COM-1813SOFT
DVB-S2x LDPC encoder/decoder
VHDL source code overview / IP core

Overview

The COM-1813SOFT is a LDPC code error correction encoder/decoder compliant with DVB-S2 and S2x specifications. It is written in generic portable VHDL.

The entire **VHDL source code** is deliverable.

Key features and performance:

- Includes encoding, decoding, BER tester, PRBS11 test sequence generator.
- Compliant with ETSI EN 302 307-1 V1.4.1 (2014-11) (DVB-S2) and .
  ETSI EN 302 307-2 V1.1.1 (2015-02) (DVB-S2X)
- User-selected configuration:
  - code block lengths n_{ldpc}: 64800, 32400, 16200 bits
  - Code rates
  - VL-SNR puncturing and shortening
  - Maximum number of iterations
- Typical Bit Error Rate / Frame Error Rate for rate 1/2 n_{ldpc} = 16200:
  BER < 10^{-7} FER < 10^{-4} @ E_b/N_o = 0.9 dB
- Throughput:
  Encoding: up to 1.3 Gbits/s
  Decoding: 100 - 400 Mbits/s payload bits
- Trade-off device utilization vs throughput using parallel decoding:
  N_PAR_DEC = 45/90/180/360
- Two decoding algorithms: lambda-min for best BER performance or normalized min-sum for minimum device utilization
- Provided with IP core:
  - VHDL source code
  - Matlab .m file for simulating the encoding and decoding algorithms, for generating stimulus files for VHDL simulation and for end-to-end BER/FER performance analysis at various signal-to-noise ratios
  - VHDL testbench

Portable VHDL code

The code is written in generic standard VHDL and is thus portable to a variety of FPGAs. The code was developed and tested on a Xilinx 7-series FPGA but is expected to work similarly on other targets. No manufacturer-specific primitive is used.

Implemented codes

Normal frame n_{ldpc} = 64800 LDPC code identifiers:
1/4, 1/3, 2/5, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9, 9/10
Short frame n_{ldpc} = 16200 LDPC code identifiers:
1/4 or 1/5, 1/3, 2/5, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9, 11/45, 4/15, 14/45, 7/15, 8/15, 26/45, 32/45 VL-SNR: 1/5 SF2, 11/45 SF2, 1/5, 11/45, 1/3
Medium frame n_{ldpc} = 32400 LDPC code identifiers:
VL-SNR: 1/5, 11/45, 1/3

Comblock LLC • 845-N Quince Orchard Boulevard • Gaithersburg, Maryland 20878 • U.S.A.
Telephone: (240) 631-1111 www.ComBlock.com
© ComBlock LLC 2026 Issued 2/15/2026

Configuration

Synthesis-time configuration parameters

The following constants are user-defined in the decoder component generic section prior to synthesis. These parameters generally define the size of the decoder embodiment.

Synthesis-time configuration parameters	
Encoder / Decoder	
FRAME_TYPE_EN(2:0)	Instantiate the resources necessary to implement various frame lengths bit 0 = DVB-S2 normal frame, bit 1 = DVB-S2 and DVB-S2x short frame bit 2 = DVB-S2x medium frame
Decoder	
Number of soft-quantized bits at the decoder input **IN_NBITS**	LLR input precision: typically 4 or 5. A minor performance improvement can be achieved with 5-bits. Although **IN_NBITS** can be programmed in the range 4 to 8 soft-quantized bits, the decoder performance does not improve beyond 5 bits.
Decoder maximum number of iterations **N_ITER_MAX**	The higher the number of iterations, the better the error correction performance. Not much improvement above 50. The decoder stops the iterative process as soon as all parity checks are verified, or when it reaches **N_ITER_MAX** iterations, whichever happens first.
Number of parallel decoders **N_PAR_DEC**	number of parallel check node processors, valid values: 45, 90, 180, 360 Trade-off device utilization vs throughput

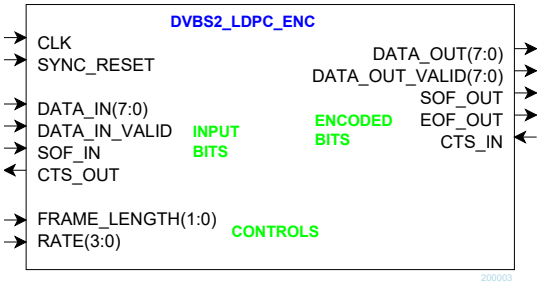
I/Os

General

CLK: input
The synchronous clock. The user must provide a global clock (use BUFG). The CLK timing period must be constrained in the .xdc file associated with the project.

SYNC_RESET: input
Synchronous reset. The reset MUST be exercised at least once to initialize the internal variables. It must be exercised whenever a control parameter is changed.

Encoder



FRAME_TYPE(1:0):
LDPC coded block n_{ldpc}:
0 = normal frame, n_{ldpc} = 64800 bits
1 = short frame, n_{ldpc} = 16200 bits
2 = medium frame, n_{ldpc} = 32400 bits

Must be defined 2 clocks before the first input Byte and frozen throughout a frame encoding.

RATE(4:0): coding rate:
0: rate 1/4 or 1/5 (normal, short, medium frames)
1: rate 1/3 (normal, short, medium frames)
2: rate 2/5 (normal, short frames)
3: rate 1/2 (normal, short frames)
4: rate 3/5 (normal, short frames)
5: rate 2/3 (normal, short frames)
6: rate 3/4 (normal, short frames)
7: rate 4/5 (normal, short frames)
8: rate 5/6 (normal, short frames)
9: rate 8/9 (normal, short frames)
10: rate 9/10 ((normal frames))
11: rate 11/45 (normal, short, medium frames)
12: rate 4/15 (short frames)
13: rate 14/45 (short frames)
14: rate 7/15 (short frames)

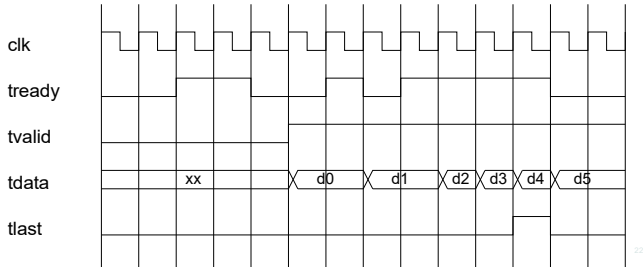
15: rate 8/15 (short frames)
16: rate 26/45 (short frames)
17: rate 32/45 (short frames)

Must be defined 2 clocks before the first input Byte and frozen throughout a frame encoding.

VLSNR_EN: Very-Low SNR frame enable(1)/disable(0). VL-SNR shortening and puncturing.

VLSNR_SF2: SF2 frame enable(1)/disable(0) to distinguish between LDPC code identifiers 1/5 and 1/5SF2. See [2] table 19d.

The encoder input follows the AXI4-stream interface definition:



DATA_IN(7:0): Input data is read one Byte at a time. Bits are packed LSb first. Always a full Byte, no partial Byte allowed. DATA_IN is read when both **DATA_IN_VALID** and **DATA_IN_READY** are '1'.

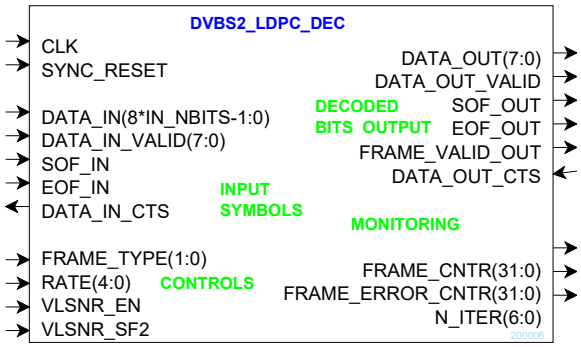
DATA_IN_VALID: input.
1 CLK-wide pulse indicating that DATA_IN is valid.

DATA_IN_LAST: optional last input Byte in a Frame. 1 CLK-wide pulse.

DATA_IN_READY: output.
Clear-**T**o-**S**end flow control. '1' indicates that the encoder is ready to accept another input byte. The encoder stops requesting input data when the input elastic buffer is 3/4 full.

The encoder outputs mirror its inputs:
DATA_OUT(7:0), **DATA_OUT_VALID(7:0)**, **SOF_OUT**, **EOF_OUT**, **CTS_IN**.

Decoder



DATA_IN(8*IN_NBITS-1:0): eight soft-quantized input symbols. Each symbol represents a Log-Likelihood Radio (LLR):
log(Pr(x = 1)/Pr(x = 0))
The LLR precision (**IN_NBITS**) is selectable at the time of synthesis. Typical values are 4- or 5-bit soft-quantization. The soft-quantized input symbols are expected to be symmetrical around zero, for example ranging from -7 to +7 or -15 to +15 although this rule is enforced within.

Convention: throughout the code, a positive symbol represents a '1', negative a '0'. The eight symbols are packed LSb first.

DATA_IN_VALID(7:0): 1 CLK-wide Byte indicating that **DATA_IN** is valid and how many input samples are supplied. xFF when coded input consists of 8 samples. The last input word at the end of frame sometimes consists of 4 or 6 samples (when configured as VL_SNR). In this case, **DATA_IN_VALID** is xF0 or FC respectively.

SOF_IN / EOF_IN: inputs Start Of Frame and End Of Frame. 1 CLK-wide pulses.
A aligned with **DATA_IN_VALID**.
Each frame consists of 16200,32400 or 64800 symbols entered 8 at a time.

DATA_IN_CTS: output **C**lear-**T**o-**S**end flow control. '1' indicates that the decoder is ready to accept another group of 8 parallel input symbols.

The decoder outputs mirror its inputs:
DATA_OUT(7:0), **DATA_OUT_VALID**, **SOF_OUT**, **EOF_OUT, DATA_OUT_CTS**.
The decoded bit stream **DATA_OUT** is sent out one Byte at a time. Bits are packed LSb first. **FRAME_VALID_OUT** indicates whether the

output frame was successfully decoded (‘1’ when it passed all checknode verifications) or not (i.e. decoding reached the maximum number of iterations).

Performance

Encoder information throughput

The maximum encoder (information) input rate depends on the codeword length (n_{ldpc}), the encoding Rate and the processing clock frequency.

n_{ldpc} \ coding rate	1/2	2/3	4/5	8/9
64800-bit codeword	12920 clocks / frame 714 Mbits/s @ 285 MHz	11690 clocks / frame 1053 Mbits/s @ 285 MHz	10778 clocks / frame 1370 Mbits/s @ 285 MHz	10027 clocks / frame 1637 Mbits/s @ 285 MHz
16200-bit codeword	3345 clocks / frame 613 Mbits/s @ 285 MHz	2960 clocks / frame 1039 Mbits/s @ 285 MHz	2724 clocks / frame 1318 Mbits/s @ 285 MHz	2530 clocks / frame 1622 Mbits/s @ 285 MHz

Decoder iteration time

Each decoding iteration takes TS_n+15 clocks, where TS_n is the number of TimeSlots (clocks) needed to process all CheckNodes during one decoding iteration

Decoder configuration	Ts_n short frame		TS_n normal frame		TS_n medium frame	
	45 parallel decoders	360 parallel decoders	45 parallel decoders	360 parallel decoders	45 parallel decoders	360 parallel decoders
rate 1/4	1099	218	4321	559	2305	320
rate 1/3	1209	255	4801	612	2401	299
rate 2/5	1297	284	5185	663		
rate 1/2	1103	177	5041	631		
rate 3/5	1587	407	6337	842		
rate 2/3	1204	297	4801	639		
rate 3/4	1073	234	5041	683		
rate 4/5	1020	168	5185	740		
rate 5/6	1117	232	5281	882		
rate 8/9	1107	188	4334	684		
rate 9/10	N/A	N/A	4334	671		
rate 11/45 VL-SNR SF2	1089	204			2182	331
rate 4/15 VL-SNR	1322	528				
rate 14/45	1262	291				
rate 7/15	1668	509				
rate 8/15	1704	471				
rate 26/45	1540	412				
rate 32/45	1353	326				

In addition, the decoder input frame takes $n_{ldpc}/8$ clocks, and the decoder output k_{ldpc} *8 clocks.*
Thus the total number of clocks for decoding a given frame is
(number of decoding iterations)* TS_n + (n_{ldpc} * (1 + coding rate)/8)

The minimum number of iterations is one.

Decoder average number of iterations vs Eb/No

The average number of iterations affects the overall decoder throughput. It is a function of n_{ldpc}, rate, and the threshold operating Eb/No.

For n_{ldpc} = 16200 (short frame), BER < 10^{-7}, Lambda-min algorithm

Decoder configuration	Eb/No (dB)	Average number of decoding iterations
rate 1/4	0.4	27.9
rate 1/3	0.5	21.7
rate 2/5	0.8	18.7
rate 1/2	1.0	19.2
rate 3/5	1.8	10.2
rate 2/3	2.0	11.6
rate 3/4	2.5	10.1
rate 4/5	2.8	10.3
rate 5/6	3.3	8.0
rate 8/9	3.9	8.6

For n_{ldpc} = 64800 (normal frame), BER < 10^{-7}, Lambda-min algorithm

Decoder configuration	Eb/No (dB)	Average number of decoding iterations
rate 1/4	0.3	32.5
rate 1/3	0.4	26.5
rate 2/5	0.7	23.0
rate 1/2	1.0	19.4
rate 3/5	1.4	18.5
rate 2/3	1.9	14.5
rate 3/4	2.4	12.4
rate 4/5	2.7	13.1
rate 5/6	3.0	13.2
rate 8/9	3.7	11.1
rate 9/10	3.9	7.4

Decoder throughput

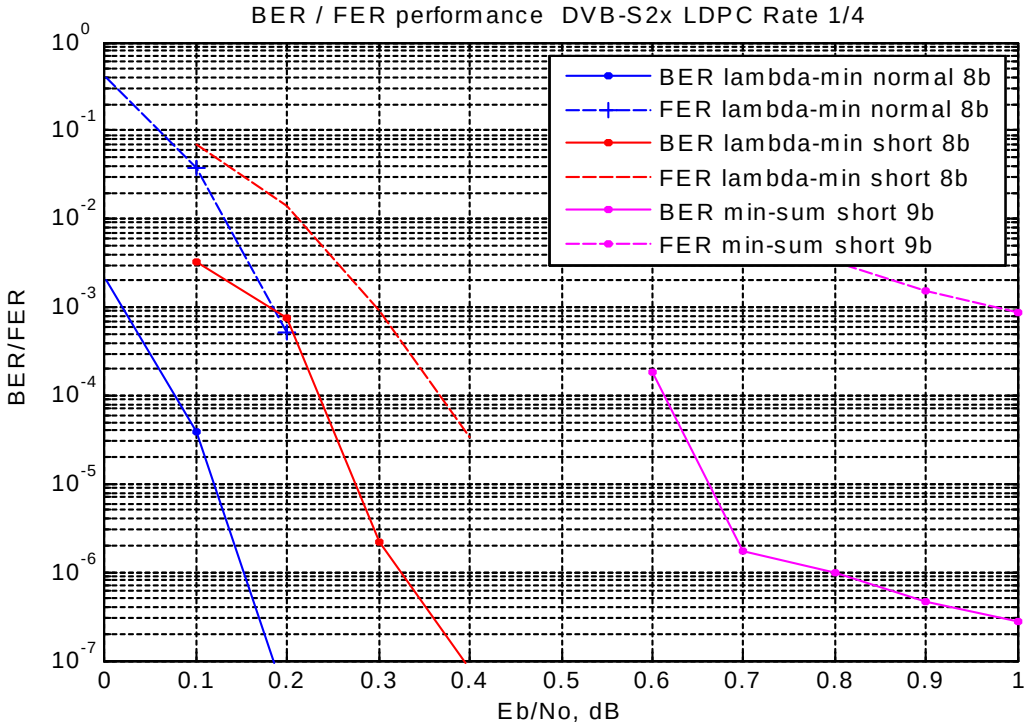
Throughput examples (at Eb/No threshold for 10^{-7} BER):

FPGA speed	Configuration	Average decoded throughput at threshold Eb/No (payload bits) N_PAR_DEC = 45
Zynq Ultrascale+ @250 MHz	n_{ldpc} =16200 rate 1/2	83 Mbits/s
Zynq Ultrascale+ @250 MHz	n_{ldpc} =16200 rate 8/9	286 Mbits/s
Zynq Ultrascale+ @220 MHz	n_{ldpc} = 64800 rate 1/2	64.8 Mbits/s

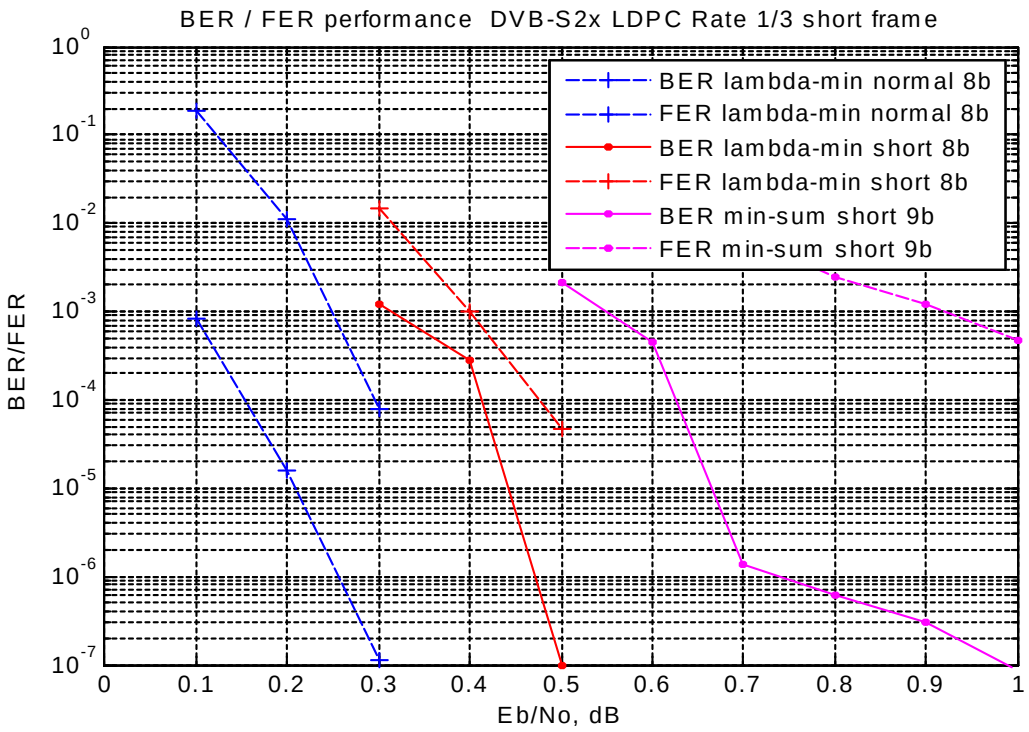
BER/ FER performance

The decoded errors are somewhat bursty in nature, with many error-free decoded frames followed by an occasional erroneous frame with multiple bit errors. Therefore, we also express the decoder performance in terms of frame error rate (FER). Note that the FER refers to a k_{ldpc}-bit frame whereas the DVB-S2 standard uses a PER measure which refers to a 188-Byte packet.

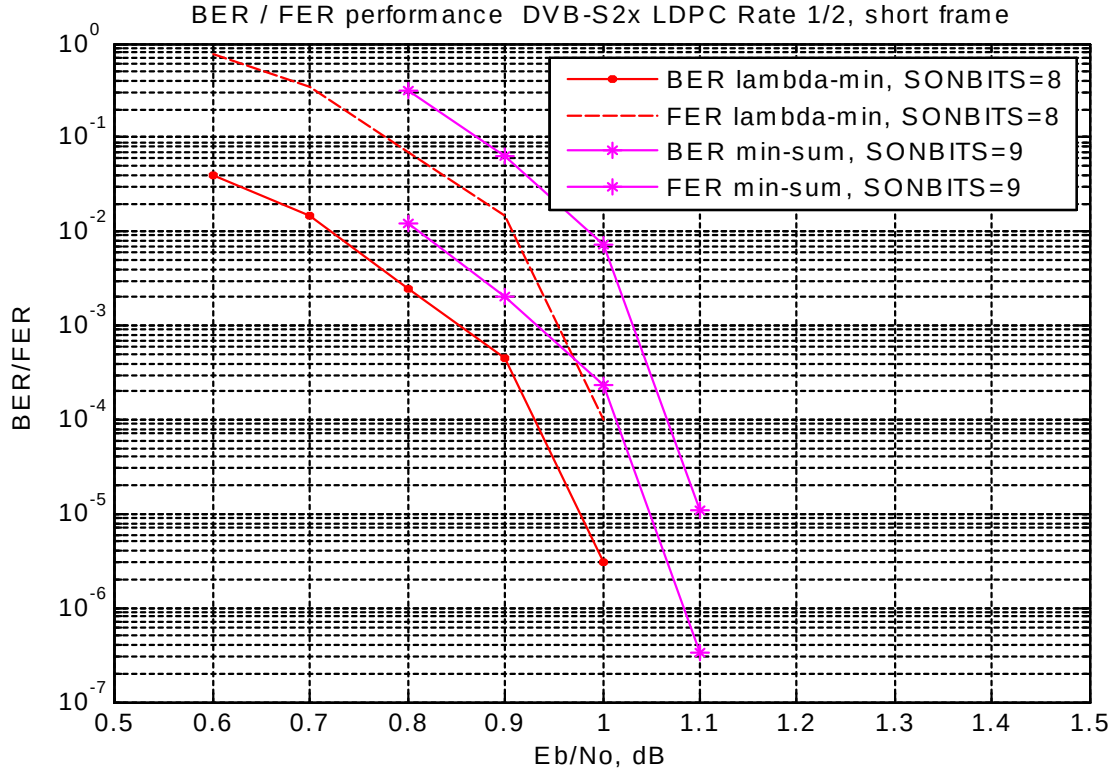
Test conditions: rate 1/4, 50 iterations, 5-bit soft-quantization



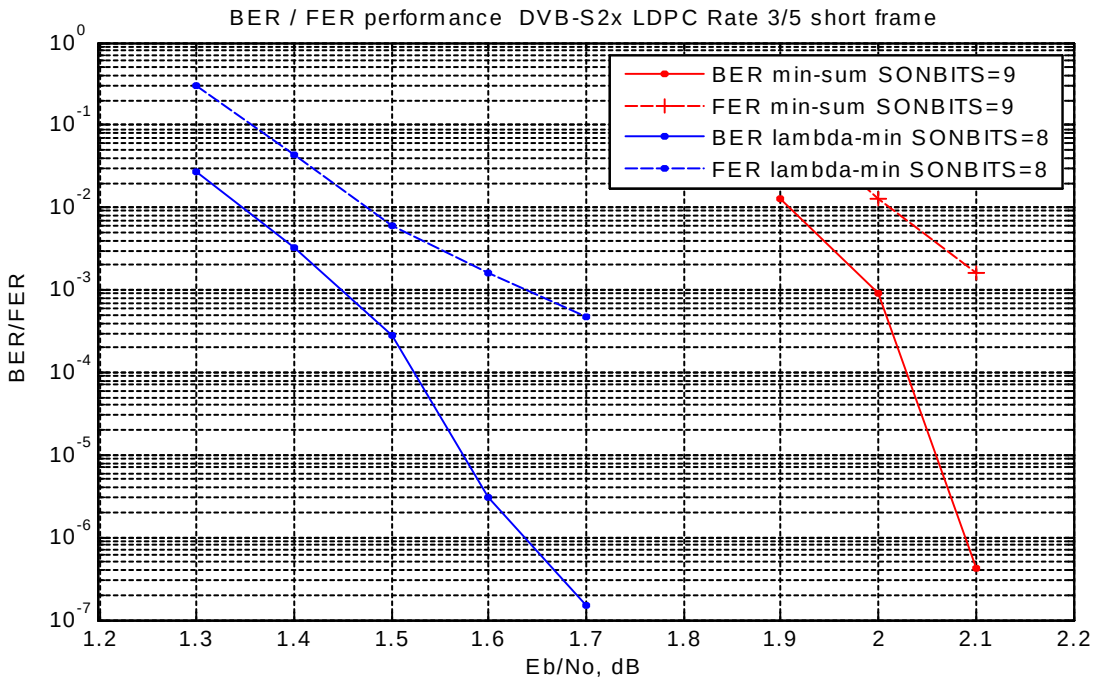
Test conditions: rate 1/3, 50 iterations, 5-bit soft-quantization



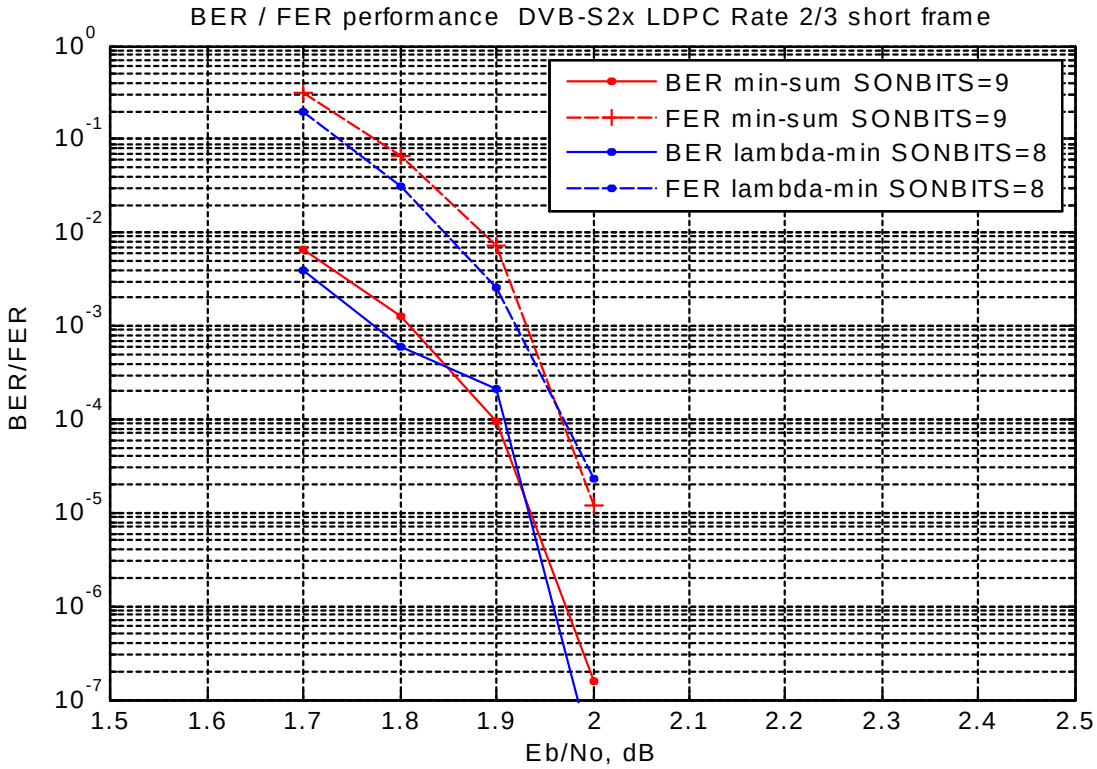
Test conditions: rate 1/2, 50 iterations, 5-bit soft-quantization
Comparing lamba-min and min-sum algorithms



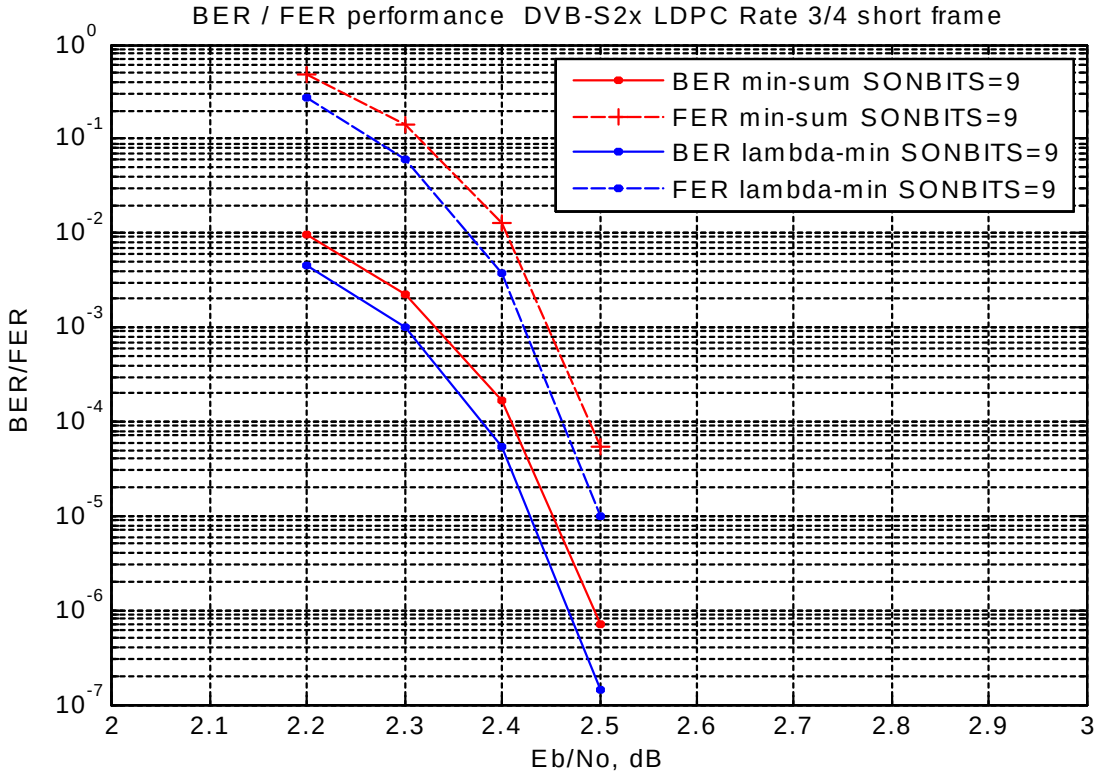
Test conditions: rate 3/5, 50 iterations, 5-bit soft-quantization


BER / FER performance DVB-S2x LDPC Rate 3/5 short frame
BER/FER
Eb/No, dB
BER min-sum SONBITS=9
FER min-sum SONBITS=9
BER lambda-min SONBITS=8
FER lambda-min SONBITS=8
10^0
10^-1
10^-2
10^-3
10^-4
10^-5
10^-6
10^-7
1.2
1.3
1.4
1.5
1.6
1.7
1.8
1.9
2
2.1
2.2

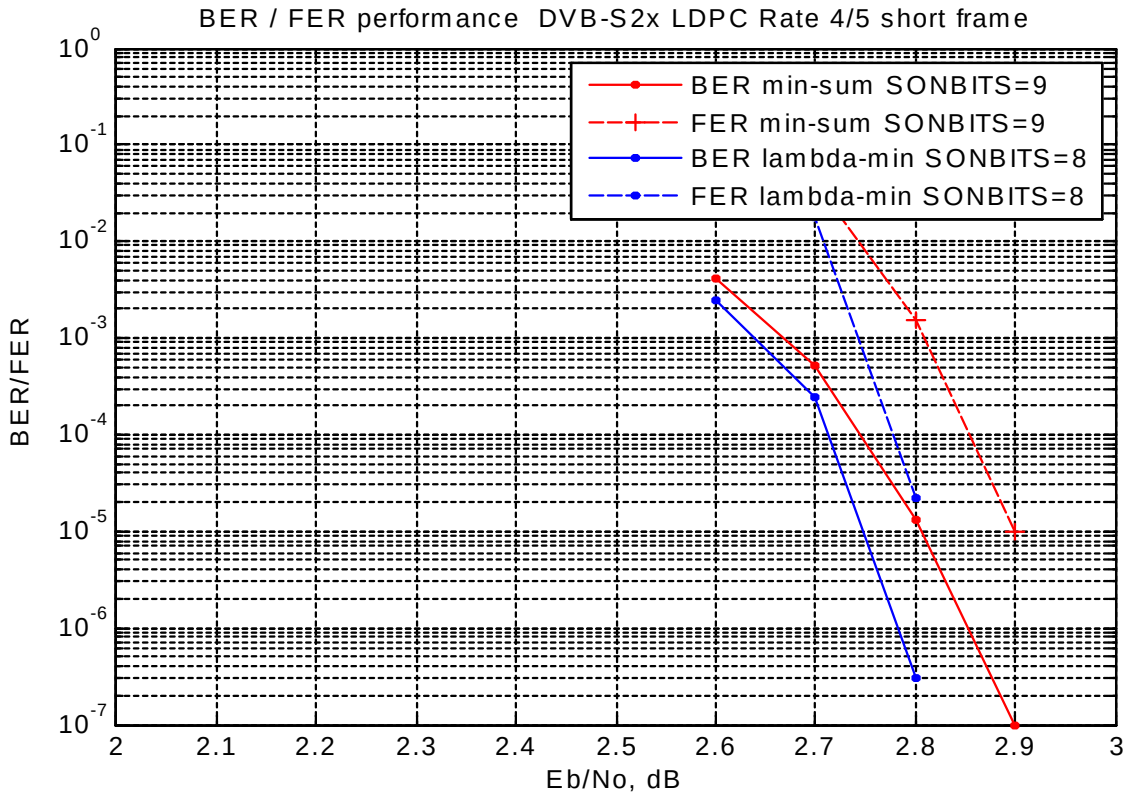
Test conditions: rate 2/3, 50 iterations, 5-bit soft-quantization



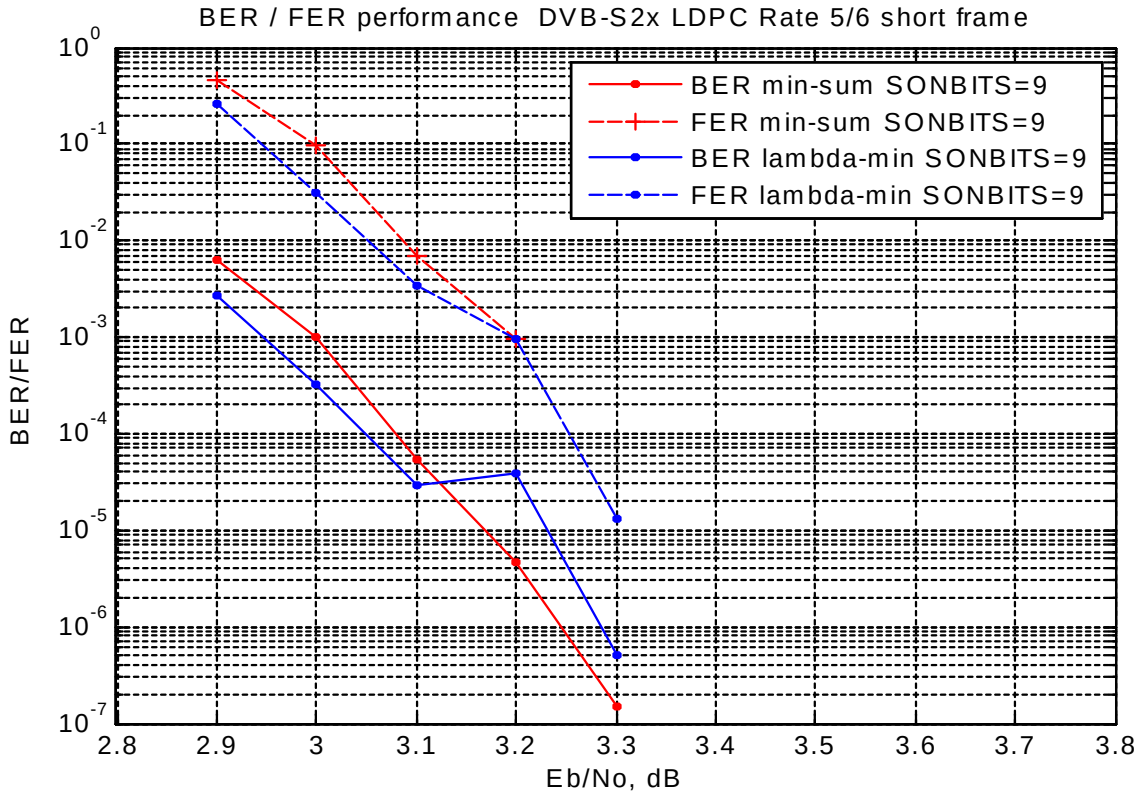
Test conditions: rate 3/4, 50 iterations, 5-bit soft-quantization, SO_NBITS=9



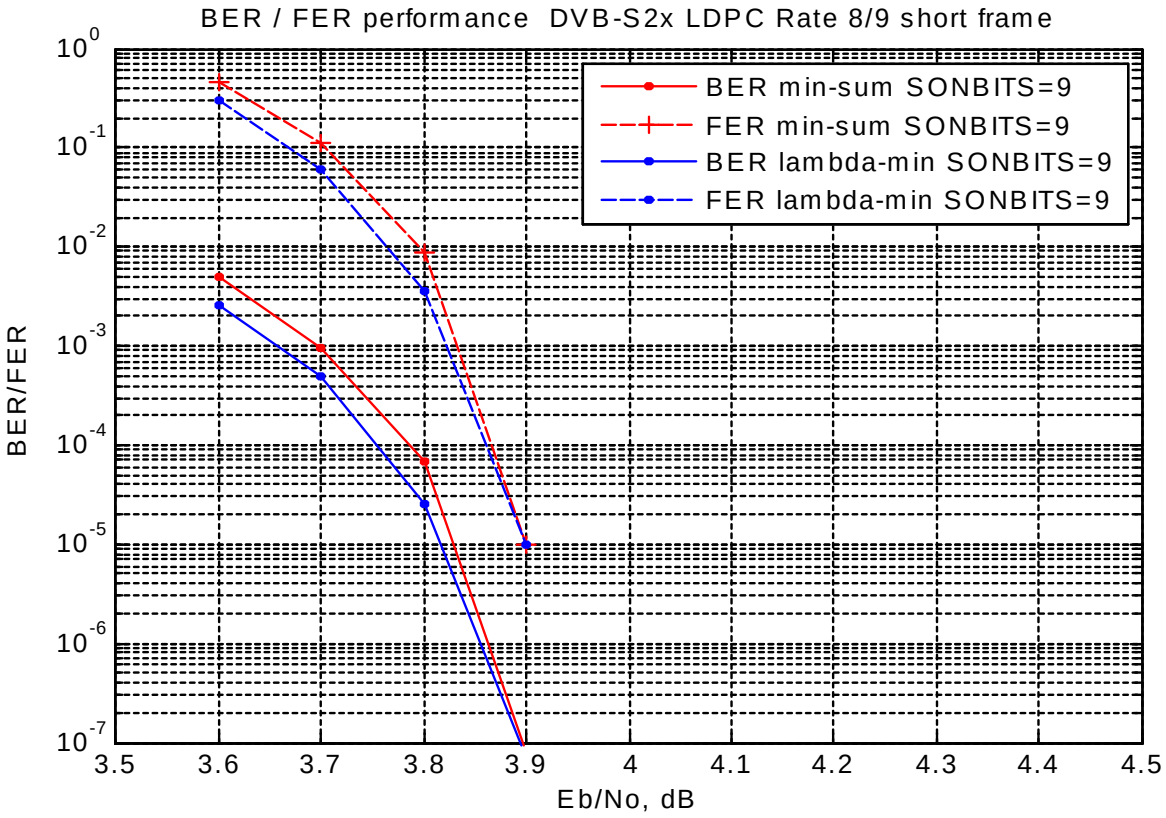
Test conditions: rate 4/5, 50 iterations, 5-bit soft-quantization



Test conditions: rate 5/6, 50 iterations, 5-bit soft-quantization



Test conditions: rate 8/9, 50 iterations, 5-bit soft-quantization



Computation precision

The computation precision (SO_NBITS in the *DVBS2_LDPC_DEC.vhd* component) affects the BER. We selected SO_NBITS = 8 bits as a good tradeoff between performance and device utilization. SO_NBITS = 9 could also be used but the performance improvement is negligible.

Software Licensing

The COM-1813SOFT is supplied under the following key licensing terms:

1. A nonexclusive, nontransferable license to use the VHDL source code internally, and
2. An unlimited, royalty-free, nonexclusive transferable license to make and use products incorporating the licensed materials, solely in bit stream format, on a worldwide basis.

The complete VHDL/IP Software License Agreement can be downloaded from http://www.comblock.com/download/softwarelicense.pdf

Configuration Management

The current software revision is 0.

Directory	Contents
/doc	Specifications, user manual, implementation documents
/src	.vhd source code,.pkg packages, .xdc constraint files (Xilinx) One component per file.
/sim	VHDL test benches
/matlab	Matlab .m file for simulating the encoding and decoding algorithms, for generating stimulus files for VHDL simulation and for end-to-end BER performance analysis at various signal to noise ratios

Project files:

Xilinx Vivado v2020 project file: project_1.xpr

More generally, a tcl file can help construct the complete project: /project_1/project_1v2020.tcl
It includes all the constituent components.

VHDL development environment

The VHDL software was developed using the following development environment:

Xilinx Vivado 2020 for synthesis, place and route and VHDL simulation

Reference documents

[1] ETSI EN 302 307-1 v1.4.1 DVB-S2 specifications, 2014-11
Applicable sections:
Section 5.3.2 Inner encoding (LDPC)

[2] ETSI EN 302 307-2 v1.1.1 DVB-S2X extensions specifications, 2015-02

[3] "Implementation of an LDPC decoder for the DVB-S2, -T2 and -C2

% standards",2010 Thesis, Cédric Marchand

[4] “Implementing the NASA Deep Space LDPC Codes for Defense Applications”, Zhao, Long, 2013.

[5] 'Efficient Implementations of the Sum-Product Algorithm for Decoding LDPC Codes",

Xiao-Yu Hu, Evangelos Eleftheriou, Dieter-Michael Arnold, and Ajay Dholakia, 2001

Device Utilization Summary

DVB-S2 LDPC encoder n_{ldpc} = 16200 short-frame only		Xilinx xczu7eg utilization
Registers	3417	0.7%
LUTs	5602	2.4%
Block RAM/FIFO 36Kb	0	0%
DSP48	0	0%
GCLKs	1	
DVB-S2 LDPC encoder normal, short, medium frames		
Registers	4944	1.1%
LUTs	6137	2.7%
Block RAM/FIFO 36Kb	5.5	1.8%
DSP48	0	0%
GCLKs	1	

DVB-S2 LDPC decoder
Lambda-min algorithm
n_{ldpc} = 64800 normal-frame only
IN_NBITS = 5, SO_NBITS = 8

N_PAR_DEC	**45**	**90**	**180**	**360**
LUTs	20278	36034	88744	176081
Registers	10172	18281	35580	69555
Block RAM/FIFO 36Kb	173	247	330	540
DSP48	0	0	0	0
GCLKs	1	1	1	1

DVB-S2 LDPC decoder
Lambda-min algorithm
n_{ldpc} = 16200 short-frame only
IN_NBITS = 5, SO_NBITS = 8

N_PAR_DEC	**45**	**90**	**180**	**360**
LUTs	18846	40751	75000	138506
Registers	10017	18640	35042	67666
Block RAM/FIFO 36Kb	86.5	97	120	173.5
DSP48	0	0	0	0
GCLKs	1	1	1	0

DVB-S2 LDPC decoder
Lambda-min algorithm
normal, short, medium frames
IN_NBITS = 5, SO_NBITS = 8

N_PAR_DEC	**45**	**90**	**180**	**360**
LUTs	20343	36859	88839	176267
Registers	10204	18295	35611	69571
Block RAM/FIFO 36Kb	205	247	330	540
DSP48	0	0	0	0
GCLKs	1	1	1	1

DVB-S2 LDPC decoder
Normalized min-sum algorithm
n_{ldpc} = 64800 normal-frame only
IN_NBITS = 5, SO_NBITS = 8

N_PAR_DEC	**45**	**90**	**180**	**360**
LUTs	12733	22870	62627	120167
Registers	8026	13924	26882	52211
Block RAM/FIFO 36Kb	173	247	330	540
DSP48	0	0	0	0
GCLKs	1	1	1	1

DVB-S2 LDPC decoder
Normalized min-sum algorithm
n_{ldpc} = 16200 short-frame only
IN_NBITS = 5, SO_NBITS = 8

N_PAR_DEC	**45**	**90**	**180**	**360**
LUTs	13108	27277	48528	86718
Registers	7807	14285	26351	50242
Block RAM/FIFO 36Kb	86.5	97	120	156
DSP48	0	0	0	0
GCLKs	1	1	1	1

DVB-S2 LDPC decoder
Normalized min-sum algorithm
normal, short, medium frames
IN_NBITS = 5, SO_NBITS = 8

N_PAR_DEC	45	90	180	360
LUTs	12439	23809	62733	122133
Registers	8075	13943	26909	52230
Block RAM/FIFO 36Kb	205	247	330	540
DSP48	0	0	0	0
GCLKs	1	1	1	1

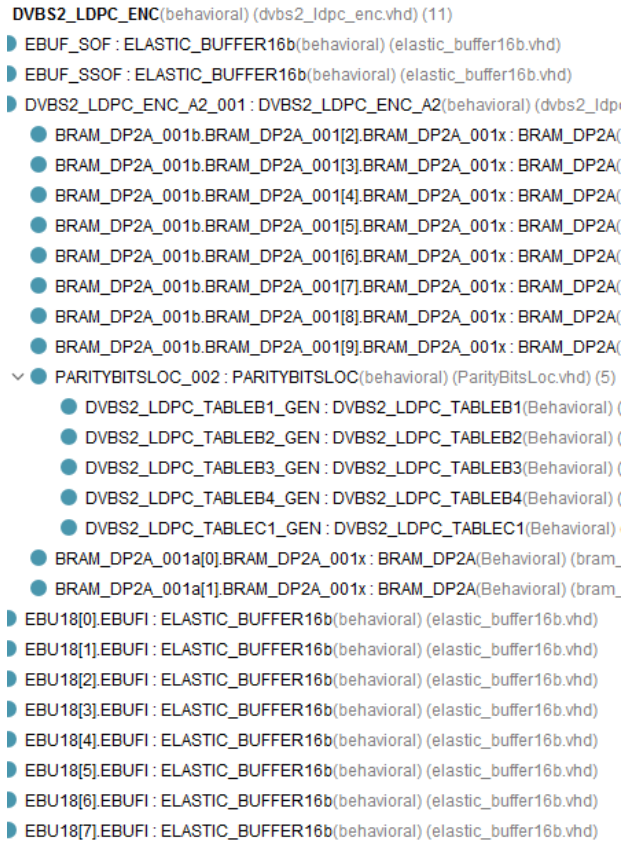
Maximum clock frequency

The entire design uses a single global clock CLK. Typical maximum clock frequencies for various FPGA families are listed below:

Device family	
Xilinx xczu7eg -1 (slowest) speed grade	**Decoder** Lambda-min algorithm Short frame only **270 MHz** Normal frame only **220 MHz**
Xilinx xczu7eg -1 (slowest) speed grade	**Decoder** Normalized min-sum algorithm Short frame only **305 MHz** Normal frame only **250 MHz**
Xilinx xczu7eg -1 (slowest) speed grade	**Encoder** All three frame lengths **343 MHz**
Xilinx xczu7eg -1 (slowest) speed grade	**Encoder** Short frame only **372 MHz**

VHDL components overview

Encoder top level

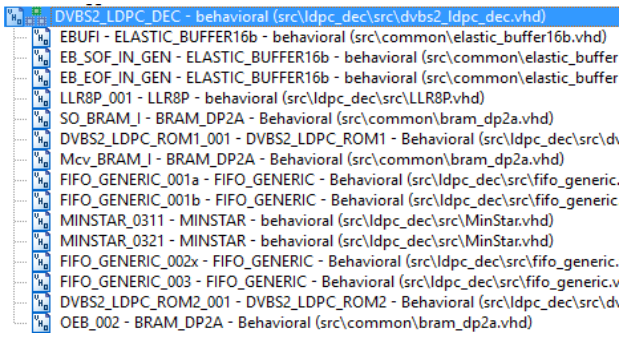


The *DVBS2_LDPC_ENC.vhd* component buffers the input Byte stream and computes the parity bits for each input frame. The concatenated information bits and parity bits are sent to the output. Both inputs and outputs are 8-bit parallel.

The *DVBS2_LDPC_ENC_A2* component computes the parity bits for each group of 360 input bits. The parity bits locations are read from lookup tables by *PARITYBITSLOC*.vhd, in accordance with the specifications [1] annex tables B, C and D. Each parity location is expressed as (row, column) such that parity location = column*q + row. The matlab function `paritybitsloc` generates the ROM contents from the specifications annex tables (see \ src\ldpc_enc\matlab)

BRAM_DP2A.vhd is a generic dual-port memory, used as input and output elastic buffers. Memory is inferred for code portability (no manufacturer-specific primitive is used).

Decoder top level



DVBS2_LDPC_DEC.vhd performs the iterative error correction decoding. The decoding stops when all parity checks are verified or when the number of decoding iterations reaches the maximum N_ITER_MAX, whichever occurs first. Eight input symbols are entered in parallel to maximize throughput. The lambda-min algorithm (algo2) is used for best BER performance. Alternatively, the normalized min-sum algorithm (algo3) could be used instead for a more compact and higher speed implementation.

LLR8P.vhd computes the LLR for each soft-decision input sample. The LLR is $2*y_i/\sigma^2$ where y_i are the soft-decoded input samples and σ^2 the noise variance. Although the component can scale the samples as a function of the SNR, a fixed SNR is set in the code as a tradeoff between computation precision and algorithm accuracy.

DVBS2_LDPC_ROM1.vhd is a generic dual-port ROM customized for reading the attributes of each Identity Matrix involved in parity checking: v-node, shift, etc. It is large enough to store such information for all supported code rates and lengths. In effect, scanning the non-zero elements of the parity check matrix H horizontally from top to bottom. The table contents is generated by the matlab utility `dvbs2_ldpc_rom1.m`. This matlab program re-orders the CNGs to minimize the total number of clocks per iteration while alleviating memory access conflicts.

MINSTAR2.vhd computes the minstar* function as described in [4] and [5].

MINSTAR.vhd computes the minstar* function as described in [4] while using fewer resources than *MINSTAR2.vhd.*

FIFO_GENERIC.vhd: synthesizable generic FIFO. organized as a circular buffer. Generally synthesized as LUTs, LUTRAMs (not BRAM)

DVBS2_LDPC_ROM2.vhd is a generic dual-port ROM customized for reading the amount of right circular shift while transferring the decoded bits to the output elastic buffer. The table contents is generated by the matlab utility `dvbs2_ldpc_rom2.m`.

INFILE2SIM.vhd reads an input file. This component is used by the testbench to read a 5-bit soft-quantized encoded bit stream generated by the `dvbs2_ldpc.m` or `dvbs2_ldpc_vhdlalgox.m` Matlab programs for various Eb/No cases.

SIM2OUTFILE.vhd writes three 12-bit data variables to a tab delimited file which can be subsequently read by Matlab (load command) for plotting or analysis.

VHDL simulation

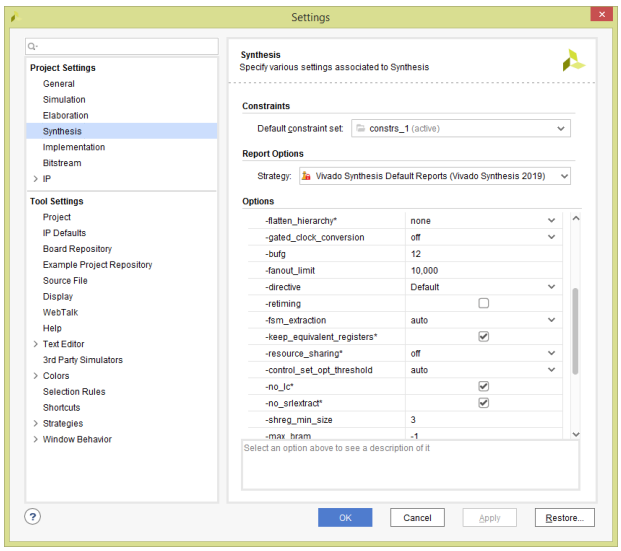
Three testbenches provide bit-accurate VHDL simulation avenues:

- *tb_DVBS2_LDPC_CODEC.vhd* is an end-to-end simulation testbench encompassing encoder, decoder, PRBS-11 test sequence generation and BER tester. Set the FRAME_TYPE and RATE constants before starting the VHDL simulation. The objective is to verify end-to-end compatibility. For noise simulation, see tb_DVBS2_LDPC_DEC.vhd.

- *tb_DVBS2_LDPC_ENC.vhd* is the encoder testbench.

- *tb_DVBS2_LDPC_DEC.vhd* is the decoder testbench. Its input consists of soft-quantized noisy samples generated by the Matlab programs `dvbs2_ldpc.m` and `dvbs2_ldpc_vhdlalgox.m` Set `outputFile = 1` in the Matlab program to save the stimulus file which will be used as input to the VHDL decoder.

Xilinx Vivado: Synthesis settings
(* denotes changes from the default settings)



Matlab simulation

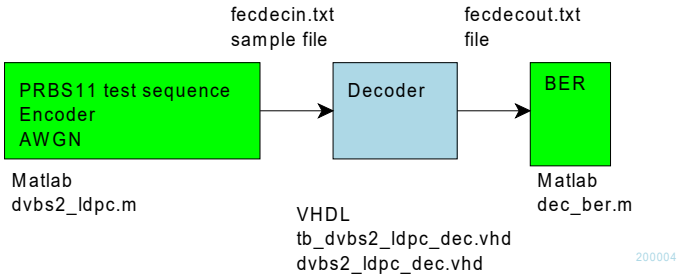
The `dvbs2_ldpc.m` program

- generates a stimulus file `fecdecin.txt` for use as input to the decoder VHDL simulation. The file includes a frame of pseudo-random (PRBS11) data bits, LDPC encoding, Additive White Gaussian Noise and 4- or 5-bit soft-quantization.
- Performs end-to-end BER performance analysis of the LDPC-codec over a noisy (AWGN) channel.

The `dvbs2_ldpc_vhdlalgo2.m` program simulates a lambda min decoding algorithm representative of the actual VHDL implementation, instead of a generic decoding algorithm.

Likewise, the `dvbs2_ldpc_vhdlalgo3.m` program simulates a normalized min-sum decoding algorithm.

The `dec_ber.m` program reads a file of decoded data `fecdeccout.txt`
generated by VHDL simulation and compare it with the original PRBS-11 test sequence. It counts the number of bit errors.



When moving the project folder location, be sure to change accordingly the FILENAME file paths in *tb_dvbs2_ldpc_dec.vhd* INFILE2SIM and SIM2OUTFILE components generic section.

The following .m programs were used during the design:

`dvbs2_ldpc_H.m` generates the parity check matrices H for the selected code rate and codeword length.

Alternatively, the matlab function `dvbs2ldpc()` also generates the parity check matrix H, although the use is limited to a codeword length nldpc=64800.

The function `paritybitsloc()` is a design utility to generate VHDL-format ROM contents for the look-up table components *dvbs2_ldpc_tableBX.vhd* and *dvbs2_ldpc_tableC1.vhd* .

`dvbs2_ldpc_rom1.m` generates the decoder ROM1 table contents. This matlab program optimizes the CNGs processing order to yield the smallest number of clocks per iteration while alleviating SO memory access conflicts. For each decoding timeslot, the ROM1 points to the VNG index, the left circular shift, the last VNG in a check node group and other relevant attributes.

`dvbs2_ldpc_rom2.m` generates the decoder final circular shift values when transferring the decoded bits to the output elastic buffer.

Implementation Overview

The decoder architecture follows the thesis in reference document [3].

Acronyms

Acronym	Definition
AWGN	Additive White Gaussian Noise
BRAM	Dual-port Block RAM
BER	Bit Error Rate
CN	Check Node
CNG	Check Nodes Group
DVB	Digital Video Broadcast
FER	Frame Error Rate
IM	Identity Matrix
LDPC	Low-Density Parity-Check
LLR	Log-Likelihood Ratio
LSb	Least Significant bit
Mcv	Messages from Check nodes to Variable nodes
Mvc	Messages from Variable nodes to Check nodes
MSb	Most Significant bit
N/A	Not Applicable
PER	Packet Error Rate (refers to a 188-Byte packet)
PRBS-11	Pseudo-Random Binary Sequence, 2047-bit period
SO	Soft Output
TS	Time Slot
VL-SNR	Very Low Signal-to-Noise Ratio
VN	Variable Node
VNG	Variable Nodes Group

ComBlock Ordering Information

COM-1813SOFT DVB-S2x LDPC encoder/decoder. VHDL source code / IP core

ECCN: EAR99

Contact Information

Comblock LLC • 845-N Quince Orchard Boulevard • Gaithersburg, Maryland 20878-1676 • U.S.A.

Phone: (240) 631-1111
E-mail: info@comblock.com

www.comblock.com