# A high performance processor architecture for multimedia applications☆

Shafqat Khan [a], Muhammad Rashid [b,*], Faraz Javaid [c]

[a] *COMSATS Institute of Information Technology Abbottabad, Pakistan*
[b] *Umm Al-Qura University (UQU), Makkah, Saudi Arabia*
[c] *Center for Advance Studies in Engineering (CASE), Islamabad, Pakistan*

A B S T R A C T

In this paper, an efficient sub-word parallelism (SWP)-enabled Reduced instruction-set Computer (RISC) architecture is proposed. The proposed architecture can perform efficiently for both conventional and multimedia-oriented applications. Speed-up for multimedia applications is achieved by adding the customized SWP instructions in RISC processor core. Rather than operating on a single data, customized instructions perform parallel computations on multiple pixels, packed in word-size registers. The sub-word-sizes in SWP instructions are selected, based upon the pixel sizes (8, 10, 12, 16-bit) in modern multimedia applications. The *SWP-RISC* processor is designed and implemented on two different CMOS technology nodes (90 nm and 45 nm). The performance of processor is characterized for different multimedia applications and compared with the state-of-the-art TMS320C64X processor.

© 2017 Elsevier Ltd. All rights reserved.

## 1. Introduction

Nowadays, modern electronic devices like smart phones, digital cameras, automobiles etc. have to perform a variety of tasks. For the scheduling, controlling and smooth functionality of multiple tasks, a processor core is required in electronic devices. Most of the embedded processor designs are based upon Reduced instruction-set Computer (RISC) architecture because of its simplicity and optimization in instruction-set. The hardware architecture of RISC-based processor can be easily optimized for high speed applications using various performance enhancement techniques [1].

### 1.1. Related work

Several enhancements have been proposed in RISC architecture to increase the performance. For example, instructions pertaining to JPEG and H.264 encoding standards have been implemented and incorporated in the instruction-set of a RISC processor core [2,3]. Similarly, a RISC instruction-set architecture for a flexible Multiply-Accumulate (MAC) unit of a Very-Large-Instruction-Word (VLIW) Digital Signal Processor (DSP) core for 32, 16 and 8-bit data computations is presented in [4]. While Rashid et al. [2,3] and Huu et al. [4] advocate the use of instruction-set extension, a sub-word parallelism (SWP)

---

☆ Reviews processed and approved for publication by Editor-in-Chief.
* Corresponding author.
  *E-mail address:* mfelahi@uqu.edu.sa (M. Rashid).

technique has been proposed in some multimedia processors for the enhancement of performance at operation level [5,6]. Consequently, several SWP-based multimedia extensions have been implemented in both general purpose as well asDSP architectures. The typical examples of this trend are the inclusion of Multi Media Extension (MMX)and the introduction of streaming Single-Instruction-Multiple-Data (SIMD) extensions SSE, SSE2 and SSE3in IA-32 Intel architecture [7].

In addition to the Intel architecture, a SIMD-based general purpose datapath with efficient operation structure for motion estimation algorithm is presented in [8]. Similarly, a configurable and data-reusable instruction-set processor, exploiting the symmetry of search pattern to reduce the redundant data loading, is presented in [9]. However, most of these multimedia enhancements are based upon the conventional sub-word-sizes of 8, 16, 32-bit without considering the pixel sizes (8, 10, 12-bit) in most modern multimedia application [10,11]. As a result, the under utilization of processor resources occurs which ultimately reduces the performance. In [12], an extended RISC-V processor core, specifically designed for near threshold (NT) operation in tightly coupled multi-core cluster, has been proposed for ultra low power systems. Similarly, Hussain et al. [13] presents an integrated self-aware computing model, mitigating the power dissipation of a heterogeneous re-configurable multi-core architecture, by dynamically scaling the operating frequency of each core.

To summarize, in most of the customized processing cores, the main focus is to increase the performance by incorporating the instructions pertaining to the targeted applications without considering the characteristics of the data that needs to be processed. This lack of coordination between data and the underlying hardware architecture reduces the overall performance of the processor.

### 1.2. Our work

In this paper, a customized RISC processor core is proposed for multimedia applications. The performance of processor is increased, both at instruction level as well as at operation level. At instruction level, the customized instructions related to the operations required in modern multimedia applications, are incorporated in the RISC instruction-set. The customized multimedia instructions are implemented using synopsis processor design tool [14]. These instructions help to perform the multimedia-oriented operations like discrete cosine transform (DCT), motion estimation, discrete wavelet transform (DWT), image enhancement, pixel summations, sum of absolute difference etc. very efficiently.

At operation level, the performance of RISC processor is improved by using the sub-word parallelism (SWP) in the architecture of different arithmetic operators. In SWP, multiple pixels are packed in word-size registers and parallel computations are performed on the packed pixels. Consequently, the word-size resources are utilized to maximum extend even for the low precision pixel data. In this work, multimedia-oriented sub-word-sizes (8, 10, 12, 16-bits) are considered instead of classical sub-word-sizes (8, 16, 32-bits).

The implementation results of the proposed processor in terms of area, power and critical path prove the viability of our approach. Furthermore, the performance is compared with the recent TMS320C64x processor [15], having classical multimedia instruction-set. Due to efficient implementation and better coordination between pixel data and hardware architecture, the proposed processor in this paper provides higher speed-up for modern multimedia applications.

The rest of this paper is organized as follows: Section 2 sets the stage for the designing of *SWP-RISC* core, targeted at multimedia applications. Section 3 presents the architecture of customized multimedia instructions, added in the *SWP-RISC* core. Furthermore, the resources required for implementing each customized instruction are also analyzed. Section 4 describes the hardware implementation of *SWP-RISC* processor. Moreover, the chip area, timings and power consumption of the proposedprocessor are compared with the conventional RISC processor. Section 5 explains the performance of proposed processor when working on different multimedia applications. The performance is also compared with the state-of-the-art digital signal processor TMS320C64X, having conventional multimedia extension. Finally, the conclusion is presented in Section 6.

## 2. Setting the stage

This section starts with the brief overview of RISC architecture followed by the multimedia-oriented SWP technique which is used in the *SWP-RISC* architecture. Finally, the advantages of using ADL for processor designing are explained.

### 2.1. Overview of selected RISC architecture

RISC is a type of architecture that utilizes a small, highly-optimized set of instructions [1,4]. In this paper, SWP-enabled multimedia enhancements are proposed in five-stage RISC architecture. This RISC architecture is selected because of its simplicity and the hardware corresponding to multimedia instructions can easily be implemented on a RISC platform without major modifications. RISC under consideration in this article has 32 32-bit general purpose registers (GPRs) and few special-purpose registers (e.g. program counter and instruction register). In the selected RISC architecture, the range of program memory is kept limited to 32 Kilo word memory locations (0x0000 to 0x7FFF). Each memory location is 32-bits in size. The data memory stores the input data on which the program operates as well as the final results computed from the operations. Each memory location is 32-bits in length such that the data can be transferred from memory to GPR in one clock cycle. The defined range is kept limited to 32 Kilo word memory locations (0x8000 to 0xFFFF). Furthermore, the load-store