

UNIVERSIDADE FEDERAL DE RORAIMA

Disciplina: Introdução à Sistemas Embarcados

Aluno: Joshua Kook Ho Pereira

Curso: Ciência da Computação

SEMA: Na Approach Based on Internal Measurement to Evaluate Energy Efficiency of Android Applications

A obra tem como objetivo a quantificação do consumo de energia gasta por aplicativos baseados em Android através de medição interna como uma maneira de ajudar desenvolvedores encontrar gargalos e desenvolver métodos otimizados de consumo de energia. É afirmado que com o avanço da tecnologia os dispositivos requerem cada vez mais poder computacional, o que resulta em uma necessidade de melhorar o consumo de bateria destes dispositivos.

Várias maneiras são consideradas para esse fim, inclusive o próprio aplicativo de gerenciamento de bateria disponibilizado pelo sistema operacional. O artigo propõe então o SEMA (Self Energy Metering for Android – Auto Metragem de Energia para Android), uma estimativa baseada na medição interna focada em dispositivos Android.

Após um breve resumo de algumas técnicas e ferramentas de medição, é apresentado uma visão geral do método a ser implementado: extrair amostras de voltagens e correntes do circuito gerenciador de bateria, e usa essa informação do consumo de energia durante a execução de um código combinado com o tempo de execução para se obter uma estimativa do consumo de energia.

Ao final da execução do aplicativo, as amostras são analisadas e geram um log com tempo de execução, media de energia, energia total e todas as amostras coletadas. Os dados são usados para calcular o consumo multiplicando média de energia por tempo de execução. Para executar esse método foi utilizada a programação orientada à objetos na linguagem JAVA que monitoravam as funções do dispositivo, monitorar o consumo de energia e o serviço de bateria.

Foram feitos experimentos com uma placa Cubieboard A20 e uma placa Arduino usado para coletar dados da voltagem e da corrente para calcular o consumo, colocando um resistor entre a placa A20 e a bateria e ligando-o ao Arduino.

Após vários testes feitos com diversos algoritmos, os resultados foram comparados com um cálculo de medição externa e com um aplicativo usados para este fim. Foi observado que os resultado da SEMA diferem em mais de 20% dos outros métodos, mas que está relativamente numa posição intermediária entre os resultados.

Na conclusão, é avaliado as vantagens do SEMA (como a sua flexibilidade e maior precisão), com um destaque em sua precisão, o que dá aos desenvolvedores mais granulidade no desenvolvimento de algoritmos.

Por fim, se planeja conduzir experimentos em outros dispositivos Android, visando verificar a precisão de outros circuitos gerenciadores de bateria, e também melhorar a implementação para usar a energia integral ao invés da média de energia para o cálculo do consumo.

Integração de Características Preemptivas à Técnica de Escalonamento Dinâmico de Tensões e Frequências Intra-Tarefa

O artigo fala sobre a diminuição do consumo de energia do processador utilizando a técnica de escalonamento dinâmico de tensões e frequências do processador intra-tarefa, através da colaboração aplicativos de tempo real e o sistema operacional

O problema proposto é que devemos obter o menor consumo de energia através da troca de tensões e frequências, dependendo da tarefa sendo executada, garantindo que esta cumpra seu papel dentro do deadline e com o menor gasto possível, deixando ao processador o menor tempo possível de ociosidade. Um dos obstáculos é no contexto de múltiplas tarefas num sistema de escalonamento de prioridade fixa, além de saber se estas tarefas podem ser executadas o maior tempo possível nas suas tensões e frequências ideais.

Em suma, o problema a ser tratado é: será possível reduzir o consumo de energia do processador por meio de um método colaborativo entre aplicações de tempo real e o SO, de maneira que os dois trabalhem juntamente no restabelecimento eficiente de tensões e frequências do processador.

O objetivo é tratado em 3 objetivos específicos:

- Estabelecer um canal de comunicação entre o controlador de tensões e frequência do processador e aplicativos de tempo real;
- Dar ao SO o controle das tensões e frequência do processador, com base nos dados informados pelos aplicativos de tempo real;
- Demonstrar através de experimentos que o método proposto reduz o consumo de energia do processador e maximiza os ganhos obtidos com a técnica DVFS intra-tarefa diante de preempções.

A solução proposta é uma reestruturação do núcleo do SO para que o Governor, o escalonador e as aplicações de tempo real possam trabalhar em conjunto para prover uma solução eficiente aos problemas apresentados e reduzir o consumo de energia do processador.

A metodologia consiste em: 1) definir a lista de tarefas que serão executadas em tempo real e seus parâmetros; 2) definir tensões e frequências ideais de cada tarefa, assumindo o pior caso de execução; 3) aplicar um teste de escalonabilidade nos parâmetros anteriores para certificar de que as tarefas são escalonáveis e que não haverá violações nas premissas temporais; 4) inserir manualmente trechos de código nas tarefas de tempo real, passando as informações calculadas ao sistema operacional, preferencialmente no início da função principal de cada tarefa; 5) colocar as tarefas de tempo real em execução no SO.

Através dos experimentos se concluiu que mesmo executando todas as tarefas no pior caso, teve uma redução do tempo ocioso do processador e também houve uma diminuição no consumo de energia em volta de 6%, demonstrando que uma abordagem colaborativa entre as aplicações e o núcleo do SO permite gerenciar melhor as trocas de contexto do processador dependendo da tarefa que esteja em execução no sistema.

Em conclusão, o método proposto pelo artigo visa otimizar o gerenciamento das trocas de contexto do processador, para poder reduzir a quantidade de códigos estáticos inseridos dentro do código fonte das tarefas de tempo real para realizar as validações das tensões e frequências a serem utilizadas pelo processador; proporcionando otimizações na diminuição dos tempos de folga das tarefas e minimização do consumo de energia dinâmica do processador.