

Analog vs. Digital Spatial Transforms: A Throughput, Power, and Area Comparison

Zephan M. Enciso¹, Seyed Hadi Mirfarshbafan², Oscar Castañeda²,
Clemens JS. Schaefer¹, Christoph Studer³, and Siddharth Joshi¹

¹Department of Computer Science and Engineering, University of Notre Dame, Notre Dame, IN, USA

²Department of Electrical and Computer Engineering, Cornell Tech, New York, NY, USA

³Department of Information Technology and Electrical Engineering, ETH Zurich, Zurich, Switzerland

e-mail: zenciso@nd.edu; sm2675@cornell.edu; oc66@cornell.edu; cschaefer@nd.edu; studer@ethz.ch; sjoshi2@nd.edu

Abstract—Spatial linear transforms that process multiple parallel analog signals to simplify downstream signal processing find widespread use in multi-antenna communication systems, machine learning inference, data compression, audio and ultrasound applications, among many others. In the past, a wide range of mixed-signal as well as digital spatial transform circuits have been proposed—it is, however, a longstanding question whether analog or digital transforms are superior in terms of throughput, power, and area. In this paper, we focus on Hadamard transforms and perform a systematic comparison of state-of-the-art analog and digital circuits implementing spatial transforms in the same 65 nm CMOS technology. We analyze the trade-offs between throughput, power, and area, and we identify regimes in which mixed-signal or digital Hadamard transforms are preferable. Our comparison reveals that (i) there is no clear winner and (ii) analog-to-digital conversion is often dominating area and energy efficiency—and not the spatial transform.

I. INTRODUCTION AND CONTRIBUTIONS

Sensing and processing multiple analog signal channels simultaneously is commonly encountered in a variety of fields including healthcare (ultrasound), multi-antenna communication, machine learning, imaging, and computer vision. Efficiently processing parallel streams of analog signals remains a challenging task due to the increasingly stringent latency and energy requirements imposed on the underlying hardware. Because spatial transforms, in contrast to spectral or time-interleaved transforms, have no temporal dependencies between inputs, they are highly amenable to parallel processing in area and energy efficient analog and digital circuits. This property of spatial transforms naturally raises the question of whether spatial transforms are more efficiently implemented using analog circuitry or through digital designs.

Previous work [1] indicates that analog spatial processing can be efficiently implemented using capacitor arrays. These results suggest that analog processing prior to digitization can relax the requirements of the analog-to-digital converters (ADCs), improving the system’s overall energy efficiency. Digital transforms come in various flavors, including streaming

and time-interleaved architectures; see, e.g., [2]. However, not much is known about the efficacy of massively-parallel transforms that are suitable for spatial processing of high-dimensional signals. Most importantly, to the best of our knowledge, no systematic comparison between analog and digital spatial transforms exists, which leaves the question of which of the two approaches is more beneficial in practice.

This paper represents a first attempt to systematically compare state-of-the-art analog and digital circuit designs with respect to area, throughput, and power for implementing spatial transforms. We focus on analog and digital circuits for spatial Hadamard transforms implemented in the same commercial, general-purpose 65 nm CMOS technology. We first detail the analog and digital circuit designs, provide reference post-layout implementation results, and compare their input and output signal-to-noise ratio (SNR) behaviors. We then study the area efficiency (area per throughput) and energy efficiency (power per throughput) trade-offs by considering the area and power of ADCs. Our comparison enables us to identify operation regimes for which analog or digital designs are preferable.

II. BACKGROUND

A. Hadamard Transform Basics

In order to compare analog vs. digital spatial transforms, we focus on the Hadamard transform (HT), which finds widespread use for data compression, compressive sensing, imaging, and locality sensitive hashing. The Hadamard transform is essentially a matrix-vector product of a Hadamard matrix \mathbf{H}_m by a vector $\mathbf{x} \in \mathbb{R}^M$ with $M = 2^m$, i.e., $\mathbf{y} = \mathbf{H}_m \mathbf{x}$. A Hadamard matrix \mathbf{H}_m of dimension $2^m \times 2^m$ can be constructed recursively. By defining $\mathbf{H}_0 = 1$, we can construct Hadamard matrices for natural numbers m as

$$\mathbf{H}_m = \frac{1}{\sqrt{2}} \begin{bmatrix} +\mathbf{H}_{m-1} & +\mathbf{H}_{m-1} \\ +\mathbf{H}_{m-1} & -\mathbf{H}_{m-1} \end{bmatrix}. \quad (1)$$

To avoid an explicit matrix-vector product that involves $M^2 - M$ additions and subtractions, one typically resorts to the fast Hadamard transform (FHT). The FHT repeatedly applies 2^{m-1} Hadamard transforms of size $m = 2$ (so-called radix-2 butterfly operations $\mathbf{y} = \mathbf{H}_2 \mathbf{x}$) in m stages as illustrated by the dataflow graph in Fig. 1. Note the scale factors $1/\sqrt{2}$, which

The work of SHM, OC, and CS was supported by ComSenTer, one of six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA. The work of OC and CS was also supported by Xilinx, Inc. and by the US NSF under grants ECCS-1408006, CCF-1535897, CCF-1652065, CNS-1717559, and ECCS-1824379.

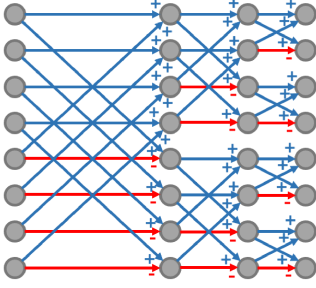


Fig. 1. Illustration of the dataflow graph of an $M = 8$ fast Hadamard transform (FHT). The FHT consists of $m = \log_2(M) = 3$ stages each performing $M/2$ two-dimensional Hadamard transforms on permuted inputs.

ensure that Euclidean norms are preserved, i.e., $\|y\| = \|x\|$, can be compensated either in every stage or at the end of the FHT; for the explicit Hadamard transform, the scale factors are typically included at the end of the matrix-vector product. The digital Hadamard transform implementation relies on the FHT, whereas the analog Hadamard transform effectively implements an explicit matrix-vector product using only capacitors.

B. Prior Analog/Mixed-Signal Spatial Transform

The analog circuit implementing HT closely follows the principles developed in previously fabricated mixed-signal spatial signal processing circuits [3]. This prototype implements analog matrix-vector multiplication using continuous-time multiplying digital-to-analog converters (MDACs) to form the matrix coefficients, which are then multiplied with differential analog inputs. Using capacitors in this fashion results in highly linear circuits that (i) weight the analog AC signals and (ii) linearly sum them onto a common node, resulting in 84 dB of signal separation performance for real-time beamforming of multiple-input multiple-output (MIMO) orthogonal frequency-division multiplexing (OFDM) signals [3]. Each capacitor in the MDAC uses a shielded structure in which, driven bottom and top plates shield the internal node from parasitics. By implementing continuous-time weighting of the analog signal, one mitigates capacitor switching and thus minimizes both CV^2 switching energy and kT/C noise. Consequently, capacitor sizing is primarily determined by matching requirements. We will describe a suitable analog HT design in Section III-A.

C. Prior Digital Spatial Transform

The fast Fourier transform (FFT) is among the most prominent digital spatial transforms and finds widespread use in communication systems, e.g., for beamspace processing [4]. FFT hardware design is an extremely mature area and state-of-the-art FFT designs can be generated automatically with SPIRAL [2]. In contrast, only a handful of custom FHT designs have been reported in the open literature; see, e.g., [5]. Existing parallel FHTs support relatively small dimensions (e.g. up to $M = 16$) and are typically applied to two-dimensional images for data compression. FHTs are extremely hardware-friendly as they only involve additions and subtractions. Furthermore, the simplicity of in-place processing minimizes the storage of sequential HT engines. Nevertheless, not much is known

TABLE I
POST-LAYOUT RESULTS FOR 128-POINT ANALOG HADAMARD TRANSFORMS WITH DIFFERENT UNIT CAPACITORS IN 65 NM CMOS

| C_{unit} [fF] | C_{unit} area [μm^2] | Array area [mm^2] | $f_{3\text{dB}}$ @ $14.4 \mu\text{S}$ g_{driver} [GHz] | Cap. mismatch σ_u/C_u [arb. unit] |
|---------------------------|---|---------------------------------|--|---|
| 0.68 | 2.25 | 0.078 | 4.65 | 0.06 |
| 1.5 | 4.41 | 0.153 | 2.55 | 0.024 |
| 2.0 | 5.76 | 0.200 | 2.03 | 0.016 |
| 4.0 | 10.24 | 0.356 | 1.1 | 0.01 |

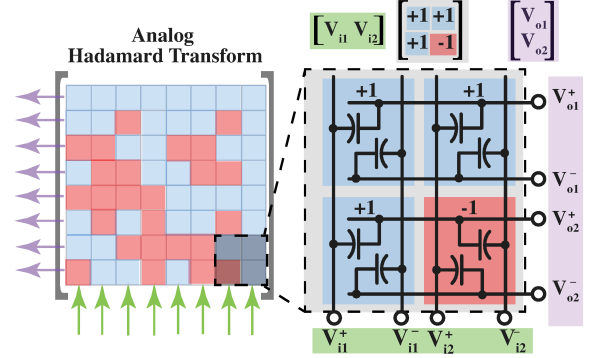


Fig. 2. Illustration of an 8×8 Hadamard transform matrix with details provided for a representative 2×2 sub-block. Differentially encoded inputs, V_{ij}^+ and V_{ij}^- , are either added or subtracted onto an output differential pair, V_{ok}^+ and V_{ok}^- , through capacitive coupling. The addition/subtraction occurs when the Hadamard transform matrix entry is a $+1/-1$.

for larger Hadamard transforms that are suitable for spatial processing. We will describe a digital FHT design suitable for spatial processing in Section III-B.

III. IMPLEMENTATION DETAILS

A. Mixed-Signal Implementation

Our analog HT implements a 128×128 HT matrix using a differential capacitor structure as shown in Fig. 2. The inputs and the outputs of this block are continuous-time, differential analog signals. Since the HT is a fixed-transform, this leads to a compact cell which is then repeatedly tiled in layout, each bottom plate is driven by one of the polarities of the differential signals. We place two complementary instances of the array to ensure that both polarities of the signal see a constant capacitive load. The capacitor array was laid out in TSMC 65 nm CMOS, with the capacitors occupying metal layers 4, 5, and 6. The area, maximum frequency, and unit capacitor values entered in Table I are from post-extraction simulation, they were verified against 10b to 14b data converters [3], [6] that we previously taped out. To derive a realistic cut-off frequency for the system, we set the output conductance of the array drivers to $14 \mu\text{S}$. Table I summarizes the design across multiple array sizes. When $C_{\text{unit}} = 4 \text{ fF}$ the HT capacitor array size is comparable to the digital implementations in Table II. Aggressive scaling of the unit capacitors to sub-femto-farad results in $f_{3\text{dB}} > 4 \text{ GHz}$ and consequently a $f_{\text{nyq}} > 8 \text{ GHz}$.

B. Digital Architecture and Implementation

Our digital FHT implements a fully-unrolled decimation in frequency architecture using radix-2 butterflies, as illustrated in

TABLE II
POST-LAYOUT RESULTS FOR 128-POINT DIGITAL FAST HADAMARD TRANSFORMS (FHTs) WITH 5B TO 10B INPUT PRECISION IN 65 NM CMOS

| Input res. [bit] | Area [mm ²] | Max. freq. [GHz] | Power [mW] | Area eff. [mm ² /GT/s] | Energy eff. [pJ/T] |
|---------------------|----------------------------|---------------------|---------------|--------------------------------------|-----------------------|
| 5 | 0.195 | 1.603 | 346.7 | 0.122 | 216.4 |
| 6 | 0.236 | 1.605 | 431.4 | 0.147 | 268.8 |
| 7 | 0.277 | 1.439 | 440.6 | 0.192 | 306.2 |
| 8 | 0.314 | 1.429 | 517.0 | 0.219 | 361.9 |
| 9 | 0.341 | 1.431 | 575.9 | 0.239 | 402.5 |
| 10 | 0.394 | 1.377 | 617.1 | 0.287 | 448.0 |

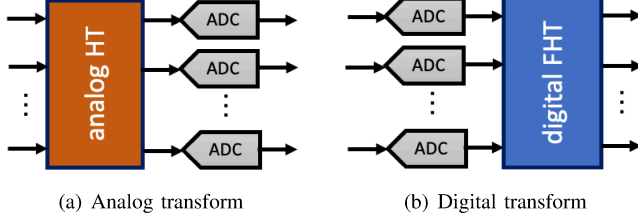


Fig. 3. Comparison methodology. For the analog transform, we first apply the Hadamard transform using passive, capacitor circuits followed by converting the analog signal using 128 ADCs; for the digital transform, we first use 128 ADCs followed by the digital fast Hadamard transform (FHT).

Fig. 1. The 128-point FHT implementation consists of $m = 7$ stages, where each stage contains 64 radix-2 butterflies that perform addition and subtraction of the two inputs. Since the output bitwidth of an adder/subtractor is one bit more than that of its input, we allow the odd-numbered stages to increase the bitwidth by one—the even-numbered stages apply a scale factor of $\frac{1}{2}$, thereby maintaining the bitwidth. Consequently, the outputs of the design have only 4b more resolution than the inputs, which reduces area and ensures proper normalization of the FHT. In order to minimize the critical path of our FHT design, the outputs of each stage are pipelined.

Table II shows post-layout results for 128-point FHTs ranging from 5b to 10b input precision in TSMC 65 nm CMOS. We note that these are—to the best of our knowledge—the first implementation results of digital 128-point Hadamard transforms reported in the open literature. The cell density is around 80% for all digital designs. Since our architecture is fully unrolled and pipelined, the maximum sustained throughput (in transforms per second) equals the maximum clock frequency. The area and net power consumption scale roughly linearly with the number of input bits and the precision has a marginal effect on the maximum clock frequency.

IV. COMPARISON

A. Methodology

Fig. 3 illustrates the comparison methodology used in this paper. In order to arrive at a fair comparison between both approaches, we include the area and power of analog-to-digital converters (ADCs) that would otherwise be present in a real-world system. Additionally we account for signal attenuation incurred during the analog transform (11.3 dB for the 128-point HT), by correspondingly increasing the SNR requirement from the downstream ADC. To this end, for the analog transform,

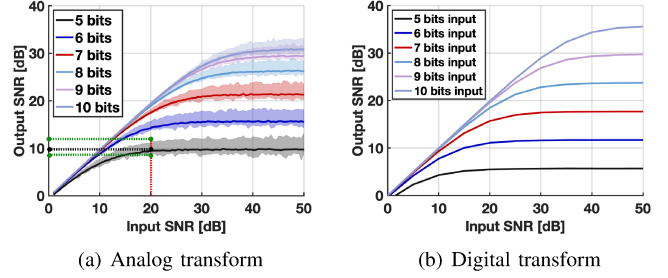


Fig. 4. Input vs. output SNR for analog and digital Hadamard transforms. (a) Shows the effect of quantization and capacitor mismatch for the analog HT implemented using a capacitor array composed of 0.68 fF unit capacitors. The shaded area represents the spread in achievable output SNR with a solid line, representing the lowest point for a 90% yield. At an input SNR of 20 dB the spread in output SNR due to mismatch is highlighted by the dotted lines. (b) Shows the output precision of the digital FHT design.

we first use the analog Hadamard transform design detailed in Section III-A followed by a dedicated ADC for each of the 128 analog outputs. For the digital transform, we first use a set of 128 ADCs to convert the analog inputs followed by the digital FHT design. For both transform designs, we pick ADCs from [7] that match the resolution with signal-to-quantization-noise ratio (SQNR) of the analog or digital transform, as well as the maximum achievable bandwidth by the individual designs.

B. Input SNR vs. Output SNR

As a first step, we study the accuracy and linearity of the two approaches. To characterize the input and output SNR, we consider the input signal model $\mathbf{x} = \mathbf{s} + \mathbf{n}$, where \mathbf{s} is the signal vector and \mathbf{n} is the noise vector; both are i.i.d. zero-mean Gaussian. The signal and noise variances are determined by input SNR. We then measure the output SNR as

$$SNR_{\text{out}} = \frac{\mathbb{E}[\|\mathbf{y}\|^2]}{\mathbb{E}[\|\mathbf{y} - \hat{\mathbf{y}}\|^2]}, \quad (2)$$

where $\mathbf{y} = \mathbf{H}\mathbf{s}$ is the output of an ideal, noise-free Hadamard transform and $\hat{\mathbf{y}}$ is the quantized output of transforming $\mathbf{x} = \mathbf{s} + \mathbf{n}$ using either the analog HT or the digital FHT.

For the analog design, we consider the effect of capacitor mismatch on the HT. All analog HT results were extracted from 400 Monte-Carlo trials of capacitor mismatch with 400 trials per SNR. Using the methodology described in [8] and our fabricated IC [3], we estimate the mismatch coefficient for the capacitors to be $A = 2\%\sqrt{1 \text{ fF}}$.

Fig. 4(a) shows the effect of this mismatch for $C_{\text{unit}} = 0.68 \text{ fF}$ on the SNR of a transformed signal, for various output ADC resolutions. At a target input SNR of 20 dB, the mismatch creates a spread of possible values; the dotted lines in Fig. 4(a) indicate the maximum and minimum output SNRs observed over 400 Monte-Carlo trials for an input SNR of 20 dB. For the digital transform, we use a bit-true golden model to extract the output SNR via Monte-Carlo simulations. Fig. 4(b) shows the SNR transfer behavior of the digital FHT. We observe that the output SNR is lower than that of the analog transform for less than 7b input resolution—for higher resolution, the digital FHT achieves higher output SNR.

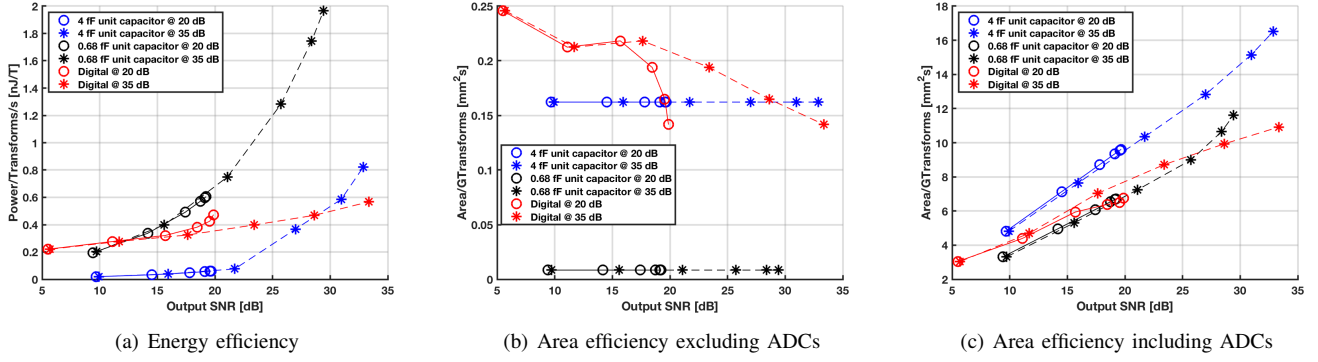


Fig. 5. Energy and area efficiency vs. output SNR trade-offs. (a) Although the analog design with 0.68 fF unit capacitors achieves higher f_{3dB} , operating at such frequencies requires expensive ADCs, which annihilate the benefit of compact analog circuitry. The analog design with 4 fF unit capacitors achieves lower f_{3dB} , which is conducive to power efficient ADCs. For the digital FHT, the ADC power is comparable to that of the digital part. (b) Shows the area efficiency without the ADC area, which reveals that analog transforms can be more compact and suffer from no area increase due to the fixed array size. (c) Shows the area efficiency with the ADC area, which shows that the ADC area is substantial, effectively resulting in designs of comparable efficiency.

C. Area-efficiency and Energy-efficiency Trade-offs

Fig. 5(a) compares the energy efficiency obtained from two analog configurations (with unit capacitors 4 fF and 0.68 fF) and the digital implementations. While the analog HT design with the smaller unit capacitor operates at a higher bandwidth, the energy and area overheads of high-frequency ADCs are detrimental to the combined system efficiency. Indeed, the 4 fF array shows superior energy efficiency than the 0.68 fF array, primarily due to a more energy-efficient ADC. As expected, at higher resolutions (output SNR ≥ 30 dB), the digital design is more energy-efficient. Examining the energy contribution of the ADCs shows that the ADC power is comparable to the power of the digital FHT power, but it dominates the power of the analog HT. This disparity is explained by the ADC SNDR increasing by 12 dB to compensate for capacitor induced attenuation in the analog signal path (insertion loss).

Fig. 5(b) compares the area efficiency of the three designs, where we exclude the ADC area. In this comparison, the analog circuits are much more area efficient, with the smaller array ($C_{unit} = 0.68$ fF) delivering an order of magnitude higher throughput than the digital FHT. However, when ADC area is included in the comparison, Fig. 5(c) reveals that this advantage is immediately negated. Indeed, the area efficiency for all three designs now becomes comparable, in part due to the costly ADCs required for high-speed operation. Moreover, we cannot identify a clear design point that is better across categories, i.e., while the slower operation due to larger capacitors leads to improved energy efficiency, the larger area also reduces throughput. As expected, the digital FHT is consistently better than analog HTs at very high resolution—when ADC overheads are completely accounted for.

V. CONCLUSIONS AND OUTLOOK

We studied the area and energy efficiency of implementing spatial Hadamard transforms through passive analog circuits and massively-parallel digital circuits. All of our designs have been implemented in the same 65 nm CMOS technology. Our analysis reveals that neither design is an outright winner in

all categories. We note that the Hadamard transform uniquely advantages the analog design, leading to extremely compact and energy-efficient implementations. Despite this, our analysis reveals that the ADCs heavily influence the overall area and energy efficiency of spatial Hadamard transforms, indicating that further optimizations must include data converter design. For analog spatial transforms to truly deliver, we would need: (i) the ADC to be co-designed with the analog processing and (ii) circuit topologies that exploit transform sparsity must be employed to minimize insertion loss. Finally, an extensive comparison between analog and digital spatial Fourier transforms, which are useful for emerging millimeter-wave communications systems, is part of future work.

REFERENCES

- [1] S. Joshi, C. Kim, S. Ha, and G. Cauwenberghs, "From algorithms to devices: Enabling machine learning through ultra-low-power VLSI mixed-signal array processing," in *IEEE Custom Integrated Circuits Conference (CICC)*, Apr. 2017, pp. 1–9.
- [2] M. Püschel, J. M. Moura, J. R. Johnson, D. Padua, M. M. Veloso, B. W. Singer, J. Xiong, F. Franchetti, A. Gacic, Y. Voronenko *et al.*, "SPIRAL: code generation for DSP transforms," *Proceedings of the IEEE*, vol. 93, no. 2, pp. 232–275, Jun. 2005.
- [3] S. Joshi, C. Kim, S. Ha, Y. M. Chi, and G. Cauwenberghs, "21.7 2pJ/MAC 14b 8×8 Linear Transform Mixed-Signal Spatial Filter in 65nm CMOS with 84dB Interference Suppression," in *IEEE International Solid-State Circuits Conference (ISSCC)*, Feb. 2017, pp. 364–365.
- [4] S. H. Mirfarshbafan and C. Studer, "Sparse Beamspace Equalization for Massive MU-MIMO mmWave Systems," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2020, pp. 1773–1777.
- [5] Y.-W. Huang, B.-Y. Hsieh, T.-C. Chen, and L.-G. Chen, "Analysis, fast algorithm, and VLSI architecture design for h. 264/AVC intra frame coder," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 3, pp. 378–401, Feb. 2005.
- [6] C. Kim, S. Joshi, C. M. Thomas, S. Ha, L. E. Larson, and G. Cauwenberghs, "A 1.3mW 48MHz 4 channel MIMO baseband receiver with 65dB harmonic rejection and 48.5dB spatial signal separation," *IEEE Journal of Solid-State Circuits*, vol. 51, no. 4, pp. 832–844, 2016.
- [7] B. Murmann *et al.*, "ADC performance survey 1997–2020," *Online*: <http://www.stanford.edu/murmann/adcsurvey.html>, 2020.
- [8] H. Omran, H. Alahmadi, and K. N. Salama, "Matching Properties of Femtofarad and Sub-Femtofarad MOM Capacitors," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, no. 6, pp. 763–772, Apr. 2016.