Development of Embedded Multicore Systems Using TAU

Célio Estevan Morón  
Federal University of São Carlos  
Department of Computer Science, Brazil  
celio@dc.ufscar.br

It has become evident for some time that it is not economically wise to speed up further the processor clock in order to get an increase in the computer power. One of the alternatives, the one that is proving to be very successful, is to use several cores within a single chip. This trend has reached all areas of computing, from mainstream computation throughout embedded systems. In the form of specialized high-performance systems, or dedicated embedded systems, multiprocessor machines have been present among us for decades. The problem is to produce it in a form as easy as sequential programming. This new trend will change the way we think about the whole development process. We will show that it is possible to develop a multicore embedded system application using existing tools and the model-driven development process proposed. The process is carried out in a cyclic way generating more refined versions of the application. The proposed process will use two tools: VisualRTXC for generating the multithread communication/synchronization structures and a performance tool called TAU for tuning the final implementation. A Case Study shows, through a small application, the development process proposed and the steps involved in the implementation of an embedded system. This development process allows the developer to rapidly move between design concepts and generated C code, while at the same time provides a visual abstraction and design aids for each of the typical phases of the embedded software development life cycle.

Case Study: Mandelbrot - Code that generates the Mandelbrot set is a favorite target for evaluating performance in embedded systems. Embedded systems generally require a high amount of image processing to perform and the Mandelbrot set can be adjusted to demand the computer power necessary for evaluation.

Figure 1 - Layer Diagram  
Figure 2 - Code Diagram  
Figure 3 – TAU ParaProf

Conclusion - We have shown that with two existing tools, it is possible to develop multicore systems in an efficient manner. Our approach for the process of development is carried out by a combination of a graphical tool with a performance tool. Using a graphical tool, it is possible to better understand the communication/synchronization of tasks, while the performance tool allows to tune the system.