



## **Laboratório de Circuitos Lógicos (CIC0231)** **- Projeto Final -**

### **ZEPTOPROCESSADOR-II DE 16 BITS**

#### **1. INTRODUÇÃO:**

Os processadores são os circuitos digitais mais complexos que temos hoje em dia, principalmente os de alto desempenho. O conceito por trás de um processador envolve a leitura de uma instrução de uma memória, decodificar a instrução, executá-la e voltar a ler a próxima instrução. O presente projeto desafia o aluno a desenvolver um processador programável simples, o ZeptoProcessador-II, com apenas 8 instruções.

#### **2. OBJETIVOS**

Apresentar ao aluno os conceitos básicos que envolvem o projeto de processadores usando circuitos digitais. O aluno deverá implementar um ZeptoProcessador-II capaz de executar um programa com as seguintes instruções:

- 1) addi: Adição com Imediato
- 2) subi: Subtração com Imediato
- 3) andi: And bitwise com Imediato
- 4) ori: Or bitwise com Imediato
- 5) xori: Xor bitwise com Imediato
- 6) beq: Salto Condicional
- 7) bleu: Salto Condicional (Unsigned)
- 8) bles: Salto condicional (Signed)

E executar programas com até 4096 instruções.

#### **3. METODOLOGIA**

Todo o sistema digital deve ser implementado como um projeto no software de simulação Deeds, com a interação com o usuário feita através do uso de uma memória ROM para armazenar o programa, botão de reset e sinal de clock que pode ser manual.

O projeto deve ser baseado em interligação configurável de blocos funcionais de acordo com a instrução a ser executada.

O ZeptoProcessador-II gerado deverá ser testado pela execução de uma bateria de programas de testes.



#### 4. ESPECIFICAÇÕES DO SISTEMA DIGITAL

O ZeptoProcessador-II deve possuir:

- i) Uma memória de instruções capaz de armazenar até 4096 instruções de 32 bits cada uma.
- ii) Um registrador PC de 16 bits que indica o endereço na memória da instrução executada.
- iii) Um banco de 16 registradores R0...R15 que podem armazenar números de 16 bits. O registrador R0 possui o valor fixo 0.
- iv) Uma Unidade Lógico-Aritmética de 16 bits.

As instruções possuem 32 bits de tamanho codificados com os seguintes campos:

|    |    |    |    |    |    |   |   |    |   |   |   |        |   |   |   |
|----|----|----|----|----|----|---|---|----|---|---|---|--------|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7  | 6 | 5 | 4 | 3      | 2 | 1 | 0 |
| Rd |    |    |    | Ra |    |   |   | Rb |   |   |   | Opcode |   |   |   |

|          |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 31       | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Imediato |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |

**OpCode:** são 4 bits que definem a instrução a ser executada

| OpCode | Mnemônico            | Nome                     | Operação   |
|--------|----------------------|--------------------------|--|
| 0000   | addi Rd, Ra, Rb, Imm | Soma com imediato        | $Rd = Ra + Rb + Imm$   |
| 0001   | subi Rd, Ra, Rb, Imm | Subtração com imediato   | $Rd = Ra - Rb - Imm$   |
| 0010   | andi Rd, Ra, Rb, Imm | And bitwise com imediato | $Rd = Ra \& Rb \& Imm$   |
| 0011   | ori Rd, Ra, Rb, Imm  | Or bitwise com imediato  | $Rd = Ra   Rb   Imm$   |
| 0100   | xori Rd, Ra, Rb, Imm | Xor bitwise com imediato | $Rd = Ra \oplus Rb \oplus Imm$   |
| 0101   | beq Ra, Rb, Imm      | Salto se igual           | $Ra == Rb ? PC = PC + Imm : PC = PC + 1$                                     |
| 0110   | bleu Ra, Rb, Imm     | Salto se menor ou igual  | $Ra \leq Rb ? PC = PC + Imm : PC = PC + 1$<br>Ra e Rb considerados sem sinal |
| 0111   | bles Ra, Rb, Imm     | Salto se menor ou igual  | $Ra \leq Rb ? PC = PC + Imm : PC = PC + 1$<br>Ra e Rb considerados com sinal |

**Ra:** 4 bits que definem o registrador Ra (de R0 a R15)

**Rb:** 4 bits que definem o registrador Rb (de R0 a R15)

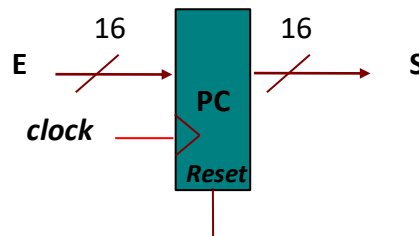
**Imediato:** Número de 16 bits



Geralmente, os processadores são compostos dos seguintes elementos básicos que devem ser projetados pelo aluno:

### 1) Registrador PC

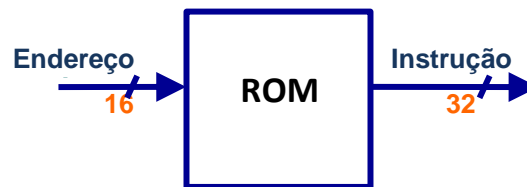
Trata-se de um registrador (use o modelo do Deeds) que armazena o endereço da instrução a ser executada. O número de bits deste registrador depende da quantidade de memória de instruções utilizada.



O dado de entrada E é escrito no registrador PC quando ocorrer a subida da borda de clock. Caso Reset=1, ao vir a borda de subida de clock o registrador é resetado, isto é, PC=0x0000.

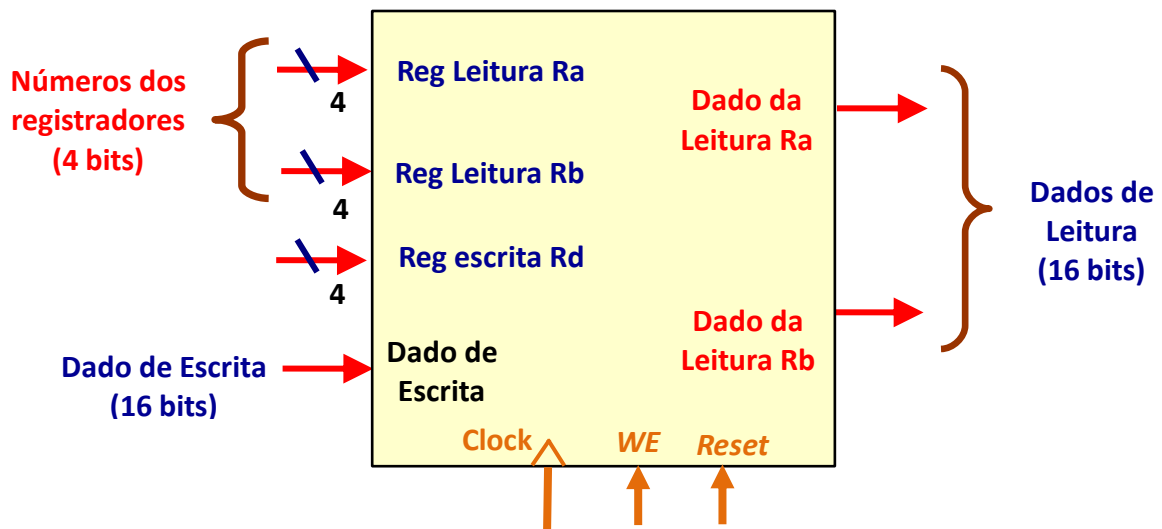
### 2) Memória de Instruções

Consiste de uma memória do tipo ROM (use o modelo do Deeds) que deve ser programada com o código binário do programa do usuário. Dica: Use 2 módulos de ROM de 4K x 16 bits.



### 3) Banco de Registradores

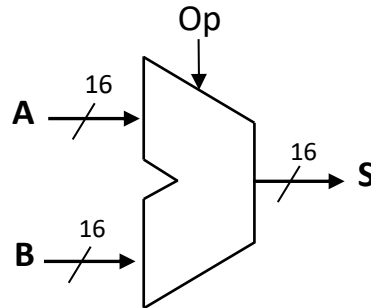
Para a execução das instruções tem-se disponível um banco de 16 registradores de 16 bits cada um. O Banco de Registradores deve possibilitar a leitura de 2 registradores Ra e Rb simultaneamente e a escrita no registrador Rd se WE=1 e vier a borda de subida do clock. Caso Reset=1 na borda de subida do clock, todos os 16 registradores são resetados.





#### 4) Unidade Lógico-Aritmética

Circuito digital (pode usar a ULA que tem pronta no Deeds) capaz de realizar as operações de soma e subtração de números de 16 bits em complemento de 2. A operação deve ser selecionável por um sinal Op de 3 bits (000-add, 001-sub, 010-and, 011-or, 100-xor).



#### 5) Comparador == e <= com sinal e sem sinal

Projete um circuito combinacional que receba 2 números A e B de 16 bits e indique em dois sinais de saída se  $A \leq B$  e se  $A == B$ . Porém é necessário identificar se os números devem ser considerados com sinal ou sem sinal, assim acrescente um sinal de controle SU, onde  $SU=0$  indica comparação sem sinal e  $SU=1$  indica comparação com sinal.

#### 6) Bloco de Controle

Projete um circuito combinacional que recebe na entrada o OpCode da instrução a ser executada e gere todos os sinais de controle necessários (WE do BR, Op da ULA, SU do Comparador e seleção de multiplexadores).

#### 7) Sinais de Monitoramento

Acrescente ao seu processador quatro displays de 7 segmentos hexadecimais que monitore os sinais: PC, Instrução, Ra e Rb.

### 5. EXEMPLOS DE PROGRAMAS

Em todos os programas exemplos aqui apresentados é pressuposto que o ZeptoProcessador-II inicia resetado (PC=0x0000 e Registradores=0x0000)

1) R1 = Contador de -16 a 16

| Endereço        | Código hexadecimal | Instrução        | Comentário          |
|-----------------|--------------------|------------------|---------------------|
| 0x0000          | 0xFFFF 1001        | subi R1,R0,R0,16 | R1=-16              |
| 0x0001          | 0x0010 2000        | addi R2,R0,R0,16 | R2=16               |
| 0x0002          | 0x0001 2201        | subi R2,R2,R0,1  | R2=R2-1 devido ao = |
| 0x0003    Loop: | 0x0001 1100        | addi R1,R1,R0,1  | R1=R1+1             |
| 0x0004          | 0xFFFF 0127        | jles R1,R2,-1    | R0<=R1? Loop : Next |
| 0x0005    Fim:  | 0x0000 0115        | beq R1,R1,0      | J Fim               |



- a. Filme o funcionamento
- b. Desenhe o diagrama temporal da execução

2)  $R3 = \text{Soma dos números ímpares de } 0 \text{ a } 15$

| Endereço        | Código hexadecimal | Instrução        | Comentário            |
|-----------------|--------------------|------------------|-----------------------|
| 0x0000          | 0x0001 1000        | addi R1,R0,R0,1  | R1=1 primeiro ímpar   |
| 0x0001          | 0x000F 2000        | addi R2,R0,R0,15 | R2=15 valor final     |
| 0x0002          | 0x0001 2200        | addi R2,R2,R0,1  | R2=R2+1 (devido ao =) |
| 0x0003    Loop: | 0x0004 0216        | jleu R2,R1,4     | R2<=R1 ? Fim : Next   |
| 0x0004          | 0x0000 3310        | addi R3,R3,R1,0  | R3=R3+R1 somatório    |
| 0x0005          | 0x0002 1100        | addi R1,R1,R0,2  | R1=R1+2 próximo ímpar |
| 0x0006          | 0xFFFD 0005        | beq R0,R0,-3     | J Loop                |
| 0x0007    Fim:  | 0x0000 0335        | beq R3,R3,0      | J Fim                 |

- a. Filme o funcionamento
- b. Desenhe o diagrama temporal da execução

**Responda:**

Qual a maior frequência de clock que seu ZeptoProcessador consegue executar corretamente todos os programas? O que limita?

**Para os seguintes itens:**

Elabore os algoritmos e implemente no ZeptoProcessador, onde os programas iniciam com:

| Endereço | Código hexadecimal | Instrução       | Comentário          |
|----------|--------------------|-----------------|---------------------|
| 0x0000   | 0x0000 1000        | addi R1,R0,R0,0 | R1 = 0    Resultado |
| 0x0001   | 0xFFFF 2000        | addi R2,R0,R0,X | R2 = X              |
| 0x0002   | 0/YYYY 3000        | addi R3,R0,R0,Y | R3 = Y              |
| 0x0003   | ...                | ...             | ...                 |
| 0x0004   | ...                | ...             | ...                 |
| 0xZZZZ   | 0x0000 0115        | beq R1,R1,0     | Mostra o resultado  |

- 3)  $R1 = R2 \times R3$ : Multiplicação de dois números sem sinal ( $R2 < 256$  e  $R3 < 256$ )
  - a. Filme o funcionamento
  - b. Desenhe o diagrama temporal da execução na maior frequência
- 4)  $R1 = R2 \times R3$ : Multiplicação de dois números com sinal ( $-181 < R2 < 181$  e  $-181 < R3 < 181$ )
  - a. Filme o funcionamento
  - b. Desenhe o diagrama temporal da execução na maior frequência



- 5)  $R1 = R2 / R3$ : Divisão inteira de dois números sem sinal ( $R2 < 65536$  e  $R3 < 65536$ )
  - a. Filme o funcionamento
  - b. Desenhe o diagrama temporal da execução na maior frequência
- 6)  $R1 = R2 \% R3$ : Resto da divisão inteira de dois números sem sinal ( $R2 < 65536$  e  $R3 < 65536$ )
  - a. Filme o funcionamento
  - b. Desenhe o diagrama temporal da execução na maior frequência
- 7) Execute o programa dado pelo professor no Aprender3 no seu processador.
  - a. Filme o funcionamento
  - b. Desenhe o diagrama temporal da execução na maior frequência

## 6. AVALIAÇÃO E CRONOGRAMA

Avaliação do protótipo será feita através do Relatório e por programas de testes elaborados pelo professor e monitores.

De acordo com o cronograma definido no Plano de Aulas:

- a) No dia 30/04/2021 deve ser entregue pelo Aprender3 o relatório do pré-projeto em formato .pdf com o nome 'Pre-Projeto\_A14.pdf', contendo a proposta de solução do projeto (Diagrama de blocos ou esquemático), atividades já realizadas e atividades ainda por fazer para a conclusão do projeto.
- b) No dia 21/05/2021 deve ser entregue pelo Aprender3 um arquivo com o nome 'Projeto\_A14.zip' contendo todos os arquivos necessários para a execução do ZeptoProcessador-II no Deeds, programas (conteúdos da memória em arquivos .drs) e o .pdf do Relatório no formato usual.

## 7. SUMÁRIO

Este projeto visa aplicar os conhecimentos adquiridos ao longo das disciplinas Circuitos Lógicos e Laboratório de Circuitos Lógicos para o desenvolvimento de um sistema digital programável, o ZeptoProcessador-II de 8 instruções.

## 8. EQUIPAMENTOS E MATERIAL

- ~~Kit de Desenvolvimento DE2~~
- ~~Programa Quartus-II v.13.0~~
- ~~1 Mini-teclado matricial 4x4 (Hu infinito R\$ 6,90)~~
- Software Deeds