

18º Congresso Nacional de Iniciação Científica

TÍTULO: TESTE DE DESEMPENHO DE UNIDADES DE PROCESSAMENTO EM TEMPO REAL

CATEGORIA: EM ANDAMENTO

ÁREA: CIÊNCIAS EXATAS E DA TERRA

SUBÁREA: Engenharias

INSTITUIÇÃO(ÕES): FACULDADE DE ADMINISTRAÇÃO DE EMPRESAS - FACAMP

AUTOR(ES): IAGO

ORIENTADOR(ES): JOSÉ MARTINS JUNIOR

1. RESUMO

Sistemas computacionais de tempo real são aqueles em que a execução de tarefas é feita de forma previsível, e o seu tempo de execução não é afetado pela quantidade de tarefas a serem feitas. O seu uso no dia a dia da maioria das pessoas não é comum, o que é, são os sistemas de computacionais tradicionais, não determinísticos. Para atividades que requerem um controle preciso de automação, os sistemas tradicionais não podem ser aplicados, uma vez que o tempo de resposta, não é fixo, o que diminui a confiabilidade (que pode ser exemplificada como o processamento de áudio *Hi-Fi*). Junto a esse critério, outro que também é importante e pode ser endereçado nos sistemas de tempo real é a criticalidade, onde tarefas em que uma vez não tenham seus objetivos atingidos, a vida de pessoas pode ser colocada em risco (como equipamentos biomédicos, freio *Antiblockier-Bremssystem* - “ABS”, controle de aeronaves). Por fim, atua ao contrário de soluções de tempo real, com tempos de tarefas constantes, permitindo padronizar e processar com maior facilidade.

2. INTRODUÇÃO

Nos dias atuais, boa parte dos sistemas operacionais de computadores que a grande maioria dos usuários usam, não são determinísticos. E, para o uso destes usuários, mesmo que estejam a executar somente uma tarefa (como ver e-mails ou redigir um texto, por exemplo) já tem em seu computador algumas dúzias de tarefas diferentes para serem aferidas, desde o nível do usuário (isto é, o processo do navegador diretamente, a processos de sistema e processos em segundo plano). E essas tarefas são processadas uma por vez (em um sistema de núcleo único, *single core*), alternando entre as diferentes tarefas em uma velocidade extremamente alta. A escolha da próxima tarefa não pode ser prevista, pois é feita de forma dinâmica com base em uma série de critérios (desde carga do núcleo, temperatura, prioridade, tipo de instrução, localização no cachê, quantidade de ciclos necessários), definindo assim, a característica não determinística do sistema. Isso significa que não há preempção síncrona, ou seja, a interrupção de um processo que está sendo processado em um núcleo para o chaveamento a outro processo não segue um padrão (KANT, 2010; TANENBAUM, 2016).

Por não seguir um padrão de interrupção, sistemas não determinísticos não são indicados para controle de processos de automação, onde a atuação deve ocorrer no menor tempo possível após uma dada entrada, e nos casos não determinísticos, essa atuação é inconsistente e irregular. Em soluções de tempo real, por outro lado, o tempo de atuação é cíclico e constante, portanto, pode ser padronizado e processado com certa facilidade (um exemplo de aplicação é o frio ABS). Isso também pode ser visto no *hardware*, que é mais simples (RISC), sem *pipelines*, sem cachês, de forma a ser otimizado para a atividade de tempo real, tanto no processamento, quanto na atuação de I/O (KANT, 2010).

3. OBJETIVOS

Objetiva-se, por este trabalho, aferir o desempenho de uma solução de tempo real *open-source* em laços de controle em comparação a soluções convencionais. Especificamente, por meio de testes empíricos de tempo de resposta na placa *open-source BeagleBone Black Rev. C*, entre a dupla de micro controladores PRU e o núcleo Cortex-A8.

4. METODOLOGIA

A pesquisa envolveu a revisão bibliográfica dos conceitos de sistemas computacionais de tempo real (sistemas operacionais e hardware) e sistemas computacionais tradicionais, viabilizando a elaboração de cenários de teste que terão os seus dados coletados através de um osciloscópio.

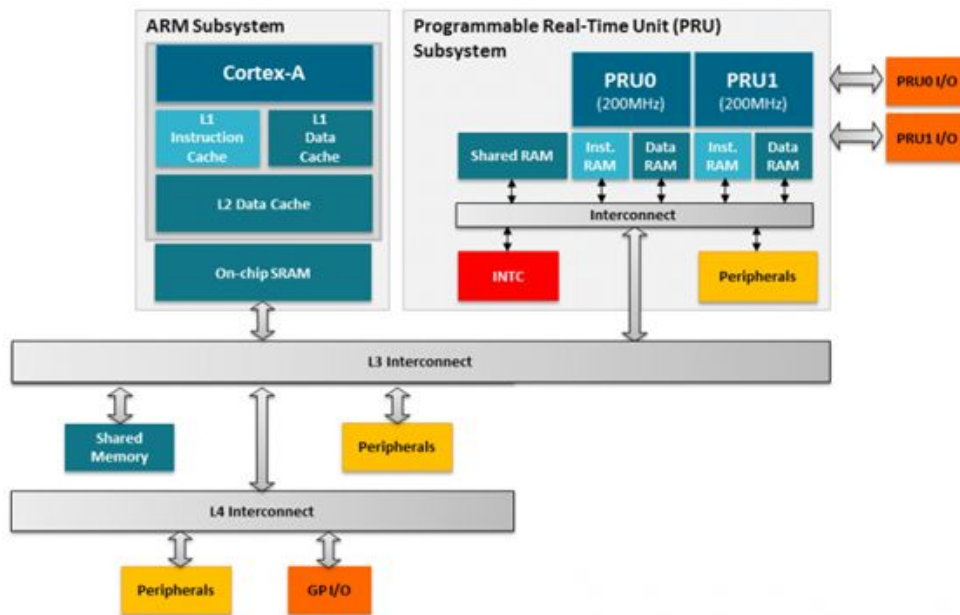
5. DESENVOLVIMENTO

Primeiramente, foi realizado a revisão bibliográfica, definindo conceitos importantes como restrições temporais, que podem ser estrita (*hard* - se não respeitada, ocorre uma falha) ou tolerante (*soft* - há considerações que podem ser cumpridas sem afetar o objetivo do sistema) que são extremos e a atividade a ser realizada está em um ponto entre eles. Outros conceitos como a criticalidade e rigorosidade também foram explorados (a frequência de falhas, e a mitigação dessas, e a os danos que podem ocorrer em vista do não rigor) (SHAW, 2003).

Por outro lado, é importante o conceito do *hardware* empregado para realizar essas tarefas, em seu meio, todas as operações tem o seu tempo de execução definido em ciclos (ex.: 3 ciclos para acessar a DRAM), além de ter acesso direto a I/O, periféricos e interrupções - a figura 1 ilustra esse conceito e compara com a

arquitetura ARM, que possui caches e outros elementos não determinísticos. Com isso em mente, é possível elaborar um circuito base para analisar o tempo de ação após uma entrada, para a PRU e o núcleo Cortex, e realizar a medição com o osciloscópio (BeagleBoard.org Foundation, 2018).

Figura 1: Diagrama de blocos do SoC do BeagleBone Black



Fonte: http://processors.wiki.ti.com/images/c/c1/ARM_PRU_SOC_block_diag.png

6. RESULTADOS PRELIMINARES

Até o presente momento, foi documentada a base teórica (processamento em tempo real e convencional, a placa BeagleBone e seu ambiente de compilação, o Code Composer Studio, inclusive com um código desenvolvido em C para familiarização com o ambiente), e a instalação do Debian 9.3 2018-01-28, RTOS open source.

7. FONTES CONSULTADAS

- BeagleBoard.org Foundation. **PRU-ICSS Resources**. BeagleBoard.org Foundation, 2018. Disponível em: <<http://beagleboard.org/pru>> Acessado em: 25 agosto de 2018
- SHAW, Alan C. **Sistemas e software de tempo real**. Porto Alegre: Bookman, 2003. 240p
- TANENBAUM, Andrew S.; BOS, Herbert. **Sistemas Operacionais Modernos**. 4ª ed. São Paulo: Pearson Education do Brasil, 2016. 864p
- KANT, Krishna. **Computer-Based Industrial Control**. PHI Learning Pvt. Ltd., 2004.