Decoder Module

Description:

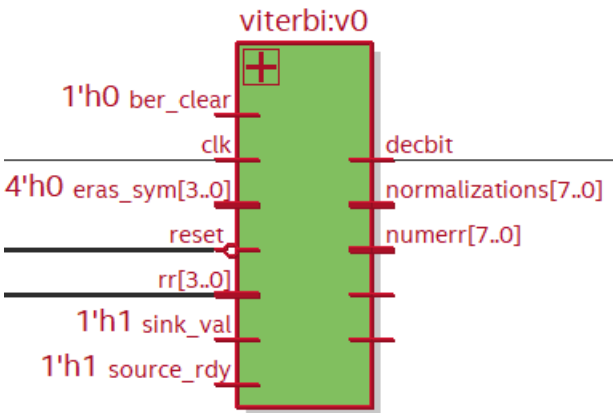

Figure 1: Viterbi decoder

The Viterbi decoder is matched to the convolutional encoder's trellis structure. Hard decision decoding is chosen to simplify the decoding process by making binary decisions on received bits. A traceback depth of 32 is selected to balance decoding accuracy and complexity. Continuous mode ensures that the decoder processes the input stream consistently and the architecture is parallel for faster decoding to reduce computational delays. Additionally, soft bits of 1 mean hard decision decoding. The IP catalog below shows the parameters selected for the Viterbi decoder.

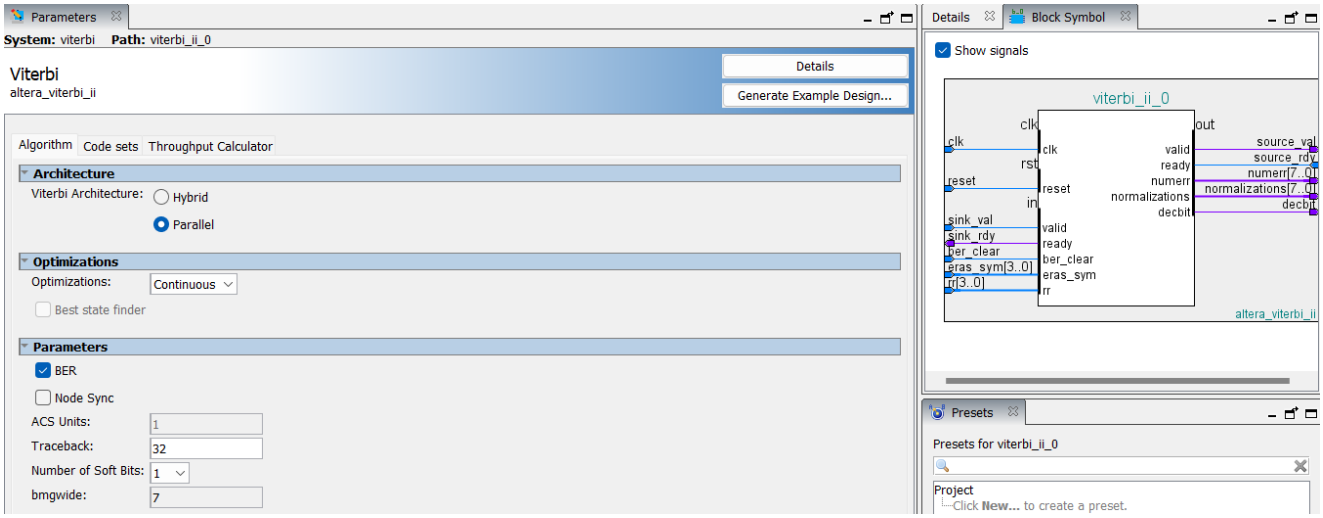

Figure 2: IP catalog for Viterbi decoder

The RTL diagram below shows a detailed implementation. At each clock cycle of 32 MHz clock frequency the Viterbi module takes 4 bits of input and outputs 1 bit.

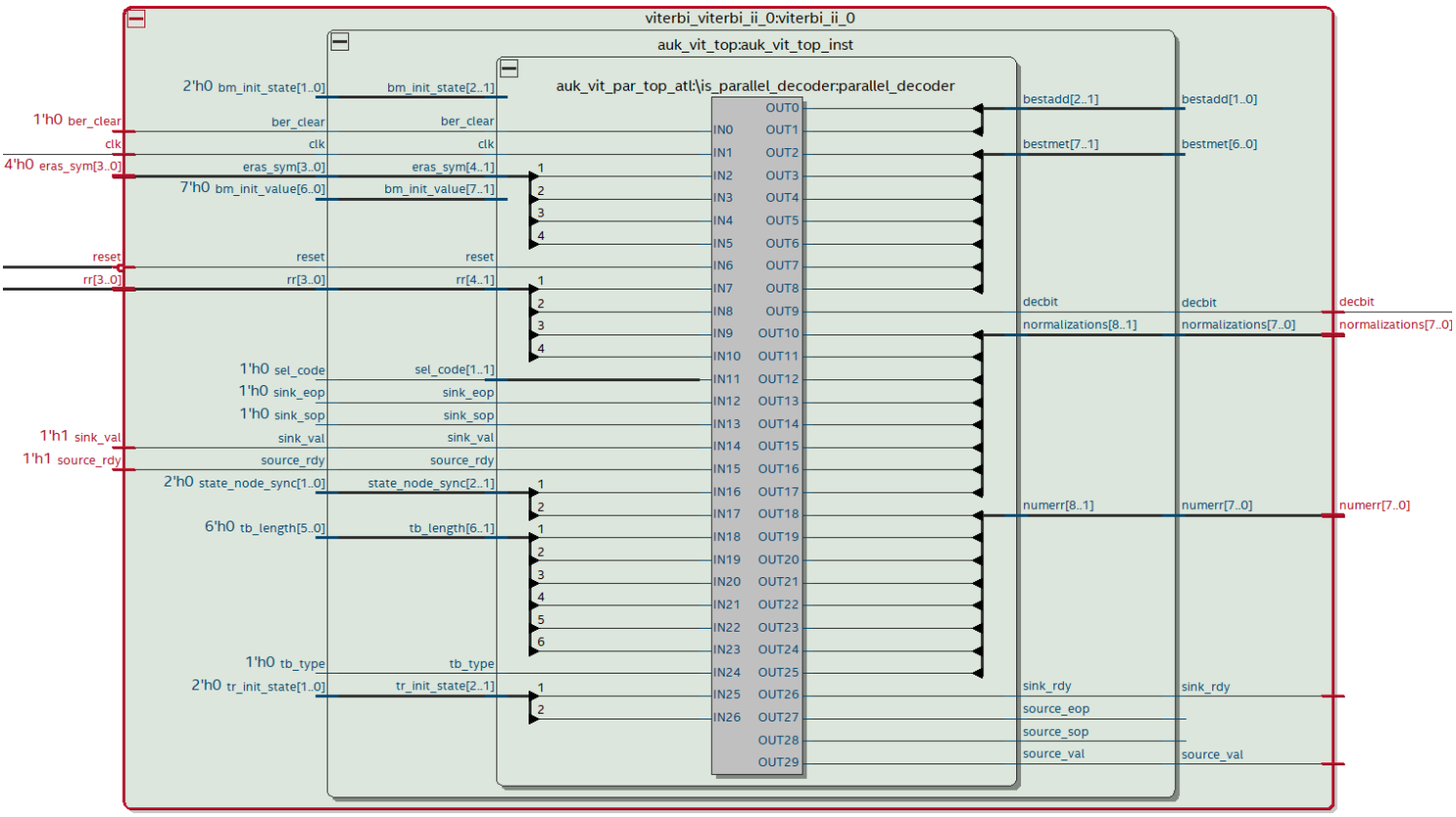

Figure 3: RTL diagram of Viterbi decoder

The IP core guide gave all the information on the signals involved and which files have the template to instantiate the Viterbi module and the black box module. The input signal rr is the input which is connected to the demod_bits port. The signal numrr counts the number of errors occurred and is outputted by the module. Images below from the user guide describe signals that I assigned values accordingly.

Signal Name	Avalon-ST Name	Direction	Description
eras_sym[Nmax:1]	dat	Input	When asserted, eras_sym Indicates an erased symbol. Both rr and eras_sym are Avalon-ST dat inputs
rr	dat	Input	Data input, which takes in *n* symbols, each softbits wide per clock. In TCM mode the rr width is (2*N* × *softbits*:1); in Viterbi mode the rr width is (*nmax* × *softbits*:1).Both rr and eras_sym are Avalon-ST dat inputs
sink_eop	eop	Input	End of packet (block) signal. sink_eop delineates the packet boundaries on the rr bus. When sink_eop is high, the end of the packet is present on the dat bus. sink_eop is asserted on the last transfer of every packet. This signal applies to block decoding only.
sink_rdy	ready	Output	Data transfer enable signal.The interface sink drives sink_rdy and controls the flow of data across the interface. sink_rdy behaves as a read enable from sink to source. When the source observes sink_rdy asserted on the clk rising edge, it can drive the Avalon-ST data interface signals and assert sink_val as early as the next clock cycle, if data is available. In the hybrid architecture, sink_rdy is asserted for one clock cycle at a time. If data is not available at the time, you have to wait for the next sink_rdy pulse.
sink_sop	sop	Input	Start of packet (block) signal. sop delineates the packet boundaries on the rr bus. When sink_sop is high, the start of the packet is present on the rr bus. sink_sop is asserted on the first transfer of every packet This signal applies to block decoding only.
sink_val	val	Input	Data valid signal. sink_val indicates the validity of the data signals. sink_val is updated on every clock edge where sink_rdy is sampled asserted, and holds its current value along with the dat

continued...

Signal Name	Avalon-ST Name	Direction	Description
			bus where sink_rdy is sampled deasserted. When sink_val is asserted, the Avalon-ST data interface signals are valid. When sink_val is deasserted, the Avalon-ST data interface signals are invalid and you must disregard them. To determine whether new data has been received, the sink qualifies the sink_val signal with the previous state of the sink_rdy signal.
sink_data	data	Input	In Qsys systems, this Avalon-ST-compliant data bus includes all the Avalon-ST input data and configuration signals. The signals are in the following order from MSB to LSB: • In • State_node_sync • Ber_clear • Sel_code • Tb_type • Tb_length • Tr_init_state • Bm_init_state • Bm_init_value • Eras_symRr

Signal	Avalon-ST Name	Direction	Description
decbit	dat	Output	The decbit signal contains output bits when source_val is asserted.
source_eop	eop	Output	End of packet (block) signal. if you select continuous optimization, this signal is left open and you must remove it from the testbench.
source_rdy	ready	Input	Data transfer enable signal. The sink interface drives `source_rdy` and uses it to control the flow of data across the interface. `ena` behaves as a read enable from sink to source. When the source observes `source_rdy` asserted on the `clk` rising edge it drives, on the following `clk` rising edge, the Avalon-ST data interface signals and asserts `source_val`. The sink captures the data interface signals on the following `clk` rising edge. If the source is unable to provide new data, it deasserts `source_val` for one or more clock cycles until it is prepared to drive valid data interface signals.
source_sop	sop	Output	Start of packet (block) signal. if you select continuous optimization, this signal is left open and you must remove it from the testbench.
source_val	val	Output	Data valid signal. The IP core assers `source_val` high for one clock cycle, whenever there is a valid output on the `decbit` signal.
out_data	data	Output	In Qsys systems, this Avalon-ST-compliant data bus includes all the Avalon-ST output data and configuration signals. The signals are in the following order from MSB to LSB: • `Numerr` • `BestAdd` • `BestMet` • `Normalizations` • `Decbit`