Viterbi Decoder

February 8, 1998 | Product Specification

CAST, Inc.

24 White Birch Drive
Pomona, New York 10907 USA
Phone: +1 914-354-4945
Fax: +1 914-354-0325
E-Mail: info@cast-inc.com
URL : www.cast-inc.com

Features

- Hard Decision Decoder
- Trace-back method for survivor memory
- Branch Metrics computations can be added by CAST for different applications
- Parameterized architecture allows for customization of the following functions:
  - Number of states in the trellis
  - Number of bits to represent transition values
  - Add-Compare-Select (ACS) cells
  - The length of the trace-back
  - The length of the received burst
  - The initial path metric for state 0
  - The survivor memory (RAM) word length

Applications

- Wireless telecommunication (satellites, digital cellular phones, etc.)
- Consumer electronics (CD players, etc.)

AllianceCORE™ Facts

Core Specifics[1]		
Device Family	Spartan	XC4000XL
CLBs Used	425	425
IOBs Used	109[2]	109[2]
System Clock f_{max}	11.55 Mhz	19.98 Mhz
Device Features Used	Global Buffers	

Supported Devices/Resources Remaining[1]		
	I/O	**CLBs**
XC4013XL-09	82[2]	151
XCS30-3	82[2]	151

Provided with Core	
Documentation	Core Documentation Sample Decoder
Design File Formats	.ngo or XNF Netlist or VHDL Source RTL
Constraint Files	NCF
Verification Tool	VHDL
Schematic Symbols	Viewlogic
Evaluation Model	None
Reference designs & application notes	Sample Design
Additional Items	None

Design Tool Requirements	
Xilinx Core Tools	M1.3
Entry/Verification Tool	VHDL RTL

Support
Support provided by CAST, Inc.

Notes:
1. Results obtained for the parameters in the table below.
2. Assuming all core signals are routed off-chip.

Parameter	Value
Number of States in trellis	16
Number of bits to represent transition values	8
Add-Compare-Select (ACS) cells	4
Length of the trace-back	30
Length of the received burst	5
Initial path metric for state 0	-4
Survivor memory (RAM) word length	8

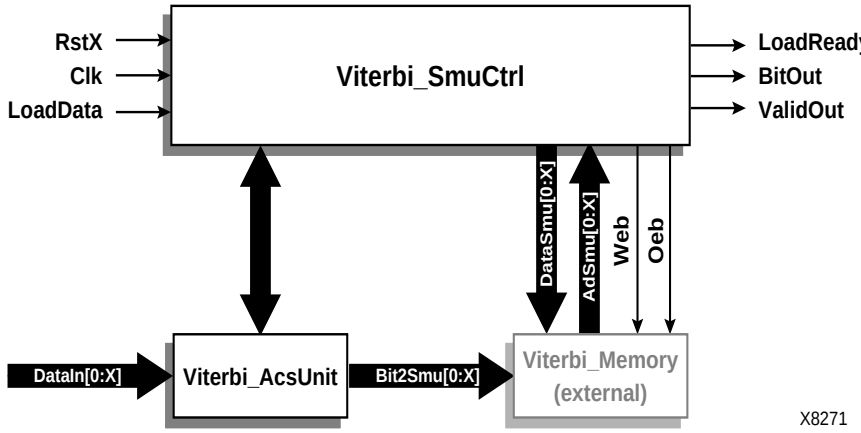

Figure 1: Viterbi Block Diagram

General Description

A Viterbi decoder is used for decoding convolutional codes. It can also be used to produce the maximum likelihood estimate of the transmitted sequence over a channel with inter-symbol interference (ISI).

The source code version of the core is parameterizable and therefore easily adapted to a wide variety of applications. The data given in this fact sheet is for an example only. The Viterbi Decoder can be delivered in a specific format (see Ordering Information).

Functional Description

The Viterbi Decoder is partitioned into modules as shown in Figure 1 and described below.

Viterbi_AcsUnit

ACS (add-compare-select) unit calculates the path metrics to find the minimum path. The number of AcsUnits is parameterizable.

Viterbi_SmuCtrl

SmuCtrl block handles the survivor memory management. The state-machine controls the alternation of reading new branch metrics and trace-back. During trace-back it reads the decision values from the memory and outputs the decoded bits. It reconstructs the actions of the encoder in the reversed order by updating the State-register with a decision value pointed by the former State-value. As a result of it, the decoded bits are also output in reverse order.

Viterbi_Memory

Viterbi memory is a RAM memory which stores the trace-back values during calculation. This block is external to the core and is added by the user or can be included by CAST for additional cost.

Core Modifications

The Viterbi Decoder core can be customized to include:

- Number of states in the trellis
- Number of bits to represent transition values
- Add-Compare-Select (ACS) cells
- The length of the trace-back
- The length of the received burst
- The initial path metric for state 0
- The survivor memory (RAM) word length
- Addition of Viterbi Memory block

Please contact CAST directly for any required modifications. Use the fax-in checklist at the end of this specification.

Pinout

The pinout of the Viterbi Decoder core has not been fixed to specific FPGA I/O, allowing flexibility with a users application. Signal names are shown in the block diagram in Figure 1, and in Table 1.

Verification Methods

The core model has been extensively tested using various settings of the parameters. A special encoder circuit is used as a testbench for full testing of the decoder.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
Input Interface Signals		
RstX	Input	Asynchronous reset, active low
Clk	Input	Common Clock
LoadData	Input	Enables Data Loading
DataIn[0:X]	Input	Transition values input, the width depends on the number of ACS-units
Output Interface Signals		
LoadReady	Output	Indicates Ready for Data
BitOut	Output	Decoded Output Bit
ValidOut	Output	Indicates the Valid Output Data
External Memory Interface Signals		
Bits2Smu[0:X]	Output	Output data to external Memory
DataSmu[0:X]	Input	Input data from external Memory
Web	Output	Write Enable to external Memory
Oeb	Output	Read Enable to external Memory
AdSmu[0:X]	Output	Address to external Memory

Recommended Design Experience

The user must be familiar with HDL design methodology as well as instantiation of Xilinx netlists in a hierarchical design environment.

Ordering Information

The Viterbi Decoder is available for purchase directly from CAST, Inc. If the XNF Netlist format is required, a setting for the parameters will have to submitted before the netlist can be created. The Viterbi core is licensed from Hantro Products Oy. Ask CAST, Inc. for details.

Related Information

A.B. Carlson, Communications systems, McGraw-Hill, 1986.

E.A. Lee, D.G. Messerschmitt, Digital Communication, Boston, MA Kluwer Academic Publishers, 1988.

G.D. Forney, Jr., “The Viterbi algorithm:, Proc. IEEE, vol. 61, pp.268-277, March 1973.

G. Fettweis, H. Meyr, “High-Speed Parallel Decoding: Algorithm and VLSI-Architecture”, IEEE Comm. Magazine, pp. 46-55, May 1991.

G. Feygin, P.G.Gulak, “Architectural Tradeoffs for Survivor Sequence Memory Management in Viterbi decoders”, IEEE Trans. Commun. Tech., vol. 41, no. 3, March 1993.

R. Cypher, C.B.Shung, “Generalized Trace-Back techniques for Survivor Memory Management in the Viterbi Algorithm”, Journal of VLSI Signal Processing, 5, 85-94 (1993).

Motorola, “Convolutional Encoding and Viterbi Decoding Using the DSP56001 with a V.32 Modem Trellis Example, Motorola Inc. 1989.

B. Sklar, “Digital Communications, fundamentals and applications”, Prentice Hall International Inc. 1988.

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
+1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/tblpart.htm

Viterbi Implementation Request Form

To: **CAST, Inc.**
FAX: **+1 914-354-0325**
E-mail: **info@cast-inc.com**

From: _____

Company: _____

Name: _____

Address: _____

Country: _____

Phone: _____

Fax: _____

E-mail: _____

CAST, Inc. will customize and deliver a Xilinx optimized netlist for the specific Viterbi Decoder implementation you require. Please fill out and fax or email this information to CAST so they can provide you with an accurate response to, and quote for your requirements.

Implementation Issues

1. Number of states in trellis: _____

2. Number of bits to represent transition values: _____

3. Add-compare-select cells: _____

4. Length of trace-back: _____

5. Length of receiveR burst:_____

6. Initial path metric for state 0: _____

7. Survivor memory word length: _____

Business Issues

8. Indicate timescales of requirement:
   a. date for decision _____
   b. date for placing order _____
   c. required delivery date _____

9. Indicate your area of responsibility:
   _____ decision maker
   _____ budget holder
   _____ recommender

10. Has a budget been allocated for the purchase?
    Yes _____ No _____

11. What volume do you expect to ship of the product that will use this core? _____

12. What major factors will influence your decision?
    _____ cost
    _____ customization
    _____ testing
    _____ implementation size

13. Are you considering any other solutions?