COM-1811SOFT
CCSDS LDPC C2 code encoder/decoder
VHDL source code overview / IP core

Overview

The COM-1811SOFT is a high-speed LDPC code error correction encoder/decoder written in generic VHDL.

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

Key features and performance:

- Includes encoding, decoding, frame synchronization and data randomization.
- Compliant with the C2 code specified in CCSDS 131.0-B-3, Blue Book, section 7.3
  - C2 code, rate 223/255
  - fixed-length frame size (8160,7136)
- Typical Bit Error Rate / Frame Error Rate:
  BER < 8.7 10^{-8} FER < 1 10^{-5} @ E_b/N_o = 3.9 dB
- Throughput:
  Encoding: > 1 Gbits/s
  Decoding: 50 - 500 Mbits/s depending on FPGA type and operating E_b/N_o
- 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

Target Hardware

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.

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	
No generic section	
Decoder	
Number of soft-quantized bits at the decoder input **IN_NBITS**	Typical values: 4. A minor performance improvement can be achieved with 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.

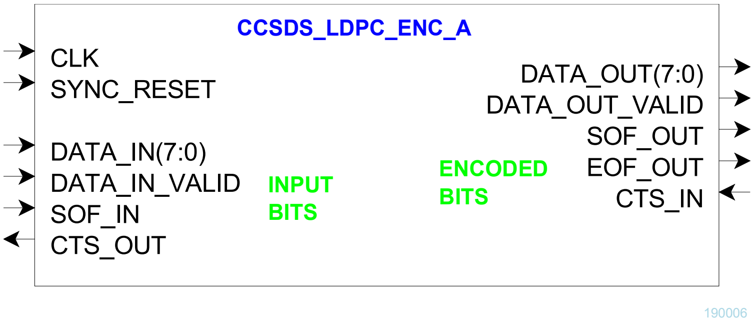
I/Os

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



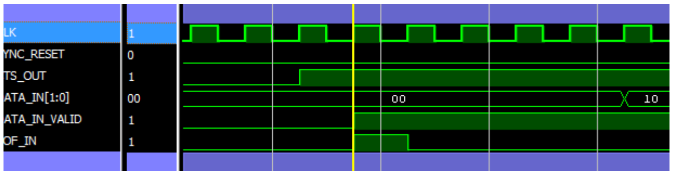
DATA_IN(7:0): Input data is read one Byte at a time. Bits are packed LSb first.

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

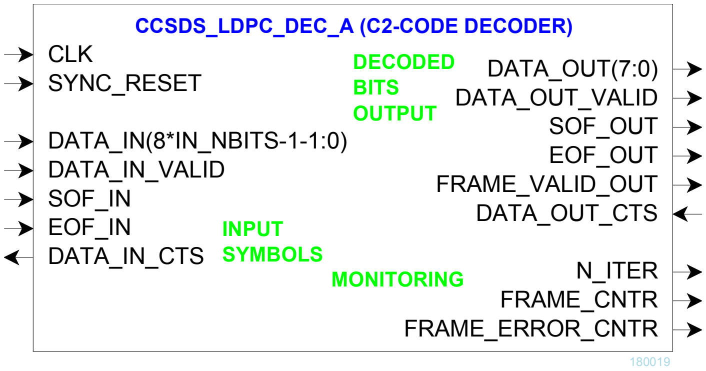
SOF_IN: input Start Of Frame. 1 CLK-wide pulse. The SOF is aligned with **DATA_IN_VALID**. Note that there is no need for an end of frame as the input frame size must be fixed at 7136 bits or 892 Bytes. Input bits in excess are discarded.

CTS_OUT: output.
Clear-To-Send 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**, **SOF_OUT**, **EOF_OUT**, **CTS_IN**.



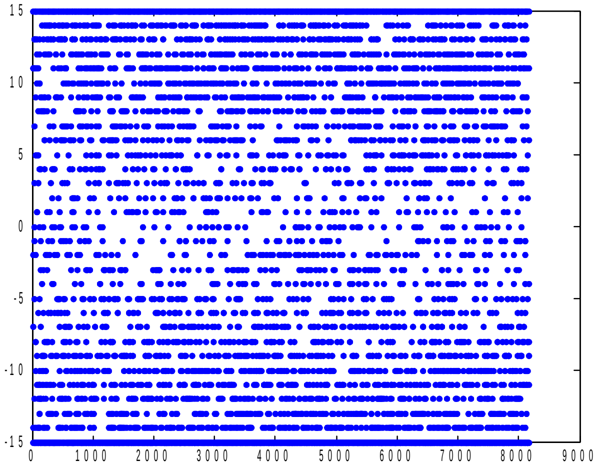
Decoder



DATA_IN: eight soft-quantized input symbols. The 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 LLR8P.vhd

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

Usage: it is expected that the demodulator preceding this decoder will normalize the demodulated samples prior to soft-quantization by using an AGC loop. The AGC target level is important in maximizing the decoder BER performance. The optimum level is such that the noiseless input samples are at the quantized range boundaries. Below is an example of quantized input samples for Eb/No = 3.5 dB



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

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 7136 symbols, entered 8 at a time.

DATA_IN_CTS: output Clear-To-Send 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**.
Output data **DATA_OUT** is sent one Byte at a time. Bits are packed LSb first.

The **FRAME_VALID_OUT** flag indicates whether the frame was successfully decoded (1) or not (0). This information, together with the actual number of decoding iterations **N_ITER** are available during the entire output frame, i.e. between **SOF_OUT** and **EOF_OUT**.

FRAME_CNTR, FRAME_ERROR_CNTR can be read starting one CLK after **SOF_OUT**.

Performance

Encoder throughput

The maximum encoder throughput corresponds to one 7136-bit input frame, and thus one 8160 output frame, every 1103 clocks. For example, with a 200 MHz processing clock, the encoding throughput is 1.29 Gbits/s uncoded bits, or 1.48 Gbits/s encoded bits.

Decoder latency

The decoder can only handle one frame at a time. The latency between input SOF and decoded output EOF depends on the number of decoding iterations needed to correct all errors.

- Best case (1 iteration): 550 clks + 1020 input words (8160 input samples) + 895 output words (7154 decoded bits)
- Worst case (50 iterations): case (1 iteration): 50*550 clks + 1020 input words (8160 input samples) + 895 output words (7154 decoded bits)
- Note that the new frame input and previous frame output can be concurrent.

Decoder throughput

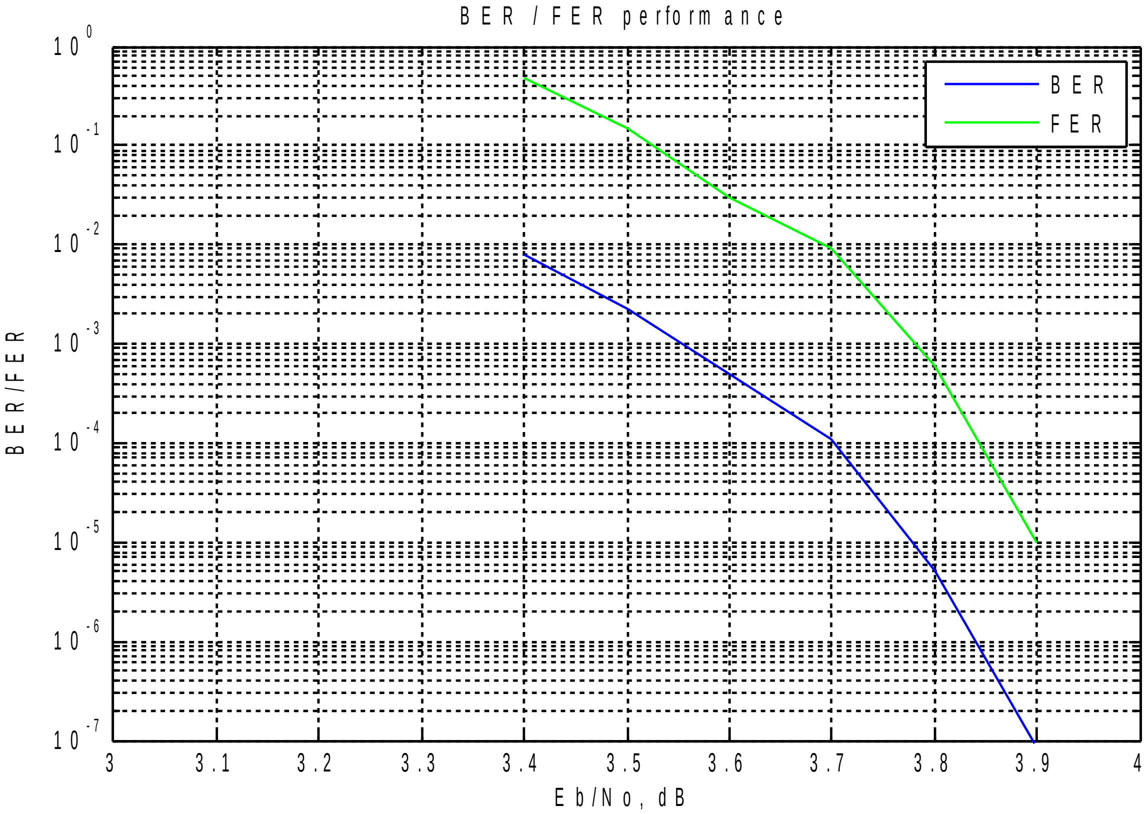
The average decoder throughput depends on the worst case Eb/No and FPGA technology.

FPGA speed	min Eb/No	Average number of decoding iterations	Average decoded throughput
Artix7-100T 80 MHz	3.5 dB	17.6	53 Mbits/s decoded
Artix7-100T 80 MHz	3.7 dB	9.2	93 Mbits/s decoded
Artix7-100T 80 MHz	3.8 dB	7.6	109 Mbits/s decoded
Artix7-100T 80 MHz	4.0 dB	5.8	135 Mbits/s decoded
Artix7-100T 80 MHz	4.5 dB	3.8	183 Mbits/s decoded
Artix7-100T 80 MHz	5.0 dB	2.8	223 Mbits/s decoded
Artix7-100T 80 MHz	6.0 dB	1.7	292 Mbits/s decoded
Artix7-100T 80 MHz	7.0 dB	1.1	351 Mbits/s decoded

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).

50 iterations:
BER = 7.8 10^{-3} FER = 0.48 @ E_b/N_o = 3.4 dB
BER = 2.2 10^{-3} FER = 0.15 @ E_b/N_o = 3.5 dB
BER = 5.0 10^{-4} FER = 0.03 @ E_b/N_o = 3.6 dB
BER = 1.1 10^{-4} FER = 0.009 @ E_b/N_o = 3.7 dB
BER = 6 10^{-6} FER = 6 10^{-4} @ E_b/N_o = 3.8 dB
BER < 8.7 10^{-8} FER < 1 10^{-5} @ E_b/N_o = 3.9 dB



Software Licensing

The COM-1811SOFT 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 122620.

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
/bin	.bit configuration files (for use with ComBlock COM-1800 FPGA development platform)

Project files:

Xilinx ISE 14 project file: com-1811.xise
Xilinx Vivado v2017.4 project file: project_2.xpr

VHDL development environment

The VHDL software was developed using the following development environment:

(a) Xilinx ISE 14.7 for synthesis, place and route
(b) Xilinx Vivado 2017.4 for synthesis, place and route and VHDL simulation

The entire project fits easily within a Xilinx Artix7-100T. Therefore, the ISE project can be processed using the free Xilinx WebPack tools.

Portable VHDL code

The VHDL source code is written in generic VHDL and thus can be ported FPGAs from various vendors. No Xilinx CORE nor Xilinx primitive is used.

Reference documents

[1] CCSDS “Recommended Standard for TM Synchronization and Channel Coding”,

CCSDS 131.0-B-3, Blue Book, September 2017.

Applicable sections:
Section 7.3: Low-density parity check code with rate 223/255 (for C2 code)

Section 9: Frame Synchronization

Section 10: Pseudo-Randomizer

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

[3] '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

Device: Xilinx Artix7-100T

C2 code (8160,7136) LDPC encoder		% of Xilinx Artix7-100T
Registers	3219	2.5%
LUTs	6873	10.8%
Block RAM/FIFO 36Kb	0.5	0.4%
DSP48	0	0%
GCLKs	1	3.1%

for LQNBITS=9

C2 code (8160,7136) LDPC decoder		% of Xilinx Artix7-100T
Registers	18870	14.9%
LUTs	27253	43.0%
Block RAM/FIFO 36Kb	45	33.3%
DSP48	8	3.3%
GCLKs	1	3.1%

Clock and decoding speed

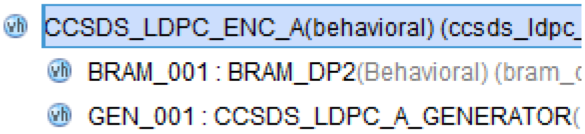
The entire design uses a single global clock CLK. Typical maximum clock frequencies for various FPGA families are listed below:

Device family	**Encoder**	**Decoder**
Xilinx Artix 7 -1 speed grade	239 MHz	81 MHz*

(*) as of 12/26/20. Likely to be optimized in subsequent releases

VHDL components overview

Encoder top level

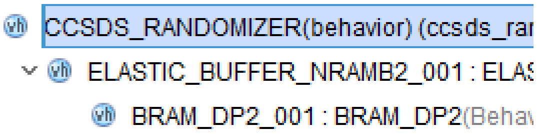


```
CCSDS_LDPC_ENC_A(behavioral) (ccsds_ldpc_
    BRAM_001 : BRAM_DP2(Behavioral) (bram_c
    GEN_001 : CCSDS_LDPC_A_GENERATOR(
```

The *CCSDS_LDPC_ENC_A* component computes the parity bits for an input frame of 7154 bits using the generator matrix for the systematic (8176,7154) subcode (C2 code). Both inputs and outputs are 8-bit parallel. Thus, the maximum encoded rate is slightly below 8 encoded bit per clock.

The *CCSDS_LDPC_A_GENERATOR* component generates the first rows of two 511*511 circulant matrices for the LDPC C2 code.

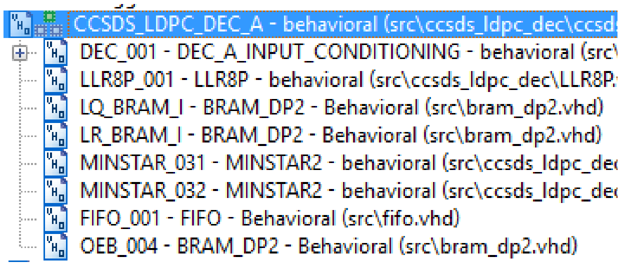
BRAM_DP2.vhd is a generic dual-port memory, used as input and output elastic buffers. Memory is inferred for code portability (no Xilinx primitive is used).



```
CCSDS_RANDOMIZER(behavior) (ccsds_ran
    ELASTIC_BUFFER_NRAMB2_001 : ELAS
        BRAM_DP2_001 : BRAM_DP2(Behav
```

The *CCSDS_RANDOMIZER.vhd* component performs bit stream pseudo-randomization and sync marker insertion as per sections 9 and 10 of the specifications [1].

Decoder top level



```
CCSDS_LDPC_DEC_A - behavioral (src\ccsds_ldpc_dec\ccsd
    DEC_001 - DEC_A_INPUT_CONDITIONING - behavioral (src\
    LLR8P_001 - LLR8P - behavioral (src\ccsds_ldpc_dec\LLR8P.
    LQ_BRAM_I - BRAM_DP2 - Behavioral (src\bram_dp2.vhd)
    LR_BRAM_I - BRAM_DP2 - Behavioral (src\bram_dp2.vhd)
    MINSTAR_031 - MINSTAR2 - behavioral (src\ccsds_ldpc_dec
    MINSTAR_032 - MINSTAR2 - behavioral (src\ccsds_ldpc_dec
    FIFO_001 - FIFO - Behavioral (src\fifo.vhd)
    OEB_004 - BRAM_DP2 - Behavioral (src\bram_dp2.vhd)
```

CCSDS_LDPC_DEC_A.vhd performs the iterative error correction decoding. The decoding stops when all 1022 parity checks are verified or when the number of decoding iterations reaches the maximum N_ITER_MAX, whichever occurs first. Each decoding iteration takes 530 clocks. Eight input symbols are entered in parallel to maximize throughput.

DEC_A_INPUT_CONDITIONING.vhd inserts 18 zero symbols and discard the last two received symbols from the 8160-symbol received frame (as per CCSDS standard).

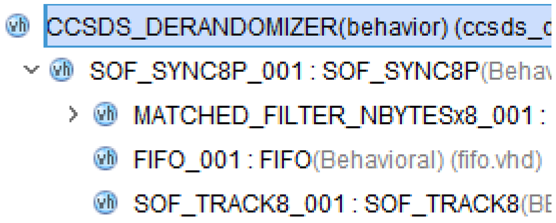
LLR8P.vhd computes the LLR for each soft-decision input sample. The LLR is $2*y_i/\sigma^2$ where y_i are the soft-decoded input samples and σ^2 the noise variance. Although the component can scale the samples as a function of the SNR, a fixed SNR is set in the code as a tradeoff between computation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

MINSTAR2.vhd computes the minstar* function as described in [2] and [3].

MINSTAR.vhd computes the minstar* function as described in [2] while using fewer resources than *MINSTAR2.vhd.*

INFILE2SIM.vhd reads an input file. This component is used by the testbench to read a soft-quantized encoded bit stream generated by the `ccsds_ldpc_a.m` Matlab program 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.



```
CCSDS_DERANDOMIZER(behavior) (ccsds_c
  SOF_SYNC8P_001 : SOF_SYNC8P(Beha
    MATCHED_FILTER_NBYTESx8_001 :
    FIFO_001 : FIFO(Behavioral) (fifo.vhd)
    SOF_TRACK8_001 : SOF_TRACK8(BE
```

The *CCSDS_DERANDOMIZER.vhd* component detects and removes the periodic sync markers, reconstructs the start of frame and end of frame pulses and descrambles the received soft-quantized bit stream. It complies with sections 9 and 10 of the specifications [1].

Matlab simulation

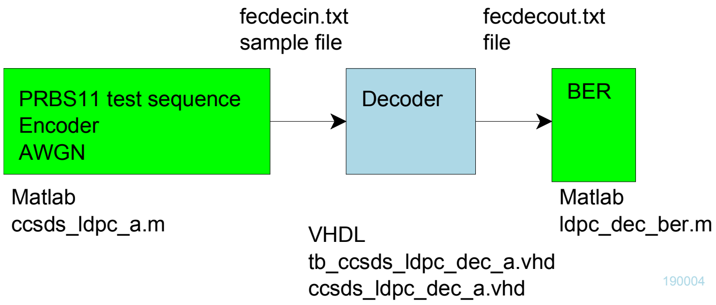
`ccsds_ldpc_a_H.m`: generates the 8176 x 1022 parity check matrix H for the CCSDS C2 family LDPC code with rate 223/255, as per [1], section 7.3

`ccsds_ldpc_a_G.m`: constructs the generator matrix G for the CCSDS C2 family LDPC code with rate 223/255, as per [1] section 7.3.4 and Annex C.

The `ccsds_ldpc_a.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 soft-quantization.
- Performs end-to-end BER performance analysis of the LDPC-codec over a noisy (AWGN) channel.

The `ldpc_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_ccsds_ldpc_dec_a.vhd* INFILE2SIM and SIM2OUTFILE components generic section.

Implementation Overview

C2 code decoder

A received frame consists of 8160 soft-quantized symbols with 4 or 5 bits each.

The received frame is extended and shortened as follows: insert 18 zero symbols and drop the last two received symbols, for a new frame size of 8176 symbols at the decoder input.

The decoder output frame comprises 7154 bits.

The decoder uses two groups of block RAMs to store the LQ(0:8175) and Lrji(0:32703) respectively

Computations are performed with the following precision:
LQ: 9-bit signed fixed-point 6.3
Lrji: 7 bit signed fixed point 3.3
LLR input: 4 or 5-bit

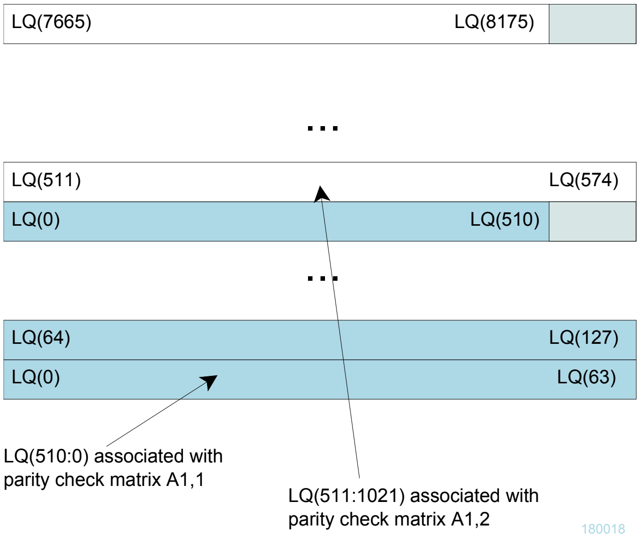
Messages between v-nodes and c-nodes are computed in parallel by groups of 64 to maximize throughput. It takes 32 clocks to compute the v-nodes to c-nodes messages.

Lr memory organization

All 32704 Lrji are saved in dual-port block RAM with signed fixed-point format 4.2.
11 BRAMs are used, with 6 Lrji per 36-bit word, thus allowing access to 64 Lrji per clock.
The dual-port A-side is reserved exclusively for writing while the B-side is for reading only.

LQ memory organization

The LQi are stored in 64 BRAMs with 8-bit wide data path. The LQ memory organization is illustrated below:


LQ(7665)
LQ(8175)
…
LQ(511)
LQ(574)
LQ(0)
LQ(510)
…
LQ(64)
LQ(127)
LQ(0)
LQ(63)
LQ(510:0) associated with parity check matrix A1,1
LQ(511:1021) associated with parity check matrix A1,2
180018

Thus, 64 LQi can be read and written in one clock cycle. Each BRAM stores 128 LQi with a precision of 8-bits, for a total of 1Kbit.

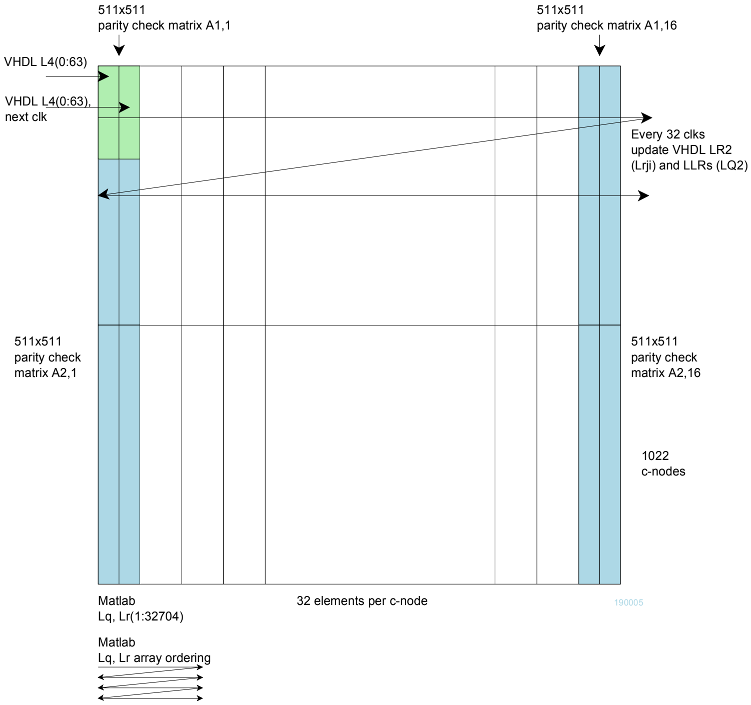
Update: the final code actually uses 9-bit precision for LQi

Matlab-VHDL

Although the Matlab program `ccsds_ldpc_a.m` and the VHDL component ccsds_ldpc_dec_A.vhd implement the same fundamental algorithm, the VHDL code can be difficult to understand. To help follow the algorithmic steps, a list of the key variable names in VHDL and their corresponding names in the Matlab program is shown below:

Algorithm	Matlab variable	VHDL variable
LLR	Lc(1:8176)	LLR1 8 sample-word, fixed-point format 5.3
LQ_i	LQ(1:8176)	LQ36 64-sample word, before saving to LQ memory. LQ1 64-sample word after reading from LQ memory. LQ2 re-arranged LQ_i for c-nodes. 64 c-nodes elements for 32 clocks All are signed fixed-point format 5.3
Lr_{ji}	Lr(1:32704)	LR2x 64-samples before saving to memory. LR1x 64 samples signed after reading from memory. Both are signed fixed point format 5.3
Lq_{ij}		L4 or L7(after 32-clk delay) signed fixed-point format 5.3

A key algorithmic difference between the reference Matlab simulation and the VHDL code is the LQi / Lrji update rate. The Matlab algorithm updates the LQi and Lrji in-block once every iteration, while the VHDL code does a progressive update every 32 CLKs (= every 64 check nodes). Consequently, the VHDL code converges faster and requires fewer iterations.



Acronyms

Acronym	Definition
AWGN	Additive White Gaussian Noise
CCSDS	Consultative Committee For Space Data Systems
BRAM	Block RAM
CCSDS	Consultative Committee For Space Data Systems
LDPC	Low-Density Parity-Check
LLR	Log-Likelihood Ratio
LSb	Least Significant bit
MSb	Most Significant bit
PRBS-11	Pseudo-Random Binary Sequence, 2047-bit period

ComBlock Ordering Information

COM-1811SOFT CCSDS LDPC C2 code encoder/decoder. VHDL source code / IP core

Contact Information

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

Telephone: (240) 631-1111
Facsimile: (240) 631-1676
E-mail: info@comblock.com