Wide Unsigned Integer Multiplier ACcumulator (WUIMAC)

Gil Glass
Allen Michalski
Damien Fedoryka

Real-Life Applications

Digital filters are used in a wide variety of applications such as audio signal processing equipment, IF processing equipment, and delta-signal modulator ($\Delta\Sigma$M) A/D converters. All of these digital filtering mechanisms require a multiplier accumulator whose purpose is to multiply individual signal samples by filter coefficients [6].

Software implementations of digital filters can be found in many of today's *digital signal processors* (DSPs). These are generally highly configurable for a wide variety of applications [1].

Hardware implementations tend to be used more in specialized applications where the characteristics of the filter are to be predefined [1].

Functional Requirements

The purpose of the WUIMAC project is to create hardware and software implementations of a multiplier-accumulator (MAC) for use in a digital filters.

Multiplier-accumulators can be found in various sizes in various applications. These may range from 1×11 bits in some $\Lambda\Sigma$Ms [2] to 16×16 and 32×32 bit varieties [3].

The WUIMAC will implement accumulation of up to 16 32×32 partial products, resulting in an 80-bit result.*

As it stands now, the only control signals needed in a hardware design are a clock, an asynchronous (or synchronous, depending on design) clear signal, parallel (or serial, again, depending on design) multiplicand input, multiplier input, and parallel data output.

Software arguments will consist of 32-bit multiplier and multiplicand arguments with the function returning up to an 80-bit number.* Note that, in software, the number will have to returned from the function in phases as very few, if any processors will handle 80-bit numbers.

* Multiplying a 32-bit number by a 32-bit number can yield up to a 64-bit number. Each time a 64-bit number is added to another, a carry out may be generated. Therefore, adding 16 of these can result in a result which is 80 bits wide.

Optimization Criteria

As is specified in the project assignment, the design will be optimized to reduce latency. This makes sense since most digital filter applications are intended to operate in real time on potentially high-frequency signals.

CAD Tools

We will use CAD tools available both in the ECE lab and at our places of work.

Hardware descriptions will be written in VHDL-93 and simulated using vsim - Version 5.4d. This operation will be performed on CPE02, running SunOS version 5.8, remotely.

Software will be developed using Microsoft Visual C++ version 6.0 from our places of work.

Assumptions

Hardware

When designing the hardware, it will be assumed that all standard logic gates (up to 4-input) are available via the VHDL library. It will further be assumed that elementary components such as D-flip flops will also be available.

Delays are in terms of gates. One gate = one delay regardless of its size. D-flip flops will be presumed to require 2 gate delays to propagate an input to an output following a clock signal or a reset.

Component area will be in terms of gates. One two-input gate = one area unit. Each additional input adds ½ unit of area.

The exact interface of the circuit will be determined during the design phase.

Software

C will be used to implement the WUIMAC in software. Library functions have not yet been determined. Since this software will be intended to run on a microcontroller or DSP, library functions (e.g. DLLs) will probably not be available.

Exact function arguments will be determined during the design phase.

Test Plan

Hardware

The VHDL design will be tested using a series of test vectors to be cycled through the MAC. These numbers will be chosen to, as fully as possible, test the functionality of the circuit. Since these numbers will be inherently very large, they have not been listed here.

Circuit speed will be determined from ModelSim simulator reports for worst case test vectors (TBD).

Circuit area will be calculated by counting the number of gates.

Software

Software will be verified using the same test vectors as those used for the hardware.

Function execution time will be estimated based on assembly language output listings and instruction execution times.

Memory requirements will be estimated based on linker generated list files. Program (ROM) space will be available as well variable (RAM) space.

References

The following references are being consulted for information about digital filters and MACs:

1. *Application Note AN9603.2, "An Introduction To Digital Filters"* - Intersil™

2. *Application Note AN53, "The CS5322 Digital Filter"* – Crystal, A Cirrus Logic Company

3. *"Digital Filter Techniques Using the 80C196"* – Intel™ Corporation

4. *"Design and Low Speed Testing of a Four-Bit RSFQ Multiplier-Accumulator"* – Herr, Mancini, Gaj, et. al.[***]

5. *FPGA Digital Filter Application Note* - Atmel™

6. *"Digital Signal Processing"* – Richard A. Roberts and Cliffort T. Mullis, Copyright © 1987 by Addison-Wesley Publishing Co. Inc.

[***] The reliability of this particular source is questionable. ☺