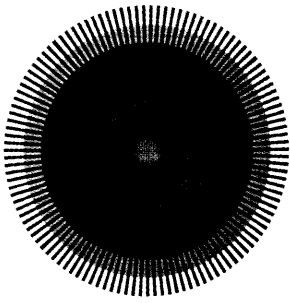


DATA-PAC

Introduction To
H112 DIGITAL CONTROLLER

Prepared by
The H112 Team

Honeywell



DATA-PAC

WHO

THE H112 TEAM

WHAT

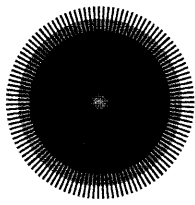
WILL INTRODUCE YOU TO THE
H112 MINICOMPUTER

WHERE

HERE AND NOW

WHEN

AS SOON AS YOU PLUG IN THE H112
AND GO TO NEXT PAGE TO GET SOME
ACTION.



The H112 team presents to you the Honeywell 112 Controller. It is a low cost digital processor featuring high speed, efficient use of memory, and the adaptability needed to tailor the unit to your needs in on-line, real-time applications. Honeywell has configured the H112 to meet practically any application by use of standard, off-the-shelf, plug-modules. Memory is field expandable from 4K to 8K, using a standard 4K plug-in module. The plug-in approach also allows a user to share one control panel among several H112 installations.

The H112 has four types of input/output: Programmed I/O; Load Mode Transfers; Machine Control Interface; and Direct Data Channel.

The standard programmed input/output structure in the H112 is a single-cable, duplex, "party line" system. Six I/O instructions, plus reassignable, programmed priority interrupts make the controller's parallel transfer I/O bus capable of a wide variety of applications.

TO MAKE THE 112 RUN WE MUST

FIRST – "REVIEW ITS CONTROLS AND DISPLAYS"

(Look at Control Panel on H112)

POWER ON/OFF SWITCH

Controls operating power to controller.

Push on upper portion of switch to turn power on.

Displays

Indicators within the black band on the upper portion of the control panel display the contents of the A, P, or W registers. The register to be displayed is selected by pushing on the bottom portion of the register switch located on the lower left side of the control panel. The A register is the accumulator used in the H112 system. Its function is to perform arithmetic operations in conjunction with the serial adder and to hold the results. The A register (12-bit) also functions as an interface for I/O operations. The P register is the program counter and normally contains the address of the next instruction to be executed. In the standard 4K memory, the register has 12 bits. In the 8K memory version of the H112, the register has 13 bits. The W register is a 12-bit working register used for memory data interface and instruction execution. The control panel displays the contents of the A register following an LDA or INA instruction when the machine is running, with the overflow flop as bit 13. The other registers may be displayed when the machine is halted and the RUN switch is in the STOP position. (The bottom portion of the switch is in toward the face of the control panel and the top portion of the switch is out from the face of the control panel.)

Switch Register - Thirteen switches are provided to allow entry of data for bits 1-13, which correspond to the display indicators. Register must be selected to enter data. Data can be entered only in the stop position because of the interlock feature of the control panel.

*Register Select

Switches A, P, and W select the desired register for display or data entry. *Select only one register at a time.*

CLEAR

The CLEAR switch clears the selected register.

RUN/STOP

The STOP position of the RUN/STOP switch halts the controller. Once in the stopped mode, registers can be selected, displayed and modified; in this mode, the controller can be stepped using the START switch. The controller reads the instruction from memory, executes the instruction, increments the P register, and halts each time the START switch is depressed.

*STORE

To store data, the memory location is entered in the P register, the STORE switch is depressed, the data is placed in the W register. The START switch is depressed to store the contents of the W register in the location specified by the program counter. The program counter is then automatically incremented.

*FETCH

To retrieve data from a memory location, the location is entered in the P register and then the FETCH and START switches are depressed. The data is displayed in the W register. The P register is incremented.

*LOAD

The load switch enables the controller to input load mode paper tapes. Load format tapes are loaded by setting the P register to the starting location, setting the RUN/STOP switch to STOP, depressing the LOAD switch, then the START switch. The paper tape is loaded into the ASR reader and the ASR's start switch operated. More on this later.

RUN

The RUN indicator, when lighted, signifies that the controller is running the stored program or operating in the load mode.

START

In addition to the Store, Load, etc. functions, the START switch, when depressed, causes the controller to begin program execution at the address specified by the P register if the RUN/STOP switch is set to run.

MASTER CLEAR

The MASTER CLEAR switch clears main registers, plus control addressing, and I/O F/F's, but does not affect the A and W registers.

Switch Positions

When the controller has power on and is running normally, the upper part of all two position switches are in toward the panel.

Auxiliary Control

In addition to the standard control panel, or when full displays are not required for a specific application, plug-in connections are available for external control of the following functions: Run/Stop, Start, Master Clear, Load, and single instruction execution.

*Each respective switch is selected (on) when the bottom portion of the switch is in toward the face of the control panel and the top portion of the switch is out from the face of the control panel.

ASR33 INPUT/OUTPUT DEVICE

To run the demo package, operate the ASR33 as listed below:

1. Plug the line cord into the ASR33 110V outlet on the back of the H112 controller.
2. Turn "on" the H112.
3. Set the control knob to the on-line position.
4. Press line feed and carriage return keys to make sure paper is feeding.

FOR MORE ASR 33 FACTS READ ON - - - - -

The components of the Teletype unit and their functions are:

Control Knob

The control knob of the ASR33 Teletype console has the following three positions.

Line

The Teletype console is energized and connected to the H112 as an input/output device under computer control.

Off

The Teletype console is de-energized.

Local

The Teletype console is energized for off-line operation under control of the Teletype keyboard and switches exclusively.

Keyboard

The ASR33 keyboard is similar to that of a standard typewriter. The keyboard contains four rows of keys that generate 8-bit ASCII codes. All letters are upper case; the SHIFT key being used only for special punctuation marks and symbols which correspond to upper-case positions on standard typewriters. The keyboard does not lock in the upper-case position. Hence, the SHIFT key must be held depressed to produce the special characters.

Control functions are generated by holding down the control key (CTRL) while the particular function is being pressed. The keyboard is mechanically interlocked for all keys except SHIFT, CTRL and REPT, thus preventing more than one key from being depressed at a time.

Printer

The printer provides a typed copy of input and output at a maximum rate of ten characters per second. When the Teletype unit is on-line (LINE), the copy is generated by the computer or keyboard when the Teletype unit is off-line (LOCAL), the copy is automatically generated whenever a key is struck.

Paper Tape Reader

The paper tape reader is used to input data punched on eight-channel perforated paper tape. The reader control positions are:

Start

Activates the reader; reader sprocket wheel is engaged and operative.

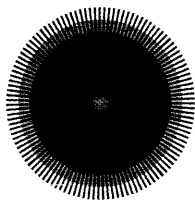
Neutral

Reader sprocket wheel is engaged; to stop reader, move lever toward free.

Free

Deactivates the reader; reader sprocket wheel is disengaged.

The reader will accept and read paper or mylar tape 1.0 in. wide (8-channel) of any thickness. The packing density of the data on the tape must be 10 characters per inch. Reading speed is 10 characters per second. If the punch is turned on, any characters being read by the reader, generated by the keyboard or transferred from the controller will be punched. The reader will operate only when a tape is mounted and the cover is down.



The reader is controlled either manually by the START/STOP/FREE lever on the reader housing, or by programmed operation. The first character read is the one placed over the read pins. The FREE position of lever releases the tape for insertion or removal. The reader, by programmed control, is started by an X-ON character output. The reader is stopped (under both manual and program control) by an X-OFF character on the tape, by running out of tape, or by the lever being placed in STOP.

Paper Tape Punch

The paper tape punch is used to perforate eight-channel rolled oiled paper tape at a maximum rate of ten characters per second. The punch controls are:

- REL.
Disengages the tape to allow tape removal or loading.
- B.SP.
Backspaces the tape one space for each firm depression of B.SP. button.
- ON
Activates the paper tape punch.
- OFF
Deactivates the paper tape punch.

Oiled paper tape, 1.0 in. wide (8-channel), and 0.003-0.005 in. thick, is recommended for use in the punch. The punch housing can accommodate a normal 855 ft. roll of this type of tape. The packing density of the data, as it is punched, is 10 characters per inch. Maximum punching speed is 10 characters per second. Tape is mounted in the punch by pressing the release (REL) button on the punch housing and inserting the end of the blank tape under the roller at the rear of the housing. The punch is actuated manually by ON and OFF buttons on the housing. Tape is punched by depressing the ON button for any input to, or output from the ASR33. A tape leader of 20 blank characters is produced by each depression of the HERE IS key. Each depression of the B.SP. (back space) pushbutton, back spaces one character position. The REL button allows insertion and removal of tape. The paper tape format is shown below.

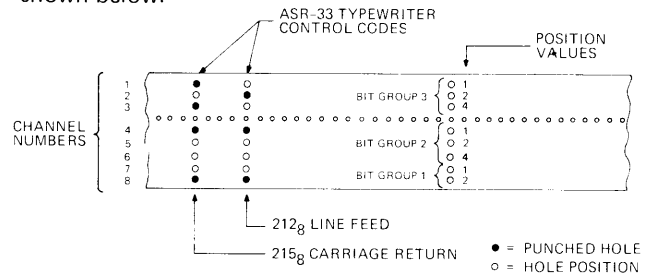
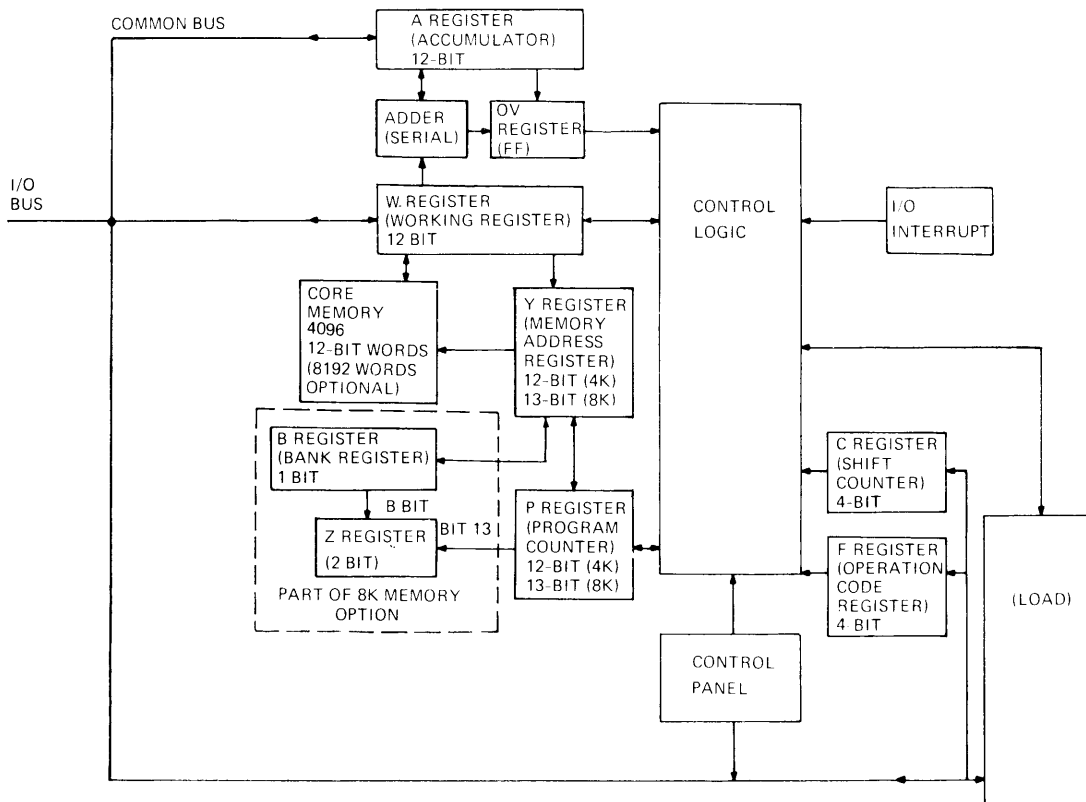


Figure 2. ASCII Punched Tape Format

SECOND – A FEW H112 MACHINE FACTS



H112 Controller – System Block Diagram

Type

Parallel, binary, stored program

Addressing

Single-word addressing with single-level indirect addressing

Number System

Two's complement

Circuitry

Integrated, DTL

Instruction Complement

Five instruction types, making up 37 standard instructions; skip instructions are microprogrammable

Word Length

12 bits

Control Panel

Optional control panel is plug-in. The panel has facilities for display of the A, P, or W registers as well as RUN/STOP, STORE, FETCH, START, LOAD, and MASTER CLEAR functions. Interlocks prevent accidental entry in the run mode.

Input/Output Rate

One 12-bit word/3.39 μ s or 295K words/sec (Direct Data Channel Option)

Input/Output Bus

Party line with priority interrupt structure. Optional bidirectional direct data channels (2).

Memory

Coincident current, random access, ferrite core; 12-bit word length; one or two 4K modules, field expandable to 8K. Memory cycle time 1.69 μ s.

Peripherals and Options

10-cps ASR33 paper tape reader/punch/printer. High speed paper tape reader (400 cps). A second 4K memory module for 8K memory capability.
Real Time Clock.
Analog Input/Output.
Digital Input/Output.
Power Failure Interrupt.
Power Restart.
Direct Data Channel.
Asynchronous Single Line Controllers
Option Drawers and Power Supplies
High speed paper tape punch (110 characters/sec)

Physical Characteristics

Power consumption: 200W, 115V AC 10%, single phase, 48 to 62 Hz (60 0.5 Hz for ASR33)

Dimensions: 19-inch rack mount, 7 inches high by 24 inches deep

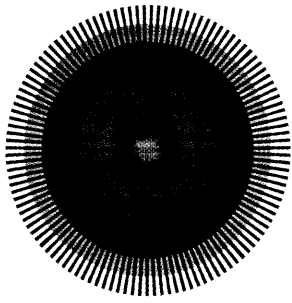
Environment: Room ambient (less I/O devices): 0 to 50°C
Storage: -20°C to 85°C

REGISTER COMPLEMENT

A Register	Accumulator	12 bits
W Register	Working Register	12 bits
P Register	Program Counter	12 bits*
B Register	Bank Register	1 bit**
Z Register	Save Register	2 bits**
OV Register	Overflow F/F	1 bit
Y Register	Memory Address Register	12 bits*
C Register	Shift Counter	4 bits
F Register	Operation Code Register	4 bits

* 13 bits with 8K option

** 8K only

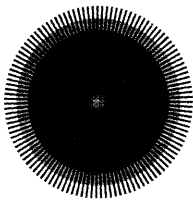


DATA-PAC

DEMONSTRATION PROGRAMS

Try one of these demonstration programs:

1. *Tic-Tac-Toe*
2. *Write Your Own Book*
3. *MAXBUG*
4. *Instruction Test*
5. *Memory Test*



DEMONSTRATION PROGRAM 1 TIC-TAC-TOE

To make the 112 play Tic-Tac-Toe, first the paper tape loader (LDROAR) must be loaded as follows:

1. Turn 112 power on.
2. Place teletype in 'on-line' mode.
3. Mount LDROAR tape in the teletype reader at blank just prior to loader portion.
4. Press master clear.
5. Set load switch to ON.
6. Select register switch P.
7. Using the panel switch register, set the P register to 7600_g.
8. De-select the P register.
9. Press start. The I/O typewriter (ASR33) is used as the loading device, therefore, the Start/Stop/Free lever must be moved to the start position before the reader will start. The loader will automatically be read and will stop at the stop character.
10. Set load switch to off.

NOTE

The paper tape loader has now been loaded. If the LDROAR is in memory, steps 1-10 may be skipped.

11. Press master clear.
12. Set P register to 7600_g.
13. Set Run/Stop switch to run.
14. Mount the object tape (TIC-TAC-TOE demo tape) to be loaded into the ASR33 reader.
15. Press start.

The Tic-Tac-Toe tape contains a transfer block which causes the loader to transfer control to the program after the load is complete. Thus, when the loading process has terminated, the program will start automatically. The program occupies memory locations 0 to 2435_g. The program starts at location 200_g.

PLAYING TIC-TAC-TOE

To play the game, just follow the instruction typed on the ASR. The algorithm is such that the machine can't lose no matter who starts. On the other hand it can't win against an optimum defense.

If you select not to continue with the program and answer the question asked by the program negatively, the machine will halt. To start again, just press START. To restart the program without the header information (instructions) start the controller at location 1000_g.

WINNING AT TIC-TAC-TOE

Human ego being what it is, we had to find a way of beating the machine. (After all, who would admit to being beaten by a \$5000 Computer?)

The only way we found was the equivalent of performing a prefrontal lobotomy on the program, by changing the following locations:

Location	Change from	to
1121 _g	SNZ (302 _g)	NOP (200 _g)
1173 _g	SNZ (302 _g)	NOP (200 _g)

This change can be done via MAXBUG or from the front panel as follows:

1. Set Run/Stop switch to stop.
2. Press master clear.
3. Select register switch P.
4. Press clear.
5. Set P to location 1121_g.
6. Deselect register switch P.
7. Select register switch W.
8. Press clear.
9. Place the instruction NOP (200_g) into the W register.
10. Select the store switch.
11. Press start.

12. Deselect register switch W.
13. Repeat steps 3 through 12 (except in step 5, P should be set to 1173_g).
14. Deselect the store switch.
15. Select register switch P.
16. Press clear.
17. Set P to 200_g.
18. Deselect register switch P.
19. Set Run/Stop switch to run.
20. Press start.

We won't tell you any more about how to beat the machine except that it is now vulnerable to a multiple attack.

DEMONSTRATION PROGRAM 2 WRITE A CHAPTER

This is the H112 Demonstration Program used to introduce the Controller at the ISA Show. The program is a six-chapter narration. Each chapter contains pertinent details regarding the H112. The printout is designed for easy reading and random chapter selection. Also, we will tell you how to write your own chapter.

To make the 112 present its chapter, first the paper tape loader (LDROAR) must be loaded as follows:

1. Turn 112 power on.
2. Place teletype in 'on-line' mode.
3. Mount LDROAR tape in the teletype reader at blank just prior to loader portion.
4. Press master clear.
5. Set load switch to ON.
6. Select register switch P.
7. Using the panel switch register, set the P register to 7600_g.
8. Deselect the P register.
9. Press start. The I/O typewriter (ASR33) is used as the loading device. Therefore, the Start/Stop/Free lever must be moved to the start position before the reader will start. The loader will automatically be read and will stop at the stop character.
10. Set load switch to off.

NOTE

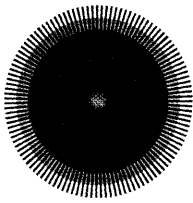
The paper tape loader has now been loaded. If the LDROAR is in memory, steps 1-10 may be skipped.

11. Press master clear.
12. Set P register to 7600_g.
13. Set Run/Stop switch to run.
14. Mount the ISA Demonstration Program tape in the ASR33 reader.
15. Press start.

The Demonstration Program will be loaded when the reader stops. Note that the entire paper tape will not be loaded at this point.

The program is designed to take in messages, store them in memory and print out on demand. The remainder of the tape is messages to be loaded as follows:

16. Set the Run/Stop switch to stop.
17. Press master clear.
18. Select the register switch P.
19. Press clear.
20. Set P equal to 200_g.
21. Deselect the register switch P.
22. Set the Run/Stop switch to run.
23. Press start. The ASR will carriage return and line feed.
24. After the teletype stops, type in I0 and start the ASR33 reader.
25. After the teletype stops, type in I1 and start the ASR33 reader.
26. After the teletype stops, type in I2 and start the ASR33 reader.



27. After the teletype stops, type in I3 and start the ASR33 reader.
28. After the teletype stops, type in I4 and start the ASR33 reader.
29. After the teletype stops, type in I5 and start the ASR33 reader.

NOTE

Ignore what is being printed on the ASR33.

Once the demo is loaded, type any number 0 through 5. This will print one of six different chapters.

If you want to make up your own narration input messages, 0 to 5 may be changed by typing 1 followed by the number of the message to be changed. The message (including carriage returns and line feeds) may now be typed on the ASR. When you are finished inputting the message, type a rubout character to tell the program that you are finished. When the number associated with the message is typed, your message will be typed. If you make a mistake in typing, you can delete the previous character by typing a (back arrow). Multiple back arrows will delete multiple characters. The maximum lengths of the individual message are:

To restart the demo program:

1. Set the Run/Stop switch to stop.
2. Press master clear.
3. Select the P register switch.
4. Press clear.
5. Set P = 200₈.
6. Deselect the P register switch.
7. Set the Run/Stop switch to run.
8. Press start.

<i>Message Assigned</i>	<i>Memory Locations Assigned</i>
0	500 ₈ - 2477 ₈
1	2500 ₈ - 3777 ₈
2	4000 ₈ - 4777 ₈
3	5000 ₈ - 5777 ₈
4	6000 ₈ - 6777 ₈
5	7000 ₈ - 7777 ₈

**DEMONSTRATION PROGRAM 3
MAXBUG**

MAXBUG is a debugging aid designed to assist in the checkout of other programs. The main purpose is to open memory locations for display and alteration of contents. Although this sort of thing can be done from the control panel, MAXBUG enables this to be done very conveniently from the ASR. In addition to the accessing of memory locations, other functions normally required during debugging can be performed. The list of commands performed by MAXBUG is as follows:

Access a memory location (type and alter contents).

Dump memory (list a block of memory locations in octal).

Mnemonic dump (list memory locations as instruction mnemonics).

Punch paper tape.

Load paper tape.

Enter a value in a block of memory.

Reproduce a section of memory in another part of memory.

Compare a paper tape to memory.

Jump to a memory location.

Search a portion of memory for a specific bit pattern.

In addition to typing and accepting in octal, MAXBUG also communicates in the instruction mnemonics.

LOADING MAXBUG

MAXBUG is supplied in object format paper tape; therefore it must be loaded by the software loader. To load the program, proceed as follows:

1. Turn 112 power on.
2. Place teletype in on-line mode.
3. Mount LDROAR tape in the teletype reader just prior to loader portion.
4. Press master clear.
5. Set load switch to on.
6. Select register switch P.

7. Using the panel switch register, set the P register to 7600_g.
8. De-select the P register.
9. Press start. The I/O typewriter (ASR33) is used as the loading device, therefore the Start/Stop/Free lever must be moved to the Start position before the reader will start. The loader will automatically be read and will stop at the stop character.

MAXBUG will be loaded into memory locations 6000 to 7177 (just below the loader). When the loading process is finished, start MAXBUG as follows:

10. Set load switch to off.

NOTE

The paper tape loader has now been loaded. If the LDROAR is in memory, steps 1-10 may be skipped.

11. Press master clear.
12. Set P register to 7600_g.
13. Set the Run/Stop switch to run.
14. Mount the MAXBUG program tape in the ASR33 reader.
15. Press start.

OPERATING MAXBUG

When MAXBUG starts, it performs a carriage return, line feed and types "***" to indicate that it is ready for a command. A command consists of a single alphabetic character followed by one, two or three octal numbers separated by commas. The command is terminated by a carriage return. If an error in typing is made, the command is terminated by a / (slash) after which MAXBUG types "***". The alphabetic character commands are the first letter of the list given previously. The octal numbers following the alphabetic command will specify memory locations involved.

In the case where two memory locations are specified, the first address entered must be less than the second entered. MAXBUG will not increment across a 4K bank boundary, but can operate on addresses, either in the same bank in which it resides, or in the opposite bank.

DETAILED OPERATIONS

Operator Command:

Daaaaa,bbbb(CR)

aaaaa and bbbbb are addresses of from one to five octal digits.

If bbbbb is omitted, it is assumed equal to aaaaa. For example, to dump the first sector of core type DC, 177 and carriage return.

Program Response:

A listing of core from location aaaaa through bbbbb is produced on the I/O typewriter. The listing is printed with eight locations on a line. Each line is preceded by the location of the first word on that line. The operator may terminate the core dump operation by pressing the BREAK key on the I/O typewriter.

Mnemonic Core Dump

Operator Command:

Maaaaa,bbbb(CR)

aaaaa and bbbbb are addresses of from one to five octal digits.

If bbbbb is omitted, it is assumed equal to aaaaa. For example, to dump in mnemonics the first sector of core, type MO, 177 and carriage return.

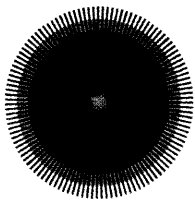
Program Response:

A listing of core from location aaaaa through bbbbb is typed by the I/O typewriter. The listing is typed with one location on each line.

Each line will consist of:

- a. The location (five octal digits).
- b. The mnemonic operation code (three alphabetic characters).
- c. An asterisk if the indirect bit is set.
- d. The contents of the location (four octal digits).

The operator may terminate the core dump by pressing the BREAK key on the I/O typewriter.



Access a Memory Location

Operator Command:

Aaaaaa(CR)

aaaaa is an address of from one to five octal digits. For example, to access location 1000, type A1000 and carriage return.

Program Response:

The program types the location, contents of the location (octal), the instruction mnemonic, *(asterisk) and waits for an operator response.

Operator Response:

The operator may type one of the following:

- a. cccc(CR), which changes the contents of the location to cccc; the program then types the next location, its contents and waits for the operator response.
- b. aaa*cccc (CR), which changes the contents of the location to the instruction specified by mnemonic aaa, indirect address indicator*, and address cccc. The program then types the next location, its contents and waits for the operator response. If the entered address, cccc, is between 0 and 177, the address is placed in the instruction with a zero sector bit; an address greater than 177 is truncated to 7 bits and a "this sector" bit is placed in bit 8.
- c. Carriage return, which will cause the program to advance to the next location without changing the contents of the current location.
- d. Slash, which will cause the program to exit from the access routine and wait for the next command.

If in the example above, location 1000 contained the instruction INA01, MAXBUG would type

1000 4001 INA.

To change that instruction to ADD indirectly through location 1226, type ADD*1226 and a carriage return. To change location 1000 to octal 7237, type 7237 and a carriage return.

Enter Into Memory

Operator Command:

Eaaaaa,bbbb,xxxx(CR)

aaaaa and bbbbb are addresses of from one to five octal digits; xxxx is a number of from one to four octal digits, if bbbbb is omitted, it is assumed equal to aaaaa; if xxxx is omitted, it is assumed equal to zero. For example, to clear the first half of core, type EO,3777, 0 and carriage return.

Program Response:

All memory cells from aaaaa through bbbbb are set to xxxx.

Punch Memory

Operator Command:

Paaaaa,bbbb(CR)

aaaaa and bbbbb are addresses of from one to five octal digits; if bbbbb is omitted, it is assumed equal to aaaaa.

Program Response:

The contents of memory from location aaaaa through bbbbb are punched on the I/O typewriter in load mode format. The punched tape may be loaded via the L command of MAXBUG or by the hardware loader. The tape format is described in the H112 Programmer's Reference Manual, Appendix B.

Since the punch on the ASR is manual, the operator must turn it on. After the carriage return, turn the punch off while MAXBUG punches the trailing leader.

The operator may terminate the memory dump by pressing the BREAK key on the I/O typewriter.

Compare to Memory

Operator Command:

Caaaaa,x(CR)

x is 1 for I/O typewriter, 5 for high-speed paper tape reader.

Program Response:

A load mode tape (or one punched by a "P" command) is compared with the memory locations starting at aaaaa and the total number of differences are printed in octal on the I/O typewriter. The address of the differing memory location, the contents of the tape, and the current contents of memory are printed on the I/O typewriter if tape is compared from the high-speed reader.

Jump to Location

Operator Command:

Jaaaaa,xxxx(CR)

aaaaa is an address of from one to five octal digits;
xxxx is a number of from one to four octal digits; if
either is omitted, 0 is assumed.

Program Response:

xxxx is placed in the A register and execution begins
at aaaaa. For example, to jump to location 10 with A
register contents of zero, type J10 and carriage return.

Search Memory

Operator Command:

Saaaaa,bbbb,xxxx,yyyy(CR)

aaaaa and bbbbb are addresses of from one to five
octal digits; xxxx is a number of from one to four
octal digits; yyyy is a mask.

Program Response:

All memory locations from location aaaaa through
location bbbbb are compared to xxxx; both numbers are
masked by (ANDed with) yyyy before the compare; all
locations which match are printed on the I/O typewriter;
the operator may terminate the operation by pressing the
BREAK key on the I/O typewriter. For example, to search
all of core (lower 4K) for all INA instructions (regardless of
device address), type SO,777,4000,7700 and carriage re-
turn.

Reproduce Memory

Raaaaa,bbbb,cccc(CR)

aaaaa,bbbb and cccc are addresses of from one to
five octal digits. Address cccc may be in the bank
opposite aaaaa and bbbbb.

Program Response:

Memory locations from aaaaa through bbbbb are
copied into corresponding locations starting with cccc.

As an example of the use of this command after
loading an un-debugged program into memory, a copy may
be reproduced in another unused part of core. If, later,
during the debugging process, the program "blows up" and
a reload is desired, the copy saved may be moved down.
Thus the reload time is much faster than rereading the

original tape. To reproduce the first 1K of core in the third
quarter, type RO,1777,4000 and carriage return.

Load Into Memory

Operator Command:

Laaaaa,x(CR)

aaaaa is the starting address of the load, X=1 for the
I/O typewriter, X=5 for the high speed reader.

Program Response:

A load mode tape is loaded into memory starting at
address aaaaa from the selected device.

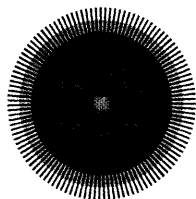
The program is in core locations 200 through 231. (It also
uses 232 and 233.) To enter the program, start MAXBUG
(at location 6000). MAXBUG will type "***". To start
entering, type A200 and a carriage return. MAXBUG will
type out 200 followed by the current contents of location
200. To enter the first instruction, type IRS232 and a
carriage return. MAXBUG will make up that instruction
taking care of the instruction field, sector bit and address,
put it in location 200 and then type out the contents of
201. To enter the next instruction, type INA01 and a
carriage return. MAXBUG does not care for a space
between INA and 01. If a mistake is made, type a / (slash).
MAXBUG will ignore what has been typed on that line and
give the "***". It is not necessary to go back and retype the
lines already correctly entered. If the error is made at
location 215, type A215, a carriage return and STA233
after MAXBUG stops typing the current contents. When
you get down to location 225, the 212 may be typed in,
MAXBUG accepts the mnemonics or octal inputs.

Checking the Program

When the entire program has been entered, it can be
checked by listing the contents of the memory. MAXBUG
will list the contents in response to M200, 231 followed by
a carriage return. The listing will be as follows:

Address	Contents (octal)	Instruction
200	3232	IRS
201	4001	INA
202	1600	JMP
etc.		

MAXBUG will also list in octal only by typing D200,231
and carriage return. In all cases the printout may be
prematurely terminated by depressing the "break" key,
while the ASR is typing



Running the Program

Once the program is in and checked, it can be executed by transferring control to the beginning of the program (location 200). This can be done by typing J200 and a carriage return. The function of the program is to type out random octal numbers. To cause a printout, the space bar

or any key on the ASR is operated (except shift or control). In response, the ASR will do a carriage return, line feed and type a four octal digit random number. The program is useful for wagering for coffee and other games of chance.

SAMPLE PROGRAM

ENTERING THE PROGRAM

A sample program that can be entered via MAXBUG is shown below:

LOCATION	INSTRUCTION	COMMENTS
200	IRS232	Increment random number
201	INA01	Did someone type a character?
202	JMP200	No, go back and increment
203	LDA225	Someone typed, load A with carriage return
204	OTA02	Output if ready
205	JMP204	Not ready, go back and try again
206	LDA226	Load A with a line feed
207	OTA02	Output if ready
210	JMP207	Not ready, go back and try again
211	LDA232	Load A with -4
212	STA233	Store in a location for counting
213	LDA232	Load A with random number
214	RAR11	Shift to the right 9 places ($9_{10} = 11_8$)
215	STA232	Save A in random number
216	ANA230	Make all but the 3 LSB=0
217	ADD231	Add constant to make ASCII character
220	OTA02	Output if ready
221	JMP220	Not ready, go back and try again
222	IRS233	Increment counter, done?
223	JMP213	Not done typing, go back and do next character
224	JMP200	Done, go back and wait for input
225	215	ASCII carriage return
226	212	ASCII line feed
227	7774	MINUS 4
230	0007	MASK
231	260	Constant to make ASCII number

The program operates by constantly incrementing a memory location while waiting for an input from the ASR. When the ASR key is operated, the program outputs a carriage return and line feed. It then separates the first (left) three bits of the memory location being incremented, makes it into an ASCII character and outputs it on the ASR. The program repeats the operation three more times for the remainder of the 12-bit location. The separation and outputting of the number is done by way of a loop which is executed four times. After the program has outputted the entire word, it returns to the beginning of the program awaiting another input. By changing location 202 to a JMP201, the program will type successive octal numbers instead of random numbers in response to the space bar.

By changing location 227 to -3 (7775), -2 (7776), -1 (7777), the length of the random number will be reduced to 3, 2, or 1 characters, respectively. It does not work well in the successive number case.

More Tricks With MAXBUG

Returning control to MAXBUG (by stopping the machine, setting P=6000 going to run and then start) enables other features to be used.

Punching The Program

The program can be punched out on paper tape in load mode by typing P200,231 and carriage return. The ASR punch must be manually turned on after the carriage return, and then manually turned off after the punching has terminated and the trailing leader is being punched.

Filling Memory

The sector of core used by the program may be cleared by typing E200,377,0 and a carriage return. The clearing can be verified by D200,377 and carriage return. When you are tired of 0000's, the break key will wake up MAXBUG.

Loading Tapes

To reload the program that was punched, load the tape into the ASR reader and type L200,1 and carriage return. The tape may also be loaded by stopping the machine, pressing the master clear switch, setting P=200, operating the load switch and start. Then put the tape into the ASR reader and move the load lever on the ASR to the start position.

Reproducing Program

The program can be copied into another place within memory by typing R200,231,400 and a carriage return. The program will then be copied into locations 400 to 431 with no alterations of the original in 200-231. Because 400 is the start of next sector, and this program will run in the beginning of any sector, (not true of all programs), the copy in 400 can be executed by a J400 command.

Searching Memory

MAXBUG can be also used for searching memory for certain data fields. For example, to search the original program (in 200) for all LDA instructions regardless of address, type S200, 231, 0400, 3400 and a carriage return. The number after the S is the first location of the search, the second is the last location of the search, the next is the quantity searched for and the last is a mask which is anded with both the memory location contents and the quantity searched for. The mask in this case contains "ones" in the instruction field only (bits 9 to 11). MAXBUG types out the location of the LDA instruction. (203, 206, 211, and 213).

Comparing Tape to Memory

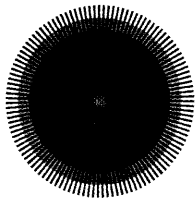
To demonstrate the final MAXBUG command, alter several locations in the original program. For instance, alter locations 205, 210, and 215 by issuing the "A" (access) command of MAXBUG. Next, load the tape produced previously by the P (punch) command in the ASR reader. Type C200,1 and carriage return. MAXBUG will compare the tape to the corresponding memory locations and type out the total numbers of errors found (in octal). If this compare is done using a high-speed reader, MAXBUG will type out the address of the locations differing, the core contents and the tape contents. The compare feature is valuable for verifying a tape punch. The compare feature may also be used in debugging programs. Once a new program has been loaded, the usual practice is to start it and see what happens. Usually, the program halts or hangs up in a loop somewhere. From this, the programmer must deduce what has happened. With the compare feature and the high-speed reader, the locations which differ may be typed out. This will give a clue as to which locations were erroneously altered, which subroutines were not called, which subroutines were called, and the last calling address.

DEMONSTRATION PROGRAM 4 INSTRUCTION TEST

Instruction test is a program that tests and verifies that the instructions of the H112 Controller are working correctly. This test program is a set of subroutines bound together such that all test subroutines are executed sequentially. The instruction test program will test the instructions in either a 4K or an 8K H112 Controller. Provisions are made such that a NOP instruction can be replaced with a jump instruction to isolate any given test.

The following instructions are *not* tested by this routine:

OTA	Output transfer from accumulator
INA	Input transfer to accumulator
SKS	Skip if set
OCP	Output control pulse
SMK	Set mask



ENB	Enable interrupts
INH	Inhibit interrupts
STL	Stall on line
TAB	Transfer accumulator to bank register
TBA	Transfer bank register to accumulator
ITS	Interrupt save
ITR	Interrupt return

Two counters are included at the end of the program to count the number of times the instruction test goes through.

LOADING THE INSTRUCTION TEST PROGRAM

The instruction test program is supplied in Load Mode format and therefore, doesn't require the software loader. To load the program, proceed as follows:

1. Mount tape (INST TST) in the ASR33 reader.
2. Press master clear.
3. Set Run/Stop switch to stop.
4. Set load switch to ON.
5. Press H112 start.
6. Move the ASR33 reader Start/Stop/Free lever to the start position.

The instruction test program will automatically be read and will stop at the stop character.

The correct loading of the program can be ascertained as follows:

1. Deselect the load switch.
2. Press register select switch P. The P register should read 1231₈.

OPERATING THE INSTRUCTION TEST PROGRAM

Having accomplished the loading, the program may be started as follows:

1. Press clear.
2. Deselect the register switch P.
3. Press master clear.
4. Set Run/Stop switch to run.
5. Press start.

The program should halt in location 0100₈. This test checks the halt instruction on the first pass of the test. After this halt, press start. The program should run. If the program fails to run, check P counter for proper location and refer to listing to find routine exhibiting failure. This program will display the number of passes in A register, after each pass.

DEMONSTRATION PROGRAM 5 MEMORY TEST

Memory test is a program that tests and verifies that the memory for the H112 Controller is working correctly. This program will exercise the 4K or the 8K memory version of the H112 Controller. It will load memory with all zeros then check each location for errors. It will then sequentially load and check the memory with ones, each location with its own address, alternate zeros and ones, reverse the zeros and ones, store zeros and then walk ones through memory, zeros and ones in the worst-case pattern through memory, then reverse the worst case in above test until halted manually or by an error.

LOADING OF MEMORY TEST

The memory test program is supplied in load mode and, therefore, doesn't require the software loader. To load the program, proceed as follows:

1. Mount tape (MEM TST) in the ASR33 reader.
2. Press master clear.
3. Set Run/Stop switch to stop.
4. Set load switch to on.
5. Press start.
6. Move the ASR33 Reader Start/Stop/Free lever to the start position.

OPERATING THE MEMORY TEST PROGRAM

Having accomplished the loading, the program may be started as follows:

1. Set the load switch to off.
2. Press master clear.
3. Set Run/Stop switch to run.
4. Press start.

Any errors detected will cause the controller to halt in location 0062, return the run switch to the stop position and address the start switch. The A register will display the number of passes through a 4K memory bank. Depressing the start switch will then cause the A Register to display the operation in progress, the faulty memory address and, finally, the contents of the faulty address. The memory test exercises the entire memory and will wipe out all other program in core.

IMPORTANT – YOUR SOFTWARE LOADER PAPER TAPE LOADER (LDROAR)

If we assume that the core memory of the H112 is entirely empty, the first question is "how to get the paper tape programs into the machine?" The answer is that the H112 has a hardware loader which can load a paper tape into memory. This is called the "load mode" which takes three bits from each character on paper tape and when four characters have gone by, the H112 has enough bits to fill one memory word (12 bits). The load mode logic continues to strip three bits from each paper tape character and "pack" four sets of three into each location until the logic recognizes a stop code.

As it turns out, the assemblers for the H112 produce a different format tape, the "object mode" tape. Basically, this format selects six bits from each paper tape character and when two characters have gone by, there are enough bits to fill one memory word. There are other features of the object format tape such as error detection. The object format is more efficient requiring about half as much paper tape (two tape characters per memory word instead of four in load mode) and, therefore, loads into the machine almost twice as fast. But the problem is, "how to load a paper tape in the object mode?" The answer is the paper tape loader.

In the demo kit, several programs are supplied in the "load mode" format and several in the "object mode". Both the write-up of the program and the program tape itself tell the mode of the tape. The "load mode" tapes are loaded into the H112 with the load mode logic (hardware in the machine), while object tapes are loaded by running the paper tape loader program.

It is important to note that the paper tape loader is supplied in the "load mode". In order to get anything into an empty memory, it is going to have to be in load mode. Once you get the paper tape loader into core, you can load either style tape.

LOADING THE PAPER TAPE LOADER (WITH CONTROL PANEL)

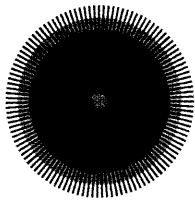
Unroll the loader program tape. Notice that there is about two feet of tape with holes, followed by a short stretch of leader, a few characters, more leader and finally about

8-1/2 feet of paper tape with holes as shown in the Leader Tape Format illustration. The first part is the bootstrap program and the isolated characters are the relocation address. Neither of these are required when the control panel is used in conjunction with the ASR to load the loader. The final 8-1/2 feet of punched tape is the loader program itself, which we will now load.

1. Mount the tape in the ASR33 reader tape at blank just prior to loader portion. The ASR must be *on-line* for proper data transfer.
2. Set run/stop to stop (bottom in).
3. Press master clear.
4. Set load switch to ON, (bottom in).
5. Select register switch P (bottom in).
6. Using the panel switch register, set the P register to the first location of the sector into which the loader is to be loaded; P register bits 1 through 7 must be 0's. We use 7600.
7. De-select the P register (top in).
8. Press start. When the I/O typewriter (ASR-33) is used as the loading device, the Start/Stop/Free lever must be moved to the Start position. The loader will automatically be read and will stop at the stop character.
9. Press master clear.
10. Set load switch to off. (top in)

The loader is now in memory. Now you can load an object mode tape, for example, MAXBUG, TIC-TAC-TOE or the ISA DEMO program. Once the loader is in core, an object tape is loaded as follows:

1. Master clear.
2. Set P register to the first location of the loader (same location established in step 6, above). We are going to run the loader.
3. Set Run/Stop switch to run.
4. Mount the object tape to be loaded into the ASR33 reader.
5. Press (H112's) START. The ASR will load the object tape. If the tape does not contain a transfer block, the loader will halt. Repeat steps 4 and 5 for additional object tapes. If the object tape contains a transfer block, the loader will transfer control to the specified address.



LOADING THE LOADER WITHOUT THE CONTROL PANEL

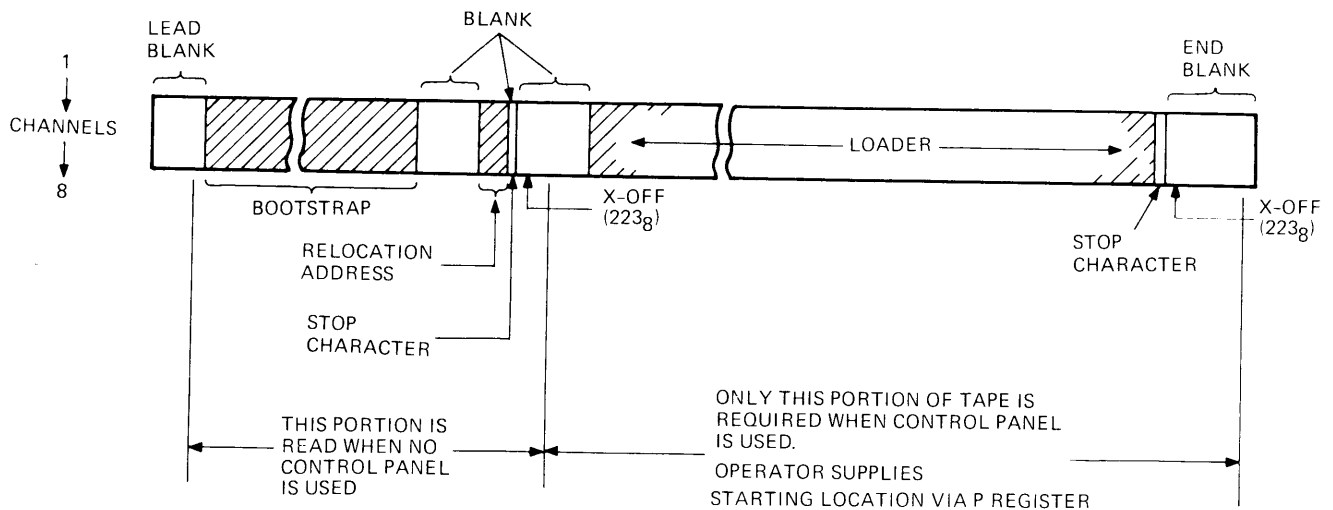
The bootstrap and the relocation address on the loader tape are for use when the H112 does not have a control panel. The Programmer's Reference Manual, page 7-7 and 7-8 illustrate this. They can also be used when the panel is present. The relocation address will place the loader in 7600_8 in a 4K machine or 17600_8 in an 8K version.

The relocation address for the loader may be changed. First, prepare a duplicate tape. The relocation address may then be replaced with a new section of tape with the desired address. The relocation address may also be changed by locating the two relocation characters, on the reproduced loaded tape, and by repunching those two characters with "8" on the I/O typewriter keyboard; the coded 8 causes a punched hole in Channel 4, indicating "delete". Repunch the desired characters in the blank space preceding the "delete" characters. Take care that the two characters are punched in the proper order (first, bits 1-3; then bits 4-6).

DETAILED LOADER DESCRIPTION

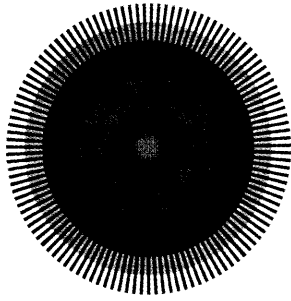
Detailed loader operation is described in the H112 Programmer Reference Manual, page 7-3 and 7-4.

Two loaders are available LDROHR for the high speed paper tape reader and LDROAR for the ASR33. They are distributed as system tapes in load mode format in the configuration shown in the Loader Tape Format illustration shown previously. The standard tapes are provided with a relocation address such that the loader is placed in 17600_8 in an 8K machine or 7600_8 in a 4K machine. The loaders are one sector long and are relocatable by sector.



NOTES:

1. LOADER CHARACTERS = $0_8 \dots 7_8$ (ASCII CODES 260 - 267 OR $020_8 - 027_8$)
2. LOAD MODE IGNORES BLANK TAPE. STOP IS BY STOP CHARACTER ONLY.
3. FOUR TAPE CHARACTERS = ONE 12-BIT MEMORY WORD.
4. RELOCATION ADDRESS IS TWO TAPE CHARACTERS FOLLOWED BY STOP CODE.
ADDRESS BITS: TAPE CHARACTER 1 = BITS 1-3 OF SECTOR NUMBER;
TAPE CHARACTER 2 = BITS 4-6 OF SECTOR NUMBER
5. A PUNCHED CHANNEL 7 INDICATES STOP CHARACTER (SUCH AS "H" -ASCII 310).



H112 SOFTWARE AND OPTION SUMMARY

Software

A comprehensive package of H112 programs is available to the user as part of the standard system. These programs, designed for a wide range of user skills, include the most widely used programming practices and conventions. The software includes the following items: CAP12 Assembler, SAP-12 Assembler, Utility Debug, Loader, Math Library and Diagnostics.

Each complete program in the software package consists of the following items:

- a. *Program Listings* — Program Listings are a product of the assembly process and are a graphic representation of the Source Program. The Program Listings also depict the object code generated by the assembly process.
- b. *Object Program* — The Object Program is the end product of the assembly process and represents the program in machine intelligible form.

A Programmers Reference Manual has been designed to instruct personnel on the use of the controller and its software package. By frequent reference to the manual, the user can progressively increase his general knowledge of the software and how to apply it for specific applications.

CAP12 Assembly Program

CAP12 is a two-pass symbolic language assembly program. The purpose of CAP12 is to translate from a symbolic language, convenient to the programmer, to binary code intelligible to the H112. Translation is generally on a one-for-one basis; that is, for each source statement written by the programmer, one machine instruction is generated by CAP12.

CAP12 source language includes the capability of symbolic addressing, mnemonic machine codes, complex address expressions and automatic desectorization.

SAP-12 Assembly Program

The SAP-12 performs the same functions as the CAP12 except that it is run on a Honeywell H-316 or DDP-516 Computer.

Loader Program

The H112 Paper Tape Loader is a program which loads assembler output object program paper tapes into the H112 core memory. The loader program reads the object program tape, which is the ASCII format, and formats 12-bit words for memory.

Debug Utility Program

The H112 Debug Utility program is a programming aid which allows the programmer to communicate with the H112 central processor during the program checkout process. Program functions include 10 basic operator commands and program responses. Program responses generally include memory dump operations via I/O typewriter typed or punched outputs, access to memory for program changes, and entry of data via I/O typewriter or tape reader.

A list of commands and program response is provided below:

A	=	Access a memory location
C	=	Compare to memory
D	=	Dump core to octal
E	=	Enter into memory
J	=	Jump to
L	=	Load tape into memory
M	=	Mnemonic core dump
P	=	Punch memory
R	=	Reproduce memory
S	=	Search memory

The Debug Utility program may reside in any part of memory. Two versions of the Debug Utility are provided:

Minimum version (MINBUG) — allows the use of the A, D, J, L, and P commands only.

Full version (MAXBUG) — allows the use of all commands.

Diagnostics

All extensive package of verification and test programs is provided with the H112 including routines for verifying the operation of the control unit, the arithmetic unit, core memory, and the available input/output devices. These routines generate information indicating the operational status of the equipment being verified.

Mathematical

The following mathematical routines are available:

- a. Multiply
- b. Divide

H112 OPTION

TELETYPEWRITER OPTION

MODEL NO. 112-25-1

The ASR33 I/O Teletypewriter Option consists of a Teletype Model ASR33 Teletypewriter and the control logic needed to transfer data from the device to the controller and from the controller to the device. The device consists of a *paper tape reader, a paper tape punch, a printer and a keyboard*. The maximum rate of information transfer is *10 characters per second*. The device may operate on-line or off-line (local). When the device is off-line, tapes may be generated and reproduced and messages may be typed as on a normal typewriter. When it is on-line data may be transferred to the controller via the load mode function or via programmed inputs. In the load mode, four octal digits of three bits each are loaded sequentially into each 12 bit location in memory. Under program control each 8 bit character is transferred to the lowest order 8 bits of the accumulator. Data is transferred from the controller only via programmed outputs. Both input and output transfers may be operated under interrupt control.

The keyboard is similar to that of a standard typewriter. It contains four rows of keys that generate 8-bit ASCII codes.

All letters are upper case. The shift key is only used for special punctuation marks and symbols which correspond to upper case positions on normal typewriters.

Control functions (such as X-ON, which turns on the paper tape reader and X-OFF, which turns it off) are generated by holding down the control key (CTRL) while the particular function key is being pressed (Q and S, respectively, in the example given above). The keyboard does not lock in the upper case position so that SHIFT must be held depressed to produce the special characters.

H112 OPTION

HIGH SPEED PAPER TAPE READER OPTION

MODEL NO. 112-20 RACK MOUNTED

NO. 112-22 PORTABLE

The High Speed Paper Reader Option consists of a Digitronics Model 2540, Perforated Paper Tape Reader and the control logic needed to transfer data from the reader to the controller. Tape motion is started by de-energizing the

brake solenoid and energizing the pinch roller solenoid so that the tape comes in contact with a continuously rotating capstan. Tape motion is stopped by de-energizing the pinch roller and re-energizing the brake.

Each time a sprocket hole is sensed an INA will be given by the controller under either program or interrupt control. This causes the control logic to gate an 8-bit character through line drivers directly into the I/O bus. The character is transferred to the lowest order 8-bits of the A-register and the control logic is reset during the INA.

Information may also be transferred from the reader to the controller without a software program using the load mode function. The tape is mounted in the same way as for programmed I/O transfer, then pressing the LOAD push-button followed by the START button on the control panel.

The standard reader speed is 400 characters per second. The first character is read within 10 ms of receiving a start command. The reader will stop before reading the next character after a stop command. In the standard configuration, the interface logic is contained in the H112 housing and is connected to the reader by a five-foot cable.

The portable tape reader is the same in every way except for packaging. The reader, its control logic and a power supply are mounted together in a separate housing which may be moved from controller to controller. It is linked to the controller via an I/O bus cable that plugs into the vacant I/O bus connector of the last interface logic in the configuration.

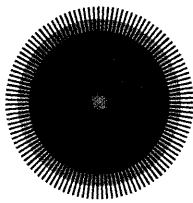
H112 OPTION

DIRECT DATA CHANNEL

MODEL NO. H112-11

The Direct Data Channel provides an I/O path for highspeed data transfers between the controller and I/O devices. Transfers between the channel located in the controller and subchannels located in the peripheral devices utilize both the programmed I/O bus and the DDC bus. All transfers are under subchannel control, however, a transfer may be initiated by a command from the program to the peripheral device over the I/O bus. The address of the memory location to be transferred as well as the direction and time of transfer are under subchannel control. The option gives the H112 a direct memory transfer capability.

The input/output characteristics of the H112 have been carefully tailored to provide maximum interface flexibility, with the widest possible interface tolerances. I/O signals are true in the ground states and relatively immune to noise. Timing requirements give adequate time for a device to respond; and wide strobe pulses eliminate capacitive losses.



H112 OPTION

REAL-TIME CLOCK OPTION

MODEL NO.	112-14-1	10 msec clock
	112-14-2	1 msec clock
	112-14-3	external clock

The Real Time Clock option allows the H112 to keep track of real time by generating a standard interrupt at preset intervals. The period is variable by adjustment of a potentiometer. Another model is available which will generate an interrupt in response to an external signal. Through the use of appropriate subroutines, the program can keep track of time by counting the interrupts generated by the clock or periodically check or control some external process.

The Real Time Clock option uses the controller I/O Bus and generates a pulse on the standard interrupt signal line. If the Real Time Clock mask flip-flop is set, and Run Clock instruction has been executed, the clock pulse will generate an interrupt.

In the internally clocked version, interrupts may be generated over a range of .75 msec to 2.5 msec or 6 msec to 20 msec. Using the externally clocked version and a 60 Hz line trigger, an interrupt will be generated every 16.7 msec. Any external frequency may be used to generate interrupts.

H112 OPTION

POWER FAILURE INTERRUPT OPTION

MODEL NO. 112-16

The Power Failure Interrupt Option will generate a standard interrupt whenever AC line voltage falls below the specified limits for the H112 Digital Controller.

Through the use of appropriate subroutines, the contents of the Program Counter Accumulator and overflow flop may be saved. The subroutines can also allow for machinery shutdown control or other external action.

There is a 1 millisecond of program execution time between power execution time between power failure and memory lockout.

The Power Failure Interrupt Option uses the controller I/O bus. It generates a standard interrupt along the interrupt signal line.

If Power Failure Interrupt mask flip-flop is set and a power fault signal is generated by the power supply, an interrupt will occur.

The Power Failure Interrupt request will last as long as a power fault condition is sensed by the machine.

H112 OPTION

POWER RESTART

MODEL NO. 112-17

The Power Restart option is designed to restart the H112 Controller after a power failure. The Power Restart action will occur each time power is applied or the Power switch is operated and the Run/Stop switch is in the RUN position. The restart action is inhibited when power is applied and the Run/Stop switch is in the STOP position.

When the restart action occurs, the controller will start program execution at memory location 3 with interrupts inhibited, the Bank bit reset and the Overflow flip-flop reset.

The Power Restart option also includes a power supply delay circuit which inhibits startup for no longer than 30 seconds after a power failure.

This delay also occurs after power turn-off.

The Power Failure interruption Option (No. 112-16) is a pre-requisite for the Power Restart Option.

H112 OPTION

ASYNCHRONOUS SINGLE LINE CONTROLLER

The Asynchronous Single Line Controller (ASLC) interfaces signals and data between the H112 Digital Controller I/O Bus and an Asynchronous Data Set or Modem. In general, when in the Transmit mode, the ASLC accepts parallel data words from the H112 I/O bus and transmits the data to a modem in serial form. In the Receive mode, the ASLC receives serial data from the modem and assembles this data into a parallel word for presentation to the H112 I/O bus. The construction is of silicon integrated circuits mounted on epoxy glass printed circuit boards with wire wrapped interconnections.

The ASLC has the following performance features:

The signalling speed will be controlled within the ASLC for asynchronous operation.

The ASLC is wired to handle seven-level code with eight-bit even parity in an eleven-bit start/stop character (one start bit, two stop bits).

Either odd or even parity is available - One, one and one-half, or two stop bit formats can be supplied.

Custom signalling speeds can be specified.

Receive aperture time is equal to the sum of the stop bit plus one data bit period.

Order of transmission is low bit first.

Distortion:

- Transmit 3% maximum per character.
- Receive 45% maximum per character.

The complete subsystem will be contained within a 1 x 3 OMNI-BLOC. Connection to the H112 I/O bus will be by means of two cable PACS. Connection to the data modem will be by means of a 30-foot cable with an EIA RS-232 B compatible 25-pin connector.

Models

Model 112-40 provides a half duplex H112 Controller interface for operation with the Bell System 103A or equivalent switched network data set. The ASLC is designed to accommodate an eleven-bit code consisting of seven data bits, one parity bit, one start bit, and two stop bits at a transmission speed of 110 bits per second.

Model 112-41 provides a half duplex H112 Controller interface for operation with the Bell System 103F or equivalent data set over a dedicated voice grade circuit. The ASLC is designed to accommodate an eleven-unit code consisting of seven data bits, one parity bit, one start bit, and two stop bits at a transmission speed of 110 bits per second.

H112 OPTION

Model 112-42 provides a half duplex H112 Controller interface for operation with the Bell System 202C or equivalent data sets over the switched telephone network. The ASLC accommodates an eleven-unit code consisting of seven data bits, one parity bit, one start bit, and two stop bits at a transmission speed of 1,200 bits per second.

Model 112-43 provides a half duplex H112 Controller interface for operation with the Bell System 202D or equivalent data set over a dedicated voice grade circuit. The ASLC accommodates an eleven-unit code consisting of seven data bits, one parity bit, one start bit, and two stop bits at a transmission speed of 1,800 bits per second.

Model 112-44 allows direct Controller to Controller communication without use of modems or data sets at a signalling speed up to 10,000 bits per second.

Option 112-XX-1 provides for one start bit and one stop bit.

Option 112-XX-2 provides for one start bit and one and one-half stop bits.

Option 112-XX-3 provides for odd parity detection. Standard parity in the ASLC is even.

H112 OPTION

MINI-DIGITAL CONTROL INTERFACE (MINCON)

The MINCON option provides for interfacing between the H112 and up to 24 dedicated input status bits and up to 24 dedicated output control bits. The data is arranged in the form of two input and two output words, any one of which may be addressed at a time.

The logic is contained in a 1 x 3 μ -BLOC which is relocatable within the H112 mainframe drawer. Up to two MINCONs may be used with each controller. The input signal to MINCON may be a standard logic signal (DTL or TTL) or a contact. The output circuit provides the uncommitted collector of a transistor capable of switching up to 100 ma and 24 volts.

H112 OPTION

DIGITAL CONTROL INTERFACE OPTION (DICON)

The DICON option consists of a 3 x 3 μ -BLOC which contains the logic to interface between the H112 controller and up to eight customer oriented input/output option PACs. It is usually contained in a separate option drawer and has its own power supply. Up to two DICONs may be used with one H112.

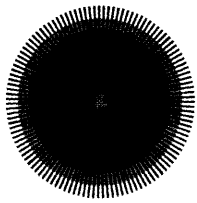
The option PACs include:

Status Input PAC

Used to test the status of external logic signals. These input signals may take the form of switch contacts, 0 to 6 volt logic, or 0 to 24 volt logic levels depending on the option PAC used. Sixteen input lines are available, of which either the upper 12 or the lower 12 may be interrogated under program control. *With this PAC, the controller can keep track of such binary variables as fluid levels (high/low), motors or lights (on/off), micro-switch limit detectors (exceeded or not), etc.*

Asynchronous Input PAC

Used where changes of input data require immediate action by the controller. This option provides an interrupt signal when any one of up to eight inputs changes state. By means of jumpers, the PACs can be made to respond to either "0" to "1" or "1" to "0" or both logic transitions. Input signals may be in the form of switch contacts, 0 to 6 volt logic or 0 to 24 volt logic depending on the option PAC used. This PAC monitors binary inputs which are critical enough to require immediate controller action such as fluid level overflow, motor overheat, pressure alarm, etc.



Counter Input PAC

Contains an eight bit counter which is used to count customer input pulses. It is supplied in two versions. The *Event Counter* is reset to zero at the time its contents are transferred to the controller, after which it is free to continue counting. When a count of 128 (half of full scale) is reached an interrupt is generated and the counter remains enabled. In this way it is possible to keep a continuous count of external events, parts, etc., without having to interrupt program operation for each individual event. Ample time is available between the interrupt signal and the full scale count to transfer data to the H112.

The *Preset Counter* can be preset to the two's complement of a number, "N" under program control. It is then free to count "n" input pulses before reaching the "zero" state at which time an interrupt signal is generated and the count is inhibited. In this way, the controller may take appropriate action to trigger a particular output operation if necessary. This makes it possible for the controller to allow a predetermined number of events to take place before some action is taken without having to interrupt the program for each event.

H112 OPTION

Universal Output PAC

Supplied with either twelve flip-flop or twelve single-shot multi-vibrator outputs. The single-shots are available with various time duration outputs. All outputs may be used to either sink or supply current to some output device. This PAC makes it possible for the controller to turn motors, valves, status display lamps, etc., on or off in response to varying input parameters or computer outputs.

Power Output PAC

Similar to universal output option except that only eight power flip-flops or eight power single-shots are provided. The output circuits are only capable of sinking current.

D/A Output PAC

Used to convert 10-bit digital output data to an analog signal with an accuracy of 0.1%. Options include various ranges of positive, negative, and bipolar analog voltage outputs and various ranges of current analog outputs. This output option may be used to drive such devices as CRT display terminals, analog servo systems, X-Y plotters, etc.

Alarm Printer and Logging Typewriter Option PACs

Used to interface between the H112 and up to four teletype receive only alarm printers (ASR35 RO) and up to two IBM Model "B" output logging typewriters respectively. Both units operate at a rate of ten characters per second, and make it possible to periodically log tabular data relating to process operation or to record data which exceeds some preset limits.

ANALOG SUBSYSTEM (ADCON)

The analog subsystem options provide a means of monitoring analog signals, converting them into digital form and entering them onto the H112 I/O bus. Three versions are available, as described below. Each consists of a logic portion (contained in a $1 \times 3 \mu\text{-BLOC}$) which provides the controls for input channel selection and data transfer to the H112, and an analog portion which provides for input multiplexing, buffering and A/D conversion. The available models are:

112-50-1

A 10-bit bipolar A/D converter, preceded by a single ended buffer amplifier and a 16 channel multiplexer. The multiplexer is expandable to 32 channels. The analog portion of this version is contained in a $2 \times 3 \mu\text{-BLOC}$.

112-50-2

A 12-bit A/D converter preceded by a sample and hold circuit and a 16 channel multiplexer. This model is expandable to 32 analog input channels. The analog portion of this model is contained in a $2 \times 3 \mu\text{-BLOC}$.

112-50-3

This model is identical to the 112-50-2 except that the analog multiplexer is expandable to 64 channels in blocks of 16. The analog portion of this model is contained in a $3 \times 3 \mu\text{-BLOC}$.

The Analog Subsystem option is housed in a separate H112 option drawer and requires its own power supply. Connection to the mainframe I/O bus is by a 14-foot (max) cable. Operation can be in any of four modes (all under program control) including either random or sequential input channel selection and either synchronous or asynchronous data conversion and entry. Asynchronous operation would normally be used to monitor relatively slowly varying signals. For such applications, Model 112-50-1 would normally be adequate. For rapidly varying signals, the sample/hold circuits in models 112-50-2 and 112-50-3 become essential. To reduce aperture time still further, conversions may be synchronized with some external signal in the synchronous mode.

Characteristic Data:

Model 112-50-1

A/D conversion time:	10 μsec
Accuracy (25°C):	$\pm .06\%$
Accuracy (over specified temperature range):	$\pm .5 \text{ LSB}$
Aperture Time:	10 μsec

Model 112-50-2

A/D conversion time: 20 μ sec
Accuracy (25°C): $\pm .025\%$
Accuracy (over specified temperature range): ± 1.5 LSB
Aperture Time: 150 nsec

Model 112-50-3

A/D conversion time: 20 μ sec
Accuracy (25°C): $\pm .025\%$
Accuracy (over specified temperature range): ± 1.5 LSB
Aperture Time: 150 nsec

H112 OPTION

HIGH SPEED PAPER TAPE PUNCH (BRPE)

The High Speed Punch option consists of a Model BRPE 11 paper tape punch (rack mounted) with the drive circuitry and control logic contained in a 1 x 3 μ -BLOC located in the H112 mainframe housing. During operation, an initial five second delay is built into the interface logic to allow the punch motor to come up to speed. Following this, the punch accepts data under interrupt control at a rate of 110 characters per second. If no data is supplied for a period of twenty seconds or more, the punch is shut down to prevent undue wear on the mechanism. The option normally includes both the interface electronics plus the punch and its power supply (both mounted in a separate slide chassis), but the interface circuitry may be ordered separately.