

Low Latency Transpose Memory for High Throughput Signal Processing

⁺*Mohamed El-Hadedy, ^{*} Sohan Purohit, ^{*}Martin Margala, ⁺Svein J. Knapskog

⁺ *Norwegian University of Science and Technology, Trondheim, Norway*

^{*} *University of Massachusetts Lowell, USA*

{hadedy, Svein.J.Knapskog}@q2s.ntnu.no, {Sohan_Purohit, Martin_Margala}@uml.edu

Abstract

This paper presents the design and analysis of a power and area efficient, low latency transpose memory structure for use in adaptive signal processing systems. The proposed architecture achieves significant improvements in system throughput over competing designs. We demonstrate the throughput performance of the proposed memory on FPGA as well as ASIC implementations. The memory was employed in a watermarking architecture previously proposed. The new memory design allows for 2X speed up in performance for the watermarking algorithm and up to 10X speedup for 2D DCT and IDCT algorithms compared to previously published work, while consuming significantly lower power and area.

1. Introduction

Recent advances in multimedia technology have prompted a lot of research in the design and development of signal processing architectures. With the constant evolution of new, more efficient signal processing algorithms, it has become imperative for modern research groups to be able to deliver efficient high throughput architectures for the same. As a result, several architectures have been proposed for high throughput signal processing applications.

Media processing applications are often considered as extremely data intensive applications. This means that the algorithm being implemented requires repetitive operations to be performed on the same data set. As a result the role of memory has been traditionally seen only as scratch pad memory for data and instruction memory for configuring the computational units. However it is important to note that signal processing algorithms such as DCT, DWT, as well as techniques used for digital watermarking which may include encryption, e.g. with powerful algorithms such as AES, require several manipulations to matrix based data. For instance, the 2D DCT requires the first stage computed DCT to be transposed before the second DCT is applied. This implies that the data has to be manipulated within the memory itself without using the processing elements. From an architectural point of view such operations often translates into valuable clock cycles for the memory based operations as well as increased power consumption due to

the large number of memory accesses involved, as well as potential read/write contention issues.

As a result, the role of transpose memory in these architectures becomes extremely important. Previously, several architectures for DSP using transpose memory arrays have been proposed. However, in most cases the memory structure is tailored specifically for the particular architecture, thereby compromising its adaptability. In this paper we present a novel architecture for a transpose memory subsystem. The design is extremely simple, area and power efficient and provides significant throughput improvements over previous work. To demonstrate the increased performance achieved by using this memory, results of 2D DCT and IDCT implementations using the proposed memory structure are presented. The memory structure is evaluated in both a FPGA as well as an ASIC implementation using the IBM 90nm CMOS process.

The rest of the paper is organized as follows. Section 2 highlights the architecture as well as the VLSI and the FPGA implementations of the proposed memory subsystem. Implementations of popular DSP benchmark designs using the proposed memory system are discussed in Section 3 along with a comparison with other architectures of their performance results. Finally some concluding remarks are made.

2. Proposed Transpose Memory Sub-system

The architectural setup of the proposed transpose memory subsystem is shown in Fig. 1. It consists of an 8X8 array of 12 bit cell modules. As shown in Fig.2, the array is configured to receive inputs from the horizontal as well as vertical direction as well as to shift data in both horizontal as well as vertical directions. One of the most important aspects of the proposed memory design is the ability to shift data in either direction as well as to switch the direction of data flow during the circuit operation. This is achieved through cell design as shown in Fig. 3. Each cell consists of a 12 bit register, 12 bit 2:1 multiplexer at the input and a 12 bit 2:1 de-multiplexer at the output. The multiplexer and de-multiplexer allow the register file to accept data from either direction as well as to transmit data in either horizontal or vertical direction. All the switching circuits, i.e at the input as well as at the output, are controlled through a common control signal. The signal generated by the control unit switches the select lines of the multiplexer and de-multiplexer to switch the

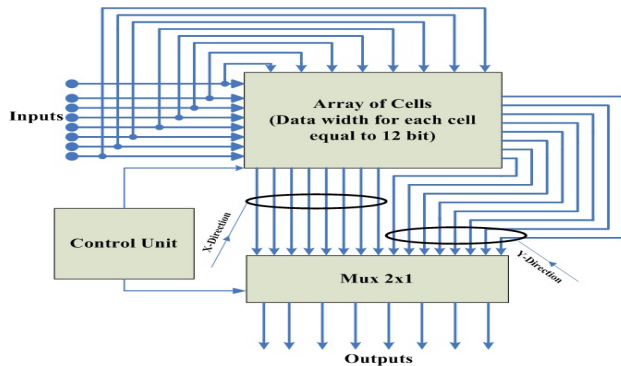


Figure 1: Top level architecture of the transpose memory block

direction of data flow after every eight cycles. This implies that the array can now accept an 8X8 data block in the horizontal direction, and start transmitting the transpose through the vertical direction. It should be noted that at the same time as the memory outputs the transpose

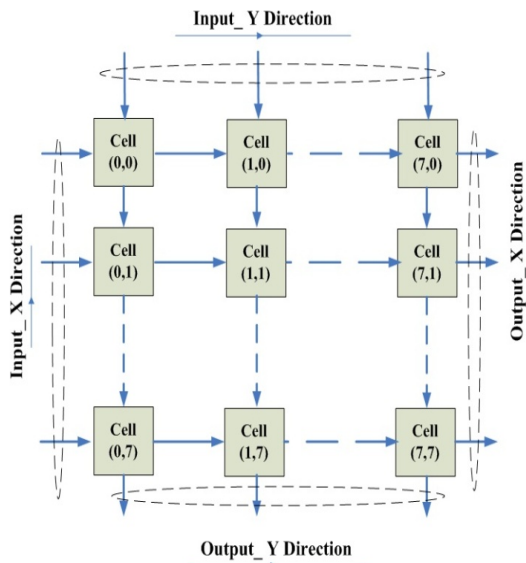


Figure 2: Internal organization of cells array

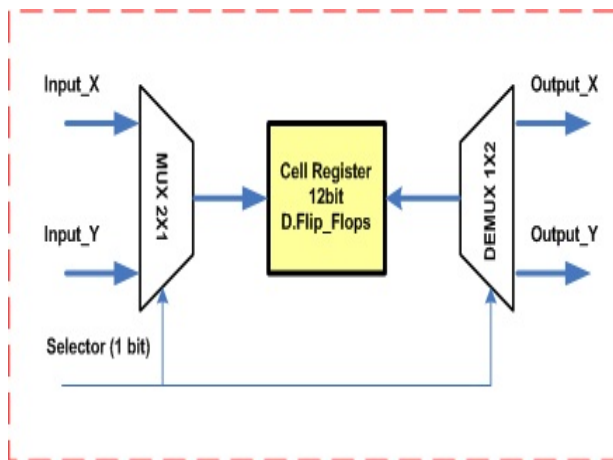


Figure 3: Internal view of Cell

in the vertical direction, it can continue accepting new data through the vertical input. Compared to other designs which require the memory to be completely cleared before a new block can be accepted, the proposed arrangement described here saves 8 clock cycles per 8X8 DCT block to be processed. This allows massive throughput enhancements by just modifying the transpose memory block in DSP architecture. This will be demonstrated in the next section.

The proposed transpose memory scheme was implemented in ASIC as well as on the Virtex XCV800 FPGA from Xilinx to evaluate its performance and throughput in signal processing applications. The ASIC implementation was performed in IBM's 90nm CMOS technology. The register file uses 12 TSPC flip-flops in each memory cell. TSPC was chosen due to its robust behavior and excellent timing characteristics. The multiplexer and de-multiplexer were implemented using a transmission gate to save area as well as power in the ASIC implementation. It should be noted that to reduce the complexity of the architecture, a single control signal is extracted from the control unit to switch the flow of data between horizontal and vertical directions. This applies to the input as well as output direction of the data. However this places an extremely large amount of load on the signal and hence careful buffering is necessary. Moreover, as each register unit includes 12 D flip-flops it is necessary to carefully design the clock tree as well. Furthermore, the memory unit is designed to switch between horizontal and vertical dataflow after every eight sets of data which corresponds to 8 clock cycles. As a result it is necessary for the signal controlling the data flow direction always to be in complete synchronization with the clock signal. For this purpose the clock and control signals are buffered identically so as to ensure valid data being latched into the flip-flops at each stage and preventing skew induced timing violations in the circuit. Fig. 4 shows the layout of the proposed transpose memory implemented in IBM 90 nm technology. The entire design occupies 22800 μm^2 and consumes a worst

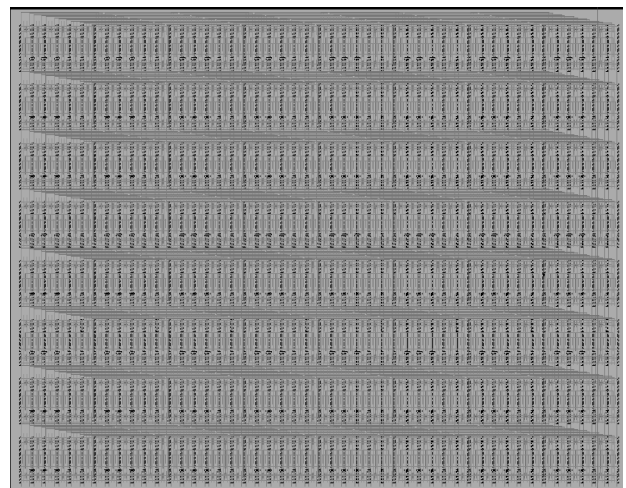


Figure 4: Layout of transpose memory module in IBM 90 nm process

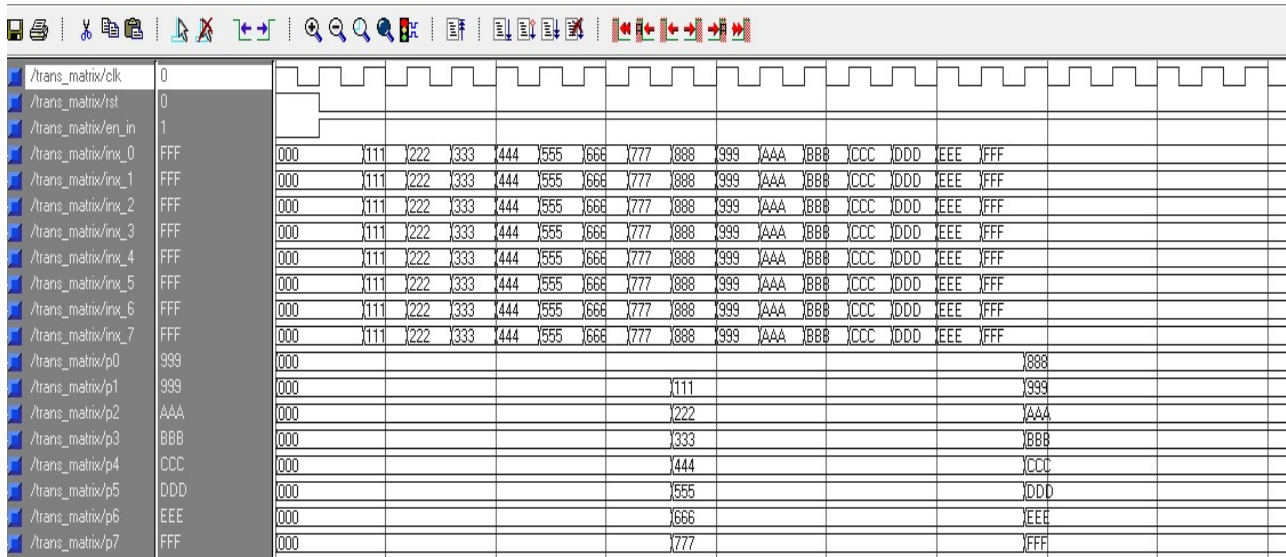


Figure 5: Simulation waveforms demonstrating functionality of the proposed memory sub-system

case power of 2.15 mW at 0.8 GHz and 1 V operating voltage.

The transpose memory architecture proposed here is intended to increase throughput in signal processing architectures while consuming minimum power and area. Hence, to verify the advantages of the new memory sub-system, we implemented several signal processing algorithms utilizing the new memory unit. The following section describes in detail the applications of the new transpose memory system.

3. Implementation of Signal Processing Algorithms

The functionality of the proposed transpose memory was verified on the Xilinx Virtex XCV800 FPGA. Fig. 5 shows the simulation waveforms for the memory system. It can be observed that the memory unit initially accepts the data inputs 111 through 888 as column inputs from Inx_0, Inx_1, ..., Inx_7 ports and after 8 clock cycles transmits the transposed output in the form of rows from P0, P1, P2, ... and P7 ports. The unit occupies 461 slices on the Xilinx Virtex FPGA while supporting a maximum frequency of 160 MHz.

To observe the improvement in throughput using the proposed memory, we implemented 8x8 2D DCT [1], 8X8 2D IDCT [1] and the encoder for Modified Mid-band Exchange Coefficient (MMBEC) [2] for digital watermarking. The primary objective was to observe the improvement in throughput and performance, as well as the reduction in system area. We observed a tremendous increase in throughput compared to previously published results. The proposed memory subsystem also allows a latency time of just 8 clock cycles for DCT and IDCT compared to 22 cycles in [3] and 10 cycles in [4] while

supporting slightly higher operating frequencies as well. The ASIC implementation too will provide peak throughput almost twice that observed in the earlier implementation.

Table 1 shows throughput improvement for 2D DCT, IDCT implementations for different block sizes, compared with other techniques described in relevant literature. Table 2 shows throughput for execution MMBEC watermarking algorithm using the new transpose memory architecture. It can be observed that for a 2D DCT and IDCT implementation, using the proposed transpose memory sub-system allows for almost 10 times increase in throughput over previous work. For the MMBEC algorithm, the new transpose memory allows almost 50% reductions in total execution time thus resulting in tremendous increase in the system throughput performance. On the Virtex XCV800 it was observed that the maximum supported frequency increased, while the area on FPGA occupied by the implementations is reduced considerably. This validates the advantages of the proposed memory scheme. It was observed that during normal operation, the memory scheme enables us to save up to 8 clock cycles for each 8X8 block of coefficient in DCT and IDCT implementations, thus resulting in a massive improvement in throughput.

4. Conclusion

This paper presented the ASIC and FPGA implementations of a new transpose memory structure to be used in high throughput signal processing architectures. Simulation results show that the proposed memory allows a saving of 8 clock cycles per 8X8 block of DCT coefficients, compared to traditional implementations proposed earlier by others. The ASIC implementation was shown to be capable of extremely

Table 1: Performance results for 2D DCT and IDCT using proposed transpose memory

Algorithm	Resolution	Execution Time (u sec)			
		Proposed	[2]	[3]	[4]
DCT	8X8	0.38	0.84	01.43	2.21
	16X16	0.95	2.10	04.63	8.39
	32X32	3.24	7.16	17.43	33.11
	64X64	12.38	27.37	68.63	131.97
	128X128	48.95	108.21	273.43	527.43
	256X256	195.24	431.58	1092.63	2109.28
	512X512	780.38	1725.05	4369.43	8436.69
	1024X1024	3120.95	6898.95	17476.63	33746.3
IDCT	8X8	1.6	0.86	-----	2.99
	16X16	2.2	2.16	-----	11.5
	32X32	4.6	7.35	-----	45.53
	64X64	14.2	28.1	-----	181.65
	128X128	52.2	111.13	-----	726.12
	256X256	206.2	443.24	-----	2904.03
	512X512	820.6	1771.67	-----	11615.68
	1024X1024	3278.2	7085.4	-----	46462.24

Table 2: Performance improvement in MMBEc implementation using proposed transpose memory

Algorithm	Execution Time (u sec)		
	Resolution	Proposed	[2]
MMBEC	8X8	1.59	.97
	16X16	4.28	2.43
	32X32	15.05	8.28
	64X64	58.13	31.66
	128X128	230.44	125.2
	256X256	919.67	499.35
	512X512	3676.59	1995.97

high resource efficiency with a remarkable throughput improvement of up to 10 times over competing DCT and IDCT implementations.

An important observation made during the design of this system is that it is easily integrable as an adaptive memory unit into modern coarse grained reconfigurable architectures and upon modification can serve the dual role of a scratchpad register file as well as transpose memory during normal and DSP applications respectively. Our future work in this area will involve integrating this memory as an embedded unit in adaptable architectures for signal processing applications.

5. References

[1] K.Z Bukhari, G.K Kuzmanov, S. Vassiliadis, "DCT and IDCT Implementations on Different FPGA Technologies", *Proceedings of ProRISC 2002*, pp-232-235, Netherlands, 2002.

[2] M. E. Elhadedy, A. H. Madian, H. I. Saleh, M. A. Ashour " Hardware implementation of the Encoder Modified Mid-band exchange coefficient technique (MMBEC) based on FPGA", , Proceedings of 19th International Conference on Microelectronics ICM2007, pp-43-46, Cairo, Egypt Dec.2007

[3]S. Hsia, C, Tsai, S Wang, K. Hung," Transposed Memory Free Implementation for Cost-Effective 2D DCT Processor", *Journal of Signal Processing Systems* ,vol 58,pp161-172, 2010.

[4] J. Huang, J. Lee," Efficient VLSI Architecture for Video Transcoding", *IEEE Transactions on Consumer Electronics*, vol.55, no.3, pp-1462-1470, August 2009