

# Batalha Naval

## Projeto - MC613

Felipe Pessina e Rafael Figueiredo Prudencio

7 de Maio de 2018

# Conteúdo

<b>1</b>	<b>Diagrama de Blocos</b>	<b>3</b>
<b>2</b>	<b>Descrição Funcional</b>	<b>4</b>
2.1	File . . . . .	4
2.2	Memória . . . . .	5
2.3	Draw Map . . . . .	5
2.4	Input/Output . . . . .	5
2.4.1	Mouse . . . . .	5
2.4.2	Monitor . . . . .	5
2.5	Player . . . . .	5

# 1 Diagrama de Blocos

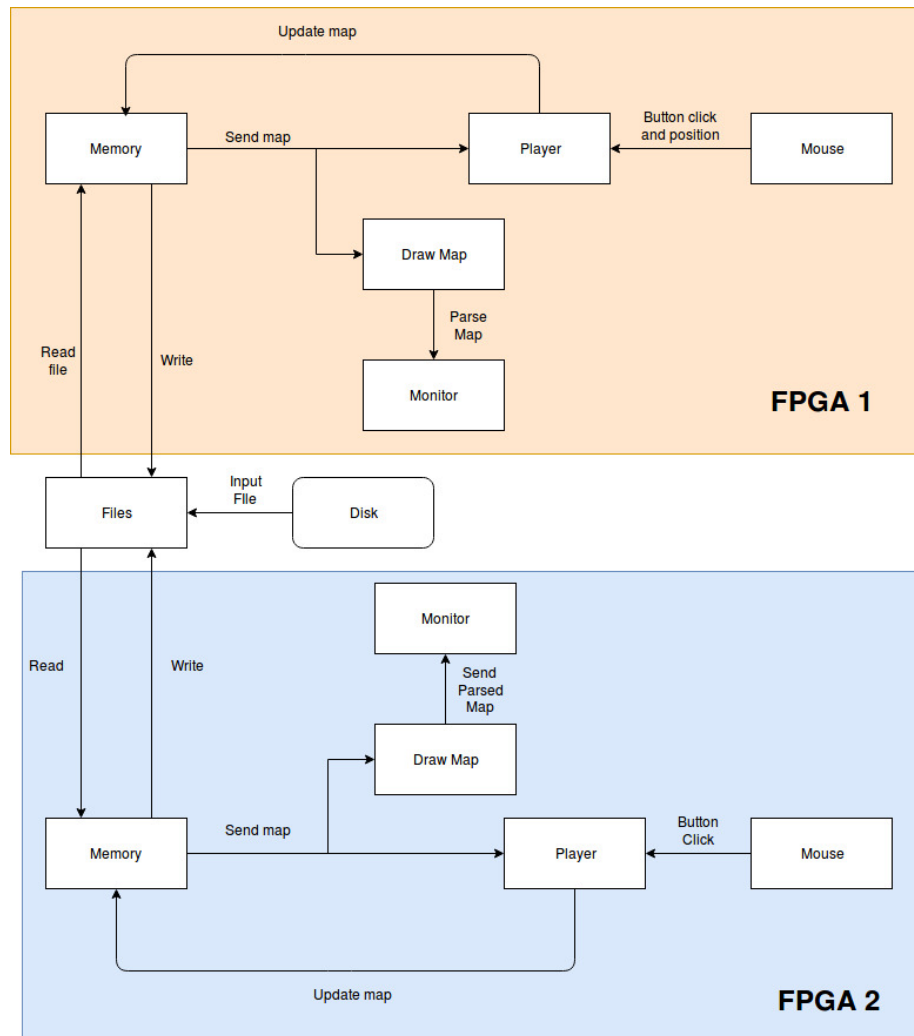


Figura 1: Diagrama de blocos de alto nível para uma implementação de batalha naval.

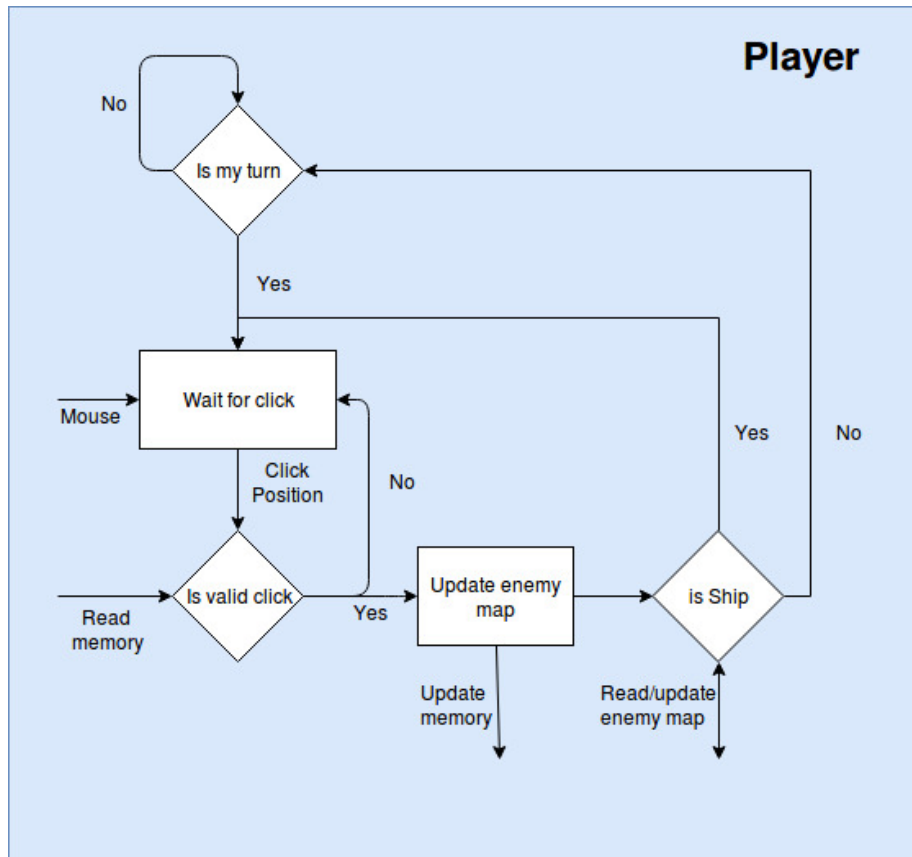


Figura 2: Máquina de estados para a entidade Player.

## 2 Descrição Funcional

### 2.1 File

O bloco File da Figura 1 representa um arquivo que auxilia no compartilhamento dos mapas entre as duas placas. Serão mantidos dois mapas, um para cada jogador. Cada mapa é representado por uma matriz que terá as informações referentes ao estado de cada posição do mapa. Os estados possíveis são: navio, água, navio descoberto e água descoberta. Dessa forma, um jogador pode ler do mapa do oponente os estados navio descoberto e água descoberta para manter um mapa na memória do que ele sabe a respeito do oponente. Por sua vez, o oponente pode ler todos os estados para manter na respectiva memória os navios que possui e as posições que já foram descobertas do seu mapa. As duas placas podem solicitar uma leitura ou realizar uma escrita para atualizar o estado dos mapas na respectiva memória, ou enviar da sua memória o mapa do jogo para

o arquivo.

## **2.2 Memória**

Para cada placa FPGA, há uma instância da entidade Memory da Figura 1. Cada entidade armazena os dois mapas do arquivo na memória e permite com que seja feita a leitura e a escrita na memória. Uma escrita na memória deve trigger uma alteração que deve ser feita no arquivo e a leitura é feita conforme necessário.

## **2.3 Draw Map**

A entidade Draw Map da Figura 1 é responsável pela leitura do mapa do jogo e escrita dele na memória do monitor. São necessárias duas dessas entidades, uma para cada placa que deve atualizar o monitor de cada jogador. Esta entidade tem como entrada os mapas da memória e como saída os valores dos pixels com os endereços onde devem ser gravados na memória do monitor. Os valores dos pixels serão determinados a partir do estado de cada posição nos mapas mantidos na memória. Um jogador tem acesso a todos os estados do seu mapa, mas apenas aos estados navio descoberto e água descoberta do mapa do seu oponente.

## **2.4 Input/Output**

### **2.4.1 Mouse**

Cada placa FPGA terá um mouse como dispositivo de entrada. A entidade Play receberá a posição do mouse sempre que houver um click.

### **2.4.2 Monitor**

Cada jogador também precisa de um monitor, que manterá uma imagem do estado do jogo. A escrita na memória do monitor será feita pela entidade Draw Map.

## **2.5 Player**

A entidade Player na Figura 1 é responsável pelo tratamento das jogadas. Um diagrama mais detalhado da entidade Player está indicado na Figura 2. Cada player tem acesso às informações do seu mouse. O turno de cada jogador será armazenado no arquivo e, conseqüentemente na memória das duas FPGAs. A entidade Player primeiro deve verificar se é sua vez para então esperar por clicks do jogador. Após um click, deve-se verificar se é uma jogada válida. Caso não seja, devemos continuar esperando por clicks. Caso seja, devemos atualizar o mapa na memória e verificar se acertamos um navio inimigo ou não. Em caso de acerto, podemos realizar uma nova jogada. Caso não seja, devemos escrever

no mapa de memória que é o turno do jogador oponente e passarmos para o estado de esperar nosso turno.