

Sistemas de Hardware e Software

Mutirão C 2018

Igor

19-02-2018

Apresentação

Sistemas de Hardware e Software

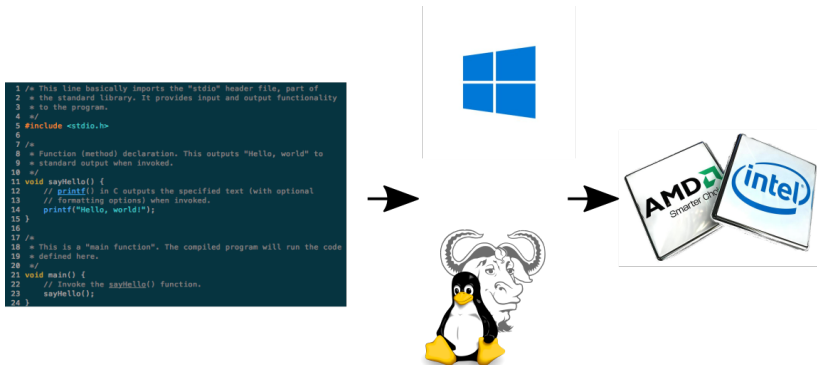


Figure 1: Estudamos a interação entre o programa, o sistema operacional e o hardware

Sistemas de Hardware e Software

Vamos continuar com o embarcado:

1. Abram o projeto HW-SW na solução da aula
2. Rodem o projeto. O quê o botão faz?
3. Localizem a função `processImage(..)` no arquivo `main.c`.

Sistemas de Hardware e Software

```
for (j=1;j<imgW-1;j++){  
    for (i=1;i<imgH-1;i++){  
        val = (int) 4 * imgIn[i][j]  
            - imgIn[i-1][j]  
            - imgIn[i+1][j]  
            - imgIn[i][j-1]  
            - imgIn[i][j+1];  
  
        imgOut[i][j] = abs(val);  
    }  
}
```

Sistemas de Hardware e Software

Vamos inverter a ordem dos loops e contar os tempos

Sistemas de Hardware e Software

Vamos inverter a ordem dos loops e contar os tempos

- ▶ ordem ji: *31.558 us*

- ▶ ordem ij: *26.164 us*

O loop ij é 20% mais eficiente que o loop ji!

Sistemas de Hardware e Software

Vamos inverter a ordem dos loops e contar os tempos

- ▶ ordem *ji*: *31.558 us*

- ▶ ordem *ij*: *26.164 us*

O loop *ij* é 20% mais eficiente que o loop *ji*!

Mudem o clock para 4Mhz e façam o mesmo experimento.

O quê está acontecendo?

As CPUs presentes em PCs (e muitos sistemas embarcados) são otimizadas para acesso **sequencial** usando a memória *cache*.

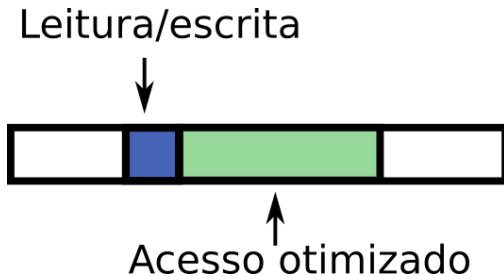


Figure 2: Acesso é otimizado quando acessamos as posições de um vetor em ordem crescente

O cache

O cache é uma memória pequena mas muito rápida usada para acelerar o desempenho de acesso à memória RAM

Cache	
Cache size	16 KB for instruction cache, 16 KB for data cache
Number of sets	256 for instruction cache, 128 for data cache
Number of ways	2 for instruction cache, 4 for data cache
Number of words per cache line	8 words (32 bytes)
ECC on Cache	Embedded

Figure 3: Configuração do cache do νC

O cache

O cache é uma memória pequena mas muito rápida usada para acelerar o desempenho de acesso à memória RAM

Cache	
Cache size	16 KB for instruction cache, 16 KB for data cache
Number of sets	256 for instruction cache, 128 for data cache
Number of ways	2 for instruction cache, 4 for data cache
Number of words per cache line	8 words (32 bytes)
ECC on Cache	Embedded

Figure 3: Configuração do cache do νC

Nossa imagem possui $320 \times 320 = 100kb$. Ela é *pequena* perto do cache do μC (16kb)

Loop interno - ji

```
for (j=1;j<imgW-1;j++){  
    for (i=1;i<imgH-1;i++){  
        val = (int) 4 * imgIn[i][j]  
            - imgIn[i-1][j]  
            - imgIn[i+1][j]  
            - imgIn[i][j-1]  
            - imgIn[i][j+1];  
  
        imgOut[i][j] = abs(val);  
    }  
}
```

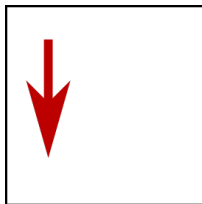


Figure 4:

Loop interno - ij

```
for (i=1;i<imgH-1;i++){  
    for (j=1;j<imgW-1;j++){  
        val = (int) 4 * imgIn[i][j]  
            - imgIn[i-1][j]  
            - imgIn[i+1][j]  
            - imgIn[i][j-1]  
            - imgIn[i][j+1];  
  
        imgOut[i][j] = abs(val);  
    }  
}
```

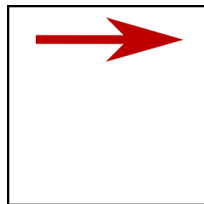


Figure 5:

O quê está acontecendo?

A ordem dos loops que privilegia o acesso sequencial (por linha) tem desempenho superior!

Importante:

1. Temos controle total do microprocessador
2. Nossa imagem é pequena

O quê está acontecendo?

A ordem dos loops que privilegia o acesso sequencial (por linha) tem desempenho superior!

Importante:

1. Temos controle total do microprocessador
2. Nossa imagem é pequena perto do tamanho do cache.
3. Nossos acessos à memória são extremamente locais.
4. Menor consumo de tempo implica em eficiência energética e/ou maior tempo para processamento.

Sistemas de Hardware e Software

Vamos olhar agora para a função que desenha as imagens no LCD.

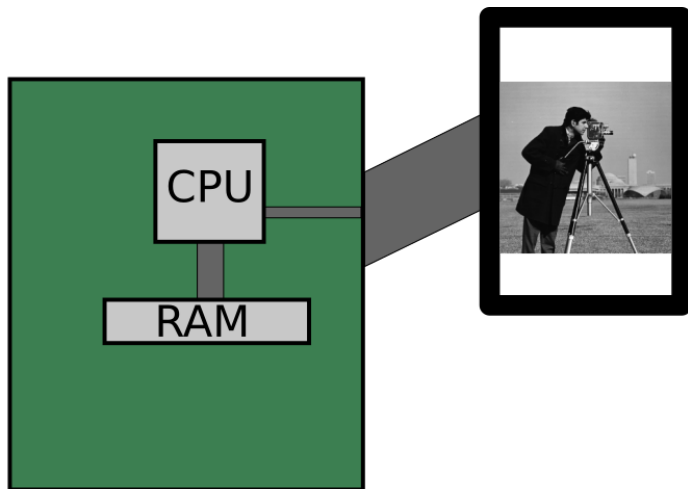
```
void imgShow(ili9488_color_t image[320][320], uint32_t time){  
    ili9488_color_t *imgOutLCD = (ili9488_color_t *) BOARD_SDRAM  
  
    matrixToLCD(image, imgOutLCD, 320, 320);  
    ili9488_draw_pixmap(0, 320/4, 320, 320, imgOutLCD);  
    ...  
}
```

A função `ili9488_draw_pixmap` é provida pelo sistema embarcado e está diretamente ligada ao modo de funcionamento do LCD.

Sistemas de Hardware e Software

Video

Barramentos



■ = Barramento

Figure 6: Componentes do sistema se comunicam através de barramentos

Barramentos

Diversos componentes de um sistema estão ligados por barramentos de diferentes velocidades:

- ▶ Discos
- ▶ Dispositivos USB
 - ▶ Teclado
 - ▶ Mouse
 - ▶ Tela touchscreen
 - ▶ Webcam
- ▶ Interfaces de rede

A interação com estes dispositivos depende de vários fatores que precisam ser levados em conta ao programar um sistema

Sistemas de Hardware e Software

- ▶ Representação de dados na CPU
- ▶ Representação de programas na CPU
 - ▶ Assembly
- ▶ Conceitos de Sistemas Operacionais
 - ▶ Hierarquia de memória
 - ▶ Entrada e saída
 - ▶ Processos e Sinais
 - ▶ Programação concorrente