BL—Z8000微型计算机系统

软件参考手册

中国科学院物理研究所
福建电子计算机厂

软 件 参 考 手 册

ZMOS

Z8000多用户操作系统

目　录

9. 连接程序

10. 注释文件

11. 查询程序

12. 系统初始化文件

1. 总论

本章将对操作系统给出一个概略的描述，并定义在手册中所用到的一些术语。ZMOS 是 Central Data 公司编写的一种多用户操作系统，此操作系统适用于该公司的基于Z8000的多总线(Multibus)处理机板。关于系统硬件的详细情况可参阅各有关电路板的参考手册。

在本手册的全部正文内，以及向操作系统发所有的命令时，除了前面标有"H"者（例如H36），一切数字都当作十进制的。标有"H"的数字表示16进制数（基数为16）。

概述

ZMOS是为支持Central Data 公司的基于Z8000的中央处理机板而设计的，可以工作于多用户的情况。全部系统资源的管理是通过调用操作系统来完成的，从而对整个系统提供了高度的一致性和保密性。此系统可使系统的各用户分享文件，并自动防止由于两人访问同一数据记录所造成的无效数据。系统还提供一个简单而有效的保密机构，此机构允许文件由某些用户分享，但是限制其它用户访问。

给每个用户取一个名字，此名字被系统用以指示发出各种不同信息的用户，最后一个编辑文件者，等等。用户在最初进入系统时需要用口令（Password），这提供了一种高级保密措施。

总之，这个系统是为原来不具备计算机经验的人员使用而设计的。系统对每一条命令都有"辅助说明"（"help"）部分以资参考，并配以"计算机式样"的信息和格式。

终端

操作系统的软件能够支持的用户远多于硬件实际上能处理的数目。在正常情形下，一些实际限制(由于切换用户和处理终端设备的中断请求所引起的额外负担)使系统能处理的终端数不超过8个。如果在系统中采用智能输入/输出板（能完全处理所有终端设备的中断请求，从而释放出Z8000处理机的大量处理时间），则系统有可能处理至少两倍于上述数量的终端，最多可达32个。

当然，对每一个系统都必须分别根据每一终端的平均工作负载加以估算以确定其终端的最大数目。如果大多数终端在任何时刻都不同时使用，或者终端的使用只需要极少量的处理（如进行编辑、登入订单，等等），则此系统可以支持大量的终端。反之，如果在白天时大多数终端处于工作状态，经常做着很重的处理任务(进行汇编或编译等)，则能够得到有效服务的终端最大数目将会减少。

各式各样类型的终端能接入系统而不需要改变任何程序。此操作系统是为支持非智能终端而设计的，尽管只要编写专用的驱动程序也可以应用智能终端。在整个系统中采用标准控制代码对终端执行各种功能。终端驱动程序按照终端执行要求功能所需要的方式变换这些控制代码。

在整个系统中，大写字符和小写字符都能用来敲入命令、文件名字，等等，其效果相同。这就允许用户在通常情形下按小写方式，仅当需要时才换档至大写方式。

对每一个终端指定一道程序，当电源开始加至系统时即执行此程序。这道程序称为起动

程序，在极大多数情形下它是ZMOS的开工程序（sign—on program）。然而，用户可以有选择权，强令某个终端进入固定的程序。这样，这些终端的用户就不需要敲入名字。

每一个用户还有一道控制程序，在用户被接受后，以及每当用户在系统工作时按Control—X键时，就执行此程序。此控制程序常常就是ZMOS的命令处理程序，它允许用户敲入新的操作系统命令。当然，任何程序都能用于控制目的。另一种控制程序的例子可以是“指定”程序，此程序列出用户可能执行的几种程序的清单。然后用户可以由此清单中选择出他希望运行的程序。在这种工作方式，用户不需要涉及ZMOS命令语言。如果用户在其控制程序中敲Control—X键，用户即脱离系统，返回至终端起动程序。

磁盘存储

为处理很大的数据库，本系统能支持多达268兆位组的联机磁盘存储。所用的磁盘驱动器可以是软盘设备到大的卡式盘或温彻斯特盘。与终端设备类似，只需编写出相应的驱动程序，就能将不同类型的磁盘驱动器和控制器板加接至系统。

所有存储在磁盘上的信息都是以文件形式存在的。这些文件实质上类似于书本，因为每一文件通常只包含一种类型的信息。例如，一个文件可以包含公司的薪金名单记录，而另一个文件可以包含存货清单记录。每一文件又可以分割为相同大小的记录，这些记录就是信息的实际上的“页”。以存货清单为例，每一个记录可以包含元件号、数量、价格以及可能属于存货清单中每一元件的任何其它信息。文件中对应每一项有一个记录。因此，在薪金名单文件中，对每一名雇员有一个记录，而在存货清单文件中，对每一种元件有一个记录。

当程序需要由文件得到信息时，它就要求一个具体的记录。然后程序将记录分割为区段（如像元件号、数量和价格），这些区段将用作处理。能够看出，操作系统处理文件的各个记录，而程序必须说明所要求的记录，并将记录按照其应用目的所需划分为区段。

为了支持系统中的程序，提供了两种基本文件结构：随机存取和带索引的按序存取。随机存取结构最简单，对文件中的每个记录给定一个号数，文件开始处的记录为零，顺序向下排。对记录进行访问只需向操作系统提供相应的号数，然后操作系统将适当的数据送回给程序。旧的记录可以用同样方法进行改写，而以前存储的数据则被破坏。随机存取文件所具有的记录可包含2位组至512位组，按照2的倍乘增加（2、4、8、16……）。这些记录也能够用在每次读出或写入前将记录号加1的办法进行按序存取。

系统中的大多数文件是带索引的按序存取文件（ISAM）。这些文件要灵活得多，因为它们不是按照其在磁盘上的相对位置进行访问，而是利用每个记录中的一个区段进行访问的。例如，在一个存货清单文件中，元件号可以是用作访问的区段（关键区段），它允许用户程序说明对应于某个特殊元件号的记录应被读入。对于随机存取文件也可以这样做，但是必须事先对每一种可能的元件号在磁盘上分配空间，即使在号数之间存在间隙也需如此。ISAM文件在加入新记录时才分配磁盘存储，从而防止了磁盘空间的浪费。在ISAM中，任意数目的区段可以是关键区段，允许用户根据各种各样的参数去访问数据。而在用随机存取文件时，这是绝对不可能的。

ISAM文件系统为每一关键区段建立一个索引，在检索某个特殊记录时就要用此索引。这个索引具有“树”的结构，由顶部检索起，直到底部，它指向实际的数据记录。由于此树

状结构工作十分有效，尽管要检索的项数要大于随机存取文件，检索仍只需作最少次数的磁盘访问。

系统中的每个ISAM文件实际上由磁盘上两个具体的文件组成。第一个文件含有文件的数据记录，与一个简单的随机存取文件十分相像。另一个文件含有全部索引，这些索引是在根据关键区段值以确定特殊记录的位置时所需要的。这两个文件分别称为数据文件和索引文件。

系统中全部文件的建立方式都是使其能够易于改变大小。每个文件的目录包含一个该文件当前占用的磁盘扇区（实际位置）的清单。此清单中的每一项包括起始扇区号和长度（以扇区计）。在一个目录中能够有多到77个这样的项，这允许磁盘上的文件可分割为这样多部分而不需要通知用户知道。当加入新扇区至此文件时，总是尽可能将这些新扇区加于物理上靠近已经在文件中的其它扇区处。因此，这个方法使用户不必增加访问时间（因为扇区在磁盘上靠拢在一起），而几乎完全不需要用户为了能够加入更多的扇区，事先去“压缩”文件或磁盘。

系统中的磁盘驱动器是按一种独特的方式编号的。系统中的每块磁盘控制器板分配8个磁盘驱动器号。磁盘驱动程序决定由系统指定给控制器板的8个“逻辑”号是怎样分给实际磁盘设备的。大多数情形下，磁盘控制器板只能处理4个驱动器或8个驱动器，因此，在磁盘驱动器连接至控制器时，驱动器的逻辑号与磁盘驱动器号之间有一直接关系。

按此方法，如果系统中有两个磁盘控制器（例如，一个软盘控制器和一个卡式盘控制器），而连接至控制器的只有两个软盘驱动器和一个卡式盘驱动器，则系统能够访问的磁盘驱动器的逻辑号如下：0、1和8。在最初“安装”磁盘驱动器后，访问每个驱动器就利用名字，而不用逻辑号了。因此，通常系统管理人是唯一需要知道磁盘驱动器号的人。

组

组的概念对于理解本系统的全面能力是十分重要的。组被用来提供文件的保密性，它防止某些用户去访问某些文件，而允许经认可的用户访问他们所需要的文件。组的概念应用于计算机系统还是新近才出现的，但是它提供了一种十分强有力的手段或者保密措施。

组仅仅是一种将若干文件和用户相互联系在一起的方法。本系统可以有多至127个组，每一组通常只与系统的一个很窄的范围发生关系。一个组的管理人能够让用户成为该组的“成员”，允许这些用户去访问某些文件，在这些文件的访问代码表中列有该组（见下文）。

作为组的一个例子可以是这样一个机构，它从事有关公司销售方面的全部活动。这个组可以关联几个文件（应收帐目、待决定单、存货清单，等等），因此每一个该组成员应当被允许访问这些文件。不属于该组成员的用户不允许访问上述文件，除非他们所属的某个组也具有这些文件的访问权。

应当注意到，用户可以是多至8个组的成员，所以如果用户在系统中有几项活动，他能够属于为履行他的工作所需的所有组。而且，一个文件可以指定能对其进行访问的组，组的数目可多至8个。这些组具有阅读特许权或改写特许权（见本章下文中关于阅读/改写访问的部分）。因此，一个文件能够被多至8个不同组的用户利用，使能采取各式各样的用户和文件隔离方式。

在上述例子中，用两个组就能更为有效。允许低级售货雇员对待决定单文件进行改写访问和对存货清单文件进行阅读访问，但是不允许其对应收帐目文件进行访问。高级雇员可以允许对所有三个文件进行改写访问，使他们在有关公司销售信息方面具有不受限制的能力。

系统提供一个特殊的组，属于该组的任何用户可允许无限制地对所有系统资源进行访问。这个组称为“特权组”，特权组的成员被自动授予对全部文件的改写访问权，而且还能够作某些系统级的操作，这些操作是其它任何组都不允许作的。一个这样操作的例子是删除系统（见本手册的系统命令参考部份）。由于特权组具有这样的性质，只有在绝对需要的情形，才授予用户特权组成员资格，这是因为这些用户能对系统进行完全控制。

关于组的最后一个概念是对程序也可给与组的特许权，这和用户的情形十分相像。利用此点，一个程序文件（例如包含薪金名单程序的文件）可以被登录为组的成员，最多可以成为四个组的成员。按此方法，能够建立一个特殊的组以包含与薪金名单信息有关的全部文件（雇员文件、时间卡文件等），而薪金名单处理程序可以加入作为该组成员。在此情形下，薪金名单处理程序就能够访问这些文件，即使程序的用户不是这一特殊组的成员。这样，用户不被允许用他自己编制的程序去阅读或改写文件中的数据，但只要允许他访问薪金名单程序（由于用户具有对应的阅读访问代码），他就能用完全是程序定义的方法去访问数据。

组的编号从1至127，每组有一个名字。当一个新组被建立时（由特权组用户建立），必须提供该组的说明，使其它用户知道这个组是为什么目的而建立的。

系统中除特权组外还有两个特殊的组，126组和127组。126组用作所有注释文件的改写代码，这就允许注释文件处理程序能充分访问这些文件（因为注释文件处理程序属于126组）。127组是为系统打印机驱动程序所设的组，用来访问进程间通信块（Inter—Process Communication Block），为了解关于打印机驱动程序更多的信息，可参阅本手册的第6章。

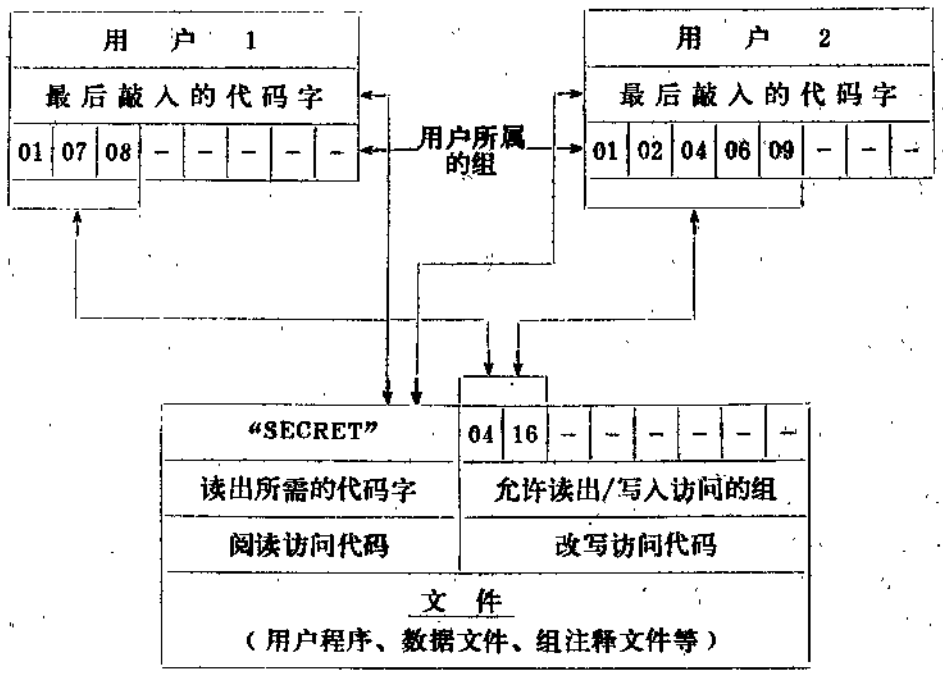
在正常组中，组的成员有两种类型：普通成员和组管理人。组管理人具有普通成员所不能作的一些附加功能，其中包括加入人组、从组中删除人、列出组的所有当前成员，以及删除该组所拥有的文件。

应当指出，一旦组建立后，就应当将其看作相对永久性的，因为要删除一个组以及属于该组的全部访问代码项可能需要化费很大力气。

访问代码

每个文件都具有与其相关联的阅读访问代码和改写访问代码。有对应的阅读代码就允许用户去读出文件记录、编译或汇编文件(如文件尚未代真)，以及执行文件（如文件已是汇编成的目的码）。有改写代码就允许用户完全控制文件，从读出记录和写入记录到改变访问代码或删除记录。只是拥有该文件的组的管理人才能删除整个文件。

两种访问代码是彼此独立的，尽管二者按同样的方式工作。每个访问代码是8个位组的数据，它可以是代码字，也可以是多至8个组的清单。如前文所描述的那样，如果给出的是组的清单，则用户必须属于其中的一个组才能获准访问该文件。代码字访问允许系统的任何用户进行访问，只需他们知道这个专有的代码字。此代码字最多可以有8个字符，而且必须在访问文件之前由用户敲入。已经指出，阅读访问代码和改写访问代码是彼此独立的。因



图例表示系统的两个用户和一个文件。用户必须知道，"SECRET"这个词是为读出文件所需的该文件阅读代码字。如果用户属于4组或16组，则他自动被允许读出/写入此文件。利用这些规则，如果用户1知道代码字，则他仅能读出此文件。但用户1决不能写入此文件。用户2能够读出和写入此文件而不必知道任何代码字，因为他属于4组。

此，作为文件访问代码的一个例子可以是阅读代码字"SECRET"，而改写访问代码仅允许两个组进行访问。假如你是这两个组中某个组的成员，你将根本不必知道该阅读代码字，即便你只要去阅读文件而对文件不作任何改写。

两种访问代码都可以保留为空白的，这样就允许系统中任何用户去访问此文件。这种选择功能应当慎用，并甚少用于改写访问代码。

文件名字和类型

系统中的每个磁盘文件都具有一个相关联的名字和类型，此文件名字和类型是程序在作任何读写操作前用以选择文件的。选择文件的这个过程称为"打开"文件。在此过程中，操作系统将有关该文件的信息填入系统的表内，这些信息在以后读写文件时将被用到。

文件名字的长度可以从1到8个字符，其第一个字符总是字母，接着的字符可以是字母，也可以是数字。文件名字常常选择得使其对该文件的用途有所说明。例如，包含薪金名单程序的文件可以称为PAYROLL，而包含发给每个雇员的薪金额数据的文件可以命名为PAYDATA。

系统中的汇编程序和编译程序（以及许多其它程序）对于每一个被处理的输入文件产生出若干输出文件。由于这个原因，对每一个文件还要给予文件类型，这就允许在系统中有几个文件具有相同的名字，但是属于不同的类型。采用汇编程序的例子，有一个输入源程序文

件，其输出有汇编成的目的码文件，以及几种选择输出文件，包括错误清单文件，输出清单文件和连接表文件（关于汇编程序的操作以及汇编程序所要用到和产生的文件等的详细情况载于本手册第 8 章中）。

不要求用户对由汇编程序产生的每一个文件提供不同的名字，而是对每一个输出文件用预先定义的文件类型来加以区分，其文件名字仍保持与输入文件相同。

用户要进行操作的文件的类型在整个系统中是隐含的。例如，如果用户敲入命令去编辑一个文件，则系统将认为被编辑的是一个原代码文件。如果用户要执行此同名文件（假定原代码已经过汇编），则系统将知道所需要的是一个目的代码文件。同样，连接文件将知道所需要的是连接表文件，而编辑程序中的"寻找下一个错误"命令将检索一个错误清单文件。按此方式，用户只涉及基本文件名字，而系统则关心几乎所有文件类型的产生和查寻。

当用户在一个命令中必需说明文件类型时，有几个命令允许用户只给出主类型（REG，ISAM等），而不必给出完全的文件类型（ISAM—TEXT等）。当在磁盘上检索文件时，找到的是该主类型中具有指定名字的第一个文件。例如，如果磁盘上有两个文件具有相同的名字，其类型分别为ISAM—BAS和ISAM—TEXT，当只用总类型ISAM来指定文件时，被找到的将是ISAM—BAS文件。这是因为在下文的类型清单中子类型BAS在子类型TEXT之前。作系统调用时，用户可以用传送类型号HFx的方法指明只给出主文件类型。例如，为指明任意IMAG文件，将使用文件类型HF1。

本系统的标准文件类型列表如下。类型号是一个数字，在整个系统中贮存时都是以该数字代表类型，以节省存储空间。类型名称是指定给该数字的符号化的名字，所有系统命令都是用类型名称敲入信息和进行显示的。

必须指出，由于系统中对每一个ISAM文件实际上存在两个物理文件（一个索引文件和一个数据文件），对于ISAM采用了三种文件类型。第一种类型是在同时访问索引文件和数据文件二者时所用的——访问ISAM文件的常规方法。另外两种类型是实际的索引文件和数据文件，在显示打开文件或磁盘目录时可以看到这两种文件类型。通常，用户应当将ISAM文件视作单个（逻辑上）文件，因为是系统照顾有关使用两个物理文件的全部操作。

类型号	类型名称	说明
H00	REG—DATA	通用数据文件
H10	REG—PCOM	内存的永久公共存储区
H20	REG—BTMP	磁盘位图
H30	REG—ERR	错误清单文件
H40	REG—LINK	连接表文件
H50	REG—SYMB	符号表文件
H01	IMAG—CPRG	公共象文件，程序
H11	IMAG—RPRG	正规象文件，程序
H21	IMAG—REL	浮动的公共象文件
H81	IMAG—OHLP	公共象文件，辅助部分

H91	IMAG—RHLP	正规象文件，辅助部分
H02	ISAM—SRC	汇编源程序代码，ISAM
H12	ISAM—BAS	BASIC源程序代码，ISAM
H22	ISAM—COB	COBOL源程序代码，ISAM
H32	ISAM—FORT	FORTRAN源程序代码，ISAM
H42	ISAM—PASC	PASCAL源程序代码，ISAM
H52	ISAM—PNOT	个人注释文件，ISAM
H62	ISAM—GNOT	组注释文件，ISAM
H72	ISAM—CMD	命令文件，ISAM
H82	ISAM—TEXT	任何文本文件
H92	ISAM—GRP	组文件，ISAM
HA2	ISAM—DIR	磁盘目录文件，ISAM
HB2	ISAM—ERR	磁盘/奇偶 错误 记录，ISAM
HC2	ISAM—DATA	通用文件类型，ISAM
Hx3	ISMI—××××	ISAM索引文件，见上文
Hx4	ISMD—××××	ISAM数据文件，见上文

标准编辑键

为了简化整个系统的用户接口，提供了一个标准的终端输入例行程序，它被应用于所有的程序中。此输入例行程序是用来使用户可以借助简单的控制按键进行字符和字符串的编辑。由于这些控制键在任何为系统编写的程序中皆执行同样的功能，用户的培训就应容易得多了。

用到的控制键列表如下，并对每一功能加以解释。建议用户在编辑程序中用这些按键作实验，因为实际经验比起单只阅读手册，是一种好得多的学习方法。

按　键	功　能
DEL 或 BS	从输入行删除一个字符。
Control—N	从输入行删除一个字符串。
Control—D	删除整个输入行。
Control—E	将输入行复写至编辑缓冲区内，并清除屏幕上的行。借助下列的复写按键能够将编辑缓冲区中的字符再复写回当前行。
Control—U	将编辑缓冲区中的下一个字符复写至输入行。
Control—I	将编辑缓冲区中的下一个字符串复写至输入行。
Control—P	将编辑缓冲区的剩余内容复写至输入行。
回　车	将显示的输入行送入。
Escape	由程序定义——通常是忽略整行。

除了这些编辑按键外，系统还支持Control—O键对终端进行接通和中止显示。在任何时

刻，用户都可以按Control—O以中止向屏幕发送数据。再按此键，将使显示从其停止处继续工作。这一特点允许用户在需要用显示去读出或者抄写下重要信息时使显示停住。

进程间通信

ZMOS操作系统的一个更为先进的特点就是进程间通信系统。此系统允许独立操作的一些进程（程序）彼此进行通信和同步。这种通信可以按照三种方式进行，即通过标志、暂时公共存储或者永久公共存储。要求怎样的通信方式取决于进行通信的程序。

每当在两个或多个进程间建立起进程间通信时，操作系统就产生一个进程间通信块(IPCB)以供使用。这个控制块贮存下列有关的全部信息：通信方式、允许访问的进程，以及当前连接至控制块（使用控制块）的进程。

如果利用标志作进程间通信，则在IPCB中含有一个64位的标志字。此标志字能够由任何连接至IPCB的进程进行阅读和改写，而进程可能阻塞（进入“保持”状态），一直到某些标志位被改写成所要求的花样。提供了各种系统调用来设置和清除标志位、读标志字，以及阻塞进程以等待某些位发生改变。

暂时公共存储为通信程序提供了更大的灵活性。采用这种通信方式，每个连接至IPCB的进程将具有一个存储段，此存储段为该进程与所有其它连接至IPCB的进程所共有。当此公共存储中任意一个位组被某个进程改变时，所有其它进程都将在他们的地址空间中看到其相应的位组发生改变。应当指出，每个进程可以有不同的段号，用以表示此公共存储，因为硬件存储管理电路作映象的工作，使每个进程对应于同一物理存储。

永久公共存储是所有三种类型中功能最强的，而且是较之其它两种应用得更多的方法。采用永久公共存储，当第一个进程连接至IPCB时，一个磁盘文件(文件类型为REG—PCOM)被送至公共存储区。这种方式允许将以前贮存的数据在用到存储器以前就自动送至公共存储。然后其它进程能够连接至IPCB（完全如同暂时公共存储方式），而在最后一个进程结束以后，公共存储由操作系统自动返回至磁盘文件。除了磁盘操作外，永久公共存储的工作方式与暂时公共存储的相同。

在所有三种进程间通信方式中，系统调用设计得允许任一进程可以首先连接至IPCB（事实上，直至第一个进程尝试去连接时才建立起IPCB），而不要求任何预先规定的进程连接顺序。还提供了系统调用去锁住IPCB。由于两个进程在同一时刻去改变标志或者公共存储中同一地址的内容会引起无效数据，锁住IPCB就能防止这种可能性。关于进程间通信的完全信息可参阅本手册第3章，在该章中阐明了有关这些操作的全部系统调用。

提交程序

ZMOS操作系统向其用户提供了SUBMIT命令这样一种十分有用的特点（见本手册第2章关于命令的详细说明）。此命令允许用户具有独立于用户终端执行的一个或多个程序。按此方式，用户可以在他做着人机对话工作(如进行编辑、命令查询等)的同时，运行一个耗费时间的程序（如汇编、作薪金名单等）。

利用这个特点，即使一个只具有几个终端的小系统也能够充分利用Z8000的处理能力。系统还直接支持一个打印程序，它可以在所有其它系统工作照常继续时进行打印文件。文件