

FEC Core Area Comparison and Model

Martin Langhammer
October 21, 2012

Overview

- **This presentation will show the relative areas of FEC cores used in recent 802.3bs meetings**
 - Focusing on Reed Solomon and BCH
- **A modelling method will be introduced to allow a quick area calculation for similar types of cores**
 - Only primary school math required
- **Quick tutorial on Reed Solomon and BCH core architectures**
 - Block diagrams

Caveats

- **This presentation does not consider the merits of any FEC**
 - Gain
 - Latency
 - Suitability for a channel or application
- **This presentation introduces a model to allow a relative area comparison of different Reed Solomon and BCH FECs**
 - Based on codeword parameters (n,k)
 - Throughput important consideration (parallelism)
 - Monolithic or individual pipes

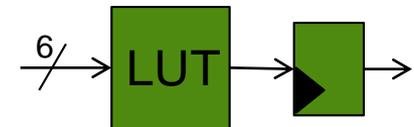
Modelling Complications - FPGA vs. ASIC

■ Memory vs. Logic

- FPGA has some amount of memory blocks interspersed with logic
 - Subfield Inversion (polynomial calculation, Forney)
 - Delay lines

■ Different types of logic

- FPGA typically basic building block 6 input LUT (look up table)
 - (Altera: ALM, Xilinx: 6LUT)
- FPGA Registers free with logic



■ Performance

- ASIC typically 650MHz, 2 clocks per polynomial iteration¹
- FPGA typically 325MHz, >>2 clocks per iteration

■ Latency vs. Latency

- 100ns ASIC vs. 250-350ns FPGA

■ Summary: exact comparison cannot be made, too many variables

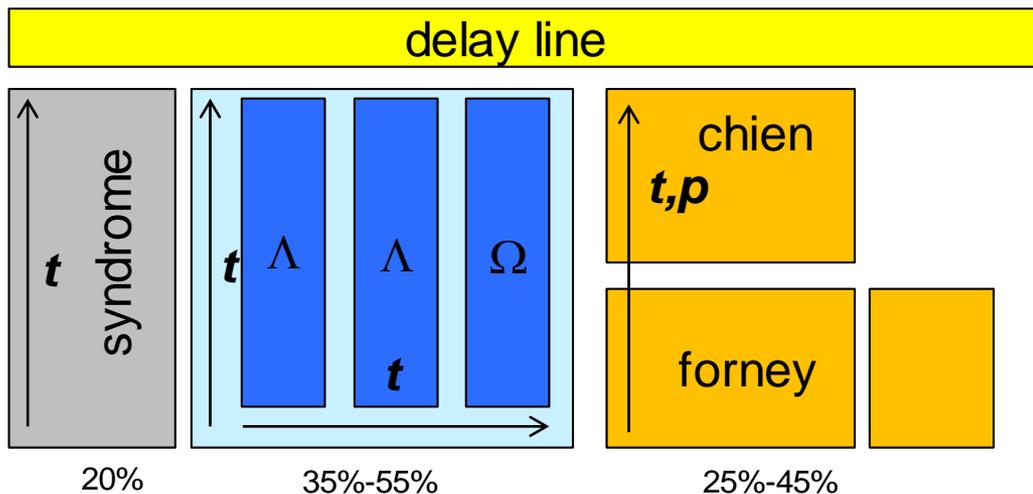
- First model will ignore effects of registers on area

1. Wang_z_3bs_01_0914 "In 100G KR, parallelism for RS-FEC is best set as 160bits/cyc."

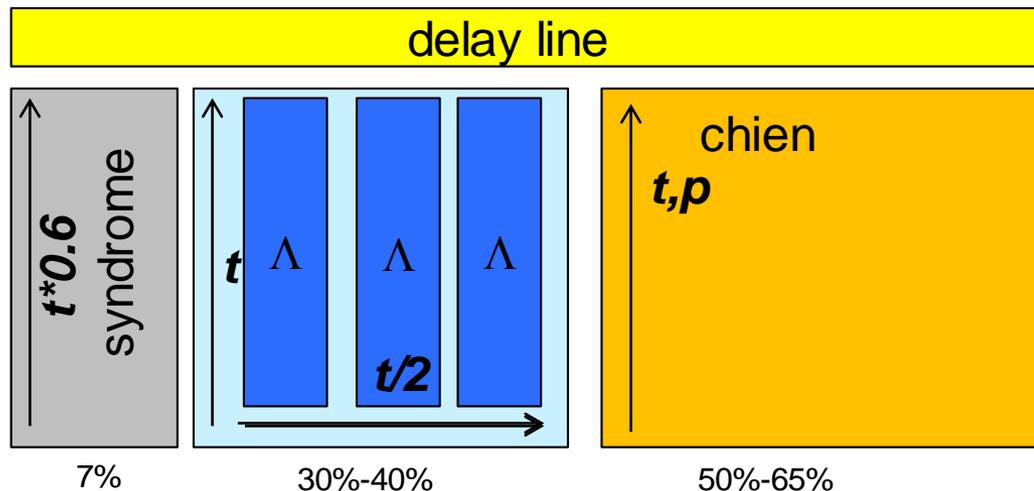
Modelling Complications - FPGA vs. FPGA

- **Different FPGA speed grades**
 - Slow, Medium, Fast (significant premium)
- **Medium typically 325 MHz+**
 - 330 bit wide input = 100Gbps
- **Fast typically 475 MHz**
 - 220 bit wide input = 100 Gbps
- **Latency longer – systolic array polynomial calculation**
- **New high performance 100G FPGA RS core**
 - Lower latency
 - Not in this analysis
- **Will focus on current technology, medium speed grade**
 - Volume production part – 2012 released technology
 - Available FEC Core
 - 325 MHz pushbutton (non-constrained) performance for any Reed Solomon and BCH parameters

Reed Solomon and BCH Block Diagrams



Reed Solomon



BCH

FEC Core Sizes

Type	Codeword	Area (6LUT)	Relative Area
RS KR4	(528,514,7)	10654	1
RS KP4	(544,514,15)	26554	2.5
BCH ¹	(2858,2570,24)	106806	10
BCH ²	(9193,8192,71)	425000	40

1. Cole_3bs_02b_0914
2. Takahara_3b_01a_0914

All results for mid-speed grade 28nm FPGA devices^{3,4}

3. 2012 production devices
4. mid range volume devices 200K-400K 6LUT



Reed Solomon vs. BCH Considerations

■ Syndromes

- Reed Solomon: calculate every syndrome
- BCH: calculate odd syndromes, generate even syndromes by $GF()^2$

■ Polynomial Calculation

- Reed Solomon: $2t$ iterations
- BCH: t iterations

■ Error Location and Value Calculation

- Reed Solomon: Chien and Forney
- BCH: Chien only

■ BCH $GF() > RS GF()$

- Area scaling proportional to $GF()^2$

■ BCH $t \gg RS t$ for same gain

- RS symbol based , BCH bit based, so t normalized = $t/GF()$

■ BCH OH $\gg RS OH$

BCH implementation simpler, but larger

Reed Solomon vs. BCH Area Calculation

■ Syndromes

- Reed Solomon: $p_{rs} \times m_{rs} \times t_{rs}$
- BCH: $p_{bch}/m_{bch} * t_{bch} * 0.6$
 - Effect of parallelism cancels out, **somewhat larger**

■ Polynomial Calculation

- Reed Solomon: t_{rs}^2
- BCH: $t_{bch}^2/2 \times 0.8$
 - $t_{bch} \gg t_{rs}$, a lot larger

■ Error Location and Value Calculation

- Reed Solomon: $p_{rs} \times t_{rs}$
- BCH: $p_{bch} * t_{bch} * 0.375$
 - $t_{bch} \gg t_{rs}$ and $p_{bch} \gg p_{rs}$, significantly larger

Worked Example – Reed Solomon to Reed Solomon

RS(528,514,7) @ 10654 6LUT=> RS(544,514,15) @ 26554 6LUT

- **Overall Scaling $(GF()_1/GF()_2)^2 = 1$**
- **Syndrome : 20% Area**
 - Scaling $(t_1/t_2) = (15/7) = 2.15x$
- **Polynomial Calculation : 35%σ55% Area**
 - Scaling $(t_1/t_2)^2 = 2.15^2 = 4.6x$
- **Correct (Chien, Forney) : 25%σ45% Area**
 - Scaling 1: (t_1/t_2)
 - Scaling 2: 0.75 (baseline Forney calculation) } 1.6x
- **Total $(0.2*2.15) + (0.4*4.6) + (0.4*1.6) = 2.9$**
 - Difference due to systolic array scaling – more efficient for longer vs. medium numbers

Worked Example – Reed Solomon to BCH

RS(528,514,7) @ 10654 6LUT => BCH (2858,2570,24) @ 106806 6LUT

■ **Overall Scaling $(GF()_1/GF()_2)^2 = 1.44$**

■ **Syndrome : 20% Area => 7% Area**

- Scaling 1: 0.6 (BCH syndromes odd only, use S^2 for even symbols)
 - Scaling 2: $(t_1/t_2) = (24/7) = 3.4x$
- } 2.1x

■ **Polynomial Calculation : 35%σ55%=> 30%σ40%Area**

- Scaling 1: $(t_1/t_2)^2 * 1/2 = 5.9$
 - Scaling 2: 0.8 (no Ω)
- } 4.7x

■ **Correct (Chien Only) : 25%σ45% Area => 55%σ65% Area**

- Scaling 1: $(p_1/p_2) = (330/33) = 10$
 - Scaling 2: $(t_1/t_2) = (24/7) = 3.4$
 - Scaling 3: No Forney : $0.5 * .75 = 0.375$
- } 12.5x

■ **Total $1.44(.20*2.1 + .4(4.7) + .4*12.5) = 10.5$**

Next Steps

■ Power modelling

- BCH and Reed Solomon have similar peak power requirements
 - Proportional to area
- BCH likely greater sustained power requirements
 - Error threshold switching

■ More complex analysis

- Complicated by lack of definitive area model

Summary

- **Accurate modelling difficult**
 - Technology differences – ASIC/ASIC, ASIC/FPGA, FPGA/FPGA
 - Different algorithms – polynomial calculations
- **BCH and Reed Solomon not apples to apples comparison**
 - Proposed codes have different gains
 - Gain vs. Gain differences depending on channel
 - Different error tolerances to bursts
- **In general BCH more expensive than Reed Solomon**
 - Larger field
 - Longer t
 - Greater p



Thank You