

Relatório Lab. AOC II – ULA 4 bits + Arduino

Atenção, este é um trabalho integrado com a disciplina de AED3.

A data de entrega já está disponível no SGA. Para o laboratório de AC a entrega se dará através da apresentação do trabalho em sala.

Para a disciplina de AED3 um arquivo deverá ser submetido no SGA da disciplina no local determinado.

O trabalho poderá ser realizado em grupos de no máximo 3 pessoas.

O valor do trabalho é de 10 pontos para o Laboratório de AC2, consultar o prof. para AED3.

Todo o material a ser entregue (Lab e AED3) está detalhadamente descrito ao final do trabalho.

Introdução

Neste exercício você deverá criar 2 programas. Um no hardware externo (Arduino) e outro no PC, que será a interface com o usuário. A idéia é ler um programa escrito pelo usuário, transformá-lo em mnemônicos gerando outro programa e finalmente passá-lo ao Hardware externo através da porta serial e realizar algum processamento nesse Hardware. O resultado será observado nos 4 Leds conectados ao Hardware externo.

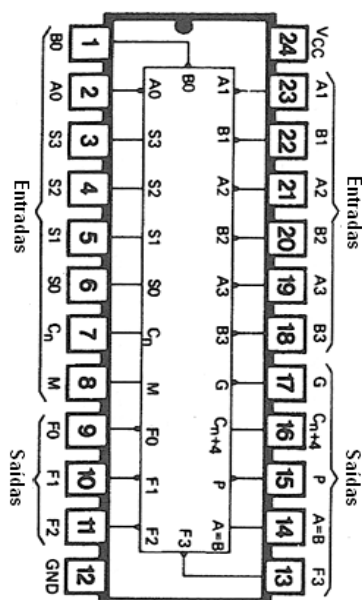
PARTE 1 – O Hardware externo

1) Descrição do trabalho

O Hardware deverá executar todas as instruções da ALU de 4 bits vista (74181).

Você deverá elaborar um programa no Arduino que utilize a entrada serial para receber as entradas necessárias ao funcionamento da ALU (dados e instruções) e as saídas deverão ser 4 Leds ligados aos pinos 13, 12, 11 e 10.

A figura 1 a seguir repete o conjunto de instruções da ALU.



FUNCTION TABLE					
MODE SELECT INPUTS				ACTIVE HIGH INPUTS & OUTPUTS	
S ₃	S ₂	S ₁	S ₀	LOGIC (M = H)	ARITHMETIC** (M = L) (C _n = H)
L	L	L	L	A	<div>Não iremos usar as inst. aritméticas</div>
L	L	L	H	A + B	
L	L	H	L	AB	
L	L	H	H	Logical 0	
L	H	L	L	A/B	
L	H	L	H	B	
L	H	H	L	A ⊕ B	
L	H	H	H	A/B	
H	L	L	L	A + B	
H	L	L	H	A ⊕ B	
H	L	H	L	B	
H	L	H	H	AB	
H	H	L	L	Logical 1	
H	H	L	H	A + B	
H	H	H	L	A + B	
H	H	H	H	A	

L = Low voltage ou 0

H = High voltage ou 1

Seu programa no arduino deverá ser capaz de receber 3 dados da seguinte forma:

Um primeiro valor representando as entradas A0, A1, A2 e A3.

Um segundo valor representando as entradas B0, B1, B2 e B3.

Um terceiro valor representando a instrução desejada, S0, S1, S2 e S3.

Assim, se fornecermos pela comunicação serial na IDE do Arduino os seguintes 3 valores:

1 2 4, estaremos passando para a ALU as seguintes informações:

Valor de $A = 1$, valor de $B=2$ e a instrução desejada=4 ou $S=4$. A ALU projetada no arduino deverá então realizar, conforme o conjunto de instruções da ALU (de acordo com a Fig. 1) a instrução (AandB)negado.

A Figura 2 ilustra estas instruções já retiradas da Figura 1 e agora já renomeadas de acordo com um mnemônico para cada instrução.

Observe que, para não haver confusão nos valores, pode ser interessante usar os números em Hexadecimal, assim, se passarmos ao Arduino os seguintes dados A A A, o significado será:

Valor de $A = 10$, valor de $B=10$ e a instrução desejada ou $S=10$. A ALU projetada no arduino deverá então realizar, conforme o conjunto de instruções da ALU (e de acordo com a Fig. 1) a instrução B.

A saída da ALU deverá estar presente nas saídas:

Pino 13 = F3

Pino 12 = F2

Pino 11 = F1

Pino 10 = F0

Pergunta: Qual seria o significado de passarmos para o Arduino os seguintes valores “3 5 6”, como ficariam os LEDs ligados na Saída?

Resp:

Entrada dos valores para $A = 0011$

Entrada para os valores de $B = 0101$

Entrada para os valores de $S = 0110$

Como $S=0110$ indica que queremos a instrução $A \text{ xor } B$, a saída F seria 0110 ou os leds dos pinos 11 e 12 acesos.

Função	Mnemônico	Código Hexa
A'	An	0
$(A+B)'$	nAoB	1
$A'B$	AnB	2
0 Lógico	zeroL	3
$(AB)'$	nAeB	4
B'	Bn	5
$A \oplus B$	AxB	6
AB'	ABn	7
$A' + B$	AnoB	8
$(A \oplus B)'$	nAxB	9
B	B	A
AB	AB	B
1 lógico	umL	C
$A+B'$	AoBn	D
$A+B$	AoB	E
A	A	F

Figura 2: Instruções e Mnemônicos
(ativos em 1- high)

Parte 2 – O Software no PC

1) Descrição do trabalho

Você deverá criar um programa em C, C++, C# ou Java que transforme um texto nas instruções a serem executadas e permita a sua execução linha a linha através do console. Para isso, o programa deverá inicialmente ler um arquivo contendo um texto original com os mnemônicos (instruções a serem executadas) e gerar um segundo texto, onde cada linha seja transformada nos valores que serão disponibilizados para a porta serial e conseqüentemente o Arduino. Esse segundo texto deverá ser um arquivo gravado com os respectivos valores a serem enviados para a porta serial porém no formato hexadecimal.

Você deverá utilizar o conjunto de instruções que a ULA possui ilustrados nas Figuras 1 e 2. A Figura 3 ilustra um pequeno exemplo de código a ser transformado. A Figura 4 ilustra o programa a ser gerado. Os nomes dos arquivos indicados nas Figuras 3 e 4 correspondem aos nomes que você deverá utilizar no programa para leitura e escrita. A Figura 5 ilustra a arquitetura completa do sistema proposto.

```
inicio:  
A=5;  
B=2;  
An;  
Bn;  
ABn;  
A=D;  
AB;  
fim.
```

Figura 3: Exemplo do programa de teste
"74181.alu"

```
520  
525  
527  
D2B
```

Figura 4: Programa gerado
"74181.hex"

Como se vê, seu programa deverá transmitir à porta serial o programa de teste. Cuidado que o programa a ser enviado deverá ser a partir do arquivo contendo o programa gerado (Programa no formato .hex, Figura 4) e não o programa fonte original (programa no formato .alu, Figura 3).

O que você deve observar é que para o teste a tecla "enter" será pressionada diversas vezes e, a cada "enter" dado no teclado, uma instrução deverá ser executada. Atenção, as linhas de atribuição de valores não contam como instruções, ou seja no programa exemplo das Figuras 3 e 4, daríamos apenas 4 "enter"s porque ele realiza apenas 4 instruções, apesar de possuir 3 atribuições de valores às variáveis A e B.

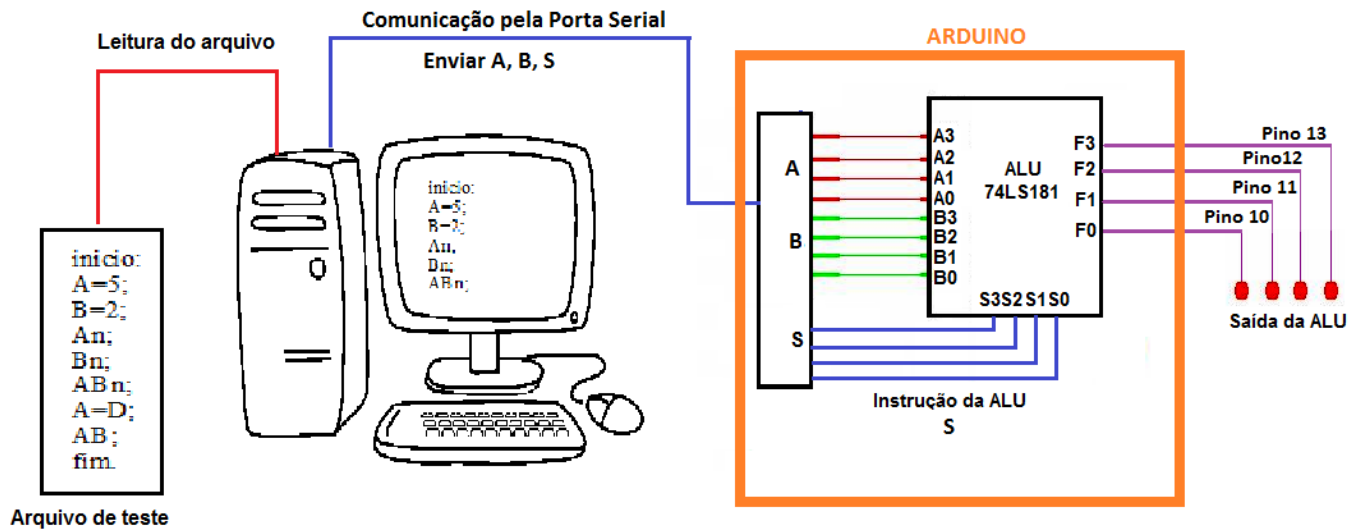


Figura 5: Arquitetura do sistema proposto

O ciclo de execução da máquina pode ser entendido através da Figura 6 a seguir, onde cada "enter" pressionado no teclado corresponderá à execução de uma instrução.

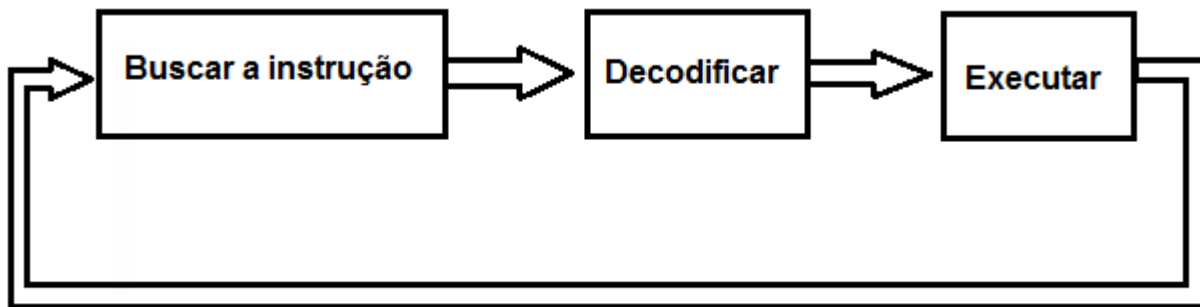


Figura 6: Ciclo de execução de uma instrução

2) O que apresentar ao final do trabalho

Para o Laboratório de AC2

- 1) Um programa no Arduino que simule uma ALU (74181) e receba os valores dos dados e instruções através da porta serial.
- 2) Um programa de acesso em C/C++/Java (com muitos comentários!) que:
 - a) **leia um programa fonte (com os dados e os mnemônicos), você deverá criar um programa fonte de teste. Durante a aula um outro programa será utilizado para verificação do trabalho.**
 - b) **gere um arquivo hexa correspondente aos dados e instruções e**
 - c) **envie dados e instruções através da porta Serial para a ALU.**

Lembre-se de só executar a instrução ao ser dado um "enter" do teclado. O programa de teste possuirá o nome "74181.alu" e você só terá acesso a ele no momento do teste. O formato será o mesmo descrito na Figura 3 e os mnemônicos da Figura 2. O programa gerado deverá possuir o nome "74181.hex". Para o seu teste crie seu próprio programa fonte, lembre-se de procurar testar todas as instruções possíveis.

Cada grupo fará a apresentação dos programas, (será a nota desse relatório, não haverá entrega pela internet, apenas a apresentação.)

Cada grupo deverá estar com os programas em um pendrive (Programa do Arduino e programa em C++, Java ou C#).

Eu irei testar individualmente os programas de cada grupo em uma máquina do Laboratório, alunos que participaram do trabalho mas ausentes na apresentação, não terão nota.

Grupos que não estiverem com os programas não terão nota no relatório, o mesmo acontecendo com trabalhos copiados.

Para a disciplina de AED3

- 1) Deverá ser submetido pelo SGA no local determinado, um arquivo de log no formato txt com o nome de "74181.log".

As seguintes informações deverão estar presentes neste arquivo:

- O dia e horário em que a tradução do programa do formato '.alu' para o '.hex' foi realizada.
- A quantidade total de linhas do programa .alu e do programa gerado .hex.
- A quantidade de atribuições a variáveis encontradas no arquivo .alu.
- A quantidade de instruções encontradas no arquivo .alu.
- O tempo total necessário para a leitura, tradução e gravação do programa (de .alu para .hex).

Comece já, você nunca terá tanto tempo!