



UNIVERSIDADE ESTADUAL DE CAMPINAS  
Faculdade de Tecnologia – FT



VINÍCIUS DESTRI GALAFASSE

## SISTEMA DE *FORCE FEEDBACK* APLICADO EM UMA GARRA ROBÓTICA

Limeira, 2021

**Vinícius Destri Galafasse**

**SISTEMA DE *FORCE FEEDBACK* APLICADO EM UMA  
GARRA ROBÓTICA**

Trabalho de conclusão de curso apresentado à Faculdade de Tecnologia da Universidade Estadual de Campinas como parte dos requisitos para a obtenção do título de Engenheiro de Telecomunicações.

**ORIENTADOR: PROF<sup>a</sup> DR.<sup>a</sup> TALIA SIMÕES DOS SANTOS XIMENES**

Este exemplar corresponde à versão final do Trabalho de Conclusão de curso defendido por Vinícius Destri Galafasse sob orientação da Prof<sup>a</sup>. Dr<sup>a</sup>. Talia Simões dos Santos Ximenes.

**Limeira, 2021**

Ficha catalográfica  
Universidade Estadual de Campinas  
Biblioteca da Faculdade de Tecnologia  
Luiz Felipe Galeffi - CRB 8/10385

G13s Galafasse, Vinícius Destri, 1998-  
Sistema de *force feedback* aplicado em uma garra robótica / Vinícius Destri Galafasse. – Limeira, SP : [s.n.], 2021.

Orientador: Talía Simões dos Santos Ximenes.  
Trabalho de Conclusão de Curso (graduação) – Universidade Estadual de Campinas, Faculdade de Tecnologia.

1. Robôs móveis. 2. Robôs - Sistemas de controle. 3. Microcontroladores. I. Ximenes, Talía Simões dos Santos, 1980-. II. Universidade Estadual de Campinas. Faculdade de Tecnologia. III. Título.

Informações adicionais, complementares

**Título em outro idioma:** Force feedback system applied over robotic clamp

**Palavras-chave em inglês:**

Mobile robots

Robots - Control systems

Microcontrollers

**Titulação:** Bacharel

**Banca examinadora:**

Talía Simões dos Santos Ximenes [Orientador]

Francisco José Arnold

Rangel Arthur

**Data de entrega do trabalho definitivo:** 24-11-2021



UNIVERSIDADE ESTADUAL DE CAMPINAS  
Faculdade de Tecnologia – FT



VINÍCIUS DESTRI GALAFASSE

## SISTEMA DE *FORCE FEEDBACK* APLICADO EM UMA GARRA ROBÓTICA

BANCA EXAMINADORA:

Talía Simões dos Santos Ximenes

Francisco José Arnold

Rangel Arthur

A ata de defesa com as respectivas assinaturas dos membros da banca encontra-se no processo de vida acadêmica do aluno

Limeira, novembro de 2021

## **AGRADECIMENTOS**

Agradeço a Professora Dra. Talia Simões dos Santos Ximenes por abraçar a ideia deste projeto e fornecer toda a orientação e suporte necessário como auxílio ao meu desenvolvimento.

Agradeço a Unicamp e a todos os professores do curso de Engenharia de Telecomunicações pela oportunidade de crescer e aprender dentro deste ambiente acadêmico ao longo dos meus anos de formação.

Agradeço meus pais Denise Destri Galafasse e José Roberto Galafasse por todo o suporte, educação e carinho que sempre me deram, por acreditarem na minha capacidade e me ajudarem diretamente e indiretamente no meu desenvolvimento e neste projeto.

Agradeço a Jéssica Severiano Soares por todo o suporte e carinho que me ajudaram a passar por esta etapa com mais tranquilidade.

Agradeço ao Fabrício Cravo Gomes, Matheus Costa Crivelaro e Rafael Pinto Dias por sempre estarem comigo caminhando juntos neste curso e em todas as minhas principais memórias desta etapa da minha vida.

Agradeço a Deus pela minha saúde, força e motivação para sempre seguir em frente.

## RESUMO

O controle de atuadores robóticos está presente em muitas áreas. Buscando explorar novas e mais precisas interações entre o ser humano e sistemas mecatrônicos, este trabalho consiste no desenvolvimento de protótipo de uma garra mecânica controlada remotamente por um atuador manual para explorar a tecnologia de *Haptic Feedback*. *Haptic Feedback* é um tipo de tecnologia utilizada para estimular a função sensorial de tato. Essa é uma área de recentes estudos e testes principalmente dentro de equipamentos de lazer e possivelmente equipamentos hospitalares. É uma primeira tentativa em expandir a interação entre os seres humanos e as máquinas para além de somente a visão e a audição como sentidos predominantes. O sistema principal deste trabalho é composto por uma garra robótica com transdutores de carga, controle e cálculo da conversão da força aplicada através de um microcontrolador. Foi projetado um mecanismo que simule dinamicamente a força aplicada no atuador operado manualmente. O sistema permitiu a simulação de *Force Feedback* de maneira rápida e estável para até 3,5 quilos aplicados ao transdutor da garra, atuando concomitantemente 2 servo motores e exibindo a massa aplicada no monitor serial da Arduino IDE.

**Palavras chave:** *Force Feedback Clamp*, *Haptic Feedback*, Atuador, Mão robótica, Garra robótica, Microcontrolador, Transdutor de carga, Sensor, Robô, Tecnologia.

## LISTA DE FIGURAS

|                                                                                                                                 |    |
|---------------------------------------------------------------------------------------------------------------------------------|----|
| Figura 1 - Ponto focal produzido pelas ondas ultrassônicas da tecnologia Ultraleap .....                                        | 14 |
| Figura 2 - Diagrama simplificado da estrutura interna de um microcontrolador [14] .....                                         | 19 |
| Figura 3 - Diagrama de blocos representando o funcionamento do sistema completo .....                                           | 23 |
| Figura 4 – Fluxograma do código do sistema .....                                                                                | 25 |
| Figura 5 - Fluxo geral do código do sistema .....                                                                               | 25 |
| Figura 6 - Diagrama do sistema de controle central do protótipo integrando o microcontrolador ao transdutor de carga.....       | 26 |
| Figura 7 - Primeiro teste de controle do servo motor com um potenciômetro realizado no Tinkercad .....                          | 27 |
| Figura 8 - Função utilizada para o controle do motor da garra robótica .....                                                    | 29 |
| Figura 9 - Fonte de alimentação para protoboard utilizada no protótipo.....                                                     | 30 |
| Figura 10 - Fonte de alimentação utilizada para alimentar a fonte da <i>protoboard</i> .....                                    | 31 |
| Figura 11 - Esquema de montagem da balança utilizando o transdutor de carga de 5 quilos .....                                   | 32 |
| Figura 12 - Esquema elétrico de uma ponte de Wheatstone [27] .....                                                              | 33 |
| Figura 13 - Módulo conversor ADC 24 bits HX711 .....                                                                            | 34 |
| Figura 14 - Esquema de conexão da célula de carga com o HX711 [31].....                                                         | 35 |
| Figura 15 - Código utilizado para a calibração da célula de carga utilizada no projeto.....                                     | 37 |
| Figura 16 - Modificação do código com o fator de calibragem do transdutor de carga.....                                         | 37 |
| Figura 17 - Função que captura e exibe o valor em kg aplicado ao transdutor .....                                               | 38 |
| Figura 18 - Sistema idealizado para realizar o <i>Force Feedback</i> no gatilho.....                                            | 39 |
| Figura 19 - Balança confeccionada com madeira e a célula de carga de 5kg. Arduino Uno utilizado como referência de escala ..... | 40 |
| Figura 20 - Protótipo totalmente montado .....                                                                                  | 41 |
| Figura 21 - Valor máximo de 3,5kg aplicado sobre o protótipo.....                                                               | 43 |
| Figura 22 - Trecho do código da biblioteca do HX711 que desabilita as interrupções .....                                        | 44 |
| Figura 23 - Trecho do código da biblioteca do HX711 que reabilita as interrupções .....                                         | 45 |
| Figura 24 - Trecho do código da biblioteca do HX711 com a explicação da necessidade de desabilitar as interrupções.....         | 45 |
| Figura 25 - Sistema de detecção de movimento com um piezelétrico .....                                                          | 50 |
| Figura 26 - Sistema de detecção de movimento com uma argola e um contato elétrico .....                                         | 51 |

## LISTA DE ABREVIações

|      |                                       |
|------|---------------------------------------|
| ADC  | Conversor analógico-digital           |
| C.I. | Circuito Integrado                    |
| CM   | Centro de Massa                       |
| FFC  | <i>Force Feedback Clamp</i>           |
| FF   | <i>Force Feedback</i>                 |
| HF   | <i>Haptic Feedback</i>                |
| IDE  | Ambiente de Desenvolvimento Integrado |
| MC   | Microcontrolador                      |
| MMHF | <i>Multi-modal Haptic Feedback</i>    |



## SUMÁRIO

|                                                                                       |           |
|---------------------------------------------------------------------------------------|-----------|
| <b>1 – INTRODUÇÃO .....</b>                                                           | <b>10</b> |
| <b>1.1- Haptic Feedback .....</b>                                                     | <b>11</b> |
| 1.1.1– Haptic Feedback nas áreas de lazer e entretenimento.....                       | 12        |
| 1.1.2 – Haptic Feedback na área médica .....                                          | 15        |
| <b>1.2– Objetivos.....</b>                                                            | <b>17</b> |
| <b>2 – APROFUNDAMENTO TEÓRICO .....</b>                                               | <b>18</b> |
| <b>2.1 – Microcontroladores.....</b>                                                  | <b>18</b> |
| <b>2.2 – Arduino .....</b>                                                            | <b>19</b> |
| <b>2.3 – Limitações do Arduino UNO para sinais analógicos.....</b>                    | <b>20</b> |
| <b>2.4 – Sistemas mecatrônicos .....</b>                                              | <b>21</b> |
| <b>3 – DESENVOLVIMENTO .....</b>                                                      | <b>23</b> |
| <b>3.1 – Sistema de controle.....</b>                                                 | <b>24</b> |
| <b>3.2 – Sistema de microservo motor.....</b>                                         | <b>28</b> |
| <b>3.3 – Fonte de alimentação do sistema.....</b>                                     | <b>30</b> |
| <b>3.4 – Sistema detector de força com o transdutor de carga.....</b>                 | <b>32</b> |
| <b>3.5 – Sistema de force feedback aplicado ao gatilho .....</b>                      | <b>38</b> |
| <b>4 – RESULTADOS E DISCUSSÃO .....</b>                                               | <b>40</b> |
| <b>4.1 – Conflito entre &lt;Servo.h&gt; e “HX711.h” .....</b>                         | <b>43</b> |
| <b>5 – CONCLUSÕES.....</b>                                                            | <b>47</b> |
| <b>6 – PERSPECTIVAS DE CONTINUAÇÃO.....</b>                                           | <b>48</b> |
| <b>6.1 – Modelagem da garra robótica [37].....</b>                                    | <b>48</b> |
| <b>6.2 – Dimensionamento do sistema da braçadeira acoplada ao gatilho .....</b>       | <b>49</b> |
| <b>6.3 – Mecanismo para retorno do gatilho e afrouxamento do Force Feedback .....</b> | <b>49</b> |
| <b>6.4 – Dimensionamento do motor da garra .....</b>                                  | <b>52</b> |
| <b>6.5 – Aplicação de tecnologias wireless entre os módulos do sistema.....</b>       | <b>52</b> |
| <b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>                                                | <b>54</b> |

## 1 – INTRODUÇÃO

A interação entre o homem e a máquina vem crescendo há décadas. Essa interação pode ser agrupada em três grandes fases [1]: na 1ª fase, com o surgimento de máquinas calculadoras, os primeiros computadores e o eventual desenvolvimento ao modelo que se tem hoje. O tipo de comando digital sempre foi dominante, obtendo-se respostas ao pressionar um botão ou uma tecla.

Na 2ª fase foram introduzidos elementos que intermediam os sentidos do homem para a máquina. Dispositivos biométricos, telas interativas *touch screen*, transcrição de voz para texto e sensores de captação de movimentos corporais são algumas das tecnologias que se alavancaram dentro dessa fase. Diferente de outras áreas onde uma nova tecnologia substitui a antiga, tratando-se de comandos de máquina, essas novas alternativas convivem até hoje com o tradicional modelo de entradas e respostas digitais da 1ª fase.

A 3ª fase iniciou com a exploração de uma área considerada puramente ficção até um passado recente. A tecnologia da vez é realizar a interação entre o homem e a máquina por ondas cerebrais, pelo pensamento. Principalmente na área médica há diversos protótipos e sistemas criados para realizar o movimento de próteses pela mente.

No mundo atual é impossível imaginar desenvolvimento tecnológico sem envolver algum tipo de robô, software, sistema de controle, equipamento etc. onde, muitas vezes, sequer é necessário a atuação do ser humano nas tarefas executadas. Entretanto, ainda há diversos ramos onde falta confiabilidade ou profundo desenvolvimento dos equipamentos utilizados para deixá-los sem supervisão ou intervenção manual, ou área na qual é explorada justamente a interação entre o homem e os sistemas robóticos puramente para entretenimento.

Nos tempos modernos, a automação e utilização de atuadores robóticos está presente em muitas áreas. Pensando principalmente na área médica, onde já são praticadas cirurgias à distância através de máquinas e até mesmo *joysticks* [2,3], torna-se interessante apresentar um nível extra de precisão e resposta do que está acontecendo do outro lado do sistema, principalmente por ser uma atividade extremamente invasiva e delicada envolvendo vidas.

No ramo do entretenimento há uma vertente que explora a imersão dos seres humanos com a tecnologia, aplicando sensações e experiências para além do mundo

real. Equipamentos de realidade expandida, realidade virtual, salas com sistemas acústicos extremamente elaborados proporcionam essa maior imersão através dos sentidos da visão e da audição. Controles anatômicos, *joysticks* com motores vibratórios, detectores de movimento, assentos inteligentes etc. proporcionam a imersão através do tato e da cinética.

Tomando o conceito de uma mão robótica e um nível de controle relacionado com a 2ª fase de interações, este trabalho busca iniciar uma linha de desenvolvimento no aperfeiçoamento da resposta fornecida de um equipamento robótico real para uma pessoa, projetando um sistema que transmita a força que uma garra robótica exerce sobre a carga, corpo etc. de volta ao operador do sistema, visando obter informações instantâneas e dinâmicas de possíveis relevos, deformações e irregularidades na resistência do corpo, e principalmente transmitir a noção da força que está sendo aplicada.

### **1.1- Haptic Feedback**

O estudo de *Force Feedback* abordado neste trabalho pode ser considerado uma das vertentes da tecnologia conhecida como *Haptic Feedback* (HF). O HF, ou *Feedback Tátil*, é um tipo de tecnologia utilizada para estimular a função sensorial de tato. Essa é uma área de recentes estudos e testes principalmente dentro de equipamentos de lazer, e é uma primeira tentativa em expandir a interação entre os seres humanos e as máquinas para além de somente a visão e a audição como sentidos predominantes.

O sistema somatossensorial se refere a todo o mecanismo de receptores e processamento cerebral presente em seres vivos responsável por identificar os diferentes tipos de sensações táteis. Esse sistema pode ser dividido em 12 tipos de receptores especializados em captar as diferentes sensações de vibração, pressão, temperatura, textura, movimento etc.

As tecnologias voltadas para HF podem ser divididas em 5 categorias [4]:

- Vibro táteis: pequenos motores criam vibrações para estimular principalmente as mãos e dedos. Muito presente em jogos para *smartphones* e cada vez mais integrada em controles de consoles, como o do *PlayStation 5* e do *Nintendo Switch*;

- Ultrassônicas: são dispositivos que produzem ondas ultrassônicas para gerar diferentes micro impactos e sensações de pressão nas mãos do usuário, não necessitando de contato direto entre a pessoa e o equipamento;
- Micro fluidos: são tecnologias que utilizam ar ou líquidos para produzir diferentes volumes e temperaturas em pequenas cápsulas ou blísteres para estimular a pele;
- Controle de força: equipamentos que geram resistência física, tração ou provocam impactos no usuário, podendo chegar a sensações que atingem o corpo inteiro. Geralmente são sistemas mais robustos, caros e que demandam maior cuidado para não passar a linha tênue que separa um *force feedback* proveitoso de uma sensação de dor;
- Superfícies táteis: simulam diferentes tipos de atrito para provocar sensações de movimento e resistência nas mãos do usuário.

O sistema de FF desenvolvido neste trabalho pode ser categorizado como controle de força, e conforme apresentado acima, as justificativas para ser um protótipo de pequena escala são fortalecidas pelo maior custo e cuidado necessário a ser tomado para esse tipo de tecnologia.

#### *1.1.1– Haptic Feedback nas áreas de lazer e entretenimento*

A área de entretenimento é uma das maiores candidatas a receber projetos e protótipos para explorar novas tecnologias, principalmente quando tais tecnologias não representam ameaças ou maiores complicações aos seres humanos diante de possíveis maus funcionamentos.

A principal maneira como a indústria de entretenimento pode utilizar ferramentas e equipamentos tecnológicos como sua aliada é através de inovações que aumentem a imersão de seus usuários e/ou telespectadores com o sistema. Imersão, apesar de ser um termo simples e subentendido por praticamente todos, é algo um tanto subjetivo e que traz consigo muitas portas de possibilidades.

Imersão no entretenimento é, muitas vezes, atingida através de equipamentos ou sistemas eletrônicos e computacionais que geram respostas visuais e auditivas ao usuário final. Tanto a visão como a audição são sentidos extremamente importantes

para a que os seres humanos sintam o que ocorre ao seu redor. Apoiando-se nessa ideia, são criados sistemas audiovisuais que permitem expandir essa noção de sensação para elementos além da realidade física e gerar o que será chamado neste trabalho como imersão tecnológica.

Nos últimos anos, diversos equipamentos foram desenvolvidos para aumentar a imersão tecnológica no mundo de entretenimento. Como ótimo exemplo para um sistema visual pode-se citar os inúmeros óculos de realidade virtual que permitem ao indivíduo vivenciar um mundo ou situação virtual como se estivesse dentro daquele ambiente. Para sistemas sonoros, inúmeras empresas focam em produzir *headsets* com tecnologias refinadas para espalhar o som ao redor da cabeça, eliminar qualquer barulho ou ruído externo através de sistemas de cancelamento de ruído passivo e/ou ativo, fones anatômicos e fabricados com materiais leves e que geram o mínimo de desconforto ao ponto da pessoa “esquecer” que está usando o *headset*. Outra aproximação para tecnologias sonoras são salas equipadas com sistemas de autofalantes extremamente especializados e controlados por software para produzir sons ambientes, passando a sensação que tais sons estão presentes no mundo real.

Como é possível notar pela ênfase até o momento, por muito tempo, tecnologias audiovisuais foram as principais exploradas para o mundo do lazer e entretenimento, entretanto visão e audição não são os únicos sentidos que podem ser explorados para expandir a imersão tecnológica. De modo geral, utilizar odores para estimular o olfato das pessoas não é de grande interesse, pois tal sentido é extremamente volátil entre indivíduos e pode gerar complicações facilmente em casos de manifestações patológicas, como alergias, além da alta necessidade de esterilização/manutenção para evitar a contaminação de tais sistemas com fungos ou outros agentes invasores que podem ser transmitidos pelo ar. Complicações similares ocorrem em sistemas que interagem com o paladar. Sendo assim, resta o tato como candidato para explorar no universo da imersão tecnológica.

Tecnologias de HF são uma nova tendência de pesquisa principalmente dentro do mercado de entretenimento. Por muito tempo a principal maneira de transmitir sensações de tato foi através da utilização de motores para produzir vibrações e movimentos em controles ou acentos. Apesar da utilização desses motores já estar

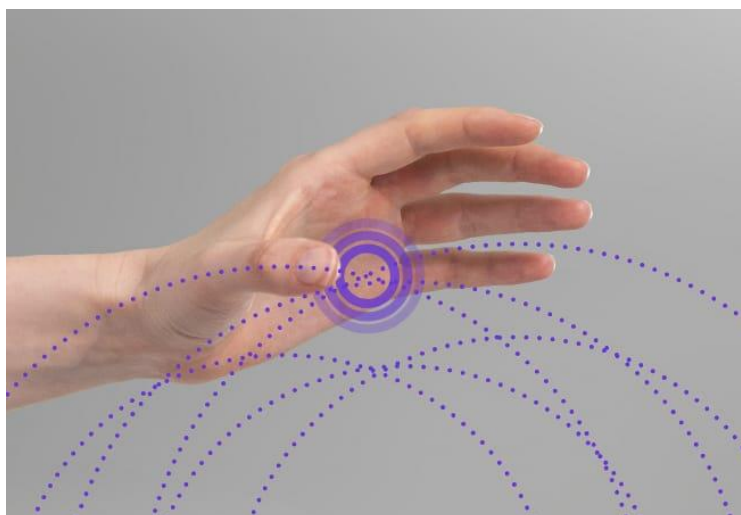
consolidada há um bom tempo, novas maneiras para refinar tais sistemas estão em constante pesquisa e desenvolvimento.

Um dos exemplos mais atuais referentes a esse mundo é o controle do videogame *Playstation 5*. O controle *DualSense* do console é equipado com diversos micromotores chamados de *voice-coil actuators*, tecnologia utilizada em autofalantes que permite as diferentes vibrações necessárias para produzir toda a faixa de frequência audível pelo ser humano [5].

Através dessa tecnologia, o *DualSense* é capaz de emular diversas sensações de terrenos, texturas e movimentos nos dedos do jogador. Movimentos em alta frequência passam a sensação de andar sobre o gelo ou derrapar. Ciclos entre altos e baixos movimentos nos botões geram a sensação de caminhar sobre algo bambo, como uma ponte suspensa.

Outra tecnologia recente focada em HF é o produto em desenvolvimento pela empresa Ultraleap [6], que busca através de equipamentos ultrassônicos produzir virtualmente sensações de tato na mão que recebe essas ondas. A Figura 1 mostra o ponto focal produzido pelas ondas ultrassônicas dessa tecnologia.

**FIGURA 1 - PONTO FOCAL PRODUZIDO PELAS ONDAS ULTRASSÔNICAS DA TECNOLOGIA ULTRALEAP**



O sistema da Ultraleap utiliza um equipamento composto por uma malha de pequenos autofalantes que emitem ondas ultrassônicas em diferentes frequências e instantes de tempo. De acordo com o algoritmo utilizado para produzir padrões nessas

ondas, é possível obter um ponto focal onde todas se coincidem, e é nesse ponto que ocorre a sensação de toque na palma da mão. Através de dezenas de pontos focais gerados pela malha, são produzidos padrões que emulam diferentes movimentos e sensações, como a varredura da mão, formigamento, cliques, abrir e fechar círculos etc.

### *1.1.2 – Haptic Feedback na área médica*

A área médica sempre é uma das grandes motivadoras para avanços tecnológicos. Pesquisas e criações de medicamentos ao longo da história possibilitaram melhores e maiores alternativas para o tratamento de enfermidades nos seres humanos, assim como o desenvolvimento de ferramentas cirúrgicas precisas como o bisturi, bisturi elétrico, estetoscópio e suturas absorvíveis pelo organismo [7].

O desenvolvimento e avanço tecnológico em equipamentos que auxiliam o dia a dia no mundo hospitalar, como desfibriladores, máquinas de raio X, sondas etc. torna inegável o fato de que tecnologias de ponta são facilmente encontradas na área médica.

Diversas máquinas e equipamentos hospitalares foram desenvolvidos com o intuito tanto de auxiliar os profissionais em exames e procedimentos delicados quanto em evitar ao máximo o contato direto entre esses profissionais e os pacientes. Em muitas situações é interessante que haja esse distanciamento humano buscando propiciar um ambiente altamente estéril e controlado, livre de ruídos e movimentações desnecessárias.

Há um tempo que existem máquinas na área médica capazes de executar cirurgias à distância. Tais máquinas funcionam com um profissional cirurgião especializado em controlar os atuadores do sistema responsáveis por transmitir as ações que as pinças, bisturis, agulhas, alicates etc. devem desempenhar no corpo do paciente. Um exemplo de tais equipamentos é o utilizado no procedimento de colecistectomia robótica.

A colecistectomia consiste na cirurgia para retirada da vesícula não saudável do paciente. Inicialmente, tal procedimento era 100% manual e exigia cortes invasivos no abdômen, trazendo alto risco de infecções, hemorragias e complicações pós-

operatório. Há aproximadamente três décadas, esse procedimento foi substituído pela laparoscopia [8] e videolaparoscopia, onde são utilizadas ferramentas inseridas através de pequenas incisões realizadas no abdômen do paciente, e todo o processo é visualizado e operado externamente, reduzindo drasticamente o nível de invasão desse procedimento e consequentemente os possíveis riscos da colecistectomia.

Mais recentemente, a laparoscopia começa a ser substituída pela colecistectomia robótica [9]. Essa cirurgia consiste na atuação indireta do cirurgião no paciente. Através de uma mesa de controle dotada de *displays*, atuadores, *joysticks* e botões de controle, o profissional opera o equipamento robótico instalado na mesa de cirurgia

Com o auxílio dessa tecnologia, a colecistectomia robótica fornece extrema precisão em movimentos, elimina riscos de tremores, pressão, proporciona um ambiente mais estéril e torna a remoção da vesícula um procedimento ainda menos invasivo e mais seguro.

Como é possível perceber, avanços tecnológicos na área médica e equipamentos que permitem exames e operações indiretas trazem novos níveis de precisão, *feedback* e segurança. Entretanto, não são todas as situações que podem ser realizadas totalmente por equipamentos robóticos no paciente. Por exemplo, muitos exames para identificar anomalias, cistos e tumores dentro do tecido epitelial, liposo ou muscular do paciente precisam ser feitos através do apalpamento do corpo para identificar irregularidades de dureza e textura. Visando avançar tecnologicamente nesse tipo de necessidade é que são estudados sistemas de HF para equipamentos médicos.

Os sistemas de HF para equipamentos de cirurgias indiretas estão em desenvolvimento e avaliações. Em sua escala mais simples, a maior parte desse *feedback* é referente a força aplicada pelas partes robóticas no corpo do paciente. Essa força é replicada nos atuadores que o profissional cirurgião está controlando. Os maiores desafios desse tipo de tecnologia, tratando-se de um ambiente hospitalar, são a esterilização, custo, anatomia e dimensões desses equipamentos e sensores táteis.



Estudos revelam que a utilização de tecnologias de HF para essas operações teleguiadas reduz consideravelmente falhas cirúrgicas por fornecer maior resposta ao cirurgião do que está acontecendo com o corpo do paciente. Alguns protótipos foram feitos para implementar além da resposta da força aplicada, sensações térmicas capturadas pelo equipamento de operação, porém tais testes não demonstraram impactos significativos.

Qualquer tentativa de implementar mais de uma maneira de transmitir sensações táteis para o operador pode ser classificada dentro do chamado *multi-modal haptic feedback* (MMHF). Apesar de estudos e testes de equipamentos que proporcionem MMHF para cobrir as sensações de temperatura, texturas, viscosidade etc. [10, 11, 12], o HF apenas para controle de força é o de mais fácil implementação e com maiores ganhos de desempenho, portanto é o mais abordado até o momento [13].

## **1.2– Objetivos**

- Estudar a área da ciência voltada para tecnologias de HF
- Encontrar as principais aplicações e motivações para o desenvolvimento desse tipo de tecnologia
- Aplicar conhecimentos de microcontroladores, eletroeletrônica e mecatrônica para desenvolver um protótipo de HF baseado no controle de uma garra mecânica com *feedback* de força para a mão/dedo da pessoa
- Coletar e analisar os resultados obtidos, buscando corrigir e/ou otimizar partes do funcionamento
- Propor o dimensionamento de um sistema funcional em maior escala com alternativas para as tecnologias empregadas

## **2 – APROFUNDAMENTO TEÓRICO**

O desenvolvimento deste trabalho foi fundamentado com o estudo das áreas que envolvem as tecnologias utilizadas no protótipo. Para um projeto de *Force Feedback* utilizando um atuador real, neste caso, a garra mecânica, é necessário expandir a justificativa utilizando-se dos conceitos de sistemas mecatrônicos/robótica.

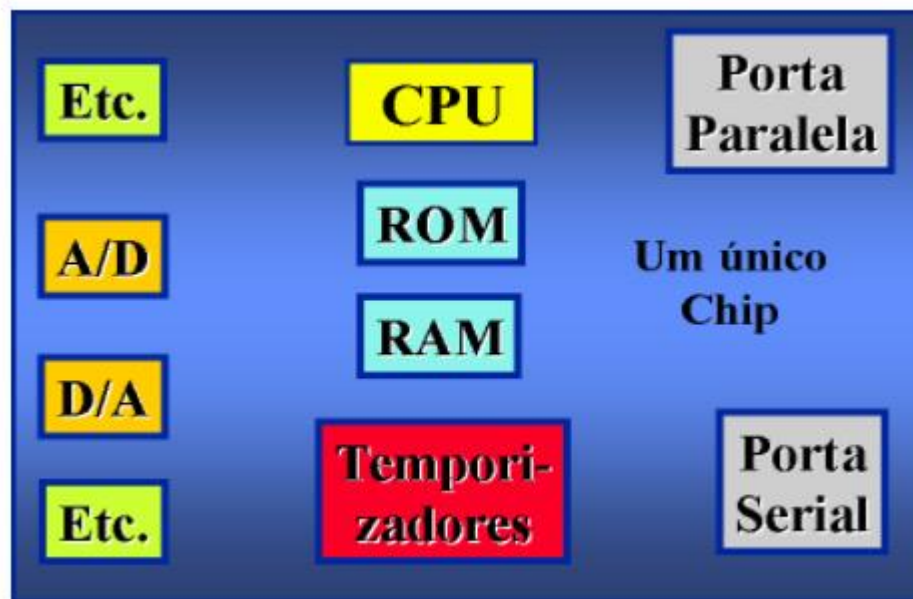
Em relação ao sistema de controle desenvolvido no projeto, foram utilizados os conceitos e materiais mais recentes elaborados sobre o funcionamento de microcontroladores, circuitos conversores de sinal analógico-digital e de transdutores.

### **2.1 – Microcontroladores**

Microcontroladores são circuitos integrados (CI) compostos por um processador, memória e periféricos codificáveis para funcionarem como entradas ou saídas. Esses dispositivos são amplamente utilizados em aplicações de sistemas embarcados, como brinquedos, relógios digitais, eletrodomésticos, controles remotos, sistemas automotivos e ferramentas eletromecânicas. De modo geral, um microcontrolador pode ser esquematizado conforme Figura 2.

Conforme a Figura 2, observa-se que um microcontrolador nada mais é que um simples computador completo com todos os componentes e mecanismos necessários para sua programação e funcionamento conforme desejado. Dentro desses CIs há um microprocessador responsável pelo controle e gerenciamento de todas as instruções programadas e executadas no sistema.

FIGURA 2 - DIAGRAMA SIMPLIFICADO DA ESTRUTURA INTERNA DE UM MICROCONTROLADOR [14]



## 2.2 – Arduino

O microcontrolador utilizado neste trabalho é o ATmega328, utilizado na placa controladora popularmente conhecida como Arduino UNO. O Arduino é uma plataforma de computação física embarcada utilizada para a prototipagem de sistemas eletromecânicos [15].

Uma placa Arduino pode interagir com o ambiente através de seu hardware e pode ser conectada a um computador ou rede para sua programação e comunicação de dados. Dessa forma, essas placas são ideais para o desenvolvimento de protótipos relacionados a automação de sistemas eletrônicos

Por ser uma plataforma *open source*, os sistemas Arduino possuem extenso suporte de uma grande comunidade mundial de desenvolvedores e entusiastas, e a cada dia surgem novos softwares e hardwares que facilitam e expandem as aplicações da família de microcontroladores suportadas por esses sistemas [16].

### **Arduino UNO (especificações):**

- Microcontrolador: ATmega328
- Tensão de operação: 5V
- Tensão de entrada (recomendado): 7-12V
- Tensão de entrada (limite): 6-20V
- Pinos I/O digitais: 14 (6 PWM)
- Pinos de entrada analógica: 6
- Corrente DC para pinos I/O: 40mA
- Corrente DC para pinos 3,3V: 50mA
- Memória Flash: 32KB (0,5KB para *bootloader*)
- SRAM: 2KB
- EEPROM: 1KB
- Clock: 16MHz
- Comprimento: 68,58mm
- Largura: 53,34mm
- Peso: 150g

As placas Arduino contam com inúmeros módulos e periféricos projetados com o intuito de facilitar o desenvolvimento de protótipos nas mais diversas áreas. Esses módulos geralmente contam com especificações compatíveis com as do microcontrolador e possuem bibliotecas de software codificadas para fácil implementação dentro do Ambiente de Desenvolvimento Integrado (IDE) do Arduino [17].

### **2.3 – Limitações do Arduino UNO para sinais analógicos**

Para aplicações com entradas analógicas, o Arduino UNO conta com 6 pinos de entrada analógica, e por padrão, utiliza uma resolução de 1023 (10 bits) para valores de 0 a 5 Volts. Esses pinos podem ser lidos através da função `analogRead()`. Para este trabalho, esses pinos analógicos serão utilizados na leitura do sensor de

movimento da mão, o qual foi idealizado e projetado como um potenciômetro acoplado ao sistema de gatilho analógico manual.

Para produzir um sinal analógico de saída, o UNO não possui hardware que gere um sinal puramente analógico. Ao invés disso, dispõe de 6 pinos com funcionalidade PWM (*Pulse Width Modulation*) que através da função `analogWrite`, produz um sinal PWM com *Duty Cycle* configurável, simulando o comportamento de uma saída analógica. Somente as placas MKR, Nano 33 IoT, Zero e Due contam com saída analógica pura, porém não houve a necessidade de utilização para este projeto.

Diferentemente dos pinos de entrada analógica que possuem resolução de 1023, os pinos PWM funcionam com o mapeamento de 0 a 5 Volts dentro da escala de 255 (8 bits ou 1 byte). É importante que essa informação seja levada em consideração para compreender que os valores obtidos através do PWM não serão tão precisos quanto valores analógicos de entrada no sistema. Para este trabalho, sinais de saída analógicos foram estudados visando duas funcionalidades:

- Atuação do motor da garra robótica;
- Geração do sinal responsável pelo controle de tração no gatilho analógico manual.

Para o desenvolvimento de uma saída analógica pura onde não é possível alterar o microcontrolador utilizado para outro que tenha suporte nativo a essa tecnologia, podem ser utilizadas duas alternativas [18]:

- Adicionar um chip DAC (*Digital to Analog Conversor*) externo, como o MCP4821/2, MCP4725 etc.;
- Usar um filtro passa baixas (malha RC) na saída do pino PWM [19].

## **2.4 – Sistemas mecatrônicos**

Mecatrônica é um termo utilizado para sinalizar o estudo e desenvolvimento da área da ciência que integra e comporta ambos os campos da mecânica e da eletrônica [20]. O termo foi originado no Japão por volta de 1960 e espalhou-se pelo mundo todo.

A mecatrônica pode ser dividida principalmente nas seguintes subcategorias:

- Mecânica: área que estuda os movimentos. Através da mecânica são desenvolvidos os sistemas baseados em mecânica clássica, mecânica relativista e mecânica quântica;
- Eletrônica: área que estuda o uso de circuitos eletroeletrônicos para desenvolver sistemas de armazenamento, transmissão e processamento de informações analógicas e digitais;
- Computação: área que busca soluções baseadas em algoritmos matemáticos implementados em sistemas de controle de entradas e saídas de dados.

De modo geral, sistemas e equipamentos mecatrônicos contemplam a junção de peças mecânicas controladas por sistemas eletrônicos. Esses sistemas possuem em sua grande maioria sensores para comunicação dos agentes externos que exercem algum tipo de influência em seu funcionamento, motores para produzir movimentação quando necessário conforme a programação e rotina do sistema, equipamentos de entrada de dados e ações para permitir a interferência humana quando necessário e interface com painéis de controle, *displays* e/ou computadores para transmitir dados de funcionamento.

O projeto de uma garra robótica com FF se enquadra dentro de um sistema mecatrônico pois possui:

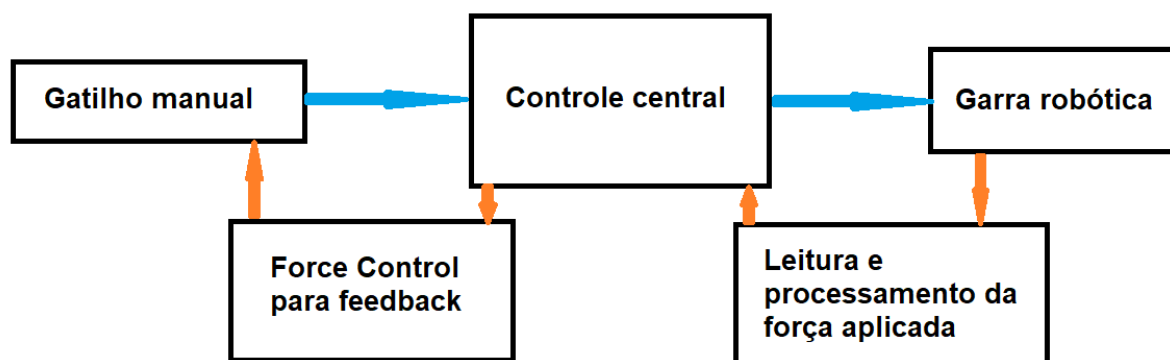
- Sistemas mecânicos: estrutura desenvolvida para a garra, desde o material idealizado para sua confecção até os motores [21] e transdutores utilizados em sua modelagem. Gatilho manual com o sistema de controle de tração através de uma haste e um servo motor;
- Sistemas eletrônicos: comunicação do transdutor de carga por um módulo ADC de 24 bits conectado ao microcontrolador. Controle do sinal do gatilho através do transdutor resistivo;
- Sistemas de computação: programação do Arduino UNO para controlar todo o sistema eletromecânico e utilização da comunicação serial do MC com a IDE no computador para auxiliar com os estudos de comportamento do protótipo.

### 3 – DESENVOLVIMENTO

O desenvolvimento deste trabalho foi dividido em etapas de validação, ou seja, todo o percurso a ser realizado contou com passos de pesquisa, documentação, construção e checagem de funcionamento de maneira progressiva, onde só após a validação final de um bloco de desenvolvimento, foi dada continuidade com a próxima etapa. Cada etapa foca em fazer funcionar um elemento importante do projeto.

O funcionamento do sistema desenvolvido pode ser representado e explicado através do seguinte diagrama de blocos (Figura 3).

FIGURA 3 - DIAGRAMA DE BLOCOS REPRESENTANDO O FUNCIONAMENTO DO SISTEMA COMPLETO



A entrada do sistema é um gatilho atuado manualmente por uma pessoa. A movimentação desse gatilho é inserida em um sistema de controle composto por um microcontrolador que, por sua vez, converte essa leitura em um sinal responsável pela movimentação da garra robótica.

Conforme a garra aperta um objeto, um transdutor de carga captura e envia a massa em kg aplicada sobre a estrutura da garra para um circuito conversor que se comunica novamente com o microcontrolador. Desta vez, o microcontrolador processa esse sinal e o converte em uma saída proporcional responsável por acionar o sistema de *Force Control* que simula a sensação de força no sistema do gatilho.

A metodologia empregada para o desenvolvimento desse projeto pode ser quebrada principalmente nas seguintes etapas:

- Preparação da *protoboard* com a fonte de alimentação adequada ao projeto;

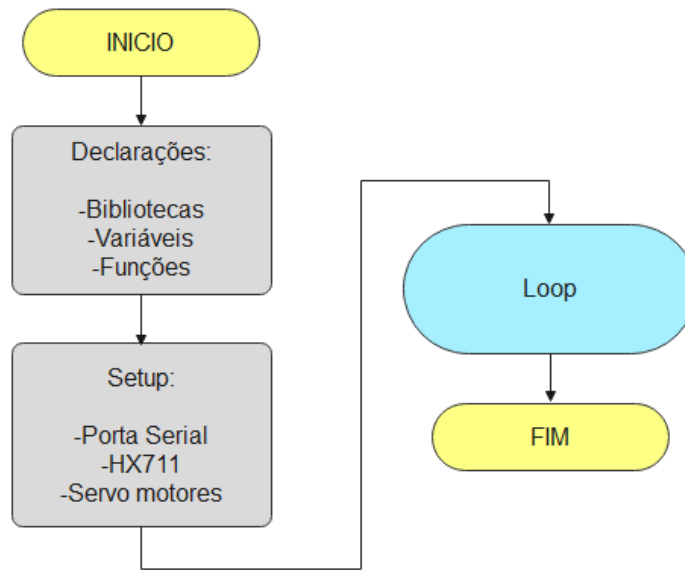
- Projeto e integração do potenciômetro do gatilho com o servo motor da garra robótica através do Arduino UNO;
- Calibração da célula de carga utilizada no sistema e desenvolvimento do código que supre a aplicação;
- Desenvolvimento do sistema de controle do transdutor de carga com o módulo HX711;
- Desenvolvimento do sistema responsável pela simulação da força com um servo motor;
- Testes isolados do bloco de controle da garra e do bloco de medição da força;
- Acoplamento e testes de todos os módulos do projeto ao mesmo tempo;
- Refinamentos do código aplicado para obter respostas mais rápidas e precisas;
- Resolução de conflitos de funcionamento entre os blocos do sistema;
- Levantamento de soluções e alternativas para otimizar e escalar o projeto a dimensões de uso real.

### **3.1 – Sistema de controle**

Para iniciar com o desenvolvimento do sistema de controle, é possível representar o código e lógica aplicada ao microcontrolador da placa Arduino Uno através de fluxogramas simplificados. A Figura 4 mostra o fluxograma geral do código do sistema.



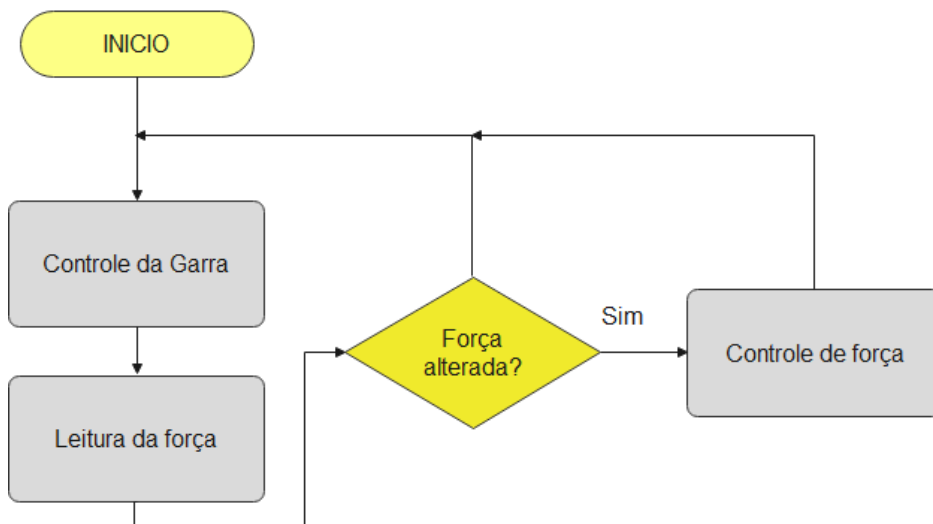
**FIGURA 4 – FLUXOGRAMA DO CÓDIGO DO SISTEMA**



Inicialmente são realizadas as configurações das bibliotecas, variáveis e funções empregadas no código, para realizar o setup de inicialização das portas e periféricos que se comunicam com o Arduino. Após esse setup, o sistema entra em seu regime de funcionamento infinito até que seja desligado ou reiniciado.

Esse *loop* infinito de funcionamento pode ser representado através do seguinte fluxograma (Figura 5).

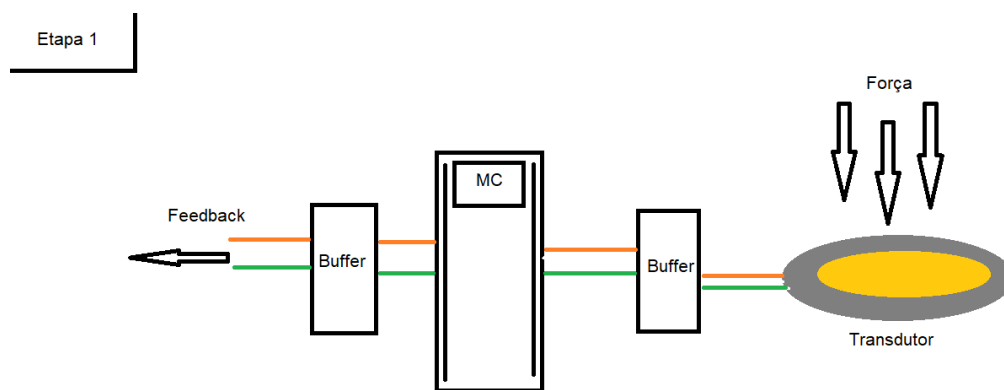
**FIGURA 5 - FLUXO GERAL DO CÓDIGO DO SISTEMA**



Inicialmente é realizada a função de controle da garra de acordo com a leitura do sistema do gatilho. Em seguida é feita a leitura da força aplicada pela garra, ou mais precisamente, a massa em kg aplicada sobre sua estrutura. Se houve alteração dessa massa aplicada, o sistema identifica que é necessário atuar o mecanismo de controle de força acoplado ao gatilho. Se não houve alteração, nenhuma variação é aplicada no sistema de controle de força.

A primeira etapa do projeto consistiu na estruturação e codificação do sistema central utilizando o microcontrolador e os transdutores do sistema. A Figura 6 ilustra o diagrama do sistema de controle do protótipo.

**FIGURA 6 - DIAGRAMA DO SISTEMA DE CONTROLE CENTRAL DO PROTÓTIPO INTEGRANDO O MICROCONTROLADOR AO TRANSDUTOR DE CARGA**



Na imagem acima, “Buffer” se refere apenas a uma parte do sistema responsável pela interface entre os componentes externos e o microcontrolador, adaptando tanto as conexões necessárias como a maneira que a informação deve ser convertida e enviada ao Arduino.

As primeiras simulações foram realizadas através do software online TinkerCad, um programa de modelagem 3D gratuito da Autodesk e que também conta com uma ferramenta de simulação de circuitos eletroeletrônicos e integração com a placa Arduino UNO R3.

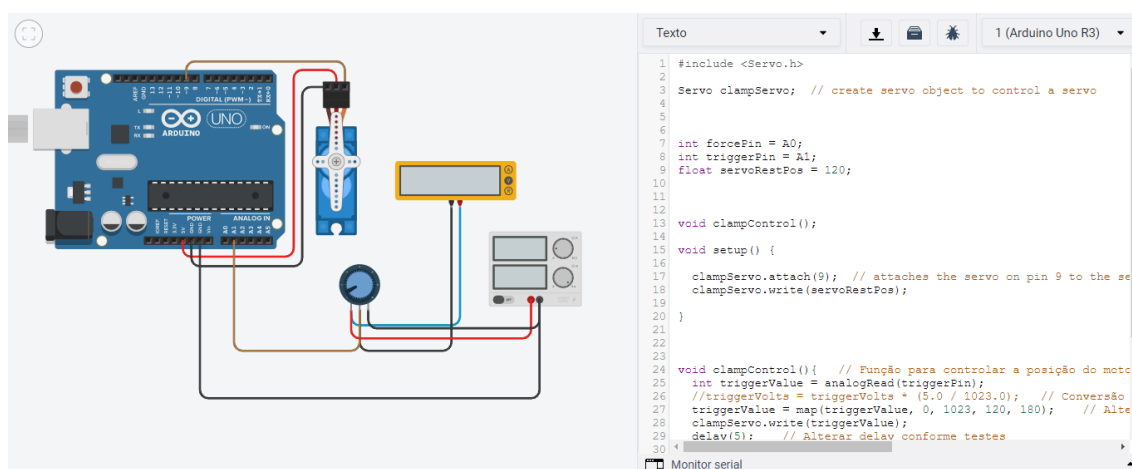
Apesar de auxiliar em testes simples iniciais em um projeto, o TinkerCad se limita a utilização dos componentes existentes em seu banco de dados, e das bibliotecas de programação nativas suportadas pela família Arduino. Para projetos

mais complexos, deve-se procurar alguma outra ferramenta de simulação [22], ou desenvolver diretamente com a IDE do Arduino e componentes reais.

A programação do Arduino conta com o auxílio do uso de bibliotecas, assim como a maioria das plataformas de programação. Bibliotecas podem ser facilmente adicionadas ao código e permitem a utilização de funções previamente configuradas para trabalhar com determinadas peças de hardware conectadas ao sistema.

O Tinkercad permite a digitação do código para o Arduino e executa a simulação e animação do sistema conforme programado. Na Figura 7 está apresentada a primeira simulação de teste para este projeto, composta apenas pela placa UNO R3, o servo motor conectado ao pino 9 e um potenciômetro representando o detector de movimentação do gatilho manual. Ao lado direito está um exemplo do código digitado na plataforma para realizar a simulação.

**FIGURA 7 - PRIMEIRO TESTE DE CONTROLE DO SERVO MOTOR COM UM POTENCIÔMETRO REALIZADO NO TINKERCAD**



O controle do sistema é dado por meio de um gatilho que altera a resistência de um potenciômetro responsável por transmitir ao Arduino o quanto o gatilho foi pressionado. Essa resposta determina também o posicionamento da garra robótica. Ou seja, se uma pessoa pressionar o gatilho até 30% de seu movimento máximo, a garra deve ser fechada em 30% a partir da sua posição completamente aberta. De maneira similar, se o gatilho está pressionado em 30% e a pessoa solta-o para 13%,

a garra deve recuar até estar posicionada em apenas 13% a partir da sua posição completamente aberta.

Tal tipo de controle de um potenciômetro a partir de um gatilho pode ser encontrado em aceleradores de autorama [23]. A principal diferença é que o acelerador fecha o circuito pelo qual passa a corrente elétrica que induz a movimentação do carrinho na pista de autorama, enquanto o controlador deste protótipo passa apenas a tensão presente na seção do potenciômetro que forma o divisor de tensão de entrada.

### **3.2 – Sistema de microservo motor**

Conforme apresentado ao final da seção anterior, a utilização de bibliotecas no Arduino facilita o trabalho com componentes de hardware. Para a comunicação com os sistemas de motores do projeto, foi utilizada a biblioteca <servo.h>. Essa biblioteca permite o uso de funções para controle preciso de posição de servo motores através da sua atribuição a algum pino de saída. Para permitir a utilização dessa biblioteca no projeto, o escopo foi desenvolvido pensando na utilização de um microservo motor com a função de abrir ou fechar a garra robótica.

Microservo motores projetados para trabalhar com as placas Arduino podem ser alimentados através do próprio pino de 5V, desde que estejam sem carga em seus eixos para que não seja exigida muita corrente elétrica. Os pinos do Arduino fornecem no máximo 40 mA de corrente elétrica. Como precaução, é preferível utilizar fontes de alimentação externa para evitar o mau funcionamento do sistema, ou na pior das hipóteses, danificação permanente da placa do microcontrolador.

O servo motor utilizado no protótipo foi o Micro servo 9g SG90 TowerPro, cujas especificações são:

- Tensão de Operação: 3,0 – 7,2V
- Ângulo de rotação: 180 graus
- Velocidade: 0,12 seg/60Graus (4,8V) sem carga
- Torque: 1,2 kg.cm (4,8V) ~ 1,6 kg.cm (6,0V)
- Temperatura de Operação: -30°C ~ +60°C

- Tipo de Engrenagem: Nylon
- Tamanho do cabo: 245mm
- Dimensões: 32 x 30 x 12mm
- Peso: 9g

Conforme é possível observar nas especificações, esse motor possui baixo torque. Com a limitação de 1,2 kg.cm não foi possível montar a estrutura da garra robótica utilizando esse motor para movimentar o sistema. Portanto, o protótipo consistiu em uma representação do comportamento de um equipamento dimensionado para trabalhar com maiores valores de carga e torque, onde a força exercida sobre o transdutor foi aplicada de forma manual.

Conforme o escopo do protótipo, o motor 9g SG90 deve ser controlado por um mecanismo atuador manual, podendo este se assemelhar a um gatilho, alavanca, botão analógico etc. Esse atuador é responsável por transmitir a movimentação do dedo da pessoa para a garra robótica, onde o MC se encarrega de converter a distância pressionada para a angulação da abertura da garra. O valor final utilizado para o campo de movimentação do motor foi de 90° a 180°, e devido à ausência de qualquer sistema de caixa de redução, consequentemente essa é a amplitude de movimentação da própria garra, permitindo a utilização de pequenos objetos que se encaixem dentro do espaço interno.

Para controlar o micro servo inicialmente foi desenvolvida uma função utilizando os comandos da biblioteca <servo.h>, conforme o código representado na Figura 8.

**FIGURA 8 - FUNÇÃO UTILIZADA PARA O CONTROLE DO MOTOR DA GARRA ROBÓTICA**

```
void clampControl(){ // Função para controlar a posição do motor da garra de acordo com
    int triggerValue = analogRead(triggerPin); // a tensão gerada pelo potenciômetro do atuador
    triggerValue = map(triggerValue, 0, 1023, 90, 180); // Alterar ângulo conforme testes
    clampServo.write(triggerValue);
    delay(2); // Alterar delay conforme testes
}
```

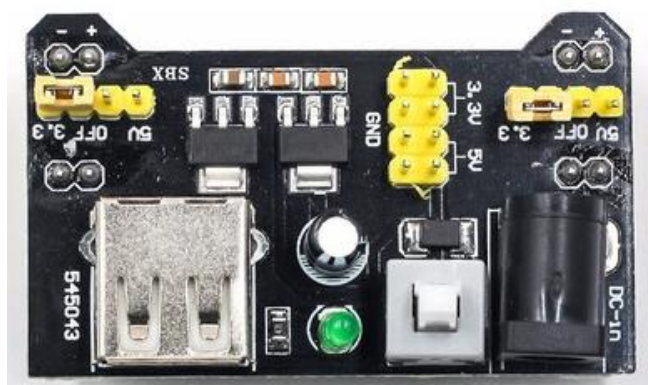
O comando *triggerValue = map(triggerValue, 0, 1023, 90, 180)* consiste em converter o valor obtido do potenciômetro atuador para um ângulo entre 90° e 180°. O potenciômetro fornece uma tensão de entrada ao pino analógico triggerPin, entretanto esse valor é convertido em uma escala de 1023 valores, resultando em uma resolução de 4,88mV por divisão. Para encontrar o valor correspondente em tensão, basta multiplicar o número obtido pelo MC por 4,8mV.

Para atribuir a posição correspondente ao micro servo utilizado na garra, é necessário mapear novamente esse resultado para valores entre 90 e 180. Dessa forma, uma unidade na escala de 1023 corresponde a um ângulo de 0,08°, mantendo-se dentro dos limites: 0 de entrada fornece 90 de saída, e 1023 de entrada fornece 180 de saída. Devido à utilização da biblioteca para o servo motor, esta já se encarrega de processar o valor de saída e fornecer o sinal necessário para deslocar ou manter o motor na posição correta.

### 3.3 – Fonte de alimentação do sistema

Para a alimentação dos componentes externos do protótipo foi utilizado um módulo de fonte ajustável com dois canais de saída para acoplar em *protoboards* de 400 ou 830 furos, conforme a Figura 9.

FIGURA 9 - FONTE DE ALIMENTAÇÃO PARA PROTOBOARD UTILIZADA NO PROTÓTIPO



- Tensão de Entrada: 6.5-12 V (DC) ou fonte de alimentação USB;
- Tensão de saída: 3,3V / 5V, canal duplo;

- Corrente máxima de saída: 700mA;
- Dimensões: 5,3cm x 3,5cm.

A utilização desse módulo de alimentação efetivamente expandiu a limitação de corrente máxima de 40mA fornecido pelo Arduino para 700mA, valor suficiente para a montagem íntegra do projeto. O módulo foi alimentado por uma fonte de 12V e 2A, bivolt, 24W, suficiente para atender com uma boa margem de folga o projeto. A Figura 10 mostra a fonte de alimentação utilizada.

**FIGURA 10 - FONTE DE ALIMENTAÇÃO UTILIZADA PARA ALIMENTAR A FONTE DA *PROTOBOARD***



Quando se utiliza diversas fontes de alimentação, é importante interconectar o terminal comum (GND) de todas para evitar diferentes potenciais de referência, o que pode provocar mau funcionamento ou até mesmo danificação de algum componente eletrônico. Sendo assim, foram conectados externamente os GND da fonte de alimentação da *protoboard* para garantir o nivelamento da referência apenas como medida extra de precaução, pois já se trata de um módulo feito para trabalhar com dois canais em um mesmo projeto.

Por fim, os GND foram conectados ao pino GND do Arduino para garantir que ambos trabalhem com o mesmo potencial de referência. Esta sim é uma medida necessária em 100% dos casos para evitar principalmente o mau funcionamento da placa Arduino. Para alimentar a placa do Arduino foi utilizada o tempo todo a conexão USB entre ela e um *laptop*.

### 3.4 – Sistema detector de força com o transdutor de carga

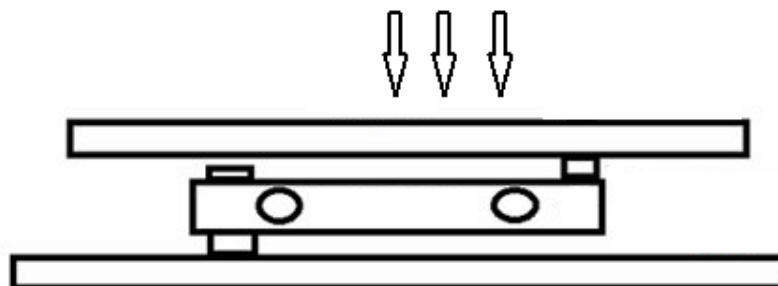
Para a montagem do sistema detector da força aplicada pela garra, foi utilizado um transdutor capaz de trabalhar com até 5kg [24], conforme especificações a seguir:

**Célula de carga 5 kg (especificações):** transdutor para captar a força aplicada pela garra.

- Carga nominal: 0 a 5 kg
- Sensibilidade de saída:  $1.0 \text{ mV/V} \pm 0.1 \text{ mV/V}$
- Tensão de funcionamento recomendada: 3 ~ 12 VDC
- Tensão máxima de operação: 15 VDC
- Impedância de entrada:  $1000 \pm 50 \text{ Ohm}$
- Impedância de saída:  $1000 \pm 50 \text{ Ohm}$
- Faixa de compensação de temperatura:  $-10^{\circ}\text{C} \sim 40^{\circ}\text{C}$
- Material: liga de alumínio
- Peso: 31 gramas

O sistema da célula de carga projetado neste trabalho consistiu em uma pequena balança de madeira [25] confeccionada manualmente conforme a modelagem da Figura 11.

FIGURA 11 - ESQUEMA DE MONTAGEM DA BALANÇA UTILIZANDO O TRANSDUTOR DE CARGA DE 5 QUILOS

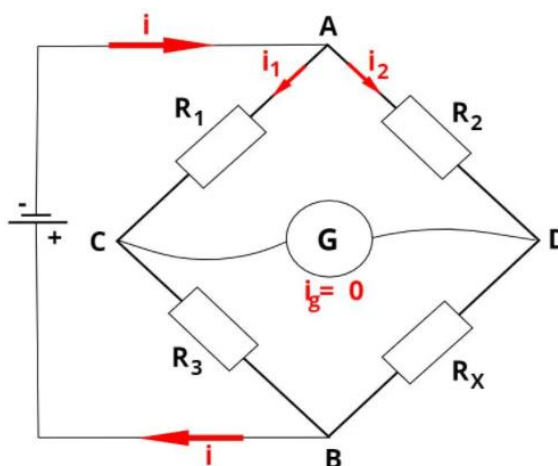




Nessa configuração, qualquer peso exercido sobre a placa superior provoca uma pequena deformação no centro da célula de carga que está fixada pelas extremidades em ambas as placas. Essa deformação em pequena escala é suficiente para alterar a resistência interna do componente. Essa alteração é processada pelo módulo HX711 [26].

Internamente, uma célula de carga pode ser representada por uma ponte de Wheatstone da seguinte forma (Figura 12).

FIGURA 12 - ESQUEMA ELÉTRICO DE UMA PONTE DE WHEATSTONE [27]



Uma Ponte de Wheatstone é um circuito elétrico inventado por Samuel Hunter Christie e difundido por Charles Wheatstone [28]. Esse esquema consiste em quatro resistores e um galvanômetro, sendo dois resistores dimensionados com valores conhecidos, um resistor variável e uma resistência desconhecida.

Caso a ponte esteja em equilíbrio, ocorre igualdade entre o produto das resistências cruzadas, ou seja, conforme a Eq. 1:

$$R_2 * R_3 = R_1 * R_x \quad (1)$$

Essa igualdade pode ser demonstrada através da 2ª Lei de Kirchhoff que determina que a soma das tensões elétricas em cada elemento de uma malha em um circuito é sempre nula:

$$\sum U_{ADC} = 0 \rightarrow -i_2 R_2 + 0 + i_1 R_1 = 0 \rightarrow \frac{i_1}{i_2} = \frac{R_2}{R_1}$$

$$\sum U_{DBC} = 0 \rightarrow -i_2 R_x + i_1 R_3 + 0 = 0 \rightarrow \frac{i_1}{i_2} = \frac{R_x}{R_3}$$

$$\therefore \frac{R_2}{R_1} = \frac{R_x}{R_3} \rightarrow R_2 \cdot R_3 = R_1 \cdot R_x$$

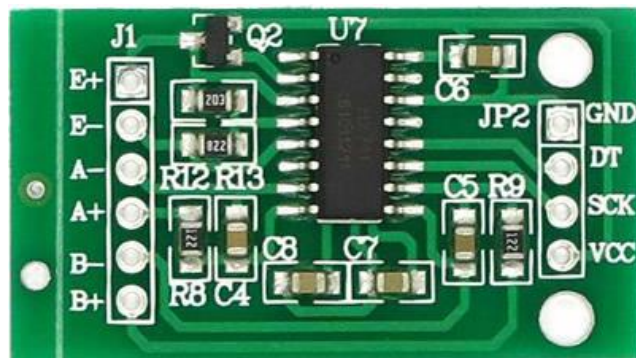
No caso da célula de carga, os nós C e D são usados como terminais de saída. Para o controle do sensor de carga de 5 kg deste projeto, foi utilizado o módulo conversor ADC HX711 [29].

**Módulo HX711 (especificações):** interface entre o MC e os transdutores de carga.

- Tensão de operação: 4,8 à 5,5V DC
- Corrente de operação: 1,6mA
- Temperatura de operação: -20 à 85°C
- Interface SPI
- Dimensões: 29 x 17 x 4mm (sem os pinos)

O módulo HX711 (Figura 13) cumpre com a função de interface entre células de carga e placas Arduino. Esse pequeno circuito funciona como um conversor Analógico/Digital (ADC) de 24 bits que recebe e processa a variação de resistência produzida pelos transdutores de carga e envia um sinal digital ao microcontrolador.

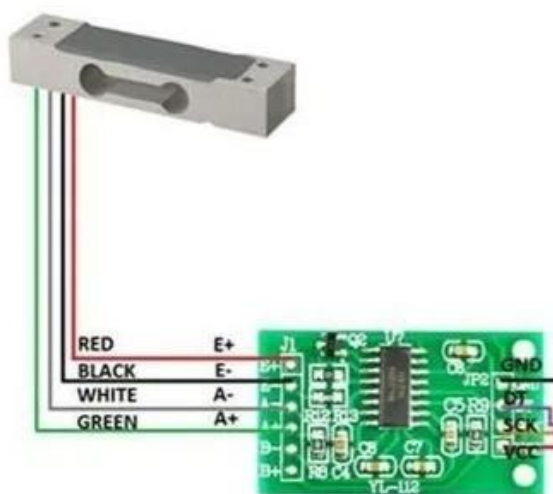
**FIGURA 13 - MÓDULO CONVERSOR ADC 24 BITS HX711**



Os pinos VCC e GND são conectados a uma fonte de alimentação. O pino DT é o responsável pela transmissão dos dados de maneira serial do conversor para o

MC. O pino SCK mantém o controle de *clock* do módulo através do MC. Do outro lado, as entradas E+, E-, A+, A-, B+, B- são conectadas nas células de carga conforme o padrão de fios utilizado nos transdutores [30]. Conforme apresentado anteriormente, neste projeto foi usada uma célula de carga de 5 kg apenas, portanto sua conexão no HX711 ocorreu conforme o diagrama da Figura 14.

**FIGURA 14 - ESQUEMA DE CONEXÃO DA CÉLULA DE CARGA COM O HX711 [31]**



Os terminais E+ e E- do HX711 fornecem a tensão de entrada no transdutor, e os terminais A+ e A- recebem a diferença de tensão produzida pela variação da resistência interna do sensor. Sem nenhuma carga aplicada em estado de repouso, a resistência interna é balanceada e funciona como um divisor de tensão de razão  $\frac{1}{2}$ .

Com essa configuração é utilizado o canal A do HX711. O Canal A pode ser programado para operar com um ganho de 128 ou 64, permitindo trabalhar com as resoluções de tensão de  $\pm 20\text{mV}$  ou  $\pm 40\text{mV}$  respectivamente, utilizando-se uma fonte de alimentação de 5V. O Canal B possui ganho fixo de 32. A escolha do canal A se deve a dois motivos: resolução mais precisa para trabalhar com o sensor de carga, e a biblioteca utilizada para comunicar o Arduino com o HX711 possuir fácil implementação do ganho de 128 com o sistema.

A biblioteca utilizada é a de Bogde [32] para realizar a interface entre placas Arduino e o módulo HX711. Essa biblioteca possui funcionamento 100% compatível com o Arduino UNO. Através dela é possível processar a informação da deformação

da célula de carga em um instante de tempo ou uma média de vários instantes consecutivos, apesar dessa funcionalidade acrescentar lentidão a execução do código proporcional a quantidade de instantes utilizados na média.

Essa biblioteca também conta com funções para trabalhar de maneira não bloqueadora no Arduino (prosseguir com a execução do código caso ocorra algum mau funcionamento na leitura), funções para escrever as unidades medidas de maneira absoluta ou utilizando alguma unidade de medida, e conta com uma função para tarar o transdutor e eliminar a interferência dos materiais que constituem o equipamento (neste caso, a placa de madeira apoiada na célula de carga).

Cada célula de carga sai de fábrica com alguma variação em sua composição interna, e conseqüentemente no equilíbrio da ponte de Wheatstone. Para calibrar essa peça foram utilizadas funções da biblioteca em conjunto com um corpo de prova de peso conhecido. Esse corpo foi o *Smartphone* Moto G7 plus, com 176g. A partir da calibração com uma carga conhecida, foi possível determinar o fator de correção da célula de 5kg e implementar essa constante nos cálculos do programa [33], atingindo uma precisão de medida de 0,001kg.

O código desenvolvido para essa calibração utiliza algumas funções disponíveis na biblioteca do HX711. Conforme a Figura 15, a primeira configuração da célula de carga é feita com a chamada da função `loadCell.set_scale()` sem nenhum parâmetro. Ao imprimir o valor no monitor serial da IDE com a média das últimas 5 leituras é retornado um valor flutuante e descalibrado devido à natureza do transdutor.

Ao colocar a carga conhecida de 176g esse valor retornado se estabiliza. Para obter o fator de calibragem do sistema, deve-se dividir essa constante pela massa utilizada em kg e passar esse novo parâmetro na função `loadCell.set_scale(cellFactor)`. O valor obtido foi de aproximadamente 74000.

**FIGURA 15 - CÓDIGO UTILIZADO PARA A CALIBRAÇÃO DA CÉLULA DE CARGA UTILIZADA NO PROJETO**

```
//Load cell calibration code for HX711

#include "HX711.h"

HX711 loadCell;

const int LOADCELL_DOUT_PIN = 2;
const int LOADCELL_SCK_PIN = 3;
float cellFactor;

void setup() {
    // put your setup code here, to run once:
    Serial.begin(9600);
    while (!Serial) {
        ; // wait for serial port to connect. Needed for native USB port only
    }

    loadCell.begin(LOADCELL_DOUT_PIN, LOADCELL_SCK_PIN);
    loadCell.set_scale();
    loadCell.tare();
    delay(500);
}

void loop() {
    // put your main code here, to run repeatedly:
    Serial.println(loadCell.get_units(5));
}
```

Após a calibração, o código foi modificado com a seguinte parametrização, deixando dessa maneira o sensor de carga devidamente ajustado via software (Figura 16).

**FIGURA 16 - MODIFICAÇÃO DO CÓDIGO COM O FATOR DE CALIBRAGEM DO TRANSDUTOR DE CARGA**

```
cellFactor= (74000/0.176);
loadCell.begin(LOADCELL_DOUT_PIN, LOADCELL_SCK_PIN);
loadCell.set_scale(cellFactor);
loadCell.tare();
delay(500);
```

Com o fator de correção aplicado na função `loadCell.set_scale`, a balança foi capaz de registrar valores com precisão de 0,001kg conforme dito anteriormente. Para utilizar essa configuração no código principal do projeto, basta replicar os comandos onde a constante de correção já é fornecida como parâmetro. A principal diferença entre o código utilizado para a calibração e o código do projeto é, além do nome da

variável, a quantidade de medidas utilizadas para capturar e imprimir no monitor serial a força aplicada (Figura 17).

FIGURA 17 - FUNÇÃO QUE CAPTURA E EXIBE O VALOR EM KG APLICADO AO TRANSDUTOR

```
void cellReading(){  
    if (clampCell.is_ready()) {  
        Serial.print("HX711 reading: ");  
        Serial.println(clampCell.get_units());  
    } else {  
    }  
}
```

Ao fornecer qualquer valor para o comando *clampCell.get\_units*, o valor retornado é uma média entre a quantidade de medições informadas no parâmetro. Quanto maior o número informado, mais preciso será o valor retornado pela média, entretanto há um compromisso entre precisão e velocidade. Infelizmente, qualquer tentativa de utilizar algum valor acima de uma amostra por medida resultou em inserção de atraso de processamento no sistema e comprometeu a resposta da garra, picotando as leituras do potenciômetro e a movimentação do motor. Portanto, manteve-se a chamada do comando *clampCell.get\_units()* sem informar qualquer valor, o que seria o equivalente a chama-lo como *clampCell.get\_units(1)*.

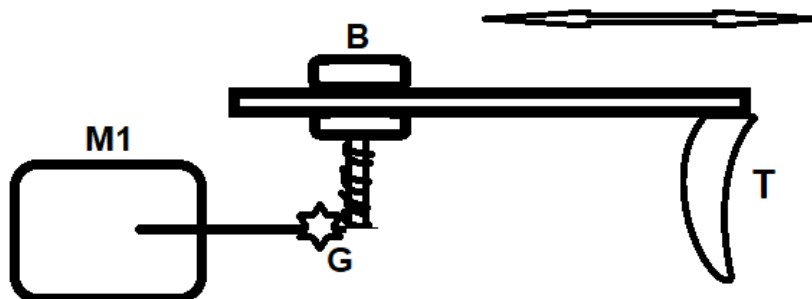
### 3.5 – Sistema de force feedback aplicado ao gatilho

Para simular o sistema de *force feedback* no gatilho acionado pelos dedos, foi utilizado outro servo motor 9g SG90 TowerPro. A ideia é que este motor seja o responsável por controlar de maneira refinada o aperto de uma braçadeira que determina o nível do atrito aplicado ao eixo do gatilho, desta forma possibilitando a simulação da força exercida pela garra do outro lado do sistema.

O posicionamento do servo motor é diretamente proporcional a força aplicada no transdutor de 5kg. Zero de força aplicada mantém o motor na posição configurada como mínima. Essa posição representa o desaperto total da braçadeira.

O protótipo desenvolvido utiliza apenas essa conversão direta da força aplicada na garra ao ângulo do eixo do motor para ilustrar e validar a possibilidade de controlar o *force feedback* através dessa relação intermediada pelo microcontrolador. A ideia de efetivamente utilizar esse motor para apertar uma braçadeira que provoque resistência à movimentação do gatilho não foi desenvolvida em primeiro momento, e será expandida dentro dos estudos de perspectivas de continuação. A Figura 18 mostra a imagem do sistema idealizado nesse trabalho.

FIGURA 18 - SISTEMA IDEALIZADO PARA REALIZAR O *FORCE FEEDBACK* NO GATILHO



O motor M1 é controlado pela força exercida na garra. Sua movimentação controla a engrenagem G responsável por apertar ou afrouxar a braçadeira B. Essa braçadeira simula a força da garra, exercendo maior ou menor atrito na haste conectada ao gatilho T. Quanto mais apertada a braçadeira, mais esmagada fica a haste que passa por ela, tornando mais difícil o aperto do gatilho T.

Sob tais condições, esse mecanismo funciona apenas para quando o gatilho for apertado. Para que a braçadeira seja afrouxada e permita o retorno do gatilho a posições mais abertas, é necessário discutir soluções e mecanismos para a identificação desse movimento contrário. Tais discussões serão apresentadas nas seções de perspectivas de continuação.

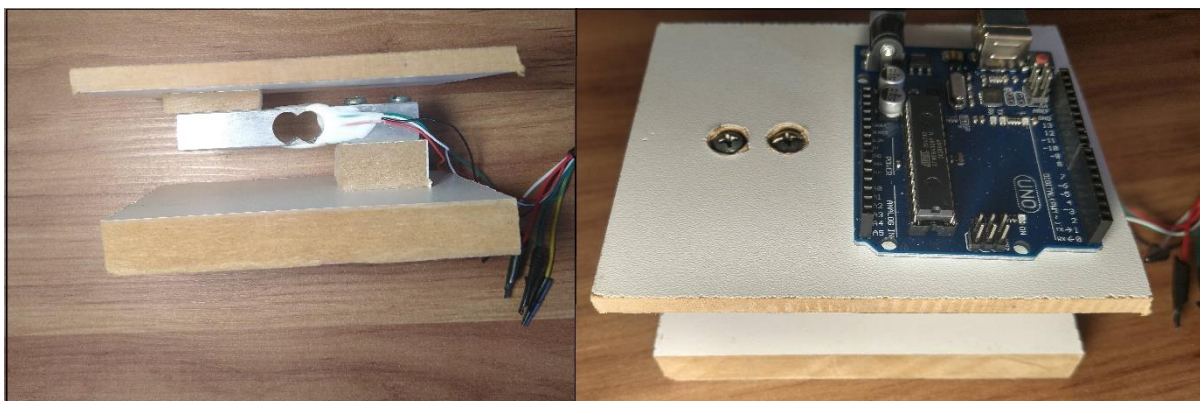
## 4 – RESULTADOS E DISCUSSÃO

Nesta seção são apresentados e discutidos os resultados obtidos ao longo do desenvolvimento do protótipo através de explicações acompanhadas de figuras com imagens do protótipo real desenvolvido.

Também são discutidos os obstáculos e soluções encontradas para resolver problemas de compatibilidade entre módulos do sistema e problemas para permitir o funcionamento mecânico do sistema idealizado.

Conforme apresentado anteriormente, para a utilização do transdutor de carga de 5kg, foi confeccionada uma estrutura em madeira para representar uma balança a fim de coletar a força exercida sobre essa parte do sistema. A Figura 19 mostra essa estrutura.

**FIGURA 19 - BALANÇA CONFECCIONADA COM MADEIRA E A CÉLULA DE CARGA DE 5KG. ARDUINO UNO UTILIZADO COMO REFERÊNCIA DE ESCALA**



Conforme é observado na Figura 19, os parafusos de fixação do transdutor na placa de madeira foram inseridos em cavidades para manter a superfície devidamente nivelada.

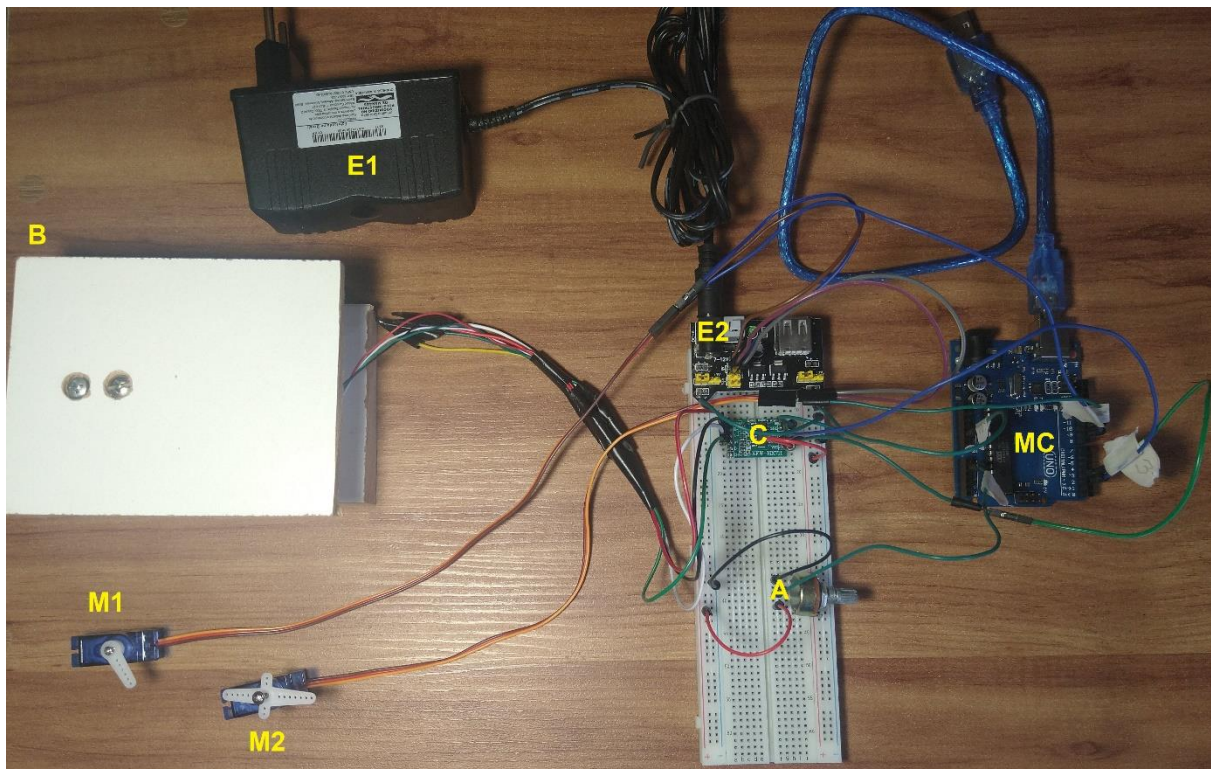
Para conexão da balança na *protoboard* do projeto, os fios que vieram embutidos na célula de carga foram emendados por pressão através de conectores macho-fêmea e fixados com fita isolante para manter sua posição e estabilidade da emenda.

Na Figura 20 é apresentado todo o protótipo montado. Cada elemento do sistema está rotulado com uma letra para a legenda e explicação do funcionamento obtido.



No link ([https://youtu.be/YAGO\\_jStyMc](https://youtu.be/YAGO_jStyMc)) encontra-se um vídeo com a demonstração do funcionamento do protótipo. Já no link (<https://youtu.be/O5TvcR1HAT4>) encontra-se um vídeo com a demonstração do funcionamento do protótipo com uma breve explicação do que está sendo executado.

FIGURA 20 - PROTÓTIPO TOTALMENTE MONTADO



Legenda:

- A – Potenciômetro 10k $\Omega$  (representação do gatilho)
- B – Estrutura com o transdutor de carga de 5kg
- C – Módulo conversor HX711
- E1 – Fonte de alimentação 12V 2<sup>a</sup>
- E2 – Fonte de alimentação para protoboard 5V 700mA
- MC – Microcontrolador Arduino Uno
- M1 – Motor da garra robótica (representação da movimentação)
- M2 – Motor do *Force Feedback* (representação do *feedback*)

A fonte de 12V E1 fornece tensão para a fonte da *protoboard* E2. Essa fonte E2, conforme já apresentado na Seção 3.3 energiza a *protoboard* com 5V em ambas

as trilhas de energia, e possui conexão entre seus GND. O MC é o Arduino responsável pelo controle de todo o sistema, está conectado a todas as partes necessárias e é alimentado pelo *laptop* através do cabo USB AB.

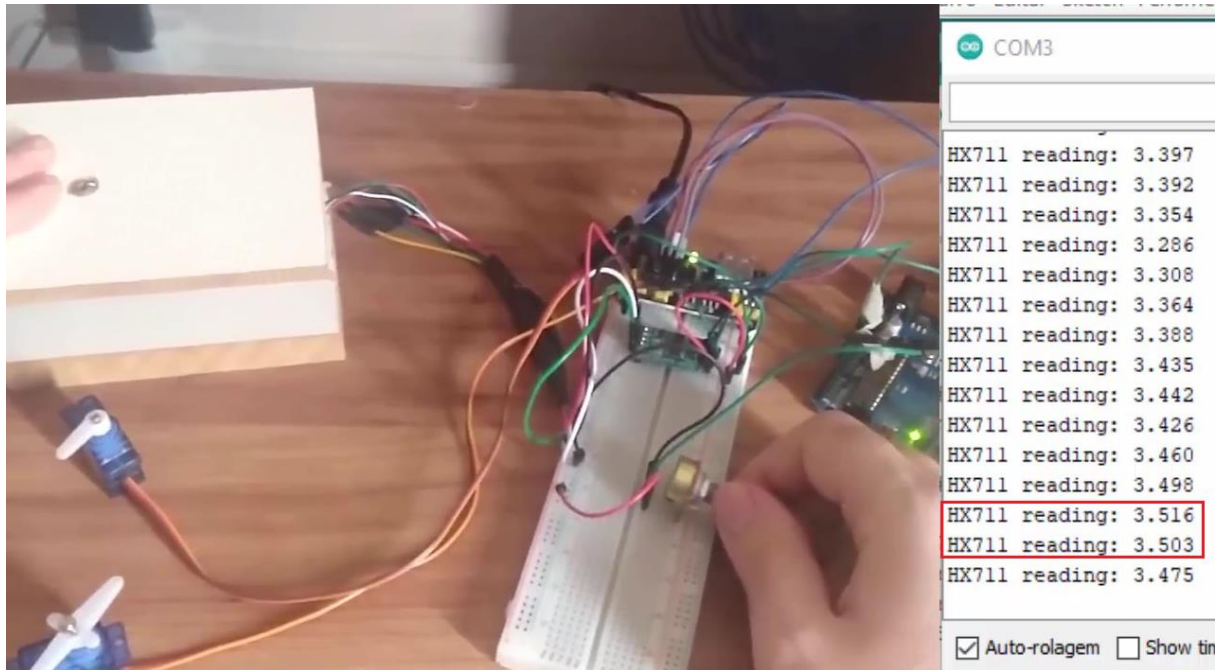
O potenciômetro A de 10k $\Omega$  representa o gatilho atuado pelo dedo e controla o movimento do micro servo M1, com conversão proporcional de 0-5V posicionando o motor entre 90°-180°. O motor M2 é responsável por simular o aperto da braçadeira que em um sistema completo esmagaria ou afrouxaria a haste do gatilho, dificultando ou facilitando o movimento do dedo.

Por fim, a balança B recebe a força aplicada sobre sua superfície e está conectada ao módulo HX711 acoplado na *protoboard* através dos pinos soldados em sua placa [34], representado pela letra C. O módulo realiza a conversão e comunicação da informação com o MC. O MC atualiza o valor da força no monitor serial da IDE do Arduino e controla o posicionamento de M2 de acordo com o valor obtido.

Os resultados coletados com o funcionamento do protótipo não seguiram um procedimento padrão de caracterização do sistema, mas sim aplicações aleatórias de forças sobre a célula de carga manualmente e utilizando objetos variados. Já em relação a movimentação da garra, foram realizados controles figurativos da movimentação do motor. Portanto, não foram gerados gráficos ou tabelas relacionando possíveis variações de grandezas com as respostas do sistema, ou a realização de uma modelagem de um sistema de controle seguindo uma metodologia de equacionamento dos diferentes blocos desse controle que foi apresentado na Figura 03.

A massa máxima aplicada sobre o sistema foi de 3,5kg, conforme Figura 21. A limitação desse valor será discutida na seção a seguir.

FIGURA 21 - VALOR MÁXIMO DE 3,5KG APLICADO SOBRE O PROTÓTIPO



#### 4.1 – Conflito entre <Servo.h> e “HX711.h”

Os primeiros testes de cada parte do protótipo foram realizados de maneira individual e isolada, portanto cada componente funcionou conforme o esperado. O controle dos motores através da programação utilizando a biblioteca <servo.h> resultou em precisão e tempo de resposta satisfatórios. Similarmente foram obtidos bons resultados com o conversor HX711 utilizando a biblioteca “HX711.h”, onde a precisão e o tempo de resposta para a coleta apenas do valor instantâneo foram suficientes para desenvolver o protótipo.

Quando foram iniciados os testes do sistema juntando ambas as partes, motores e transdutor de carga, ocorreu mau funcionamento do projeto. Os motores deveriam se manter estáveis nas angulações desejadas conforme programado, porém em instantes aleatórios apresentavam pequenos espasmos de curtíssima duração (aproximadamente 200ms) e com amplitude de aproximadamente  $10^\circ$  [35].

Foram realizadas medições com o multímetro para verificar a possibilidade de instabilidades de alimentação, flutuações na corrente elétrica possivelmente provocadas pelo conjunto do transdutor e HX711, e/ou más conexões entre os fios e a *proto*board, algo relativamente comum nesses tipos de projetos. Não foi constatado nenhum tipo de anomalia de origem eletroeletrônica, e caso os sistemas fossem

testados individualmente novamente, apresentavam funcionamento normal, garantindo que não ocorreu nenhuma danificação ao hardware.

Após descartada a possibilidade de falhas de hardware, foi verificado o código do projeto. Em nenhuma parte do programa há algum laço ou rotina que resulte no travamento ou alteração no comportamento dos pinos atribuídos aos motores, portanto não foi constatado problema no código desenvolvido. Sendo assim, restou investigar possíveis conflitos originados internamente pelas bibliotecas. Foi constatado que a biblioteca utilizada para o HX711 interferiu no funcionamento da biblioteca dos servos.

A biblioteca <servo.h> utiliza uma rotina de interrupção de aproximadamente 4µs através do Timer1, de 16 bits. O controle de um servo motor por interrupções de *timer* permite o livre funcionamento do resto do sistema.

A biblioteca “HX711.h” foi desenvolvida de maneira colaborativa pela comunidade e gerenciada por Bogdan Necula, com livre direito de acesso, utilização e modificação, desde que acompanhada de sua MIT *license*, portanto o acesso ao código é de fácil acesso. Ao analisar o código fonte HX711.cpp, foi encontrado um tratamento que desativa todas as interrupções do sistema para executar os processos críticos de leitura do valor do HX711. Tal segmento de código encontra-se na Figura 22.

FIGURA 22 - TRECHO DO CÓDIGO DA BIBLIOTECA DO HX711 QUE DESABILITA AS INTERRUPTÕES

```
#if HAS_ATOMIC_BLOCK
    ATOMIC_BLOCK(ATOMIC_RESTORESTATE) {

    #elif IS_FREE_RTOS
        // Begin of critical section.
        // Critical sections are used as a valid protection method
        // against simultaneous access in vanilla FreeRTOS.
        // Disable the scheduler and call portDISABLE_INTERRUPTS. This prevents
        // context switches and servicing of ISRs during a critical section.
        portMUX_TYPE mux = portMUX_INITIALIZER_UNLOCKED;
        portENTER_CRITICAL(&mux);

    #else
        // Disable interrupts.
        noInterrupts();
    #endif
```

Após a conclusão desse comando *If* [36] é executada a seção crítica de leitura do módulo conversor. Ao final da leitura, as interrupções são reativadas, conforme o código mostrado na Figura 23.

FIGURA 23 - TRECHO DO CÓDIGO DA BIBLIOTECA DO HX711 QUE REABILITA AS INTERRUPÇÕES

```
#if IS_FREE_RTOS
// End of critical section.
portEXIT_CRITICAL(&mux);

#elif HAS_ATOMIC_BLOCK
}

#else
// Enable interrupts again.
interrupts();
#endif
```

A explicação para a necessidade desse tratamento foi dada da seguinte forma (Figura 24).

FIGURA 24 - TRECHO DO CÓDIGO DA BIBLIOTECA DO HX711 COM A EXPLICAÇÃO DA NECESSIDADE DE DESABILITAR AS INTERRUPÇÕES

```
// Protect the read sequence from system interrupts. If an interrupt occurs during
// the time the PD_SCK signal is high it will stretch the length of the clock pulse.
// If the total pulse time exceeds 60 uSec this will cause the HX711 to enter
// power down mode during the middle of the read sequence. While the device will
// wake up when PD_SCK goes low again, the reset starts a new conversion cycle which
// forces DOUT high until that cycle is completed.
//
// The result is that all subsequent bits read by shiftIn() will read back as 1,
// corrupting the value returned by read(). The ATOMIC_BLOCK macro disables
// interrupts during the sequence and then restores the interrupt mask to its previous
// state after the sequence completes, insuring that the entire read-and-gain-set
// sequence is not interrupted. The macro has a few minor advantages over bracketing
// the sequence between `noInterrupts()` and `interrupts()` calls.
```

Conforme o comentário inserido dentro do próprio código, a justificativa para desativar as interrupções durante o processo de leitura é que caso o tempo de pulso no pino PD\_SCK do módulo ultrapasse 60µs, o modo de baixo consumo do HX711 será ativado, e ao retornar será provocado um erro de leitura onde todos os bits subsequentes serão capturados com o valor 1.

As únicas interrupções utilizadas no sistema foram as da biblioteca de controle dos motores. Sendo assim, as partes do código fonte que desabilitam e retomam as interrupções do sistema foram removidas, eliminando o comportamento de espasmos aleatórios.

Conforme explicado, o possível problema que pode ocorrer ao retirar essa proteção contra interrupções durante o processo crítico de leitura do HX711 é a corrupção dos dados e consequentemente do funcionamento do sistema. O tempo que o pino SCK pode ficar ativado não deve exceder 60µs. Foi verificado que o projeto apresenta comportamento estável para até 4kg de massa aplicada no transdutor. Acima disso, o tempo limite de leitura é excedido, e devido à falta de proteção contra interrupções, o sistema entra em colapso. Portanto, os testes foram limitados a trabalhar com até 3,5kg, inserindo uma margem de segurança de 0,5kg, até que seja estudada outra solução para trabalhar com esses componentes em conjunto.

## 5 – CONCLUSÕES

Este trabalho consistiu no estudo e desenvolvimento da tecnologia de *Force Feedback* aplicada em um sistema eletromecânico de uma garra robótica. Para tal, foram utilizados componentes eletrônicos compatíveis com o microcontrolador Arduino UNO para ênfase na simulação do sistema eletrônico de controle.

A utilização do Arduino UNO foi suficiente para acoplar as entradas e saídas de *interface* com o potenciômetro do gatilho, os servo motores da garra e do FF, e as conexões com o módulo HX711 responsável pela conversão do sinal do transdutor de 5 kg. A alimentação do sistema foi feita através de uma fonte de 5V contínuo adaptada a *proto board* para evitar danificação dos pinos de alimentação do Arduino por excesso de corrente elétrica e demonstrou-se suficiente para o protótipo.

A calibração realizada para a célula de carga de 5 kg e a estrutura confeccionada para seu funcionamento apresentou resultados confiáveis e estáveis, e mesmo após manter o sistema ligado por longos períodos e em diferentes temperaturas ambiente, os resultados obtidos se mantiveram dentro do desejado. Portanto, valida-se também que esse transdutor que funciona conforme os princípios da ponte de Wheatstone atende o desenvolvimento do projeto estudado.

Conforme discutido nos resultados, o principal obstáculo encontrado no sistema é o conflito no controle de interrupções do Arduino devido ao funcionamento do HX711 e o funcionamento dos motores. Devido a esse conflito, foi possível atingir até 3,5 kg de massa aplicada ao transdutor de maneira estável, apesar de suportar até 5 kg. Para prosseguir com estudos e testes com forças maiores é necessário revisar as tecnologias e compatibilidade no momento de desenvolver o código do sistema.

Portanto, limitando-se a 3,5 kg foi possível simular o *Force Feedback* com sucesso através dos componentes utilizados no projeto e confirmar esse valor atingido através do monitor *serial* da Arduino IDE.

## 6 – PERSPECTIVAS DE CONTINUAÇÃO

Este foi o início de um projeto desenvolvido com o intuito de estudar novas aplicações para tecnologias de *Haptic Feedback*, mais precisamente o controle e simulação da força exercida por um sistema robótico.

Por tratar-se de um protótipo cujo objetivo principal foi validar os conceitos e tecnologias idealizadas para obter-se o resultado desejado, ficam abertas diversas possibilidades para prosseguir com o desenvolvimento do sistema e deixá-lo com a efetiva funcionalidade de uma garra capaz de apertar os mais diversos objetos e um controlador que receba um *force feedback* refinado e consistente.

Nas seções a seguir estão algumas sugestões de continuidade do projeto que foram estudadas e idealizadas ao longo do desenvolvimento deste trabalho, porém não foram desenvolvidas devido a limitação de recursos, espaço, tempo e alto custo de implementação.

### 6.1 – Modelagem da garra robótica [37]

O protótipo desenvolvido para este trabalho consistiu apenas em uma representação em pequena escala com ênfase no funcionamento eletrônico do sistema de *Force Feedback*. Ou seja, para torná-lo funcional é necessária a modelagem de uma garra robótica que de fato se movimente de acordo com o controle de posição de um servo motor.

As características mais importantes a serem levadas em consideração para essa modelagem são a geometria da garra e o material utilizado em sua confecção. A combinação desses dois fatores determina o nível de imprecisão que será inserido no sistema devido às deformações do material, assim como o centro de massa (CM) que deve ser utilizado para o dimensionamento do servo motor. A posição desse CM influencia na determinação do torque que será aplicado ao eixo do motor, pois quanto mais distante o CM do motor, maior o torque. Caso o acoplamento não seja de forma direta, mas sim através de polias, isso também deve ser levado em consideração.



## **6.2 – Dimensionamento do sistema da braçadeira acoplada ao gatilho**

O mecanismo de controle do FF no gatilho expandido na Seção 3.5 foi idealizado, mas não confeccionado. Portanto, para efeito de simulação no protótipo a conversão de angulação do servo motor utilizado nesse controle para a quantidade de força simulada foi feita de forma arbitrária.

Dessa forma, como possibilidade de continuação fica a necessidade de determinar o material utilizado para a haste do gatilho que será pressionada e o material da braçadeira. A partir disso é possível obter de forma matemática ou empírica, conforme os coeficientes de atrito, a relação de aperto dessa braçadeira para força aplicada no gatilho.

Por fim, é possível dimensionar um sistema de engrenagens e rosca para converter a angulação do servo motor na quantidade que o mecanismo deve ser girado para pressionar a braçadeira de acordo com a força.

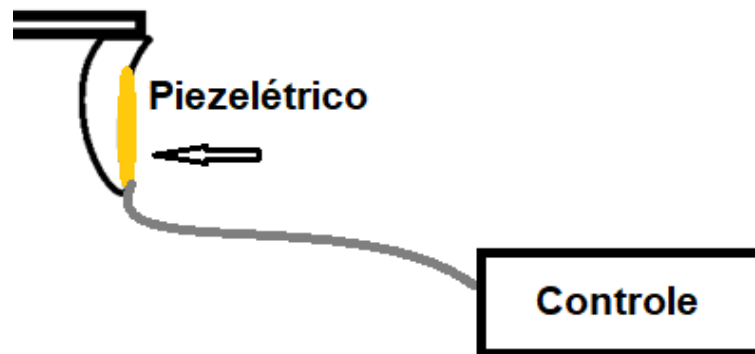
## **6.3 – Mecanismo para retorno do gatilho e afrouxamento do Force Feedback**

Conforme apresentado na Seção 3.5, o gatilho do sistema que é apertado pelo dedo de uma pessoa possui uma haste que deve ser esmagada por uma braçadeira de maneira proporcional a força exercida pela garra robótica. Apesar do gatilho não ter sido desenvolvido, apenas representado pelo potenciômetro de 10k $\Omega$ , durante o desenvolvimento teórico do projeto foi observado que o sistema idealizado possui um obstáculo: identificar quando o gatilho não está mais sendo pressionado para afrouxar a braçadeira e realizar o movimento para abrir a garra.

Há diversas alternativas para identificar que o gatilho não está mais sendo pressionado e deve ser aberto. Durante os estudos deste sistema foram levantadas duas soluções:

I – Inserir um pequeno circuito com um transdutor piezelétrico no gatilho para captar a variação de força que o dedo exerce e dessa maneira determinar se ele está realizando um movimento de pressionamento ou afrouxamento. A Figura 25 mostra essa ideia.

**FIGURA 25 - SISTEMA DE DETECÇÃO DE MOVIMENTO COM UM PIEZELÉTRICO**

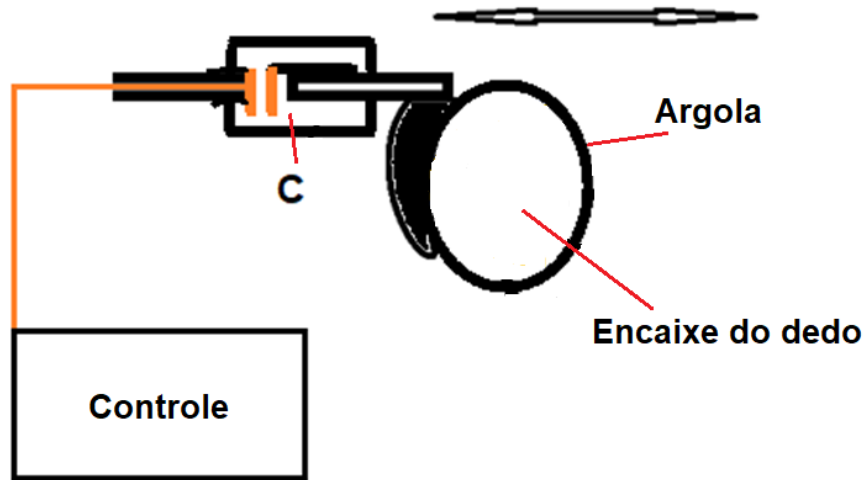


Utilizar apenas esse controle para tomar a decisão provocaria eventuais falhas no sistema caso o material apertado pela garra apresente uma mudança de comportamento na resistência que ele impõe ao ser pressionado. Por exemplo: caso a garra esteja apertando um ovo e a casca quebre, a força exercida subitamente cairia para zero, provocando o afrouxamento da braçadeira. Com esse afrouxamento, a variação de força detectada pelo sistema com o piezelétrico no gatilho tomaria a decisão de abrir a garra, o que não deve ocorrer.

Por isso, para tal alternativa funcionar, deve ser levado em conta também a variação da força que a garra está exercendo para tomar a decisão, portanto essa solução aumentaria a complexidade do código utilizado. Além disso, apesar dessa solução tomar uma decisão do que deve ser feito, ela não conta com um mecanismo para realmente movimentar o gatilho para a direção contrária do aperto e acompanhar o recuo do dedo.

II – Projetar um gatilho com uma forma justa em torno de todo o dedo da pessoa, deixando a decisão puramente na movimentação do dedo e forçando o resto do sistema a acompanhá-lo, vide Figura 26.

FIGURA 26 - SISTEMA DE DETECÇÃO DE MOVIMENTO COM UMA ARGOLA E UM CONTATO ELÉTRICO



Conforme a Figura 26, uma argola fixa no gatilho se ajusta ao redor do dedo da pessoa, portanto qualquer movimento que o dedo realizar, o gatilho acompanha. Para realizar o movimento de abertura da garra, há um pequeno mecanismo na haste que é esmagada pela braçadeira.

A parte da haste conectada ao gatilho é móvel e alinhada com o outro lado da haste. Entre elas há um contato 'C' metálico fechando um circuito de controle e um pequeno vão. Quando o dedo recua, esse contato é aberto, sinalizando ao sistema que a braçadeira deve ser afrouxada para permitir o livre recuo da haste. Caso a pessoa decida apertar novamente, o pequeno contato será fechado e sinalizará para que o motor aperte novamente a braçadeira de maneira proporcional a força atual da garra.

Esta solução não acrescenta tanta complexidade igual a solução do piezelétrico e já conta com o mecanismo para recuar a garra. Os pontos negativos são que esse pequeno vão entre as partes da haste para abrir o contato afeta a fidelidade do *Force Feedback* no dedo, pois insere uma pequena zona morta, e o movimento da garra é controlado apenas pelo dedo, portanto caso a pessoa abandone o sistema no meio, a garra não retornará a sua posição completamente aberta.

#### **6.4 – Dimensionamento do motor da garra**

Conforme os resultados do projeto, o protótipo desenvolvido permitiu o trabalho de maneira estável para até 3,5 kg de massa aplicada na garra. Entretanto, o micro servo motor utilizado não é capaz de aplicar o torque necessário para essa força, pois conta com apenas 1,2 kg.cm para 5V de alimentação, portanto a força foi aplicada ao transdutor de forma manual.

Para o desenvolvimento de um sistema 100% funcional é necessário o dimensionamento desse motor de forma que atenda a força aplicada ao sistema com uma margem de segurança. Para tal, deve ser levado em conta principalmente a estrutura da garra, conforme apresentado na Seção 5.1.1, a forma de acoplamento do motor e a direção de movimentação da garra.

Caso seja apenas uma pinça que se movimenta verticalmente, há o peso da garra influenciando diretamente no torque exercido, porém caso seja uma pinça na orientação horizontal, há a força peso perpendicular ao seu acoplamento no motor.

#### **6.5 – Aplicação de tecnologias wireless entre os módulos do sistema**

O sistema desenvolvido foi todo projetado utilizando conexões através de fios condutores. Enquanto para um pequeno protótipo e para testes em ambientes controlados é possível manter um sistema todo conectado por condutores, as possibilidades de aplicação são limitadas a distâncias e espaços físicos pré-determinados.

A modularização do projeto em diferentes blocos permite que sejam implementadas soluções de comunicação sem fio, dessa maneira flexibilizando a instalação de seus componentes em diferentes locais e aumentando o leque de cenários onde é possível trabalhar com essa tecnologia.

Um exemplo de comunicação *wireless* é a utilização do módulo *bluetooth* HC-06 do Arduino. Através dele seria possível realizar essa separação dos diferentes módulos do projeto para comunicação dos dados TTL: gatilho, sistema de controle e garra. O principal ponto de atenção ao realizar essa separação é a necessidade de fontes de alimentação para cada parte.

**Módulo HC-06 (especificações):** Interface para comunicação *bluetooth* entre os blocos do protótipo.

- Pinos: VCC, GND, TXD, RXD
- Tensão de alimentação: 3,6-6V
- Cobertura do sinal: até 10m
- Especificação Bluetooth: v2,0 -EDR
- Frequência: banda 2,4GHz ISM
- Modulação: GFSK
- Potência de transmissão: 4 dBm, classe 2
- Sensibilidade: -84dBm, 0,1% BER
- Taxa de dados: Assíncrono 2,1 Mbps; Síncrono 1 Mbps
- Perfis suportados: *Bluetooth Serial (port slave)*
- Alimentação: 5VDC, 50mA
- Taxa máxima de transmissão serial: 1382400bps
- Configuração padrão: 9600bps/Senha123
- Dimensões: 4,3 \* 1,6 \* 0,7 cm
- Antena embutida na placa

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] RODRIGUES, JC. Os três níveis da interação homem-máquina. **Update or die**, Setembro, 2016. Disponível em: <https://www.updateordie.com/2016/09/25/os-tres-niveis-da-interacao-homem-maquina/>. Acesso em: 13 ago 2021.
- [2] BORGES, Rebeca. Cirurgiões realizam cirurgias a distância no Canadá. **Unasus**, Agosto, 2014. Disponível em: <https://www.unasus.gov.br/noticia/cirurgioes-realizam-cirurgias-distancia-no-canada>. Acesso em: 14 ago 2021.
- [3] MORSCH, José Aldair. A TELECIRURGIA É UMA ESPECIALIDADE DA TELEMEDICINA. Morsch Telemedicina, Fevereiro, 2018. Disponível em: <https://telemedicinamorsch.com.br/blog/cirurgia-a-distancia>. Acesso em: 14 ago 2021.
- [4] ULTRALEAP. **Ultraleap**, 2021. What is Haptic Feedback?. Disponível em: <https://www.ultraleap.com/company/news/blog/what-is-haptic-feedback/>. Acesso em: 16 ago 2021.
- [5] NISHINO, Hideaki. Apresentamos o DualSense, o Novo Controle Sem Fio do PlayStation 5. **Playstation.blog**, Abril, 2020. Disponível em: <https://blog.br.playstation.com/2020/04/07/apresentando-dualsense-o-novo-controle-sem-fio-do-playstation-5/>. Acesso em: 16 ago 2021.
- [6] ULTRALEAP. Ultraleap, 2021. Haptics. Disponível em: <https://www.ultraleap.com/haptics/>. Acesso em: 16 ago 2021.
- [7] MELDAU, Débora Carvalho. Suturas. **Infoescola**, 2021. Disponível em: <https://www.infoescola.com/medicina/suturas/>. Acesso em: 22 ago 2021.
- [8] AIR LIQUIDE HEALTHCARE. **Air Liquide Healthcare**, 2021. Laparoscopia. Disponível em: <https://www.airliquidehealthcare.com.br/especialidades/laparoscopia>. Acesso em: 22 ago 2021.
- [9] SALVALAGGIO, Paolo Rogério de Oliveira. Colecistectomia robótica: entenda como funciona o procedimento. **Clínica Hepatogastro**, Julho, 2021. Disponível em: <https://hepatogastro.com.br/colecistectomia-robotica-entenda-como-funciona-o-procedimento/>. Acesso em: 23 ago 2021.
- [10] Okamura AM. Haptic feedback in robot-assisted minimally invasive surgery. *Curr Opin Urol*. 2009;19(1):102-107. doi:10.1097/MOU.0b013e32831a478c. Disponível em: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2701448/>. Acesso em: 23 ago 2021.
- [11] Kobayashi Y, Moreira P, Liu C, Poignet P, Zemiti N, Fujie MG. Haptic feedback control in medical robots through fractional viscoelastic tissue model. *Annu Int Conf IEEE Eng Med Biol Soc*. 2011;2011:6704-6708. doi:10.1109/IEMBS.2011.6091653. Disponível em: <https://pubmed.ncbi.nlm.nih.gov/22255877/>. Acesso em: 23 ago 2021.
- [12] Abiri, A., Pensa, J., Tao, A. et al. Multi-Modal Haptic Feedback for Grip Force Reduction in Robotic Surgery. *Sci Rep* **9**, 5016 (2019). <https://doi.org/10.1038/s41598-019-40821-1>. Disponível em: <https://www.nature.com/articles/s41598-019-40821-1>. Acesso em: 24 ago 2021.

- [13] HENEY, Paul. Challenges of building haptic feedback for surgical robots. **The Robot Report**, Julho, 2019. Disponível em: <https://www.therobotreport.com/haptic-feedback-design-challenges-surgical-robots/>. Acesso em: 24 ago 2021.
- [14] ZELENOSKY, Ricardo. Arquitetura de Microcontroladores Modernos. **MZ Editora**, 2002. Disponível em: [http://www.mzeditora.com.br/artigos/mic\\_modernos.htm](http://www.mzeditora.com.br/artigos/mic_modernos.htm). Acesso em: 24 ago 2021.
- [15] COUTINHO, Thiago. Veja como o Arduino oferece inúmeras possibilidades para a fabricação de protótipos. **Voitto**, Outubro, 2020. Disponível em: <https://www.voitto.com.br/blog/artigo/arduino-o-que-e>. Acesso em: 14 Jul 2021.
- [16] THOMSEN, Adilson. Qual Arduino Comprar? Conheça os Tipos de Arduino. **Filipe Flop**, Fevereiro, 2014. Disponível em: <https://www.filipeflop.com/blog/tipos-de-arduino-qual-comprar/>. Acesso em: 14 Jul 2021.
- [17] ARDUINO. **Arduino CC**, 2021. Language Reference. Disponível em: <https://www.arduino.cc/reference/en/>. Acesso em: 16 Jul 2021.
- [18] STACKEXCHANGE. **Arduino**, 2017. How to output a true analog voltage at output pin. Disponível em: <https://arduino.stackexchange.com/questions/31664/how-to-output-a-true-analog-voltage-at-output-pin>. Acesso em: 16 Jul 2021.
- [19] ARDUINO. **Arduino CC**, 2017. Low Pass Filter for Pulse Width Modulation?. Disponível em: <https://forum.arduino.cc/t/low-pass-filter-for-pulse-width-modulation/498556/15>. Acesso em: 17 Jul 2021.
- [20] ALCIATORE, David G. HISTAND, Michael B. **Introdução à Mecatrônica e aos Sistemas de Medições**. 4ª Ed. Mc Graw Hill Education. Cap. 1. Disponível em: <https://statics-submarino.b2w.io/sherlock/books/firstChapter/119235712.pdf>. Acesso em: 20 Jul 2021.
- [21] STRAUB, Matheus Gebert. BRAÇO ROBÓTICO ARDUINO COM SERVO MOTOR E JOYSTICK. **Usina Info**, Julho, 2016. Disponível em: <https://www.usinainfo.com.br/blog/braco-robotico-arduino-com-servo-motor-e-joystick/>. Acesso em: 20 Jul 2021.
- [22] TYRSINA, Radu. 5 best Arduino simulators for PC to use today. **Windows Report**, Dezembro, 2020. Disponível em: <https://windowsreport.com/arduino-simulators/>. Acesso em: 20 Jul 2021.
- [23] NOBRES DO GRID. **Nobres do Grid**, Abril, 2015. A eletrônica dos aceleradores. Disponível em: [http://www.nobresdogrid.com.br/site/index.php?option=com\\_content&view=article&id=1506:a-eletronica-dos-aceleradores&catid=117:autorama-uma-paixao&Itemid=288](http://www.nobresdogrid.com.br/site/index.php?option=com_content&view=article&id=1506:a-eletronica-dos-aceleradores&catid=117:autorama-uma-paixao&Itemid=288). Acesso em: 02 ago 2021.
- [24] MARINOSTORE, **Marinostore**, 2021. Célula de carga 5kg. Disponível em: <https://www.marinostore.com/sensores/celula-de-carga-5kg>. Acesso em: 02 ago 2021.
- [25] AL-MUTLAQ, Sarah. Load Cell Amplifier HX711 Breakout Hookup Guide. **Sparkfun**, 2021. Disponível em: <https://learn.sparkfun.com/tutorials/load-cell-amplifier-hx711-breakout-hookup-guide/all>. Acesso em: 10 ago 2021.
- [26] AVIA SEMICONDUCTOR. **Sparkfun**, 2021. 24-Bit Analog-to-Digital Converter (ADC) for Weigh Scales. Disponível em:

[https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711\\_english.pdf](https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf). Acesso em: 10 ago 2021.

[27] UTILCELL. **Utilcell**, 2021. Load cell electrical circuit. Disponível em: <https://www.utilcell.com/en/load-cell-electrical-circuit/> . Acesso em: 10 ago 2021.

[28] HELERBROCK, Rafael. Ponte de Wheatstone. **Mundo Educação**, 2021. Disponível em: <https://mundoeducacao.uol.com.br/fisica/ponte-wheatstone.htm>. Acesso em: 12 ago 2021.

[29] LOCATELLI, Caroline. Balança com Célula de Carga e HX711. **Curto Circuito**, Dezembro, 2019. Disponível em: <https://www.curtocircuito.com.br/blog/Categoria%20Arduino/balanca-com-celula-de-carga-e-hx711>. Acesso em: 13 ago 2021.

[30] MYBOTIC. Tutorial to Interface HX711 Balance Module With Load Cell. **Instructables Circuits**, 2017. Disponível em: <https://www.instructables.com/How-to-Interface-HX711-Balance-Module-With-Load-Ce/>. Acesso em: 13 ago 2021.

[31] DEGRAWST. Arduino Scale With 5kg Load Cell and HX711 Amplifier. **Instructables Circuits**, 2017. Disponível em: <https://www.instructables.com/Arduino-Scale-With-5kg-Load-Cell-and-HX711-Amplifi/>. Acesso em: 13 ago 2021.

[32] BOGDE et al. HX711. **Github**, 2021. Disponível em: <https://github.com/bogde/HX711>. Acesso em: 14 ago 2021.

[33] BRAINY-BITS. How to connect and use a Weight Sensor + HX711 with an Arduino – Tutorial. **Youtube**, Julho, 2018. Disponível em: <https://www.youtube.com/watch?v=pZpZdu97JYw>. Acesso em: 14 ago 2021.

[34] ARDUINO. **Arduino CC**, 2020. HX711, 5kg Load Cell, Arduino Uno showing constant -1 reading. Disponível em: <https://forum.arduino.cc/t/hx711-5kg-load-cell-arduino-uno-showing-constant-1-reading/638050/4>. Acesso em: 03 set 2021.

[35] ARDUINO. **Arduino CC**, 2020. Regular Servo Jitter - every 57 seconds. Disponível em: <https://forum.arduino.cc/t/regular-servo-jitter-every-57-seconds/665684/6>. Acesso em: 04 set 2021.

[36] GNU. **GNU**. Index of Directives. Disponível em: <https://gcc.gnu.org/onlinedocs/cpp/Index-of-Directives.html#Index-of-Directives>. Acesso em: 04 set 2021.

[37] BUFFONI, Salete Souza de Oliveira. Deformação. **ESCOLA DE ENGENHARIA INDUSTRIAL METALÚRGICA DE VOLTA REDONDA**, DISCIPLINA: RESISTÊNCIA DOS MATERIAIS. Disponível em: <http://www.professores.uff.br/salete/wp-content/uploads/sites/111/2017/08/aula5.pdf>. Acesso em: 06 set 2021.