

doi:10.3969/j.issn.1001-893x.2015.05.011

引用格式:王鹏,涂友超,龚克.弹载数据链系统实时 RS 译码器设计[J].电讯技术,2015,55(5):527-532.[WANG Peng,TU Youchao,GONG Ke.Design of Real-time RS Decoder for Missile-borne Data Link System[J].Telecommunication Engineering,2015,55(5):527-532.]

弹载数据链系统实时 RS 译码器设计*

王 鹏^{1,2,**},涂友超¹,龚 克¹

(1. 信阳师范学院,河南 信阳 464000;2. 中国空空导弹研究院 航空制导武器航空科技重点实验室,河南 洛阳 471009)

摘 要:为了提高传输可靠性,各种差错控制编码技术已经被广泛应用在弹载武器数据链系统中。RS(Reed-Solomon)码具有很强的抗错误能力,且码长可以灵活控制,十分适合在弹载数据链系统中应用。设计了三种不同码率的 RS 码,并在修正的欧几里德算法基础上进一步优化,实现了一种新型 RS 码实时译码器。为减少系统复杂度,该译码器复用 4 组基本运算单元以完成错误位置多项式和错误值多项式计算,同时也没有插入额外的流水线结构,译码过程所需的 GF(2⁸)域求逆运算则通过查找表结构实现。整个设计已经在 Altera 公司的 EP2S15 器件上通过综合和验证,与同类设计相比占用资源大大减少,适合于高可靠性导弹数据链系统开发。

关键词:弹载数据链;RS 译码器;修正的欧几里德算法;实时处理

中图分类号:TN919.6 **文献标志码:**A **文章编号:**1001-893X(2015)05-0527-06

Design of Real-time RS Decoder for Missile-borne Data Link System

WANG Peng^{1,2},TU Youchao¹,GONG Ke¹

(1. Xinyang Normal University, Xinyang 464000, China; 2. Aviation Key Laboratory of Science & Technology on Airborne Guided Weapons, China Airborne Missile Academy, Luoyang 471009, China)

Abstract:To improve transmission reliability, Forward Error Correction (FEC) techniques have been widely used in many missile-borne weapon data link systems. Reed-Solomon (RS) code has been proved to be efficient for correcting error. Also, its code length can be controlled flexibly, showing great advantage for missile-borne data link system. Three different RS codes are designed to adapt different environment. Furthermore, the widely used Modified Euclidean Algorithm for RS decoding is improved. A novel real-time RS decoder architecture is presented. To attain lower complexity, 4 basic calculation cells of decoder are multiplexed internally to solve the error-locator polynomial and error value polynomial. No additional pipeline architecture is inserted. Besides, the complicated inverse operation over GF(2⁸) is completed via Look-up Table (LUT). The whole design has been implemented successfully on EP2S15 FPGA chip of Altera Corp. Compared with existing RS decoders, the proposed architecture can reduce hardware resources greatly while still meeting required performance. So the presented strategy is much suitable for high-reliability missile data link system design.

Key words: missile-borne data link; RS decoder; modified Euclidean algorithm; real-time processing

1 引 言

数据链系统在导弹发射后仍能保持与载机、基

地的双向数据通信,导弹可根据这些信息完成对工作状态及动作轨迹的修正^[1-2]。因此,设计良好的

* 收稿日期:2014-10-22;修回日期:2015-04-02 Received date:2014-10-22;Revised date:2015-04-02

基金项目:国家高技术研究发展计划(863 计划)项目(2013AA7072012D);河南省高等学校重点科研项目(15A510011);河南省科技厅基础与前沿项目(142300410446)

Foundation Item: The National High-tech R&D Program of China (863 Program) (2013AA7072012D); The Key Research Foundation for Universities of Henan Province (15A510011); The Basic and Frontier Project of the Science & Technology Department of Henan Province (No. 142300410446)

** 通讯作者:pengpwn1985@163.com **Corresponding author:** pengpwn1985@163.com

数据链系统可以显著提高导弹作战性能,是整个武器系统的关键组件。近年来,由于作战环境日益复杂,各类电磁干扰层出不穷,新型导弹对数据链系统的性能要求也越来越高^[3]。

早期的数据链系统多采用模拟发射接收技术,但随着数字化技术的不断发展,目前新研数据链系统已经逐步转向数字化,此时数据链通道可视为一个典型的数字通信系统。而在数字通信系统中,采用前向纠错编码(Forward Error-correction Coding)可以极大地改善数据传输质量,降低误码率^[4],在各种数字通信中应用十分广泛^[5]。

由于数据链通信对实时性要求很高,因此不适合采用长度较长的纠错码,但同时又希望码字具有很强的纠错和抗突发错误能力,较为合适的选择就是里德-所罗门码(Reed-Solomon Code, RS Code),它最早由 Reed 和 Solomon 于 1960 年提出^[6],其最小距离等于它的奇偶校验符号数加一。RS 码是最重要的一类极大最小距离可分(Maximum Distance Separable, MDS)码,在纠正随机符号错误和随机突发错误方面非常有效,能提供很高的数据可靠性。目前,RS 码已经在许多通信系统中得到了广泛应用,如 DVB-T、IEEE802.16a/d 等。

RS 码虽然纠错能力很强,但译码复杂度也较高,其主要难点在于求解关键方程以得到错误位置多项式和错误值多项式,目前广泛使用的算法主要是 Berlekamp-Massey(BM)算法和 Euclidean 算法。同时,RS 码还可以在频域进行译码,第一种这样的译码算法由 Gore 提出^[7],之后由 Blahut^[8]进行了改进。但频域译码需要进行额外的傅里叶变换和反变换处理,增加了计算复杂度,因此一般情况下时域 RS 译码可以更有效地用于硬件实现。

谢大刚等人^[9]对 RS 码在数据链系统中的应用开展了初步分析与仿真,证实了 RS 码的引入可以大幅度改善数据链系统的抗干扰性能,但该文仅限于 Matlab 仿真,没有具体的硬件电路实现。此外,刘翠海等人^[10]也在某数据链模拟系统中使用了 RS 码,但设计是通过 VC++ 软件完成,译码实时性仍难以达到弹载平台的要求。

针对上述研究现状,并考虑到现代导弹对数据链系统的需求,本文设计了三种不同的 RS 码,长度均为 240 字节,分别是 RS(240,224)、RS(240,192)和 RS(240,176)码,均由长度为 255 字节的 RS 码缩短 15 字节而成,这三种码率下可以纠正的最大错误字节数分别是 8、24 和 32,可适合于不同的信道

环境。RS 码的每个码元取自域 $GF(2^8)$,其域生成多项式为 $p(x) = x^8 + x^4 + x^3 + x^2 + 1$ 。

弹载数据链是典型的实时通信系统,传统基于 CPU 架构的软件串行执行机制难以满足需要,因此本文的研究重点是 RS 译码器的硬件电路实现。按照设计要求,系统最高净比特符号率为 16.243 Mb/s。传统的 RS 译码算法需要 $2t$ 个运算单元级联阵列以完成一次迭代运算,其中 t 为 RS 码可纠正最大错误符号数,对应于该 RS 码, t 最大为 32,此时如果依然采用该设计,则需要 64 个基本运算单元,且每个运算单元均包含有 4 个 $GF(2^8)$ 域乘法器和 2 个加法器,因此该方案会占用极大资源。考虑到 RS 码最高比特符号率为 16.243 Mbps,因此本文采用复用改进 Euclidean 算法(ME)运算单元的方式来完成 RS 译码,共采用 4 个 ME 单元,即可在 16.243 MHz 的工作时钟下完成所有三种码率下的 RS 实时译码。整个译码器已经在 Altera 的 Stratix II 系列 EP2S15 器件上布局布线成功,占用资源仅为 6860 个 ALUT,比完全并行版本占用资源大大减少^[11-12],布线后最高工作时钟达到了 72.54 MHz,已远超过系统要求,而且整个设计的参数完全是可配置的,只需作少量修改即可用于其他 RS 实时译码。

2 RS 译码算法

目前,时域译码依然是 RS 译码算法的主流,它主要包括三个步骤,即校正子计算、关键方程求解(Key Equation Solver, KES)以及错误位置和错误值计算。其中 KES 部分是整个算法中最复杂的环节, BM 算法和 Euclidean 算法就是两种最常用的 KES 求解算法,但传统的 BM 和 Euclidean 算法需要完成 GF 域中复杂的求逆运算,很难达到较高的工作频率。Brent 和 Kung 在 Euclidean 算法的基础上提出了修正的 Euclidean 算法^[13],即 ME 算法,消除了 GF 域中复杂的求逆运算,可以达到更低的译码复杂度。设 RS 码字长度为 $n(n=2^m-1)$,信息码长度为 k ,可纠正最大符号数记为 t ,接收向量 $R(x) = R_{n-1}x^{n-1} + R_{n-2}x^{n-2} + \dots + R_1x^1 + R_0x^0$,则 ME 算法可描述如下:

(1) 校正子 S 计算,共有 $2t$ 个校正子,分别记作 $S_1, S_2, S_3, \dots, S_{2t}, S_i$ 计算公式如下(α 为 $GF(2^m)$ 上的本原):

$$S_i = \sum_{j=0}^{n-1} R_j (\alpha^i)^j; \quad (1)$$

(2) 利用 ME 算法迭代解出关键方程,首先初始化定义在 $GF(2^m)$ 上的多项式 $R_0(x) = x^{2t}, Q_0(x) =$

$S(x), L_0(x) = 0, U_0(x) = 1$, 然后令 $\deg(R)$ 和 $\deg(Q)$ 分别代表 $R(x)$ 和 $Q(x)$ 的度, a^i 和 b^i 则代表 $R(x)$ 和 $Q(x)$ 的最高项非零系数, $l = \deg(R(x)) - \deg(Q(x))$, 进行如下迭代:

若 $l_{i-1} \geq 0$, 则

$$\begin{cases} R_i(x) = [b_{i-1}R_{i-1}(x)] - x^{|l_{i-1}|} [a_{i-1}Q_{i-1}(x)] \\ Q_i(x) = Q_{i-1}(x) \\ L_i(x) = [b_{i-1}L_{i-1}(x)] - x^{|l_{i-1}|} [a_{i-1}U_{i-1}(x)] \\ U_i(x) = U_{i-1}(x) \end{cases}, \quad (2)$$

否则

$$\begin{cases} R_i(x) = [a_{i-1}Q_{i-1}(x)] - x^{|l_{i-1}|} [b_{i-1}R_{i-1}(x)] \\ Q_i(x) = R_{i-1}(x) \\ L_i(x) = [a_{i-1}U_{i-1}(x)] - x^{|l_{i-1}|} [b_{i-1}L_{i-1}(x)] \\ U_i(x) = L_{i-1}(x) \end{cases}; \quad (3)$$

(3) 该迭代过程最多持续 $2t$ 次, 当检测到 $\deg(R_i(x)) < t$ 时即可终止, 此时错误位置多项式 $\sigma(x) = L_i(x)$, 错误值多项式 $\omega(x) = R_i(x)$;

(4) 利用 Chien 搜索算法找出错误位置, 采用 Forney 公式 (见公式 (4)) 计算错误值 err_i :

$$err_i = \frac{\omega(\alpha^i)}{\sigma'(\alpha^i)} = \frac{\omega(\alpha^i) * \alpha^i}{\sigma_{\text{odd}}(\alpha^i)}. \quad (4)$$

式中, α^i 为 $\sigma(x)$ 的根, $\sigma_{\text{odd}}(x) = \sigma_1 x + \sigma_3 x^3 + \sigma_5 x^5 + \dots$ 。

计算出错误值后, 将其与错误位置处的原始接收码字在 $GF(2^8)$ 域相加, 即得到纠错后的码字。若当前位置不是错误位置, 则依然输出原始码字。

3 实时 RS 译码器设计

在三种 RS 码率中, 以 RS(240, 176) 码的纠错能力最强, 它最多可以纠正 32 个字节错误, 但其译码复杂度也最高, 当采用 ME 算法进行关键方程求解时, 在最坏情况下需要 64 次迭代才可以解出错误位置多项式和错误值多项式, 这也是整个 RS 实时译码器的瓶颈所在, 但整个系统对时钟要求并不高, 因此设计的重点是降低系统复杂度, 同时也要满足 RS 实时译码。

设计译码器工作时钟为 16.243 MHz, 码字传输模式为 8 位并行输入输出, 这样传输一组 RS 码字向量需要 $240 \times 8 = 1920$ 个周期。为满足实时要求, 译码器必须在最多 1920 个时钟周期内完成校正子

计算、求解关键方程、纠错码字输出这三个步骤。它们也分别对应于译码器的校正子计算模块、KES 计算模块以及码字纠错模块。其中在 KES 模块中, 若仅由一个 ME 单元完成 KES 运算, 则最坏情况下需要约 4000 个时钟周期才能装成关键方程求解, 无法完成实时数据处理, 因此共采用了 4 组 ME 单元并行操作, 将 KES 模块操作时钟压缩到了 1024 个时钟周期。同时, 传统 RS 译码器为提高输出效率, 常采用多级流水线设计^[14], 但该结构中每一级流水线均要寄存中间结果, 占用了大量寄存器资源, 同时各级流水线实际工作时钟并不匹配, 也造成了多级模块长期处于空闲状态, 硬件实际利用率很低。同时, 为进一步节省资源, 各模块之间并没有插入流水线, 而是采用顺序执行的方式完成 RS 译码过程。这样, 校正子计算模块每隔 255 个时钟即可输出计算完毕后的校正子, KES 模块最多耗时 1024 个时钟周期, 码字纠错输出模块最多需要 224 个时钟输出纠错后的信息符号, 因此即使 3 级模块顺序执行也足以在 1920 个时钟完成整个译码过程。

整个译码器框图如图 1 所示, 下面分别介绍各子模块设计。

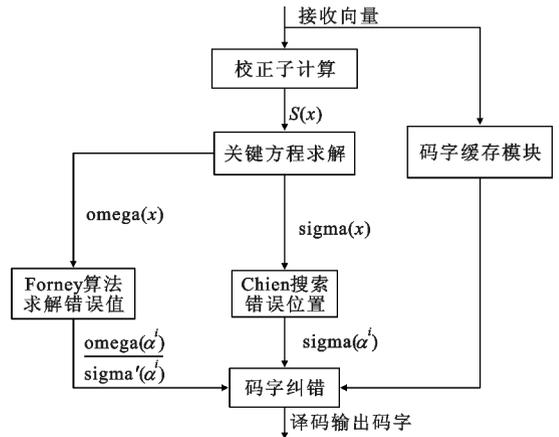


图 1 RS 译码器结构框图
Fig. 1 RS decoder architecture

3.1 校正子计算模块

校正子计算公式 (1) 可以展开如下:

$$S_i = R_{n-1}(\alpha^i)^{n-1} + R_{n-2}(\alpha^i)^{n-2} + \dots + R_1(\alpha^i) + R_0 = (\dots((R_{n-1}\alpha^i + R_{n-2})\alpha^i + R_{n-3})\alpha^i + \dots + R_1)\alpha^i + R_0. \quad (5)$$

式中, $1 \leq i \leq 64$ 。已经证明, 通过公式 (5) 所示的展开式来计算校正子是最优的^[15], 然而, 系统中采用的是由 RS(255, M) 系统码截短 15 字节得到的截短码, 因此若采用公式 (4) 进行校正子计算, 当所有

240 字节向量输入完毕后,还需乘以固定常数(α^{15})ⁱ才能得到最终校正子,这相当于额外增加了 $2t$ 个 $GF(2^8)$ 域常数乘法器,占用大量逻辑资源。为减少常数乘法器的使用,在本设计中采用将各校正子寄存器与固有的固定系数 α^i 常数乘法器进行额外循环操作 15 次来完成上述运算,相当于牺牲了 15 个时钟周期,但节省了 64 个 $GF(2^8)$ 域乘法器。

校正子计算模块结构如图 2 所示,接收 RS 码字向量以倒序方式输入,即传输顺序为 $R_{n-1}, R_{n-2}, \dots, R_1, R_0$ 。采用 64 组这样的结构即可在 255 个时钟周期内完成最多 64 个校正子计算。

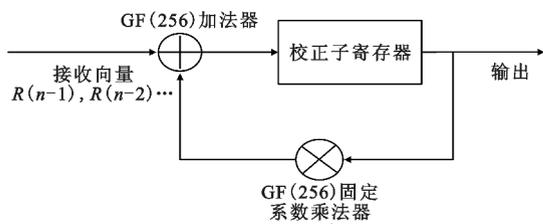


图 2 校正子 S_i 计算模块
Fig. 2 Syndrome S_i calculator module

3.2 KES 求解模块

KES 模块是整个译码器的核心,也是最复杂的模块,该设计采用了 ME 算法完成关键方程求解,其算法流程如前所述。

整个模块在计算过程中需要不断更新 R, Q, L, U 这 4 个定义在 $GF(2^8)$ 上的多项式的值,经分析得知,随着迭代次数的增加, R 和 Q 多项式的度不断下降,而 L 和 U 多项式的度则不断上升,在最坏情况下,当对 RS(240, 276) 进行译码时,迭代次数可达 64 次,可以得到 R, Q, L, U 多项式的最高度数分别是 64, 63, 32, 32, 而每个多项式系数均为 $GF(2^8)$ 中元素,需要用 8 位寄存器存储,因此在该模块中需要分别用 65, 64, 33, 33 组 8 位寄存器来存储 R, Q, L, U 多项式的全部系数,总计需要 1560 个单比特寄存器。

KES 模块中用于完成 R, Q, L, U 多项式系数更新的模块称作 ME 运算单元 (ME_cell),传统设计中通常采用 $2t$ 个这样的运算单元组成级联阵列,这样可以做到只耗费一个时钟周期就完成一次 ME 迭代运算,但该设计只在 t 较小时有一定优势,随着 t 的增加,占用资源会急剧上升,而该 RS 码最大纠错符号 t 达到了 32,并不适合采用该方案,因此在本设计中仅采用了 4 组 ME 运算单元,配合数据复用器

Mux 依次完成各多项式所有系数更新。

图 3 给出了 KES 模块的详细内部结构,首先由度检测模块对 R, Q 多项式度进行检测,若 $\text{deg}(R)$ 已经小于 t ,则由主控模块发出终止指令,直接输出计算结果,否则根据 R 和 Q 度大小不同设置对应的 Mux 数据选择器输出,并在送入 ME_cell 前预先完成公式(2)、(3)中的移位运算;然后由 4 组 ME_cell 来完成 R, Q, L, U 多项式系数的更新,每组 ME_cell 包含 4 个 $GF(2^8)$ 乘法器和 2 个 $GF(2^8)$ 域加法器,可完成公式(2)、(3)中的一组 R, Q, L, U 多项式系数更新。最坏情况下有 64 组多项式系数需要更新,因此 4 组 ME_cell 并行工作可以在 16 个时钟周期内完成一次 ME 算法迭代数据更新,最多耗费 $64 \times 16 = 1024$ 个时钟周期即可解出错误位置和错误值多项式。

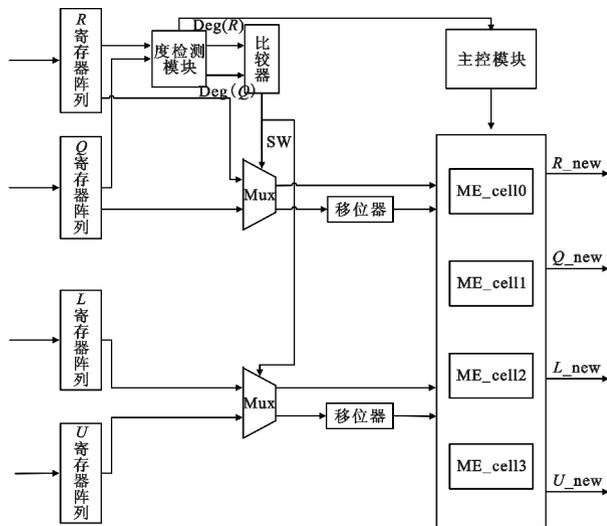


图 3 KES 模块结构图
Fig. 3 KES module architecture

与校正子计算模块所使用的常数 $GF(2^8)$ 域乘法器不同,该模块中 ME_cell 单元使用的是可变系数 $GF(2^8)$ 有限域乘法器,其复杂度也远高于前者,主要实现方法有查找表结构、标准基结构、对偶基结构以及正交基结构等。其中标准基结构不需要进行基的转换,可与任意输入、输出系统匹配,而且运算规则简单,因此更易于实现,本设计采用基于标准基的 Mastrovito 结构实现了有限域乘法器,并为乘法器输出增加了一级寄存器寄存,以减少关键路径延迟。

3.3 码字纠错输出模块

该模块完成 Chien 搜索错误位置和 Forney 纠错两大功能,然后输出纠错后的码字信息字节部分,当 Chien 搜索模块检测到错误位置时,将计算得到的

错误值与存放在缓冲区中的码字符号相异或,即可得到纠错后的码字。

传统的 Chien 搜索算法是从码字最高位开始译码,且输出是倒序后的所有码字,在本设计中对此作了改进,使之可以自然序只输出信息字节部分,其工作原理为:对码字第一个信息字节,相当于原始 RS(255, M) 码的第 16 个信息字节,要判断该位置是否为错误位置,只需判断 $\alpha^{(256-16)} = \alpha^{240}$ 是否为 $\sigma(x)$ 的根,以此类推,对码字第 i 个字节,首先计算 $\alpha^{(241-i)}$ 是否为 $\sigma(x)$ 的根,若是则根据 Forney 公式计算错误值进行纠错,否则直接输出原始码字。

图 4 是改进后的 Chien 搜索算法的硬件实现结构,其中 $\sigma_{\text{odd}}(\alpha^i)$ 输出用作 Forney 公式计算。

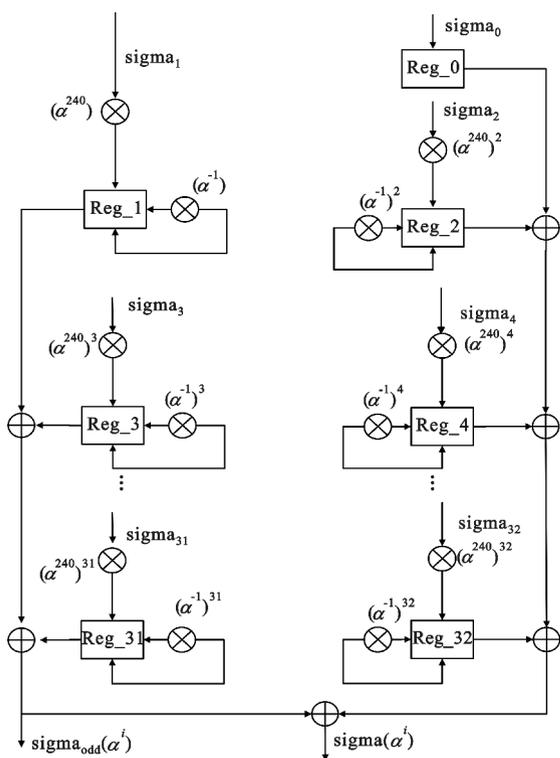


图 4 Chien 搜索模块结构图

Fig. 4 Chien search module architecture

同样地,将图 4 结构中的系数替换为 $\omega(x)$ 多项式系数,即可用作 Forney 公式计算,若结果为零,则当前位置即为错误位置,利用公式(4)计算出错误值,并与缓存中的原始码字相异或,即得到纠错后的码字并输出,公式(4)中 GF(2⁸) 域除法操作可通过乘以其逆元实现,逆元运算由一个由 ROM 构建的 256×8 位查找表完成。

该模块完成一次码字输出所占用时钟数恰为 RS 码信息字节数,最多耗费 224 个时钟周期。

4 译码器 FPGA 实现及验证

使用 RTL 级可综合 Verilog HDL 代码编写整个设计,在 ModelSim 10. 1d 下通过了功能和时序仿真,并在 Altera 公司的 EP2S15 器件上进行了实际验证。该译码器输出工作波形如图 5 所示,可以看到,在 Sbyte_start 置高后的下一个时钟周期,Data_valid 信号也变为高,代表对应输出的译码码字已经有效,图 5 输出译码码字为 0b00010011, 11111100... 经 Testbench 自动波形比较证实,该译码器可以纠正所有错误数小于等于 t 个错误情形。

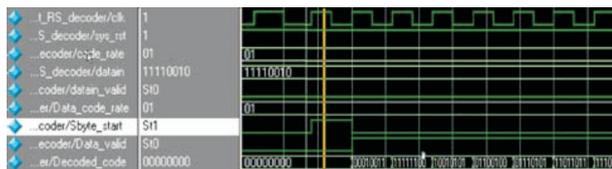


图 5 RS 译码器输出波形图

Fig. 5 Outputting waveform of RS decoder

译码器在 Quartus II 10. 0 下布线后具体资源占用情况如表 1 所示。

表 1 译码器占用资源统计

Table 1 Resource consumption of decoder		
资源	占用数量	占用百分比/(%)
ALUT	6860	54.9
寄存器	3773	30.1
Memory bit	6144	1.5

该设计可以支持 RS(240, 224)、RS(240, 192) 和 RS(240, 176) 所有三种 RS 码率的实时译码。设计布线后最高工作频率为 72. 54 MHz, 比采用 3 级流水线设计方案最高工作频率略低, 但占用资源与同类设计相比大大减少^[16-17], 而且事实上仅需 16. 243 MHz 的工作时钟即可完成系统设计要求的实时译码。

5 结束语

本文根据弹载数据链通道的特点, 设计了三种不同码率的 RS 码, 纠错能力从 8 字节到 32 字节不等。并设计了对应的低复杂度译码器, 为减少逻辑资源占用, 取消了模块间的流水线设计, 将校正子计算、关键方程求解、Chien 搜索和 Forney 计算全部放在一级执行, 仍可完成实时译码。该设计在性能和复杂度中取得了较好的折衷, 整个译码器也已经在 Altera 公司 EP2S15 器件上得到了验证, 完全可以用于低复杂度译码芯片开发。该项工作未来还将进一

步深入开展,一方面为了满足更高的实时性要求,需要对设计的关键路径进行优化,使最高工作频率达到100 MHz甚至更高;另一方面,希望将其他优秀的前向纠错编码如 BCH 码等与 RS 码级联,再配合交织技术,构建更为复杂的差错控制编解码算法,从而达到即使在强电磁干扰环境下仍能正常工作的要求。

参考文献:

- [1] 刘畅,杨云翔,夏薇. 2013年国外弹道导弹发展回顾[J]. 导弹与航天运载技术,2014,25(1):46-49.
LIU Chang, YANG Yunxiang, XIA Wei. Review of World Ballistic Missiles in 2013[J]. Missiles and Space Vehicles, 2014, 25(1): 46-49. (in Chinese)
- [2] 卫旭芳,吴催生,姚长虹. 基于变步长的 LMS 算法在数据链抗干扰中的应用[J]. 火力与指挥控制,2013,38(11):60-62.
WEI Xufang, WU Cuisheng, YAO Changhong. Application on Data Link Anti-Jam of Variable Step-Size LMS Method[J]. Fire Control & Command Control, 2013, 38(11): 60-62. (in Chinese)
- [3] 赵国艳,周林. 天基数据链抗干扰问题研究[J]. 电讯技术,2013,53(11):1412-1417.
ZHAO Guoyan, ZHOU Lin. Study on Anti-Jamming Issues of Space-based Data Link[J]. Telecommunication Engineering, 2013, 53(11): 1412-1417. (in Chinese)
- [4] Muntner M, Wolf J. Predicted performance of error-control techniques over real channels[J]. IEEE Transactions on Information Theory, 1968, 7(14): 640-650.
- [5] Bahl L, Cocke J, Jelinek F. Optimal decoding of linear codes for minimizing symbol error rate[J]. IEEE Transactions on Information Theory, 1974, 20(2): 284-287.
- [6] Reed S, Solomon G. Polynomial codes over certain finite fields[J]. Journal of the Society for Industrial and Applied Mathematics, 1960, 8(8): 300-304.
- [7] Gore W C. Transmitting Binary Symbols with Reed Solomon Codes [C]//Proceedings of 1973 International Conference on Information Science and System. Princeton, NJ: IEEE, 1973: 495-497.
- [8] Blahut R E. Transform Techniques for Error-Control Codes [J]. IBM Journal of Research and Development, 1979, 34(5): 299-315.
- [9] 谢大刚,吴楠,王春. 数据链系统 RS 码性能仿真与分析[J]. 微波学报,2010,8(4): 282-284.
XIE Dagang, WU Nan, WANG Chun. Performance Analysis and Simulation of RS Code in Data Link System[J]. Journal of Microwaves, 2010, 8(4): 282-284. (in Chinese)
- [10] 刘翠海,姜波,温东. RS 码类的构建及在数据链模拟系统中的应用[J]. 电讯技术,2013,53(1):63-67.
LIU Cuihai, JIANG Bo, WEN Dong. Construction of a Class for RS Code and Application in Data Link Simulation System[J]. Telecommunication Engineering, 2013, 53(1): 63-67. (in Chinese)

- [11] Ueng Y L, Yang C J, Wang K C, et al. A Multimode Shuffled Iterative Decoder Architecture for High-Rate RS-LDPC Codes [J]. IEEE Transactions on Circuits and Systems, 2010, 57(10): 2790-2803.
- [12] 张为,潘博阳,王皓. 一种基于 LCC 算法的新型 RS 码译码器[J]. 北京理工大学学报,2013,20(3):276-279.
ZHANG Wei, PAN Boyang, WANG Hao. A novel RS Recoder based on LCC Algorithm [J]. Transactions of Beijing Institute of Technology, 2013, 20(3): 276-279. (in Chinese)
- [13] Richard P B, Kung H T. Systolic VLSI Arrays for Polynomial GCD Computations [R]//CMU Computer Science Department Report. Pittsburgh, Pennsylvania: Carnegie Mellon University, 1982.
- [14] Lin G, Xu Y, Liu B. An iterative decoding technique and architecture for RS concatenated TCM coding systems [J]. IEEE Transactions on Consumer Electronics, 2010, 56(3): 1288-1296.
- [15] Shu L, Costello D J. Error Control Coding [M]. Davis, California: Pearson Education, Inc., 2004: 148-149.
- [16] 石宇,黑勇,乔树山. 一种用于 PLC 系统的多码率 RS 码译码器[J]. 微电子学与计算机,2014,31(2):57-61.
SHI Yu, HEI Yong, QIAO Shushan. A multi-rate RS decoder for PLC system [J]. Microelectronics & computer, 2014, 31(2): 57-61. (in Chinese)
- [17] 屈鹏,柏鹏. 一种 RS(15,9)译码器的 FPGA 实现[J]. 科学技术与工程,2013,14(4):2980-2985.
QU Peng, BAI Peng. FPGA based Realization of a RS(15,9)decoder [J]. Science Technology and Engineering, 2013, 14(4): 2980-2985. (in Chinese)

作者简介:



王鹏(1985—),男,河南太康人,2010年于同济大学获工学博士学位,现为高级工程师,主要研究方向为雷达信号处理;

WANG Peng was born in Taikang, Henan Province, in 1985. He received the Ph. D. degree from Tongji University in 2010. He is now a senior engineer. His research concerns radar signal processing.

Email: pengpwn1985@163.com

涂友超(1971—),男,河南信阳人,2011年于南京工业大学获博士学位,现为副教授,主要研究方向为无线电引信;

TU Youchao was born in Xinyang, Henan Province, in 1971. He received the Ph. D. degree from Nanjing University of Technology in 2011. He is now an associate professor. His research concerns wireless radar fuse.

龚克(1977—),男,河南淮阳人,2013年于东南大学获博士学位,现为副教授,主要研究方向为微波通信技术。

GONG Ke was born in Huaiyang, Henan Province, in 1977. He received the Ph. D. degree from Southeast University in 2013. He is now an associate professor. His research concerns microwave communications.