Due to the diversity of the application areas, embedded systems are characterized by requirements in terms of different aspects of the system. These requirements are in turn expressed as constraints imposed on different system resources such as space, time, and energy. The constraints often interact or even conflict with one another in a complex manner. Therefore, in designing a real-time embedded system, the tradeoff relationship among the different optimization criteria should be taken into account in order to build cost-effective systems. We propose a design optimization approach for real-time embedded systems, which balances a flexible tradeoff among the system's code size, execution time, and energy consumption.

The tradeoff between code size and execution time is enabled by using a dual instruction set processor, which supports a reduced instruction set in addition to a full instruction set. While the reduced instruction set can be used to reduce code size by providing smaller instructions, a program compiled into the reduced instruction set typically runs slower than the same program compiled into the full instruction set. Motivated by this observation, we exploit this tradeoff by selectively using the two instruction sets for different sections within a program. On the other hand, the tradeoff between execution time and energy consumption is exploited by using a variable voltage processor, which provides a mechanism to adjust the processor's supply voltage at run time. By lowering the supply voltage, the energy consumption of the processor can be reduced. However, the execution times of programs are lengthened because the operating clock frequency must be decreased according to the lowered supply voltage. Based on this property, we control the tradeoff by assigning an appropriate voltage/frequency setting to each task.

The proposed design approach combines a code generation technique and a system-level design parameter derivation algorithm. The code generation technique, called selective code transformation, determines the instruction set to be used for each basic block in such a way that the program's execution time is minimized while the code size does not exceed a given upper bound. As a result, the technique summarizes the tradeoff relationship between the program's code size and execution time in the form of a set of possible choices of different versions of code for a given program. Based on the results given by the selective code transformation, a system-level optimization algorithm derives a set of design parameters used by the compiler and the operating system. That is, the technique generates the code generation decision along with the frequency setting for each task, in such a way that the system cost function is minimized while the tasks collectively satisfy the resource constraints.

The main contributions of this thesis is summarized as follows. First, we identify the tradeoff relationship involving code size, execution time, and energy consumption, and propose a system optimization framework that flexibly balances this tradeoff. Second, the proposed technique mathematically formulates a multi-directional optimization problem and provides an algorithmic solution to it. Finally, the optimization is automatically guided by an abstract specification of system requirements given by the system developer.

* Note: The text above is the abstract of the thesis. You can use the full-text by clicking the "Link" (URL) of item record and the viewer program will be installed automatically. In case of having problem with the viewer program installation, please download it manually using this url http://pdfdrm.snu.ac.kr/snudrm/download/PDFReaderSetup.exe