

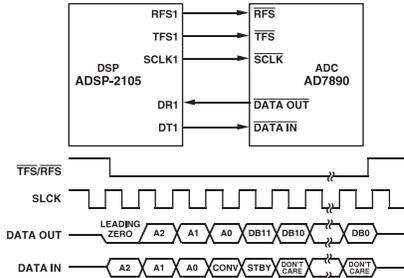
# 应用工程师问答—19

## 与串行转换器接口—1

作者: Eamon Nash

问: 我需要将数据转换器装入一个紧凑的空间,我觉得选择串行接口会有帮助。在选择与使用方面,我需要了解些什么?

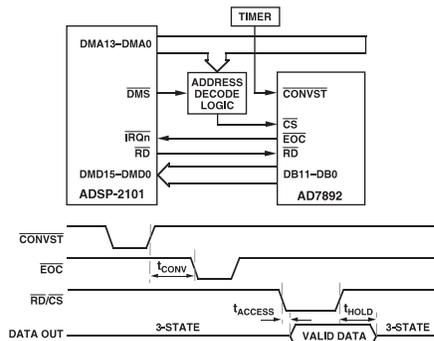
答: 让我们先看看串行接口是怎样工作的,然后将其与并行接口相比较。这样,我们就能更清楚地了解串行数据转换器。



如图所示, AD7890 八通道多路复用 12 位串行 A/D 转换器(ADC)连接到了 ADSP-2105 数字信号处理器(DSP)的串行端口。同时还显示 DSP 用于与 ADC 通信的时序。构成转换结果的 12 比特作为串行数据流通过单线路传输。数据流还包括三个额外的比特位,用以识别 AD7890 的多路复用器目前所选择的输入通道。为了区分串行数据流中的比特位,必须提供一个时钟信号(SCLK),通常由 DSP 提供;不过,有时候 ADC 以输出形式提供该时钟。DSP 通常(但并不总是)在通信开始的一个周期,或如图所示在传输期间(TFS/RFS),提供额外的有效帧脉冲。

在此例中, DSP 的串行端口用于对 ADC 内部的一个 5 位寄存器进行编程。寄存器的比特位控制以下这些功能: 选择要转换的通道, 将器件转入省电模式, 以及开始转换。显而易见, 在这种情况下, 串行接口必须为双向的。

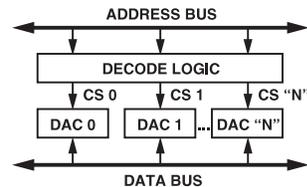
另一方面, 并行 ADC 直接(或可能通过缓冲器)连接与其接口的处理器的数据总线。如图所示, AD7892 与 ADSP-2101 接口。转换完成后, AD7892 向 DSP 发出中断, DSP 通过单次读取 ADC 的解码内存地址作为响应。



串行与并行数据转换器的主要区别在于所需接口线路的数量。从节省空间角度来看,由于减少了器件的引脚数量,串行转换器具有明显的优势。因此,完全有可能将 12 位串行 ADC 或 DAC 置入 8 引脚 DIP 或 SO 封装中。更重要的是,串行接口连接只需更少的 PCB 走线,所以能够节省电路板空间。

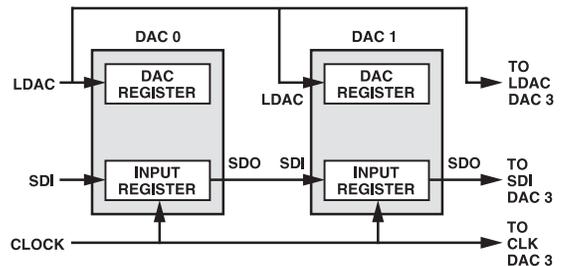
问: 我的数模转换器既要远离中央处理器,同时要彼此远离。最佳实现方法是什么?

答: 首先,您必须决定使用串行 DAC 还是并行 DAC。若使用并行 DAC, 如图所示, 可将每个 DAC 映射到内存映射的 I/O 位置。然后, 通过对适当 I/O 位置执行简单的写指令, 就可对每个 DAC 进行编程。不过, 这种配置也有明显的缺点。它要求并行数据总线以及一些控制信号都连接到所有的远程站点。显然, 仅有两根线缆的串行接口更加经济实惠。



一般来说, 串行转换器不能映射到处理器的内存中。但是, 一些串行 DAC 可连接到处理器的串行 I/O 端口。这样, 该处理器的其他端口可用于产生片选信号, 单独使能 DAC。片选信号要求每个设备到接口都有一条信号线。但是, 处理器上可配置为发送片选信号的信号线数量可能有限。

解决这个问题的一种途径就是使用能以菊花链形式连接在一起的串行 DAC。该图显示如何将多个 DAC 连接到单个 I/O 端口。每个 DAC 都有串行数据输出(SDO)引脚, 可连接至菊花链中下一个 DAC 的串行数据输入(SDI)引脚。LDAC 与 SCLK 并行连接到菊花链中的所有 DAC。由于读入 SDI 的数据最终出现在 SDO (N 个时钟周期之后), 因此单个 I/O 端口可访问多个 DAC。尽管如此, 该端口必须输出一个长的数据流(每个 DAC 的 N 位乘以链上的器件数量)。该配置最大的优势在于无需器件解码。所有器件在同一 I/O 位置都是有效的。菊花链的主要缺点是可及性(或延迟)。即使为了改变单个 DAC 的状态, 处理器都必须仍然从 I/O 端口输出完整的数据流。

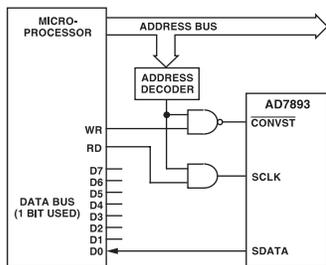


问：既然串行数据转换器节省如此多的空间与走线,为什么没有被广泛用于每个空间敏感的应用呢?

答：串行接口的主要缺点是以速度换取空间。例如,对并行 DAC 编程,只需将数据放到数据总线,然后通过单脉冲将数据送入 DAC 中。然而,对串行 DAC 进行写操作时,数据比特必须按顺序送入(N 个时钟脉冲用于 N 位转换器),并在其后跟随一个负载脉冲。处理器的 I/O 端口将耗费大量时间与串行转换器通信。因此,吞吐量超过 500ksp/s 的串行转换器并不常见。

问：我的 8 位处理器没有串行端口。是否有什么办法能够将 AD7893 之类的 12 位串行 ADC 与处理器并行总线进行连接?

答：这完全可以做到,只要使用一个外部移位寄存器,它以串行加载(并且是异步地),然后读入处理器的并行端口。然而,如果这个问题前提是“无需外部逻辑”,则串行 ADC 可当作 1 位并行 ADC 来连接。将转换器的 SDATA 引脚连接到处理器的一个数据总线(下图中它连接到 D0)。使用一些解码逻辑,可将转换器映射到处理器的一个内存单元,这样可通过 12 个连续读指令读取转换结果。然后,其它软件指令将整合 12 个字节的最低有效位,读入单个 12 位并行字。



这种技术有时称为“位拆裂”,从软件角度看非常低效。但是,在处理器运行速度比转换器快得多的应用中,也可以接受。

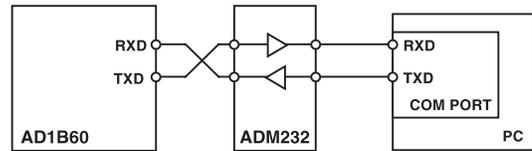
问：在最后一个例子中,处理器写入信号的门控版本用于启动 AD7893 的转换。此方法会有什么问题吗?

答：很高兴您能提出这个问题。在此例中,可通过向 AD7893 的映射内存单元虚拟写入来启动转换。没有数据进行交换,但是处理器提供启动转换所需的写脉冲。从硬件角度来说,它能避免产生转换信号,所以这种配置非常简单。

然而,在交流数据采集应用中,必须对信号进行定期采样,因此并不推荐使用该技术。即使处理器通过编程对 ADC 进行周期写操作,写脉冲的相位抖动也会严重降低可实现信噪比(SNR)。门控过程有可能使写信号抖动更加严重。例如,采样时钟相位抖动等级小至 1 ns,也将使理想的 100kHz 正弦波的 SNR 降低到大约 60dB(低于 10 个有效位分辨率)。此外还存在一种危险,即采样信号的过冲与噪声会进一步降低模数转换的完整性。

问：何时应选择带异步串行接口的转换器?

答：异步链路允许器件之间不受时钟约束地交换数据。必须先对器件编程,从而使用同一数据格式,包括设置特定的数据速率(通常以波特表示,或比特/秒)。定义如何启动与结束传输的协议也是必要的。我们通过称之为起始位与停止位的可识别数据序列来实现。传输也可包括便于实现误差检测的奇偶校验位。



该图显示 AD1B60 数字化信号调理器如何与计算机的异步 COM 端口连接。这是一个 3 线式双向接口(为了更清晰,省略了地线)。请注意,接收与传输线路在线路另一端互换了角色。

在器件偶尔进行通信的应用中,异步数据链路非常有用。由于每一次传输都包括起始位与停止位,在任何时间,器件都可简单地输出数据来发起通信。由于不再需要时钟和控制信号,因此器件之间的连接数量也将减少。

问：我考虑选用的 ADC 的数据手册中推荐在串行接口上使用非连续时钟。为什么?

答：技术规格可能要求在进行转换时,时钟保持不活动状态。某些 ADC 有这样的要求,因为连续的数据时钟会馈送到器件的模拟部分,反而影响转换的完整性。如果 I/O 端口有帧脉冲,连续的时钟信号会在转换过程中变得不连续。它可当作门控信号,仅在数据传输时,使能转换器的串行时钟。

问：器件如何实现 SPI 或 MICROWIRE 兼容?

答：SPI(串行外设接口)和 MICROWIRE 分别是摩托罗拉和美国国家半导体公司开发的串行接口标准。大多数同步串行转换器能够很容易地与这些端口接口,但是,在某些情况下,额外的“胶合”逻辑是必需的。

问：好吧,我决定抛开偏见,在设计中使用串行 ADC。我已经按照数据手册的说明将其连接好。在微控制器读取转换结果时,ADC 似乎一直输出 FFF<sub>HEX</sub>,这是怎么回事?

答：可能通信有问题。需检查一下 ADC 与处理器之间的连接,以及定时与控制信号是如何设置的。另外还需要检查中断结构。下一次将继续这个话题,讨论在设计串行接口过程中遇到的问题。■