



**UNIVERSIDADE FEDERAL DA FRONTEIRA SUL
CAMPUS CHAPECÓ
CIÊNCIA DA COMPUTAÇÃO**

**ALISSON LUAN DE LIMA PELOSO
NADINI CRISTINA DA ROSA**

**BATALHA NAVAL:
UM JOGO FEITO EM CIRCUITOS DIGITAIS**

**CHAPECÓ
2019**

1. APRESENTAÇÃO

O projeto “Batalha Naval”, solicitado na matéria de Circuitos Digitais como trabalho avaliativo, trata de uma implementação do jogo de batalha naval utilizando circuitos digitais. Com o objetivo de criar uma codificação para o posicionamento dos barcos na batalha naval, tratamos nosso “mar” como uma matriz 4x4 com posições 0,1,2,3,4,...,16 (0000,0001,0010,...,1111) e aplicamos uma codificação única, na qual cada posição recebe um código binário de 4 bits diferentes. Assim, ao tentar adivinhar onde está localizado o barco na matriz, é necessário acertar o código de codificação.

Foram utilizados os programas: Logisim, Quartus II, Tinkercad e a Altera DE1 (hardware do projeto). O Logisim foi utilizado para gerar o circuito de codificação e a aplicação do circuito completo. No Quartus II, foi implementado o circuito da mesma forma como no Logisim, usado como plataforma de testes e ainda para transmissão da codificação para o Altera DE1. Já o Tinkercad, como simulação da montagem das protoboards com a codificação utilizada.

2. DESCRIÇÃO DA SOLUÇÃO

Para descrever a solução do projeto serão utilizadas explicações sobre a codificação do circuito, sua tabela-verdade, suas simplificações, o circuito utilizando portas lógicas pelo aplicativo Logisim e o circuito utilizando CIs pelo aplicativo Tinkercad.

2.1. CODIFICAÇÃO

As posições codificadas se deram a partir de uma matriz 4x4, onde os valores para a codificação foram escolhidos de forma aleatória, gerando a segunda matriz, como na figura abaixo.

A tabela codificada foi usada para a realização da tabela-verdade do circuito final.

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

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

(Figura 1 - Tabela Original vs Tabela Codificada)

2.2. TABELA VERDADE

Para a entender a tabela-verdade do circuito é importante saber que as entradas decodificadas são as entradas de ambos navios e as entradas codificadas são as que o usuário terá que inserir para ter a chance de acertar algum dos navios.

Entradas Decodificada				Entradas Codificadas			
A	B	C	D	S1	S2	S3	S4
0	0	0	0	0	1	1	1
0	0	0	1	0	1	0	1
0	0	1	0	0	0	0	1
0	0	1	1	1	0	1	0
0	1	0	0	1	1	1	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	0	0
0	1	1	1	1	0	0	1
1	0	0	0	0	0	1	1
1	0	0	1	0	0	0	0
1	0	1	0	0	0	1	0
1	0	1	1	1	1	0	1
1	1	0	0	1	0	1	1
1	1	0	1	1	0	0	0
1	1	1	0	1	1	1	0
1	1	1	1	1	1	0	0

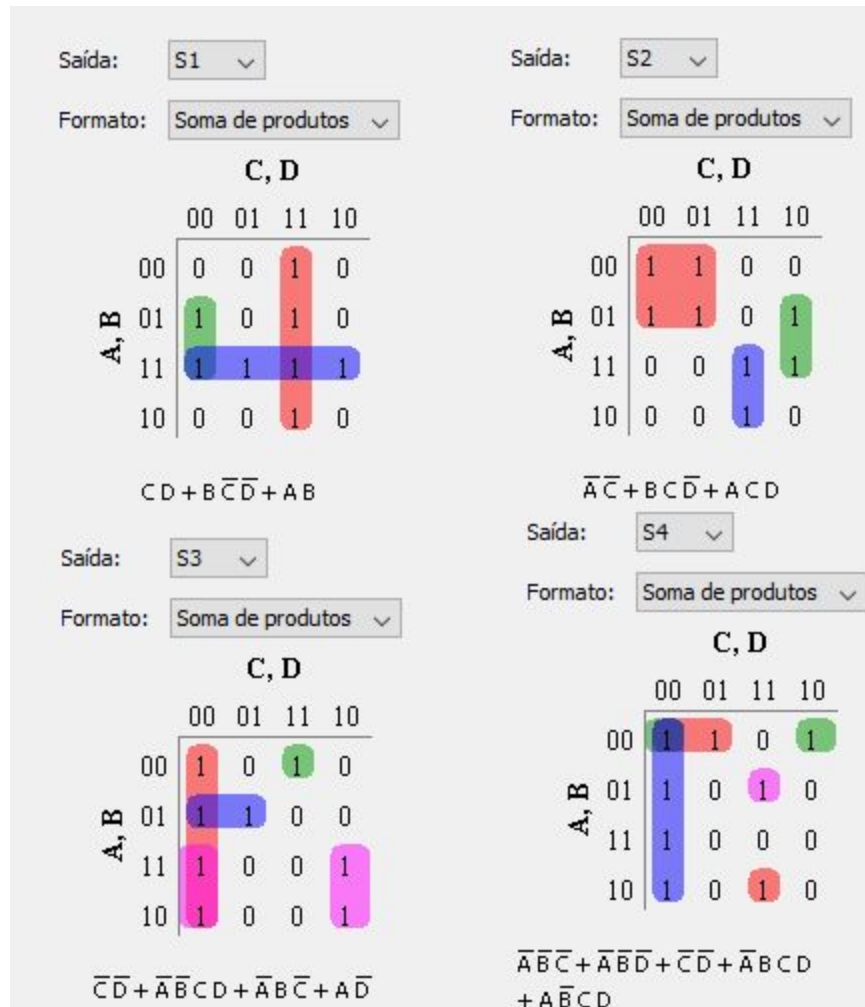
(Figura 2 - Tabela-Verdade)

2.3. SIMPLIFICAÇÕES

Foi inserido a tabela-verdade no programa Logisim para gerar as equações simplificadas presentes na Figura 3. Para realizar esse procedimento também era possível utilizar o Mapa de Karnaugh localizando os minitermos e, assim, encontrando a expressão simplificada da codificação.

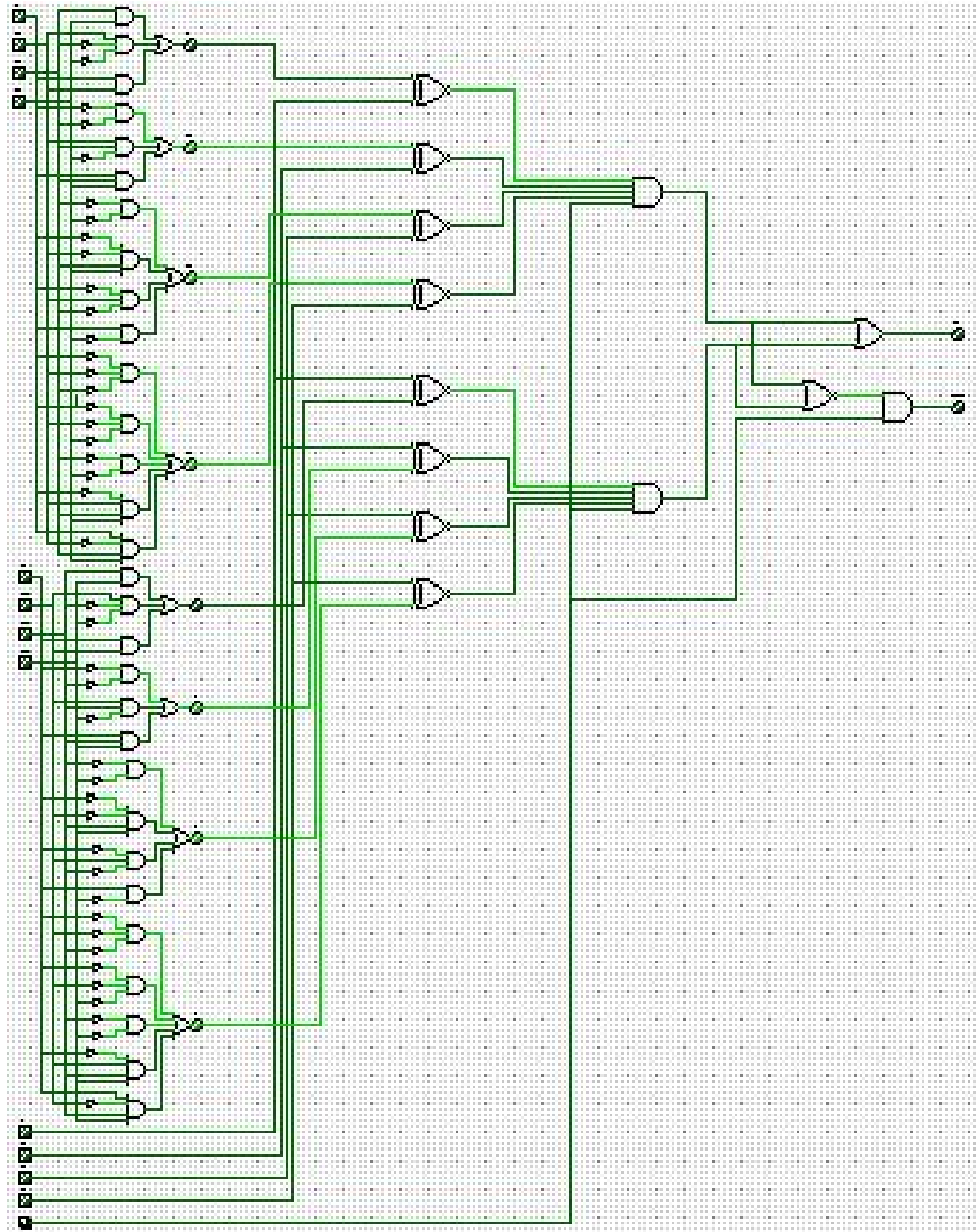
Equações Simplificadas	S1 = $CD + B \sim C \sim D + AB$
	S2 = $\sim A \sim C + BC \sim D + ACD$
	S3 = $\sim C \sim D + \sim A \sim B C D + \sim A B \sim C + A \sim D$
	S4 = $\sim A \sim B \sim C + \sim A \sim B \sim D + \sim C \sim D + \sim A B C D + A \sim B C D$

(Figura 3 - Expressões de simplificação)

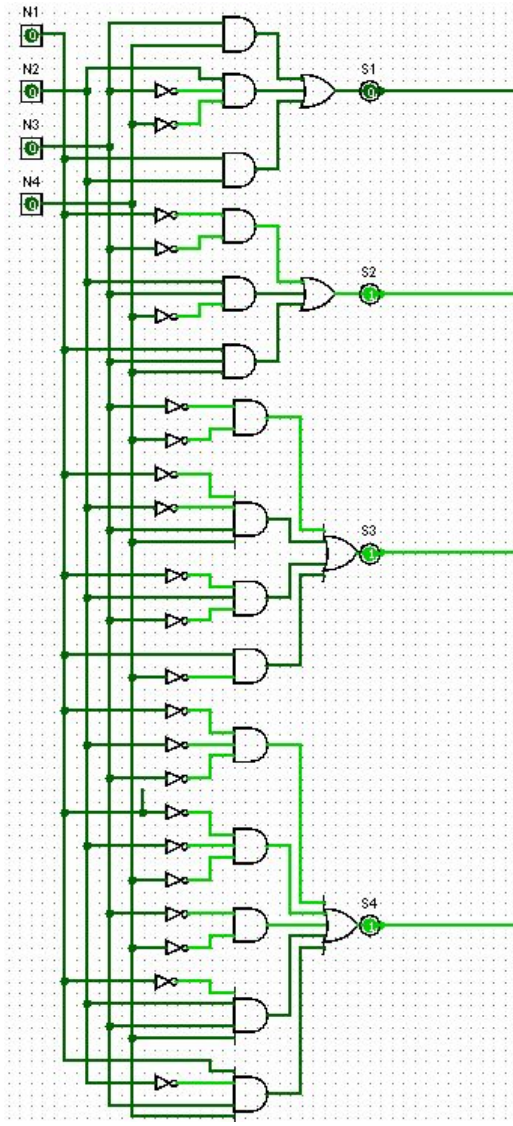


(Figura 4 - Mapas de Karnaugh gerados pelo Logisim)

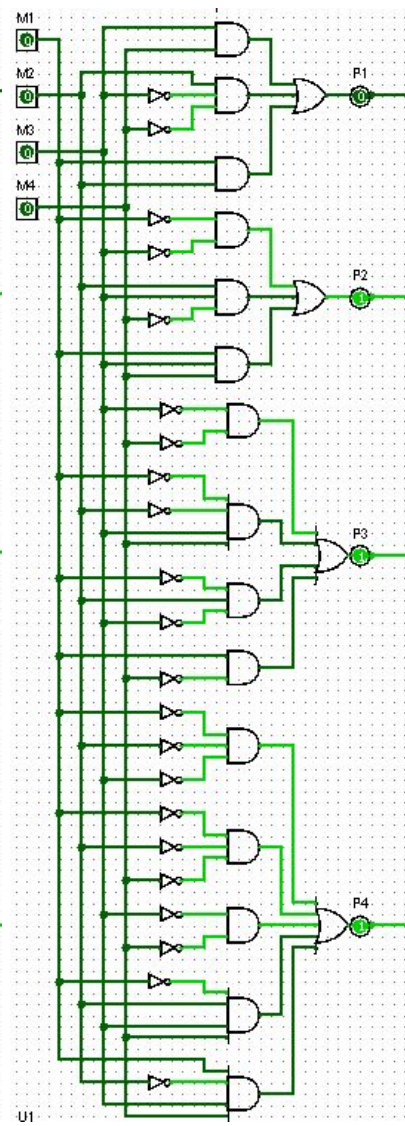
2.4. CIRCUITO USANDO PORTAS LÓGICAS (LOGISIM)



(Figura 5 - Circuito completo no Logisim)



(Figura 5.1 - Barco 1 (Logisim))



(Figura 5.2 - Barco 2 (Logisim))



(Figura 5.3 - Usuário e Disparo (Logisim))

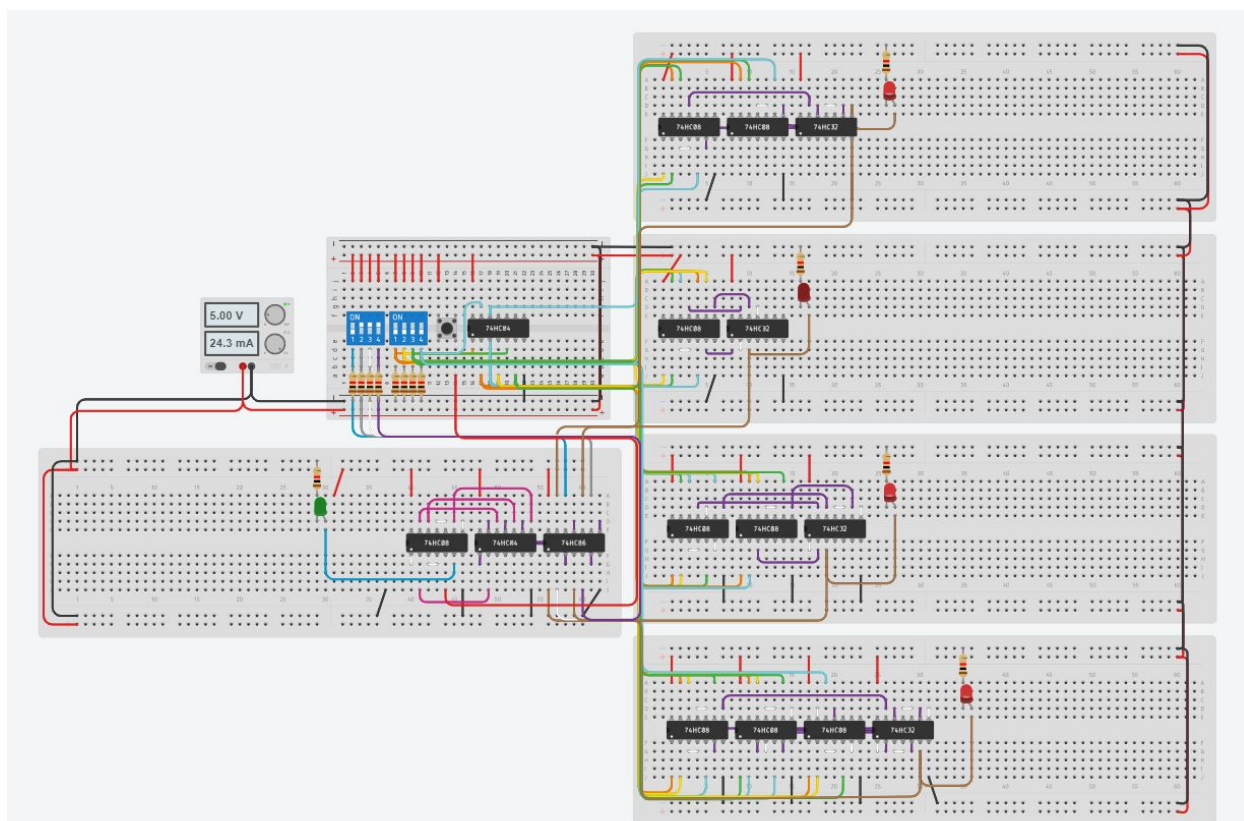
O Barco 1 (N1,N2,N3,N4) possui a codificação ligada nas saídas (S1,S2,S3,S4) como na Figura 5.1 e o Barco 2 (M1,M2,M3,M4) possui a mesma codificação, porém ligada nas saídas (P1,P2,P3,P4), como na Figura 5.2.

As entradas U1, U2, U3, U4 e o botão D (Figura 5.3) representam as entradas do usuário e o disparo. Elas estão sendo comparadas com as saídas codificadas dos barcos utilizando XNORs e depois direcionadas para uma porta AND que só terá resultado “1” se a entrada do usuário estiver de acordo com algum dos barcos e o botão de disparo estiver pressionado.

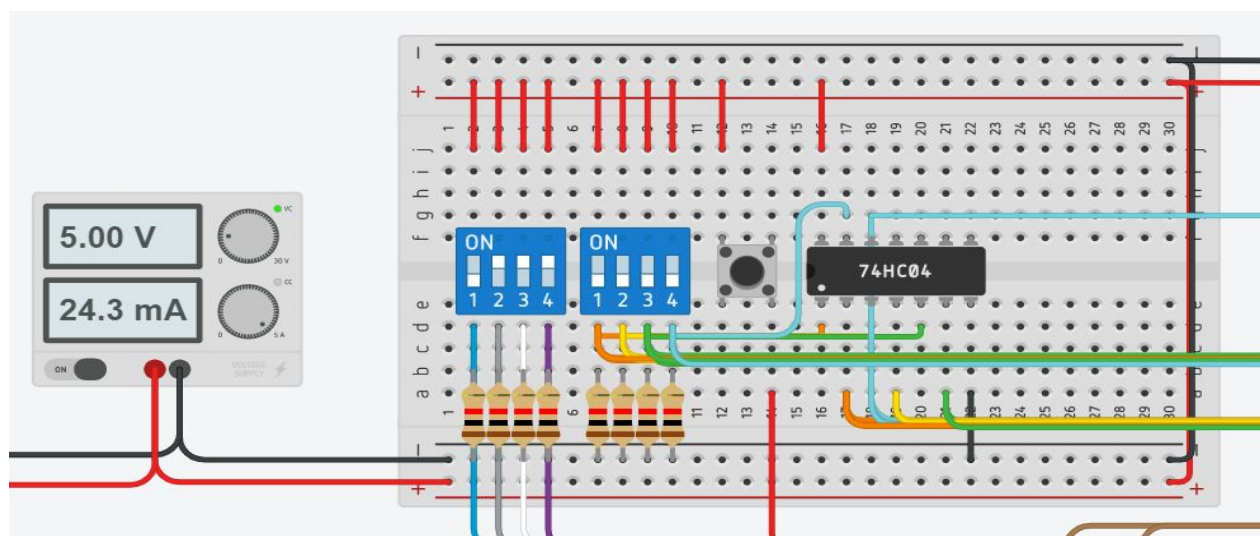
2.5. CIRCUITO USANDO CIs (TINKERCAD)

Para implementar o circuito no Tinkercad, foi necessário ter o circuito no Logisim completo, sendo assim, mais simples e prático para iniciar o projeto do circuito. Para esta parte do projeto foi utilizado apenas a implementação de um dos barcos. A figura 6 mostra o projeto completo e funcional (pode-se acessá-lo a partir deste [link](#)).

No início, conecta-se a protoboard em uma fonte de alimentação e nela insere-se oito chaves (quatro para o uso do navio e quatro para o uso do usuário) e um botão de disparo. As chaves do navio são também ligadas à uma porta NOT para uso posterior.

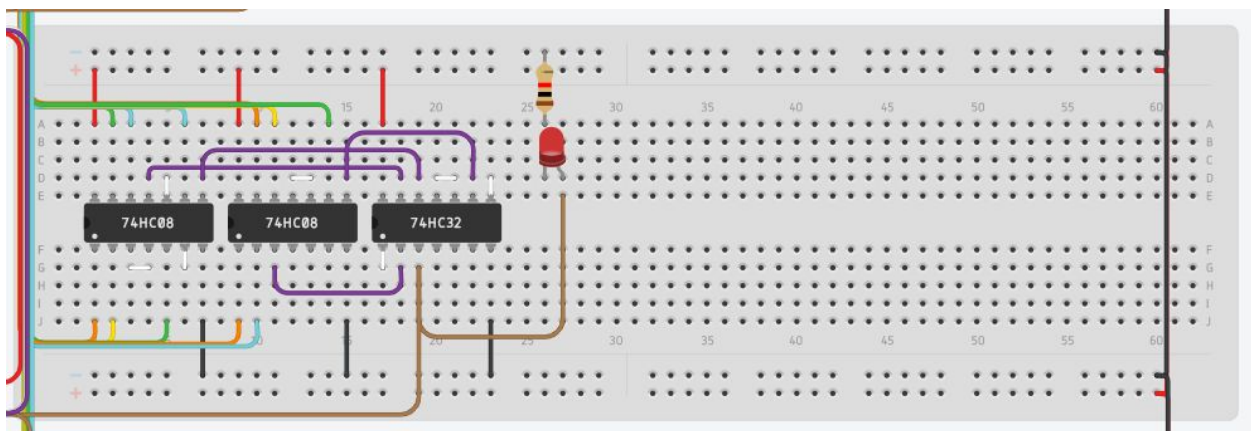
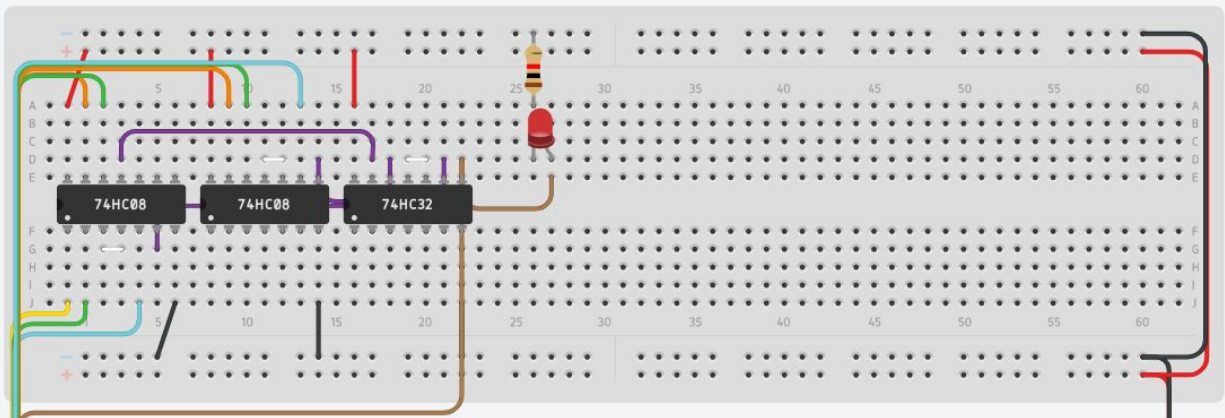
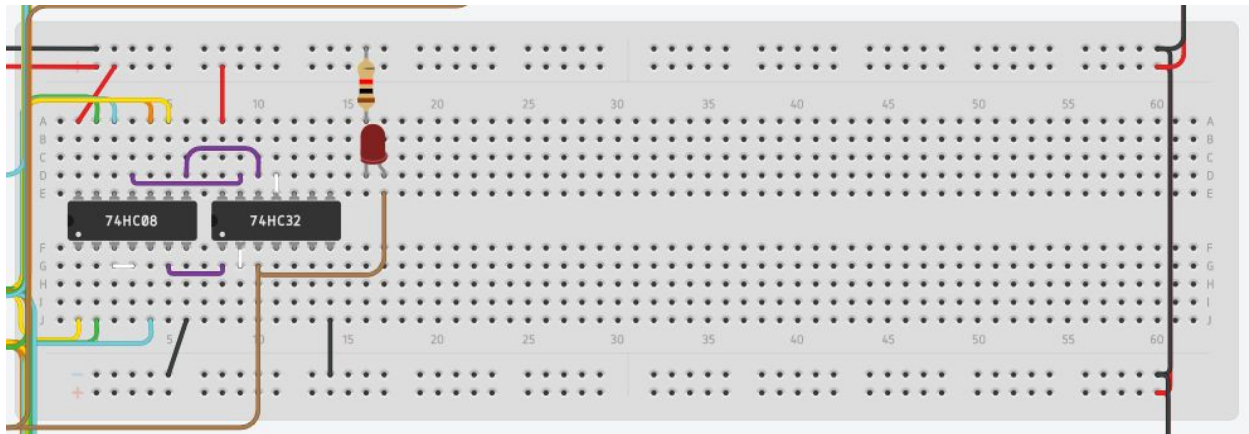


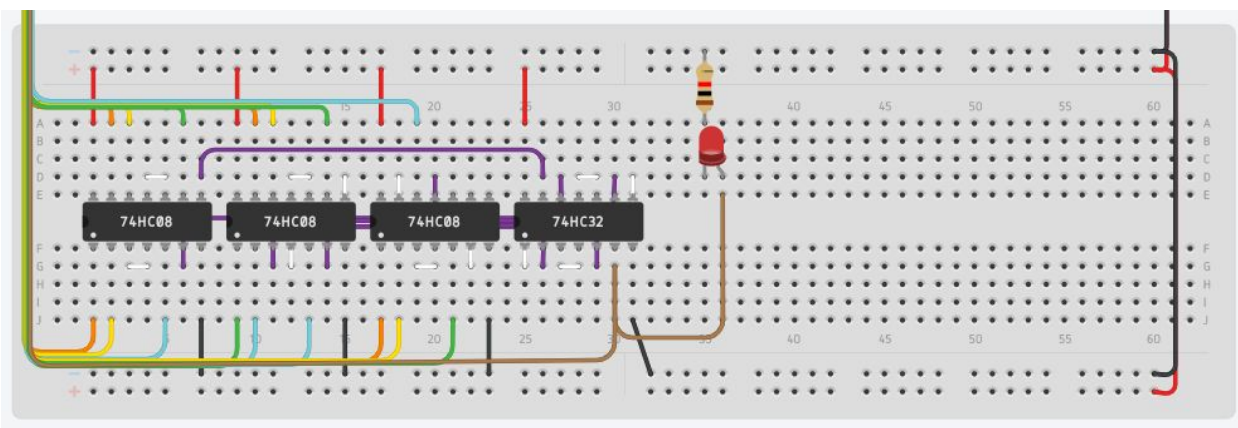
(Figura 6 - Circuito CIs completo no Tinkercad)



(Figura 6.1 - Protoboard conectada à fonte de alimentação, chaves, botão de disparo e porta NOT)

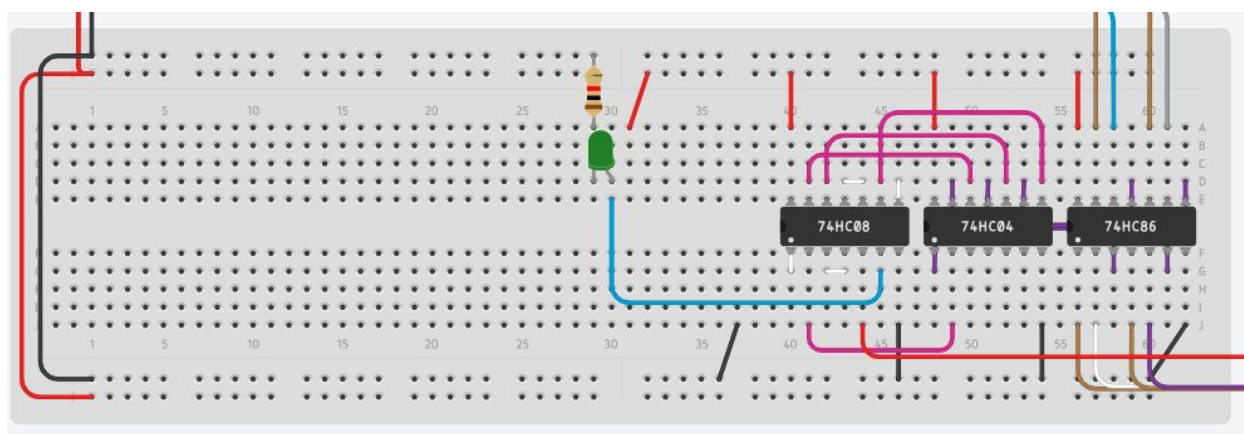
Implementações das saídas S do circuito, representadas pelo fio marrom, como mostra a figura abaixo. Cada protoboard possui um led para testes.





(Figura 6.2 - Implementação das saídas S1, S2, S3 e S4)

Por fim, todas as saídas S são conectadas, juntamente com as entradas do usuário, à portas XNOR, que, se no final o resultado delas forem 1, ou seja, o jogador acertar a codificação do navio, o led verde acenderá e ele ganhará.



(Figura 6.3 - Portas finais do circuito)

3. CONCLUSÃO

O desenvolvimento do projeto foi um tanto proveitoso para desenvolver as habilidades necessárias para criar um circuito funcional do zero. Porém, o uso das ferramentas foi difícil, visto que elas não funcionavam como especificado.

O uso do Tinkercad foi deveras complicado, uma vez que o aplicativo, a partir de certo tamanho de circuito implementado, começava a dar “lag”, as vezes chegando ao ponto de nem executar. Entretanto, quando finalmente finalizado, ele funcionou da maneira que deveria.

Já o uso do Logisim foi essencial para facilitar o desenvolvimento do projeto, pois nele teve-se o circuito resultante da simplificação da codificação, o que possibilitou a realização do restante do projeto.

No Quartus, apesar de ser um aplicativo totalmente manual, foi apenas replicado o circuito resultante do Logisim. Porém, teve um evidente mal funcionamento da placa DE1 no interruptor SW0.

Desse modo, quando tudo foi finalizado, percebe-se que o projeto foi proveitoso, houveram poucas dúvidas no decorrer do seu desenvolvimento e foi possível absorver bastante conteúdo a partir dele.