

## **NORMAS PARA ELABORAÇÃO E ENTREGA DO RELATÓRIO DE ATIVIDADE PRÁTICA**

Olá, estudante. Tudo bem?

As atividades práticas visam desenvolver competências para a atuação profissional. Elas são importantes para que você vivencie situações que te prepararão para o mercado de trabalho.

Por isso, trazemos informações para que você possa realizar as atividades propostas com êxito.

### **1. Que atividade deverá ser feita?**

- A(s) atividades a ser(em) realizada(s) estão descritas no Roteiro de Atividade Prática, disponível no AVA.
- Após a leitura do Roteiro, você deverá realizar a(s) atividade(s) prática(s) solicitadas e elaborar um documento **ÚNICO** contendo todas as resoluções de acordo com a proposta estabelecida.
- O trabalho deve ser autêntico e contemplar todas as resoluções das atividades propostas. Não serão aceitos trabalhos com reprodução de materiais extraídos da internet.

### **2. Como farei a entrega dessa atividade?**

- Você deverá postar seu trabalho final no AVA, na pasta específica relacionada à atividade prática, obedecendo o prazo limite de postagem, conforme disposto no AVA.
- Todas as resoluções das atividades práticas devem ser entregues em um **ARQUIVO ÚNICO** de até 10 MB.
- O trabalho deve ser enviado em formato Word ou PDF, exceto nos casos em que há formato especificado no Roteiro.
- O sistema permite anexar apenas um arquivo. Caso haja mais de uma postagem, será considerada a última versão.

#### **IMPORTANTE:**

- A entrega da atividade, de acordo com a proposta solicitada, é um critério de aprovação na disciplina.
- Não há prorrogação para a postagem da atividade.

Aproveite essa oportunidade para aprofundar ainda mais seus conhecimentos.

Bons estudos!

# Roteiro Aula Prática



**SISTEMAS DIGITAIS E MICROPROCESSADORES**

# ROTEIRO DE AULA PRÁTICA 1

**NOME DA DISCIPLINA:** SISTEMAS DIGITAIS E MICROPROCESSADORES

**Unidade:** CIRCUITOS LÓGICOS COMBINACIONAL E SEQUENCIAL

**Seção:** CIRCUITOS COMBINACIONAIS

## OBJETIVOS

**Definição dos objetivos da aula prática:**

- Introduzir a utilização do simulador Logisim;
- Utilizar entradas e saídas no simulador Logisim;
- Utilizar o circuito codificador de prioridade;
- Utilizar o circuito decodificador;
- Simular o funcionamento de um circuito digital no Logisim.

## INFRAESTRUTURA

**Instalações – Materiais de consumo – Equipamentos:**

### LABORATÓRIO DE INFORMÁTICA

**Equipamentos:**

- Desktop Lab Informatica - Positivo C6300
- ~ 1 computador para cada 2 alunos

## SOLUÇÃO DIGITAL

- LOGISIM (Simulador)

LogiSim: Software para simulação digital de circuitos lógicos.

## EQUIPAMENTO DE PROTEÇÃO INDIVIDUAL (EPI)

Não se aplica

## PROCEDIMENTOS PRÁTICOS

## Procedimento/Atividade nº 1 (Somente virtual)

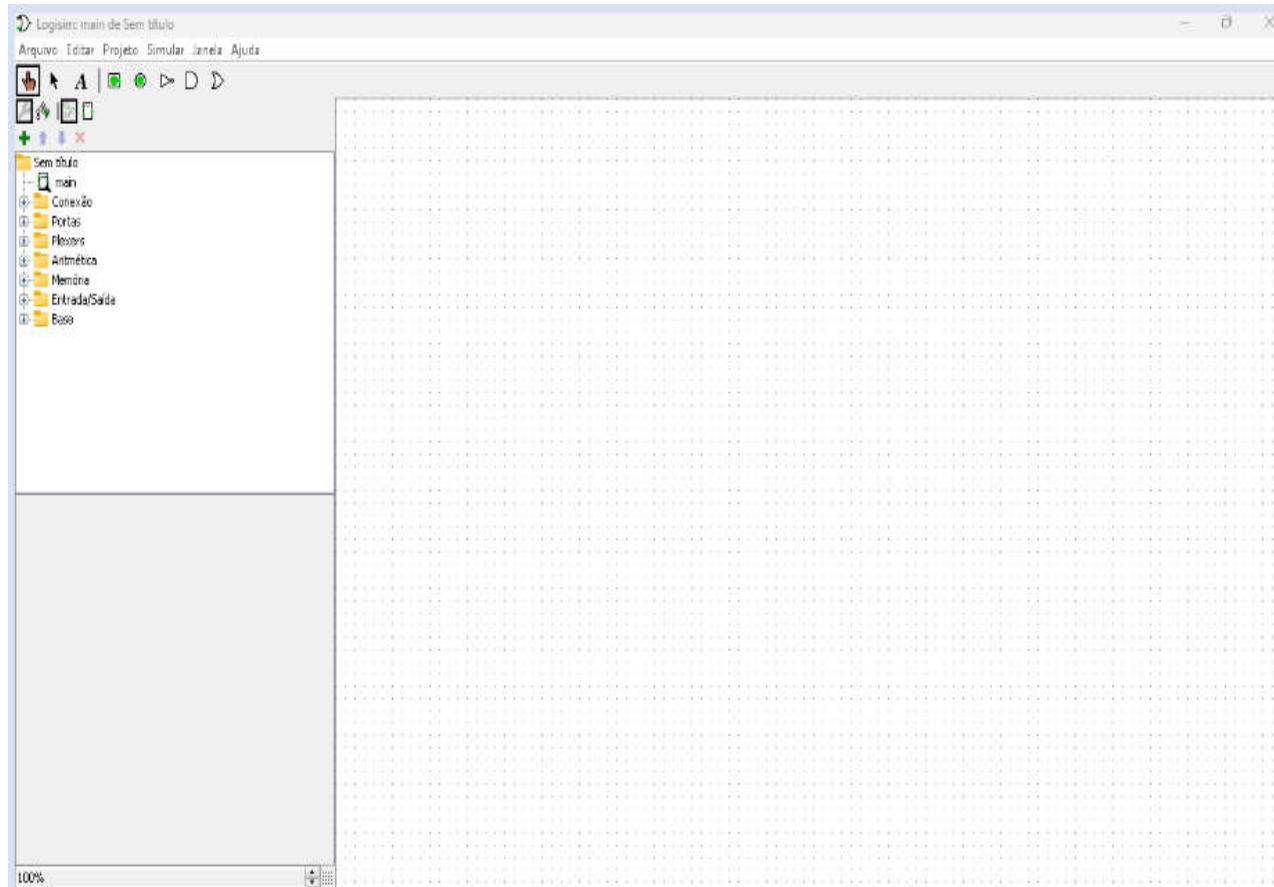
### Atividade proposta:

Implementar um circuito que registre a produção de uma fábrica através de 8 teclas, a cada produção será pressionada a quantidade produzida (1, 2, 3, 4, 5, 6, 7, 8). O circuito deve demonstrar na saída o número de itens produzidos na sequência decimal.

### Procedimentos para a realização da atividade:

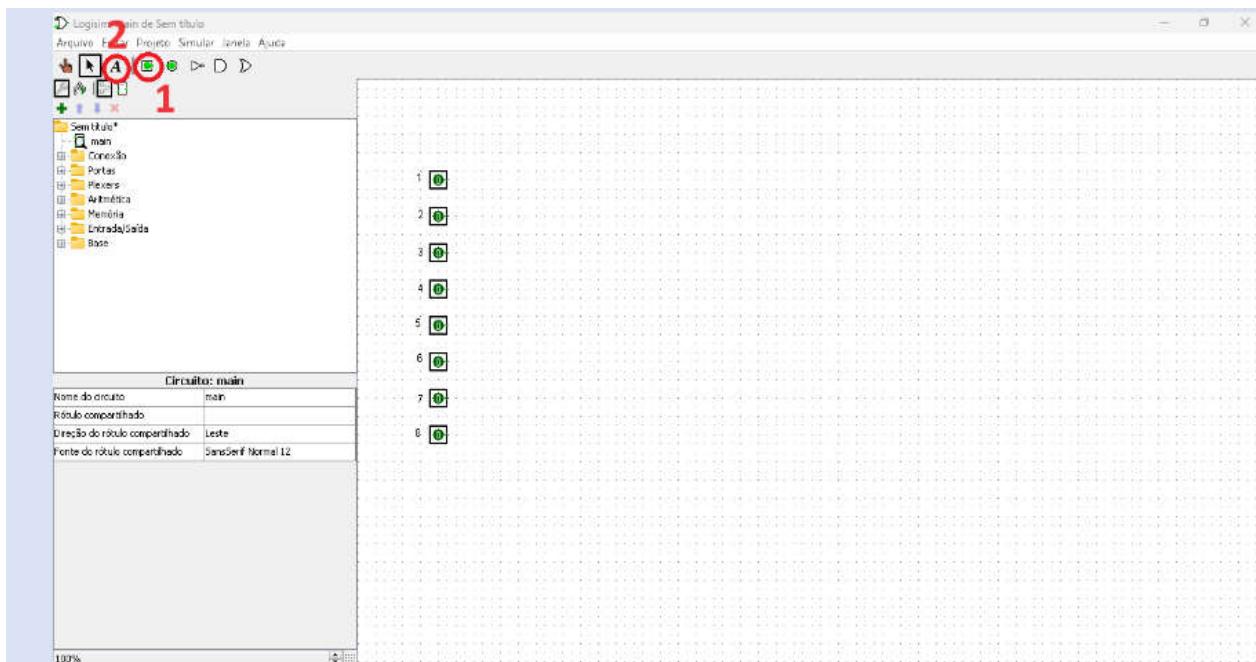
Caro estudante, iniciaremos uma sequência de operações para a montagem e simulação de um circuito digital com codificador e decodificador de acordo com a atividade proposta. É muito importante que você não pule nenhuma etapa demonstrada, para que no final obtenhamos o resultado desejado.

1º Passo: Abra o software simulador Logisim. Observe se a sua tela condiz com a tela mostrada na imagem abaixo:

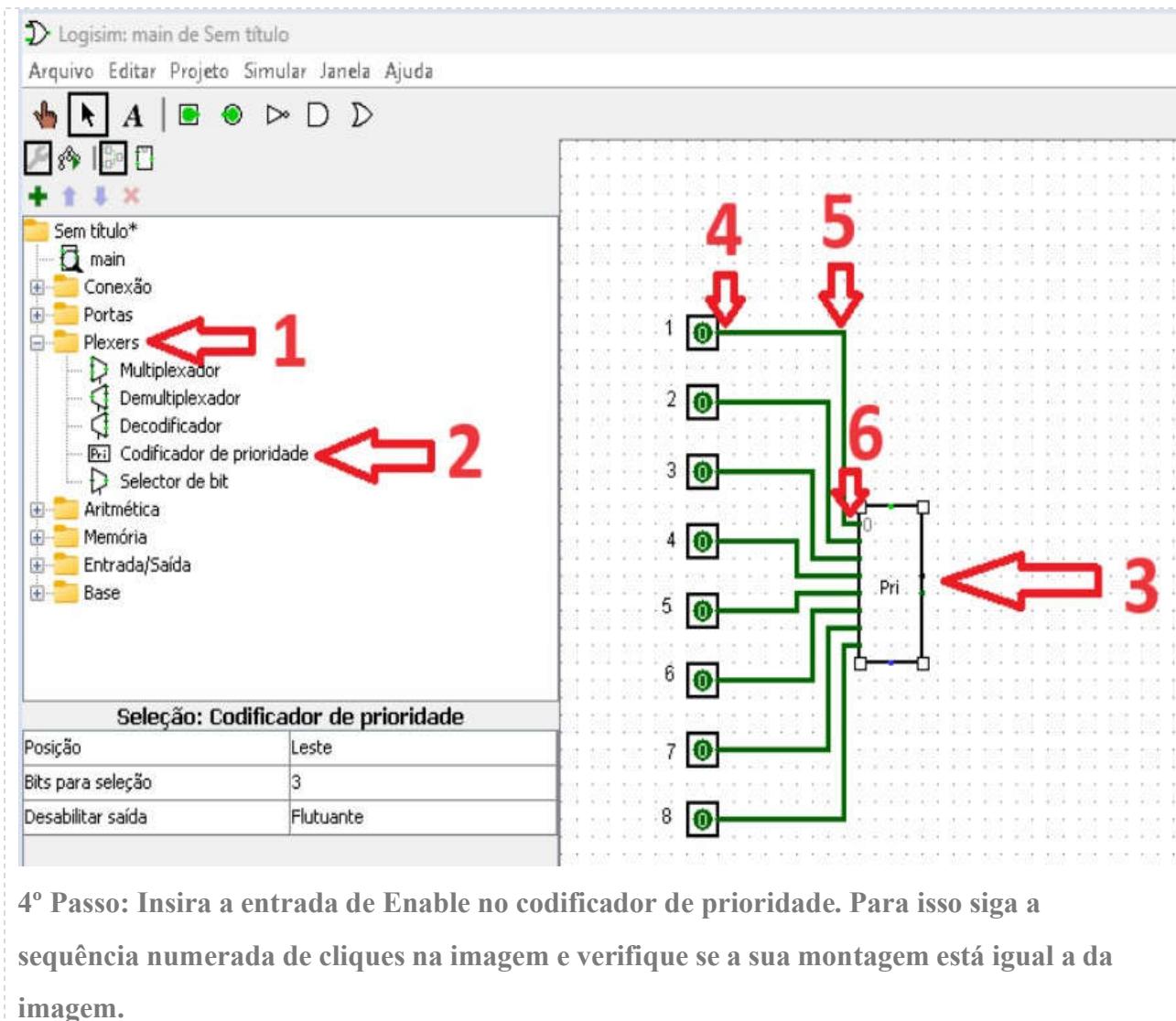


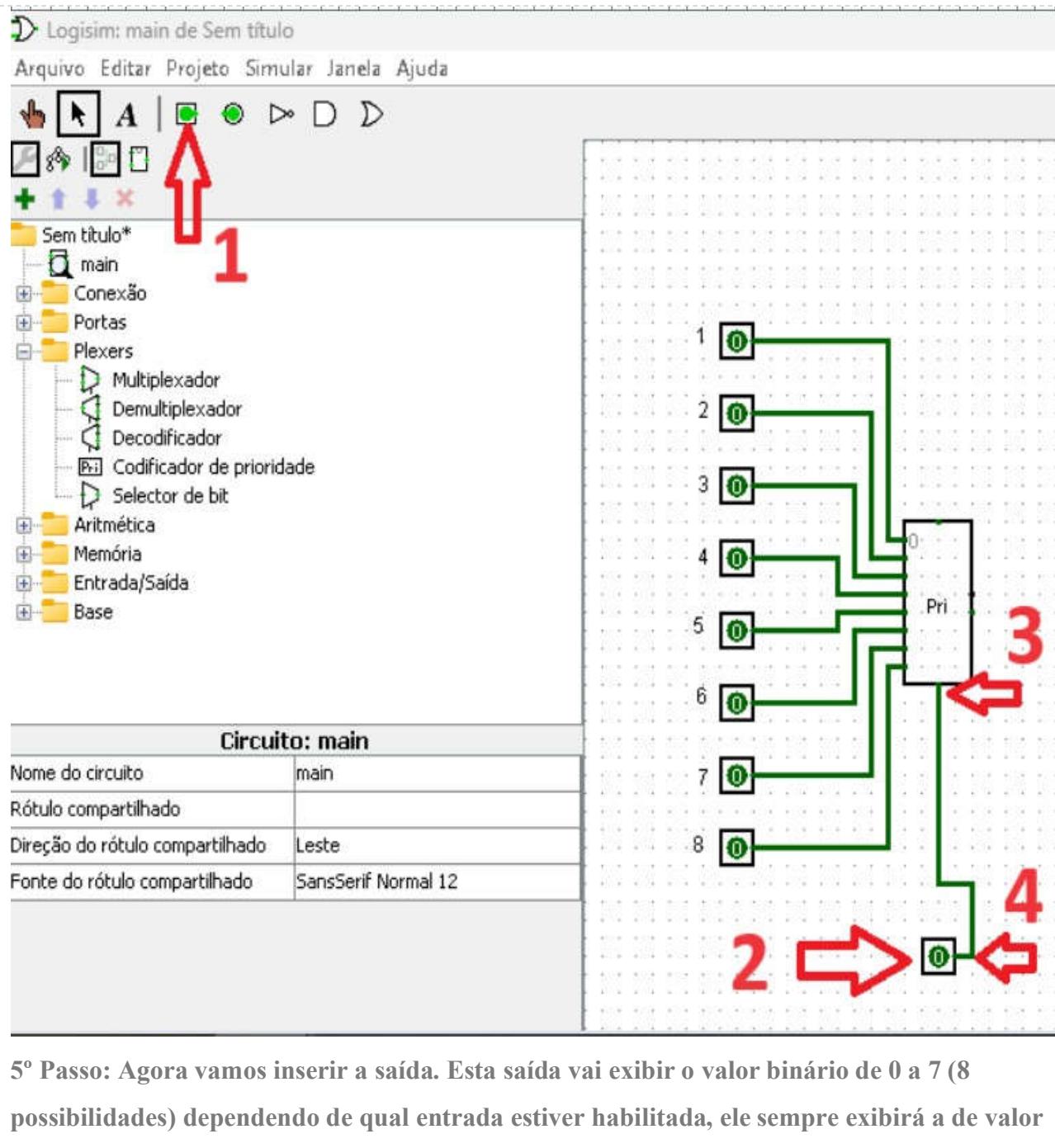
2º Passo: Como a nossa atividade consiste em registrar a produção de uma fábrica através de 8 teclas, vamos inseri-las e colocar os respectivos rótulos em cada uma. Para isso, clique neste ícone localizado no ponto 1 mostrado na imagem abaixo. Coloque 8 teclas alinhadas conforme a imagem.

Logo após coloque os rótulos para saber qual tecla corresponde a qual número, clicando no ícone **A**, indicado como o ponto 2 na imagem. Verifique se a sua montagem está igual à imagem abaixo.



**3º Passo:** Vamos inserir o Codificador de Prioridade. Esse codificador mostra na sua saída a maior entrada selecionada. Para inseri-lo siga a sequência numerada de cliques indicada na figura. Depois, ligue as outras 7 teclas no codificador, conforme a figura.





5º Passo: Agora vamos inserir a saída. Esta saída vai exibir o valor binário de 0 a 7 (8 possibilidades) dependendo de qual entrada estiver habilitada, ele sempre exibirá a de valor mais alto. Para inserir a saída siga a sequência numerada de cliques indicada na imagem.

Logisim: main de Sem título

Arquivo Editar Projeto Simular Janela Ajuda

**1**

Seleção: Pino

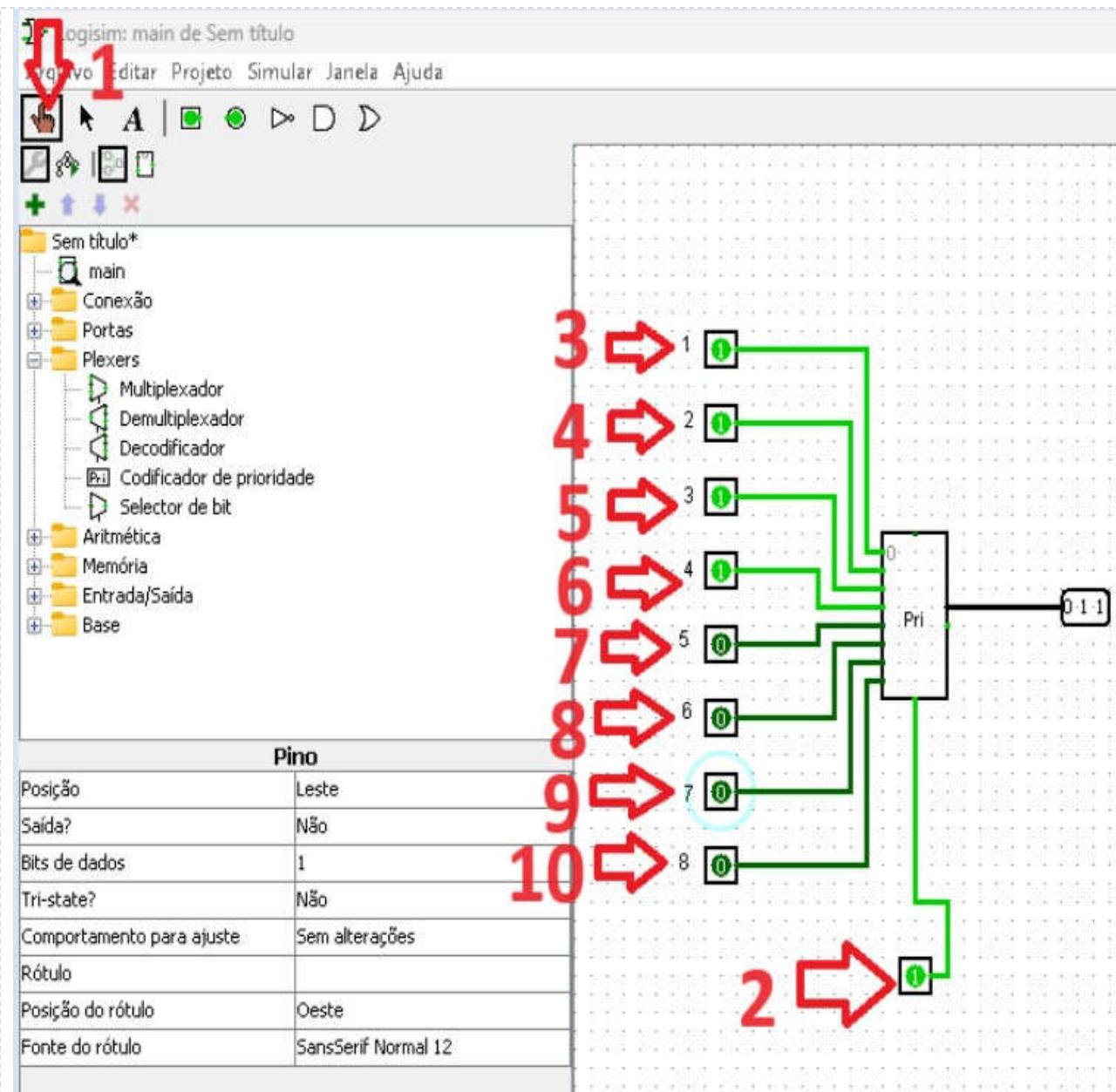
Posição	Oeste
Saída?	Sim
Bits de dados	3
Tri-state?	3
Componente para saída	4

**2**

**4**

**5**

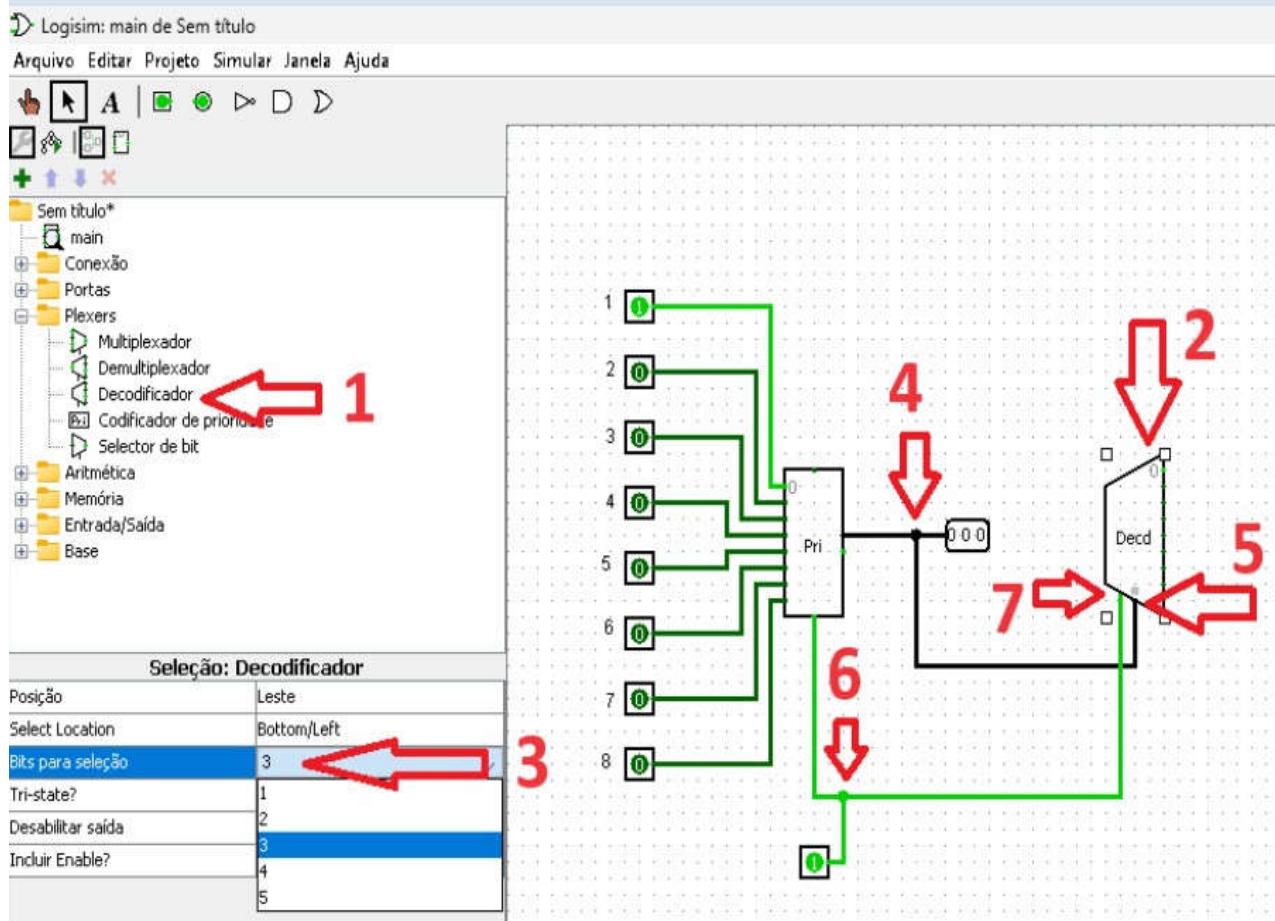
**6º Passo:** Agora vamos simular o funcionamento do codificador de prioridade. Para isso siga a sequência indicada de cliques. Preencha a tabela abaixo com entradas e saídas do codificador de Prioridade:



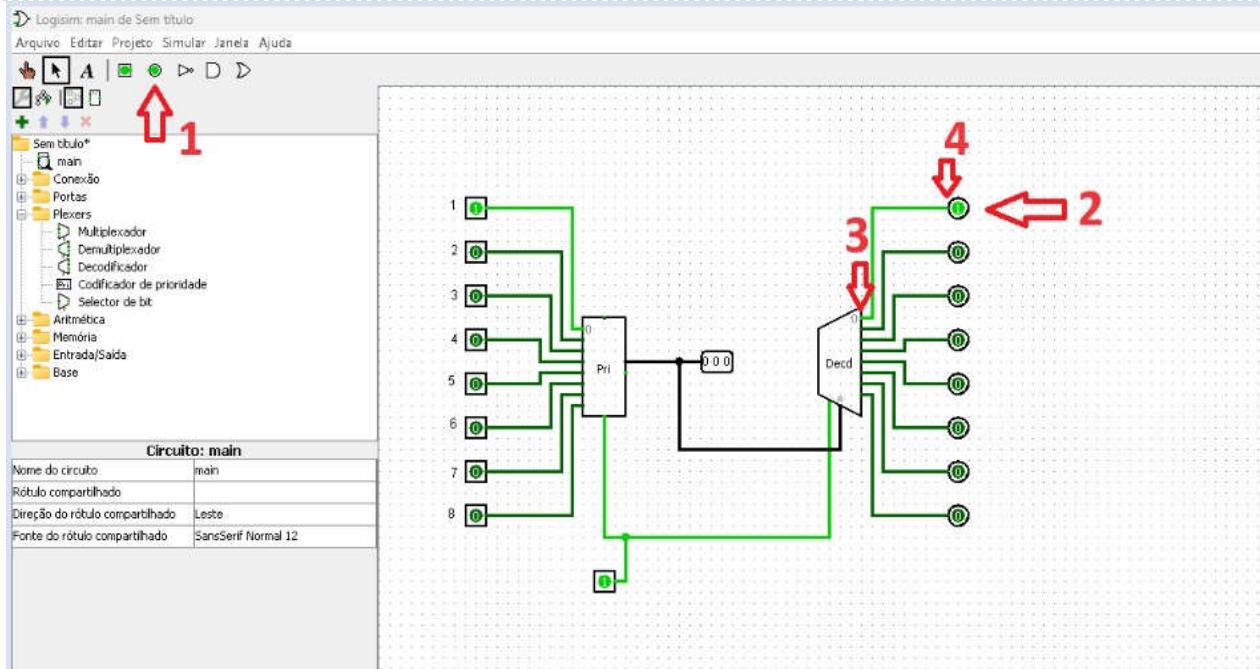
Enable	Tecla Acionada	Saída
0	Tecla de 1 a 8 açãoada	
1	Nenhuma tecla açãoada	
1	1	000
1	2	
1	3	
1	4	
1	5	
1	6	
1	7	

Note que a primeira tecla é ligada na entrada 0 (o codificador tem entradas de 0 a 7), então a saída exibe o valor “0” em binário (000), mesmo que a primeira tecla esteja acionada. E 8ª tecla exibirá a saída 7 em binário na saída (111).

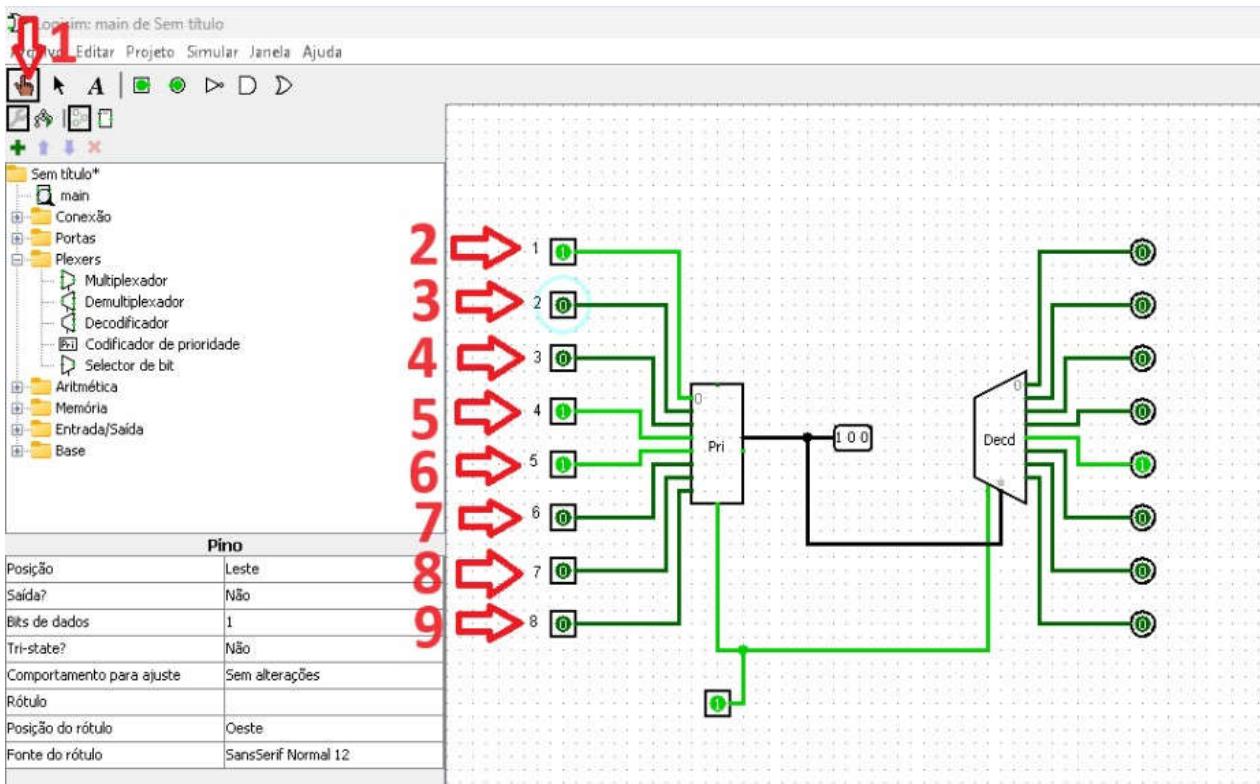
7º Passo: Iniciaremos agora a inserção do decodificador. Para executar este passo siga a sequência de cliques, onde você vai adicionar o decodificador no circuito (clique 1 e 2), no clique 3 você vai definir quantos bits ele receberá na entrada (o mesmos 3 bits de saída do Codificador de Prioridade). Nos cliques de 4 a 7 são feitas as conexões do circuito.



8º Passo: Neste passo vamos inserir as saídas do decodificador. Para inseri-las siga a sequencia de cliques da imagem e repita para as saídas 1 a 7.



**9º Passo:** Neste passo vamos testar todas as possibilidades do circuito digital, que já está pronto! Clique em todas as teclas de entrada e perceba quais saídas acionam e complete a tabela.



Enable	Tecla Acionada	Saída do Decodificador
0	Tecla de 1 a 8 acionada	
1	Nenhuma tecla acionada	
1	1	

1	2	
1	3	
1	4	
1	5	
1	6	
1	7	

**10º Passo:** Faça um resumo explicando suscintamente a lógica de funcionamento do circuito, escolha 3 combinações de chaves diferentes e explique o resultado obtido na saída..

#### **Checklist:**

- Codificador de prioridade funcionando pelas entradas e saídas.
- Decodificador funcionando pelas entradas do codificador e pelas saídas do decodificador.
- Tabelas preenchidas através da simulação
- Prints do circuito montado com 3 combinações diferentes de chaves e sua respectiva explicação
- 10º Passo escrito.

## **RESULTADOS**

### **Resultados de Aprendizagem:**

Utilizar o Logisim para executar o codificador de prioridade funcionando pelas entradas e saídas.

# ROTEIRO DE AULA PRÁTICA 2

**NOME DA DISCIPLINA:** SISTEMAS DIGITAIS E MICROPROCESSADORES

**Unidade:** CIRCUITOS LÓGICOS COMBINACIONAL E SEQUENCIAL

**Seção:** FLIP-FLOPS E CIRCUITOS CORRELATOS

## OBJETIVOS

**Definição dos objetivos da aula prática:**

Projetar, simular e montar circuitos lógicos.

## INFRAESTRUTURA

**Instalações – Materiais de consumo – Equipamentos:**

### LABORATÓRIO DE INFORMÁTICA

**Equipamentos:**

- Desktop Engenharia Positivo Master D3400
- ~ 1 para cada 2 alunos

## SOLUÇÃO DIGITAL

- LOGISIM (Simulador)

LogiSim: Software para simulação digital de circuitos lógicos.

## EQUIPAMENTO DE PROTEÇÃO INDIVIDUAL (EPI)

Não se aplica.

## PROCEDIMENTOS PRÁTICOS

**Procedimento/Atividade nº 1 (Físico e Virtual)**

**Atividade proposta:**

Aprendendo a usar a ferramenta Logisim.

Contador binário de 4 bits e contador de década BCD.

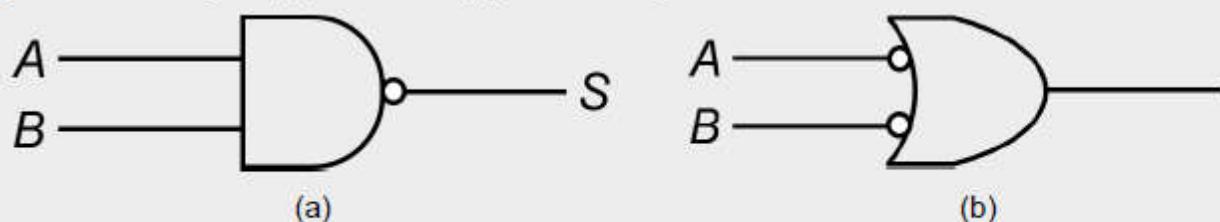
### Procedimentos para a realização da atividade:

Exercício 1:

O Logisim permite projetar e simular circuitos digitais. Foi planejado como ferramenta educacional para ajudar você a aprender como os circuitos funcionam.

Para praticar o uso do Logisim, vamos construir um circuito com uma porta NAND, Figura 1(a), e verificar sua equivalência com a porta OR negativa, Figura 1(b).

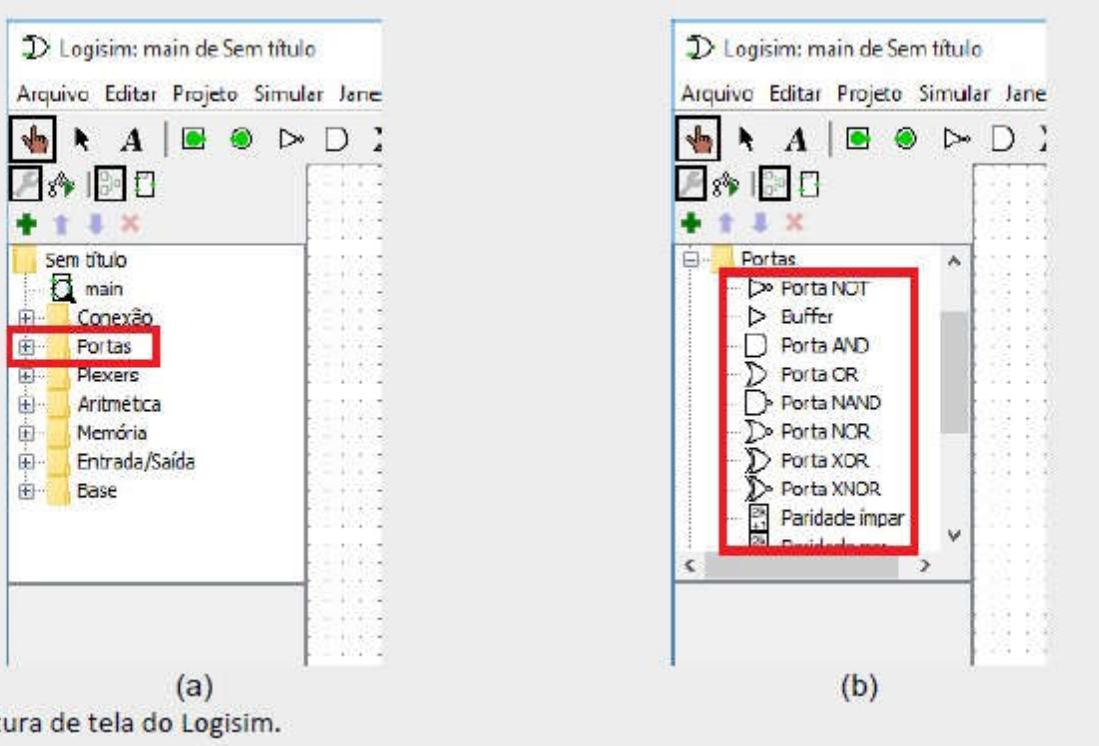
Figura 1 Circuito lógico (a) porta NAND (b) porta OR negativa



Fonte: elaborado pelo autor.

Após baixar e executar o Logisim, o primeiro passo será acrescentar as portas lógicas no ambiente de simulação. Para isso clique na pasta “Portas”, Figura 2(a), assim a lista de portas disponíveis será mostrada, Figura 2(b).

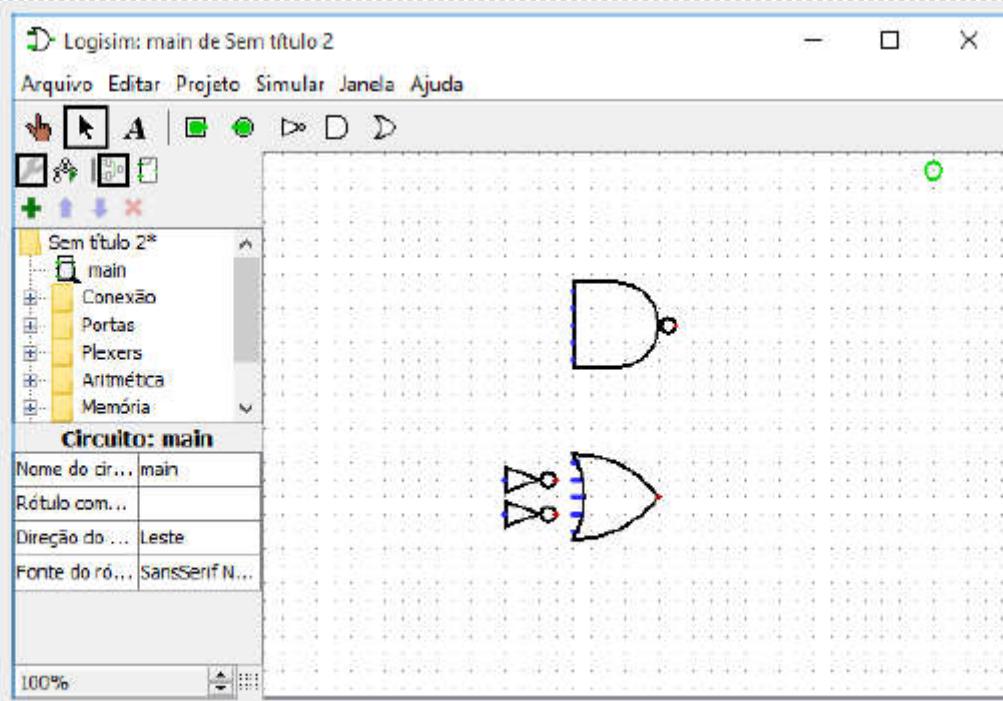
Figura 2 Tela inicial do Logisim.



Clique na porta NAND (, a quinta da lista) e em seguida, clicar na área de edição onde você desejar que a primeira porta NAND deva ficar. Certifique-se de deixar espaço suficiente

para outras coisas do lado esquerdo. Em seguida, clicar na porta OR (, a quarta da lista) e a coloque um pouco mais abaixo, adicione duas portas inversoras em frente de cada uma das entradas da porta OR, Figura 3.

Figura 3 Posicionamento das portas.



Fonte: captura de tela do Logisim.

Observar que haverá cinco pontos do lado esquerdo de cada porta. Esses serão os pontos onde os fios poderão ser conectados.

Acontece que nós iremos usar apenas dois deles para o nosso circuito, mas para outros circuitos, você poderá achar mais útil ter mais de duas conexões em uma porta lógica.

Agora queremos adicionar as duas entradas A e B no diagrama.

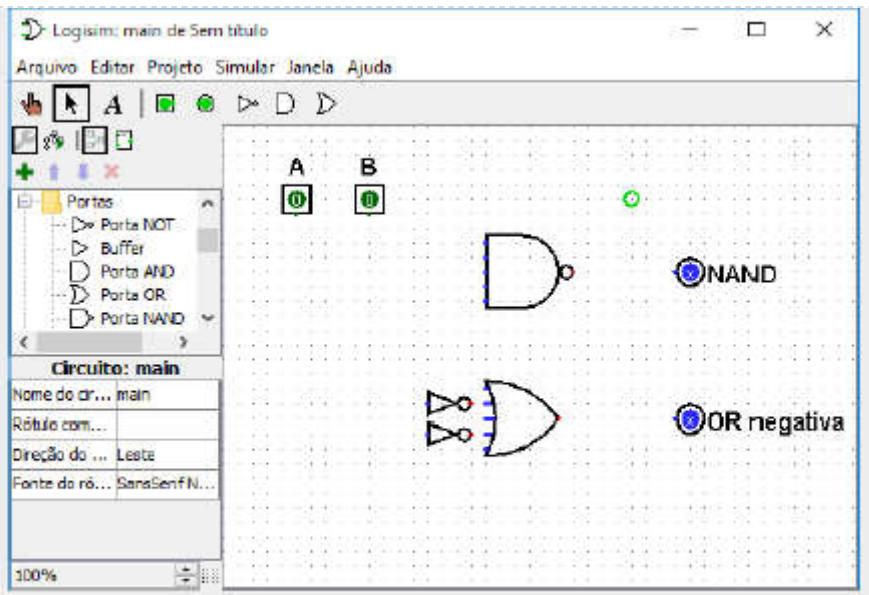


Selecione uma entrada ( ), e coloque os pinos voltados para baixo (utilizando o atributo “posição” da entrada).



Você também deverá colocar uma saída junto a cada uma das portas, usando ( ), Figura 4.

Figura 4 Posicionamento das entradas e saídas



Fonte: captura de tela do Logisim.

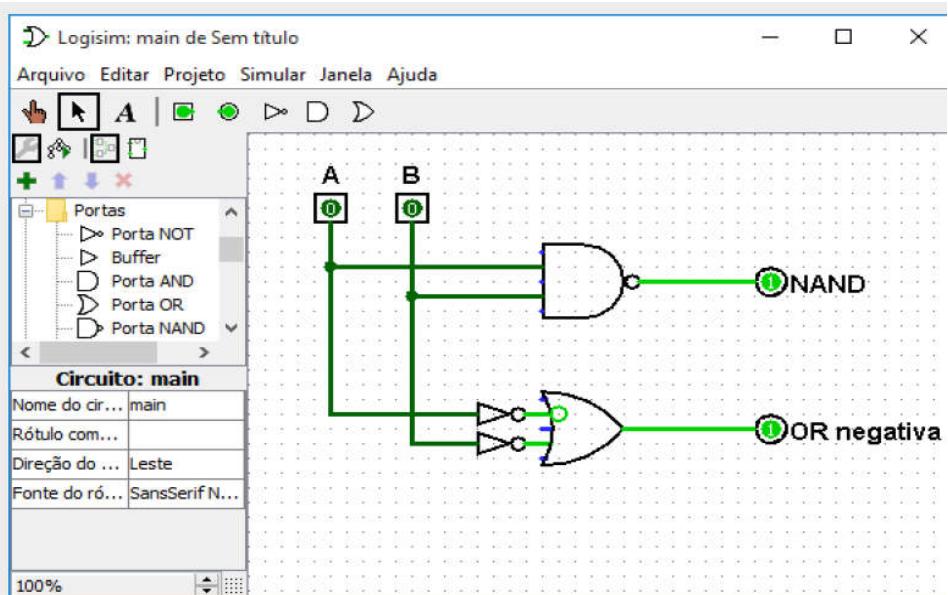
Depois de ter todos os componentes colocados na área de desenho (tela), você estará pronto para começar a adicionar as conexões.

Selecionar a ferramenta Editar ( ). Quando o cursor estiver sobre um ponto que receberá uma extremidade do fio, um pequeno círculo verde será desenhado em torno dele.

Pressione o botão do mouse e arraste até onde você quiser que a outra extremidade do fio vá.

As conexões (fios) em Logisim deverão ser horizontais ou verticais, Figura 5.

Figura 5 Conexão entre as portas



Fonte: captura de tela do Logisim.

O passo final será testar o circuito para garantir que ele realmente realiza o que pretendemos.

O Logisim já está simulando o circuito.

Se olharmos atentamente para o nosso circuito da Figura 5, veremos que para as entradas  $A=0$  e  $B=0$ , tanto a saída da porta NAND quanto a da porta OR negativa estão no nível lógico 1.

Agora tentaremos outra combinação das entradas. Selecione a ferramenta Testar (Poke) () e altere as entradas, clicando sobre elas.

Cada vez que fizer isso sobre a entrada, seu valor será alternado. Quando você mudar o valor de entrada, o Logisim irá mostrar-lhe que os valores seguirão pelos fios marcando-os com a cor verde-claro para indicar um valor 1 ou verde escuro (quase preto) para indicar um valor 0.

Com isso preencha a tabela verdade para as duas portas em questão:

Tabela verdade da porta NAND

A	B	S
0	0	1
0	1	
1	0	
1	1	

Tabela verdade da porta OR negativa

A	B	S
0	0	1
0	1	
1	0	
1	1	

Lembrando que uma maneira de verificar a equivalência entre dois circuitos lógicos é verificar se suas tabelas verdade são iguais.

Agora que já sabemos como usar o Logisim, vamos projetar e simular um circuito meio somador.

Projete e simule um circuito lógico para um meio somador binário com duas entradas (A e B) e duas saídas.

Uma para o bit da soma (S) e outra para o bit “vai um” (C), como indicado na Figura 6.

Figura 6 Circuito meio somador de 2 bits



Fonte: elaborado pelo autor

Para iniciar o seu projeto preencha a tabela verdade para cada uma das saídas, de acordo com

as regras da aritmética binária para a soma.

Tabela verdade para a soma

A	B	S
0	0	
0	1	
1	0	
1	1	

Tabela verdade para o “vai um”

A	B	C
0	0	
0	1	
1	0	
1	1	

Feito isso, use as portas lógicas que você já conhece (AND, OR, NOT, NAND, NOR, XOR e XNOR) e projete o circuito lógico do meio somador.

(Dica: o circuito do meio somador é bastante simples e pode ser executado apenas com duas portas lógicas).

Uma vez que o seu circuito foi projetado simule ele na ferramenta Logisim e verifique se ele executa exatamente as operações esperadas na tabela verdade do meio somador.

Exercício 2:

Os contadores são circuitos lógicos projetados encadeando flip-flops, eles podem ser classificados como assíncronos ou síncronos, de acordo com a forma que eles recebem os pulsos de o clock.

Nos contadores assíncronos, o primeiro flip-flop recebe um sinal de clock externo e cada flip-flop subsequente recebe o clock através da saída do flip-flop anterior.

Nos contadores síncronos, o mesmo sinal de clock é fornecido para todos os flip-flops de forma que eles recebem o clock simultaneamente.

- Projete um contador de 4 bits assíncrono e simule no Logisim.

Os contadores também podem ser projetados de modo a ter um número de estados menor em sua sequência que o valor máximo de . Esse tipo de sequência é chamada de sequência truncada.

Para se obter uma sequência truncada, é preciso forçar que o contador recicle seu estado antes que ele passe por todos os estados possíveis e isso pode ser feito de forma simples utilizando

um contador assíncrono.

- Projete um contador de década BCD que conta de 0 (0000) até 9 (1001).

#### **Checklist:**

Faça o download do Logisim

Preencha a tabela verdade para o meio somador, verifique se ela está realmente correta. Essa etapa é fundamental para o projeto de um circuito lógico.

Revise as tabelas verdade das portas lógicas que você conhece, a resposta para o seu projeto está nestas portas.

Execute a sua simulação e verifique se o circuito se comportou como esperado

Explore o ambiente do Logisim, os flip-flops estão localizados na pasta Memória e o sinal de clock está localizado na pasta conexão.

Simule os pulsos de clock clicando com a “ferramenta testar”

## **RESULTADOS**

### **Resultados de Aprendizagem:**

Para essa aula prática espera-se que o aluno se familiarize com a ferramenta Logisim e seja capaz de projetar e simular circuitos lógicos simples, com poucas portas lógicas. Além disso, se espera que o aluno consiga simular circuitos contadores síncronos e assíncronos.

# ROTEIRO DE AULA PRÁTICA 3

**NOME DA DISCIPLINA:** SISTEMAS DIGITAIS E MICROPROCESSADORES

**Unidade:** PROGRAMAÇÃO DE MICROPROCESSADORES E MICROCONTROLADORES

**Seção:** MICROCONTROLADOR: ARDUINO

## OBJETIVOS

**Definição dos objetivos da aula prática:**

Compreender o funcionamento de um microcontrolador, realizando a programação de algoritmos básicos em um Arduino

## INFRAESTRUTURA

**Instalações – Materiais de consumo – Equipamentos:**

**SISTEMAS ELÉTRICOS, ELETRÔNICOS E AUTOMAÇÃO**

**Materiais de consumo:**

- ARDUINO UNO R3 + CABO USB  
~ 1 por grupo
- KIT JUMPER MACHO/MACHO PARA PROTOBOARD  
~ 1 por grupo
- LED VERMELHO 5 MM  
~ 5 por grupo
- POTENCIOMETRO 0 - 100 OHM  
~ 1 por grupo
- PROTO BOARD 2420 PONTOS  
~ 1 por grupo
- RESISTOR 220 OHM  
~ 5 por grupo

## SOLUÇÃO DIGITAL

- ARDUINO SOFTWARE (IDE) (Software)

Arduino Software (IDE): ambiente de desenvolvimento integrado (IDE) para a plataforma Arduino, permitindo a criação de programas para controlar dispositivos eletrônicos.

- TINKERCAD (Software)

Programa de modelagem tridimensional online gratuito que roda em um navegador da web, conhecido por sua simplicidade e facilidade de uso.

## EQUIPAMENTO DE PROTEÇÃO INDIVIDUAL (EPI)

Não se aplica.

## PROCEDIMENTOS PRÁTICOS

### Procedimento/Atividade nº 1 (Somente Físico)

#### Atividade proposta:

Conhecendo o Arduino

Sistemas de controle simples

#### Procedimentos para a realização da atividade:

1.1 Crie um programa para Arduino Uno que repete qualquer caractere recebido pela porta serial, semelhante a um eco. Envie seu programa para o Arduino e teste seu funcionamento.

1.2 Crie um programa para Arduino Uno que faz com que 5 LEDs pisquem sequencialmente de forma cíclica, simulando um registrador de deslocamento. Nesse programa cada um dos LEDs deverá ficar aceso por 200 ms, um de cada vez. Portanto, o LED 1 acende, passado o tempo, ele se apaga e o LED 2 é aceso e assim sucessivamente. Envie seu programa para o Arduino e teste seu funcionamento, ligando 5 LEDs nos pinos de 8 a 12 do Arduino. Não se esqueça de colocar um resistor de 220 O em série com cada LED para limitar a corrente.

2.1 Crie um programa para Arduino Uno para o controle da luminosidade de um LED ligado no pino 10 por meio de um potenciômetro com seu pino central ligado no pino A0 e suas extremidades ligadas em 5 V e GND. Conforme mudança da posição do potenciômetro a luminosidade do LED deve aumentar ou diminuir. Por exemplo, se ele estiver na metade de sua excursão, o LED deverá estar com 50 % de sua luminosidade. Envie seu programa para o

Arduino e teste seu funcionamento.

Não se esqueça de colocar um resistor de 220 O em série com o LED para limitar a corrente.

2.2 Crie um programa para Arduino Uno para o controle da luminosidade de um LED ligado no pino 10 por meio da comunicação serial. O comando a ser enviado deverá ser em ASCII, sempre com 3 caracteres entre 000 e 255, indicando do brilho do LED mínimo e máximo, respectivamente.

Envie seu programa para o Arduino e teste seu funcionamento.

Não se esqueça de colocar um resistor de 220 O em série com o LED para limitar a corrente.

#### **Checklist:**

Realize a elaboração dos códigos conforme orientado.

Monte os circuitos necessários para que seja possível testar os algoritmos elaborados.

Envie o programa para o arduino e realize diversos testes de funcionamento.

Realize a elaboração dos códigos conforme orientado.

Monte os circuitos necessários para que seja possível testar os algoritmos elaborados.

Envie o programa para o arduino e realize diversos testes de funcionamento.

#### **Procedimento/Atividade nº 2 (Somente Virtual)**

#### **Atividade proposta:**

A proposta da atividade é desenvolver o conhecimento para aplicar o Arduino a um projeto real utilizando a linguagem de programação C, desenvolvendo um projeto completo Monitor de Umidade do Solo aplica os conhecimentos adquiridos para utilizar o Arduino e suas saídas Digitais e analógicas abrindo um mundo de possibilidades.

#### **Procedimentos para a realização da atividade:**

Para realizar o desenvolvimento Monitor de Umidade do Solo com o Arduino vamos precisar separar os componentes necessários para a construção do circuito, separando seus componentes e realizando suas conexões e por fim vamos construir o algoritmo de programação para realizar a operação. Todo esse processo deve ser realizado no Tinkercad, criando um novo projeto do tipo “Circuito”

##### **1. Componentes para o Projeto:**

Para construir este projeto precisamos dos seguintes componentes:

- 1 ARDUINO;
- 1 PROTOBOARD ;
- 5 LEDS, sendo:

1 VERMELHO, 1 AMARELO, 1 LARANJA, 1 AZUL e 1 VERDE;

- 5 resistores de 220 Ohms;
- MÓDULO SENSOR DE UMIDADE DO SOLO;

Separe-os para realizar as conexões.

2. Conexões a serem realizadas:

- a. Conecte as saídas 9,10,11, 12 e 13 ao ânodo do LED;
- b. Crie a conexão do catodo dos LEDS para os Resistor de 220 Ohms;
- c. Conecte o outro terminal dos Resistores ao GND (Negativo) do Arduino;
- d. Crie as conexões de forma organizada e limpa;
- e. Conecte o Pino VCC do Sensor de Umidade de Solo na porta A0;
- f. Conecte o Pino GND do Sensor de Umidade de Solo no GND do Arduino;
- g. Conecte o Pino SIG do Sensor de Umidade na Porta A1;

3. Algoritmo na linguagem C:

Ao iniciar o desenvolvimento de código texto você terá as duas funções padrões do Arduino a Setup() e a Loop()

a. Defina as variáveis antes do bloco void setup():

- Crie uma variável para armazenar o valor do Sinal de Umidade chamado umidade com valor 0;
- Crie a variável led\_vermelho com a porta 13;
- Crie a variável led\_amarelo com a porta 12;
- Crie a variável led\_laranja com a porta 11;
- Crie a variável led\_azul com a porta 10;
- Crie a variável led\_verde com a porta 9;
- Crie uma variável pin\_vcc com a porta A0;
- Crie uma variável pin\_sensor com a porta A1

As variáveis com nome inicial led\_ realizará o controle para acender cada LED à medida que o valor da variável umidade for alterado, já o pin\_vcc e pin\_sensor serão os de controle do sensor de umidade.

b. No bloco void setup(), crie as definições para as portas, utilize-as da seguinte forma:

- Defina as variáveis com inicio led\_ como saída;
- Defina o pin\_vcc como saída;
- Defina o pin\_sensor como entrada;
- Defina Serial.begin(9600); para que possamos verificar no meio do

processo os valores gerados pelo sensor de umidade.

- c. E por fim, construa a execução das ações do seu código dentro do bloco void loop() utilizando as funções digitais e analógica do arduino;
- Inicie o código aplicando tensão no pin\_vcc;
  - Aplique a variável umidade criada os valores do pin\_sensor com a função AnalogRead;
  - Crie uma linha com o código Serial.println(umidade); para verificar os valores da variável umidade no monitor serial;
  - Defina que todos os leds estarão desligados ainda neste momento
  - Crie os blocos de verificação para acender cada led em um determinado parâmetro, utilize as seguintes regras:
    - Se menor que 200 acenderá o led\_vermelho
    - Se menor que 400 acenderá o led\_amarelo
    - Se menor que 600 acenderá o led\_laranja
    - Se menor que 800 acenderá o led\_azul
    - Se for diferente destes valores acenda o led\_verde

Após seguir todas as etapas, execute a simulação para ver a operação.

O Projeto deve ligar o led\_vermelho quando a variável umidade estiver menor que 200 e a medida que é aplicado os valores controlados pela barra de controle no simulador do sensor de umidade, irá acender os leds na sequência de amarelo, laranja, azul e verde.

O circuito será um loop infinito e manterá a luz acesa conforme os valores recebidos na variável umidade.

#### **Checklist:**

- Realizar as separações dos componentes necessários para o projeto;
- Realizar as conexões do circuito eletrônico;
- Realizar a elaboração do código de programação C;
- Executar o projeto;
- Verificar se o projeto segue o escopo planejando.

## RESULTADOS

### **Resultados de Aprendizagem:**

Compreender como é o processo de desenvolvimento de firmware para microcontroladores, utilizando a plataforma Arduino para o projeto de sistemas de controle variados, que podem ser reproduzidos facilmente, utilizando todas as funcionalidades existentes em um microcontrolador.

# ROTEIRO DE AULA PRÁTICA 4

**NOME DA DISCIPLINA:** SISTEMAS DIGITAIS E MICROPROCESSADORES

**Unidade:** PROGRAMAÇÃO DE MICROPROCESSADORES E MICROCONTROLADORES

**Seção:** NOVAS ARQUITETURAS E TENDÊNCIAS

## OBJETIVOS

**Definição dos objetivos da aula prática:**

Compreender o funcionamento de um microcontrolador, realizando a programação de algoritmos em um Arduino

## INFRAESTRUTURA

**Instalações – Materiais de consumo – Equipamentos:**

**SISTEMAS ELÉTRICOS, ELETRÔNICOS E AUTOMAÇÃO**

**Materiais de consumo:**

- ARDUINO UNO R3 + CABO USB  
~ 1 por grupo
- KIT JUMPER MACHO/MACHO PARA PROTOBOARD  
~ 1 por grupo
- MODULO DS1302 RELOGIO TEMPO REAL ARDUINO  
~ 1 por grupo
- PROTO BOARD 2420 PONTOS  
~ 1 por grupo
- RESISTOR 1000000 OHM  
~ 1 por grupo
- SENSOR DHT11 DE TEMPERATURA E UMIDADE PARA ARDUINO  
~ 1 por grupo

## SOLUÇÃO DIGITAL

- ARDUINO SOFTWARE (IDE) (Software)

Arduino Software (IDE): ambiente de desenvolvimento integrado (IDE) para a plataforma Arduino, permitindo a criação de programas para controlar dispositivos eletrônicos.

## EQUIPAMENTO DE PROTEÇÃO INDIVIDUAL (EPI)

Não se aplica.

## PROCEDIMENTOS PRÁTICOS

### Procedimento/Atividade nº 1 (Somente Físico)

#### Atividade proposta:

Relógio em tempo real e sensor de temperatura e umidade  
Estação meteorológica

#### Procedimentos para a realização da atividade:

1- Crie um programa para Arduino Uno que realiza a leitura de data e hora do módulo relógio em tempo real DS1302 e a envia na serial juntamente com a temperatura e umidade medidas pelo sensor DHT11 a cada 1 minuto. Saiba que o sensor demora alguns segundos para realizar a medida, então leia a data e hora do módulo para fazer a amostragem com precisão 1 min.

Não utilize a função delay. Envie seu programa para o Arduino e teste seu funcionamento. Não se esqueça de colocar um resistor de pull-up no pino de dados do DHT11.

2- Crie um programa para Arduino Uno que simula uma estação meteorológica, utilizando o módulo relógio em tempo real DS1302 e o sensor de temperatura e umidade DHT11. A cada 1 min deve ser feita uma medida de umidade e temperatura, que deve ser enviada via porta serial juntamente com data e hora da medida. Saiba que o sensor demora alguns segundos para realizar a medida, então leia a data e hora do módulo para fazer a amostragem com precisão 1 m. Não utilize a função delay.

Salve os dados de data e hora na memória EEPROM do Arduino juntamente com o valor de temperatura e umidade. Para a data utilize 1 byte para o dia, um para o mês, um para o ano (considere apenas os dígitos decimais), um para hora e um para os minutos, totalizando 5 bytes.

Utilize mais dois bytes, um para a temperatura em °C e um para a umidade em %, desconsiderando as casas decimais.

Faça uma lógica para evitar o estouro da EEPROM e faça com que ao receber o caractere "L" via serial, o microcontrolador retorne todos os dados armazenados. No momento que os dados forem lidos, eles podem ser apagados ou escritos novos dados sobre eles.

Envie seu programa para o Arduino e teste seu funcionamento.

Obs.: A memória EEPROM do Arduino Uno possui 1024 posições (0 a 1023) de 1 byte cada.

#### **Checklist:**

Realize a elaboração dos códigos conforme orientado.

Monte os circuitos necessários para que seja possível testar os algoritmos elaborados.

Envie o programa para o arduino e realize diversos testes de funcionamento.

Realize a elaboração dos códigos conforme orientado.

Monte os circuitos necessários para que seja possível testar os algoritmos elaborados.

Envie o programa para o arduino e realize diversos testes de funcionamento.

## **RESULTADOS**

### **Resultados de Aprendizagem:**

Compreender como é o processo de desenvolvimento de firmware para microcontroladores, utilizando a plataforma Arduino para o projeto de sistemas de controle variados, que podem ser reproduzidos facilmente, utilizando todas as funcionalidades existentes em um microcontrolador.

# ROTEIRO DE AULA PRÁTICA 5

**NOME DA DISCIPLINA:** SISTEMAS DIGITAIS E MICROPROCESSADORES

**Unidade:** PROGRAMAÇÃO DE MICROPROCESSADORES E MICROCONTROLADORES

**Seção:** APLICAÇÕES PARA PROCESSADORES ARM

## OBJETIVOS

### Definição dos objetivos da aula prática:

Compreender o funcionamento de um microcontrolador, realizando a programação de algoritmos em um Arduino

## INFRAESTRUTURA

### Instalações – Materiais de consumo – Equipamentos:

#### SISTEMAS ELÉTRICOS, ELETRÔNICOS E AUTOMAÇÃO

##### Materiais de consumo:

- ARDUINO UNO R3 + CABO USB  
~ 1 por grupo
- KIT JUMPER MACHO/MACHO PARA PROTOBOARD  
~ 1 por grupo
- PROTO BOARD 2420 PONTOS  
~ 1 por grupo
- RESISTOR 220 OHM  
~ 3 por grupo
- LED RGB 5 MM  
~ 1 por grupo
- SENSOR HC-SR04 MEDIDOR DE DISTÂNCIA POR ULTRASOM PARA ARDUINO  
~ 1 por grupo

## SOLUÇÃO DIGITAL

- ARDUINO SOFTWARE (IDE) (Software)

Arduino Software (IDE): ambiente de desenvolvimento integrado (IDE) para a plataforma Arduino, permitindo a criação de programas para controlar dispositivos eletrônicos.

- TINKERCAD (Software)

Programa de modelagem tridimensional online gratuito que roda em um navegador da web, conhecido por sua simplicidade e facilidade de uso.

## EQUIPAMENTO DE PROTEÇÃO INDIVIDUAL (EPI)

Não se aplica

## PROCEDIMENTOS PRÁTICOS

### Procedimento/Atividade nº 1 (Somente Físico)

#### Atividade proposta:

Sensor ultrassom com LED RGB.

Sensor ultrassom com timer.

#### Procedimentos para a realização da atividade:

1- Crie um programa para Arduino Uno que indica por meio de um LED RGB a distância medida pelo sensor ultrassônico. Para esse experimento, quando um objeto estiver a mais de 50 cm a cor do LED deve ser totalmente verde.

Com a redução da distância a cor do LED deve ir ficando amarela até a distância de 25 cm, quando se torna totalmente amarela. A partir desse ponto, o LED deve ir se tornando vermelho com a aproximação do objeto e se tornar totalmente vermelho quando a distância for inferior a 5 cm.

Envie seu programa para o Arduino e teste seu funcionamento.

Para ligar o LED RGB utilize os pinos 9, 10 e 11 do Arduino, que possuem saídas PWM.

Não se esqueça de colocar um resistor em série com cada cor do LED para limitar a corrente que passa por ele.

2- Crie um programa para Arduino Uno que realiza a leitura da distância medida pelo sensor ultrassônico e a envie via serial a cada 1 s.

A medida deve estar em centímetros. Programe também o LED ligado ao pino 13 do Arduino para piscar com velocidade proporcional a distância.

Assim, quando a distância medida pelo sensor for de 50 cm ou mais, o LED deverá mudar seu estado a cada 1 s.

Com a redução da distância esse período deve mudar proporcionalmente.

Perceba que você deve ter duas bases de tempo nesse programa, uma para amostragem e envio da distância via serial e outra para o LED, para isso utilize uma interrupção do timer para te ajudar.

Envie seu programa para o Arduino e teste seu funcionamento.

#### **Checklist:**

Realize a elaboração dos códigos conforme orientado.

Monte os circuitos necessários para que seja possível testar os algoritmos elaborados.

Envie o programa para o arduino e realize diversos testes de funcionamento.

### **Procedimento/Atividade nº 1 (Somente Virtual)**

#### **Atividade proposta:**

Sensor ultrassom com LED RGB.

#### **Procedimentos para a realização da atividade:**

Para realizar essa simulação, será necessário criar um novo projeto do tipo “Circuito” no Tinkercad. Feito isso, adicione os seguintes elementos na montagem:

1 Arduino Uno R3

1 Placa de ensaio pequena

1 LED RGB

3 Resistores de 220 Ohms

1 Sensor de distância ultrassônico

Adicionados o elementos, posicione os resistores, LED e sensor na protoboard corretamente, realizando as ligações necessárias no Arduino. Para ligar o LED RGB utilize os pinos 9, 10 e 11 do Arduino, que possuem saídas PWM. O sinal do sensor de distância pode ser ligado no pino de sua preferência, evitando os pinos 0, 1 e 13, além dos já utilizados para o LED.

O procedimento a ser realizado é o seguinte:

Crie um programa para Arduino Uno que indica por meio de um LED RGB a distância medida pelo sensor ultrassônico. Para esse experimento, quando um objeto estiver a mais de 50 cm a

cor do LED deve ser totalmente verde.

Com a redução da distância a cor do LED deve ir ficando amarela até a distância de 25 cm, quando se torna totalmente amarela. A partir desse ponto, o LED deve ir se tornando vermelho com a aproximação do objeto e se tornar totalmente vermelho quando a distância for inferior a 5 cm.

Elaborado o código, realize a simulação no Tinkercad, clicando sobre o sensor ultrassônico para ajustar a distância que ele estará medindo.

Não se esqueça de colocar um resistor de 220 Ohms em série com cada cor do LED para limitar a corrente que passa por ele.

Para que o circuito funcione no Tinkercad, faça uma pesquisa sobre como fazer que o sensor de distância ultrassônico funcione. Com isso, será possível elaborar a lógica necessária para que o sistema funcione como o desejado.

### **Checklist:**

Realize a elaboração dos códigos conforme orientado.

Monte os circuitos necessários no simulador para que seja possível testar os algoritmos elaborados.

Realize a simulação a fim de verificar o funcionamento da montagem.

## **RESULTADOS**

### **Resultados de Aprendizagem:**

Compreender como é o processo de desenvolvimento de firmware para microcontroladores, utilizando a plataforma Arduino para o projeto de sistemas de controle variados, que podem ser reproduzidos facilmente, utilizando todas as funcionalidades existentes em um microcontrolador.