

1º Trabalho Prático

INF01059 Sistemas Embarcados (2020/1)

Grupo 6

Francisco P. Knebel, Gabriel A. Zillmer
Pedro H. Augustin, Rodrigo D. Madruga

Sumário

Motivação

Dispositivos

- ARM-A15
- ADSP-BF522
- Silicon Labs C8051F120
- Análise

Algoritmos

- Ordenação: Quadsort
- Vetor: Algoritmo de média, desvio e minmax
- Sinais: Algoritmo de Goertzel

ARM-A15 no GEM5

ADSP-BF522 no VisualDSP++

C8051F120 (8051) no RKit

Análise dos Resultados

- melhor relação entre custo (\$) e desempenho (s)
- paridade de frequência de operação
- minimização do consumo de energia

Conclusão

Motivação

Explorar as relações de **custo-benefício** nas diferentes arquiteturas e organizações disponíveis para **processadores embarcados**.

Neste trabalho, consideraremos:

- a arquitetura de propósito geral ARM;
- os processadores DSP da família Blackfin;
- e os microcontroladores 8051.

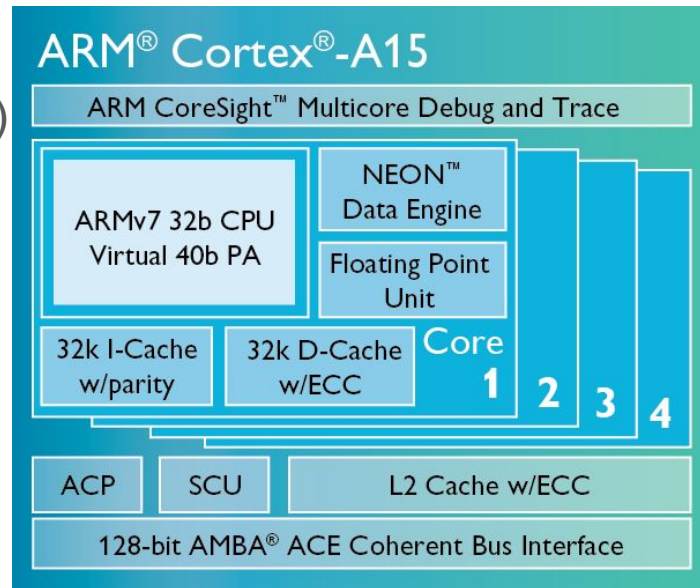
Dispositivos - ARM A15

CPU Clock: 1.0 GHz a 2.5GHz;

Cache: L1 64KB (32KB I-Cache + 32KB D-Cache)

L2 até 4MB

Extensão DSP e SIMD



Dispositivos - ADSP-BF522

CPU Clock 400MHz;

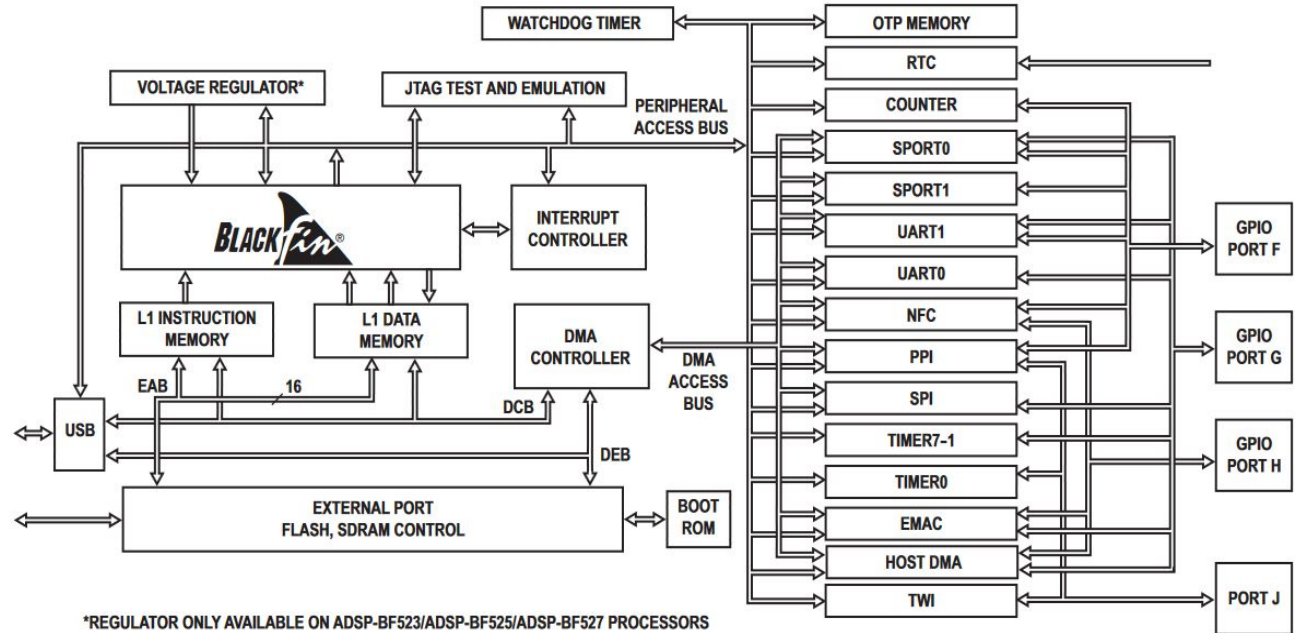
132KB on-chip memory;

2x 16-bit MACs;

2x 40-bit ALU;

4x 8-bit video ALU;

1x 40-bit barrel shifter.



Dispositivos - Silicon Labs C8051F120

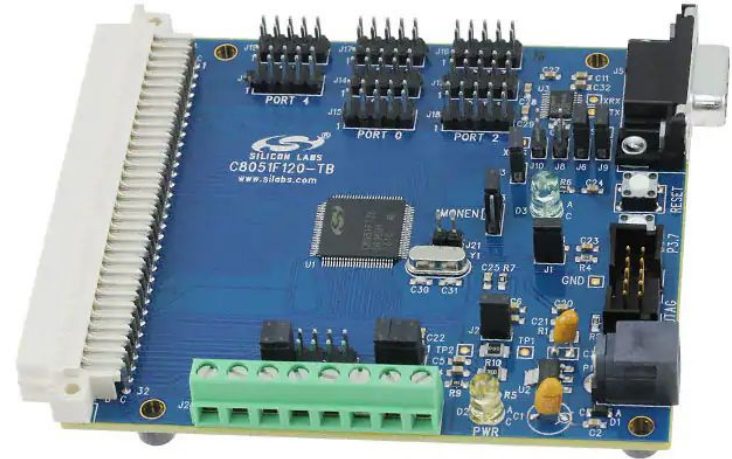
CPU Clock 12 à 33MHz;

8448 bytes internal data RAM (8k + 256);

5x general-purpose 16-bit timers;

2x 12-bit programmable update scheduling DACs;

SPI, SMBus/I2C, and (2) UART interfaces



Dispositivos - Análise

Dispositivo	Custo (USD)	Frequência (MHz)	Potência Média Dissipada (W)
ARM A15	Med: 45.00 Min: 25.00	2000	4.0
ADSP-BF522 (Analog Devices, Inc.)	Med: 8.60	400	0.2
Microcontrolador <i>Silicon Labs</i> C8051F120	Med: 20.00 Min: 19.00	12	4.0

Algoritmos - Ordenação: Quadsort

<https://github.com/scandum/quadsort>

Variação de merge sort, estável, não-recursivo, mais rápido que o quicksort.

Baseado na ideia de “quad swap”. Algoritmos tradicionais usam uma troca binária (verificação se duas variáveis estão ordenadas por meio de uma terceira variável). Quadsort elimina uma comparação para sequências em ordem e adiciona uma se sequência puramente aleatória.

Best	Average	Worst	Stable	Memory
n	$n \log n$	$n \log n$	Sim	n

Algoritmos - Vetor: Algoritmo de média, desvio e minmax

Implementação própria (<https://github.com/FranciscoKnebel/embarcados-t1>)

São calculados a média, valor máximo, valor mínimo, desvio e desvio absoluto médio de um vetor com 65530 valores de 16 bits sem sinal.

Algoritmos - Sinais: Algoritmo de Goertzel

Fonte: <https://stackoverflow.com/questions/8835806/c-c-goertzel-algorithm-with-complex-output-or-magnitudephase>

Técnica aplicada para Transformadas de Fourier em DSPs.

Para um número reduzido de frequências, o Algoritmo de Goertzel é numericamente mais eficiente que o FFT.

ARM-A15 no GEM5

Frequência	Quadsort	Vetor	Goertzel
2000 MHz	22.48 ms	6.42 ms	4.47 ms
1000 MHz	44.96 ms	12.61 ms	8.89 ms

ADSP-BF522 no VisualDSP++

Frequência	Quadsort	Vetor (*/**)	Goertzel*
400 MHz	30,54 ms	20,68 ms	26,13 ms
1000 MHz	12,21 ms	8,25 ms	10,45 ms

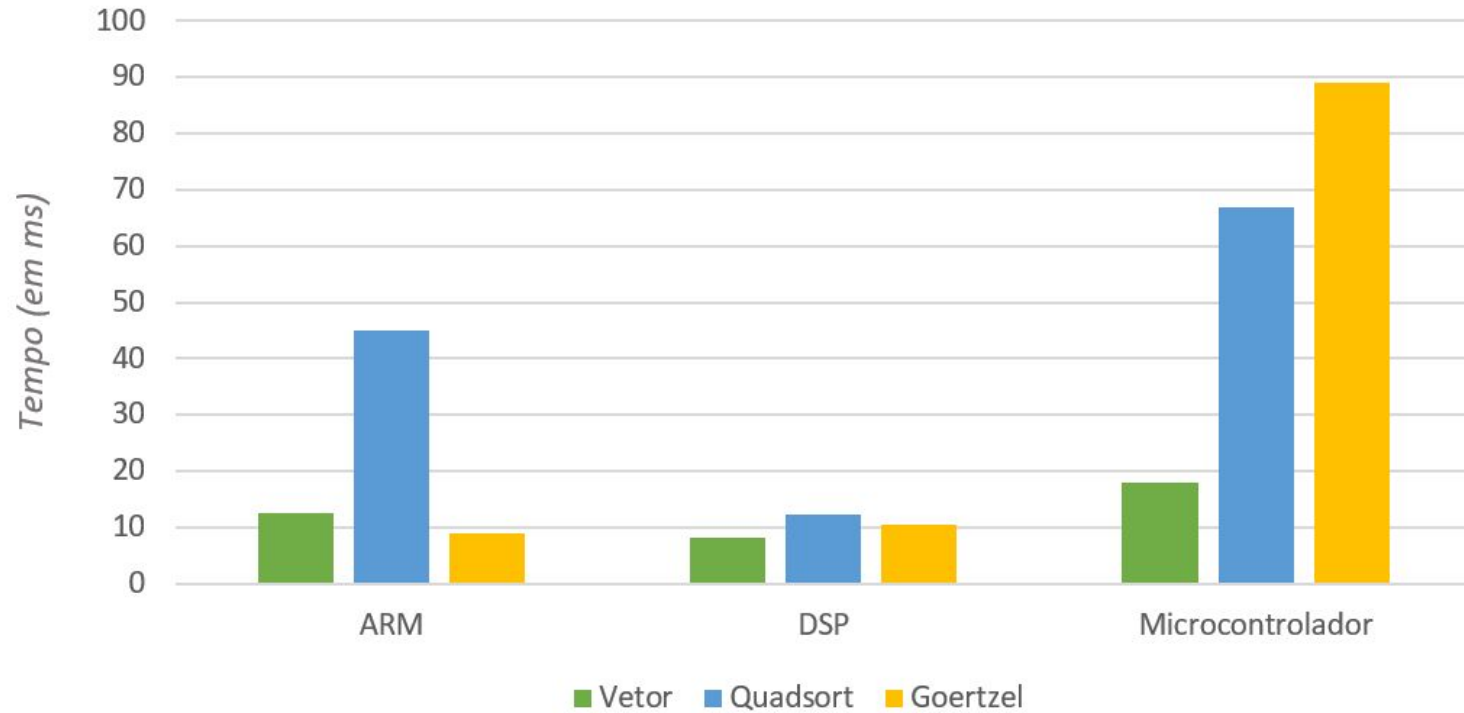
* problemas relacionados a bibliotecas sem suporte na plataforma (complex, sys/time, etc.)

** simulado no VisualDSP++ com um vetor de 7700 elementos. Tempo normalizado para uma simulação com 65530 elementos.

C8051F120 (8051) no RKit

Frequência	Quadsort	Vetor	Goertzel
12 MHz	5590 ms	1512 ms	7413 ms
1000 MHz	67 ms	18 ms	89 ms

Benchmark (a 1GHz)



Análise dos Resultados

melhor relação entre custo (\$) e desempenho (s)

Cenário ideal: A15

Cenário real:

- A15 precisa de motherboard, se torna caro e de difícil implementação;
- DSP é o de menor custo;
- 8051 tem maior suporte e histórico.

Análise dos Resultados

paridade de frequência de operação dos dispositivos

Cenário ideal: DSP se destaca pelos resultados.

Cenário real: DSP realmente se destaca mesmo com frequência nativa.

Análise dos Resultados

minimização do consumo de energia

Cenário ideal: DSP se destaca pelo baixo consumo.

Cenário real:

- DSP realmente se destaca pois:
 - o 8051 selecionado não é destinado para baixo consumo;
 - naturalmente o A15 terá o consumo maior, por ser de propósito geral.

Conclusão

Conforme esperado, o DSP se destacou pelo baixo consumo e baixo preço, justamente por ser um dispositivo de uso específico, em contraste com o A15 e o 8051 que são de uso genérico.

Para cada necessidade, uma ferramenta. O 8051 tem como apelo o suporte de longa data e simplicidade arquitetural e o A15 um alto throughput com suporte a OS de uso geral.

Implementação dos algoritmos disponível em
<https://github.com/FranciscoKnebel/embarcados-t1>

Referências

- https://www.analog.com/media/en/technical-documentation/data-sheets/A_DSP-BF522_BF523_BF524_BF525_BF526_BF527.pdf
- <https://www.silabs.com/documents/public/data-sheets/C8051F12x-13x.pdf>
- <https://developer.arm.com/ip-products/processors/cortex-a/cortex-a15>

1º Trabalho Prático

INF01059 Sistemas Embarcados (2020/1)

Grupo 6

Francisco P. Knebel, Gabriel A. Zillmer
Pedro H. Augustin, Rodrigo D. Madruga