



### Features

- $n = 65,536$  or  $16,384$ ,  $k = 239$ ,  $m = 1$ -bit symbols
- Capable of correcting (12, 10 or 8) 1-bit symbols
- Support for shortened frames
- 2 clock cycle latency
- Simple handshake protocol for reliable interfacing
- Fully synchronous design
- Very high speed operation
- Comprehensive verification plan provided

### General Description

The SALxx552E consists of verilog IP for implementing the  $t$ -error-correcting BCH forward error correction encoder, where  $t = 12, 10, \text{ or } 8$ .

The device treats incoming data as coefficients of a message polynomial over the field  $GF(2)$  generated by the product of minimal polynomials over consecutive powers of a field element belonging to  $GF(2^{16})$  or  $GF(2^{14})$

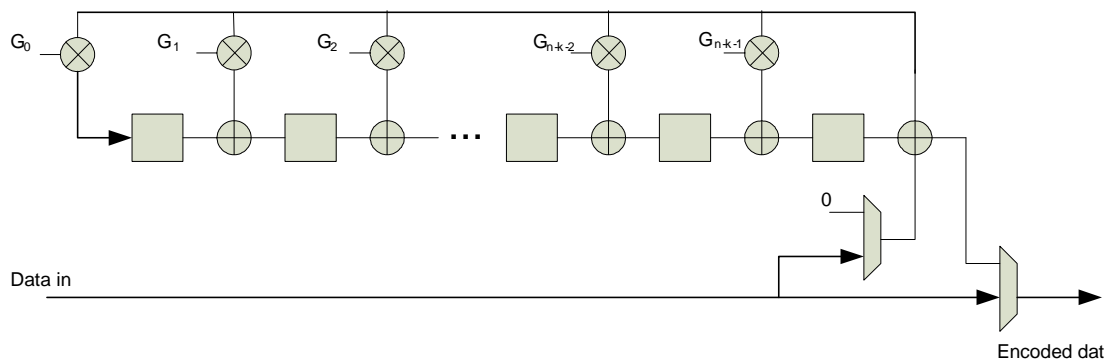


Figure 1: Encoder Block Diagram

## Theory of Operation

The SALxx552d decoder device is capable of decoding BCH codewords of maximum length 64,536 or 16,384. Shortened codewords are supported using the code\_len signal input to the device. There are three possible values for the error correcting capability 't', namely 12, 10, and 8.

The code generator polynomial is determined internally as the product of the first 't' polynomials in table 1, for the 64,800 length code, or the first 't' polynomials in table2, for the 16,200 length code.

$g_1(x)$	$1 + x^2 + x^3 + x^5 + x^{16}$
$g_2(x)$	$1 + x + x^4 + x^5 + x^6 + x^8 + x^{16}$
$g_3(x)$	$1 + x^2 + x^3 + x^4 + x^5 + x^7 + x^8 + x^9 + x^{10} + x^{11} + x^{16}$
$g_4(x)$	$1 + x^2 + x^4 + x^6 + x^9 + x^{11} + x^{12} + x^{14} + x^{16}$
$g_5(x)$	$1 + x + x^2 + x^3 + x^5 + x^8 + x^9 + x^{10} + x^{11} + x^{12} + x^{16}$
$g_6(x)$	$1 + x^2 + x^4 + x^5 + x^7 + x^8 + x^9 + x^{10} + x^{12} + x^{13} + x^{14} + x^{15} + x^{16}$
$g_7(x)$	$1 + x^2 + x^5 + x^6 + x^8 + x^9 + x^{10} + x^{11} + x^{13} + x^{15} + x^{16}$
$g_8(x)$	$1 + x + x^2 + x^5 + x^6 + x^8 + x^9 + x^{12} + x^{13} + x^{14} + x^{16}$
$g_9(x)$	$1 + x^5 + x^7 + x^9 + x^{10} + x^{11} + x^{16}$
$g_{10}(x)$	$1 + x + x^2 + x^5 + x^7 + x^8 + x^{10} + x^{12} + x^{13} + x^{14} + x^{16}$
$g_{11}(x)$	$1 + x^2 + x^3 + x^5 + x^9 + x^{11} + x^{12} + x^{13} + x^{16}$
$g_{12}(x)$	$1 + x + x^3 + x^6 + x^7 + x^9 + x^{11} + x^{12} + x^{16}$

Table 1: BCH Polynomials for n = 64,800

$g_1(x)$	$1 + x + x^3 + x^5 + x^{14}$
$g_2(x)$	$1 + x^6 + x^8 + x^{11} + x^{14}$
$g_3(x)$	$1 + x + x^2 + x^6 + x^9 + x^{10} + x^{14}$
$g_4(x)$	$1 + x^4 + x^7 + x^8 + x^{10} + x^{12} + x^{14}$
$g_5(x)$	$1 + x^2 + x^4 + x^6 + x^8 + x^9 + x^{11} + x^{13} + x^{14}$
$g_6(x)$	$1 + x^3 + x^7 + x^8 + x^9 + x^{13} + x^{14}$
$g_7(x)$	$1 + x^2 + x^5 + x^6 + x^7 + x^{10} + x^{11} + x^{13} + x^{14}$
$g_8(x)$	$1 + x^5 + x^8 + x^9 + x^{10} + x^{11} + x^{14}$
$g_9(x)$	$1 + x + x^2 + x^3 + x^9 + x^{10} + x^{14}$
$g_{10}(x)$	$1 + x^3 + x^6 + x^9 + x^{11} + x^{12} + x^{14}$
$g_{11}(x)$	$1 + x^4 + x^{11} + x^{12} + x^{14}$
$g_{12}(x)$	$1 + x + x^2 + x^3 + x^5 + x^6 + x^7 + x^8 + x^{10} + x^{13} + x^{14}$

Table 2: BCH Polynomials for n = 16,200

## Signal Descriptions

The module pinout is shown in the figure below, and in table 1. The signals are conveniently organized into functional groups as follows:

### Clock and Reset

The design is fully synchronous with a single clock signal. The reset signal is synchronous and needs to be asserted for at least one full clock cycle to reset internal logic.

### Control signals

Two signals control flow of data into the device, *din\_rdyin\_n* and *in\_dp\_n*. The *din\_rdyin\_n* signal indicates that data into the device is valid. The *in\_dp\_n* signal indicates that message bits are being shifted into the device.

Two signals control flow of data out of the device, *dout\_rdyout\_n* and *dout\_rdyin\_n*. The *dout\_rdyout\_n* signal indicates that data out of the device is valid.

### Data signals

The data are clocked in on *din* and clocked out on *dout*.

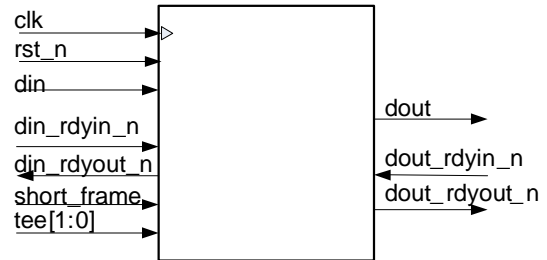


Figure 2: Component pinout

Pin	Sense	Width	Description
clk	in	1	Clock
rst_n	in	1	Synchronous reset
din	in	1	Serial data (message) in
din_rdyin_n	in	1	Indicates serial data in is valid
din_rdyout_n	out	1	Indicates it's OK to shift data in
in_dp_n	in	1	'1' indicates data in, '0' = ignore input (during parity phase)
dout	out	1	Data out
dout_rdyin_n	in	1	Indicates to the device that it's OK to shift data out
dout_rdyout_n	out	1	Indicates that data is available to be shifted out
short_frame	in	1	Indicates that frame size is 16,200, not 64,800
tee	in	1	Indicates the error correcting ability of the code

Table 1. Component pinout

## Waveforms

### Input

The input functional timing is shown below. *Din\_rdyin\_n* is used as an input data enable, *in\_dp\_n* is used to indicate when data (as opposed to parity) is being shifted into the device.

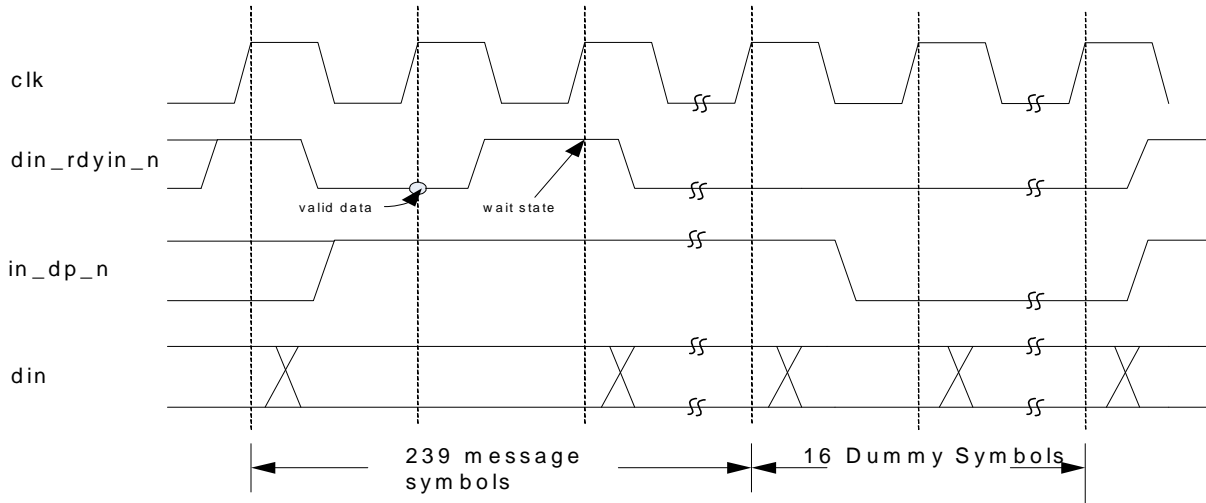


Figure 3: Input timing

### Output

The output functional timing is shown below. *Dout\_rdyout\_n* is used as an output data ready indication, *out\_dp\_n* is used to indicate when data (as opposed to parity) is being shifted out of the device.

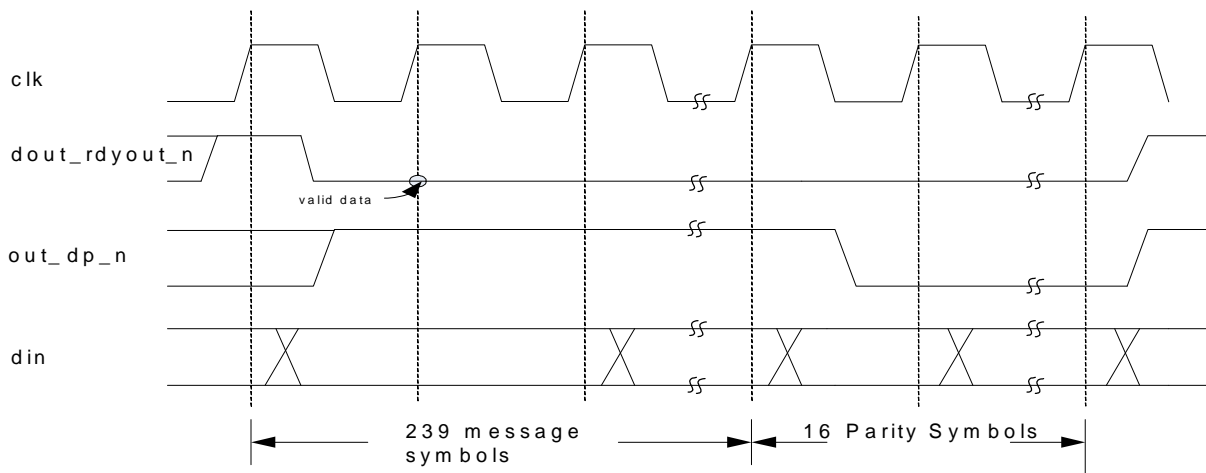


Figure 4. Output timing

## Module Verification

The SALxx552E has been subjected to extensive verification to ensure the highest quality product possible. A comprehensive test plan was implemented which included the following:

- High-quality random data source
- High-quality random noise source
- Extensive flow-control simulations
- Verification of operation against known data sequences

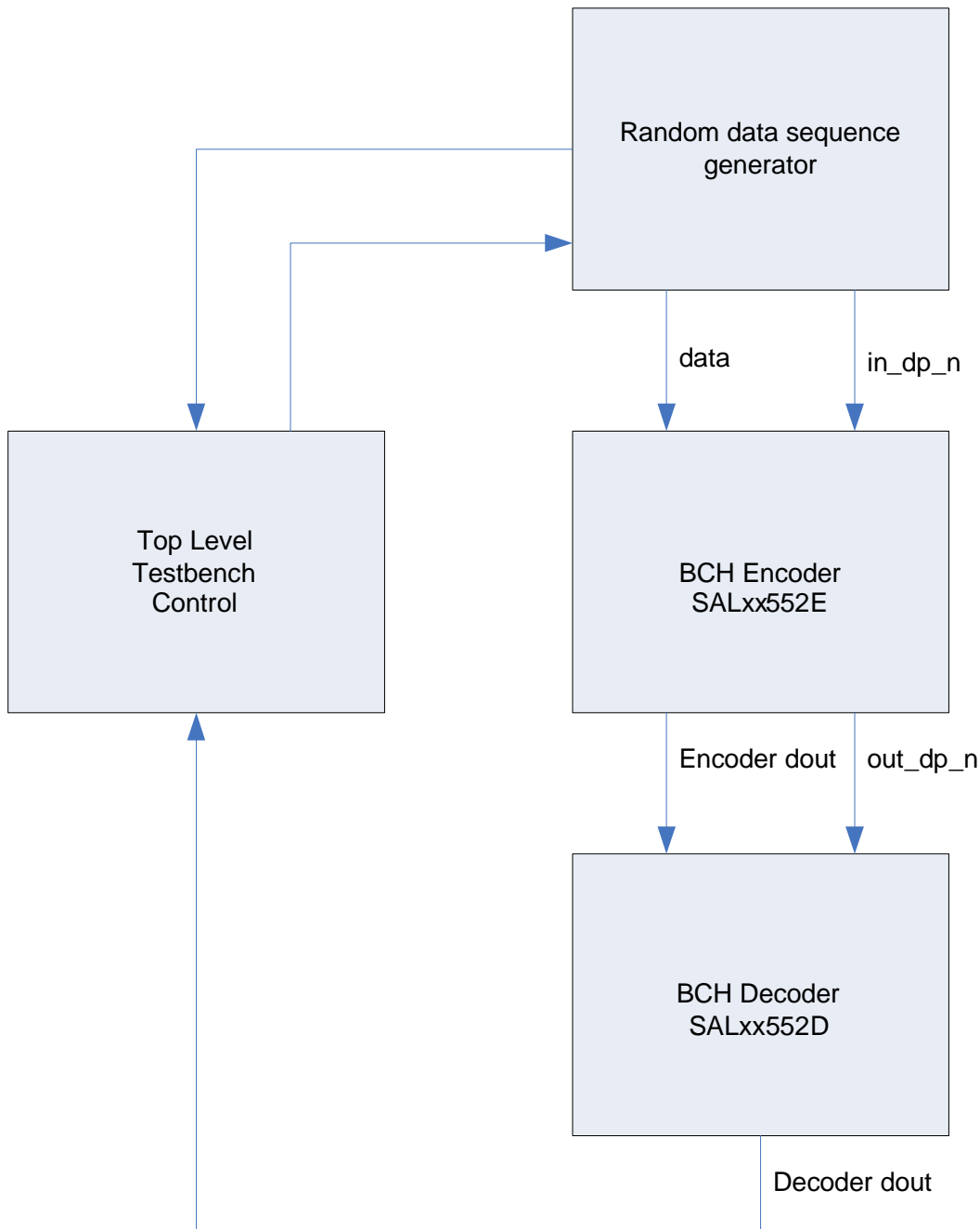
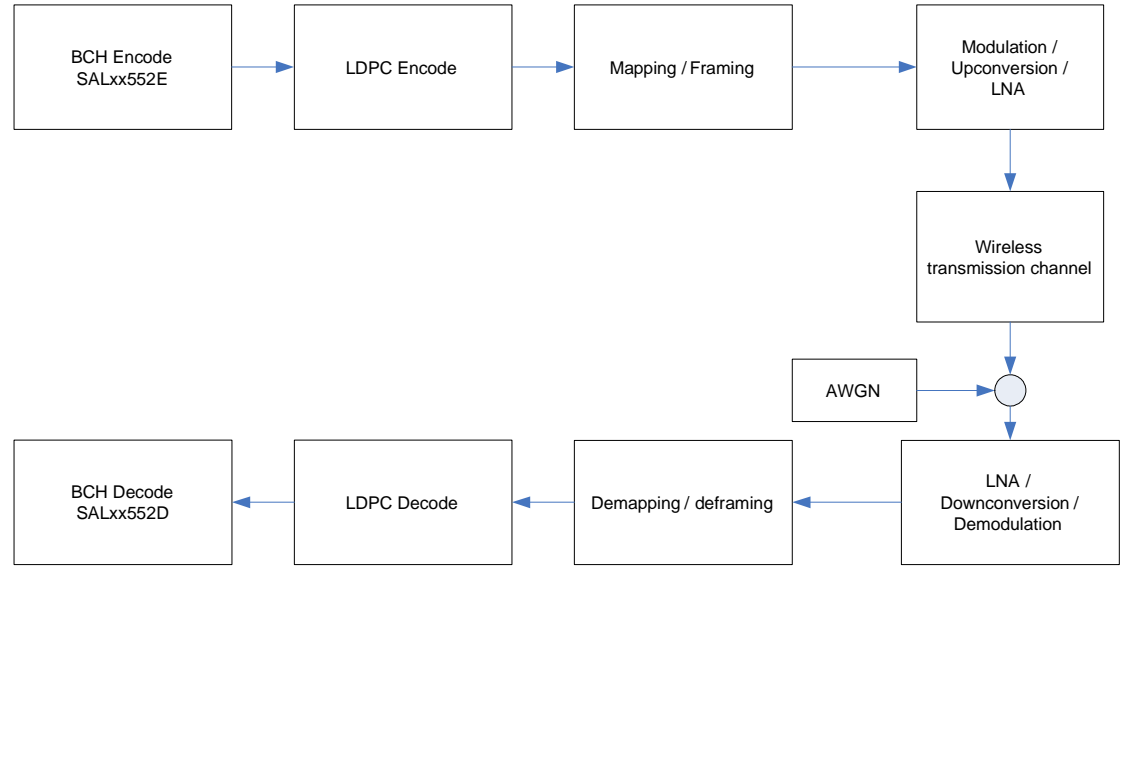


Figure 5: Testbench Block Diagram

## Application: DVB-S2 Satellite communications



## Ordering Information

**SALxx552E** DVB-S2 BCH Encoder

About Salamander:

**Salamander Error Correction** develops and sells error correction modules of the highest quality worldwide.

Salamander Error Correction is a division of Komodo Industries, Inc.

Salamander Error Correction:  
5330 Carroll Canyon Rd., ste 200-G  
San Diego, CA 92121  
Ph: (858) 373-2112  
Fax: (858) 373-2112  
sales@salamander-ecc.com