



美国数据通用公司

外围设备程序员参考手册

美国数据通用公司

外 围 设 备

程 序 员 参 考 手 册

(015-000021-09)

天津市无线电技术研究所



内 容 简 介

本手册系根据美国数据通用公司 (DGC) 《USER'S MANUAL PROGRAMMER'S REFERENCE, PERIPHERALS》(015-000021-09) 译出, 为 NOVA 和 ECLIPSE 系列计算机程序员手册的姊妹篇。它提供了有关 DGC 各种外设的功能指标, 指令格式, 与外设通讯方式、总线约定, 和对每一种外设的程序设计举例等资料, 对于 NOVA 和 ECLIPSE 计算机, 以及 DJS100 系列计算机的维护、使用, 选用和增配外设, 特别是用户专用接口设计, 熟悉和掌握系统软件, 编制用户应用软件都是必要的。对于 DJS100 系列计算机的生产、设计和科研人员都具有一定的实用价值。

本手册由天津市无线电技术研究所技术情报室翻译, 并由各有关专业的科研人员进行了技术校对。由于我们水平不高, 尚望广大读者给予指正。

美国数据通用公司 外围设备程序员参考手册

出 版	天津市无线电技术研究所
及 发 行	天津市马场道 127 号
印 刷	天津大学印刷厂

1982年7月

前 言

本外围设备程序员参考手册系NOVA和日蚀系列程序员参考手册 (Programmer's Reference Manuals for the NOVA Line & ECLIPSE Computers)的姊妹篇。它提供了对标准的 DGC 外部设备和各种 DGC机型之间输入/输出 (I/O) 传输进行程序设计所需的一般原理和详细说明。

本手册共包括六章。第一章介绍数据通用公司 (DGC) 计算机的I/O程序设计原理并列举了 I/O 程序设计的几个完整的实例。其余章节叙述 DGC 的几个外围设备系列。第二章终端；第三章硬拷贝设备；第四章磁带设备，包括工业标准磁带机和 DGC 盒式磁带机；第五章固定头磁盘、盒盘和盘组；第六章模/数和数/模转换，以及模/数转换系统。附录包含若干标准表提供了有关设备代码、字符编码和 I/O操作所需要的定时数字。

在阅读本手册前，程序员应对DG公司计算机的程序设计基本上有所了解，如NOVA系列计算机程序员参考手册 (DGC015-000023)，以及日蚀计算机程序员参考手册 (DGC015-000024) 所述内容。建议还应该通晓 DG 公司的汇编程序手册 (DGC093-000017) 中所述内容，以及用汇编语言所写程序设计例子的操作。

另外，在接口手册 (DGC015-000031) 中还包括技术性更强的内容，该手册叙述了 I/O 总线的技术操作，说明非标准的外围设备如何与DG公司计算机相连接。

由于本手册的某些修改部分是按照新的版本规格排印的，所以某些章节与其它手册不同。请读者对此缺点加以原谅。

目 录

前言

第一章输入 / 输出程序设计	(1)
§ 1.1 输入 / 输出概述	(1)
1.1.1 引言	(1)
1.1.2 信息种类	(1)
1.1.3 信息传输种类	(2)
1.1.4 程序中断设备	(2)
1.1.5 摘要	(3)
§ 1.2 I/O 指令系统	(3)
1.2.1 引言	(3)
1.2.2 典型的控制器	(4)
1.2.3 指令格式	(6)
1.2.4 指令	(8)
§ 1.3 程序中断设备	(11)
1.3.1 引言	(11)
1.3.2 操作	(12)
1.3.3 指令	(15)
1.3.4 优先中断	(17)
§ 1.4 数据通道装置	(22)
1.4.1 引言	(22)
1.4.2 控制器结构	(23)
1.4.3 传送顺序	(24)
1.4.4 数据通道地址变换 (MAP) 选择	(25)
1.4.5 程序设计	(25)
§ 1.5 计时	(26)
1.5.1 引言	(26)
1.5.2 直接程序控制	(26)
1.5.3 数据通道控制	(27)
§ 1.6 程序设计实例	(29)
1.6.1 引言	(29)
1.6.2 实例 1	(29)
1.6.3 实例 2	(32)
1.6.4 实例 3	(36)
1.6.5 实例 4	(51)
第二章终端	(64)

§ 2.1 终端概述.....	(64)
§ 2.2 电传打字机.....	(64)
2.2.1 引言.....	(65)
2.2.2 指令.....	(66)
2.2.3 程序设计.....	(67)
2.2.4 计时.....	(68)
2.2.5 几点考虑.....	(68)
2.2.6 程序设计实例.....	(69)
§ 2.3 DGC 6012显示器.....	(71)
2.3.1 引言.....	(72)
2.3.2 指令.....	(73)
2.3.3 控制字符.....	(75)
2.3.4 控制键.....	(77)
2.3.5 程序设计.....	(78)
2.3.6 计时.....	(79)
2.3.7 几点考虑.....	(79)
第三章 硬拷贝.....	(80)
§ 3.1 硬拷贝外设概述.....	(80)
§ 3.2 纸带读入机.....	(80)
3.2.1 引言.....	(81)
3.2.2 指令.....	(81)
3.2.3 程序设计.....	(82)
3.2.4 计时.....	(83)
3.2.5 几点考虑.....	(83)
3.2.6 程序设计实例.....	(83)
§ 3.3 纸带穿孔机.....	(84)
3.3.1 引言.....	(85)
3.3.2 指令.....	(86)
3.3.3 程序设计.....	(87)
3.3.4 计时.....	(87)
3.3.5 几点考虑.....	(87)
3.3.6 程序设计实例.....	(87)
§ 3.4 卡片读入机.....	(89)
3.4.1 引言.....	(90)
3.4.2 指令.....	(91)
3.4.3 程序设计.....	(92)
3.4.4 计时.....	(93)
3.4.5 错误状态.....	(93)
3.4.6 程序设计实例.....	(94)

§ 3.5 宽行打印机	(97)
3.5.1 引言	(97)
3.5.2 指令	(98)
3.5.3 程序设计	(100)
3.5.4 控制字符	(101)
3.5.5 计时	(102)
3.5.6 程序设计实例	(102)
§ 3.6 增量绘图仪	(105)
3.6.1 引言	(106)
3.6.2 指令	(107)
3.6.3 程序设计	(107)
3.6.4 计时	(108)
3.6.5 几点考虑	(108)
3.6.6 程序设计实例	(108)
第四章 磁带	(111)
§ 4.1 磁带概述	(111)
4.1.1 工业兼容磁带机	(111)
4.1.2 DGC 盒式磁带机	(114)
§ 4.2 工业兼容磁带子系统	(114)
4.2.1 引言	(117)
4.2.2 指令	(118)
4.2.3 程序设计	(121)
4.2.4 计时	(128)
4.2.5 错误状态	(128)
§ 4.3 DGC 盒式磁带子系统	(129)
4.3.1 引言	(130)
4.3.2 指令	(131)
4.3.3 程序设计	(134)
4.3.4 计时	(138)
4.3.5 错误状态	(139)
第五章 磁盘	(141)
§ 5.1 DGC 磁盘概述	(141)
5.1.1 DGC 磁盘数据格式	(141)
5.1.2 磁盘驱动器的存取	(142)
5.1.3 DGC 磁盘子系统	(143)
5.1.4 共享磁盘的几点考虑	(143)
§ 5.2 固定头磁盘子系统	(144)
5.2.1 引言	(145)

5.2.2 指令.....	(145)
5.2.3 程序设计.....	(148)
5.2.4 计时.....	(150)
5.2.5 磁盘格式.....	(150)
5.2.6 错误状态.....	(151)
§ 5.3 4047A和 4047B 盒式磁盘子系统.....	(151)
5.3.1 引言.....	(153)
5.3.2 指令.....	(153)
5.3.3 程序设计.....	(157)
5.3.4 计时.....	(161)
5.3.5 错误状态.....	(162)
§ 5.4 4048A 磁盘组子系统.....	(163)
5.4.1 引言.....	(164)
5.4.2 指令.....	(165)
5.4.3 程序设计.....	(169)
5.4.4 计时.....	(173)
5.4.5 错误状态.....	(173)
§ 5.5 4057A 磁盘组子系统.....	(176)
5.5.1 引言.....	(177)
5.5.2 指令.....	(177)
5.5.3 程序设计.....	(182)
5.5.4 计时.....	(186)
5.5.5 错误状态.....	(186)
§ 5.6 4231A磁盘组子系统.....	(189)
5.6.1 引言.....	(190)
5.6.2 指令.....	(190)
5.6.3 程序设计.....	(195)
5.6.4 格式.....	(199)
5.6.5 计时.....	(199)
5.6.6 错误状态.....	(199)
§ 5.7 6045, 6050, 6051型盒式DG/磁盘子系统.....	(201)
5.7.1 引言.....	(203)
5.7.2 控制器寄存器.....	(203)
5.7.3 指令.....	(204)
5.7.4 程序设计.....	(208)
5.7.5 格式形成.....	(213)
5.7.6 计时.....	(213)
5.7.7 错误状态.....	(214)
5.7.8 多磁盘驱动器.....	(216)

5.7.9 双处理机的几点考虑	(217)
§ 5.8 6030系列 DG 软磁盘子系统	(220)
5.8.1 引言	(222)
5.8.2 控制器寄存器	(222)
5.8.3 指令	(223)
5.8.4 程序设计	(227)
5.8.5 格式形成	(231)
5.8.6 计时	(232)
5.8.7 错误状态	(232)
5.8.8 多磁盘驱动器	(235)
5.8.9 双处理机的几点考虑	(236)
§ 5.9 4234系列 10MB 盒式磁盘子系统	(239)
5.9.1 引言	(240)
5.9.2 控制器寄存器	(241)
5.9.3 指令	(241)
5.9.4 程序设计	(245)
5.9.5 格式形成	(249)
5.9.6 计时	(249)
5.9.7 错误状态	(250)
5.9.8 多磁盘驱动器	(253)
§ 5.10 6060 系列 DG/磁盘存储器子系统	(254)
5.10.1 引言	(256)
5.10.2 控制器寄存器	(257)
5.10.3 指令	(257)
5.10.4 程序设计详细情况	(264)
5.10.5 格式形成	(267)
5.10.6 计时	(268)
5.10.7 详细的命令说明	(269)
5.10.8 故障标志	(270)
5.10.9 错误状态	(273)
5.10.10 数据错误的校正	(274)
§ 5.11 6063—6065系列 DG 固定头磁盘子系统	(277)
5.11.1 引言	(279)
5.11.2 控制器寄存器	(279)
5.11.3 指令	(279)
5.11.4 命令排队格式	(283)
5.11.5 程序设计	(286)
5.11.6 状态校验	(288)
5.11.7 计时	(289)

5.11.8	错误状态	(289)
5.11.9	双处理机的几点考虑	(290)
5.11.10	数据错误的校正	(290)
§ 5.12	6070 系列盒式 DG/磁盘子系统	(294)
5.12.1	引言	(295)
5.12.2	控制器寄存器	(295)
5.12.3	指令	(296)
5.12.4	程序设计	(301)
5.12.5	格式形成	(305)
5.12.6	诊断方式	(306)
5.12.7	计时	(306)
5.12.8	错误状态	(307)
5.12.9	多磁盘驱动器	(310)
5.12.10	双处理机的几点考虑	(311)
第六章	模拟/数字和数字/模拟	(315)
§ 6.1	DGC A/D 和 D/A 子系统概述	(315)
6.1.1	模拟数据和数字数据	(315)
6.1.2	数据精度	(317)
§ 6.2	4120 系列模/数转换器系统	(318)
6.2.1	配置	(319)
6.2.2	操作	(320)
6.2.3	指令: 单转换方式	(322)
6.2.4	程序设计: 单转换方式	(326)
6.2.5	指令: 数据通道方式	(327)
6.2.6	程序设计: 带有数据通道选件的单转换方式	(333)
6.2.7	计时的几点考虑	(334)
§ 6.3	4180 系列数/模转换器系统	(335)
6.3.1	配置	(336)
6.3.2	操作	(336)
6.3.3	指令: 数/模	(337)
6.3.4	程序设计: 数/模	(338)
6.3.5	显示控制选件	(340)
6.3.6	指令: 显示器控制	(341)
6.3.7	程序设计: 显示器控制	(342)
附录 A	输入/输出设备码和数据通用公司助记符	(345)
附录 B	八进制和十六进制转换	(347)
附录 C	ASCII—128 字符编码	(348)
	EBCDIC 字符码 (扩充的二进制编码的十进制交换码)	(353)
附录 D	NOVA 系列计算机指令执行时间	(361)
	ECLIPSE 计算机指令执行时间	(363)
附录 E	辅助编码	(370)

第一章 输入/输出程序设计

§ 1.1 输入/输出概述

1.1.1 引言

输入/输出乃是在计算机的中央信息处理机 (CPU) 与外围设备, 如卡片读入机, 宽行打印机, 终端设备和磁盘机等之间传输信息的过程。某些外围设备, 如卡片读入机, 可向系统输入信息。而有些外围设备, 如宽行打印机, 是将系统的信息输出。还有些外围设备, 如终端, 则既可输入又可输出。其它外围设备, 如磁盘机, 乃供存贮计算机系统之中的信息之用。所以, 外围设备主要有二个用途: 1. 向计算机提供与周围环境通讯的工具; 2. 为内存补充了外存贮器。

外围设备通常包括二部分, 即一个设备和一个控制器, 但也可以包括一个转换器。设备, 有时亦称驱动器 (drive), 传送器 (transport) 或者终端, 是用以读, 写, 存贮, 或处理信息的。例如终端的键盘“读入”信息; 绘图仪“写出”信息, 磁带机“存贮”信息; 而A/D转换器“处理”信息。

控制器是计算机与设备之间的接口, 可解释计算机发给外设的命令, 并在二者之间传送信息。例如, 活动臂磁盘控制器可将来自计算机的磁道地址翻译成供磁盘驱动器存取机构执行的定位命令。存取机构将读/写磁头定位后, 控制器立即将所收到的来自计算机的数据字翻译成磁盘驱动器需要的位序列。

转换器是某些外围设备需要的附加部分, 用于完成外围设备与控制器之间的通讯连接。其功能与外围设备, 或者控制器的功能相同, 或者兼有二者的功能。由于程序员接触不到转换器, 所以它通常是不透明的。

在计算机与控制器之间传送全部信息的通讯通道, 称为输入/输出 (I/O) 总线。I/O总线的中心部分是16位宽的双向共享数据总线。由于此总线是由全部控制器以及CPU所共享的, 根据需要它是半双工的, 即在任何时候它只能发生一个操作。全部信息在I/O总线上传输的方向对计算机来说是有规定的。“输出”总是指由计算机向控制器传送信息; “输入”总是指由控制器向计算机传送信息。

1.1.2 信息种类

在计算机与控制器中间传送的信息可以分为三类: 即状态, 控制和数据。状态信息可告知计算机有关外围设备的状态, 是否忙? 是否“准备好”? 操作是否正确? 等。控制信息由计算机传送给控制器, 告知外围设备要作什么。数据乃是当读, 写, 存贮, 或处理时由设备发出 (或送到设备) 的信息。

不管所传输信息的类型,全部信息都要被传送。可以传输一位;也可以传输构成一个字节(或字符)的几位;构成一个字的16位,或者是构成一个字块的一组字。

1.1.3 信息传输种类

在计算机与外围设备之间可以采用二种方法之一进行传输;即直接程序控制下传输或者数据通道控制下传输。在直接程序控制下发生的信息传输,是在CPU的累加器与控制寄存器之间传输“字”,或字的一部分。当执行程序中的一条适当的I/O指令时,则发生这种传输。在数据通道控制下完成的信息传输,一般是通过控制器的一个寄存器在计算机的内存贮器与各设备之间每次传输一组数据,一个字。只要是程序使用I/O指令,确定对某一外围设备的传输,就可以经数据通道自动地传输数据组。

(1) 直接程序控制

信息传输的直接程序控制,也称为“程序设计的I/O”,是在CPU与外围设备之间传输单个字或部分字的一种方法。用这种方法传输数据的外围设备,有终端,纸带读入器和穿孔器,卡片读入器,宽行打印机和绘图器。由于数据是通过累加器传送的,很适合于程序处理或完成判定。例如在输入的情况下,程序可以判定是否要根据刚刚读出的字或字符的值,来读另一个字或字符。

然而,因为每传输一个字符或字至少必须执行一条指令(由于信息必须存在内存中,很可能必须执行几条指令),那么直接控制仅能对那些不必需迅速传输大量信息的外围设备有效。

(2) 数据通道控制

诸如磁盘和磁带机一类的某些外围设备可用于传输大块数据。为了减少所必需的程序开销,这些块是在数据通道控制下传输的。把用于建立数据通道传输的命令汇编在一个累加器中,并在直接程序控制下传输给控制器。因此,数据块是经数据通道在内存和控制器之间自动地直接进行传输的。

数据块的数据通道传输一旦建立起来,并且已用程序启动,就不再需要使用程序操作来完成传输。在进行数据块传输的同时,程序可继续执行其他任务。每当控制器准备好传输字块中的一个字时,就请求直接访问内存。在获准访问时,就传输该字。因为在传输每个字时不必执行几条指令,字块可以按高速率传输,在某些情况下可以按高于每秒一百万字的速率进行。

由于经数据通道实际传输字时,可能与正在执行的程序指令相矛盾,在传输每个字时,程序暂停。除程序执行所需时间加长外,对于程序设计人员来说此暂停是透明的。

1.1.4 程序中断设备

当直接程序控制下或数据通道控制下传输信息时,程序必须能够判定何时完成传输,以便能够启动一个新传输或继续进行一个依赖于刚完成的传输任务。外围设备具有状态标志为

程序提供这种必需的信息。I/O 指令系统允许程序检验这些标志的状态,并根据检验结果进行判断。然而,这些状态检验是费时间的,因此,为了避免连续地进行这种检验,DG 公司的所有计算机都加了一个程序中断设备。

程序中断设备可提供外围设备一种方便的方法通知处理机为要求的程序服务。这是通过使外围设备根据优先级而中断正常的程序流来实现的。当外围设备完成了一种操作,或遇到了要求处理机干预的情况时,就可以请求处理机程序中断。程序可中断进程中的程序而允许这一请求,并保存发生中断处的地址,将控制转移到中断处理。中断处理程序能够辨别哪些外围设备要求服务并将控制转移到对该外围设备的服务程序。在为该外围设备服务以后,程序可以将系统恢复到发生中断时的状态。

对于需要向许多设备大量输入/输出的计算机系统,可以建立多级优先中断的结构,最多达十六级。设立这种结构可以向那些对计算机系统的有效操作具有关键性作用的设备提供快速服务。对于不太重要的设备也可以尽可能有效地提供服务。优先级中断结构与其余的程序中断设备一样,也是在程序的直接控制之下。

1.1.5 摘 要

在介绍 I/O 程序设计的以下章节中将详细地说明上述的概念。对于完成直接程序控制传输所需要的指令,是按照其与控制器和 CPU 的交互作用来讨论的。程序中断设备的结构原理,是与安排优先级结构的方法一起叙述的。在讨论了经数据通道完成数据块传输方法之后,再讨论定时概念,它是为设计一个处理 I/O 的有效系统所必需的。最后,将叙述在有关 I/O 程序设计的前言中所讨论的方法。

§ 1.2 I/O 指令系统

1.2.1 引 言

在计算机与各种外围设备中间的信息传输,是由程序控制的,此程序具有八条指令,形成了 I/O 指令系统。这些指令可以使程序与外围设备控制器通讯,并控制程序中断设备。本手册仅包括用于此目的的那些 I/O 指令,还有用于特殊处理功能的附加 I/O 指令;选择指令将在 ECLIPSETM 和 NOVA[®] 系统计算机的程序员参考手册中详细说明。

特殊的 I/O 指令的效果必须取决于所寻址的外围设备,但是 I/O 指令(装入寄存器,读出寄存器,发出控制信号,和检验标志)所提供的一般功能与所有外围设备相同;不同的外围设备仅能以不同的方法使用现有的功能。为了弄清 I/O 指令所完成的一般功能,以及弄清外围设备控制器如何典型地使用这些功能,就需要对外围设备控制器的结构有所了解。

1.2.2 典型的控制器

从程序的观点来看, 外围设备控制器的操作就象是信息寄存器、控制寄存器和状态标志的集合, 以此建立起通讯。使用这些寄存器和标志, 程序可以在计算机和设备中间传送数据, 并监控设备的操作。信息寄存器可以当作计算机和设备之间传输信息的暂存器。在输入操作时, 设备可以将信息存入控制器的寄存器中, 然后计算机再读出寄存器的内容。在输出操作时, 计算机将信息存在控制器的寄存器中, 必要时设备可以存取信息。控制寄存器是由程序装入的, 并用控制寄存器来控制设备的操作。由设备设置状态标志, 以便反映其当前的操作情况。程序通过使用 I/O 指令可以检查这些状态标志并可改变某些状态标志, 以及改变外围设备的操作状态。

寄存器和标志之间的区别, 通常是一个信息的内容。标志包括一个“一位的信息”, 而寄存器却是由若干位组成。在寄存器中用于传输一“块”信息的相邻的一组位, 称为“字段”。例如, 在一个磁带控制器的寄存器中, 13至15位作为一个控制段一起操作, 可从子系统的八台磁带机中选择一台。

下面的几段仅叙述典型控制器的几个基本部分。外围设备使用程序中断设备或数据通道所需的附加结构, 将在叙述那些设备的章节中加以讨论。后面所提到的只是指典型的控制器工作; 控制器如系根据其控制的专用设备而专门设计的, 则不完全适合这里所提出的典型。

(1) 信息寄存器

控制器通常包括有一个或几个信息寄存器。使用各 I/O 指令, 程序可将数据和控制信息从处理机的累加器送到这些寄存器中, 并可将数据和状态信息从寄存器读到累加器中。这里所讨论的三种信息, 即数据、控制和状态, 引出了三种主要的信息寄存器, 将在下面叙述。

1) 数据寄存器

数据寄存器(或数据缓冲器)是在控制器中, 用于存储在设备和计算机之间传输的数据。因为计算机与设备的操作速度一般是不同的, 所以需要这种缓冲器。由于几乎所有的外围设备操作都包括了在计算机与设备之间的数据的一个字的传输或者数据中字的一部分传输, 故而几乎所有的外围设备控制器都包括一个数据缓冲器。如果外围设备是在程序直接控制下传输数据, 则可由程序直接存取数据缓冲器。使用 I/O 指令实现在控制器的寄存器与中央处理机的累加器中间的数据传输。如果外围设备是在数据通道控制下进行数据传输, 那么就可以在控制器的寄存器与存储器中间直接进行数据传输。使用数据通道时, 控制器中的数据缓冲器不需用, 通常也不用程序存取。

2) 控制寄存器

控制寄存器允许程序向控制器提供设备操作所需的信息, 例如磁盘驱动器或磁带机号, 数据块大小, 以及命令说明。一个控制信息单位就叫做一个“控制参数”。控制参数的典型用法是可以允许程序由子系统的若干台外围设备中选择一台, 选择要完成的操作, 以及选择控制器中的标志和计数器的初始值。程序给控制器指定控制参数, 它是使用一条 I/O 指令, 其中将需要的参数编码成在传递时使用的累加器的相应字段。

3) 状态寄存器

使用状态寄存器可向程序指示出外围设备的详细状态。状态寄存器主要包括状态标志,也可包括控制参数。状态寄存器中包含的控制参数,通常在外围设备操作过程中是要改变的,并且对于程序来说很重要的一点就是必须要检查外围设备操作的过程。例如,以单一操作方式传输磁盘上连续扇区的信息的程序,能够读出操作过程中当前扇区地址和扇区计数,以便决定操作还差多少才能完成。状态标志由控制器设置,以便指出错误情况,或通知计算机有关外围设备的基本状态。

将控制器的寄存器分成上述三类,只是一种普通的分类法。寄存器可以包括不止一种信息。最常见的情况是:在用程序输入时寄存器作为控制寄存器使用,而在用程序读出时寄存器又作为状态寄存器使用。前几段中所述的磁盘地址/扇区计数器乃是这种控制和状态寄存器组合的一个例子。

(2) “忙” (BUSY) 和“完成” (DONE) 标志

“忙”和“完成”标志是控制器中的二个基本标志,可以具有双重作用。它们在指出外围设备的基本状态的同时,又可用程序检验,以确定那种状态。另外,程序可处理这些标志,以便控制外围设备的操作。为了使外围设备进行操作,程序把“忙”标志置1。在操作期间“忙”标志保持此状态,表示外围设备正在使用中,而且不应受到程序的干扰。当外围设备完成了其操作时,控制器将“忙”标志置0,“完成”标志置1,来表示此状态。将“完成”标志置1,能用于发出程序中断。然而,程序是否发生中断,取决于中断设备的状态。不管中断设备处于何种状态,在“完成”标志置1之前,外围设备都不会发出中断请求。所以“完成”标志置1的定义就是“发出程序中断请求”。由此可见,程序既可以将“完成”标志置0并将“忙”标志置1而启动下一个操作,又可以将二个标志都置0而使外围设备闲置(清零)。

(3) 其它状态标志

对于比较简单的外围设备,只是“忙”和“完成”标志就可以提供足够的状态信息,就可使程序为外围设备服务。然而,更复杂的外围设备一般则需要附加若干状态标志以对程序指定更完备的内部操作条件。这些附加状态标志与“忙”和“完成”标志之间的区别,在于程序对这些标志的检验方法。程序可以用一条I/O指令直接检验“忙”和“完成”标志,但是对任何其它状态标志的检验,都要求首先将其值从状态寄存器读入累加器。每一个状态标志是由控制器对状态寄存器中具有的16个位的位置之一来指定的。这样,程序就可以在读出状态字之后对其进行必要的检验。

错误标志 (ERROR FLAG):

表示外围设备操作中的错误和故障的状态标志,称谓“错误标志”。按照错误标志置位后对外围设备操作的影响,可以分成二种错误标志。第一种错误标志,即被动的错误标志,仅仅是在操作过程中发生有关的错误时由控制器来设置的。对于这类错误,不立即向程序指出,允许继续完成操作。第二种错误标志,即主动的错误标志,当程序企图启动一个不允许的操作时则由控制器设置之。在这种情况下,决不会开始操作,而“完成”标志立即置1并通知程序。这种错误标志用于防止发生不可能发现的严重错误。在上述两种情况下,当

程序被通知；外围设备是在“完成”的情况下，它必须回答是否错误。在假定新起始的操作圆满完成之前仍需检验适当的错误标志即可。

例如，在许多状态标志中，磁带机的控制器包括有错误标志，指出奇偶错误和非法操作。在读操作过程中，当读出带有不正确的奇偶检验位的字符时，则奇偶错误标志置1。不立即向程序通知错误，允许将读操作完成。当程序必须检查错误时，可以在操作完成时检查奇偶错误。此时，可以采取适当的行动，例如试图再一次读出误读的磁带部分，或者用控制台终端打印出错误信息。另一方面，如要发生非法操作就置“非法”标志，可停止启动操作。控制器立即将“完成”标志和“非法”标志均置1，以告知程序。磁带机的非法操作包括在“写保护”的磁带上写以及磁带在带始标处退带。

1.2.3 指令格式

I/O 指令的一般格式如下：

0	1	1	AC	操作码			控制		设备代码						
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

第0—2位是011，以此作为 I/O 指令的标记；第3—4位指定累加器；第5—7位包含操作码；第8—9位指定标志控制功能或检验条件；第10—15位指定设备代码。

(1) 设备代码字段

I/O指令中第10—15位可选择要响应指令的外围设备。指令格式可以有64种设备码，编号由0至77。在各种计算机中，设备代码0不指派给任何外围设备，设备码77用于实现许多特定的处理机功能，例如，读控制台开关和控制程序中断设备。根据不同的计算机为处理机选件和特点保留若干的特定设备码。其余设备代码可供基准外围设备使用。DG公司还将许多设备代码指定给标准的外围设备，而汇编程序可以识别这些设备码的助记符。标准设备代码用途及其有关助记符表列于附录A。

(2) 标志控制字段

“忙”标志和“完成”标志既可以用 I/O指令的第8位和第9位所规定的控制功能实现操

表 1.1

F 字 段	命 令	助记符	控 制 功 能
00	(无)	(略)	无
01	起 动	S	将“忙”标志置1，“完成”标志置0，而起动外设
10	消 除	C	将“忙”和“完成”标志均置0，而使外设闲置
11	脉 冲	P	对控制器加脉冲以达到特殊的效果，如产生效果则因外设而异

作或检测，又可以是其指定的测试条件。在那些允许标志操作的命令中，第8位和第9位可以被当作F字段。标志控制命令以及其有关助记符和典型完成的位组态、功能（见表1.1）。

在允许测试的I/O指令中，第8位和第9位称为T字段。其所选择的位组态，助记符，和检验条件如下（见表1.2）：

表 1.2

T 字 段	助 记 符	如遇以下情况则跳过下条指令
00	BN	“忙”标志为1（非零）
01	BZ	“忙”标志为0（零）
10	DN	“完成”标志为1（非零）
11	DZ	“完成”标志为0（零）

I/O指令系统的二个重要特点，是由于标志控制字段的性质而产生的。首先，因为标志控制字段与操作码字段是分开的，一条I/O指令可在控制器与计算机中间传输信息，同时可控制外围设备的操作。其次，使用标志控制字段作为T字段，可以允许直接测试一条指令中控制的“忙”或“完成”标志，这样可以用程序迅速地完成了对外围设备基本状态的判断。

（3）操作代码字段

3位的操作码字段可选择8条I/O指令之一。其中有二条指令，不指定信息传送；而是第8位、第9位可以指定上述的一个控制功能或标志检测条件。其余六条指令包括在计算机与指定外围设备控制器中间的信息传送，并且还指定在完成了信息传送后将要完成的控制功能。所以，程序可对任一控制器中的寄存器进行存取，最多可达六个寄存器。六个寄存器中最多有三个可作为输出寄存器，可以用程序装入数据或者控制信息。其余三个是输入寄存器，程序可由其中读出数据或状态信息。完全可能的而且事实上非常普遍的，是二个不同的I/O指令（一个输入，一个输出）可以访问控制中的同一寄存器。然而，I/O指令的系统实际上决不需要这样；而控制器中的六个不同寄存器都可以被程序存取。

为了以通用的形式给予I/O指令一些名称和助记符，用程序存取的外围设备控制器中的寄存器，是用字母标志来访问的。这三个输入寄存器称谓“A输入缓冲器”，“B输入缓冲器”和“C输入缓冲器”。同样地，三个输出寄存器称谓“A输出缓冲器”，“B输出缓冲器”和“C输出缓冲器”。这样，例如，要从外围设备控制的A输入缓冲器读出数据，就要向该外围设备发出“DATA IN A”指令，其助记符为DIA。

8个操作码，与其有关助记符，以及指定的指令如下表所列（表1.3）：