Karthik Narayanan, Santosh Madiraju

EEL6935 - Embedded Systems Seminar

Topic: Communication

31st Jan, 2013.

Communication Architectures for Dynamically Reconfigurable FPGA Designs .

Pionteck, T., Albrecht, C. ; Koch, R. ; Maehle, E. ; Hubner, M. ; Becker, J. Parallel and Distributed Processing Symposium, 2007. IPDPS 2007. IEEE International Page(s): 1 - 8 26-30 March 2007.

Dynamic Reconfiguration.

- Dynamic Reconfiguration – What does it mean?
  - Enables a device to change its configuration during normal system execution.
  - Different combination of hardware modules impose different demands on bandwidth and latency of communication network
- Suitable approach is an ad-hoc flexible communication infrastructure

- Classification of Communication architectures.
  - Bus based
  - Network on chip
- Each structure has its advantages and disadvantages.

Constraints:

- Performance Parameters:
  - Latency:
    - Network element latency
    - Path latency
  - Bandwidth
    - Data delivered per unit time.
  - Throughput
    - Sum of data rates delivered by all the links.
  - Parallelism
    - # simultaneous data transfers.

Constraints:

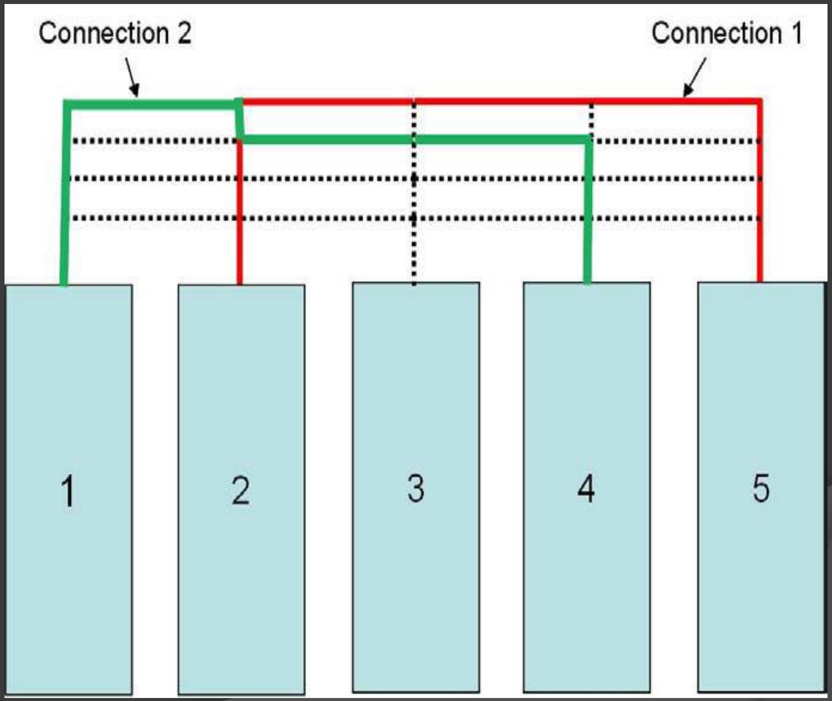
- Design Parameters:
  - Flexibility
    - Support different communication patterns without loss of performance.
  - Scalability
    - Fixed performance irrespective of system size and characteristics
  - Extensibility
    - Ability to extend to larger designs.
  - Modularity
    - Ability to segment into sub modules.

Bus Based: Reconfigurable Multiple Bus on Chip(RMBoc)

- Bus based architectures- first ones developed for dynamically reconfigurable FPGA designs.
- Parallel bus segments between processing nodes and a cross point per processing node.
- Communicating via sending requests over bus system.

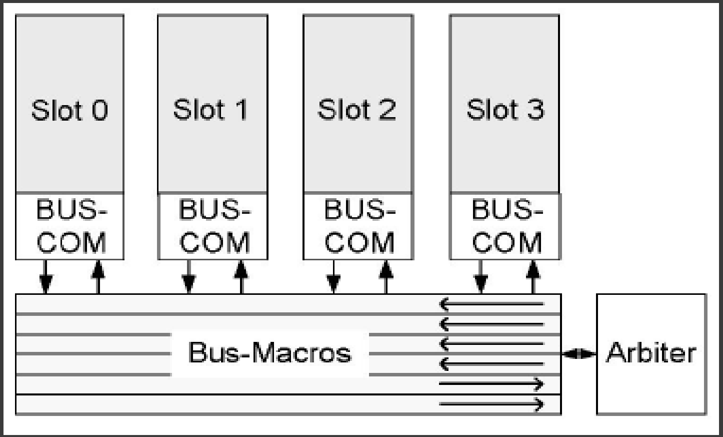
RMBoc

- New channels of communication are located at the top segments.
- Working:
  - Request
  - Reply
  - Cancel
  - Destroy



Bus Based: BUS-COM

- Uses un-segmented buses assigned to hardware modules by an arbiter.
- Hardware modules connected to bus system via a BUS-COM interface.
- Each bus has 32 time slots and virtual network topology can be created using them.
  - Static time slots – guaranteed bus access time.
  - Dynamic time slots – additional access time.

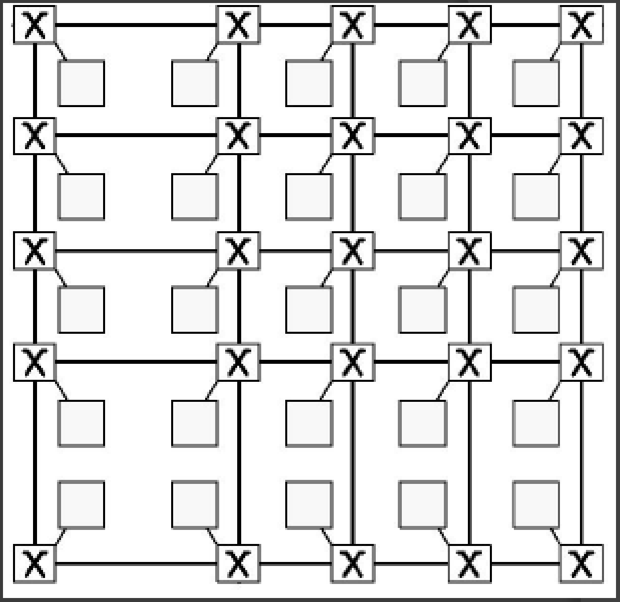


NoC based: Dynamic Network on Chip (DyNoC)

- 2D array of PE's
  and routers, each PE is
  connected to one router.

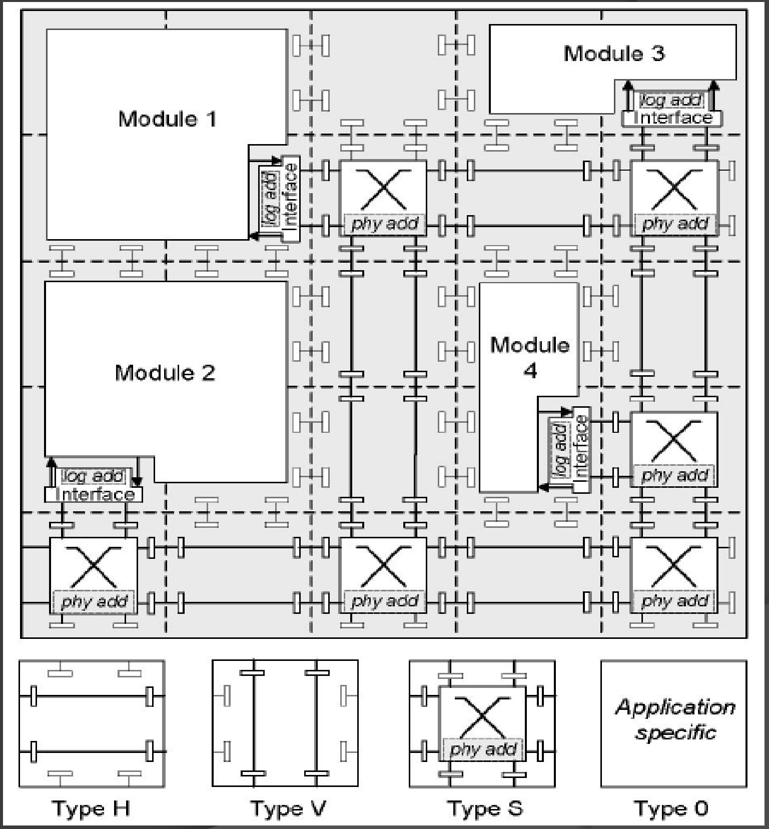
- Hardware modules can
  be mapped onto multiple PE's.

- Routers between PE's used for one module can be used as additional hardware resources.



NoC based: Configurable Network on Chip (CoNoChi)

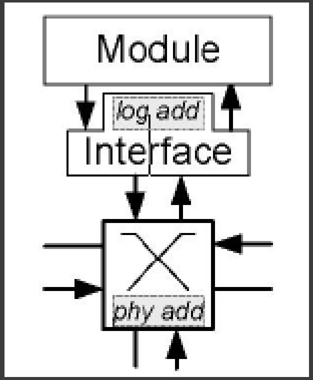
- Virtual cut through switches with four equal full duplex links.
- Abstraction – i*j grid of tiles of type O,S,H,V
  - O – Module, S – Switch
  - H,V – Horizontal and Vertical links.
- Runtime adaptation by replacing the appropriate tiles.
- Switches can be added or removed by global control unit without stalling NoC.

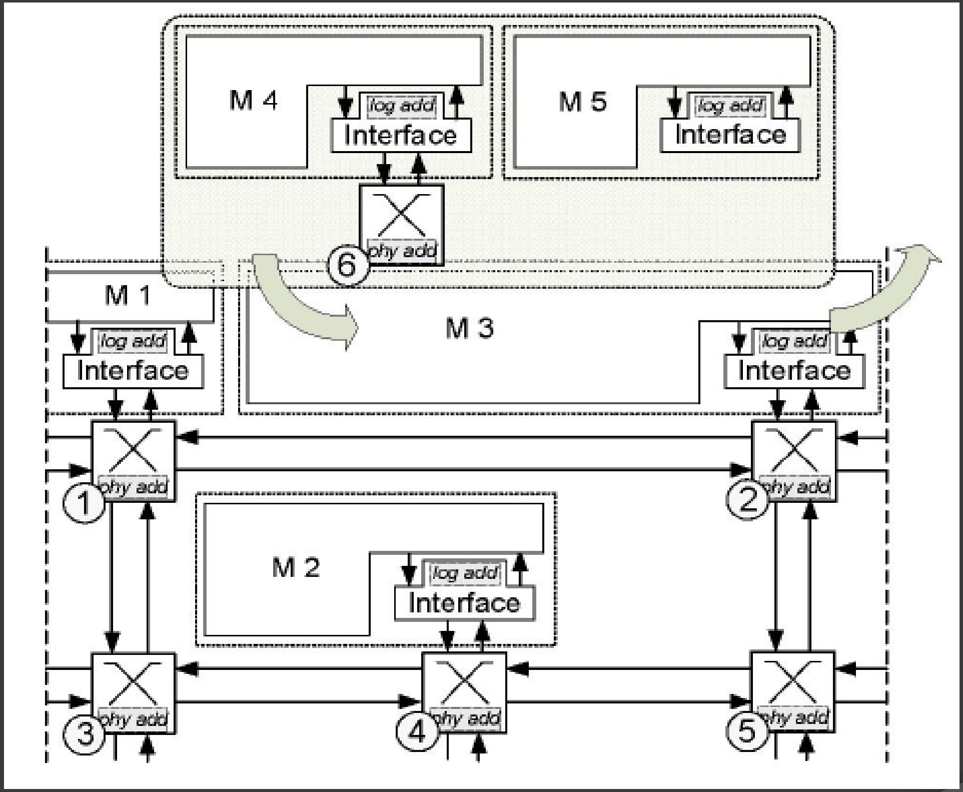


CoNoChi

- Protocol contains physical and logical addresses.
- Routing to hardware modules based on physical addresses.
- Interfaces used to connect switches to hardware modules use the logical addresses to identify the processing entity.

Working:


Module
log add
Interface
phy add


M 4
log add
Interface
M 5
log add
Interface
6
phy add
M 1
log add
Interface
M 3
log add
Interface
1
phy add
2
phy add
M 2
log add
Interface
3
phy add
4
phy add
5
phy add

Technical aspects:

- All architectures provide sufficient support for dynamically reconfigurable FPGA designs.
- Minimal area requirements for connecting four modules using 32 bit link.

RMBoc	BUS-COM	DyNoC	CoNoChi
5084	294	1480	1640

- Bus based vs. NoC based.
  - Shape
    - Bus based – restrict shape of modules
    - NoC based – arbitrary rectangular shaped modules.
  - Area
    - CoNoChi and BUS-COM do not include area of control units.
    - RMBoc alone includes all resources.
    - Larger number of modules/ larger modules require more switches. So more area over head.

Observation – Performance Parameters:

- Frequency and bandwidth do not make much difference in the architectures.

- Latency:
  - Bus based – latency ~ 1
  - NoC based – scales with number of switches.
    - CoNoChi – number of switches depend on number of hardware modules
    - DyNoC – Number of switches also depend on size of hardware modules

- Maximmum Parallelism:
  - Bus based - Providing segmented buses and parallel busses.
  - NoC – packet switching, so communication channel not established exclusively.
    - Theoritically limited by number of links.

Observation: Structural Parameters:

Architecture	Flexibility	Medium Scalability	Extensibility	Modularity
RMBoC	High	Medium	Low	Medium
BUS-COM		Medium	Medium	Medium
DyNoC	Low	High	High	High
CoNoChi	High	High	High	High

Structural Parameters:

- Scalability:
  - Bus based limited by number of buses and parallel data transfers.
- Extensibility:
  - Bus based limits extensibility
  - NoC – components can be added at the border.
- Modularity:
  - NoC – Can have modules of varying rectangular size.
- Flexibility:
  - CoNoChi – flexible because of distributed routing tables.
  - Next is BUS-COM because of virtual network topology

Conclusion:

- Overview of communication systems in runtime reconfigurable systems.
- Performance Parameters and Structural Parameters to be considerd when selecting the appropriate one.
- RMBoC is slightly superior to BUS-COM because of segmented buses.
- CoNiChi offers best structural parameters and best conceptual support for dynamic reconfiguration.

A Light-Weight Network-on-Chip Architecture for Dynamically Reconfigurable Systems.

Corbetta, S., Rana, V. ; Santambrogio, M.D. ; Sciuto, D.

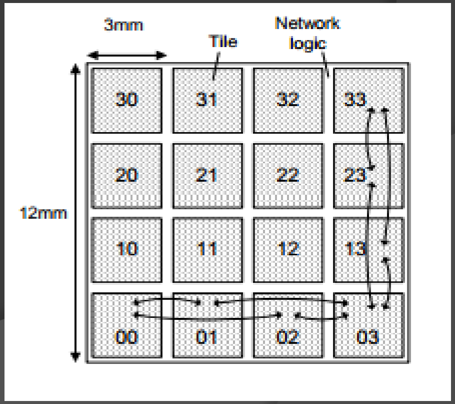
Embedded Computer Systems: Architectures, Modelling, and Simulation, 2008. SAMOS 2008. International Conference on Page(s): 49- 56 21-24 July 2008.

Introduction

- Light-weight Noc is a novel communication infrastructure which keeps up with the dynamic changes of the application requirements.
- Tile based Network on Chip approach

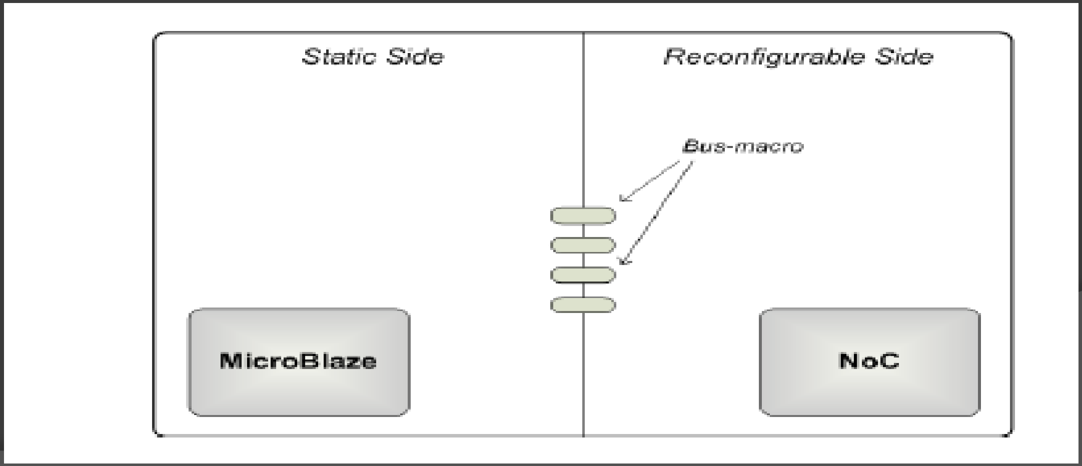
  Communication layer is entirely decoupled from Computation layer

  Supports dynamic reconfiguration at the communication fabric level



Architecture:

- Communication infrastructure deployed on an ad-hoc reconfigurable architecture.
- Architecture model is composed of 2 parts:
  - Static Architecture : E.g. processor , I/O .
  - Reconfigurable Architecture
  - A special Hardware Bus Macro is employed to route signal between the Static side and the Reconfigurable side.



Network Based Approach

- Communication infrastructure will be based on the use of basic network elements.
- Packet Switching is used instead of Circuit Hardwired Connections.
  - Interface design complexity of the cores is reduced.
  - Possibility to tailor the protocol gives flexibility to extend the design and also define application driven communication schemes.
  - Enables the network to define Fault tolerant schemes at both levels .
  - Parallelism and efficiency are increased.

  - A Packet switched system overcomes flexibility and scalability limitations imposed by point to point systems .

Network Based Approach

- The entire interconnection is based on a single element : Switch
- Switch can be seen as a high level black box which forwards packets based on the destination.
- Has basic I/O and computational elements, used to buffer requests and forward data.
- The network guarantees a predefined QoS based on the topology.

Light Weight Infrastructure:

- Strict Physical Constraints, performance ,power consumption and resource utilization are some of the demands.
- Qualities which Define the Light Weigh Architecture:
  - Interface design
  - Communication among switches and cores
  - Switching and Routing Components
  - Inner Switch Connection
  - Reconfiguration task

Light Weight Approach

- Dynamic Reconfiguration must not affect the rest of the circuits functioning.
- The reconfiguration process is directly proportional to the bit stream size.
- The bit stream size is dependent on the dimensions of the circuit thus leading to a light weight network approach.

Layered Approach

- The computational and Communicational layers are completely decoupled.
- The logic space used to assemble and disassemble data into packets is also decoupled.
- Packets are formed with the help of network interface modules
- Data is forwarded by the switches on the network

Reliable Communication

- Reliable Communication achieved by two aspects :

  - Persistent Communication among the switches
  - Redundant communication path among each communicating core

- Persistent Communication addresses the switching network which needs to be minimalist w.r.t resources
  - E.g. A single bit can be used to denote congestions in a switch eliminating the need for a central system controller.
  - Topology or interconnections could thus be changed dynamically.

- Routing can be chosen at run time either by dynamically reconfiguring routing tables or by providing added features to the nodes.

Dynamic Features

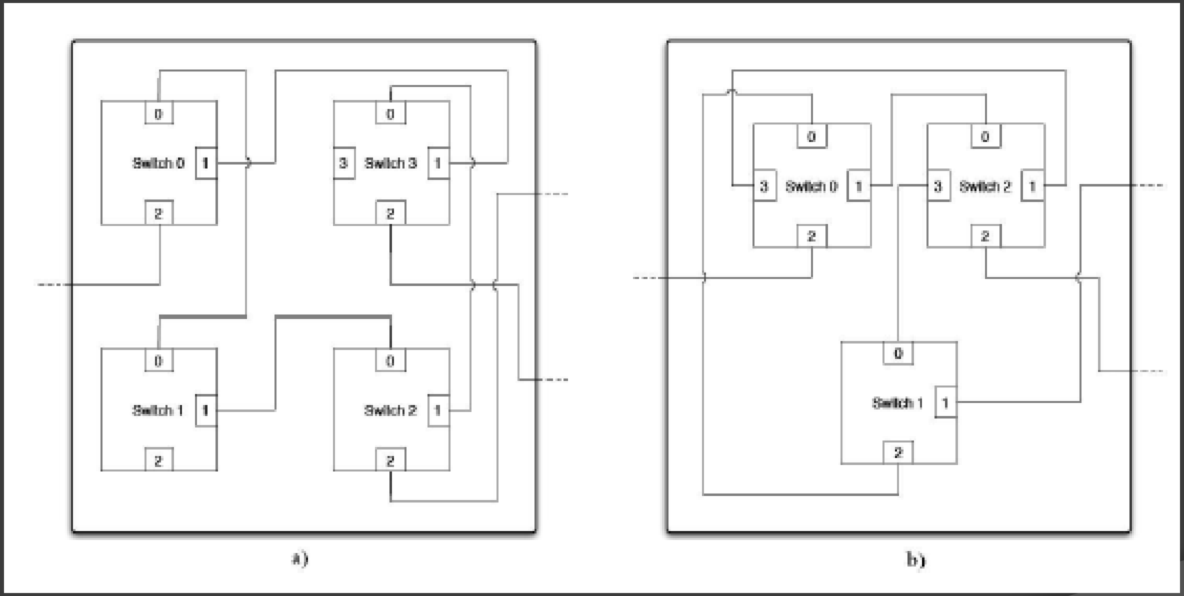
- Fault tolerant communication is implemented by controlling the traffic and changing the routing path.
- Routing scheme followed is simple
- The routing information can be stored in 2 ways
  - LUT-based storage: Routing information is hard coded in the component.
    - Lacks Flexibility and Efficiency
  - Memory based storage: Routing information is stored in the BRAM blocks in the FPGA
    - More flexible and can be easily reconfigured
    - Routing information is completely decoupled with reconfigurable logic
    - Implemented by either using microprocessor to change the content of the memory   or by a new bit file.

Experimental Setup:

- The network is generated on a Xilinx Vertex-II Pro FPGA hosted on a VP-20 development board from Avnet.
- The board is divided into a static and a reconfigurable module.
- A Micro Blaze soft-core processor is used to execute the software.
- The processor is connected through the OPB to other peripherals.
- An initiator and a target are connected to the network.

Experimental Setup:

- Two different topologies are loaded
- The simple app executed reads and writes data from and to specific addresses.
- The output is read from the serial port.



Results

Bit Stream Size (Hardware and Software)

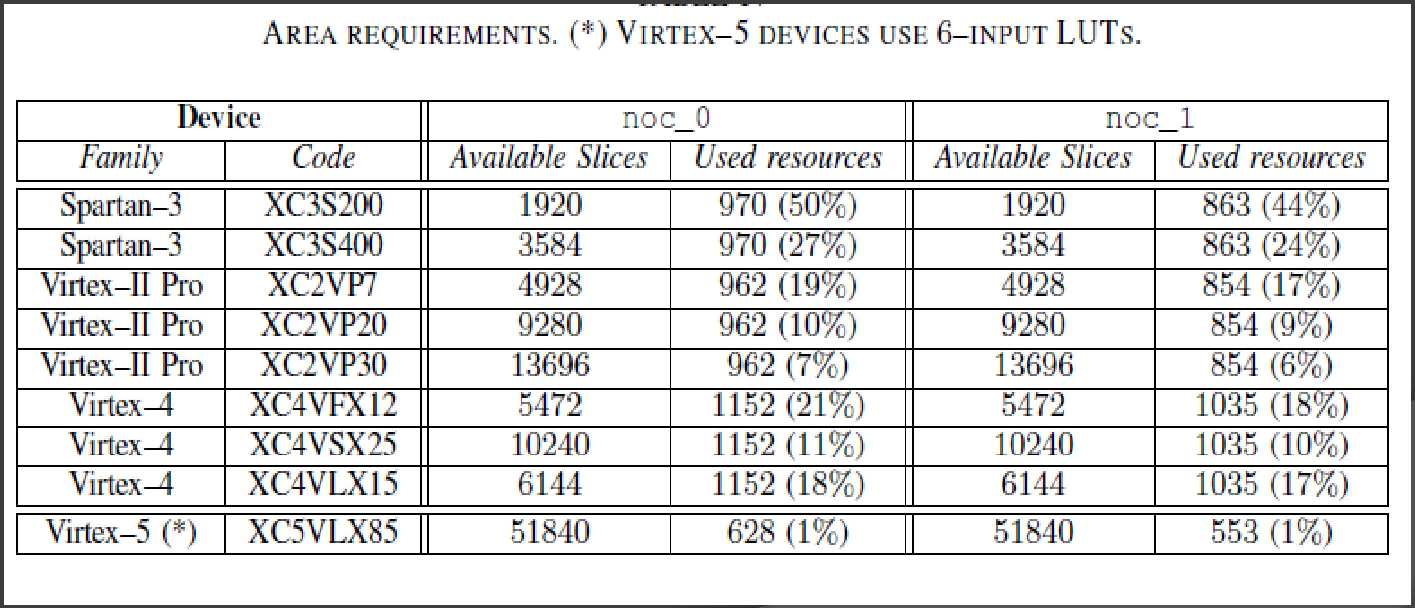
Bit Stream	Type	Size
Static side and noc_0	Complete	1.003KB
Static side and noc_1	Complete	1.003KB
Noc_0	Partial	227KB
Noc_1	Partial	227KB

Reconfiguration Time of the Network

Starting Configuration	Final Configuration	Reconfiguration time
Noc_0	Noc_1	222ms
Noc_1	Noc_0	222ms

Results:

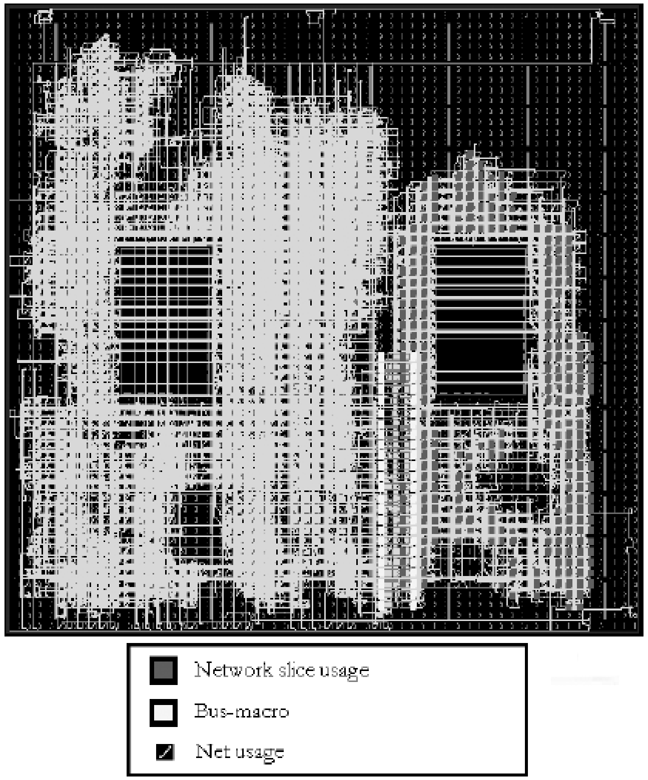
- The following table shows the area requirements of the two network topologies for different Xilinx FPGA devices.
- The network noc_0 is made of 4 switches where as the network noc_1 is made of 3.

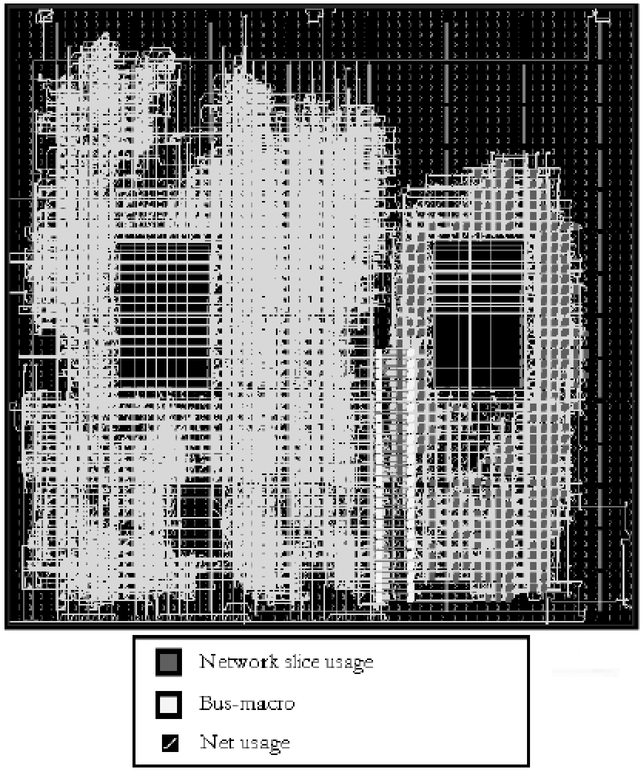


AREA REQUIREMENTS. (*) VIRTEX–5 DEVICES USE 6–INPUT LUTS.

Device		noc_0		noc_1	
Family	*Code*	*Available Slices*	*Used resources*	*Available Slices*	*Used resources*
Spartan–3	XC3S200	1920	970 (50%)	1920	863 (44%)
Spartan–3	XC3S400	3584	970 (27%)	3584	863 (24%)
Virtex–II Pro	XC2VP7	4928	962 (19%)	4928	854 (17%)
Virtex–II Pro	XC2VP20	9280	962 (10%)	9280	854 (9%)
Virtex–II Pro	XC2VP30	13696	962 (7%)	13696	854 (6%)
Virtex–4	XC4VFX12	5472	1152 (21%)	5472	1035 (18%)
Virtex–4	XC4VSX25	10240	1152 (11%)	10240	1035 (10%)
Virtex–4	XC4VLX15	6144	1152 (18%)	6144	1035 (17%)
Virtex–5 (*)	XC5VLX85	51840	628 (1%)	51840	553 (1%)

Implementation of the Architectures





Conclusion

- Network on chip design of communication architecture is a suitable approach which guarantees the flexibility and adaptability for Dynamically Reconfigurable Networks.
- By decoupling the communication and computation layers, it is easier to reconfigure the underlying network without affecting the performance of the entire system.
- A Light weight network is essential to satisfy the constraints posed by dynamic reconfiguration.
- A dynamic routing is realized, in which routing information relies on the current network status.

Future Work and Issues with the Research

- Improve the capabilities of the network
- Fault Tolerance is yet to be achieved with respect to application execution.
- The research is aimed at only embedded applications and is to be seen if it can be applicable for large scale systems.
- How performance varies with varying topologies and networks is not addressed.
- Power consumption was not addressed.