

Take Advantage of Leftover Multipliers and Block RAMs

Here are some ways to make your designs more efficient.

by Peter Alfke
Director, Applications Engineering, Xilinx
peter.alfke@xilinx.com

The Virtex™-II Platform FPGA offers many multipliers and block RAMs: four each in the smallest XC2V40, 40 in the mid-range XC2V1000, and even more in the high-end devices. Many, if not most, designs will not require exactly all these functional blocks. It is thus of interest to explore alternate uses for leftover blocks.

Use the Multiplier as a Shifter

A multiplier can be used as a logic or arithmetic shifter. One operand is routed to the output, shifted by n positions, if the other operand is a power of two (2^n).

Because the sign-bit, or MSB (most significant bit), cannot be used to control the shift, the 18x18 2s-complement multiplier can shift by 0 to 16 positions.

Of the 36 multiplexer output lines, those less significant than the shifted data lines, are automatically filled with zeros; those more significant than the shifted data are filled with zeros or ones, depending on the state of the MSB input. This is the natural result of the 2s-complement multiplication.

You can either perform a logic shift of 17 input bits by holding the MSB input “low”, or perform an arithmetic shift of an 18-bit 2s-complement number, effectively sign-extending the MSB. (A conventional CLB-based shifter would have to use an array of n multiplexers, each with n inputs, and require a large amount of routing resources.)

In any case, shifters larger than 18 bits, and barrel shifters of any length will require external OR gating of the outputs.

Block ROM State Machines

Because block RAMs can be configured with any set of initial values, they make excellent dual-port registered ROMs. As shown in Figure 1, one half of the block can be used as a fast FSM (Finite State Machine), and the other half can be used for 36 additional parallel outputs.

The dual-port memory is divided into two completely independent, half-size, single-port memories by tying the MSB address bit of one port “high” (A) and the other one “low” (B).

To create a 256-state FSM, Port A is configured 2K x 9 and is used as a 1K x 9 single-port ROM. Eight outputs are fed back as address inputs, stepping through the 256 states. The remaining two address inputs determine the four-way branch. Any of the 256 states can conditionally branch to any set of four new states, under the control of the two address inputs.

Meanwhile, Port B is configured 512 x 36 and is used as a 256 x 36 single-port ROM. Port B receives the same 8-bit state-defining address as port A, and it drives 36 outputs that can be arbitrarily defined for each state.

Without any loss of speed (200 MHz max), you can easily modify the design to a 128-state FSM with an eight-way branch, or a 64-state FSM with a 16-way branch. If you need additional branch control inputs, they can be combined in an input multiplexer.

The advantages of this design are:

- Low cost (zero if the block RAM is otherwise not needed)
- High speed
- No layout or routing issues
- Complete design freedom.

More Specialized Uses of a Block RAM

A little creativity can go a long way. Here are some more design ideas. These solutions are compact and fast, and compete well against more conventional CLB-based implementations:

- 20-bit binary counter, or 18-bit binary up-down counter, in one block ROM, configured 1K x 18, running up to 200 MHz.
- Six-digit BCD (Binary Coded Decimal) counter in one block ROM, configured 512 x 36, plus one CLB, running up to 300 MHz. (These counters use one port for the less significant half of the counter, and the other port for the more significant half. This is possible because the count algorithm, stored in the ROM, is common to both halves.)

- Two independent 11-bit binary to 4-digit BCD converters, with the block ROM configured 1K x 18 and the LSBs (Least Significant Bits) not passing through the converters.
- Two independent 3-digit BCD to 10-bit binary converters, with the block ROM configured 2K x 9 and the LSBs not passing through the converters.
- Sine-cosine look-up tables using one port for sine, the other one for cosine, with 90 degree-shifted addresses, 18 bit amplitude, 10-bit angular resolution.
- μ -law to/from A-law telephony code converter, or μ /A-law to linear converter.

Conclusion

We encourage you to analyze any Virtex-II design for leftover multipliers and block RAMs. You can use them to unburden the logic fabric where possible. Furthermore, using multipliers as shifters, and block RAM as state machines, also simplifies the design effort, significantly reduces routing overhead and power consumption, and achieves higher performance.

It is hard to beat this combination – especially when it comes for free.

Figure 1 - Block RAM configured as 256-state FSM with 36 additional parallel outputs

