

Rafael Dias Campos

# **Jogo de Xadrez com Manipuladores Robóticos**

Belo Horizonte

2022

Rafael Dias Campos

## **Jogo de Xadrez com Manipuladores Robóticos**

Trabalho de Conclusão de Curso apresentado  
ao Curso de Engenharia de Computação do  
Centro Federal de Educação Tecnológica de  
Minas Gerais, como requisito parcial para a  
obtenção do título de Bacharel em Engenharia  
de Computação.

Centro Federal de Educação Tecnológica de Minas Gerais – CEFET-MG

Departamento de Computação

Curso de Engenharia da Computação

Orientador: Ramon da Cunha Lopes

Belo Horizonte

2022

# Resumo

Atualmente, existe uma grande procura por funcionários especializados em Tecnologia da Informação (TI) e áreas similares, sendo percebida no mundo todo uma carência de profissionais qualificados para atuar nessas áreas. Esse problema ocorre parcialmente devido a um desinteresse à capacitação técnica por parte de crianças e jovens, frequentemente devido a uma concepção de que essas áreas são difíceis e/ou desinteressantes. Com o foco neste aspecto, esse projeto visa desenvolver um sistema que possibilita o jogo de Xadrez entre duas pessoas por meio de manipuladores robóticos. Este sistema pode ser apresentado para crianças e jovens em feiras educativas e eventos similares com o objetivo de introduzir conceitos básicos e instigar o interesse pelas áreas de computação, elétrica e controle. Sistemas já existentes para mover peças de xadrez com braços mecânicos normalmente são utilizados para que uma máquina jogue contra um jogador humano em campeonatos e com isso apresentam alto custo monetário, visto que priorizam velocidade e precisão nos movimentos. Por isso, eles não são muito adequados para usos educacionais e têm-se a necessidade de desenvolver um sistema mais barato e com foco na simplicidade e na capacidade de proporcionar divertimento para os jogadores. Nesse projeto serão utilizados manipuladores robóticos comerciais já adquiridos pela instituição de ensino. Para o controle dos manipuladores, serão utilizados dois *joysticks*, um para cada jogador, que enviam sinais para uma placa de controle dos manipuladores por meio de um microcontrolador. Até o momento, foi desenvolvida uma placa de controle para enviar os sinais para os manipuladores robóticos, e um *software* para realizar a leitura dos *joysticks* e enviar os sinais para a placa de controle. Ainda está em desenvolvimento o *software* que implementará a lógica do jogo de Xadrez, e o controle do segundo manipulador. **Palavras-chave:** Manipuladores Robóticos. Controle Digital. Xadrez.

# Abstract

Currently, there is a great demand for specialized professionals in IT and in similar areas, and it is perceived worldwide a shortage of qualified professionals to work in these areas. This problem occurs partially due to a lack of technical training on the part of children and young people, often due to a conception that these areas are difficult and / or uninteresting. Focusing on this aspect, this project aims to develop a system that allows the game of Chess between two people through robotic manipulators. This system can be presented to children and young people in educational fairs and similar events with the aim of introducing basic concepts and instigating interest in the areas of computing, electrical and control. Systems already exist to move chess pieces with mechanical arms usually used to allow a machine to play against a human player in tournaments and with that they present high monetary cost, for they prioritize speed and precision in movements. Therefore, they are not very suitable for educational uses and there is a need to develop a cheaper system and with a focus on simplicity and the ability to provide fun for players. In this project will be used commercial robotic manipulators already acquired by the educational institution. For the control of the manipulators, two joysticks will be used, one for each player, which send signals to a control board through a microcontroller. So far, a control board has been developed to send signals to the robotic manipulators and a software to read the joysticks and send signals to the control board. The software that implements the logic of the Chess game is still under development, as well as the control of the second manipulator.

**Keywords:** Robotic Arms. Digital Control. Chess.

# ***Lista de ilustrações***

Figura 1 – Manipulador Robótico Mentor . . . . .	10
Figura 2 – Manipulador Robótico Azul . . . . .	10
Figura 3 – Manete para os jogadores . . . . .	11
Figura 4 – Largura do pulso do sinal de controle . . . . .	12
Figura 5 – Microcontrolador Arduino Mega 2560 . . . . .	12
Figura 6 – Módulo de Ponte H . . . . .	13
Figura 7 – Montagem do Sistema . . . . .	14
Figura 8 – Esquemático Simplificado de um CI L293D e um CI 74LS02 . . . . .	15
Figura 9 – Layout da placa de controle . . . . .	17
Figura 10 – Placa de controle produzida . . . . .	17

# **Lista de tabelas**

Tabela 1 – Características do manipulador robótico Mentor . . . . .	10
Tabela 2 – Características do manipulador robótico RD5NT . . . . .	10

# **Lista de abreviaturas e siglas**

CI              Circuito Integrado

CEFET-MG    Centro Federal de Educação Tecnológica de Minas Gerais

STEM          *Science, Technology, Engineering and Mathematics* [Ciência, Tecnologia, Engenharia e Matemática]

PID             *Proportional Integral Derivative* [Proporcional Integral Derivativo]

PWM            *Pulse Width Modulation* [Modulação de Largura de Pulso]

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>8</b>
<b>2</b>	<b>PLANEJAMENTO</b>	<b>9</b>
<b>2.1</b>	<b>Escolha do equipamento</b>	<b>9</b>
2.1.1	Manipuladores	9
2.1.2	Manete para os jogadores	11
2.1.3	Microcontrolador	11
2.1.4	Placa de Controle	12
2.1.5	Computador	13
<b>2.2</b>	<b>Projeto do sistema</b>	<b>14</b>
<b>3</b>	<b>DESENVOLVIMENTO</b>	<b>15</b>
<b>3.1</b>	<b>Desenvolvimento da placa de controle</b>	<b>15</b>
<b>3.2</b>	<b>Leitura das manetes</b>	<b>17</b>
<b>3.3</b>	<b>Cálculo da posição</b>	<b>18</b>
<b>3.4</b>	<b>Controle dos motores</b>	<b>18</b>
<b>4</b>	<b>CONCLUSÕES PRELIMINARES</b>	<b>19</b>
	<b>REFERÊNCIAS</b>	<b>20</b>
	<b>ANEXOS</b>	<b>21</b>
	<b>ANEXO A – ESQUEMÁTICO DA PLACA DE CONTROLE</b>	<b>23</b>

# 1 Introdução

Atualmente, existe uma grande procura por funcionários especializados em Tecnologia da Informação (TI) e áreas similares, sendo percebida no mundo todo uma grande carência de profissionais qualificados para atuar nessas áreas. Apesar do grande crescimento nas áreas de STEM (Science, Technology, Engineering and Mathematics) [Ciência, Tecnologia, Engenharia e Matemática], o número de profissionais qualificados nessas áreas não acompanha esse crescimento e a perspectiva é de que essa situação se acentue ainda mais no futuro (LEPRINCE-RINGUET, 2021).

Devido a essa carência de profissionais, torna-se importante a busca por formas de incentivar o aprendizado e a busca por conhecimento por parte dos jovens. Visando solucionar esse problema, foi decidido realizar um trabalho que incorpore conceitos de robótica, já que seu uso em atividades com crianças consegue influenciar positivamente o desenvolvimento de habilidades da área de STEM (DOROUKA; PAPADAKIS; KALOGIANNAKIS, 2020).

Com base nisso, foi proposto realizar o desenvolvimento de uma plataforma que utilize recursos computacionais passível de ser utilizada para demonstrar conceitos nas áreas de computação, elétrica e controle. Para aumentar o interesse por esta plataforma foi definido que ela deve permitir que os participantes joguem uma partida de Xadrez. Correlacionando essas ideias, implementou-se um jogo de Xadrez que pode ser jogado através de braços robóticos.

Este trabalho visa desenvolver um sistema de controle de manipuladores robóticos que permitam que dois jogadores participem em uma partida de Xadrez. O sistema deve ser intuitivo e fácil de ser utilizado, de forma que os jogadores não precisem de conhecimentos avançados para utilizá-lo. Além disso, os materiais utilizados devem ser de baixo custo, para facilitar a sua produção e demonstrar que é possível desenvolver um produto de qualidade utilizando materiais baratos.

Com o desenvolvimento dessa plataforma, será possível demonstrar conceitos de computação, elétrica e controle de forma prática e divertida. Ela poderá ser facilmente transportada para diferentes locais e apresentada em eventos, como feiras de ciências e escolas. Dessa forma, ela pode promover e instigar a busca por conhecimento, além de atrair futuros profissionais para a área de TI.

## 2 Planejamento

Antes de iniciar o desenvolvimento do projeto, foi necessário especificar os recursos necessários para a montagem do sistema, assim como a forma em que eles serão utilizados. Esta seção descreve a escolha do equipamento e o projeto inicial do sistema.

### 2.1 Escolha do equipamento

Para o desenvolvimento do projeto, foram disponibilizados, pela instituição CEFET-MG, dois manipuladores robóticos e diversos dispositivos que podem ser utilizados para seu controle. A partir desse equipamento, e de outros disponíveis no mercado, foi decidido como o projeto seria realizado. Os equipamentos utilizados para o projeto estão descritos a seguir:

#### 2.1.1 Manipuladores

Os principais elementos deste trabalho são os manipuladores robóticos, portanto foi feito inicialmente um estudo sobre seu funcionamento e sobre como seu controle pode ser realizado para movimentar as peças de xadrez.

Foi disponibilizado um manipulador robótico de modelo Mentor de cor preta, conforme a figura 1, e um manipulador de modelo RD5NT de cor azul, conforme a figura 2. Esses manipuladores possuem características diferentes, portanto seu controle deve ser realizado de forma diferente. Eles possuem diversas juntas movimentadas por motores de corrente contínua, e permitem que o manipulador funcione de forma similar a um braço humano. Além disso, cada junta possui um potenciômetro que indica a posição atual do eixo, por meio de um sinal analógico.

O manipulador Mentor apresenta 5 graus de liberdade e um *gripper* que pode ser utilizado para pegar e soltar objetos. Além disso, ele possui caixas de redução em seus motores, o que permite que ele mantenha sua posição mesmo após o desligamento dos motores. Suas dimensões e faixas de movimento são apresentadas na tabela 1 ([OLUNLOYO; AYOMOH; ADEOTI, 2011](#)).

Já o manipulador RD5NT possui apenas 4 graus de liberdade e um *gripper*. Ele não possui caixas de redução, o que faz com que ele perca sua posição quando os motores são desligados. Portanto, seu controle deve ser realizado de forma contínua, para que ele permaneça na posição desejada. Suas dimensões e faixas de movimento são apresentadas na tabela 2 ([RIUL; MONTENEGRO; FERREIRA, 2018](#)).

Tabela 1 – Características do manipulador robótico Mentor

Eixo	Movimento angular (graus)	Comprimento (mm)
Eixo 0 ( <i>Waist</i> )	210	185
Eixo 1 ( <i>Shoulder</i> )	180	165
Eixo 2 ( <i>Elbow</i> )	230	150
Eixo 3 ( <i>Left Wrist Axe</i> )	320	0
Eixo 4 ( <i>Right Wrist Axe</i> )	320	0
<i>Wrist Pitch</i>	140	-
<i>Wrist Roll</i>	320	-

Tabela 2 – Características do manipulador robótico RD5NT

Eixo	Movimento angular (graus)	Comprimento (mm)
Eixo 0 ( <i>Waist</i> )	293	110
Eixo 1 ( <i>Shoulder</i> )	107	120
Eixo 2 ( <i>Elbow</i> )	284	160
Eixo 3 ( <i>Wrist Axe</i> )	360	0

Figura 1 – Manipulador Robótico Mentor



Fonte: <http://arquivo.eng.br/robotica>

Figura 2 – Manipulador Robótico Azul



Fonte: <http://arquivo.eng.br/robotica>

### 2.1.2 Manete para os jogadores

Para que os jogadores possam interagir com os manipuladores, foi utilizado uma manete de modelo *batpad*, que possui dois *joysticks*, conforme a figura 3. Cada um desses *joysticks* deve ser alimentado com 5V e envia sinais que indicam sua posição nos eixos X e Y, além de um botão que pode ser pressionado.

Os sinais de posição são enviados de forma analógica, com valores de tensão que variam de 0V a 5V. O sinal do botão é enviado de forma digital, com valores de tensão de 0V ou 5V, sendo que 0V indica que o botão está pressionado.

Cada jogador possui uma manete para controlar um dos manipuladores robóticos.

Figura 3 – Manete para os jogadores



Fonte: <https://www.robocore.net/acessorios-robocore/controle-batpad>

### 2.1.3 Microcontrolador

Para realizar a integração entre a manete e os manipuladores, foi utilizado um microcontrolador Arduino Mega 2560, conforme a figura 5. Esse microcontrolador possui um *clock* de 16MHz, 256KB de memória *flash*, 8KB de memória *RAM* e 4KB de memória *EEPROM*. Ele apresenta 16 entradas analógicas e 54 entradas e saídas digitais, sendo que 15 delas podem ser utilizadas como saídas *PWM* (ARDUINO, 2022).

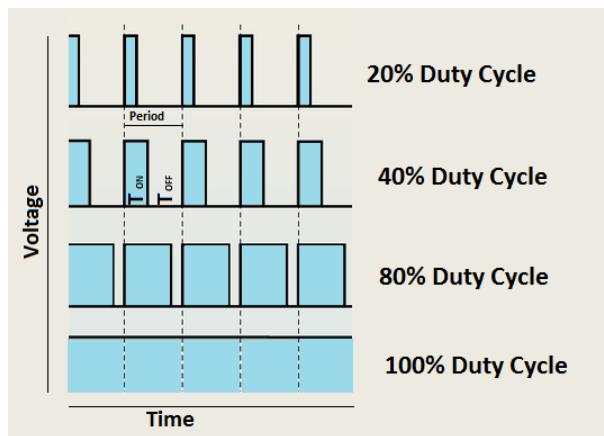
Cada microcontrolador é responsável por controlar um dos manipuladores robóticos. Ele realiza a leitura dos sinais analógicos e digitais provenientes da manete, que se referem à posição atual do *joystick* e o estado do botão, e a leitura de sinais analógicos provenientes do manipulador robótico, que indicam a posição atual de cada junta. A partir desses sinais, ele realiza o cálculo das posições desejadas de cada junta do manipulador e envia sinais de controle para elas.

Os sinais de controle são digitais do tipo *PWM* (*Pulse Width Modulation*), que variam de 0V a 5V, para possibilitar o controle da velocidade de rotação dos motores através da variação da largura do pulso. Valores maiores de *Duty Cycle* [Ciclo de Trabalho]

resultam em maior velocidade de rotação do motor, pois o sinal permanece em nível alto por um período maior de tempo. Por outro lado, valores menores de *Duty Cycle* resultam em menor velocidade de rotação do motor, pois o sinal permanece em nível alto por um período menor de tempo. Para definir o sentido de rotação do motor, foi utilizado um sinal digital que indica se ele deve girar no sentido horário ou anti-horário.

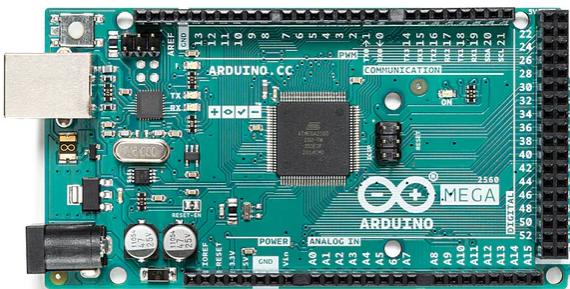
Por fim, o microcontrolador é conectado a um computador através de um cabo USB. O Arduino é responsável por comunicar com o computador quais os movimentos que o jogador deseja realizar com seu manipulador robótico. A partir disso, o computador responde se o movimento é permitido ou não.

Figura 4 – Largura do pulso do sinal de controle



Fonte: <https://create.arduino.cc/projecthub/muhammad-aqib/arduino-pwm-tutorial-ae9d71>

Figura 5 – Microcontrolador Arduino Mega 2560



Fonte: <https://store.arduino.cc/products/arduino-mega-2560-rev3>

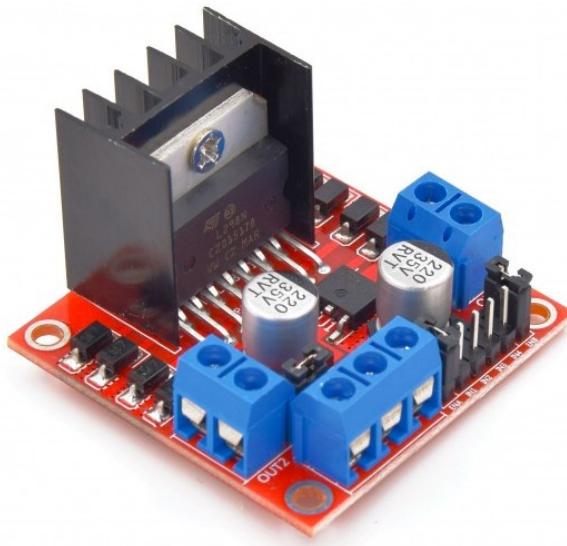
#### 2.1.4 Placa de Controle

Para controlar os motores dos manipuladores, é necessário o uso de uma placa de controle para converter os sinais de baixa potência provenientes do Microcontrolador em sinais de maior potência que movimentam as juntas dos manipuladores robóticos. Essa

placa é alimentada com 12v e ela recebe os sinais digitais de direção e de *PWM* do Arduino e movimenta os motores de acordo.

Para obter essa funcionalidade, foi feito o uso de módulos de ponte H, similar aos utilizados na placa apresentada na figura 6. Uma ponte H é um Circuito Integrado (CI) utilizado para aplicar uma tensão variável a um componente através de um sinal de *PWM*. Ela também permite alterar a direção em que a corrente é aplicada no componente, o que possibilita inverter o sentido de rotação de um motor (DIGILENT, 2012).

Figura 6 – Módulo de Ponte H



Fonte: <https://www.smart-prototyping.com/L298N-Dual-H-bridge-Motor-Driver-Board>

### 2.1.5 Computador

Para gerenciar o jogo, foi necessário o uso de um computador. Ele é responsável por receber realizar a comunicação com o microcontrolador para gerenciar o jogo de Xadrez.

Para comunicar com o Arduino, é utilizado um cabo USB e comunicação serial. A comunicação serial é um protocolo de comunicação assíncrona que permite a transmissão de dados entre dois dispositivos. Nesse caso, o Arduino envia o movimento desejado pelo jogador e o computador responde se o movimento é permitido ou não.

A verificação do movimento é feita através de um algoritmo que implementa a lógica do jogo de Xadrez. Através dele, são avaliadas as regras de movimentação de cada peça, as regras de captura e de xeque. Também são verificadas as regras de xeque-mate e de empate por falta de material. Com base nessas regras e no estado atual do tabuleiro, o computador responde se o movimento é permitido ou não.

## 2.2 Projeto do sistema

Após a definição de todos os equipamentos a serem utilizados, foi feito o projeto do sistema. Esse projeto consiste na definição de como os dispositivos serão interligados e como o sistema irá funcionar.

A manete deve ser alimentada com 5V e ser conectada ao Arduino para permitir a leitura dos sinais de entrada. Essa conexão é realizada com 10 cabos, sendo 5 para cada joystick com seu respectivo botão.

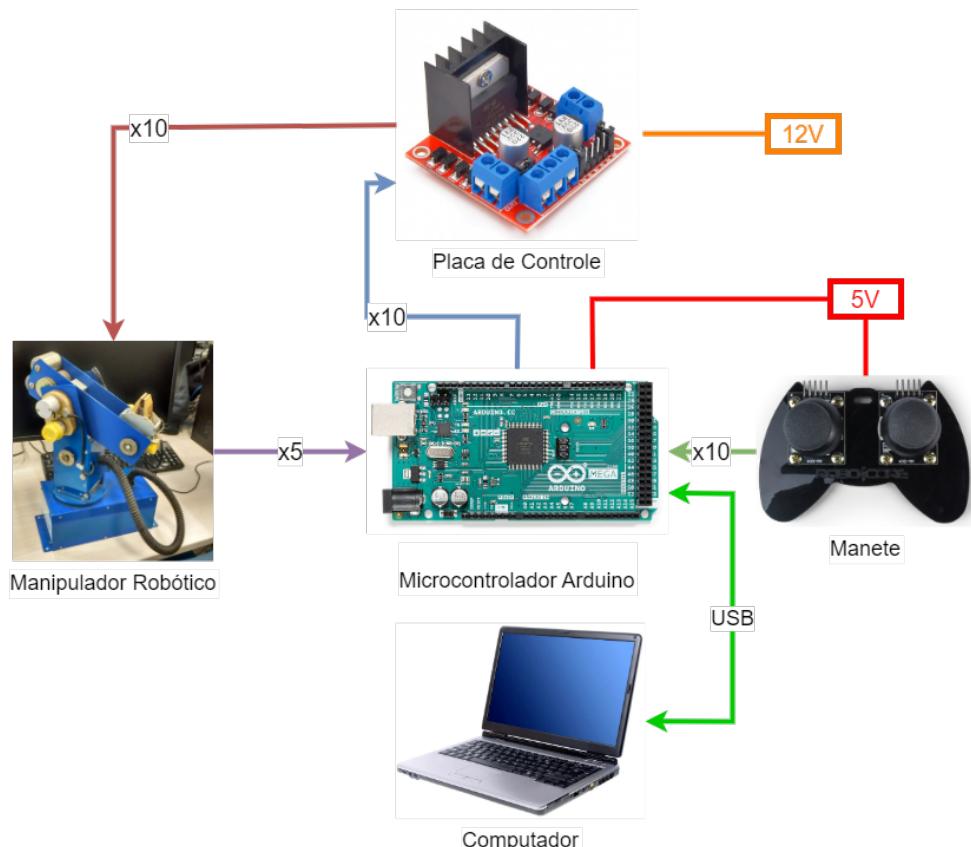
A placa de controle deve ser alimentada com 12V e também deve ser conectada ao Arduino para receber os sinais de controle. Essa conexão é realizada com 10 cabos, 2 para o controle de cada junta do manipulador robótico. Essa placa também deve ser conectada aos motores do manipulador robótico, também com 10 cabos, 2 para cada junta.

O manipulador robótico deve ser conectado ao Arduino para o envio dos sinais de posição de cada junta. Essa conexão é realizada com 5 cabos, um para cada junta.

Por fim, o Arduino deve ser alimentado com 5V e deve ser conectado a um computador para a implementação da lógica do jogo.

A montagem do sistema é mostrada na Figura 7.

Figura 7 – Montagem do Sistema



Fonte: Do próprio autor

# 3 Desenvolvimento

Após o planejamento do projeto, foi feito o desenvolvimento de cada etapa. Inicialmente, foi feito o desenvolvimento da placa de controle para poder acionar os motores. Em seguida, o *software* para ler os dados das manetes e enviar os comandos para a placa de controle foi desenvolvido.

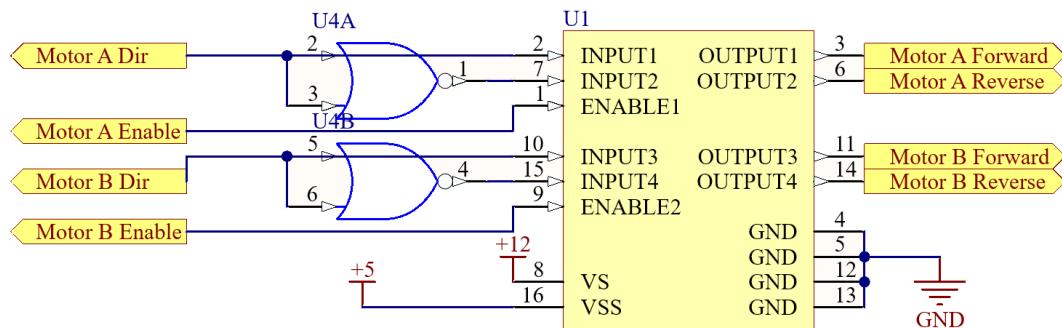
## 3.1 Desenvolvimento da placa de controle

Primeiramente, foi feito o desenvolvimento da placa de controle dos manipuladores, pois é necessária para testar o funcionamento dos motores e do *software* que será desenvolvido. Para isso, foi necessário definir quais componentes utilizar e como conectá-los.

Conforme descrito na subseção 2.1.4, a placa de controle deve utilizar uma ponte H para o controle de cada junta. Para isso, foi escolhido o CI L293D, que possui duas pontes H e suporta tensões de 12V. Como é necessário controlar 6 motores, foram utilizados 3 CI L293D.

Para simplificar o controle e evitar problemas de acionamento duplo das entradas das pontes H, foi utilizado o CI 74LS02 como um inversor lógico. Dessa forma, a placa de controle possui para cada junta uma entrada de *Enable* para ligar/desligar o acionamento da junta, e uma porta de *Direction* para definir a direção de movimentação dela. A partir dessas entradas, o CI L293D é acionado e o motor é controlado. A Figura 8 mostra o esquemático simplificado de um CI L293D e um CI 74LS02.

Figura 8 – Esquemático Simplificado de um CI L293D e um CI 74LS02



Fonte: Do próprio autor

Com os componentes principais definidos foi feito o desenvolvimento do esquemático da placa de controle com o auxílio do software *Altium Designer*. O Anexo A mostra o esquemático completo da placa de controle. Nesse esquemático foram adicionados resistores

de *pulldown* para garantir que as entradas dos CI L293D e 74LS02 permaneçam em nível lógico baixo caso não estejam conectadas ao microcontrolador. Também foram adicionadas *LEDs* para indicar a alimentação de 5V e 12V da placa.

Após o desenvolvimento do esquemático, foi feita a montagem da Placa de Circuito Impresso (PCB), ainda com o auxílio do software *Altium Designer*. Para isso, os componentes foram posicionados no *layout* da placa, tendo em vista a economia de espaço e a necessidade de manter os componentes próximos para facilitar sua conexão. Em seguida, as trilhas e vias que realizam a conexão dos componentes foram desenhadas. Para permitir a conexão de todos os componentes, foi necessário utilizar uma placa com 2 camadas. A Figura 9 mostra o *layout* final da placa de controle.

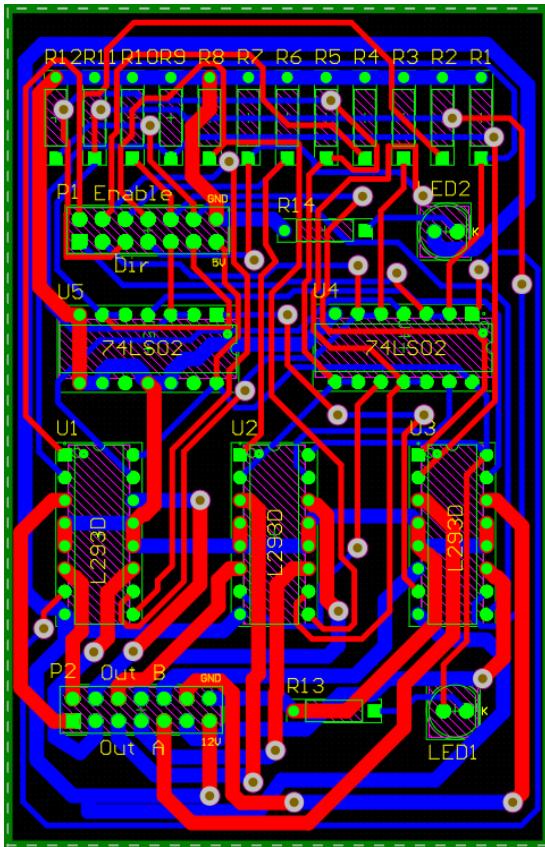
Com o *layout* finalizado, foi feita a produção da PCB de forma manual. Para isso, o negativo do *layout* foi impresso em uma folha de transparência. Depois, uma placa de circuito impresso de duas camadas foi cortada no tamanho desejado.

Em seguida, foi feita a transferência do *layout* para a placa. Para isso, uma fina camada de tinta fotossensível destinada para PCB foi aplicada sobre a placa. Essa tinta foi pré-curada à 75°C durante 15 minutos com o auxílio de uma base de aquecimento, para garantir que ela não se descolasse da placa. Após a pré-cura, a tinta foi exposta à luz ultravioleta por 3 minutos, utilizando a transparência com o *layout* como máscara. Em seguida, a placa foi submersa em uma solução de carbonato de sódio para realizar a revelação do *layout*. Após a placa ser revelada, a tinta foi curada à 85°C durante 30 minutos.

Esse processo de transferência do *layout* foi repetido para a segunda camada da placa. Em seguida, foi utilizada uma solução de percloreto de ferro para corroer as áreas de cobre que não receberam tinta. Após a corrosão, a placa foi mergulhada em uma solução de hidróxido de sódio para remover a tinta.

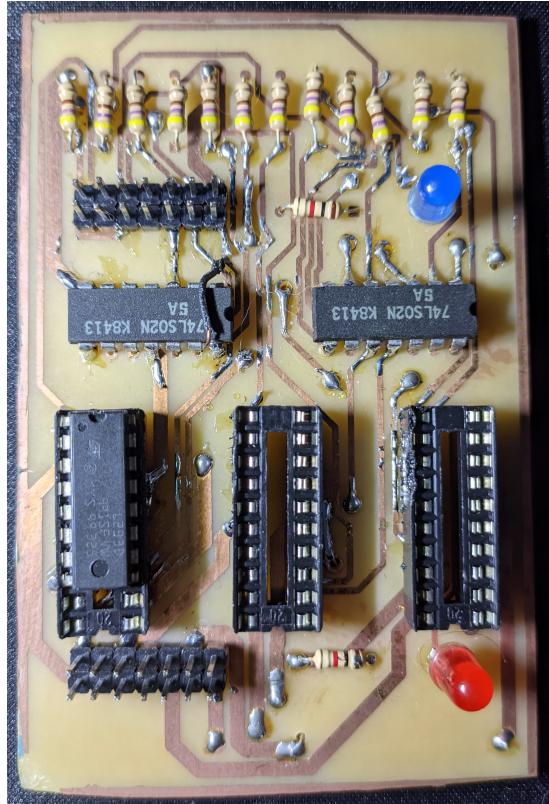
Após a corrosão da PCB, foi feita a perfuração das vias e dos furos para os componentes. Por fim, foi feita a soldagem dos componentes na placa e a conexão das vias. A Figura 10 mostra a placa de controle montada.

Figura 9 – Layout da placa de controle



Fonte: Do próprio autor

Figura 10 – Placa de controle produzida



Fonte: Do próprio autor

### 3.2 Leitura das manetes

Após a montagem da placa de controle, foi desenvolvido um *software* com o microcontrolador para realizar a leitura dos dados das manetes. Conforme descrito na subseção 2.1.2, cada manete possui dois *joysticks* com dois eixos e um botão cada.

Para realizar a leitura dos eixos dos *joysticks*, foram utilizadas as entradas analógicas do microcontrolador. Como o Arduino utiliza um conversor analógico-digital de 10 bits, os valores lidos variam de 0 a 1023. Valores próximos de 512 representam a posição central do *joystick*, enquanto valores próximos de 0 ou 1023 representam as posições extremas. Para aprimorar a usabilidade da manete, foi implementada uma área de *deadzone*, na qual o valor lido é considerado como zero, para evitar que o manipulador se movimente sem a intenção do usuário.

Para realizar a leitura do botão, foi utilizada uma entrada digital. Esses botões possuem um *pull-up* interno, o que significa que o valor lido é 1 quando o botão não está pressionado e 0 quando o botão está pressionado.

Os valores lidos são armazenados em uma variável, que é utilizada para realizar o

controle do movimento do manipulador robótico.

### 3.3 Cálculo da posição

Para calcular a posição do manipulador robótico, foi utilizado como base o *software* desenvolvido anteriormente para ler os dados das manetes.

Para essa primeira parte do trabalho, o cálculo de posição foi feito de forma individual para cada junta. Dessa forma, o usuário pode utilizar os 4 eixos dos *joysticks* para controlar o manipulador robótico. Além disso, o usuário pode utilizar o botão de cada manete para controlar a abertura e o fechamento da garra.

### 3.4 Controle dos motores

O controle dos motores foi incorporado no *software* desenvolvido na subseção anterior, que realiza o cálculo da posição desejada das juntas.

Primeiramente foi implementada a leitura da posição atual de cada junta, utilizando as entradas analógicas do microcontrolador. Os valores de 10 bits lidos pelo Arduino são convertidos para graus, utilizando como referência as faixas de movimento descritas nas tabelas 1 e ??.

Depois disso, foi feita a implementação de um controlador para o manipulador. Esse controlador é executado a cada 10ms, de acordo com a configuração da interrupção do *timer* do Arduino. Nessa interrupção, o microcontrolador calcula o erro entre a posição desejada e a posição atual de cada junta, com base nos valores lidos do manipulador e das manetes. Em seguida, o valor de saída é calculado utilizando um algoritmo PID. Esse algoritmo utiliza três constantes ( $K_p$ ,  $K_i$  e  $K_d$ ) e os valores do erro atual, da integral dos erros anteriores e da derivada do erro atual para calcular a saída. Por fim, esse valor de saída é convertido em um valor de *duty cycle*, e é utilizado para gerar o sinal PWM enviado para a placa de controle.

## 4 Conclusões Preliminares

Durante o desenvolvimento do projeto, foi possível finalizar a placa de controle e um *software* básico que permite o controle independente de cada junta. Nessa primeira etapa de desenvolvimento, não foi possível implementar a lógica do jogo de Xadrez, tarefa que será realizada na próxima etapa.

O processo de produção da placa de controle foi mais demorado do que o inicialmente esperado, em parte pela necessidade de um cuidado especial para garantir que ambas as camadas estejam alinhadas. Além disso, o processo de perfuração e soldagem da placa foi um pouco complicado e demorado. Apesar disso, o resultado final foi satisfatório e a placa de controle apresenta um funcionamento adequado para o controle dos manipuladores.

O *software* desenvolvido para o controle dos manipuladores, apesar de funcional, não facilita o controle deles. Realizar o controle independente de cada junta é um processo complicado e pode provocar um desinteresse dos jogadores pelo produto desenvolvido. Para a próxima etapa, esse *software* será melhorado para que o controle dos manipuladores seja realizado apenas em duas dimensões de forma que o microcontrolador será responsável pelo cálculo dos ângulos das juntas.

A integração dos microcontroladores com o computador também será desenvolvida na próxima etapa do trabalho. Essa integração será responsável por verificar se os movimentos realizados pelos jogadores são validos e por identificar quando um jogador ganhou ou perdeu o jogo.

## Referências

- ARDUINO. *Arduino® MEGA 2560 Rev3 Datasheet*. 2022. Disponível em <https://docs.arduino.cc/static/7a2c58aba77ca2a88e9a4b43a8b6a072/A000067-datasheet.pdf> . Acesso em 10/12/2022.
- DIGILENT. *H-Bridges*. 2012. Disponível em <https://learn.digilentinc.com/Documents/325> . Acesso em 10/12/2022.
- DOROUKA, P.; PAPADAKIS, S.; KALOGIANNAKIS, M. Tablets and apps for promoting robotics, mathematics, stem education and literacy in early childhood education. *Int. J. Mobile Learning and Organisation*, Vol. 14, No. 2, 2020, Rethymnon and Heraklion, Crete, Greece, 2020.
- LEPRINCE-RINGUET, D. *The shortage of tech workers is about to become an even bigger problem for everyone*. [S.l.]: ZDNET, 2021. Disponível em <https://www.zdnet.com/article/the-shortage-of-tech-workers-is-about-to-become-an-even-bigger-problem-for-everyone> . Acesso em 10/11/2022.
- OLUNLOYO, V. O.; AYOMOH, M.; ADEOTI, I. On the mentor arm position placement problem: A forward kinematics analysis. 2011.
- RIUL, J. A.; MONTENEGRO, P. H. de M.; FERREIRA, G. de S. Controle neural de três elos de um robô de cinco graus de liberdade. 2018.

## Anexos



# ANEXO A – Esquemático da Placa de Controle

