Cascadable Block RAM

In the Virtex-5 block RAM architecture, two 32K x 1 RAMs can be combined to form one 64K x 1 RAM without using local interconnect or additional CLB logic resources. Any two adjacent block RAMs can be cascaded to generate a 64K x 1 block RAM. Increasing the depth of the block RAM by cascading two block RAMs is available only in the 64K x 1 mode. Further information on cascadable block RAM is described in the Additional RAMB18 and RAMB36 Primitive Design Considerations section. For other wider and/or deeper sizes, consult the Creating Larger RAM Structures section. Figure 4-7 shows the block RAM with the appropriate ports connected in the Cascadable mode.

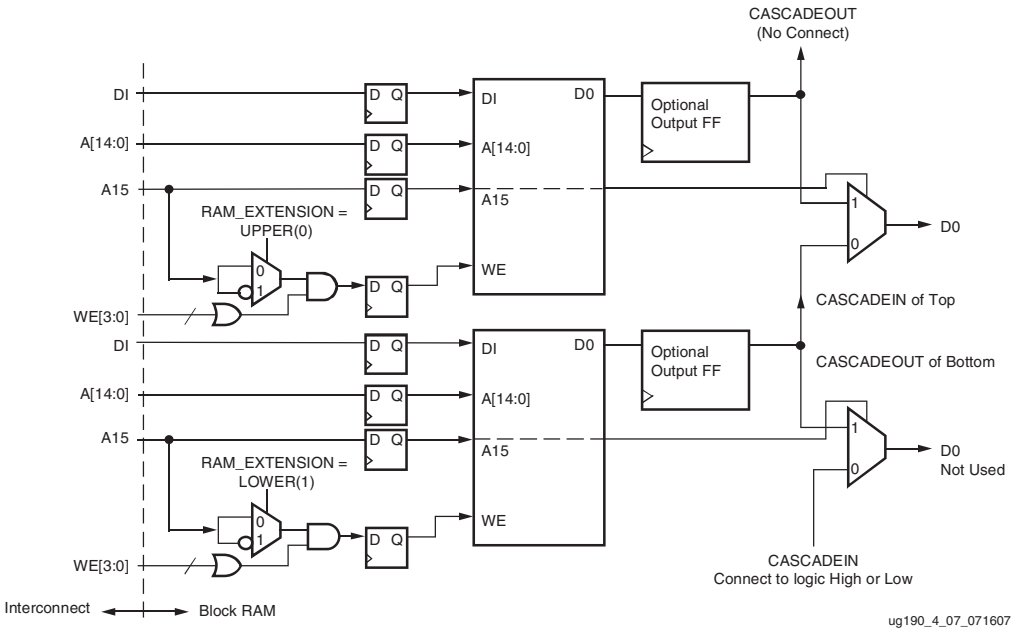

Figure 4-7: **Cascadable Block RAM**

Byte-wide Write Enable

The byte-wide write enable feature of the block RAM gives the capability to write eight bit (one byte) portions of incoming data. There are four independent byte-wide write enable inputs to the RAMB36 true dual-port RAM. There are eight independent byte-wide write enable inputs to block RAM in simple dual-port mode (RAMB36SDP). Table 4-4 summarizes the byte-wide write enables for the 36K and 18K block RAM. Each byte-wide write enable is associated with one byte of input data and one parity bit. All byte-wide write enable inputs must be driven in all data width configurations. This feature is useful when using block RAM to interface with a microprocessor. Byte-wide write enable is not available in the multirate FIFO or ECC mode. Byte-wide write enable is further described in the Additional RAMB18 and RAMB36 Primitive Design Considerations section. Figure 4-8 shows the byte-wide write-enable timing diagram for the RAMB36.

Table 4-4: **Available Byte-wide Write Enables**

Primitive	Maximum Bit Width	Number of Byte-wide Write Enables
RAMB36	36	4
RAMB36SDP	72	8

Table 4-4: **Available Byte-wide Write Enables *(Continued)***

Primitive	Maximum Bit Width	Number of Byte-wide Write Enables
RAMB18	18	2
RAMB18SDP	36	4

When the RAMB36 is configured for a 36-bit or 18-bit wide data path, any port can restrict writing to specified byte locations within the data word. If configured in READ_FIRST mode, the DO bus shows the previous content of the whole addressed word. In WRITE_FIRST mode, DO shows a combination of the newly written enabled byte(s), and the initial memory contents of the unwritten bytes.

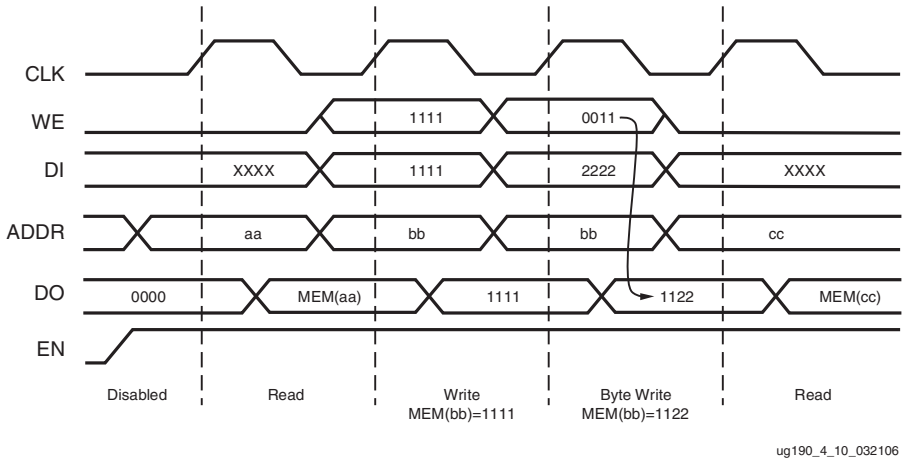

Figure 4-8: **Byte-wide Write Operation Waveforms (x36 WRITE_FIRST)**

Block RAM Error Correction Code

Both block RAM and FIFO implementations of the 36 Kb block RAM support a 64-bit Error Correction Code (ECC) implementation. The code is used to detect single and double-bit errors in block RAM data read out. Single-bit errors are then corrected in the output data.

Block RAM Library Primitives

The Virtex-5 FPGA block RAM library primitives, RAMB18 and RAMB36, are the basic building blocks for all block RAM configurations. Other block RAM primitives and macros are based on these primitives. Some block RAM attributes can only be configured using one of these primitives (e.g., pipeline register, cascade, etc.). See the Block RAM Attributes section.

The input and output data buses are represented by two buses for 9-bit width (8 + 1), 18-bit width (16 + 2), and 36-bit width (32 + 4) configurations. The ninth bit associated with each byte can store parity/error correction bits or serve as additional data bits. No specific function is performed on the ninth bit. The separate bus for parity bits facilitates some designs. However, other designs safely use a 9-bit, 18-bit, or 36-bit bus by merging the regular data bus with the parity bus. Read/write and storage operations are identical for all bits, including the parity bits.