

Sistemas Digitais I	Projeto	Data	MS
PCS3115			

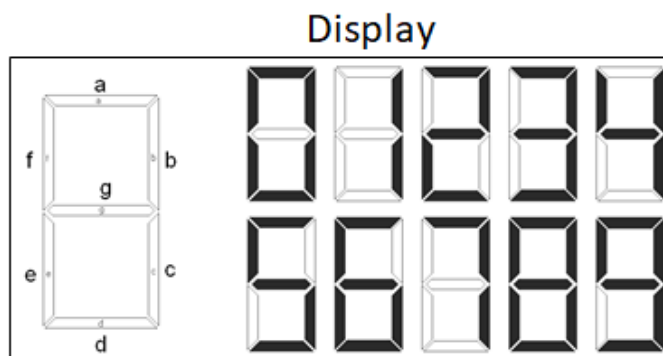
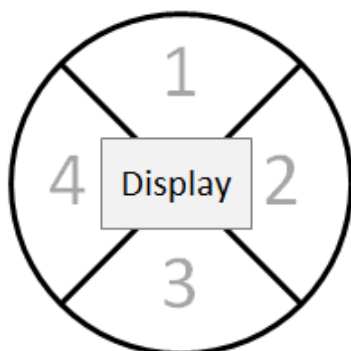
**PROJETO 1.** Praticado por botequeiros de todas as idades acima de 18 anos, a Purrinha (<https://pt.wikipedia.org/wiki/Porrinha>) é um jogo clássico que combina estratégia, raciocínio matemático, e amigos com tempo sobrando. As regras desse tradicional eSport multijogador são as seguintes:

1. Cada jogador recebe 3 palitos, moedas ou qualquer outra coisa de pequeno porte
2. Em cada rodada, os jogadores colocam uma certa quantidade (*num*) de palitos na mão, e então deixam essa mão fechada sobre a mesa.
3. Cada jogador então dá um "palpite" (*guess*) de qual é a soma dos palitos nas mãos de todos os jogadores, incluindo a sua. Os palpites costumam ser lançados em uma certa ordem (por exemplo, sentido horário), e não pode haver repetição dos palpites dados.
4. Após todos lançarem seus palpites, todos os jogadores abrem a mão e conta-se o número de palitos. Ganha a rodada o jogador que acertar esse total.

Neste projeto, o objetivo é implementar o hardware básico para esse jogo, capaz de implementar os mecanismos acima na forma de um circuito puramente combinatório, e dando suporte a exatamente 4 jogadores. A entidade que representa o jogo é a seguinte:

```
entity purrinha is
  port (
    num4, num3, num2, num1:      in  bit_vector(1 downto 0);
    guess4, guess3, guess2, guess1: in  bit_vector(3 downto 0);
    display:                     out bit_vector(6 downto 0));
end entity;
```

Nesta entidade: os valores de "num4" a "num1" representam os números de palitos na mão de cada jogador, podendo, portanto, assumir os valores 0 a 3; "guess4" a "guess1" representam os palpites de cada jogador, podendo ir de 0 a 12 (embora não seja proibido pelo jogo "queimar" seu palpite ao escolher um número maior que 12); e *display* representa a saída de 7 bits que ativa os LEDs abcdefg correspondentes no display de 7 segmentos mostrado abaixo. Esse display é colocado no meio da mesa para anunciar o vencedor da rodada (1, 2, 3, ou 4), ou então que nenhum jogador acertou (coloca-se um "-" no display, acendendo apenas o LED g, com o código "0000001"), ou que dois ou mais jogadores acertaram mas deram o mesmo palpite (coloca-se um "E" no display, acendendo-se todos os LEDs exceto b e c, com o código "1001111")



Os exemplos abaixo mostram alguns valores de entrada e saída esperados para esse circuito (os valores de “num” e “guess” são representados como números inteiros):

num				guess				display	Comentário
4	3	2	1	4	3	2	1		
0	0	0	0	6	6	15	0	1	Vitória do jogador 1: ignore que 4 e 3 deram o mesmo palpite errado
0	1	2	3	6	6	15	0	E	Erro, pois dois jogadores (4 e 3) acertaram dando o mesmo palpite
1	1	3	3	6	6	15	0	-	Ninguém acertou o palpite (a soma deu 8)

### **Instruções para Entrega**

Você deve acessar o link específico para o projeto dentro do tópico “Projetos” no e-Disciplinas, já logado com seu usuário e senha, o que o levará à página apropriada do juiz eletrônico. Nessa atividade, você pode enviar **um único arquivo** codificado em UTF-8. O nome do arquivo não importa, mas sim a descrição VHDL que está dentro: a entidade na sua solução deve ser idêntica à “*purrrinha*” apresentada neste enunciado ou o juiz não irá processar corretamente seu arquivo. Caso você use outras entidades para auxiliar na sua solução, coloque-as nesse mesmo arquivo

Quando acessar o link no e-Disciplinas, o navegador abrirá uma janela para envio do arquivo. Selecione-o e envie para o juiz. Jamais recarregue a página de submissão pois seu navegador pode enviar o arquivo novamente, o que vai ser considerado pelo juiz como um novo envio e pode prejudicar sua nota final. Caso desista do envio, simplesmente feche a janela. Depois do envio, a página carregará automaticamente o resultado do juiz, quando você poderá fechar a janela. A nota dada pelo juiz é somente para a submissão que você acabou de fazer.

O prazo para a submissão das soluções no Juiz é aquele definido no e-disciplinas. O Juiz aceitará até 5 submissões para cada atividade deste projeto. Sua submissão será corrigida imediatamente e sua nota será apresentada. A maior nota dentre as submissões será considerada. Apesar dessa facilidade de fazer algumas submissões, **não use o juiz como ferramenta de debug**: como boa prática de engenharia, faça seus testbenchs e utilize o GHDL ou o EDAPlayground para validar suas soluções **antes** de submetê-las ao juiz. Se você não fizer isso, há grandes chances de que você desperdice oportunidades de submissão para erros simples, que poderiam ser corrigidos antes da submissão... O testbench em si não deve fazer parte da submissão, pois o juiz tem seu próprio testbench.

**Importante**: como o circuito buscado deve ser puramente combinatório, você **NÃO PODE** utilizar comandos para construção de circuitos sequenciais, como **process**. A presença desse tipo de comando no seu código levará à rejeição da submissão, atribuindo-se a ela nota zero mesmo que ela esteja funcionando corretamente. Você pode (e, comumente, precisa) usar comandos sequenciais no seu testbench, o que não deve causar qualquer problema na sua nota dado que o seu testbench não fará parte da sua submissão.