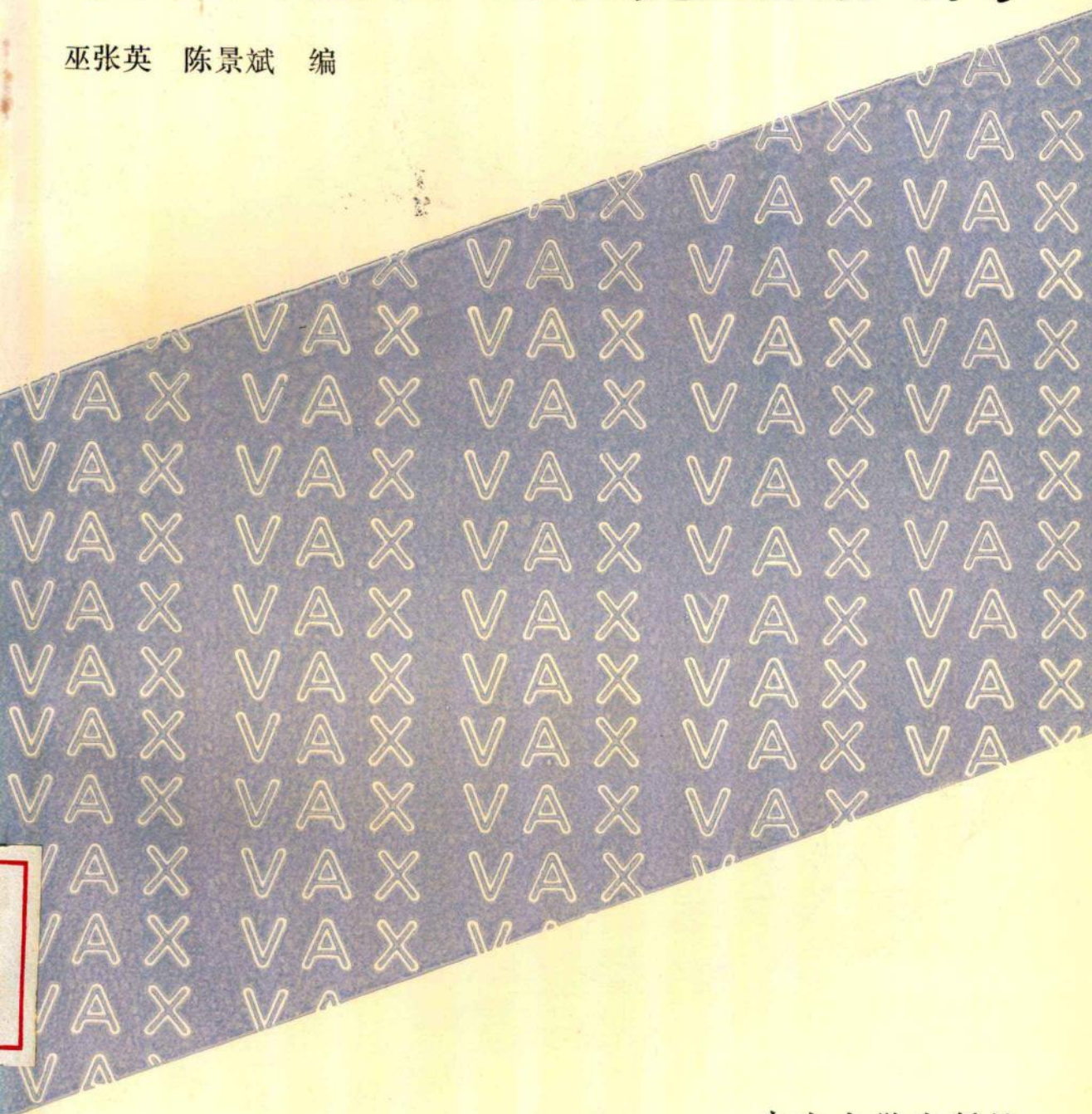


VAX CFMS

中文表格管理系统应用入门

巫张英 陈景斌 编



中山大学出版社

VAX CFMS

中文表格管理系统使用入门

巫张英 陈景斌编

中山大学出版社

VAX CFMS 中文表格管理系统使用入门

巫张英 陈景斌编

中山大学出版社出版发行
广东省新华书店经销
中山大学印刷厂印刷

787×1092毫米 16开本 7.25印张 16.9万字

1991年5月第1版 1991年5月第1次印刷

印数:1—3000册

*

登记证号(粤)第11号

ISBN7-306-00392-5/TP.3

定价:4.60元

前 言

在 VAX 系列机上推广使用的中文表格管理系统 CFMS (Chinese Forms Management System) 是英文表格管理系统 FMS 的扩充兼容版本。由于加入了中文支持, 所以, CFMS 对于开发在终端屏幕上使用汉字和英文表格的应用程序是一个非常理想的工具。

通常, 要使用电子计算机大都会通过终端屏幕来输入/输出数据。在用高级语言 (如编译 BASIC、COBOL 或 Pascal 等等) 进行屏幕画面布局、光标走向等的程序设计时, 如果这时的软件支持环境只有操作系统 (如 CVMS) 及某个高级语言 (如 BASIC), 那么, 要进行满意的屏幕输入/输出的程序设计是十分困难的, 其工作量也非常大, 即使设计出来, 程序也十分冗长。但是, 如果软件支持环境中增加了 CFMS, 并使用与 CFMS 有软件接口的高级语言 (如 BASIC、COBOL 或 Pascal 等等) 来进行关于屏幕输入/输出的程序设计, 就会使程序简短、屏幕画面布局美观而清晰、光标移动灵活。因此, 使用 CFMS 不但可以大大提高程序设计的效率, 而且可以大大为用户提供使用方便。

例如: 在一个不使用 CFMS 的 BASIC 源程序中, 为了使输入一个数据后, 可以对输入数据进行修改 (即重输), 要有如下一段程序语句:

```
L1:  INPUT "A="; A
      INPUT "认可 (Y/N) ?"; Z$
      IF Z$ <> "Y" THEN GOTO L1 END IF

L2:  INPUT "B="; B
      INPUT "认可 (Y/N) ?"; Z$
      IF Z$ <> "Y" THEN GOTO L2 END IF

L3:  INPUT "C="; C
      INPUT "认可 (Y/N) ?"; Z$
      IF Z$ <> "Y" THEN GOTO L3 END IF
```

```
L9:  INPUT "I="; I
      INPUT "认可 (Y/N) ?"; Z$
      IF Z$ <> "Y" THEN GOTO L9 END IF
```

由上述程序段可以看出, 虽然实现了可以输入 9 个数据, 并具有在每输入一个数据后能立即对刚输入的数据是否要重输 (即修改) 进行确认的功能, 但是, 没有提供对前面已输入的数据进行重输的功能 (如已输入了数据给 A 和 B, 现在正要给 C 输入数据, 在这时就不能输入数据给 A 或 B, 而只能输入数据给 C 了), 也没有提供对光标定位和光标向前后方向移动的功能。因此, 对用户来说, 用上述方法来实现终端屏幕上输入数据是很不方便的。

然而，如果使用了 CFMS，预先建立好接收数据的表格，在 BASIC 源程序中，除了几句有关 CFMS 初始准备的语句外，仅使用

```
CALL FDS$GETAL (STRING.X, FLD.TRM.L)
```

这样一个语句。而在运行程序时，就好像在终端屏幕上填表格那样来输入数据。光标开始是在接收第一个数据的字段，等待输入数据。输入数据后，按 Tab 键，光标就移到接收第二个数据的字段，等待输入第二个数据。同样，按 Tab 键，光标又移到第三个字段，等待输入第三个数据。但是，如果发现前面某字段的数据不对，那么通过按 BS 键（一次或多次），光标就可回退到前面的字段，等待重新输入需更改的数据。总之，用户只是通过按 Tab 键或按 BS 键，就可在屏幕上把光标移到所需要的字段上，等待输入数据。

由此可知，使用 CFMS 会給要在终端屏幕上输入/输出数据用户提供极大的方便。

除此之外，CFMS 对开发和维护 CFMS 应用软件人员也提供了特有的方便条件。在开发 CFMS 时，表格的建立与应用程序的编程是分别进行的。因此，对于表格中的某些文本、某些属性及某些数据可以独立进行更改。特别是对于程序中用到的某些以后需要修改的常数，可以取出来放在表格中，以后需要改变这些常数时，只需更改放在表格中的数据，而不必去重新修改、编译和连接原来的程序。这就提高了 CFMS 应用软件的易维护性和可扩充性。

我们单位是国内最早引进 DEC 公司的 VAX CFMS 的单位之一。我们以 CFMS 作为软件开发工具之一成功地开发出规模比较大的学校行政信息管理系统。本书就是编者在多年实践基础上写成的。

本书的第一、二、三、四及第六章由巫张英编写，第五章及附录由陈景斌编写。全书由巫张英主编。

虽然我们在 CFMS 应用软件开发方面做了一些工作，有了一些实践经验，但限于水平，书中难免有欠妥或谬误之处，尚请读者指正。

编 者

1990 年 3 月于中山大学

目 录

第一章	CFMS 概述	(1)
§ 1.1	CFMS 组成部分	(1)
§ 1.2	CFMS 应用程序开发步骤	(2)
第二章	建立表格	(7)
§ 2.1	调用中文表格编辑程序	(7)
§ 2.2	中文表格编辑	(9)
§ 2.3	表格阶段	(10)
§ 2.4	格式阶段	(14)
§ 2.5	赋值阶段	(21)
§ 2.6	数据阶段	(27)
§ 2.7	次序阶段	(29)
§ 2.8	测试阶段	(31)
§ 2.9	退出阶段	(32)
第三章	建立及维护表格库	(34)
§ 3.1	建立表格库	(34)
§ 3.2	把表格插入到表格库	(35)
§ 3.3	替换表格库中的表格	(36)
§ 3.4	从表格库中抽出表格	(37)
§ 3.5	从表格库中删除表格	(38)
第四章	中文表格应用辅助公用程序	(40)
§ 4.1	获得表格的描述	(40)
§ 4.2	列出表格库的目录	(42)
§ 4.3	建立内存长驻表格	(43)
§ 4.4	建立 UAR 向量的目标模块	(44)
第五章	中文表格驱动程序调用	(46)
§ 5.1	中文表格驱动程序调用表示法	(46)
§ 5.2	CFMS 与 VAX-11 BASIC 之间的语言接口	(48)
§ 5.3	初始准备及结束处理调用	(50)
§ 5.4	显示表格调用	(53)
§ 5.5	基本输入/输出调用	(54)
§ 5.6	多字段一起输入/输出调用	(57)
§ 5.7	滚动区输入/输出调用	(59)
§ 5.8	取回字段有关信息调用	(61)
§ 5.9	其他方面的一些调用	(62)
第六章	特定作用附程序 (UAR)	(65)
§ 6.1	字段完成 UAR	(65)
§ 6.2	求助 UAR	(69)
§ 6.3	功能键 UAR	(73)

第七章	CFMS 应用实例	(76)
§ 7.1	第一个实例	(76)
§ 7.2	第二个实例	(88)
附录 A	中文表格驱动程序调用列表	(100)
附录 B	CVMS 和 CFMS 状态码	(108)

第一章 CFMS 概述

DEC 推出的 VAX CFMS 中文表格管理系统与 VAX-11 FMS 英文表格管理系统兼容,它一方面管理在 VT100 兼容及 VT200 兼容的终端上运行的英文表格,另一方面又能替使用 VT82 或 VT382 中文终端的应用程序处理汉字表格。

用户可以利用 CFMS 来制造具有背景文本和字段的汉字及英文影像表格。CFMS 为影像表格增添了许多特性。CFMS 的交互性使操作员输入的数据即时得到确认,防止误入,如需输入数字时不致错入字母、需填上的字段不致留空等等。而且,CFMS 能使应用程序对输入的数据进行附加的确认和处理。因此,一般错误会立即发现,终端用户与计算机系统之间维持着不间断的顺畅交互作用。此外,CFMS 还为终端用户提供其他特性,包括:

- 可卷动表格,以便在屏幕上显示更多信息行。
- 只要操作员一按求助键,终端立即显示求助信息。
- 享用一系列的视觉效果,如字体可选择双倍大小、双倍宽度及表格可选择互相重叠等。

对于应用程序开发者来说,表格不仅包含背景文本和字段,而且还包含:适用于整张表格的某些属性,如表格的宽度;适用于个别字段的属性,如右边对齐。表格定义也容许应用程序开发者指定光标在字段间的移动次序,即可不遵照常规的自左至右、由上而下的次序,而是根据需要自行指定次序;而且可以指定附程序的程序名,以便在填完字段后或操作员按功能键时,CFMS 可自动调用附程序,或藉指定附程序名辅助标准 CFMS 求助请求处理。除表格可存储参数并传送给子程序外,CFMS 还提供附加参数存储器存储其他参数,以便程序执行时用户可以从表格说明中读出这些参数。

CFMS 支持下列语言:

- VAX-11 BASIC
- VAX-11 C
- VAX-11 BLISS-32
- VAX-11 COBOL
- VAX-11 FORTRAN
- VAX-11 PASCAL
- VAX-11 PL/I
- VAX-11 Rdb/VMS

§ 1.1 CFMS 组成部分

CFMS 中文表格管理系统由下列五个部分组成,用户可以利用这些组成部分来制造表格,以及编写和执行应用程序。

1. 中文表格编辑程序

中文表格编辑程序供用户设计、修改、测试和存储表格。用户使用中文表格编辑进行交互式的表格设计和修改；屏幕不断显示用户正在编辑的表格现状。中文表格编辑程序采用了菜单（即项目单）方式并且提供相应的小键盘编辑功能，使用户能比较容易地掌握并使用它来制造自己所需要的表格。

2. 中文表格库管理程序

中文表格库管理程序供用户建立和维护表格库文件。用户可在表格库中插入、替换、抽出或删除表格。

3. 中文表格驱动程序

中文表格驱动程序是一组附程序，供用户通过中文表格编辑来存取所制表格。

CFMS 应用程序通过用户放在源程序内的表格驱动程序调用来存取表格。所有表格驱动程序调用都按用户在表格编辑时赋值的名字来引用表格或表格内的数据。

4. 中文表格测试程序

中文表格测试程序供用户测试表格而无须事先把表格放入表格库内或编写应用程序。中文表格测试程序利用中文表格驱动程序显示用户的表格及进行输入/输出操作。当用户使用中文表格测试程序时，用户在屏幕上所见的表格，将与在应用程序运行时所见的一样，且用户可输入数据到表格内的字段里及检查输入的数据。

5. 中文表格应用辅助程序

在开发 CFMS 应用程序的过程中，用户可以使用中文表格应用辅助程序来进行如下操作：

- (1) 如果应用程序含有特定作用附程序，用户可制造特定作用附程序的向量模块。
- (2) 如果用户在应用程序中要使用长驻内存表格，用户可把表格转换为长驻内存格式。
- (3) 可列出表格库文件的目录。
- (4) 可为表格内所有字段制造 COBOL 数据说明。
- (5) 可印出表格编排格式的硬拷贝。

§ 1.2 CFMS 应用程序开发步骤

一、如何开发 CFMS 应用程序

用户开发 CFMS 应用程序通常包括（但不限于）以下步骤：

- (1) 使用中文表格编辑程序制造表格及使用中文表格库管理程序制造表格库。
- (2) 编写应用源程序。
- (3) 编译应用程序以获得目标模块。

- (4) 编写所需的特定作用附程序。
- (5) 编译所需的特定作用附程序。
- (6) 使用中文表格应用辅助程序所需的特定作用附程序向量模块。
- (7) 使用中文表格应用辅助程序制造所需的长驻内存表格。
- (8) 连接下列模块, 以形成可执行映像:
 - 主程序目标模块
 - 特定作用附程序目标模块
 - 特定作用附程序向量模块
 - 所有长驻内存表格
 - 中文表格驱动程序

二、 如何连接 CFMS 应用程序

在 HSYS\$ LIBRARY:HSYIMGLIB.OLB 和 SYS\$ LIBRARY: HSYOBLIB.OLB 这两个库内, 分别放置着中文表格驱动程序的共用映像和 CFMS 目标模块。这两个库必须与 CFMS 应用程序一起进行连接, 以产生可执行的映像。

例 1 假定无特定作用附程序和长驻内存表格, 则连接 CFMS 应用程序目标模块 PROG.OBJ 的连接命令如下:

```
$ LINK PROG.OBJ,SYS$ LIBRARY:HSYOBLIB.OLB / LIBRARY,-
  HSY$ LIBRARY:HSYIMGLIB.OLB / LIBRARY
```

此命令产生可执行映像 PROG.EXE。

例 2 假定特定作用附程序的目标模块为 VALID.OBJ 和 CHECK.OBJ, 特定作用附程序的向量模块为 TWOUAR.OBJ, 则连接 CFMS 应用程序目标模块 KYGL.OBJ 的连接命令如下:

```
$ LINK KYGL.OBJ,VALID.OBJ,CHECK.OBJ,-
  TWOUAR.OBJ,SYS$ LIBRARY:HSYOBLIB.OLB / LIBRARY,-
  HSY$ LIBRARY:HSYIMGLIB.OLB / LIBRARY
```

此命令产生可执行映像 KYGL.EXE。

三、 举例说明

为了使读者对 CFMS 应用能尽快有一个概貌, 方便上机实习, 故给出下面的简单例子。

例 3

(1) 假定用 \$ CFMS RESULT 命令进入中文表格程序编辑好表格 (如图 1-1)。

输入学生成绩

学	号: 99999
课	程: XXXXXXXX
成	绩: 999
认可(Y/N): A	

图 1-1 RESULT 表格

注: 表格 RESULT 包含有四个字段, 有关各字段的属性如下:

① 学号字段

字段名为 ST_NO, 属性有自动垮栏、字位全填、字段必填和监管态用。字段的求助文本为“请输入 5 个数字的学号”。

② 课程字段

字段名为 COURSE, 属性有字段必填。

字段的求助文本为“请输入课程名称, 最多可输入四个中文字”。

③ 成绩字段

字段名为 RESULT, 属性有字段必填。

字段的求助文本为“请输入学生成绩, 最多可输入 3 位数字”。

④ 认可字段

字段名为 CONFIRM, 属性有字段必填。

字段的求助文本为“若认可, 请输入字母 Y; 若不认可, 请输入字母 N”。

(2) 建立表格库 RESLIB.FLB

\$ CFMS / LIB / CREATE RESLIB.FLBRESULT.FRM

(3) 输入下面 INPUT_RESULT.BAS 源程序

00001 !程序名: INPUT_RESULT.BAS

!程序功能: 输入学生成绩

00100 MAP (RESULT_MAP) STRING ST_RESULT_STRING = 16%

MAP (RESULT_MAP) STRING ST_NO_X = 5%,

COURSE_X = 8%,

RESULT_X = 3%

MAP (RESULT_BUF) STRING RESULT_STRING_X = 17%

MAP (RESULT_BUF) STRING RESULT_ST NO_X = 5%,

RESULT_COURSE_X = 8%,

RESULT_RESULT_X = 3%,

```

                                RESULT_CONFIRM_X = 1%
DECLARE LONG CONSTANT FDK$K_KP_PER = 110%, !小键盘.
                                FDK$K_FT_NTR = 0%, !RETURN
                                FDK$K_FT_NXT = 1%, !Tab
                                FDK$K_FT_ATB = 3%!自动跨栏
DECLARE LONG FMS_TCA_L(3%),
                                FMS_WKSP_L(3%)
DECLARE LONGFLD.TRM.L !字段终止符变量
00500 OPEN "RESULT_FILE.DAT" AS FILE #1%, !打开学生成绩文件
                                ORGANIZATION INDEXEDFIXED,
                                ALLOW READ,
                                ACCESS MODIFY,
                                PRIMARY KEYST_NO_X,
                                MAP RESULT_MAP
00600 CALL FDK$ATERM(FMS TCA L( ), 12%, 20%) !连接终端
CALL FDK$AWKSP(FMS WKSP L( ), 3000%) !建立工作空间
CALL FDK$LOPEN("RESLIB",21%) !打开表格库
CALL FDK$SPADA(1%) !置小键盘为应用态
CALL FDK$CDISP("RESULT") !显示表格
00700 MAIN_LOOP:
    WHILE -1%
        CALL FDK$SPOFF !关闭监管态
        CALL FDK$GET(RESULT ST_NO_X,FLD.TRM.L,"ST NO") !输入学号
        EXIT MAIN_LOOP IF FLD.TRM.L = FDK$K_KP_PER
        CALL FDK$SPON !打开监管态
        CALL FDK$GETAL(RESULT_STRING_X,FLD.TRM.L) !输入其他各项
        ITERATE MAIN_LOOP IF FLD.TRM.L = FDK$K_KP_PER
        IF RESULT_CONFIRM_X <> "Y" THEN
            CALL FDK$PUTL(" * * * : 跳过") !显示一条信息
            ITERATE MAIN_LOOP
        END IF
        ST_NO_X = RESULT_ST_NO_X
        COURSE_X = RESULT_COURSE_X
        RESULT_X = RESULT_RESULT_X
        PUT #1% !将新记录写入文件
        CALL FDK$PUTL(" * * * : 已加记录!") !显示一条信息
    NEXT
10000 END_OF_PROGRAM:
    CALL FDK$LCLOS !关闭表格库

```

```
CALL FDV$ SPADA(0%) !置小键盘为应用态
CALL FDV$ CLEAR !清屏幕
CALL FDV$ DWKSP(FMS_WORKSPACE_L( )) !分离工作空间
CALL FDV$ DTERM(FMS_TERMINAL_L( )) !分离终端
CLOSE 1% !关闭学生成绩文件
```

19999 END

(4) 编译源程序

```
$ BASIC INPUT_RESULT
```

5. 连接

```
$ LINK INPUT_RESULT,-
SYS$ LIBRARY:HSYOBJLIB.OLB / LIBRARY,-
HSY$ LIBRARY:HSYIMGLIB.OLB / LIBRARY
```

6. 运行

```
$ RUN INPUT_RESULT
```

这时，屏幕上显示出表格 RESULT，用户可根据提示或通过按 Help 键得到的求助信息来输入有关内容。

第二章 建立表格

用户要在运行 CFMS 应用程序的映像时能够使用表格，就必须根据需要先建立好表格。CFMS 所提供的中文表格编辑程序是用户建立表格的一个非常有用的工具。特别是中文表格编辑程序提供菜单、问卷、小键盘功能键等功能，使新 CFMS 用户容易掌握它来建立表格。

§ 2.1 调用中文表格编辑程序

中文表格编辑程序是一个让用户在 VT82、VT382、VT100 兼容或 VT200 兼容等终端上制造和编辑表格的交互式公用程序。

一、设定终端特性

(1) 用户运行应用程序之前，键入下列命令来确保操作系统知悉用户的终端属性和屏幕特性。

```
$ SET TERMINAL/INQUIRE
```

操作系统将向用户的终端询问并记录其特性。如用户的终端特性与操作系统的记录不同，用户的 CFMS 应用程序可能不会正确地运行。CFMS 也需要 ANSI CRT 与 DEC CRT 属性恰当地设置。

(2) 要调用 CFMS 公用程序及应用程序，用户必须运行 HSY\$ SYSTEM: HANZIGEN.EXE 来设定输入装置位为 "HANZI"。

```
$ RUN HSY$ SYSTEM: HANZIGEN.EXE
```

```
HANZIGEN> SET / OUTPUT = HANZI
```

```
HANZIGEN> EXIT
```

(3) 利用命令过程文件来简化设定终端特性的击键操作。

为了使用户不必每次进入系统后都要通过键盘来键入上述 (1 和 2 中列出的) 命令，用户可以在自己的用户目录中利用 EDT 文本编辑程序建立下面二个文件：

① 命令过程文件 A.COM

```
$ SET TERMINAL/INQUIRE
```

```
$ ASSIGN / USER B.DAT SYS$INPUT
```

```
$ RUN HSY$ SYSTEM: HANZIGEN.EXE
```

```
$ EXIT
```

② 一般数据文件 B.DAT

SET / OUTPUT=HANZI

EXIT

用户进入系统以后,可在 DCL 命令级发出命令:

\$ @A

便可运行而不必键入先前 (1) 和 (2) 中所述的那些命令。

二、CFMS / EDIT 命令

用户可在 DCL 命令级 (通常出现提示符 \$) 使用 CFMS / EDIT 命令来调用中文表格编辑。

1. 命令格式

CFMS / EDIT { 表格文件说明
 表格库说明 / FORM__NAME = 表格名 }
 [/ [NO]OUTPUT [= 文件说明]]

表格库说明 代表一个表格库的文件说明

表格文件说明 代表一个表格文件的文件说明 (表格文件是含有单个二进制表格的文件。)

表格名 代表一个表格的名字

文件说明 代表一个文件的说明

注: 在 CFMS 命令中, 方括号 [] .. 表示方括号 [] 内的项目是任选的。大括号 { } .. 括起备待选项目的项目单。

(下同)

2. 命令描述

从上述的命令格式可以看出, 用户可用表格文件说明或表格库说明作为输入。表格文件说明这个参数代表用户要制造或编辑的表格的文件说明。如果用户未指定文件类型, 则 CFMS 根据缺省规则假定文件类型为 .FRM。表格库说明这个参数代表用户要编辑的表格所属表格库的文件说明, 如用户未指定文件类型, CFMS 假定文件类型为 .FLB。表格名这个参数代表用户要编辑的表格的名字; 而文件说明则代表输出文件的文件说明。

注意: /EDIT 是缺省的 CFMS 命令, 可不键入 /EDIT 而只键入 CFMS。

3. 限定词

/OUTPUT [= 文件说明]

/NOOUTPUT

用户要制造输出文件时, 应用 /OUTPUT 限定词加以指定。

如果未明确指定输出文件应放进那一个目录, 则输出的缺省目录有下列二种可能性:

(1) 如果指定了 /OUTPUT 限定词, 缺省目录将是当前目录。

(2) 如果未指定 /OUTPUT 限定词, 缺省目录将与输入目录相同。

如果用户未指定输出文件的文件说明, 则缺省输出文件说明有下列三种可能性:

(1) 用户制造新表格时, 缺省输出文件说明是用户提供的文件名加上文件类型 .FRM。

(2) 用户对表格文件里的现有表格进行编辑时, 缺省输出文件说明是版本号加 1 的输入文件说明。

(3) 用户对表格库内现有表格进行编辑时, 缺省输出文件说明是表格名加上文件类型 .FRM。(如果表格名含有 \$ 符或底线符, 将删去 \$ 符及底线符, 并截取表格名至头 9 个字符。)

如果用户不要制造输出表格, 应使用 /NOOUTPUT 限定词。此限定词适用于查看用户无权写入的目录里的表格。

4. 例

(1) \$ CFMS / EDIT STUDENT

或

\$ CFMS STUDENT

这二个命令效果上是一样的, 都是制造或编辑在表格文件 STUDENT.FRМ 里的表格。

(2) \$ CFMS SAMP / FORM_NAME=DEPOSIT / OUTPUT=DEPST

本例编辑表格库 SAMP.FLB 里的表格 DEPOSIT。输出文件为 DEPST.FRМ。

§ 2.2 中文表格编辑

在上节中已提到, 用户通过 CFMS / EDIT 命令调用中文表格编辑来制造或编辑表格。

制造或编辑表格包括进行多项操作, 例如:

- (1) 设定背景文本
- (2) 设定字段
- (3) 选取表格、字段、影像与行属性
- (4) 确定字段存取次序
- (5) 制造卷动区
- (6) 将特定作用附程序与表格相联
- (7) 将既定数据与表格相联

中文表格编辑可供用户在不同的阶段进行上述操作。用户在发出 CFMS / EDIT 命令 (例如:

CFMSSTUDENT 后, 屏幕上就显示出表格编辑菜单 (图 2-1)。表格编辑菜单上明确列出了中文表格编辑的七个阶段, 即:

1 表格 2 格式 3 赋值 4 数据 5 次序 6 测试 7 退出

用户制造新表格时，按上述阶段的先后次序来进行编辑操作较为有效。但是，中文表格编辑也可供用户不按次序使用各阶段，前后均可。用户可参照以下步骤来制造新表格：

- (1) 赋值各个表格属性、功能键与求助特定作用附程序的名字和数据，以及按需要赋值初始字段属性
- (2) 排表格的背景文本与字段格式
- (3) 赋值字段属性
- (4) 赋值既定数据
- (5) 赋值字段存取次序
- (6) 测试表格
- (7) 保存表格

中文表格编辑菜单是用户调用中文表格编辑后第一个出现在终端屏幕上的影像。菜单显示中文表格编辑的各个阶段。用户要进入某个阶段，只须键入该阶段的英文名或号码即可。当用户要进行其他操作时，可返回菜单并选择其他阶段。

表格编辑菜单

任选阶段： _____

1	表格	赋值表格属性
2	格式	制造或修改表格内容
3	赋值	赋值字段属性
4	数据	输入既定数据项目
5	次序	修改字段输入次序
6	测试	以表格驱动器测试表格
7	退出	结束编辑程序

表格名： STUDENT
输入文件： STUDENT.FRM;2

图 2-1 中文表格编辑菜单

§ 2.3 表格阶段

表格阶段让用户赋值表格属性、标识表格的特定作用附程序及赋值初始字段属性。表格属性是影响整张表格的特性，如屏幕色背景及屏幕宽度的列数等。特定作用附程序是用户将之与个别表格相联系、并在运行时间当操作员按功能键或 Help 键时，CFMS 调用的一些子程序。在表格阶段给初始字段属性赋值的功能使用户可以预选字段属性，以便中文表格编辑可将这些属性赋值于用户在当前编辑工作中制造的字段。