**Motivation**

- H.264/AVC is nowadays the de facto standard for video applications, due to its high coding efficiency and flexibility;
- The computational complexity significantly limits its usage in embedded systems of low and medium complexity;
- The most typical video applications, such as video telephony, conferencing or surveillance, are usually aimed for such systems;
- Modern multimedia embedded systems consists of quite complex heterogeneous multi-core structures, with one GPP and multiple specialized hardware accelerators;
- Parallelization overhead and communication time often impose great penalties in the performance levels of such systems;
- An efficient hardware / software co-design approach is therefore mandatory to successfully implement this class of applications.

Complexity requirements of H.264/AVC coding software

- A new multi-core H.264/AVC video encoder is herein proposed, by applying a novel hardware / software co-design methodology;
- Suitable for implementing small / medium complexity video coding embedded systems;
- Besides one GPP, multiple specialized processing cores are used as hardware accelerators for the most time consuming operations;
- A standardized, flexible and highly efficient bus is used to interconnect all the cores;
- The software component implements a highly modular and parallel H.264/AVC video encoder optimized for embedded system applications.

**Hardware / Software Partitioning**

- The hardware and the software components of the system are designed together to obtain the intended performance levels;
- Code profiling tools are usually applied to assess the performance and memory requirements and help fine tuning the system;
- At the hardware level, the designer must select (and quite often implement) the system CPU, hardware accelerators, peripheral devices, memory and the corresponding interconnection structure;
- The software component addresses the design of a program to efficiently implement the application algorithms and to support the communications between all the system hardware components;
- The code is further optimized by taking into consideration the characteristics of the hardware components and by applying the most complex and efficient modes of the software compiler tools.

**H.264/AVC Multi-Core Video Encoder**

- The video encoding operations were partitioned so that the ASIP IP core performs the ME operation and the Leon3 processor all the remaining operations;
- The ASIP IP core was enhanced with several improvements and optimizations, in order to increase its communication bandwidth and optimize the interface to the AMBA 2.0 APB bus;
- The p264 [2] parallel programming framework was chosen as the system video encoding application;
- Some modifications were introduced to the p264 base software description, fine tuning it for embedded systems implementations;
- Efficient APIs were developed for interaction with the hardware accelerators, guaranteeing optimal efficiency for communications;
- The APIs allow to easily replace original software implementations of a functional block of the video encoder by a system call to the corresponding hardware accelerator.

**Implementation and Results**

- The multi-core architecture was implemented in a GR-CPCI-XC4V development board;
- Several ME algorithms were programmed in the ME IP core: FSBM, DS and 3SS;
- The video encoding system was validated by encoding the bream, carphone, forearm, mobile and table tennis QCIF sequences.

Complexity requirements of H.264/AVC coding software

- Huge reduction in the computation time of the ME operation;
- Transfer times for the pixel data (MB and SA) and for the ME results (MV) are negligible about 0% of the total encoding time;
- The obtained speedup values successfully validate the adopted methodology and hardware / software design partitioning.

**Most relevant publications:**


**Acknowledgments:** This work was supported by the Portuguese Foundation for Science and Technology (INESC-ID multimannual funding) through the PIDDAC Program funds and by the PROTEC Program funds under the research grant SFRH / PROTEC / 50106 / 2009.