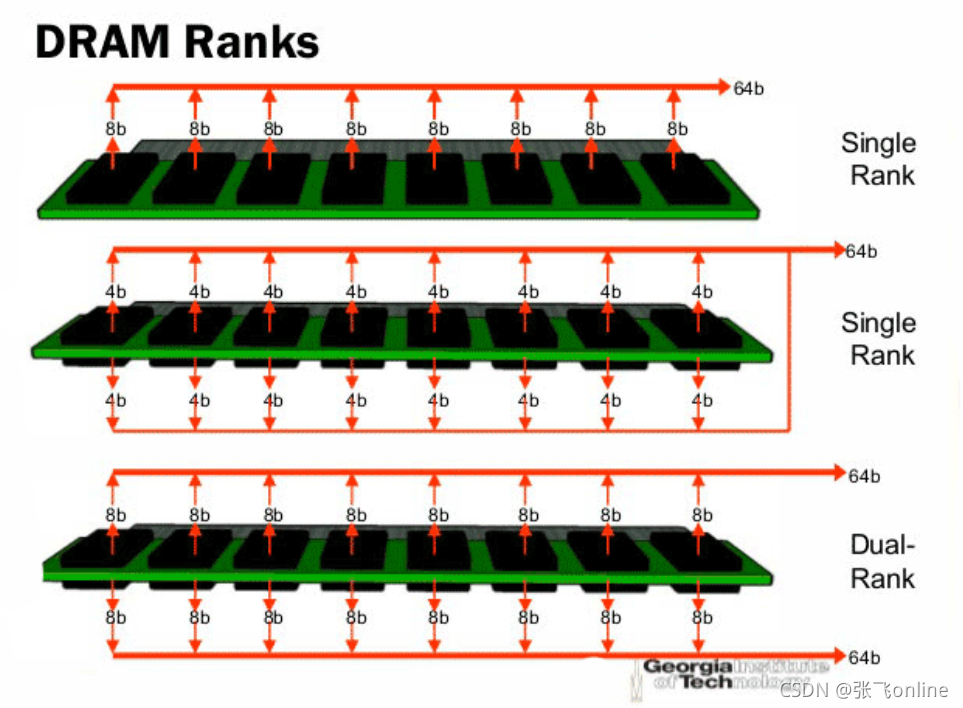
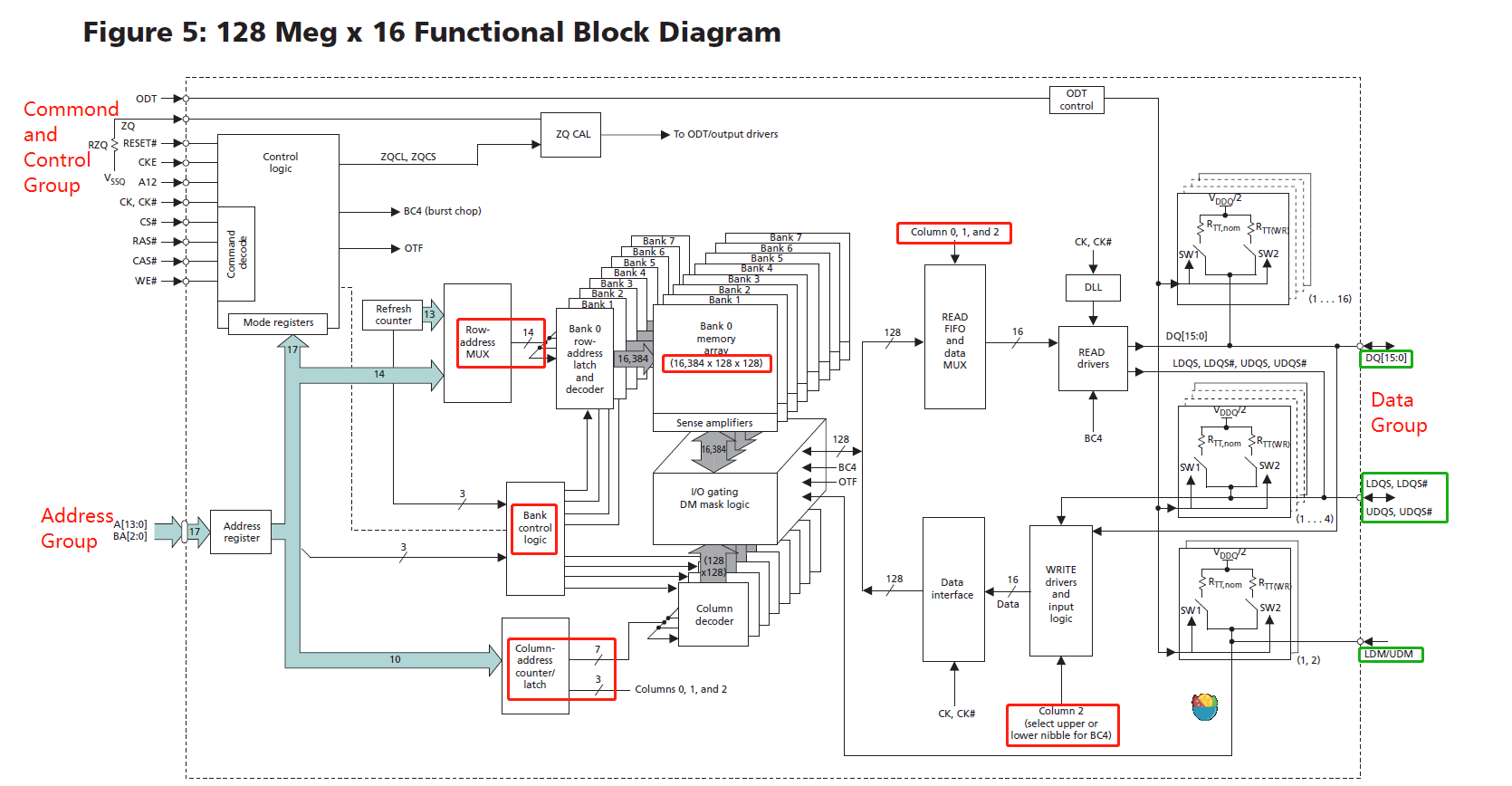
# DDR基本结构



Chip：ddr芯片，ddr颗粒

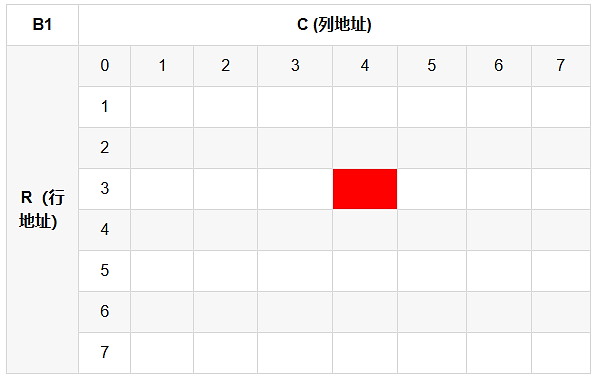
Rank：也叫物理Bank，一组64bit的chip级联





Bank：存储阵列，DDR3中每个颗粒内部通常有8个Bank，这些bank共享一个memory I/O controller，在每个Bank内部的读写可以并行进行

由行（row，也称page）、列（column）组成，~~存储单位大小为数据线位宽~~

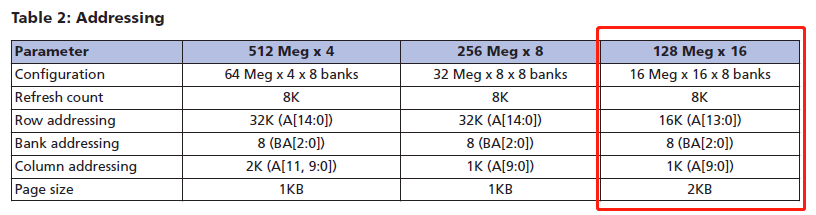


寻址时，首先发送激活命令，以及Bank地址、Row地址，选择某一个 Bank中的某一行；然后发送读写命令和Col地址，此时Bank地址和Row地址仍是有效的，选择这一Bank这一行中的某一列Col，这样就定位到了具体的存储单元。

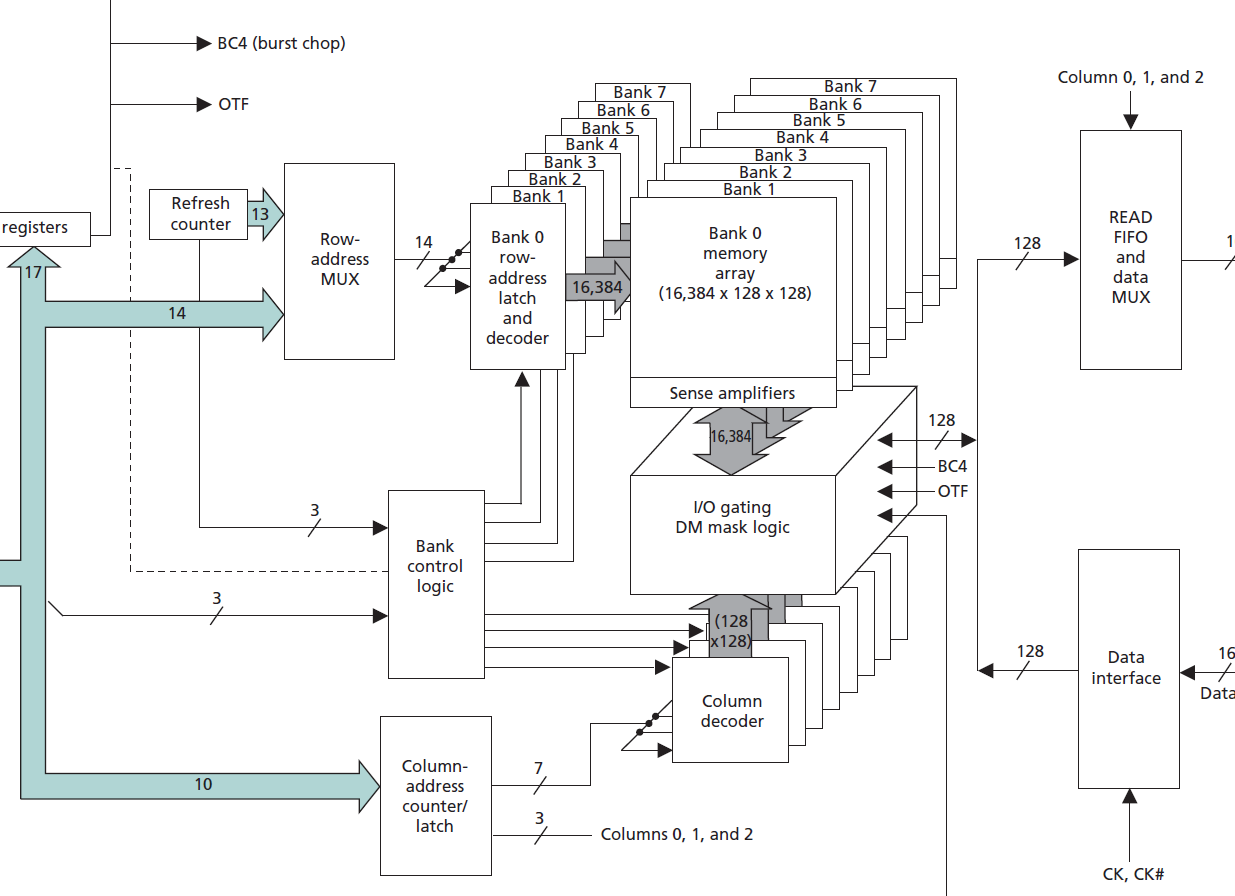
## 内存容量计算：

DDR3型号：MT41J128M16XX-125

**MT41J128M16 – 16 Meg x 16 x 8 Banks**



Row \* Col \* banks \* datawidth



Row：2^14 Col: 2^7 datawidth: 16bit prefetch: 8n banks: 8

16384 \* 128 \* 16 \* 8 \* 8 = 128M \*16

该ddr数据线位宽是16bit，prefetch数据为16\*8=128bit

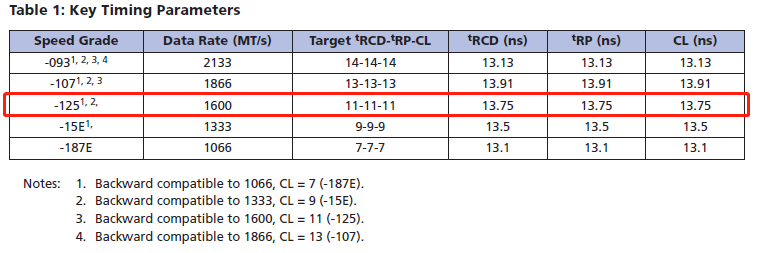
也就是ddr内部一个时钟可以提取128bit数据

8n-prefetch n为数据位宽 ddr内部时钟是内核时钟的四倍

该ddr存储空间是8个bank 16384行128列 128bit每个cell（对应8n）

### 时间参数：

DDR3型号：MT41J128M16XX-125



tFAW：四个激活命令的最小间隔

tRRD：ACTIVE-to-ACTIVE的最小时间间隔

tRAS：ACTIVE-to-PRECHARGE的最小时间间隔

tRCD：ACTIVE-to-READ/WRITE的最小时间间隔

tWTR：WRITE-to-READ的最小时间间隔

CWL：write CAS latency

CL：read CAS latency

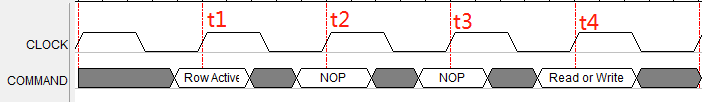
tRP：预充电需要precharge的原因，关闭当前行，打开新行时，要求新的bit line充电到VDD/2

**行选通周期**

Bank地址与相应的行地址是同时发出的，此时这个命令称之为**“行激活”（Row Active）**。

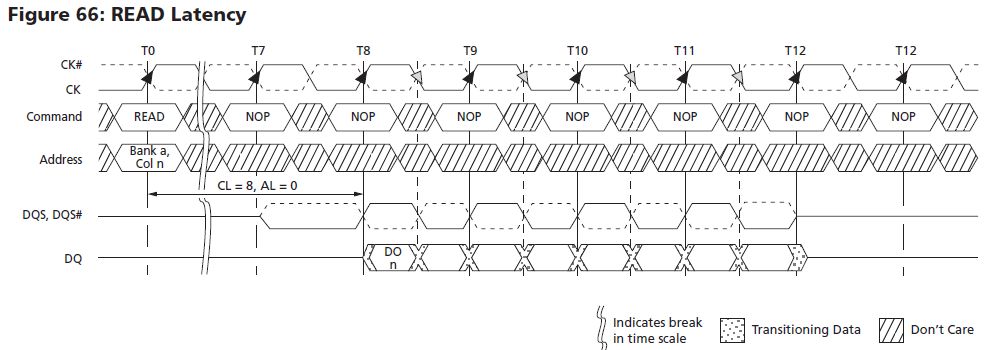
在此之后，将发送列地址寻址命令与具体的操作命令（是读还是写），这两个命令也是同时发出的，所以一般都会以“读/写命令”来表示列寻址。根据相关的标准，从行有效到读/写命令发出之间的间隔被定义为**tRCD**，即RAS to CAS Delay（RAS至CAS延迟，RAS就是行地址选通脉冲，CAS就是列地址选通脉冲）

广义的tRCD以时钟周期（tCK，Clock Time）数为单位，比如tRCD=3，就代表延迟周期为两个时钟周期



**列地址脉冲选通潜伏期**

相关的列地址被选中之后，将会触发数据传输，但从存储单元中输出到真正出现在内存芯片的 I/O 接口之间还需要一定的时间（数据触发本身就有延迟，而且还需要进行信号放大），这段时间就是非常著名的 CL（CAS Latency，列地址脉冲选通潜伏期）。CL 的数值与 tRCD 一样，以时钟周期数表示 CL仅针对读取操作



CL：CAS latency

AL：CAS additive latency

RL：Read latency

RL=CL+AL

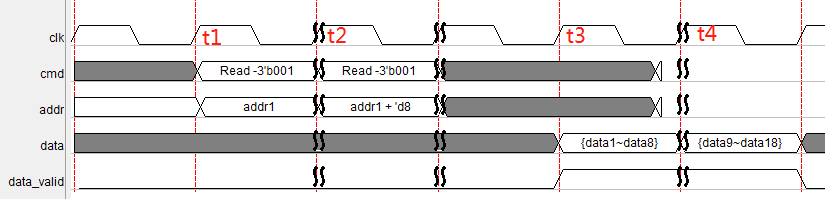
AL 延迟允许紧跟激活命令后发出读命令，器件内部将读命令延迟 AL 个时钟周期后执行。

### 突发传输

**突发（Burst）**是指在同一行中相邻的存储单元连续进行数据传输的方式，连续传输的周期数就是突发长度（Burst Lengths，简称BL）。

在进行突发传输时，只要指定起始列地址与突发长度，内存就会依次地自动对后面相应数量的存储单元进行读/写操作而不再需要控制器连续地提供列地址。这样，除了第一笔数据的传输需要若干个周期（主要是之前的延迟，一般的是tRCD+CL）外，其后每个数据只需一个周期的即可获得。

DDR3 BL=8



t1 ,t2  /  t3，t4时刻不一定连续；要看ddr是否准备好，具体到代码就是app\_rdy 并不是一直为高

### ACTIVE Operation

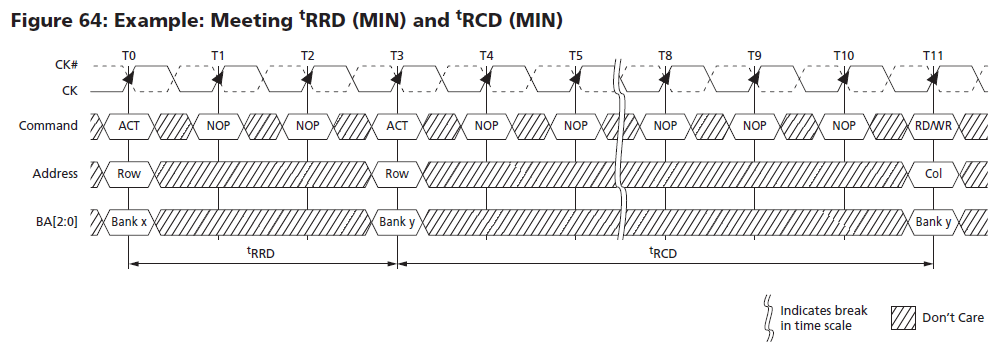
首先需要行选通，才能读取/写入

在bank中，任何READ to READ命令延迟或WRITE to WRITE命令延迟都限制为tCCD（MIN）。

在同一个bank中，只有在前一个激活行关闭（预充电）后，才能向同一存储体中的不同行发出后续激活命令。

对同一存储体的连续激活命令之间的最小时间间隔由tRC定义。

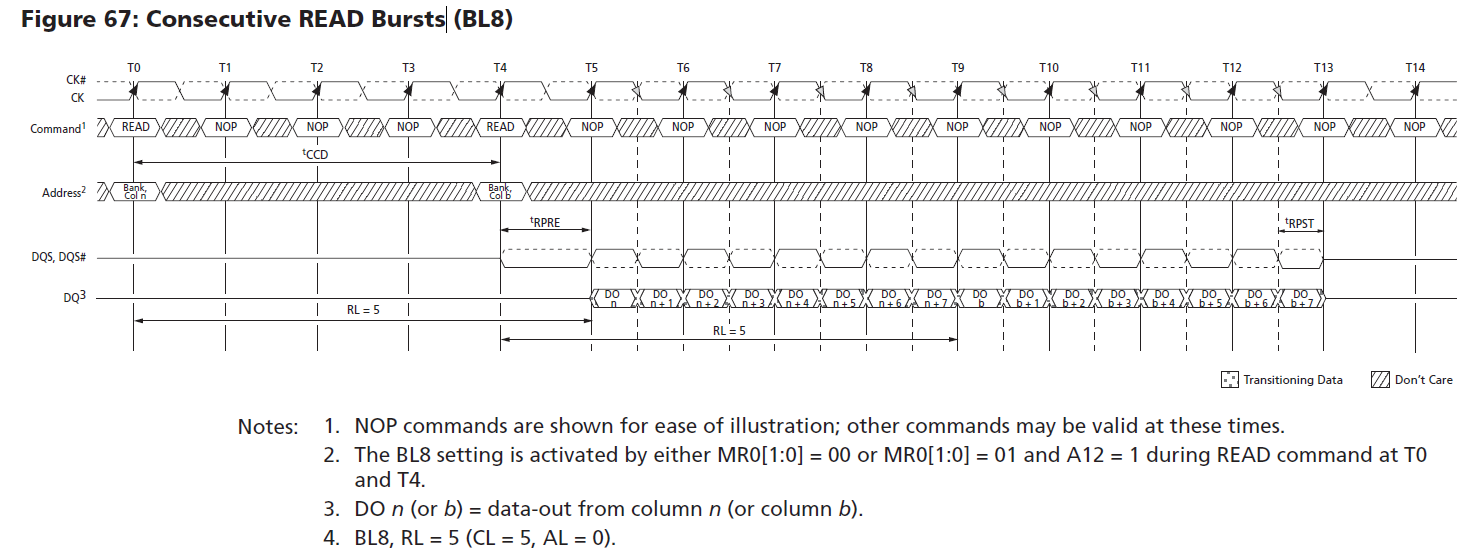
不同bank之间的连续激活命令之间的最小时间间隔由tRRD定义。



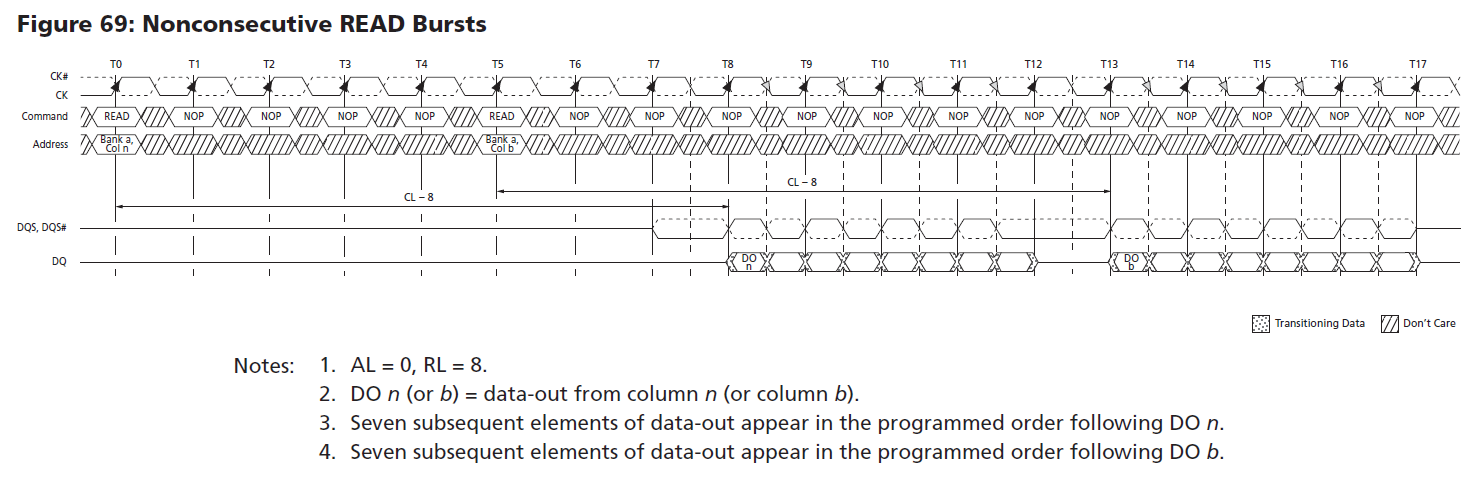
### Read Operation

连续读取突发

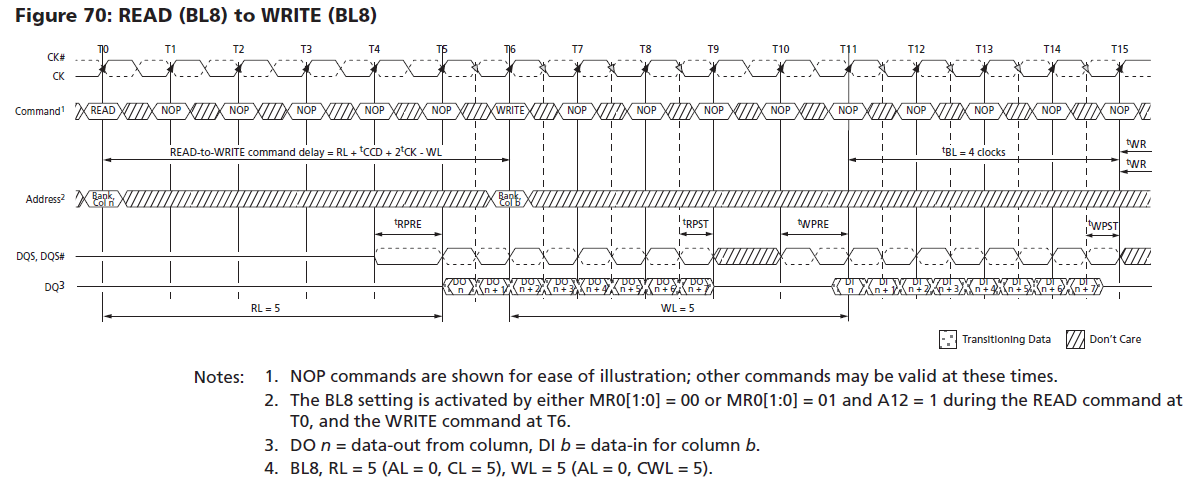
新的READ命令应在第一个READ命令之后tCCD周期发出。



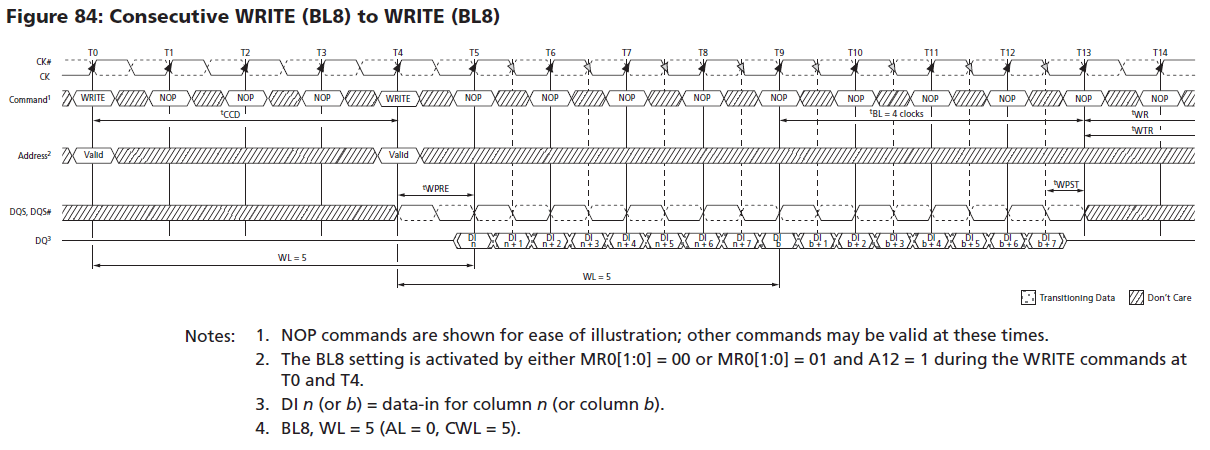
由于DQ、DQS总线是双向的，所以使用一段DQS前导码来区分读写数据，读前 导码是一个周期的DQS低电平信号，时间参数为tRPRE，读后导码是BC4/BL8数据 末尾半个周期的DQS低电平信号，时间参数为tRPST。并且读DQS和读数据是边沿对齐的

非连续突发读取

Read(BL8) to Write(BL8)

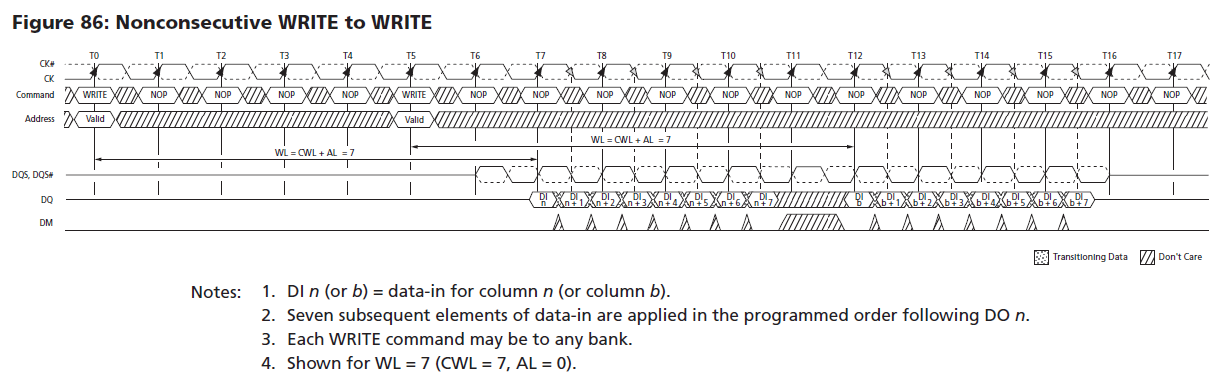


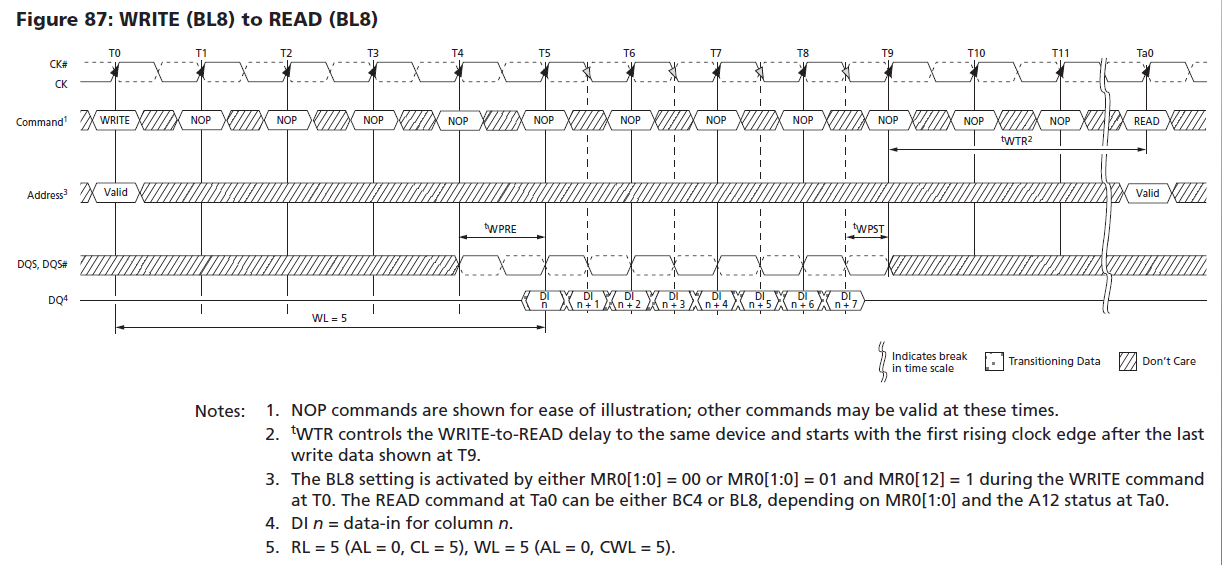
### Write Operation



WL = CWL+AL

写前导码是一个周期的DQS翻转 信号，写DQS的上升沿和第一个写DQ的中心对齐，之后的DQS上升沿、下降沿分别 指向数据DQ的中心。



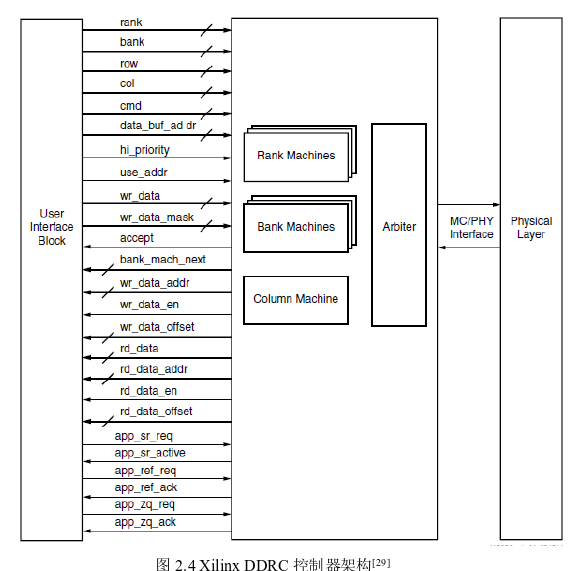


# 如何控制DDR工作

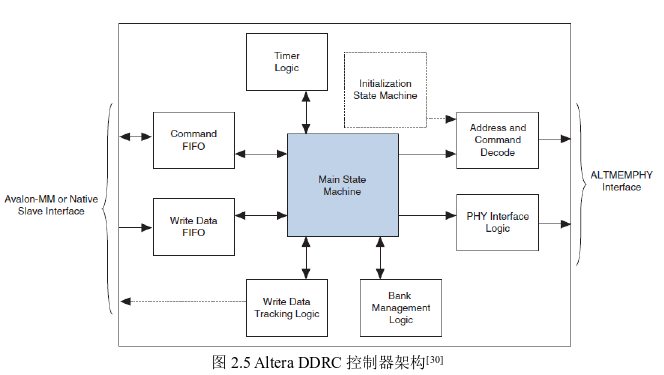
### 2.1 DDRC

DDRC是内存控制器的核心，决定了外部访问DDR存储器的带宽。

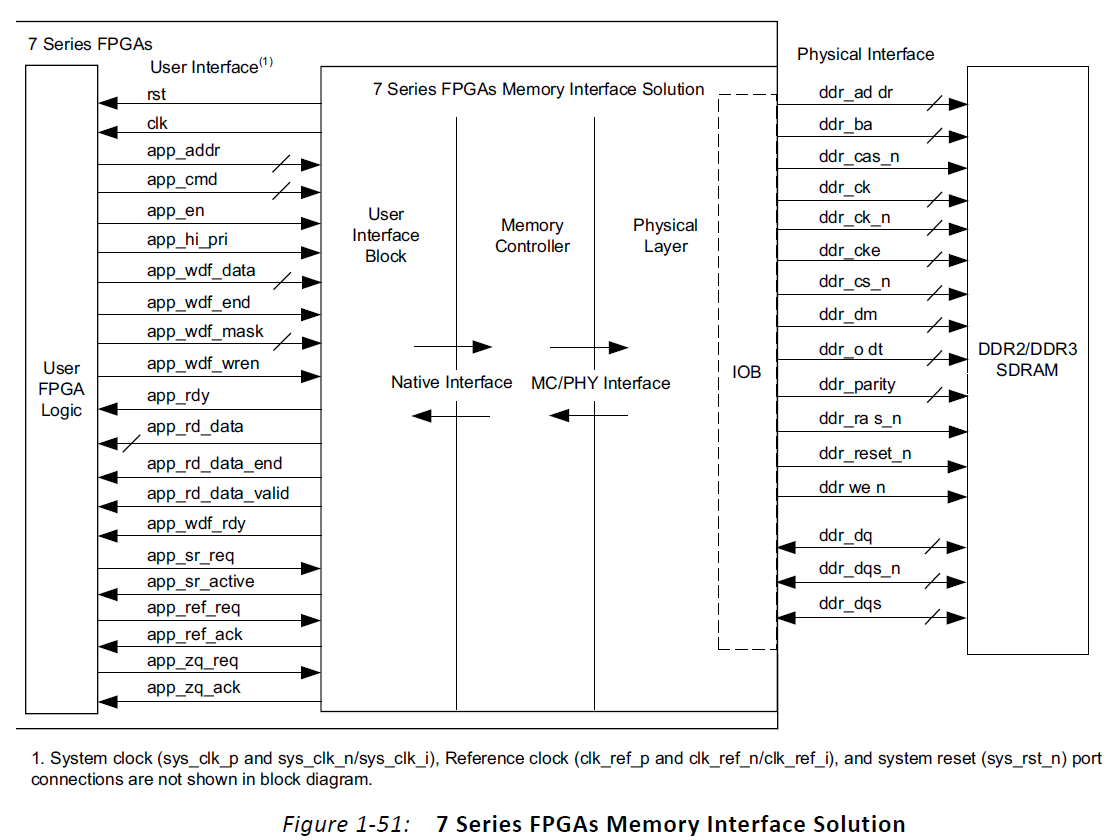
DDRC一般有两种典型结构。区别是Xilinx类型的结构将DDR内存初始化放在 DDR PHY中进行，这也是Synopsys商用DDR控制器所采用的结构。Altera和大部分论文采用的结构是将DDR内存初始化放在DDRC内部完成。



一个读写命令先经过Bank处理模块，产生行命令和列命令，再进入列处理模块，产生满足DDR3 时序的命令，发送到物理层。仲裁模块决定读写命令的优先级，Rank模块分配哪一个Rank。Xilinx的控制器内部通过封装，无法对其进行修改，大大降低了应用的灵活性。并且其内部没有商用的重排序功能，所以效率相对较低。



### 2.2 Mig核心架构



Memory Controller（MC）的前端向UI块提供用户接口，允许用户进行读写请求，并提供将数据从UI块移动到外部存储器设备的机制；MC后端连接到物理端口，并处理该模块的所有接口请求。内存控制器还提供重新排序功能（Reorder）来重新排序接收的请求，以优化数据吞吐量和延迟。

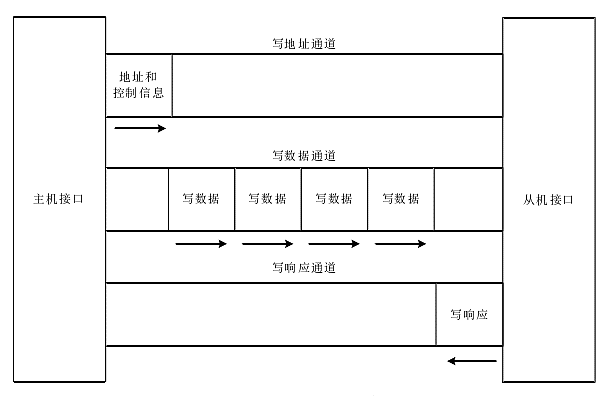
#### 2.2.1 Physical Layer

|  |  |  |
| --- | --- | --- |
| 物理接口 | | |
| 信号名称 | 方向 | 简介 |
| ddr3\_dq[15:0] | In/Out | 低8位和高8位数据信号线，共16位数据信号线  发送写数据，或接受度数据 |
| ddr3\_dqs\_n  ddr3\_dqs\_p | In/Out | 数据选通信号，输出读取的数据，输入写入的数据  一个DQS对应8个DQ |
| ddr3\_addr[13:0] | Input | 地址选择信号，行选通和列选通复用  A13 A12/BC# A11 A10/AP A[9:0] |
| ddr3\_ba[2:0] | Input | Bank地址选择，定义在load mode期间加载哪种模式寄存器  (MR0 MR1 MR2 MR3)  ACTIVE READ WRITE PRECHARGE |
| ddr3\_ras\_n | Input | 控制信号，不同的组合代表不同的命令 |
| ddr3\_cas\_n |
| ddr3\_we\_n |
| ddr3\_reset\_n | Input | DDR复位信号，低电平有效。正常操作过程中，保持高电平 |
| ddr3\_ck\_p  ddr3\_ck\_n | Input | 差分时钟信号，由DDR Controller输出。所有控制和地址输入信号在CK上升沿和CK#的下降沿交叉处被采样，输出数据选通(DQs,DQS#)参考与CK和CK#的交叉点 |
| ddr3\_cke | Input | 时钟信号使能。CKE为高电平时，启动内部时钟信号、设备输入缓冲以及输出驱动单元。CKE低电平时则关闭上述单元，进入低功耗模式 |
| ddr3\_cs\_n |  | DDR芯片使能，用于多个RANK时的RANK组选择 |
| ddr3\_dm | In/Out | 数据掩码 |
| ddr3\_odt | Input | ODT使能(高)和禁止(低)片内终端电阻，在常操作使能时,ODT仅对下面的引脚有效:DQ[7：0]、DQS、DQS#和DM。如果通过LOADMODE命令禁止,ODT输入被忽略。ODT的参考值是VREFCA引脚。 |
| LDM | Input | 写入数据的低字节输入掩码信号。当LDM在写入访问期间与输入数据一起被采样为高电平时，低字节输入数据被屏蔽。 |
| UDM | Input | 写入数据的高位字节输入掩码信号。当UDM在WRITE访问期间与输入数据一起被采样为HIGH时，高位字节输入数据被屏蔽。 |

#### 2.2.2 AXI总线接口

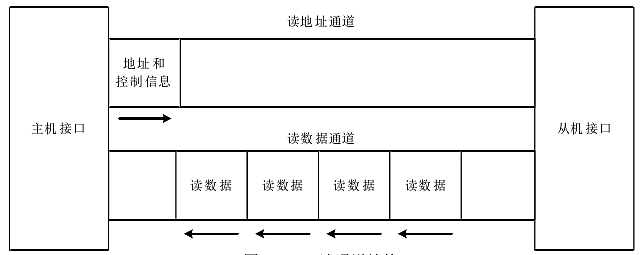
AXI总线在主机和从机之间设置了5个独立的通道，它们是写地址通道、写数据 通道、写响应通道、读地址通道，以及读数据通道（读响应信号合并到读数据通道）。5个通道相互独立，都只支持单向传输，共同使用一套握手机制：VALID/READY 机制。

AXI写通道机制：



主机（Master）先在写地址通道上告知从机（Slave）本次传输 （Transaction）的特性，包括地址信息和控制信息。之后，在写数据通道向从机写入 数据，一次传输操作中可以包括多个数据传输（Data Transfers）。最后，从机在接收到 写数据后，通过写响应通道将本次传输操作的响应告知主机。主机以收到从机的响应 信号，作为本次传输操作结束的标志。

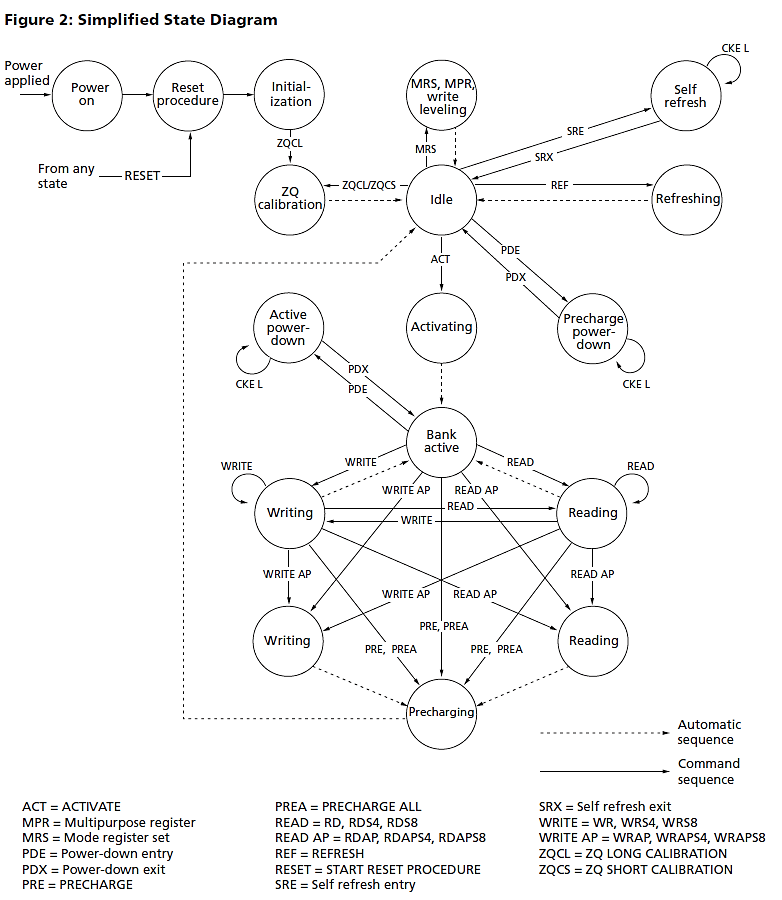
AXI读通道机制：



读操作只涉及两个通道，如图2.3所示，首先主机在读地址通道上写入本次读操 作待读取数据的地址信息和控制信息。从机接收到地址后，将该地址上的读数据和读 响应信号通过读数据通道同时传输给主机。

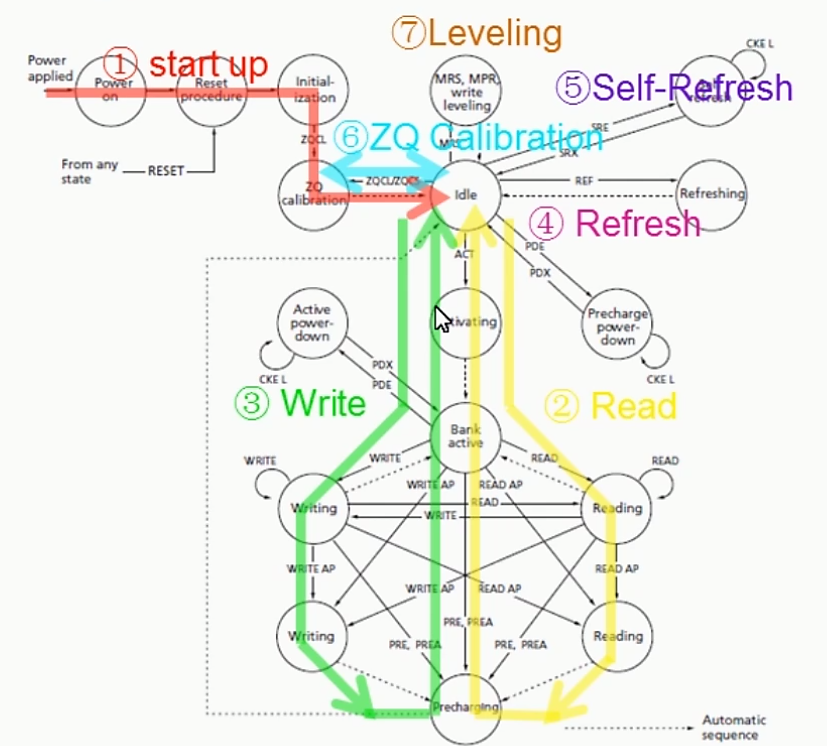
# DDR3 SDRAM工作过程

## 3.1 单个Bank状态机过程



DDR3颗粒内部单个Bank的状态机命令跳转过程

由于DRAM 电容充放电读写数据的关系，每一个命令之间的跳转必须满足特定的时间参数。下图的实线箭头代表了状态之间通过命令使能进行跳转，虚线箭头代表状态机自动跳转。



DDR3 SDRAM首先要经过上电复位和初始化过程，这个过程中完成 ZQ长校准和模式寄存器的配置，完成上电初始化后，DDR3 SDRAM进入IDLE状态。

随后可进行读操作、写操作、定时刷新、自刷新、Power-Down操作，以及ZQ短校准操作。

在进行读/写操作前，要先进行激活（打开行）操作，读/写完成后要发送预充电（关闭）命令。

自动刷新：正常模式下 自刷新：低功耗

复位：异步复位，拉低保持100ns

RESET#拉高前，CKE保持低电平不变

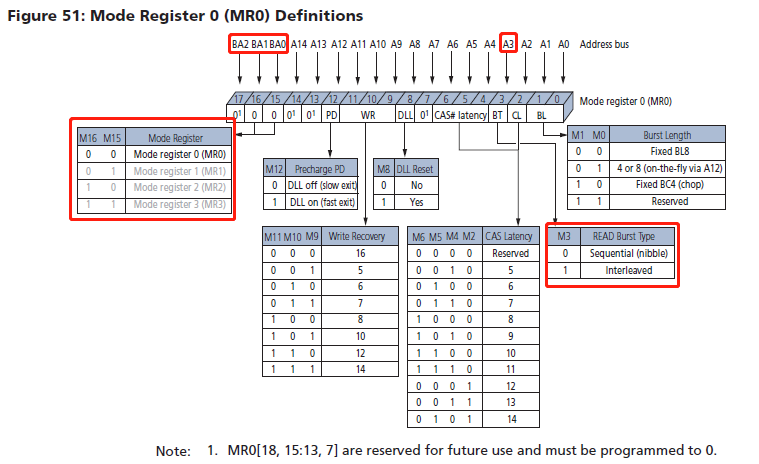
MRS：Mode register set

模式寄存器设置，MR没有缺省值，所以MR必须在上电或者复位后被完全初始化，DDR才能正常工作

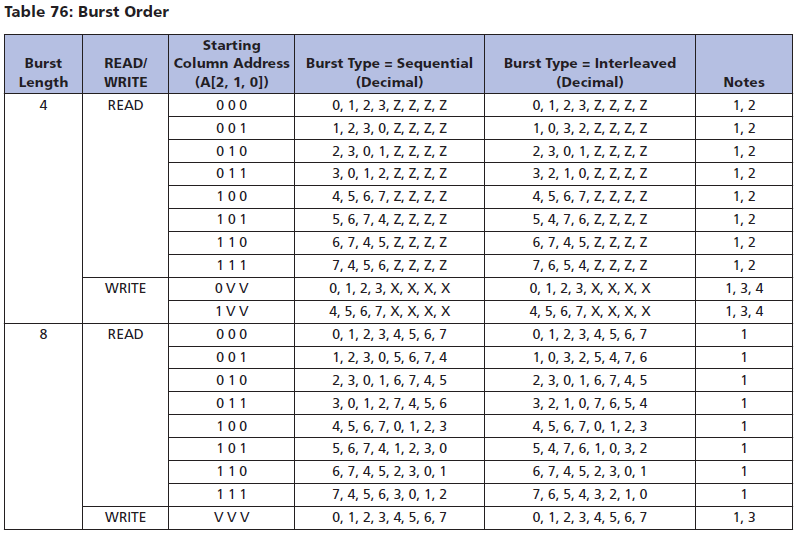
MRS命令需要所有bank处于idle状态，且只能在precharged状态下进行

# 定点回答

## 4.1 CA[2:0]有什么用？



列地址A0，A1，A2这3bit被用于Burst Order功能，A3被用于Burst Type功能



更形象地理解就是，对于Bank中的Memory Array，每个Memory Cell是一个Byte的集合体。CA[9:3]选中一行中的一个特定Byte，再有CA[2:0]选择从这个Byte的哪个位置开始操作。

CA3既参与列地址译码，也决定Burst是连续读取还是交错读取

## 4.2 BA[2:0]有什么用？

MR0用来存储DDR3的不同操作模式的数据：包括突发长度、读取突发种类、CAS长度、测试模式、DLL复位等。

MR1用来存储是否使能DLL、输出驱动长度、Rtt\_Nom、额外长度、写电平使能等。

MR2用来存储控制更新的特性， Rtt\_WR阻抗，和CAS写长度。

MR3用来控制MPR。

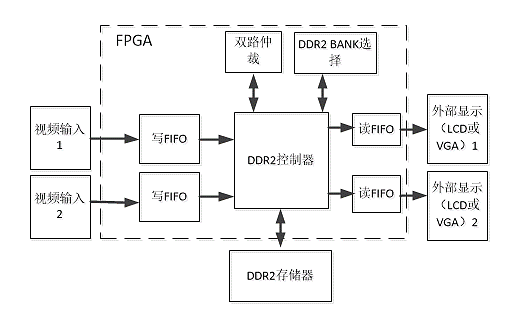
模式寄存器设置，MR没有缺省值，所以MR必须在上电或者复位后被完全初始化，DDR才能正常工作

## 4.3 DDR能不能同时读写？如果能，如何同时读写？

共用一组地址线 同时读写到了DDR侧仍是串行执行

## 4.4 DDR多个控制器指什么？什么时候使用多个控制器？

控制器控制外部对DDR的访问。多个控制器一般用来控制多个DDR，控制器到DDR的物理连线需要分开。当多个控制器控制一个DDR的时候，由于物理接口共用一套，所以多个控制的指令在DDR侧都是串行执行的。



多用户 一个控制器，当第一路使用DDR控制资源时候，第二路只能等待第一路完成以后，再占用，如此循环。

## 4.5 DDR具体怎么存储的 Bank Chip? 又如何读取？

4块16bit的DDR3的BA[2：0]、Addr[13：0]是并行连接到CPU

CPU认为只有一块内存，访问的时候按照BA0~BA2和Addr0~Addr13给出地址

4块chip都收到地址，将特定存储cell的16it数据送到数据线上/将数据线上的16bit写入到指定地址

凭借数据线的连接，第一个chip Addr0~15连CPU的0~15，以此类推

4 \* 2 \* 8=64B（bank\*cellbyte\*burstlength）

CPU一次交换的数据量 Cacheline一般是64B

## 4.6 DDR3影响效率的因素

（1）理论带宽

1600MHz \* 64bit = 12.8GBps

（2）连续读取、写入，形成流水线，避免浪费带宽

读写切换：

写命令+写数据 → 读命令 等待tWTR时间

读命令+读数据 → 写命令 等待RL+tCCD+2tCK-WL 较小

（3）行切换

**同Bank不同行的访问，是DDR3访问效率最低的一种情况**，以两次跨行写访问 为例，写数据完成后，tWR时间之后，才能关闭本行。关闭后，需要tRP时间之后，才能重新激活另一行，激活后，等待tRCD之后，才能再次发送写命令。写命令之后， CWL时间之后，才能发送写数据。因此应该尽可能降低行切换出现的次数。

tRP：预充电需要precharge的原因，关闭当前行，打开新行时，要求新的bit line充电到VDD/2

（4）跨行数据

当一次突发操作待写入/待读出的数据刚好处在一行的结尾，数据的一部分在下一 行，如果下一行没有激活，就会造成数据丢失的情况，后半部分数据就不会生效。

因为DDR3地址复用的特性，在访问一个地址前先要进行激活操作（即行地址选通），再进行读/写操作（即列地址选通），对于读/写操作有以下三种情况：

1）页冲突（Page Conflict）：即**访问同Bank不同行（Bank已激活）**。要访问Bank 中已经有一行打开，访问地址处于不同行，这时要先关闭（Precharge）已打开的行， 再打开（Active）新的行地址，最后发送读/写命令。

tRP：Row Precharge Time 表示行预充时间，用来设定在另一行能被激活之前，现有的工作行需要充电时间

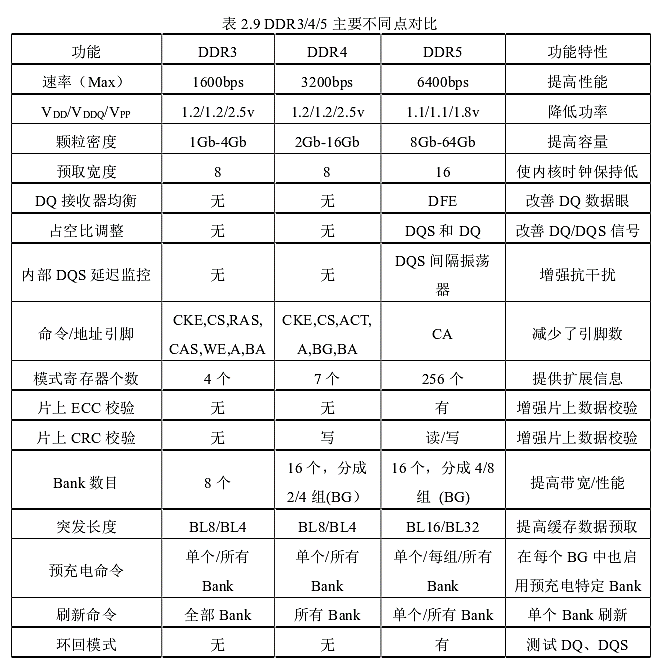
2）页丢失（Page Miss）：即**访问Bank的某一行（Bank未激活）**。要访问的Bank 是未激活的状态，可能是第一次对该Bank进行访问，也可能是上一次对该Bank访问结束，已经关闭了打开的行。这时要先打开（Active）访问的行地址，再发送读/写指令。

3）页命中（Page hit）：即**访问同Bank同行（Bank已激活）**。要访问的行地址在之前操作中已经打开，这时再访问该行就不用再进行激活，可以直接发送读/写命令。

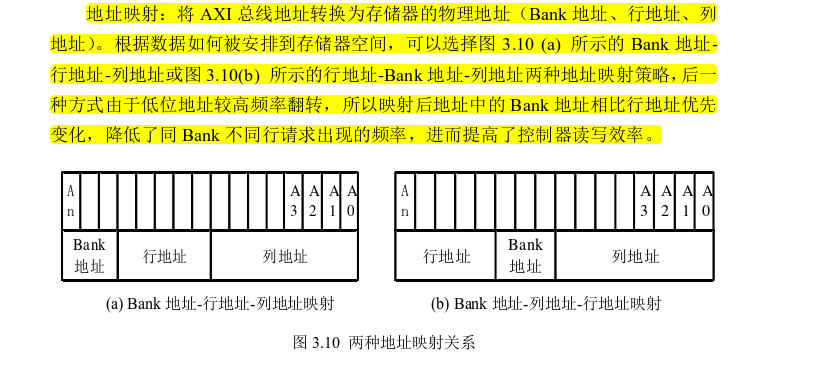
以DDR3-1600为例，对三种不同读/写寻址情况延迟计算如表所示。



尽可能减少Page Conflict出现的几率，增大Page Hit出现的几率。因此读/写重排序模块的功能是对读/写命令进行重新排序，当新进入的命令和刚 才出去的命令是同Bank同Row，就要对新进入命令插队，以便优先处理Page Hit命 令，同时将Page Conflict的两个命令分隔开。



## 4.7 User interface与Memory的映射方式



## 4.8 性能估计

