

# 50Gbps以太网的双模纠错代码解决方案

2017年10月

Venugopal  
Santhanam,  
新思科技ASIC数字设计工程师

## 简介

据《IEEE 802.3以太网带宽评估报告》预测，到2020年，对以太网端口速度的容量需求将达到10Tbps（每秒兆兆字节），从早期的1Mbps（每秒兆字节）跃升到10Gbps（每秒千兆字节），如今达到400Gbps。详见图1所示。按照这样高的速度，以太网物理层执行的时钟数据恢复电路至关重要。为了实现最高效的以太网组网，设计人员必须选择最优的以太网PCS配置，使其SoC能够满足高性能数据中心应用的要求。例如，100Gbps以太网配置可提供10个通路，达到10Gbps的数据速率，而这没有速率为25Gbps的4通路或50Gbps的2通路效率高。

带宽的提高推动着以太网物理层技术的更多创新，以应对多方面的挑战，例如信道损耗、信号间干扰，更重要的是错误检测和纠错。在传送和接收数据时，检测和纠正错误的同时保持较小硅片面积和较低功耗都很关键。前向纠错（FEC）就是这样一种技术，可以检测接收器中的突发错误并进行纠错，而不需要重传数据，因为数据的重传成本很高，而且导致效率低下和延迟。FEC基于Reed-Solomon编写的纠错代码，目前称为Reed-Solomon前向纠错（RS-FEC）。此后，IEEE 802.3标准定义了不同的RS-FEC模式，以支持当前的25Gbps以太网和不断演变的50Gbps以太网速度。本文探讨了以太网物理层实施的常见的Reed-Solomon前向纠错方法如何帮助SoC设计人员保证较小的硅片面积和较低的功耗，甚至是在超出10Gbps的高速运行情况下。



图1: 网速图 [1]

## 挑战: 不同通路速率的不同RS-FEC解码器

对于25Gbps以太网, IEEE 802.3bj标准定义了一种RS-FEC模式, 该模式提供了多达7个信号的纠错能力。对于具有4级脉冲振幅 (PAM-4) 符号的50Gbps PHY速率, IEEE 802.3cd标准定义了RS-FEC模式, 可提供多达15个符号的纠错能力。根据IEEE 802.3第73条自动协商结果, 端口中的PCS层可能需要支持/运行于25Gbps和50Gbps线路速率。每个RS-FEC模式下的代码解码功能都不同, 是物理层中消耗资源最多的部分。为了在同一个端口中支持25Gbps和50Gbps线路速率, RS-FEC解码器的两个专用实例分别执行每项解码功能, 分别是RS-FEC (528,514) 和RS-FEC (544,514), 这样导致硅片的面积变大。为了解决这一难题, 设计人员需要一种常用的RS-FEC解码器, 最终在物理层解决方案中实现单一的RS-FEC模式。

图2显示了支持4通路25Gbps和2通路50Gbps配置的100Gbps以太网解决方案的实施。

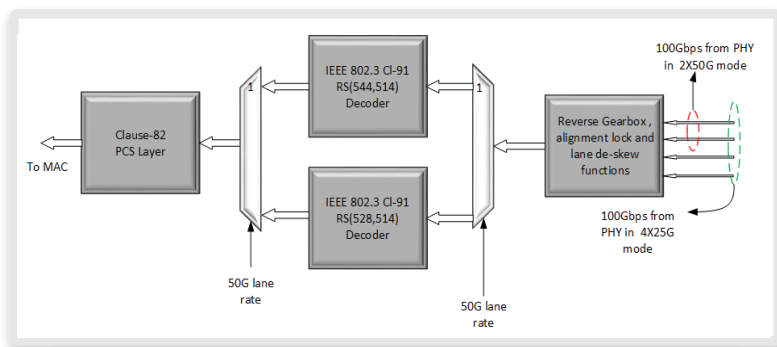


图2: 支持RS-FEC (528,514) 和RS-FEC (544,514) 解码功能的实施举例

Reed-Solomon代码 [2] 属于非二进制循环代码, 符号由 $m$ 位序列组成, 其中 $m$ 是任何正整数, 其值大于2。R-S ( $n, k$ ) 代码是 $m$ 位, 其数值是 $0 < k < n < 2m + 2$ , 其中 $k$ 代表编码的数据符号数字, 而 $n$ 代表编码块中代码符号的总数。对于最传统的R-S ( $n, k$ ) 代码,  $(n, k) = (2m-1, 2m-1-2t)$ , 其中 $t$ 是代码的符号纠错能力,  $n - k = 2t$ 是奇偶符号数量。

采用 $m$ 位符号的标准RS-FEC代码 ( $n, k$ ) 的解码器实施方式如下:

- ▶ 消息位数:  $k * m$  位
- ▶ 代码字中的位数:  $n * m$  位
- ▶ 奇偶位数:  $(n-k)*m$  位
- ▶ 伴随值数:  $2t = (n-k)$  个符号
- ▶ 纠错能力:  $t = (n-k)/2$  个符号

然后执行以下解码步骤:

- ▶ 伴随式计算 ( $2t$ ) 值用于检测收到的代码字中是否存在错误
- ▶ 错误定位 ( $\lambda$ ) 多项式 (程度) 和错误计值 ( $\Omega$ ) 多项式 (程度 $(t-1)$ ) 的关键方程
- ▶ Chien-Forney搜索, 用于查找收到的代码字中的错误指示
- ▶ 错误计值, 用于纠正检测到的错误。

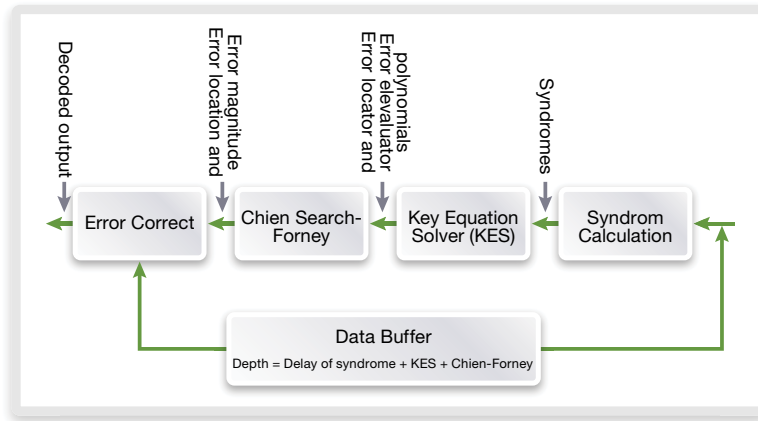


图3: RS-FEC解码器块实施

伴随式计算是每个时钟 (p) 的并行符号数函数, 因此, 从输入代码字开始起, 伴随值计算过程中的延迟为时钟的ceil (n/p) 数量。Ceil(x) 函数将参数 ‘x’ 舍入为最接近的整数。见下列所示:

x	cell (x)
2	2
2.4	3
2.9	3
-2.7	-2
-2	-2

由于伴随值, 代码字中错误的位置和幅度由错误指示多项式发现。一般情况下, 错误指示多项式采用Berlekamp-Massey (BM) 算法确定。采用无求逆BM算法 [3] 的错误指示多项式进行2t迭代。典型的实施包括折板结构, 其中的计算结果与同一个时钟周期内的多次迭代对应。例如, 如果每个时钟周期有2次迭代, ceil[2t/2] 时钟用于确定错误指示多项式 (λ)。

错误计值多项式用于发现收到的代码字中存在错误的大小。这采用错误定位多项式和伴随值而确定。有了错误指示符和伴随值, 错误计值多项式可以在 ‘t’ 迭代中发现。例如, 通过将3次迭代的计算结果包含到单个时钟周期内, 错误计值多项式就可以在ceil[t/3] 时钟周期内确定。一旦错误指示符和错误计值多项式可用, 代码字中的错误位置和错误大小可以使用Chien-Forney算法而发现 – 这是 “t” 和 “p” 函数。

在收到完整代码字时, RS-FEC解码器需要一些时间 (特定数量的时钟) 来确定错误指示符或错误计值多项式。在错误指示符或错误计值多项式计算过程中, 需解码的输入代码字被缓存。表1列出了两个RS-FEC解码函数的主要区别。

参数	值	RS FEC (528,514)	RS FEC (544,514)
伴随项数量	2t	14	30
错误指示多项式 (λ)	t	7	15
错误计值多项式次数 (Ω)	(t-1)	6	14
无求逆BerleyKemp算法迭代次数	2t + t	14 + 7 = 21	30 + 15 = 45

表1: RS FEC (528, 514) 和RSFEC (544, 514)的主要区别

图4显示了典型RS-FEC解码器的区域划分 (包括缓冲) 以及作为纠错能力函数 (t) 的复杂度, 以及输入/输出符号并行性(p)。

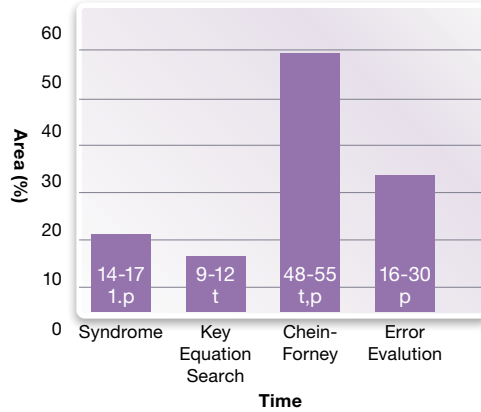


图4: RS-FEC解码器的区域划分和复杂度

## 建议的解决方案: 采用双模RS-FEC进行解码

单个解决方案可以支持两种RS FEC解码器, 因为RS-FEC解码函数 - (528,514) 和(544,514) - 在同一个伽罗瓦域GF(210) [2]定义, 并且使用相同的本原多项式  $p(x) = x^{10} + x^3 + 1$ , 并以 $\alpha$ 作为原根。因此, 通过对伴随值计算、错误指示和计值多项式进行适当而且谨慎的更改, 两个解码器函数可以共存在一个解码器实例中。下文着重介绍为支持双模式而必需对RS-FEC解码器不同子块进行的更改。

### 伴随值计算

RS-FEC (528,514) 中的伴随值计算逻辑是RS-FEC (544,514) 中计算逻辑的一个子集。因此, 根据运行模式, 逻辑的每个部分 (与RS-FEC (528,514) 模式相关) 或整个逻辑都被启用。

### 关键方程 - 错误指示与错误计值多项式

关键方程 (错误指示与错误计值多项式) 采用伴随值输入而完成计算过程。由于两个解码器的纠错能力不同, 多项式次数和迭代次数也不同。

关键方程解算器用于获取错误指示 ( $\lambda$ ) 和错误计值 ( $\Omega$ ) 多项式。这些多项式的次数在RS-FEC (528,514) 和RS-FEC (544,514) 解码函数中不同, 如下表2所示。

多项式次数	RS FEC (528,514)	RS FEC (544,514)
错误指示 ( $\lambda$ )	7	15
错误计值 ( $\Omega$ )	6	14

表2: RS-FEC (528,514) 和RS-FEC (544,514) 的关键方程区别

伴随值用于获取错误指示多项式。由于RS-FEC (528,514) 和RS-FEC (544,514) 模式下的伴随值项数不同 (分别是14和30), 只有与模式对应的有效伴随值项才用于计算错误指示多项式。

由于伴随值, 采用无求逆BM算法的错误指示多项式进行 $2t$ 迭代 (在RS-FEC (544,514) 中进行30次迭代, 在RS-FEC (528,514) 中进行14次迭代)。

由于一个时钟周期内进行 $N$ 次迭代, 这个过程需要 $\text{ceil}[2t/N]$ 个时钟, 对于RS-FEC (544,514) 和RS-FEC (528,514), 最大值分别是  $\lceil 30/N \rceil$  个时钟和  $\lceil 14/N \rceil$  个时钟, 用以确定错误指示多项式 ( $\lambda$ )。

有了错误指示符和伴随值，错误计值多项式可以在 ‘t’ 迭代中发现。通过将N次迭代的计算结果包含到单个时钟周期内，错误计值多项式就可以在 $\text{ceil}[t/N]$ 时钟周期内确定，对于RS-FEC (544,514) 和RS-FEC (528,514) 模式，分别是 $\text{ceil}[15/N]$  和 $\text{ceil}[7/N]$  时钟周期。由于 $N=3$ ，计算关键方程式所用的时钟周期数量见下表3所列。

多项式	迭代次数	RS (544,514)	RS (528,514)
错误指示 ( $\lambda$ )	30	$30/3 = 10$	$14/3 = [14/3] + 1 = 5$
错误计值 ( $\Omega$ )	15	$15/3 = 5$	$= 7/3 = [7/3] + 1 = 3$

表3: 双模关键方程迭代

### Chien-Forney搜索-解算关键方程式

Chien-Forney搜索的目的是查找错误指示的根源，这样做是为了指出错误在接收到的代码字中的位置。Chien Forney算法用于确定错误大小。

在RS-FEC (528,514) 模式下，错误位置和大小从接收到的528个符号组成的代码字中计算。在RS-FEC (544,514) 模式下，错误位置和大小从接收到的544个符号组成的代码字中计算。因此，根据运行模式，发现错误的符号位置可能位于有限的范围内，这样，在两个模式间可以使用相同的Chien-Forney搜索逻辑。

### 错误计值

通过从Chien-Forney块中获得的错误大小，纠错可以简单地对来自存储器并具有特定错误大小的数据进行XOR操作。两个解码器函数中的纠错块相同。

### 结果

下表4列出了独立和双模RS-FEC解码器解决方案的相关区域：

RS FEC解码器	相关区域
RS (528,514)	1.00
RS (544,514)	1.84
双模	2.01

表4: 独立和双模RS-FEC的相关区域

建议的双模解决方案显著减小了硅片面积：

- ▶ 在IEEE 802.3 -bj/cd 100Gbps模式下减少约200K个门
- ▶ 在IEEE 802.3 bs 200/400Gbps模式下减少约400K个门

在支持多端口的企业网元中，建议的解决方案为每个端口节省了200K/400K个门的面积。例如，支持128 个100Gbps端口的网络交换机可减少 $128 * 200K = \text{约}25.6\text{M}$ 个门。

建议的多模解决方案减少了静态功耗，因为仅适用于了一个解码器实例。除了支持RS-FEC (528,514) 和RS-FEC (544,514) 外，该解决方案还可通过扩展而适应RS-FEC (272,258) (25G联盟正考虑RS-FEC解码器的低延时版本 [8])。

## 结论

以太网端口速度已经从早期的1Mbps提高到10Gbps, 如今已达到400Gbps, 并且预计将达到1 Tbps。在定义网络设备间的以太网物理层信号速率时, 设计人员必须考虑物理编码子层的能力, 这使得选择最优以太网PCS配置的任务非常重要。PHY信号速率的提高也带来了挑战, 例如信道损耗、符号间干扰, 更重要的是错误检测和纠错。在发送和接收数据时, 必须检测并纠正错误, 同时保持较小的硅片面积和较低的功耗。

Reed-Solomon前向纠错 (RS FEC) 这样的纠错代码无需传输数据即可检测并纠正错误, 这样不仅成本很高, 而且导致效率低下和延迟。然而, 两种RS-FEC模式 – RS-FEC (544,514) 和RS-FEC (528,514) – 需要专用的实例, 因此, 设计人员需要在以太网物理层采用一种通用的实施方法。Synopsys的DesignWare企业PCS控制器IP支持这种实施, 同时显著减小了硅片面积, 并降低了静态功耗。

## 参考资料

1. <https://www.nextplatform.com/2016/03/24/construction-zones-ethernet-roadmap/>
2. [http://ptgmedia.pearsoncmg.com/images/art\\_sklar7\\_reed-solomon/elementLinks/art\\_sklar7\\_reed-solomon.pdf](http://ptgmedia.pearsoncmg.com/images/art_sklar7_reed-solomon/elementLinks/art_sklar7_reed-solomon.pdf)
3. Reed-Solomon解码器的高速架构 – Dilip V. Sarwate、Naresh R. Shanbhag
4. [http://www.ieee802.org/3/bj/public/mar12/gustlin\\_01\\_0312.pdf](http://www.ieee802.org/3/bj/public/mar12/gustlin_01_0312.pdf)
5. 基于复合域的低复杂性并行有限域乘法器结构 - Nihal kindap
6. 基于复合有限域的GF (214) 和GF (216) 乘法器并行结构 - Jianing Su, Zhenghao Lu
7. 低延时FEC建议, 2017年4月 - Eric Baden – Broadcom、Oded Wertheim - Mellanox
8. [DRAFT 0.1 LL FEC 25G Consortium.pdf](https://internal.25gethernet.org/wg/TWG/document/267)见<https://internal.25gethernet.org/wg/TWG/document/267>