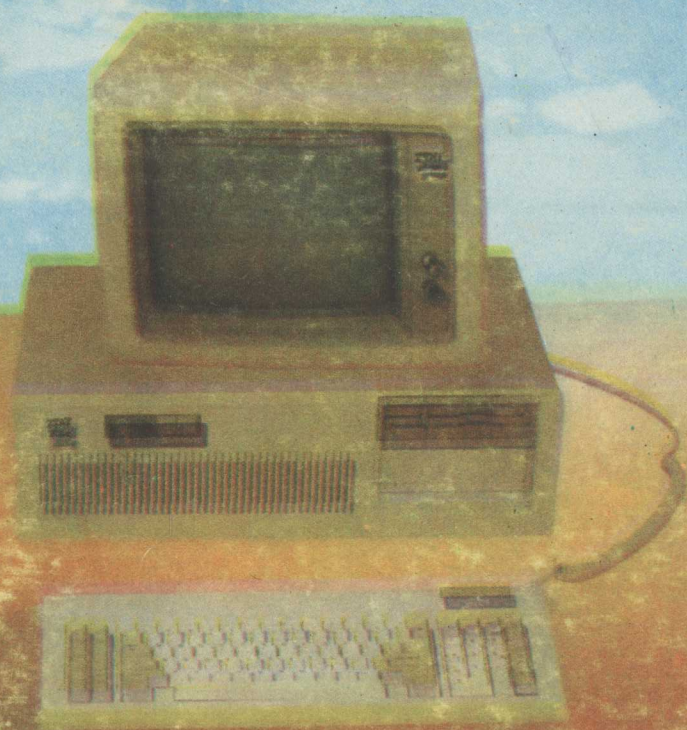


IBM PC/AT(IT-AT)实用技术丛书之六

IBM PC/AT 技术参考手册

(IT-AT适用)

中册



ROM BIOS 程序

软磁盘.....	(2)
磁盘.....	(59)
键盘.....	(108)
打印机.....	(140)
RS232	(146)
视频.....	(155)
BIOS	(227)
BIOS 1	(240)
BIOS 2	(293)
ORGS	(314)

软磁盘

标题 软磁盘 日期 01-12-84 软磁盘 BIOS

.LIST

C INCLUDE SEGMENT.SRC
C CODE SEGMENT BYTE PUBLIC
C

0000

PUBLIC DISK_INT.1
PUBLIC SEEK
PUBLIC DSSETTE.SETUP
EXTRN DDS. NEAR

0-----中断 13-----

0 软磁盘I/O

0 此接口提供访问 5 1/4 软磁盘驱动器，支持 320/360K 软磁盘驱动器和 1.2M 软磁盘驱动器
0 输入

0 (AH)=0

复位软磁盘系统

硬复位到 NRC，准备命令，在所有驱动器上要求的重新校准

0 (AH)=1

将系统的状态读入(AH)

使用来自最后一个操作的 DISKETTE.STATUS

0 用于读/写/验证/格式化的寄存器

0 (DL)-驱动器号(允许0-1，要检查的值)

0 (DH)-磁头号(允许0-1，要检查的值)

0 (CH)-磁道号(不要检查的值)

0 磁体 驱动器 磁道号

0 320/360 320/360 0-39

0 320/360 1.2M 0-39

0 1.2M 1.2M 0-79

0 (CL)-磁区号(不要检查的值，不用于格式化)



媒体

驱动器

扇区号

320/360

320/360

1-8/9

320/360

1.2M

1-8/9

1.2M

1.2M

1-15

(AL)-扇区数(不要检查的值)

媒体

驱动器

扇区的最大数

320/360

320/360

8/9

320/360

1.2M

8/9

1.2M

1.2M

15

(ES, BX)-缓冲器的地址(要求用于验证)

(AH)=2

将要求的扇区读入存储器

(AH)=3

从存储器写要求的扇区

(AH)=4

验证要求的扇区

(AH)=5

将要求的磁道格式化

为了格式化操作,缓冲器指针(ES, BX)必须指向用于磁道所要求地址区段的集合,每个区段由4个字节(C, H, R, N)组成,其中C=磁头号, H=磁头号, R=扇区号, N=每个磁道的字节数(00=128, 01=256, 02=512, 03=1024)。在磁道上每个扇区必须有一个入口。这个信息用于在读/写过程中找出请求的扇区。

在使软磁盘格式化之前,必须引用这个程序的功能调用17来设置要格式化的软磁盘的类型。

为了格式化320/360K,或1.2M软磁盘驱动器两个中的一个的320/360K媒体,必须将DISK-BASE的格式化参数的间隔长度改变到050H,还有EOT(磁道结束)的参数(磁道上最后一个扇区)必须设置到扇区/磁道的要求号-对于320K为8,对于360K为9。采用在绝对地址0-78上定位的磁盘指针指向DISK-

BASE.

在完成 320/360K 格式化操作时,应把各个参数恢复到它们相应的初始值。
 直接存取存储器类型。

(AH) = 15
 寄存器

(AH) - 若不置进位标志, 返回, 否则错误

00 - 不存在驱动器

01 - 软磁盘, 没有更改线可用

02 - 软磁盘, 有更改线可用

03 - 硬磁盘

(DL) - 驱动器号(允许 0-1, 要检查的值)

(AH) = 16
 磁盘变化线状态

寄存器

(AH) = 00 - 磁盘变化线无效

06 - 磁盘变化线有效和置进位

(DL) - 驱动器号(允许 0-1, 检查值)

(AH) = 17
 为格式化直接存取存储器类型

寄存器

(AL) - 00 - 不使用

01 - 在 320/360K 驱动器中的软磁盘 320/360K

02 - 在 1.2M 驱动器中的软磁盘 320/360K

03 - 在 1.2M 驱动器中的软磁盘 1.2M

(DL) - 驱动器号(允许 0-1, 要检查的值; 在软磁盘接到所使用板时或不使用)

在指定 1.2M 字节软磁盘驱动器时, 只检查磁盘变化状态。如果找到的磁盘变化线成为有效的
 话, 发生下列作用:

打算将磁盘变化线复位到无效状态。

如果打算接着为格式化设置直接存取存储器 (DASD) 类型和返回磁盘变化错误码。

如果打算破坏返回问题时错误码和把直接存取存储器 (DASD) 类型设置到一个推出不知道的媒体类型的预先决定状态。

如处于无效的磁盘变化结束或为格式化设置直接存取存储器类型。

数据变量——磁盘指针

到软磁盘参数的当前设置的双字指针

输出

AH = 操作的状态

定义状态位等于这个模块的数据段中的 DISKETTE STATUS 变量。

CY = 0 成功的操作 (在返回时 AH = 0, 除了没直接存取存储器类型, AH = (16))。

CY = 1 不成功的操作 (AH 有出错原因)

对于读/写/验证

DS, BX, DX, CH, CL 保留

注: 如果用软磁盘代码报告一个错误, 相应的作用是复位软磁盘, 然后再试读出操作, 由于没有马达启动延迟, 所以在该时要求三个再试来保证不是马达启动产生的问题。

软磁盘状态机器-绝对地址 40, 90 和 91

(驱动器 0-90, , 驱动器 1-91)

各位情况

PUBLIC DISKETTE.IO.1

DISKETTE.IO.1 PPROC FAR

```

0000      FB      STI
0000      S3      PUSH BX
0001      51      PUSH CX
0002      1E      PUSH DS
0003      56      PUSH SI
0004      57      PUSH DI
0005      55      PUSH BP
0006      52      PUSH DX
0007      8B EC   MOV BP, SP
0008      BE ----R MOV SI, DATA
000D      8E DE   MOV DS, SI
000F      80 FC 01 CMP AH, 1
0012      76 0F   JBE R4

0014      80 FA 01 CMP DL, 1
0017      76 0A   JBE R4

0019      C6 06 0041 R 01R5; DISKETTE.STATUS, BAD CMD, 无效驱动器地址, 终止
001E      BE 0000 MOV SI, 0
0021      EB 49   JMP SHORT OK

0023      50      PUSH AX
0024      E8 010C R CALL JI
0027      5E      POP SI
0028      8B D6   MOV DX, SI

```

```

; > ORG 0EC99H的入口点
; 接通中断回送
; 保存地址

; 保存段寄存器的值
; 在操作过程中保存全部寄存器

; 建立磁头参数指针

; 置数据区
; 检查复位和状态操作
; 若是的, 跳过驱动器检查

; 检查驱动器号的正确性
; 若正确, 继续

; 保证建立返回状态
; 终止命令

; 保存原操作以后重试
; 调用复位来保证恢复DI
; 入口恢复原操作
; 测试取原操作

```


002A	80	FE 01	CMP	DH, 1	！ 看是否是复位，还是状态操作
002D	76	3D	JBE	OK	！ 跳过状态更新
002F	F6	06 008F R 01	TEST	HF_CNTRL, DUAL	！ 转决定控制器板的类型
0034	74	36	JZ	OK	！ 软磁盘附加板
0036	80	FE 15	CMP	DH, 15H	！ 读磁盘变化状态或磁盘类型命令
0039	73	31	JAE	OK	！ 若是的，跳过状态处理
003B	8B	56 00	MOV	DX, (BP)	！ 恢复驱动器参数
003E	32	FF	XOR	BH, BH	！ 为该驱动器的媒体状态建立地址
0040	8A	DA	MOV	LI, DL	！
0042	8A	26 0041 R	MOV	AH, DISKETTE_STATUS	！ 取操作状态
0046	0A	E4	OR	AH, AH	！ 看是否有错误
0048	75	4C	JNZ	RETRY	！ 转到检查媒体变更
004A	8A	A7 0090 R	MOV	AH, DSK_STATE[BX]	！ 取驱动器的媒体状态
004E	F6	C4 10	TEST	AH, DETERMINED	！ 看是否已经设置媒体状态
0051	75	14	JNZ	OK2	！ 若设置，就不改变状态
0053	8A	CC	MOV	CL, AH	！ 取现在状态
0055	80	E1 07	AND	CL, STATE_MSK	！ 隔离出状态号
0058	80	C1 03	ADD	CL, 3	！ 增加已经设置的状态
005B	80	E4 F8	AND	AH, REV_STATE	！ 清除状态号
005E	0A	E1	OR	AH, CL	！ 置新的状态号
0060	80	CC 10	OR	AH, DETERMINED	！ 进行媒体状态设置
0063	8B	A7 0090 R	MOV	DSK_STATE[BX], AH	！ 保存在驱动器状态指示器

0067	C6	37	0092	R 00	OK2:	MOV	DSK_STATE[BX+2], 0	, 清已起动的原始状态操作
006C	EB	0004		OK:	MOV	BX, 4	, 取马达等待参数	
006F	8B	D6			MOV	DS, SI	, 再取原操作	
0071	50				PUSH	AX	, 保存返回值	
0072	E8	0382	R		CALL	GET_PARM		
0075	88	26	0040	R	MOV	MOTOR_COUNT, AH	, 为马达置定时器计数	
0079	58				POP	AX	, 恢复返回值	
007A	80	FE	15		CMP	DH, 015H	, 看是否读直接存取存储器(DASD)操作	
007D	75	05			JNE	R20	, 若不是, 跳过	
007F	86	E0			XCHG	AH, AL	, 将结果放在 AH	
0081	F8				CLC		, 置成功的操作	
0082	EB	08			JMP	SHORT R19	, 转开	
0084	8A	26	0041	R	MOV	AH, DISKETTE_STATUS	, 取操作状态	
0088	80	FC	01		CMP	AH, 1	, 置进行标志表示成功或失败	
008B	F5				CMC		, 恢复全部寄存器	
008C	5A			R19:	POP	DX		
008D	5L				POP	BP		
008E	5F				POP	DI		
008F	5H				POP	SI		
0090	F				POP	DS		
0091	59				POP	CX		
0092	5B				POP	BX		
0093	CA	0002			RET	2	, 恢复地址 , 扔掉保存的标志	
0096	80	3E	0041	R 06	RETRY, CMP	DISKETTE_STATUS, MEDIA_CHANGE	, 检查磁盘更改错误	

009B	74 54	JE	OK1	真错误就不重试
;				
009D	8A A7 0090 R	MOV	AH, DSK_STATE[BX]	取驱动器的媒体状态
00A1	80 E4 07	AND	AH, STATE_MSK	隔离出状态
00A4	80 FC 03	CMP	AH, 3	看看是否在状态3
00A7	73 BE	JAE	OK2	若为建立的状态, 则为真错误
;				
;-----处理状态 0, 1 和 2				
;				
00A9	FE C4	INC	AH	试下一个状态
00AB	80 FC 03	CMP	AH, 3	看看是否是非建立的状态的退出。
00AE	75 02	JNE	R2	若是, 将复位跳到开始处
;				
00B0	B4 00	MOV	AH, 0	在退出后, 试下一个状态
00B2	8A AF 0092 R	MOV	CH, DSK_STATE[BX+2]	取开始重试状态
00B6	80 E5 07	AND	CH, STATE_MSK	封锁状态位
00B9	3A EC	CMP	CH, AH	所试的全部状态
00BB	74 47	JE	OK3	若是, 则为真的错误
;				
;-----为重新试建建立状态指示器				
;				
00BD	8A AF 0090 R	MOV	CH, DSK_STATE[BX]	取状态指示器
00C1	D0 C5	ROL	CH, 1	将传送率送到低阶位
00C3	D0 C5	ROL	CH, 1	;
00C5	80 E5 03	AND	CH, TRAN_MSK	隔离出传送率位
00C8	EF CD	DEC	CH	转换到下一个速率
00CA	80 FD FF	CMP	CH, 0FFH	看看是否发生溢出

00CD	75 02	JNE R3	! 若没有溢出, 转到 R3
00CF	B5 02	MOV CH, XRATE	! 置到下一个速率
00D1	D0 CD	ROR CH, 1	! 将传进位放回对应的地方
00D3	D0 CD	ROR CH, 1	!
00D5	80 FC 01	CMP AH, 1	! 看看该状态是否要求双步
00D8	75 03	JNE R9	! 若不要求, 跳过设置双步
00DA	80 CD 20	OR CH, DOUBLE_STEP	! 转向所要求的双步
00DD	0A E5	OR AH, CH	! 状态组合成新指示器
00DF	88 A7 0090 R	MOV DSK_STATE[BX], AH	! 保存作为新指示器
! -----建立实际的重试操作			
00E3	8B 56 00	MOV DX, [BP]	! 恢复来自堆栈的参数
00E6	8B 4E 0A	MOV CX, [BP+10]	!
00E9	8B 5E 0C	MOV BX, [BP+12]	!
00EC	8B C6	MOV AX, SI	!
00FE	E9 0023 R	JMP R4	! 重试操作
00F1	8B 56 00	MOV DX, [BP]	! 恢复驱动器参数
00F4	E8 0604 R	CALL READ_DSKCHNG	! 转接磁盘更改线状态
00F7	75 03	JNZ OK4	! 若有效, 驱动器中没有软磁盘, 超时
00F9	E9 0067 R	JMP OK2	! 若无效, 在驱动器中有磁盘, 磁头更改
00FC	C6 06 0041 R 80	MOV DISKETTE_STATUS, TIME_OUT	! 若驱动器空, 表示超时

0101	E9 0067 R	JMP	OK2	
0104	C6 87 0090 R 80	MOV	DSK_STATE[BX], POA_START	; 错误启动接通电源假定
0109	E9 0067 R	JMP	OK2	
010C		DISKETTE_IO.1	ENDP	
				-----决定新的媒体状态, 在这里需要复位磁盘更改线
010C		PROC	NEAP	
010C	80 FC 01	CMP	AH, 1	; 测试复位和状态操作
010F	76 76	JBE	J1E	; 跳过状态检查和更新
0111	F6 06 008F R 01	TEST	HF_CNTRL, DUAL	; 转决定控制器板的类型
0116	74 11	JZ	J1A	; 软磁盘附加板
C118	80 FC 15	CMP	AH, 15H	; 测试磁盘更改状态或磁盘类型
011B	73 6A	JAE	J1E	; 跳过状态检查和更新
011D	50	PUSH	AX	; 保存原始参数
011E	53	PUSH	BX	; 保存参数
011F	51	PUSH	CX	; .
0120	52	PUSH	DX	; .
0121	E8 0604 R	CALL	READ_DSKCHNG	; 转读磁盘更改线状态
0124	74 0C	JZ	J1I	; 跳过处理盘更改线
0126	E9 05E2 R	JMP	J1F	; 处理磁盘更改线有效

0129	50	J1A.	PUSH	AX	保存原始参数
012A	53		PUSH	BX	保存参数
012B	51		PUSH	CX	,
012C	52		PUSH	DX	,
012D	E8 0604 R		CALL	READ_DSKCHNG	为软磁盘附加板选择驱动器
0130	EB 51		JMP	SHORT J1H	忽略磁盘更改状态
0132	8A 87 0090 R	J1I.	MOV	AL, DSK_STATE[BX]	为驱动器取媒体状态信息
0136	0A C0		OR	AL, AL	根本不检查状态信息
0138	75 06		JNZ	J1D	若存信息, 则不为缺省
013A	B0 80		MOV	AL, POA_START	将缺省取为状态0
013C	88 87 0090 R		MOV	DSK_STATE[BX], AL	将缺省建立为状态1
0140	3C 61	J1D.	CMP	AL, POA_DUAL	查看是否双步速率
0142	75 1E		JNE	J1G	跳过磁道检查
0144	8B 4E 0A		MOV	CX, [BP+10]	取原始磁道参数
0147	80 FD 28		CMP	CH, 40	查看磁道是否通过软磁盘(320)的末端
014A	72 16		JB	J1G	若不是, 转试该状态的操作
014C	C6 87 0090 R 02		MOV	DSK_STATE[BX], 02H	置下一个状态试试算法
0151	B0 02		MOV	AL, 02H	将新状态放在工作寄存器
0153	8A B7 0092 R		MOV	DH, DSK_STATE[BX+2]	取操作起始状态
0157	0A F6		OR	DH, DH	检查操作开始
0159	75 13		JNZ	J1C	若预先已启动, 跳过设置

015B	C6 87 0092 R 61	MOV	DSK_STATE[BX + 2], POA_DUAL	建立起始状态
0160	EB 0C	JMP	SHORT J1C	跳过已经进行了的下一个状态
0162	8A 97 0092 R	MOV	DL, DSK_STATE[BX + 2]	取起始媒体状态
0166	0A D2	OR	DL, DL	看看这是原操作, 还是重试
0168	75 04	JNZ	J1C	若是重试, 就忽略
016A	88 87 0092 R	MOV	DSK_STATE[BX + 2], AL	保存作为启动数据速率
016E	8A 0E 008B R	MOV	CL, LASTRATE	取最后一个所选择的数据速率
0172	3A C1	CMP	AL, CL	比较最后一个操作
0174	74 0D	JE	J1H	若相同, 进行选新的传送率
0176	A2 008B R	MOV	LASTRATE, AL	为下一次检查保存新的传送率
0179	D0 C0	ROL	AL, 1	将传送率移到低位
017B	D0 C0	ROL	AL, 1	
017D	24 03	AND	AL, TRAN_MSK	清所有位, 但数据传送率位除外
017F	BA 03F7	MOV	DX, 03F7H	地址软磁盘控制寄存器
0182	EE	OUT	DX, AL	置数据传送率
0183	5A	POP	DX	恢复参数
0184	59	POP	CX	
0185	5B	POP	BX	
0186	58	POP	AX	
0187	8A F0	MOV	DH, AL	在 DH 中保存“扇区”
0189	80 26 008F R 7F	AND	MOTOR_STATUS, 01FH	指示一个读操作
018E	0A E4	OR	AH, AH	AH = 0
0190	74 38	JZ	DISK_RESET	

0192	FE CC	DEC	AH	! AH = 1
0194	74 76	JZ	DISK_STATUS	
0196	C6 06 0041 R 00	MOV	DISKETTE_STATUS, 0	; 复位状态指示器
019B	FE CC	DEC	AH	; AH = 2
019D	74 6E	JZ	DISK_READ	
019F	FE CC	DEC	AH	! AH = 3
01A1	75 03	JNZ	J2	; 测试磁盘验证
01A3	E9 0240 R	JMP	DISK_WRITE	
01A6				; 测试磁盘验证
01A6	FE CC	DEC	AH	; AH = 4
01A8	74 6C	JZ	DISK_VERIFY	
01AA	FE CC	DEC	AH	! AH = 5
01AC	74 6C	JZ	DISK_FORMAT	
01AE	80 EC 10	SUB	AH, 10H	! AH = 15H
01B1	75 03	JNZ	J3	; 跳过磁盘类型操作
01B3	E9 0698 R	JMP	DISK_TYPE	; 转入完成磁盘类型操作
01B6	FE CC	DEC	AH	! AH = 16H
01B8	75 03	JNZ	J4	; 跳过磁盘更改状态
01BA	E9 0646 R	JMP	DISK_CHANGE	; 转入检查磁盘更改线状态
01BD	FE CC	DEC	AH	! AH = 17H
01BF	75 03	JNZ	J5	; 坏命令
01C1	E9 070D R	JMP	FORMAT_SET	; 转入为格式化置媒体/驱动器类型


```

01C4 C6 06 0041 R 01      MOV     DISKETTE_STATUS, BAD_CMD, 没有传送的扇区
01C9 C3                    RET     ; 未定义的操作
01CA J1                    ENDP

```

-----复位软磁盘系统

```

01CA PROC NEAR
01CA BA 03F2             DX, 03F2H ; 适配器控制口
01CD FA                 ; 没有中断
01CE A0 003F R          AL, MOTOR_STATUS ; 哪一个马达通电
01D1 24 3F              AND     AL, 03FH ; 去掉不使用位
01D3 B1 04              MOV     CL, 4 ; 移位计数
01D5 D2 C0              ROL     AL, CL ; 将马达值送到高半字节, 驱动器选择低半字节
01D7 0C 08              OR      AL, 8 ; 转向中断允许
01D9 EE                OUT     DX, AL ; 复位适配器
01DA C6 06 003E R 00    MOV     SEEK_STATUS, 0 ; 置全部驱动器上所要求的重算
01DF C6 06 0041 R 00    MOV     DISKETTE_STATUS, 0 ; 为软磁盘置正确的状态
01E4 EB 00              JMP     S + 2 ; I/O 等待状态
01E6 0C 04              OR      AL, 4 ; 断开复位
01E8 EE                OUT     DX, AL ; 断开复位
01E9 FB                 ; 重新允许中断
01EA E8 051A R          CALL    CHK_STAT_2 ; 在复位后面进行读出中断状态
01ED A0 0042 R          MOV     AL, NEC_STATUS ; 忽略错误返回和进行自己的测试
01F0 3C C0              CMP     AL, 0C0H ; 测试驱动器准备好的状态
01F2 74 06              JZ      J7 ; 一切都正确

```