字级模型检查 - 避免奔腾 FDTV 错误 "

E.M. Clnrke M.Khairat X. Zhao[†‡]

*卡内基梅隆大学计算机科学学院，宾夕法尼亚州匹兹堡 16213

英特尔开发L-b-.，英特尔公司，6300 NE Elam Young Pkwy, Hillsboro, OR 87124

摘要

奔腾处理器中的除法错误已广为人知，它凸显了对算术电路进行形式验证的重要性。基于双年决策图（BDDS）的符号模型检查技术在验证控制逻辑方面取得了成功。然而，由于缺乏将布尔矢量转换为整数的函数的适当表示方法，这种技术无法用于验证算术运算。

我们开发了一种新的电路验证技术。这项新技术被称为 *"字模型检查"（word It "ol model checking）*，已成功用于验证基于奔腾处理器 SRT 算法的二维和平方根计算电路。该技术可以同时处理电路中的控制逻辑和数据路径。状态变量的总数超过 600 个（比以前其他符号模型检查器处理的任何电路都要 大得多）。

1 Introduction

证明算术运算的**正确性**一直是一个重要问题。最近，奔腾处理器中的除法错误备受关注，凸显了这一问题的重要性。为了验证这种情况，有必要表示和处理将布尔矢量映射到整数值的函数a。我们使用混合判定图（HDDS）[5] 来表示算术电路验证中出现的整数函数。对于与数据位相对应的**状态变量**，我们的**表示方法**类似于二进制矩图（BMD），而对于与控制信号相对应的**状态变量**，我们的**表示方法**类似于多三态矩图（MTBDD）。通过使用这种表示法，我们能够处理既有控制逻辑又有宽数据路径的电路。

我们对将布尔向量映射到整数的函数的表示，使我们能够扩展 tempnrof 主题。

本研究部分由美国国家科学基金会（National Science Foundation）赞助，资助编号为 CC£t-8722684，由半导体公司（Semiconductor Reoeucch Corporation）根据 82-DJ-28t 合同赞助，并由美国阿肯色州奥姆努特科学ymcmsCener 莱特实验室赞助。CC£t-8722684, Semiconductor Reoeucch Corporation under contract 82-DJ-28t, and by the Wright Laboratory, AomnnuticslSymcmsCener, Ak MxceMcenñCommand, USAF
在高级研究计划局（ARPA）的支持下
FSD6l6-BE-1-1550。

DAC 96 - 06/96 Las Vegas, NV, USA

在传统的模型检查系统中，规范是用命题时态逻辑来表达的，电路设计和协议是用状态转换系统来建模的。在传统的模型检查**系统**中，规范是用命题时态逻辑来**表达**的，电路设计和协议则被建模为状态转换系统。使用一种高效的搜索程序来自动确定这些系统是否满足这些规范。这种方法的主要缺点是，如果被验证的系统有许多组件可以进行平滑转换，就会出现状态爆炸。由于 BDDs 的出现，模型检查技术可验证的转换系统的规模急剧扩大[2]。尽管这些apm6oiic模型检查技术在验证控制逻辑方面取得了成功，但它们不能直接用于验证算术运算。
circuits.

在字级模型检查系统中，电路节点的命题用 BDD 表示，计算方法与原始符号模型检查**系统**完全相同。字是命题数组，每个字对应一个比特。表达式由应用于字的算术运算组成。利用算术运算的算法，可以计算**词**和表达式的混合决策**图**。原子公式可以是表达式之间的关系，它们的 BDD **表示**可以通过处理算术关系的算法来计算。在生成原子公式的 BDD 表示之后，静态公式和时态公式的 BDDS 将以与普通模型检查相同的方式进行计算。特别是，两种情况下的定点计算完全相同。

通过使用词级模型检查**系统，**我们可以得出
我们成功地验证了基于**奔腾**处理器使用的 SRT 算法的除法和水根运算电路。我们能够同时处理控制逻辑和数据路径。控制逻辑的**有限**状态机**中**的所有状态都已得到验证。此外，我们还证明了保证数据值正确性和防止溢出的不变**属性。**状态机变量总数超过 600 个（比 SMV 以前检查过的 任何电路都要多得多）。

2　词级 C'I'L

基于二进制判定图（BDD）的符号模型检查技术在验证控制逻辑方面取得了成功[2]。然而，由于缺乏将布尔矢量转换为整数的函数的适当表示方法，这种技术无法满足验证电路的需要。我们已经尝试了前几节介绍的不同表示法。幸运的是，无论是 MTBDD 还是 BDD 数组表示法，在验证电路时都存在一些基本问题。

算术电路的符号化。对于这类应用中出现的函数，可能的值a 的数量是比特数的指数。因此，MTBDDs 的大小也是指数级的。另 一 方 面 ，BDD 阵列上的算术运算非常昂贵。特别是，由于组合乘法器中间位的 BDD aize 是其操作数长度的指数级， 因此 BDD 数组表示法对乘法运算来说也是指数级的。布莱恩特和陈[1]已经证明，BMDa 为某些具有指数级 siae MTBDD8 的函数提供了一种契约表示法。他们利用这种表示法验证了一些算术电路的数据路径。如果电路的 BMD 与规格完全相同，他们就能断定电路是正确的。然而，根据实现和控制逻辑的不同，可能会出现电路正确但 BMDs 不一致的情况。此外，由于他们的技术无法处理不等式，因此无法检查避免 Pentinm 所需的某些属性。

CEEOE.

我们使用混合决策图（BDDS）[5] 来表示算术循环验证中出现的整数函数。特别是，对于与数据位相对应的状态变量，我们使用反里德-穆勒变换，而对于与控制信号相对应的状态变量，我们使用同一变换。因此，对于数据变量，这种表示法就像 BMD，而对于控制变量，则像 MTBDD。通过使用这种表示法，我们能够处理同时具有控制逻辑和宽数据 pstha 的电路。由于这种表示法是混合决策图的特例，因此可以应用前面章节中提到的所有算法。

通过使用这种表示法，我们扩展了符号模型检查系统 SMV [7]，使其也能处理涉及数据字之间关系的属性。在最初的 SMV 系统中，原子公式只能包含状态变量。在扩展的系统中，我们允许原子公式也可以是表达式之间的等式或不等式。这些表达式用混合 BDD 表示。我们的逻辑如下：

e 原子命题：Up = (pi , ..., p'}

o 命题 a-l 公式：*Prop ::m Ap | Proph Prop Prop*

- 词：*词: ::--（道具，教授--，道具）*

"Expresaiona：
Exp ::m Constant jYord I next(jYord) | tsp O- jsp
S F 则 jsp，否则 ñzp，其中 O- 可以是 -}-、- 或 x。

- 原子公式：*AF ::= Up* I]+ I E}（Sep - tsp），其中 - 可以是 =、c 或 J。
一个给定等式的一个可能的下一个等式。因此，在表达式中使用下一个状态运算符时需要路径量化。

a 静态公式：*SP* ::= >f' I *SF* A *SP* I *NSF*

时间公式：*TF ::= SF* I >> ** >F \ ETF* I MX T> I (> | E}[TJ U *TF]*

一个模型是

"州a：*S* -- 2^-.

"过渡关系：A .C *S* 't *S*

"初始候选国*St* C *Al*

- 原子命题的估值映射 Y : *Ap* x *S* --- §0, *I}*

逻辑语义如下 .

o 命题式：*P : Prop* x *S -- j0, I j*

e Words：*W : 词* x *S -- N*

$$P(p_i, s) = V(p_i, s)$$
$$P(f_1 \wedge f_2) = P(f_1, s) \wedge P(f_2, s)$$
$$P(\neg f, s) = \neg P(f, s)$$
$$W((f_0, f_1, \ldots, f_n), s) = \sum_{i=0} P(f_i, s) 2^i$$

- 表达式： 状态 s 用于处理 Exprensiona 中出现的 next-8tate 运算符。

ñ(ci Q- es, r, s') = E(ei , *a, o') Q- E(ez, u, s') E('iI*
/ then ei else e" e , s ') =
if (r }= /) then A(ct, r, s') else *E(yy,* s, s')
A(tr, a, a') a W(ui, a)
(Oext(tu), S, 6')\ "(u/, J')

- 原子公式

```
4     @t M W ;,*) 1
* |= A(m o)'v
      V$. ( #)- (f(,*,*)*f(o,#x)
*= E(,-m)w
      3#.fl*,*)A(f(m,*z) f{az,z$))
```

在状态 s Mehen 中，公式 A（ei - es）为真（ ei es）在艺术继承者 stste8 中成立。 同样，当（ei - ez ）对某个后继站成立时，E（ei - e ）在后继站中为真。 ez）对某个后 继站 成立。

o SF 和 TF 的语义与 CTL 相同。

这种逻辑自然可以分为三层。顶层包含原子公式、静态公式和临时公式。第二层包含词和前置词。第三层包含原子命题和命题公式。顶层和底层的所有对象都是布尔函数，而第二层的对象则是将布尔向量映射到整数的函数。因此，在词级模型分析系统中，所有的原子命题、命题、原子公式、时态公式都用 BDDa 表示，而词和表达式则用混合决策图表示。

3 字级模型检查

模型检查是一种在状态转换图中找出特定 CTL 公式为真的状态 a 集的技术。有一种名为 EMC 的模型检查器可以利用高效的 grsph 遍历技术解决这个问题。如果模型表示为状态转换图，算法的复杂度与图的大小和公式的长度成 线性关系。该算法在实际应用中速度相当快[3, 4]。但是，当从一个有许多进程或部件的有限状态并发系统中提取状态-转换图时，模型的大小可能会爆炸。在 *符号模型检查*（symbolic *model checking* ztt'terna）[2]中，BDDS 被用来表示转换关系和状 态 集。模型检查过程通过对这些 BDDS 进行定点操作来完成。通过使用符号模型检查技术，可验证的过渡**系统**的规模大大增加。虽然这种技术在验证控制逻辑方面取得了成功，但却不能直接用于验证算术电路。这是因为涉及整数值字的表达式无 法 正确处理。

为了能够对上一节讨论的逻辑进行模型检查，我们需要在混合决策图上实现各种运算。我们考虑了标量乘法、两个函数的加法和乘法以及 if-then-else 运算。虽然ee算法的最坏情况复杂度可能是指数级的，但在实践中，这种算法运行得相当好。字级属性的模型检查要求计算满足 /i - /z 的赋值行为，其中 - 可以是 =、/、c、J、T 或 J 中的一个。我们开发了一种高效算法，可以计算满足算术关系的赋值集的 BDD。特别是，对于线性表达式，该算法具有线性复杂度[5]。

既然我们能够处理算术运算和算术关系，那么就有可能扩展符号模型检查算法，使其能够验证字级属性。在 BDDS 中，过渡关系的所有命题都是以与原始符号模型检查系统完全相同的方式生成的。混合决策分析单词（/o、/i、.....、/n）的语法表示可组合为

$$\prod_{i=1}^{n} (\text{if } /; \text{ tben } 2' \text{ else } 0)$$

通过上述操作即可计算。莫特表达式的混合判定符表示也可以通过类似的操作来计算。唯一的例外是通过变量替换进行的 "*noct* "操作。替换操作是将词的混合决策图中的所有当前状态变量替换为与之对应的下一个状态变量。获得代表使代数关系为真的变量赋值集的 BDD 的算法可以用来计算原子公式的 BDD。在生成原子公式的 BDD 表示之后，静态公式和时间公式的 BDD 计 算 方法与普通模型分析方法相同。尤其是两种情况下的 定点计算完全相同。

由于我们使用了与普通模型检查算法相同的算法来计算过渡关系。当过渡关系没有准确的表示时，字级模型检查算法就不能很好地工作。举个例子，让我们来看看一个多路复用器。假设和 p 是输入寄存器，z 是输出寄存器。假设转换关系可以表示如下

```
*d*¥ )'TrO )^(n^()'-*ñ
```

显然，由于乘法器中间位的 BDD 表示是指数级的，因此转换关系的 BDD 表示具有指数级的大小。这个问题有时可以通过连接分解转换关系来避免。设 £、j 和 â 分别是编码 z、p 和 z 当前值的状态变量。假设我们要验证一个形式为 /（s, p, z）的词级属性。可能会出现 next(z)；如果出现了，我们可以在字级别上用 s x p 将其替换，得到一个新公式。希望得到的公式与 z 无关，公式的 BDD 表示可以表示为 /'（*， p）。在这种情况下，我们可以将 *Tr'* 作为过渡关系来执行 £ xpoint 操作。即使 /' 取决于 z 的某些位，我们通常也可以通过删除给出不需要的位值的连接词来获得更简单的转换关系。

4 SRT 除法电路的验证

通过使用字级模型检查系统，我们成功验证了基于奔腾处理器使用的 SRT 算法的除法和平方根运算电路。我们能够同时处理控制逻辑和数据路径。我们研究的除法电路有 5 个状态：*空闲*、初始、*循环*、*最后*和返回。该电路可进行*除法*和*重除法*两种不同的运算。

闲置 *init -- loop*---- l'izt -- idle*

*剩余*操作时，步骤如下

idleinit --- loop ---- lost --- rem --- idle*

图 1 给出了电路在循环状态下的数据路径。所有字都有 70 位。不过，只有部分余数和除数倍数的前导位才用于计算下一个循环的商数字。

我们对电路的控制逻辑和数据路径都进行了验证。有限状态机的所有状态都已检查完毕。让 r 是部分余数，q 是商，d 是除数。我们检验了以下属性：

o 表达式 r + q d 总是等于左移红利，即 r + q- d = 2"-红利。

"计算不会溢出。保证
by $-\frac{8}{3}d \le r \le \frac{8}{3}d$.

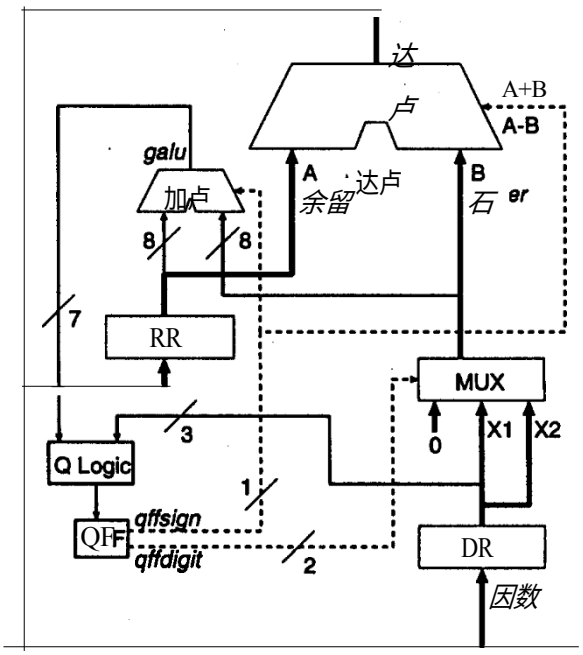

图 1：分割电路的数据路径

例如，我们已经证明，在初始状态下，再分子是红利，商为零。因此，r -}- q- *d* 的初始值等于红利。此外，上述不等式在初始状态也成立。

SPEC £0 (at at e - **init** -kr) **- 股息** t q - 0)

SPEc fis (at ate - init - ' (-8) - 'd '- 3 - r ' - 8 - d)

我们已经证明，在循环状态下，不等式总是成立的，而且 r -J- q - d 在 左移时是不变的。

SPEC âG (ate - loop -b)
fi I((-8)-d <- 3-r <- 8-d) U ate- la "t1)

SPec fiG ((ate - ioop z ((-8) -d '- 3-r '- 8-d))-' A(
(r+q*r)*4 - next (r+q*r)))

上述性质足以保证在循环状态下，r -} g *d* 总是等于左移后的红利。对于*末尾*和剩余状态，我们也证明了类似的**性质**。此外，我们还验证了计算阿卡雷根的电路。我们验证的这个电路的有根变量总数超过 600 个（比 SMV 以前验证过的 任何电路都要多）。

我们计划在更多电路上进行实验。我们计划在更多电路上进行实验。可能的应用包括浮点乘法器、浮点加法器等。我们的算术关系式求解算法对线性方程和不等式的求解效果非常好。尽管目前的算法也能处理一些非线性方程和不等式，但仍有可能扩展这一算法或找到能处理更多非线性方程和不等式的新算法。复杂的非线性方程和不等式。

这种技术还有一个问题。它只能用于保持数据精确值的电路。当四舍五入发生时，函数变得不那么规则，混合 BDD 表示的大小可能会减小。在这种情况下，四舍五入后得到的新值可以用一个不等式系统来描述，验证过程就简化为求解 8uch ayatems。在另一个研究项目中，我们建立了一个基于符号计算系统 Mathematica 的定理证明器。该定理验证器名为 *Analptica* [6]，在处理方程和不等式方面相当出色。我们相信，经过一些修改，Analytics 将有助于解决计算机运算中由于舍入而产生的不等式。

5 未来研究方向

我们使用字级符号模型检查来修复奔腾 FDIV 漏洞，并成功验证了修正后的电路。在本 文 中，我们介绍了对基于

参考资料

[1] R.E. Bryant 和 Y. A. Chen.用二进制矩图验证算术函数。In *Proceed- ings oJ the f2nd A OM f'IEEE Design Automation Oon-* Jcrence, pagee 535-541.IEEE 计算机学会出版社，1995 年 6 月。

[2] J.J. R. Burch, E. M. Clarke, K. L. McMillan, D. L. Dill, and L. J. Bwang.符号模型分析： 10^{20} state8 and beyond.*信息与计算》*，98（2）：142- 170，1992 年 6 月。

[3] E.M. Clarke and E. A. Emerson.分支时间时态逻辑的同步化骨架合成。In *Logic of Programs： Workshop, Yorktown Heights, N Y, Mop 1981, Volume* 131 of Lecture *Note- in C!omputer Scienm.*Springer-Verlag, 1981.

[4] E.E. M. Clarke, E. A. Emerson, and A. P. Sistla.Automatic verification of 8nite-state concurrent systems ueing temporal logic specifications.*ACFM* Tronroctions on *Pro- grooming* 7 "ng "ogr" *and Systems,* 8(2):244-263, 1986.

[5] E.M. Clarke, M. Fujits, and X. Zhao.Bybrid Deci- sion Diagrams - overcoming the limitations of If TBDDs and BM De.In I roceedinpr *oJ the 1995 Proceedings oJ the IEEE International Oon,ference on Computer Aided De- ign,* **psges** 159-163.IEEE 计算机学会**出版社，**1995 年 11 月。

(6]E. M. Clarke 和 X. Zhao.Analytica：A theorem prover for Mathematics.*The Journal off Mathtmatica,* 3(1), 1993.

[7] K.L. McMillan.Sprnboiic *Model* t7ficckinp.

[8] G. S. Taylor.除法和水根的兼容硬件。In *Proceedings oJ the Fish IEEE Sympo-* riurn on *Computer Arithmetic,* 1993.