3GPP Turbo Encoder v4.0

DS319 June 24, 2009 **Product Specification**

Features

- Drop-in module for Virtex®-4, Virtex-5, Virtex-6, Spartan®-6, Spartan-3, and Spartan-3E FPGAs
- Implements the 3GPP/UMTS specification [Ref 1] [Ref 2]
- Core contains the full 3GPP interleaver
- Full 3GPP block size range 40 - 5114 supported
- Up to 16 simultaneous data channels
- Double-buffered symbol memory for maximum throughput
- Rate 1/3 parity outputs for 3GPP and optional outputs for rate 1/5 for additional error correction capability
- Flexible interfacing by means of optional control signals
- External RAM option
- External address generator option enables interfacing to customer-designed interleaver
- Available through the Xilinx CORE Generator™ 11.2 and later software

Applications

This version of the Turbo Convolution Code (TCC) encoder is designed to meet the 3GPP mobile communication system specification [Ref 1] [Ref 2]. Optional outputs for code rate 1/5 are provided for additional flexibility.

General Description

The theory of operation of the Turbo Codes is described in the paper by Berrou, Glavieux, and Thitimajshima. [Ref 3].

The 3GPP Turbo Encoder input and output ports are shown in Figure 1 and the internal architecture is shown in Figure 2. It is a block-based processing unit where blocks of between 40 and 5114 bits are input via the DATA_IN port and processed. Each block of data is processed in two identical Recursive Systematic Convolutional (RSC) encoders, which generate high-weight codes. RSC1 processes the raw input data while RSC2 processes an interleaved version of the input data. The coding operates on the principle that if an input symbol is corrupted in the sequence from RSC1, then it is unlikely also to be corrupted in the re-ordered sequence from RSC2, and vice versa.

The delay shown in Figure 2 is used to indicate that the input data is delayed before passing through RSC1. This ensures that the systematic data from RSC1 and RSC2 are block-aligned at the output.

For the 3GPP standard [Ref 1] [Ref 2], a rate 1/3 turbo encoder is specified. In this case only the systematic and parity0 signals are generated for each RSC. For additional flexibility and error correction capability, the user can select a rate 1/5 option, in which case the two parity1 outputs will be made available.

Often some of the encoded output bits need not be transmitted, so they are omitted, or *punctured* from the output stream. Puncturing offers a dynamic trade-off between code rate and error performance. When the channel is noisy or the data requires more protection, extra redundancy can be added, therefore lowering the code rate. Puncturing is not implemented as part of the core; all outputs are provided by the core which allows the user maximum flexibility.

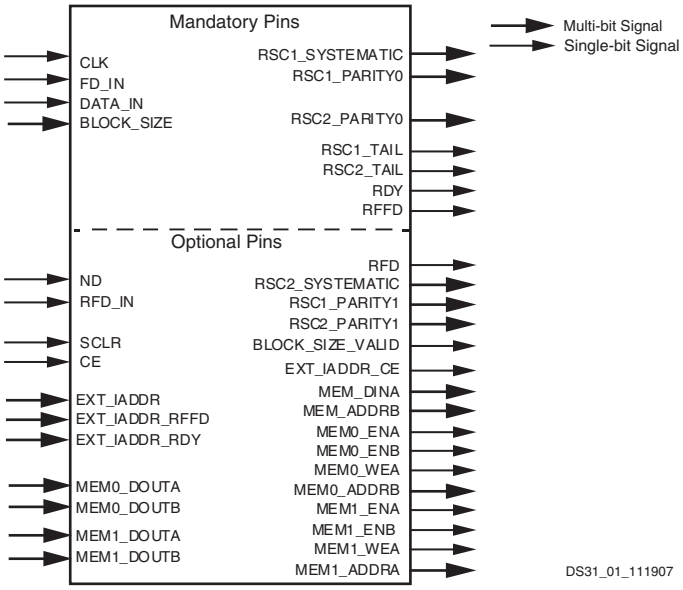
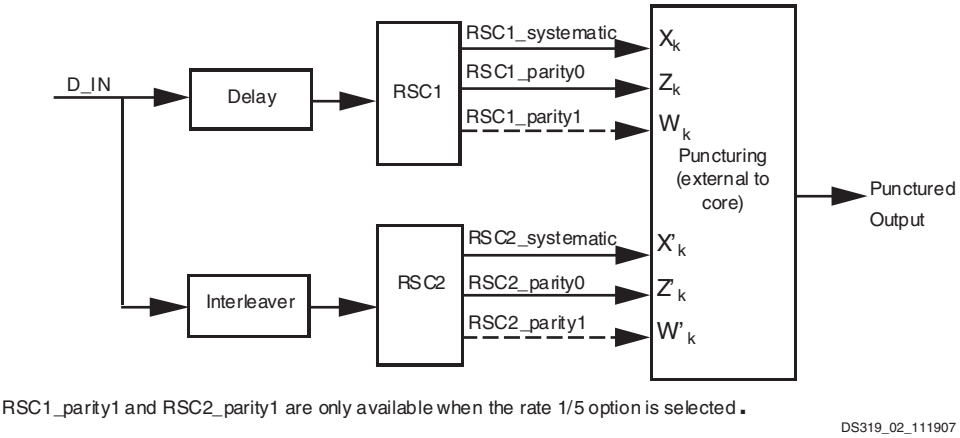


Figure 1: **TCC Encoder Structure**



RSC1_parity1 and RSC2_parity1 are only available when the rate 1/5 option is selected.

Figure 2: **TCC Encoder Structure**

Recursive Systematic Convolution (RSC) Encoder Structure

The schematic for each of the two RSCs and the transfer functions for the outputs are shown in Figure 3. Parity1(Y_1) may be optionally enabled if a rate 1/5 encoder is desired. This is not required by the 3GPP specification [Ref 1] [Ref 2].

After a block of input data has been coded, the RSCs must return to the initial zero state. To force the RSC back to the all zero state, the input value is set equal to the feedback value by setting the control switch, SW1, to the lower position for three clock cycles. During the first three tail bit periods, RSC2 is disabled, and the control switch of RSC1 is set to the lower position to output the RSC1 tail bits on the RSC1 systematic and parity outputs. During the last three tail bit periods, RSC1 is disabled and the control switch of RSC2 is set to the lower position to output the RSC2 tail bits on the RSC2 systematic and parity outputs.

The RSC2 systematic tail bits are also output on the RSC1 Systematic output. This avoids the need for the user to connect the RSC2_SYSTEMATIC port.

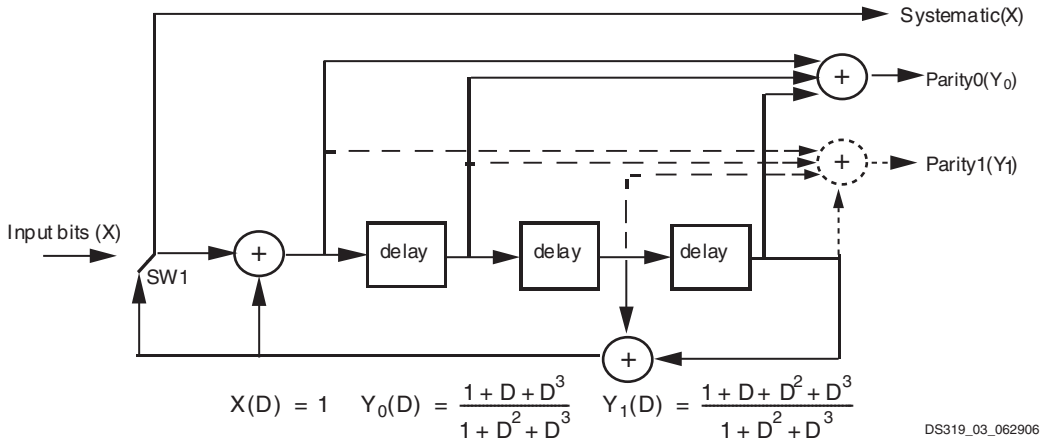


Figure 3: **TCC RSC Structure**

Input/Output Ports (Basic Configuration)

The I/O ports for the basic configuration are summarized and described in Table 1. Additional I/O ports apply if either the external address generator option or the external RAM option is selected. These are described later in this data sheet.

Table 1: **I/O Ports, Basic Configuration**

Pin	Sense	Port Width (bits)	Description
FD_IN	Input	1	**First Data** - When this is asserted (High) on a valid clock edge, the encoding process is started.
ND	Input (optional)	1	**New Data** - When this is asserted (High) on a valid clock edge, a new input value is read from the DATA_IN port.
SCLR	Input (optional)	1	**Synchronous Clear** - When this is asserted (High) on an active clock edge, the encoder is reset.
CE	Input (optional)	1	**Clock Enable** - When this is deasserted (Low), rising clock edges are ignored and the core is held in its current state.
CLK	Input	1	**Clock** - All synchronous operations occur on the rising edge of the clock signal.
BLOCK_SIZE	Input	13	**Block Size** - The block size of the current encode operation.
DATA_IN	Input	1 to 16	**Data Input** - The data to be encoded.
RSC1_SYSTEMATIC	Output	1 to 16	**RSC1_systematic** - The systematic output from RSC1.
RSC1_PARITY0	Output	1 to 16	**RSC1_parity0** - The parity0 output from RSC1.
RSC1_PARITY1	Output (optional)	1 to 16	**RSC1_parity1** - The parity1 output from RSC1.
RSC2_SYSTEMATIC	Output (optional)	1 to 16	**RSC2_systematic** - The systematic output from RSC2.
RSC2_PARITY0	Output	1 to 16	**RSC2_parity0** - The parity0 output from RSC2.
RSC2_PARITY1	Output (optional)	1 to 16	**RSC2_parity1** - The parity1 output from RSC2.
RSC1_TAIL	Output	1	**RSC1_tail** - Indicates that the tail bits are being output on RSC1 when asserted (High).
RSC2_TAIL	Output	1	**RSC2_tail** - Indicates that the tail bits are being output on RSC2 when asserted (High).
RFFD	Output	1	**Ready For First Data** - When this is asserted (High), the core is ready to start another encoder operation.
RFD	Output (optional)	1	**Ready For Data** - When this is asserted (High), the core is ready to accept input on the DATA_IN port.
RDY	Output	1	**Ready** - Indicates that there is valid data on the systematic and parity outputs when asserted (High).
BLOCK_SIZE_VALID	Output (optional)	1	**Block Size Valid** - Indicates that the block size which was sampled on the BLOCK_SIZE port on the last valid-FD is a valid 3GPP block size in the range 40 through 5114. The BLOCK_SIZE_VALID port is not available if the external address generator option is selected.

Clock (CLK)

With the exception of asynchronous clear, all operations of the core are synchronized to the rising edge of CLK. If the optional CE pin is enabled, an active rising clock edge occurs only when CE is High, and if CE is Low, the core is held in its current state.

Clock Enable (CE)

Clock enable is an optional input pin that is used to enable the synchronous operation of the core. When CE is High, a rising edge of CLK is acted upon by the core, but if CE is Low, the core remains in its current state. An active rising clock edge is on one which CE (if enabled) is sampled High.

Synchronous Clear (SCLR)

The SCLR signal is optional. When it is asserted High on a valid clock edge, the core is reset to its initial state, that is., the core is ready to process a new block. Following the initial configuration of the FPGA, the core is automatically in the reset state, so no further SCLR is required before an encoding operation can take place. If the CE input port is selected, SCLR is ignored if CE is Low.

Data In (DATA_IN)

The DATA_IN port is a mandatory input port which carries the unencoded data. The input process is started with a valid-FD signal, and data is read serially into the DATA_IN port on a clock-by-clock basis. Block size clock cycles are, therefore, required to input each block. DATA_IN may be qualified by the optional ND port. See "New Data (ND)."

New Data (ND)

The optional ND signal is used to indicate that there is new input data to be read from the DATA_IN port. For example, if the input block size is 100, then 100 active High *ND-samples* are required to load a block of data into the encoder. ND is also used to qualify the FD_IN input. See "First Data (FD_IN)."

First Data (FD_IN)

FD_IN is a mandatory input port which is used to start the encoder operation. FD_IN is qualified by the optional ND input. If ND is not selected, a *valid-FD* means simply that FD_IN is sampled High on an active rising clock edge. If ND is selected, a *valid-FD* means that FD_IN and ND are both sampled High on an active rising clock edge.

When a *valid-FD* occurs, the first data is read from the DATA_IN port, and the block size is read from the BLOCK_SIZE port. The core then continues loading data, until a complete block has been input.

The FD_IN input should only be asserted when the RFFD output is High. See "Ready For First Data (RFFD)."

Block Size (BLOCK_SIZE)

This port determines the size of the block of data that is about to be written into the encoder. The block size value is sampled on an active rising clock edge when FD_IN is High and ND (if selected) is High. If an invalid block size, (that is, not in the range 40-5114) is sampled, the behavior of the core is not specified.

Ready For First Data (RFFD)

When this signal is asserted High, it indicates that the core is ready to accept an FD_IN signal to start a new encoding operation. When a valid-FD signal is sampled, the RFFD signal goes Low and remains Low until it is safe to start another block.

Ready For Data (RFD)

When this optional pin is asserted High, it indicates that the core is ready to accept new input data. If RFD is selected, then it is High during the period that a particular block is input. When *block size* samples of data have been input, the RFD signal goes Low to indicate that the core is no longer ready to accept data.

RSC1 Systematic Output (RSC1_SYSTEMATIC)

RSC1_SYSTEMATIC is a mandatory output port, which is a delayed version of the uninterleaved input data.

During trellis termination, the RSC1_SYSTEMATIC port also carries systematic and parity tail bits.

RSC1 Parity0 Output (RSC1_PARITY0)

RSC1_PARITY0 is a mandatory output port, which is the Y0 output from RSC1. (See Figure 2.)

During trellis termination, the RSC1_PARITY0 port also carries systematic and parity tail bits.

RSC1 Parity1 Output (RSC1_PARITY1)

RSC1_PARITY1 is an optional output port, which is the Y1 output from RSC1. This port is enabled if Code Rate 1/5 is enabled. (See Figure 2.)

RSC2 Systematic Output (RSC2_SYSTEMATIC)

RSC2_SYSTEMATIC, if enabled, is a delayed and interleaved version of the input data. This port is disabled when the Multiplexed Tail Bits option is selected

RSC2 Parity0 Output (RSC2_PARITY0)

RSC2_PARITY0 is a mandatory output port, which is the Y0 output from RSC2. (See Figure 2.)

During trellis termination, the RSC2_PARITY0 port also carries systematic and parity tail bits.

RSC2 Parity1 Output (RSC2_PARITY1)

RSC2_PARITY1 is an optional output port, which is the Y1 output from RSC2. This port is enabled if Code Rate 1/5 is enabled. (See Figure 2.)

RSC1 Tail Output (RSC1_TAIL)

RSC1_TAIL is a mandatory output port, which is asserted High at the end of each output block to indicate trellis termination of RSC1. RSC1_TAIL goes High for two clock cycles for multiplexed tail bits, or three clock cycles for non-multiplexed tail bits.

RSC2 Tail Output (RSC2_TAIL)

RSC2_TAIL is a mandatory output port, which is asserted High at the end of each output block to indicate trellis termination of RSC2. RSC2_TAIL goes High for two clock cycles for multiplexed tail bits, or three clock cycles for non-multiplexed tail bits.

Ready (RDY)

This signal is asserted High when there is valid data on the SYSTEMATIC and PARITY output ports, including the tail bits. RDY is asserted for block size+four clock cycles for multiplexed tail bits, or block size+six clock cycles for non-multiplexed tail bits.

Ready For Data In (RFD_IN)

RFD_IN is an optional pin which can be used to inhibit the output of the core while allowing current input operations to continue. RFD_IN is intended to be used as a means for a downstream system using the encoded data to indicate that it is not ready to handle any further data. If RFD_IN is Low, all systematic and parity outputs, RSC1_TAIL, RSC2_TAIL, and RDY outputs are inhibited. The operation of RFD_IN is described in further detail later in this document.

Multichannel Operation

Up to 16 simultaneous data channels may be supported.

If multichannel operation is selected, then the width of the DATA_IN, SYSTEMATIC and PARITY ports are increased to *N*, where *N* is the number of data channels. Multichannel mode assumes that the *N* channels are of identical block size and are block-aligned. Control signals CE, ND, FD_IN, and RFD_IN have the same effect on all *N* channels. If the external RAM option is selected, the external memory data input and output ports also have width N.

Functional Description

Double-Buffering

Figure 4 illustrates the LOAD and OUTPUT operations.

Data blocks are written alternately to the two symbol memories, MEM0 and MEM1, and also read out alternately, thereby maximizing the data throughput.

If the block size increases, or the core is inhibited by negating ND, the LOAD operation may take longer than the OUTPUT operation, in which case the RDY output port may be driven low to indicate the OUTPUT operation is complete. Similarly, If the block size decreases, or the core is inhibited by negating RFD_IN, the OUTPUT operation may take longer than the LOAD operation, in which case the RFD port may be driven low to indicate that the LOAD operation is complete, and the core is no longer ready for data. The effect of changing the block size is shown in Figure 4(a), and the effects of ND and RFD_IN are shown in Figure 4(b).

Due to pruning (omitted addresses) in the internal interleaver, RDY will go Low for a few cycles between blocks; however, once an OUTPUT operation has started, RDY will only go Low again after the last tail bit has been output.

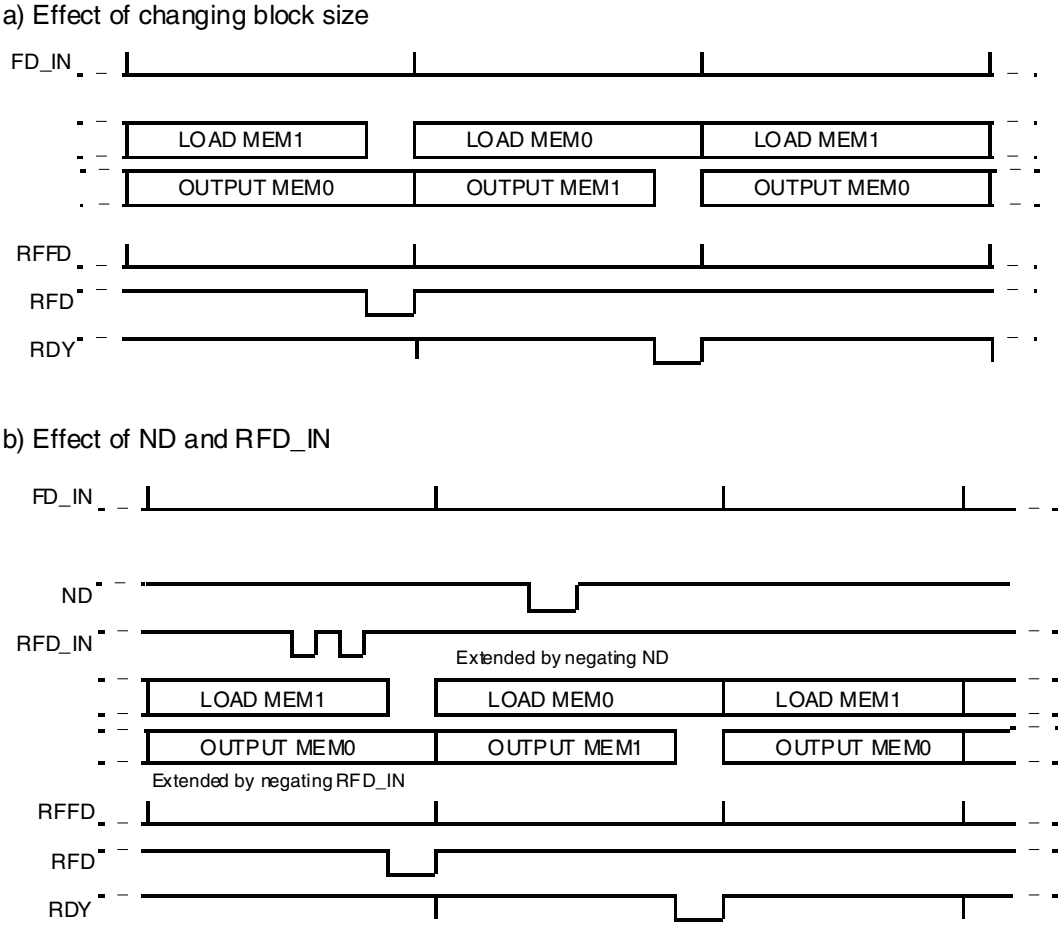


Figure 4: **Double-Buffered Mode Data Throughput**

Input Control Signals

Figure 5 shows the signals associated with the data input side of the core. If, on an active rising edge of CLK, FD_IN and ND (if selected) are both sampled High, this is known as a valid-FD, or valid First Data signal.

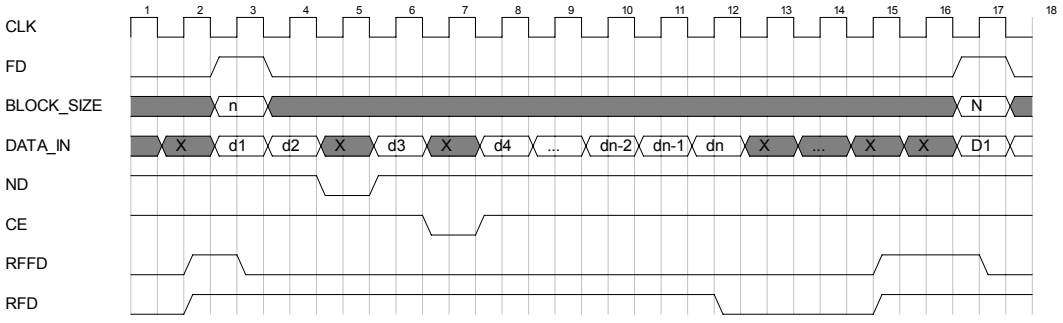


Figure 5: **Input Timing**

When a valid-FD is sampled, the RFFD signal is driven Low to indicate that the core is no longer waiting for FD_IN. The block size, in this case *n*, of the current input block is sampled on the BLOCK SIZE port, and the first data symbol, *d1*, is sampled on the DATA_IN port.

A High on the ND port indicates that the value on the DATA_IN port is new data. If ND is sampled Low, then the DATA IN port is not sampled, and the internal write address does not advance.

The core continues to input data, until *n* new data samples have been accepted, whereupon RFD (if selected) is normally driven Low to indicate that the core is no longer ready for data, and the core stops sampling the DATA_IN port.

When the core is ready to accept a new block of data, RFFD and RFD are both driven High. The time at which this occurs is determined by how long it takes the core to output the current output block, and this depends on its block size, and on whether or not the output side of the core is inhibited by negating RFD_IN. If the input operation completes and the output cycle is already complete, it is possible for RFFD to be asserted, and a new input cycle to be started without RFD going Low.

After asserting RFFD, the core waits until the next valid-FD is sampled, whereupon a write cycle is started, in this case, with block size *N*.

The behavior of the core is not specified if, on a valid-FD, an invalid block size is sampled, or RFFD is sampled Low (core not ready for a new block). If either of these conditions occurs, the core must be reset by asserting SCLR.

Trellis Termination

When the Multiplexed Tail Bits option is selected, as is the case for 3GPP systems, the trellis termination bits of RSC1 and RSC2 are remultiplexed, over four clock cycles, onto the RSC1_SYSTEMATIC, RSC1_PARITY0 and RSC2_PARITY0 output ports. These bits are then transmitted in the following sequence:

$$x_{n+1},\ z_{n+1},\ x_{n+2},\ z_{n+2},\ x_{n+3},\ z_{n+3},\ x'_{n+1},\ z'_{n+1},\ x'_{n+2},\ z'_{n+2},\ x'_{n+3}\ ,z'_{n+3}.$$

When the Multiplexed Tail Bits option is not selected, the RSC2 tail bit data is switched to the RSC1_SYSTEMATIC output and, therefore, overrides the RSC1_SYSTEMATIC values. Typically, the parallel output of the systematic and parity bits from each encoder are serialized before transmission. Also, in general, the RSC2_SYSTEMATIC data is only transmitted during the tail bit period, so by multiplexing the RSC2_SYSTEMATIC data onto the RSC1_SYSTEMATIC port, there is no need to output RSC2_SYSTEMATIC.

Output Control Signals

The RDY signal is driven High to indicate that there is valid data on the systematic and parity ports. In addition, the RSC1_TAIL and RSC2_TAIL outputs are provided to indicate trellis termination of RSC1 and RSC2, respectively. A High on RSC1_TAIL indicates that RSC1 tail bits are being output, and a High on RSC2_TAIL indicates that RSC2 tail bits are being output. The output timing for 3GPP compliant multiplexed tail bits is shown in Figure 6. Output timing for optional parallel output of tail bits is shown in Figure 7.

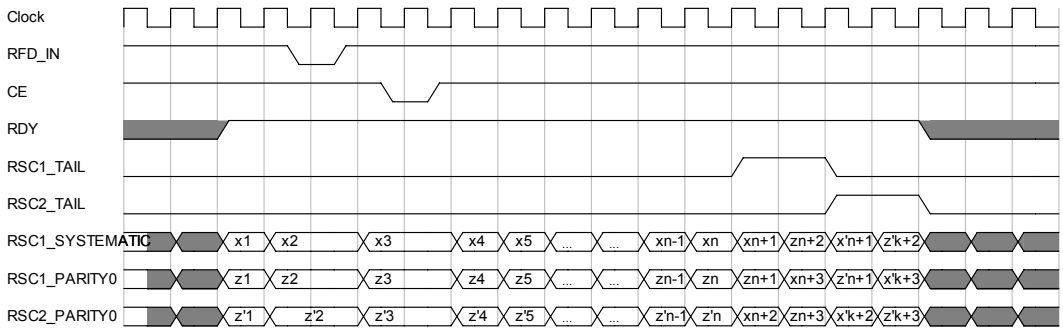
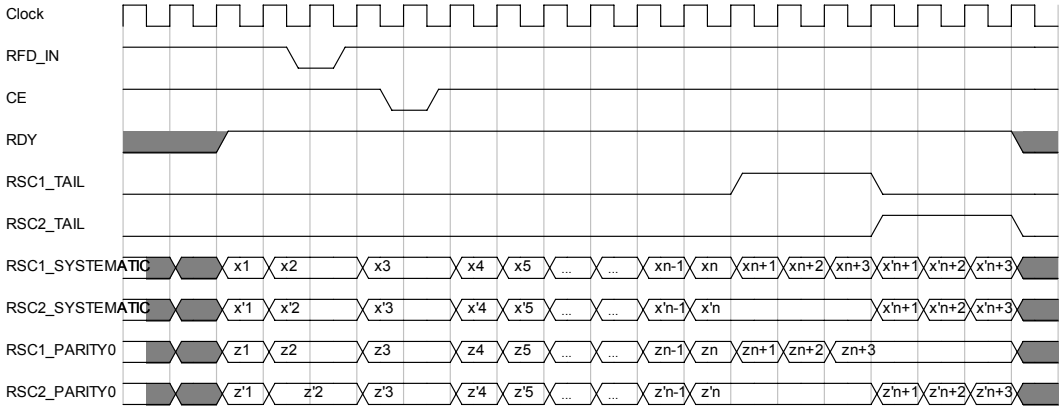


Figure 6: **Output Timing (Multiplexed Tail Bits)**



x1...xn are the uninterleaved input bits, delayed, for block size n
x'1...x'n are the interleaved input bits
z1...zn+3 are the RSC1 Parity0 output bits
z'1...z'n+3 are the RSC2 Parity0 output bits

Figure 7: **Output Timing (Non-Multiplexed Tail Bits)**

Flow control on the output side can be implemented with the optional RFD_IN input port. If the RFD_IN port is sampled Low on an active rising clock edge, the RSC output ports, RDY, and the internal circuitry associated with these outputs are frozen.

The input side of the core is not directly affected by the RFD_IN port. However, if RFD_IN is deasserted often enough, the time taken to output a block can be extended such that the assertion of RFFD is delayed, which acts to prevent overrun on the input side.

If rate 1/5 is selected, the RSC1_PARITY1 and RSC2_PARITY1 outputs are enabled. These have the same timing as RSC1_PARITY0 and RSC2_PARITY0, respectively.

External Address Generator Option

The I/O ports for the external address generator option are described in Table 2 and should be connected as shown in Figure 8.

Table 2: **I/O Ports for External Address Generator Option**

Pin	Sense	Port Width (bits)	Description
EXT_IADDR	Input	min 6 max 13	**External Interleaved Address -** May be configured to have any width from 6 through 13 bits.
EXT_IADDR_RFFD	Input	1	**External Address Generator Ready For FD** - Indicates that the external address generator has finished processing the current block, and is ready to start calculating the addresses for a new block. If the external address generator needs time to calculate the first interleaved address of a block, EXT_IADDR_RFFD may be asserted early to achieve maximum throughput.
EXT_IADDR_RDY	Input	1	**External Interleaved Address Ready** - Indicates that a valid address is on the EXT_IADDR port.
EXT_IADDR_CE	Output	1	**External Address Generator Clock Enable** - Indicates that the encoder is ready to receive an address from the external address generator. It depends on the state of the encoder and the CE and RFD_IN inputs, if selected.

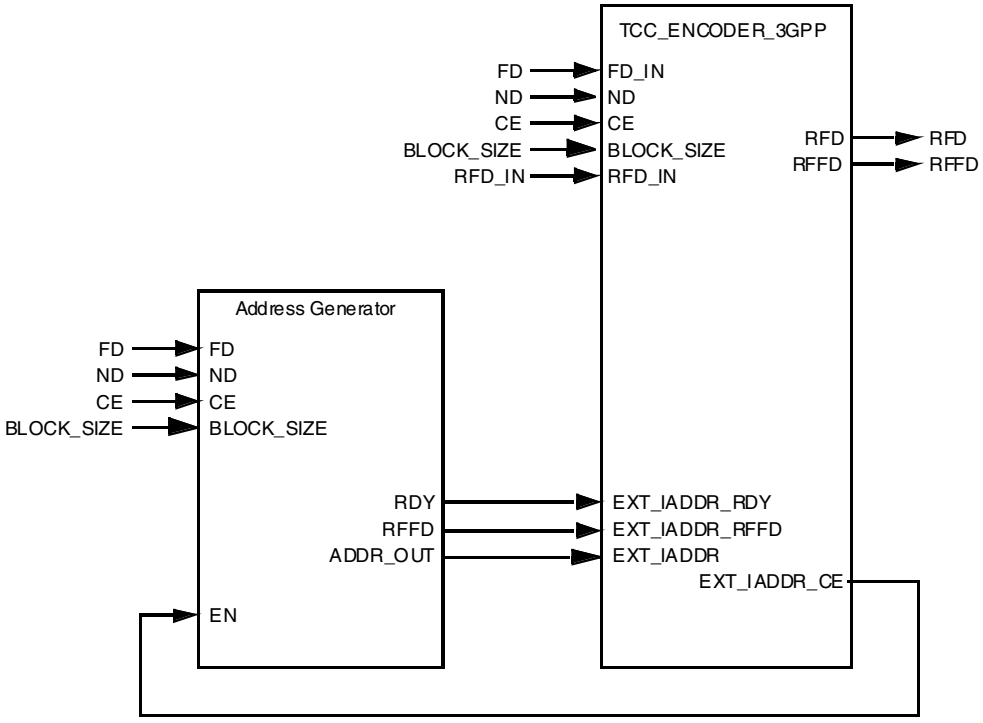


Figure 8: **External Address Generator Option**

External RAM Option

I/O Ports

The I/O ports for the external RAM option are summarized in Table 3 and should be connected as shown in Figure 9.

Table 3: **I/O Ports for External RAM Option**

Pin	Sense	Port Width (bits)	Description
MEM_DINA	Output	1 to 16	**Data inputs, Port A -** Common to both external RAMs.[1]
MEM_ADDRB	Output	6 to 13	**Port B address** - Common to both external RAMs. This is the Interleaved read address.[2]
MEM0_ENA	Output	1	Port A Enable, MEM0
MEM0_ENB	Output	1	Port B Enable, MEM0
MEM0_WEA	Output	1	**Port A Write Enable, MEM0**[3]
MEM0_ADDRA	Output	6 to 13	**Port A address for MEM0 -** Alternately carries the write address and the uninterleaved read address.[2]
MEM0_DOUTA	Input	1 to 16	**Data Outputs, Port A -** Uninterleaved systematic data for RSC1. Active when outputting from MEM0.[1]
MEM0_DOUTB	Input	1 to 16	**Data Outputs, Port B** - Interleaved systematic data for RSC2. Active when outputting from MEM0.[1]
MEM1_ENA	Output	1	Port A Enable, MEM1
MEM1_ENB	Output	1	Port B Enable, MEM1
MEM1_WEA	Output	1	**Port A Write Enable, MEM1**[3]
MEM1_ADDRA	Output	6 to 13	**Port A address for MEM1 -** Alternately carries the write address and the uninterleaved read address.[2]
MEM1_DOUTA	Input	1 to 16	**Data Outputs, Port A -** Uninterleaved systematic data, for RSC1. Active when outputting from MEM1.[1]
MEM1_DOUTB	Input	1 to 16	**Data Outputs, Port B** - Interleaved systematic data, for RSC2. Active when outputting from MEM1.[1]

Notes:

1. Width of memory data buses equals the number of channels.
2. Width of memory address buses is 13 if the address generator is internal, or is equal to the width of the external address generator.
3. Port B is read only, so the WEB ports of both external RAMs should be connected to logic 0, and the DINB ports may be left unconnected.

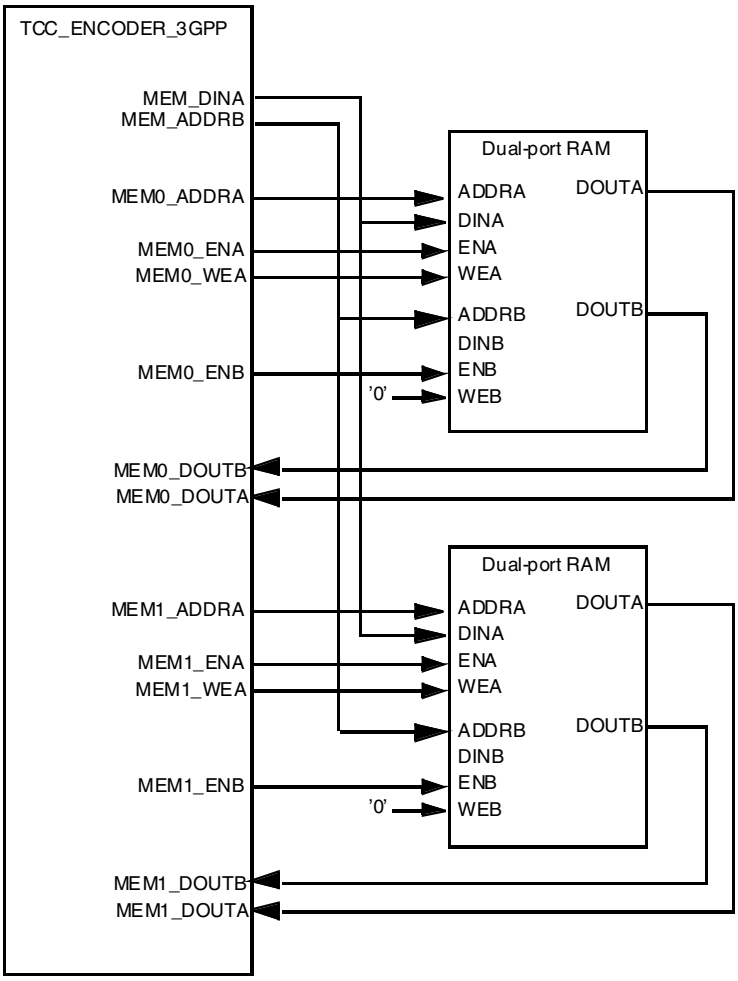


Figure 9: **External RAM Option**

Performance and Resource Usage

Table 4: **Performance and Resource Usage**

Description	Resource	Notes	Virtex-5 FPGA XC5VSX95T
Single channel, Internal RAM, Internal address generator. CE, SCLR, RFD_IN, ND, RFD, BLOCK_SIZE_VALID. Rate=1/3.	Area (LUT/FF pairs)	1	708
	LUTs/FFs	1	644/446
	Block Memories 36K/18K	3	1/3
	DSP48 blocks		4
	Speed -1/-3 (MHz)	1,2	255/339
	Latency for block size k. min/max (cycles)	4	k+4/67
Single channel, Internal RAM, Internal address generator. Rate=1/3.	Area (LUT/FF pairs)	1	649
	LUTs / FFs	1	549/432
	Block Memories 36K/18K	3	1/3
	DSP48 blocks		4
	Speed -1/-3 (MHz)	1,2	255/332
	Latency for block size k. min/max (cycles)	4	k+4/67
Single channel, External RAM, Internal address generator. Rate=1/3.	Area (LUT/FF pairs)	1	677
	LUTs / FFs	1	553/483
	Block Memories 36K/18K	3	1/3
	DSP48 blocks		4
	Speed -1/-3 (MHz)	1,2	227/310
	Latency for block size k. min/max (cycles)	4	k+4/69

Notes:

1. Area and maximum clock frequencies are provided as a guide. They may vary with new releases of Xilinx implementation tools, etc.
2. Maximum clock frequencies are shown in MHz for -1/-3 parts. Clock frequency does not take jitter into account and should be de-rated by an amount appropriate to the clock source jitter specification.
3. This is the total number of 36k block RAMs used when map was run. In reality, two 18k block RAM primitives can usually be packed together, giving an absolute minimum total block RAM usage of block RAMs (36k) + (block RAMs (18k)/2) (rounded up).
4. Latency is measured from the clock edge on which FD is sampled to the clock edge on which RDY is asserted. Minimum latency is for k > 70. Maximum latency is for k < 50.

References

1. 3G TS.25.222 V5.5.0 (2003-06), *Multiplexing and Channel Coding (TDD)*, Technical Specification Group Radio Access Network, 3rd Generation Partnership Project.
2. 3G TS.25.212 V6.0.0 (2003-12), *Multiplexing and Channel Coding (FDD)*, Technical Specification Group Radio Access Network, 3rd Generation Partnership Project.
3. *Near Shannon Limit Error-correcting Coding and Decoding Turbo Codes*, C. Berrou, A. Glavieux, and P. Thitimajshima, IEEE Proc 1993 Int Conf. Comm., pp1064-1070.

Evaluation

An evaluation license is available for this core. The evaluation version of the core operates in the same way as the full version for several hours, dependent on clock frequency. Operation is then disabled and the data output will not change. If you notice this behavior in hardware, it probably means you are using an evaluation version of the core. The Xilinx tools warn that an evaluation license is being used during netlist implementation. If a full license is installed in order for the core to run on hardware, delete the old XCO file and recreate the core from new.

Support

Xilinx provides technical support for this LogiCORE product when used as described in the product documentation. Xilinx cannot guarantee timing, functionality, or support of product if implemented in devices that are not defined in the documentation, if customized beyond that allowed in the product documentation, or if changes are made to any section of the design labeled *DO NOT MODIFY*.

Ordering Information

The Turbo Encoder core is provided under the SignOnce IP Site License and can be generated using the Xilinx ISE® CORE Generator v11.2 or higher. The CORE Generator software is shipped with Xilinx ISE Foundation™ Series Development software.

To access the full functionality of the core, including simulation and FPGA bitstream generation, a full license must be obtained from Xilinx. For more information, visit the Turbo Encoder product page.

Contact your local Xilinx sales representative for pricing and availability of additional Xilinx LogiCORE IP modules and software. Information about additional modules is available from the Xilinx IP Center.

France Telecom, for itself and certain other parties, claims certain intellectual property rights covering Turbo Codes technology, and has decided to license these rights under a licensing program called the Turbo Codes Licensing Program. Supply of this IP core does not convey a license nor imply any right to use any Turbo Codes patents owned by France Telecom, TDF or GET. Contact France Telecom for information about its Turbo Codes Licensing Program at the following address:

France Telecom R&D
VAT/TURBOCODES
38, rue du Général Leclerc
92794 Issy Moulineaux
Cedex 9
France

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
11/11/04	1.0	Initial Xilinx release
04/28/05	2.0	Updated for re-designed 3GPP Turbo Encoder (v2.0)
09/28/06	3.0	Updated to version 3.0 standards and Xilinx tools 8.2i.
03/24/08	3.1	Updated to version 3.1 standards and Xilinx tools 10.1i.
06/24/09	4.0	Updated to version 4.0 and Xilinx tools 11.2i

Notice of Disclaimer