ZERO OVER HEAD LOOP FIR PROCESSOR

Sourindu Chatterjee
Department of Electrical and Computer Engineering, Florida State University

SUBJECT AND PURPOSE

The purpose of this project is to implement the hardware specialized technique found in most DSP processors, ZERO OVER HEAD LOOPING towards the development of a FIR microprocessor. The filter coefficients would be real-time and can be read both as form of a signal and, or through the Machine Instruction File.

DEFINITION OF THE PROBLEM

Well the implementation of Zero Overhead Loop (ZOL) for various embedded DSP microprocessors is profound, where a Hardware abstraction is provided for the programmer to code in a segment of assembly instruction to be iterated independently for a specified number of times but little recourse is available for the implementation of such a design for a specific filtering microprocessor design [1]. This project is about implementation of Zero Overhead Loop control in Hardware which would be coded in Very Fast Hardware Description Language (VHDL) for the design of a FIR microprocessor, such that the execution of loop branching takes only a single cycle to execute the loop contents. The ZOL is a design unit which should be implemented as such that it comes as zero cost and should really make no difference whether loop unrolling is done or not [3]. There is another challenge associated with the development of the FIR microprocessor that is to provide a technique for coefficients to be programmable whether during runtime or during compilation.

IMMEDIATE BACKGROUND OF THE PROBLEM

In this problem a design of the Zero Over Head loop is going to be implemented. ZOL (Zero Over-Head Loop) can be explained as a buffer of instructions which is iterative and a usually a single loop control is used for designing such a iterative buffer. This is especially an architectural solution for eliminating the overheads associated with loop branching for execution of structured algorithms. In this case these algorithms or instructions would mostly be multiply and accumulate or the sum of products code [2]. The design for some DSP microprocessor like DSP16000 is possible by storing loop indexing, initial, step and final values in local register and a single process unit is employed for manipulating the control schedule for the whole loop [2]. Though various techniques can be adapted for implementing ZOL the basic principal of single cycle loop branching and zero cost loop structure remains the same.

Another part of the problem is the design of the Finite Impulse Response Processor itself. The FIR filter is usually termed as linear, all zero filter design [1] [5]. The microprocessor architecture can be designed based on two basic techniques: Direct Form & Transposed Form. Though both structures requires same hardware resources the transposed form structure is easy to implement and realize in hardware. The filter coefficients in this case can be constant or programmable, can be taken as input signal or can be read from the machine instruction file ".MIF" depending on the final design of system. Well the former would be a real-time filter design with runtime input where the later would be a filter design with compile time input.

NEED FOR SOLUTION TO THE PROBLEM

We know that the most fundamental equation in FIR filter design is the sum-of-product

(SOP) equation, $y[n] = \sum h[k] * x\ [n{-}k]$ [6] [5], which can be also termed as convolution in the time domain of input signal x and impulse response h. This is the reason ZOL is fundamental to the execution of this kernel equation, which requires multiply, accumulate and data fetches repeated over n number of times every time kernel equation is computed [6][1][3] inside a loop.

The issue relating to ZOL is very important and in some cases pivotal to the efficiency of the DSP microprocessor. In this project the task is to develop a FIR microprocessor with ZOL support, we know that most of the task that the microprocessor control unit would perform is looping through the multiplication and accumulation instructions, which delivers the sum of product style results for the loop branch. Thus to provide an efficient and reliable architecture to execute repetitive instruction within a single cycle execution of loop control is very essential and somewhat paramount to the purpose of DSP. This is utterly true because most of the tasks performed by DSP microprocessors are on real-time systems where the cycles per instruction is needed to be as less as possible, whether it is ALU instruction or branching or looping instruction the reduction is execution time is important and a critical factor in determining effectiveness of the system [4].

FEASIBILITY AND USEFULLNESS OF THE SOLUTION

We all know that Gene`s law drives the development of the DSP microprocessors, where the ratio of the power consumption to the instruction per second is sets as a decreasing linearity over time. Thus to maintain the effectiveness of the low power high performance DSP microprocessor unit the effectiveness of ZOL is preeminent. ZOL will not only reduce the time of execution of instruction but also will reduce the size of the code.

SCOPE STATEMENT

In this project it is planned to have the input data and the coefficients limit set to 32 bits but this is not fixed and is a subject to change. The filter coefficients or filter length is not fixed, it is easier to simulate with fewer coefficients and thus will be a smaller number.

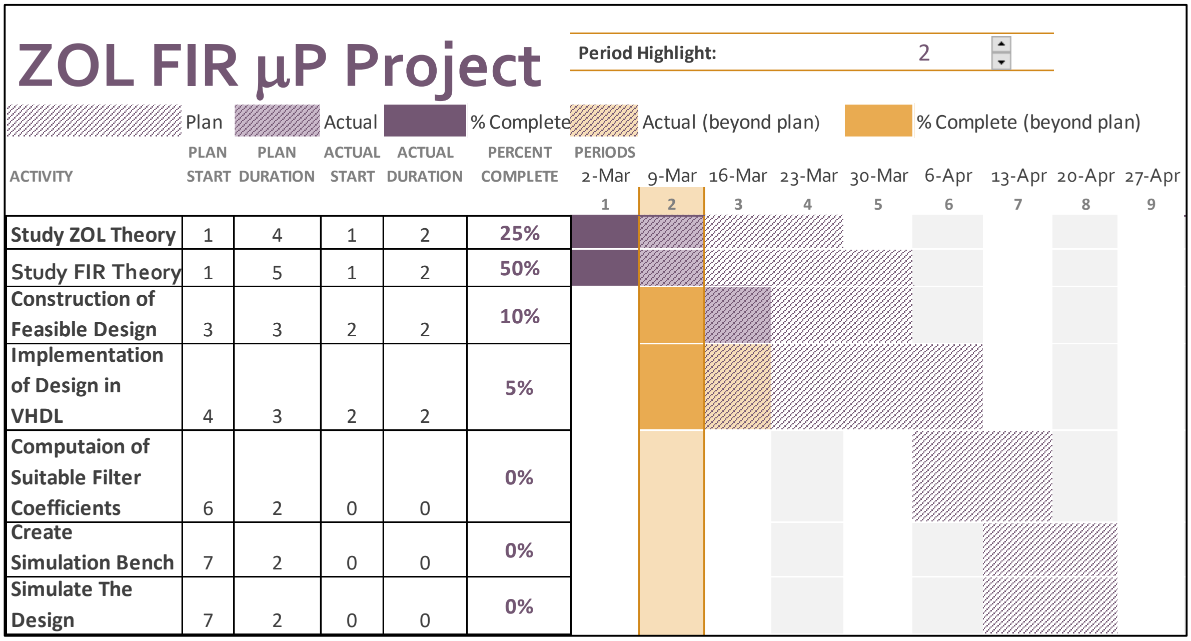
PROCEDURE

The project will be mapped by a VHDL code written and compiled with Quartus II software. The simulation would be done with ModelSim, it is assumed that two input signal would be used to get the simulated output signal. The use of Matlab would also be there to generate filter coefficients set.

FACILITIES TO BE USED

QuatrusII, ModelSim and Matlab software would be used for doing the project work.

TIME AND WORK SCHEDULE



ZOL FIR µP Project

Period Highlight: 2

▨ Plan ▨ Actual ■ % Complete ▨ Actual (beyond plan) ■ % Complete (beyond plan)

PERIODS: 2-Mar (1), 9-Mar (2), 16-Mar (3), 23-Mar (4), 30-Mar (5), 6-Apr (6), 13-Apr (7), 20-Apr (8), 27-Apr (9)

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Study ZOL Theory	1	4	1	2	25%
Study FIR Theory	1	5	1	2	50%
Construction of Feasible Design	3	3	2	2	10%
Implementation of Design in VHDL	4	3	2	2	5%
Computaion of Suitable Filter Coefficients	6	2	0	0	0%
Create Simulation Bench	7	2	0	0	0%
Simulate The Design	7	2	0	0	0%

CONCLUSION

In summary, this project proposal is set to give right idea about the implementation of ZOL in FIR microprocessor, which is a very arduous task. The design of the microprocessor is not yet finalized and can be changed several time depending on the feedback of the system during simulation. The basic idea is to deliver a workable system design for further development and modifications.

REFERENCES

[1] U. Meyer-Baese: Digital Signal Processing with Field Programmable Gate Arrays (Springer, 4th edition, 2014).

[2] Zero-overhear loop controller for implementing multimedia algorithms
Nikolaos Kavvadias and Spiridon Nikolaidis,Section of Electronics and Computers, Department of Physics Aristotle University of Thessaloniki.

[3] Effective Exploitation of a Zero Overhead Loop Buffer
YuhongWang, David Whalleyt, Department of Computer Science, Florida State University
Gang-RyungUh, SanjayJinturkar, Chris Bums, Vincent Cao, Lucent Technologies,Allentown

[4] Application of Zero Overhead Loop and Address Generator in General Purpose Computing
Jinyung Namkoong.

[5] EEL 5722 DSP with FPGA lecture Notes.

[6] http://microchip.wikidot.com/dsp0201:zero-overhead-loops