

虚拟图像显示器的设计与实现

——IRSA-2型遥感图像分析 处理系统的发展之二

李丽 高朋

(中国科学院遥感应用研究所)

1985年12月8日收到

IRSA-2型遥感图像分析处理系统以 ECLIPSE S 140 计算机为主机,主要用户设备为 COMTAL VISION ONE/20 图像显示器。要发展一个完整的计算机图像分析处理系统,必须在完成计算机和图像显示设备的硬件连接、并能在机器语言下正确运行后,将用户设备纳入计算机的操作系统。S140 计算机的 AOS 操作系统本身具有复杂和严密的结构,将用户设备,特别是同时使用程序输入输出及高速直接数据通道的图像显示器纳入 AOS 操作系统时,极易影响 AOS 操作系统的功能和速度。因此,设计和实现能纳入操作系统的完善的用户设备驱动软件,是发展计算机图像分析处理系统的最根本和最重要的环节。

ECLIPSE S 140 计算机与 COMTAL 图像显示器连接 中存在的问题

在 S 140 计算机与图像显示器连接过程中,由于显示设备属于用户设备,因此有专门的设备驱动程序,并且应当纳入操作系统下运行。对于用户设备驱动程序, AOS 操作系统受到如下限制:第一,所有与用户设备有关的程序,变量必须全部放入公共区,一切非系统子程序调用参数传递均无效。这一限制破坏了 FORTRAN 程序的常规编写方法。子程序调用时不能传递变量,使与显示设备有关的应用程序与标准磁盘输入输出程序库无法联接,影响了程序标准化。第二,所有与用户设备有关的程序均必须以常驻内存方式运行。ECLIPSE S 140 计算机 AOS 操作系统,是一种复杂的多用户操作系统,它根据各用户的优先级动态地分配计算机资源。各用户进程按其优先级高低轮流进入计算机,而其它进程则在磁盘中等待。然而,对于用户设备驱动程序,由于它必须常驻内存,操作系统无法将它转移到磁盘。因此,当有设备驱动程序运行时,所有其它用户进程必须等待,直到设备驱动程序运行结束才能继续运行,这样就破坏了 AOS 操作系统的多用户性。另外,用户设备驱动程序不仅不能从计算机转移到磁盘,而且,操作系统也不能将它从磁盘转移到计算机。所以,对所有常驻内存的程序不能利用系统功能 SWAP 或 CHAIN 来传递控制,而必须在 AOS 的命令行解释系统 (CLI) 下,以 PROC 命令运行。这意味着所有与显示设备有关的应用程序将无法纳入图像处理分管系统的多级分支功能清单

(MENU),也就破坏了整个图像分管系统的完整性。第三,用户设备驱动程序必须在超级用户(SUPERUSER MODE)下运行。超级用户是 AOS 操作系统为系统开发和管理人员设计的,它在系统中具有很大的权力,可直接进入任何一个用户。如果让图像处理分管系统的用户进入超级用户,不仅系统有可能遭到破坏,而且亦不利于系统保密。

二、几种解决问题的方法

为了解决上述存在的问题,我们在图像显示器驱动程序和应用程序之间加了一个软件接口,接口程序常驻内存,以批处理方式在后台运行。它能接收其它进程的数据,并将其写入图像显示设备,或者从显示器读出数据并发给请求的进程。应用程序通过软件接口完成对显示器的读写访问,而不是直接调用显示器的驱动程序。这时,其它所有与图像显示器有关的应用程序对显示器的读写,均由对物理设备的读写变成对虚拟 COMTAL 图像显示器软件接口的读写。脱离了物理设备,上述问题就可得到较完满的解决了。

实现软件接口的关键问题是接口进程与应用进程之间的数据传递。为了解决这一问题,先后尝试了三种不同的解决方法:利用磁盘文件作数据传递媒介;利用内存共享区传递数据以及进程通讯技术。

(一) 磁盘文件数据传递

这种方法是利用磁盘文件作数据传递的媒介。应用程序首先打开一个磁盘文件,将一行图像数据和相应的控制信息写入该文件,然后关闭磁盘文件。接口程序打开同一个文件,从中读出数据和控制信息,并将数据写入图像显示器。为了避免两个进程同时访问磁盘文件,接口程序中设置了分时控制。由于这种方法每读或写一行数据就需要打开、关闭一次文件,因此,这种方法的速度较慢。

(二) 扩展内存共享区

这种方法是利用扩展内存进行数据传递。应用程序从磁盘中读出图像数据,经分析后放入扩展内存的共享区域。接口进程不停地从该共享区内读出数据,检查标志位。当标志位置位时,表示共享区中已送入新的数据,接口则将其写入图像显示器。这是一种快速的数据传送方法。但是,这里存在着复杂的分时问题,如果分时不当,常驻内存的接口程序将停止应用程序,并将整个操作系统挂起。

(三) 进程通讯

第三种方法为进程通讯(Enter-process Communicatio)方法。在此方法中,接口进程在读写图像显示器之前等待接收进程通讯数据,其中包括四个控制字和一行图像数据。收到数据后,接口进程根据控制字作相应的动作;写一行数据到图像显示器或从图像显示器读出一行数据,并发送给用户进程。用户进程通过 IPC 向接口进程发送控制字和图像数据。数据发出后,用户进程继续向前,准备再一次与接口进程通讯。从磁盘中读下一行数据或提出新的读图像显示器的请求。这种方法不仅速度快,而且不存在进程之间的分

时间问题。

对上述三种方法, 经过反复实验之后, 最终确定采用进程通讯技术完成 ECLIPSE S 140 计算机和 COMTAL 显示设备的软件接口。

三、虚拟图像显示器的设计与实现

接口程序 DISPCOMTAL 采用进程通讯进行数据传递, 速度快, 功能强, 可以很好地完成对图像显示器的各种操作, 包括读写图像刷新存储器、图形存储器 and 功能存储器, 传递各种 COMTAL 图像显示设备命令等。

(一) 进程通讯原理

进程通讯是在多用户操作系统下产生的一种进程之间的数据交换方式。在多用户操作系统中, 各用户的应用程序均作为进程在系统管理下运行, 进程之间保持相应的独立, 一般不发生任何联系, 也无法发生直接联系。不过当需要在进程之间信息交换时, 系统提供了进程通讯的手段。信息通过通讯缓冲区进行交换, 两个用户进程彼此呼应, 有发有收。图 1 表示进程通讯的简单过程。

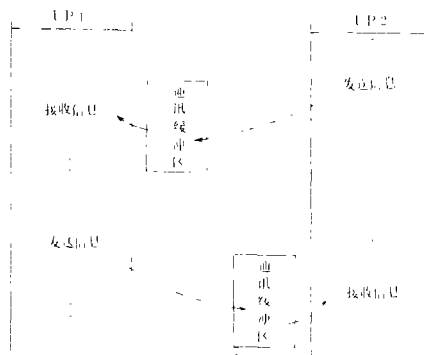


图 1 进程通讯的简单过程

图 1 进程通讯的简单过程

Fig. 1 The Block Diagram of Enter-process Communication

在 AOS 操作系统下, 进程间的具体通讯过程可分为四个步骤:

(1) 定义通讯缓冲区大小, 规定进程的局部端口 (Local Port) 以及其它进程通讯的控制信息。例如: 用户控制字和系统控制字等。

(2) 建立进程通讯端口 (Global Port)。

(3) 寻找接收或发送进程的通讯端口。

(4) 发送或接收通信信息, 并根据系统控制字作相应的处理。例如, 当执行接收操作, 但无其它进程发来信息时, 根据系统控制字, 挂起进程, 等待通讯信息, 或者错误返回。

(二) 数据结构

接口程序 DISPCOMTAL 与其它进程间的通讯每次传递 260 字的数据。整个数据由两部分组成：控制信息和图像数据，其分配格式如图 2 所示。

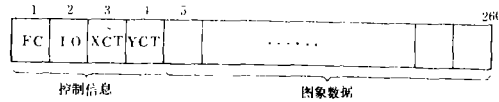


图 2 DISPCOMTAL 传递的控制信息和图像数据结构

Fig. 2 The Control words Transmitted by DISPCOMTAL and Image Data Structure

FC (第一个字): COMTAL 显示器功能码, 当 $FC = 1$ 时, 为图像/图形数据输入/输出。

IO (第二个字): 输入/输出控制码, 当 $IO = 0$ 时, 写显示器操作; 当 $IO = 1$ 时, 读图像显示刷新存储器。

XCT (第三个字)。一次写入图像刷新存储器的像元个数。一般取 $XCT = 512$ 。

YCT (第四个字): 显示器图像行号。设 IMG 和 L 分别为图像新存储器号和在该图像存储器中的行号, 则 $YCT = (IMG - 1) * 2048 + L$ 。

数据区(5-260): 共 256 个字。存放一行图像数据。

(三) 软件接口程序 DISPCOMTAL

DISPCOMTAL 为实现 S 140 计算机和图像显示器之间的软件接口程序。其流程如图 3 所示。

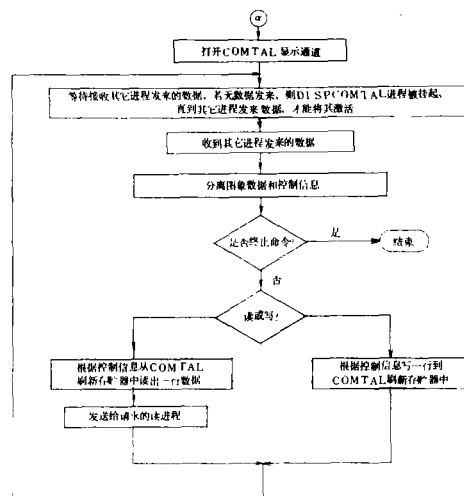


图 3 DISPCOMTAL 程序流程图

图 3 DISPCOMTAL 程序流程图

Fig. 3 The Flow chart of DISPCOMTAL Program

(四) 虚拟图像显示器读写程序 WRCT

子程序 WRCT 为 DISPCOMTAL 的配套程序,专门用于 DISPCOMTAL 的读写。其它各种应用程序通过调用该子程序来完成对软件接口 DISPCOMTAL 的读写。WRCT 程序的流程如图 4 所示。

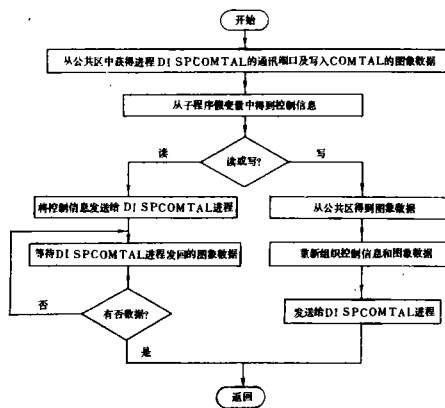


图 4 虚拟显示器读写程序 WRCT 的流程图

Fig. 4 The Flow chart of Virtual Memory Read-write Program WRCT

(五) 磁盘图像显示程序 DISIPC

这里以磁盘图像显示程序 DISIPC 为例来说明应用程序与软件接口 DISPCOMTAL 的通讯方法。DISIPC 程序的流程如图 5 所示。

(六) 进程通讯程序的呼应

这里将以磁盘图像显示程序 DISIPC 为例,说明接口进程与用户进程之间的相互呼应。

1. 接口进程 DISPCOMTAL

- (1) 开辟进程通讯缓冲区,设置进程通讯程序信息包,
- (2) 建立进程通讯端口,
- (3) 接收其它进程发来的通讯信息,
- (4) 向读请求进程回送图像数据。

2. 图像显示应用程序 DISIPC 与 DISPCOMTAL 接口程序的呼应

(1) 开辟自身的通讯缓冲区,设置进程通讯系统子程序所需要的控制信息包,以及建立进程通讯端口。当进程通讯端口名已存在,即已有用户使用图像显示器时,考虑到共享图像显示器,应安排建立第二个或第三个通讯端口。当通讯端口超过三个时,则必须等待,

- (2) 寻找自身的通讯端口号,

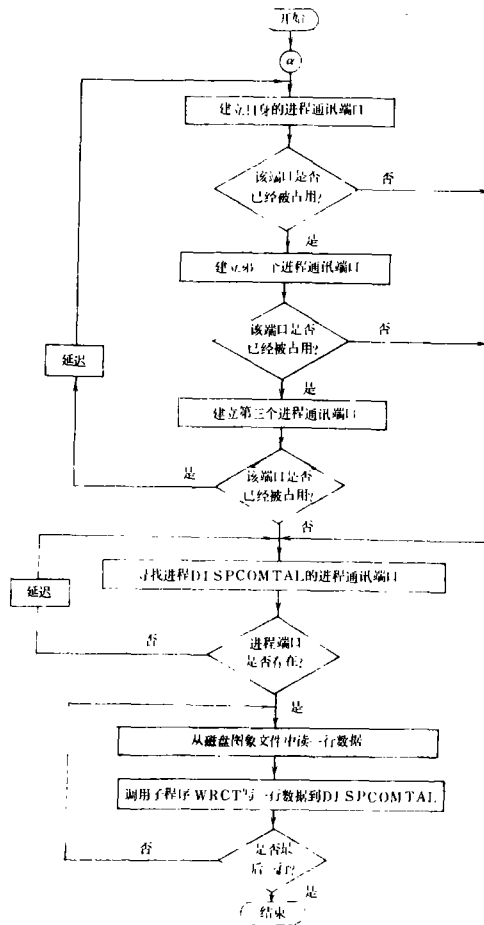


图 5 DISIPC 程序流程图

Fig. 5 The Flow Chart of DISIPC Program

- (3) 寻找接口进程 DISPCOMTAL 的通讯端口名。
 - (4) 向接口进程 DISPCOMTAL 发送数据或请求,
 - (5) 等待接口进程回送图像数据。
- 以上是进程通讯的简单过程。

四、结 束 语

采用进程通讯方式交换数据的软件接口,妥善地解决了前面提到的联结中存在的问题,将显示器作为用户设备纳入了 AOS 操作系统。使显示器的驱动接受图像处理软件系统的管理,工作速度快,在图像数据传输过程中不会阻塞别的用户工作,实现了多用户,使显示器具有了共享性。COMTAL 显示器有多个图像平面,图影平面和功能存储器,通过软件接口 DISPCOMTAL 可以将图像,图影等操作并行进行。即对用户而言,相当于

构成了各自的虚拟图像显示器,因而,进一步提高了设备效率达到充分利用系统资源的目的。此项工作为发展完整实用的 IRSA-2 遥感图像分析处理系统打下了基础。

Implementation of the Virtual Image Display in IRSA-2 Digital Image Processing System Development of the IRSA-2 Digital Image Processing System

Li Li Gao Peng

(Institute of Remote Sensing Application, Academia Sinica)

Abstract

In the article, the software interface of the IRSA-2 Digital Image Processing Subsystem with the AOS operating system of the Data General Eclipse S140 computer is described. Few alternatives have been compared and tested, and the Inter-Process Communication technique has been used. With this technique, the user device is connected into the AOS operating system which founded the base for development of the IRSA-2 Digital Image Processing System. In IRSA-2 system, the image display device can be shared by multiusers by means of utilization of virtual image display technique. Therefore, the computer resources are used efficiently.