**1.** Faça um estudo do MIPS destacando os seguintes aspectos:

**a. tamanho do dado a ser processado**

*Inteiros: 32 bits;*

*Ponto flutuante: 32 bits (precisão simples), 64 bits (precisão dupla);*

*Caracteres: 8 bits;*

*Lógico: 8 bits.*

**b. espaço de endereçamento de memória**

*2³² bytes (Esse valor vem do fato que o espaço de endereçamento de memória depende do tamanho das instruções no processador, que no caso do MIPS é 32 bits. Se um endereço tem x bits o espaço de endereçamento é 2x bytes). Lembrar que a memória é dividida em BYTES (8 bits) para não haver desperdício quando se guardar Caracteres ou valores lógico, isso significa que Inteiros ocupam 4 espaços na memória.*

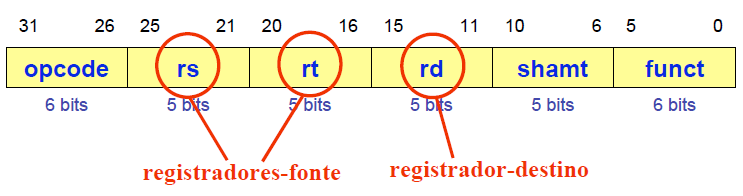
**c. número de registradores**

*No MIPS existem 32 registradores ($0..$31), que podem guardar tanto dados como endereço (chamados de registradores de propósito geral), com 32 bits cada um. (Esse valor tem relação com o tamanho dos dados do tipo inteiro).*

**d. formato e tamanho das instruções**

*O tamanho das instruções do MIPS é fixo e tem 32 bits (o que facilita a implementação). Existem 3 tipos de instruções:*

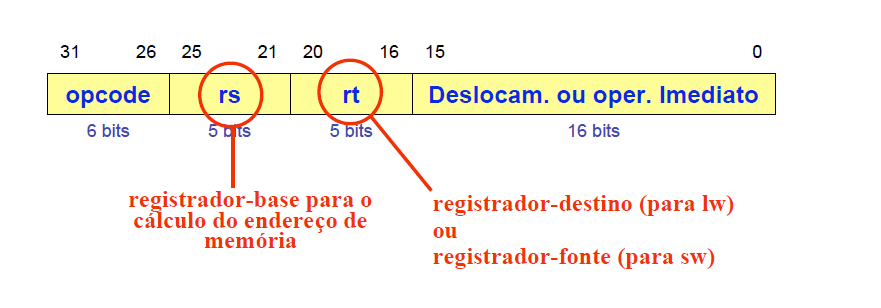
*-****Tipo R****: Instruções que envolvem registradores (operações lógicas, aritméticas e de deslocamento)*

**

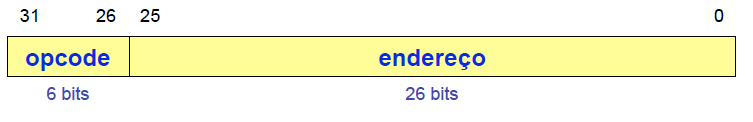
*“funct” define a operação a ser feita pela ALU*

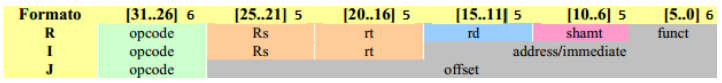
*“shamt” (shift amount) é usado em instruções de deslocamento*

*-****Tipo I****: Instruções que fazem transferência de informações e desvios (lw, sw, beq e imediatos)*

**

*-****Tipo J****: Desvio incondicional (j, jal)*

**

**

**2.** Faça um estudo comparativo entre uma implementação monociclo e uma implementação multiciclo, considerando desempenho e custo de implementação em função da área de silício.

*Em uma implementação monociclo todas as instruções são realizadas em apenas um ciclo de clock, dessa maneira a implementação fica mais simples. Porém, o clock deve ser definido pela instrução mais lenta, assim uma instrução mais rápida não manda a saída no momento em que termina pois tem que esperar o próximo pulso. Além disso mais componentes são necessários para o armazenamento de dados durante a instrução e a realização de operações aritméticas.*

*Já em uma implementação multiciclo, cada etapa é realizada em um pulso de clock, dessa maneira haverá um número variável de pulsos para cada instrução e o clock é definido pela etapa mais longa. Assim o processador é usado por completo de uma maneira mais eficiente sem a necessidade de esperas sem tarefas sendo realizadas. Além disso, o número de componentes caros (área de silício) é menor. Um multiciclo é mais complicado de ser implementado pela necessidade de vários estados e de um controlador mais rebuscado, além da necessidade do uso de registradores.*

**3.** Tanto na arquitetura projetada em sala de aula como na descrita no livro todas as instruções aritméticas envolvem três registradores e possuem o mesmo formato. Quais as vantagens e desvantagens desta decisão de projeto?

***Vantagens:***

*-Registradores são mais rápidos do que ler da memória.  
-Quando 3 registradores podemos manter os operandos além do resultado.utilizamos*

*-Usando sempre 3 registradores para as instruções faz com que todas sigam um padrão tornando mais fácil de implementar.*

***Desvantagens:***

*-É necessário carregar todas as variáveis da memória para os registradores, tornando o código mais extenso.*

**4.** Porque o processador MIPS se caracteriza como uma arquitetura LOAD/STORE? Quais as vantagens e desvantagens desta decisão no projeto da arquitetura do processador?

*O processador MIPS é caracterizado como uma arquitetura Load/Store porque a única maneira de acessar a memória é através dessas instruções (lw e sw), o resto é tudo feito via registradores,pois tem sua arquitetura base do tipo register-register.*

***Vantagens:***

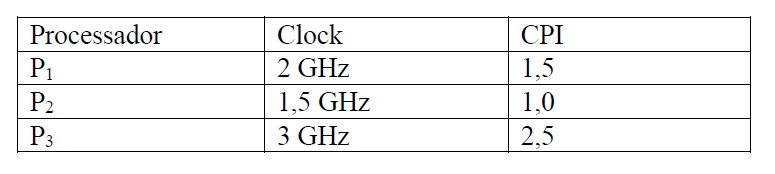
*- A leitura e escrita em registradores são mais rápidas do que na memória, o que faz com que as instruções sejam realizadas em menos tempo.*

*- O formato das instruções lw/sw R, desloc(Rbase) facilita o acesso a arrays e pilha, pois o deslocamento pode ser variável em tempo de execução.*

***Desvantagens:***

*-É necessário carregar todas as variáveis da memória para os registradores, tornando o código mais extenso.*

**5.** Considere três processadores P1, P2, e P3 executando o mesmo conjunto de instruções com as frequências de clocks descritas na tabela abaixo.



**a. Qual processador tem o melhor desempenho?**

Tempo = n° de instruções x CPI x Período obs: Período = 1/Frequencia

Desempenho = 1 / Tempo

(Os 3 processadores realizam uma quantidade I de instruções)

Tempo P1 = I x 1,5 x (1/(2x10**9**)) = I x 3/2 x (1/(2x10**9**)) = I x 3/(4x10**9**)

Tempo P2 = I x 1,0 x (1/(1,5x10**9**)) = I x 1 x (2/(3x10**9**)) = I x 2/(3x10**9**)

Tempo P3 = I x 2,5 x (1/(3x10**9**)) = I x 5/2 x (1/(3x10**9**)) = I x 5/(6x10**9**)

Tempo P1/Tempo P2 = (I x 3/(4x10**9**)) / (I x 2/(3x10**9**)) = 1,25 (P2 é menor)

Tempo P2/Tempo P3 = (I x 2/(3x10**9**)) / (I x 5/(6x10**9**)) = 0.8 (P2 é menor)

Ou seja, o processador P2 é o que tem melhor desempenho.

**b. Se os processadores executam um programa em 10 segundos, encontre o número de ciclos e de instruções para cada um.**

**Tempo de Execução = CPU Ciclos/Frequência do Clock**

**P1: 10=CPU Ciclos/2x10^9 CPU Ciclos = 20x10^9**

**P2: 10=CPU Ciclos/1,5x10^9 CPU Ciclos = 15x10^9**

**P3: 10=CPU Ciclos/3x10^9 CPU Ciclos = 30x10^9**

**Número de instruções = CPU Ciclos/CPI**

**P1: Número de instruções=20x10^9/1,5 Número de instruções= (40/3)x10^9**

**P2: Número de instruções=15x10^9/1 Número de instruções= (15)x10^9**

**P3: Número de instruções=30x10^9/2,5 Número de instruções= (12)x10^9**

**c. Estamos tentando reduzir o tempo em 30%, mas isto levaria a um aumento de 20% no CPI. Qual a taxa de clock para conseguir esta redução?**

Novos CPIs:

**CPI P1: 1,8**

**CPI P2: 1,2**

**CPI P3: 3**

**Tempo de execução agora será de 7 segundos**

**Tempo de execução = (Número de instruções x CPI)/Frequência do Clock**

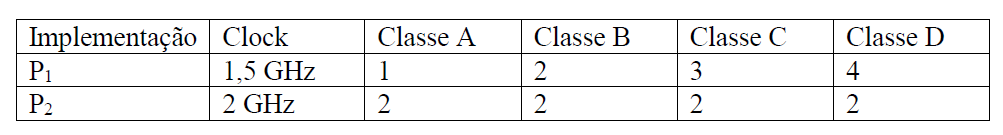
**P1: Frequência do Clock = (((40/3)x1,8)/7) GHz = (24/7) GHz**

**P2: Frequência do Clock = ((15x1,2)/7) GHz = (18/7) GHz**

**P3: Frequência do Clock = ((12x3)/7) GHz = (36/7) GHz**

**By: Ricardo**

**6.** Considere duas implementações diferentes do mesmo conjunto de instruções, que pode ser classificado em 4 classes de instruções: A, B, C, e D. A frequência de clock e CPI para cada uma das implementações é dada na tabela abaixo.



Dada uma aplicação com 10^6 instruções sendo 10% de instruções da classe A, 20% de instruções da classe B, 50% de instruções da classe C e 20% de instruções da classe D.

a. Qual implementação tem melhor desempenho?

b. Qual o CPI global para cada implementação?

c. Calcule o número de ciclos de clock para cada implementação.

a)

**P1: Ciclos CPU = (10^5 x 1) + (10^5 x 2 x 2) + (10^5 x 5 x 3) + (10^5 x 2 x 4) = 28 x 10^5**

**Desempenho = freq/ciclos = 1,5x10^9/28x10^5 = 1,5x10^4 / 28**

**P2: Ciclos CPU = (10^5 x 2) + (10^5 x 2 x 2) + (10^5 x 5 x 2) + (10^5 x 2 x 2) = 20 x 10^5**

**Desempenho = freq/ciclos = 2x10^9/20x10^5 = 2x10^9/2x10^6 = 10^3**

**P1/P2 = (1,5x10^4/28)/10^3 = 0,5...**

**P1 é 0,5 vezes a velocidade de P2**

**Logo, P2 apresenta melhor desempenho.**

b)

**P1: CPI = 28x10^5/10^6 CPI = 2,8**

**P2: CPI = 20x10^5/10^6**

**CPI = 2**

c)

**Não entendi bem essa, porque a resposta parece já ter sido obtida acima, mas se fosse na hora da prova faria isso…**

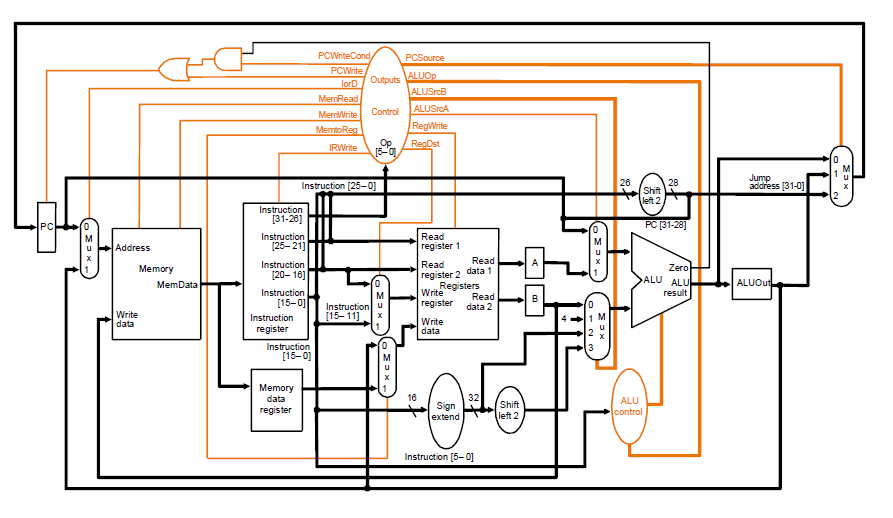
**Frequência é o número de ciclos de clock em 1 segundo, logo:**

**P1: 28x10^5**

**P2:20x10^5**

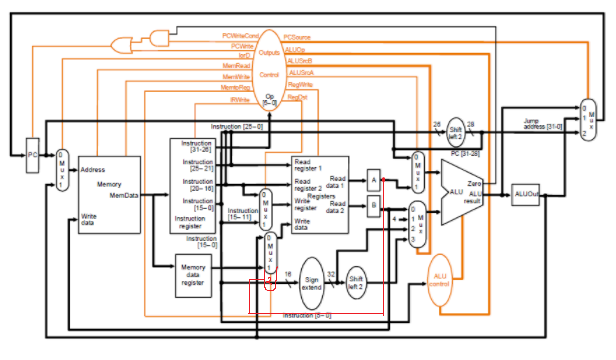
**(Usando a fórmula Ciclos CPU = Número de Instruções x CPI (encontrado na letra b) obtém-se a mesma resposta.)**

**7.** Suponha que foi inserida a instrução move rt, rs, que copia o conteúdo do registrador rs para o registrador rt. Considerando a implementação apresentada no livro texto, como se poderia implementar esta instrução preservando todos os outros registradores de forma a ser ter o menor CPI. Quais as modificações na unidade de processamento e na unidade de controle e qual seria o CPI desta instrução?



**Figura 1 - Unidade de processamento do MIPS (CPU do Livro)**

*Resposta:*

****

Seria necessário a mudança do fio MemToReg para 2 bits e colocar um novo estado que envia o valor ‘2’ para o mux por MemToReg para se ter o valor de rs em WriteData

Busca -> Decod -> Move (3 CPI) -> Busca

**8.** Considere a instrução move rt, rs do MIPS, que copia o conteúdo do registrador rs no registrador rt. Esta instrução é frequentemente utilizada junto com a instrução de desvio condicional conforme a sequência mostrada abaixo

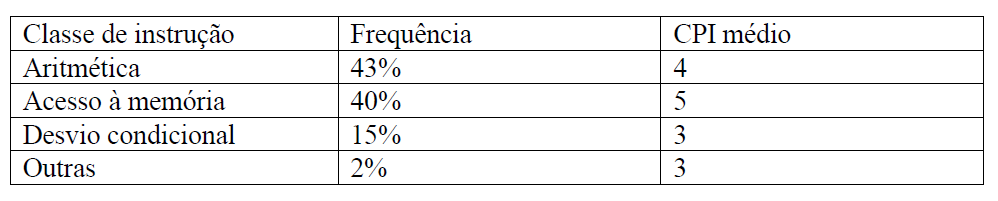
bne ri, rzero, X

move rt,rs

X: .....

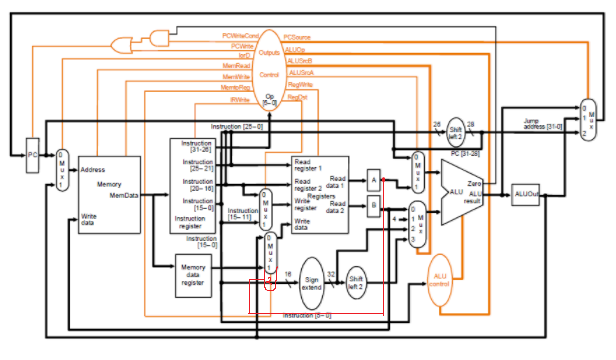
Face à crescente utilização da sequência de instruções acima, foi incluída uma nova instrução cmove rd, rs, rt que copia o conteúdo do registrador rs no registrador rd se o conteúdo do registrador rt for igual a zero. Esta instrução substitui a sequência de instruções mostrada acima.

Para uma determinada aplicação a frequência de execução das instruções é dada pela tabela abaixo:



a. Implemente a instrução cmove de forma que ela possua o mesmo CPI da instrução move. Quais as modificações na unidade de processamento e na unidade de controle?

b. Considerando que a instrução cmove possui o mesmo CPI da instrução move, implementada na questão anterior e se esta instrução substitui 20% das instruções de desvio condicional, a inclusão desta instrução melhoraria o desempenho da máquina?? Justifique. Se sim, qual o speedup?

**a)** ****

Mesma coisa do move em relação a MemToReg e o mux controlado por essa flag. Adicional constante 0 entrando no mux ALUScrA e o aumento desse fio para poder controlar o mux (2bit) e o controle precisa receber a flag da ALU (equal than) para a comparação com 0

O estado cmove manda ALUScrA = 2; ALUscrB = 0; ALUOp = 111; MemToReg = 2; RegDst = 1;

if(ET == 1) {

RegWrite = 1

}

Busca -> Decod -> cmove (3 CPI)

**b)** Sim, pois bne + move seriam 6 ciclos de clock, enquanto cmove seria somente 3.

20% de 15% = 3%

20% de 2% = 0.4%

Before

Ari 43% CPI 4

Acesso Mem 40% CPI 5

Desvio Cond 15% CPI 3

Outros (MOVE [Teoria]) 2% CPI 3

4\*0,43 + 5\*0,4 + 3\*0,15 + 3\*0,02 = **4,23**

After

Ari 43% CPI 4

Acesso Mem 40% CPI 5

Desvio Cond 12% CPI 3

Cmove 3,4% CPI 3

Outros (MOVE [Teoria]) 1,6% CPI 3

4\*0,43 + 5\*0,4 + 3\*0,12 + 3\*0,034 + 3\*0,016 = **4,23**