

Intel 86/330A 与 VDU-140 联机系统的实现

崔志明

提 要 彩色显示系统在指挥、控制与通讯系统的建设中起着越来越重要的作用。本文介绍利用 Intel 86/330A 与 VDU-140 显示设备组成彩色显示系统, 并着重描述在 iRMX86 操作系统上开展的工作。

目前, 计算机系统在各个领域中应用的深度和广度都在发展, 在指挥、控制与通讯系统中, 用计算机进行实时管理的彩色显示系统得到了广泛应用。

为了在比较短的时间内, 用较少的硬件配置, 建立比较实用的性能较高的显示系统, 我们选择了 Intel 86/330A 与 VDU-140 图形显示设备组成彩色显示系统, 并且力图在不对原系统硬件进行修改的基础上, 通过串行口用软件进行联机通讯。实现方法是把我们的联机软件作为一个作业, 挂入原 iRMX86 实时操作系统中, 这就生成了管理该彩色显示系统的实时操作系统。并且成功地实现了二十多条联机命令和二十多条作图命令, 同时还能够及时保存字符、图形档案等。

由于我们是在对 iRMX86 实时操作系统进行改造和配置的基础上来实现该系统的, 因此, 也就较好地实现了利用高级语言进行高效率作图, 实时处理显示档案。

一、系统结构与联机方法

Intel 86/330A 是一个 16 位的微机系统, CPU 采用 8086, 除主存可达到 1 兆字节外, 还有 35 兆温盘和一个软盘驱动器。iRMX86 操作系统是 86/330A 软件的核心, 它是一个实时多任务的操作系统。在该操作系统的支持下, 可运行 PL/M-86 高级语言。在 PL/M-86 语言中, 可以嵌套使用许多系统调用, 这就大大地方便了系统程序的编制。

VDU-140 图形显示设备是日本 Ai Electronics Corp 研制生产的。它作为终端能与多种微机进行联机, 并且功能较强, 可靠性好, 它的 CRT 采用 14 吋彩色监视器 (具有八种颜色), 分辨率为 640×400 , CPU 为 8088, 通讯是用两个 RS-232-C 接口,

并且可用多种输入手段。

我们把 iSBC534 板插入 86/330A 的机箱中, 作为 86/330A 与 VDU-140 信息传输的媒介。iSBC534 板上有四个使用 8251A 芯片的串行接口, 通过这些接口, 就能容易地组成彩色显示系统 (见图 1)。

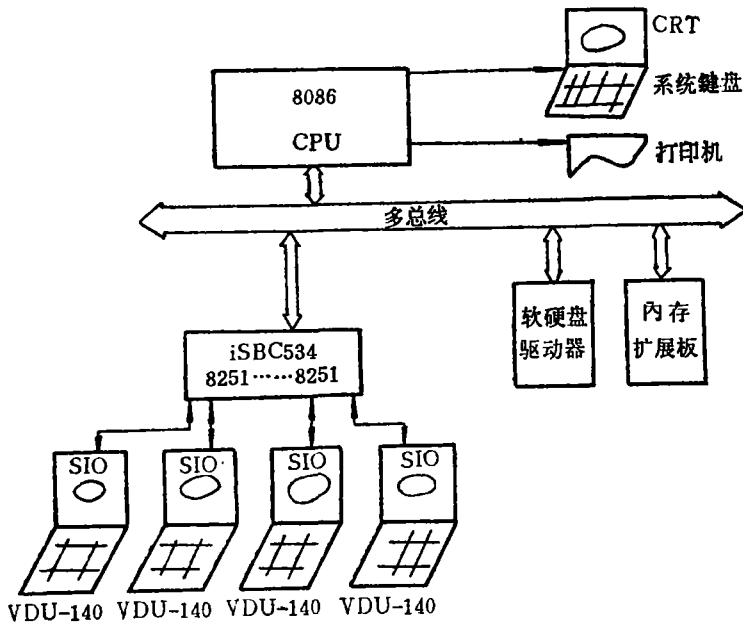


图 1 彩色显示系统结构图

534 板通过多总线与 86/30 板相连, VDU-140 通过标准电缆, 按 RS-232-C 规程与 534 板相接。VDU-140 上的串行口使用 Z-80SIO 芯片, 这就为编程提供了很好的条件。由于 Z-80SIO 是可编程的双通道器件, 因此我们选择 2 通道来与 534 板上的 8251 A 控制的接口进行数据通讯, 并作以下处理:

1. 利用异步方式将数据按规定格式相互通讯。
2. 采取全双工方式, 并满足偶校验要求。
3. 波特率定为 9600bps。

除此之外, 还利用 iSBC534 板的 8259A 芯片来控制四个通道的所有中断信号, 以便 iRMX86 操作系统进行管理并作实时处理。

二、软件实现

联机软件实现的关键是对 iRMX86 操作系统进行改造和配置, 以便让它来实时管理整个彩色显示系统, 概括地说: 就是用 PL/M-86 语言编写一个用户作业 (User Job) 并把它挂到 iRMX86 的根作业 (Root Job) 上去, 让该作业作为实时操作系统的一部分, 以便核心程序对它进行管理和调度。

在 iRMX86 操作系统中, 诸作业形成一棵作业树, 各个子系统可看成相应的作业, 除了根作业外, 树中的每个作业都从母体得到资源, 而作业由任务和任务所需要的资源

组成。它的执行状态可以是运行，就绪，睡眠，挂起或睡眠挂起状态之一，每个任务都有一个优先级，核心程序根据优先级来对任务进行调度。各个任务之间的通信可通过信箱 (Mailbox) 或信号量 (Semaphore) 进行。

我们写的用户作业的结构以及各个任务之间的关系见图 2，它共有八个 I/O 任务和其它 5 个任务及其所需的资源组成。由于 iSBC 534 板包括四个 8521A USART，每个 USART 有两个中断，一个为接收有效，一个为发送有效，这些中断都是由 iSBC534 板的 8259 A 进行控制的，核心程序根据 8259 A 寄存器的内容确定 8 个中断源中究竟那个发中断，然后把一个“元”送给正等待的相应 I/O 任务。

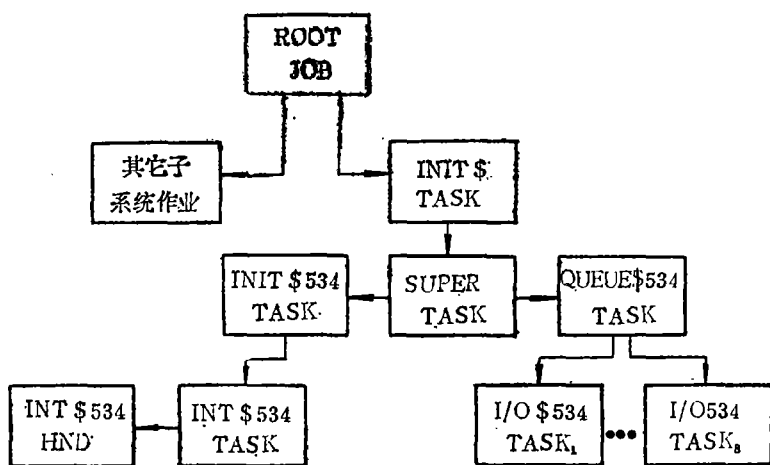


图 2 实时系统作业结构

下面简要地介绍一下各个任务的工作过程：

1. INT \$ TASK

这是一个总初始化任务。

- 调用 CREATE \$ MAILBOX 建立一批信箱。
- 调用 CREATE \$ TASK 建立一个 (SUPER \$ TASK) 任务。
- 调用 SUSPEND \$ TASK 挂起自身。

2. SUPER \$ TASK

这个管理任务当 INIT \$ TASK 挂起后，就从就绪状态变成运行状态。

- 调用 CREATE \$ TASK 建立初始化 534 任务 (INIT \$ 534 \$ TASK)。
- 调用 CREATE \$ TASK 建立队列 534 任务 (QUEUE \$ 534 \$ TASK)。
- 调用 RECEIVE \$ MESSAGE 从信箱中请求信件。

3. INIT \$ 534 \$ TASK

- 调用 CREATE \$ SEGMENT 建立一个数据段。
- 调用 CREATE \$ MAILBOX 建立一个请求信箱。
- 调用 CREATE \$ MAILBOX 建立一个响应信箱。
- 调用 CREATE \$ TASK 建立一个中断任务 (INT \$ 534 \$ TASK)。
- 调用 CREATE \$ SEMAPHORE 建立八个信号量。

- 调用初始化 534 板硬件的子程序。
- 返回。

4. INT \$ 534 \$ TASK

这是一个中断任务。

- 调用 SET \$ INTERRUPT 设置一个中断处理程序 (INT \$ 534 \$ HND)。

100 P 4: • 调用 WAIT \$ INTERRUPT 等待中断处理程序的信号。

- 查询 8259 A 判何中断。
- 调用 SEND \$ UNITS 向相应信号量送一个元。
- 无条件转 100 P 4。

5. INT \$ 534 \$ HND

这是一个中断处理程序, 当某一个中断产生, 它就开始执行。

- 调用 GET \$ LEVEL 取得中断级。
- 调用 SIGNAL \$ INTERRUPT 向中断任务发中断信号。
- 返回。

6. QUEUE \$ 534 \$ TASK

100 P 6: • 调用 RECEIVE \$ MESSAGE, 请求 SUPER \$ TASK 发来的信件。

- 将信件进行排队。
- 将排头信件中 I/O 数据段进行检验。
- 调用 SEND \$ MESSAGE 将有效数据段送至请求信箱中。
- 无条件转 100 P 6。

7. I/O \$ 534 \$ TASK

这是一个对 534 板进行实际操作的任务。

100 P 7: • 等待从响应信箱中收信。

100 P 8: • 按要求将数据写入缓存。

- 从指定信号量中请求元。
- 修改计数器和指针。
- 判断计数器为 0 否, 是 0 转 100 P 7。
- 不是转 100 P 8。

各任务的程序框图及彼此联系的详细内容在这就不一一赘述了。

我们把这些任务组成一个用户作业, 在调试成功之后, 运用 ICU(……), 将它配置到 iRMX86 的根目录上去。由于该用户作业已成为 iRMX86 操作系统的一部分, 因此利用高级语言进行作图也就能够实现, 并获得较高的效率。

三、系统特点

本系统保留了 86/330 A 系统的特点外, 还具备以下特点:

1. 86/330A 系统键盘作图与 VDU—140 键盘作图方式相同。
2. 按优先级对多台 VDU—140 进行实时管理, 有效利用了 86/330A 的各种资源。

3. 不改变 86/330A 系统的硬件, 保持原系统的完整性。

4. 在 iRMX86 OS 上进行改动, 方便了用户使用。

这一系统的组成, 实际上是用现成的计算机系统与显示设备的联接, 因而减少了处理环节, 缩短了研制周期, 提高了系统的灵活性, 同时对 iRMX86 改造和配置进行实践, 有利于我们对该实时操作系统进一步开展工作。

The Implementation of an On-line System Based on Intel 86/330A and VDU-140

Cui Zhiming

Abstract

Colour Display Systems play an ever-increasing important role in the development of C³ (command, control and communications) systems. In the paper we discuss a practical colour display system with Intel 86/330A system as the host microcomputer and the VDU-140s terminals. The discussion centres on the work in the development and the configuration of iRMX86 operating system.