

SDRAM 的工作原理

(注：本资料内容来源网上，整理目的在于方便大家学习和交流。)

引言

在信息处理中，特别是实时视频图像处理中，通常都要对实现视频图像进行处理，而这首先必须设计大容量的存储器，同步动态随机存储器 [SDRAM](#) 虽然有价格低廉、容量大等优点，但因 [SDRAM](#) 的控制结构复杂，常用的方法是设计 [SDRAM](#) 通用控制器，这使得很多人不得不放弃使用 [SDRAM](#) 而使用价格昂贵的 [SRAM](#)。为此，笔者在研究有关文献的基础上，根据具体情况提出一种独特的方法，实现了对 [SDRAM](#) 的控制，并通过利用 [FPGA](#) 控制数据存取顺序来实现对数字视频图像的旋转，截取、平移等实时处理。[SDRAM](#) 的控制原理，如图 1 所示。

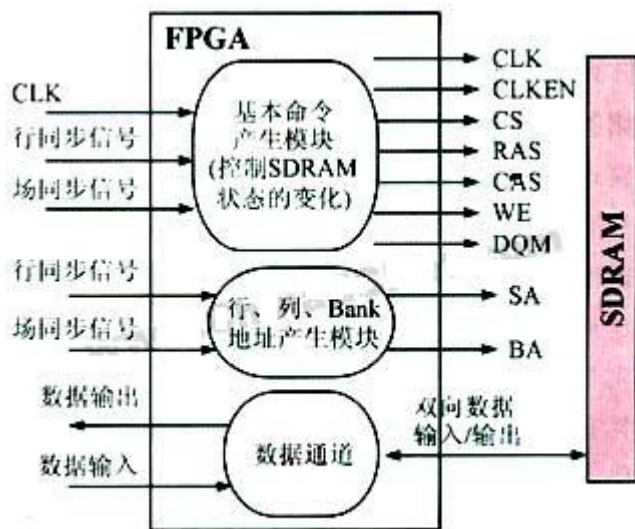


图1 SDRAM控制原理图

SDRAM 控制原理图

[SDRAM](#) 基本操作原理

本文以三星公司的 [SDRAM](#) 器件 [K4S561632C](#)[4]为例来说是说明 [SDRAM](#) 的工作原理。

[SDRAM](#) 的结构特点

存储器的最初结构为线性，它在任何时刻，地址线中都只能有一位有效。设容量为 $N \times M$ 的存储器有 $S_0 - S_{n-1}$ 条地址线；当容量增大时，地址选择线的条数也要线性增多，利用地址译码虽然可有效地减少地址选择线的条数，但这种存储器的长宽比太大，显然，这在工业上是无法实现的。而且由于连线的延时与连线的长度成正比，这样的设计会使存储器的存取速度很慢。为了解决这个问题，现在常用的存储器都是将存储单元设计成阵列形状，使其长宽比接近 1:1。这样，电路就必须增加一个列地址译码器，才能选出正确的存储单元。这样，整个存储器的地址线被分为行地址线和列地址线，行地址线将要选择执行读或写操作的行，而列地址线则可从被选中的一行中再选出一个用于真正执行读或写操作的存储单元。

[SDRAM](#) 的行地址线和列地址线是分时复用的，即地址线要分两次送出，先送行地址线，再送列地址线。这样可进一步减少地址线的数量、提高器件的性能，但寻址过程会由此变得复杂，新型的 [SDRAM](#) 的容量一般比较大，如果还采用简单的阵列结构，就会使存储器的字线和位线的长度、内部寄生电容及寄生电阻都变得很大，从而使整个存储器的存取速度严重下降，实际上，现在 [SDRAM](#) 一般都以 Bank（存储体或存储块）为组织，来将 [SDRAM](#) 分为很多独立的小块，然而由 Bank 地址线 BA 控制 Bank 之间的选择；[SDRAM](#) 的行、列地址线贯穿所有的 Bank；每个 Bank 的数据位宽同整个存储器的相同。这样，Bank 内的字线和位线的长度就可被限制合适的范围内，

从而加快存储器单元的存取速度，另外，BA 也可以使被选中的 Bank 处于正常工作模式，而使没有被选中的 Bank 工作在低功耗模式下，这样还可以降低 [SDRAM](#) 的功耗。

为了减少 MOS 管的数量、降低功耗、提高集成度和存储容量，[SDRAM](#) 都是利用其内部电容存储信息，由于电容的放电作用，必须每隔一段时间给电容充电才能使存储在电容里的数据信息不丢失，这就是刷新过程，这种机制使 [SDRAM](#) 的控制过程变的更加复杂，从而给应用带来难度。

三星公司的 [SDRAM \(K4S561632C\)](#) 的外部同步时钟速率可在一定的频率范围内连续变化，最高频率可达到 133MHz，每块 [SDRAM](#) 内含四个独立的 Bank；它的基本存储单元都是按照阵列排列的，它的数据位宽和整个存储器的位宽相同，同时支持多种读写模式；所有的输入信号均以时钟的上升沿为基准，这使得地址、控制和数据输入到缓冲器的时间可保持一致且建立和保持的时间很小；该器件使用完全流水线型内部结构；另外，它还具有突发长度可编程、延迟可编程等优点。这些优点使得 [K4S561632C](#) 能广泛的应用于宽频带、高性能存储器应用系统。

[SDRAM](#) 的基本信号

[SDRAM](#) 的基本信号可以分成以下几类：

(1) 控制信号：包括片选 (CS)、同步时钟 (CLK)、时钟有效 (CLKEN)、读写选择 (WE)、数据有效 (DQM) 等；

(2) 地址选择信号：包括行地址选择 (RAS)、列地址选择 (CAS)、行/列地址线 (SA0—[SA12](#)) 分时复用、Bank 块地址线 (BA0—BA1)；

(3) 数据信号：包括双向数据端口 (DQ0—DQ15)、接收数据有效信号 (DQM) 控制等。DQM 为低时，写入/读出有效。

对 [SDRAM](#) 的基本命令

要正确的对 [SDRAM](#) 进行操作，就需要输入多种命令：包括模式寄存器设置、预充电、突发停止、空操作等命令。[SDRAM](#) 内部的状态会根据表 1 的命令进行转移，其中，命令 COM={CS#；RAS#；CAS#；WE#}。

模式寄存器的规定

利用模式寄存器 (Mode Register) [5] 可通过装载模式寄存器命令 (LOADMODE REGISTER) 进行编程，这组信息将会一直保存在模式寄存器中，直到它再次被编程或器件掉电为止；它规定了 [SDRAM](#) 的操作模式，包括突发长度、突发类型、CAS 延迟时间、运行模式及写突发模式、具体格式如图 2 所示，模式寄存器 M0—M2 用于规定突发长度 (Burst [LENGTH](#))。M3 用于规定突发类型 BT (Burst [TYPE](#))，M3=0 时，突发类型是连续的；M3=1 时，突发类型是交错的，M4—M6 用于规定 CAS 延迟的时钟周期数，M7—M8 规定运行模式，M9 规定写突发模式 WB (Write Burst Mode)，当 M9=0 时，按实际编程的突发长度存取，当 M9=1 时，则按单个存取单元写入，但可按实际编程的突发长度读出，M10 和 M11 为保留位，可供未来使用。在模式寄存器装载期间，地址 A12 (M12) 必须被驱动至低电平，本方案中，模式寄存器的值为 0x220h。

表1 SDRAM的基本命令

命令名称 (COM)	值	地址线
NOP (空操作)	0111b	可任意
激活 (选择bank并且激活相应的行)	0011b	可任意
读 (选择bank和列地址, 并且开始突发读取)	0101b	Bank/行地址
写 (选择bank和列地址, 并且开始突发写入)	0100b	Bank/列地址
突发终止 (让相应的bank中BANKE失效或禁止该bank失活)	0110b	可任意
预充电 (让相应的bank中的行失活或者让该bank失活)	0010b	视需预充电Bank个数不同进行编码
载入模式寄存器	0000b	模式寄存器的值
刷新	0001b	可任意

注: 前面标“#”标记的表示在低电平下有效

初始化操作

[SDRAM](#) 在上电以后必须先对其进行初始化操作, 而后才能对其进行其他操作。出初始化操作具体步骤如下:

(1) [SDRAM](#) 在上电以后需要等待 100—200 μ s, 在等待时间结束后还至少要执行一条空操作命令;

(2) [SDRAM](#) 执行一条预充电命令后, 要执行一条空操作命令, 这两个操作会使所有的存储单元进行一次预充电, 从而使所有阵列中的器件处于待机状态;

(3) [SDRAM](#) 要执行两条自刷新命令, 每一条刷新命令之后, 都要执行一条空操作命令, 这些操作可使 [SDRAM](#) 芯片内部的刷新及计数器进入正常运行状态, 以便 [SDRAM](#) 为模式寄存器编程做好准备;

(4) 执行加载模式寄存器 (LOAD MODE REGISTER) 命令, 完成对 [SDRAM](#) 工作模式的设定。完成以上步骤后, [SDRAM](#) 即可进入正常工作状态, 以等待外部命令对其进行读、写、预充电和刷新等操作。

[SDRAM](#) 的基本读写操作

[SDRAM](#) 的基本读操作[3]需要控制线和地址线相配合并发出一系列命令来完成。[SDRAM](#) 的读操作只有突发模式 (Burst Mode); 而写操作则可以有突发写和非突发写两种模式, 具体如下:

(1) 带有预充电的突发读写模式, 该模式一次能够访问的列地址的最大数为 1、2、4、8;

(2) 不带有预充电的全页读写、此模式可任意控制一次能够访问的列地址的最大数。根据实际情况, 本设计只选取了与本方法相关的基本操作命令, 并没有设计通用的控制器, 图 3 是本方案的 [SDRAM](#) 内部状态转移图。

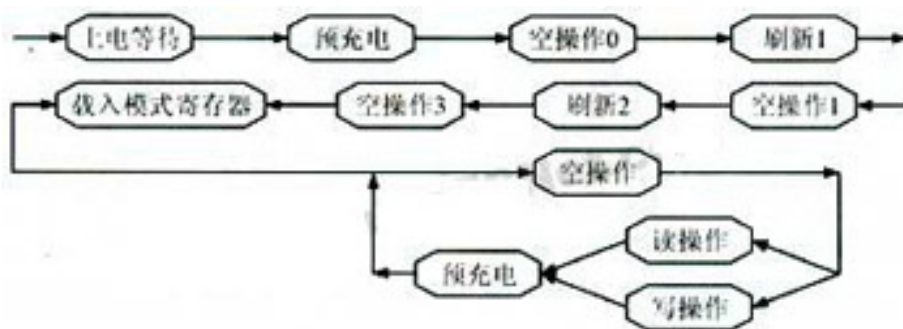


图3 SDRAM内部状态转移图

SDRAM 内部状态转移图

表2 SDRAM内Bank的控制状态

BA的值 (BA0: BA1)	A10 (地址10) 的值	控制状态	被选中的Bank
00b	0	读、写、打开行、预充电	Bank A
01b	0	读、写、打开行、预充电	Bank B
10b	0	读、写、打开行、预充电	Bank C
11b	0	读、写、打开行、预充电	Bank D
任意	1	行预充电	所有Bank

Bank 乒乓操作

由于 Bank 内的行与行之间具有关联性，因此，当其中一个 Bank 的读或写操作结束后，必须执行一次预充电命令以关闭正在操作的行、预充电命令执行后，会有一个 tRP 的延时，延时完成后才能向同一 Bank 行（或其他行）发出新的激活命令。由于 Bank 之间是相互独立的，因此，在一个 Bank 进行正常的读或写操作时，可以对另外几个 Bank 进行预充电或空操作；当一个 Bank 的进行预充电期间也可以直接调用另一个已经进行预充电的 Bank，而并不需要等待，具体的 Bank 控制过程要参考特定的器件数据手册。三星公司的 [K4S561632](#) 内的 Bank 控制状态如图 2 所列。

常用的三种寻址方式

[K4S561632C](#) 器件的常用寻址方法有以下三种：

(1) 页命中 PH (Page Hit)：若寻址的行与所在的 Bank 的空闲的，即该 Bank 内所有的行是关闭的，那么此时便可直接发送行有效命令，这种情况下，数据读取前的总耗时为 tRCD+CL；（tRCD 为 RAS 到 CAS 的延时；CL 为 CAS latency）。

(2) 页快速命中 PFH (Page [FAST](#) Hit) 或页直接命中 PDH (Page Direct Hit)：如果要寻址的行正好是在正常读或写，即要寻址的行正处于被选通的有效状态，那么此时可直接发送列寻址命令，这种情况下，数据读取前的总耗时仅为 CL，这就是所谓的背靠背 ([BACK](#) to [BACK](#)) 寻址；

(3) 页错失 PM (Page Miss)：若要寻址的行所在的 Bank 中已经有一个行处于激活状态（未关闭），这种现象而称为寻址冲突，这样，就必须要进行预充电来关闭正在工作的行、然后再对其他的行发送行有效命令，其总耗时为：tRP+tRCD+CL。（tRP

为 Row precharge time)。

在以上三种寻址方式中,PFH 是最理想的寻址方式,PM 则是最糟糕的寻址方式,实际应用中要尽量采用 PFH 寻址方式而应避免采用 PM 寻址方式。

减少延迟的方法

自动预充电技术是一种有效的减少延迟的方法,它通过自动在每次行操作之后进行预充电操作来减少对同一 Bank 内的不同行寻址时发生冲突的可能性,但是,如果要在正在读或写的行完成操作后马上打开同一 Bank 的另一行时,仍然存在 tRP 的延迟。交错式控制是另一种更有效的减少延迟的方法,即在一个 Bank 工作时,对另一个 Bank 进行预充电或者寻址(此时要寻址的 Bank 是关闭的),预充电与数据的传输交错执行,当访问下一个 Bank 时, tRP 已过,这样就可以直接进入行有效状态,如果配合得比较理想,那么就可以实现无间隔的 Bank 交错读或写,因此, Bank 之间的切换可使存储效率成倍提高,并能够大大地提高多组 SDRAM 协同工作时的性能。

Bank 乒乓操作写入

一个由行、场同步信号控制的计数器在预定的时刻会产生一个控制信号,先发出激活相应 Bank 的激活命令(ACTIVE),并锁存相应的 Bank 地址(由 BA0、BA1 给出)和行地址(由 A0-A12 给出)。一个周期后再给出列地址和写入命令;在 CL 个周期后,便可将所需写入的数据依次送到数据总线上,当计数器的计数周期到达时,系统会产生一个控制信号,并向 SDRAM 发送预充电(PRECHARGE)命令(如果使用了可编程长度,则在这前要使用突发终止命令),以关闭已经激活的页。在下一个视频行同步信号来临时,系统将重复以上操作,并如此循环下去,具体操作如图 4 所示,图中左边的一、二、三……为所对应的视频行同步信号,右边 1 至 511……则代表对应的存储器中的列地址(即是屏幕上对应的像素的位置)。

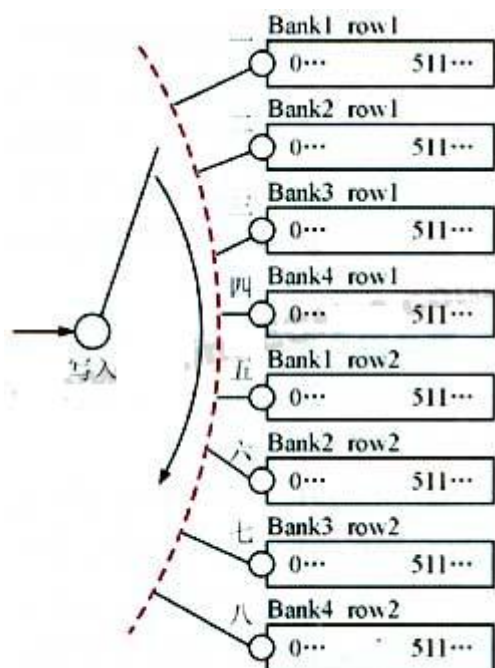


图4 Bank乒乓操作写入示意图

Bank 乒乓操作读出

由行、场同步信号控制的一个计数器可在预定的时刻产生一个控制信号，它首先发出相应的 Bank 激活命令 (ACTIVE)，并锁存相应的 Bank 地址 (由 BA0、BA1 给出) 和行地址 (由 A0-A12) 给出，然后在一个周期后给出列地址和读命令，当计数器的计数周期到达时，系统会产生一个控制信号，并向 SDRAM 发出预充电 (PRECHARGE) 命令，以关闭已经激活的页 (如果使用了可编程长度，则在这之前要使用突发终止命令)。之后，再在下一个视频行同步信号来临时重复以上操作，如此循环 (具体操作如图 5)，图中左边的一、二、三……为所对应的视频行同步信号，右边 1 至 511……代表对应的存储器中的列地址 (即是屏幕上对应的像素的位置)。

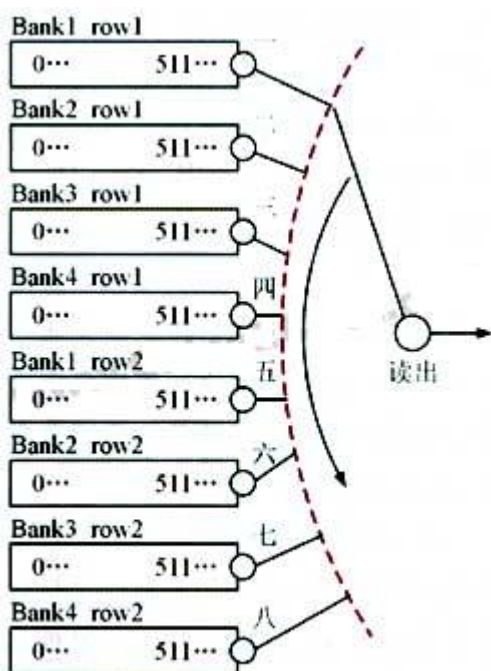


图5 Bank乒乓操作读出示意图

场乒乓操作

为了 SDRAM 能正确进行读和写两个操作，本方案选用两场乒乓操作 [6] 来实现数据存取，实际上，就是在不同的时间对两场轮换进行读或写，一场读而另一场写，其原理如图 6 所示。当开关 K1 在 1 位置，K2 在 4 位置是时，A 写 B 读；反之，当开关 K1 在 3 位置，K2 在 2 位位置时，A 读 B 写。如此循环往复。

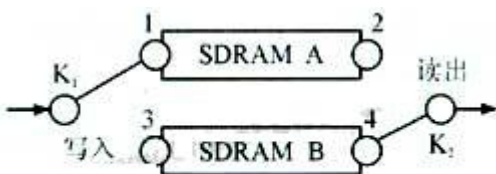


图6 两场乒乓操作原理图

若以场同步信号的二分频计数器 $F/2$ 为读写控制信号，假设 SDRAM A 在 $F/2$ 时为 1 写、2 读，则 SDRAM B 在 $F/2$ 时为 2 写、1 读，两场即为一个场乒乓操作周期，读写信号均在场同步信号为高电平时有效，由于存在消隐期，消音所以，将会有一段时间读写都无效 (读写信号都是低电平，此时 SDRAM 进入预充电状态)，其总体时序如图 7 所示。

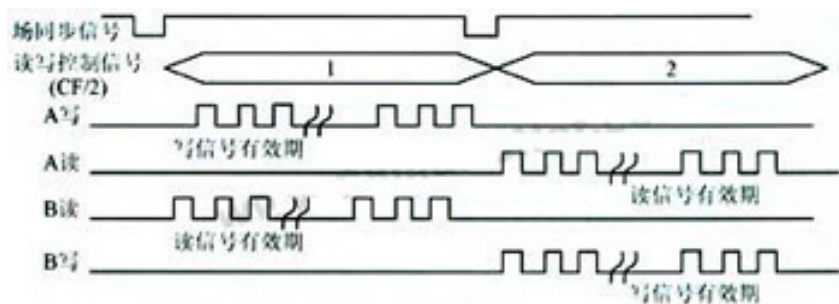


图7 体读写序示意

结束语

[SDRAM](#) 的控制过程虽然很复杂，但如果根据实际情况进行取舍以满足实际系统要求，那么，[SDRAM](#) 的控制过程还是比较简单的，在实验中，使用 [ALTERA](#) 公司的 Cyclone [FPGA](#) 器件进行设计时，程序设计可采用 Verilog 语句来实现对实时视频信号的采集，并通过改变计数器的周期及 [SDRAM](#) 的行、列地址线和时序就可以对任意位置的视频图像进行平移、旋转、截取等处理，此外，由于程序设计采用了化整为零和参数化设计思想，因而结构透明、简单；对于特定容量的 [SDRAM](#) 的特定工作模式而言，该方法只需根据器件重新设定参数而不要重新编写程序，因而具有较强的通用性。

