On Chip Communication Networks for Multiprocessor Turbo Decoders

Harshini.N

Department of Electronics and Communication Engineering

Vijaya Vittala Institute of Technology

Bangalore-560077

Abstract-The multi-application specific instruction processor (ASIP) architecture gives flexible high-throughput turbo decoders. This brief proposes a network-on-chip (NoC) structure for multi-ASIP turbo decoders. High speed downlink packet access extends and improves the existing of telecommunication networks using the WCDMA protocols. The process of turbo decoding is studied, and the addressing patterns for turbo codes in long term evolution (LTE) and High Speed Downlink Packet Access (HSDPA) are analyzed. Based on this analysis, two techniques, sub-networking and calculation sequence, are proposed for reducing the complexity of the NoC. The implementation results show that the proposed structure gives an improvement of high through-put for HSDPA and high throughput/area efficiency compared with state-of-the-art NoC solutions.

Index Terms-Network-on-chip (NoC), turbo decoder, VLSI and LTE.

I. INTRODUCTION

The smartphones and the PCs brought huge applications with high through-put and efficiency in the wireless networks. So the data rate transmission has been increased from kilobytes/sec to gigabytes/sec which is required in both national and international mobile telecommunications. The wireless 3GPP (LTE) and HSDPA has bought effective evolution in the mobile networks.

The turbo decoders has a wide demand in modern wireless systems for using near Shannon performance. Turbo decoders occupies more physical layer computations which results in occupation of large chip area such as, code length, code generator and interleaving patterns. ASIP is used for High through-put for multi-application instructions.

Manuscript received October 26, 2013; revised June 9, 2014 and November 11, 2014; accepted December 28, 2014. Date of publication February 6, 2015; date of current version December 24, 2015. This work was supported in part by the National Science and Technology Major Project of China under Grant 2011ZX03003-003-03, in part by the State Key Laboratory of ASIC and System, and in part by the China Southern Grid Yunnan Electric Power Science Research Institute under Project K-YN2012-477.

Q. Yang and X. Zhou are with the State Key Laboratory of ASIC and System, Fudan University, Shanghai 200433, China (e-mail: yangqingqing@fudan.edu.cn; xiaofangzhou@fudan.edu.cn).

G. E. Sobelman is with the Department of Electrical and Computer Engineering, University of Minnesota. Minneapolis, MN 55455 USA (e-mail: sobelman@umn.edu).

X. Li is with the State Key Laboratory of ASIC and System, Fudan University, Shanghai 200433, China, and also with the State Key Laboratory of Transducer Technology, Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Science, Shanghai 200050, China (e-mail: xxli@mail.sim.ac.cn).

Color versions of one or more of the figures in this paper are available online athttp://ieeexplore.ieee.org. Digital Object Identifier 10.1109/TVLSI.2015.2390264

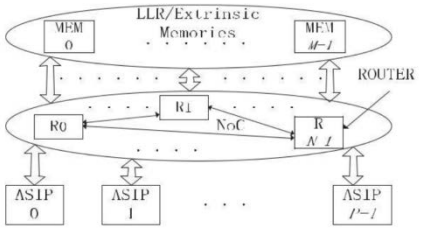

Fig. 1. Multi-ASIP for turbo decoding.

The fig. 1 shows the architecture formed by numerous ASIPs along with network on chip. This provides the needs of a specific application for flexibility and guides for selecting the correct number of ASIPs.

According to the previous research on the turbo decoders the main cause of the complexity in ASIP and NOC architectures, was low efficiency when it as compared with the decoders. In this brief, we focus on the NoC architecture. Our proposed work on NOC architecture supports high throughput and low complexity

II. MULTI-ASIP ARCHITECTURE USING TURBO DECODERS

Here the code frames with N-bits of information are divided into P non overlapped windows where single windows are processed by application specific instruction processors. By using ASIPs helps for memory addressing building radix for decoding algorithms and LLRs (Log likelihood ratios) of extrinsic information. Since the decoding process is programmed in the ASIPs, and the NoC provides full connectivity between ASIPs and memories, this kind of architecture can satisfy the flexibility requirement of various turbo codes.

The turbo decoding process in the multi-ASIP is briefly described in the following paragraphs.

After channel LLRs are available in the memories, parallel ASIPs inject reading transactions (RTs) into the NoC. These RTs reach the destination memories via routers, and then the memories read out the LLRs and LEs, sending them back into the NoC as feedback transactions (FTs). After these FTs arrive at the ASIPs, they are used for the calculation of new LEs and hard decisions. Finally, new LEs are sent back to memories as write transactions (WTs).

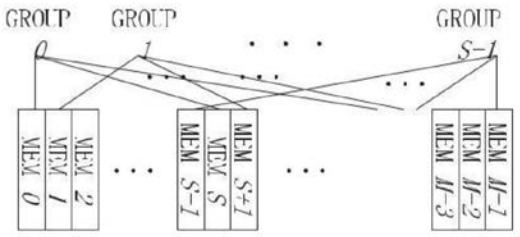

Fig. 2. M memory banks split into S groups.

III. ADDRESSING MODE ANALYSIS

Generally speaking, the complexity of a NoC is related to the number of routers: for a given topology, as the number of routers increases, transactions will need more hops to get to their destinations and stay longer in the network, thus more buffer memories will be required. Another extreme situation is the crossbar structure whose latency is very low, but the long wires which cross the whole decoder will degrade the timing. If we could divide the whole network into several sub-networks and use crossbars to connect ASIPs' ports to these sub-networks, the new network would still have the ability to resolve conflicts as a whole network but with less complexity and latency. The key issue is whether transactions can be assigned to these sub-networks uniformly. If in a certain period all transactions are sent to one sub-network, then each sub-network will require a large amount of buffering. Fortunately, in turbo decoding, since the interleaved has been designed to redistribute the information bits as randomly as possible, the addressing will be uniformly distributed in the long run. Considering that the ASIPs read/write data from/to memories, we split the *M* memory banks into *S* logical groups. Then, the NoC can be divided into *S* sub-networks according to the partition of memory groups. The maximum and average value of CV*(t,T,M,S)* for LTE and HSDPA with *P*= 16 ASIPs generating 64 addresses per cycle. Without loss of generality, we choose the maximum code length parameters for LTE and HSDPA, which are *N* =6144 and *N* =5114, respectively. For LTEs turbo code, thedifference in flow between sub-networks changes drastically when the period is very short, i.e., *T*= 4. However, for a longer period, the flow between sub-networks is much more uniform. For the turbo code in HSDPA, the maximum CV*(t,T,M,S)* is below 0.3 and the average of CV*(t,T,M,S)* is below 0.2, indicating that the flow is uniform.

As long as the transactions can be distributed into these subnet-works in time and these sub-networks can sustain that number of transactions, then these sub-networks can share these transactions equally. Address translation is straight forward the address hashing is performed to increase the bank availabity. Several network topologies exits for servers configured to accept multiple client corrections, these are controlled with the topology.

IV. PROPOSED NOC ARCHITECTURE

A. Topology and NoC Data Format

The proposed NoC architecture is shown in Fig. 5. Since the whole NoC is divided into parallel sub-networks, RTs and WTs from ASIP must be distributed to these sub-networks. This process is finished by DISTRIBUTORs. Each DISTRIBUTOR buffers data from ASIP ports, and sends them to different sub-networks according to the rule which partitions memories into *S* groups: a transaction will be sent to the DEST ADDR modulo *S* sub-network, where DEST ADDR is the destination memory bank address. As a memory bank receives an RT from the sub-network, it will send a FT back

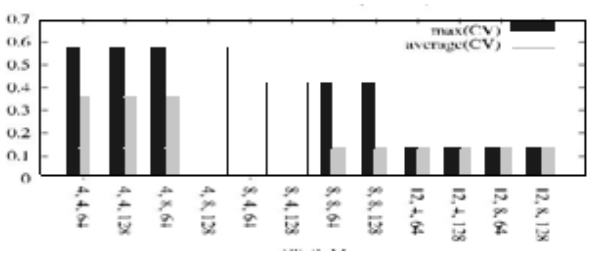

Fig. 3. CV statistics for LTE

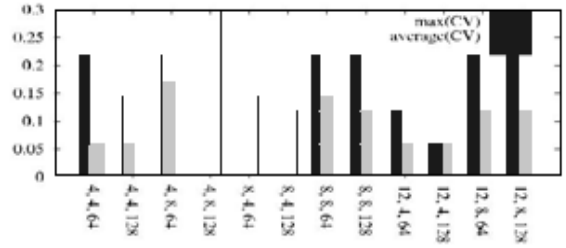

Fig. 4. CV statistics for turbo codes

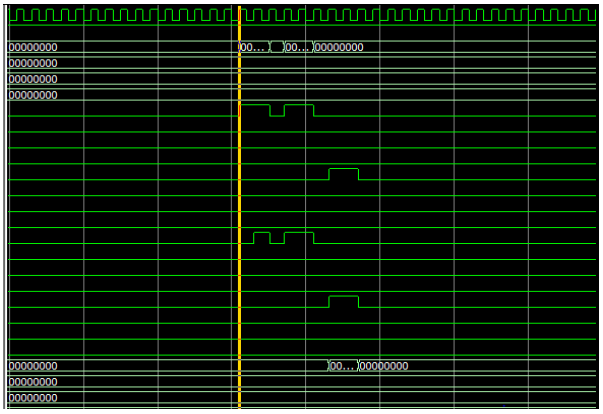

Fig. 5. Top wave simulation.

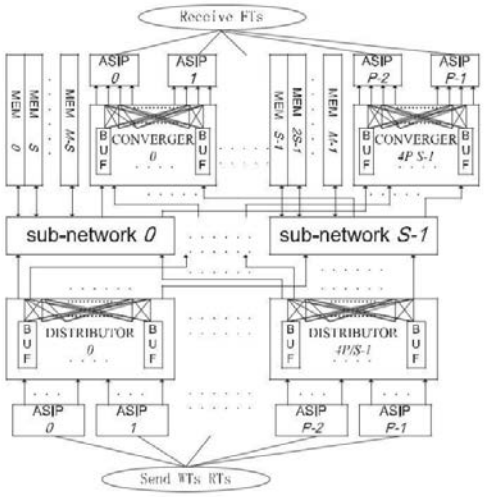

Fig. 6. Proposed NoC architecture.

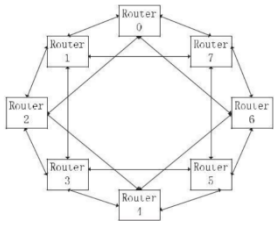

Fig. 7. Topology of one sub-network.

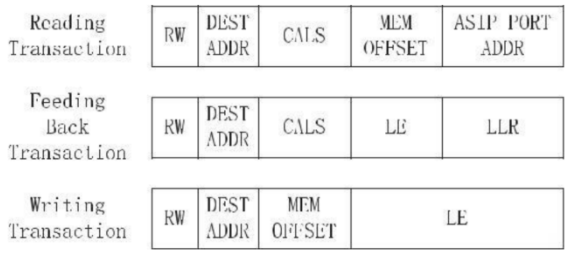

Fig. 8. NoC data format.

This FT will go along routers back to the router corresponding to the reading ASIP, and then get to the ASIP through a CONVERGER which has the reverse effect of the DISTRIBUTOR module. Both the DISTRIBUTOR and CONVERGER are implemented by *S*×*S* crossbar switches with input buffers. These two modules are essential for splitting the whole network into parallel sub-networks. Note that for drawing convenience, the same ASIP blocks are shown at both the top and the bottom of the figure.

The topology of one sub-network with eight routers. We use a ring style topology. Since the ASIPs will send RTs to memories and the memories will send FTs to ASIPs at the same time. The Read/Write (RW) field indicates whether the transaction is for reading or writing. In an RT, the RW field is READ, and in an FT

In turbo decoding, there exists a trellis sequence. Thus, the processing of LLR/LEs will follow a fixed order. This order is given by the TRI of LLR/LEs in a window, whose bit width is usually determined by *W*. For example, in LTE the maximum value of *N* is 6144, so the bit width of TRI in a window is at least nine for *P*= 16. This occupies about 1/3 of the bit width of RT and FT. Moreover, since the TRI represents the processing order, it can also be used for priority selection in the routing algorithm. However, the wider the priority, the longer the critical path that will be required. To reduce the bit width of the TRI, we introduce a quantity called the CALS instead.

A. Routing Policy

Generally speaking, there are two kinds of routing policies:

1) Deterministic routing and 2) adaptive routing. The adaptive scheme usually determines the path of a transaction according to the networkcondition. It has high-bandwidth utilization but needs to employ a complex arbitration scheme. The deterministic scheme is much simpler; we use logic to determine the path. However, more area is required to store data and bandwidth utilization will be lower.

In the proposed NoC, we adopt the deterministic routing policy for its shorter critical path and lower complexity. Transactions will be routed along the shortest path, and for each hop the far most router in the path will be chosen. This kind of routing policy can reduce the total number of hops with the cost of some delay, which will be decreased by a proper arbitration scheme. The CALS is used for priority comparison. The priority decreases in the anticlockwise direction, so earlier NoC transactions have a higher priority. The priority comparison can be implemented using the modulo comparison algorithm. Since WTs have no CALS field and will not incur FTs, we give WTs the highest priority. This kind of priority scheme provides quality of service, but also has a high complexity and a long critical path.

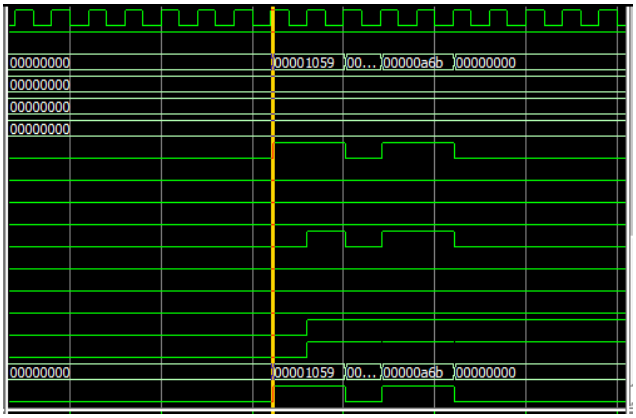

Fig. 9. Pipe priority wave simulation

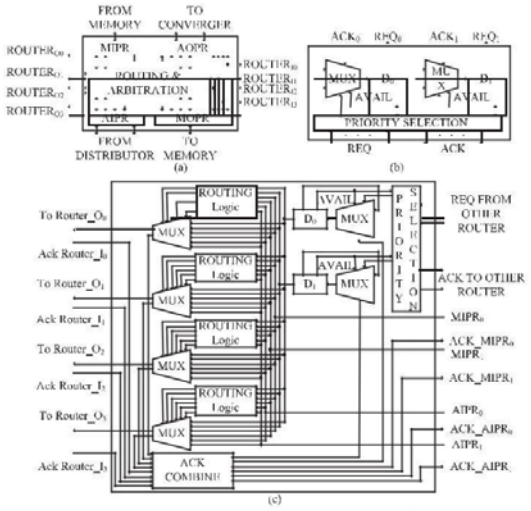

Fig. 10.(a) Router architecture. (b) Pipeline registersmodule.(c) R&A module

B. Flow Control

Flow control is used to solve the deadlock problem. We use two kinds of schemes at the ASIP side: 1) CALS based and 2) ASIP-balancing based.

The CALS-based scheme is to constrain the number of transactions in the NoC for each ASIP. We denote the CALS of RTs sent to NoC as $CALS_{RT}$, and the CALS of FTs currently processed.

When the buffer in a DISTRIBUTOR is full, the connected ASIP will be halted waiting for free space in the buffer. In that situation some ASIPs will send more RTs and some will send less. This would lead to an imbalance of the CALS in RTs and FTs, which could invalidate the CALS-based flow control scheme. In addition, there may be some ASIPs that send too many RTs to the NoC, thus intensifying congestion.

To solve these problems, we introduce a scheme to balance the number of halted cycles in the ASIPs.

We record the number of paused cycles (N_{halt}) for every ASIP. When the difference in halted cycles exceeds a given limit called $N_{\text{halt\_limit}}$, the ASIPs having the smallest N_{halt} will be blocked.

C. Router Architecture

An example of a sub-network with eight routers is given. For timing considerations, we use pipeline registers to reduce the critical path. In addition to these registers are used for data buffering. The structure of the ASIP input pipeline registers, ASIP output pipeline registers (AOPR), memory input pipeline registers, and memory output pipeline registers (MOPR). For each pipeline register module, there are two NoC transaction registers, which accept requests in a ping-pong style. The ROUTING and ARBITRATION (R&A) module selects data at the ports to send to other Routers and stores data from other routers. Because of the routing policy, the destination of data from the neighboring two routers is the current router,

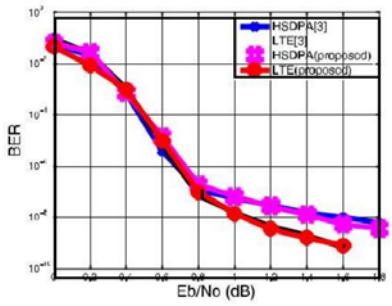

Fig.11. BER performance of HSDPA(N= 5114) and LTE(N= 6144).

TABLE I

COMPARIS ON RESULTS

Architecture	Proposed	FA[3][b]	AP[3][7][b]
Technology	0.13um	0.13um	0.13um
Clock Freq(MHz)	300	200	200
Area(mm^2)	3.3[a]	5.68	3.45
Max Throughput (Mbps)[c]	565(HSDPA) 694(LTE)	372(HSDPA) 312(LTE)	372(HSDPA) 312(LTE)
Throughput to area ratio(Mbps/mm^2)[c]	171(HSDPA) 210(LTE)	65(HSDPA) 55(LTE)	112(HSDPA) 90(LTE)
Power Efficiency (mW/Mbps)	0.76(HSDPA) 0.62(LTE)	- -	1.36(HSDPA)[d] 1.64(LTE)

(a) For a fair comparison with references, this value includes two parts: the NoC area ($2.8mm^2$) and interleaving address memory area ($0.5mm^2$).
(b) The throughput in reference [3] is for eight iterations. However, [3] used a shuffling algorithm, which needs a factor of 1.7 more iterations to achieve the same decoding performance.
(c) Since [2] gave a list of results, we chose the highest throughput from Table I in [2], which corresponds to to R=1.0 and D=4 parameters.
(d) We obtained the power data for [3] from [7].

So data from these two routers will be directly stored into AOPR or MOPR The structure of the R&A module. From considerations of complexity and critical path, only the CALS-based priority is used in the design of the pipeline registers. The R&A module used a fixed priority scheme.

V. IMPLEMENTATION RESULTS

We implemented a cycle-accurate multi-ASIP architecture simulator in System C. This was used to generate the test bench and estimate the decoding throughput. Then, we implemented the NoC circuit with Verilog-Hardware Description Language and synthesized it in a 0.13-μm ASIC standard cell library. We choose six for $It_{\text{seq}}$and five forIt_{int}. The LLR and LE are 6 and 8 bits, with three fractional bit gives the simulated bit error rate (BER) performance curve, which is very close. The maximum frequency is 300 MHz. Through simulations, we found 5 bits to be sufficient for CALS, so we chose R= 32. In addition, $\text{CALS}_{\text{limit}}$ is 10 and $N_{\text{halt\_limit}}$ is six in both LTE and HSDPA cases.

We use P= 16 ASIPs for high throughput. Considering memory conflicts in HSDPA, M= 128 memory banks are adopted to provide higher memory bandwidth. Thus, two NoCs are implemented, each connected to 64 memory banks, 32 ASIP-reading ports and 32 ASIP-writing ports alternately. S is selected to be eight, which is suitable in our design case. Larger S leads to higher imbalance between sub-networks, but smaller S causes more latency.The implementation results and comparison with previous work are listed in Table I.

Our architecture improves the throughput-to-area ratio by 163% for HSDPA and by 281% for LTE com-pared with the fully adaptive (FA) design, and by 53% for HSDPA and 133% for LTE compared with all pre-calculated (AP) design in. Our architecture belongs to the FA category, which stores and forward packets according to the routing algorithm.

As for the AP architecture, that requires an external memory to store the routing information, as pointed in, which amounts to 64 Mb for the HSDPA case. The advantages of our approach are due to the sub-network scheme and the CALS technique. For each sub-network, there are only eight routers, which reduces the complexity of each router and the critical path. The adoption of CALS reduces the bit width of NoC packets and simplifies the priority selection. Thus, even though we have 128 routers, the area is still smaller than that of the design in which uses 64 routers.

The reduction in power that we have achieved arises from two aspects of the design. First, our architecture requires a fewer number of iterations than shuffle decoding to achieve the same BER performance. Second, our architecture requires fewer hops. This is due to the fact that when the ASIPs operate in the sequential mode they only need to access nearby memories. In most cases, these memories will be connected to the same router as the ASIP.

V. CONCLUSION

In this brief, we have proposed a novel NoC architecture for a multi-ASIP turbo decoder. By dividing the entire network into several sub-networks and adopting the notion of the calculation sequence, we successfully achieved a much higher throughput with a lower NoC total area. Comparisons with previous designs for area, throughput, and power efficiency have been given. Our design techniques can also be used in other applications, such as Low Density ParityCheck decoding. The sub-network scheme can be employed wherever there is a uniform addressing mode.

REFERENCES

[1] T. Vogt and N. Wehn, "A reconfigurable ASIP for convolutional and turbo decoding in an SDR environment," *IEEE Trans. Very Large ScaleIntegr. (VLSI) Syst.*, vol. 16, no. 10, pp. 1309–1320, Oct. 2008.

[2] O. Muller, A. Baghdadi, and M. Jézéquel, "From parallelism levels to a multi-ASIP architecture for turbo decoding," *IEEE*

Trans. Very LargeScale Integr. (VLSI) Syst., vol. 17, no. 1, pp. 92–102, Jan. 2009.

[3] M. Martina and G. Masera, "Improving network-on-chip-based turbo decoder architectures," *J. Signal Process. Syst.*, vol. 73, no. 1, pp. 83–100, Oct. 2013.

[4] M. Martina and G. Masera, "Turbo NOC: A framework for the design of network-on-chip-based turbo decoder architectures," *IEEETrans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 10, pp. 2776–2789,Oct. 2010.

[5] C. Studer, C. Benkeser, S. Belfanti, and Q. Huang, "Design and imple-mentation of a parallel turbo-decoder ASIC for 3GPP-LTE," *IEEE J.Solid-State Circuits*, vol. 46, no. 1, pp. 8–17, Jan. 2011.

[6] C. Shung, P. Siegel, G. Ungerboeck, and H. Thapar, "VLSI architectures for metric normalization in the Viterbi algorithm," in *Proc. GLOBECOM*, Apr. 1990, pp. 1723–1728.

[7] M. Martina, G. Masera, H. Moussa, and A. Baghdadi, "On chip intercon-nects for multiprocessor turbo decoding architectures," *MicroprocessorsMicrosyst.*, vol. 35, no. 2, pp. 167–181, Mar. 2011.

[8] F. Naessens *et al.*, "A 10.37 mm^2 675 mW reconfigurable LDPC and turbo encoder and decoder for 802.11n, 802.16e and 3GPP-LTE," in *Proc. IEEESymp. VLSI Circuits (VLSIC)*, Jun. 2010, pp. 213–214.