Kahn Process Networks and system level design

Kees Vissers and Pieter van der Wolf
Philips Research
UC Berkeley
vissers@eecs.berkeley.edu
Outline

• Introduction
  System level design problem

• Y-chart

• Kahn Process Networks

• Case study: MPEG-2 video decoder

• Coprocessor in Trimedia architectures

• Mapping

• Simulation Results
Future Consumer Multimedia Terminals

Welcome to Web-DVD

QuickTime

What's New?

PAL+

DVD-CDx frontend

PSTN or ISDN modem

DVD camcord.

DVB tuner

RF tuner

SCART interface

DV digital camcord. interface

CVBS interface

audio L/R interface

speakers

remote keyboard interface

RC5 interface

headphones interface

IEEE1394 interface

camera interface

cable modem

ATV

ISDN

PCMCIA interface

camera interface

mic interface

modem
Architecture Design

Goal
define a system architecture that implements desired functionality while satisfying cost and performance constraints

Current practice
from pictures and back-of-the-envelope calculations to synthesizable VHDL

Trends
increasing programmability, heterogeneity, complexity and use of IP

HW/SW Co-design
HW-SW choices, but also HW-HW choices, SW-SW choices

Need
support the exploration of the architecture design space of heterogeneous systems
From Applications to Architecture

Notes:
• not one, but a set of applications
• exploration of system architecture prior to HW / SW synthesis of blocks
General Scheme: The Y-Chart

Architecture → Mapping → Performance Analysis → Performance Numbers → Applications
General Scheme: The Y-Chart

- Architecture
- Mapping
- Performance Analysis
- Performance Numbers
- Applications
Distinction Application - Architecture

- Application imposes **workload** on **resources** provided by architecture
  - computation and communication workload
  - processing resources, communication resources, memory resources
- Application defines **timing constraints** for execution by architecture
- Architecture defines how **fast** a mapped application will execute
Levels of Abstraction

- Back-of-the-envelope
- Abstract executable models
- Cycle-accurate models
- Synthes. VHDL

Alternative realizations

Opportunities vs. Abstraction vs. Cost

High

Low
Y-chart Based Design Flow

Application C-code → Application model

Architecture specification → Architecture model

Mapping → Workload analysis

Mapping → Performance analysis
System design

- Model the system: Models of Computation, a.o. Kahn Process Networks
- Quickly evaluate a wide range of options
  - DSE (design space exploration) at abstract level
- Simulation to handle data dependent applications correctly
Design Flow

- Kahn API
- Application C-code
- Architecture specification
- Library blocks
- Kahn model application
- Architecture model
- Dedicated blocks
- Workload analysis
- Mapping
- Simulation
Producer-Consumer example

while(1)
{
  execute()
  write 1
  write 2
}

while(1)
{
  read 1
  execute()
  write 2
}

while(1)
{
  read 1
  execute()
  read 2
  execute()
}
Application Modeling

**Parallel Processes**
- internally sequential

**FIFO buffered Channels**

**Process Ports**

**Traces**

**API functions**

Use **API** to transform C-code into **Kahn Process Network** with proper **grain sizes**

Expose **parallelism** and **communication**

Reports **workload statistics** upon execution
MPEG Decoding

Used in:

- New High Definition TV standard in the USA, ATV
- DVD standard for movies in standard definition resolution
- Video Conferencing in low resolution (e.g. H263 standard)
Instrumentation of C-code

void Tidct(void)
{
    Appl_port* mb_F_In = get_appl_port("Tidct.mb_F_In");
    Appl_port* mb_f_Out = get_appl_port("Tidct.mb_f_Out");
    Ctrl_port* task_ctrl = get_ctrl_port("Tidct");
    ...

    while(1) {
        mb_F_In->read(mb_F);
        FastIdct(mb_F, &nr_of_bytes, &mb_prop, seq.block_cnt);
        Clip(mb_F, mb_prop.cbp, seq.block_cnt);
        task_ctrl->execute(IDCT_MB);
        mb_f_Out->write(mb_F, 768);
    }
}
Workload Analysis

<table>
<thead>
<tr>
<th>Task</th>
<th>Instruction</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tidct</td>
<td>IDCT_MB</td>
<td>12514</td>
</tr>
<tr>
<td>Tadd</td>
<td>Skipped_MB</td>
<td>158</td>
</tr>
<tr>
<td>Tadd</td>
<td>Intra-MB</td>
<td>2037</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Channel</th>
<th>#Tokens</th>
<th>#Bytes</th>
<th>#MinBytes</th>
<th>#MaxBytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>predict_data</td>
<td>88218</td>
<td>5645952</td>
<td>64</td>
<td>64</td>
</tr>
<tr>
<td>predict_mv</td>
<td>12514</td>
<td>400448</td>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Mapping of Processes onto processors

Implementation of the blocking read and write semantics in hardware and/or software

Potentially mapping more than one process onto one processor: scheduling

Mapping of the fifo buffers and communication onto the architecture.
Trimedia processor architecture: VLIW core + coprocessors

UC Berkeley, September 30, 1999 / Page 20
Architecture Specification MPEG Decoder
Architecture Model MPEG Decoder
Trace Driven Simulation

(Data dependent) traces from application model are executed by architecture model
Architecture Modeling

Standard building block approach of non-functional library models

Blocks must be configured with instructions and associated latencies
Mapping

Application model
Architecture model
Mapping

- Processes get mapped to processors

\[ \text{HW / SW choice, HW / HW choice, SW / SW choice} \]

Processor performance given by **latencies of instructions**, e.g. latency of IDCT

Estimation of latencies is done off-line

- Ports get mapped to interfaces -> channels get mapped onto communication resources
Performance Analysis

- Simulations yield information on such metrics as:
  - processor utilization
  - read / write stalls on processor ports
  - bus utilization
  - wait times for bus

- Parameterized model can be used for automated Design Space Exploration
  - Use of Design Space Exploration Tools of Philips Research
  - Helps to identify valid ranges of parameters and to perform sensitivity analysis
  - Define budgets for latencies of processors
Example: Wait times for bus of VideoIn Unit

![VInput stalls for bus]

"int_VIn_o1.dat"
Example: Wait times for bus of Storage Unit
Design Space Exploration
Conclusions

• MPEG case illustrates the usage of **Kahn Process Network** models in Industrial Research

• Mapping of these models onto architectures

• Make a clear distinction between application and architecture (workload <--> resources)

• Start project with the construction of an **executable parallel model** of the application(s)
  - Standard API for application modeling --> **reusable** application models
  - Workload analysis

• Architecture models can be constructed quickly and conveniently
  - **Abstract level**
    - Library of **reusable** building block