

Cyclone® IV 器件具有嵌入式存储器结构，满足了 Altera® Cyclone IV 器件设计对片上存储器的需求。嵌入式存储器结构由一列列 M9K 存储器模块组成，通过对这些 M9K 存储器模块进行配置，可以实现各种存储器功能，例如：RAM、移位寄存器、ROM 以及 FIFO 缓冲器。

本应用笔记包含以下几部分内容：

- “ 存储器模式 ”（第 3-7 页）
- “ 时钟模式 ”（第 3-14 页）
- “ 设计考量 ”（第 3-15 页）

概述

M9K 存储器模块支持以下特性：

- 每模块 8,192 个存储器位（包括奇偶校验位，每模块共 9,216 位）
- 用于每一个端口的独立读使能（rden）与写使能（wren）信号
- Packed 模式，该模式下 M9K 存储器模块被分成两个 4.5 K 单端口 RAM
- 可变端口配置
- 单端口与简单双端口模式，支持所有端口宽度
- 真双端口（一个读和一个写，两个读，或者两个写）操作
- 字节使能，实现写入期间的数据输入屏蔽
- 用于每一个端口（端口 A 和 B）的时钟使能控制信号
- 初始化文件，在 RAM 和 ROM 模式下预加载存储器中的数据

表 3-1 列出了 M9K 存储器所支持的特性。

表 3-1. M9K 存储器特性汇总

特性	M9K 模块
配置（深度 × 宽度）	8192 × 1
	4096 × 2
	2048 × 4
	1024 × 8
	1024 × 9
	512 × 16
	512 × 18
	256 × 32
	256 × 36
奇偶校验位	✓
字节使能	✓
Packed 模式	✓
地址时钟使能	✓
单端口模式	✓
简单双端口模式	✓
真双端口模式	✓
嵌入式移位寄存器模式 (注释 1)	✓
ROM 模式	✓
FIFO 缓冲器 (注释 1)	✓
简单双端口混合位宽支持	✓
真双端口混合位宽支持 (注释 2)	✓
存储器初始化文件 (.mif)	✓
混合时钟模式	✓
上电条件	输出端清零
寄存器异步清零	仅限读地址寄存器和输出寄存器
锁存器异步清零	仅限输出锁存器
读或写操作触发	读写：时钟上升沿
相同端口 read-during-write	输出端设置为 Old Data 或者 New Data
混合端口 read-during-write	输出端设置为 Old Data 或者 Don' t Care

表 3-1 注释：

- (1) 需要外部逻辑单元 (LE) 的 FIFO 缓冲器和嵌入式移位寄存器，以实现控制逻辑。
- (2) ×32 和 ×36 的位宽模式不可用。



要了解关于 Cyclone IV 器件的 M9K 存储器模块数量的详细信息，请参考 *Cyclone IV 器件手册* 第 1 卷中的 *Cyclone IV 器件系列概述* 章节。

控制信号

时钟使能控制信号对进入输入与输出寄存器的时钟以及整个 M9K 存储器模块进行控制。该信号将时钟禁用，使 M9K 存储器模块侦测不到任何的时钟边沿，从而不会执行任何操作。

rden 与 wren 控制信号控制 M9K 存储器模块的每一个端口上的读写操作。当不需要操作时，您可以分别将 rden 或者 wren 信号禁用，从而降低功耗。

奇偶校验位支持

通过奇偶校验位以及内部逻辑资源，用于错误检测的奇偶校验检查是可以实现的。Cyclone IV 器件 M9K 存储器模块支持每一个存储字节含有一个奇偶校验位，您可以将该位用作奇偶校验位或者额外的数据位。实际上，该位不执行奇偶校验功能。

字节使能支持

Cyclone IV 器件 M9K 存储器模块支持字节使能，该功能屏蔽了输入数据，这样仅写入数据中的指定字节。未被写入的字节保留之前写入的值。wren 信号以及字节使能 (byteena) 信号一起控制 RAM 模块的写操作。byteena 信号默认为高电平（使能的），这种情况下写操作仅由 wren 信号来控制。byteena 寄存器的清零端口是不存在的。当写端口具有 ×16，×18，×32，或者 ×36 位数据位宽时，M9K 模块将支持字节使能。

字节使能以独热码的形式运行，并且 byteena 的最低位对应于数据总线的最低字节。例如，若 byteena = 01，并且在 ×18 模式下使用一个 RAM 模块，则 data[8..0] 被使能，而 data[17..9] 被禁用。同样地，若 byteena = 11，则 data[8..0] 以及 data[17..9] 均被使能。字节使能为高电平有效。

表 3-2 列出了字节选择。

表 3-2. Cyclone IV 器件 M9K 模块的 byteena 模块 (注释 1)

byteena[3..0]	受影响的字节			
	datain ×16	datain ×18	datain ×32	datain ×36
[0] = 1	[7..0]	[8..0]	[7..0]	[8..0]
[1] = 1	[15..8]	[17..9]	[15..8]	[17..9]
[2] = 1	—	—	[23..16]	[26..18]
[3] = 1	—	—	[31..24]	[35..27]

表 3-2 注释：
(1) 可能存在字节使能的任意组合。

图 3-1 显示了 wren 和 byteena 信号是如何控制 RAM 操作的。

图 3-1. Cyclone IV 器件 byteena 功能波形图 (注释 1)

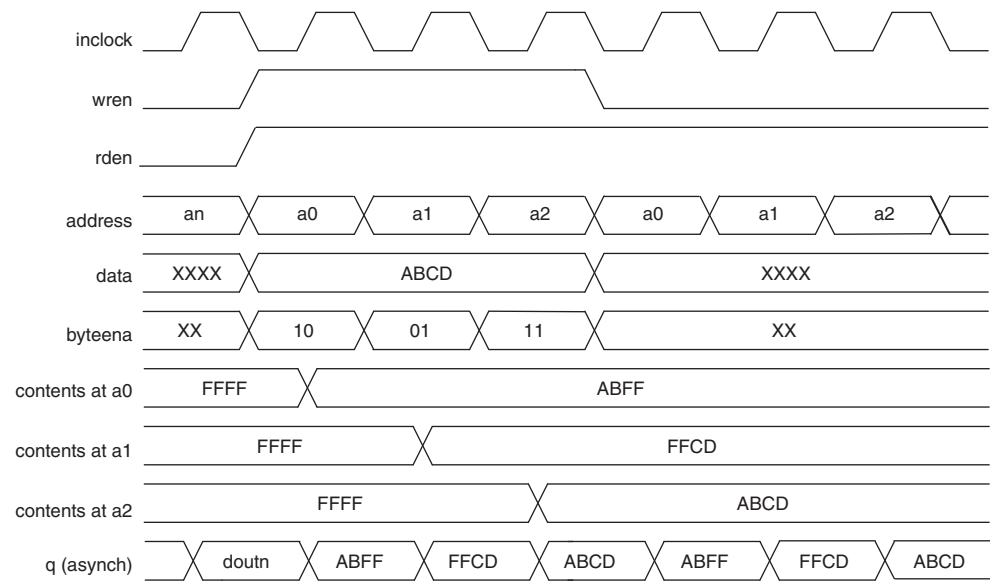



图 3-1 注释:

(1) 对于这一功能波形, 选择 **New Data** 模式。

当一个 byteena 位在写周期中被置低时, 存储器中的旧数据会出现在相应的数据字节输出端。当一个 byteena 位在写周期中被置位时, 相应的数据字节输出取决于 Quartus® II 中所选择的设置。该设置既可以是在相应位置上新写入的数据, 也可以是原有的旧数据。

 当单独的 M9K 存储器模块的 PortA 以及 PortB 数据位宽是 8 位或者 9 位的倍数时, 字节使能仅适用于真双端口 (True Dual-Port) 存储器配置。

Packed 模式支持

Cyclone IV 器件 M9K 存储器模块支持 packed 模式。在下列条件下, 您可以在单一模块中实现两个单端口存储器模块:

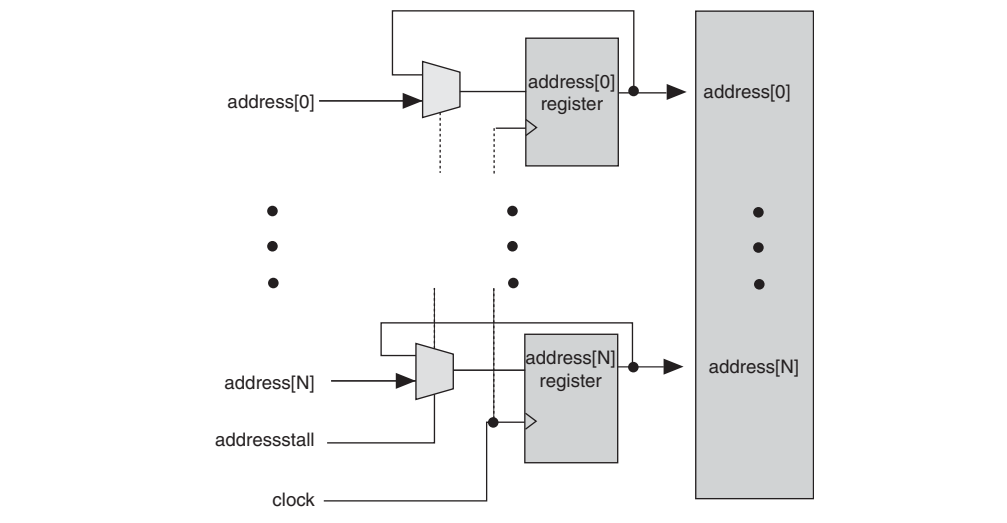
- 每一个独立模块的容量均小于或等于 M9K 模块容量的一半。每一个独立模块的最大数据宽度为 18 位宽。
- 每一个单端口存储器模块均被配置成单时钟模式。要了解关于 packed 模式支持的详细信息, 请参考第 3-8 页的 “**单端口模式**” 和第 3-15 页的 “**单时钟模式**”。

地址时钟使能支持

Cyclone IV 器件 M9K 存储器模块支持低电平有效地址时钟使能，只要 `addressstall` 信号为高电平 (`addressstall = '1'`)，低电平有效地址时钟使能就会保持之前的地址值。当在双端口模式下配置 M9K 存储器模块时，每个端口都有各自独立的地址时钟使能。

图 3-2 显示了地址时钟使能结构图。通过使用多路复用器，地址寄存器的输出将送回至它的输入端。多路复用器输出是由地址时钟使能 (`addressstall`) 信号选择的。

图 3-2. Cyclone IV 器件地址时钟使能结构图



地址时钟使能通常在高速缓存错失期间用于增进高速缓存存储器应用的有效性。地址时钟使能的默认值为低电平。

图 3-3 和图 3-4 分别显示了读写周期过程中的地址时钟使能波形。

图 3-3. Cyclone IV 器件地址时钟使能在读周期中的波形图

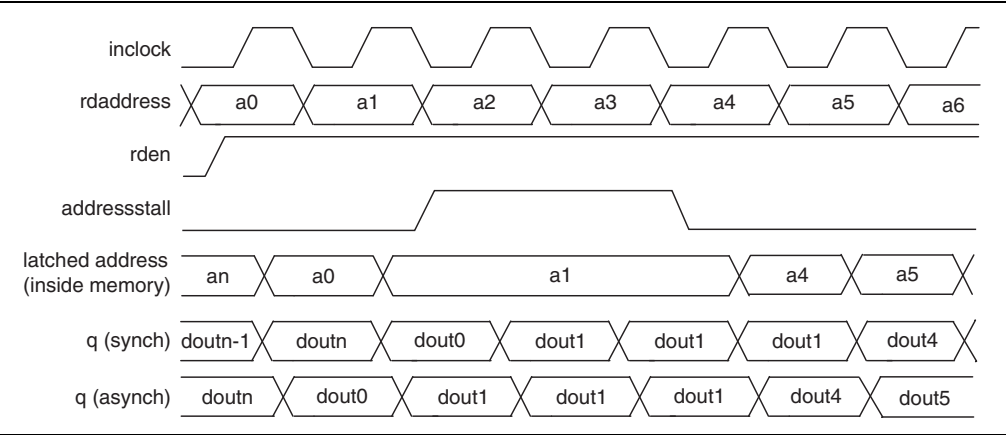
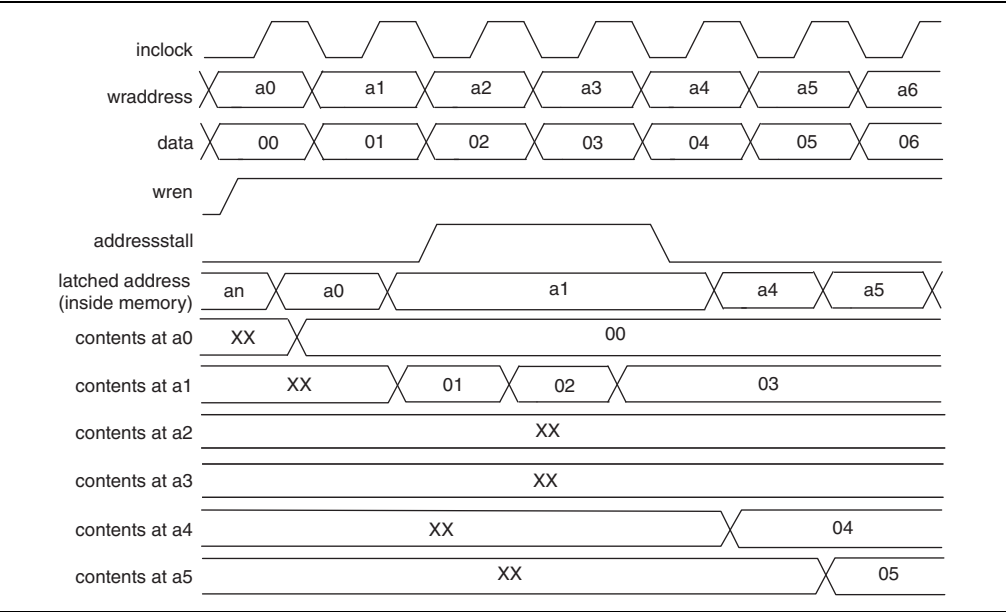


图 3-4. Cyclone IV 器件地址时钟使能在写周期中的波形图



混合位宽支持

M9K 存储器模块支持混合数据位宽。当使用简单双端口、真双端口、或者 FIFO 模式时，混合位宽支持实现了读写不同的数据位宽到 M9K 存储器模块。要了解关于每一个存储器模式所支持的不同位宽的详细信息，请参考第 3-7 页的“[存储器模式](#)”。

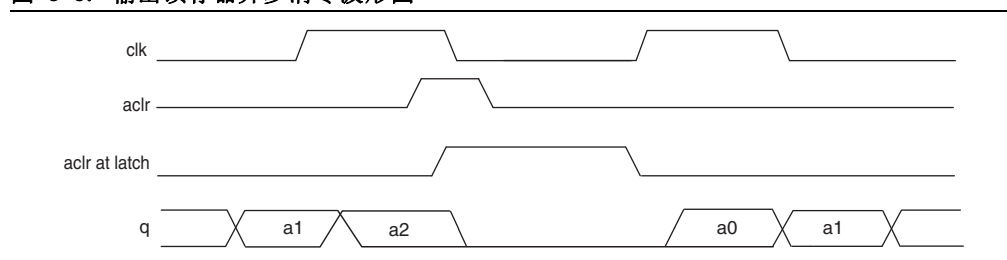
异步清零

Cyclone IV 器件仅支持读地址寄存器、输出寄存器以及输出锁存器的异步清零操作，输入寄存器（除了读地址寄存器）不支持异步清零。当对输出寄存器进行异步清零操作时，异步清零信号即刻对输出寄存器清零，效果立马可见。如果您的 RAM 不使用输出寄存器，那么通过使用输出锁存器异步清零特性，您仍然能够对 RAM 输出端进行清零。

 在读操作过程中，对读地址寄存器置位异步清零有可能会存储数据的损坏。

图 3-5 显示了异步清零特性的功能波形。

图 3-5. 输出锁存器异步清零波形图



 通过使用 Quartus II RAM MegaWizard™ Plug-In Manager，您可以有选择地使能每个逻辑存储器的异步清零功能。

 要了解更多信息，请参考 *RAM Megafunction User Guide*。

通过三种方式可以对 M9K 模块中的寄存器进行复位：

- 器件上电
- 将 aclr 信号仅用于输出寄存器
- 使用 DEV_CLRn 选项来进行全器件复位

存储器模式

Cyclone IV 器件 M9K 存储器模块使您能够在多种操作模式下实现完全同步 SRAM 存储器。Cyclone IV 器件 M9K 存储器模块不支持异步（未寄存的）存储器输入。

M9K 存储器模块支持下列模式：

- 单端口
- 简单双端口
- 真双端口
- 移位寄存器
- ROM
- FIFO



M9K 存储器模块输入寄存器上建立及保持时间的违规将可能导致存储器数据的损坏，这种情况会发生在读写操作期间。

单端口模式

单端口模式支持从单一地址上的异时读写操作。图 3-6 显示了 Cyclone IV 器件 M9K 存储器模块的单端口存储器配置。

图 3-6. 单端口存储器 (注释 1), (注释 2)

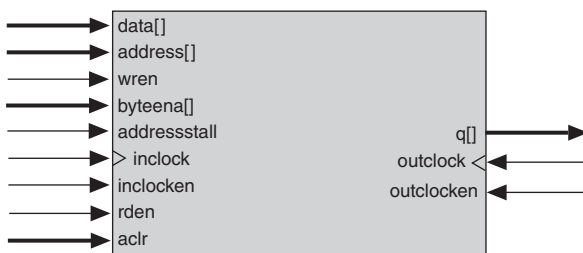


图 3-6 注释:

- (1) 您可以在单一 M9K 模块上实现两个单端口存储器模块。
- (2) 欲了解更多信息，请参考第 3-4 页的“Packed 模式支持”。

写操作期间，RAM 输出行为是可以配置的。如果在写操作期间激活 rden 信号，则 RAM 输出端会显示相应地址上正在写入的新数据，或者原有的旧数据。如果在 rden 信号未激活的情况下执行写操作，那么 RAM 输出端将保留它们在最近的 rden 信号有效期间所保持的值。

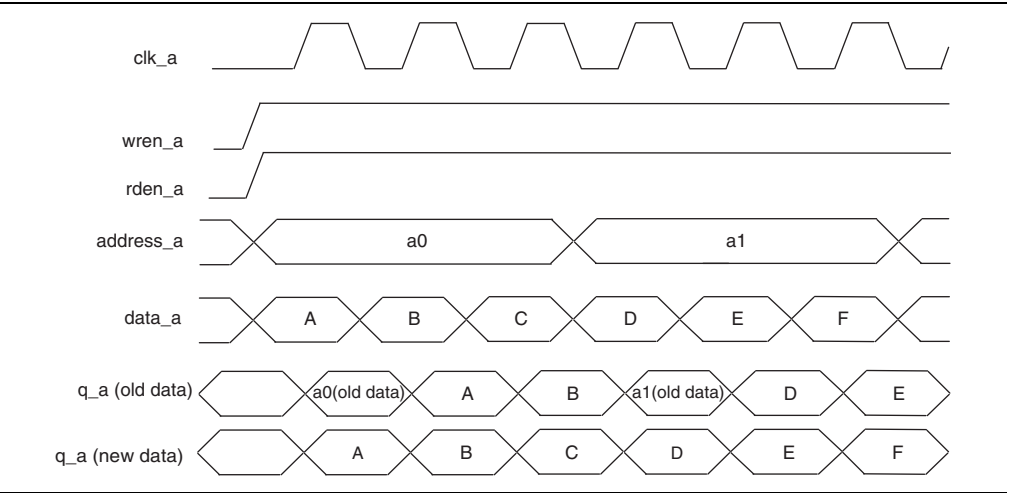
要选择所需的行为，需要在 Quartus II 的 RAM MegaWizard Plug-In Manager 中，将 **Read-During-Write** 选项设置成 **New Data** 或者 **Old Data**。欲了解关于 read-during-write 模式的详细信息，请参考第 3-15 页的“Read-During-Write 操作”。

单端口模式下，M9K 模块的端口位宽配置如下所示：

- 8192 × 1
- 4096 × 2
- 2048 × 4
- 1024 × 8
- 1024 × 9
- 512 × 16
- 512 × 18
- 256 × 32
- 256 × 36

图 3-7 显示了具有未寄存的输出端的单端口模式下读写操作的时序波形。寄存 RAM 的输出端只会将 q 输出延长一个时钟周期。

图 3-7. Cyclone IV 器件单端模式时序波形图



简单双端口模式

简单双端口模式支持到不同位置的同时读写操作。图 3-8 显示了简单双端口存储器配置。

图 3-8. Cyclone IV 器件简单双端口存储器（注释 1）

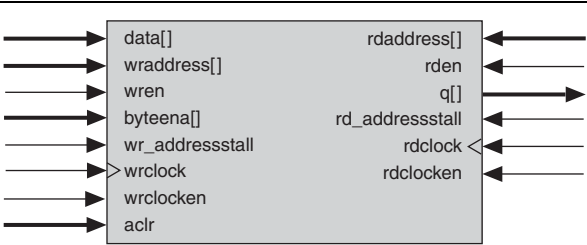


图 3-8 注释：

(1) 除了所示的读或写时钟模式，简单双端口 RAM 也支持输入或输出时钟模式。

Cyclone IV 器件 M9K 存储器模块支持混合位宽配置，以及不同的读写端口位宽。表 3-3 列出了混合位宽配置。

表 3-3. Cyclone IV 器件 M9K 模块混合位宽配置（简单双端口模式） (1/2)

读端口	写端口								
	8192 × 1	4096 × 2	2048 × 4	1024 × 8	512 × 16	256 × 32	1024 × 9	512 × 18	256 × 36
8192 × 1	✓	✓	✓	✓	✓	✓	—	—	—
4096 × 2	✓	✓	✓	✓	✓	✓	—	—	—
2048 × 4	✓	✓	✓	✓	✓	✓	—	—	—
1024 × 8	✓	✓	✓	✓	✓	✓	—	—	—

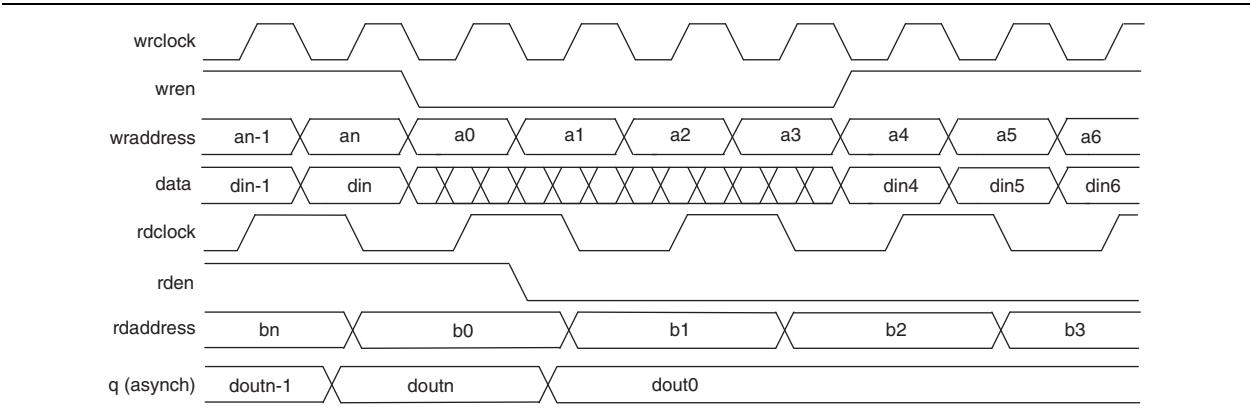
表 3-3. Cyclone IV 器件 M9K 模块混合位宽配置（简单双端口模式） (2/2)

读端口	写端口								
	8192 × 1	4096 × 2	2048 × 4	1024 × 8	512 × 16	256 × 32	1024 × 9	512 × 18	256 × 36
512 × 16	✓	✓	✓	✓	✓	✓	—	—	—
256 × 32	✓	✓	✓	✓	✓	✓	—	—	—
1024 × 9	—	—	—	—	—	—	✓	✓	✓
512 × 18	—	—	—	—	—	—	✓	✓	✓
256 × 36	—	—	—	—	—	—	✓	✓	✓

在简单双端口模式中，M9K 存储器模块支持独立的 wren 与 rden 信号。在没有进行读操作的时候，将 rden 信号保持在低电平（无效状态），从而降低功耗。相同地址上的 Read-during-write 操作能够在相应位置上输出 “New Data” 数据，或者输出 “Old Data”。要选择所需的行为，需要在 Quartus II 的 RAM MegaWizard Plug-In Manager 中将 **Read-During-Write** 选项设置成 **New Data** 或者 **Old Data**。要了解关于这一行为的详细信息，请参考第 3-15 页的 “**Read-During-Write 操作**”。

图 3-9 显示了具有未寄存的简单双端口模式下读写操作的时序波形。寄存 RAM 的输出端只会将 q 输出延长一个时钟周期。

图 3-9. Cyclone IV 器件简单双端口时序波形图



真双端口模式

真双端口模式支持两端口操作的任何组合：在两个不同时钟频率上的两个读操作、两个写操作，或者一个读操作和一个写操作。图 3-10 显示了 Cyclone IV 器件真双端口存储器配置。

图 3-10. Cyclone IV 器件真双端口存储器 (注释 1)

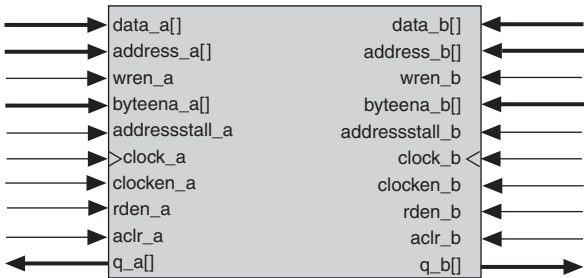


图 3-10 注释：
(1) 除了所示的独立时钟模式，真双端口存储器还支持输入或输出时钟模式。


 真双端口模式下，M9K 模块的最大位宽配置为 512 × 16 位（带有奇偶校验的 18 位）。

表 3-4 显示了可能的 M9K 模块混合端口位宽配置。

表 3-4. Cyclone IV 器件 M9K 模块混合位宽配置（真双端口模式）

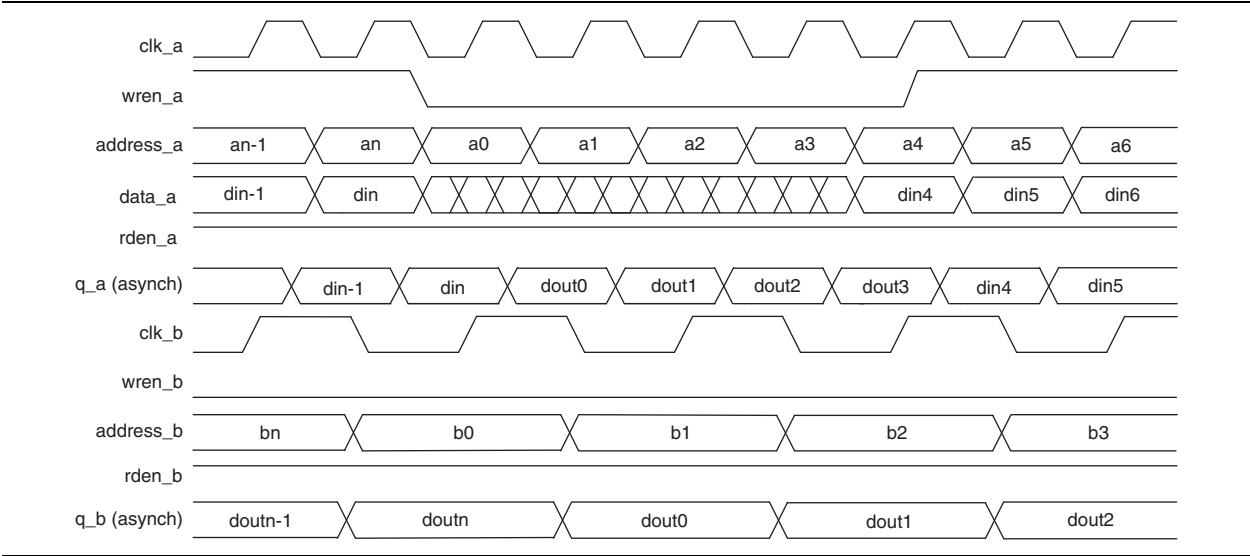
读端口	写端口						
	8192 × 1	4096 × 2	2048 × 4	1024 × 8	512 × 16	1024 × 9	512 × 18
8192 × 1	✓	✓	✓	✓	✓	—	—
4096 × 2	✓	✓	✓	✓	✓	—	—
2048 × 4	✓	✓	✓	✓	✓	—	—
1024 × 8	✓	✓	✓	✓	✓	—	—
512 × 16	✓	✓	✓	✓	✓	—	—
1024 × 9	—	—	—	—	—	✓	✓
512 × 18	—	—	—	—	—	✓	✓

在真双端口模式中，M9K 存储器模块支持独立的 wren 与 rden 信号。在没有进行读操作的时候，将 rden 信号保持在低电平（无效状态），从而降低功耗。相同地址上的 Read-during-write 操作能够在相应位置上输出 “Don’ t Care” 数据，或者输出 “Old Data”。要选择所需的行为，需要在 Quartus II 的 RAM MegaWizard Plug-In Manager 中将 Read-During-Write 选项设置成 Don’ t Care 或者 Old Data。要了解关于这一行为的详细信息，请参考第 3-15 页的 “Read-During-Write 操作”。

在真双端口模式中，您可以从端口 A 或者端口 B 随时访问存储器位置。然而，当从两个端口同时访问同一存储器位置时，您一定要避免可能发生的写冲突。当您试图从两个端口同时写入到相同地址时，会出现写冲突。这会导致在该地址存储未知数据。Cyclone IV 器件 M9K 存储模块中没有集成冲突解决电路。您必须处理 RAM 模块外部的地址冲突。

图 3-11 显示了端口 A 上的写操作与端口 B 上的读操作的真双端口时序波形。寄存 RAM 的输出端只会将 q 输出延长一个时钟周期。

图 3-11. Cyclone IV 器件真双端口时序波形图



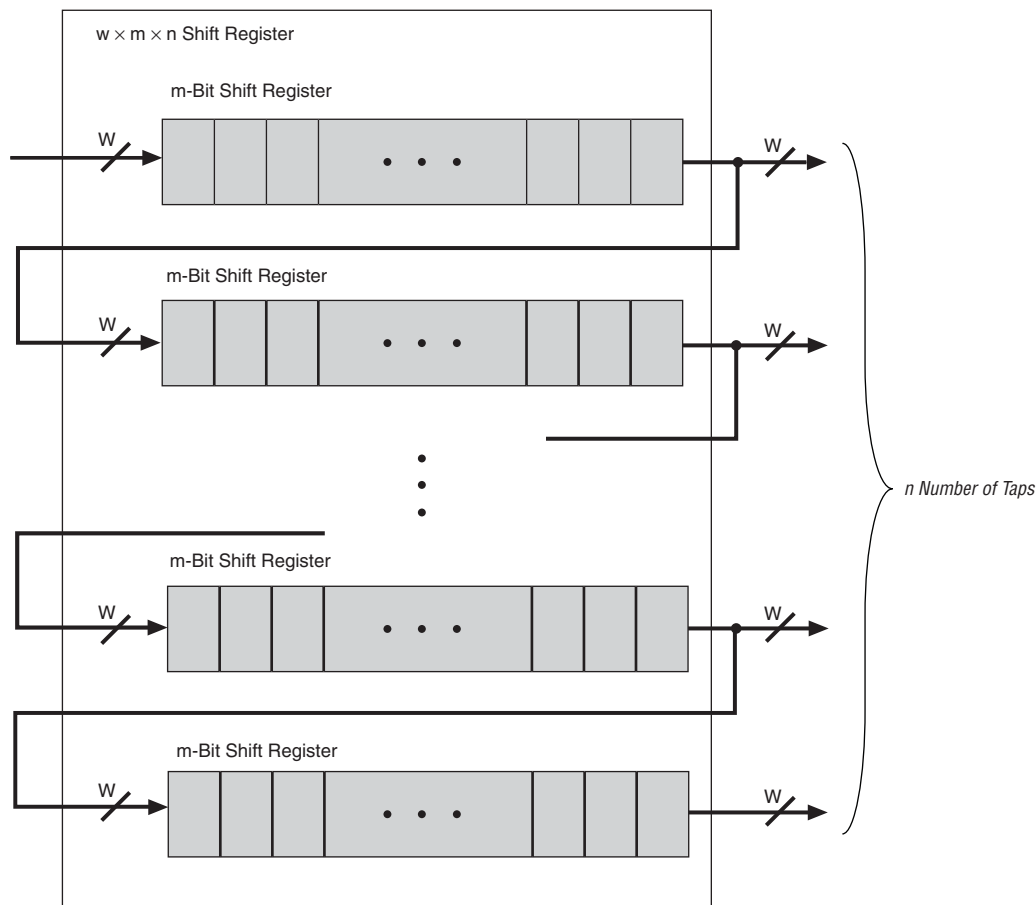
移位寄存器模式

Cyclone IV 器件 M9K 存储器模块能够实现数字信号 (DSP) 应用透过使用的移位寄存器，例如：有限脉冲响应 (FIR) 滤波器、伪随机数生成器、多通道滤波，以及自相关和互相关函数。这些以及其它 DSP 应用都要求本地数据存储，通常通过标准触发器来实现，这些标准触发器迅速消耗大型移位寄存器的很多逻辑单元。更有效的方法是将嵌入式存储器用作移位寄存器模块，这样可以节省很多逻辑单元以及布线资源。

一个 $(w \times m \times n)$ 移位寄存器的容量是由输入数据宽度 (w)、抽头的长度 (m)，以及抽头的数量 (n) 来决定，并且必须小于或等于最大存储器位数，也就是 9,216 位。另外， $(w \times n)$ 的容量必须小于或等于模块的最大宽度，也就是 36 位。如果需要一个容量更大的移位寄存器，则需要将 M9K 存储器模块串联起来使用。

图 3-12 显示了移位寄存器模式中的 Cyclone IV 器件 M9K 存储器模块。

图 3-12. Cyclone IV 器件移位寄存器模式配置



ROM 模式

Cyclone IV 器件 M9K 存储器模块支持 ROM 模式。**.mif** 文件对这些模块的 ROM 中的数据进行初始化。ROM 的地址线是寄存的，输出端可以被寄存，也可以不被寄存。ROM 的读操作与单端口 RAM 配置中的读操作相同。

FIFO 缓冲器模式

Cyclone IV 器件 M9K 存储器模块支持单时钟或者双时钟 FIFO 缓冲器。当从一个时钟域到另一个时钟域传输数据时，会用到双时钟 FIFO 缓冲器。Cyclone IV 器件 M9K 存储器模块不支持同时读写操作在一个空白 FIFO 缓冲器中。




要了解关于 FIFO 缓冲器的详细信息，请参考 *Single- and Dual-Clock FIFO Megafunction User Guide*。

时钟模式

Cyclone IV 器件 M9K 存储器模块支持下列时钟模式：

- Independent （独立）
- Input or output （输入或输出）
- Read or write （读或写）
- Single-clock （单时钟）

当使用输入或输出时钟模式时，如果在同一地址位置执行同时读或写操作，则输出读数据将是未知的。如果要求输出数据是一个可预测值，则需要使用单时钟模式或者 I/O 时钟模式，并且在 MegaWizard Plug-In Manager 中选择相应的 read-during-write 行为。

 违反存储器模块输入寄存器上建立及保持时间将可能导致存储器数据的损坏，这种情况会发生在读写操作期间。


 异步清零功能仅在读地址寄存器、输出寄存器，以及输出锁存器中可用。

表 3-5 列出了时钟模式 VS 存储器模式所支持的矩阵。

表 3-5. Cyclone IV 器件存储器时钟模式

时钟模式	真双端口模式	简单双端口模式	单端口模式	ROM 模式	FIFO 模式
独立	✓	—	—	✓	—
输入或输出	✓	✓	✓	✓	—
读或写	—	✓	—	—	✓
单时钟	✓	✓	✓	✓	✓

独立时钟模式

Cyclone IV 器件 M9K 存储器模块能够实现真双端口存储器的独立时钟模式。在这一模式中，独立的时钟可用于每一个不同的端口（端口 A 与端口 B）。clock A 控制端口 A 侧上的所有寄存器，而 clock B 则控制端口 B 侧上的所有寄存器。另外，每个端口均支持端口 A 和端口 B 寄存器的独立时钟使能。

输入或输出时钟模式

Cyclone IV 器件 M9K 存储器模块支持输入或输出时钟模式，以用于 FIFO、单端口、真双端口以及简单双端口存储器。在这一模式中，输入时钟控制存储器模块的所有输入寄存器，其中包括数据、地址、byteena、wren 以及 rden 寄存器。输出时钟控制数据输出寄存器。另外，每一个存储器模块端口均支持用于输入与输出寄存器的独立时钟使能。

读或写时钟模式

Cyclone IV 器件 M9K 存储器模块能够实现用于 FIFO 以及简单双端口存储器的读或写时钟模式。在这一模式中，写时钟控制数据输入、写地址和 wren 寄存器。同样的，读时钟控制数据输出、读地址和 rden 寄存器。M9K 存储器模块支持独立的时钟使能，以用于读时钟以及写时钟。

当使用读或写模式时，如果在同一地址执行同时读或写操作，则输出读数据将是未知的。如果要求输出数据是一个可预测值，则需要使用单时钟模式，输入时钟模式，或者输出时钟模式，并且在 MegaWizard Plug-In Manager 中选择相应的 read-during-write 行为。

单时钟模式

Cyclone IV 器件 M9K 存储器模块能够实现单时钟模式，以用于 FIFO、ROM、真双端口，简单双端口以及单端口存储器。在这一模式中，您可以通过单时钟以及时钟使能来控制 M9K 存储器模块中的所有寄存器。

设计考量

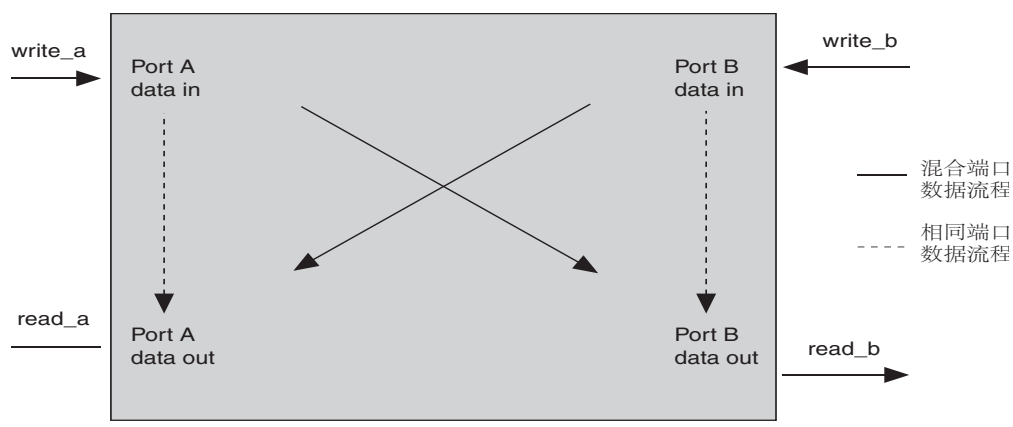
本章节对采用 M9K 存储器模块进行设计时需要考虑的因素作了描述。

Read-During-Write 操作

第 3-16 页的 “相同端口 Read-During-Write 模式” 与 “混合端口 Read-During-Write 模式” 介绍了在写操作期间从相同地址读取数据时各种 RAM 配置的功能性。

有两个 read-during-write 数据流程：相同端口 (same-port) 与混合端口 (mixed-port)。图 3-13 显示了这两个数据流程之间的区别。

图 3-13. Cyclone IV 器件 Read-During-Write 数据流程



相同端口 Read-During-Write 模式

这一模式适用于单端口 RAM 或者真双端口 RAM 的同一端口。在相同端口 read-during-write 模式中提供了两个输出选择：**New Data** 模式（或者直通模式）和 **Old Data** 模式。在 **New Data** 模式中，在相同时钟周期的上升沿写入新数据，并且可以在该上升沿使用这些数据。在 **Old Data** 模式中，RAM 输出显示了原有旧数据在写操作开始前。

当与 byteena 信号一起使用 **New Data** 模式时，您能够对 RAM 的输出进行控制。当 byteena 信号为高电平时，存储器中写入的数据将传递至输出端（直通）。当 byteena 信号为低电平，屏蔽数据不会写入存储器，而存储器中的旧数据会出现在输出端。因此，输出数据可以是由 byteena 信号所决定的新旧数据组合。

图 3-14 与图 3-15 分别显示了 **New Data** 和 **Old Data** 模式下的相同端口 read-during-write 行为的简单功能波形。

图 3-14. 相同端口 Read-During-Write: New Data 模式

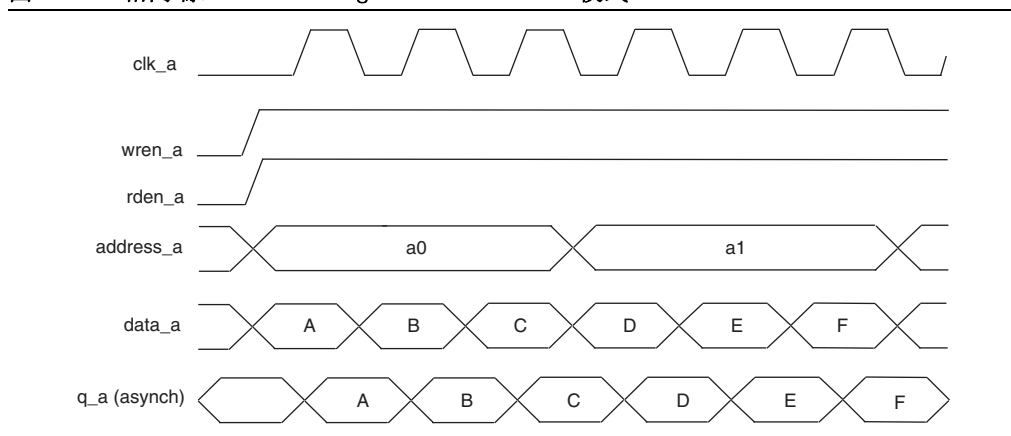
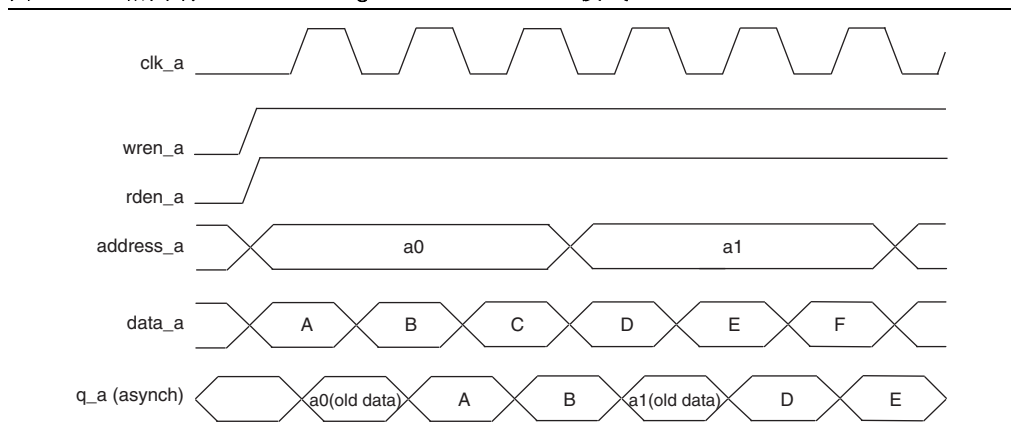


图 3-15. 相同端口 Read-During-Write: Old Data 模式



混合端口 Read-During-Write 模式

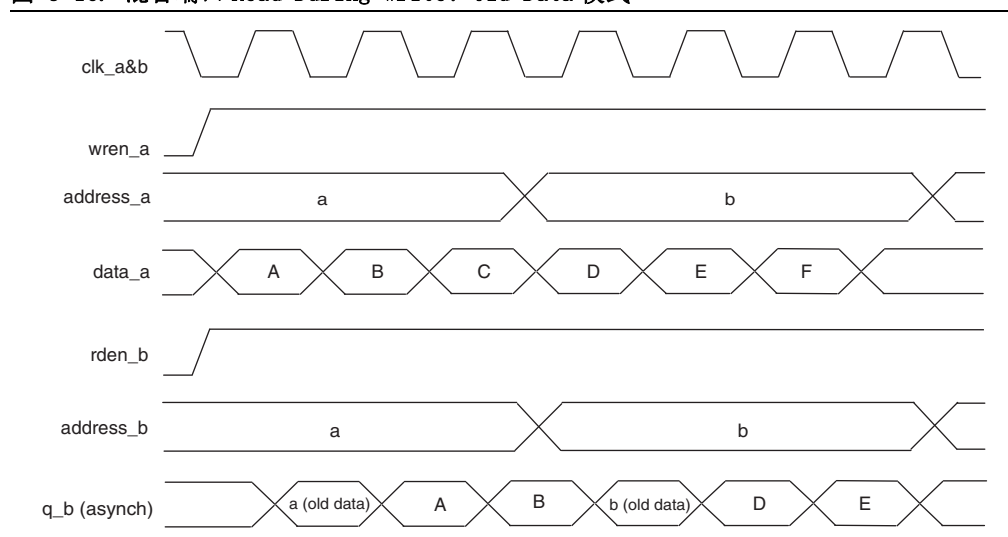
这一模式适用于简单或者真双端口模式下的 RAM，其中一个端口用于读取数据，而另一个端口在同一地址写入数据于同一时钟。

在这一模式中提供了两个输出选择：**Old Data** 模式或者 **Don't Care** 模式。在 **Old Data** 模式中，不同端口上的 read-during-write 操作会使 RAM 输出端显示相应地址上的旧数据。在 **Don't Care** 模式中，同样的操作会导致 RAM 输出端上的一个 “Don't Care” 或者未知值。

要了解关于如何实现所需的行为，请参考 *RAM Megafunction User Guide*。

图 3-16 显示了 **Old Data** 模式下的混合端口 read-during-write 行为的简单功能波形。在 **Don't Care** 模式中，旧数据由 “Don't Care” 来替换。

图 3-16. 混合端口 Read-During-Write: Old Data 模式



对于使用双时钟的混合 read-during-write 操作，时钟之间的关系决定了存储器的输出行为。如果将同一时钟用于两个时钟，则输出端是相应地址上的旧数据。然而，如果使用不同的时钟，那么在混合端口 read-during-write 操作期间，输出端数据则是未知的。这一未知值可能会是相应地址上的旧值，也可能会是新值，这取决于读操作是开始于写操作之前还是之后。

冲突解决

当在真双端口模式下使用 M9K 存储器模块时，在同一地址尝试两个写操作是可行的。由于 M9K 存储模块中没有集成冲突解决电路，因此会产生正在写入该地址的未知数据。所以，您必须实现 RAM 模块外部的冲突解决逻辑。

上电条件与存储器初始化

不管输出寄存器被使用还是被旁路，Cyclone IV 的 M9K 存储器模块输出端都会上电至零（清零）。通过使用 **.mif** 文件，所有 M9K 存储模块均支持初始化。您可以在 Quartus II 中创建 **.mif** 文件，并在设计中实例化存储器时通过 RAM MegaWizard Plug-In Manager 来指定它们的使用。即使存储器是预初始化的（例如，使用一个 **.mif** 文件），它也会通过输出端清零进行上电。只有上电后的后续读操作会输出预初始化的值。

 要了解关于 **.mif** 文件的详细信息，请参考 *RAM Megafunction User Guide* 和 *Quartus II Handbook*。

功耗管理

Cyclone IV 器件的 M9K 存储器模块时钟使能支持对每一个 M9K 存储器模块时钟的控制，以降低 AC 功耗。需要使用 **rden** 信号来确保读操作仅在必要时出现。如果您的设计不要求 read-during-write，那么在写操作期间或者存储器未操作期间，需要通过置低 **rden** 信号以降低功率。Quartus II 会自动对未使用的 M9K 存储器模块进行断电，从而降低静态功耗。

文档修订历史

表 3-6 显示了本章节的修订历史。

表 3-6. 文档修订历史

日期	版本	修订内容
2011 年 11 月	1.1	更新了“ 字节使能支持 ”章节。
2009 年 11 月	1.0	首次发布。