

Lab AC2

Preparativo para o interpretador da ALU, Comunicação do PC com um Hardware externo

Este relatório será elaborado em duas aulas de laboratório.

Na primeira aula deverá ser realizado um programa para o arduino.

Na segunda aula um programa no PC deverá ser escrito e esse programa deverá se comunicar com o arduino programado com o programa da aula anterior.

A apresentação do programa final é presencial (ele será apresentado ao professor ao final da segunda aula) portanto, a presença nas duas aulas é necessária e será computada.

Primeira Aula:

As seguintes ações deverão ser realizadas para essa aula:

- Verificar se a máquina possui a IDE do Arduino e abri-la.
- Em uma pasta conhecida da sua máquina, baixar e descompactar o arquivo Rel05_Envia_Arduino.zip.
- Abrir o arquivo Introducao_Arduino.pdf que estará dentro da pasta descompactada.

Após conhecer todo o material acima, que deverá ser realizado:

- Considerar os seguintes Leds ligados aos seguintes pinos do Arduino:

Saída do Arduino	13	12	11	10	9	8	7	6	5	4	3	2
Pino Num:												
Cor do Led												

- Escrever um programa na IDE do arduino que receba diferentes letras e ligue as saídas de acordo com o funcionamento de um semáforo.

Considere o o cruzamento abaixo ilustrado pela Figura 1 e onde não é permitida nenhuma conversão (à direita ou à esquerda) e imagine cada grupo de 3 leds da barra acima representando um determinado semáforo.

Exemplo:

S1 = led13, led12 e led11

S2 = led9, led8 e led7

S3 = led6, led5 e led4

Onde S1, S2 e S3 representam os 3 semáforos da Figura 1

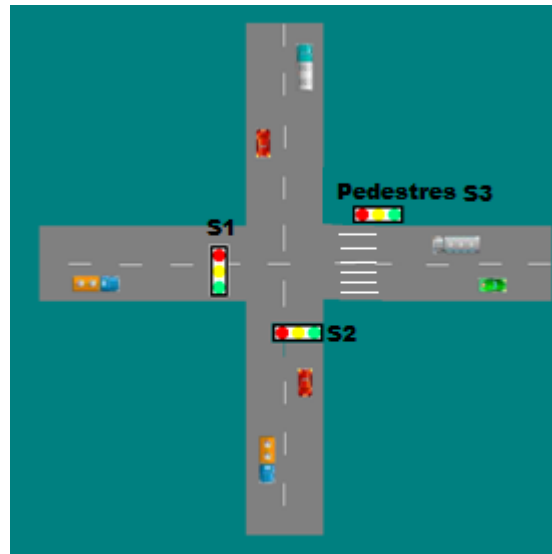


Figura 1 – Exemplo do cruzamento

Com esta associação de Leds e semáforos, você deverá escrever um programa no Arduino que receba cada estado dos semáforos e os acione ou seja:

Vamos considerar um estado inicial “a”, os leds que deverão estar acesos são:

Led13, Led7 e Led4 ou seja, S1 fechado, S2 aberto e S3 aberto.

Nesse contexto, o próximo estado “b” deveria ser:

Led 13, Led8 e Led5 ou seja, S1 fechado, S2 no amarelo e S3 também no amarelo

Seguindo este raciocínio, você deverá elaborar todos os próximos estados até retornarmos ao estado inicial “a”.

Portanto, seu programa deverá receber pela porta serial de comunicação os valores dos estados que você determinou e ligar os desligar os leds de acordo com os valores enviados.

Segunda Aula:

- 1) De posse do programa anterior e já carregado no Arduino, você deverá elaborar um programa em C, C++, C# ou Java que envie a sequência de estados (a, b, c etc) que irá acionar os leds de acordo com o funcionamento dos semáforos evitando acidentes e/ou situações perigosas.

Atenção: A lógica deverá estar no seu programa e não no Arduino. O programa que deverá estar no arduino simplesmente deverá receber alguma coisa pela porta serial e acionar um determinado padrão

de LEDs. Toda a temporização deverá estar contida no seu programa e não no Arduino, a base de tempo deverá ser a tecla “enter”.

Resumindo

- 2) Para os semáforos você poderá utilizar os Leds da forma que quiser. Mas lembre-se que existem dois semáforos para veículos e um para pedestres, nesse cruzamento não é permitida nenhuma conversão dos veículos (Uma sugestão é usar as saídas conforme sugerido nos exemplos)
- 3) As transições deverão ser controladas no teclado pela tecla “enter”. Essa é uma pequena máquina de estados finita (FSM) onde as transições são controladas pelo teclado e cada estado corresponde a uma determinada saída.

O que apresentar ao professor na segunda aula:

Nos primeiros 30 minutos cada grupo fará a apresentação dos programas nos arduinos.

Durante a aula cada grupo deverá elaborar um programa em C++, Java ou C# que envie os valores necessários para as transições.

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.

O restante da aula será com o próximo relatório a ser divulgado.

Obs.: mais um simulador pode ser obtido em:

<https://circuits.io/search/?q=aocpuc>