

Aluno (a): _____ RA: ._____.

Aluno (a): _____ RA: ._____.

Aluno (a): _____ RA: ._____.

Valor: 25,0 pts Data: 03/11/2024 Nota: _____

ROTEIRO DE TRABALHO FINAL

Desenvolvimento de um Sistema Embarcado no EasyEDA

Introdução:

O desenvolvimento de sistemas embarcados tem desempenhado um papel fundamental na evolução tecnológica, permitindo a integração de hardware e software em soluções compactas e eficientes. Com o avanço das tecnologias de microcontroladores e o surgimento de plataformas de design de circuitos eletrônicos acessíveis, como o EasyEDA, tornou-se mais viável e acessível criar sistemas embarcados personalizados.

Neste trabalho final, propomos a criação de um sistema embarcado utilizando a plataforma EasyEDA. É uma ferramenta online poderosa que oferece recursos completos de esquemáticos, design de PCB e simulação de circuitos eletrônicos. Essa plataforma simplifica o processo de desenvolvimento, permitindo a visualização e a prototipagem rápida de projetos de sistemas embarcados.

O objetivo deste trabalho é proporcionar aos participantes uma oportunidade prática de projetar, desenvolver e implementar um sistema embarcado funcional, abrangendo desde a fase de definição de requisitos até a fabricação do PCB e a depuração do sistema. Durante o desenvolvimento do projeto, os participantes ganharão experiência em diversas etapas, como a criação de esquemáticos, o design de PCB, a simulação de circuitos e a realização de testes e depuração.

Além disso, este trabalho final tem como propósito incentivar a criatividade e a inovação, permitindo que os participantes escolham a função específica que o sistema embarcado irá desempenhar. Pode ser um sistema de controle de temperatura, um dispositivo de automação residencial, um sistema de monitoramento de sensores, entre outros. A liberdade de escolha permite que os participantes explorem áreas de interesse e desenvolvam soluções adaptadas às necessidades e demandas atuais.

Ao final do projeto, espera-se que os participantes sejam capazes de compreender e aplicar os conceitos de design de sistemas embarcados, utilizando o EasyEDA como ferramenta para projetar, simular e fabricar um sistema funcional. Além disso, a documentação detalhada do projeto garantirá a replicabilidade e servirá como referência para futuros desenvolvimentos nessa área.

O presente roteiro de trabalho final fornecerá as diretrizes e etapas para a execução do projeto, permitindo que os participantes adquiram habilidades práticas e conhecimentos sólidos na área de sistemas embarcados. O desenvolvimento dessas competências será fundamental para enfrentar os desafios tecnológicos e promover a inovação no campo dos sistemas embarcados.

TRABALHO FINAL

Neste trabalho final, nosso objetivo é explorar e desenvolver um sistema embarcado utilizando a plataforma EasyEDA. O EasyEDA, uma ferramenta online completa, nos oferece recursos essenciais, como criação de esquemáticos, design de PCB e simulação de circuitos eletrônicos, tornando o processo de desenvolvimento mais acessível e eficiente.

O foco principal será projetar e implementar um sistema embarcado personalizado, com a finalidade de desempenhar uma função específica que atenda às necessidades e demandas atuais. Podemos considerar diversas opções, como controle de temperatura para aplicações industriais ou residenciais, automação residencial para melhorar o conforto e a eficiência energética, ou até mesmo um sistema de monitoramento de sensores para coleta de dados ambientais ou de saúde.

Ao escolher a função específica do sistema embarcado, levaremos em consideração a relevância e o impacto prático da solução proposta, bem como a complexidade técnica e os recursos disponíveis. Isso nos permitirá explorar e aplicar os recursos do EasyEDA de forma eficaz para projetar o circuito eletrônico, desenvolver o layout da placa de circuito impresso e realizar simulações para validar o funcionamento do sistema.

Ao final deste trabalho, teremos criado um sistema embarcado funcional e customizado, demonstrando a nossa capacidade de aplicar os conhecimentos adquiridos em design de sistemas embarcados e utilizar o EasyEDA como uma ferramenta poderosa para o desenvolvimento de projetos eletrônicos.

Etapa 1: Definição dos requisitos do sistema

1. Definir a função específica que o sistema embarcado irá realizar.
2. Determinar os requisitos básicos do sistema, como sensores, atuadores, interface de usuário, conectividade, etc.
3. Documentar os requisitos em detalhes para orientar o desenvolvimento do projeto.

Etapa 2: Esquemático

1. Utilizar o EasyEDA para criar o esquemático do sistema embarcado.
2. Adicionar os componentes necessários, incluindo microcontrolador, sensores, atuadores, interfaces de comunicação, etc.
3. Conectar os componentes de acordo com as especificações do projeto.
4. Verificar se o esquemático está correto e sem erros.

Etapa 3: Design de PCB

1. Utilizar o EasyEDA para realizar o design de PCB com base no esquemático.
2. Posicionar os componentes no layout da placa de acordo com as restrições de espaço e requisitos de projeto.
3. Rotear as trilhas para garantir a correta conexão elétrica entre os componentes.
4. Verificar se o design de PCB atende às especificações do projeto e às limitações de fabricação.

Etapa 4: Simulação

1. Utilizar as ferramentas de simulação do EasyEDA para testar o funcionamento do circuito eletrônico.
2. Verificar se o sistema embarcado atende aos requisitos definidos na etapa de definição.
3. Realizar simulações de diferentes cenários para garantir a robustez e confiabilidade do sistema.

Etapa 5: Fabricação e montagem

Exportar os arquivos Gerber a partir do EasyEDA para fabricação da placa de circuito

Entrega:

O trabalho deve ser entregue no diário de bordo no formato pdf contendo:

- Introdução
- Objetivo do sistema embarcado
- Lista de componentes
- Esquemático elétrico
- Placa de circuito impresso
- Imagem 3D do circuito
- Conclusão
- ***não obrigatório:** Simulação; programação; PCB; montagens e testes.

Cronograma de Desenvolvimento:

O cronograma de desenvolvimento do trabalho final é desenvolvido em 3 etapas, sendo:

1. Terça-feira, 12/11/2024:
 - Definir a função específica do sistema embarcado.
 - Pesquisar e selecionar os principais componentes eletrônicos necessários.
2. Terça-feira, 19/11/2024:
 - Criar o esquemático do circuito eletrônico no EasyEDA.
 - Iniciar o design da placa de circuito impresso (PCB) no EasyEDA.

3. Terça-feira, 26/11/2024:

- Concluir o design da placa de circuito impresso (PCB) no EasyEDA.
- Realizar a simulação do circuito eletrônico para verificar seu funcionamento.

4. Terça-feira, 03/12/2024:

- Data final de envio do trabalho.

Finalizar o projeto da placa de circuito impresso (PCB) com todos os componentes posicionados corretamente.

ETAPA 01

Etapla 01: Definição dos requisitos do sistema embarcado

O sistema embarcado será composto pelos seguintes componentes essenciais que são:

Microcontrolador: Você terá a liberdade de escolher um microcontrolador adequado às necessidades do projeto. O microcontrolador será responsável por controlar as operações do sistema e executar as funções desejadas.

Sensor(es) e/ou atuador(es): Será utilizado pelo menos um sensor e/ou atuador, dependendo dos requisitos do sistema. Os sensores serão responsáveis por coletar dados do ambiente, enquanto os atuadores serão responsáveis por realizar ações físicas com base nessas informações.

Pinos de conexão: Serão incluídos pinos de conexão para alimentação externa, permitindo a conexão de uma fonte de energia ao sistema. Além disso, serão fornecidos pinos de conexão específicos para a gravação do microcontrolador, facilitando a programação e a depuração do sistema.

Além dos componentes obrigatórios mencionados acima, também serão utilizados outros componentes básicos, como **resistores** e **diódos**, que desempenham papéis cruciais no circuito elétrico do sistema embarcado.

A escolha e a especificação detalhada dos componentes específicos dependerão dos requisitos do projeto e das funcionalidades desejadas para o sistema embarcado.

Sugestões:

1. Sistema embarcado para acionamento de lâmpada e(ou) outros dispositivos remotamente.
2. Sistema de controle de acesso com RFID, teclado, etc (fechadura eletrônica)
3. Sistema de monitoramento da qualidade do ar com display LCD.

Lista dos componentes a serem utilizados no trabalho:

Componentes	Tipos e padronizações:
Resistor SMD	0805 ou 0603
Diodo SMD	SOD-123, SOD-323, SOT-23
Transistor SMD	SOT-23, SOT-223, SOT-323
Led SMD	0603, 0805, 1206
Capacitor SMD	0402, 0603, 0805
Conector (BORNE)	KF301, KF301"X"
Relê (Comum de 5 terminais)	SRD-05
Pin Macho	A2541WV, A2541WV-"X"
Push Button	TS6655TP
Opto acoplador PC-817	PC817
ESP12E	ESP-12E, ESP-12E_C89297
ATMEGA 328	ATMEGA328 DIP, ATMEGA328

EASYEDA

O EasyEDA é uma plataforma online que oferece uma variedade de ferramentas para o design de circuitos eletrônicos, desde a criação de esquemáticos até o design de PCB e a simulação de circuitos. Com uma interface intuitiva e acessível, o EasyEDA é adequado tanto para iniciantes quanto para profissionais da área de eletrônica.

Características principais do EasyEDA:

Esquemáticos: O EasyEDA permite criar esquemáticos de circuitos eletrônicos de forma fácil e rápida. Com uma biblioteca extensa de componentes, é possível selecionar os dispositivos desejados e conectá-los com facilidade.

Design de PCB: Uma das características mais poderosas do EasyEDA é a capacidade de projetar PCBs. Após a criação do esquemático, é possível migrar para o ambiente de design de PCB e posicionar os componentes no layout da placa. O EasyEDA oferece recursos avançados de roteamento automático e manual para garantir conexões elétricas adequadas.

Simulação de circuitos: O EasyEDA possui uma ferramenta de simulação integrada, permitindo aos usuários testar o comportamento do circuito eletrônico antes de fabricar o PCB. É possível simular o desempenho de componentes eletrônicos, verificar a resposta em frequência, analisar sinais digitais e muito mais.

Colaboração e compartilhamento: O EasyEDA é uma plataforma colaborativa, que permite compartilhar projetos com outros usuários. Essa funcionalidade é especialmente útil para trabalhos em equipe ou para receber feedback de outros especialistas. Também é possível importar e exportar projetos em diferentes formatos, facilitando a colaboração com outras ferramentas de design.

Biblioteca de componentes: O EasyEDA oferece uma ampla biblioteca de componentes eletrônicos, incluindo microcontroladores, sensores, atuadores, componentes passivos e muito mais. Essa biblioteca é constantemente atualizada e ampliada, facilitando a seleção e uso dos componentes corretos nos projetos.

Integração com serviços de fabricação: O EasyEDA possui integração com diversos serviços de fabricação de PCB, permitindo que os usuários convertam seus designs em arquivos Gerber prontos para produção. Essa integração simplifica o processo de fabricação do PCB e permite a criação de protótipos de forma mais rápida e eficiente.

Acesso online e multiplataforma: O EasyEDA é uma plataforma baseada em nuvem, o que significa que os usuários podem acessá-la a partir de qualquer dispositivo com conexão à internet.

Além disso, o EasyEDA é compatível com diferentes sistemas operacionais, incluindo Windows, macOS e Linux.

Essas características tornam o EasyEDA uma ferramenta abrangente e conveniente para o desenvolvimento de sistemas embarcados. Com sua interface amigável e recursos avançados, o EasyEDA é uma escolha popular para profissionais e entusiastas da eletrônica que desejam projetar e desenvolver sistemas embarcados de alta qualidade.

Link de acesso: <https://easyeda.com/pt>

Desenvolvimento:

PARTE 1 Link do vídeo: <https://youtu.be/8E4qu44FJ58>

PARTE 2 Link do vídeo: <https://youtu.be/DsxXDfKuAvA>

Link complementar: <https://www.youtube.com/watch?v=pU0XS0tD6vM>

Exemplo:

1. Introdução

A automação residencial tem ganhado destaque nas últimas décadas devido aos avanços na tecnologia e ao crescimento da Internet das Coisas (IoT). Casas inteligentes, ou "smart homes", são aquelas que integram dispositivos tecnológicos que podem ser controlados remotamente, proporcionando conveniência, segurança e eficiência energética aos moradores. O projeto de um sistema de acionamento remoto visa facilitar o controle de aparelhos elétricos, como luzes, ventiladores ou outros dispositivos, utilizando aplicativos móveis ou assistentes de voz. Isso permite aos usuários operar seus dispositivos com facilidade, mesmo à distância, melhorando a experiência de uso e otimizando o consumo de energia.

2. Objetivo do Sistema Embarcado

O objetivo deste sistema embarcado é desenvolver uma solução de automação residencial que permita o acionamento e desligamento remoto de dispositivos elétricos. A

operação será realizada por meio de um aplicativo de smartphone ou via comandos de voz, usando plataformas populares como Amazon Alexa ou Google Assistant. O sistema será capaz de integrar dispositivos residenciais comuns, proporcionando controle total ao usuário e possibilitando monitoramento em tempo real. O propósito é trazer maior comodidade, segurança e eficiência energética, automatizando o ambiente residencial de maneira acessível e escalável.

3. Lista de Componentes

A lista de componentes necessários para o desenvolvimento do sistema de acionamento remoto inclui:

Microcontrolador (ex: ESP32 ou ESP8266): O coração do sistema, responsável por processar os comandos recebidos via Wi-Fi ou Bluetooth e acionar os dispositivos conectados.

Módulo de comunicação Wi-Fi/Bluetooth: Integrado ao microcontrolador (como no ESP32 ou ESP8266), permite a comunicação com o aplicativo ou assistente de voz.

Relés: Utilizados para acionar dispositivos de alta tensão (110V/220V) com segurança, a partir dos sinais de baixa tensão gerados pelo microcontrolador.

Fonte de alimentação: Fornece a energia necessária para o funcionamento do microcontrolador e dos periféricos.

Sensor de corrente (opcional): Para monitoramento do consumo de energia em dispositivos conectados.

LEDs indicadores: Para mostrar o status de conexão e o estado dos dispositivos (ligado/desligado).

Resistores e capacitores: Componentes passivos usados para ajuste de circuitos, filtragem e proteção.

Placa de Circuito Impresso (PCB): Para montagem e organização dos componentes de forma compacta e segura.

Módulo de interface com aplicativo (software): Para criar uma interface amigável no smartphone ou integrar com serviços de comando de voz.

Componentes	Tipos e padronizações:
Resistor SMD	0805 ou 0603
Diodo SMD	SOD-123, SOD-323, SOT-23

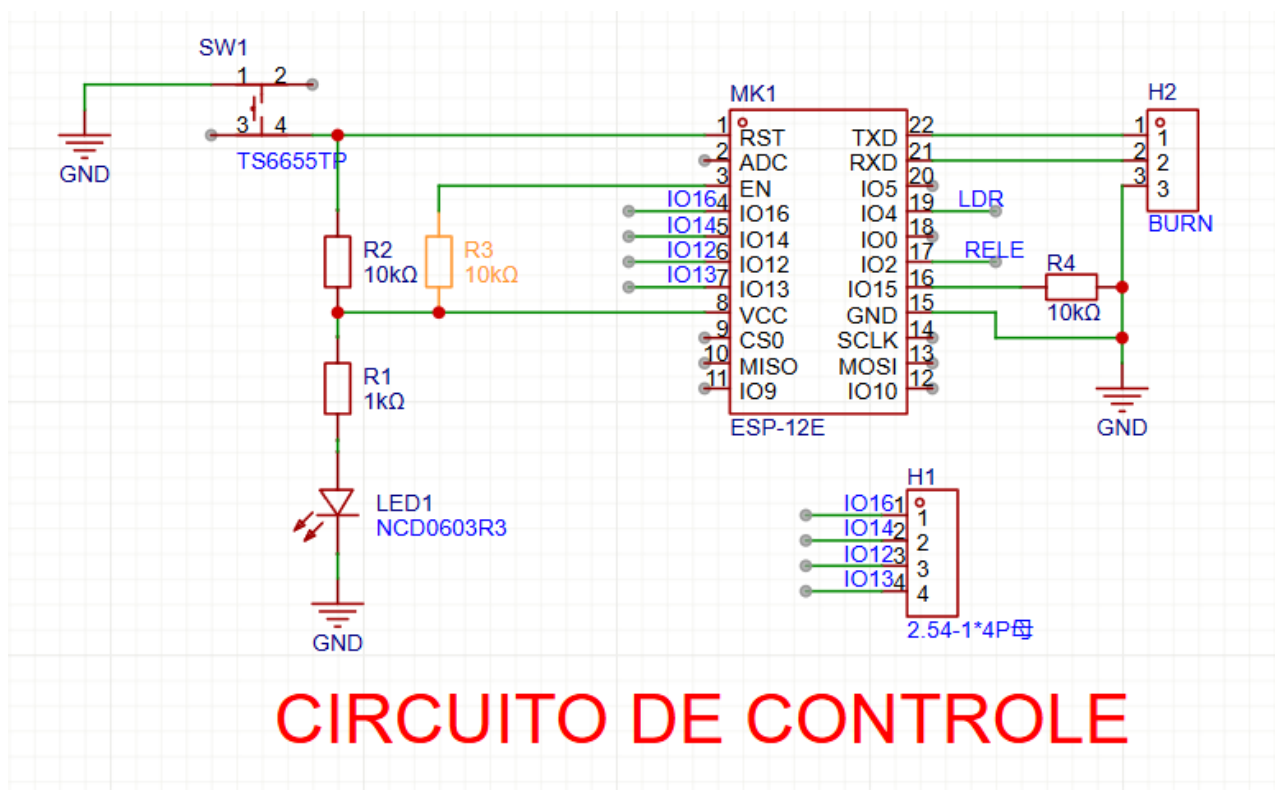
Transistor SMD	SOT-23, SOT-223, SOT-323
Led SMD	0603, 0805, 1206
Capacitor SMD	0402, 0603, 0805
Conector (BORNE)	KF301, KF301"X"
Relê (Comum de 5 terminais)	SRD-05
Pin Macho	A2541WV, A2541WV-"X"
Push Button	TS6655TP
Opto acoplador PC-817	PC817
ESP12E	ESP-12E, ESP-12E_C89297
ATMEGA 328	ATMEGA328 DIP, ATMEGA328

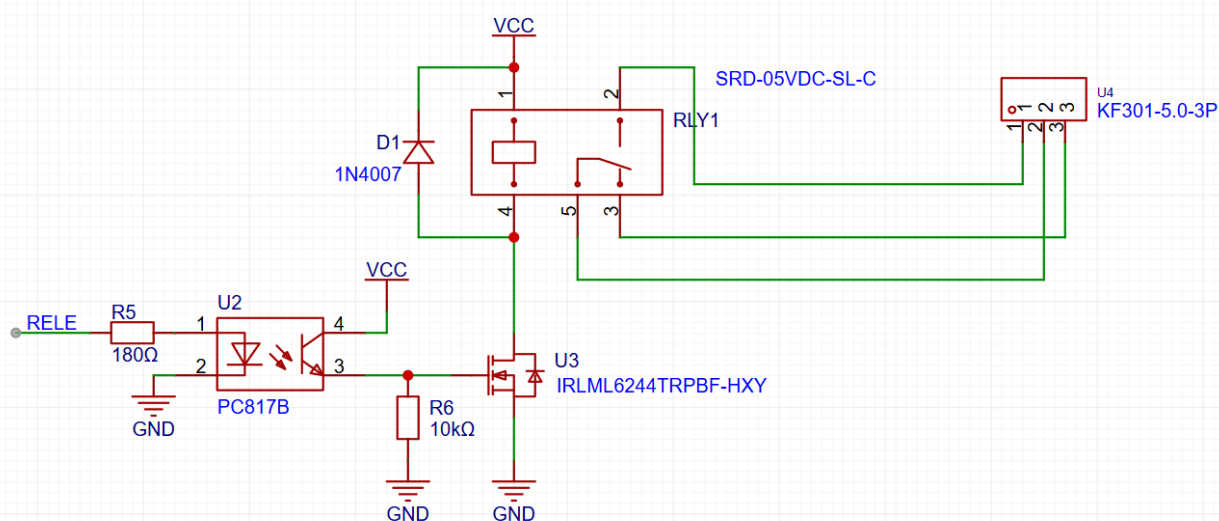
4. Esquema Elétrico:

Vídeo do desenvolvimento:

PARTE 1 Link do vídeo: <https://youtu.be/8E4qu44FJ58>

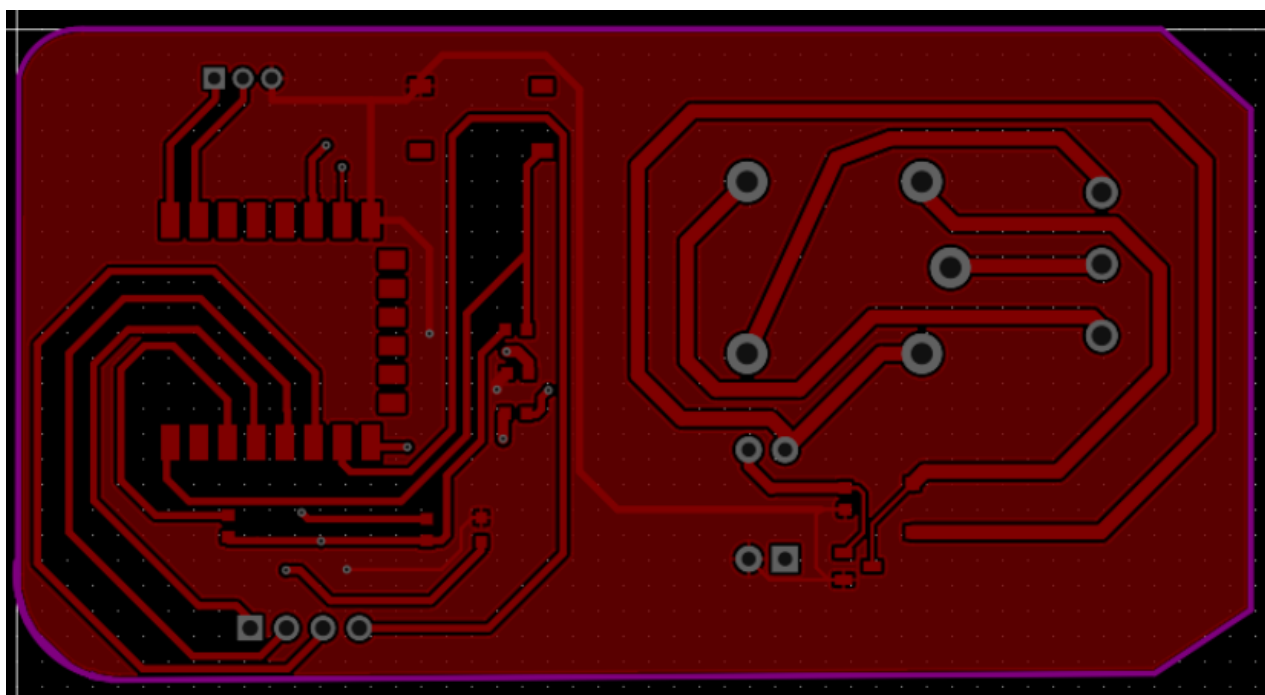
PARTE 2 Link do vídeo: <https://youtu.be/DsxXDfKuAvA>

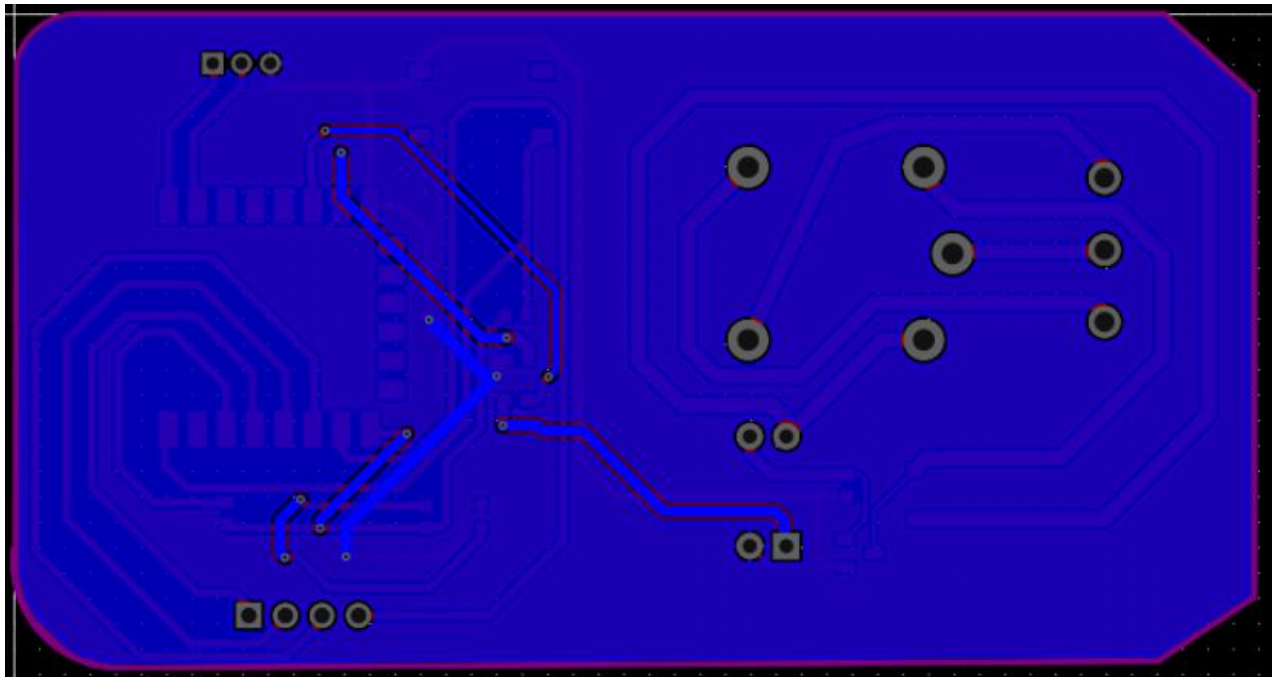




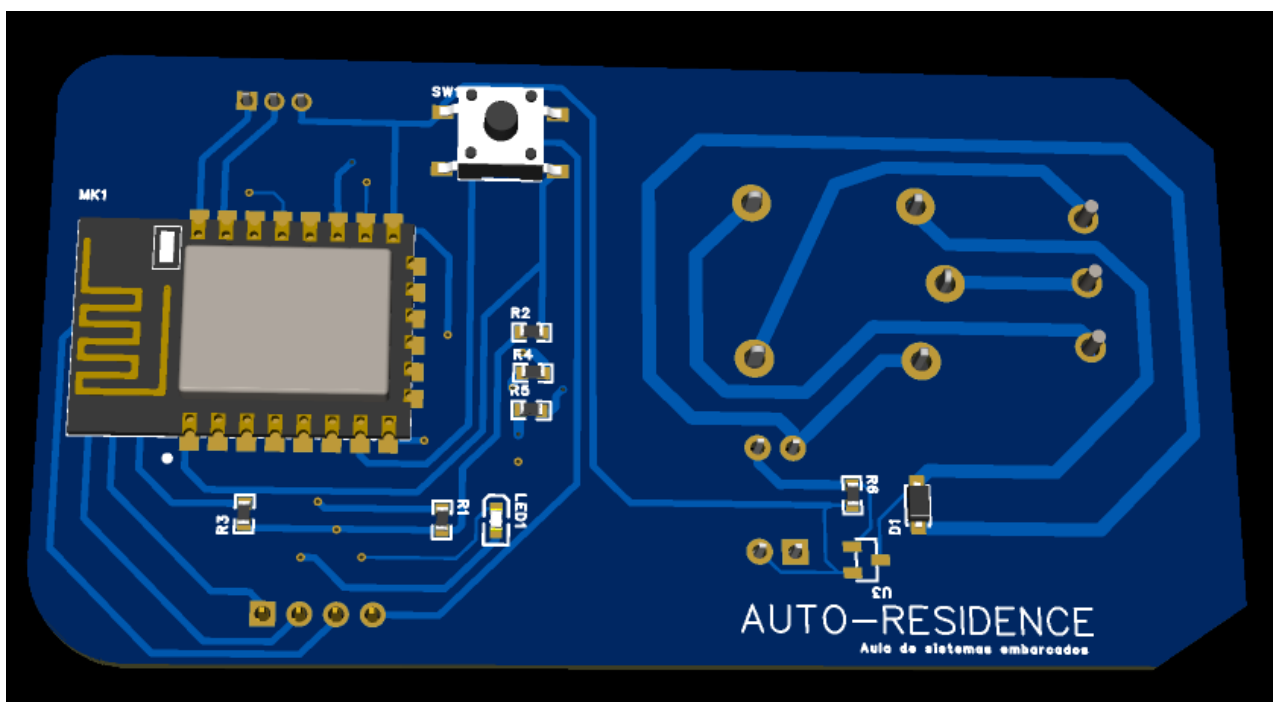
CIRCUITO DE ACIONAMENTO

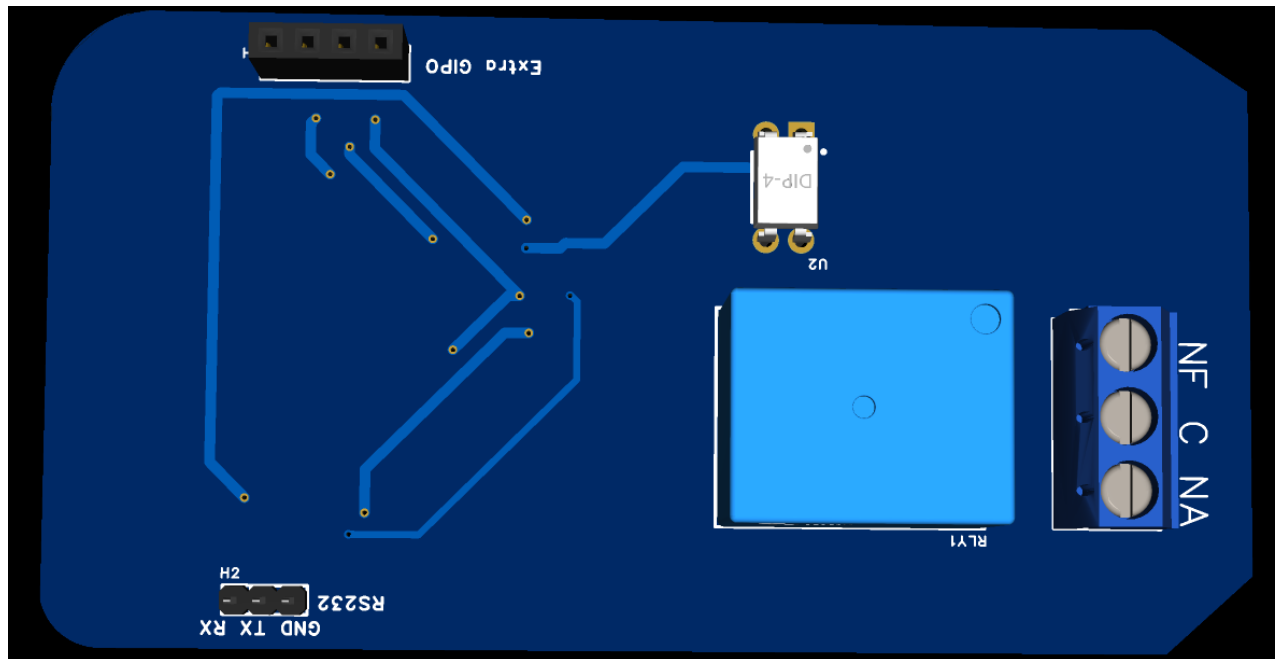
5. Layout PCB:





6. Visão 3D do sistema embarcado





7. Conclusão

O desenvolvimento de um sistema de acionamento remoto residencial por aplicativo ou comando de voz representa um importante passo na direção da automação e modernização das casas inteligentes. Utilizando tecnologias embarcadas e conectividade por meio de redes Wi-Fi ou Bluetooth, o projeto não só proporciona comodidade e praticidade aos usuários, como também promove o uso eficiente de energia e melhora a segurança residencial. Ao integrar componentes como microcontroladores, relés e sensores, o sistema se torna uma solução robusta e escalável para diferentes necessidades. Com a crescente popularidade da IoT e a constante evolução das tecnologias de comando de voz, este projeto se alinha às tendências modernas, oferecendo uma aplicação prática e com potencial para ser ampliada ou integrada com outras plataformas de automação.

8. Programação

```
#include <ESP8266WiFi.h>
#include <PubSubClient.h>

#define DEBUG

#define M1 D4    //pino de saida para acionamento da Lampada L1
#define L2 D1    //pino de saida para acionamento da Lampada L1

//informações da rede WIFI
const char* ssid = "2.4G";           //SSID da rede WIFI
const char* password = "@12345678";  //senha da rede wifi
```

```

    //informações do broker MQTT - Verifique as informações geradas
elo CloudMQTT
    const char* mqttServer = "hairdresser.cloudmqtt.com";
/server
    const char* mqttUser = "fwhrqdpmwd";           //user
    const char* mqttPassword = "NkSjYsF3L0gmr";     //password
    const int mqttPort = 15483;                     //port
    const char* mqttTopicSub = "casa/L";            //tópico que
era assinado
    const char* mqttTopicSub2 = "casa/L2";          //tópico que
era assinado

WiFiClient espClient;
PubSubClient client(espClient);

void setup() {

    Serial.begin(115200);
    pinMode(M1, OUTPUT);
    pinMode(L2, OUTPUT);
    WiFi.begin(ssid, password);

    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
#ifdef DEBUG
        Serial.println("Conectando ao WiFi..");
#endif
    }
#ifdef DEBUG
    Serial.println("Conectado na rede WiFi");
#endif

    client.setServer(mqttServer, mqttPort);
    client.setCallback(callback);

    while (!client.connected()) {
#ifdef DEBUG
        Serial.println("Conectando ao Broker MQTT...");
#endif

        if (client.connect("ESP8266Client", mqttUser, mqttPassword
) {
#ifdef DEBUG
            Serial.println("Conectado");
#endif

        } else {
#ifdef DEBUG
            Serial.print("falha estado ");
            Serial.print(client.state());
#endif
            delay(2000);

```

```

    }
}

//subscreve no tópico
client.subscribe(mqttTopicSub);
client.subscribe(mqttTopicSub2);
}

void callback(char* topic, byte* payload, unsigned int length)

    //armazena msg recebida em uma string
    payload[length] = '\0';
    String strMSG = String((char*)payload);

#ifdef DEBUG
    Serial.print("Mensagem chegou do tópico: ");
    Serial.println(topic);
    Serial.print("Mensagem:");
    Serial.print(strMSG);
    Serial.println();
    Serial.println("-----");
#endif

    //Converter String para inteiro

    int PWM = strMSG.toInt();
    //aciona saída conforme msg recebida
    digitalWrite(M1, PWM); //PWM Motor
    Serial.print(" PWM : ");
    Serial.println(PWM);

}

//função pra reconectar ao servido MQTT
void reconnect() {
    //Enquanto estiver desconectado
    while (!client.connected()) {
#ifdef DEBUG
        Serial.print("Tentando conectar ao servidor MQTT");
#endif

        bool conectado = strlen(mqttUser) > 0 ?
            client.connect("ESP8266Client",
mqttUser, mqttPassword) :
            client.connect("ESP8266Client");

        if (conectado) {
#ifdef DEBUG
            Serial.println("Conectado!");
#endif
            //subscreve no tópico
            client.subscribe(mqttTopicSub, 1); //nivel de
qualidade: QoS 1

```

```
        client.subscribe(mqttTopicSub2, 1); //nível de
qualidade: QoS 1
    } else {
#ifdef DEBUG
        Serial.println("Falha durante a conexão.Code: ");
        Serial.println( String(client.state()).c_str());
        Serial.println("Tentando novamente em 10 s");
#endif
        //Aguarda 10 segundos
        delay(10000);
    }
}

void loop() {
    if (!client.connected()) {
        reconnect();
    }
    client.loop();
}
```