Heterogeneity &application in multi-core platforms

By:
Midia Reshadi, Ph.D
CE Department of Science and Research Branch of Islamic Azad University(SRBIAU)
reshadi@ieee.org
reshadi@srbiau.ac.ir
www.irprofs.com/reshadi

Shekhar Borkar, Intel 2006

Agenda

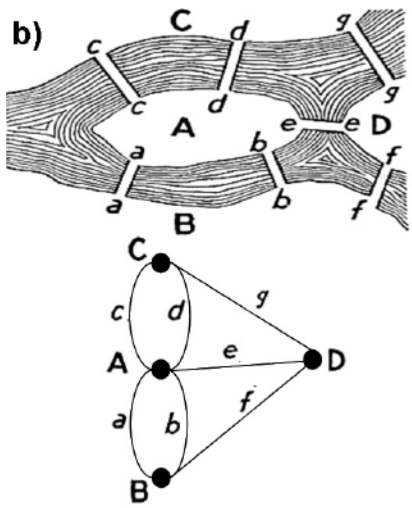
1 • Review of graph theory
2 • Reminder [Previous session]: IP core and heterogeneity in multi-core
3 • Application description and modeling
4 • Known application and core graphs
5 • More complex
6 • Scheduling
7 • Tools

The review of graph theory

- The science of networks is primarily based on the graph theory
- One of the most successful developments in mathematics ever
- The graph theory originates in 1736
- when Leonhard Euler offered the first rigorous solution to the "Seven Bridges of Konigsberg" problem

Leonhard Euler & Seven Bridges of Konigsberg

Leonhard Euler [1707-1783]



Euler's Graph

1

The review of graph theory

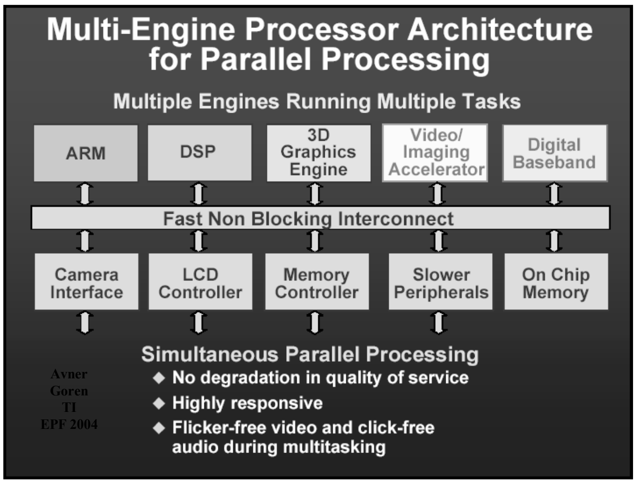
- The applications of static graphs:
  - The register allocation
  - Scheduling
  - compiler design
  - VLSI design;
- Erdos and Renyi: random Graphs
  - Graph and probability theories
  - Modeling:
    - Social communication
    - Collaboration or biological networks

2

Review: IP Core

- Define core as a pre-designed, pre-verified hardware piece that can be used as a building block for large and complex applications on an IC
  - R. Gupta, Y. Zorian, Introducing core-based system design, IEEE Des. Test Comput. 14 (4) (1997) 15–25.
- IP core can be:
  - Processor /Processing Element (PE)
  - Memory
  - DSP
  - IO module
  - …

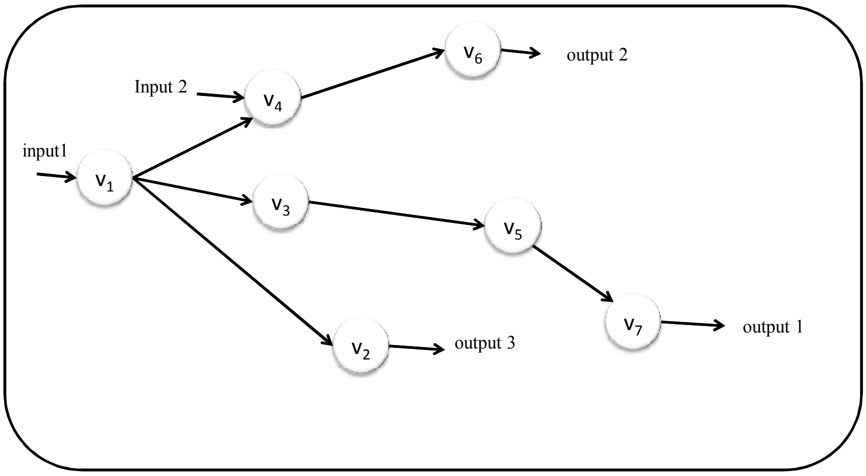
IP Core



Perspective

- **Definition 1.** ***An application characterization graph (ACG),*** App = App(V_{App},E_{App}), is a directed graph, where
- each vertex $v_i \in V_{app}$ denotes a computational module of the application referred to as a task,
- and each directed edge $e_{ij} \in E_{App}$ characterizes the communication process from vertex v_i to vertex v_j , where:

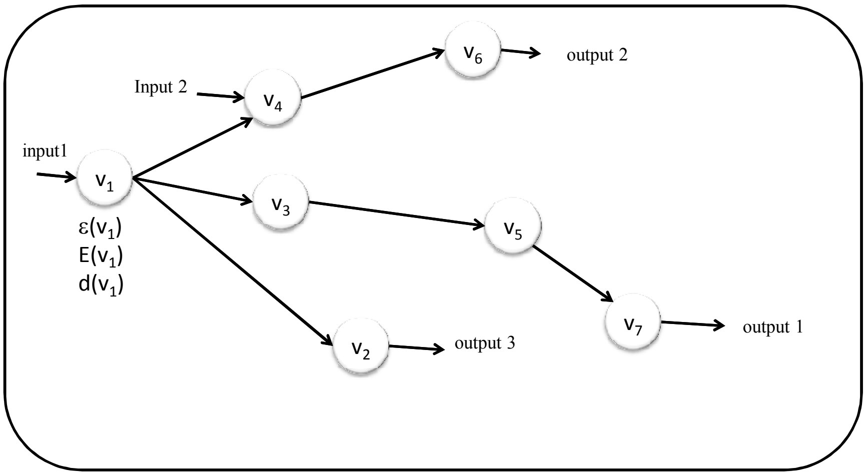
V=راس   E=یال



Cont,…

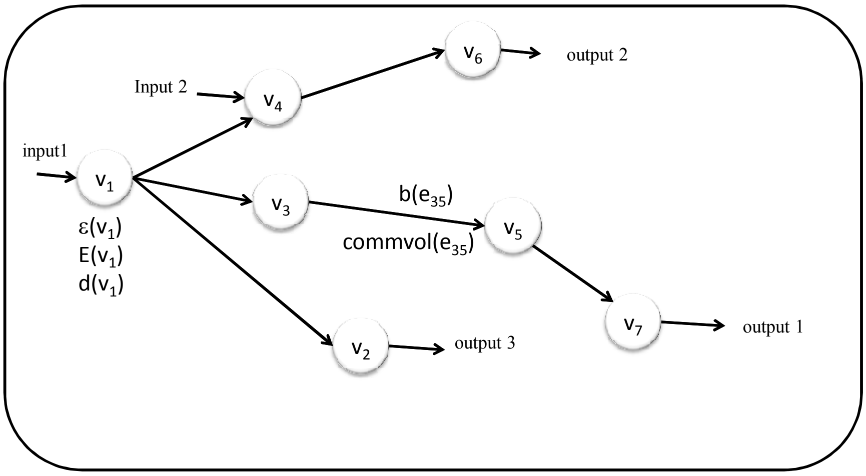
- Each vertex $v_i \in V_{App}$ is annotated with specific task information:
  - execution time $\varepsilon(v_i)$
  - energy consumption $E(v_i)$
  - execution deadlines $d(v_i)$

Cont,…

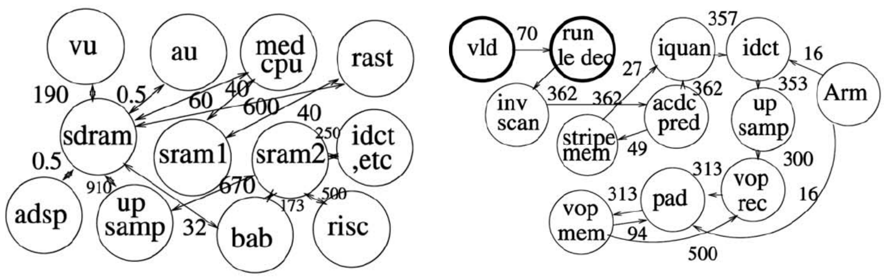


- In turn, each arc $e_{ij} \in E_{App}$ is tagged with: Application specific information
  - communication volume commvol(e_{ij}) from vertex v_i to vertex v_j in bits
- and specific design constraints e.g.:
  - communication bandwidth b(e_{ij}) in bits per second,
  - latency requirements lat(e_{ij}) in seconds, etc

Cont,…



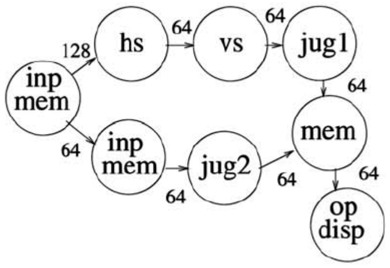
Popular application graphs



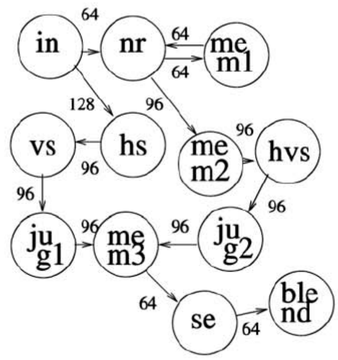
MPEG4

VOPD

Know application graphs

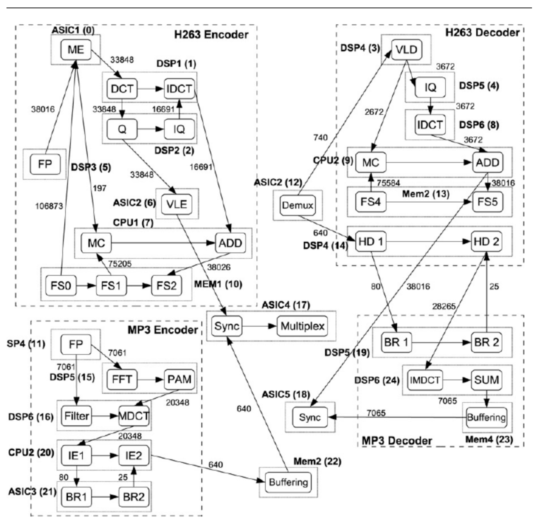


PIP



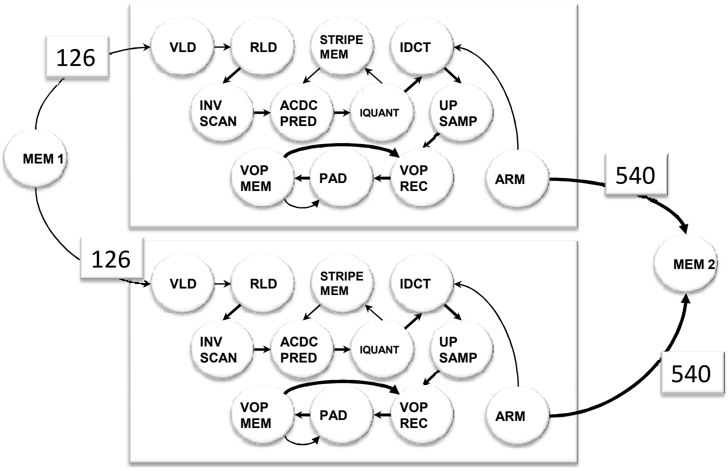
MWD

Know application graphs



Communication graph of the MMS

dVOPD



More complex appication

- Embedded Systems Synthesis Benchmarks Suite (E3S)
- Co-Synthesis Benchmarks Mailing list (cosynth-benchmarks@ee.princeton.edu)

Overview of the benchmarks

<A> Task types

- auto-indust

  <B> Target embedded system

  - cords:
    - Distributed system synthesis, supports reconfigurable hardware although we haven't included such processing elements in E3S, yet
  - cowls:
    - wireless client-server system
  - mocsyn:
    - system-on-chip
- networking
- telecom
- consumer
- office-automation

Overview of the benchmarks

File hierarchy

- <A>-<B>.fig:
  - The xfig format figure
    - Example: consumer-cowls.fig
- <A>-<B>.graph:
  - The graph structure
- <A>-<B>.tgff:
  - The complete problem specification and resource database
- <A>.html:
  - These are the "raw" HTML files from EEMBC (www.eembc.org), included with permission.

The subsidiary score

- Drawing the *-mocsyn.graph in visio or power point
- Importing the core information of tgff files to the excel or matlab
- Example:

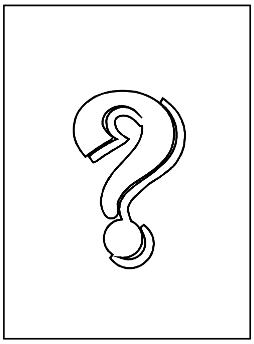
```
# AMD ElanSC520-133 MHz -- square
@CORE 0 {
# price, buffered, max_freq, width,  height, density,  preempt_power, commun_en_bit, io_en_bit
,idle_power
 33    1       1.33e+08 3.10e-03 3.10e-03 0.275    0          0          0        0.16
#----------------------------------------------------------------------------
# type, version, valid, task_time, preempt_time, code_bits, task_power,
# Angle to Time Conversion
0      0     1     9e-06     150E-6      6.9e+04   1.6

# Basic floating point
1      0     1     2.3e-05   150E-6      5.8e+04   1.6
```


TGFF tool

- Installation
  - 1. Download package
    - http://rw4.cs.uni-sb.de/users/sander/html/gsvcg1.html.
  - 2. mkdir (target dir)
  - 3. cd (target dir)
  - 4. tar -xzvf (filename)
  - 5. Read README
  - 6. make
  - 7. Enjoy.

Scheduling



QUESTION?