

### Primeira Prova

1) Projete um circuito combinacional que computa a paridade par de uma palavra de quatro bits. O bit de paridade deve ser gerado de tal forma que o número de bits em **1** no quinteto seja par. Use um decodificador ou um seletor, além de outros componentes que julgar necessário. [6 pontos]

2) Projete uma máquina de estados que recebe uma seqüência de quatro bits em sua entrada ent, os copia na saída sai e insere um bit de paridade par na seqüência de saída, a cada quatro bits recebidos. Seu projeto deve conter um diagrama de estados, as funções de próximo estado e de saída, bem como a implementação destas funções com flip-flops tipo D. [12 pontos]

3) Traduza para assembly do MIPS a função abaixo: [12 pontos]

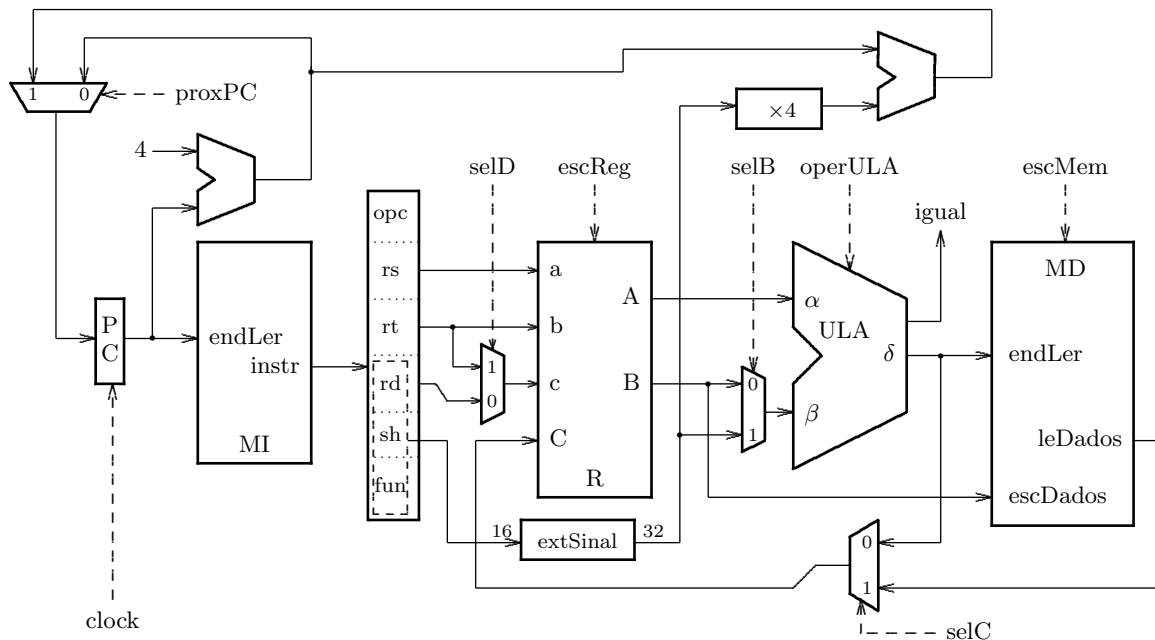
```
int fat(int n) {
    int i,j;
    j=1;
    if (n > 1)
        for (i=1; i <= n; i++)
            j = j*i;
    return(j);
}
```

## Segunda Prova

1) Mostre como implementar a instrução **BRANCH-AND-LINK** definida abaixo. Sua resposta deve conter: [20 pontos]

- (i) um diagrama claro e limpo com o circuito de dados do processador;
- (ii) um diagrama de tempos completo da execução desta instrução;
- (iii) explique por que o endereço de retorno é em  $PC+8$ ?

**bal desl** #  $\$31 \leftarrow PC+8$  ,  $PC \leftarrow (PC+4) + (\text{ext}(\text{desl}) \ll 2)$  (formato I)



2) Traduza o trecho de programa abaixo assembly do MIPS. [10 pontos]  
 Para facilitar a correção indique os registradores como *ri*, *rv*, etc.

```
typedef struct A {
    int a;
    int b;
    int d;
    int m;
} Atype;

Atype v[1024];
int i;
for (i=0; i < 1024; i+=1) {
    v[i].d = v[i].a / v[i].b;
    v[i].m = v[i].a % v[i].b;
}
```



**Exame Final**

1) Suponha que uma biblioteca VHDL contenha a seguinte primitiva, que é um *circuito combinacional*: `mult-por-1(A,B,S,R,N) ≡ R <= ( S ? A+B : 0&A )`

sendo  $A$  e  $B$  inteiros representados em  $N$  bits,  $R$  é um inteiro representado em  $N + 1$  bits,  $S$  é um bit, e  $X\&Y$  representa a concatenação de  $X$  com  $Y$ ,  $N$  é o parâmetro que indica a largura do componente `mult-por-1`, e  $( x ? y : z )$  é a expressão de seleção da linguagem C. [40 pontos]

(a) Usando várias instâncias de `mult-por-1`, mostre como implementar um multiplicador combinacional de  $5 \times 5$  bits (cinco×cinco);

(b) Supondo que o pior caso do tempo de propagação do componente `mult-por-1` seja proporcional a  $N$ , qual o pior caso do tempo de propagação do multiplicador? Justifique;

(c) Usando registradores de largura apropriada, mostre como transformar o circuito do item (a) para permitir a multiplicação de mais de um número simultaneamente, i.e. mostre como segmentar o multiplicador;

(d) Dê uma especificação para a temporização do seu projeto no item (c) e a justifique.

2) Considere um novo modo de endereçamento a ser adicionado ao conjunto de instruções do MIPS. Neste novo modo, o primeiro operando e o resultado são o mesmo registrador e o segundo operando é buscado da memória. Por exemplo, a instrução `addm` é definida como

```
addm r1, desloc(r2) # r1 := r1 + Mem[ desloc + r2 ]
```

(a) Mostre como implementar a instrução `addm` no processador visto em aula. Adicione quaisquer circuitos necessários e indique os sinais que controlam a execução desta instrução;

(b) desenhe um diagrama de tempo para a execução desta instrução no processador com suas adições. [40 pontos]

3) Traduza o trecho de programa abaixo para assembly do MIPS. [30 pontos]

*Para facilitar a correção indique os registradores como ri, rx, etc.*

```
int fun(int,int);
int a, b, i, j, x[NNN], y[MMM], z[PPP];
a = 16 * y[68000];
j = fun(a,13);
b = x[ y[ z[j] ] ];
```