Low-power adaptive filter based on RNS components

G.L. Bernocchi, G.C. Cardarilli, A. Del Re, A. Nannarelli* and M. Re

Department of Electronics, University of Rome Tor Vergata, Rome, Italy
*Dept. of Informatics & Math. Modelling, Technical University of Denmark, Kongens Lyngby, Denmark

***Abstract*— In this paper a low-power implementation of an adaptive FIR filter is presented. The filter is designed to meet the constraints of channel equalization for fixed wireless comunications that typically requires a large number of taps, but a serial updating of the filter coefficients, based on the Least Mean Squares (LMS) algorithm, is allowed. Previous work showed that the use of the Residue Number System (RNS) for the variable FIR filter grants advantages both in area and power consumption. On the other hand, the use of a binary serial implementation of the adaptation algorithm eliminates the need for complex scaling circuits in RNS. The advantages in terms of area and speed of the presented filter, with respect to its two's complement counterpart, are evaluated for implementations in standard cells.**

I. Introduction

Adaptive filters are often used for channel equalization in digital communication systems (for time-varying channels) and for echo cancellation in full-duplex communication systems [10], [11]. Different updating algorithms (LMS, RLS, FBLS) can be used depending on the performance of the required adaptive algorithm such as, for example, steady state error and convergence rapidity. Some classes of applications, require a relative small number of coefficients for the FIR filter and a fast adaptation. In this case, a parallel implementation of the updating algorithm is feasible due the small number of taps. Other applications requires a large number of taps, but block or serial updating of the filter coefficients is used since the channel variations in time are slow with respect to the data rate.

In this paper, a low-power implementation of an adaptive LMS FIR filter is presented. The adaptive filter has been designed to match the requirements of low rate time variable channels characterized by high distortion in amplitude and phase, as in the case of satellite links, power line carrier communications and beam-forming networks. A serial implementation of the LMS algorithm has been carried out, obtaining a reduction in terms of area and power consumption with respect to a parallel implementation, at the cost of a slower adaptation process.

The variable FIR filter has been implemented by using components implemented in the Residue Number System (RNS), since it has been proved that the RNS grants great advantages in terms of area, speed and power consumption over their two's complement counterparts in the implementation of application specific Digital Signal Processing (DSP) systems, and FIR filters in particular [6], [7], [8]. These advantages have been proved also for RNS adaptive filters [9], [5]. The proposed mixed RNS-Binary architecture achieves significant savings in both area and power consumption. Specifically, the use of a binary implementation of the adaptation algorithm avoids the use of complex and expensive scaling circuits in RNS, while the large variable FIR takes advantage of the RNS implementation. The paper is organized as follows: Section II presents the background material while in Section III the filter design is shown. In Section IV the adaptive filter architecture and the implementation results are presented, while in Section V the conclusions are drawn.

II. Background on Adaptive filtering

In adaptive filtering the variable filter keeps modifying its coefficients until the output signal $d^*(n)$ matches the desired signal $d(n)$. The modification of the coefficients is driven by the adaptive algorithm. A model of the adaptive system is sketched in Fig. 1. The input signal $d(n)$ is distorted by the channel, the Radio Frequency (RF) and by the acquisition part of the system. This distortion is modeled with a FIR filter characterized by a strong disequalization in amplitude and phase. Given the input $d(n)$, the signal $x(n)$ is expressed as

$$x(n) = \mathbf{h}_c^H \mathbf{d}_n. \tag{1}$$

The variable filter is a p order FIR with coefficients

$$\mathbf{w}_n = [w_n(0),\ w_n(1),\ \cdots,\ w_n(p)]^T \tag{2}$$

and the error signal is

$$e(n) = d(n) - d^*(n) \tag{3}$$

The variable filter estimates the desired signal by the convolution of the input signal with its impulse response. In vector notation this is expressed as

$$d^*(n) = \mathbf{w}_n^T \mathbf{x}(n) \tag{4}$$

where

$$\mathbf{x}(n) = [x(n),\ x(n-1),\ \cdots,\ x(n-p)]^T \tag{5}$$

is the input vector to the adaptive filter. The updated coefficients are

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \mathbf{\Delta w}_n. \tag{6}$$

The adaptive algorithm generates the correction factor $\mathbf{\Delta w}_n$ from the input and error signals. The selection of the algorithm

1-4244-0921-7/07 $25.00 © 2007 IEEE.

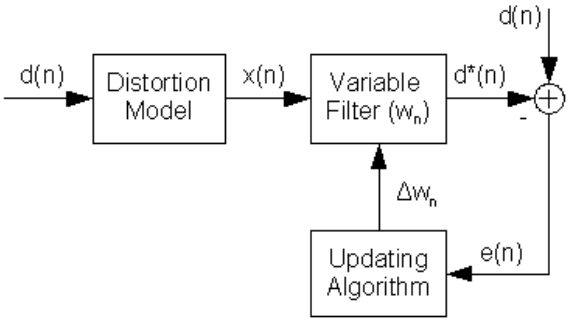


Fig. 1. Generic adaptive system model

TABLE I
SYSTEM REQUIREMENTS

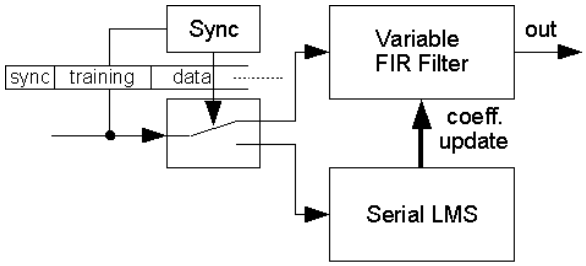
Description	Before equalization	After Equalization
Amplitude distortion	5 dB_{pp}	0.5 dB_{pp}
Phase distortion	10 $degree_{pp}$	2 $degree_{pp}$
Input sampling rate	160 Msps	

to be used (e.g. LMS, RLS, FBLS etc.) and the number of taps of the variable filter depend on system requirements, such as maximum input disequalization and residual output mismatch, convergence rapidity of the algorithm, etc. Some systems require a relative small number of coefficients and a small time for adaptation, allowing a parallel hardware implementation of the logic for coefficient upgrade. Other types of applications need a large number of taps in the FIR filter (high channel distortion), but allow for block or serial updating of the adaptive FIR filter coefficients, since the channel characteristic variations are slow with respect to the system clock.

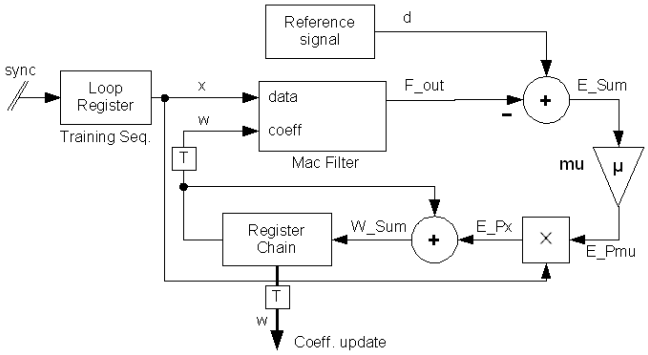
III. ADAPTIVE FILTER DESIGN

The design of the proposed adaptive filter is based on the typical constraints for slowly time-variable channel equalization systems, with particular focus on a beam-forming network equalizer for satellite applications. For this type of systems, the adaptive filter requirements are mainly related to the system specifications, such as the antenna beam orientation, the multi-beam and nulling creation, etc, and to the actual hardware implementation, such as the linearization of the acquisition chains (RF feeder, gain, phase distortion, ADCs disuniformity, etc). These requirements are translated in the amplitude and phase residual distortion after the equalizer.

The amplitude and phase distortions before and after the equalizer are summarized in Table I. Since the non stationary channel is varying slowly, a training sequence is used to update the filter coefficients. In a time-slot mode, as in GSM and theory of chaos-based communication systems [13], a specified amount of time is dedicated to the transmission of the training sequence, a particular signal known by the equalizer, and there is a trade-off between the communication efficiency and the



(a) Adaptive filter block diagram



(b) Detailed serial LMS block diagram

Fig. 2. Adaptive filter simulation model

equalizer performances. That is, the longer is the time elapsed between two consecutive training sequences, the greater is the communication efficiency (useful bits over total transmitted bits) at expenses of a poorer and slower convergence that results in a degradation of the system performance. On the other hand, if the time elapsed between two consecutive training sequences is very short, a poor communication efficiency is obtained.

The LMS algorithm has been chosen for the coefficient updating, because it is not very expensive in terms of hardware requirements with respect to other algorithms, and results sufficiently fast in tracking. The coefficient update vector $\mathbf{\Delta w}_n$ using the LMS algorithm is

$$\mathbf{\Delta w}_n = \mu \cdot \mathbf{x}(n) \cdot e(n). \tag{7}$$

where μ is a scaling factor chosen to normalize the error with respect to the input signal power.

In time-slotted trasmissions, new data for the equalizer are available only when the training sequence is received and, therefore, a sufficient amount of time is available for channel recovery between two consecutive training sequences. In our implementation, the new estimated coefficients are computed by using a serial architecture for the LMS algorithm, granting significative savings in power consumption, since one Multiply and Accumulate (MAC) unit is used instead of a parallel FIR filter. Moreover, the filter coefficients are updated one at a time, avoiding the use of a large number of multipliers for the parallel implementation of eq. 7 and obtaining complexity

TABLE II

BIT WIDTHS OBTAINED BY FXP SIMULATIONS.

Signal Name	Word Length	Signal Name	Word Length
x	12	F_prd*	24
d	12	F_acc*	30
w	12	F_out	12
mu	7	E_Sum	13
		E_Pmu	20
		E_Px	32
		W_sum	33

* *In MAC Filter.*

and power consuption reductions. In Fig. 2 the block diagram of the model used for the simulations of the adaptive filter is shown. The behavior of the system is described in the following:

1) once the training sequence has been detected, it is stored in the loop register;
2) the loop register feeds the MAC filter whose output is used to calculate the error signal $e(n)$;
3) the MAC filter coefficients, stored in the register chain, are updated one at a time by means of eq. 6;
4) once all the MAC filter coefficients have been updated they are used for the training sequence filtering;
5) the system restarts from point two, using the samples from the same training sequence until a new sequence is received;
6) when a new training sequence is detected, his samples are stored in the loop register, substituting the old ones, and the coefficients stored in the register chain are copied in the parallel FIR filter;
7) the system restarts from point two.

Floating Point (FLP) simulations have been carried out to define the number of the taps of the equalizer obtaining $N = 192$. The input samples wordlength $x(n)$ to the system is 12 bits, (mainly related to the Analog to Digital Converter (ADC) state of the art for space applications).

The number of bits to be used for the internal signals, obtained by Fixed Point (FXP) simulations, are shown in Tab. II. The number of bits used to represent the coefficients of the variable filter is $w = 12$, resulting in a dynamic range of the FIR filter of 32 bits. If the scaling factor μ is chosen to be a power of two, only one multiplication is needed for the serial implementation of the LMS algorithm.

The results of the fixed point simulations are shown in Fig. 3. The magnitude and group delay of the disequalized channel response (star marker), the reference channel (circle marker) and the output of the adaptive filter (bullet marker) show a good matching with the desired response both in magnitude and phase.

IV. IMPLEMENTATION OF THE VARIABLE FILTER

The Variable Filter of Fig. 2(a) is a high-order (192 taps) filter with a quite large dynamic range (32 bits). Previous work

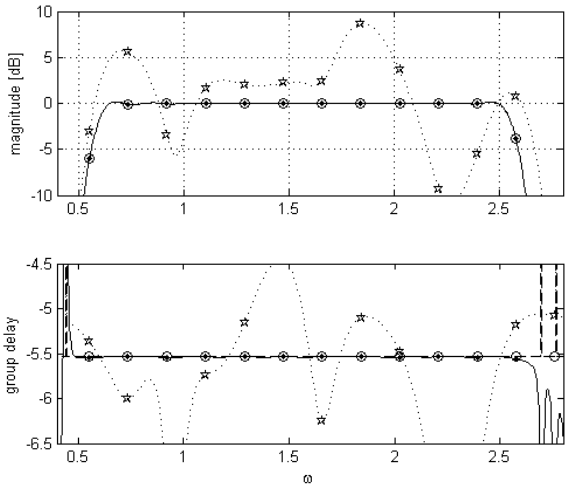


Fig. 3. Magnitude and phase response of signals in passband (simulation time: $5.5 \times 10^6\ T_{clk}$). $\star \cdots$ channel to be recovered; $\circ\!-\!-$ Reference channel; $\bullet\!-$ equalized channel.

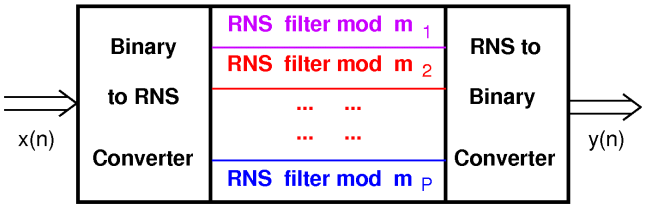


Fig. 4. FIR filter implemented in RNS.

on FIR filters ([14] and [6]) demonstrated that, for high-order filters, the implementation in RNS is convenient in terms of both area and power dissipation.

A Residue Number System is defined by a set of co-prime integers $\{m_1, m_2, \ldots, m_P\}$ which identify the base of the representation. Any integer X such that $0 \leq X \leq \Pi_{i=1}^{P} m_i$ has a unique representation in RNS. In RNS, operations such as addition and multiplication, can be performed in parallel paths, one for each modulus of the base. As a consequence, operations on large wordlengths can be split into several modular operations executed in parallel and with reduced wordlength [1].

A FIR filter is decomposed in a number of parallel filters of reduced wordlength as shown in Fig. 4 for a p-modulus base. The area and power dissipation overhead introduced by the conversions from the conventional two's complement (TCS) representation to the RNS and vice versa, is negligible for high-order filters (see [14] for more detail).

The characteristics of the RNS filters depend on the choice of moduli to cover the given dynamic range. In order to select the RNS base that gives the best tradeoffs delay/area/power, we used the tool of [12]. The selection of moduli done by the tool is based on the characterization of all the RNS components of the filter (modular adders and multipliers, converters, etc.)

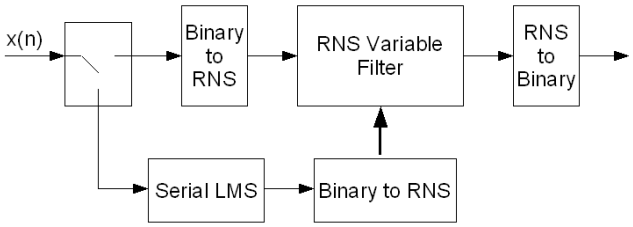


Fig. 5. Hybrid architecture.

performed for several timing conditions.

For the 192-tap and 32-bit dynamic range filter of this work, the tool selected the following set of moduli (RNS base): { 5, 7, 11, 13, 17, 19, 23, 128 }.

The architecture of the system is shown in Fig. 5. The variable filter has been implemented in RNS and the binary-to-RNS and a RNS-to-binary blocks are used for the conversions of the input and output signals. An additional binary-to-RNS block has been used to convert the updated binary coefficient from the serial LMS block. The additional latency introduced by the converters (two clock cycles per converter), does not affect the filter operations once the coefficients have been updated and the filter starts processing the signal.

Table III reports the characteristics of the Variable Filter implemented in RNS and in TCS (conventional FIR filter in two's complement). Both filters were synthesized by Synopsys Design Compiler by using the STM $90\ nm$ library of standard cells [15].

The data in Table III show that the implementation of the Variable Filter in RNS offers savings in area and power dissipation of about 50%.

V. Conclusion

In this work we investigated the implementation of a low-power adaptive filter, based on the use of binary components for the LMS algorithm and on RNS components for the variable filter. The filter has been designed to meet the constraints of channel equalization for fixed wireless communications that typically require a large number of taps but allow a serial updating of the filter coefficients. A comparison of the two's complement and the mixed component implementations has been carried out by synthesizing both filters by using Synopsys Design Compiler and the STM $90\ nm$ library of standard cells. The comparison shows that the mixed binary-RNS architecture offer relevant savings in area and power dissipation, without any degradation in performance.

Acknowledgment

This work was supported by University of Rome Tor Vergata, Department of Electronics and Denmark Technical University, Department of Informatics and Mathematical Modelling through visiting professor grants.

TABLE III

Results of filters implementations

	clock freq. $[MHz]$	Area $[mm^2]$	Power $[mW]$
TCS	200	3.1	215
RNS	200	1.7	125

References

[1] N.S. Szabo and R.I. Tanaka, "Residue Arithmetic and its Applications in Computer Technology", *New York: McGraw-Hill*, 1967. October 28 – November 1, 2005, Page(s):1620–1624.

[2] M.A. Sodestrand, W.K. Jenkins, G. A. Jullien, F. J. Taylor, "Residue Number System Arithmetic: Modern Applications in Digital Signal Processing", *New York: IEEE Press*, 1986.

[3] S.K. Mitra, J. F. Kaiser, "Handobook for Digital Signal Processing", *Wiley & Sons*, , 1993 Chapt. 15.

[4] S.K. Mitra, J. F. Kaiser, "Handobook for Digital Signal Processing", *Wiley & Sons*, 1993, Chapt. 9.

[5] W.K. Jenkins, B.A. Schnaufer, "Fault Tolerant Adaptive Filters Based on the Block LMS Algorithm", *IEEE International Symposium on Circuits and Systems*, 3-6 May 1993, Page(s):862 - 865.

[6] G.C. Cardarilli, A. Del Re, A. Nannarelli, M. Re, "Low Power and Low Leakage Implementation of RNS FIR Filters", *Conference Record of the Thirty-Ninth Asilomar Conference on Signals, Systems and Computers*, October 28 – November 1, 2005, Page(s):1620–1624.

[7] G.C. Cardarilli, A. Del Re, A. Nannarelli, M. Re, "Power characterization of digital filters implemented on FPGA", *IEEE International Symposium on Circuits and Systems*, ISCAS2002, Volume 5, 26–29 May 2002, Pages V. 801-804.

[8] A. Del Re, A. Nannarelli, M. Re, "Implementation of digital filters in carry-save residue number system", *Conference Record of the Thirty-Ninth Asilomar Conference on Signals, Systems and Computers*, Volume 2, 4–7 Nov. 2001 Pages 1309-1313.

[9] J. Ramirez, U. Meyer-Baese, A. Garcia and A. Lloris, "Design and Implementation of RNS-based Adaptive Filters", *Proceedings of the 13th International Conference on Field Programmable Logic and Applications*, Lisbon – Portugal, September 1–3, 2003, pp. 1135–1138.

[10] Y. Tsuda, T. Shimamura, "An improved NLMS algorithm for channel equalization", *IEEE International Symposium on Circuits and Systems*, ISCAS2002, Volume 5, 26–29 May 2002, Pages V. 353356.

[11] J.V. Lopez, J.C. Sanchez, H.P. Meana, "Adaptive echo canceller using a modified LMS algorithm" *Proceedings of the 2nd International Conference on Electrical and Electronics Engineering*, 7–9 Sept. 2005 Pages 93 96.

[12] A. Nannarelli, A. Del Re and M. Re, "A Tool for Automatic Generation of RTL-Level VHDL Description of RNS FIR Filters", *Proceedings of Design, Automation and Test in Europe Conference and Exhibition*, 16–20 Feb. 2004, Vol. 1, Pages 686-687.

[13] Wai M. Tam, Francis C.M. Lau and Chi K. Tse, "An Improved Multiple Access Scheme for Chaos-Based Digital Communications Using Adaptive Receivers", *Proceedings of the 2004 International Symposium on Circuits and Systems*, 2004. ISCAS '04. 23–26 May 2004, Vol. 4, Pages: 605–608.

[14] G. C. Cardarilli, A. Del Re, R. Lojacono, A. Nannarelli, and M. Re, "RNS Implementation of High Performance Filters for Satellite Demultiplexing," *Proc. of IEEE Aerospace Conference*, vol. 3, pp. 1365–1379, Mar. 2003.

[15] STMicroelectronics. 90nm CMOS090 Design Platform. [Online]. Available: *http://www.st.com/stonline/prodpres/dedicate/soc/asic/90plat.htm*