APPLE - 1
OPERATION MANUAL

APPLE COMPUTER COMPANY
770 Welch Road
Palo Alto, Calif. 94304
SPECIFICATIONS

MICROPROCESSOR: MOS TECHNOLOGY 6502

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

VIDEO OUTPUT:

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:

On-board RAM Capacity: 16-pin, 4K Dynamic, type 4096 (2104)
8K bytes (4K supplied)

POWER SUPPLIES:

Input Power Requirements: 45 Volts @ 3 amps, +/- 12 Volts @0.5 amp
and -5 Volts @ 0.5 amps
Recommended Transformers: 8 to 10 Volts AC (RMS) @ 3 amps,
26 to 28 Volts AC (RMS) Center-Tapped, 1A.

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 is a complete microprocessor system, consisting of a Mos Technology 32 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 uppercase alpha (A-F, R).

It is therefore convenient, though it’s not essential, to have a keyboard equipped with uppercase 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 10 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: 0 A9 b 0 b AA b 29 b EF b FF b E8 b 8A b 4b 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).
Clear decimal arithmetic mode.

Mask for DSP data direction register.

Set it up.

KBD and DSP control register mask.

Enable interrupts, set CA1, CB1, for positive edge sense/output mode.

"<-"?

Yes.

ESC?

Yes.

Advance text index.

Auto ESC if > 127.

" ".

Output it.

CR.

Output it.

Initialize text index.

Backup text index.

Beyond start of line, reinitialize.

Key ready?

Loop until ready.

Load character. B7 should be '1'.

Add to text buffer.

Display character.

CR?

No.

Reset text index.

For XAM mode.

O->X.

Leaves $7B if setting STOR mode.

$90 = XAM, $7B = STOR, $AE = BLOK XAM.

Advance text index.

Get character.

CR?

Yes, done this line.

",""?

Skip delimiter.

Set BLOCK XAM mode.

":"?

Yes, set STOR mode.

"R"?

Yes, run user program.

$90->L.

and H.

Save Y for comparison.

Get character for hex test.

Map digits to $9-9.

Digit?

Yes.

Map letter "A"-"F" to $FA-FF.

Hex letter?

No, character not hex.

Hex digit to MSD of A.

Shift count.

Hex digit left, MSB to carry.
i
6502 HEX MONITOR LISTING (continued)

FF75 26 28 ROL L
FF77 26 29 ROL H
FF79 C0 BNE HEXSHIFT
FF7A D0 F8 INY
FF7C C8 BNE NEXTHEX
FF7D D0 E0 NOTHEX
FF7F C4 2A CPY YSAV
FF81 F0 97 BEQ ESCAPE
FF83 24 2B BIT MODE
FF85 50 19 BVC NOTSTOR
FF87 A5 28 LDA L
FF89 81 26 STA (STL, X)
FF8B E6 26 INC STL
FF8D D0 B5 BNE NEXTITEM
FF8F E6 27 INC STH

FF91 4C 44 FF TONEXTITEM
FF94 6C 24 FF RUN
FF97 30 2B NOTSTOR
FF99 A2 02 BMI XAMNEXT
FF9B B5 27 SETADR
FF9D 95 25 LDA L-1, X
FF9F 95 23 STA STL-1, X

FFA1 CA DEX
FFA2 D0 F7 BNE SETADR
FFA4 D0 14 BNE PRDATA

FFA6 A9 8D LDA #$8D
FFA8 20 EF FF JSR ECHO
FFAB A5 25 LDA XAMH
FFAD 20 DC FF JSR PRBYTE

FFB0 A5 24 LDA XAML
FFB2 20 DC FF JSR PRBYTE
FFB5 A9 BA LDA #$BA
FFB7 20 EF FF JSR ECHO

FBBA A9 A0 PRDATA

FBBC 20 EF FF JSR ECHO

FFBE A1 24 LDA (XAML, X)

FFC4 86 2B XAMNEXT

FFC7 A5 24 LDA XAML

FFC8 C5 28 CMP L

FFCA A5 25 LDA XAMH

FFCG E5 29 SBC H

FFCE B0 C1 BCS TONEXTITEM

FFD0 E6 24 INC XAML

FFD2 D0 G2 BNE MOD8CHK

FFD4 E6 25 INC XAML

FFD6 A5 24 MOD8CHK

FFD8 29 07 LDA XAML

FFDA 10 C8 AND #$07

FFDC 48 PRBYTE

FFDD 4A PHA

FFDE 4A LSR

FFDF 4A LSR

FFE0 4A LSR

FFE1 20 E5 FF JSR PRHEX

FFE4 68 PLA

FFE5 29 0F PRHEX

FFE7 09 B0 AND #$0F

FFE9 C9 BA ORA #$BA

Rotates into LSD.
Rotates into MSD's.
Done 4 shifts?
No, loop.
Advance text index.
Always taken. Check next character for hex.
Check if L, H empty (no hex digits).
Yes, generate ESC sequence.
Test MODE byte.
B6 = 0 for STOR, 1 for XAM and BLOCK XAM
LSD's of hex data.
Store at current 'store index'.
Increment store index.
Get next item. (no carry).
Add carry to 'store index' high order.
Get next command item.
Run at current XAM index.
B7 = 0 for XAM, 1 for BLOCK XAM.
Byte count.
Copy hex data to
'store index'.
And to 'XAM index'.
Next of 2 bytes.
Loop unless X = 0.
NE means no address to print.
CR.
'Examine index' high-order byte.
Output it in hex format.
Low-order 'examine index' byte.
Output it in hex format.
':'.
Blank.
Output it.
Get data byte at 'examine index'.
Output it in hex format.
Compare 'examine index' to hex data.
Not less, so no more data to output.
Increment 'examine index'.
Check low-order 'examine index' byte
For MOD 8 = 0
Always taken.
Save A for LSD.

MSD to LSD position.

Output hex digit.
Restore A.
Mask LSD for hex print.
Add "$".
Digit?
6502 HEX MONITOR LISTING (continued)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>FFEB</td>
<td>90 92</td>
</tr>
<tr>
<td>FFED</td>
<td>69 96</td>
</tr>
<tr>
<td>FFEF</td>
<td>2C 12 D0 ECHO</td>
</tr>
<tr>
<td>FFF2</td>
<td>30 FB</td>
</tr>
<tr>
<td>FFF4</td>
<td>8D 12 D0</td>
</tr>
<tr>
<td>FFF7</td>
<td>60</td>
</tr>
<tr>
<td>FFF8</td>
<td>00 00 (unused)</td>
</tr>
<tr>
<td>FFFA</td>
<td>00 0F (NMI)</td>
</tr>
<tr>
<td>FFFC</td>
<td>00 FF (RESET)</td>
</tr>
<tr>
<td>FFFE</td>
<td>00 00 (IRQ)</td>
</tr>
<tr>
<td>BCC ECHO</td>
<td>Yes, output it.</td>
</tr>
<tr>
<td>ADC $06</td>
<td>Add offset for letter.</td>
</tr>
<tr>
<td>BIT DSP</td>
<td>DA bit (B7) cleared yet?</td>
</tr>
<tr>
<td>BMI ECHO</td>
<td>No, wait for display.</td>
</tr>
<tr>
<td>STA DSP</td>
<td>Output character. Sets DA.</td>
</tr>
<tr>
<td>RTS</td>
<td>Return.</td>
</tr>
</tbody>
</table>

HARDWARE NOTES

<table>
<thead>
<tr>
<th>Page 0 Variables</th>
<th>Other Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>XAML 24</td>
<td>IN 200-27F</td>
</tr>
<tr>
<td>XAMH 25</td>
<td>KBD D010</td>
</tr>
<tr>
<td>STL 26</td>
<td>KBD CR D011</td>
</tr>
<tr>
<td>STH 27</td>
<td>DSP D012</td>
</tr>
<tr>
<td>L 28</td>
<td>DSP CR D013</td>
</tr>
<tr>
<td>H 29</td>
<td>PIA</td>
</tr>
<tr>
<td>YSAV 2A</td>
<td>Decode A15, A14, A13, A12 to $DXXX</td>
</tr>
<tr>
<td>MODE 2B</td>
<td></td>
</tr>
</tbody>
</table>

KBD/DSP Interface

- KBD STROBE
- ASCII to display
- One Shot (3.5 usec)
- RDA (UART style)
- DA (UART style)
- ASCII to display
- PIA 6820
SECTION III

HOW TO EXPAND THE APPLE SYSTEM

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 01), RF goes low, and remains low for one clock cycle. 02 is inhibited during a refresh cycle, and the processor is held in 01 (it's inactive state). Dynamic memories, which must clock during refresh cycles, should derive their clock from 00, which is equivalent to 02, 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 02 only. Refer to Apple "TechNotes" for a variety of interfacing examples.

DMA:

The Apple system has full DMA capability. For DMA, the DMA control line tri-states the address buss, thus allowing external devices to control the buss. Consult MOS TECHNOLOGY 6502 Hardware Manual for details. (For DMA use, the solder jumper on the board, marked "DMA", must be broken.)

For the 6502 microprocessor, the RDY line is used to halt the processor for single stepping, or slow ROM applications. Refer to Apple "Tech Notes" for examples.

SOFTWARE CONSIDERATIONS:

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

Read Key from KBD:

\[
\text{LDA KBD CR (D011)} \\
\text{BPL} \\
\text{LDA KBD DATA (D010)}
\]

Output to Display:

\[
\text{BIT DSP (D012)} \\
\text{BPL} \\
\text{STA DSP (D012)}
\]

PIA Internal Registers:

KBD Data \( D010 \)

High order bit equals 1.

KBD Control Reg. \( D011 \)

High order bit indicates "key ready".

Reading key clears flag. Rising edge of KBD sets flag.

DSP DATA \( D012 \)

Lower seven bits are data output, high order bit is "display ready" input (1 equals ready, 0 equals busy)

DSP Control Reg. \( D013 \)
If more than one source for RDY use open-collector gate 7401 (not '00)

(Slow ROM address decoded)

SLOW ROM

(Note: Features not needed may be omitted)

SINGLE STEP FOR 6502

ADDRESS DISPLAY

[Diagram of address display circuit]

100 Aφ  A1
100 A2  A3
100 A4  A5
100 A6  A7
100 A8  A9
100 A10 A11
100 A12 A13
100 A14 A15
100 SYNC

SYNC

D0
D1
D2
D3
D4
D5

Q0
Q1
Q2
Q3
Q4
Q5

74174

CL

74174

CL

74174

CL

74174
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 the 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.