



LORRAN ANTÔNIO LINO

**ESTUDO, DESENVOLVIMENTO E IMPLEMENTAÇÃO DE
UM MANIPULADOR ROBÓTICO CINCO DOF.**

LAVRAS-MG

2019

LORRAN ANTÔNIO LINO

**ESTUDO, DESENVOLVIMENTO E IMPLEMENTAÇÃO DE UM MANIPULADOR
ROBÓTICO CINCO DOF.**

Trabalho de conclusão de curso apresentado à
Universidade Federal de Lavras, como parte
das exigências do Curso de Engenharia de
Controle e Automação, para a obtenção do
título de Bacharel.

Prof. Dr. Leonardo Silveira Paiva

Orientador

LAVRAS-MG

2019

**Ficha catalográfica elaborada pelo Sistema de Geração de Ficha Catalográfica da Biblioteca
Universitária da UFLA, com dados informados pelo(a) próprio(a) autor(a).**

Lino, Lorrان Antônio.

Projeto de um manipulador robótico com cinco graus de
liberdade e uma garra para objetos cilíndricos / Lorrان Antônio

Lino. - 2019.

75 p. : il.

Orientador(a): Leonardo Paiva.

TCC (graduação) - Universidade Federal de Lavras, 2019.

Bibliografia.

1. Robótica. 2. Manipulador robótico. 3. Sistemas embarcados.
I. Paiva, Leonardo. II. Título.

LORRAN ANTÔNIO LINO

**ESTUDO, DESENVOLVIMENTO E IMPLEMENTAÇÃO DE UM MANIPULADOR
ROBÓTICO CINCO DOF.**

**STUDY, DEVELOPMENT AND IMPLEMENTATION OF A ROBOTIC
MANIPULATOR FIVE DOF**

Trabalho de conclusão de curso apresentado à
Universidade Federal de Lavras, como parte
das exigências do Curso de Engenharia de
Controle e Automação, para a obtenção do
título de Bacharel.

Prof. Dr. Leonardo Silveira Paiva. UFLA
Prof. Dr. Fabio Domingues de Jesus. UFLA
Ma. Thais Martins Mendes. UFLA

Prof. Dr. Leonardo Silveira Paiva

Orientador

LAVRAS-MG

2019

Dedico em primeiro lugar a Deus por me motivar e não deixar eu desistir nas horas mais difíceis, por me dar forças e sabedoria.

Dedico a minha mãe Maria por estar ao meu lado e ser exemplo de perseverança, que mesmo com as maiores dificuldades não nada faltar e sempre me apoiou nas minhas escolhas.

Dedico ao meu pai Adilson por não deixar faltar nada e me auxiliar em tudo que precisei.

AGRADECIMENTOS

A Deus, por ser minha maior fonte de força, perseverança e a base de minhas conquistas.

Aos meus pais, por não deixar faltar nada mesmo nos momentos mais difíceis.

À Universidade Federal de Lavras, especialmente ao Departamento de Automática, pela oportunidade.

Ao Leonardo Paiva pela orientação prestada na elaboração deste trabalho, pelo apoio e principalmente paciência

Aos professores, pela dedicação proporcionada durante o curso, que me auxiliou com seus conhecimentos e sempre me instruiu de forma que pudesse me tornar um bom profissional.

Aos técnicos administrativos e funcionários do DEG, DAT e DCC/ UFLA que me ajudaram de alguma forma na construção do protótipo.

Ao Núcleo de estudos em eficiência energética por me ajudar a adquirir conhecimentos práticos e experiências com sistemas eletrônicos.

A todos os colegas de universidade, pois muitos se tornaram amigos.

MUITO OBRIGADO!

“O sucesso nasce do querer, da determinação e persistência em se chegar a um objetivo. Mesmo não atingindo o alvo, quem busca e vence obstáculos, no mínimo fará coisas admiráveis. ” (Jose de Alencar)

RESUMO

Este trabalho visa o estudo, desenvolvimento e implementação de um manipulador do tipo braço mecânico antropomórfico com cinco graus de liberdade e uma garra para manipular objetos ou ferramentas, acionados por um conjunto de sete servomotores de alto torque. A sua estrutura é feita em madeira resistente e o seu controle é por meio de um sistema embarcado contendo um microcontrolador atmega328p disposto na placa ARDUINO, com dois modos de operações, automático e manual. No modo automático, o manipulador é responsável por detectar um objeto por meio de um sensor ultrassônico e, com movimentos pré-programados, mover o objeto de um ponto ao outro dentro do seu espaço de trabalho. Já no modo manual é utilizado o módulo ESP8266 para o controle por meio de dispositivos conectados à rede WI-FI. Este manipulador posteriormente será utilizado para colheita de frutas junto com um robô móvel. Aplicando assim os conceitos e conhecimentos adquiridos no curso de engenharia de controle e automação.

Palavras-chave: Manipulador. Microcontrolador. Robô. Servomotor.

ABSTRACT

This work aims at the study, development and implementation of an anthropomorphic mechanical arm type manipulator with five Degrees of Freedom and a claw to manipulate object or tools, driven by a set of seven high torque servomotors. Its structure is made of sturdy wood and its control is by means of the embedded system containing atmega328p microcontroller arranged in the ARDUINO board, with two modes of operations, automatic and manual. In automatic mode, the manipulator is responsible for detecting an object by means of an ultrasonic sensor and, with preprogrammed movements, move the object from one point to another within its workspace. In manual mode, the ESP8266 module is used for control via devices connected to the WI-FI network. This manipulator will later be used for picking fruit along with a mobile robot. Applying the concepts and knowledge acquired in the course of control and automation engineering

Keywords: Manipulator, Microcontroller, Robot, Servomotor.

LISTA DE FIGURAS

Figura 1– Exemplo de manipulador utilizado nas indústrias	18
Figura 2 - Robô agrícola autônomo voltado para a colheita de aspargos.....	20
Figura 3 - Manipulador cartesiano.....	21
Figura 4 - Manipulador cilíndrico	21
Figura 5 - Manipulador esférico	21
Figura 6 - Manipulador ACARA.....	21
Figura 7 - Manipulador Paralelo.....	21
Figura 8 - Manipulador articulado.....	21
Figura 9 - Tipos de juntas inferiores.....	22
Figura 10 - Órgão terminal do tipo garra de dois dedos.....	23
Figura 11 - Garra Articulada	24
Figura 12 - Robô Motoman com ponta de solda por arco ao punho como efetuator.	24
Figura 13 - Servomotores com seus sistemas de engrenagens.	26
Figura 14 - Representação dos sinais e as respectivas posições de um servomotor.....	27
Figura 15 - Fluxograma da metodologia aplicada.....	29
Figura 16 - Parte lateral e frontal da base do manipulador.....	30
Figura 17 - Parte lateral e frontal da base do manipulador.....	30
Figura 18 - Chapa superior da base do manipulador	30
Figura 19 - Chapa superior da base do manipulador fabricada	30
Figura 20 - Parte trazeira da base do manipulador.	30
Figura 21 - Parte trazeira da base do manipulador.	30
Figura 22 - Base completa do manipulador	31
Figura 23 - Base completa do manipulador fabricada.....	31
Figura 24 - Ombro do manipulador.....	31
Figura 25 - Ombro fabricado do manipulador.....	31
Figura 26 - Braço do manipulador.....	31
Figura 27 – Braço fabricado	31
Figura 28 - Chapas do antebraço do manipulador.	31
Figura 29 – Antebraço fabricado	31
Figura 30 - Chapas do punho do manipulador	32
Figura 31 – Primeira parte do punho projetado	32
Figura 32 - Segunda peça que compõem o punho.....	32
Figura 33 – Segunda parte do punho fabricado.....	32
Figura 34 - Garra do manipulador	32
Figura 35 – Garra projetada.....	32
Figura 36 - Vista lateral do manipulador montado por meio de software.....	34
Figura 37 - Vista superior do manipulador montado por meio de software.....	35
Figura 38 - Vista frontal do manipulador montado por meio de software	36
Figura 39 - Placa de um ARDUINO UNO.....	37
Figura 40 - Parte do <i>DATASHEET</i> do servo motor.....	38
Figura 41 - Diagrama de funcionamento do HC-SR04	39

Figura 42 - Funcionamento do Sensor ultrassônico.	39
Figura 43 – Módulo ESP8266 com soquete para uso em placas PCB	40
Figura 44 - Especificações gerais do ESP8266	40
Figura 45 - Circuito montado no <i>PROTEUS</i>	41
Figura 46 - Código para a simulação.....	42
Figura 47 - Protótipo montado com seus 5 Graus de Liberdade e a garra	43
Figura 48 - Vista lateral do protótipo	43
Figura 49 - Espaço de trabalho do manipulador desenvolvido	43
Figura 50 – Marcações do espaço de trabalho.....	44
Figura 51 - Parte do algoritmo para as posições iniciais das unhas do manipulador	47
Figura 52 - Parte do algoritmo utilizados no teste 1	48
Figura 53 – Continuação do algoritmo para o teste 1	48
Figura 54 – Parte do algoritmo contendo a sequência de movimentos realizados.	50
Figura 55 – Continuação do algoritmo contendo a sequência de movimentos realizados	50
Figura 56 – Parte do algoritmo utilizado na terceira etapa do teste 1	51
Figura 57 – Parte do algoritmo utilizado para o teste 2.....	53
Figura 58 – Parte do algoritmo utilizado nesta segunda etapa do teste 2.....	54
Figura 59 – Continuação do algoritmo utilizado nesta segunda etapa do teste 2.....	55
Figura 60 – Parte do algoritmo utilizado na terceira etapa do teste 2	56
Figura 61 – Pagina Web criada.....	57

LISTA DE TABELAS

Tabela 1 - Configuração física dos manipuladores	21
Tabela 2 – Peças desenhadas e fabricadas (Continua)	30
Tabela 3 - Sequência de movimentos para as posições iniciais	46
Tabela 4 - Sequência de movimentos para pegar o objeto	47
Tabela 5 – Sequência de movimentos realizados para mover o objeto até o ponto desejado e transferir o líquido do objeto para o copo.....	49
Tabela 6 – Sequência de movimentos para descartar o objeto vazio	50
Tabela 7 – Sequência de movimentos para pegar o objeto.....	52
Tabela 8 - Sequência de movimentos realizados para mover o objeto até o ponto desejado e transferir o líquido do objeto para o copo.....	53
Tabela 9 – Sequência de movimentos para descarte	55
Tabela 10 - Custos para o desenvolvimento do manipulador.....	61

LISTA DE SIGLAS

IDE – Integrated Development Environment

ISO – International Organization for Standardization

RIA – Robotic Industries Association

SCARA – Selective Compliance Assembly Robot Arm

VAA – Veículo Autônomo Agrícola

GPS – Global Positioning System

IFR – International Federation of Robotics

DOF – Degrees-Of-Freedom

PWM – Pulse Width Modulation

AC – Alternating Current

DC – Direct Current

WI-FI – Wireless Fidelity;

USB – Universal Serial Bus

RAM – Random Access Memory

GPIOs – General Purpose Input/Output

TCP – Transmission Control Protocol

GND – graduated neutral density filter

SUMÁRIO

1	INTRODUÇÃO.....	15
2	OBJETIVOS	15
2.1	Objetivo geral	15
2.2	Objetivos específicos	16
3	REFERENCIAL TEÓRICO	16
3.1	Robótica.....	16
3.2	Industria e Agricultura	18
3.3	Manipuladores.....	20
4	MATÉRIAS E MÉTODOS.....	28
4.1	Desenvolvimento	29
4.2	RESULTADO E DISCUSSÃO.....	40
4.2.1	Teste 1	45
4.3.2	Teste 2	51
4.2.2	Teste 3	56
4.2.3	Análise dos resultados obtidos	58
4.4	Limitações e Trabalhos Futuros	59
4.5	Custos	61
5	CONCLUSÃO.....	61
	REFERENCIAS	63
	APÊNDICE A.....	65
	APÊNDICE B.....	66
	APÊNDICE C.....	74

1 INTRODUÇÃO

Desde do início da evolução, o ser humano vem buscando desenvolver ferramentas ou utensílios que auxiliam na realização de tarefas do cotidiano. E assim veio surgindo os conhecimentos relacionados com mecânica, elétrica, acionamento, entre outros, e a busca por novas tecnologias. O interesse por tecnologia foi aumentando cada vez mais, e a partir da revolução industrial, com o surgimento da máquina a vapor, a sociedade investe cada vez mais nesta área.

Apesar da robótica ser uma ciência relativamente nova, há indícios de desenhos e mecanismos robóticos a bastante tempo, como é o caso dos projetos de Da Vinci de 1495 (KUTEKEN, 2014). O termo robô surgiu em 1920 em uma peça teatral e a partir de então o fascínio por esta área cresceu cada vez mais. A robótica é uma combinação de várias áreas como: mecânica, computação, eletrônica, elétrica e automação. Atualmente é possível encontrar a aplicação da robótica nos mais diversos campos aplicações como: agricultura, medicina, aeronáutica, astronomia, química, em indústrias e até mesmo dentro de residências e com o avanço tecnológico, que está possibilitando a criação de processadores mais rápidos e eficientes, sensores mais precisos e materiais mais leves, a aplicação da robótica tende a crescer ainda mais. Dentro da robótica se encontram os manipuladores robóticos, que são definidos como um sistema multifuncional reprogramável capaz de mover, posicionar e manipular peças, objetos ou ferramentas, que estão presentes principalmente no âmbito industrial, onde buscam realizar tarefas de forma precisa e eficiente.

O presente trabalho irá descrever todo o embasamento teórico e prático para a construção de um manipulador, levando em consideração os aspectos principais como cinemática, área de trabalho, acionamento, controle, tipo de programação do braço manipulador e no caso o *software* utilizado para elaboração do programa. Sendo, portanto, o intuito do presente trabalho a aplicação dos conhecimentos e experiências adquiridos durante o curso.

2 OBJETIVOS

2.1 Objetivo geral

Este trabalho tem como objetivo desenvolver um manipulador com cinco Graus de Liberdade que seja capaz de alcançar e mover objetos entre vários pontos dentro do seu espaço

de trabalho para posteriormente ser acoplado a uma plataforma móvel e ser utilizado em diferentes atividades, como por exemplo, colheita em ambientes externos.

2.2 Objetivos específicos

Dentre os objetivos específicos, podem ser citados:

- Simular o circuito eletrônico do manipulador por meio do *software* *PROTEUS*;
- Projetar os componentes mecânicos que compõem o manipulador por meio do *software* *SOLIDWORK*;
- Confeccionar a estrutura mecânica do manipulador e o seu espaço de trabalho;
- Desenvolver o sistema embarcado e a programação de controle do manipulador por meio da IDE do *ARDUINO*;

3 REFERENCIAL TEÓRICO

Neste tópico serão apresentados os conceitos relacionados ao trabalho desenvolvido. Toda a área da robótica relacionada com manipuladores, os mecanismos no qual permitem a movimentação, a estrutura mecânica, o espaço de trabalho e suas cinemáticas.

3.1 Robótica

Há várias definições do que é um robô, na definição da ISO 8373 (VALÉIRIO, 2014), “Um robô industrial é um manipulador de multiuso que seja programável, com controle automático e que possui 3 ou mais eixos, podendo ser fixo ou móvel para utilização em aplicações de automação industrial”

Já na definição dita oficial da Associação das Industrias de Robótica (*Robotic Industries Association - RIA*) (PAZOS, 2002): “Um robô é um manipulador reprogramável, multifuncional, projetado para mover matérias, peças, ferramentas ou dispositivos especiais em movimentos variáveis programadas para a realização de uma variedade de tarefas”.

Na definição moderna dada por meio dos dicionários Houaiss (HOUAISS, 2000)., “Um robô é uma máquina, autômato de aspecto humano, capaz de se movimentar e agir, com mecanismos de comando de controle automático e que possa realizar tarefas repetitivas”

Abaixo, segundo Kruteken, Schirmer, Souza e Pscheidt, é apresentado a evolução dos robôs, de forma cronológica citando os grandes acontecimentos relacionados:

1495 – Desenhado o primeiro Cavaleiro mecânico Por Leonardo Da Vinci;

1801 - J. Jacquard inventou um tear programável;

1959 – Primeiro robô comercial introduzido pela Planet Corporation;

1960 – Desenvolvido o primeiro robô Unimate com controle numérico e hidráulico;

1971 – O desenvolvimento do braço de Stanford com atuação elétrica, produzido na universidade de Stanford.

1978 – Produção do manipulador PUMA pela Unimation

1979 – Desenvolvimento do manipulador SCARA por meio da Universidade de Yamanashi;

1981 - Desenvolvimento de robô com atuadores diretos, por meio de Carnegie Mellon.

1983 – Projeto de linha flexível de montagem automatizada com a utilização de robôs

1997 – A Honda anuncia primeiro robô humanoide que sobe escadas

2000 – Desenvolvimento do robô móvel para realizar expedição na Antártida, por Carnegie Mellon;

2001 – Desenvolvimento dos robôs humanoides Hoap-1 por Fujitsu

2003 – Introdução do robô KUKA Robocoaster o primeiro que transporta passageiros,

2010 – A NASA envia o primeiro robô humanoide para o espaço, o R2.

Atualmente – Desenvolvimento de robôs indústrias mais sensíveis e com maior precisão;

Os robôs possuem inúmeras aplicabilidades, sendo utilizados em diversas áreas, pois possuem grandes vantagens tais como: aumento de produtividade, capacidade de operar em ambientes hostis ou com materiais perigos, maior precisão em relação aos humanos, entre outras. No âmbito industrial os robôs são mais utilizados, mas com o avanço das tecnologias pode-se encontrar robôs em vários ambientes realizando diversas tarefas. Neste tópico serão apresentados alguns exemplos de aplicações da robótica na indústria e na agricultura, onde a sua utilização vem tornando cada vez mais essencial.

3.2 Indústria e Agricultura

A aplicação da robótica na indústria é essencial e vem se tornando cada vez mais comum, devido à capacidade de exercer vários tipos de atividades produtivas como: transporte de objetos, posicionamento de ferramentas de trabalho em locais que trazem risco ao homem, entre outros. As grandes vantagens com a aplicação da robótica no âmbito industrial são: repetitividade, precisão, rapidez, segurança e otimização dos processos. Um exemplo de robô utilizado nas indústrias é o veículo guiado automaticamente (*Automatic Guided Vehicle - AVGs*), que realiza o transporte de materiais para a linha de montagem (NOGUEIRA; TEXEIRA; FREITAS, 2015). Outros tipos de robôs utilizados nas indústrias são os manipuladores que podem realizar diversas tarefas. Na Figura 1 é mostrado um exemplo de aplicação de robôs manipuladores na indústria.

Figura 1– Exemplo de manipulador utilizado nas indústrias



Fonte: SOUZA, 2005

A robótica aplicada na área da agricultura é cada vez mais utilizada, apesar de ser recente devido as dificuldades encontradas com sua utilização no campo como: solo irregular, umidade, chuvas, alimentação dos componentes eletrônicos, entre outras. Entretanto o grande avanço da tecnologia voltada para sensores, motores, materiais e baterias de longa duração, possibilitou a sua utilização. Os robôs mais utilizados nessa área são: os robôs móveis, os veículos autônomos

e robôs de colheitas. Mas pode-se encontrar manipuladores que atuam dentro de estufas e na parte de análise de plantas e solos. (HACKENHAAR; HACKENHAAR; ABREU, 2014).

Com os gastos energéticos para a produção de alimentos aumentando cada vez mais, a robótica é aplicada na agricultura de forma a tentar diminuí-los, já que a eficiência e praticidade dos robôs ultrapassou a dos humanos (TANKE, 2011).

Outro fator que influencia na aplicação da robótica na agricultura é a produção sustentável, que por meio de técnicas e instrumentos buscam a menor degradação do meio ambiente e a produtividade com sustentabilidade (CRESTANA, 2012).

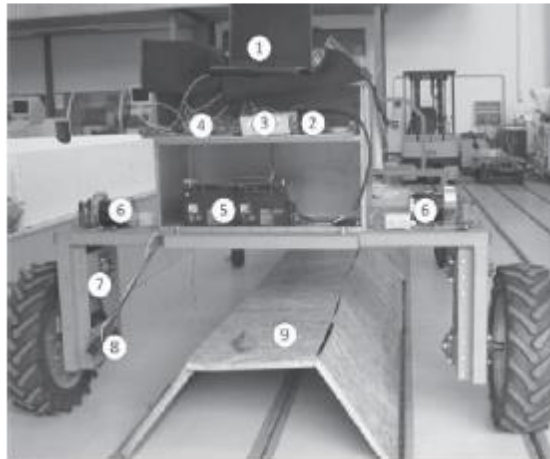
Os robôs moveis são robôs que possuem algum modo de locomoção sejam rodas, pernas, lagartas ou esteiras, além disso, possuem assistência inteligente para a locomoção ou controle dos sistemas (HACKENHAAR; HACKENHAAR; ABREU, 2014). Estes robôs são utilizados na agricultura para a navegação no campo com diversas finalidades, pois geralmente são robôs leves e não destroem o plantio e o solo. O primeiro robô utilizado na agricultura de campo foi projetado em 1998 por Astrand e Baerveldt e teve o objetivo de controlar as ervas daninha do plantio (ASTRAND; BAERVELDT, 2002). Um outro exemplo de robô utilizado na agricultura é o robô colhedor de frutas, que é a união de um robô móvel que possui uma visão artificial para rastrear toda a plantação e oito braços robóticos para colher as frutas. (INOVAÇÃO TECNOLÓGICA,2007)

O Robô AgBot desenvolvido na Universidade de Luisiana tem a função de fertilização do solo e plantio de sementes. Tal robô utiliza um sistema de GPS que traça um mapa da terra e por meio de um sistema de identificação sensorial que marca as estruturas. Este robô é alimentado por energia solar por meio de placas fotovoltaicas e também é utilizado como robô segurança, em que possui uma câmera de visão noturna montada sobre uma base giratória (360°), um detector de movimentos e um sistema de alarme. (MOON, 2008).

Os veículos autônomos que são utilizados na agricultura são os tratores, colheitadeiras, entre outros que possuem um sistema de navegação e controle totalmente automático e mais sofisticados, pois estes veículos operam em áreas externas e desestruturadas. Os veículos autônomos utilizam várias teorias de controle robótico tornando-se assim eficiente nas aplicações agrícolas.

Um exemplo de robô utilizado na agricultura é desenvolvido por Dong, utilizado para colher aspargos, conforme a Figura 2. Este robô autônomo possui uma plataforma móvel e é capaz de detectar o posicionamento da planta por meio de sensores ultrassônicos, obtendo um maior rendimento computacional devido ao menor processamento requerido para a coleta de dados e redução dos custos de investimento (BATISTA,2016).

Figura 2 - Robô agrícola autônomo voltado para a colheita de aspargos



Fonte: BATISTA, 2016

Nas estufas são utilizados tanto os robôs móveis quanto os robôs fixos, responsáveis por fazer o controle das plantas dentro do ambiente. O Agrobot, que é uma plataforma móvel com uma visão estereoscópica e um braço manipulador com uma pinça para o controle de pragas nas estufas (HACKENHAAR; HACKENHAAR; VIEIRA,2014).

3.3 Manipuladores

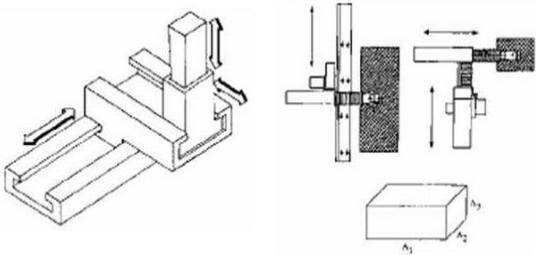
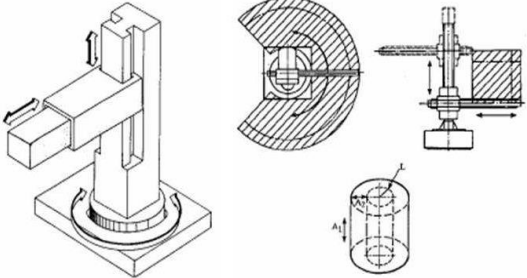
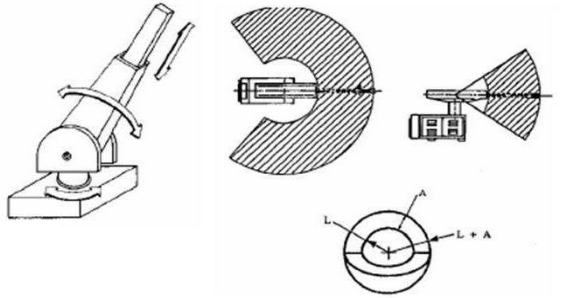
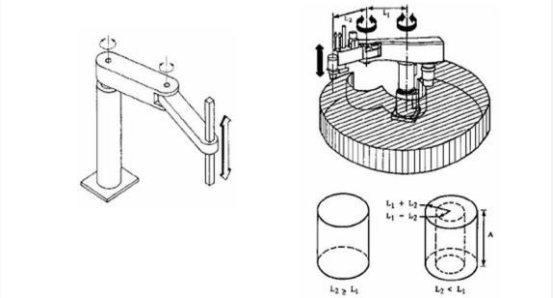
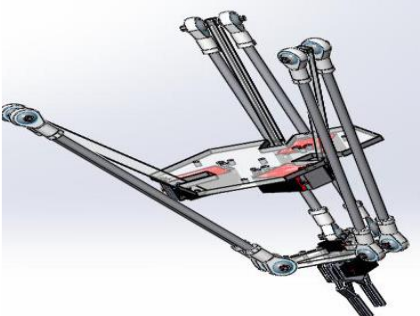
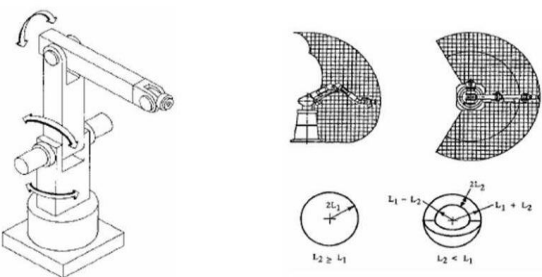
De acordo com *RIA* (Robotic Industries Association) (PIMENTA, 2009), um manipulador robótico apresenta a seguinte definição: “ Sistema multifuncional reprogramável capaz de mover, posicionar e manipular peças, objetos, ferramentas ou dispositivos especiais ” (PIMENTA, 2009).

Já segundo a definição da (ISO-10218), “Um manipulador é uma máquina com vários graus de liberdade, controlada automaticamente, reprogramável, multifuncional, que pode ter base fixa ou móvel com aplicação na automação industrial”.

Outra possível definição é dada por ROSARIO,2005: “Um braço mecânico é um manipulador projetado para realizar diversas tarefas e ser capaz de repeti-las. Para executá-las, o robô move partes, objetos, ferramentas ou dispositivos especiais segundo movimento e pontos pré-programáveis”.

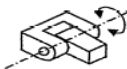

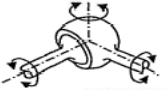
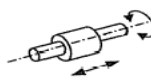
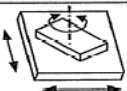
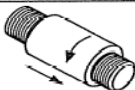
As configurações físicas de um manipulador estão relacionadas com os tipos de juntas que este possui (CARRARA, 1999). De acordo com a federação Internacional de robótica (*International Federation of Robotics - IFR*), as principais configurações básicas quanto a estrutura mecânica dos manipuladores são: manipulador cartesiano, manipulador cilíndrico, manipulador esférico, manipulador SCARA, manipulador articulado ou antropomórfico e manipulador paralelo. A tabela 1 ilustra os principais tipos de manipuladores.

Tabela 1 - Configuração física dos manipuladores

<p>Figura 3 - Manipulador cartesiano.</p> 	<p>Figura 4 - Manipulador cilíndrico</p> 
<p>Figura 5 - Manipulador esférico</p> 	<p>Figura 6 - Manipulador ACARA</p> 
<p>Figura 7 - Manipulador Paralelo</p> 	<p>Figura 8 - Manipulador articulado</p> 

As juntas de um manipulador são responsáveis por ligar os elos e possibilitar os seus movimentos. Segundo Reuleaux (FERREIRA; IPUCHIMA; SILVA, 2012) existem seis tipos de juntas cinemática inferiores que são mostradas na Figura 9.

Figura 9 - Tipos de juntas inferiores.

JUNTAS CINEMÁTICAS INFERIORES			
Designação	Geometria	Símbolo	Graus de liberdade
Rotação		R	1
Translação ou prismática		T	1
Esférica ou globular		E	3
Cilíndrica		C	2
Plana		P	3
Helicoidal ou de parafuso		H	1

Fonte: FLORES, 2007.

Em um manipulador, são utilizados apenas dois tipos de juntas as prismáticas e as rotacionais.

A maioria dos manipuladores têm dois grupos de juntas: juntas principais (localizada mais próxima da base) que, geralmente são três, e a juntas do punho que são utilizadas para a movimentação do *End Efeitor* (FERREIRA; IPUCHIMA; SILVA, 2012).

O Grau de Liberdade (*Degrees-Of-Freedom - DOF*) de um manipulador refere-se aos movimentos independentes que cada articulação poderá realizar dentro do espaço cartesiano. A definição do Grau de Liberdade é: “O número de movimentos de acionamento que um determinado mecanismo necessita para que a localização dos seus corpos seja completamente conhecida em relação a um referencial predefinido” corresponde ao número de atuadores necessários para realizar o movimento de um referencial até o ponto desejado (FERREIRA; IPUCHIMA; SILVA, 2012).

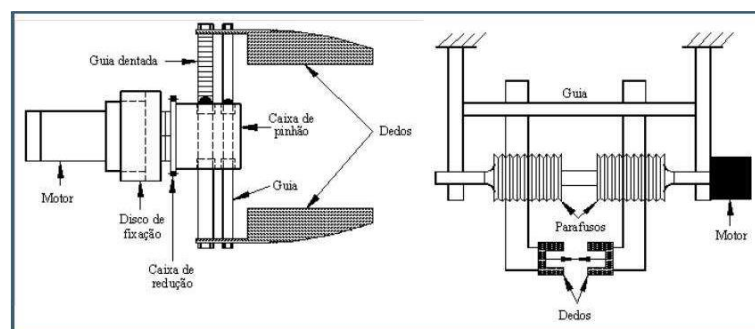
Geralmente o Grau de Liberdade de um manipulador é igual ou maior que o número de articulações que este possui. Quando a articulação realiza apenas um movimento relativo em relação ao eixo, esta articulação possui apenas um Grau de Liberdade. Já quando a articulação realiza dois movimentos em relação aos eixos, esta articulação possui dois Graus de Liberdade (FERREIRA; IPUCHIMA; SILVA, 2012).

Como visto anteriormente existe vários tipos de juntas e para cada existe o seu Grau de Liberdade. Nas juntas prismáticas (P), rotacionais (R) e helicoidais (H) tem-se um Grau de Liberdade, já as juntas esféricas possuem três Graus de Liberdades (Pimenta,2009). Ademais, conhecer o Grau de Liberdade de um manipulador é essencial para análise e síntese da cinemática do manipulador.

Os órgãos terminais ou *End-effector* dos manipuladores são mãos, garras ou ferramentas conectadas ao pulso e são responsáveis por realizarem determinado tipo de tarefa, seja manipular objetos de vários tipos e tamanhos, furar, soldar, entre outros. Os órgãos terminais são geralmente projetados para realizar uma tarefa específica, mas também podendo ter órgãos terminais que realize vários tipos de tarefa (Groover, 1989). Existem uma gama de variedades de órgãos terminais, podendo ser classificados em duas categorias: Garras e Ferramentas

As garras são efetadores responsáveis por pegar, segurar e transportar objetos dentro do espaço de trabalho do manipulador. Geralmente utilizam os princípios mecânicos para segurar um objeto por meio de dedos que exercem uma forma contra o objeto para segurá-lo (Groover, 1989). Um exemplo desse efetador é a garra de dois dedos, mostrada na Figura 10, é um órgão terminal mais simples onde possui movimentos paralelos ou rotacionais e tem pouca versatilidade na manipulação de objetos, pois existe uma limitação na abertura dos dedos limitando assim o dimensionamento do objeto a ser manipulado (Carrara, 1999).

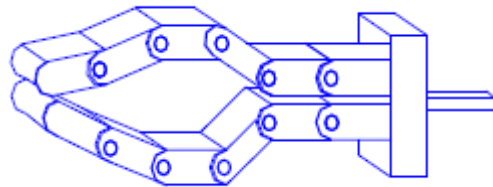
Figura 10 - Órgão terminal do tipo garra de dois dedos.



Fonte: PAZOS, 2002.

Outro exemplo de órgão terminal do tipo garra são as garras articuladas que se assemelham a mão humana, podem manipular objetos de forma irregular, pois possuem maior versatilidade possuindo maior quantidade de elos (Carrara, 1999). A Figura 11 mostra um exemplo de uma garra articulada.

Figura 11 - Garra Articulada



Fonte: CARRARA,2008

As ferramentas são utilizadas como órgãos terminais, fixadas no último elo ou punho do manipulador. São utilizados para realizar algum tipo de tarefa como: furar, soldar, pinta, parafusar, recortar, entre outros (FERREIRA; IPUCHIMA; SILVA, 2012). A Figura 12 mostra um exemplo de manipulador com esse tipo de órgão terminal.

Figura 12 - Robô Motoman com ponta de solda por arco ao punho como efetuador.



Fonte: MOTOMAN ROBOT, 2002.

Outros conceitos relacionados à teoria de manipuladores essenciais para o estudo e desenvolvimento dos mesmos são relacionados ao espaço de trabalho, cinemática, dinâmica e sistema de acionamento.

O espaço de trabalho ou volume de trabalho de um manipulador é o termo que está relacionado com o espaço onde este pode posicionar seu punho. O espaço de trabalho de um manipulador é todos os pontos possíveis de ser alcançado pelo punho do manipulador (PAZOS, 2002).

A cinemática de um manipulador é o estudo dos movimentos realizados por este sem levar em conta as forças exercidas para tais movimentos. Para tratar dos movimentos é necessário desenvolver mecanismos para representar a posição de determinado ponto do manipulador no tempo. A representação depende da posição das juntas e dos elos, e tendo a base do manipulador como ponto de referência. Para se obter a cinemática de um manipulador é preciso ter os conhecimentos de geometria, trigonometria e cálculo vetorial, independentemente do tipo de manipulador. Tendo que se considerar caso a caso, pois não existe uma regra para determinação da cinemática dos manipuladores (Groover, 1989).

Existem dois métodos para conhecer os movimentos de um manipulador. Quando se deseja calcular a posição cartesiana no espaço, bem como a orientação do punho, com base nos conhecimentos dos ângulos das juntas, é utilizado a cinemática direta. Já quando se deseja calcular as posições angulares a partir da posição no espaço utiliza-se a cinemática inversa (FERREIRA; IPUCHIMA; SILVA, 2012).

A dinâmica de um manipulador permite relacionar a evolução temporal da sua configuração (posição, características e movimentos) com as forças e momentos que ele exerce e atuam sobre este (LOPEZ, 2002). O comportamento dinâmico de um manipulador pode ser descrito por um conjunto de equações diferenciais chamadas de equação dinâmica de movimento e estão relacionadas com as energias, as forças, os momentos, os atritos e as coordenadas generalizadas (ROMANO, 2005).

Conhecer o comportamento dinâmico de um manipulador é de extrema importância, pois as forças que este deve realizar interferem diretamente na precisão dos movimentos, assim quando se deseja controlar o robô, estas forças têm que ser levada em consideração (LOPEZ, 2002).

O sistema de acionamento de um manipulador refere-se aos seus atuadores que serão responsáveis por realizar a movimentação das juntas, podem ser caracterizados como: servo motor e motor de passos

O servo motor é um dispositivo eletrônico que recebe um sinal elétrico em sua entrada e posiciona o seu eixo em uma determinada posição angular. Os seus movimentos produzidos são circulares, porem limitados de 0° a 180° (padrão). Por serem pequenos e compactos e alta precisão no posicionamento do seu eixo, são bastante utilizados na robótica. São compostos por circuito de controle, motor, sistema de engrenagens, potenciômetro e caixa (FERREIRA; IPUCHIMA; SILVA, 2012).

- O circuito de controle é responsável por distribuir a energia correta para os demais componentes e controlar a posição por meio da comparação do valor PWM com a posição do potenciômetro (FERREIRA; IPUCHIMA; SILVA, 2012);
- O motor elétrico é responsável por converter a energia elétrica em energia mecânica, e é utilizado no servo motores para realizar a movimentação do seu eixo. Existem dois tipos motores utilizados nos servos, os de corrente continua (DC) e os de corrente alternada (AC), sendo os de corrente continua utilizados na maioria dos servos (FERREIRA; IPUCHIMA; SILVA, 2012);
- O sistema de engrenagem é responsável por aumentar a força e diminuir a alta velocidade entregue por meio do motor. A relação de transmissão afetará esses itens, ou seja, quanto maior a relação de transmissão, maior será a relação entre velocidade e força. As engrenagens podem ser metálicas, que proporciona maior força, menos desgastes e maior precisão, ou de plástico, que são mais baratas e leves (FERREIRA; IPUCHIMA; SILVA, 2012). A Figura 13 apresenta dois tipos de servomotores com suas respectivas engrenagens.

Figura 13 - Servomotores com seus sistemas de engrenagens.

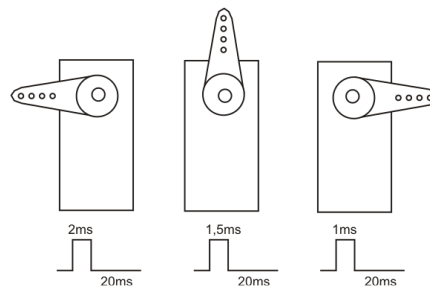


Fonte: FERREIRA, IPUCHIMA, SILVA, 2012

- O potenciômetro é acoplado ao eixo de saída e é utilizado para definir a direção e a velocidade de giro do motor para comparar com a sua posição atual. O valor de sua resistência elétrica indica a posição angular em que se encontra o eixo (FERREIRA; IPUCHIMA; SILVA, 2012).
- A caixa envolve e protege todo o mecanismo do servo motor. Portanto, deve ser rígida para suportar os grandes esforços mecânicos que podem ser produzidos e fatores externos que podem interferir no seu funcionamento (FERREIRA; IPUCHIMA; SILVA, 2012).

O funcionamento do servo se dá por meio da modulação por largura de pulso (PWM) aplicado na sua entrada, posicionando o seu eixo conforme este valor. Estes pulsos têm largura pré-determinada que varia de 1×10^{-3} a 2×10^{-3} segundos. Podendo assim variar a posição do eixo do servo motor por meio do valor de largura de pulso (FERREIRA; IPUCHIMA; SILVA, 2012), conforme mostrado na Figura 14.

Figura 14 - Representação dos sinais e as respectivas posições de um servomotor.



Fonte: FERREIRA, IPUCHIMA, SILVA, 2012

Na figura acima:

- O sinal de 1 ms corresponde a posição 0° do seu eixo;
- O sinal de 1,5 ms corresponde a posição 90° do seu eixo;
- O sinal de 2 ms corresponde a posição 180° do seu eixo.

Os motores de passos são dispositivos eletromecânicos que convertem sinais elétricos (pulsos) em movimentos mecânicos que geram variações angulares discretas. O eixo do motor é racionado em pequenos incrementos angulares, denominados “passos”. A rotação do eixo está

relacionada com os pulsos elétricos e a sequência de tais pulsos aplicados nos terminais do motor. Como possui um controle preciso dos movimentos e muito utilizado na robótica (BRITES; SANTOS, 2008).

O funcionamento consiste no uso de solenoides alinhados de dois a dois que quando energizados atraem o rotor fazendo-o se alinhar com o eixo determinado pelos solenoides, causando pequenas variações de ângulo denominados passo. A velocidade e o sentido de movimento são determinados pela forma como cada solenoide é ativada (BRITES; SANTOS, 2008).

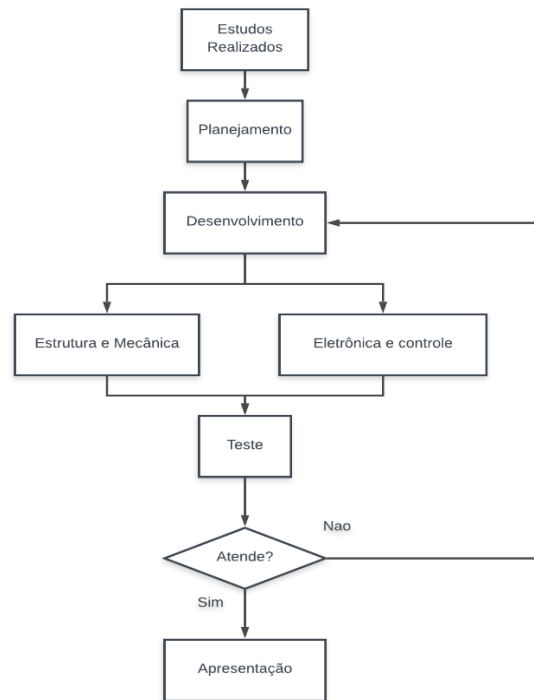
As vantagens de se utilizar um motor de passo são: alta precisão no seu posicionamento, maior precisão no torque aplicado, excelente resposta de aceleração e desaceleração e a maior movimentação angular do seu eixo. Já as desvantagens são: baixo desempenho em altas velocidades, complexidade de operação e ocorrência de ressonância por controle inadequado (BRITES; SANTOS, 2008).

4 MATÉRIAS E MÉTODOS

Nesta seção serão apresentados os métodos que foram utilizados para o desenvolvimento do manipulador, sua parte estrutural, mecânica, eletrônica e de controle. E também serão apresentadas as suas características e os cálculos realizados com base nas teorias apresentadas.

O diagrama de fluxo que representa a metodologia aplicada é representado na Figura 15

Figura 15 - Fluxograma da metodologia aplicada



Fonte: Do autor

Conforme mostrado no fluxograma, primeiro iniciou-se realizando o estudo do projeto a fim de se obter um melhor desempenho no seu desenvolvimento, buscando introduzir os conceitos adquiridos no curso de engenharia de controle e automação. Avaliou-se os custos para o desenvolvimento do projeto, ou seja, o preço da matéria prima e dos componentes que foram utilizados.

No planejamento, foi elaborado um cronograma para melhor controle e execução no desenvolvimento do projeto, onde estão definidas as atividades realizadas e seus respectivos prazos. Este cronograma pode ser visto no Apêndice A.

4.1 Desenvolvimento

Nesta seção serão apresentados os métodos para o desenvolvimento e construção do protótipo, desde os desenhos e simulação até a montagem e fabricação do manipulador e sua parte eletrônica.

A estrutura mecânica do manipulador é constituída de base, ombro, braço, antebraço, punho e garra. Foram desenhadas cada parte do manipulador utilizando o software *SOLIDWORK* para se ter um melhor desempenho no seu desenvolvimento e foi feita a fabricação de cada parte em madeira. Na Tabela 2 são apresentadas as peças fabricadas e seus respectivos desenhos feitos.

Tabela 2 – Peças desenhadas e fabricadas (Continua)

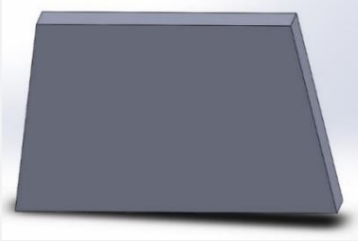

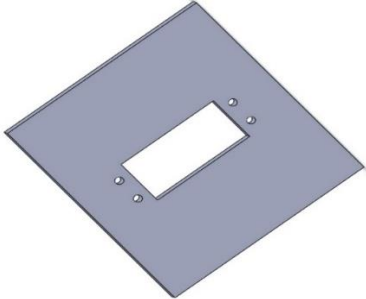



PEÇAS DESENHADAS VIA SOFTWARE	PEÇAS FABRICADAS
<p data-bbox="252 701 825 790">Figura 16 - Parte lateral e frontal da base do manipulador</p> 	<p data-bbox="869 701 1442 790">Figura 17 - Parte lateral e frontal da base do manipulador</p> 
<p data-bbox="293 1068 786 1158">Figura 18 - Chapa superior da base do manipulador</p> 	<p data-bbox="906 1068 1406 1158">Figura 19 - Chapa superior da base do manipulador fabricada</p> 
<p data-bbox="301 1509 772 1599">Figura 20 - Parte trazeira da base do manipulador.</p> 	<p data-bbox="919 1509 1390 1599">Figura 21 - Parte trazeira da base do manipulador.</p> 

Tabela 2 – Peças projetadas e fabricadas (Continua)

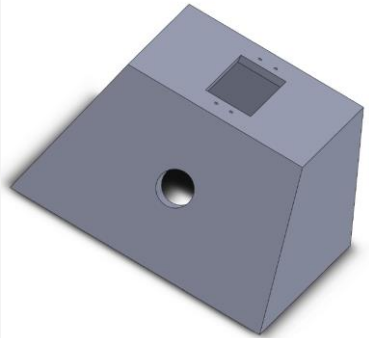

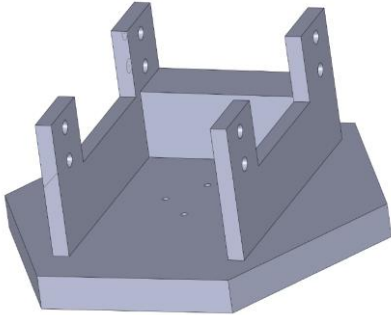



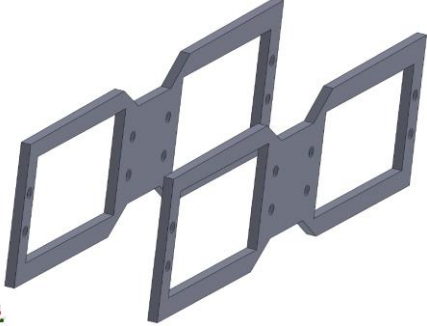

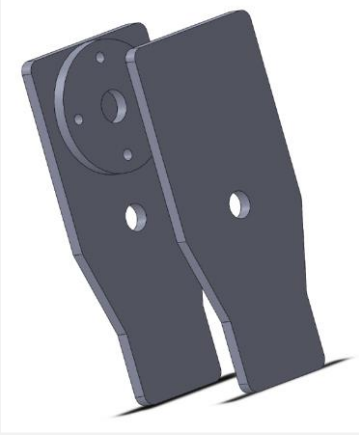

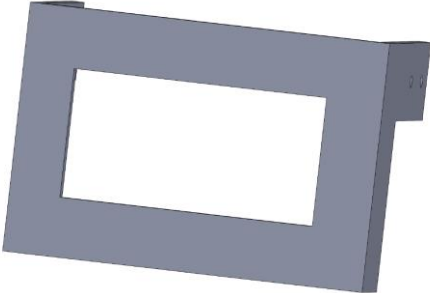



PEÇAS DESENHADAS VIA SOFTWARE	PEÇAS FABRICADAS
<p data-bbox="264 344 815 383">Figura 22 - Base completa do manipulador</p> 	<p data-bbox="882 344 1437 432">Figura 23 - Base completa do manipulador fabricada</p> 
<p data-bbox="312 763 767 801">Figura 24 - Ombro do manipulador</p> 	<p data-bbox="868 763 1453 801">Figura 25 - Ombro fabricado do manipulador</p> 
<p data-bbox="320 1182 759 1220">Figura 26 - Braço do manipulador</p> 	<p data-bbox="975 1182 1342 1220">Figura 27 – Braço fabricado</p> 
<p data-bbox="312 1615 767 1702">Figura 28 - Chapas do antebraço do manipulador.</p> 	<p data-bbox="951 1615 1374 1653">Figura 29 – Antebraço fabricado</p> 

Tabela 2 – Peças projetadas e fabricadas (Conclusão)

PEÇAS DESENHADAS VIA SOFTWARE	PEÇAS FABRICADAS
<p data-bbox="245 344 836 383">Figura 30 - Chapas do punho do manipulador</p> 	<p data-bbox="916 344 1394 434">Figura 31 – Primeira parte do punho projetado</p> 
<p data-bbox="268 875 815 965">Figura 32 - Segunda peça que compõem o punho</p> 	<p data-bbox="916 875 1394 965">Figura 33 – Segunda parte do punho fabricado</p> 
<p data-bbox="316 1429 756 1467">Figura 34 - Garra do manipulador</p> 	<p data-bbox="970 1429 1337 1467">Figura 35 – Garra projetada</p> 

Fonte: Do autor

A base é responsável por fixar o manipulador no seu espaço de trabalho. Para a sua construção foram utilizados chapas em formato quadrado e quadrilátero (trapézio) com as

dimensões 9,1 cm de comprimento, 8,24 cm de largura e 0,37 cm de espessura para a chapa quadrada. A chapa quadrada tem a abertura para a fixação do servo motor. Para as 4 chapas no formato de trapézio as dimensões são 6,27 cm de comprimento, 8,61 cm de largura menor, 10,98 cm de largura maior e 1,05 cm de espessura. Note que uma chapa tem um furo central onde será passado os fios do servo. Como pode ser observado nas Figuras 16,17,18,19,20,21, 22 e 23.

Para o ombro foi projetada uma peça no formato de hexágono com o raio de 10,9 cm, que está acoplada ao eixo do servo motor da base. Esta peça possui suportes para fixação de dois servomotores. Conforme mostrado nas Figuras 24 e 25.

Para a projeção do braço foram feitas duas chapas de 0,2 cm de espessura, 10,2 cm de comprimento e 2,13 cm de largura. Estas chapas são ligadas por um parafuso de 5,16 cm de comprimento e são acoplados dois servomotores, o do ombro e o do braço, com a utilização de dois componentes de formato circular com furo central para o eixo. Que são mostrados nas Figura 26 e 27.

Para o antebraço projetou-se duas chapas de 2 mm de espessura, 4,27 cm de comprimento e 2,86 cm de largura. Estas possuem duas aberturas para fixar os servos do braço e cotovelo. Conforme mostrado nas Figura 28 e 29.

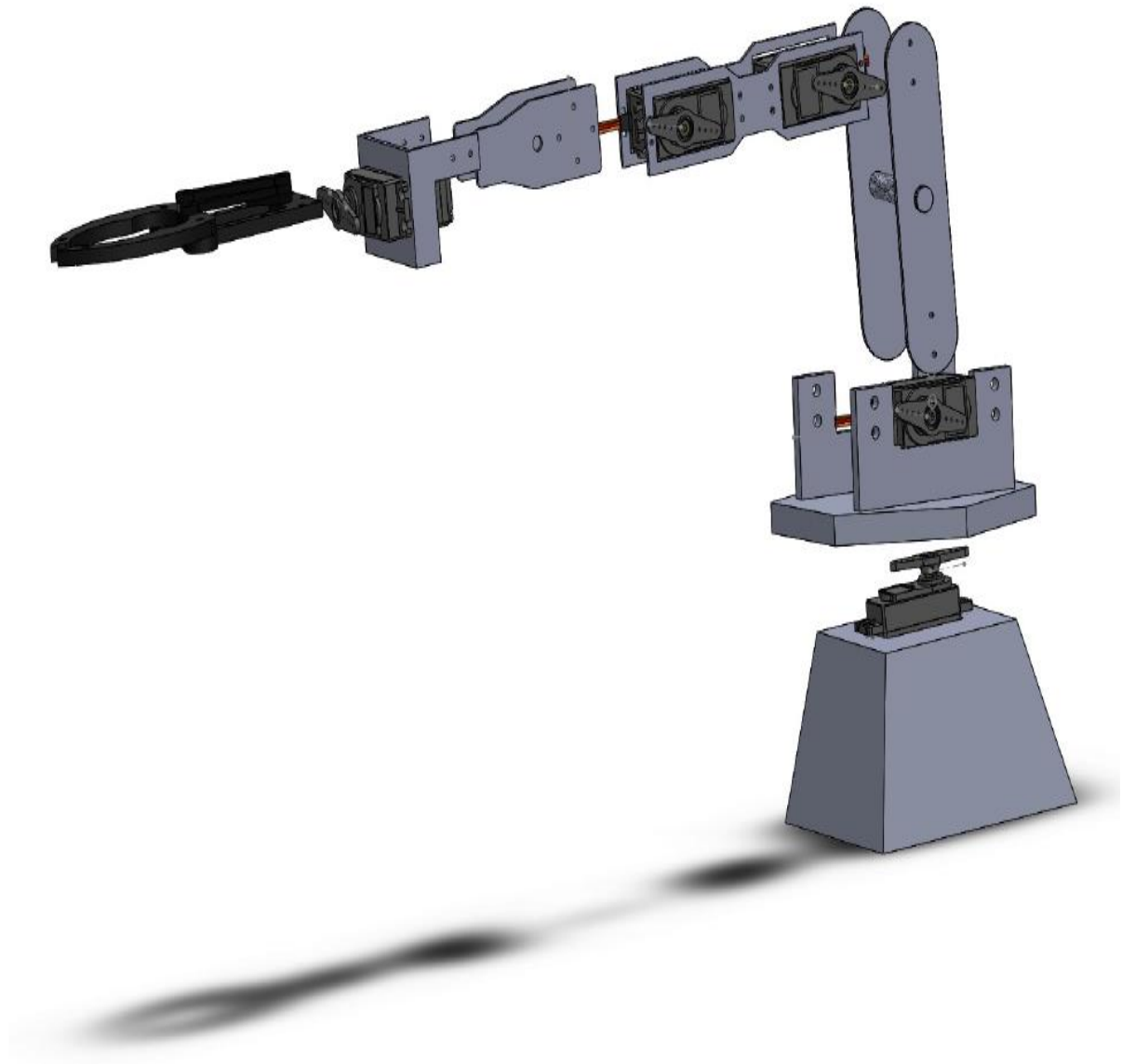
Para o punho, foram projetadas duas peças. A primeira possui duas chapas para acoplar no servo do antebraço e têm as dimensões: 3,73 cm de comprimento, 1,7 cm de largura menor, 2,9 cm de largura maior, 2 mm de espessura. Conforme mostrado na figura 30 e 31

Na segunda peça que compõem o punho foram projetadas duas chapas no formato de L e uma chapa que as interligam com as seguintes dimensões: 5,7 cm de comprimento, 38,86 cm de largura, 2 mm de espessura com abertura para a fixação do servo do punho. As chapas em L têm as dimensões: 3,8 de comprimento, 3,7 cm de largura e 2mm de espessura. Conforme mostrado na figura 32 e 33

Para a garra foram projetadas um conjunto de peças, onde o seu mecanismo de abertura e fechamento funciona conforme a posição do eixo do servo motor. Conforme mostrado na figura 34 e 35

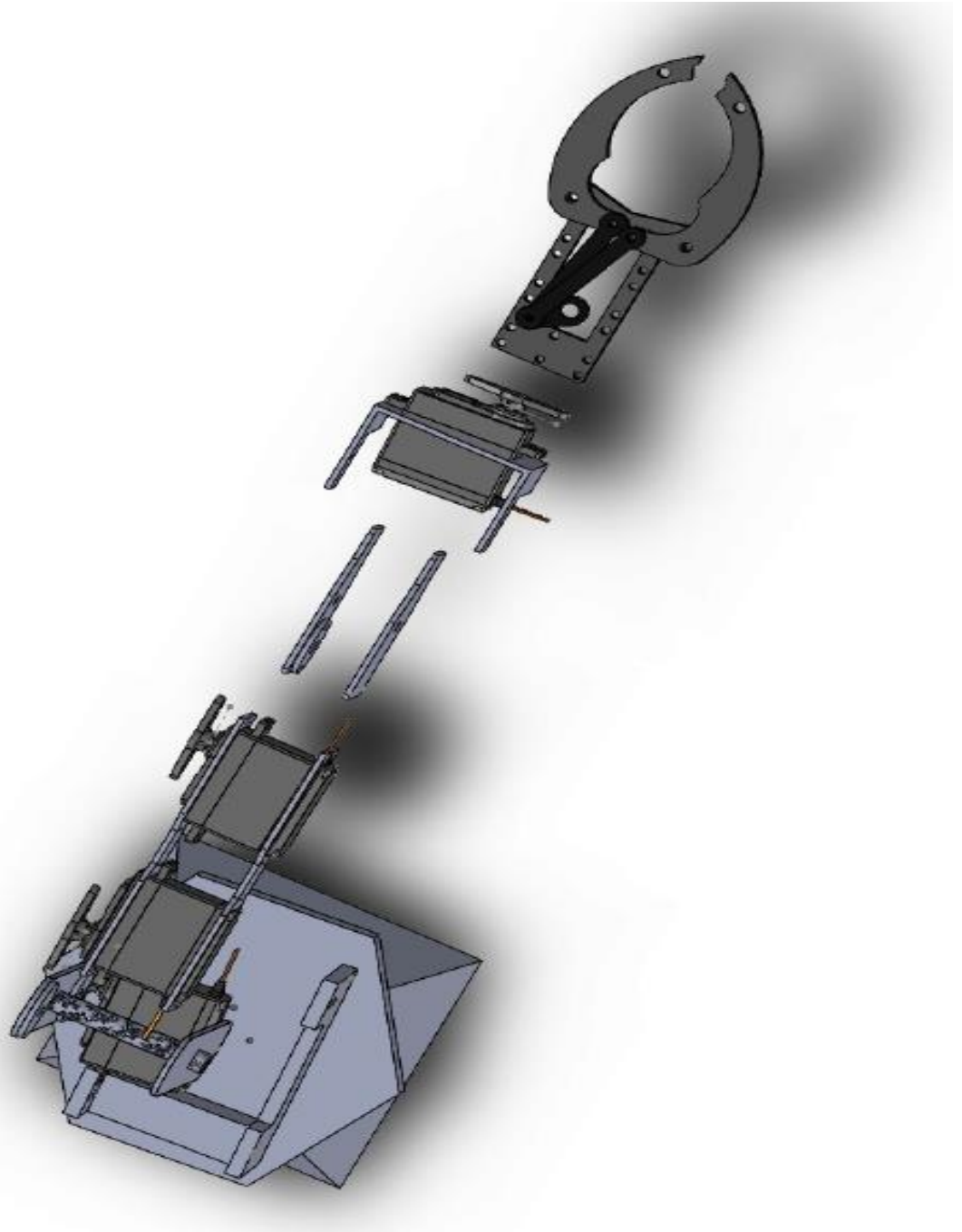
Nas Figuras 36, 37 e 38 pode ser visto a montagem do manipulador via *SOLIDWORK*. Note que as figuras apresentam a posição de cada peça e a fixação dos Servomotores nas suas posições, não são demonstrados os parafusos que serão utilizados, mas sim os seus furos.

Figura 36 - Vista lateral do manipulador montado por meio de software



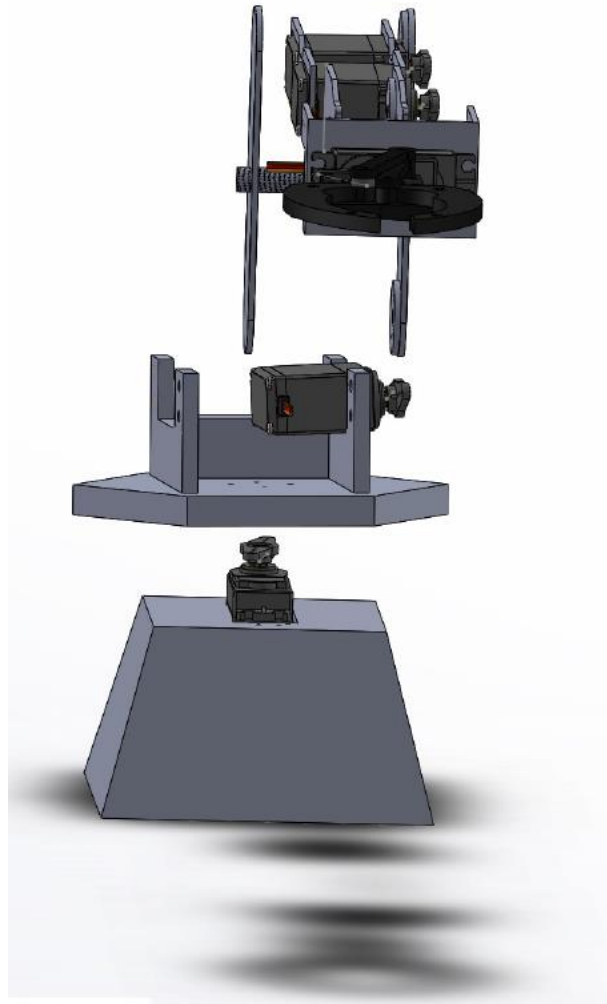
Fonte: Do autor

Figura 37 - Vista superior do manipulador montado por meio de software



Fonte: Do autor

Figura 38 - Vista frontal do manipulador montado por meio de software



Fonte: Do autor

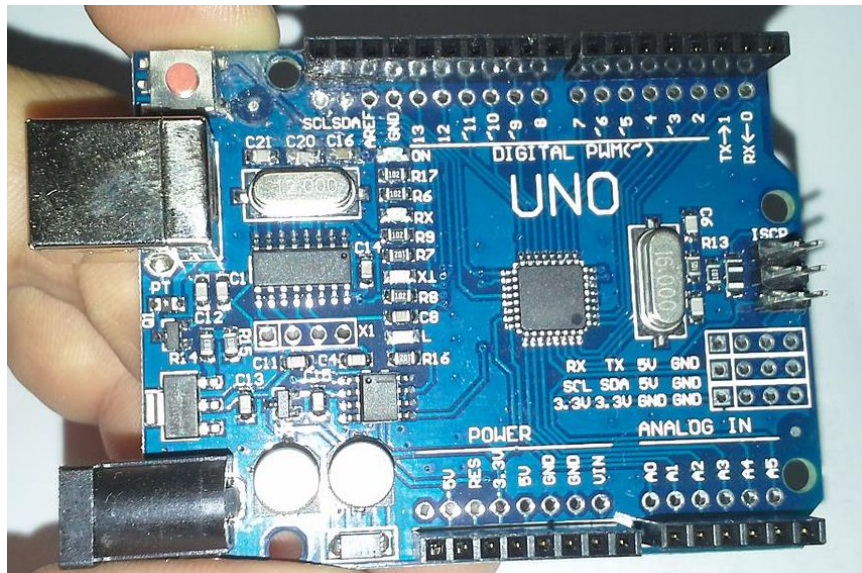
O sistema de controle consiste em um microcontrolador, um sensor, os servomotores que realizaram os movimentos do manipulador e o módulo *WI-FI*, que será destinado ao modo de operação manual por meio da rede *WI-FI*.

O microcontrolador a ser utilizado é o *ATMEGA328P*, que é utilizado na plataforma *ARDUINO* e será responsável por realizar o controle do sistema. *ARDUINO* é uma plataforma flexível *OPEN-SOURCE* de hardware e software para a prototipagem eletrônica (MELO,2012). O hardware é composto por uma placa com um circuito microcontrolado, onde são colocados todos os componentes necessários para realizar a comunicação com o computador. A placa de *ARDUINO* possui diversos conectores (digitais e analógicos) para interagir com o meio externo. O modelo utilizado neste trabalho foi o *ARDUINO UNO* que

possui 14 pinos de entrada e saída digital (pinos 0-13), 6 pinos de entrada analógica (pinos a0 até a5) e seis pinos de saída analógica (pinos 3,5,6,9,10 e 11) (MELO,2012).

A alimentação pode ser de dois métodos por uma porta USB ou por meio do adaptador AC que recebe 9 volts (MELO,2012). A Figura 39 mostra a placa de *ARDUINO*.

Figura 39 - Placa de um ARDUINO UNO



Fonte: Do autor

O software de programação é a *IDE*, que é um software livre, onde é criado os Sketches para a placa *ARDUINO*. A linguagem utilizada para a criação dos programas é *C/C++*, que é transmitido para o compilador *AVR-GCC*, que realiza a tradução dos comandos para uma linguagem que pode ser compreendida pelo microcontrolador (SOUZA,2013). O ciclo de programação do *ARDUINO* consiste em:

- 1- Conexão da placa a uma porta USB do computador
- 2- Desenvolvimento de um Sketch com comandos para a placa
- 3- Upload do Sketch para a placa, utilizando a comunicação USB
- 4- Aguardar a reinicialização, após ocorrerá à execução do Sketch criado

A grande vantagem da utilização de microcontroladores é que a partir do momento que o software é gravado na placa, não precisa mais da utilização do computador (desde de que ela seja alimentada), sendo seus comandos realizados de forma precisa (SOUZA,2013).

Os servomotores que serão utilizados para realizar a movimentação são os MG995 e MG996r, que são do tipo Hobby e são controlador por PWM. Estes servomotores possuem uma força de 14 kgf/cm para os MG995 e 15 Kgf/cm para os MG996r. A sua alimentação varia de 4.8 V até 7.2 V. São utilizados um total de 7 servomotores, 5 para o braço e 2 para o punho, dando ao manipulador 5 juntas rotacionais e a abertura da garra, portanto o manipulador possui 5 graus de liberdade. A Figura 40 mostra a parte do *DATASHEET* do servo motor, onde contém as informações do torque e da velocidade, e das tensões de alimentação.

Figura 40 - Parte do *DATASHEET* do servo motor.

TowerPro MG996R - Standard Servo

Basic Information

Modulation:	Analog
Torque:	4.8V: 130.5 oz-in (9.40 kg-cm) 6.0V: 152.8 oz-in (11.00 kg-cm)
Speed:	4.8V: 0.17 sec/60° 6.0V: 0.14 sec/60°
Weight:	1.94 oz (55.0 g)
Dimensions:	Length: 1.60 in (40.6 mm) Width: 0.78 in (19.8 mm) Height: 1.69 in (42.9 mm)
Motor Type:	? (add)
Gear Type:	Metal
Rotation/Support:	Dual Bearings



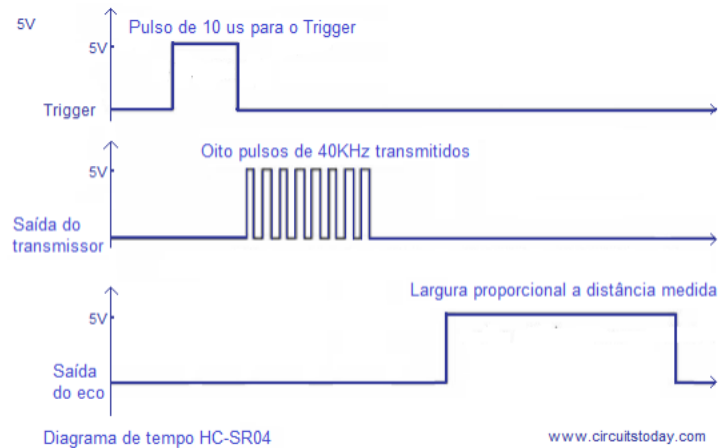
Brand:	Tower pro
Product Number:	? (add)
Suggested Retail:	? (add)
Street Price:	11.95 USD

Fonte: DATASHEET MG996r

O sensor ultrassônico HC-SR04 que é utilizado tem a finalidade de detectar o objeto que será manipulado pelo braço robótico e mensurar a distância que este objeto está em relação ao manipulador. Neste sensor há um circuito de controle, um transmissor e um receptor ultrassônico. Ele fornece medidas de 20 mm até 4000 mm de distância, com uma precisão de 3 mm (NAKATANI, GUIMARÃES, Neto, 2014).

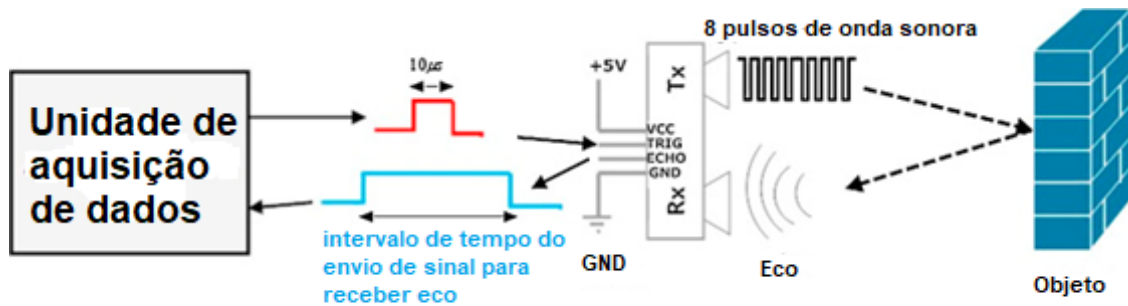
O seu funcionamento se dá por meio da transmissão de 8 ciclos de pulsos ultrassônicos a 40 kHz e esperar pelo sinal refletido. Ao recebê-lo o pino *echo* será colocando em *high* e sofrerá um *delay* proporcional à distância (NAKATANI, GUIMARÃES, Neto, 2014), conforme pode ser visto nas Figuras 41 e 42.

Figura 41 - Diagrama de funcionamento do HC-SR04



Fonte: NAKATANI, GUIMARÃES, Neto, 2014

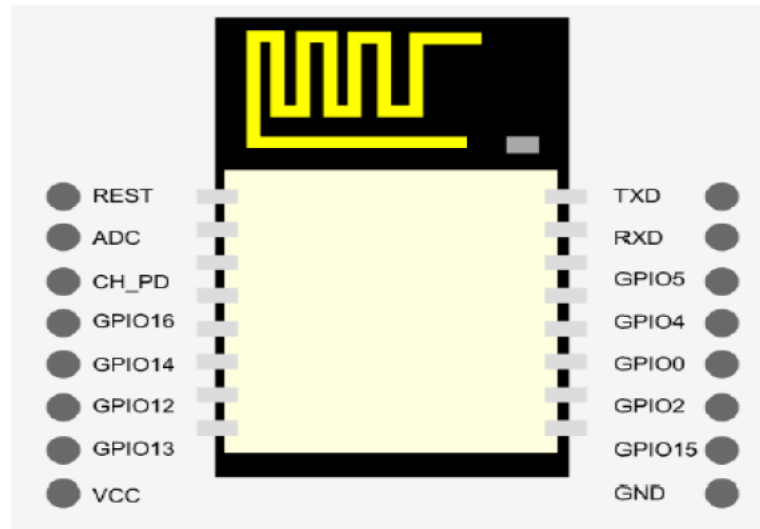
Figura 42 - Funcionamento do Sensor ultrassônico.



Fonte: PANDA, 2016

O módulo *WI-FI* consiste em um microcontrolador com um sistema de comunicação *WI-FI*. Apesar de possuir um processador próprio é bastante utilizado como módulo *WI-FI* para *ARDUINO*. Este suporta as redes 802.11 b/g/n e possui um adaptador (EP01) que possui um conversor de nível (OLIVEIRA,2017). Este módulo tem o intuito de fazer a operação no modo manual do manipulador, por meio de smartphone com rede sem fio. A figura 43 mostra o módulo com o seu soquete.

Figura 43 – Módulo ESP8266 com soquete para uso em placas PCB



Fonte: OLIVEIRA, 2017.

Na Figura 44 são mostradas as especificações gerais do módulo.

Figura 44 - Especificações gerais do ESP8266

Voltagem	3.3V
Consumo de Corrente	10 μ A
Memória Flash	16MB max (512k normal)
Processador	Tensilica L106 32 bit
Velocidade do processador	80-160MHz
RAM	32K + 80K
GPIOs	17(multiplexada com outras funções)
Analogico para digital	1 entrada com 1024 de resolução
Suporte 802.11	b/g/n/d/e/i/k/r
Maxima corrente de conexão TCP	5

Fonte: OLIVEIRA, 2017

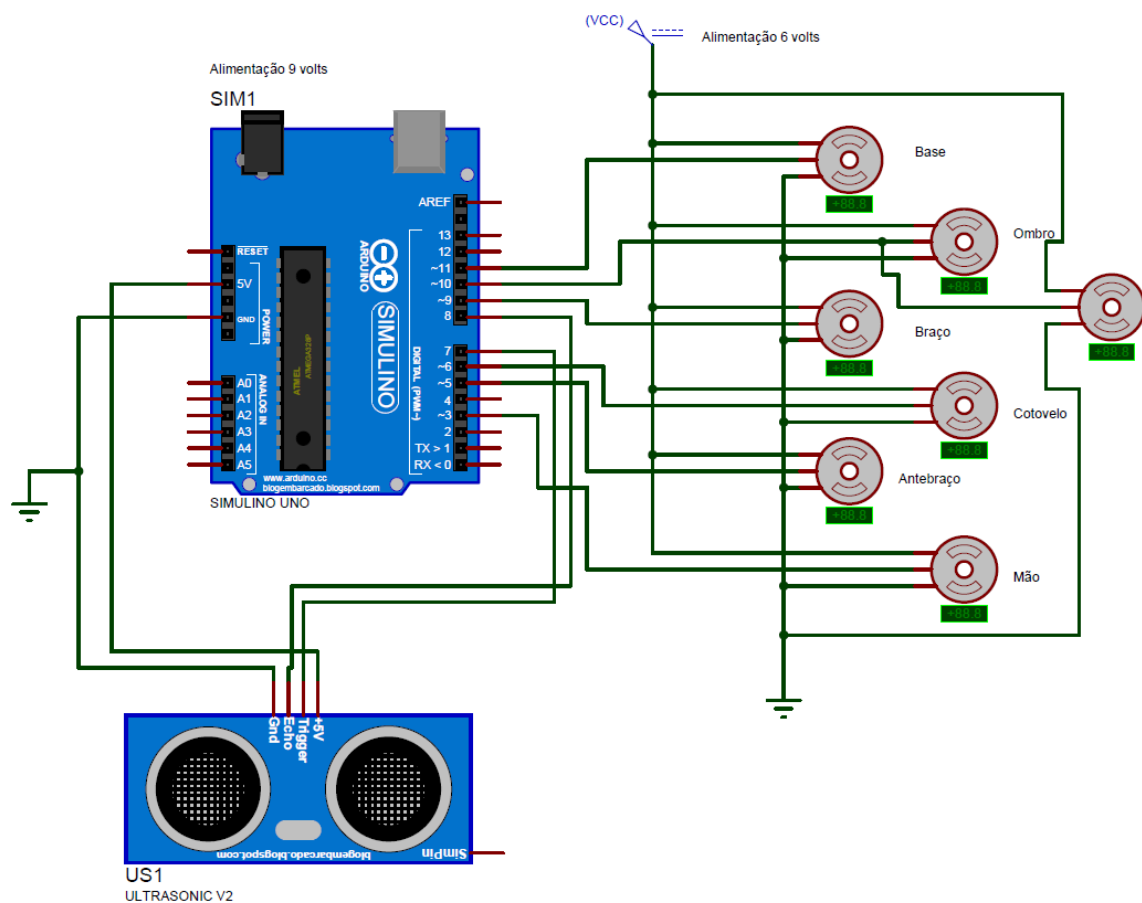
4.2 RESULTADO E DISCUSSÃO

O primeiro passo para realizar o controle do manipulador, foi montar o circuito por meio software *PROTEUS*, conforme mostrado na Figura 45 o circuito é constituído 7 servomotores

que são responsáveis por cada articulação do manipulador, entretanto a junta do ombro possui dois servomotores, eles são alimentados com uma fonte externa de 6 volts e seus pinos de controle são ligados nos pinos PWM do *ARDUINO*.

Já o sensor ultrassônico, como não consome uma corrente alta, é alimentado direto no *ARDUINO* por meio do pino de 5 volts e GND, os pinos *trigger* e *echo* são ligados aos pinos digitais 8 e 7. A alimentação da placa é por meio da porta USB ou Porta SIM1 que permite a conexão de uma fonte de 9 volts.

Figura 45 - Circuito montado no *PROTEUS*



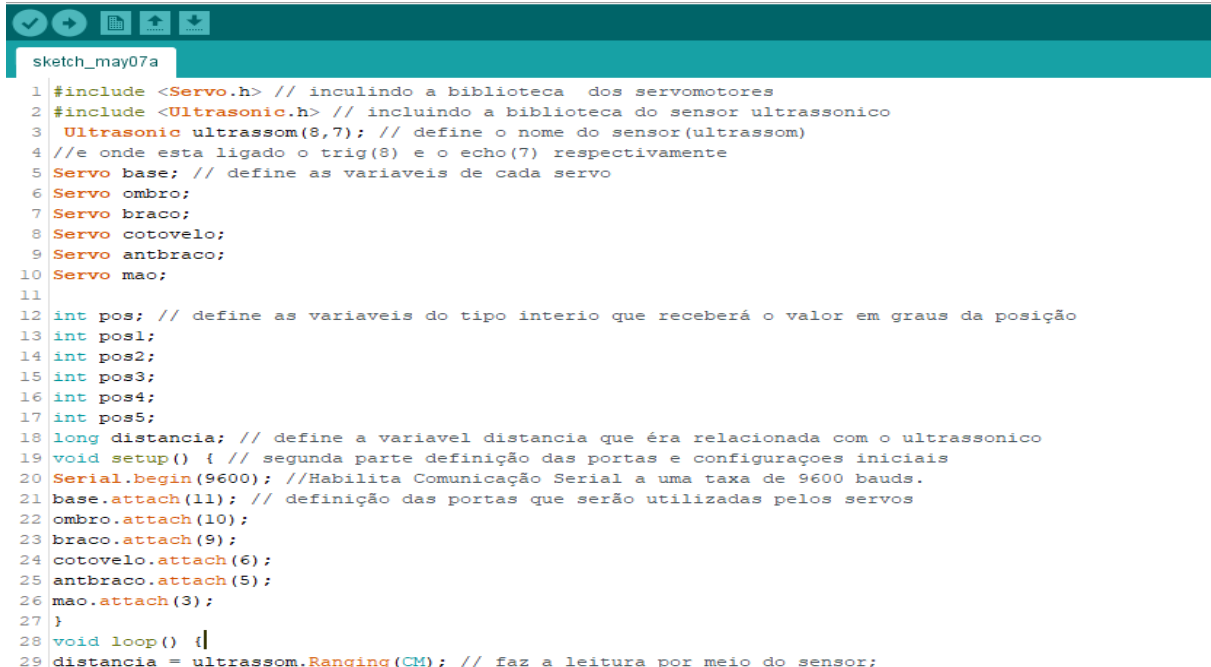
Fonte: Do autor

Após a montagem do circuito no software, deu início a programação de teste por meio da *IDE* do *ARDUINO*, que será utilizado para simular o circuito no *PROTEUS*. Na programação em C++ feita na *IDE*, o algoritmo tem um corpo pré-definido que é dividido em três partes. Na primeira parte são definidas as bibliotecas que serão utilizadas, as variáveis e seus tipos. Na

segunda parte são definidas as portas e as configurações iniciais. E na terceira parte é um laço infinito que contém os processos e operações que serão realizados.

A Figura 46 mostra o algoritmo que foi criado para a simulação, com seus respectivos comentários.

Figura 46 - Código para a simulação.



```

1 #include <Servo.h> // incluindo a biblioteca dos servomotores
2 #include <Ultrasonic.h> // incluindo a biblioteca do sensor ultrassonico
3 Ultrasonic ultrassom(8,7); // define o nome do sensor(ultrassom)
4 //e onde esta ligado o trig(8) e o echo(7) respectivamente
5 Servo base; // define as variaveis de cada servo
6 Servo ombro;
7 Servo braco;
8 Servo cotovelo;
9 Servo antbraco;
10 Servo mao;
11
12 int pos; // define as variaveis do tipo interio que receberá o valor em graus da posição
13 int pos1;
14 int pos2;
15 int pos3;
16 int pos4;
17 int pos5;
18 long distancia; // define a variavel distancia que é relacionada com o ultrassonico
19 void setup() { // segunda parte definição das portas e configurações iniciais
20 Serial.begin(9600); //Habilita Comunicação Serial a uma taxa de 9600 bauds.
21 base.attach(11); // definição das portas que serão utilizadas pelos servos
22 ombro.attach(10);
23 braco.attach(9);
24 cotovelo.attach(6);
25 antbraco.attach(5);
26 mao.attach(3);
27 }
28 void loop() {
29 distancia = ultrassom.Ranging(CM); // faz a leitura por meio do sensor;

```

Fonte: Do autor

O algoritmo acima foi aplicado no circuito montado no software para simular e verificar se as ligações estavam corretas, para posteriormente ser feito os testes práticos.

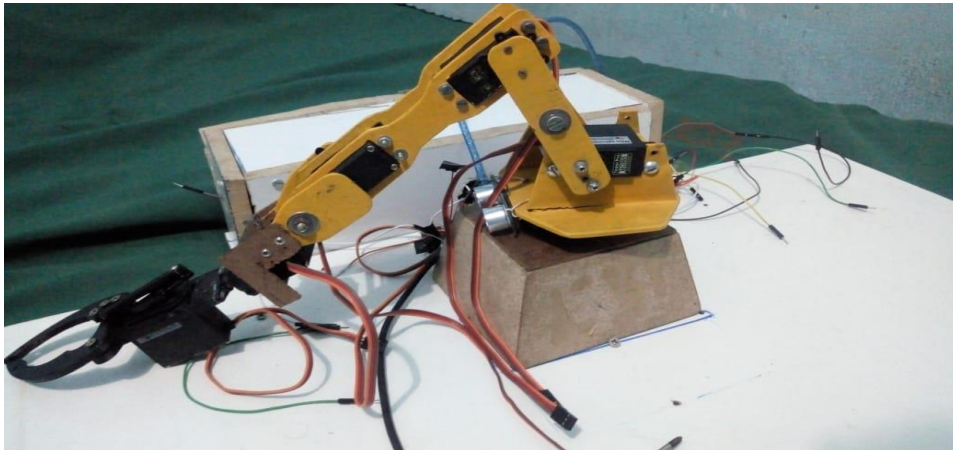
O protótipo do manipulador desenvolvido é do tipo antropomórfico (RRR) com as 5 juntas rotacionais, inclusive as juntas principais, com rotação de 0 a 180 graus. Já a seu órgão terminal é do tipo garra dois dedos. Este manipulador se assemelha muito com o manipulador de PUMA (TEXEIRA,1998), com diferença de um grau de liberdade a menos referente ao punho não esférico. As Figuras 47 e 48 mostram o protótipo do manipulador pronto.

Figura 47 - Protótipo montado com seus 5 Graus de Liberdade e a garra



Fonte: Do autor

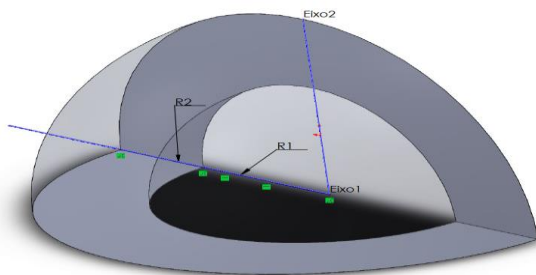
Figura 48 - Vista lateral do protótipo



Fonte: Do autor

Como este manipulador possui todas suas juntas rotacionais o seu espaço de trabalho é mostrado na Figura 49.

Figura 49 - Espaço de trabalho do manipulador desenvolvido



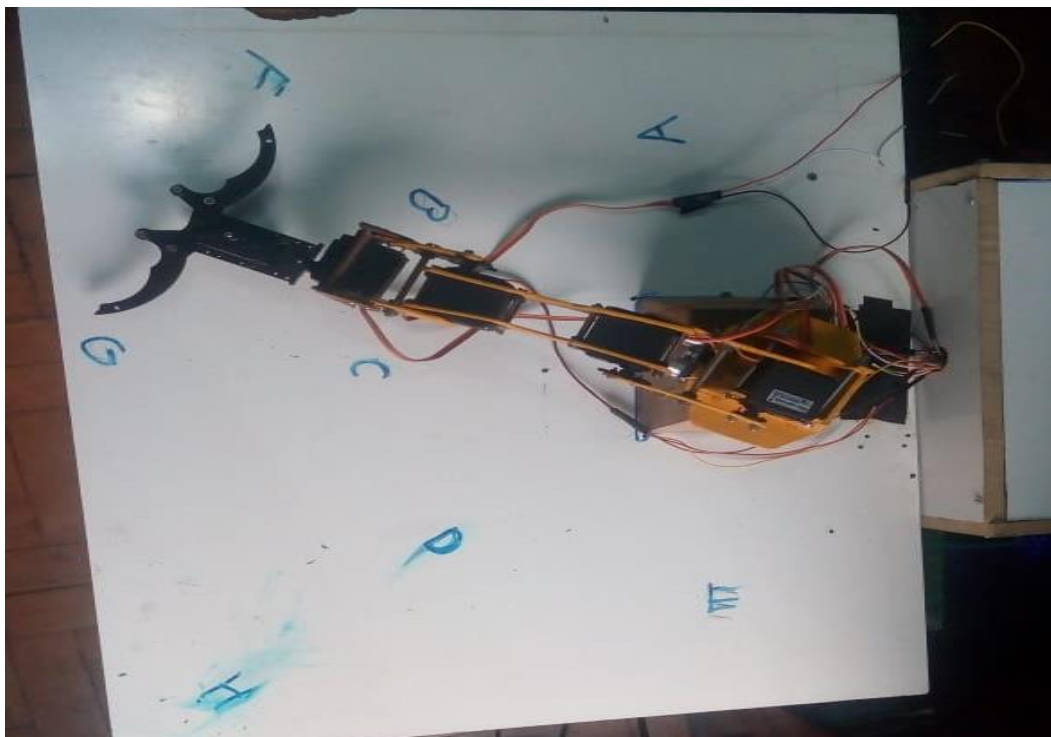
Fonte: Do autor

Note que R1 é o raio interno da esfera menor e representa a menor distância que o manipulador consegue alcançar, que neste caso é 10 cm. Já o R2 é o raio da esfera maior e representa o comprimento do braço todo estendido, que é 33 cm. Para calcular o espaço de trabalho (V) utilizou a Equação 1. Que está relacionada com o cálculo do volume das esferas, pois como já foi visto o espaço de trabalho é sempre dado em unidades volumétricas. Então para o manipulador desenvolvimento seu espaço de trabalho é 1.575 m³.

$$V = \frac{1}{4} * \pi * R2^3 - \frac{1}{4} * \pi * R1^3 \quad (1)$$

Foram marcados pontos referente ao espaço de trabalho, conforme a Figura 50. Note que os pontos A, B, C, D e E são referentes a esfera de raio menor da esfera, ou seja, pontos de menor distancia que o manipulador consegue alcançar. Já os pontos F, G e H refere-se aos pontos de maiores alcances.

Figura 50 – Marcações do espaço de trabalho



Fonte: Do autor.

As análises de forças foram feitas com relação ao ponto extremo do manipulador, ou seja, mover um objeto com seu comprimento todo estendido. Utilizando a lei de Newton, Equação 2, e a lei dos momentos, Equação 3, e sabendo que os pontos onde tem que exercer maiores força são os pontos da base e do ombro, pois estão mais distantes do objeto. Os servomotores suportam de 11 kg/cm a 15 kg/cm, ou seja, a 1 cm do seu eixo o servo suporta 11 kg a 15kg, e a massa do braço é de 0.25 kg, sem considerar a massa da base. Então para estimar a massa do objeto a ser movido utilizou a Equação 4.

$$\sum F = 0 \quad (2)$$

$$\sum Mo = 0 \quad (3)$$

$$m2 = m1 * \frac{d1}{d2} \quad (4)$$

Onde:

- m2 - massa 2 suportada no eixo
- m1 - massa suportada pelo servo
- d1 – distancia a 1 cm do eixo
- d2 – distancia que o objeto está do eixo

Para o caso extremo o d2 é de 33 cm, m1 é 15 kg e d1 é 1 cm, então a massa suportada no comprimento máximo do braço é de 0,4545 kg. Como a massa do manipulador interfere para a obtenção da massa do objeto tem-se que, subtraí-la do valor máximo e assim encontrar a massa do objeto, que neste caso e 0,2025 kg.

Nos tópicos seguintes serão apresentados os testes que foram realizados e a análise do seu funcionamento do manipulador desenvolvido a fim de verificar está apto para aplicações em projetos futuros.

4.2.1 Teste 1

Para este teste utilizou a alimentação do protótipo uma fonte de 5 volts e 3 amperes. Já o objeto utilizado foi uma lata de alumínio com um pouco de líquido, com o peso estimado de 0,150 kg.

O primeiro teste foi dividido em 3 partes. A primeira parte consistiu em detectar o objeto com o líquido e por meio de uma sequência de movimentos e pega-lo. A segunda parte foi realizada a movimentação até o ponto desejado e transferência do líquido para o copo, por meio de uma sequência de movimentos. E a terceira parte consiste em colocar o objeto na posição final de descarte. As rotações são em relação a posição do eixo dos servos.

Antes de iniciar a primeira parte foi feito a movimentação das juntas para posicioná-las nas posições iniciais afim de não atrapalhar na detecção do objeto. A Tabela 3 mostra estes movimentos.

Tabela 3 - Sequência de movimentos para as posições iniciais

Sequencia	Juntas	Movimentos
1	Ombro (junta 2)	Rotação para 70°
2	Braço (junta 3)	Rotação para 90°
3	Cotovelo (junta 4)	Rotação para 90°
4	Antebraço (junta 5)	Rotação para 180°
5	Mão(servo 6)	Rotação para 180°

Fonte: Do autor

A Figura 51 mostra parte do algoritmo em que são passados os valores para cada servo das juntas do manipulador para se posicionar na posição inicial, por meio do comando *servo.write()*. Note que o algoritmo criado é idêntico ao da simulação, por isso são utilizados os mesmos comandos para as configurações iniciais e as mesmas bibliotecas, assim é mostrado só parte do algoritmo contendo os comandos para transferir os valores de ângulos para cada servo. O próprio ARDUINO já faz a conversão do valor inserido para PWM.

Figura 51 - Parte do algoritmo para as posições iniciais das juntas do manipulador

```

tcc $ FollowLine Teste
27 }
28
29 void loop() {
30 // put your main code here, to run repeatedly:
31 pos1=70;
32 ombro.write(pos1);
33 delay(1000);
34 pos2=90;
35 braco.write(pos2);
36 delay(1000);
37 pos3=90;
38 cotovelo.write(pos3);
39 delay(1000);
40 pos4=180;
41 antbraco.write(pos4);
42 delay(1000);}
43 for (pos5 = 0; pos5 <= 180; pos5 += 1) { // goes from 0 degrees to 180 degrees
44 // in steps of 1 degree
45 mao.write(pos5); // tell servo to go to position in variable 'pos'
46 delay(15);
47 }

```

Fonte: Do autor

Na primeira parte, como o sensor ultrassônico está fixado na base do manipulador, moveu-se a base de 0° até 160° para varrer o espaço de trabalho em busca do objeto com o líquido. Se o objeto estiver a uma distância entre 15 cm e 20 cm, o manipulador irá realizar uma sequência de movimentos para pegá-lo. A Tabela 4 mostra a sequência de movimentos pré-definidos, note que quando o sensor detecta o objeto é feito um ajuste na posição da base de 5°, pois a estrutura não está localizada no centro da base, é utilizado um intervalo de um minuto entre cada movimento para que o movimento de uma junta não interfira no movimento da outra junta.

Tabela 4 - Sequência de movimentos para pegar o objeto

Sequencia	juntas	Movimentos
1	Base (junta1)	Posição atual +5°
2	Ombro(junta 2)	Rotação em 35°
3	Cotovelo (junta 4)	Rotação para 160°
4	Braço (junta 3)	Rotação para 5°
5	Cotovelo (junta 4)	Rotação para 100°
6	Antebraço(junta 5)	Rotação para 180°
7	Mão (servo 6)	Rotação para 60°

Nas Figuras 52 e 53 é mostrado parte do algoritmo utilizado neste teste. Neste algoritmo inicialmente foi utilizado o laço *FOR* para a movimentação da base e do sensor até detectar a posição do objeto. Para fazer a leitura do sensor é utilizado o comando *Ultrassom.Ranging(CM)*, onde já retorna o valor da distância em centímetros e é utilizando o comando condicional *IF* para realizar os movimentos para manipular o objeto caso a condição seja satisfeita.

Figura 52 - Parte do algoritmo utilizados no teste 1

```
tcc $ FollowLine Teste
48 for (pos = 0; pos <= 160; pos++) {
49 base.write(pos);
50 distancia = ultrassom.Ranging(CM);
51 Serial.print(distancia); //imprime o valor da variável distancia
52 Serial.println("cm");
53 delay(500);
54 if( distancia >= 15 && distancia <= 20){
55 pos=pos+5;
56 base.write(pos);
57 delay(1000);
58 pos1=35;
59 ombro.write(pos1);
60 delay(1000);
61 pos3=160;
62 cotovelo.write(pos3);
63 delay(1000);
64 pos2=5;
65 braco.write(pos2);
66 delay(1000);
67 pos3=100;
```

Fonte: Do autor

Figura 53 – Continuação do algoritmo para o teste 1

```
68 cotovelo.write(pos3);
69 delay(1000);
70 pos4=180;
71 antbraco.write(pos4);
72 delay(1000);
73 for (pos5 = 180; pos5 >= 60; pos5 -= 1) {
74     mao.write(pos5); // tell servo to go to position in variable 'pos'
75     delay(15);
76 }
77 delay(1500);
78 pos2=20;
79 braco.write(pos2);
80 delay(1000);
```

Fonte: Do autor

Se o sensor não detectar nenhum objeto na rotação da base 0° até 170°, esta será girada de 170° até 0° em busca do objeto, assim enquanto não for encontrado um objeto na distância estipulada, o manipulador ficará realizando o movimento da base até encontrá-lo.

Na segunda parte, o ponto desejado onde estava localizado o copo era no ponto E (180°) e os movimentos para mover o objeto até este ponto são mostrados na Tabela 5. Entre a sequência 8 e 9 é utilizado um intervalo de 5 minutos, para garantir que o líquido seja transferido todo para o copo.

Tabela 5 – Sequência de movimentos realizados para mover o objeto até o ponto desejado e transferir o líquido do objeto para o copo

Sequencia	Juntas	Movimentos
1	Ombro (junta2)	Manteve em 60°
2	Cotovelo (junta 4)	Rotação para 100°
3	Braço (junta 3)	Rotação para 90°
4	Base (junta1)	Rotação para 170°
5	Braço (junta 3)	Rotação para 40°
6	Ombro (junta2)	Rotação para 30°
7	Braço (junta 3)	Rotação para 20°
8	Antebraço (junta 4)	Rotação para 50°
9	Antebraço (junta4)	Rotação para 180°
10	Braço (junta 3)	Rotação para 30°
11	Ombro (junta 2)	Rotação para 60°
12	Braço (junta 3)	Rotação para 90°

Fonte: do autor

As Figuras 54 e 55 mostram parte do algoritmo contendo os comandos utilizados para passar os valores das variáveis para os servos. Semelhante a primeira parte do teste.

Figura 54 – Parte do algoritmo contendo a sequência de movimentos realizados.

```

81 ////////////////////////////////////////////////////////////////////Parte 2
82 pos1=60;
83 ombro.write(pos1);
84 delay(1000);
85 pos3=100;
86 cotovelo.write(pos3);
87 delay(1000);
88 pos2=90;
89 braco.write(pos2);
90 delay(1000);
91 pos=170;
92 base.write(pos);
93 delay(1000);
94 pos2=40;
95 braco.write(pos2);
96 delay(1000);
97 pos1=30;
98 ombro.write(pos1);
99 delay(1000);

```

Fonte: Do autor

Figura 55 – Continuação do algoritmo contendo a sequência de movimentos realizados

```

100 pos2=20;
101 braco.write(pos2);
102 delay(1000);
103 for (pos4 = 180; pos4 >= 50; pos4 -= 1) {
104     antbraco.write(pos4);
105     delay(15);
106 }
107 delay(5000);
108 for (pos4 = 50; pos4 <= 180; pos4 += 1) {
109     antbraco.write(pos4);
110     delay(15);
111 }
112 delay(1000);
113 pos2=30;
114 braco.write(pos2);
115 delay(1000);
116 pos1=60;
117 ombro.write(pos1);
118 delay(1000);
119 pos2=90;
120 braco.write(pos2);
121 delay(1000);

```

Fonte: Do autor

Na parte 3, o objeto vazio foi movido até o ponto de descarte (ponto H). Para isso foi definido os seguintes movimentos mostrado na Tabela 6 e na Figura 56.

Tabela 6 – Sequência de movimentos para descartar o objeto vazio

Sequencia	Juntas	Movimentações
1	Base (junta 1)	Rotação para 120°
2	Cotovelo (junta 4)	Rotação para 160°

3	Ombro (junta 2)	Rotação para 120°
4	Cotovelo (junta 4)	Rotação para 110°
5	Mão (servo 6)	Rotação para 180°
6	Braço (junta 3)	Rotação para 90°
7	Ombro (junta 2)	Rotação para 60°
8	Base (junta 1)	Rotação para 0°

Fonte: Do autor

Figura 56 – Parte do algoritmo utilizado na terceira etapa do teste 1

```

119 delay(1000);
120 ///////////////////////////////////////////////////Parte 3
121 pos=120;
122 base.write(pos);
123 delay(1000);
124 pos3=160;
125 cotovelo.write(pos3);
126 delay(1000);
127 pos1=120;
128 ombro.write(pos1);
129 delay(1000);
130 pos3=110;
131 cotovelo.write(pos3);
132 delay(1000);
133 pos5=180;
134 mao.write(pos5);
135 delay(1000);
136 pos2=90;
137 braco.write(pos2);
138 delay(1000);
139 pos1=60;
140 ombro.write(pos1);
141 delay(1000);
142 ]

```

Fonte: Do autor

Após realizado as três partes do processo para este teste as juntas voltaram as suas posições iniciais e um novo ciclo será começado.

4.3.2 Teste 2

Para este teste foi utilizado uma lata com líquido no formato cilíndrico com um peso estimado de 200 g. para alimentar o manipulador foi utilizado uma fonte de 7 volts, pois como o objeto estava na extremidade do espaço de trabalho com 5 volts os servos não teriam forças para mover o objeto.

Neste teste, análogo ao teste 1, foi feita a sua divisão em três partes. A primeira parte consistiu em detectar o objeto nos pontos (F, G, H) e pegá-lo. A segunda parte consistiu em mover o objeto até o ponto onde estava localizado o copo (ponto E) e transferir o líquido do objeto para o copo. Já na parte 3 do teste consistiu em descartá-lo no ponto A.

Antes de iniciar o teste foi definido as posições iniciais de cada servo da junta para que, com a movimentação da base não encostasse no objeto. Para isso foi definido os movimentos para cada junta, semelhante ao do teste 1.

Após a definição das posições iniciais, iniciou-se a primeira etapa do processo. Análogo ao teste 1, como o sensor ultrassônico está fixado na sua base, estabeleceu-se que a base girasse de 0° a 160° para varrer o espaço de trabalho em busca de detectar a presença do objeto. Neste teste foi definido que, se o objeto estivesse a uma distância entre 30 cm a 40 cm, iniciaria as movimentações para pegá-lo. Após detectado o objeto os movimentos para pegá-lo são mostrados na Tabela 7.

Tabela 7 – Sequência de movimentos para pegar o objeto.

Sequência	Juntas	Movimentos
1	Base (junta 1)	Posição atual $+5^\circ$
2	Cotovelo (junta 4)	Rotação para 160°
3	Ombro (junta 2)	Rotação para 130°
4	Cotovelo (junta 4)	Rotação para 130°
5	Mão (servo 6)	Rotação para 60°

Fonte: Do autor

Do mesmo modo que no teste 1, a Figura 57 mostras parte do algoritmo utilizado. Os comandos são os mesmos a única diferença é com relação os valores dos ângulos para cada servo.

Figura 57 – Parte do algoritmo utilizado para o teste 2

```

48 for(pos = 0;pos <= 160;pos++){
49   base.write(pos);
50   distancia = ultrassom.Ranging(CM);
51   Serial.print(distancia); //imprime o valor da variável distancia
52   Serial.println("cm");
53   delay(500);
54   if( distancia >= 30 && distancia <= 40){
55     pos3=160;
56     cotovelo.write(pos3);
57     delay(1000);
58     pos1=130;
59     ombro.write(pos1);
60     delay(1000);
61     pos3=130;
62     cotovelo.write(pos3);
63     delay(1000);
64     for (pos5 = 180; pos5 >= 60; pos5 -= 1) {
65       mao.write(pos5);           // tell servo to go to position in variable 'pos'
66       delay(15);
67     }

```

Fonte: Do autor

Note que os movimentos são em relação ao eixo do servo, assim quando diz que a mão (junta 6) realizou a rotação para 60°, significa que por meio do seu mecanismo será realizado o fechamento para o diâmetro do objeto. É utilizado o mesmo intervalo entre as sequencias de movimentos que no teste 1.

Caso não seja encontrado um objeto a base irá girar no sentido contrário de 170° até 0° e ficará realizando esta movimentação de 0° até 170° e de 170° até 0° até que seja detectado um objeto na distância definida.

Na segunda parte deste teste, para mover o objeto até o ponto onde estava o copo foram realizados os seguintes movimentos, conforme mostrado na Tabela 8. Semelhante ao teste 1, o intervalo entre a sequência de movimento 7 e 8 é de 5 minutos, para garantir a transferência total do líquido para o copo. As Figuras 58 e 59, mostram parte do algoritmo utilizado nesta segunda parte do teste 2.

Tabela 8 - Sequência de movimentos realizados para mover o objeto até o ponto desejado e transferir o líquido do objeto para o copo

Sequência	Juntas	Movimentos
1	Ombro (junta2)	Rotação para 75°
2	Cotovelo (junta 4)	Rotação para 100°

Figura 59 – Continuação do algoritmo utilizado nesta segunda etapa do teste 2

```

92 delay(1000);
93 pos1=30;
94 ombro.write(pos1);
95 delay(1000);
96 pos2=10;
97 braco.write(pos2);
98 delay(1000);
99 for (pos4 = 180; pos4 >= 50; pos4 -= 1) {
100     antbraco.write(pos4);
101     delay(15);
102 }
103 delay(5000);
104 for (pos4 = 50; pos4 <= 180; pos4 += 1) {
105     antbraco.write(pos4);
106     delay(15);
107 }
108 delay(1000);
109 pos2=80;
110 braco.write(pos2);
111 delay(1000);
112 pos1=90;
113 ombro.write(pos1);
114 delay(1000);
115 pos2=90;

```

Fonte: Do autor

Na etapa 3 deste teste a sequência de movimentos realizados para descartar o objeto vazio no ponto A é mostrado na Tabela 9 e Figura 60. Como o ponto de descarte agora é em um ponto de raio menor do espaço de trabalho os movimentos foram diferentes do que no teste 1.

Tabela 9 – Sequência de movimentos para descarte

Sequência	juntas	Movimentos
1	Base (junta1)	Rotação para 10°
2	Ombro (junta 2)	Rotação para 30°
3	Braço (junta 3)	Rotação para 10°
4	Mão (junta6)	Rotação para 180°
5	Ombro (junta 2)	Rotação para 80°
6	Braço (junta 3)	Rotação para 90°
7	Base (junta1)	Rotação para 90°

Fonte: Do autor

Figura 60 – Parte do algoritmo utilizado na terceira etapa do teste 2

```

117 delay(1000);
118 //////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////part 3
119 pos=10;
120 base.write(pos);
121 delay(1000);
122 pos1=30;
123 ombro.write(pos1);
124 delay(1000);
125 pos2=10;
126 braco.write(pos2);
127 delay(1000);
128 pos5=180;
129 mao.write(pos5);
130 delay(1000);
131 pos2=90;
132 braco.write(pos2);
133 delay(1000);
134 pos1=90;
135 ombro.write(pos1);
136 delay(1000);
137 }
138 }

```

Fonte: Do autor

Após realizados as 3 partes do teste todas as juntas voltam as suas posições iniciais e um novo ciclo se inicia.

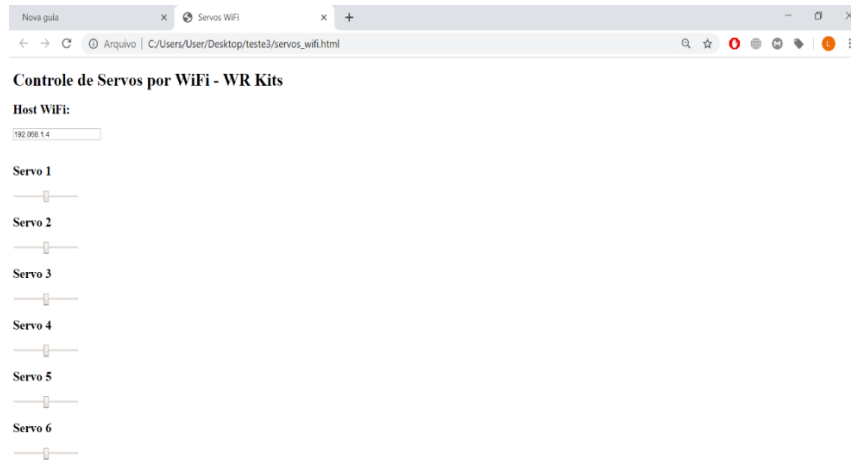
4.2.2 Teste 3

Neste teste foi utilizado o modo de operação manual do manipulador onde, por meio de um dispositivo ligado à rede WI-FI, o operador pode operá-lo de uma página WEB. Para isso foi utilizado o módulo ESP8266 conectado ao ARDUINO e criou-se uma página HTML onde foi possível operar cada servo do manipulador.

Para operar cada servo foram utilizadas barras de rolagem, pois isso facilita o modo de operação ao invés de ficar digitando valores de ângulos. Essas barras variam de 0° a 180° para cada servo.

Foi proposto este teste, pois em uma indústria ou qualquer outra área onde se empregam manipuladores, as vezes torna-se inviável mapear todos os pontos alcançáveis pelo manipulador, para realizar a operação automática. Então possibilitar o operador posicionar o manipulador em qualquer ponto do seu espaço de trabalho torna-o mais eficiente e prático. Na Figura 60 é mostrado a página Web que foi criada para este teste.

Figura 61 – Pagina Web criada



Fonte: Do autor

Note que tem uma caixa onde é passado o endereço de IP do módulo, assim é possível acessá-lo de qualquer dispositivo ligado à rede.

Para o algoritmo desenvolvido para este teste são utilizadas duas bibliotecas, a dos servos e a do módulo ESP8266, onde é feita a configuração para se conectar à rede WI-FI e fazer a comunicação dos comandos da página com o ARDUINO.

Para a criação da página foi utilizado a linguagem HTML, apesar de ser bem simples o seu funcionamento é bastante eficaz. Como já foi citado, foram utilizados os comandos para criar as barras de rolagem e a caixa onde é inserido o IP para se comunicar com o módulo.

Os comandos utilizados para a criação da caixa onde é colocando o IP são:

```
<div style="margin: 0; width: 600px; height: 90px
  <FORM NAME = 'myform2' ACTION="" METHOD="GET">
    <h2> host WIFI: </h2>
    <INPUT TYPE ="text" NAME = "Inputbox" VALUE=192.168.0.4>
  </FORM>
</div>
```

Note que é definido o tamanho da caixa por meio do *WIDTH* e *HEIGHT*. O comando *h2* define o nome da caixa.

Já para a barra de rolagem são utilizados os comandos:

```
<input type="range" min="15" max="175" onmouseup="mouseUpSr1(this.value)" />
```

Onde são definidos os valores máximos e mínimos referente a angulação do servo e é definido o botão utilizado para arrastar a barra de rolagem.

O algoritmo utilizado para este teste pode ser visto no Apêndice B e o algoritmo para a criação da página é mostrado no Apêndice C.

4.2.3 Análise dos resultados obtidos

Nos testes 1 e 2 o objetivo principal consistia em verificar se o espaço de trabalho calculado era consistente com os movimentos que o manipulador realizaria. E foi possível notar que o manipulador conseguiu se posicionar dentro do espaço calculado de forma precisa com um pequeno erro de ± 1 cm. Como o manipulador é do tipo antropomórfico ou articulado suas juntas todas são do tipo rotacional e o erro encontrado para cada junta foi de $\pm 2^\circ$.

Por meio dos testes foi possível perceber que a estrutura fabricada para o protótipo foi bastante satisfatória e apesar de sofrer pequenas deformações devido ao peso dos Servomotores, não interferiu no objetivo de manipular objetos. Foi utilizada uma pequena massa no ombro do manipulador para dar maior equilíbrio no eixo do servo da base e diminuir a sua deformação.

Na parte eletrônica percebe-se que com uma alimentação de 5 volts o manipulador apresentou dificuldades para realizar os movimentos, o que já era de se esperar por meio dos cálculos feitos. Já com uma alimentação de 6,5 volts a 7 volts o manipulador realizou as tarefas de forma coerente.

Como a junta que sofreu mais esforço para mover os objetos dos pontos mais distantes, foi a do ombro foi inserido mais um servo motor para auxiliar no manuseio dos objetos, podendo assim utilizar objetos com uma massa maior, que é mostrado no Teste 2.

No teste 3, que consistia em operar o manipulador por meio de uma rede WI-FI, pode-se perceber que o módulo utilizado junto com o servo conseguiu satisfazer o objetivo. Notou-se que não foi preciso realizar nenhum tipo de ajuste nas angulações, pois as posições enviadas para cada servo foram alcançadas por cada junta.

No teste 1 foram realizadas 5 repetições com o objetivo de analisar se o manipulador iria alcançar todos os pontos desejados. E estas repetições foram satisfatórias. Já para o teste 2

foram realizadas 4 repetições que foram satisfatórias. E no teste 3 foram realizadas 5 repetições onde operador controlava a posição do servo e o funcionamento foi conforme o desejado.

4.4 Limitações e Trabalhos Futuros

As dificuldades que foram encontradas no decorrer do desenvolvimento deste trabalho são apresentadas nesta seção. Estas dificuldades estão relacionadas com o tipo de material utilizado, a falta de componentes eletrônicos entre outros que serão vistos a seguir:

- **Materiais eletrônico**

A dificuldade relacionada aos materiais eletrônicos foi de encontra-los com um preço razoável, considerando que a cidade em que foi realizado o projeto (no caso, cidade de Lavras), trata-se de uma cidade de pequeno porte, sendo, portanto, difícil de encontrar componentes e quando encontram são com preços muito altos. Por isso os componentes foram comprados na Internet e o prazo para entrega destes, atrasou no desenvolvimento do protótipo.

Os servos do tipo MG996r comprados tinham uma qualidade inferior, pois agarravam na movimentação dos eixos, por isso um servo queimou e para os demais, ajustes tiveram que ser feitos

Houve bastante dificuldade na utilização do módulo, pois, o valor de SETBOUND estava diferente dos modelos encontrados na internet, então foi feito um ajuste no valor da comunicação serial do ARDUINO para os valores que o modelo do módulo adquirido possuía.

- **Montagem do protótipo**

As dificuldades encontradas com a montagem do protótipo foram de encontrar os materiais, pois como não houve nenhum auxílio da universidade com recursos financeiros. Foi buscada a utilização de madeiras já utilizados para outros fins e com isso houve chapas que racharam ou quebraram e tiveram que ser substituídas.

Como os pesos servos eram altos e os materiais inferiores, não foi possível evitar pequenas deformações, que tornavam o manipulador um pouco torto.

Na garra do manipulador utilizou-se a impressora 3D do NLMT para a sua fabricação e o modelo fabricado saiu um pouco diferente do modelo desenhado, por causa do tipo de material que foi utilizado na sua fabricação.

- **Desenvolvimento do sistema eletrônico**

A dificuldade encontrada em relação a parte eletrônica foi por causa da quantidade de fios utilizados, que atrapalhou na organização do circuito montado na Protoboard e interferiu na leitura do sensor, tendo que ser feito uma reorganização dos fios dos servomotores

- **Programação e algoritmo**

Na programação a dificuldade foi relacionada com a criação da página HTML, pois o autor não possui nenhuma experiência com essa linguagem.

- **Melhorias**

Para melhorar o protótipo seja visualmente e no seu funcionamento, pode-se substituir toda a estrutura feita em madeira por alumínio que apresenta maior resistência e é mais leve, evitando deformações e rachaduras na estrutura.

Outra forma de melhorar o protótipo é utilizando motor de passos ao invés de Servomotores, para que se possa ganhar precisão, um controle eficiente de velocidade e aceleração e um maior espaço de trabalho, pois os motores de passos não possuem restrições na sua angulação.

Incluir sistema de visão computacional por câmera, para servir de apoio operacional ou até como suporte de sistema autônomo.

4.5 Custos

Foi montado a Tabela 11 com todos os materiais utilizados para a fabricação do protótipo. Alguns materiais foram doados e outros emprestado afim de se obter uma maior economia nesse projeto.

Tabela 10 - Custos para o desenvolvimento do manipulador

Itens	Descrição	Quantidade (unidades)	Preço unitário (RS)	Preço total (RS)
1	Servo motor MG996r	4	30,00	120,00
2	Servo motor MG995	4	30,00	120,00
3	ARDUINO	1	Emprestado	0,00
4	Garra	1	5,00	5,00
5	Chapas de madeira compensada 1 (220cmx160cm)		Doação	0,00
6	Chapas de madeiras 1 (60cmx30cm)		Doação	0,00
7	Chapas de MDF	1	Doação	0,00
8	Protoboard	1	20,00	20,00
9	Fios Jumpers macho - macho	40	15,00	15,00
10	Fios jumpers Macho-fêmea	40	15,00	15,00
11	Sensor ultrassónico	1	Emprestado	0,00
12	Modulo WI-FI	1	25,00	25,00
TOTAL				320,00

Fonte: do Autor

5 CONCLUSÃO

Com o desenvolvimento deste trabalho pode-se obter um grande aprendizado, pois proporcionou ao autor aplicar os conhecimentos teóricos adquiridos no decorrer do curso de

bacharel em Engenharia de Controle e Automação, utilizando os conceitos de mecânica, eletrônica, robótica, controle e computação.

Com este trabalho foi possível adquirir maior experiência em ferramentas que não são aprofundadas ou estudadas no curso. Que são: desenho de peças e modelos em 3d, criação de uma página HTML e simulação dos circuitos utilizando PROTEUS.

O protótipo conseguiu atender objetivos principais do trabalho, como foi mostrado nos testes práticos. O manipulador conseguiu mover os objetos de forma automática e de forma manual com grande eficiência.

Foi possível perceber que em um modelo real existem fatores que influenciam no seu funcionamento e isso não pode ser percebido no modelo simulado, possibilitando ao autor a resolução de problemas reais.

Mesmo com as dificuldades e problemas encontrados, este protótipo poderá ser utilizado em aplicações futuras de forma satisfatória com alguns pequenos ajustes para melhorar ainda mais o seu funcionamento.

REFERENCIAS

- ASTRAND, B.; BAERVELDT, A. J. **An agricultural mobile robot with vision-based perception for mechanical weed control**. Halmstad Sweden: Universidade Halmstad, 1999.
- BRITES, F. G.; SANTOS, V. P. A. **Motor de passos**. Niterói, Rio de Janeiro: UFF, 2008.
- BATISTA, A. V. R. **Robô irrigador multifuncional de baixo custo para Agricultura familiar (rirrig)**. 2016. 91 p. Dissertação (Mestrado em Engenharia Agrícola) – Universidade Federal do Ceará, Fortaleza, 2016.
- CARRARA, V. **Robótica**. Mogi das Cruzes, São Paulo: UBC. 1999.
- CRESTANA, S.; FRAGALLE, E. P. A trilha da quinta potência: um primeiro ensaio sobre ciência e inovação, agricultura e instrumentação agropecuária brasileira. **Revista Eixo**, Brasília, v. 1, n. 1, p. 6-27, 2012.
- FERREIRA, M.; IPUCHIMA, S.; SILVA, W. **Projeto de um braço robótico para fins didático com quatro graus de liberdade e uma garra**. Manaus, Amazonas: UNIP, 2012. Disponível em: <<https://www.ebah.com.br/content/ABAAAhBp4AJ/tcc-final-braco-robotico-rev-final>>. Acesso em: 28 jan. 2019.
- FLORES, P.; CLARO, J. C. P. **Cinemática de Mecanismos**. 1. ed. Coimbra:2007, 236 p.
- GROOVER; WEISS; NAGEL. O. **Robótica, Tecnologia e Programação**. 14. ed. São Paulo: 1989. xiv, 546 p.
- HACKENHAAR, N. M.; HACKENHAAR, C.; ABREU, Y. V. **Robotica na agricultura**. Campo Grande: 2014.
- INOVAÇÃO TECNOLÓGICA. **Robôs agrícolas farão colheita de frutas de forma autônoma**. Disponível em: <www.inovacaotecnologica.com.br/noticias/noticia.php?artigo=010180070830>. Acesso em: maio/2019.
- JUBELLI, F. J.; LIMA, M.; HORSTMANN, R. L. **Manipulador paralelo em configuração delta**. 2013. 105 p. Monografia (tecnólogo em Mecatrônica) – Instituto Federal de Ensino, Joinville, 2013
- KUTEKEN; S. **Ciência cognitiva e Robótica humanoide: Uma visão sobre literatura**. São Paulo: UEC. 2014.
- LOPES, A. M. **Robótica Industrial**. Universidade do Porto, 2001.
- Nof, S. Y. **Handbook of Industrial Robotics**, 2nd ed. 1999.
- NOGUEIRA, A. D.; TEXEIRA, V.; FREITAS, W. R. **Aplicação de robótica em processos de transporte industrial**. Lins, São Paulo: Centro Universitário de Lins, 2015.
- NORTON, R. L **Cinemática e Dinâmica dos Mecanismos**. 1ª Ed. Rio de Janeiro, 2010.

OLIVEIRA, R. R. **Uso do microcontrolador RSP8226 para automação residencial**. Universidade federal do Rio de Janeiro. 2017.

PANDA, K. G.; AGRAWAL, D.; NSHIMIYIMANA, A. Effects of environment on accuracy of ultrasonic sensor operates in millimeter range. **Perspectives in Science**, Silchar, 2016
Disponível em:
<<https://www.sciencedirect.com/science/article/pii/S221302091630163X>>

PAZOS, F. **Automação de Sistemas e Robótica**. 1ª Ed. Rio de Janeiro, 2002.

PIMENTA, T. T. **Controle de manipulador robótico**. Rio de Janeiro: Universidade católica do Rio de Janeiro, 2009.

ROMANO; V. F. **Robótica industrial: aplicação na indústria de manufatura e de processo**. 1ª Ed. 2002;

ROSARIO, João Maurício. **Princípios de Mecatrônica**. 1ª Ed. São Paulo, 2005.

SANTOS, V. M. F. **Robótica industrial**. Universidade de Aveiro, 2003;

SCHIRMER, Leo. **Modelagem do Robothrom** – um manipulador de barras paralelas. Joinville, Santa Catarina: Universidade do estado de Santa Catarina. 2005.

SOUZA, F. J.A.M. **Robotica**. 2005;

Souza, F. **Apostila de arduino**. 2013

TANKE, N. F. et al. **Automation of Hydroponic Installations using a Robot with Position Based Visual Feedback**. Pittsburgh: The Robotics Institute, Carnegie Mellon University, 2011.

TEXEURA, R. A. **Estratégia de controle de um manipulador puma 560 utilizando redes neurais artificiais**. 1998. 185 p. Dissertação (Mestrado em Engenharia Elétrica) – Universidade Federal de Minas Gerais, Belo Horizonte, 1998.

VALERIO, R; GARCIA, M. **Futuro da robótica**. IV Seminário multidisciplinar ENIAC. Vol 1, 2014.

APÊNDICE B – Algoritmo utilizado no Teste 3

```
//  
=====
```

```
// --- Bibliotecas ---  
#include <SoftwareSerial.h> //inclui biblioteca software serial  
#include <Servo.h> //inclui biblioteca para controle dos servo
```

```
//  
=====
```

```
// --- Objetos ---  
SoftwareSerial esp8266(5, 4); //pino TX do ESP8266 ligado no digital 5 e  
RX no digital 4  
Servo sv; //servo motor 1  
Servo sv1; //servo motor 2  
Servo sv2; //servo motor 3  
Servo sv3; //servo motor 4  
Servo sv4; //servo motor 5  
Servo sv5; //servo motor 6
```

```
//  
=====
```

```
// --- Constantes e Mapeamento de Hardware ---  
#define DEBUG true //DEBUG no serial monitor  
#define serv 11 //servo motor 1  
#define serv1 10 //servo motor 2  
#define serv2 9 //servo motor 3  
#define serv3 6 //servo motor 4  
#define serv4 5 //servo motor 5  
#define serv5 3 //servo motor 6
```

```
//  
=====
```

```
// --- Protótipo das Funções ---  
String sendData(String command, const int timeout, boolean debug);  
//
```

```

=====
// --- Variáveis Globais ---
int pos, //armazena posição do servo 1
pos1, //armazena posição do servo 1
pos2, //armazena posição do servo 2
pos3, //armazena posição do servo 1
pos4, //armazena posição do servo 1
pos5,
vel =10, //velocidade dos servos (100 -> 90° in s)(1-> 90° in 9 s)

posmin = 5, //posição mínima do servo 1
pos1min = 5, //posição mínima do servo 2
pos2min = 5, //posição mínima do servo 1
pos3min = 5, //posição mínima do servo 2
pos4min = 5, //posição mínima do servo 1
pos5min = 5,
posmax = 175, //posição máxima do servo 1
pos1max = 175, //posição máxima do servo 2
pos2max = 175, //posição máxima do servo 1
pos3max = 175, //posição máxima do servo 2
pos4max = 175, //posição máxima do servo 1
pos5max = 175; //posição máxima do servo 2

```

```
//
```

```

=====
// --- Configurações Iniciais ---

```

```
void setup()
```

```
{
```

```
// -- Configura Servos e Posições Iniciais --
```

```
sv.attach(serv);
```

```
sv1.attach(serv1);
```

```
sv3.attach(serv3);
```

```
sv2.attach(serv2);
```

```
sv4.attach(serv4);
```

```
sv5.attach(serv5);
```

```
sv1.write(pos1max);
```

```
sv2.write(pos2max);
```

```
// -- Inicia Comunicação Serial --
```

```
Serial.begin(9600);
```

```
esp8266.begin(19200);
```

```

// -- Configura WiFi --
// =====
sendData("AT+RST\r\n", 2000, DEBUG);
//reset no modulo
sendData("AT+CWMODE=1\r\n", 1000, DEBUG);
//modo station
sendData("AT+CWJAP=\"Nome da Rede\", \"Senha da Rede\"\r\n", 2000,
DEBUG); //conecta com rede Wifi (nome e senha)
while(!esp8266.find("OK"));
//aguarda até que haja conexão

sendData("AT+CIFSR\r\n", 1000, DEBUG);
//mostra endereço IP
sendData("AT+CIPMUX=1\r\n", 1000, DEBUG);
//permite múltiplas conexões
sendData("AT+CIPSERVER=1,80\r\n", 1000, DEBUG);
//inicia web server na porta 80

} //end setup

//
=====
=====
// --- Loop Infinito ---
void loop()
{

if (esp8266.available()) //verifica se existem dados disponíveis no
ESP8266
{
if (esp8266.find("+IPD, ")) //novo comando?
{
//Sim
String msg;
esp8266.find("?"); //movimenta cursor até
encontrar conteúdo da mensagem
msg = esp8266.readStringUntil(' '); //lê mensagem
String command = msg.substring(0, 3); //"sr1" = comando para o servo
#1 e "sr2" = comando para o servo #2
String valueStr = msg.substring(4); //os 3 caracteres seguintes
informam o ângulo desejado
int value = valueStr.toInt();
if (DEBUG) {
Serial.println(command);
Serial.println(value);
}
}
}
}

```

```

}
delay(100);

//-- controle do servo 1 --
if(command == "sr")
{
  //limita ângulo do servo
  if (value >= posmax) value = posmax;

  if (value <= posmin) value = posmin;

  sv.attach(serv); //conecta servo1
  while(pos != value)
  {
    if (pos > value)
    {
      pos -= 1;
      sv.write(pos);
      delay(100/vel);
    }
    if (pos < value)
    {
      pos += 1;
      sv.write(pos);
      delay(100/vel);
    }
  } //end while

} //end controle servo1

//-- controle do servo 3 --
if(command == "sr2")
{
  //limita ângulo do servo
  if (value >= pos2max) value = pos2max;

  if (value <= pos2min) value = pos2min;

  sv2.attach(serv2); //conecta servo3
  while(pos2 != value)
  {

```

```
    if (pos2 > value)
    {
        pos2 -= 1;
        sv2.write(pos2);
        delay(100/vel);
    }
    if (pos2 < value)
    {
        pos2 += 1;
        sv2.write(pos2);
        delay(100/vel);
    }
} //end while
```

```
} //end controle servo 3
```

```
//-- controle do servo 2 --
```

```
if(command == "sr1")
{
    //limita ângulo do servo
    if (value >= pos1max) value = pos1max;

    if (value <= pos1min) value = pos1min;

    sv.attach(serv1); //conecta servo2
    while(pos1 != value)
    {
        if (pos1 > value)
        {
            pos1 -= 1;
            sv.write(pos1);
            delay(100/vel);
        }
        if (pos < value)
        {
            pos1 += 1;
            sv1.write(pos1);
            delay(100/vel);
        }
    }
} //end while

} //end controle servo2
```

```

//-- controle do servo 4 --
if(command == "sr3")
{
    //limita ângulo do servo
    if (value >= pos3max) value = pos3max;

    if (value <= pos3min) value = pos3min;

    sv3.attach(serv3); //conecta servo3
    while(pos3 != value)
    {
        if (pos3 > value)
        {
            pos3 -= 1;
            sv3.write(pos3);
            delay(100/vel);
        }
        if (pos3 < value)
        {
            pos3 += 1;
            sv.write(pos3);
            delay(100/vel);
        }
    } //end while

} //end controle servo4

```

```

//-- controle do servo 5 --

if(command == "sr4")
{
    //limita ângulo do servo
    if (value >= pos4max) value = pos4max;

    if (value <= pos4min) value = pos4min;

    sv4.attach(serv4); //conecta servo5
    while(pos4 != value)
    {
        if (pos4 > value)
        {
            pos4 -= 1;
            sv4.write(pos4);

```

```
        delay(100/vel);
    }
    if (pos4 < value)
    {
        pos4 += 1;
        sv4.write(pos4);
        delay(100/vel);
    }
} //end while
```

```
} //end controle servo1
```

```
//-- controle do servo 6 --
```

```
if(command == "sr5")
```

```
{
```

```
    //limita ângulo do servo
```

```
    if (value >= pos5max) value = pos5max;
```

```
    if (value <= pos5min) value = pos5min;
```

```
    sv5.attach(serv5); //conecta servo6
```

```
    while(pos5 != value)
```

```
    {
```

```
        if (pos5 > value)
```

```
        {
```

```
            pos5 -= 1;
```

```
            sv5.write(pos5);
```

```
            delay(100/vel);
```

```
        }
```

```
        if (pos5 < value)
```

```
        {
```

```
            pos5 += 1;
```

```
            sv5.write(pos5);
```

```
            delay(100/vel);
```

```
        }
```

```
    } //end while
```

```
} //end controle servo 2
```

```
}
```

```
}
```

```
}
```



```
//
```

```
=====
```

```
=====
```

```
// --- Função ---
```

```
String sendData(String command, const int timeout, boolean debug)
```

```
{
```

```
String response = "";
```

```
esp8266.print(command);
```

```
long int time = millis();
```

```
while ( (time + timeout) > millis())
```

```
{
```

```
while (esp8266.available())
```

```
{
```

```
char c = esp8266.read();
```

```
response += c;
```

```
}
```

```
}
```

```
if (debug)
```

```
{
```

```
Serial.print(response);
```

```
}
```

```
return response;
```

```
} //end sendData
```

APÊNDICE C – Algoritmo de criação da pagina web

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<meta name="author" content="WR Kits">
<title>Servos WiFi</title>
<script src="jquery.js"></script>
<script>
$.ajaxSetup({timeout:1000});
function mouseUpSr1(angle)
{
  TextVar = myform2.inputbox.value;
  ArduinoVar = "http://" + TextVar + ":80";
  $.get( ArduinoVar, { "sr1": angle } ) ;
  {Connection: close};
}
function mouseUpSr2(angle)
{
  TextVar = myform2.inputbox.value;
  ArduinoVar = "http://" + TextVar + ":80";
  $.get( ArduinoVar, { "sr2": angle } ) ;
  {Connection: close};
}
function mouseUpSr3(angle)
{
  TextVar = myform3.inputbox.value;
  ArduinoVar = "http://" + TextVar + ":80";
  $.get( ArduinoVar, { "sr3": angle } ) ;
  {Connection: close};
}
function mouseUpSr4(angle)
{
  TextVar = myform2.inputbox.value;
  ArduinoVar = "http://" + TextVar + ":80";
  $.get( ArduinoVar, { "sr4": angle } ) ;
  {Connection: close};
}
function mouseUpSr5(angle)
{
  TextVar = myform2.inputbox.value;
  ArduinoVar = "http://" + TextVar + ":80";
  $.get( ArduinoVar, { "sr5": angle } ) ;
  {Connection: close};
}
function mouseUpSr6(angle)
{
  servos_wifi
  TextVar = myform2.inputbox.value;
  ArduinoVar = "http://" + TextVar + ":80";
```

```
$.get( ArduinoVar, { "sr6": angle } ) ;
{Connection: close};
}
</script>
</head>
<body>
<h1> <b> Controle de Servos por WiFi - WR Kits </b> </h1>
<div style="margin: 0; width:600px; height:90px;">
<FORM NAME="myform2" ACTION="" METHOD="GET">
<h2> Host WiFi: </h2>
<INPUT TYPE="text" NAME="inputbox" VALUE="192.068.1.4">
</FORM>
</div>
<h2> Servo 1 </h2> <input type="range" min="15" max="175"
onmouseup="mouseUpSr1(this.value)" />
<h2> Servo 2 </h2>
<input type="range" min="15" max="175"
onmouseup="mouseUpSr2(this.value)" />
<h2> Servo 3 </h2> <input type="range" min="15" max="175"
onmouseup="mouseUpSr3(this.value)" />
<h2> Servo 4 </h2>
<input type="range" min="15" max="175"
onmouseup="mouseUpSr4(this.value)" />
<h2> Servo 5 </h2> <input type="range" min="15" max="175"
onmouseup="mouseUpSr5(this.value)" />
<h2> Servo 6 </h2>
<input type="range" min="15" max="175"
onmouseup="mouseUpSr6(this.value)" />
</body>
</html>
```