

Sistemas Digitais

Curso de Engenharia Informática – 1.º ano

2017/2018

Trabalho de Grupo #2

1. Objetivos

Este trabalho é composto por três partes. A primeira parte consiste no projeto e desenvolvimento duma ALU. Na segunda parte do trabalho é pretendido o desenvolvimento dum banco de registos e na terceira parte a interligação entre a ALU e o banco de registos.

2. Enunciado

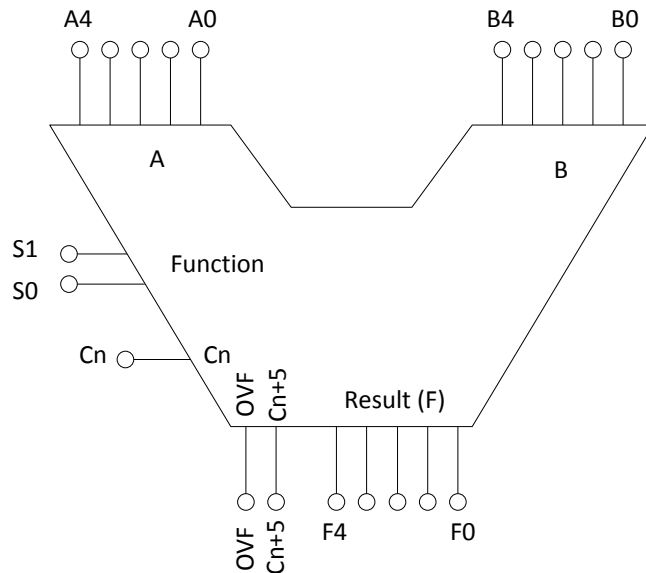
Parte 1

Uma ALU (*Arithmetic and Logic Unit*) é um sistema digital que permite realizar uma série de operações lógicas sobre sequências binárias. Inicialmente os valores a operar são provenientes do banco de registos (de dois dos registos) e a operação a executar depende da sequência de controlo colocada à entrada da ALU. A figura seguinte mostra um esquema da ALU de 5 bits a desenvolver. Os dados para operação são colocados nas entradas A e B (bits $A_4 - A_0$ e $B_4 - B_0$, respetivamente), sendo o resultado da operação colocado nos portos $F_4 - F_0$. As operações a executar pela ALU dependem da sequência de controlo S_1S_0 conforme a tabela da figura abaixo.

Pretende-se o desenvolvimento duma ALU que realize as operações descritas na tabela consoante os sinais de controlo. A ALU a desenvolver deve possuir ainda a entrada C_N de modo a poder ser associada a outro circuito idêntico para construir ALUs com tamanho de palavra maior (p. ex. 10 bits, 15 bits, 30 bits, etc.) e ainda os sinais de saída C_{N+5} e OVR, que correspondem ao bit de transporte da parcela N+5 e ainda ao sinal de *overflow*, respetivamente.

O sinal de *overflow* indica se a capacidade de representação para o número de bits usado na ALU (5 bits) nas operações de soma e subtração foi ultrapassado. Só ocorre *overflow* quando se somam dois números positivos (ou subtração dum número positivo com um

número negativo) e o resultado dá negativo, ou quando se somam dois números negativos (ou subtração dum número negativo com um número positivo) e o resultado dá positivo. O bit OVF deverá ficar a 1 no caso de ocorrer *overflow*.



S1	S0	Operação
0	0	$A + B$
0	1	$A - B$
1	0	$A \text{ and } B$
1	1	$A \text{ xor } B$

A	Input (5 bits)
B	Input (5 bits)
F	Result (5 bits)
Cn	Carry in
Cn+5	Carry out
OVF	<i>Overflow</i>

A operação a realizar pela ALU depende do número de Grupo, como indicado no anexo. No seu relatório coloque a tabela de funcionamento da ALU para o seu caso concreto.

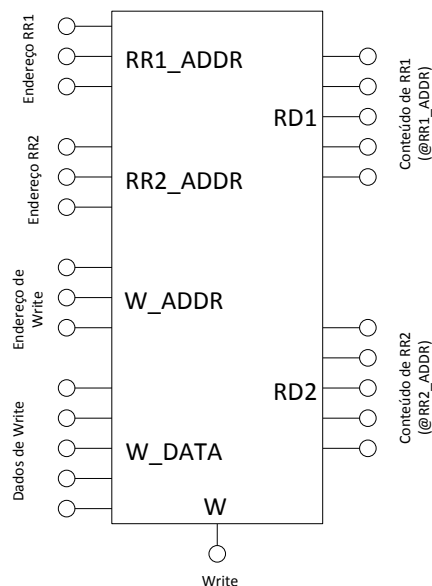
1. Crie uma macro com a ALU.
2. Crie um circuito que utiliza esta macro e ilustre o funcionamento do mesmo com um exemplo de todas as funções realizadas.
3. **Extra 1:** Implemente um mecanismo de *carry look-ahead* de modo a acelerar as operações de soma.
4. **Extra 2:** Utilizando macros desenvolvida para a ALU de 5 bits, construa uma ALU de 10 bits obtida a partir de 2 ALUs de 5 bits devidamente conectadas.

Parte 2

Nesta componente do trabalho pretende-se projetar e desenvolver um banco de registos (*register file*). O banco de registos deve de possuir um total de 8 registos onde cada registo possui 5 bits.

O banco de registos deve permitir ler dois registos em simultâneo, cujos endereços (000 a 111) são colocados nas entradas RR1 e RR2 (*Read Register 1* e *Read Register 2*) sendo o seu conteúdo colocado nas saídas RD1 e RD2 (*Read Data 1* e *Read Data 2*), respetivamente.

Para escrever num dado registo o seu endereço deve de ser colocado na entrada WR_ADDR (*Write Register*) e os dados a escrever no registo na entrada W_DATA (*Write Data*). O registo só deve de ser escrito quando o sinal de controlo W (*Write*) é colocado a 1. A figura abaixo representa um esquema do banco de registos.



1. Crie uma macro com o banco de registos.
2. Crie um circuito que utiliza esta macro e ilustre o funcionamento do mesmo.
3. **Extra 3:** Utilizando a macro desenvolvida para o banco de registos de 5 bits, construa um banco de registos de 10 bits.

Parte 3

Com as macros desenvolvidas para a ALU e para o banco de registos projete um circuito digital no Digital Works onde cada entrada da ALU (A e B) é ligada às saídas de cada um dos registos (RD1 e RD2) e, selecionando os valores para os sinais de controlo S_1S_0 , esta apresente na saída F o resultado da operação.

1. Teste devidamente o funcionamento deste sistema digital e apresente exemplos do seu correto funcionamento.
2. **Extra 4:** Conecte o banco de registo de 10 bits com a ALU de 10 bits de modo a ter os blocos básicos para um processador de 10 bits: Banco de registos + ALU.
O que seria necessário fazer para se ter um processador de 20 bits?

3. Conclusão

No final deve submeter as macros realizadas e os circuitos completos juntamente com um breve relatório usando a ligação disponibilizada na página da disciplina. Qualquer dúvida deve ser colocada no fórum da UC, ou no horário de atendimento aos alunos.

Cotações

Parte 1 - ALU	Macro	5 pontos
	Circuito (com exemplos de funcionamento no relatório)	1 ponto
	Extra 1	1 ponto
	Extra 2	1 ponto
Parte 2 – Banco de Registos	Macro	5 pontos
	Circuito (com exemplos de funcionamento no relatório)	1 ponto
	Extra 3	1 ponto
Parte 3 – Banco de Registos + ALU	Circuito (com exemplos de funcionamento no relatório)	2 pontos
	Extra 4	1 ponto
Relatório	Apresentação, Clareza e Conteúdo	2 pontos

Tabela de funcionamento da ALU

As funções a realizar pela ALU para cada grupo seguem o seguinte método:

1. Dado o número de grupo, verifica-se a sequência de funções a utilizar na tabela seguinte:

Grupo	Seq	Grupo	Seq	Grupo	Seq	Grupo	Seq
1	0,1,2,3	9	0,3,2,1	17	3,1,0,2	25	2,1,3,0
2	2,0,1,3	10	2,0,3,1	18	2,3,1,0	26	1,3,0,2
3	3,0,1,2	11	0,2,1,3	19	3,2,0,1	27	0,3,2,1
4	1,0,2,3	12	3,1,2,0	20	1,0,3,2	28	2,3,0,1
5	2,1,3,0	13	1,3,0,2	21	1,2,0,3	29	1,3,2,0
6	1,2,3,0	14	3,0,2,1	22	2,1,0,3	30	2,0,1,3
7	3,2,1,0	15	0,3,1,2	23	1,3,2,0	31	1,2,0,3
8	0,2,3,1	16	2,3,0,1	24	0,1,3,2	32	3,2,0,1

2. Com esta sequência, verifica-se a função correspondente na tabela da ALU, que é sequencialmente a função a realizar:

S1	S0	Operação
0	0	$A + B$
0	1	$A - B$
1	0	$A \text{ and } B$
1	1	$A \text{ xor } B$

Por exemplo, considere-se que para um dado grupo a sequência é 3,2,0,1. Assim, para $S_1S_0 = 11$, a função a realizar é a 3, que corresponde a “ $A \text{ xor } B$ ”; para $S_1S_0 = 10$, a função a realizar é a 2, que corresponde à função “ $A \text{ and } B$ ”; e assim sucessivamente. A tabela completa seria

$S_2S_1S_0$	Função	Operação
1 1	3	$A \text{ xor } B$
1 0	2	$A \text{ and } B$
0 0	0	$A + B$
0 1	1	$A - B$

Bom trabalho.

Miguel Tavares

2017/12/04