1.2 GS/s Hadamard Transform Front-End For Compressive Sensing in 65nm CMOS

Osama U. Khan, David D. Wentzloff

University of Michigan, Ann Arbor, MI, USA.

***Abstract* — In this paper we present a digitally assisted front-end that performs analog computation of the Walsh Hadamard Transform (WHT) for GHz ADCs using compressive sensing to reduce power. The circuit consumes 11.2mW of power while sampling at 1.2GHz. The achieved compression rate assuming an ideal ADC is 59.4% with a mean square error of 0.3%. Comparison with the Nyquist ADC shows that using the sub-Nyquist ADC with the proposed WHT front-end reduces power by a factor of about 6x.**

***Index Terms* — Compressed sensing, sub-Nyquist ADC, Walsh Hadamard Transform.**

I. Introduction

Compressive Sensing (CS) theory states that given a signal is sparse in one domain (e.g. a pulse in the time domain); it can be sampled randomly in an orthogonal domain (e.g. the frequency domain) at a rate less than suggested by the Nyquist sampling theorem. The sparse signal can then be recovered with high probability from these compressed samples, but with an error proportional to the compression rate, by using a recovery algorithm (e.g. L1 minimization) [1]-[2].

Our signal of interest is a baseband ultra-wide-band (UWB) pulse with a width on the order of a nanosecond which is sparse in the time domain. The Nyquist sampling rate for a UWB pulse requires ADC sampling at a frequency greater than 500MHz, which usually results in prohibitively large power consumption in the ADC. This power can be a significant fraction of the total power consumption of the entire system. In [3] [4] [5] the power consumption for a 6-bit ADC with GHz sampling rate is greater than 150mW, which might be excessive for certain wireless applications. Sampling the UWB signal below the Nyquist rate, while maintaining a certain performance, may lead to a low power alternative solution.

There exist many potential orthogonal domains suitable for compressive sensing. In this paper, we implement the Walsh Hadamard Transform (WHT) as a measurement matrix to be used in CS. The reason for choosing the WHT is Walsh codes are a sequence of ±1's that multiply the time signal and the inversion can be easily implemented in hardware.

Fig. 1 shows the block diagram of a system exploiting CS and the discrete-time WHT front-end. In this work an Analog WHT is computed on the incoming sparse signal, the output of which is sub-Nyquist sampled by randomly choosing K samples out of N Nyquist samples. These compressed samples are then post processed by a recovery algorithm [6] which reconstructs the original sparse signal. Simulation was performed to compare the proposed system with a Nyquist ADC. An ideal WHT is computed on the sparse signal and the resulting Hadamard coefficients are quantized with finite resolution. It is found that for $K \geq N/3$ random samples, the Mean Square Error (MSE) between the original and the recovered signal is limited by the finite ADC resolution rather than the number of sparse samples, K. Further, for the same Mean Square Quantization Error (MSQE) a sub-Nyquist ADC quantizing a sparse time signal in the Hadamard domain can save one bit of resolution. Combining these two factors, a sub-Nyquist ADC with the same Figure of Merit (FoM) can be made lower power by a factor of about 6.

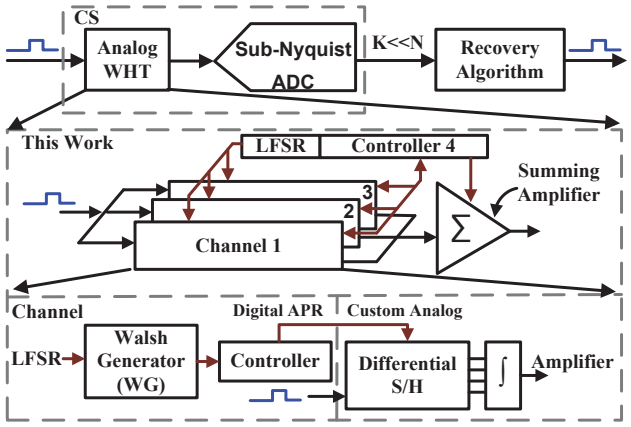

Fig. 1 Block diagram

II. Circuit Description

The circuit computes a 64-point WHT. The system comprises custom analog and synthesized digital logic. The function of the latter is to provide the necessary control (e.g. Walsh codes) to the analog blocks.

The Hadamard coefficients are the inner products of the input signal with the Walsh codes as shown in matrix form in Fig. 2. To compute the WHT the incoming signal is correlated with the Walsh codes using a GHz sampling rate and discrete-time integration. The speed requirement of the sampling and integration are met by splitting the correlation operation into three identical time-interleaved channels as shown in Fig. 1. A 6-bit LFSR generates a pseudo random number which is used by the on-chip

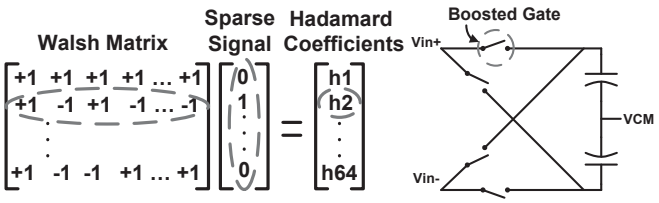

Fig. 2 Hadamard Transform Fig. 3 Input S/H

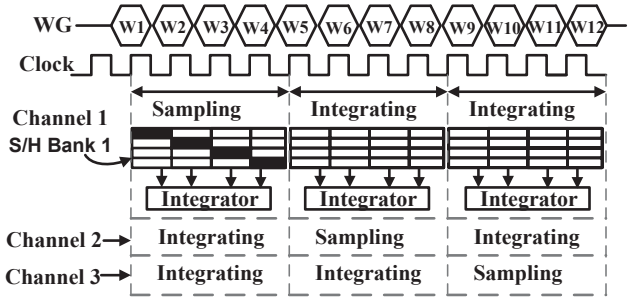

Fig. 4 Timing diagram

Walsh code generator (WG) [7] to generate the Walsh code of that sequency. The input is sampled with either positive or negative polarity depending on if it is multiplied with +1 or -1 in a Walsh sequence. The inversion is facilitated by the cross connections at the input sampling network as shown in Fig. 3. The sampling network is fully differential to mitigate charge injection and clock feed-through. Each channel accommodates four differential S/H circuits and hence the input is continuously sampled at the Nyquist rate by time-interleaving the 12 S/H circuits. All switches in the sampling network are implemented as NMOS switches with boosted gates to a supply voltage of 1.4V, which is provided by a level converter circuit.

Each channel has two phases, a sampling phase and an integration phase, controlled by state machines. Fig. 4 shows the timing diagram and a complete cycle for channel 1. During the first four clock cycles, Channel 1 is sampling the input. In the next four clock cycles, Channel 1 begins integrating its sampled values while Channel 2 enters the sampling mode. Similarly in the next four clock cycles, Channel 2 starts integrating while Channel 3 samples and Channel 1 completes the integration phase. The cycle repeats until the input signal is correlated with the 64-point generated Walsh sequence. The state machine then connects the output of the three channels to the summing amplifier followed by an output buffer. The output settling time is conservatively set to 32 clock cycles (26.7ns), after which the next random Hadamard coefficient is computed.

The integration time for each integrator is 6 clock cycles, or 5ns. In order to achieve this speed an integrator based on ZCBC (Zero Crossing Based Comparator) [8] is used as shown in Fig. 5. Since there is no closed-loop feedback, the ZCBC response time is fast but at the cost of overshoot at the output which needs to be compensated. For this purpose a binary weighted 7-bit current DAC is used which is enabled only during the charge transfer phase of the ZCBC.

III. Measurements

The chip is fabricated in a 65nm CMOS process. The active area is 0.1425mm^2. The total power consumption is 11.2mW. The dynamic digital power at 1.2GHz is 5.4mW while the analog section consumes 5.8mW. The measured Hadamard coefficients are shown in Fig. 6, along with an ideal Matlab computation of the coefficients (using the measured input pulse shown in Fig. 7.) for comparison. In order to exploit CS, K=26 random Hadamard coefficients are selected from the total N=64 measurements as shown in the figure. This results in a compression rate of $(N-K)/N$=59.4%. The compressed samples are then post processed in Matlab using a L1 minimization recovery algorithm and the input pulse is recovered with a mean square error of 0.3% as shown in Fig. 7.

The recovered pulse with and without CS are scaled in Matlab to match the input pulse amplitude for better comparison. The measurements clearly show that the accuracy requirements on the computed Hadamard coefficients are relaxed. Fig. 8 shows the measured overshoot calibration by the 7-bit current DAC of the ZCBC based integrator. The overshoot can be calibrated within ±15mV (differential) which is close to expected from simulated results. Fig. 9 shows the die photo.

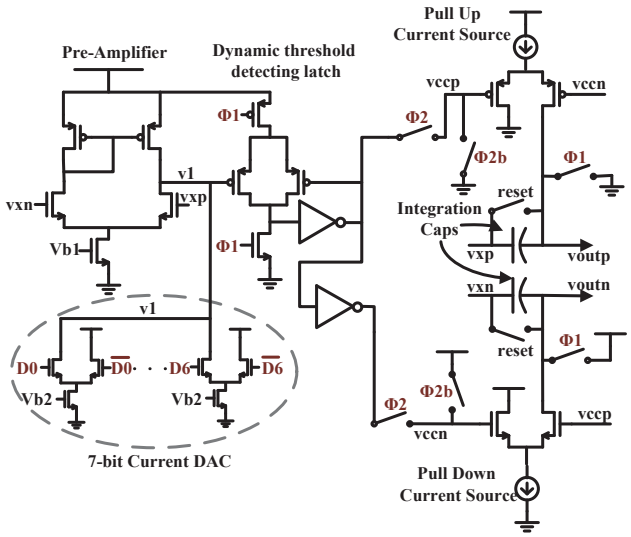

Fig. 5 ZCBC based Integrator

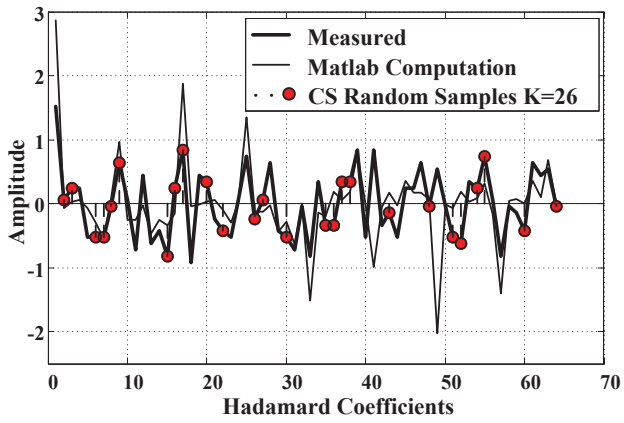

Fig. 6 Measured Hadamard coefficients

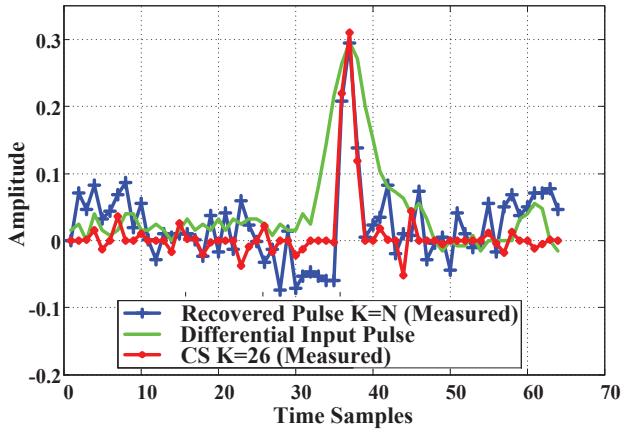

Fig. 7 Recovered pulse

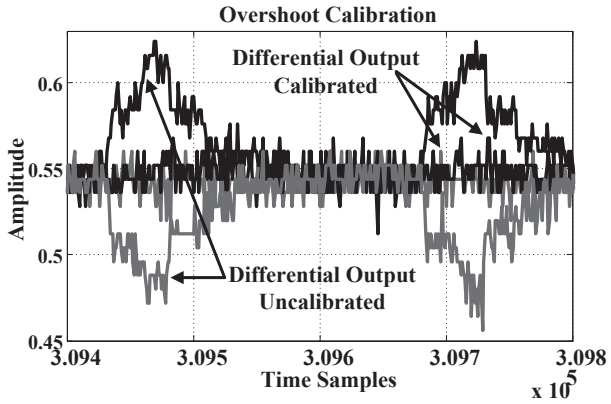

Fig. 8 Integrator overshoot calibration

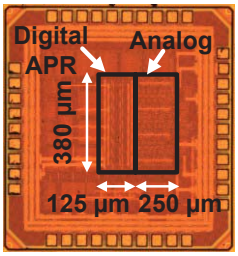

Fig. 9 Die Photo

IV. Conclusion

We have explored the application of compressed sensing to reduce power for GHz ADCs required for sampling ultra-wide-band (UWB) signals. It is found that the sub-Nyquist ADC with the proposed WHT front-end is about 6x lower power as compared to Nyquist ADC for the same FoM. The WHT front-end circuit consumes 11.2mW of power while sampling at 1.2GHz. The achieved compression rate assuming an ideal ADC is 59.4% with a mean square error of 0.3%.

Acknowledgement

This work is supported by the Signal Processing, Information Technology Program of the National Science Foundation under Award number CCF-0910765.

References

[1] Donoho, D.L. Compressed Sensing, *IEEE Trans. On Information Theory*, Vol. 52, No.4, April 2006.

[2] E. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?", *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406 –5425, dec. 2006.

[3] P. Scholtens and M. Vertregt, "A 6-b 1.6-Gsample/s flash ADC in 0.18µm CMOS using averaging termination," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 12, pp. 1599 – 1609, Dec 2002.

[4] K. Uyttenhove and M. Steyaert, "A 1.8-v 6-bit 1.3-GHz flash ADC in 0.25µm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 38, no. 7, pp. 1115 – 1122, July 2003.

[5] C. Sandner, M. Clara, A. Santner, T. Hartig, and F. Kuttner, "A 6-bit 1.2-Gs/s low-power flash-ADC in 0.13µm digital CMOS," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 7, pp. 1499 –1505, July 2005.

[6] SPGL1: A solver for large-scale sparse reconstruction, http://www.cs.ubc.ca/labs/scl/spgl1/

[7] Kitai, R., A Hazard-Free Walsh-Function Generator, *IEEE Trans. On Instrumentation and Measurements*, Feb. 1972.

[8] Brooks, L. A 12b 50MS/s Fully Differential Zero-Crossing Based ADC without CMFB, *International Solid State Circuit Conference* 2009.