

Study and Simulation of an Example Redundant FIR Filter

Joakim Alvbrant and J Jacob Wikner

Department of Electrical Engineering

Linköping University

SE-581 83 Linköping, Sweden

{joakim.alvbrant, jacob.wikner}@liu.se

Abstract—In this paper we present a study and simulation results of the structure and design of a redundant finite-impulse response (FIR) filter. The filter has been selected as an illustrative example for biologically-inspired circuits, but the structure can be generalized to cover other signal processing systems. In the presented study, we elaborate on signal processing properties of the filter if we apply a redundant architecture where different computing paths can be utilized. An option is to utilize different computing paths as inspired by biological architectures (BIAs). We present typical simulation results for a low-pass filter illustrating the trade-offs and costs associated with this architecture.

BIA; Filters, Redundant signal processing, DSP

I. INTRODUCTION

Up until today the Moore's law, i.e., the amount of function and performance in an integrated device, is doubled every 18th month (or so) for the same area and cost, has been on average correct for systems on chip (SOC). This "law" has mostly been powered by the shrinking of the device sizes. However, rather soon, or even today, the sizes of the actual devices are determined by a few layers of atoms and encounters significant problems with accuracy and reliability [1].

To overcome these kinds of limitations, new techniques are constantly being developed on the manufacturing side [2]. Additionally, new architectures are employed to reach further, for example so called biologically inspired architectures (BIAs) have been suggested in many applications, e.g., wireless sensor networks, vision systems and parallel computing.

The definition of a BIA is vague, however, we conclude that in BIAs we have an increased amount of redundancy, a certain amount of adaptivity and evolution. We can allow many parallel processing elements, as in the nervous system [3]. In turn, these parallel processing elements are (in its context) "simple" and constitute a repetitive pattern. Traditionally, any BIA, e.g. artificial neuron networks (ANNs), are traditionally compared to the neurons and synapses in the brain: we have "simple" building blocks (the neurons) in a complex system (the brain) with a comparative simple interaction (spiking networks through connections of broadcasting type). Further on, the brain consists of repetitive patterns of interconnected neurons, for example the brain column [4].

Concludingly, in our opinion, using BIAs does not necessarily understand mimicking the (human) brain and its functions per se. Instead, we limit ourselves to, for example, structures where we have a relatively high probability of errors in the design process, but instead of discarding these samples we accept them and even take their influence into account in the design phase. Two such examples would be the future, increasingly inaccurate processing of integrated CMOS transistors, or say organic electronics. The transistor transfer characteristics become more and more statistic in nature.

With BIAs we also understand redundancy. If the circuits are faulty, they are faulty. This means that additional paths are required to "repair" the damage. With more redundant parts we have higher probability that we can switch to a functional path in case of dysfunc processing paths. Approaches like these, e.g., diversity schemes, are of course old strategies to improve reliability and quality of service. The cost is additional hardware and scheduling overhead. Power and throughput can be maintained as dysfunc paths can be turned off.

In the case of shrunken CMOS geometries we have however an interesting trade-off: the smaller the devices are, the more error-prone they are. At the same time though, we also see that the area penalty is less significant. We can trade the probability of subcomponent error against degree of redundancy (i.e. area overhead). This design strategy could result in simpler production tests (if any!).

In this paper we first outline in Section II the method and the model of a redundant FIR filter. Section III compiles the simulation results that illustrate the BIA system. In Section IV we discuss the results and associate with the method and model. Finally, the paper is concluded in Section V and some future work is suggested.

II. METHOD AND MODEL

In this section we describe the structure and architecture of the redundant finite-impulse response (FIR) filter. For example how the parallelization is performed at each processing node and how an averaging effect is obtained in summation nodes. An architectural view of a third-order FIR filter is given in Figure 1 with x being the input signal and y the output.

The nominal transfer function of an FIR filter, in the frequency domain, is given by

$$\frac{Y(\omega T)}{X(\omega T)} = H(\omega T, \vec{c}) = \sum_{i=0}^N c_i e^{-j\omega T i} \quad (1)$$

where $\vec{c} = (c_0, c_1, c_2, \dots, c_N)$ are the filter coefficients, N is the filter order and ωT is the normalized frequency with respect to the sample frequency of the filter, $f_s = 1/T$.

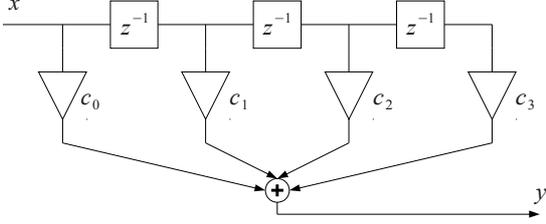


Figure 1: FIR filter architecture.

In order to clarify the method, we redraw the filter in Figure 1 slightly and present a more uniform view in Figure 2. By doing so the computational effort, assuming no special tricks for the multiplication yet, is the same for each sub block (dashed in the figure), here on called a processing element. As for any FIR filter we see that we can cascade with more processing elements to increase the order of the filter.

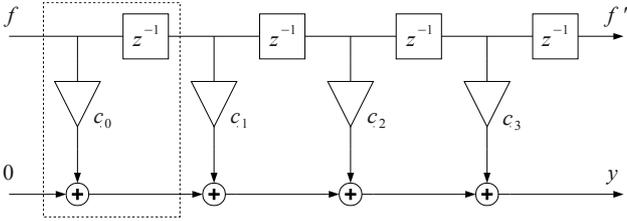


Figure 2: Same FIR filter but with the process element shown.

Towards making the system redundant we introduce several parallel paths. Let x_i be the number of parallel processing elements for each coefficient c_i . The resulting system is outlined in Figure 3 where we have indicated all possible paths between the first two processing elements only.

Each processing element consists of two sub blocks, the upper part, further referred as data path, and the lower, referred as filter path. Each sub block is assigned a stochastic binary variable to describe whether the block is functional or not. A "1" indicates if the block is functional and a "0" otherwise. For the data path the variables $r_{i,j}$ are used and can conveniently be ordered as elements in a matrix \vec{R} . Similarly, stochastic binary variables describing the function of elements in the filter path are $s_{i,j}$ and aligned in matrix \vec{S} . As can be seen in the first layer of Figure 3 the input to each processing element for coefficient c_i are the sum of all previous processing elements for coefficient c_{i-1} . Thus, to get correct scaling, the gain element a_i should be selected as $a_i = 1/x_{i-1}$ with $a_0 = 1$.

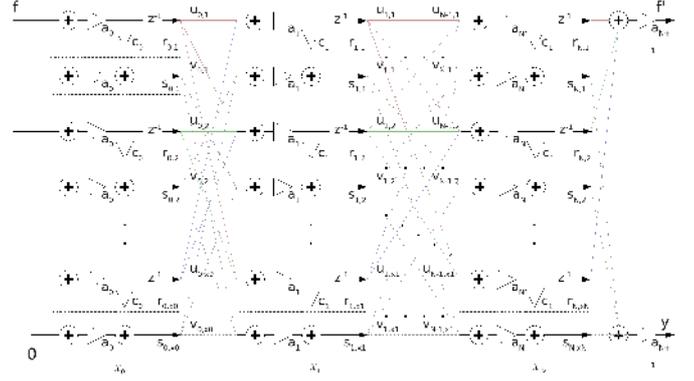


Figure 3: Illustration of the redundancy introduced in the FIR filter.

Realizing that all the input summations for all processing elements in each column are the same, we can redraw Figure 3 into Figure 4. The different terms have been collected.

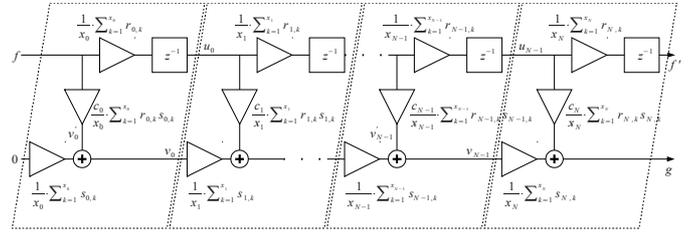


Figure 4: Effective system representation of redundant filter.

The amplitude of the data path in the n th cell can be expressed as

$$f_n = f \cdot \prod_{p=0}^n \frac{1}{x_p} \sum_{k=1}^{x_p} r_{p,k} z^{-1} \quad (2)$$

where $r_{p,k}$ is the probabilistic variable and x_p is ... The contribution of tap c_n in cell n to the filter path is then

$$v'_n = f \cdot \prod_{p=0}^{n-1} \left(\frac{1}{x_p} \sum_{k=1}^{x_p} r_{p,k} z^{-1} \right) \frac{c_n}{x_n} \sum_{q=1}^{x_n} r_{n,q} s_{n,q} \quad (3)$$

The contribution from cell n to the output g in the filter path can be written as

$$g_n = v'_n \prod_{m=n+1}^N \frac{1}{x_m} \sum_{a=1}^{x_m} s_{m,a} \quad (4)$$

Eventually, all contributions from the different taps are to be summarized and thus is

$$g = \sum_{n=0}^N g_n \quad (5)$$

Combining the equations gives a master expression as:

$$g = f \cdot \sum_{n=0}^N \prod_{p=0}^{n-1} \left(\frac{1}{x_p} \sum_{k=1}^{x_p} r_{p,k} z^{-1} \right) \dots \quad (6)$$

$$\frac{c_n}{x_n} \sum_{q=1}^{x_n} r_{n,q} s_{n,q} \prod_{m=n+1}^N \left(\frac{1}{x_m} \sum_{a=1}^{x_m} s_{m,a} \right)$$

and by setting $f=1$, and $z^{-1}=e^{-j\omega T}$, we get the filter's transfer function as

$$H(\omega T, \bar{c}, \bar{x}, \bar{R}, \bar{S}) = \sum_{n=0}^N \prod_{p=0}^{n-1} \left(\frac{1}{x_p} \sum_{k=1}^{x_p} r_{p,k} \frac{c_n}{x_n} e^{-j\omega T} \right) \dots \quad (7)$$

$$\sum_{q=1}^{x_n} r_{n,q} s_{n,q} \prod_{m=n+1}^N \left(\frac{1}{x_m} \sum_{a=1}^{x_m} s_{m,a} \right)$$

Given the design parameters \bar{c} and \bar{x} , together with a stochastic realization of matrices \bar{R} and \bar{S} , the transfer function can be calculated for the specific realization using (7).

When manufacturing integrated devices the degree of e.g. dust particles and other imperfections sets the probability of getting correct functionality. Since the imperfections tend to be small and hence local, the neighboring devices may very well be working correctly, it all comes to the ratio of the dust particle and the device. So, if the size of the imperfections is much smaller or in the same range as the implementation of a processing element a significant correlation between elements in the \bar{R} and \bar{S} matrices is unlikely, but if the imperfections is larger there is. If so, one may consider scrambling the mapping of the physical location of the processing element with the functional location in the filter to reduce correlation.

III. SIMULATION RESULTS

As an illustrative example we have chosen an FIR filter with a normalized -3 -dB cut-off frequency of

$$\omega_c T = \pi/2, \quad (8)$$

i.e., a half-band filter. With a stop-band ripple of approximately -55 dB a filter order of $N=20$ is required. Using (1), we show the ideal (nominal) transfer characteristics of the example FIR filter in Figure 5.

We also assume that the element in the matrix can either operate or not. It is also assumed that the probability of error is equally high in the data path as in the signal path. The probability of a dysfunc element is in turn given by a binary distribution with $P_r(\text{non-operational element}) = \epsilon$. A nonfunctional element is assumed to produce "0" (or at least constant) data, i.e., no random noise. In this work, we further choose $x_i = x, i \in [0, N]$, i.e, a symmetrical design where no optimization is chosen wrt. to the redundancy (x_i).

The simulation results shown in this work are generated from groups of statistical outcomes generated by MATLAB. For each group, we have used 2000 samples. Notice that with the example order we have in reality $2^{2 \cdot 20}$ possible cases, and it is not likely to capture the worst-case scenario.

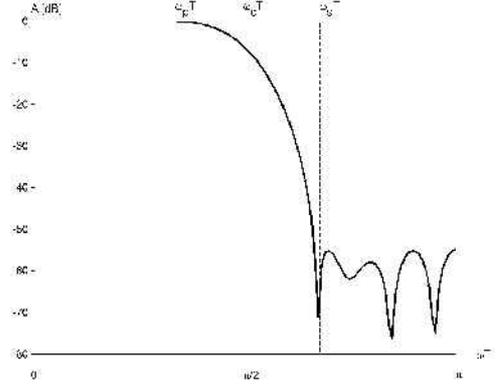


Figure 5: Gain characteristic of FIR-filter in normal case for reference.

Since the expectation value of the gain in each stage is $(1-\epsilon)$, and there are $N+1$ stages plus one conditional transition from data path to filter path, each coefficient C_n needs in total to be compensated by a factor of $(1-\epsilon)^{-(N+2)}$ to get correct gain in the passband.

As a performance measure we choose the ratio between the minimum passband gain and the maximum stopband gain in a certain band of interest, that is

$$A_{att} = \frac{A_{pass, min}}{A_{stop, max}} \quad (9)$$

within the normalized frequencies $[0, \omega_p T = 0.34 \cdot \pi]$ and $[\omega_s T = 0.68 \cdot \pi, \pi]$, respectively.

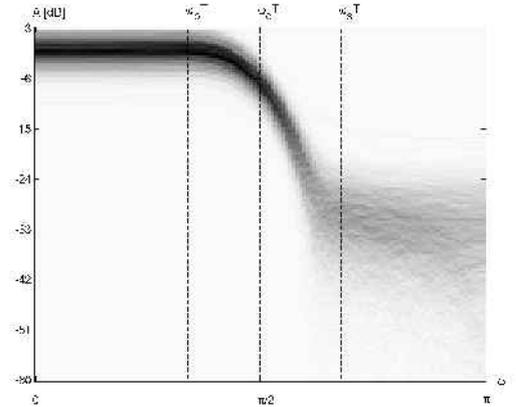


Figure 6: Example of gain-histogram (with $\epsilon=0.05$ and $x=20$). Dark areas represents high density, while light represents low.

With variations in components and weighting of the different signal branches -as from (7)- we get a histogram transfer function as shown in Figure 6 indicating the density of the transfer function. The expectation value of the graph is in turn identical to the transfer characteristics of Figure 5.

Using our cost measure from (9), the pass band density as such is interesting, i.e., the variation of the magnitude. In the stop band, the attenuation relative to the pass band in the worst-case is of interest, i.e., $A_{att,min}$.

Simulations were performed for different error probabilities, ϵ , whilst the number of parallel processing elements, x , was swept. In each point, an ensemble of 2000 realizations of the transfer function were calculated to estimate the minimum pass band attenuation, $A_{att,min}$. The results are found in Figure 7. Further, the standard deviation of the transfer function at $\omega_p T$ is plotted in Figure 8 where we see that the variance in the pass band decreases with increasing x .

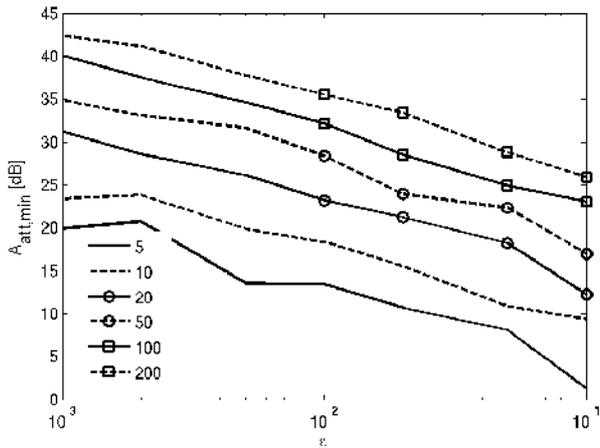


Figure 7: Filter performance given error probability and number of processing elements. Each point represents the worst case realization in each simulated ensemble.

IV. DISCUSSION

As seen in Figure 6, the pass band variations are rather well-centered with high density at center and are not unlikely to have a Gaussian-like distribution. The standard deviation of the pass band can be well-controlled by the amount of parallel processing elements as seen in Figure 8. It is especially effective for cases of high failure probability.

The passband relies on additions of terms from each stage in the filter and thus, the gain mismatch of each stage will be comparatively small to the ideal end-result. On the other hand, the stopband relies on cancellation of terms, and thus will accumulate mismatch from each stage and will be considerably larger. Thus, the stopband is much more sensitive as seen in Figure 6. These results align well with FIR filter design practice, where the stopband sensitivity effectively sets the required word length of the filter. The worst-case performance is controllable with the number of processing elements, as seen in Figure 7 and performance compared to the reference filter may be impossible to reach. We have limited us to have maximum of 200 parallel processing elements, due to limited

simulation time. The amount of redundancy required to get "almost-reference" performance can be studied in the future.

It can also be shown that the transfer function with respect to the filter coefficients, $\vec{c}=(c_0, c_1, c_2, \dots, c_N)$, is indeed convex which means that we can perform an optimization wrt. to redundancy and filter coefficients. The target is to find a simple and fast deterministic expression of the expectancy value of the deviation, from the desired filter response, in square, to get reasonable execution times.

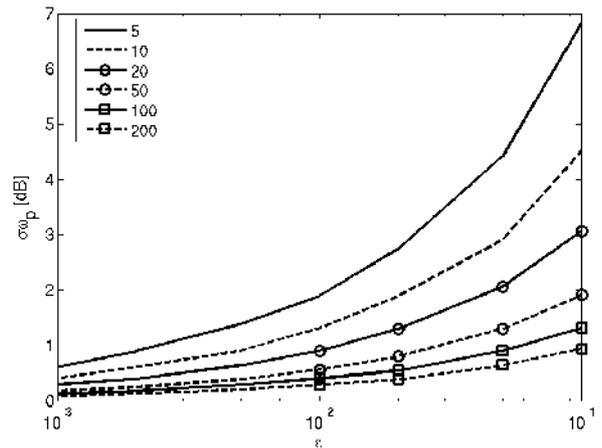


Figure 8: Standard deviation of filter response at $\omega_p T$ for different error probability number of processing element.

V. CONCLUSIONS

As a study on ways to utilize redundancy in an FIR filter, we have studied the effects of several signal paths in a low-pass filter and how dysfunc processing elements would effect performance. As is expected from filter theory, we mainly see variations in the pass and stop band attenuation and it was also verified that the stopband variation is higher, i.e., higher sensitivity.

REFERENCES

- [1] Lewyn, L.L.; Ytterdal, T.; Wulff, C.; Martin, K.; , "Analog Circuit Design in Nanoscale CMOS Technologies," *Proceedings of the IEEE* , vol.97, no.10, pp.1687-1714, Oct. 2009
- [2] Kuhn, K.J.; , "Moore's Law Past 32nm: Future Challenges in Device Scaling," *Computational Electronics, 2009. IWCE '09. 13th International Workshop on* , pp.1-6, 27-29 May 2009
- [3] Tortora, G.J. and Derrickson, B., *Principles of Anatomy and Physiology*, 12th ed. Hoboken, NJ: John Wiley & Sons, 2009
- [4] Guyton, A.C. and Hall, J.E., *Textbook of Medical Physiology*, 11th ed. Philadelphia, PA: Elsevier, 2006