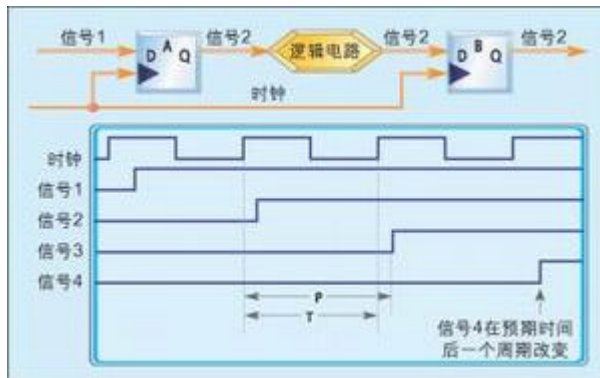


大型设计中 FPGA 的多时钟设计策略

Tim Behne 软件与信号处理部经理 Microwave Networks 公司
Email: timothyb@microwavenetworks.com

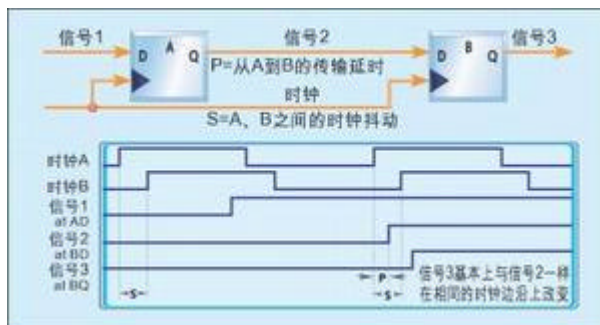
利用 FPGA 实现大型设计时，可能需要 FPGA 具有以多个时钟运行的多重数据通路，这种多时钟 FPGA 设计必须特别小心，需要注意最大时钟速率、抖动、最大时钟数、异步时钟设计和时钟/数据关系。设计过程中最重要的一步是确定要用多少个不同的时钟，以及如何进行布线，本文将对这些设计策略深入阐述。

FPGA 设计的第一步是决定需要什么样的时钟速率，设计中最快的时钟将确定 FPGA 必须能处理的时钟速率。最快时钟速率由设计中两个触发器之间一个信号的传输时间 P 来决定，如果 P 大于时钟周期 T ，则当信号在一个触发器上改变后，在下一个逻辑级上将不会改变，直到两个时钟周期以后才改变，如图 1 所示。



传输时间为信号在第一个触发器输出处所需的保持时间加上两级之间的任何组合逻辑的延迟，再加两级之间的布线延迟以及信号进入第二级触发器的设置时间。无论时钟速率为多少，每一个 FPGA 设计所用的时钟必须具有低抖动特性。抖动 S 是触发器的一个时钟输入到另一个触发器的时钟输入之间的最大延迟。为使电路正常工作，抖动必须小于两个触发器之间的传输时间。

图 2 显示了如果抖动大于传输时间($S > P$)将出现的情况，该电路用时钟的两个上升沿来延迟信号 1。然而，信号 1 上的一个改变会在相同的时钟周期上传输到的信号 3 上，从而引起信号 2 的改变。因为 $S > P$ ，电路将不能不正常。

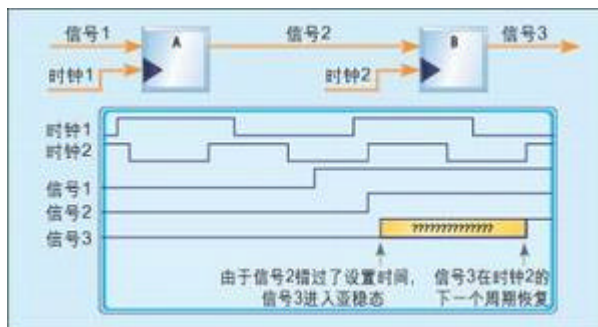


须注意的是，时钟速率与传输延时并没有什么关系，甚至普通的 100bps 时钟也会出现抖动

问题。这意味着虽然 FPGA 供应商宣称他们的芯片具有较短的传输时间和很高的时钟速率，但抖动问题可能会严重，甚至那些没有运行在最高速率上的设计也是如此。

好在 FPGA 供应商已经认识到时钟抖动的影响，并在他们的芯片中提供低抖动的布线资源。这些特殊的布线能够在芯片中一个给定范围内的任何两个触发器之间提供一个确定的最大抖动。部分产品的低抖动资源覆盖了整个芯片，而其它的则可能只覆盖了 FPGA 逻辑块中的一个特定的行或列。对于一个需要很多不同时钟源的设计，这些低抖动 FPGA 是比较理想的选择。

多时钟设计的最严重问题之一是用异步时钟将两级逻辑结合在一起。由于异步时钟会产生亚稳态，从而严重降低设计性能，或完全破坏设计所能实现的功能。在触发器的时序要求产生冲突时(设置时间和保持时间)将产生亚稳态，触发器的最终输出是未知的，并使整个设计处于不确定状态。如果有一级逻辑要将数据异步地发送到另一级，图 3 所示的情形将不能满足触发器的设置和保持时间要求。确切地说，如果设计中含有异步逻辑将有可能产生亚稳态。在处置异步资源时必需非常小心，因为这可能产生一些很严重的问题。



多时钟设计

本文以电信应用中的 E3 多路复用/解复用设计为例。如图 4 所示，多路复用器接收来自一组独立线路接口芯片的 16 个独立 E1 信道，每一个信道都工作于 2.048MHz；经复用后，这些 E1 流组合成 4 个 E2 流，分别工作在 8.0448MHz；4 个 E2 流最后组合成一个 E3 流，以 34.368Mbps 的速率串行发送出去。在接收端执行相反的操作：解复用器从 E3 流提取 4 个 E2 数据流，然后从 E2 流提取 16 个 E1 流，最终将 E1 流发送到接收端的线路接口芯片。



这些 E1 线路接口在发送和接收时都独立工作，因此 2.048MHz 的时钟速率可以有 +/- 20ppm

的偏差。同样，因为大多数系统同时发送和接收数据，分立的多路复用器和多路解复用器将提供 2 个独立的 E3 流(发送和接收)。因此，两个 34.368MHz 的时钟可以存在细微的差异。

由于 E2 流是在芯片上产生的，这些 E2 多路复用器可以共享同一个 8.448MHz 时钟。然而，由于接收的数据速率与我们所设计的板无关(且不能假定所有 E2 多路复用器使用相同时钟)，所以 E2 解复用器时钟必须能工作在略为不同的速率下。

此外，假定设计中需要一个由工作频率为 1MHz 的处理器控制的独立 SPI(串行外围接口)总线接口，该接口用于状态和控制。这样一来，设计中总共用了 32 个 2.048MHz 时钟，5 个 8.448MHz 时钟，2 个 34.368MHz 时钟和一个 1MHz 时钟，总共多达 40 个时钟。

本设计中最快时钟是 34.368MHz E3 时钟。FPGA 的最大时钟速率的确定很重要，因为设计的差异将影响到该最大值。然而，在芯片商的资料手册中常常可以看到“全局时钟设置及保持时间”和“至 CLB 输出的时钟”两个参数，将这两个参数的最大值相加，再增加 25%就能可以得到最小时钟周期的初略值，在最大时钟速率条件下允许 10%的余量，以保证过热条件下能正常工作。因此，我们设置的最小速率为 40MHz，很多较新的 FPGA 都能够很容易地支持该频率。事实上，FPGA 供应商已经推出了超过 300MHz 的器件。

在确定了能满足最大频率要求的 FPGA 后，就需要保证有足够的空间来实现你的设计。如果所选的 FPGA 没有足够的余量，就不能提供足够的布线资源来满足设计的时序约束。通常芯片供应商宣称的速率是最佳条件下的速率，FPGA 供应商一般建议 FPGA 逻辑在布线功能开始明显变差以前可以用到 80%。在选择 FPGA 器件时，建议在新的设计时最好使 FPGA 逻辑用到 50%左右，这样就允许计算起始设计大小出现超差，以及为在设计起动后产生不可避免的设计变更留出空间。如果最终的设计只占用低于 50%的资源，则可以使用同一系列中较小的 FPGA 以降低成本。

通过时序约束来规定慢时钟速率，从而可以改进设计中最快时钟的布线。在多路复用器例子中，如果设置 FPGA 布线工具 SPI 总线时钟为 1MHz，而 E3 时钟为 40MHz，布线工具将尽量使 E3 时钟的逻辑电路模块相邻布局。如果由于空间的限制而不能将全部电路布局在一起，则首先应将 SPI 逻辑另外布局，因为 SPI 逻辑可以处理更长传输延迟。所有 FPGA 供应商的布线工具都能规定这些较慢时钟速率。

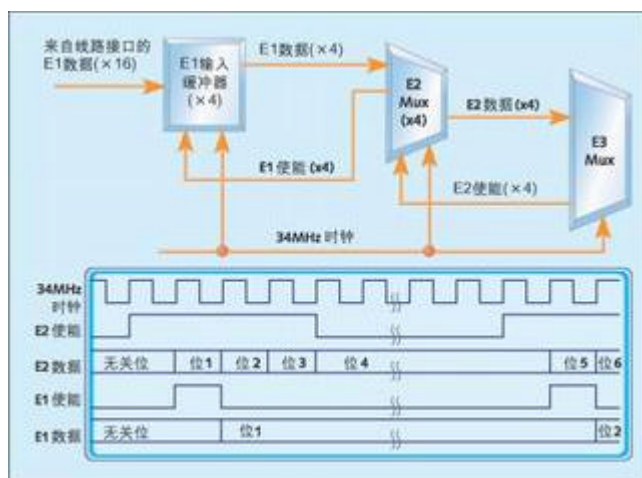
减少时钟数量

根据市场调查，目前还没有哪个 FPGA 器件能够支持这种多路复用器/解复用器设计所需的 40 个时钟。所以，我们必须减少所需要的时钟数。

首先了解 E2 和 E3 多路复用器的时钟。前面已经分析了 4 个 E2 多路复用器工作在相同时钟下的可接受度，E3 多路复用器运行于比 E2 时钟高得多的速率，必需使用一个不同的时钟。但是，如果从 E3 时钟中引出 E2 时钟是否可行呢？因为 E3 多路复用器要从每个 E2 支路得到数据，我们可以在需要 E2 多路复用器给我们数据时，简单地将脉冲送给每个多路复用器。我们没有去掉任何时钟，但 E2 时钟现在是基于 E3 时钟。

如果在所有的多路复用器中也使用同样的时钟，并且只使用一个使能信号来告诉 E2 多路复用器什么时候工作，这时会产生什么问题呢？如果 E3 多路复用器用 34.368MHz 时钟产生使能信号，在这些使能信号上的抖动不会比用在 FPGA 中任何其它同步逻辑更大。所以，

使能信号可以使用正常(高抖动)布线资源 这样就不需要单独的8.448MHz多路复用器时钟，读取 E1 数据缓冲器的数据时也是一样。换言之，如果 E2 多路复用器需要数据，它可以激活到特定缓冲器的使能信号。到缓冲器的时钟本身能够保持 E3 多路复用器所用的 34.368MHz 时钟，如图 5 所示。



最后，我们检查 16 个从线路接口芯片输入到 FPGA 的 E1 时钟。这些时钟会产生下面几个问题：首先，16 个时钟将占用太多可用芯片时钟布线资源；其次，在同一个 FPGA 中使用 16 个异步时钟来驱动相互邻近的触发器，由于地弹、串扰和其它效应将产生噪声问题。例如，由于噪声的原因，一个正边沿触发器会在下降边沿时改变输出状态，此类问题将难以处理。

作为一种可能的解决方案，我们推荐使用一个最快的时钟来对 16 个 E1 时钟采样。16 个输入时钟都接近 2.048MHz，并且还有一个 34.368MHz 的系统时钟。这样，我们可以用 34.368MHz 时钟来对 16 个 E1 时钟分别进行采样，并将结果存储在一个 16 位存储器中(每个 E1 时钟一位)。然后，我们可以使用一个算法来检测在 E1 时钟上由低至高的转换，为每一个 E1 数据信号产生一个使能信号，并在下一个周期(34.368MHz)中存储数据。

要成功实现这种方案，还必需了解时钟-数据关系以避免在数据变化时对数据采样，参见图 6。请注意在时钟采样电路的第一级中使用了两个触发器以确保在亚稳态下正常工作。另外需要注意的是，数据和时钟必须具有相同时钟周期数的延迟。

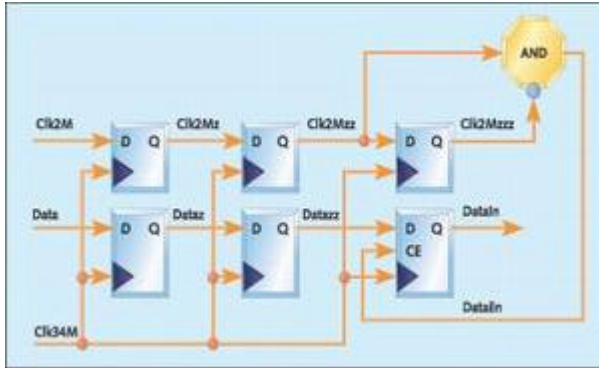
我们已经成功地将多路复用器的时钟减少到一个时钟，同样的方法可否用在解复用器呢？E3 解复用器必须采用一个外部输入时钟，这是因为驱动 E3 输入数据的同一个片外器件利用到该时钟。由于 E3 解复用器知道在什么时候发送数据到 E2 解复用器，并能对每个 E2 解复用器产生使能信号，而四个 E2 解复用器能工作在与 E3 多路复用器相同的主时钟下。同样，E2 多路复用器能够为每个 E1 流产生使能信号。

如果我们假设线路接口芯片能够接受有间隙的时钟(gapped clock)，一旦确定发出 E1 使能信号,我们只需要发送一个时钟脉冲至线路接口。然而，只需要简单地发送使能信号本身至接口芯片而不必产生一个新的时钟。因为送至接口的数据将在使能信号的下降沿产生改变(参见图 5)，我们需要确认接口在时钟的上升沿进行采样。因为使能信号仅在线路接口芯片上而不是在 FPGA 内用作一个时钟，就没有必要在一个低抖动源中进行布线。注意这是在知道将再也不会用主时钟的连续脉冲送数据到相同的从属器件中时才这样做。

1MHz SPI 时钟并不能简单去掉，但我们现在通过使用使能信号和时钟过采样技术，将原先 40 个时钟减少到 3 个，这样我们就有了更大的器件选择范围。

异步时钟

在用异步时钟产生任何逻辑前应该尽量先考虑采用其它替代方法，用异步时钟的组合逻辑是产生亚稳态问题的主要原因。同样，当违反触发器的设置和保持时间约束时，在一个短时间内输出将具有不确定性，并且将最终设定在“1”或“0”上，确切的状态不可预知。



幸运的是对于亚稳态性问题已经有一些解决方案。图 6 说明了这一方案，这是一种双寄存器方法：进入第一级触发器的数据与时钟异步，所以第一级触发器几乎肯定是亚稳态；然而，只要亚稳态的长度小于时钟的周期，第二级触发器就不会进入亚稳态。但是，FPGA 供应商很少提供亚稳态时间，尽管该时间一般小于触发器的设置和保持时间之和。

如果时钟不是太快而且能满足时序约束的话，像图 6 所示的电路将可能不会产生亚稳态。只要所有输出到触发器的通路由相同时钟驱动，即使第一级触发器的输出可用，通常还是需要像图 6 中电路来将亚稳态隔离到一条短线。采用这种方法后，将不太可能出现由于电路的改变而无意地在无时钟驱动的逻辑中用到该亚稳态线。

如果读数据的是一个计数器，像从一个异步 FIFO 读或写地址，你应该考虑下列情况：一个传统的 3 位计数器在状态之间有一个、两个或三个位的变化，例如读数发生在计数器从“011”到“100”变化的瞬间，则所有三个位的值将不确定，读的值会是八种可能状态中的任一种。如果计数器是使用格雷码，如表所示，则每次仅有一位发生状态改变，如果读数发生在计数器变化的瞬间，则只有一个位会有问题，所以在读操作中只有两种可能结果，而且这两种可能结果是计数器正好在读以前的值和正好在读以后的值时。因为读正好发生在计数器产生变化的瞬间，你不可能确切地说哪个值是正确的，即两者都应该认为是有效的。

另一个避免异步时钟问题的方法是忽略较慢的时钟，并用较快的时钟来采样。这需要数据有特殊的成帧特性(例如，具有一个前导码)来定义数据边界。这是一个常用的方法，在差不多每一个具有 UART 形式的嵌入式系统都有应用。该方法是：采用一个非常快的时钟，比如数据符号率的 16 倍，在连续发现 15 个起始字符后开始采样，则下一个 16(左右)位相当于送的第一个位，再下一个 16(左右)位对应下一个位，并以次类推。