

# Processador PoliLegv8

## Processador Monociclo (2/3)

Edson S. Gomi

PCS - Departamento de Engenharia de Computação e Sistemas Digitais  
Escola Politécnica da Universidade de São Paulo

Outubro, 2020

# Agenda

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

- 1 Implementação do processador monociclo
- 2 Controle da ULA
- 3 Unidade de Controle Principal
- 4 Operação do Fluxo de Dados
- 5 Função Combinatória da Unidade de Controle Principal
- 6 Extensão da Unidade de Controle Principal para a instrução de salto incondicional B

# Instruções do PoliLEG que serão implementadas

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

- O PoliLEG terá um subconjunto das instruções do ARMv8 (LEGv8 do Hennessy e Patterson):
  - Load Register Unscaled (LDUR)
    - LDUR X1, [X2,40] :  $X1 = \text{Memory}[X2+40]$
  - Store Register Unscaled (STUR)
    - STUR X1, [X2,40] :  $\text{Memory}[X2+40] = X1$
  - Instruções lógico-aritméticas ADD, SUB, AND e ORR
    - ADD X1, X2, X3 :  $X1 = X2+X3$
  - Compare and Branch Zero (CBZ)
    - CBZ X1, 25 : if ( $X1==0$ ) goto PC+100
  - Branch (B)
    - B 2500 : goto PC+10000

# Opcodes das instruções

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

| Instrução | Opcode      | Tamanho do<br>Opcode (bits) | Formato    |
|-----------|-------------|-----------------------------|------------|
| LDUR      | 11111000010 | 11                          | Formato-D  |
| STUR      | 11111000000 | 11                          | Formato-D  |
| ADD       | 10001011000 | 11                          | Formato-R  |
| SUB       | 11001011000 | 11                          | Formato-R  |
| AND       | 10001010000 | 11                          | Formato-R  |
| ORR       | 10101010000 | 11                          | Formato-R  |
| CBZ       | 10110100    | 8                           | Formato-CB |
| B         | 000101      | 6                           | Formato-B  |

# Formatos das instruções

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

| Name       | Fields |              |              |             |        |        |        | Comments                                |
|------------|--------|--------------|--------------|-------------|--------|--------|--------|---|
| Field size |        | 6 to 11 bits | 5 to 10 bits | 5 or 4 bits | 2 bits | 5 bits | 5 bits | All LEGv8 instructions are 32 bits long |
| R-format   | R      | opcode       | Rm           | shamt       |        | Rn     | Rd     | Arithmetic instruction format           |
| I-format   | I      | opcode       | immediate    |             |        | Rn     | Rd     | Immediate format                        |
| D-format   | D      | opcode       | address      |             | op2    | Rn     | Rt     | Data transfer format                    |
| B-format   | B      | opcode       | address      |             |        |        |        | Unconditional Branch format             |
| CB-format  | CB     | opcode       | address      |             |        |        | Rt     | Conditional Branch format               |
| IW-format  | IW     | opcode       | immediate    |             |        |        | Rd     | Wide Immediate format                   |

- Formato-D para LDUR e STUR
- Formato-R para ADD, SUB, AND e ORR
- Formato-CB para CBZ
- Formato-B para B

# Formatos dos Campos das Instruções

## Implementação do processador monociclo

### Controle da ULA

#### Unidade de Controle Principal

#### Operação do Fluxo de Dados

#### Função Combinatória da Unidade de Controle Principal

#### Extensão da Unidade de Controle Principal para a instrução de salto incondicional B

|               |        |       |       |     |     |
|---------------|--------|-------|-------|-----|-----|
| Field         | opcode | Rm    | shamt | Rn  | Rd  |
| Bit positions | 31:21  | 20:16 | 15:10 | 9:5 | 4:0 |

a. R-type instruction

|               |              |         |       |     |     |
|---------------|--------------|---------|-------|-----|-----|
| Field         | 1986 or 1984 | address | 0     | Rn  | Rt  |
| Bit positions | 31:21        | 20:12   | 11:10 | 9:5 | 4:0 |

b. Load or store instruction

|               |       |         |     |
|---------------|-------|---------|-----|
| Field         | 180   | address | Rt  |
| Bit positions | 31:26 | 23:5    | 4:0 |

c. Conditional branch instruction

# Visão Geral do Fluxo de Dados do PoliLEG

## Implementação do processador monociclo

### Controle da ULA

### Unidade de Controle Principal

### Operação do Fluxo de Dados

### Função Combinatória da Unidade de Controle Principal

### Extensão da Unidade de Controle Principal para a instrução de salto incondicional B

- As unidades funcionais do Fluxo de Dados proposto para o PoliLEG são:
  - 1 Memórias, para instruções e dados
  - 2 Registrador, para PC
  - 3 Banco de registradores
  - 4 ULA
  - 5 Multiplexadores
  - 6 Sign-Extension
- Como o requisito é que as instruções executem em 1 ciclo de relógio, as unidades funcionais são duplicadas nos casos em que houver necessidade de uso simultâneo.

# Visão Geral do Fluxo de Dados do PoliLEG

## Implementação do processador monociclo

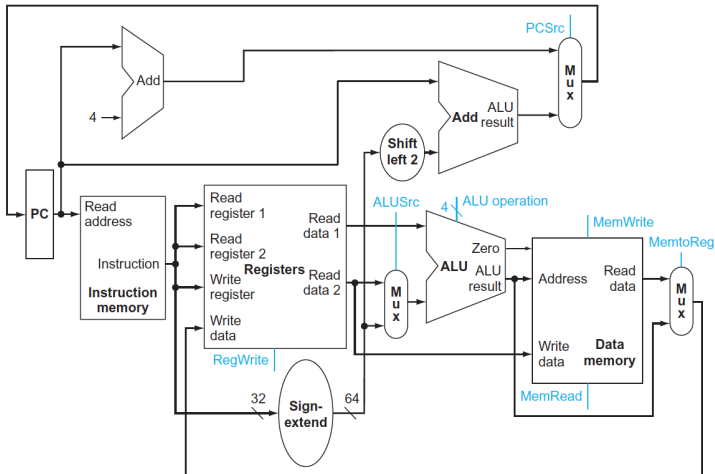
### Controle da ULA

Unidade de  
Controle  
Principal

## Operação do Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B





# ALU Control

## Implementação do processador monociclo

## Controle da ULA

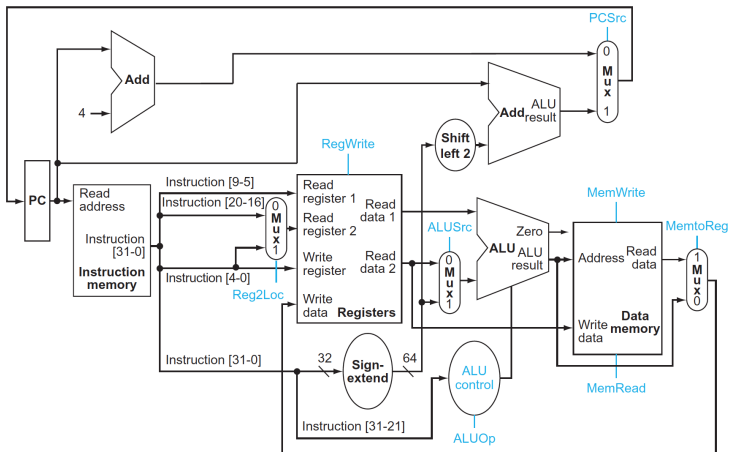
Unidade de  
Controle  
Principal

## Operação do Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

O primeiro controle a ser implementado será o ALU Control, que gera o ALU Operation:



# Fluxo de Dados da ULA de 1 bit

Implementação  
do processador  
monociclo

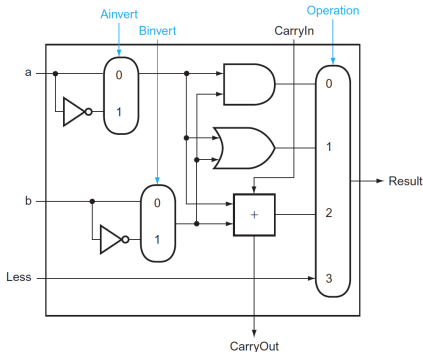
Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B



- O ALU Operation é formado pelos sinais *Ainvert* (1 bit), *Binvert* (1 bit) e *Operation* (2 bits)
- A ULA de 64 bits é construída com 63 cópias da ULA de 1 bit e uma ULA com detecção de *overflow* para o bit mais significativo

# Sinal de controle ALU Operation

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

| ALU control lines | Function     |
|-------------------|--------------|
| 0000              | AND          |
| 0001              | OR           |
| 0010              | add          |
| 0110              | subtract     |
| 0111              | pass input b |
| 1100              | NOR          |

Os 4 sinais de controle da ULA formam o ALU Operation e são representados nesta tabela no seguinte formato:  
<Ainvert, Binvert, Operation1, Operation0>

# Sinal de Controle (ALUOp)

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

| Instruction | ALUOp | Instruction operation      | Opcode field | Desired ALU action | ALU control input |
|-------------|-------|----------------------------|--------------|--------------------|-------------------|
| LDUR        | 00    | load register              | XXXXXXXXXX   | add                | 0010              |
| STUR        | 00    | store register             | XXXXXXXXXX   | add                | 0010              |
| CBZ         | 01    | compare and branch on zero | XXXXXXXXXX   | pass input b       | 0111              |
| R-type      | 10    | ADD                        | 10001011000  | add                | 0010              |
| R-type      | 10    | SUB                        | 11001011000  | subtract           | 0110              |
| R-type      | 10    | AND                        | 10001010000  | AND                | 0000              |
| R-type      | 10    | ORR                        | 10101010000  | OR                 | 0001              |

- O sinal de controle Operation é gerado a partir do opcode e do sinal controle de 2 bits ALUOp:
- ALUOp indica se a operação a ser executada deve ser add (00) para LDUR e STUR, passar a entrada b (01) para CBZ, ou ser determinada pela operação codificada em opcode (10).

# Tabela Verdade do sinal ALUOp

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

ALUOp é gerado a partir dos bits 31-21 de opcode:

| ALUOp  |        | Opcode field |       |       |       |       |       |       |       |       |       |       | Operation |
|--------|--------|--------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-----------|
| ALUOp1 | ALUOp0 | I[31]        | I[30] | I[29] | I[28] | I[27] | I[26] | I[25] | I[24] | I[23] | I[22] | I[21] |           |
| 0      | 0      | X            | X     | X     | X     | X     | X     | X     | X     | X     | X     | X     | 0010      |
| X      | 1      | X            | X     | X     | X     | X     | X     | X     | X     | X     | X     | X     | 0111      |
| 1      | X      | 1            | 0     | 0     | 0     | 1     | 0     | 1     | 1     | 0     | 0     | 0     | 0010      |
| 1      | X      | 1            | 1     | 0     | 0     | 1     | 0     | 1     | 1     | 0     | 0     | 0     | 0110      |
| 1      | X      | 1            | 0     | 0     | 0     | 1     | 0     | 1     | 0     | 0     | 0     | 0     | 0000      |
| 1      | X      | 1            | 0     | 1     | 0     | 1     | 0     | 1     | 0     | 0     | 0     | 0     | 0001      |

| ALUOp | Instrução         | Operação da ULA            |
|-------|-------------------|----------------------------|
| 00    | LDUR e STUR       | add                        |
| 01    | CBZ               | passar a entrada b         |
| 10    | lógico-aritmética | determinada pela instrução |

# Unidade de Controle Principal

Implementação  
do processador  
monociclo

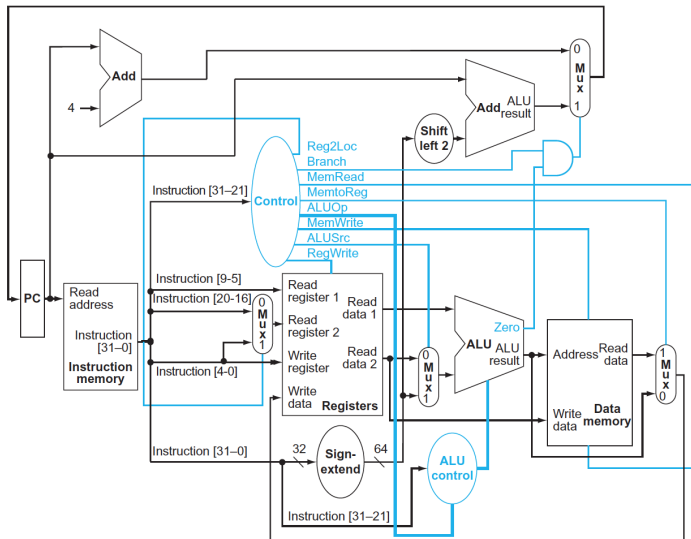
Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B



# Configuração dos sinais de controle

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

A configuração dos sinais de controle é completamente determinada pelo campo opcode da instrução:

| Instruction | Reg2Loc | ALUSrc | MemtoReg | RegWrite | MemRead | MemWrite | Branch | ALUOp1 | ALUOp0 |
|-------------|---------|--------|----------|----------|---------|----------|--------|--------|--------|
| R-format    | 0       | 0      | 0        | 1        | 0       | 0        | 0      | 1      | 0      |
| LDUR        | X       | 1      | 1        | 1        | 1       | 0        | 0      | 0      | 0      |
| STUR        | 1       | 1      | X        | 0        | 0       | 1        | 0      | 0      | 0      |
| CBZ         | 1       | 0      | X        | 0        | 0       | 0        | 1      | 0      | 1      |

# A função dos 7 sinais de controle

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

A Unidade de Controle Principal gera todos os sinais de controle a partir de opcode, exceto PCSrc:

| Signal name | Effect when deasserted  | Effect when asserted  |
|-------------|---|---|
| Reg2Loc     | The register number for Read register 2 comes from the Rm field (bits 20:16).       | The register number for Read register 2 comes from the Rt field (bits 4:0).                             |
| RegWrite    | None.   | The register on the Write register input is written with the value on the Write data input.             |
| ALUSrc      | The second ALU operand comes from the second register file output (Read data 2).    | The second ALU operand is the sign-extended, lower 16 bits of the instruction.                          |
| PCSrc       | The PC is replaced by the output of the adder that computes the value of $PC + 4$ . | The PC is replaced by the output of the adder that computes the branch target.                          |
| MemRead     | None.   | Data memory contents designated by the address input are put on the Read data output.                   |
| MemWrite    | None.   | Data memory contents designated by the address input are replaced by the value on the Write data input. |
| MemtoReg    | The value fed to the register Write data input comes from the ALU.                  | The value fed to the register Write data input comes from the data memory.                              |



# Sinal PCSrc

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

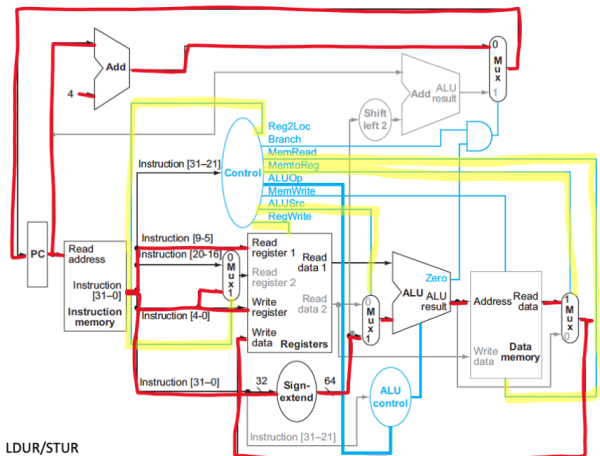
Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

- O sinal PCSrc seleciona qual será o próximo valor de PC e não é gerado diretamente do opcode.
- Este sinal é asserted se a instrução for CBZ e o sinal Zero da ULA também é asserted.
- Caso contrário, PC será  $PC + 4$ .

# Instrução Tipo-R (ADD/SUB/AND/ORR)

Funcionamento do ALU Control para instruções do tipo-R:



# Instrução Tipo-R (ADD/SUB/AND/ORR)

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

Funcionamento do controle do Fluxo de Dados para instruções lógico-aritméticas, como por exemplo ADD X1, X2, X3:

- A instrução é lida da memória de instruções e PC é incrementado ( $PC = PC + 4$ )
- Os registradores X2 e X3 são lidos do banco de registradores
- A Unidade de Controle Principal configura as linhas de controle
- A ULA calcula o resultado
- O resultado é escrito em X1, no banco de registradores

# Operação de instrução de load (LDUR) ou store (STUR)

Implementação  
do processador  
monociclo

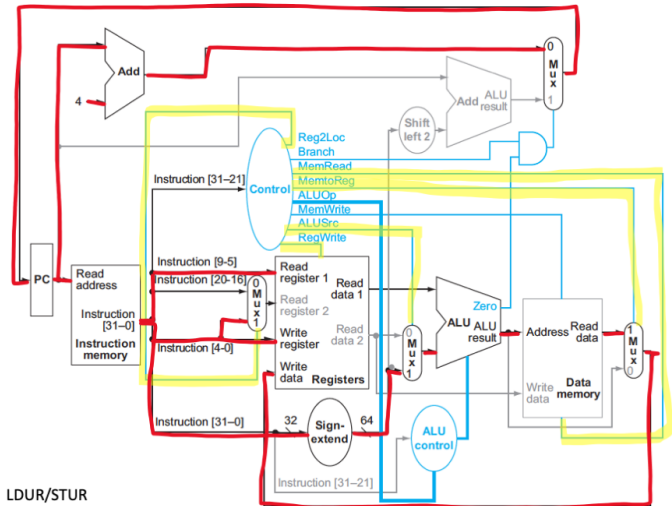
Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de salto  
incondicional B



LDUR/STUR

# Operação de instrução de load (LDUR) ou store (STUR)

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

Funcionamento do controle do Fluxo de Dados para a instrução LDUR, como por exemplo LDUR X1, [X2,offset]:

- A instrução é lida da memória de instruções e PC é incrementado ( $PC = PC + 4$ )
- O registrador X2 é lido do banco de registradores
- A ULA calcula  $X2 + \text{sinal-estendido}(\text{offset})$ . O offset é obtido do campo de 9 bits correspondente na instrução
- O resultado da ULA é usado como endereço para a memória de dados
- O dado é lido da memória de dados para o registrador X1

# Operação da instrução CBZ

Implementação  
do processador  
monociclo

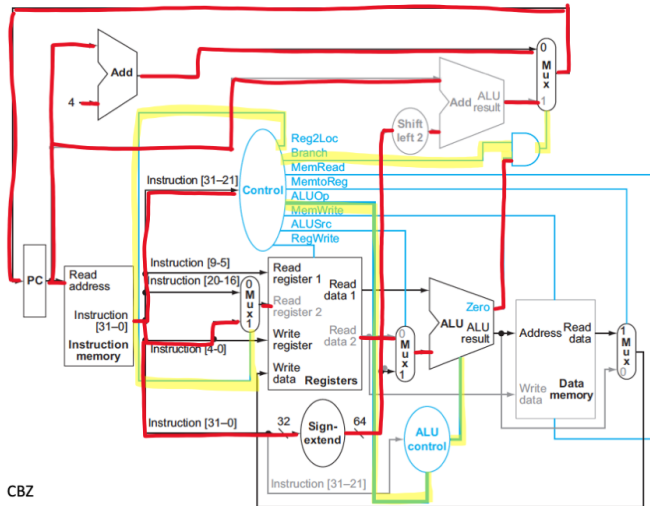
Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de salto  
incondicional B



CBZ

# Operação da instrução CBZ

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

Funcionamento do controle do Fluxo de Dados para a instrução CBZ, como por exemplo CBZ X1, offset:

- A instrução é lida da memória de instruções e PC é incrementado ( $PC = PC + 4$ )
- O registrador X1 é lido do banco de registradores, usando os bits 4:0 (Rt) da instrução
- A ULA passa o dado lido do banco de registradores
- O valor de PC é somado com sinal-estendido(offset). Os 19 bits do campo offset da instrução são deslocados de 2 bits para a esquerda. O resultado da soma é o endereço do salto
- O sinal Zero da ULA decide qual endereço será escrito no PC

# Tabela Verdade da Unidade de Controle Principal

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

| Input or output | Signal name | R-format | LDUR | STUR | CBZ |
|-----------------|-------------|----------|------|------|-----|
| Inputs          | I[31]       | 1        | 1    | 1    | 1   |
|                 | I[30]       | X        | 1    | 1    | 0   |
|                 | I[29]       | X        | 1    | 1    | 1   |
|                 | I[28]       | 0        | 1    | 1    | 1   |
|                 | I[27]       | 1        | 1    | 1    | 0   |
|                 | I[26]       | 0        | 0    | 0    | 1   |
|                 | I[25]       | 1        | 0    | 0    | 0   |
|                 | I[24]       | X        | 0    | 0    | 0   |
|                 | I[23]       | 0        | 0    | 0    | X   |
|                 | I[22]       | 0        | 1    | 0    | X   |
|                 | I[21]       | 0        | 0    | 0    | X   |
| Outputs         | Reg2Loc     | 0        | X    | 1    | 1   |
|                 | ALUSrc      | 0        | 1    | 1    | 0   |
|                 | MemtoReg    | 0        | 1    | X    | X   |
|                 | RegWrite    | 1        | 1    | 0    | 0   |
|                 | MemRead     | 0        | 1    | 0    | 0   |
|                 | MemWrite    | 0        | 0    | 1    | 0   |
|                 | Branch      | 0        | 0    | 0    | 1   |
|                 | ALUOp1      | 1        | 0    | 0    | 0   |
|                 | ALUOp0      | 0        | 0    | 0    | 1   |



# Operação da instrução B (Salto Incondicional)

Implementação  
do processador  
monociclo

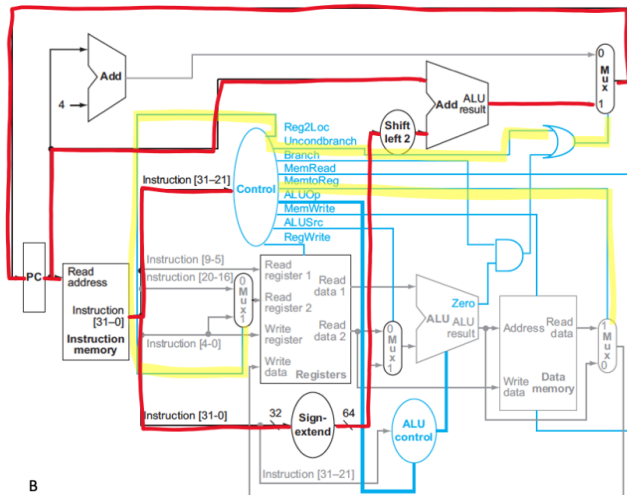
Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de salto  
incondicional B



# Operação da instrução B (Salto Incondicional)

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B

Funcionamento do controle do Fluxo de Dados para a instrução B, como por exemplo B offset:

- O Fluxo de Dados do processador é estendido para incluir o sinal de controle Uncondbranch
- O endereço do salto é calculado somando PC com o sinal-estendido(offset) usando os 26 bits do campo de endereço imediato da instrução.
- Uma lógica adicional OR é usada para fazer com que o novo valor de PC seja o resultado desta soma.

Obrigado!



Universidade de São Paulo



DEPARTAMENTO DE ENGENHARIA DE  
COMPUTAÇÃO E SISTEMAS DIGITAIS

PCS

# Referências

Implementação  
do processador  
monociclo

Controle da ULA

Unidade de  
Controle  
Principal

Operação do  
Fluxo de Dados

Função  
Combinatória da  
Unidade de  
Controle  
Principal

Extensão da  
Unidade de  
Controle  
Principal para a  
instrução de  
salto  
incondicional B



D. Patterson and J. Hennessy.

*Computer Organization and Design ARM Edition: The Hardware Software Interface.*

The Morgan Kaufmann Series in Computer Architecture and Design. Elsevier Science, 2016.