Reconfigurable Processor for Real-Time Adaptive Sample Rate Notch Filtering

Richard H. Strandberg, Jean-Claude Le Duc*, Zsu-Yao Yang, Luis G. Bustamante
Vojin G. Oklobdzija and Michael A. Soderstrand
Department of Electrical and Computer Engineering
University of California
Davis, CA 95616

Abstract

Adaptive sample rate filtering has been shown to be effective in notch filter realization for removing narrowband push-to-talk interference. A prototype architecture is proposed for application to adaptive sample rate filtering which is capable of being reconfigured in real-time. This proposed architecture features Field Programmable Gate Array (FPGA) devices and ROM, and is to be demonstrated on audio signals using an XC4000 BORG prototyping board. A comparison is made between the computational complexity of the proposed architecture for adaptive sample rate filtering to that of traditional adaptive techniques which update filter coefficients. The results provide a measure of the advantages of the adaptive sample rate approach for notch filter applications.

1 Introduction

A significant problem in mobile communications is narrowband interference due to nearby terrestrial transmitters that cause difficulties in decoding spread-spectrum BPSK signals. Although spread-spectrum communication systems have inherent interference reduction (anti-jamming) capabilities [1], increased receiver performance is obtained by filtering out narrowband interference prior to pseudonoise correlation. Adaptive notch filters have been used to deal with this problem, but computationally intensive adaptation techniques are required to update the many filter coefficients needed to achieve sharp cut-off filter. Furthermore, in order to implement the updated coefficients, many general purpose multipliers must be used.

Adaptive sample rate filters [2] require the tuning of only a single parameter, the sample rate f_s, for frequency scaling of a fixed notch filter. Since fixed-coefficient filters can be implemented without multipliers [3], this technique results in computationally efficient low-cost implementations and can be used in real-time applications. Using filters with optimized *canonical signed digit* (CSD) coefficients [4] makes it possible to accomodate the higher order filters without the gate requirements of general purpose multipliers. The multiply-adds required in the adaptation can also be realized using an optimized CSD multiplication.

2 Adaptive Sample Rate Algorithm

A notch filter can be constructed by subtracting a narrow frequency band of a signal's frequency response from the complete frequency response, i.e.,

$$H_N(z) = 1 - H_{BPF}(z) \quad (1)$$

$$H_{BPF}(z) = \frac{1-r^2}{2}\frac{1-z^{-2}}{1-2rcos\theta z^{-1}+r^2z^{-2}} \quad (2)$$

where r is the pole radius, $\theta = 2\pi\frac{\omega}{\omega_s}$ is the angle of the poles on the unit circle where ω is the center frequency of the notch and ω_s is the sampling frequency.

Using the adaptive sample rate algorithm for a notch filter with a fixed center frequency of $\frac{\pi}{2}$ eliminates narrowband interferers between f_{low} and $f_{high} = 2f_{low}$. Depending on the location of the broadband signal, frequency conversion may be required to ensure elimination of interferers throughout the band of interest. Using a notch filter of width Δf, there will be a minimum of $M+1$ possible sample rates f_{si} given by:

$$M = \frac{f_{high}-f_{low}}{\Delta f} \quad (3)$$

$$f_{si} = 4f_{low} + 4i\Delta f \quad (4)$$

*Ecole Supérieure d'Ingénieur en Electrotechnique et Electronique, 93162 Noisy le Grand CEDEX FRANCE

where i is incremented from 0 to M. The sample rate is updated (by the LMS gradient) by changing the input N to a numerically controlled oscillator (NCO) as shown in Figure 1.

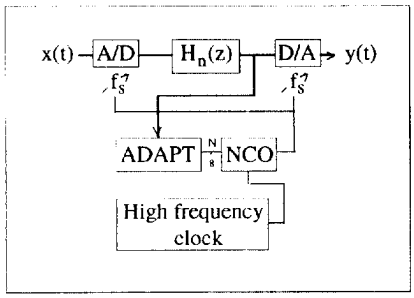

Figure 1: Adaptive Sample Rate Notch Filter

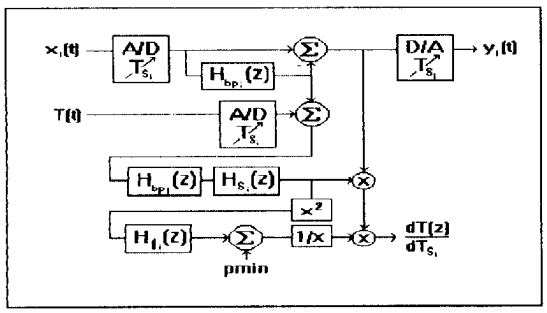

Figure 2: ASR cascadable notch section.

3 Development Platform

The XC4000 BORG board [5], (developed at the University of California, Santa Cruz) is a PC-based prototyping tool containing an array of four Xilinx user-programmable XC4000 FPGAs and is to be utilized as a development platform for our adaptive sample rate filter. This board was upgraded with Xilinx 10,000-gate XC4010 FPGAs to provide ample logic capacity for prototyping. Although the BORG board provides a flexible hardware platform facilitating rapid designs, it is unable to perform real-time reconfiguration of individual FPGAs. It configures FPGAs in a daisy wheel fashion, and thus, is unable to configure one Xilinx chip without disrupting the others. Through the use of schematic entry tools and Xilinx XACT tools, binary bit files are created and subsequently downloaded to configure the FPGAs.

4 Prototype Architecture

The basic scheme for the hardware architecture requires two Xilinx 4010DPC84 FPGAs to hold the components of the adaptive sample rate (ASR) cascadable notch filter section [6] shown in Figure 2.

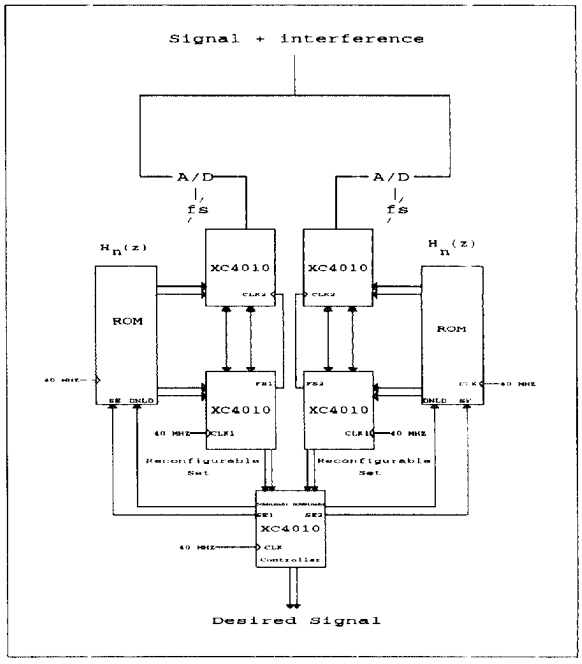

Figure 3: Proposed dual-channel reconfigurable filter.

Our proposed reconfigurable adaptive sample rate architecture will be clocked at 40 Mhz and will require four XC4010-5 FPGAs be routed to a controller and ROM containing downloadable code for a range of notch filters $H_N(z)$ as shown in Figure 3.

The controller will select two initial notch filters in parallel, one filter for each pair of FPGAs. The controller then downloads the code for the proper filter configuration to the FPGAs containing the poorer performing notch. Thus, a real-time reconfigurable processor with the ability to reconfigure at least two FPGAs separately is required. Xilinx reconfigurable XC4000/A/H FPGAs, for instance, can be reprogrammed on the order of milliseconds.

4.1 Functional Simulation

The adaptive filter to be demonstrated will use a notch filter centered at $\frac{\pi}{2}$. To examine the functionality of our hardware model, we clock in adaptively sampled data of pseudorandom BPSK and sinusoidal interference generated from our mathematical (MATLAB) model of the hardware and see if the hardware

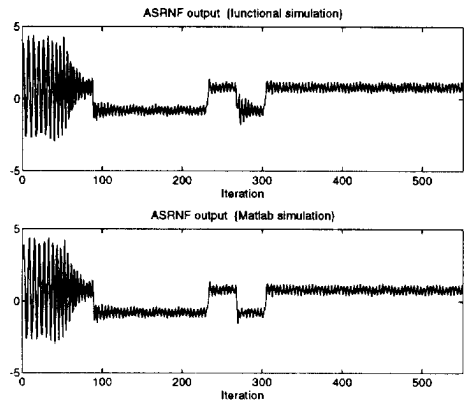

Figure 4: Notch output, functional simulation.

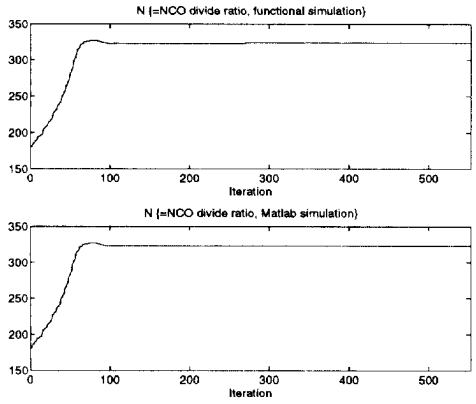

Figure 5: NCO divide ratio, functional simulation

simulator generates the proper value of N. With an 8.00 Mhz clock, the adaptive algorithim can eliminate narrowband interferers of frequencies between 5.5 Khz and 11.0 Khz. For a sinusoidal interferer of 6.215 Khz and using an 8.00 Mhz clock, the proper value of N should be 321.8. Results from both functional simulations (hardware simulator and mathematical model) are as shown in Figures 4 and 5. It is evident the NCO divide ratio N converged to the proper value.

4.2 Hybrid Simulation

Hybrid simulations were done where the first half of the cascadable section to (and including) the first multiplier were implemented on Xilinx XC4003APC84-6 and XC4010DPC84-6 FPGAs with the rest of the calculations done in software. For a sinusoidal interferer of 6.906 Khz added to a square wave and using an 8.00 Mhz clock, the proper value of N should be 289.6. Results comparing the hybrid simulation to the mathematical model are shown in Figures 6 and 7. Again, the NCO divide ratio N converged to its proper value. Note the convergence of N is smoother for the hybrid simulation (upper graph in figure) than for the mathematical (MATLAB) simulation (lower graph in figure). This smoothness is probably artificial since, in the hybrid simulation, calculations done in software were done at a greater resolution.

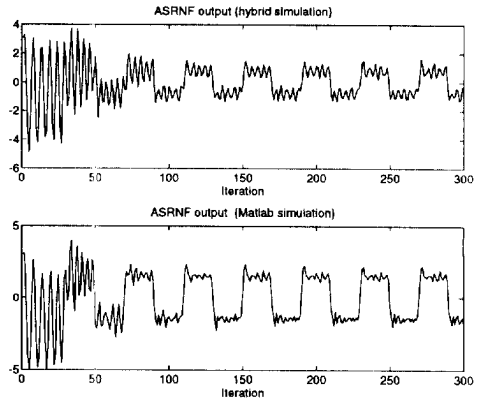

Figure 6: Notch output, hybrid simulation.

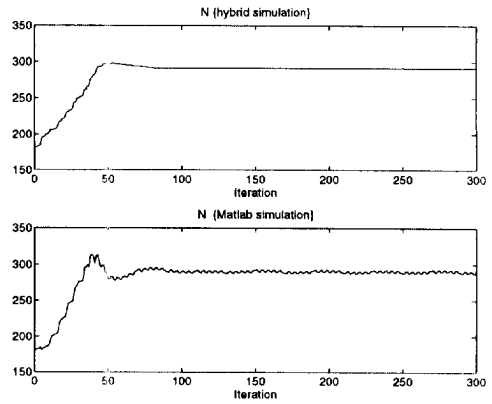

Figure 7: NCO divide ratio, hybrid simulation.

5 Conclusion

An adaptive sample rate notch filter has been developed and implemented in a hardware/software hybrid system. An entirely hardware approach would appear to work well in implementing an adaptive sample rate notch filter based upon functional simulations. Although currently the hardware for the cascadable adaptive sample rate notch filter section comprises approximately 25,000 gates, further minimization such as using the techniques discussed in the Introduction would enable the design to fit in two Xilinx XC4010DPC84-5 FPGAs. Since FPGAs are routinely clocked at higher frequencies such as 40 Mhz, if we load a series of notch filters from 0.1π to 0.9π and use values of N between 10 and 20, we can eliminate in-

terferers at frequencies between 100Khz and 1.8 Mhz.

After proving functionality of the design by demonstrating the elimination of audio interferers our design can be implemented as an application specific integrated circuit (ASIC). Using MOSIS 1.2 μm technology where sample rates of 100Mhz are possible using the above mentioned series of notch filters and values of N, we can eliminate interferers at frequencies between 250Khz and 4.5 Mhz.

References

[1] R. Scholtz, "The spread spectrum concept," *IEEE Transactions on Communications*, vol. 25, pp. 748–755, Aug. 1977.

[2] R. Strandberg, M. Soderstrand, and H. Loomis, "Elimination of narrow-band interference using adaptive sampling rate notch filters," in *Proceedings IEEE Asilomar Conference on Circuits, Systems and Computers*, (Pacific Grove, CA), pp. 861–865, Nov. 1992.

[3] R. H. Strandberg, J.-C. Le Duc, L. Bustamante, V. G. Oklobdzjia, and M. A. Soderstrand, "Implementation of adaptive sample rate kwan-martin notch filter using efficient realizations of reciprocal and squaring circuit," in *Proceedings IEEE Asilomar Conference on Circuits, Systems and Computers*, (Pacific Grove, CA), Oct. 1994.

[4] R. Hartley, "Optimization of canonic signed digit multipliers for filter design," in *Proceedings IEEE Internationl Symposium on Circuits and Systems*, (Singapore), pp. 1992–1995, June 1991.

[5] P. Chan, M. Schlag, and M. Martin, "BORG: A reconfigurable prototyping board using field-programmable gate arrays," in *First International ACM/SIGDA Workshop on Field-Programmable Gate Arrays*, (Berkeley, CA), pp. 47–51, Feb. 1992.

[6] M. Soderstrand, H. Loomis, and K. Rangarao, "Elimination of narrow-band interference in BPSK-modulated signal reception," in *Proceedings IEEE Internationl Symposium on Circuits and Systems*, (Singapore), pp. 2798–2801, June 1991.