SOFTWARE REFERENCE MANUAL

Digital Computer System

Model H8





HEATH COMPANY
BENTON HARBOR, MICHIGAN 49022



Contents

SPECIAL DISCLAIMER

Heath cannot provide consultation on user-developed programs or modified versions of Heath Software products.

These software products were developed for the Heath Company by the Wintek Corporation. Software copyrights reside with Wintek Corporation.



INTRODUCTION



TABLE OF CONTENTS

PANEL MONITOR (PAM-8) 0-7
CONSOLE DEBUGGER (BUG-8) 0-8
HEATH TEXT EDITOR (TED-8) 0-8
HEATH ASSEMBLY LANGUAGE (HASL-8) 0-10
BENTON HARBOR BASIC 0-11 TAPE FILES 0-12 System Record Structure 0-13 Label Record Format 0-14 System Data Formats 0-14 Reading the Displays 0-15
USING THE MAGNETIC TAPE SYSTEM
USING THE PAPER TAPE SYSTEM 0-18 Paper Tape Operating Hints 0-19
PRODUCT INSTALLATION 0-19 Creating a Configured Tape 0-19 Loading From a Configured Tape 0-21 Copying an Existing Memory Tape 0-22 Installing a Pitch 0-22 Using an ASR Console 0-23 Using a 110 Baud Console Terminal 0-23 Console Interface 0-23 Reporting Software Problems 0-25
APPENDIX A (ASR Patches) 0-26
APPENDIX B (Console Driver Listing) 0-36
APPENDIX C (I/O and Memory Maps) 0-50
APPENDIX D (ASCII Characters) 0-52
APPENDIX E (Decimal to Octal Tables) 0-56
APPENDIX F (Memory Table, Offset Octal & Decimal Boundaries
INDEX



This Software Reference Manual includes all the information you will need to be thoroughly familiar with the software products supplied with your H8 Computer. These software products are: the front Panel Monitor, PAM-8; the Console Debugger routine, BUG-8; the Heath Test Editor, TED-8; the Heath Assembly Language, HASL-8; and Benton Harbor BASIC, Heath Company's version of Dartmouth BASIC. Extended Benton Harbor BASIC, which is available as an optional accessory and includes such additional features as string manipulation, is also included in this Manual.

This book is intended as a reference manual, and, as such, it is as complete as possible. Examples are included to help you understand exactly how the Heath software products carry out their instructions; but they are not designed to teach you programming. If you have never used a text editor and assembler, for example, we recommend that you obtain some instruction from other sources, such as the "Heathkit Continuing Education" courses, prior to reading this material. If you have used editors and assemblers, this Manual will tell you about the special features in the Heath Text Editor (TED-8) and the Heath Assembly Language (HASL-8).

This introduction describes each product briefly and covers those aspects of the packages that are common to all. A separate section then follows for each software product. Each section provides detailed reference information and is followed by one or more Appendices for that product. Be sure to read all of this introductory section so you have a good overview of all of the products.

Heath software products feature a high degree of commonality in many of the modules which make up the individual products. For example, all software products which use the console terminal employ a software module called the Console Terminal Driver. This common usage of the console terminal driver permits you to move easily from one software product to the other, as the operating features are similar. Likewise, all tape handling is carried out through a common tape handling package, and once these features are understood, they are applicable to all products.

Heath software is supplied in three forms: cassette magnetic tape, paper tape, and read/only memory (ROM). The Panel Monitor (PAM-8) is supplied in a ROM (programs supplied in ROM cannot be modified by the user). The Console Debugger (BUG-8), the Heath Text Editor (TED-8), the Heath Assembly Language (HASL-8), and BASIC are supplied with the H8 in cassette form. They are optionally available in paper tape form. The cassettes and the paper tapes are compatible with the required error checking and synchronizing characters used by the front panel monitor system.

A printed copy of the panel monitor source listing is provided to aid you in using PAM-8. The Console Driver Listing and the partial listing (including entry points) of the BASIC floating point package and other BASIC utility packages are also included. All other programs are supplied in binary object forms and listings are **not** available.



PANEL MONITOR (PAM-8)

The ROM Panel Monitor, which is permanently located in the lower 1024 bytes of memory, permits you to load, execute, and debug programs written in 8080 machine language. The Heath Panel Monitor also makes use of the first 64 locations of random access memory. The H8 front panel is used as an I/O device, and it is assigned port numbers 360 and 361. With the Heath Panel Monitor, you can:

- 1. Examine the contents of a memory location.
- 2. Change the contents of a memory location (enter a new program, for example, or modify an old program).
- 3. Examine the contents of any of the 8080 registers.
- 4. Change the contents of any of the 8080 registers.
- 5. Start or stop the execution of a user-written program.
- 6. Execute a user program, a single instruction at a time.
- 7. Dump a program onto either magnetic or paper tape, with error detection codes and synchronization data.
- 8. Load a program from paper or magnetic tape into the desired memory locations.
- 9. Breakpoint a user program.
- Reinitialize to a power up status.

The Heath Panel Monitor also offers the following features:

- 1. The user may automatically increment or decrement memory addresses which are being examined or modified.
- The user may automatically increment or decrement through the registers which are being examined or modified.
- 3. The user is provided with a visual indication of the current mode in which the panel monitor is operating.
- 4. The user is provided with audio feedback upon valid and invalid command and data entry.
- 5. The H8 front panel utilizes an octal display rather than the more difficult to read binary display.
- The front panel key switches and display are available for your programs.
- 7. The front display is operated on a continuously updated basis and, therefore, is active even during the execution of a user program. This feature permits the user to monitor either registers or memory location while his program is operating.

PAM-8 provides the fundamental tape routines by which the user loads all other programs, including the Heath supplied software and user-written software into the computer.

CONSOLE DEBUGGER (BUG-8)

BUG-8 allows you to perform very sophisticated operations from a console terminal with a full active keyboard and display. BUG-8 resides in H8 memory, using approximately 3,000 bytes of storage. You can use BUG-8 to write, load, execute, and debug machine language programs in the H8 computer in octal, decimal, or ASCII format. This package also has many of the features included in PAM-8.

With the Heath Console Debugger, you can:

- 1. Examine the contents of memory locations.
- 2. Alter the contents of memory locations.
- 3. Examine the contents of the CPU registers.
- 4. Alter the contents of the CPU registers.
- 5. Start program execution.
- 6. Execute a program in a single step form.
- 7. Set break points with multiple hit capability.
- 8. Clear break points.
- 9. Load programs from magnetic tape or paper tape.
- 10. Dump programs onto magnetic tape or paper tape.

BUG-8 is an advanced monitor, permitting you to prepare extensive software in machine code format that can be readily debugged and then recorded on a mass storage unit for future use.

HEATH TEXT EDITOR (TED-8)

The Heath TED-8 Text Editor is a general purpose, line-oriented text editor that is used primiarly to prepare source code that can be assembled by the Heath Assembly Language (HASL-8). But while this is its primary purpose, it is also useful for such things as letter writing, preparation of club newspapers, and manuscript editing.

This software product requires an H8 system with 8192 bytes of memory, an ASCII keyboard for text entry, and an ASCII display for text display. If large files are to be used or files are to be saved, a separate input/output tape unit is recommended.



With the Heath Text Editor, you can:

- 1. Read text from a pre-existing text file.
- 2. Create text for a new file.
- 3. Output text to a named tape file.
- 4. Insert new text after a given line.
- 5. Search the text for a given character string.
- 6. Delete a given line or lines.
- 7. Print a particular line or lines.
- 8. Replace a given line.
- 9. Edit a given string; that is, replace a particular string with another string.

All the above functions are supported by a number of special features, some of which are only available on the Heath Text Editor. Some of these features are:

- 1. A wide scope of range expressions, including:
 - A. First line.
 - B. Last line.
 - C. Single line.
 - D. Line to line.
- 2. Count and string versions of range expressions, which permit you to edit lines, plus or minus a certain number of lines from a given line, or to edit all lines containing a certain string.
- 3. You have the option of selecting one of three optional modes. Optional mode A prints the line after operating on it, optional mode B prints the line before operating on it, and optional mode BA prints the line before and after operating on it.
- 4. The use of a Qualifier String (Qualifier Strings permit operating only on the lines containing designated strings).
- Tab. This command lets you set tab stops for entering text. The editor is constructed so that tabs do not occupy extensive user storage.
- 6. A Use statement, which provides a line count and memory usage information.



7. File Labeling Procedures to create new file names in either the input or output mode.

Under the H8 text editor, source code is prepared for the Heath Assembly Language (HASL-8). Once the source code has been prepared, it is written to a cassette tape or paper tape output file. Once this has been done, the user proceeds to the assembler.

HEATH ASSEMBLY LANGUAGE (HASL-8)

Heath Assembly Language runs on a Heath H8 Computer using about 8192 bytes of memory. This program assembles source code and produces object code. HASL-8 utilizes all the standard 8080 mnemonics, extended mnemonics, and numerous psuedo instructions.

Some of the special features of HASL-8 are that it:

- 1. Recognizes five operators: plus, minus, *, /, unary-.
- 2. Recognizes four token operand expressions:
 - A. Integers.
 - B. Symbols.
 - C. Character strings.
 - D. The origin symbol.

HASL-8 is a two pass assembler. Before the user starts assembly, it asks if a binary output is to be generated. On the second pass, it produces the binary if directed to do so, as well as the appropriate listing. The binary object code may be placed on a specified output device or may be placed directly in memory.

HASL-8 features the same terminal controls as do other Heath programs, including a suspend output mode and a discard output mode.



BENTON HARBOR BASIC

BENTON HARBOR BASIC is a modified version of Dartmouth BASIC, an easy-to-learn-and-use conversational language.

The BENTON HARBOR BASIC system is interpretive. That is, it executes each statement as it comes to it. BENTON HARBOR BASIC utilizes an H8 computer with 8K of memory, and appropriate terminal and paper tape or magnetic tape handling capability.

Extended BENTON HARBOR BASIC requires 12K of memory and offers strings. Some of the features of BENTON HARBOR BASIC are:

- 1. Three different data types:
 - A. Numeric data, which has over six digits of accuracy and lies in the range of 10^{-39} to 10^{+38} . Numeric data may be either fixed or floating point.
 - B. Strings, which can be from 0 to 255 characters.
 - C. Boolean values, which permit logical operations.
- 2. Multidimensioned variables.
- 3. BASIC supports fifteen operators, which are:

```
A. –(unary) NOT
```

B. ↑ Exponentiation

C. * /

D. + -

E. <, <=, =, <>, >=, and >

F. OR

G. AND

- 4. Free Format Programs.
- 5. Multiple statements per line.
- 6. Enhanced expression and conditional statement facilities.



BASIC features both command and program modes, where statements may be executed immediately after the line is written or numbered lines may be used so the program will not be executed until a RUN statement is executed.

BASIC also features command completion. In the command mode, BASIC checks inputted characters and, as soon as there are sufficient characters to establish a unique command, the command is completed. This feature saves considerable typing time and reduces errors.

NOTE: In order to fully use the Heath Software package, you must not only review the special features of Heath programs, but you must also know how to use monitors, debuggers, Text Editors, Assemblers, and BASIC. Once you have learned to use such programs, this Software Reference Manual will be an invaluable quick reference on how to carry out specific functions within the H8 software packages.

TAPE FILES

This section describes the tape format used in the Heath H8 Computer System. Tape formats are identical, regardless of the media used. The following terms are used to define the Heath H8 Tape format.

- FILE A logically complete set of data. For example, a memory dump causes the FILE to be written on the tape. Although several files may be written onto one tape, the files are each totally independent of any other information written on that tape. A file consists of one or more records.
- RECORD A record is a discrete block of data written to the tape transport. Each record must be read all at one time. It is not possible to read part of the record, pause, and then read the rest. Each record contains a CRC-16 Check. Each file has a first and last record. They may be the same record in a one-record file. The records in the file are numbered so a missing record can be detected.



System Record Structure

As discussed on Page 0-12, all H8 files consist of one or more records. All of the records have the same format.

SYNs	STX E O F	TYPE	SEQ	COUNT	Data Data	CRC-16
------	-----------	------	-----	-------	-----------	--------

SYN From 20 to 40 ASCII Synchronizing Idle (026) characters.

STX An ASCII STX character. This character, preceded by at least 10 SYN characters, indicates the start of a record. The SYN characters and the STX character are not included in the CRC. Note that a gap may be required between records to allow the tape transport to start and stop.

EOF End of file. This flag is the high-order bit in the 'TYPE' byte. If set, it indicates that this is the last record in the file. The record is otherwise normal, and may contain data.

TYPE This 7-bit field (the 8th bit is 'EOF') indicates the type of the record. All records in a file have the same type. The data field's format is type dependent. See below for a description of file types.

SEQ This field is an 8-bit sequence counter, used to detect missing records. If a label record is present in the file, it is record #0. The first data record is record #1. If the file contains no label record, the first record is record #1. Note that the record following record #255 is record #0, but is not a label record.

COUNT This two-byte field contains a count of the number of bytes in the Data field. The high-order byte of the count appears first. Note that the count may be zero, indicating that there is no data field.

DATA This field contains the data. Its format is dependent upon the record type. Its length is set in 'count'.



CRC-16 This is a polynomial remainder check, computed byte-wise upon the entire record (starting with the EOF/TYPE byte) from $(X + 1) * (X \uparrow 5 + X + 1)$

This checksum provides nearly flawless error detection.

ERROR	<u>DETECTION RATE</u>
Cinals hit same	1000/
Single bit error	100%
Double bit errors	100%
An odd number of bits in error	100%
An error burst <17 bits long	100%
An error burst $>= 17$ bits	99.997%

Label Record Format

Some file types require a label record to be present, and some require that no label record be present. A label record is detected by its record number of 0. Except for the contents of the data field, a label record has the same format as the other records in the file. The data field consists of a string of 7-bit ASCII characters which comprise the file's label. The 8th bit should be 0 for all characters.

System Data Formats

The following section describes the data formats associated with the various file types. There are currently three file types:

- 1. Memory Image.
- 2. BASIC programs.
- 3. Compressed Text.

MEMORY IMAGE (Type = 001)

The file type 'memory image' is used when you dump or load programs from H8 memory. This file type has no label record, the first record in the file is #1. The file may consist of one or more records. The format of the data field is:

ENTRY ADDR Program bytes

Where ENTRY = the program's entry point address, and ADDR = the address to start loading this group of program bytes. If there are multiple records in this data file, the 'entry' portion of each record should be identical.



NOTE: The COUNT field in the record header does not include the 4 bytes for ENTRY and ADDR. Thus, an empty record of this type has a zero COUNT field, but still contains the ENTRY and ADDR in the data field. Note that the high-order byte comes first for the ENTRY and ADDR fields.

BASIC PROGRAM (Type = 002)

This file type is used by BASIC when you load and dump programs. The file always has a label record (#0), and always has only one data record, #1. The data field contains the BASIC program in a special internal format. This file type can not be processed by the text editor.

COMPRESSED TEXT (Type = 003)

This file type is used by TED-8 and HASL-8 for source statements. It always has a label record (record #0) and has one or more data records. The data field in each record should not exceed 512 bytes. Lines should not be split between records. Each line is compressed according to the following format:

- 1. All characters are 7-bit ASCII, with the parity bit zero.
- 2. The carriage return and line-feed characters are not used. The end of line is indicated by a 000 byte.
- 3. Strings of spaces are represented by the value $200_8 + N$, where 'N' is the number of spaces in the series. Thus, a single blank is encoded as 201_8 ; ten blanks are encoded as 212_8 .
- 4. The maximum line length is 127 characters.

Reading the Displays

When the H8 computer is reading or writing data on a tape transport, the front panel displays are continually displaying data about the tape operation. Data about tape operation is displayed into two areas:

ADDRESS LEDs — Display the number of bytes left in the record when the transport is reading or writing data. The Address LEDs do not display any information during the inter-record gap. The address LEDs display the actual address being loaded when a memory image load or dump is executed. During a memory image operation, the Data LEDs display the data being entered into or read from memory.



DATA LEDs — Display the type of data and the record number. The right-hand-most data LED displays the type of data being read or written. This information is displayed as:

DISPLAY	DATA/TYPE
1.	Memory Image
2.	A BASIC program.
3.	Compressed text.

The two left hand LEDs display the octal record count within the particular file. As noted earlier, a file may contain one or more records. When 178 records are exceeded, the record count in the two left hand data LEDs starts over at 00. When the last record is read, the extreme left hand data LED displays a two or a three if the record count is between 108 and 178. Using this information, you can readily observe the type of data being handled by the H8. Figure 0-1 shows how these displays are used.

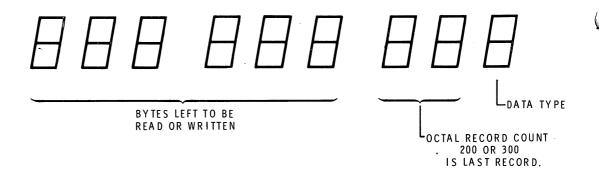


Figure 0-1

For	exam	ple
-----	------	-----



indicates the first and last data record of compressed text. NOTE: This could be the 1st or the 17_8 th data record.



indicates first data record of a memory image file.



indicates a label record of a compressed text file. Note, the label record is record number 0.



indicates the third and last record of a compressed text file.



indicates the last record of a memory image file.



USING THE MAGNETIC TAPE SYSTEM

The Heath H8-5 serial card supports two different configurations of magnetic tape recorders. You may use a single tape recorder or a dual tape recorder. The most versatile system operation is obtained by using two tape recorders with independent control. However, you can achieve a perfectly workable and somewhat less expensive system with the single recorder. Dual recorders are preferable when you are assembling long programs, as it is necessary to read a few records of the source program and then assemble this source material, generating the appropriate binary output before reading additional source records. If you use a single recorder, you must change cassettes frequently.

Recorder Operating Hints

Observe the following guidelines carefully to get the maximum operating efficiency from your H8 Cassette Recorder system.

- 1. Use only Heath approved cassette tape recorders. Although there are a variety of very good tape recorders on the market, only those models tested and approved by the Heath Company will assure successful operation.
- 2. Use a high-quality cassette recording tape. Once again, the Heath approved tape is required to assure success.
- 3. Be sure the tape is off the leader before recording a file. Frequently cassettes have excessively long leaders. Therefore, the initial portion of a file placed on tape may lose the synchronizing characters, and be lost.
- 4. Keep the tape and the recorder clean. Dirty tapes tend to cause drop outs which cause tape errors.
- 5. Label your tapes. There is nothing more frustrating than a good program written onto a tape that is unlabeled and therefore not recoverable later.

USING THE PAPER TAPE SYSTEM

The H10 Paper Tape Reader/Punch is two independent devices. It is both a paper tape punch and a paper tape reader. When it is used with the H8-2 Parallel I/O Interface, operation of the H10 becomes identical to operation with dual cassette recorders.



Paper Tape Operating Hints

The following hints will help insure you of maximum effectiveness with your paper tape operating system.

- 1. Use oiled paper tape. Oiled paper tape helps keep the punch system operating smoothly and reduces punch wear.
- 2. Keep the punch well adjusted. After the first few hours of operation, carefully observe the punch mechanism to be sure it is still properly aligned.
- 3. Label all your tapes. Just like magnetic tapes, unlabeled paper tapes are virtually useless.
- 4. Keep the chad cleaned up. Excessive loose chad tends to cause reliability problems.

PRODUCT INSTALLATION

Use the following procedure as a guideline when you install your magnetic or paper tape operating system. Remember, a major part of your H8 System purchase is the software. The installation of the software should be treated with as much care and thought as the installation of the hardware products. Without the software, the hardware is of little or no value.

Creating a Configured Tape

The Heath H8 software is supplied on distribution tapes. These tapes enable you to configure the software for your own particular needs. Use the following procedure to create a configured tape from the software distribution tape.

- A. Load tape in reader.
- B. Ready tape transport.
- C. Press LOAD on H8 front panel.
- D. Wait for a single beep indicating successful load.
- E. Press GO on H8 front panel.*

*NOTE: Any software patches received with the product should be entered before you execute their step. Once entered, they become a permanent part of the configured tape. (See Page 0-22.)

G. Configure the software product as desired, answering each of the following questions. Prompt each question by typing its first character on the console terminal keyboard. Simply type a return or do not prompt the question if you wish to leave them as distributed. The questions are:

AUTO NEWLINE (Y/N)?

A yes (Y) response to this question directs the product to generate a new line each time the print head (or cursor) moves out of the last column of the console terminal. This function is distributed preset to Y.

BKSP = 00008/

The backspace character is normally a control H (00008 decimal). When used with the video terminal or other backspacing devices, the control H generates a true backspace. The backspace character may be changed to other ASCII printing or nonprinting characters (See Appendix D) such as a backslash, if a non-backspacing terminal is used. This new character will be considered a true backspace by the software.

CONSOLE LENGTH = 00080/

The console length is initially set at 80 (decimal) characters, which is the normal width of a video terminal. This may be changed to other common values such as 132 characters for a wide printing terminal, or to 72 characters for a teleprinter.

NOTE: The maximum number of characters per line is a function of each software product. See the individual sections to determine the maximum permissible characters per line.

HIGH MEMORY = XXXXX/

When the software product is initially started, the limit of available high memory is determined. All products start at 040 100 (offset octal). If you wish not to use a certain portion of high memory, a new high memory limit (decimal count) should be typed in. If the upper memory limit is set too low, the new limit will be refused (the terminal bell will sound).

LOWER CASE (Y/N)?

A Yes (Y) response to LOWER CASE configures the software product to input lower case letters and output lower case letters. An N (no) in response to the question configures the product to work with upper case only terminals. This function is distributed preset to N.



PAD = 4/

The pad characters (nulls) are inserted following a carriage return. The pad characters are sent at this time to allow the print head time to return to the left-hand margin. For video terminals, and most teleprinters, the number of pad characters may be changed to zero. If you do not know how many pad characters are required for your terminal, initially try zero. If you appear to to be overtyping (or missing characters) at the beginning of lines, increase the pad count until the overtyping stops. You may enter up to a maximum of 9 pad characters.

RUB OUT = 00127/

The rub out character is set as 127 (decimal). If you desire to use a special rub out character, you may change it by entering in a new decimal number identifying a different ASCII character. See Appendix D.

SAVE?

A yes (Y) in response to this question directs the software product to generate a memory image of the configured product. This memory image of the configured product should be the tape you use regularly to load your program. This will avoid your having to configure the product on a regular basis. Before executing the save command, be sure the tape transport at the dump output is ready.

To use the product directly from the distribution tape, type the return key at any time rather than typing a key which prompts a question.

NOTE: It is very important that you immediately configure products as you will use them, and then place your original software distribution tape in an appropriate place for safe keeping. Use the above procedure any time you wish to configure the product.

Loading From a Configured Tape

Loading from a configured tape is a very simple procedure. It is the recommended way to normally load the software. The procedure is:

- 1. Load tape in the tape transport.
- 2. Ready the tape transport.
- 3. Press LOAD on the H8 front panel.
- 4. Wait for a single beep, indicating a successful load.
- 5. Press GO key on the H8 front panel.
- The console terminal will respond with the product description and its prompt character. The product is now ready to use in a preconfigured form.



Copying an Existing Memory Tape

Use the following procedure to copy a memory image tape. Be sure to use this procedure. Memory image tapes should not be copied on an audio-to-audio basis. An audio-to-audio copy may not work. To copy a tape:

- 1. Load the source tape in the tape transport.
- 2. Ready the tape transport.
- 3. Press LOAD on the H8 front panel.
- 4. Wait for a single beep, indicating a successful load.
- 5. Load a blank tape into the dump tape transport.
- 6. Ready the dump tape transport.
- 7. Press DUMP on the H8 front panel.
- 8. Wait for a single beep, indicating a completed dump.
- 9. Repeat steps 7 and 8 to produce a second dump, creating a double copy.

The product is now copied and ready to be used. IMPORTANT: Make at least one double copy of the distribution tapes you received with the H8 as a protection against accidental tape damage. Once the tape is copied, if magnetic tape is being used, the read-only plugs should be knocked out of the back of the cassette.

Installing a Patch

To implement a patch supplied with the distribution software tapes or those in Appendix A, load the distribution tape following steps A - P in "Creating a Configured Tape" (Page 0-19). Alter the memory contents at the locations shown in the desired patch(s), inserting the new data given in the patch. Refer to Page 1-10 "Displaying and Altering Memory Locations" for the appropriate procedure to modify a memory location. For example, to use Option Patch #1 on BENTON HARBOR BASIC:

- 1. Load the distribution tape through step D.
- 2. Change the contents of location 041 010 to 316 and the contents of location 064 077 to 001. This completes step D.
- Finish the configuration continuing with step E. BENTON HARBOR BASIC is now patched to 5.01.00.I and will supply two stop bits on each transmitted ASCII character.



Using an ASR Console

The following procedure allows you to use an ASR console as the main load/dump port, as well as the console terminal with an H8 system. An example of such an ASR (automatic send/receive) console would be the Teletype Corporation Model 33 Teleprinter. Perform the initial load by first setting the port interchange switch to the port interchange position on your H8-5 Serial I/O card. The tapes are then read in, in accordance with the procedure outlined under "Creating a Configured Tape." Once the tapes are read in, PAM-8 should be used to patch the software to the ASR console terminal configuration.

The appropriate patches for the ASR console terminal configurations of the software products are found in Appendix A of this section. Once the patches are accomplished, the GO key may be pressed and the normal configuration procedure takes place. Leave the Port Interchange switch in the Port Interchange position as long as the tape handler on the console terminal is used as the main load dump terminal.

Using a 110-Baud Console Terminal

If you use a 110-baud console terminal such as a teleprinter, one extra stop bit must be added to the ASCII characters sent by the H8. This change is made at configuration time and should be done any time you use a 110-baud console, regardless of whether or not you use the terminal as an ASR console (such as a teleprinter with a paper tape reader/punch) or simply as the console terminal. The patch is listed in Appendix A.

Console Interface

Appendix B contains a listing of the H8 Console Driver, a software module included in all H8 software packages which utilize a console terminal. A console driver is a general-purpose software package, providing you with such special characters as Control A, Control B, Control C, Control D, Control O, Control P, Control S, and Control Q.

The characters Control-A, -B, -C, and -D are available for use within the program. For example, most Heath programs use Control-C as a general purpose cancel. The other characters are permanently assigned in the console driver, and therefore in all Heath software products using the console driver. These characters are assigned as follows:

CONTROL-O

Control-O toggles the output discard flag. When the output discard flag is set, output to the console terminal is stopped, but program execution continues. Typing Control-O once sets the discard flag. Typing the Control-O again resets the output flag, permitting output to the console terminal to resume.

CONTROL-P

Control-P resets the output discard flag. Typing Control-P insures the output discard flag is not set. NOTE: Control-O toggles the flag, but Control-P only resets it.

CONTROL-S

Control-S sets the suspend output flag.

CONTROL-Q

Control-Q resets the suspend output flag.

The above control characters are not echoed to the console terminal when they are typed as is a normal character. NOTE: Many Heath Software products use Control-H, Control-I and Rubout. These characters are not used by the console driver but are passed directly through to the program for individual processing. They are also echoed to the console terminal.

CONSOLE DRIVER

The console driver also provides all capabilities for communicating with the console terminal and a tape transport at the load/ dump ports. If you, the user, develop any of your own software packages, we recommend that you incorporate this console driver rather than attempting to develop your own console driver.

The use of the control characters is explained on Page 3 of the console driver listing and in the individual software product reference sections.

The console driver also provides two front panel entry points. These are listed on Page 5 of the console driver listing. They are:



PROGRAM COUNTER*	ENTRY TYPE
040 100	Program Reset entry point. All text buffers, etc., are effectively cleared and the product is restarted. This is known as a "cold" or "hard" start.
040 103	Program restart entry point. Product is restarted with text buffers, etc., intact. This is known as a "warm" or "soft" start.

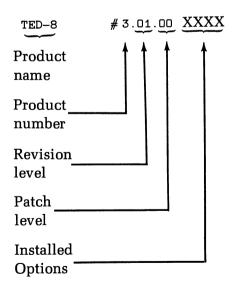
*NOTE: Set the value of the program counter using the front panel monitor. Then press GO.

Reporting Software Problems

Every effort has been made to keep the Heath H8 Software products free of defects. Should you suspect that a Heath H8 system software product may be defective, review the following procedure before contacting Heath Company.

- 1. Attempt to reload all software to be sure it has not been damaged.
- 2. Reconfigure the product from the distribution tape in an attempt to duplicate the problem.
- 3. If the problem persists, document the apparent product problem and the software version code. NOTE: It is extremely important that you know your exact software version code when you contact the Heath Company about your software product.

For example:



NOTE: Heath Company can not consult on user developed programs or modified versions of the Heath Software products.

APPENDIX A

This appendix is a listing of the patches required to use an ASR console terminal as both the load/dump port and as the console terminal (option patch #2). A patch to convert the output to two-stop bits is also included (option patch #1).

	both their console terminal port, such as a Teletsre ches are witch must be set to interchanse.	VERIFY command from BUG-8. It will no lonser no lonser no lonser no lonser no lonser affective cuning the presentions.	051 303 036 041	
OFILUN FAICH #₹	USE: This Retch is used for systems which have both their con and load/dumr device attached to the same rort, such as model 33 ASR. For those systems, these ratches are required. Note that the port interchanse switch must be	NOTES: These Patches remove the VERIFY command from BUG-8. It will no long to the valid. Also, CLL-C is no longer effective curins take presentions. The proper procedure for aborting a load or dume is to return to the front Fanel moniton. (Via.the front Panel Report Poctor to the warm start address: 040103, and press 'GD',	040107 370 040112 370 040112 371 040123 371 040126 371 041004 072 041004 072 041004 072 041004 072 041004 072 041004 072 041004 072 040051 251 046 040051 251 046 040051 051 041 303 360 052 051171 057 053 053027 055 052 315 036 041 303 305 050 315 366 0	

	TED - 8 03.01.xx.
	OPTION PATCH #2 ASR CONSOLE
USE	T.D.C.
NOTES	These ratches remove the VERIFY command for yalid. CNTRL-C.10. 10. 10. 10. 10. 10. 10. 10. 10. 10.
040107 040112 040115 040115	
040123 040126 041004 041170 042036 046267	
046274 045324 047146 047158 053044 053044 055101	324 315 357 315 311 336
057143	003 001

MATERIA Libra Land Libra Land Libra Land L	HASL - 8 04.01.xx.
Distance Active Companies Distance	ATCH
Dis location of the state of	CONSOLE
7. 270 2. 270 2. 271 2. 271	This ratch is used for systems which have both their con and load/dumr device attached to the same rort, such as model 33 ASR. For those systems, these ratches are required. Note that the port interchange switch must be
2.30 2.31	None.
9.70 9.71 9.71 9.71 9.71 10.40 6.65 10.40 6.65	
271 271 271 272 272 202 203 203 203 203 203 203 203 203 20	
357 972 972 104 646 104 646 105 105 105 105 105 105 105 105	
001 001 001 001 001 001 001 001	
104 0.66 303 036 041 315 015 060 303 036 041 001	07.0
	064 061 104 060 303 036 041 31 303 036 041

MORE: This Patch: is used for systems which have both their console terminal and lead/dump device statched to the same roit; such set is leighted white their console terminal and lead/dump device statched to the same roit; such set is reculred. More that their console terminal model, 33 ASR. For those systems, these satches are roulred. More that prove recorded to the set is not longer statched for its continual to the rooper rooper recorded to the very many large. CIL-C is not longer effective during the rooper rooper rooper rooper recorded to the rooper	
USE: This Patch is used for sand load/dump device attended 133 ASR. For those Note that the Port Internated 133 ASR. For those Note that the Port Internated 134 ASR. For those Note that the Port Internated 135 ASR. For those Note that the Proper Procedure for The Warm start address: the Warm Start address: 040115 371 040123 371	
USE: This patch is used for stand load/dump device attained and load/dump device attained by the that the port inter NoTES: These patches remove the be valid, Also, CIL-C is The proper procedure for the front panel monitor the front panel monitor the warm start address: the warm start address: the warm start address: 240012 371 04012 371 04012 371 04012 371 04012 371 04012 371 04012 371 04012 371 04012 371 04012 371 04010 371 04010 371 04010 371 04010 371 04010 371 04010 371 04010 371 04010 371 04010 371 04010 371 04010 371 071 071 071 071 071 071 071 071 071 0	
NOTES: These Patches remove the be valid. Also, CTL—C is the Proper Also, CTL—C is the Front Part Address: The Warm start address: the front Part address: the front Part address: the Address: the Warm start address: the Addres	
The Proper Procedure for the front Panel monitor the warm start address: the warm start address: the warm start address: the warm start address: 040112 370 040112 371 040124 371 040126 371 04104 072 04104 072 04104 072 04104 072 04104 067 341 303 057071 033 041 303 071076 017 003 315 033 034 041 303 071076 017 003 315 035 041 303 075202 001 325 002 303 036 041 075202 001) SET:
040107 370 040112 370 040112 371 040123 371 040126 371 040126 371 041004 072 041004 072 044354 303 252 046 046252 265 341 310 315 036 04 057071 036 041 057171 036 041 067164 303 263 046 070060 106 071 071076 017 003 315 036 041 30 071076 001 325 002 303 036 04	
040107 370 040115 371 040123 371 040124 371 040126 371 041014 072 044356 303 252 046 044356 303 252 046 046252 245 341 310 315 036 04 046252 046 302 160 067 341 30 057071 036 041 057071 036 041 070060 106 071 07106 315 325 002 303 036 04	
040126 371 041004 072 041014 072 044556 303 252 046 046252 265 341 310 315 036 04 046252 044 302 160 067 341 30 057071 036 041 057071 036 041 057074 017 003 315 036 041 36 071076 017 003 315 036 041 36 071076 017 003 315 036 041 36 071076 017 003 315 036 041 36 075202 001	
046262 044 302 160 067 341 30 057071 003 057171 036 041 067166 303 263 046 070034 311 070000 106 071 071076 017 003 315 036 041 36 071106 315 325 002 303 036 04 075202 001	
070034 311 070060 106 071 071076 017 003 315 036 041 36 071106 315 325 002 303 036 04 075202 001	

	EXTENDED BENTON HARBOR BASIC 10.01.xx.
	OPTION PATCH #2 ASR CONSOLE
ușE:	4. E K. D
NOTES:	
040107 040112 040115	
040120 040123 040126 041004	371 371 371 072
041014 044307 04307 050324 050326	
064726 065110 100157	
101142 101142 102303 102313 106173	

BUG - 8 02.01.xx.	
OPTION PATCH #1	
2 STOP BITS	
USE: This patch is inserted for systems which use a terminal devrees requiring 2 stop bits. This should not be used for devices can run with only one stop bit.	erminal device for devices which
NOTES: None.	
 041010 316 056357 001	
TED - 8 03.01.xx.	
OPTION PATCH #1	
2 STOP BITS	
USE: This patch is inserted for systems which use a terminal devices in requiring 2 stop bits. This should not be used for devices in the orly one stop bit.	terminal device for devices which
NOTES: None.	
041010 316 064077 001	
	- The second sec

WOTES FOR STOP ALTON CARLOT AND ALTON CA	HASL - 8 04.01.xx	
WITTER Note: The part of the system which refer devices which century is not the used for devices which is not the used	2	#
ONTOTES Notice		or systems which use a terminal device his should not be used for devices which op bit.
211 000 100 100 100 100 100 100 100 100	NOTES: None.	
	:	

BENTON HARBOR BASIC 05.01.xx.
OPTION PATCH
2 SIUP BIIS
USE: This Patch is inserted for systems which use a terminal device requiring 2 stop bits. This should not be used for devices which can run with only one stop bit.
:
041010 316 075201 001
-
:

		:
	10.01.xx.	:
	OPTION PATCH #1	:
	2 STOP BITS	:
J	USE: This Patch is inserted for systems which use a terminal device requiring 2 stop bits. This should not be used for devices which can run with only one stop bit.	
Z	NOTES:	; ;
		:
J #1	041010 316 106173 001	:
' :		:
		: : :
		:
		:
		:
		:
		:
		:
		:
		:
		:
		:
		:
		:
		:
		:
:		<u>:</u>
		:
		:
		:
		:



APPENDIX B

Console Driver Listing



					<u> </u>
HEATH H8 CONSOLE DRIVER INTERRUPT-TIME CONSOLE DRIVER	IVER.			HEATH X8ASM V1.1 06/21/77 17:50:05 01-AFR-77 PAGE 1	
	U, W, 4, N, 40, V, 00 * * * * * * * * * *	HEATH H	HEATH HB SOFTWARE CO JGL, 01/01/77, FDR.* COPYRIGHT 1977 BY HE	HEATH HB SOFTWARE CONSOLE DRIVER. JGL, 01/01/77, FDR.*HEATH* COMPANY. COPYRIGHT 1977 BY HEATH COMPANY, BENTON HARBOR, MI.	
	* * * * * * * * * * * * * * * * * * *	THE FOLCONSOLE ALL SOF	THE FOLLOWING CONTAIN CONSOLE DRIVER, THESHALL SOFTWARE FRODUCTS ALL PROGRAMS WISHING SHOULD USE THESE ROU	THE FOLLOWING CONTAINS THE TEXT FOR THE HEATH H8 CONSOLE DRIVER, THESE EXACT ROUTINES ARE USED IN ALL SOFTWARE FRODUCTS. ALL PROGRAMS WISHING TO COMMUNICATE WITH THE CONSOLE SHOULD USE THESE ROUTINES.	
040.100	22.00 20.00 20.00 **	ORG ASSEMBL	ORG. 40100A. ASSEMBLY CONSTANTS.	START OF USER RAM	
000.007 040.037 111.111 222.222 333.333	23 BELL 24 ULVEC 25 START 26 RESTART 27 CONFIG 29	EQU EQU EQU F EQU EQU XTEXT	0070 040037A 111111A 222222A 333333A UB251	ASCII BELL HBMTR USER INTERRUFT VECTORS DUMMY START LABEL DUMMY RESTART LABEL DUMMY CONFIGURE LABEL	



1, 2, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10	1.250	1 1 2 10 10 10 10 10 10 10 10 10 10 10 10 10	33. ** 34. ** 35. ** 35. ** 35. ** 35. ** 36. ** 37. UNIT BE END 100000009 1 12 FOR BITS 38. UNIT BE END 100000009 2 5TOF BITS 39. VILLA END 010000009 2 5TOF BITS 40. VILLA END 000000009 5 5TOF BITS 42. VILLA END 000000009 5 5TOF PAITY 43. VILLA END 000000009 7 5TOF PAITY 44. VILLA END 000000009 7 5TOF PAITY 45. VILLA END 000000009 7 5TOF PAITY 46. VILLA END 000001009 7 5TOF PAITY 46. VILLA END 000000109 7 5TOF PAITY 46. VILLA END 000000109 7 5TOF PAITY 46. VILLA END 000000009 7 5TOF PAITY 47. VILLA END 0000000009 7 5TOF PA	23.	00000000000000000000000000000000000000	* * * * * * * * * * * * * * * * * * *	. 101	
355 ** HÖDE INSTRUCTION CONTROL BITS. 36X UMI.1B EQU 01000000B 1 157DF BITS 39X UMI.1B EQU 1000000B 2 51DF BITS 39X UMI.1B EQU 00100000B EVEN PARITY 40X UMI.1C EQU 00100000B EVEN PARITY 42X UMI.1C EQU 0000100B 5 BIT CHARACTERS 44X UMI.1R EQU 0000100B 6 BIT CHARACTERS 45X UMI.1R EQU 0000100B 7 BIT CHARACTERS 45X UMI.1R EQU 0000100B 7 BIT CHARACTERS 45X UMI.1A EQU 0000100B 6 BIT CHARACTERS 45X UMI.1A EQU 0000100B 7 BIT CHARACTERS 50X ** CDHHAND INSTRUCTION BITS. 51X UCI.RE EQU 0000001B CLOCK X 64 53X UCI.RE EQU 0000001B ERROR RESET 53X UCI.RE EQU 0001000B INTERNAL RESET 53X UCI.RE EQU 0001000B ERROR RESET 55X UCI.RE EQU 0001000B ERROR RESET 55X UCI.RE EQU 0001000B ERROR RESET 55X UCI.RE EQU 0001000B FRANTY ERROR FROM 64X USR. FE EQU 0001000B FRANTY ERROR 65X USR. FE EQU 0001000B FRANTY ERROR 65X USR. FE EQU 0001000B FRANTY ERROR 65X USR. FE EQU 00000100B FRANTY ERROR 65X USR. FE EQU 00000010B FRANTY FRANTY FREEDY 65X USR. FE EQU 00000010B FRANTY FRANSWITTER READY 65X USR. FE EQU 00000010B FRANSWITTER	1	25X ## HODE INSTRUCTION CONTROL BITS. 34X UNIT 18 EQU 010000000 1 1 2 FOP BITS 35X UNIT 18 EQU 010000000 1 1 1 2 FOP BITS 35X UNIT 18 EQU 010000000 1 1 1 2 FOP BITS 35X UNIT 18 EQU 01000000	1	15.5. ## 'NOWE INSTRUCTION CONTROL BITS. 15.6. MAINTER END 010000000 1 17.05 PBT 15. 15.6. MAINTER END 010000000 1 17.05 PBT 17. 15.6. MAINTER END 01000000 1 17.05 PBT 17. 15.6. MAINTER END 0100000 1 17.05 PBT 17. 15.6. MAINTER END 0100000 1 17.05 PBT 17. 15.6. MAINTER END 0100000 1 17.05 PBT 17. 15.6. MAINTER END 010000 1 17.05 PBT 17. 15.6. MAINTER END 01.05 PBT 17. 15.6. MAINTER END 01.05 PBT 17. 15.6. M	000 000 000 000 000 000 000 000 000 00	** UM11128 UM1117288 UM1117288 UM1117288 UM111728 UM11173	101	
39X UHI.1B EQU 01000000B 1 1510F BITS 39X UHI.1B EQU 11000000B 2 510F BITS 39X UHI.1B EQU 11000000B 2 510F BITS 40X UHI.FE EQU 0001000B 5 BIT CHARACTERS 43X UHI.LE EQU 00001000B 7 BIT CHARACTERS 43X UHI.1A EQU 00001000B 7 BIT CHARACTERS 44X UHI.1A EQU 0000100B 8 BIT CHARACTERS 45X UHI.1A EQU 0000100B CLOCK X 14 49X UHI.1AX EQU 0000100B CLOCK X 44 50X X** COHMAND INSTRUCTION BITS, 51X UCI.RE EQU 0000000B ERROR ESET 52X UCI.RE EQU 0001000B ERROR ESET 53X UCI.RE EQU 0001000B ERROR ESET 53X UCI.RE EQU 0001000B ERROR ESET 54X UCI.RE EQU 0001000B ERROR ESET 55X UCI.RE EQU 0001000B ERROR ENGRE 55X UCI.RE EQU 0001000B ERROR ENGRE 55X UCI.RE EQU 0001000B FRAMING ERROR 64X USR.FEE EQU 0001000B FRAMING ERROR 64X USR.FEE EQU 0001000B FRAMING ERROR 64X USR.TXE EQU 0001000B FRAMING ERROR 65X USR.TXE EQU 0000100B FRAMING ERROR 65X USR.TXE EQU 0000010B FRAMINTER ERPOY 65X USR.TXE EQU 0000001B FRAMINTER ERPOY 65X USR.TXE EQU 0000010B FRAMINTER ERPOY 65X USR.TXE EQU 0000010B FRAMINTER ERPOY 65X USR.TXE EQU 0000010B FRAMINTER ERPOY 65X USR.TXE EQU 000001B FRAMINTER ERPOY 65X USR.TXE EQU 0000010B FRAMINTER ERPOY 65X USR.TXE EQU 00000000B FRAMINTER ERPOY 65X USR.TX	38X VIII.18 E EUL 100000005 1 172 FIDE BITS 40X VIII.18 EUU 110000000 1 172 FIDE BITS 40X VIII.18 EUU 010000000 1 172 FIDE BITS 40X VIII.18 EUU 010000000 1 172 FIDE BITS 41X VIII.18 EUU 000010000 0 1 17 CHARACTERS 41X VIII.18 EUU 00001000 0 1 17 CHARACTERS 41X VIII.18 EUU 00001000 0 1 17 CHARACTERS 41X VIII.18 EUU 00000100 0 1 17 CHARACTERS 41X VIII.18 EUU 00000010 0 1 17 CHARACTERS 41X VIII.18 EUU 00000010 0 1 17 CHARACTERS 41X VIII.18 EUU 00000010 0 1 17 CHARACTERS 51X VIII.18 EUU 01000000 0 1 17 CHARACTERS 52X VIII.18 EUU 00000000 0 1 17 CHARACTERS 52X VIII.18 EUU 000000000 0 1 17 CHARACTERS 52X VIII.18 EUU 0000000000 0 1 17 CHARACTERS 52X VIII.18 EUU 0000000000 0 1 17 CHARACTERS 52X VIII.18 EUU 0000000000 0 1 17 CHARACTERS 52X VIII.18 EUU 0000000000000000000000000000000000	3	33 WH I IF EQU 010000008 1 17 FF ETP ETP 33 W UH I LB EQU 1100000008 1 17 FF ETP 40 W UH I FE EQU 000000008 2 10 F ETT 41 W UH I LB EQU 000000008 2 11 C GARGTERS 42 W UH I LB EQU 000000009 3 11 C GARGTERS 43 W UH I LB EQU 000000009 3 11 C GARGTERS 44 W UH I LB EQU 000000009 3 11 C GARGTERS 45 W UH I LB EQU 00000009 2 11 C GARGTERS 45 W UH I LB EQU 00000009 2 11 C GARGTERS 46 W UH I LB EQU 00000009 2 11 C GARGTERS 47 W UH I LB EQU 00000009 2 11 C GARGTERS 48 W UH I LB EQU 00000009 C LLORY X 44 48 W UH I LB EQU 00000009 C LLORY X 44 48 W UH I LB EQU 00000009 ERETOR RESET 50 W UH I LB EQU 00000009 ERETOR RESET 51 W UH I LB EQU 00000009 ERETOR RESET 52 W UH I LB EQU 00000009 FRAMING EREOR 53 W UH I LB EQU 00000009 FRAMING EREOR 54 W UH I LB EQU 00000009 FRAMING EREOR 55 W UH I LB EQU 000000010 FRAMING EREOR 55 W UH I LB EQU 000000010 FRAMING EREOR 55 W UH I TREE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 000000010 FRAMING EREOR 55 W UB I TRE EQU 0000000000 FRAMING EREOR 55 W UB I TRE EQU 0000000000 FRAMING EREOR 55 W UB I TRE E	33X UH : 18 EQU	000.100 000.200 000.340 000.000 000.000 000.001 000.001 000.003 000.003 000.100 000.004 000.004	MI 118 WM 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1		VTROL BITS.
39x UNI.2B EQU 11000000B 2 STOP BITS 40x UNI.FE EQU 00010000B 6VEN PARITY 41x UNI.FE EQU 0001000B 5 BI PARITY 42x UNI.L5 EQU 00000000B 5 BI CHARACTERS 43x UNI.L5 EQU 00000100B 7 BIT CHARACTERS 44x UNI.18 EQU 00001100B 8 BIT CHARACTERS 45x UNI.18 EQU 00001100B 8 BIT CHARACTERS 45x UNI.18 EQU 0000110B CLOCK X 14 49x UNI.64x EQU 0000001B CLOCK X 44 49x UNI.64x EQU 0000001B CLOCK X 64 55x UCI.RE EQU 0000000B REABER-ON CUNIRUL P 53x UCI.RE EQU 00010000B RECEIVE ENABLE 53x UCI.RE EQU 00010000B RECEIVE ENABLE 55x UCI.RE EQU 00010000B RECEIVE ENABLE 55x UCI.RE EQU 00010000B RECEIVE ENABLE 55x UCI.RE EQU 00010000B FRANSMIT ERED 55x UCI.RE EQU 00010000B FRANSMIT ERED 55x UCI.RE EQU 00010000B FRANSMIT ERED 55x USR.RE EQU 00010000B FRANSMIT ERED 55x USR.RE EQU 00010000B FRANSMIT ERED 55x USR.RE EQU 00000010B FRANSMIT ERED 55x USR.RES EQU 00000010B FRANSMITTER READY 55x USR.TXR EQU 00000010B TRANSMITTER READY 55x USR.TXR EQU 000000010B TRANSMITTER READY 55x USR.TXR EQU 0000000000B TRANSMITTER READY 55x USR.TXR EQU 000000000B TRANSMITTER READY 55x USR.TXR EQU 000000000B TRANSMITTER READY 55x USR.TXR EQU 00000000B TRANSMITTER READY 55x USR.TXR EQU 0000000B TRANSMITTER READY 55x USR.TXR EQU 00000000B TRANSMITTER READY 55x USR.TXR EQU 0000000B TRANSMITTER READY 55x USR.TXR EQU 0000000B TRANSMITTER READY 55x USR.TXR EQU 00000000B TRANSMITTER READY 55x USR.TXR EQU 0000000B TRANSMITTER READY	40X UMI 28 EQU 01000008 2 517 PRITY 40X UMI 126 EQU 00000008 19 FARITY 41X UMI 126 EQU 00000009 5 BIT CHARACTERS 44X UMI 12. EQU 000010008 6 BIT CHARACTERS 44X UMI 12. EQU 000010008 7 BIT CHARACTERS 45X UMI 12. EQU 000010008 1 BIT CHARACTERS 45X UMI 12. EQU 000000118 CLOCK X 14 49X UMI 12. EQU 000000118 CLOCK X 14 49X UMI 12. EQU 000000118 CLOCK X 14 49X UMI 12. EQU 000000118 CLOCK X 14 55X UCI 16 EQU 000000018 REGISTOR RESET 55X UCI 16 EQU 001000008 REGISTOR FROM RESET 55X UCI 16 EQU 0000000108 REGISTOR FROM REGISTOR F	939, WILL PE EQU 01000000 2 STOP BITS 40X WILL PE EQU 010000000 1 STOP BITS 41X WILL PE EQU 00000000 1 STIT CHARACTERS 43X WILL S EQU 00000100	93 WILLE END 110000008 2 STOP BITS 40 WILLE END 010000008 1 STOP BITS 42 WILL FE END 000000008 1 STOP PRITY 43 WILL SEND 000001008 2 STOP BITS 43 WILL SEND 00000100	43X UNIT-PR FED 001000000 2 STOR PRITY 43X UNIT-PR FED 000000000 5 STOR PRITY 44X UNIT-PR FED 000000000 5 STOR PR FED 00000000 5 STOR PR FED 00000000 5 STOR PR FED 000000000 5 STOR PR FED 00000000 5 STOR P S S S S S S S S S S S S S S S S S S	00000000000000000000000000000000000000	######################################		: :
43X UMI.PE ERU 0010000B EVE PARITY 41X UMI.PE ERU 0000000B EVE PARITY 42X UMI.LS ERU 00000100B 5 BIT CHARACTERS 43X UMI.LS ERU 00001100B 7 BIT CHARACTERS 45X UMI.LX ERU 00001100B 8 BIT CHARACTERS 45X UMI.LX ERU 0000010B CLUCK X 14 49X UMI.AX ERU 0000001B CLUCK X 14 49X UMI.AX ERU 0000001B CLUCK X 64 51X UMI.AX ERU 0000001B CLUCK X 64 52X UCI.IK ERU 0000001B ERERE ERO EXPERENTE 53X UCI.RE ERU 00010000B RECELVE ENABLE 54X UCI.RE ERU 00010000B RECELVE ENABLE 55X UCI.RE ERU 00000010B RECELVE ENABLE 55X UCI.RE ERU 00000010B RECELVE EROR 63X USR.RE ERU 00010000B PURRUN ERROR 63X USR.RE ERU 00000100B RECELVER REGIN 64X USR.RX ERU 00000100B RECELVER REGIN 64X USR.RX ERU 00000100B RECELVER REGIN 65X USR.RX ERU 00000100B RECELVER REGIN 65X USR.RX ERU 00000100B RECELVER REGIN 65X USR.RX ERU 00000010B REARDRE TRANSMITTER REPRINTER PROPERTY CONSTRUCTOR TRANSMITTER REGIN 65X USR.RX ERU 000000010B TRANSMITTER REGIN 65X USR.RX ERU 00000010B TRANSMITTER REGIN 65X USR.RX ERU 00000010B TRANSMITTER REGIN 65X USR.RX ERU 00000010B TRANSMITTER REGIN	440X WIN THE FERU 0000000B UNERPRINT FILE FROM 0000000B S BIT CHARACTERS 42X WIN.1.5 ERU 00000000B S BIT CHARACTERS 44X WIN.1.5 ERU 00000100B S BIT CHARACTERS 44X WIN.1.1.5 ERU 0000100B S BIT CHARACTERS 44X WIN.1.1.5 ERU 0000100B S BIT CHARACTERS 44X WIN.1.1.5 ERU 0000100B CLOCK X 16 42X WIN.1.1.5 ERU 0000010B CLOCK X 16 42X WIN.1.1.5 ERU 0000010B CLOCK X 16 42X WIN.1.1.5 ERU 00000010B CLOCK X 16 42X WIN.1.1.5 ERU 00000010B CLOCK X 16 52X WIN.1.6 ERU 00010000B REAGHER FROM UNINKUL FLAG 52X WILT ER ERU 00010000B REAGHER ERROR FOR 52X WILT ER ERU 00010000B REAGHER ERROR FROM CONTROL ERROR FOR 52X WILT ERU 00000010B FRAMING ERROR 62X WISH ER ERU 000000010B FRAMING ERROR 62X WISH ER ERU 00000000B FRAMING ERROR 62X WISH ER ERU 00000000B FRAMING ERROR 62X WISH ERU 00001000B FRAMING ERROR 62X WISH ER ERU 00001000B FRAMING ERROR 62X WISH ERU 0000100B FRAMING ERROR 62X WISH ERU 0000100B FRAMING ERROR 62X WISH ERU 0000100B FRAMING ERROR 62X WISH ERU 0000010B FRAMING ERROR 62X WISH ERU 0000100B FRAMING ERROR 62X WISH ERU 0000010B FRAMING ERROR 62X WISH ERU 00000010B FRAMING ERROR 62X WISH ERU 0000010B FRAMING ERROR 62X WISH ERU 00000010B FRAMING ERROR 62X WISH ERU 0000010B FRAMING ERROR 62X WISH ERU 00000010B FRAMING ERROR 62X WISH ERU 00000000B FRAMING ERROR 62X WISH ERROR 62X	41X UMI PE EBU 00010000B USE PARITY 41X UMI LE EBU 0000000B 5 BIT CHARACTERS 42X UMI LE EBU 00000000B 7 BIT CHARACTERS 42X UMI LE EBU 00000000B 1 BIT CHARACTERS 45X UMI LEX EBU 00000000B CLUCK X 14 44X UMI LEX EBU 0000000B REGISTER RESET 52X UCI LE EBU 00100000B REGISTER FLAG 52X UCI LE EBU 00100000B REGISTER FLAG 52X UCI LE EBU 00100000B REGISTER FLAG 52X UCI LE EBU 00010000B REGISTER FLAG 52X UCI LE EBU 00010000B REGISTER FROM 52X UCI LE EBU 0001000B REGISTER FROM 52X USR TRE EBU 0001000B REGISTER FROM 52X USR TRE EBU 00000010B TRANSHITTER FROM 52X USR TRE EBU 00000010B TRANSHITTER FROM 52X USR TRE EBU 00000010B TRANSHITTER FROM 52X USR TRE EBU 000000010B TRANSHITTER FROM 52X USR TRE EBU 0000000010B TRANSHITTER FROM 52X USR TRE EBU 0000000010B TRANSHITTER FROM 52X USR TRE EBU 00000000010B TRANSHITTER FROM 52X USR TRE EBU 0000000000000000000000000000000000	41X MIT.15 ERU 000100006 USE PARTY 1 41X MIT.15 ERU 000100006 S BIT CHRAGTERS 43X MIT.12 ERU 00001000 S BIT CHRAGTERS 43X MIT.12 ERU 000011000 B BIT CHRAGTERS 43X MIT.14 ERU 000011000 B BIT CHRAGTERS 43X MIT.15 ERU 000011000 B BIT CHRAGTERS 43X MIT.15 ERU 000001100 C CLOCK X 44 43X MIT.18 ERU 000001100 CLOCK X 64 43X MIT.18 ERU 000000110 CLOCK X 64 43X MIT.18 ERU 000000110 CLOCK X 64 53X WIT.18 ERU 000000110 ERROR RESET 53X WIT.18 ERU 000000100 FRANSHITER ERPTY 53X WIT.18 ERU 000000100 TRANSHITER ERPTY 54X WIT.18 ERU 0000010000 WERRUR ERROR 54X WIT.18 ERU 000001000 TRANSHITER ERPTY 54X WIT.18 ERU 000001000 TRANSHITER ERPTY 54X WIT.18	14X UNIT PE END 000100009 1 STATE PRINTY 14X UNIT PE END 000100009 2 STATE CHARACTERS 14X UNIT PE END 00010009 2 STATE CHARACTERS 14X UNIT PE END 000100109 2 STATE CHARACTERS 14X UNIT STATE	000,000 000,000 000,001 000,001 000,001 000,001 000,003 000,004 000,002 000,002	UM11.P.P.P.E. UM11.P.P.P.E. UM11.L.8 V.M.I.L.8		:
42X UM1.L5 EQU 0000000B 5 BIT CHARACTERS 43X UM1.L6 EQU 00000100B 6 BIT CHARACTERS 45X UM1.L8 EQU 00000100B 8 BIT CHARACTERS 45X UM1.L8 EQU 0000001B CLOCK X 14 48X UM1.1X EQU 0000001B CLOCK X 14 48X UM1.64X EQU 0000001B CLOCK X 15 50X ** COMMAND INSTRUCTION BITS. 51X UCI.RE EQU 0000000B RECEIVE ENABLE 52X UCI.RE EQU 00010000B RECEIVE ENABLE 55X UCI.RE EQU 00010000B RECEIVE ENABLE 55X UCI.RE EQU 00010000B RECEIVE ENABLE 55X UCI.RE EQU 0000010B RECEIVE ENABLE 55X UCI.RE EQU 00000010B RECEIVE ENABLE 55X UCI.RE EQU 00010000B RANGE 55X UCI.RE EQU 00010000B RANGE 55X USR FE EQU 00010000B RANGE 55X USR FE EQU 00010000B RANGE 64X USR FE EQU 0000000B REROR 64X USR FE EQU 0000000B REROR 65X USR FE EQU 0000000B REROR 65X USR FER EQU 0000000B REROR 65X	42X UNILLS EQU 00000000B 5 BIT CHARACTERS 43X UNILLS EQU 00001000B 7 BIT CHARACTERS 44X UNILLS EQU 0000100B 7 BIT CHARACTERS 44X UNILLS EQU 00000100B 1 BIT CHARACTERS 44X UNILLS EQU 00000010B CLOCK X 14 48X UNILLS EQU 00000010B CLOCK X 14 50X *** COMMAND INSTRUCTION BITS 51X UCT. IR EQU 0000000B READER-ON TUNINUL HLAG 52X UCT. IR EQU 0000000B READER-ON TUNINUL HLAG 53X UCT. IR EQU 0000000B READER-ON TUNINUL HLAG 53X UCT. IR EQU 0000000B READER ESST 55X UCT. IR EQU 0000000B READER ESST 55X UCT. IR EQU 0000000B READER ENGRE 55X UCT. IR EQU 0000000B READER ENGRE 55X UCT. IR EQU 00000010B TRANSHIT ENABLE 55X UCT. IR EQU 00000010B TRANSHITTER ENTOR 55X USR. FE EQU 00000010B TRANSHITTER ENTOR 55X USR. TXR EQU 000000010B TRANSHITTER ENTOR 55X USR. TXR EQU 00000010B TRANSHITTER ENTOR 55X USR. TXR EQU 00000010B TRANSHITTER ENTOR 55X USR. TXR EQU 00000010B TRANSHITTER ENTOR 55X USR. TXR EQU 00000000000000000000000000000000000	43X MM1.LS EQU 00000000B 5 BIT CHARACTERS 44X MM1.LZ EQU 00001000B 7 BIT CHARACTERS 44X MM1.LS EQU 00001000B 7 BIT CHARACTERS 45X MM1.LS EQU 00000100B 1 BIT CHARACTERS 45X MM1.LS EQU 00000010B 1 CLOCK X 14 45X MM1.LAX EQU 00000010B 1 CLOCK X 14 45X MM1.LAX EQU 00000010B 1 MERNAL RESET 55X MCI.LR EQU 0100000B INTERNAL RESET 55X MCI.LR EQU 0000000B REARCACH UNINIOU L'LAG 55X MCI.LR EQU 0000000B REARCACH EGNILE 55X MCI.LR EQU 0000000B MARTY ENABLE 55X MCI.LR EQU 0000000B MARTY ENABLE 55X MCI.LR EQU 0000000B MARTY ERROR 55X MCI.RR EQU 0000000B MARTY ERROR 55X MCI.RR EQU 0000000B MARTY ERROR 55X MCI.RR EQU 00000000B MARTY ERROR 55X MCI.RR EQU 0000	43X UM1.LS EQU 00000000B 5 BTT CHARACTERS 44X UM1.LS EQU 00001000B 7 BTT CHARACTERS 44X UM1.LS EQU 00001000B 8 LDCR X 14 44X UM1.1X EQU 00000100B 8 LDCR X 14 44X UM1.1X EQU 0000010B CLOCK X 14 44X UM1.1X EQU 0000010B CLOCK X 14 49X UM1.1X EQU 0000010B CLOCK X 14 49X UM1.1X EQU 0000010B CLOCK X 14 50X ** CCHMAND INSTRUITS CLOCK X 64 50X ** CCHMAND INSTRUITS EQU 0000100B REGISTER END 0000000B REGISTER END 000000B REGISTER END 0000000B REGISTER END 000000B REGISTER END 0000000B REGISTER END 000000B REGISTER END 00000B REGISTER E	433, UNT.1.5 EQU. 00000000 5 S. P. CHARACTERS 443, UNT.1.5 EQU. 00000000 8 S. P. P. CHARACTERS 443, UNT.1.5 EQU. 00000000 B. P. P. P. CHARACTERS 443, UNT.1.5 EQU. 00000000 B. P. P. P. CHARACTERS 443, UNT.1.5 EQU. 00000000 B. P.	000,000 000,001 000,001 000,002 000,003 000,003 000,003 000,004 000,004 000,004	UMI.L5 UMI.L6 UMI.L7 UMI.L8 UMI.L8 VMI.17 VM	:	EVEN PAKITY USE PAKITY
43X UMI.LS EQU 0000100B 6 BIT CHARACTERS 45X UMI.LS EQU 0000100B 8 BIT CHARACTERS 45X UMI.1X EQU 000010B CLOCK X 1 45X UMI.1X EQU 0000001B CLOCK X 14 49X UMI.64X EQU 0000001B CLOCK X 14 49X UMI.64X EQU 0000001B CLOCK X 64 55X UCI.RE EQU 0010000B READER-DN UNIRUL P 53X UCI.RE EQU 0010000B READER-DN UNIRUL P 53X UCI.RE EQU 0010000B READER-DN UNIRUL P 54X UCI.RE EQU 00000010B RABELE INTERRUPTS F 55X UCI.RE EQU 00000010B RABELE INTERRUPTS F 55X UCI.RE EQU 00000010B RABELE 55X UCI.RE EQU 0000001B RABELE 55X UCI.RE EQU 00000010B RABELE 55X USR.FE EQU 0000000B RABELE 55X USR.FE EQU 000000B RABELE 55X USR.FE EQU 000000B RABELE 55X USR.FE EQU 0000000B RABELE 55X USR.FE EQU 000000B RABELE 55X USR.FE EQU 00000B RABELE 55X USR.FE EQU 000000B RABELE 55X USR.FE EQU 000000B RABELE 55X USR.FE EQU 00000B RABELE 55X USR.F	43x UM1.1.6 EUU 00000100B	43x UM1.L6 EQU 00000100B & BIT CHARACTERS 44x UM1.L2 EQU 0000100B Z BIT CHARACTERS 45x UM1.L3 EQU 00000100B Z BIT CHARACTERS 45x UM1.154 EQU 0000000B CLOCK X 14 47x UM1.164 EQU 0000000B CLOCK X 14 47x UM1.164 EQU 0000000B CLOCK X 14 55x UCT.TR EQU 0000000B READER RESET 55x UCT.RE EQU 0000000B READER PRANCE 55x UCT.RE EQU 00000000B READER PRANCE 55x UCT.RE EQU 0000000B READER PRANCE 55x UCT.RE EQU 0000000B READER PRANCE 55x UCT.RE EQU 00000000B READER PRANCE 5	44X WITLLE EQU 00000100B 6 BIT CHARACTERS 44X WITLLY EQU 0000100B 8 BIT CHARACTERS 44X WITLLY EQU 0000100B CLCRX X 14 44X WITLLY EQU 0000010B CLCRX X 14 44X WITLLY EQU 00000010B CLCRX X 14 45X WITLLY EQU 00000010B CLCRX X 14 45X WITLLY EQU 00000010B CLCRX X 14 55X WITLLY EQU 0000000B REABLE FOR WITLLY END CLCRX EXET CHARACTERS 55X WITLLY EQU 0001000B RECEIVE ENGLE 55X WITLLY E EQU 00000001B FRANTING ENOR CLCRX ENGLE 55X WITLLY E EQU 00000001B FRANTING ENOR CLCRX ENGLE 55X WITLLY E EQU 00000000B FRANTING ENOR CLCRX ENGLE 55X WITLLY E EQU 00000000B FRANTING ENOR CLCRX ENGLE 55X WITLLY EQU 00000000B FRANTING ENOR CLCRX ENGLE 55X WITLLY EQU 00000000B FRANTING ENOR CLCRX ENGLE 55X WITLLY EQU 00000000B FRANTING ENOR CLCRX ENGLY EN	433 UNIT LE EUU 00000100B 6 BIT CHARACTEES 443 UNIT LE EUU 00000100B 1 BIT CHARACTEES 443 UNIT LES EUU 00000100B 1 BIT CHARACTEES 443 UNIT LES EUU 0000010B 1 CLUCK X 1 A 443 UNIT LES EUU 0000010B 1 CLUCK X 1 A 543 UNIT LES EUU 0000000B 1 NTERNAL REST 553 UCT: RE EUU 00010000B 1 RERAIR REST 553 UCT: RE EUU 0000000B 1 RERAIR REST 554 UCT: RE EUU 0000000B 1 RERAIR REST 555 UCT: REST 555 UCT: RE EUU 0000000B 1 RERAIR REST 555 UCT: REST 555 UCT: RE EUU 0000000B 1 RERAIR REST 555 UCT: REST 55	000.004 000.003 000.003 000.003 000.003 000.003 000.000	UMI.L6 UMI.L8 UMI.L8 UMI.16X UMI.64X		5 BIT CHARACTERS
45X UMI.1X ERU 00001100B B BIT CHARACTERS 46X UMI.1X ERU 00000010B CLOCK X 14 49X UMI.64X ERU 00000010B CLOCK X 44 49X 50X ** COMMAND INSTRUCTION BITS, 51X UCI.FE ERU 0010000B FRADER-DN UNINUL FSX UCI.FE ERU 00000010B FRADER-DN TRANSMIT ENABLE 57X UCI.FE ERU 00000010B FRADER-DN FRADER FROM 62X USR.FE ERU 00000000B FRADER-FRADER-CASA USR.FE ERU 00000100B FRADER-FRADER-CASA USR.FE ERU 00000100B FRADER-FRADE	45X UNI LIS EQU 00001100B B BIT CHARACTERS 47X UNI LISK EQU 0000010B CLOCK X 14 47X UNI LISK EQU 00000010B CLOCK X 14 49X UNI AX EQU 00000010B CLOCK X 64 49X UNI AX EQU 00000010B CLOCK X 64 49X UNI AX EQU 0100000B ERRERESET SEX UCI RE EQU 00100000B ERRERESET ENBEREDE SEX UCI RE EQU 00100000B ERRERESET ENBEREDE SEX UCI RE EQU 000000010B ERRERE INTERRUPTS FLAG 57X UCI RE EQU 000000010B ERRERE INTERRUPTS FLAG 57X UCI RE EQU 000000010B FRAMELE INTERRUPTS FLAG 57X UCI RE EQU 000000010B FRAMELE INTERRUPTS 57X UCI RE EQU 00010000B FRAMING ERROR 67X UCI RE EQU 00010000B FRAMING ERROR 67X UCI RE EQU 00010000B FRAMING ERROR 67X USR FRE EQU 00010000B FRAMING ERROR 67X USR FRE EQU 000000010B FRAMING FRANCE 67X USR FRE EQU 000000010B FRANCE FRANCE 67X USR FRE EQU 000000010B FRANCE FRANCE 67X USR FRE EQU 000000010B FRANCE FRAN	45X UH.1.18 EÜU 00001100B B BIT CHARGTERS 44X UH.1.1X EÜU 0000001B CLOCK X 14 47X UH.1.14X EÜU 0000001B CLOCK X 14 49X UH.1.64X EÜU 0000001B CLOCK X 64 49X UH.1.64X EÜU 0000001B CLOCK X 64 52X UCT.1R EÜU 0100000B INTERNAL RESET 53X UCT.1R EÜU 0010000B REÄBERENT UN IKUL PLAG 54X UCT.1R EÜU 0010000B REÄBERENT PLAG 55X UCT.1R EÜU 0010000B REÄBERENT FLAG 55X UCT.1E EÜU 00010000B REÄBERE INTERNUFTS FLAG 55X UCT.1E EÜU 00010000B FRÄMFIT ENÄBLE 55X UCT.1E EÜU 0000001B FRÄMFIT ENÄBLE 55X UCT.1E EÜU 0000001B FRÄMFING ERRÖR 64X USR. FE EÜU 0010000B PRÄMFIT ERRÖR 64X USR. EE EÜU 00010000B REÄBERT 64X USR. EE EÜU 00000100B REÄBERT 64X USR. TXR EÜU 00000100B REÄBTITER READY 65X ÜSR. RXR EÜU 0000010B RAÄNTITER READY 65X ÜSR. RXR EÜU 0000010B RAÄNTITER READY 65X ÜSR. RXR EÜU 0000010B RAÄNTITER READY 65X ÜSR. TXR EÜU 0000010B RAÄNTITER READY	45X UNILLE ERU 0000100B 8 BIT CHARACTERS 44X UNILLE ERU 0000001B CLOCK XI 49X UNILLEX ERU 0000001B CLOCK XI 49X UNILLEX ERU 0000001B CLOCK XI 50X ** COMMAND INSTRUCTION BITS. 51X UCI.RE ERU 00010000B READER-ON LUNINUL FLAG 53X UCI.RE ERU 00010000B READER-ON LUNINUL FLAG 54X UCI.RE ERU 00010000B READER-ON LUNINUL FLAG 55X UCI.RE ERU 0001000B READER-ON LUNINUL FLAG 55X UCI.RE ERU 0001000B READER-ON LUNINUL FLAG 55X UCI.RE ERU 0001000B READER-ON LUNINUL FLAG 55X UCI.RE ERU 0000001B TRAMSHIT ENABLE 55X UCI.RE ERU 0000001B TRAMSHIT ENABLE 55X UCI.RE ERU 00000001B TRAMSHIT ENABLE 55X UCI.RE ERU 00000001B TRAMSHITER ENPTY 64X USK.RE ERU 00000000B TRAMSHITER ENPTY 65X USK.RE ERU 00000000B TRAMSHITER ENPTY	4.5 WITTER END 00000100 B BIT CHARACTERS 4.5 WITTER END 00000100 CLOCK X 64 4.8 WITTER END 00000100 CLOCK X 64 4.8 WITTER END 00000100 CLOCK X 64 4.8 WITTER END 00000100 CLOCK X 64 5.5 WITTER END 00100000 INTERNAL RESET CAN UNITED THE END 00100000 ERROR RESET CHARLE END 00000100 ERROR RESET CHARLE END 00001000 ERROR RESET CHARLE END 0000100 ERROR RESET CHARLE END	0000,0014 0000,0002 0000,0003 0000,0003 0000,000	UMI.L8 UMI.15X UMI.64X		6 BIT CHARACTERS 7 RIT CHARACTERS
46X UMI.1X EQU 0000001B CLOCK X 14 47X UMI.164X EQU 0000001B CLOCK X 64 49X UMI.64X EQU 0000001B CLOCK X 64 49X 50X ** COMMAND INSTRUCTION BITS. 51X UCI.R EQU 00010000B READER-ON CUNIKUL F 53X UCI.RE EQU 00010000B RECEIVE ENABLE 55X UCI.RE EQU 00000010B RECEIVE ENABLE 55X UCI.RE EQU 00000010B RAMBLE INTERRUPTS F 55X UCI.RE EQU 00000010B RAMBLE INTERRUPTS F 55X UCI.RE EQU 00000010B RAMBLE INTERRUPTS F 55X UCI.RE EQU 00000000B FRANKE 55X UCI.RE EQU 00000000B FRANKE 55X UCI.RE EQU 00000000B FRANKE 63X USR.FE EQU 00001000B FRANKING ERROR 63X USR.FE EQU 0000100B FARMING ERROR 64X USR.TXE EQU 00000100B TRANSMITTER EMPTY 65X USR.TXE EQU 00000100B TRANSMITTER READY 65X USR.TXE EQU 00000100B TRANSMITTER READY 65X USR.TXE EQU 00000001B TRANSMITTER READY 65X USR.TXE EQU 00000010B TRANSMITTER READY 65X USR.TXE EQU 00000010B TRANSMITTER READY 65X USR.TXE EQU 00000001B TRANSMITTER READY	44X UNI.1X EQU 00000001B CLOCK X 14 49X UNI.16X EQU 00000001B CLOCK X 14 49X UNI.16X EQU 00000001B CLOCK X 14 49X UNI.16X EQU 00000001B CLOCK X 14 50X X UCI.R EQU 01000000B READER-ON CUNIRUL LEGU 53X UCI.RE EQU 00010000B READER-ON CUNIRUL LEGU 55X UCI.RE EQU 00010000B RECRUE FENGLE 55X UCI.RE EQU 00000010B READER INTERRUPT'S FLAG 55X UCI.RE EQU 0000000B PREADER INTERRUPT'S FLAG 55X UCI.RE EQU 00010000B UNERRUPT'S FRAGE 55X UCI.RE EQU 00010000B TRANSMITTER EMPTY 65X USR, FE EQU 00001000B TRANSMITTER EMPTY 65X USR, FE EQU 00000001B TRANSMITTER READTY 65X USR, TXR EQU 000000001B TRANSMITTER READTY	44x UHI.14x ERU 00000010B CLOCK 14 44x UHI.14x ERU 00000010B CLOCK 14 49x UHI.14x ERU 0000001B CLOCK 14 49x UHI.14x ERU 0000001B CLOCK 14 49x UHI.14x ERU 0000001B CLOCK 14 50x WC LIR ERU 0100000B INTERNAL RESET 52 UCI.RE ERU 00010000B RECELVE ENABLE 53 UCI.RE ERU 00010000B RECELVE ENABLE 55 UCI.RE ERU 00000010B RECELVE ENABLE 55 UCI.RE ERU 00010000B UVERNIN ERROR 62 USR TXE ERU 00010000B PRATITY ERROR 62 USR TXE ERU 0000010B REASITY ERROR 65 USR TXE ERU 00000010B REASITY ERROR 65 USR TXE ERU 000000010B REASITY ERROR 65 USR TXE ERU 000000000B USR TXE ERU 00000000B USR TXE ERU 0000000B USR TXE ERU 000000B USR TXE ERU 00000B USR TXE ERU 00000B USR TXE ERU 00000B USR TXE	44x UM1.1X EBU 00000010B CLCCK X 14 49x UM1.14x EBU 00000010B CLCCK X 44 49x UM1.14x EBU 00000010B CLCCK X 44 49x UM1.14x EBU 00000010B CLCCK X 44 50 X ** CDMHAMD INSTRUCTION BITS. 51 X UCI. IR EBU 0100000B INTERNAL RESET 52	4.6.X. UPL 1.1.X. EUU 0.000000018 D.LCGX X 1.4. 4.6.X. UPL 1.1.X. EUU 0.000000018 D.LCGX X 1.4. 4.6.X. UPL 1.1.X. EUU 0.000000118 D.LCGX X 1.4. 4.7.X. UPL 1.6.X. EUU 0.000000118 D.LCGX X 1.4. 5.2.X. UPL 1.1.X. EUU 0.000000018 D.READEX-IN-LUNINUL F.LAGY 6.2.X. USR T.YE EUU 0.000000018 D.READEX-IN-LUNINUL F.READEX 6.2.X. USR T.YE EUU 0.0000000018 D.READEX-IN-LUNINUL F.READEX 6.2.X. USR T.YE EUU 0.00000000018 D.READEX-IN-LUNINUL F.READEX 6.2.X. USR T.YE EUU 0.0000000018 D.READEX-IN-LUNINUL F.READEX-IN-LUNINUL F.READEX-IN-LUN	000,000 000,002 000,003 000,040 000,020	UMI.16X UMI.16X UMI.64X	:	8 BIT CHARACTERS
49X UMI.64X EQU 000000011B CLUCK X 15 49X WII.64X EQU 000000011B CLUCK X 64 49X COMMAND INSTRUCTION BITS. 50X W. COMMAND INSTRUCTION BITS. 51X UCI.RC EQU 00010000B READER-ON CUNIKUL FEXA UCI.RC EQU 00010000B READER-ON CUNIKUL FEXA UCI.RC EQU 00000010B RECEIVE EAABLE 55X UCI.TC EQU 00000010B RANSHIT ENABLE 55X UCI.TC EQU 00000010B FRANSHIT ENABLE 59X ** STATUS READ COMMAND BITS. 61X USR.FC EQU 0001000B FRANSHIT ENABLE 62X USR.FC EQU 0001000B FARAHING ERROR 63X USR.FC EQU 0001000B FARITY ERROR 64X USR.TX EQU 0000100B FARITY ERPTY 64X USR.FX EQU 0000010B TRANSMITTER READY 65X USR.FX EQU 00000010B TRANSMITTER READY 65X USR.FX EQU 00000001B TRANSMITTER READY 65X USR.FX EQU 00000001B TRANSMITTER READY 65X USR.FX EQU 00000001B TRANSMITTER READY	47X UMI.16X EQU 00000010B CLOCK X 64 49X UMI.64X EQU 00000010B CLOCK X 64 49X 49X 50X ** COMMAND INSTRUCTION BITS. 51X 52X UCI.1R EQU 00010000B READER-ON CUNINCL F 53X UCI.1R EQU 0001000B RECEIVE ENABLE 55X UCI.1R EQU 00000010B RABLE INFRRUPTS F 56X UCI.1R EQU 00000010B RABLE INFRRUPTS F 56X UCI.1R EQU 00000010B FRABLE INFRRUPTS F 56X UCI.1R EQU 00010000B FRABLE INFRRUPTS F 56X UCI.1R EQU 00010000B FRABLE INFRRUPTS F 56X UCI.1R EQU 0001000B FRABLE INFRRUPTS F 56X UCI.1R EQU 0001000B FRABLE INFRRUPTS F 66X USR.FE EQU 0001000B FRABITY EROR F 65X USR.FE EQU 00000100B FRABLE F 65X USR.FE EQU 00000100B FRABITY FROM F 65X USR.FE EQU 000000100B FRABLE F 65X USR.FE EQU 00000010B FRABITY F 65X USR.FE EQU 00000010B FRABLE F 65X USR.FE EQU 000000010B FRABLE F 65X USR.FE EQU 0000000010B FRABLE F 65X USR.FE EQU 00000000010B FRABLE F 65X USR.FE EQU 00000000010B FRABLE F 65X USR.FE EQU 0000000000000000000000000000000000	49X UMI.64X EQU 00000010B CLUCK X 64 49X UMI.64X EQU 00000011B CLUCK X 64 49X 49X COMMAND INSTRUCTION BITS. 51X UCI.TR EQU 0100000B INTERNAL RESET 53X UCI.RE EQU 00010000B ERROR RESET 55X UCI.RE EQU 00000010B TRANSMIT ENABLE 55X UCI.TE EQU 00000010B TRANSMIT ENABLE 55X UCI.TE EQU 00000010B TRANSMIT ENABLE 55X UCI.TE EQU 00010000B FRAMING ERROR 61X USR.FE EQU 00010000B FRAMING ERROR 61X USR.FE EQU 00010000B FRAMING ERROR 62X USR.DE EQU 00010000B FRAMING ERROR 63X USR.PE EQU 00010000B FRAMING ERROR 63X USR.FE EQU 00010000B FRAMING ERROR 63X USR.FE EQU 0000100B FRAMINTER READY 65X USR.TXE EQU 00000010B TRANSMITTER READY 65X USR.TXE EQU 00000001B TRANSMITTER READY 65X USR.TXE EQU 000000001B TRANSMITTER READY 65X USR.TXE EQU 000000001B TRANSMITTER READY 65X USR.TXE EQU 0000000000000000000000000000000000	47X UMI.164X EQU 00000010B CLUCK X 64 49X 50X ** COMMAND INSTRUCTION BITS, 51X 52X UCI.1R EQU 0100000B INTERNAL RESET 52X UCI.1R EQU 000000B READER-ON UUNINUL P 52X UCI.1R EQU 000000B READER-ON UUNINUL P 53X UCI.1R EQU 0000010B RECEIVE ENABLE 55X UCI.1R EQU 0000010B RECEIVE ENABLE 55X UCI.1R EQU 0000010B RECEIVE ENABLE 55X UCI.1R EQU 00000010B RECEIVE ENABLE 55X UCI.1R EQU 00000010B RECEIVE ENABLE 55X UCI.1R EQU 00000010B PARITY ERROR 61X USR. PE EQU 0000100B PARITY ERROR 62X USR. PE EQU 0000100B PARITY ERROR 64X USR. TXE EQU 00000100B TRANSMITTER EMPTY 65X USR. TXE EQU 00000100B TRANSMITTER EMPTY 65X USR. TXE EQU 00000100B TRANSMITTER EMPTY 65X USR. TXE EQU 00000010B TRANSMITTER READY 65X USR. TXE EQU 00000001B TRANSMITTER READY 65X USR. TXE EQU 000000001B TRANSMITTER READY 65X USR. TXE EQU 000000000 TRANSMITTER READY 65X USR. TXE EQU 000000000 TRANSMITTER READY 65X USR. TXE EQU 000000000 TRANSMITTER READY 65X USR. TXE EQU 0000000000 TRANSMITTER READY 65X USR. TXE EQU 000000000 TRANSMITTER READY	49.X UM1.44X EQU 00000001B CLUCK X 64 50.X UM1.64X EQU 0000001B CLUCK X 64 50.X ULI.R EQU 00010000B READERSET 53.X ULI.R EQU 00010000B READERSET 53.X ULI.R EQU 00010000B READERSET 53.X ULI.R EQU 000010B READERSET 53.X ULI.R EQU 0000010B READERSET 53.X ULI.R EQU 0000010B READERSET 53.X ULI.R EQU 0000100B READERSET 53.X ULI.R EQU 0000100B READERSET 54.X USR.FE EQU 0000100B READERSET 54.X USR.FE EQU 0000010B READERSET 55.X USR.FE EQU 00000010B READERSET 55.X USR.FE EQU 00000001B READERSET 55.X USR.FE EQU 000000001B READERSET 55.X USR.FE EQU 0000000000000000000000000000000000	000,002 000,003 000,000 000,040 000,020	UMI.16X	:	CLOCK X 1
### ### ##############################	### ### ##############################	49x	### COMMAND INSTRUCTION BITS. 51X COMMAND INSTRUCTION BITS.	50X *** COMMAND INSTRUCTION BITS. 51X UELTR EUU 0000000B REABER-DN LUNI HULL F 53X UELTR EUU 00010000B REABER-DN LUNI HULL F 53X UELTR EUU 00010000B REABER ENGRE RESET 55X UELTE EUU 0000010B REABER ENGRE	000.100 000.040 000.020 000.004	*		CLOCK X 16 CLOCK X 64
\$1X 52X UCI.IR EQU 0100000B INTERNAL RESET 53X UCI.ER EQU 00010000B ERECEIVE ENABLE 54X UCI.ER EQU 00010000B ERECEIVE ENABLE 55X UCI.IE EQU 00000010B ERECEIVE ENABLE 55X UCI.IE EQU 00000010B FRANSMIT ENABLE 59X ** STATUS READ COMMAND BITS. 60X 61X USR.FE EQU 00100000B FRAMING ERECR 61X USR.FE EQU 00010000B FRAMING ERECR 63X USR.FE EQU 00001000B FRANSMITTER EMPTY 64X USR.TXE EQU 00000010B FRANSMITTER EMPTY 65X USR.FXE EQU 00000010B TRANSMITTER READY 65X USR.FXE EQU 00000010B TRANSMITTER READY 65X USR.FXE EQU 00000001B TRANSMITTER READY 65X USR.FXE EQU 00000001B TRANSMITTER READY	51X	51X	51X	52X UCI IR EQU 00100000B INTERNAL RESET 53X UCI FAG EQU 00010000B READER-DN LUNINUL P 55X UCI FE EQU 00001000B RECED FROM ERE EN EREC FOR EAST 55X UCI IE EQU 00000010B EREC FOR EAST 55X UCI IE EQU 00000010B EREC FOR EAST 55X UCI IF EQU 00000010B FRAMING EREC FOR 55X USR DE EQU 0001000B FRAMING EREC FOR 55X USR DE EQU 0001000B FRAMING EREC FOR 55X USR DE EQU 00000100B FRAMING EREC FOR 55X USR TXE EQU 00000100B FRAMING EREC FOR 55X USR TXE EQU 00000100B FRAMING EREC FOR 55X USR TXE EQU 00000010B FRAMINTTER READY 55X USR TXE EQU 00000010B FRAMINTTER READY 55X USR TXE EQU 00000001B FRAMINTTER FRAMINTTER FOR TXE FOR 55X USR TXE EQU 00000001B FRAMINTTER FOR TXE FOR 55X USR TXE FOR 55	000.100 000.040 000.020 000.004	**	JANĒ	75.
53X UCI, FR EQU 0010000B READER-NO CUNIKUL F 54X UCI, ER EQU 00010000B ERROR RESET 55X UCI, ER EQU 0001000B ERROR RESET 55X UCI, IE EQU 00000010B FRÀMILE INTERRUPTS F 59X ** STÀTUS READ CONHAND BITS, 60X USR, FE EQU 0001000B FRÀMING ERROR 63X USR, FE EQU 0001000B FARITY ERROR 64X USR, TXE EQU 0000010B FARITY ERROR 64X USR, TXE EQU 0000010B FARITY ERROR 65X USR, FXX EQU 0000010B TRANSMITTER EMPTY 65X USR, FXX EQU 00000010B TRANSMITTER READY 65X USR, FXX EQU 00000010B TRANSMITTER READY 65X USR, TXX EQU 00000010B TRANSMITTER READY	53X UCI.KD EQU 0010000B READERANT LEGU 0010000B FREDERING ENGRESET 55X UCI.KE EQU 00010000B FRECEIVE ENABLE 55X UCI.KE EQU 00000100B FRECEIVE ENABLE 55X UCI.TE EQU 00000010B FRANSMIT ENABLE 55X UCI.TE EQU 00000010B FRANSMIT ENABLE 58X 59X ** STATUS READ COMMAND BITS. 60X	53X UCI.KG EQU 0010000B KEADERANL FOR CUNIKUL FOR EQU 00010000B ERROR RESET 55X UCI.KE EQU 00010000B ERROR RESET 55X UCI.KE EQU 00000100B FRECEIUE ENABLE 55X UCI.TE EQU 00000010B FRANSMIT ENABLE 58X 57X UCI.TE EQU 00000010B FRANSMIT ENABLE 58X 59X ** STATUS READ CONHAND BITS. 60X STATUS READ 0010000B FRANSMIT ENABLE 53X USR.DE EQU 00010000B FRANSMIT ERROR 63X USR.TE EQU 0000100B FRANSMITTER EMPTY 64X USR.TE EQU 00000100B FRANSMITTER EMPTY 65X USR.FE EQU 00000100B FRANSMITTER EMPTY 65X USR.FE EQU 00000100B FRANSMITTER EMPTY 65X USR.FE EQU 00000100B FRANSMITTER READY 65X USR.FE EQU 00000010B FRANSMITTER READY 65X USR.FE EQU 00000010B FRANSMITTER READY 65X USR.FE EQU 00000010B FRANSMITTER READY 65X USR.FEXT CONSLX	53% UCI.RG EQU 00010000 READER-NOL CUNIKUL F 54% UCI.ER EQU 00010000 ERROR RESET 55% UCI.RE EQU 00000100 ERROR RESET 55% UCI.RE EQU 00000100 ERROR RESET 56% UCI.TE EQU 00000010 TRANSHIT ENABLE 58% STATUS READ COMHAND BITS. 60% STATUS READ CONHAND BITS. 60% STATUS READ 00010000 OVERRUN ERROR 63% USR.PE EQU 00010000 PARITY EROR 63% USR.PE EQU 00001000 TRANSHITTER EMPTY 64% USR.TXR EQU 00000100B TRANSHITTER EMPTY 65% USR.TXR EQU 00000010B TRANSHITTER EMPTY 65% USR.TXR EQU 00000010B TRANSHITTER READY 65% USR.TXR EQU 00000001B TRANSHITTER READY	S3X UCI : KG	000 040 000 020 000 004			:
54X UCI.ER EQU 0001000B ERROR RESET 55X UCI.RE EQU 0000100B RECETUÉ ENABLE 55X UCI.IE EQU 0000010B RABLE INTERRUPTS F 59X ** STATUS READ CONHAND BITS. 60X USR.FE EQU 0001000B FRAMING ERROR 62X USR.DE EQU 0001000B PARITY EROR 63X USR.TXE EQU 0000100B FARITY EROR 64X USR.TXE EQU 00000100B FARITY EROR 65X USR.TXE EQU 00000100B FARITY EROP 65X USR.TXE EQU 00000100B TRANSMITTER EMPTY 65X USR.XX EQU 00000010B TRANSMITTER READY 65X USR.XX EQU 00000001B TRANSMITTER READY	54X UCI,ER EQU 0001000B ERROR RESET 55X UCI,RE EQU 00000100B RECEIVE ENABLE 57X UCI,TE EQU 00000010B TRANSMIT ENABLE 58X 59X ** STATUS READ CONHAND BITS, 60X 61X USR,FE EQU 0001000B FRANT ERROR 62X USR,TE EQU 0001000B FARITY ERROR 63X USR,TE EQU 0000100B FARITY ERROR 64X USR,TE EQU 0000100B TRANSMITTER EMPTY 65X USR,TE EQU 0000010B TRANSMITTER EMPTY 65X USR,TE EQU 00000010B TRANSMITTER READY 65X USR,TE EQU 00000010B TRANSMITTER READY 65X USR,TE EQU 00000001B TRANSMITTER READY 65X USR,TEXT EQU 00000001B TRANSMITTER READY 65X USR,TEXT EQU 00000001B TRANSMITTER READY	54X UCI, ER EQU 0001000B ERROR RESET 55X UCI, RE EQU 0000010B RECEIVE ENABLE 55X UCI, RE EQU 0000010B RECEIVE ENABLE 57X UCI, TE EQU 0000001B TRANSMIT ENABLE 58X 59X ** STATUS READ CONHAND BITS. 60X 51X USR, FE EQU 0010000B FRAMING ERROR 63X USR, PE EQU 00010000B FARITY ERROR 64X USR, TXE EQU 0000100B FARITY ERROR 65X USR, TXE EQU 0000010B TRANSMITTER EMPTY 65X USR, RXK EQU 0000010B TRANSMITTER EMPTY 65X USR, RXK EQU 0000010B TRANSMITTER EMPTY 65X USR, RXK EQU 00000010B TRANSMITTER READY 65X USR, RXK EQU 00000001B TRANSMITTER READY 65X USR, RXX EXTEXT CONSCIPE 65X USR, RXX EXCENTING CONSCIPE 65X USR, RXX E	54X UCI,ER EQU 0001000B ERROR RESET 55X UCI,RE EQU 0000100B RECETUE ENABLE 55X UCI,IE EQU 0000010B RECETUE ENABLE 57X UCI,TE EQU 00000001B TRANSMIT ENABLE 59X ** STATUS READ CONHAND BITS. 59X WS. FE EQU 00010000B FRAMING ERROR 62X USR.DE EQU 00010000B PARITY EROR 63X USR.PE EQU 00001000B PARITY EROR 64X USR.TXE EQU 00000100B TRANSMITTER EMPTY 65X USR.TXE EQU 00000100B TRANSMITTER EMPTY 65X USR.TXE EQU 00000010B TRANSMITTER EMPTY 65X USR.TXE EQU 00000010B TRANSMITTER READY 65X USR.TXR EQU 00000001B TRANSMITTER READY 65X USR.TXR EQU 000000001B TRANSMITTER READY 65X USR.TXR EQU 0000000001B TRANSMITTER READY 65X USR.TXR EQU 0000000001B TRANSMITTER READY 65X USR.TXR EQU 00000000000000000000000000000000000	54X UCI.ER EQU 00001000B ERROR RESET 55X UCI.RE EQU 00000010B ENABLE ENABLE 55X UCI.RE EQU 00000010B ENABLE ENABLE 55X UCI.TE EQU 00000010B TRANSMIT ENABLE 55X USR.FE EQU 00010000B TRANSMIT ENABLE 55X USR.FE EQU 00010000B FRANT ERROR 64X USR.FE EQU 00010000B FRANT ERROR 64X USR.FE EQU 0000100B TRANSMITTER ERPTY 65X USR.FXE EQU 0000100B TRANSMITTER READY 65X USR.FXE EQU 00000010B TRANSMITTER READY 65X USR.FXE EQU 00000001B TRANSMITTER READY 65X USR.FXE EQU 000000001B TRANSMITTER READY 65X USR.FXE EQU 0000000001B TRANSMITTER READY 65X USR.FXE EQU 0000000000000000000000000000000000	000.020	UCITO		KEADER-ON CONTROL FLAG
S5X UCI.RE EQU	S5X UCI.RE EQU	55X UCI, RE EQU 00000100 RECETUE ENABLE 55X UCI, IE EQU 00000010 ENABLE INTERUPTS F 55X UCI, IE EQU 00000010 TRANSMIT ENABLE 58X 59X ** STATUS READ CONHAND BITS. 60X 50X 50X 50X 50X 50X 50X 50X 50X 50X 5	### 55X UCI.RE EQU 00000100B RECEIVE ENABLE	### ### ##############################	000.000	UCI.ER		ERROR RESET
56X UCI,1E EUU 00000010H ENABLE INTERUOTIS T 59X ** STATUS READ CONHAND BITS. 59X WS,FE EQU 0010000B FRAHING ERROR 62X USR,FE EQU 0001000B PARITY ERROR 63X USR,FE EQU 00000100B FARITY ERROR 64X USR,TXE EQU 0000010B FARITY ERROR 65X USR,FXX EQU 0000010B TRANSMITTER EMPTY 66X USR,TXE EQU 0000001B TRANSMITTER READY 65X USR,TXX EQU 00000001B TRANSMITTER READY	56X UCI,1E EUU 00000010B ENABLE INTERUCTIS T 57X UCI,TE EQU 00000001B TRANSHIT ENABLE 58X 59X ** STATUS READ CONHAND BITS. 60X	56X UCI,1E EUU 00000010B ENABLE INTERUCTS T 57X UCI,TE EQU 00000001B TRANSMIT ENABLE 58X 59X ** STATUS READ CONHAND BITS. 60X 60X 60X 60X 60X 60X 60X 60X 60X 60X	56X UCI,1E EUU 00000010B ENABLE INTERUOTIS T 57X UCI,TE EQU 00000001B TRANSHIT ENABLE 58X 51X USR,FE EQU 0010000B FRAHING ERROR 61X USR,PE EQU 0001000B PARITY EROR 63X USR,PE EQU 00001000B PARITY EROR 64X USR,TXE EQU 00000100B TRANSHITTER ENFTY 65X USR,TXR EQU 00000010B TRANSHITTER ENFTY 65X USR,TXR EQU 00000001B TRANSHITTER READY 65 X USR,TXR EQU 00000001B TRANSHITTER READY	55X UCITE ENU 00000010B ENABLE INTERNALIS T 59X WCITE ENU 00000001B TRANSMIT ENABLE 58X UCITE ENU 00000001B TRANSMIT ENABLE 62X USR.FE ENU 00001000B PARITY EROR 62X USR.TXE ENU 00000100B TRANSMITTER EMPTY 65X USR.TXE ENU 00000001B TRANSMITTER EMPTY 65X USR.TXE ENU 00000001B TRANSMITTER READY 65X USR.TXE ENU 00000001B TRANSMITTER READY 65X USR.TXE ENU 00000001B TRANSMITTER READY 65X USR.TXTEXT CONSLX	000.000	UCI.RE	:	: L
59X	59X UST. IE ERU COCCOCIE ITERNSTIT ENTER 59X ** STATUS READ COMMAND BITS. 60X 61X USR.FE ERU COCTOCOCE FRANTING ERROR 61X USR.PE ERU COCTOCOCE PARTIY ERROR 64X USR.TXE ERU COCCOCOCE TRANSHITTER EN 65X USR.TXE ERU COCCOCOCE TRANSHITTER EN 66X USR.TXR ERU COCCOCOCE TRANSHITTER ER 66X USR.TXR ERU COCCOCOCE TRANSHITTER READY 66X USR.TXR ERU COCCOCOCE TRANSHITTER READY 66X USR.TXR ERU COCCOCOCE TRANSHITTER RE	59X UST. IE ERU COCCOCIE 59X XX STATUS READ COMMAND BITS. 59X XX STATUS READ COMMAND BITS. 61X USR. FE ERU COCCOCOCE 63X USR. FE ERU COCCOCOCE 63X USR. FE ERU COCCOCOCE 64X USR. TXE ERU COCCOCOCE 65X USR. TXE ERU COCCOCOCOCE 65X USR. TXE ERU COCCOCOCOCOCE 65X USR. TXE ERU COCCOCOCOCOCOCOCOCOCOCOCOCOCOCOCOCOCOC	59X ULI, IE ERU COCOCOLIB 59X ** STATUS READ COMMAND BITS, 60X USR.FE ERU COCIOCOCOB GUERRUN ERROR 62X USR.PE ERU COCOCIOCOB GUERRUN ERROR 64X USR.TXE ERU COCOCIOCB FARITY EROR 65X USR.TXR ERU COCOCOLOCB TRANSMITTER EN 66X USR.TXR ERU COCOCOCOLB TRANSMITTER RE 66X USR.TXR ERU COCOCOCOLB TRANSMITTER RE 66X USR.TXR ERU COCOCOCOLB	59X WA STATUS READ COMMAND BITS. 59X STATUS READ COMMAND BITS. 59X STATUS READ COMMAND BITS. 59X STATUS READ COMMAND BITS. 52X USR DE EQU 0001000B FRANTY ERGE COMMAND BITS. 52X USR TXR EQU 0000100B TRANSHITTER READ COMMONDIB TRANSHITTER REGENERATION TO STATE	4>>	UCI.IE		ກ:
59X ** STATUS READ COMMAND BITS. 60X 61X USR.FE EQU 00010000B DVERRUN ERROR 62X USR.DE EQU 00001000B FARITY ERROR 64X USR.TE EQU 00000100B TRANSMITTER EM 65X USR.TXR EQU 0000010B TRANSMITTER ED 66X USR.TXR EQU 0000001B TRANSMITTER READY 65 USR.TXR EQU 0000001B TRANSMITTER RE	59X ** STATUS READ COMMAND BITS, 60X 60X 62X USR.FE EQU 00010000B FRANING ERROR 63X USR.FE EQU 00001000B FARITY ERROR 64X USR.TXE EQU 00000100B TRANSHITTER EM 65X USR.TXE EQU 00000100B TRANSHITTER EM 66X USR.TXR EQU 00000010B TRANSHITTER RE 66X USR.TXR EQU 0000001B TRANSHITTER RE 66X USR.TXR EQU 0000001B TRANSHITTER RE	59X ** STATUS READ COMMAND BITS, 60X 61X USR.FE EQU 00010000B DVEREUN ERROR 63X USR.PE EQU 00001000B PARITY ERROR 64X USR.TXE EQU 00000100B TRANSMITTER EM 65X USR.TXR EQU 00000010B TRANSMITTER EM 66X USR.TXR EQU 0000001B TRANSMITTER RE 65 USR.TXR EQU 0000001B TRANSMITTER RE 65 USR.TXR EQU 0000001B TRANSMITTER RE 65 USR.TXR EQU 0000001B TRANSMITTER RE	59X ** STATUS READ COMMAND BITS, 60X 61X USR.FE EQU 00010000B FRANING ERROR 63X USR.PE EQU 00001000B PARITY EROR 64X USR.TXE EQU 00000100B FARITY EROR 65X USR.TXR EQU 0000010B FREEIVER READY 65X USR.TXR EQU 0000001B TRANSMITTER RE 65 USR.TXR EQU 00000001B TRANSMITTER RE	59X ** 50X	000.000	UCI+1E E	ο.	LAHNOM I LENHERE
61X USR.FE EQU 00100000B FRAHING ERROR 62X USR.DE EQU 00010000B DVERRUN ERROR 63X USR.TXE EQU 0000100B FARITY ERROR 65X USR.TXE EQU 00000100B TRANSHITTER EM 65X USR.TXR EQU 00000010B TRANSHITTER RE 65X USR.TXR EQU 00000011B TRANSHITTER RE 65 USR.TXR EQU	61X USR.FE EGU 0010000B FRAMING ERROR 62X USR.DE EGU 0001000B PARITY ERROR 63X USR.PE EGU 00000100B FARITY ERROR 64X USR.TXE EGU 0000010B TRANSMITTER EM 64X USR.TXR EGU 0000001B TRANSMITTER RE 65 USR.TXR EGU 0000001B TRANSMITTER RE 65 XTEXT CONSLX	61X USR.FE EQU 00100000B FRAHING ERROR 62X USR.DE EQU 00010000B OVERRUN ERROR 63X USR.PE EQU 00001000B FARITY ERGÉ 64X USR.TXE EQU 00000100B TRANSMITTER EM 65X USR.TXR EQU 0000001B TRANSMITTER RE 66X USR.TXR EQU 00000001B TRANSMITTER RE 65X USR.TXR EQU 00000001B	61X USR.FE EQU 0010000B FRAHING ERROR 62X USR.DE EQU 0001000B OVERRUN ERROR 63X USR.PE EQU 0000100B FARITY EROR 64X USR.TXE EQU 0000010B TRANSNITTER EM 64X USR.TXR EQU 0000001B TRANSNITTER RE 657 USR.TXR EQU 00000001B TRANSNITTER RE	61X USK.FE EQU 0001000B FRAMING ERRUR 62X USR. DE EQU 0001000B PARITY ERRUR 63X USR. TXE EQU 0000100B FARITY ERRUR 64X USR. TXE EQU 00000010B FEELUER FEELU 66X USR. TXR EQU 0000001B TRANSMITTER FEELU 65X USR. TXR EQU 00000001B TRANSMITTER FEELU 65X USR EXTRANSMITTER		*	КЕАД СОММАКД	BITS.
62X USR. DE EQU 00010000B DVERBUN ERROR 63X USR. PE EQU 0000100B PARITY EROR 64X USR. TXE EQU 00000100B TRANSMITTER EM 65X USR. TXR EQU 00000010B TRANSMITTER ED 66X USR. TXR EQU 0000001B TRANSMITTER RE 67 USR. TXEXT CONSLX	62X USR.DE EQU 00010000B DVERRUN ERROR 63X USR.PE EQU 00000100B PARITY ERROR 64X USR.TXE EQU 0000010B TRANSMITTER EM 64X USR.TXR EQU 0000001B TRANSMITTER RE 65 USR.TXR EQU 0000001B TRANSMITTER RE 65 XTEXT CONSLX	62X USR.DE ERU 00010000B DVERRUN ERROR 63X USR.PE ERU 0000100B PARITY ERRÖR 65X USR.TXE ERU 00000100B TRANSMITTER EM 65X USR.TXR ERU 00000001B TRANSMITTER RE 65 USR.TXR ERU TRANSMITTER RE 65 USR.TXR ERU TRANSMITTER RE 67	62X USR. DE EQU 0001000B DVERBUN ERRDR 63X USR. PE EQU 0000100B PARITY ERDR 65X USR. TXE EQU 0000010B TRANSMITTER EM 66X USR. TXE EQU 0000001B TRANSMITTER RE 65 USR. TXEXT CONSLX	62X USR.DE EQU 00001000B PARTIY ERFÜR 63X USR.PE EQU 0000100B TRANSMITTER EN 64X USR.TXR EQU 0000010B RECEIVER REALIY 65X USR.TXR EQU 00000001B TRANSMITTER RE 65Y USR.TXR EQU 000000001B TRANSMITTER RE 65Y USR.TXR EQU 00000001B TRANSMITTER RE 65Y USR.TXR EQU 000000001B TRANSMITTER RE 65Y USR.TXR EQU 000000000000 TRANSMITTER RE 65Y USR.TXR EQU 0000000000 TRANSMITTER RE 65Y USR.TXR EQU 0000000000 TRANSMITTER RE 65Y USR.TXR EQU 000000000 TRANSMITTER RE 65Y USR.TXR EQU 0000000000 TRANSMITTER RE 65Y USR.TXR EQU 000000000 TRANSMITTER RE 65Y USR.TXR EQU 00000000 TRANSMITTER RE 65Y USR.TXR EQU 0000000 TRANSMITTER RE 65Y USR.TXR EQU 000000 TRANSMITTER RE 65Y USR.TXR EQU 00000 TRANSMITTER RE 65Y U	000,040	USK, FE	:	FRAMING ERROR
63X USR.FE EQU 00001000B FARITY ERRER 64 65X USR.TXE EQU 0000010B TRANSMITTER EN 66X USR.TXR EQU 00000001B TRANSMITTER RE 65 USR.TXR XTEXT CONSLX	63X USR.PE EQU 00001000B PARITY ERGR 64X USR.TXE EQU 00000100B TRANSMITTER EM 64X USR.TXR EQU 0000001B TRANSMITTER RE 65 USR.TXR EQU 0000001B TRANSMITTER RE 67 XTEXT CONSLX	63X USR.PE EQU 00001000B FARITY ERGR 64X USR.TXE EQU 00000100B TRANSHITTER EM 65X USR.TXR EQU 00000001B TRANSHITTER RE 65 USR.TXR XTEXT CONSLX	63X USR.PE EQU 00001000B PARITY ERRE 64X USR.TXE EQU 00000010B RECEIVER ER 65X USR.TXR EQU 00000011B TRANSMITTER RE 65 USR.TXR EQU 00000011B TRANSMITTER RE 67 USR.TX XTEXT CONSLX	63X USR.TXE EQU 00000100B FARITY ERGR 65Y USR.TXE EQU 00000100B TRANSMITTER EN 64X USR.TXR EQU 0000001B TRANSMITTER RE 65 USR.TXR EQU 00000001B TRANSMITTER RE 65 USR.TXR EQU 000000001B TRANSMITTER RE 65 USR.TXR EQU 0000000001B TRANSMITTER RE 65 USR.TXR EQU 00000000000000000000000000000000000	000.020	USR.OE		OVERRUN ERROR
65X USR.KXR EUU 00000010B KECEIVER READY 64X USR.TXR EQU 0000001B TRANSMITTER RE 67 XTEXT CONSLX	SSX USR.TXR ENU 0000001B TRANSMITTER RE 657 USR.TXR ENU 0000001B TRANSMITTER RE 657 USR.TXR ENU 0000001B	SSX.USR.KXR ENU 0000001B RECEIVER READY 66X USR.TXR EQU 0000001B TRANSMITTER RE 67 CONSLX	65X USR.TXR EQU 0000001B RECEIVER READY 64X USR.TXR EQU 0000001B TRANSMITTER RE 67 XTEXT CONSLX	SSX USR.RXR EUU 0000001B 64X USR.TXR EQU 0000001B 7FANSMITTER RE 67 USR.TXR EQU 0000001B 7FANSMITTER RE 7 CONSLX	000.010	USR.PE		PARITY EKROR
66X USR.TXR EQU OOOOOOOIB TRANSMITTER RE	66X USR.TXR EQU OOOOOOOIB TRANSMITTER RE	66X USR.TXR EQU OOOOOOOIB TRANSHITTER RE 67 XTEXT CONSLX	66X USR.TXR EQU OOOOOOOIB TRANSMITTER RE 67 XTEXT CONSLX	66X USR.TXR EQU OOOOOOOIB TRANSMITTER RE 67 XTEXT CONSLX	\$00.000	LISE RXE	:	KECELVER READY
67 XTEXT CONSLX	67 XTEXT CONSLX	67 XTEXT CONSLX	67 XTEXT CONSLX	SONSLX TONSLX	000,000	USR. TXR		TRANSMITTER READY
					040,100		: ⊥×	
							(
							i	

HEATH H8 CONSOLE DRIVER SYSTEM I/O DRIVER		HEATH XBASM V1.1 06/21/77 17:50:42 01-APR-77 PAGE 3
		CONSL - SYSTEM CONSOLE AND 1/0 DRIVER.
	71X 72X * * 73X * *	CONSL IS A GENERAL FURFOSE CONSOLE DRIVER. IT IS A STANDARD PRODUCT USED IN ALL HEATH HB SOFTWARE (WHICH COMMUNICATES WITH
	•	30LE DEVICE).
	•	1) FORT ROUTINES. THESE ARE PLACED IN CONSOLE SO THEY HAVE THE SAME LOCATION IN ALL PRODUCTS. FORT ADDRESSES MAY BE CHANGED BY PATCHING THESE POINTMES
	•	
		2) THE CONSOLE DRIVER FACKAGE. THIS FACKAGE CONSISTS OF THREE ROUTINES: *RCHAR - READ A SINGLE CHARACTER
		*WCHAR - WRITE A SINGLE CHARACTER *PRSCI - PRESET CONSOLE AND TAPE HARTS.
		THE CONSOLE FACKAGE FROVIDES SOPHISTICATED SUFFORT FOR ITS CALLERS:
		INTERRUPT PROCESSING FOR INPUT CHARACTERS
	•	SPECIAL CONTROL CHARACTER PROCESSING.
	•	NOTE THAT IF THE CONSOLE PACKAGE IS USED BY ANY RUNNING
	•	ROUTINE, ALL ROUTINES RUNNING WITH IT MUST USE IT ALSO.
		THIS IS BECAUSE *CONSL* WILL PROCESS INPUT CHARACTERS AT Interrupt time, beating out any task-time routine which
		ATTEMPTS TO READ CHARACTERS.
	XX	SPECIAL CHARACTER FROCESSING.
	××	*CONSL* SUPPORTS 8 SPECIAL CHARACTERS;
	X	
		CIL-A USEK DEFINED CONTROL FLAGS. THESE CAN BE CHECKED CIL-B AI TASK TIME. OR THE USER PROGRAM CAN SET UP AN
	X:	INTERRUPT VECTOR WHICH IS ENTERE
	× ×	TIME, WHEN ANY OF THESE CHARACTE
	X80	CTL-0 TOGGLE DISCARDING OUTPUT CHARACTERS
	110X *	SUSPEND OUTPUT
	X.	CIL-W



		CONSOLE DRIVER A	
			ASSEMBLY CONSTANTS.
	*		
	116X 117X **	I/O FORT ADDRESS	ES•
000	118X 118X 119X IP. FIP	3720	CONSOLE DATA
000.372	120X OF.CDF	:	DATA OUT FORT
000,373	121X IP.CIS	EQU 3730	INPUT STATUS
000.373	122X OP.CIS		INFUL SIATUS
000,373	124X 1F.005 124X 0F.005	EQU 3730	CONSOLE OUTPUT STATUS OUT PORT
	125X		
000,370	126X IP.TDP		TAPE DATA IN FURI
000,371 000,371	128X IP.TSP 129X OP.TSP	EQU 3710 EQU 3710	TAPE STATUS IN PORT
· · · · · · · · · · · · · · · · · · ·			
	31X	*CSLCTL (CONSOLE	(CONSOLE CONTROL FLAG) BITS.
		TUCKETTETE XEETE	THE APARTY THE SEPTIMENTALE CONTRIBUTION OF THE SEPTIMENTALE CONTRIBUTION
	134X *	CHARACTERS ARE S	STRUCK.
	X 20 X	THEOR DAN DE CYN	AT 100 - 11 ME - 4
	* * * * * * * * * * * * * * * * * * *	CHARACTER ROUTINE IS	ENTERED (AT INTERRUPT TIME)
000.001	39X 40X	EQU	SUSPEND OUTPUT
000,002	141X CC.DMF 142X CC.DMF	EQU 02	DISCARD OUTPUT CTI-A
000.020	143X CC.CTLB	EGU	CTL-B
000.040	144X CC.CTLC	EQU	C1L-C
000,100	4 UX	E.W.O.	0.15-10

ETEATHKIT [®]

HEATH H8 CONSOLE DRIVER SYSTEM I/O DRIVER	VSOLE DRIVI ORIVER	 R					HEATH X8ASM VI.1 06/21/77 17:51:24 01-AFR-77 FAGE 5
		; 	47X **	:	FROGRAM	PROGRAM ENTRY POINTS.	
040,100	303 333	333 1	149X ENT	TRY	UNF UNF	CONF16 RESTART	PROGRAM RESTART ENTRY POINT PROGRAM RESTART ENTRY POINT
		- -	502X *	-	PORT RO		
		(ALL PRO CHANGEA	ALL FROGRAMS MUST USE CHANGEABILITY.	THESE ROUTINES, TO ALLOW FORT ADDRESS
040.106	333 372	ਜ ਜ ਜੋ	:#:	:	Z	IP, CDP	CONSOLE DATA IN
040.111		4 F4 F4 F	2. U.		OUT	OP.CDP	CONSOLE, DATA, OUT.
040.114	333 373	न न्यें न्यें र	: *	ISI	- Z	IP.CIS	CONSOLE INFUT STATUS IN
040.116. 040.117 040.121	323 373 311	A A A	65X 65X 45X 68X 88X 88X	ISO	OUT RET	0F.CIS	CONSOLE INPUT STATUS OUT
040.122	333 373	ਜ ਜ -	69X 70X \$C 71X	OSI	IN	IP, COS	CONSOLE OUTPUT STATUS IN
040.125	: :	ਜਜਜ	727 733 745 745	080	OUT RET	0F.C0S	CONSOLE OUTPUT STATUS DUT
040.130	333 370 311	तेनि ने	76X \$T	NIG	IN	IP.TDP	TAPE DATA IN
040.133		ਜ ਜ:ਜੌ	78X 79X \$1 80X	DOUT.	OUT	OF.TDF	TAFE DATA OUT
040.136	333 371	H H	:₩	SIN	IN RET	IP, TSP	TAPE STATUS IN
040.141	323 371	a a ā	-	SOUT	OUT RET	OF.TSP	TAPE STATUS OUT
		ā ā ř	188X ** 189X *		REMOTE	ENTRY POINTS FO	FOR CONSOLE DRIVER.
040.144	303 255 (040	₩	CHAR	J.W.	*RCHAR.	READ ONE CHARACTER
040.147	303 332 0	040	3	CHAR	JMP	*WCHAR.	WRITE ONE CHARACTER
646.155	?. 						

HEATHKIT	

HEATH H8 SYSTEM I	HEATH HB CONSOLE DRIVER SYSTEM 1/0 DRIVER	IVER					HEATH XBASM V1.1 06/21/77 17:51:48 01-AFR-77 FAGE 6
			197X * 198X * 199X	*	DATA AN	DATA AND BUFFERS.	
040.15	155 000 156		;₩ ;	INBUF	DB DS	0 m	INPUT BUFFER COUNT TYPE AHEAD BUFFER. FIRST BYTE = NEXT CHARACTER
000	035			*INBUFL !	Eau	*-*INBUF-2	MAX LENGTH OF BUFFER
				*	CONTROL	CHARACTER TABLE	***
			*** 508X 508X		DB.	CHARAMASKA	
					IF.CHAR.	*C\$LÇTL. =	(*CSLCTL.;AND: MASK) ;DK. VALUE
040	4000	210		*CSIR	D.R.	000,3279,0000	
040.	22002.	220	214X		8	028,1570,2200	CIL-E
040	25 003	240 040 000	215X		80 80	038,1378,2408	CTL-D
040	3 017	002	217X		DB.	170,1770,0020	C1L-0
040 . 24	236 020 175 241 021 176	000	218X 219X		DB:	210,1750,0000	CTL-P CTL-0
040	14 023	001	220X		DB	230,1760,0010.	:
040	7 177		221X		DB	1770	END OF TABLE
				*	- DISD\$	ADDRESS OF INTE	INTERRUPT TIME CONTROL CHARACTER PROCESSOR.
			225X *		\$CSIC C	ONTAINS THE ADDA	*CSIC CONTAINS THE ADDRESS OF THE ROUTINE CONSL ENTERS WHENEVER A
					ARE SET	/CLEARED IN \$CSL	CTL, AND THE ROUTINE IS ENTERED AT INTERRUPT
					TIME.A THE BIT	FIER IHE USER RUS FROM *CSLCTL,	OUTINE.HAS.COMPLETED.FROCESSING.IT.SHOULD.CLEAR AND RETURN TO THE SYMBOL '\$RET'. IF INTERRUPT
					Processing. 18		.NOT. REQUIRED. LEAVE.THE. \$RET. ADDRESS.IN. \$GSIG
040.2	250 110 040			*CSIC	ĎΨ	*RET	ADDRESS OF USER ROUTINE FOR CTL-A THROUGH CTL-D.
040					Q.B.	ø	CONSOLE, CONTROL, BYTE
040.253	253 000			COLNO 1	08 108	0 8	CONSOLE CURSOR POSITION
***************************************	:		•		100	# TNB IF /10000	
• 000	000		239X	-	P.F.J ERRNZ	*/1000A	ALL DATA MUST BE IN SAME PAGE
						Q	
						ľ	

#IFATHKIT®

2433 ** ** ** ** ** ** ** ** ** ** ** ** *	243 ** * * * * * * * * * * * * * * * * *	243 ** * * * * * * * * * * * * * * * * *	243X ** 245X * 245X * 245X * 245X * 245X * 245X * 247X * 247X * 247X * 247X * 252X * 252X * 252X * 247X * 247		, 100 - // 5 E 15 OT+170+/1
245X * \$RCHAR IS CALLED TO READ 246X * ENTRY 246X * ENTRY 248X * ENTRY 249X * ENTRY 255X * CHAR & PUSH 249X * CHAR & P	245 * * * * * * * * * * * * * * * * * * *	245 * * * * * * * * * * * * * * * * * * *	244X * 244X * 244X * 246X * 246X * 246X * 248X * 249X * 259X * 259X * 251X * 25	- READ	RACTER.
240X * ENTRY NONE 248X * ENTRY NONE 248X * EXIT (A) = CHAR (PARI 248X * EXIT (A) = CHAR (PARI 252X * * * * * * * * * * * * * * * * * * *	2450.** 2470.** 2490.** 2490.** 2490.** 2490.** 2490.** 2490.** 2490.** 2490.** 2490.** 2490.** 2490.** 2490.** 2490.** 2490.** 2500.** 2510.** 2520.** 2540.** 2540.** 2540.** 2550.** 2540.** 2550.*	245. * ENTRY NONE 249X * ENTRY NONE 249X * EXIT (4) = CHAR (PARI 249X * USES A:F EXIT 341 155.040 255X * RCHAR! MOV 312 261 040 255X * RCHAR! MOV 32 255X * RCHAR! 32 261X MOV 32 32 310 040 265X MOV 32 32 32 32 32 32 32 32 32 32 32 32 32 3	240X * 240X * 248X * 248X * 248X * 249X * 250X 250X 251X *RCHAR: 176 253X *RCHAR: 176 255X * 312 261 040 256X 312 261 040 256X		A CONSOLE
248X * EXIT (A) = CHAR (PARI 2498X * LSES A)F CHAR (PARI 255X * CHAR, PUSH H H * INBUF C	248 * EXIT (4) = CHAR (PARI 250	248 * EXIT (4) = CHAR (PARI 2498 * EXIT (4) = CHAR (PARI 2508 * EXIT (4) = CHAR (PARI 2508 * EXIX * CHAR (PARI 2508 * EXI	248X * 249X * 249X * 249X * 249X * 250X 250X 250X 250X 250X 250X 250X 250X	NON	
250X 345 255X \$641, 155, 040, 253X \$72, \$72, \$72, \$72, \$73, \$74, \$74, \$74, \$74, \$74, \$74, \$74, \$74	250X 345 252X \$KCHAR, PUSH H 041, 155, 040, 253X 176 217 252X \$KCHAR, PUSH H 217 252X \$KCHAR, PUSH H 312 261 040, 255X 325 325 325 325 326 326 326 327 326 327 327 327 328 328 328 329 329 329 329 329 329 329 329 329 329	250X 345 252X \$RCHAR, PUSH H 941 156 253X RCHAR HOV 27 251X RCHAR HOV 27 252X RCHAR HOV 27 255X JZ \$KCHAR HOV 26 254X \$RCHAR HOV 36 255X DI 36 259X PUSH DI 36 259X PUSH DI 36 259X PUSH DI 37 251 0.40 255X HOV HI 37 310 0.40 255X RCHAR GI 37 32 310 0.40 255X RCHAR GI 36 0.40 255X RCHAR BI 37 32 310 0.40 255X RCHAR BI 36 0.40 255X RCHAR BI 37 32 310 0.40 255X RCHAR BI 37 32 310 0.40 255X RCHAR BI 36 0.40 255X RCHAR BI 37 32 310 0.40 255X RCHAR BI 37 32 32 0.40 255X RCHAR BI 37 32 28 0.40 275X RCHAR BI 37 32 28 0.40 275X RCHAR BI 37 33 31 0.40 275X RCHAR BI 37 32 28 0.40 275X RCHAR BI 37 32 28 0.40 275X RCHAR BI 37 33 31 0.40 275X RCHAR BI 37 33 31 0.40 275X RCHAR BI 37 32 28 0.40 275X RCHAR BI 37 32 28 0.40 275X RCHAR BI 37 32 28 0.40 275X RCHAR BI 37 33 31 0.40 275X RCHAR BI 37 34 28 28 28 28 28 28 28 28 28 28 28 28 28	250X 345 251X 041 155 040 253X 176 254X \$RCHAR1 247 312 261 040 256X 312 261 040 256X	(A) = A,F	TY BIT CLEARED)
345 252X \$RCHAR, FUSH H H H H H H H H H H	345 345 346 346 347 346 347 348 348 348 348 348 348 348 348 348 348	345 346 347 348 348 341 345 348 348 348 348 348 348 348 348 348 348	345 252X \$KCHAK. 241,155,040, 253X 176, 255X 312,261,040, 256X 373,373		
041 155 040 253X LXI HY\$INBUF 176 255X AND APA 312 261 040 256X JZ \$KCHARI 363 255X BUZ 365 256X DLX 365 260X DCR MOV 043 265X MOV DrM 043 265X MOV DrM 043 265X MOV DrM 376 173 264X CPI 1730 376 173 264X CPI 1730 376 173 266X CPI 1410 377 265X MOV APA 378 310 040 265X JNC FII 1410 378 260X KCHARA EQU 378 270X \$KCHARA EQU 378 270X \$KCHARA DR D 378 270X \$KCHARA D 378 270X \$KCHARA DR D 378 270X \$	941 155 040 253X *RCHARI MOV A*M 217 261 040 255X ARCHARI MOV A*M 312 261 040 256X JZ *RCHARI 325 259X PUSH D 326 220X NOV D*M 326 220X NOV B*M 327 261X NOV A*M 328 310 040 265X URCHARA EBU **CHAR3 328 310 040 265X B*CL*S 326 040 226X B*CL*S 327 310 040 265X B*CL*S 328 311 040 265X B*CL*S 341 226X B*CHARA EBU **M 4 M 4 M 4 M 4 M 4 M 4 M 4 M	941.155.040 253X *RCHARI MOV A*M 247 312 261 040 256X JZ *RCHARI 363 258X DI *RCHARI 364 255X DI *RCHARI 365 258X DI *RCHARI 365 264X MOV D*M 376 173 264X MOV D*M 376 173 264X MOV A*M 377 1040 265X MOV A*M 378 310 040 265X MOV A*M 378 325 040 2573X *RCHARA DI *RCHAR3 043 311 040 273X *RCHARA DI *RCHAR3 043 311 040 273X *RCHARA DI *RCHAR3 043 311 040 273X *RCHAR3 EI 378 311 040 273X *RCHAR3 EI 378 325 040 273X MOV A*M 365 273X *RCHAR3 DI *RCHAR3 378 325 040 273X MOV A*M 365 273X *RCHAR3 DI *RCHAR3 378 328 329 289X *RCHAR3 EI 379 289X *RCHAR3 EI 371 288X *RCHAR3 EI 372 288X *RCHAR3 EI 373 311 288X *RCHAR3 EI 374 288X *RCHAR3 EI 375 288X *RCHAR3 EI 377 288X *RCHAR3 EI 378 378 378 378 378 378 378 378 378 378	041 155 040 253X 176 254X \$KCHAK1 247 255X 312 261 040 256X	I	SAUE (HL)
247 261 040 255X ANA ANA ANA ANA ANA ANA ANA ANA ANA AN	247. 263 ANN ANN AND AND AND AND AND AND AND AND	247. 261 040 256X JZ \$\frac{2}{2}\$\frac{2}	247 251 040 255X 312 261 040 256X 373 262X	H, & INBUF	(HL) = ADDRESS OF CHARACIER FOINTER (A) = COUNTER
363 259X DI	363 2587	363 259X DI DI DEE DI	77.7.	*RCHAR1	WAIT FOR INTERRUPT TO READ CHARACTER
0.55 0.56 0.57 0.55	0.55 0.65 0.65 0.65 0.63	265 265 126 261 126 261 127 126 263 100 376 376 376 376 376 377 378 379 370 360 370 360 370 370 371 372 373 373 373 373 373 373 373	303 100	6	INTERLOCK SEQUENCE
126 261X MOV D+M 043 266X INX H MOV G+M 376 173 264X MOV G+M 376 173 264X MOV G+M 376 173 266X MOV G+M 376 141 266X JNC \$RC1.5 376 141 266X JNC \$RC1.5 326 040 265X JNCHARA EQU 365 270X \$RCHARA EQU 365 270X \$RCHARA EQU 365 270X \$RCHARA DCR D 372 325 040 273X RCHARA DN H MAN H	126 261X MOV D,M 127 262X INX H 1730 376 173 264X CPI 1730 372 310 040 265X JNC \$RC1.5 376 141 266X CPI 1410 376 141 266X CPI 1410 376 141 266X CPI 1410 377 325 040 268X \$RCHARA EQU \$*C1.5 272 \$	126 261X MOV D'M 1043 266X INX H 376 173 264X CPI 1730 322 310 040 265X JNC \$FC1.5 326 040 265X JNC \$FC1.5 326 040 268X SUI 400 325 310 040 268X SUI 400 326 040 268X SUI 400 327 XFCHARA EQU *-1 365 270X \$FC1.5 FUSH FSW 043 272X \$FCHAR2 DCR D 372 325 040 273X MOV H 053 275X MOV H 054 275X MOV H 055 275X MOV H 057 275X MOV H 058 275X MOV H 059 275X MOV H 051 275X MOV H 052 275X MOV H 053 311 040 279X JMP \$FCHAR2 361 288X FCHAR3 EI 362 288X FCHAR3 EI 363 288X FCHAR3 EI 364 288X FCHAR3 EI 365 288X FCHAR3 EI 367 288X FCHAR3 EI 368 288 288 FCHAR3 EI 368 FCHAR3 EI 368 FCHAR3 EI 368 FCHAR3 EI 368 F	245 045 260X	3 E	DECREMENT COUNT
176 376 177 376 177 376 173 376 376 376 376 376 376 377 376 377 377	176 376 177 376 177 376 376 377 377 378 378 378 378 378 378 378 378	176 376 177 376 177 376 377 377 377 377 377 377 377 377 3	126 261X	E.	(D) = COUNT-1
322 310 040 265X JNC \$FC1.5 376 141 266X JC FI 1410 326 040 267X JC FI 1410 326 040 268X SCHARA EQU *-1 365 270X \$FC1.5 FUSH FSW SCHAR3 025 270X \$FC1.5 FUSH FSW HCHAR3 043 272X \$FCHAR2 DCR D STAX MOV A,M HCHAR3 043 275X MOV M,A HCHAR3 043 275X MOV M,A HCHAR3 043 277X MOV M,A HCHAR3 043 278X MOV M,A HCHAR3 043 278X MOV M,A HCHAR3 304 279X JMP \$FCHAR2 280X JMP \$FCHAR2 280X JMP \$FCHAR3 361 283X FOP HCT HCT HCT A311 2883X FOP HCT HCT A311 2883X FOT HCT HCT HCT A311 2883X FOT HCT HCT HCT A311 2883X FOT HCT HCT HCT A311 2883X FOT HCT HCT A311 2883X FOT HCT HCT HCT HCT A311 2883X FOT HCT HCT HCT HCT A311 2883X FOT HCT HCT HCT HCT A311 2883X FOT HCT HCT HCT HCT HCT HCT HCT HCT HCT HC	322 310 040 264X UNI \$FC1.5 376 113 376 141 266X CPI 1410 322 310 040 267X UC \$FC1.5 326 040 267X UC \$FC1.5 326 040 267X UC \$FC1.5 326 040 267X SCHARR BUJ 400 365 270X \$FCHARR BUJ FSW 572 372 325 040 273X UN	322 310 040 265X UNC \$FC1.5 376 141 322 310 040 265X UC FI 1410 326 040 268X \$FCHARA \$UI 400 326 040 268X \$FCHARA \$UI 400 326 040 268X \$FCHARA \$UI 400 327 000 \$FC1.5 PUSH \$FCHAR3 025 272X \$FCHAR2 DCR D 043 272X \$FCHAR2 DCR D 053 225 040 273X DCX H 100	176 176 176 176 176 176	A*X	(A) = READ CHARACTER BEETT TO NOTE CASE
376 141 266X CPI 1410 332 310 040 267X JC \$RC1.5 269X \$RCHARA EQU #-1 269X \$RCHARA EQU #-1 365 270X \$RC1.5 PUSH FSW 271X PUSH FSW 372 325 040 274X JM HR 176 275X BCHAR2 DCR DCR DCR 176 275X BCHAR2 DCR DCR DCR 176 275X BCHAR2 DCR DCR HR 167 275X BCHAR2 DCR HR 167 275X BCHAR3 BCR 167 275X BCHAR3 EI 167 276X BCHAR3 EI 167 279X JNY HR 167 279X JNY HR 167 279X JNY HR 167 279X JNY HR 167 279X JNY HR 168 279X JNP FSW 281X \$RCHAR3 EI 361 283X POP H 361 283X POP H 361 283X POP H 361 283X POP H	376 141 266X CPI 1418 332 310 040 267X JC \$FC1.5 326 040 269X \$FCHARA EQU #-1 365 269X \$FCHARA EQU #-1 365 270X \$FC1.5 PUSH FSW 025 271X PUSH FSW 043 272X \$FCHAR2 DCR D 053 275X PUSH FSW 063 275X PUSH FSW 176 275X PUSH FSW 176 275X PUSH FSW 177 275X PUSH FSW 187 275X PUSH FSW 187 275X PUSH FSW 187 275X PUSH PUSH 187 275X PUSH PUSH 187 275X PUSH PUSH 187 289X PUSH PUSH 188 341 289X FETH RET	376 141 266X CPI 1410 332 310 040 267X UC #RC1.5 326 040 269X #RCHARA EUU #-1 269X #RCHARA EUU #-1 365 270X #RC1.5 FUSH FSW 025 272X #RCHAR2 DCR D 372 325 040 273X UN H H 043 275X MOV A H 053 275X MOV A H 053 275X MOV A H 053 275X MOV A H 053 275X MOV B H 303 311 040 273X UMP #RCHAR2 361 280X UMP FSW 373 281X #RCHAR3 EI 361 283X POP H 311 285X RET RET	376 1/3 264X 322 310 040 265X	1/34 \$RC1 • 5	SEE IT COMEN CASE
326 040 268X \$CHARA EQUI 400 326 040 268X \$RCHARA EQUI *-1 270X \$RC1.5 FUSH FSU 2570X \$RC1.5 FUSH FSU 372 325 040 273X RCHAR2 DCR 176 273X RCHAR2 DCX HRCHAR3 176 275X MOV H.A HRCHAR2 275X MOV H.A HRCHAR2 275X MOV H.A HRCHAR2 275X MOV H.A 303 311 040 279X JMP \$RCHAR2 280X SO3 311 040 279X JMP FSU 351 281X \$RCHAR3 EI 283X RCHAR3 EI 283X RCHAR3 EI 284X RCHAR3 341 284X RET	326 040 268X 8CHARA EQU 400 326 040 268X 8CCHARA EQU 8-1 270X 8CC1.5 PUSH FSU 852 270X 8CC1.5 PUSH FSU 025 270X 8CCHAR2 DCR D 870 H H H H H H H H H H H H H H H H H H H	326 040 268X SUI 400 326 040 268X \$FCHARA EQUI 400 326 040 268X \$FCHARA EQUI 4-1 365 270X \$FCHAR2 DCR B BOOK BY BOOK BY BOOK BY BOOK BY BOOK BY BOOK BY BOOK BOOK	376 141 266X		TO NOT 1 DUED CASE
269X \$RCHARA EQU *-1 365 270X \$RC1.5 FUSH FSW 025 272X \$RCHAR2 DCR D 372 325 040 273X MOV A.M 176 273X MOV A.M 043 274X MOV A.M 053 311 040 279X MOV M.A 303 311 040 279X JMP \$RCHAR2 361 281X \$RCHAR3 EI 361 283X POP D 361 284X POP H 311 285X RET	269X \$RCHARA EQU *-1 365	269X \$RCHARA EQU *-1 365	352 310 040 20/A		AAKE LIPPER
365 270X \$RC1.5 PUSH FSW 271X 270X \$RC1.5 PUSH FSW 025 272X \$RCHAR2 DCR D \$722 325 040 273X 1NX H	365 270X \$RC1.5 PUSH FSW 025 272X \$RCHAR2 DCR D 372 325 040 273X JM \$RCHAR3 043 274X INX H 176 275X MOV A,M 053 276X MOV M,A 043 277X MOV M,A 044 A MOV M,A 045 277X MOV M,A 047 277X MOV M,A 048 277X MOV M,A 049 277X MOV M,A 049 277X MOV M,A 041 277X MOV M,A 043 277X MOV M,A 043 277X MOV M,A 044 A M,A 045 277X MOV M,A 047 277X MOV M,A 048 277X MOV M,A 049 277X MOV M,A 040 277X M	365 270X \$RC1.5 PUSH FSW 025 272X \$RCHAR2 DCR D 372 325 040 273X	269X \$RCHARA		ZEROED FOR LOWER CASE
312 325 325 040 273X \$RCHAR2 DCR D \$RCHAR3 315 343 311 040 279X MDU MA \$RCHAR2 255X MDU MA \$855 363 341 040 279X JMP \$RCHAR3 255 303 311 040 279X JMP \$RCHAR3 255 373 281X \$RCHAR3 E1 PSW 282X PDP D 336 341 283X PDP H 331 283X PDP H 331 285X PDP H 331 285X RET RET	312 325 340 273X \$RCHAR2 DCR D \$FCHAR3 315 043 325 040 273X MOV A+M 316 176 275X MOV A+M 321 043 275X MOV M+A 321 043 275X MOV M+A 322 303 311 040 279X JMP \$RCHAR2 280X 281X \$RCHAR3 E1 282X 73 281X \$RCHAR3 E1 283X POP PSM 330 341 284X POP H 331 311 285X RET	312 372 325 040 273X \$KCHAR2 DCK D \$KCHAR3 312 372 325 040 273X DK	365 270X \$RC1.5		SAVE 11
312 372 325 040 273X JM \$KCHAK3 315 043 272X INX H 316 053 275X MOV H,A 320 167 277X MOV H,A 321 043 279X JMP \$KCHAK2 322 303 311 040 279X JMP \$KCHAK2 322 303 311 040 279X JMP \$KCHAK2 325 373 281X \$KCHAR3 E1 282X POP FSW 326 341 283X FOP H 331 311 285X KET	312 372 325 040 273X JM \$KCHAK3 315 043 272X MDV A H 316 176 275X MDV A H 320 167 275X MDV H H 321 043 277X MDV H H 321 043 311 040 279X JMP \$KCHAK2 280X JMP \$KCHAK2 325 373 281X \$KCHAK3 E1 F0P FSW 326 341 285X F0P H 331 311 285X RET	312 372 325 040 273X JM \$KCHAK3 315 043 272X JM H 316 275X MDU A,M 321 043 275X MDU H,A 321 043 275X MDU H,A 322 303 311 040 279X JMP \$KCHAK2 326 373 281X \$KCHAK3 EI 326 341 282X FOP FSW 330 341 283X FOP H 331 285X KET	311 025 272X *RCHAR2		MOVE OTHERS DOWN IN QUEUE
315 176 275X MOV A,M 317 176 275X MOV H,A 317 053 275X MOV H,A 4,M 320 167 275X MOV H,A 4,M 321 043 275X MOV H,A 4,M 4,A 322 303 311 040 279X JMP \$RCHAR2 280X 280X 280X FOP FSW 526 341 282X FOP H H H H H H H H H H H H H H H H H H H	315 174	315 174	312 372 325 040 273X		NO MORE
317 053 276X DCX H 320 167 277X MOU H'A 321 043 2178X JNY H 322 303 311 040 279X JMP \$RCHAR2 325 373 281X \$RCHAR3 EI FSW 326 361 282X PDP PSW 327 321 283X POP H 331 311 285X RET RET	317 053 276X DCX H 320 167 277X MOV H'A 321 043 277X MOV H'A 322 043 21040 279X JMY \$RCHAR2 280X JMY \$RCHAR2 325 373 281X \$RCHAR3 E1 FSW FOP D SW H 331 311 285X RET H EXIT: (A 331 311 285X RET	117 053 276X DCX H 150 167 277X MDV M+A 152 043 21040 279X JMY \$RCHAR2 152 303 311 040 279X JMP \$RCHAR2 152 373 281X \$RCHAR3 E1	315 043 274X 316 176 275X	τ. Έ	
320 197 278X	320 197 279X 100 H7H 321 343 311 040 279X JMP \$RCHAR2 325 373 283X \$RCHAR3 EI FSW B B B B B B B B B B B B B B B B B B B	320 197 278X 100 H7H 321 043 321 040 279X JMP \$RCHAR2 280X JMP \$RCHAR2 326 343 283X \$POP PSW H RESTORE 330 341 283X \$POP H H RET H R	317 053 276X	-	
322 303 311 040 279X JMP \$RCHAR2 280X 281X \$RCHAR3 EI PSW RESTORE 325 351 282X PDP D D 327 321 283X PDP H 330 341 284X RET HET EXIT: (A	322 303 311 040 279X JMP \$RCHAR2 280X 225 3373 282X \$FOF PSW 227 321 283X FOF H 330 341 284X FOF H 331 311 285X RET RET	322 303 311 040 279X JMP \$RCHAR2 280X 225 3573 282X \$RCHAR3 EI RESTORE 326 341 283X POP H 330 311 285X RET RET EXIT: (A	320 167 277X 321 043 528X	Œ.I	
280X 325 373 281X *RCHAR3 EI RESTORE 326 341 283X FOP D 330 341 284X POP H 331 311 285X RET HET EXIT: (A	225 373 280X 326 341 283X FCHAR3 EI FSW 327 321 283X FOF D 330 341 284X POP H 331 311 285X RET H EXIT: (A	280X 325 373 281X *RCHAR3 EI RESTORE 326 321 283X FOP D 330 341 284X FOP H 331 311 285X RET RET EXIT: (A	322 303 311 040 279X	\$KCHAR2	
326 361 282X POP PSW 527 321 283X POP D 330 341 284X POP H 331 311 285X RET RET (A	326 361 282X POP PSW 327 321 283X POP D 330 341 284X POP H 331 311 285X RET RET (A	326 361 282X POP PSW 327 321 283X POP D 330 341 284X POP H 331 311 285X RET H EXIT: (A	280X 325 373 281X \$RCHAR3		
330 341 285X POP H 331 311 285X RET EXIT: (A)	330 341 284X POP H 331 311 285X RET H EXIT: (A)	330 341 2842 FOP H 331 311 285X RET RET EXIT: (A)	326 361 282X	PSU	•
334344			330 341 2653 330 341 2664X 2664X 2664X 2664X 2664X	a I	

HEATHKIT

1987 1987 1987 1988	287X ** 289X ** 289X ** 289X ** 289X ** 299X *	- WRITE SINGLE CHARACTER.
289X # \$40HR IS CALLED TO OUTPUT A SINGLE CHARACTER. 290X # ENTIR NAME 290X # ENTIR	288X ** 289X ** 290X ** 291X ** 292X *	TS CALLED TO DITTELL A STAGLE
290. * ENTRY (A) = CHGRGCTER 292. * ENTRY (A) = CHGRGCTER 292. * ENTRY (A) = CHGRGCTER 292. * ENTRY (B) = CHGRGCTER 293. * SAS SAS SAS SAS SAS SAS SAS SAS SAS S	290X * 291X * 292X * 293X * 294X 365 296X * 294X 332 333 040 299X 332 333 040 299X 332 333 040 299X 331 302X 332 333 040 300X 303 300X 304X 311 305X * 306X	110000
255 278 4 CALL 277 278 240 278 4 CALL 278 278 4 CALL 278 240 278 4 CALL 278 240 278 4 CALL 278 240 278 4 CALL 278 278 278 27	365 292X ** 293X ** 295X ** 29	(A) = CHARACTER
3.65 252 494 257 \$40CHAR, PUSH FSW SAVE CHARACTER CHARACTER CHAPA \$525 494 257 \$40CHAR, PUSH FSW CHAPA \$6SUT.	294X 365 296X \$WCHAR. 072 252 040 298X 332 333 040 298X 332 333 040 298X 332 354 040 300X 365 302X 361 305X 362 354 040 300X 365 007 311X 312X \$WCHAR2 315 122 040 315X \$WCHAR2 315 122 040 315X \$WCHAR2 315 122 040 318X 312X \$325X \$X 361 312X \$325X \$X 361 323X \$X 362 354 040 320X 361 317 040 320X 333X \$X 33X \$X 3X	:
3.6.5 2.6.4 auchtaki, Publi Fed. Grace Character Control. 3.6.2 2.6.4 auchtaki, Publi Ferri Control. 3.6.2 2.6.4 auchtaki, Publi Ferri Control. 3.6.2 2.6.4 auchtaki, Luba Ferri Collinia. 3.6.2 2.6.4 auchtaki, Collinia. 3.6.2 2.6.4 auchtaki, Collinia. 3.6.2 2.6.4 auchtaki, Collinia. 3.6.3 3.6.4 auchtaki, Collinia. 3.6.4 auchtaki, Collinia. 3.6.4 auchtaki, Collinia. 3.6.5 auchtaki, Auchtaki, Collinia. 3.6.5 auchtaki, Auchtaki, Collinia. 3.6.5 auchtaki,	365 296X \$WCHAR. 072 252 040 297X \$WCHAR. 037 298X 332 333 040 297X \$WCHAR. 302X 303X 341 306X 341 306X 341 306X 341 306X 342 354 040 308X 065 309X 365 309X 365 312X 312X 312X 315 122 040 315X \$WCHAR. 315 122 040 315X \$X 312X 313X \$X 312X 311X 315 122 040 315X \$X 312X 311X 315 122 040 315X \$X 311X 315 122 040 315X \$X 311X 315 117 040 320X 333X \$X 33X \$X 33	
972, 252, 940, 227X, swicherl, LDA, 955,LCT, CHECK CONTROL. 932, 353 940, 297X RARN Z CC. LIDT AND TO WAIT 937, 297X RARN Z CC. LIDT AND TO WAIT 937, 354, 940, 303X RARN Z CC. LIDT AND TO THE CHARACTER, ECHO RELL. 937, 354, 940, 303X RARN Z CC. LIDT AND TO THE CHARACTER, ECHO RELL. 937, 340, 340, 340, 340, 340, 340, 340, 340	072 252 040 297X \$WCHART 037 333 040 300X 037 298X 341 302X 341 302X 341 302X 341 302X 341 302X 341 302X 342 354 040 303X 342 354 040 315X 312X 312X 312X 312X 312X 312X 312X 312	PSW
327 298X ERRAZ CC + HLD - 1 322 333 404 304 305 30	037 332 333 341 365 365 365 365 365 365 365 365	*CSLCTL
352 353 040 350X	332 333 040 300X 341 301X 341 302X 341 302X 341 300X 341 300X 300X 300X 300X 300X 300X 300X 300X	CC.HLD-1
937 994X RAR CC. DMP-2 342 354 949 393X WRAK CC. DMP-2 342 354 949 339X RAR SERVIC SERVICE SUSCARD CHAR 344 303X WC \$\$\text{4}\text{CHARACTER}\$ \text{CCT} \text{DISCARD} \text{CHARACTER}\$ \text{342} \text{343} \text{344}	037 362 362 362 362 362 363 311 365 365 367 368 368 368 368 368 368 368 368	\$WCHAR1
322, 354, 949, 302A 341, 304X 341, 304X 341, 304X 354, 304X 3504X 3504X 3504X 3504X 3504X 3504X 3504X 3504X 365, 305X 365, 305X 3704X 371, 305X 372, 305X 373,	322 354 040 303X 341 305X 306X 306X 306X 306X 306X 309X 309X 309X 309X 311X 311X 312X 313X 313X 313X 313X 314X 315X 316X 316X 316X 316X 317X 317X 317X 317X 317X 317X 311X 31	:
341 304X 504X 5	361 311 305X 306X 306X 308X 308X 308X 308X 311X 312X 322X 32X 3	COLUMNIA SUCHARO AN TO
306.5 306.5 306.5 306.5 308.7 308.8 308.8 308.8 308.8 311.0 312.0 313.8 313.8 313.8 314.8 315.122.0 315.2 315.8 316.8 317.8 317.8 318.8 31	306X 306X 307X ** 308X 308X 308X 311X	PSW CTL-0 IN EFFECT,
965 3098 x	065 308X 076 007 310X 365 311X 312X 313X * 313X * 313X * 314X 315 122 040 315X * 322 354 040 318X 361 11 040 320X 361 11 040 320X 322X * 322X * 32X *	LOWN TYPE-AHEAD
0.0 0.0	365 007 310X 311X 311X 311X 311X 311X 311X 311X 311X	REMOVE LAST CHARACTER INPUT TO MAKE
345 311X PUSH FSW STORE CHARACTER. 312X TYPE CHARACTER. 313X TYPE CHARACTER. 313X TYPE CHARACTER. 314X TYPE CHARACTER. 315 122 040 315X SWCHAR2 CALL FORSI 316X SWCHAR2 CALL FORSI 322 334 040 318X JAX JAX JAX JAX JAX JAX JAX JAX JAX JA	365 311X 312X 312X 313X * 314X 314X 322 322 324 321 322 322 322 322 322 322 322	
312 x	312X * 313X * 313X * 314X * 314X * 316X * 311X * 31	STORE
315 122 040 314X 314X 315 122 040 315X \$\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\	315.122.040.315X \$WCHAR2.322.354.040.318X.34X.341.040.320X.323X.** 361.310.040.320X.** 362.354.040.320X.** 362.354.** 322	CHARACTER.
937 316X ERRNZ USR.TXR-1 322 354 040 318X RAR swCHAR2 WAIT FOR ROOM 341 319X JAP FSW DOUT 341 319X JAP FSW DOUT 351 111 040 320X ** \$FRSCL - FRESET CONSOLE DRIVERS. 323X \$ \$FRSCL - FRESET CONSOLE DRIVERS. 324X \$ \$FRSCL - FRESET CONSOLE DRIVERS. 325X * \$FRSCL - FRESET CONSOLE DRIVERS. 3	332 354 040 318X 361 111 040 320X 363 111 040 320X 322X ** 322X ** 332X ** 333X ** 333	\$C081
0372 317X RAR \$WUCHAR2 WAIT FOR RODM 322 354 040 318X JNC \$WUCHAR2 WAIT FOR RODM 322 354 040 318X JNC \$WUCHAR2 WAIT FOR RODM 322 354 040 318X DNC FSW 323 ** \$FRSCL - FRESET CDNSOLE DRIVERS, 323 ** \$FRSCL - FRESET CDNSOLE DRIVERS, 324 * \$FRSCD IS CALLED TO FRESET TERMINAL OFFRAITONS, 325 ** IT ASSUMES THAT THE SAME FORT IS BEING USED FOR CONSOLE INFUT 326 ** THE INFUT FORT ROUTINES, 327 ** THE INFUT FORT ROUTINES, 329 ** ENTRY NONE 329 ** EXIT NONE 330 ** FRSCL, MUI A,2010, 333 ** CALL \$TSOUT 315 147 040 335 ** CALL \$TSOUT 315 147 040 335 ** CALL \$TSOUT 315 141 040 335 ** CALL \$TSOUT	937 364 363 111 040 320 322 322 322 322 324 324 325 326 326 326 326 327 327 328 329 329 329 329 329 329 329 329	USR.TXR
342 354 040 318X 318X 318X 318X 318X 318X 318X 318X	362 354 040 3185 363 111 040 320X 322X ** 323X ** 322X ** 322	100L FF 4 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
303 111 040 320X JMF \$CDOUT OUTPUT AND EXIT 322X **	303 111 040 320X 322X ** 323X ** 324X * 325X * 326X * 326X * 326X * 322X * 332X * 342X * 34	HAKA WAII FUR
322X ** *FRSCL - FRESET CONSOLE DRIVERS. 324X * *FRSCL - FRESET CONSOLE DRIVERS. 325X * IT ASSUMES THAT THE SAME FORT IS BEING USED FOR CONSOLE INFUT 325X * IT ASSUMES THAT THE SAME FORT IS BEING USED FOR CONSOLE INFUT 325X * THE INFUT FORT ROUTINES. 325X * THE INFUT FORT ROUTINES. 325X * EXIT NONE 325X * EXIT NONE 325X * EXIT NONE 332X * USES A.F 333X * USES A.F 333X * CALL *CISO GUARANTEE OUT OF MODE-SET 315 117 040 335X CALL *TSOUT FORCE INTO MODE-SET 315 141 040 333X CALL *TSOUT FORCE INTO MODE-SET	322X ** 323X * 323X * 326X * 326X * 326X * 329X * 329X * 332X * 342X * 3	007
323X *	323X * 324X * 324X * 325X * 325X * 325X * 329X * 329X * 339X * 331X * 332X * 332X * 332X * 332X * 332X * 335X * 33	- PRESET
325X * IT ASSUMES THAT THE SAME FORT IS BEING USED FOR CONSOLE INFUT 326X * CONSOLE OUTFUT, RECAUSE THE CONFIGURATION IS DONE THROUGH 327X * THE INFUT FORT ROUTINES. 329X * ENTRY NONE 329X * ENTRY NONE 330X * LOSES A,F 331X * USES A,F 333X 076 201 334X *FRSCL, MVI A,2010 315 117 040 335X CALL *TSOUT FORT CONFIGURATION IS DONE THROUGH 4C1SO 315 117 040 338X CALL *TSOUT FORT CONFIGURATION NODE-SET 315 141 040 339X CALL *TSOUT FORCE INTO MODE-SET	3255 * 3256 * 3256 * 3256 * 3257 * 3297 * 32	TO CALLED TO BEFORE TERMINAL OPERATIONS.
326X * CONSOLE OUTPUT, BECAUSE THE CONFIGURATION IS DONE THROUGH 327X * THE INPUT FORT ROUTINES. 329X * ENTRY NONE 329X * EXIT NONE 330X * LEXIT NONE 331X * USES A,F 332X 333X * CALL \$CONFIGURATION IS DONE THROUGH 332X 333X * CALL \$CONFIGURATE OUT OF MODE-SET 315 117 040 333X CALL \$TSOUT FORCE INTO MODE-SET 315 141 040 339X CALL \$TSOUT FORCE INTO MODE-SET	326X * 329X * 329X * 329X * 329X * 329X * 339X * 331X * 333X * 333X * 333X * 335X * 335X * 515 117 040 35X * 51	R CONSOLE INFUT
327X * THE INPUT FORT ROUTINES. 329X * ENTRY NONE 329X * ENTRY NONE 330X * ENTRY NONE 331X * USES A,F 333X 976,201 334X \$FRSCL, MVI A,2010 315,117 040 335X CALL \$TSOUT FORCE INTO MODE-5 315,117 040 339X CALL \$TSOUT FORCE INTO MODE-5	327X * 329X * 329X * 329X * 329X * 339X * 331X * 332X * 333X * 333X * 335X * 315 117 040 335X * 575	ONE THROUGH
329X * EXIT NONE 330X * EXIT NONE 331X * USES A,F 332X 332X 332X 333X 076, 201 334X *FRSCL, MVI A,2010 315 117 040 335X CALL *CISOT 076 100 335X MVI A,UCL, IR 315 141 040 339X CALL *TSOUT FORCE INTO MODE-5	359X ** 330X ** 331X * 332X * 332X * 335X 335X 335X 315 117 040 335X 335X 335X 335X 335X 335X 335X 335	INPUT FORT ROUTINES.
330X * EXIT NONE 331X * USES A,F 331X * USES A,F 332X 332X 332X 332X 332X 332X 333X 333	330X * 331X * 331X * 331X * 332X * 332X * 332X * 335X \$ 315 117 040 335X \$ 117 040 335X \$ 110 040 35X	:
331X * USES A,F 331X * USES A,F 332X 332X 333X 333X 335X 315 117 040 335X CALL \$CISO 60ARANTEE OUT OF 315 141 040 339X CALL \$150UT FORCE INTO MODE-5	331X * 332X	:
333X 976 201 315 117 040 335X 415 141 040 335X 6ALL \$150117 040 337X 6ALL \$15007 \$15 141 040 339X 6ALL \$15007 \$15007 \$15007 \$15007 \$15007 \$15007 \$15007 \$15007 \$15007 \$15007 \$15007 \$15007 \$15007 \$15007 \$15007	076 201 334X \$FRSCL. 315 117 040 335X 315 141 040 335X 076 100 337X 315 141 040 338X	
976 201 334X \$FKSCL, MVI A;2010 315 117 040 335X CALL \$CISO GUARANTEE OUT OF 315 141 040 336X CALL \$TGOUT 076 100 338X MVI A;0C1.1R 315 141 040 339X CALL \$TSOUT FORCE INTO MODE-5	076 201 334X \$FRSCL. 315 117 040 335X 315 141 040 336X 076 100 337X 315 117 040 338X	
315 141 040 336X CALL *ISOUT 076 100 337X MVI A.UCI.IR 315 117 040 339X CALL \$GISO 315 141 040 339X CALL \$TSOUT FORCE INTO MODE-S	315, 141, 040, 336X 076, 100 315, 117, 040, 338X 315, 141, 040, 338X	GUARANTEE OUT OF
076 100 337X MVI A.UCI.IR 315 117 040 339X CALL \$GISO. 315 141 040 339X CALL \$TSOUT	076 100 337X 315, 117, 040 338X	
315 117 040 339X	315.117.040338X	A, UCI, IR
513 141 040 557A CMLL #15001		
	313 141 C40 337A	



**	HEATHKIT®

SSS	CONSOLE INTERRUPT PROCESSOR. READ INTERRUPTS ENTER HERE FROM PAM-8. NONE (EI' AND 'RET' NONE SCISI USR.RXR SCISI H * SINBUFL A * M A
3555 * 356 * 356 * 357 *	FER HERE FROM PAM-8. ESERUE REGISTERS HAVE DATA FROM INFU SEE IF ROOM IN THE TROOM ROOM C) = ADDRESS OF CHAR FUT CHARACTER IM IT ORE IT SEQUENCES. ECIAL USER INTERRUPT
357 * ENTRY NONE 359 * * EXIT 'EI' AND 'RET' 359 * * EXIT 'EI' AND 'RET' 360	ESERVE REGISTERS HAVE DATA FROM INPU SEE IF ROOM IN THE ROOM ROOM IN IT ORE IT SEQUENCES. ECIAL USER INTERRUPT
359X * EXIT 'EI' AND 'RET' 360X * USES NONE NONE 361X 362X 362X USES NONE 362X 362X 363X \$CSINT PUSH H 64 365 346 362 364 365X 6CSINT PUSH FSW 652 364 661 041 365X 6CSIX POP FSW 652 373 373 372X ** HAVE DATA INPUT INTERRUP 664 311 35 040 375X \$CSIX MYI A*SINBUF 664 076 035 376X 6CSIX POP FSW 665 317 373 374X 6CSIX POP FSW 666 318 377X 8CSIX HAVE DATA INPUT INTERRUP 667 374 375X \$CSIX MYI A*SINBUF 668 375 376X 6CSIX POP FSW 669 311 376X 376X 6CSIX APD CHARACTER TO QUEUE. 669 316 374 376X 6CSIX MOV A*M 673 176 386X 6CSIX MOV A*M 673 176 386X 6CSIX MOV A*M 674 177 386X MOV A*M 675 315 06 040 386X 6MI 1770 676 315 06 040 386X 6MI 1770 677 177 346 177 386X 6MI 1770 678 167 167 167 167 167 167 167 167 167 167	ESERVE REGISTERS HAVE DATA FROM INPU SEE IF ROOM IN THE
341X 343X ±52X ±CSINT PUSH H 044 345 045 315 114 040 345X CALL #CISI 050 344 002 346X CALL #CISI 052 304 061 041 346X CALL #CISI 055 341 346X CALL #CISI 055 341 346X CALL #CISI 056 341 356X FOP FOP H 057 373 373X ** HAVE DATA INFUT INTERRUP 064 076 035 375X EI KT H;*INBUFL 065 313 375X CMP M 064 076 035 375X CMP M 065 334 350 040 375X CMP M 065 334 350 040 375X CMP M 067 334 350 040 375X CMP M 067 334 350 040 376X ADD CHARACTER TO QUEUE. 380X * ADD CHARACTER TO QUEUE. 381X ADD CHARACTER TO QUEUE. 382X *CALL *CDIN 074 205 385X MOV L;A 075 315 106 040 386X ADD CALL *CDIN 103 167 386X MOV M;A 391X CHECK FOR SPECIAL CONTROL 391X CHECK FOR SPECIAL CONTROL 391X CTL-D	ESERVE REGISTERS HAVE DATA FROM INPU SEE IF ROOM IN THE' ROOM ROOM ROOM ROOM ROOM SEE IF ROOM IN THE' SEE IF ROOM IN THE' SECUENCES.
043 345 363X \$CSINT FUSH H 044 365 364X FUSH FUSH FSW 050 315 114 040 365X ANI USR.RX 050 346 002 368X \$CSIX FUP FSW 055 364 061 041 367X FET FOP FSW 055 361 368X \$CSIX FUP FSW 055 361 368X \$CSIX FUP FSW 056 311 372X FET FOP H 064 076 035 373X ** HAVE DATA INFUT INTERFUP 373X ** HAVE DATA INTERFUP 373X ** HAVE DATA INTERFUP 064 076 035 376X FCSII LXI H,\$INBUF 065 276 377X CMP M 066 276 377X CMP M 067 334 350 040 378X CSI3 INR M 073 176 389X ADD L 074 075 389X ADD L 075 315 106 040 386X CALL 076 315 106 040 386X ADD 077 078 078 389X ADD 078 078 078 078 078 078 078 078 078 078 078 079 070 070 070 070 070 071 071 074 075 070 072 074 075 070 070 074 075 070 070 070 075 076 070 070 077 070 070 070 078 070 070 078 070 070 0	ESERVE REGISTERS HAVE DATA FROM INPU SEE IF ROOM L) = ADDRESS OF CHAR FUT CHARACTER IM IT SEQUENCES. ECIAL USER INTERRUPT
045 315 114 040 365X CALL \$CISI 050 346 002 366X ANI USR.RXR 055 304 061 041 367X FCISI 056 311 359X FCISI FOP FSW 057 373 372X ** HAVE DATA INFUT INTERRUP 064 076 035 375X ** HAVE DATA INFUT INTERRUP 064 076 035 375X ** HAVE DATA INFUT INTERRUP 065 331 375X ** HAVE DATA INFUT INTERRUP 065 311 572X ** HAVE DATA INFUT INTERRUP 066 076 035 375X FCSII LXI H;*INBUF 067 334 350 040 375X FCSII LXI H;*INBUF 067 334 350 040 375X FCSII LXI H;*INBUF 068 076 035 375X ADD CHARACTER TO QUEUE. 077 078 078 078 ADD CHARACTER TO QUEUE. 078 078 078 078 ADD CHARACTER TO QUEUE. 078 078 078 078 078 ADD CHARACTER TO QUEUE. 079 070 070 070 070 070 070 070 070 070	HAVE DATA FROM INPU SEE IF ROOM IN THE' ROOM ROOM IN IT ORE IT SEQUENCES. ECIAL USER INTERRUPT
052 304 061 041 367X CNZ \$CSI1 055 341 368X \$CSIX POP FSW 055 341 359X ESIX POP FSW 055 341 370X EI 060 311 372X ** HÀVE DATA INPUT INTERRUP 064 074 035 375X \$CSI1 LXI H/\$INBUFL 064 076 035 375X CMP H 065 274 377X CMP H 065 276 334 350 040 379X CMP H 072 064 379X ADD CHARACTER TO QUEUE. 072 064 389X ADD L 073 156 389X ADD L 074 157 389X ADD L 075 157 389X ADD L 076 315 106 040 386X ADD L 077 078 078 389X ADD L 078 078 315 106 040 386X ADD L 078 315 107 389X ADD L 078 315 107 389X ADD SALL \$CDIN 101 346 177 387X ADD SALL \$CDIN 103 167 389X ADD CHECK FOR SPECIAL CONTROL 391X CHECK FOR SPECIAL CONTROL 392X CHECK FOR SPECIAL CONTROL 3930X CHECK FOR SPECIAL C	HAVE DATA FROM INFU SEE IF ROOM IN THE ROOM L) = ADDRESS OF CHAR FUT CHARACTER IM IT ORE IT SEQUENCES. ECIAL USER INTERRUPT
055 341 359X E1 H H H H H H H H H H H H H H H H H H	SEE IF ROOM IN THE TOOM ROOM L) = ADDRESS OF CHAR FUT CHARACTER IM IT ORE IT SEQUENCES. ECIAL USER INTERRUPT
050/ 313 371X RET 372X ** HÀVE DATA INFUT INTERRUP 372X CAP ADD CHARACTER TO QUEUE. 372X ADD CHARACTER TO CONTROL 374X ADD CHARACTER TO CONTROL 374X ADD CHARACTER TO CTL-D CONTROL 372X ADD CTL-D CTL-D	SEE IF ROOM IN THE TROOM ROOM L) = ADDRESS OF CHAR FUT CHARACTER IM IT ORE IT SEQUENCES. ECIAL USER INTERRUPT
3.7.2. ** HAVE DATA INFUT INTERRUP 3.7.2. ** HAVE DATA INFUT INTERRUP 3.7.4. ** HAVE DATA INFUT INTERRUP 3.7.4. ** CSII LXI H;*INBUF H,*INBUF H,*IN	SEE IF ROOM IN THE TOOM ROOM L) = ADDRESS OF CHAR FUT CHARACTER IM IT ORE IT SEQUENCES. ECIAL USER INTERRUPT
064 041 155 040 375X \$CSII LXI H;\$INBUF 064 076 035 377X CMP M 065 276 334 350 040 378X CC \$FCSI2 300X * ADD CHARACTER TO QUEUE. 380X * ADD CHARACTER TO QUEUE. 381X MOU A,M 072 064 382X \$CSI3 INR M 073 176 384X ADD L,A 074 205 384X ADD L,A 075 157 385X ADD L,A 101 346 177 385X ADD L,A 103 167 386X ADD MAA 1103 167 386X ADD CALL \$CDIN 1103 167 386X ADD MAA 399X CHECK FOR SPECIAL CONTRO 391X CTL-A TD CTL-D	ESS OF CHAR
064 076 035 376X MUI A,*INBUFL 065 276 377X CMP M 065 276 378X CMP M 067 334 350 040 379X CC \$*CC \$*CS12	ESS OF CHAR CTER
066 2/6 37/X CMP	ESS OF CHAR CTER
379X 380X * ADD CHARACTER TO QUEUE. 380X *CSI3 INR M 072 176 383X ADD L 074 205 384X ADD L 075 157 385X ADD L 076 315 106 040 386X ANI 1770 1103 167 389X ANI 1770 1103 167 389X ANI 1770 391X CHECK FOR SPECIAL CONTRO	ESS OF CHAR
381X 382X \$CSI3 INR M 383X MDV A;M 074 205 384X ADD L 075 157 385X ADD L;A 076 315 106 040 386X CALL \$CDIN 101 346 177 387X ANI 1770 103 167 388X MDV M;A 390X CHECK FOR SPECIAL CONTRO 391X CTL-A TO CTL-D	ESS OF CHAR CTER
073 176 383X M0V A,M 074 205 384X ADD L 075 157 385X MOV L,A 076 315 106 040 386X CALL *CDIN 101 346 177 387X MOV M,A 103 167 389X MOV M,A 399X CHECK FOR SPECIAL CONTRO 391X CTL-A TO CTL-D	ESS OF CHAR
075 157 385X M00 L,A 076 315 106 040 386X CALL *CDIN 101 346 177 387X ANI 1770 103 167 388X M00 M,A 390X CHECK FOR SPECIAL CONTRO	ESS OF CHAR CTER R INTERRUPT
076 315 106 040 386X CALL *CDIN 101 346 177 387X ANI 1770 103 167 389X MOU M'A 389X CHECK FOR SPECIAL CONTRO 391X CTL-A TO CTL-D	TIEK
103 167 389X MOV M*A 389X 389X CHECK FOR SPECIAL CONTRO 391X CTL-A TO CTL-D	R INTERRUPT
389X 390X * CHECK FOR SPECIAL CONTRO 391X 392X * CTL-A TO CTL-D	R INTERRUPT
* CTL-A TO CTL-D	ERRUPT
□	
**************************************	CLEAR DISCARD MODE
* CTL-6	CLEAR HOLD MODE SET HOLD MODE
397X •104 376 040 398X CFI ′ ′	~
1.106 320 399X RNC 400X	NOT CONTROL CHARACTER
401X * HAVE CONTROL CHARACTER.	
, 056 212 403X HUI	
093	POINT AT NEXT ELEMENT IN TABLE
CME	CONFARE, TO CHARACTER
1 330 407X RC RC 043 408X INX H	IS NOT IN LIST
302 111 041 409X JNE	

- XX	EATHKITS
- XX	E TITIE

HEA	HEATH HB CONSOLE DRIVER INTERRUPT-TIME CONSOLE DRIVER,	VER.		НЕАТН ХВАЅМ VI.1 06/21/77 17:53:40 01-AFR-77 FAGE 11	
	041.121 072 252 040 041.124 246 240 041.125 256 040 041.127 062 252 040 041.132 365 155 041.137 360 041.137 360	4 10 X 4 11 2 X 4 11 2 X 4 11 2 X 1 1 2 X 1 1 2 X 4 11 2 X 4 11 2 X 4 11 9 X 4 11 9 X 4 11 9 X 4 12 0 X 4 13 X 4 14 17 X 4 15 X 4 15 X 4 16 X 4 17 X 4 16 X 6 X 6 X 7	LDA \$CSLCTL ANA M INX H XAA M SET \$CTL STA \$CSLCTL FUSH FSW MVI L,***INBUF DECREMEN POF FSW HAVE SPECIAL CONTROL, CALL USER,	SET *CTLCTL BITS SET *CTLCTL BITS SAVE BITS DECREMENT CHARACTER FROM BUFFER RETURN IF NOT CTL-A THROUGH CTL-D	
	041,140 052 250 040 041,143 351		LHLD \$CSIC PCHL	CALL USER ROUTINE	
ል የያ የያ የያ የ	O41.144 ASSEMBLY COMPLETE 426 STATEMENTS O ERRORS DETECTED 22612 BYTES FREE	426 END	₽		
					: :

#EATHKIT®	
	_

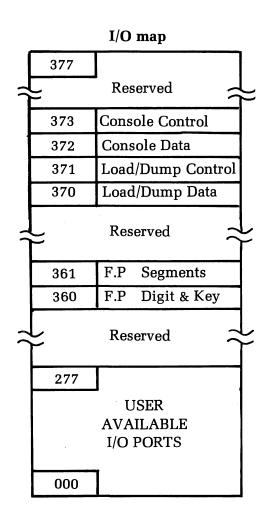
CROSS RE *CDIN	REFERENCE T	FTARIF								
*CDIN							PAGE 12	ρį		
	040106	158L	386							
*CISI	040114	164L	365	270	741	350				
\$C150	040117	120L	315		÷.	2:				:
\$000\$	040125	173L								:
*CSI1	041061	367 309L	375L 378							
\$CS13	041072	382				: : : : : : :				
*CSIU	041111	212E	403							
\$CS1C	040250	232L	423							
*CSINT	041043	347	363L							
\$CS1X	040050	234L	297	410	414					
#CSLLEN	040254	236L								i
\$INBUF	040155	201L	204	238	253	344	3/5 416			
*INBUFL	000035	204E	376							:
*PRSCL	040152	195L 195	334L							
*RC1.5	040310	265	267	270L				· · · · · · · · · · · · · · · · · · ·		
\$RCHAR	040144	191L								:
** ** CHAK	040255	191	252L							
*RCHAR1	040261	254L	256							:
*RCHARU	040311	2/2L 273	277 281L							
*RCHARA	040307	269E								
\$RET	040110	159L	232							
IUL	040130	176L 1701								
ZISI*	040136	182								
\$TSOUT	040141	185L	336	339	342					:
*WCHAR	040147	193L	1760							
**************************************	040333	74,62	300							:
*WCHAR2	040354	303	315L	318						
•	0000040	2385	239	1						
UIVEC	040037	24E	346	348						:
BELL	000000	23E 140E	310							
	000010	143E								
CC.CTLC	000040	144E								:
CC.CTLI	000100	145E	202							
	00000	140F	298							
COLNO	040253	2351	1							:
config	333333	27E	149							
ENTRY	040100	149L 110F	158							:
IP.CIS	000372	117E 121E	164							:
IF.COS		123E	170							
IF. TDF		126E	176		:					
IP.TSF		128E	182							
OF. CDF.		120E	161							:
OP.CIS	000373	122E	167							
OP CDS.	•	1245	1/4							:
OF, TDP		- 1/6	^							

222222 111111	222222 111111 256 1000000 0000000000000000000000000000	222222 111111 256 1000000 000001 526 0000000 0000000 0000000000000 000000	111111 256 150 111111 256 349 000000 526 349 000000 526 349 000000 376 349 0000000 376 349 0000000 376 349 0000000 376 349 0000000 376 349 000000000000000000000000000000000000	111112 256 111111 256 1000002 246 000002 526 0000002 536 0000002 536 0000003 349 0000003 346 0000003 346 0000003 346 0000003 346 0000000 3476 0000000 3476 00000000 3476 0000000 3476 00000000 3476 00000000 3476 000000000 3476 000000000000000000000000000000000000	1.212111 7.7.		PAGE 13
HY 111111 2000 IE 0000000 1 2000 IR 0000000 52E 0000000 52E 16X 0000001 53E 16X 0000001 53E 18X 0000001 53E 18 000001 53E	HY 1 111111 2 25E ER 000020 IR 000100 S2E 000004 S2E 000001 16X 000001 16X 000001 16X 000001 16X 000001 17E 000001 18 000010 18 0000010 18 00000010 18 00000010 18 0000000000	HY 1 11111 2 25E ER 000020 IR 0000002 IR 0000004 RE 00000001 IR 0000001 IR 000001 IR 0000001 IR 000001 IR 000001	HY 111111 255 ER 000020 IR 000100 52E 000004 53E 10 000001 53E 11 000001 53E 12 000001 53E 13 000100 53E 14 000000 75E 15 000000 75E 16 000000 75E 17 000000 75E 18 0000000 75E 18 000000 75E 18 0000000 75E 18 000000 75E 18 0000000 75E 18 000000 75E 18 0000000 75E 18 000000 75E 18 0000000 75E 18 000000 75E 18 0000000 75E 18 0000000 75E 18 0000000 75E 18 0000000 75E 18 0	HY 111111 2000 IE 0000000 52E IR 0000000 52E 0000001 52E IR 0000001 52E II 0000001 52E II 0000001 52E II 0000001 52E II 0000001 52E II 0000000 52E II 000000 52E II 00000 5			
ER 000020 IF 0000020 RE 0000003 RE 0000003 RE 0000004 RE 0000001 RE 0000001 RE 0000001 RE 0000003 RE 000003 RE 000003 RE 000003 RE 000003 RE 000003 RE 000003 RE 00003 RE 000	ER 000020 IF 000020 IF 000002 IF 000003 IF 0000003 IF 0000004 IF 0000004 IF 0000004 IF 0000001 IF 0	ER 000020 IF 0000020 RE 0000003 RE 0000004 RE 0000004 RE 0000001 RE 0000001 RE 0000001 RE 0000003 RE 0000003 RE 0000003 RE 0000003 RE 00000003 RE 0000003 RE 000003 RE 00003 RE 000	ER 000020 IJE 000020 IJE 000002 IJE 000002 IJE 0000003 IJE 0000003 IJE 0000004 IJE 0000004 IJE 0000001 IJE 00000001 IJE 0000001 IJE 00000001 IJE 0000001 IJE 00000001 IJE 0000001 IJE 0000	ER 000020 IF 000020 RE 0000020 RE 0000003 RE 0000004 RE 0000001 RE 0000001 RE 0000001 RE 0000003 RE 000003 RE 00003 RE 00	ART 222222 RT 111111	26E 25E	150
NO 00000 NO 000000 NO 00000 NO 000000 NO 00000 NO 0000	TR 0000004 10 0000004 TE 0000001 10 0000001 11 0000000 12 0000001 13 000000 14 0000000 15 0000000 16 0000000 17 0000000 18 000000 18 00000 18 0000 18 0000 18 00000 18 00000 18 00000 18 0000	TR 0000000	TR 0000004 1.6 0000004 1.6 0000001 1.6 0000001 1.7 0000001 1.8 0000001 1.8 0000003 1.9 0000003 1.9 0000003 1.9 00000003 1.9 0000003 1.9 0000003 1.9 0000003 1.9 0000003 1.0 000003 1.0 00003 1.0 0003 1.0 0003	F	ER 000020	54E	349
0000000 0000000 0000000 0000000 0000000	0000000 0000000 0000000 0000000 0000000	0000004 0000001 0000001 00000001 00000001 000000	0000000 0000001 0000001 00000001 0000000	0000000 0000000 0000000 0000000 0000000	1E 000002	70F	2,4.4. 2,4.4.
0000000 0000000 0000000 0000000 0000000	0000040 0000002 0000002 0000003 0000003 0000003 0000003 0000003 0000003 0000004 0000003 0000004 00000003 00000003 00000003 00000003 00000003 00000003 00000000	0000000 0000000 0000000 0000000 0000000	0000040 0000001 0000002 0000003 0000003 0000003 0000003 0000000 0000000 0000000 0000000 000000	0000040 0000001 00000001 00000001 00000003 00000003 00000004 0000000000	RE 000004	in in	(4.4.V)
0000001 0000000 0000000 0000000 0000000 000000	0000000 0000000 0000000 0000000 0000000	0000001 0000000 0000000 0000000 0000000 000000	0000000 0000000 0000000 0000000 0000000	0000001 0000001 0000000 0000000 0000000 000000	KO 000040	53E	£
0000000 0000000 0000000 0000000 0000000	0000000 0000000 0000000 0000000 0000000	0000000 0000000 0000000 0000000 0000000	0000000 0000000 0000000 0000000 0000000	0000000 0000000 0000000 0000000 0000000	1E 000001	3/5	74.5 4.5 4.5
0000001 0000000 0000000 0000000 0000001 0000010 0000010 0000010 0000010 000000	0000001 0000000 0000000 0000000 0000000 0000010 000000	0000001 0000000 0000000 0000000 0000001 0000010 0000010 0000010 000000	0000001 0000000 0000000 0000000 0000000 000000	0000001 0000000 00000000 0000000 0000001 0000010 0000010 0000010 000000	18 000100	37E	0.40 W.40
0000300 0000000 0000000 0000000 0000010 0000010 0000010 0000010 000000	0000000 0000000 0000000 0000000 0000010 000000	0000300 0000000 0000000 0000000 0000010 0000010 0000010 000000	0000000 0000000 0000000 0000000 0000000	0000300 0000000 0000000 0000000 0000010 0000010 0000010 0000010 0000010 000000	1X 000001	46E	
0000003 0000003 00000004 00000010 00000010 0000002 0000002 0000001 0000001 0000001 0000001 0000001 000000	0000003 00000000 00000000 00000000 000000	0000003 00000004 00000004 0000010 0000010 0000010 0000010 00000010 000000	0000003 0000003 00000004 00000000 00000000	0000003 0000003 00000004 0000010 0000010 0000020 0000020 00000010 0000001 0000001 0000001 000000	2B 000300	39E	
0000000 0000010 0000010 0000010 0000020 0000020 0000020 0000001 0000001 0000001 0000001 000000	0000000 0000010 0000010 0000010 0000020 0000020 0000010 0000011 0000001 000001 0000001 000000	0000000 0000010 000000	0000000 00000004 00000010 00000020 00000020 00000020 00000010 0000001 0000001 0000001 000000	0000000 0000010 000000	64X 000003	7.88E	
00000000000000000000000000000000000000	0000000 0000010 0000010 00000020 00000010 00000010 00000010 0000001 000000	00000000000000000000000000000000000000	00000000000000000000000000000000000000	00000000000000000000000000000000000000	000000	1000	
0000010 0000014 0000020 0000040 0000010 0000010 0000001 0000001 000000	0000010 0000014 0000020 0000040 0000010 0000010 0000002 0000001 0000001 0000001 0000001 0000001 000000	0000010 0000014 0000002 0000004 0000001 0000001 0000001 0000001 0000001 000000	0000010 0000014 0000020 0000040 0000020 0000010 0000002 0000004 0000001 0000001 0000001 0000001 0000001 000000	0000010 0000014 0000020 0000040 0000010 0000001 0000001 0000001 000000	1.6 000004	4 4 3 53	
					1.7 0000010	44E	
PA 000020 PE 000040 41E PE 000000 61E PE 000010 63E TXR 000001 65E TXR 000001 66E BYTES FREE	P# 000020 PE 000040 40E FE 000040 63E PE 000002 63E TXE 000001 64E TXX 000001 66E BYTES FREE	PA 000020 PE 000040 41E PE 000000 63E PE 000002 63E TXR 000004 64E TXR 000001 66E BYTES FREE	P# 000020 PE 000040 40E PE 000040 63E PE 00002 63E TXR 000001 64E TXR 000001 66E BYTES FREE	PA 000020 PE 000040 41E PE 000000 PE 000010 63E FXR 000002 65E TXR 000001 66E BYTES FREE	L8 000014	45E	340
FE 000040 000040 FE 000010 FXR 000002 TXE 000001 FXR 000001 FXR 000001 FXR 000001 FXR 000001 FXR 000001	FE 000040 0E 000020 FE 000010 FE 000002 TXE 000004 TXE 000001 BYTES FREE	FE 000040 000020 FE 000010 FE 000002 TXE 000001 FXE 000001 FXE 000001 FXE 000001 FXE 000001 FXE 000001	FE 000040 0E 000020 FE 000010 FX 000002 TXF 000001 TXR 000001 FX 000001 FX 000001 FX 000001 FX 000001	FE 000040 000020 FE 000010 FXR 000002 TXR 000001 FXR 000001 FXR 000001 FXR 000001 FXR 000001 FXR 000001	FA 000020	41E	
PE 000020 PE 000010 FXR 000002 TXR 000004 TXR 000001 TXR 000001 BYTES FREE	PE 000020 63E PE 000010 63E FXR 000002 65E TXR 000001 66E FXR 000001 66E	PE 000020 DE 000010 FXR 000002 TXR 000001 TXR 000001 BYTES FREE	PE 000020 63E PE 000010 63E FXR 000002 65E TXR 000001 66E FXR 000001 66E FXR 000001 66E	PE 000020 DE 000010 FXR 000002 TXR 000001 TXR 000001 TXR 000001 BYTES FREE	FE 000040	40E	
0000010 000002 0000004 0000001 0000001 64E 66E	000001 000002 000004 000001 000001 000001 000001 000001 000000	0000010 000002 0000004 0000001 0000001 64E	0000010 000002 0000004 0000001 646E S FREE	0000010 000002 0000004 0000001 0000001 0000001 0000001 0000000	FE 000040	61E 42E	
RXR 000002 TXE 000004 TXR 000001 66E BYTES FREE	RXR 000002 TXE 000004 TXR 000001 66E BYTES FREE	RXR 000002 TXE 000004 TXR 000001 66E BYTES FREE	RXR 000002 TXE 000004 TXR 000001 666E BYTES FREE	RXR 000002 TXE 000004 TXR 000001 66E BYTES FREE	FE 000010	63E	
TXE 000004 64E TXR 000001 66E BYTES FREE	TXE 000004 64E TXR 000001 66E BYTES FREE	TXE 000004 64E TXR 000001 66E BYTES FREE	TXE 000004 64E TXR 000001 66E BYTES FREE	TXE 000004 64E TXR 000001 66E BYTES FREE	KXR 000002	65E	366
BYTES FREE	BYTES FREE	BYTES FREE	BYTES FREE	BYTES FREE	TXE 000004	64E	
BATES FREE	PATES FREE	PATES FREE	PATES FREE	PATES FREE	I AR UUUUU	100	916
					BYTES FREE		



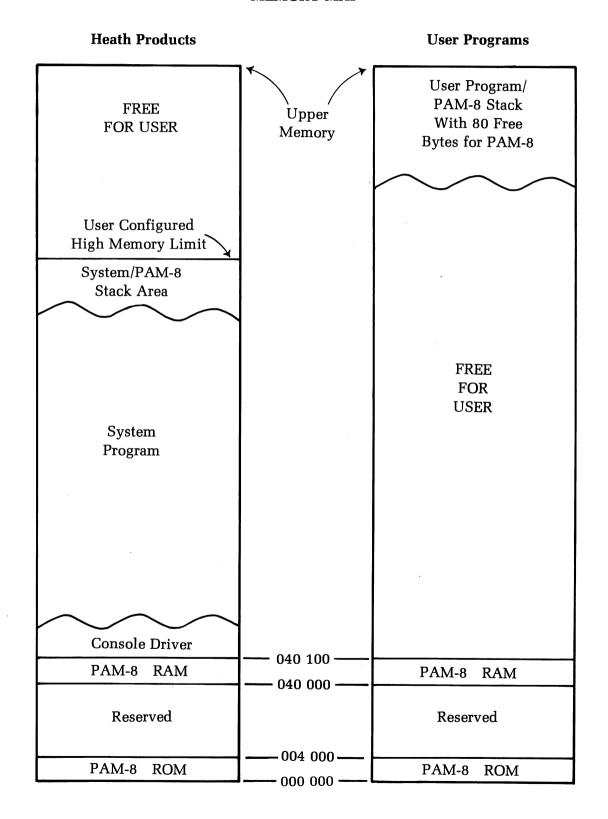
APPENDIX C

This appendix contains I/O and Memory Maps for the H8 software products.





MEMORY MAP





APPENDIX D

ASCII Characters

7-BIT			
OCTAL CODE	DECIMAL CODE	CHARACTER	DESCRIPTION
000	0	NUL	NULL, TAPE FEED, CONTROL SHIFT P.
001	1	SOH	START OF HEADING; ALSO SOM, START OF
	_		MESSAGE, CONTROL A,
002	2	STX	START OF TEXT; ALSO EOA, END OF ADDRESS, CONTROL B,
003	3	ETX	END OF TEXT: ALSO EOM, END OF MESSAGE CONTROL C.
004	4	EOT	END OF TRANSMISSION (END): CONTROL D,
005	5	ENQ	ENQUIRY; ALSO WRU, CONTROL E,
006	6	ACK	ACKNOWLEDGE. ALSO RU, CONTROL F.
007	7	BEL	RINGS THE BELL. CONTROL G.
010	8	BS	BACKSPACE: ALSO FEO, FORMAT EFFECTOR
			BACKSPACE SOME MACHINES, CONTROL H.
011	9	HT	HORIZONTAL TAB. CONTROL I
012	10	${f LF}$	LINE FEED (NEW LINE):
			ADVANCES PAPER TO NEXT LINE, DUPLICATED
			BY CONTROL J.
013	11	VT	VERTICAL TAB (VTAB). CONTROL K.
014	12	$\mathbf{F}\mathbf{F}$	FORM FEED TO TOP OF NEXT PAGE (PAGE).
			CONTROL L.
015	13	CR	CARRIAGE RETURN TO BEGINNING OF LINE.
			DUPLICATED BY CONTROL M.
016	14	SO	SHIFT OUT: CHANGES RIBBON COLOR TO RED.
			CONTROL N.
017	15	SI	SHIFT IN: CHANGES RIBBON COLOR
			TO BLACK. CONTROL O.
020	16	DLE	DATA LINK ESCAPE. CONTROL P (DCO).
021	17	DC1	DEVICE CONTROL 1, TURNS TRANSMITTER
			(READER) ON, CONTROL Q (XON).
022	18	DC2	DEVICE CONTROL 2, TURNS PUNCH OR AUXIL-
			IARY ON, CONTROL R (TAPE, AUX ON).
023	19	DC3	DEVICE CONTROL 3, TURNS TRANSMITTER
			(READER) OFF, CONTROL S (XOFF).
024	20	DC4	DEVICE CONTROL 4, TURNS PUNCH OR AUXIL-
		· ·	IARY OFF. CONTROL T (TAPE, AUX OFF).
025	21	NAK	NEGATIVE ACKNOWLEDGE: ALSO ERR. ERROR. CONTROL U.
026	22	SYN	SYNCHRONOUS IDLE (SYNC). CONTROL V.
027	34	ETB	END OF TRANSMISSION BLOCK: ALSO LEM.
			LOGICAL END OF MEDIUM, CONTROL W



7-BIT	•		
OCTAL	DECIMAL		•
<u>CODE</u>	CODE	<u>CHARACTE</u>	<u>DESCRIPTION</u>
030	24	CAN	CANCEL (CANCL). CONTROL X.
031	25	EM	END OF MEDIUM. CONTROL Y.
032	26	SUB	SUBSTITUTE. CONTROL Z.
033	27	ESC	ESCAPE. PREFIX.
034	28	FS	FILE SEPARATOR. CONTROL SHIFT L.
035	29	GS	GROUP SEPARATOR. CONTROL SHIFT M.
036	30	RS	RECORD SEPARATOR. CONTROL SHIFT N.
037	31	US	UNIT SEPARATOR. CONTROL SHIFT O.
040	32	SP	SPACE.
041	33	!	
042	34	"	
043	35	#	
044	36	\$	
045	3 <i>7</i>	%	
046	38	· &	
047	39	,	ACUTE ACCENT OR APOSTROPHE.
050	40	(
051	41	j	
052	42	*	
053	43	+	
054	44	,	
055	45	-	
056	46	•	
057	47	1	
060	48	0	
061	49	1	
062	50	2	
063	51	3	
064	52	4	
065	53	5	
066	54	6	
067	55	7	
070	56	8	
071	5 <i>7</i>	9	
072	58 50	:	
073	59	;	
074	60	<	
075	61	=	
076 077	62	; >	
100	63		
100	64 65	@ ^	
101	66	A B	
102	67	В С	
103	68	D D	
104	69	E	
106	70	F	
107	70 71	G	•
	• -		



7-BIT			•	
OCTAL	DECIMAL		•	
CODE	CODE	CHARACTER	-	DESCRIPTION
110	72	H		
111	73	Ι		
112	74	J		
113	75	K		
114	76	L		
115	77	M		
116	78	N		
117	79	О		
120	80	P		
121	81	Q		
122	82	R		
123	83	S		
124	84	T		
125	85	U		
126	86	V		
127	87	W		
130	88	X		
131	89	Y		
132	90	Z		
133	91	[SHIFT K	
134	92]	SHIFT L	
135	93	1	SHIFT M	
136	94	←	SHIFT N	
137	95		A COTTATE OF A LIFE	
140	96		ACCENT GRAVE.	
141	97	a L		
142	98	b		
143 144	99	c d		
144 145	100 101			
145 146	101	e f		
147	102			
150	103	g h		
151	105	i		
152	106	j		
153	107	k		
154	108	ì		
155	109	m		
156	110	n		
157	111	0		
160	112	p		
161	113	q		
162	114	r		
163	115	s		
164	116	t		
165	117	u		
166	118	v		
167	119	w		



7-BIT OCTAL CODE	DECIMAL CODE	CHARACTER	DESCRIPTION
170	120	x	
171	121	у	
172	122	Z	
173	123		
174	124		
175	125		THIS CODE GENERATED BY ALT MODE.
176	126		THIS CODE GENERATED BY ESC KEY (IF PRESENT)
177	127	DEL	DELETE, RUB OUT.



Appendix E

Decimal To Octal Tables

for 0 to 255₁₀

<u>DECIMA</u>	L OCTAL	DECIMAL	<u>OCTAL</u>	DECIMAL	<u>OCTAL</u>
0	0 .	37	45	74	112
1	1	38	46	75	113
2	2	39	47	76	114
3	3	40	50	77	115
4	4	41	51	78	116
5	5	42	52	79	117
6	6	43	53	80	120
7	7	44	54	81	121
8	10	45	55	. 82	122
9	11	46	56	83	123
10	12	47	5 <i>7</i>	84	124
11 ·	13	48	60	85	125
12	14	49	61	86	126
13	15	50	62	87	127
, 14	16	51	63	88	130
. 15	17	52	64	89	131
16	20	53	65	90	132
17	21	54	66	91	133
18	22	55	67	92	134
19	23	56	70	93	135
20	24	57	71	94	136
21	25	58	72	95	137
22	26	59	73	96	140
23	27	60	74	97	141
24	30	61	<i>7</i> 5	98	142
25	31	62	76	99	143
26	32	63	<i>77</i>	100	144
27	33	64	100	101	145
28	34	65	101	102	146
29	35	66	102	103	147
30	36	67	103	104	150
31	37	68	104	105	151
32	40	69	105	106	152
33	41	70	106	107	153
34	42	71	107	108	154
35	43	. 72	110	109	155
36	44	73	111	110	156



<u>DECIMAL</u>	OCTAL	DECIMAL	<u>OCTAL</u>	DECIMAL	OCTAL
111	157	160	240	209	321
112	160	161	241	210	322
113	161	162	242	211	323
114	162	163	243	212	324
115	163	164	244	213	325
116	164	165	245	214	326
117	165	166	246	215	327
118	166	167	247	216	330
119	167	168	250	217	331
120	170	169	251	218	332
121	171	170	252	219	333
122	172	171	253	220	334
123	173	172	254	221	335
124	174	173	255	222	336
125	175	174	256	223	337
126	176	175	257	224	340
127	177	176	260	225	341
128	200	177	261	226	342
129	201	178	262	227	343
130	202	179	263	228	344
131	202	180	264	229	345
132	204	181	265	230	346
133	205	182	266	231	347
134	206	183	267	232	350
135	207	184	270	233	351
136	210	185	271	234	352
137	211	186	272	235	353
138	212	187	273	236	354
139	213	188	274	237	355
140	214	189	275	238	356
141	215	190	276	239	357
142	216	191	277	240	360
143	217	192	300	241	361
144	220	193	301	242	362
145	221	194	302	243	363
146	222	195	303	244	364
147	223	196	304	245	365
148	224	197	305	246	366
149	225	198	306	247	367
150	226	199	307	248	370
151	227	200	310	249	371
152	230	201	311	250	372
153	231	202	312	251	373
154	232	203	313	252	374
155	233	204	314	253	375
156	234	205	315	254	376
157	235	206	316	255	377
158	236	207	317	200	
159	237	208	320		
100	20,	. 200	020		



APPENDIX F

Memory Table

Offset Octal and Decimal Boundaries

Hi Byte	Lo Byte	Decimal Boundary
A15A8	A7A0	
0.04	0 0 0	1024
020	0 0 0	4096
0 4 0	0 0 0	8192
060	0 0 0	12288
1 0 0	000	16384
1 2 0	0 0 0	20480
1 4 0	0 0 0	24576
160	0 0 0	28672
2 0 0	0 0 0	32768
2 2 0	0 0 0	36864
2 4 0	0 0 0	40960
2 6 0	0 0 0	45056
3 0 0	0 0 0	49152
3 2 0	0 0 0	53248
3 4 0	0 0 0	57344
3 6 0	0 0 0	61440
3 7 7	3 7 7	65535*

For example, if you have 12K bytes in an H8, the lower boundary is at 8192, or 040 000 offset octal. The upper boundary is at 8K + 12K = 20K (20480), or 120 000 Octal.

^{*}NOTE: 65,535 is the last location in a memory addressed by 16 bits.



INDEX

ASR Console, 0-22

Basic Program Format, 0-15

Checksum, 0-12 Compressed, Text Format, 0-15 Configured Tape, 0-19 Console Debugger (BUG-8), 0-8 Console Driver, 0-23 Copying Tapes, 0-22 CRC-16, 0-12 Control Characters, 0-23

Data Formats, 0-14 Distribution Tapes, 0-19 Displays, 0-15

Entry Point, 0-23

File, 0-12 Front Panel Displays, 0-15 Front Panel Entry Points, 0-23

Heath Assembly Language (HASL-8), 0-10 Heath Text Editor (TED-8), 0-8

Installation, 0-19 ff.,

Label Record, 0-14 Loading, 0-21

Magnetic Tape, 0-18 Memory Image Format, 0-14

110-Baud Console, 0-22

Patch Installation 0-22 Pad Characters, 0-20 Panel Monitor (PAM-8), 0-7 Paper Tape, 0-18

ROM, 0-7 Record, 0-12 Record Structure, 0-13

Software Installation, 0-19 ff., Software Distribution Tape, 0-19 Software Problems, 0-24 Software Version Code, 0-24

Tape Files, 0-12 Tape Format, 0-12