

(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国际局



PCT



(43) 国际公布日
2007 年 2 月 15 日 (15.02.2007)

(10) 国际公布号
WO 2007/016871 A1

(51) 国际专利分类号:
H03M 13/41 (2006.01)

(21) 国际申请号: PCT/CN2006/001989

(22) 国际申请日: 2006 年 8 月 7 日 (07.08.2006)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
200510036378.8
2005 年 8 月 8 日 (08.08.2005) CN

(71) 申请人 (对除美国外的所有指定国): 北京大学深圳研究生院 (PEKING UNIVERSITY SHENZHEN GRADUATE SCHOOL) [CN/CN]; 中国广东省深圳市南山区西丽深圳大学城北大校区, Guangdong

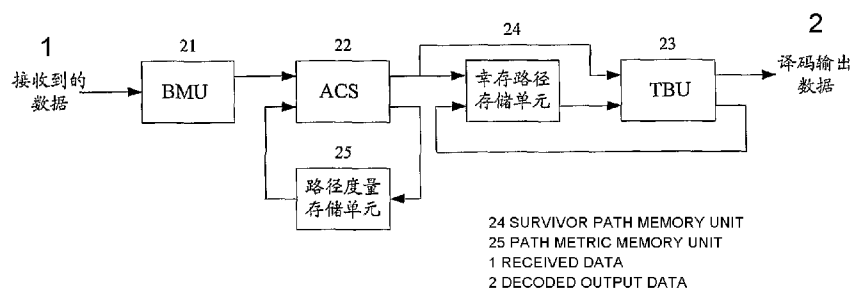
518055 (CN)。华为技术有限公司 (HUAWEI TECHNOLOGIES CO., LTD.) [CN/CN]; 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。

(72) 发明人; 及
(75) 发明人/申请人 (仅对美国): 王一 (WANG, Yi) [CN/CN]; 中国广东省深圳市南山区西丽深圳大学城北大校区, Guangdong 518055 (CN)。王新安 (WANG, Xin'an) [CN/CN]; 中国广东省深圳市南山区西丽深圳大学城北大校区, Guangdong 518055 (CN)。陈惠明 (CHEN, Huiming) [CN/CN]; 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。张国新 (ZHANG, Guoxin) [CN/CN]; 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。肖高发 (XIAO, Gaofa) [CN/CN]; 中国广东省深圳市南山区西丽深圳大学城北大校区, Guangdong 518055 (CN)。洪波 (HONG, Bo) [CN/CN]; 中国广东省深圳市南山区西丽深圳大学城

[见续页]

(54) Title: VITERBI DECODER AND METHOD FOR DECIDING DATA BIT WIDTH OF THE ADD-COMPARE-SELECT UNIT THEREOF

(54) 发明名称: 一种维特比译码器以及决定其中加比选单元数据位宽的方法



(57) Abstract: The present invention relates to a Viterbi decoder, and a method for deciding minimal data bit width of add-compare-select unit, so as to reduce realization area of hardware of the Viterbi decoder and delay time of key path of the add-compare-select unit; it is effectively resolve overflow of PM value, and is generally suitable for parallel /serial/mixed type construction, it includes: the first step, span S_{PM} of path metric value (PM) of each column of the grid chart is calculated by the first expression, i.e. $S_{PM} \leq n_0 \cdot (m-1) \cdot (2^{\text{width}} - 1)$; the second step, value of i is determined by the second expression, i.e. $2^{i-2} \leq S_{PM} < 2^{i-1}$; the third step, it checks whether value of i meets with the third expression, i.e. $(2^{i-1} - 1) + S_{PM} \cdot (2^{\text{width}} - 1) < 2^i$, if value of i meets the third expression, the minimal data bit width of add-compare-select unit is i bits; if value of i doesn't meet with the third expression, the minimal data bit width of add-compare-select unit is $(i+1)$ bits.

[见续页]

WO 2007/016871 A1



北大校区, Guangdong 518055 (CN)。赵腾飞(ZHAO, Tengfei) [CN/CN]; 中国广东省深圳市南山区西丽深圳大学城北大校区, Guangdong 518055 (CN)。蓝文广(LAN, Wenguang) [CN/CN]; 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。

(74) 代理人: 北京集佳知识产权代理有限公司(UNITALEN ATTORNEYS AT LAW); 中国北京市朝阳区建国门外大街22号赛特广场7层, Beijing 100004 (CN)。

(81) 指定国 (除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LV, LY, MA,

MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。

(84) 指定国 (除另有指明, 要求每一种可提供的地区保护): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), 欧洲 (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG)。

本国际公布:

— 包括国际检索报告。

所引用双字母代码及其它缩写符号, 请参考刊登在每期PCT公报期刊起始的“代码及缩写符号简要说明”。

(57) 摘要:

本发明涉及一种维特比译码器, 以及一种确定最小加比选单元的数据位宽的方法, 从而减小维特比译码器硬件实现面积以及加比选单元关键路径的延迟时间; 可以有效地解决 PM 值溢出问题, 并且能普遍适用于 Viterbi 的并行/串行/混合型结构, 包括: 第一步, 根据公式一计算网格图中的每一列的路径度量值 (PM) 的跨度 S_{PM} , $S_{PM} \leq n_0 \cdot (m-1) \cdot (2^{width} - 1)$; 第二步, 再根据公式二确定 i 值: $2^{i-2} \leq S_{PM} < 2^{i-1}$, 第三步, 验证所述 i 值是否满足公式三, $(2^{i-1} - 1) + S_{PM} + \left\lfloor \frac{n_0}{2} \right\rfloor \cdot (2^{width} - 1) < 2^i$, 如果所述 i 值满足所述公式三, 则所述加比选单元的最小数据位宽为 i 比特; 如果所述 i 值不满足所述公式三, 则所述加比选单元的最小数据位宽为 $(i+1)$ 比特。

一种维特比译码器以及决定其中加比选单元数据位宽的方法

技术领域

本发明涉及一种基于专用集成电路 (ASIC) 的维特比 (Viterbi) 译码器, 尤其涉及一种决定 Viterbi 译码器中的加比选 (ACS 相加-比较-选择)

5 单元电路的数据位宽的方法。

背景技术

卷积码是一种常用的差错控制编码。卷积码 (n_0, k_0, m) 表示该卷积码编码器将 k_0 比特信息段编成 n_0 比特的码组, 而且所编的 n_0 长比特的码组不仅同当前 k_0 比特信息段有关联, 还同前面的 $(m-1)$ 个 ($m>1$, 整数, 我们称其为约束长度) 信息段有关联。卷积码用生成序列表示输入与输出之间的关系, 生成序列表示为 $g_{i,j} = (g_{i,j}^1 \ g_{i,j}^2 \ \dots \ g_{i,j}^L \ g_{i,j}^m)$, 其中 $i=1, 2, \dots, k_0$; $j=1, 2, \dots, n_0$; $L=1, 2, \dots, m$ 。 $g_{i,j}^L$ 表示了各个寄存器的输入输出端 (第 L 组的第 i 个节点) 到第 j 个模 2 加法器输入端的连接线的情况。如果有连接线则 $g_{i,j}^L=1$, 如果无连接线则 $g_{i,j}^L=0$ 。可以看出, 卷积码有很强的关

10 联性, 能够很好的进行纠错。图 1 给出一种典型的卷积编码器—— $(2, 1, 7)$ 卷积编码器, 即该卷积编码器有 2 比特输出, 1 比特输入, 约束长度为 7, 寄存器阶数等于 $(m-1)$, 即寄存器阶数为 6; 该卷积码的生成序列为 $g_{1,1} = (1011011)_2$ 和 $g_{1,2} = (1111001)_2$, 为了表述简便, 用八进制表示为 $(133)_8$ 和 $(171)_8$ 。

20 卷积码的一种主要的纠错译码方法就是 Viterbi 算法。Viterbi 算法是一种基于最大后验概率的卷积译码算法, 具有较强的克服突发错误的能力, 在数据通信、数据记录与数字信号处理领域都得到广泛的应用。Viterbi 的译码过程即是通过接收的过程找出与卷积码输入数据流最相似的路径。遍历整个状态变化的网格图, 并计算每条路径产生的码距, 当 Viterbi 译码器

25 输入完最后一个数据的时候, 就可以得到最小的码距值, 而到达这个状态的路径就是要寻找的路径, 根据这条径和相关的信息可以得出译码的输出。

图 2 是一个通常的 Viterbi 译码器的结构框图。分支度量单元 (BMU)

-2-

21 接收数字信号, 并计算分支度量值作为概率的信息, 这里的分支度量值取无符号数。加比选单元 (ACS) 22 从 BMU21 中读取分支度量值并利用该分支度量值更新与网格中每个状态对应的原路径度量 (PM) 值。ACS22 将已更新的 PM 值互相比较并输出被选中的 PM 值及相应的选择位。在路径度量存储单元 25 中, 由 ACS22 选出的 PM 值在后继的步骤中被送回 ACS22 单元。幸存路径存储器 24 存储从 ACS22 输出的选择位。回溯单元 TBU23 利用存储在幸存路径存储器 24 中的选择位实现回溯操作并输出译码序列。

Viterbi 的各种应用中, 数据输出速率和功耗面积的需求有很大的差别。

10 当 Viterbi 检测器应用在低速的蜂窝电话系统时, 要求速度可以低于 1Mb/s, 但是要求有非常低的功耗; 在用于调制解调器中的网格编码解调时, 输出速率要求在几十 Kb/s, 但是对于功耗和芯片的面积以及价格上就有非常严格的限制; 在另一个极端, 高速的 Viterbi 检测器也用于磁盘驱动的读信道, 要求输出超过 600Mb/s, 相应的对于面积和功耗要求就比较低。

15 Viterbi 的实现结构同它所需要完成的功能需求紧密相关, 主要分为串行、并行和混合型结构。

1、串行设计: ACS22 每次从路径度量存储单元 25 读取一个状态 (共有 2^{m-1} 个状态) 的 PM 值, 即每个时钟只完成一个状态的计算, 2^{m-1} 个时钟处理网格表中的一列。这种结构不受约束长度 m 限制, 只消耗固定的面积, FPGA 的面积消耗最小, 但是译码延时也最大。

20 2、并行设计: ACS22 每次从路径度量存储单元 25 读取 2^{m-1} 个状态的 PM 值, 即一个时钟完成对所有状态的更新。并行设计的优点是可以实现高速 Viterbi 译码, 最高译码速率可以达到时钟频率; 需要的存储单元也比较少, 不需要两个路径度量存储单元乒乓式操作。它的缺点也显而易见: 硬件资源消耗大。

25 3、混合型: 又称串并结合型。ACS22 每次从路径度量存储单元 25 读取 n 个状态的 PM 值, 即每个时钟处理 n (其中, $n < 2^{m-1}$) 个状态, $2^{m-1}/n$ 个时钟周期完成网格表的一列。在具体实现时, 可综合考虑面积和速度来选择 n 值 (n 越大, 速度越快, 面积越大)。

-3-

可见,各种 Viterbi 结构主要的区别体现在 ACS22 单元, ACS22 单元也是 Viterbi 中需要资源最大的部件。

图 3 为常规的 ACS22 结构。两个加法器 31A、31B 用于将上一列对应状态的 PM 值与来自 BMU21 的分支度量值相加。两个上一列的 PM 值可以来自路径度量存储器(串行/混合型),也可以来自相应的上级寄存器并行,统称为路径度量存储单元 25。所得的值经过比较器(CMP)32,并通过 2 选 1 的多路选择器 MUX33 选择输出对应的幸存 PM 值和幸存路径值。ACS 单元是 Viterbi 译码器的核心电路,但是,PM 值会随着时间的推移一直累加下去,如果不加以控制,ACS 单元会有溢出的隐患,造成严重的译码错误。

Viterbi 译码器通过寻找最小 PM 值的路径来进行译码,由 ACS 单元的电路结构可知,我们只需保留各个状态 PM 值的大小关系即可,即是说所有 PM 值的基数对于 Viterbi 译码没有意义,可以去掉,而且这样做可以避免 PM 值的溢出问题。一般的控制方法是每隔一段时间比较所有的 PM 数值,找出其中最小的一个数值,然后将所有的 PM 值减去这个数值,这样即可以保证加法器不溢出,也可以保持各个 PM 数值之间的大小关系不变。在数字系统硬件实现时,采用的方法是:将每一列所有 PM(假设为 i 位)的最高位参与运算,如果结果为 '1',说明 $PM_{any} \geq 2^{i-1}$,那么就产生一个溢出控制信号,用这个信号将网格图中下一列所有的 PM 最高位置 '0',实际相当于将所有 PM 值减掉基数 2^{i-1} 。因此,希望 ACS 单元的数据位宽要足够大,既能保证各个路径的大小关系不变,又可以有效地解决溢出问题。

从改善硬件实现和关键路径角度出发,希望 ACS 单元的数据位宽尽量小。但是,如果 ACS 的数据位宽过小,既不足以保存各个路径的大小关系不变,也会出现 PM 值溢出的现象;如果 ACS 位宽过大,则会增加硬件实现电路和关键路径的延时,因此需要恰当的 ACS 位宽。

发明内容

本发明要解决的技术问题在于提供一种高可靠性的 Viterbi 译码器,以及一种能确定 Viterbi 译码器 ACS 单元最小位宽的方法,。

本发明提供的维特比译码器,包括顺序处理接收到的数据的分支度量

-4-

单元 BMU、加比选单元 ACS、幸存路径存储器、和回溯单元 TBU, 以及将所述加比选单元 ACS 选出的路径度量 PM 值在后继的步骤中再送回加比选单元 ACS 的路径度量存储单元, 所述维特比译码器的输入是 $(n_0, 1, m)$ 卷积码, 每个输入数据的位宽用 width 表示, 所述加比选单元 ACS 的数据位宽 i 由公式一、公式二和公式三同时决定, 所述公式一如下:

$$S_PM \leq n_0 \cdot (m-1) \cdot (2^{\text{width}} - 1);$$

所述公式二如下:

$$2^{i-2} \leq S_PM < 2^{i-1}$$

所述公式三如下:

$$(2^{i-1} - 1) + S_PM + \left\lfloor \frac{n_0}{2} \right\rfloor \cdot (2^{\text{width}} - 1) < 2^i.$$

如果满足所述公式一和公式二的所述 i 值全部不满足所述公式三, 则所述加比选单元的最小数据位宽为 $(i+1)$ 比特。

本发明提供的另一种维特比译码器, 包括顺序处理接收到的数据的分支度量单元 BMU (21)、加比选单元 ACS (22)、幸存路径存储器 (24)、和回溯单元 TBU (23), 以及将所述加比选单元 ACS (22) 选出的路径度量 PM 值再送回加比选单元 ACS (22) 的路径度量存储单元 (25), 其特征在于, 所述维特比译码器的输入是 $(n_0, 1, m)$ 卷积码, 每个输入数据的位宽用 width 表示, 所述加比选单元 ACS(22) 的数据位宽 i 由公式一、公式二和公式三同时决定, 所述公式一如下:

$$S_PM_{gs} S_PM \leq n_0 \cdot (m-1) \cdot (2^{\text{width}} - 1);$$

所述公式二如下:

$$2^{i-2} \leq S_PM_{gs} < 2^{i-1}$$

所述公式三如下:

$$S_PM_{gs} \leq H_PM_{(gs,00)} \cdot (2^{\text{width}} - 1),$$

其中, $H_PM_{(gs,00)}$ 指用于硬判决时, 向 Viterbi 译码器输入全 0 数据时的最大 PM 跨度; S_PM_{gs} 为网格图中的每一列 PM 跨度。

其中, 将所有状态的 PM 值清 0, 向 Viterbi 译码器输入全 0 的数据流,

-5-

然后考察所述加比选单元输出的全部度量值获得度量跨度 PM。

本发明公开了一种维特比译码器中决定加比选单元数据位宽的方法，用于确定所述加比选单元的最小数据位宽，所述维特比译码器的输入是 $(n_0, 1, m)$ 卷积码，每个输入数据的位宽用 width 表示，包括下述步骤：

- 5 第一步，根据公式一计算网格图中的每一列的路径度量值 (PM) 的跨度 S_PM ，所述公式一如下：

$$S_PM \leq n_0 \cdot (m-1) \cdot (2^{width} - 1);$$

第二步，再根据公式二确定 i 值，所述公式二如下：

$$2^{i-2} \leq S_PM < 2^{i-1};$$

- 10 第三步，验证所述 i 值是否满足公式三，所述公式三如下：

$$(2^{i-1}-1) + S_PM + \left\lfloor \frac{n_0}{2} \right\rfloor \cdot (2^{width} - 1) < 2^i;$$

如果所述 i 值满足所述公式三，则所述加比选单元的最小数据位宽为 i 比特；如果所述 i 值不满足所述公式三，则所述加比选单元的最小数据位宽为 (i+1) 比特。

- 15 按照下述步骤实现所述第一步、第二步和第三步：

第一步，根据所述 $(n_0, 1, m)$ 卷积码的生成序列，计算网格图中的每一列 PM 跨度 S_PM_{gs} 如下：

$$S_PM_{gs} \leq H_PM_{(gs,00)} \cdot (2^{width} - 1),$$

其中 $H_PM_{(gs,00)}$ 指用于硬判决时，向 Viterbi 译码器输入全 0 数据时的最大

- 20 PM 跨度；

第二步，再根据公式二确定 i 值，所述公式二如下：

$$2^{i-2} \leq S_PM_{gs} < 2^{i-1};$$

第三步，验证所述 i 值是否满足公式三，所述公式三如下：

$$(2^{i-1}-1) + S_PM_{gs} + \left\lfloor \frac{n_0}{2} \right\rfloor \cdot (2^{width} - 1) < 2^i.$$

- 25 本发明提出一种用于确定 ACS 单元最小位宽的方法，在解决 PM 值溢出问题的同时，可以有效地较少 ACS 单元的硬件实现面积和关键路径延时，

并能普遍适用于 Viterbi 的各种不同结构。在此基础上, 本发明提出了一种改进的 ACS 电路, 本电路通过增加简单的逻辑电路、进位位逻辑单元 A 和 B, 将各个加法器的最高位单独处理, 使其可以在 ACS 电路输出之前产生, 这样最高位积累单元就可以与 ACS 电路同时执行, 不增加关键路径的长度。

- 5 产生 ACS 的比较选择信号时, 先对 PM 值的最高位处理, 再处理低位值。因为软判决和硬判决的区别主要在 BMU 单元, 体现在 ACS 单元只是数据位宽的不同, 因此本发明可以兼容软判决和硬判决。

- 10 本发明的最高位累加单元针对 Viterbi 的各种结构采用相近的不同结构, 使得本发明可以普遍适用于串行、并行和混合型 Viterbi 结构, 而且不同 Viterbi 结构下的电路结构相差不大。

附图说明

本发明包括如下附图:

图 1 是一种典型的卷积编码器;

图 2 是 Viterbi 译码器的结构框图;

- 15 图 3 是常规的 ACS 电路结构;

图 4 是本发明一个具体实施例的 ACS 电路结构;

图 5 本发明一个具体实施例 ACS 电路的数据流图。

具体实施方式

下面结合附图对本发明做进一步详细说明。

- 20 本发明公开的技术方案是一种可以有效减小 ACS 单元数据位宽的方法, 并在此基础上提出一种 ACS 电路, 可以解决 PM 值溢出的问题。

首先说明决定 ACS 单元最小位宽的方法:

- 25 对于一般二进制 (n_0, k_0, m) 编码器来说, 每次输入的是 k_0 个信息元, 有 2^{k_0} 个可能的信息组, 这相应于从码树每一节点上分出的分支数有 2^{k_0} 条, 每个节点相应于 2^{k_0} 种不同的信息组输入, 并且每条都有 n_0 个码元, 作为与此相应的输出子码, 相应的每个 ACS 也有 2^{k_0} 个加法器以及 2^{k_0} 路数据比较器。一般在通信系统中采用的卷积码都取 $k_0=1$, 本实施例也是针对 $(n_0, 1, m)$ 卷积码的 Viterbi 译码器。另外, Viterbi 每接收一个数据, 用 width

位来表示。很显然,用 width 参数可以表示判决方式,即 width=1 为硬判决, width>1 为软判决。

基于以上的分析,本发明通过以下三步,可以通过确定最大 PM 值和最小 PM 值之差,即 PM 跨度值,来决定 ACS 单元所需的最小位宽。

- 5 1、对于 $(n_0, 1, m)$ 卷积码,网格图中的每一列的 PM 跨度(用 S_PM 表示)有: $S_PM \leq n_0 \cdot (m-1) \cdot (2^{width} - 1)$ 。

即对于 $(n_0, 1, m)$ 卷积码,每一列所有节点的最大 PM 跨度为 $n_0 \cdot (m-1) \cdot (2^{width} - 1)$,假设某一系列 t 的 PM 最小值状态为 S_X ,它的度量值为 PM_{min} 。在网格图中,经过 $(m-1)$ 列后,该状态就可以将其度量值延伸到所有的 2^{m-1} 个状态。我们知道对于 n_0 输出的卷积码由同一点出发的 PM 值在网格图上前进 $(m-1)$ 步之后,PM 跨度最大为 $n_0 \cdot (m-1) \cdot (2^{width} - 1)$ 。如果网格图前进 $(m-1)$ 步后,所有的网格状态都是由该点出发的延伸值,则度量跨度最大为 $n_0 \cdot (m-1) \cdot (2^{width} - 1)$;否则,说明剩余的 PM 值比由该点出发的延伸值还要小,即小于 $n_0 \cdot (m-1) \cdot (2^{width} - 1)$ 。

- 15 2、对于有确定生成序列的 $(n_0, 1, m)$ 卷积码,网格图中的每一列 PM 跨度(用 S_PM_{gs} 表示,指确定生成序列的 PM 跨度)还会有所降低,即:

$$S_PM_{gs} \leq H_PM_{(gs,00)} \cdot (2^{width} - 1),$$

20 其中 $H_PM_{(gs,00)}$ 指用于硬判决时,向 Viterbi 译码器输入全 0 数据的最大 PM 跨度。

由第 1 部分可知, $(n_0, 1, m)$ 卷积码有 $S_PM \leq n_0 \cdot (m-1) \cdot (2^{width} - 1)$,但是确定的生成序列使得各状态之间有着微妙的联系,网格图的拓扑结构总是保证会有更小的其它路径的度量值作为幸存路径 PM 值,因此会有 $S_PM_{gs} \leq S_PM$ 。因为网络拓扑结构与输入数据的软硬判决无关,因此,只需考虑硬判决情况,先求出最大硬判决跨度 $H_PM_{(gs, max)}$,然后再拓展到 width 位宽即可。

寻找确定生成序列的 $(n_0, 1, m)$ 卷积码的 $H_PM_{(gs, max)}$ 的步骤是这

样的：首先，将所有状态的 PM 值清 0，以免在初始状态引入 PM 跨度。接着，向 Viterbi 译码器输入全 0 的数据流，然后通过 Matlab 或其它软件，考察其 ACS 输出的全部度量值以及度量跨度，全 0 输入的序列使得 Viterbi 译码器的硬判决 PM 跨度最大。

5 $H_PM_{(gs,00)}$ 就是确定生成序列的 $(n_0, 1, m)$ 卷积码的 $H_PM_{(gs, \max)}$ 。

首先，介绍一下“源”和“汇”的概念，某个状态 S 的“源”和“汇”的定义如下：“源”指的是在网格上可以通过一条支路到达状态 S 的状态；

“汇”指的是网格上以 S 为“源”的状态。从网格图的分析可知，状态 0 和状态 $2^{m-1}-1$ 都是以自己作为“源”和“汇”的。以状态 0 为例，网格图
10 中，状态 $0 \rightarrow$ 状态 0 的输出值为“00”。因此，当以全 0 的比特流作为 Viterbi 译码器的输入流时，幸存路径就是 $0 \rightarrow 0 \rightarrow 0 \rightarrow 0$ 。这种幸存路径受其它状态的干扰最小，因为幸存路径始终是 0 状态，没有别的状态能够介入到幸存路径中；而且，对其它状态的 PM 值的影响也最大，因为总是 0 状态有最小的路径度量 (PM) 值，它的延伸值很容易成为其它状态的幸存路径。即
15 是说，它是最接近于 1 中所讨论的最大度量跨度情况的。因此，可以推出它就是度量跨度最大的路径，即 $H_PM_{(gs,00)}$ 就是确定生成序列的 $(n_0, 1, m)$ 卷积码的 $H_PM_{(gs, \max)}$ 。

同理，可知状态 $2^{K-1}-1$ 也是度量跨度最大的路径，也即是它们的度量跨度值是相同的，都为 $H_PM_{(gs, \max)}$ 。因为 $H_PM_{(gs,00)}$ 是由网格图的拓扑结构
20 决定的，因此经过足够多的列之后， H_PM 便不再增加，而是稳定到一定的数值，这就是要求的 $H_PM_{(gs,00)}$ 。

如果 Viterbi 每个输入数据用 width 位宽来表示 (即 width=1 时，为硬判决；width>1 时，为软判决)，则确定生成序列的 $(n_0, 1, m)$ 卷积码的
25 $S_PM_{gs} \leq H_PM_{(gs,00)} \cdot (2^{\text{width}} - 1)$ 。Viterbi 输入数据用 width 位宽来表示，就是将硬判决的输入数据 ‘0’ 和 ‘1’ 对应量化成 “0” 和 “ $2^{\text{width}}-1$ ”。在无噪声的情况下，也即是说将 $S_PM_{(gs, \max)}$ 从硬判决的 $H_PM_{(gs,00)}$ 变成了
 $H_PM_{(gs,00)} \cdot (2^{\text{width}} - 1)$ ；在有噪声的情况下，就 S_PM_{gs} 最大的全 0 输入而言，如果噪声很小，0 信号经过噪声信道对应量化成 “ $0 \sim 2^{\text{width}-1}-1$ ”，既是说依然是倾向于硬判决的 0 值，这会使得最小度量状态不变，而最小度量值增

-9-

大, 因而 $S_PM_{gs} \leq H_PM_{(gs,00)} \cdot (2^{width} - 1)$; 若噪声很大, 0 信号经过噪声信道对应量化成 “ $2^{width-1} \sim 2^{width}-1$ ”, 既是说 Viterbi 的输入信号倾向于硬判决的 1 值, 这就使得幸存路径不再是 0 状态, 因此也有 $S_PM_{gs} \leq H_PM_{(gs,00)} \cdot (2^{width} - 1)$ 。

- 5 3、对于 $(n_0, 1, m)$ 卷积码, 首先由 (1) 式确定 i 值。如果对应的 i 值能够满足 (2) 式 ((2) 式中 $\left\lfloor \frac{n_0}{2} \right\rfloor$ 表示不大于 $\frac{n_0}{2}$ 的最大整数), 则确定该 $(n_0, 1, m)$ 卷积码可以用 i 比特位宽的 ACS 单元; 如果对应的 i 值不能满足 (2) 式, 则说明 i 比特可能不满足 ACS 对数据范围的要求, 需要用 $(i+1)$ 比特位宽的 ACS 单元。如果是有确定生成序列的卷积码, 只需将 S_PM 换成相
- 10 应的 S_PM_{gs} , 求出对应的 ACS 位宽即可, 很显然 $i_{gs} \leq i$ 。

$$2^{i-2} \leq S_PM < 2^{i-1} \quad (1)$$

$$(2^{i-1}-1) + S_PM + \left\lfloor \frac{n_0}{2} \right\rfloor \cdot (2^{width} - 1) < 2^i \quad (2)$$

- 15 为了能够保存每一列的 S_PM , 可以用比 S_PM 值多一位的 i 比特来表示 ACS 单元数据, 但是, 还要保证当所有 PM 值的最高位都为 1 时, 最大的 PM 值不会溢出。因此, 需要 (2) 式来进行选择。

假设第 t 列的最小 PM 值为节点 S_X 的 $PM_{(min, t)}$, 则 $(t+1)$ 列的最小 PM 值 $PM_{(min, t+1)}$ 有:

$$PM_{(min, t)} \leq PM_{(min, t+1)} \leq PM_{(min, t)} + \left\lfloor \frac{n_0}{2} \right\rfloor \cdot (2^{width} - 1) \quad (3)$$

- 20 对于 (3) 的前半部分这里就不再叙述了, 可以在讲解卷积码的相关书籍中找到对应的解释; 对于 (3) 的后半部分, t 列的每个状态在 $t+1$ 时刻均可以延伸出两个状态, 分别对应输入值为 0 和 1。因为一般卷积码的 n_0 个输出都与输入即时相关, 所以相同状态的两个不同输入, 势必造成两种状态的输出互异, 因此从 S_X 延伸的两个状态最小值一定不大于 $PM_{(min, t)} + \left\lfloor \frac{n_0}{2} \right\rfloor \cdot (2^{width} - 1)$, 如果 $PM_{(min, t+1)}$ 依然是 S_X 的延伸, 则有:

$$25 \quad PM_{(min, t+1)} \leq PM_{(min, t)} + \left\lfloor \frac{n_0}{2} \right\rfloor \cdot (2^{width} - 1);$$

-10-

如果 $PM_{(\min, t+1)}$ 不是 S_X 的延伸, 则说明 $PM_{(\min, t+1)}$ 值会比 $PM_{(\min, t)} + \left\lfloor \frac{n_0}{2} \right\rfloor \cdot (2^{\text{width}} - 1)$

更小。综上所述, 可以得到 (3) 式。

由 (1) 式确定的 i 值, 如果能够满足 (2) 式则该 $(n_0, 1, m)$ 卷积码可以用 i 比特位宽的 ACS 单元; 如果不满足 (2) 式, 则需要用 $(i+1)$ 比特位宽的 ACS 单元。考虑极限情况, 当某一列的 $PM_{(\min, t)} = 2^{i-1} - 1$ 时, 由 (3) 式, 假设 $PM_{(\min, t+1)}$ 取最大值, 则有

$$\begin{aligned} PM_{(\min, t+1)} &= PM_{(\min, t)} + \left\lfloor \frac{n_0}{2} \right\rfloor \cdot (2^{\text{width}} - 1) \geq 2^{i-1}; \text{ 则该列的最大 PM 值为:} \\ PM_{(\max, t+1)} &\leq PM_{(\min, t+1)} + S_PM \\ &= PM_{(\min, t)} + \left\lfloor \frac{n_0}{2} \right\rfloor \cdot (2^{\text{width}} - 1) + S_PM \quad (4) \\ &= (2^{i-1} - 1) + \left\lfloor \frac{n_0}{2} \right\rfloor \cdot (2^{\text{width}} - 1) + S_PM \end{aligned}$$

如果满足 (2) 式, 则有 $PM_{(\max, t+1)} < 2^i$, 说明 $2^{i-1} \leq PM_{(\max, t+1)} < 2^i$ ($PM_{(\max, t+1)}$ 表示 $(t+1)$ 列的任意节点的 PM 值), 因此可以用 i 比特位宽的 ACS 单元; 如果不能满足 (2) 式, 说明当 $PM_{(\min, t+1)} \geq 2^{i-1}$ 时, 有 $PM_{(\max, t+1)} \geq 2^i$, i 比特数据不足以保存 PM 值差异, 需要增加 1 位, 用 $(i+1)$ 比特位宽的 ACS 单元。

通过以上三步可以确定任意 $(n_0, 1, m)$ 卷积码的对应 ACS 位宽。下面以 $(2, 1, 7)$ 卷积码和 $(3, 1, 7)$ 卷积码为例进行说明, 假设都是进行 $\text{width}=3$ 的软判决:

例 1, $(2, 1, 7)$ 卷积码: 由 (1) 式有 $2^6 < 2 \times (7-1) \times 2^3 = 84 < 2^7$, 所以 $i=8$, 而且由 (2) 式有 $2^7 - 1 + 84 + 1 \times 7 = 218 < 2^8$, 所以 $(2, 1, 7)$ 卷积码可以用 8 位的 ACS 单元。如果确定 $(2, 1, 7)$ 卷积码的生成多项式为 $(133)_8$ 、 $(171)_8$, 通过 Matlab 软件仿真可以得到 $H_PM_{(gs, 00)}=8$ 。则由 (1) 式有 $2^5 < 8 \times 7 = 56 < 2^6$, 所以 $i=7$, 而且由 (2) 式有 $2^6 - 1 + 56 + 1 \times 7 = 126 < 2^7$, 所以对于生成序列为 $(133)_8$ 、 $(171)_8$ 的 $(2, 1, 7)$ 卷积码可以用 7 位的 ACS 单元。

-11-

例 2, (3, 1, 7) 卷积码: 由 (1) 式有 $2^6 < 3 \times (7-1) \times 2^3 = 126 < 2^7$, 所以 $i=8$, 但是由 (2) 式有 $2^7 - 1 + 126 + 1 \times 7 = 260 > 2^8$, 因此 (2, 1, 7) 卷积码需要用 $8+1=9$ 位的 ACS 单元。如果确定 (3, 1, 7) 卷积码的生成多项式为 $(133)_8$ 、 $(145)_8$ 、 $(175)_8$, 通过 Matlab 软件仿真可以得到 $H_PM_{(gs,00)}=12$ 。

5 则由 (1) 式有 $2^6 < 12 \times 7 = 84 < 2^7$, 所以 $i=8$, 而且由 (2) 式有 $2^7 - 1 + 84 + 1 \times 7 = 218 < 2^8$, 所以对于生成序列为 $(133)_8$ 、 $(145)_8$ 、 $(175)_8$ 的 (3, 1, 7) 卷积码可以用 8 位的 ACS 单元。

10 在确定 ACS 单元位宽 i 之后, 本发明实施例提出一种改进的 ACS 电路, 可以有效地解决 PM 值溢出问题, 而且相对于传统的 ACS 设计也减小了硬件实现和关键路径延时, 并且能够普遍适用于 Viterbi 的并行/串行/混合型结构。

如图 4 所示是本发明的改进 ACS (输入数据为 i 比特位宽) 电路单元结构图, 在该电路中, 并没有采用 i 比特位宽的加法器、比较器, 而是采用 $(i-1)$ 比特位宽加法器和比较器, 然后通过增加进位位逻辑单元 45A、45B 和 Logic 基本逻辑单元 46 对最高位进行单独处理。

下面给出图 4 中进位逻辑单元 45A、45B 和 Logic 基本逻辑单元 46 的真值表。

c_out	add_c1	over_bit	c1	over1
0	0	0	0	0
0	0	1	X	0
0	1	0	1	0
0	1	1	X	0
1	0	0	1	0

-12-

1	0	1	0	0
1	1	0	X	1
1	1	1	1	0

这是以 45A 为例 (X 状态为不可能出现的状态) 的进位逻辑单元真值表。它的输入有三个, 分别是网格图中上一列相应状态的进位输出 c_out 、本状态的加法器 41A 进位输出 add_c_1 和上一列的溢出控制位 $over_bit$, 输出有两个, 分别为本加法器的最高位输出 c_1 和该状态 PM 值溢出标志位 $over_1$ 。当出现 PM 值溢出问题时, 对应于真值表中的 $\{c_out, add_c_1\} = 2'b11$, 即上一时刻的最高位输出为 1, 本时刻的加法器 41A 进位位也为 1, 说明本数据已经两次超过 2^{i-1} , 即已经超过 2^i , 如果这时候 $over_bit=1$, 即所有状态 PM 值都已经大于等于 2^{i-1} , 所以 PM 值可以减掉 2^{i-1} , 进位位 c_1 输出为 1; 如果这时候 $over_bit=0$, 则说明这时本加法器的输出已经超过 2^i , 但是还没有所有状态 PM 值都大于等于 2^{i-1} , 则该 PM 值一定是被舍弃的 PM 值, 所以, 我们令 c_1 输出为 X, 而 $over_1$ 为 1 (表示该数即将被舍弃)。通过对真值表的分析可以得出 $c_1 = c_out \overline{over_bit} + add_c_1$, $over_1 = c_out \& \overline{over_bit} \& add_c_1$ 。

对于基本逻辑单元 Logic46, 它的输入有五个分别是两个加法器 41A、41B 的进位位 c_1 、 c_2 , 溢出标志位 $over_1$ 、 $over_2$ 和比较器 (CMP) 42 的输出结果 x , 输出有两个, 一个是选择电路 (MUX) 43 的选择信号 a_1 和最高位的选择结果 a_2 。这里, 溢出标志位 $over_1$ 、 $over_2$ 是为了防止非幸存路径的溢出问题。

为了分析逻辑关系方便, 先分析 c_1 、 c_2 和 x 的逻辑关系, 令它们的输出分别对应为 a_1_temp 和 a_2_temp , 再考虑 $over_1$ 、 $over_2$ 、 a_1_temp 和 a_2_temp 之间的逻辑关系。

所以, Logic 基本逻辑单元 46 分为两个真值表格来表示。

-13-

c ₁	c ₂	x	a _{1_temp}	a _{2_temp}
0	0	0	0	0
0	0	1	1	0
0	1	0	0	0
0	1	1	0	0
1	0	0	1	0
1	0	1	1	0
1	1	0	0	1
1	1	1	1	1

over ₁	over ₂	a _{1_temp}	a _{2_temp}	a ₁	a ₂
0	0	0	0	0	0
0	0	1	1	1	1
0	1	0	0	0	0
0	1	1	0	0	0
1	0	0	1	1	1
1	0	1	1	1	1
1	1	0	0	X	X
1	1	1	1	X	X

对于第二部分, {over₁, over₂} = 2'b11 时, 即本 ACS 单元的两个 PM 值都出现溢出现象, 通过前面决定 ACS 数据位宽的分析, 可知这是不可能发

-14-

生的情况，可以不用理会。通过对两个真值表的分析可以得出：

$$a_1 = \overline{over_1} + \overline{over_2}c_1x + \overline{over_2}c_1\overline{c_2} + \overline{over_2}xc_2;$$

$$a_2 = \overline{over_1} + \overline{over_2}c_1c_2$$

下面详细解释一下本电路的性能改进：

- 5 1、本发明实施例提供的能够解决溢出问题的 ACS 电路，虽然本电路增加了一些逻辑电路，但是，这些逻辑电路只是由简单的与或非门组成，消耗资源很少；而且，通过本方法已经使 ACS 的数据位宽达到最小。综合考虑，本发明有效地减小了硬件电路资源。

- 10 2、时间特性分析：图 5 给出本发明 ACS 电路的数据流图，假设最终确定 ACS 数据位宽为 i。该图与图 4 相比在电路表现形式上做了如下改动：将 Logic 中的 a_1 和 a_2 分开表示，并将 a_1 进行变形，如下式所示：

$$\begin{aligned} a_1 &= \overline{over_1} + \overline{over_2} \cdot c_1 \cdot x + \overline{over_2} \cdot c_1 \cdot \overline{c_2} + \overline{over_2} \cdot c_2 \cdot x \\ &= (\overline{over_2} \cdot c_1 + \overline{over_2} \cdot \overline{c_2}) \cdot x + (\overline{over_1} + \overline{over_2} \cdot c_1 \cdot \overline{c_2}) = ax + b; \end{aligned}$$

- 15 其中 $a = \overline{over_2} \cdot c_1 + \overline{over_2} \cdot \overline{c_2}$ ， $b = \overline{over_1} + \overline{over_2} \cdot c_1 \cdot \overline{c_2}$ 。因为 x 信号出现的较晚，因此 a_1 信号可以分为两部分执行：a、b 电路先执行；收到 x 信号后，再执行 $a_1 = ax + b$ 的操作。

- 20 从数据流的角度看，本发明 ACS 电路可以分成三个阶段，每个阶段的电路可以同时执行：第一阶段包括两个 (i-1) 位加法器 41A、41B 和 45A、45B 中的 $\overline{over_bit}$ 非门、两个与门相关的电路，很显然这个阶段的关键路径是 (i-1) 位加法器。第二阶段的电路在第一阶段电路之后执行，包括 45A 和 45B 中的两个或门 (c_1 、 c_2)、两个与门 ($\overline{over_1}$ 、 $\overline{over_2}$)、46 中的 a_2 、64 输入与非门、(i-1) 位 CMP(x) 以及 46 中的 a 和 b。而第三阶段电路 $a_1 = ax + b$ 只能在比较器 CMP 输出比较信号 x 之后执行。

- 25 在第一阶段电路之后，数据流分为三股，第一股数据流：与门 ($\overline{over_1}$ 、 $\overline{over_2}$)、或门 (c_1 、 c_2) → 与门 (a_2) → 最高位积累单元；第二股数据流：CMP (x)；第三股数据流与门 ($\overline{over_1}$ 、 $\overline{over_2}$)、或门 (c_1 、 c_2) → 46 中的 a 和 b。如图所示，后两股数据流交汇于 $a_1 = ax + b$ 部分，显然比较器决定了后两股

数据流的关键路径。第一股数据流与后两股数据流不相关，实际上三股数据流会合并成两股。

先分析 2^{m-1} 输入与门的延时：假设该多输入与门用二输入的与门级联实现，则需要 $(m-1)$ 级的二输入与门（第一级 2^{m-2} 个二输入与门，以此类推，最后一级为 1 个二输入与门），因此最大延时为 $t_{\max} = (m-1) \times t_{\text{二输入与门}}$ ，在实际实现时一定会采取延时更小的电路，延时一定不会大于 $t_{\max}$ 的值。最高位积累单元 44 延时最大的情况是 Viterbi 采用并行结构即需要 2^{m-1} 输入与门的情况，因此有 $t_{44} \leq (m-1) \times t_{\text{二输入与门}}$ 。因此，最高位积累单元延时小于 $(i-1)$ 位比较器 + $(a_1 = ax+b) + 2$ 选 1 的多路选择器。

10 综上所述，整个电路的关键路径为： $(i-1)$ 位加法器 + $(i-1)$ 位比较器 + 二输入与门 + 二输入或门 + 2 选 1 多路选择器。很显然，本电路使得关键路径有所降低。

3、本电路普遍适用于串行、并行与混合型 Viterbi 结构。在并行结构时，输入为上一列 2^{m-1} 个 ACS 单元的 PM 值最高位，进行 2^{m-1} 输入的与操作，输出为本列的 over_bit 位；在串行/混合型结构时，输入为上一列 n (n 为采用的 ACS 单元个数) 个 ACS 单元的 PM 值最高位以及上一列已经积累的 over_bit 值，进行 $n+1$ 输入的与操作，将上一列的所有 2^{m-1} 个状态的最高位都搜集全之后将本列的 over_bit 位输出。

20 综上所述，本发明公开了一种维特比译码器，包括顺序处理接收到的数据的分支度量单元 BMU、加比选单元 ACS、幸存路径存储器、和回溯单元 TBU，以及将所述 ACS 选出的 PM 值在后继的步骤中再送回 ACS 单元的路径度量存储单元，在所述加比选单元 ACS 后端设置最高位积累单元。

25 所述的最高位积累单元用来进行网格图中每一列的所有 2^{m-1} 个节点 PM 值的最高位与操作。在并行结构时，输入为上一列 2^{m-1} 个 ACS 单元的 PM 值最高位，进行 2^{m-1} 输入的与操作，输出为本列的 over_bit 位；在串行/混合型结构时，输入为上一列 n (n 为采用的 ACS 单元个数) 个 ACS 单元的 PM 值最高位以及上一列已经积累的 over_bit 值，进行 $n+1$ 输入的与操作，将上一列的所有 2^{m-1} 个状态的最高位都搜集全之后将本列的 over_bit 位输出。

30 在所述加比选单元 ACS 中还包括将各个加法器的最高位单独处理的

-16-

两个进位处理逻辑电路 A 和 B、以及简单逻辑电路；所述进位处理逻辑电路 A 的输入有三个，分别是网格图中上一列相应状态的进位输出 c_out 、本状态的加法器进位输出 add_c_1 和上一列的溢出控制位 $over_bit$ ，输出有两个，分别为本加法器的最高位输出 c_1 和该状态 PM 值溢出标志位 $over_1$ ，其

5 输入输出的逻辑关系为：

$$c_1 = c_out \& \overline{over_bit} + add_c_1;$$

$$over_1 = c_out \& \overline{over_bit} \& add_c_1;$$

所述进位处理逻辑电路 B 的输入有三个，分别是网格图中上一列另一个相应状态的进位输出 c_out 、本状态的加法器进位输出 add_c_2 和上一列的溢出控制位 $over_bit$ ，输出有两个，分别为本加法器的最高位输出 c_2 和该状态

10 PM 值溢出标志位 $over_2$ ，其输入输出的逻辑关系为：

$$c_2 = c_out \& \overline{over_bit} + add_c_2;$$

$$over_2 = c_out \& \overline{over_bit} \& add_c_2;$$

所述简单逻辑电路的输入有进位位 c_1 、 c_2 ，溢出标志位 $over_1$ 、 $over_2$ 和所述

15 比较器输出结果 x ；输出有选择器的选择信号 a_1 和最高位的选择结果 a_2 ，其输入输出的逻辑关系为：

$$a_1 = over_1 + \overline{over_2} c_1 x + \overline{over_2} c_1 \overline{c_2} + \overline{over_2} x c_2; \quad a_2 = over_1 + \overline{over_2} c_1 c_2,$$

所述最高位的选择结果 a_2 输入所述最高位积累单元，在将上一列的所有 2^{m-1} 个状态的最高位搜集全之后从所述最高位积累单元输出本列溢出控制位 $over_bit$ 。

20

本发明实施例公开的一种用于维特比译码器的加比选单元电路，包括顺序处理输入数据的两个加法器 A、B、比较器 CMP、和多路选择器 MUX，防止两个加法器 A、B 溢出的最高位选通电路包括将各个加法器的最高位单独处理的进位处理逻辑电路 A 和 B、以及简单逻辑电路，在最高位选通

25 电路后端设置最高位积累单元。

本发明公开了一种用于维特比译码器的加比选单元电路还具有下述附加技术特征：

-17-

所述进位处理逻辑电路 A 的输入有三个, 分别是网格图中上一列相应状态的进位输出 c_out 、本状态的加法器 A 的进位输出 add_c_1 和上一列的溢出控制位 $over_bit$, 输出有两个, 分别为本加法器 A 的最高位输出 c_1 和该状态 PM 值溢出标志位 $over_1$, 其输入输出的逻辑关系为:

$$\begin{aligned} 5 \quad c_1 &= c_out \& \overline{over_bit} + add_c_1; \\ over_1 &= c_out \& \overline{over_bit} \& add_c_1; \end{aligned}$$

所述进位处理逻辑电路 B (45B) 的输入有三个, 分别是网格图中上一列另一个相应状态的进位输出 c_out 、本状态的加法器 B 的进位输出 add_c_2 和上一列的溢出控制位 $over_bit$, 输出有两个, 分别为本加法器 B 的最高位输出 c_2 和该状态 PM 值溢出标志位 $over_2$, 其输入输出的逻辑关系为:

$$\begin{aligned} c_2 &= c_out \& \overline{over_bit} + add_c_2; \\ over_2 &= c_out \& \overline{over_bit} \& add_c_2; \end{aligned}$$

所述简单逻辑电路的输入有所述两个加法器 A、B 的进位位 c_1 、 c_2 , 溢出标志位 $over_1$ 、 $over_2$ 和所述比较器 CMP 的输出结果 x ; 输出有所述选择器 MUX 的选择信号 a_1 和最高位的选择结果 a_2 , 其输入输出的逻辑关系为:

$$a_1 = over_1 + \overline{over_2}c_1x + \overline{over_2}c_1\overline{c_2} + \overline{over_2}xc_2; \quad a_2 = over_1 + \overline{over_2}c_1c_2$$

所述最高位的选择结果 a_2 输入所述最高位积累单元, 在将上一列的所有 2^{m-1} 个状态的最高位搜集全之后从所述最高位积累单元输出本列溢出控制位 $over_bit$;

20 所述的最高位积累单元用来进行网格图中每一列的所有 2^{m-1} 个节点 PM 值的最高位与操作, 将上一列的所有 2^{m-1} 个状态的最高位都搜集全之后将本列的 $over_bit$ 位输出。

-18-

权 利 要 求

1、一种维特比译码器，包括顺序处理接收到的数据的分支度量单元 BMU (21)、加比选单元 ACS (22)、幸存路径存储器 (24)、和回溯单元 TBU (23)，以及将所述加比选单元 ACS (22) 选出的路径度量 PM 值在后
5 继的步骤中再送回加比选单元 ACS (22) 的路径度量存储单元 (25)，其特征在于，所述维特比译码器的输入是 $(n_0, 1, m)$ 卷积码，每个输入数据的位宽用 width 表示，所述加比选单元 ACS(22) 的数据位宽 i 由公式一、公式二和公式三同时决定，所述公式一如下：

$$S_PM \leq n_0 \cdot (m-1) \cdot (2^{\text{width}} - 1);$$

10 所述公式二如下：

$$2^{i-2} \leq S_PM < 2^{i-1}$$

所述公式三如下：

$$(2^{i-1} - 1) + S_PM + \left\lfloor \frac{n_0}{2} \right\rfloor \cdot (2^{\text{width}} - 1) < 2^i。$$

2、根据权利要求 1 所述的一种维特比译码器，其特征在于，如果满足
15 所述公式一和公式二的所述 i 值全部不满足所述公式三，则所述加比选单元的最小数据位宽为 $(i+1)$ 比特。

3、一种维特比译码器，包括顺序处理接收到的数据的分支度量单元 BMU (21)、加比选单元 ACS (22)、幸存路径存储器 (24)、和回溯单元 TBU (23)，以及将所述加比选单元 ACS (22) 选出的路径度量 PM 值再送
20 回加比选单元 ACS (22) 的路径度量存储单元 (25)，其特征在于，所述维特比译码器的输入是 $(n_0, 1, m)$ 卷积码，每个输入数据的位宽用 width 表示，所述加比选单元 ACS(22) 的数据位宽 i 由公式一、公式二和公式三同时决定，所述公式一如下：

-19-

$$S_PM_{gs} S_PM \leq n_0 \cdot (m-1) \cdot (2^{width} - 1);$$

所述公式二如下:

$$2^{i-2} \leq S_PM_{gs} < 2^{i-1}$$

所述公式三如下:

$$5 \quad S_PM_{gs} \leq H_PM_{(gs,00)} \cdot (2^{width} - 1),$$

其中, $H_PM_{(gs,00)}$ 指用于硬判决时, 向 Viterbi 译码器输入全 0 数据时的最大 PM 跨度; S_PM_{gs} 为网格图中的每一列 PM 跨度。

4、根据权利要求 3 所述的一种维特比译码器, 其特征在于, 将所有状态的 PM 值清 0, 向 Viterbi 译码器输入全 0 的数据流, 然后考察所述加比
10 选单元输出的全部度量值获得度量跨度 PM。

5、一种维特比译码器中决定加比选单元数据位宽的方法, 用于确定所述加比选单元的最小数据位宽, 所述维特比译码器的输入是 $(n_0, 1, m)$ 卷积码, 每个输入数据的位宽用 width 表示, 其特征在于, 包括下述步骤:

第一步, 根据公式一计算网格图中的每一列的路径度量值 PM 的跨度

15 S_PM , 所述公式一如下:

$$S_PM \leq n_0 \cdot (m-1) \cdot (2^{width} - 1);$$

第二步, 再根据公式二确定 i 值, 所述公式二如下:

$$2^{i-2} \leq S_PM < 2^{i-1};$$

第三步, 验证所述 i 值是否满足公式三, 所述公式三如下:

$$20 \quad (2^{i-1} - 1) + S_PM + \left\lfloor \frac{n_0}{2} \right\rfloor \cdot (2^{width} - 1) < 2^i;$$

如果所述 i 值满足所述公式三, 则所述加比选单元的最小数据位宽为 i 比特; 如果所述 i 值不满足所述公式三, 则所述加比选单元的最小数据位宽为 $(i+1)$ 比特。

6、根据权利要求 5 所述的一种维特比译码器中决定加比选单元数据位

-20-

宽的方法，其特征在于，按照下述步骤实现所述第一步、第二步和第三步：

第一步，根据所述 $(n_0, 1, m)$ 卷积码的生成序列，计算网格图中的每一列 PM 跨度 $S_{PM_{gs}}$ 如下：

$$S_{PM_{gs}} \leq H_{PM_{(gs,00)}} \cdot (2^{width} - 1),$$

5 其中 $H_{PM_{(gs,00)}}$ 指用于硬判决时，向 Viterbi 译码器输入全 0 数据时的最大 PM 跨度；

第二步，再根据公式二确定 i 值，所述公式二如下：

$$2^{i-2} \leq S_{PM_{gs}} < 2^{i-1};$$

第三步，验证所述 i 值是否满足公式三，所述公式三如下：

$$10 \quad (2^{i-1} - 1) + S_{PM_{gs}} + \left\lfloor \frac{n_0}{2} \right\rfloor \cdot (2^{width} - 1) < 2^i.$$

7、根据权利要求 6 所述的一种维特比译码器中决定加比选单元数据位宽的方法，其特征在于，确定 $H_{PM_{(gs,00)}}$ 的步骤包括：将所有状态的 PM 值清 0，向 Viterbi 译码器输入全 0 的数据流，然后通过 Matlab 或其它软件考察所述维特比译码器中所述加比选单元输出的全部度量值获得度量跨度。

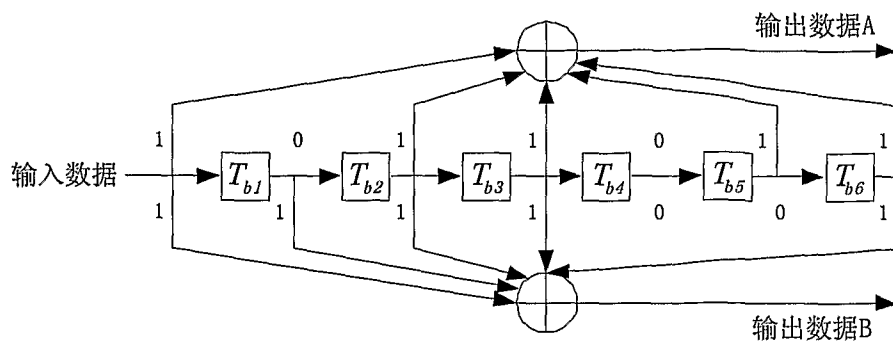


图 1

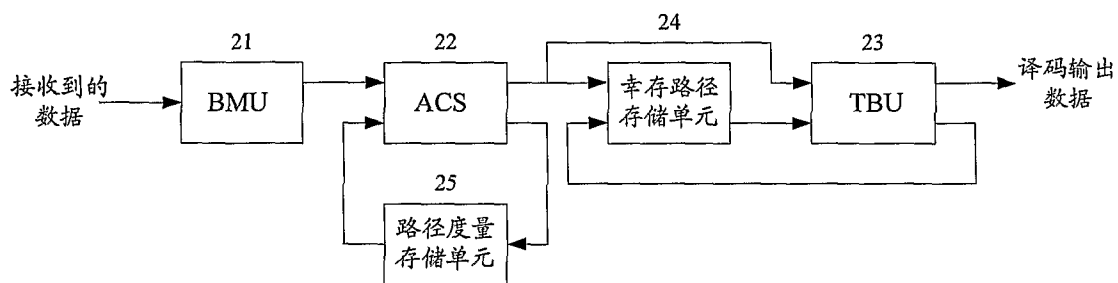


图 2

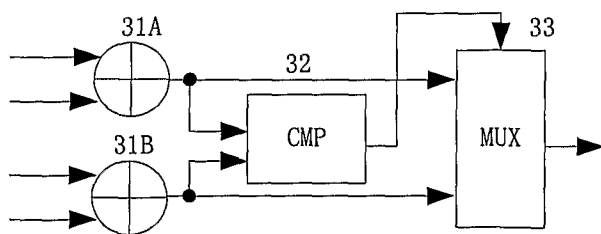


图 3

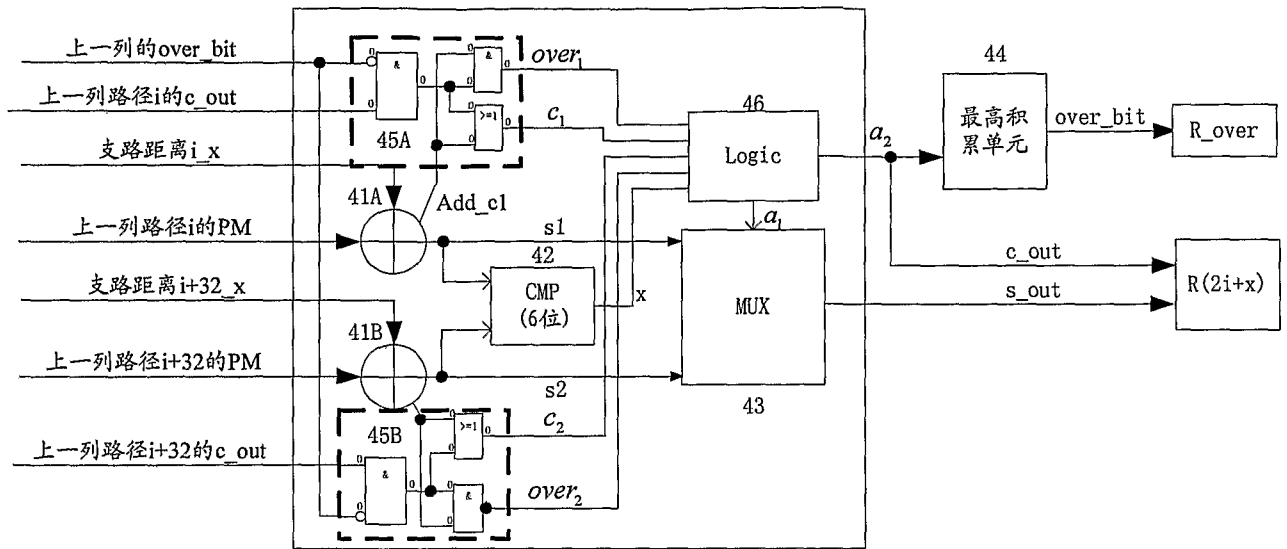


图4

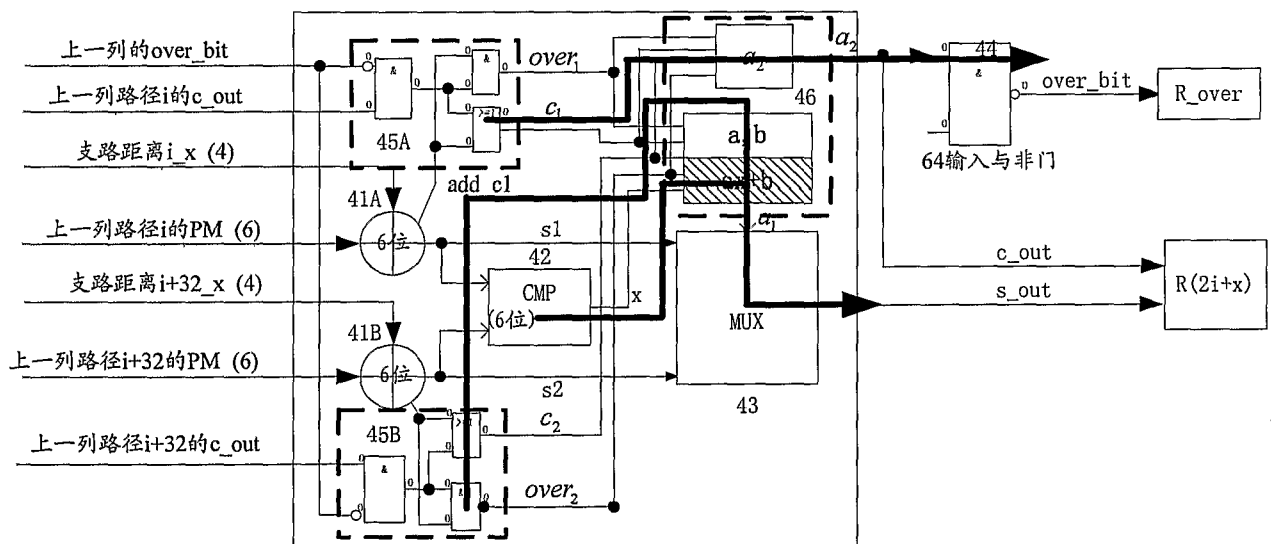


图5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2006/001989

A. CLASSIFICATION OF SUBJECT MATTER

H03M13/41(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H03M13/41(2006.01)i

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

CNPAT

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

WPI EPODOC PAJ CNPAT(Viterbi decod+ ACS path-metric width trace-back))

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	WO,A1,0103308,(FRAUNHOFER GES FOERDERUNG ANGEWANDTEN),11.Jan.2001(11.01.2001),the whole document.	1-7
A	US,A1,2003188142,(INTEL CORP),02.Oct.2003 (02.10.2003) ,the whole document.	1-7
A	CN,A,1409899,(SHARP KK),09.Apr.2003(09.04.2003),the whole document.	1-7

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:

“A” document defining the general state of the art which is not considered to be of particular relevance

“E” earlier application or patent but published on or after the international filing date

“L” document which may throw doubts on priority claim (S) or which is cited to establish the publication date of another citation or other special reason (as specified)

“O” document referring to an oral disclosure, use, exhibition or other means

“P” document published prior to the international filing date but later than the priority date claimed

“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

“&”document member of the same patent family

Date of the actual completion of the international search
10.Oct 2006(10.10.2006)

Date of mailing of the international search report

Name and mailing address of the ISA/CN
The State Intellectual Property Office, the P.R.China
6 Xitucheng Rd., Jimen Bridge, Haidian District, Beijing, China
100088
Facsimile No. 86-10-62019451

Authorized officer

WANG,Yankun

Telephone No. 86-10-62084987



INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2006/001989

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
WO,A1,0103308	11.Jan.2001(11.01.2001)	EP,A1,1192719 AU,A,5032299	03.Apr.2002 (03.04.2002) 22.Jan.2001 (22.01.2001)
US,A1,2003188142	02.Oct.2003 (02.10.2003)	NONE	
CN,A,1409899	09.Apr.2003 (09.04.2003)	AU,B2,2000229439B WO,A1,0169796 AU,B2,2000229439B KR,A,2002048963 EP,A1,1265367 AU,A,2943900 CN,C,1211933C KR,B,100490815B	14.Oct.2004 (14.10.2004) 20.Sep.2001 (20.09.2001) 24.Sep.2001 (24.09.2001) 24.Jun.2002 (24.06.2002) 11.Dec.2002 (11.12.2002) 24.Sep.2001(24.09.2001) 20.Jul.2005(20.07.2005) 24.May.2005(24.05.2005)

国际检索报告

国际申请号

PCT/CN2006/001989

A. 主题的分类

H03M13/41(2006.01) i

按照国际专利分类表(IPC)或者同时按照国家分类和 IPC 两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

H03M13/41(2006.01) i

包含在检索领域中的除最低限度文献以外的检索文献

CNPAT

在国际检索时查阅的电子数据库(数据库的名称, 和使用的检索词(如使用))

WPI EPODOC PAJ CNPAT(维特比 译码器 加比选 路径度量 宽 回溯,Viterbi decod+ ACS path-metric width trace-back)

C. 相关文件

类 型*	引用文件, 必要时, 指明相关段落	相关的权利要求
A	WO,A1,0103308,(FRAUNHOFER GES FOERDERUNG ANGEWANDTEN),11.1 月.2001(11.01.2001),全文.	1-7
A	US,A1,2003188142,(INTEL CORP),02.10 月 2003 (02.10.2003) ,全文	1-7
A	CN,A,1409899,(夏普公司),09.4 月 2003(09.04.2003), 全文	1-7

☐ 其余文件在 C 栏的续页中列出。

☒ 见同族专利附件。

* 引用文件的具体类型:

“A” 认为不特别相关的表示了现有技术一般状态的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件, 或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布, 与申请不相抵触, 但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件, 单独考虑该文件, 认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件, 当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时, 要求保护的发明不具有创造性

“&” 同族专利的文件

国际检索实际完成的日期

10.10 月 2006 (10.10.2006)

国际检索报告邮寄日期

23.11.2006 (23.11.2006)

中华人民共和国国家知识产权局(ISA/CN)

中国北京市海淀区蓟门桥西土城路 6 号 100088

传真号: (86-10)62019451

授权官员

王艳坤

电话号码: (86-10)62084987



国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2006/001989

检索报告中引用的 专利文件	公布日期	同族专利	公布日期
WO,A1,0103308	11.1 月 2001(11.01.2001)	EP,A1,1192719 AU,A,5032299	03.4 月 2002 (03.04.2002) 22.1 月 2001 (22.01.2001)
US,A1,2003188142	02.10 月 2003 (02.10.2003)	无	
CN,A,1409899	09.4 月 2003 (09.04.2003)	AU,B2,2000229439B WO,A1,0169796 AU,A,2000229439 KR,A,2002048963 EP,A1,1265367 AU,A,2943900 CN,C,1211933C KR,B,100490815B	14.10 月 2004 (14.10.2004) 20.9 月 2001 (20.09.2001) 24.9 月 2001 (24.09.2001) 24.6 月 2002 (24.06.2002) 11.12 月 2002 (11.12.2002) 24.9 月 2001 (24.09.2001) 20.7 月 2005(20.07.2005) 24.5 月 2005(24.05.2005)