1R-08

Exploiting the FFT Acceleration for MFCC-Speech Recognition Using a RISC-V Microcontroller

Wu Xiaoting, Ckristian Duran and Cong-Kha Pham
University of Electro-Communications (UEC), Tokyo, Japan
Email:{xiaoting, duran}@vlsilab.ee.uec.ac.jp, phamck@uec.ac.jp

***Abstract*—Mel-Frequency Cepstral Coefficients (MFCC) is a technique used to obtain a signal's power spectrum for recognition applications. The Mel-coefficients are used in the different speech recognition methods, obtaining a high recognition rate. However, the most processing time is used for the Fast Fourier Transform (FFT). This work presents an FFT accelerator implemented in Field-Programmable Gate Array (FPGA), using a RISC-V based microcontroller. The FFT implementation increases the performance by 20.7% of the Mel-coefficients extraction compared to the software implementation. The implementation occupies a 1987 Look-Up Tables (ALUT) and 244 Flip-Flops (FF), and 16384 Memory Bits in 256 bits configuration, representing a 77% smaller than the implemented RISC-V processor.**

***Index Terms*—RISC-V, SoC, MFCC, FFT, Speak Recognition.**

I. Introduction

Automatic Speech Recognition (ASR) is used in many applications for interacting with machines, improving comfort and performance in society [1]. The Mel-Frequency Cepstral Coefficients (MFCC) is one of the most speech feature extraction for ASR applications. However, the Fast Fourier Transform (FFT) and the Discrete Cosine Transform (DCT) are the most expensive steps in the process to obtain the Mel-coefficients [2]. In this work, we present a FFT accelerator implemented in a RISC-V environment. The processing time of the FFT is reduced by 160× in comparison to the software implementation in the RISC-V environment. The overall extraction pre-processing of the feature extraction decrease a 20.7%, using the FFT accelerator over all the samples. The FFT accelerator implementation occupies a 2182-LUT and 244-FF, and 65536 Memory Bits in 1024 bits flavor, representing a 16% smaller in comparison with the RISC-V core. Using the 256 configuration, the resources are 1987-LUT and 244-FF, and 16384 Memory Bits, representing a 77% smaller in comparison with the RISC-V core.

II. MFCC Flow Extraction

The MFCC is used to extract the power spectrum based on the DCT. The power spectrum is used in the ASR system. Fig. 1 shows the MFCC flow to extract the Mel-coefficients. The *Signal Acquisition* highlighted in red sampling the data with a determinate sample frequency. In another way, the *Digital Signal Processing* highlighted in green processes the samples to obtain the Mel-coefficients. First, a *Pre-Emphasis* stage frames the data and prevents spectral leakage. The *FFT* converts the data into the frequency domain. In addition, the *Mel Filter Bank* is included to extract enough energy information in the low-frequency bands, to match the behavior of the human ear [3]. Finally, the *DCT* obtain the Mel-coefficients, transform them back to the time domain.

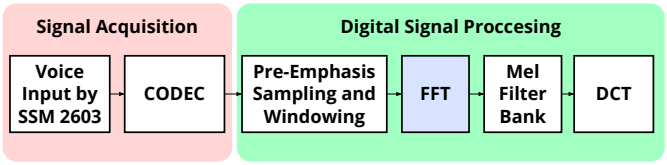

Fig. 1. MFCC flow implemented.

III. Hardware Implementation

A. System-on-Chip

The MFCC flow described in the section II is implemented using a RISC-V environment [4]. Fig. 2 illustrates the block diagram of the SoC implemented. The SoC is composed of a rocket core with 16-KB of instruction and data cache [5], a debug module, an SPI for external memory, a UART, a 256-MB of Dynamic Random-Access (DDR) memory, a system interruption, a coder/decoder (CODEC) and an FFT accelerator. In addition. the SoC uses a TileLink for system and peripheral buses [6].

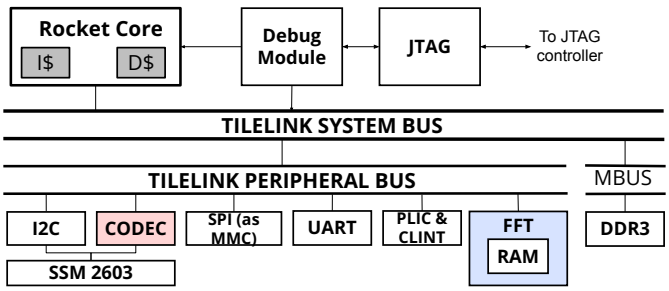

Fig. 2. Block diagram of the SoC implemented.

FFTアクセラレーションを活用したRISC -VベースのMFCC音声認識に関する研究
†ウ　ショウティン　電気通信大学大学院　情報理工学研究科
‡ドラン　クリスティアン　電気通信大学
‡範　公可　電気通信大学

The CODEC (SSM 2603) can be sample the data with 8, 22, 48, and 96-kHz, using the $I2C$ to configure the sampling frequency. Besides, the analog to digital converter inside the CODEC sends data sampled with 16-bit unsigned to the *Digital Signal Processing* stage.

B. FFT accelerator

Fig. 3 illustrates the architecture of the implemented FFT accelerator. This FFT uses one Butterfly Unit for two samples to perform the sum and multiply operations. The dual-port RAMs contain the real and imaginary components of the signal and its calculated spectrum. The data will be directed according to the scrambled address from the Address Generation Unit. This unit provides the logic for maintaining the order of the samples by performing calculations according to the current iteration and sample. A similar logic is used for the Twiddle Factor ROM, which provides the rotation factors for the Butterfly Unit according to the FFT algorithm. Finally, the memory routing logic will make the reading and writing interleaved between both memories to avoid overwriting.

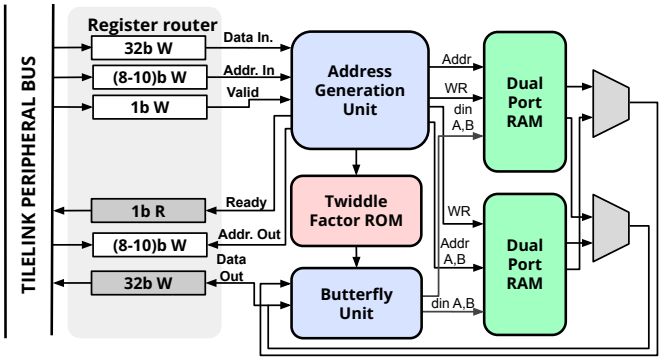

Fig. 3. Block diagram of the FFT accelerator implemented.

IV. Results

The microcontroller is implemented in ALTERA Cyclone V 5CSXFC6D6F31 FPGA. Table I shows the resources of the SoC implementation in FPGA. The microprocessor represents 19 % of all SoC. The DDR3 controller and the FFT accelerator represent the majority of the resources with 59%, and 16%, respectively. The FFT is configurable at synthesis time to three different lengths: 1024, 512, and 256 bits. The overhead of the RISC-V processor in comparison with accelerators are 16%, 57%, and 77%, respectively. The TileLink bus represents less than 1% of the SoC.

Table II shows the execution times of the implemented FFT. The application runs different lengths of sampled data, necessary for voice recognition. The software includes the preprocessing stages of filtering, windowing, FFT, and feature extraction using MFCC. The major overhead is presented in the MFCC, as the implemented processor only supports integer operations, and the feature requires floating-point operations. Other operations besides MFCC are implemented in software or hardware using fixed-point operations only. By implementing the hardware in FFT for 256 samples, the performance is 160 times over the software counterpart. The FFT increases the performance by 20.7% overall implementations for the overall extraction.

TABLE I
FPGA implementation results by module.

	ALUTs	FFs	MEM BITS
Rocket Core	7775	5034	68608
Debug Module	829	843	0
I2C	189	135	0
CODEC	151	193	16348
SPI	318	222	128
UART	155	140	128
ROM	120	180	0
DDR3	7838	7967	237084
TilelInk Bus	2643	1188	384
FFT 1024	2182	244	65536
FFT 512	2012	244	32768
FFT 256	1987	244	16384

TABLE II
Execution performance in clock cycles in FPGA.

	Execution Time @50 MHz @48 KSPS				
MCycles [ms]	Filter	Hann Wind.	FFT 256		Feature Extraction
			HW	SW	
32768 Samples	675.4 13.5	3127.2 62.5	326.1 6.5	54056.8 1081.1	212567.1 4251.3
65536 Samples	1363.4 25.2	6871.9 137.4	654.2 13.1	105750.6 2115.0	468336.4 9766.8
131072 Samples	2701.5 54.0	12267.4 245.3	1306.6 26.1	240768.4 4815.3	934005.8 18680.1

V. Conclusion

A Mel-Frequency Cepstral Coefficients process is optimized using an FFT hardware accelerator. The FFT implementation improves the performance 160× in comparison with the software implementation, using a RISC-V rocket core with integer operations. The implementation occupies 1987-LUT and 224-FF and 16384 Memory Bits with a 77% smaller resources of the RISC-V core, using the FFT 256 configuration. The overhead of the RISC-V processor in comparison with accelerators are 16%, 57% and 77%, for FFT 1024, 512, and 256 bits, respectively.

References

[1] I. López-Espejo *et al.*, "Deep Spoken Keyword Spotting: An Overview," *IEEE Access*, pp. 1–1, 2021.
[2] H. Kou *et al.*, "Optimized MFCC feature extraction on GPU," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 7130–7134.
[3] A. Jain *et al.*, "Evaluation of MFCC for speaker verification on various windows," in *International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014)*. IEEE, may 2014. [Online]. Available: https://doi.org/10.1109%2Ficraie.2014.6909144
[4] A. Waterman *et al.*, "The RISC-V Instruction Set Manual, Volume I: User-Level ISA, Version 2.0," EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2014-54, May 2014.
[5] RISC-V Foundation, "Rocket Chip Generator," 2019. [Online]. Available: https://github.com/chipsalliance/rocket-chip
[6] SiFive, Inc., "SiFive TileLink Specication," Aug. 2019. [Online]. Available: https://www.sifive.com/documentation/tilelink/tilelink-spec/