



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO
PRÓ-REITORIA DE GRADUAÇÃO
DEPARTAMENTO DE ENGENHARIA E TECNOLOGIA
CURSO DE CIÊNCIAS E TECNOLOGIAS

FRANCISCO VICTOR DE ARAÚJO FONTENELE

**TÍTULO: CHAVE PARTIDA DIRETA COM REVERSÃO REMOTA ATRAVÉS DO
ARDUINO COM BLUETOOTH.**

MOSSORÓ

2025

FRANCISCO VICTOR DE ARAÚJO FONTENELE

**TÍTULO: CHAVE PARTIDA DIRETA COM REVERSÃO REMOTA ATRAVÉS DO
ARDUINO COM BLUETOOTH.**

Monografia apresentada a Universidade Federal Rural do Semi-Árido como requisito para obtenção do título de Bacharel em Ciências e Tecnologia.

Orientador: Nome Completo, Prof. Dr. Idalmir de Souza Queiroz Júnior.

MOSSORÓ

2025

© Todos os direitos estão reservados a Universidade Federal Rural do Semi-Árido. O conteúdo desta obra é de inteira responsabilidade do (a) autor (a), sendo o mesmo, passível de sanções administrativas ou penais, caso sejam infringidas as leis que regulamentam a Propriedade Intelectual, respectivamente, Patentes: Lei nº 9.279/1996 e Direitos Autorais: Lei nº 9.610/1998. O conteúdo desta obra tomar-se-á de domínio público após a data de defesa e homologação da sua respectiva ata. A mesma poderá servir de base literária para novas pesquisas, desde que a obra e seu (a) respectivo (a) autor (a) sejam devidamente citados e mencionados os seus créditos bibliográficos.

F683c Fontenele, Francisco Victor de Araújo.

Chave Partida Direta com Reversão Remota do
Arduino com Bluetooth. / Francisco Victor de
Araújo Fontenele. - 2025.
34 f. : il.

Orientador: Idalmir de Souza Queiroz Júnior.
Monografia (graduação) - Universidade Federal
Rural do Semi-árido, Curso de Ciência e
Tecnologia, 2025.

1. Arduino. 2. Motor. 3. Bluetooth. I. Júnior,
Idalmir de Souza Queiroz, orient. II. Título.

Ficha catalográfica elaborada por sistema gerador automático em conformidade
com AACR2 e os dados fornecidos pelo autor(a).

Biblioteca Campus Mossoró / Setor de Informação e Referência

Bibliotecária: Keina Cristina Santos Sousa e Silva

CRB: 15/120

FRANCISCO VICTOR DE ARAÚJO FONTENELE

**TÍTULO: CHAVE PARTIDA DIRETA COM REVERSÃO REMOTA ATRAVÉS DO
ARDUINO COM BLUETOOTH.**

Monografia apresentada a Universidade
Federal Rural do Semi-Árido como requisito
para obtenção do título de Bacharel em
Ciências e Tecnologias

Defendida em: 04 /08 / 2025.

BANCA EXAMINADORA

Idalmir de Souza Queiroz Júnior.
Nome do Orientador, Prof. Dr. (UFERSA)
Presidente

Elias Samuel Soares Cezário
Nome do Examinador Interno, Prof. (UFERSA)
Membro Examinador

Francisco Magno Monteiro Sobrinho
Nome do Examinador Externo, Prof. Dr. (CET, SENAI-RN)
Membro Examinador

AGRADECIMENTOS

Agradeço primeiramente a Deus por essa grande oportunidade, de me permitir fazer parte desta universidade, por me fazer chegar até aqui, creio que do começo ao fim, ele esteve presente e por não deixar faltar nada desde o dia que resolvi, vir para esta universidade por me fazer conhecer pessoas, professores e projetos na UFRSA e por me permitir entregar esse projeto.

Agradeço minha mãe Ana Glivia de Araújo e meu padrasto Daniel Roque Pessoa, pelo apoio desde sempre, pelas palavras de animo, por nunca me deixarem desistir dessa caminhada, foram aqueles que mesmo longe, eu sinto que estavam perto sempre orando por mim, trazendo palavras de ânimo e de incentivo, só tenho que agradecer a eles, e dizer que Deus pague a eles tudo.

Agradeço também minha namorada Alecsandra, pelo apoio mesmo de longe, com suas palavras de verdade e de sabedoria, pelo amor que só ela sabe proporcionar e por sempre mostrar o jeito certo de fazer e de nunca deixar Deus por nada.

Agradeço ao Orientador Professor Dr. Idalmir de Souza Queiroz Júnior pela paciência, pela oportunidade de ter sido meu orientador, pela ajuda nas horas vagas, pelos ensinamentos, algumas coisas eu não sabia, obrigador por ter me ajudado mesmo com meu déficit em alguns assuntos.

Agradeço a Banca Examinadora por dedicar seu tempo para avaliar este trabalho, agradeço a oportunidade de poder apresentar esta monografia para o encerramento deste curso.

Agradeço aos meus Amigos do Embarcação (Valton, Lay, Marcela, Gustavo, Catarina), pelo apoio de todos os encontros, sempre tentando ajudar como podiam e por nunca me deixar desanimado e desistir, agradeço a Isaac Kevin, meu amigo de TCC que esteve ai nessa carreira junto comigo, agradeço também as minhas amigas: Marina, Clea e Débora, de alguma forma, sempre presentes perguntando se estava bem e pela calma e tranquilidade que vocês souberam me transmitir: “vai dar certo”, a frase que mais ouvi esse semestre.

É uma disciplina que promove, com visão integrada, o gerenciamento e o compartilhamento de todo o ativo de informação possuído pela empresa. Esta informação pode estar em um banco de dados, documentos, procedimentos, bem como em pessoas, através de suas experiências e habilidades.

Gartner Group

RESUMO

Atualmente a automação e o uso de um controle remoto de dispositivos eletrônicos vêm crescendo muito no mundo no geral, a ideia de realizar uma determinada tarefa a distância sem estar presente ou de forma manual é algo cada vez mais presente no cotidiano, um exemplo bem pratico é o uso de acionamentos de motores para portas eletrônicas ou portões elétricos. O presente TCC aborda a aplicação do uso de uma Arduino com um modulo *Bluetooth*, HC-05, com o objetivo de realizar um acionamento de partida direta com reversão com um motor de pequeno porte, através do uso do aplicativo *Bluetooth Electronics* que controla o motor de forma remota de um dispositivo móvel, mostrando que pode ser acessível e simples para uso domiciliar ou industrial de acordo com o motor que for ser utilizado, monofásico ou trifásico, assim sendo uma forma de automação simples e de baixo custo. Este projeto vem demonstrar que é possível ter um controle remoto de forma automatizada da ligação de um motor por um smartphone.

Palavras-chave: Automação, Arduino, HC-05, Motor, Baixo custo.

ABSTRACT

Automation and the use of remote control for electronic devices are currently growing rapidly worldwide. The idea of performing a specific task remotely, without being present or manually, is increasingly present in everyday life. A very practical example is the use of motor drives for electronic doors or electric gates. This final project addresses the application of an Arduino with a Bluetooth module, HC-05, to perform a direct-reverse start drive with a small motor. This application uses the Bluetooth Electronics app to remotely control the motor from a mobile device. This demonstrates that it can be affordable and simple for home or industrial use, depending on the motor being used—single-phase or three-phase—and thus constitutes a simple and low-cost form of automation. This project demonstrates the possibility of automated remote control of a motor's connection via a smartphone.

Keywords: Automation, Arduino, HC-05, Motor, Low Cost.

LISTA DE FIGURAS

Figura 1	–	Componentes Motor Monofásico	13
Figura 2	–	Vista em corte Motor Trifásico	14
Figura 3	–	Arduino UNO, pinagem do Arduino	16
Figura 4	–	Relé de 1 canal 5V, Relé de 4 canais 5V	17
Figura 5	–	Potenciômetro	18
Figura 6	–	Modulo Bluetooth HC-05	19
Figura 7	–	Motor monofásico, Motor trifásico	20
Figura 8	–	Diagrama Multifilar partida direta	21
Figura 9	–	Diagrama multifilar reversão	21
Figura 10	–	Bluetooth Electronics, Pannel	24
Figura 11	–	Bluetooth Electronics, Botões	24
Figura 12	–	Motor Trifásico ligado a relé	25

Sumário

1. INTRODUÇÃO	11
2. FUNDAMENTOS TEÓRICOS.....	12
2.1 Arduino	12
2.2 Motor	12
2.2.1 Motor monofásico	13
2.2.2 Motor Trifásico	14
2.2.3 Acionamento	15
3. MATERIAIS E MÉTODOS	15
3.1 Materiais	15
3.1.1 Arduino UNO.....	16
3.1.2 Módulo Relé.....	16
3.1.3 Potenciômetro	17
3.1.4 Sensor Modulo Bluetooth.....	18
3.1.5 Motores.....	19
3.2 Montagem do projeto.....	20
3.3 Código.....	22
3.4 <i>Bluetooth Electronics</i>	23
4. RESULTADOS	24
5. CONCLUSÃO	26
REFERÊNCIAS	27
APÊNDICE A – CODIGO FONTE.....	30

1. INTRODUÇÃO

Com o aumento do desenvolvimento tecnológico tem impulsionado a demanda por automação e do controle e do monitoramento de equipamentos nas grandes indústrias. Atualmente, muitos desses processos são realizados manualmente, o que resulta na falta de um controle preciso e, conseqüentemente, em ineficiências, diante disso a procura por sistemas automatizados se intensifica, o que contribui diretamente para a consolidação da Indústria 4.0.

A Indústria 4.0, que teve esse termo usado pela primeira vez em 2011, que de acordo com Silveira (2017) diz que existem 7 pilares, sendo eles: Internet das Coisas, Segurança cibernética, *Big Data Analytics*, Computação em nuvem, Robótica avançada, Inteligência Artificial, Novos materiais, sendo essas as tecnologias mais relevantes para a implementação e funcionamento da Indústria 4.0, um uso muito visto é o da Internet das coisas. A relação entre coisas (produtos, serviços, lugares) e pessoas por meio das plataformas e das tecnologias que podem se conectar tanto perto de uma pessoa como muito longe é visto, na visão industrial, uma forma de otimizar a produção.

Complementando o conceito da Internet das Coisas, o *smartphone* emergiu como um avanço tecnológico de grande relevância. Enquanto seu uso para soluções cotidianas já é uma realidade para a população, a sua aplicação no ambiente industrial está em plena ascensão. Essa crescente popularidade se deve à constante difusão de ferramentas e aplicativos práticos, que permitem maior controle e monitoramento de processos. (ARAÚJO, 2025)

Dessa forma, o objetivo geral deste trabalho é desenvolver um sistema de controle e automação para motores de pequeno porte, utilizando um microcontrolador Arduino e um módulo *Bluetooth*, com o intuito de permitir a operação remota por meio de um dispositivo móvel. Para a concretização deste objetivo, serão realizados o projeto e a montagem do circuito eletrônico, que integrará os componentes principais. Em seguida, a programação do microcontrolador, permitindo o controle do motor através de comandos enviados via *Bluetooth*. Além do uso de um aplicativo em um dispositivo móvel. Por fim, o sistema será validado por meio de testes práticos, e a viabilidade econômica e de baixo custo.

2. FUNDAMENTOS TEÓRICOS

2.1 Arduino

O Arduino teve seu surgimento em 2005, na Itália, com um professor chamado Massimo Banzi e David Cuartielles, que ensinavam eletrônica e programação de computadores a seus alunos de *design*, para que eles usassem em seus projetos de arte, interatividade e robótica. Porém, ensinar eletrônica e programação para pessoas que não são da área não era uma tarefa simples, então se juntaram junto com um dos alunos de Massimo, David Mellis, que criou a linguagem de programação do Arduino. (RIOS et al, 2012, p. 3).

O Arduino é uma pequena placa de microcontrolador que contém uma conexão USB, sendo possível a ligação com um computador. Contendo diversos terminais que permitem a conexão com dispositivos externos, como motores, reles, sensores luminosos, diodos a laser, alto-falantes, microfones e etc, podendo ser controlado diretamente pelo computador ou pode trabalhar de forma autônoma. (MONK, 2014, P.2). Ele tem entradas de dados (IN) e saída de dados (OUT) com um oscilador de cristal de 16 Mhz, um regulador de tensão de 5 V, botão de reiniciar, plugue de alimentação, pinos conectores, e alguns LEDs para facilitar a verificação de seu funcionamento (RIOS et al, 2012, p. 3).

2.2 Motor

Os motores elétricos são máquinas que convertem energia elétrica em energia mecânica, cujo funcionamento se baseia na Lei de Faraday. Dentre os tipos mais empregados na indústria, destaca-se o motor de indução trifásico, que é amplamente utilizado devido ao seu baixo custo e fácil manutenção.

Segundo a Empresa de Pesquisa Energética (EPE), as indústrias brasileiras consomem 30% da energia elétrica gerada e são responsáveis por quase 40% da eletricidade consumida no país. Essa elevada demanda é impulsionada por equipamentos como bombas, motores, compressores, refrigeradores, sistemas de iluminação e ar-condicionado. Nesse contexto, o impacto dos motores elétricos é ainda mais significativo: o Programa Nacional de Conservação de Energia Elétrica (Procel) aponta que eles consomem 68% da energia elétrica das fábricas. (MOTORES,2024)

2.2.1 Motor monofásico

Motores elétricos monofásicos são dispositivos que transformam energia elétrica em energia mecânica usando uma única fase de corrente alternada. Eles consistem basicamente em um estator fixo, que gera um campo magnético rotativo, e um rotor, que gira dentro deste campo como é mostrado na Figura 1 com todos os seus componentes. A interação entre os campos magnéticos do estator e do rotor cria o movimento rotativo que é utilizado para realizar trabalhos mecânicos. Eles são o contrário dos motores trifásicos, que utilizam três fases de corrente para criar um campo magnético com o desempenho mais uniforme e eficiente, os motores monofásicos dependem de uma única fase, o que pode resultar em desempenho menos suave, por não gerar torque de partida, ele depende de um capacitor. (MOTORES, 2024)

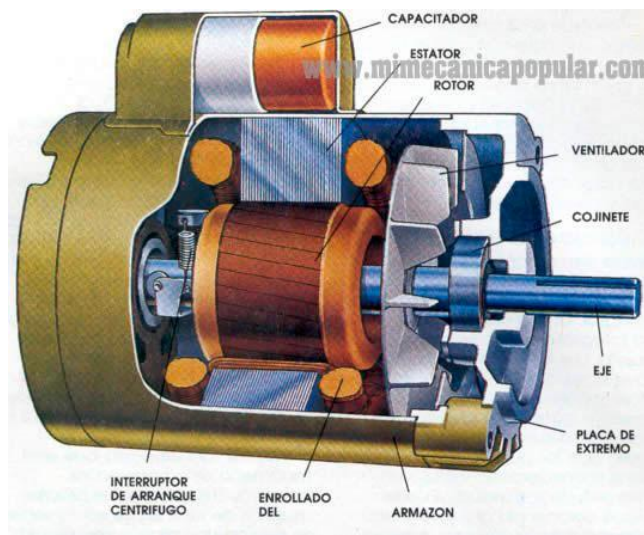


Figura 1 – Componentes Motor Monofásico
Fonte:MANUTENÇÃO, (2025)

Esses motores são fundamentais em muitas aplicações, desde eletrodomésticos a equipamentos industriais leves, por serem mais fáceis de instalar e muito mais econômicos em sistemas onde a energia trifásica não está disponível.

2.2.2 Motor Trifásico

Nas indústrias o Motor trifásico é um dos mais utilizados por ser durável e sua facilidade de ligação e de controle ele é um tipo de motor elétrico, de indução, constituído por estator, rotor, carcaça, tampa, ventilador, eixo, enrolamento trifásico, caixa de ligação, terminais, rolamentos como é mostrado na Figura 2. Ele recebe esse nome devido ao tipo de alimentação que recebe que se refere à quantidade de fases que a instalação elétrica precisa fornecer para o seu funcionamento. Enquanto o motor monofásico é mais utilizado em residências, o trifásico é mais popular na indústria, onde pode ser aplicado no acionamento de bombas, elevadores, ventiladores, entre diversos outros equipamentos. (CRAVO,2022)

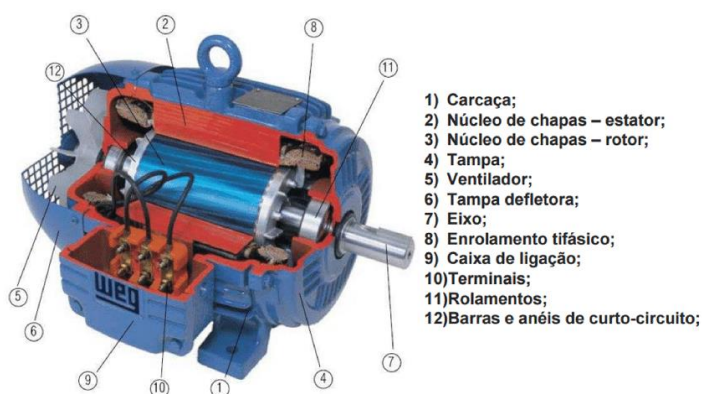


Figura 2 – Vista em corte Motor Trifásico
 Fonte: SANTOS, (2016)

Esse tipo de motor é encontrado em uma grande variedade de potências, dadas em quilowatts ou cavalos vapor. Comparado a um modelo monofásico de mesma potência, o trifásico tem manuseio mais simples e é menor. O motor trifásico também é chamado como motor de corrente alternada (AC), motor assíncrono trifásico, atuador trifásico, entre outras denominações. (CRAVO,2022)

2.2.3 Acionamento

O acionamento de motores pode ser realizado de diversas formas, que variam em complexidade e custo. De acordo com Praxedes (2011), as soluções podem ser simples e de baixo custo ou sofisticadas e de alto custo. Os sistemas de acionamento podem ser classificados, por exemplo, como convencionais ou eletromecânicos.

Cada um desses sistemas possui vantagens e desvantagens. Enquanto os sistemas mais simples, por exemplo, não são capazes de fornecer um controle preciso sobre a corrente e outros parâmetros do motor, os sistemas mais avançados exigem mão de obra especializada e qualificada e têm um custo mais elevado, porém, oferecem um controle preciso sobre a corrente. Os sistemas mais convencionais e conhecidos são:

- Partida Direta;
- Partida Estrela-Triângulo;
- Partida com Auto-trafo Compensador.

Já para os sistemas avançados ou Eletrônicos:

- Partida com chave Soft-Starter;
- Partida com Inversor de Frequência.

No presente projeto o acionamento utilizado é o partida direta sendo um método de acionamento de corrente alternada, onde o motor é conectado diretamente a rede elétrica. Em outras palavras, ela se dá quando aplicamos a tensão nominal sobre os enrolamentos do estator do motor, de maneira direta. (SEGUNDO; RODRIGUES.2015)

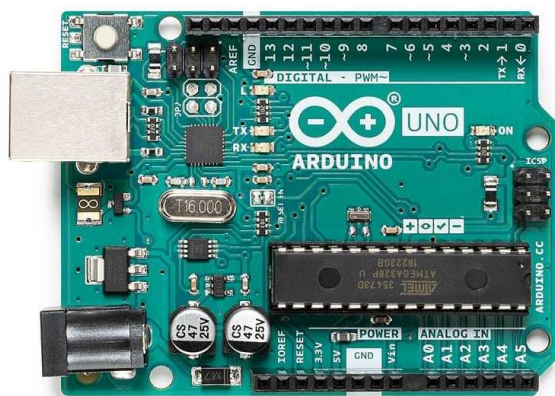
3. MATERIAIS E MÉTODOS

3.1 Materiais

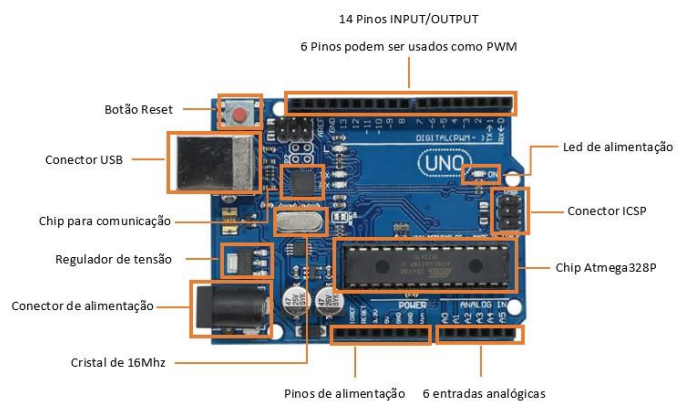
Para o presente projeto, foram usados materiais do Laboratório de engenharia 1 e disponibilizados pelo Professor orientador.

3.1.1 Arduino UNO

A placa utilizada é a do Arduino UNO como mostra a Figura 3, bem como a sua pinagem, que de acordo com o site oficial do Arduino(2025), é um uma placa com o micro controlador Atmega328, possuindo 14 pinos de entrada/saída digitais 6 entradas analógicas, podendo 6 serem usadas como saída , um ressonador cerâmico de 16 MHz, uma conexão USB, um conector de alimentação, um conector ICSP e um botão de reset.



(a)



(b)

Figura 3 – Arduino Uno(a), pinagem do Arduino UNO(b).
Fonte: ARDUINO, (2025)

3.1.2 Módulo Relé

Utilizando também o Relé, que é um dispositivo eletromecânico capaz de controlar circuitos de grandes tensões e correntes a partir de pequenas tensões e correntes de outro circuito. O funcionamento de um relé se dá pela seguinte forma: quando o circuito de controle, com pequenas tensões e correntes, envia uma corrente para a bobina do relé, que gera um campo magnético que atrai o contato, fechando o circuito de carga. Para o controle do sistema, será utilizado o relé, que é um dispositivo eletromecânico capaz de controlar circuitos de alta tensão e corrente a partir de sinais de baixa tensão e corrente. O seu princípio de funcionamento baseia-se na criação de um campo magnético: quando uma corrente elétrica de baixa intensidade é enviada para a bobina do relé, ela gera um campo magnético que atrai

o contato móvel, fechando o circuito de carga e permitindo a passagem de uma corrente maior. (MAIA,2012)

Nesse projeto, foi utilizado módulos reles, que são reles que podem ser integrados ao Arduino com mais facilidade, tendo sido utilizado 3 módulos relé, 2 com apenas 1 canal de 5V para fazer a partida direta, e 1 modulo com 4 reles para fazer a reversão mostrado na Figura 4.



(a)

(b)

Figura 4 – Relé de 1 canal 5V (a), Relé de 4 canais 5V (b).

Fonte: ELETROGATE, (2025)

3.1.3 Potenciômetro

Tem também o potenciômetro que é definido como sendo um resistor de tensão variável. Ele é composto por uma faixa de material resistivo (geralmente grafite) ligada entre seus dois terminais externos. Nesse material, desliza um cursor, ligado diretamente ao terminal central do potenciômetro. Esse cursor pode ser movimentado através de um eixo rotativo ou um pino de plástico ou metal. Quando alterada a posição do cursor, altera a resistência entre o terminal central e os dois terminais externos do potenciômetro. (PATSKO, 2006)

O potenciômetro usado contém 3 terminais que foi colocado na protoboard e conectado ao Arduino, um para ser colocado no GND do Arduino, outro no VCC e outro nas portas analógicas, a porta escolhida foi a A0 ele será importante nesse projeto, podendo ser feito um *timer* que será importante na hora de trocar para reversão.

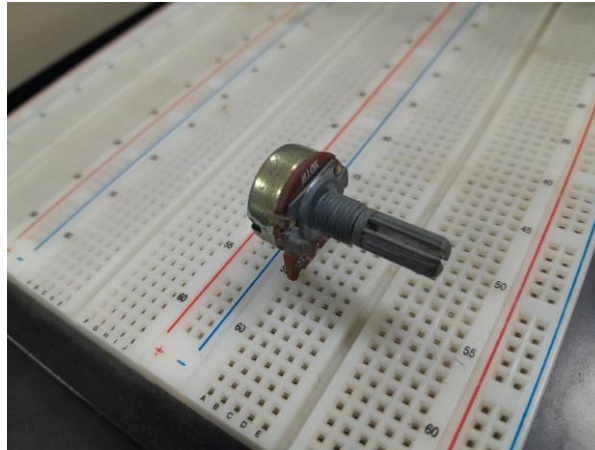


Figura 5 – Potenciômetro.
Fonte: O autor, (2025)

3.1.4 Sensor Modulo Bluetooth

O Módulo *Bluetooth* como é mostrado na Figura 6, foi utilizado por ser de baixo custo e por também não consumir dados de internet, mas ele possibilita transmitir e receber dados através de comunicação sem fio. De acordo com Silva (2022), ele atua em modo mestre (master) que é estabelecer conexões com outros dispositivos e envia informações, em modo escravo (slave), que apenas recebe as informações enviadas pelo mestre e em modo *loopback*, que recebe informações do mestre e depois as retorna de volta. Possuindo seis pinos: VCC, GND, TXD, RXD, Key ou EN e State

VCC é a alimentação de 3,6 a 6V e GND a referência que deve ser a mesma do Arduino. Os pinos RXD e TXD são de recepção e transmissão de dados, o Rx do módulo deve ser ligado ao Tx do Arduino e vice-versa. O pino State indica se o módulo está pareado ou não, e o pino EN (enable) é utilizado para habilitar comandos AT do módulo. Abaixo estão as especificações do HC-05:(SILVA, 2022)

- Tensão de operação: 3,6V – 6VDC
- Frequência de operação: 2,4GHz
- Nível de sinal lógico: 3,3V
- Protocolo Bluetooth: v2.0+EDR
- Banda: ISM

- Modulação: GFSK
- Segurança: autenticação e criptografia 24
- Modo de funcionamento: master / slave
- Temperatura de operação: $-40^{\circ} \sim 105^{\circ}\text{C}$
- Alcance do sinal: $\sim 10\text{m}$
- Senha padrão (PIN): 1234 ou 0000

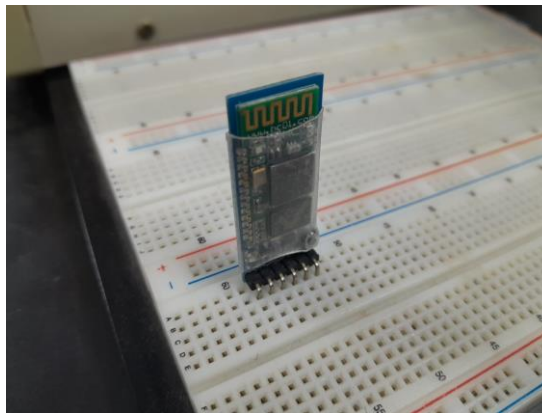


Figura 6 –Modulo Bluetooth HC-05.
Fonte: O autor, (2025)

3.1.5 Motores

Os motores utilizados foram os motores Monofásico e o Trifásico, como é mostrado na Figura 7. Os motores utilizados, tem modificações nas caixas de chave de partida, feitas pelo orientador, onde estão saindo fios para serem conectados nos relés para o projeto, o monofásico com 4 fios, para ligar o motor e desligar. No motor trifásico que é onde ocorre a reversão está saindo 6 fios, onde 2 ficam para alimentação, 2 para a partida normal e 2 para a reversão.



(a)



(b)

Figura 7 – Motor monofásico(a), Motor trifásico(b).

Fonte: O autor, (2025)

3.2 Montagem do projeto

Com a aquisição de todos os componentes listados, iniciou-se a fase de implementação do projeto. Para garantir um domínio completo sobre a eletrônica do circuito antes de acionar o motor, optou-se, sob orientação do professor, por realizar testes preliminares com LEDs. Essa etapa permitiu validar o esquema de ligações do Arduino com a protoboard e o relé, servindo como base para a montagem final. Inicialmente, o projeto contava com apenas um relé, mas a necessidade de dois relés foi identificada para controlar o acionamento de dois LEDs. O módulo *Bluetooth* HC-05 foi utilizado como a principal interface de comunicação remota em todas as etapas de teste e, também, no controle do motor. Os comandos enviados ao HC-05 foram definidos pelas variáveis 1 (ligar), 2 (desligar) e 0 (reversão).

Para a validação do código e das conexões com o relé, foi proposto, como etapa intermediária, o acionamento de uma lâmpada via *Bluetooth*, replicando o mesmo controle utilizado para os LEDs. A principal diferença nesse teste foi a transição da alimentação do circuito: em vez da tensão de 5V fornecida pelo Arduino, a lâmpada foi conectada a uma fonte de tensão mais alta, simulando, de forma mais precisa, as condições de operação do motor. Esse procedimento serviu como uma importante medida de segurança, reduzindo o risco de acidentes durante a fase de implementação e acionamento do motor.

O acionamento do motor monofásico foi realizado por meio de uma partida direta, cujo diagrama multifilar, similar ao da Figura 8, sendo apenas uma imagem que demonstra como foi montado pelo orientador, foi fornecido pelo professor orientador. Embora o diagrama original fosse para um motor trifásico, as orientações propostas permitiram sua adaptação para o motor monofásico. A lógica de programação foi mantida de forma similar ao acionamento da lâmpada, o que agilizou a implementação. Do contator do motor monofásico, quatro fios foram conectados aos relés, sendo um configurado como normalmente aberto (NA) e o outro como normalmente fechado (NF). O relé NA era responsável por ligar o motor, e o NF por desligá-lo. Para garantir a segurança durante a reversão da direção do motor, foi utilizada uma abordagem diferente. Em vez de um *delay* programado, optou-se pelo uso de um potenciômetro. No contexto do código, o potenciômetro atuou como um "*timer*", controlando o tempo de operação do motor antes de permitir a mudança de direção, o que previne trocas de sentido abruptas e que possa danificar o sistema.

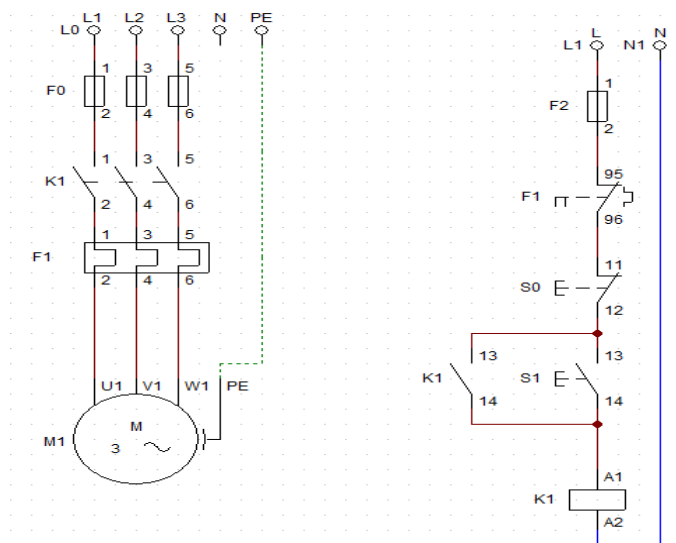


Figura 8 – Diagrama Multifilar partida direta trifásica.
Fonte: O autor, (2025)

orientador para representar de maneira similar como foi feito, sendo representada pelo ‘0’ no código.

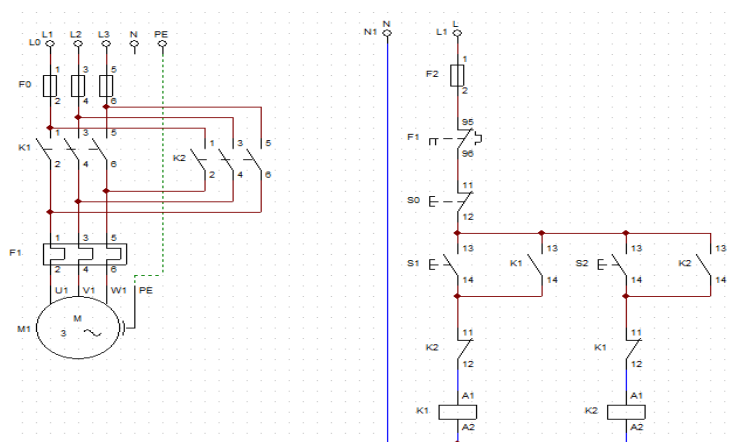


Figura 9 – Diagrama multifilar reversão.

Fonte: O autor, (2025)

A partida com reversão foi realizada apenas para o motor trifásico, pois bastaria inverter duas fases, enquanto que no motor monofásico necessitaria inverter as conexões da bobina de partida sendo necessária uma chave de partida um pouco mais complexa. Para ter uma segurança maior na hora de realizar a reversão, foi incluído um atraso (*DELAY*) com o objetivo de esperar o motor parar antes de permitir a reversão. Foi proposto um potenciômetro para regular o tempo de reversão do motor, evitando uma troca muito rápida no sistema.

3.3 Código

O código, que se encontra no Apêndice A, foi feito na IDE do Arduino. No código é apresentado a biblioteca *softwareserial.h*, que de acordo com site do próprio Arduino, nada mais é do que uma biblioteca que permite a comunicação serial em outros pinos digitais de uma placa Arduino, utilizando *software* para replicar a funcionalidade. Através dessa biblioteca é possível utilizar o módulo *Bluetooth* pelas portas 10 e 11 (RX e TX).

No IDE do Arduino são mostrados são mostradas 2 funções essenciais, a *void setup()*, que é executada uma vez no início quando o programa é rodado, e o *void loop()* que é executado várias vezes, mas nesse código tem mais 6 funções adicionais para o código ficar melhor na hora de visualizar, além de não ficar muita coisa escrita no *void setup()*, elas são: *voidpararmotor()*, *voidaplicartimersegurança()*, *voidligarfrente()*, *voidligarreverso()*, *voidexibirstatus()* e uma função para o “*Delay*”.

As principais funções são o *voidpararmotor()*, *ligarfrente()* e *ligarreverso()*. Em cada uma delas é onde está escrito a comunicação entre o serial monitor do computador, onde pode-se ver a situação atual do motor e pulsos de 200 ms, para quando vai ativar um comando, além de informar o estado do motor de parado, motor ligado, ou se ele está revertendo.

Um aspecto crucial observado durante a implementação do projeto foi a necessidade de implementar um tempo de espera (*delay*) ou um temporizador (*timer*) entre as operações de acionamento do motor, especialmente para a reversão imediata de seu sentido de rotação. Para solucionar essa questão, optou-se por utilizar um potenciômetro, que atua como um regulador de tensão, variando de 0V a 5V. Essa variação de tensão é lida em uma porta analógica do Arduino e, por meio da função *map*, é convertida em um valor de tempo. Por escolha de projeto, esse tempo foi limitado a um máximo de 30 segundos. Para a execução do controle de tempo, foram empregadas as funções *aplicartimer()* e *lertempodelay()*.

Por fim, no *void exibirstatus()* do motor apresenta o estado do motor para o *void loop()*. Esta função comunica via serial, cada ação realizada.

3.4 Bluetooth Electronics

Para realizar uma partida direta com reversão com o uso do módulo *Bluetooth* comandada pelo celular, foi escolhido o aplicativo *Bluetooth Eletronics*, por ser um aplicativo fácil de manusear, fácil de entender pelas opções de programa presente nele. O aplicativo foi desenvolvido pela Keuwlself, com o intuito de desenvolver a comunicação entre dispositivos de forma simples (KEUWLSOLF,2019).

Ao iniciar o aplicativo, o usuário é direcionado à tela principal, que apresenta uma variedade de painéis para seleção. Para operar o sistema, o procedimento padrão inclui a escolha ou edição de um painel, a conexão com o dispositivo via *Bluetooth* e, por fim, a ativação da função de execução para iniciar a comunicação. Para este projeto, foi desenvolvido um painel simplificado, como é ilustrado na Figura 10, com o objetivo de executar as operações específicas de controle. A plataforma do aplicativo oferece nove tipos de controles que podem ser implementados: textos, botões, interruptores, controles deslizantes, *pads*, acelerômetros, indicadores, gráficos e terminais. (KEUWLSOLF,2019) Como mostra na Figura 8, o escolhido foram os botões, por ser mais simples de fazer e mais

fácil de visualizar se ocorrer um erro na comunicação entre o aparelho celular e o módulo HC-05.

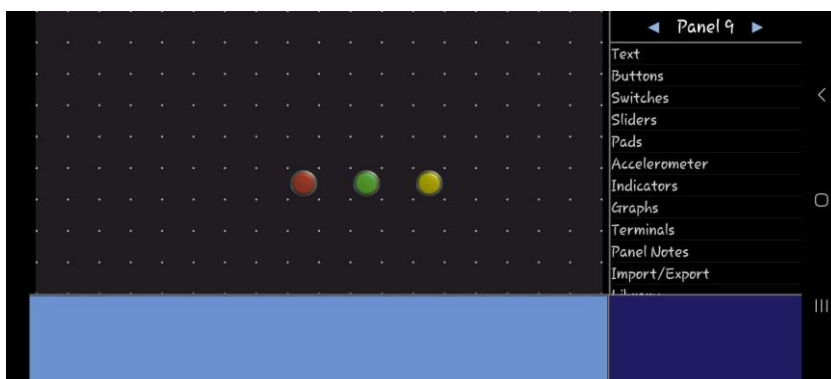


Figura 10 – Bluetooth Electronics, Painei.
Fonte: O autor (2025)

Para configurar cada botão tem que ser escrito nele o que ele deve mandar para o módulo HC-05 e o que ele deve receber, como é mostrado na Figura 11, o botão verde, por exemplo, foi escolhido para desligar, enviando e recebendo o comando '2', sendo esta, uma forma simples de usar e que serviu bem para seu propósito.

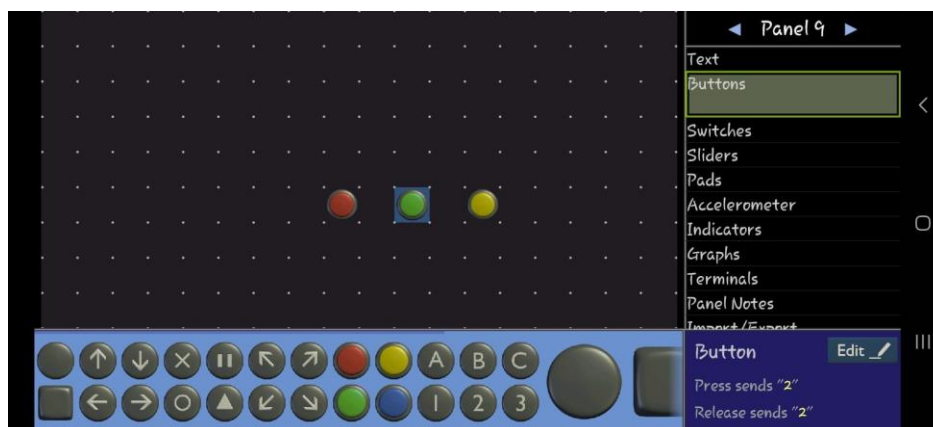


Figura 11 – Bluetooth Electronics, Botões.
Fonte: O autor (2025)

4. RESULTADOS

Os resultados obtidos foram satisfatórios, demonstrando que os objetivos do projeto foram plenamente alcançados. Foi possível acionar, desligar e reverter o sentido de rotação de um motor de forma remota, utilizando um dispositivo móvel (celular) conectado ao sistema via *Bluetooth*, por meio do módulo HC-05. Conforme ilustrado na Figura 12, é apresentado

todos os componentes do sistema interligados. A imagem mostra o motor trifásico conectado aos relés, que por sua vez estão ligados ao Arduino com o módulo HC-05, e o dispositivo móvel (aparelho celular) pareado a ele.

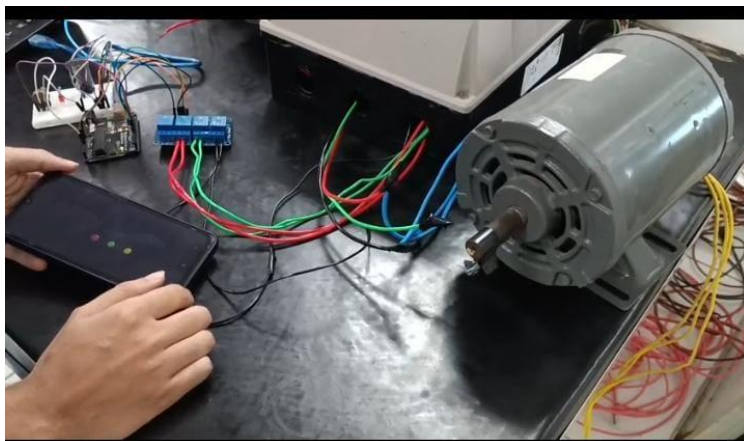


Figura 12 – Motor Trifásico ligado a relé.
Fonte: O autor, (2025)

5. CONCLUSÃO

O que pode ser concluído desse projeto é que, através do Arduino, mesmo sendo uma placa para uso educacional e para coisas mais simples como ligar um LED com resistor, pode ser feito algo mais muito bem elaborado e que pode e não pode ser usado apenas em sala de aula. Com esse projeto para ligar um motor com uso do *Bluetooth*, pode ser usado por empresas para monitoramento, atrás do próprio celular, com uma distancia até considerável de 10 metros sendo o Limite, para caso aconteça algo, a própria pessoa mesmo não estando perto, pode desligar o motor ter total controle, este projeto mostra que com as ferramentas certas, qualquer um que queira aprender um pouco sobre, pode aprender sobre eletrônica embarcada e sobre o conhecimento de motores para realizar um grande projeto.

Mas neste projeto foi usado o Arduino com *Bluetooth*, que acaba por ser limitado, o modulo *Bluetooth* pode alcançar até 10 metros, como foi feito em um laboratório, foi feito bem próximo a conexão entre o HC-05 e o celular, mas para grandes empresas teria que ser feito um limite para a distância, isso claro para grande empresas, para empresas de grande porte o uso mais viável seria com o Arduino com o *WI-FI Shields*, ou até mesmo com o ESP32, mas isso fica para futuros estudos.

REFERÊNCIAS

ARAÚJO FILHO, J. S.; SANTANA, I. A. **Desenvolvimento de uma interface homem-máquina para acionamento e monitoramento de motores de indução trifásicos**. 2025. 48 f. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Elétrica) - Instituto Federal de Goiás, Jataí, 2025. Disponível em: <http://repositorio.ifg.edu.br:8080/handle/prefix/2174>. Acesso em: 20 jul. 2025.

ARDUINO. *SoftwareSerial Library*. [S. l.], 2025. **Arduino Documentation**. Disponível em: <https://docs.arduino.cc/learn/built-in-libraries/software-serial/>. Acesso em: 20 jul. 2025.

CHAPMAN, Stephen J. **Fundamentos de maquinas elétricas**. 5. ed. Porto Alegre: AMGH, 2013. Disponível em: https://www.academia.edu/40609060/Fundamentos_de_Maquinas_Eletricas_Chapman. Acesso em: 20 jul. 2025.

CRAVO, Edilson. **Motor trifásico**: o que é, como funciona, tipos e como ligar. **Blog Kalatec**, [S. l.], 23 set. 2022. Disponível em: <https://blog.kalatec.com.br/motor-trifasico/>. Acesso em: 24 jul. 2025.

ELETROGATE. **Módulo relé 1 canal 5V com optoacoplador**. [S. l.], 2025. Disponível em: <https://www.eletrogate.com/modulo-rele-1-canal-5v>. Acesso em: 28 jul. 2025.

ELETROGATE. **Módulo relé 4 canais 5V com optoacoplador**. [S. l.], 2025. Disponível em: <https://www.eletrogate.com/modulo-rele-4-canais-5v>. Acesso em: 25 jul. 2025

KEUWLISOFT. **Bluetooth Electronics**: User Guide. **Keuwlsoft**, [S. l.], 2019. Disponível em: <https://www.keuwl.com/apps/bluetoothelectronics/userguide/>. Acesso em: 20 jul. 2025.

MAIA, Gustavo Moura Fé. **Acionamento remoto de portões eletrônicos via celular através de microcontrolador**. 2012. 60 f. Trabalho de Conclusão de Curso (Graduação em Engenharia da Computação) - Centro Universitário de Brasília, Brasília, 2012. Disponível em: <https://repositorio.uniceub.br/jspui/handle/123456789/3104>. Acesso em: 19 jul. 2025.

MANUTENÇÃO de motor elétrico monofásico. Eletroval Motores Elétricos, Valinhos, [s. d.]. Disponível em: <https://www.eletovalmotores.com.br/manutencao-motor-eletrico-monofasico>. Acesso em: 25 jul. 2025.

MONK, Simon. **Projetos com Arduino e Android:** Use seu Smartphone ou Tablet para controlar o Arduino. Porto Alegre: Bookman, 2014. Disponível em: https://books.google.com.br/books?hl=pt-BR&lr=&id=7Yg6AgAAQBAJ&oi=fnd&pg=PP2&dq=projetos+arduino+android&ots=1TA bH7WaIv&sig=BxmiTGDhcCAyWkPa3SweWbnus04&redir_esc=y#v=onepage&q=projetos%20arduino%20android&f=false. Acesso em: 13 jul. 2025.

MOTORES elétricos monofásicos: aplicações comuns e suas características específicas. **iBob**, [S. l.], 2025. Disponível em: <https://ibob.com.br/motores-eletricos-monofasicos-aplicacoes-comuns-e-suas-caracteristicas-especificas/>. Acesso em: 24 jul. 2025

MOTORES elétricos são responsáveis pelo consumo de 68% da energia elétrica das fábricas, diz Procel. **Canal da Bioenergia**, [S. l.], 20 fev. 2024. Disponível em: <https://www.canalbioenergia.com.br/motores-eletricos-sao-responsaveis-pelo-consumo-de-68-da-energia-eletrica-das-fabricas-diz-procel/>. Acesso em: 24 jul. 2025.

PATSKO, Luís F. **Tutorial:** aplicações, funcionamento e utilização de sensores. [S. l.]: Maxwell Bohr, 2006. Disponível em: <https://pt.scribd.com/document/63024458/Tutorial-Eletronica-Aplicacoes-e-Funcionamento-de-Sensores>. Acesso em: 21 jul. 2025.

PRAXEDES, M. F. **A evolução dos sistemas de partida dos motores elétricos trifásicos – MIT.** 2011. 50 f. Trabalho de Conclusão de Curso (Graduação em Engenharia Elétrica) - Universidade São Francisco, Itatiba, 2011. Disponível em: <https://lyceumonline.usf.edu.br/salavirtual/documentos/2076.pdf>. Acesso em: 23 jul. 2025.

RIOS, Jefferson et al. **Introdução ao Arduino.** Campo Grande: [s. n.], 2012. Disponível em: https://www.academia.edu/5229813/Introdução_ao_Arduino?auto=download. Acesso em: 15 jul. 2025.

SAKURAI, Ruudi; ZUCHI, Jederson Donizete. **As revoluções industriais até a Indústria 4.0. Revista Interface Tecnológica**, Taquaritinga, SP, v. 15, n. 2, p. 480–491, 2018. Disponível em: <https://revista.fatectq.edu.br/interfacetecnologica/article/view/386>. Acesso em: 24 jul. 2025.

SEGUNDO, Alan R. K.; RODRIGUES, Cristiano L. C. **Eletrônica de Potência e Acionamentos Elétricos**. Ouro Preto: IFMG, 2015. 129 p. Disponível em: <https://www.ifmg.edu.br/ceadop3/apostilas/eletronica-de-potencia-e-acionamentos-eletricos/view>. Acesso em: 7 ago. 2025.

SILVA, Jander Couto da. **Automação residencial com Arduino**. 2022. 51 f. Monografia (Graduação em Tecnologia em Mecatrônica Industrial) - Instituto Federal de Educação, Ciência e Tecnologia do Amazonas, Campus Manaus Distrito Industrial, Manaus, 2022. Disponível em: <http://repositorio.ifam.edu.br/jspui/handle/4321/1077>. Acesso em: 23 jul. 2025.

SILVEIRA, C. B. **O que é a Indústria 4.0 e como ela vai impactar o mundo**. Citisystems, [S. l.], 2017. Disponível em: <https://www.citisystems.com.br/industria-4-0/>. Acesso em: 21 jul. 2025.

SANTOS, Aristoteles. **Estudo da influência da tensão na partida e operação dos motores de indução, utilizando o ATPDraw**. [S. l.]: ResearchGate, 2016. Disponível em: https://www.researchgate.net/publication/310502963_ESTUDO_DA_INFLUENCIA_DA_TENSAO_NA_PARTIDA_E_OPERACAO_DOS_MOTORES_DE_INDUCAO_UTILIZANDO_O_ATPDraw. Acesso em: 23 jul. 2025.

APÊNDICE A – CODIGO FONTE

```

#include <SoftwareSerial.h>

SoftwareSerial bluetooth(10, 11); // RX, TX

// Pinos dos relés
const int releAlimentacao = 7;
const int releFrente = 8;
const int releReverso = 9;

// Pino do potenciômetro para o timer
const int potPin = A0;

// Variáveis de controle
char comando;
char estadoMotor = 'P'; // 'P'=parado, '1'=frente, '0'=reverso
unsigned long ultimoComando = 0;
const unsigned long DEBOUNCE_DELAY = 500; // Anti-spam delay
const unsigned long PULSO_RELE = 200; // Pulso de 200ms para os relés

// Estados dos relés
// Relé de alimentação: HIGH = NC (normalmente fechado), LOW = NA (normalmente aberto)
// Relés de direção: HIGH = NA (normalmente aberto), LOW = NC (normalmente fechado)
const int ALIM_NC = HIGH; // Alimentação normalmente fechado
const int ALIM_NA = LOW; // Alimentação normalmente aberto
const int DIR_NA = HIGH; // Direção normalmente aberto
const int DIR_NC = LOW; // Direção normalmente fechado

void setup() {
  // Configuração dos pinos como saída
  pinMode(releFrente, OUTPUT);
  pinMode(releReverso, OUTPUT);
  pinMode(releAlimentacao, OUTPUT);

  // Estado inicial dos relés
  digitalWrite(releAlimentacao, ALIM_NC); // Alimentação NC (motor ligado)
  digitalWrite(releFrente, DIR_NA); // Frente NA (desligado)
  digitalWrite(releReverso, DIR_NA); // Reverso NA (desligado)

  // Inicialização das comunicações seriais
  Serial.begin(9600);
  bluetooth.begin(9600);

```

```

Serial.println("=== CONTROLE DE MOTOR COM PULSOS DE 200MS ===");
Serial.println("Comandos: 1=Frente, 0=Reverso, 2=Parar");
Serial.println("Potenciômetro: Ajusta tempo de segurança (1-30s)");
Serial.println("LÓGICA DOS RELÉS:");
Serial.println("- Alimentação: NC->NA(200ms)->NC para desligar");
Serial.println("- Direção: NA->NC(200ms)->NA para acionar");
Serial.println("Sistema iniciado e pronto para uso!");
}

```

// Função para parar completamente o motor

```

void pararMotor() {
    Serial.println("Parando motor...");

    // Pulso no relé de alimentação (NC -> NA -> NC)
    Serial.println("Aplicando pulso de DESLIGAMENTO na alimentação...");
    digitalWrite(releAlimentacao, ALIM_NA); // Vai para NA (desliga motor)
    delay(PULSO_RELE); // Mantém por 200ms
    digitalWrite(releAlimentacao, ALIM_NC); // Volta para NC (pronto para ligar)

    // Garante que os relés de direção estejam em NA
    digitalWrite(releFrente, DIR_NA);
    digitalWrite(releReverso, DIR_NA);

    estadoMotor = 'P'; // Atualiza o estado para Parado
    Serial.println("MOTOR PARADO - Pulso de 200ms aplicado");
}

```

// Função para ler o tempo de delay do potenciômetro

```

unsigned long lerTempoDeDelay() {
    int valorPot = analogRead(potPin);
    unsigned long tempo = map(valorPot, 0, 1023, 1000, 30000);

    Serial.print("Potenciômetro: ");
    Serial.print(valorPot);
    Serial.print(" -> Tempo: ");
    Serial.print(tempo / 1000);
    Serial.println("s");

    return tempo;
}

```

// Função para aplicar timer de segurança na reversão

```

void aplicarTimerSeguranca() {
    unsigned long tempoDeEspera = lerTempoDeDelay();
    Serial.print("Aguardando timer de segurança: ");
    Serial.print(tempoDeEspera / 1000);
    Serial.println("s");

    unsigned long inicio = millis();
}

```

```

while (millis() - inicio < tempoDeEspera) {
    unsigned long restante = tempoDeEspera - (millis() - inicio);
    if (restante > 0 && restante % 5000 == 0) {
        Serial.print("Restam: ");
        Serial.print(restante / 1000);
        Serial.println("s");
    }
    delay(100);
}

Serial.println("Timer concluído!");
}

void ligarFrente() {
    // SE o motor estava em REVERSO, aplica o timer de segurança
    if (estadoMotor == '0') {
        Serial.println("Iniciando reversão REVERSO -> FRENTE...");
        pararMotor(); // Para o motor primeiro
        aplicarTimerSeguranca();
    }

    // Pulso no relé de FRENTE (NA -> NC -> NA)
    Serial.println("Aplicando pulso de FRENTE...");
    digitalWrite(releFrente, DIR_NC); // Vai para NC (aciona frente)
    delay(PULSO_RELE); // Mantém por 200ms
    digitalWrite(releFrente, DIR_NA); // Volta para NA

    estadoMotor = '1'; // Atualiza o estado para Frente
    Serial.println("MOTOR CONFIGURADO PARA FRENTE - Pulso de 200ms
    aplicado");
}

void ligarReverso() {
    // SE o motor estava para FRENTE, aplica o timer de segurança
    if (estadoMotor == '1') {
        Serial.println("Iniciando reversão FRENTE -> REVERSO...");
        pararMotor(); // Para o motor primeiro
        aplicarTimerSeguranca();
    }

    // Pulso no relé de REVERSO (NA -> NC -> NA)
    Serial.println("Aplicando pulso de REVERSO...");
    digitalWrite(releReverso, DIR_NC); // Vai para NC (aciona reverso)
    delay(PULSO_RELE); // Mantém por 200ms
    digitalWrite(releReverso, DIR_NA); // Volta para NA

    estadoMotor = '0'; // Atualiza o estado para Reverso
    Serial.println("MOTOR CONFIGURADO PARA REVERSO - Pulso de 200ms
    aplicado");
}

```

```

}

// Função para exibir status atual do sistema
void exibirStatus() {
  Serial.println("=== STATUS DO SISTEMA ===");
  Serial.print("Estado do motor: ");
  switch(estadoMotor) {
    case 'P': Serial.println("PARADO"); break;
    case '1': Serial.println("FRENTE"); break;
    case '0': Serial.println("REVERSO"); break;
  }

  Serial.println("Estado dos relés:");
  Serial.print("- Alimentação: ");
  Serial.println(digitalRead(releAlimentacao) == ALIM_NC ? "NC (Motor pode ligar)"
: "NA (Motor desligado)");
  Serial.print("- Frente: ");
  Serial.println(digitalRead(releFrente) == DIR_NA ? "NA (Inativo)" : "NC (Ativo)");
  Serial.print("- Reverso: ");
  Serial.println(digitalRead(releReverso) == DIR_NA ? "NA (Inativo)" : "NC (Ativo)");

  int valorPot = analogRead(potPin);
  Serial.print("- Potenciômetro: ");
  Serial.print(valorPot);
  Serial.print(" (");
  Serial.print(map(valorPot, 0, 1023, 1, 30));
  Serial.println("s)");
  Serial.println("=====");
}

void loop() {
  // Verifica comandos via Bluetooth
  if (bluetooth.available()) {
    comando = bluetooth.read();

    // Anti-spam: não processa múltiplos comandos muito rápidos
    if (millis() - ultimoComando < DEBOUNCE_DELAY) {
      return;
    }
    ultimoComando = millis();

    Serial.print("Comando recebido: ");
    Serial.println(comando);

    switch(comando) {
      case '1':
        // Só liga se não estiver já na frente
        if (estadoMotor != '1') {
          ligarFrente();
        }
      }
    }

```

```

    } else {
        Serial.println("Motor já está configurado para FRENTE");
    }
    break;

case '0':
    // Só liga se não estiver já no reverso
    if (estadoMotor != '0') {
        ligarReverso();
    } else {
        Serial.println("Motor já está configurado para REVERSO");
    }
    break;

case '2':
    // Só para se não estiver já parado
    if (estadoMotor != 'P') {
        pararMotor();
    } else {
        Serial.println("Motor já está PARADO");
    }
    break;

case 'S': // Comando adicional para status
    exhibirStatus();
    break;

default:
    Serial.println("Comando inválido!");
    Serial.println("Use: 1=Frente, 0=Reverso, 2=Parar, S=Status");
    break;
}
}

// Verifica se há comandos via Serial (para debug)
if (Serial.available()) {
    char cmdSerial = Serial.read();
    if (cmdSerial == '1' || cmdSerial == '0' || cmdSerial == '2' || cmdSerial == 'S') {
        comando = cmdSerial;
        // Processa o comando diretamente
        ultimoComando = millis();
    }
}
}
}

```