



CENTRO DE EDUCAÇÃO PROFISSIONAL HÉLIO AUGUSTO DE SOUZA

**SALA AUTÔNOMA
AUTOMAÇÃO RESIDENCIAL**

Curso: Técnico em Eletrônica

Turno: Noite

Mód.: III

Componente Curricular: Projeto

Alunos: Camilly Gurgel dos Santos

Nº02

Débora Cavalcante da Silva

Nº03

Julio Oscar Retamales

Nº08

Lucineide Maria de Brito

Nº 09

Wanderson Augusto da Cunhas Santos

Nº14

Professor Orientador: Antônio Carlos Cintra Pereira

São José dos Campos - SP

2023



Camilly Gurgel dos Santos
Débora Cavalcante da Silva
Julio Oscar Retamales
Lucineide Maria de Brito
Wanderson Augusto da Cunha Santos

Sala Autônoma

Projeto ou Monografia apresentado(a)
como requisito parcial para a obtenção do
título de técnico em Eletrônica pelo Centro
de Formação Profissional Hélio Augusto de
Souza em São José dos Campos

Professor Orientador: Antônio Carlos
Cintra Pereira

São José dos Campos - SP

2023



AGRADECIMENTOS

Dedico este trabalho a todos que, de alguma forma, foram pilares essenciais na construção deste caminho acadêmico. Cada página deste relatório técnico carrega um pouco da influência, apoio e inspiração dessas pessoas.

Aos nossos professores e orientadores, verdadeiros guias nesta jornada intelectual. Suas lições transcenderam a sala de aula, moldando nossos pensamentos críticos e inspirando-nos a ir além dos limites do conhecimento.

Às famílias, pelo suporte incondicional e compreensão nas horas de ausência. Cada conquista é compartilhada, e este trabalho é também uma expressão do nosso compromisso conjunto com a excelência.

Aos autores e pesquisadores cujas obras foram faróis de sabedoria em nosso percurso acadêmico. Cada citação, cada referência, é um tributo ao conhecimento que moldou nossa compreensão.

Que este relatório técnico não seja apenas um reflexo de nosso esforço, mas uma homenagem sincera a todos que contribuíram para nosso crescimento acadêmico e pessoal.



As raízes do estudo são amargas, mas seus frutos são doces.

- Aristóteles



Resumo

O sistema desenvolvido engloba o uso de sensores de presença, os quais interagem de forma sinérgica para determinar a abertura e fechamento automático das portas e janelas. A automação é efetuada por atuadores elétricos que garantem movimentos suaves e precisos, criando um ambiente adaptável às condições climáticas e às necessidades do usuário.

A eficiência energética é um ponto central na concepção deste projeto, com a implementação de algoritmos inteligentes que otimizam o uso de recursos elétricos, minimizando o consumo de energia durante o funcionamento do sistema. O resultado é uma sala automatizada que não apenas proporciona comodidade ao usuário, mas também atende a critérios sustentáveis.

O TCC apresenta os detalhes do desenvolvimento do sistema, desde a seleção de componentes até a programação dos algoritmos de automação. A metodologia aplicada incluiu simulações, prototipagem e testes práticos para validar a eficácia do sistema.

Os resultados obtidos indicam que a automação de portas e janelas não apenas simplifica a vida cotidiana do usuário, mas também contribui para a criação de ambientes mais eficientes e alinhados com as demandas contemporâneas. Este trabalho oferece uma contribuição significativa para a pesquisa em automação residencial, destacando-se pela integração de tecnologias avançadas e pela busca contínua por soluções inovadoras.

Palavras Chaves : Automação, Residencial, Sistema, Janelas e Portas



Abstract

The system developed involves the use of presence sensors, which interact synergistically to determine the automatic opening and closing of doors and windows. The automation is carried out by electric actuators that guarantee smooth and precise movements, creating an environment that is adaptable to climatic conditions and the user's needs.

Energy efficiency is central to the design of this project, with the implementation of intelligent algorithms that optimize the use of electrical resources, minimizing energy consumption during the system's operation. The result is an automated room that not only provides convenience for the user, but also meets sustainable criteria.

The TCC presents the details of the system's development, from the selection of components to the programming of the automation algorithms. The methodology applied included simulations, prototyping and practical tests to validate the system's effectiveness.

The results obtained indicate that door and window automation not only simplifies the user's daily life, but also contributes to the creation of more efficient environments in line with contemporary demands. This work makes a significant contribution to research into home automation, standing out for its integration of advanced technologies and its continuous search for innovative solutions.

Keywords : Automation, Residential, System, Windows and Doors



Lista de figuras

Figura 1	Sensor Ultrassônico	Página 12
Figura 2	Arduino	Página 13
Figura 3	Motor Linear	Página 14
Figura 4	CNC Shield	Página 15
Figura 5	Diagrama de Blocos	Página 15
Figura 6	Fluxograma	Página 16
Figura 7	Representação Led's e sensores	Página 18
Figura 8	Representação Circuito	Página 18



Lista de quadros

Quadro 1 Cronograma

Página 15



Sumário

1	Introdução	10
1.1	Objetivo	10
1.1.1	Objetivo Específico	10
1.2	Justificativa	10
2	Materiais	10
2.1	Sensores	11
2.1.1	Sensor Ultrassônico	11
2.1.2	Processamento de dados	11
2.2	Arduino	12
2.3	Tomada de Decisões	12
2.3.1	Linguagem C++	13
2.4	Controles	13
2.4.1	Motor Linear	13
2.4.2	CNC Shield	13
3	Métodos e Procedimentos	14
3.1	Cronograma	14
3.2	Diagrama de Bloco	14
3.3	Fluxograma	15
4	Desenvolvimento	16
4.1	Descrição de Montagem	16
4.2	Código	17
5	Resultado e Discussão	28
5.1	Resultado Geral	28
6	Conclusão	28



1. Introdução

Porque implantar a automação residencial no Brasil?

A automação residencial já não é mais algo do futuro. Cada vez mais populares, as tecnologias utilizadas dentro de casa estão transformando em realidade o que antes era apenas ficção.

Automação residencial diz respeito a um conjunto de tecnologias com a capacidade de programar eventos em uma casa e de tornar automático o funcionamento de diversos equipamentos.

Por meio de sistemas integrados e da conexão à internet, a automação residencial visa criar uma casa inteligente. Uma das tecnologias mais importantes nesse processo é a IoT (Internet of Things), que já vem impactando o dia a dia em muitas residências.

O objetivo é oferecer facilidades e mais praticidade aos moradores através do controle e do gerenciamento remoto da residência. Isso é feito por meio de um simples dispositivo móvel conectado à internet, como um smartphone ou tablet.

Assim, você pode controlar os aparelhos de casa de qualquer lugar que estiver. Além disso, é possível programar sensores, fechaduras e luzes, por exemplo, para que determinado comando seja executado automaticamente.

Além de facilitar a vida dos moradores, você pode criar uma casa totalmente personalizada conforme o seu estilo de vida, necessidades e hábitos.

A automação residencial gera várias vantagens para o consumidor final, como as facilidades no controle das funcionalidades das casas por meio de uma única interface, a possibilidade de gerar economia de energia com a automatização da iluminação e do controle de temperatura, além de outros aparelhos eletrônicos que podem ser desligados remotamente.

1.1. Objetivo

Mostra o funcionamento autônomo de uma forma simples de entender, onde qualquer um possa compreender a razão do porque se trata de uma boa opção a se pensar.

1.1.1. Objetivo específico

Material para estudos básicos da área de eletrônica com Arduino e C++

1.2. Justificativa

Para expandir o conceito de como funciona essa nova tecnologia, tornando possível mais pessoas trabalharem nessa área, onde possam produzir equipamentos autônomos que facilitem a qualidade de vida de todos.

2. Materiais

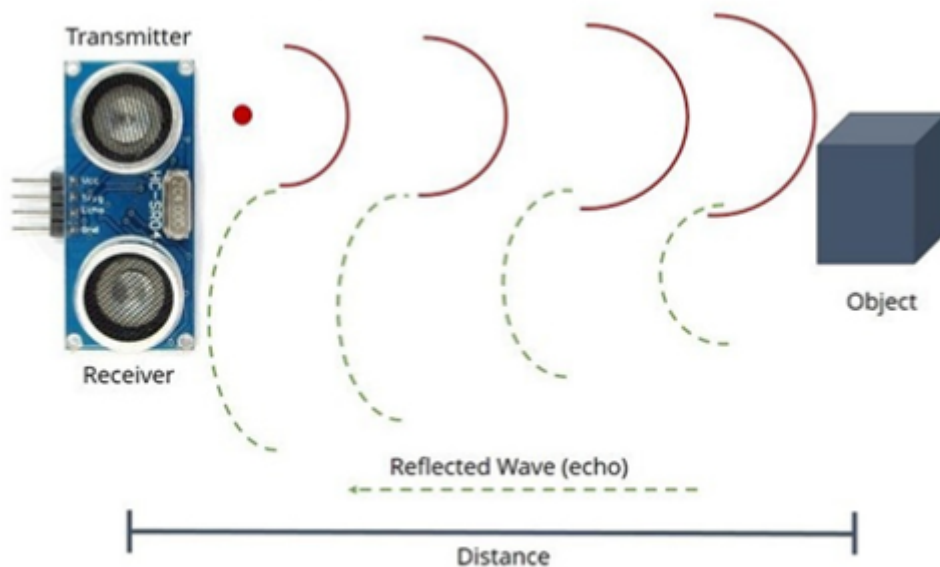
Uma maquete demonstrativa em escala reduzida de uma sala onde se apresenta janela e porta automatizadas, que se movem de forma independente sem a necessidade de intervenção humana. Esses dispositivos são equipados com sistemas de percepção, tomada de decisão e controle, permitindo a interação com o indivíduo de forma autônoma. A fundamentação teórica da sala automatizada envolve diversos conceitos e tecnologias fundamentais , que são discutidas nos textos a seguir

2.1. Sensores

Os sensores são componentes essenciais para maquete autônoma, pois fornecem informações sobre o ambiente externo e a presença de um indivíduo.

2.1.1. Sensor Ultrassônico

Utilizamos sensores ultrassônicos que funcionam medindo o tempo de propagação do eco, isto é, o intervalo de tempo entre o impulso sonoro emitido e o eco recebido de volta. Os sensores emitem pulsos de ultrassom ciclicamente. Quando um objeto reflete estes pulsos, o eco resultante é recebido e convertido em um sinal elétrico. Sendo o sensor principal para análise da área próximo a porta, compreendendo quando deve-se abrir a porta, mantê-la aberta ou fechá-la



Fonte: www.usinainfo.com.br

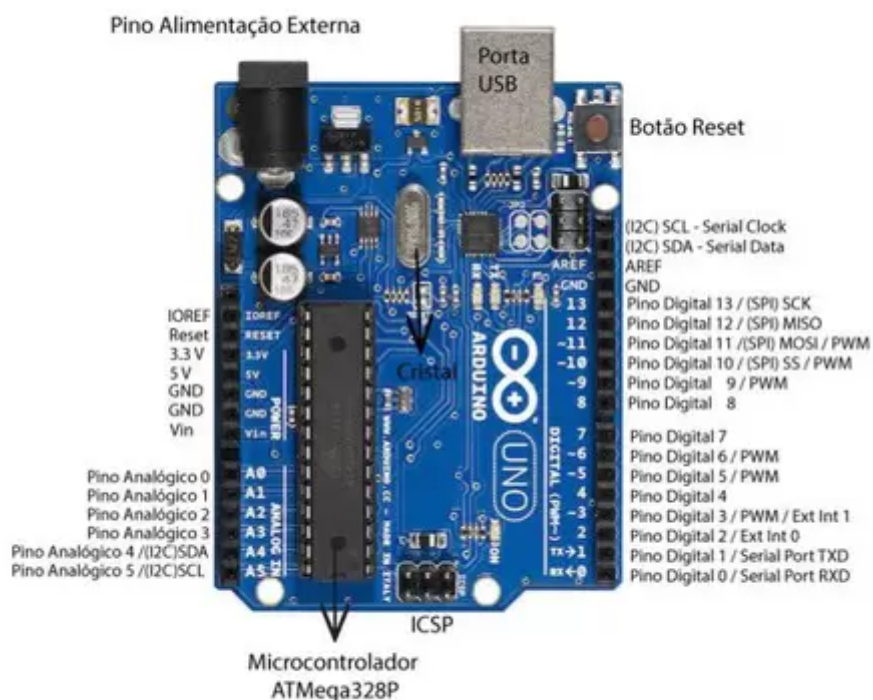
2.1.2. Processamento de dados

Após a coleta de dados pelos sensores, é necessário processar essas informações para que a porta automática possa tomar decisões. O processamento

de dados envolve etapas como filtragem, fusão de sensores e extração de características relevantes. Os algoritmos de aprendizado de máquina e visão computacional são amplamente utilizados para processar os dados dos sensores e extrair informações úteis.

2.2. Arduino

Utilizamos o arduino para processar os dados pois é uma placa de prototipagem eletrônica de código aberto (open-source) e hardware livre. Sendo bem fácil de mexer e programar, podendo usá-la para armazenar todos os dados necessários para o procedimento de análise da porta e janela automatizadas.



Fonte: portal.vidadesilicio.com.br

2.3. Tomada de Decisões

Com base nas informações coletadas e processadas, a porta deve tomar decisões sobre seu trajeto. Algoritmos de planejamento de trajetória são empregados para determinar a rota mais adequada, considerando limitações de velocidade, regras de direção e outros fatores relevantes. Além disso, os sistemas de tomada de decisão também devem considerar aspectos de segurança e priorizar ações que minimizem riscos.

2.3.1. Linguagem C++

Utilizamos C++ onde se torna bem mais útil para produção de um algoritmo pela flexibilidade e também por trabalhar em conjunto direto com o IDE do arduino, fazendo com que o robô faça a escolha de acordo com algoritmo programado em seu sistema.

2.4. Controles

2.4.1. Motor Linear

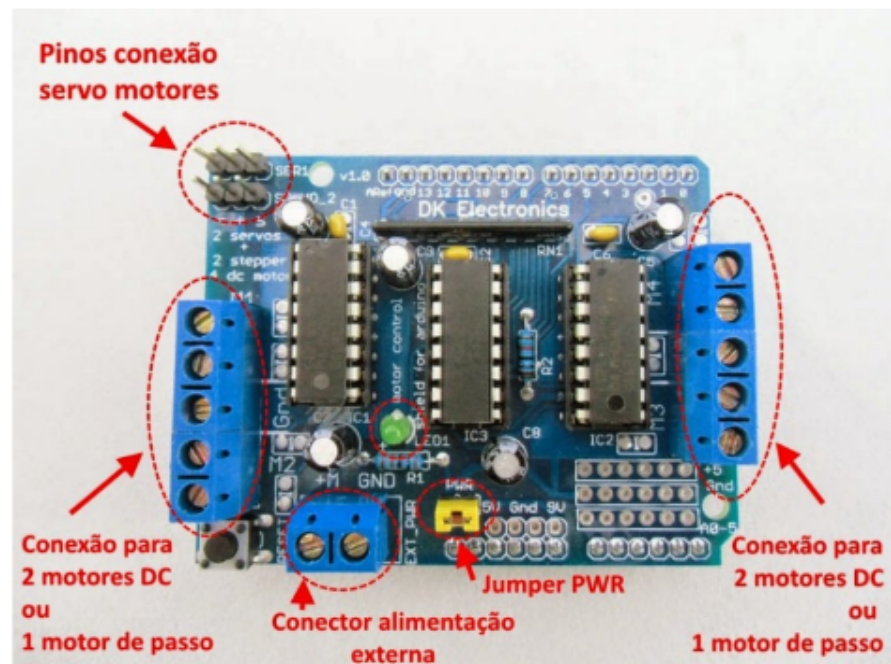
Foi feita a aplicação de um motor linear no projeto de automação de portas e janelas. O motor linear converte eficientemente energia elétrica de corrente contínua em movimento mecânico, sendo escolhido pela capacidade de controle preciso e reversibilidade. Amplamente utilizado em automação residencial, o motor linear oferece eficiência energética e resposta rápida, tornando-o ideal para sistemas que exigem movimentos suaves e bidirecionais.



Fonte: <https://www.amazon.com.br/lineares-acabamento-deslizante-equipamentos-deslizantes/dp/B097X521CQ>

2.4.2. CNC Shield

Uma extensão do Arduino para o controle dos motores, ele controla melhor o circuito recebendo informações do arduino e passando com mais precisão para cada motor, sendo eficiente também por conseguir colocar outros equipamentos nele para trabalhar em conjunto de acordo com que foi programado e passado para ele do Arduino, exemplo como o uso do sensor ultrassônico e do servo motores.



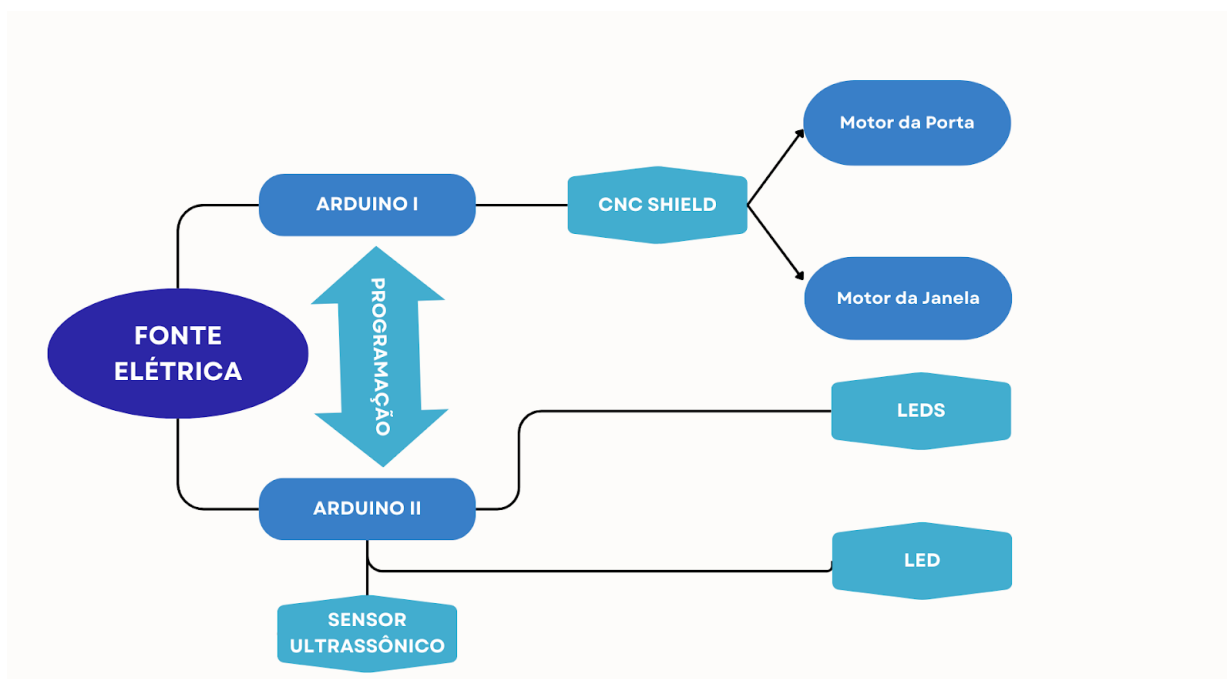
Fonte:arduinoocia.com.br

3. Métodos e Procedimentos

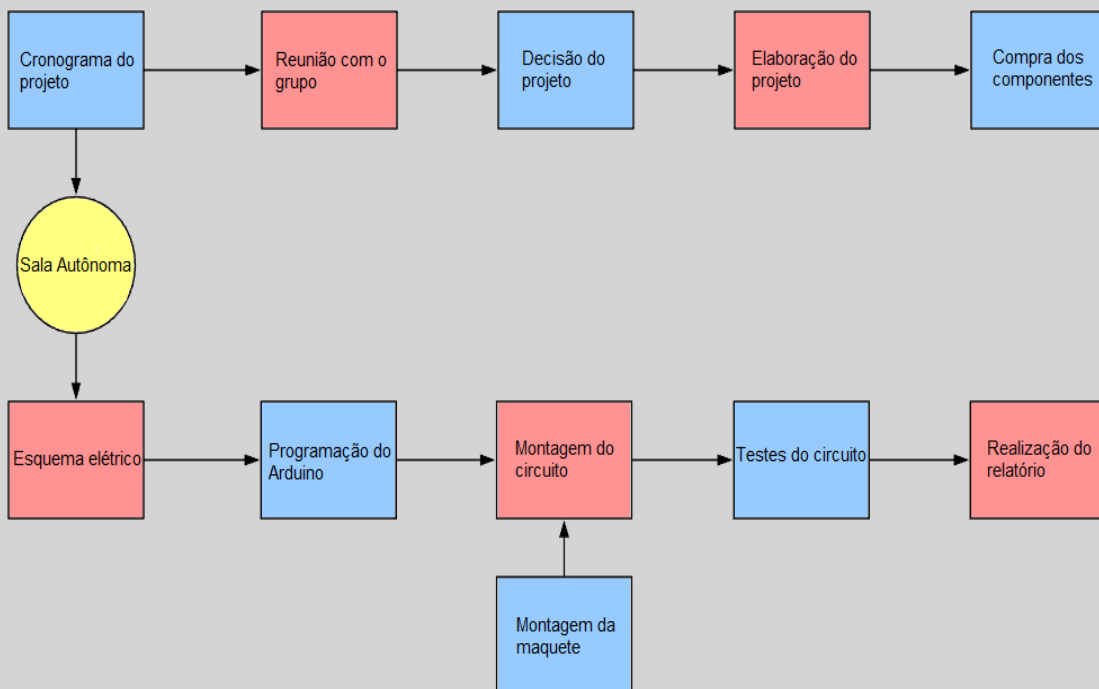
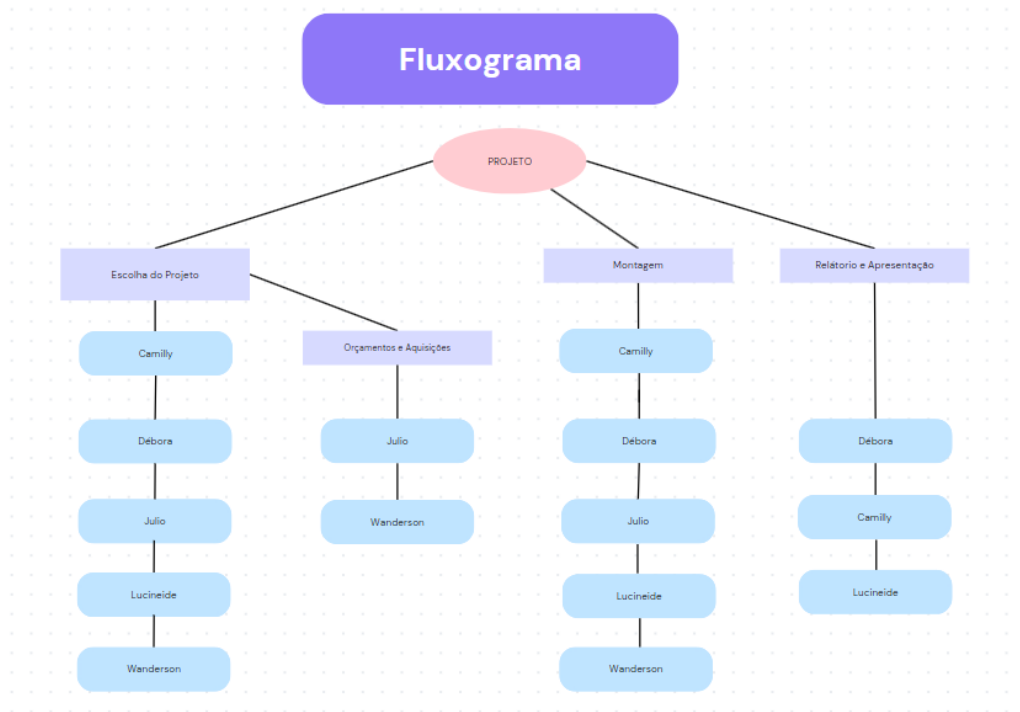
3.1. Cronograma

Materias / Custos	Produção da programação	Conceito de Montagem	Processo de montagem	Testes/ Finalizações
4 Semanas	4 Semanas	2 Semanas	2 Semana	2 Semanas

3.2. Diagramas de Bloco



3.3. Fluxograma





4. Desenvolvimento

4.1. Descrição de Montagem

Dando início ao desenvolvimento, iniciamos primeiramente com o desenvolvimento do Código de programação que viria ser utilizado, onde inserimos os comandos do monitor serial para cada componente presente na maquete, dando ênfase nos comandos da CNC Shield.

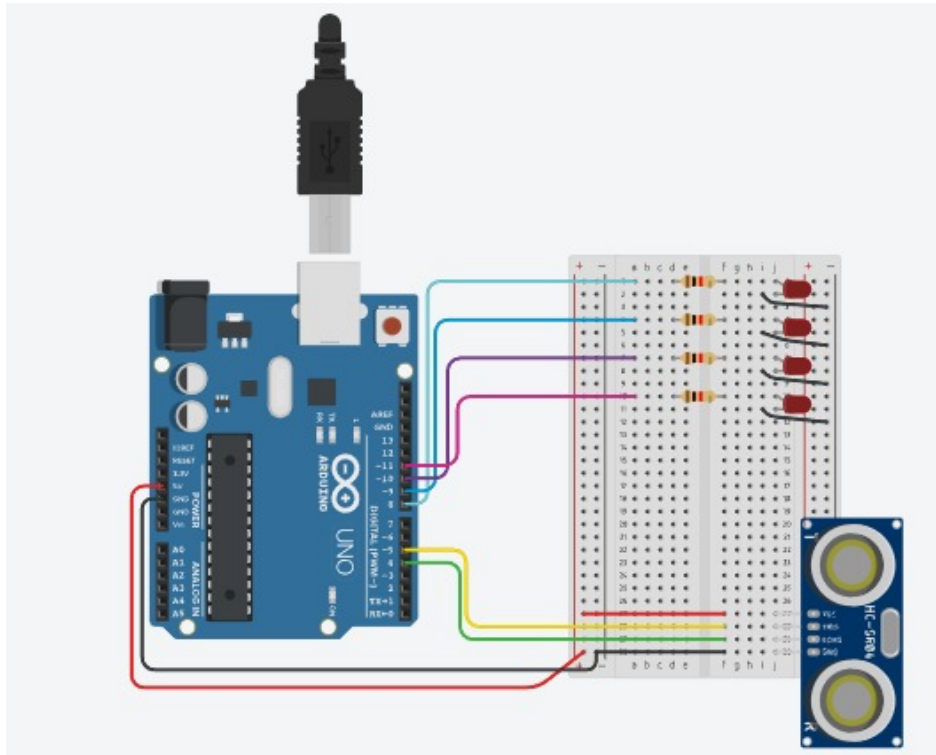
Para a ligação dos Motores Lineares a Shield foi necessário soldagem para os fios (o processo de inserção dos motores a maquete e soldagem dos fios foram repetidas para a porta e para a janela).

Dando continuidade, inserimos cada motor linear em seus respectivos locais, onde em conjunto foram inseridas as peças que seriam utilizadas para simbolizar a porta e janela. A porta terá como adicional de comando o sensor ultrassônico que ao detectar a presença de um indivíduo dentro da delimitação determinada irá acionar automaticamente a abertura da porta.

Após esses procedimentos, nos atentamos à montagem e colocação de cada componente em cada local determinado, onde utilizamos o uso de parafusos para fixar os motores lineares responsáveis pelo deslocamento da porta e janela. Para a Maquete, realizamos a perfuração para a entrada do Sensor Ultrassônico ao motor linear correspondente a porta e as ligações necessárias para a Shield. Concluída essa fase, inserimos o arduino mega ao centro da maquete, e logo interligamos e posicionamos a Shield acima do arduino.

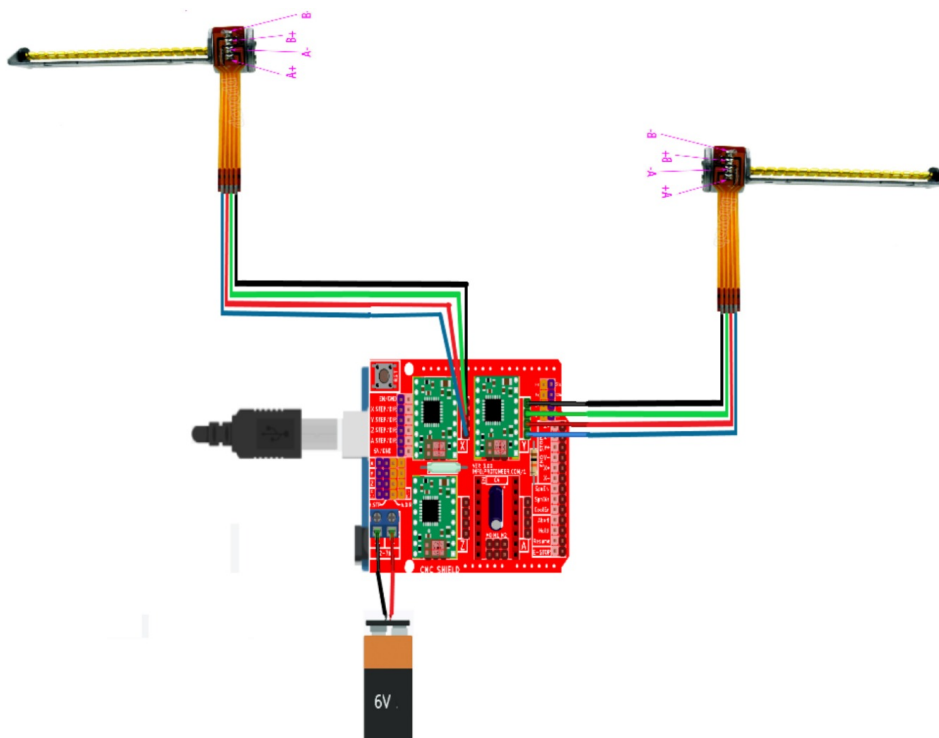
Como complemento, adicionamos a maquete e o código de programação comando para alguns LED's que vieram a ser adicionados na maquete para simbolizar o funcionamento correto do Sensor Ultrassônico, iluminação, cafeteira e outros equipamentos e sistemas que podem ser ativados nesse processo de automação.

Representação Led's e sensores



Logo após, colocamos o cabo USB do arduino ao próprio e conectamos ao computador para transferência do código criado para o arduino. O código foi criado com programação C++.

Por fim, para conceder a energia que dará funcionamento aos motores fizemos a utilização de uma fonte de 12V junto a um regulador de tensão para que pudéssemos controlá-la a 6V, tensão essa necessária e mais do que suficiente para um bom funcionamento dos motores lineares.





Na representação acima foi utilizada de um pilhas de 6V, para simbolizar a fonte de 12V com o regulador de tensão.

4.2. Código

```
const int enPin = 8;
const int stepXPin = 2; // X.STEP
const int dirXPin = 5; // X.DIR
const int stepYPin = 3; // Y.STEP
const int dirYPin = 6; // Y.DIR
int pulseWidthMicros = 100; // microseconds
int millisBtwnSteps = 1000;
const int openDuration = 300;
const int pinLedA = 31; //lampada do quarto
const int pinLedB = 33; //cafeteira
const int pinLedC = 35; //radio
const int pinLedD = 39; //Luz do sensor
const int pinTrig = 24;
const int pinEcho = 26; //branco
const int Media_Interactions = 10;
const int Distance_Treshold = 10;
bool has_AutoGate = false;
String Serial_Data = "";
String Serial1_Data = "";
void setup()
{
  Serial.begin(9600);
  Serial.setTimeout(10);
  Serial.flush();
  Serial.println("Serial USB Inicializada.");
```



```
Serial1.begin(9600);
Serial1.setTimeout(10);
Serial1.flush();
Serial.println("Serial 1 Inicializada.");
Serial1.println("Serial 1 Inicializada.");
pinMode(enPin, OUTPUT);
digitalWrite(enPin, LOW);
pinMode(stepXPin, OUTPUT);
pinMode(dirXPin, OUTPUT);
pinMode(stepYPin, OUTPUT);
pinMode(dirYPin, OUTPUT);
pinMode(pinLedA, OUTPUT);
pinMode(pinLedB, OUTPUT);
pinMode(pinLedC, OUTPUT);
pinMode(pinLedD, OUTPUT);
pinMode(pinTrig, OUTPUT);
pinMode(pinEcho, INPUT);
Serial.println(F("CNC Shield iniciada"));
}
void loop()
{
  Read_Serial();
  Read_Serial1();
  distancia();
}
//=====
=====STRING SPLIT
```



```
void Read_Serial()
{
    if (Serial.available() > 0)
    {
        int inChar = Serial.read();
        if (inChar != '\n')
        {
            Serial_Data += (char)inChar;
        }
        else
        {
            Split_String();
        }
    }
}

void Read_Serial1()
{
    if (Serial1.available() > 0)
    {
        int inChar = Serial1.read();
        if (inChar != '\n')
        {
            Serial1_Data += (char)inChar;
        }
        else
        {
            Split_String_1();
        }
    }
}
```



```
}  
}  
void Split_String()  
{  
    int CommaIndex = 0;  
    for (int i = 0; i < Serial_Data.length(); i++)  
    {  
        if (Serial_Data[i] == ',')  
        {  
            CommaIndex = i;  
        }  
    }  
  
    int State = (Serial_Data.substring(Serial_Data.length() - 2,  
Serial_Data.length())).toInt();  
    int Port = (Serial_Data.substring(1, CommaIndex)).toInt();  
    Switch_Action(Serial_Data);  
    Serial_Data = "";  
}  
void Split_String_1()  
{  
    Switch_Action(Serial1_Data);  
    //Serial.print("=====> Serial BT DATA ");  
    //Serial.println(Serial1_Data);  
    Serial1_Data = "";  
}  
void moveMotor(int stepPin, int dirPin, int duration)  
{  
    unsigned long startTime = millis();
```



```
while (millis() - startTime < duration)
{
    digitalWrite(dirPin, HIGH);
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(pulseWidthMicros);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(millisBtwnSteps);
}
}

void reverseMoveMotor(int stepPin, int dirPin, int duration)
{
    unsigned long startTime = millis();
    while (millis() - startTime < duration)
    {
        digitalWrite(dirPin, LOW); // Reverse
        digitalWrite(stepPin, HIGH);
        delayMicroseconds(pulseWidthMicros);
        digitalWrite(stepPin, LOW);
        delayMicroseconds(millisBtwnSteps);
    }
}

void Switch_Action(String command)
{
    char Value[20];
    command.toCharArray(Value, 20);
    Serial.println(Value[0]);
    switch (Value[0])
    {
```



case '1':

```
Serial.println(F("Opening Motor X"));
Serial1.println(F("Opening Motor X"));
moveMotor(stepXPin, dirXPin, openDuration);
Serial.println(F("Motor X Stopped"));
Serial1.println(F("Motor X Stopped"));
break;
```

case '2':

```
Serial.println(F("Closing Motor X"));
Serial1.println(F("Closing Motor X"));
reverseMoveMotor(stepXPin, dirXPin, openDuration);
Serial.println(F("Motor X Stopped"));
Serial1.println(F("Motor X Stopped"));
break;
```

case '4':

```
Serial.println(F("Opening Motor Y"));
Serial1.println(F("Opening Motor Y"));
moveMotor(stepYPin, dirYPin, openDuration);
Serial.println(F("Motor Y Stopped"));
Serial1.println(F("Motor Y Stopped"));
break;
```

case '5':

```
Serial.println(F("Closing Motor Y"));
Serial1.println(F("Closing Motor Y"));
reverseMoveMotor(stepYPin, dirYPin, openDuration);
Serial.println(F("Motor Y Stopped"));
Serial1.println(F("Motor Y Stopped"));
break;
```



case '7':

```
Serial.println(F("Simultaneously Opening Motors X and Y"));
Serial1.println(F("Simultaneously Opening Motors X and Y"));
moveMotor(stepXPin, dirXPin, openDuration);
moveMotor(stepYPin, dirYPin, openDuration);
Serial.println(F("Motors X and Y Stopped"));
Serial1.println(F("Motors X and Y Stopped"));
break;
```

case '8':

```
Serial.println(F("Simultaneously Closing Motors X and Y"));
Serial1.println(F("Simultaneously Closing Motors X and Y"));
reverseMoveMotor(stepXPin, dirXPin, openDuration);
reverseMoveMotor(stepYPin, dirYPin, openDuration);
Serial.println(F("Motors X and Y Stopped"));
Serial1.println(F("Motors X and Y Stopped"));
break;
```

case '3':

```
Serial.println(F("Custom Command: Opening Motor X and Closing Motor Y"));
Serial1.println(F("Custom Command: Opening Motor X and Closing Motor Y"));
moveMotor(stepXPin, dirXPin, openDuration);
reverseMoveMotor(stepYPin, dirYPin, openDuration);
Serial.println(F("Motors X and Y Stopped"));
Serial1.println(F("Motors X and Y Stopped"));
break;
```

case '6':

```
Serial.println(F("Custom Command: Opening Motor Y and Closing Motor X"));
Serial1.println(F("Custom Command: Opening Motor Y and Closing Motor X"));
moveMotor(stepYPin, dirYPin, openDuration);
```




```
reverseMoveMotor(stepXPin, dirXPin, openDuration);  
Serial.println(F("Motors X and Y Stopped"));  
Serial1.println(F("Motors X and Y Stopped"));  
break;  
case 'a':  
    digitalWrite(pinLedA, !digitalRead(pinLedA)); // Inverte o estado do LED  
    break;  
case 'b':  
    digitalWrite(pinLedB, !digitalRead(pinLedB));  
    break;  
case 'c':  
    digitalWrite(pinLedC, !digitalRead(pinLedC));  
    break;  
case 'd':  
    Serial.println(F("Opening Motor X"));  
    Serial1.println(F("Opening Motor X"));  
    moveMotor(stepXPin, dirXPin, openDuration);  
    Serial.println(F("Motor X Stopped"));  
    Serial1.println(F("Motor X Stopped"));  
    Serial.println(F("Opening Motor Y"));  
    Serial1.println(F("Opening Motor Y"));  
    moveMotor(stepYPin, dirYPin, openDuration);  
    Serial.println(F("Motor Y Stopped"));  
    Serial1.println(F("Motor Y Stopped"));  
    delay(100);  
    digitalWrite(pinLedA, LOW);  
    digitalWrite(pinLedC, HIGH);  
    delay(100);
```



```
digitalWrite(pinLedB, HIGH);
delay(5000);
digitalWrite(pinLedC, LOW);
break;

case 'n':
Serial.println(F("Closing Motor X"));
Serial1.println(F("Closing Motor X"));
reverseMoveMotor(stepXPin, dirXPin, openDuration);
Serial.println(F("Motor X Stopped"));
Serial1.println(F("Motor X Stopped"));
Serial.println(F("Closing Motor Y"));
Serial1.println(F("Closing Motor Y"));
reverseMoveMotor(stepYPin, dirYPin, openDuration);
Serial.println(F("Motor Y Stopped"));
Serial1.println(F("Motor Y Stopped"));
digitalWrite(pinLedA, LOW);
digitalWrite(pinLedB, LOW);
digitalWrite(pinLedC, LOW);
break;

case 't':
Serial.println(F("Opening Motor X"));
Serial1.println(F("Opening Motor X"));
moveMotor(stepXPin, dirXPin, openDuration);
Serial.println(F("Motor X Stopped"));
Serial1.println(F("Motor X Stopped"));
Serial.println(F("Closing Motor Y"));
Serial1.println(F("Closing Motor Y"));
reverseMoveMotor(stepYPin, dirYPin, openDuration);
```



```
Serial.println(F("Motor Y Stopped"));
Serial1.println(F("Motor Y Stopped"));
digitalWrite(pinLedA, HIGH);
delay(100);
digitalWrite(pinLedB, LOW);
delay(100);
digitalWrite(pinLedC, HIGH);
break;
default:
    // Handle unknown commands or provide a default case if needed.
    break;
}
}

void distancia() {
    digitalWrite(pinTrig, LOW);
    delayMicroseconds(2);
    digitalWrite(pinTrig, HIGH);
    delayMicroseconds(10);
    digitalWrite(pinTrig, LOW);

    long duration = pulseIn(pinEcho, HIGH);
    int distance = duration * 0.034 / 2;
    //Retirar na hora do projeto a Serial.Print
    //Serial.print("Distancia: ");
    //Serial.print(distance);
    //Serial.println(" cm");
    if (distance <= 5) {
        digitalWrite(pinLedD, HIGH); // Liga o LED
```



```
reverseMoveMotor(stepXPin, dirXPin, openDuration);  
Serial.println(F("Opening Motor X"));  
Serial1.println(F("Opening Motor X"));  
// Aguarda 3 segundos  
delay(1000);  
Serial.println(F("aguardando X"));  
Serial1.println(F("aguardando X"));  
delay(3000);  
//Inverte o motor após 3 segundos  
moveMotor(stepXPin, dirXPin, openDuration);  
digitalWrite(pinLedD, LOW);  
Serial.println(F("Closing Motor X"));  
Serial1.println(F("Closing Motor X"));  
// Mantém-se disponível para detecção contínua  
}  
// Delay opcional para evitar leituras muito frequentes  
delay(1000);  
}
```

5. Resultados e Discussão

5.1. Resultados Gerais

A maquete executa as funções que lhe foram delegadas, como utilizar-se de um sensor ou de comando através do celular para acionar os motores e assim abrir e fechar a porta e janela.

6. Conclusão

A conclusão deste estudo reforça a visão de uma revolução na concepção de espaços residenciais, materializada na implementação da sala autônoma com portas e janelas automáticas. Ao longo deste trabalho, mergulhamos nas nuances da automação residencial, destacando não apenas a complexidade técnica envolvida, mas também os benefícios tangíveis proporcionados por esse ambiente inovador.

A sala autônoma apresentada não é simplesmente uma demonstração de tecnologia avançada; é uma resposta pragmática às necessidades contemporâneas,



buscando criar ambientes que se moldem dinamicamente às demandas do usuário e do meio ambiente. A concepção cuidadosa do projeto, desde o design até a seleção de componentes, reflete a importância de encontrar um equilíbrio entre funcionalidade, estética e sustentabilidade.

Ao contemplar os resultados obtidos, fica claro que a sala autônoma é mais do que uma mera conveniência tecnológica; é uma promessa de espaços residenciais mais inteligentes, conectados e adaptáveis. Este projeto representa um passo significativo em direção a uma nova era na concepção de ambientes domésticos, onde a tecnologia não apenas facilita a vida cotidiana, mas também contribui para a criação de lares mais eficientes e sustentáveis.

As perspectivas futuras desta abordagem envolvem a continuidade da pesquisa e inovação, com a expansão das aplicações da automação residencial em diversos contextos. À medida que contemplamos o horizonte, vislumbramos residências que não apenas incorporam tecnologias avançadas, mas também evoluem constantemente para atender às necessidades dinâmicas de seus habitantes.

Neste ponto, encerramos este estudo, cientes de que a sala autônoma representa não apenas uma conquista técnica, mas uma visão tangível de como podemos transformar nossas residências em espaços mais inteligentes, sustentáveis e adaptáveis às demandas do futuro.



Referências

<https://web.tecnico.ulisboa.pt/~ist186955/wordpress/2016/12/14/162/#:~:text=Um%20motor%20DC%20%C3%A9%20um,for%C3%A7as%20produzidas%20por%20campos%20magn%C3%A9ticos.>

<http://www.um.pro.br/arduino/index.php?c=ArduinoUno>

<https://srituhobby.com/how-to-use-a-cnc-shield-step-by-step-guide-with-grbl/>

<https://smallhammer.cc/product/sd1>

<https://www.usinainfo.com.br/sensor-ultrassonico-502>