

Laboratório de Arquitetura e Organização de Computadores I

Prof. Carlos Augusto Guerra Carneiro

“From electrons to asteroids”

for (int i=0; i<nPixels-1;++i){
 pBitmap[i] = (pBitmap[i]+pBitmap[i+1])/2
 break;

```

mov    bl, [ecx]
mov    dl, [ecx+esi*4]
add    ebx, edx
mov    bl, [ecx+2]
mov    dl, [ecx+esi*4+2]
shr    ebx, 2
mov    [ecx+esi*4], ebx

```

0011	10000101	11110000	11001010	01000001	01111001	00110100	11101001
0101	00011000	10111000	11011011	00111000	11101111	00001100	01101100
0100	10001110	01111001	11011100	10110000	01100000	11000000	10101100

gainful
employment
of Maxwell's
equations

$\oint_{\partial V} \mathbf{E} \cdot d\mathbf{A} = \frac{Q(V)}{\epsilon_0}$
 $\oint_{\partial V} \mathbf{B} \cdot d\mathbf{A} = 0$
 $\oint_{\partial S} \mathbf{E} \cdot d\mathbf{l} = -\frac{\partial \Phi_{B,S}}{\partial t}$
 $\oint_{\partial S} \mathbf{B} \cdot d\mathbf{l} = \mu_0 I_S + \mu_0 \epsilon_0 \frac{\partial \Phi_{E,S}}{\partial t}$

Objetivos

- Proporcionar a prática em laboratório de projetos básicos de processadores.
- Proporcionar a prática em laboratório de projetos utilizando linguagem Assembly para processadores.

Considerações

- Colaboração é estimulada, a cópia de trabalho de outro grupo é penalizada.
 - Trabalho copiado é anulado, seja cópia de colega ou cópia de outra fonte qualquer.
- No horário do laboratório de AOC1 trabalhe somente para AOC1.
- É de responsabilidade de TODOS os alunos do grupo terem a cópia do trabalho.
- O aluno deverá enviar toda semana ao final da aula o que produzir.
- Não serão aceitos trabalhos depois do prazo. Submeta na data de entrega até o ponto que tenha conseguido desenvolver.
- Diálogo é fundamental, o trabalho do professor não é só cobrar, é também orientar.
- A orientação é direcionada às dúvidas; vocês tem que perguntar quando tiverem alguma questão.

Tarefas da primeira aula

- module decodificador (input [2:0] entrada, output [7:0] saida);
- module multiplexador (input [15:0] entrada0, entrada1, entrada2, entrada3, input [1:0] controle, output [15:0] saida);
- module registrador (input [15:0] dado, input clock, habilita_escrita, output [15:0] saida);
- module somador_subtrator(input [15:0] entrada0, entrada1, input controle, output [15:0] saida);
-

Tarefas da primeira aula

- Implementação dos módulos apresentados no slide anterior.
- Pesquisar sobre as palavras reservadas do verilog “generate” e “parameter”.
- Adaptar a implementação dos módulos de maneira a utilizar as palavras chaves citadas anteriormente “generate” e “parameter”.

Referências

- PATTERSON, D. A.; HENNESSY, J. L. Organização e projeto de computadores: a interface hardware/software. 3. ed. Rio de Janeiro: Campus, 2005 (Apêndice B, PDF disponível no moodle).
- MANO, M. Morris; CILETTI, Michael D. Digital Design. 4. Ed. Upper Saddle River, N. J.: Prentice-Hall, 2007 (Seções selecionadas disponíveis no moodle).
- <http://en.wikipedia.org/wiki/Verilog>