Design Methodology for the SVENm Multimedia Engine

Florian H. Seitner[3], Josef Meser[1], Gerold Schedelberger[2], Andreas Wasserbauer[2], Michael Bleyer[3], Margrit Gelautz[3], Markus Schutti[2], Ralf M. Schreier[1], Premysl Vaclavik[1], Gerald Krottendorfer[1], Günther Truhlar[2], Thomas Bauernfeind[4], Philipp Beham[2]

Contact
Florian Seitner, seitner@ims.tuwien.ac.at,
Gerald Krottendorfer, gkrottendorfer@odmsemi.com,
Markus Schutti, Markus.Schutti@infineon.com
Thomas Bauernfeind, bfnd@riic.at

Interactive Media Systems Group
Institute of Software Technology
Vienna University of Technology

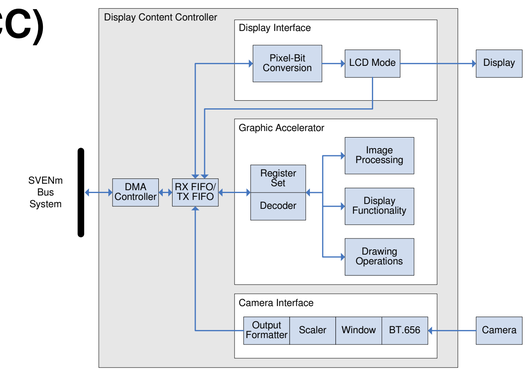
Display Content Controller (DCC)

Video Input
- RAW RGB data
- ITU-BT.656 / 601 YCbCr stream
- Compressed stream mode (e.g. MPEG TS)

Video Output
- MIPI DPI / DBI 2.0, RGB data
- ITU-BT.656, YCbCr data
- TVout Composite
- Double-buffering with V-sync support

Video Pre- / Postprocessing
- Windowing / Scaling / Mirroring / Rotation
- Format / Color conversion
- Overlay with alpha blending

2D / 3D Graphics



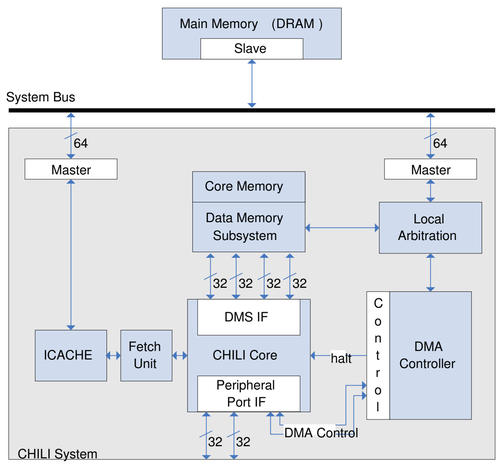
(DCC)

Generic HW Generation based on
- Set of Camera / Display Interfaces
- Set of Imaging / Graphics Features
- Set of Performance Settings
- Type / Number of Bus Interfaces
- Bus behavior

CHILI Vector Processor

CHILI Design
- CHILI Core with **32bit / 4 Slots / 8 SIMD**
- High performance for **signal processing** and **control code**
- **Compiler friendly** instruction set
- Fully **programmable** (C / Assembler)
- C-Compiler (LLVM, GCC) and instruction set simulator available

CHILI Processor Features
- Separate instruction and data path
- 16-bit SIMD operands
- 64 32-bit general purpose registers
- 128-bit core memory interface
- 64 KB instruction cache
- 64 KB data SRAM (core memory)
- 64-channel data load and store DMA controller
- 1.92 GMAC 16-bit operations (@ 240 MHz)



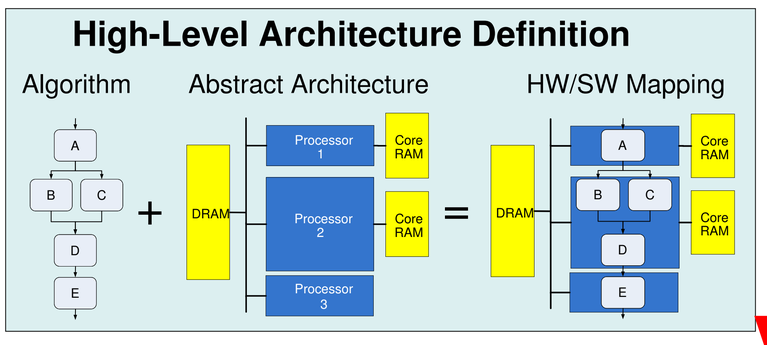
(CHILI Structure)

Design flow

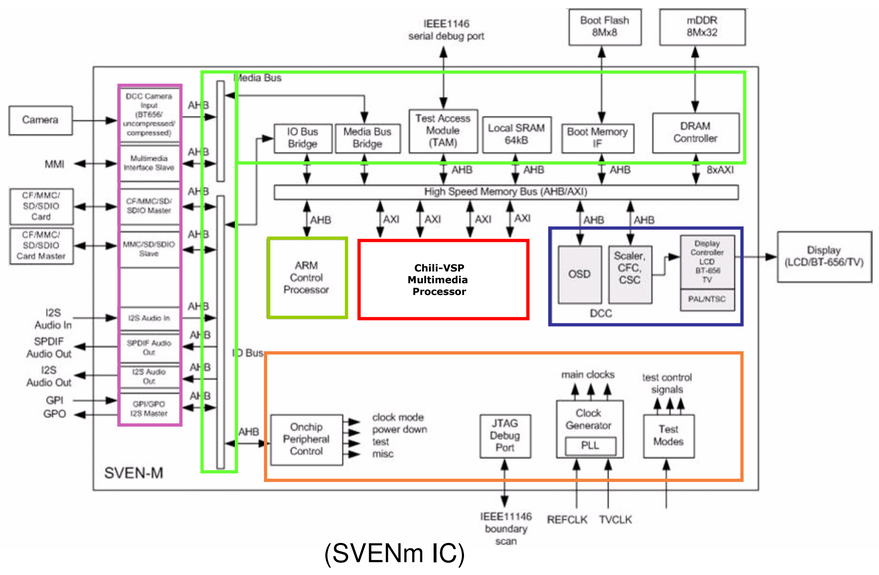
Partition Assessment Tool (PAT)

PAT is a simulator which provides a fast way to
- Describe video coding algorithms and HW architectures by an **high-level architecture description**
- Estimate the **run-time behavior** of these architectures
  - Can I achieve video encoding and decoding requirements on my current hardware?
  - What HW is required to handle my specifications?
  - What is the optimal algorithm partitioning?
- **Explore new system designs** in a flexible way

Design Methodology
1. Defining an architecture by a **High-Level Architecture Description**
2. Collecting of profilings / measurements / expertise of the platform components / software
3. Estimating the system behavior with PAT based on available information from Step 2
4. Design goals reached? If yes, design finished.
5. If not, adaptation of hardware architecture and back to Step 2



SVENm IC

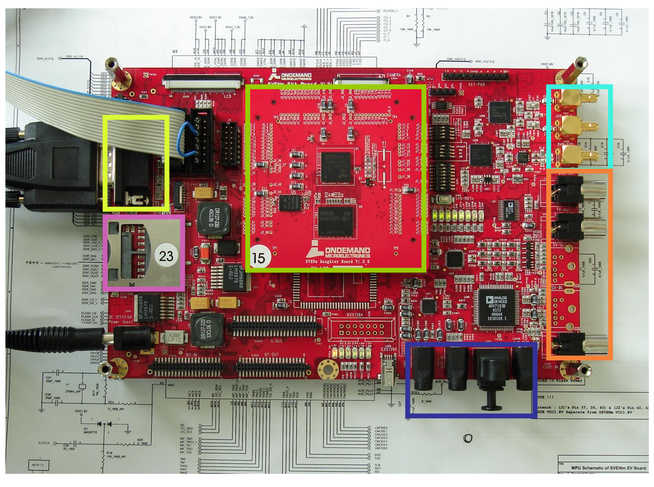


(SVENm IC)

- Video Encode / Decode
- Graphic Processor (DCC)
- Audio, Stream Mux
- Test, Debug, Control
- Interfaces
- Peripherals

Evaluation Board

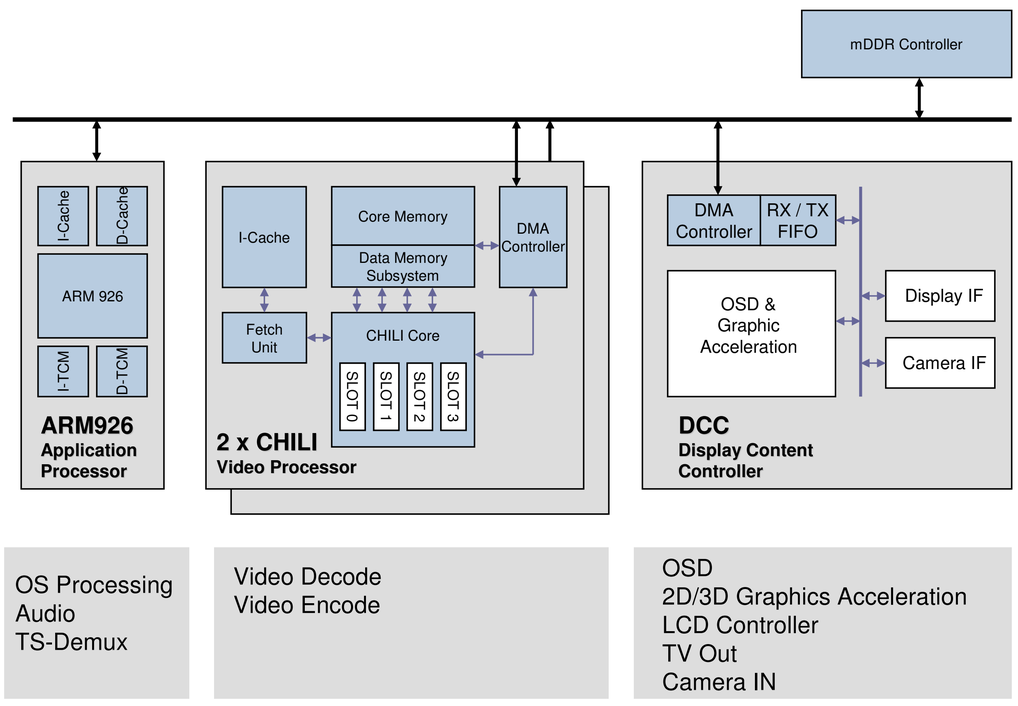
- SVENm Board
- Digital video interfaces
- Analog video interfaces
- Audio interfaces
- SD card slot
- Serial interface



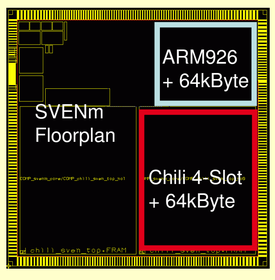
(Evaluation Board)

SVENm Multimedia Engine

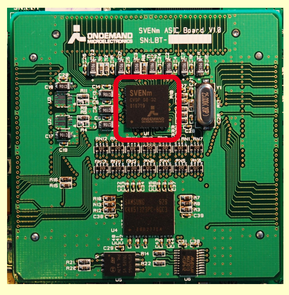
- Video / multimedia companion
- Targets H.264 encoding / decoding at SD resolution (PAL / NTSC D1)



(SVENm Structure)



(SVENm Floorplan)



(SVENm Board)

Conclusions

The Partition Assessment Tool (PAT) can estimate the run-time behaviour and performance of a multi-core decoder system. Architecture evaluations are possible before the decoder architecture is effectively built. Additionally, the PAT allows software design explorations for supporting the partitioning of the decoder software. Based on these results, the requirements of the decoding and displaying tasks in terms of computational complexity, data bandwidth and latency can be considered during the development of such a decoder platform. The resulting architecture, the SVENm multimedia engine, is capable of **decoding H.264 baseline at D1@30 Hz (3.2 Mb/s)**.

Acknowledgment

This work has been supported by the Austrian Federal Ministry of Transport, Innovation, and Technology under the FIT-IT project VENDOR (Project nr. 812429).

Project partners

ON DEMAND Microelectronics[1]
DICE: Danube Integrated Circuit Engineering[2]
Vienna University of Technology, Institute of Software Technology and Interactive Systems[3]
Johannes Kepler Universität Linz, Research Institute for Integrated Circuits[4]