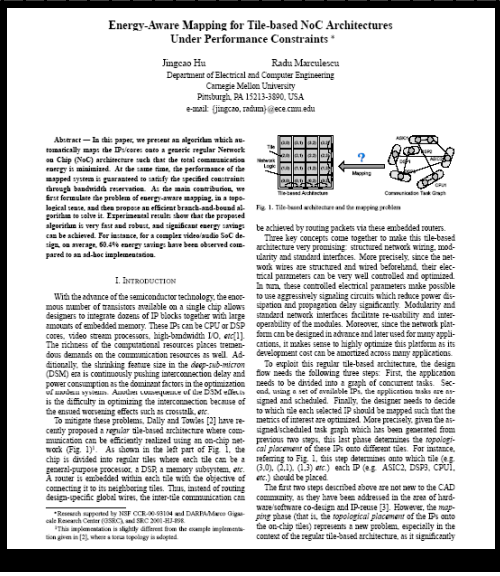
Mapping Applications onto the Chip

Midia Reshadi, Ph.D
CE Department of
Science and Research Branch
Islamic Azad University
(SRBIAU)
reshadi@iees.org
http://m-reshadi.net

Agenda

- First step to the application specific Ics
- Partially Branch & Bound mapping
- Nmap

1st References



Energy-Aware Mapping for Tile-based NoC Architectures Under Performance Constraints *

Jingcao Hu Radu Marculescu
Department of Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, PA 15213-3890, USA
e-mail: {jingcao, radum}@ece.cmu.edu

Abstract — In this paper, we present an algorithm which automatically maps the IP cores onto a generic regular Network on Chip (NoC) architecture such that the total communication energy is minimized. At the same time, the performance of the mapped system is guaranteed to satisfy the specified constraints through bandwidth reservation. As the main contribution, we first formulate the problem of energy-aware mapping, in a topological sense, and then propose an efficient branch-and-bound algorithm to solve it. Experimental results show that the proposed algorithm is very fast and robust, and significant energy savings can be achieved. For instance, for a complex video/audio SoC design, on average, 60.4% energy savings have been observed compared to an ad-hoc implementation.

I. Introduction

With the advance of the semiconductor technology, the enormous number of transistors available on a single chip allows designers to integrate dozens of IP blocks together with large amounts of embedded memory. These IPs can be CPU or DSP cores, video stream processors, high-bandwidth I/O, *etc*[1]. The richness of the computational resources places tremendous demands on the communication resources as well. Additionally, the shrinking feature size in the *deep-sub-micron* (DSM) era is continuously pushing interconnection delay and power consumption to the dominant factors in the optimization of modern systems. Another consequence of the DSM effects is the difficulty in optimizing the interconnection because of the ensued worsening effects such as crosstalk, *etc*.

To mitigate these problems, Dally and Towles [2] have recently proposed a regular tile-based architecture where communication can be efficiently realized using an on-chip network (Fig. 1)[1]. As shown in the left part of Fig. 1, the chip is divided into regular tiles where each tile can be a general-purpose processor, a DSP, a memory subsystem, *etc*. A router is embedded within each tile with the objective of connecting it to its neighboring tiles. Thus, instead of routing design-specific global wires, the inter-tile communication can be achieved by routing packets via these embedded routers.

*Research supported by NSF CCR-00-93104 and DARPA/Marco Gigascale Research Center (GSRC), and SRC 2001-HJ-898.

[1] This implementation is slightly different from the example implementation given in [2], where a torus topology is adopted.

Fig. 1. Tile-based architecture and the mapping problem

Three key concepts come together to make this tile-based architecture very promising: structured network wiring, modularity and standard interfaces. More precisely, since the network wires are structured and wired beforehand, their electrical parameters can be very well controlled and optimized. In turn, these controlled electrical parameters make possible to use aggressively signaling circuits which reduce power dissipation and propagation delay significantly. Modularity and standard network interfaces facilitate re-usability and interoperability of the modules. Moreover, since the network platform can be designed in advance and later used for many applications, it makes sense to highly optimize this platform as its development cost can be amortized across many applications.

To exploit this regular tile-based architecture, the design flow needs the following three steps: First, the application needs to be divided into a graph of concurrent tasks. Second, using a set of available IPs, the application tasks are assigned and scheduled. Finally, the designer needs to decide to which tile each selected IP should be mapped such that the metrics of interest are optimized. More precisely, given the assigned/scheduled task graph which has been generated from previous two steps, this last phase determines the *topological placement* of these IPs onto different tiles. For instance, referring to Fig. 1, this step determines onto which tile (e.g. (3,0), (2,1), (1,2) *etc*.) each IP (e.g. ASIC2, DSP3, CPU1, *etc*.) should be placed.

The first two steps described above are not new to the CAD community, as they have been addressed in the area of hardware/software co-design and IP-reuse [3]. However, the *mapping* phase (that is, the *topological placement* of the IPs onto the on-chip tiles) represents a new problem, especially in the context of the regular tile-based architecture, as it significantly

1st References

- **Energy-Aware Mapping for Tile-based NoC Architectures Under Performance Constraints**
  - **Authors :** Jingcao Hu, Radu Marculescu
    - Won Best paper award
    - And
    - **Most Influential Papers of 10 Years in DATE Conference**

Citation

- 250 Citation !!!

Energy-aware mapping for **tile-based NoC architectures** under performance constraints
J Hu, R Marculescu - … , 2003. Proceedings of the ASP-DAC …, 2003 - ieeexplore.ieee.org
Abstract - In this paper, we present an algorithm which au- tomatically maps the IPdcores onto a generic regular Network on Chip (**NoC**) **architecture** such that the total communication energy is minimized. At the same time, the performance of the mapped system is guaranteed to ...
Cited by 250 - Related articles - All 23 versions

Cited by 250

2st reference

Bandwidth-Constrained Mapping of Cores onto NoC Architectures

Srinivasan Murali, Giovanni De Micheli
Computer Systems Lab
Stanford University
Stanford, California 94305
{smurali, nanni}@stanford.edu

Abstract

We address the design of complex monolithic systems, where processing cores generate and consume a varying and large amount of data, thus bringing the communication links to the edge of congestion. Typical applications are in the area of multi-media processing. We consider a mesh-based Network on Chip (NoC) architecture, and we explore the assignment of cores to mesh cross-points so that the traffic on links satisfies bandwidth constraints. A single-path deterministic routing between the cores places high bandwidth demands on the links. The bandwidth requirements can be significantly reduced by splitting the traffic between the cores across multiple paths. In this paper, we present NMAP, a fast algorithm that maps the cores onto a mesh NoC architecture under bandwidth constraints, minimizing the average communication delay. The NMAP algorithm is presented for both single minimum-path routing and split-traffic routing. The algorithm is applied to a benchmark DSP design and the resulting NoC is built and simulated at cycle accurate level in SystemC using macros from the ×pipes library. Also, experiments with six video processing applications show significant savings in bandwidth and communication cost for NMAP algorithm when compared to existing algorithms.

Keywords: Systems on Chips, Networks on Chips, cores, mapping, bandwidth, routing.

1 Introduction

Present and future *Systems on Chip* (SoC) are designed using preexisting components such as processors, DSPs, memory arrays [1], which we call *cores*. The use of standard hardwired busses to interconnect these cores is not scalable. To overcome this problem, *Networks on Chips* (NoCs) have been proposed and used for interconnecting the cores [2, 3, 4] and replacing dumb physical routing. The use of on-chip interconnection network has several advantages, including better structure, performance and modularity.

In several application domains, such as multi-media processing, the bandwidth requirement between the cores in

Figure 1. Block diagram of Video Object Plane Decoder, with communication BW (in MB/s).

SoCs is increasing. The aggregate communication bandwidth between the cores is in the GBytes/s range for many video applications. In the future, with the integration of many applications onto a single device and with increased processing speed of cores, the bandwidth demands will scale up to much larger values [3]. As an example of a media processing application, a Video Object Plane decoder [7] is shown in Figure 1. Each block in the figure corresponds to a core and the edges connecting the cores are labeled with bandwidth demands of the communication between them. As seen from the figure, the bandwidth demands are in the order of hundreds of MBytes/s.

Networks on Chip can be designed in different ways, according to the network architecture and protocol chosen. In this paper, we limit our considerations to *mesh/torus* networks and to *packet-switched* data transmission. Note that our techniques are not limited to mesh topologies, but we stick to this restriction to be more specific in this paper. Packet switching leads to better link utilization. Nevertheless, packet switching with single-path deterministic routing places high bandwidth demands on the network links. By allocating higher bandwidth across the links of the NoC, more energy is dissipated. Thus, it is important to balance the bandwidth needs across the different links. In this paper we apply packet switching in both single and multi-path routing, where the traffic between two end-nodes is split across many paths.

1530-1591/04 $20.00 (c) 2004 IEEE

2st reference

- **Bandwidth-Constrained Mapping of Cores onto NoC Architectures**
  - **Authors:** Srinivasan Murali, Giovanni De Micheli

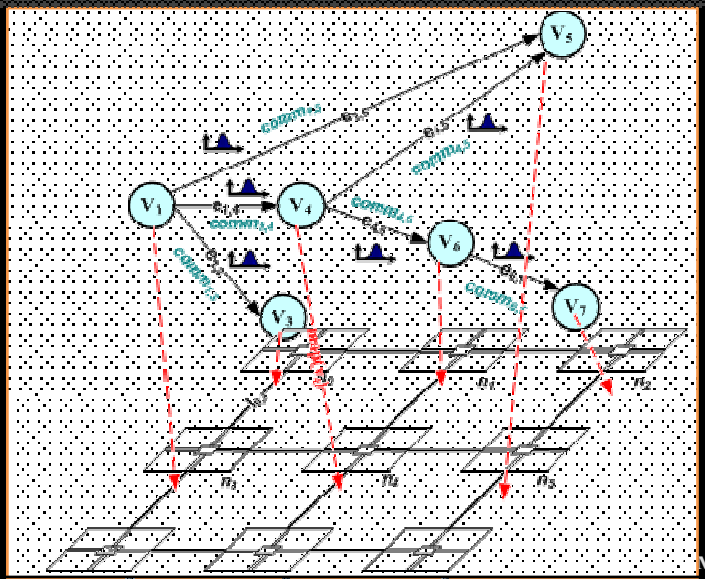
Citation

- 293 Citation !!!!!

Bandwidth-constrained **mapping** of cores onto **NoC architectures**
S Murali, G De Micheli - 2004 - computer.org
... ments on the links of a **NoC** are known for a particular ap- plication, thus the bandwidth constraints in the **NoC** archi- tecture need to be satisfied by the **mapping**. In [8], a branch and bound algorithm is proposed that maps cores onto a **tile-based NoC architecture** satisfying the ...
Cited by 293 - Related articles - All 19 versions

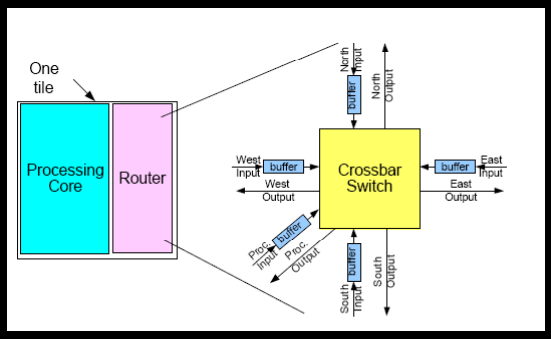
Cited by 293

Mapping Basics



Midia Reshadi

Basic Definition:: tile



© Midia Reshadi

The Energy Model

- bit energy (E_{bit}) metric: for a router
  - T. T. Ye, L. Benini, G. De Micheli, "Analysis of power consumption on switch fabrics in network routers," *Proc. DAC, June 2002.*
- E_{bit}:

$$E_{bit} = E_{Sbit} + E_{Bbit} + E_{Wbit}$$

  - E_{Sbit} → The energy consumed by switch
  - E_{Bbit} → buffering
  - E_{wbit} → interconnection wires
    - It is assumed that the buffers are implemented in SRAM or DRAM

The Energy Model

- Extending the router energy model to the tile-based network
- Note:
  - E_{Bbit} becomes dominant when congestion happens since accessing and refreshing the memory are very expensive in terms of power consumption.
- E_{Wbit} : The energy consumed on the wires inside the switch fabric.

The Energy Model

- E_{Lbit}: Energy consumed on the links
- **The average energy consumed in sending one bit of data from a tile to its neighboring tile**
  - $E_{bit} = E_{Sbit} + E_{Bbit} + E_{Wbit} + E_{Lbit}$
- Link length is typically in the order of *mm*,
  - $E_{Bbit} + E_{Wbit} << E_{Lbit}$
- so:
  - $E_{bit} = E_{Sbit} + E_{Lbit}$

The Energy Model

- Average energy consumption of sending one bit of data from tile t_i to tile t_j :

$$E_{bit}^{t_i,t_j} = n_{hops} \times E_{S_{bit}} + (n_{hops} - 1) \times E_{L_{bit}}$$

  - n_{hops}: number of hops

Problem Formulation

- Objective
  - Mapping IPs onto different tiles such that
    - The total communication energy consumption is minimized
    - Guaranteeing the performance of the system

Problem Formulation

- **Definition 1:**
- **An *Application Characterization Graph (APCG)***
- *G*= G(C,A): A *directed graph*
  - c_i *represents* one selected IP/core
  - $a_{i,j}$ represents the communication from c_i to c_j .
- Associated with each $a_{i,j}$ as arc properties:
  - $v(a_{i,j})$: The communication volume (*bits)* from c_i to c_j
  - $b(a_{i,j})$: The bandwidth requirement from vertex c_i to c_j ,
    - The minimum bandwidth (*bits/sec.) that* should be allocated by the communication network.

Problem Formulation

- **Definition 2**
- **An *Architecture Characterization Graph (ARCG)***
- *G'* = G(T, P): A *directed graph*
  - Each vertex t_i represents one tile in the architecture
  - Each directed arc $p_{i,j}$ represents the routing path from t_i to t_j
- Associated with each $p_{i,j}$ as arc properties:
  - $e(p_{i,j})$: The average energy consumption (*joule) of sending one bit of* data from tile t_i to t_j → E^{t_i,t_j}
  - $L(p_{i,j})$: the set of links that make up the path $p_{i,j}$

Problem Formulation

- Minimizing the communication energy consumption under performance constraints:
- ***Given*** an APCG and an ARCG that satisfy:

$$size(APCG) \leq size(ARCG)$$

- ***Find*** a mapping function map() from APCG to ARCG which minimizes:

$$\min\{Energy = \sum_{\forall a_{i,j}} v(a_{i,j}) \times e(p_{map(c_i),map(c_j)})\}$$

Problem Formulation

- such that:

$$\forall c_i \in C, \quad map(c_i) \in T$$
$$\forall c_i \neq c_j \in C, \quad map(c_i) \neq map(c_j)$$
$$\forall \text{ link } l_k, B(l_k) \geq \sum_{\forall a_{i,j}} b(a_{i,j}) \times f(l_k, p_{map(c_i),map(c_j)})$$

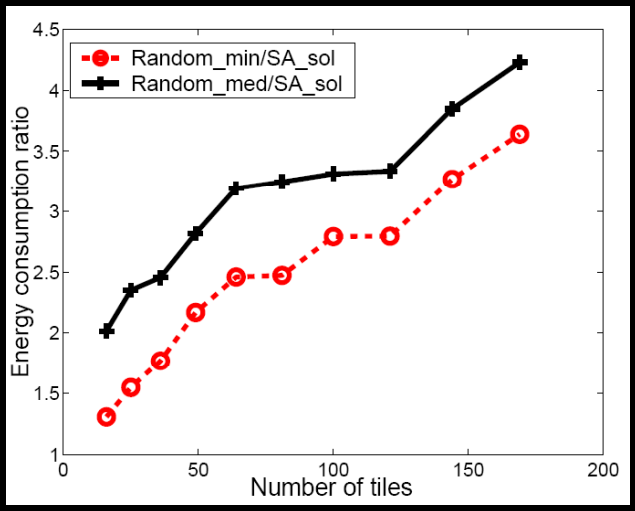
- Where $B(l_k)$ is the bandwidth of link l_k :

$$f(l_k, p_{m,n}) = \begin{cases} 0 & : \quad l_k \notin L(p_{m,n}) \\ 1 & : \quad l_k \in L(p_{m,n}) \end{cases}$$

Significance of the Problem

- Scenario:
  - The number of IPs: 3x3 to 13x13
  - 3000 random generated mapping for each benchmark
  - Optimizing method: Simulated Annealing (SA)
- Annotations:
  - Random_min/SA_sol:
    - Minimum energy for random mapping/Simulated annealing
  - Random_med/SA_sol:
    - Average energy for random mapping/Simulated annealing

Significance of the Problem

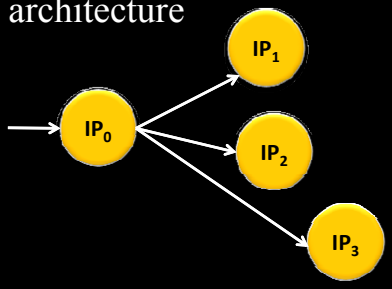


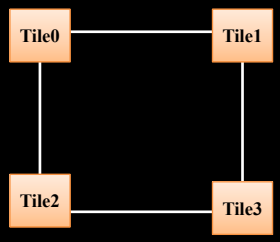
Significance of the Problem

- Unfortunately, the mapping problem is NP-hard problem
  - 4 x 4: tiles: 16! mappings
  - 10x10 tiles that are anticipated in five years or less
- This paper proposed the *branch-and-bound* algorithm

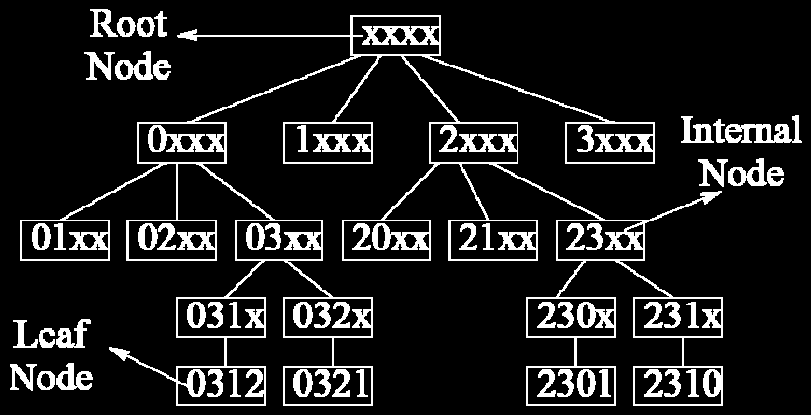
The Data Structure

- Walking through the search tree represents the whole searching space
- Example :
  - Mapping an application with 4 IPs onto a 2x2 tile architecture



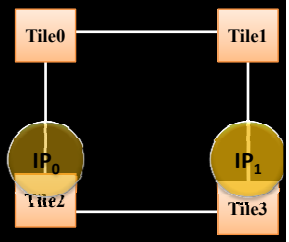


The Data Structure



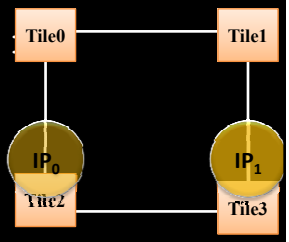
The Data Structure

- Example:
  - "23xx" → IP0 and IP1 are mapped to Tile2 and Tile3 respectively, while IP2 and IP3 are still unmapped



The Data Structure

- **Definition 3**
- The ***cost*** of a node is the energy consumed by the communication among those IPs that have already been mapped.
- For instance
  - The cost of the node labeled "23xx"
    - $v(a_{0,1})\ e(p_{2,3}) + v(a_{1,0})\ e(p_{3,2})$

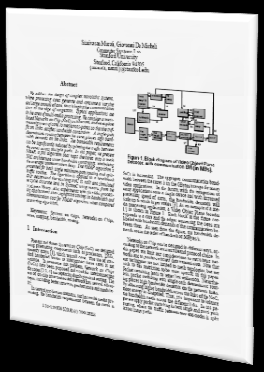


The Branch-and-Bound Algorithm

- *Branch:*
  - An unexpanded node is selected from the tree
  - The next unmapped IP is enumeratively assigned to the set of remaining unoccupied tiles
  - The corresponding new child nodes are generated.
- *Bound:*
  - Each of the newly generated child nodes are inspected to see if it is possible to generate the best leaf nodes later.

NMAP

- **Bandwidth-Constrained Mapping of Cores onto NoC Architectures**
  - **Authors:** Srinivasan Murali, Giovanni De Micheli



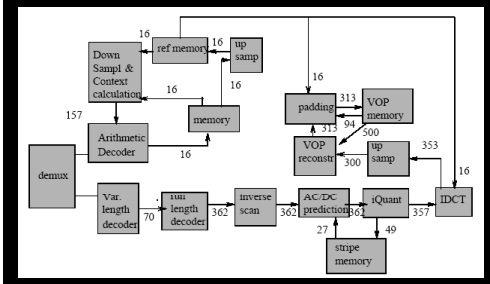
Mathematical Formulation of the Mapping Problem

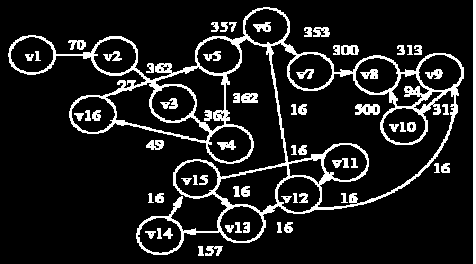
- **Definition 1**
- ***The core graph is a directed graph → G(V,E)***
  - *Each vertex vi ε V*
    - *representing a core*
  - *Each edge (v_i, v_j), $e_{i,j}$ ε E,*
    - *representing the communication between the cores v_i and v_j .*
  - *The weight of the edge $e_{i,j}$, denoted by $comm_{i,j}$,*
    - *represents the bandwidth of the communication from vi to vj .*

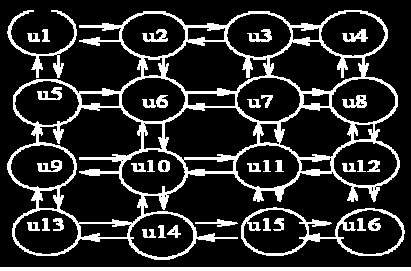
Mathematical Formulation of the Mapping Problem

- **Definition 2**
- ***The NoC topology graph → a directed graph***
  - P(U, F) *with each vertex u_i ε U*
    - *representing a node in the topology*
  - *Each edge (u_i , u_j), denoted as fi,j ε* F
    - *representing a direct communication between the vertices* ui *and uj.*
  - *The weight of the edge $f_{i,j}$, denoted by $bw_{i,j}$*
    - *represents the bandwidth available across the edge $f_{i,j}$.*

Mathematical Formulation of the Mapping Problem







Mapping function

- Mapping function map:

$$map : V \rightarrow U \text{ , s.t. } map(v_i) = u_j, \forall v_i \in V, \exists u_j \in U$$

$$|V| \leq |U|$$

- flow of single commodity (d^k), $k = 1, 2, \cdots |E|$.
  - The communication between each pair of cores (i.e. each edge ei,j ε E)

Mapping function

- The value of d^k $vl(d^k)$
  - Represents the bandwidth of communication across the edge

$$D = \left\{ \begin{array}{l} d^k : vl(d^k) = comm_{i,j}, k = 1, 2, \cdots |E|, \forall e_{i,j} \in E, \\ \text{with } source(d^k) = map(v_i), dest(d^k) = map(v_j) \end{array} \right\}$$

- The bandwidth constraints

$$\sum_{k=1}^{|E|} x_{i,j}^k \leq bw_{i,j}, \ \forall i, j \in 1, 2, \cdots, |U|$$