VA08S Features

- 64 or 256 state (constraint length 7 or 9) Viterbi decoder
- Up to 213 MHz internal clock
- Rate 1/2, 1/3, or 1/4 (inputs can be punctured for higher rates)
- 4–bit received signed magnitude data
- Up to length 64 block decoding including tail
- Estimated channel bit error outputs
- 725 slices and 4 BlockRAMs (Virtex, Virtex–E, Spartan–II, Spartan–IIE), 670 slices and 1 BlockRAM (Virtex–II, Virtex–II Pro), 707 slices and 1 BlockRAM (Spartan–3, Virtex–4)
- Asynchronous logic free design
- Available as EDIF core and VHDL simulation core for Xilinx Virtex, Virtex–E, Spartan–II, Spartan–IIE, Virtex–II, Virtex–II Pro, Spartan–3 and Virtex–4 FPGAs under SignOnce IP License
- Available as VHDL core for ASICs
- Low cost university license also available

64 State Features

- Up to 17.6 Mbit/s decoding speed
- Optional or standard code polynomials

256 State Features

- Up to 5.3 Mbit/s decoding speed
- Optional or 3GPP™/3GPP2 code polynomials

Introduction

The VA08S is a 64 or 256 state error control decoder using the maximum likelihood Viterbi algorithm. The decoder is designed to efficiently decode short block lengths with maximum flexibility, allowing it to decode various communications standards, as well as custom coding solutions.

The VA08S uses eight add–compare–select (ACS) circuits in parallel 8 or 32 times to decode 64 or 256 state convolutional codes, respectively. A single external 2Kx8 synchronous RAM (implemented with four 2Kx2 BlockRAMs in Virtex/ Virtex–E/Spartan–II/Spartan–IIE or one 2Kx8 BlockRAMs in Virtex–II/Spartan–3) is used to perform the traceback. Smaller memories can be used with 64 state only operation. In synchronous operation, 10 or 34 clock cycles are required per received symbol for 64 or 256 states, respectively.

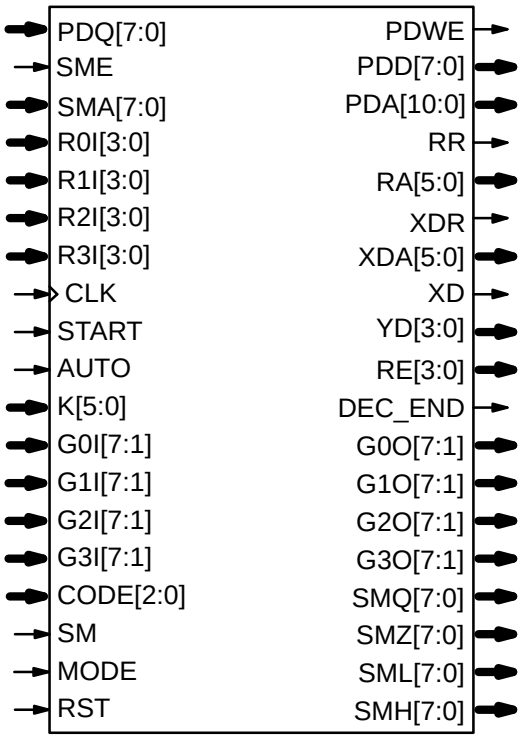


Figure 1: VA08S schematic symbol.

Asynchronous operation requires 11 or 35 clock cycles per received symbol for 64 or 256 states, respectively.

For 64 state, input data lengths from 1 to 58 bits can be decoded. For 256 state, input data lengths from 1 to 56 bits can be decoded.

The decoder works by first calculating the state metrics and storing all the path decision bits into memory. At the end of the block, a traceback is performed and the decoded bits are output in reverse order. This allows much faster decoding for short block lengths compared to a continuous decoder due to long decoder delays.

The decoder also outputs the zero–state, minimum non–zero–state and maximum non–zero–state state metrics at the end of decoding. The state metrics can also be read one at a time for diagnostic purposes at the end of the block.

Figure 1 shows the schematic symbol for the VA08S decoder. The EDIF core can be used with Xilinx Integrated Software Environment (ISE) software to implement the core in Xilinx FPGA's. The VHDL core can be used in ASIC designs.

Table 1 shows the performance achieved with various Xilinx parts. T_{cp} is the minimum clock period over recommended operating conditions. These performance figures may change due to device utilisation and configuration.

Table 1: Performance of Xilinx parts.

Xilinx Part	T_{cp} (ns)	K=7* (Mbit/s)	K=9* (Mbit/s)
XCV100E–6	12.931	6.40	1.93
XCV100E–7	11.549	7.17	2.17
XCV100E–8	9.165	9.04	2.73
XC3S200–4	10.133	8.17	2.47
XC3S200–5	8.844	9.36	2.83
XC2V250–4	8.470	9.78	2.95
XC2V250–5	7.367	11.25	3.40
XC2V250–6	6.297	13.16	3.98
XC2VP2–5	7.369	11.24	3.40
XC2VP2–6	6.598	12.55	3.79
XC2VP2–7	5.667	14.62	4.42
XC4VLX15–10	6.354	13.04	3.94
XC4VLX15–11	5.424	15.27	4.62
XC4VLX15–12	4.693	17.65	5.34

*Synchronous operation

Signal Descriptions

AUTO	Automatic Decoding
CLK	System Clock
CODE	Code Select (see Table 2)
DEC_END	Decoder End
G0I–G3I	Code Polynomial Input
G0O–G3O	Code Polynomial Output
K	Data Length 1 to 58 (64 state) 1 to 56 (256 state)
MODE	Maximum Rate Select 0 = Rates 1/2 to 1/3 1 = Rates 1/2 to 1/4
PDA	Path Decision Address
PDD	Path Decision Data
PDQ	Path Decision Input
PDWE	Path Decision Write Enable
R0I–R3I	Received Data
RA	Received Data Address
RE	Estimated Symbol Error
RR	Received Data Ready
RST	Synchronous Reset
SM	Code State Select 0 = 64 States (*K*=7) 1 = 256 States (*K*=9)
SME	State Metric Enable
SMA	State Metric Address
SMH	Maximum Non–Zero–State SM
SML	Minimum Non–Zero–State SM
SMQ	State Metric Output
SMZ	Zero–State State Metric (SM)
START	Decoder Start
XD	Decoded Data Output
XDA	Decoded Data Address
XDR	Decoded Data Ready
YD	Decoded Symbol Output

Code Selection

Figure 2 gives a block diagram of a 256 state (constraint length nine) non–systematic encoder. To decode 256 state encoded data, select SM=1. X is the data input and Y0 to Y3 are the coded outputs. GiIj = $g_i^j \in \{0, 1\}, 0 \le i \le 3, 1 \le j \le 7$, correspond to the code polynomial coefficients which are used by the decoder.

The encoder polynomials are defined as

$$g_i(D) = 1 + g_i^1 D + g_i^2 D^2 + ... + g_i^7 D^7 + D^8 \quad (1)$$

where D is the delay operator and + indicates modulo–2 (exclusive OR) addition. It is usual practice to express the coefficients in octal notation, e.g., $g_0 = 561_8 = 101110001_2 \equiv g_0(D) = 1 + D^2 + D^3 + D^4 + D^8$. This corresponds to G0I[7:1] = 0001110_2.

The code polynomial inputs G0I[7:1] to G3I[7:1] or the 3GPP™ [1] and 3GPP2 [2] convolutional code standards are selected by CODE[2:0]. The codes are given in Table 2 in octal notation. When CODE = 1 or 2, inputs R2I[2:0] and R3I[2:0] are internally grounded. When CODE = 3, inputs R3I[2:0] are internally grounded. G0O[7:1] to G3O[7:1] reflect the code polynomials that are selected.

Table 2: Convolutional Codes.

CODE	Standard – Rate	g0	g1	g2	g3
0	–	G0I	G1I	G2I	G3I
1	3GPP–1/2	561	753	–	–
2	3GPP2–1/2	753	561	–	–
3	3GPP–1/3 3GPP2–1/3	557	663	711	–
4	3GPP2–1/4	765	671	513	473
5	1/2	171	133	–	–
6	1/3	171	133	165	–
7	1/4	173	167	135	111

Figure 3 shows the 64 state (constraint length seven) encoder. To decode 64 state encoded data, select SM=0. To simplify decoder complexity the code polynomials are given by

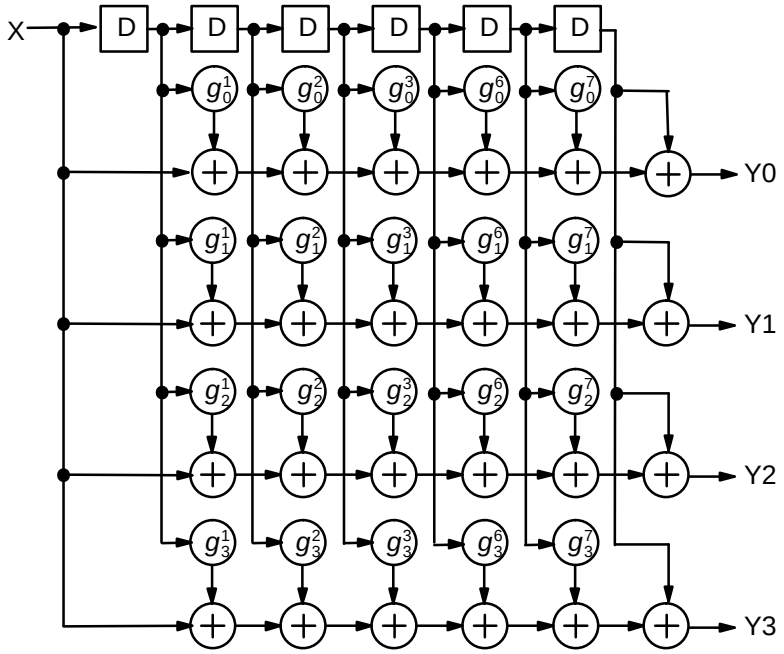


Figure 3: 64 state non–systematic convolutional encoder.

$$g_i(D) = 1 + g_i^1 D + g_i^2 D^2 + g_i^3 D^3 + g_i^6 D^4 + g_i^7 D^5 + D^6 \quad (2)$$

Note that GiI4 and GiI5 must be set to zero when 64 state mode is selected. For example, if g0 = 171, then G0I[7:1] = 0000111. Table 2 shows the 64 state codes selected for CODE = 5 and 6, corresponding to standard rate 1/2 and 1/3 convolutional codes. For CODE = 7 and rate 1/4, a code was selected from [3]. R2I and R3I are internally set to zero when CODE = 5 and R3I is internally set to zero when CODE = 6.

Viterbi Decoder

The Viterbi decoder is designed to be very flexible and can be operated in block mode only.

Theory of Operation

The Viterbi decoding algorithm [4] finds the most likely transmitted sequence given the received noisy sequence.

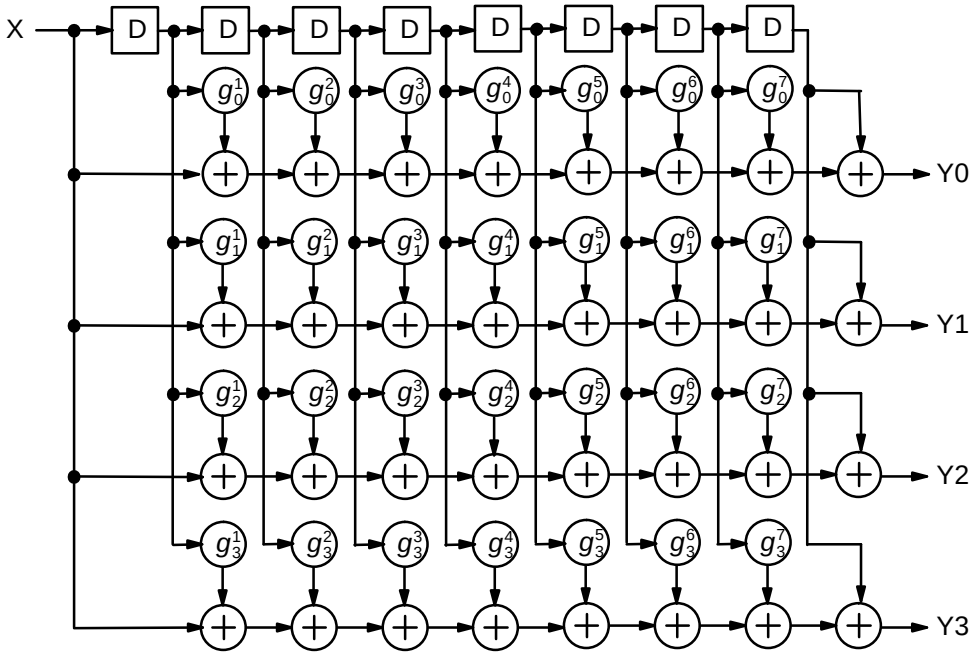


Figure 2: 256 state non–systematic convolutional encoder.

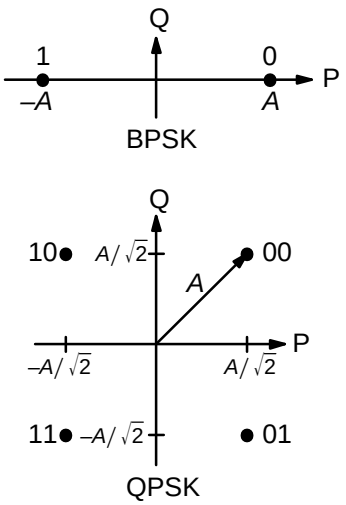


Figure 4: BPSK and QQPSK signal sets.

For binary phase shift keying (BPSK) or quadrature phase shift keying (QPSK) modulation the received signal is described by

$$R_k^i = A((1 - 2y_k^i)/\sqrt{m} + n_k^i) \quad (3)$$

where A is the signal amplitude, $y_k^i \in \{0, 1\}$, $i = 0$ to 3 correspond to the coded bits, $m = 1$ for BPSK or $m = 2$ for QPSK, and n_k^i is a Gaussian distributed random variable with zero mean and normalised variance σ^2. Figure 4 shows the signal sets for BPSK and QPSK. We have

$$\sigma^2 = \left(2mR\frac{E_b}{N_0}\right)^{-1} \quad (4)$$

where E_b/N_0 is the energy per bit to single sided noise density ratio and $R = k/n$ is the code rate (k is the number of information bits and n is the number of coded bits).

Since a zero is transmitted as $+A/\sqrt{m}$ and a one is transmitted as $-A/\sqrt{m}$ the sign bit of a noiseless R_k^0 in two's complement notation is equal to d_k.

Due to quantisation and limiting effects the value of A should also be adjusted according to the received signal to noise ratio. A program called *cmap* for calculating the optimum values of A is included with the cores.

The value of A directly corresponds to the 4–bit signed magnitude inputs (described in more detail later). The 4–bit inputs have 15 quantisation regions with a central dead zone. The quantisation regions are labelled from –7 to +7. For example, one could have $A = 3.7$. This value of A lies in quantisation region 3 (which has a range between 3 and 4).

Example 1: Rate 1/3 BPSK code operating at $E_b/N_0 = 3$ dB. From (4) we have $\sigma^2 = 0.75178$. Using cmap we have that $A = 4.11$.

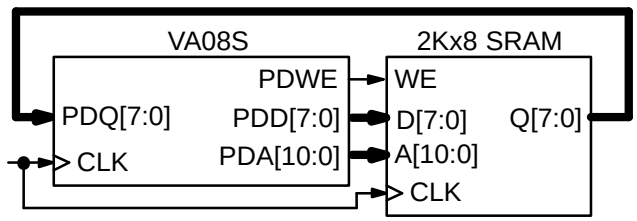
Example 2: Rate 1/2 QPSK code operating at $E_b/N_0 = 4$ dB. From (4) we have $\sigma^2 = 0.19905$. Using cmap we have that $A = 6.24$. Note that the amplitude in each dimension is $A/\sqrt{2} = 4.42$.

Decoder Operation

The path decision RAM is not included in the VA08S core. A 2Kx8 synchronous RAM is required and can be constructed from four RAMB4_S2 Virtex/Virtex–E/Spartan–II/Spartan–IIE BlockRAMs or one RAMB16_S9 Virtex–II/Spartan–3 BlockRAMs. PDWE, PDA[10:0], PDD[7:0] and PDQ[7:0] are the path decision write enable, address, data in and data out signals of the RAM. Figure 5 shows how to connect a 2Kx8 synchronous RAM to these signals. This allows the path decision memory to be reused for other applications, e.g., the interleaver memory of a 3G turbo decoder.

Address bits PDA[10:5] store the path decision bits for each received symbol. If the input data length is K, address bits range from PDA[5] to PDA[5+ $\lceil \log_2 K \rceil$] where $\lceil X \rceil$ rounds up X to the nearest integer. All other address bits are set to zero. For example, if the maximum K used is 32, then PDA[10] = 0. Thus, we can halve the traceback memory to 1Kx8. The minimum required memory has address depth 8K for 64 states and 32K for 256 states.

If 64 state mode is only selected then the external RAM can be reduced to size 512x8. Addresses PDA[4:3] are not used, so the address bus is



Inputs				Outputs	
WE	CLK	A[10:0]	D[7:0]	Q[7:0]	RAM(A)
0	↑	A	X	RAM(A)	RAM(A)
1	↑	A	D	X	D

Figure 5: Path decision RAM schematic and truth table.

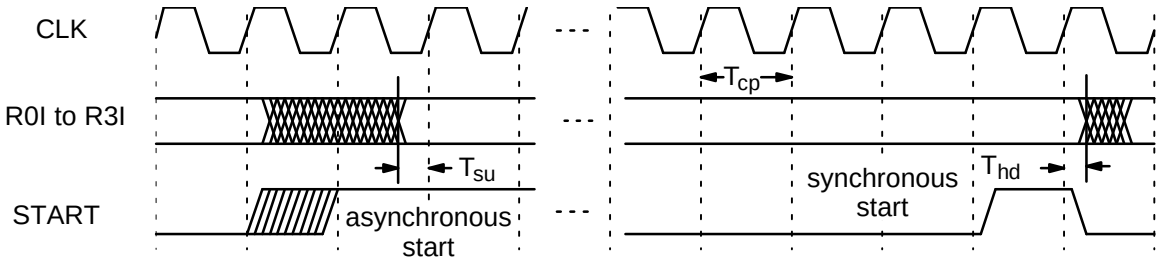


Figure 6: Asynchronous and synchronous Input timing.

PDA[10:5],PDA[2:0]. This can be implemented with one RAMB4_S8 BlockRAM.

The VA08S uses eight ACS circuits in parallel. Thus, 32 clock cycles are required to perform 256 ACS operations or 8 clock cycles for 64 ACS operations. An additional 2 clock cycle overhead is also required.

The decoder uses a rising edge detector circuit at the START input to start decoding the received data. If the high period of the START input is greater than the CLK period, the decoder will start decoding. To detect the next rising transition, the START input must be low for a least one CLK period.

This allows the decoder to be operated in synchronous or asynchronous operation. Synchronous operation requires 10 or 34 clock cycles per decoded bit for 64 or 256 states, respectively. Asynchronous operation requires 11 or 35 clock cycles per decoded bit for 64 or 256 states, respectively.

When AUTO is high, only one START signal is required to start decoding. Input data must be input synchronously. The input data must be ready after RR goes high. The address bus RA can be used to read the input data from memory.

When AUTO is low, a START signal must be applied for each received symbol. This allows the decoder to operated asynchronously.

Figure 6 shows the relationship between the START input and R0I, R1I, R2I, R3I. In synchronous operation, these inputs must be valid from $2T_{cp}–T_{dsu}$ to $35T_{cp}+T_{dhd}$ ($11T_{cp}+T_{dhd}$) after the rising edge of START (T_{cp}, T_{dsu}, and T_{dhd} are the decoder clock period, setup time, and hold time, respectively). This implies that data should change one clock cycle after the rising edge of START.

In asynchronous operation these signals must be valid from $T_{cp}–T_{dsu}$ to $35T_{cp}+T_{dhd}$ ($11T_{cp}+T_{dhd}$) after the rising edge of START. Data must therefore change immediately after the rising edge of START.

Figure 7 shows the Viterbi decoder input timing with AUTO = 1. If AUTO = 0, the START signal must go high every time new data is to be input. That is, the RR signal is not automatically generated every 10 or 34 clock cycles. RR will go high every time START goes high with AUTO = 0. DEC_END goes low after the first START signal. The traceback starts automatically once all the data is received.

Figure 8 shows the Viterbi decoder output timing. The decoded output XD, the re–encoded outputs YD[3:0] and estimated channel BER outputs RE[3:0] are output in reverse time order. XDR goes high when the data is output with address XDA[5:0]. DEC_END goes high at the end of decoding.

RE[3:0] are obtained by exclusive ORing the appropriately delayed sign bit of the inputs with YD[3:0]. At low BER, these outputs can be used to give a good estimate of the channel BER.

Data Format

The decoder uses four bit signed magnitude quantisation for R0I to R3I. Table 3 shows the four bit quantisation ranges. Note that 0 and 8 indicate the central dead zone and have the same range. Note that most analog to digital to digital (A/D) circuits do have a central dead zone. For maximum performance, we recommend that five bit A/Ds are used with the output converted to four bits so that the appropriate ranges are obtained.

Table 3: Quantisation for R0I to R3I.

Decimal	Binary	Range
7	0111	6.5↔∞
6	0110	5.5↔6.5
⋮	⋮	⋮
2	0010	1.5↔2.5
1	0001	0.5↔1.5
0	0000	–0.5↔0.5
8	1000	–0.5↔0.5
9	1001	–1.5↔–0.5
10	1010	–2.5↔–1.5
⋮	⋮	⋮
14	1110	–6.5↔–5.5
15	1111	–∞↔–6.5

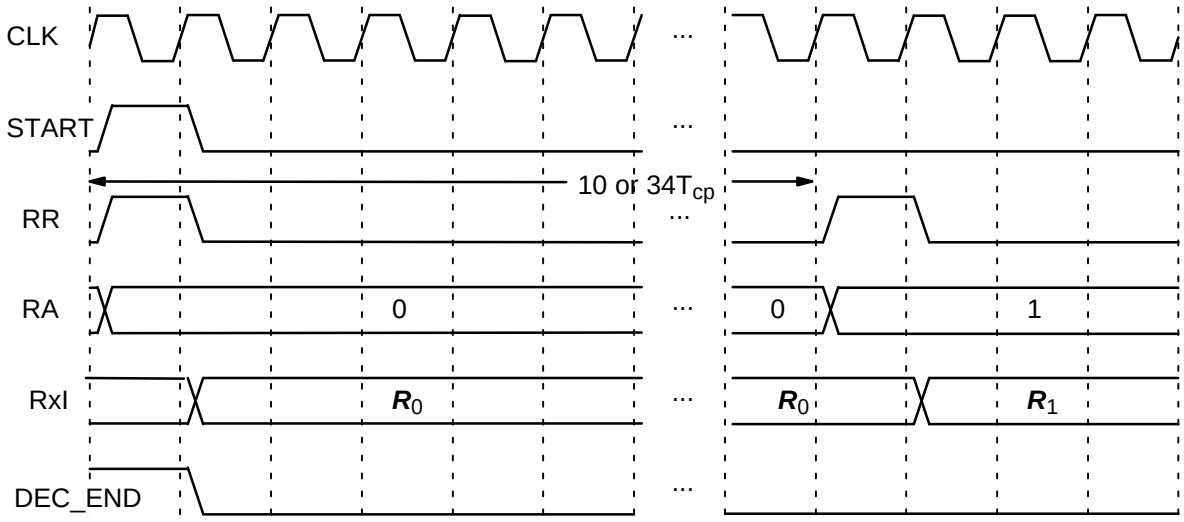


Figure 7: Viterbi Decoder Input Timing (AUTO = 1).

For input data quantised to less than 4–bits, the data should be mapped into the most significant bit positions of the input, the next bit equal to 1 and the remaining least significant bits tied low. For example, for 3–bit received data R0T[2:0], where R0T[2] is the sign bit, we have R0I[3:1] = R0T[2:0] and R0I[2:0] = 1. For punctured input data, all bits must be zero, e.g., R1I[3:0] = 0.

Punctured Code Operation

Manual puncturing can be performed by forcing R0I[2:0] to R3I[2:0] low. For example, rate 2/3 can be obtained by puncturing a rate 1/2 code with puncturing patterns of 11 for R0I and 10 for R1I. That is, R0I is not punctured, while R1I is forced low every other decoded bit.

Mode Selection

To minimise the decoder complexity, the MODE input can be used with the schematic symbols to select only those rates that are expected to be used. When MODE is low, the decoder can decode rate 1/2 and 1/3 codes. When MODE is high, the decoder can decode rate 1/2, 1/3, and 1/4 codes. MODE should only be connected to GND or VCC.

Other inputs

The RST input when high synchronously forces all flip–flops low. This is useful for VHDL simulations where flip–flops are initially in an unknown state.

State Metric Outputs

At the end of decoding three different state metrics are automatically output. SMZ[7:0], SML[7:0] and SMH[7:0] corresponds to the state metric (SM) for state zero, the minimum non–zero–state SM and the maximum non–zero–state SM. The term non–zero–state refers to all states except the zero state. The metrics range from 0 to 255, with the smaller metrics corresponding to more likely values.

All the state metrics can be read at the end of decoding. When SME is high, the state metric at address SMA[7:0] is read out at SMQ[7:0] one

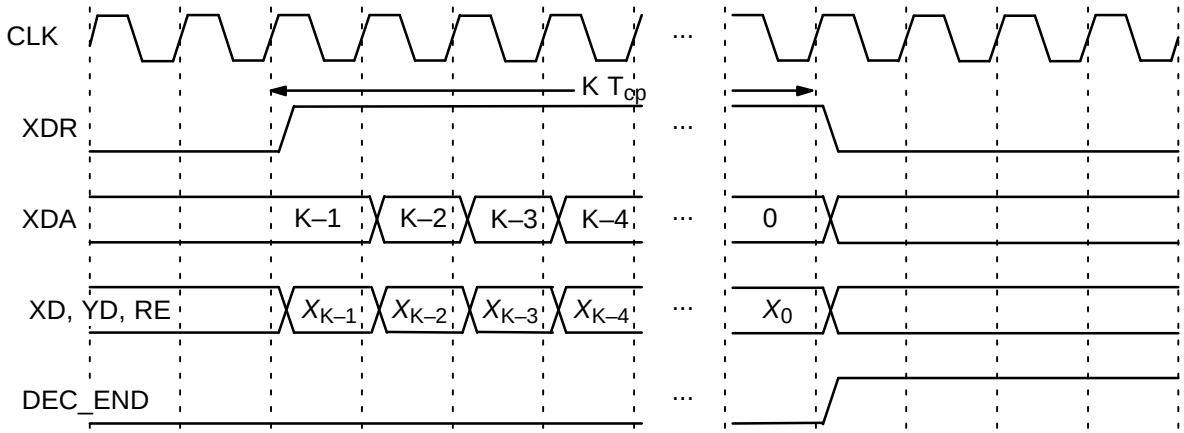


Figure 8: Viterbi Decoder Output Timing.

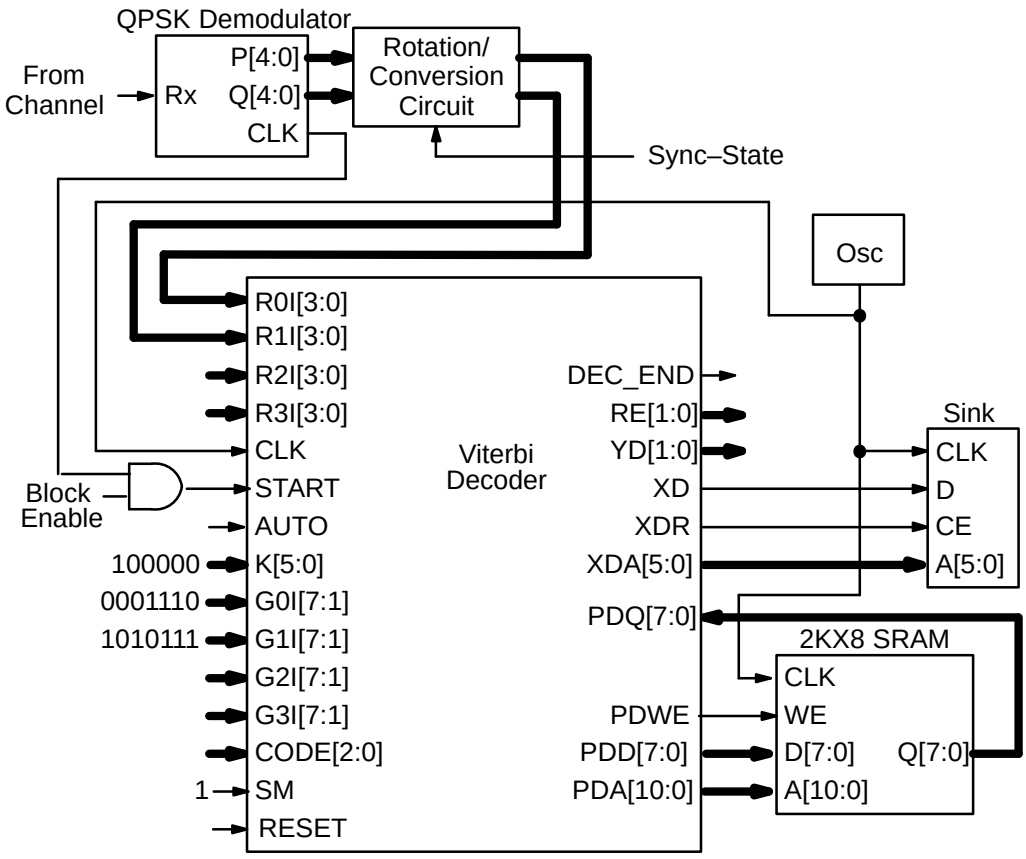


Figure 9: Block diagram of rate 1/2 QPSK codec.

clock cycle latter. SME can go high only after XDR goes high. SMA[7:6] needs to be set to zero for 64 state operation.

Decoder Speed

For synchronous operation, the decoding time for data length K is 10(K+6)+K+2 = 11K+62 and 34(K+8)+K+2 = 35K+274 clock cycles for 64 and 256 states respectively.

For asynchronous operation, the minimum decoding time is 11(K+6)+K+2 = 12K+68 and 35(K+8)+K+2 = 36K+282 clock cycles for 64 and 256 states, respectively.

Example

In this section we give an example of how the VA08S can be used as a block rate 1/2 QPSK decoder. Note that the VA08S does not perform any synchronisation. This needs to be performed external to the chip. The decoder is operated asynchronously.

Figure 9 shows how the VA08S with MODE = VCC and RST = GND can be configured for rate 1/2 QPSK operation. RESET is an asynchronous common reset available in Xilinx devices. The code used is $g_0 = 561_8 = 101110001_2 \rightarrow$ G0I[7:1] = 0001110_2, $g_1 = 753_8 = 111101011_2 \rightarrow$ G1I[7:1] = 1010111_2 (which is 180° rotationally invariant).

Note that unconnected inputs are pulled down to ground. Since the code is invariant to 180° phase rotations, differential encoding and decoding can be used if desired.

The state metric outputs are not used (SME and SMA[7:0] are low). The signals SME, SMA[7:0], SMQ[7:0], SMZ[7:0], SML[7:0], SMH[7:0], RR and RA[5:0] are not shown. The signal Block Enable is high while valid data is being received for the block. A block length of 32 is being used.

The demodulator output is assumed to be in 5–bit two's complement form. A small circuit is used to convert this to the 4–bit central dead zone format.

Ordering Information

SW–VA08S–SOS (SignOnce Site License)
SW–VA08S–SOP (SignOnce Project License)
SW–VA08S–VHD (VHDL ASIC License)
SW–VA08S–UNI–n (University License)

All licenses include EDIF and VHDL cores. The VHDL cores can only be used for simulation in the SignOnce and University licenses. The Uni-

versity license is only available to tertiary educational institutions such as universities and colleges and is limited to n instantiations of the core. The SignOnce and ASIC licenses allows unlimited instantiations.

Note that *Small World Communications* only provides software and does not provide the actual devices themselves. Please contact *Small World Communications* for a quote.

References

[1] Third Generation Partnership Project (3GPP), "Universal mobile telecommunications system (UMTS); Multiplexing and channel coding (FDD)," 3GPP TS 25.212 version 5.2.0 Release 5, Sep. 2002.

[2] Third Generation Partnership Project 2 (3GPP2), "Physical layer standard for cdma2000 spread spectrum systems, Revision D," 3GPP2 C.S0002–D Version 1.0, 13 Feb. 2004.

[3] P. J. Lee, "New short constraint length, rate 1/*N* convolutional codes which minimize the required SNR for given desired bit error rates," *IEEE Trans. Commun.*, vol. COM–33, pp. 171–177, Feb. 1985.

[4] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inform. Theory*, vol. IT–13, pp. 260–269, Apr. 1967.

Small World Communications, 6 First Avenue,
Payneham South SA 5070, Australia.
info@sworld.com.au ph. +61 8 8332 0319
http://www.sworld.com.au fax +61 8 7117 1416