

114. lui \$t0, 0x00A8
mtcl \$t0, \$f8

$$\$f8 = 0x00A80000 = \underbrace{1}_{S} \underbrace{00000000}_{E} \underbrace{10101000000000000000}_{f}$$

$$E = 1000000_2 = 129_{10} \Rightarrow Ex_4 = 2$$

$$1.0101 \times 2^2 \xrightarrow{\text{com sinal}(-)} -5,25$$

115. \$f2 = 0x3A600000 \$f4 = 0x8A600000

sub. s \$f0, \$f2, \$f4

$$\$f2 = \underbrace{0}_{S} \underbrace{0111010101100000000000000000}_{E} \underbrace{0000000000000000}_{f}$$

$$E = 01110100_2 = 116_{10} \Rightarrow Ex_4 = -11$$

$$0 < 1.11 \times 2^{-11}$$

$$\$f4 = \underbrace{1}_{S} \underbrace{0111010101100000000000000000}_{E} \underbrace{0000000000000000}_{f}$$

$$E = 01110100_2 = 116_{10} \Rightarrow Ex_4 = -11$$

$$0 > 1.11 \times 2^{-11}$$

$$\$f2 - \$f4 = \underbrace{1.11 \times 2^{-11}}_{>0} - \underbrace{1.11 \times 2^{-11}}_{<0} =$$

$$= (1.11 + 1.11) \times 2^{-11}$$

$$= 11.10 \times 2^{-11} = 1.110 \times 2^{-10}$$

$$Ex_4 = -10 \Rightarrow E = 117_{10} = 01110101_2$$

$$\begin{array}{r} C.A \\ 1.1 \\ +1.1 \\ \hline 11.10 \end{array}$$

o resultado tem sinal positivo

$$\underbrace{0}_{S} \underbrace{01110101110000000000000000000000}_{E}$$

$$\$f0 = 0x3AE00000$$

116. \$f4 = 0x3F10 0000

$$= 0011\ 1111\ 0001\ 0000\ 0000\ 0000\ 0000\ 0000$$

S E f

$$E = 0111110_2 \Rightarrow E_4 = -1$$

$$0 < 1.001 \times 2^{-1}$$

\$f6 = 0x408C 0000

$$= 0100\ 0000\ 1000\ 1100\ 0000\ 0000\ 0000\ 0000$$

S E f

$$E = 10000001_2 \Rightarrow E_4 = 2$$

$$0 < 1.00011 \times 2^2$$

add. s \$f8, \$f4, \$f6

$$\underset{>0}{1.001 \times 2^{-1}} + \underset{>0}{1.00011 \times 2^2} =$$

$$= (0.001001 + 1.00011) \times 2^2$$

$$= 1.001111 \times 2^2$$

$$E_4 = 2 \Rightarrow E = 10000001$$

C.A

$$\begin{array}{r} 0.001001 \\ + 1.000110 \\ \hline 1.001111 \end{array}$$

o resultado tem sinal positivo

$$\$f8 = 0\ 10000001\ 0011110000\ 0000000000000000$$

S E f

$$\$f8 = 0x409E 0000$$

// _____ //

\$f2 = 0x3F90 0000

$$= 0011\ 1111\ 1001\ 0000\ 0000\ 0000\ 0000\ 0000$$

S E f

$$E = 01111110_2 \Rightarrow E_4 = 0$$

$$0 < 1.001 \times 2^0$$

\$f4 = 0xBEA0 0000

$$= 1011\ 1110\ 1010\ 0000\ 0000\ 0000\ 0000\ 0000$$

S E f

$$E = 0111101_2 \Rightarrow E_4 = -2$$

$$0 > 1.01 \times 2^{-2}$$

mul. s \$f0, \$f2, \$f4

C.A

$$\begin{array}{r} 1.001 \\ \times 1.010 \\ \hline 0.000 \\ 10.010 \\ 0000 \\ \hline + 1.001 \\ \hline 1.011010 \end{array}$$

$$1.001 \times 2^0 \times 1.01 \times 2^{-2} = (1.001 \times 1.01) \times 2^{-2} =$$

$$= 1.01101 \times 2^{-2}$$

$$E_4 = -2 \Rightarrow E = 0111101_2$$

o resultado tem sinal negativo

$$1\ 0111101\ 011010000000000000000000$$

S E f

$$\$f0 = 0xBEB4 0000$$

(continuação)

116. \$f2 = 0x258C0000

= 0101 0101 1000 1100 0000 0000 0000 0000
S E F

E = 0101011₂ ⇒ E₄ = -52

0 < 1.00011 × 2⁻⁵²

div. s \$f0, \$f2, \$f4

$$(1.00011 \times 2^{-52}) / (1.11 \times 2^3) =$$

$$= 0.101 \times 2^{-55}$$

$$= 1.01 \times 2^{-56}$$

$$E_{4k} = -56 \Rightarrow E = 01000111_2$$

C.A

1.00011 1.11
- 0.111 0.101

0.0011
- 0.0

1.11
- 1.11

0.00

o resultado tem sinal positivo

0 01000111 010 0000 0000 0000 0000 0000
S E F

\$f0 = 0x23A00000

117.

1.1111 1111 1111 1110 | 1000 0000

1110 | 1000
→ 1111

16 bits para f

impar + próximo

1.1111 1111 1111 1111

118. .data

ko: .float 1,5

.text

.glob main

main: l.s \$f6, ko # \$f6 = 1,5

c.le.s \$f8, \$f6

bne endwhile

while: (...)

endwhile: (...)

119. Formato IEEE754 prec simples

Par + próximo

$$100,110110000000000000010110_2$$

$$\hookrightarrow 1,00110110000000000000010110_2 \times 2^2$$

Parte fracionária: 00110110000000000000010110

$$\hookrightarrow 001101100000000000000110$$

$$G=1$$

$$R=0$$

$$S=0$$

Em precisão simples: 1,001101100000000000000110

120. Single cycle $f_{\max} = 2 \text{ GHz}$

\hookrightarrow DLW dita a f_{\max} $t_{\text{LW}} = \frac{1}{2 \times 10^9} =$

$$t_{\text{LW}} = t_{\text{RH}} + \max(t_{\text{RRF}}, t_{\text{CNTL}}, t_{\text{SE}}) + t_{\text{ALU}} + t_{\text{RH}} + t_{\text{WRF}}$$

$$\Leftrightarrow 500 \text{ ps} = 175 \text{ ps} + \max(45 \text{ ps}, 10 \text{ ps}, 0 \text{ ns}) + t_{\text{ALU}} + 175 \text{ ps} + 15 \text{ ps}$$

$$\Leftrightarrow t_{\text{ALU}} = 90 \text{ ps}$$

121. Single cycle SW

$$t_{\text{SW}} = t_{\text{RH}} + \max(t_{\text{RRF}}, t_{\text{CNTL}}, t_{\text{SE}}) + t_{\text{ALU}} + t_{\text{WM}}$$

$$\Leftrightarrow \frac{1}{f_{\text{SW}}} = 12 \text{ ns} + \max(4 \text{ ns}, 1 \text{ ns}, 0 \text{ ns}) + 5 \text{ ns} + 4 \text{ ns}$$

$$\Leftrightarrow f_{\text{SW}} = 40 \text{ MHz}$$

122. Single cycle BEQ

$$t_{\text{BEQ}} = t_{\text{RH}} + \max(\max(t_{\text{RRF}}, t_{\text{CNTL}}) + t_{\text{ALU}}, t_{\text{SE}} + t_{\text{SL}} + t_{\text{ADD}}) + t_{\text{OCP}}$$

$$\Leftrightarrow \frac{1}{f_{\text{BEQ}}} = 11 \text{ ns} + \max(8 \text{ ns}, 2 \text{ ns}) + 1 \text{ ns}$$

$$\Leftrightarrow f_{\text{BEQ}} = 50 \text{ MHz}$$

123. Single cycle tipo R

$$\begin{aligned}
 t_{R-type} &= t_{RM} + \max(t_{RPF}, t_{CNTL}) + t_{ALU} + t_{WRF} \\
 &= 12ns + 3ns + 6ns + 1ns \\
 &= 22ns
 \end{aligned}$$

124. A arquitetura single cycle é caracterizada pelo modelo Harvard:

existem 2 memórias distintas (1 para dados e outra para instruções);
 é possível obter a dados e a instruções no mesmo ciclo de relógio;
 memória de dados e instruções podem ter comprimentos de palavra diferentes.

Uma instrução é executada num único ciclo de relógio

A unidade de controle é combinatória, gera os sinais que controlam
 mux's, controlam a leitura e escrita das memórias e banco de registros.

125. Endereço de memória que contém a próxima instrução a ser executada.

126.

Instrução	OpCode	ALUOp	Branch	RegDst	ALUSrc	Mem to Reg	Reg write	Mem Read	Mem write
LW	100011	00	0	0	1	1	1	1	0
SW	101011	00	0	X	1	X	0	0	1
ADDI	001000	00	0	0	1	0	1	0	0
SLTI	001010	11	0	0	1	0	0	0	0
BEQ	000100	01	1	X	0	X	0	0	0
R-format	000000	10	0	1	0	0	1	0	0

127. Jal → será necessário ligar a saída do somador ($PC+4$) à entrada
 W. Data do Register File (esta entrada precisa de 1 mux); a entrada R. Reg #1
 precisa de mux para encaminhar uma entrada adicional, uma constante (11111,
 i.e., endereço o registro \$31).

$JT \rightarrow$ A saída R.Data #1 do Register File terá que ser ligada a uma entrada adicional do mux que encaminha o PC/JTA, assim o conteúdo do registro modificado em JT poderá ser o PC. Este mux terá 3 entradas: quando o sinal "Jump" = "00", encaminha PC+4 ou BTA; quando "Jump" = "01", encaminha JTA; quando "Jump" = "10", encaminha JTA do JT.

("Jump" = "11" \rightarrow não usado)

128. $set \$3, \$5, \$9$

Esta instrução é realizada na ALU através da subtração entre o 1º operando e o 2º ($[\$5] - [\$9]$); o resultado desta diferença será analisado para determinar a condição set if less than: esta condição é verdadeira se o MSB da diferença for '1'; i.e., resultado um n.º negativo.

$\$5: 0xFF120008$

$\$9: 0x00C00FFF$

$$\underbrace{0xFF120008}_{<0} < \underbrace{0x00C00FFF}_{>0}$$

$\$3: 1$

129. Single cycle

$$\begin{aligned} a) \quad t_{JAL-R} &= t_{RM} + \max(t_{RRF}, t_{CNTL}) + t_{ALU} + t_{WRF} \\ &= 12 + 5 + 7 + 2 = 26 \text{ ns} \end{aligned}$$

$$\begin{aligned} t_{LW} &= t_{RM} + \max(t_{RRF}, t_{CNTL}, t_{SE}) + t_{ALU} + t_{RM} + t_{WRF} \\ &= 12 + 5 + 7 + 12 + 2 = 38 \text{ ns} \end{aligned}$$

$$\begin{aligned} t_{SW} &= t_{RM} + \max(t_{RRF}, t_{CNTL}, t_{SE}) + t_{ALU} + t_{WM} \\ &= 12 + 5 + 7 + 4 = 28 \text{ ns} \end{aligned}$$

$$\begin{aligned} t_{BEQ} &= t_{RM} + \max(\max(t_{RRF}, t_{CNTL}) + t_{ALU}, t_{SE} + t_{SL2} + t_{ADD}) + t_{JtPC} \\ &= 12 + 12 + 1 = 25 \text{ ns} \end{aligned}$$

$$t_J = t_{RM} + \max(t_{CNTL}, t_{SL2}) + t_{JtPC} = 12 + 2 + 1 = 15 \text{ ns}$$

129. b) Instrução + longo 3 LW ($t_{LW} = 38 \text{ ns}$)

$$f_{\max} = \frac{1}{38 \times 10^{-9}} \approx 26 \text{ MHz}$$

130. clock variável

15% LW

15% SW

40% tipo R

20% Branch

$t_{\text{clock-fixo}} = 38 \text{ ns}$

10% J

$$T_{\text{MED}} = 0,15 \times 38 + 0,15 \times 28 + 0,4 \times 26 + 0,2 \times 25 + 0,1 \times 15 = 26,8 \text{ ns}$$

$$\frac{\text{Des. clock-variável}}{\text{Des. clock-fixo}} = \frac{n^{\circ} \text{ inst.} \times t_{\text{clock-fixo}}}{n^{\circ} \text{ inst.} \times t_{\text{clock-variável}}} = \frac{38}{26,8} = 1,42$$

131. f_{\max} em multi-cycle

Valor dos tempos de atraso dos elementos operativos: $t_{RH} = 12 \text{ ns}$

$$f_{\max} = \frac{1}{12 \times 10^{-9}} = 83 \text{ MHz}$$

132. a. odd \$t0, \$t2, \$t1

new \$t0, 0 (\$t3)

Seq \$t0, \$t1, next

SW

BEQ

clock

PC Write Cond	0	X	0	0	0	X	0	1
PC Write	0	1	0	0	0	1	0	0
Mem Write	0	0	0	0	1	0	0	0
Mem Read	0	1	0	0	0	1	0	0
Mem to Reg	0	X	X	X	X	X	X	X
IR Write	0	1	0	0	0	1	0	0
ALU Sel A	X	0	0	1	X	0	0	1
ALU Sel B	XX	01	11	10	XX	01	11	00
ALU Op	XX	00	00	00	XX	00	00	01
ImmD	X	0	X	X	1	0	X	X
PC Source	XX	00	XX	XX	XX	00	XX	01
Reg Write	1	0	0	0	0	0	0	0
Reg Dst	1	X	X	X	X	X	X	X

133. (Igual ao anterior)

134. (Baseado ao 132.)

135 a) add \$t0, \$t2, \$t1 #0x00400000

sub \$t0, 0(\$t3)

seq \$t0, \$t1, next

SW

BEQ

clock	1	2	3	4	5	6	7	8
PC	0x00400004	0x00400004	0x00400008	0x00400008	0x0040000B	0x00400008	0x0040000C	0x0040000C
IR	0x00404020	0x00404020	0xAD680000	0xAD680008	0xAD680000	0xAD680000	*	*
MDR	?	?	0xAD680000	?	?	?	*	?
ALUR ₀₁	?	0x00400008	BTA: 0x00400008	0x00000028	?	0x0040000C	*	0x90FF FF64
ALUR ₀₁	0x90FF FF84	?	0x00400008	0x00400008	0x00000028	?	0x0040000C	*
ALUR ₀₂	?	0	0	0	?	0	*	0
A	0x90FF FF64	0x90FF FF64	0x90FF FF64	0x00000028	0x00000028	0x00000028	0x00000028	0x90FF FF84
B	0x00000020	0x00000020	0x00000020	0x90FF FF84	0x90FF FF84	0x90FF FF84	0x90FF FF84	0x00000020
ALUR ₀₄	XX	00	00	00	XX	00	00	01

* → é preciso o código máquina da instrução.

136. a) Data path single-cycle

$$3 + 5 \times (6) + 2 = 35 \text{ ciclos}$$

\$1: r1 = 0x10 1º ite

\$2: r2 = 0x20

alu \$5, \$1, \$2

ciclo feito 5 vezes

r1++	
0x14	2º ite
0x18	3º ite
0x1C	4º ite
0x20	5º ite

b) Data path multi-cycle

$$5 \times (3 + 1 \times 5) + 4 \times (2 + 4 \times 5) + 3 \times (1 \times 5) = 143 \text{ ciclos}$$

137. $0x00000004 : 0x2C$

$\uparrow 1 = 0x10$ 1ª interação

Single-cycle

$$3 + 8 \times 6 + 2 = 53 \text{ ciclos}$$

$\uparrow 1 + +$	Nº interação
$0x14$	2ª
$0x18$	3ª
$0x1C$	4ª
$0x20$	5ª
$0x24$	6ª
$0x28$	7ª
$0x2C$	8ª

ciclo feito 8 vezes

Multi-cycle

$$5 \times (3 + 1 \times 8) + 4 \times (2 + 4 \times 8) + 3 \times (1 \times 8) = 215 \text{ ciclos}$$

138. O datapath single-cycle tem 2 memórias enquanto o multi-cycle tem apenas uma; o single-cycle tem uma ALU e 2 somadores separados para calcular

$PC+4$ e o BTA enquanto o multi-cycle tem apenas uma ALU

139. Multi-cycle 2º primeiros ciclos

operações realizadas na ALU: 1º ciclo: cálculo de $PC+4$
2º ciclo: cálculo de BTA

145. A unidade de controle da implementação pipeline é uma unidade combinatória que gera os sinais de controle em função do código da instrução.

Os sinais MemRead e MemWrite são propagados até a fase MEM, o sinal RegWrite é propagado até WB, o sinal Branch é propagado até ID (sendo resolvida)

146. O período do relógio tem que ser ajustado ao maior dos tempos de atraso provocado pelos elementos operativos

147. a) $\text{mant}(\text{tempo de atraso}) = 10\text{ns}$

$$f_{\text{máx}} = \frac{1}{10 \times 10^{-9}} = 100\text{MHz}$$

$$b) f_{\text{máx}} = \frac{1}{8 \times 10^{-9}} = 125\text{MHz}$$

$$c) f_{\text{máx}} = 100\text{MHz}$$

148. Hazard estrutural, de dados e controle

149. Forwarding (hazard de dados)

150. O hazard de dados ocorre quando um registro operando de uma instrução necessita do valor do operando mas este registro é "destino" de outra instrução que regue em estágio(1) mais avançado(1) no pipeline

151. Delayed Branch. Esta técnica em trocar a instrução imediatamente a seguir ao branch pelo branch, i.e, esta próxima instrução entra 1º no pipeline e só depois é que entra o branch. Este espaço onde fica a próxima instrução é designado por delayed branch slot.