motor up to speed, as described in Note 1.

Byte 2	Byte 3		Hex #'s	Meaning
1100 1000	0000 0011	=	C8 03	= Head, Slow, CCW, 3 Units
1101 0000	0000 0011	=	D0 03	= Head, Medium, CCW, 3 Units
1101 1000	0000 1001	=	D8 09	= Head, Fast, CCW, 9 Units

Opcode **C3, move wait**, causes the motor to run while the rest of the program waits until the motor is finished. **CC, move continue**, starts the motor and continues with the rest of the program. With both of these commands, the motor turns to the absolute position indicated by the lowercase "d" bits. The two bits provided in byte 2 are only needed for the Robot drive motor, to record longer distances. The acceptable values for the other motors are listed in the "Data in Memory," Page 32). Note that the direction bit "D" is ignored, since the motor is given an exact destination.

Opcodes D3 and DC are very similar to C3 and CC, except that they use relative positioning of the motors instead of absolute. With this, a 05 in the last byte would cause the motor to move five steps CW (or CCW, as the "D" bit dictated) instead of going to the 05 position for that motor.

Opcodes E3 and EC differ in that they use an offset number for the second and third bytes of the program word. These opcodes use absolute positioning, and the position byte is in a memory location that is determined by the offset number and the index register. The index register can be modified within the program to change the motor movement as the program dictates.

Opcodes F3 and FC are written with a memory location where byte 2 of the motor word is stored. The CPU uses this address to find both bytes, since byte 3 is located immediately after byte 2. Since the motor data is in memory, you can write the program to set or modify the motor's running parameters. Note that these opcodes also position the motor to an absolute point rather than running it in a relative direction and distance.

Opcode FD, move all, is a special command that allows you to position the Robot's seven arm, head, and steering motors just as you want them. These absolute position commands are executed in order, beginning with the extend motor and ending with the steering motor.

SPEAK

The basic speak command word consists of the opcode followed by either one or two bytes telling where to find the data (phoneme listing). Like the motor commands, the speak commands can be either **speak continue** (**61** and **71**) or **speak wait** (**62** and **72**). And they can tell where the phoneme string is by using either extended addressing (giving the actual memory address) or using the index register.

Program: This sample uses the speak wait command while the Robot says it can talk. Then it uses the speak continue command to have the Robot talk about turning its head while it is doing just that.

ENTER	COMMENTS
A	
A	
0400	
72 FA93	Speak wait, a phrase from memory
72 0411	Speak wait, phonemes at 0411
71 FADA	Speak continue, phrase from memory
D3 C815	Move wait, head 15 units CW
D3 CC15	Move wait, head 15 units CCW
20FE	Wait here
2F	Phoneme
00	Phoneme
0D	Phoneme
1E	Phoneme
FF	End of phonemes
*	

*To run a program after you enter it, press RESET, A or 1 (to select either Robot or Machine Language), DO, and the beginning address.

JUMP WHEN SPEAKING

Opcode **BF**, **jump when speaking**, is a jump command that is only obeyed when the CPU is in the speak portion of the program. The command word consists of the opcode, followed by the two-byte memory address, telling the CPU where to jump. The data at the new address will be treated as phonemes, until an FF or another BF is reached. Then the CPU will return (still in the speak operation) to the step right after the jump if speaking command. Upon returning, it may find either more phonemes, another BF, or another FF to terminate the speech operation. Notice that the only two nonphoneme commands recognized by the CPU during speech are the BF and the FF.

Note that there is a limit to the use of BF jumps. The CPU can only make eight jumps without an FF; then the program will crash. However, if each BF branch has its own FF at the end, you can use any amount you need.

Program: This program uses the BF command to string together nine of the phrases in memory. Since each phrase has its own FF, no problem occurs. Then the program, makes the Robot laugh, using a BF jump into an endless loop. Since this does not provide any FF's, the program crashes after eight ha-ha's. Check your program after the crash, some of the commands may have been changed near the entry to the loop.

ENTER	COMMENTS
A	
A	
0420	
72 0423	Speak command
BF FAB4	1st jump
BF FA93	2nd jump
BF FAAA	3rd jump
BF FAC0	4th jump
BF FADA	5th jump
BF FAF1	6th jump
BF FB09	7th jump
BF FC9D	8th jump
BF FD46	9th jump
DB	"Ha-ha" phonemes in...
D5	an endless loop...

ENTER	COMMENTS
C8	without any FF's
9B	
95	
88	
1B	
15	
08	
03	
BF 043E	The jump back to "Ha-ha"

(See *, Page 26)

ABORT

Each abort command consists of only the opcode. It is applied to devices which, once started, will remain running for some time, even while other operations may be taking place. These are the motors and the speech option. the abort command turns off the device immediately, without waiting for it to complete the programmed action. Normally, the abort is used with a conditional branch statement to stop some action if a particular condition exists. (One example is to use the ultrasonic ranging to "abort" the drive motor if the Robot gets too close to an obstruction).

The abort command is not useful when you use speak wait or move wait type commands because the CPU does not go on to read the abort commands until after the move wait is already completed. Note that **04, abort arm**, is used for the head, shoulder, extend, rotate, pivot, and gripper motors.

Program: This program tells the Robot to listen for a loud sound (a handclap), and drive forward when it hears one. While moving forward, the Robot listens for another loud sound and aborts the drive motor when the sound is heard. NOTE: If the noise sensor is too sensitive, noise from the Robot's own motor might stop the motor. To adjust the Robot's sensitivity to noise, change the value of the 30 in the step **8130** (lower number, more sensitive) at memory location 0499. Or, you may adjust the "Sound" control on the sense board. The program returns to the beginning after a stop, so you can start and stop it as often as you like.

If you intend to use this program for demonstrations, you may wish to begin the program with a command to turn the drive wheel somewhat, so the Robot will go in circles. Run it without the charger, so the cord will not become tangled. After you have adjusted the sensitivity (and perhaps the pause lengths in "Start" and "Stop") you can make the Robot respond by commanding "Start!" and "Stop!"

ENTER	COMMENTS
A	
A	
0480	
	SENSOR ON, WAIT TO STABILIZE
42	Enable Sound Sensor
8F 0020	Pause (don't sense) for 2 seconds
	START
8D 0E	BSR "Listen"
CC 08FF	Motor Move Continue (drive forward slow, until FF)
8F 0010	Pause (don't sense) for 1 second
	STOP
8D 06	BSR "Listen"
02	Abort Drive Motor
8F 0010	Pause (don't sense) for 1 second
20 F0	BRA to "Start"
	LISTEN
B6 C240	LDAA from C240 (sense port)
0C	CLC
81 30	CMPA to 30
25 F8	BCS (if A is less than 30, branch back to "Listen")
39	RTS (if greater, go back to "Start" or "Stop")

(See *, Page 26)

ENABLE AND DISABLE

The enable and disable commands provide the controls for the light, sound, and motion detectors, the ultrasonic ranging, and the display ("Abort" is a different command, for speech and motors). The command word consists of only the opcode. This will turn on (enable) or turn off (disable) the selected device. Subsequent commands transfer data to or from the enabled devices.

Program: The preceding program, used to demonstrate the abort commands, also used the enable for the sound detector.

BRANCH IF BUSY

The branch command words contain an opcode byte plus an offset byte, telling the CPU where to go if the branch is executed. Note that these are conditional commands, the branch will occur only if the specified item(s) are busy. As in the case of the abort commands, the reference to "arm" here includes the head along with all of the other shoulder and arm motors. If any one of them is busy, the branch will occur.

This command is useful if you wish to cause or prevent a specific occurrence, and coordinate this with some other Robot action. An example would be to cause the light detector device to operate whenever the head is rotating.

Program: In this example, the Robot turns its head from position 02 to BE, and back, and turns on the ultrasonic ranging. Then it uses the **1E, branch if arm busy,** to repeatedly read and display the range and number of hits from the sonar (range is displayed on the left two characters, number of hits is on the two right characters). When the head stops, the program falls through the 1E command to the 39(RTS), and the display stops, showing the final count. NOTE: Change 20 FE to **20 EE** for continuous operation; change 01 at memory location 0463 to **83** for faster (Machine Language) operation of the scan and display, and change the other 01 at memory location 0478 to **3F** to get Robot Language operation in the rest of the program.

ENTER	COMMENTS
A	
A	
0450	
	SET COUNT TO ZERO
86 00	LDAA with zero
97 10	STAA at 0010
97 11	STAA at 0011
	TURN HEAD RIGHT & SCAN
CC D0 BE	Move Head to BE
8D 07	BSR to Sonar
	TURN HEAD LEFT & SCAN
CC D0 02	Move Head to 02
8D 02	BSR to Sonar
20 FE	BRA Beginning
	SONAR
45	Enable Sonar
01	NO-OP
96 11	LDAA Sonar Range
BD F64E	JSR REDIS
BD F7AD	JSR OUTBYT (range)
4F	CLRA
BD F7C8	JSR OUTCH (blank)
BD F7C8	JSR OUTCH (blank)
96 10	LDAA Sonar Hits
BD F7AD	JSR OUTBYT (hits)
01	NO-OP
1E E8	Branch If Arm Busy (to Sonar)
39	RTS

MISCELLANEOUS REMAINING COMMANDS

There are a few remaining interpreter commands which do not fit into groups as the rest of them have. These commands will be discussed here, one at a time. A single program at the end of this section will demonstrate all of these commands.

Zero

Opcode **21, zero,** is used in the utility mode initializing operation (the operation that occurs when you press 31 from the Executive Mode). Zero drives each motor in turn to the end of its travel. Then it positions some of them to mid-position. See "Initializing" in the User's Manual for further information about the actions that take place.

Return To Executive Mode

Opcode **3A, return to exec**, stops the program and takes the Robot from the Robot Language operation to the Executive ("ready") Mode. This is the most effective way of causing a program to terminate automatically at some specific desired point.

Change To Machine Language

Opcode **83, change to machine language**, causes the CPU to leave the interpreter and process program commands only in 6800 Machine Language. You may wish to use this command if you have a section of program that runs too slowly, and if that part of the program is in Machine Language. Since it takes very little time to switch between Machine and Robot Languages, you are free to experiment with switching back and forth within a program in the interests of improving running speed. The Machine Language software interrupt, 3F, is the counterpart to this command, and switches the CPU back into the interpreter (Robot Language).

Sleep

Opcode **87, sleep**, puts the Robot into a battery-saving sleep, with all unnecessary circuitry shut off. Note however, that the Robot will only obey the sleep command if you have the Sleep/Normal switch in the Sleep position. This switch is on the experimental board, along with the Abort button.

The entire sleep word includes the opcode and two bytes. The two bytes record the number (in hex) of time periods the Robot is to sleep. Since each time period is 10 seconds long, the command 87 FF FF would put your Robot to sleep for 655,350 seconds, or just over 7 days 14 hours. Immediately after awakening from a sleep, the CPU processes the next instruction in the program.

Pause

The opcode **8F, pause**, is similar to the sleep command in that it stops all apparent activity during the period dictated by the next two bytes. The pause command, however, does not shut off any circuitry and uses 1/16 second time periods instead of 10 second. Thus, the longest pause, an 8F FF FF, would last only about 68 minutes and 16 seconds.

Program:

The following program uses all of the "miscellaneous" commands to evaluate the most efficient way of wasting time. It starts in the Robot Language and uses zero to initialize the Robot. After that, it goes into a "do" loop, counting back from 6000 to use a little time (about 41 seconds) in the Robot Language, while showing "do" on the display. When it is done, it flashes "done" on the display. Then the 83 command switches to the faster Machine Language and the CPU performs the same "do" loop (in about 0.3 second), showing how much faster the Machine Language is at wasting time. Again, it uses the display to indicate when it is done. The 3F command switches back to Robot Language. Since the 8F pause command is an easier way to waste time, the Robot pauses for 10 seconds, and then shows when it is done. Finally, it goes to sleep (87) for 30 seconds. Note that the display is turned off during sleep, even though the program tries to leave the word "do" on it. After the nap, the Robot thanks you for the rest and reminds you to reset the sleep switch to Normal. When you acknowledge this, the 3A sends the Robot back to the Executive Mode.

To run the program, set the Sleep switch to the "Sleep" position. Then press the keys to run the program in the Robot Language. If your Robot skips the sleep portion of the program, it is because the sleep switch wasn't set.

MEMORY ADDRESS	ENTER	COMMENTS
	A	
	A	
	0700	
		DO THE "ZERO" OPERATION
0700	21	ZERO
		FIRST DO LOOP (ROBOT LANGUAGE)
0701	8D 3E	BSR to "Do"
0703	8D 45	BSR to "Delay"
0705	83	Change to Machine Language
0706	8D 49	BSR to "Done"
0708	8D 40	BSR to "Delay"
		SECOND DO LOOP (MACHINE LANGUAGE)
070A	8D 35	BSR to "Do"
070C	8D 3C	BSR to "Delay"
070E	8D 41	BSR to "Done"
0710	8D 38	BSR to "Delay"
		WRITE "PAUSE"
0712	BD F65B	JSR CLRDIS
0715	BD F7E5	JSR OUTSTR
0718	67	Letter "P"
0719	77	Letter "A"
071A	3E	Letter "U"
071B	5B	Letter "S"
071C	CF	Letter "E"
071D	3F	SWI to Robot Language
071E	72 075C	SPEAK "Pause"
0721	8F 00A0	PAUSE for 10 seconds
0724	BD F65B	CLRDIS
0727	BD F7E5	OUTSTR
072A	5B	Letter "S"
072B	0E	Letter "L"
072C	4F	Letter "E"
072D	4F	Letter "E"
072E	E7	Letter "P"
072F	72 076F	SPEAK "Sleep"
0732	87 0003	SLEEP for 30 seconds
0735	72 0783	SPEAK "Thank You"
0738	83	Change to Machine Language
0739	BD F65B	JSR CLRDIS
073C	BD F777	JSR INCH
073F	3F	SWI to Robot Language
0740	3A	Return to Executive Mode

MEMORY ADDRESS	ENTER	COMMENTS
		WRITE "DO"
0741	BD F65B	JSR CLRDIS
0744	BD F7E5	JSR OUTSTR
0747	3D	Letter "D"
0748	9D	Letter "O"
0749	39	RTS
		DELAY
074A	CE 6000 .	LDX with 6000
074D	09	DEX
074E	26 FD	BNE to 074D
0750	39	RTS
		WRITE "DONE"
0751	BD F65B	JSR CLRDIS
0754	BD F7E5	JSR OUTSTR
0757	3D	Letter "D"
0758	1D	Letter "O"
0759	15	Letter "N"
075A	CF	Letter "E"
075B	39	RTS
		PHONEMES "PAUSE"
075C	15	AH1
075D	00	EH3
075E	09	I3
075F	29	Y
0760	2D	W
0761	0B	I1
0762	09	I3
0763	18	L
0764	65	P
0765	7D	AW
0766	70	AW2
0767	52	Z
0768	3E	PA1
0769	0D	N
076A	15	AH1
076B	23	UH3
076C	37	U1
076D	3F	STOP
076E	FF	END
		PHONEMES "SLEEP"
076F	4D	N
0770	55	AH1
0771	63	UH3
0772	77	U1

Heathkit®

MEMORY ADDRESS	ENTER	COMMENTS
0773	43	PA0
0774	15	AH1
0775	00	EH3
0776	09	I3
0777	29	Y
0778	2D	W
0779	0B	I1
077A	09	I3
077B	18	L
077C	1F	S
077D	18	L
077E	3C	E1
077F	22	Y1
0780	25	P
0781	3F	STOP
0782	FF	END
		PHONEMES "THANKYOU"
0783	79	TH
0784	6F	AE1
0785	80	EH3
0786	94	NG
0787	99	K
0788	22	Y1
0789	36	IU
078A	37	U1
078B	37	U1
078C	1D	F
078D	34	02
078E	34	02
078F	2B	R
0790	38	THV
0791	32	UH1
0892	31	UH2
0793	03	PA0
0794	6B	R
0795	42	EH1
0796	40	EH3
0797	5F	S
0798	6A	T
0799	7E	PA1
079A	25	P
079B	18	L
079C	3C	E1
079D	21	AY
079E	12	Z
079F	2B	R
07A0	3C	E1
07A1	1F	S

MEMORY ADDRESS	ENTER	COMMENTS
07A2	02	EH1
07A3	00	EH3
07A4	2A	T
07A5	0C	M
07A6	15	AH1
07A7	00	EH3
07A8	09	I3
07A9	29	Y
07AA	1F	S
07AB	18	L
07AC	3C	E1
07AD	22	Y1
07AE	25	P
07AF	1F	S
07B0	2D	W
07B1	0B	I1
07B2	09	I3
07B3	2A	T
07B4	10	CH
07B5	03	PA0
07B6	2F	AE1
07B7	00	EH3
07B8	0D	N
07B9	1E	D
07BA	78	THV
07BB	40	EH3
07BC	42	EH1
07BD	4D	N
07BE	25	P
07BF	16	OO1
07C0	36	IU
07C1	11	SH
07C2	0C	M
07C3	15	AH1
07C4	00	EH3
07C5	09	I3
07C6	29	Y
07C7	12	Z
07C8	21	AY
07C9	0B	I1
07CA	2B	R
07CB	35	O1
07CC	37	U1
07CD	19	K
07CE	21	AY
07CF	29	Y
07D0	3F	STOP
07D1	FF	RETURN

"PROGRAM-EXTERNAL" DATA AND INTERRUPT LOCATIONS

INTRODUCTION

The least sophisticated programs tell the Robot to do a series of things, always in order, without variation. These programs can be called self-contained because they run without requiring data that is external to the program. More sophisticated programs call for data from outside the original program, and use that data to affect how the Robot acts. This data is added from the Robot's sensors or is created by the CPU and stored in some special place in memory. Typical types of data that may be "external" to your program are:

What time is it? (from the Real Time Clock)
How far to an obstruction? (from Ranging)
What is the position of the head? (from memory)
Is the light level above a given value? (from Light Sensor)

In addition, there are special inputs that can interrupt a running program, or start a program with the push of a single key. All of these special features that allow more creative programming are discussed here.

DATA IN MEMORY

The position of each motor is stored in a specific place in RAM. When you initialize the Robot after turning it on, you are making sure that the motor position and the stored "position information" agree. The following table shows where the information is stored in memory for each motor, and the range of the stored number.

RAM LOCATION	MOTOR	RANGE
0000	Extend	(full in) 00 to 98 (full out)
0001	Shoulder	(down) 00 to 86 (raised)
0002	Rotate	(CCW) 00 to 93 (CW)
0003	Pivot	(up) 00 to A5 (down)
0004	Gripper	(closed) 00 to 75 (open)
0005	Head	(CW) 00 to C2 (CW)
0006	Steering	(left turn) 00 to 93 (right turn)

Ultrasonic ranging stores the most recent range reading at 0011, and counts the number of "hits" (ranges measured) at 0010. Location 0EFC counts to FF and resets at a rate of 1024 Hz driven from the real time clock.

You can easily load the CPU with information from any of these points using the machine language command LDAA.

Program: In the program for the branch if busy commands, Page 28, you can see how the Robot examines the ultrasonic range and number of "hits" from 0010 and 0011.

USER DEFINED KEYS

If the Robot is in the Executive ("ready") Mode, the F, C, and 9 keys may serve a special purpose. Press any one of these keys and the CPU goes to a special location in RAM where three bytes of memory have been set aside. Normally, it does not find any instructions there, so it returns the Robot to the Executive Mode. However, if you place an instruction in that special place, the CPU will execute that instruction, in Robot Language.

Typically, you will enter a jump command there, with the address of a program you frequently use. When you press the key, the CPU will go to that special place in memory and then jump to the program, arriving in Robot Language. This allows you to simply and quickly run a favorite program (such as having the Robot stand guard at night) with a single key stroke.

KEY	MEMORY LOCATIONS
USER 1	0030, 0031, 0032.
USER 2	0033, 0034, 0035.
USER 3	0036, 0037, 0038.

These keys only serve this function when the Robot is in the Executive ("ready") Mode. Remember that the Robot automatically goes to the Robot Language from the user keys; if you wish to use the faster Machine Language, you must add the switch to Machine Language (83) to the program you will be running.

Program: From the executive mode, press A, A, 0030, 7E (the jump command), and the address of the beginning of a program you now have in memory. Press RESET. When you press USER 1, the program will run.

INTERRUPT VECTORS

Just as there are three user keys, there are three interrupt vectors. And, just like the user keys, each vector has a special three-byte place in memory where you can enter a jump to the interrupt routine (program). Normally, there is a simple return from subroutine (RTS) command there, so that nothing happens unless you wish. However, the interrupt vectors differ from the user keys in most other ways.

Motion Detect	0027, 0028, 0029
Trigger	002A, 002B, 002C
Experimental Board IRQ (low)	002D, 002E, 002F

First of all, the information stored at the interrupt memory locations is reset every time the CPU goes to the Executive Mode. In other words, every time you press RESET it wipes out the jumps you placed in the interrupt vectors. The most obvious way around this is to have your program itself place the jump instructions at the interrupts' special memory locations.

A second difference is that the interrupt vectors won't work unless you have cleared the interrupt mask (opcode 0E, CLI). Of course the CPU also resets the interrupt mask at every possible opportunity (every time you reset and every time you use an interrupt), so you must "clear" in the program just prior to using the interrupt.

Finally, the interrupt vectors won't work while the Robot is sleeping.

The interrupt routine itself must be in Machine Language, not Robot Language. This is true because, while the CPU is in the interrupt routine, the interrupt mask is set to prevent additional interrupts, **including help from the Robot Language interpreter.** For this reason you expressly want to use the interrupt routine to modify data in the program or stack counter and keep the rest of the activity in the program itself, where you can still use Robot Language.

The end of your interrupt routine is the Return From subroutine (RTS, opcode **39**), since the interrupt is treated like a subroutine. When the CPU does this, it returns to the original program at the spot where it was when the interrupt occurred. There it continues with that program, including any changes that were made by the interrupt program. We recommend that you do not use RTI (opcode 3B), unless you have enough programming expertise to understand all of the implications of its use on your program.

Program: This program uses the Trigger interrupt to change the Robot's speech pattern. Normally, the Robot will sit there muttering "nothing...nothing..." but if you pull the trigger on the Teaching Pendant, the Robot will come to life long enough to yell "Trigger!" Notice that the interrupt subroutine is kept very short and everything possible is placed in the program, where Robot Language is acceptable. Since this program is a bit long, a few headings are included to make it easier to understand.

MEMORY ADDRESS	ENTER	COMMENTS
	1	
	A	
	0500	
		PREPARE INTERRUPT VECTOR
0500	86 7E	LDAA
0502	B7 002A	STAA
0505	86 05	LDAA
0507	B7 002B	STAA
050A	86 3D	LDAA
050C	B7 002C	STAA
050F	0E	CLI
		PROGRAM BODY
0510	86 00	LDAA
0512	B7 0543	STAA Put zero into 0543.
0515	B6 0543	LDAA
0518	26 13	BNE If 0543 is not zero, the subroutine has changed it. Go to Speak "Trigger."
051A	3F	ROBOT LANGUAGE
051B	72 0522	SPEAK "NOTHING"
051E	83	MACHINE LANGUAGE
051F	7E 0515	JMP
		"NOTHING"
0522	0D	N
0523	23	UH
0524	39	TH
0525	39	TH
0526	09	I3
0527	29	Y
0528	14	NG
0529	3E	PA1
052A	3E	PA1
052B	3E	PA1
052C	FF	END OF SPEECH
		SPEAK "TRIGGER"
052D	3F	ROBOT LANGUAGE

MEMORY ADDRESS	ENTER	COMMENTS
052E	72 0535	SPEAK
0531	83	MACHINE LANGUAGE
0532	7E 0510	JMP back and reset 0543 to zero.
0535	AA	T
0536	AB	R
0537	89	I3
0538	A7	I
0539	9C	G
053A	BA	ER
053B	83	PA0
053C	FF	END OF SPEECH
		SUBROUTINE
053D	86 01	LDAA
053F	B7 0543	STAA Change the zero to a one, at 0543.
0542	39	RTS
0543		Data
*		

REAL TIME CLOCK DATA

There are a number of program applications that require the Robot to have some sense of actual time. You may want the program to turn some sense on or off, or perform some action, at a specific time. Typical applications include having the Robot act as an alarm clock in the morning, or become a house guard during the night hours. This section tells you how a program can interrogate the clock to get time information.

Refer to Figure 2 and its table for this discussion. The clock receives and sends time data on lines D0 through D3. Since these lines can carry only one number at a time, you must tell the clock which digit you want (units second, tens second, units minute, tens minute, units hour, tens hour, week of the month, units day, tens day, units month, tens month, units year, tens year). Which number you want is selected on lines A0 through A3 as shown in the table.

*To run a program after you enter it, press RESET, A or 1 (to select either Robot or Machine Language), DO, and the beginning address.

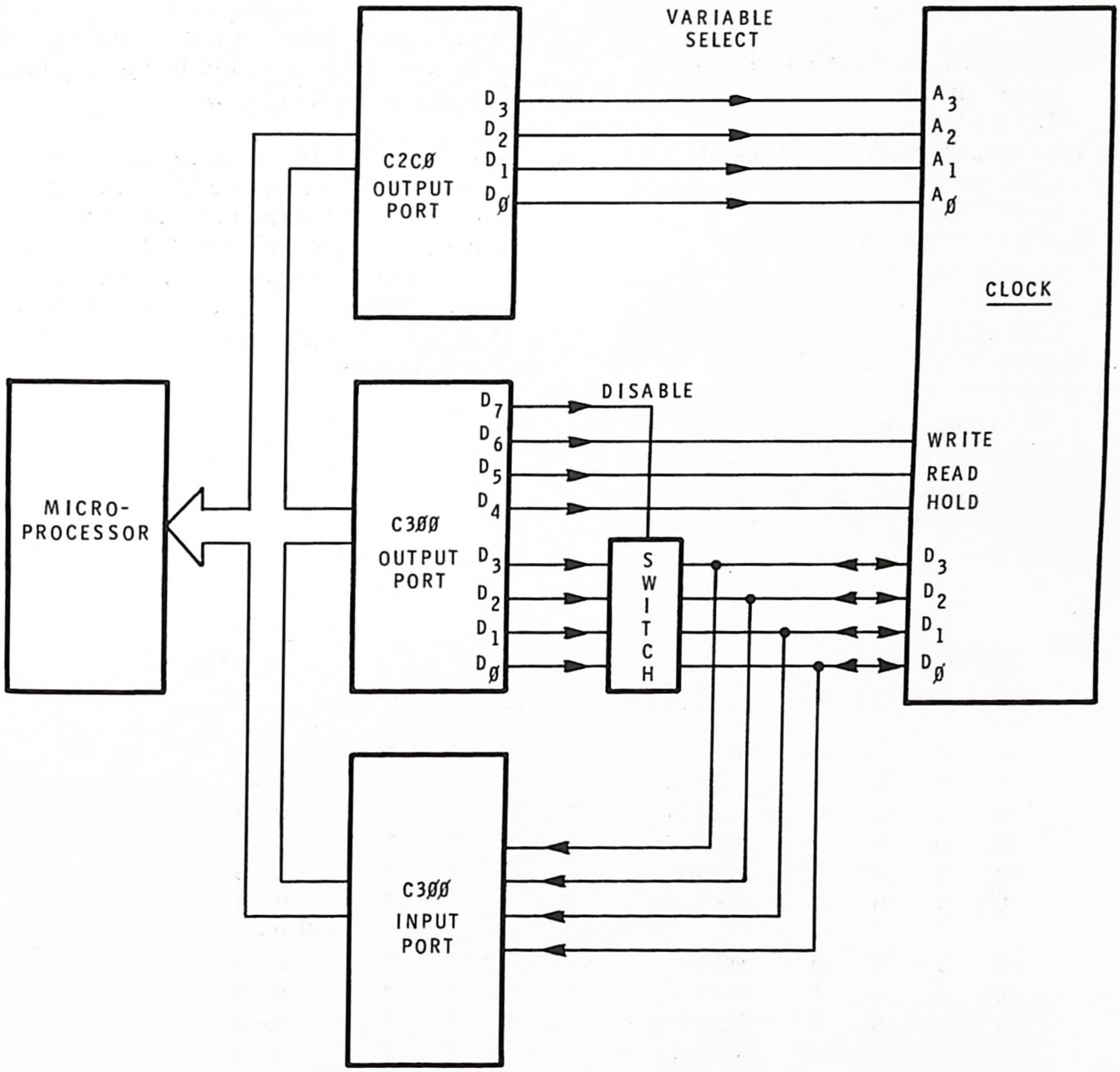

VARIABLE SELECT
C2CØ OUTPUT PORT
D3
D2
D1
DØ
A3
A2
A1
AØ
CLOCK
MICRO-PROCESSOR
C3ØØ OUTPUT PORT
D7
D6
D5
D4
D3
D2
D1
DØ
DISABLE
SWITCH
WRITE
READ
HOLD
D3
D2
D1
DØ
C3ØØ INPUT PORT

FIGURE 2

Program: This program will demonstrate how to call up and display the units figure for seconds. The program works best in Machine Language.

ENTER	COMMENTS
1	
A	
054A	
	TELL CLOCK WE WANT TO READ
86A0	LDAA with A0.
B7 C300	STAA at C300.
	TELL WHAT WE WANT TO READ
86 00	LDAA with 00.
B7 C2C0	STAA at C2C0.
	READ AND DISPLAY NUMBER
B6 C300	LDAA from C300.
BD F64E	JSR REDIS.
BD F7B5	JSR OUTHEX.
	WAIT ABOUT 1 SECOND, THEN REPEAT
CE 0300	LDX with 0300.
09	DEX.
26 FD	BNE.
20 EA	BRA to beginning.

(See *, on Page 35)

NOTES:

1. If you wanted to write to the clock instead of read from it, change memory address 054B from A0 to C0.

2. If you wish to sample the tens of seconds, change memory address 0550 from 00 to 01, as shown under "Select Inputs" in the table. Similarly, tens of hours would be 05, and the ones digit of years would be 0B.

3. Note that 7:00 P.M. would be represented (in the table under Data I/O) by a 07 in the HRS(1) output and a 40 (0100) in the HRS(10) column, due to the addition of the bit at D2 to represent P.M. In the 24-hour format, 7 P.M. (19:00) would become HRS(1)=09 (1001) and HRS(10)=09 (1001). The 1 in D3 signifies that the clock is in the 24-hour format.

SELECT INPUTS				VARIABLE	DATA I/O				RANGE
A3	A2	A1	A0	SELECTED	D3	D2	D1	D0	
0	0	0	0	SEC(1)	*	*	*	*	0--9
0	0	0	1	SEC(10)		*	*	*	0--5
0	0	1	0	MIN(1)	*	*	*	*	0--9
0	0	1	1	MIN(10)		*	*	*	0--5
0	1	0	0	HRS(1)	*	*	*	*	0--9
0	1	0	1	HRS(10)	+	+	*	*	0--1
									OR 0--2
0	1	1	0	WEEK		*	*	*	0--6
0	1	1	1	DAY(1)	*	*	*	*	0--9
1	0	0	0	DAY(10)		†	*	*	0--3
1	0	0	1	MO(1)	*	*	*	*	0--9
1	0	1	0	MO(10)				*	0--1
1	0	1	1	YR(1)	*	*	*	*	0--9
1	1	0	0	YR(10)	*	*	*	*	0--9

* May be either a 1 or a 0.
\+ D2 is 1 for pm, D2 is 0 for am.
D3 is 1 for 24 hr format, D3 is 0 for 12 hr format.
†D2 is 1 during 2nd month for leap year, D2 is 0 for other years.

Pseudo Random Number Generator

The real time clock also outputs a 1024 Hz signal, which increments a special location in memory (0EFC) 1024 times a second. That makes this location count from 00 to FF (to 256 in decimal) four times a second. If you read this location at a fairly slow and irregular rate, it will provide random (hex) numbers. Remember, however, that a program which called for numbers rapidly and regularly would produce non-random numbers. The calling for numbers must be relatively slow and irregular (such as is done by humans, in the following program).

Program: This program will provide you with a "random" number when you press the 0 key.

ENTER	COMMENTS
1	
A	
056A	
	SET UP
0E	CLI
86 AF	LDAA with AF.
B7 C300	STAA at C300 (clock port).
	CLEAR DISPLAY
BD F64E	JSR REDIS
	WAIT FOR 0 KEY
BD F777	JSR INCH
81 00	CMPA with 00.
26 F9	BNE to Wait For 0 Key.
	GET "RANDOM" NUMBER
B6 0EFC	LDAA from 0EFC,
BD F7AD	JSR OUTBYT.
20 EE	BRA to Clear Display.

SENSE PORTS

Ultrasonic range figures are stored in memory, but other senses are addressed at the ports indicated on the Port Map on fold-out from Page 119 of the Appendix. Read from these port addresses just as though they were memory.

Program: The "Abort" program on Page 27 is an example that reads from a sense port.

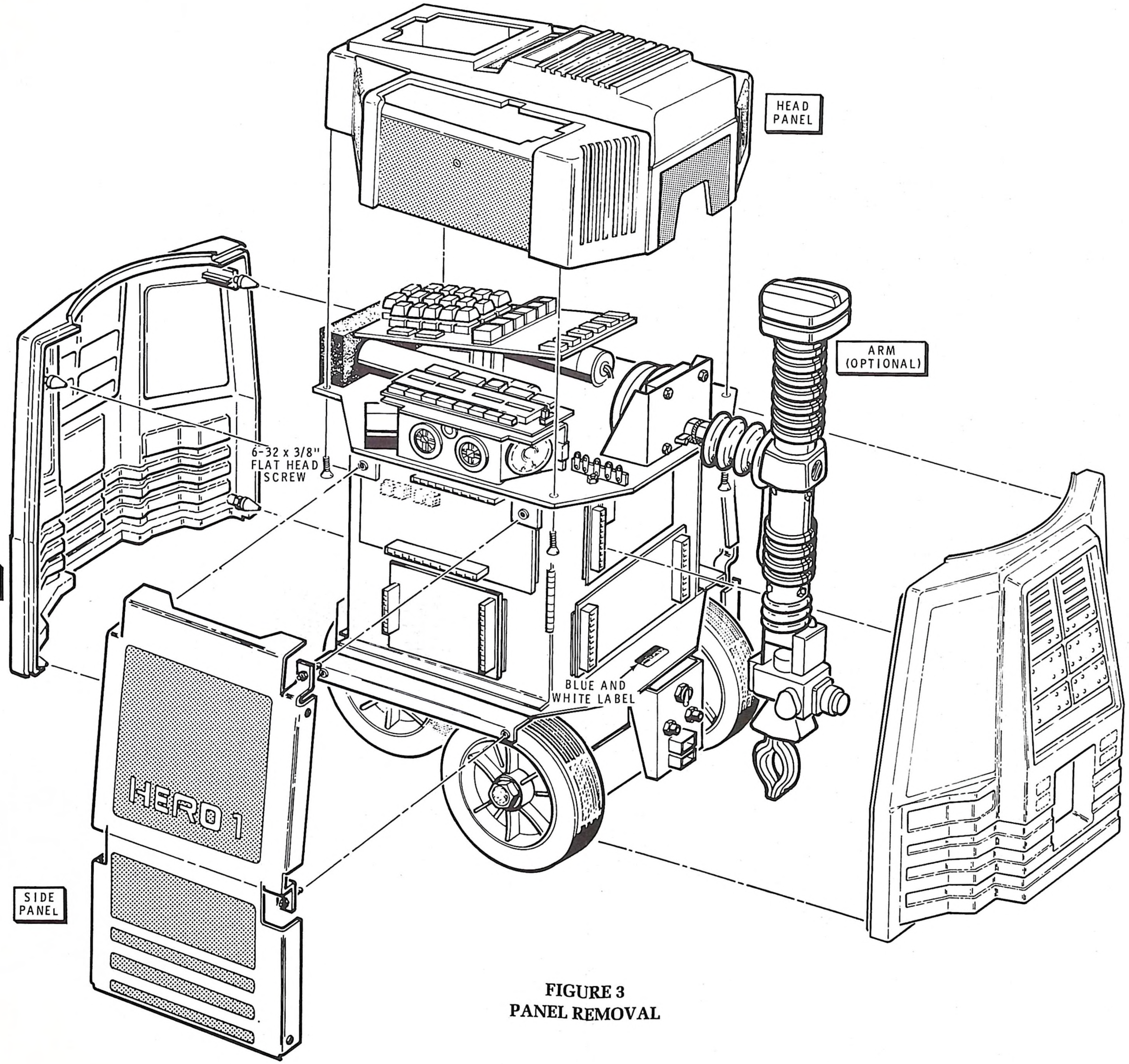


FIGURE 3
PANEL REMOVAL

DISASSEMBLY AND COMPONENT LOCATIONS

This section of the Manual describes the removal of body and head panels; and the location of batteries, fuses, main components, and the various plugs and connectors in the Hero Robot. Mechanical assemblies, such as the main drive, steering and head motors, and the arm assembly, are also included.

Large illustrations are bound into this Manual so they will not be lost in the future. They fold out so you can continue to see them even as you read other pages. If you need to remove or replace a component, check the appropriate illustration for identification of connectors, plugs, or hardware that may also have to be removed. Read the information that follows for additional hints and suggestions.

PANEL REMOVAL (FIGURE 3)

You can remove the head panel without removing body panels. Simply rotate the head to about 45° from straight ahead to gain access to the four 6-32 × 3/8" flat head screws. Remove the screws and lift the head straight upward.

Retainer pins on the front and back body panels are seated in grommets in the side panels. Therefore, the front and back panels must be removed before either side panel can be removed. Simply pull outward on the panel to unseat the retainers.

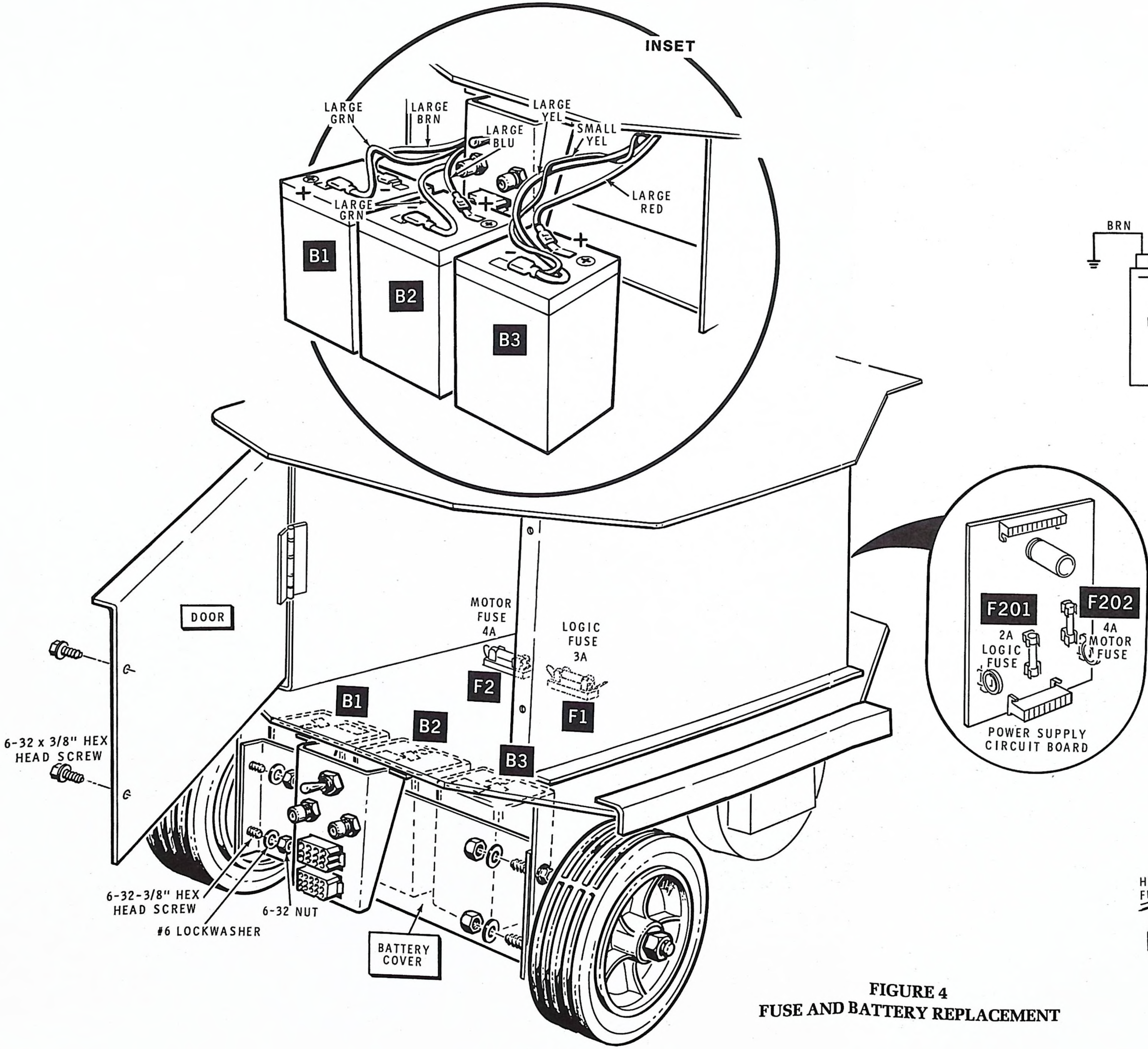
FUSE AND BATTERY LOCATIONS (FIGURE 4)

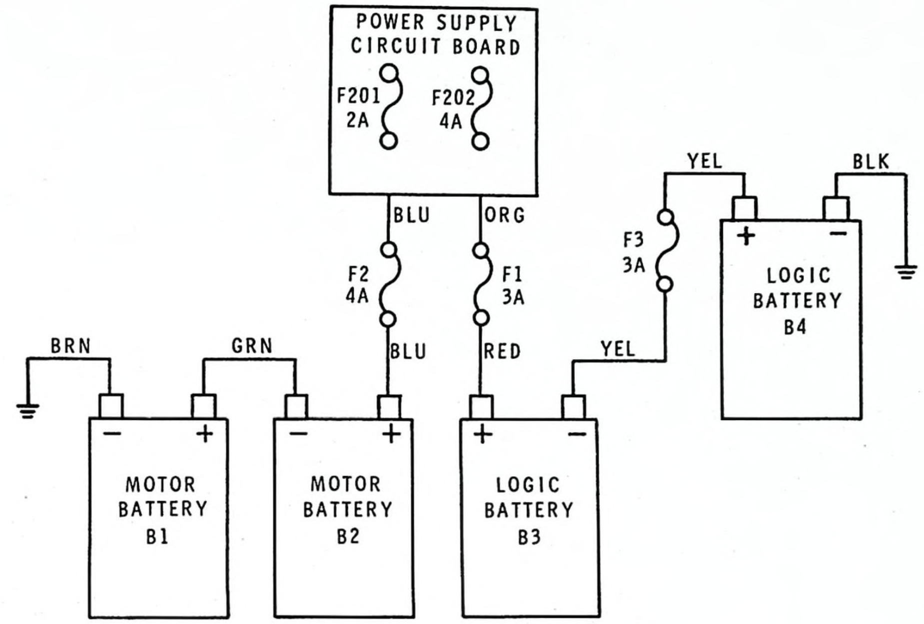
Five fuses are used in Hero. Two are located on the base plate inside the torso, one on the head base plate, and two on the power supply circuit board. Always replace a fuse with another of the same rating.

CAUTION: Disconnect the charger, and then remove the two fuses from inside Hero's torso and the fuse on the head base plate before you remove the batteries or other components from Hero.

Battery B4 is located on the head base plate. To replace this battery, remove the 6-32 nuts, lockwasher, and solder lug, and the battery retainer. Be sure to reconnect the yellow wire to the positive battery terminal and the black wire to the negative terminal, and reinstall the hardware as shown.

Batteries B1, B2, and B3 are located in the battery compartment behind the battery cover. Remove the 6-32 × 3/8" hardware and the battery cover for access to these batteries. Refer to the Inset drawing for the proper connections to the positive (+) and negative (–) battery terminals.





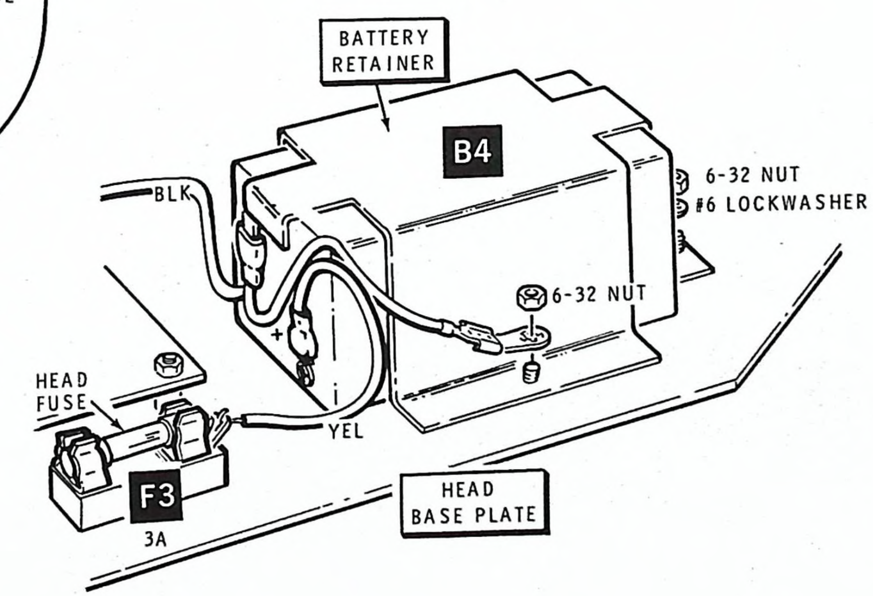


FIGURE 4
FUSE AND BATTERY REPLACEMENT

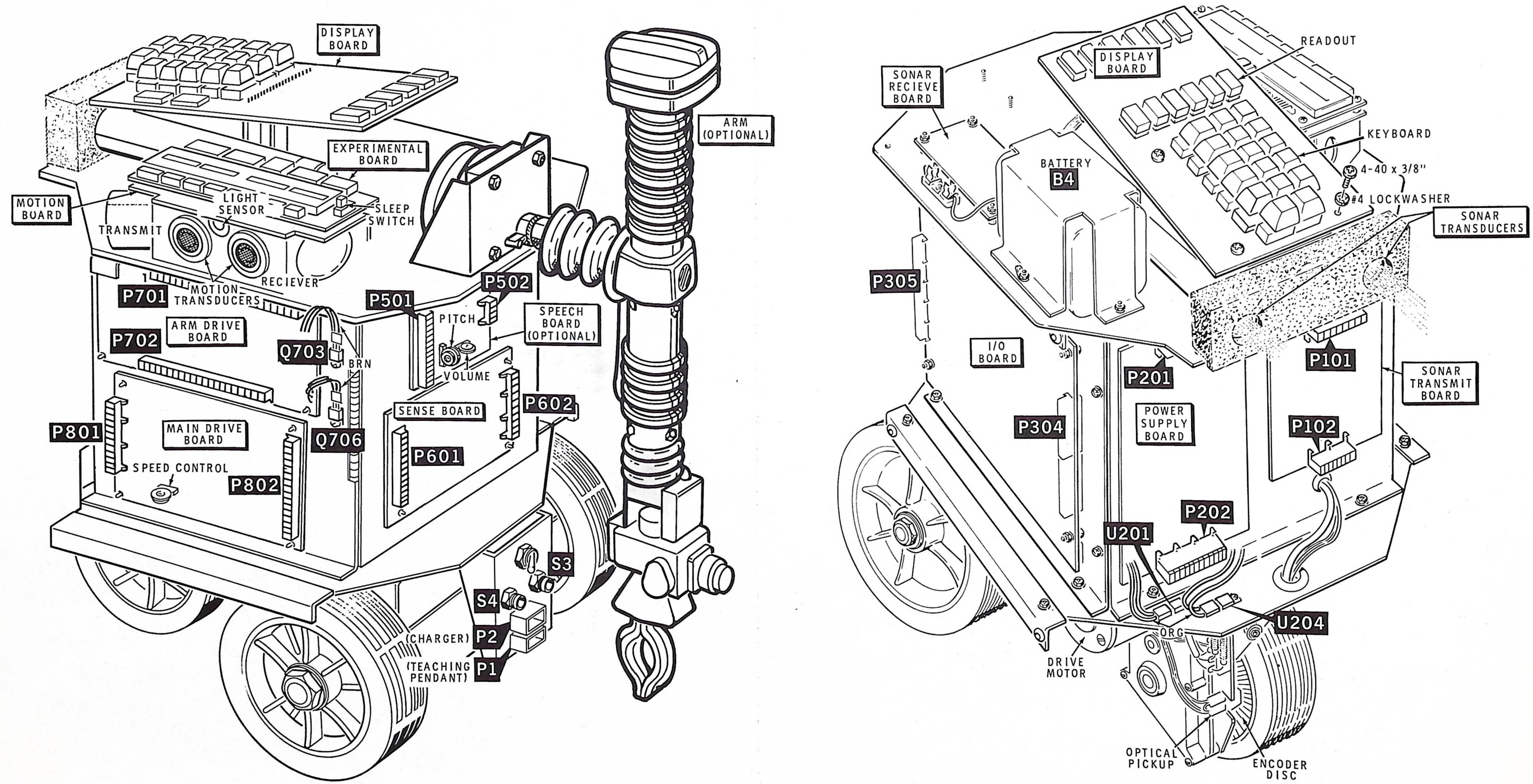


FIGURE 5
MAIN COMPONENT LOCATIONS

MAIN COMPONENT LOCATIONS (FIGURE 5)

If you must remove a circuit board from the torso panels, pry it off of its circuit board spacers and connectors very carefully to avoid breaking either the board or the spacers. Pry one end partly off of its pins and then pry the other end. If you remove the power supply board or the arm drive board, be sure to remove the connectors from the ICs or transistors that are mounted on the base plate or side panel. When you reinstall either of these boards, replace the connectors with the wire colors as shown.

The display board is mounted with 4-40 hardware and is connected with a 24-pin flat connector to the foil side of the circuit board. Be sure to reinstall the connector onto its pins before you remount the display board.

Hardware must be removed before you can remove the experimental board, the motion board, or the sonar receive board. Be sure to remount these boards in the same manner and with the same hardware that you removed.

PLUGS AND CONNECTORS (FIGURE 6)

Use this illustration to quickly locate plugs and connectors that may be referenced in the Troubleshooting and/or other sections of this Manual. If you remove a connector from its pins, be sure to reconnect it carefully so all the pins enter the correct holes in the connector shell.

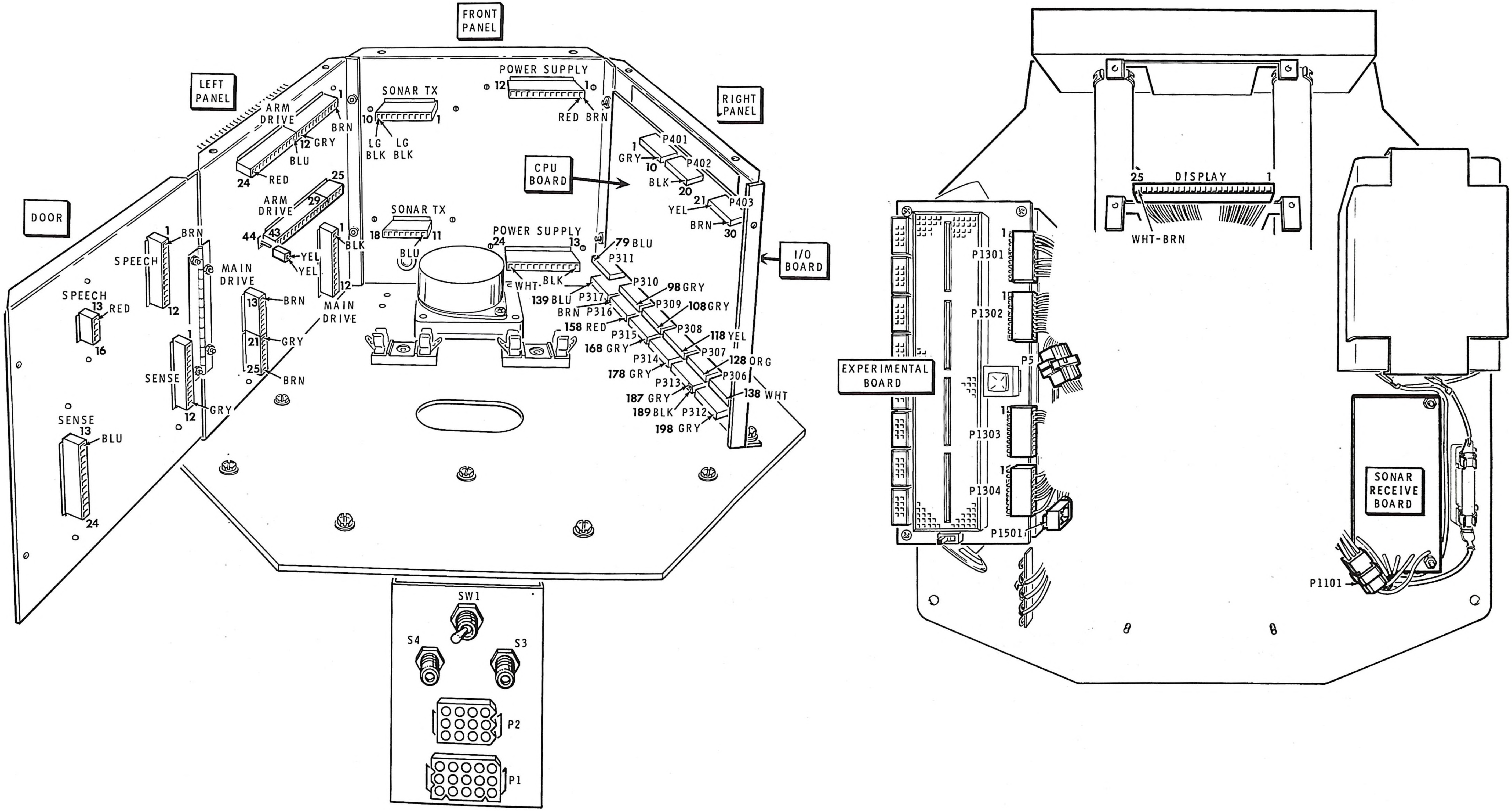
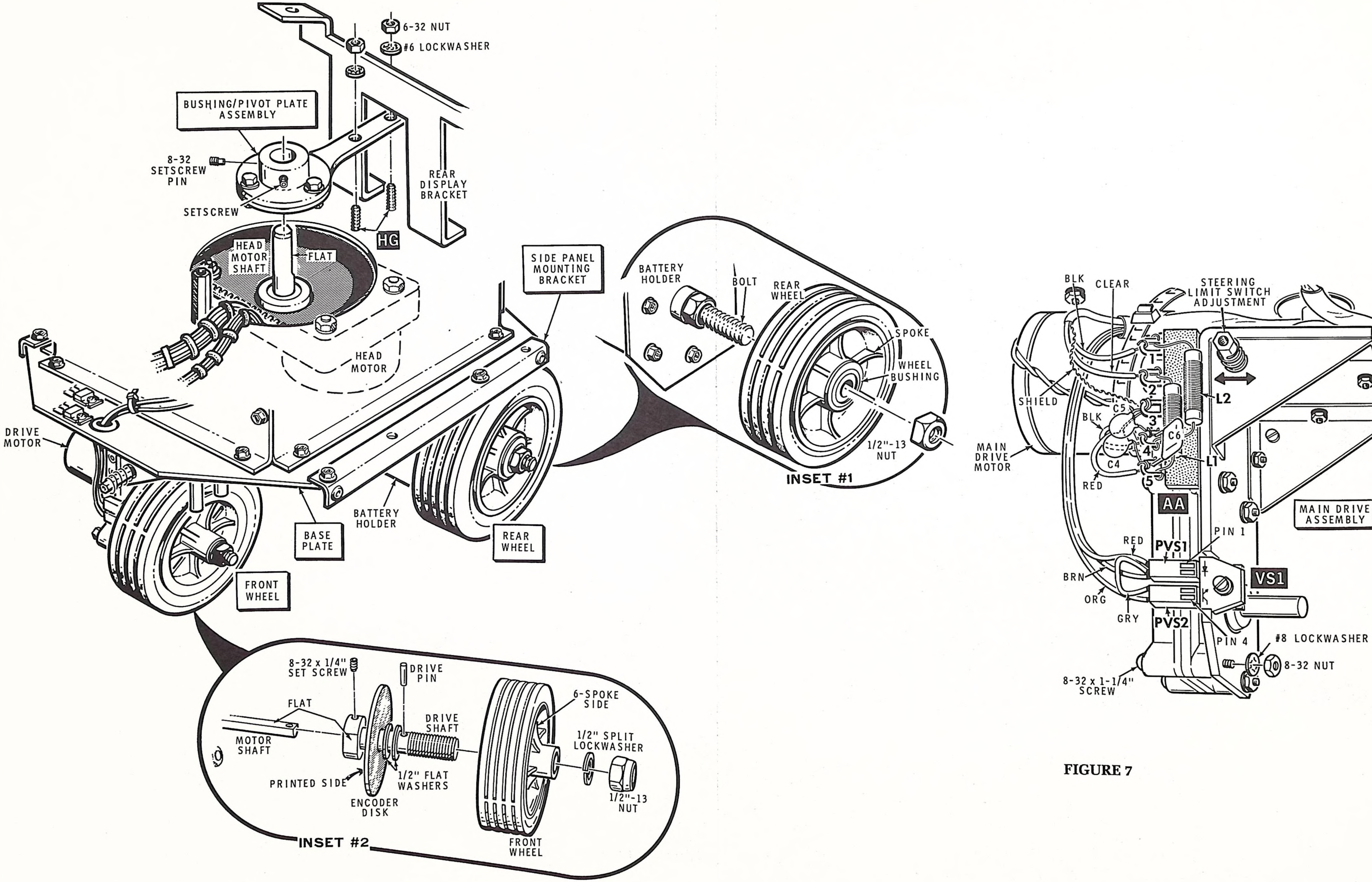


FIGURE 6
PLUGS AND CONNECTORS

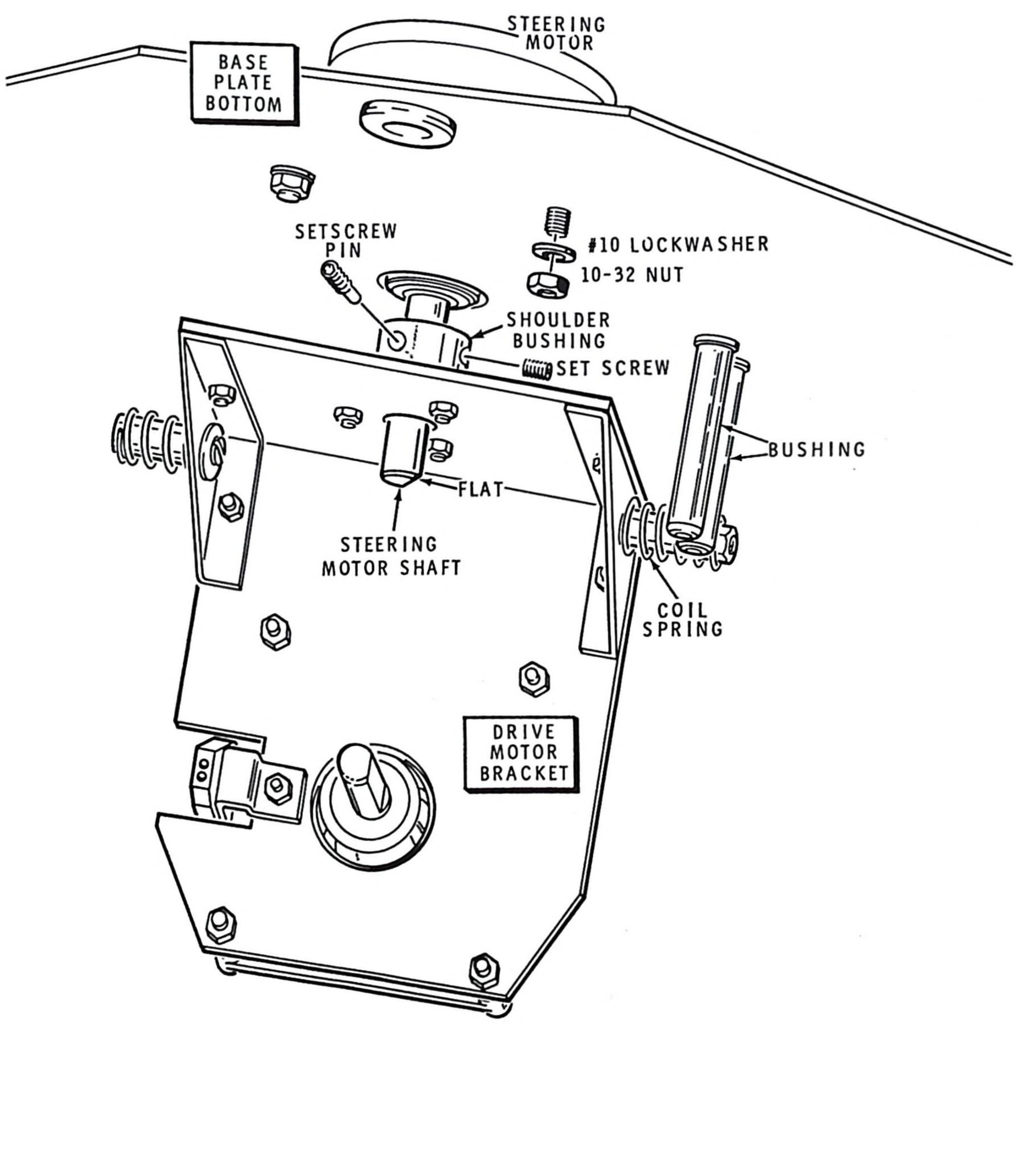


6-32 NUT
#6 LOCKWASHER
BUSHING/PIVOT PLATE ASSEMBLY
8-32 SETSCREW PIN
SETSCREW
REAR DISPLAY BRACKET
HG
HEAD MOTOR SHAFT
FLAT
HEAD MOTOR
SIDE PANEL MOUNTING BRACKET
DRIVE MOTOR
BASE PLATE
BATTERY HOLDER
REAR WHEEL
FRONT WHEEL
BATTERY HOLDER
BOLT
REAR WHEEL
SPOKE
WHEEL BUSHING
1/2"-13 NUT
INSET #1
8-32 x 1/4" SET SCREW
DRIVE PIN
FLAT
MOTOR SHAFT
DRIVE SHAFT
6-SPOKE SIDE
1/2" SPLIT LOCKWASHER
PRINTED SIDE
1/2" FLAT WASHERS
ENCODER DISK
1/2"-13 NUT
FRONT WHEEL
INSET #2
BLK
CLEAR
STEERING LIMIT SWITCH ADJUSTMENT
SHIELD
1
2
3
4
5
C5
C6
C4
L2
L1
BLK
RED
MAIN DRIVE MOTOR
AA
MAIN DRIVE ASSEMBLY
PIN 1
RED
PVS1
VS1
BRN
ORG
GRY
PVS2
PIN 4
#8 LOCKWASHER
8-32 NUT
8-32 x 1-1/4" SCREW

FIGURE 7

MECHANICAL ASSEMBLIES (FIGURES 7 AND 8)

The main drive assembly, steering motor, and head motor are all illustrated in Figure 7 with mounting hardware shown. The arm assembly is shown as an exploded view in Figure 8. Use these illustrations if you ever need to remove one of these assemblies; they will help you to be sure of replacing the correct hardware as you remount the assembly.



ADJUSTMENTS

Your Hero Robot was adjusted during the assembly and initial test for nominal operation. However, the following adjustment procedure will allow you to achieve optimum performance of the sound, light, and motion sensing, the sonar transmit frequency, voice volume and pitch, and the time-keeping of the clock.

You may want to perform only those adjustments for a function that appears to need it, or you may want to go through all of the adjustments to further familiarize yourself with the operation and capabilities of Hero.

ARM AND SHOULDER ROTATION

ARM ROTATION

When the arm extends, it can rotate the entire wrist assembly 90 degrees during the last 2 inches of extension. This rotation is mechanical, and cannot be controlled electronically.

The factory-assembled Robot includes this extend-rotation, to give more mobility. However, you may wish to remove this action, since it makes some carrying and picking up with the gripper more difficult (the gripper changes orientation when the arms extends).

You can change the arm action between "rotating" and "non-rotating" in your Robot by reversing the position of the two arm shells. To do this, refer to Figure 8A and follow these instructions:

1. If the arm is mounted on the Robot, reverse the instructions in the next section "Mounting The Arm," and dismount the arm.

2. Lay the arm on a flat surface, and remove the seven 6-32 × 3/8" black phillips head screws and nuts from the arm shells.

3. The shells have a shoulder plug, shoulder bushing, and motor bracket slots that help keep them aligned and held together. Remove the top shell by carefully working it straight away from the lower section, at these slots.

4. In the same manner remove the bottom shell, and the shoulder plug from the slot in the arm shell it occupys.

5. To reverse the two shells, put the (formerly top) shell in place under the arm assembly. To it, add the shoulder bushing and shoulder plug in their slots, and the motor bracket in its slot. Then carefully place the motor assembly leads in the groove as shown.

6. Note the track inside the remaining arm shell. The 1/4" bushing on the arm must fit into this track. for a "rotating" arm, this track should have a curve as shown in the inset drawing; if you want a "non-rotating" arm, the track should be straight.

7. Position the arm shell loosely in place. Make sure the motor leads are in their groove and will not be pinched. Install one of the 6-32 × 3/8" screws at the wrist, as shown, to hold the wrist cord retainer.

( ) Use the 6-32 × 3/8" hardware to secure the shells together.

The arm rotation is now properly set.

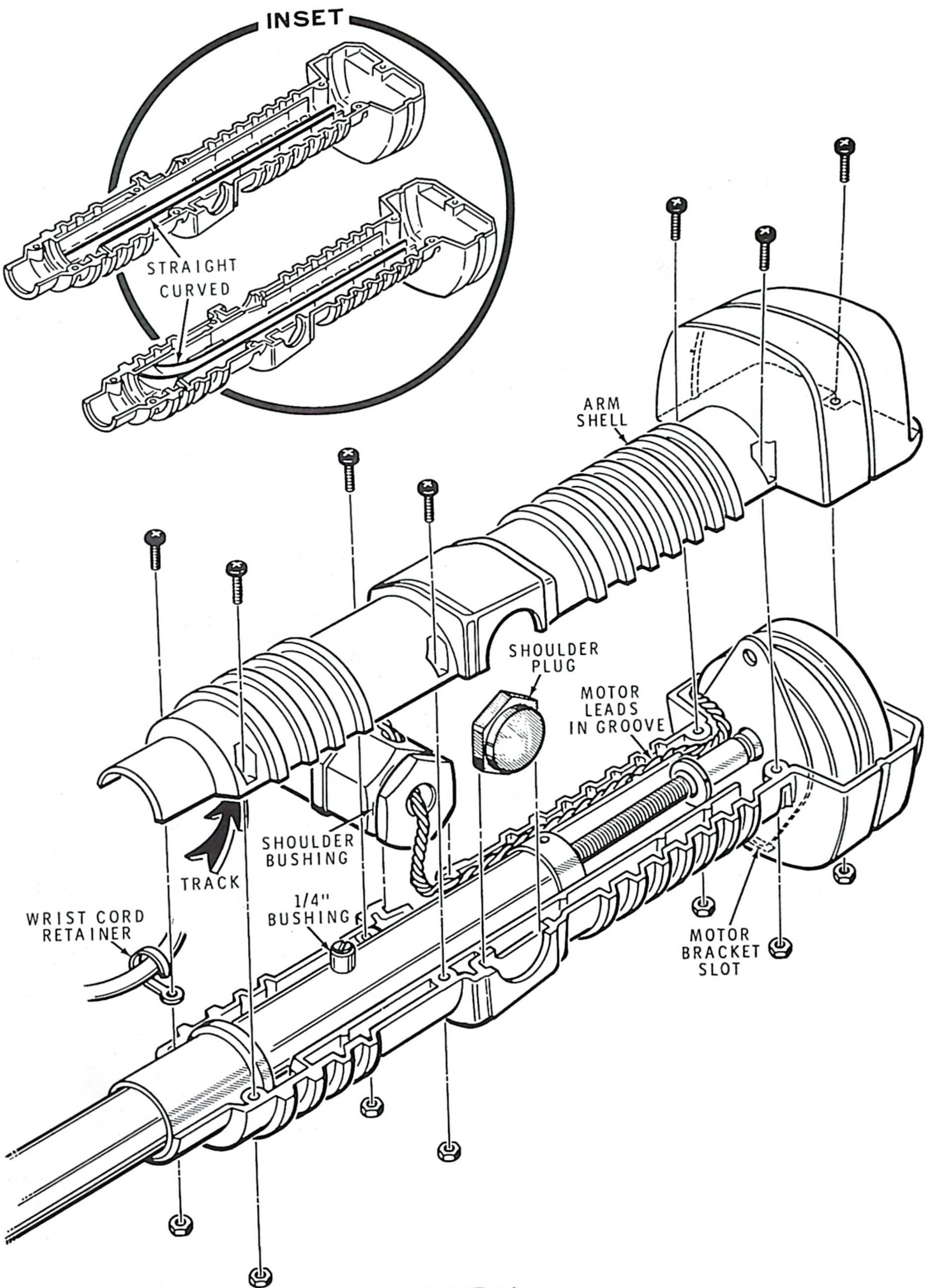

INSET
STRAIGHT
CURVED
ARM
SHELL
SHOULDER
PLUG
MOTOR
LEADS
IN GROOVE
SHOULDER
BUSHING
TRACK
WRIST CORD
RETAINER
1/4"
BUSHING
MOTOR
BRACKET
SLOT

FIGURE 8A

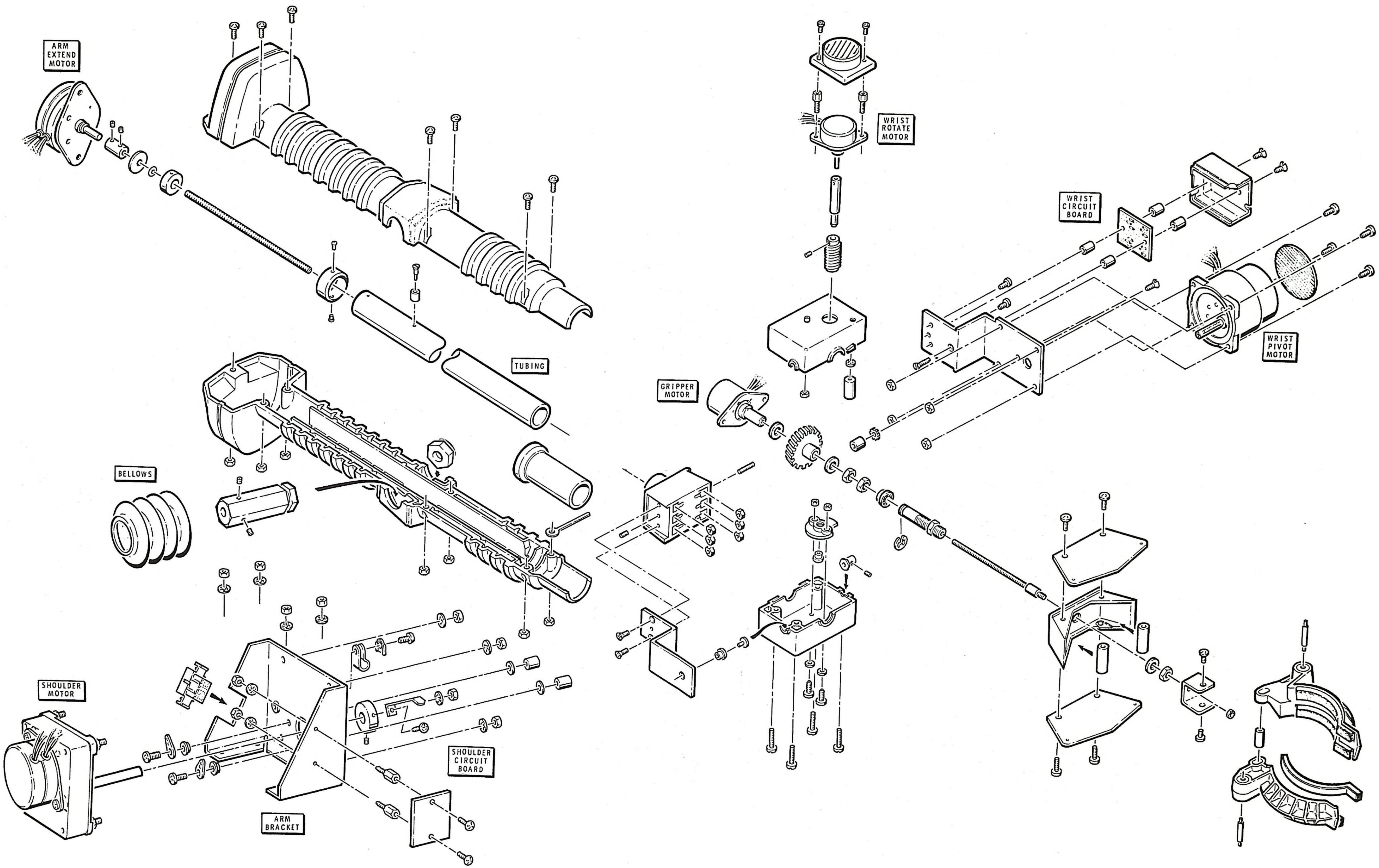

ARM EXTEND MOTOR
WRIST ROTATE MOTOR
WRIST CIRCUIT BOARD
WRIST PIVOT MOTOR
TUBING
GRIPPER MOTOR
BELLOWS
SHOULDER MOTOR
SHOULDER CIRCUIT BOARD
ARM BRACKET

FIGURE 8

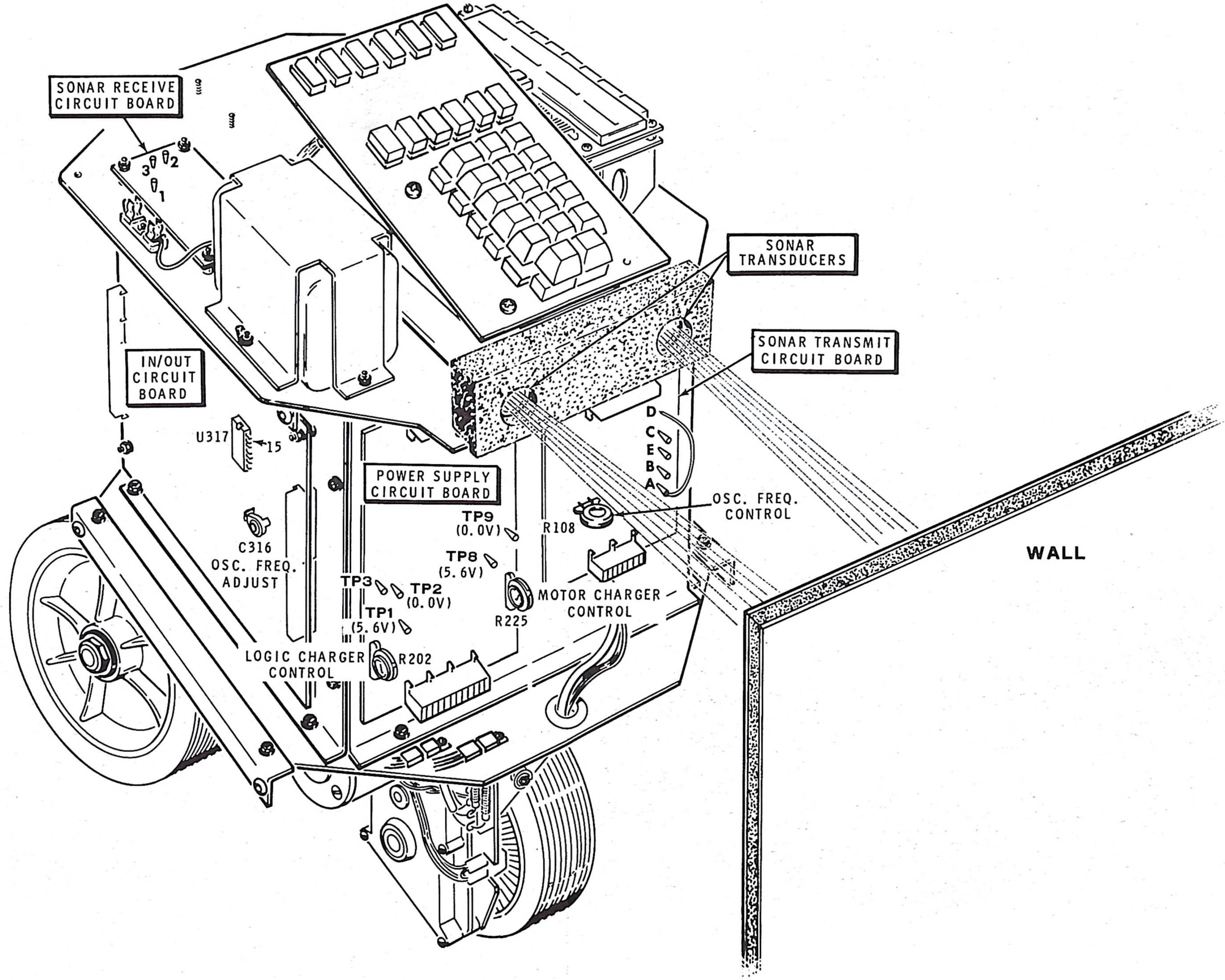

SONAR RECEIVE CIRCUIT BOARD
3
2
1
IN/OUT CIRCUIT BOARD
U317
15
C316 OSC. FREQ. ADJUST
POWER SUPPLY CIRCUIT BOARD
TP9 (0.0V)
TP8 (5.6V)
TP3
TP2 (0.0V)
TP1 (5.6V)
LOGIC CHARGER CONTROL
R202
R108
R225
MOTOR CHARGER CONTROL
SONAR TRANSDUCERS
SONAR TRANSMIT CIRCUIT BOARD
D
C
E
B
A
OSC. FREQ. CONTROL
WALL

FIGURE 9

ARM MOUNTING

The arm mounts on the shaft extending from the bracket at the back of the head. The head panel must be off to mount the arm. Refer to Figure 8B for the following steps.

- Remove the cable clamp and D-washer from the bracket and place the clamp on the arm cable, 5″ from the arm body. See the Inset drawing in Figure 8B.
- Loosen the setscrew in the actuator bushing so the bushing turns freely on the motor shaft.
- Loosen the setscrews in the arm, if necessary, and slide the arm onto the shaft. Then, turn the arm clockwise until the gripper is straight up.
- Reinstall the cable clamp and D-washer; then plug in connector P1502. Note the position of the ridges in the two connectors.
- Slide the arm outward until you can see the flat on the motor shaft. Then turn the arm counterclockwise until either setscrew is over the flat.
- Slide the arm onto the shaft until about 5/16″ of shaft is exposed and tighten the setscrew onto the flat. Then tighten the other setscrew.
- Note the position of the arm and rotate the actuator bushing so its spring contact is at a right angle (90°) to the arm. Then tighten the setcrew in the actuator bushing.

CAUTION: Do not attempt to manually rotate the arm after it is secured to the shoulder motor shaft. Damage to the arm is sure to result.

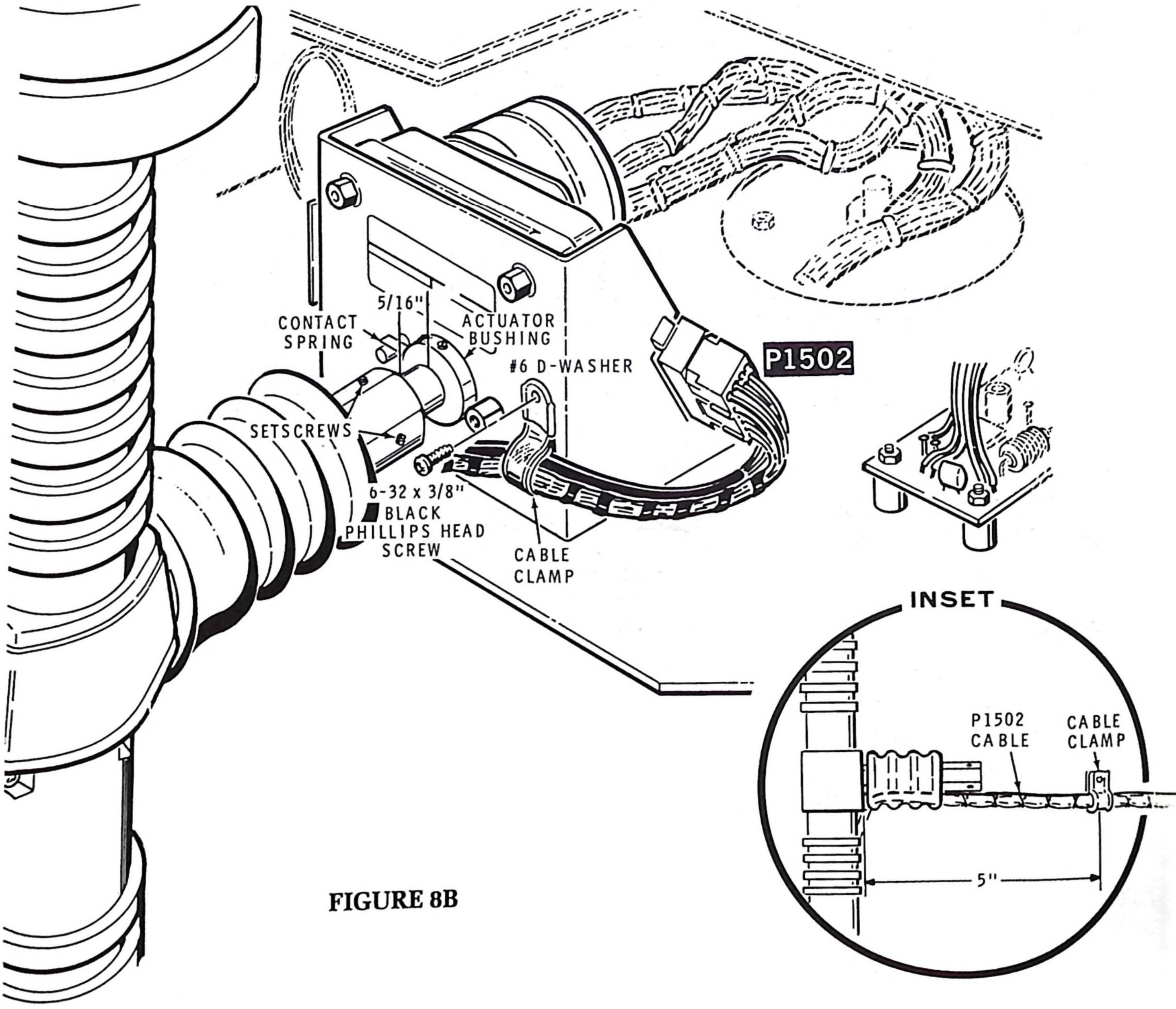


FIGURE 8B

SHOULDER ROTATION

NOTE: Normally, the Robot arm is positioned about straight down at this time. If, after you "initialize" the Robot later in this Manual, the arm is not in a down position, you may wish to adjust it.

To adjust the arm position, loosen the setscrew in the actuator bushing and turn the bushing. If you wish the arm to end up in a more clockwise position, turn the bushing an equal amount counterclockwise, and vice versa. Note that you may have to move the arm so that you can adjust the bushing without bending the bushing spring. After you reposition the bushing, tighten the bushing setscrew and initialize the Robot to check the new arm position.

POWER SUPPLY

Locations of the two adjustments on the power supply circuit board are shown in Figure 9, (fold-out from Page 46) along with the test points used. You will need a high impedance voltmeter to make these adjustments.

LOGIC BATTERY

1. Be sure your Charger is NOT connected. Then remove the two fuses from the base plate inside the torso of Hero.
2. Connect your Charger to Hero and to an AC power receptacle. Then place the power switches of both in the ON position. (Make sure the Sleep switch is OFF first.)
3. Connect the negative lead of your voltmeter to Hero's base plate (chassis), and connect the positive lead to TP1 on the power supply circuit board.
4. Adjust control R202 for a reading of 5.6 volts on the meter.
5. With the positive meter lead still connected to TP1, connect the negative to TP2. Then readjust control R202 for a zero meter reading.

MOTOR BATTERY

1. Connect the negative lead of your voltmeter to the chassis and the positive lead to TP8. Adjust control R225 for a reading of 5.6 volts on the meter.
2. Leave the positive meter lead connected to TP8, and connect the negative lead to TP9. Then readjust R225 for a zero reading on the meter and disconnect the meter test leads.

This completes the adjustment of the power supply. Disconnect your Charger. Then replace the two fuses on Hero's base plate.

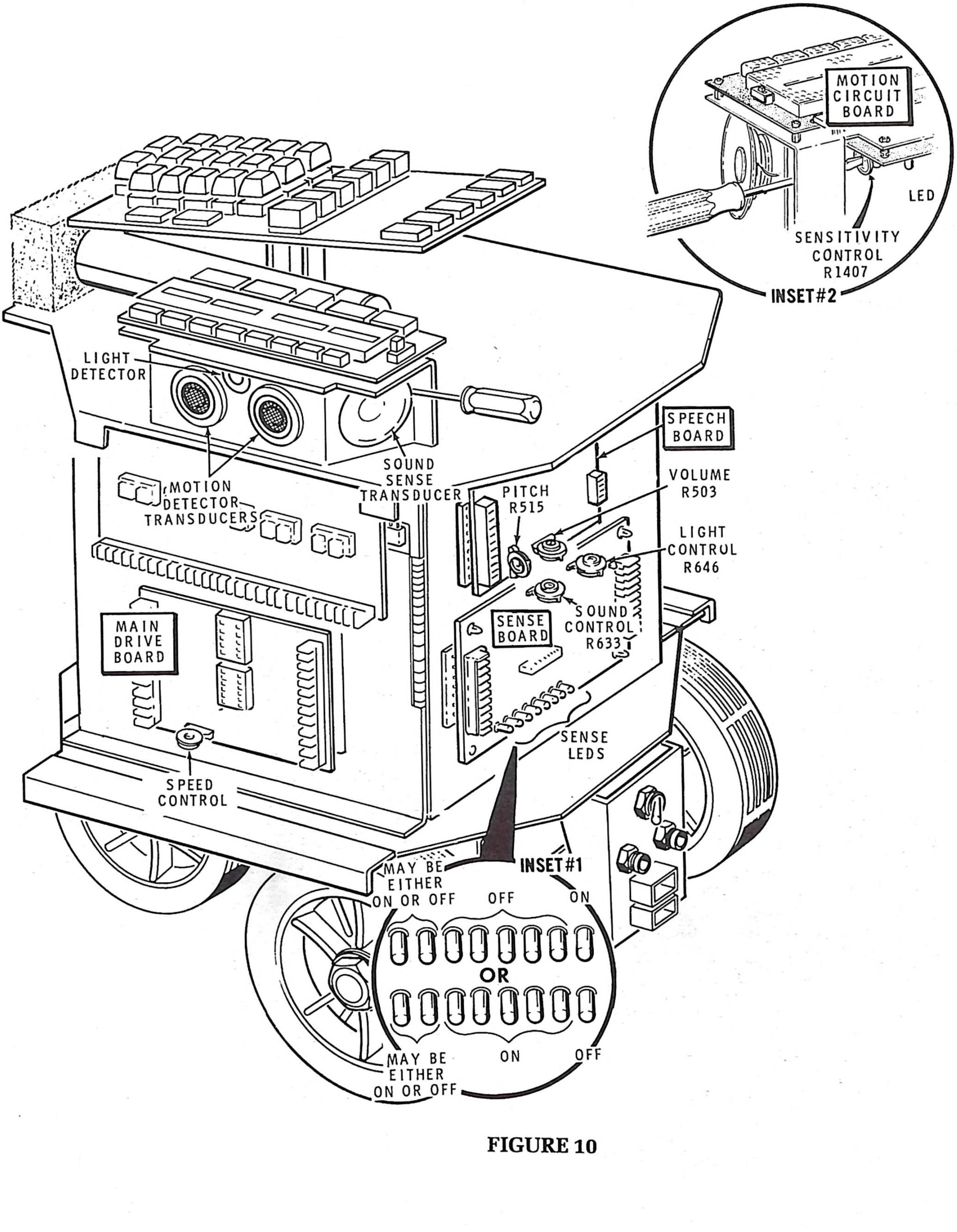

MOTION
CIRCUIT
BOARD
LED
SENSITIVITY
CONTROL
R1407
INSET#2
LIGHT
DETECTOR
MOTION
DETECTOR
TRANSDUCERS
SOUND
SENSE
TRANSDUCER
SPEECH
BOARD
PITCH
R515
VOLUME
R503
LIGHT
CONTROL
R646
SOUND
CONTROL
R633
SENSE
BOARD
MAIN
DRIVE
BOARD
SENSE
LEDS
SPEED
CONTROL
MAY BE
EITHER
ON OR OFF
INSET#1
OFF
ON
OR
MAY BE
EITHER
ON OR OFF
ON
OFF

FIGURE 10

SONAR TRANSMIT FREQUENCY

To make this adjustment, you must connect an oscilloscope or AC voltmeter to TP1 on the sonar **receive** circuit board and make the adjustment on the sonar **transmit** circuit board. Refer to Figure 9 (fold-out from Page 46) for the location of the test points and adjustments.

1. Turn Hero so its sonar tubes point toward a wall or large flat surface four to six feet away.

2. Use the keyboard to turn on the sonar system:

   AA 0205 45 83 96 11 BD F6 4E BD F7 AD CE 20
   00 09 26 FD 20 F0
   RESET AD 0205

3. Connect an oscilloscope or AC voltmeter to TP1 on the sonar receive circuit board. Expect a reading of approximately 0.7 volts)

4. Temporarily remove the jumper wire from point A on the sonar transmit circuit board.

5. Adjust Osc Freq control R108 for a maximum amplitude signal on the oscilloscope or voltmeter. Then replace the jumper wire in point A.

Check the operation of the sonar by moving your hand back and forth in front of the sonar tubes. The reading on the Robot's display should decrease as you move your hand toward the tubes and increase as you move it away.

NOTE: If the indication remains constant as you move your hand in front of the transducer tubes, move the jumper wire on the sonar transmit circuit board from A to B and repeat the previous step. If you still do not get the proper response, move the jumper wire from B to E, then from E to C. The position of the jumper wire determines the width of the transmitted pulse to compensate for variations in the sensitivity of the sonar transducers. The jumper should be connected to a point furthest from point D where the display will respond to objects at various distances.

When you have completed the sonar adjustment, disconnect the oscilloscope or voltmeter.

SPEED CONTROL

The SPEED CONTROL is located on the main drive circuit board shown in Figure 10, fold-out from this Page.

1. Set the SPEED CONTROL to midrange.

2. Plug in the pendant.

3. Turn the POWER switch ON.

4. Turn the SELECTOR switch on the remote control to N.

5. Push key 4.

6. Turn the SELECTOR switch to each speed setting and observe the speed of that function. Then adjust the SPEED CONTROL to the desired speed.

NOTE: The SPEED CONTROL adjustment changes the speed of all three settings. If you turn the control too far clockwise or counterclockwise, two of the speeds may be the same.

NOTE: If you write programs that use both light and sound sensors (described on Page 49), do not use disable commands when you switch from one sensor to the other unless you insert a delay (pause command) of at least two seconds.

SENSE

Figure 10 (fold-out from Page 48) shows the locations of the sound and light controls and the test points used in the following adjustments.

Connect your Charger to Hero and to an AC power receptacle. Then place the power switches of both in the ON position. Check to be sure the jumper wire is in place between TP1 and TP2 on the sense circuit board.

SOUND

1. Use the keyboard to key in the sound function:

   AA 0220 42 20 FE RESET AD 0220

2. Rotate Sound control R633 fully clockwise. Most of the LEDs at the bottom of the sense board should glow.

NOTE: The LEDs form a binary counter, not a linear indicator. The left-most LED represents the least significant bit while the right-hand LED represents the most significant bit. Midrange in the following adjustments will be achieved when the LEDs glow in either of the patterns shown in Inset drawing #1 of Figure 10.

3. Rotate the control counterclockwise until the LEDs flicker in response to sounds (such as normal conversation). Set the control to the position where the softest sounds expected will just light one or two of the left-hand LEDs. Loudest sounds should light two or more of the right-hand LEDs.

LIGHT

1. Use the keyboard to key in the light function:

   AA 0225 41 20 FE RESET AD 0225

2. Rotate Light control R646 fully counterclockwise. Most of the LEDs at the bottom of the sense board should glow.

3. Rotate the control clockwise until two or three left-hand LEDs flicker in response to faint light and right-hand LEDs glow steadily in the brightest light. This control acts like a sensitivity control and should be set to a point that allows the LEDs to respond over the full dynamic range of light levels in which Hero will operate.

MOTION DETECTOR

Inset drawing #2 on Figure 10 shows the location of the Sensitivity control on the Motion circuit board. Remove the head panel from Hero to make this adjustment.

1. Enter the following on the keyboard to turn on the motion detector:

   AA 0230 4B 20 FE RESET AD 0230

2. Pass the blade of a screwdriver through the opening under the Experimental board and into the slot of the Sensitivity control (R1407) on the motion board.

3. Move your hand in front of the motion detector transducers and observe the LED on the motion circuit board. The LED should flicker with moderately slow movement.

4. Adjust the Sensitivity control counterclockwise to increase, or clockwise to decrease the sensitivity of the motion detector.

When you are satisfied with the sensitivity adjustment, remove the screwdriver and replace Hero's head panel.

SPEECH

If you have a Speech option installed, you can adjust the pitch and volume from baritone to soprano, and from a whisper to a shout. Figure 10 shows the location of the adjustment controls.

1. With your Hero's power turned on, press RESET and you should hear "ready" from the speaker.
2. Press RESET again and again as you adjust Volume control R503 for your desired listening level and Pitch control R515 for the pitch you prefer. The proper setting for these controls is determined by your personal preference, but others may understand the Robot more easily if the speech is a slow rich baritone.

TIME

If you have a frequency counter, you can accurately set the timer with a single adjustment. Otherwise, you can set the time with a known standard and compare every day, adjusting for fast or slow as needed.

FREQUENCY COUNTER METHOD

1. Refer to Figure 9 (fold-out from Page 44) and connect your frequency counter to pin 15 of integrated circuit U317 on the I/O circuit board.
2. Adjust trimmer capacitor C316 for a reading of 32.768 kHz on the counter.
3. Disconnect the counter from the I/O circuit board.

TIME STANDARD METHOD

1. Preset Freq Adj capacitor C316 with its slot horizontal as shown in Figure 9.
2. Enter 35 on the keyboard. Then enter the exact time (from the phone company, for example) with two digits for the hour, two for the minute, and two zeros for the second. Note that the seconds will reset to 00 regardless of the digits you enter.
3. With A P 24 displayed, enter D for AM (morning), E for PM (afternoon), or F for a 24 hour clock. After this entry, the Robot will return to the executive mode and you can perform other functions or complete the adjustments.

To read the time, enter 37. The clock will continue to display until you press RESET, and will keep on counting time even with the Robot power switch turned off.

4. After at least 12 hours, read the time (enter 37) and compare it with a known standard. If the clock has gained, adjust capacitor C316 slightly counterclockwise. If the clock is slow, adjust the capacitor slightly clockwise.
5. Check the clock against a known time standard every 12 or more hours. Readjust C316 as necessary until the clock keeps reasonably accurate time.

STEERING

If Hero does not travel in a straight line on a flat surface, you can make the "straight ahead" more accurate by adjusting the limit switch spring on the main drive assembly. First determine whether correction needs to be to the right or to the left. Then refer to Figure 7 (fold-out from Page 42) and loosen the limit switch spring hex post. Move the spring post in the direction of the desired correction; then tighten the post.

Reinitialize by pressing 31 and let Hero cycle through the "homing" sequence. After the steering motor has "homed," check to see that it is straight ahead. If not, readjust the spring post slightly.

DRIVE WHEEL SENSOR

Figure 5 (fold-out from Page 40) shows the optical pickup and the encoder disk on the drive wheel. This system senses the drive wheel rotation best when there is 0.15" between the pickup and the disk. If it is necessary to adjust the pickup, loosen the pickup screw slightly, reposition the pickup, and retighten the screw.

This completes the adjustments of Hero. Replace whatever hardware or panels you may have removed.

TROUBLESHOOTING

Should a problem develop in your Hero, the first step in troubleshooting is to determine what function doesn't operate, or what it does wrong. Then, by making a few tests in the circuits that affect the function, you can localize the problem to a specific circuit board and perhaps to a particular circuit on the board.

This section of the Manual is divided into three sections. In the "Operation Check," you will power up Hero and test its functions for normal operation. Suggested areas to test are listed for each function in case you do not obtain the indicated results. In the "Initial Tests" you will do certain things and observe the response. If you do not get the indicated response, you will be directed to one or more sections of the "Troubleshooting Chart."

CAUTION: If you are not experienced in the repair of digital electronic circuits, we recommend that you limit yourself to tracing a problem to a given circuit board and have the Heath Company repair the circuit board for you. See "Service Information" on the inside rear cover of this Manual.

CAUTION: Some vacuum tube voltmeters and other instruments that use two-conductor line cords, or have three-conductor cords that have been adapted for two-prong plugs, can have dangerous voltage on their common (ground) leads. Do not attempt to service the Robot with such instruments. Refer to the manual for your instrument for safety information.

NOTE: Before you suspect a circuit board or component, recheck the internal interconnecting cables and plugs. Be sure each plug is seated on its proper connector pins. Refer to Figure 6 for the location of the connectors and plugs. Also check to be sure each circuit board is properly seated on its connector pins and that no pin is bent under the circuit board.

OPERATION CHECK

In this section, you will check the normal operation of Hero to determine which functions do not operate properly. This will help you to quickly close in on the cause of a problem, rather than making unnecessary tests.

Read each instruction completely through so you will know what results to expect, and then perform the check. If you do not obtain the indicated results, go at once to the suggested section in the "Initial Tests." Try to resolve a problem as you encounter it, and then continue with these "Operation Checks."

I. POWER UP

Connect your charger to the Robot and to an AC receptacle. Then place the power switches of the charger and the Robot in the ON position. (Make sure the Sleep switch is OFF, first.)

The readout on the display board should say "HEro1.0" for about ten seconds, and then change to a moving dash.

If it does not, check the charger (see "Power Supply Test," Page 55). Then go to "Logic +12V and +5V Supplies," "Motor +12V Supply," and "Display" in the "Troubleshooting Charts."

The speech option should say "ready."*

If it does not, go to "Speech" and "Input/Output" in the "Troubleshooting Charts," beginning on Page 62.

II. INITIALIZE

Enter 31 on the keyboard to "initialize" the Robot. The following sequence of events should occur:

- Arm extend goes fully in.*
- Shoulder pivot goes fully down*
  (If it does not, go to "Arm Drive" and "Input/Output" in the "Troubleshooting Charts.")
- Wrist rotates fully counterclockwise.*
- Wrist pivots fully up.*
- Gripper closes.*
- Head turns to clockwise limit.
- Drive wheel turns left to limit.
- Head returns to center.
- Drive wheel returns to center.
- Computer returns to Executive Mode, displays "HEro1.X", and says "ready."*
  (If it does not, go to "Main Drive" and "Input/Output" in the "Troubleshooting Charts.")

The preceding Operation Checks involved the "heart" of Hero and circuits that affect other operations. If you have a problem with the motion, sense, sonar, or other operations, refer to that particular section in the following "Initial Tests."

* The events marked with an asterisk will occur only if your Robot has the necessary Arm or Speech option. However, the Robot will wait during the time allotted for arm movement, even if it does not have the arm.

INITIAL TESTS

POWER SUPPLY TEST

1. Connect your charger to the Robot and to an AC receptacle. Then place both the charger and the Robot power switches in the ON position. The Robot should say "ready,"* and the readout on the display board should read "HEro1.X" for several seconds and then change to a moving dash.

2. Press RESET and "HEro1.X" should again appear, and the Robot should repeat "ready."*

If you got the proper response in those two tests, or if you heard "ready"* when you pressed RESET but no display was visible, skip the next two tests and proceed to "Display Tests."

3. Check to see that the charger is ON (red indicator lamp glowing). If not, check the charger fuse.

4. If the charger is on and the white charging lamp does not glow, check fuses F1 and F2 on the base plate inside Hero. Also check the charger plug connection to Hero.

If, after making these checks, you still do not obtain the proper response, there is very likely a problem in the power supply. Proceed to "Logic +12V and +5V Supplies" and "Motor +12V Supply" in the Troubleshooting Charts.

DISPLAY TESTS

The following tests will check the operation of the readout on the display board and the microprocessor on the CPU board. Make these tests with the charger and Hero connected and turned on.

1. Press RESET. The display should read "HEro1.X" for about 10 seconds and then change to a single dash that moves across the display.

2. Enter the following program, which will test key recognition and verify the display response. Press each key in the order shown below and observe the display. If any of the LEDs do not display the indicated characters, or if segments of one or more LEDs do not light, refer to "Display" in the Troubleshooting Chart. The "*****" represent five blank spaces on the display.

KEYBOARD ENTRY	DISPLAY
RESET	HEro1.X
A	A.*****
E	____Ad.
0123	0123XX
C	0123__
45	012345
E	____Ad.
6789	6789XX
C	6789__
AB	6789Ab
E	____Ad.
CDEF	C.dEFXX

Clock Check

Since the clock on the Input/Output circuit board generates timing pulses that control other functions, check its operation in the following manner. If you find that you are unable to set or display the time, refer to "Input/Output" in the Troubleshooting Charts.

1. Enter 35 and the display should read "HH (two blanks) SS". Then enter two digits for the hour, two for the minute, and two for the second. The display should now read "A (blank) P (blank) 24".

2. Press D for AM (morning), E for PM (afternoon), or F for a 24-hour clock. The Robot should return to the Executive Mode.

To read the time, enter 37. The time should continue to display until you press RESET.

* The event marked with an asterisk will occur only if your Robot has the Speech option.

MOTION DETECTOR TESTS

Enter the following program to enable (activate) the motion detection circuitry. Observe the display as you press each key. As you enter the last digit, the display should turn off and the motion detection circuit should be activated. Verify this by observing the LED on the motion circuit board as you perform the tests below (1 and 2). See inset drawing #1 on Figure 10. Refer to "Motion" in the "Troubleshooting Charts" if you do not get the proper response at any point.

KEYBOARD ENTRY	DISPLAY
RESET	HEro1.X
A	A.*****
A	----Ad.
0230	0230--
4B	0231--
20	0232--
FE	0233--
RESET	HEro1.X
A	A.*****
D	----do.
0230	

(The * indicates a blank on the display.)

1. Move your hand toward and away from the motion detector transducers. The LED should flicker, or light and stay lit for a short time whenever motion is detected.

2. If the LED did not flicker or light, adjust the Sensitivity control on the motion circuit board — counterclockwise to increase sensitivity — and try again.

SENSE BOARD TESTS

The light detection and sound detection circuits, on the sense circuit board, will be tested separately.

Light Detection Tests

Enter the following program to enable the light detection circuits. Observe the display as you press each key in order.

KEYBOARD ENTRY	DISPLAY
RESET	HEro1.X
A	A.*****
A	----Ad.
0225	0225--
41	0226--
20	0227--
FE	0228--
RESET	HEro1.X
A	A.*****
D	----do.
0225	(blank)

Observe the eight LEDs on the sense board as you make the following tests. Their on and off conditions should change with a change in the amount of light striking the light sensor. If you do not get the proper response at any point, refer to "Sense" in the "Troubleshooting Charts."

1. Rotate the Robot's head so that no direct bright light falls on the light sensor.

2. Increase the amount of light on the sensor by shining a flashlight on it or by reflection from a mirror or a sheet of white paper. The pattern of the LEDs should change in response to the change in light.

NOTE: The LEDs form a binary counter, not a linear indicator. Although maximum response would be indicated when all LEDs are lit, and minimum when only one or two at the end are lit, midrange would be indicated by a pattern like either shown in inset drawing #2 on Figure 10.

Sound Detection Tests

Enter the following program to enable the sound detector circuitry. Observe the display as you press each key.

KEYBOARD ENTRY	DISPLAY
RESET	HEro1.X
A	A.*****
A	----Ad.
0230	0230--
42	0231--
20	0232--
FE	0233--
RESET	HEro1.X
A	A.*****
D	----do.
0230	(blank)

Verify the operation of the sound detection circuit by observing the LEDs on the sense circuit board in the following manner. If you do not get the proper response in a step, refer to "Sense" in the "Troubleshooting Charts."

1. With the room as quiet as possible, see that only one or two LEDs at the end are lit.

2. Speak in a loud voice and see that most of the LEDs light. See the note under "Light Detection Tests" above.

SONAR TEST

Enter the following program to enable the sonar circuitry. Observe the display as you press each key in order. As you enter the last digit, two figures should show at the left-most side of the display.

KEYBOARD ENTRY	DISPLAY
RESET	HEro1.X
A	A.*****
A	----Ad.
0205	0205--
45	0206--
83	0207--
96	0208--
11	0209--
BD	020A--
F6	020b--
4E	020C--
BD	020d--
F7	020E--
AD	020F--
CE	0210--
20	0211--
00	0212--
09	0213--
26	0214--
FD	0215--
20	0216--
F0	0217--
RESET	HEro1.X
A	A.*****
D	----do.
0205	

If you do not get the proper results in a step, refer to "Sonar" in the Troubleshooting Charts.

1. Rotate the Robot's head so the sonar tubes point squarely at a wall about 5 feet away. The display should show a value* and remain fairly constant. (Remember, however, that ultrasonic noise can be present, even in the home. The refrigerator or air conditioner motor, jingling keys, and other noises could affect the ultrasonic range sensor.)

2. Place your hand approximately 2 feet in front of the sonar tubes. The display should decrease considerably. The more steady you hold your hand, the more constant the display.

* NOTE: The value is displayed in hexadecimal (base 16) rather than the normal (base 10) number system. As a value increases above 9, letters (A through F) will display before 10. Thus, a value of 15 displays as F.

SPEECH TESTS

If you have the Speech option, you can test its operation in the following manner:

Press RESET. The display should read "HEro1.X" and the Robot should say "ready."

2. Enter the following program on the keyboard. Observe the display as you press each key. As you enter the last digit, the display should be blank and the Robot should say "Hello, my name is Hero." If you do not get the correct response at any point, refer to "Speech" in the "Troubleshooting Charts."

KEYBOARD ENTRY	DISPLAY
A	A.*****
A	____Ad.
0200	0200__
72	0201__
FA	0202__
4B	0203__
20FE	0204__
RESET	HEro1.X
A	A.*****
D	____do.
0200	(blank)

3. Press RESET and listen for the word "ready" as the display reads "HEro1.X". Then enter A D 0200. The phrase should repeat.

TEACHING PENDANT TEST

Use this test if your Robot does not respond as it should when you are using the teaching pendant.

1. If the Robot has not been initialized since it was turned on, enter 31 to initialize it.

2. Connect the Teaching Pendant to the Robot's back panel.

3. Enter the following program on the keyboard. Observe the display as you press each key.

KEYBOARD ENTRY	DISPLAY
RESET	HEro1.X
1	1.*****
A	____Ad.
0240	0240__
BD	0241__
F6	0242__
4E	0243__
B6	0244__
C2	0245__
80	0246__
BD	0247__
F7	0248__
AD	0249__
CE	024A__
10	024B__
00	024C__
09	024D__
26	024E__
FD	024F__
20	0250__
EF	0251__
RESET	HEro1.X
1	1.*****
D	____do.
0240	

4. The display will show the pendant's position as received at the Robot's input port ($#C280). Set the pendant switches as follows:
   Rotary switch to N.
   Function switch to ARM.
   Motion switch to center (released).
   Trigger switch to out (released).

5. Position the switches as shown below. The display should show the corresponding code. Do not pull the Trigger or move any other switch unless you are told to do so. If you get the proper response at each switch setting, the teaching pendant is operating properly. If you do not get the proper response at any point, proceed to step 6.

SWITCH SETTING	DISPLAY
As set above	8E
Rotary to WRIST PIVOT	9E
Rotary to WRIST ROTATE	AE
Rotary to GRIP	BE
Rotary to ARM PIVOT	DE
Rotary to ARM EXTEND	EE
Rotary to HEAD	FE
Rotary back to N	8E
Function to BODY	0E
Motion to RIGHT	06
Motion to LEFT	0A
Motion CLEAR (released)	0E
Trigger IN (pulled)	0F

6. Since the pendant has some active components in it, it is possible for one specific pendant position to cause an improper output to the Robot port. If you have trouble with one or two specific pendant settings, refer to the following chart. Place the pendant switches in the questionable setting and verify that the output to the Robot corresponds to that shown in the chart. This will tell you if the teaching pendant is operating correctly.

FUNCT. SW.	MOTION SW.	TRIGR. SW.	ROTARY SWITCH: HEAD	EXTEND	SHOULDER	N	PIVOT	ROTATE	GRIP
			TEACHING PENDANT OUTPUT (HEX)						
ARM	CLEAR	OUT	FE	EE	DE	8E	9E	AE	BE
ARM	CLEAR	IN	FF	EF	DF	8F	9F	AF	BF
ARM	LEFT	OUT	FA	EA	DA	8A	9A	AA	BA
ARM	LEFT	IN	FB	EB	DB	8B	9B	AB	BB
ARM	RIGHT	OUT	F6	E6	D6	86	96	A6	B6
ARM	RIGHT	IN	F7	E7	D7	87	97	A7	B7
BODY	CLEAR	OUT	7E	6E	5E	05	1E	2E	3E
BODY	CLEAR	IN	7F	6F	5F	0F	1F	2F	3F
BODY	LEFT	OUT	7A	6A	5A	0A	1A	2A	3A
BODY	LEFT	IN	7B	6B	5B	0B	1B	2B	3B
BODY	RIGHT	OUT	76	66	56	06	16	26	36
BODY	RIGHT	IN	77	67	57	07	17	27	37

DRIVE MOTOR AND STEERING MOTOR

As you test the drive motor, you will need plenty of room for Hero to move about. You will use the teaching pendant to control the movement of the Robot in these tests, so you should use the previous test to verify the proper operation of the teaching pendant before you begin this test.

If you do not get the indicated results in any of these tests, refer to "Main Drive" in the "Troubleshooting Charts."

1. If you are not sure the Robot has been initialized, press 31 and allow the Robot to initialize.

2. Connect the teaching pendant. Then press 4 on the keyboard. The display should show a 4, followed by a blank and random characters.

3. Set the pendant's Function switch to the BODY position, and the Rotary switch to the N (center) position.

4. Press the trigger and either side of the Motion switch. The drive wheel should turn left or right as directed by the Motion switch.

5. Release the trigger and Motion switches and the drive wheel should stay where you pointed it. Press only the trigger and the drive wheel should return to straight ahead.

6. Change the Rotary switch to (WRIST) PIVOT (which is also the "forward slow" position). Press the trigger and the Robot should begin to move forward.

7. Steer the Robot by pressing either the right or left side of the Motion switch as you hold the trigger. Remember that the drive wheel will begin to straighten out when you release the Motion switch and stop when you release the Trigger.

ARM DRIVE (BOARD AND MOTORS) TESTS

If you have the Arm option installed, you can test its operation in the following manner. If you do not get the proper results in any of these tests, refer to "Arm Drive" in the "Troubleshooting Charts."

1. Be sure the Robot has been initialized.

2. If you have not done so in previous steps, connect the teaching pendant to the Robot and press 4 to put the Robot in the Manual Mode.

3. Turn the teaching pendant Rotary switch to N (neutral) and the Functionn switch to ARM.

NOTE: Each position of the stepper motors is stored as a number in Hero's memory system. As each motor rotates, its number is automatically updated by the CPU. In the Manual Mode, this absolute number shows as the two right-hand digits in the display.

4. Turn the teaching pendant Rotary switch through each of the positions shown in the following chart and compare the two right-hand digits of the display at each position. These are the normal values for the home position of each function and should reset to these values after each initialization.

SWITCH POSITION	DISPLAY
Head	62
Extend	00
Arm Pivot	00
N (Neutral)	Blank
Wrist Pivot	00
Wrist Rotate	4d
Gripper	00

5. Select the ARM PIVOT position of the Rotary switch.

6. Press and hold the trigger as you press the right side of the Motion switch. The arm should raise and the two right-hand digits should increase. Once the arm has raised about six inches, release the Motion switch while still pressing the trigger.

7. Press the left side of the Motion switch to lower the arm and notice that the right-hand numbers in the display decrease. Leave the arm in some position other than its initialized position.

8. In the same manner as in steps 5, 6, and 7, select each of the arm and wrist movement functions and verify that they operate with their relative positions displayed. Leave each motor in some position near the middle of its operation (not in the initialized position).

9. Once you have verified the operation of each function of the arm, turn the Rotary switch to N (neutral) and press RESET to return to the Executive Mode. The display should read "HEro1.X" (and the Speech option should say "ready").

10. Enter 32 on the keyboard. Each arm function should return from its place in midposition to its home position and the display should read "HEro1.X".

INPUT/OUTPUT (I/O) AND CPU BOARDS

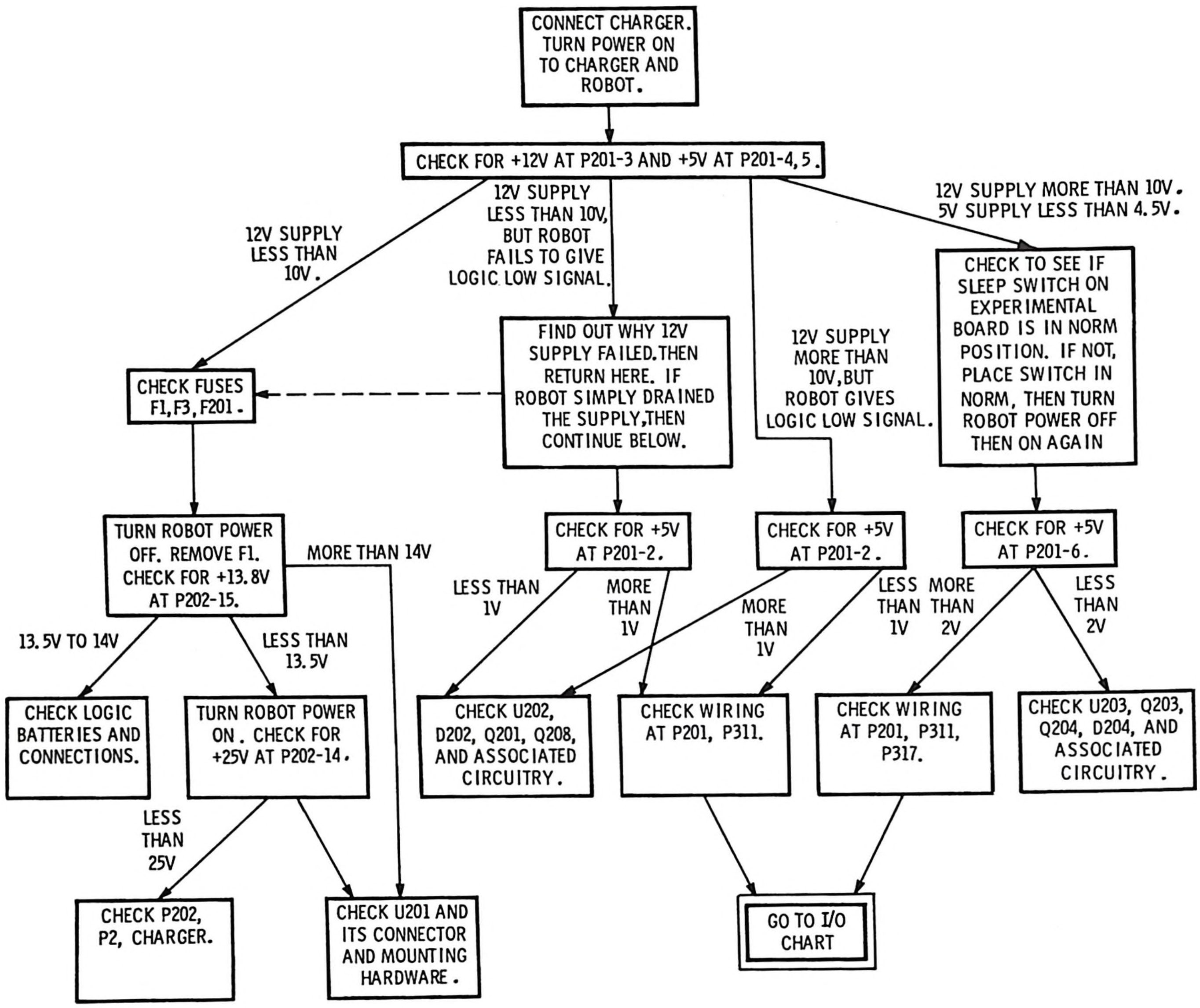
These two circuit boards constitute the on-board computer for your Robot. Their interactions and interconnections are quite complex, and servicing them could require considerable patience and skill. Unless you can readily identify the problem using the following tips or the "Troubleshooting Charts," you may find that it is best to return either or both of these boards to the Heath Company for service.

If you have installed an optional ROM at U417, make sure that the jumpers are correctly placed to mate this ROM to the rest of the system. Make sure all of the ROM's pins are making proper contact and none of the board connectors came off during the installation.

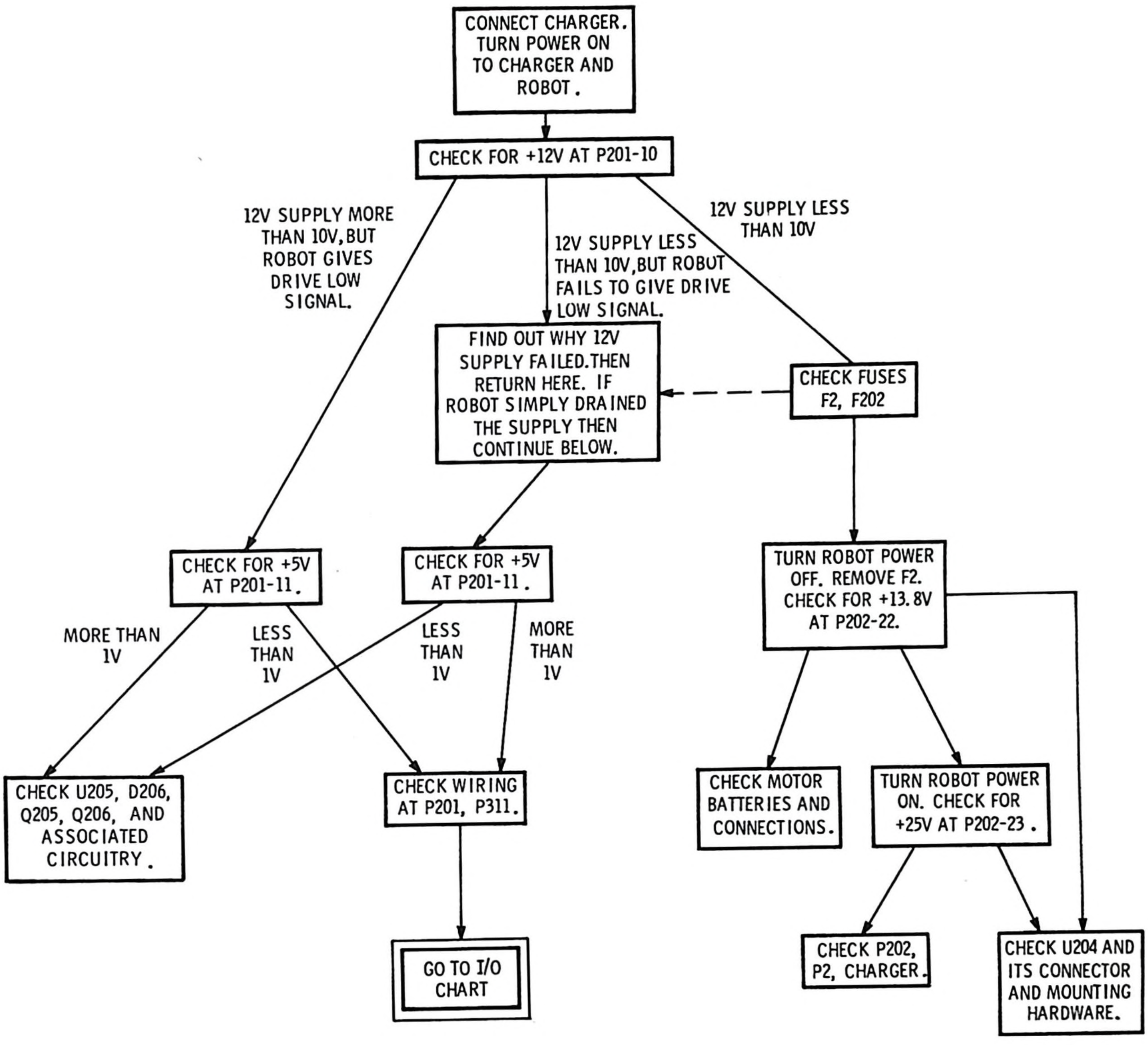
Sometimes you can locate semiconductor-caused problems by swapping IC's with identical Heath part numbers (remember that not all 74LS374's are considered identical by Heath — do not switch one type with the other).

TROUBLESHOOTING CHARTS

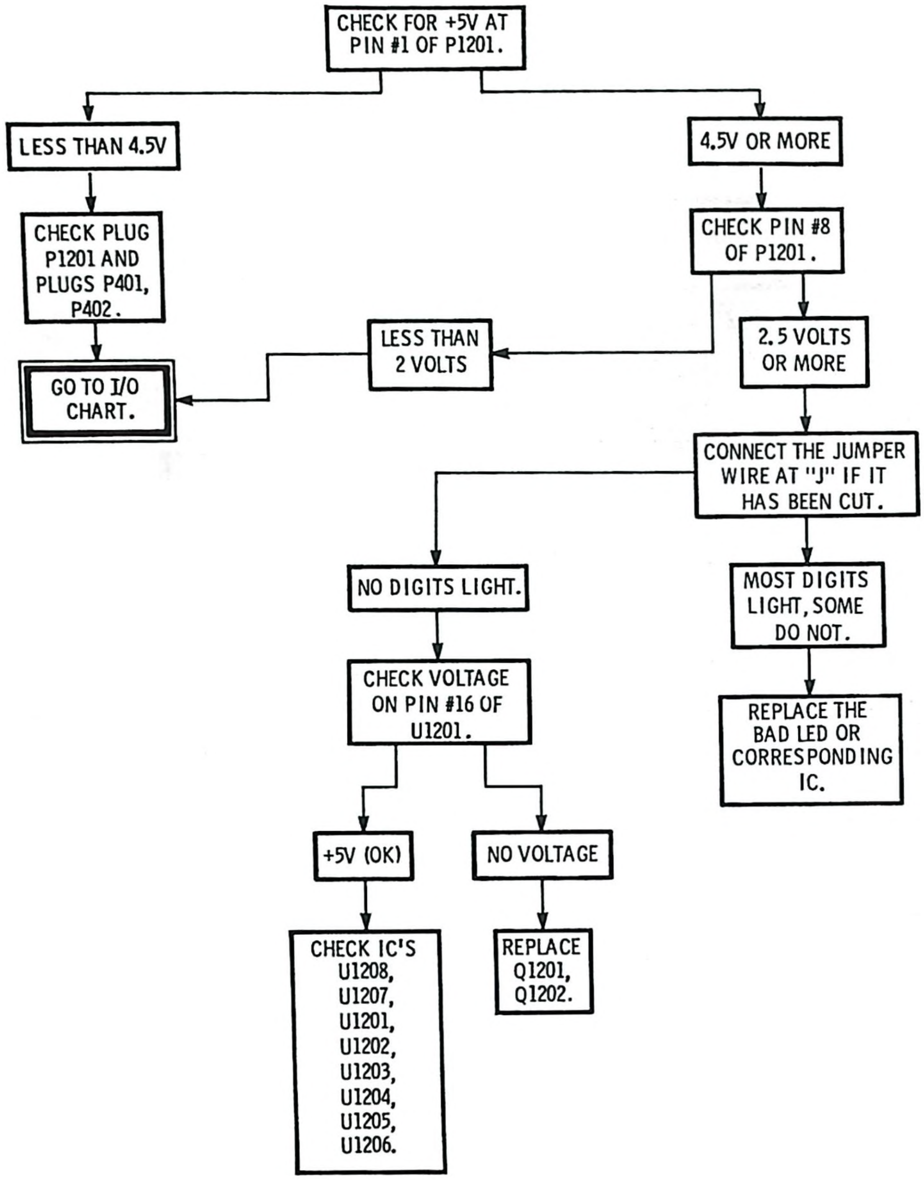
LOGIC +12V AND +5V SUPPLIES



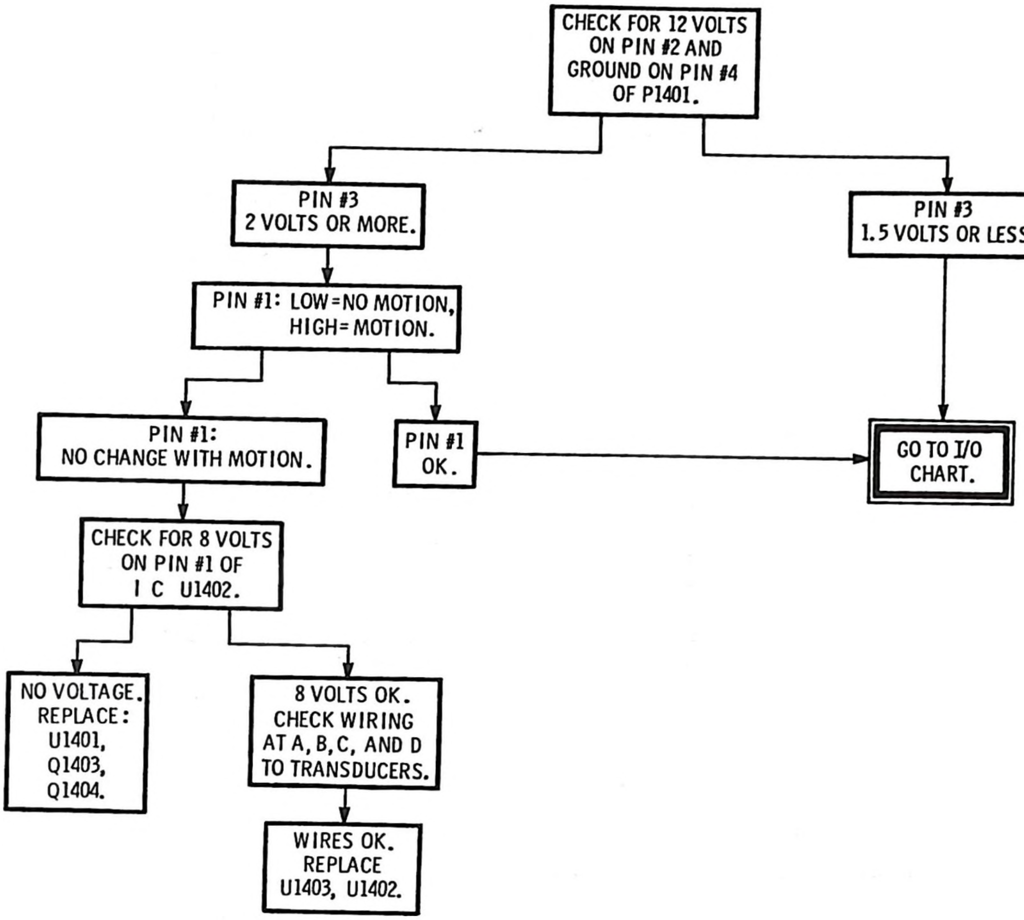
MOTOR +12V SUPPLY



DISPLAY

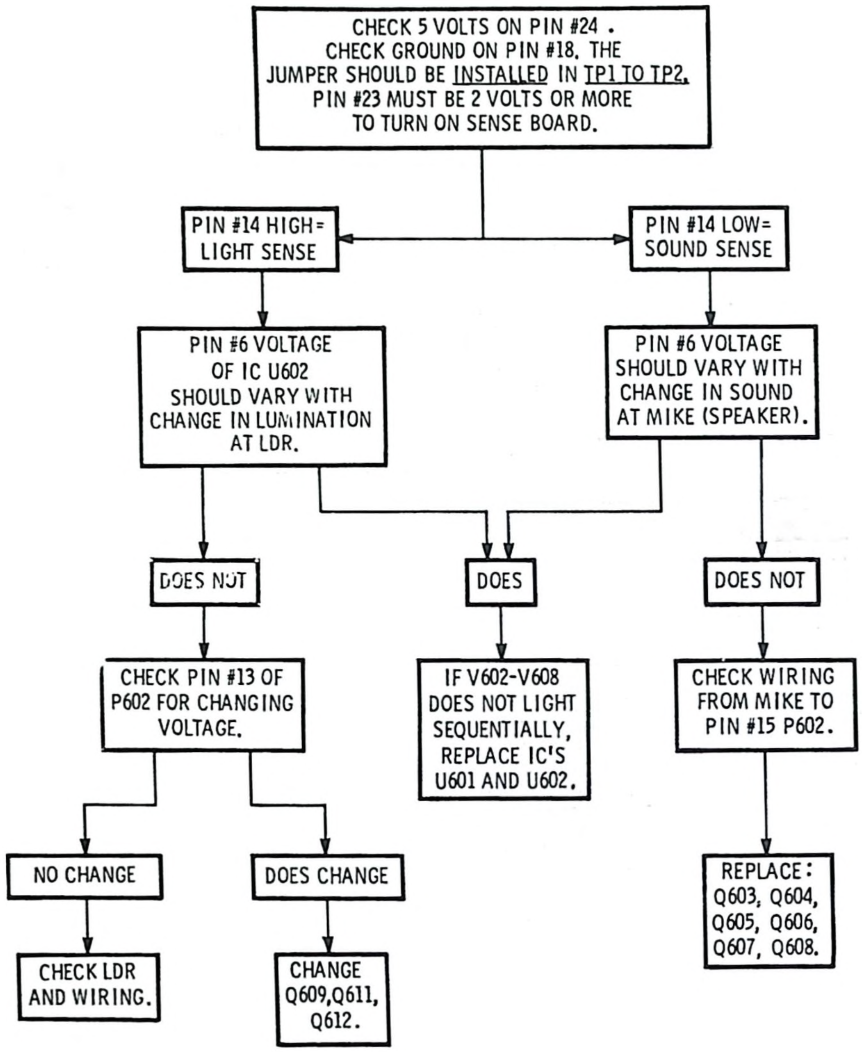

CHECK FOR +5V AT PIN #1 OF P1201.
LESS THAN 4.5V
4.5V OR MORE
CHECK PLUG P1201 AND PLUGS P401, P402.
CHECK PIN #8 OF P1201.
LESS THAN 2 VOLTS
2.5 VOLTS OR MORE
GO TO I/O CHART.
CONNECT THE JUMPER WIRE AT "J" IF IT HAS BEEN CUT.
NO DIGITS LIGHT.
MOST DIGITS LIGHT, SOME DO NOT.
CHECK VOLTAGE ON PIN #16 OF U1201.
REPLACE THE BAD LED OR CORRESPONDING IC.
+5V (OK)
NO VOLTAGE
CHECK IC'S U1208, U1207, U1201, U1202, U1203, U1204, U1205, U1206.
REPLACE Q1201, Q1202.

MOTION

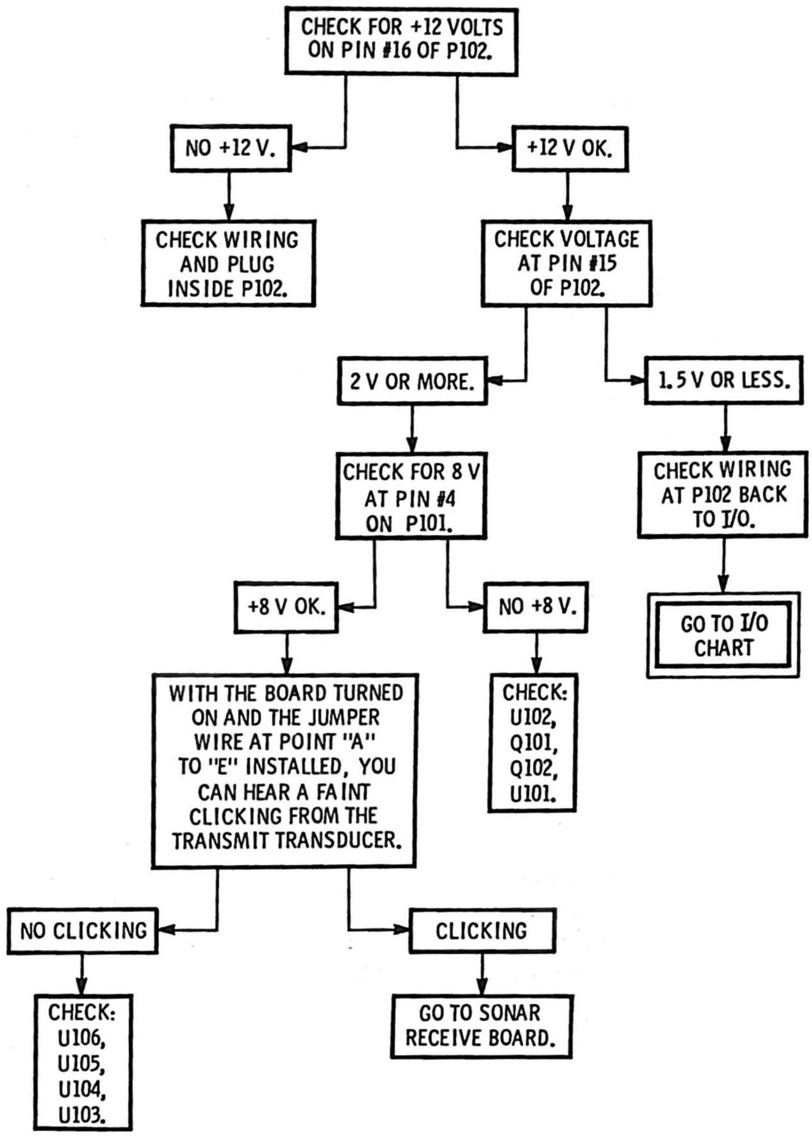


NOTE: Ultrasonic devices can be fooled by ultrasonic "noise" coming from motors, jingling keys, or other common sources. If the circuitry seems to be operating correctly but the results of operation are irregular, suspect the presence of ultrasonic noise.

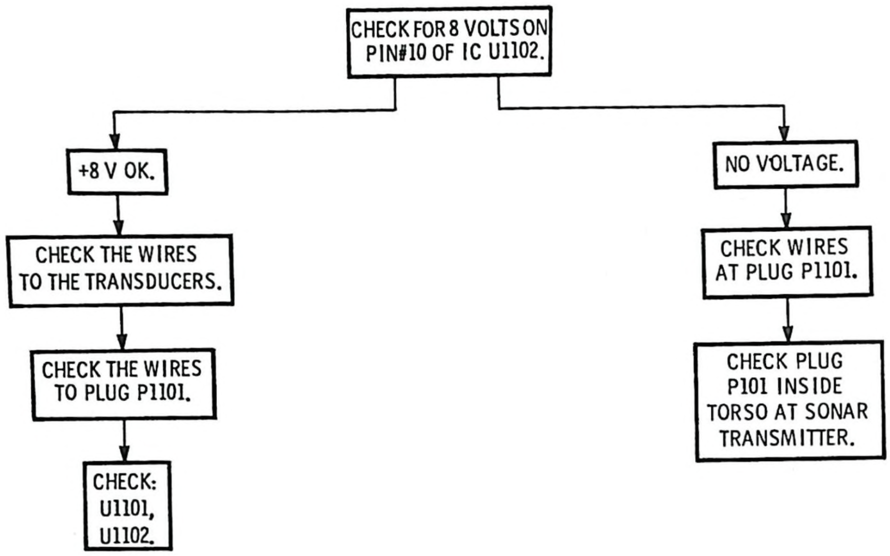
SENSE


CHECK 5 VOLTS ON PIN #24 .
CHECK GROUND ON PIN #18. THE
JUMPER SHOULD BE INSTALLED IN TP1 TO TP2.
PIN #23 MUST BE 2 VOLTS OR MORE
TO TURN ON SENSE BOARD.
PIN #14 HIGH=
LIGHT SENSE
PIN #14 LOW=
SOUND SENSE
PIN #6 VOLTAGE
OF IC U602
SHOULD VARY WITH
CHANGE IN LUMINATION
AT LDR.
PIN #6 VOLTAGE
SHOULD VARY WITH
CHANGE IN SOUND
AT MIKE (SPEAKER).
DOES NOT
DOES
DOES NOT
CHECK PIN #13 OF
P602 FOR CHANGING
VOLTAGE.
IF V602-V608
DOES NOT LIGHT
SEQUENTIALLY,
REPLACE IC'S
U601 AND U602.
CHECK WIRING
FROM MIKE TO
PIN #15 P602.
NO CHANGE
DOES CHANGE
REPLACE:
Q603, Q604,
Q605, Q606,
Q607, Q608.
CHECK LDR
AND WIRING.
CHANGE
Q609,Q611,
Q612.

SONAR TRANSMIT



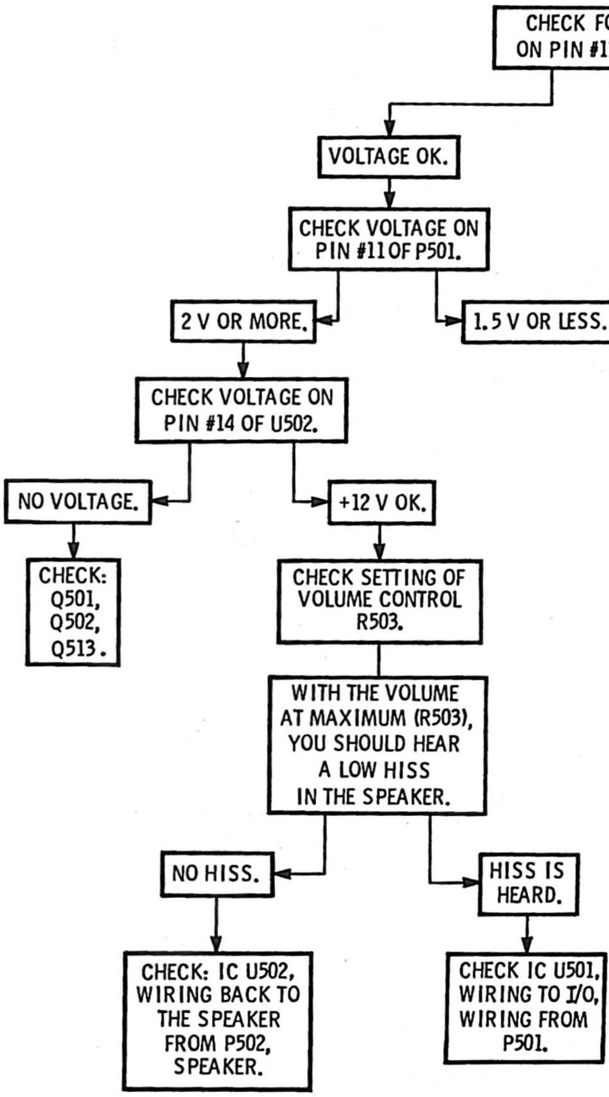
SONAR RECEIVE*

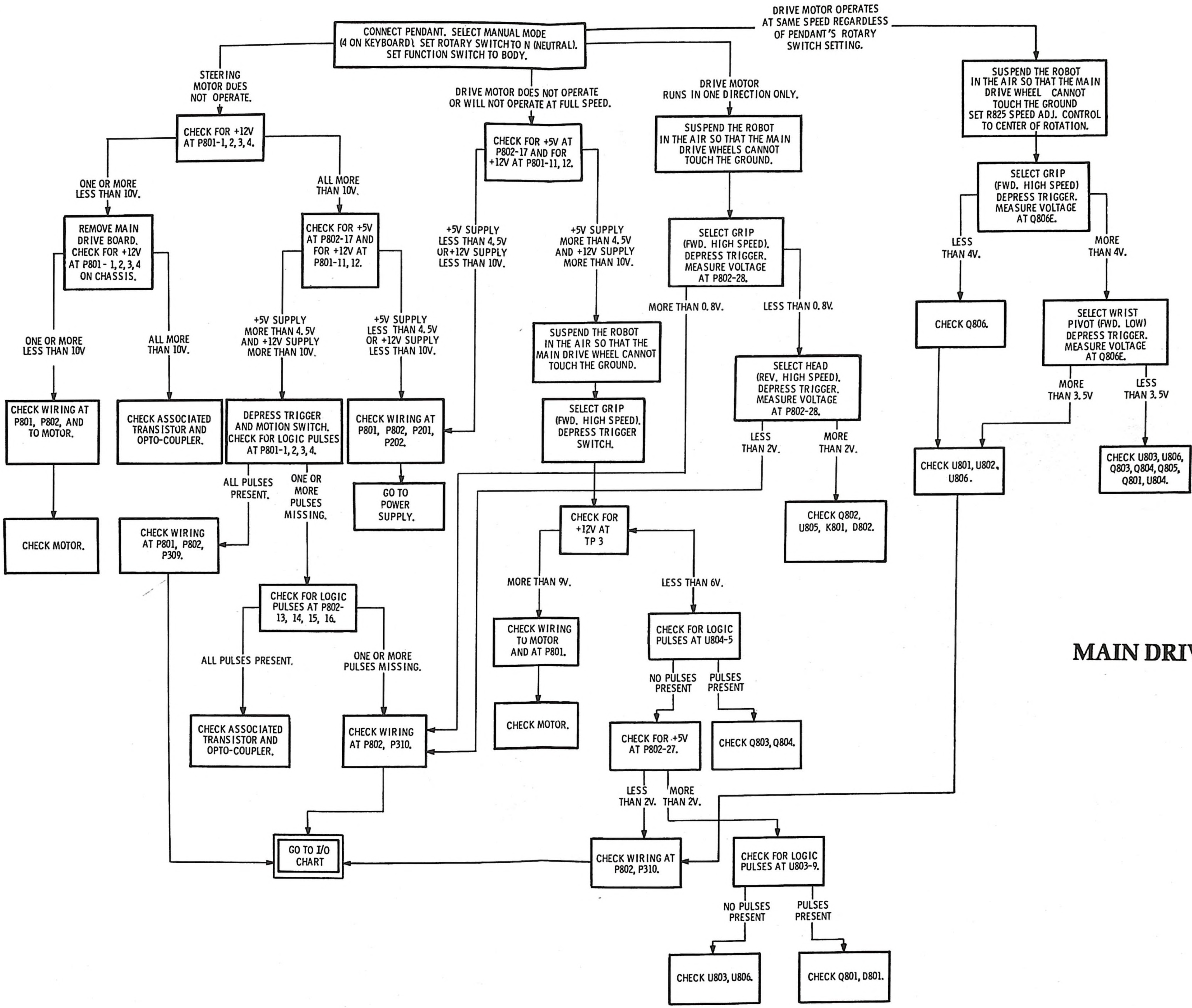


*CHECK SONAR TRANSMIT FIRST.

NOTE: Ultrasonic devices can be fooled by ultrasonic "noise" coming from motors, jingling keys, or other common sources. If the circuitry seems to be operating correctly but the results of operation are irregular, suspect the presence of ultrasonic noise.

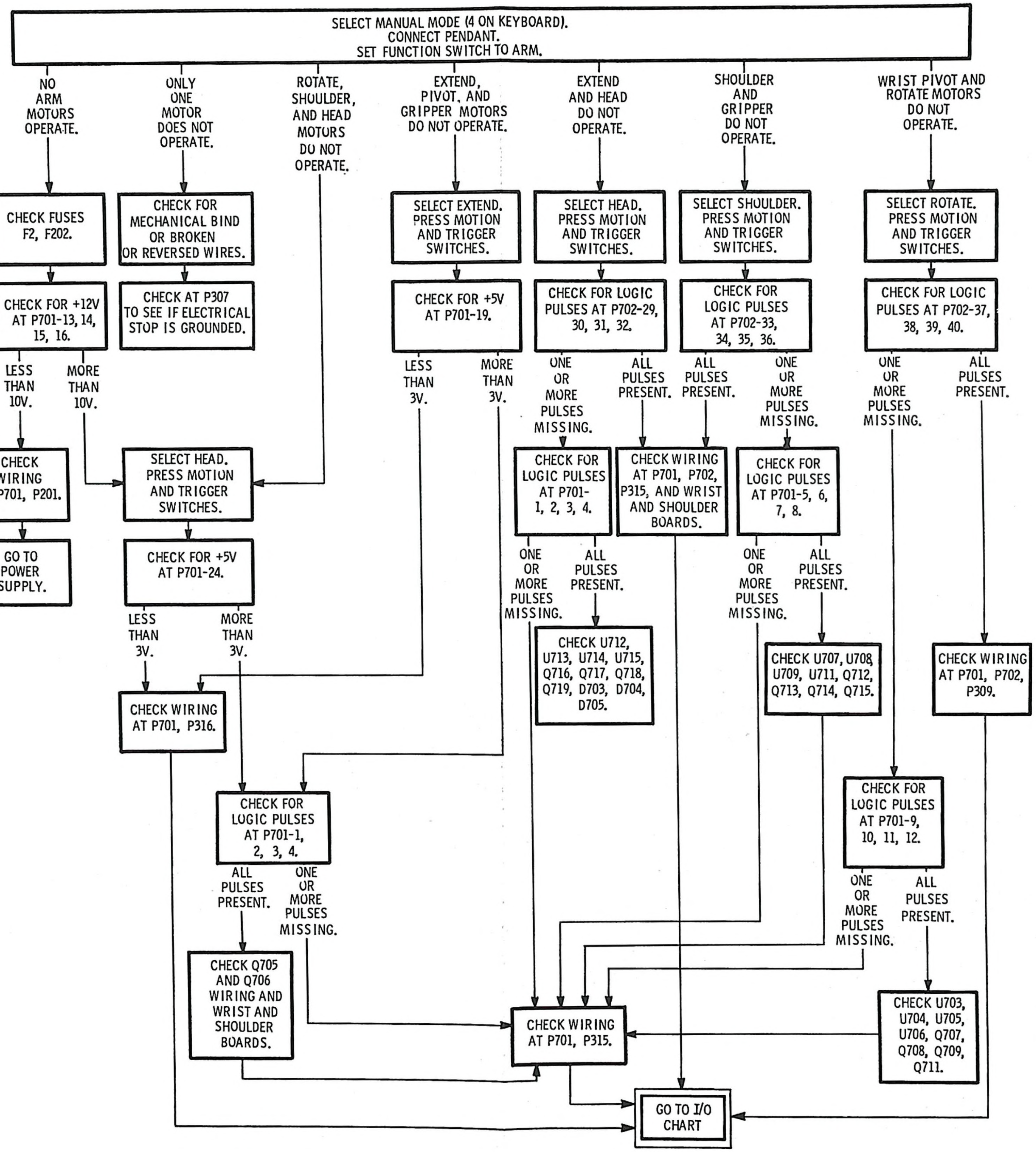
SPEECH


CHECK FOR +12 V ON PIN #13 OF P502.
VOLTAGE OK.
NO VOLTAGE.
CHECK VOLTAGE ON PIN #11 OF P501.
CHECK WIRING TO I/O.
GO TO I/O CHART
2 V OR MORE.
1.5 V OR LESS.
CHECK VOLTAGE ON PIN #14 OF U502.
NO VOLTAGE.
+12 V OK.
CHECK: Q501, Q502, Q513.
CHECK SETTING OF VOLUME CONTROL R503.
WITH THE VOLUME AT MAXIMUM (R503), YOU SHOULD HEAR A LOW HISS IN THE SPEAKER.
NO HISS.
HISS IS HEARD.
CHECK: IC U502, WIRING BACK TO THE SPEAKER FROM P502, SPEAKER.
CHECK IC U501, WIRING TO I/O, WIRING FROM P501.


CONNECT PENDANT. SELECT MANUAL MODE (4 ON KEYBOARD) SET ROTARY SWITCH TO N (NEUTRAL). SET FUNCTION SWITCH TO BODY.
STEERING MOTOR DOES NOT OPERATE.
DRIVE MOTOR DOES NOT OPERATE OR WILL NOT OPERATE AT FULL SPEED.
DRIVE MOTOR RUNS IN ONE DIRECTION ONLY.
DRIVE MOTOR OPERATES AT SAME SPEED REGARDLESS OF PENDANT'S ROTARY SWITCH SETTING.
CHECK FOR +12V AT P801-1, 2, 3, 4.
ONE OR MORE LESS THAN 10V.
ALL MORE THAN 10V.
REMOVE MAIN DRIVE BOARD. CHECK FOR +12V AT P801-1, 2, 3, 4 ON CHASSIS.
ONE OR MORE LESS THAN 10V
ALL MORE THAN 10V.
CHECK WIRING AT P801, P802, AND TO MOTOR.
CHECK MOTOR.
CHECK ASSOCIATED TRANSISTOR AND OPTO-COUPLER.
CHECK FOR +5V AT P802-17 AND FOR +12V AT P801-11, 12.
+5V SUPPLY MORE THAN 4.5V AND +12V SUPPLY MORE THAN 10V.
+5V SUPPLY LESS THAN 4.5V OR +12V SUPPLY LESS THAN 10V.
DEPRESS TRIGGER AND MOTION SWITCH. CHECK FOR LOGIC PULSES AT P801-1, 2, 3, 4.
ALL PULSES PRESENT.
ONE OR MORE PULSES MISSING.
CHECK WIRING AT P801, P802, P309.
CHECK FOR LOGIC PULSES AT P802-13, 14, 15, 16.
ALL PULSES PRESENT.
ONE OR MORE PULSES MISSING.
CHECK ASSOCIATED TRANSISTOR AND OPTO-COUPLER.
CHECK WIRING AT P802, P310.
CHECK WIRING AT P801, P802, P201, P202.
GO TO POWER SUPPLY.
GO TO I/O CHART
CHECK FOR +5V AT P802-17 AND FOR +12V AT P801-11, 12.
+5V SUPPLY LESS THAN 4.5V OR +12V SUPPLY LESS THAN 10V.
+5V SUPPLY MORE THAN 4.5V AND +12V SUPPLY MORE THAN 10V.
SUSPEND THE ROBOT IN THE AIR SO THAT THE MAIN DRIVE WHEEL CANNOT TOUCH THE GROUND.
SELECT GRIP (FWD. HIGH SPEED). DEPRESS TRIGGER SWITCH.
CHECK FOR +12V AT TP 3
MORE THAN 9V.
LESS THAN 6V.
CHECK WIRING TO MOTOR AND AT P801.
CHECK MOTOR.
CHECK FOR LOGIC PULSES AT U804-5
NO PULSES PRESENT
PULSES PRESENT
CHECK FOR +5V AT P802-27.
CHECK Q803, Q804.
LESS THAN 2V.
MORE THAN 2V.
CHECK WIRING AT P802, P310.
CHECK FOR LOGIC PULSES AT U803-9.
NO PULSES PRESENT
PULSES PRESENT
CHECK U803, U806.
CHECK Q801, D801.
SUSPEND THE ROBOT IN THE AIR SO THAT THE MAIN DRIVE WHEELS CANNOT TOUCH THE GROUND.
SELECT GRIP (FWD. HIGH SPEED). DEPRESS TRIGGER. MEASURE VOLTAGE AT P802-28.
MORE THAN 0.8V.
LESS THAN 0.8V.
SELECT HEAD (REV. HIGH SPEED). DEPRESS TRIGGER. MEASURE VOLTAGE AT P802-28.
LESS THAN 2V.
MORE THAN 2V.
CHECK Q802, U805, K801, D802.
SUSPEND THE ROBOT IN THE AIR SO THAT THE MAIN DRIVE WHEEL CANNOT TOUCH THE GROUND SET R825 SPEED ADJ. CONTROL TO CENTER OF ROTATION.
SELECT GRIP (FWD. HIGH SPEED) DEPRESS TRIGGER. MEASURE VOLTAGE AT Q806E.
LESS THAN 4V.
MORE THAN 4V.
CHECK Q806.
SELECT WRIST PIVOT (FWD. LOW) DEPRESS TRIGGER. MEASURE VOLTAGE AT Q806E.
MORE THAN 3.5V
LESS THAN 3.5V
CHECK U801, U802, U806.
CHECK U803, U806, Q803, Q804, Q805, Q801, U804.

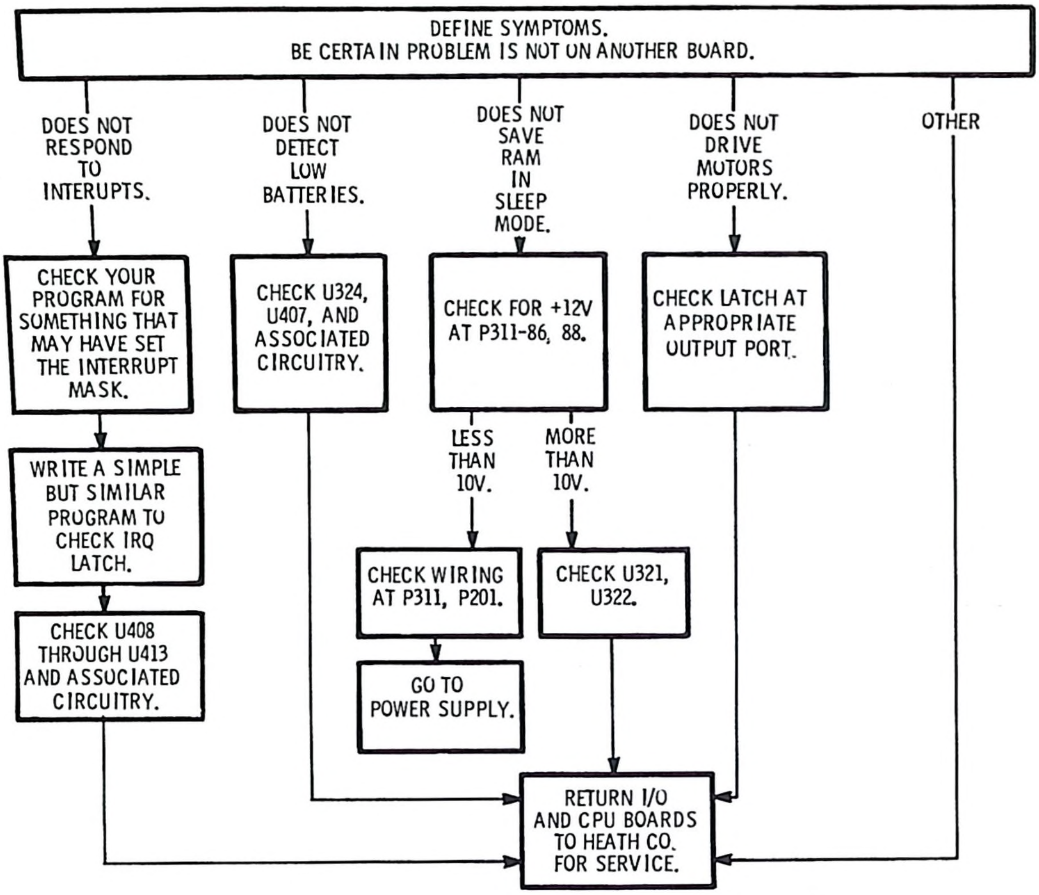
MAIN DRIVE

ARM DRIVE



SELECT MANUAL MODE (4 ON KEYBOARD). CONNECT PENDANT. SET FUNCTION SWITCH TO ARM.
NO ARM MOTORS OPERATE.
ONLY ONE MOTOR DOES NOT OPERATE.
ROTATE, SHOULDER, AND HEAD MOTORS DO NOT OPERATE.
EXTEND, PIVOT, AND GRIPPER MOTORS DO NOT OPERATE.
EXTEND AND HEAD DO NOT OPERATE.
SHOULDER AND GRIPPER DO NOT OPERATE.
WRIST PIVOT AND ROTATE MOTORS DO NOT OPERATE.
CHECK FUSES F2, F202.
CHECK FOR MECHANICAL BIND OR BROKEN OR REVERSED WIRES.
SELECT EXTEND. PRESS MOTION AND TRIGGER SWITCHES.
SELECT HEAD. PRESS MOTION AND TRIGGER SWITCHES.
SELECT SHOULDER. PRESS MOTION AND TRIGGER SWITCHES.
SELECT ROTATE. PRESS MOTION AND TRIGGER SWITCHES.
CHECK FOR +12V AT P701-13, 14, 15, 16.
CHECK AT P307 TO SEE IF ELECTRICAL STOP IS GROUNDED.
CHECK FOR +5V AT P701-19.
CHECK FOR LOGIC PULSES AT P702-29, 30, 31, 32.
CHECK FOR LOGIC PULSES AT P702-33, 34, 35, 36.
CHECK FOR LOGIC PULSES AT P702-37, 38, 39, 40.
LESS THAN 10V.
MORE THAN 10V.
LESS THAN 3V.
MORE THAN 3V.
ONE OR MORE PULSES MISSING.
ALL PULSES PRESENT.
CHECK WIRING AT P701, P201.
GO TO POWER SUPPLY.
SELECT HEAD. PRESS MOTION AND TRIGGER SWITCHES.
CHECK FOR +5V AT P701-24.
CHECK FOR LOGIC PULSES AT P701-1, 2, 3, 4.
CHECK WIRING AT P701, P702, P315, AND WRIST AND SHOULDER BOARDS.
CHECK FOR LOGIC PULSES AT P701-5, 6, 7, 8.
CHECK U712, U713, U714, U715, Q716, Q717, Q718, Q719, D703, D704, D705.
CHECK U707, U708, U709, U711, Q712, Q713, Q714, Q715.
CHECK WIRING AT P701, P702, P309.
CHECK WIRING AT P701, P316.
CHECK FOR LOGIC PULSES AT P701-1, 2, 3, 4.
CHECK FOR LOGIC PULSES AT P701-9, 10, 11, 12.
CHECK Q705 AND Q706 WIRING AND WRIST AND SHOULDER BOARDS.
CHECK WIRING AT P701, P315.
CHECK U703, U704, U705, U706, Q707, Q708, Q709, Q711.
GO TO I/O CHART

INPUT/OUTPUT (I/O AND CPU)



THEORY OF OPERATION

This section of the Manual is primarily intended as a technical service aid for the person with some background in electronics and microprocessor operation. The following descriptions, along with the schematic diagrams, should give such a person all that is needed to service and understand the Robot. However, you may also use this information if you are not a trained technician to gain considerable understanding of how microprocessor-controlled devices work.

The block/interconnect diagram and discussion briefly deal with the whole Robot electronic package. Following that are sections that discuss each Robot circuit board in depth. Refer to the corresponding schematic diagrams while reading these discussions.

BLOCK/INTERCONNECT DIAGRAM

The Block/Interconnect Diagram shows the basic electronic Robot without the distraction of the mechanical considerations. The microprocessor (CPU) board and the I/O (input/output) boards are shown together, just as they are connected together in the Robot.

The I/O and CPU are the brains of your Robot; with the CPU receiving information, storing data and instructions, making decisions, and sending out instructions. The I/O supports this process by guiding the heavy traffic of data into and out of the CPU.

Each input or output is given an address in the hexadecimal numbering system, just like a memory address. The CPU then "reads" from and "writes" to a specific address (courtesy of the I/O board) to communicate with the Robot senses.

The keyboard provides a special type of communication, since this is the method of direct input to the Robot. You can communicate through the keys, and the Robot responds through the display.

The Robot's eyes, ears, and other senses are provided by the circuit boards. The sense board measures light and noise levels for the CPU. Ultrasonic ranging boards transmit and receive the signal to measure the distance to an object in front of the transmitter. The motion detect board senses amplitude modulation of its ultrasonic signal whenever there is motion nearby. The Robot's clock and calendar are located on the I/O board and are available for applications requiring the measurement of time.

The Robot's muscles are the motors. Most motors are stepper motors, which move a specific distance in response to a pattern of pulses from the CPU. These pulses are amplified by the arm drive or main drive boards. By keeping track of the number of pulses sent to these motors, the CPU can tell where the motor shaft is positioned.

The remaining motor is a permanent magnet, DC voltage, drive motor. Its position is recorded by the disc and sensor mounted on the drive wheel.

The heart of the Robot is the charger, along with the Robot's power supply board and battery system. There are two 12 volt systems in the Robot. One supplies power to the logic operations and the other supplies power for the motors. The logic voltage to the sensors is turned on and off at each sense board in response to commands from the CPU.

The experimental board gives you the opportunity to produce circuits and connect them to the Robot's CPU. See the circuit description and the port map for more information.

The optional speech board responds to commands from the CPU by sending phonemes (synthesized parts of human speech) to a speaker. By grouping the phonemes properly, the CPU can cause the speech board to create words and phrases.

CIRCUIT DESCRIPTION

CHARGER

The charger is a separate unit which furnishes unregulated DC current to the Robot's charging circuits. The power supply circuit in the Robot controls this power for charging the Robot batteries or operating the Robot. A red (power on) light and a white (charging rate) light are mounted on the charger's front panel with the on-off switch. Refer to the charger schematic while reading the rest of this description.

Line voltage from the power plug is stepped down to approximately 20 volts AC by transformer T1. This voltage lights red "power on" light V1 whenever the charger is on. The 20 VAC secondary voltage is also applied through the combination of R1 and R2 to full-wave bridge rectifier BR1. The resistor combination of R1 and R2 serves two functions: (1) It limits the maximum current from the charger so the Robot's batteries will not be damaged; and (2) it provides a voltage across it, which is proportional to the charge current, causing light V2 to glow with a brilliance directly proportional to charge current.

The output of bridge rectifier BR1 is filtered by C1 and goes to connector plug P101.

Jumpers in the output plug (P101) allow isolation of the batteries in the Robot if the charger is not plugged in. This is to avoid draining the batteries through the Power Supply circuitry.

POWER SUPPLY

The power supply has duplicate circuitry to charge and monitor the status of the two independent 12 volt battery systems on the Robot. This board has two charger regulating circuits, two 5.6 volt reference sources, two low voltage sense circuits, and a single +5 volt switching regulator. The following discussions, deal with only one regulator, reference, or sense circuit (the one for the motor batteries). The discussion applies equally, however, to the identical circuit for the logic batteries.

Charger Regulating Circuits

The charger regulating circuits provide a variable charging current to the batteries. The charger operates in a current limit mode to deliver the maximum safe charging current when the batteries are low. Then it automatically switches to a constant voltage float-charge mode when the batteries approach full charge.

When the batteries are near full charge, the current drawn from the charger is relatively small and there is approximately 27 VDC at terminal 23, the input to regulator U204. The output of the regulator is set by the position of R225 and is maintained at 13.9 volts to provide a float charge to the batteries. The output leaves the circuit board at pin 22 and goes through the jumper on plug P101, and then returns to the board at pin 21. This float charge can be continued indefinitely without damage to the batteries.

However, when the batteries are nearly discharged, the current drawn from the charger is near maximum. Then the internal impedance of the charger limits the output to as little as 14 volts. Below 16 volts, U204 acts essentially as a resistor whose voltage drop varies with current. Thus, the maximum charging current to the batteries is limited by the internal impedance of the charger and the impedance of U204.

The 5.6 Volt Reference Circuit

For maximum battery life, the regulator circuits must be accurate. This reference circuit has been provided on the circuit board to help you accurately adjust the charger regulators.

Transistors Q205 and Q206 are connected as a constant current source for reference diode D206. This maintains the voltage across D206 at 5.6 volts within a very close tolerance.

When the regulator is properly adjusted to provide the desired 13.9 volts, it will also provide 5.6 volts at TP8. This 5.6 volts is due to the precision voltage divider R226 and R227. In this way, you can use a voltmeter between TP9 and TP8 and adjust R225 to give a zero reading. This method minimizes voltmeter error by using a zero reading.

Low Voltage Sense Circuit

The low voltage sense circuit is a comparator which uses the battery voltage and the voltage from the reference circuit. Whenever the battery voltage falls below 10 volts, the comparator sends a low voltage signal to the Robot's microprocessor.

The 5.6 volt reference voltage developed across D206 is also applied to the noninverting input of comparator U205. The battery voltage is applied across voltage divider R233 and R234 to the inverting input of comparator U205. If the batteries are fully charged, the voltage at voltage divider will be greater than the voltage at the noninverting input. Therefore, the output of the comparator at pin 7 will be low.

But if the battery voltage falls below 10 volts, the voltage appearing at the inverting input to the comparator will fall below 5.6 volts. When this happens, the output of the comparator goes high signaling a low battery condition.

Five Volt Switching Regulator

All 5-volt logic power on the Robot is furnished from the switching regulator, made up of U203, Q203, Q204 and their associated circuitry. The circuit operates by rapidly switching series pass transistor Q204 on and off. The ratio of the on time to the off time determines the average voltage applied to the load.

U203 is a pulse width modulator. It functions by comparing a fraction of the output voltage (at pin 1) with a fixed reference voltage (at pin 2). The component values in this circuit have been chosen so that whenever the output voltage falls below 5 volts, the circuit responds by increasing the width of the pulse appearing at pins 12 and 13. This results in increasing the on time for Q203 and Q204, which raises the average output voltage. An increase in the output voltage causes the pulse width at pins 12 and 13 to decrease, resulting in a decrease in the average output voltage. The net result is to maintain the output voltage at 5 volts. Switching transients which would normally appear on the output are removed by the action of L201, C206, and C207.

TEACHING PENDANT (REMOTE)

The teaching pendant notes the position of its four switches (Rotary, Function, Motion, and Trigger) and encodes them. This encoded information goes through the cable to the microprocessor as a 7-bit status word. Refer to the Remote schematic as an aid in understanding the following description.

With the exception of Rotary switch SW904, the switches in the remote unit merely open or close circuits to pull-up resistors (some of which are located elsewhere in the Robot). This action causes the logic level on the associated lines to change. Rotary switch SW904, however, connects to a diode encoder to furnish a 3-bit output at R, S and T. A unique 3-bit output is generated for each of the seven positions of the Rotary switch.

With the Rotary switch as shown, the inputs to U901 A, B, and C are pulled high by the 3300 Ω pull-up resistors (R902, R904, and R905). This results in a 000 output at R, S, and T. If the Function switch is now moved to the (Arm Pivot) 4 position, the cathodes of diodes D906 and D907 are connected to ground. This causes a low level to appear at the inputs to U901 A and C, resulting in a 101 output at R, S and T. In a similar manner, other positions of the Function switch ground different combinations of diodes, resulting in a unique output code for each switch position.

MAIN DRIVE BOARD

Circuits on the main drive board control the speed and direction of the main drive motor and provide the power interface between the CPU and the steering and drive motors. Refer to the main drive circuit board schematic while reading the following description.

Steering Motor Interface

The four windings from the steering motor are connected to connector P801, pins 1 thru 4, on the main drive board. A pulse pattern from the CPU arrives at connector P802, pins 13 through 16, when the steering motor is to be set in motion. A low pulse on pin 13, for example, causes the diode in U811 to emit light. This, in turn, causes the transistor in U811 to conduct, which turns on Q811. Since one of the windings in the steering motor is connected to the collector of Q811, the result is a current flow in the motor winding.

As the pulse pattern varies on pins 13 thru 16, the current pulses in the motor windings connected to pins 1 thru 4 vary accordingly, to turn the motor.

Main Drive

The main drive motor is a DC permanent magnet type. Its direction of rotation is determined by the direction of the current flow through its single winding. Motor speed is determined by the average current flow through the winding.

This average current is controlled by a pulse width modulator (PWM) circuit, U803A, U803B, and its associated components. The output of this circuit, which appears at pin 3 of U803B, is a rectangular waveform whose duty cycle is controlled by the voltage present at pin 11 of IC U803B. This control voltage is determined by the back EMF (electro-motive-force) of the drive motor (which we will discuss later) and speed commands from the CPU.

Speed commands appear on pins 21 through 27 of plug P802 as a 6-bit binary word. This information is applied to the FET switches inside IC's U801 and U802.

The combination of switches turned on at any instant determines the voltage at pin 8 of U801. This is applied to pin 3 of difference amplifier U806. The output at pin 4 is applied to the voltage control input of the pulse width modulator, pin 11 of U803B, and sets the basic speed of the drive motor.

The back EMF of the motor is sensed at the collector of Q804 and fed to the positive input of difference amplifier U806 via resistors R822 and R824. Any variation in the motor speed is reflected as a level change in the back EMF. This EMF change is coupled from pin 4 of U806 to pin 11 of U803B, and provides speed control. Q805 is used as a switch to hold the input of U806 pin 2 low during the time the motor receives a drive pulse.

When the pulse at pin 9 of U803B is low, D801 and Q801 conduct, turning on the transistor in U804. This turn son pre-driver Q803 and motor driver Q804 conducting current through the motor windings connected across pins 7 and 8 and 9 and 10 of P801. Q801 is used as a switch to turn off the motor when the data at pin 27 is low.

The direction of current flow through the motor is determined by relay K801. In the energized position, as shown, current to the motor flows out pins 7 and 8, through the motor, then back into pins 9 and 10. With the relay deenergized, motor current flows in the opposite direction. The relay is controlled by the logic level appearing at pin 28 of the circuit board. This level is set by the CPU to determine drive motor direction.

ARM DRIVE

The two-phase signal pattern necessary to drive the various stepper motors on the robot is generated by the microprocessor. This pattern determines whether the motor runs forward or backward, and how fast. The arm drive board increases the power capacity of the pattern, and aids in the selection of the proper motor.

There are two different types of circuits on this board. The first is the "supply select" circuits (there are two), which supply the +12 volt motor voltage to the motor windings. The second is the "pattern select" circuits (there are three), which are controlled by the signal pattern from the microprocessor. This type of circuit controls the current flow through the motor windings by switching the ground side of the windings.

The figure below shows how the combination of any one pattern select circuit and one supply select circuit is used to select a motor. For example, if you wished to run the gripper motor, the microprocessor would use the pattern select 2 circuit and the supply select 2 circuit.

	PATTERN SELECT 1 CIRCUIT	PATTERN SELECT 2 CIRCUIT	PATTERN SELECT 3 CIRCUIT
SUPPLY SELECT 1 CIRCUIT	WRIST ROTATE	SHOULDER	HEAD
SUPPLY SELECT 2 CIRCUIT	WRIST PIVOT	GRIPPER	EXTEND

The supply select circuits are U701 through Q703 and U702 through Q706. Supply select circuit 1 (the U701 circuit) is selected as an example. A high from the microprocessor turns on photocoupler U701. It then turns on Q701, Q702, and Q703, which supplies +12 volts to the windings of the wrist rotate, shoulder, and head motors. Diode D701 protects against inductive voltages generated in the windings. In the same manner, the supply select 2 circuit would supply voltage to the wrist pivot, gripper and extend motors.

The pattern select circuits operate as follows. Pattern select circuit 2 is chosen as the example. This means that the ground side of the windings from the shoulder and the gripper motors will be affected. The pattern comes in to optical couplers U707 through U711. A high at the coupler turns on the Darlington pair (Q712 through Q715) to connect the motor winding to ground and let current flow. The order and speed of the pattern determines how fast and in what direction the motor runs. Diodes D703 through D706 serve to isolate the windings of the head motor from the extend motor supply voltage. Other diodes in the shoulder and wrist boards serve the same purpose for the other five stepper motors.

The above combination of supply select circuit 1 and pattern select circuit 2 would cause the shoulder motor to run.

MOTORS

There are two types of motors in your Robot. The main drive motor is called a DC permanent magnet type. This kind of motor turns in the direction determined by the polarity of the voltage applied to it. The speed of the motor is controlled by the average magnitude of the applied voltage. In the Robot, we change motor direction by reversing the polarity of the voltage, and vary the speed by varying the "off" and "on" time of the applied voltage (thereby varying the average voltage). This is called pulse width modulating the applied voltage.

The other type of motor used in the Robot is the stepper motor. This type is used for all motors except the drive motor. A stepper motor turns in response to a four-step pulse sequence applied to its windings. The motor rotates a given distance for each pulse set. Reversing the sequence of the pulses reverses the movement of the motor. The microprocessor keeps track of the position of the motor by keeping track of the pulse sequences applied to the motor. This information is kept in memory as long as the Robot is turned on, but is lost when the Robot is turned off. The "initialize" operation is then used to properly position all motors (except the drive motor) when the Robot is turned back on again.

ULTRASONIC RANGING (SONAR) TRANSMITTER

The ultrasonic ranging system transmits a short ultrasonic pulse and then measures the time until the first echo of the pulse returns. This time is proportional to the distance the sound has traveled.

The ultrasonic ranging system is turned on by a high input from the microprocessor at P102 pin 15. This turns Q101 on, allowing the +12 volts from pin 16 to reach the 8-volt regulator. The +8 volts from the regulator supplies both the transmitter board and the receiver board (through P101 pin 4).

The transmitter circuit generates a signal consisting of eight cycles of ultrasonic (32 kHz) energy to drive the ultrasonic transmit transducer. In addition, it sends signals to control the receiver and timer (for measuring how long it takes for the ultrasonic pulse's echo to return). It operates in the following manner:

U106A is configured as an RC oscillator (frequency adjustable at R108) operating at approximately 32 kHz. The output of the oscillator is fed to NOR gate U105D. Then, when U105D is enabled by a low voltage on its pin 12 input, it passes the 32 kHz signal on to U106B & C to drive the transmit transducer.

This is how U105 handles the 32 kHz signal and generates other operating signals: The 32 kHz signal is also used for timing the circuit operations. It goes to a 12-stage binary counter, U103. The 12th stage output goes to U104, which is used as a 2-stage counter. This makes the Q output of U104 (pin 13) into the output of a 14-stage counter. This output (pin 13) goes high after 4096 counts. By going high, it sets the latch (U105A & B) and the inverter (U105C), and enables gate U105D with a low. Then the 32 kHz signal is transmitted.

At the same time, the high signal from the Q output of the 14-stage counter also resets both U103 (pin 11) and U104 (pins 4 and 10), to start a new count cycle. After counting eight more cycles from the 32 kHz oscillator, U103 pin 5 goes high. This resets the latch-inverter of U105 (at pin 6) so that U105D stops passing the signal for the transmitter transducer. The 14-stage counter then continues for about 1/4 second until it reaches 4096 again and the cycle repeats.

More things happen at U106, however, than just the passing of eight cycles of signal. When U105 pin 12 goes low, it also discharges C112 quickly and sends the output of U106D high. This is a blanking signal to inhibit the receiver during transmission. At the end of the eight cycles, pin 12 goes high, but R112 and C112 delay U106D for a moment; this allows any transmitter ringing to die away before the receiver starts.

An additional output from the same point (to P101 pin 8) resets the receiver (which turns off after receiving the first echo) so it will be ready when the blanking pulse is removed. A final output (from U105 pin 4) goes high during the eight-cycle transmission to start the timer on the I/O board. This timer records how long the echo takes to return, thus measuring the distance.

ULTRASONIC RANGING (SONAR) RECEIVER

The receiver waits for the first echo of the transmitter's sonic pulse and notifies the timer (on the I/O board) when the echo is received at transducer A12.

The received echo is amplified by a two-stage amplifier consisting of U1101 and U1102. This offers a 15 MHz bandwidth and a total gain of 80 dB. The inputs of these ICs are biased to half of the supply voltage by voltage divider R1107/R1108. The output of U1102 is made compatible with U1103 by C1108, R1109, and clamping diode D1101.

A blanking pulse from the transmit board keeps U1103A from passing U1102's first signal (that of the actual transmitted pulse). After the blanking pulse is removed, the positive-going echo signal from U1102 is applied to pin 2 of U1103A, causing it to trigger. U1103 is a dual one-shot whose output pulse width for the "A" stage is determined by the values of R1112 and C1111. Thus, the output of U1103A at pin 13 will always be a constant amplitude, constant width pulse, regardless of the strength of the received echo. This pulse leaves the board from point C and goes to the sonar timer circuits (on the I/O board).

The pulse from U1103A also returns to U1103B on pin 9. This causes U1103B to trigger and send a negative-going pulse back to the clear input (pin 3) of U1103A. R1113 and C1112 make this pulse last longer than one complete transmit cycle, so U1103A is immediately reset and held in the reset mode after the first echo. This prevents it from responding to echoes received after the first echo. However, U1103A must then be enabled for the next transmit cycle. The reset pulse from the transmit board does this by resetting U1103B at pin 11, which then removes the reset from the A section of U1103 (pin 3).

SENSE BOARD

The sense board accepts inputs from the head-mounted light dependant resistor (LDR1) and dynamic microphone (speaker A15), digitizes them, and makes them available to the microprocessor.

Power to the sense board is controlled by Q601. When a high level enable signal from the microprocessor is present at P602 pin 23, Q602 turns on and drives Q601 into saturation. This supplies +5 volts to all circuits on the sense board.

Sound Sense Amplifier

Inputs from dynamic microphone A15 are received at P602 pin 15. They are amplified by Q603, Q604, Q606, and Q607, and then applied to emitter follower Q608. The output of Q608 is the analog input for analog-to-digital converter U602. Here, the analog signal is converted to an 8-bit digital word which is made available to the microprocessor on P601 pins 5 thru 12.

The 8-bit output of A to D converter U602 is also applied to inverter-driver U601. This device drives the eight light-emitting diodes to provide a visible indication of the digital word furnished to the microprocessor.

Light Sense Amplifier

Input from light dependent resistor LDR1 comes into the sense board on P602 pin 13. This signal is amplified by emitter follower Q612, and then applied to the analog input (pin 6) of the A to D converter. The action of the A to D converter (U602) and led driver (U601) is exactly the same as described for the sound sense amplifier.

Sound/Light Switch

The sound and light sense functions cannot be enabled simultaneously. A signal from the microprocessor at P602 pin 14 determines which of the two functions is selected.

When the signal at P602 pin 14 is high, the light function is selected. Q609 is turned off, turning Q611 off, and the light signal is allowed to reach Q612. At the same time, Q605 is turned on, grounding the sound input to Q606.

When the signal at P602 pin 14 is low, the sound function is selected. Q605 is turned off, allowing the sound signal to reach Q606. However, Q609 is turned on and turns Q611 on also. This grounds the light input to Q612.

Bias Adjustments

R633 (sound) and R646 (light) set the operating bias on the amplifiers. The adjustment of these controls sets the dynamic range over which the sense circuits will respond.

Analog To Digital Converter

The clock timing for A to D converter U602 is furnished by the phase 2 clock signal from the CPU board. This signal comes into the sense board at P602 pin 21 and is then applied to pin 4 of the A to D converter.

MOTION DETECTOR

The motion detector circuit board is located in the head, below the experimental board. It uses ultrasonics to detect any motion around it by sensing changes in the amplitude of the reflected signal. When it detects motion,.it signals the Robot's microprocessor.

It uses two ultrasonic transducers. A transmit transducer floods the immediate area with a continuous 35 kHz signal. The receive transducer constantly picks up the signal reflected from objects in the vicinity. If some reflecting object moves, that will result in variations in the amount of energy reflected back to the receive transducer. The receive circuit detects this as low frequency amplitude modulation. An active low-pass filter detects the low frequency modulating signal and causes a logic high level to be sent, indicating that motion has been detected.

Receive transducer A14 acts as the frequency determining element in an oscillator circuit made up of U1402 E and D and associated components. The output of the oscillator is coupled through U1402 B, C and F to drive transmit transducer A13, which is connected between C and D.

When there is motion within the sensitivity range of the motion detector, the total signal reflected back to the receiver transducer will be amplitude modulated. The low frequency modulation is detected by Q1401 and associated circuitry in the base circuit.

The detected signal is amplified by U1403A and then applied to bandpass amplifier U1403B. The feedback components around U1403B allow it to pass only signals centered around 45 Hz. This low frequency signal is amplified by Q1402 and then applied to U1403C.

U1403C acts as a comparator. Whenever the signal on pin 13 falls below the bias level established by R1417 and R1418 on pin 12, the output on pin 14 switches high. This causes V1401 to light and sends a high level "motion detected" signal to P1401 pin 1.

As with other boards, Q1404 and Q1403 control power to the board, in response to the signal from the microprocessor, and U1401 provides the regulated 8 volts for operation.

SPEECH

Circuits on the speech board can generate the 64 basic sounds, called phonemes, which make up the English language. Under microprocessor control, these 64 sounds may be strung together in any sequence to simulate human speech. The instantaneous pitch of any phoneme may be varied by the microprocessor over four levels to provide inflection. The reference pitch and volume are adjustable with controls mounted on the board.

Refer to the schematic diagram, provided with the speech accessory Manual, as you read the following paragraphs.

A six-bit address from the computer's speech port ($C240) is applied to speech synthesizer U501 through P501 pins 5 through 10. The strobe line (P501 pin 3) then goes high momentarily to latch the address into the synthesizer and begin generation of the phoneme. When phoneme generation is complete, the request line from (P501 pin 4) the synthesizer goes high, signaling the microprocessor that it is ready for the next phoneme. The microprocessor responds by sending a new phoneme address, which starts the process all over again. This sequence continues until all phonemes have been output.

The data coming in on P501 pins 1 and 2 controls the instantaneous pitch of the phoneme being voiced. Manipulating these two lines during phoneme generation produces inflection.

The setting of R15 controls the clock frequency of the synthesizer, which determines the reference pitch and the speed of the phonemes.

Phonemes generated by the synthesizer appear on pins 21 & 22 of U501. From here, they are applied across Volume control R503 and then feed audio amplifier U502. Here they are amplified then routed to P502 pin 15 of the circuit board, which connects to the loudspeaker in the head.

Switch Q501 turns on power to the speech board in response to a high signal from the microprocessor power port through P501 pin 11 and Q502.

DISPLAY BOARD

The display board is the primary means of communication between the operator and the Robot CPU. It contains six 8-segment LED's to output data, and a hexadecimal keypad for inputting data.

Power to the display board is controlled by Q1201 and Q1202. When there is a high input at P1201 pin 8, Q1202 conducts and causes Q1201 to conduct, furnishing +5 volts to all circuits on the board.

Each of the eight segments of each display is directly addressable by the CPU. In other words, the capability exists to illuminate any segment or pattern of segments on the displays, using latches U1201 through U1206. Once illuminated, each segment will remain illuminated until it is turned off. Here is how these segments are controlled.

U1201 through U1206 are 8-bit serial-in parallel-out latches. They are connected in parallel to the data-in and the data address-in lines. All six latches receive a bit of data (from line $\overline{D0}$) at pin 13; a high would cause an LED segment to turn off, a low would cause it to turn on. At the same instant, they are told which LED segment is to be controlled; this "address" is received on pins 1, 2, and 3 (address lines A0, A1, and A2).

However, none of the latches responds to this information unless it receives an enable signal, from U1208, on latch "enable" input pin 14 (output pins 2 through 7 of U1208). The CPU selects the correct latch for each piece of data and directs U1208 via input pins 13, 14, and 15 (address lines A4, A5, and A6). U1208 then sends a low to the selected latch to enable it.

Pin 15 of each latch is the clear input. If it is pulled low (by test jumper J), all latch outputs will go low and all LEDs should light.

The hex keyboard is arranged in a 3 by 6 switch matrix (row 6 has only 1 key, and the Reset key is actually separate). With no key depressed, the row lines which connect to U1209 pins 6, 4, 10, 12, 14, and 2 are pulled high by pull-up resistors R1261 through R1266.

During the key scanning routine, the microprocessor sequentially sends out a low logic level on address lines A0, A1, & A2. Each time the low level is sequenced, data lines D0 thru D5 are read. Any key closure will result in a unique combination of address line low and data lines (D0 through D5) low. The key scanning takes place so rapidly that several complete scans are made during even the shortest button-press.

The RESET button, when depressed, places a low level on the microprocessor reset line, initiating the CPU reset routine.

CPU BOARD

The CPU board contains the microprocessor, address decoding, data and address line buffers, main memory, and interrupt handling circuits. A cassette interface conditioning circuit and conditioning circuits for the NMI interrupt and Reset lines are also on the board.

U401 is a 6808 microprocessor. Circuitry within the processor, together with C416, C417, and Y401 make up a 3.58 MHz oscillator which provides the master timing signal for the system.

Address lines A8 thru A15 are buffered by U402 and applied to P408, which is provided for possible future expansion of the system.

A13 thru A15 along with VMA are applied to address decoder U403. The outputs from U403 on pins 15, 14, 13, 12, 11, 10, 9, and 7 represent decoded addresses to 8k memory blocks beginning with the 0 to 8k block on pin 15, and ending with the 56 to 64k block on pin 7.

Two of the 8k block decoded signals (from pins 10 and 11) are connected to P406 pins 1 and 2. These are the 32k to 40k and 40k to 48k blocks, respectively (addresses* $8000 to $9FFF and $A000 to $BFFF).

Two additional 8k decoded block signals from pins 12 and 13 are connected to P406 pins 1 and 2. They are the 16k to 24k and 24k to 32k blocks (addresses* $4000 to $5FFF and $6000 to $7FFF) respectively. These decoded 8k blocks, as well as buffered address lines A0 thru A7 connected to P406, are for possible future expansion of the system.

*NOTE: Memory addresses are given in hexadecimal, since that number system is used with the CPU. The dollar sign ($) before a number indicates that that number is a hexadecimal number.

Additional address decoding for the displays, keyboard, I/O ports, ROM and RAM is provided by U405, U406, U421, and U422. The memory map in the Appendix (fold-out Page 119) shows the memory addresses allocated to these various areas.

U416 is a 2k × 8 RAM. This RAM responds to addresses* from $0000 to $07FF. Addresses $0000 to $0038 are reserved for use by the monitor and should not be used for user programming.

U415 is an additional 2k × 8 user RAM. This RAM responds to addresses $0800 to $0FFF. Addresses $0EDF to $0FFF are reserved for use by the monitor and should not be used for your programs.

U418 is an 8k × 8 ROM which contains the system monitor. The standard system comes with jumpers J6, J2, and J4 connected so that U418 responds to addresses from $E000 to $FFFF.

U417 is an optional user ROM which can be either an 8k × 8, a 4k × 8, or a 2k × 8 device. With jumper J12 in place, the ROM at U417 will be enabled with all addresses from $2000 to $4000. For an 8k × device at U417, jumpers J8 and J10 should also be in place. A 4k × 8 ROM in U417 requires J8 and J9, while a 2k × 8 requires J7 and J9. See the Appendix (Page 118) for information about the jumper configurations and memory addresses for all of the different memory options.

U407 through U410 are each dual JK flip-flops. These devices make up an 8-bit interrupt status register which may be read by the microprocessor at input port $C200 (U412). This circuit operates as follows:

With no interrupts into the system, all eight of the flip-flops are cleared; each Q output (pins 8 and 13, of U407 through U410) is high. All eight Q outputs are connected to the input of U413. Since all of these inputs are high, the output of U413 is low. The low out of U413 is inverted by U423B and applied to the IRQ input of the microprocessor. A high level at this input represents a noninterrupt condition to the microprocessor.

Any one of eight different circuits can cause an interrupt to the CPU. Each of the eight circuits is connected to the clock input (pins 1 and 5) of one of the eight flip-flops. When an interrupt is generated, the circuit involved will cause one of the clock lines to go low.

The J inputs on pins 14 and 7 of each flip-flop are tied high. The K inputs on pins 3 and 10 are tied low. With this set of conditions existing, a negative-going transition on any clock input (pins 1 or 5) will cause the corresponding Q output to go high and the Q output to go low. A low on any of the Q outputs will cause one of the inputs to U413 to go low, resulting in a high output from U413. The high out of U413 is inverted by U423B and applied to the IRQ input to the microprocessor, where it initiates an interrupt sequence.

Since there is only one IRQ input on the microprocessor, a means must be provided to enable it to determine which of the eight possible interrupt conditions caused the interrupt. This is accomplished by U412.

The Q outputs (pins 12 and 13) of each of the interrupt status flip-flops (U407 through U410) are connected to one of the inputs of Port $C200 (U412). The first thing the Robot's executive program does when it responds to an interrupt is to read this port (at pins 26 through 33) to determine what has caused the interrupt. After it services the interrupt, the microprocessor resets the interrupt status flip-flop, thereby readying it to respond to the next interrupt. It does this in the following manner:

Pins 2 and 6 of the interrupt status flip-flops are the clear inputs. These pins are connected to each of the outputs of output port $C200 (U411). By writing a '1' to a bit of this port, the microprocessor can selectively clear any of the flip-flops.

U425D and its associated circuitry condition the input from a cassette player to make the data TTL compatible. The input from the cassette on pin 27 of P403 is applied across D401 and D402 where it is amplitude limited before being applied to the input of comparator U425D. The positive feedback provided by R404 adds to the high gain of the comparator to produce TTL level pulses with sharp leading and trailing edges at the output on pin 13. This output is fed to the I/O board where it is applied to bit 7 of input port $C260.

INPUT/OUTPUT CIRCUIT BOARD

The Input/Output (I/O) circuit board is the main interface between the Central Processing Unit (CPU) and the Robot motors and sensors. It receives CPU's output instructions at the output ports and relays those instructions to the various devices, and also relays the Input from the devices to the CPU. In addition, it contains the ultrasonic ranging timer circuits, the real time clock, and the sleep circuits.

Output Ports (U305 through U311)

U305 is typical of the output ports on the board. It contains eight D-type flip-flops which store the data from the CPU board, loaded from the buffered data bus (at pins 3, 4, 7, 8, 13, 14, 17, and 18). The stored data is then available on the output lines to its specific location (the experimental board, for U305).

Notice that the output ports are connected in parallel to the buffered data lines (P305 pins 62 through 69). The port is selected, and the data is accepted by it, in response to the signal pulse at the port's pin 11. This signal, generated by the address decoding circuitry on the CPU board, is applied only to the port that the CPU selects. U305 is selected when the CPU goes into the "write" mode, and selects address $C220 (at P305 pin 45).* The CPU memory address circuitry then sends the pulse to U305 pin 11. The signal on pin 11 goes low momentarily. The data on the buffered data bus is loaded into U305 on the rising edge of this select signal. In this manner, the CPU (under program control) outputs data to the output ports, and from there to the various motors and sensors.

Input Ports (U301 through U304, U314)

U303 is a typical input port. It contains eight three-state buffers which can supply the eight-bit data word from the sense board to the data bus. The port only does this when it receives an enable signal from the CPU. The enable signal is applied to pins 1 and 19 and (like the output port enable signal) comes from the address decoding circuitry on the CPU board. When $C240* appears on the address bus with the CPU in the "read" mode, the signal on pins 1 and 19 of U303 goes low, enabling the buffer (and essentially connecting the input from the sense board to the data bus). All the other input ports are in a high impedance state at this time. In this same manner, the CPU, under program control, can selectively enable any remaining input port to read the data present.

*NOTE: All memory addresses are given in hexadecimal, since that number system is used with the CPU. The dollar sign ($) before a number indicates that that number is a hexadecimal number.

Real Time Clock

The real time clock circuit is contained within U315. This IC uses a 32.768 kHz crystal (Y301) in a reference oscillator to keep track of real time. Counters within U315 count the seconds, minutes, hours, days, months, and years. These counters may be loaded (to set the time & date) or read (to read the time & date) over a four-bit data bus on pins 9, 10, 11, and 12. Input port U314 is used to read the four-bit clock data bus; output port U316 is used to write data from the buffered data lines to the clock. U316 also outputs bits to pins 18, 3, and 2 of the clock, which are HOLD (stops timekeeping), READ (sets up clock for read operation), and WRITE (sets up clock for write operation), respectively.

Pins 4, 5, 6, and 7 of the clock are address lines. The address comes from output port $C2C0 and selects the desired counter, which is connected to the data bus for reading or loading. U320 is connected to the clock input and output ports and helps decode read/write instructions from the CPU.

We use a sample of the 32 kHz clock reference signal from U315 pin 17, buffer it with U317F, and then use it as a timing reference in the ultrasonic ranging timer circuits.

Ultrasonic Ranging Timer

The timer measures the passage of time between the transmission of an ultrasonic pulse and the reception of an echo. This time period is a measure of the distance to the object that caused the echo.

This, basically, is how the timer works. A pulse comes into the I/O board on P311 pin 80 when the ultrasonic transmitter is turned on. This pulse starts twelve-bit binary counter U313 (counting the 32 kHz signal from the real time clock). Then U323B signals the CPU to read counter U313 when one of two things happens: either the echo is received at P311 pin 79, or the counter reaches its maximum. The following explains this circuitry in greater detail.

The first transmit pulse arrives on P311 pin 80. This passes through U317C to flip-flop U318 and enables NAND gate U319C with a high. U319C allows the 32 kHz from clock U315 and U317F to pass. U313 then begins to count. The count always begins from zero because U313 is reset (at pin 11) by the same transmit pulse, coming through U323A from U317C. Finally, the same transmit pulse resets U318, which is part of the output circuit to the CPU.

The pulse representing the first echo is received at P311 pin 79. It passes through U317A and B to U318 (pin 4) and stops NAND gate U319C from passing the 32 kHz signal to the counter. In addition, the signal sets U318. The high output from pin 12 goes through OR gate U323B and signals the CPU to read the value at the counter.

If the counter reaches its maximum before an echo is recognized by the receiver, the circuitry responds this way: Counter pin 14 (representing the highest bit) goes high. This signal sets the latch composed of U319A and D. This again passes through OR gate U323B to call for the CPU to read the value. This means that anything beyond the maximum range (about 8 feet) will give a reading as though it were exactly at the maximum range.

When the CPU is informed that a range reading is ready, it completes whatever step it is actually doing, and then interrogates the counter on port $C220. After reading the value, the CPU sends a "clear" signal to the counting/timing circuitry along P304 pin 33. Like the transmit pulse, this resets counter U313 and output flip-flop U318. All of this happens so fast that the max-ranged counter does not even have time to count one more beat.

Sleep Timer

The sleep timer permits the Robot to go into a "sleep" mode (where it consumes very little power) under program control. Each sleep period lasts 10 seconds, and during that time power is removed from all circuits except the RAM, the 10 second timer (U321), and some of the power supply circuits. Total current drain in the sleep mode is about 50 milliamps. The sleep mode function is accomplished by the action of dual one-shot U321 and associated circuitry.

A sleep command is initiated when the pin 16 output of U311 (output port $C2E0) goes high. This high triggers the A section of dual one-shot U321, causing its Q output (pin 7) to go low. The low level is inverted by Q301 and then fed to the power supply, where it shuts down the +5 volt supply. The time constant of R317 and C324 determines the 10 second period of the one-shot. When the circuit times out, the pin 7 output goes high to re-enable the +5 volt supply. Note that +5 volts for the components that don't "sleep" is provided from the +12 volts by U326 and U327.

Since the RAM power (V_{SB} from U326) is maintained during the sleep interval, transient voltages generated during the power-up cycle could possibly corrupt the data. To prevent this from happening, the read/write line to the RAM must be held high (in the 'read' state) during the entire sleep period and for approximately 100 milliseconds afterward. This gives the power supply time to stabilize before anything can be written into the RAM. This is accomplished by U322A and B, and the B section of dual one shot U321 (output at P304 pin 53).

The read/write line to the RAM passes through U322B. As long as the input on pin 5 is low, the output on pin 6 will follow the input. This is the normal condition. During the sleep interval, the Q output of U321 on pin 6 is high, resulting in a high on pin 8 of U322 and a high input on pin 5. This high holds the output at pin 6 high, which holds RAM in the read mode.

When the sleep period has transpired, the positive-going signal at the Q output of U321A on pin 7 triggers the B section, resulting in a high at the Q output of the B section (pin 10) — which has the same effect as already discussed.

Miscellaneous

Q305 through Q308 are the speech inflection drivers. Speech inflection requires greater drive voltage than is supplied from the CPU board. These transistors use the +12 volts from P311 pin 86 to provide sufficient voltage for the speech inflection.

U324A and U325 connect to VS1 at the drive wheel, and transfer the sensing of the wheel's rotation back to the CPU.

Q302 and U324B relay the low logic voltage signal to the CPU. Q303 and U324C relay the low drive voltage signal to the CPU. Q304 relays the motion detect signal to the CPU.

EXPERIMENTAL BOARD

The experimental board provides a convenient location for constructing your own circuits for use with the Robot. It also includes a means for interfacing the circuits to the Robot's microprocessor. It includes a 'breadboard' area in the center of the board, +5 volt and +12 volt supplies, an I/O port, a user-defined interrupt, and the read/write line from the CPU. Since this board has a port to the Robot's microprocessor, you can use it to add other senses, interface with a home computer, or control other devices from the Robot.

The Sleep switch and the Abort switch are also on the experimental board. When the Sleep switch is in the Sleep position, it places a low level on P1304 pin 10. When the Abort switch is pressed, it places a low level on P1302 pin 9, resulting in a NMI to the microprocessor, which will immediately suspend execution of the program.

CIRCUIT BOARD X-RAY VIEWS

NOTE: to find the PART NUMBER of a component for the purpose of ordering a replacement part:

A. Find the circuit component number (U1, C3, etc.) on the "Circuit Board X-Ray View."

B. Locate this same number in the "Circuit Component Number" column of the "Replacement Parts List."

C. Adjacent to the circuit component number, you will find the PART NUMBER and DESCRIPTION, which must be supplied when you order a replacement part.

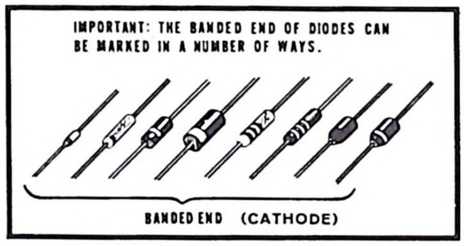
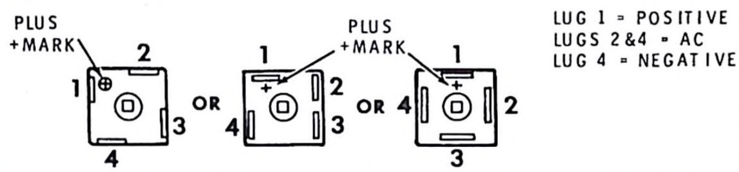
The actual X-ray views of the circuit boards are located on the corresponding schematic diagrams.

SEMICONDUCTOR IDENTIFICATION CHARTS

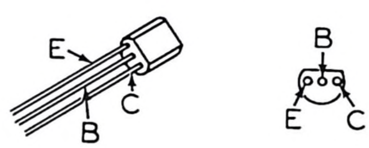
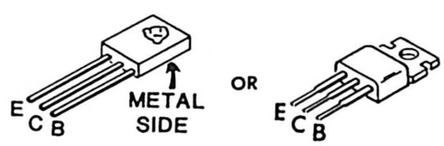
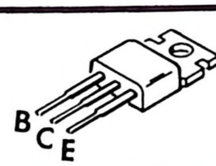
The following charts will help you identify the diodes, transistors, and integrated circuits used in the ET-18 Robot and charger. The circuit component numbers indicate where you will find the semiconductors:

CIRCUIT COMPONENT NUMBERS	ARE ON THE:
001-099	Robot chassis or in the Charger
101-199	Sonar transmit board
201-299	Power supply board
301-399	I/O board
401-499	CPU board
501-599	Speech board
601-699	Sense board
701-799	Arm drive board
801-899	Main drive board
901-999	Remote controller
1101-1199	Sonar receive board
1201-1299	Display board
1301-1399	Experimental board
1401-1499	Motion board
1501-1599	Wrist/shoulder board

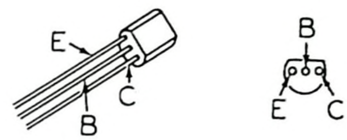
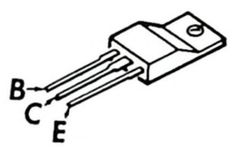
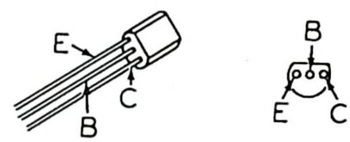
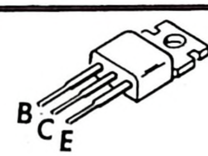
DIODES

CIRCUIT COMPONENT NUMBER	HEATH PART NUMBER	MAY BE REPLACED WITH	IDENTIFICATION
D1101	56-26	1N191	
D101, D301-D307, D401-D403, D501, D801, D901, D912, D1401	56-56	1N4149	
D601, D602	56-602		
D707–D711 D716–D719	56-605	1N4746A	
D202, D206	56-616	1N5232B	
D201, D205	57-42	3A1	
D203, D207, D701–D706, D802, D803, D804	57-65	1N4002	
D204	57-607	1N5817	
BR1	57-88	MDA990	

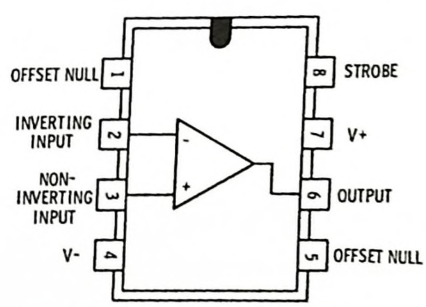
TRANSISTORS

CIRCUIT COMPONENT NUMBER	HEATH PART NUMBER	MAY BE REPLACED WITH	IDENTIFICATION
Q608, Q612	417-233	2N3643	
Q201, Q202, Q205, Q206, Q604, Q607, Q609, Q803, Q806	417-235	2N4121	
Q1201	417-263	SJE607	
Q204	417-289	2N6109	

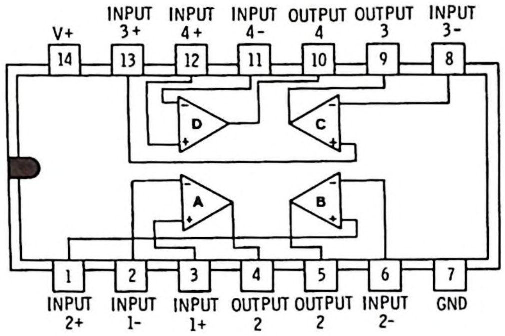
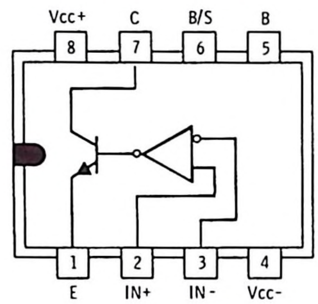
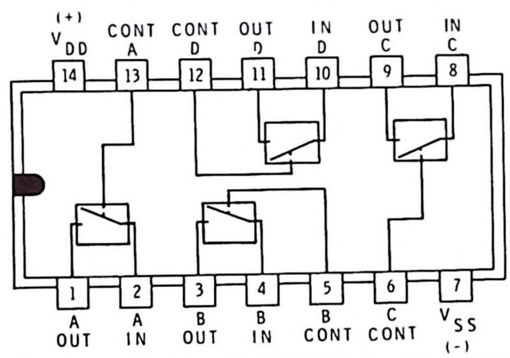
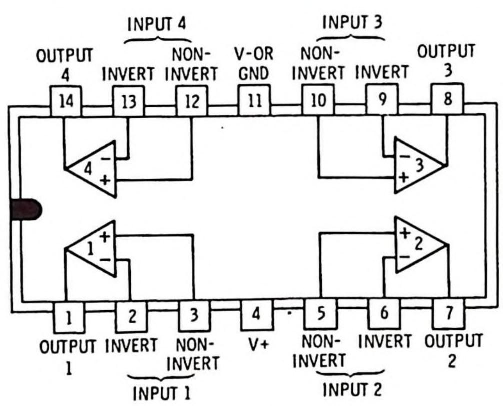
TRANSISTORS (Cont'd.)

CIRCUIT COMPONENT NUMBER	HEATH PART NUMBER	MAY BE REPLACED WITH	IDENTIFICATION
Q203	417-295	MPSL51	
Q102, Q301–Q304, Q502, Q602, Q603, Q605, Q606, Q611, Q701, Q704, Q1401, Q1402, Q1404	417-801	MPSA20	
Q802	417-881	MPSA13	
Q703, Q706, Q804	417-857	MJE976	
Q801, Q805	417-864	MPSA05	
Q101, Q501, Q601, Q702, Q705, Q1403	417-865	MPS-A55	
Q803, Q1202	417-881	MPSA13	
Q707–Q709, Q711–Q719, Q808, Q809, Q811	417-918	2N6387	

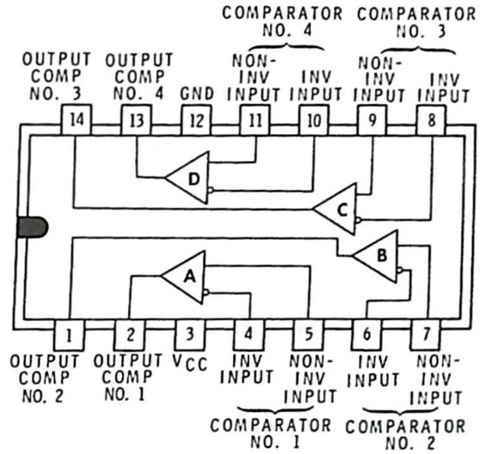
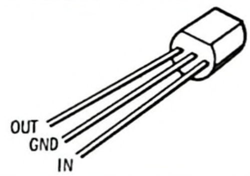
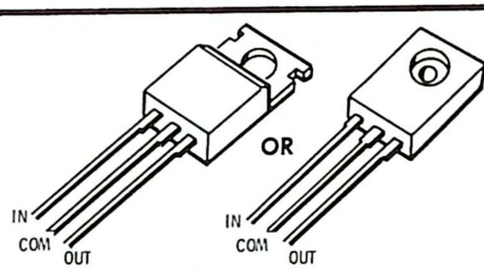
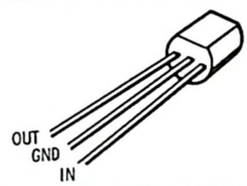
INTEGRATED CIRCUITS (IC'S)

CIRCUIT COMPONENT NUMBER	HEATH PART NUMBER	MAY BE REPLACED WITH	IDENTIFICATION
U1101, U1102	442-715	CA3130	

INTEGRATED CIRCUITS (IC'S) (Cont'd.)

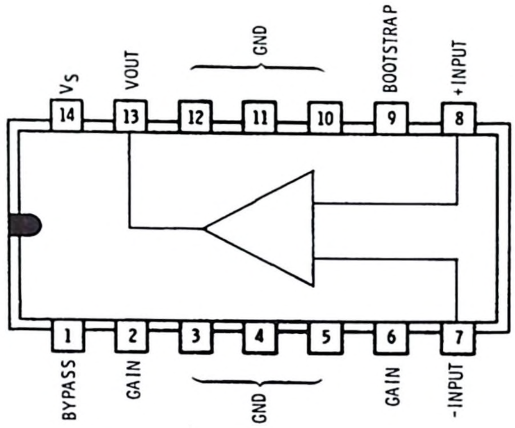
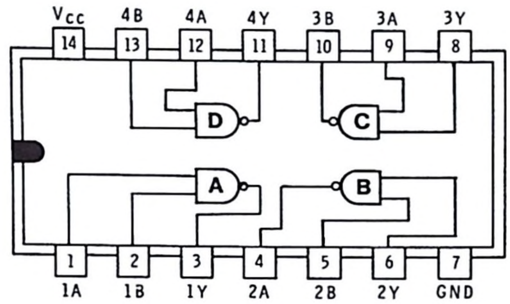
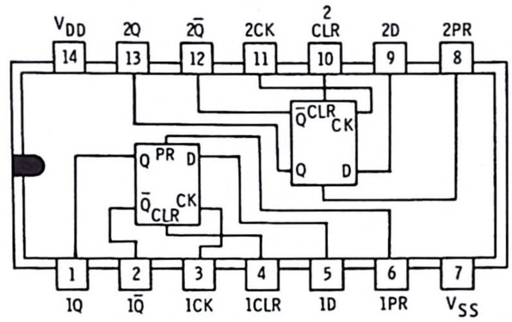
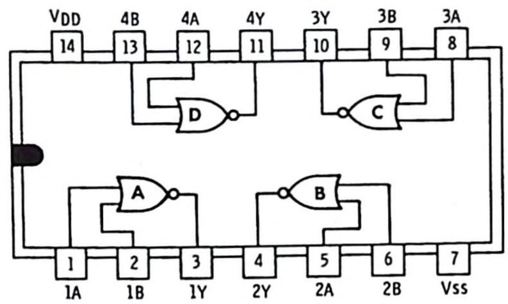
<table>
<tr><th>CIRCUIT COMPONENT NUMBER</th><th>HEATH PART NUMBER</th><th>MAY BE REPLACED WITH</th><th>IDENTIFICATION</th></tr>
<tr><td>U806</td><td>442-71</td><td>LM3900</td><td>
</td></tr>
<tr><td>U202, U205</td><td>442-75</td><td>LM311N</td><td>
</td></tr>
<tr><td>U801, U802</td><td>442-99</td><td>CD4016</td><td>
</td></tr>
<tr><td>U1403</td><td>442-602</td><td>LM324N</td><td>
</td></tr>
</table>

INTEGRATED CIRCUITS (IC'S) (Cont'd.)

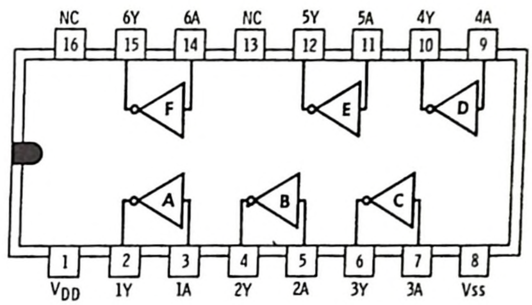
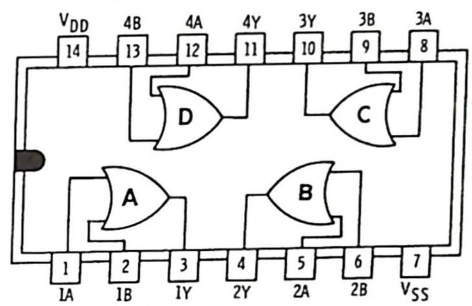
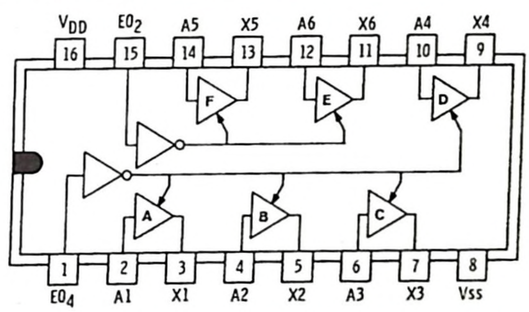
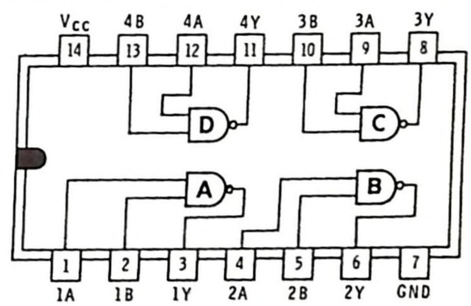
CIRCUIT COMPONENT NUMBER	HEATH PART NUMBER	MAY BE REPLACED WITH	IDENTIFICATION
U425	442-616	LM2901	
U326, U327	442-627	78L05	
U201, U204	442-674	UA7812	
U102, U1401	442-681	UA78L08	
U203	442-700	SG3524	

Heathkit®

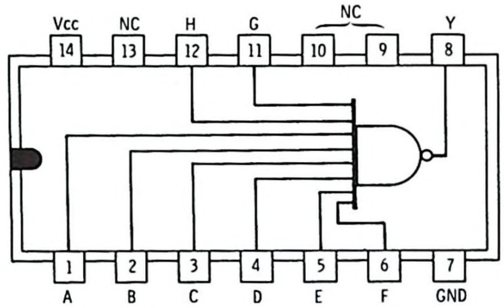
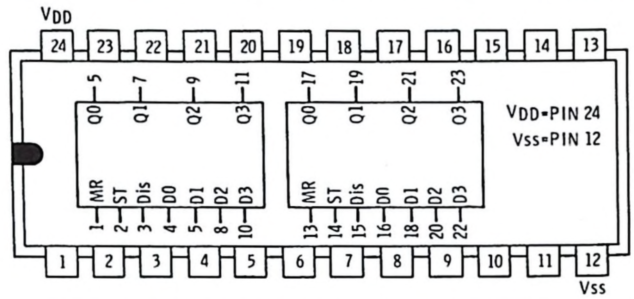
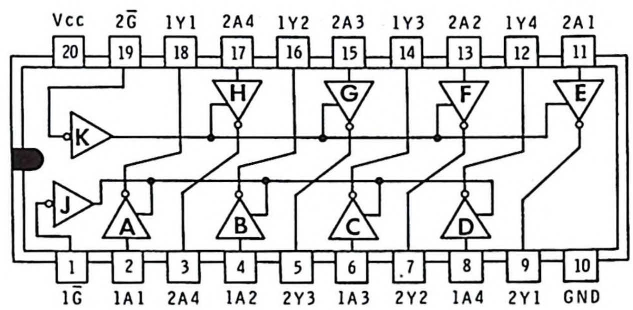
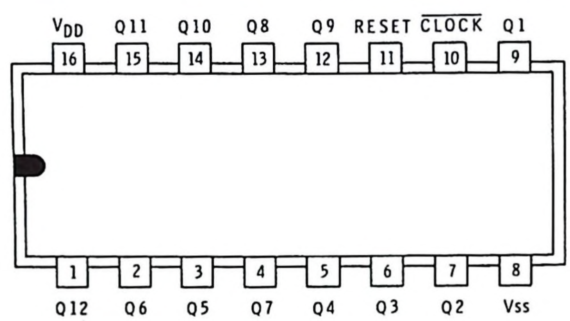
INTEGRATED CIRCUITS (IC'S) (Cont'd.)

<table>
<tr><th>CIRCUIT COMPONENT NUMBER</th><th>HEATH PART NUMBER</th><th>MAY BE REPLACED WITH</th><th>IDENTIFICATION</th></tr>
<tr><td>U502</td><td>442-762</td><td>LM388N-1</td><td rowspan="2">
</td></tr>
<tr><td>U803</td><td>442-740</td><td>LM556</td></tr>
<tr><td>U319, U901</td><td>443-603</td><td>CD4011AE</td><td>
</td></tr>
<tr><td>U104, U318</td><td>443-607</td><td>MC4013AL</td><td>
</td></tr>
<tr><td>U105</td><td>443-695</td><td>CD4001AE</td><td>
</td></tr>
</table>

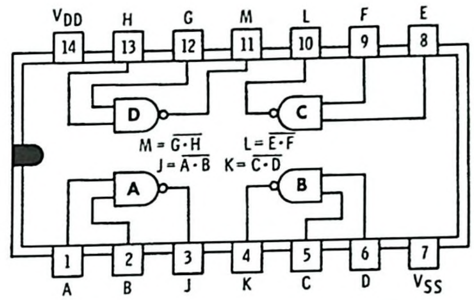
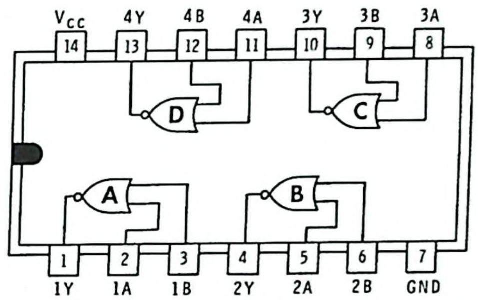
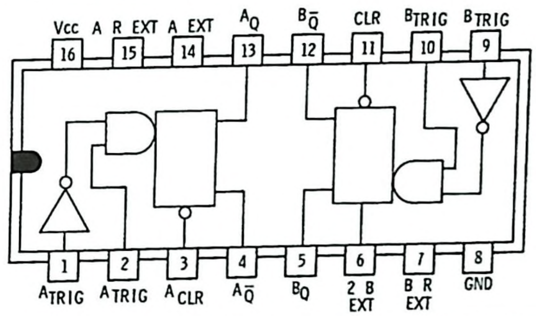
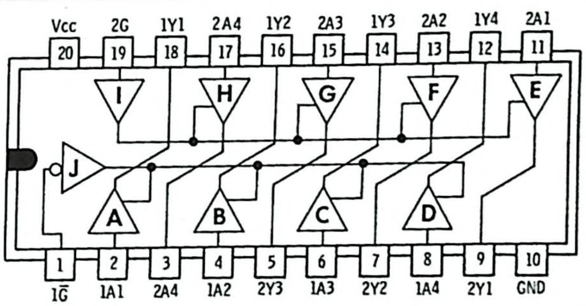
INTEGRATED CIRCUITS (IC'S) (Cont'd.)

CIRCUIT COMPONENT NUMBER	HEATH PART NUMBER	MAY BE REPLACED WITH	IDENTIFICATION
U317, U1402	443-701	MC4049CP	
U323	443-706	MC4071	
U314, U1207, U1209	443-720	80C97	
U423, U424	443-728	74LS00	

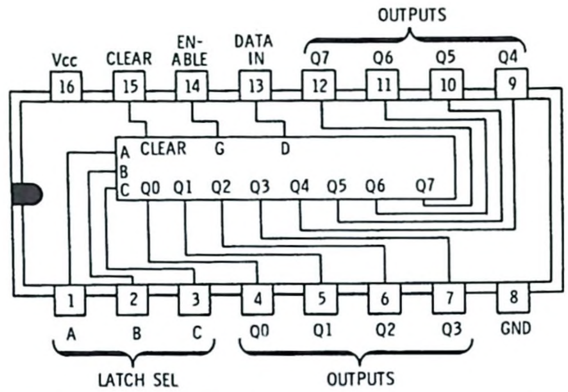
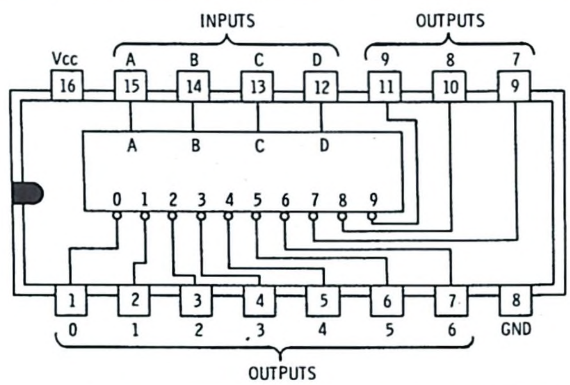
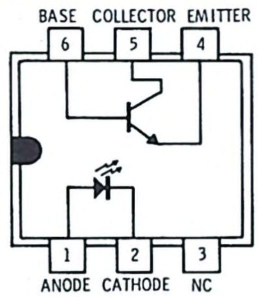
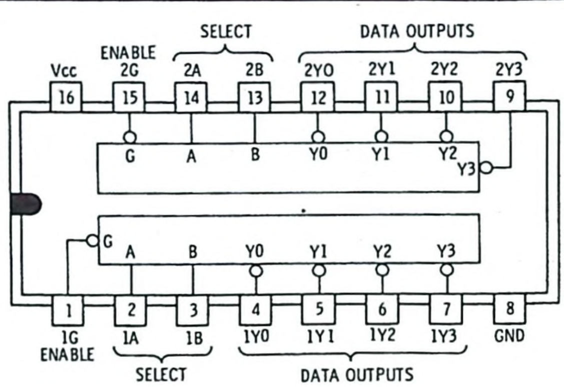
INTEGRATED CIRCUITS (IC'S) (Cont'd.)

CIRCUIT COMPONENT NUMBER	HEATH PART NUMBER	MAY BE REPLACED WITH	IDENTIFICATION
U413	443-732	74LS30	
U316	443-736	MC4508	
U601	443-754	74LS240	
U103, U313	443-760	MC4040	

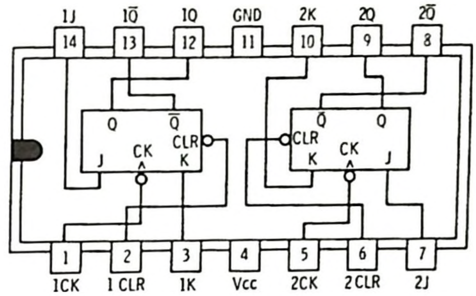
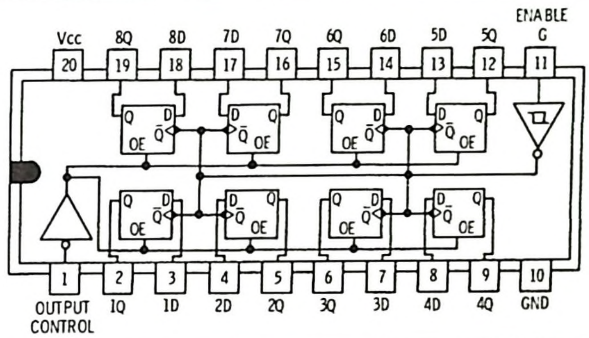
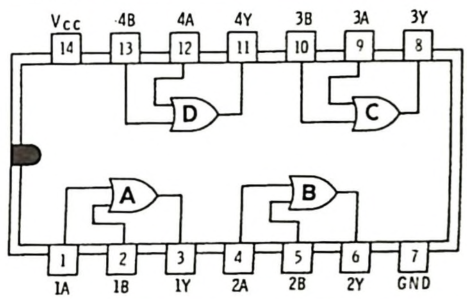
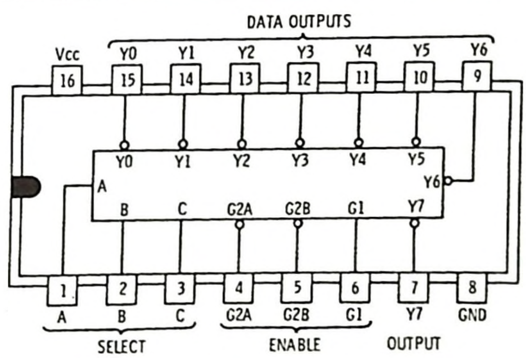
INTEGRATED CIRCUITS (IC'S) (Cont'd.)

<table>
<tr><th>CIRCUIT COMPONENT† NUMBER</th><th>HEATH PART NUMBER</th><th>MAY BE REPLACED WITH</th><th>IDENTIFICATION</th></tr>
<tr><td>U106, U324</td><td>443-778</td><td>MC4093</td><td>DUAL-IN-LINE PACKAGE

</td></tr>
<tr><td>U320</td><td>443-779</td><td>74LS02</td><td>
</td></tr>
<tr><td>U325, U1103</td><td>443-785</td><td>MM74C221</td><td>
</td></tr>
<tr><td>U301-U304, U312, U402, U404, U412, U419, U420</td><td>443-791</td><td>74LS244</td><td>
</td></tr>
</table>

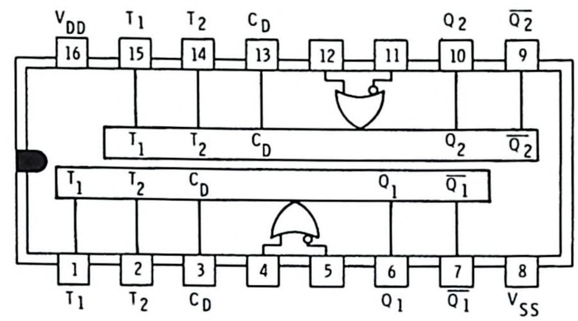
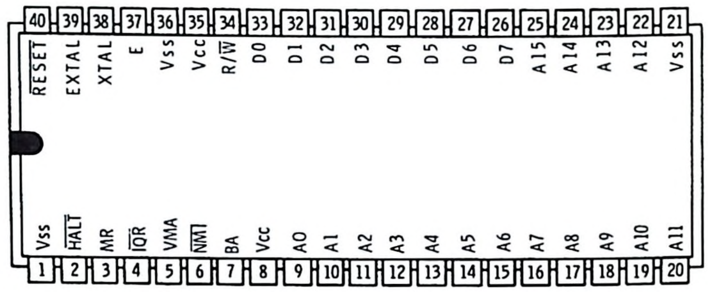
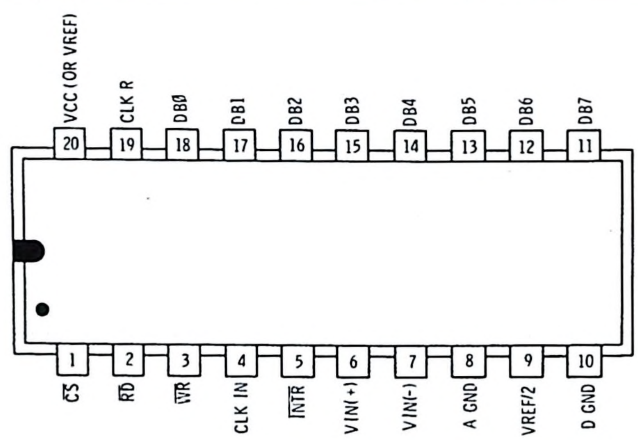
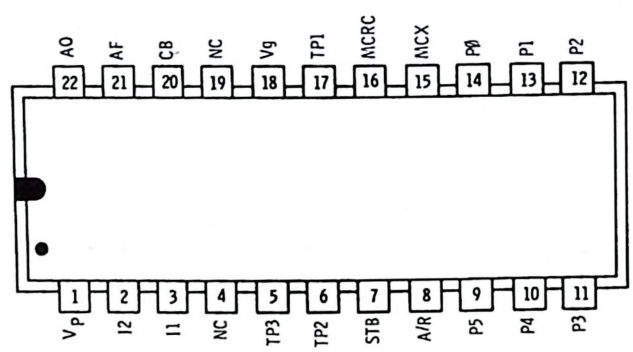
INTEGRATED CIRCUITS (IC'S) (Cont'd.)

CIRCUIT COMPONENT NUMBER	HEATH PART NUMBER	MAY BE REPLACED WITH	IDENTIFICATION
U1201–U1206	443-804	74LS259	
U1208	443-807	74LS42	
U101, U701–U709, U711–U715, U804–U811	443-808	4N26	
U405	443-822	74LS139	

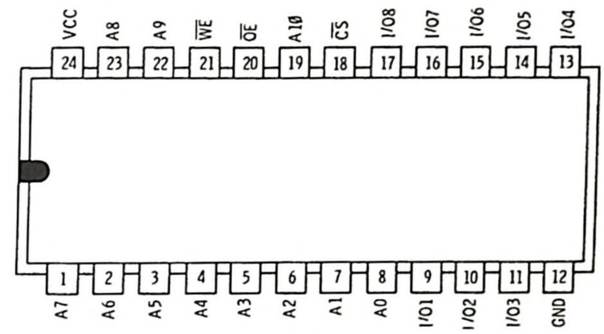
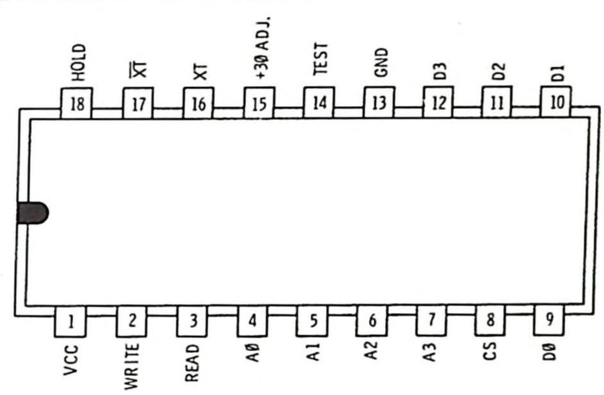
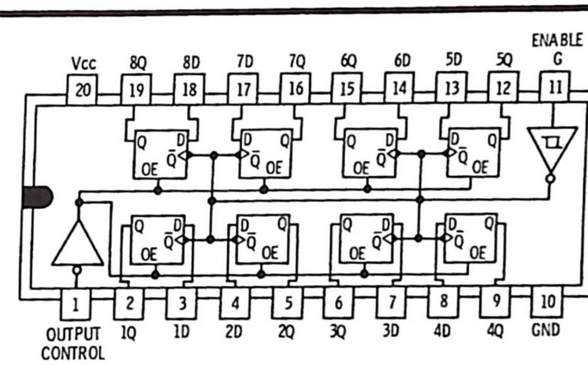
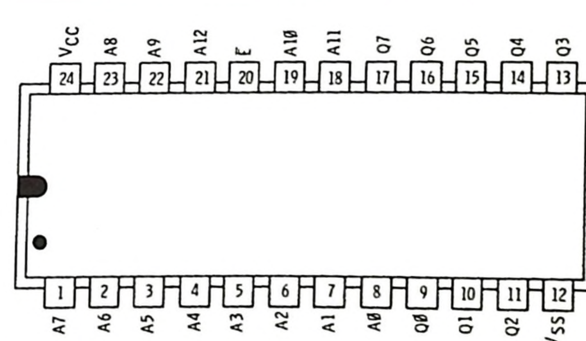
INTEGRATED CIRCUITS (IC'S) (Cont'd.)

CIRCUIT COMPONENT NUMBER	HEATH PART NUMBER	MAY BE REPLACED WITH	IDENTIFICATION
U407–U410	443-828	74LS73	
U305, U306, U308, U309, U411	443-863	74LS374	
U322, U414	443-875	74LS32	
U403, U406, U421, U422	443-877	74LS138	

INTEGRATED CIRCUITS (IC'S) (Cont'd.)

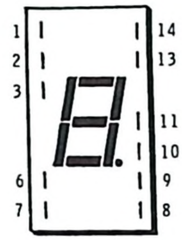
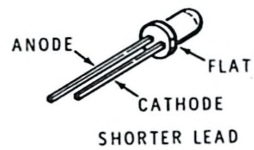
CIRCUIT COMPONENT NUMBER	HEATH PART NUMBER	MAY BE REPLACED WITH	IDENTIFICATION
U321	443-916	MC4538	
U401	443-939	MC6808	
U602	443-949	ADC0804	
U501	443-995	SC01	 NC=NO CONNECTION TPX=NO CONNECTION

INTEGRATED CIRCUITS (IC'S) (Cont'd.)

CIRCUIT COMPONENT NUMBER	HEATH PART NUMBER	MAY BE REPLACED WITH	IDENTIFICATION
U415	443-1016	6116	
U315	443-1055	MSM5832	
U307, U310, U311	443-1117	74LS374*	
U418	444-198-1	8k × 8 bit ROM. Available only from Heath Company.	

NOTE: 443-1117 is an integrated circuit which will only work if supplied by specific manufacturers. Do not use 443-863, even though it has the same generic number.

LEDS

CIRCUIT COMPONENT NUMBER	HEATH PART NUMBER	MAY BE REPLACED WITH	IDENTIFICATION
V1201–V1206	411-875	7-SEG LED	 PIN NO. 1. CATHODE A 2. CATHODE F 3. COMMON ANODE 4. NO PIN 5. NO PIN 6. NC 7. CATHODE E 8. CATHODE D 9. COMMON ANODE 10. CATHODE C 11. CATHODE G 12. NO PIN 13. CATHODE B 14. COMMON ANODE
V601–V608 V1401	412-640	RED LED	

REPLACEMENT PARTS LIST

To order a replacement part, use the Parts Order Form supplied. If a Parts Order Form is not available, refer to "Replacement Parts" inside the rear cover of this Manual.

SONAR TRANSMIT CIRCUIT BOARD

RESISTORS — CONTROL

NOTE: The following resistors are rated at 1/4-watt and have a tolerance of 5% unless otherwise noted.

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
R101	6-681-12	680 Ω resistor
R102	6-472-12	4700 Ω resistor
R103	6-472-12	4700 Ω resistor
R104	6-103-12	10 kΩ resistor
R105	6-681-12	680 Ω resistor
R106	6-100-12	10 Ω resistor
R107	6-562-12	5600 Ω resistor
R108	10-312	10 kΩ control
R109	6-103-12	10 kΩ resistor
R110	Not used	
R111	6-103-12	10 kΩ resistor
R112	6-684-12	680 kΩ resistor

CAPACITORS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
C101	25-859	.47 μF electrolytic
C102	25-859	.47 μF electrolytic
C103	21-761	.01 μF glass
C104	27-63	.022 μF Mylar*
C105	Not used	
C106	21-761	.01 μF glass
C107	21-22	220 pF ceramic
C108	21-22	220 pF ceramic
C109	Not used	
C110	Not used	
C111	21-761	.01 μF glass
C112	27-70	.0022 μF Mylar

DIODE — TRANSISTORS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
D101	56-56	1N4149 diode
Q101	417-865	MPSA55 transistor
Q102	417-801	MPSA20 transistor

INTEGRATED CIRCUITS (IC's)

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
U101	443-808	4N26 opto-isolator
U102	442-681	78L08 regulator
U103	443-760	14040
U104	443-607	14013
U105	443-695	4001
U106	443-778	14093

*DuPont Registered Trademark.

POWER SUPPLY CIRCUIT BOARD

RESISTORS — CONTROL

NOTE: The following resistors are rated at 1/4-watt and have a tolerance of 5% unless otherwise noted.

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
R201	6-152-12	1500 Ω resistor
R202	6-1500-12	150 Ω, 1% resistor
R203	Not used	
R204	Not used	
R205	6-6500-12	650 Ω, 1% resistor
R206	6-473-12	47 kΩ resistor
R207	6-8873-12	887 kΩ, 1% resistor
R208	6-9093-12	909 kΩ, 1% resistor
R209	Not used	
R210	Not used	
R211	Not used	
R212	6-472-12	4700 Ω resistor
R213	Not used	
R214	6-511-12	510 Ω resistor
R215	6-102-12	1000 Ω resistor
R216	6-152-12	1500 Ω resistor
R217	6-4751-12	4750 Ω, 1% resistor
R218	6-5491-12	5490 Ω, 1% resistor
R219	6-472-12	4700 Ω resistor
R220	Not used	
R221	6-4751-12	4750 Ω, 1% resistor
R222	6-4751-12	4750 Ω, 1% resistor
R223	6-303-12	30 kΩ resistor
R224	6-152-12	1500 Ω resistor
R225	6-1500-12	150 Ω, 1% resistor
R226	Not used	
R227	Not used	
R228	6-6500-12	650 Ω, 1% resistor
R229	6-473-12	47 kΩ resistor
R230	Not used	
R231	Not used	
R232	Not used	
R233	6-8873-12	887 kΩ, 1% resistor
R234	6-9093-12	909 kΩ, 1% resistor
R235	6-472-12	4700 Ω resistor
R236	6-681	680 Ω, 1/2-watt resistor
R237	6-681	680 Ω, 1/2-watt resistor

CAPACITORS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
C201	25-922	.68 µF electrolytic
C202	21-752	.1 µF ceramic
C203	21-752	.1 µF ceramic
C204	25-900	1 µF electrolytic
C205	25-880	10 µF vertical electrolytic
C206	21-752	.1 µF ceramic

CAPACITORS (Cont'd.)

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
C207	25-893	1000 µF electrolytic
C208	27-74	.01 µF Mylar
C209	Not used	
C210	Not used	
C211	25-864	10 µF axial lead electrolytic
C212	27-74	.01 µF Mylar
C213	27-74	.01 µF Mylar
C214	25-922	.68 µF electrolytic
C215	21-752	.1 µF ceramic
C216	21-752	.1 µF ceramic
C217	25-880	10 µF vertical electrolytic
C218	25-900	1 µF electrolytic

DIODES

D201	57-42	3A1
D202	56-669	1N5524D
D203	57-65	1N4002
D204	57-607	1N5817
D205	57-42	3A1
D206	56-669	1N5524D
D207	57-65	1N4002

TRANSISTORS

Q201	417-235	2N4121
Q202	417-235	2N4121
Q203	417-295	MPSL51
Q204	417-289	2N6109
Q205	417-235	2N4121
Q206	417-235	2N4121

INTEGRATED CIRCUITS (IC'S)

U202	442-75	311
U203	442-700	3524
U205	442-75	311

MISCELLANEOUS

F201	421-3	2-ampere 3AG fuse
F202	421-5	4-ampere 3AG fuse
L201	46-71	650 µH choke

Heathkit®

INPUT/OUTPUT

RESISTORS

NOTE: The following resistors are rated at 1/4-watt and have a tolerance of 5% unless otherwise noted.

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
R301	NOT USED	
R302	6-473-12	47 kΩ
R303	6-473-12	47 kΩ
R304	6-473-12	47 kΩ
R305	6-271-12	270 Ω
R306	6-104-12	100 kΩ
R307	6-104-12	100 kΩ
R308	6-272-12	2700 Ω
R309	6-471-12	470 Ω
R310	6-471-12	470 Ω
R311	6-272-12	2700 Ω
R312	6-471-12	470 Ω
R313	6-272-12	2700 Ω
R314	6-103-12	10 kΩ
R315	6-473-12	47 kΩ
R316	6-103-12	10 kΩ
R317	6-104-12	100 kΩ
R318	6-225-12	2.2 MΩ
R319	6-102-12	1000 Ω
R320	6-471-12	470 Ω
R321	6-272-12	2700 Ω
R322	6-222-12	2200 Ω
R323	6-104-12	100 kΩ
R324	6-103-12	10 kΩ
R325	6-102-12	1000 Ω
R326	6-223-12	22 kΩ
R327	6-104-12	100 kΩ
R328	6-104-12	100 kΩ
R329	6-104-12	100 kΩ
R330	Not used	
R331	6-683-12	68 kΩ
R332	6-223-12	22 kΩ
R333	6-104-12	100 kΩ
R334	6-104-12	100 kΩ
R335	6-683-12	68 kΩ
R336	6-104-12	100 kΩ
R337	6-472-12	4700 Ω
R338	6-472-12	4700 Ω
R339	6-472-12	4700 Ω
RP301	9-119	Resistor pack (8 × 10 kΩ)
RP302	9-119	Resistor pack (8 × 10 kΩ)
RP303	9-119	Resistor pack (8 × 10 kΩ)

CAPACITORS — TRIMMER

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
C301	21-769	.01 μF glass
C302	21-769	.01 μF glass
C303	21-769	.01 μF glass
C304	21-769	.01 μF glass
C305	21-769	.01 μF glass
C306	21-22	220 pF ceramic
C307	21-769	.01 μF glass
C308	21-22	220 pF ceramic
C309	21-22	220 pF ceramic
C310	Not used	
C311	21-769	.01 μF glass
C312	27-77	.1 μF Mylar
C313	21-99	.2 μF ceramic
C314	21-769	.01 μF glass
C315	21-3	10 pF ceramic
C316	31-57	2.7—20 pF trimmer
C317	21-769	.01 μF glass
C318	25-858	.33 μF electrolytic
C319	25-858	.33 μF electrolytic
C320	Not used	
C321	25-858	.33 μF electrolytic
C322	25-858	.33 μF electrolytic
C323	25-858	.33 μF electrolytic
C324	25-925	4.7 μF electrolytic
C325	27-85	.22 μF Mylar
C326	21-769	.01 μF glass
C327	21-769	.01 μF glass
C328	21-769	.01 μF glass
C329	21-769	.01 μF glass
C330	Not used	
C331	21-27	5000 pF (.005 μF) ceramic
C332	21-99	.2 μF ceramic
C333	21-99	.2 μF ceramic
C334	25-948	100 μF electrolytic

DIODES — TRANSISTORS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
D301	56-56	1N4149 diode
D302	56-56	1N4149 diode
D303	56-56	1N4149 diode
D304	56-56	1N4149 diode
D305	56-56	1N4149 diode

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION

DIODES — TRANSISTORS (Cont'd.)

D306	56-56	1N4149 diode
D307	56-56	1N4149 diode
D308	56-97	1N3017B diode
Q301	417-801	MPSA20 transistor
Q302	417-801	MPSA20 transistor
Q303	417-801	MPSA20 transistor
Q304	417-801	MPSA20 transistor
Q305	417-801	MPSA20 transistor
Q306	417-235	2N4121 transistor
Q307	417-801	MPSA20 transistor
Q308	417-235	2N4121 transistor

INTEGRATED CIRCUITS (IC's)

U301	443-791	74LS244
U302	443-791	74LS244
U303	443-791	74LS244
U304	443-791	74LS244
U305	443-863	74LS374
U306	443-863	74LS374
U307	443-863	74LS374
U308	443-863	74LS374

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION

INTEGRATED CIRCUITS (IC's) (Cont'd.)

U309	443-863	74LS374
U310	443-863	74LS374
U311	443-863	74LS374
U312	443-791	74LS244
U313	443-760	4040
U314	443-720	80C97
U315	443-1055	5832
U316	443-736	4508
U317	443-701	4049
U318	443-607	4013
U319	443-603	4011
U320	443-779	74LS02
U321	443-916	4538
U322	443-875	74LS32
U323	443-706	4071
U324	443-778	4093
U325	443-785	74C221

MISCELLANEOUS

Y301	404-624	32.768 kHz crystal

Heathkit®

CPU CIRCUIT BOARD

RESISTORS

NOTE: The following resistors are rated at 1/4-watt and have a tolerance of 5% unless otherwise noted.

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
R401	6-332-12	3300 Ω
R402	6-101-12	100 Ω
R403	6-103-12	10 kΩ
R404	6-104-12	100 kΩ
R405	6-102-12	1000 Ω
R406	6-102-12	1000 Ω
R407	6-103-12	10 kΩ
R408	6-332-12	3300 Ω
R409	6-104-12	100 kΩ
R410	Not used	
R411	6-332-12	3300 Ω
R412	6-332-12	3300 Ω
R413	6-332-12	3300 Ω
R414	6-332-12	3300 Ω
R415	6-332-12	3300 Ω
R416	6-104-12	100 kΩ
R417	6-223-12	22 kΩ
R418	6-332-12	3300 Ω
R419	6-332-12	3300 Ω
R420	Not used	
R421	6-332-12	3300 Ω
R422	6-104-12	100 kΩ
RP401	9-119	Resistor pack (8 × 10 kΩ

CAPACITORS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
C401	21-769	.01 μF glass
C402	21-769	.01 μF glass
C403	21-769	.01 μF glass
C404	21-769	.01 μF glass
C405	27-73	.047 μF Mylar
C406	21-769	.01 μF glass
C407	21-769	.01 μF glass
C408	21-769	.01 μF glass
C409	21-769	.01 μF glass
C410	Not used	
C411	21-769	.01 μF glass
C412	21-769	.01 μF glass
C413	21-769	.01 μF glass
C414	21-769	.01 μF glass
C415	21-769	.01 μF glass
C416	20-100	30 pF mica
C417	20-100	30 pF mica
C418	21-769	.01 μF glass
C419	21-769	.01 μF glass

CAPACITORS (Cont'd.)

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
C420	Not used	
C421	21-769	.01 μF glass
C422	21-769	.01 μF glass
C423	21-769	.01 μF glass
C424	21-769	.01 μF glass
C425	21-769	.01 μF glass
C426	21-769	.01 μF glass
C427	21-769	.01 μF glass
C428	25-930	2.2 μF axial lead electrolytic

DIODES

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
D401	56-56	1N4149
D402	56-56	1N4149
D403	56-56	1N4149

INTEGRATED CIRCUITS (IC's)

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
U401	443-939	6808 microprocessor
U402	443-791	74LS244
U403	443-877	74LS138
U404	443-791	74LS244
U405	443-822	74LS139
U406	443-877	74LS138
U407	443-828	74LS73
U408	443-828	74LS73
U409	443-828	74LS73
U410	443-828	74LS73
U411	443-863	74LS74
U412	443-791	74LS244
U413	443-732	74LS30
U414	443-875	47LS32
U415	443-1027	6116
U416	443-1027	6116
U417		ROM #2 (optional)
U418	444-198-1	ROM #1
U419	443-791	74LS244
U420	443-791	74LS244
U421	443-877	74LS138
U422	443-877	74LS138
U423	443-728	74LS00
U424	443-728	74LS00
U425	442-616	LM2901

MISCELLANEOUS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
Y401	404-238	3579.545 kHz crystal

SPEECH CIRCUIT BOARD
(Accessory)

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION

RESISTORS — CONTROLS

NOTE: The following resistors are rated at 1/4-watt and have a tolerance of 5% unless otherwise noted.

R501	6-272-12	2700 Ω resistor
R502	6-562-12	5600 Ω resistor
R503	10-390	20 kΩ control
R504	6-681-12	680 Ω resistor
R505	6-511-12	510 Ω resistor
R506	6-511-12	510 Ω resistor
R507	Not used	
R508	6-472-12	4700 Ω resistor
R509	6-103-12	10 kΩ resistor
R510	Not used	
R511	6-681-12	680 Ω resistor
R512	6-472-12	4700 Ω resistor
R513	6-229-12	2.2 Ω resistor
R514	6-222-12	2200 Ω resistor
R515	10-312	10 kΩ control

CAPACITORS

C501	20-115	300 pF mica
C502	27-85	.22 µF Mylar

CAPACITORS (Cont'd.)

C503	25-907	4.7 µF nonpolarized electrolytic
C504	21-75	100 pF ceramic
C505	25-924	2.2 µF electrolytic
C506	21-143	.05 µF ceramic
C507	25-905	470 µF electrolytic
C508	25-905	470 µF electrolytic
C509	25-905	470 µF electrolytic
C510	25-880	10 µF electrolytic

DIODE — TRANSISTORS — INTEGRATED CIRCUITS (IC's)

D501	56-56	1N4149 diode
Q501	417-865	MPSA55 transistor
Q502	417-801	MPSA20 transistor
U501	443-995	SC01 IC
U502	442-762	LM388N-1 IC

MISCELLANEOUS

A16	401-176	Speaker
L501	45-47	2 mH RF choke

SENSE CIRCUIT BOARD

RESISTORS — CONTROLS

NOTE: The following resistors are rated at 1/4-watt and have a tolerance of 5% unless otherwise noted.

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
R601	Not used	
R602	Not used	
R603	Not used	
R604	6-472-12	4700 Ω resistor
R605	6-472-12	4700 Ω resistor
R606	6-472-12	4700 Ω resistor
R607	6-681-12	680 Ω resistor
R608	6-681-12	680 Ω resistor
R609	6-681-12	680 Ω resistor
R610	Not used	
R611	6-681-12	680 Ω resistor
R612	6-681-12	680 Ω resistor
R613	6-681-12	680 Ω resistor
R614	6-681-12	680 Ω resistor
R615	6-681-12	680 Ω resistor
R616	6-681-12	680 Ω resistor
R617	6-472-12	4700 Ω resistor
R618	6-100-12	10 Ω resistor
R619	6-472-12	4700 Ω resistor
R620	Not used	
R621	6-564-12	560 kΩ resistor
R623	6-472-12	4700 Ω resistor
R624	6-472-12	4700 Ω resistor
R625	6-470-12	47 Ω resistor
R626	6-472-12	4700 Ω resistor
R627	6-472-12	4700 Ω resistor
R628	6-472-12	4700 Ω resistor
R629	6-471-12	47 Ω resistor
R630	Not used	
R631	6-225-12	2.2 MΩ resistor
R632	6-472-12	4700 Ω resistor
R633	10-928	1 MΩ control
R634	6-470-12	47 Ω resistor
R635	6-470-12	47 Ω resistor
R636	6-103-12	10 kΩ resistor
R637	6-102-12	1000 Ω resistor
R638	6-472-12	4700 Ω resistor
R639	6-472-12	4700 Ω resistor
R640	Not used	
R641	6-102-12	1000 Ω resistor
R642	6-102-12	1000 Ω resistor
R643	6-472-12	4700 Ω resistor
R644	6-472-12	4700 Ω resistor
R645	6-104-12	100 kΩ resistor
R646	10-1049	2 MΩ control
R647	6-103-12	10 kΩ resistor
R648	6-470-12	47 Ω resistor
R649	6-102-12	1000 Ω resistor

CAPACITORS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
C601	21-762	.1 µF glass
C602	21-762	.1 µF glass
C603	25-212	22 µF tantalum
C604	21-762	.1 µF glass
C605	21-762	.1 µF glass
C606	25-918	100 µF electrolytic
C607	27-138	.033 µF Mylar
C608	21-722	330 pF ceramic
C609	21-140	.001 µF ceramic
C610	Not used	
C611	27-138	.033 µF Mylar
C612	25-918	100 µF electrolytic
C613	21-722	330 pF ceramic
C614	21-762	.1 µF glass
C615	25-918	100 µF electrolytic
C616	25-864	10 µF electrolytic
C617	25-918	100 µF electrolytic

TRANSISTORS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
Q601	417-865	MPSA55
Q602	417-801	MPSA20
Q603	417-801	MPSA20
Q604	417-235	2N4121
Q605	417-801	MPSA20
Q606	417-801	MPSA20
Q607	417-235	2N4121
Q608	417-233	2N3643
Q609	417-235	2N4121
Q610	Not used	
Q611	417-801	MPSA20
Q612	417-233	2N3643

INTEGRATED CIRCUITS (IC's)

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
U601	443-754	74LS240
U602	443-949	ADC0804

MISCELLANEOUS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
V601	412-611	Red LED
V602	412-611	Red LED
V603	412-611	Red LED
V604	412-611	Red LED
V605	412-611	Red LED
V606	412-611	Red LED
V607	412-611	Red LED
V608	412-611	Red LED
D601	56-602	Germanium diode
D602	56-602	Germanium diode

ARM DRIVE CIRCUIT BOARD

RESISTORS

NOTE: The following resistors are rated at 1/4-watt and have a tolerance of 5% unless otherwise noted.

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
R701	6-271-12	270 Ω
R702	6-222-12	2200 Ω
R703	6-103-12	10 kΩ
R704	6-471	470 Ω, 1/2-watt
R705	6-472-12	4700 Ω
R706	6-472-12	4700 Ω
R707	6-271-12	270 Ω
R708	6-222-12	2200 Ω
R709	6-103-12	10 kΩ
R710	Not used	
R711	6-471	470 Ω, 1/2-watt
R712	6-472-12	4700 Ω
R713	6-472-12	4700 Ω
R714	6-271-12	270 Ω
R715	6-271-12	270 Ω
R716	6-472-12	4700 Ω
R717	6-271-12	270 Ω
R718	6-271-12	270 Ω
R719	6-472-12	4700 Ω
R720	Not used	
R721	6-271-12	270 Ω
R722	6-271-12	270 Ω
R723	6-472-12	4700 Ω
R724	6-271-12	270 Ω
R725	6-271-12	270 Ω
R726	6-472-12	4700 Ω
R727	6-271-12	270 Ω
R728	6-271-12	270 Ω
R729	6-472-12	4700 Ω
R730	Not used	
R731	6-271-12	270 Ω
R732	6-271-12	270 Ω
R733	6-472-12	4700 Ω
R734	6-271-12	270 Ω
R735	6-271-12	270 Ω
R736	6-472-12	4700 Ω
R737	6-271-12	270 Ω
R738	6-271-12	270 Ω
R739	6-472-12	4700 Ω
R740	Not used	
R741	6-271-12	270 Ω
R742	6-271-12	270 Ω
R743	6-472-12	4700 Ω
R744	6-271-12	270 Ω
R745	6-271-12	270 Ω
R746	6-472-12	4700 Ω
R747	6-271-12	270 Ω
R748	6-271-12	270 Ω
R749	6-472-12	4700 Ω
R750	Not used	

RESISTORS (Cont'd.)

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
R751	6-271-12	270 Ω
R752	6-271-12	270 Ω
R753	6-472-12	4700 Ω

DIODES

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
D701-D706	57-65	1N4002
D707-D711	56-605	1N4746A
D716-D719	56-605	1N4746A

TRANSISTORS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
Q701	417-801	MPSA20
Q702	417-865	MPSA55
Q703	Not used	
Q704	417-801	MPSA20
Q705	417-865	MPSA55
Q706	Not Used	
Q707	417-918	2N6387
Q708	417-918	2N6387
Q709	417-918	2N6387
Q710	Not Used	
Q711	417-918	2N6387
Q712	417-918	2N6387
Q713	417-918	2N6387
Q714	417-918	2N6387
Q715	417-918	2N6387
Q716	417-918	2N6387
Q717	417-918	2N6387
Q718	417-918	2N6387
Q719	417-918	2N6387

INTEGRATED CIRCUITS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
U701	443-808	4N26
U702	443-808	4N26
U703	443-808	4N26
U704	443-808	4N26
U705	443-808	4N26
U706	443-808	4N26
U707	443-808	4N26
U708	443-808	4N26
U709	443-808	4N26
U710	Not used	
U711	443-808	4N26
U712	443-808	4N26
U713	443-808	4N26
U714	443-808	4N26
U715	443-808	4N26

MAIN DRIVE CIRCUIT BOARD

RESISTORS

NOTE: The following resistors are rated at 1/4-watt and have a tolerance of 5% unless otherwise noted.

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
R801	6-2001-12	2000 Ω, 1%
R802	6-4001-12	4000 Ω, 1%
R803	6-8001-12	8000 Ω, 1%
R804	6-1602-12	16 kΩ, 1%
R805	6-3202-12	32 kΩ, 1%
R806	6-6402-12	64 kΩ, 1%
R807	6-1503-12	150 kΩ, 1%
R808	6-102-12	1000 Ω
R809	6-103-12	10 kΩ
R810	not used	
R811	6-5002-12	50 kΩ, 1%
R812	6-103-12	10 kΩ
R813	6-221-12	220 Ω
R814	6-102-12	1000 Ω
R815	6-471-12	470 Ω
R816	6-223-12	22 kΩ
R817	6-222-12	2200 Ω
R818	6-223-12	22 kΩ
R819	6-101-2	100 Ω, 2-watt, 5%
R820	not used	
R821	6-222-12	2200 Ω
R822	6-103-12	10 kΩ
R823	6-124-12	120 kΩ
R824	6-104-12	100 kΩ
R825	10-390	20 kΩ control
R826	6-104-12	100 kΩ
R827	6-104-12	100 kΩ
R828	6-682-12	6800 Ω
R829	6-222-12	2200 Ω
R830	not used	
R831	6-471-12	470 Ω
R832	6-223-12	22 kΩ
R833	6-471-12	470 Ω
R834	6-223-12	22 kΩ
R835	6-471-12	470 Ω
R836	6-223-12	22 kΩ
R837	6-471-12	470 Ω
R838	6-223-12	22 kΩ
R839	6-102-12	1000 Ω
R840	not used	
R841	6-102-12	1000 Ω
R842	6-102-12	1000 Ω
R843	6-102-12	1000 Ω

CAPACITORS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
C801	27-77	.1 μF Mylar
C802	21-176	.01 μF ceramic
C803	27-77	.1 μF Mylar
C804	21-176	.01 μF ceramic
C805	21-43	.001 μF ceramic
C806	25-880	10 μF electrolytic

DIODES

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
D801	56-56	1N4149
D802	57-65	1N4002
D803	57-65	1N4002
D804	57-65	1N4002

TRANSISTORS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
Q801	417-864	MPSA05
Q802	417-881	MPSA13
Q803	417-235	2N4121
Q804	417-857	MJE5976
Q805	417-864	MPSA05
Q806	417-235	2N4121
Q807	417-918	2N6387
Q808	417-918	2N6387
Q809	417-918	2N6387
Q810	not used	
Q811	417-918	2N6387

INTEGRATED CIRCUITS (IC's)

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
U801	442-99	CD4016
U802	442-99	CD4016
U803	442-740	LM556 or MC1456
U804	443-808	4N26
U805	443-808	4N26
U806	442-71	LM3900
U807	443-808	4N26
U808	443-808	4N26
U809	443-808	4N26
U810	not used	
U811	443-808	4N26

RELAY

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
K801	69-106	DPDT

REMOTE CONTROL UNIT

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION

CIRCUIT BOARD PARTS

Diodes

D901	56-56	1N4149
D902	56-56	1N4149
D903	56-56	1N4149
D904	56-56	1N4149
D905	56-56	1N4149
D906	56-56	1N4149
D907	56-56	1N4149
D908	56-56	1N4149
D909	56-56	1N4149
D910	Not used	
D911	56-56	1N4149
D912	56-56	1N4149

RESISTORS

NOTE: The following resistors are rated at 1/4-watt and have a tolerance of 5% unless otherwise noted.

R901	6-332-12	3300 Ω
R902	6-332-12	3300 Ω

RESISTORS (Cont'd.)

R903	6-332-12	3300 Ω
R904	6-332-12	3300 Ω
R905	6-332-12	3300 Ω
R906	6-332-12	3300 Ω

Miscellaneous

SW901	64-882	Pushbutton switch
U901	443-603	4011 integrated circuit

CASE PARTS

SW902	60-91	Slide switch
SW903	61-44	Rocker switch
SW904	63-1392	Rotary switch
	95-655	Case, right side
	95-656	Case, left side
	390-2298	Panel label
	462-1134	Trigger
	462-1135	Knob

SONAR RECEIVE CIRCUIT BOARD

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION

RESISTORS

NOTE: The following resistors are rated at 1/4-watt and have a tolerance of 5% unless otherwise noted.

R1101	6-103-12	10 kΩ
R1102	6-102-12	1000 kΩ
R1103	6-104-12	100 kΩ
R1104	6-103-12	10 kΩ
R1105	6-102-12	1000 kΩ
R1106	6-103-12	10 kΩ
R1107	6-104-12	100 kΩ
R1108	6-103-12	10 kΩ
R1109	6-104-12	100 kΩ
R1110	Not used	
R1111	6-470-12	47 Ω
R1112	6-333-12	33 kΩ
R1113	6-473-12	47 kΩ

CAPACITORS

C1101	21-176	.01 µF ceramic

CAPACITORS (Cont'd.)

C1102	21-75	100 pF ceramic
C1103	25-880	10 µF electrolytic
C1104	21-176	.01 µF ceramic
C1105	25-880	10 µF electrolytic
C1106	27-73	.047 µF Mylar
C1107	21-75	100 pF ceramic
C1108	21-176	.01 µF ceramic
C1109	25-883	47 µF electrolytic
C1110	Not used	
C1111	20-106	390 pF mica
C1112	25-912	3.3 µF electrolytic
C1113	25-883	47 µF electrolytic

MISCELLANEOUS

L1101	45-47	2 mH RF choke
U1101	442-715	CA3130 integrated circuit
U1102	442-715	CA3130 integrated circuit
U1103	443-785	74C221 integrated circuit
D1101	56-26	1N191 diode

Heathkit®

DISPLAY CIRCUIT BOARD

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION

RESISTORS

NOTE: The following resistors are rated at 1/4-watt and have a tolerance of 5% unless otherwise noted.

R1201	6-682-12	6800 Ω
R1202	Not used	
R1203	6-103-12	10 kΩ
R1204	6-222-12	2200 Ω
R1205	6-470	47 Ω, 1/2-watt
R1206	6-471-12	470 Ω
R1207	6-471-12	470 Ω
R1208	6-471-12	470 Ω
R1209	6-471-12	470 Ω
R1210	Not used	
R1211	6-471-12	470 Ω
R1212	6-471-12	470 Ω
R1213	6-471-12	470 Ω
R1214	6-471-12	470 Ω
R1215	6-471-12	470 Ω
R1216	6-471-12	470 Ω
R1217	6-471-12	470 Ω
R1218	6-471-12	470 Ω
R1219	6-471-12	470 Ω
R1220	Not used	
R1221	6-471-12	470 Ω
R1222	6-471-12	470 Ω
R1223	6-471-12	470 Ω
R1224	6-471-12	470 Ω
R1225	6-471-12	470 Ω
R1226	6-471-12	470 Ω
R1227	6-471-12	470 Ω
R1228	6-471-12	470 Ω
R1229	6-471-12	470 Ω
R1230	Not used	
R1231	6-471-12	470 Ω
R1232	6-471-12	470 Ω
R1233	6-471-12	470 Ω
R1234	6-471-12	470 Ω
R1235	6-471-12	470 Ω
R1236	6-471-12	470 Ω
R1237	6-471-12	470 Ω
R1238	6-471-12	470 Ω
R1239	6-471-12	470 Ω
R1240	Not used	
R1241	6-471-12	470 Ω
R1242	6-471-12	470 Ω
R1242	6-471-12	470 Ω
R1243	6-471-12	470 Ω
R1244	6-471-12	470 Ω
R1245	6-471-12	470 Ω
R1246	6-471-12	470 Ω
R1247	6-471-12	470 Ω
R1248	6-471-12	470 Ω
R1249	6-471-12	470 Ω
R1250	Not used	
R1251	6-471-12	470 Ω
R1252	6-471-12	470 Ω

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION

RESISTORS (Cont'd.)

R1253	6-471-12	470 Ω
R1254	6-471-12	470 Ω
R1255	6-471-12	470 Ω
R1256	6-471-12	470 Ω
R1257	6-471-12	470 Ω
R1258	6-471-12	470 Ω
R1259	6-472-12	4700 Ω
R1260	Not used	
R1261	6-822-12	8200 Ω
R1262	6-822-12	8200 Ω
R1263	6-822-12	8200 Ω
R1264	6-822-12	8200 Ω
R1265	6-822-12	8200 Ω
R1266	6-822-12	8200 Ω

CAPACITORS

C1201	25-920	68 μF electrolytic
C1202	Not used	
C1203	25-880	10 μF electrolytic
C1204	21-761	.01 μF glass
C1205	21-761	.01 μF glass
C1206	21-761	.01 μF glass
C1207	21-761	.01 μF glass
C1208	21-761	.01 μF glass
C1209	21-761	.01 μF glass
C1210	Not used	
C1211	21-761	.01 μF glass
C1212	21-761	.01 μF glass
C1213	21-761	.01 μF glass

TRANSISTORS

Q1201	417-263	SJE607
Q1202	417-881	MPSA13

INTEGRATED CIRCUITS (IC's)

U1201	443-804	74LS259
U1202	443-804	74LS259
U1203	443-804	74LS259
U1204	443-804	74LS259
U1205	443-804	74LS259
U1206	443-804	74LS259
U1207	443-720	80C97
U1208	443-807	74LS42
U1209	443-720	80C97

MISCELLANEOUS

	64-839	Pushbutton switch
V1201	411-875	7-segment LED
V1202	411-875	7-segment LED
V1203	411-875	7-segment LED
V1204	411-875	7-segment LED
V1205	411-875	7-segment LED
V1206	411-875	7-segment LED

EXPERIMENTAL CIRCUIT BOARD

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
SW1301	64-901	Pushbutton switch
SW1302	60-623	Slide switch
	432-880	Large connector block with 128 metal inserts
	432-973	4-pin connector block

MOTION CIRCUIT BOARD

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION

RESISTORS — CONTROL

NOTE: The following resistors are rated at 1/4-watt and have a tolerance of 5% unless otherwise noted.

R1401	6-333-12	33 kΩ resistor
R1402	6-683-12	68 kΩ resistor
R1403	6-104-12	100 kΩ resistor
R1404	6-471-12	470 Ω resistor
R1405	6-104-12	100 kΩ resistor
R1406	6-103-12	10 kΩ resistor
R1407	6-104-12	100 kΩ resistor
R1408	6-223-12	22 kΩ resistor
R1409	10-326	500 kΩ control
R1410	6-682-12	6800 Ω resistor
R1411	6-275-12	2.7 MΩ resistor
R1412	6-103-12	10 kΩ resistor
R1413	6-105-12	1 MΩ resistor
R1414	6-473-12	47 kΩ resistor
R1415	6-224-12	220 kΩ resistor
R1416	6-104-12	100 kΩ resistor
R1417	6-563-12	56 kΩ resistor
R1418	6-104-12	100 kΩ resistor
R1419	6-102-12	1000 Ω resistor
R1420	6-100-12	10 Ω resistor
R1421	6-103-12	10 kΩ resistor
R1422	6-102-12	1000 Ω resistor
R1423	6-272-12	2700 Ω resistor
R1424	6-474-12	470 kΩ resistor

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION

CAPACITORS

C1401	21-163	.001 µF ceramic
C1402	21-176	.01 µF ceramic
C1403	21-722	330 pF ceramic
C1404	25-859	.47 µF electrolytic
C1405	27-77	.1 µF Mylar
C1406	25-880	10 µF electrolytic
C1407	27-77	.1 µF Mylar
C1408	27-42	.0068 µF Mylar
C1409	27-77	.1 µF Mylar
C1410	25-900	1 µF electrolytic
C1411	27-77	.1 µF Mylar
C1412	25-880	10 µF electrolytic
C1413	21-176	.01 µF ceramic

TRANSISTORS

Q1401	417-801	MPSA20
Q1402	417-801	MPSA20
Q1403	417-865	MPSA55
Q1404	417-801	MPSA20

INTEGRATED CIRCUITS (IC's)

U1401	442-681	78L08
U1402	443-701	4049
U1403	442-602	LM324

MISCELLANEOUS

D1401	56-56	1N4149 diode
V1401	412-640	Light-emitting diode (LED)

WRIST/SHOULDER CIRCUIT BOARD

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
SHOULDER CIRCUIT BOARD		
D1501	57-65	1N4002 diode
D1502	57-65	1N4002 diode
D1503	57-65	1N4002 diode
D1504	57-65	1N4002 diode
D1505	57-65	1N4002 diode
D1506	57-65	1N4002 diode
D1507	57-65	1N4002 diode

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
WRIST CIRCUIT BOARD		
D1501	57-65	1N4002 diode
D1502	57-65	1N4002 diode
D1503	57-65	1N4002 diode
D1504	57-65	1N4002 diode
D1505	57-65	1N4002 diode
D1506	57-65	1N4002 diode
D1507	57-65	1N4002 diode
D1508	57-65	1N4002 diode
D1509	57-65	1N4002 diode
D1510	Not used	
D1511	57-65	1N4002 diode
D1512	57-65	1N4002 diode
D1513	57-65	1N4002 diode

ARM PARTS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
PLASTIC — NYLON — RUBBER PARTS		
	92-757	Circuit board cover
	204-2595	Gripper bracket
	204-2596	Wrist bracket
	214-233	Bearing housing
	214-234	Motor housing top
	214-235	Motor housing bottom
	214-236	Wrist rotate motor cover
	485-40	Shoulder plug
	92-756	Arm shell (curved track)
	92-782	Arm shell (straight track)
	266-1197	Gripper
BRACKETS — PLATE		
	205-1879-1	Cover plate
	267-18-1	Motor mounting bracket
	267-19-1	Wrist mounting bracket

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
MOTORS		
A4	420-642	Shoulder
A5	420-631	Arm extend
A6	420-624	Gripper
A7	420-625	Wrist pivot
A8	420-626	Wrist rotate
MISCELLANEOUS		
	347-87	10-wire spiral cable
	390-2319	Decorative label (shoulder)
	390-2324	Round decorative label (wrist motor)

CHARGER

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
C1	25-910	3300 μF electrolytic capacitor
BR1	57-88	MDA990 bridge rectifier
R1	3-22-10	2.5 Ω, 10-watt, 5% resistor
R2	3-22-10	2.5 Ω, 10-watt, 5% resistor
R3	6-102-1	1000 Ω, 1-watt, 5% resistor
SW1	61-49	Switch
T1	54-1027	Power transformer

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
V1	412-72	28-volt lamp
V2	412-87	6-volt lamp
CB1	65-77	.4-ampere circuit breaker (120 VAC)
CB1	65-78	.2-ampere circuit breaker (240 VAC)

BASE, HEAD, & TORSO

BASE PARTS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
A1	420-637	Drive motor
A2	420-640	Steering motor
C1	21-176	.01 μF ceramic capacitor
C2	21-176	.01 μF ceramic capacitor
SW1	61-3	Switch
U201	442-674	7812 integrated circuit
U204	442-674	7812 integrated circuit

HEAD PARTS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
A11/A12	473-34	Sonar transducer (pair)
A13/A14	473-34	Sonar transducer (pair)
A15	401-163	Speaker
	418-45	6-volt battery
F3	421-2	3-ampere fuse
LDR1	9-67	Photo resistor, 133 kΩ
	92-745	Head panel
	94-633	Sonar pad
	209-95	Decorative grille
	390-2315	ET-18 label
	390-2316	Keyboard label
	390-2317	Experimental board label
	446-736	Readout window

TORSO PARTS

CIRCUIT Comp. No.	HEATH Part No.	DESCRIPTION
A3	420-646	Head motor
B1	418-45	6-volt battery
B2	418-45	6-volt battery
B3	418-45	6-volt battery
F1	421-2	3-ampere fuse
F2	421-5	4-ampere fuse
L1	45-62	26 μH choke
L2	45-62	26 μH choke
Q703	417-857	MJE5976 transistor
Q706	417-857	MJE5976 transistor
	95-657	Body front
	95-658	Body rear
	203-2113-2	Body side panel
	205-1883-1	Encoder disk
	390-2072	Body label
	655-17	Drive wheel
	655-18	Rear wheel
C3	21-21	200 pF ceramic capacitor
C4	21-21	200 pF ceramic capacitor
C5	27-85	.22 μF Mylar capacitor

APPENDIX

Heathkit®

PROGRAMMER'S INFORMATION SHEET

Robot owners may copy this sheet to use as a handy aid for use when programming their Robot.

INTERPRETER COMMANDS

COMMAND FORM*	TITLE
02	Abort Drive Motor
03	Abort Steering Motor
04	Abort Arm Motors
05	Abort Speech
1C OO	Branch if Base Busy
1D OO	Branch if Steering Busy
1E OO	Branch if Arm Busy
1F OO	Branch if Speech Busy
21	Zero
3A	Return to Executive ("Ready")
(3F)	(Change to Robot Language)
41	Enable Light Detector
42	Enable Sound Detector
45	Enable Ultrasonic Ranging
4B	Enable Motion Detector
4E	Enable Display
51	Disable Light Detector
52	Disable Sound Detector
55	Disable Ultrasonic Ranging
5B	Disable Motion Detector
5E	Disable Display
61 OO	Speak, Continue (index)
62 OO	Speak, Wait (index)
71 MM MM	Speak, Continue (extended)
72 MM MM	Speak, Wait (extended)
83	Change to Machine Language [3F, SWI changes back to Robot]
87 XX XX	Sleep (immediate)
8F XX XX	Pause (immediate)
BF MM MM	Jump if Speaking (extended)
C3 SS XX	Motor Move, wait abs (immediate)
CC SS XX	Motor Move, continue abs (immed)
D3 SS XX	Motor Move, wait rel (immediate)
DC SS XX	Motor Move, continue rel (immed)
E3 OO	Motor Move, wait abs (index)
EC OO	Motor Move, continue abs (index)
F3 MM MM	Motor Move, wait abs (extended)
FC MM MM	Motor Move, continue abs (extended)
FD **	Motors, Move All abs (immediate)

* XX = distance, position, time, etc. byte
SS = select motor, speed, direction byte
MM = memory address byte
OO = offset number byte
**Seven motor position bytes: extend, shoulder, rotate, pivot, gripper, head, and steering.

MOTOR CONTROL COMMANDS (Byte 1 = Opcode)

	Byte 2	Byte 3
hex no. =	S S	X X
binary =	mmms sDdd	dddd dddd

where:

mmm are the motor select bits.

000 = drive motor.	100 = wrist pivot motor.
001 = extend motor.	101 = gripper motor.
010 = shoulder motor.	110 = head motor.
011 = wrist rotate motor.	111 = steering motor.

ss are the speed select bits:
01 = slow. 10 = medium. 11 = fast.

D selects which way the motors run (for relative commands only). If D equals:

0 = the position increases as the motor runs.*
1 = the position decreases as the motor runs.*
*"Position" is the number stored in memory for that motor.

dd dddd dddd = "Position" — how far, or to where, the motor turns. Highest 2 bits are for drive motor only.

SUBROUTINES

INCH (F777)	Key debounce.
OUTCH (F7C8)	Create character.
OUTHEX (F7B5)	Output 1 hex character.
OUTBYT (F7AD)	Output 2 hex characters.
CLRDIX (F65B)	Clear display, set DIGADD.
OUTSTR (F7E5)	Output character string.
DSPLAY (F6F9)	OUTBYT, used 1, 2, or 3 times.
IHB (F796)	Input hex byte, to display and accumulator A.
REDIS (F64E)	Reset DIGADD for left-most position.
DIGADD (0FF2, 0FF3)	Location for storing the next character's display position.

MEMORY ADDRESSES AND RANGES

(Useable RAM 0039 to 0EdE)

Motor	Address	Range
Extend Position	0000	(in)00–98(out)
Shoulder Position	0001	(dn)00–86(up)
Rotate Position	0002	(CCW)00–93(CW)*
Pivot Position	0003	(up)00–A5(dn)
Gripper Position	0004	(closed)00–75(open)
Head Position	0005	(CCW)00–C2(CW)
Steering	0006	(L turn)00–93(R turn)
Ultrasonic Ranging Hits	0010	
Ultrasonic Range	0011	

I/O PORT ADDRESSES AND VARIABLES

Sense Board C240
Experimental (In) C2A0
Experimental (Out) C220

READING CLOCK DATA

1. STAA (A0) at $C300.
2. STAA (variable select) at $C200.
3. LDAA (time variable) from $C300.

INPUT (VARIABLE SELECT)	OUTPUT (TIME VARIABLE)	
00	SEC(1)	
01	SEC(10)	
02	MIN(1)	
03	MIN(10)	
04	HR(1)	
05	HR(10)	+4 = P.M. +8 = 24 HR. CLOCK
06	DAY OF WEEK	
07	DAY(1)	
08	DAY(10)	+4 = FEB. LEAP YEAR
09	MO(1)	
0A	MO(10)	
0B	YR(1)	
0C	YR(10)	

AUTOMATIC JUMPS

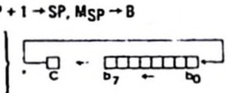
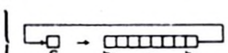
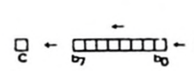
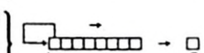
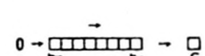
(Executive Mode)

User 1	0030, 0031, 0032.
User 2	0033, 0034, 0035.
User 3	0036, 0037, 0038.

INTERRUPT VECTORS

Motion Detect	0027, 0028, 0029.
Trigger	002A, 002B, 002C.
Experimental Board IRQ (low)	002D, 002E, 002F.

6808 INSTRUCTION SET

ACCUMULATOR AND MEMORY OPERATIONS	MNEMONIC	ADDRESSING MODES IMMED OP	~	#	DIRECT OP	~	#	INDEX OP	~	#	EXTND OP	~	#	INHER OP	~	#	BOOLEAN/ARITHMETIC OPERATION (All register labels refer to contents)	COND. CODE REG. 5 H	4 I	3 N	2 Z	1 V	0 C
Add	ADDA	8B	2	2	9B	3	2	AB	5	2	BB	4	3				A + M → A	↕	•	↕	↕	↕	↕
	ADDB	CB	2	2	DB	3	2	EB	5	2	FB	4	3				B + M → B	↕	•	↕	↕	↕	↕
Add Acmltrs	ABA													1B	2	1	A + B → A	↕	•	↕	↕	↕	↕
Add with Carry	ADCA	89	2	2	99	3	2	A9	5	2	B9	4	3				A + M + C → A	↕	•	↕	↕	↕	↕
	ADCB	C9	2	2	D9	3	2	E9	5	2	F9	4	3				B + M + C → B	↕	•	↕	↕	↕	↕
And	ANDA	84	2	2	94	3	2	A4	5	2	B4	4	3				A • M → A	•	•	↕	↕	R	•
	ANDB	C4	2	2	D4	3	2	E4	5	2	F4	4	3				B • M → B	•	•	↕	↕	R	•
Bit Test	BITA	85	2	2	95	3	2	A5	5	2	B5	4	3				A • M	•	•	↕	↕	R	•
	BITB	C5	2	2	D5	3	2	E5	5	2	F5	4	3				B • M	•	•	↕	↕	R	•
Clear	CLR							6F	7	2	7F	6	3				00 → M	•	•	R	S	R	R
	CLRA													4F	2	1	00 → A	•	•	R	S	R	R
	CLRB													5F	2	1	00 → B	•	•	R	S	R	R
Compare	CMPA	81	2	2	91	3	2	A1	5	2	B1	4	3				A − M	•	•	↕	↕	↕	↕
	CMPB	C1	2	2	D1	3	2	E1	5	2	F1	4	3				B − M	•	•	↕	↕	↕	↕
Compare Acmltrs	CBA													11	2	1	A − B	•	•	↕	↕	↕	↕
Complement, 1's	COM							63	7	2	73	6	3				$\overline{M}$ → M	•	•	↕	↕	R	S
	COMA													43	2	1	$\overline{A}$ → A	•	•	↕	↕	R	S
	COMB													53	2	1	$\overline{B}$ → B	•	•	↕	↕	R	S
Complement, 2's (Negate)	NEG							60	7	2	70	6	3				00 − M → M	•	•	↕	↕	1	2
	NEGA													40	2	1	00 − A → A	•	•	↕	↕	1	2
	NEGB													50	2	1	00 − B → B	•	•	↕	↕	1	2
Decimal Adjust, A	DAA													19	2	1	Converts Binary Add. of BCD Characters into BCD Format	•	•	↕	↕	↕	3
Decrement	DEC							6A	7	2	7A	6	3				M − 1 → M	•	•	↕	↕	4	•
	DECA													4A	2	1	A − 1 → A	•	•	↕	↕	4	•
	DECB													5A	2	1	B − 1 → B	•	•	↕	↕	4	•
Exclusive OR	EORA	88	2	2	98	3	2	A8	5	2	B8	4	3				A ⊕ M → A	•	•	↕	↕	R	•
	EORB	C8	2	2	D8	3	2	E8	5	2	F8	4	3				B ⊕ M → B	•	•	↕	↕	R	•
Increment	INC							6C	7	2	7C	6	3				M + 1 → M	•	•	↕	↕	5	•
	INCA													4C	2	1	A + 1 → A	•	•	↕	↕	5	•
	INCB													5C	2	1	B + 1 → B	•	•	↕	↕	5	•
Load Acmltr	LDAA	86	2	2	96	3	2	A6	5	2	B6	4	3				M → A	•	•	↕	↕	R	•
	LDAB	C6	2	2	D6	3	2	E6	5	2	F6	4	3				M → B	•	•	↕	↕	R	•
Or, Inclusive	ORAA	8A	2	2	9A	3	2	AA	5	2	BA	4	3				A + M → A	•	•	↕	↕	R	•
	ORAB	CA	2	2	DA	3	2	EA	5	2	FA	4	3				B + M → B	•	•	↕	↕	R	•
Push Data	PSHA													36	4	1	A → M_{SP}, SP − 1 → SP	•	•	•	•	•	•
	PSHB													37	4	1	B → M_{SP}, SP − 1 → SP	•	•	•	•	•	•
Pull Data	PULA													32	4	1	SP + 1 → SP, M_{SP} → A	•	•	•	•	•	•
	PULB													33	4	1	SP + 1 → SP, M_{SP} → B	•	•	•	•	•	•
Rotate Left	ROL							69	7	2	79	6	3				M	•	•	↕	↕	6	↕
	ROLA													49	2	1	A 	•	•	↕	↕	6	↕
	ROLB													59	2	1	B	•	•	↕	↕	6	↕
Rotate Right	ROR							66	7	2	76	6	3				M	•	•	↕	↕	6	↕
	RORA													46	2	1	A 	•	•	↕	↕	6	↕
	RORB													56	2	1	B	•	•	↕	↕	6	↕
Shift Left, Arithmetic	ASL							68	7	2	78	6	3				M	•	•	↕	↕	6	↕
	ASLA													48	2	1	A 	•	•	↕	↕	6	↕
	ASLB													58	2	1	B	•	•	↕	↕	6	↕
Shift Right, Arithmetic	ASR							67	7	2	77	6	3				M	•	•	↕	↕	6	↕
	ASRA													47	2	1	A 	•	•	↕	↕	6	↕
	ASRB													57	2	1	B	•	•	↕	↕	6	↕
Shift Right, Logic.	LSR							64	7	2	74	6	3				M	•	•	R	↕	6	↕
	LSRA													44	2	1	A 	•	•	R	↕	6	↕
	LSRB													54	2	1	B	•	•	R	↕	6	↕
Store Acmltr.	STAA				97	4	2	A7	6	2	B7	5	3				A → M	•	•	↕	↕	R	•
	STAB				D7	4	2	E7	6	2	F7	5	3				B → M	•	•	↕	↕	R	•
Subtract	SUBA	80	2	2	90	3	2	A0	5	2	B0	4	3				A − M → A	•	•	↕	↕	↕	↕
	SUBB	C0	2	2	D0	3	2	E0	5	2	F0	4	3				B − M → B	•	•	↕	↕	↕	↕
Subract Acmltrs.	SBA													10	2	1	A − B → A	•	•	↕	↕	↕	↕
Subtr. with Carry	SBCA	82	2	2	92	3	2	A2	5	2	B2	4	3				A − M − C → A	•	•	↕	↕	↕	↕
	SBCB	C2	2	2	D2	3	2	E2	5	2	F2	4	3				B − M − C → B	•	•	↕	↕	↕	↕
Transfer Acmltrs	TAB													16	2	1	A → B	•	•	↕	↕	R	•
	TBA													17	2	1	B → A	•	•	↕	↕	R	•
Test, Zero or Minus	TST							6D	7	2	7D	6	3				M − 00	•	•	↕	↕	R	R
	TSTA													4D	2	1	A − 00	•	•	↕	↕	R	R
	TSTB													5D	2	1	B − 00	•	•	↕	↕	R	R

INDEX REGISTER AND STACK POINTER OPERATIONS	MNEMONIC	IMMED OP	IMMED ~	IMMED #	DIRECT OP	DIRECT ~	DIRECT #	INDEX OP	INDEX ~	INDEX #	EXTND OP	EXTND ~	EXTND #	INHER OP	INHER ~	INHER #	BOOLEAN/ARITHMETIC OPERATION	5 H	4 I	3 N	2 Z	1 V	0 C
Compare Index Reg	CPX	8C	3	3	9C	4	2	AC	6	2	BC	5	3				$(X_H/X_L) - (M/M+1)$	•	•	7	↕	8	•
Decrement Index Reg	DEX													09	4	1	X − 1 → X	•	•	•	↕	•	•
Decrement Stack Pntr	DES													34	4	1	SP − 1 → SP	•	•	•	•	•	•
Increment Index Reg	INX													08	4	1	X + 1 → X	•	•	•	↕	•	•
Increment Stack Pntr	INS													31	4	1	SP + 1 → SP	•	•	•	•	•	•
Load Index Reg	LDX	CE	3	3	DE	4	2	EE	6	2	FE	5	3				$M \to X_H, (M+1) \to X_L$	•	•	9	↕	R	•
Load Stack Pntr	LDS	8E	3	3	9E	4	2	AE	6	2	BE	5	3				$M \to SP_H, (M+1) \to SP_L$	•	•	9	↕	R	•
Store Index Reg	STX				DF	5	2	EF	7	2	FF	6	3				$X_H \to M, X_L \to (M+1)$	•	•	9	↕	R	•
Store Stack Pntr	STS				9F	5	2	AF	7	2	BF	6	3				$SP_H \to M, SP_L \to (M+1)$	•	•	9	↕	R	•
Indx Reg → Stack Pntr	TXS													35	4	1	X − 1 → SP	•	•	•	•	•	•
Stack Pntr → Indx Reg	TSX													30	4	1	SP + 1 → X	•	•	•	•	•	•

JUMP AND BRANCH OPERATIONS	MNEMONIC	RELATIVE OP	RELATIVE ~	RELATIVE #	INDEX OP	INDEX ~	INDEX #	EXTND OP	EXTND ~	EXTND #	INHER OP	INHER ~	INHER #	BRANCH TEST	5 H	4 I	3 N	2 Z	1 V	0 C
Branch Always	BRA	20	4	2										None	•	•	•	•	•	•
Branch If Carry Clear	BCC	24	4	2										C = 0	•	•	•	•	•	•
Branch If Carry Set	BCS	25	4	2										C = 1	•	•	•	•	•	•
Branch If = Zero	BEQ	27	4	2										Z = 1	•	•	•	•	•	•
Branch If ⩾ Zero	BGE	2C	4	2										N ⊕ V = 0	•	•	•	•	•	•
Branch If > Zero	BGT	2E	4	2										Z + (N ⊕ V) = 0	•	•	•	•	•	•
Branch If Higher	BHI	22	4	2										C + Z = 0	•	•	•	•	•	•
Branch If ⩽ Zero	BLE	2F	4	2										Z + (N ⊕ V) = 1	•	•	•	•	•	•
Branch If Lower Or Same	BLS	23	4	2										C + Z = 1	•	•	•	•	•	•
Branch If < Zero	BLT	2D	4	2										N ⊕ V = 1	•	•	•	•	•	•
Branch If Minus	BMI	2B	4	2										N = 1	•	•	•	•	•	•
Branch If Not Equal Zero	BNE	26	4	2										Z = 0	•	•	•	•	•	•
Branch If Overflow Clear	BVC	28	4	2										V = 0	•	•	•	•	•	•
Branch If Overflow Set	BVS	29	4	2										V = 1	•	•	•	•	•	•
Branch If Plus	BPL	2A	4	2										N = 0	•	•	•	•	•	•
Branch To Subroutine	BSR	8D	8	2										See Special Operations	•	•	•	•	•	•
Jump	JMP				6E	4	2	7E	3	3					•	•	•	•	•	•
Jump To Subroutine	JSR				AD	8	2	BD	9	3					•	•	•	•	•	•
No Operation	NOP										01	2	1	Advances Prog. Cntr. Only	•	•	•	•	•	•
Return From Interrupt	RTI										3B	10	1	See special Operations			10			
Return From Subroutine	RTS										39	5	1		•	•	•	•	•	•
Software Interrupt	SWI										3F	12	1		•	S	•	•	•	•
Wait for Interrupt	WAI										3E	9	1		•	11	•	•	•	•

CONDITIONS CODE REGISTER OPERATIONS	MNEMONIC	INHER OP	INHER ~	INHER #	BOOLEAN OPERATION	5 H	4 I	3 N	2 Z	1 V	0 C
Clear Carry	CLC	0C	2	1	0 → C	•	•	•	•	•	R
Clear Interrupt Mask	CLI	0E	2	1	0 → I	•	R	•	•	•	•
Clear Overflow	CLV	0A	2	1	0 → V	•	•	•	•	R	•
Set Carry	SEC	0D	2	1	1 → C	•	•	•	•	•	S
Set Interrupt Mask	SEI	0F	2	1	1 → I	•	S	•	•	•	•
Set Overflow	SEV	0B	2	1	1 → V	•	•	•	•	S	•
Acmltr A → CCR	TAP	06	2	1	A → CCR			12			
CCR → Acmltr A	TPA	07	2	1	CCR → A	•	•	•	•	•	•

CONDITION CODE REGISTER NOTES:

(Bit set if test is true and cleared otherwise)

1. (Bit V) Test: Result = 10000000?
2. (Bit C) Test: Result = 00000000?
3. (Bit C) Test: Decimal value of most significant BCD Character greater than nine? (Not cleared if previously set.)
4. (Bit V) Test: Operand = 10000000 prior to execution?
5. (Bit V) Test: Operand = 01111111 prior to execution?
6. (Bit V) Test: Set equal to result of N ⊕ C after shift has occurred
7. (Bit N) Test: Sign bit of most significant (MS) byte of result = 1?
8. (Bit V) Test: 2's complement overflow from subtraction of LS bytes?
9. (Bit N) Test: Result less than zero? (Bit 15 = 1)
10. (All) Load Condition Code Register from Stack. (See Special Operations)
11. (Bit I) Set when interrupt occurs. If previously set, a Non Maskable Interrupt is required to exit the wait state.
12. (ALL) Set according to the contents of Accumulator A.

LEGEND:

OP	Operation Code (Hexadecimal);
~	Number of MPU Cycles;
#	Number of Program Bytes;
+	Arithmetic Plus;
−	Arithmetic Minus;
•	Boolean AND;
M_{SP}	Contents of memory location pointed to be Stack Pointer;
+	Boolean Inclusive OR;
⊕	Boolean Exclusive OR;
$\overline{M}$	Complement of M;
→	Transfer Into;
0	Bit = Zero;
00	Byte = Zero;
H	Half-carry from bit 3;
I	Interrupt mask
N	Negative (sign bit)
Z	Zero (byte)
V	Overflow, 2's complement
C	Carry from bit 7
R	Reset Always
S	Set Always
↕	Test and set if true, cleared otherwise
•	Not Affected
CCR	Condition Code Register
LS	Least Significant
MS	Most Significant

ROM OPTIONS AND CPU JUMPER PLACEMENT

The Robot is supplied with an 8k × 8 mask programmable ROM at U418 on the CPU board. Right next to the ROM are the three jumpers, configured to facilitate proper operation with the system. If you ever add another ROM at U417, or change the operating system ROM at U418, you will need the information below to place the jumpers for each position correclty.

You may use any one of three different sizes of ROMs (2k × 8, 4k × 8, or 8k × 8) at either location, provided that the ROMs conform to the industry standard pinouts for such devices. EPROMs, mask programmable ROMs, and electrically programmable ROMs are all acceptable; typical EPROMs, with the correct pinouts from Motorola, are MCM2716 (2k), MCM2532 (4k), and MCM68764 (8k).

Refer to the following two sections for standard jumper placement, for the standard ROM in U418, and for all other types of ROM in both U418 and U417. Figure 11 shows the ROM locations as seen inside the Robot body, and illustrates the jumper positions. The table below indicates which jumpers are required for each position and ROM type.

	U418			U417		
ROM TYPE	J1 or J2	J3 or J4	J5 or J6	J7 or J8	J9 or J10	J11 or J12
2k × 8	J1	J3	*	J7	J9	*
4k × 8	J2	J3	J5	J8	J9	*
8k × 8	J2	J4	J6	J8	J10	J12*

*READ THE SECTION BELOW DEALING WITH THE TYPE OF ROM THAT YOU HAVE.

U418 MONITOR ROM

2k × 8 ROM

J1 is needed to tie the CE input low. J3 ties V_{PP} input to V_{CC}. J5 places ROM beginning address at F800. NOTE: This produces a redundant condition in memory from F000 to F7FF.

4k × 8 ROM

J2 connects the A11 line. J3 ties V_{PP} to V_{CC}. J5 places the ROM beginning address at F000.

8k × 8 ROM

J2 connects the A11 line. J4 connects the A12 line. J6 places the ROM beginning address at E000.

U417 OPTIONAL ROM

2k × 8 ROM

J7 ties the CE input low. J9 ties the V_{PP} input to V_{CC}. The choice of J11 or J12 depends on what ROM is in U418. If U418 is an 8k, select J12 to prevent overlapping addresses. If U418 has been changed to a 2k or a 4k, select either J11 or J12. J11 makes the ROM beginning address E000 and produces redundant conditions in memory from E800 to EFFF. J12 makes the ROM beginning address 2000 and produces redundant conditions in memory from 2800 to 3FFF.

4k × 8 ROM

J8 connects the A11 line. J9 ties the V_{PP} input to V_{CC}. Again, the J11 versus J12 decision depends on what is in U418. If U418 is an 8k, select J12 to prevent overlapping of addresses. If U418 has been changed to a 2k or a 4k, select either J11 or J12. J11 makes the ROM beginning address E000 without any redundancy. J12 makes the ROM beginning address at 2000 and produces a redundant condition in memory from 3000 to 3FFF.

8k × 8 ROM

J8 connects the A11 line. J10 connects the A12 line. J12 must be selected to place the beginning address at 2000. Do not select J11, doing so could damage both ROMs.

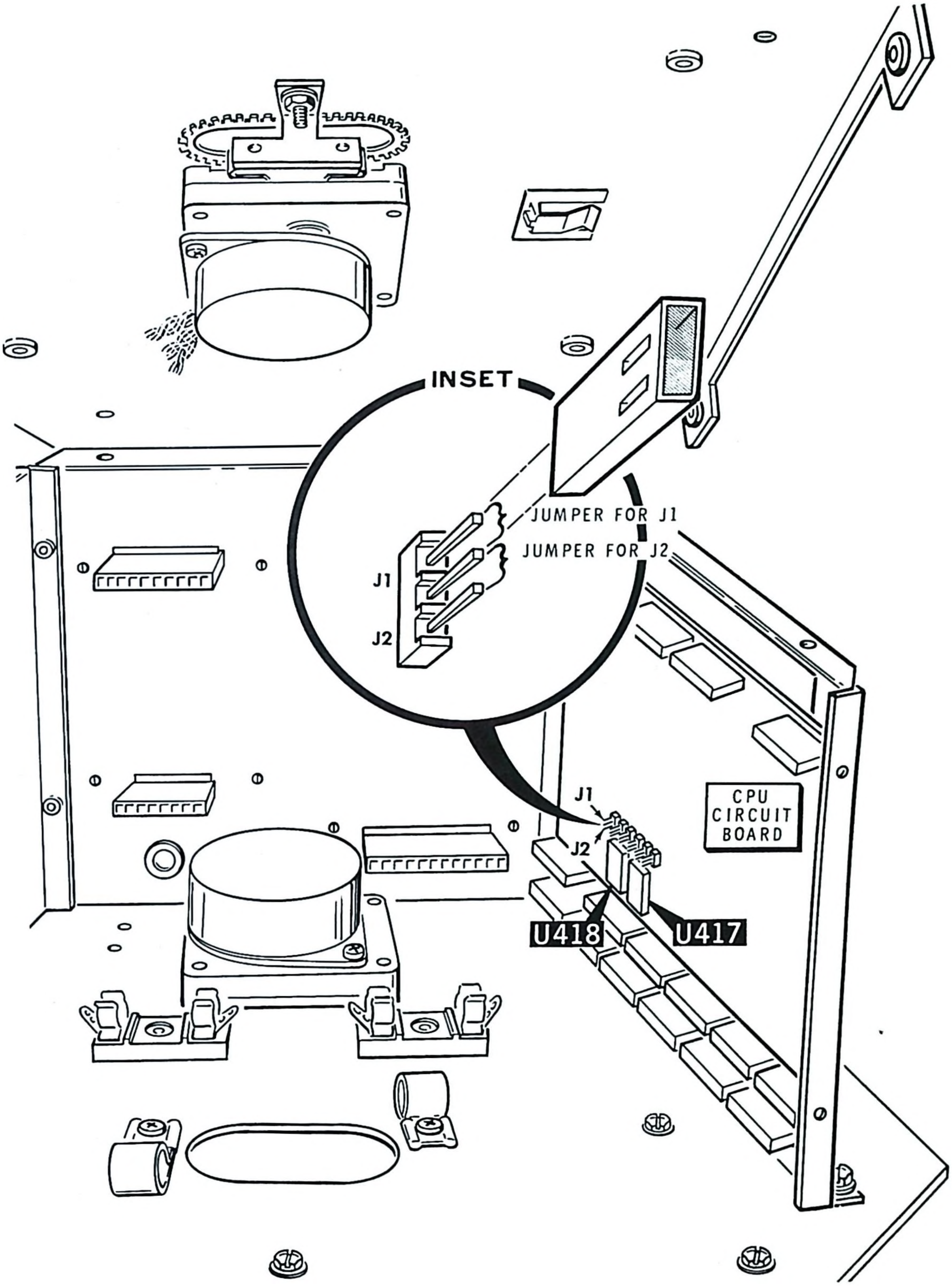
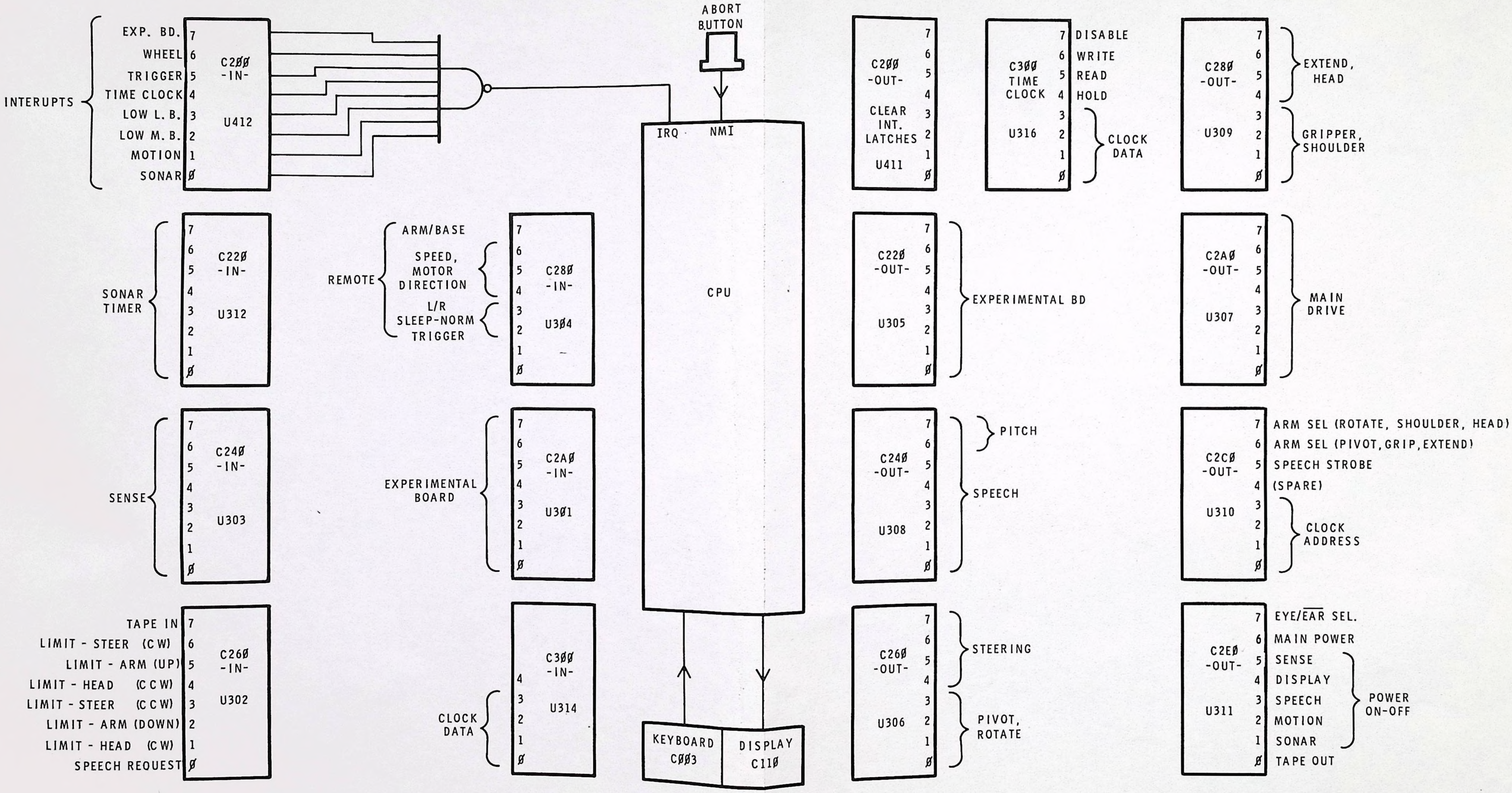

INSET
JUMPER FOR J1
JUMPER FOR J2
J1
J2
J1
J2
CPU CIRCUIT BOARD
U418
U417

FIGURE 11


IN PORTS
OUT PORTS
INTERUPTS
EXP. BD.
WHEEL
TRIGGER
TIME CLOCK
LOW L. B.
LOW M. B.
MOTION
SONAR
C2ØØ -IN-
U412
ABORT BUTTON
IRQ
NMI
CPU
C2ØØ -OUT-
CLEAR INT. LATCHES
U411
C3ØØ TIME CLOCK
U316
DISABLE
WRITE
READ
HOLD
CLOCK DATA
C28Ø -OUT-
U309
EXTEND, HEAD
GRIPPER, SHOULDER
SONAR TIMER
C22Ø -IN-
U312
REMOTE
ARM/BASE
SPEED, MOTOR DIRECTION
L/R
SLEEP-NORM
TRIGGER
C28Ø -IN-
U3Ø4
C22Ø -OUT-
U305
EXPERIMENTAL BD
C2AØ -OUT-
U307
MAIN DRIVE
SENSE
C24Ø -IN-
U303
EXPERIMENTAL BOARD
C2AØ -IN-
U3Ø1
C24Ø -OUT-
U308
PITCH
SPEECH
C2CØ -OUT-
U310
ARM SEL (ROTATE, SHOULDER, HEAD)
ARM SEL (PIVOT, GRIP, EXTEND)
SPEECH STROBE
(SPARE)
CLOCK ADDRESS
TAPE IN
LIMIT - STEER (CW)
LIMIT - ARM (UP)
LIMIT - HEAD (CCW)
LIMIT - STEER (CCW)
LIMIT - ARM (DOWN)
LIMIT - HEAD (CW)
SPEECH REQUEST
C26Ø -IN-
U302
CLOCK DATA
C3ØØ -IN-
U314
KEYBOARD CØØ3
DISPLAY C11Ø
C26Ø -OUT-
U306
STEERING
PIVOT, ROTATE
C2EØ -OUT-
U311
EYE/EAR SEL.
MAIN POWER
SENSE
DISPLAY
SPEECH
MOTION
SONAR
TAPE OUT
POWER ON-OFF

PORT MAP

PROGRAMS LIST

MEMORY ADDRESS	PAGE NO.	DESCRIPTION
0200–0204	58	"SPEECH TEST"
0205–0211	48	"ULTRASONIC ADJ."
0220–0222	49	"SOUND DETECT ADJ."
0225–0227	49	"LIGHT DETECT ADJ."
0230–0232	49	"MOTION DETECT ADJ."
0300–0325	20	"INCH"—Moves head as 1 of 3 keys is pressed.
0326–032C	21	"OUTCH"—Creates a character on display.
032D–0333	21	"OUTHEX" & "OUTBYT"—Sends 1 or 2 hex numbers to the display.
0335–0342	22	"CLRDIS" & "OUTSTR"—Creates "HELLO" on the display.
0345–0354	22	"DSPLAY" & "REDIS"—Sends 2, 4, or 6 hex numbers to the display.
0360–0375	23	"IHB"—Speaks whatever is at the address that you key in.
0400–0415	26	"SPEAK"—Talks and turns its head.
0420–044A	26	"JUMP IF SPEAKING"—Strings together phrases from memory, and laughs.
0450–0475	28	"BRANCH IF BUSY"—Scan horizon and display the range and number of hits.
0480–0496	27	"ABORT" & "ENABLE"—Start and stop Robot, using noise as trigger.
0500–0543	34	"TRIGGER INTERRUPT"—Trigger causes Robot to say "Trigger!".
054A–0564	36	"REAL TIME CLOCK"—Display shows seconds ticking by.
056A–0581	37	"RANDOM"—Generates pseudo random numbers on the display.
0600–0627	16	"SAMPLE PROGRAM"—Light LED segments in sequence.
0700–07D1	30	"MISCELLANEOUS"—Wastes time by do loop, pause, and sleep.

INDEX

This index is meant to compliment the Table of Contents. If you know what section of the Manual you wish to look at, use the Table of Contents to find the Page number. But if you only know a subject or keyword that you wish to investigate, begin by looking here.

Abort Command, 27
Adjustments, 43
  Arm Rotation, 44
  Drive Wheel Sensor, 51
  Motion Detector, 49
  Power Supply, 47
  Sense Board, 49
  Shoulder Rotation, 47
  Speech, 50
  Speed, 48
  Steering, 51
  Time, 50
  Ultrasonics, 48
Branch Commands, 26, 28
Breakpoints, 14
Circuit Description, 72
  Arm Drive, 74
  CPU, 79
  Charger, 72
  Display, 78
  Experimental, 82
  I/O, 80
  Main Drive, 74
  Motion Detector, 77
  Motors, 75
  Power Supply, 72
  Sense, 76
  Speech, 77
  Teaching Pendant, 73
  Ultrasonic Ranging Receiver, 76
  Ultrasonic Ranging Transmitter, 75
Clock, Real Time, 34
Customer Service, (inside back cover)
Data Stored in Memory, 32
Enable-Disable Commands, 27
FCC Warning, 2
HINZVC, 10
Interpreter, 23
Keyboard Operation,10
Memory-Stored Data, 32
Motor Control Commands, 25
Motor Memory Address, 32
Pause Command, 29
Port Map, (fold-out from Page 119)
Programs List, 120
Programmers' Information Sheet, 115
Random Number Generator, 37
Replacement Parts List, 99
ROM Subroutines, (see "Subroutines")
Semiconductor Identification Charts, 85
6800 Instruction Set, 116
Sleep Command, 29
Speak Commands, 26
Specifications, 7
Subroutines, 20
  CLRDIS, 22
  DIGADD, 21
  DSPLAY, 22
  IHB, 23
  INCH, 20
  OUTBYT, 22
  OUTCH, 21
  OUTHEX, 21
  OUTSTR, 22
  REDIS, 23
Theory of Operation, 71
User Defined Keys, 32
Warning, FCC 2
Wiring:
  Schematics, (fold-in)
  Interconnect Diagram, (fold-out)
X-Ray Views, 83
Zero Command, 29

CUSTOMER SERVICE

REPLACEMENT PARTS

Please provide complete information when you request replacements from either the factory or Heath/Zenith Computers and Electronics Centers. Be certain to include the **HEATH** part number exactly as it appears in the parts list.

ORDERING FROM THE FACTORY

Print all of the information requested on the parts order form furnished with this product and mail it to Heath. For telephone orders (parts only) dial 616 982-3571. If you are unable to locate an order form, write us a letter or card including.

- Heath part number.
- Model number.
- Date of purchase.
- Location purchased or invoice number.
- Nature of the defect.
- Your payment or authorization for COD shipment of parts not covered by warranty.

Mail letters to: Heath Company
Benton Harbor
MI 49022
Attn: Parts Replacement

Retain original parts until you receive replacements. Parts that should be returned to the factory will be listed on your packing slip.

OBTAINING REPLACEMENTS FROM HEATH/ZENITH COMPUTERS AND ELECTRONICS CENTERS

For your convenience, "over the counter" replacement parts are available from the Heath/Zenith Computers and Electronics Centers listed in your catalog. Be sure to bring in the original part and purchase invoice when you request a warranty replacement from a Heath/Zenith Computers and Electronics Centers.

TECHNICAL CONSULTATION

Need help with your kit? — Self-Service? — Construction? – Operation? — Call or write for assistance; you'll find our Technical Consultants eager to help with just about any technical problem except "customizing" for unique applications.

The effectiveness of our consultants service depends on the information you furnish. Be sure to tell us:

- The Model number and Series number from the blue and white label.
- The date of purchase.
- An exact description of the difficulty.
- Everything you have done in attempting to correct the problem.

Also include switch positions, connections to other units, operating procedures, voltage readings, and any other information you think might be helpful.

Please do not send parts for testing, unless this is specifically requested by our Consultants.

Hints: Telephone traffic is lightest at midweek — please be sure your Manual and notes are on hand when you call.

Heath/Zenith Computers and Electronics Centers facilities are also available for telephone or "walk-in" personal assistance.

REPAIR SERVICE

Service facilities are available, if they are needed, to repair any portions of your Robot that need service. (Kits that have been modified, soldered with paste flux or acid core solder, cannot be accepted for repair.) Identify the questionable area by using the "Troubleshooting" section of your Technical Manual and Heath's Technical Consultants (if necessary), then return only the questionable portion for service. Never send a complete Robot unless you are instructed to do so by a Technical Consultant.

If it is convenient, deliver the questionable portion personally to a Heath/Zenith Computers and Electronics Centers. For warranty parts replacement, supply a copy of the invoice or sales slip.

If you should need to ship some portion of the Robot to the factory, attach a letter containing the following information directly to that portion of the Robot:

- Your name and address.
- Date of purchase and invoice number.
- Copies of all correspondence relevant to the service of the kit.
- A brief description of the difficulty.
- Authorization to return your kit COD for the service and shipping charges. (This will reduce the possibility of delay.)

Package any portions of the Robot that you wish to ship in a strong carton with at least THREE INCHES of resilient packing material (shredded paper, excelsior, etc.) on all sides. Contact the Heath Company for instructions for sending the entire Robot or large portions of it.

Seal the carton with reinforced gummed tape, and mark it "Fragile" on at least two sides. Remember, the carrier will not accept liability for shipping damage if the unit is insufficiently packed. Ship by prepaid express, United Parcel Service, or insured Parcel Post to:

Heath Company
Service Department
Benton Harbor, Michigan 49022

595-3071-03