

# On the Performance of Timing-Error Tolerant Programmable FIR Filters

P N Whatmough<sup>†‡</sup> and I Darwazeh<sup>†</sup>

<sup>†</sup> University College London, <sup>‡</sup> ARM Ltd, Cambridge

**Abstract:** A modified transposed direct-form programmable FIR filter is proposed to combat potential transient timing-errors in low voltage operation. The architecture allows the isolation of errors in the filter taps in combination with simple error correction schemes. Two such schemes are investigated with ideal error detection. The filter stop-band rejection in the presence of timing-errors for both schemes is presented from simulation results of the proposed filter.

## 1. Introduction

Finite Impulse Response (FIR) filters are a class of important digital signal processing (DSP) functions. Efficient implementation can be challenging for high-throughput programmable filters since they require many parallel multiply-accumulate (MAC) operations which can consume considerable power and silicon area.

Timing errors can arise due to sub-critical voltage scaling, process variation and changes in temperature. Architectures that can tolerate a small but non-zero timing error rate while satisfying the required performance specification are highly desirable as they potentially allow reduced power consumption and/or improved fabrication yield. This paper considers the implementation of a fully-parallel programmable FIR filter. Such structures have only data-path logic, and no control logic, which makes for a tractable problem, since tolerating timing-errors in control logic is a considerable challenge.

In the following sections we will look briefly at CMOS energy and delay and consider sub-critical voltage scaling as a technique to lower power consumption and hence motivate the study of timing-error tolerant VLSI architectures. We will then propose a new architecture for a programmable FIR filter and present simulation results.

### 1.1. CMOS Energy and Delay

Static CMOS logic uses the finite drive current of the transistors in the design to charge/discharge the load capacitance,  $C_L$ . The load capacitance is the sum of the MOS gate capacitance of all fan-out loads, metal interconnect (net) loads and all associated parasitics. Charging the load capacitance requires energy to manipulate the charge ( $Q = C_L V_{dd}$ ) at potential  $V_{dd}$ , which is given by:

$$E_{dynamic} = \frac{Q^2}{C_L} = C_L V_{dd}^2 \quad (1)$$

Half of this energy is dissipated when charging  $C_L$  through the PMOS networks and the remaining half is dissipating when discharging through the NMOS networks. We can determine power dissipation as the energy expended per unit of time. Given the clock frequency of the system,  $f_{clk}$ , and an estimate of the average probability of transition between logic states during the clock cycle,  $P_{0 \rightarrow 1}$ , we have

$$P_{dynamic} = P_{0 \rightarrow 1} f_{clk} C_L V_{dd}^2 \quad (2)$$

Significantly, we can observe that the power consumption is proportional to the clock frequency. Presuming  $f_{clk}$  is fixed by the required system throughput, we have two degrees of freedom with which to reduce power consumption,  $C_L$  and  $V_{dd}$ , which can be traded-off, bearing in mind that they are inextricably linked by the propagation delay,  $t_d$ , which can be described for combinatorial logic, to first order, by

$$t_d = \frac{C_L V_{dd}}{I} \propto \frac{C_L V_{dd}}{\beta (V_{dd} - V_t)^\alpha} \quad (3)$$

where  $\alpha$  is the device velocity saturation index and  $\beta$  is the device transconductance. In turn, the propagation delay ultimately determines the minimum clock period,  $t_{clk}$ , such that  $t_{clk} \leq t_d$ . Hence, there exists the familiar dichotomy between power consumption and performance. It is observed that low-power and high performance

design are often interchangeable, in that we can optimise propagation delay for the same power dissipation (achieving higher performance) or optimise power dissipation for the same propagation delay (low-power design). Only dynamic (switching) energy dissipation is considered in this paper, but it should be borne in mind that for deep sub-micron processes, threshold voltages are no longer high enough to neglect sub-threshold currents and thus static energy dissipation becomes relevant, if not dominant, for energy-constrained designs.

## 1.2. Sub-Critical Voltage Scaling

The propagation delay of a digital combinatorial circuit is a function of the input operands, such that the average propagation delay over all possible input combinations,  $t_{d,average}$  is smaller than the worst-case delay,  $t_{d,max}$ . The traditional digital design approach accounts for the worst-case delay to guarantee correct operation for all input operands. For the lowest power consumption we aim to scale close to the critical voltage, where  $t_{d,max} = t_{clk}$ , ignoring the margin used to account for process, voltage and temperature variation, which will not be discussed here. However, if we abandon the premise of logically correct operation, we can scale beyond the critical voltage for even greater power reduction. For example, if we reduce  $V_{dd}$  to the point where  $t_{d,max} > t_{clk}$ , (but typically  $t_{d,average} < t_{clk}$ ), the longest paths in the circuit, when sensitised by suitable input data, will not evaluate within the clock period,  $t_{clk}$ , and bit errors will start to appear internally and eventually may reach the output.

Thus a sub-critically scaled system is subject to timing-errors, that are the occurrence of critical operands such that  $t_d > t_{clk}$  for a given  $V_{dd}$ . Timing-errors are somewhat transient; deterministic by nature, but depending on certain stimulus input conditions, which in practice may occur extremely rarely, if ever. These errors represent a new source of noise in our system which we need to either correct or tolerate within our performance envelope. Many systems, such as general purpose computing, would not be amenable to such transient logic errors, but many DSP functions are designed to achieve certain quantitative performance criteria (e.g. signal-to-noise ratio) whereby soft errors of a certain quantity may not necessarily prevent satisfactory performance.

In practical examples of sub-critically scaled systems, dynamic voltage scaling (DVS) is often employed so that the voltage scaling can vary to track changes in temperature and input data statistics in order to maintain a manageable (small but non-zero) logic error rate. For the sake of brevity, we will not consider DVS, but instead assume a static simulation environment without process, voltage and temperature variation. A signal processing system that can tolerate timing-errors is referred to as Soft-DSP by Hedge and Shanbhag who have pioneered a number of different approaches [1], which they refer to collectively as Algorithmic Noise Tolerance (ANT).

## 2. VLSI Design of FIR Filters

Figure 1 shows a 3<sup>rd</sup>-order FIR structure known as the direct-form. This structure is programmable for both symmetric and non-symmetric impulse responses (optimisations for symmetric structures were disregarded in order to retain full programmability). A popular alternative implementation is the transposed-form (not illustrated), which has the advantage that the adder tree is pipelined without having to add additional registers, reducing the critical path and hence the structure is suitable for high performance implementations. The main limitation is the high fan out for the input register, especially in the case of a long filter. It is possible to use a hybrid version of the direct and transposed forms where there are some registers in the input line and some in the adder tree in order to balance the adder tree critical path and the input register fan-out.

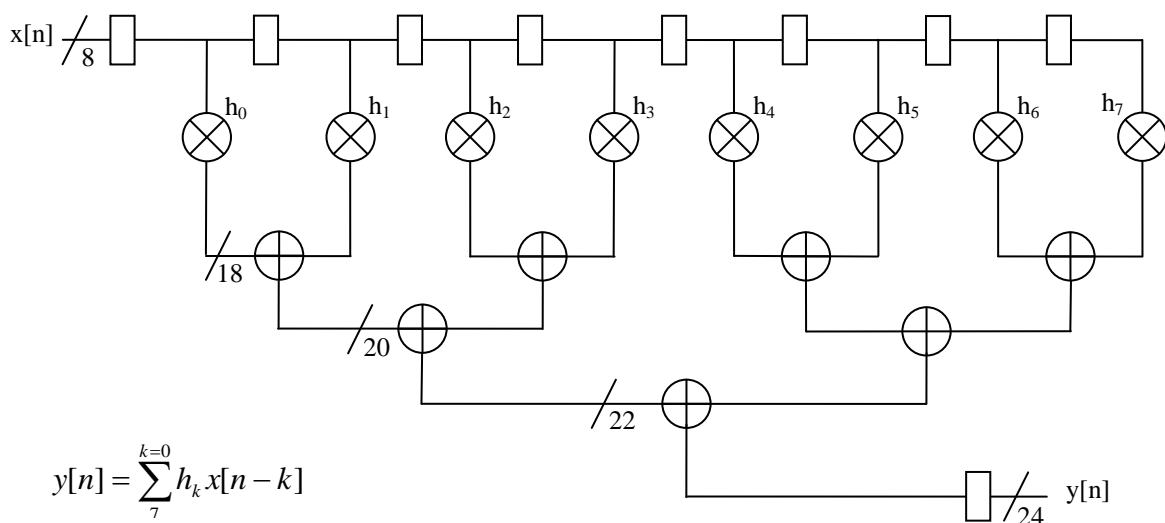


Figure 1: 7<sup>th</sup>-order direct-form FIR

## 2.1. Timing-Error Tolerant Architecture

The individual taps of the transposed form FIR are separated by registers, so timing-errors will occur independently. However, because the adder is still on the end of the critical path, even if only one tap experiences a timing-error, it will be added to all the other tap results along the pipeline and corrupt the output sum result. In order to adapt the transposed-form FIR such that the adder is no longer on the critical path, an additional pipeline register has been added to each tap after the multiplier (Figure 2). This forces the multiplier onto the end of the critical path, and significantly reduces the chances of an error occurring in the adder, since the critical path of the adder is substantially shorter than that of the multiplier for operands of similar precision. Hence, a single tap error will now no longer corrupt previous error results from other taps along the pipelined adder tree.

Instead of fully correcting incorrect tap results, which would incur a latency and power penalty, we take a heuristic approach that is appropriate to the specific requirements of DSP systems. We will investigate two simple techniques to mitigate the effect of errors occurring at the multiplier output. The first and most obvious is to zero the output of any multiplier that contains an error in the output result [2]. The second is to correct the sign bit (MSB) of any multiplier result that is found to contain an error. The MSBs can be considered to be the most susceptible to timing errors for the majority of classical arithmetic circuits that use LSB-first computation (e.g. ripple-carry adders and array multipliers). The MSB correction is trivial to calculate from the operand sign bits.

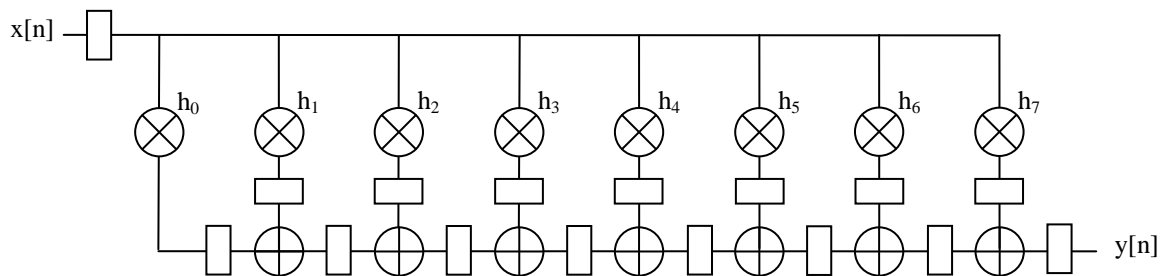


Figure 2: Modified 7<sup>th</sup>-order transposed direct-form FIR with additional pipeline stage

## 3. Simulation Results of Timing-Error Tolerant FIR Filter

To investigate the effect of over-clocking errors on FIR filters, a simple experiment was designed. A simulation testbed was developed containing a 7<sup>th</sup>-order FIR filter, with normalised cut-off frequency of  $\frac{1}{2}\pi$  radians/sample, 10-bit coefficients and 8-bit input data words. Full precision is maintained in the arithmetic to avoid the influence of rounding and truncation effects. The filter consists of multiplier taps implemented in an Altera Stratix-III 65nm FPGA library using the OEM tools. The Verilog netlist for the multiplier taps was elaborated in the simulator and back-annotated with the extracted cell delay information. Interconnect (net) delay was not modelled since this can be somewhat unpredictable in FPGA, especially for global routing to IOs.

The adder tree was not synthesized, but is modelled in behavioural RTL (and therefore will not generate errors). Each multiplier tap has a behavioural RTL copy (error free), used to calculate the correct tap result for error detection purposes. This represents ideal error detection and correction. Since the tools employed are not able to produce a SPICE netlist, it was not possible to investigate sub-critical voltage scaling in simulation to produce timing-errors. Instead over-clocking was used to induce equivalent errors in the simulation. This means we can only speculate on the reduction in power consumption that would be achieved with voltage scaling based on the over-clocked performance. Metastability and other issues arising due to transitions within the setup and hold time of the register are not modelled.

White Gaussian noise samples were used to test the filter frequency response up to the Nyquist frequency at  $\pi$  radians/sample. The critical clock period was found using a static analysis to be 5.5 ns. Starting from this point, where no errors occur and the response matches the theoretical, we scaled the clock period down to 4.2ns and measured the stopband rejection (SNR), taken as the ratio of pass-band power to stop-band power (both of which are calculated as the integral of the power spectral density).

The results are given in Figure 3, where the SNR is plotted along with the tap word error rate, which represents the average number of tap results exhibiting one or more bit errors per input sample (per clock cycle). We see that beyond about 5 ns the SNR degrades rapidly for the filter with no correction. The sign correction technique appreciably limits the degradation in SNR as we over-clock the filter. However, since we can only correct errors in the MSB, beyond 4.9 ns, when additional high-order bit-errors start to occur, the performance degrades at a

similar rate to the conventional filter. In the case of the tap zeroing technique, the effect of tap errors on SNR depends strongly on which coefficient is in error (and therefore set to zero for the current clock cycle). If a coefficient of small absolute magnitude is zeroed, the reduction in SNR can be negligible, but for a larger coefficient, the filter frequency shaping properties are significantly compromised. The tap zeroing technique performs exceptionally well; we are able to maintain a degradation in SNR of <5 dB down to a clock period of 4.3 ns. The concern with the tap zeroing technique centres on a potential marked loss in signal energy if many taps happen to be zero all at once. However, since we are not expecting to operate with a large error rate, this is not expected to be a problem.

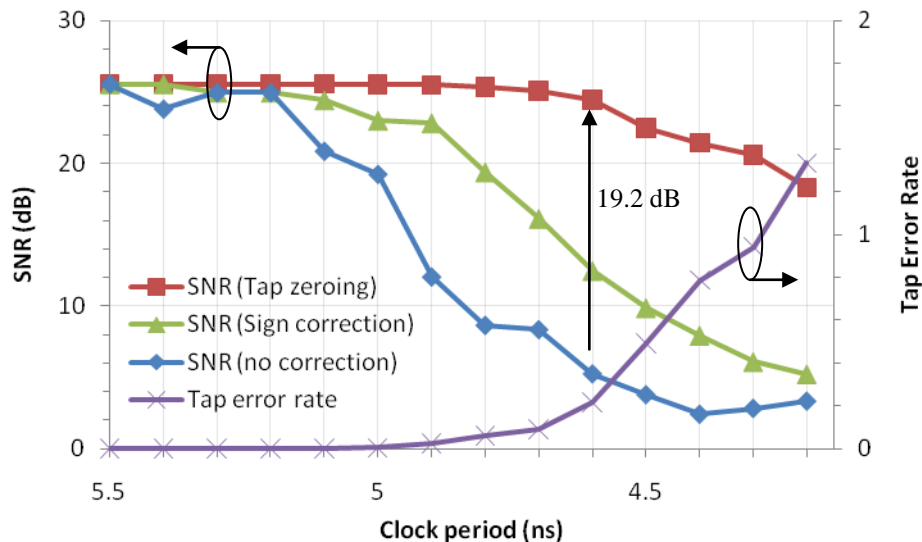


Figure 3: Over-clocked simulation results for proposed 7<sup>th</sup>-order FIR with simplified error correction

## 7. Conclusions

DSP architectures that offer improved robustness to timing-errors are of increasing interest to highly-integrated deep sub-micron ASIC design. Our motivation to study this area is low-power design through sub-critical voltage scaling, but many other issues related to yield, reliability and reducing design margins provide impetus to study logically robust implementation.

Considering a programmable FIR filter, we modified the transposed direct-form architecture to isolate potential timing-errors and then applied two simple error correction schemes. The design was simulated, using over-clocking to stress the architecture with timing-errors. Studying the stop-band rejection of the filter, we observed that both the employed techniques offer considerably improved robustness over the nominal implementation. In particular, the tap zeroing technique is able to maintain <1 dB degradation in stop-band rejection up to a 16 % reduction in clock period. This represents a 19.2 dB improvement in stop-band rejection over the nominal architecture in the presence of timing-errors.

The results presented here use ideal error detection, but this could be practically implemented at the circuit-level using shadow latches which sample on a delayed clock [3]. Future work will focus on examining the overhead of such an implementation and quantifying the energy savings that can be realised for a realistic design.

## Acknowledgments

The authors acknowledge ARM Ltd and EPSRC for supporting this research. Danny Kershaw and Dave Bull from ARM Ltd are also kindly acknowledged for their support.

## References

- [1] "Soft digital signal processing", R. Hegde and N. R. Shanbhag, IEEE Trans. on VLSI Systems, vol. 9, no. 6, pp. 813-823, Dec 2001.
- [2] "Variation-Aware Low-Power Synthesis Methodology for Fixed-Point FIR Filters", J.H Choi, N. Banerjee, K. Roy, IEEE Trans. on Computer-Aided Design of ICs and Systems, Vol. 28, Issue: 1, pp 87-97, Jan. 2009
- [3] "Razor: a low-power pipeline based on circuit-level timing speculation", D. Ernst, N. Kim, S. Das, S. Pant, R. Rao, T. Pham, C. Ziesler, D. Blaauw, T. Austin, K. Flautner, T. Mudge, Proceedings of 36th Annual IEEE/ACM International Symposium on Microarchitecture, 3-5 Dec. 2003