

转轴编码器构成的智能仪器混合矩阵面板的实现

罗明, 余立民, 倪明, 柴小丽

(华东计算技术研究所嵌入系统部, 上海 200233)

摘要: 介绍转轴编码器的原理, 从而提出面板中按键和转轴编码器的混合矩阵连接方案。它具有电路简单、I/O 资源占用少、软件实现容易的特点。通过定时中断扫描、基于向量的数据处理、基于 FIFO 的数据存取、基于时间记录的转轴编码器识别、软件去抖以及 PS/2 协议的软件实现等技术手段, 实现该例的 PS/2 接口面板, 其具有低成本、性能可靠稳定、响应及时、字符可加速输出等优点, 并在高端智能仪器中得到很好的应用。

关键词: 转轴编码器; 定时中断扫描; 向量; FIFO 堆栈; PS/2 接口

Realization of Matrix Panel of Intelligent Instrument Using Rotating Shaft Encoder

LUO Ming, YU Limin, NI Ming, CHAI Xiaoli

(Embedded System Dept., East China Institute of Computer Technology, Shanghai 200233)

【Abstract】 A matrix connection of the panel using buttons and rotating shaft encoders is developed after the principle of rotating shaft encoders is introduced. This type of connection is characteristic of simplified circuit, less I/O resource exhausting, ease of soft realization. Then, with the help of efficient soft method, for example, timing-interrupt scanning, vector-based data processing, FIFO-based data accessing, time-recording-based knob recognizing, soft debounce processing and soft realizing of PS/2 protocol, a PS/2 interfaced panel is realized, which is applied in advanced intelligent instrument with the strong points of low cost, high performance, in-time responding, speed-up output.

【Key words】 Rotating shaft encoder; Timing-interrupt scanning; Vector; FIFO stack; PS/2 interface

高性能的面板是智能仪器的人机接口的重要组成部分。它的功能以及性能直接影响智能仪器人机交互的友好性。当前通用计算机的 84/104 键盘就是一种人们都很熟悉的人机接口部件。但这种键盘只有按键, 没有转轴编码器, 不能很好地适应智能仪器的人机交互的人性化要求。转轴编码器是一种新型的用于电子仪器调节的器件。它的外形与普通的转轴电位器很相似, 它通过旋转的方向控制参数值的增加或减少, 并通过转动的角度来控制增减量。在如今的数字化、智能化仪器中, 它大大提高了人机交互的人性化程度。

1 转轴编码器的原理

本例所用的转轴编码器是机械式双脉冲转轴编码器, 它有 3 个脚 A、B、C, 在内部结构上等效于在 AC、BC 间各有一个开关(如图 1 所示)。

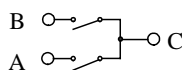


图 1 转轴编码器的原理性电路

转轴转动一周有若干个停顿点(本例中转轴编码器的停顿点数为 24), 当它处于停顿点时, 3 个引脚之间互不导通。当它从一个停顿点旋转到下一个停顿点的时候, AC、BC 就会各导通一次, 但是其导通的先后顺序与其转动的方向有关。转轴编码器的引脚导通时序关系如图 2 所示。

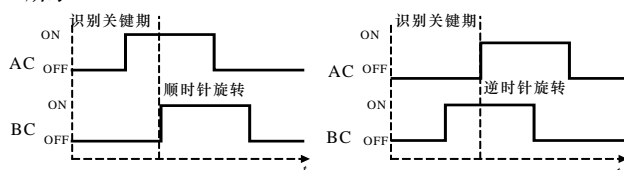


图 2 转轴编码器的引脚导通时序关系

2 硬件连接方案

2.1 几种硬件连接方案的对比分析

在详细讲述本文实现的方案之前, 首先简要地对几种方案作一下对比分析(见表 1)。

表 1 混合面板的几种实现方案的对比分析

No.	实现方法	优点	缺点
1	接口实现、按键扫描和转轴编码器识别各用一个 MCU 完成, 局部总线互连。按键常规扫描, 转轴编码器动作引起外部中断	功能划分明确	电路复杂, 需中断判决电路和较多 MCU, 不易实现字符输出加速
2	CPLD 识别转轴编码器动作, 并将其转化成开关的通断动作。然后用常规的矩阵扫描处理之	思想简单	响应不及时, 易“漏”动作, 实现字符输出加速方面先天不足, 效果不佳
3	高速时钟驱动 CPLD 对转轴编码器进行扫描, CPLD 识别其动作和动作持续时间, 生成结果, 并向 MCU 发送读数据中断请求; MCU 在读数据并分析, 输出字符	易实现, 响应及时, 字符输出可加速	电路稍显复杂, 成本偏高
4	采用按键、转轴编码器混合矩阵连接, MCU 定时扫描矩阵中按键或转轴编码器状态, 并判断识别之	易实现, 响应及时, 字符输出可加速, 电路简单, 成本低廉, I/O 占用较少	

作者简介: 罗明(1981—), 男, 硕士生, 主研方向: 计算机系统结构; 余立民, 工程师; 倪明, 研究员; 柴小丽, 高工

收稿日期: 2005-11-07 **E-mail:** swry_luoming@163.com

2.2 混合矩阵连接方式

笔者曾尝试过表 1 中的前 3 种方案,但它们在某些方面都不及本文提出的方案。

经多次分析和实践,本文最终提出并实现了第 4 种简单有效的硬件连接方式,即混合矩阵连接方式。该连接方式硬件电路结构简单(基本上,除 MCU 和混合矩阵以外,没有其它的电路),成本低,I/O 资源消耗少,软件实现容易,同时面板还实现了字符可加速输出。它克服了其它 3 种方案的所有缺点。

为方便理解,下面先以 8×4 按键矩阵为例分析一下传统键盘的矩阵连接(如图 3(a)),继而提出按键和转轴编码器混合矩阵连接。从图 3(a)中,很容易看出虚线圈内的部分与图 1 中的转轴编码器的原理电路非常相似。类似地,我们也很顺利地得到图 3(b)中的混合矩阵连接。

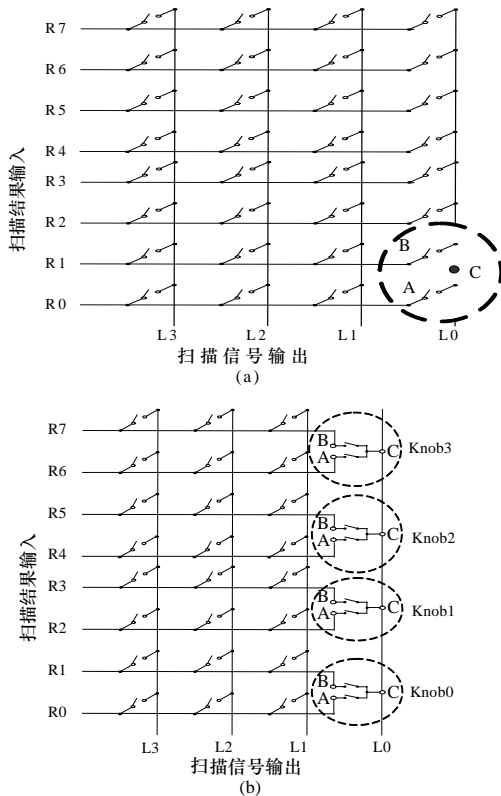


图 3 矩阵方式的硬件原理性连接

当然,实际的按键和转轴编码器数量可以更多,但需要注意的是:为方便 MCU 的软件处理,在混合矩阵连接中,按键与转轴编码器的连接不要在同一列中,而且行的数量最好不要超过 8 (为方便软件采用向量处理,建议采用 8 行的连接方式)。前者主要与软件处理的方式有关,后者主要与所选用的 MCU 的字长有关。

2.3 系统连接

面板上采用的 MCU 是 ATMEL 公司一款高性价比 8 位单片机 AT89C51。考虑到在智能仪器的技术研发阶段经常需要用到标准 PC 机键盘进行软硬件的各种调试或者测试,这里还用简单的电路构成了一个 PS/2 主、从键盘复用单元(主键盘口是双向的,从键盘口是单向的)。这样,仪器在正常运行时,可用面板上提供的按键和转轴编码器进行控制,而在技术研发阶段,可用标准 PC 机键盘通过从 PS/2 口输入各种信息。本文实现的面板的硬件原理性连接如图 4 所示。

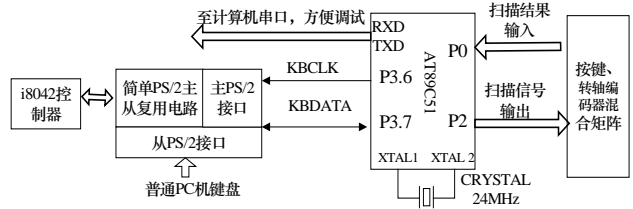


图 4 智能仪器的面板硬件原理性连接

3 软件实现

3.1 软件实现方式

嵌入式系统的软件实现往往是由系统的功能决定,与整个硬件系统相互配合的。本文实现的智能仪器面板,要求具有性能可靠稳定,响应及时。同时,它也要求具备输出可加速,即按照正常速度转动时,转轴编码器每转动一格(从一个停顿点旋转到相邻的下一个停顿点),面板就输出一个字符;转轴编码器转动的速度越快,每转动一格面板所输出的字符个数就越多。这个特点必须要求软件能够记录相关的时间信息。为此,本例中采用“主程序+定时中断服务程序”的软件实现方式,语言采用 C 语言。

3.2 主程序

主程序主要是做一些初始化的工作,同时负责软件实现 PS/2 接口,以便与仪器的主机(以下简称“主机”)的 i8042 控制器用 PS/2 协议进行双向通信。更重要的,主程序还要翻译 FIFO 队列中的有效信息字节,并将翻译结果发送给主机,同时也接收来自主机的命令,并处理这些命令。为方便理解,下面说明几个即将用到的名词或者变量的含义:

- 键值定义矩阵表(以下简称“键值表”):映射按键、转轴编码器的位置与键值的对应关系。
- FIFO: 以双字节为单位的先进先出堆栈,存储信息字节(第 2 个字节的每 bit 记录对应键是否有通断的状态变化。第 1 个字节记录键状态变化方向、扫描列位置、重复次数)。
- 状态行向量: 记录某一扫描列中键的导通状态的字节向量,每 bit 代表一个键。
- 历史状态记录矩阵表(以下简称“历史状态表”): 记录上一次扫描时键的状态的矩阵表,它由 N 个历史状态行向量组成,即一个字节型的数组 LastState[N],针对图 3 来说,N 就等于 4。
- CurrentState[N]: 记录当前的键状态的矩阵表。
- SpeedUpTimeFlagLine: Time 记录最近转动的转轴编码器的 AC 或 BC 导通的时间,Flag 记录 AC 或 BC 导通是否超时,Line 记录需要加速的转轴编码器所在的扫描列位置。
- SpeedUpInfo: 与 SpeedUpTimeFlagLine 配合,记录扫描列位置上转轴编码器的动作。
- uChar: unsigned char。
- ButtonFirstLine: 矩阵中按键的起始扫描列位置。

下面是主程序的主流程,如图 5 所示。

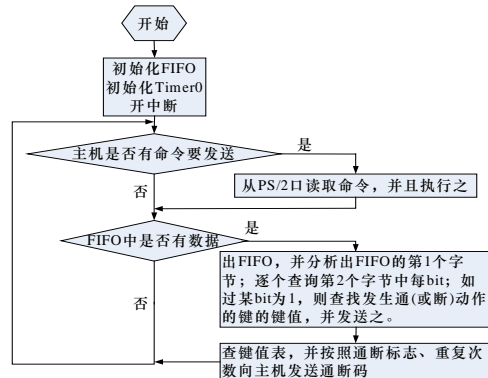


图 5 主程序的流程

3.3 中断服务程序

本例关键就在于矩阵的扫描识别，这些都在定时中断扫描程序中完成。所谓定时中断扫描，即由 MCU 内部定时器每隔一定时间触发一次中断。中断服务程序负责完成矩阵的扫描以及扫描结果的处理工作。需要说明的是，中断的时间间隔需选得比较恰当：既不能太短，否则太频繁的中断会影响主程序的处理时间，甚至会发生多重中断的情况；也不能太长，否则响应不及时，“漏”响应，也不利于时间记录。

为减少中断服务程序的执行时间，本文采用了基于向量的数据处理方式，向量的位长与 MCU 的字长一致。在正常的转动速度下，转轴编码器的 AC、BC 的导通时间非常短（大约为几个毫秒），比按键导通的时间短得多，且按键无需加速处理。因此，为减少中断服务程序的平均执行时间，可以定时器的定时间隔为基准时间，通过变量计时来控制按键的扫描处理的时间间隔与转轴编码器的扫描处理的时间间隔不同。显然，对时间要求不高的按键扫描来说，其扫描的时间间隔完全可以与正常按键去抖动的的时间一致，大约为 20ms~40ms。而转轴编码器的扫描时间间隔应该小于 1ms。二者的处理可比较独立。

为方便讲述，下面对按键和转轴编码器的处理流程分别进行描述。

3.3.1 转轴编码器的扫描

图 6 描述了转轴编码器的处理流程。

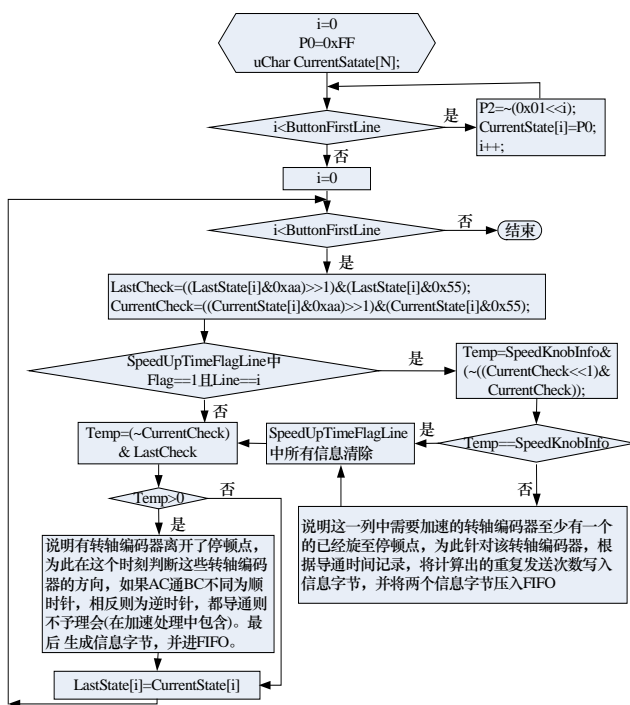


图 6 转轴编码器的中断扫描程序流程

根据图 6 的流程，可知识别转轴编码器的关键时期就如同图 2 所标明的的那样，在 AC、BC 相继导通的时间间隙内。每当转轴编码器从一个停顿点转到 AC 或者 BC 中有一个导通的时候，中断服务程序在下次中断时确认转轴编码器是否仍然处于导通状态，如果确认仍然导通，则开始记录其导通的时间，直到转轴编码器转到下一个停顿点，定时中断程序首先判断该转轴编码器导通时间是否超时，然后按照预先设

定好的字符输出加速等级生成两个信息字节，并送入 FIFO 队列中。为区分转轴编码器的转动的方向，在软件处理中赋予转轴编码器两个键值，分别对应转轴编码器的左、右两个旋转方向。字符输出的个数与转过的停顿点的次数呈正向相关性。为了识别的准确性，两个方向上的识别最好对等，否则有可能出现误识别。

3.3.2 按键的扫描

按键的处理，相对比较容易一些，也有很多资料讲解纯按键构成的矩阵键盘的扫描，尽管采用的不是定时中断扫描，但基本原理差不多。为突出本文重点，这里不再用流程图的形式说明，只作简单的文字说明：每当扫描程序检测到按键从断开状态变成导通状态时，系统会在下一次扫描时确认上次该按键是否仍然处于导通状态。如果仍然导通则生成信息字节，并将信息字节压入 FIFO。如果系统对按键有连击的要求，则在后面连续的若干次扫描中，检测该按键是否仍然导通，若仍然导通，则继续生成关于该按键的信息字节。一旦检测到按键已经断开，则后面不再对该键进行连击处理。

值得说明的是，信息字节的生成和按键的处理也都是基于向量的。按键和转轴编码器的去抖动延时实际上分别是它们的连续两次扫描的间隔。

4 讨论与总结

本文实现的面板，只记录转轴编码器代表的参数的相对改变程度，不记录其绝对改变程度。参数变化的步长由智能仪器的上层应用软件确定，变化的速度与转轴的转动速度、字符输出加速等级、转轴编码器的停顿点数目呈正向相关性。

但是，随着转轴编码器的停顿点数目的增多，同样旋转速度下转过两个停顿点之间的时间间隔就会越短。为保证能够准确、无遗漏地识别转轴编码器的每一次动作，定时中断的时间间隔也要相应地越短。然而，为系统能够正常稳定的工作，中断服务程序的执行时间在整个系统执行时间中所占的比例不能太大。因此，除采用向量处理方式和以变量计时区别两种扫描的时间间隔的方法以外，还一定以提高晶体的频率为代价。所以，实际情况下，开发者应根据 MCU 的频率使用要求以及矩阵的规模，来权衡选择合适的转轴编码器、晶体频率以及中断的时间间隔。

通过具体实践表明，本文所实现的智能仪器面板，能够很好地应用于高端智能仪器上。目前它不仅运用在高性能示波器上，而且还应用在其它测试分析仪上。实践结果表明，它具有结构简单，成本低廉，性能稳定的特点，且扩展容易。

参考文献

- 1 Brey B B. The Intel Microprocessor—Architecture, Programming and Interfacing (Fifth Edition)[M]. Beijing: Publishing House of Electronics Industry, 2001.
- 2 Chapweske A. The AT-PS/2 Keyboard Interface [Z]. <http://www.Computer-Engineering.org>.
- 3 Chapweske A. Keyboard Scan Codes[Z]. <http://www.Computer-Engineering.org>.
- 4 Chapweske A. PS/2 Mouse/Keyboard Protocol[Z]. <http://www.Computer-Engineering.org>.
- 5 李 军. 51 系列单片机高级实例开发指南[M]. 北京: 北京航空航天大学出版社, 2004.