XAPP702 (v1.6) 2006 年 2 月 22 日

使用 Virtex–4 器件实现的 DDR2 的控制器

作者：Lakshmi Gopalakrishnan

提要

DDR2 SDRAM 器件提供了比 DDR SDRAM 指标所要求的更好的新功能，并允许 DDR2 器件在 400 Mb/s 或更高的数据率下运行。高数据率要求 FPGA 中的控制器和 I/O 具备更高的性能。控制器的运行须与存储器的操作速度同步，以获得所需带宽。

简介

本应用指南描述了在 Virtex™–4 器件中实现 267–MHz DDR2 控制器，并与 Micron DDR2 SDRAM 器件接口。本技术文档简要介绍了 DDR2 SDRAM 器件的功能，详细说明了与高速 DDR2 存储器接口时控制器的运行情况，并描述了此控制器后端用户接口。使用 Verilog 编写的参考设计可以从 Xilinx 网站下载。

DDR2 SDRAM 器件概述

概述部分介绍 DDR2 SDRAM 器件所具有的功能，作为下一代 DDR 器件与上一代之间的主要区别。

DDR2 SDRAM 器件采用 SSTL 1.8V I/O 标准，利用 DDR 架构实现高速运行。存储器使用此控制器提供的差分时钟运行。命令在时钟的每个正边沿寄存。双向数据选通脉冲 (DQS) 与接收端中的用于采样的数据一起传输。在读时 DQS 由 DDR2 SDRAM 器件传输，而在写时则由控制器传输。DQS 与用于读的数据边沿对齐，与用于写的数据中心对齐。

对 DDR2 SDRAM 器件的读和写访问为突发式；访问以激活命令寄存开始，然后是读或写命令。在激活命令下寄存的地址位用于选择要访问的组和行。在读或写命令下寄存的地址位用于为突发访问选择组和起始列位置。

DDR2 控制器设计包括一个用户后端接口，用于生成写地址、写数据和读地址。这些信息存储在四个异步的 FIFO 中，以实现后端和控制器模块间的地址和数据同步。根据 FIFO 中数据的可用性和命令逻辑块发出的命令，控制器会考虑存储器的计时要求，然后向存储器发出正确命令。

控制器发出的 DDR2 命令

表 1 列出了控制器发出的命令。这些命令是存储器使用下列控制信号探测到的：

- 行地址选择 ($\overline{RAS}$)
- 列地址选择 ($\overline{CAS}$)
- 写使能 ($\overline{WE}$) 信号
- 时钟使能 (CKE) 在整个器件配置过程中和器件配置后设定为 High
- 芯片选择 ($\overline{CS}$) 在整个器件运行过程中设定为 Low

表 1: **DDR2 命令**

选择	功能	行地址选择	列地址选择	写使能信号
1	加载模式 (Load Mode)	L	L	L
2	自动刷新 (Auto Refresh)	L	L	H
3	预充电 (Percharge)(1)	L	H	L
4	组激活 (Bank Activate)	L	H	H
5	写 (Write)	H	L	L
6	读 (Read)	H	L	H
7	空操作 / 空闲 (No Operation/IDLE)	H	H	H

注：

1. 地址信号 A10 在预充电所有组期间设定为 High，在单个组预充电期间设定为 Low。

模式寄存器

模式寄存器 (MR) 用于定义 DDR2 SDRAM 的具体运行模式，包括突发长度 (Burst Length)、突发类型、CAS 延迟 (CAS Latency) 和运行模式的选择。图 1 所示为控制器所使用的模式寄存器的功能。

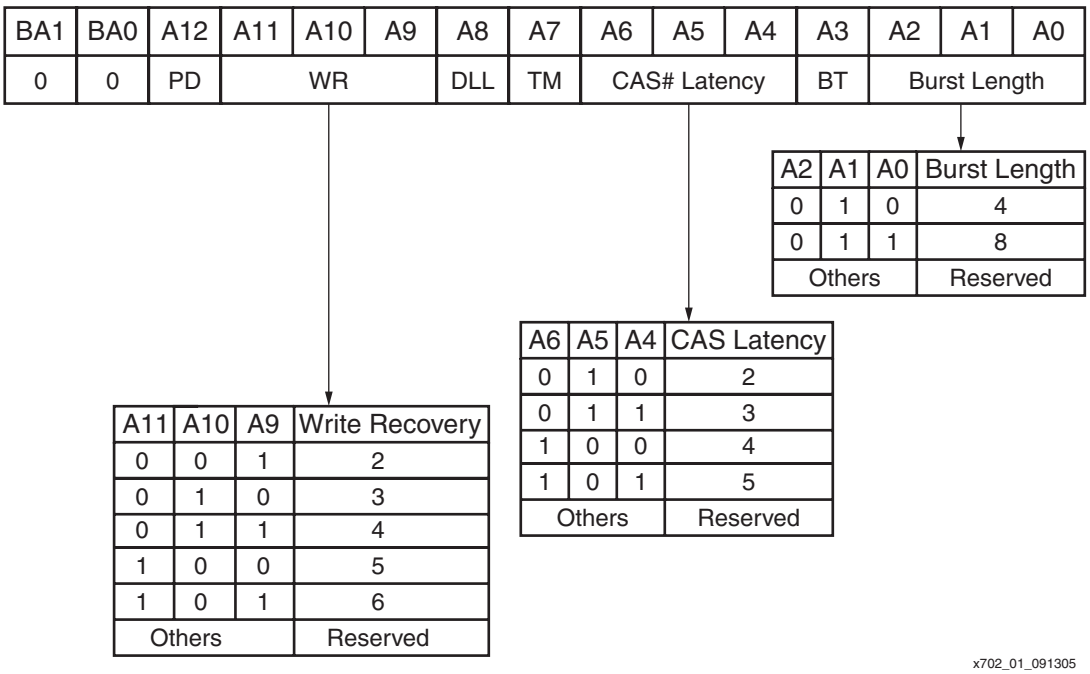


图 1: **模式寄存器**

组地址 BA1 和 BA0 用于选择模式寄存器。表 2 所示为组地址位的配置。

表 2: **组地址位配置**

BA1	BA0	模式寄存器
0	0	MR
0	1	EMR1
1	0	EMR2
1	1	EMR3

扩展模式寄存器

扩展模式寄存器 (EMR) 控制除 MR 支配的功能之外的其他功能。如 表 3 所示，这些附加功能有：DLL 使能 / 无效、输出驱动力、片上终端 (ODT)、Posted CAS 附加延迟 (AL)、片外驱动器阻抗校准 (OCD) 、 $\overline{\text{DQS}}$ 使能 / 无效、RDQS/$\overline{\text{RDQS}}$ 使能 / 无效、输出无效 / 使能。

表 3: **扩展模式寄存器**

BA1	BA0	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
0	1	输出	RDQS	$\overline{\text{DQS}}$	OCD 程序			R_{TT}	Posted CAS			R_{TT}	ODS	DLL

DDR2 SDRAM 参考设计使用一个为 0 的 AL。该设计中未采用 OCD。

扩展模式寄存器 2 (EMR2)

组地址设为 10，此处的 BA1 设为 High，BA0 设为 Low。地址位全部设为 Low。

扩展模式寄存器 3 (EMR3)

组地址位设为 11，此处的 BA1 和 BA0 设为 High。地址位全部设为 Low。

初始化顺序

此控制器状态机使用的初始化顺序遵循 DDR2 SDRAM 指标。该接口须满足存储器的电压要求。在功率稳定的情况下，在有效时钟 (CK, CK#) 后至少 200 µs 中，应用 NOP 或 DESELECT 命令，之后时钟使能 (CKE) 被设定为 HIGH。以下是初始化命令的发送顺序 :

1. CKE 被置位为 HIGH 之后至少 400 ns，发出 PRECHARGE ALL 命令。
2. 随后，设定 BA0 为 LOW、BA2 为 LOW、BA1 为 HIGH，发出 EMR(2) 命令。
3. 设定 BA0 为 HIGH、BA1 为 HIGH、BA2 为 LOW，发出 EMR(3) 命令。
4. 设定 A0、BA1 和 BA2 为 LOW，发出 EMR 命令，将存储器 DLL 置为使能。BA0 设定为 HIGH。
5. 设定 A8 为 HIGH，BA0、BA1 和 BA2 为 LOW，发出模式寄存器设置 (MRS) 命令，以复位存储器 DLL。
6. 发出 Precharge ALL 命令。
7. 发出两个自动刷新命令。
8. 最后，设定 A8 为 LOW，发出 MRS 命令。用运行参数对器件进行初始化时需要此命令 (无 DLL 复位)。
9. 完成步骤 8 后至少 200 个时钟，执行 OCD 校准 (片外驱动器阻抗调整)。如果未使用 OCD 校准，则发出一个带默认 OCD 命令 (A9=A8=A7=1) 的 EMR，随之发出带 OCD 命令的 EMR。

DDR2 SDRAM 器件现在可以开始正常运行了。

完成此系列操作后，控制器向 DDR2 SDRAM 发出虚拟读命令。在虚拟读期间，数据通路模块使用存储器发出的 DQS 来判断进入的 DQS 和内部系统时钟 CLK0 之间的关系。数据通路模块决定 DQS 与系统时钟对齐所需时间后，控制器开始读使能校准。

DQ/DQS 校准后，控制器发出模式写命令，以针对存储器器件写已知模式。随后，在已知模式写到的位置，读命令发出。读回的数据能帮助计算读命令和接收数据之间的时钟周期延迟情况。这一计算信息将被寄存，用于在正常读操作时生成读使能。

此读使能校准在 data_write 和 rd_data 模块中实现。在模式写过程中，控制器向存储器发出写命令，在 data_write 模块中生成已知写数据模式。在模式读过程中，控制器向相同位置发出读命令，生成 ctrl_rden 信号。该信号被延迟以对齐从存储器中收到的读数据。此延迟逻辑在 pattern_compare8 模块中实现。被延迟的 ctrl_rden 在正常读操作中使用。pattern_compare8 模块置位 COMP_DONE 信号，向控制器表明读使能校准已完成。控制器模块现在即可开始正常读 / 写操作了。

存储器接口发生器 (MIG) 工具（1.5 版）为每组生成一个读使能逻辑（Pattern_Compare8 模块）。如果组有四个数据位，MIG 工具将生成一个 Pattern_Compare4 模块。

预充电命令

预充电命令用于取消特定组中开放的行的活动状态。在预充电命令发出的一定时间 (t_{RP}) 后，此组在接下来的行激活命令中可以使用。输入 A10 确定组 (一个或全部) 是否需要预充电。

自动刷新命令

DDR2 器件需每 7.8 μs 刷新一次。要求自动刷新计数器的电路放置在控制器内。控制器使用 DCM 的 CLKDV 输出，为自动刷新计数器提供所需的低频时钟。要节省 DCM 的 CLKDV 输出使用的 BUFG，设计者可以使用 DCM 的高频 CLK0 输出或 DCM 的 CLK/4 输出（用于 IDELAY 电路）为刷新计数器计时。如果自动刷新电路的时钟改变，mem_interface_top_parameters_0.v 文件中的 max_ref_count 也应做相应改变。

auto_ref 信号标志出发出自动刷新命令之需。在控制器发出自动刷新命令之前，此信号一直为 High。在发出自动刷新命令前，控制器要完成当前活动组中的操作。

激活命令

需要先使用激活命令激活 DDR2 SDRAM 存储器内某个组中的某一行，然后才能向该组发送读或写命令。行被激活后，读或写命令就可以按照 t_{RCD} 指标发送到行。DDR2 SDRAM 器件也支持 Posted CAS AL。这些器件可以使读或写命令在 t_{RCD} 指标规定的时间之前发出，方法是利用 AL 时钟周期延迟内部器件对读或写命令的实际寄存时间。DDR2 控制器在激活命令之后发出读或写命令时，支持 CAS AL。

当控制器发现进入的地址所指向的组内行并非当前活动的行时，即会发出地址冲突信号。控制器随之发出预充电命令停用活动行，同时向新的行发出激活命令。

读命令

读命令用于发起对活动行的突发式读访问。BA0 和 BA1 上的值选定组地址，而 A_0 – A_1 上提供的地址输入选定起始列位置。读突发结束后，只要还未预充电，此行仍可用于后面的访问。

图 2 所示为 AL 为 0 的读命令的情况。因此，该情形中的读延迟与 CAS 延迟 (CL = 3) 相同。

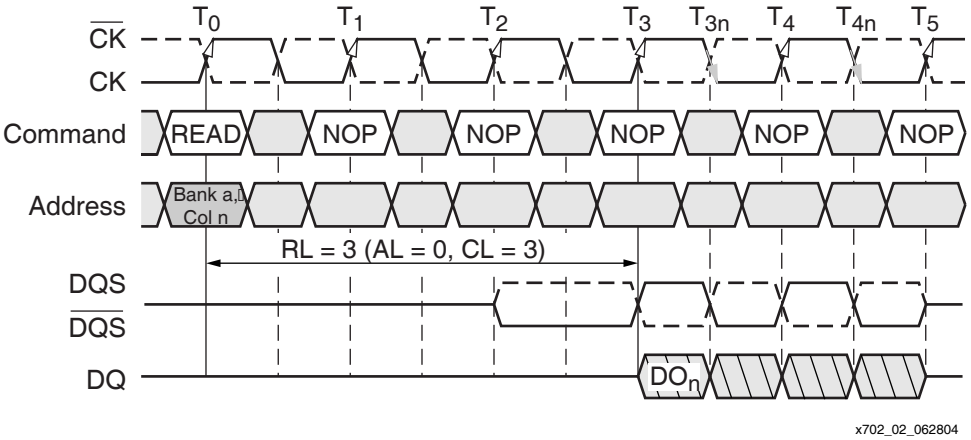


图 2: **读命令示例**

写命令

写命令用于发起对活动行的突发式访问。BA0 和 BA1 上的值选择组地址，而地址输入 A_0 – A_1 的值选择活动行内的起始列位置。DDR2 SDRAM 使用一个写延迟 (WL)，其值等于读延迟减去一个时钟周期。

写延迟 = 读延迟 –1 = (附加延迟 + CAS 延迟) – 1。

图 3 所示为写延迟为二的突发写情况。写命令和 DQS 信号的首个上升边沿之间的时间间隔由写延迟确定。

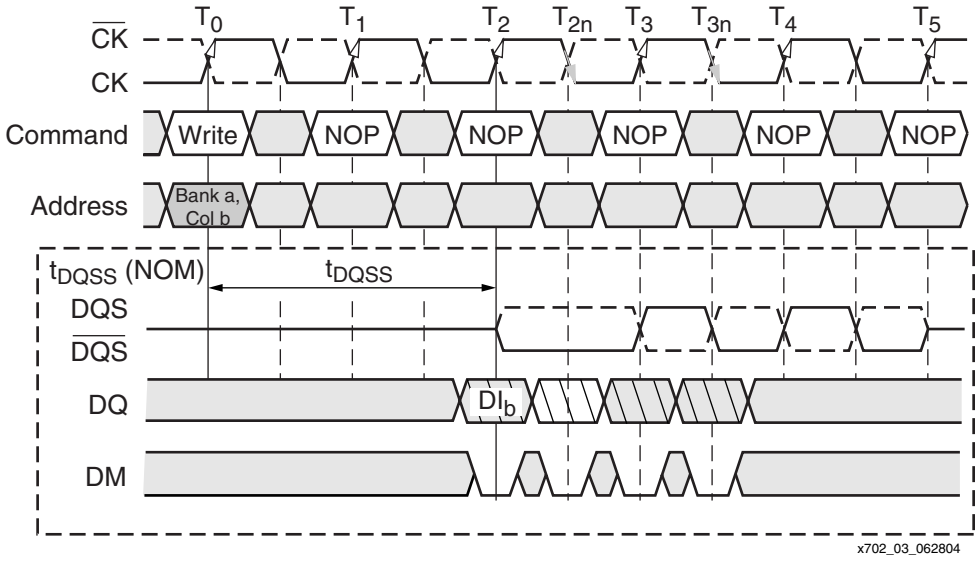
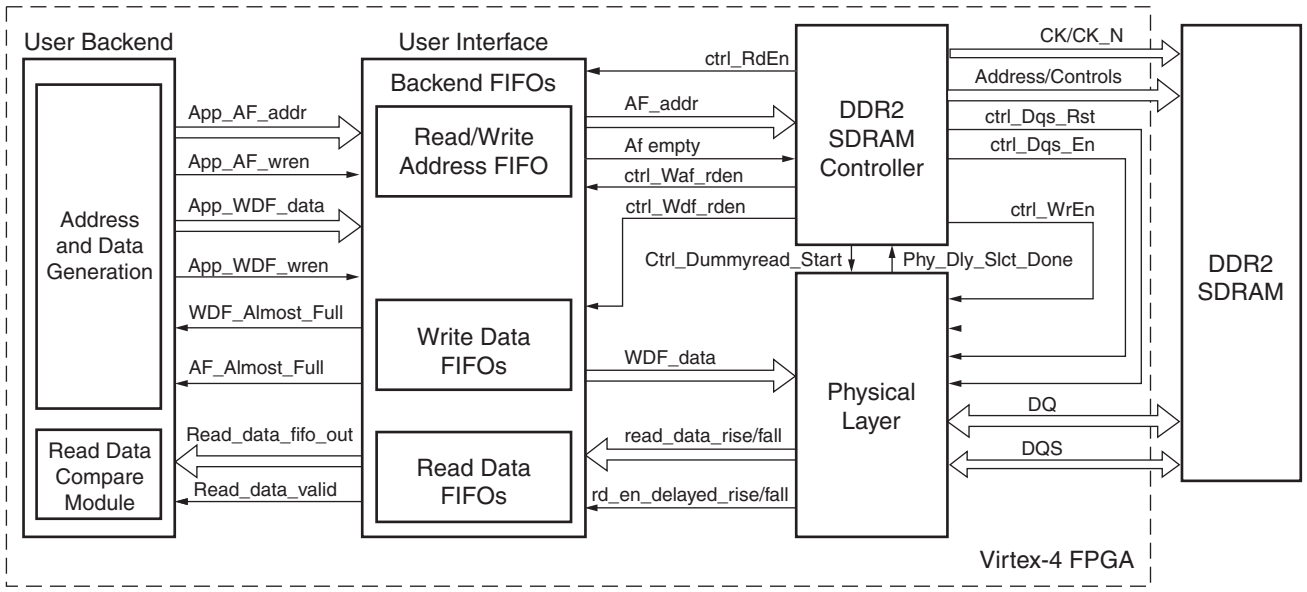


图 3: **写命令示例**

DDR2 接口实现

DDR2 SDRAM 接口包括图 4 所示各模块。



X702_04_021606

图 4: **DDR2 SDRAM 接口框图**

用户后端

此参考设计的后端提供了地址和数据模式，用于测试 DDR2 控制器的各设计方面。用户后端包括以下模块：后端状态机模块、读数据比较器模块及地址和数据生成器模块。

地址和数据生成模块生成各种写到存储器的地址和数据模式。地址值与动态命令请求预存储在用作 ROM 的一个 Block RAM 中。存储的地址和动态命令值已被选用来测试对 DDR2 SDRAM 器件中不同行和组的访问。数据模式生成器包括一个状态机，用于发出上升边沿数据模式。上升边沿数据与下降边沿数据相反。通过向地址 FIFO 发出读使能信号，后端状态机对用户后端应用进行模拟。

用户接口

后端用户接口包括三个 FIFO：写数据 FIFO、读与写地址 FIFO 和读数据 FIFO。前两个 FIFO 由用户后端模块访问。读数据 FIFO 由数据通路模块进行访问，并存储采集到的读数据。

用户到控制器的接口

表 4 列出用户接口与控制器之间的信号。

表 4: **用户接口和控制器之间的信号列表**

端口名称	端口宽度	端口描述	说明
Af_addr	36	用户接口中地址 FIFO 的输出。这些地址位的映射： • 存储器地址（CS、组、行、列）：[31:0] • 备用：[32] • 动态命令请求：[35:33]	监控 FIFO 满状态标志，将相应地址写地址 FIFO
Af_empty	1	用户接口地址 FIFO 空状态标志输出。在写数据 FIFO 满状态标志置位前，用户应用在此信号置位后可写地址 FIFO。	FIFO16 近空标志
ctrl_af_RdEn	1	用户接口中地址 FIFO 的读使能输入。	当控制器状态随动态命令请求发生变化呈写、读、加载模式寄存器、全部预充电、自动刷新或激活时，此信号保持一个时钟周期有效。图 5 表示突发长度为 8 时的时序波形，包含四个背靠背式写及随后的四个背靠背式读。
ctrl_wdf_RdEn	1	用户接口中写数据 FIFO 的读使能输入。 .	首次写状态后，控制器保持此信号在两个时钟周期内有效。突发长度为 4，此信号保持在两个时钟周期内有效，突发长度为 8 则保持四个时钟周期内有效。图 5 所示为时序波形。在发出写命令前，与写地址关联的数据 FIFO 中必须具备充足的数据，以满足对突发长度的要求。例如，如果数据总线为 64 位，突发长度为 4，在发出写命令前，用户应针对每个写地址向写数据 FIFO 中输入两个 128 位数据字。

存储器地址 (Waf_addr) 包括深的存储器接口的列地址、行地址、组地址及芯片选择宽度。

列地址

[col_ap_width – 1:0]

行地址

[col_ap_width + row_address – 1:col_ap_width]

组地址

[col_ap_width + row_address + bank_address – 1:col_ap_width + row_address]

芯片选择

[col_ap_width + row_address + bank_address + chip_address – 1:col_ap_width + row_address + bank_address]

动态命令请求

对于控制器的普通操作来说，表 5 所列命令不是必需的命令。如果实际应用需要，用户可以选择请求这些命令。

表 5: **可选命令**

命令	描述
000	加载模式寄存器 (Load Mode Register)
001	自动刷新 (Auto Refresh)
010	全部预充电 (Percharge All)
011	激活 (Active)
100	写 (Write)
101	读 (Read)

图 5 表示四个连续写随之以四个连续读，突发长度为 8。表 6 列出图 5 所用到的状态信号的值。

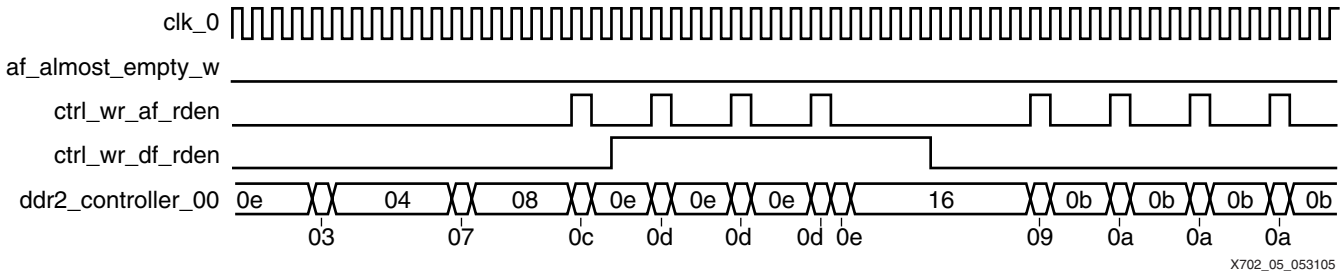


图 5: **连续读随之以连续写，突发长度为 8**

表 6: **图 5 所用到的状态信号的值**

状态信号值	描述
0c	首次写
0d	突发写
09	首次读
0a	突发读
0e	写等待
03	预充电
04	预充电等待
07	激活
08	激活等待
16	写至读
0b	读等待

控制器到物理层接口

表 7 列出控制器与物理层之间的信号。

表 7: **控制器和物理层之间的信号**

端口名称	端口宽度	端口描述	说明
ctrl_Dummyread_Start	1	从控制器到物理层的输出。置位后，物理层会在存储器初始化后开始选通脉冲和数据校准。	在读选通脉冲开始转入虚拟读状态后就会置位此信号。此信号会在 phy_Dly_Slct_Done 信号置位后变为无效。
phy_Dly_Slct_Done	1	物理层到控制器的输入，表明校准完毕。	数据位被延迟而与 FPGA 全局时钟中心对齐后，会置位此信号。phy_Dly_Slct_Done 信号置位后，ctrl_Dummyread_Start 信号就会变为无效。此信号被置位后开始正常运行。
ctrl_Dqs_Rst	1	针对写选通脉冲前导位从控制器到物理层的输出。	一次写期间，此信号会保持一个时钟周期有效。CAS 延迟与 AL 值将决定首次写状态之后经过多少个时钟周期，此信号被置位。图 6 针对突发长度为 8 的四个背靠背式写，显示了当 CAS 延迟为 3，AL 为 0 时此信号的时序波形。
ctrl_Dqs_En	1	针对写选通脉冲从控制器到物理层的输出。	写期间突发长度为四时，此信号将保持三个时钟周期有效，突发长度为 8 时是保持五个时钟周期有效。CAS 延迟和 AL 值将决定首次写或突发写状态之后经过多少个时钟周期，此信号被置位。图 6 针对突发长度为 8 的四个背靠背式写，显示了当 CAS 延迟为 3，AL 为 0 时此信号的时序波形。
ctrl_WrEn	1	针对写数据三态控制从控制器到物理层的输出。	写期间突发长度为 4 时，此信号保持两个时钟周期有效，突发长度为 8 时保持四个时钟周期有效。CAS 延迟与 AL 值决定首次写或突发写状态之后经过多少个时钟周期，此信号被置位。图 6 针对突发长度为 8 的四个背靠背式写，显示了当 CAS 延迟为 3，AL 为 0 时此信号的时序波形。

图 6 所示为从控制器到物理层控制信号的时序波形。表 8 列出图 6 所使用的状态信号的值。

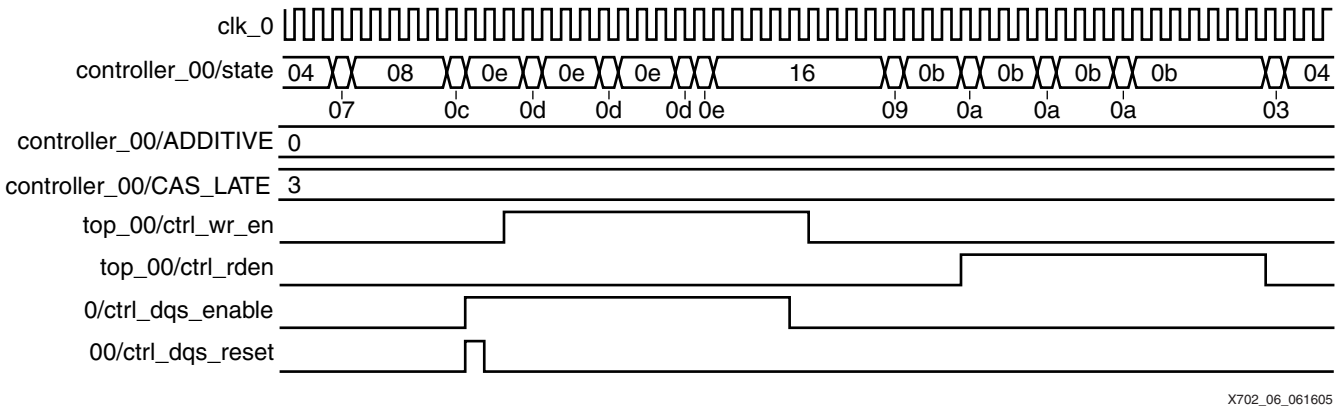


图 6: **从控制器到物理层的控制信号时序波形**

表 8: **图 6 所用的状态信号的值**

状态信号值	描述
0c	首次写
0d	突发写
09	首次读
0a	突发读
0e	写等待
03	预充电
04	预充电等待
07	激活
08	激活等待
16	写至读
0b	读等待

控制器实现

控制器状态机按正确顺序发布命令，同时会考虑存储器的时序要求。图 7 表示控制器状态机中的不同阶段。

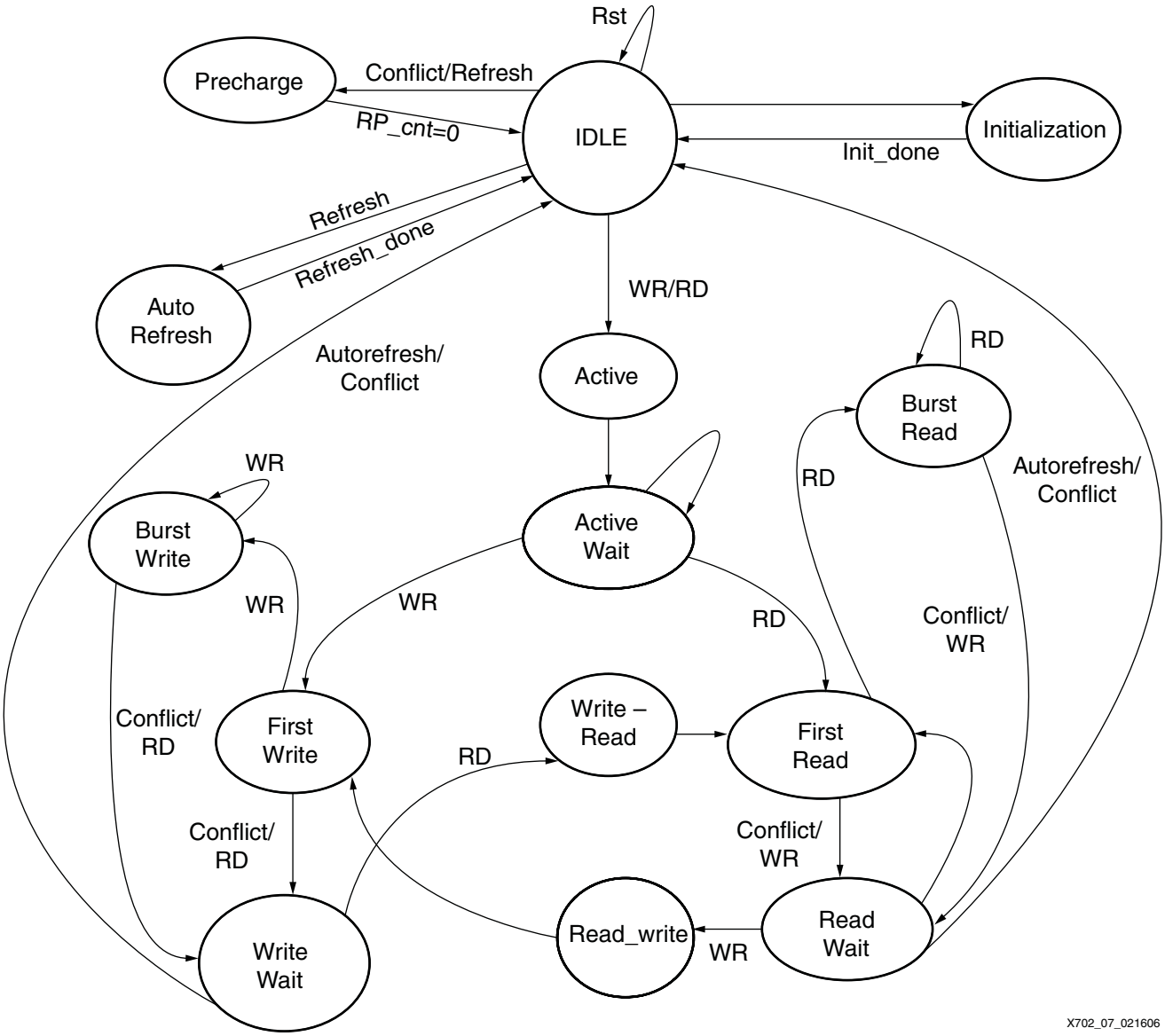


图 7: **DDR2 控制器状态机**

控制器向存储器发出命令前：

1. 控制器向读 / 写地址 FIFO 发出读使能信号。
2. 如果所有组都已预充电，控制器会激活相应组中的某行；或将这些组、行地址与已处于活动状态的组、行地址进行比较。如果存在冲突，在进入读 / 写状态前，控制器会预充电活动组，然后发出一个激活命令。
3. 在写状态下，如果控制器检测到读命令，就会等 write_to_read 时间结束后再发出读命令。同样，在读状态中，发现来自命令逻辑模块的写命令后，控制器会等待 read_to_write 的时间，然后发送写命令。
4. 如果后端用户应用发出了动态命令请求，如预充电、自动刷新、激活或加载模式寄存器，控制器就会发出预充电命令。
5. 在发送到 DDR2 存储器前，命令已经过流水处理而与地址信号同步。

设计层级

图 8 所示为始于一个顶层模块 ddr2 的设计层级。

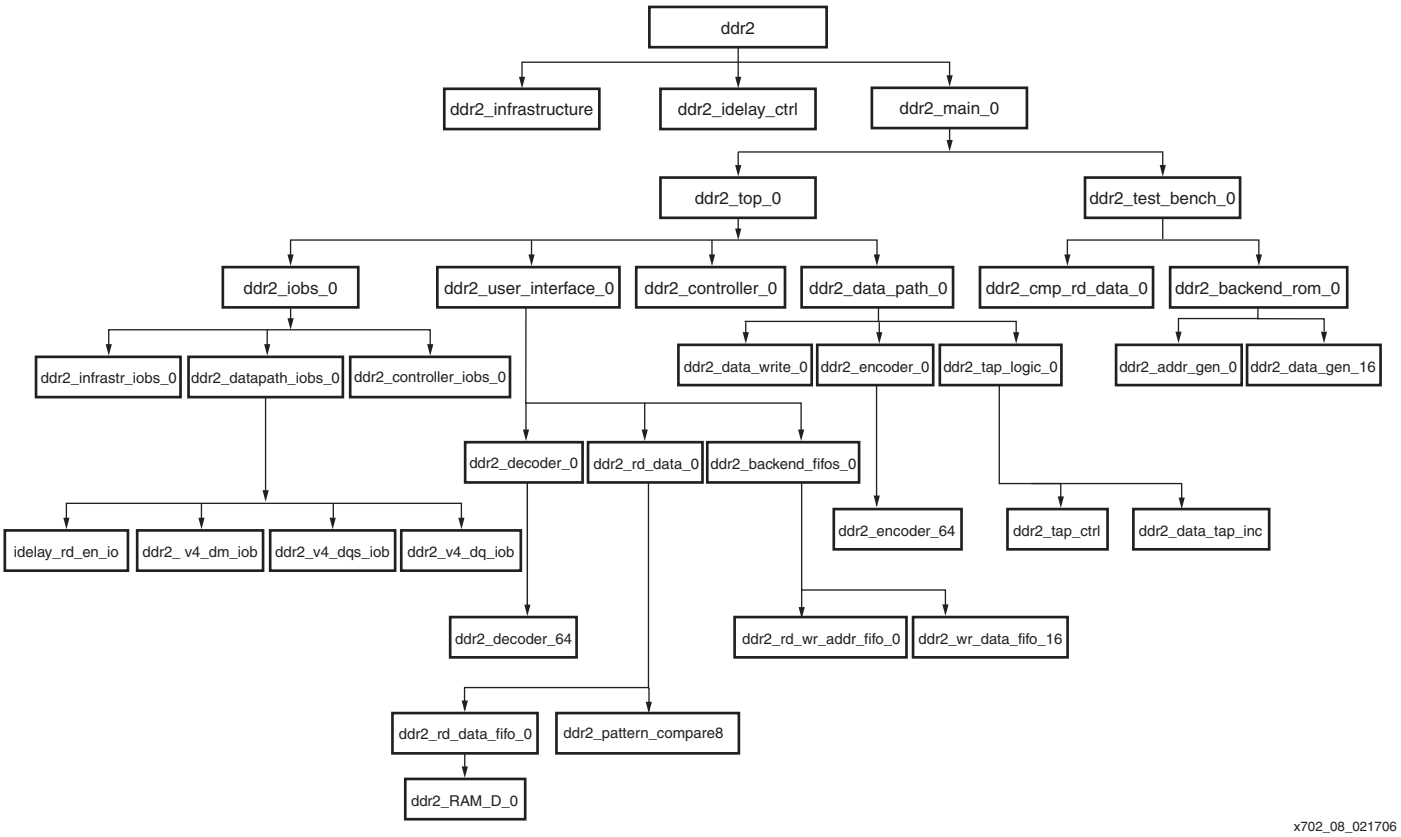


图 8: **设计层级**

参考设计

64 位 DDR2 SDRAM 接口的资源利用情况（包括可综合测试平台）列于表 9 中。

表 9: **频率为 267 MHz 时 64 位 DDR2 SDRAM 存储器接口的资源利用情况**

资源	利用率	说明
Slice	1791	包括可综合测试平台及后端用户逻辑接口。
全局时钟缓冲器 (BUFG)	5	包括一个用于 IDELAY 的 BUFG 参考时钟。
DCM	1	如果 FPGA 中的其余设计与存储器接口具有相同的频率，那么整个 FPGA 设计只需一个 DCM。
Block RAM (FIFO16)	5	五个 FIFO16 全部用来生成后端写数据和读 / 写地址。

此 DDR2 SDRAM 存储控制器参考设计采用了直接时钟数据采集技术，并与 MIG 工具集成。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，网址为：

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

结论

在本参考设计中，通过利用 Virtex–4 DCM、IOB 及差分时钟树，DDR2 控制器可按所需速度运行。

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 6 月 13 日	1.1	修改 “DDR2 接口实现 ” 部分。
2005 年 7 月 15 日	1.2	修改表 4 中的端口名称，更改了 “ 用户到控制器的接口 ” 中的内容，并对图 4 和图 8 作修改。
2005 年 7 月 29 日	1.3	修改图 4 并更新参考设计文件。
2005 年 9 月 14 日	1.4	修改图 1 并更新 “ 参考设计 ” 部分。
2005 年 11 月 18 日	1.5	修改 “ 自动刷新命令 ” 部分。
2006 年 2 月 22 日	1.6	更新图 4、图 7 和图 8，并更新表 4、表 6、表 7 和表 8。