ABSTRACT

Real-time Embedded systems have geared up their applications in the field of smart phones, mobile devices, automobile and medical instrumentation. Along with the growth of technology, the functional requirements of the systems have also grown to multi folds. To accommodate these ever-increasing functional requirements, the hardware has to be designed to include the maximum capability which in turn increases the power consumption. In CMOS devices, the power consumption is directly proportional to the operating frequency. So the power consumption has to be dynamically managed using the software techniques. Dynamic Voltage and Frequency Scaling (DVFS) feature provided in most of the recent processors has enabled the power optimization through software. The operating frequency of the processor can be reduced on the fly through the software. Many processors in the market today have the DVFS support with multiple voltage/frequency levels.

The execution time of the tasks can be stretched to use the laxity by reducing the clock frequency. The factor by which the frequency is reduced from the maximum clock frequency is called the slow down factor. The slow down factors are calculated using the off-line and the on-line techniques; off-line techniques reclaim the laxity based on the worst case execution time whereas on-line techniques reclaim the laxity based on the actual execution time of the tasks. The factors that are to be considered during the design of the energy efficient scheduling algorithms are Deadline constraints, Complexity, the number of frequency switching and energy consumption. The timing requirement of the tasks has to be met as most of the real-time systems are hard. The complexity of the algorithms should be kept minimal to avoid

overloading of the kernel in Real Time Operating System. Considerable time and energy overheads are involved during frequency switching even though the algorithms assume that the overheads in frequency switching are negligible. The energy consumption of the tasks is one of the prime design metrics which is the major contributor to the product cost and size.

The contribution of the thesis includes both off-line and on-line techniques for the calculation of the slow down factors for non-preemptive periodic tasks to be scheduled in uni-processor systems.

i) Two static algorithms for calculating the slow down factor is proposed namely Slow Down using Greatest Common Divisor (SD-GCD) and Hyper Period Based Method (HPBM). In both the algorithms the energy consumption and the number of frequency switching is reduced. In HPBM, the complexity of the algorithm is also reduced.

ii) A dynamic algorithm called DynaClam is proposed which calculates the slow down factor using actual execution time of the tasks. The advantage of this algorithm is that it caters the on-line timing requirement of the task with reduced computational complexity.

Portable real-time systems have started to use multi-core processors to meet the complex and increasing functional requirements. The design of the scheduling algorithms should be such that all the cores in the processor are effectively utilized. This research is extended to energy aware scheduling of the periodic non-preemptive tasks in homogenous multi-core environment and proposes two approaches using partitioned scheduling. After the partitioning of the tasks to the cores, the processor is operated at the lower

frequency using the DVFS feature to reduce the available laxity in the cores while meeting the deadline of the tasks.

iii) In the first approach, task partitioning is done using the existing Leakage Aware Largest Task First algorithm and Modified Simulated Annealing (MSA) based approach is proposed for reducing the energy consumption using DVFS feature.

iv) In the second approach, Firefly based heuristic technique is applied for task partitioning and MSA for reducing energy consumption. Task partitioning algorithms are applied off-line while energy reduction algorithms are applied on-line. The energy consumption and the laxity of the tasks have reduced to a considerable level and the utilization of the processor has also increased.

An indigenous tool is developed to analyze the performance of the energy efficient scheduling techniques.

v) A tool named, “SPARK”, is developed using MATLAB to test the effectiveness of the energy efficient scheduling algorithms. SPARK is capable of generating synthetic tasks with user defined utilization and range of task periods. After scheduling of the tasks using either Rate Monotonic or Earliest Deadline First algorithms, SPARK calculates energy consumption, laxity and utilization.

vi) The proposed HPBM method is applied to a real-time application, “Smart Soil Analyzer” which employs sensors to measure soil moisture, temperature and soil pH. The

measured power consumption of Smart Soil Analyzer with HPBM algorithm is comparatively lesser than USFI algorithm.

The experimental results of the proposed energy efficient algorithms for both uni-processor and multi-core processors indicate that the energy consumption is certainly reduced to a remarkable level and the overhead in frequency switching is also significantly reduced.

As a future work, the energy efficient scheduling algorithms can be applied to non-periodic and sporadic tasks. Accurate modeling of the processors with energy and time overhead can be considered to exactly analyze the effectiveness of the algorithms.