Designing an Application Specific Instruction Set Processor for FECs in LTE-Advanced

Hojun Kim, Dongjae Song, Soongyu Kwon, and Jong Tae Kim

***Abstract*—This paper presents a dynamically reconfigurable application-specific instruction set processor (ASIP) for the application domain of channel coding in LTE-Advanced. The ASIP consists of a specialized pipeline with 11 stages and a dedicated communication and calculations structure. Logic synthesis revealed a maximum clock frequency of 55 MHz and a total gate size is 413,295. Compared to the previous research result our ASIP saves about 20% of the area, but provides much more flexibility. Simulation results show that it meets the average data rate of LTE-Advanced.**

***Index Terms*—ASIC, LTE-Advanced, FEC**

I. Introduction

Long Term Evolution (LTE) Advanced which is introduced by 3GPP is one of the candidate for next generation mobile communication. LTE-Advanced is the next major step of LTE which based on OFDMA/SC-FDMA to fulfill the requirement of IMT-Advanced. One of the key technology to fulfill the spec of IMT-Advanced is channel coding. LTE-Advanced use the Viterbi coding and Turbo coding and Wibro-Evolution use Viterbi coding, Turbo coding and LDPC coding as the channel coding.

Next generation mobile communication system should support various communication standard to seamless and smooth interface across heterogeneous network. Therefore flexibility of modem is one of the key factors. To support several channel coding, the earlier studies have been focused on designing several FEC or multi-mode FEC using ASIC. However designing an ASIC in today's deep submicron geometries is harder than ever, and the problems continue to worsen with shrinking geometries. An alternative implementation style to ASICs that is rapidly emerging is the use of programmable platforms, or ASIPs. An ASIP is a component used in system-on-a-chip design. The instruction set of an ASIP is tailored to benefit a specific application. ASIP offers the availability of custom sections for time critical tasks and offer flexibility through an instruction-set. They can be finely tuned to run a small range of applications very efficiently, while keeping the ability to run other tasks through a micro-code program. In this paper, we present the processor which supports two FECs (Turbo decoding, Viterbi decoding), using the ASIP based on LTE-Advanced.

Manuscript received May 9, 2010

Hojun Kim and Soongyu Kwon are with the Department of Mobile Systems Engineering, Sungkyunkwan University, Suwon, Republic of Korea. Dongjae Song and Jong Tae Kim are with the School of Information and Communication Engineering, Sungkyunkwan University, Suwon, Republic of Korea (e-mail: jtkim@skku.edu).

II. ASIP Flow using LISATek

The LISA processor design platform is an environment that allows the automatic generation of software development tools for architecture exploration, hardware implementation, software development tools for application design and hardware-software co-simulation interfaces from one sole specification of the target architecture in the LISA language. The LISA language is aiming at the formalized description of programmable architecture, their peripherals and interfaces. It was developed to close the gap between purely structural oriented languages and instruction set languages for architecture exploration and implementation purposes of a wide range of modern programmable architecture.

ASIP flow using the LISATek is shown in Figure 1 [1]. Architecture design requires the designer to work in two fields. on the one hand, the development of the software part including C-compiler, assembler, linker and simulator and on the other hand the development of the target architecture itself. In Target Architecture part, designer decide the specification how to design the ASIP and what kind of application designer will apply. Designer should also decide the base processor, because LISATek offer the base architecture to designer to design easily. In LISA Description part, designer organize the architecture using the method which add or subtract instructions, and revise the pipeline stage. Since LISA language is strongly oriented towards the programming language C, it is easy and intuitive to design application specific instruction. After checking the grammar and architecture of the ASIP, build the compiler, assembler, linker and simulator using the C-Compiler. Then designer compile, assemble, link and simulate the application. Designer can see the profiling of application in ASIP. Designer repeat the above procedures if the result does not hit the target. If the result hit the target, designer generate the VHDL file of the ASIP, and the synthesize the VHDL and analyze the result again.

III. Specification of FECs in LTE-Advanced

A. Specification of Turbo Code

The Turbo encoder specified in the LTE specification uses parallel concatenated convolutional code (PCCC). The Turbo encoder is implemented with two 8-state constituent encoders and one Turbo code internal interleaver. The transfer function of the 8-state constituent code for PCCC is following, as in (1).

ISBN: 978-988-18210-0-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

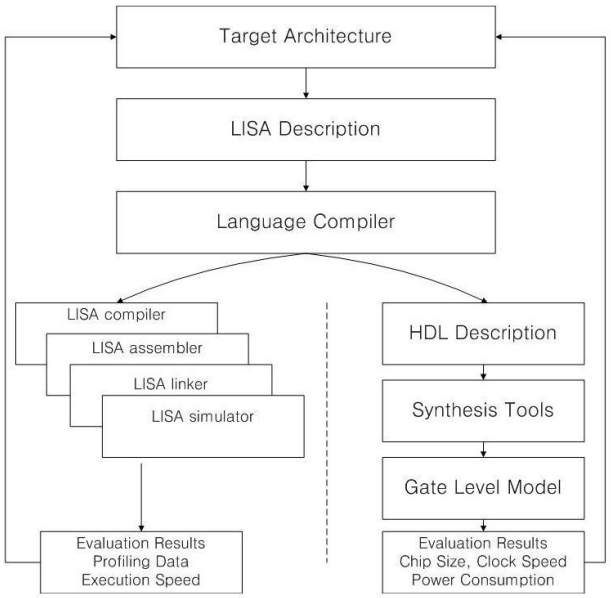

Fig. 1. ASIP flow using LISATek

$$G(D)=\left[1,\ \frac{g_1(D)}{g_0(D)}\right] \quad (1)$$

$$where \quad g_0(D)=1+D^2+D^3, \quad g_1(D)=1+D+D^3$$

The bits input to the Turbo code internal interleaver are denoted by $X_0, X_1, \cdots, X_{K-1}$ where K is the number of input bits. The bits output from the Turbo code internal interleaver are denoted by $X'_0, X'_1, \cdots, X'_{K-1}$. The relationship between the input and output bit is following as in

$$X'_i = X_{\pi(t)}, \quad i=0,1,\cdots,K-1 \quad (2)$$

where the relationship between the output index i and the input $\pi(i)$ index satisfies the following quadratic form like (3).

$$\pi(i)=(f_1 i+f_2 i^2) \bmod K \quad (3)$$

The parameters f_1 and f_2 depend on the block size K. Table 1 shows the interleaver parameters specified in the 3GPP Technical Specification.

B. Viterbi Code

Data encoded at a coding rate 1/2, 1/3 and 1/4 in convolutional encoder, and the constraint length is 5, 6, 7 and 9. Viterbi decoder follow the constraint length of the convolution encoder, the number of state is 16 , 32, 64 or 256.

C. Branch Metric Calculation Unit (BMC)

In order to provide enhanced flexibility for all different decoding configurations, the BMC has been divided into three major units: Mode selection unit, BM calculation unit, and BM distribution unit. These three units belong to 2 pipeline stages respectively. Mode select unit and BM calculation unit are a part of BMC1. We choose the FEC among convolution (K=5 to 9) and turbo code in Mode selection unit. BM calculation unit compute the branch metric.

BM distribution unit distribute the branch metric value to the each path according to the reference codeword in BMC2 stage. In BM distribution unit, we distribute the different number of branch metric values to each path according to the FEC, 32 values in K=5, 64 values in K=6, 128 values in K=7, 512 values in K=9 and 16 values in Turbo. Therefore, we divide the BM distribution unit into 4 parts. When Viterbi decoder with constraint length 5 and turbo decoder are selected, only BMC_K_5 is activated. When Viterbi decoder with constraint length 6 is selected, BMC_K_5 and BMC_K_6 are activated, and when Viterbi decoder with constraint length 7 is selected, BMC_K_5, BMC_K_6 and BMC_K_7 are activated. All of the BM distribution unit are operating when viterbi decoder with constraint length 9 is selected.

Table 1. Turbo Code Internal Interleaver Parameters

i	K_i	f_1	f_2
1	40	3	10
2	48	7	12
3	56	19	42
4	64	7	16
5	72	7	18
6	80	11	20
7	88	5	22
8	96	11	24
9	104	7	26
10	112	41	84

D. Add Compare Select Unit (ACS)

In order to provide enhanced flexibility for all different decoding configurations and reduce the power consumption, the ACS has been divided into four major units: Addition unit, Decision vector unit, Maximum unit, and Small state unit. Addition unit is a part of ACS1 stage, Decision vector unit and Maximum unit belong to ACS2 stage and Small state unit is in ACS3 stage. Addition unit add the branch metric value which comes from BMC block and path metric value which is stored in path metric memory. Using the added value, Decision vector unit compare two values which come into the each node and choose larger value. In Maximum unit, the maximum value is chosen among all of the path metrics. When we compute the maximum value in Maximum unit, if we use the result of Decision vector unit, we just compare 16 values instead of 32 values in Viterbi decoder with constraint length 5, 32 values instead of 64 values in Viterbi decoder with constraint length 6, 64 values instead of 128 values in Viterbi decoder with constraint length 6 and 256 values instead of 512 values in Viterbi decoder with constraint length

ISBN: 978-988-18210-0-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

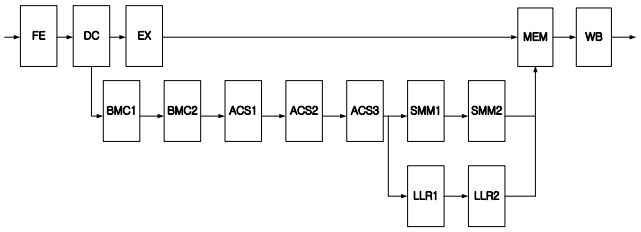

Fig. 2. Instruction pipeline with 11 stages

6. According to the result of Maximum unit, Small state unit decide the small state value.

Addition unit, Decision vector unit, Maximum unit, and Small state unit are divided into 4 blocks. To save the power consumption, we turn off the rest of block when we don't need to operate. That is, when Viterbi decoder with constraint length 5 is selected, only ACS1_K_5, ACS2_K_5, ACS2_1_K_5, and ACS3_K_5 are activated, but When Viterbi decoder with constraint length 9 , all of the part in ACS stage are operated. In Turbo decoder, ACS block is used to compute the forward state metric and backward state metric. Each state metric can be calculated using ACS1_K_5 and ACS2_K_5. The maximum value and small state value are not necessity in Turbo decoder, so Maximum unit and Small state unit are turn off.

E. Survivor Management Unit (SMM)

The flexible traceback units, which form the core of the SMM, use the decoder type, constraint length, current state and the decision vector value stored at a Decision vector memory. The SMM has been divided into three major units: Decision vector memory unit, small state unit and Traceback unit.

Decision vector memory unit and small state unit belong to same stage SMM1, and Traceback unit is in SMM2 stage. In SMM1 stage, after old decision vector is written in another memory address, new decision vector value is written in the first Decision vector memory and small state value is written in the small state register. Traceback unit can be made with one multiplexer and need different state according to the constraint length, so we put the 4 traceback unit for different constraint length.

F. Likelihood Ratio Unit (LLR)

This block is used for Turbo decoder, this block has been divided into three major units: LLR computation unit, Information unit and Additional Information unit. LLR computation unit belong to LLR1, and Information unit and Additional Information unit are in LLR2 stage. LLR computation unit calculate the likelihood ratio using branch metric, forward state metric, and backward state metric, and Information unit get the final information. In Additional information unit, it computes the additional information to decide reliance of the information.

G. Additional Simple Instructions

We put the several simple instructions which are not supported by based 32bit-RISC processor for quick execution.

- Multiplier/Division/Remainder instruction
- Interleaver instruction

  Interleaver instruction is often used in Turbo decoder, the instruction is made like (3).
- Multiply-accumulate instruction (MAC)

  Multiply-accumulate instruction is used to calculate the memory address or the order of array like $(a \times b) + c$ or $(a + b) \times c$.

Table 2. IMT-Advanced requirements related to LTE-Advanced requirements

		Requirement ITU-R	Requirements LTE-A
		M.2134	TR 36.913
Peak data rate(Gbps)		1	1-(DL)
			0.5-(UL)
Latency		C-Plane<100ms	C-Plane<50ms
		U-Plane<10ms	U-Plane<5ms
Peak spectral efficiency(bps/Hz)	DL	15(4X4)	30(8X8)
	UL	6.75(2X4)	15(4X4)
Average spectral efficiency (bps/Hz/cell)	DL	2.2(4X2)	2.4(2X2)
			2.6(4X2)
			3.7(4X4)
	UL	1.4(2X4)	1.2(1X2)
			2.0(2X4)
Cell edge user spectral efficiency (bps/Hz/cell/user)	DL	0.06(4X2)	0.07(2X2)
			0.09(4X2)
			0.12(4X4)
	UL	0.03(2X4)	0.04(1X2)
			0.07(2X4)
Mobility		Up to 350km/h	Up to 350km/h
Bandwidth		>40MHz	Up to 100MHz

IV. Experimental Result of ASIP for Viterbi and Turbo Code

Requirements for 4th generation and the specification of LTE-advanced are listed in Table 2 [2]. The downlink peak data rate of LTE-advanced is 1Gbps and uplink peak data rate is 500Mbps. So system can support maximum 600 Mbps for downlink and 300Mbps for uplink, also should support average 48 Mbps, 52 Mbps, or 74 Mbps for downlink and 24 Mbps or 40 Mbps for uplink. The maximum clocking frequency for our ASIP is 55Mbps, so we meet the average

ISBN: 978-988-18210-0-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Table 3. The number gates of Viterbi and Turbo

Decoder Type	number of gate
Viterbi(7)	128,523
Viterbi(9)	428,494
Viterbi(3-9)	443,706
Turbo	103,348
Viterbi(3-7) + Turbo	207,109
VITURBO [3]	517,968
OUR ASIP	413,295

data rate for LTE-advanced.

Table 3 shows the number of gates of viterbi decoder when it is designed with ASIC. According to the Table 3, the number of gate for Viterbi which constraint length is 7 is 128,523, and the number of gate for Viterbi which constraint length is 9 is 428,494. The number of gate for turbo decoder is 103,348. VITURBO is the multi mode FEC which is designed using ASIC. The number of gate for VITURBO is 517,968. The total gate count of tour ASIP core is 413,295 gates. Compared to the processors of VITURBO with 517,968 for the core, respectively our ASIP saves 20.3% of the area, but provides much more flexibility.

V. Conclusion

LTE-Advanced which is introduced by 3GPP is one of the leading candidates for next generation mobile communication. One of the methods to satisfy the requirements of IMT Advanced among LTE-Advanced technique is channel coding (convolutional coding, and turbo coding). To support several channel coding, the earlier studies have been focused on designing several FEC or multi-mode FEC using ASIC. However designing an ASIC in today's deep submicron geometries is harder than ever, and the problems continue to worsen with shrinking geometries. Therefore design the multi-mode FEC using ASIP is rapidly emerging.

In this paper, we design the multi-mode FEC using LISATek to support several FEC in LTE-Advanced. The total gate count of the ASIP core is 413,295 gates. The maximum clocking frequency for out ASIP is 55 MHz and the data rate is 55 Mbps that meets the average data rate for LTE-advanced.

References

[1] Kurt Keutzer, Sharad Malik, and A. Richard Newton, "*From ASIC to ASIP: The Next Design Discontinuity*," 2002 IEEE International Conference on Computer Design: VLSI in Computers and Processors.

[2] DavidMart´ın-Sacrist´an, Jose F.Monserrat, Jorge, Cabrejas-Pe˜uelas, Daniel Calabuig, Salvador Garrigas, and Narc´ıs Cardona,"*On theWay towards Fourth-GenerationMobile: 3GPP LTE and LTE-Advanced*," EURASIP Journal onWireless Communications and Networking Volume 2009.

[3] Joseph R. Cavallaro and Mani Vaya, "*VITURBO: A RECONFIGURABLE ARCHITECTURE FOR VITERBI AND TURBO DECODING*," Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. (ICASSP '3). vol. 2, 6-10 April 2003, pp. II - 497-500 vol.2.

ISBN: 978-988-18210-0-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)