



**UNIVERSIDADE FEDERAL DA FRONTEIRA SUL**

CIÊNCIA DA COMPUTAÇÃO

MARCO AURÉLIO LUNARDI

Disciplina: Circuitos Digitais

Batalha Naval

Chapecó

12 de julho de 2023

## **1. APRESENTAÇÃO.**

O projeto de batalha naval proposto na matéria de Circuitos Digitais do segundo semestre de Ciência da Computação foi realizado usando os seguintes softwares: BooleanTT, usado para simplificar as equações obtidas; Logisim, onde foi montado o circuito com auxílio das ferramentas do programa; e Tinkercad, software online onde foi montado o circuito usando protoboards e CI's. Todas essas ferramentas fizeram parte do desenvolvimento do projeto em suas respectivas partes: desenvolvimento da codificação, tabela verdade, simplificação e construção do circuito.

## 2. DESCRIÇÃO DA SOLUÇÃO:

### 2.1 Codificações

Na codificação das posições da matriz 4x4, os valores originais das posições (Figura 1) foram embaralhados, e a partir disso, teve origem uma nova matriz 4x4(Figura1.1), selecionada no arquivo disponibilizado pelo professor(foi escolhido a codificação 13).

Linhas				
00	0000	0001	0010	0011
01	0100	0101	0110	0111
10	1000	1001	1010	1011
11	1100	1101	1110	1111
Colunas	00	01	10	11

**Figura 1.**

No final, cada posição faz “referência” a outra posição diferente da tabela original. Por exemplo: na Figura 1, a posição 0101, recebe na Figura 1.1 a “posição codificada” 0111, não representando mais o valor anterior.

Linhas				
00	0101	0000	1111	1110
01	1010	0111	0100	1000
10	1100	0110	0010	1011
11	0001	1101	1001	0011
Colunas	00	01	10	11

**Figura 1.1.**

## 2.2 Tabela verdade

Entradas Codificadas				Saídas decodificadas nas posições 4x4					
E1	E2	E3	E4		S1	S2	S3	S4	
0	1	0	1		0	0	0	0	0
0	0	0	0		0	0	0	1	1
1	1	1	1		0	0	1	0	2
1	1	1	0		0	0	1	1	3
1	0	1	0		0	1	0	0	4
0	1	1	1		0	1	0	1	5
0	1	0	0		0	1	1	0	6
1	0	0	0		0	1	1	1	7
1	1	0	0		1	0	0	0	8
0	1	1	0		1	0	0	1	9
0	0	1	0		1	0	1	0	10
1	0	1	1		1	0	1	1	11
0	0	0	1		1	1	0	0	12
1	1	0	1		1	1	0	1	13
1	0	0	1		1	1	1	0	14
0	0	1	1		1	1	1	1	15

**Figura 2.**

A tabela verdade foi construída com base na tabela codificada da Figura 2. A partir do método de Soma de Produtos (SoP), é possível obter as expressões. A linha 0, por exemplo, representa a posição na matriz 0000 (primeira linha e primeira coluna) e sua posição codificada é 0101. Pegando cada resultado (S1, S2, S3 e S4) e usando o método SoP, conseguimos desta forma obter as equações e posteriormente simplificá-las, como veremos no próximo tópico.

## 2.2 Simplificações

Equações gerais	$EQ1 = (A * B * C' * D') + (A' * B * C * D') + (A' * B' * C * D') + (A * B' * C * D) +$
	$(A' * B' * C' * D) + (A * B * C' * D) + (A * B' * C' * D) + (A' * B' * C * D)$
	$EQ2 = (A * B' * C * D') + (A' * B * C * D) + (A' * B * C' * D') + (A * B' * C' * D') +$
	$(A' * B' * C' * D) + (A * B * C' * D) + (A * B' * C' * D) + (A' * B' * C * D)$
	$EQ3 = (A * B * C * D) + (A * B * C * D') + (A' * B * C' * D') + (A * B' * C' * D') +$
	$(A' * B' * C * D') + (A * B' * C * D) + (A * B' * C' * D) + (A' * B' * C * D)$
	$EQ4 = (A' * B' * C' * D') + (A * B * C * D') + (A' * B * C * D) + (A * B' * C' * D') +$
	$(A' * B * C * D') + (A * B' * C * D) + (A * B * C' * D) + (A' * B' * C * D)$
Equações simplificadas	$S1 = A'.C.D' + A.B.C' + B'.D$
	$S2 = A'BC'D' + AC'D + B'C'D + A'CD + AB'D'$
	$S3 = A'BC'D' + B'CD + AB'C' + A'B'C + ABC$
	$S4 = ABC'D + B'C'D' + A'CD + B'CD + BCD'$

**Figura 3.**

A Figura 3 mostra as equações gerais de cada saída e suas simplificações. Por exemplo a EQ1, em sua forma simplificada a partir da Simplificação de Karnaugh, resulta em S1, e assim respectivamente para cada equação. Para obter a forma simplificada de cada equação foi usado o aplicativo BooleanTT, disponível na PlayStore e AppStore. A partir dos minitermos de cada equação e aplicando no aplicativo, teve origem as seguintes simplificações:

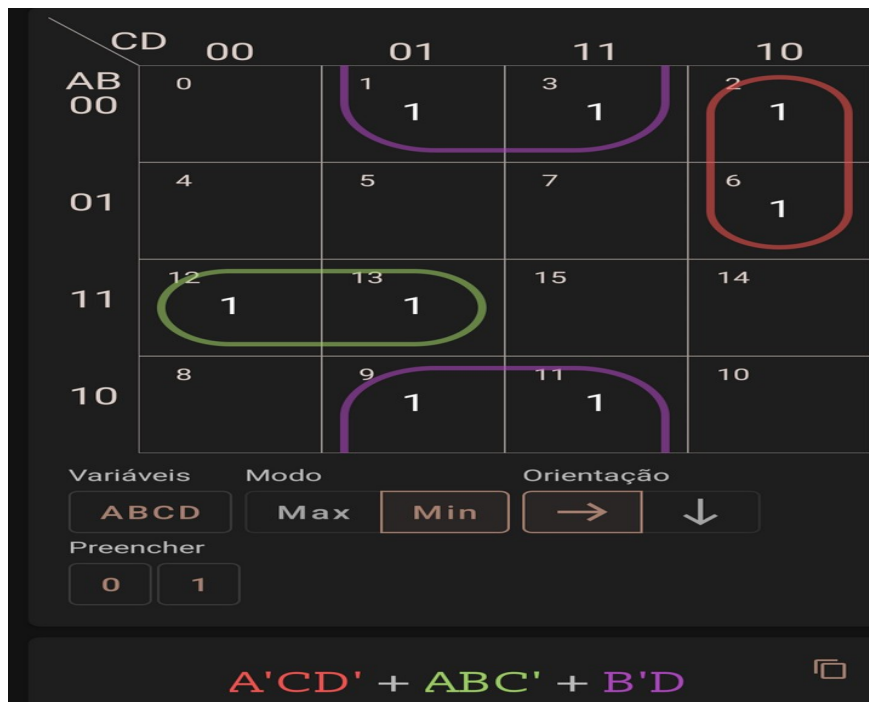


Figura 3.1. Simplificação (S1) da Equação 1 (EQ1).

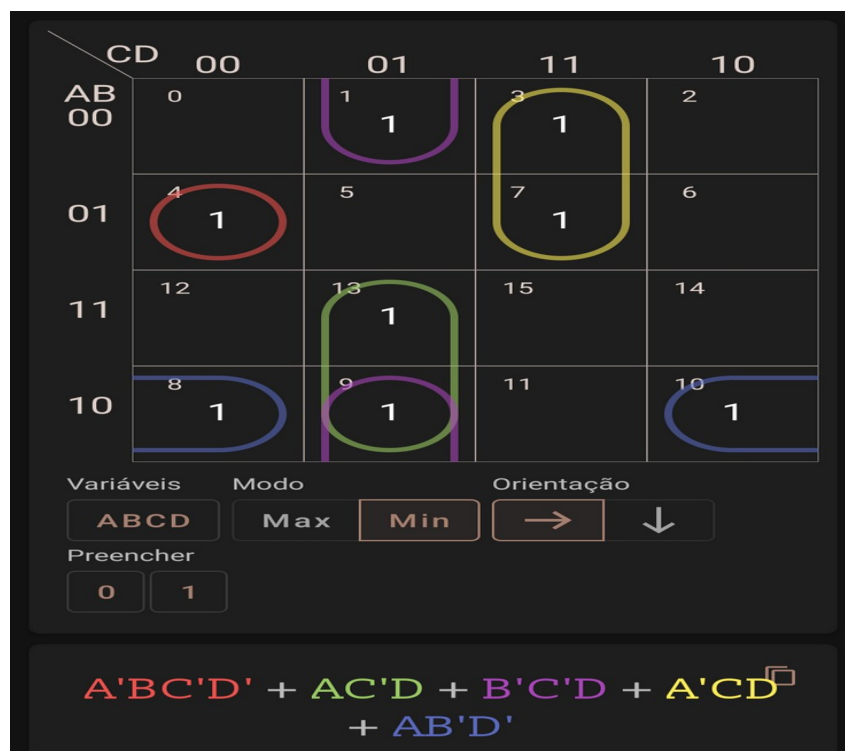


Figura 3.2. Simplificação (S2) da Equação 2 (EQ2).



Figura 3.3. Simplificação (S3) da Equação 3 (EQ3).

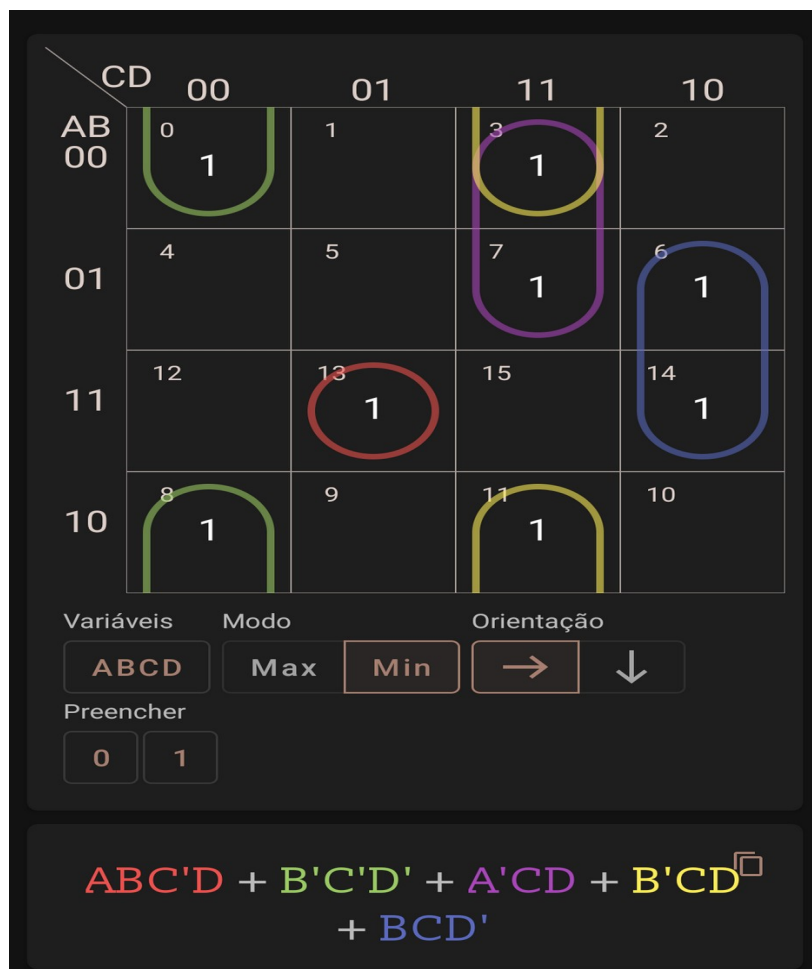


Figura 3.4. Simplificação (S4) da Equação 4 (EQ4).

## 2.4 Circuito usando portas lógicas (logisim)

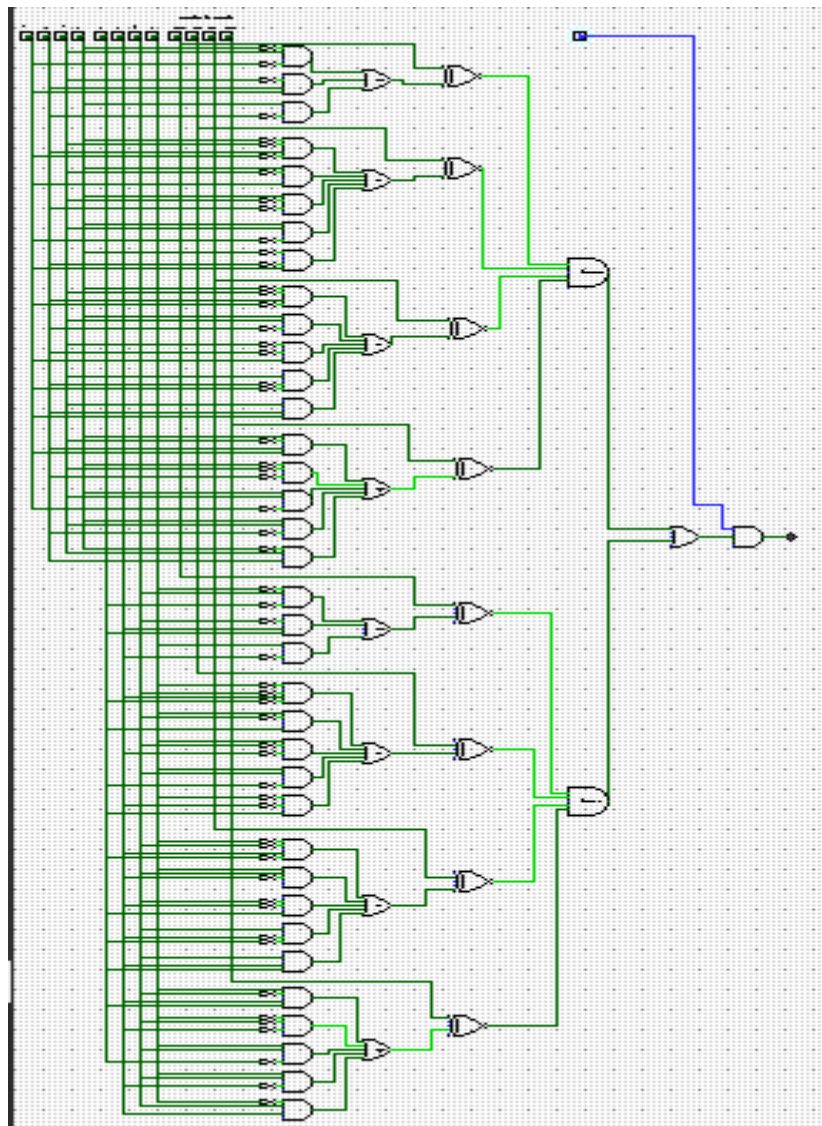
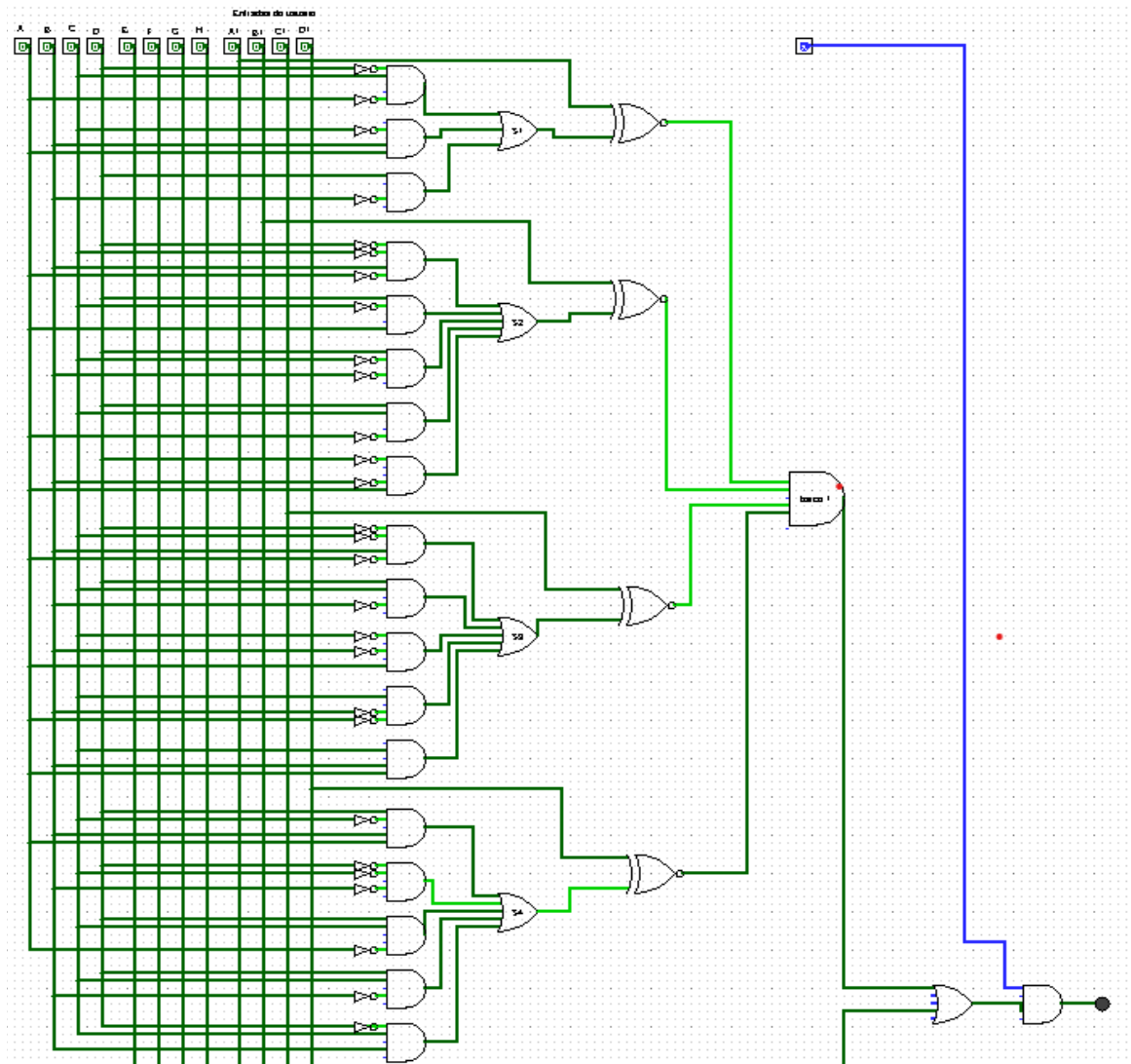
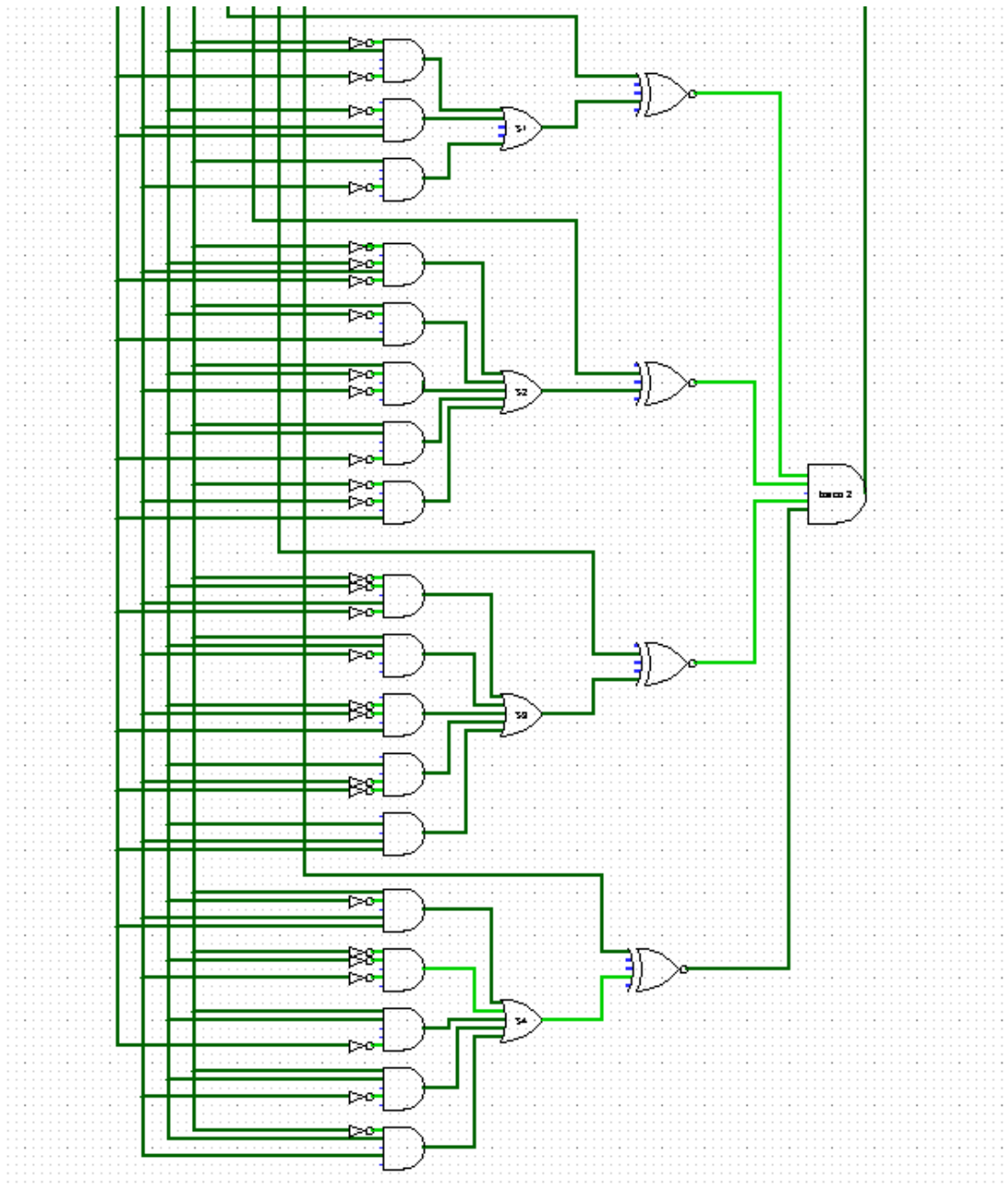


Figura 4. Circuito logisim completo





**Figura 4.1 Barco 1 do circuito do logisim**



**Figura 4.2. Barco 2 do circuito do logisim**

As entradas A,B,C,D correspondem ao primeiro barco, já as entradas E,F,G,H ao segundo e as entradas A1,B1,C1,D1 são as do usuário(Para melhor visualização abrir o projeto). Na primeira parte do circuito está o Barco 1 (figura 4.1) com suas devidas equações S1,S2,S3 e S4. A segunda parte contém o Barco 2 (figura 4.2) com as mesmas equações. As saídas desses barcos estão conectadas a uma “OR”, se uma dessas saídas for 1 o resultado da “OR” será 1. O resultado está conectado a uma AND junto com

o botão de atirar. Se o botão estiver em 1 e o resultado for 1 (se algum barco for atingido), o led irá ligar.

## 2.5 Circuito usando CIs (tinkercad) .

Disponível em: [https://www.tinkercad.com/things/0zch2Ogf6m0?sharecode=oVKVajXubqCqnPiZHgAnI2H\\_ZrBxfxKKU3qLCjlsWOk](https://www.tinkercad.com/things/0zch2Ogf6m0?sharecode=oVKVajXubqCqnPiZHgAnI2H_ZrBxfxKKU3qLCjlsWOk)

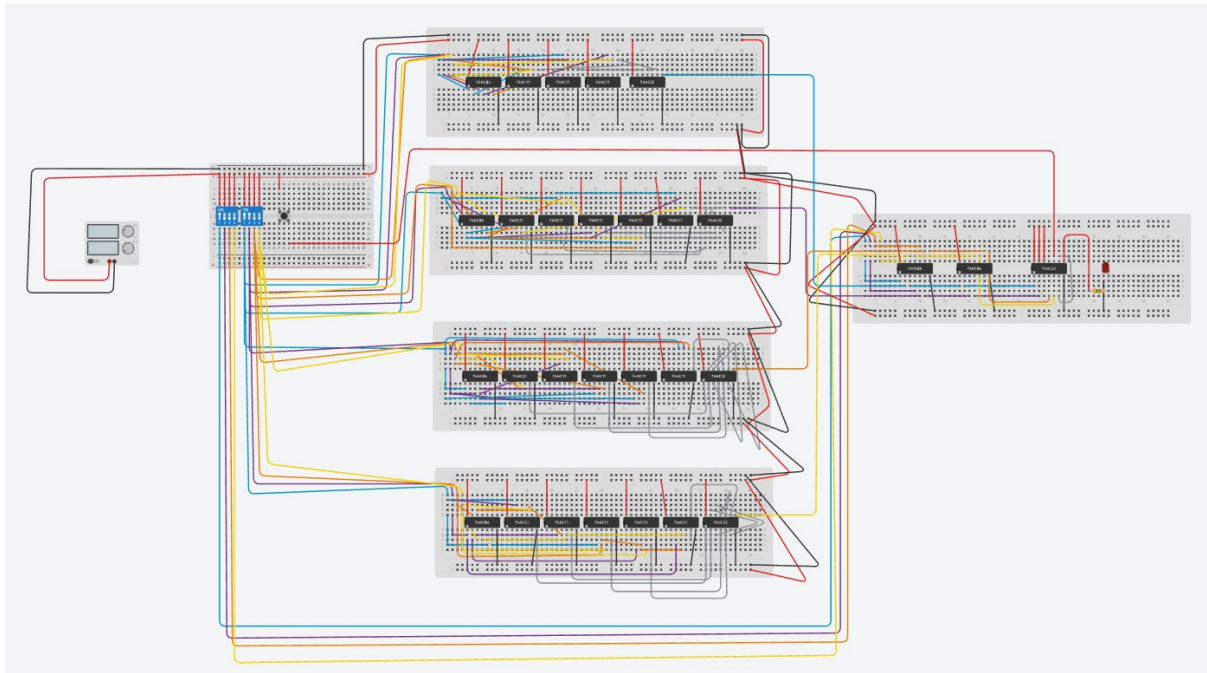
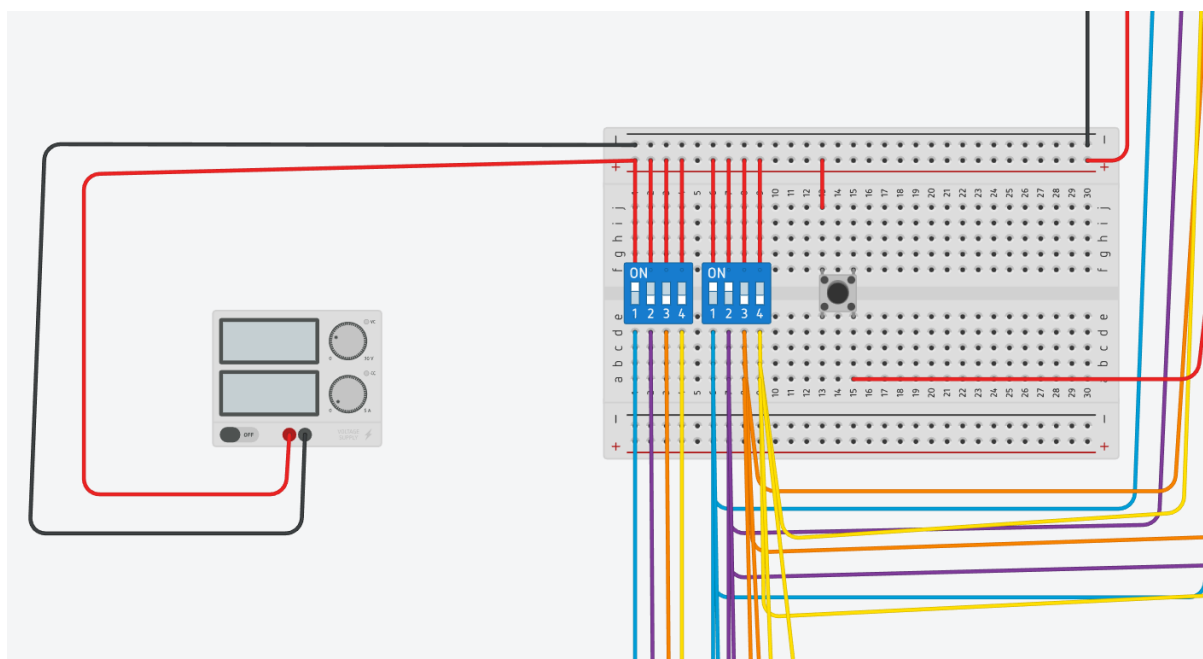


Figura 5. Foto do Circuito Completo.



**Figura 5.1. Foto da protoboard de jogadas do barco e do usuário e também do botão de tiro.**

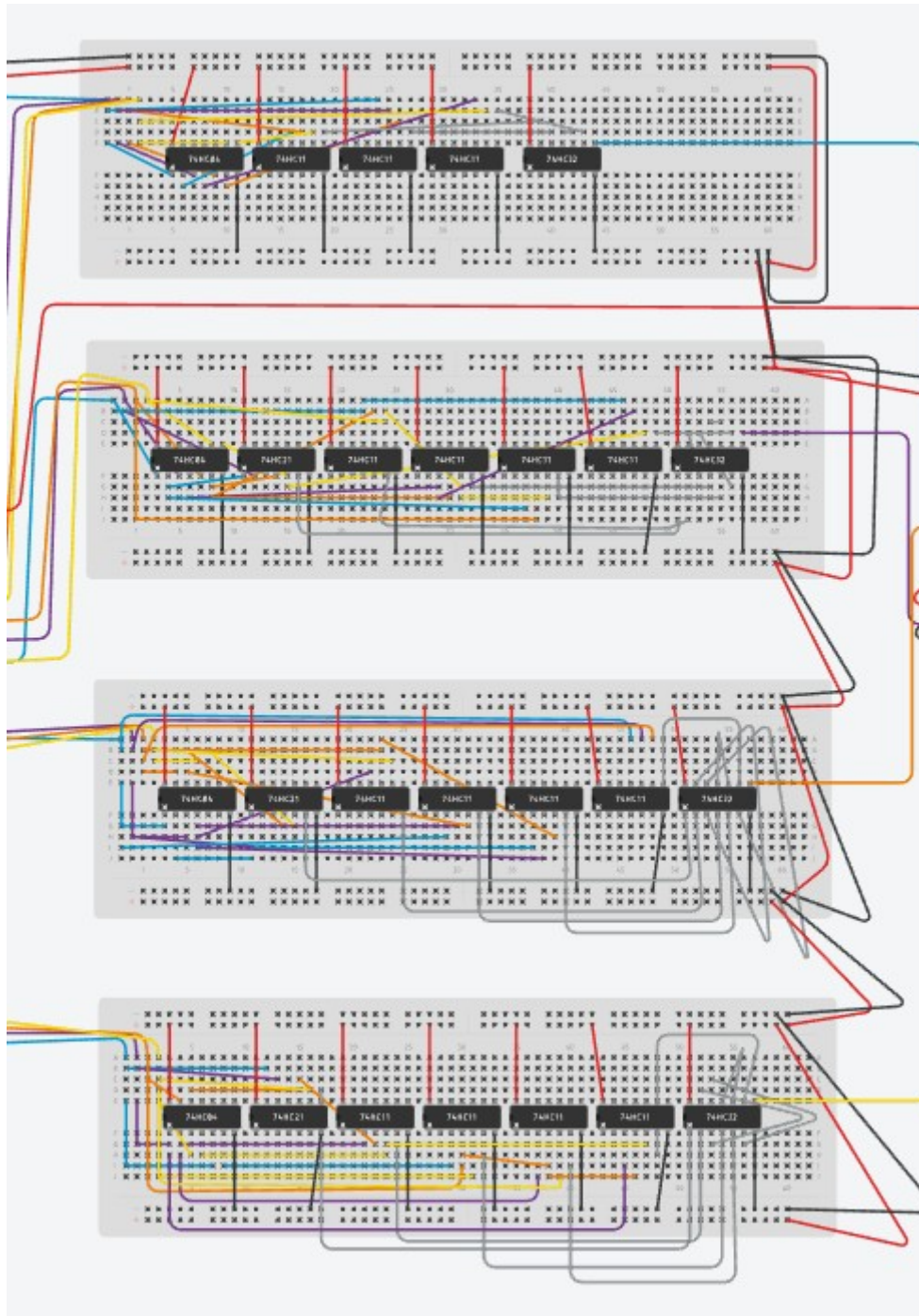
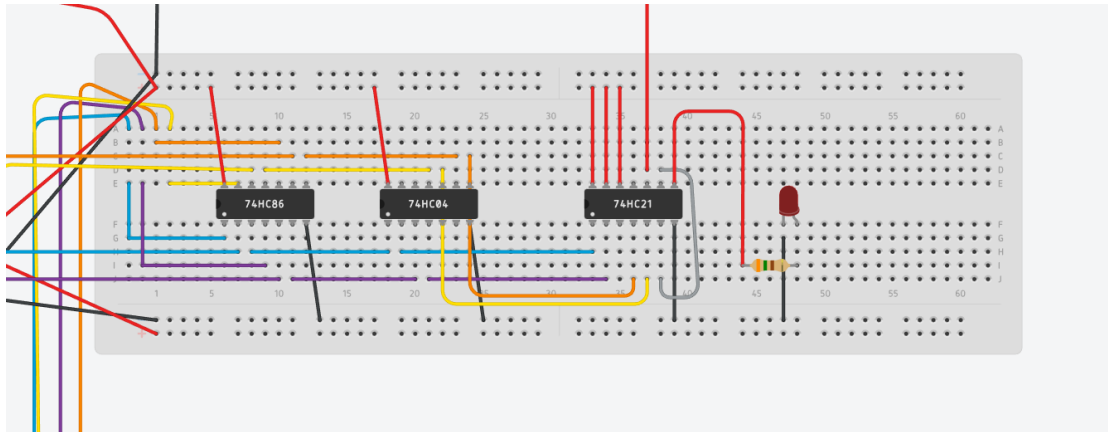


Figura 5.2. Foto das protoboards com as equações simplificadas.



**Foto 5.3. Foto da protoboard final com as comparações das equações e a jogada do usuário.**

## 2.6 Execução do jogo

[https://drive.google.com/file/d/1sn0\\_mM3zTJvdWvrHKLbTvA6gUN1K0s1T/view?usp=drive\\_link](https://drive.google.com/file/d/1sn0_mM3zTJvdWvrHKLbTvA6gUN1K0s1T/view?usp=drive_link)

No desenvolvimento do circuito, as entradas A,B,C e D foram relacionadas com as cores dos fios. A entrada A é representada pela cor Azul, a B cor roxa, C cor laranja e a D amarela. A primeira protoboard da Figura 5.2 representa S1, a segunda protoboard representa S2 e assim segue para as demais. A última protoboard a direita da Figura 5.3 foi utilizada para realizar a comparação da posição do barco com a jogada do usuário. A comparação foi feita usando as portas XOR e NOT, se as entradas forem iguais o resultado da XOR será 0, logo com a NOT ficará 1. Caso a posição do barco codificada coincida com a jogada pelo usuário o led irá acender.

## 3. CONCLUSÃO

O trabalho de desenvolver o jogo batalha naval colocou em prática grande parte do conteúdo aprendido durante o semestre de Circuitos Digitais. O trabalho proporcionou novas experiência com hardware e software, além do conhecimento que adquirimos de forma desafiadora.

A plataforma Tinkercad é muito útil para a realização do trabalho, pois simula muito bem a construção em uma protoboard real, com maior agilidade para a simulação do circuito. Os softwares Logisim e BooleanTT auxiliam na simplificação das expressões e na obtenção do circuito.