

PARALLELIZATION OF FAST HAAR TRANSFORM ALGORITHMS ON DUAL-CORE SPECIALIZED PROCESSORS

Ibragimov Sanjarbek Salijanovich

Andijan State Technical Institute, associate professor, PhD

e-mail: sanjari07@yahoo.com

Annotation: This article addresses the issue of parallelizing digital signal processing methods based on the Haar transform using Andrews' algorithm on dual-core specialized processors. The study develops mechanisms for parallel execution of the fast Haar transform algorithm by effectively utilizing the hierarchical memory model, DMA controller, and core architecture of the Blackfin ADSP-BF561 processor. Algorithmic experiments were conducted in C++ using the VisualDSP++ platform, based on block-polynomial basis functions (constant, linear, quadratic). Practical tests compared performance between the single-core ADSP-BF533 and dual-core ADSP-BF561 processors, achieving speed-up coefficients ranging from 1.14 to 1.80. The results demonstrate that multi-core computing significantly improves signal processing efficiency and reduces processing time.

Keywords: Fast Haar Transform (FHT), Andrews Algorithm, Parallel Computation, Dual-Core Processor, Blackfin ADSP-BF561, Digital Signal Processing (DSP), Piecewise Polynomial Basis Functions, VisualDSP++ Development Environment, Acceleration Coefficient.

INTRODUCTION

The Blackfin ADSP-BF561 processors are symmetric dual-core specialized processors [2,8,9]. The ADSP-BF561 employs a hierarchical three-level memory architecture. The first-level (L1) memory operates at the core clock frequency, although its capacity is relatively limited. Each Blackfin core is equipped with 100 KB of L1 memory, which includes the following components [4]:

- 16 KB instruction memory (SRAM/cache)
- 16 KB instruction memory (SRAM)
- 32 KB data memory (SRAM/cache)
- 32 KB data memory (SRAM)
- 4 KB scratchpad memory (fast SRAM)

The second-level (L2) memory consists of a 128 KB SRAM that is integrated with the core. The L2 memory is shared between both cores and can store common instructions and data. To optimize data exchange between L1 and L2 memory, the processor architecture includes a dedicated four-channel internal Direct Memory Access (DMA) controller.

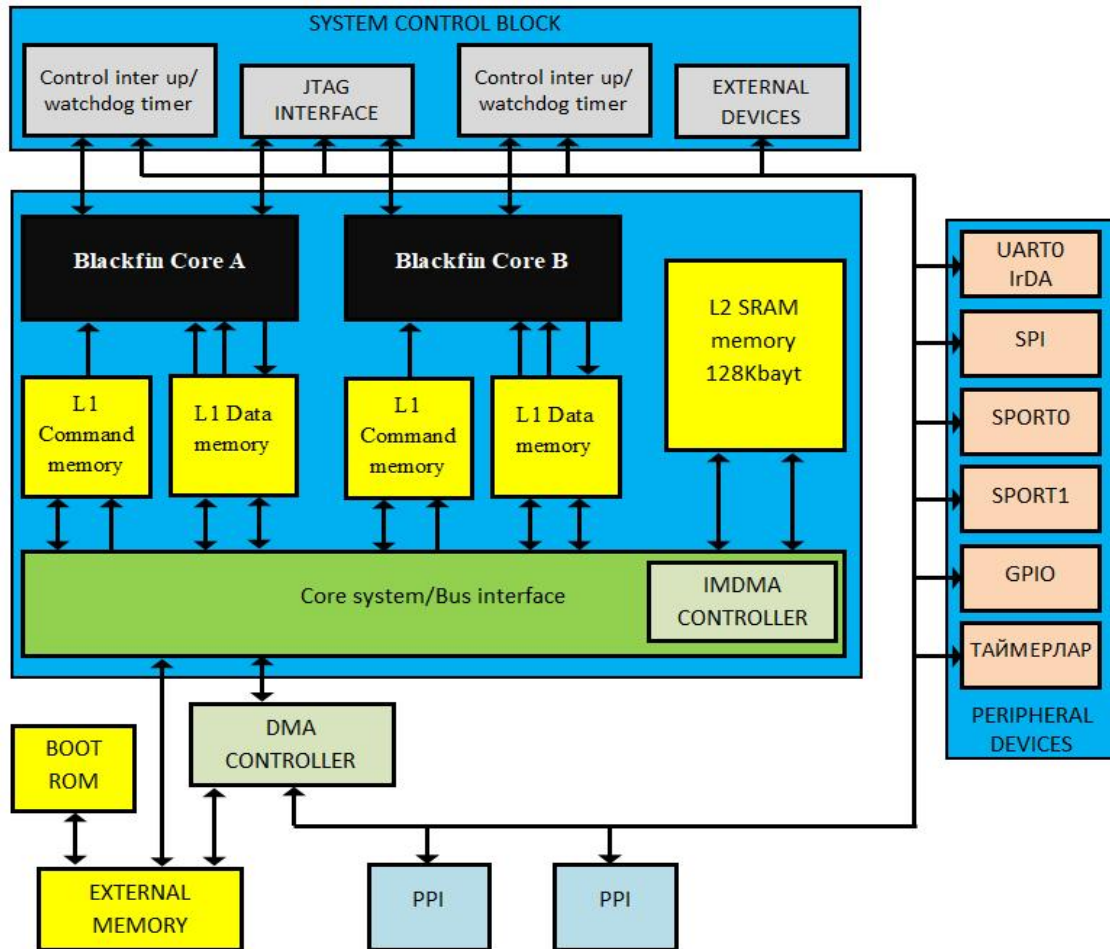


Figure 1. Functional Block Diagram of the ADSP-BF561 Processor

In the hierarchical memory architecture of the Blackfin processor, the third-level (L3) memory refers to external memory. The external memory space can map up to four SDRAM banks ranging from 16 MB to 512 MB, as well as four asynchronous memory banks (such as ROM, SRAM, EEPROM, or Flash), each with a capacity of up to 64 MB [1,6].

The Blackfin processor instruction set is optimized such that the most frequently used instructions are encoded using 16-bit opcodes. Digital Signal Processing (DSP) instructions are implemented as multifunctional operations encoded in 32-bit format. The internal bus architecture and computational units are designed to enable each processor core to execute multiple instructions per cycle, thereby enhancing code density.

In dual-core Blackfin ADSP-BF561 processors, parallel computing algorithms are utilized for solving practical tasks. Computational processes are distributed separately across each core [2,6-9].

RESEARCH METHODOLOGY

A parallel computation algorithm based on the Fast Haar Transform (FHT) was developed for digital processing of one-dimensional geophysical signals on dual-core specialized processors of the Blackfin ADSP-BF561 architecture.

According to the Fast Haar Transform method proposed by Andrews, in each iteration, the array X is formed by computing the sums of consecutive pairs from the previous X array, while the array C contains the differences of the same consecutive pairs.

$$X(i)=X(2i)+X(2i+1) \quad (1)$$

$$C\left(i+\frac{N}{2}\right)=X(2i)-X(2i+1).$$

The sum of the X values computed in the final iteration is equal to the following expression:

$$C(0)=X(0)+X(1) \quad (2)$$

Here, $i=0\dots N/2-1$, $j=1\dots k$. The number of iterations is determined by the formula $k=\log_2 N$.

The algorithm of the parallel computation process of the Fast Haar Transform based on Andrews' method on a dual-core specialized processor is presented in Figure 2.

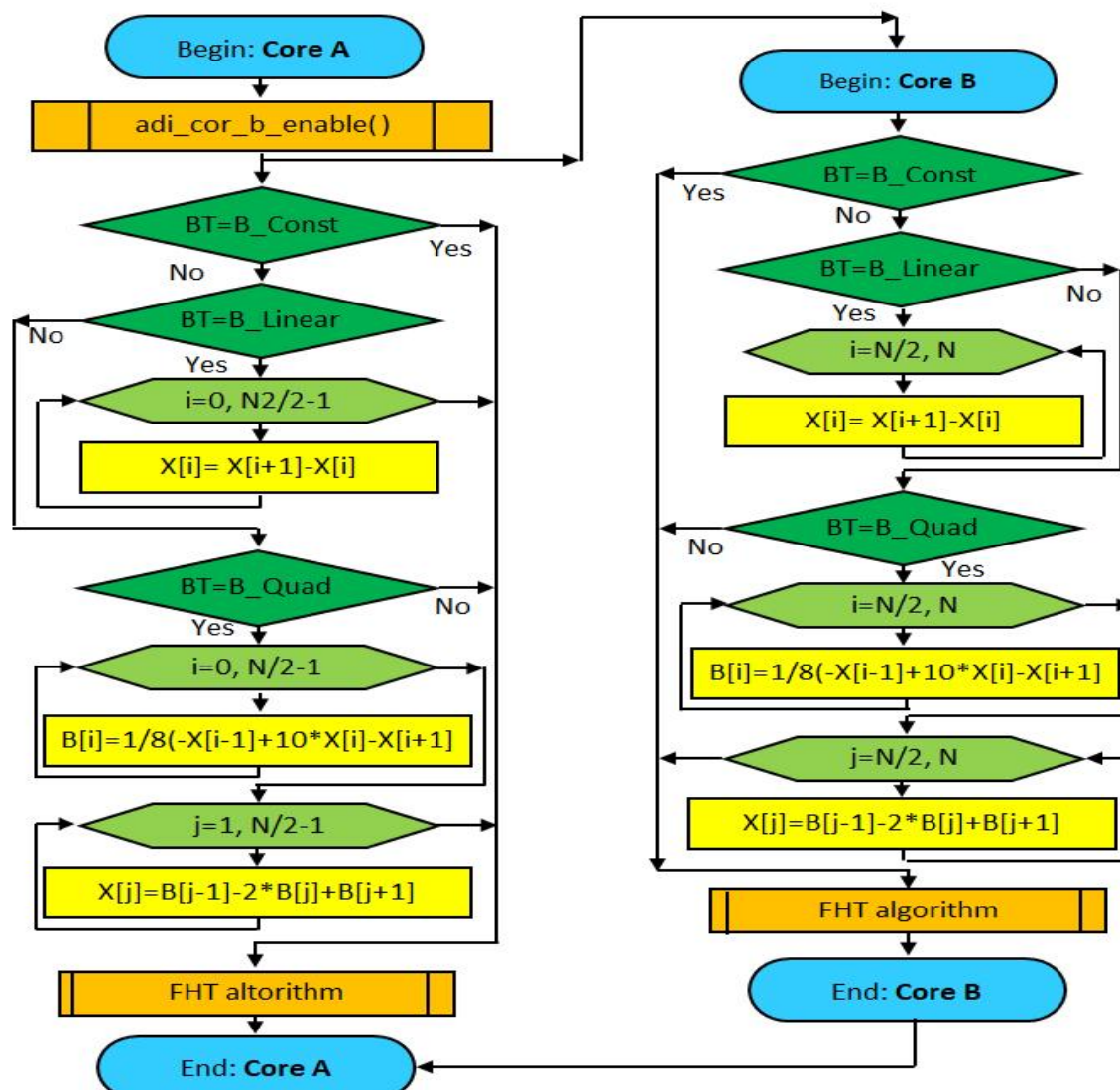


Figure 2 Parallel Computation Algorithm for Andrews-Based Fast Haar Transform on the Dual-Core Processor Blackfin ADSP-BF561

Based on this algorithm, the processes of inputting and outputting array values are executed sequentially on Core A. The computation of the Fast Haar Transform (FHT) is evenly distributed between Core A and Core B. Specifically, for each iteration, the first half of the array elements is processed by Core A, while the second half is processed by Core B, enabling parallel execution. Initially, the program code on Core A is launched, and Core B is activated using the `adi_core_b_enable()` function.

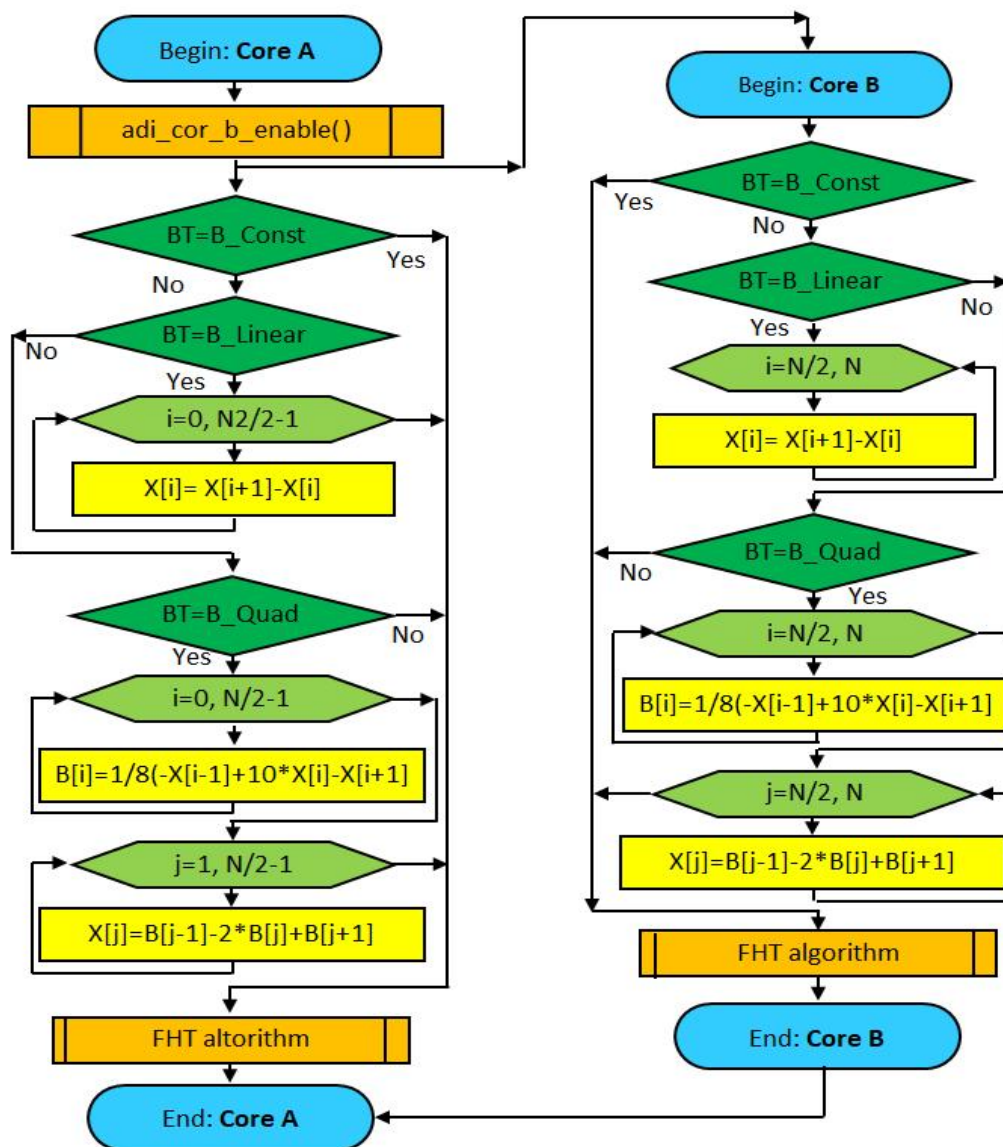


Figure 3. Block Diagram of the Parallel Computation Algorithm for Piecewise-Polynomial Bases on a Dual-Core Specialized Processor

The parallel computation algorithm of Fast Haar Transforms (FHT) using piecewise-polynomial bases on dual-core processors is presented in Figure 3.

RESULTS

Fast Haar Transforms based on Andrews' method were implemented using piecewise-polynomial bases on ADSP BF-533 (single-core) and ADSP BF-561 (dual-core) specialized processors from the Blackfin processor family, utilizing the C++ programming language within the VisualDSP++ development environment. The processors selected for the computations had the same clock frequency: the ADSP BF-533 core operated at 600 MHz, and each core of the ADSP BF-561 also operated at 600 MHz.

To evaluate the computation time of Fast Haar Transforms based on piecewise-polynomial bases on these processors, experiments were conducted using the analytical function $y=e^x$ as input data [3,7]. The results of the study are presented in Table 1.

Table 1.

Comparison of Computation Times for Fast Haar Transforms Based on Piecewise-Polynomial Bases on Single-Core and Dual-Core Processors

N	Piecewise Constant			Piecewise Linear			Piecewise Quadratic		
	ADSP BF-533 (sec.)	ADSP BF-561 (sec.)	Acceleration Coefficient	ADSP BF-533 (sec.)	ADSP BF-561 (sec.)	Acceleration Coefficient	ADSP BF-533 (sec.)	ADSP BF-561 (sec.)	Acceleration Coefficient
16	0,00110	0,00096	1,14	0,00110	0,00100	1,10	0,00112	0,00097	1,15
32	0,00385	0,00297	1,30	0,00382	0,00295	1,30	0,00386	0,00278	1,39
64	0,01429	0,00950	1,50	0,01423	0,00994	1,43	0,01432	0,00914	1,57
128	0,05498	0,03414	1,61	0,05471	0,03567	1,53	0,05502	0,03322	1,66
256	0,21560	0,12723	1,69	0,21426	0,13206	1,62	0,21544	0,12534	1,72
512	0,85368	0,49602	1,72	0,84398	0,50379	1,68	0,85341	0,48413	1,76
1024	3,38911	1,94438	1,74	3,35060	1,93456	1,73	3,39144	1,90746	1,78
2048	13,45478	7,66087	1,76	13,30188	7,52543	1,77	13,49794	7,53445	1,79
4096	53,41548	30,26043	1,77	52,80848	29,72544	1,78	53,85680	29,89671	1,80

In this table:

N – number of array elements;

The Acceleration Coefficient [3,7,9] is calculated using the following formula:

$$\text{Acceleration Coefficient} = \frac{T_{\text{single-core}}}{T_{\text{dual-core}}}$$

CONCLUSION

As seen from Table 1, the dual-core processor requires approximately 1.77 to 1.8 times less computation time compared to the single-core processor, resulting in acceleration coefficient of 1.8. This indicates that parallelization of computations using multi-core processors significantly reduces processing time and improves overall efficiency compared to single-core execution.

One of the main advantages of developing parallel programs for dual-core processors on the VisualDSP++ platform is the ability to create separate lightweight program modules for each core and the shared memory. This enables both cores to perform different computations in parallel simultaneously. Additionally, in the shared L2 memory design, both cores can access common data and functions concurrently, further enhancing the potential for parallel processing.

REFERENCES:

1. Андрейченко Д.К., Велиев В.М., Ерофтиев А.А., Портенко М.С. Теоретические основы параллельного программирования. Саратов 2015 – 282с.
2. Вальпа О.Д. Разработка устройств на основе цифровых сигнальных процессоров фирмы Analog Devices с использованием Visual DSP++. – М.: Горячая линия-Телеком, 2007. – 270 с
3. Воеводин, В.В. Параллельные вычисления. СПб.: БХВ-Петербург. 2002. - 608 с.
4. Гергель В.П. Высокопроизводительные вычисления для многоядерных многопроцессорных систем. Учебное пособие – Нижний Новгород; Изд-во ННГУ им. Н.И.Лобачевского, 2010 – 420с.
5. Зайнидинов Х.Н, Усмонов Б.Ш. Архитектура компьютеров и компьютерных систем. // Ташкентский университет информационных технологий им. Мухаммада Ал-Хоразмий. –Т., 2020. -640с. ISBN 987-9943-5805-5-8
6. ADSP-BF561 Blackfin® Processor Hardware Reference, Analog Devices, Inc. 2013.
7. Deergha Rao K., Swamy M.N.S. Digital Signal Processing Theory and Practice // Springer Nature Singapore Pte Ltd. 2018 – 799 p.
8. Zaynidinov H.N., Ibragimov S.S., Tojiboyev G'.O. Comparative Analysis of the Architecture of Dual-Core Blackfin Digital Signal Processors. International Conference On Information Science And Communications Technologies: Applications, Trends And Opportunities <http://www.icisct2021.org/> ICISCT 2021, November 3-5, 2021. <https://ieeexplore.ieee.org/document/9670135> (SCOPUS). p.1-5
9. Zaynidinov H.N., Ibragimov S.S., Tojiboyev G'.O., Nurmurodov J.N. Efficiency of Parallelization of Haar Fast Transform Algorithm in Dual-Core Digital Signal Processors. 2021 8th International Conference on Computer and Communication Engineering (ICCCE). 22-23 June 2021, Kuala Lumpur, Malaysia (SCOPUS). p. 7-12