APPLE-1
OPERATION MANUAL

APPLE COMPUTER COMPANY
770 Welch Road
Palo Alto, Calif. 94304
The Apple Computer is a complete microprocessor system, consisting of a MOS Technology 6502 microprocessor and support hardware, integral video display electronics, dynamic memory and refresh hardware, and fully regulated power supplies. It contains resident system monitor software, enabling the user, via the keyboard and display, to write, examine, debug, and run programs efficiently; thus being an educational tool for the learning of microprocessor programming, and an aid in the development of software.

The integral video display section and the keyboard interface renders unnecessary the need for an external teletype. The display section contains its own memory, leaving all of RAM for user programs, and the output format is 40 characters/line, 24 lines/page, with auto scrolling. Almost any ASCII encoded keyboard will interface directly with the Apple system.

The board has sockets for up to 8K bytes of the 16 pin, 4K type, RAM, and the system is fully expandable to 65K via the edge connector. The system uses dynamic memory (4K bytes supplied), although static memory may also be used. All refreshing of dynamic memory, including all "off-board" expansion memory, is done automatically. The entire system timing, including the microprocessor clock and all video signals, originates in a single crystal oscillator.

Further, the printed circuit board contains a "breadboard area", in which the user can add additional "on-board" hardware (for example, extra PIA's, ACIA's, EROM's, and so on).

This manual is divided into three Sections:

Section I GETTING THE SYSTEM RUNNING.
Section II USING THE SYSTEM MONITOR. (listing included)
Section III EXPANDING THE SYSTEM.

Please read Section I thoroughly, before attempting to "power-up" your system, and study Section III carefully before attempting to expand your system. In addition to this manual, Apple "Tech Notes" are available which contain examples of expansion hardware and techniques.

SECTION I
GETTING THE SYSTEM RUNNING

The Apple Computer is fully assembled, tested, and burned in. The only external devices necessary for operation of the system are: An ASCII encoded keyboard, a video display monitor, and AC power sources of 8 to 10 Volts (RMS) @ 3 amps and 28 Volts (RMS) @ 1 amp. The following three articles describe the attachment of these devices in detail.

Keyboard:

Any ASCII encoded keyboard, with positive DATA outputs, interfaces directly with the Apple system via a "DIP" connector. If your keyboard has negative logic DATA outputs (rare), you can install inverters (7404) in the breadboard area. The strobe can be either positive or negative, of long or short duration. The "DIP" keyboard connector (B4) has inputs for seven DATA lines, one STROBE line, and two normally-open pushbutton switches, used for RESET (enter monitor), and CLEAR SCREEN (see schematic diagram, sheet 3 of 3, for exact circuitry). This keyboard connector also supplies three voltages, (+5V, +12V, and -12V) of which one or more may be necessary to operate the keyboard. Pin 15 of the keyboard connector (B4) must be tied to +5V (pin 16) for normal operation.

NOTE: The system monitor accepts only upper-case alpha (A-F, R).

It is therefore convenient, though it's not essential, to have a keyboard equipped with upper-case alpha lock (usually in the electronics). Either of the following suggested circuits may be used to provide alpha lock capability, if needed, and can be built in the breadboard area.
The Apple Computer outputs a composite video signal (composite of sync and video information) which can be applied to any standard raster-scan type video display monitor. The output level is adjustable with the potentiometer located near the video output Molex connector, J2. The additional two outside pins on the Molex connector supply +5 and +12 volts, to be used in future Apple accessories. The composite video signal can also be modulated at the proper RF frequency, with an inexpensive commercially available device, and applied to the antenna terminals of a home television receiver. Since the character format is 40 characters/line, all television receivers will have the necessary bandwidth to display the entire 40 characters. Two large manufacturers of video display monitors, which connect directly with the Apple Computer, are Motorola and Ball. The mating four-pin Molex connector is provided.

AC Power Sources:

Two incoming AC power sources are required for operation: 8 to 10 VAC (RMS) at 3 amps, and 28 VAC (RMS) Center-Tapped at 1 amp. These AC supplies enter the system at the Molex connector, J1. The 8 to 10 volts AC provides the raw AC for the +5 volt supply, while the 28 VCT supplies the raw AC for the +12 and -12 volt supplies, and the -5V supply is derived from the -12V regulated output.

The board, as supplied, requires no more than 1.5 amps DC from the +5V supply, while the regulator is capable of supplying 3 amps. The remaining 1.5 amps DC from the +5V supply is available for user hardware expansion (provided suitable transformer ratings are employed).

A suitable source of the raw AC voltages required, are two commercially available transformers; Stancor P/N P-8380 or equivalent (8 to 0 volts at 3 amps), and Stancor P/N P-8667 or equivalent (28VCT at 1 amp). Simply wire the secondaries to the mating six-pin Molex connector supplied, and wire the primaries in parallel, as shown in the schematic diagram (power supply section, Dwg. No. 00101, sheet 3 of 3).

TEST PROGRAM

After attaching the keyboard, display, and AC power sources, you can try a simple program to test if your system and the attachments are functioning together properly. While it does not test many possible areas of the microprocessor system, the test program will test for the correct attachment of the keyboard, display, and power supplies.

FIRST:
Hit the RESET button to enter the system monitor. A backslash should be displayed, and the cursor should drop to the next line.

SECOND:
Type— [a9 h 0 a A b 2 a F F b E 8 b 8 A b 4 C b 2 b 0 (RET)]
(0 is a zero, NOT an alpha "0"; b means blank or space; and (RET) hit the "return" key on the keyboard)

THIRD:
Type— [0 A (RET)]
(This should print out, on the display, the program you have just entered.)

FOURTH:
Type— [R (RET)]
(R means run the program.)

THE PROGRAM SHOULD THEN PRINT OUT ON THE DISPLAY A CONTINUOUS STREAM OF ASCII CHARACTERS. TO STOP THE PROGRAM AND RETURN TO THE SYSTEM MONITOR, HIT THE "RESET" BUTTON. TO RUN AGAIN, TYPE : R (RET).
SECTION II USING THE SYSTEM MONITOR

The Hex Monitor is a PROM program in locations FF00 to FFFF (hex) which uses the keyboard and display to perform the front panel functions of examining memory and running programs. The monitor program is entered by hitting (RESET), which displays backslash – return. A backslash alone (cursor remains on same line as backslash) indicates bad page 0 RAM.

Commands are typed on a "line-at-a-time" basis with editing. Each line may consist of any number of commands (up to 128 characters). None are executed until (RETURN) is typed. The (SHIFT-O) (backarrow) backspaces and echoes an underline. The (ESC) cancels a line and echoes backslash-return.

One or more hexadecimal digits (0–9, A–F) are used for address and data values. Addresses use the four least significant digits of a group, and data values, the two least significant digits. The following examples illustrate the variety of acceptable commands:

1. Opening a location (examining the contents of a single address).
   USER TYPES/ 4F (RET)
   MONITOR TYPES/ $04F: 0F (contents of 4F)

2. Examining a block; from the last examined location, to a specified one.
   USER TYPES/ .5A (RET)
   MONITOR TYPES/ $050: 00 01 02 03 04 05 06 07 $058: 08 09 0A

Note: 4F is still considered the most recently opened location.

3. Combining examples 1 and 2 to print a block of memory in a single command.
   USER TYPES/ 4F 5A (RET)
   MONITOR TYPES/ $050: 00 01 02 03 04 05 06 07 $058: 08 09 0A

Note: Only the first location of the block (4F) is considered "opened".

4. Examining several individual locations at once.
   USER TYPES/ 4F 52 b 56 b 58 5A (RET)
   MONITOR TYPES/ $04F: 0F $052: 02 $056: 06 $058: 08 09 0A

Note: 56 is considered the most recently "opened" location. The "b" is a blank or comma, and is a delimiter for separation purposes only. A string of delimiters has the same effect as a single one (bbb is as effective as b).

5. Examining several blocks of memory at once.
   USER TYPES/ 4F 52 b 56 b 58 5A (RET)
   MONITOR TYPES/ $04F: 0F $052: 02 $056: 06 $058: 08 09 0A

Note: 58 is considered the most recently "opened" location. Refer to example 2.

6. Examining successive blocks.
   USER TYPES/ 4F 52 (RET)
   MONITOR TYPES/ $04F: 0F $052: 00 01 02 $056: 06 07 $058: 08 09 0A

Note: Location 30 is considered opened and now contains 30.

7. Depositing data in a single location.
   USER TYPES/ 30 A0 (RET)
   MONITOR TYPES/ $030: FF (prior contents)

Note: Location 30 is considered opened and now contains 30.

8. Depositing data in successive locations from that last used in a deposit command.
   USER TYPES/ : A1 b A2 b A3 b A4 b A5 (RET)
   (This deposits A1 in location 31, A2 in 32, and so on.)

9. Combining examples 7 and 8 in a single command.
   USER TYPES/ 30 A0 b A1 b A2 b A3 b A4 b A5 (RET)
   MONITOR TYPES/ $030: FF
   USER TYPES/ : A2 b A3 (RET)
   USER TYPES/ : A4 b A5 (RET)

10. Depositing data in successive locations with separate commands.
    USER TYPES/ 30 A0 b A1 (RET)
    MONITOR TYPES/ $030: FF
    USER TYPES/ : A2 b A3 (RET)
    USER TYPES/ : A4 b A5 (RET)

NOTE: Capital letters enclosed in parenthesis represent single keystrokes. Example: (RET) means hit the "return" key.
Note: A colon in a command means "start depositing data from the most recently deposited location, or if none, then from the most recently opened one.

11. Examining a block, then depositing into it.

   USER TYPES/  30.35 (RET)
   MONITOR TYPES/  @039: A9 A1 A2 A3 A4 A5 A6
   USER TYPES/  :B0 b B1 b B2 b B3 b B4 b B5 (RET)

Note: New data deposited beginning at most recently opened location (39)

12. Run a program at a specified address.

   USER TYPES/  19F0 R (RET)
   MONITOR TYPES/  19F0: A9 (contents)

Note: The cursor is left immediately to the right of the "A9"; it is not returned to the next line.

13. Run at the most recently examined location.

   USER TYPES/  19F0 (RET)
   MONITOR TYPES/  19F0: A9
   USER TYPES/  R (RET)

14. Enter a program into memory and run it in one line.

   USER TYPES/  40: A9 b 0 b 20 b EF b FF b 38 b 69 b
   b 4C b 40 b 0 R (RET)
   MONITOR TYPES/  40: FF (prior contents of 40)

15. An "on line" error correction.

   USER TYPES/  40: A1 b A2 b A3A4A5A6 b A7
   (data A6 will be loaded in location 42)

   USER TYPES/  40506070: AA
   (data AA will be loaded in location 6070)

16. Useful routines in monitor which can be accessed by user programs.

   GETLINE:    location FF1F:
      monitor entry point
      (jumping to FF1F will enter monitor and echo carriage return. You can then examine memory locations with the monitor.)

   ECHO:       location FFEF:
      prints one byte (ASCII)
      (data from "A" (accumulator), contents of "A" not disturbed. Example: 20 b EF b FF (JRS ECHO)).

   PRBYTE:     location FFDC:
      prints one byte (HEX)
      (data from "A", contents of "A" disturbed.)

   PRHEX:      location FFE5:
      prints one hex digit
      (data from four least significant bits of "A", contents of "A" disturbed.)

NOTE: RAM locations 0024 to 002B are used as index pointers by the monitor, and are invalid for user use, when using monitor. Also, locations 0200 to 027F are used as input buffer storage, and are also invalid for user use when using the monitor.
| FF 00 | D8 | RESET | CLD | Clear decimal arithmetic mode. |
| FF 01 | 58 | CLI   |     |                             |
| FF 02 | AF 7F | LDY #$7F |     | Mask for DSP data direction register. |
| FF 04 | 8C 12 D0 | LDA #$A7 |     | Set it up. |
| FF 07 | A9 A7 | STA KBD CR |     | KBD and DSP control register mask. |
| FF 09 | 8D 11 D0 | STA DSP CR |     | Enable interrupts, set CAI, CBI, for positive edge sense/output mode. |
| FF 0C | 8D 13 D0 | CMP #$9B |     | "\"" |
| FF 0F | C9 DF | BEQ BACKSPACE |     | Yes. |
| FF 11 | F0 13 | CMP #$DF |     | ESC. |
| FF 13 | C9 9B | BEQ ESCAPE |     | Yes. |
| FF 15 | F0 03 |     |     | Advance text index. |
| FF 17 | C8 | INY |     | Auto ESC if >127. |
| FF 18 | 10 0F | BPL NEXTCHAR |     | "\"" |
| FF 1A | A9 DC | LDA #$DC |     | Output it. |
| FF 1C | 20 EF FF | JSR ECHO |     | CR. |
| FF 1F | A9 8D | LDA #$8D |     | Output it. |
| FF 21 | 20 EF FF | JSR ECHO |     | Initialize text index. |
| FF 24 | A0 01 | LDY #$01 |     | Backup text index. |
| FF 26 | 88 | DEY |     | Beyond start of line, reinitialize. |
| FF 27 | 30 F6 | BMI GETLINE |     | Key ready? |
| FF 29 | AD 11 D0 | LDA KBD CR |     | Loop until ready. |
| FF 2C | 10 FB | BPL NEXTCHAR |     | Load character. B7 should be '1'. |
| FF 2E | AD 10 D0 | LDA KBD |     | Add to text buffer. |
| FF 31 | 90 00 02 | STA IN, Y |     | Display character. |
| FF 34 | 20 EF FF | JSR ECHO |     | CR? |
| FF 37 | C9 8D | CMP #$8D |     | No. |
| FF 39 | D0 D4 | BNE NOTCR |     | Reset text index. |
| FF 3B | A0 FF | LDY #$FF |     | For XAM mode. |
| FF 3D | A9 00 | LDA #$00 |     | 0→X. |
| FF 3F | AA | TAX |     | Leaves $7B if setting STOR mode. |
| FF 40 | 0A | SETSTOR | ASL | $00 = XAM, $7B = STOR, $AE = BLOK XAM. |
| FF 41 | 85 2B | SETMODE | STA MODE | Advance text index. |
| FF 43 | C8 | BLSKIP | INY | "\"" |
| FF 44 | B9 00 02 | NEXT ITEM | LDA IN, Y | Key ready? |
| FF 47 | C9 8D | CMP #$8D | LDA IN, Y | Loop until ready. |
| FF 49 | F0 D4 | BEQ GETLINE | CMP #$8D | Load character. B7 should be '1'. |
| FF 4B | C9 AE | CMP #$AE | CMP #$8D | Add to text buffer. |
| FF 4D | 90 F4 | BCC BLSKIP | Beq GETLINE | Display character. |
| FF 4F | F0 F0 | BEQ SETMODE | CMP #$BA | CR? |
| FF 51 | C9 BA | BEQ SETSTOR | CMP #$BA | No. |
| FF 53 | F0 EB | CMP #$D2 | CMP #$BA | Reset text index. |
| FF 55 | C9 D2 | BEQ RUN | CMP #$D2 | For XAM mode. |
| FF 57 | F0 3B |     | CMP #$D2 | 0→X. |
| FF 59 | 86 28 | STX L |     | Leaves $7B if setting STOR mode. |
| FF 5B | 86 29 | STX H |     | $00 = XAM, $7B = STOR, $AE = BLOK XAM. |
| FF 5D | 84 2A | STY YSAV |     | Advance text index. |
| FF 5F | B9 00 02 | NEXTHEX | LDA IN, Y | Get character for hex test. |
| FF 62 | 49 B0 | EOR #$B0 |     | Map digits to $0-9. |
| FF 64 | C9 0A | CMP #$0A |     | Digit? |
| FF 66 | 90 06 | BCC DIG |     | Yes. |
| FF 68 | 69 88 | ADC #$88 |     | Map letter "A"-"F" to $FA-FF. |
| FF 6A | C9 FA | CMP #$FA |     | Hex letter? |
| FF 6C | 90 11 | BCC NOTHEX |     | No, character not hex. |
| FF 6E | 0A | DIG | ASL | Hex digit to MSD of A. |
| FF 6F | 0A | ASL |     | Shift count. |
| FF 70 | 0A | ASL |     | Hex digit left, MSB to carry. |
6502 HEX MONITOR LISTING (continued)

FF75  26 28  ROL L  Rotate into LSD.
FF77  26 29  ROL H  Rotate into MSD's.
FF79  CA    DEX   Done 4 shifts?
FF7A  D0 F8  BNE HEXSHIFT
FF7C  C8    INY   No, loop.
FF7D  D0 E0  BNE NEXTHEX
FF7F  C4 2A  NOTHEX  Advance text index.
FF81  F0 97  CPY YSAV  Always taken. Check next character for hex.
FF83  24 2B  BIT MODE  Check if L, H empty (no hex digits).
FF85  50 10  BVC NOTSTOR  Yes, generate ESC sequence.
FF87  A5 28  LDA L  Test MODE byte.
FF89  81 26  STA (STL, X)  B6 = 0 for STOR, 1 for XAM and BLOCK XAM
FF8B  E6 26  INC STL  LSD's of hex data.
FF8D  D0 B5  BNE NEXTITEM  Store at current 'store index'.
FF8F  E6 27  INC STH  Increment store index.
FF91  4C 44 FF  TONENTITEM  Get next item. (no carry).
FF94  6C 24 90  JMP NEXTITEM  Add carry to 'store index' high order.
FF97  30 2B  NOTSTOR  Get next command item.
FF99  A2 92  LDX #$@2  Run at current XAM index.
FF9B  B5 27  SETADR  B7 = 0 for XAM, 1 for BLOCK XAM.
FF9D  95 25  STA STL-1, X  Byte count.
FF9F  95 23  STA XAML-1, X  Copy hex data to 'store index'.
FFA1  CA    DEX  And to 'XAM index'.
FFA2  D0 F7  BNE SETADR  Next of 2 bytes.
FFA4  D0 14  NXTPRTN  Loop unless X = 0.
FFA6  A9 8D  BNE PRDATA  NE means no address to print.
FFA8  20 EF FF  RUN  CR.
FFAB  A5 25  JMP (XAML)  Output it.
FFAD  20 DC FF  JSR ECHO  'Examine index' high-order byte.
FFA9  A0 24  JSR PRBYTE  Output it in hex format.
FFBF  A5 24  LDA XAMH  Low-order 'examine index' byte.
FFB2  20 DC FF  JSR PRBYTE  Output it in hex format.
FFB5  A9 BA  LDA #$BA  "".
FFB7  20 EF FF  JSR ECHO  Output it.
FFBA  A9 A0  jsR PRDATA  Blank.
FFBC  20 EF FF  JSR ECHO  Output it.
FFBD  A1 24  LDA (XAML, X)  Get data byte at 'examine index'.
FFC1  20 DC FF  JSR PRBYTE  Output it in hex format.
FFC4  86 2B  XAMNEXT  "".
FFC7  A5 24  STA XAML  Compare 'examine index' to hex data.
FFC8  C5 28  CMP L  Not less, so no more data to output.
FFCA  A5 25  LDA XAMH  Increment 'examine index'.
FFCC  E5 29  SBC H  Increment 'examine index'.
FFCE  B9 C1  BCS TONENTITEM  For MOD 8 = 0
FFDF  E6 24  INC XAML  Check low-order 'examine index' byte
FFD2  D0 92  BNE MOD8CHK  Always taken.
FFD4  E6 25  INC XAMH  Save A for LSD.
FFD6  A5 24  MOD8CHK  MSD to LSD position.
FFD8  29 97  LDA XAML  Output hex digit.
FFDA  10 C8  AND #$@7  Restore A.
FFDC  48  PRBYTE  Mask LSD for hex print.
FFDD  4A  LSR  Add "@".
FFDE  4A  LSR  Digit?
FFDF  4A  LSR
FEE0  4A  LSR
FEE1  20 E5 FF  JSR PRHEX
FEE4  68  PLA
FEE5  29 9F  PRHEX  Output hex digit.
FEE7  99 B0  ORA #$@F  Restore A.
FEE9  C9 BA  CMP #$@B
6502 HEX MONITOR LISTING (continued)

<table>
<thead>
<tr>
<th>Hex Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FFEB 92</td>
<td>BCC ECHO</td>
</tr>
<tr>
<td>FFED 69</td>
<td>ADC #$06</td>
</tr>
<tr>
<td>FFEE 2C</td>
<td>BIT DSP</td>
</tr>
<tr>
<td>FFF2 30</td>
<td>BMI ECHO</td>
</tr>
<tr>
<td>FFF4 8D</td>
<td>STA DSP</td>
</tr>
<tr>
<td>FFF7 69</td>
<td>RTS</td>
</tr>
<tr>
<td>FFF8 00</td>
<td>HEX MONITOR LISTING (continued)</td>
</tr>
<tr>
<td>FFFA 00</td>
<td>(unused)</td>
</tr>
<tr>
<td>FFFC FF</td>
<td>(RESET)</td>
</tr>
<tr>
<td>FFEE 00</td>
<td>(IRQ)</td>
</tr>
</tbody>
</table>

Yes, output it.
Add offset for letter.
DA bit (B7) cleared yet?
No, wait for display.
Output character. Sets DA.
Return.

Hardware Notes

Page 0 Variables

- XAML 24
- XAMH 25
- STL 26
- STH 27
- L 28
- H 29
- YSAV 2A
- MODE 2B

Other Variables

- IN 26-27F
- KBD D0-16
- KBD CR D0-11
- DSP D12
- DSP CR D13

KBD/DSP Interface

- B1
- B2
- B3
- B4
- B5
- B6
- B7

- PA0 D0-D7
- DATA BUS

- PA1 RS0
- A0

- PA2 RS1
- A1

- PA3 CS0
- A2

- PA4 CS1
- A3

- PA5 CS2
- A4

- PA6 E
- A5

- PA7 Vcc
- A6

- PA7 Vss
- A7

- CPA

- PA1 DA (UART style)
- DATA BUS

- PA0 DA (UART style)
- R/W

- CB1 R/W
- One Shot

- CB2
- R/W

- PB7
- ASCII to display

- PR6
- ASCII to display

- PB5

- PR4

- PB3

- PR2

- PB1

- PR1

- PB0

- PIA

- 6820
The Apple system can be expanded to include more memory and IO devices, via a 44-pin edge connector. The system is fully expandable to 65K, with the entire data and address busses, clocks, control signals (i.e. IRQ, NMI, DMA, RDY, etc.), and power sources available at the connector. All address lines are TTL buffered, and data lines can drive ten equivalent capacitive loads (one TTL load and 130pf) without external buffers. All clock signals are TTL. The Apple system runs at approximately 1 MHz (see spec sheet) and is fully compatible with 6800/6500 style timing.

Three power sources are available at the edge connector: +5 volts regulated, and raw DC (approximately +/- 14V) for the +12V, -12V, and -5V supplies. If +12V, -12V, or -5V supplies are required, EXTERNAL REGULATORS MUST BE USED. An excess of 1.5 amps from the "on-board" regulated +5V supply is available for expansion (assuming suitable transformer ratings are employed). Exercise great care in the handling of the raw DC, as no short-circuit protection is provided.

REFRESH:

Four out of every 65 clock cycles is dedicated to memory refresh. At the start of a refresh cycle (150 ns after leading edge of Q1), RF goes low, and remains low for one clock cycle. Q2 is inhibited during a refresh cycle, and the processor is held in Q1 (its inactive state). Dynamic memories, which must clock during refresh cycles, should derive their clock from Q0, which is equivalent to Q2, except that it continues during a refresh cycle. Devices, such as PIA's, will not be affected by a refresh cycle, since they react to Q2 only. Refer to Apple "Tech Notes" for a variety of interfacing examples.

SOFTWARE CONSIDERATIONS:

The sequences listed below are the routines used to read the keyboard or output to the display.

Read Key from KBD:

LDA KBD CR (D@ll)
BPL
LDA KBD DATA (Del@)

Output to Display:

BIT DSP (Dq12)
BPL
STA DSP (Da12)

PIA Internal Registers:

KBD Data Dq10
High order bit equals 1.

KBD Control Reg. Dq11
High order bit indicates "key ready". Reading key clears flag. Rising edge of KBD sets flag.

DSP DATA Dq12
Lower seven bits are data output. High order bit is "display ready" input (1 equals ready, 0 equals busy)

DSP Control Reg. Dq13
**SPECIFICATIONS**

**MICROPROCESSOR:** MOS TECHNOLOGY 6502

- **Microprocessor Clock Frequency:** 1.023 MHz
- **Effective Cycle Frequency:** 0.960 MHz (Including Refresh Waits)

**VIDEO OUTPUT:** Composite positive video, 75 ohms, level adjustable between zero and +5Vpp.

- **Line Rate:** 15734 Hz
- **Frame Rate:** 60.05 Hz
- **Format:** 40 characters/line, 24 lines; with automatic scrolling
- **Display Memory:** Dynamic shift registers (1K x 7)
- **Character Matrix:** 5 x 7

**RAM MEMORY:** 16-pin, 4K Dynamic, type 4096 (2104)

- **On-board RAM Capacity:** 8K bytes (4K supplied)

**POWER SUPPLIES:** +5 Volts @ 3 amps, +/- 12 Volts @ 0.5 amps, and -5 Volts @ 0.5 amps

- **Input Power Requirements:** 8 to 10 Volts AC (RMS) @ 3 amps, 26 to 28 Volts AC (RMS) Center-Tapped, 1A
- **Recommended Transformers:**
  - Stancor # P-8380 or Triad F31-X
  - Stancor # P-8667 or Triad F40-X

APPLE COMPUTER COMPANY
770 Welch Road, Suite 154
Palo Alto, California 94304
Phone: (415) 326-4248
The Apple Computer Company hereby warrants each of its products, and all components therein contained, to be free from defects in materials and/or workmanship for a period of thirty (30) days from date of purchase. In the event of occurrence of malfunction, or other indication of failure attributable directly to faulty workmanship and/or material, then, upon return of the product to the Apple Computer Company, at 770 Welch Road, Palo Alto, California, 94304 (postage prepaid), the Apple Computer Company will, at its option, repair or replace said products or components thereof, to whatever extent Apple Computer Company shall deem necessary, to restore said product to proper operating condition. All such repairs or replacements shall be rendered by the Apple Computer Company, without charge to the customer.

The responsibility for the failure of any Apple Computer product, or component thereof, which, at the discretion of the Apple Computer Company, shall have resulted either directly or indirectly from accident, abuse, or misapplication of the product, shall be assumed by the customer, and the Apple Computer Company shall assume no liability as a consequence of such events under the terms of this warranty.

While every effort, on the part of Apple Computer Company, is made to provide clear and accurate technical instruction on the use, implementation, and application of its products, the Apple Computer Company shall assume no liability in events which may arise from the application of such technical instruction, nor shall the Apple Computer Company be held liable for the quality, interconnection, or application of peripheral products, which may have been recommended by Apple Computer Company, but which have not been supplied as part of the product.

This warranty contains and embodies the limits of responsibility of the Apple Computer Company, with regard to its products, and no other liability is expressed, implied, or should be assumed by the purchaser, and in no event shall the Apple Computer Company be held liable for the loss of time, effort, or transportation costs, nor for loss of potential profits or other consequential losses which might arise from the purchase, assembly, use, application, or subsequent sale of the products of Apple Computer Company, nor from any instructions and/or technical information thereto related.
If more than one source for RDY use open-collector gate 7401 (not '00)
(Slow ROM address decoded)

SLOW ROM

(SLOW ROM)

SINGLE STEP FOR 6502

ADDRESS DISPLAY

(ADDRESS DISPLAY)
NOTE 7
MICROPROCESSOR

NOTE 8
6502

NOTE 9

SEE NOTE 7

NOTE 10

NOTE 11

NOTE 12
UNREGULATED +12V

This drawing is only conditionally issued, and neither receipt nor possession thereof conveys any right to reproduce it or any part thereof, or manufacture an Apple Computer, under written permission from Apple Computer, Inc.
ALL OUTPUTS OF C1, 2519 (PINS 7 THRU 12), ARE PROVIDED WITH "PULL-DOWN" RESISTORS, 1000 OHMS EACH, TO -12V.

INTERCONNECTIONS BETWEEN SCHEMATIC SECTIONS ARE INDICATED BY AREAS.

ALL RESISTORS ARE IN OHMS, 1/4W, 9%

ALL CAPACITOR VALUES ARE IN MICROFARADS.

ALL DEVEES ARE TYPE 1101.

ALL GATES ARE 74 SERIES DEVICES.

(See additional notes, sheet 2/3)