Dynamically Reconfigurable Systems-on-Chip for Video-based Driver Assistance

Walter Stechele
Technische Universität München
Institute for Integrated Systems

Basic goals of reconfigurable computing are to combine the programmability and flexibility of microprocessors with high performance and low power consumption of dedicated architectures. First ideas were published in the 60s, but the technology in those years was too restricted to implement them. It was in 1986 when Xilinx offered the first commercial FPGA. Today, due to Moore's Law the semiconductor technology enables FPGAs with millions of reconfigurable logic elements, multiple CPUs, and Memory in a System-on-Chip (SoC). But semiconductor technology is just one enabling factor for reconfigurable computing. Design techniques and programming techniques are of equal importance. Today, design techniques are well established for dedicated logic, either ASIC or FPGA, and CPUs, but there is a lack of design techniques for reconfigurable systems. Programming techniques are highly sophisticated for single CPUs, but just of medium efficiency for parallel CPUs, e.g. VLIW (Very Long Instruction Word). For reconfigurable systems, work on reconfigurable compilers started in Princeton as early as in the beginning 90s, but it is still a topic of ongoing research.

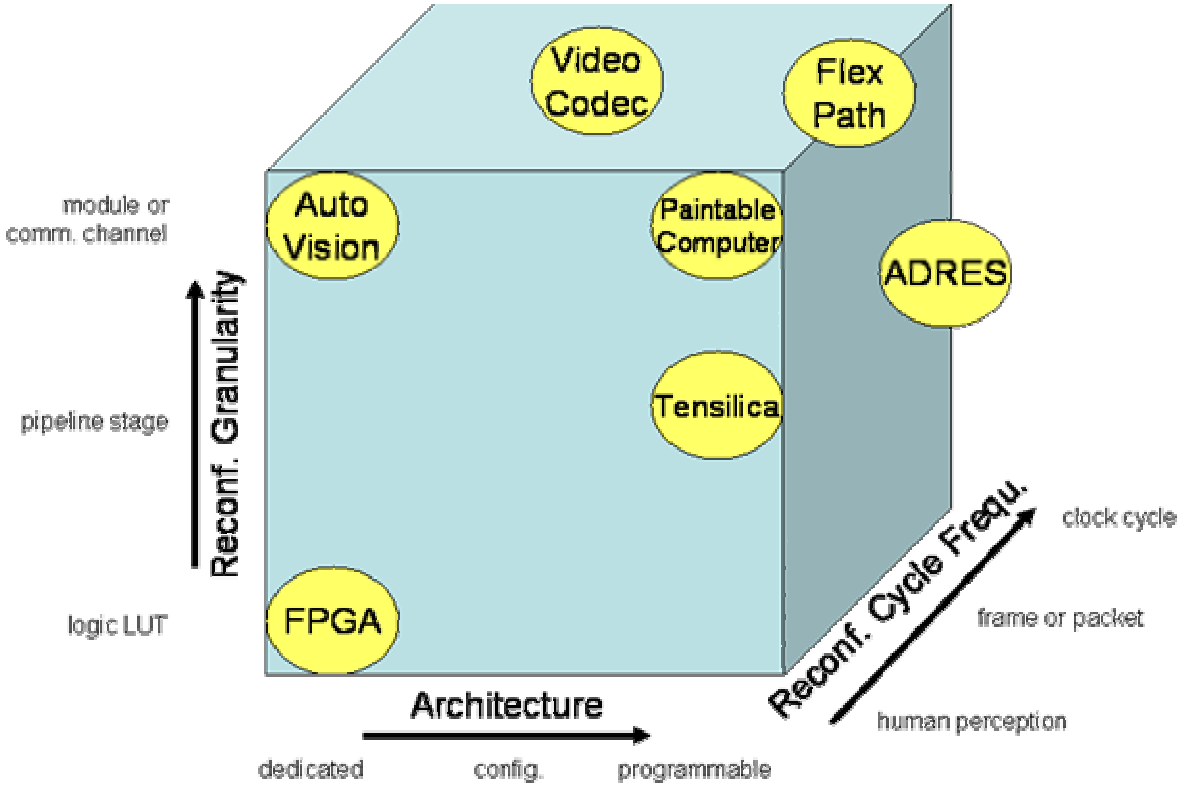

Fig. 1: Design space for dynamically reconfigurable Systems-on-Chip.

The design space for dynamically reconfigurable SoCs can be seen in three dimensions: 1) the system architecture for computation and communication, ranging from dataflow-oriented dedicated logic blocks to instruction flow-oriented microprocessor cores, from dedicated

Dagstuhl Seminar Proceedings 06141
Dynamically Reconfigurable Architectures
http://drops.dagstuhl.de/opus/volltexte/2006/744

point-to-point connections to Networks-on-Chip. 2) the granularity of reconfigurable elements, ranging from simple logic Look-Up-Tables to complex hardware accelerator engines and reconfigurable interconnect structures. 3) the configuration life cycle, ranging from application changes (in the order of seconds) to instruction-based reconfiguration (in the order of nanoseconds).

We propose to use dynamically reconfigurable computing for video processing in driver assistance applications. In future automotive systems, video-based driver assistance will improve security. Video processing for driver assistance requires real time implementation of complex algorithms. A pure software implementation, based on low cost embedded CPUs in automotive environments, does not offer the required real time processing. Therefore hardware acceleration is necessary. Dedicated hardware circuits (ASICs) can offer the required real time processing, but they do not offer the necessary flexibility. Specific driving conditions, e.g. highway, country side, urban traffic, tunnel, require specific optimized algorithms. Reconfigurable hardware offers high potential for real time video processing and adaptability to various driving conditions.

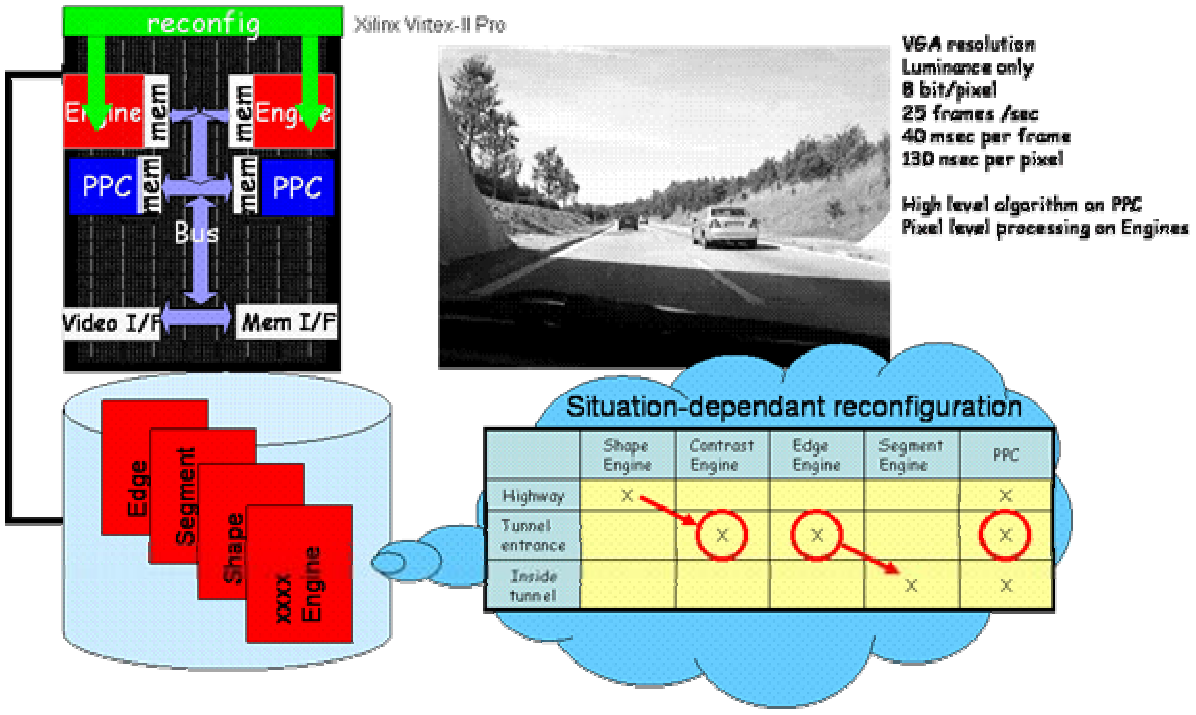

Fig. 2: AutoVision processor, using dynamically reconfigurable hardware accelerators for video processing in driver assistance applications.

Our system architecture consists of embedded CPU cores for high-level application code, dedicated hardware accelerator engines for low level pixel processing, and an application-specific memory system. The hardware accelerators and the memory system are dynamically reconfigurable, i.e. hardware accelerator engines can be exchanged during runtime, controlled by the application code on the CPU. The life cycle of a configuration depends on the change of driving conditions. A requirement on the reconfiguration time is given by the frame rate of the video signal, e.g. 40 msec for the exchange and relocation of new engines.

This work is supported by the German Research Foundation (DFG) under the priority program SPP 1148 “Reconfigurable Computing”, by the Xilinx University Program XUP, and by the group of Prof. Becker in University of Karlsruhe.