**Command Per Clock(CPC)**

Command Per Clock(CPC：指令比率，也有翻译为： **首命令延迟**)，一般还被描述为DRAM Command Rate、CMD Rate等。由于目前的DDR内存的寻址，先要进行P-Bank的选择（通过DIMM上CS片选信号进行），然后才是L-Bank/行激活与列地址的选择。这个参数的含义就是指在P-Bank选择完之后多少时间可以发出具体的寻址的L-Bank/行激活命令，单位是时钟周期。　该参数的默认值为Disable(2T)，

**CAS Latency Control(tCL)**

一般我们在查阅内存的时序参数时，如“3-4-4-8”这一类的数字序列，上述数字序列分别对应的参数是“CL-tRCD-tRP-tRAS”。这个3就是第1个参数，即CL参数。

CAS Latency Control(也被描述为tCL、CL、CAS Latency Time、CAS Timing Delay)，CAS latency是“ **内存读写操作前列地址控制器的潜伏时间**”。 **CAS控制从接受一个指令到执行指令之间的时间**。因为CAS主要控制十六进制的地址，或者说是内存矩阵中的列地址，所以它是最为重要的参数，在稳定的前提下应该尽可能设低。内存是根据行和列寻址的，当请求触发后，最初是tRAS（Activeto Precharge Delay），预充电后，内存才真正开始初始化RAS。一旦tRAS激活后，RAS（Row Address Strobe ）开始进行需要数据的寻址。首先是行地址，然后初始化tRCD，周期结束，接着通过CAS访问所需数据的精确十六进制地址。期间从CAS开始到CAS结束就是CAS延迟。所以CAS是找到数据的最后一个步骤，也是内存参数中最重要的。

**RAS# to CAS# Delay(tRCD)**

该值就是“3-4-4-8”内存时序参数中的第2个参数，即第1个4。RAS# to CAS# Delay(也被描述为：tRCD、RAS to CAS Delay、Active to CMD)，表示" **行寻址到列寻址延迟时间**"，数值越小，性能越好。对内存进行读、写或刷新操作时，需要在这两种脉冲信号之间插入延迟时钟周期。

**Min RAS# Active Timing(tRAS)**

表示“ **内存行有效至预充电的最短周期**”，

**Row Precharge Timing(tRP)**

表示 **"** **内存行地址控制器预充电时间**"，预充电参数越小则内存读写速度就越快。

**Row Cycle Time(tRC)**

　　可选的设置：Auto，7-22，步幅值1。

　　Row Cycle Time(tRC、RC)，表示“ **SDRAM行周期时间**”，它是 **包括行单元预充电到激活在内的整个过程所需要的最小的时钟周期数**。

　　其计算公式是：

**row cycle time (tRC) = minimum row active time(tRAS) + row precharge time(tRP)。**

**Row to Row Delay(RAS to RAS delay)(tRRD)**

　　可选的设置：Auto， 0-7，每级以1的步幅递增。

Row to Row Delay，也被称为RAS to RAS delay (tRRD)，表示" **行单元到行单元的延时**"。

**二：DDR操作**

所以寻址的流程是先指定bank地址，再指定行地址(row)，最后指列地址(column)来确定基本存储单元，

DDR3芯片的PAGESIZE = 2^column \* 数据线位宽/8，如：MT41J1G4 – 128 Meg x 4 x 8 banks

PAGESIZE = 2^11 \* 4/8 = 1KB

3  DDR3控制器16bit/32bit概念

这儿所说的16bit/32bit指的是整个内存控制器以多长为单位进行存储，而不是单个DDR3芯片的基本存储单元32bit表示内存控制器以32bit为单位访问内存，即给定一个内存地址。内存芯片会给内存控制器 32bit的数据到数据线上，当然该32bit数据可能不来自同一个DDR3芯片上。下面分析用两个16bit的DDR3内存如何拼成32bit的DDR3

第一片16bit DDR3的BA0,BA1,BA2连接CPU的BA0, BA1, BA2  
第二片16bit DDR3 的BA0,BA1,BA2连接CPU 的BA0, BA1, BA2  
第一片16bit DDR3的A0~A13连接CPU的A0~A13  
第二片16bit DDR3的A0~A13连接CPU的A0~A13  
第一片16bit DDR3的D0~D15连接CPU的D0~D15  
第二片16bit DDR3的D0~D15连接CPU的D16~D31

分析下该实例。  
bank地址线是3bit所以单个16bit DDR3内部有8个bank.  
行地址(row)A0~A13共14bit说明每个bank有2^14行  
列地址(column)A0~A9共10bit说明每个bank有2^10列

每个bank大小是2^14 \* 2^10 \* 16 = 16M \* 16bit = 32MB

每个bank有16M个基本存储单元,总共有16M\*8=128M个地址

单个芯片总大小是 32MB \* 8 = 256MB

从前面的连线可知两块16bit DDR3的BA0~BA2和D0~D14是并行连接到内存控制器，所以内存控制器认为只有一块内存，访问的时候按照BA0~BA2和A0~A13给出地址。两块16bit DDR3都收到了该地址，给出的反应是要么将给定地址上的2个字节读到数据线上，要么是将数据线上的两个字节写入到指定的地址。

此时内存控制认为自己成功的访问的了一块32bit的内存，

所以内存控制器每给出一个地址，将访问4个字节的数据，读取/写入。这4字节数据对应到内存控制器的D0~D31，又分别被连接到两片DDR3芯片的D0~D15，这样32bit就被拆成了两个16bit分别去访问单个DDR3芯片的基本存储单元。

注意：尽管DDR3芯片识别的地址只有128M个，但由于内存控制器每访问一个内存地址，将访问4个字节数据，所以对于内存控制器来说，能访问的内存大小仍是512M，只不过在内存控制器将地址传给DDR3芯片时，低两位被忽略，也就是说DDR3芯片识别的地址只有128M个。  
比如内存控制器访问地址0x00000000，0x00000001，0x00000002，0x00000003，但对DDR3来说，都是访问地址0x00000000。

DQS,DQS#                             I/O

数据选通。读时是输出，边缘与读出的数据对齐。写时是输入，中心与写数据对齐。

DQS是DDR中的重要功能，它的功能主要用来在一个时钟周期内准确的区分出每个传输周期，并便于接收方准确接收数据。每8bit数据都有一个DQS信号线，它是双向的，在写入时它用来传送由内存控制器发来的DQS信号，读取时，则由芯片生成DQS向内存控制器发送。完全可以说，它就是数据的同步信号。

在读取时，DQS与数据信号同时生成（也是在CK与CK#的交叉点）。而DDR内存中的CL也就是从CAS发出到DQS生成的间隔，DQS生成时，芯片内部的预取已经完毕了，由于预取的原因，实际的数据传出可能会提前于DQS发生（数据提前于DQS传出）。

DQS在读取时与数据同步传输

而在接收方，一切必须保证同步接收，不能有偏差。这样在写入时，芯片不再自己生成DQS，而以发送方传来的DQS为基准，并相应延后一定的时间，在DQS的中部为数据周期的选取分割点（在读取时分割点就是上下沿），从这里分隔开两个传输周期。这样做的好处是，由于各数据信号都会有一个逻辑电平保持周期，即使发送时不同步，在DQS上下沿时都处于保持周期中，此时数据接收触发的准确性无疑是最高的。