



**UNIVERSIDADE DO SUL DE SANTA CATARINA**

**HENRIQUE HOFFMANN**

**CONTROLE DE ACESSO AUTOMÁTICO  
POR MEDIÇÃO DE TEMPERATURA CORPORAL**

Palhoça

2021

**HENRIQUE HOFFMANN**

**CONTROLE DE ACESSO AUTOMÁTICO  
POR MEDIÇÃO DE TEMPERATURA CORPORAL**

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia Elétrica da Universidade do Sul de Santa Catarina como requisito parcial à obtenção do título de Engenheiro Eletricista.

Orientadora: Prof<sup>ra</sup>. Francielli Scarpini Barbosa Cordeiro, Me. Eng<sup>a</sup>

Palhoça

2021

**HENRIQUE HOFFMANN**

**CONTROLE DE ACESSO AUTOMÁTICO  
POR MEDIÇÃO DE TEMPERATURA CORPORAL**

Este Trabalho de Conclusão de Curso foi julgado adequado à obtenção do título de Engenheiro Eletricista e aprovado em sua forma final pelo Curso de Engenharia Elétrica da Universidade do Sul de Santa Catarina.

Palhoça, 26 de junho de 2021.

---

Professora e Orientadora Francielli Scarpini Barbosa Cordeiro, Me. Eng<sup>a</sup>.  
Universidade do Sul de Santa Catarina

---

Prof. Fabiano Max da Costa, Esp. Eng.  
Universidade do Sul de Santa Catarina

---

Convidado, Christian Lemos Botelho, Eng.  
Intelbras S/A - Indústria de Telecomunicação Eletrônica Brasileira

## **AGRADECIMENTOS**

Agradeço, primeiramente, a Deus por me dar força e saúde para concluir este desafio e me abençoar com a conclusão desta etapa da minha vida.

A minha família, meu pai José Rogério Hoffmann, mãe Maiza Costa Hoffmann, irmão Eduardo Stahelin Costa Hoffmann e irmã Camila Cristina Hoffmann (in memoriam) que apesar de todas as dificuldades, me ajudaram na realização do meu sonho.

Agradeço também a minha noiva Thais Silveira que me incentivou e sempre esteve ao meu lado nos momentos mais difíceis, tornando-se essencial para a conclusão do curso.

A minha orientadora Francielli Scarpini Barbosa Cordeiro por todo apoio e paciência ao longo da elaboração do meu projeto final e a todas as pessoas que direta ou indiretamente contribuíram para a realização da minha pesquisa.

“Concentre todos seus pensamentos na tarefa que está realizando. Os raios de sol não queimam até que sejam colocados em foco.” (Alexander Graham Bell).

## RESUMO

Tendo em vista que o aumento da temperatura corporal é um dos principais sintomas para o COVID-19, pesquisa-se sobre uma medida automatizada para o controle de acesso através da análise da temperatura corporal utilizando o microcontrolador Arduino e o sensor de temperatura MLX90614, a fim de evitar o contágio pelo novo coronavírus. Para tanto, é necessário um estudo sobre o sensor de temperatura a ser utilizado, o desenvolvimento de um software para comunicação entre os dispositivos de controle de acesso e medição de temperatura via Arduino e o desenvolvimento de um hardware automatizado para liberação ou não do acesso do cliente e/ou funcionário através da medição da temperatura. Realiza-se, então, uma pesquisa em artigos científicos acerca dos sintomas do COVID-19, da análise dos equipamentos eletrônicos utilizados para o desenvolvimento do projeto, e acompanhamento dos testes e resultados. Diante disso, verifica-se que os resultados do projeto se tornam viáveis para aplicação em estabelecimentos para o controle de pessoas, assim evitando o contágio da doença, o que impõe a constatação de que o método utilizado traz uma confiabilidade para o protótipo, tornando-se uma ferramenta de fácil construção e utilização, com um custo acessível para implementação.

Palavras-chave: Controle de Temperatura. Microcontrolador. MLX90614. COVID-19.

## **ABSTRACT**

Since the increase in body temperature is one of the main symptoms of COVID-19, an automated way to control access is sought through body temperature analysis using the Arduino microcontroller and the MLX90614 temperature sensor, in order to avoid contagion with the new coronavirus. For this to happen, it is necessary to carry out a study on the temperature sensor to be used, the development of software for communication between access control devices and temperature measurement using Arduino and the development of automated hardware for release or not. customer or employee access through temperature measurement. Then, a search is made in scientific articles about the symptoms of COVID-19, the analysis of electronic equipment used for the development of the project, and monitoring of tests and results. Therefore, it was verified that the results of the project become viable for application in establishments for the control of people, avoiding the contagion of the disease, which imposes the observation that the method used brings reliability to the prototype, however the sensor of temperature did not behave as specified by the supplier, making it a tool of easy construction and use, with an affordable cost for implementation, but it is necessary to apply a more accurate temperature sensor.

**Keywords:** Temperature Access Control. Microcontroller. MLX90614. COVID-19.

## LISTA DE ILUSTRAÇÕES

Figura 1 - Controlador de acesso facial .....	18
Figura 2 - Pistola de medição de temperatura .....	18
Figura 3 - Tamanho do Arduino Nano .....	19
Figura 4 - Arduino UNO .....	21
Figura 5 - Arduino Mega .....	22
Figura 6 - Arduino Leonardo.....	23
Figura 7 - Arduino Due .....	24
Figura 8 - Arduino Mega ADK .....	24
Figura 9 - Arduino Nano .....	25
Figura 10 - Arduino Pro Mini.....	26
Figura 11 - Arduino Esplora.....	27
Figura 12 - Sensor MLX90614.....	29
Figura 13 - Diagrama Funcional MLX90614.....	30
Figura 14 - Pinagem do sensor MLX90614 .....	30
Figura 15 - Zonas do IR.....	31
Figura 16 - Diagrama de blocos MLX90614.....	32
Figura 17 - Precisão em relação ao ambiente .....	33
Figura 18 - Ponto de visão.....	34
Figura 19 - Especificações HC-SR505 .....	34
Figura 20 - Equipamentos utilizados .....	36
Figura 21 - Diagrama de ligação dos componentes.....	37
Figura 22 - Plataforma de testes .....	37
Figura 23 - Projeto finalizado (Vista posterior).....	38
Figura 24 - Projeto finalizado (Vista superior).....	38
Figura 25 - Versão do software Arduino .....	39
Figura 26 - DMT 7200 FT .....	40

## LISTA DE GRÁFICOS

Gráfico 1 - Curva de aprendizagem sensor na testa .....	42
Gráfico 2 - Curva de aprendizagem sensor no punho .....	42

## LISTA DE TABELAS

Tabela 1 - Portas digitais .....	39
Tabela 2 - Curva de aprendizagem do sensor .....	41

## LISTA DE QUADROS

Quadro 1 - Especificações dos principais modelos de Arduino .....	20
Quadro 2 - Linguagens de programação mais utilizadas.....	27
Quadro 3 - Descrição dos pinos MLX90614.....	30

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO.....</b>	<b>13</b>
1.1	JUSTIFICATIVA .....	14
1.2	DEFINIÇÃO DO PROBLEMA .....	14
1.3	OBJETIVOS .....	15
<b>1.3.1</b>	<b>Objetivo Geral .....</b>	<b>15</b>
<b>1.3.2</b>	<b>Objetivos Específicos.....</b>	<b>15</b>
1.4	DELIMITAÇÕES .....	15
1.5	METODOLOGIA DE PESQUISA.....	16
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA.....</b>	<b>17</b>
2.1	CONTROLADORES DE ACESSO .....	17
2.2	MICROCONTROLADORES.....	19
2.3	ARDUINO .....	19
<b>2.3.1</b>	<b>Arduino UNO.....</b>	<b>21</b>
<b>2.3.2</b>	<b>Arduino Mega 2560.....</b>	<b>21</b>
<b>2.3.3</b>	<b>Arduino Leonardo.....</b>	<b>22</b>
<b>2.3.4</b>	<b>Arduino Due.....</b>	<b>23</b>
<b>2.3.5</b>	<b>Arduino Mega ADK.....</b>	<b>24</b>
<b>2.3.6</b>	<b>Arduino Nano .....</b>	<b>25</b>
<b>2.3.7</b>	<b>Arduino Pro Mini.....</b>	<b>25</b>
<b>2.3.8</b>	<b>Arduino Esplora .....</b>	<b>26</b>
2.4	LINGUAGEM DE PROGRAMAÇÃO .....	27
<b>2.4.1</b>	<b>Linguagem C.....</b>	<b>28</b>
2.5	SENSORES DE TEMPERATURA.....	28
<b>2.5.1</b>	<b>Sensor de temperatura MLX90614 .....</b>	<b>29</b>
<b>2.5.2</b>	<b>Princípio de processamento do sinal.....</b>	<b>31</b>
2.6	SENSOR DE PRESENÇA HC-SR505.....	34
<b>3</b>	<b>DESENVOLVIMENTO.....</b>	<b>35</b>
3.1	EQUIPAMENTOS UTILIZADOS .....	35
3.2	PROTÓTIPO.....	36
3.3	LINGUAGEM DE PROGRAMAÇÃO .....	39
<b>4</b>	<b>CONCLUSÃO.....</b>	<b>43</b>
	<b>REFERÊNCIAS .....</b>	<b>44</b>

<b>ANEXOS .....</b>	<b>46</b>
<b>ANEXO A – CÓDIGO DE PROGRAMAÇÃO .....</b>	<b>47</b>

## 1 INTRODUÇÃO

Em dezembro de 2019, o Centro de Controle e Prevenção de Doenças (CDC) da China identificou um surto de doença respiratória em trabalhadores de um mercado de alimentos de Wuhan, capital da província de Hubei. Posteriormente, identificou-se como causador da doença Covid-19 um novo coronavírus, denominado SARS-CoV-2 (MINISTÉRIO DA SAÚDE, 2020).

A doença disseminou-se rapidamente na província de Hubei e, desde então, atingiu 235 países dos cinco continentes, causando mais de 1 milhão de mortes. A Organização Mundial da Saúde (OMS), declarou a Covid-19 uma pandemia em 11 de março de 2020 (WHO, 2020).

Ainda segundo a (WHO, 2020), os principais sintomas da Covid-19 são febre, tosse seca e cansaço. Outros sintomas menos comuns incluem dores no corpo, congestão nasal, dor de cabeça, conjuntivite, dor de garganta, diarreia, perda de paladar e olfato.

Preocupados com o avanço no número de casos registrados com o novo coronavírus, os órgãos mundiais de saúde junto aos líderes governamentais aderiram medidas preventivas para reduzir o contágio da doença (WHO, 2020).

Para evitar o contágio, algumas medidas simples foram adotadas: higienização das mãos com água e sabão, manter o distanciamento social, uso de máscaras, disponibilização de álcool em gel 70% para higienização nos estabelecimentos comerciais, controle na taxa de ocupação do ambiente e aferição de temperatura na entrada de estabelecimentos (WHO, 2020).

Observando-se estas medidas preventivas, nota-se que em estabelecimentos comerciais está sendo muito utilizado o termômetro infravermelho sem contato para aferição da temperatura.

Como uma das formas de contágio da doença se dá pela proximidade de pessoas infectadas, os termômetros infravermelhos são um risco para os profissionais que estão à frente fazendo esse procedimento, pois os clientes precisam ser medidos a distâncias entre 0,5m e 1,5m. Estes equipamentos são utilizados por seu baixo custo comparado as soluções mais seguras.

O dispositivo que será utilizado para aferição da temperatura será o Arduino, acoplado a um sensor de temperatura MLX90614 que irá coletar os dados para que o microcontrolador possa decidir se o acesso será ou não liberado para o cliente ou funcionário. Esse comando será direcionado a uma fechadura magnética que ficará responsável por destravar a porta.

Este trabalho tem o objetivo de desenvolver um controlador de acesso que atenda essa demanda de controle de usuários e/ou funcionários com febre a estabelecimentos com precisão e um custo mais acessível que possibilite a automação e segurança de todos.

## 1.1 JUSTIFICATIVA

Com o aumento de casos referente ao COVID-19, a busca de tecnologias para aferição de temperatura corporal vem aumentando significativamente se.

De acordo com o decreto nº 21.569, de 15 de maio de 2020, que exige a aferição de temperatura de todos os clientes e funcionários antes de adentram a ambientes fechados, para isso, os setores comerciais estão utilizando diversos métodos para cumprir e se adequar a legislação.

Um deles é a utilização de termômetros infravermelhos sem contato, um método muito invasivo e arriscado para as pessoas que o utilizam, tornando-se um processo arriscado.

Outra alternativa são os controladores de acesso facial com detecção de temperatura e também as câmeras térmicas, porém são soluções com um alto custo de investimento por parte dos empresários, que em virtude da crise causada pela pandemia preferem não investir grandes somas financeiras nessas soluções, assim arriscando seus funcionários.

Este trabalho tem como objetivo principal criar uma forma simples e automatizada de controlar o acesso de pessoas com temperatura alta em ambientes comerciais, evitando assim o contágio pelo novo coronavírus, utilizando sensores, micro controladores e atuadores de uso comum na indústria. Esta automatização trará mais segurança aos colaboradores que estão na linha de frente do combate ao COVID-19 nos estabelecimentos comerciais.

## 1.2 DEFINIÇÃO DO PROBLEMA

Tendo em vista que um dos sintomas característicos para o COVID-19 é a febre, muitos locais estão fazendo essa aferição de temperatura manualmente na entrada dos estabelecimentos, com isso formam-se aglomerações nas entradas dos estabelecimentos e também na contaminação pelo vírus.

Para agilizar o processo de medição e diminuir o risco de contaminação pelo vírus, o controlador de acesso com medição de temperatura gerencia a entrada de pessoas com temperaturas altas, evitando assim a contaminação de outras pessoas, de forma automática e sem contato.

Com a solução de medição de temperatura automatizada proposta nesse trabalho de conclusão de curso, utilizando o Arduino, um sensor de temperatura MLX90614 e os componentes necessários de funcionamento que serão descritos nos capítulos abaixo, tornando o custo muito mais viável.

### 1.3 OBJETIVOS

O objetivo principal do trabalho consiste em criar uma forma simples e automatizada de controlar o acesso de pessoas com temperatura alta em ambientes comerciais, evitando assim o contágio pelo novo coronavírus.

Desenvolvimento de um controlador de acesso por medição de temperatura corporal utilizando Arduino e o sensor de temperatura MLX90614, para automatizar o processo de medição de temperatura das pessoas que queiram adentrar estabelecimentos, com a função de limitar o acesso de pessoas com sintomas de febre, que é um dos sintomas característicos para o COVID-19.

#### 1.3.1 Objetivo Geral

Automatizar o processo de medição de temperatura de pessoas que queiram adentrar à estabelecimentos que por lei são obrigados a fazer esse tipo de medição na entrada dos estabelecimentos.

#### 1.3.2 Objetivos Específicos

- Estudo sobre o sensor de temperatura MLX90614;
- Desenvolvimento de um software para comunicação entre os dispositivos de controles de acesso e medição de temperatura via Arduino;
- Desenvolvimento de hardware automatizado para liberação ou não do acesso do cliente e/ou funcionário através da medição da temperatura;

### 1.4 DELIMITAÇÕES

Este trabalho limita-se ao estudo e implementação de um protótipo microcontrolado capaz de realizar o acionamento de uma fechadura elétrica, controladas de forma inteligente

através da aferição da temperatura especificada, para estabelecer segurança e confiabilidade do sistema de controle de acesso térmico.

## 1.5 METODOLOGIA DE PESQUISA

Segundo Rauen (2015, p.140) “a revisão da bibliografia consiste em um conjunto essencial de procedimentos que, embora se apresente numa sequência, deve ser executado em todas as etapas da pesquisa.” Com base nesta afirmação, pode-se entender que a metodologia adequada para este projeto é a de revisão bibliográfica.

O método de pesquisa adotado para este trabalho será o de revisão bibliográfica e implementação experimental, esta revisão busca artigos científicos acerca dos sintomas do SARS-CoV-2, os componentes eletrônicos para desenvolvimento do projeto, o desenvolvimento do projeto em si e a implementação experimental na realização dos testes e acompanhamento dos resultados.

O método de implementação experimental se faz fundamental para este trabalho de conclusão de curso, para colocar em prática os estudos aqui realizados e confirmação dos objetivos desejados.

## 2 FUNDAMENTAÇÃO TEÓRICA

Os sistemas de controle de acesso são dispositivos que têm a função de manter a segurança de determinados locais, como empresas, estabelecimentos comerciais, hotéis entre outros.

O uso de dispositivos de controle de acesso vem se tornando algo comum nos mais diversos ramos de atividades e ambientes organizacionais com a função de manter a segurança nos locais de atuação.

Segundo Peixoto (2013), há uma necessidade de maior segurança para usuários e instituições, o que aumenta a procura por este tipo de tecnologia. Os desafios de implementação e utilização diminuiram com o avanço da tecnologia, porém “ainda possuem alto custo de implantação e geralmente não possuem padronização das funcionalidades, principalmente em se tratando do sistema que os gerenciam” (PEIXOTO, 2013, p. 8).

Neste capítulo será apresentada uma revisão literária dos principais tópicos, tecnologias e conceitos que serão utilizados no decorrer deste trabalho. É nessa etapa que os sintomas da doença Covid-19 serão descritos, bem como as características do microcontrolador Arduino, conceitos de controle de acesso e assuntos que complementam o trabalho.

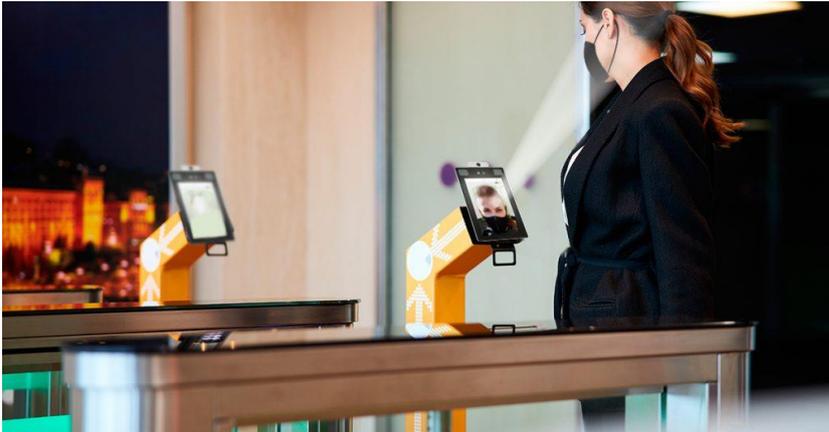
### 2.1 CONTROLADORES DE ACESSO

Existem atualmente diversos tipos de controladores de acesso com o propósito semelhante ao abordado neste trabalho. Por se tratar de um assunto recente, os controladores de acesso com detecção de temperatura ainda são pouco divulgados, desta forma não é comum encontrar na bibliografia descrições detalhadas de tais plataformas.

As diferenças encontradas entre os sistemas de controle de acesso se dão de acordo com a necessidade, complexidade, propósito, necessidade de *hardware* e na forma de identificação e liberação de acesso.

Segundo o blog VMI Security (2018), os principais sistemas de controle de acesso são através de cartões com *RFID (Radio Frequency Identification)* onde o acesso é feito através de uma rádio frequência, acesso biométrico, catracas, detector de metais, Raio-x e reconhecimento facial conforme mostrado na Figura 1.

Figura 1 - Controlador de acesso facial



Fonte: GSC Segurança Eletrônica, 2020.

Este trabalho aborda a utilização do sistema de controle de acesso com medição de temperatura corporal, visto que a febre é um dos principais sintomas do novo coronavírus.

Diversos estabelecimentos estão utilizando termômetros infravermelhos, de formatos semelhantes ao da Figura 2, para medir a temperatura manualmente de clientes e funcionários na entrada do local. Com este termômetro o risco de contágio aumenta cada vez mais.

Figura 2 - Pistola de medição de temperatura



Fonte: Portal Tocantins, 2020.

A fim de otimizar esse processo e oferecer uma maior segurança, este trabalho tem o propósito de gerenciar a entrada/saída de pessoas em indústrias, empresas, hospitais, condomínios e comércios, oferecendo uma solução sem contato, de forma automatizada e com um preço mais acessível comparado a soluções de controle de acesso por temperatura.

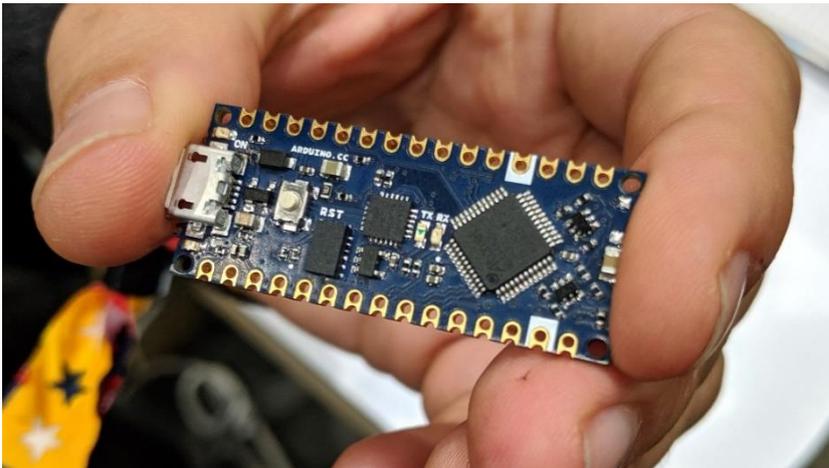
## 2.2 MICROCONTROLADORES

Microcontroladores por serem pequenos e possuírem uma melhor relação custo/benefício, estão presentes em praticamente tudo que envolve eletrônica. Segundo CUNHA (2007), a definição de sistema embarcado é “Colocar capacidade computacional dentro de um circuito integrado, equipamento ou sistema”. Um sistema completo e independente, mas que será preparado para realizar apenas uma determinada tarefa. A Figura 3 exemplifica o tamanho de um Arduino Nano.

Uma vez que um sistema embarcado é definido para ser utilizado para um fim específico, ele pode ser otimizado do tamanho necessário para a aplicação desejada e ser portátil através da replicação do software desenvolvido para utilização dele em outros hardwares.

Exemplos de sistema embarcados encontrados comumente no mercado são: o *Raspberry Pi*, *mbed*. O Arduino que será utilizado para implementação deste trabalho.

Figura 3 - Tamanho do Arduino Nano



Fonte: Hackday, 2019.

## 2.3 ARDUINO

O Arduino foi criado em 2005 com o objetivo de ser um dispositivo fácil de programar, funcional e barato. Os criadores foram 5 pesquisadores: Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino e David Mellis.

Ele possui um conceito de *hardware* livre, desta forma a contribuição de todos os usuários é importante para modificar, melhorar e personalizar o Arduino utilizando um mesmo *hardware* base.

Segundo Banzi o Arduino é uma plataforma de computação física de fonte aberta, baseada em uma placa simples de entradas/saídas, assim como um ambiente de desenvolvimento que implementa a linguagem de processamento.

Desta forma foi utilizada uma placa composta por um microcontrolador Atmel, com circuitos de entrada e saída, que possibilita a integração com computadores via USB, e programada via IDE (*Integrated Development Environment*), utilizando uma linguagem baseada em C/C++.

Existem diversos tipos de Arduino, cada um deles podendo ser utilizado para uma necessidade específica, porém todos seguem a mesma essência do primeiro desenvolvido, que seja uma plataforma de fácil utilização. O Quadro 1 mostra algumas diferenças entre os principais modelos e nos capítulos a seguir estão listados esses tipos de Arduino e uma breve descrição de cada um deles.

Quadro 1 - Especificações dos principais modelos de Arduino

	UNO	MEGA 2560	LEONARDO	DUE	ADK	NANO	PRO MINI	ESPLORA
<b>Microcontrolador</b>	ATmega328	ATmega2560	ATmega32u4	AT91SAM3X8E	ATmega2560	ATmega168 (versão 2.x) ou ATmega328 (versão 3.x)	ATmega168	ATmega32u4
<b>Portas digitais</b>	14	54	20	54	54	14	14	-
<b>Portas PWM</b>	6	15	7	12	15	6	6	-
<b>Portas analógicas</b>	6	16	12	12	16	8	8	-
<b>Memória</b>	32K (0,5K usado pelo bootloader)	256K (8K usado pelo bootloader)	32K (4K usado pelo bootloader)	512K disponível para aplicações	256K (8K usado pelo bootloader)	16K (ATmega168) ou 32K (ATmega328) (bootloader: 2K)	16K (2K usado pelo bootloader)	32K (4K usado pelo bootloader)
<b>Clock</b>	16Mhz	16Mhz	16Mhz	84Mhz	16Mhz	16Mhz	8Mhz (modelo 3.3v) ou 16Mhz (modelo 5v)	16Mhz
<b>Conexão</b>	USB	USB	Micro USB	Micro USB	USB	USB Mini-B	Serial/Módulo USB externo	Micro USB
<b>Conector para alimentação externa</b>	Sim	Sim	Sim	Sim	Sim	Não	Não	Não
<b>Tensão de operação</b>	5V	5V	5V	3.3V	5V	5V	3.3 ou 5V, dependendo do modelo	5V
<b>Corrente máxima portas E/S</b>	40mA	40mA	40mA	130mA	40mA	40mA	40mA	-
<b>Alimentação</b>	7-12Vdc	7-12Vdc	7-12Vdc	7-12Vdc	7-12Vdc	7-12Vdc	3.3-12V (modelo 3.3v) ou 5-12V (modelo 5v)	5V

Fonte: FlipFlop, 2014

### 2.3.1 Arduino UNO

Dentre os tipos de Arduino, o UNO, mostrado na Figura 4, é o mais famoso e vendido no mercado. Ele possui um bom número de portas disponíveis, e grande compatibilidade com outras placas disponíveis no mercado. Possui processador ATMEGA328, 14 portas digitais, sendo que 6 delas podem ser usadas como saídas PWM (*Pulse Width Modulation*), ou seja, moduladas por largura de pulsos, e 6 portas analógicas. Possui também 32Kb de memória. A alimentação pode ser via conexão USB ou conector para alimentação (7 a 12 Vdc) e alimentação de trabalho de 5Vdc.

Esta placa, na versão com soquete, permite a troca do chip microcontrolador ATMEGA328 facilmente em caso de dano ao microcontrolador ou se o mesmo for utilizado em projetos dedicados. Existe também a placa Arduino Uno versão SMD (*Surface mount device*), com o microcontrolador soldado na placa (ARDUINO, 2020).

Figura 4 - Arduino UNO



Fonte: Arduino, 2020

### 2.3.2 Arduino Mega 2560

O Arduino Mega 2560, mostrado na Figura 5, possui um microcontrolador ATmega2560 e 54 portas digitais, das quais 15 podem ser usadas como PWM, além de 15 portas analógicas. Possui um *clock* de 16 Mhz, conexão USB e conector para

alimentação externa e memória de 256Kb. É ideal para projetos mais elaborados que exijam grande número de entradas e saídas (ARDUINO, 2020).

Figura 5 - Arduino Mega



Fonte: Arduino, 2020

### 2.3.3 Arduino Leonardo

O Arduino Leonardo, mostrado na Figura 6, é uma placa muito parecida com o Arduino Uno, mas com Atmega32u4, possui 20 portas digitais, sendo que 7 delas podem ser usadas como PWM, e 12 como portas analógicas, possui um *clock* de 16 Mhz e conexão para alimentação 7 a 12Vdc e memória de 32Kb. Possui também um conector micro-usb para conexão ao computador.

Outra característica do Arduino Leonardo é que ele possui um chip de conexão USB integrado ao microcontrolador, o que faz com que o computador o reconheça como um teclado ou mouse, e não necessariamente como uma porta serial (ARDUINO, 2020).

Figura 6 - Arduino Leonardo



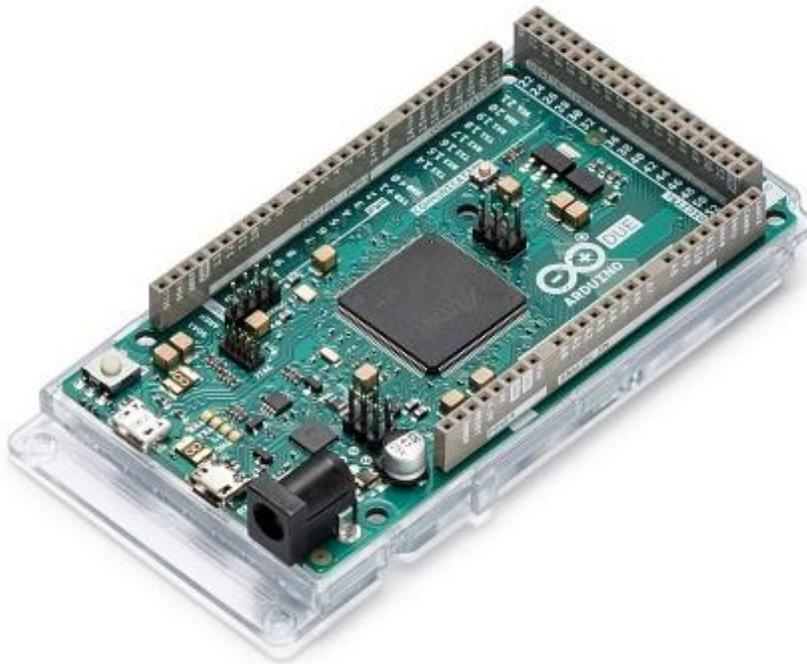
Fonte: Arduino, 2020

#### 2.3.4 Arduino Due

O Arduino Due, mostrado na Figura 7, é a placa com maior capacidade de processamento, baseada em um microcontrolador ARM de 32 bits e 512 Kb de memória. O *bootloader* já vem gravado de fábrica em uma memória ROM dedicada.

Possui 54 portas digitais, das quais 12 podem ser usadas como PWM, e 12 portas analógicas. Possui também 4 chips controladores de portas seriais, conexão USB e conector para alimentação 7 a 12Vdc (ARDUINO, 2020).

Figura 7 - Arduino Due



Fonte: Arduino, 2020

### 2.3.5 Arduino Mega ADK

O Arduino Mega ADK, mostrado na Figura 8, utiliza também o microcontrolador ATmega2560, possui uma conexão USB dedicada à ligação com dispositivos baseados em Android, possui 54 portas digitais, das quais 15 podem ser usadas como PWM, 16 portas analógicas, 4 chips dedicados à comunicação serial, *clock* de 16 Mhz e conexão ao computador via USB. Também possui conector para alimentação 12Vdc (ARDUINO, 2020).

Figura 8 - Arduino Mega ADK



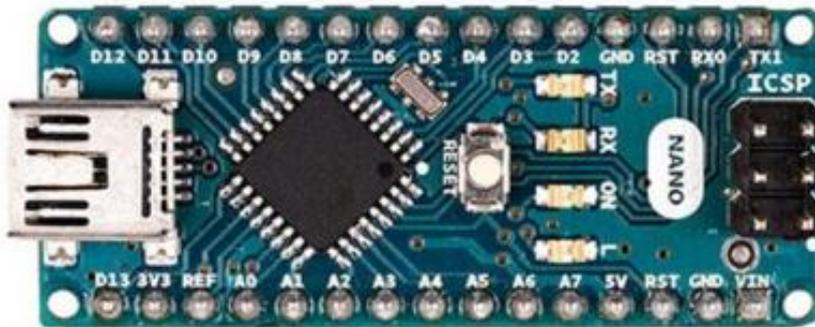
Fonte: Arduino, 2020

### 2.3.6 Arduino Nano

O Arduino Nano, mostrado na Figura 9, é uma placa compacta baseada no microcontrolador ATmega328 (para placas Arduino Nano versão 3.x), ou ATmega168 (versão 2.x). Ao contrário das outras placas, não possui conector para alimentação externa, sendo alimentada por um conector USB.

Possui versões de 16 Kb ou 32 Kb de memória. Seu tamanho reduzido (4,3 de comprimento x 1,85 de largura) faz dessa placa uma boa opção para projetos compactos que exijam atualização constante de software (ARDUINO, 2020).

Figura 9 - Arduino Nano



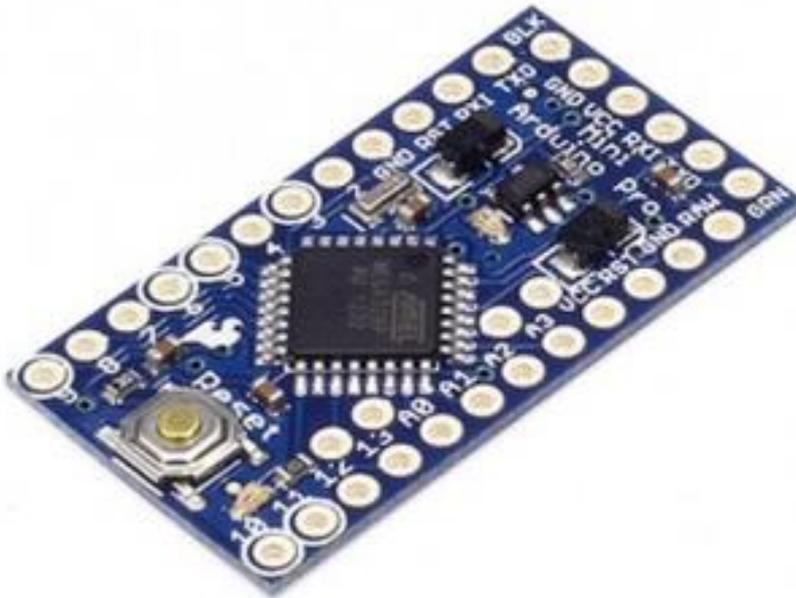
Fonte: Arduino, 2020

### 2.3.7 Arduino Pro Mini

O Arduino Pro Mini também possui uma placa compacta, ideal para projetos permanentes e que não necessitem de grande poder de processamento ou constante atualização. O microcontrolador utilizado por esta placa é o ATmega168 que tem *clocks* disponíveis de 8 Mhz e 16 Mhz.

Possui 14 portas digitais, sendo que 6 podem ser usadas como PWM, e 8 portas analógicas. Para comunicação com o computador, pode ser adquirido um módulo USB separadamente, ou utilizada uma placa Arduino para programação (ARDUINO, 2020).

Figura 10 - Arduino Pro Mini



Fonte: Arduino, 2020

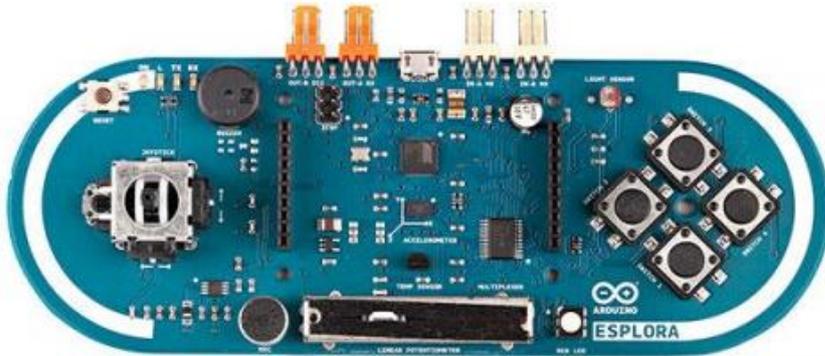
### 2.3.8 Arduino Esplora

O Arduino Esplora é uma placa diferente de todas as outras da família Arduino, principalmente por possuir diversos sensores na sua construção. Nessa placa vem embutido um *buzzer*, um *joystick*, um potenciômetro deslizante, um sensor de temperatura, um acelerômetro, um led RGB, um sensor de luz (LDR), 4 *push-buttons* e um microfone. Além de tudo isso, ainda possui um soquete para tela LCD.

É uma placa indicada para quem deseja aprender Arduino sem se preocupar muito com eletrônica, uma vez que os componentes já estão embutidos na placa. Assim o usuário pode se concentrar na parte de programação.

Baseado no Arduino Leonardo, utiliza o microcontrolador Atmega32U4, com *clock* de 16 Mhz e 32 KB de memória (4K usados pelo *bootloader*). Possui 2 conectores de 3 pinos, para conexão de módulos adicionais. A conexão com o microcomputador utiliza cabo micro-usb (ARDUINO, 2020).

Figura 11 - Arduino Esplora



Fonte: Arduino, 2020

## 2.4 LINGUAGEM DE PROGRAMAÇÃO

Segundo Gotardo (2015) uma linguagem de programação é um método padronizado que é utilizado para expressar as instruções de um programa a um computador programável. Essa linguagem deve seguir um conjunto de regras sintáticas e semânticas para definir um programa de computador (*software*).

A comunicação feita entre o *hardware* e o *software*, desenvolvida por um ser humano, normalmente é um algoritmo. Segundo Medina e Fertig (2006) um algoritmo é um procedimento passo a passo para solucionar um problema, ou ainda, uma sequência detalhada de ações que devem ser executadas para realização de alguma tarefa.

Existem diversas linguagens de programação sendo utilizadas no momento por programadores do mundo todo. Segundo o site Tiobe (2020), a linguagem C lidera a lista das mais utilizadas, sendo seguida na sequência por Java, Python e C++. O Quadro 2 mostra informações da lista de linguagens mais utilizadas no mundo no ano de 2020.

Quadro 2 - Linguagens de programação mais utilizadas

<i>Programming Language</i>	2020
<b>C</b>	1°
<b>Java</b>	2°
<b>Python</b>	3°
<b>C++</b>	4°
<b>C#</b>	5°
<b>JavaScript</b>	6°
<b>PHP</b>	7°
<b>SQL</b>	8°
<b>R</b>	9°
<b>Swift</b>	10°

Fonte: Tiobe, 2020.

### 2.4.1 Linguagem C

Neste trabalho a comunicação com o Arduino UNO será a linguagem C. Essa linguagem foi criada por Dennis M. Ritchie e Ken Thompson em 1972 e foi baseada na linguagem B (Gotardo, 2015).

Essa linguagem se tornou muito popular entre os programadores por ser flexível, portátil, poderosa e foi desenhada para ser de propósito geral, estruturada, procedural e padronizada.

Segundo Cocian (2004), a linguagem C possui diversas vantagens, uma delas é a linguagem em “alto nível” e de “baixo nível” ao mesmo tempo, permitindo assim um controle total do *hardware* e *software* por parte de quem está programando, permitindo assim com que não dependa do sistema operacional utilizado.

Outros pontos positivos que tornaram essa linguagem tão popular são mencionados abaixo:

- Portabilidade do compilador;
- Conceito de bibliotecas padronizadas;
- A quantidade e variedade de operadores poderosos;
- A sintaxe elegante;
- O fácil acesso ao hardware quando se faz necessário;
- A facilidade com que as aplicações podem ser otimizadas, tanto na codificação quanto na depuração, pelo uso de rotinas isoladas e encapsuladas.

Essa linguagem também foi escolhida pelo fato de que o autor do trabalho já possui familiaridade com ela, facilitando assim a implementação do *software*.

## 2.5 SENSORES DE TEMPERATURA

Existem diversas famílias de sensores de temperatura que são utilizados para atuação em conjunto com a plataforma Arduino. Eles são utilizados para medir a temperatura da água, para controle de umidade, temperatura corporal, e outras diversas outras aplicações imagináveis.

Aliado a eles, são utilizados outros dispositivos atuadores para realizar uma ação a fim de um resultado esperado para aplicação.

Este capítulo irá explicar um pouco sobre as famílias dos sensores de temperatura e por fim a escolha para o projeto.

### 2.5.1 Sensor de temperatura MLX90614

Com o propósito de encontrar um sensor que tenha um grau de assertividade alto, com um custo acessível e fácil de encontrar no mercado de *hardwares*, foi utilizado o sensor MLX90614 da fabricante Melexis.

Figura 12 - Sensor MLX90614



Fonte: MELEXIS, 2019.

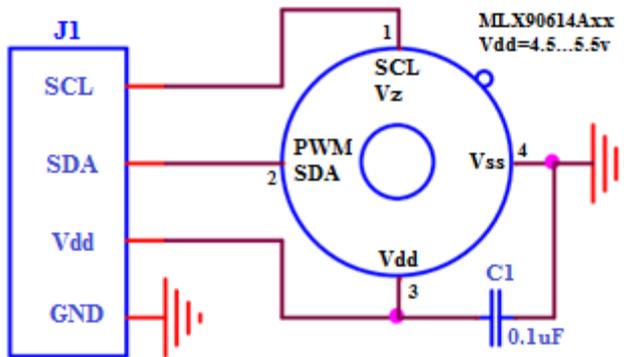
O sensor MLX90614 é um termômetro infravermelho para medições de temperatura sem contato.

Ele possui um amplificador de baixo ruído, ADC de 17 bits e um *DSP* (*Digital Signal Processor* em português Processador Digital de Sinais) capaz de fornecer uma alta precisão, chegando a precisão de 0,5 °C (0 a +50 graus) (MELEXIS, 2019).

Como a utilização do sensor será para medição de temperaturas em seres humanos, onde a temperatura média é de 36 °C e 37,5 °C (MD SAÚDE, 2020), a faixa que ele trabalha com precisão de 0,5 °C é o ideal para este projeto.

A Figura 13 apresenta o diagrama funcional do sensor utilizado:

Figura 13 - Diagrama Funcional MLX90614

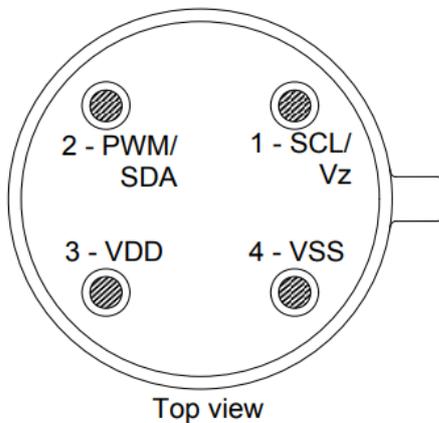


Fonte: Datasheet MLX90614, 2019.

A saída digital pode ser utilizada como modulação PWM, sendo por largura de pulso. O PWM tem por padrão 10 bits é configurado para transmitir continuamente a temperatura medida na faixa de  $-20$  a  $120$  ° C, com uma resolução de saída de  $0,14$  ° C.

A Figura 14 mostra a pinagem e o Quadro 3 apresenta a descrição dos pinos do sensor MLX90614.

Figura 14 - Pinagem do sensor MLX90614



Fonte: Datasheet MLX90614, 2019.

Quadro 3 - Descrição dos pinos MLX90614

PINOS	DESCRIÇÃO
VSS	Aterramento do componente
SCL / VZ	Entrada serial para comunicação.
PWM / SDA	Entrada/Saída digital. A temperatura está disponível neste pino modulado em PWM. Compatível com SMBus.
VDD	Tensão de alimentação

Fonte: Datasheet MLX90614, 2014.

### 2.5.2 Princípio de processamento do sinal

O sensor de temperatura MLX90614 é controlado por uma máquina de estado interna, que é responsável por controlar as medições e cálculos da temperatura do objeto a ser medido e do ambiente ao redor deste objeto. Após isso faz o processamento das temperaturas enviando os dados colhidos através da saída PWM ou pela interface SMBus.

Ele possui 2 sensores infravermelhos para detecção de temperatura, conforme exemplificado na Figura 15. As saídas dos sensores são amplificadas por um amplificador *chopper* de baixo ruído e baixo deslocamento com ganho programável, convertido por um Modulador Sigma-Delta para um único fluxo de bits e envia os dados para o *DSP (Digital Signal Processor)* fazer o processamento.

Figura 15 - Zonas do IR



Fonte: Datasheet MLX90614

O sinal é tratado por filtros passa baixa para redução da largura de banda do sinal de entrada, para atingir o desempenho de ruído desejado e a taxa de atualização. A saída do filtro é o resultado da medição e está disponível na memória RAM do dispositivo.

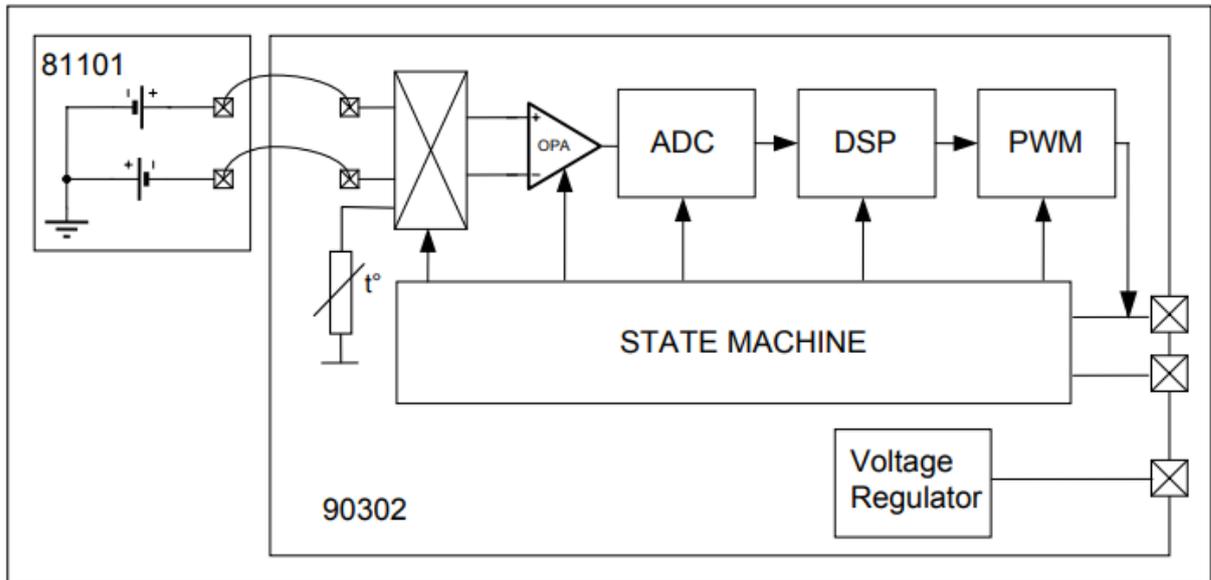
Ele possui 3 células diferentes: Um para o sensor de temperatura integrado e outros 2 para os sensores IR.

Com base nos resultados das medições, a temperatura ambiente correspondente e a temperatura do objeto são calculadas. Ambas as temperaturas calculadas têm uma resolução de 0,01 °C.

Os dados são carregados nos registros da máquina de estado do PWM, representando os dados medidos (Melexis, 2007).

A Figura 16 mostra o diagrama de blocos de como funciona a aquisição dos dados e o processamento do sensor de temperatura.

Figura 16 - Diagrama de blocos MLX90614

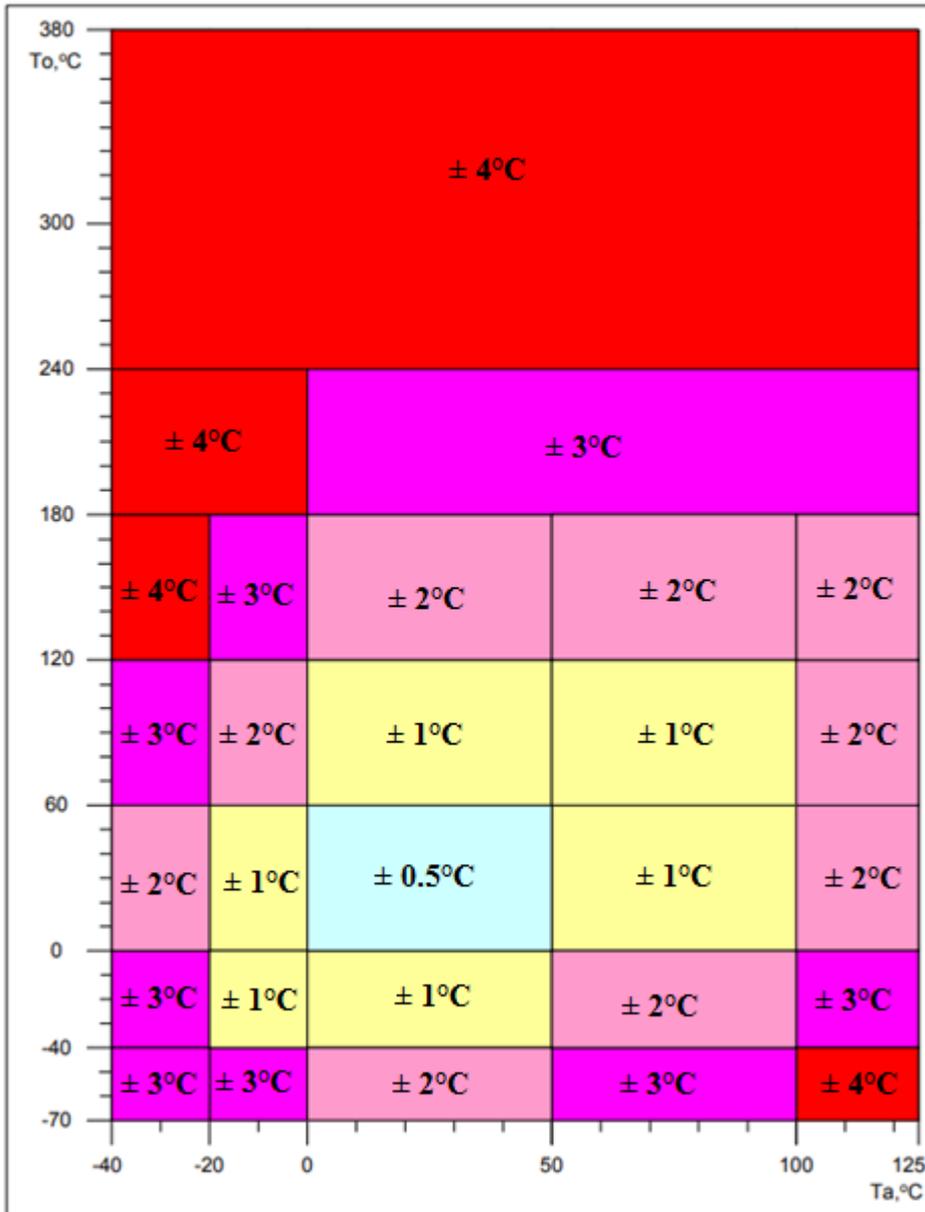


Fonte: Datasheet MLX90614

O sensor MLX90614 possui uma precisão de temperatura variando de acordo com a temperatura do ambiente.

A Figura 17 mostra que precisão de  $0,5^{\circ}\text{C}$  se aplica quando a temperatura do ambiente ( $T_a$ ) está entre  $0$  e  $50^{\circ}\text{C}$  e a temperatura do objeto ( $T_o$ ) a ser medido está entre  $0$  e  $60^{\circ}\text{C}$ , que será a temperatura na qual este projeto é proposto.

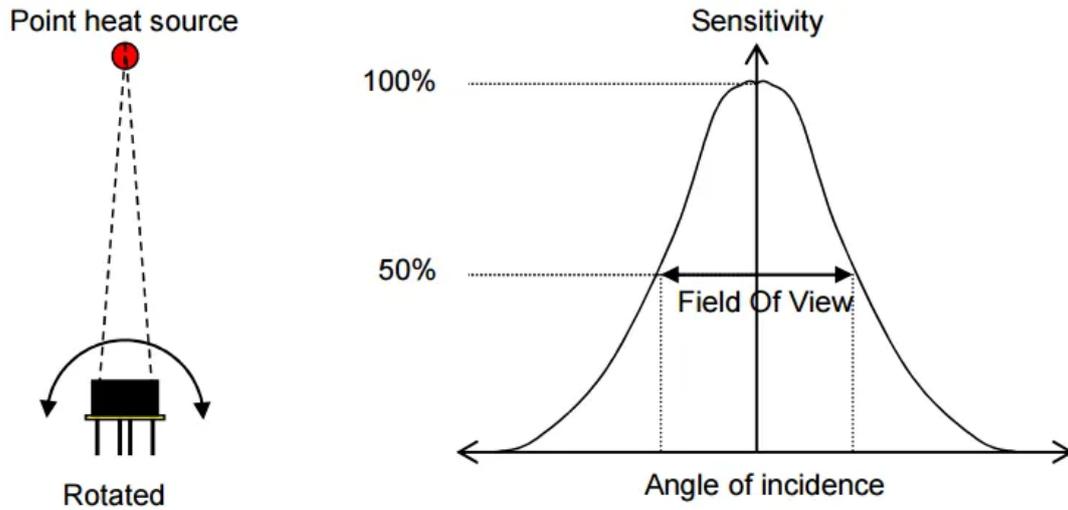
Figura 17 - Precisão em relação ao ambiente



Fonte: Datasheet MLX90614

É importante saber o campo de visão para melhor direcionar o posicionamento para medida. A Figura 18 mostra que o pico está ao centro do sensor e quanto maior a dispersão, menor o grau de sensibilidade do sensor.

Figura 18 - Ponto de visão



Fonte: Do bit ao Byte, 2009.

## 2.6 SENSOR DE PRESENÇA HC-SR505

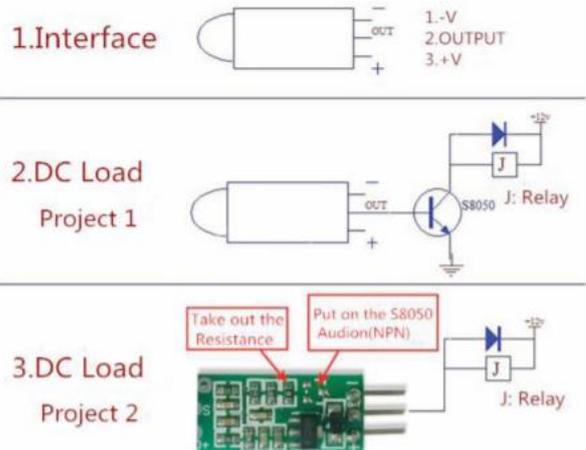
O sensor de movimento HC-SR505 utiliza o infravermelho para detectar um corpo à sua frente. Por conta da sua alta sensibilidade se torna um equipamento de alta precisão e confiabilidade. Pelo seu tamanho ser menor e seu baixo consumo, é amplamente utilizado em projetos alimentados a bateria.

As especificações do sensor HC-SR505 podem ser verificadas na Figura 19.

Figura 19 - Especificações HC-SR505

### Product Specification

Operating voltage range: DC4.5-20V  
 Quiescent Current: <60uA  
 Level output: High 3.3V / Low 0V  
 Trigger: reusable trigger (default)  
 Delay Time: The default 8S + -30%  
 Board Dimensions: 10 \* 23mm  
 Induction angle: <100 degrees cone angle  
 Sensing distance: 3 meters  
 Working temperature: -20 to +80 degrees  
 Sensor Lens Dimensions: Diameter 10mm



Fonte: Datasheet HC-SR505, 2021.

### 3 DESENVOLVIMENTO

Essa seção aborda todas as etapas de desenvolvimento, será apresentada a ideia inicial do projeto para uma automação de controle de acesso por temperatura. Detalhando a construção do protótipo, escrita do código de programação, os testes e resultados obtidos.

Conforme o objetivo proposto na seção 1.3 deste trabalho, foi desenvolvido um sistema capaz de controlar o acesso de pessoas com temperatura alta em ambientes comerciais, evitando assim o contágio pelo novo coronavírus.

Foi utilizado o microcontrolador Arduino Uno, com linguagem de programação C, com sensor de presença PIR e sensor de temperatura MLX90614.

A opção pelo uso do microcontrolador Arduino Uno trouxe as seguintes vantagens: baixo custo, simples plataforma de criação e melhoria de produto amplamente utilizada pelos universitários, facilidade de compra em mercado nacional e internacional, entradas e saída digitais e analógicas.

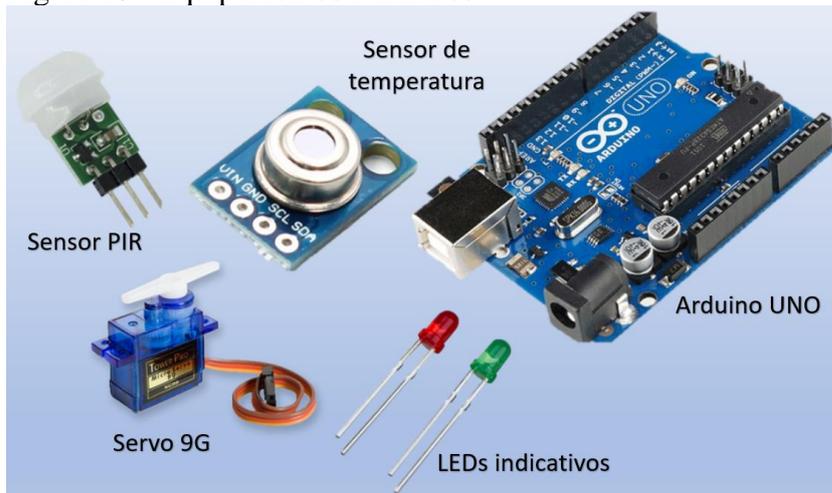
O projeto foi desenvolvido e pensado como protótipo para ser utilizado em ambientes comerciais, tendo isso em vista a simplicidade na utilização se faz algo fundamental.

#### 3.1 EQUIPAMENTOS UTILIZADOS

Baseado na fundamentação teórica descrita nos capítulos anteriores, a Figura 20 e a lista abaixo mostram os componentes eletrônicos escolhidos para desenvolvimento desse projeto:

- Plataforma Arduino UNO;
- Sensor de temperatura MLX 90614;
- Placa circuito Protoboard;
- Servo Motor 9G;
- Leds indicadores;
- Sensor de presença PIR;
- Jumpers de comunicação;

Figura 20 - Equipamentos utilizados



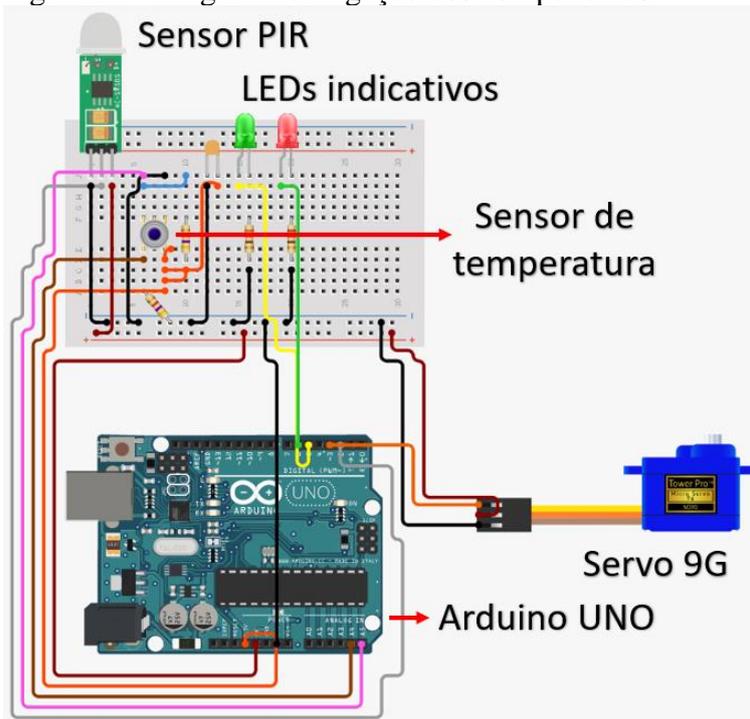
Fonte: Autor, 2021.

### 3.2 PROTÓTIPO

Para desenvolvimento do circuito foi utilizada a plataforma online circuit.io, uma plataforma gratuita para modelagem de circuitos utilizando o Arduino Uno como base. A opção de escolha desse site foi a facilidade de ser um software gratuito e amigável, assim como o conhecimento da plataforma.

Através do site é possível informar os componentes que irão compor o projeto e com ele iniciar o protótipo deste trabalho. A Figura 21 mostra o diagrama de ligação dos componentes deste projeto.

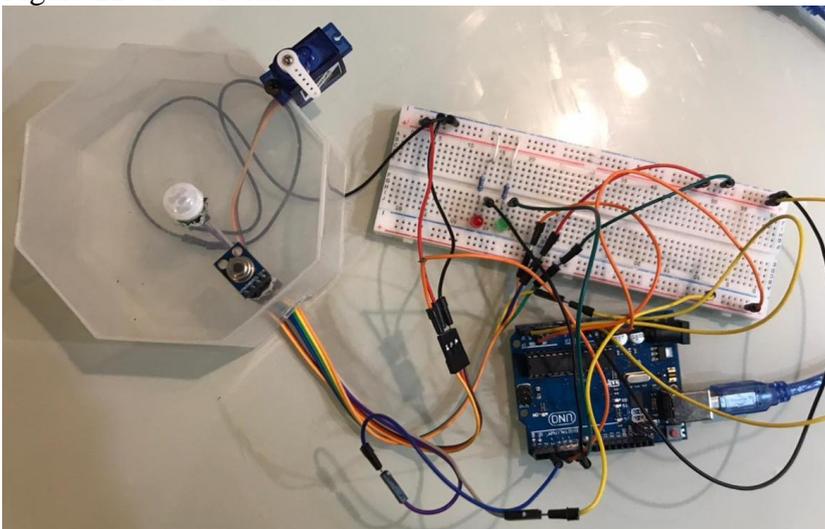
Figura 21 - Diagrama de ligação dos componentes



Fonte: Circuit.io, 2021.

Durante o desenvolvimento do trabalho, foi criada uma plataforma de testes para validar os dados e informações obtidos do sensor de temperatura, a Figura 22 mostra a plataforma que foi utilizada para validação e conclusão do código fonte deste trabalho.

Figura 22 - Plataforma de testes



Fonte: Autor, 2021.

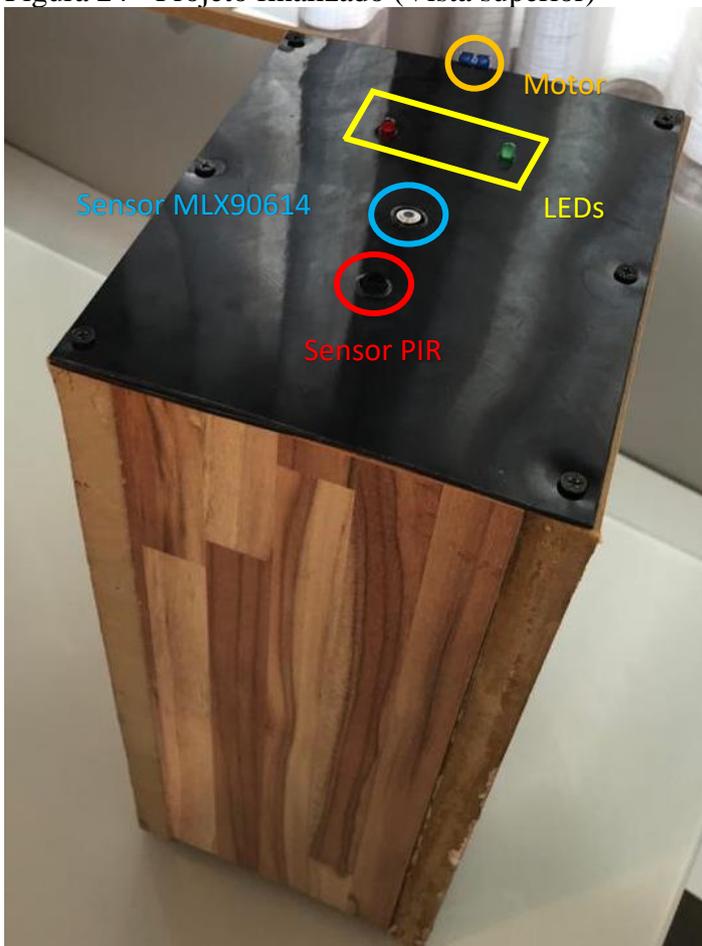
A Figura 23 e a Figura 24 mostram como o projeto ficou após os testes e validações dos dados obtidos do sensor de temperatura.

Figura 23 - Projeto finalizado (Vista posterior)



Fonte: Autor, 2021.

Figura 24 - Projeto finalizado (Vista superior)



Fonte: Autor, 2021.

### 3.3 LINGUAGEM DE PROGRAMAÇÃO

Para o desenvolvimento da linguagem de programação no Arduino Uno, foi utilizada a própria IDE do Arduino versão do software 1.8.13, conforme a Figura 25, que estava conectada a porta COM do computador.

Figura 25 - Versão do software Arduino



Fonte: Arduino, 2021.

Na linguagem de programação do microcontrolador foram utilizadas algumas bibliotecas, mencionadas abaixo:

- Arduino.h;
- LED.h;
- Servo.h;
- Wire.h;
- Adafruit\_MLX90614.h;

Além das definições das bibliotecas utilizadas, a definição das portas digitais foi de acordo com a Tabela 1 abaixo:

Tabela 1 - Portas digitais

Porta	Digital
2	PIR
3	SERVO
5	LED_VERDE
6	LED_VERMELHO

Fonte: Autor, 2021.

A lógica do sistema foi feita com uma programação em loop aguardando o sinal do sensor PIR, assim que o sensor de presença detecta movimento o sensor de temperatura começa a capturar as temperaturas do objeto que está próximo a ele.

Para uma temperatura mais próxima do real, foi utilizada uma média com mil capturas do objeto durante um tempo de 1 segundo, após esse tempo ele calcula a média e, se o valor for maior que 34 °C e menor de 37,5 °C do objeto, ele libera o acesso e acende um led indicador verde, aguarda o tempo de 10 segundos para voltar novamente a barreira a posição inicial. Caso o valor esteja fora desse intervalo o acesso não é liberado e acende um led vermelho e a barreira permanece fechada para não liberar o acesso.

Durante o desenvolvimento do projeto, foi notado que a precisão do sensor MLX90614 não estava conforme o especificado, desta forma foi utilizada a câmera térmica Intelbras DMT 7200 FT. A Figura 26 mostra uma foto do sensor com suas especificações de temperatura, isso foi necessário para criar uma curva de aprendizagem do sensor.

Figura 26 - DMT 7200 FT



#### Medição de temperatura

Faixa de medição de temperatura	+30°C ~ + 45°C
Varição de medição	No máximo $\pm 0,5^{\circ}\text{C}$ (Temperatura de operação entre 15°C a 35°C)
Modo de medição de temperatura	Centro / temperatura máxima
Distância de medição de temperatura	Entre 0,5 e 1,5 metros

Fonte: Datasheet DMT 7200 FT, 2021.

Para uma melhor curva de aprendizagem, foram retiradas 20 amostras de temperatura da testa e do punho e feito a média dos valores obtidos.

Conforme orientação no datasheet do sensor de temperatura MLX90614, as medidas foram retiradas até a altura de 60 cm do sensor, para que a variação de medição seja de no máximo  $\pm 0,5^{\circ}\text{C}$ . Para verificar a temperatura com o DMT 7200 FT, a distância de medição foi de 1 metro, tendo assim a mesma variação de medição.

A Tabela 2 mostra os valores do sensor e a média que foi feita para corrigi-lo.

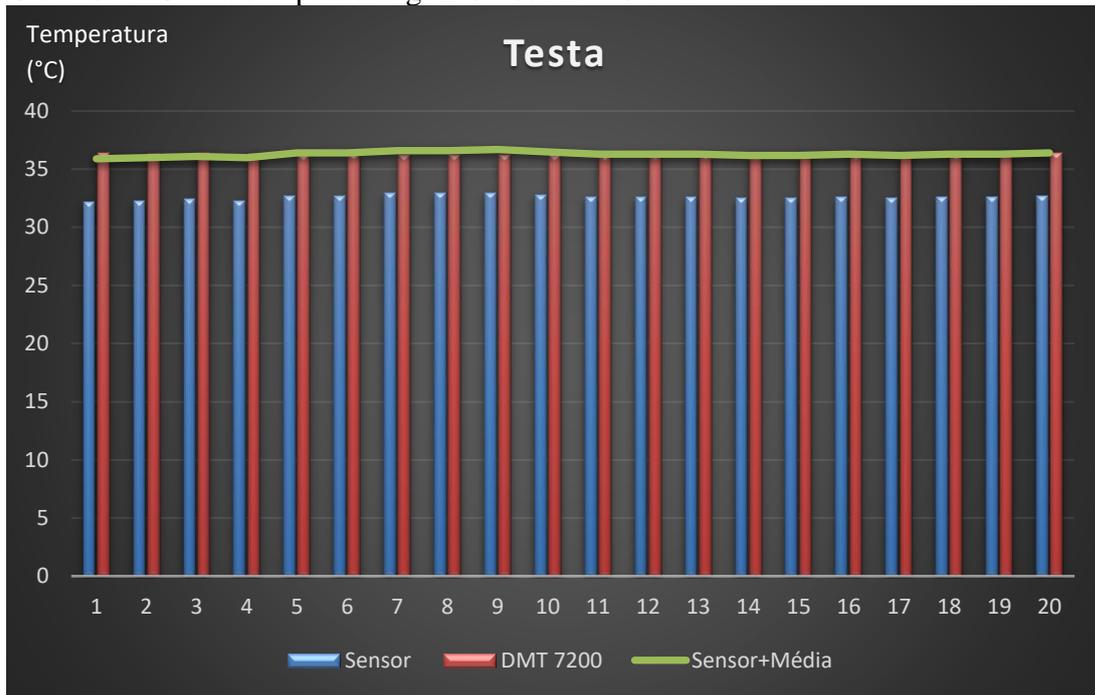
Tabela 2 - Curva de aprendizagem do sensor

Amostra	Testa		Punho	
	Sensor (°C)	DMT 7200 FT (°C)	Sensor (°C)	DMT 7200 FT (°C)
1	32,2	36,4	31,9	36,4
2	32,3	36,3	32,3	36,4
3	32,4	36,4	31,9	36,5
4	32,3	36,3	32,6	36,6
5	32,7	36,2	31,9	36,5
6	32,7	36,3	30,9	36,4
7	32,9	36,2	32,4	36,4
8	32,9	36,2	31,9	36,3
9	33,0	36,2	32,1	36,4
10	32,8	36,2	32,2	36,3
11	32,6	36,2	32,2	36,3
12	32,6	36,3	32,9	36,3
13	32,6	36,3	31,9	36,3
14	32,5	36,3	31,6	36,4
15	32,5	36,4	32,3	36,3
16	32,6	36,3	32,3	36,3
17	32,5	36,4	32,6	36,3
18	32,6	36,3	32,0	36,3
19	32,6	36,4	33,0	36,3
<b>20</b>	32,7	36,4	32,1	36,3
<b>Média</b>		<b>3,7</b>		<b>4,2</b>

Fonte: Autor, 2021

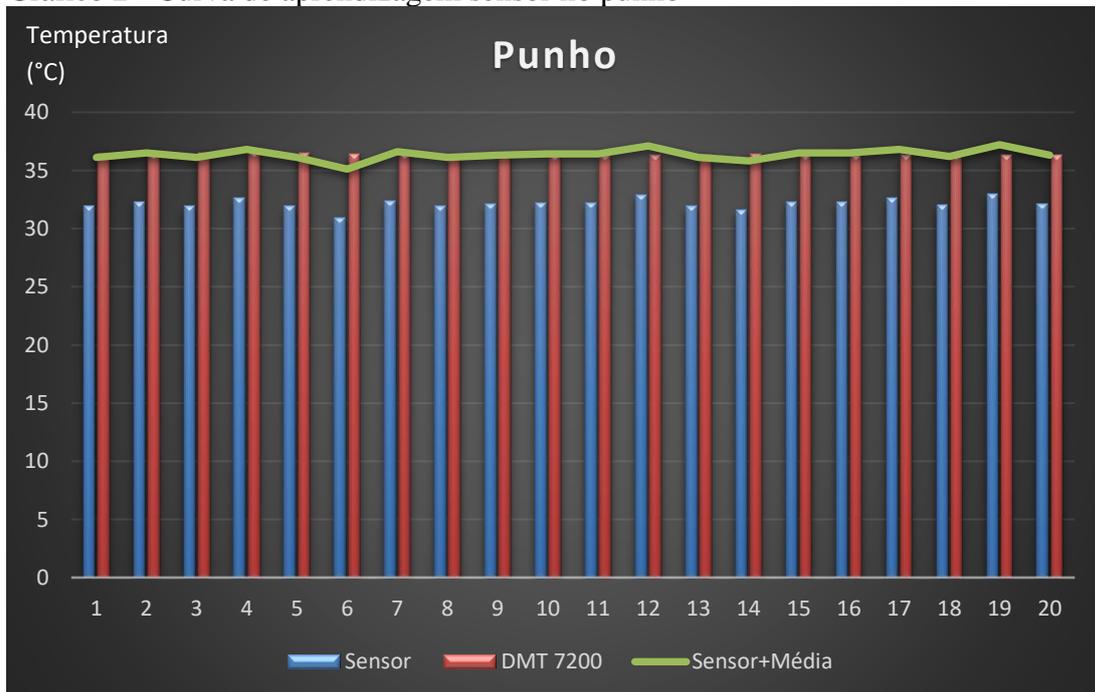
O Gráfico 1 mostra a diferença da medida do sensor MLX90614 para a câmera térmica DMT 7200 FT em comparação a medida feita na testa, é possível notar a diferença média de 3,7 °C. Já o Gráfico 2 mostra a comparação com o punho onde a diferença média de 4,2 °C. Tendo esses dois valores de média, foi feito um ajuste na medida do sensor de +4 °C, para mostrar o valor mais próximo do real.

Gráfico 1 - Curva de aprendizagem sensor na testa



Fonte: Autor, 2021.

Gráfico 2 - Curva de aprendizagem sensor no punho



Fonte: Autor, 2021.

É possível afirmar que através da temperatura capturada no sensor e com o ajuste feito no código fonte, o controlador de acesso se torna uma ferramenta de controle de pessoas que pode ser aplicada em estabelecimentos com um baixo custo de investimento.

## 4 CONCLUSÃO

A busca por formas de reduzir a crescente curva da doença Covid-19, fez com que o uso da tecnologia fosse amplamente utilizado. Dito isto, o desenvolvimento de um controlador de acesso automático por medição de temperatura corporal se tornou algo fundamental, tendo em vista a obrigatoriedade da medição de cliente e funcionários antes de adentrarem a estabelecimentos comerciais.

Este trabalho de conclusão de curso teve como objetivo principal a criação de um controlador de acesso simples e automático para evitar o contágio em ambientes comerciais, tendo sido alcançado com sucesso com o estudo dos sintomas da doença, sensores de temperatura, microcontroladores e sensor de presença.

Para chegar no sucesso do objetivo principal deste trabalho, foram realizadas as etapas de um estudo sobre o sensor de temperatura MLX90614, desenvolvimento de um software de comunicação entre os dispositivos de controle de acesso e medição de temperatura utilizando o microcontrolador Arduino e o desenvolvimento de um protótipo automatizado para liberar ou não o acesso ao interior do estabelecimento.

Verificou-se durante o andamento deste trabalho de conclusão de curso que o sensor escolhido não entregou a precisão necessária, tendo isso em vista foi necessário criar uma curva de aprendizado do sensor para reduzir a diferença entre a temperatura real.

No entanto, pode-se dizer que os desafios e dificuldades encontradas principalmente com o sensor de temperatura levaram a um vasto conhecimento sobre a doença e eletrônicos utilizados, ampliando o crescimento pessoal e profissional. Com o intuito de continuação deste projeto, sugere-se a continuação de melhorias como:

- Utilização de outros sensores de presença para detecção de passagem para fechamento automático da barreira;
- Buscar um sensor de temperatura infravermelho com uma maior precisão;
- Implementação de um sistema para liberar álcool em gel após a medida da temperatura;
- Incluir sensores para controlar também a quantidade de pessoas no interior do estabelecimento.

Com o estudo e o protótipo finalizado, o benefício para os pequenos comerciantes, que estão passando por dificuldades financeiras por conta da redução de clientes, este projeto poderá auxiliar quanto aos cuidados sanitários exigidos pela OMS.

## REFERÊNCIAS

- ARDUINO.CC. **Boards & Modules**. 2020. Disponível em: <<https://store.arduino.cc/usa/arduino/boards-modules>>. Acesso em: 16 out. 2020.
- ATHOS ELECTRONICS. **Tipos de Arduino –Qual Arduino escolher**. 2020. Disponível em: <<https://athoselectronics.com/qual-arduino-comprar/>>. Acesso em: 12 out. 2020
- BABOS, Flávio. **Sensor de Temperatura Arduino: LM35, DHT11, DS18B20 e TMP36**. 2020. Disponível em: <<https://flaviobabos.com.br/sensor-de-temperatura-arduino/>>. Acesso em: 10 out. 2020.
- BANZI, Massimo. **Primeiros passos com o Arduino**. São Paulo: Ed. Novatec, 2011. Disponível em: <<https://s3.novatec.com.br/capitulos/capitulo-9788575222904.pdf>>. Acesso em: 20 out. 2020.
- COCIAN, Luis Fernando Espinosa. **Manual da linguagem C**. Canoas: Ed. ULBRA, 2004.
- CUNHA, Alessandro. **Sistemas Embarcados, Revista Saber Eletrônica**. 414. Ed. Saber, BRASIL, 2007. Disponível em: <[http://files.comunidades.net/mutcom/ARTIGO\\_SIST\\_EMB.pdf](http://files.comunidades.net/mutcom/ARTIGO_SIST_EMB.pdf)>. Acesso em: 05 out. 2020.
- DO BIT AO BYTE. **Sensor de temperatura IR MLX90614**. 2009. Disponível em: <<https://www.dobitaobyte.com.br/sensor-de-temperatura-ir-mlx90614/>>. Acesso em: 24 out. 2020.
- FLIPEFLOP. **O que é Arduino**. 2014. Disponível em: <<https://www.filipeflop.com/blog/o-que-e-arduino/>>. Acesso em: 10 out. 2020.
- FLIPEFLOP. **Qual Arduino comprar**. 2014. Disponível em: <<https://www.filipeflop.com/blog/tipos-de-arduino-qual-comprar/>>. Acesso em: 12 out. 2020.
- MD.SAÚDE. **Febre – o que é, causas, sintomas e tratamento**. 2020. Disponível em: <<https://www.mdsaude.com/doencas-infecciosas/febre/>>. Acesso em: 14 out. 2020.
- MEDINA, Marco; FERTIG, Cristina. **Algoritmo e programação: Teoria e Prática**. São Paulo: Ed. Novatec, 2006.
- MELEXIS. **Digital plug & play infrared thermometer in a TO-can**. 2019. Disponível em: <<https://www.melexis.com/en/product/MLX90614/Digital-Plug-Play-Infrared-Thermometer-TO-Can>>. Acesso em: 20 out. 2020.
- MINISTÉRIO DA SAÚDE. **Plano de Contingência Nacional para Infecção Humana pelo novo Coronavírus COVID-19**, 2020. Disponível em: <<https://portalarquivos2.saude.gov.br/images/pdf/2020/fevereiro/13/plano-contingencia-coronavirus-COVID19.pdf>>. Acesso em 05 de out. 2020
- PEIXOTO, Thiago Moratori. **Sistema de Controle de Acesso Utilizando Dispositivos Embarcados**. 2013. 45 f. Monografia (Bacharelado em Ciência da Computação) – Instituto de Ciências Exatas, Universidade Federal de Juiz de Fora, Juiz de Fora, 2013. Disponível em: <<http://monografias.nrc.ice.ufjf.br/tcc-web/exibePdf?id=115>>. Acesso em: 05 out. 2020.

RAUEN, Fábio José. **Roteiros de iniciação científica: os primeiros passos da pesquisa científica desde a concepção até a produção e a apresentação**. Palhoça: Ed. Unisul, 2015.

TIOBE. **November Headline: Python is unstoppable and surpasses Java**. 2020. Disponível em: <<https://www.tiobe.com/tiobe-index/>>. Acesso em: 30 out. 2020.

VMI SECURITY. **Conheça os 6 principais tipos de controle de acesso nas empresas**. 2018. Disponível em: <<https://blog.vmisecurity.com/controle-de-acesso-nas-empresas/>>. Acesso em: 05 out. 2020.

WORLD HEALTH ORGANIZATION (WHO), 2020. Disponível em: <<https://www.who.int/emergencies/diseases/novel-coronavirus-2019>>. Acesso em 05 de out. 2020

WORLD HEALTH ORGANIZATION (WHO), *Coronavirus disease (COVID-19) advice for the public: When and how to use masks*, 2020. Disponível em: <<https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public/when-and-how-to-use-masks>>. Acesso em 05 de out. 2020

**ANEXOS**

## ANEXO A – Código de programação

```

TCC_Final_20-05-2021$ LED.cpp LED.h Switchable.cpp Switchable.h
//Incluir bibliotecas
//#include "Arduino.h" //Biblioteca do Arduino
#include "LED.h" //Biblioteca dos LED
#include "Servo.h" //Biblioteca do Servo 9G
#include "Wire.h" //Biblioteca Wire
#include "Adafruit_MLX90614.h" //Biblioteca do sensor MLX90614

//Definindo as conexões digitais
#define LED_VERDE 5 //LED Verde - Aprovado para entrar no ambiente
#define LED_VERMELHO 6 //LED Vermelho - Reprovado para entrar no ambiente
#define PIR 2 //Pino de dados do Sensor PIR (Central)
#define SERVO 3 //Pino de dados do Servo Motor 9g (Laranja)

//Variáveis globais e definições
const int servo9gPosInic = 8; //Posição inicial do servo motor 8°
const int servo9gPosFinal = 110; //Posição final 110°
float media = 0; //Variavel para fazer a media de temperatura
float temp = 0; //Variavel para armazenar a temperatura

Adafruit_MLX90614 mlx = Adafruit_MLX90614();
Servo servo9g;

void setup()
{
  Serial.begin(9600);
  while (!Serial) ; // Espera a conexão da porta USB

  Serial.println("*****");
  Serial.println("| TCC Henrique Hoffmann - Controlador de Acesso térmico |");
  Serial.println("*****");

  mlx.begin();

```

```

Serial.println("\nPosicione sua mao em frente ao sensor\n");

pinMode(PIR, INPUT);           //Define o Pino onde está o sensor PIR como entrada
}

//Loop do sistema
void loop() {
  servo9g.attach(SERVO);
  servo9g.write(servo9gPosInic); //Mantém a cancela fechada

  while (digitalRead(PIR) == HIGH) { //Se a leitura do sensor PIR for positiva, segue para a medida da temperatura
    media = 0;
    temp = 0;
    for (int i = 0; i < 1000; i++) //Calculo de media até 1000 leituras
    {
      temp = temp + mlx.readObjectTempC(); //Soma o valor da temperatura atual com a anterior
      delayMicroseconds(100); //Aguarda 100 microsegundos
    }
    media = 4.015 + (temp / (1000)); //Calcula o valor da media

    Serial.print ("Valor da Media: "); //Exibe o valor da media no monitor serial
    Serial.println (media, 1); //Imprime o valor da media com uma casa decimal
    delay (1000); //Aguarda 1000 milisegundos

    if (media >= 34 && media <= 37.5) { //Enviando o comando para o servo motor ficar conectado para receber os comandos
      servo9g.attach(SERVO);
      Serial.println("\nBem vindo\n");
      digitalWrite (5, HIGH); //Acende o led verde
      delay(15); //Aguarda 15 milisegundos
      for (int i = servo9gPosInic; i <= servo9gPosFinal; i += 1) { //Suavizando a descida da cancela
        servo9g.write(i); //Colocando a posição do servo motor para a posição final
        delay(5);
      }
    }
  }
}

```

```

    }
    servo9g.detach(); //Desconectando o servo motor para conservar energia
    digitalWrite (5, LOW); //Apaga o led verde
    Serial.println("\nPosicione sua mao em frente ao sensor\n");
}
else {
    Serial.println("\nProcure o ambulatório\n");
    digitalWrite(6, HIGH); //Liga o Led Vermelho
    delay(3000); //Espera 3000 milisegundos ou 3 segundos
    digitalWrite(6, LOW); //Desliga o Led Vermelho;
    Serial.println("\nPosicione sua mao em frente ao sensor\n");
}
}
}
}

```

TCC\_Final\_20-05-2021 § LED.cpp LED.h Switchable.cpp Switchable.h

```
#include "LED.h"
```

```
LED::LED(const int pin) : Switchable(pin){
```

```
}
```

TCC\_Final\_20-05-2021 § LED.cpp LED.h Switchable.cpp Switchable.h

```
#ifndef RGBLED_H
```

```
#define RGBLED_H
```

```
#include "Switchable.h"
```

```
class LED : public Switchable
```

```
{
```

```
public:
```

```
    LED(const int pin);
```

```
};
```

```
#endif // __LED_H
```

TCC\_Final\_20-05-2021 § LED.cpp LED.h Switchable.cpp § Switchable.h

```
#include "Switchable.h"
#include <Arduino.h>

Switchable::Switchable(const int pin) : m_pin(pin){
    pinMode(m_pin, OUTPUT);
    off();
}

void Switchable::on(){
    digitalWrite(m_pin, 1); //high
    m_state = true;
}

void Switchable::off(){
    digitalWrite(m_pin, 0); //low
    m_state = false;
}

void Switchable::toggle(){
    digitalWrite(m_pin, !m_state); //low
    m_state = !m_state;
}

void Switchable::dim(int dimVal){
    analogWrite(m_pin, dimVal);
}

bool Switchable::getState(){
    return m_state;
}

void Switchable::setState(bool state){
    digitalWrite(m_pin, state);
    m_state = state;
}
```

TCC\_Final\_20-05-2021 § LED.cpp LED.h Switchable.cpp § Switchable.h §

```
#ifndef _SWITCHABLE_H_
#define _SWITCHABLE_H_

class Switchable
{
public:
    Switchable(const int pin);
    void on();
    void off();

    void toggle();

    void dim(int dimVal);

    bool getState();

    void setState(bool state);

private:
    const int m_pin; //output pin
    bool m_state; //current state
};

#endif // _SWITCHABLE_H_
```