

# Irrigação inteligente

Luis Henrique Matos Sales e João Lucas Sidney Rodrigues

<sup>1</sup>Departamento de Ciência da Computação da Universidade Federal de Roraima (UFRR) – Boa Vista – RR – Brasil

luismatossales@hotmail.com, JoaoLucaSidney@outlook.com

**Resumo:** Neste trabalho foi implementado um sistema de irrigação simulado na plataforma TinkerCad, utilizando um arduino uno como microcontrolador e sensores de luz, temperatura e gás. O usuário pode escolher entre uma das três culturas de plantas pré-definidas no sistema que alteraram tanto a frequência de irrigação tanto o tempo de irrigação, a irrigação não é acionada a não ser que a temperatura esteja acima de 25°C, para casos em que o valor do gás ultrapasse 50 ou a temperatura passe de 60°C o sistema entenderá como incêndio e ligara a irrigação para tentar apagar o fogo e caso o nível de iluminação captado pelo fotoregistor esteja abaixo de 250 será ligado uma luz externa para iluminar a plantação.

## 1. Introdução

Este trabalho possui foco em resolver o problema de irrigação para diferentes culturas de plantações agrícolas, sendo campas de realizar a irrigação na medida e em tempo pré-determinado para cada cultura e detectar possíveis incêndios e tentar extinguí-los. O ambiente que proporcionou a construção deste trabalho foi o TinkerCad com a url <https://www.tinkercad.com>. Esta plataforma permite a criação de simulações de circuitos com arduino e a criação de projetos de construção de objetos 3D como uma casa.

## 2. Descrição do Projeto

Os componentes que foram utilizados na construção deste projeto foram um arduino UNO como unidade controladora, um micro servo motor para realizar o movimento de abrir e das torneiras, um sensor de temperatura TMP36, um sensor de gás para medir o nível de CO<sub>2</sub> no ambiente, um fotoregistor para detectar se a luz externa será ligado ou não, um potenciômetro para controlar a intensidade da tela LCD, uma tela LCD para exibir os dados que estão sendo coletados e os eventos que estão ocorrendo como irrigação ou incêndio, dois botões um para mudar o tipo cultura e outra para selecionar, três capacitores de 1Kohm, dois capacitores 220ohms e um capacitor de 4,7Kohms. O ideal caso este projeto fosse implementado na vida real seria criar-se uma API em nuvem para que o usuário possa controlar e ter acesso a essas informações de qualquer lugar além do uso de um sensor de umidade que encontra-se indisponível na plataforma TinkerCad até o presente momento da construção deste trabalho.

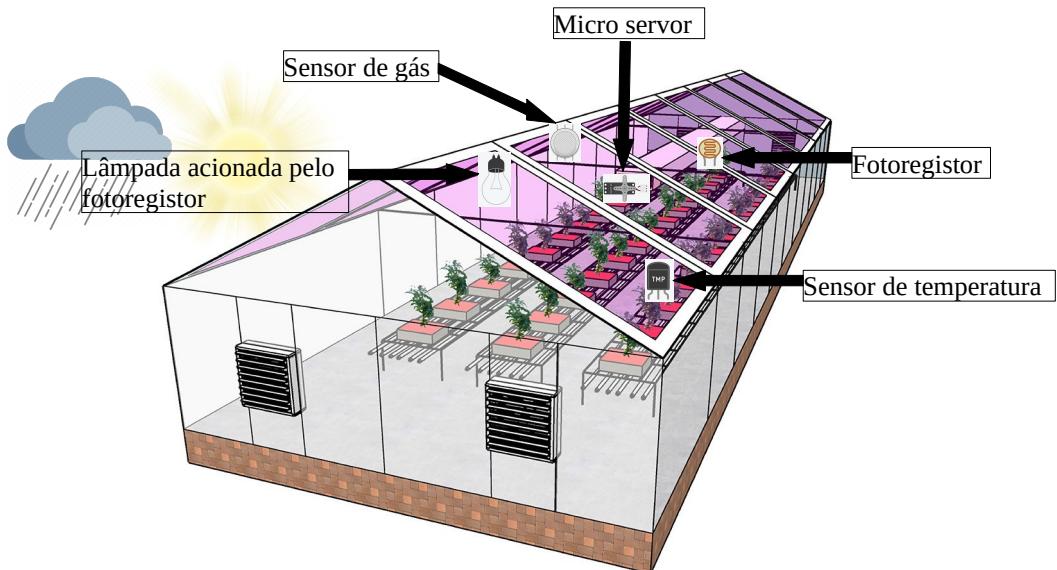


Figura 1: Big Picture

A figura 1 apresenta a Big Picture do trabalho dispostos simbolicamente em uma estufa este trabalho também pode ser adaptado para plantações externas.

## Storyboard



Ao entrar no sistema selecione a cultura desejada para irrigar.



Os dados de Temperatura, nível de gás e claridade serão coletados rotineiramente.



## 2.1. Modelagem do Sistema Proposto

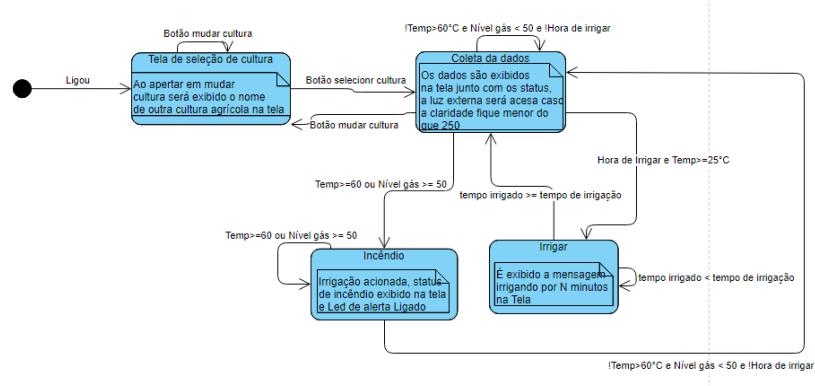
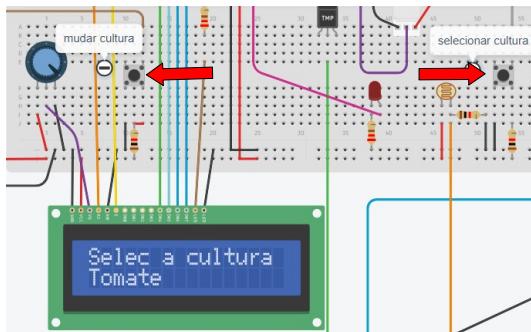
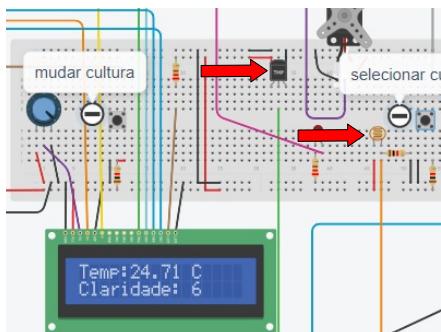


Figura 2: Máquina de estados

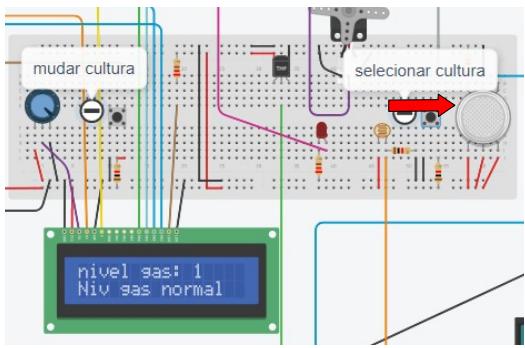
## Fluxo do sistema



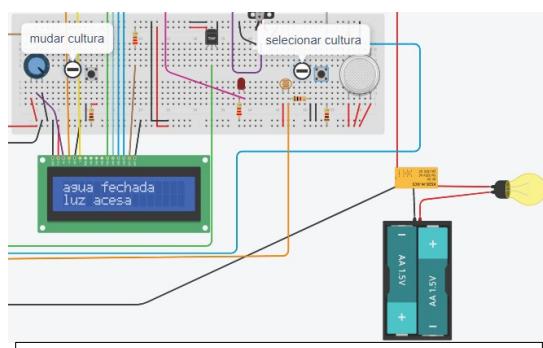
Ao ligar o sistema selecione uma das culturas disponíveis no sistema através dos botões indicados na imagem.



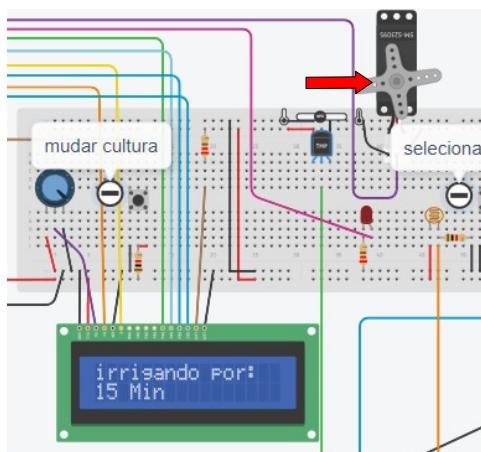
Após selecionar a cultura será apresentado alguns dados do ambiente, na figura acima é possível ver temperatura e claridade coletados dos sensores de temperatura e do fotoregistor indicados na imagem.



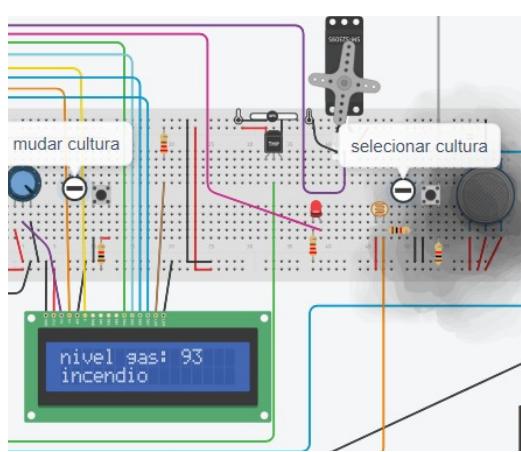
Nível de gás coletado só sensor de gás indicado na imagem se o nível de gás estiver abaixo de 50 está normal caso contrário o sistema entende como incêndio.



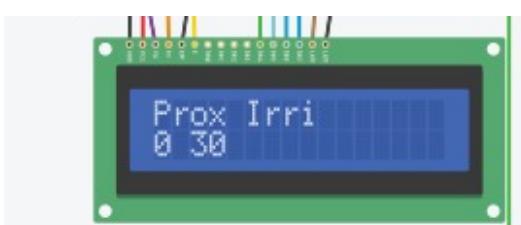
Exibição dos status da irrigação e da luz, para que a luz acenda é preciso um valor de claridade coletado pelo fotoregistor menor do que 250.



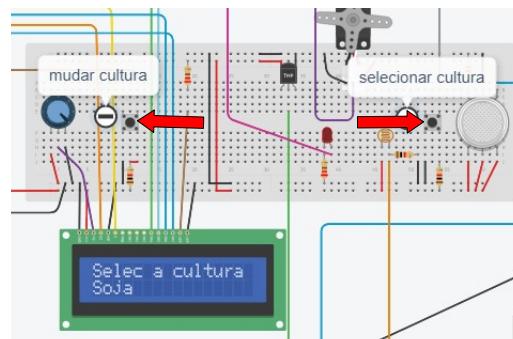
Após atingir um tempo pré determinado para a cultura selecionada e se a temperatura estiver acima de 25°C será iniciada a irrigação, o tempo de irrigação dependerá da cultura escolhida, o servo motor irá girar 180° para simular a torneira.



Caso o nível de gás ultrapasse 50 o sistema irá ligar o led vermelho como sinal de alerta e iniciar a irrigação na tentativa de apagar o fogo.



Um último parâmetro exibido na tela é o momento em que ocorrerá a próxima irrigação.



Caso o botão mudar cultura seja pressionado novamente o usuário é levado a o menu de seleção de cultura novamente onde deverá escolher a cultura desejada clicando nos botões indicados.

## 2.2. Esquema de Conexões

O microcontrolador utilizado neste trabalho foi o arduino uno sendo ele uma placa microcontroladora contendo o ATmega328P, o arduino possui 14 pinos de I/O digital, 6 pinos analógicos, um oscilador de 16MHz e comunicação USB.

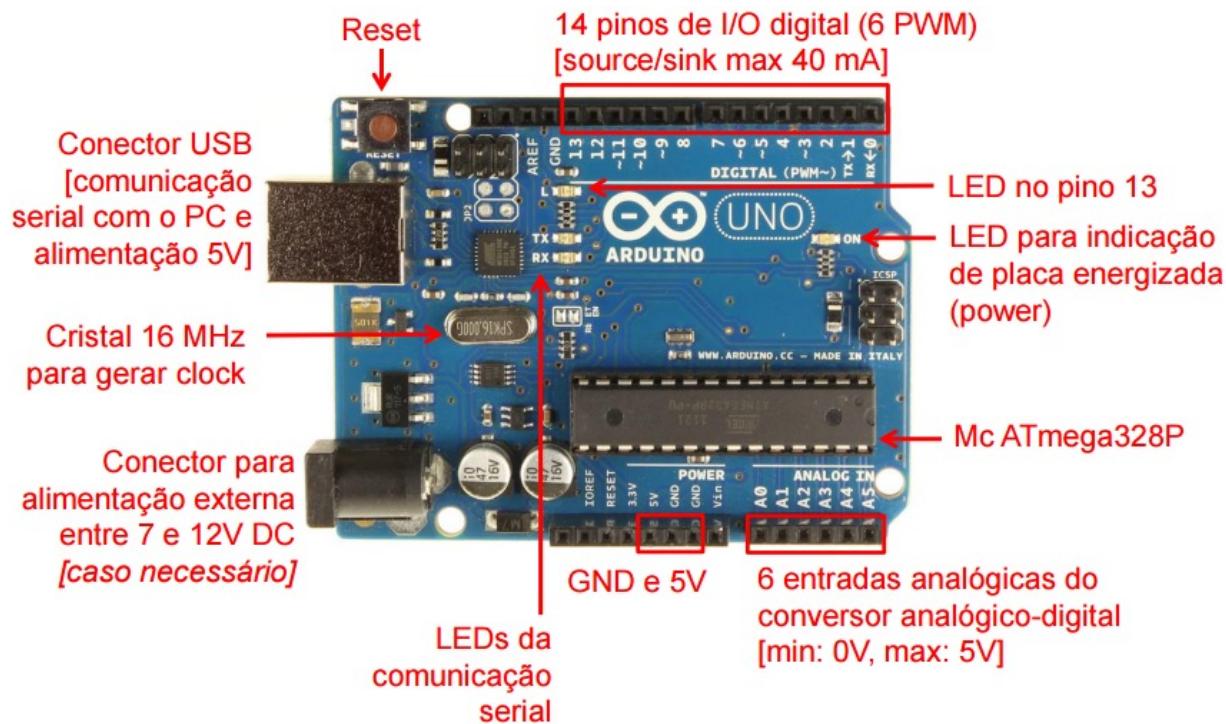


Figura 3: Arduino R3

Na figura 2 é possível visualizar o esquema de conexões presente em um arduino uno.

Esquema de conexões e pinagem dos componentes utilizados.

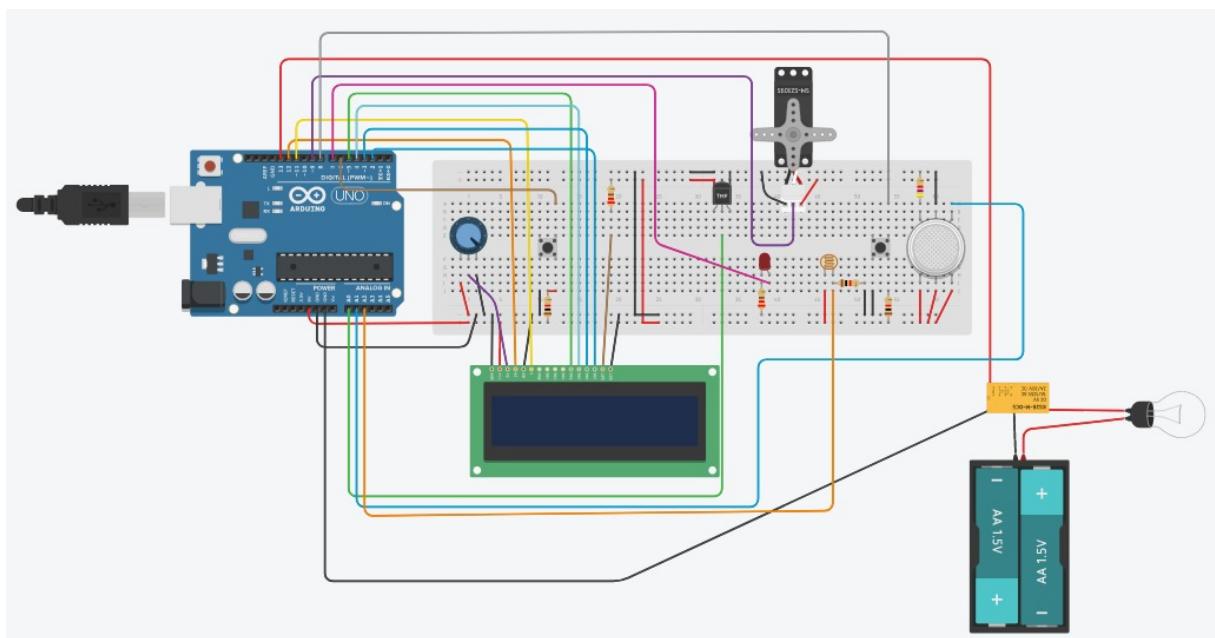


Figura 4: esquema de conexões

A figura 3 apresenta um esquemático de conexões dos componentes utilizados no projeto que consiste:

- Potenciômetro

Terminal 1: Pino 5V do arduino;

Limpador: Pino V0 da tela LCD;

Terminal 2: Pino GND do arduino;

- Tela LCD

GND: Pino GND do arduino;

VCC: Pino 5V do arduino;

V0: Pino Limpador do potenciômetro;

RS: Pino 12 do arduino;

RW: Pino GND do arduino;

E: Pino 11 do arduino;

DB4: Pino 5 do arduino;

DB5: Pino 4 do arduino;

DB6: Pino 3 do arduino;

DB7 Pino 2 do arduino;

LED anodo: Pino 5V através de um resistor de 220ohms;

LED catódico: Pino GND do arduino;

- Sensor de Temperatura[TMP36]

Potência: Pino 5V do arduino;

Vout: Pino A0 do arduino;

GND: Pino GND do arduino;

- Micro servor motor

Potência: Pino 5V do arduino;

Sinal: Pino 9 do arduino;

GND: Pino GND do arduino;

- Fotoregistor

Terminal 1: Pino 5V do arduino;

Terminal 2: Pino A2 do arduino e Pino GND do arduino através de um resistor de 1Kohm;

- Sensor de gás

A1: Pino GND do arduino através de um resistor de 4.7Kohm;

H1: Pino GND do arduino;

A2: Pino A1 do arduino;

B1: Pino 5V do arduino;

H2: Pino 5V do arduino;

B2: Pino 5V do arduino;

- LED Vermelho

Catodo: GND do arduino através de um resistor de 220ohms;

Anodo: Pino 7 do arduino;

- Push button 1

Terminal 2b: Pino 6 do arduino;

Terminal 1a: Pino GND do arduino;

Terminal 2a: Pino 5V do arduino através de um resistor de 1Kohm;

- Push button 2

Terminal 2b: Pino 8 do arduino;

Terminal 1a: Pino GND do arduino;

Terminal 2a: Pino 5V do arduino através de um resistor de 1Kohm;

- Relé DPDT

Terminal 1: Pino GND do arduino;

Terminal 16: Pino 13 do arduino;

Terminal 8: Terminal 1 da lâmpada;

Terminal 4: Negativo da bateria de 3V;

- Lâmpada

Terminal 1: Terminal 8 do Relé DPDT;

Terminal 2: Positivo da bateria de 3V;

- Bateria de 3V

Negativo: Terminal 4 do Relé DPDT;

Positivo: Terminal 2 da lâmpada;

### 3. Testes e Avaliação Experimental

Caso de Teste 1: Selecionando Tomate e mantendo a temperatura abaixo de 25°C sem fumaça e claridade acima de 250, o sistema exibiu os dados coletados normalmente e os status exibidos foram sempre torneira fechada, ambiente normal e luz apagada pois, como para haver irrigação um dos critérios é que a temperatura esteja acima de 25°C então não houve irrigação, como não havia fumaça o ambiente se manteve estável e por estar sendo coletado níveis de claridade acima de 250 a luz se manteve apagada.

Caso de Teste 2: Selecionando Soja e mantendo a temperatura acima de 25°C sem fumaça e claridade abaixo de 250, o sistema exibiu os dados coletados normalmente e os status exibidos foram de irrigando durante 10 minutos com o servo motor se movimentando 180° e num intervalo de 5 segundos foi exibido torneira fechada com o servo em 0° novamente, além disso foi exibido ambiente normal e luz ligada pois, como a temperatura estava acima de 25°C o sistema seguiu com a rotina normal de irrigação, com não havia fumaça o ambiente se manteve estável e por estar sendo coletado um nível de claridade abaixo de 250 a luz se manteve ligada.

Caso de Teste 3: Selecionando milho e mantendo a temperatura acima de 25°C com fumaça acima do nível 50 e nível de claridade abaixo de 250, o sistema exibiu os dados coletados normalmente porém, o status irrigando com o servo virado 180°, o LED vermelho foi ligado e o status do ambiente se alterou para incêndio, com luz ligada pois, como a temperatura estava acima de 25°C o sistema deveria seguir a rotina normal de irrigação porém como havia muita fumaça no ambiente o sistema reconheceu como um incêndio e manteve a irrigação ligada, acendeu a luz de alerta e como a claridade estava abaixo de 250 a luz se manteve ligada.

### 4. Considerações Finais

Apesar da limitação de não haver o sensor de umidade na plataforma TinkerCad foi possível simular um sistema de irrigação inteligente capaz de se ajustar para diferentes culturas que podem ser selecionadas pelo usuário, o sistema também capaz de reagir a possíveis casos de incêndios acionando a irrigação para apagar o fogo, onde é exibido o status de incêndio na tela LCD e o LED de alerta e ligado.

Link do projeto no tinkercad: <https://www.tinkercad.com/things/0OMK5Xkzhd-irrigacao>

Link Github: [https://github.com/henriqueannicchino/LuisHenrique\\_JoaoLucas\\_projetoFinal\\_ISE\\_UFRR\\_2020.2](https://github.com/henriqueannicchino/LuisHenrique_JoaoLucas_projetoFinal_ISE_UFRR_2020.2)

## 5. Referências

<https://www.tinkercad.com/things/1VC9zgkYWp5-escrevendo-no-lcd-pela-serial>

<https://www.tinkercad.com/things/3KpXc5UrWdB-micro-servo-motor-com-arduino>

<https://www.tinkercad.com/things/7KlkHaxx20c-tmp36-temperature-sensor-with-arduino>

<https://www.tinkercad.com/things/hrcCM7G0u3X-fotoresistor>

<https://www.tinkercad.com/things/3qAiJWseKP3-sensor-de-gas>

<https://www.tinkercad.com/things/aAOpHW2A2hq-acender-lampada-com-sensor-de-luz-e-de-presenca>