



XAPP703 (v2.3) 2006 年 9 月 6 日

利用 Virtex-4 器件实现 QDR II SRAM 接口

作者: Derek Curd

提要

本应用指南说明了利用 Virtex™-4 器件实现 2 字或 4 字突发四倍数据速率 (Quad Data Rate, QDR™II) SRAM 接口及其时序的详细信息。此综合参考设计利用 Virtex-4 系列独有的 I/O 和时钟控制功能, 实现了很高的性能水平。

本解决方案中介绍的直接时钟控制方法在最大程度降低资源使用的同时, 大大简化了 FPGA 中的读数据采集任务。此方案提供了一个简单的用户接口, 可通过利用一个或多个 QDR II 接口简便地集成到一个完整的 FPGA 设计中。

简介

QDR SRAM 器件为满足更高的带宽存储要求而开发, 以网络和电信应用为目标。基本 QDR 架构具有独立的读、写数据通路, 便于同时操作。每个时钟周期内, 两个通路均使用双倍数据速率 (DDR) 发射发出两个字, 一个在时钟上升沿发出, 一个在时钟下降沿发出。结果, 在每个时钟周期内会传输四个总线宽度的数据 (两个读和两个写), 这就是四倍数据速率的由来。

QDR I 和 QDR II 的规范由 QDR 联盟 (Cypress、IDT、NEC、Samsung 和 Renesas) 共同定义和开发。有关 QDR 规范、QDR 联盟和 QDR 存储器产品的其他信息, 请参考第 23 页“附录”。

QDR 存储器器件以 2 字突发和 4 字突发架构提供存储器器件。针对每个读或写请求, 2 字突发器件传输两个字。DDR 地址总线用于在前半个时钟周期允许读请求, 在后半个时钟周期允许写请求。与之相反, 4 字突发器件针对每个读或写请求传输四个字, 这样便只需要一个单倍数据速率 (SDR) 地址总线, 能最大程度地利用数据带宽。读、写操作请求必须在交替的时钟周期 (即不重叠) 内进行, 以分享地址总线。

本应用指南中讨论的参考设计针对的是 2 字或 4 字突发 QDR II SRAM 器件。QDR II 架构的独特性能之一是源同步时钟 (CQ) 输出, 它与器件输入时钟 (K) 频率锁定, 与通过读通路输出 (Q) 传输的数据边沿对齐。CQ 时钟输出使用 QDR II 存储器器件内的延迟锁定环 (DLL) 电路重设时间, 使其与 Q 数据输出对齐。对于在远端器件 (在此设计中为 Virtex-4 器件) 上进行的读数据采集操作, 此种时钟前向 (亦称源同步) 接口方法允许较长的时序余量, 并且使我们得以实现在此参考设计中使用的简单而精致的直接时钟控制方法。本应用指南将详细讨论该方法。

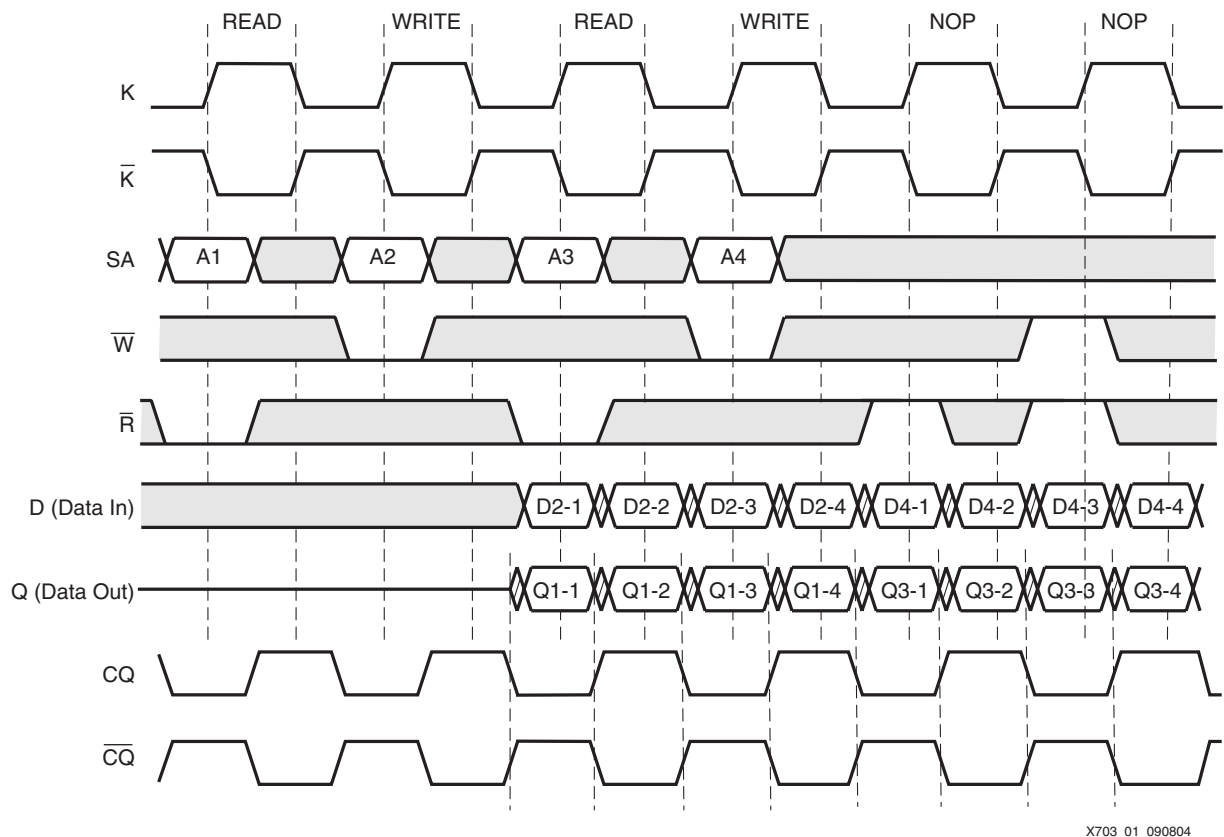
图 1 是一个时序图, 表明了 4 字突发 QDR II 存储器接口上的并发读 / 写操作。QDR II 存储器的所有输入与输入时钟 (K 和 \bar{K}) 同步, 且通常采用与 K 和 \bar{K} 时钟边沿中心对齐的方式传输给存储器。请务必注意; 低有效的读控制 (\bar{R}) 和写控制 (\bar{W}) 引脚在时钟周期内交替出现, 以实现单独 SDR 地址总线 (SA) 的共享。

写总线数据输入 (D) 值在 DDR 模式下传输到存储器, 开始于写控制引脚激活后的下一个 K 时钟上升沿。读总线数据输出 (Q) 值在 DDR 模式下从存储器中传出, 并与 CQ 和 \bar{CQ} 源同步回送时钟输出边沿对齐。在 \bar{CQ} 时钟输出的上升沿 (紧随 \bar{K} 输入时钟的下一个上升沿之后), 读总线中的第一个字开始传输。

© 2004 ? 006 Xilinx, Inc. All rights reserved. All Xilinx trademarks, registered trademarks, patents, and further disclaimers are as listed at <http://www.xilinx.com/legal.htm>. All other trademarks and registered trademarks are the property of their respective owners. All specifications are subject to change without notice.

NOTICE OF DISCLAIMER: Xilinx is providing this design, code, or information "as is." By providing the design, code, or information as one possible implementation of this feature, application, or standard, Xilinx makes no representation that this implementation is free from any claims of infringement. You are responsible for obtaining any rights you may require for your implementation. Xilinx expressly disclaims any warranty whatsoever with respect to the adequacy of the implementation, including but not limited to any warranties or representations that this implementation is free from claims of infringement and any implied warranties of merchantability or fitness for a particular purpose.

在从数据输入 (D) 字中选择要写存储器的特定字节时, QDR II 还可使用低有效的字节写 (\overline{BW}) 使能引脚。为清晰起见, 图 1 中省略了这些信号。

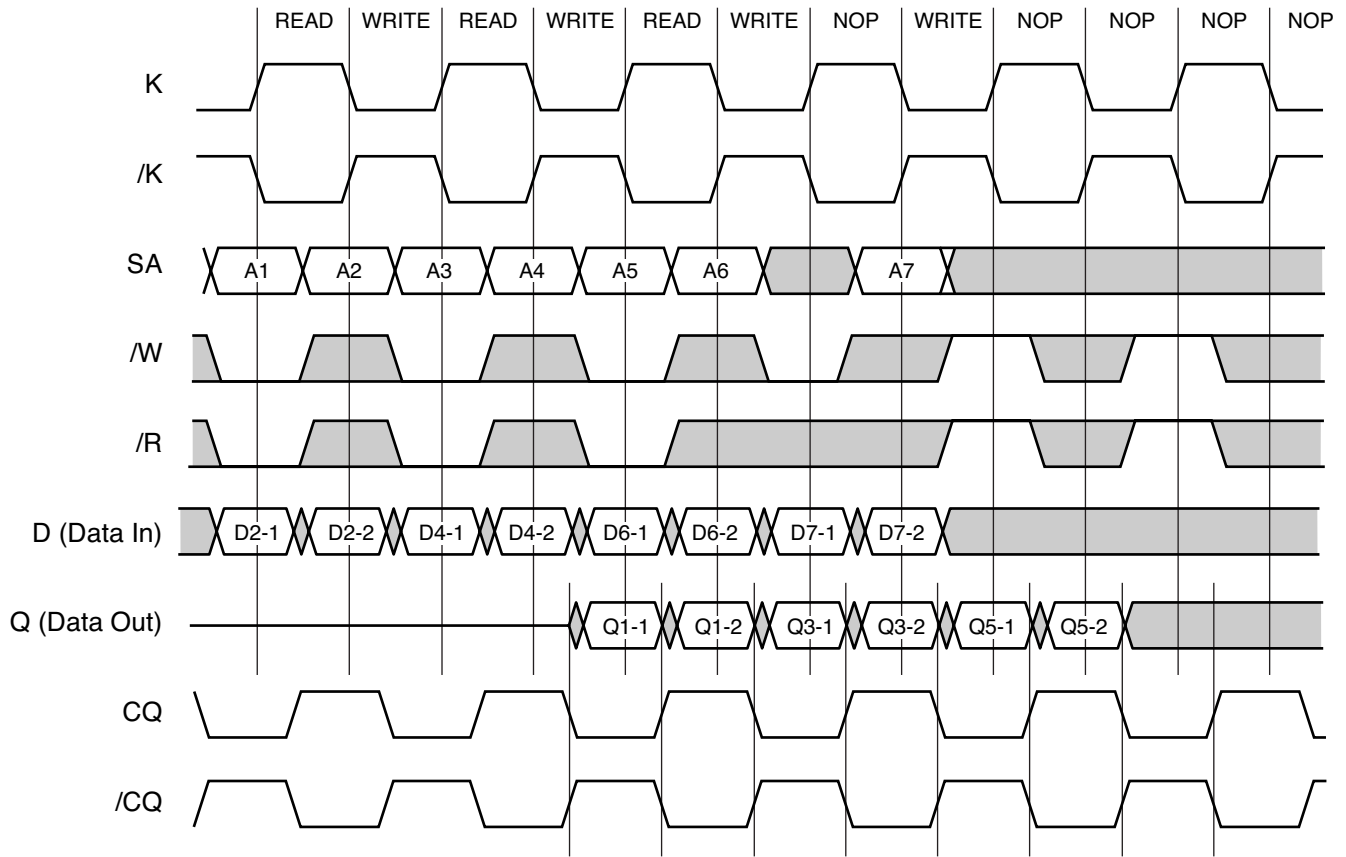


X703_01_090804

图 1: 带并发读与写操作的 4 字突发 QDR II SRAM 时序图

图 2 是一个时序图, 表明了 2 字突发 QDR II 存储器接口上的并发读 / 写操作。在时钟的前半个周期, DDR 地址总线允许读地址传输给存储器; 在时钟的后半个周期, DDR 地址总线允许写地址出现其中。因此, 低有效的读控制 (\overline{R}) 和写控制 (\overline{W}) 引脚可在同一时钟周期内设定。

两个写总线数据输入 (D) 值在 DDR 模式下传输到存储器, 开始于写地址确认前的那个 K 时钟上升沿。读总线数据输出 (Q) 值在 DDR 模式下从存储器中传出, 并与 CQ 和 \overline{CQ} 源同步回送时钟输出边沿对齐。在 \overline{CQ} 时钟输出的上升沿 (紧随 K 输入时钟的下一个上升沿之后), 读总线中的第一个字开始传输。

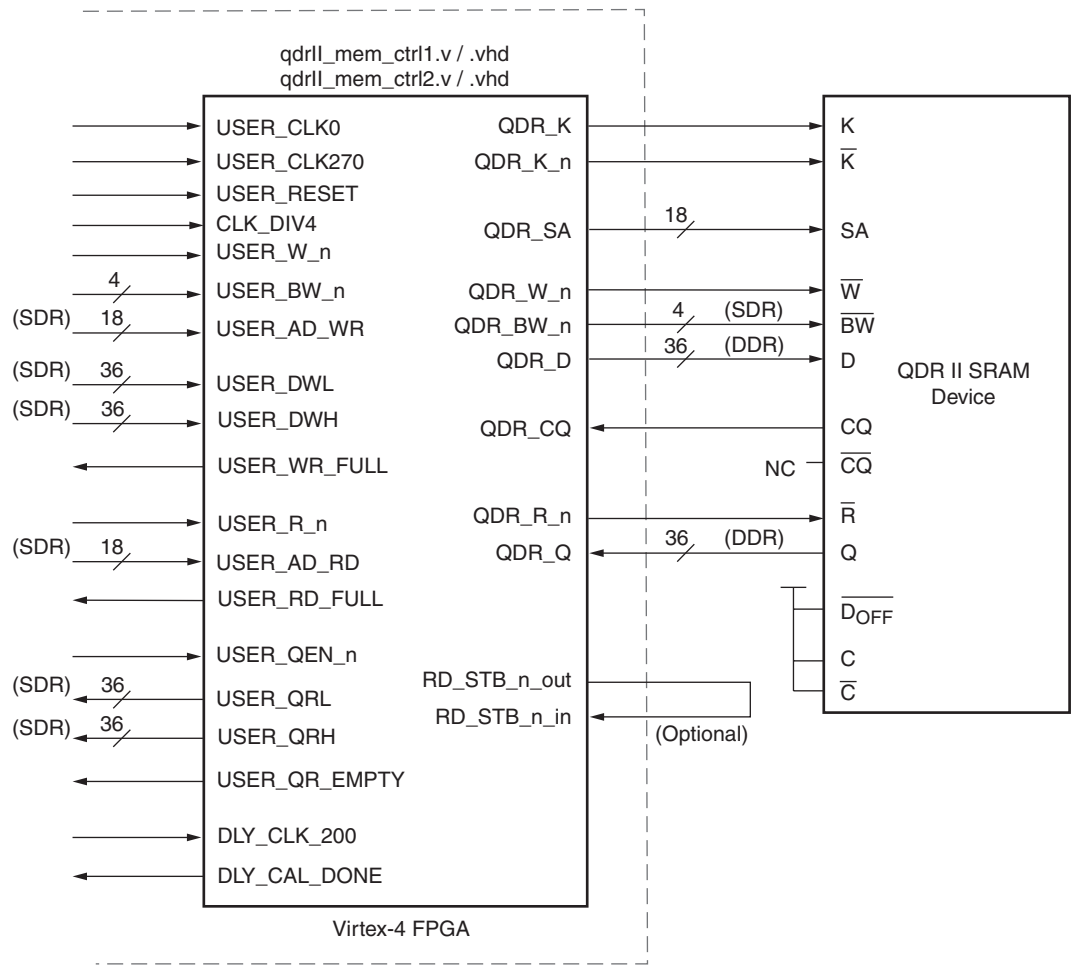


X703_02_072705

图 2: 带并发读与写操作的 2 字突发 QDR II SRAM 时序图

设计概述

图 3 为 Virtex- 4 QDR II 参考设计的高层次框图，表明了 QDR II 存储器器件的外部连接以及用于发出读 / 写命令的内部 FPGA 架构接口。



x703_03_080905

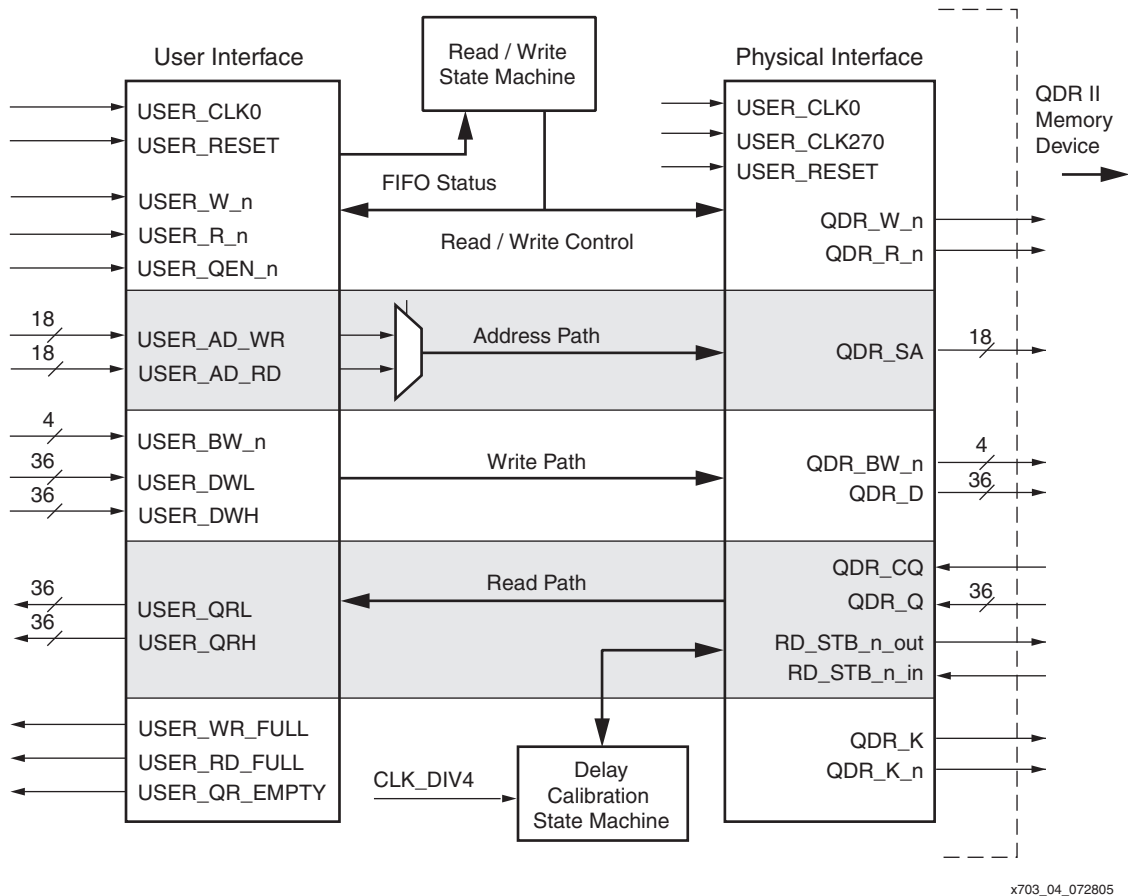
图 3: Virtex- 4 QDR II 参考设计框图

在图 3 中，QDR II 器件的 \overline{D}_{OFF} 、C 和 \overline{C} 引脚均被设定为 High。此配置启用 QDR II 器件的 CQ 源同步回送时钟功能，对于参考设计的正确运行也很有必要。

此参考设计的 RD_STB_n_out 输出本质上与 QDR_R_n 读选通脉冲完全相同，该脉冲的通路为电路板布线全程，以匹配进、出存储器器件的延迟。这会创建一个数据有效选通脉冲，与通过读通路 (Q) 从存储器返回的值同步。此选通脉冲会进入 RD_STB_n_in 输入，并会在此参考设计中被用作读数据 FIFO 的写使能信号。此外还有一种方法，可使读数据 FIFO 的写使能写全部在 Virtex- 4 器件内生成——只需将读选通脉冲适当延迟几个周期，使其与从存储器中返回的读数据值 (Q) 同步即可。“读 FIFO 选通脉冲生成”部分对这两种采集 FIFO 数据的方法进行了更加详细的讨论。

如图 4 所示，Virtex- 4 QDR II 参考设计由以下四个主要部分组成：

- 用户接口
- 物理接口
- 读 / 写状态机
- 延迟校准状态机



x703_04_072805

图 4: QDR II 参考设计的组成部分

此用户接口使用完全基于 SDR 信号的简单协议创建读 / 写请求。此模块主要由 FIFO16 基元构成，用于在读 / 写操作执行之前或之后存储相应的地址和数据值。有关此用户接口时序协议的详情，请参见下一部分。

读 / 写状态机主要负责监控用户接口模块内 FIFO 的状态，调整用户接口和物理接口间的数据流，并向外部存储器器件发出实际读 / 写命令。此状态机确保读 / 写操作按照 QDR II 存储器规范的要求，以并发方式在最短延期内完成。

物理接口负责生成正确的时序关系和 DDR 信号，以便以符合其命令协议和时序要求的方式与外部存储器器件通讯。

延迟校准状态机是直接时钟控制方法中一个不可分割的组成部分，用于大大简化在 FPGA 内采集读数据的任务，同时又实现最佳性能。Virtex-4 器件的每个输入引脚都包含一个可动态调整的可编程延迟元件 (IDELAY)，用于控制 5 ns 窗口中输入通路上的延迟时间。延迟校准状态机利用此独特的功能调整读数据从存储器器件返回的时序，因此它无需任何复杂的本地时钟控制或数据重新采集技术即可直接与全局 FPGA 系统时钟 (USER_CLK0) 实现同步。下文包含有关直接时钟控制方法的详情。

表 1 总结了 Virtex- 4 QDR II 参考设计的规范，包括性能目标和器件应用细节。

表 1: Virtex- 4 QDR II 参考设计规范

参数		规范 / 细节	
最大频率 (按速度级别)	-10	200 MHz	
	-11	250 MHz	
	-12	275 MHz	
器件利用率	Slices	174	
	GCLK 缓冲器个数	3	
	FIFO16 (Block RAM)	6	
QDR II SRAM 运行			2 字 /4 字突发
总线宽度			36 位读 /36 位写
I/O 标准			HSTL_I_18 (1.8V 信号)
HDL 语言支持			Verilog/VHDL
要验证的存储器器件	仿真	2 字突发	Cypress CY7C1314BV18 (512Kx36-bit)
		4 字突发	Samsung K7R323684M (1Mx36-bit)
	硬件	2 字突发	Cypress CY7C1314BV18 (512Kx36-bit)
		4 字突发	Samsung K7R323684M (1Mx36-bit)

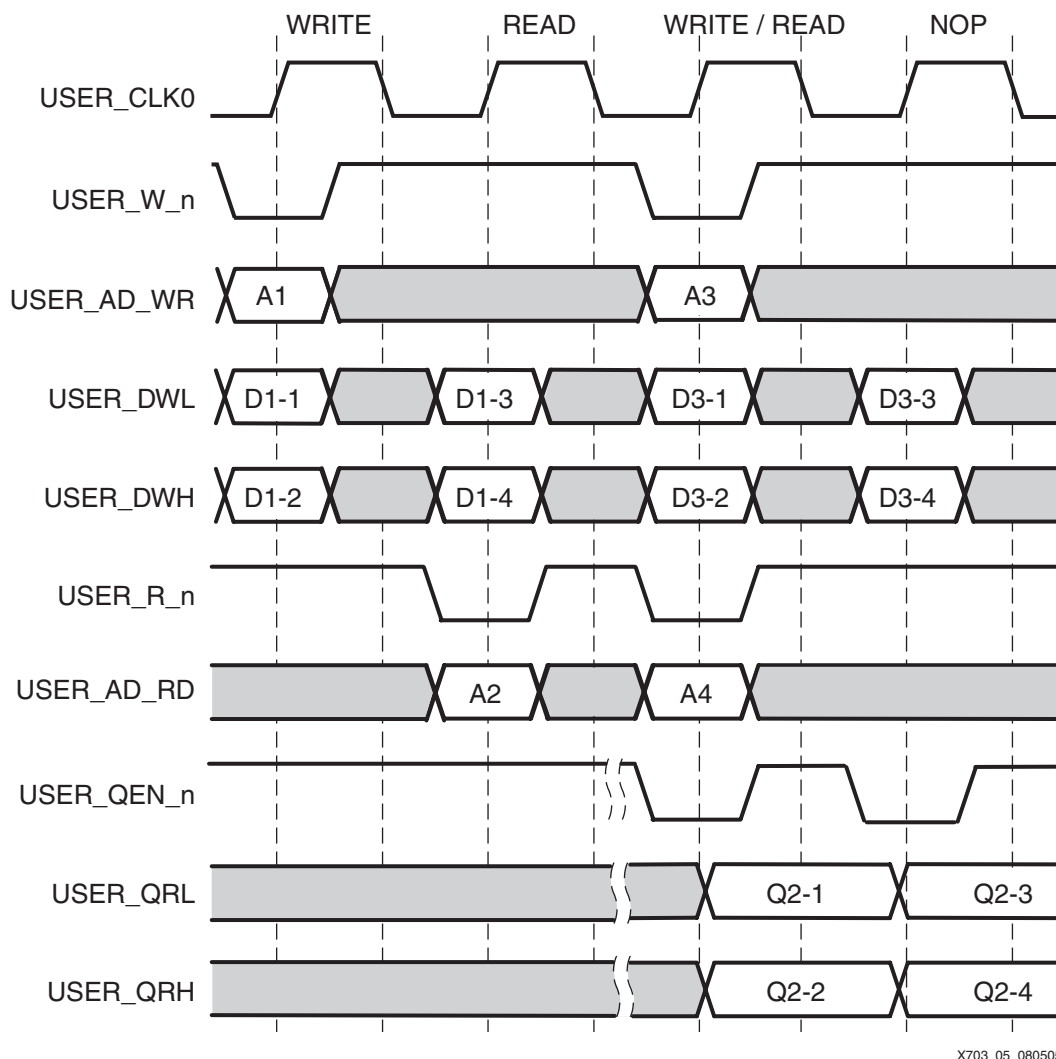
实现细节

实现 QDR II 参考设计的目的是利用 Virtex- 4 系列的独特功能。I/O、时钟控制和存储元件技术方面的发展使此设计的高性能和全承包操作得以实现。下列部分更详细地说明此设计的实现。

用户接口

用户接口模块利用六个 FIFO16 模块来存储读 / 写操作的地址和数据值。三个 FIFO16 模块用于写命令，一个用于存储写地址 (USER_AD_WR) 和字节写使能 (USER_BW_n) 信号，另外两个用于存储要写到存储器的 Low (USER_DWL) 和 High (USER_DWH) 36 位数据字。读命令也使用三个 FIFO16 模块，一个用于存储读地址 (USER_AD_RD)，另两个用于存储由于执行读而从存储器中返回的 Low (USER_QRL) 和 High (USER_QRH) 36 位数据字。

图 5 表明使用 4 字突发参考时钟控制向用户接口发布读 / 写要求所需的时序协议。如前所述，此接口使用与主 FPGA 设计系统时钟 (USER_CLK0) 同步了的所有 SDR 信号。



X703_05_080505

图 5: 4 字突发用户接口时序协议

写请求在 USER_CLK0 上升沿期间通过低有效的 USER_W_n 信号创建。此 18 位写地址 (USER_AD_WR) 必须在这同一时钟边沿传输。此时，要写到存储器的第一个和第二个 36 位数据字也会分别出现在 36 位 USER_DWL 和 USER_DWH 输入总线中。4 字突发的第三个字和第四个字会在下一个 USER_CLK0 上升沿分别传输给 USER_DWL 和 USER_DWH。

读请求在 USER_CLK0 上升沿期间通过低有效的 USER_R_n 信号创建。此 18 位读地址 (USER_AD_RD) 必须在这同一时钟边沿传输。执行读命令后，4 字突发值存储在读数据 FIFO 中。在 USER_CLK0 上升沿，一个低有效的 USER_QEN_n 会检索这些值并将其传输给 36 位 USER_QRL 和 USER_QRH 输出，其中第一个和第二个字在第一个周期传输，其中 USER_QEN_n 设定为低；第三个和第四个字在下一个周期传输，其中 USER_QEN_n 设定为 Low。

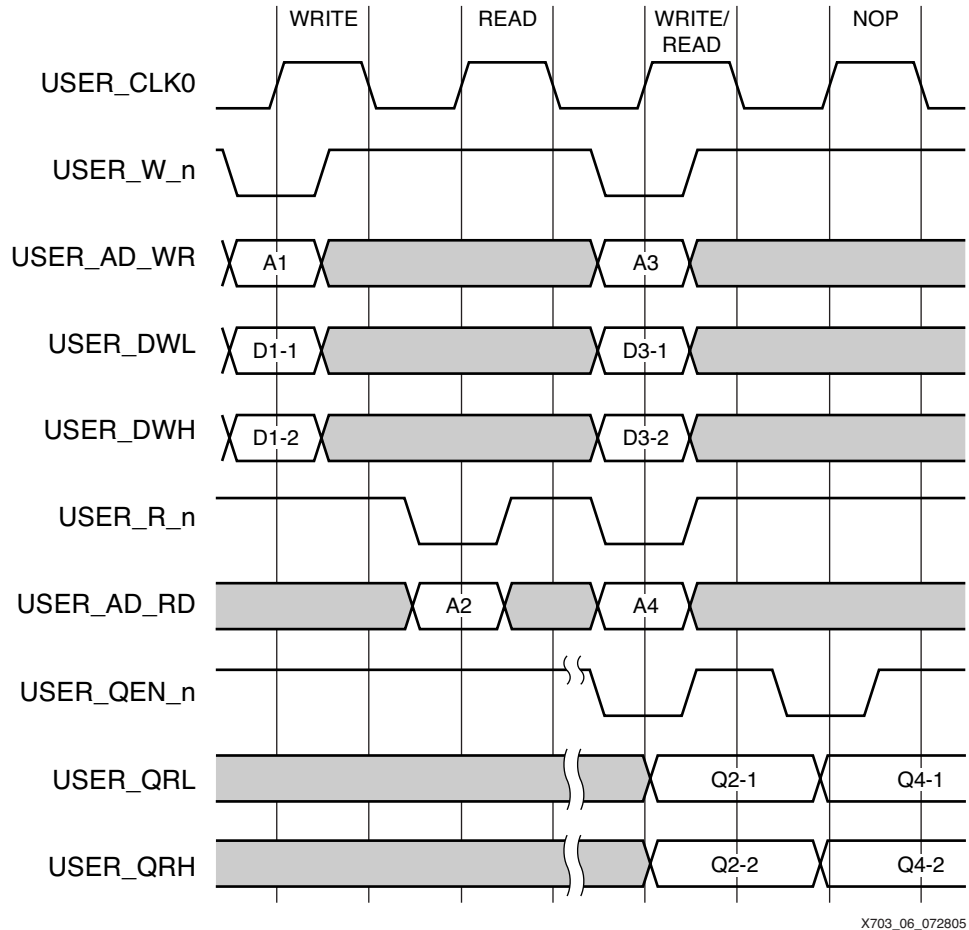
与 QDR II 存储器不同，用户接口在同一时钟周期接受读、写请求（如图 5 所示的第三个周期）。读 / 写状态机管理对外部存储器器件的读与写要求的交替，从而减轻用户接口的负担。

用户接口还提供了一批用于指明读 / 写 FIFO 状态的信号，在图 5 中未显示。高有效的 USER_WR_FULL 输出表明写 FIFO 已满。此情况表明，写请求队列缩减之前将不再接受任何写

请求。USER_WR_FULL 为 High 时创建的任何写请求都将被直接忽略。类似情况适用于读请求的 USER_RD_FULL 信号。

高有效的 USER_QR_EMPTY 输出表明不再有读数据值存储在读数据 FIFO 中。在此情况下从 USER_QRL 和 USER_QRH 总线读数值的尝试将被忽略。此情况会一直持续，直到执行其他读命令并且有相关数据值存入读数据 FIFO。

图 6 表明使用 2 字突发参考时钟控制向用户接口发布读 / 写请求所需的时序协议。写请求在 USER_CLK0 上升沿期间通过低有效的 USER_W_n 信号创建。此 18 位写地址 (USER_AD_WR) 必须在这同一时钟边沿传输。此时，要写到存储器的第一个和第二个 36 位数据字也会分别传输给 36 位 USER_DWL 和 USER_DWH 输入总线。在所有其他方面，2 字突发用户接口协议与上述 4 字突发的情况相似。



X703_06_072805

图 6: 2 字突发用户接口时序协议

读 / 写状态机

图 7 为 4 字突发读 / 写状态机的状态图。此状态机负责调节用户接口和物理接口之间的数据流。它根据保存在用户接口 FIFO 中的请求向外部存储器器件发出读 / 写命令。

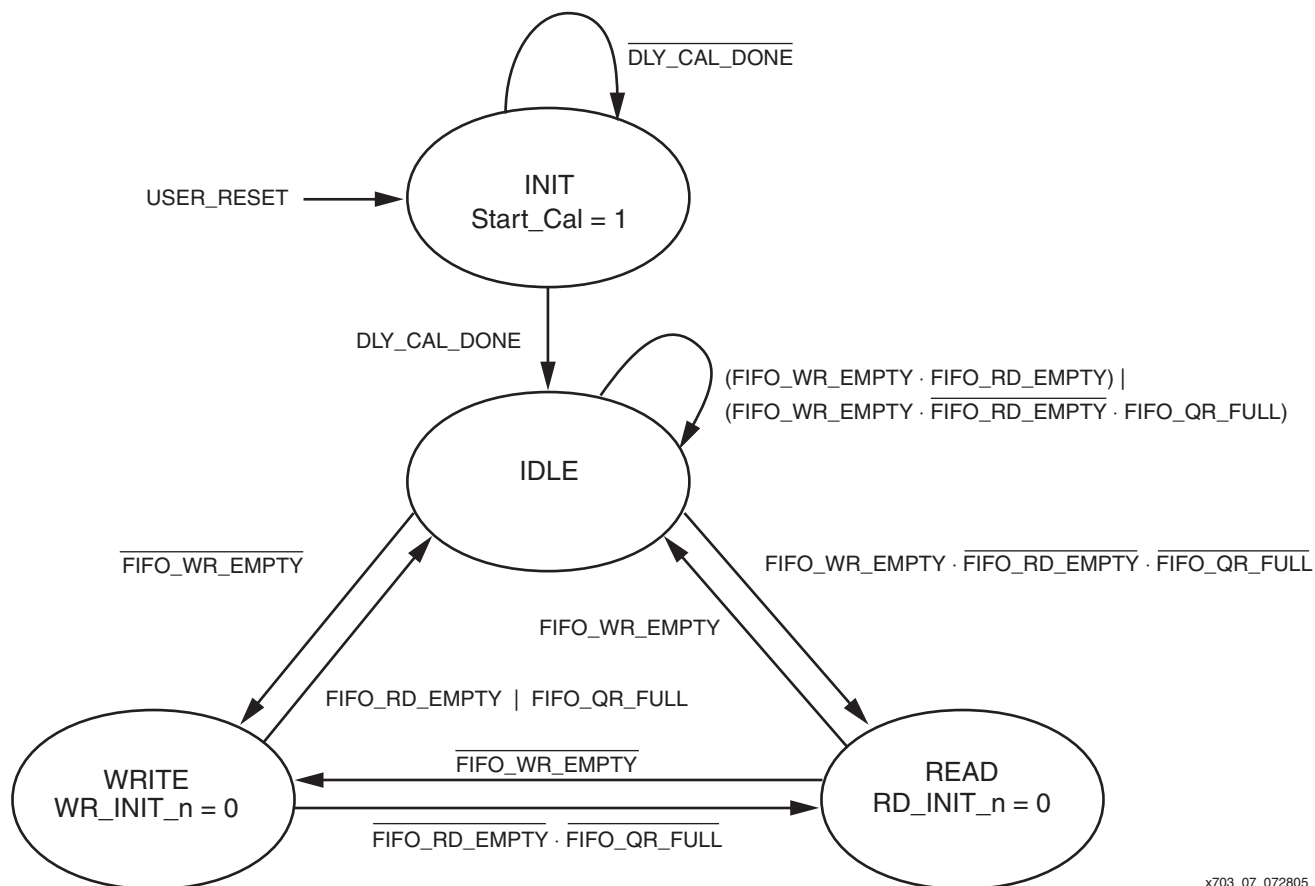
USER_RESET 每次都会将状态机恢复到 INIT 状态，此时存储器暂停运行，直到延迟校准状态机完成 IDELAY 模块上的延迟调整，使所有 QDR_Q 输入与 FPGA 系统时钟 (USER_CLK0) 的读通路数据中心对齐。校准操作完成的信号是一个高有效的 DLY_CAL_DONE 输入，该输入将读 / 写状态机转换到空闲状态，以等候来自用户接口的读 / 写请求。

在空闲状态，写命令首先假设向存储器写数据必须始终发生在有效读数据出现之前。当没有待处理的读或写请求时，此状态机将在空闲状态下循环运行。

用户接口 FIFO 中包含待处理写请求会使状态机转入写状态，在此状态下，写命令通过内置 WR_INIT_n 选通脉冲发出。此选通脉冲从 FIFO 中取出写地址和数据值，并使外部 QDR_W_n 写控制选通脉冲进入存储器器件。

如果包含待处理读请求，此状态机会随之转入读状态，此时内部 RD_INIT_n 选通脉冲激活。RD_INIT_n 选通脉冲从 FIFO 中取出读地址，并向存储器器件发送一个外部 QDR_R_n 选通脉冲。作为此处理过程的结果之一，对读数据 FIFO 中返回值的采集也将进行。

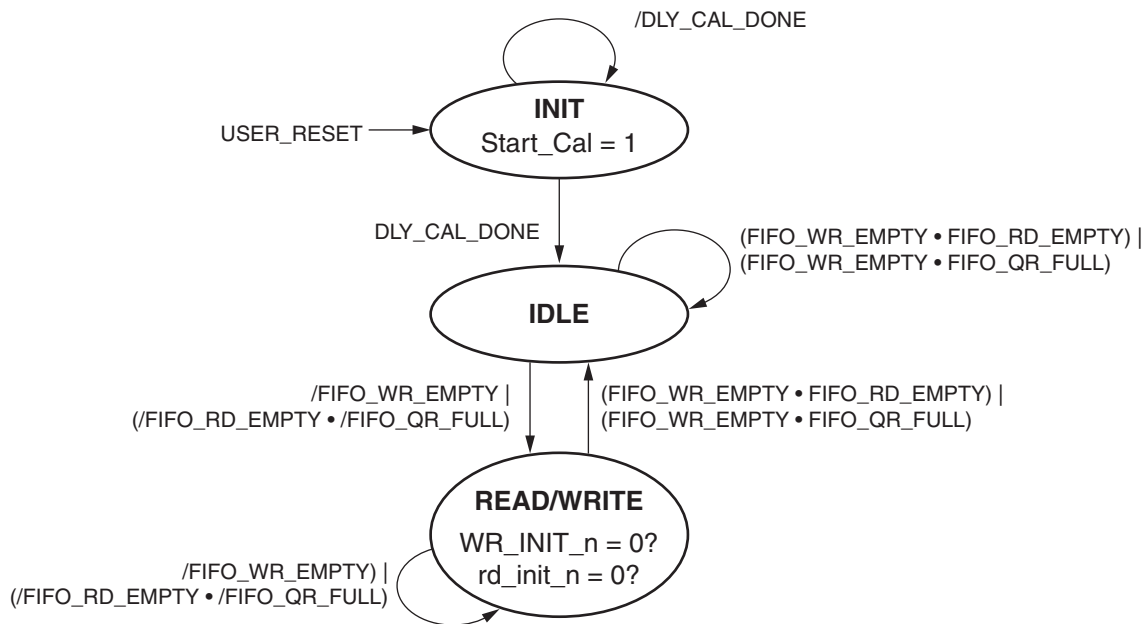
读 / 写状态机持续监控用户接口 FIFO 状态信号，以确定是否存在待处理读 / 写请求。连续不断的并发读 / 写请求流将导致状态机只在读状态和写状态之间转换，以确保正确无误地将请求交替发送到外部存储器。一组只写请求将导致空闲状态和写状态轮流出现，同样，一组读请求也会在空闲状态和读状态间转换。



x703_07_072805

图 7: 4 字突发读 / 写状态机

图 8 为 2 字突发读 / 写状态机的状态图。此状态机的运行与 4 字突发状态机的运行十分相似，但有一点例外，即针对存储器的读和写请求由单个 READ_WRITE 状态处理。所有 2 字突发 QDR II 存储器器件都允许读和写请求在同一个时钟周期发生，允许这些操作从同一种状态启动。



x703_08_080505

图 8: 2 字突发读 / 写状态机图

物理接口

Virtex- 4 QDR II 参考设计的物理接口为针对外部存储器件的读 / 写命令的通信生成实际的 I/O 信号和时序关系，包括 DDR 数据信号。它提供必要的时序余量及为满足整体设计性能规范所需的 I/O 信号标准。QDR II 设计的所有 I/O 信号都使用 HSTL-I 发送。此部分详细介绍物理接口的每个部件。

时钟机制

QDR II 设计大量使用在所有 Virtex- 4 器件 I/O 模块中都能见到的输入 (Input DDR, IDDR) 和输出 (Output DDR, ODDR) 基元。这些内置 DDR 寄存器功能极大地简化了以下任务：生成准确的时钟、地址和数据，以及与 QDR II 存储器器件通信的控制信号。IDDR 和 ODDR 基元都有多种运行模式，可分别确定采集或传输的 DDR 数据以怎样的方式出现在 FPGA 架构和 I/O 引脚中。有关 IDDR 和 ODDR 运行模式的详情，可在 [UG070](#): 《Virtex- 4 用户指南》的第 8 章“高级 SelectIO 逻辑资源”中找到。

QDR II 设计中的时钟机制（图 9）使用反沿 (opposite-edge) 模式的 ODDR 寄存器来为存储器件生成 QDR_K 和 QDR_K_n 时钟。因为时钟信号的时序与 QDR II 地址、数据和控制信号几乎相同，此时钟前向机制会有效地从时序余量因素中删除 FPGA 的 clock-to-out 参数。因此，就 clock-to-out 参数而言，所有外部传输信号都得以“匹配”。

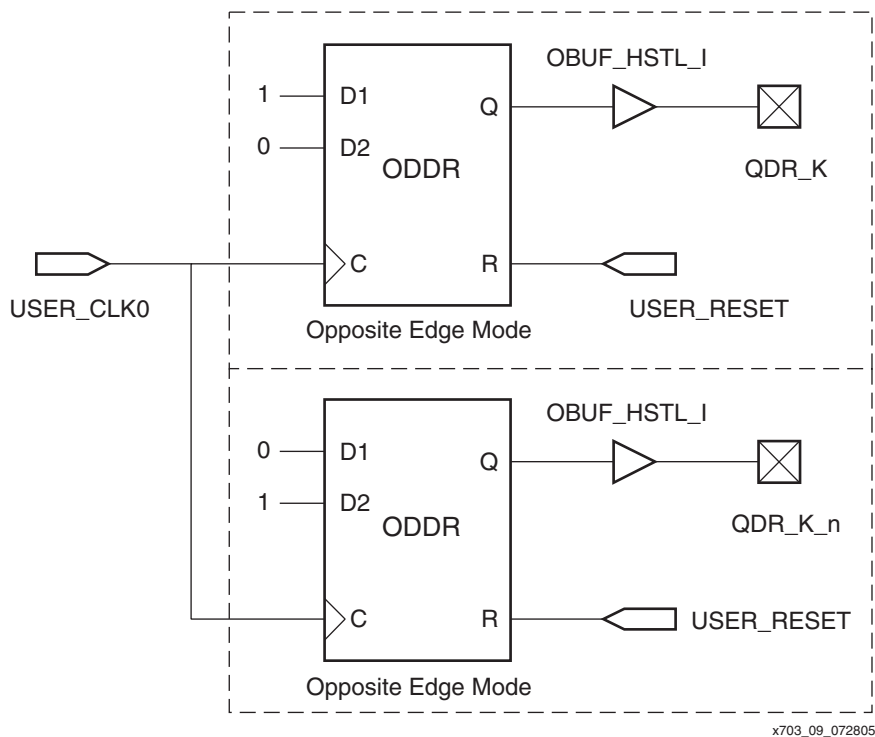


图 9: 基于 ODDR 寄存器功能的时钟前向机制

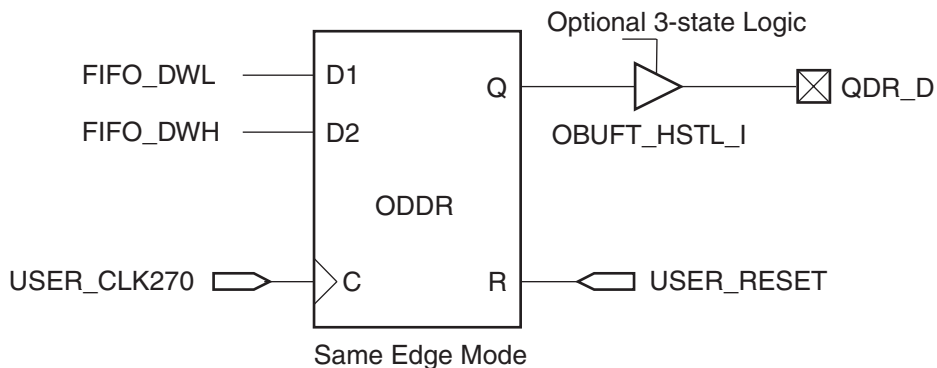
写通路

QDR II 存储器的写通路包括执行写操作所必需的地址、数据和控制信号。写地址 (QDR_AD_WR)、控制选通脉冲 (QDR_W_n) 和字节写使能 (QDR_BW_n) 信号都使用 SDR 格式。不过，写数据值 (QDR_D) 利用 DDR 信号在规定的时钟周期实现所需的 2 字或 4 字突发。

所有这些写通路信号在传输时都必须与 QDR_K 和 QDR_K_n 时钟边沿中心对齐。因此，这些信号的输出寄存器与 USER_CLK270 时钟同步。信号以同样的频率运行，但与 USER_CLK0 异相 270°（时钟周期的 75%）。这能确保对输入 QDR_K 和 QDR_K_n 时钟边沿而言，存储器件有足够的建立与保持余量。

图 10 说明了如何使用 USER_CLK270 和 ODDR 寄存器为 QDR_D 写数据通路生成 DDR 信号。ODDR 寄存器配置为同沿 (same edge) 模式，允许在 USER_CLK270 的同一上升沿从 FPGA 架构中同时采集两个 36 位数据字 (FIFO_DWL 和 FIFO_DWH)。FIFO_DWL 值在此上升沿后立即传送到 QDR_D 写数据总线，随后，FIFO_DWH 值在 USER_CLK27 的下一个下降沿从 ODDR 模块传出。对此过程加以重复，以生成一个 4 字写数据突发。

使用 I/O 模块中的单个触发器以类似的方式生成读 / 写地址、字节写使能和读 / 写控制选通脉冲，以创建与 USER_CLK270 同步的 SDR 信号。



x703_10_072805

图 10: 写数据通路实现

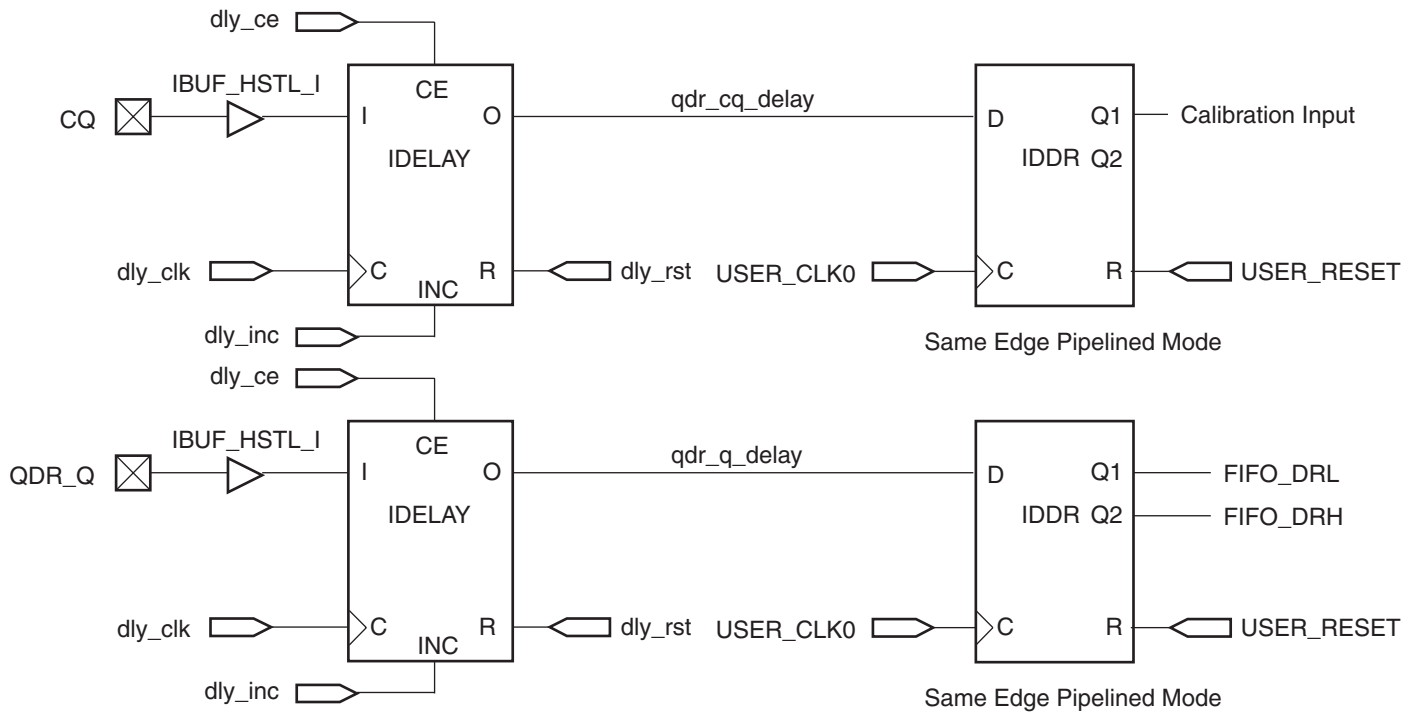
读通路

虽然读数据采集操作从根本上说比写数据传输更具挑战性，但本 Virtex- 4 QDR II 参考设计的直接时钟控制方法极大地简化了该任务。

如前所述，每个 Virtex- 4 输入引脚都具有可编程延迟元件 (IDELAY)。IDELAY 元件可以动态调整，以控制输入通路上的延迟时间。每个 IDELAY 模块每 75 ps 有 64 次 tap 延迟，这就可以在 5 ns 窗口对输入信号时序进行调整。使用 IDELAY 模块也要求用 IDELAY 元件例化 I/O 组 (bank) 中的 IDELAYCTRL 基元。IDELAYCTRL 模块使用 200-MHz 参考时钟 (± 1000 ppm 容差) 以准确地将 IDELAY tap 延迟值校准到 75 ps，不受处理过程、电压和温度变化的影响。图 3 中进入顶层设计的 DLY_CLK_200 输入充当 QDR II 设计的 200-MHz 时钟输入。

图 11 表明使用 IDELAY 基元来实现读数据采集的直接时钟控制方法。正如“简介”中所述，此方法依赖于 QDR II 存储器器件对 CQ 源同步回送时钟的使用。此时钟信号被用作“培训”信号，以中心对齐 QDR 读数据与 FPGA 系统时钟 USER_CLK0。

CQ 时钟通过一条相同通路将 FPGA 输入 QDR_Q 数据总线信号，后者是一个 HSTL 输入缓冲器，紧接它后面的是 IDELAY 模块，然后是一个 IDDR 寄存器。所有 IDELAY 模块均配置为可变延迟模式，这就允许 tap 延迟设置进行动态调整。此外，所有 IDELAY 模块由来自延迟校准状态机的同一组信号 (dly_clk、dly_ce、dly_inc) 控制。此状态机和 IDELAY 模块都与 QDR II 设计的 CLK_DIV4 时钟输入同步。该时钟处于相位中，但仅为主时钟 USER_CLK0 频率的四分之一，以确保延迟校准逻辑不在此设计的关键通路。

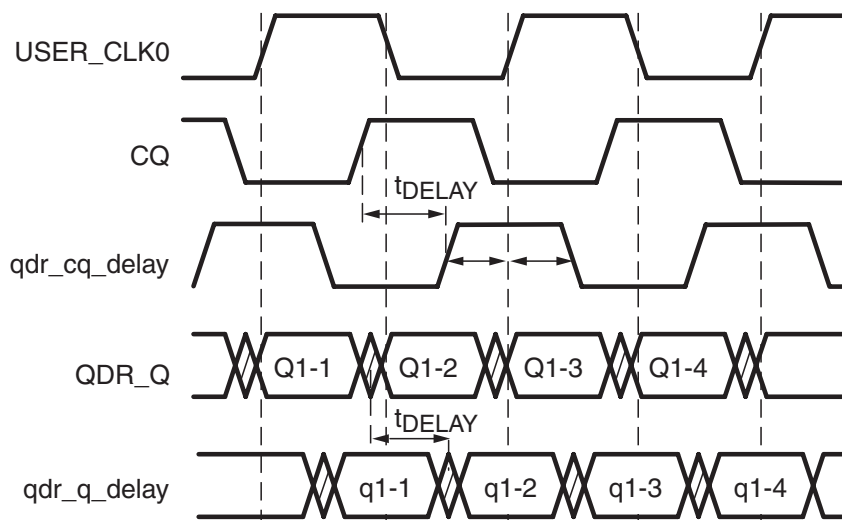


x703_11_072805

图 11: 实现读通路数据采集的直接时钟控制

延迟校准状态机监控 IDDR 寄存器在 USER_CLK0 上升边沿采集的 CQ 时钟输入状态。边沿检测算法通过改变此信号的 IDELAY tap 延迟设置查找 CQ 时钟的上升和下降边沿的位置。找到这些边沿后，对 tap 延迟设置进行调整，以将 CQ 时钟边沿中心对齐 USER_CLK0 上升边沿。CQ 时钟与来自存储器器件的 QDR_Q 数据总线边沿对齐。当同一 tap 延迟设置应用于此总线时，USER_CLK0 信号就被自动置于输入读数据字的数据有效窗口的中间。通过这种方式，读数据值可被直接采集到 FPGA 系统时钟域内，无需使用复杂的数据重新采集技术或高级时序分析（通常在跨越时钟边界时需要）。

图 12 表明如何利用相同的 tap 设置，通过 IDELAY 模块延迟 CQ 时钟和 QDR_Q 信号，以将这些信号中心对齐 USER_CLK0。qdr_cq_delay 和 qdr_q_delay 信号代表 IDDR 寄存器的输入流在流经 IDELAY 元件后的波形。



x703_12_072805

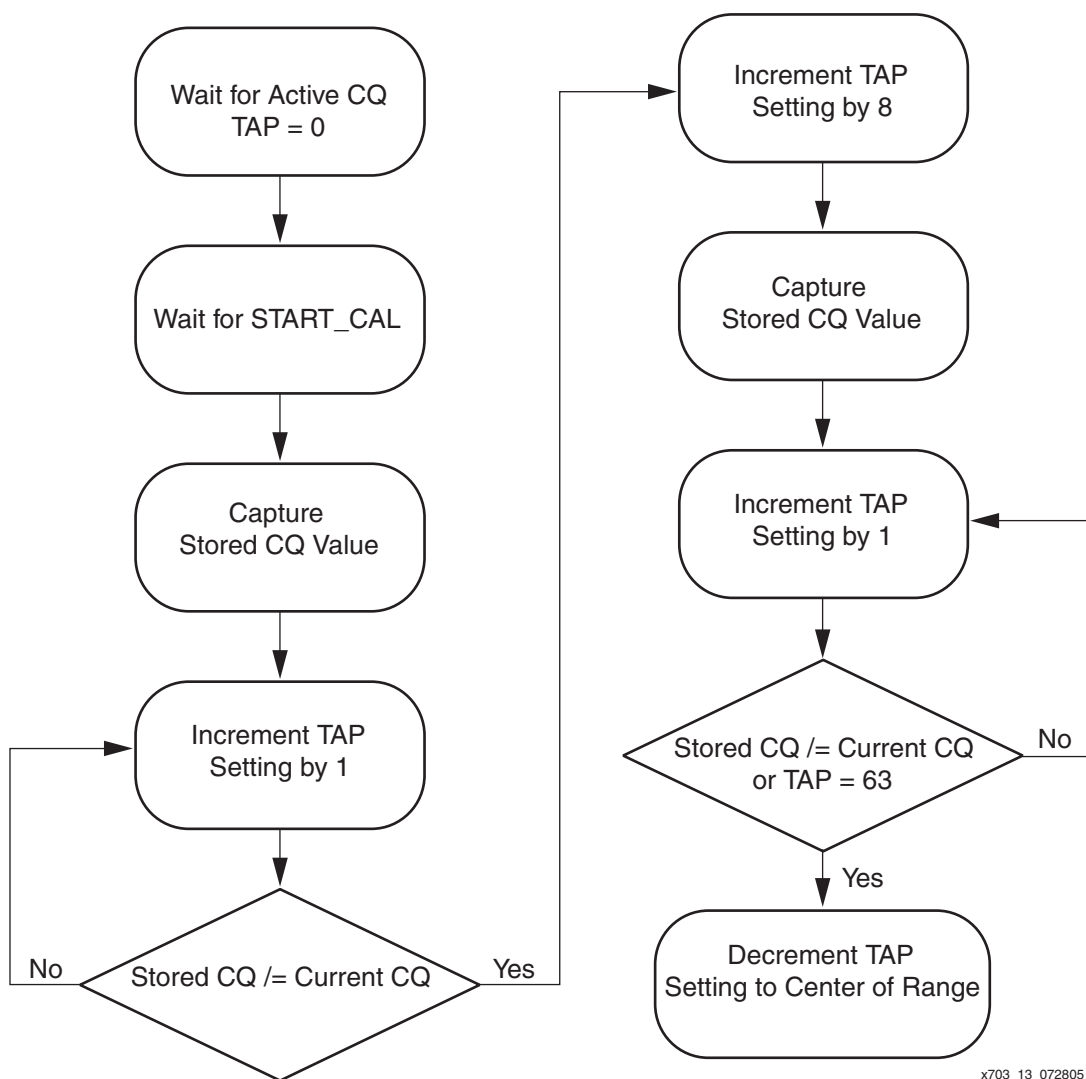
图 12: 使用 tap 延迟将 QDR_Q 输入与 USER_CLK0 对齐

图 13 显示延迟校准状态机所采用的一种算法，用以根据检测到的 CQ 时钟边沿，为 QDR_Q 输入确定正确的 IDELAY tap 设置。起初，CQ 时钟和 QDR_Q 总线信号的 tap 延迟值都设置为零。状态机起初寻找 1024 个连续时钟边沿，以等候有效的 CQ 时钟出现。这保证了边沿检测培训顺序拥有稳定的时钟信号。

随后，此状态机等候主读 / 写状态机启动延迟校准。CQ 校准输入值随后于初始 tap 延迟设置采集到。此保存 CQ 值（1 或 0）为边沿检测比较确立了基线。此时，状态机开始为 CQ 时钟和 QDR_Q 总线信号增加 tap 延迟设置。在每个新的 tap 设置，在 IDDR 寄存器中采集的自由运行的 CQ 时钟值与最初的保存 CQ 值相比较，以确定 CQ 信号一个边沿的位置。此过程一直继续下去，直到当前 CQ 值改变为保存 CQ 值的相反状态，表明边沿已检测到。通过该过程确定的最终 tap 延迟值成为延迟窗口的一个终点。

CQ 时钟下一边沿的位置使用类似方法确定。起初，tap 延迟设置从当前点以八为增量增加，以远离 CQ 转换区可靠地采集新的保存 CQ 值。然后，状态机开始增加 tap 延迟设置，并检测采集到的 CQ 值之状态与保存 CQ 值相比较是否有所变化，若有变化，即表明另一边沿已检测到。此过程一直继续，直到找到下一个边沿位置，或 tap 延迟值达到最大值 63。此最终 tap 点确定延迟窗口的另一个终点。

延迟校准状态机的最后操作是减少 tap 延迟设置，以使 CQ 和 QDR_Q 信号返回延迟窗口范围中点，从而将 USER_CLK0 信号在进入 Virtex-4 器件的读数据字之有效数据窗口内准确居中。因为 QDR_Q 信号的 tap 延迟设置总是与 CQ 输入设置步骤锁定，CQ 居中算法完成后，按照定义，QDR_Q 信号便也与 USER_CLK0 的边沿中心对齐。



x703_13_072805

图 13: QDR_Q 数据有效窗口中用于居中 USER_CLK0 的边沿检测算法

读 FIFO 选通脉冲生成

如“设计概述”部分所述，有两种方法可针对用户接口模块的读数据 FIFO 生成写使能信号。无论起初选择哪种方法，最好都在 PCB 上包含“方法 1”中提到的电路板布线。这样，就可以在原型测试期间灵活地评估两种方法的效能。

“方法 1”要求使用 qdrll_mem_ctrl1.v / .vhd 和 read_burst1.v / .vhd 文件，而方法 2 需使用 qdrll_mem_ctrl2.v / .vhd 和 read_burst2.v / .vhd 文件。其他所有 HDL 文件在两种方法中均需使用。

方法 1

本参考设计的 RD_STB_n_out 输出与 QDR_R_n 读选通脉冲信号完全相同。RD_STB_n_out 信号需沿着电路板布线回送，所经路程等于到外部存储器器件往返布线的长度。下列方程式表示回送电路板布线的延迟 ($t_{\text{DELAY_LOOP}}$)。

$$t_{\text{DELAY_LOOP}} = t_{\text{DELAY_TRACE1}} + t_{\text{DELAY_TRACE2}}$$

其中 $t_{\text{DELAY_TRACE1}}$ 代表从 Virtex-4 器件到 QDR II 器件的布线延迟（如写通路）， $t_{\text{DELAY_TRACE2}}$ 代表从 QDR II 器件返回 Virtex-4 器件的布线延迟（读通路）。

当回送布线通过 RD_STB_n_in 输入返回 Virtex-4 器件时，读选通脉冲已发生延迟，且与读通路上从存储器返回的数据值保持同步。此延迟选通脉冲用作读数据的写使能信号参考设计内的 FIFO。

图 14 表明 FIFO 读选通脉冲逻辑的实现。RD_STB_n_in 信号通过与 CQ 和 QDR_Q 信号具有相同 tap 延迟设置的 IDELAY 元件进行传输。IDDR 器件通过 USER_CLK0 时钟采集选通脉冲，信号中添加流水线级数 (pipeline stages)，以在相应时钟周期内将读数据字写用户接口 FIFO。或多或少的流水线级数会被自动添加，以正确对齐读数据字与高、低频字 FIFO 输入，添加的数量多少取决于 USER_CLK0 最终居中于 CQ 时钟周期的高频相位还是低频相位。

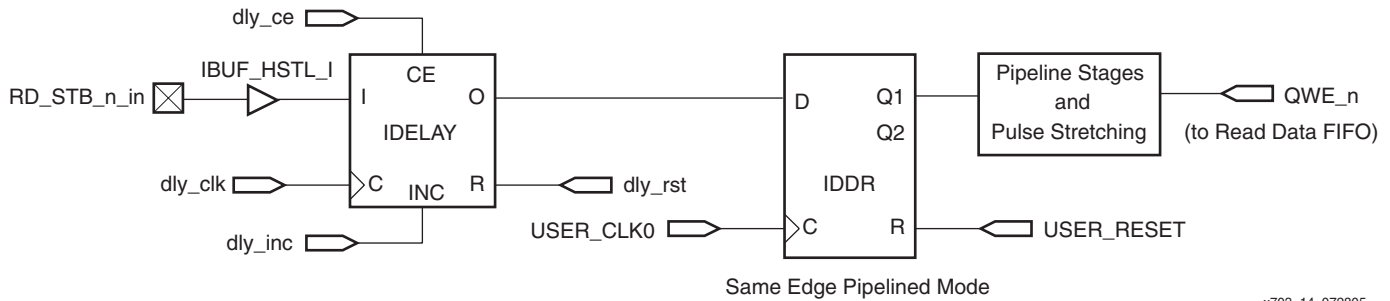


图 14: 读 FIFO 选通脉冲逻辑

方法 2

作为方法 1 的替代方法，只需利用 SRL16 移位寄存器功能将内部读选通脉冲适当延迟数个时钟周期，即可生成读数据 FIFO 的写使能。这样，写使能就能与从 QDR II 器件中返回的读数据 (Q) 保持同步。这种方法之所以可行，是因为使用了“直接时钟”机制，在这种机制中，Virtex-4 器件内部一切均与 USER_CLK0 同步。

插入读选通脉冲通路中的时钟周期延迟之数量取决于 RD_FIFO_DELAY 值，此值位于 qdrll_mem_ctrl2.v / .vhd 文件顶部附近。默认情况下，此值设置为 2 (二进制 0010)，对大多数系统应为适宜。不过，该值需根据需要调整大小，以使写使能选通脉冲与进入 FIFO 的读数据对齐。

如果得当实现，方法 1 更佳，因为它可以在读数据通路 (Q) 信号进入 Virtex-4 器件时“跟踪”其时序。不过，方法 2 也可以有效实现，而且在某些情况下可能会获得更高的运行性能。这两种方法都可以实现“设计概述”部分详述的性能目标。

参考设计

ISE 设计示例

与本应用指南相关的参考设计文件包含一个完整的顶层 ISE 设计示例，可用于演示或开发。该设计实现了两个 36 位 QDR II 存储控制器，这样，便获得连接到一对 QDR II 器件的一个 72 位接口。一个 36 位接口使用“方法 1”实现，以生成读 FIFO 选通脉冲，另一个 36 位接口使用“方法 2”实现。如图 15 所示，本 ISE 设计示例包含主 QDR II 存储控制器内核 (qdrll_mem_ctrl1.v / .vhd 和 qdrll_mem_ctrl2.v / .vhd)、基于 DCM 的时钟生成器及硬件测试平台 (test_bench.v / .vhd)，这些器件都在顶层模块中例化，以提供演示本设计运行的足够的驱动力。

本 ISE 设计示例还包含将该设计应用于 ML461 Virtex-4 存储器接口板所需要的引脚锁定指令和时序限制，请访问 Xilinx 存储器资料库网站获取详情。网址为：

<http://www.xilinx.com/cn/memory>。

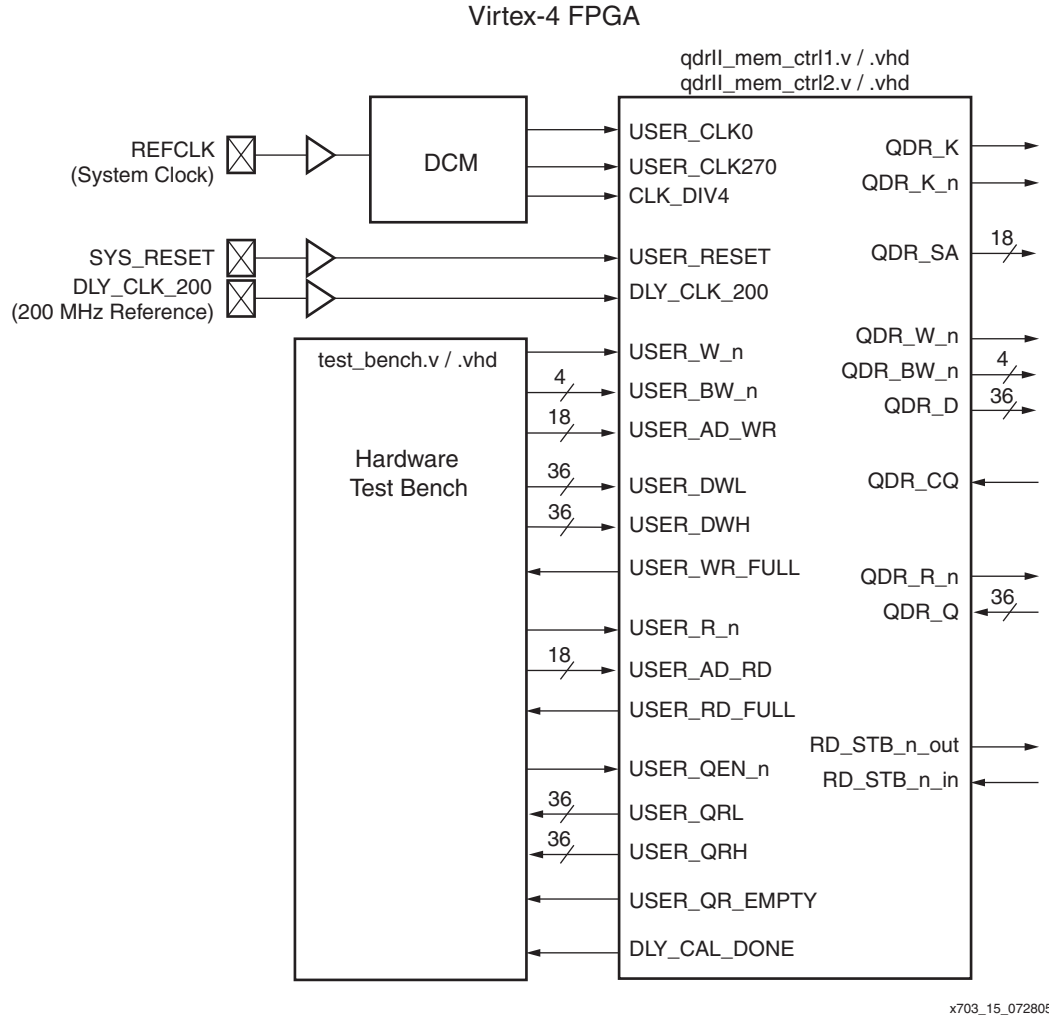


图 15: ISE 设计示例 (兼容 ML461 板)

测试平台及验证

本 Virtex- 4 QDR II 参考设计包括用于对设计进行功能性验证的 HDL 测试平台。所提供的测试平台已使用 Samsung 4 字突发 K7R323684M (1M x 36-bit) 或 Cypress 2 字突发 CY7C1314BV18 (512Kx36-bit) Verilog 器件模型进行了全面验证。ISE 设计示例中的内嵌式硬件测试平台也已通过仿真和硬件两种方式进行了全面验证。

而且, 图 15 所示的 ISE 设计示例已在 Virtex- 4 XC4VLX25ES 器件上使用 ML461 存储器接口板进行了验证, 结果符合“设计概述”部分所列的功能和性能要求。

板设计的考虑因素

尽管 Virtex- 4 系列产品提供了许多与 I/O 和时钟相关的高级功能, 大大简化了存储器接口设计, 但为了使接口可靠而高效, 仍需注意基本的板设计标准。

需特别强调的是, 读和写通路接口的源同步特性要求接口时钟、数据及控制信号具有相匹配的电路板布线长度。

例如, QDR II 器件输入信号 (QDR_K、QDR_K_n、QDR_W_n、QDR_R_n、QDR_SA、QDR_BW_n 和 QDR_D) 的布线长度必须完全匹配, 以将控制、地址及数据线接至具备充足建

立与保持余量的存储器器件。物理接口的实现可确保这些信号在离开 FPGA 器件输出时与 QDR_K 和 QDR_K_n 时钟边沿中心对齐。电路板布线必须确保这种关联状态能持续到存储器器件输出。

同样，QDR II 器件输出信号（QDR_Q、QDR_CQ）必须具备完全匹配的布线长度，以使这些信号在 Virtex- 4 器件的输入端实现边沿对齐。要实现直接时钟读数据采集方法，这一点非常关键。所有合理的板设计工具都可以在可接受的最小容错范围内与这些布线相匹配。

集成

将 Virtex-II QDR II 参考设计集成到完整的 FPGA 项目中比较简单。将 qdrII_mem_ctrl1.v / .vhd 或 qdrII_mem_ctrl2.v / .vhd 存储器控制器内核作为标准 HDL 模块例化到一个现有设计中。通常情况下，此内核模块会与完整设计的其余部分共享其他器件资源（包括 DCM 和全局时钟缓冲器 (global clock buffer)），这样可进一步减少专门用于存储器接口的资源。

如“读通路”部分所述，当 IDELAY 元件用在 I/O 模块内部时，IDELAYCTRL 基元也必须例化，并用 200-MHz 参考时钟控制，以校准 tap 延迟元件。在每个使用 IDELAY 元件的 I/O 组中，必须包含一个例化 IDELAYCTRL 基元。在本 ISE 设计示例中，有四个 I/O 组包含 IDELAY 元件，因此，在 read_burst1.v / .vhd 或 read_burst2.v / .vhd 模块中有四个 IDELAYCTRL 实例。这四个实例在 top.ucf 文件中用位置 (LOC) 约束锁定。为使 ISE 工具能正确利用 IDELAYCTRL 基元，这样做是必要的。有关 IDELAYCTRL 的更多详情，请参阅《Virtex- 4 用户指南》第 8 章“高级 SelectIO 逻辑资源”。

将 qdrII_mem_ctrl1.v / .vhd 或 qdrII_mem_ctrl2.v / .vhd 模块集成到新设计中时，read_burst1.v / .vhd 或 read_burst2.v / .vhd 模块中 IDELAYCTRL 实例的数量，以及对这些实例的 LOC 约束，应根据包含 IDELAY 元件的 I/O 组的数量和位置加以更改。来自多个 IDELAYCTRL 模块的 RDY 输出只需通过 OR 操作组合到一起。

FPGA 设计项目中可包含 QDR II 参考设计的多个实例，以创建多个独立的存储器接口或增加单个存储器接口的深度或宽度。

尽管本参考设计的设计初衷是最大程度提高性能、简化操作并提供简单明了的用户接口，但经过适当修改，现有设计也可以满足特定的应用需求。例如，去除用户接口 FIFO 可最大限度地降低读 / 写周期的延迟。进行其他配置亦有可能。

时序分析

本 Virtex- 4 QDR II 参考设计利用该器件的独特 I/O 和时钟功能，可最大限度地提高性能与时序余量，同时大大降低对详细布置和引脚分配分析的需求。

本部分针对地址 / 控制通路、写数据通路及读（或采集）数据通路给出一个时序分析示例。

地址 / 控制通路

先前已讨论过，读 / 写地址总线、字节写使能信号和读 / 写控制选通脉冲都与 USER_CLK270 时钟保持同步。这可以确保，就来自 USER_CLK0 的输入 QDR_K 和 QDR_K_n 时钟边沿而言，这些 SDR 信号具备对于存储器器件的充足建立与保持余量。

表 2 根据用速度级别为 -11 的一个 Virtex-4 器件实现的 250-MHz 4 字突发 QDR II 存储器器件接口，显示针对这些信号的一个时序分析示例。

表 2: 地址和控制信号时序分析

参数	值 (ps)	前沿 不确定度	后沿 不确定度	描述
T _{CLOCK}	4000	-	-	时钟周期为 250 MHz
T _{CLOCK_SKEW_FPGA}	±50	50	50	TRACE 分析所得时钟歪斜
T _{PACKAGE_SKEW}	±30	30	30	组内最大包歪斜
T _{SETUP}	500	500	0	存储器数据单上的建立时间
T _{HOLD}	500	0	500	存储器数据单上的保持时间
T _{PCB_LAYOUT_SKEW}	±50	50	50	电路板布线间的最大歪斜 (以预估匹配公差为依据)
T _{PHASE_OFFSET_ERROR_DCM}	±140	140	140	不同 DCM 输出间的最大偏移量
T _{JITTER}	±50	50	50	与 USER_CLK0 和 USER_CLK270 之间的差异相关联的抖动元件
总不确定度	-	820	820	
有效窗	2360	820	3180	最差窗 = 2360 ps

图 16 说明地址和控制信号时序余量。由于这些信号为 USER_CLK270 所引用，就 QDR_K 时钟边沿而言，后沿余量 (trailing edge margin) 大于前沿余量 (leading edge margin)。这样，允许使用的全局时钟缓冲器就会比较少，但仍能在前沿上提供充足的余量。

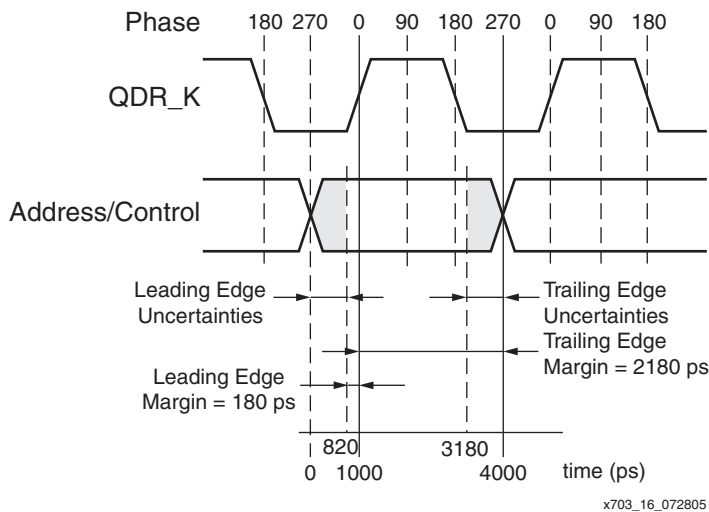


图 16: 地址和控制信号时序余量

写数据通路

写数据 (QDR_D) 也与 USER_CLK270 保持同步。不过，写数据字作为 DDR 值进行传输，因此在 QDR_K 和 QDR_K_n 上升沿均需具备充足的建立与保持余量。相应地，表 3 所示写通路的时序分析包括了存储器时钟的最大占空比失真。此分析亦适用于 250-MHz 4 字突发 QDR II 存储器器件和速度级别为 -11 的 Virtex-4 器件。

表 3: 写数据通路时序分析

参数	值 (ps)	前沿 不确定度	后沿 不确定度	描述
T _{CLOCK}	4000	-	-	时钟周期为 250 MHz
T _{CLOCK_PHASE}	2000	-	-	时钟相位 (时钟周期的 50%)
T _{DCD}	150	-	-	存储器时钟的占空比失真
T _{DATA_PERIOD}	1850	-	-	总数据周期 T _{CLOCK_PHASE} - T _{DCD}
T _{CLOCK_SKEW_FPGA}	50	50	50	TRACE 分析所得时钟歪斜
T _{PACKAGE_SKEW}	±30	30	30	组内最大包歪斜
T _{SETUP}	350	350	0	存储器数据单上的建立时间
T _{HOLD}	350	0	350	存储器数据单上的保持时间
T _{PCB_LAYOUT_SKEW}	±50	50	50	电路板布线间的最大歪斜 (以预估匹配公差为依据)
T _{PHASE_OFFSET_ERROR_DCM}	±140	140	140	不同 DCM 输出间的最大偏移量
T _{JITTER}	±50	50	50	与 USER_CLK0 和 USER_CLK270 之间的差异相关的抖动元件
总不确定度	-	670	670	最差条件下前后不确定度不可能同时发生
有效窗	510	670	1180	最差窗 = 510 ps

图 17 说明写数据通路时序余量。图中仅显示了有关 QDR_K 的分析。有关 QDR_K_n 的分析与之完全相同。

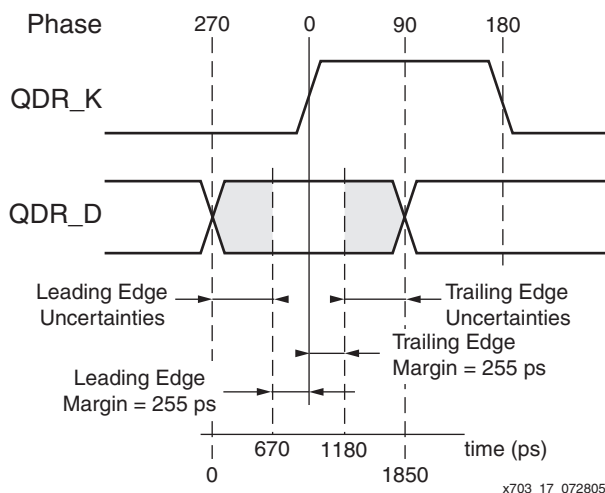


图 17: 写数据通路时序余量

读数据通路（数据采集）

读数据通路 (QDR_Q) 值将使用此前介绍过的直接时钟技术，直接采集到 USER_CLK0 时钟域中。因此，必须就 USER_CLK0 进行数据采集时序分析，还必须考虑 IDELAY tap 的延迟解析。另外，尽管来自存储器的 CQ 源同步回送时钟在边沿检测算法（中心对齐 QDR_Q 总线 and USER_CLK0）中只用作“培训”信号，但仍需考虑 CQ 时钟和 QDR_Q 总线间潜在的歪斜。表 4 显示通过接口将 QDR II 器件连接到速度级别为 -11 的 Virtex-4 FPGA 时的读时序分析。

表 4: 读数据通路时序分析

参数	值 (ps)	描述
T _{CLOCK}	4000	时钟周期为 250 MHz。
T _{CLOCK_PHASE}	2000	时钟相位（时钟周期的 50%）。
存储器不确定度		
T _{MEM_DCD}	150	接收时钟的占空比失真。
T _{CQ_TO_Q_SKEW}	600	存储器数据单上的 CQ-to-data 歪斜。
FPGA 不确定度		
T _{SAMP}	500	此参数涵盖了整个 PVT 中 Virtex-4 DDR 输入寄存器的总体抽样误差，包括 IOB 寄存器的建立与保持、时钟抖动及 150 ps 的 tap 不确定度。
T _{CLOCK_SKEW}	100	TRACE 分析所得时钟歪斜。
T _{PACKAGE_SKEW}	20	组内最大包歪斜。
T _{PCB_LAYOUT_SKEW}	50	电路板布线间的最大歪斜（以预估匹配公差为依据）。

表 4: 读数据通路时序分析 (续表)

参数	值 (ps)	描述
IDELAY tap 抖动	480	抖动由通过 IDELAY 的延迟数据造成。12 ps 的 IDELAYPAT_JIT 用作最差 tap 数，等于时钟周期的四分之三。
数据窗	100	

结论

本应用指南说明了利用 Virtex-4 器件实现 2 字或 4 字突发 QDR II SRAM 接口及其时序的详细信息。直接时钟方法的应用大大简化了 FPGA 内的读数据采集任务，且为当前及下一代 QDR II SRAM 存储器器件提供了高效而强大的可扩展存储器接口。

QDR-II SDRAM 接口的参考设计集成了 MIG 工具。此工具支持 Xilinx CORE Generator™ 软件。要获取此设计的最新版本，请通过 Xilinx 网站下载 IP 更新信息，网址为：

http://www.xilinx.com/cn/xlnx/xil_sw_updates_home.jsp

附录

QDR II SRAM 参考资料

- QDR 联盟：<http://www.qdrsram.com>
- QDR II SRAM 存储器器件供应商
 - ◆ Cypress 半导体：<http://www.cypress.com/>
 - ◆ Renesas 科技：<http://www.renesas.com/>
 - ◆ IDT 公司：<http://www.idt.com/>
 - ◆ Samsung 半导体：<http://www.samsung.com/>
 - ◆ NEC 公司：<http://www.necel.com/memory/en/index.html>

修订历史

下表说明此技术文档的修订历史。

日期	版本	修订
2004 年 9 月 10 日	1.0	Xilinx 最初版本。
2005 年 5 月 12 日	2.0	<ol style="list-style-type: none"> 1. 修改延迟校准状态机，以在四分之一速率时钟 (CLK_DIV4) 下运行。 2. 修订延迟校准状态机算法，以从零开始 IDELAY tap 计数。 3. 推出生成读 FIFO 写使能选通脉冲的两种方法。 4. 更新 ISE 设计示例，以显示两种读 FIFO 选通脉冲生成方法的实现。 5. 修改时序分析部分，对时序数量进行了更新。 6. 发布反映全部所列更改的新参考设计文件。

日期	版本	修订
2005 年 8 月 10 日	2.1	1. 增加 2 字突发存储器器件文档。 2. 增加图 2、图 6 和图 8。 3. 增加 2 字突发参考设计文件。
2006 年 4 月 11 日	2.2	更新参考设计链接。
2006 年 9 月 6 日	2.3	更新表 1 和表 4；删除了图 18；还进行了版面改动。