



(12) 发明专利

(10) 授权公告号 CN 102033823 B

(45) 授权公告日 2014. 03. 19

(21) 申请号 200910173969. 8

US 6701454 B1, 2004. 03. 02, 全文.

(22) 申请日 2009. 09. 27

CN 101452420 A, 2009. 06. 10, 全文.

(73) 专利权人 中兴通讯股份有限公司

审查员 李娇

地址 518057 广东省深圳市南山区高新技术产业园科技南路中兴通讯大厦法务部

(72) 发明人 吴清政 朱宝旺 金铁军

(74) 专利代理机构 北京同达信恒知识产权代理有限公司 11291

代理人 黄志华

(51) Int. Cl.

G06F 12/16(2006. 01)

(56) 对比文件

CN 1725187 A, 2006. 01. 25, 说明书第 5 页第 12-22 行, 说明书第 7 页第 12-14 行, 说明书第 10 页第 18-22 行.

CN 1530795 A, 2004. 09. 22, 说明书第 1 页第 21-23 行.

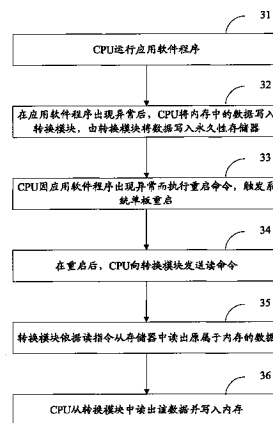
权利要求书1页 说明书6页 附图8页

(54) 发明名称

一种保存数据的方法及装置

(57) 摘要

本发明公开了一种保存数据的方法, 用于实现在系统重启后, 内存中的数据不丢失。该方法由系统单板、转换模块、内存和永久性存储器实现, 其中系统单板包括 CPU; 所述方法包括以下步骤: CPU 运行应用软件程序; 在出现异常后, CPU 将内存中的数据写入转换模块, 由转换模块将数据写入永久性存储器; CPU 执行重启命令, 触发系统单板重启; 在重启后, CPU 向转换模块发送读命令; 转换模块依据读指令从存储器中读出原属于内存的数据; CPU 从转换模块中读出该数据并写入内存。本发明还公开了用于实现所述方法的装置。



1. 一种保存数据的方法,其特征在于,该方法由系统单板、转换模块、内存和永久性存储器实现,其中系统单板包括 CPU ;所述方法包括以下步骤:

CPU 运行应用程序;

在应用程序出现异常后,CPU 将内存中的数据写入转换模块,由转换模块将数据写入永久性存储器;

CPU 因应用程序出现异常而执行重启命令,触发系统单板重启;

在重启后,CPU 向转换模块发送读命令;

转换模块依据读命令从永久性存储器中读出原属于内存的数据 ;CPU 从转换模块中读出该数据并写入内存;

其中,所述 CPU 将内存中的数据写入转换模块,由转换模块将数据写入永久性存储器,具体为 :转换模块依据来自系统单板的时钟信号,生成并向永久性存储器发送时钟信号,使永久性存储器与转换模块保持时序一致,CPU 依据系统单板的时序和总线的数量将内存中的数据写入转换模块,由转换模块依据永久性存储器的地址线和数据线的宽度将数据写入永久性存储器。

2. 如权利要求 1 所述的方法,其特征在于,CPU 向转换模块发送读命令之前,系统单板重启后产生复位信号,触发转换模块复位。

3. 如权利要求 1 所述的方法,其特征在于,CPU 向转换模块发送读命令的步骤包括 :CPU 通过系统单板与转换模块之间的帧信号向转换模块发送读命令;

转换模块依据读命令从永久性存储器中读出原属于内存的数据之前,通过与永久性存储器之间的控制信号将读命令发送给永久性存储器。

4. 一种计算机设备,其特征在于,包括:

永久性存储器,用于保存数据;

内存,用于保存数据;

系统单板中的 CPU,用于运行应用程序,在出现异常后,将内存中的数据写入转换模块;执行重启命令,触发系统单板重启,在重启后,向转换模块发送读命令,以及从转换模块中读出数据并写入内存;

转换模块,用于将数据写入永久性存储器;以及依据读命令从永久性存储器中读出原属于内存的数据;

其中,所述 CPU 将内存中的数据写入转换模块,由转换模块将数据写入永久性存储器,具体为 :转换模块依据来自系统单板的时钟信号,生成并向永久性存储器发送时钟信号,使永久性存储器与转换模块保持时序一致,CPU 依据系统单板的时序和总线的数量将内存中的数据写入转换模块,由转换模块依据永久性存储器的地址线和数据线的宽度将数据写入永久性存储器。

5. 如权利要求 4 所述的计算机设备,其特征在于,系统单板重启后产生复位信号,触发转换模块复位。

6. 如权利要求 4 所述的计算机设备,其特征在于,CPU 通过系统单板与转换模块之间的帧信号向转换模块发送读命令;

转换模块通过与永久性存储器之间的控制信号将读命令发送给永久性存储器。

一种保存数据的方法及装置

技术领域

[0001] 本发明涉及计算机领域,特别是涉及保存数据的方法及装置。

背景技术

[0002] 随着计算机技术的发展,各行各业都在利用计算机来提高办公效率。各种应用软件也层出不穷。然而不能保证所有的软件都百分百的运行正常,时而出现大大小小的问题。

[0003] 其中,有些问题可能只要重启软件即可,有些问题可能导致系统重启。当因故障而重启系统时,内存中的数据无法保留,也就是说故障现场的相关信息无法被记录和分析,给后续的故障跟踪、定位和排除带来很大的难度。以及,可能造成重要数据的丢失。

[0004] 现有技术通过采用内存自刷新方式来解决该问题。这需要芯片组在硬件上支持内存自刷新,当需要系统重启时,触发内存条进入自刷新模式。在重启后,基本输入输出系统(Basic Input Output System, BIOS)不初始化保留段内的内存空间。这样,可以将重要的数据存储保留段内,则不会因系统重启而丢失数据,并且可以对这部分数据进行分析从而定位故障。该技术需要芯片组、BIOS和内存条在硬件上的支持。

[0005] 然而,目前只有少量的芯片组、BIOS和内存条可支持该技术,对于不支持该技术的设备,因故障而重启系统后,仍然无法保存内存中的数据。

发明内容

[0006] 本发明实施例提供一种保存数据的方法及装置,用于实现在系统重启后,内存中的数据不丢失。

[0007] 一种保存数据的方法,该方法由系统单板、转换模块、内存和永久性存储器实现,其中系统单板包括CPU;所述方法包括以下步骤:

[0008] CPU运行应用软件程序;

[0009] 在出现异常后,CPU将内存中的数据写入转换模块,由转换模块将数据写入永久性存储器;

[0010] CPU执行重启命令,触发系统单板重启;

[0011] 在重启后,CPU向转换模块发送读命令;

[0012] 转换模块依据读指令从存储器中读出原属于内存的数据;CPU从转换模块中读出该数据并写入内存;

[0013] 其中,所述CPU将内存中的数据写入转换模块,由转换模块将数据写入永久性存储器,具体为:所述转换模块依据来自所述系统单板的LPC时钟信号,生成并向所述永久性存储器发送时钟信号,使所述永久性存储器与所述转换模块保持时序一致,所述CPU依据所述系统单板的时序和总线的数量将所述内存中的数据写入所述转换模块,由所述转换模块依据所述永久性存储器的地址线和数据线的宽度将数据写入所述永久性存储器。

[0014] 一种计算机设备,包括:

[0015] 存储器,用于保存数据;

[0016] 内存模块,用于保存数据;

[0017] 系统单板中的 CPU,用于运行应用软件程序,在出现异常后,将内存中的数据写入转换模块;执行重启命令,触发系统单板重启,在重启后,向转换模块发送读命令,以及从转换模块中读出数据并写入内存;

[0018] 转换模块,用于将数据写入永久性存储器;以及依据读指令从存储器中读出原属于内存的数据;

[0019] 其中,所述 CPU 将内存中的数据写入转换模块,由转换模块将数据写入永久性存储器,具体为:所述转换模块依据来自所述系统单板的 LPC 时钟信号,生成并向所述永久性存储器发送时钟信号,使所述永久性存储器与所述转换模块保持时序一致,所述 CPU 依据所述系统单板的时序和总线的数量将所述内存中的数据写入所述转换模块,由所述转换模块依据所述永久性存储器的地址线和数据线的宽度将数据写入所述永久性存储器。

[0020] 本发明实施例中,应用软件出现异常,在系统单板重启前,将内存中的重要数据写入外挂的永久性存储器,在系统单板重启后,再将存储器中的数据读到内存。避免了因重启而导致的内存中的数据丢失。

附图说明

[0021] 图 1 为本发明实施例中计算机设备的结构图;

[0022] 图 2 为本发明实施例中转换模块的结构图;

[0023] 图 3A 为本发明实施例中保存数据的主要方法流程图;

[0024] 图 3B 为本发明实施例中保存数据的详细方法流程图;

[0025] 图 4 为本发明实施例中从内存中读出数据的方法流程图;

[0026] 图 5 为本发明实施例中 LPC 的存储模式写时序图;

[0027] 图 6 为本发明实施例中 fireware 的存储模式写时序图;

[0028] 图 7 为本发明实施例中将数据写回内存的方法流程图;

[0029] 图 8 为本发明实施例中 LPC 的存储模式读时序图;

[0030] 图 9 为本发明实施例中 fireware 的存储模式读时序图。

具体实施方式

[0031] 本发明实施例中,应用软件出现异常,在系统单板重启前,将内存中的重要数据写入外挂的永久性存储器,在系统单板重启后,再将存储器中的数据读到内存。避免了因重启而导致的内存中的数据丢失。

[0032] 参见图 1,本实施例中的计算机设备包括应用软件模块 101、检测模块 102、写模块 103、转换模块 104、系统单板 105、读模块 106、存储器 107 和内存模块 108。其中,系统单板 105 与转换模块 104 连接,转换模块 104 与存储器 107 连接。系统单板 105 包括 CPU 和芯片组。应用软件模块 101、检测模块 102、写模块 103 和读模块 106 是运行在 CPU 内且由软件实现的模块。转换模块 104 可具体为现场可编程门阵列(Field-Programmable Gate Array,FPGA)或复杂可编程逻辑器件(Complex Programmable Logic Device,CPLD)等。存储器 107 可以是同步静态随机存取存储器(Synchronous static random access memory,SSRAM)等。内存模块 108 由内存条实现。

[0033] 应用软件模块 101 包括应用软件程序,在应用软件程序出现异常后,生成并发送异常信号。

[0034] 检测模块 102 用于在检测到异常信号后,触发写模块 103。

[0035] 写模块 103 用于将内存 108 中的数据写入转换模块 104。具体的,写模块 103 依据系统单板 105 的时序和总线的数量将内存 108 中的数据写入转换模块。其中通过系统单板 105 产生的 LPC 时钟信号,使转换模块 104 与写模块 103 保持时序一致。写模块 103 在完成写操作后,触发应用软件模块 101,使应用软件模块 101 触发 CPU 执行重启命令。

[0036] 转换模块 104 用于将数据写入永久性存储器 107;以及依据读指令从存储器 107 中读出原属于内存 108 的数据。具体的,转换模块 104 依据永久性存储器 107 的地址线和数据线的数量将数据写入永久性存储器。并且,转换模块 104 依据来自系统单板 105 的 LPC 时钟信号,生成并向存储器 107 发送时钟信号,使存储器 107 与转换模块 104 保持时序一致。在系统单板 105 重启后,转换模块 104 通过与存储器 107 之间的控制信号将读指令发送给存储器 107,然后从存储器 107 中读出原属于内存 108 的数据。

[0037] 系统单板 105 用于在应用软件程序出现异常后重启。系统单板 105 重启后产生复位信号,触发转换模块复位。

[0038] 读模块 106 用于在系统单板 105 重启后自动启动,并向转换模块 104 发送读命令;以及从转换模块 104 中读出数据并写入内存 108。其中,读模块 106 通过系统单板 105 与转换模块 104 之间的帧信号向转换模块 104 发送读命令。

[0039] 存储器 107 用于保存数据。

[0040] 内存模块 108 用于保存数据。

[0041] 系统单板 105 通过地址数据复用总线与其它模块传递地址和数据,该地址数据复用总线通常是 4 位。而存储器 107 通过地址线与其它模块传递地址,通过数据线与其它模块传递数据,即地址和数据通过不同的线路传输。并且通常情况下,存储器 107 的地址线和数据线的位数高于 4 位。这需要转换模块 104 在系统单板 105 与存储器 107 之间进行时序转换。下面对转换模块 104 的结构等进行介绍。

[0042] 参见图 2,本实施例中转换模块 104 包括总线接口 201、地址单元 202、写数据单元 203、读数据单元 204、时钟单元 205、控制单元 206 和复位接口 207。

[0043] 总线接口 201 与系统单板 105 的地址数据复用总线连接,传输地址和数据。

[0044] 地址单元 202 用于接收来自总线接口 201 的地址并保存,当收到的地址满足地址线的宽度时,通过地址线将地址发送给存储器 107。

[0045] 写数据单元 203 用于接收来自总线接口 201 的数据并保存,当收到的数据满足数据线的宽度时,通过数据线将数据写入存储器 107。

[0046] 读数据单元 204 用于通过数据线从存储器 107 中读出数据并保存,依据地址数据复用总线的宽度将数据发送给总线接口 201。

[0047] 时钟单元 205 用于接收系统单板 105 产生的时钟信号,并依据该时钟信号控制地址单元 202、写数据单元 203 和读数据单元 204 的时序,以及向存储器 107 发送时钟信号,使存储器 107 保持时序一致。

[0048] 控制单元 206 用于接收系统单板 105 产生的帧信号,帧信号包括写命令和读命令等。控制单元 206 依据帧信号生成相应的控制信号并发送给存储器 107。

[0049] 复位接口 207 用于接收系统单板 105 产生的复位信号,并受该复位信号的触发,复位转换模块 104。

[0050] 以上介绍了计算机设备的内部结构和功能,下面对保存数据的过程进行详细介绍。

[0051] 参见图 3A,本实施例中保存数据的主要方法流程如下:

[0052] 步骤 31:CPU 运行应用软件程序。

[0053] 步骤 32:在应用软件程序出现异常后,CPU 将内存中的数据写入转换模块,由转换模块将数据写入永久性存储器。

[0054] 步骤 33:CPU 因应用软件程序出现异常而执行重启命令,触发系统单板重启。

[0055] 步骤 34:在重启后,CPU 向转换模块发送读命令。

[0056] 步骤 35:转换模块依据读指令从存储器中读出原属于内存的数据。

[0057] 步骤 36:CPU 从转换模块中读出该数据并写入内存。

[0058] 由于 CPU 是通过运行程序来实现功能,这些程序可划分为多个模块,下面通过这些模块来描述保存数据的过程。

[0059] 参见图 3B,本实施例中保存数据的详细方法流程如下:

[0060] 步骤 301:应用软件程序出现异常后,生成并发送异常信号。

[0061] 步骤 302:检测模块在检测到异常信号后,触发写模块。

[0062] 步骤 303:写模块将内存中的数据写入转换模块,由转换模块将数据写入永久性存储器。

[0063] 步骤 304:应用软件程序的异常引起系统单板重启,并在重启后,自动启动读模块。

[0064] 步骤 305:读模块向转换模块发送读命令。

[0065] 步骤 306:转换模块依据读指令从存储器中读出原属于内存的数据;读模块从转换模块中读出该数据并写入内存。

[0066] 参见图 4,本实施例中从内存中读出数据的方法流程如下:

[0067] 步骤 401:应用软件程序出现异常,应用软件模块 101 生成异常信号并发送。

[0068] 步骤 402:检测模块 102 检测到异常信号并触发写模块 103。

[0069] 步骤 403:写模块 103 通过系统单板 105 向转换模块 104 发送写命令。

[0070] 步骤 404:转换模块 104 向存储器 107 发送写命令。

[0071] 步骤 405:写模块 103 依据系统单板的时序通过系统单板 105 向转换模块 104 发送地址和内存模块 108 中的数据。例如地址数据复用总线的宽度是 4 位。存储器 107 的地址线宽度是 20 位,数据线宽度是 8 位。

[0072] 步骤 406:转换模块 104 依据存储器 107 的地址线和数据线的宽度缓存多帧地址和数据,直到满足存储器 107 的地址线宽度和数据线宽度。例如,转换模块 104 缓存 5 帧地址和 2 帧数据。

[0073] 步骤 407:转换模块 104 向存储器 107 发送地址和数据。

[0074] 步骤 408:应用软件模块 101 触发系统单板 105 重启。

[0075] 在步骤 405 中,系统单板 105 有多种工作模式,如低脚位数 (Low Pin Count,LPC) 的存储 (memory) 模式和硬件 (fireware) 的存储模式等。不同的工作模式有不同的时序。

参见图 5 和图 6 所示的 LPC 的存储模式写时序图和 firmware 的存储模式写时序图,在 LPC 的存储模式中,在起始信号 (Start) 后,先传输 32 位地址 (A[31] ~ A[0]),再传输 8 位数据 (D[7] ~ D[0]),然后经过 2 个时钟周期 (TAR) 和一个同步周期 (Sync),再传下一个地址和数据,重复这个过程直到所有重要数据传输完毕。在 firmware 的存储模式中,Preamble 部分包括了起始位和地址位,然后先传低位数据 (Data Lo) 再传高位数据 (Data Hi),经过 1 个时钟周期再传下一个地址和数据。LCK 表示 LPC 总线时钟 ;LFRAME 表示 LPC 总线的帧信号 ;LAD 表示 LPC 总线的数据地址复用总线 ;start 表示起始标志 ;CYCTYPE 表示周期类型 ;TAR 表示转向 (Turn-Around),即 LPC 总线的控制权会在这段时间里转交 ;sync 表示同步,即可在该时段加入等待周期达到外设和主机的同步。其中,重要数据可以是内存模块 108 中的所有数据,也可以是内存模块 108 中预设存储空间内的数据。在步骤 406 中,转换模块 104 缓存多帧地址和数据。例如地址数据复用总线的宽度是 4 位,存储器 107 的地址线宽度是 20 位,一个地址的长度是 32 位。则转换模块 104 每缓存 5 帧地址向存储器 107 发送一次,但是一个地址只占 8 帧,缓存 5 帧后还剩 3 帧,不足 5 帧,此时缓存 3 帧后便可向存储器 107 发送地址。例如,一个数据的长度是 8 位,数据线的宽度也是 8 位,则转换模块 104 每缓存 2 帧数据向存储器 107 发送一次。由于写模块 103 是先向转换模块 104 发送 32 位地址,再发送 8 位数据,所以转换模块 104 也是先向存储器 107 发送地址,再发送数据。

[0076] 可以预先规定存储器 107 中的一段存储空间用于存储内存中的数据,写模块 103 发送的地址就是这段存储空间的地址。读模块 106 向转换模块 104 发送的地址也是这段存储空间的地址。然后转换模块 104 从存储器 107 的这段存储空间中读出数据并返回给读模块 106,由读模块 106 将数据写入内存模块 108。下面对重启后将数据写回内存的过程进行介绍。

[0077] 参见图 7,本实施例中将数据写回内存的方法流程如下:

[0078] 步骤 701:系统单板 105 重启,并自动启动读模块 106。

[0079] 步骤 702:系统单板 105 产生复位信号,触发转换模块 104 复位。产生复位信号的过程可以与启动读模块 106 的过程同时进行。

[0080] 步骤 703:读模块 106 通过系统单板 105 与转换模块 104 之间的帧信号向转换模块 104 发送读命令。

[0081] 步骤 704:转换模块 104 通过与存储器 107 之间的控制信号将读指令发送给存储器 107。

[0082] 步骤 705:读模块 106 通过系统单板 105 将地址发送给转换模块 104。

[0083] 步骤 706:转换模块 104 依据收到的地址从存储器 107 中读出原属于内存的数据。

[0084] 步骤 707:转换模块 104 依据地址数据复用总线的宽度将数据通过系统单板 105 发送给读模块 106。

[0085] 步骤 708:读模块 106 通过系统单板 105 将收到的数据写入内存。

[0086] 在步骤 705-707 中,参见图 8 和图 9 所示的 LPC 的存储模式读时序图和 firmware 的存储模式读时序图,在 LPC 的存储模式中,在起始信号 (Start) 后,读模块 106 先传输 32 位地址 (A[31] ~ A[0]),经过 2 个时钟周期 (TAR) 和一个同步周期 (Sync),转换模块 104 传输 8 位数据 (D[7] ~ D[0]) 给读模块 106,经过 1 个时钟周期再传下一个地址和数据,重复这个过程直到所有重要数据传输完毕。在 firmware 的存储模式中,Preamble 部分包括了

起始位和地址位,读模块 106 先传输 32 位地址 (A[31] ~ A[0]) 给转换模块 104,经过 1 个时钟周期 (TAR) 和一个同步周期 (Sync),转换模块 104 传输 8 位数据 (D[7] ~ D[0]) 给读模块 106,经过 1 个时钟周期再传下一个地址和数据,重复这个过程直到所有重要数据传输完毕。转换模块 104 接收来自读模块 106 的地址的过程同写时序中的描述。关于读数据的过程,例如,地址数据复用总线的宽度是 4 位,数据线的宽度是 8 位,一个数据的长度也是 8 位,则转换模块 104 从存储器 107 中读出一个 8 位数据后,依据地址数据复用总线的宽度,将该数据分为两帧发送给读模块 106,先发送低 4 位后发送高 4 位。

[0087] 用于实现本发明实施例的软件可以存储于软盘、硬盘、光盘和闪存等存储介质。

[0088] 本发明实施例中,应用软件出现异常,在系统单板重启前,将内存中的重要数据写入外挂的永久性存储器,在系统单板重启后,再将存储器中的数据读到内存。避免了因重启而导致的内存中的数据丢失,并且便于对故障发生原因进行定位。

[0089] 显然,本领域的技术人员可以对本发明进行各种改动和变型而不脱离本发明的精神和范围。这样,倘若对本发明的这些修改和变型属于本发明权利要求及其等同技术的范围之内,则本发明也意图包含这些改动和变型在内。

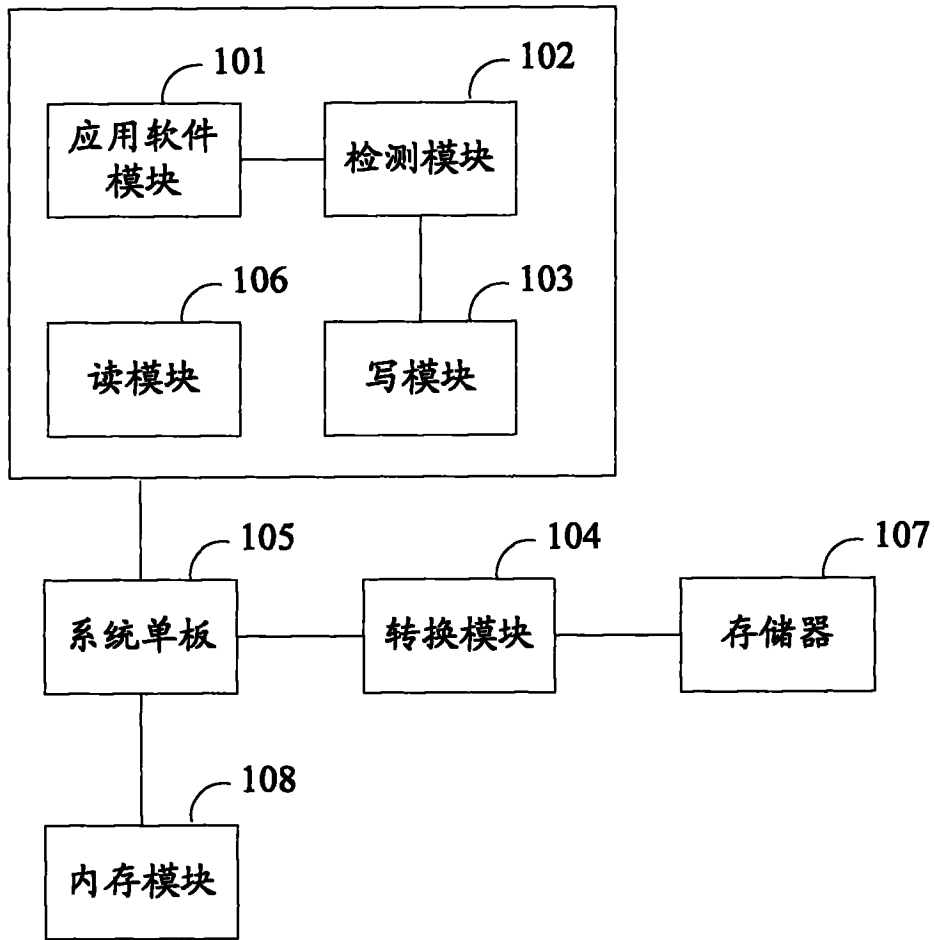


图 1

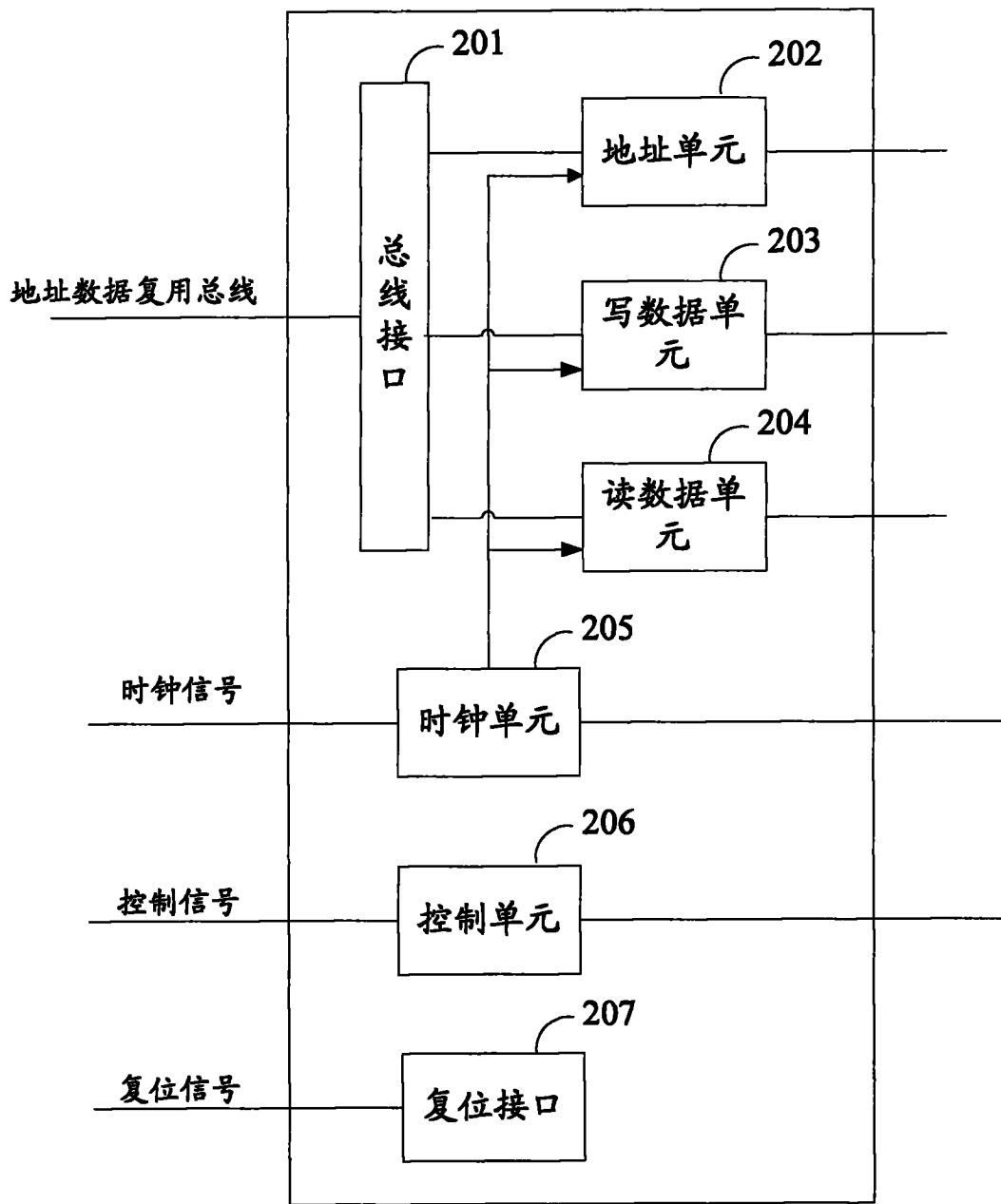


图 2

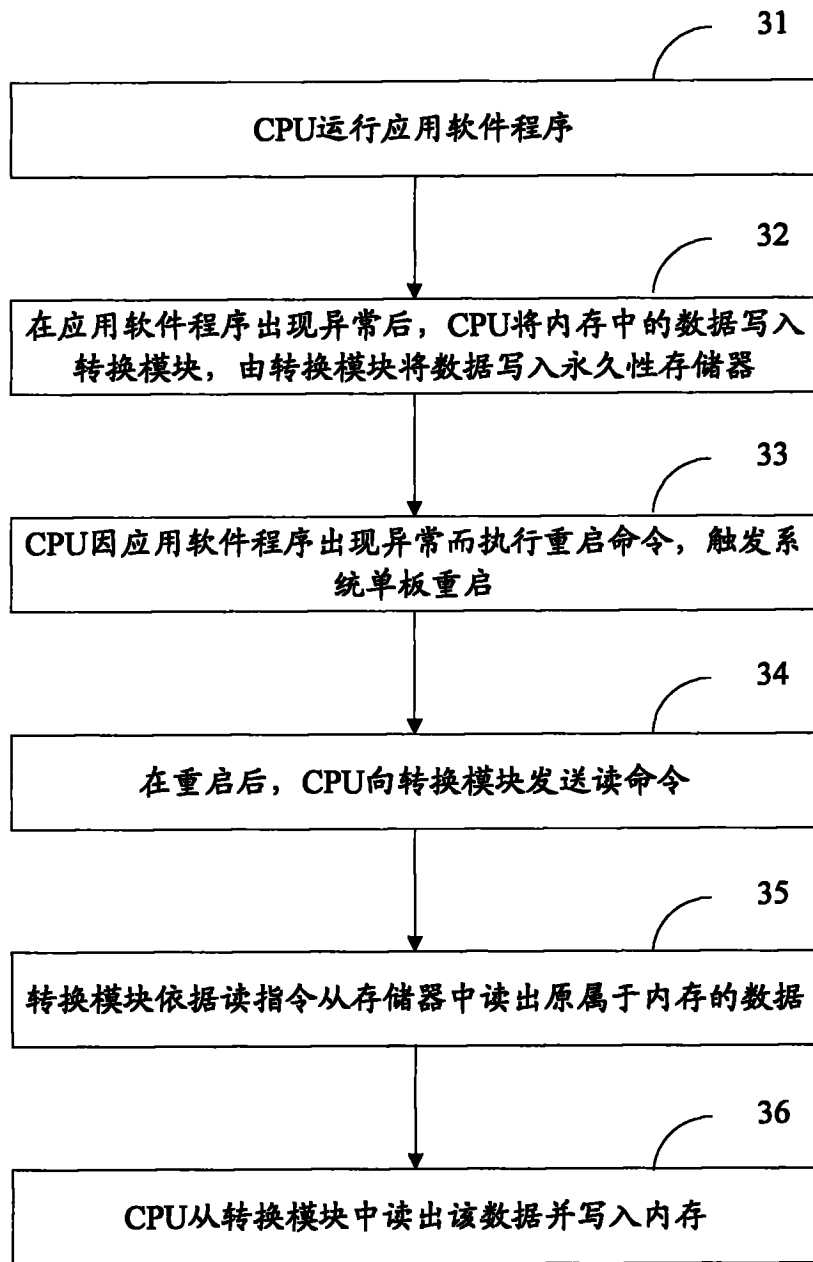


图 3A

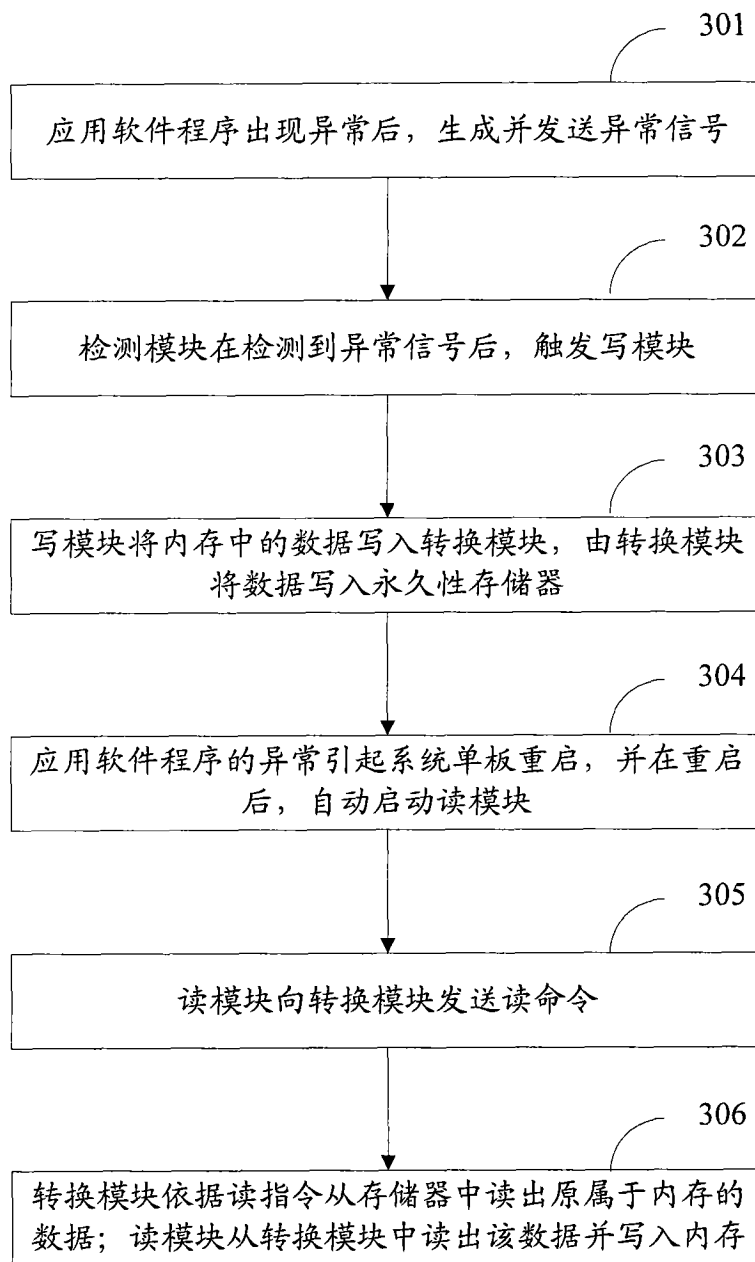


图 3B

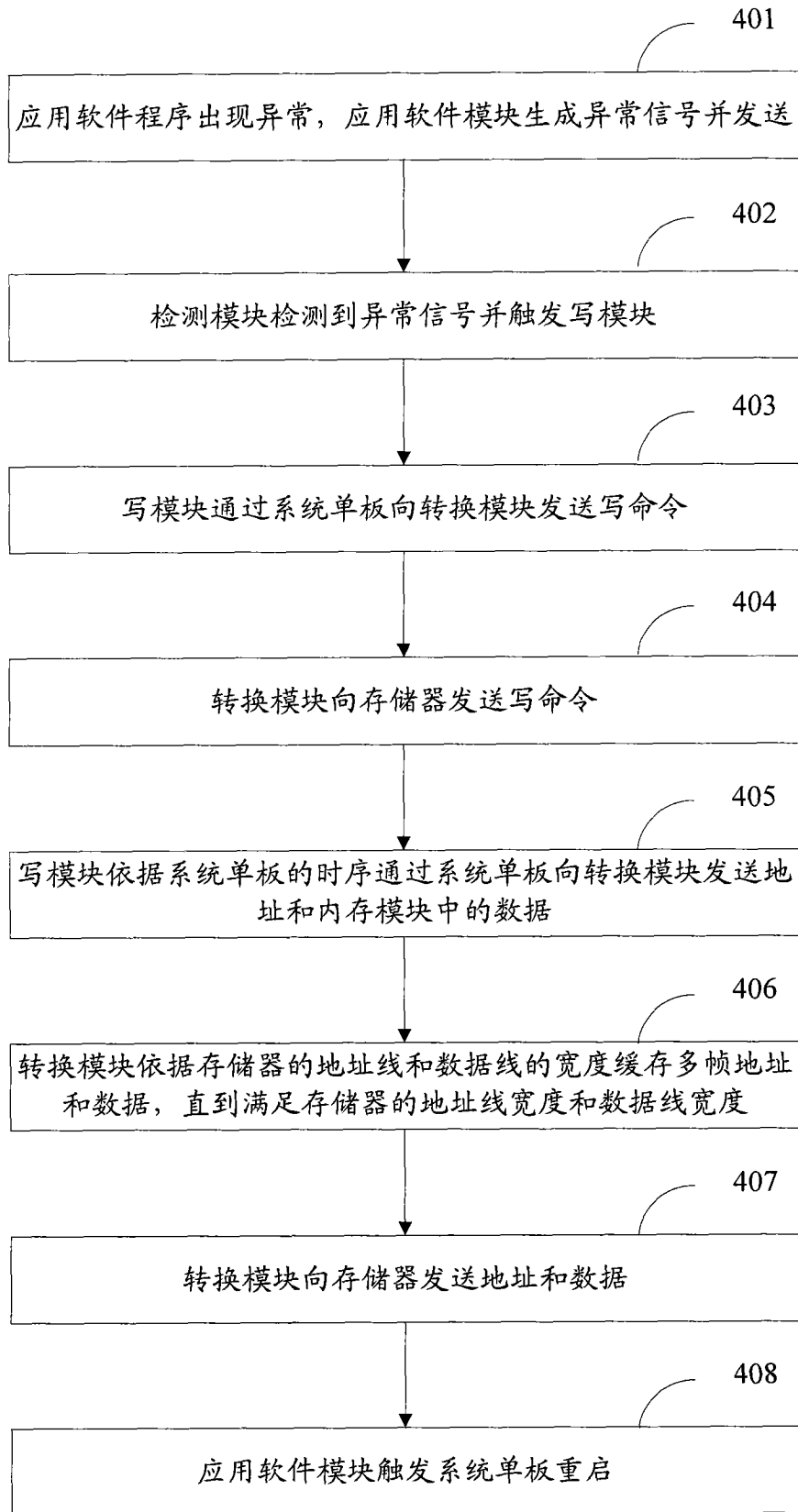


图 4

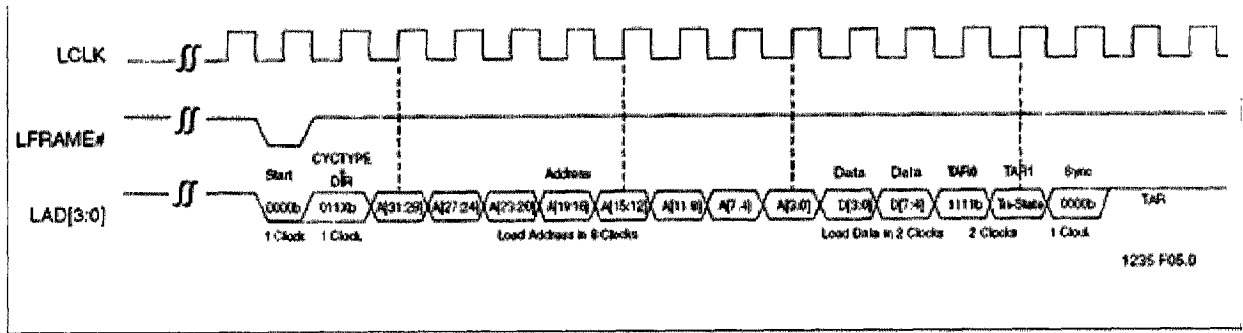


图 5

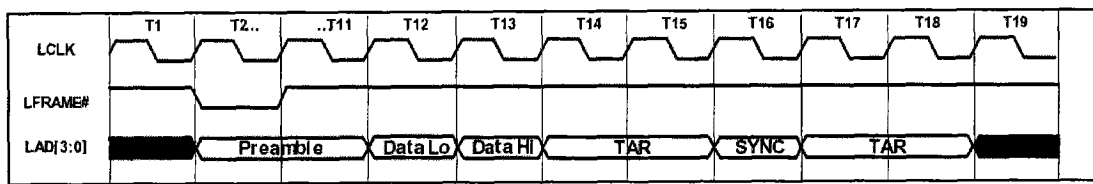


图 6

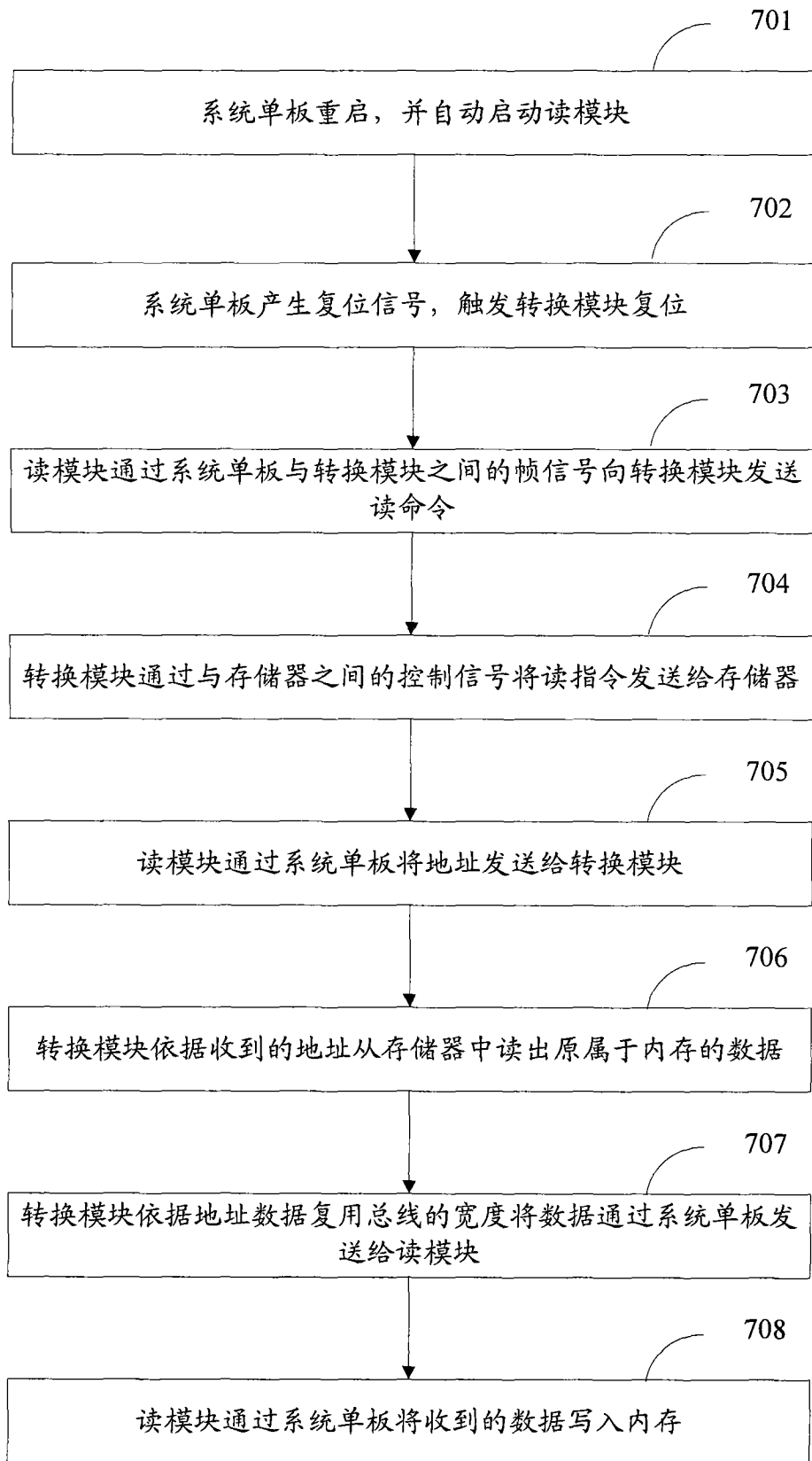


图 7

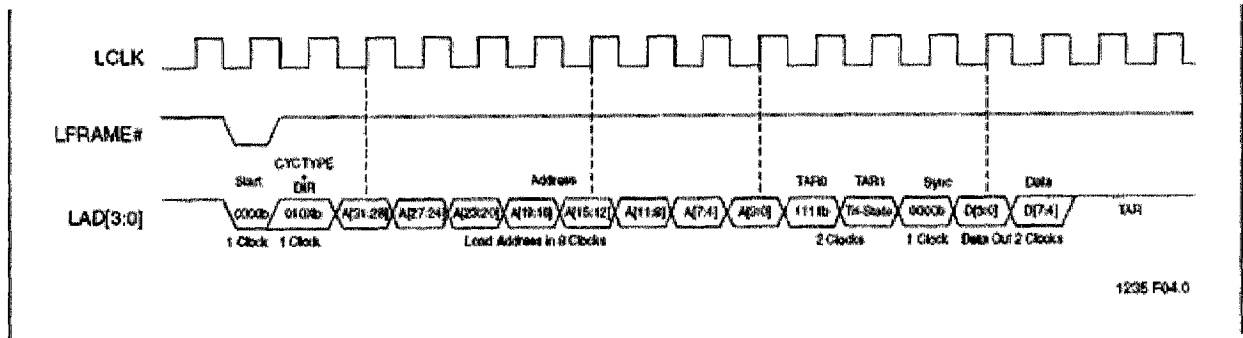


图 8

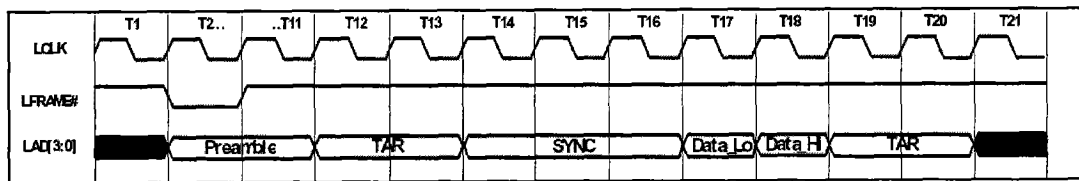


图 9