(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 108460287 A
(43)申请公布日 2018.08.28

(21)申请号 201810236110.6

(22)申请日 2018.03.21

(71)申请人 南通大学
地址 226000 江苏省南通市啬园路9号

(72)发明人 景为平 钱波 景一欧

(74)专利代理机构 南京汇盛专利商标事务所（普通合伙）32238
代理人 吴静安 吴扬帆

(51)Int.Cl.
G06F 21/60(2013.01)
G06F 21/62(2013.01)

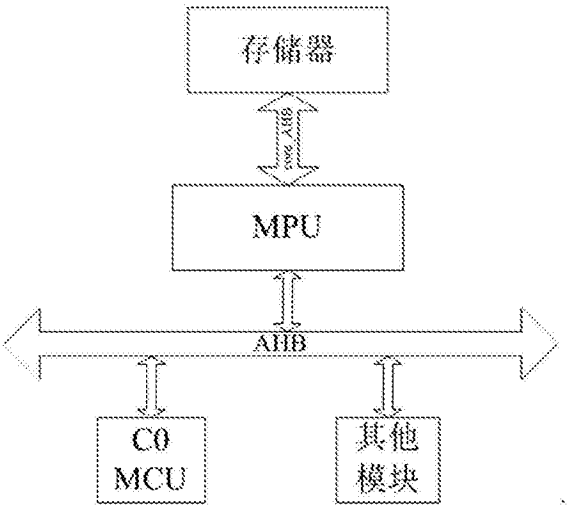
权利要求书1页 说明书5页 附图3页

(54)发明名称

内存保护单元中用户控制区域的划分方法及内存保护系统

(57)摘要

本发明的内存保护单元中用户控制区域的划分方法，在C0 MCU为内核开发出的SoC系统的基础上，设置了芯片内部存储器FLASH的区域权限管理，根据用户的设定将用户与FLASH的相应区域构建关联，形成具有用户属性的区域，并设定各区域的独立的访问权限，在接收用户访问权限时，若区域与用户的对应权限不匹配，则向C0 MCU返回错误信息，阻止不符合的访问。有益效果：从硬件上保证系统资源的所有权，对于资源不被非法访问提供了一种很好的机制，有效的实现了对芯片内部存储器的保护。



CN 108460287 A

1.一种内存保护单元中用户控制区域的划分方法，其特征在于在CO MCU为内核开发出的SoC系统的基础上，设置了芯片内部存储器FLASH的区域权限管理，根据用户的设定将用户与FLASH的相应区域构建关联，形成具有用户属性的区域，并设定各区域的独立的访问权限，在接收用户访问权限时，若区域与用户的对应权限不匹配，则向CO MCU返回错误信息，阻止不符合的访问。

2.根据权利要求1所述的内存保护单元中用户控制区域的划分方法，其特征在于FLASH区域的数据根据用户设定可设置加解密存储，加解密存储操作由对应的使能信号来控制。

3.根据权利要求1所述的内存保护单元中用户控制区域的划分方法，其特征在于所述具有用户属性的区域用户代码区UC和用户数据区UD，不同用户之间可以通过寄存器配置是否允许对应用户的读、写、执行，是否允许其他用户读、写、执行。

4.采用如权利要求1-3任一项所述的内存保护单元中用户控制区域的划分方法的内存保护系统，包括存储器、内存保护单元以及微处理器，所述存储器通过内存保护单元与微处理器通信连接，其特征在于所述内存保护单元包括控制部分、加解密部分、输出部分以及访问违反机制判定部分：

所述控制部分包括：

程序指针比较单元，实现程序指针和各区域配置边界地址的比较，以确定程序所在的区域；

地址比较单元，实现访问地址和各区域配置边界地址的比较，以确定访问地址所在的区域；

访问权限判定单元，以确定实际发生的访问是否符合设定的访问权限；

访问违规判定单元，根据各区域的访问地址和访问权限有效的判定结果，确定用户对目标地址访问是否有效；

所述加解密部分，通过寄存器设定各个区域是否实行加密存储；

所述输出部分，根据控制部分和加解密部分处理的结果，确定是否向周边总线输出访问信息；

所述访问违反机制判定部分，对当前访问的权限进行识别判定，若果当前的权限允许，则可以继续通过总线访问对应的内存单元；如果当前区域的访问权限不允许当前操作，那么就会产生访问权限违反错误。

5.根据权利要求4所述的内存保护系统，其特征在于所述加解密部分包括地址加密与数据加密，地址加密实行乱序加密规则，只处理偏移地址部分。

6.根据权利要求5所述的内存保护系统，其特征在于地址加密的算法中采用了s盒的模板；数据加解密采用SPECK32算法，并对所有数据进行加密。

7.根据权利要求4所述的内存保护系统，其特征在于所述输出部分与控制部分信息交互为：若控制部分判定此次访问有效，则输出访问信息；若控制部分判定访问违规，则阻止此次访问。

内存保护单元中用户控制区域的划分方法及内存保护系统

技术领域

[0001] 本发明涉及嵌入式系统领域，尤其涉及一种内存保护单元中用户控制区域的划分方法及内存保护系统。

背景技术

[0002] 在嵌入式系统经常会遇到多任务的操作和控制在，任务运行的时候，系统中必须提供一种机制来保证正在运行的任务不被其他任务破坏或者影响其他任务的操作。实现上述目的通常有软件保护和硬件保护两种途径。软件保护是指仅靠软件的维护来实现来保护系统资源的作用。在多任务的系统中，通过运行操作系统来达到任务间的同步与通信，以实现软件保护的作用。然而，利用软件来协调多任务的运行，一般会出现一些不可避免的问题。比如，当CPU想要对一个通信串口寄存器进行访问时，如果有其他的任务正在使用该串口，那么这种方式是没有办法来阻止该项操作，若想成功使用此串口，必须通过操作系统来协调，合理控制任务的运行。此类的非法访问一般很容易破坏经过该串口的通信，造成一定的资源浪费和不合理的使用。

[0003] 另一方面来说，如果该系统有专门的硬件来检测和限制系统的资源使其不被非法访问，在一定程度上可以保证资源的所有权。在执行操作任务时，需要遵循硬件维护的规则，对其区域进行合理地配置访问权限，在硬件程度上实现了资源的保护。当CPU访问没有权限的区域时，会主动监视其操作，对于非法访问则会屏蔽。

发明内容

[0004] 本发明目的在于克服上述现有技术的不足，提供了一种内存保护单元中用户控制区域的划分方法及内存保护系统，并主要针对用户控制区域的实现作了一定的验证，具体由以下技术方案实现：

所述内存保护单元中用户控制区域的划分方法，在CO MCU为内核开发出的SoC系统的基础上，设置了芯片内部存储器FLASH的区域权限管理，根据用户的设定将用户与FLASH的相应区域构建关联，形成具有用户属性的区域，并设定各区域的独立的访问权限，在接收用户访问权限时，若区域与用户的对应权限不匹配，则向CO MCU返回错误信息，阻止不符合的访问。

[0005] 所述内存保护单元中用户控制区域的划分方法的进一步设计在于，FLASH区域的数据根据用户设定可设置加解密存储，加解密存储操作由对应的使能信号来控制。

[0006] 所述内存保护单元中用户控制区域的划分方法的进一步设计在于，所述具有用户属性的区域用户代码区UC和用户数据区UD，不同用户之间可以通过寄存器配置是否允许对应用户的读、写、执行，是否允许其他用户读、写、执行。

[0007] 采用所述的内存保护单元中用户控制区域的划分方法的内存保护系统，包括存储器、内存保护单元以及微处理器，所述存储器通过内存保护单元与微处理器通信连接，其特征在于所述内存保护单元包括控制部分、加解密部分、输出部分以及访问违反机制判定部

分：

所述控制部分包括：

程序指针比较单元，实现程序指针和各区域配置边界地址的比较，以确定程序所在的区域；

地址比较单元，实现访问地址和各区域配置边界地址的比较，以确定访问地址所在的区域；

访问权限判定单元，以确定实际发生的访问是否符合设定的访问权限；

访问违规判定单元，根据各区域的访问地址和访问权限有效的判定结果，确定用户对目标地址访问是否有效；

所述加解密部分，通过寄存器设定各个区域是否实行加密存储；

所述输出部分，根据控制部分和加解密部分处理的结果，确定是否向周边总线输出访问信息；

所述访问违反机制判定部分，对当前访问的权限进行识别判定，若果当前的权限允许，则可以继续通过总线访问对应的内存单元；如果当前区域的访问权限不允许当前操作，那么就会产生访问权限违反错误。

[0008] 所述内存保护系统的进一步设计在于，所述加解密部分包括地址加密与数据加密，地址加密实行乱序加密规则，只处理偏移地址部分。

[0009] 所述内存保护系统的进一步设计在于，地址加密的算法中采用了s盒的模板；数据加解密采用SPECK32算法，并对所有数据进行加密。

[0010] 所述内存保护系统的进一步设计在于，所述输出部分与控制部分信息交互为：若控制部分判定此次访问有效，则输出访问信息；若控制部分判定访问违规，则阻止此次访问。

[0011]

本发明的优点如下：

本发明的内存保护单元中用户控制区域的划分方法及内存保护系统从硬件上保证系统资源的所有权，对于资源不被非法访问提供了一种很好的机制，有效的实现了对芯片内部存储器的保护。另外，增加在其中添加了加解密处理模块，增强了数据的机密性。

[0012]

附图说明

[0013] 图1为MPU在整个SoC系统中的结构示意图。

[0014] 图2为内存保护单元的功能示意图。

[0015] 图3为UC区域权限违反的仿真示意图。

[0016] 图4为UD区域权限违反的仿真示意图。

[0017] 图5为FLASH区域数据加密的仿真示意图。

[0018] 图6为FLASH区域地址加密的仿真示意图。

具体实施方式

[0019] 以下结合附图，对本发明的技术方案进行详细说明。

[0020] 如图1,本实施例的内存保护单元中用户控制区域的划分方法,在C0 MCU为内核开发出的SoC系统的基础上,设置了芯片内部存储器FLASH的区域权限管理,在整个MPU模块的设计中,引入了用户来实现管理,每个用户有自己对应的区域,对应的区域设置独立的访问权限,在出现区域与权限不匹配的时候,则会产生对应的违反错误,并且阻止不符合的访问。另外,还设置了独立的加密保护模块,FLASH区域的数据在必要的时候也可以设置加解密存储,其加解密存储有对应的使能信号来控制。在整个模块的设计中,最主要的是各个区域的划分以及对应的控制寄存器的配置,必须保持用户与区域的相对独立,以便于在验证的阶段可以很好的触发错误条件来实现相关功能的验证。下文主要对于用户控制区域的设计做了详细的介绍。

[0021] 本实施例运用了软硬件协同设计的方法,作为其中的内存保护单元,其内部设计架构采用硬件的方式设计,其配置信息采用软件的方式来实现。MPU在整个SoC系统中的作用就是负责保护芯片内部的存储器,当CPU要访问存储器区域时,需要首先经过MPU模块来判断是否有权限去访问该区域。

[0022] 对于用户区域来说,其主要是指有控制权限的FLASH区域。对FLASH区域保护:具有用户属性的区域;用户代码区UC和用户数据区UD。不同用户之间可以通过寄存器配置是否允许本用户读、写、执行,是否允许其他用户读、写、执行。FLASH各用户区域范围可灵活配置,各用户之间范围不允许越界,可以配置FLASH区域是否加密存储。

[0023] 本实施例的内存保护系统,包括存储器、内存保护单元以及微处理器,存储器通过内存保护单元与微处理器通信连接。内存保护单元包括控制部分、加解密部分、输出部分以及访问违反机制判定部分。

[0024] 进一步的,控制部分主要由程序指针比较单元、地址比较单元、访问权限判定单元以及访问违规判定单元组成。程序指针比较单元,实现程序指针和各区域配置边界地址的比较,以确定程序所在的区域。地址比较单元,实现访问地址和各区域配置边界地址的比较,以确定访问地址所在的区域。访问权限判定单元,以确定实际发生的访问是否符合设定的访问权限。访问违规判定单元,根据各区域的访问地址一致性和访问权限有效的判定结果,确定用户对目标地址访问是否有效。

[0025] 加解密部分,可以灵活设计各个区域是否实行加密存储。对于地址部分实行乱序加密规则,由于基地址都是一样的,所以对于地址的加密只处理其偏移地址部分。本实施例在地址加密的算法中采用了s盒,这种s盒的设计模板类似于一种查表算法,每一个数据都有固定的值与其对应。另外,当CPU向存储区域写入数据的时候,可以对其进行相关的加密操作,同样读取数据的时候则进行解密操作。数据加解密采用SPECK32算法,该算法整体采用轮函数结构, F函数包含密钥的线性和非线性的变化。区域的加解密功能由控制寄存器设定。

[0026] 输出部分,根据控制部分和加解密部分处理的结果,确定是否向周边总线输出访问信息。若控制部分判定此次访问有效,则输出访问信息,实现访问目的;若控制部分判定访问违规,则阻止此次访问信息。

[0027] 访问违反机制判定部分,当芯片通过AHB总线访问内存时,MPU通过比对每个区域的高低地址来确定,当前所要访问的地址所属于哪个区域。当对应的地址匹配的时候,当前执行的读写信号就会根据对应区域的访问权限来确定是否可以执行当前的操作。若果当前

的权限语序，则可以继续通过总线访问对应的内存单元；如果当前区域的访问权限不允许当前操作，那么就会产生访问权限违反错误(violation)。

[0028] 用户代码区UC和用户数据区UD的总体功能的实现为：首先配置用户控制寄存器，将对应区域的权限以及相关边界方位设定好，当CPU发起访问，其访问信息首先通过AHB总线进入到MPU模块，然后经过UC、UD模块的控制部分进行有效判定，最终通过输出部分向周边总线输出或屏蔽该次访问信息。

[0029] 表1 UC、UD区域信号描述

信号名称	信号描述	信号类型
core_HADDR	当前执行的区域	输入
core_pc	代码所在的区域	输入
acce_perm	当前执行的操作	输入
uc/ud_valid	UC/UD区域有效的标识	输入
uc_sc_pc_match	本用户区域匹配	输入
uc/ud_regn_perm	UC/UD区域的权限	输入
ucrh/l_reg, udrh/l_reg	UC/UD区域的范围	输入
uc_pc_match	确定代码是否在UC区域	输出
uc/ud_regn_violt	访问UC/UD区域权限违反	输出
uc/ud_addr_match	地址比对匹配	输出

具体实现功能及寄存器描述：

1.根据代码所在的区域与UC/UD区域范围比较，产生match信号，确定代码运行的区域，即确定执行该段代码的用户。

[0030] 2.根据地址与UC/UD区域范围比较，产生地址match信号，确定访问的地址是否在该UC/UD区域。

[0031] 3.根据当前操作类型acce_perm以及处理后的匹配信号，比较该区域允许的操作类型regn_perm，产生权限违反错误(violt)。

[0032] 4.当访问该区域地址匹配并且访问权限错误时，将产生区域违反信号，输出到MPU寄存器模块。

[0033] 对于用户数据区及用户代码区来说，用户寄存器由控制寄存器和状态寄存器组成。控制寄存器主要是用户写入，用来配置对应的区域的访问权限。状态寄存器则反映访问权限错误的信息，当发生访问错误的时候，错误标志位将会被置起，一旦被置为1后便无法再更新为0，必须等到本用户将该错误状态清除后，才能重新恢复为0，从而继续使用。

[0034] 以下对用户区域进行仿真验证

UC、UD区域的仿真及验证也就是用户对UC、UD区域的访问权限的验证以及加解密的验证。

[0035] 用户可以合理设置UC、UD、区域，可以配置4个具有用户属性的区域，用户代码区UC和用户数据区UD各4个。对于UC、UD区域的权限设置及配置有效标识(valid)则由用户控制寄存器分配。对于用户控制寄存器来说，当对应的valid无效时，可直接由CPU配置。用户设置权限配置，不同的用户访问自己和其他用户区域。测试对应用户区是否被安全保护，可通过查询MPU状态寄存器中的状态来判断。

[0036] 1.权限违反的验证

在验证UC、UD区域功能的时候，首先将验证代码放入到事先设置好的用户区域，并对每个用户的代码区和数据区的权限作相关的设定，执行访问其他区域的代码。仿真结果如下面图3与图4所示。由图3、图4可知，将UC和UD区域的权限设置为仅本用户可以读写，其他用户不可以对其进行操作；当其他用户访问该UC/UD区域时，会产生权限错误，阻止此次访问，从而实现保护FLASH的作用。

[0037] 2.加解密的验证

FLASH区域的加密则是可以控制的，当需要加密的时候，只需要将加密控制位使能信号置上即可。波形如下图5和图6所示：可以看到地址只加密其中[6:2]几位，数据则是全部加密。

[0038] 以上所述，仅为本发明较佳的具体实施方式，但本发明的保护范围并不局限于此，任何熟悉本技术领域的技术人员在本发明揭露的技术范围内，可轻易想到的变化或替换，都应涵盖在本发明的保护范围之内。因此，本发明的保护范围应该以权利要求的保护范围为准。

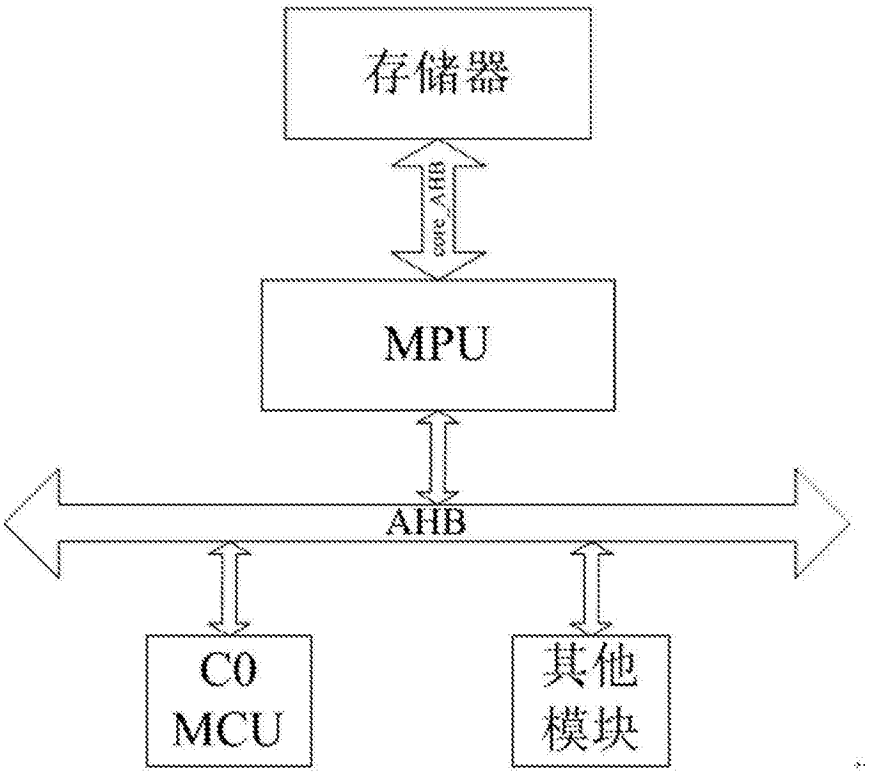

存储器
core AHB
MPU
AHB
C0
MCU
其他
模块

图1

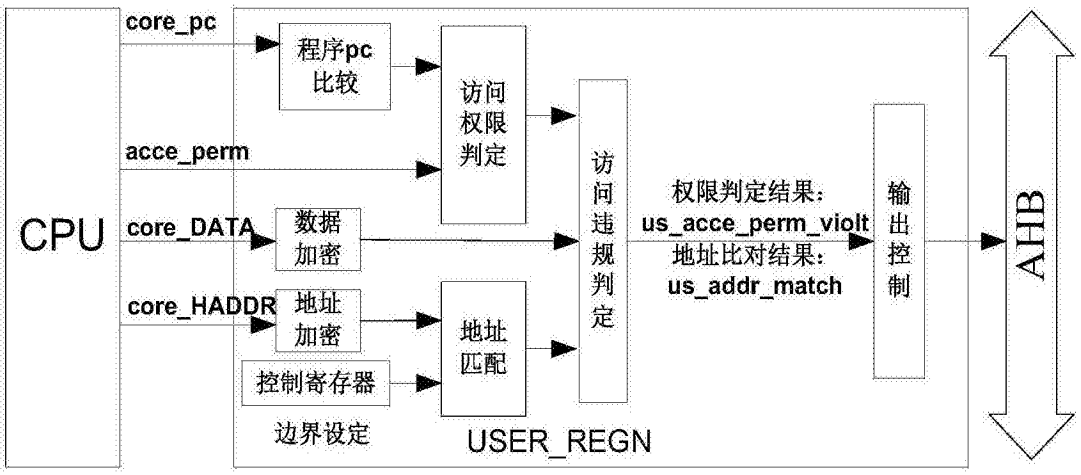

CPU
core_pc
acce_perm
core_DATA
core_HADDR
程序pc
比较
访问
权限
判定
数据
加密
地址
加密
控制寄存器
地址
匹配
访问违规判定
权限判定结果：
us_acce_perm_violt
地址比对结果：
us_addr_match
输出控制
AHB
边界设定
USER_REGN

图2

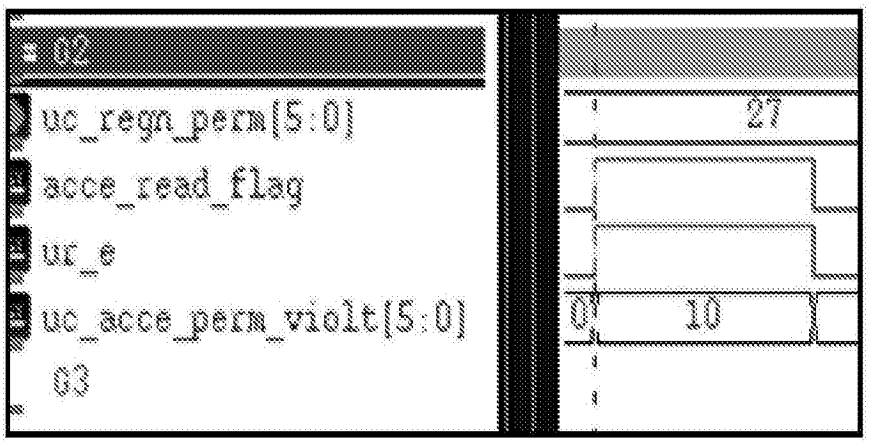

uc_regn_perm[5:0]
acce_read_flag
ur_e
uc_acce_perm_violt[5:0]
27
0
10

图3


ud_regn_perm[3:0]
acce_read_flag
ur_e
ud_acce_perm_violt[4:0]
3
8

图4

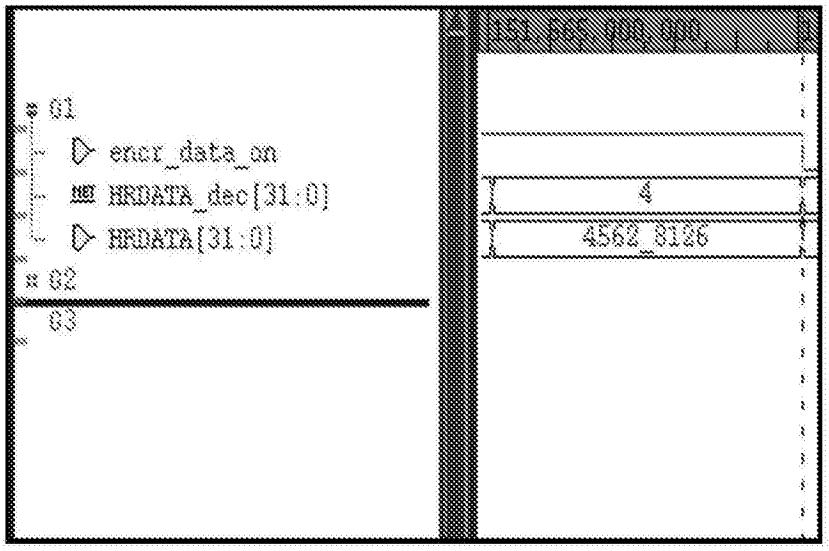

encr_data_en
HRDATA_dec[31:0]
HRDATA[31:0]
4
4562_8126

图5

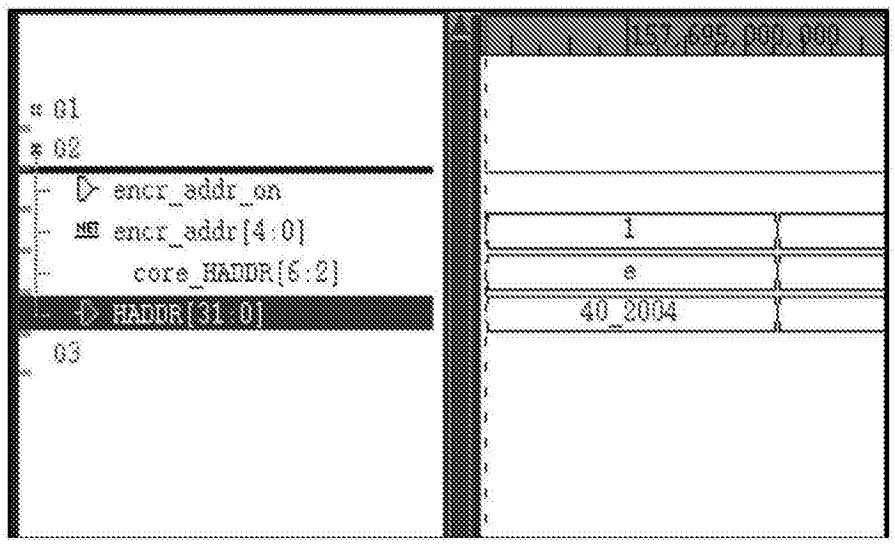

01
02
encr_addr_en
encr_addr[4:0]
core_HADDR[6:2]
HADDR[31:0]
03
1
8
40_2004

图6