

# APPLE II 资料(一)

语言系统安装和操作手册

MX-80-II 打印机接口用户手册



# 目 录

|                   |        |
|-------------------|--------|
| 第一章 APPLE 机概况     | ( 1 )  |
| § 1.1 电源          | ( 1 )  |
| § 1.2 主板          | ( 1 )  |
| § 1.3 和APPLE 对话   | ( 1 )  |
| § 1.3.1 键盘        | ( 2 )  |
| § 1.3.2 读取键盘      | ( 3 )  |
| § 1.4 APPLE 视频显示  | ( 5 )  |
| § 1.4.1 视频插座      | ( 5 )  |
| § 1.4.2 屏幕显示格式    | ( 5 )  |
| § 1.4.3 屏幕存储器     | ( 6 )  |
| § 1.4.4 屏幕页       | ( 6 )  |
| § 1.4.5 屏幕开关      | ( 6 )  |
| § 1.4.6 程序文本方式    | ( 7 )  |
| § 1.4.7 低分辨率图象方式  | ( 8 )  |
| § 1.4.8 高分辨率图象方式  | ( 10 ) |
| § 1.5 其他输入/输出     | ( 12 ) |
| § 1.5.1 扬声器       | ( 12 ) |
| § 1.5.2 盒带接口      | ( 12 ) |
| § 1.5.3 游戏I/O 插座  | ( 13 ) |
| § 1.5.4 电铃输出      | ( 13 ) |
| § 1.5.5 一位输入      | ( 14 ) |
| § 1.5.6 模拟输入      | ( 14 ) |
| § 1.5.7 选通输出      | ( 14 ) |
| 第二章 和APPLE 对话     | ( 15 ) |
| § 2.1 标准输出        | ( 15 ) |
| § 2.1.1 暂停列表的动能   | ( 15 ) |
| § 2.1.2 屏幕显示范围的调整 | ( 15 ) |
| § 2.1.3 正常和反相的选择  | ( 16 ) |
| § 2.2 标准输入        | ( 17 ) |
| § 2.2.1 RDKEY 子程序 | ( 17 ) |

|                    |        |
|--------------------|--------|
| § 2.2.2 GETLN子程序   | ( 17 ) |
| § 2.3 ESCAPE 码     | ( 18 ) |
| § 2.4 总清周期         | ( 20 ) |
| <b>第三章 系统监控</b>    | ( 22 ) |
| § 3.1 进入监控程序       | ( 22 ) |
| § 3.2 建立并运行机器语言程序  | ( 28 ) |
| § 3.3 其他各种监控命令     | ( 32 ) |
| § 3.4 监控程序的特殊技巧    | ( 32 ) |
| § 3.5 建立你自己的命令     | ( 34 ) |
| § 3.6 监控命令摘要       | ( 35 ) |
| § 3.7 一些有用的监控子程序   | ( 36 ) |
| § 3.8 监控的特殊单元      | ( 39 ) |
| § 3.9 小汇编指令格式      | ( 39 ) |
| <b>第四章 存储器组织</b>   | ( 41 ) |
| § 4.1 RAM 存储器      | ( 41 ) |
| § 4.2 ROM 存储器      | ( 43 ) |
| § 4.3 I/O 单元       | ( 43 ) |
| <b>第五章 输入输出结构</b>  | ( 44 ) |
| § 5.1 内建 I/O       | ( 44 ) |
| § 5.2 外围插件的输入输出    | ( 45 ) |
| § 5.3 外围插件的输入输出空间  | ( 46 ) |
| § 5.4 外围插件的 ROM 空间 | ( 46 ) |
| § 5.5 输入输出程序的提示    | ( 47 ) |
| § 5.6 外围插件的中间结果RAM | ( 48 ) |
| § 5.7 CSW/KSW开关    | ( 49 ) |
| § 5.8 ROM的扩充       | ( 50 ) |
| <b>第六章 硬件配置</b>    | ( 51 ) |
| § 6.1 微处理器         | ( 51 ) |
| § 6.2 系统定时         | ( 52 ) |
| § 6.3 电源           | ( 53 ) |
| § 6.4 ROM 存储器      | ( 54 ) |
| § 6.5 RAM 存储器      | ( 55 ) |
| § 6.6 视频发生器        | ( 56 ) |
| § 6.7 视频输出插座       | ( 57 ) |
| § 6.8 内建的 I/O      | ( 57 ) |

|        |               |        |
|--------|---------------|--------|
| § 6.9  | “用户 1” 转接线    | ( 58 ) |
| § 6.10 | 游戏 I/O 插座     | ( 58 ) |
| § 6.11 | 键盘            | ( 59 ) |
| § 6.12 | 键盘插座          | ( 59 ) |
| § 6.13 | 盒带接口插件        | ( 60 ) |
| § 6.14 | 电源插座          | ( 60 ) |
| § 6.15 | 扬声器           | ( 61 ) |
| § 6.16 | 外围插座          | ( 61 ) |
| 附录 A   | 6502 微处理器指令系统 | ( 56 ) |
| 附录 B   | 专用单元          | ( 74 ) |
| 附录 C   | 自动启动 ROM 清单   | ( 80 ) |
|        | 监控程序 ROM 清单   | (116 ) |
|        | 符号表数字命令       | (153 ) |
|        | 符号表字符命令       | (156 ) |

# 第一章 APPLE机概况

## § 1.1 电 源

机器内部左侧的金属盒就是电源。提供+5V, -5.2V, +11.8V和-12.0V四种电压。它是高频开关式电源,具有多种保护性能,以保证不同的输出电源之间的稳定。计算机的主电源线直接插入电源的后部,电源开关也装在电源上,以防接触电源的高压。

## § 1.2 主 板

绿色的大印制板装在机壳的最底部,它就是计算机本身。在该板(通常称作主板)上有大约80片集成电路和少量其他元件,在板的中央,就在其后部8个镀金槽号的前面是一块最大的集成电路,它是APPLE的大脑:6502微处理器,在APPLE中它以每秒1023000个机器周期的速度运行,在一秒钟内能进行50万次以上的加法或减法运算。寻址范围为65,536字节。具有56种指令,13种寻址方式。

在6502微处理器的下方有6个IC插座,可插入略小的组件。这些组件是APPLE的只读存储器(ROM)ROM中有许多开机后要立即用到的程序,如APPLE自动启动监控程序和扩展的BASIC解释程序。

ROM的右下方,在白色方框中装有24个IC插座分三行,每行8个插座可部分地或全部地装上IC,它们是APPLE的主存储器(RAM),存储容量可为4,096字节到49,152字节。每行可装8片IC,可以是4K的芯片,也可以是16K的芯片,这两种器件可以组合起来使用,但是在一行之内必须为同一个品种的芯片。这样,APPLE主板可有9种RAM存储容量,RAM存储器存储运行程序时所用的全部程序和数据。但注意,去掉电源后存储在RAM中的信息随即消失。

APPLE主板上的其他器件有以下功能:

控制信息从计算机的一部分流向另一部分;

搜集外部的数据;

把信息显示在电视屏幕上;

和由扬声器发声响。

主板的后部有8个外围槽号,每个槽号可插入一个外围插件,其功能如下;

扩充RAM;

扩充ROM;

联打印机;

联其他输入/输出设备。

## § 1.3 和APPLE对话

你与你的APPLE联系是通过手指头,APPLE所用的大部分程序和语言都要你通过键

盘来表达。此键盘类似一般的打字机键盘，只有少量特殊键略有变动外，大致相同。关于键上的意义，请查阅BASIC编程手册和扩展BASIC指南。

你是用手指来表达语言，用眼睛来“听”。APPLE通过在黑白或彩色电视屏幕上显示文字，数字，符号和图象来告诉你它现在地作什么。

#### § 1.3.1 键盘

APPLE的键盘规格：

键数：52键

编码：大写ASCII码

编码数量：91

输出：7位，加一个选通脉冲

电源要求：+ 5 V 120mA

- 12V 50 mA

换档键：2个键

特殊键：CTRL

ESC

RESET

REPT

←→

存贮单元：

16进制

十进制

数据

\$ C000

49152 - 16384

清除

\$ C010

49168 - 16368

APPLE有一个打字机式的52键键盘，采用美国标准信息交换码（ASCII）。96个大写ASCII字笼中的91个可由键盘直接产生。表2列出键和ASCII代码的对照。

键盘通过一个16线电缆与主板连接，16线电缆附有一个插头，可插入标准的IC插座。该电缆的一端联至键盘，另一端的插头插入主板的键盘插座，该插座在主板的前方，在键盘的下方。该插座电气说明见第六章。

APPLE上使用的大部分语言都有命令或语句，可简化从键盘输入程序（例如BASIC中的INPUT和GET语句），程细也能直接读取键盘。

表2 键所对应的ASCII码

| KEY   | Alone  | CTRL   | SHIFT  | Both   | KeY    | Alone | CTRL  | SHIFT | Both  |
|-------|--------|--------|--------|--------|--------|-------|-------|-------|-------|
| Space | \$ A 0 | \$ A 0 | \$ A 0 | \$ A 0 | RETURN | \$ 8D | \$ 8D | \$ 8D | \$ 8D |
| 0     | \$ B 0 | \$ B 0 | \$ B 0 | \$ B 0 | G      | \$ C7 | \$ 87 | \$ C7 | \$ 87 |
| 11    | \$ B1  | \$ B1  | \$ A1  | \$ A1  | H      | \$ C8 | \$ 88 | \$ C8 | \$ 88 |
| 21    | \$ B2  | \$ B2  | \$ A2  | \$ A2  | I      | \$ C9 | \$ 89 | \$ C9 | \$ 89 |
| 3#    | \$ B3  | \$ B3  | \$ A3  | \$ A3  | J      | \$ CA | \$ 8A | \$ CA | \$ 8A |
| 4 \$  | \$ B4  | \$ B4  | \$ A4  | \$ A4  | K      | \$ CB | \$ 8B | \$ CB | \$ 8B |
| 5 %   | \$ B5  | \$ B5  | \$ A5  | \$ A5  | L      | \$ CC | \$ 8C | \$ CC | \$ 8C |
| 6 &   | \$ B6  | \$ B6  | \$ A6  | \$ A6  | M      | \$ CD | \$ 8D | \$ DD | \$ 9D |
| 7'    | \$ B7  | \$ B7  | \$ A7  | \$ A7  | N▲     | \$ CE | \$ 8E | \$ DE | \$ 9E |
| 8 C   | \$ B8  | \$ B8  | \$ A8  | \$ A8  | O      | \$ CF | \$ 8F | \$ CF | \$ 8F |
| 9)    | \$ B9  | \$ B9  | \$ A9  | \$ A9  | P@     | \$ D0 | \$ 90 | \$ C0 | \$ 80 |
| : *   | \$ BA  | \$ BA  | \$ AA  | \$ AA  | Q      | \$ D1 | \$ 91 | \$ D1 | \$ 91 |
| ; +   | \$ BB  | \$ BB  | \$ AB  | \$ AB  | R      | \$ D2 | \$ 92 | \$ D2 | \$ 92 |
| , <   | \$ AC  | \$ AC  | \$ BC  | \$ BC  | S      | \$ D3 | \$ 93 | \$ D3 | \$ 93 |
| - =   | \$ AD  | \$ AD  | \$ BD  | \$ BD  | T      | \$ D4 | \$ 94 | \$ D4 | \$ 94 |
| • >   | \$ AE  | \$ AE  | \$ BE  | \$ BE  | U      | \$ D5 | \$ 95 | \$ D5 | \$ 95 |
| / ?   | \$ AF  | \$ AF  | \$ BF  | \$ BF  | V      | \$ D6 | \$ 96 | \$ D6 | \$ 96 |
| A     | \$ C1  | \$ 81  | \$ C1  | \$ 81  | W      | \$ D7 | \$ 97 | \$ D7 | \$ 97 |
| B     | \$ C2  | \$ 82  | \$ C2  | \$ 82  | X      | \$ D8 | \$ 98 | \$ D8 | \$ 98 |
| C     | \$ C3  | \$ 83  | \$ C3  | \$ 83  | Y      | \$ D9 | \$ 99 | \$ D9 | \$ 99 |
| D     | \$ C4  | \$ 84  | \$ C4  | \$ 84  | Z      | \$ DA | \$ 9A | \$ DA | \$ 9A |
| E     | \$ C5  | \$ 85  | \$ C5  | \$ 85  | →      | \$ 88 | \$ 88 | \$ 88 | \$ 88 |
| F     | \$ C6  | \$ 86  | \$ C6  | \$ 86  | ←      | \$ 95 | \$ 95 | \$ 95 | \$ 95 |
|       |        |        |        |        | ESC    | \$ 9B | \$ 9B | \$ 9B | \$ 9B |

## § 1.3.1 读取键盘

键盘送出 7 位信息，这 7 位信息形成一个字符，这 7 位信息和另一个表示按键的信号，存于固定存贮单元供所有程序使用，程序通过读该单位的内容来辨认键盘的现行状态。在按下键盘上的一个键以后该键盘存贮器单元变成 128 或更大，这个特殊数值就表示按的是什么键。表 3 是 ASCII 字符和它对等的数值之间的对照。这个存贮单元会永远记住这个特殊数值，直到你按了其他键，它的值才会改变，或者是你的程序告诉这个单元将它的值改变或忘掉。

表3 ASCII字符集

| Decimal: | 128   | 144   | 160   | 176   | 192   | 208   | 224   | 240   |
|----------|-------|-------|-------|-------|-------|-------|-------|-------|
| Hex:     | \$ 80 | \$ 90 | \$ A0 | \$ B0 | \$ C0 | \$ D0 | \$ E0 | \$ F0 |
| 0        | \$ 0  | nnl   | dle   |       | 0     | @     | P     | p     |
| 1        | \$ 1  | soh   | dc1   | !     | 1     | A     | Q     | q     |
| 2        | \$ 2  | stx   | dc2   | "     | 2     | B     | R     | r     |
| 3        | \$ 3  | etx   | dc3   | *     | 3     | C     | S     | s     |
| 4        | \$ 4  | eot   | dc4   | \$    | 4     | D     | T     | t     |
| 5        | \$ 5  | eno   | nak   | %     | 5     | E     | U     | u     |
| 6        | \$ 6  | ack   | syn   | &     | 6     | F     | V     | v     |
| 7        | \$ 7  | bel   | etb   | ,     | 7     | G     | X     | w     |
| 8        | \$ 8  | bs    | can   | (     | 8     | H     | W     | x     |
| 9        | \$ 9  | ht    | em    | )     | 9     | I     | Y     | y     |
| 10       | \$ A  | lf    | sub   | *     | :     | J     | Z     | z     |
| 11       | \$ B  | vt    | esc   | *     | ;     | K     | [     | {     |
| 12       | \$ C  | ff    | fs    | ,     | <     | L     | /     |       |
| 13       | \$ D  | cr    | gs    | —     | =     | M     | ]     | }     |
| 14       | \$ E  | so    | rs    | ,     | >     | N     | ^     | —     |
| 15       | \$ F  | si    | us    | /     | ?     | O     | -     | o     |
|          |       |       |       |       |       |       |       | rab   |

一旦程序接收而且知道已经按下的一个键，它会马上告诉键盘存贮单元将目前记的字符数值“释放”，以准备接收新的按键。它的作法是读了键盘存贮单元之后再读另一个特殊单元，此时键盘存贮单元记的数值，立即减至128以下，而且它一直保持这个小于128的值，直到你再按其他的键。这个动作称作“清除键盘选通”。程序可以读或写此特殊存贮单元，读写的数值是不关紧要的。它仅仅是清除键选通的参考单元，由于我们每次读完一个键，只是去掉其选通脉冲，因此要恢复原来的键码时，只要把键盘存贮单元中的值再加上128(16进制为\$ 80)即可。

表1 键盘的特殊单元

| 单元:<br>十六进制 | 十进制           | 说明                    |
|-------------|---------------|-----------------------|
| \$ C000     | 49152 - 16384 | Keyboard Data         |
| \$ C010     | 49168 - 16368 | Clear Keyboard Strobe |

**RESET**键(在键盘右上方)并不产生一个ASCII码，它直接连到微处理器，当此键一被按下之后，一切处理均停止。当此键释放后计算机开始一个清除周期，详见第二章。

**CTRL**和**SHIFT**键本身也不产生键码，它们仅仅是改变其他键所产生的键码。



**REPT** 键,若是单独被按下,它就产生和上一次按的键,相同的键码,若是你同时按  
时 **REPT** 键和其他一个字符键的话,其作用就象你以每秒按10次键的速率,重复这个键一  
样。或者释放字符键,或者释放 **REPT** 键,重复都会停止。

在键盘左下角有一个指示灯,指示是否有电。

表3中,2个或3个小写字母的字母组是标准的ASCII控制符的缩写。并非此表中的全部都能由键盘产生,尤其最右两列(小写字母)。〔(左括号)/(斜杠)、-(底线)和控制符“fs”、“us”及“rub”等都不能使用。

在表3中,每个字符的ASCII码的数值,就是将此字符所在行和所在列的数值相加,即得此字符十进制或十六进制数值。

## § 1.4 APPLE视频显示

APPLE视频显示。

显示型式: 存贮器在系统RAM中。

显示方式: 程序文本方式,低分辨率图象方式和高分频率图象方式。

程序文本显示容量: 960字符(24行,40列)

字符型式:  $5 \times 7$  点阵

字符集: 大写ASCII字符。共64个字符

字符型: 正常、反相、闪烁

图象容量: 低分辨率图象方式,分成1.920方块,  $40 \times 48$  矩阵,高分辨率图象方式,分成53,760点,  $280 \times 192$  矩阵

颜色的数目: 低分辨图象方式16种

高分辨图象方式6种

### § 1.4.1 视频插座

在主板的后面右角处有一个标着“VIDEO”的金属插座。该插座使APPLE和监视器之间可用电缆联结。电缆的一端应为RCA录音公插头,以便插入APPLE,另一端的插头应与使用的具体设备相配。由APPLE出来的视频信号为混合的彩色,正问视频信号。在联接插座附有一个小电位器,可调视频输出信号的大小,可由0伏调到1伏峰值。

另外还有二处视频信号输出,信号峰值2伏,不可调。

### § 1.4.2 屏幕显示格式

APPLE的屏幕显示格式有三种:

#### (1) 程序文本:

APPLE能显示24行的数字,特殊符号和大写字母,每行40个字符。这些字符由高7点,宽5点的点矩阵所组成,每个字符两边及上边均空一点。

#### (2) 低分辨图象:

APPLE能以  $40 \times 48$  的矩阵,显示1920个彩色方块,每个方块的色彩各有16种颜色。方块和方块之间没有空白,因此,两个相邻的同一种颜色的方块就构成了一个更大的方块。

### (3) 高分辨率图象:

APPLE II 能显示宽280点, 高192点的彩色点矩阵。这些点和形成字符的黑白点大小相同。在高分辨率图象方式中共有6种颜色变化, 黑、白、红、兰、绿和紫色, 屏幕上任何一点都可以是黑白或是彩色。

由屏幕上显示的格式, 可知道目前它在哪一种方式中。因此当看到屏幕上有文字及数字时你可以确定APPLE II目前是在程序文本方式中。当看到屏幕上充满彩色方块, 那么APPLE II可能就是在低分辨率图象方式中, 在两种图象方式显示时, 可以有4行程序文本显示于屏幕的最下端, 这四行占了低分辨率图象的最下8行彩色方块, 余下为 $40 \times 40$ 矩阵方块, 在高分辨率图象方式显示时。这四行文字占了屏幕最下32行点, 余下为 $280 \times 160$ 点阵, 可以采用混合方式同时显示文字和图象, 但是不能同时显示两种图象。

#### §1.4.3 屏幕存贮器

视频显示利用系统RAM存贮器中的信息产生显示。每一个存贮单元的数值, 控制在屏幕上分别显示一个字符; 2个选在一起的彩色方块或一行上的7个点。在程序文本和低分辨率图象方式中, 有1024个单元的存贮空间用作屏幕显示的信息源。程序文本和低分辨率图象共用这一存贮区。在高分辨率图象方式中使用另一个较大的存贮区, 占8192个单元, 高分辨率图象显示较多的信息, 因此, 所需的存贮区较大。每一个存贮区通称为“页”。保存高分辨率图象的存贮区有时称作“图画缓冲区”因为它通常用来存贮一张图片或一个图形。

#### §1.4.4 屏幕页

对每种显示方式, 实际上都有二个存贮区, 作为其显示信息的来源。第一个存贮区, 称为“首页”, 第二个存贮区称为“次页”或“第二页”, 次页紧接在第一页之后, 其大小和第一页相同。次页对于要立即显示资料非常有用。程序还可使用两页存贮区, 来画卡通, 先显示一页, 再画另一页, 然后立即换页就可以达到卡通的效果。

程序文本和低分辨率图象公用一个存贮区首页和次页。混合方式也适用于第二页存贮区, 但两页无法重迭显示。

表4 显示的存贮区

| Screen page | Begins at:<br>Hcx | Decimal Ends at: |       |         |       |
|-------------|-------------------|------------------|-------|---------|-------|
| Text/Lo-Rse | primary           | \$ 400           | 1024  | \$ 7FF  | 2047  |
|             | Secondary         | \$ 800           | 2048  | \$ BFF  | 3071  |
| Hi-Res      | primary           | \$ 2000          | 8192  | \$ 3FFF | 16383 |
|             | Secondary         | \$ 4000          | 16384 | \$ 5FFF | 24575 |

#### §1.4.5 屏幕开关

用来决定显示的各种方式, 页次及是否为混合型的元件称作“软开关”。称他们是开关是因为它们只有两种不同的状态, 例如开或关, 文本或图象。至于它们称之为“软”开关是因为他们由计算机软件控制。

程序能通过访问某开关的特定单元“掷”开关, 与读、写的数据毫无关系, 只要访问到此单元地址就掷了开关。

有8个特殊的存贮单元来控制屏幕软开关的设定。它们是成对设置的, 即当访问一对存

贮单元之一，使和它对应的方式变成了“ON”，同一组的另一个方式自动变成“OFF”，这些软开关如下：

| 表 5 屏幕软开关        |         |         |                                     |  |
|------------------|---------|---------|-------------------------------------|--|
| Location:<br>Hcx | Decimal |         | Description:                        |  |
| \$ C050          | 49232   | - 16304 | Display a GRAPHICS mode             |  |
| \$ C051          | 49233   | - 16303 | Display TEXT mode                   |  |
| \$ C052          | 49234   | - 16302 | Display all TEXT or GRAPHICS        |  |
| \$ C053          | 49235   | - 16301 | Mix TEXT and a GRAPHICS mode        |  |
| \$ C054          | 49236   | - 16300 | Display the primary page (page 1)   |  |
| \$ C055          | 49237   | - 16299 | Display the secondary page (page 2) |  |
| \$ C056          | 49238   | - 16298 | Display LO-RES GRAPHICS mode *      |  |
| \$ C057          | 49239   | - 16297 | Display HI-RES GRAPHICS mode *      |  |

这些开关一共有十种不同的组合：

| 表 6 屏幕方式的组合           |                    |                    |  |                       |                    |                    |
|-----------------------|--------------------|--------------------|--|-----------------------|--------------------|--------------------|
| primary page          |                    |                    |  | Secondary page        |                    |                    |
| Screen                | Switches           |                    |  | Screen                | Switches           |                    |
| All Text              | \$ C054            | \$ C051            |  | All Text              | \$ C055            | \$ C051            |
| All Lo-Res Graphics   | \$ C054<br>\$ C052 | \$ C056<br>\$ C050 |  | All Lo-Res Graphics   | \$ C055<br>\$ C052 | \$ C056<br>\$ C050 |
| All Hi-Res Graphics   | \$ C054<br>\$ C052 | \$ C057<br>\$ C050 |  | All Hi-Res Graphics   | \$ C055<br>\$ C052 | \$ C057<br>\$ C050 |
| Mixed Text and Lo-Res | \$ C054<br>\$ C053 | \$ C056<br>\$ C050 |  | Mixed Text and Lo-Res | \$ C055<br>\$ C053 | \$ C056<br>\$ C050 |
| Mixed Text and Hi-Res | \$ C054<br>\$ C053 | \$ C057<br>\$ C050 |  | Mixed Text and Lo-Res | \$ C055<br>\$ C053 | \$ C057<br>\$ C050 |

要使APPLE进入上述的任何一种方式，程序只需访问设定该方式开关的存储单元地址。机器语言程序应使用十六进制地址，BASIC程序PEEK和POKE用十进制地址，设置开关的次序，可随意，但是进入图象方式后，最好将使用TEXT/GRAPHIC开关的次序列为最后，以免图象会消失掉。

#### § 1.4.6 程序文本方式

在程序文本方式中APPLE显示24行，每行40字符，屏幕上显示的每个字符，代表目前正在显示首页存储区内的一个存储单元的内容。字符集共64个字符，其中包括26个大写字母，10个数字和28个特殊字符。字符由5个点宽，7个点高的点矩阵所组成。每个字符的两边各有一个点宽的空白，以分开相邻的字符，每个字符上方有一个点宽的空白，以分开相邻的两行。这些字符通常以黑底白字显示，但也可以用白底黑字显示，或者显示闪烁字符。当显示是文本方式时，视频电路关掉忽隐忽现的彩色信号，以便获得清晰的黑白显示。

文本首页的存贮区是从1024存贮单元到2047单元，次页由2048单元开始到3071单元。在机器语言中首页存贮区的十六进制地址是由\$400到\$7FF；次页存贮区的十六进制地址是由\$800到\$BFF，每页有1024字节，在屏幕上仅显示860个字符，已足够用 每一页中剩下的64个单元，由外围接口的PROM里的程序作暂存单元用。

下图为能显示在屏幕上的64个可用的字符。

|   |    |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| @ | A  | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P |
| Q | R  | S | T | U | V | W | X | Y | Z | [ | / | ] | ^ | - | ! | " |
| # | \$ | % | & | , | [ | ] | * | , | + | - | . | / |   |   |   |   |
| 0 | 1  | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | : | ; | < | = | > | ? |   |

表7 ASCII屏幕字符

| 反相                      |   |   |    | 闪烁                  |   |   |    | 控制        |   |                 |    | 正常        |   |   |    | 小写 |  |
|-------------------------|---|---|----|---------------------|---|---|----|-----------|---|-----------------|----|-----------|---|---|----|----|--|
| Decimal                 |   |   |    |                     |   |   |    |           |   |                 |    |           |   |   |    |    |  |
| 0 16 32 48              |   |   |    | 64 80 96 112        |   |   |    | 128 144   |   | 160 176 192 208 |    | 224 240   |   |   |    |    |  |
| Hex \$00 \$10 \$20 \$30 |   |   |    | \$40 \$50 \$60 \$70 |   |   |    | \$80 \$90 |   | \$A0 \$B0 \$C0  |    | \$E0 \$F0 |   |   |    |    |  |
| 0 \$0                   | @ | P | 0  | @                   | P | 0 | @  | P         | 0 | @               | P  | 0         | @ | P | 0  |    |  |
| 1 \$1                   | A | Q | !  | 1                   | A | Q | !  | 1         | A | Q               | !  | 1         | A | Q | !  | 1  |  |
| 2 \$2                   | B | R | "  | 2                   | B | R | "  | 2         | B | R               | "  | 2         | B | R | "  | 2  |  |
| 3 \$3                   | C | S | #  | 3                   | C | S | #  | 3         | C | S               | #  | 3         | C | S | #  | 3  |  |
| 4 \$4                   | D | T | \$ | 4                   | D | T | \$ | 4         | D | T               | \$ | 4         | D | T | \$ | 4  |  |
| 5 \$5                   | E | U | %  | 5                   | E | U | %  | 5         | E | U               | %  | 5         | E | U | %  | 5  |  |
| 6 \$6                   | F | V | &  | 6                   | F | V | &  | 6         | F | V               | &  | 6         | F | V | &  | 6  |  |
| 7 \$7                   | G | W | ,  | 7                   | G | W | ,  | 7         | G | W               | ,  | 7         | G | W | ,  | 7  |  |
| 8 \$8                   | H | X | (  | 8                   | H | X | (  | 8         | H | X               | (  | 8         | H | X | (  | 8  |  |
| 9 \$9                   | I | Y | )  | 9                   | I | Y | )  | 9         | I | Y               | )  | 9         | I | Y | )  | 9  |  |
| 10 \$A                  | J | Z | *  | :                   | J | Z | *  | :         | J | Z               | *  | :         | J | Z | *  | :  |  |
| 11 \$B                  | K | [ | +  | ;                   | K | [ | +  | ;         | K | [               | +  | ;         | K | [ | +  | ;  |  |
| 12 \$C                  | L | / | ,  | <                   | L | / | ,  | <         | L | /               | ,  | <         | L | / | ,  | <  |  |
| 13 \$D                  | M | ] | -  | =                   | M | ] | -  | =         | M | ]               | -  | =         | M | ] | -  | =  |  |
| 14 \$E                  | N | ^ | .  | >                   | N | ^ | .  | >         | N | ^               | .  | >         | N | ^ | .  | >  |  |
| 15 \$F                  | O | - | /  | ?                   | O | - | /  | ?         | O | -               | /  | ?         | O | - | /  | ?  |  |

#### §1.4.7 低分辨率图象方式

低分辨率图象方式和文本方式一样，APPLE显示1024个单元的内容，但格式不同。在该方式中存贮器的每个字节不是显示ASCII字符，而是两个彩色方块，一个在另一个之上，所以屏幕可以显示宽40，高48的方块矩阵，每个方块的色彩变化各有16种。在黑白电视机上，这些彩色只会变成灰色和白色。

在低分辨率图象中，该页存贮区的每个字节在屏幕上代表了垂直相叠的两个方块，所以

每个字节分成相等的两部分，称作半字节。每个半字节的值可为 0 ~ 15。低半字节的值决定屏幕上相叠方块中上面方块的颜色，高半字节的值决定下面方块的颜色。全部颜色编成 0 ~ 15 的数字，如表 8。

表 7 分别给出了正常，反相和闪烁显示时 64 种字符的十进制和十六进制代码。

|       |    |  |        |
|-------|----|--|--------|
| \$ 00 | 0  |  | \$ 400 |
| \$ 01 | 1  |  | \$ 480 |
| \$ 02 | 2  |  | \$ 500 |
| \$ 03 | 3  |  | \$ 580 |
| \$ 04 | 4  |  | \$ 600 |
| \$ 05 | 5  |  | \$ 680 |
| \$ 06 | 6  |  | \$ 700 |
| \$ 07 | 7  |  | \$ 780 |
| \$ 08 | 8  |  | \$ 428 |
| \$ 09 | 9  |  | \$ 4A8 |
| \$ 0A | 10 |  | \$ 528 |
| \$ 0B | 11 |  | \$ 5A8 |
| \$ 0C | 12 |  | \$ 628 |
| \$ 0D | 13 |  | \$ 6A8 |
| \$ 0E | 14 |  | \$ 728 |
| \$ 0F | 15 |  | \$ 7A8 |
| \$ 10 | 16 |  | \$ 450 |
| \$ 11 | 17 |  | \$ 4D0 |
| \$ 12 | 18 |  | \$ 550 |
| \$ 13 | 19 |  | \$ 5D0 |
| \$ 14 | 20 |  | \$ 650 |
| \$ 15 | 21 |  | \$ 6D0 |
| \$ 16 | 22 |  | \$ 750 |
| \$ 17 | 23 |  | \$ 7D0 |
| \$ 18 | 24 |  |        |
| \$ 19 | 25 |  |        |
| \$ 1A | 26 |  |        |
| \$ 1B | 27 |  |        |
| \$ 1C | 28 |  |        |
| \$ 1D | 29 |  |        |
| \$ 1E | 30 |  |        |
| \$ 1F | 31 |  |        |
| \$ 20 | 32 |  |        |
| \$ 21 | 33 |  |        |
| \$ 22 | 34 |  |        |
| \$ 23 | 35 |  |        |
| \$ 24 | 36 |  |        |
| \$ 25 | 37 |  |        |
| \$ 26 | 38 |  |        |
| \$ 27 | 39 |  |        |

图 1 程序文本图表

图 1 是在文本方式中，显示存贮区的地址分配图表。每个在屏幕上显示的字符都标出了它存贮单元的地址。

表8 低分辨图象的颜色

| 十进制 | 十六进制 | 颜 色         | 十进制 | 十六进制 | 颜 色         |
|-----|------|-------------|-----|------|-------------|
| 0   | \$ 0 | Black       | 8   | \$ 8 | Brwon       |
| 1   | \$ 1 | Magenta     | 9   | \$ 9 | Orange      |
| 2   | \$ 2 | Dark Blue   | 10  | \$ A | Grey2       |
| 3   | \$ 3 | Purple      | 11  | \$ B | Pink        |
| 4   | \$ 4 | Dark Green  | 12  | \$ C | Light Green |
| 5   | \$ 5 | Grey 1      | 13  | \$ D | Yellow      |
| 6   | \$ 6 | Medium Blue | 14  | \$ E | Aquamarine  |
| 7   | \$ 7 | L gkt Blue  | 15  | \$ F | White       |

(颜色可能随电视的不同而改变,尤其是那些没有色调控制的电视。可利用主板右上方COLOR TRIM可变电容来调色彩。)

所以,一个字节,若其十六进制数值为\$ D8,那么屏幕上将显示一个棕色方块叠在一个黄色方块之上,若采用十进制运算,将此字符的十进制数值除以16,则所得的商为下面方块的颜色,所得的余数为下面方块的颜色。

图2是低分辨率图象方式中,存贮区和屏幕的对应图表在屏幕上的每个方块的位置都标以对应的存贮单元地址。

因为低分辨率图象显示和文本显示共有相同的一块存贮区。因此在文本方式和低分辨率图象方式之间进行转换,就会发生一些有趣的现象。例如,屏幕正在低分辨率图象方式,且屏幕TEXT/GRAPHICS屏幕开关转为文本方式,那么整个屏幕就会布满似乎随意排列的字上布满了彩色方块,现在将符,有时甚至还有反相和闪烁显示。同理,由文本方式进入低分辨率图象方式也有类似的结果。

#### §1.4.8 高分辨率图象方式

APPLE的第二种图象显示称作高分辨率图象。在高分辨率图象方式中它以280点宽,192点高的点阵,显示出53760个点。屏幕上能显示黑、白、紫、绿、红和蓝六种颜色,但个别点的色彩有些限制。

高分辨率图象方式从一个8192字节的存贮区取数据,该存贮区通常称作“图画缓冲区”这里有两个图画缓冲区,一个首页使用,另一个次页使用。这两个缓冲区是独立的且与文本方式及低分辨率图象方式所用的存贮区是分开的。高分辨率图象方式的页图画缓冲区在存贮单元8192~16383。次页图画缓冲区接在第一页之后从存贮单元16384到24575,十六进制地址计算,第一页在\$ 2000~3FFF,紧接着的次页在\$ 4000~\$ 5FFF。若APPLE安装16K或少于16K的存贮器,则不能使用次页,若存贮器容量小于16K则不能用高分辨率图象方式。

屏幕上的每一个点表示图画缓冲区内的一位,每个字节中的7位显示在屏幕上,余下的一位用来选择此字节其他位的色彩,每一行显示40个字节。一行中第一个字节的最低有效位(第1位)显示在屏幕的最左端,接着是第二位。然后是第三位等等。最高有效位(第8位)不显示。接着是下一个字节的第一位等等。一行总共有280点,整个屏幕共有192行。

在黑白电视机或显示器上,每个点所对应的位若是“ON”(或者等于1)则显示出白



它会显示紫色。若一点在最后列（279列）或者任一奇数列，则显示绿色。若两点并排在一起，则两个点都显示白色。若一个字节的非显示位为“ON”则紫色和绿色分别由蓝色和红色所取代。

因此，高分辨率图象方式中，可用六种颜色，遵循以下规定：

- （1）在偶数列的点必定是黑色，紫色或蓝色。
- （2）在奇数列的点必定是黑色，绿色或红色。
- （3）每个字节必定是紫/绿字节或蓝/红字节，而绝不可能在同一个字节中同时有绿、蓝、绿和红，或紫和红。
- （4）任何两个彩色点并排在一起，将全显示白色，即使此两点分别属于不同的字节。

## § 1.5 其他输入/输出

输入：盒带输入

三个一位的数字输入

四个模拟输入

输出：盒带输出

内装的扬声器

四个“电铃”式输入

选通脉冲输出

### § 1.5.1 扬声器

在APPLE机箱内的左侧，键盘下方有一8Ω的小型扬声器，它直接接到APPLE的线路内部，程序可使扬声器产生各种声响。此扬声器受一软开关控制，此开关能使扬声器的纸盒处于“入”和“出”两个位置。该软开关与控制各种显示方式的开关不同，它是一个联结开关。程序每访问一次与扬声器开关所对应的存贮器地址，扬声器就改变状态，由“入”到“出”或者相反的动作。每次改变状态，由“入”到“出”，或者相反的动作。每次改变状态，扬声器就会产生音调，通过频繁地，连续地访问扬声器开关的地址，程序可使扬声器产生规则的音调。

扬声器的软开关所对应的存贮单元为49200，访问该地址（或等效地址-16336或十六进制地址\$C030），都会使扬声器产生声响。

程序可以利用读或写来访问该扬声器开关的存贮单元，读写的数字无关紧要，只和该地址是否被利用有关。请注意，6502微处理器在执行写操作。之前，必定先执行读操作。因此，若用“写”来改变该软开关，它实际上就会开关两次。环式软开关，如扬声器开关的含义是，对其特殊的存贮单元进行写操作之后，它读然由在此动作之前的状态。

### § 1.5.2 盒带接口

在主板后方，VIDEO插座的右侧有两个录音插座，分别标明输入(IN)和输出(OUT)，可以用两端有录音插头的电缆，一端插在这里，另一端接盒式磁带机，这样，就可把信息存到音频盒带上，或从音频盒带读回信息。

标有“OUT”的插座还联到主板的另一个软开关上。这个开关与扬声器开关相同，也是一个环式开关，该软开关所对应的特殊存贮单元是49184（或等效-16352或十六进制的\$C020）。访问该单元会使输出插座上的电压从0伏变到-25毫伏，或由25毫伏回到0毫



伏。电缆的另一端插入盒式磁带机的 MICRO PHONE 输入端。若磁带机在记录, 将记录一个声音, 重复访问开关的存贮单元, 就会记录一段声调, 调整声调的高低和长短后计算机的信息可记录在磁带上, 保存备以后使用, 在系统监控程序中有把数据记录到磁带上的程序。

预先警告: 你若通过写特殊存贮单元来触发盒带输出软开关, 那么实际上会在磁带上记录两个信息, 道理与扬声器开关相同。因此, 只能用读操作来触发盒带输出软开关。

另一个标有“IN”的插座用来由盒式磁带机输出信息, 主要的目的在于把磁带上的记录, 译成数据并存于存贮器, 因此, 存于磁带上的程序和数据可读回, 并重复使用。

输入电路从盒式磁带机的 EARPHONE 插座获得峰对峰为 1 伏的信号, 然后将信号转换成一串 0 和 1。每次输入信号由正跳到负时, 输入电路立即改变状态, 例如它正在送出 1, 则立即改为送出 0, 反之亦然。程序由查存贮单元 49248 (或 -16288 或十六进制的 \$C060) 可知道输入电路的状态。若从该存贮单元读出的值大于或等于 218, 则盒带输入电路的状态为“1”; 若使存贮单元的值小于 128, 则状态为“0”。

尽管 BASIC 程序也可以读到盒带输入电路的状态, 但是由于其程序速度太慢, 因此在读到输入电路状态时, 已经失去意义了。系统监控程序中有一段读磁带和译码的程序。

### § 1.5.3 游戏 I/O 插座

设置游戏 I/O 插座的目的是为了连接特殊的输入输出装置以提高程序效率, 尤其是游戏程序。该插座可连接:

- (1) 三个一位输入
- (2) 四个一位输出
- (3) 一个数据选通
- (4) 四个模拟输入

以上全部输入, 输出都可由程序控制。通常 APPLE 配一对游戏操纵器, 经过电缆, 插入游戏 I/O 插座。每个操纵器上有两个转盘分别连到模拟输入, 另有两个按钮, 则连接插头的一位输入。

### § 1.5.4 电铃输出

四个一位输出称作电铃。每个电铃输出可作其他电子设备的输入, 或者和某些电路相连以驱动灯泡, 继电器和扬声器等等。

每个电铃都由一个软开关控制。这些软开关的存贮器地址共有四对。每对控制一个电铃。若访问某一对的第一个地址, 则对应的电铃输出转换到“OFF”, 若访问这一对的第二个地址则电铃输出转为“ON”。当一个电铃输出是“OFF”, 它在游戏 I/O 连接插座的相应引脚上的电压, 将趋近于 0 伏。当一个电铃是“ON”, 此引脚上的电压趋近 5 伏。电铃软开关如下表。

| 表 9 电铃的特殊单元 |     |       |        |         |
|-------------|-----|-------|--------|---------|
| 电 铃         | 状 态 | 地 址   |        |         |
|             |     | 十 进 制 |        | 十六进制    |
| 0           | off | 49240 | -16296 | \$ C058 |
|             | on  | 49241 | -16295 | \$ C059 |
| 1           | off | 49242 | -16294 | \$ C05A |
|             | on  | 49243 | -16293 | \$ C05  |
| 2           | off | 49244 | -16292 | \$ C05C |
|             | on  | 49245 | -16291 | \$ C05D |
| 3           | off | 49246 | -16290 | \$ C05E |
|             | on  | 49247 | -16289 | \$ C05F |