Preliminary Data Sheet

Blue-Salmon STORM-Series PP decoder

v01.1a

1. Introduction

Core Overview

The STORM-Series PP decoder is a RTL code that implements an optimized Blue-Salmon (BS) decoder core.

BS codes are a class of error-correcting codes used to detect and correct errors that might be introduced into digital data when it is transmitted or stored. Error-correcting codes incorporate redundancy in data. With this redundancy, only a subset of all possible transmissions contains valid messages. This means the valid codes are separated from each other so errors are not likely to corrupt one valid code into another. The encoded data can then be transmitted or stored.

When recovering data, a decoder first determines if a received message is valid. This step is called error detection. If an error is detected, the decoder finds the valid message “closest” to the received one. Provided the number of corrupted symbols (bytes) does not exceed a specified range, the message found is the one that was transmitted. Thus, the decoder conducts error correction.

Blue-Salmon forward error correction can be used where extreme speed and ultra-low latency are important and good error correction performance is required. For example, the core can be used in high-speed wired networking, satellite and space communications, digital video broadcast, and data storage.

An interesting property is reasonable burst error correction performance, as it does not matter how many bits are received in error within a single input symbol. BS codes can be used in concatenation with other codes. The BS code can correct some bursts of errors that sometimes occur when the concatened decoder fails. If necessary, the burst error correction capability can be further improved by using interleaved Blue-Salmon cores.

Key Features

- STORM-Series PP decoder - designed for standard data rate.
- Pipelined architecture for improved clock frequency.
- Ultra-low latency (*n*+3 clock cycles).
- Extreme performance error correction solution when paired with an encoder of the Blue-Salmon family.
- Parameterizable Blue-Salmon decoder core.
- Compatible with all other members of the Blue-Salmon Error Correction System.
- Encoded block size selectable from 3 to 255 symbols.
- Correction of the data and parity symbols.
- Fully synchronous design using a single clock.
- Supports continuous input symbols with no gap between encoded blocks.
- Low resources and modest core area.

2. Functional Description

Theory of Operation

Properties of Blue-Salmon codes

The BS code is a block code generally designated as BS(n, k) with m-bit symbols, where k is the number of data symbols per block, n is the number of symbols the encoded message contains, and the symbol size can be in a range from one to several bits. The encoded message called "encoded block" has $n - k$ redundant parity symbols. The code can correct up to $t = (n - k) / 2$ symbols.

The Blue-Salmon Error Correction System has the following characteristics:

- $m = 8$ number of bits per symbol
- $n = 3$ to 255 number of symbols per block
- $k = n - 2$ number of data symbols per block
- $t = 1$ number of symbols that can be corrected

The BS code is a systematic code, since the encoder simply appends the parity symbols to the otherwise unchanged original data sequence.

The BS code is a linear code. In practice, this means that every possible 8-bit word is a valid symbol. For instance, any 8-bit word can be transmitted directly in the data part of an encoded block, so the encoder does not care what the nature of the data is, whether it is a binary stream separated into blocks of 8-bit symbols, ASCII codes, etc. Given a symbol size 8, the maximum BS encoded block size is 255.

A corrupted symbol can have one or more (up to 8) erroneous bits. The BS code can correct one symbol error containing one to eight corrupted bits. This property makes the Blue-Salmon system a useful tool for protecting data impacted by burst errors.

Galois field math

BS codes are based on Galois fields (GFs), also called finite fields. Rules of the GF arithmetic are different from the usual arithmetic rules. For instance, GFs are finite fields. This means that any field element, as well as a result of the element addition and multiplication, can be presented by a fixed-length binary word. To generate and decode a BS code of 8-bit symbols, an 8-bit-wide GF is used.

Shortened codes

A shortened encoded block contains fewer symbols than the maximum 255. The shortened encoded block keeps the same number of parity symbols, 2, to correct 1 symbol error. Therefore, the number of data symbols in the shortened code is reduced by the same amount as the overall encoded block length. For instance, BS(200, 198) is the shortened code of BS(255, 253). Both codes have a symbol width of 8 and use the same number of parity symbols, 2.

Decoder latency

The latency is the number of clock cycles from a symbol being sampled on **data_in** to the corrected version of that symbol appearing on **data_out**.

In the Blue-Salmon Error Correction System the latency is only dependent on the value of *n* (the number of symbols per block). The following equation is used to calculate the latency:

- Latency = *n* + 3 cycles

Decoder data rate

The data rate is the number of symbols per clock cycle that the decoder can accept continuously.

In the Blue-Salmon Error Correction System the data rate is independent on the value of *n* (the number of symbols per block). The following equation is used to calculate the data rate:

- Data Rate = 1 symbol / cycle

Latency comparison

In the following tables the Blue-Salmon decoder is compared with two top performance highly optimized cores of another error correction system:

Brand "M" Core⁎⁎⁎⁎⁎ Reed-Solomon Decoder v3.1		Blue-Salmon STORM-Series * Standard Data Rate PP Decoder (this version)		Blue-Salmon Benefits vs. Brand "M"	
Configuration (examples)	Latency (cycles)	Configuration (examples)	Latency (cycles)	Δ Latency (cycles)	Δ Latency (ratio)
RS(8, 6)	34	BS(8, 6)	11	– 23	– 67.6 %
RS(16, 14)	34	BS(16, 14)	19	– 15	– 44.1 %
RS(32, 30)	64	BS(32, 30)	35	– 29	– 45.3 %
RS(64, 62)	128	BS(64, 62)	67	– 61	– 47.7 %
RS(128, 126)	256	BS(128, 126)	131	– 125	– 48.8 %
RS(255, 253)	510	BS(255, 253)	258	– 252	– 49.4 %

Brand "X" ⁎⁎⁎⁎CORE IP Reed-Solomon Decoder v9.0		Blue-Salmon STORM-Series * Standard Data Rate PP Decoder (this version)		Blue-Salmon Benefits vs. Brand "X"	
Configuration (examples)	Latency (cycles)	Configuration (examples)	Latency (cycles)	Δ Latency (cycles)	Δ Latency (ratio)
RS(8, 6)	40	BS(8, 6)	11	– 29	– 72.5 %
RS(16, 14)	42	BS(16, 14)	19	– 23	– 54.8 %
RS(32, 30)	58	BS(32, 30)	35	– 23	– 39.7 %
RS(64, 62)	90	BS(64, 62)	67	– 23	– 25.6 %
RS(128, 126)	154	BS(128, 126)	131	– 23	– 14.9 %
RS(255, 253)	281	BS(255, 253)	258	– 23	– 8.2 %

* Note: The STORM-Series PP decoders offer basic performance.
Other Series of Blue-Salmon decoders offer ***much higher*** levels of performance.

Data rate comparison

In the following tables the Blue-Salmon decoder is compared with two top performance highly optimized cores of another error correction system:

Brand "M" Core***** Reed-Solomon Decoder v3.1			Blue-Salmon STORM-Series * Standard Data Rate PP Decoder (this version)			Blue-Salmon Benefits vs. Brand "M"	
Config. (examples)	Pipe -line Stall (cyc.)	Data Rate (symbol /cyc.)	Config. (examples)	Pipe -line Stall (cyc.)	Data Rate (symbol /cyc.)	Δ Pipe -line Stall (cycles)	Δ Data Rate (ratio)
RS(8, 6)	9	0.47	BS(8, 6)	0	1	– 9	+ 112.5 %
RS(12, 10)	5	0.71	BS(12, 10)	0	1	– 5	+ 41.7 %
RS(16, 14)	1	0.94	BS(16, 14)	0	1	– 1	+ 6.3 %
RS(32, 30)	0	1	BS(32, 30)	0	1	0	+ 0 %

Brand "X" ****CORE IP Reed-Solomon Decoder v9.0			Blue-Salmon STORM-Series * Standard Data Rate PP Decoder (this version)			Blue-Salmon Benefits vs. Brand "X"	
Config. (examples)	Pipe -line Stall (cyc.)	Data Rate (symbol /cyc.)	Config. (examples)	Pipe -line Stall (cyc.)	Data Rate (symbol /cyc.)	Δ Pipe -line Stall (cycles)	Δ Data Rate (ratio)
RS(8, 6)	6	0.57	BS(8, 6)	0	1	– 6	+ 75.0 %
RS(12, 10)	2	0.86	BS(12, 10)	0	1	– 2	+ 16.7 %
RS(16, 14)	0	1	BS(16, 14)	0	1	0	+ 0 %
RS(32, 30)	0	1	BS(32, 30)	0	1	0	+ 0 %

* Note: The STORM-Series PP decoders offer basic performance.
Other Series of Blue-Salmon decoders offer ***much higher*** levels of performance.

3. Interface Descriptions

Ports

The port signals for the STORM-Series PP decoder are defined as follows:

Signal	Direction	Description
clk	Input	Clock
clken	Input	Clock enable
clr_n	Input	Clear, active low
arst_n	Input	Asynchronous reset, active low
data_in[7:0]	Input	Data input
aux_in[NAB-1:0]	Input	Auxiliary input
val_in	Input	Valid data/aux, input side
sync_in	Input	Synchronization of encoded block, input side
data_out[7:0]	Output	Data output
aux_out[NAB-1:0]	Output	Auxiliary output
val_out	Output	Valid data/aux, output side (all symbols)
val_r1_out	Output	Valid data/aux, output side (data symbols only)
val_r0_out	Output	Valid data/aux, output side (parity symbols only)
sync_out	Output	Synchronization of decoded block, output side
err_pos_out	Output	Position of error symbol, output side
err_val_out	Output	Valid error status, output side
err_stat_out[1:0]	Output	Error status, output side

Signal Functionality

clk - input

Clock. Rising edge clock signal.

clken - input

Clock enable. When this signal is at level ‘0’ the core is frozen. In this state all inputs (except **arst_n**) are ignored and the core retains its current state. Connect to ‘1’ if not used.

clr_n - input

Clear, active low. Synchronous signal. Reset all registers to their initial state. Connect to ‘1’ if not used.

arst_n - input

Asynchronous reset, active low. Reset all registers to their initial state. Connect to ‘1’ if not used.

data_in[7:0] - input

Data input. Data to be decoded.

aux_in[NAB-1:0] - input

Auxiliary input. This input is used to pass information through the BS decoder with exactly the same latency as **data_in**. This could be used to tag each symbol with marker bits, for example. The number of **aux_in** bits (NAB) is parameterizable. Connect to ‘0’ if not used.

val_in - input

Valid data/aux, input side. This signal is used to inform the BS decoder that valid data and auxiliary information is present at the **data_in** and **aux_in** inputs.

sync_in - input

Synchronization of encoded block, input side. This input can be used to inform the BS decoder that the symbol present at the **data_in** is the first symbol of an encoded block. This signal is not required for regular operation, however it can be used to resynchronize the data flow. When this signal is set to ‘1’ during the first symbol of an encoded block nothing happens. On the contrary, when this signal is set to ‘1’ in the middle of an encoded block the BS decoder restarts the coding operations from this point. Connect to ‘0’ if not used.

data_out[7:0] - output

Data output. Decoded data.

aux_out[NAB-1:0] - output

Auxiliary output. Undecoded auxiliary information. Leave unconnected if not used.

val_out - output

Valid data/aux, output side (all symbols). This signal is used to inform that the BS decoder has valid data and auxiliary information present at the **data_out** and **aux_out** outputs. This signal is set to '1' for each data and parity symbol leaving the decoder.

val_r1_out - output

Valid data/aux, output side (data symbols only). This signal is used to inform that the BS decoder has valid data and auxiliary information present at the **data_out** and **aux_out** outputs. This signal is set to '1' only for each data symbol leaving the decoder.

val_r0_out - output

Valid data/aux, output side (parity symbols only). This signal is used to inform that the BS decoder has valid data and auxiliary information present at the **data_out** and **aux_out** outputs. This signal is set to '1' only for each parity symbol leaving the decoder.

sync_out - output

Synchronization of decoded block, output side. This signal is used to inform that the symbol present at the **data_out** is the first symbol of a decoded block. Leave unconnected if not used.

err_pos_out - output

Position of error symbol, output side. This signal is used to inform that the symbol present at the **data_out** has been corrected. Leave unconnected if not used.

err_val_out - output

Valid error status, output side. This signal is used to inform that the BS decoder has valid error status present at the **err_stat_out** output. This signal is set to '1' when the last symbol of the encoded block (the second parity symbol) is leaving the decoder.

err_stat_out[1:0] - output

Error status, output side. This status is encoded in the following manner:

- "00" = no error
- "01" = a data symbol has been corrected
- "10" = a parity symbol has been corrected
- "11" = the correcting capacity is exceeded and no correction has been made

With the Blue-Salmon codes, if the error correcting capacity of the code is exceeded, it is usually possible to detect this. However, there might be some cases where it is impossible. This is a function of the codes themselves and not the decoder implementation. It is important to note that the behavior of the Blue-Salmon decoders is the exactly the same as the Reed-Solomon ones.

If the excess of correction capacity is detected, the **err_stat_out** output is set to "11" and the symbol values on **data_out** are left unchanged.

If the excess of correction capacity is not detected, the **err_stat_out** output is set either to "01" or "10" and the symbol values on **data_out** are undefined.

Configuration Parameters

The configuration parameters for the STORM-Series PP decoder are defined as follows:

Name	Range	Description
SBS	3 to 255	Size of BS encoded block (number of symbols).
NAB	1 to 32	Number of **aux_in** and **aux_out** bits.

Core Characteristics

The Blue-Salmon decoder core has the following characteristics:

Name	Value	Description
Latency	SBS + 3 clock cycles	Delay between the data input and output.
Data rate	1 symbol per clock cycle	Continuous data rate through the decoder.

4. Product Support

Website

You can browse a variety of technical and non-technical information about the Blue-Salmon Error Correction System at:

bs.arrigonline.ch

5. List of Changes

Decoder Core

The following table lists the critical changes that were made in each revision of the decoder core:

Revision	Changes
v01.1 (August 2013)	First core version.

Documentation

The following table lists the critical changes that were made in each revision of the documentation:

Revision	Changes
v01.1a (August 2013)	Preliminary Data Sheet.

The information disclosed to you is provided solely for the selection and use of Blue-Salmon products. You may not reproduce, modify, distribute, or publicly display the information without prior written consent. Blue-Salmon products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Blue-Salmon products in critical applications.

© Copyright 2013 Pierre Arrigo — bs.arrigonline.ch