

Projeto de circuito

MAC0329 – Álgebra booleana e aplicações (DCC / IME-USP — 2020)

– Todas as etapas do projeto deverão ser feitas na plataforma <https://circuitverse.org> –

Parte 2: ULA

Ao concluir o projeto no CircuitVerse, entregue um print do seu circuito no moodle até 19/06¹

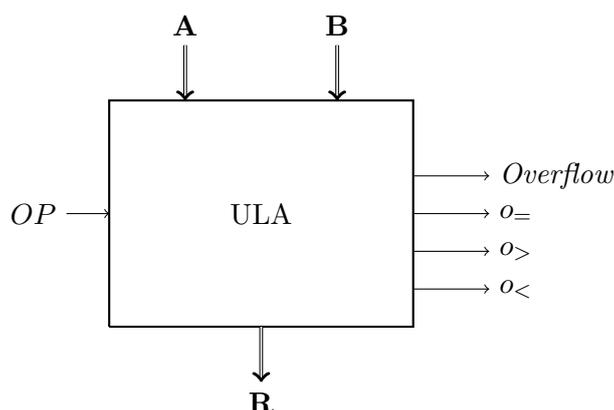
Nesta segunda etapa do projeto (Parte 2), o objetivo é a implementação de uma ULA simplificada, dando prosseguimento à etapa anterior. Na etapa anterior implementamos um circuito somador que realiza a adição de dois números de 8 *bits*. Além disso, o somador foi utilizado para realizar tanto a adição como a subtração, e os *carries* por ele devolvidos serviram para detectar *overflow* para as interpretações com sinal e sem sinal dos números operados.

Nesta etapa construiremos um comparador que, juntamente com o somador, formarão a nossa ULA. A nossa ULA é simplificada pois será capaz de realizar apenas duas operações aritméticas (adição e subtração) e comparações de dois números. **Iremos considerar apenas a interpretação sem sinal².**

1 Especificação da ULA a ser implementada

ULA é a contração do nome Unidade Lógico-aritmética. Ela é a parte da CPU que é responsável por executar as operações aritméticas e lógicas. Em nosso projeto, a ULA consistirá do circuito somador já implementado mais um circuito comparador.

A ULA deve seguir o seguinte esquema quanto às suas entradas e saídas:



Neste diagrama,

- **A** e **B** são os operandos, ambos de 8 *bits* (e deverão ser interpretados de acordo com a interpretação sem sinal, i.e., inteiros de 0 a $2^8 - 1$);
- **OP** é um *bit* para indicar o tipo de operação: se 0, deve ser calculada a adição **A + B** e se 1 deve ser calculada a subtração **A - B**; a comparação será sempre calculada;

¹Esta data de entrega é apenas uma referência. A entrega no moodle é para sabermos quem concluiu o circuito, para podermos proceder com a avaliação.

²Isso quer dizer que nosso computador só vai processar inteiros sem sinal ...

- \mathbf{R} é o resultado da operação (8 bits);
- “*overflow*” é um *bit*: se seu valor for 1, indica que a operação calculada, interpretando-se os números como sem sinal, resultou em *overflow*.
- $o_{=}$ é um *bit*: seu valor deve ser 1 se e somente se $\mathbf{A} = \mathbf{B}$, na interpretação sem sinal;
- $o_{>}$ é um *bit*: seu valor deve ser 1 se e somente se $\mathbf{A} > \mathbf{B}$, na interpretação sem sinal;
- $o_{<}$ é um *bit*: seu valor deve ser 1 se e somente se $\mathbf{A} < \mathbf{B}$, na interpretação sem sinal.

As saídas $o_{=}$, $o_{>}$ e $o_{<}$ devem estar compatíveis com as entradas correntes (\mathbf{A} e \mathbf{B}), não importando se a operação indicada na entrada OP é adição ou subtração. Além disso, faremos um comparador que funciona apenas para a interpretação sem sinal, para simplificar os circuitos.

2 Modularização

De forma similar ao EP1, organize o circuito em módulos, cada módulo correspondendo a um subcircuito. Neste sentido, além do circuito somador que fizemos na etapa 1, agora teremos o circuito comparador que também deve ser modularizado, e a ULA deverá ser composta pelo circuito somador e pelo circuito comparador. Assim, seu EP deverá ter ao menos os seguintes subcircuitos:

- somador e somabits, como no EP1
- comparador e seus subcircuitos, se for o caso; as entradas do comparador devem ser dois números de 8 bits (interpretados como sem sinal), \mathbf{A} e \mathbf{B} , e as saídas deverão ser os *flags* $o_{=}$, $o_{>}$ e $o_{<}$ como os da ULA, especificados acima
- ULA, como especificado acima. Note que o somador do EP1 tinha duas saídas para indicar *overflow* (nos casos sem e com sinal). Neste EP vocês podem usar o mesmo somador já feito no EP1 e na construção da ULA podem ignorar a saída *overflow (com sinal)* do somador.
- main, para testar a ULA (deve ser possível alterar as entradas \mathbf{A} e \mathbf{B} , e a operação a ser realizada – adição ou subtração). A cada vez que ou \mathbf{A} ou \mathbf{B} são alterados, as saídas \mathbf{R} , “*overflow*”, $o_{=}$, $o_{>}$ e $o_{<}$, devem ficar com valores consistentes com as entradas correntes.

3 Entrega e avaliação

Os circuitos deverão ser feitos na plataforma *CircuitVerse*. O circuito somador do EP1 pode ser reaproveitado integralmente.

Serão avaliados os seguintes aspectos:

- corretude dos circuitos
- aderência à organização proposta, com uma clara identificação (nome consistente) dos pinos de entrada e saída em cada circuito
- clareza na organização geral do circuito como um todo