

优化使用微机的内存

刘怀宇 (中国矿业大学综合系)

随着微型计算机硬件的不断发展,现在大多数微机都配有多于 640KB 的内存空间,如 286 配有 1 至 2MB,而 386 则更多,一般为 2 至 4MB,有的微机还配有扩充内存板。如果还使用 MS-DOS3.3 以下的磁盘操作系统,无疑是对内存资源的一种浪费,另一方面,应用软件的大型化趋势也要求使用更多的基本内存空间以提高软件运行速度。微软公司正是针对此种现实,在其 MS-DOS6 中提供了一系列内存管理程序。它不仅打破了原 DOS3.3 版本只能管理 640KB 内存的限制,而且针对不同的硬件环境,用户可选择使用不同的内存管理程序,使系统可按用户选择的方式管理内存,并最为有效地使用所有的内存空间。MS-DOS6 的推出,使 DOS 这一为用户所广泛使用的微机操作系统增添了生机。

一、微机内存的分类定义

就微机内存存储器的划分与定义,目前尚没有一个为大家所公认的明确和规范的标准,从不同的角度和技术深度会有着各不相同的划分和解释,笔者在此仅就普遍的硬件情况,从使用的角度加以简述。

微型计算机中所有内存存储器可分为四种类型:

1.基本内存(CONVENTIONAL MEMORY),也称常规内存,微机主板上 640KB 以下的内存存储空间,DOS 启动时的一些驱动和常驻程序要用到其中的一部分;另外所有的应用程序都要用到它。这一部分内存用户可通过 DOS 直接使用。DOS3.3 以下的版本也只能管理这一部分内存。

2.扩展内存(EXTENDED MEMORY),微机主板上 640KB 以上的内存存储器,这一部分内存 DOS 不能直接使用,要使用必须通过有关程序或设置引导,在 DOS3.3 中只能作为虚拟磁盘使用,而至于 DOS5, DOS6 是通过以扩展内存使用规范 XMS(EXTENDED MEMORY SPECIFICATION)为标准而编制的扩展内

存管理程序 HIMEM.SYS 来使用这一部分内存的。因此扩展内存也简称 XMS。

3.扩充内存(EXPENDED MEMORY),插在微机主板扩充槽中内存扩充板上的内存存储器,使用它也必须要有专门的程序从中引导,EMS(EXPENDED MEMORY SPECIFICATION)就是扩充内存使用规范,这一部分内存也简称 EMS。一般的微机都没有配备此种内存扩充板,目前要求必须使用扩充内存的程序也很有限。DOS5、DOS6 按 EMS 规范编制的 EMM386.EXE 驱动程序,可以在 386、486 微机上用扩展内存模拟扩充内存,使没有扩充内存的微机也能运行要求使用扩充内存的程序。

4.保留内存(RESERVED MEMORY),这一部分内存是给微机留做存储 I/O 系统数据及其各种接口卡驱动程序使用的,也称适配器内存(ADAPTER MEMORY),它一般在基本内存与扩展内存之间,不同的机型占用情况也不一样。当微机配有扩充内存时,这一部分内存中的剩余空间 DOS6 装载扩充内存使用规范。用户和应用程序不能利用这部分内存。

二、DOS6 主要内存管理文件简介

MS-DOS6 的内存管理文件主要包括以下四个:

1.MEM.EXE—系统当前内存使用情况列表命令文件。用户可以用这个命令显示各类型内存的分配情况;当前占用和剩余内存数量;还可显示当前调入内存的所有程序或模块的各方面的内存信息,如段地址、类型信息等。

2.HIMEM.SYS—扩展内存管理系统文件。它可以使用户与基本内存并列地使用计算机的扩展内存,包括高位内存区(HMA),并避免两个请求或设备驱动在同一时间对同一内存进行存取。

3.EMM386.EXE—386 档以上微机的高端内存块

(UMB)和扩充内存管理文件。它提供给用户设置和使用高端内存块的驱动功能,它还为那些没有扩充内存的微机用扩展内存模拟扩充内存。

4.MEMMAKER.EXE—386 档以上微机的内存状态设置管理文件。它通过把设备驱动程序和内存驻留程序移到高端内存等方式优化用户的计算机内存使用,使用户有更多的基本内存用于应用程序。使用这个管理软件,用户的计算机必须是 386 或 486 档次的微机,并配有扩展内存。

三、准备和使用更多的可用内存

DOS6 的内存管理特点是针对不同内存存储器的硬件情况,最优地划分不同类型内存的区域,为用户提供尽可能多的可用内存空间。

对于 286 档的微机 DOS6 的内存管理较为简单,如一台只有 1MB 内存(没有内存扩充板)的 286 微机,只需在 CONFIG.SYS 文件中加入以下两个命令行即可使用其全部内存。

```
DEVICE=C:\DOS\HIMEM.SYS
```

```
DOS=HIGH
```

其中第一行是使用扩展内存驱动,第二行是将 DOS 驻留于扩展内存中的高位内存区,重新启动 DOS 后,其内存划分情况如图 1,用 MEM 命令可显示内存使用和剩余情况如下:

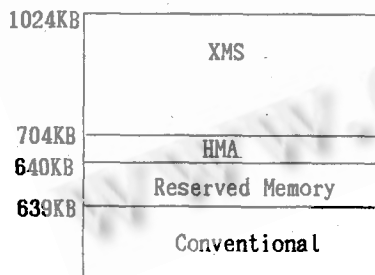


图 1 1MB 内存划分情况

它将原本应占用 640K 到 1024K 间的保留内存,移到基本内存中,并只占用其中的 1KB,将 384KB 的空间划为扩展内存,将 DOS 的大部分驱动和常驻程序驻留于扩展内存的前 64K 区域,即高位内存区(HMA),从而使启动 DOS 后的基本内存剩余空间可达 616KB。此后

,DOS 会自动地将用户的应用程序根据不同情况分配给扩展或基本内存。

Memory Type	Total =	Used	+ Free
Conventional	639K	26K	613K
Upper	0K	0K	0K
Adapter RAM/ROM	1K	1K	0K
Extended (XMS)	384K	324K	60K
Total memory	1024K	350K	673K
Total under 1 MB	639K	26K	613K
Largest executable program size			613K
Largest free upper memory block			0K
MS-DOS is resident in the high memory area.			

如四通网络汉字装载文件 INIT.COM 运行后,其本身调入基本内存,而它调用的程序则占用扩展内存。用 MEM 命令可以看到占用的基本内存只增加 3K,而扩展内存占用增加 260K。

Memory Type	Total =	Used	+ Free
Conventional	639K	23K	616K
Upper	0K	0K	0K
Adapter RAM/ROM	1K	1K	0K
Extended (XMS)	384K	64K	320K
Total memory	1024K	87K	936K
Total under 1 MB	639K	23K	616K
Largest executable program size			616K
Largest free upper memory block			0K
MS-DOS is resident in the high memory area.			

下面我们再看一看 DOS6 对一台配有 2MB 内存的 WYSE386 微机的内存是如何进行优化管理和使用的。DOS6 对于 386 微机的内存管理与划分进一步细化,当用户在其 CONFIG.SYS 文件中做如下设置时其内存分配具体如图 2:

```
DEVICE=C:\DOS\HIMEM.SYS
```

```
DEVICE=C:\DOS\EMM386.EXE RAM  
HIGHSCAN
```

```
DOS=UBM
```

```
DOS=HIGH
```

当用户将上面 CONFIG.SYS 文件第二行中的 RAM 参数改为 NOEMS 后,其内存分配具体如图 3

从以上二种情况可以看出,DOS6 将 640K 到 1024K 间的空间做为保留内存,1024K(1M)以上的 64K 为高位内存区(HMA),当用户需要用

EMM386.EXE 文件以扩展内存模拟扩充内存时,需占用该空间存放支持该功能的软件,否则它做为高端内存块(UMB)使用,1179K 以上的部分做为扩展内存使用或用于模拟扩充内存。用 MEM 命令可以看出它们不同的内存分配和使用效果。

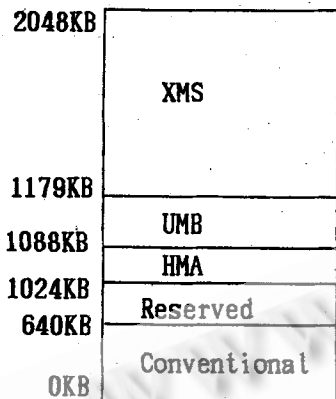


图 2 模拟扩充内存的情况

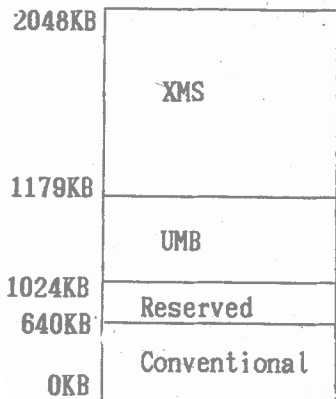


图 3 不模拟扩充内存的情况

对于第一种情况,它不仅可以使微机运行要求使用扩充内存的程序,并且可加快一般程序的运行速度,但它要占用 64KB 的 UMB 内存空间。对于第二种情况,它不能模拟扩充内存,但增加了 64KB 的 UMB 空间,可将更多的设备驱动程序移至高端内存块,从而节省更多的基本内存。因为目前要求必须使用扩充内存的程序很少,所以我们一般采用第二种设置,效果更为显著。

四、如何使用和优化高端内存块

对于 386 档微机,要特别说明其高端内存块(UMB)

的使用。使用 UMB 的程序可分为三种情况:

1.DOS 系统的驻留程序使用 UMB。如 DOSKEY.COM(命令管理文件,)在 C 盘 DOS 目录下,如果直接使用将被调入基本内存中,现要使其占用高端内存块,需在 AUTOEXEC.BAT 文件中加入:

```
LOADHIGH A:\DOS\DOSKEY.CON
C:\DOS\DOSKEY.COM
```

2.DOS 系统的设备驱动程序使用 UMB。如鼠标驱动程序 QMOUSE.SYS 在 C 盘 DOS 目录下,如果直接使用将被调入基本内存中,现要使其占用高端内存块,需在 CONFIG.SYS 文件中加入:

```
DEVICE HIGH=C:\DOS\QMOUSE.SYS
```

以上在 AUTOEXEC.BAT 和 CONFIG.SYS 文件中的定义运行环境的设置,都可以通过运行 MEMMAKER.EXE 文件,由它根据用户回答其提问的情况和调入 UMB 内的程序长度,在计算后给出最优化的存储起始段地址,并自动地形成符合用户所要求的内存运行方式的 AUTOEXEC.BAT 和 CONFIG.SYS 文件内容。

3.其它应用软件驱动文件使用 UMB。这一类驱动文件是由该应用软件系统的某些文件启动,如四通网络汉字系统中的系统调入文件(IBIT.COM)才能调入内存。如四通网络汉字系统中的 VGA 显示驱动程序(VSHELL.COM)和五笔字型输入法的驱动文件(WBX.COM),将它们调入高端内存块,不能在 AUTOEXEC.BAT 文件中加 LOADHIGH 命令,而应在 INIT.COM 成功运行之后,在系统提示符下键入

```
LH VSHELL
LH WBX
```

除 MS-DOS 系统数据,如 MSDOS.COM,COMMAND 等,HIMEM、EMM386 不能移至 UMB 中外,并不是所有的驱动程序都能在 UMB 中安全的运行。某些驱动程序因为其装入时要比运行时占用更多的内存空间,这种情况下有时 DOS 系统会将其自动转入基本内存,有的装入 UMB 后不能稳定的运行或运行无结果甚至死锁。

因为 DOS6 将一个程序装入 UMB 时,即使它只需一个较小的 UMB 就够了,DOS6 也使用最大的剩余 UMB。因此,在向 UMB 装入设备驱动程序时,应先用 MEM、C 命令查看装入内存的所有的各设备驱动程序

的占用量,一般以从大到小顺序调入。即使这样,有时也会出现 UMB 的剩余空间不足以装入下一个驱动程序的情况,这时用户应试验不同的程序组合和调入顺序,以充分利用 UMB 空间为原则,找出其最佳调入顺序和组合方案。

从下面的一个利用 DOS6 内存管理功能,优化使用内存的例子,读者可以更好地理解以上内容。

AUTOEXEC.BAT 文件内容

@ECHO OFF

SET TEMP=C:\DOS

LOADHIGH C:\DOS\DOSKEY.COM

C:\DOS\DOSKEY.COM

ECHO ON

CONFIG.SYS 文件内容

DEVICE=C:\DOS\HIMEM.SYS

DEVICE=C:\DOS\MEE386.EXE NOEMS

HIGHSCAN 256

BUFFERS=30,8

FILES=20

DOS=HIGH,UMB

DEVICEHIGH=C:\DOS\DBLSPACE.SYS(硬盘增

容的驱动程序,,43KB)

DEVICEHIGH=C:\DOS\QMOUSE.SYS/2/1

四通网络汉字驱动程序执行顺序之一

INIT

LH V SHELL

LH PY (先调入拼音输入法驱动程序)

LH WBX

四通网络汉字驱动程序执行顺序之二,(优化顺序)

INIT

LH V SHELL

LH WBX (先调入五笔字型输入法驱动程序)

LH PY

在优化调入顺序的基础上,我们可以进一步进行优化组合,如把 CONFIG.SYS 文件中的最后一行原来装入高端内存块的鼠标驱动程序改为调入基本内存,即改为

DEVICE=C:\DOS\QMOUSE.STS/2/1

用 MEM/C 命令可以列表显示不同设置情况下调入内存各程序占有内存的类型及数量。现将上面例子的四种情况内占用和剩余情况的总量汇集如下表

内存	不使用内存管理文件			优化顺序以前			优化顺序以后		优化组合以后	
类型	总量	占用量	剩余量	总量	占用量	剩余量	占用量	剩余量	占用量	剩余量
基本	640KB	236KB	404KB	640KB	74KB	566KB	43KB	597KB	37KB	603KB
高端	1024KB	以上的	内存空	155KB	115KB	40KB	146KB	9KB	152KB	3KB
扩展	间不能	做为内	存使用	869KB	477KB	392KB	477KB	392KB	477KB	392KB
合计	640KB	236KB	404KB	1664KB	666KB	998KB	666KB	998KB	666KB	998KB

五、小结

从以上例子和表格中的数据可以明显地看出:在微机装入同样程序的情况下,其基本内存从左到右系统占用量逐渐减少,剩余可用量逐渐增加;而其高端内存块则与之相反,占用量增加,剩余量减少。综上所述,使用

DOS6 的内存管理文件后,系统提供给用户可用的内存空间增加了,用户充分、合理、优化地使用 DOS6 的这些内存管理命令、功能,可以为用户的应用软件节省尽可能多的基本内存,并使扩展内存包括的高位内存区和高端内存块得到充分地利用。