Embedded Architectures: Configurable, Re-configurable, or what?

Reiner Hartenstein
University of Kaiserslautern

Re-configurable!

Pierre Paulin, STMicro (moderator); Henk Corporaal, IMEC; Reiner Hartenstein, University of Kaiserslautern; Oz Levi, Improv Systems; Marco Pavesi, Italtel; Chris Rowen, Tensilica.
Terminology has been highly confusing

„Re-configurable Hardware“ ??

this „Hardware“ is not hard !

it's Morphware

We need a concise terminology:
  a consensus is on the way
Terminology: DPU versus CPU ...

- DPU: data path unit
- DPA: DPU array
- GA: gate array
- rDPU: reconfigurable DPU
- rDPA: reconfigurable DPA
- rGA: reconfigurable GA

- DPU is no CPU: there is nothing central - like in a DPA
# Digital System Platforms clearly distinguished

<table>
<thead>
<tr>
<th>platform</th>
<th>program source running on it</th>
<th>machine paradigm</th>
</tr>
</thead>
<tbody>
<tr>
<td>hardware</td>
<td>(not programmable)</td>
<td>none</td>
</tr>
<tr>
<td>morphware</td>
<td></td>
<td></td>
</tr>
<tr>
<td>coarse grain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>fine grain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rGA (FPGA)</td>
<td>configware</td>
<td>anti machine</td>
</tr>
<tr>
<td>rDPU, rDPA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>reconfigurable data stream processor</td>
<td>flowware &amp; configware</td>
<td></td>
</tr>
<tr>
<td>data stream processor (hardwired)</td>
<td>flowware</td>
<td></td>
</tr>
<tr>
<td>instruction stream processor</td>
<td>software</td>
<td>von Neumann machine</td>
</tr>
</tbody>
</table>
flowware defines ....

... which data item at which time at which port

flowware manipulates the data counter(s) ...

... software manipulates the program counter
Configware / Flowware Compilation

rDPA
r. Data Path Array

data streams

configware
mapper
scheduler

flowware

address generator

data sequencer

wrapper

intermediate

high level source program

© 2002, reiner@hartenstein.de
http://hartenstein.de
University of Kaiserslautern
Xputer Lab
most important contributor to nano SoC

we need rDPAs for:
• cellular wireless
• multimedia
• other applications

key functionalities:
to cope with
• compute requirements
• unstable standards
• multiple standards

relative merits:
• performance
• flexibility
• time to market
• product longevity
**Processor Performance**

- **Memory (Moore's Law)**
- **Algorithmic Complexity (Shannon's Law)**
  - Normalized processor speed
  - Transistors/chip

Graph showing trends from 1960 to 2010:
- 1G
- 2G
- 3G
- 4G

- Silicon wire complexity
- Wireless

Legend:
- Microprocessor / DSP
- University of Kaiserslautern
- Xputer Lab

© 2002, reiner@hartenstein.de

http://hartenstein.de
Performance vs. Flexibility

DeMan [T. Claasen et al.: ISSCC 1999]
(*) R. Hartenstein: ISIS 1997

rDPAs go far beyond bridging the gap

MOPS / mW

1000
100
10
1
0.1
0.01
0.001
2 1 0.5 0.25 0.13 0.1 0.07

µ feature size

flexibility

throughput

hard-wired
rDPA
FPGA
don Neumann

Reconfigurable logic
rDPAs (reconfigurable computing)*

hardwired

instruction set processors

standard microprocessor

DSP
cSoC for wireless communication et al.

Xtreme processing unit (XPU) from PACT

http://pactcorp.com

Layout for UMC 0.13 mm CMOS

© 2002, reiner@hartenstein.de

http://hartenstein.de
• A Fundamental Paradigm Shift in Silicon Application

---

**Time to Market**

- **Product**
  - ASIC
  - Update 1
  - Update 2
  - Reconfigurable Product with download

---

[Tom Kean]
Appendix for discussion
Why a dichotomy of machine paradigms?

vN: unbalanced

vN bottleneck

data stream machine:

• bad message: caches do not help
• good message: no vN bottleneck
• caches not needed
Soap Chip* Platform Template

important: coarse grain morphware

*) System on a programmable Chip

http://hartenstein.de
Programming sources

von Neumann
instruction stream machine

hardwired only

Anti machine
data stream machine

reconfigurable or hardwired
Machine paradigms

von Neumann instruction stream machine

Software

data-stream machine

Flowware

Configware

embedded memory architecture*

© 2002, reiner@hartenstein.de

http://hartenstein.de
new business model needed

the key enabler: morphware
### Glossary

<table>
<thead>
<tr>
<th>DPU</th>
<th>data path unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>rDPU</td>
<td>reconfigurable DPU</td>
</tr>
<tr>
<td>DPA</td>
<td>data path array (DPU array)</td>
</tr>
<tr>
<td>rDPA</td>
<td>reconfigurable DPA</td>
</tr>
<tr>
<td>ISP</td>
<td>instruction set processor</td>
</tr>
<tr>
<td>AM</td>
<td>anti machine</td>
</tr>
<tr>
<td>AMP</td>
<td>data stream processor (*)</td>
</tr>
<tr>
<td>rAMP</td>
<td>reconfigurable AMP</td>
</tr>
</tbody>
</table>

*) no “dataflow machine”

### Digital System Platforms:

<table>
<thead>
<tr>
<th>Platform Category</th>
<th>Source “Running” on Platform</th>
<th>Machine Paradigm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware</td>
<td>(not programmable)</td>
<td>None</td>
</tr>
<tr>
<td>ISP**</td>
<td>Software</td>
<td>Von Neumann</td>
</tr>
<tr>
<td>Morphware</td>
<td>Configware</td>
<td>FPGA: None</td>
</tr>
<tr>
<td>Data Stream Processor (AMP*)</td>
<td>Flowware</td>
<td>Anti Machine</td>
</tr>
</tbody>
</table>

### Categories of Morphware:

<table>
<thead>
<tr>
<th>Morphware Use</th>
<th>Granularity (Path Width)</th>
<th>(Re)Configurable Blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reconfigurable Logic</td>
<td>• Fine Grain (FPGA) (~1 bit)</td>
<td>CLBs</td>
</tr>
<tr>
<td>Reconfigurable Computing</td>
<td>Coarse Grain (e.g. 32 bits)</td>
<td>rDPUs (e.g. ALU-like)</td>
</tr>
<tr>
<td></td>
<td>Multi Granular: By Slice Bundling</td>
<td>rDPU Slices (e.g. 4 bits)</td>
</tr>
</tbody>
</table>