A Compiler-friendly and Low-power DSP architecture

E. Matúš, G. Cichon, H. Seidel, P. Robelly, T. Limberg, G. Fettweis

Vodafone Chair Mobile Communications Systems
Technische Universität Dresden
D-01062 Dresden
{matus,cichon,seidel,robelly,limberg,fettweis}@ifn.et.tu-dresden.de

In view of the rising complexity of the algorithms embedded in software as well as the processors the replacement of the traditional assembler programming is urgently necessary by the use of high level language compilers. However only the interaction of hardware (circuit development and processor architecture) and software (compiler and algorithms) guarantees an altogether low-power system implementation tailored to the signal processing algorithms.

The goal of our research group in the VIVA project was the development of new processor architectures and compiler techniques for the minimization of the power consumption in processor-based DSP systems. The proposed design methodology considers equally hardware and software requirements during the algorithm-adapted system development.

A new DSP SAMIRA was introduced in the project. SAMIRA is particularly adapted to the development low-power, high-efficient DSPs, which is programmable in the high-level language Matlab. The proposed design methodology is based on the architectural template STA (synchronous transfer architecture). The STA template enables the development of low-power and compiler-friendly DSPs. In the STA paradigm a DSP is modelled as a collection of elementary modules, which are connected each other through an interconnection network formed by multiplexers. The modules can be functional units (FU) or memory elements. The functional units implement arbitrary operations on their input data. Modules and interconnection multiplexers are controlled by the segments of very long instruction word (VLIW). At each cycle the instruction configures the multiplexing network and the functionality of the modules. Thus, the whole system forms a synchronous network, which consumes and produces some data at each clock cycle. This enables low-power operation because only connections and FUs necessary for current operation are activated. The STA architecture offers a high degree of data reusability: Data that produced in the current cycle can be directly routed to other processing units in the following cycles without going through the register file or memory. This not only speeds up computations but it lowers power consumption and register file pressure. The STA architecture also supports data (SIMD) and instruction level parallelism (VLIW). Based on STA architecture, the SAMIRA processor provides a high computing performance with little overhead required for programmability. For example SAMIRA running at 212 Mhz can execute a 256 complex FFT in 8 μ s.