Parity Generators in FLEX 8000 Devices

May 1994, ver. 1 Application Brief 130

Summary

Files using the techniques described in this application brief are available from the Altera BBS at (408) 954-0104 in the following self-extracting file:

ab_130.exe

Parity generators are used to detect errors in data transmission. The data sender generates a parity signal by counting the number of ones in a data word and reporting whether that number is even or odd. The data receiver checks the parity signal to verify that the data was received without errors. In FLEX 8000 devices, you can implement high-speed parity generators using the carry chain feature, a fast (less than 1 ns) carry-forward function path between contiguous logic elements (LEs) within a Logic Array Block (LAB) and between adjacent LABs. You can also implement area-optimized parity generators that do not use the dedicated carry chain. Design techniques described in this application brief can be used to create design files optimized for the following characteristics (numbers in parentheses are for the parity generator without carry chains):

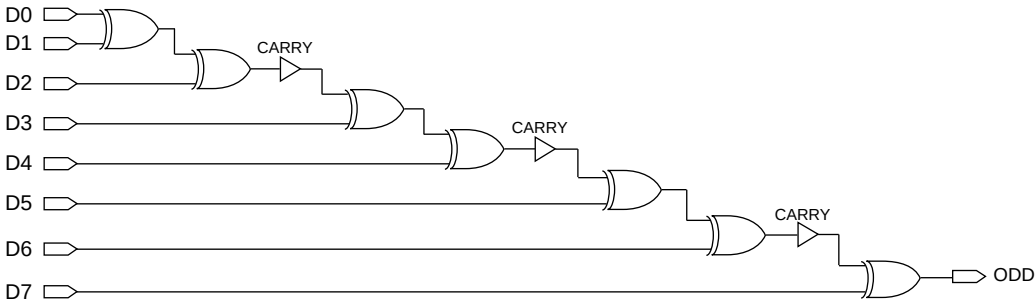
Design Goals:		*Design Results:*		
Architecture	**Optimization**	**Width**	**Logic Cells**	**Speed (MHz)**
✓ Look-Up Table	Routability	8 Bits	4	90.0 (66.7)
Product Term	✓ Speed	16 Bits	8	71.9 (66.7)
	✓ Area	32 Bits	17	51.3 (46.3)

Parity Generators with Carry Chains

Application Note 40 has been incorporated into the FLEX 8000 Programmable Logic Device Family Data Sheet.

Figure 1 shows a very-high-speed parity generator that uses carry chain logic. This parity generator accumulates the values of each bit in an 8-bit data word using a carry chain, and reports that the result is an odd number. This design can be extended to create parity generators for wider data words. Carry chain logic provides fast performance. However, since carry chains must be placed in contiguous LEs and LABs, long carry chains may reduce the routing resources available for implementing other logic. For more information about carry chains, refer to *Application Note 36 (Designing with FLEX 8000 Devices)* and *Application Note 40 (FLEX 8000 Architecture)* in this handbook.

Figure 1. High-Speed Parity Generator Implemented with Carry Chain Logic



Parity Generators without Carry Chains

Figure 2 shows an 8-bit parity generator that does not use carry chain logic. This parity generator adds the values of each bit in an 8-bit data word using XOR gates. This design runs more slowly than the design shown in Figure 1, but also uses less area and leaves more routing resources available for implementing other logic. If you wish to implement even parity generation, you can invert the ODD output signal.

Figure 2. Parity Generator without Carry Chain Logic