

GALS NOC ARCHITECTURES ON FPGA DEDICATED TO MULTISPECTRAL IMAGE APPLICATIONS

Linlin Zhang¹, Virginie Fresse¹, Anne-Claire Legrand¹, Mohammed Khalid²

¹Hubert Curien Laboratory CNRS UMR 5516, St Etienne, France
 {lin.zhang, virginie.fresse, anne.claire.legrand}@univ-st-etienne.fr

²RCIM: dept. of Electrical & Computer Eng. University of Windsor, Windsor, ON, Canada
 mkhalid@uwindsor.ca

ABSTRACT

An efficient Network on Chip (NoC) implemented on Field Programmable Gate Arrays (FPGA) is proposed for the data communication of multispectral image analysis algorithms in an adaptive architecture. The architecture design is based on the linear effort property and reusable IPs. Multispectral image data are several types of data, different length values. According to the requirement of the multispectral image data communication, this NoC used Virtual Channels (VC) with packet-switching. Implementations on FPGA show that a NoC-based communication scheme provides a superior solution than classical point-to-point communication.

1. INTRODUCTION

Image analysis applications consist of extracting some relevant parameters from one or several images. Embedded systems for real time image analysis allow computers to take appropriate actions for processing images under real-time hard constraints and often in harsh environments. Current image analysis algorithms are resource intensive so the traditional PC or DSP-based systems are unsuitable as they cannot achieve the required high-performance. In this case, the FPGA devices are widely used because they can achieve high-speed performances in a relatively small footprint. Modern FPGAs integrate many heterogeneous resources on one single chip. The resources on an FPGA continue to increase at a rate that only one FPGA is capable to handle all processing operations, including the acquisition part. That means incoming data from the sensor or any other acquisition devices are directly processed by the FPGA, therefore, no other external resources are necessary.

Today, many designers of such systems choose to build their designs on Intellectual Property (IP) cores. Most IP cores are already pre-designed and pre-tested and they can be immediately reused [1] [2] [3]. Without reinventing the wheel, the existing IPs and buses which are well defined from the designer map can be used to build the architecture directly. Although the benefits of using existing IPs are substantial, the Design Space Exploration (DSE) and IP designs remain a difficult task. It is hard to predict the number and the type of required IPs and buses from a set of existing IPs library.

In this paper, a parameterized architecture on FPGA is dedicated to image analysis implementations. Point-to-point communication was used for the Particle Image Velocimetry (PIV) processing in the prior work [4]. New implementation of the multispectral image analysis gives more constraints as large quantity of data transmission, high flexibility and well adaptation ability. Classical point-to-point communication has its drawbacks which limit its utilization; NoC with Virtual Channels data flow can be a superior solution.

In this paper, Section 2 presents the multispectral image authentication process which needs to be implemented in the architecture. Section 3 shows the global image analysis architecture. Section 4 explains why the classic point-to-point communication is not suitable for our application. Section 5 presents NoC structure proposed for architecture. Results are presented and analyzed in Section 6 and Section 7 gives the conclusion and perspectives.

2. MULTISPECTRAL IMAGE AUTHENTICATION

One multispectral image is composed of several spectral images. Every spectral image corresponds to the acquisition of the same physical area and scale but of a different spectral band (Fig.1).

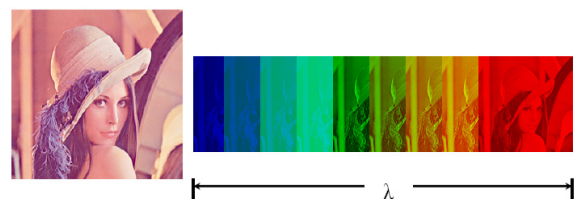


Figure 1 – Multispectral image example: Lenna.

One multispectral image is composed of several spectral images.

The aim of the multispectral image correlation is to compare two multispectral images:

- Original image (OI): its spectra have been stored in the system as the reference data.
- Compared images (CI): its spectra are acquired by a multispectral camera which is connected to the system.

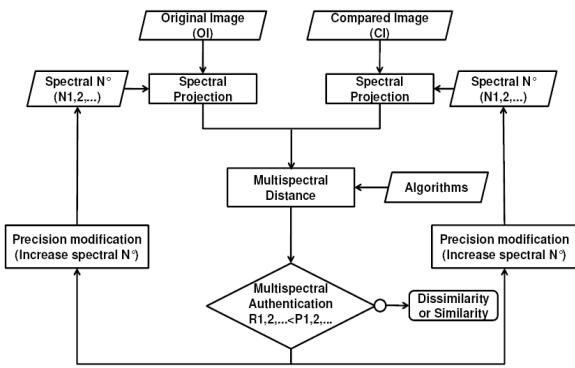


Figure 2 – General multispectral authentication process.

R–Result of the each step of calculation
P–Precision of the each multispectral distance algorithms
N–Number of the spectral band

For the art authentication process, OI is the information of the true picture, and the CI are the others “similar” pictures. With the comparison process of the authentication (Fig.2), the true picture can be found among the false ones by calculating the distance of the multispectral image data. For this process, certain algorithms require high precision operations which imply large amount of different types of data (e.g. floating-point, fixed-point, integer, BCD encoding, etc) and complex functions (e.g. square root or other nonlinear functions). Several spectral projection and distance algorithms can be used in the multispectral authentication. One analysis of these algorithms is presented in [5]. These calculations are based on the spatial and spectral data which makes the memory access as a bottleneck in the communication. This part is implemented as several processing nodes in the architecture.

3. IMAGE ANALYSIS ARCHITECTURE

General image analysis architectures have 4 types of operations which are realized as 4 types of nodes in the proposed architecture on FPGA:

- Control operation/node (CN): it controls the whole system. Commands are sent from CN to the other nodes in the architecture.
- Acquisition operation/node (AN): it connects the multispectral camera by using one USB connector. It is controlled by CN and sends the multispectral data to the memory.
- Storage operation/nodes (SN): it is the memory part which stores all the data for the multispectral image analysis.
- Processing operation/nodes (PN n°1 to n°N): it contains all the multispectral authentication algorithms which presented as several IP blocks in the architecture.

Because the command flow and the final result have really low bandwidth compared to the multispectral image data for the calculation, these two flows are grouped together as one result & command flow which transmits in the 4 flits of 8-bit wrapper. The topology for this communication is a “Ring” as shown in Fig.3.

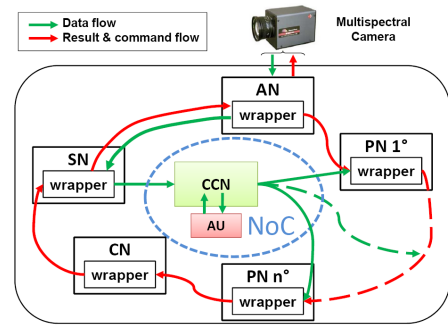


Figure 3 – The proposed architecture for image analysis application.

CCN – Central Coordination Node AU- Arbitration Unit

To run each node as its best frequency, Globally Asynchronous Locally Synchronous (GALS) [6] is used in this architecture. Because it can simplify global timing and synchronization problems, improve performance, reliability, and development time. The frequencies for each type of nodes in the architecture are shown in Section 6 with the other results. More details are presented in [4].

4. THE POINT-TO-POINT COMMUNICATION

The multispectral data used Binary-coded decimal (BCD) encoding with fixed-point floating form. Its main virtue is that it allows easy conversion to decimal digits for fast decimal calculations.

Four types of data (each type has an identical number *id*) which are defined by analyzing multispectral image algorithms are:

- Coef: Coefficient data which signifies the normalized values of difference color space vector. 56-bit DCB code (*id* “00”).
- Org: Original image data which are stored in the SN. 48-bit DCB code (*id* “01”).
- Com: Compared image data which are acquired by the multispectral camera and received from the AN. 48-bit DCB code (*id* “10”).
- Res: Result of the authentication process. 8-bit integer. (*id* “11”).

• Coefficient: 2 integral parts + 12 fractional parts => 14x4=56 bits

Header 8 bits		Data 56 bits [0,9][0,9].[0,9][0,9][0,9][0,9][0,9][0,9][0,9][0,9][0,9][0,9][0,9][0,9]												Tail 8 bits														
id	p	Int length	1 st integer	2 nd integer	1 st floating	N th floating	Constants						F	F														
71	70	68	68	65	64	-	-	-	-	-	-	-	-	-	-	12	11	10	9	8	7	6	5	4	3	2	1	0

• Original Image/Compared Image: 2 integral parts + 10 fractional parts => 12x4 = 48 bits

Header 8 bits		Data 48 bits [0,9][0,9].[0,9][0,9][0,9][0,9][0,9][0,9][0,9][0,9][0,9][0,9]										Tail 8 bits																
id	p	Int length	1 st integer	2 nd integer	1 st floating	N th floating	Constants						F	F														
63	62	61	60	59	58	57	56	-	-	-	-	-	-	-	-	12	11	10	9	8	7	6	5	4	3	2	1	0

• Result: 8 bits

Header 8 bits		Data 8 bits								Tail 8 bits																		
id	ip	Int length	Result encoding								Constants (FF)																	
23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	1	1	1	1	1	1

Figure 4 – Data forms for each defined type of data.

For the classical point-to-point communication, package-switching is used with 8-bit header and 8-bit tail. The bandwidth of the FIFOs in the communication are different corre-

spond to different length of different multispectral data. The detail of the data form is shown in Fig.4. The structure of the point-to-point communication is shown in Fig.5.

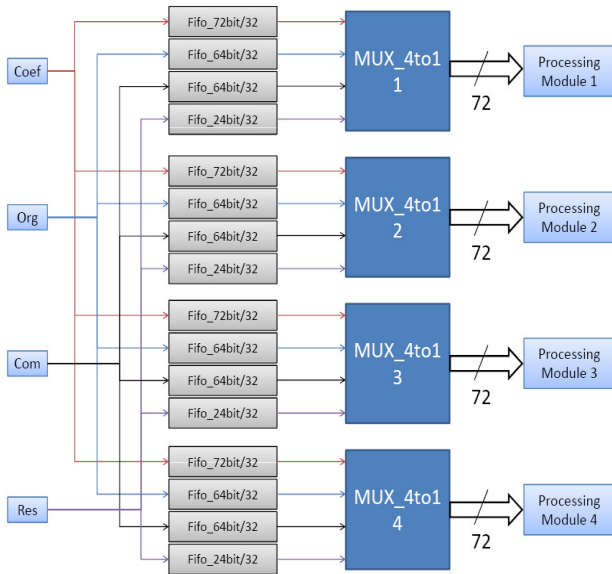


Figure 5 – The structure of the point-to-point communication corresponds to the multispectral image analysis.

5. GALS NOC ON FPGA FOR THE MULTISPECTRAL DATA

An interconnection network is characterized by its topology, routing and flow control. The topology of a network is the arrangement of nodes and channels into a graph. Routing specifies how a packet chooses a path in graph. Flow control deals with the allocation of channel and buffer resources to a packet/data as its traverses in this path. After analysis the characteristics of the multispectral data, the restrictive conditions or the requirement for the communication are:

- Four types of data at the input. Each of them has a different length. The formats of the data must be parameterized in order to make the NoC widely applicable.
- Several PNs as several output nodes. The number of PN must be parameterized according to the application.
- Frequencies of the moduels are not the same. GALS has to be used to adapt the architecture and to improve resource usage.
- For different algorithms, different multispectral data are needed during the calculation, which means different data need to be sent to the same PN at the same time.
- For one type of data, it might be used by different algorithms, which means it is necessary to send one type of data to several PNs and the same time.

5.1 Data communication structure

There are 4 nodes of input which present 4 different types of multispectral data. At first we suppose that there are 4 PNs

in the architecture. This number can be increased to 8 without changing the structure, and theoretically infinite with modification. The architecture has 4 inputs, 4 outputs with one router as shown in Fig.6.

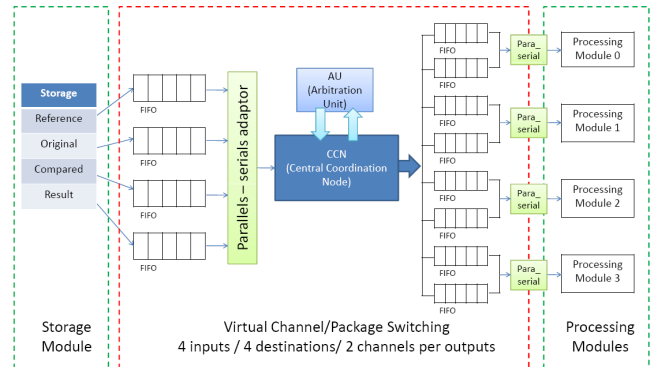


Figure 6 – The structure of the GALS NoC communication.

Compared to the point-to-point communication, the data length in this NoC is always 16 bits for all the types of data to simplify the NA function. The *tail* is changed from “FF” to “F”. This NoC sends all the data as 28 bit packet directly. So the bandwidth of all the FIFOs in VCs is 28 bits. The simplified data form for this version is shown in Fig.7.

Header 8 bits				Data 16 bits [0,15].[0,9][0,9][0,9]												Tail 4 bits													
id	p	No used		1 st integer				2 nd integer				1 st floating				3 rd floating				Constants									
27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	F	F

Figure 7 – The 28-bit packet data form for NoC communication.

5.2 Router structure

This NoC is organized as a centralized router: one node called Central Coordination Node (CCN) performs system coordination functions and one node called Arbitration Unit (AU) detects the states of data paths. The structure of the router is shown in Fig.8.

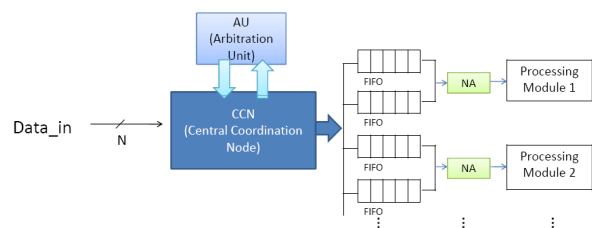


Figure 8 – CCN_AU router structure

The main task of CCN is to manage the system’s resources. It performs mapping of the newly arrived tasks on suitable computation units and inter-task communications to network channels. The structure of CCN is shown in Fig.9.

When one data (Data_in) arrives in CCN, CCN will first verify the destination of this data and send the destination information to AU to “ask” for the routing condition. With the channel states (Vc_state) sent back by the AU, CCN will

choose the available channel (Rst_etat), and send data out (Data_out).

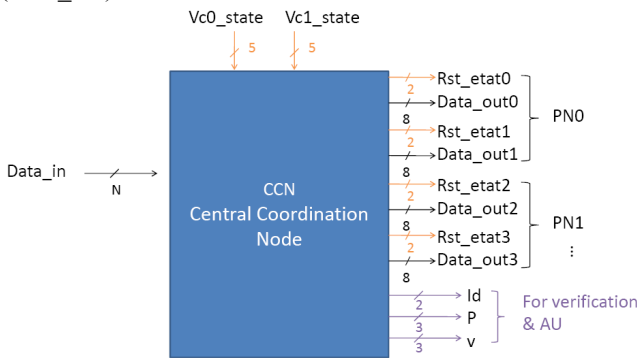


Figure 9 – Central Coordination Node structure

- Vc0_state/Vc1_state: Each output node (PN) has 2 VCs. These two signals present the states of the VCs Detected by Arbitration Unit (AU).
- Rst_etat: VC control signals which choose the available links (FIFO/VC) for the data transmission.

The AU is a Round Robin Arbiter (RRA) [7] which detects the states of all the VC at the outputs. It determines on a cycle-by-cycle basis, which VC may advance. The structure of AU is shown in Fig.10.

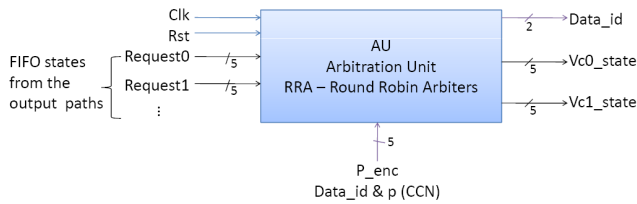


Figure 10 – Arbitration Unit structure

When the AU receives the destination information of the flit (P_enc), it will detect the available paths' states (Request) connected to the aimed output. This routing condition information will be send back to CCN in order to let CCN perform the mapping of the communication.

For the multispectral image algorithms, a large quantity of different kinds of data needs to be sent at the same time. If only one path was placed at each output, the buffer would be blocked due to the contention of data. To solve this problem, a Virtual Channel (VC) flow control is used.

5.3 VC flow control

VC flow control is a well-known technique from the domain of multiprocessor network. A VC consists of a buffer that can hold one or more flits of a packet and associated state information. Several virtual channels may share the bandwidth of a single physical channel [8]. It allows minimization of the size of the router's buffers – a significant source of area and energy over head [9] [10], while providing flexibility and good channel utilization.

During the operation of the router a VC can be in one of the following states presents in the Table 1.

Table 1. FIFO/VC states definition

FIFO states	$Vad_{in} - rst_etat(1)$ (read the data and put in the FIFO) '0' active	$Vad_{out} - rst_etat(0)$ (write the data – send data outside the FIFO) '0' active
Idle 00 VC is not used at the moment	0	0
Busy 11 Busy VC with full FIFO	1 No new data in	1 No data out
Empty 01 Busy VC with empty FIFO	0 No data in	1 Wait for all the data out
Ready 10 Busy VC with nonempty	1 Read new data to the FIFO	0 No data out

After the initialization, all VCs are in the *idle* state. When a new packet arrives on a certain VC, the state of this VC is changed to *busy*. The structure of FIFO/Channel is shown in Fig.11.



Figure 11 – AU structure

The double-clock FIFO has two different clocks. One for reading and the other for writing function. It provides the possibility of construct Time Division Multiplexing (TDM) NoC.

Package switching traffic is forwarded on a per-hop basis, each packet containing routing information as well as data in order to employ for Best Effort (BE) traffic. The routing information contains 2 parts: 8-bit Id, P, Int_1 as a data package header flit and a tail flit. Three fields in header are: a) 2-bit value for id (types of data “00” for Coef, “01” for Org, “10” for Com and “11” for Res); b) 3-bit value for p (output port/PN which means the number of the output port the packet has to go, actually varies from “000” to “011” which presents 4 PNs); c) 3-bit value for integer length which separate the integer part and the floating part of the DCB code. The tail flit is a constant “F”.

The VCs are generated by the CCN during the task mapping stage and provided to the source tile at the tile configuration stage. Each cycle the AU decides which of the ready input VCs may advance

- Conflicts at the input: at each cycle only one VC can advance from an input port.
- Conflicts at the output: at each cycle an output port can accept data from one input VC only.

For many traffic patterns, maximal throughput can only be achieved if fairness is sacrificed. To achieve this problem, more than one router can be implemented in the communication architecture.

6. IMPLEMENTATION RESULTS

6.1 Resource of the other nodes on the architecture

This parameterized TDM architecture was designed as VHDL IP blocks. The used FPGA is Altera Stratix II EP2S15F484C3. It has 6240 ALMs/logic cells. Adaptive Logic Module (ALM) is the basic building block of Stratix II devices which contains registers, adders, combinational logic, etc. It can maximize efficiency and performance of the implementation. The resource which is taken by CN, AN and SN is around 14% of the total logic cells (see Table 2).

Table 2. Resources for the Nodes in the GALS Architecture

Node	Freq (MHz)	Resources on StratixII 2S60		
		Logic cells	Registers	Mem bits
Control	150	278	265	32
Acquisition	76.923	315	226	2
Storage	100	280	424	320000
Processing	50	Depending the algorithms		

6.2 Resource of point-to-point communication

Point-to-point communication needs over 510 Input Output Blocks (IOB) pins which is overload compared to the total 339 IOBs available on FPGA Altera StratixII EP2S15F484C3 (6240 ALMs). To verify the total resources, one superior card Altera StratixII EP2S180F1508C3 is used (see Table.3). The frequency in the table for NoC Version 1 mentions only the interconnection part inside the NoC communication, not the global frequency in the DSE study.

Table 3. Comparison of the resources: Point-to-point VS. NoC Version 1

Version	Point-to-Point	Version 1 48-bit	Version 1 56-bit
Logic Utilization %	1%	2%	3%
Combinational ALUTs/%	305	1842	2118
Dedicated logic registers/%	1425	2347	2739
Total pins/%	512	344	408
Total block memory bit/%	29568	3384	3960
Frequency for F (MHz)	165.73MHz	264.34MHz	282.41MHz

For the NoC version 1, "48-bit" and "56-bit" present 2 different bandwidths. As shown in Table 3, the point-to-point communication needs less ALUTs but over 7 times more block memory bits. Furthermore, the speed of Point-to-point communication is much slower than any NoC versions. The request of the data interconnection, large quantity of registers and block memory bit is required by the storage nodes. The other hand, ALUTs is quite enough for the architecture implementation. In this condition, the NoC version is more preferable since its less register requirement and faster frequency. Furthermore, the total IOB pins which is needed by

the two kinds of NoC version 1 is much less than the point-to-point communication. It signifies NoC version can be implemented on a much smaller FPGA.

7. CONCLUSION & PERSPECTIVES

The proposed NoC is a parameterized TDM architecture which is speed, flexible and adaptive to the multispectral image analysis applications. BFT topology with VC packet-switching is used. With help of the designed NA in Version 2, data are firstly transformed as one completed packet then cut as 8 bits flit. The size of data is dynamic which is required by the multispectral algorithms.

Future work will focus on generating the completed architecture and analysing the matching algorithm architecture according to the different required data.

REFERENCES

- [1] J. Muttersbach, T. Villiger, W. Fichtner, "Practical design of globally-asynchronous locally-synchronous systems," in Proc. of *IEEE* the 6th Int. Symp. on Advanced Research in Asynchronous Circuits & Systems, pp. 52-59. April 2000.
- [2] M. Keating, P. Bricaud "Reuse Methodology manual for System-On-Chip Designs," Kluwer Academic Publisher, 1998.
- [3] M. Forsell, "A scalable high-performance computing solution for networks on chips," *IEEE Micro*, 22, 5, 2002.
- [4] V. Fresse, A. Aubert, N. Bochard "A predictive NoC architecture for vision systems dedicated to image analysis," *EURASIP Journal on Embedded Systems* Volume, Article ID 97929, 13 pages, 2007.
- [5] L. Zhang, A.-C. Legrand, V. Fresse, V. Fischer, "Adaptive FPGA NoC-based Architecture for Multispectral Image Correlation," in Proc. of *IS&T CGIV&MCS08*, pp.451-456. Terrassa, Barcelona, Spain, June 9-13, 2008.
- [6] A. Lines, "Nexus: An asynchronous crossbar interconnects for synchronous system-on-chip designs," in Proc of *IEEE* 11th Symposium on High Performance Interconnects, HOTI'03, pp. 2-9. 20-22 Aug. 2003.
- [7] P. Gupta, N. McKeown, "Designing and implementing of a Fast crossbar scheduler," in *IEEE Micro* Jan/Feb 1999, vol.19, Issue.1, pp.20-28,1999.
- [8] W. J. Dally, "Virtual-Channel flow control," in *IEEE Parallel and Distributed Systems*, vol.3, Issue.2, pp.194-205, 1992.
- [9] E. Rijpkema, K. G. W. Goossens, A. Rădulescu, J. Dielissen, J. Van Meerbergen, P. Wielage, E. Waterl, "Trade offs in the design of a router with both guaranteed and best-effort services for networks on chip," in *IEEE Proc. Computers and Digital Techniques*, Vol.150, Issue.5, pp.294-302, 2003.
- [10] H. S. Wang, L.-S. Peh, S. Malik, "A power model for routers: Modeling Alpha 21364 and InfiniBand Routers," in Proc 10th High Performance Interconnects 2002, pp.21-27, 2002.