Optimized 64-Bit Multiplier and 128-Bit / 64-Bit Divider

Project Specifications

ECE 645 4/1/02

Team Members: Michael Pocock James Wright

Hardware Design

Introduction

This document is to serve as the Project Specification and Description for the Optimized 64-Bit Multiplier and 128-Bit / 64-Bit Divider. This project will fulfill the requirements set forth by Professor Kris Gaj for the ECE 645 Final Project. A general description of the project can be found on the ECE 645 website:

http://mason.gmu.edu/~kgaj/ECE699/project2.html

The project to be completed is Project 3. It calls for the hardware design of a microprocessor capable of performing the multiplication of two 64-Bit signed integers, and the signed division of a 128-Bit number by a 64-Bit number. No software design will be done. It will be replaced with a third hardware optimization criteria.

The project will be completed by a team composed of Michael Pocock and James Wright.

Functional Requirements

- 1. It shall perform either multiplication or division based on control input.
- 2. It shall multiply two signed 64-Bit wide integers.
- 3. It shall provide a multiplication result as a 128-Bit signed integer.
- 4. It shall divide a 128-Bit signed integer by a 64-Bit signed integer.
- 5. It shall provide a division result as a 64-Bit signed integer.

Application

A real-world application of the multiplier/divider will be in a 64-bit digital signal processor (DSP) arithmetic logic unit (ALU). The ALU is a building block of any microprocessor or DSP that performs many arithmetic functions based on the control-input selection. DSP functions can be performed in both floating-point and fixed-point representations, but since floating-point operations cannot be synthesized, the ALU in this design will focus on fixed-point representations.

Optimization Criteria

The design will be optimized for three different criteria:

1. Minimum Latency

Minimum latency describes the total time it takes to compute either the multiplication or division function, or the time from when the inputs are valid to when the output is valid. This is a function of combinatorial delays, register setup times, and the operating frequency. Since the operating frequency is variable, the design latency will be measured as the maximum computation time required at the maximum frequency the design can achieve given the static hardware delays.

2. Maximum Throughput

Maximum throughput is the rate at which the design can accept new inputs and still provide valid outputs at the same rate. This is best described as the operating frequency. This design however will not be the same as the one optimized for latency. Maximum throughput will be achieved with the maximum pipelining in the design. This will mean adding "extra" registers in between combinatorial circuits to minimize combinatorial delays between registers. Thus the overall latency of the design will increase when designing for maximum throughput.

3. Minimum Area

The minimum area of the design will be achieved by reusing as many of the device resources as possible. This implies a more serial approach to the computations, where speed designs imply a parallel approach. All designs will assume a set of input registers to register the input operands as parallel words, and an output register for the result in a parallel format. However, after registering the inputs, the operand bits may be shifted serially.

Design Tools

The design will be done in VHDL code.

Simulation and Verification:

ModelSim XE/Starter 5.5b Revision: 2001.05 Date: May 24 2001 Copyright © 1990 - 2001 Model Technology Incorporated Supported through Xilinx (800) 255-7778

Implementation:

Xilinx WebPACK 4.2 Version 4.2WP0.0 Design Entry - Project Navigator Synthesis - Xilinx Synthesis Technology

Assumptions

The target architecture for the design will be a Xilinx Virtex II device. The Virtex II device has many special application embedded hardware functions, but the primary logic area of the chips are CLBs, or Configurable Logic Blocks. Each CLB has 2 LUTs

and 2 Memory components (Register/Latch). These will be the basic structures used for determining the cost and latency of the design for each of the above optimization criteria.

There are other components of the CLB the synthesizer can use to further optimize the design such as mutiplexers and carry-lookahead logic. However these components will not be used in the design of the Multiplier/Divider. The synthesizer option to use these optimizing components to the design will be disabled, forcing the logic to be implemented purely in LUTs and registers. Also, the synthesizer settings for adding IOBs (Input / Output Blocks) will be removed, since the design is not concerned with the actual I/O and routing delays, only delays through the logic.

The 4 input Look-up table is capable of implementing any Boolean algebra function with 4 inputs. Thus a 2 input AND gate has the same device resource usage and propagation delay as a 4 input AND gate. Likewise, a 3 input XOR gate has the same characteristics as a 3 input AND gate.

Since there are an equal amount of LUTs and registers in the target architecture, their cost relative to each other is equal, and they will be treated as such when measuring the area of a design.

Test Plan

Model Technology's ModelSim XE/Starter 5.5b simulation tool will be utilized to functionally verify the operation of the multiply/divide functionality. Test vectors will be generated to verify, at minimum, the worst case delay path, overflow, carry, and zero flags.

Circuit speed and area will be determined by the synthesis results of the design utilizing Xilinx WebPACK 4.2. The functional verification above will be repeated including timing information from the synthesis results.

References

Xilinx Virtex II FPGA Data Sheet