

## A novel vector-scalar architecture for mobile baseband processing

Si-Lei Shen,

*School of Software Engineering, Beijing University of Technology,  
Beijing, China*

<sup>†</sup>E-mail: shensilei@emails.bjut.edu.cn

The fast development of wireless communication protocols brings in big challenges for designing mobile baseband processor. In this paper, we propose a novel multi-core vector-scalar architecture with heuristic instruction set that can achieve high performance processing with budgeted power consumption and area cost across major computing blocks inside different communication protocols. This proposed architecture consists of four Vector-Scalar Engine Pairs (VSEPs). Each pair can support two data streams for multi-protocol application. The vector-scalar engine pair shares a common pipeline and the vector engine (VE) mainly deals with the symbol level data processing of wireless communication standard, such as OFDM (Orthogonal Frequency-Division Multiplexing) demodulation, while at the same time the scalar engine (SE) calculates the key parameters based on the heuristic instruction. We verify the performance of the architecture through benchmarking typical algorithms such as FFT (Fast Fourier Transform), Channel Estimation and MIMO (Multiple-Input Multiple-Output) detection. The results show that this proposed architecture can achieve better performance in average for 4G wireless communication.

*Keywords:* VSEPs; Multi-protocol Application; Mobile Baseband Processing.

### 1. Introduction

The 4<sup>th</sup> generation communication technology being represented by LTE (Long Term Evolution)/LTE-A (LTE-Advanced) has already come to its commercial stage, LTE-A evolves from LTE and provides higher throughput as well as backward compatibility with the existing LTE standard. The 5th generation is still in research stage by many companies such as Ericsson, Qualcomm, Huawei and Bell Laboratories. And the big challenges to designing a baseband processor are the exponentially growing baseband processing computation density and the coexistence of multi communication standards, such as 3G, 4G, and 5G[1].

Because of design simplicity, currently the ASIC (Application Specific Integrated Circuit) approach still plays the significant role in the mobile baseband chip design, for accelerating the GSM (Global System for Mobile Communication), 3G and 4G communication[2]. This ASIC hardware

acceleration approach separates the whole data stream processing into a few stages, such as I/Q data reception, filtering and imbalance compensation, frequency correction, demodulation and measurement, de-mapping and channel estimation, and decoding[3]. For each stage, a dedicated module is designed, which normally can only handle present protocol for one mode application. When multi-mode application is required, a few more modules are designed for the same processing stage and the on-chip resource reuse among those working modes.

Existence of multiple wireless communication protocols demands a high performance baseband processor which at the same time is tightly constrained by power consumption and area cost. It becomes a must to consider the performance and power as well as flexibility and upgrade cost together during the design stage. In Figure 1, the balance between flexibility and performance for all popular architectures is presented. The general single core processor has best flexibility but less performance power ratio and the ASIC has the best performance power ratio but less flexibility[4].

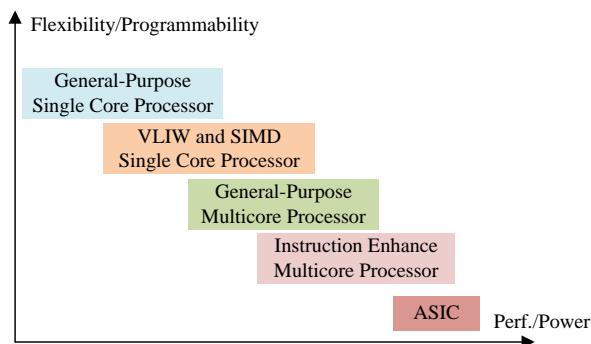


Fig. 1. Architecture balance of flexibility and performance

Software Defined Radio (SDR) baseband technology with high flexibility and low cost to upgrade is gradually replacing the traditional ASIC design method. The multi-core high performance processor architecture based on SDR technology becomes popular in recent years and it can potentially solve the problem[5]. By programming different frequencies, different types of modulation and multiple access way, the SDR processor can adapt to any standard protocol in no time and multi-standard running at the same time in one engine also becomes possible. As the protocol is evolving, increasing demand for high bit rate and small delay interval makes the design of SDR processor architecture a complex project. Many factors must be taken into consideration, such as area, performance, energy efficient. Comparing to SDE, this proposed

VSEP architecture tries to find a better way to tradeoff performance per power and the flexibility/programmability.

For wireless communication, according to the data flow features, in general, the receiver can be divided into three parts based on the characteristics of the processing tasks. As shown in Figure 2, RPU (Radio Process Unit) handles data interface with RF (radio frequency) front end, timing of processing rhythm, DC (Direct Current) offset, and AGC (Automatic Gain Control)/DAGC (Digital Automatic Gain Control), I/Q imbalance compensation, impulse killer, carrier frequency offset, filtering and cyclic prefix removing, etc. SPU (Symbol level Process Unit) deals with most of the computational complex workloads, which includes FFT (Fast Fourier Transform)/IFFT (Inverse Fast Fourier Transformation), channel estimation, MIMO(Multi-Input and Multi-Output) detection and de-mapping of constellation, etc. And the last stage is BPU (Bit level Process Unit), which takes over the de-mapping LLR (Log Likelihood Ratio) data and completes de-scramble processing, de-interleaving, derate-matching, HARQ (Hybrid Automatic Repeat reQuest) processing and turbo decoding.

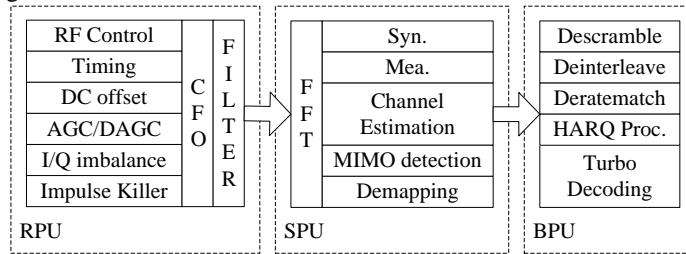


Fig. 2. Architecture of wireless data flow process

In a communication protocol, the symbol level processing is the key module that significantly impacts the performance energy ratio. Symbol level processing requires the computing units to handle multiple data at the same time. The SIMD (Single Instruction Multiple Data) processor or vector processor fits well into this requirement. This paper focuses on the discussion of the proposed novel architecture for symbol level processing which utilizes multiple vector-scalar pairs with a special heuristic instruction set architecture which can regenerate the instructions for the scalar engine in case some parameter data is required while running the vector engine.

This paper is organized as follows. Section 2 overviews the background information on SIMD processors and vector processing technology. The main concept underlying the proposed architecture is given in Section 3. Finally, evaluation results are discussed in Section 4.

## 2. Backgrounds

The fundamental algorithms inside communication standards include FFT/IFFT, channel estimation and MIMO detection which require data parallelism for higher throughput. By exploiting SIMD architecture, a few solutions have been proposed in literatures.

Schoenes, M and Eberli, S have introduced a novel DSP (Digital Signal Processing) architecture for SDR, which achieves very high data throughput by means of massively parallel arithmetic unit[6]. Based on a radix-4 butterfly structure which is optimized for complex-valued arithmetic, the processor's data-path enables extremely fast FFT computations. Furthermore, exceptional programming flexibility and increased code efficiency are offered by a reconfigurable instruction set.

Kees van Berkel et al. have presented a heterogeneous hardware architecture with the programmable vector processor EVP (Embedded Vector Processor) as the key component which can support WLAN (Wireless Local Area Networks), UMTS (Universal Mobile Telecommunications System), and other standards[7]. The SIMD width is scalable and the maximum parallelism available equals to five vector operations, plus four scalar operations, three address updates and additional loop-control.

In order to achieve a more efficient and higher performance architecture for multi-standard processing Mark Woh et al. have designed an enhanced SDR architecture, named AnySP, which consists of SIMD and scalar data paths. The SIMD data path consists of eight groups of 8-wide SIMD units, which can be configured to implement SIMD widths of 16, 32, and 64. Each of the 8-wide SIMD units is composed of groups of Flexible Functional Units (FFU). The FFUs contain the functional units of two lanes that are connected through a simple crossbar. Eight SIMD register files feed the SIMD data path. Each register file contains 16 entries where each entry is 8-element wide. The swizzle network aligns data for the FFUs. It can support a fixed number of swizzle patterns of 8-, 16-, 32-, 64-, and 128-wide elements. All these features greatly enhance the multi-standard supports and increase the performance significantly[8].

MAPro is a tiny processor for reconfigurable baseband modulation mapping, which was presented by Liang Tang et al[9]. MARro has provided a single low-cost flexible hardware platform for emerging communication protocols and applications in modern embedded systems.

Omer Anjum et al. have proposed an MPSoC (Multi-Processor System-on-Chip ) design for the baseband processing of a 20 MHz LTE system. Instead of using conventional DSPs/VLIW architectures, the proposed TA (Triggered Architecture) has been selected as processing element (PE) of the MPSoC.

Processing tasks are statically scheduled. Synchronization among the PEs is based on polling of a shared memory space[10].

Seyed A. Rooholamin and Sotirios G. Ziavras presented an innovative architecture for a VP (Vector Processor) which separates the path for performing data shuffle and memory-indexed accesses from the data path for executing other vector instructions that access the memory. This separation speeds up the most common memory access operations by avoiding extra delays and unnecessary stalls. In the lane-based VP design, each vector lane uses its own private memory to avoid any stalls during memory access instructions[11].

The above referred architectures on one side improve the performance and/or efficiency, but have some obvious limitations. They exploit the performance improvement mainly by increasing the width of SIMD, adding extra functional units and doubling the number of PEs. For some design, an extra scalar processor is integrated for control stream and supplementary computation. Also, analyzing the multiple-cycle delay of vector processing and how much time/cycles being wasted due to waiting for results of vector computing were not deeply exploited. In the following, we present an novel vector-scalar engine pairs architecture which greatly improves the efficiency of the vector processing by designing a special heuristic instruction set architecture which can automatically generate the temporary instructions for scalar engine that runs in a separate pipeline. The scalar engine can complete parameters computation at right time and feed them to the vector engine. The following section will explain the novel design in details.

### **3. Processor Design**

#### **3.1. *Proposed processor***

The proposed baseband processor is a fully programmable architecture as shows in Figure 3. A novel 4 Vector-Scalar Engine Pairs (VSEPs) architecture is designed for multi-protocol based applications, it fully supports LTE-A core algorithms processing and can be easily adapted to future 5G core algorithms processing. The proposed 64-bit heuristic instruction set architecture is designed to make sure that the vector-scalar engine pairs run efficiently without pipeline stall even if computationally intensive algorithms are being processed. A two-level memory hierarchy design (private data memory and public shared data memory) can minimize the data transactions. The shared data memory is further divided into four banks in order to be accessed by four computation pairs at the same time with the help from the on-chip interconnect unit. This interconnect unit is designed with resource scheduling pool and scheduler for high efficient routing management. In figure 3, PC means program counter, IB represents

instruction buffer, ID is for instruction decoding unit and IG is for the instruction regenerating unit that produces the new instruction sequence from the ID unit. The new instruction sequence is buffered in ISB and is waiting for decoding by the ISD unit. Once decoded, the instructions are fed into scalar engine.

With the auxiliary support of the heuristic instruction set, the scalar engine can easily handle the required parameters calculation, such as generation of Gold Sequence (GS), searching for the peak value, generation of the twiddle factor for FFT/IFFT, etc. The outcomes can be passed to the vector engine by Inter-Engine Unit (IEU) and this kind of pairing working design efficiently eliminates the possible pipeline stall for computation intensive algorithms and applications.

### **3.2. Working mode**

For a better balance between performance and power dissipation, the vector-scalar engine pairs VSEP architecture provide an efficient way for self-adaption and scalability. The VSEP pair can run independently or all the pairs run simultaneously while obtaining synchronization based on requirement of communication algorithm.

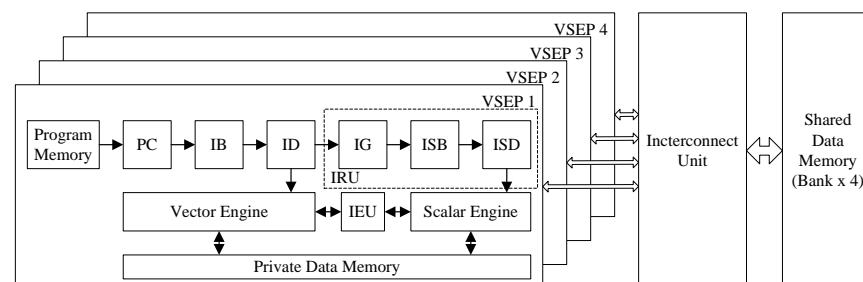


Fig. 3. The structure of the proposed processor architecture

#### *3.2.1. Single pair mode*

After loading programs into the PM (Programm Memory), the CMU (Clock Management Unit) turns on the clock of the VSEP pair. After initiation is done, the VSEP pair go to sleep, waiting for system tick. If there is data processing required by application, the system tick wakes up the scalar engine. According to the instruction sequences, scalar engine triggers the data transportation into the private memory and starts the parameters calculation. After passing the parameters into the IEU, the vector engine is fed with the required scalar data. The vector engine acquires the parameters and continues the work. At the same time, the scalar engine works on the next round of parameters calculations.

### 3.2.2. Multiple pairs mode

Taking energy dissipation into consideration, each vector-scalar engine pair comes into service freely and different tasks can be dispatched to any available vector-scalar pair. Of course, the master control unit of the baseband chip can turn off any pair when it is not needed any more. For massive data processing, such as MIMO, all four vector-scalar engine pairs can cooperate easily with the help of the on chip synchronization mechanism.

### 3.3. Design the heuristic instruction set

In order to support heavy workload operations such as FFT and MIMO matrix operations, the heuristic instruction set architecture is proposed and designed to fully utilize the hardware VSEP pairs and the instruction regeneration units IRU. The basic idea of the heuristic instruction is to divide the 64-bit instruction into two parts. The 46-bit instruction at the front is decoded for vector engine processing while the remainder instruction bit is passed to instruction regenerate unit for scalar engine processing, as shows in Figure 4.

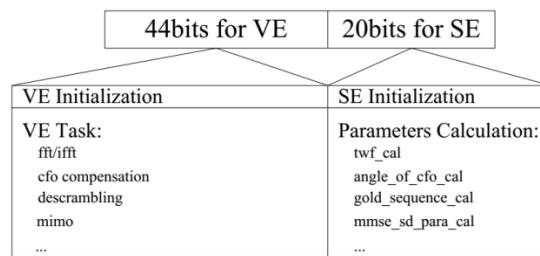


Fig. 4. The functions of proposed ISA

## 4. Evaluation

### 4.1. Application scene

A typical wireless communication processing of receiver is shown in figure 5. The processes in dashed rectangle is focused for the proposed architecture, which include: FFT, Synchronization, Measurement, Channel Estimation and MIMO Demodulation.

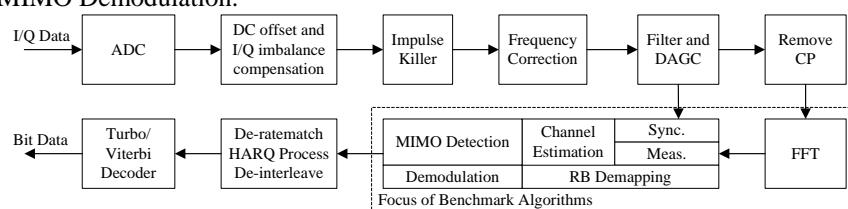


Fig. 5. A Typical wireless communication processing of receiver

#### 4.2. Experiment

To evaluate the efficiency of proposed processor, three categories of core processes for wireless communication based on OFDM (Orthogonal Frequency Division Multiplexing) technology are selected, which include FFT, Channel Estimation and MIMO Detection and the proposed architecture can perform excellently.

### 5. Conclusions

The modem stage of an SDR requires software flexibility to cope with the multitude of wireless standards, their evolution, and with algorithmic improvement (including bug fixes and in-field upgrades) without the need to re-spin an IC. The proposed architecture, with its powerful VE-SE pairs outperforms conventional DSPs by an order of magnitude or more, in a power-efficient way. Accordingly, the VE-SE pairs can be a key component of an SDR, where it can save silicon area by both intra-standard and inter-standard reuse and it can potentially handle multiple standards simultaneously.

### References

1. G. S. Kim, Y. H. Je, and S. Kim, “An adjustable power management for optimal power saving in lte terminal baseband modem,” in IEEE Transactions on Consumer Electronics, vol. 55, pp. 1847–1853, IEEE, 2009.
2. F.-L. Yuan, C. C. Wang, and T.-H. Y. D. Mark., “A multi-granularity fpga with hierarchical interconnects for efficient and flexible mobile computing,” in IEEE JOURNAL OF SOLID-STATE CIRCUITS, vol. 50, pp. 137–149, IEEE, 2015.
3. C. Studer, C. Benkeser, S. Belfanti, and Q. Huang, “Design and implementation of a parallel turbo-decoder asic for 3gpp-lte,” in IEEE JOURNAL OF SOLID-STATE CIRCUITS, vol. 46, pp. 8–17, IEEE, 2011.
4. A. Ardakani and M. Shabany, “A novel area-efficient vlsi architecture for recursion computation in lte turbo decoders,” in IEEE TRANSACTIONS ON CIRCUITS AND SYSTEM: EXPRESS BRIEFS, vol. 62, pp. 568–572, IEEE, 2015.
5. A. Barreteau, S. L. Nours, and O. Pasquier, “A case study of simulation and performance evaluation of a sdr baseband architecture,” in J Sign Process Syst (2013), vol. 73, pp. 267–279, Springer, 2013.
6. M. Schoenes and S. Eberli, “A novel simd dsp architecture for software defined radio,” in Proceedings of the 46th IEEE International Midwest Symposium on Circuits and Systems, vol. 1, pp. 1443–1446, IEEE, 2003.

7. K. van Berkel and F. Heinle, “Vector processing as an enabler for software-defined radio in handheld devices,” in EURASIP Journal on Applied Signal Processing, vol. 16, pp. 2613–2625, EURASIP, 2005.
8. M. Woh, S. Seo, and S. Mahlke, “Anysp: Anytime anywhere anyway signal processing,” in Proceedings of the 36th Annual International Symposium on Computer Architecture, pp. 128–139, IEEE, 2009.
9. L. Tang, J. A. Ambrose, and S. Parameswaran, “Mapro: A tiny processor for reconfigurable baseband modulation mapping,” in 2013 26th International Conference on VLSI Design and the 12th International Conference on Embedded Systems, IEEE, 2013.
10. O. Anjum, T. Ahonen, and J. Nurmi, “Mpsoc based on transport triggered architecture for baseband processing of an lte receiver,” in Journal of Systems Architecture, vol. 60, Elsevier, 2014.
11. S. A. Rooholamin and S. G. Ziavras, “Modular vector processor architecture targeting at data-level parallelism,” in Microprocessors and Microsystems, vol. 39, Elsevier, 2015.