(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 103226461 B
(45)授权公告日 2016.07.06

(21)申请号 201310100944.1

(22)申请日 2013.03.26

(73)专利权人 中山大学
地址 510275 广东省广州市新港西路 135 号

(72)发明人 王德明 丁颜玉 丁一 路崇 段志奎 谭洪舟

(74)专利代理机构 广州粤高专利商标代理有限公司 44102
代理人 禹小明

(51)Int.Cl.
G06F 7/72(2006.01)

(56)对比文件
US 6085210 A,2000.07.04,
WO 02/073450 A1,2002.09.19,
CN 1492316 A,2004.04.28,
TW 200842611 A,2008.11.01,
US 2012/0265794 A1,2012.10.18,
CN 1786900 A,2006.06.14,
CN 102207847 A,2011.10.05,
谢元斌,史江一,郝跃. 一种长整数模乘幂的改进算法与实现.《西安电于科技人学学报(自然科学版)》.2011, 第 38 卷(第 2 期),129-134.
Daly A, Marnane W..Efficient architectures for implementing montgomery modular multiplication and RSA modular exponentiation on reconfigurable logic.《Proceedings of the 2002 ACM》.2002,40-49.
蒋晓娜,段成华. 改进的蒙哥马利算法及其模乘法器实现.《计算机工程》.2008, 第 34 卷(第 12 期),209-211.
McLoone M, McCanny J V..Fast Montgomery modular multiplication and RSA cryptographic processor architectures.《Signals, Systems and Computers》.2004,379-384.

审查员 熊菡

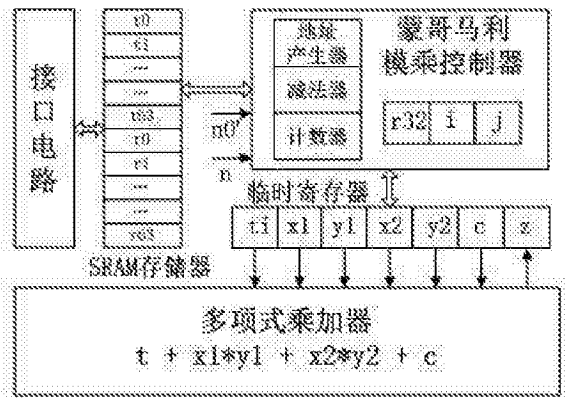
权利要求书2页 说明书6页 附图2页

(54)发明名称

一种用于电路的蒙哥马利模乘方法及其电路

(57)摘要

本发明涉及一种蒙哥马利模乘方法及其电路。方法中,模长度 k=sw,w 为算法每次处理的字长大小,s 为算法 for 循环所需的次数;输入参数包括 a、b 和模数 n,r 为存储器,其高 k 位为存放输入参数 a 和输出结果的 r2,低 k 位为存放输入参数 b 的 r1;t 为多项式临时计算结果,其余为中间变量,运算步骤为:计算 $r2=MonPro(a,b,n)=a*b*2^{k} \bmod n$;将 t 和 i 赋零值;r32=r[i];令 i 为 0 开始外循环;计算(c,z1)=t[i]+r[i]*r32;(c,z1)=z1*n0';x2=z1;x1=r[i];令 j 为 0 开始内循环;计算(c,t[i+j])=t[i+j]+x1*r[i]+x2*n[i]+c;t[i+s]=c;变量 j 加 1,当 j 小于 s 时重复内循环,否则退出内循环;变量 i 加 1,当 i 小于 s 时重复外循环,否则退出外循环;判断 t 的高 k 位是否大于 n,若是则将 r2=r2-n,否则将 t 的低 k 位赋给 r2;根据 r2 输出模乘结果。本发明能提高算法的运算速度和降低功耗。



CN 103226461 B

1.一种用于电路的蒙哥马利模乘方法，其特征在于，模长度为k，k＝sw，其中w为算法每次处理的字长大小，s为算法for循环所需的次数；输入参数包括a、b和模数n，r为一存储器，r的高k位为存放输入参数a和输出结果的r2，r的低k位为存放输入参数b的r1，t为多项式临时计算结果，r32、z1、c、x1、x2、n0'为中间变量，i和j为循环变量，其运算步骤为：

计算r2＝MonPro(a,b,n)＝a*b*2^{-k} mod n；

将t和i赋零值；

将中间变量r32赋值为r的第s个字，其中，r32表示存储器r的第32地址的数据；

令i为0开始外循环；

将r的第i个字与r32相乘，乘积结果与t的第i个字相加，结果的低k位赋给z1，高k位赋给c；

接着将z1与n0'相乘，乘积结果的低k位赋给z1，高k位赋给c，其中，n0'通过预先计算获取，n0'＝$-n0^{-1}\bmod(2^{32})$，n0是指模数n的低32位值，n从外部输入；

令x2＝z1；令x1等于r的第i个字；

令j为0开始内循环；

将x1与r的第i个字相乘后的结果和x2与n的第i个字相乘后的结果相加，相加结果再加上c之后再加上t的第i+j个字，最终结果的低k位赋给t的第i+j个字，高k位赋给c；并令t的第i+s个字等于c，然后循环变量j加1，当j小于s时重复内循环，当j大于或者等于s时退出内循环；

循环变量i加1，当i小于s时重复外循环，当i大于或者等于s时退出外循环；

判断t的高k位是否大于n，若是则将r2＝r2-n，否则将t的低k位赋给r2；

根据r2输出模乘结果。

2.一种蒙哥马利模乘电路，其特征在于，蒙哥马利模乘采用权利要求1所述的方法实现，所述电路包括依次连接的接口电路、SRAM存储器、蒙哥马利模乘控制器、临时寄存器和多项式乘加器；

接口电路与外部总线连接，将输入参数a、b写进SRAM存储器的r存储器中，并将计算完毕后的输出结果读出；

SRAM存储器包括r存储器和t存储器，用于存储输入参数a、输入参数b、中间处理数据以及最终计算结果；

蒙哥马利模乘控制器，用于产生地址和控制信号、读取SRAM存储器中的数据并放入相应寄存器中进行处理，其内设置有地址产生器、减法器、计数器和控制电路；

所述地址产生器用于产生状态跳转信号和访问SRAM存储器的地址信号；

减法器用于完成r2＝r2-n的减法操作；

计数器用于计算外部循环次数i和内部循环次数j；

控制电路控制循环的进入和退出以及在每个时钟内，根据地址产生器产生的地址从SRAM存储器中取出相应数据放到临时寄存器中，并将中间结果和多项式临时计算结果t写回到SRAM存储器中；

临时寄存器用于暂存SRAM存储器读写数据以及多项式乘加器的输入输出结果；

多项式乘加器，用于完成如下计算：将x1与r的第i个字相乘后的结果和x2与n的第i个字相乘后的结果相加，相加结果再加上c之后再加上t的第i+j个字，最终结果的低k位赋给t

的第i+j个字，高k位赋给c。

3.根据权利要求2所述的蒙哥马利模乘电路，其特征在于，所述临时寄存器中包括七个寄存器：ti寄存器、x1寄存器、y1寄存器、x2寄存器、y2寄存器、c寄存器和z1寄存器。

4.根据权利要求2所述的蒙哥马利模乘电路，其特征在于，所述SRAM存储器为双端口SRAM存储器，其在时钟的下降沿完成读操作，在时钟的上升沿完成写操作。

5.根据权利要求2所述的蒙哥马利模乘电路，其特征在于，所述SRAM存储器为双端口寄存器文件，其在时钟的下降沿完成读操作，在时钟的上升沿完成写操作。

6.根据权利要求2所述的蒙哥马利模乘电路，其特征在于，多项式乘加器对多项式采用基4的Booth编码，其产生的部分和利用4:2压缩器逐级压缩。

一种用于电路的蒙哥马利模乘方法及其电路

技术领域

[0001] 本发明涉及公钥加密领域，更具体地，涉及一种用于电路的蒙哥马利模乘方法及其电路。

背景技术

[0002] 公钥加密利用的是非对称密码，使用两个独立担忧存在着某种数学联系的密钥：公钥和私钥。通信的各方保密各自的私钥，公开其公钥，发送者使用接收者的动摇加密，接收者使用只有自己制动的私钥解密。公钥加密还可以用于解决数字签名的问题。

[0003] RSA是一种公钥加密算法，既可以作为数据加解密也可以用来数字签名和验证，这使得该算法得到了广泛的应用，例如网络信息安全、智能卡、安全芯片以及手机移动通信等。RSA算法的安全依赖于大数分解的难易性，随着计算机的飞速发展，512位密钥长度的RSA加密算法安全性受到威胁，因此高安全的应用中必须将密钥长度增加到1024位甚至2048位。密钥长度的增加使硬件电路设计变得复杂，需要占用更多的硬件资源开销，面积功耗都会大幅上升，速度也会下降。RSA加密最关键的计算步骤就是模乘，根据运算公式C＝M^EmodN，需要不断重复模乘操作以得到运算结果，模乘算法的优劣决定了整个加密运算时间的长短。

[0004] 现有设计大部分采用蒙哥马利模乘算法优化加解密时间，然而一味的提高模乘速度会使得功耗和面积增加，这对面积和功耗受限制的芯片设计带来巨大的挑战。而且传统的模乘算法是在软件中实现，这将取决于处理器的速度，每个操作都需要取指令、译码以及执行指令，大大降低了算法执行速度。随着集成电路的发展，模乘算法大多采用硬件实现，其最关键的电路便是多项式乘加器。但由于乘加器功耗大，现有做法普遍是用多个时钟实现，非常速度慢。

发明内容

[0005] 本发明旨在至少在一定程度上解决上述技术问题。

[0006] 本发明的首要目的是提供一种高速低功耗的用于电路的蒙哥马利模乘方法。

[0007] 本发明的进一步目的是提供一种高速低功耗的蒙哥马利模乘电路。

[0008] 本发明第一个目的技术方案为：

[0009] 一种用于电路的蒙哥马利模乘方法，模长度为k，k＝sw，其中w为算法每次处理的字长大小，s为算法for循环所需的次数；输入参数包括a、b和模数n，r为一存储器，r的高k位为存放输入参数a和输出结果的r2，r的低k位为存放输入参数b，t为多项式临时计算结果，r32、z1、c、x1、x2、n0’为中间变量，i和j为循环变量，其运算步骤为：

[0010] 计算r2＝MonPro(a,b,n)＝a*b*2^{-k} mod n；

[0011] 将t和i赋零值；

[0012] 将中间变量r32赋值为r的第s个字，其中，r32表示存储器r的第32地址的数据；

[0013] 令i为0开始外循环；

[0014] 将r的第i个字与r32相乘，乘积结果与t的第i个字相加，结果的低k位赋给z1，高k位赋给c；

[0015] 接着将z1与n0'相乘，乘积结果的低k位赋给z1，高k位赋给c，其中，n0`通过预先计算获取，$n0' = -n0^{-1} \bmod (2^{32})$，n0是指模数n的低32位值，n从外部输入；

[0016] 令x2＝z1；令x1等于r的第i个字；

[0017] 令j为0开始内循环；

[0018] 将x1与r的第i个字相乘后的结果和x2与n的第i个字相乘后的结果相加，相加结果再加上c之后再加上t的第i+j个字，最终结果的低k位赋给t的第i+j个字，高k位赋给c；并令t的第i+s个字等于c，然后循环变量j加1，当j小于s时重复内循环，当j大于或者等于s时退出内循环；

[0019] 循环变量i加1，当i小于s时重复外循环，当i大于或者等于s时退出外循环；

[0020] 判断t的高k位是否大于n，若是则将r2＝r2-n，否则将t的低k位赋给r2；

[0021] 根据r2输出模乘结果。

[0022] 在本方法中，多项式乘加器公式的运算只需要一个时钟周期，运算速度得到大大的提升。而且本发明的蒙哥马利模乘方法采用一体化设计集成相关操作，使得多项式乘加器公式只需要做两次加法以及两次乘法，时间复杂度为$O(s^2)$，大大简化了运算的复杂度，进一步提升运算的速度，降低系统功耗。

[0023] 本发明第二个目的技术方案为：

[0024] 一种蒙哥马利模乘电路，所述蒙哥马利模乘采用权利要求1所述的方法实现，所述电路包括依次连接的接口电路、SRAM存储器、蒙哥马利模乘控制器、临时寄存器和多项式乘加器；

[0025] 接口电路与外部总线连接，将输入参数a、b写进SRAM存储器的r中，并将计算完毕后的输出结果读出；

[0026] SRAM存储器包括r存储器和t存储器，用于存储输入参数a、输入参数b、中间处理数据以及最终计算结果；

[0027] 蒙哥马利模乘控制器用于产生地址和控制信号，读取SRAM存储器中的数据并放入相应寄存器中进行处理，其内设置有地址产生器、减法器、计数器和控制电路；

[0028] 所述地址产生器用于产生状态跳转信号和访问SRAM存储器的地址信号；

[0029] 减法器用于完成r2＝r2-n的减法操作；

[0030] 计数器用于计算外部循环次数i和内部循环次数j；

[0031] 控制电路控制循环的进入和退出以及在每个时钟内，根据地址产生器产生的地址从SRAM存储器中取出相应数据放到临时寄存器中，并将中间结果和多项式临时计算结果t写回到SRAM存储器中；

[0032] 临时寄存器用于暂存SRAM存储器读写数据以及多项式乘加器的输入输出结果；

[0033] 多项式乘加器，用于完成如下计算：将x1与r的第i个字相乘后的结果和x2与n的第i个字相乘后的结果相加，相加结果再加上c之后再加上t的第i+j个字，最终结果的低k位赋给t的第i+j个字，高k位赋给c。

[0034] 在本发明的蒙哥马利模乘电路中，多项式乘加器采用一体化设计，即用一个组合逻辑模块将两次加法、两次乘法运算集成在一起，大大降低了时间复杂度，提升了算法的运

算速度。

[0035] 在一种优选方案中，所述临时寄存器中包括七个寄存器：ti寄存器、x1寄存器、y1寄存器、x2寄存器、y2寄存器、c寄存器和z1寄存器。这七个寄存器分别用于暂存SRAM存储器读写数据以及多项式乘加器的输入输出结果。

[0036] 所述SRAM存储器为双端口SRAM存储器，其在时钟的下降沿完成读操作，在时钟的上升沿完成写操作。

[0037] 在一种优选方案中，所述SRAM存储器为双端口寄存器文件，其在时钟的下降沿完成读操作，在时钟的上升沿完成写操作。双端口寄存器功耗低，访问速度快，其在时钟的下降沿完成读操作，上升沿完成写操作，使其能够在一个时钟内完成读写操作。

[0038] 在一种优选方案中，多项式乘加器对多项式采用基4的Booth编码，其产生的部分和利用4:2压缩器逐级压缩。

[0039] 与现有技术相比，本发明技术方案的有益效果是：

[0040] 本发明的多项式乘加器采用一体化设计，将两次加法、两次乘法运算集成在一起，使多项式乘加器能够在一个时钟周期内完成多项式的一次计算，降低了算法的时间复杂度，从而提升了算法的运算速度，降低整个算法运行的功耗。

附图说明

[0041] 图1为本发明一种蒙哥马利模乘电路硬件实现架构图。

[0042] 图2为本发明多项式乘加器采用华莱士树压缩结构的示意图。

[0043] 图3为本发明关键路径图。

[0044] 图4为本发明中涉及到的门控时钟电路图。

[0045] 图5为本发明的模乘算法时序图。

具体实施方式

[0046] 附图仅用于示例性说明，不能理解为对本专利的限制；

[0047] 为了更好说明本实施例，附图某些部件会有省略、放大或缩小，并不代表实际产品的尺寸；

[0048] 对于本领域技术人员来说，附图中某些公知结构及其说明可能省略是可以理解的。

[0049] 下面结合附图和实施例对本发明的技术方案做进一步的说明。

[0050] 实施例1

[0051] 一种用于电路的蒙哥马利模乘方法，模长度为k，$k=sw$，其中w为算法每次处理的字长大小，s为算法for循环所需的次数；输入参数包括a、b和模数n、n0`，其中，n0`通过预先计算获取，$n0'=-n0^{-1}\bmod(2^{32})$，n0是指模数n的低32位值；r为一存储器，r的高k位为存放输入参数a和输出结果的r2，也就是说输出结果保存在r2中，r的低k位为存放输入参数b，t为多项式临时计算结果，r32、z1、c、x1、x2、n0'为中间变量，其中，r32表示存储器r的第32地址的数据，i和j为循环变量，如图1所示，其通过如下运算步骤计算输出结果：

S1. r2 = MonPro(a, b, n) = $a*b*2^{-k} \bmod n$

S 2. t=0， i=0；

S 3. r32=r[s]；

S 4. for (i==0; i++; i < s)

S 5. (c, z1) = t[i] + r[i]*r32；

S 6. (c, z1) = z1*n0'；

[0052] S 7. x2 = z1；

S 8. x1 = r[i]；

S 9. for (j==0; j++; j < s)

S 10. (c, t[i+j]) = t[i+j] + x1 * r[i] +x2 * n[i] + c；

S11. t[i+s] = c；

S12. if t2 > n, then (r2 = t2 -n)

S13. else (r2 = t1)。

[0053] 输出结果存储在r2中，通过r2即可得到输出结果。

[0054] 本实施例采用一体化设计集成上述操作，S10中的多项式t[i+j]+x1*r[i]+x2*n[i]+c即为多项式乘加器公式，其只需要做两次加法以及两次乘法。在本实施例中，步骤S10和S11的计算只需要一个时钟周期，大大减少了运算的时钟周期，从而提高了蒙哥马利模乘的运算速度。很明显，本实施例中算法的时间复杂度为$O(s^2)$，假设是1024位模乘运算，s＝w＝32，于是一次模乘就只需要大概1024个时钟周期，从而能够提升算法的运算速度和降低系统的功耗。

[0055] 开始工作之前，对t寄存器和i计数器清零，并从s地址的r存储器中读出内容放到r32寄存器中，然后开始第一重循环。S5、S6、S7、S8步骤为初始化操作步骤，是为了S10的高速计算做铺垫的，由于S9的循环i是不变化的，故可以在S7和S8中先计算出x1和x2，然后进入S9的s次循环，每次循环x1和x2都不需要重新载入。在第10步骤中，在时钟的上升沿将地址为i+j的t寄存器数据、x1寄存器、地址为i的r寄存器、x2寄存器、地址为i的n以及c寄存器放到多项式乘加器中，在下一个时钟上升沿将乘加器的内容保存回t存储器中，以上操作只需一个时钟便可完成复杂计算，这是速度提升的关键技术所在。

[0056] 实施例2

[0057] 本实施例以1024位模乘运算为例在硬件上实现实施例1所述的方法，以本实施例算法为基础实现的其他长度(如512位、2048位等)模乘运算及其硬件实现应当属于本发明的保护范围。

[0058] 如图2所示，一种蒙哥马利模乘电路，其中蒙哥马利模乘采用上述蒙哥马利模乘方法实现，该电路包括依次连接的接口电路、SRAM存储器、蒙哥马利模乘控制器、临时寄存器和多项式乘加器；

[0059] 接口电路与外部总线连接，将输入参数a、b写进SRAM存储器的r中，并将计算完毕

后的输出结果读出；

[0060] SRAM存储器包括r存储器和t存储器，用于存储输入参数a、输入参数b、中间处理数据以及最终计算结果。以32位字长为单位，由于是1024位模乘运算，一共需要2048位r和t，因此需要64组r[i]和t[i](i为0到63的整数)。本实施例的SRAM存储器采用双端口存储器，在低功耗的应用场景，SRAM存储器也可以采用双端口寄存器文件，功耗低，访问速度快，可在一个时钟内完成读写操作，在时钟的下降沿完成读操作，上升沿完成写操作。为了加快读写操作，r存储器和t存储器可以同时执行读操作，写操作可以不在同一个时钟内执行。此外，在SRAM存储器不操作器件，可以将其使能信号关闭，降低功耗。采用SRAM存储器的好处在于，不仅可以获取高速读写能力，还可以节省面积，同样的存储内容，用SRAM存储器要比普通的寄存器要省一倍以上面积。

[0061] 蒙哥马利模乘控制器，用于产生地址和控制信号，读取SRAM存储器中的数据并放入相应寄存器中进行处理，其内设置有地址产生器、减法器、计数器和控制电路；

[0062] 所述地址产生器用于产生状态跳转信号和访问SRAM存储器的地址信号；

[0063] 减法器用于完成r2＝r2-n的减法操作，即实现实施例1中S12步骤中的减法操作。

[0064] 计数器用于计算外部循环次数i和内部循环次数j；

[0065] 控制电路控制循环的进入和退出以及在每个时钟内，根据地址产生器产生的地址从SRAM存储器中取出相应数据放到临时寄存器中，并将中间结果和多项式临时计算结果t写回到SRAM存储器中；

[0066] 临时寄存器均为32位，包括七个寄存器，分别为ti寄存器、x1寄存器、y1寄存器、x2寄存器、y2寄存器、c寄存器和z1寄存器，用于暂存SRAM存储器读写数据以及多项式乘加器的输入输出结果。临时寄存器的增加使得从SRAM存钱到多项式乘加器的数据通路最为便捷。本发明采用了多个临时寄存器保存中间数据，寄存器功耗大，为了进一步降低功耗，其可以采用图4所示的门控时钟电路来降低寄存器的动态功耗，只有触发器需要干活时才给触发器时钟，否则不给时钟；图4的门控时钟电路包括第一寄存器D1、第二寄存器D2以及与门A1；该结构可以由DesignCompiler综合工具自动生成，在信号高电平期间锁存器是保持的，在信号低电平期间锁存器是透明的，此时数据可以传进来。

[0067] 多项式乘加器，用于完成“t+x1*y1+x2*y2+c”计算，即将x1与r的第i个字相乘后的结果和x2与n的第i个字相乘后的结果相加，相加结果再加上c之后再加上t的第i+j个字，最终结果的低k位赋给t的第i+j个字，高k位赋给c。此多项式乘加器采用一体化设计，即用一个组合逻辑模块将两次加法、两次乘法运算做到一起。对于104为的模乘运算来说，多项式采用基4的booth编码原理，共产生36个部分和，为了最大限度提高性能，本实施例采用4:2压缩器逐级压缩部分和。如图2所示，第一级压缩需要9组压缩器，第二级需要4组压缩器，第三级需要2组压缩器，第四级需要1级压缩器，第五级需要1级压缩器，第六级采用全加器将剩余的两组累加起来并得到最后结果。多项式乘加器能在一个时钟周期计算出多项式t+x1*y1+x2*y2+c的结果，并保存在SRAM存储器中，算法第9步中的循环，每轮循环只需要一个时钟，其关键技术在于，每个时钟的下降沿将SRAM的数据读出到t、x1、y1、x2、y2以及c临时寄存器中，在紧接着的上升沿将上一次多项式计算结果写回SRAM中，其关键路径涉及到SRAM读写时间、时钟延时、组合逻辑延时以及多项式乘加器延时，其中延时最大的是多项式乘加器，图3所示是本发明关键路径图，总延时T由以下几部分组成，D3触发器延时T_{D3}、组合

逻辑路径延时T_{logic}、多项式乘加器延时T_p以及SRAM存储器延时T_r，用公式表示如下：

[0068] $T = T_{D3} + T_{logic} + T_p + T_r$

[0069] 其中T_p的延时是最大的，涉及到两个32位并行乘法器的延时。本发明通过优化多项式乘加器达到较高速度；模乘电路所需时间数量级为$O(s^2)$，大大节省操作时间。如图5所示是本发明关键操作时序图，clk为系统时钟，r0，r1和r2是r存储器的开始三个存储数据，rd_address是r存储器的读地址信号，sram_out为r存储器读出的数据，wr_address是r存储器的写地址，sram_in是r存储器的写数据。从图中可以看到，经过三个时钟上升沿后，每个时钟都有SRAM写数据操作，即用一个时钟实现了第9步和第10步的算法操作。

[0070] 相同或相似的标号对应相同或相似的部件；

[0071] 附图中描述位置关系的用于仅用于示例性说明，不能理解为对本专利的限制；

[0072] 显然，本发明的上述实施例仅仅是为清楚地说明本发明所作的举例，而并非是对本发明的实施方式的限定。对于所属领域的普通技术人员来说，在上述说明的基础上还可以做出其它不同形式的变化或变动。这里无需也无法对所有的实施方式予以穷举。凡在本发明的精神和原则之内所作的任何修改、等同替换和改进等，均应包含在本发明权利要求的保护范围之内。

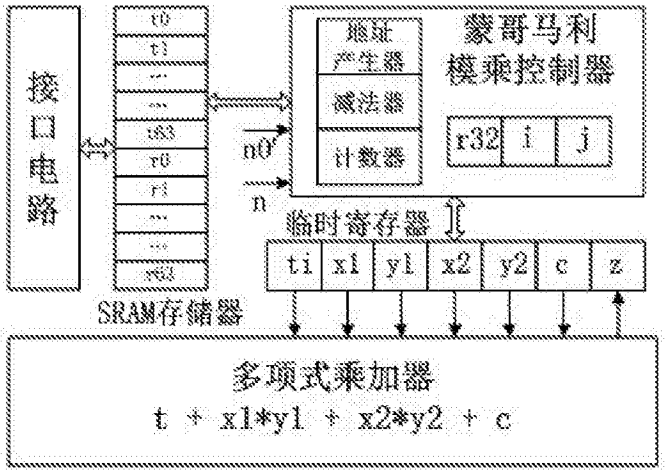

接口电路
t0
t1
...
t63
r0
r1
...
r63
SRAM存储器
m0
n
地址产生器
减法器
计数器
蒙哥马利模乘控制器
r32
i
j
临时寄存器
ti
x1
y1
x2
y2
c
z
多项式乘加器
t + x1*y1 + x2*y2 + c

图1

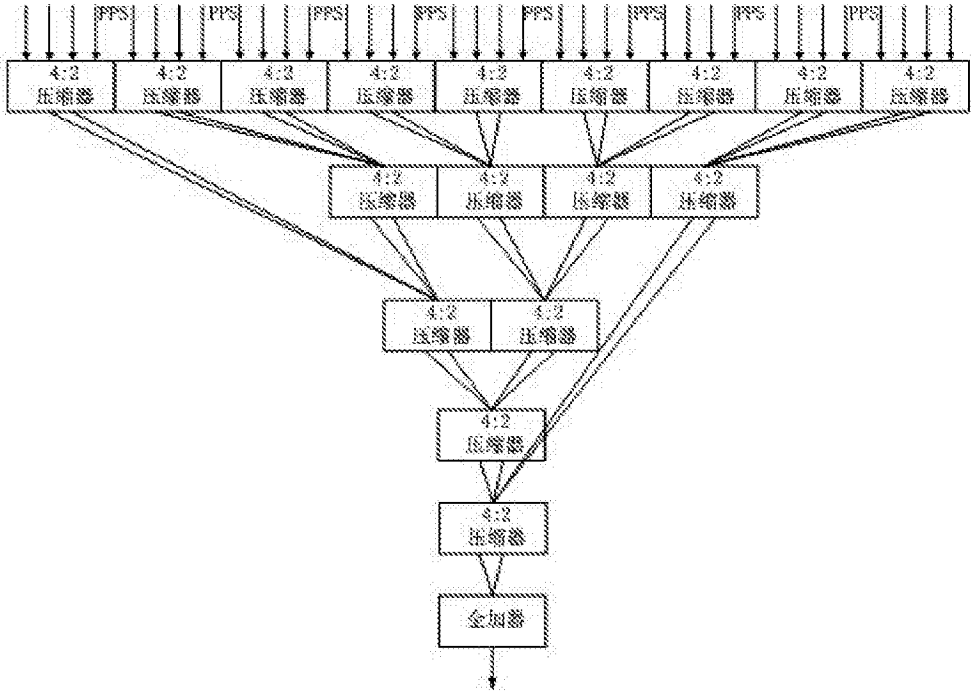

PPS
4:2 压缩器
全加器

图2

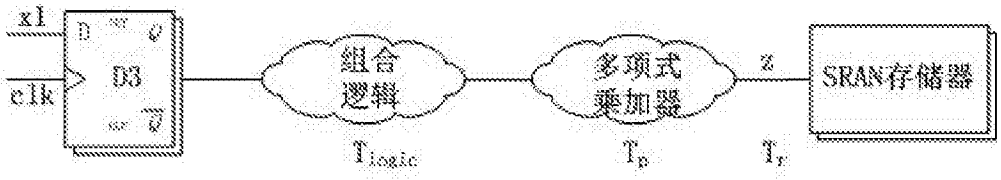

x1
clk
D3
组合逻辑
多项式乘加器
z
SRAN存储器
T_{logic}
T_p
T_r

图3

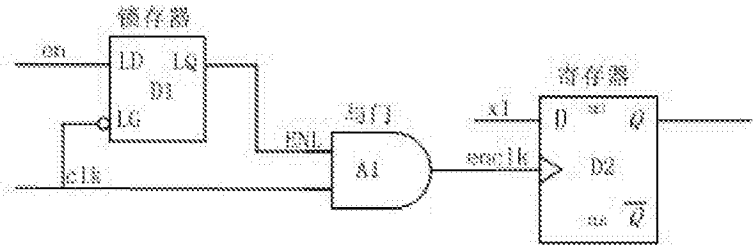


图4

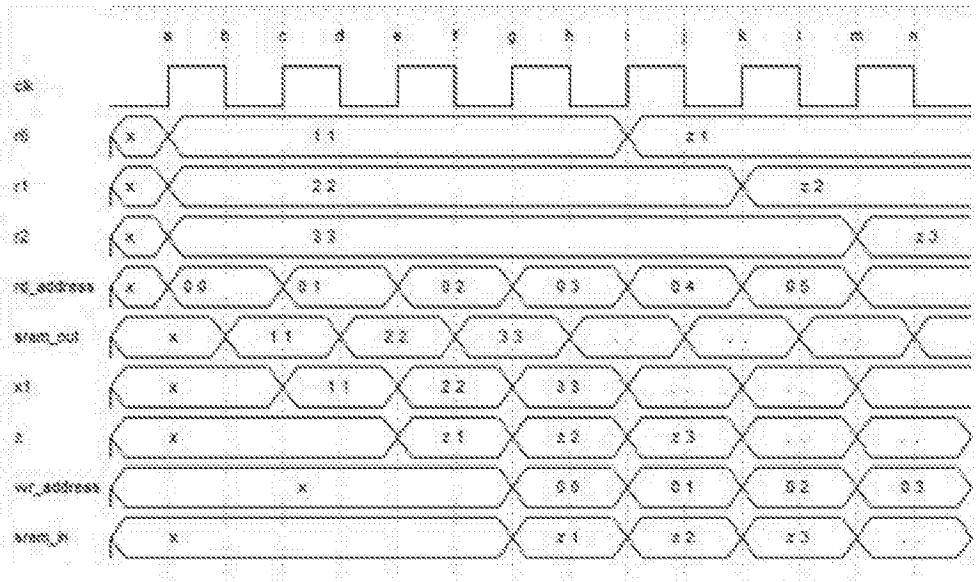


图5