BCH (1023, 903, 12) Encoder and Decoder

Application notes

Rev. [6/2012]

6-19-2012

Table of Contents

Introduction. 3

Parameters defining BCH encoder and decoder.......... 3

General parameters 3

Code parameters.......... 3

BCH encoder interface. 4

BCH decoder interface. 5

Timing diagram of the encoder.......... 6

Decoder’s timing diagram. 7

Introduction.

The BCH design includes two independent units, an encoder for generation of the BCH code, and decoder for the BCH code decoding. They can be used separately or together. They are included in separate vhdl files. The encoder is described in the file: "/src/encoder/encoder.vhd". The decoder is described in two files. One of them ("/src/decoder/decoder_components.vhd") includes essential components, the other ("/src/decoder/decoder.vhd") includes decoder's architecture. The independent parameters, design constants are in the following files: "/src/encoder/encoder_constants.vhd" for the encoder, and "/src/decoder/decoder_constants.vhd" for the decoder.

Additionally, the design includes the main unit, which tests the correct working of the encoder and decoder units ("/src/main/main.bde"). Testbenches and all scripts needed for simulation and implementation are in the "/src/main/testbench/main_TB.vhd" file.

The BCH design is available in two versions, in VHDL language (BCH_1023_90_12_vhdl), and in Verilog language (BCH_1023_903_12_verilog).

Parameters defining BCH encoder and decoder.

The Encoder and decoder have parameters, which define it. They are included in separate files (the encoder_constants.vhd for the encoder, and the decoder_constant.vhd for the decoder). For verilog version of this design, parameters are not included in separate files, but they are the part of other project files. The most important parameters are as follows:

General parameters

- **Edge** - an active clock edge for the entire project ('1' for rising edge, '0' for falling);
- **Clk_freq** – a clock frequency [MHz], for simulation

  (for this project clk_freq= 50MHz)

Code parameters

- **m -** defines size of GF. For this project m= 10
- **n** - length of codeword in bits. For this project: n= 1023
- **k** – number of information bits in a codeword. For this project: k= 903
- **t –** the maximum number of error bits that can be corrected.

  For this project: t= 12

BCH encoder interface.

The encoder and decoder have a similar interface. They differ only by one signal. The encoder has the *data_in_en* signal (output signal), while the decoder does not have it.

The encoder's symbol is presented below, in the figure 2.

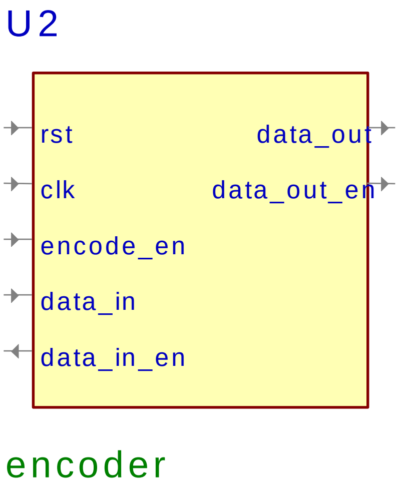

Figure 2. Encoder symbol.

All encoders' input signals are sampled the on rising clock edge, so they must be valued after it. These signals should be set on falling edge by the previous device.

All encoder's output signals (without *data_in_en*) are changed on the falling clock edge, so it allows decoder on sampling it on the rising edge.

The description of encoders' signals is presented below:

- **rst –** an input for the reset signal, is active high level. All encoder registers and flops have the synchronous reset, so it is required that this signal lasts minimum for two periods.
- **clk** – an input for the clock signal.
- **encode_en** – encodes enable signal, is active high level. When it is high, encoding is enabled, otherwise the entire encoder is disabled. It can last optionally long, and it must be valued before the rising edge.
- **data_in** – an input data for encoding. Data is sent serially on this input, and it must be valued before the rising edge.
- **data_in_en** – a strobe signal for the input data. It is an output signal, which is changed on the rising clock edge, in order to allow previous device to set its output data on the falling edge. It depends on the *encode_en* signal, when the *encode_en* signal is low, the *data_in_en* is low too.
- **data_out** – an output data for the encoder. Data is sent serially by this output to decoder or other device. It is changed on the falling clock edge, so it allows decoder to sample it on the rising edge.
- **data_out_en** – a strobe signal for the output data. It depends on the *encode_en* signal, and is low, when the *encode_en* is low. It is changed on the falling edge too.

BCH decoder interface.

The decoder's symbol is presented in the figure 3.

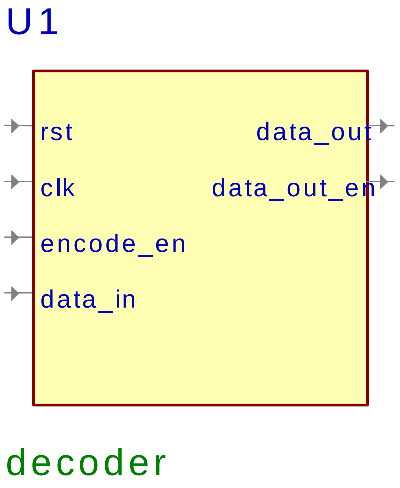

Figure 3. Decoder symbol.

All decoders' input signals are sampled on rising clock edge, so they must be valued after it. These signals are set on the falling edge by the encoder, or other device.

All decoders' output signals are changed on falling clock edge, so it allows other device to sample it on the rising edge.

The signals description is presented below:

- **rst -** an input for reset signal, is active high level. All encoder registers and flops have synchronous reset, so it is required that this signal lasts minimum for two periods.
- **clk** – an input for the clock signal.
- **encode_en** – encodes the enable signal, is active high level. When it is high, decoding is enabled, otherwise the entire decoder is disabled. It can last optionally long, and it must be valued before the rising edge.
- **data_in** – an input data for decoding. Data is sent serially on this input, and it must be valued before the rising edge.
- **data_out** – an output data for the decoder. Data is sent serially by this output to other devices. It is changed on the falling clock edge so it allows this device to sample it on the rising edge.
- **data_out_en** – a strobe signal for the output data. It depends on the *encode_en* signal. It is low, when encode_en is low. It is changed on the falling edge too.

Timing diagram of the encoder.

First the encoder's timing diagram shows only 600ns of simulation time. After 60ns of simulation information bits are sent on the encoder's input and they are available on the output after one clock period delay. It is presented in the figure 4.

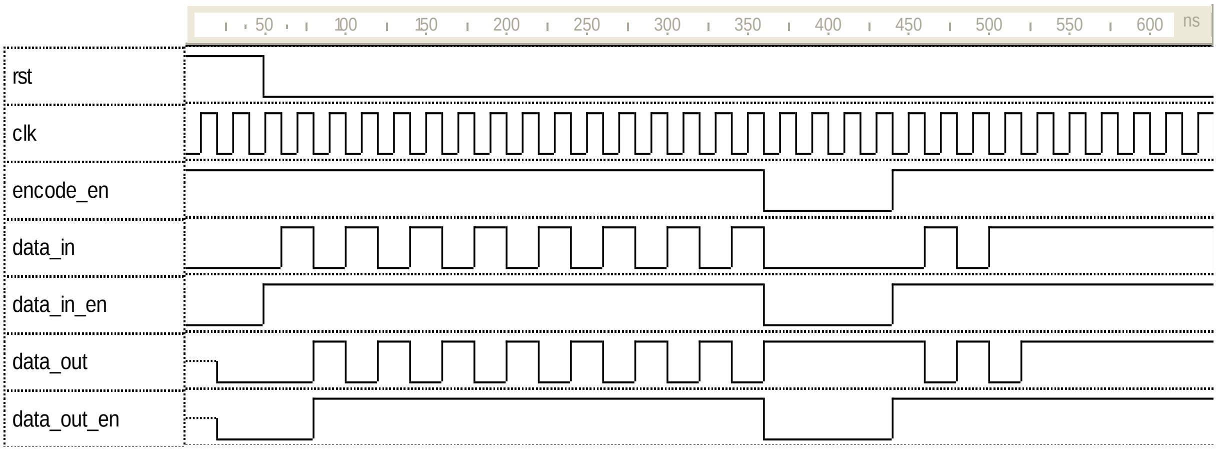

Figure 4. Timing diagram of the encoder.

The next diagram shows a simulation time from 18.10us till 18.50us. From 18.28us the parity bits are generated (after k information bits). It is presented in the figure below (figure 5).

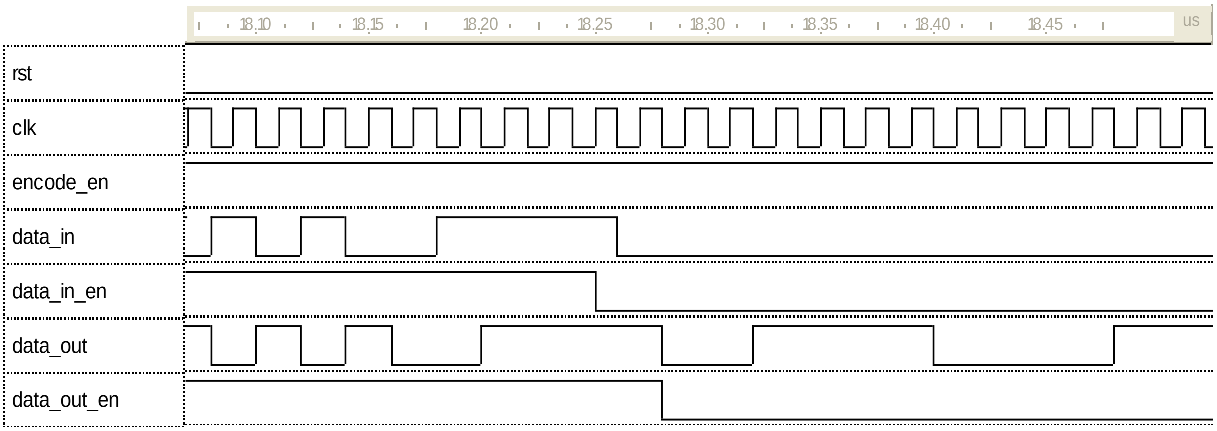

Figure 5. Timing diagram of the encoder.

Simulation clock frequency for the encoder is equal 50MHz. Reset signal is active by two clock periods. The input data (information for encoding) is sent bit after bit from the *data_in.txt* file. The output data (encoded information) is recorded in the *data_out.txt* file. All essential actions are automatically realized by the testbench.

Decoder's timing diagram.

First the encoder's timing diagram shows only 600ns of simulation time. The input data (encoded) is sent to the decoder's input after an encoding delay (one clock period). It is presented in the figure 6.

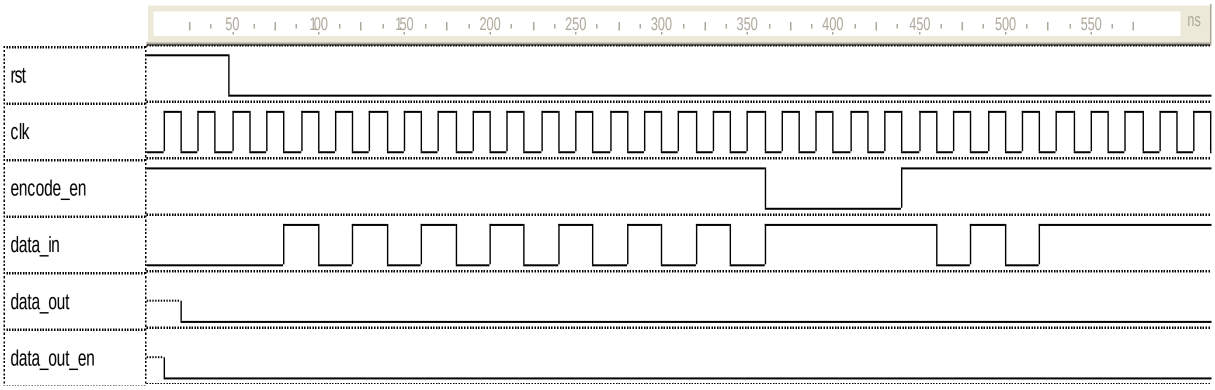

Figure 6. Timing diagram of the decoder.

The next diagram shows simulation time from 24.05us till 24.65us. From 24.08us the output data, corrected if needed, is available on the decoder output. It is presented below in the figure 7.

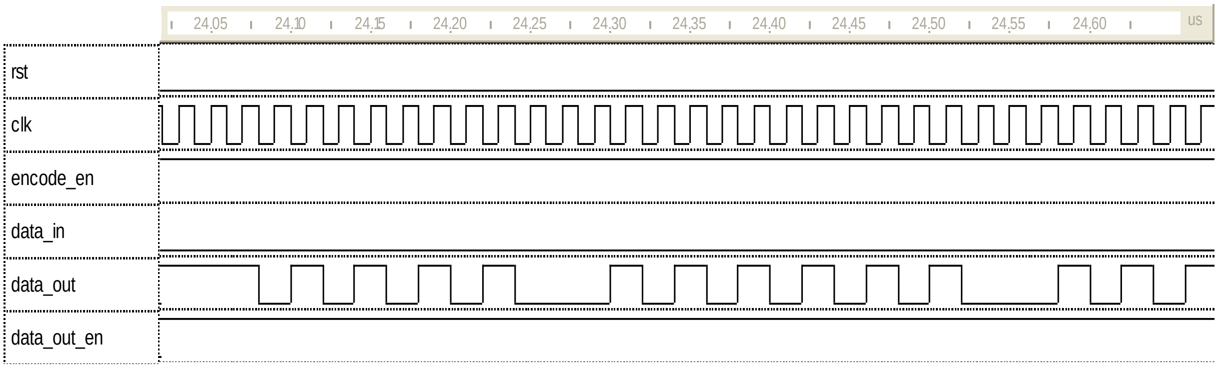

Figure 7. Timing diagram for decoder.

Simulation clock frequency for the decoder is equal 50MHz. Reset signal is active by two clock periods. The input data (encoded information) is sent bit after bit from the *data_in.txt* file. The output data (decoded information) is recorded in the *data_out.txt* file. All essential actions are automatically realized by the testbench.