Abstract

Multimedia applications are present in most mobile hand-held devices. H.264 is an emerging video coding standard, which aims at compressing high-quality video contents at low-bit rates. While the new encoding and decoding processes are similar to many previous standards, the new standard includes a number of new features and thus requires much more computation than most existing standards do. The complexity of H.264 standard poses a large amount of challenges to implementing the encoder/decoder in real-time requiring large amount of processing resources. This paper presents the design and analysis of the H.264 decoder implemented on a heterogeneous architecture (multi-CPPUs/multi-GPUs). A model-driven approach is adopted by using the standard MARTE profile of UML. Our approach is based on hybrid partitioning that combines both functional and data partitioning which is applied to find the most suitable processors (CPU or GPU) regarding the execution time. We claim that our approach allows giving a better performance, which is crucial when implemented in modern complex systems.
References

3. AISO/IEC. International standard. Part 10: Advanced video coding,
13. Elias Baaklini and all, H.264 Parallel Optimization on Graphics Processors, MMEDIA 2013 : The Fifth International Conferences on Advances in Multimedia
and Compilation Techniques, Sep., 2012.
JVT-G050, Mar.
Video-Compression Standard. IEEE Transactions on Circuits and Systems for Video
2006.
16x16 prediction for H.264/AVC” International Conference on Embedded Systems & Critical
Applications ICESCA,2008, Tunisia.
Integer Motion Estimation Implementation for H.264/AVC Baseline Encoder”, The International
using ESL Tools”, 2011 8th International Multi-Conference on Systems, Signals &Devices,SSD.
30. AMD Evergreen Family Instruction Set Arch. (vl.Od).

**Index Terms**

Computer Science

Circuits and Systems

**Keywords**

General-Purpose Graphics Processing Unit (GPGPU), Multimedia, H.264/AVC decoder, Parallel
Processing, Functional Partitioning, Data Partitioning.