

Instruções para entrega dos exercícios

- O trabalho pode ser feito em duplas. Não esqueça de identificar corretamente os membros da dupla.
- Plágio não será tolerado.
- Utilize o simulador de circuitos Digital para implementar as soluções.
- Envie um único arquivo `.dig` com o circuito que você tiver ao final da lista.
- Verifique a aula 06_aritmetica, a partir da seção “Circuitos de aritmética” para as informações vistas em sala.
- A cada exercício, modifique o circuito anterior para resolver o problema descrito.

1. Construa o circuito do somador completo, que faz a soma de dois números de um bit considerando o sinal *vem* e *vai*.

Entradas: bit *a*, bit *b*, bit *vem*.

Saídas: bit *s*, bit *vai*.

2. Faça o circuito do somador completo para números de 4 bits.

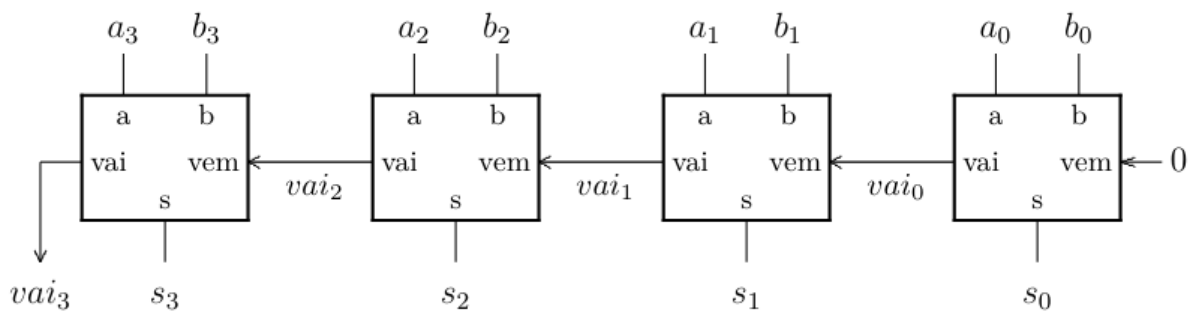


Figura 1: Circuito somador de 4 bits. Fonte: RH

Entradas: número A de 4 bits (a_3, a_2, a_1, a_0), número B de 4 bits (b_3, b_2, b_1, b_0), bit vem_0 .

Saídas: número S ($S = A + B$) de 4 bits (s_3, s_2, s_1, s_0), bit vai_3 .

3. Complemente o circuito para fazer o circuito que faz somas e subtrações.

Você deve acrescentar um sinal de entrada *inv* que indica que cada bit *b* seja invertido. Use um multiplexador para escolher o bit *b* direto da entrada ou o bit *b* invertido.

Você também deve modificar o sinal vem_0 para que, na subtração, o sinal seja ligado a 1. Dessa forma, os sinais *inv* e vem_0 executam a transformação do número B para o seu complemento de 2.

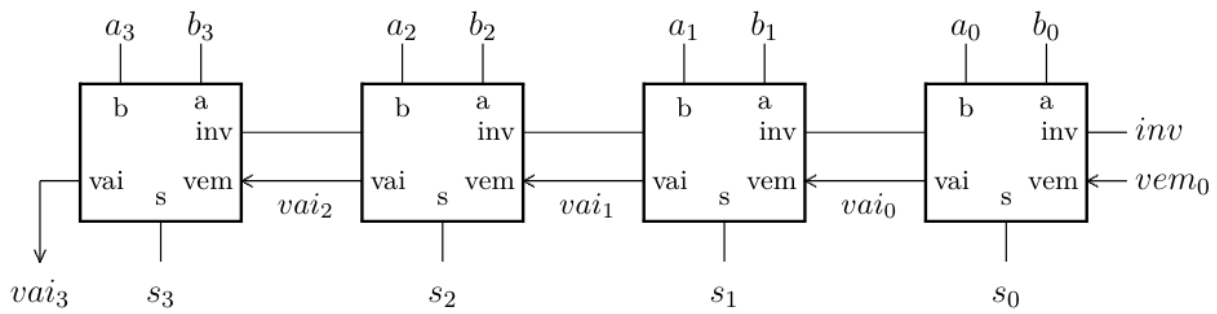


Figura 2: Circuito que efetua somas e subtrações. Fonte: RH

Entradas: número A de 4 bits (a_3, a_2, a_1, a_0), número B de 4 bits (b_3, b_2, b_1, b_0), bit inv .

Saídas: número S ($S = A + B$) de 4 bits (s_3, s_2, s_1, s_0), bit vai_3 .

4. Você deve implementar uma ULA que opera com 2 entradas de 4 bits. A ULA deve ter um sinal de entrada F que indica qual operação devemos mostrar na saída. Além disso, a ULA deve gerar um sinal de status T de 3 bits. Cada bit de status de T deve indicar se o resultado é zero, se é negativo e se ocorreu *overflow*.

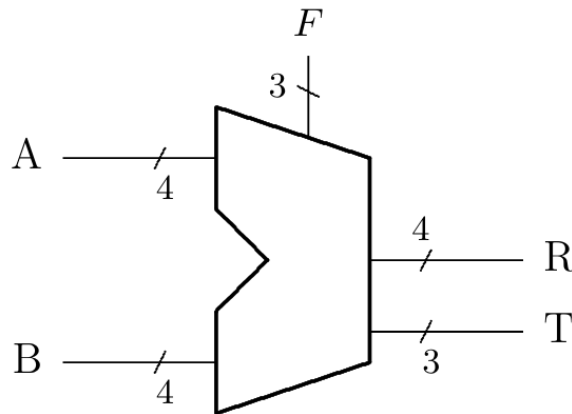


Figura 3: Unidade de lógica e aritmética. Fonte: RH

A ULA executa a operação de soma e subtração usando o circuito feito até agora. Além disso, a ULA deve ser capaz de fazer as operações de deslocamento para esquerda, deslocamento para direita, *not*, *and*, *xor*, *or*.

Cada bit da ULA irá ser parecido com o circuito da figura 4. Como você está fazendo uma ULA de 4 bits, terá 4 cópias do circuito da figura. Note que, pelo exercício anterior, você já tem as 4 cópias do circuito de soma e subtração.

Entradas: número A de 4 bits (a_3, a_2, a_1, a_0), número B de 4 bits (b_3, b_2, b_1, b_0), número F de 3 bits (você deve usar um *splitter* quando precisar acessar cada bit do número).

Saídas: número S ($S = AoperacaoB$) de 4 bits (s_3, s_2, s_1, s_0), número T de 3 bits contendo os status da operação.

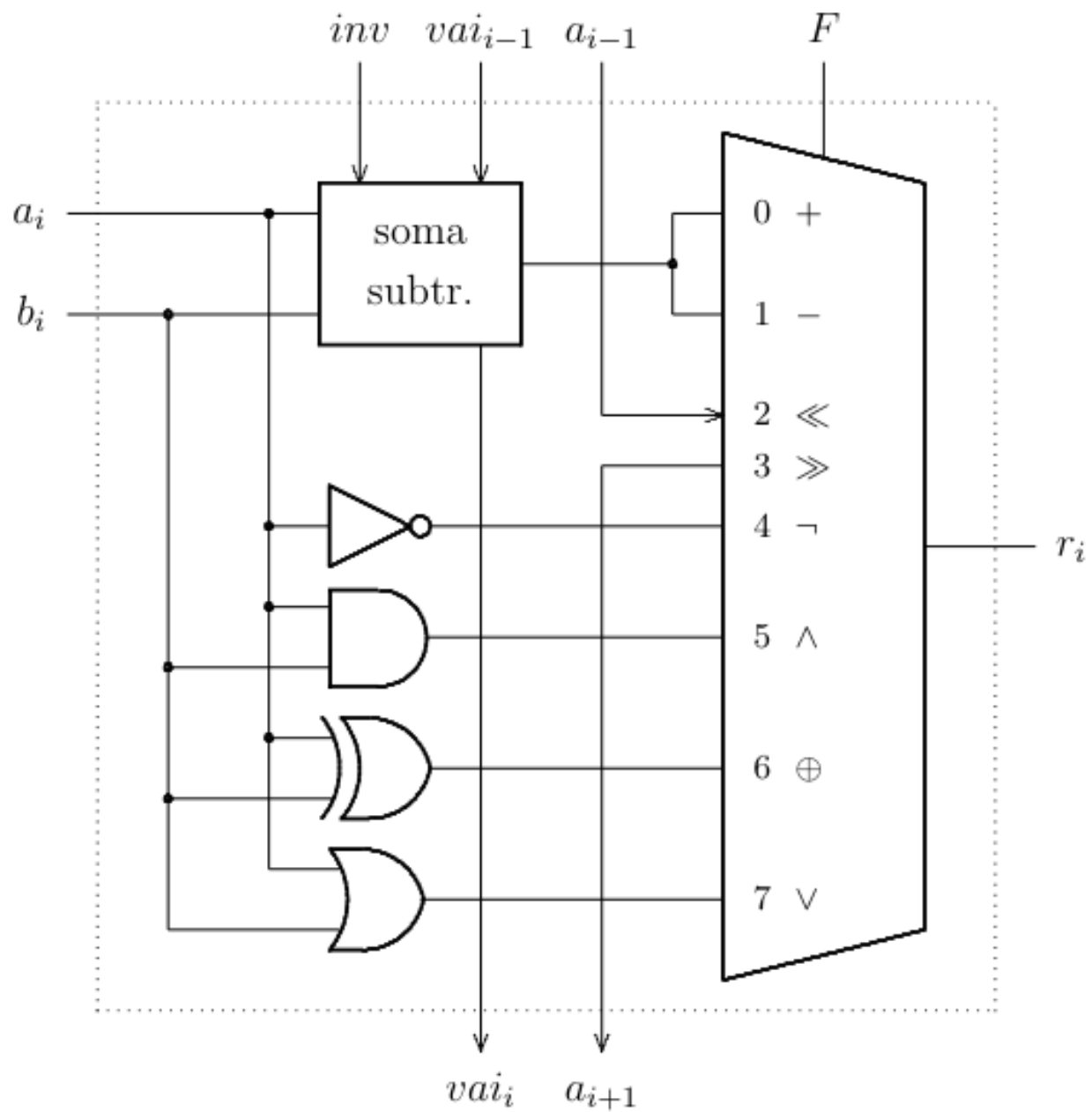


Figura 4: Fatia de um bit da ULA. Fonte: RH