XAPP383 (v1.1) August 1, 2003

Single Error Correction and Double Error Detection (SECDED) with CoolRunner-II™ CPLDs

Summary

This application note describes the implementation of a single error correction, double error detection (SECDED) design with a CoolRunner-II™ CPLD. CoolRunner-II devices are the latest CPLD from Xilinx that offer both low power and high-speed performance. A complete VHDL design is available with this application note, see **VHDL Code**, page 4.

Introduction

To improve system reliability, a designer may wish to provide an automatic error detection and correction circuit. One such example is the data communicated from the microprocessor to peripheral memory devices. This document describes a flow-through method for doing data SECDED with a CPLD. In this design, multiple parity bits are added to the data word upon a write to memory. With multiple parity bits, both single and double data errors can be detected upon reading the word from memory and correct single data errors. The CPLD provides a 2-bit error output flag for the microprocessor to handle detected double errors.

The SECDED design described here is the combinational logic for data communication between the microprocessor and memory. The data bus from the processor is 16-bit wide data, while the data written to memory is a 22-bit data word. When data is read back from the memory device, the stored parity bits are compared with a newly created set of parity bits from the read data. The result of this comparison, called the syndrome, will indicate the incorrect bit position in a single data error.

This design is a model of the Hamming code developed by R. Hamming (see **References**, page 4 for more information). SECDED for N bits of data requires K parity bits to be stored with the data where:

$$N <= 2^{K-1} - K$$

If the bits are numbered in sequence, those bit positions that represent powers of two are dedicated to parity bits. Table 1 illustrates how the 16-bit data word and parity bits are stored in memory.

Table 1: **Hamming Code Data and Parity Bits**

Bit Position	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Bit Number	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Data/Parity Bit	P5	D 15	D 14	D 13	D 12	D 11	P4	D 10	D9	D8	D7	D6	D5	D4	P3	D3	D2	D1	P2	D0	P1	P0

NOTICE OF DISCLAIMER: Xilinx is providing this design, code, or information "as is." By providing the design, code, or information as one possible implementation of this feature, application, or standard, Xilinx makes no representation that this implementation is free from any claims of infringement. You are responsible for obtaining any rights you may require for your implementation. Xilinx expressly disclaims any warranty whatsoever with respect to the adequacy of the implementation, including but not limited to any warranties or representations that this implementation is free from claims of infringement and any implied warranties of merchantability or fitness for a particular purpose.

The parity bits P0-P4 are created for single error detection and correction and are created as follows:

$$P0 = D15 \oplus D13 \oplus D11 \oplus D10 \oplus D8 \oplus D6 \oplus D4 \oplus D3 \oplus D1 \oplus D0$$

$$P1 = D13 \oplus D12 \oplus D10 \oplus D9 \oplus D6 \oplus D5 \oplus D3 \oplus D2 \oplus D0$$

$$P2 = D15 \oplus D14 \oplus D10 \oplus D9 \oplus D8 \oplus D7 \oplus D3 \oplus D2 \oplus D1$$

$$P3 = D10 \oplus D9 \oplus D8 \oplus D7 \oplus D6 \oplus D5 \oplus D4$$

$$P4 = D15 \oplus D14 \oplus D13 \oplus D12 \oplus D11$$

One additional parity bit, P5, detects double errors that are not correctable. This extra parity bit is an overall parity bit and is comprised by XOR-ing all the data bits, D15-D0 and parity bits, P0-P4.

The syndrome is created upon a memory read and provides the ability to correct single bit errors. The syndrome is created by XOR-ing the parity bits read out of memory with the newly created set of parity bits from the data stored in memory. The value of the syndrome will indicate the bit position in error (if a single error has occurred). Table 2 illustrates the value of the syndrome and the overall parity bit in detecting both single and double errors.

Table 2: **Error Detection**

Syndrome	Overall Parity (P5)	Error Type	Notes
0	0	No Error	
/=0	1	Single Error	Correctable. Syndrome holds incorrect bit position.
/=0	0	Double Error	Not correctable.
0	1	Parity Error	Overall parity, P5 is in error and can be corrected.

SECDED Design

Figure 1 illustrates the block level design of the SECDED for the CoolRunner-II CPLD. The left side of the diagram illustrates the processor interface. This interface consists of the 16-bit processor data bus, u_data[15:0], the read/write control signal, rw_n, and the error flag signal, error_out[1:0]. The right hand side describes the memory component interface, consisting of the memory data bus, mem_data[21:0].

The rw_n control signal from the processor switches the CPLD between read and write cycles. The rw_n signal will be equal to "1" for a processor read cycle and equal to "0" for a processor write cycle.

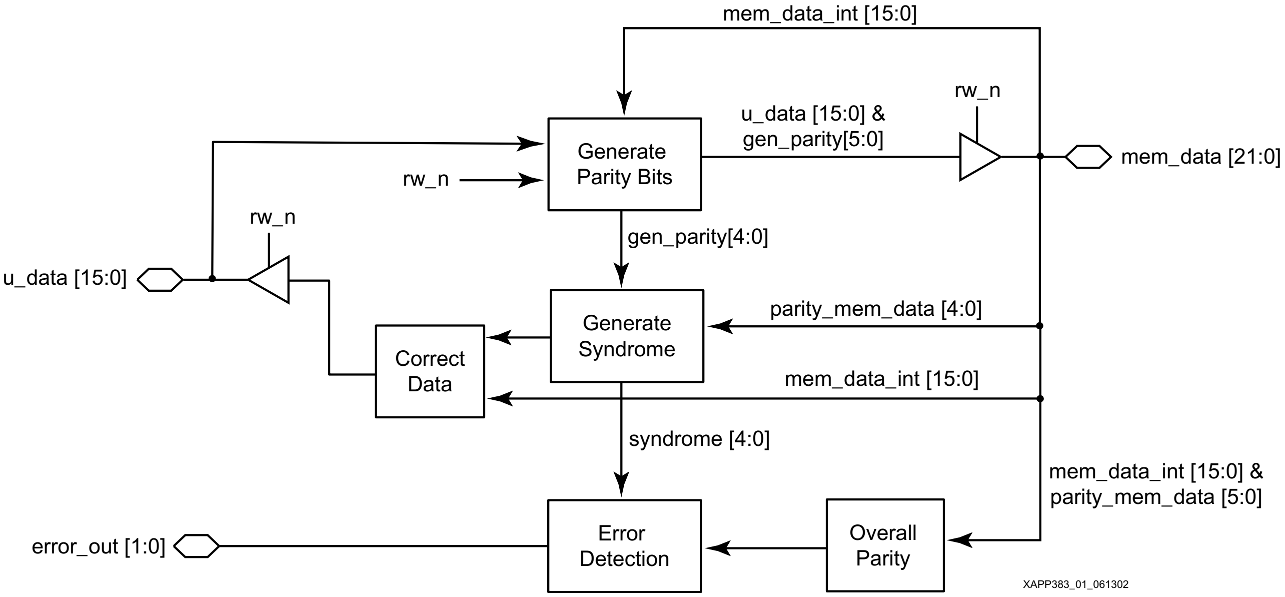

Figure 1: **SECDED Block Diagram**

The "Generate Parity Bits" block creates the parity bits to store with the processor data (u_data[15:0]) during a write cycle. In a read cycle, this block is also responsible for creating one of the inputs in generating the syndrome; this block creates the parity bits with the data word stored in memory.

The "Error Detection" block generates the error_out[1:0] flag based on the syndrome and the overall parity created from the data in memory. The error_out flag decodes to the states shown in Table 3.

Table 3: **Error Detection Flags**

error_out[1:0]	Description
00	No error has occurred.
01	Single error has been detected. Syndrome holds value of erroneous bit.
10	Double error has been detected. Not correctable
11	Parity error has occurred. Correctable.

The design illustrated here only describes the data path logic between the processor and memory device. This design can be expanded to include the control logic for interfacing with both the processor and memory device(s).

Design Implementation

The SECDED design described in this application note is targeted to a XC2C128-4VQ100 CoolRunner-II device. The device utilization data is shown in Table 4.

Table 4: **SECDED XC2C128 Device Utilization**

Device Resource	Available	Used	% Utilization
Macrocells	128	66	51%
I/O Pins	80	41	51%
Product Terms	448	212	47%
Registers	128	0	0%
Function Block Inputs	320	129	40%

Timing analysis for the XC2C128-4VQ100 device yields a T_{PD} = 13.8 ns. This calculation is based on logic that requires four macrocell levels.

VHDL Code

THIRD PARTIES MAY HAVE PATENTS ON THE CODE PROVIDED. BY PROVIDING THIS CODE AS ONE POSSIBLE IMPLEMENTATION OF THIS DESIGN, XILINX IS MAKING NO REPRESENTATION THAT THE PROVIDED IMPLEMENTATION OF THIS DESIGN IS FREE FROM ANY CLAIMS OF INFRINGEMENT BY ANY THIRD PARTY. XILINX EXPRESSLY DISCLAIMS ANY WARRANTY OR CONDITIONS, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, AND XILINX SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR A PARTICULAR PURPOSE, THE ADEQUACY OF THE IMPLEMENTATION, INCLUDING BUT NOT LIMITED TO ANY WARRANTY OR REPRESENTATION THAT THE IMPLEMENTATION IS FREE FROM CLAIMS OF ANY THIRD PARTY. FURTHERMORE, XILINX IS PROVIDING THIS REFERENCE DESIGN "AS IS" AS A COURTESY TO YOU.

XAPP383 - **http://www.xilinx.com/products/xaw/coolvhdlq.htm**

Conclusion

CoolRunner-II CPLDs are the ideal solution in providing flow-through SECDED with a processor and memory device(s). CoolRunner-II CPLDs are targeted for applications that require both low power consumption and high performance.

References

1. Stallings, William. Computer Organization and Architecture: Designing for Performance. 4th Edition. Prentice Hall. 1996.
2. Smith, Douglas. VHDL & Verilog HDL Chip Design. Doone Publications. 1996.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
09/26/02	1.0	Initial Xilinx release.
08/01/03	1.1	Minor revisions