

微型计算机原理及其应用  
实验指导书

南京航空学院

## 前 言

为了配合由南京市科技局、南京电子光导领导小组、南京电视台、南京航空学院举办的电视教学，由南航计算机科学与工程系组织编写了这本微机实验指导书。

本指导书是结合电视教学所用教材的要求，并参照兄弟院校的指导书编写的。书中的内容从基本实验开始，由浅入深地开设 8 个实验，包括熟悉单板机的使用，程序调试，程序中断，输入输出接口以及数/模、模/数转换实验。全部实验都经过实际调试。读者可根据讲课内容选做其中的部分实验。

参加编写的有：王荣照(单板机功能简介和键盘操作说明，实验 1, 2, 3)，周德安(实验 4, 5)，郝立新、张引祥(实验 6, 7, 8, 附录)。由于水平有限，编写时间仓促，缺点和错误之处恳请读者批评、指正。

编 者

1984.5.

# 目 录

## TP-801 Z80 单板机功能简介和键盘操作说明

|     |                                   |        |
|-----|-----------------------------------|--------|
| § 1 | TP-801 Z80 单板机功能简介 .....          | ( 1 )  |
| § 2 | TP-801 Z80 单板机的主要技术特性 .....       | ( 1 )  |
| § 3 | 操作步骤及注意事项 .....                   | ( 2 )  |
| § 4 | TP-801 Z80 单板机键盘操作说明 .....        | ( 3 )  |
| § 5 | TP-801 Z80 单板机的存贮空间分配 .....       | ( 8 )  |
| § 6 | TP-801 Z80 单板机 I/O 接口地址空间分配 ..... | ( 9 )  |
| 实验一 | TP-801 单板机操作及使用 .....             | ( 9 )  |
| 实验二 | 程序调试 .....                        | ( 13 ) |
| 实验三 | 自编程序调试 .....                      | ( 16 ) |
| 实验四 | Z80-CPU 可屏蔽中断实验 .....             | ( 17 ) |
| 实验五 | Z80-CTC 应用实验 .....                | ( 23 ) |
| 实验六 | Z80-PIO 接口实验 .....                | ( 29 ) |
| 实验七 | 数—模(D/A)转换实验 .....                | ( 33 ) |
| 实验八 | 模—数(A/D)转换实验 .....                | ( 38 ) |
| 附 录 | 集成电路参考资料 .....                    | ( 42 ) |

# TP-801 Z80 单板机功能简介和键盘操作

## § 1 TP-801 Z80 单板机功能简介

TP-801 单板机是以 Z80 系统器件构成的最小系统。它具有简单的开发功能：如用单步操作执行存放在 ROM 或 EPROM 中的程序；可在用户程序中设置五个断点；可对 2716/2758 EPROM 进行编程等。

本机使用八片静态读写存储器 2114 构成容量为 4K 字节的随机存储器供用户使用，其地址空间为  $2000_H - 2FFF_H$ 。同时配有 2K 字节的 ROM 存放监控程序 TPBUG-A，其地址空间为  $0000_H - 07FF_H$ ，实现键盘的各命令键功能。

本机配有音频盒式磁带机接口，用转录线将盒式磁带录音机与单板机的插孔相连，可方便地把盒式录音机作为本机的外存储器。单板机右上方的红色发光二极管可指示磁带上的信息是否输入到存储器中。利用这只二极管的指示，可在一盘磁带上录制若干个程序。

本机配有六位七段数码显示器，通常左边的四位显示地址，右边的两位显示数据。（对于 IX, IY, PC 和 SP 等 16 位寄存器而言，则用右边四位显示数据）。

本机配有 16 个十六进制的数字键和 12 个命令键，命令键是可控制的双功能键，共有  $2^0$  个命令。详细内容在下面介绍。

本机还配有三个按钮和开关：按钮  $S_1$  用来对整机提供 RESET(复位)信号。开关  $S_2$  指向 MON RST 位置，则在复位后，显示器上出现提示符“P”，并扫描键盘的输入；如指向 PROM<sub>1</sub> RST 位置，则在复位后进入 PROM<sub>1</sub> 中的用户程序。开关  $S_3$  用来选择 PROM<sub>2</sub> 中的 EPROM 处于“写入(PGM)”或“读出(READ)”的状态。

## § 2 TP-801 Z80 单板机的主要技术特性

1. 中央处理器为 Z80-CPU。
2. 时钟( $\phi$ )频率为 1.9968MHZ，便于使用 8080A 的接口电路。晶体振荡器的频率为 3.9936MHZ。
3. RAM 为 4K 字节的 2114 静态读写存储器。
4. ROM 为 2K 字节，编入 TPBUG-A 监控程序。
5. 两个 PROM 插座，可插入 4K 字节的 EPROM。
6. Z80-PIO 并行 I/O 接口芯片一片，它有两个 8 位可编程的 I/O 口(A 口和 B 口)供用户使用。
7. Z80-CTC 计数器/定时器芯片一片，它有四个通道：通道 0 供用户使用，其余由 TPBUG-A 使用。
8. 键盘共有 28 个按键，其中：

16个为十六进制数字键：0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F。

12个为命令键，当使用 TPBUG-A 时，12个命令键具有二十种功能：

|           |                  |
|-----------|------------------|
| MON'      | 换上档的监控键          |
| MON       | 换下档的监控键          |
| STEP      | 单步执行程序键          |
| DISP/MEM  | 偏移量计算/存贮单元检查键    |
| REG'/REG  | 辅助寄存器检查/寄存器检查键   |
| DUMP/LAST | 信息转贮磁带/查上一个存贮单元键 |
| LOAD/NEXT | 磁带输入/下一条键        |
| 2FB8/PROM | 用户定义功能/EPROM 写入键 |
| 2FBA/MOVE | 用户定义功能/存贮块移动键    |
| 2FBC/PORT | 用户定义功能/端口检查键     |
| 2FBF/BP   | 用户定义功能/设置断点键     |

9. 显示器共有六位 LED 数字显示，通常左四位显示地址，右两位显示数据，若显示寄存器时，右边四位显示数据。

10. 配有音频盒式磁带机接口。

11. 布线区为  $2.5 \times 7$  吋。

12. S-100 总线插孔两组。

13. 电源为  $+5V \pm 5\%$ ，1A。若要对 EPROM 进行写入，尚须接  $+25V \pm 1V$ ，30mA 的电源。

### § 3 操作步骤及注意事项

1. 请按以下规定连线：

① 直流稳压电源、盒式录音机的电源线均接至交流 220V。

② 用转录线将单板机上的 AUX 插孔与录音机上的 MIC 插孔相连。

③ 用转录线将单板机上的 EAR 插孔与录音机上的 MONITOR OUT 或 EAR PHONE 插孔相连。

④ 将单板机上的电源线接到直流稳压电源 +5V 输出端。

2. 设定单板机上开关位置：

①  $S_2$  开关设在 MON RST 位置。

②  $S_3$  开关设在 READ 位置。

3. 开机：

① 接通直流稳压电源上的 +5V 开关。

② 按 RESET 按钮  $S_1$ ，单板机进入 TPBUG-A，显示器上显示提示符“P”，表明机器可以接收键盘来的命令。

4. 输入用户程序，并进行相应的调试和操作，

5. 注意事项

① 在进行第三步时，如不出现“P”，应立即关掉电源，仔细检查导线连接是否出错，



器件与插座的接触是否可靠，各开关位置设置是否正确，直流稳压电源的输出电压是否满足  $+5V \pm 5\%$ 。

② 切忌用手直接触摸大规模集成电路的引脚。

③ 切忌用电源电压为 220V 的电烙铁对单板机进行焊接，以防损坏芯片。必须使用电烙铁时，应拔掉电烙铁的电源插头，利用余热进行焊接。

④ 不要把导线头和小起子等金属物丢在单板机上，以防短路损坏片子。

## § 4 TP-801 Z80 单板机键盘说明

本机共设有 12 个双功能命令键和 16 个数据键，一个复位按钮，两个单刀双向开关，下面分别说明。

### 1. RESET (复位) 按钮

当单板机接通电源后，如  $S_3$  处于 READ 位置， $S_2$  处于 MON RST 位置，复位按钮可使 CPU 处于初始化状态：中断允许触发器处于禁止状态；

置寄存器 I, R 的内容为  $00H$ ；

置程序计数器 PC 的内容为  $0000H$ ；

置中断方式为  $IM0$  (方式 0)。

在复位信号有效期间，CPU 从  $0000H$  地址开始执行监控程序的初始化部分，在显示器的最左端显示 TPBUG-A 的提示符“P”，等待键盘输入。如果在 RESET 前，在用户程序中使用“BP”键设置了断点，如按复位按钮，所设置的断点被消除。

### 2. 16 个十六进制数字键

这些键用来向单板机输入十六进制数字，这些数字可以是存贮单元的地址、I/O 口地址、寄存器标号、指令码，也可以是一些数据。

### 3. MON (Monitor 监控) 键

MON 键的用途是暂停执行程序或退出当前的命令，返回监控程序，准备接收新的输入命令。它与 RESET 按钮不同的是：按 RESET 按钮将破坏前面已执行过的程序的现场，而用 MON 键，一方面暂停执行程序，同时保存前面已执行过的程序的现场，以便检查程序执行的结果。

### 4. MEM (Memory Examine 存贮器单元检查) 键

MEM 键用来检查和改变存贮单元的内容，也就是通过该命令键对 RAM 单元进行读、写操作。同时可对 ROM、EPROM 进行读操作。

当显示器最左端出现提示符“P”后，先用数字键从高位到低位依次输入表示地址的四位十六进制数字，并显示在显示器的地址显示部分(左边四位)，然后再按 MEM 键，相应存贮单元的内容就显示在右边两位数据显示部分。如要修改其单元中的内容，只要重新输入两个新的数据即可。若想退出存贮器检查工作方式，只要按 MON 键即可。

### 5. PORT (Port Examine 端口检查) 键

PORT 键的功能与 MEM 键的作用相似，它是用来检查和修改端口地址中的内容。使用 PORT 键前，要先输入待检查的端口地址(两个十六进制数)，然后再按 PORT 键，相应端口地址内的数据便显示在数据显示段(右两位)。若要修改端口地址中的内容，只要重

新输入两个新数即可。退出端口检查的方法是按 MON 键。

#### 6. REG (Register Examine 寄存器检查) 键

REG 键用来检查和修改下列寄存器的内容: A、B、C、D、E、F、H、L、I、IFF、PC、IX 和 IY。对于堆栈指示器 SP 的内容只能检查, 不能修改。使用此键的方法是先输入代表寄存器号的数据键, 然后再按 REG 键, 相对应的寄存器的内容就显示在显示器的右边两位或四位上。若需要修改, 只要重新输入两个或四个新数据即可。按 MON 键, 便退出 REG 工作方式。

#### 7. REG' (Alternative Register Examine 辅助寄存器检查) 键

REG' 键用来检查或更改辅助寄存器 A'、B'、C'、D'、E'、F'、H' 和 L' 中的内容。其操作方法同 REG 键。

#### 8. BP (Break Point 断点设置) 键

设置断点是调试程序、查找程序中的错误的重要诊断手段。所谓断点就是当程序执行到此处时自动停下来, 便于你用 MEM 键或 REG 键或 PORT 键来检查存贮单元、寄存器和端口中的内容和状态, 判断程序执行是否正确。

TP-801 单板机最多能设置 5 个断点。设置方法是: 在见到提示符“P”的情况下, 在你认为有必要设置断点的地方, 把操作码的地址 (四位十六进制数) 输入进去并显示在显示器的左边四位上, 然后按 BP 键, 显示器仍显示四位地址, 便完或了一个断点的设置。如还需要继续设置断点, 先按 MON 键, 显示器显示“P”后, 再打入新的断点操作码地址, 再按 BP 键。如此重复, 最多可设置 5 个断点。如设置的断点已超过 5 个, 按 BP 键时, 显示器显示提示符“P”, 告诉你前面设置的 5 个断点已全部完整地保留下来, 现已超过 5 个断点。

如果要消除断点, 有三种方法可供选择:

- ① 按 RESET 按钮。
- ② 当显示器显示提示符“P”时, 不给断点地址就按 BP 键。
- ③ 当显示器显示提示符“P”时, 按 STEP (单步) 键。

通常 MON 键对断点设置没有任何影响, 但下列情况例外。如程序遇到 HALT 指令而停机, 此时按 MON 键, 控制 CPU 返回监控程序, 但用户程序中的断点指令 (RST8 指令) 并没有被取代, 而仍留在用户程序中。在调试程序时, 用户应注意这种情况的发生。

#### 9. STEP (单步执行程序) 键

STEP 键是调试程序的另一个重要手段, 给用户提供了只执行一条指令的工作方式。当一条指令执行完后, 显示 PC (即下一条指令的地址) 和累加器 A 的内容, 此时你可使用 MON 键, MEM 键或 REG 键来检查该指令执行是否正确。当检查或修改完毕, 再按 STEP 键, 单板机又可一条一条地执行指令。对于条件转移和条件转子程序指令, 单步操作是很有用的, 因为可通过观察 PC 显示的内容判断是否进行转移。在一步一步地执行程序中, 你还可以用 REG 键修改 PC 的内容, 迫使程序转移到一个新的地址。

STEP 键的使用方法:

① 当在程序中设置一个断点, 执行程序并在该断点处停止程序的继续执行时, 如果你按 STEP 键, 每按一次, 单板机就从断点处向下执行一条指令。

② 把要单步执行的第一条指令的地址预置在程序计数器 PC 中, 然后按一次 STEP 键, 单板机就执行 PC 所指的那一条指令。

## 10. EXEC (Execute 连续执行程序) 键

EXEC 键用来连续执行 RAM、ROM 和 EPROM 中的程序。它有两种使用方法:

① 先输入要执行程序的起始地址的四位数字, 然后按 EXEC 键, CPU 即从该地址开始执行程序。此方法用于从程序的首地址开始执行程序。

② 仅仅按下 EXEC 键。此时从保存在“用户寄存器存放区”中的 PC 的现行地址执行程序。有两种情况: a. 现在的 PC 值是执行了一段程序后的值。b. 现在的 PC 值是用寄存器检查方式送到 PC 中的值。使用这种方式时只要按一下 EXEC 键, 就以当前的 PC 指示的地址为起点开始执行程序。此方式对于在遇到一个断点后, 或在使用单步方式后, 重新开始执行程序是十分有用的。

## 11. DUMP (Cassette Dump 信息转贮) 键

当你需要把 RAM 中的程序用盒式录音机存贮起来, 以便于下次继续使用或调试时, 可使用 DUMP 命令键。在转贮过程中, 传送信息的速率为 300 波特, 即以八个 2400 赫芝的音频脉冲信息表示“1”, 以四个 1200 赫芝的音频脉冲信号表示“0”。在一段程序转贮中, 开头有 40 秒全“1”的引导信号, 末尾有 5 秒全“1”的结尾信号。

转贮的操作过程:

① 用转录线将本机的  $\rightarrow 2$ (AUX)端与录音机的“AUXILIARY”或“MIC”输入端相联, 并把磁带装入录音机。

② 用 MEM 键将 RAM 中需要转贮的信息的起始地址装入  $2FC0_H$  和  $2FC1_H$  单元 (高字节送  $2FC0_H$ 、低字节送  $2FC1_H$ )。

③ 用 MEM 键将 RAM 中需要转贮的信息的末地址高字节送  $2FC2_H$ , 低字节送  $2FC3_H$ 。

④ 按 MON' 键 (因为 DUMP 键属上档键), 使显示器出现提示符“'”, 将录音机置成录音方式, 按 DUMP 键, “'”将消失。当提示符“'”再次出现时, 表示所有的程序已转贮到磁带上。关掉录音机, 取出磁带即可。

## 12. LOAD (Cassette Load 磁带装入) 键

LOAD 键用来把磁带上的程序或信息装入到单板机的 RAM 中去, 其操作方法如下:

① 用转录线把录音机的“MONITOR OUT”或“EAR PHONE”插孔与单板机的  $J_1$ (EAR)插孔相接。

② 把磁带走到程序相应的位置。将录音机的高音控制调到最大, 低音控制调到最小, 音量控制调到最小。

③ 按 MON' 键, 使显示器显示提示符“'”, 以放音方式启动录音机。按 LOAD 键, 提示符消失。调整音量旋钮, 逐渐增大, 直至 LOAD 发光二极管亮, 然后将音量再增大 20%。在整个输入过程中, 发光二极管应一直保持发光。

④ 如果装入成功, 单板机将显示提示符“'”, 录音机可关掉。如果出现错误, 显示器上显示下一地址块的地址。若出现这种现象, 可重新调整音量和音调旋钮, 倒带再输入一次。

## 13. PROM (EPROM Programmer EPROM 写入) 键

TP-801 单板机可以将 RAM 中首地址为  $2000_H$  的程序写入到首址为  $1000_H$ 、插在 PROM<sub>2</sub> 插座中的 2716/2758 型 EPROM 中去。写入时, 还需有一个  $+25V \pm 1V$ 、30ma 的辅助电



对 EPROM 进行写入(编程)的操作方法是:

② 用 MEM 键或 LOAD 键将要写入 EPROM 的程序输入到从 2000<sub>H</sub> 地址开始的 RAM 中去。

③ 按 MON 键, 出现提示符“P”, 输入四位十六进制数(高位先输入)表示从 RAM 中写入到 EPROM 的字节数。

④ 将开关 S<sub>2</sub> 扳向 PGM 位置, 按 PROM 键, 显示器熄灭, 开始进行写入。当重新显示提示符“P”时, 表示 EPROM 写入完毕, 内容正确无误。

⑤ 若不是重新显示提示符，而是显示四位地址和两位数据，表示 EPROM 中的内容与 RAM 中内容不相同，写入出错。按 NEXT 键，监控程序将继续比较 RAM 和 EPROM 中的内容，并可能再次出现错误。其原因有两种：

- 原 EPROM 中的内容没有擦除。可重新擦除一次, 再进行编程。
- 原 EPROM 本身有缺陷。

⑥ 编程完成后, 立即将  $S_2$  扳向 READ 位置。

#### 14. NEXT (下一条) 键

NEXT 键常与 MEM 键、PORT 键和检查 EPROM 写入错误一起使用。

在存储器或端口检查时，NEXT 键用来选择下一个存储单元或端口单元，并自动地显示下一地址及其单元中的内容，起到对存储器或端口地址加“1”的作用。

在 EPROM 写入时, 如出现错误, NEXT 键可以步进检查各存储单元的内容。

TP-801 单板机配上改进后的 监控程序 TPBUG-A, 丰富了键盘的功能。即与 TPBUG 一样, 仍为 2K 字节, 但使八个命令键成为双功能键。除了其中一个作为双功能键的上档和下档选择外, TPBUG-A 相当于增加了七个新的命令。

TP-801 配了 TPBUG-A 后的 键盘如图 1—1 所示。

图中带粗黑框的按键功能与原TPBUG完全相同。除此以外的按键中NEXT键、MEM(存储器检查)键、REG(寄存器检查)键、REG'(辅助寄存器检查)键、PORT(口检查)键、PROM(EPROM写入)键、DUMP(信息转贮磁带)键、LOAD(磁带输入)键、BP(设置断点)键等与TPBUG的控制功能相同,仅在位置上有的是上档键、有的是下档键。另外的八个命令键,完全是新设置的,它们是:

|                     |                     |                     |                     |
|---------------------|---------------------|---------------------|---------------------|
| $\frac{2F18}{PROM}$ | $\frac{2FBA}{MOVE}$ | $\frac{2FBC}{PORT}$ | $\frac{2FBE}{BP}$   |
| $\frac{DISP}{MEM}$  | $\frac{REG}{REG}$   | $\frac{DUMP}{LAST}$ | $\frac{LOAD}{NEXT}$ |
| 7<br>H              | 8<br>L              | 9                   | A<br>MON            |
| 4<br>IX             | 5<br>IY             | 6<br>I              | B<br>MON            |
| 1<br>PC             | 2<br>SP             | 3<br>IFF            | C<br>STEP           |
| 0                   | F                   | E                   | D<br>EXEC           |

图1—1 键盘命令图

### 1. MON' (交换) 键

MON' 键具有 MON 键使系统进入监控程序等待键盘命令的基本功能, 同时还能完成双功能键的上下档更换。如按 MON' 键, 提示符为撇号 "'", 则上档命令键有效。如按 MON 键, 提示符为 "P", 则下档命令键有效。以 REG'/REG 键为例, REG' 为上档命令键, REG 为下档命令键。当按 MON' 后再按 REG'/REG 键, TPBUG-A 接受上档命令, 进行辅助寄存器检查。如按 MON 键再按 REG'/REG 键, TPBUG-A 接受下档命令, 进行主寄存器检查。

### 2. LAST (查上一个存贮单元) 键

LAST 键是增加的一个常用命令键, 用以访问显示器现行存贮单元的上一个存贮单元。这一点与 NEXT 键正好相反。但 NEXT 键不仅可以访问 (读或写) 下一个存贮器单元, 而且还可以访问下一输入/输出口以及 EPROM 编程时的下一个错误检查。而 LAST 键只能检查上一个存贮器单元。另外 LAST 键和 NEXT 键可以交叉使用, 灵活地更换所要访问的存贮单元。

### 3. MOVE (存贮块移动) 键

只要六位显示器左边四位显示一个地址, 不论右边两位是否显示该单元内容, 按一下 MOVE 键, 则从所显示的地址开始, 直到  $2\text{EFF}_H$  单元为止, 所有存贮单元内容均向下移动一个字节, 原显示地址单元的内容被清 0。  $2\text{F00}_H$ — $2\text{FFF}_H$  单元中的内容不会向下移动, 其中  $2\text{F00}_H$ — $2\text{F87}_H$  共 136 个字节可作为用户存放数据表格和变量。此键在用户程序调试中, 为修改程序提供了方便。

### 4. DISP (相对转移偏移量计算) 键

DISP 键的功能与 TPBUG 中起始地址  $00\text{C0}_H$  的程序的功能是一样的, 即进行相对转移指令操作数—偏移量的计算; 显示计算结果; 提示计算的有效性 (是否超过单字节范围); 计算结果写入指令的操作数字节。

例: 某 JR 指令的地址为  $2300_H$ , 转移目的地址为  $22\text{F0}_H$ , 计算相对偏移量。

操作方法:

- ① 把源指令地址  $2300_H$  送 IY 变址寄存器。
- ② 把转移目的地址  $22\text{F0}_H$  送 IX 变址寄存器。
- ③ 按 MON' 键, 再按 DISP/MEM 键。

显示器显示

|   |   |   |   |
|---|---|---|---|
| F | F | E | E |
|---|---|---|---|

FF—指示负跳转 (如 00 则指示正跳转)

EE—计算的相对偏移量结果, 并已经写入到指令操作数字节中去。

### 5. $2\text{FB8}$ 、 $2\text{FBA}$ 、 $2\text{FBC}$ 、 $2\text{FBE}$ (用户程序启动) 键四个

此四个键留给用户定义, 用于启动四个用户程序。用户程序地址的低字节在前, 高字节在后。四个地址值由用户指定, 四个键不必按次序使用。地址表的内容, 可用键盘输入。

例: 用户有一个程序在首址为  $2800_H$  的存贮器中, 只要把地址  $2800_H$  存放在  $2\text{FB8}$ 、 $2\text{FB9}$  单元中:  $2\text{FB8}: 00$

$2\text{FB9}: 28$

然后按 MON' 键, 再按  $2\text{FB8}/\text{PROM}$  键, 单板机即可转入  $2800_H$  地址执行用户程序, 因而

方便了用户。

如果用户程序已经固化在 EPROM 中，也就是说用户程序的首地址已经固定，为方便操作，建议用户使用如下简单的引导程序，由程序代替人工输入用户程序地址。如

- 用户程序 1 首地址 1200<sub>H</sub>
- 用户程序 2 首地址 1500<sub>H</sub>
- 用户程序 3 首地址 1700<sub>H</sub>
- 用户程序 4 首地址 17F0<sub>H</sub>

引导程序

```
2000H LD HL, 1200H
      LD (2FB8H), HL
      LD HL, 1500H
      LD (2FBAH), HL
      LD HL, 1700H
      LD (2FBCH), HL
      LD HL, 17F0H
      LD (2FBEH), HL
      HALT
```

这样只要你按 2FB8/PROM 键，即转用户程序 1。

按 2FBA/MOVE 键，即转用户程序 2。

按 2FBC/PORT 键，即转用户程序 3。

按 2FBE/BP 键，即转用户程序 4。

## § 5 TP-801 Z80 单板机的存贮空间分配

### 1. 存贮空间分配

|                                      |       |                                              |
|--------------------------------------|-------|----------------------------------------------|
| 0000 <sub>H</sub> —07FF <sub>H</sub> | 2K 字节 | TPBUG-A 监控程序。                                |
| 0800 <sub>H</sub> —0FFF <sub>H</sub> | 2K 字节 | PROM <sub>1</sub> ，可存放用户应用程序。                |
| 1000 <sub>H</sub> —17FF <sub>H</sub> | 2K 字节 | PROM <sub>2</sub> ，可存放用户应用程序，还可对 EPROM 进行写入。 |
| 1800 <sub>H</sub> —1FFF <sub>H</sub> | 2K 字节 | 未被使用                                         |
| 2000 <sub>H</sub> —27FF <sub>H</sub> | 2K 字节 | RAM                                          |
| 2800—2FFF <sub>H</sub>               | 2K 字节 | RAM                                          |

### 2. RAM 的存贮分配

|                                      |        |          |
|--------------------------------------|--------|----------|
| 2000 <sub>H</sub> —23FF <sub>H</sub> | 1K 字节  | 用户程序工作区  |
| 2400 <sub>H</sub> —27FF <sub>H</sub> | 1K 字节  | 用户程序工作区  |
| 2800 <sub>H</sub> —2BFF <sub>H</sub> | 1K 字节  | 用户程序工作区  |
| 2C00 <sub>H</sub> —2F87 <sub>H</sub> | 904 字节 | 用户程序工作区  |
| 2F88 <sub>H</sub> —2F9F <sub>H</sub> | 24 字节  | 监控程序工作区  |
| 2FA0 <sub>H</sub> —2FB7 <sub>H</sub> | 24 字节  | 用户程序栈工作区 |

2FB8<sub>H</sub>—2FBF<sub>H</sub> 8字节      四个用户程序入口地址  
2FC0<sub>H</sub>—2FFF<sub>H</sub> 64字节      RAM 暂存区和断点表

## § 6 TP-801 单板机 I/O 接口地址空间分配

|                    |                |
|--------------------|----------------|
| 80 <sub>H</sub>    | PI0 A口数据寄存器口地址 |
| 81 <sub>H</sub>    | PI0 B口数据寄存器口地址 |
| 82 <sub>H</sub>    | PI0 A口控制寄存器口地址 |
| 83 <sub>H</sub>    | PI0 B口控制寄存器口地址 |
| 84 <sub>H</sub>    | CTC 通道 0 口地址   |
| 85 <sub>H</sub>    | CTC 通道 1 口地址   |
| 86 <sub>H</sub>    | CTC 通道 2 口地址   |
| 87 <sub>H</sub>    | CTC 通道 3 口地址   |
| 88-8B <sub>H</sub> | 显示器段选择控制口地址    |
| 8C-8F <sub>H</sub> | 显示器位选择控制口地址    |
| 90-93 <sub>H</sub> | 键盘读键值控制口地址     |
| 94-97 <sub>H</sub> | 未使用            |
| 98-9B <sub>H</sub> | 未使用            |
| 9C-9F <sub>H</sub> | 未使用            |

## 实验一 TP-801单板机操作及使用

### 一、实验目的

1. 在阅读上面介绍的 TP-801 单板机功能简介和键盘操作说明的基础上, 掌握输入程序、执行程序 and 检查寄存器等单板机的基本使用方法。
2. 比较熟练地用单步操作和设断点的方法来执行程序、调试程序。

### 二、实验设备

TP-801 Z80 单板机一台  
5 V 2 A 直流稳压电源一台

### 三、实验内容及步骤

通过一个简单的加法运算程序来熟悉 TP-801 单板机的基本使用方法。

#### 1. 实验程序

|                   |       |      |                      |                              |
|-------------------|-------|------|----------------------|------------------------------|
| 2000 <sub>H</sub> | 3E 15 | LD   | A, 15 <sub>H</sub> ; | 把被加数 15 <sub>H</sub> 送 A 累加器 |
| 2002 <sub>H</sub> | 06 12 | LD   | B, 12 <sub>H</sub> ; | 把加数 12 <sub>H</sub> 送 B 寄存器  |
| 2004 <sub>H</sub> | 80    | ADD  |                      | ; 作加法, 结果保存在 A 累加器           |
| 2005 <sub>H</sub> | 76    | HALT |                      | ; 停机结束                       |

## 2. 输入实验程序

① 接通稳压电源开关, 调整输出电压为 $+5V \pm 5\%$ , 关掉开关, 连接到单板机 $+5V$ 接线端。请注意正负极性。

② 接通电源开关, 按 RESET( $S_1$ )按钮, 显示器应显示提示符“P”。

③ 输入程序的首地址。按数字键 2000, 地址显示段(左四位)应显示  $2000_H$ 。

④ 按 MEM(存储器检查)键, 右边二位数据显示段应显示  $2000_H$  单元中的任意值  $\times \times$ 。

⑤ 输入程序。

按数字键 3E, 把指令码  $3E_H$  写入  $2000_H$  存贮单元, 并显示在数据显示段。

按 NEXT 键。地址加 1, 形成下一个连续地址。地址显示段显示 2001, 数据段显示  $2001_H$  单元任意值  $\times \times$ 。

按数据键 15, 被加数  $15_H$  被写入  $2001_H$  存贮单元, 数据显示段显示  $15_H$ 。

按 NEXT 键。地址加 1, 地址显示段显示  $2002_H$ , 数据显示段显示  $2002_H$  单元任意值  $\times \times$ 。

按数字键 06, 指令码  $06_H$  被写入  $2002_H$  存贮单元, 数据显示段显示  $06_H$ 。

按 NEXT 键。地址加 1, 地址显示段显示  $2003_H$ , 数据显示段显示  $2003_H$  单元任意值  $\times \times$ 。

按数字键 12, 加数  $12_H$  被写入  $2003_H$  存贮单元, 数据显示段显示  $12_H$ 。

按 NEXT 键, 地址加 1, 地址显示段显示  $2004_H$ , 数据显示段显示  $2004_H$  存贮单元中的任意值  $\times \times$ 。

按数字键 80, 指令码  $80_H$  被写入  $2004_H$  存贮单元, 数据显示段显示  $80_H$ 。

按 NEXT 键。地址加 1, 地址显示段显示  $2005_H$ , 数据显示段显示  $2005_H$  存贮单元的任意值  $\times \times$ 。

按数字键 76, 指令码  $76_H$  被写入  $2005_H$  存贮单元, 数据显示段显示  $76_H$ , 程序到此输入结束。

⑥ 按 MON 键, 结束 MEM 命令, 单板机返回监控程序, 等待新的键盘命令, 显示提示符“P”。

## 3. 检查输入程序是否正确

① 按数字键 2000, 输入程序的首地址, 地址显示段显示  $2000_H$ 。

② 按 MEM 键。数据显示段应该显示  $2000_H$  单元中的内容  $3E_H$ , 否则就是输入出错。如果出错就要按照程序进行修改, 修改的方法只要重新按数字键 3E 即行。

③ 按 NEXT 键。检查下一个存贮单元内容, 此时数据显示段应该显示  $2001_H$  存贮单元中的内容  $15_H$ , 否则就是程序输入出错。

④ 重复按 NEXT 键, 数据显示段显示的内容应该和实验程序一致。

⑤ 检查完毕, 按 MON 键结束 MEM 命令, CPU 返回监控程序, 显示器显示提示符“P”。

## 4. 程序输入正确无误, 就可以执行程序

① 按数字键 2000, 输入被执行程序的首地址, 地址显示段应显示  $2000_H$ 。

② 按 EXEC 键。单板机执行实验程序。因实验程序最后一条是停机指令, 所以六位显示器全部不显示。

③ 按 MON 键。使单板机返回监控程序, 结束 EXEC 命令, 显示器显示提示符“P”。



注：不能用 RESET 键代替 MON 键结束 EXEC 命令，否则无法检查各寄存器里存放的数据和结果。

### 5. 通过检查寄存器中的内容，来分析程序执行结果的正确性。

参加运算的数和运算结果都存放在寄存器中，所以通过检查寄存器中的内容可以判断程序执行的正确性。寄存器检查的方法是：按数据键输入寄存器号，再按 REG 键，该寄存器的内容就显示在数据显示段。例：

#### ① 检查累加器 A 中的加法运算结果

按数字键 A。左边第一位显示 A，再按 REG 键，数据显示段应显示和数 27<sub>H</sub>。

#### ② 检查 B 寄存器中的加数

按 MON 键。结束检查 A 累加器命令，显示器显示提示符“P”。按数字键 B，左边第一位显示 B，再按 REG 键，数据显示段应显示加数 12<sub>H</sub>。按 MON 键返回监控程序。

### 6. 用单步方式执行实验程序

如果程序执行的结果不对，可以用单步方式逐条指令执行程序，再通过寄存器检查和存储器检查命令来检查每条指令执行的结果，找出程序中的错误。单步方式执行程序的操作方法是：把被执行的程序首地址预置在程序计数器 PC 中，然后每按一次 STEP 键，单板机就执行一条指令。

① 按数字键 1（PC 与 1 共键），左边第一位显示器显示 1。再按 REG 键，右边四位显示器显示 PC 中原来的内容 × × × ×。

② 按数字键 2000，以修改 PC 内容的方式，把程序首地址 2000<sub>H</sub> 置入 PC 中，右边四位显示器显示 2000<sub>H</sub>。

③ 按 STEP 键，单板机执行了第一条指令，停在下一条指令的地址上。此时左边四位显示器应显示下一地址 2002<sub>H</sub>，右边二位显示器始终显示累加器 A 内容。由于第一条指令的执行结果就是把被加数 15<sub>H</sub> 送累加器 A，所以不必再用寄存器检查键来检查累加器 A 的内容了。

④ 按 STEP 键，单板机执行了第二条指令停在下一条指令的地址上。此时左边四位显示器应显示下一地址 2004<sub>H</sub>，右边二位显示器仍旧显示 A 累加器原来的内容 15<sub>H</sub>。下面检查第二条指令执行结果：

按 MON 键结束 STEP 命令，显示器显示提示符“P”。

按数字键 B，再按 REG 键。此时左边第一位显示器显示 B，右边二位显示器应显示 B 寄存器内容。根据第二条指令执行的结果，B 寄存器中存放加数 12<sub>H</sub>，所以右边二位显示器显示 12<sub>H</sub>，表示第二条指令执行结果正确。

按 MON 键，结束 REG 命令，返回监控程序。

⑤ 按 STEP 键，单板机执行了第三条指令。此时左边四位显示器显示下一条指令地址 2005<sub>H</sub>，右边二位显示器显示累加器 A 内容。由于第三条指令是加法指令，所得的和数就存放在累加器 A 中，所以右边二位显示器显示的内容就是和数 27<sub>H</sub>，表示第三条指令执行结果正确。

到此为止，用单步方式执行实验程序的三条指令以及检查每条指令执行的结果，操作就完了。最后按 MON 键结束第三次 STEP 命令，返回监控程序，显示器显示提示符“P”。

### 7. 用设断点方法执行实验程序

上面介绍的第 6 点是用单步方式执行程序，单步方式可以方便地查出程序中的错误。但

是, 当程序比较复杂, 指令条数很多时就会感到单步方式太慢了。此时, 可以把程序分段, 用设断点的方法, 先把程序中的错误落实到某一段小程序中, 然后再用单步键配合找出错误, 这样调试程序既快又方便。设断点的方法也是一个很重要的方法。

下面以实验程序为例, 在  $2004_H$  地址和  $2005_H$  地址设置二个断点, 把实验程序分成两小段, 用设置断点的方法调试实验程序。

#### ① 设置两个断点

在看到显示器显示提示符“P”时, 按数字键 2004, 左边四位显示器应显示地址  $2004_H$ , 再按 BP 键, 如果仍显示  $2004_H$  地址, 表示第一个断点设置成功。

按 MON 键, 结束 BP 命令, 显示器显示提示符“P”。按数字键 2005, 显示器左边四位显示  $2005_H$ , 再按 BP 键, 如果仍显示  $2005_H$  地址, 表示第二个断点设置成功。

按 MON 键, 结束第二次 BP 命令, 显示器显示提示符“P”。注意: 不能用 RESET 键结束 BP 命令, 这样会把刚刚设置的断点消除掉。

#### ② 执行实验程序

按数字键 2000, 输入实验程序的首地址。

按 EXEC 键, 单板机执行第一小段程序的两条指令后, 遇到第一个断点停机。此时, 显示器左边四位应显示断点地址  $2004_H$ 。下面就检查执行这一小段程序的结果。

按 MON 键, 结束 EXEC 命令, 返回监控程序。按数字键 A, 再按 REG 键, 左边第一位显示器应显示 A, 右边两位显示器应显示累加器 A 的内容  $15_H$ 。

按 MON 键结束 REG 命令。

按数字键 B, 再按 REG 键, 左边第一位显示器应显示 B, 右边两位显示器应显示 B 寄存器的内容  $12_H$ , 表示第一小段程序的两条指令执行结果正确。

按 MON 键, 返回监控程序, 显示提示符“P”。

按 EXEC 键, 单板机执行第二段小程序, 遇到第二个断点停机。此时左边四位显示器显示断点地址  $2005_H$ 。

按 MON 键, 返回监控程序。

按数字键 A, 再按 REG 键, 左边第一位显示器应显示 A, 右边两位显示器应显示累加器 A 中的内容即加法的结果  $27_H$ , 表示第二小段程序执行的结果正确。

到此为止, 用设置断点的方法调试实验程序介绍完毕。值得注意的是: 在设置断点调试程序时, 如果程序是以 HALT 指令结束的, 不能在执行第二小段程序后, 再按 EXEC 键, 继续执行停机指令, 否则设置的断点不能消除。

消除断点的方法可参阅本实验指导书 §4 节第⑧小点(P.4)。

### 四、练习程序

#### 1. 用单步方式执行程序

例1: 把  $2100_H$  存储单元中的内容  $AA_H$  与  $2101_H$  存储单元中的内容  $BB_H$  进行交换。

首先用 MEM 键在  $2100_H$  单元写入  $AA_H$ ,  $2101_H$  单元写入  $BB_H$ 。

```
2000H: 21 00 21 LD HL, 2100H
2003H: 7E LD A, (HL)
2004H: 2C INC L
```

```

2005H: 46      LD    B, (HL)
2006H: 77      LD    (HL), A
2007H: 2D      DEC   L
2008H: 70      LD    (HL), B
2009H: 76      HALT

```

例 2: 把 A 累加器中的数递增 5, 并分别送 B、C、D、E、H、L 等寄存器。

```

2010H: 3E 00 LD    A, 00H
2012H: C6 05 ADD   A, 05H
2014H: 47      LD    B, A
2015H: C6 05 ADD   A, 05H
2017H: 4F      LD    C, A
2018H: C6 05 ADD   A, 05H
201AH: 57      LD    D, A
201BH: C6 05 ADD   A, 05H
201DH: 5F      LD    E, A
201EH: C6 05 ADD   A, 05H
2020H: 67      LD    H, A
2021H: C6 05 ADD   A, 05H
2023H: 6F      LD    L, A
2024H: 76      HALT

```

要求: 用单步方式和连续执行程序两种方法执行程序。检查并记录 B、C、D、E、H、L 等寄存器的内容。

2. 用设置断点的方法执行程序

例: 把累加器 A 加“1”到 05<sub>H</sub>

```

2030H: 97      SUB   A
2031H: 06 05      LD    B, 05H
2033H: 3C      LOOP, INC A
2034H: B8      CP    B
2035H: 20 FC      JR    NZ, LOOP-$
2037H: 76      HALT

```

要求: 在 2033<sub>H</sub> 地址设置一个断点, 执行程序, 记录遇到断点停机的次数和 A、B 寄存器检查的内容并分析为什么?

## 实验二 程序调试

### 一、实验目的

1. 通过实验熟悉 Z80 的指令系统和编程方法。
2. 进一步掌握用单步方式、设置断点的方法调试程序。

## 二、实验设备

TP-801 Z80 单板机一台  
5 V 2A 直流稳压电源一台

## 三、实验内容

实验程序 1: 把监控程序 0000<sub>H</sub>-00FF<sub>H</sub> 地址中的内容搬到 2400<sub>H</sub>-24FF<sub>H</sub> 地址中去。

2000<sub>H</sub>: 21 00 00 LD HL, 0000<sub>H</sub>; 把源程序首地址送 HL 寄存器对

2003<sub>H</sub>: 11 00 24 LD DE, 2400<sub>H</sub>; 把目的首地址送 DE 寄存器对

2006<sub>H</sub>: 01 00 01 LD BC, 0100<sub>H</sub>; 字节数送 BC 寄存器对

2009<sub>H</sub>: ED B0 LDIR ; 数据块传送

200B<sub>H</sub>: 76 HALT ; 停机

要求: ①检查数据搬家后 2400<sub>H</sub>-24FF<sub>H</sub> 部分单元内容与 0000<sub>H</sub>-00FF<sub>H</sub> 单元中的内容是否一致。

②如果要把 0000<sub>H</sub>-07FF<sub>H</sub> 地址中 2K 监控程序搬到 2100<sub>H</sub>-28FF<sub>H</sub> 单元中去, 怎样修改程序?

实验程序 2: 做十进制数加法  $28_D + 35_D = ( \quad )$ ? 并把结果写入 2100<sub>H</sub> 单元中去。

2010<sub>H</sub>: 3E 28 LD A, 28<sub>D</sub>

2012<sub>H</sub>: 06 35 LD B, 35<sub>D</sub>

2014<sub>H</sub>: 80 ADD A, B

2015<sub>H</sub>: 27 DAA ; 十进制调整。

2016<sub>H</sub>: 32 00 21 LD (2100<sub>H</sub>), A

2019<sub>H</sub>: 76 HALT

要求: 用单步方式执行程序, 记录十六进制数加法结果和调整后的十进制数加法结果。

实验程序 3: A 累加器加 1 到 FF<sub>H</sub> 结束。

该实验程序中有错误, 要求用单步方式执行程序, 查出并记录错误, 最后修改程序, 执行程序。

2020<sub>H</sub>: 97 SUB A; 清除累加器 A

2021<sub>H</sub>: 3C LOOP, INC A; 累加器 A 加 1

2022<sub>H</sub>: FE FF CP FF<sub>H</sub>

2024<sub>H</sub>: C2 20 20 JP NZ, LOOP

2027<sub>H</sub>: CF RST

实验程序 4: 编写三字节数加法程序并完成下式运算。

$$\begin{array}{r} 5A \quad E0 \quad B3 \\ + 27 \quad 1F \quad 6B \end{array}$$

首先把被加数和加数存放在 2100<sub>H</sub>-2105<sub>H</sub> 单元

2100<sub>H</sub>: B3

2101<sub>H</sub>: E0