

MULTIRATE AS A HARDWARE PARADIGM

B. W. Suter

K. S. Stevens

S. R. Velazquez

T. Nguyen

Air Force Res. Lab/IFGC
525 Brooks Rd
Rome NY 13441

Intel
Strategic CAD Labs
Portland, OR 97124

V Company
388 Ocean Ave, Ste 1613
Revere Beach, MA 02151

Boston University
Elec. & Comp. Eng.
Boston, MA 02215

ABSTRACT

Architecture and circuit design are the two most effective means of reducing power in CMOS VLSI. Mathematical manipulations, based on applying ideas from multirate signal processing have been applied to create high performance, low power architectures. To illustrate this approach, two case studies are presented – one concerns the design of a fast Fourier transforms(FFT) device, while the other one is concerned with the design of analog-to-digital converters.

1. INTRODUCTION

While performance remains a primary figure of merit for any CMOS design, the increase in transistor count and smaller feature size of each process generation is elevating the importance of power, skew, increasing process variations, and increased capacitance of non-local communication. As design sizes increase, the ability to view a die as a unified circuit controlled by a single frequency becomes less viable. We feel that future architectures will be modular where each section contains its own frequency domain, and where they communicate not on bidirectional shared busses but via point-to-point unidirectional communication links. Such a design has significant power and potential performance advantages when compared to “traditional” architectures.

We decided to focus on new design approaches in order to investigate bringing formal mathematics of signal processing to bear on new design realities. A fast Fourier transform (FFT) architecture, which is efficient in terms of performance and power, has been created as a first case study. Then, to show the generality of this approach, an analog-to-digital converter design is considered as a second case study.

2. CASE STUDY: FAST FOURIER TRANSFORM

The following section describes a successful case study that was undertaken to utilize concepts from multirate signal processing and asynchronous circuit design to achieve a low power, high performance design. The fast Fourier transform

was chosen since it is an algorithm that requires globally shared results.

Our mathematical approach is hierarchically formed and expressed in terms of the $W_N = \exp(-j\frac{2\pi}{N})$ notation as shown in Equation 1. The derivation can be found in [7].

$$X_{m_1}(m_2) = \sum_{n_2=0}^{N_2-1} \left[W_N^{m_1 n_2} \sum_{n_1=0}^{N_1-1} x_{n_2}(n_1) W_{N_1}^{m_1 n_1} \right] W_{N_2}^{m_2 n_2} \quad (1)$$

This notation represents N_2 FFTs using N_1 values as the inner summation, which are scaled and then used to produce N_1 FFTs of N_2 values. The total operation achieves the desired FFT of size N .

Historically, equations for FFT systems similar to our approach have been developed for two applications. In the mid 1960’s the problem of computing the FFT of a vector that was too large to fit in main memory was addressed. An approach similar to that presented here was created to limit the storage requirements in these primitive systems[2]. A second similar approach was achieved in the 1980’s for multiprocessor applications of the FFT algorithm[1]. The underlying architectures created from these equations are vastly different than that achieved here.

The goals of this case study were to attempt to take a common application area and investigate novel formal architectural approaches to architect low power and high performance with additional constraints on what we project future designs will require. We therefore emphasized in our formulation pipelining, increasing localization, hierarchy, and establishing multiple frequency domains where we attempt to push the critical path into concurrent lower frequency domains to support high performance.

The multiplicative complexity of our approach is the same as the conventional Cooley-Tukey FFT formulation, which is $O(N \log N)$. But, our approach permits localized computations, as opposed to globally computing butterflies. This in turn suggests a low power silicon implementation, which is shown in Figure 1.

The multirate formulation of this algorithm has resulted in an implementation parallelized in a pipelined fashion.

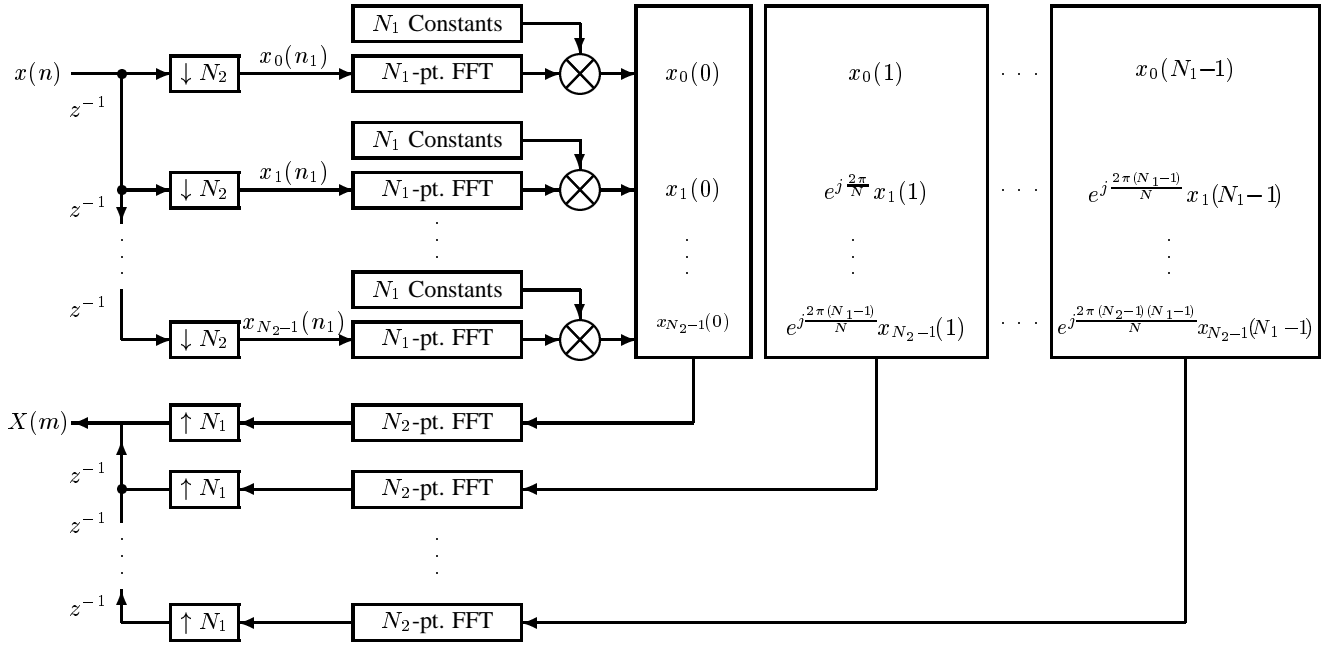


Figure 1: Low Power FFT Architecture

Each “row” in the architecture contains point-to-point uni-directional data pipelines. The entire design is implemented using asynchronous finite state machines for control.

The frequency of each horizontal track in the architecture operates at $\frac{1}{N_2}$ the frequency of the initial sample rate due to decimation – an average cycle time of 160ns using a 10ns sample rate. The asynchronous design methodology allows the rate division to occur locally with much of the circuit idle consuming only leakage current when the operation is complete.

The down arrow blocks of Figure 1 are decimators[6]. The output of the M-fold decimator for a sampled signal $X(n)$ is given by $y(n) = x(Mn)$. This is effectively a demux operation where each output is selected in order.

Each of the N_1 and N_2 blocks represent another FFT operation which can be a hierarchical instantiation of the structure in the figure where the values of $N_1 \times N_2$ equals N_1 or N_2 at the higher level in the hierarchy.

The product blocks multiply a stream of results coming from the N_1 point FFT units by a set of constant values. Both constants and results are complex numbers, requiring four multiplications and two additions per sample. The constants are calculated by $W_N^{m_1 n_2}$, where $m_1 = 0, \dots, N_1 - 1$ and $n_2 = 0, \dots, N_2 - 1$.

The large pipeline switch maps results from the product block to the N_2 FFT units. The N_2 FFT units take a transform of time displaced Fourier transform samples. Each N_1 -point FFT provides one data sample to each of the N_2 -point FFT units, the first row providing the first sample.

Chip	Power per transform
DSP-24 (DSP Arch.)	143 μ J / transform
SPIFFEE-1 (Stanford)	50 μ J / transform
space FFT	97 μ J / transform
earth FFT	18 μ J / transform

Table 1: FFT Power Comparison

Chip	Throughput
DSP-24 (DSP Arch.)	48k transforms/sec
SPIFFEE-1 (Stanford)	33k transforms/sec
Our design	25,000k transforms/sec

Table 2: FFT Performance Comparison

A stream of data $x_0(m_2), \dots, x_{N_2-1}(m_2)$ is output by the N_2 FFT units to an array of expanders[6]. The output frequency of the expanders increases N_1 fold, with each expander cell providing a single data sample.

This case study illustrates potential benefits of a mathematical approach to design for low power and implemented using self-timed methodologies. We have designed and submitted to MOSIS a circuit containing the FFT-4 logic using a radiation tolerant cell library in 0.8 μ m CMOS. The power consumption of the fabricated FFT-4 and completed implementation of the FFT-16 has been used to estimate the over-

all power efficiency of a 1024-point FFT. These results are shown in comparison with other FFT designs in Table 1. In addition to these reductions in power consumption, we achieved a remarkably high sustained throughput as can be observed in Table 2. All designs are measured using a 3.3V voltage source.

Figure 2 projects out a single “row” of the architecture presented in Figure 1. This shows the data path from input sample to output in a 256-point FFT ($N_1 = N_2 = 16$). Note that the data path switches between frequency domains of 100MHz, 6.26MHz, and 1.5625MHz. The higher frequency domains - such as input decimation and output expansion - are zones of reduced concurrency whereas the more concurrent operations function at decreased frequency. The ultra low frequency of the bulk of the circuit permits compact low energy circuit implementations, such as ripple-carry adders rather than look-ahead or array type adders, for the summation and product blocks.

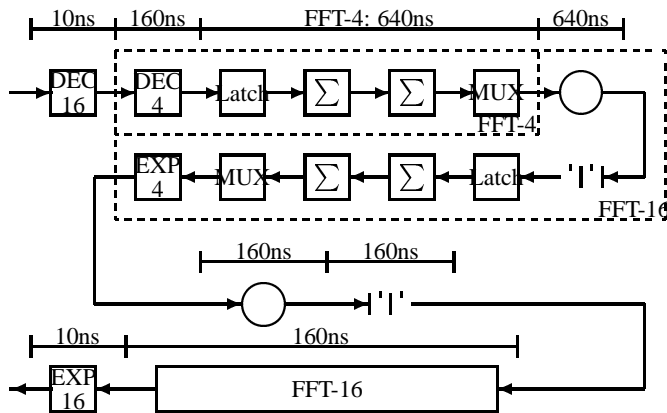


Figure 2: Data Path For Low Power FFT Architecture

3. CASE STUDY: ANALOG-TO-DIGITAL CONVERTER

The Advanced Filter Bank Analog-to-Digital Converter (AFB ADC) is a breakthrough approach to very high-speed, high-resolution analog-to-digital conversion which improves the speed of conversion by up to six times the state-of-the-art by using a parallel array of converters. The AFB ADC uses multirate filter bank signal processing to improve performance by using a combination of frequency-division and time-division multiplexing. Because of its unique architecture, the AFB ADC is continuously upgradeable as new analog-to-digital converter chips become available, thereby maintaining its performance advantage. The architecture can provide parallel channel outputs at lower data rates to ease the processing requirements of the digitized samples.

The architecture is amenable to a single-chip VLSI implementation to reduce size and power consumption. To further reduce power consumption, filters and channel ADCs can be implemented in low-power charge-domain processing[4]. We have built a 12-bit, 80 MSa/s hardware prototype for potential use in a universal, software-reconfigurable radio frequency (RF) receiver for cellular, satellite, or military communications. A 14-bit, 325 MSa/s system is currently being developed.

Using a filter bank for analog-to-digital conversion is an unconventional application of the filter bank architecture that improves the speed and resolution of the conversion over the standard Time-Interleaved array conversion technique[3]. The term “Advanced Filter Bank” denotes a filter bank with both analog and digital filters; conventional filter banks employ only discrete-time, digital filters. The AFB ADC employs analog Decomposition filters/splitters, D_k , to split the wideband analog input signal into M channel signals. The channel signals are sampled at $1/M$ the effective sample rate of the system and converted to digital signals with n -bit ADCs. The digitized channel signals are upsampled by M and reconstructed via the digital Recombination filters, R_k . The effective sample rate of the system is M times that of the channel ADCs in the array, and the resolution is n bits, the same as that of the channel ADCs in the array. The Advanced Filter Bank architecture is illustrated in Figure 3.

The AFB significantly improves the speed and resolution of the conversion by attenuating the effects of gain and phase mismatches between the converters in the array, which otherwise severely limit the resolution of the system[5]. The AFB is expected to provide analog-to-digital conversion with resolution up to 14 bits at sample rates up to 325 MSa/s, as shown in Figure 4.

The goal in the design of AFBs is to design filters that approximate the perfect reconstruction conditions as closely as possible: distortion should be small (e.g., less than a tenth of a dB deviation from ideal 0 dB) and aliasing error should not limit the resolution of the system (e.g., 85-90 dB for a 14 bit system). Given the Decomposition filters/splitters, $D_k(s)$, these perfect reconstruction constraints can be solved for the frequency response of the ideal Recombination filters, $R_k(z)$. An efficient Recombination filter design algorithm based upon the Fast Fourier Transform is developed in [8]. A discrete-time-to-continuous-time (“Z-to-S”) transform which converts a perfect reconstruction (PR) discrete-time filter bank into a near-perfect reconstruction AFB is also developed in [8].

The technical efficacy of the AFB ADC has been proven with a two-channel board-level prototype. The prototype provides full 12-bit performance at a sample rate of 80 MSa/s, twice as fast as the state-of-the-art Analog Devices AD9042 converter ICs upon which the prototype is based. The mea-

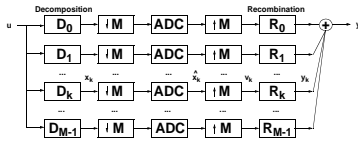


Figure 3: Advanced Filter Bank Analog-to Digital Converter Architecture

Resolution [bits]	Speed [MSa/s]	SFDR typ/min [dB]	SNR typ/min [dB]
12	80	82/74	68/62

Table 3: Measured Performance of AFB ADC Prototype

sured performance of the 12-bit AFB ADC prototype is shown in Table 3. The prototype is a two-channel architecture that employs low order analog filters/splitters and two length 32 digital FIR filters. The board-level prototype measures 10 cm by 13 cm. Note that the prototype can easily be upgraded to provide 12-bit performance at sample rates from 130 MSa/s to 260 MSa/s by incorporating two to four Analog Devices AD6640 12-bit, 65 MSa/s converter ICs. A 14-bit, 325 MSa/s system with dynamic range greater than 85 dB based upon the Lucent Technologies 65 MSa/s, 14-bit ADC integrated circuit (CSP1152A) is currently being developed.

4. CONCLUSIONS

Two examples of powerful multirate signal processing based architectures are discussed. The significantly decreased power consumption and dramatically increased throughput are the result of greater locality and increased parallelism.

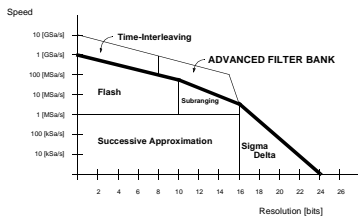


Figure 4: ADC designers face challenge of trading off the resolution(in bits) of its conversion with its speed (in samples per second). The thick line indicates the state-of-art in single chip ADCs.

5. REFERENCES

- [1] D. Bailey. FFTs in External or Hierarchical Memory. *Journal of Supercomputing*, 4:23–35, 1990.
- [2] W. Gentleman and G. Sande. Fast Fourier Transforms for Fun and Profit. In *AFIPS Conference Proceedings*, volume 29, pages 563–578, 1966.
- [3] A. Petraglia and S. K. Mitra. Analysis of Mismatch Effects Among A/D Converters in a Time-Interleaved Waveform Digitizer. *IEEE Trans. on Inst. and Meas.*, 40:831–835, 1991.
- [4] S. Paul. Analysis, design, and implementation of charge-to-digital converters. Technical report, Massachusetts Institute of Technology, Cambridge, MA, Master’s Thesis, 1995.
- [5] S. R. Velazquez. Hybrid filter banks for analog/digital conversion. Technical report, Massachusetts Institute of Technology, Cambridge, MA, PhD Dissertation, 1997.
- [6] Bruce W. Suter. *Multirate and Wavelet Signal Processing*. Academic Press, 1997.
- [7] Bruce W. Suter and Kenneth S. Stevens. Low Power, High Performance FFT Design. In *Proceedings of IMACS World Congress on Scientific Computation, Modeling, and Applied Mathematics*, pages 99–104, 1997.
- [8] S. Velazquez, T. Nguyen, and S. Broadstone. Design of Hybrid Filter Banks for Analog/Digital Conversion. *IEEE Trans. on Signal Processing*, 46:956–967, 1998.