



UNIVERSIDADE AGOSTINHO NETO
FACULDADE DE ENGENHARIA
DEPARTAMENTO DE ELECTRÓNICA E ELECTROTECNIA
(D.E.E)

TRABALHO DE FIM DE CURSO DE LICENCIATURA EM ENGENHARIA
ELECTRÓNICA E TELECOMUNICAÇÕES

TEMA:
SISTEMA DE INCUBAÇÃO ARTIFICIAL DE OVOS DE GALINHA MONITORADO
REMOTAMENTE

AUTOR:
Lupini Zé Brunel David - 99995

LUANDA, 2023



UNIVERSIDADE AGOSTINHO NETO
FACULDADE DE ENGENHARIA
DEPARTAMENTO DE ELECTRÓNICA E ELECTROTECNIA
(D.E.E)

TEMA:

**SISTEMA DE INCUBAÇÃO ARTIFICIAL DE OVOS DE GALINHA MONITORADO
REMOTAMENTE**

Trabalho de Fim de Curso apresentado à
Faculdade de Engenharia da Universidade
Agostinho Neto, como parte dos requisitos para
obtenção do título de licenciado em Engenharia
Electrónica e Telecomunicações.

Orientador: MSc. Carlos Lopes.

LUANDA, 2023

DEDICATÓRIA

Dedico este trabalho ao meu querido pai Mpindi David e a minha querida mãe Mafuta Sidamie que sempre batalharam para que não me faltasse nada apesar deles não estarem mais presente, que Deus cuide deles na sua santa glória.

AGRADECIMENTOS

Agradeço primeiramente à Deus, que me deu energia e fôlego de graça para concluir todo esse trabalho.

Agradeço aos meus amigos que sempre me apoiaram, os meus irmãos e em especial ao Muená Cabral e Iteta Cabral pela motivação e força que me foram dados.

Ao meu orientador MSc Carlos Lopes pela disposição e paciência.

Aos meus colegas, que fizeram parte do meu percurso estudantil pelo incentivo e apoio durante a longa jornada. E aos demais, que de alguma forma contribuíram directa ou indirectamente para a elaboração deste trabalho, o meu muito obrigado.

RESUMO

As automações electrónicas estão directamente ligadas com a evolução do hardware e do software. Cada vez mais, surgem pequenos sistemas com hardware e software dedicado para alguma actividade específica. Diante disto, esse estudo visa criar um sistema embarcado, com o hardware constituído de uma plataforma microcontrolada com sensor de temperatura, humidade, ultrassónico e um motor para a rolagem dos ovos. O software necessita de comunicação com a internet e é instalado no telefone, servindo como monitoramento e/ou controlo das variáveis de ambiente da incubadora. A meta é que pequenos produtores de aves possam acompanhar o progresso da chocagem, além de receberem avisos sobre valores fora da faixa ideal, fazendo com que a taxa de chocagem seja a maior possível [5].

Palavras-chave: Chocagem; Tecnologia; Celular; Monitoramento; Ovos.

ABSTRACT

Electronic automations are directly linked to the evolution of hardware and software. Increasingly, small systems appear with hardware and software dedicated to some specific activity. Given this, this study aims to create an embedded system, with the hardware constituted of a microcontrolled platform with sensors of temperature, humidity, son and ultrasonic and a motor for the rolling of the eggs. The software requires communication with the internet and is installed on the cell phone, serving as monitoring and / or control of the incubator's environment variables. The goal is that small poultry producers can monitor the hatching progress, in addition to receiving warnings about values outside the ideal range, making the hatching rate as high as possible [5].

Keywords: Hatcher; Technology; Cell Phone; Monitoring; Eggs.

LISTA DE TABELAS

Tabela 2.1: Escolha entre incubação por choco ou por incubadora .	10
Tabela 2.2: Diagnóstico de problemas de nascimento	13
Tabela 4.1: Revisão de custos	44

LISTA DE ILUSTRAÇÕES

Figura 1.1: Diagrama esquemático do projecto.	3
Figura 3.1: Pinagem de um ESP32.	15
Figura 3.2: Declarações de Pinos.	16
Figura 3.3: Pinos ADC.	16
Figura 3.4: Conexão do esp32 com display OLED.	18
Figura 3.5: Pinagem do módulo DTH11.	18
Figura 3.6: Lâmpada incandescente	19
Figura 3.7: Período de trabalho da lâmpada incandescente.	20
Figura 3.8: Espectro de audição da galinha.	21
Figura 3.9: Sinais do processo de medida com HC-SR04.	22
Figura 3.10: Pinagem do sensor ultrassônico	23
Figura 3.11: Sensor de humidade do solo higrómetro.	23
Figura 3.12: Motor 49TYJ.	24
Figura 3.13: Bomba RS385.	25
Figura 3.14: Micro ventilador 30x30x10mm 12v-3010.	26
Figura 3.15: Buzzer tipo activo.	26
Figura 3.16: Ambiente de Desenvolvimento Integrado.	27
Figura 3.17: Janela de gerenciamento de placas.	28
Figura 3.18: Interface de designer do Mit app inventor	29
Figura 3.19: Componentes de interface do Mit app inventor.	30
Figura 3.20: Firebase de Google.	31
Figura 4.1: Protótipo Concluído.	33
Figura 4.2: Fluxograma do sistema.	34
Figura 4.3: Esp32 com Sensor DHT11 e Display Oled	43

LISTA DE ABREVIATURAS E SIGLAS

CI - Circuito Integrado

DC – Corrente Direita (Contínua)

GHz - Giga Hertz

GND - Pino terra ou 0 V

IAMOC - Incubadora Artificial Monitorado Via Celular

Vin - Entrada de Tensão

kHz - kilo Hertz

LCD - liquid crystal display

LED- Light Emitting Diode

MHz - Mega Hertz

mA – Mili Ampére

OLED - Organic Ligh Emitting Diode

PCI- Placa de Circuito impresso

PWM- Pulse Width Modulation

RAM-Random Access Memory

RX - Receptor

SMS- Short Message Service

TX - Transmissor

3V3 - Alimentação de 3,3 volts

5V - Alimentação de 5 volts

SUMÁRIO

1 INTRODUÇÃO	1
1.1 Problema Científico.....	2
1.2 Campo de Acção	2
1.3 Objectivo:	2
1.3.1 Objectivo geral.....	2
1.3.2 Objectivos específicos	2
1.4 Hipótese.....	2
1.5 Metodologia.....	3
1.6 Visão Geral do Projecto	3
1.7 Organização do trabalho.....	4
2 FUNDAMENTAÇÃO TEÓRICA	5
2.1 Origem e histórico da galinha doméstica	5
2.2 Biologia dos ovos	6
2.2.1 Estrutura dos ovos.....	6
2.3 Histórico de chocagem de ovos.....	7
2.4 Incubação artificial	8
2.5 Tipos de Incubadoras.....	8
2.6 Funcionamento ótimo da máquina incubadora.	9
2.7 Desempenhos comparados das galinhas chocas e das incubadoras	9
2.8 Pré-aquecimento antes da incubação.....	10
2.9 Desenvolvimento dos embriões da galinha	10
2.10 O processo de incubação	11
2.11 Aumento do tamanho da câmara de ar	11
2.12 Perfuração da casca	11
2.13 Nível de humidade.....	12

2.14 Ovoscopia	12
2.15 Eclosão	13
2.16 Manutenção da incubadora.....	14
2.16.1 Manutenção preventiva	14
2.16.2 Manutenção planeada	14
2.16.3 Manutenção corretiva não planeada	14
3 DESCRIÇÃO DO HARDWARE E SOFTWARE	15
3.1 HARDWARE	15
3.1.1 ESP32.....	15
3.1.2 Pinagem de um ESP32	15
3.1.3 ADC – Portas Analógicas ESP32	16
3.1.4 Display de OLED.....	17
3.1.5 Sensor de Temperatura e Umidade DHT11	18
3.1.6 Sistema de aquecimento no interior da incubadora.....	19
3.1.7 Controlo de Aquecimento da Incubadora Iamoc	20
3.1.8 Sistema de controlo de nascimento dos pintos usando o sensor ultrassônico	21
3.1.9 Ondas Ultrassônicas.....	21
3.1.10 O accionamento do sensor	22
3.1.11. Velocidade de uma onda ultrassônica	22
3.1.12. Velocidade do Som no Ar	22
3.1.13. Sistema de controlo de nível da água com higrómetro	23
3.1.14. Motor de rolagem dos ovos.....	24
3.1.15. Bomba RS385	25

3.1.16. Micro Ventilador	26
3.1.17 Buzzer activo.....	26
3.2 SOFTWARE	27
3.2.1 Arduíno IDE	27
3.2.4.1 Componentes de Interface	29
3.2.5.1 Serviços oferecidos pelo Firebase.....	31
3.2.5.2 Vantagens do Firebase	32
3.2.5.3 Desvantagens do Firebase.....	32
4 IMPLEMENTAÇÃO DO PROTÓTIPO	33
4.1 Modelagem.....	33
4.2 Elaboração do Código Fonte	35
4.2.1 Código fonte final	35
4.3 Montagem do protótipo	43
4.3.1 Simulações.....	43
4.3.2 Problemas Encontrados	43
4.4 Custos para a construção do projeto	44
CONCLUSÕES	45
REFERÊNCIAS	46
APÊNDICE B – CÓDIGO FONTE TESTE DO SENSOR DHT11	49
APÊNDICE C – CÓDIGO FONTE TESTE DO SENSOR ULTRASSÔNICO.....	50
APÊNDICE D – CÓDIGO FONTE TESTE DO HIGRÓMETRO.....	51
APÊNDICE E – REALTIME DATABASE.....	53
APÊNDICE F – TELA E CÓDIGO FONTE NO APP INVENTOR	54
Tela de login e Código	54

Tela da alteração do password e Código	55
Tela de visualização das informações	55

1 INTRODUÇÃO

O consumo alimentício humano tem crescido exponencialmente conforme cresce o número da própria população mundial. Existem diversas fontes de produção alimentar que inclui a componente vegetal e as distintas componentes animais. Em todas estas formas, existem produções em larga, média e pequenas escala. O presente projecto, foi concebido para produção em pequena escala, e adapta-se a um produtor individual, que pretende trabalhar com no máximo 20 chocagens por vez. A Incubadora Iamoc é uma solução para quem deseja chocar ovos dos diferenciados tipos, porém alguns factores limitam os tipos de ovos que se deve às suas especificações e tempo de incubação. A incubadora é um dispositivo que necessita de muitas informações integradas para apresentar êxito em suas funcionalidades, os ovos necessitam de uma viragem regular, um controlo constante de temperatura e uma boa humidade interna [5].

O projecto visa construir uma incubadora que torne o processo de incubação artificial o mais inteligente possível, porém existem procedimentos que ainda dependem de auxílio humano, como para a reposição da água utilizada para manter a humidade, a viragem dos ovos e o nascimento dos pintos [5].

A Incubadora terá um sensor de temperatura e de humidade e um microcontrolador fará o monitoramento constante desses valores para garantir o desenvolvimento do embrião. Além disso, o usuário do sistema poderá acompanhar em seu telemóvel, através do aplicativo o valor desses dados. Será instalada dois sensores no interior da incubadora, um que irá detectar se algum ovo eclodiu e irá informar imediatamente ao produtor de forma remota e o outro irá medir o nível da água [5].

O projecto tem aplicações tanto comerciais, em indústrias produtoras de aves, quanto particularmente, para pequenos produtores. Actualmente, existem no mercado algumas incubadoras automatizadas, porém a maioria sem monitoramento ou controlo remotamente [5].

1.1 Problema Científico

De que forma o monitoramento via celular móvel associado a automação de equipamentos electrónicos poderá facilitar o controlo da incubadora?

1.2 Campo de Acção

O campo de acção em que este problema científico está assente é a Avicultura.

1.3 Objectivo:

1.3.1 Objectivo geral

O projecto tem como objectivo principal o desenvolvimento de uma incubadora com controlo de temperatura e humidade automatizada, monitoramento remoto, que servirá para ajudar a aumentar a produtividade no desenvolvimento de aves e minimizar prejuízos

1.3.2 Objectivos específicos

Os objectivos específicos deste projecto são:

- Criar uma caixa de madeira com lâmpada, sensor de temperatura e humidade, para fazer com que os embriões se desenvolvam de uma forma rápida e eficiente sem a intervenção humana.
- Elaborar um software para monitorar os parâmetros da incubadora.
- Instalar um sensor que possa captar o nascimento dos pintos depois de 21 dias e notificar a partir do telefone ou de forma presencial.

1.4 Hipótese

Partiu-se da hipótese de que a implementação de um sistema de incubação artificial de ovos de galinha remotamente proporcionará maior controlo da temperatura, humidade, giro dos ovos, nível da água e nascimento dos pintos.

1.5 Metodologia

O estudo do problema e as melhores ferramentas para o desenvolvimento foi realizado através de pesquisas bibliográficas incluindo livros, artigos, sites com conteúdo específico de avicultura. A maior parte do conteúdo foi pesquisa online, porém o caminho para encontrar o material adequado foi muitas vezes orientado por profissionais da área da avicultura e colegas da área de TI quando o assunto era relacionado à informática. Os resultados das pesquisas são apresentados no capítulo 2, onde são descritos mais detalhadamente.

1.6 Visão Geral do Projecto

O projecto basicamente é constituído por módulo DHT11, sensor ultrassónico, higrómetro, microcontrolador (Esp32), e demais componentes electrónicos, tais como leds e buzzer. Além dos componentes citados, foi escrito um código fonte para a comunicação entre as placas. Os dados serão enviados para a rede em tempo real. Um supervisor baseado em um aplicativo instalado do telefone interpretará esses dados e indicará ao usuário os parâmetros do sistema. A interface de usuário será desenvolvida na plataforma de programação em bloco (Mit app inventor) e toda a lógica do sistema será programada no Esp32. Os dados poderão ser transmitidos via Firebase.

O diagrama esquemático do projecto proposto pode ser observado na Figura 1.1, na qual representa de forma objectiva a composição geral do projecto.

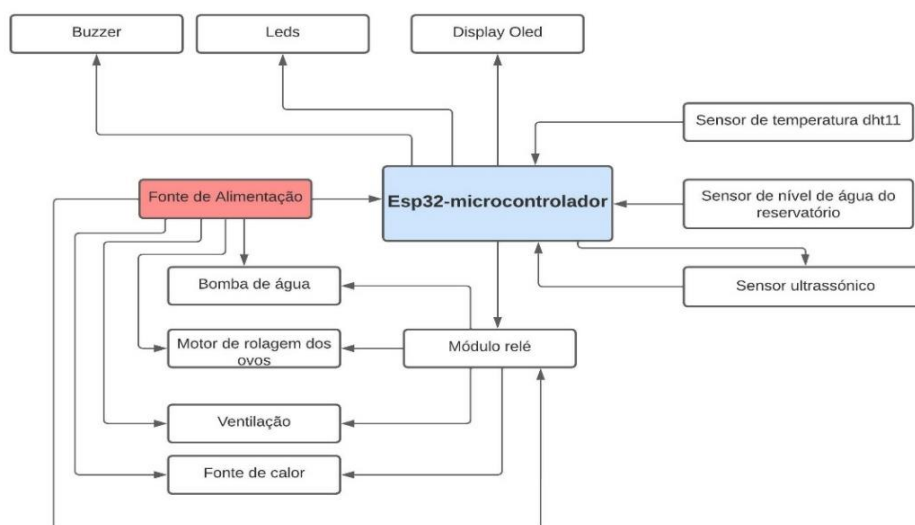


Figura 1.1: Diagrama esquemático do projecto.
Fonte: Autoria Própria

1.7 Organização do trabalho

Este trabalho é dividido em quatro capítulos, incluindo a Introdução, que trata do tema proposto, a motivação do projecto, os principais objectivos, e essa secção que descreve toda a estrutura do trabalho.

O segundo capítulo, Fundamentos Teóricos, trata de assuntos sobre a galinha, histórico de chocagem de ovos, manutenção da incubadora, tipos de incubadoras e outras informações.

O terceiro capítulo, Descrição do Hardware e Software, aborda as especificações dos componentes de controlo de comunicação serial do microcontrolador utilizado detalhado dos sensores, entre outros dispositivos e softwares

No quarto capítulo, Implementação do protótipo, é apresentado o desenvolvimento do projecto, como a integração entre o hardware e software, desenvolvimento do código na linguagem C e outras informações.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Origem e histórico da galinha doméstica

A avicultura é a criação de aves para fins alimentares, em especial, carne e ovos. Entre as espécies criadas na avicultura destacam-se o frango e a poedeira. Em menor escala, também são criadas aves como: perus, patos, gansos, codornas, avestruzes e outras. É uma actividade económica cada vez mais relevante mundialmente. Em Angola, a avicultura é considerada por muitos como a actividade mais dinâmica. [7].

As aves podem ser definidas como animais que possuem penas e, considerando a evolução dos vertebrados, são descendentes dos répteis, isto é, devido à presença de escamas nas canelas e, ainda, a outras semelhanças de natureza anatômica [13].



Figura 2.1: Galinha doméstica.

Fonte: <https://www.infoescola.com/aves/galinha/>.

A domesticação das aves é mencionada desde a antiguidade, há 3246 anos a.C. Com o passar dos tempos, as aves representaram para os povos diversos símbolos e oferendas. A galinha teve sua domesticação originada na Índia; as actuais variedades têm suas origens a partir da espécie asiática selvagem denominada *Gallus gallus* [7].

As galinhas chegam a botar mais de duzentos ovos por ano, os quais podem estar fertilizados ou não. Se a galinha acasalou com um galo antes de botar, os ovos são fertilizados e deles nascerão pintinhos. Já aqueles ovos utilizados na alimentação não são fertilizados, foram botados sem a junção. Um ovo leva cerca de 24h para ser produzido e sua coloração é determinada pela raça da galinha. Dentro das granjas os avicultores dividem as galinhas em aves de postura (produção de ovos), aves de corte (consumo da carne) e aves de dupla aptidão (postura e corte) [8].

As aves destinadas à produção de carne devem apresentar rápido ganho de peso, crescimento uniforme, empenamento precoce e de cor branca, pernas curtas, resistência a doenças, entre outras características. Já as aves destinadas para produção de ovos de consumo ou de incubação devem apresentar baixa mortalidade, alta capacidade de postura, alta percentagem de ovos grandes, ovos com casca resistente e uniforme, maturidade sexual precoce, alta fertilidade [8].

2.2 Biologia dos ovos

2.2.1 Estrutura dos ovos

O ovo é uma estrutura complexa que é constituída por: a casca (conquilha), as membranas da casca, a clara do ovo (albúmen), dois cordões torcidos (calazas), a gema e o óvulo (gérmen). Ao óvulo não fecundado também se chama o oócito ou gâmeta feminino, enquanto que o fertilizado é conhecido como embrião [2].

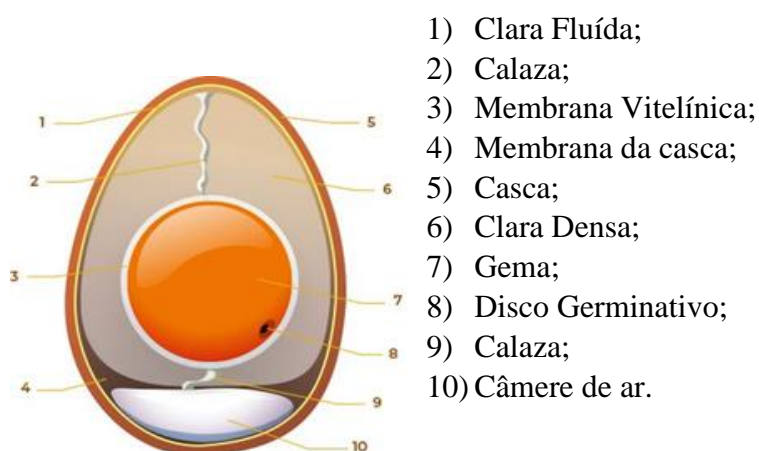


Figura 2.2: A estrutura de um ovo.

Fonte: <https://freixiovo.pt/empresa>.

Podem-se distinguir os seguintes componentes

- **Casca**, uma camada calcária, dura e rija que protege o interior do ovo.
- **Duas membranas da casca** (a membrana externa e interna); estas membranas ficam separadas na ponta mais arredondada do ovo, onde formam a **câmara de ar**. Como se perde humidade através de evaporação, a câmara de ar é maior nos ovos maduros. Esta é a razão por que os ovos velhos flutuam na água enquanto que os ovos frescos se afundam. A câmara de ar desempenha um papel importante durante a eclosão.
- **A clara do ovo** (o albúmen) é uma massa gelatinosa que se torna branca quando cozida. Nos ovos velhos a clara de ovo é menos viscosa (mais fluida), de modo que ao partir um ovo velho sobre um prato, o seu conteúdo cobrirá uma superfície maior do que um ovo fresco, recém posto.
- **Dois cordões torcidos** (as calazas) de clara de ovo contêm a gema no centro do ovo.
- **A gema** é a substância amarela que é envolta pela membrana da gema. A cor da gema depende das rações consumidas pelas galinhas.
- **O óvulo** (oócito/gâmeta feminino) encontra-se na superfície da gema. O desenvolvimento do embrião galináceo começa quando o oócito é fecundado. A gema e a clara do ovo contêm substâncias sumamente nutritivas (proteínas e gorduras) aproveitadas pelo embrião em desenvolvimento e o pintainho recém eclodido [2].

2.3 Histórico de chocagem de ovos

A chocagem artificial de ovos de aves desde o Egito antigo, cerca de 3000 anos atrás. Eram criadas câmaras enormes de tijolos de barro e com pequenos fornos com capacidade para milhares de ovos. A temperatura era controlada através da queima de palha ou outros materiais, colocados em prateleiras acima dos fornos. No topo das câmaras havia ventilações para permitir que a fumaça saísse e que o ar circulasse. A humidade era controlada espalhando água sobre os ovos quando necessário. Os ovos também eram virados duas vezes ao dia. Para saber a temperatura ideal para os ovos, os criadores viviam de fato dentro das câmaras e ajustavam o calor de acordo com sua própria percepção de temperatura. Há registros de que eles seguravam o ovo contra pálpebra, a região de nosso corpo mais sensível à temperatura, para saber se deviam aumentar ou diminuir o calor [5].

2.4 Incubação artificial

A incubação artificial é um processo cujo objectivo é fornecer artificialmente ao ovo um ambiente controlado para o desenvolvimento do embrião, procurando controlar a temperatura, a viragem do ovo, o fluxo de ar, a humidade relativa e a higiene dentro da câmara nos níveis adequados. Desvios desses factores em relação aos valores ótimos, podem inviabilizar o desenvolvimento do embrião, resultando em um aumento da mortalidade e consequentemente na diminuição da eclosão [1].

2.5 Tipos de Incubadoras

As máquinas incubadoras podem ser de estágio único ou de estágio múltiplo. Nas incubadoras de estágio único todos os ovos são introduzidos ao mesmo tempo (carga única), ficando totalmente vazias no dia da transferência para os nascedouros. Nestas incubadoras, é possível manter as condições precisas de temperatura, humidade e ventilação requeridas pelo embrião em cada fase do desenvolvimento. Além disso, o período de vazio permite a limpeza, higiene, desinfecção da máquina e práticas de manutenção [1].

Nas máquinas de estágio múltiplo vão se introduzindo séries de ovos segundo a idade, ocupando-se o espaço deixado pelos ovos transferidos aos nascedouros (carga escalonada). Estas máquinas permitem atingir as condições ótimas de temperatura e humidade pouco tempo após serem introduzidos os ovos na máquina, permitindo economizar mais energia elétrica que as de estágio único. Sabe-se que diferentes estágios de incubação requerem condições ambientais específicas para se obter o melhor processo de incubação. Estas máquinas nunca são esvaziadas, havendo nelas embriões em diferentes fases de desenvolvimento e tendo uma temperatura constante entre 36,1 e 38 °C [1].

2.6 Funcionamento ótimo da máquina incubadora.

Série de processos básicos para o ótimo desempenho da máquina:

- O fluxo de ar entre a massa de ovos deve ser consistente para que ocorra a transferência adequada de calor entre os embriões e o ambiente;
- O ar introduzido na máquina deve atender as necessidades de oxigênio dos embriões.
- É preciso que a capacidade de aquecimento e de resfriamento sejam adequados e uniformes;
- A máquina deve ter a habilidade de reduzir a humidade visando atingir as necessidades do embrião e ainda manter uniforme e adequado o resfriamento;
- Para transferir e remover calor de uma grande massa de ovos, a máquina deve ter a capacidade de manter a temperatura do ar entre 36.1 e 38 °C no período de incubação;
- A humidade dentro da incubadora deve ficar entre 55% e 65% nos 18 primeiros dias de incubação, e de 80% nos últimos 3 dias, para facilitar a eclosão dos ovos;
- Os ovos não devem ser virados depois de 18 dias [1].

2.7 Desempenhos comparados das galinhas chocas e das incubadoras

Uma galinha a chocar os ovos fornece exatamente a temperatura, humidade, ventilação e frequência de viragem apropriadas para os ovos se desenvolverem adequadamente. Os riscos que se correm pela incubação por choco são reduzidos, visto que se uma galinha choca abandonar o seu ninho perder-se-ão apenas 8-14 ovos. Uma incubadora de fabrico caseiro não cumprirá tão facilmente com as normas requeridas no que diz respeito à temperatura, humidade e ventilação. Para otimizar os resultados da incubação, é necessário ter alguma experiência para operar cuidadosamente a incubadora. Mesmo assim, os resultados podem mostrar-se inferiores aos resultados previstos. Caso o combustível se esgote ou se houver um corte de energia prolongado, perder-se-ão, pelo menos, 40 a 100 ovos, consoante a quantidade de ovos presentes na incubadora e o tamanho deste aparelho. Na tabela 2.1 apresenta-se um resumo dos vários factores que devem ser levados em conta ao escolher o método de incubação [2].

Tabela 2.1: Escolha entre incubação por choco ou por incubadora [2].

Aspecto	Choco	Incubadora
1. Técnica	(10-50 ovos simultaneamente, empregando 1 até 5 galinhas chocas) As galinhas nem sempre ficam chocas quando é necessário Nem sempre se mostram cooperativas	(40-100 ovos simultaneamente) A máquina está sempre Disponível
2. Custos de mão-de -obra	Limpeza (muito limitada) Água Abrigo Protecção	(Construção da incubadora) Verificação e regulação da fonte de calor e da temperatura Viragem artificial dos ovos Enchimento do reservatório de água
3. Custos de construção	Reduzidos: Caixas-ninhos ou cestos-ninhos Comedouras Bebedouros	Materiais locais Sensor de temperatura e humidade Sensor Ultrassónico Bomba de água Microcontrolador etc....
4. Custos de operação	Reduzidos: Rações Material de nidificação Vacinas	Electricidade Energia solar Vacinas
5. Desempenho e resultados	As condições da incubação natural são seguras (temperatura, humidade relativa, ventilação) As galinhas das raças locais ficam, geralmente, chocas durante o período de incubação, enquanto que isto é menos seguro no caso das raças melhoradas	Condições quase ótimas (sempre que haja um manejo adequado!) As raças melhoradas podem produzir ovos continuamente, durante um período prolongado
6. Riscos	Limitados	Riscos de aspecto técnico: perda de alguns ou até todos os ovos

2.8 Pré-aquecimento antes da incubação

É muito importante evitar as mudanças repentinas da temperatura, visto que provocarão a morte dos embriões. Portanto, os ovos que devem ser armazenados a uma temperatura abaixo de 20 °C, primeiro devem ser aquecidos a uma temperatura de 23-27 °C durante 12 horas, antes de serem mudados para a incubadora. A este método chama-se 'pré-aquecimento'. Um aquecimento gradual também reduz fortemente o risco de condensação que molharia os ovos, dando oportunidade a que bactérias penetrem a casca e a estraguem [2].

2.9 Desenvolvimento dos embriões da galinha

Quando os ovos fecundados são armazenados a uma temperatura inferior a 20 °C, haverá uma estagnação do desenvolvimento embrionário. O processo de incubação começará, de novo, com uma galinha choca ou numa incubadora [2].

2.10 O processo de incubação

Quando a incubação começa, o disco branco (óvulo fecundado) que se encontra em cima da gema começa a crescer e desenvolve-se num embrião. Formam-se o coração e as veias. As veias podem ser observadas a partir do ovoscópio, depois de 7-9 dias de incubação. Durante a incubação, o embrião de pinto continua a crescer dentro do ovo. Consome a maior parte da clara do ovo e parte da gema [2].

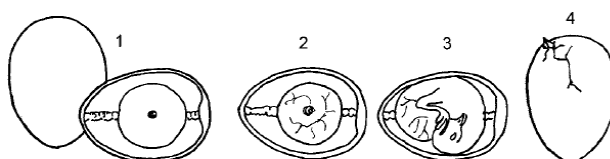


Figura 2.3: Desenvolvimento do embrião de pinto dentro do ovo [2].

2.11 Aumento do tamanho da câmara de ar

Durante a incubação a humidade presente dentro do ovo evapora e é substituída por ar na ponta mais arredondada do ovo. Portanto, a câmara de ar aumenta, gradualmente [2].

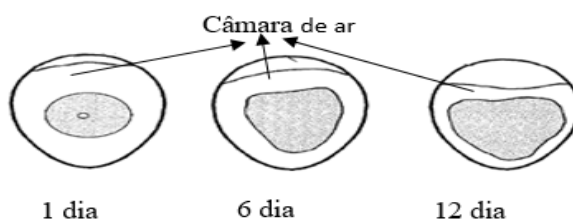


Figura 2.4: Tamanho da câmara de ar em relação ao número de dias de incubação [2].

2.12 Perfuração da casca

Alguns dias antes da eclosão, o pintinho deve usar o seu bico para perfurar a membrana debaixo da câmara de ar, de forma a começar uma respiração normal através dos pulmões. A partir de esse momento, pode-se ouvir o pintainho piando no interior do ovo. Quando o pintainho estiver acostumado a respirar normalmente começará a fazer um pequeno orifício na casca (perfuração da casca) [2].

2.13 Nível de humidade

Se a quantidade de humidade presente nos ovos não for suficiente, os ovos secarão, mas se houver demasiada humidade a câmara de ar ficará muito reduzida assim que não conterà espaço suficiente para o pintainho poder sobreviver a incubação [2].

2.14 Ovoscopia

O processo de desenvolvimento embrionário pode ser observado por um ovoscópio, que é uma fonte de luz em local escuro que permite a visualização parcial interna do ovo. Essa técnica ajuda a verificar se há defeitos da casca, tais como, rachaduras, rugosidade, casca fina ou impurezas [2].

Na Figura 2.5 apresenta-se o que se pode ver durante uma observação de um ovo a partir do ovoscópio, depois de 8 dias de incubação.

- A – Num ovo infértil pode-se distinguir apenas a sombra da gema e a câmara de ar.
- B – Num ovo fértil podem-se distinguir as veias estendidas e observam-se com maior facilidade perto da câmara de ar. O embrião pode ser visto, na forma de uma sombra escura. A câmara de ar é mais fácil de discernir do que num ovo infértil.
- C – Num ovo que contém um embrião morto, as veias contraem-se num anel em volta do embrião, que é visível como uma sombra escura. A câmara de ar é claramente visível [2].

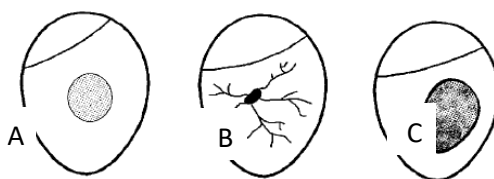


Figura 2.5: Observação de um ovo a partir do ovoscópio, depois de 8 dias de incubação [2].

2.15 Eclosão

Os pintainhos saem dos ovos depois de, aproximadamente, 21 dias (no caso de ovos de galinha). Se os pintainhos tiverem dificuldades de eclodirem, é provável que a humidade seja demasiadamente baixa. Para ajudá-los a saírem dos ovos, poder-se-á humedecer, levemente, com água a cama do ninho, de modo a que haja alguma evaporação, aumentando a humidade do ambiente [2].

Na Tabela 2.2, são listadas algumas irregularidades no nascimento de pintinhos, as quais podem auxiliar no diagnóstico do problema.

Tabela 2.2: Diagnóstico de problemas de nascimento [2].

Nascimento precoce	Temperatura de incubação alta Ovos pequenos
Nascimento tardio	Temperatura e humidade baixas na incubação. Tempo de armazenamento Ovos grandes
Pintinho pegajoso	Temperatura alta próximo ao nascimento Viragem inadequada
Umbigo não cicatrizado	Temperatura de incubação alta humidade alta próximo ao nascimento Tempo de armazenamento
Pintinho sem andar	Varição de temperatura durante a incubação; Idade do lote Manuseio do ovo na primeira semana de incubação
Pintinho anormal	Bico cruzado: hereditário ou infecção por vírus Faltando olhos: temperatura alta PESCOÇO TORTO: nutrição Pés abertos: bandeja do nascedouro lisa

2.16 Manutenção da incubadora

A medida em que a incubadora é utilizada há a necessidade em se realizar as manutenções que são divididas em três grupos: as preventivas, as planejadas e as corretivas (não planejadas).

2.16.1 Manutenção preventiva

A manutenção preventiva consiste em controlar constantemente os equipamentos e os materiais para que as suas funcionalidades continuem tendo um ótimo desempenho.

Controle constante dos equipamentos:

- Calibrar máquinas;
- Calibrar salas;
- Controlar perda de humidade;
- Controlar ovos bicados;
- Controlar via celular o nível da água e a temperatura;
- Controlar via celular o nascentimento dos pinos (nos 3 últimos dias);

2.16.2 Manutenção planeada

Com relação à manutenção planeada, ela trata de corrigir o desempenho da incubadora que esteja apresentando um desempenho menor que o desejado.

2.16.3 Manutenção corretiva não planeada

Esta é aquela manutenção da incubadora em que a falha já aconteceu. Com isso, o especialista realiza todos os reparos necessários para que o equipamento tenha o seu desempenho normal.

3 DESCRIÇÃO DO HARDWARE E SOFTWARE

Este capítulo explica as especificações e pinagens dos dispositivos usados e seu devido funcionamento nesse projecto, tanto a parte física, ou seja, o hardware, como a parte lógica, o software utilizado.

3.1 HARDWARE

3.1.1 ESP32

Desenvolvido pela empresa Espressif, o ESP32 apresenta-se como um meio inovador no desenvolvimento de projectos automatizados. Esse pequeno componente demonstra ser mais versátil, pois além do clássico módulo de comunicação Wi-Fi, apresenta um sistema com processador Dual Core, Bluetooth híbrido e múltiplos sensores embutidos, tornando a construção de sistema como internet das coisas (IoT) muito mais simples e compacto [12].

3.1.2 Pinagem de um ESP32

Inicialmente, iremos explorar o desenvolvimento de programações através do Arduíno IDE, por ser uma plataforma mais simples e popular entre aqueles que estão iniciando no mundo dos sistemas embarcados. Analisando a pinagem de um ESP32, podemos encontrar 34 portas GPIO, 3 SPI, 2 I2S, 18 canais ADC, 3 UART, 10 pinos de leitura capacitiva e PWM. Na imagem abaixo, será apresentado um breve resumo das funções presentes em cada pino [11].

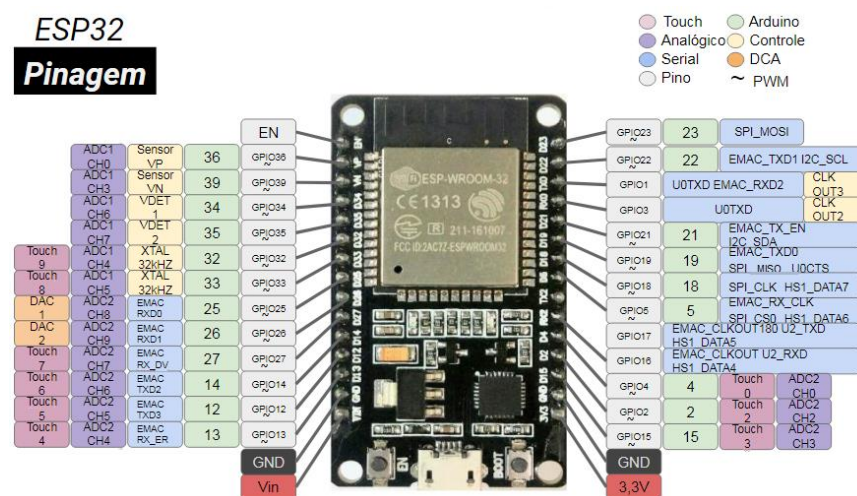


Figura 3.1: Pinagem de um ESP32.

Fonte: <https://curtocircuito.com.br/blog/Categoria%20IoT/conhecendo-esp32>.

A legenda demonstrada acima, indica que os valores destacados em verde serão utilizados na declaração de pinos em uma programação via Arduino IDE. Portanto, ler a entrada analógica de um sensor Touch, por exemplo, o sensor presente no pino GPIO04 (D4), o programador poderá optar por duas opções: Declarar o pino 4, ou touch T0 como INPUT [11].

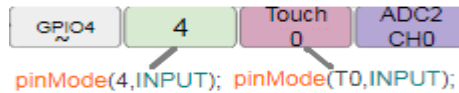


Figura 3.2: Declarações de Pinos.

Fonte: <https://curtocircuito.com.br/blog/Categoria%20IoT/conhecendo-esp32>.

3.1.3 ADC – Portas Analógicas ESP32

Portas analógicas são comumente aplicadas ao controle e análise de sensores. Os 18 pinos analógicos disponíveis em placas ESP32, operam com resolução em 12 bits, tornando a faixa de leitura de 0 a 4095. Trabalhar com qualquer pino ADC será exatamente igual ao procedimento utilizado em placas Arduino, realizando leituras e controle por meio de funções como **analogRead** e **analogWrite**. Porém, ao conectar-se ao Wi-Fi, por exemplo, os pinos ADC2 não irão fornecer leituras, pois, até o presente momento, tais pinos serão utilizados como alimentação para o módulo, portanto, se desejar atribuir uma leitura analógica a qualquer programação conectada a internet, utilize os pinos ADC1 [11].

• ADC2_CH1: 0	• ADC2_CH3: 15	• ADC2_CH7: 27
• ADC2_CH2: 2	• ADC2_CH8: 25	• ADC1_CH4: 32
• ADC2_CH0: 4	• ADC2_CH9: 26	• ADC1_CH5: 33
• ADC2_CH5: 12	• ADC1_CH7: 35	• ADC1_CH6: 34
• ADC2_CH4: 13	• ADC1_CH0: 36	• ADC1_CH2: 38
• ADC2_CH6: 14	• ADC1_CH1: 37	• ADC1_CH3: 39

Figura 3.3: Pinos ADC.

Fonte: <https://curtocircuito.com.br/blog/Categoria%20IoT/conhecendo-esp32>.

3.1.4 Display de OLED

Displays OLED são painéis visuais eletrônicos que utilizam díodos emissores de luz orgânicos (ou Organic Light-Emitting Diodes, sigla em inglês para OLED) como sua fonte de iluminação central. OLED é um tipo de tecnologia de display eletroluminescente, em que uma camada de material orgânico gera luz quando as moléculas do díodo são agitadas por uma corrente elétrica [12].

O **Display Oled 128x64** é uma solução muito boa para quem procura um display para exibição de poucas informações, mas com muita nitidez, possui uma ótima nitidez por causa da tecnologia Oled (Organic Light Emitting Diode). O **Display Oled 128x64** possui 128x64 pixels que podem ser controlados um a um através do I2C pelo chip controlador SSD1306, o **Display Oled 128x64** possui uma tela de 0.96" e não possui backlight pois o display Oled já possui luz própria, tornando dessa forma, seu projeto se torna mais econômico por não ter o consumo de energia do backlight. É compatível com os principais microcontroladores do mercado [12].

Display OLED Especificações:

- Conexão: I2C (endereço 0x3C).
- Área visível: 0.96".
- Resolução: 128 x 64.
- Controlador: SSD1315.
- Tensão de Alimentação: 3.3V / 5V.
- Tensão de I/O: 3.3V / 5V.
- Cor Letra: Branca.
- Dimensões da placa: 28x27mm.

Ligação entre o display OLED e Esp32.

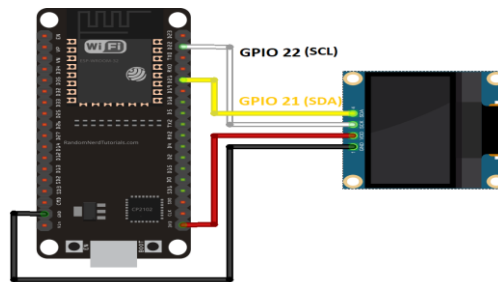


Figura 3.4: Conexão do esp32 com display OLED.

Fonte: <https://randomnerdtutorials.com/esp32-ssd1306-oled-display-arduino-ide/>.

3.1.5 Sensor de Temperatura e Umidade DHT11

O Sensor de humidade e temperatura Dht11 é um sensor de temperatura e humidade que permite fazer leituras de temperaturas entre 0 a 50° Celsius e humidade entre 20 a 90%, muito usado para projetos com Arduino. O elemento sensor de temperatura é um termistor do tipo NTC e o sensor de humidade é do tipo HR202, o circuito interno faz a leitura dos sensores e se comunica a um microcontrolador através de um sinal serial de uma via [4].

Pinagem

- Pino 1: Alimentação - 3,0 a 5,0 VDC.
- Pino 2: Saída Data.
- Pino 3: GND – 0V.

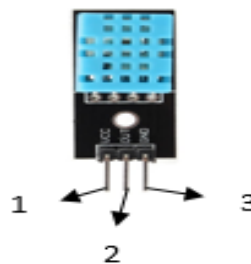


Figura 3.5: Pinagem do módulo DTH11.

Fonte: Autoria Própria.

Especificações e características

- Tensão de operação: 3,5 a 5,5VDC.
- Corrente de operação: 0,3mA.
- Corrente de operação (em stand by): 60 μ A.
- Resolução: 16 bits.
- Faixa de medição (humidade): 20 a 90%.
- Faixa de medição (temperatura): 0° a 50° celsius.
- Precisão (humidade): $\pm 5\%$.
- Precisão (temperatura): $\pm 2^\circ$ celsius.
- Tempo de resposta: 2s.

3.1.6 Sistema de aquecimento no interior da incubadora

O sistema de aquecimento da incubadora é garantido através da utilização de uma exclusiva fonte de calor, bastante presente nos mais diversificados projetos de mesma espécie, a lâmpada incandescente [14].



Figura 3.6: Lâmpada incandescente

Fonte: <https://blog.dimensional.com.br/tipos-de-lampada-diferencas-entre-fluorescente-incandescente-e-inteligente/>.

Em geral uma chocadeira de 500 ovos exige cerca de 400W.

Considerado o modelo de lâmpada mais comum a muito tempo, sempre esteve presente na maioria das casas dos Angolanos e agora está sendo substituída por lâmpadas de LED devido ao seu alto aquecimento e baixa iluminação. A lâmpada incandescente é produzida através de um exclusivo filamento de metal que quando recebe energia emite uma luz amarelada, porém desta energia que chega até a lâmpada, apenas 5% é convertida realmente à luz, o restante transforma-se em calor. Considerada a lâmpada mais quente, quando instalada tende a aquecer

o local o qual está proporcionando uma elevação na temperatura, em lugares pequenos uma lâmpada já é o suficiente, para ambiente menores, além das lâmpadas domésticas existem pequenas lâmpadas DC que também podem ser utilizadas [14].

3.1.7 Controle de Aquecimento da Incubadora Iamoc

O controle de aquecimento da incubadora Iamoc é feito através da famosa modulação por largura de pulso (PWM), sistema de comunicação digital utilizado no controle de saídas de energia. Assim como nos famosos dímeres utilizados para o controle de luminosidade de equipamentos de corrente alternada (AC), o controlador PWM realiza a verificação do tempo em 0 para determinar o brilho de um LED ou até mesmo para o controle de velocidade de motores. O controle de funcionamento do PWM baseia-se no tempo em que uma senoide permanece em HIGH ou LOW (máximo ou mínimo) em um período de 20 segundos, para o Esp32 que possui um sistema de funcionamento de 3.3V, HIGH será 3.3V e LOW será 0V [14].

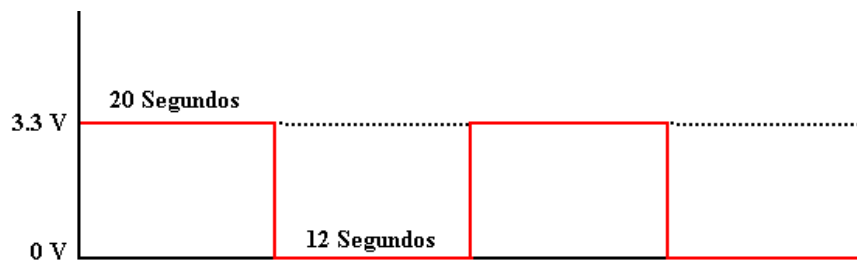


Figura 3.7. Período de trabalho da lâmpada incandescente.

Fonte: Autoria Própria

Através do controlador PWM, fiz o controle de luz da lâmpada, ao ligar a incubadora Iamoc a luz estará em seu máximo e permanecerá assim até que atinja a temperatura máxima (37.8 °C), após atingir a temperatura de 37.8 °C irá apagar a luz da lâmpada de maneira a estabilizar a temperatura e mantê-la constante [14].

3.1.8 Sistema de controlo de nascimento dos pintos usando o sensor ultrassônico

Descrição:

O Sensor de Distância Ultrassônico HC-SR04 pode ser utilizado na construção de carrinhos que desviam automaticamente de obstáculos, régua eletrônica e até aplicações de automação para detecção de distância com ótima precisão. No meu projecto usei o sensor HC-SR04 a fim de detectar a presença dos pintos no interior da incubadora [15].

Espectro de audição

O Sensor ultrassônico funciona com o princípio de emissão e recepção de ondas sonoras, numa frequência mais alta que a audição humana pode perceber. O sistema auditivo das galinhas está limitado a perceber frequências entre 125 Hz e 2000 Hz [15].

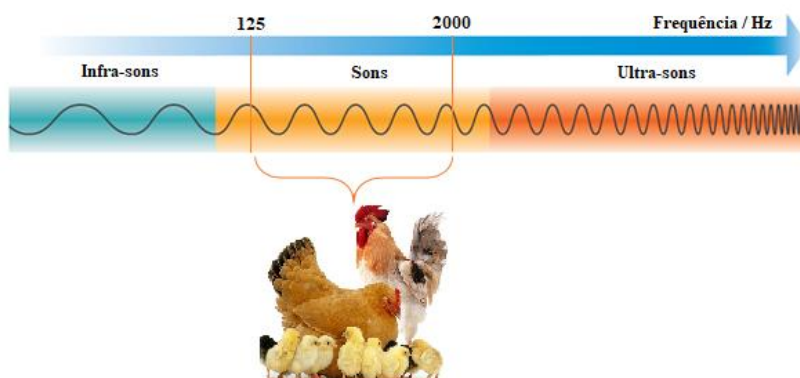


Figura 3.8: Espectro de audição da galinha
Fonte: Autoria Própria

3.1.9 Ondas Ultrassônicas

Ondas sonoras são ondas mecânicas que vibram em uma determinada frequência em Hertz, 1 Hz por segundo seria uma vibração por segundo, 2 Hz por segundo, duas vibrações em um segundo e assim por diante. Já o sensor HC-SR04, trabalha com ondas sonoras acima de 40 KHz, que são chamadas de ondas ultrassônicas. O módulo sensor possui emissor e receptor, e trabalha através do princípio da física que envolve a velocidade do som, em um determinado espaço de tempo. Dessa maneira o sensor consegue detectar um objecto e inclusive determinar sua distância. A velocidade sonora é de aproximadamente 343 m/s, com a temperatura de 20 °C [15].

3.1.10 O accionamento do sensor

O accionamento do sensor é realizado pelo envio de um pulso com 10 μ s de largura no pino TRIGGER. Em seguida, o sensor envia oito pulsos ultrassônicos para o processo de medida. Depois da recepção dos pulsos refletidos, o sinal de saída ECHO apresenta um pulso de resposta, cuja largura é proporcional à distância. Este processo de medida é ilustrado na figura 3.9 abaixo [15].

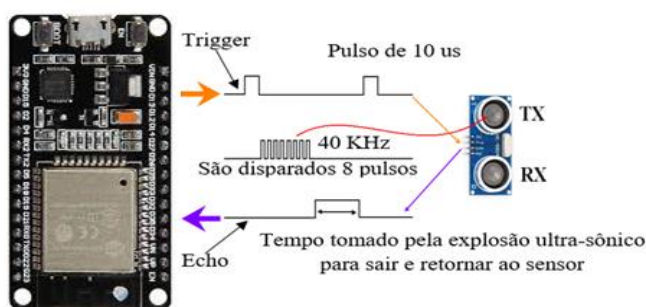


Figura 3.9: Sinais do processo de medida com HC-SR04.

Fonte: Autoria Própria

3.1.11. Velocidade de uma onda ultrassônica

A velocidade de uma onda ultrassônica, é caracterizada pelo material que está atravessando e é constante para um material independente da frequência e comprimento de onda em que: $c = f \lambda$, c velocidade do som no material (m/s), f frequência principal (ou frequência fundamental em Hz) e λ comprimento de onda (m). Apesar de a velocidade do som ser considerada uma constante característica de cada material, alguns factores como a temperatura podem acarretar variações consideráveis para cada meio [21].

3.1.12. Velocidade do Som no Ar

A velocidade do som em um gás sofre a influência da temperatura. A seguinte fórmula pode ser usada para indicar uma boa aproximação do valor da velocidade do som no ar, em função da temperatura:

$$V = 330,4 + 0,59T \text{ onde.}$$

V: velocidade em m/s.

T: temperatura em graus Celsius (°C).

Pinagem:

- Pino 1: Alimentação - 5VDC.
- Pino 2: Trigger.
- Pino 3: Echo.
- Pino 4: GND – 0V.

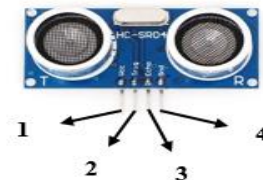


Figura 3.10: Pinagem do sensor ultrassônico
Fonte: Autoria Própria

3.1.13. Sistema de controlo de nível da água com higrómetro**Descrição:**

Este Sensor de humidade do solo higrómetro foi feito para detectar as variações de humidade no solo. Ele funciona da seguinte forma: quando o solo está seco, a saída do sensor fica em estado alto e quando húmido, a saída do sensor fica em estado baixo. O limite entre seco e húmido pode ser ajustado através do potenciômetro presente no sensor que regulará a saída digital D0. Contudo, para ter uma resolução melhor, utilizei a saída analógica A0 a fim de obter a variação de tensão de acordo com o aumento ou a diminuição da água no reservatório [16].



Figura 3.11: Sensor de humidade do solo higrómetro.
Fonte: <https://www.makerhero.com/produto/sensor-de-umidade-do-solo-higrometro/>.

Especificações:

- Tensão de Operação: 3,3-5v.
- Sensibilidade ajustável via potenciômetro.
- Saída Digital e Analógica.
- Fácil instalação.
- Led indicador para tensão (vermelho).
- Led indicador para saída digital (verde).
- Comparador LM393.
- Dimensões PCB: 3×1,5 cm.
- Dimensões Sonda: 6×2 cm.
- Comprimento Cabo: 21 cm.

Pinagem:

- VCC: 3,3-5v
- GND: 0V
- D0: Saída Digital
- A0: Saída analógica.

3.1.14. Motor de rolagem dos ovos

Utilizou-se o motor 49TYJ para fazer girar a bandeja dos ovos a fim de prevenir a aderência do embrião à membrana da casca do ovo.

Especificações:

- Motor tipo micro-ondas 3/3,6 RPM 220-240V 49TYJ.
- Motor Síncrono.
- Alimentação: 220V.
- Frequência da alimentação: 50/60Hz.
- Velocidade final: 3/3,6 RPM.
- Potência: 4W.
- Eixo: Plástico.



Figura 3.12: Motor 49TYJ.

Fonte: <https://www.hkritescher.com/Synchronous-motor-49TYJ-pd41934486.html>.

3.1.15. Bomba RS385

Esta Minibomba de diafragma e pulverização com motor de 12V para dispensador de água (RS385) é capaz de impulsionar entre 1500ml a 2000ml por minuto, sendo destacada pela sua eficiência e precisão durante sua execução em conjunto com o Arduino, razão pela qual usei a minibomba RS385 a fim de abastecer o reservatório para que haja umidade no interior da incubadora [17].

Especificações:

- Modelo: RS385.
- Materiais: metal e plástico.
- Tensão nominal: 12V.
- Corrente de trabalho: 0,5 a 0,7^a.
- Corrente em máxima eficiência: ~2^a.
- Altura de aspiração máxima: 2m.
- Elevação máxima: 3m.
- Vazão de água máxima: ~1,5 a 2 litros/min.
- Pressão: 3,05916 Kgf/cm².
- Vida útil: até 2500 horas, temperatura da água: até 80graus.
- Potência: 6 W/h (12V).
- Diâmetro de entrada e saída: ~7,6mm.
- Diâmetro do motor: 28,6mm.
- Dimensões da bomba: 90x40x35 mm.
- Peso: 106g.



Figura 3.13: Bomba RS385.

Fonte: <https://www.eletrogate.com/mini-bomba-de-agua-motor-dc-rs-385>.

3.1.16. Micro Ventilador

O micro ventilador é uma peça de funcionamento extremamente simples, mas que é indispensável para qualquer incubadora. É o ventilador o responsável por espalhar a temperatura em todas as paredes da incubadora.



Figura 3.14: Micro ventilador 30x30x10mm 12v-3010.

Fonte: <https://www.impactocnc.com/micro-ventilador-cooler-30x30x10mm-12v-3010>.

3.1.17 Buzzer activo

O buzzer é muito utilizado em projectos robóticos com o pequeno microcontrolador para emitir diferentes sinais sonoros para emitir alertas em projetos com sensores de gás, de fumaça ou de movimentos sempre que um deles detecta alguma coisa anormal permitindo ao operador verificar imediatamente o que esta ocorrendo, nesses casos, estamos falando de um buzzer ativo. Neste projecto usou-se o buzzer a fim de alertar a presença do pinto no interior da incubadora.

Especificações:

- Buzzer tipo activo.
- Tensão de operação: 4 à 8VDC.
- Corrente de operação: 30mA.
- Saída de som mínima (a 10cm): 85dB.
- Frequência de ressonância: 2300±300 Hz.
- Temperatura de operação: -27 a +70 °C.
- Material: ABS.
- Cor: Preto.
- Dimensões: 11,8 x 9mm.



Figura 3.15: Buzzer tipo activo.

Fonte: <https://www.tecnotronics.com.br/buzzer-ativo-5v-bip-continuo.html>.

3.2 SOFTWARE

3.2.1 Arduíno IDE

Uma das alternativas de software mais simples utilizadas no desenvolvimento inicial de programações em placas ESP tem sido a plataforma de desenvolvimento Arduíno, pois, este possui uma grande comunidade de desenvolvedores, que costumam disponibilizar diversos exemplos e projectos de forma Open Source [11].

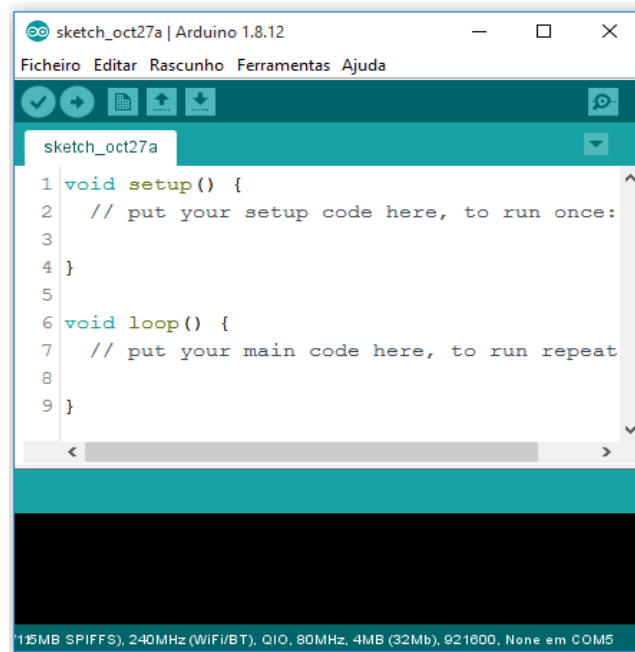


Figura 3.16: Ambiente de Desenvolvimento Integrado.
Fonte: Autoria Própria

A transferência de programações a qualquer microcontrolador, exigirá a seleção de pelo menos um dos modelos de placas compatíveis com o software, no caso de placas ESP32 ou similares, será preciso adicionar um novo conjunto nas opções de gerenciamento de placas, no menu **Arquivo >> Preferências**. Na opção URLs Adicionais para Gerenciamento de Placas, digite a seguinte URL:

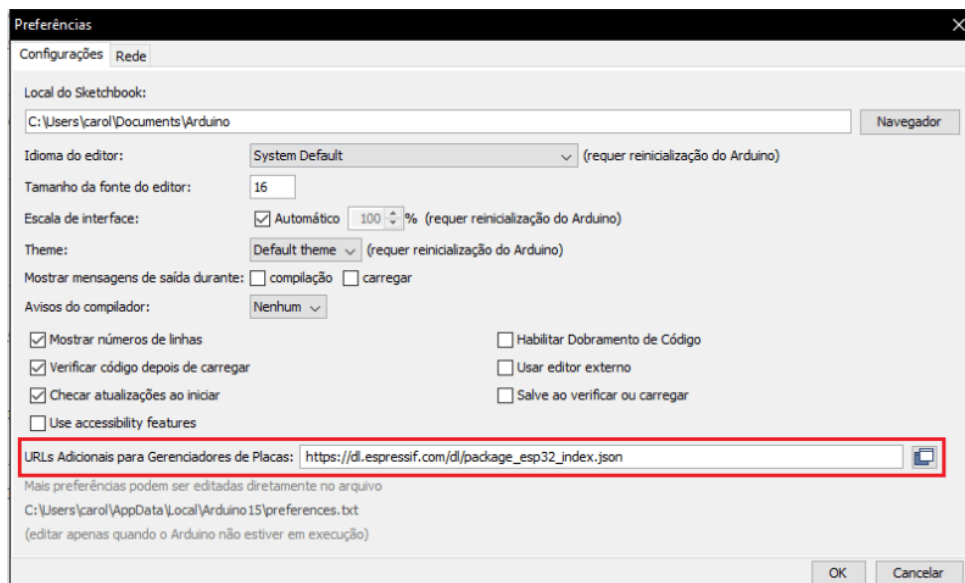


Figura 3.17: Janela de gerenciamento de placas.

Fonte: <https://www.blogdarobotica.com/2021/08/24/como-programar-a-placa-esp32-no-arduino-ide/>.

Caso haja outras URLs adicionadas, digite uma vírgula entre eles para separá-los. Confirme a adição do conjunto de placas ESP32 abrindo o menu **Ferramentas** e acessando a opção **Placa**, o modelo utilizado por placas padrão como o DOIT ESP32, será o **ESP32 Dev Module** [11].

3.2.2 Aplicativo em Android

O **MIT App Inventor** é um ambiente de programação visual intuitivo que permite a todos criar aplicativos totalmente funcionais para celulares Android, iPhones e tablets Android/iOS. Além disso, a ferramenta utiliza um estilo de programação em blocos que facilita a criação de aplicativos complexos e de alto impacto em ambientes de programação significativamente menores do que os ambientes tradicionais de programação. Tendo em vista a praticidade dessa plataforma de programação, esse artigo tem como finalidade ensinar você a criar sua conta, descrever os componentes de interface disponíveis na plataforma de modo que você sairá apto a fazer sua primeira criação [19].

3.2.3 Computador e Sistema Operacional:

- Macintosh (com processador Intel): Mac OS X 10.5 ou superior.
- Windows: Windows XP, Windows Vista, Windows 7, Windows 8 e Windows 10.
- GNU / Linux: Ubuntu 8 ou superior, Debian 5 ou superior.

3.2.4 Visão Geral no Mit App Inventor

Designer

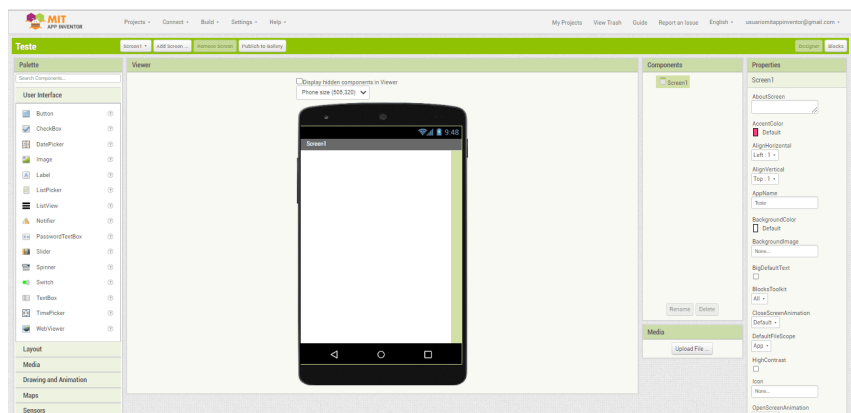


Figura 3.18: Interface de designer do Mit app inventor.

Fonte: <https://www.makerhero.com/blog/mit-app-inventor-desenvolvendo-aplicativos-para-smartphone/>.

Interface de designer do Mit app inventor é onde pode se trabalhar o app. Nessa janela pode se criar a aparência do aplicativo, e especifica quais os componentes que irão ser utilizados. Pode se escolher componentes na “Paleta” como botões, imagens, caixas de texto e funcionalidades como sensores e GPS [19].

3.2.4.1 Componentes de Interface

Os componentes de interface são aqueles que permitem ao usuário do aplicativo interagir com o programa, eles se encontram do lado esquerdo da tela do seu computador. Para colocá-los dentro do seu aplicativo basta posicionar o mouse em cima do componente que você deseja usar, apertar o botão esquerdo do mouse e arrastá-lo para dentro do celular ilustrado no meio da tela [19].

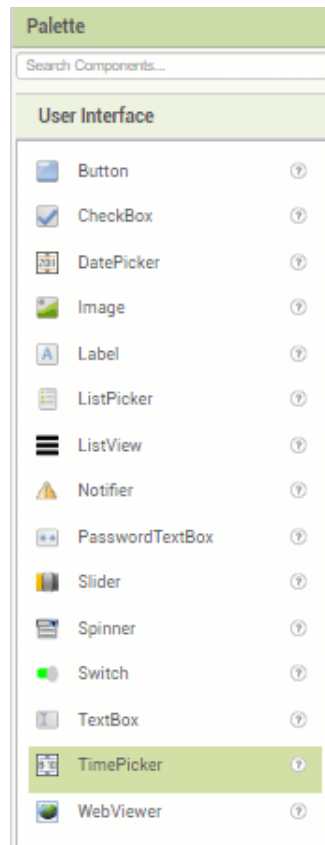


Figura 3.19: Componentes de interface do Mit app inventor.

Fonte: <https://www.makerhero.com/blog/mit-app-inventor-desenvolvendo-aplicativos-para-smartphone/>.

3.2.5 Firebase de Google

O Firebase de Google é uma plataforma digital utilizada para facilitar o desenvolvimento de aplicativos web ou móveis, de uma forma efectiva, rápida e simples. Graças às suas diversas funções, é utilizado como uma técnica de Marketing Digital, com a finalidade de aumentar a base de usuários e gerar maiores benefícios económicos. Seu principal objectivo é melhorar o rendimento dos apps mediante a implementação de diversas funcionalidades que farão do aplicativo um instrumento muito mais maleável, seguro e de fácil acesso para os usuários [10].

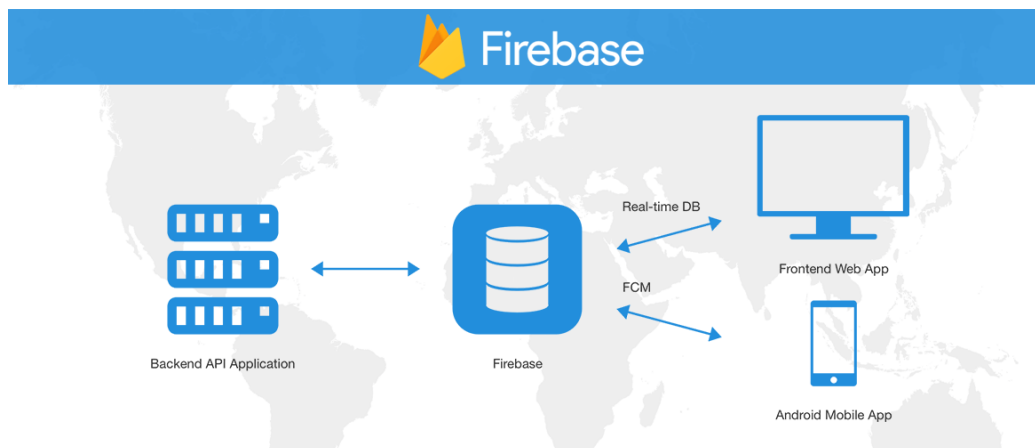


Figura 3.20: Firebase de Google.

Fonte: <https://www.ficode.co.uk/blog/firebase-realtime-database-installation-setup/>.

3.2.5.1 Serviços oferecidos pelo Firebase

Com o Firebase é possível criar aplicativos poderosos, seguros e escalonáveis, desta forma, diversos são os seus serviços disponíveis, como:

- **Cloud Firestore:** Oferece sincronização ao vivo e suporte offline, além de consultas eficientes a dados;
- **Cloud Functions:** Permite criar lógicas personalizadas que serão executadas nos aplicativos conectados ao firebase;
- **Authentication:** Com o Firebase Auth é possível gerenciar seus usuários de maneira simples e segura, oferecendo métodos de autenticação e autorização;
- **Hosting:** Permite hospedar HTML, CSS e JavaScript para seu site, além de outros ativos fornecidos pelo desenvolvedor, como gráficos, fontes e ícones;
- **Cloud Storage:** Recurso que permite o armazenamento de arquivos na nuvem para que sejam compartilhados entre os aplicativos;
- **Realtime Database:** Eficiente e de baixa latência para aplicativos para dispositivos móveis, o Realtime Database é um banco de dados com atualização em tempo real, permitindo o compartilhamento de informação entre diversos usuários de um app instantaneamente [20].

3.2.5.2 Vantagens do Firebase

Algumas praticidades e benefícios que a plataforma é capaz de proporcionar:

Dados em tempo real;

- API pronta;
- Autenticação via e-mail, Google, Facebook e GitHub;
- Segurança;
- Armazenamento de arquivos pelo Google Cloud Storage;
- Hospedagem de arquivos estáticos;
- Aplicativos altamente escaláveis;
- Sem preocupações quanto a infraestrutura [20].

3.2.5.3 Desvantagens do Firebase

Mas como nem tudo são flores, o Firebase ainda sofre com alguns pontos para melhoria, por exemplo:

- Capacidade limitada de consultas devido ao modelo de fluxo de dados do firebase;
- Os modelos de dados relacionais tradicionais não são aplicáveis ao NoSQL, seus chops SQL não serão transferidos;
- Sem instalação local;
- Não é open-source e também não possui servidores dedicados e suporte empresarial;
- Os preços podem ser difíceis de serem previstos e limitar custos pode ser incômodo [20].

4 IMPLEMENTAÇÃO DO PROTÓTIPO

Neste capítulo são apresentados tópicos fundamentais para compreensão geral da implementação do projecto, e definidas as etapas necessárias para isso, que são:

- Modelagem do sistema;
- Elaboração do código fonte para o microcontrolador Arduíno UNO;
- Montagem do protótipo;

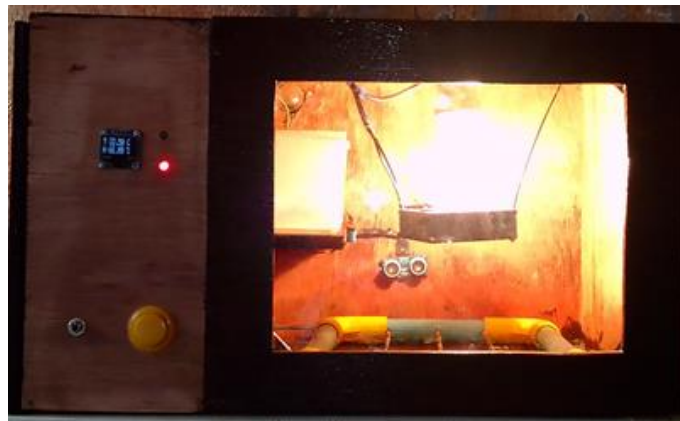


Figura 4.1: Protótipo Concluído.

Fonte: Autoria Própria

4.1 Modelagem

Primeiramente foi definido o que o projecto deveria ter de forma geral e quais os componentes necessários para fazer sua integração. Após a decisão de qual microcontrolador seria utilizado, bem como qual sensor funcionaria melhor com o sistema, e quais componentes eletrónicos os sensores activariam, foi feito um fluxograma geral do sistema, com suas funcionalidades. O projecto conta com uma placa de circuito impresso. O fluxograma da figura 4.2 ilustra o sistema.

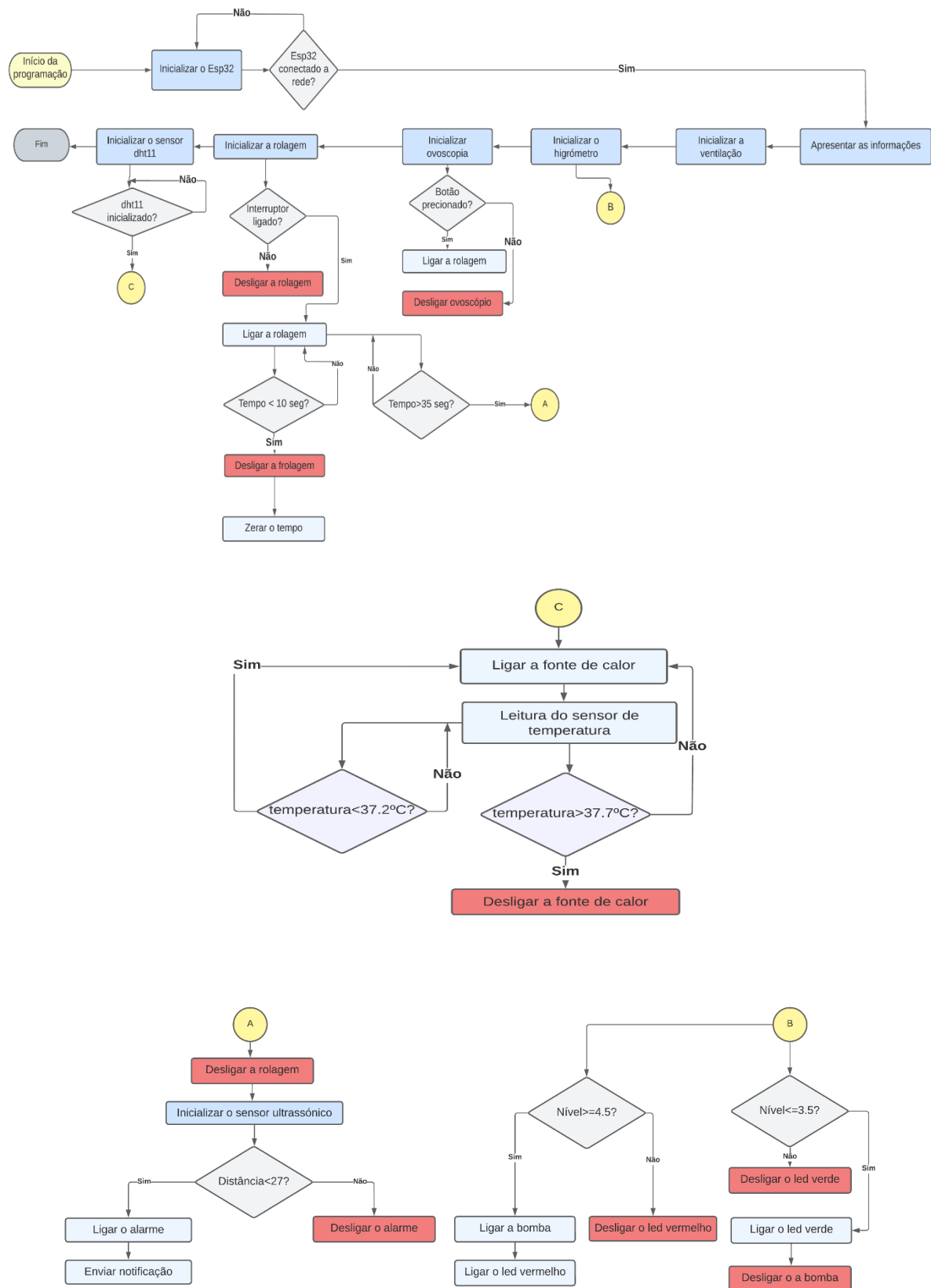


Figura 4.2: Fluxograma do sistema.
Fonte: Autoria Própria

4.2 Elaboração do Código Fonte

Após a modelagem do sistema, e a concepção de um fluxograma geral para ser seguido, foi possível a elaboração do código fonte compilado no microcontrolador. Nesta etapa foi feito um teste com o sensor DHT11, ultrassônico, e o higrômetro, compilando um código apenas para testar seu funcionamento, justamente com accionamento de outros componentes eletrônicos. Só após estes testes terem sido feitos com sucesso, foi feita a elaboração do código fonte para o protótipo final, com interação entre o microcontrolador e os dispositivos. Nesta etapa será apresentado o código utilizado na compilação com o protótipo final.

4.2.1 Código fonte final

```
#include "PushButton.h"
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <FirebaseESP32.h>
#include "DHT.h"
#include <Adafruit_GFX.h> //INCLUSÃO DE BIBLIOTECA
#include <Adafruit_SSD1306.h> //INCLUSÃO DE BIBLIOTECA
#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 32 // OLED display height, in pixels
#define OLED_RESET 4 // Reset pin # (or -1 if sharing Arduino reset pin)
#define SCREEN_ADDRESS 0x3C
#define FIREBASE_HOST "https://sensor-dht11-e17f0-default-rtdb.firebaseio.com/"
#define WIFI_SSID "IAMOC"
#define WIFI_PASSWORD "academiampitec"
#define FIREBASE_Authorization_key
"oRoXwYoj5XaOpZjCWC3PPB8BLqrDbpJME9bqUYkD"
const int trigPin = 5;
const int echoPin = 12;
#define SOUND_SPEED 0.034
#define CM_TO_INCH 0.393701
long duration;
float distanceCm;
```



```

float distanceInch;
#define DHTPIN 4
// CARGAS AC
#define LAMPAQ 18
#define LAMPOVOS 23
#define motor 14
// CARGAS DC
#define bomba 15
#define alerta 19
#define niv1 25
#define niv2 33
const int pinoSensor = 34;
float valorEntrada = 0.0; //VARIÁVEL PARA ARMAZENAR O VALOR DE TENSÃO DE
ENTRADA DO SENSOR
float valorMedida = 0.0; //VARIÁVEL PARA ARMAZENAR O VALOR DA TENSÃO
MEDIDA PELO SENSOR
float valorR1 = 30000.0; //VALOR DO RESISTOR 1 DO DIVISOR DE TENSÃO
float valorR2 = 7500.0; // VALOR DO RESISTOR 2 DO DIVISOR DE TENSÃO
int leituraSensor = 0; //VARIÁVEL PARA ARMAZENAR A LEITURA DO PINO
ANALÓGICO
unsigned long millisTarefa1 = millis();
unsigned long millisTarefa2 = millis ();
void verificaragua ();
#define pinBotao1 27
int x=0;
int y=0;
#define DHTTYPE DHT11 // DHT 11
//#define DHTTYPE DHT22 // DHT 22 (AM2302), AM2321
DHT dht(DHTPIN, DHTTYPE);
Adafruit_SSD1306 display (SCREEN_WIDTH, SCREEN_HEIGHT, &Wire,
OLED_RESET);
WiFiServer server (80);
FirebaseData firebaseData;

```

```

    FirebaseJson json;
    PushButton botao1(pinBotao1);
    void setup() {
        WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
        Serial.print("Connecting to ");
        Serial.print(WIFI_SSID);
        Serial.println();
        Serial.print("Connected");
        Serial.print("IP Address: ");
        Serial.println(WiFi.localIP());
        Firebase.begin(FIREBASE_HOST,FIREBASE_Authorization_key);
        pinMode(LAMPAQ,OUTPUT);
        pinMode(LAMPOVOS,OUTPUT);
        pinMode( bomba,OUTPUT);
        pinMode(alerta,OUTPUT);
        pinMode(motor,OUTPUT);
        pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
        pinMode(echoPin, INPUT);
        pinMode(niv1, OUTPUT);
        pinMode(niv2, OUTPUT);
        pinMode(pinoSensor, INPUT);
        digitalWrite(alerta,LOW);
        digitalWrite(LAMPOVOS,LOW);
        digitalWrite(niv1,LOW);
        digitalWrite(niv2,HIGH);
        Serial.begin(9600);
        // SerialBT.begin("ESP32test"); //Bluetooth device name
        //Serial.println("The device started, now you can pair it with bluetooth!");
        Serial.println(F("DHTxx test!"));
        dht.begin();
        display.begin(SSD1306_SWITCHCAPVCC, 0x3C); //INICIALIZA O DISPLAY COM
        ENDEREÇO I2C 0x3C
        display.setTextColor(WHITE); //DEFINE A COR DO TEXTO

```

```

display.setTextSize(2); //DEFINE O TAMANHO DA FONTE DO TEXTO
display.clearDisplay();
delay(1000);
display.setCursor(20,0); //POSIÇÃO EM QUE O CURSOR IRÁ FAZER A ESCRITA
display.print("ACADEMIA"); //ESCREVE O TEXTO NO DISPLAY
display.setCursor(33,17); //POSIÇÃO EM QUE O CURSOR IRÁ FAZER A ESCRITA
display.print("MPITEC");
display.display(); //EFETIVA A ESCRITA NO DISPLAY
delay(3000);
display.clearDisplay(); //LIMPA AS INFORMAÇÕES DO DISPLAY
display.setCursor(36,10); //POSIÇÃO EM QUE O CURSOR IRÁ FAZER A ESCRITA
display.print("IAMOC"); //ESCREVE O TEXTO NO DISPLAY
display.display(); //EFETIVA A ESCRITA NO DISPLAY
display.clearDisplay(); //LIMPA AS INFORMAÇÕES DO DISPLAY
delay(3000);
display.clearDisplay();
float h = dht.readHumidity();
// Read temperature as Celsius (the default)
float temp = dht.readTemperature();
display.setCursor(0,0); //POSIÇÃO EM QUE O CURSOR IRÁ FAZER A ESCRITA
display.print("T:"); //ESCREVE O TEXTO NO DISPLAY
display.print(temp);
display.print(" ");
display.print("C");
display.setCursor(0,17); //POSIÇÃO EM QUE O CURSOR IRÁ FAZER A ESCRITA
display.print("H:"); //ESCREVE O TEXTO NO DISPLAY
display.print(h);
display.print(" ");
display.print("%");
display.display();
display.clearDisplay();
}

```

```

void loop () {
  botao1.button_loop ();
  if ((millis () - millisTarefa2) > 35000){
    digitalWrite(motor, HIGH);
    // Clears the trigPin
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(2);
    digitalWrite(trigPin, LOW);
    duration = pulseIn(echoPin, HIGH);
    distanceCm = duration * SOUND_SPEED/2;
    distanceInch = distanceCm * CM_TO_INCH;
    Serial.print("Distance (cm): ");
    Serial.println(distanceCm);
    Serial.print("Distance (inch): ");
    Serial.println(distanceInch);
    if(distanceCm<=27&& distanceCm !=0){
      Serial.println("VALOR LIDO");
      digitalWrite(alerta,HIGH);
      delay(300);
      digitalWrite(alerta,LOW);
      delay(300);
      digitalWrite(alerta,HIGH);
      delay(300);
      digitalWrite(alerta,LOW);
      delay(300);
      digitalWrite(alerta,HIGH);
      delay(300);
      digitalWrite(alerta,LOW);
      Firebase.setInt(firebaseData,"pir", 1); }
    else {
      Serial.println("valor não lido ");

```

```

digitalWrite(alerta,LOW);
Firebase.setInt(firebaseData,"pir", 0);
}}
float h = dht.readHumidity();
float temp = dht.readTemperature();
if (isnan(h) || isnan(temp)) {
Serial.println(F("Failed to read from DHT sensor!"));
return;
}
Serial.print(F("Humidity: "));
Serial.print(h);
Serial.println(("°"));
Serial.print(F("Temperature: "));
Serial.print(temp);
Serial.println("°C");
display.setCursor(0,0); //POSIÇÃO EM QUE O CURSOR IRÁ FAZER A ESCRITA
display.print("T:"); //ESCREVE O TEXTO NO DISPLAY
display.print(temp);
display.print(" ");
display.print("C");
display.setCursor(0,17); //POSIÇÃO EM QUE O CURSOR IRÁ FAZER A ESCRITA
display.print("H:"); //ESCREVE O TEXTO NO DISPLAY
display.print(h);
display.print(" ");
display.print("°");
display.display(); //EFETIVA A ESCRITA NO DISPLAY
display.clearDisplay();
Firebase.setFloat (firebaseData,"temp",temp);
Firebase.setFloat (firebaseData,"hum",h);
if(temp<37.2){
digitalWrite(LAMPAQ,LOW);
}
if(temp>37.7){

```

```

digitalWrite(LAMPAQ,HIGH);
}
if (Firebase.get(firebaseData,"/BT")) {
if (firebaseData.dataType() == "string") {
String De1 = firebaseData.stringData();
if (De1=="1"){
digitalWrite(LAMPOVOS,HIGH); //Device1 is ON
}
else if (De1=="0"){
digitalWrite(LAMPOVOS,LOW); //Device1 if OFF
}}}
if(botao1.pressed()){
x++;
if(x==1){
Serial.println("APERTADO COM SUCESSO");
digitalWrite(LAMPOVOS,HIGH);
Firebase.setInt(firebaseData,"LED",1);
Firebase.setInt(firebaseData,"BT",0);
}
if(x==2){
digitalWrite(LAMPOVOS,LOW);
Firebase.setInt(firebaseData,"LED",0);
Firebase.setInt(firebaseData,"BT",1);
x=0;
}}
leituraSensor = analogRead(pinoSensor); //FAZ A LEITURA DO PINO ANALÓGICO E
ARMAZENA NA VARIÁVEL O VALOR LIDO
valorEntrada = (leituraSensor * 5.0) / 4095.0;
valorMedida = valorEntrada;
Serial.print("Valor medido: "); //IMPRIME O TEXTO NA SERIAL
Serial.println(valorMedida);
if( valorMedida>=4.5){
Serial.print("Valor medido: "); //IMPRIME O TEXTO NA SERIAL

```

```

Serial.println(valorMedida);
digitalWrite(niv1,HIGH);
// digitalWrite(bomba,LOW);
Firebase.setInt(firebaseData,"niv1",0);
}
else{
Firebase.setInt(firebaseData,"niv1",1);
digitalWrite(niv1,LOW);
}
if(valorMedida<=3.5){
Serial.print("Valor medido: "); //IMPRIME O TEXTO NA SERIAL
Serial.println(valorMedida);
digitalWrite(niv2,LOW);
Firebase.setInt(firebaseData,"niv2",1);
}
else{
// digitalWrite(bomba,HIGH);
digitalWrite(niv2,HIGH);
Firebase.setInt(firebaseData,"niv2",0);
}
tarefa1();
}
void tarefa1(){
// Verifica se já passou 4000 milisegundos
if((millis() - millisTarefa1) < 10000){
// Acende o led do pino 7
digitalWrite(motor, LOW);}
else{
digitalWrite(motor, HIGH);}
// Verifica se já passou 400 milisegundos
if((millis() - millisTarefa1) > 15000){
millisTarefa1 = millis();
}}

```

4.3 Montagem do protótipo

A montagem do protótipo foi feita em uma caixa de madeira. Tendo em vista que o protótipo é alimentado na rede eléctrica de 220V, ou por um cabo USB conectado a um computador, mostrou-se necessário que o protótipo tivesse uma folga dentro da caixa, para fácil movimento da rolagem dos ovos.

A caixa utilizada tem 20cm x 12cm x 3cm, e foi utilizado calha a fim de permitir a passagem dos condutores.

4.3.1 Simulações

A primeira simulação feita, foi apenas com um accionamento do sensor DHT11. Essa simulação serviu principalmente para a verificação se os componentes estavam funcionando. Além disso, a programação foi testada várias vezes, e o código fonte foi alterado em alguns momentos até conseguir o funcionamento correto, e do accionamento das cargas no momento correto.



Figura 4.3: Esp32 com Sensor DHT11 e Display Oled

Fonte: <https://how2electronics.com/micropython-esp32-with-dht11-humidity-temperature-sensor/>.

4.3.2 Problemas Encontrados

No início dos testes, tive problemas ao detectar a presença dos pintos no interior da incubadora usando o sensor pir, o problema estava na sensibilidade do sensor que levou a um atraso no cronograma de montagem e teste de hardware. A dificuldade foi vencida substituindo o sensor pir por sensor ultrassônico.

Ao interligar o app com o firebase tive muitas dificuldades em receber os dados de temperatura e humidade devido o link entre o Mit app inventor com Firebase.

4.4 Custos para a construção do projeto

A aquisição dos componentes eletrônicos para o projecto foi realizada e foi fundamental para o projecto. Abaixo na tabela 4.1 detalha os componentes e os seus devidos custos.

Tabela 4.1: Revisão de custos
Fonte: Autoria Própria

Item	Custo
Esp32	14.900 kz
Minibomba	7000 kz
Sensor Ultrassônico	3500 kz
Sensor dht11	3000 kz
Placa PCB de 9x15cm Protoboard	2500 kz
Módulo relé de 4 canais	4000 kz
Fonte de alimentação de 12V	7200 kz
Contra-Placado	8000 kz
Interruptor (Toggle) de 2 posições 3 pinos	560 kz
Regulador LM7806	520 kz
2 Capacitores	130 kz
Jumper Wire de 30cm 40p (F-F, M-F, M-M)	6000 kz
Display Oled	7000 kz
1 Botão de pressão	1000 kz
5 leds	500 kz
5 Resistores	175 kz
1 Conector p/ Adaptador AC 2.5mm fêmea	400 kz
Uma lâmpada de 100W	200 kz
Uma lâmpada de 80W	200 kz
1 Suporte	150 kz
1 m de calha	500 kz
1 motor para rolagem dos ovos	3000 kz
1 Higrómetro	2000 kz
Buzzer	650 kz
Rolo de estanho 0.5mm 100g	4000 kz
Total	75.285 kz

CONCLUSÃO

Foi desenvolvido, neste trabalho, um protótipo de Incubadora com o objectivo de apresentar uma solução para os problemas descritos no início deste documento, como fim de um usuário ser avisado sobre o nascimento dos pintos, mesmo este estando distante da incubadora. Apesar dos problemas descritos e apresentados na sessão anterior, o objectivo foi alcançado, e o sensor ultrassônico escolhido mostrou-se eficaz para a detecção de movimento necessária ao envio de notificação.

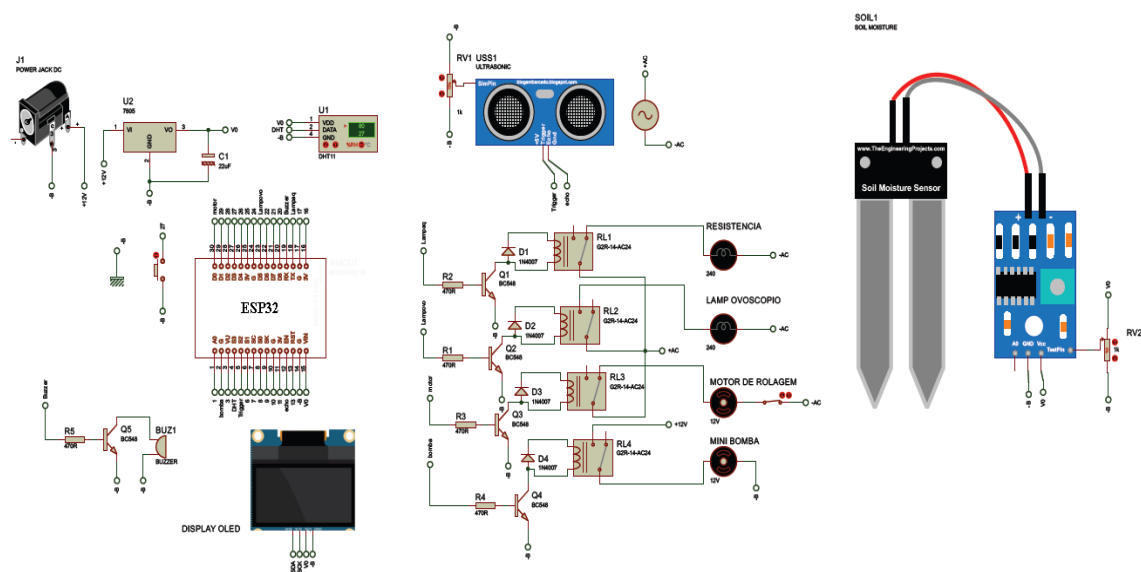
Contudo, os resultados obtidos e apresentados por esse projecto, cumprem com o proposto e planeado desde o início, e o protótipo encontra-se em funcionamento.

REFERÊNCIAS BIBLIOGRÁFICA

1. MORA, LEONARDO ALVARADO. PROCESSO DE INCUBAÇÃO ARTIFICIAL DE OVOS:. CAMPINAS : s.n., 2008. pp. 4,5.
2. Eva Kok, Janna de Feyter, Adri Vink, Farzin Wafadar, [ed.]. Melhoria da incubação de. 2011. pp. 15-29. 978-90-8573-116-0.
3. Cobb. Guia de Manejo de Incubação COBB. 2008. p. 34.
4. [Citação: 10 de Abril de 2023.] <https://how2electronics.com/micropython-esp32-with-dht11-humidity-temperature-sensor/>.
5. Treiber., Rafael Gonçalves. CHOCADEIRA DE OVOS MONITORADA VIA CELULAR. 2018. pp. 1-5.
6. Figueiredo, Aymam Cobo de. infoescola. infoescola. [Online] 2014. [Citação: 19 de Maio de 2021.] <https://www.infoescola.com/aves/galinha/>.
7. Júlio César Bertolucci Murad, Bruno Ceolin da Sila. ANIMAIS DE PEQUENO PORTE I. Brasília-DF : NT EDITORA, 2014. pp. 9,10.
8. [Citação: 17 de Maio de 2021.] <https://www.infoescola.com/aves/galinha/>.
9. freixiovo. freixiovo. [Online] 2019. [Citação: 18 de Julho de 2021.] <https://freixiovo.pt/empresa>.
10. 21 de Agosto de 2019. [Citação: 22 de Janeiro de 2022.] <https://rockcontent.com/br/blog/firebase/>.
11. 31 de Outubro de 2018. [Citação: 28 de Janeiro de 2021.] <https://curtocircuito.com.br/blog/Categoria%20IoT/conhecendo-esp32>.
12. 6 de janeiro de 2021. [Citação: 23 de junho de 2022.] <https://blog.eletrogate.com/guia-completo-do-display-oled-parte-1-o-que-e-como-funciona-2/>.
13. [Citação: 26 de Agosto de 2021.] <https://www.electrofun.pt/sensores-arduino/sensor-temperatura-humidade-dht11>.
14. 7 de abril de 2020. [Citação: 12 de outubro de 2021.] <https://www.usinainfo.com.br/blog/projeto-chocadeira-arduino-egg-incubadora-autonoma-para-ovos-de-galinha/>.

15. [Citação: 2 de Novembro de 2021.] <https://blog.fornell.com.br/2020/01/14/sensor-ultrassonico-hc-sr04/>.
16. [Citação: 18 de Novembro de 2021.] <https://www.makerhero.com/produto/sensor-de-umidade-do-solo-higrometro/>.
17. [Citação: 28 de Novembro de 2021.] <https://www.huinfinito.com.br/motores/1342-mini-bomba-de-agua-12v-rs385-pulverizacao.html>.
18. [Citação: 2 de Dezembro de 2021.] <https://www.techtudo.com.br/tudo-sobre/google-app-inventor/>.
19. [Citação: 4 de Maio de 2023.] <https://www.makerhero.com/blog/mit-app-inventor-desenvolvendo-aplicativos-para-smartphone/>.
20. 01 de Novembro de 2021. [Citação: 22 de Dezembro de 2021.] <https://www.remissaonline.com.br/blog/firebase-descubra-para-que-serve-como-funciona-e-como-usar/>.
21. [Citação: 01 de janeiro de 2022] <https://www.todamateria.com.br/velocidade-do-som/>

APÊNDICE A - ESQUEMA ELÉCTRICO DO PROJECTO



APÊNDICE B – CÓDIGO FONTE TESTE DO SENSOR DHT11

```
#include "DHT.h"

#define DHTPIN 4    // Digital pin connected to the DHT sensor
#define LED 23

#define DHTTYPE DHT11 // DHT 11 (AM2302), AM2321
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  Serial.println(F("DHTxx test!"));
  pinMode(LED,OUTPUT);
  dht.begin();
  digitalWrite(LED,LOW);
}

void loop() {
  // Wait a few seconds between measurements.
  delay(2000);
  float h = dht.readHumidity();
  float t = dht.readTemperature();
  if (isnan(h) || isnan(t) || isnan(f)) { Serial.println(F("Failed to read from DHT sensor!"));
  return; }
  Serial.print(h);
  Serial.print("°C ");
  Serial.print(t);
  Serial.print("°C ");
  if(t>32){
    digitalWrite(LED,HIGH);  }
  else{ digitalWrite(LED,LOW);
  }}
}
```

APÊNDICE C – CÓDIGO FONTE TESTE DO SENSOR ULTRASSÔNICO

```
const int trigPin = 5;
const int echoPin = 12;
//define sound speed in cm/uS
#define SOUND_SPEED 0.034
#define CM_TO_INCH 0.393701
long duration;
float distanceCm;
float distanceInch;
void setup() { Serial.begin(9600);
pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin, INPUT);
}
void loop() {
digitalWrite(trigPin, LOW);
delayMicroseconds(2);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
distanceCm = duration * SOUND_SPEED/2;
distanceInch = distanceCm * CM_TO_INCH;
Serial.print("Distance (cm): ");
Serial.println(distanceCm);
Serial.print("Distance (inch): ");
Serial.println(distanceInch);
Serial.println(distanceCm);
delay(1000);
}
```

APÊNDICE D – CÓDIGO FONTE TESTE DO HIGRÓMETRO

```
const int pinoSensor = A0; //PINO ANALÓGICO EM QUE O SENSOR ESTÁ
CONECTADO
float valorEntrada = 0.0; //VARIÁVEL PARA ARMAZENAR O VALOR DE TENSÃO DE
ENTRADA DO SENSOR
float valorMedida = 0.0; //VARIÁVEL PARA ARMAZENAR O VALOR DA TENSÃO
MEDIDA PELO SENSOR
float valorR1 = 30000.0; //VALOR DO RESISTOR 1 DO DIVISOR DE TENSÃO
float valorR2 = 7500.0; // VALOR DO RESISTOR 2 DO DIVISOR DE TENSÃO
int leituraSensor = 0; //VARIÁVEL PARA ARMAZENAR A LEITURA DO PINO
ANALÓGICO

void setup(){
pinMode(pinoSensor, INPUT); //DEFINE O PINO COMO ENTRADA
Serial.begin(9600); //INICIALIZA A SERIAL
}

void loop(){
leituraSensor = analogRead(pinoSensor); //FAZ A LEITURA DO PINO A      ANALÓGICO
E ARMAZENA NA VARIÁVEL O VALOR LIDO
valorEntrada = (leituraSensor * 5.0) / 4095.0; //VARIÁVEL RECEBE O RESULTADO DO
CÁLCULO
valorMedida = valorEntrada;
Serial.print("Valor medido: "); //IMPRIME O TEXTO NA SERIAL
Serial.println(valorMedida);
if( valorMedida>=4.5){
Serial.print("Valor medido: "); //IMPRIME O TEXTO NA SERIAL
Serial.println(valorMedida);
digitalWrite(niv1,HIGH);
// digitalWrite(bomba,LOW);
}
else{
digitalWrite(niv1,LOW);
}
```



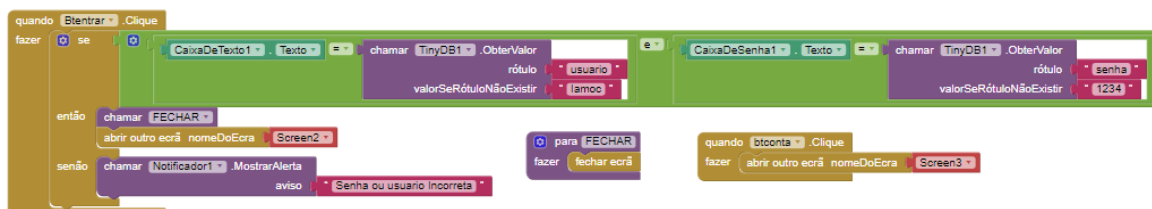
```
if(valorMedida<=3.5){  
  Serial.print("Valor medido: "); //IMPRIME O TEXTO NA SERIAL  
  Serial.println(valorMedida);  
  digitalWrite(niv2,LOW);  
}  
else{  
  // digitalWrite(bomba,HIGH);  
  digitalWrite(niv2,HIGH);  } }
```

APÊNDICE E – REALTIME DATABASE

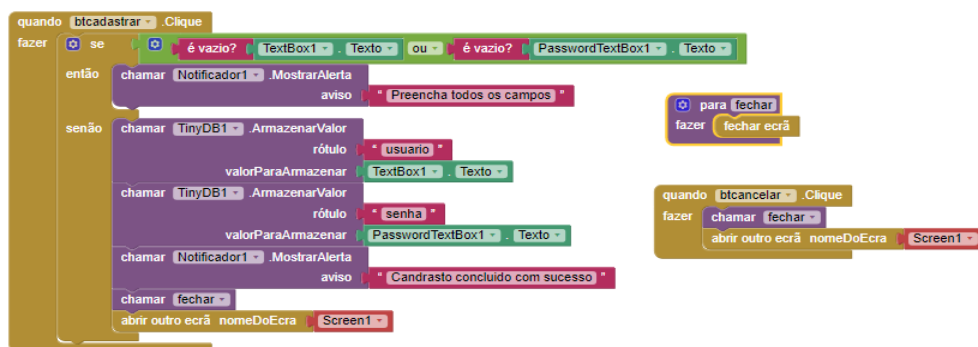
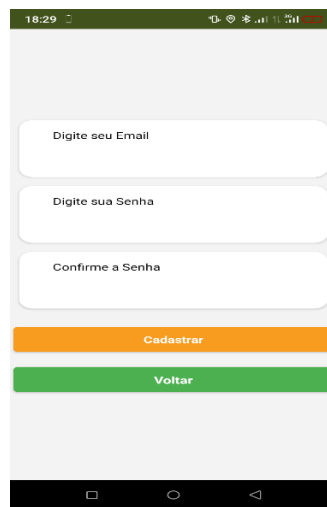


APÊNDICE F – TELA E CÓDIGO FONTE NO APP INVENTOR

Tela de login e Código



Tela da alteração do password e Código



Tela de visualização das informações

