



UNIVERSIDADE DE TAUBATÉ
Autarquia Municipal de Regime Especial
pelo Dec. Fed. nº 78.924/76
Recredenciada Reconhecida pelo CEE/SP
CNPJ 45.176.153/0001-22

Departamento de Engenharia Elétrica
Rua Daniel Danelli s/nº Jardim Morumbi
Taubaté-Sp 12060-440
Tel.: (12) 3625-4190
e-mail: eng.eletrica@unitau.br

MISTURADOR DE PRODUTOS QUIMICOS PARA CAIXAS DE GORDURA

Taubaté - SP
2018



UNIVERSIDADE DE TAUBATÉ
Autarquia Municipal de Regime Especial
pelo Dec. Fed. nº 78.924/76
Recredenciada Reconhecida pelo CEE/SP
CNPJ 45.176.153/0001-22

Departamento de Engenharia Elétrica
Rua Daniel Danelli s/nº Jardim Morumbi
Taubaté-Sp 12060-440
Tel.: (12) 3625-4190
e-mail: eng.eletrica@unitau.br

DIEGO GONÇALVES CHAGAS

MISTURADOR DE PRODUTOS QUIMICO PARA CAIXAS DE GORDURA

Diego Gonçalves Chagas

Misturador de Produtos Químicos Para Caixas de Gordura

Trabalho de Graduação apresentado ao Departamento de Engenharia Elétrica da Universidade de Taubaté, como parte dos requisitos para obtenção do diploma de Graduação em Engenharia Elétrica.

Orientador (a): Prof. Eng. Rubens Castilho Jr

Taubaté - SP
2018

**Ficha catalográfica elaborada pelo
SIBi – Sistema Integrado de Bibliotecas / UNITAU**

C433m Chagas, Diego Gonçalves
Misturador de produtos químicos para caixas de gordura /
Diego Gonçalves Chagas. – 2018.
50f. : il.

Monografia (Graduação) – Universidade de Taubaté,
Departamento de Engenharia Elétrica e Eletrônica, 2018.
Orientador: Prof. Esp. Rubens Castilho Junior,
Departamento de Engenharia Elétrica.

1. Misturador. 2. Arduino. 3. Meio ambiente. I. Título.

CDD- 670



Universidade de Taubaté
Autarquia Municipal de Regime Especial
pelo Dec. Fed. nº 78.924/76
Recredenciada Reconhecida pelo CEE/SP
CNPJ 45.176.153/0001-22

UNITAU

Departamento de Engenharia Elétrica
Rua Daniel Danelli s/nº Jardim Morumbi
Taubaté-Sp 12060-440
Tel.: (12) 3625-4190
e-mail: eng.eletrica@unitau.br

MISTURADOR DE PRODUTOS QUÍMICOS PARA CAIXAS DE GORDURA

DIEGO GONÇALVES CHAGAS

ESTE TRABALHO DE GRADUAÇÃO FOI JULGADO ADEQUADO COMO PARTE DO REQUISITO PARA A OBTENÇÃO DO DIPLOMA DE “GRADUADO EM ENGENHARIA ELÉTRICA”

BANCA EXAMINADORA:



Prof. RUBENS CASTILHO JR.
Orientador/UNITAU-DEE



Prof. ME. EDER SALIM MINHOTO
UNITAU-DEE



Eng. CARLOS HENRIQUE SILVA MOURA
Membro Externo

Dezembro de 2018

AGRADECIMENTOS

Primeiro gostaria de agradecer a Deus, pela oportunidade de chegar até aqui, pois sem ele nada seria possível.

Agradecer o apoio fundamental da minha família pela qual me deu força, coragem, compreensão, incentivo nas horas que pensei em desistir e nos momentos mais difíceis, que foram importantes para que eu pudesse entender que o futuro é feito a partir da constante dedicação e esforço do presente.

Aos docentes do curso de Engenharia Elétrica da Universidade de Taubaté, pelo conhecimento ensinado aos longos destes 5 anos.

Ao professor/orientador Rubens Castilho pela oportunidade de ter me apresentado a ideia do projeto, pelo incentivo e dedicação, correções, suporte a elaboração desse trabalho.

Aos meus amigos e colegas, nos quais fiz ao longo deste período, que foram essenciais em todas as etapas da graduação.

E a todos que direta ou indiretamente fizeram parte da minha formação acadêmica, o meu muito obrigado.

“Inovação distingue o líder de um seguidor.”

Steve Jobs

RESUMO

Neste trabalho foi motivado pelo aumento da utilização de resíduos(óleo) no dia a dia somado com preservação do meio ambiente, fator crucial nos dias de hoje. Outra motivação foi grande facilidade de encontrar módulos eletrônicos fabricados e comercializados ao redor do mundo de baixo custo. A união desses dois fatores favoreceu para a criação do protótipo automatizado. Neste documento é detalhado a criação de um misturador químico, controlado através do Arduino, responsável por realizar a mistura de um produto químico no qual ao ser inserido na caixa de gordura realiza a eliminação quase total do óleo antes de ser entregue a rede de esgoto. O sistema foi construído através da programação C/C++ na plataforma Arduino para controlar diversos atuadores no projeto. As formas de comunicação entre os módulos utilizados juntamente com o desenvolvimento da programação e montagem geral, atenderam os requisitos da proposta inicial, no qual nos permitiu a conclusão com êxito do projeto

PALAVRAS-CHAVE: Palavras-Chave: Misturador, Arduino, Meio Ambiente

ABSTRACT

In this work was motivated by the increase of the use of waste (oil) in the day to day added with preservation of the environment, crucial factor in the present day. Another motivation was great ease of finding electronic modules manufactured and marketed around the world at low cost. The union of these two factors favored the creation of the automated prototype. This document details the creation of a chemical mixer, controlled through the Arduino, responsible for mixing a chemical in which when inserted into the grease box performs the almost complete removal of the oil before being delivered to the sewer. The system was built through C / C ++ programming on the Arduino platform to control several actuators in the project. The forms of communication between the modules used together with the development of the programming and general assembly, met the requirements of the initial proposal, which allowed us to successfully complete the project

KEYWORDS: Mixer, Arduino, Environment.

LISTA DE FIGURAS

Figura 1 - Alimentação da placa Arduino Mega	16
Figura 2 - Circuito regulador de tensão	17
Figura 3 - Circuito de proteção da USB da placa Arduino MEGA 2560	17
Figura 4 - Circuito de seleção de fonte na Arduino MEGA	18
Figura 5 - Conectores de alimentação para conexão de shields e módulos na placa Arduino MEGA	19
Figura 6 - Conversor USB-serial com ATmega16u2	20
Figura 7 - Microcontrolador ATmega2560	21
Figura 8 - Pinos de entradas Analógicas	22
Figura 9 - Dimensões da Arduino Mega 2560	22
Figura 10 - Resumo de recursos da Arduino MEGA 2560	22
Figura 11 - Esquemático de uma conexão de um rele	24
Figura 12 - Moto Bomba	27
Figura 13 - Válvula Solenoide Eletherm	28
Figura 14 - Motor Unidade CD – DVD	29
Figura 15 - Display Shield 16x2 com Teclado	29
Figura 16 - Sistema de Operação do Arduino na Plataforma	31
Figura 17 - Corte para montagem da estrutura	32
Figura 18 - Esquemático pronto para instalação eletrônica	33
Figura 19 - Parte Superior do Projeto	34
Figura 20 - Caixa de Mistura	34
Figura 21 - Saída da Mistura para local destinado	34
Figura 22 - Conexão com a Válvula Solenoide	35
Figura 23 - Conexão de Produto químico à armazenagem	36
Figura 24 - Conexão entre misturador e saída para caixa de Gordura	36
Figura 25 - Etapa 1 “Pouco Resíduo”	39
Figura 26 - Etapa 2 “Meio Resíduo”	39
Figura 27 - Etapa 2 “Muito Resíduo”	40

LISTA DE TABELAS

Tabela 1 - Tabela de Dados da bobina (AT20 °C).....	25
Tabela 2 - Avaliação de Dados.....	25
Tabela 3 - Tabela de Desempenho (no valor inicial).....	26
Tabela 4 - Coluna x Vazão	27
Tabela 5 - Especificação Técnica da Moto Bomba	28

LISTA DE ABREVIATURAS E SIGLAS

PWM	PWM - Pulse Width Modulation
USB	Universal Serial Bus
CI	Circuito Integrado
DC	Corrente Direta
AC	Corrente Alteranada
LED	Light Emitting Diode)
ICSP	Circuit Serial Programming
IDE	Ambiente de Desenvolvimento Integrado
KB	Quilobyte
EEPROM	Electrically-Erasable Programmable Read-Only Memory
DVD	Digital Versatile Disc
SPI	Serial Peripheral Interface
MIPS	Milhões de Instruções Por Segundo
GND	Terra / Neutro
PIC	Programmable Interrupt Controller
ESP	Electronic Stability program
CD	Compact Disc
VEE	Tensão de alimentação negativa
LCD	Liquid Crystal display
VCC	Tensão de alimentação positiva

SUMÁRIO

1. INTRODUÇÃO	15
2. REVISÃO BIBIOGRAFICA	16
2.1. ARDUINO MEGA 2560	16
2.1.1. Alimentação Arduino MEGA	16
2.1.2. Comunicação USB	19
2.1.3. Microcontrolador da placa Arduino MEGA 256	20
2.1.4. Pinos de entradas e Saídas da placa Arduino MEGA 2560	21
2.1.5. Características físicas	22
2.2. ATUADORES	23
2.2.1. Rele Shield 8 Canais	24
2.2.1.1. Características	25
2.2.2. Moto Bomba	27
2.2.2.1. Especificação Técnicas	28
2.2.3. Válvula Solenoide Eletherm ³/₄	28
2.2.4. Motor Unidade CD – DVD	29
2.2.5. Display LCD 16X2 Shield com Teclado	29
3. METODOLOGIA	31
3.1. PLATAFAFORMA DE OPERAÇÃO	31
3.1.1. Sistema	31
3.1.2. Estrutura	32
3.1.2.1. Arquitetura do Misturador	32
3.1.2.2. Instalação dos recipientes.	33
3.1.2.3. Conexões	35
3.1.2.3.1. Recipiente Água x Recipiente Misturador.....	35
3.1.2.3.2. Recipiente Armazenagem Material Químico x Recipiente Misturador	35
3.1.2.3.3. Recipiente Misturador x Local para Liberação da Mistura	36
3.2. FUNCIONAMENTO	37
3.2.1. Relação de capacidade do tanque	38
3.2.1.1. Primeira Etapa – Pouco Resídua	38
3.2.1.2. Segunda Etapa – Relativamente Cheio.....	39
3.2.1.3. Terceira Etapa – Tanque Cheio	40
4. RESULTADOS	41

5. CONCLUSÃO.....	42
REFERÊNCIAS	43
APÊNDICE A - CODIGO FONTE UTILIZADO NO PROJETO.....	45

1. INTRODUÇÃO

Com o crescimento e popularização no mundo dos microcontroladores, somado a queda dos valores de componentes eletrônicos, baterias e motores, permitiram o desenvolvimento de pequenas plataformas automatizadas a custo acessível do mercado.

A implementação de uma plataforma robotizada envolve conceitos de eletrônica, programação e mecânica, que juntos trabalham de forma sincronizada, possibilitando a criação de um sistema que percebe através de sensores e age através de atuadores.

Para realizar toda essa sincronização de sensores e atuadores então empregamos um microcontrolador para controlar o funcionamento de cada dispositivo ligado a ele, que garante o funcionamento de forma sincronizada de todos os elementos nele conectados.

Em meio a todo sistema eletrônico também visamos o meio ambiental. Segundo (GIRAD,2010) "Hoje no Brasil, cada brasileiro consome cerca de 15 litros de óleo por ano, e grande parte desse óleo consumido acaba indo para rede de esgoto". Se não tratada adequadamente pode ocasionar contaminações no solo, na água, intervenções no clima e diversos fatores ligados a estrutura pública.

Mediante a isto, criamos um sistema capaz de fornecer e misturar uma solução química para uma caixa de gordura, muito utilizada em indústrias, residências e comércios. O sistema por si, é responsável por misturar toda a solução na água, e liberando então na caixa de gordura, onde as bactérias irão de forma química, dissolver 90% do óleo composto no sistema, para que só então seja liberado o material ao sistema de esgoto municipal, minimizando de forma gradativa o impacto ambiental.

Neste documento também iremos apresentar todos os detalhes envolvidos para criação do misturador automatizado, desde o microcontrolador utilizado até as ações implementadas no trabalho. Iremos descrever os materiais e a metodologia utilizada nesse projeto mostrando todos conceitos por trás da criação do sistema automatizado para mistura. Também iremos apresentar os resultados obtidos e a conclusão acerca do desenvolvimento do trabalho. Nos anexos iremos trazer, o esquemático do sistema, o esquema da plataforma de controle, o código fonte do misturador.

2. REVISÃO BIBLIOGRÁFICA

2.1. ARDUINO MEGA 2560

A placa de Arduino Mega 2560 pertence a plataforma Arduino a qual possui recursos interessante para elaboração de projeto. Ela possui 54 pinos de entrada e saídas digitais, sendo 15 que podem ser utilizadas como saída PWM. Com 16 entradas analógicas e 4 portas de comunicação serial. Esta placa também possui uma maior quantidade de memória que o Arduino Mega, sendo a melhor opção para projeto que necessita de uma grande quantidade de pinos de entrada e saída, além de memória de maior capacidade do programa.

Segundo (SOUZA,2018) a “placa Arduino MEGA 2560 é uma ótima opção para expandir seus projetos, quando há a necessidade de mais pinos ou quantidade de memória FLASH”. Possui desempenho parecido com a placa Arduino UNO, porém segundo (SOUZA,2018) “possibilitando maior quantidades de recursos, como mais entradas analógicas e saídas PWM”. Pode ser aplicada em automação residencial, robótica e em vários projetos eletrônicos que necessitem de muitos pinos digitais ou analógicos.

2.1.1. Alimentação Arduino MEGA

A alimentação pode ser realizada como pela porta USB quanto por uma alimentação externa, como uma fonte por exemplo, similar com a do Arduino Uno

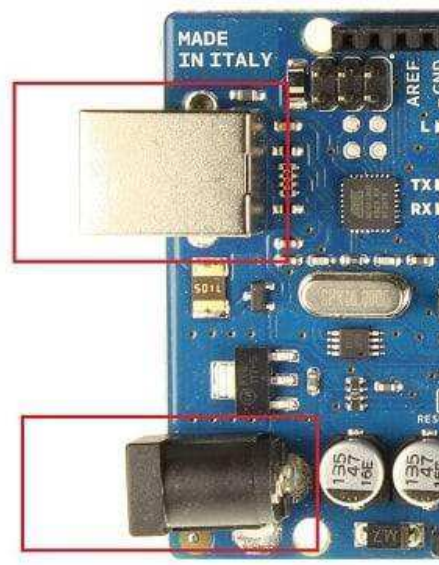


Figura 1 - Alimentação da placa Arduino Mega

A alimentação externa é realizada através de um conector Jack com positivo ao centro. A tensão da fonte externa deve estar entre 6V a 20V no máximo, porém (Thomsen, 2018) afirma que afirma que “se for alimentada com uma tensão inferior a 7V, a tensão de funcionamento da placa é de 5V, podendo ficar instável quando alimentada com uma tensão superior a 12V”. O regulador de tensão da placa pode sobreaquecer e vir danificá-la. Por isso é recomendado que a fonte externa possua tensões de 7V a 12V. O CI responsável pela regulação de tensão é o OnSemi NCP1117:

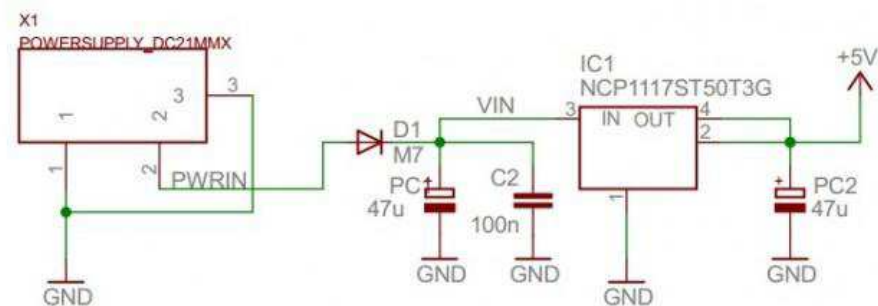


Figura 2 - Circuito regulador de tensão

Ao ser conectado o cabo USB ao computador, a tensão não precisa ser estabilizada pelo regulador de tensão, desta forma, a placa começa a ser alimentada diretamente pelo USB.

O circuito USB possui componentes que protegem sua entrada no computador no caso de algumas normalidades na transmissão de energia.

Na figura 3 é mostrado o circuito de proteção USB da placa Arduino MEGA 2560.

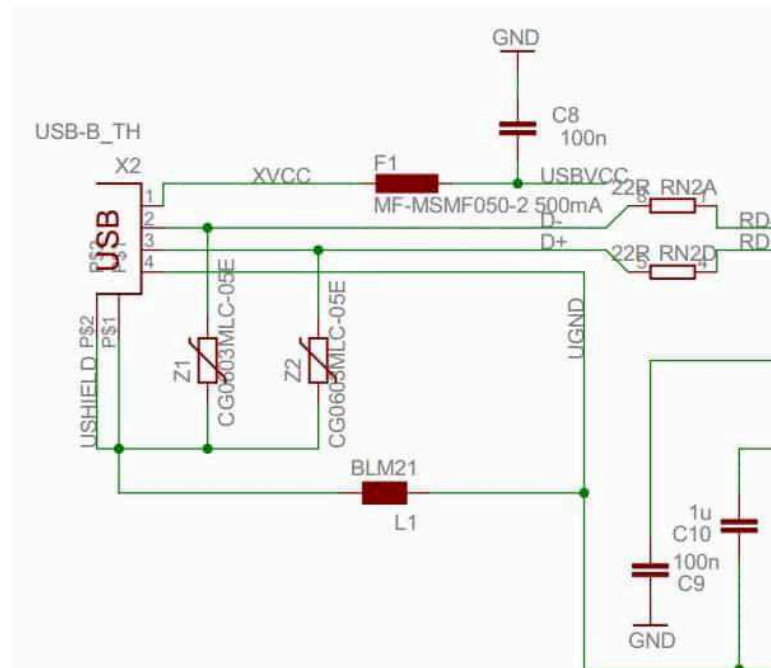


Figura 3 - Circuito de proteção da USB da placa Arduino MEGA 2560

Os varistores (Z1 e Z2) consegue suportar picos elevados. Seria preferível, ao (Thomsen, 2018) afirma que afirma que “invés de varistores, fossem conectados diodos supressores de ESD que possuem capacitância baixa”, já que estão ligados a pinos de rápida comunicação, porém o circuito funciona de forma eficiente.

Com resistores de 22 Ohms (RN2A e RN2D), limita-se uma corrente resultante de uma possível descarga elétrica causada pelo contato de um usuário com o conector USB, protegendo assim os pintos do microcontrolador.

O fusível F1 de 500mA, impede a queima da porta USB do computador, caso aconteça algum erro de projeto ou falha no circuito, que ultrapasse a corrente de 500mA quando estiver conectada ao computador.

O ferrite L1 foi colocado no circuito para que impeça a entrada de ruídos da USB externa na placa do Arduino através da terra.

(SOUZA,2018) afirma que “a placa do Arduino também conta com um circuito para comutar a alimentação automática entre tensão da fonte externa e a tensão do USB”. Caso possua uma tensão no conector DC e USB for conectada, a tensão de 5V será proveniente da fonte externa da USB e não apenas para realizar a comunicação com o computador.

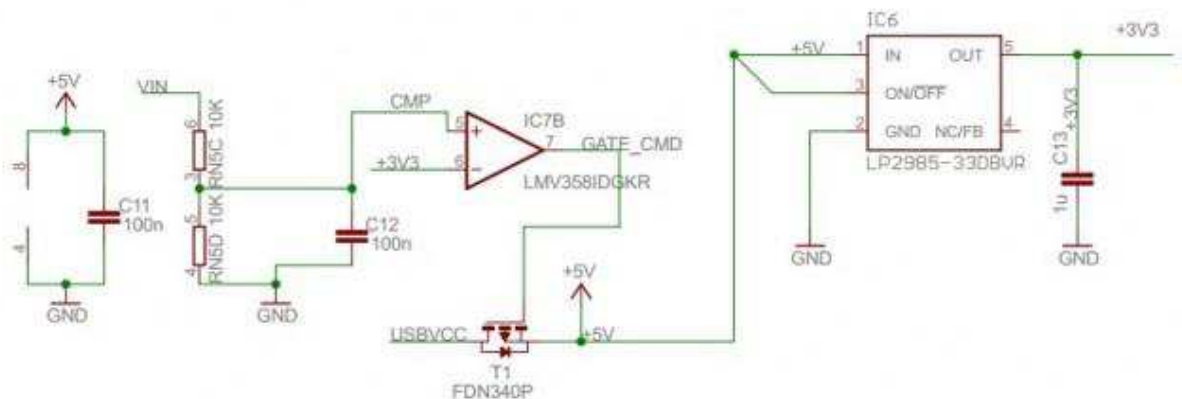


Figura 4 - Circuito de seleção de fonte na Arduino MEGA

Ao analisar o circuito acima, anotamos que existe uma placa com regulador de 3,3V modelo (U2-LP2985). Este componente é responsável por fornecer uma tensão contínua para a alimentação do Shields ou do circuito que necessite de uma tensão de 3,3V. Este regulador pode fornecer uma corrente com no máximo 50mA.

A seguir na figura 5, serão mostrados os conectores de alimentação para conexão de shields e módulos na placa Arduino.

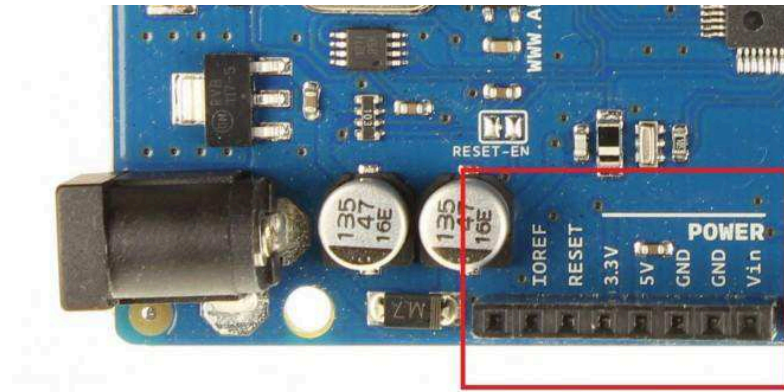


Figura 5 - Conectores de alimentação para conexão de shields e módulos na placa Arduino MEGA

IOREF - Tensão de referência para que shields possam selecionar o tipo de interface, afirma segundo (SOUZA, 2018) “assim os shields que funcionam com a placas Arduino que são alimentadas com 3,3V. podem ser adaptar para ser utilizados em 5V”.

RESET – Utilizado para RESET externo da placa quando conectado a um pino de reset no microcontrolador.

3,3 V. – Gera uma tensão de 3,3V para alimentação de shield e módulos externos com corrente máxima de 50 mA.

5 V – Gera uma tensão de 5V para alimentação de shield e módulos externos

GND - pinos de referencia, ground, terra.

VIN - pino para alimentar a placa através de shield ou bateria externa. Quando a placa é alimentada com o conector Jack a tensão da fonte estará nesse pino.

2.1.2. Comunicação USB

Para comunicação com o computador, Arduino Mega possui um microcontrolador de modelo ATMEGA16U2, conforme a figura 6.

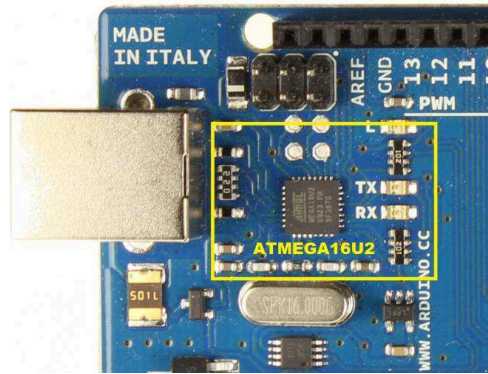


Figura 6 - Conversor USB-serial com ATmega16u2

“Este microcontrolador possibilita o upload do código binário gerado após o usuário copilar no programa. Ele possui conector ICSP para gravação de um firmware através de um programador ATMEL para que seja realizado atualizações futuras.” (ATMEL,2018)

Neste microcontrolador também estão conectados dois LEDs (TX, RX), que são devidamente controlados pelo software do microcontrolador que indica o envio e recebimento de dados da placa ao computador. Ele possui um cristal externo de 16 Mhz. Outro favor interessante que facilita a utilização desta placa, é sua conexão com o pino 13 ATMEGA16U2 ao circuito de RESET do ATMEGA2560, o qual possibilita a entrada do modo bootloader automático, quando pressionado o botão de Upload da IDE. Esta característica não estava presente nas primeiras placas fabricadas de Arduino, a qual era necessário o usuário pressionar o botão RESER antes de realizar o upload na IDE.

2.1.3. Microcontrolador da placa Arduino MEGA 256

O microcontrolador que se utiliza Arduino MEGA 2560 é o ATMEL ATmega2560, com 8 bits de arquitetura RISC. Ele também possui “256 KB de Flash 8 KB de RAM e 4 KB de EEPROM” (SHILOH, MICHAEL, 2012, p.262). Consegue operar chegando a 16 MIPS, operando em 16 MHz. Possui 4 canais de comunicação serial, 16 entradas analógicas e 15 saídas PWM, ainda possui comunicação SPI, I2C e 6 pinos de interrupções externas.



Figura 7 - Microcontrolador ATmega2560

2.1.4. Pinos de entradas e Saídas da placa Arduino MEGA 2560

Possui 54 pinos de entradas e saídas digitais que podem ser utilizadas como entrada ou saída. Os pinos operam com tensão de 5V e podem fornecer até 40 mA. Alguns desse pinos possuem funções especiais como:

Comunicação Serial - Serial 0 (RX) e 1 (TX); Serial 1: 19 (RX) e 18 (TX); Serial 2: 17 (RX) e 16 (TX); Serial 3: 15 (RX) e 14 (TX). Os pinos 0 e 1 estão conectados aos pinos do ATmega16U2 responsável pela comunicação USB.

Interrupções externas - 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2). estes pinos podem ser configurados para disparo da interrupção tanto na borda de subida ou descida, ou em níveis lógicos alto ou baixo, conforme a necessidade do projeto. Veja a função `attachInterrupt()` para mais detalhes.

PWM: os pinos 2 a 13 e 44 a 46 podem ser utilizados como saídas PWM. O sinal PWM possui 8 bits de resolução e é implementado com a função `analogWrite()`. (OLIVEIRA, 2018)

Comunicação SPI: Pinos: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS). A comunicação SPI pode ser manipulada pela função SPI library. Estes pinos estão ligados ao conector ICSP.

A Arduino Mega2560 possui 16 entradas analógicas (pinos A0 a A15), que pode ser feita a conversão com uma resolução de 10 bits, ou seja, o valor será convertido entre 0 e 1023. Por padrão a tensão de referência é conectada a 5V.

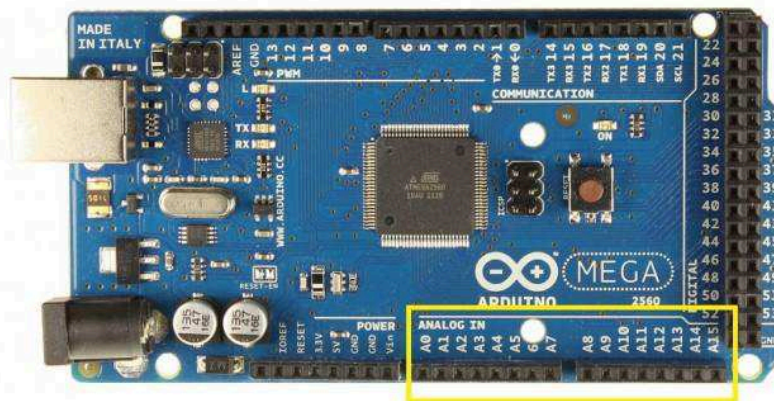


Figura 8 - Pinos de entradas Analógicas

2.1.5. Características físicas

A placa Arduino MEGA 2560 possui dimensões de 4"X 2,1".

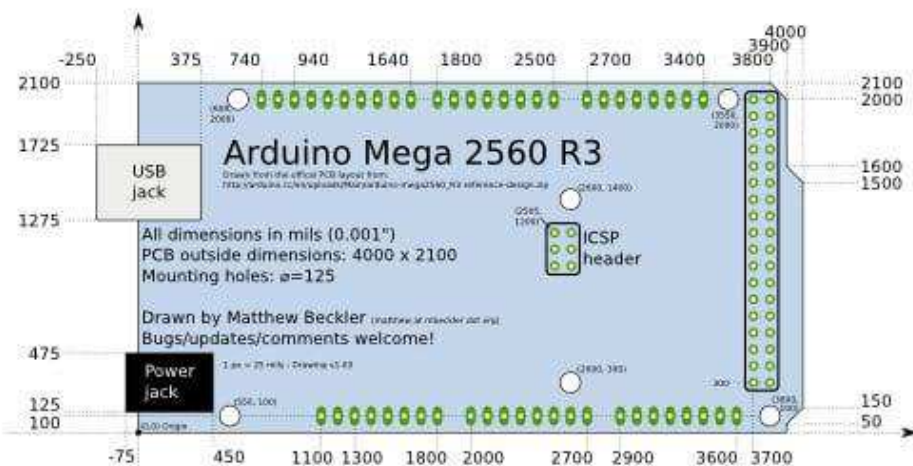


Figura 9 - Dimensões da Arduino Mega 2560

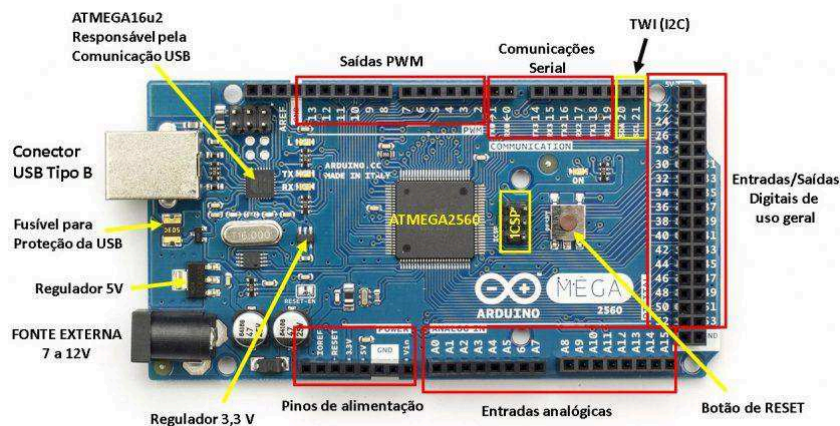


Figura 10 - Resumo de recursos da Arduino MEGA 2560

2.2. ATUADORES

Atuadores são alguns dos dispositivos eletrônicos do sistema que recebem sinais elétricos para realizar alguma ação no sistema.

Todos os atuadores funcionam em tensão diferente a 5 V, e necessitam corrente maior do que a oferecida no Arduino. Assim, o controle desses componentes é feito através de uma fonte externa de energia e reles.

Reles são interruptores eletromecânicos, que quando uma corrente elétrica percorre pelas bobinas de rele, cria um campo magnético responsável por atrair ou repulsar uma chave metálica, que fecha um circuito elétrico.

(Thomsen, 2018) afirma que “como o Arduino não oferece corrente suficiente para alimentar as bobinas do reles, a solução é utilizar um modulo com 8 canais”, que possui um transistor bipolar o qual sua base será excitada pelo rele, cujo o coletor está conectado neste mesmo rele, gerando uma corrente maior que na bobina do mesmo.

A bobina está em paralelo com o diodo. Este diodo serve também para proteção do transistor, pois quando a corrente da bobina é interrompida, ela pode gerar pico de tensão em seu terminal, que pode levar a queima do transistor. O diodo é colocado de forma que se houver algum pico de tensão no coletor do transistor, a corrente é passa por ele, e não força o transistor.

A corrente da bobina gera um campo magnético que por sua vez fecha o circuito GND de algum periférico, como neste caso o da bomba da água e o GND da válvula solenoide. O esquema de conexão de um rele, neste caso isolando apenas um canal da placa, pode ser visto na figura 13.

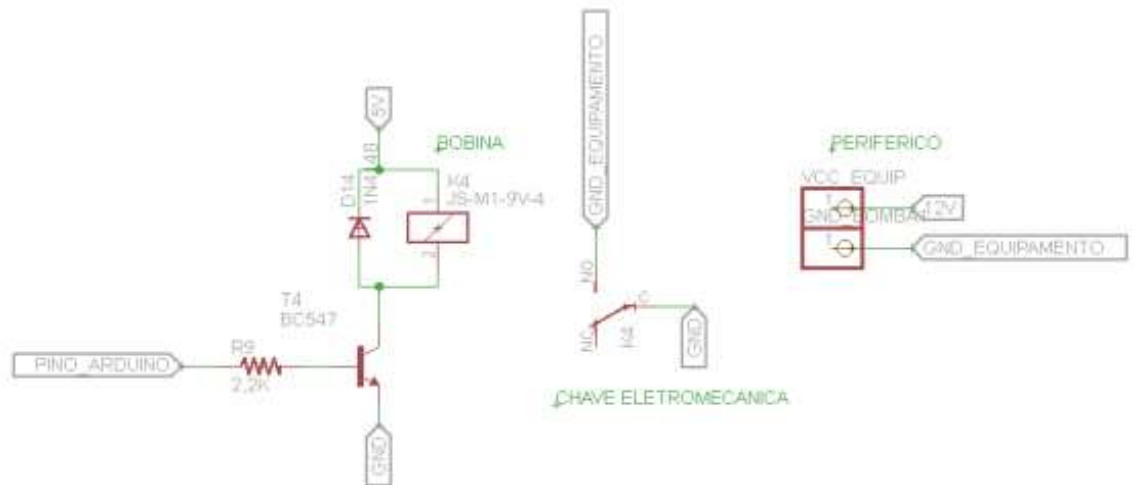


Figura 11 - Esquemático de uma conexão de um rele.

Os detalhes de cada atuador serão dados na seção a seguir.

2.2.1. Rele Shield 8 Canais

Este modulo com 8 canais operado com 5V é melhor opção para utilizarmos em um projeto, devido sua facilidade para acionamento de vários componentes em uma única placa, de forma confiável, compacta e robusta. Este rele funciona como um interruptor eletrônico, que ao aplicar tensão no terminal de entrada é acionada uma bobina que cria um campo magnético capaz de fechar ou abrir os contatos para que se possa controlar as correntes que circulam nos circuitos externos. Com isso ele se utiliza de baixa corrente para acionar seu comando e protege o controlador das correntes mais altas que circulam pelo segundo circuito.

Com este modulo de 8 canais é possível operar cargas até 220V AC, como por exemplo uma lâmpada, equipamentos eletrônicos, válvulas, motores, através do auxilio de um microcontrolador Arduino, PIC etc. Todo acionamento é possível ser realizado, sem a necessidade de construir um circuito com reles e transistores, conectores, leds e diodos.

Este modulo trabalho com o Rele modelo SRD-05VDC-SL-C, com tensão de operação de 5VDC, corrente típica de operação 15~20mA permitindo controlar cargas até 220V AC. Também possui um led indicador de status, pinagem normal aberto, normal fechado e comum. Tensão de saída (30 VDC a 1,0A) ou (250VAC a 10A) com tempo de resposta de 5~10ms

2.2.1.1. Características

Capacidade de comutação disponível por 10A apesar de design de tamanho pequeno para alta densidade P.C. borda técnica de montagem. Possui UL, CUL, TUV reconhecido. Seleção de material plástico para alta temperatura e melhor desempenho da solução química. Circuito magnético de relé simples para atender a baixo custo de produção em massa.

Coil Sensitivity	Coil Voltage Code	Nominal Voltage (VDC)	Nominal Current (mA)	Coil Resistance (Ω) $\pm 10\%$	Power Consumption (W)	Pull-In Voltage (VDC)	Drop-Out Voltage (VDC)	Max-Allowable Voltage (VDC)
SRD (High Sensitivity)	03	03	120	25	abt. 0.36W	75%Max.	10% Min.	120%
	05	05	71.4	70				
	06	06	60	100				
	09	09	40	225				
	12	12	30	400				
	24	24	15	1600				
SRD (Standard)	03	03	150	20	abt. 0.45W	75% Max.	10% Min.	110%
	05	05	89.3	55				
	06	06	75	80				
	09	09	50	180				
	12	12	37.5	320				
	24	24	18.7	1280				
	48	48	10	4500	abt. 0.51W			

Tabela 1 - Tabela de Dados da bobina (AT20 °C)

Item	Type	SRD	
		FORM C	FORM A
Contact Capacity Resistive Load ($\cos\Phi=1$)		7A 28VDC 10A 125VAC 7A 240VAC	10A 28VDC 10A 240VAC
Inductive Load ($\cos\Phi=0.4$ L/R=7msec)		3A 120VAC 3A 28VDC	5A 120VAC 5A 28VDC
Max. Allowable Voltage		250VAC/110VDC	250VAC/110VDC
Max. Allowable Power Force		800VAC/240W	1200VA/300W
Contact Material		AgCdO	AgCdO

Tabela 2 - Avaliação de Dados

Item	Type	SRD
Contact Resistance		100mΩ Max.
Operation Time		10msec Max.
Release Time		5msec Max.
Dielectric Strength		
Between coil & contact		1500VAC 50/60HZ (1 minute)
Between contacts		1000VAC 50/60HZ (1 minute)
Insulation Resistance		100 MΩ Min. (500VDC)
Max. ON/OFF Switching		
Mechanically		300 operation/min
Electrically		30 operation/min
Ambient Temperature		-25°C to +70°C
Operating Humidity		45 to 85% RH
Vibration		
Endurance		10 to 55Hz Double Amplitude 1.5mm
Error Operation		10 to 55Hz Double Amplitude 1.5mm
Shock		
Endurance		100G Min.
Error Operation		10G Min.
Life Expectancy		
Mechanically		10 ⁷ operations. Min. (no load)
Electrically		10 ⁵ operations. Min. (at rated coil voltage)
Weight		abt. 10grs.

Tabela 3 - Tabela de Desempenho (no valor inicial)

2.2.2. Moto Bomba

Neste projeto utilizou-se a moto bomba modelo S160 da Sarlobetter, conforme figura 14.



Figura 12 - Moto Bomba

Esta bomba foi projetada para funcionar como bomba de circulação, acoplada ao um filtro biológico de fundo. Silenciosa e eficaz, esta moto bomba produz intensa movimentação de água, com baixo consumo de energia elétrica. Os componentes elétricos da bomba são totalmente imersos em resina epóxi o que as tornam seguras contra os choques elétrico mesmo trabalhando continuamente submersas.

Para ser utilizada para movimentar água em desnível, haverá a queda na vazão em relação a coluna de água conforme demonstrado na tabela 4. Para determinar a vazão de água a determinamos a altura da mangueira, traçando uma reta vertical do valor da coluna de água desejada até cruzar a curva do gráfico. Podemos observar abaixo que para uma coluna de 20 cm a vazão correspondente é cerca de 125 l/h.

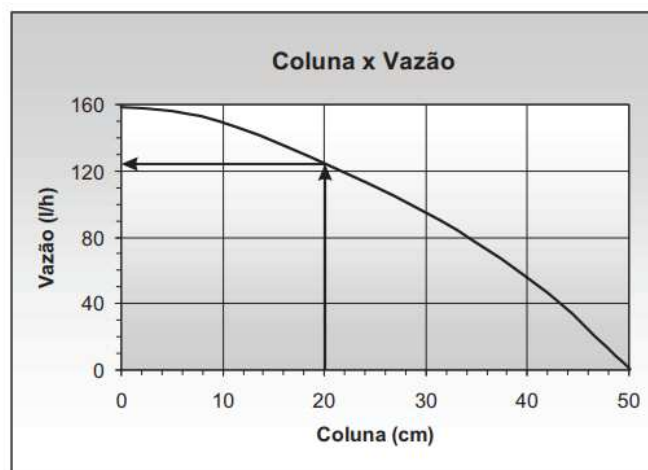


Tabela 4 - Coluna x Vazão

2.2.2.1. Especificação Técnicas

A seguir serão relacionadas às especificações técnicas da bomba, relacionadas ao modelo, voltagem, frequência, altura etc.

ESPECIFICAÇÕES TÉCNICAS

MODELO	V	Hz	W	Q max	Q mín	H max
S160	110	60	3,8	160 l/h	80 l/h	0,5 mca
	220					

Tabela 5 - Especificação Técnica da Moto Bomba

2.2.3. Válvula Solenoide Eletherm ¾

Para válvula solenoide do sistema foi utilizado um normalmente utilizada em máquinas de lavar roupas. O modelo pode ser visto na Figura 15. Assim como a bomba ela também é controlada pelo Arduino através de um rele, pois ele funciona em 12 V.



Figura 13 - Válvula Solenoide Eletherm

Este modelo de Válvula Solenoide possui simples de entrada de água ou função Vapor 220V. Foi utilizada no projeto para controle de fluxo de água com entrada rosca externa de 3/4" e Saída conexão ESPIGÃO para mangueira de 1/2" De 0,2 a 8 kgf/cm² / À 0,2kgf/cm², possui uma Vazão mínima= 7 l/min; à 8 kgf/cm²; e vazão máxima= 40 l/min. Pode ser operada com temperatura Máxima do Líquido 60°C; com uma vida Útil 50.000 operações;. A Tensão de utilização deste modelo é 220V.

2.2.4. Motor Unidade CD – DVD



Figura 14 - Motor Unidade CD – DVD

Para realizarmos a mistura dos produtos utilizamos últimos um motor de corrente contínua que é empregado nas unidades de DVD e CD. Os motores possuem uma tensão máxima de 5.9V para operarem e um eixo de 8mm. Esse motor é um dispositivo que operam aproveitando as forças de atração e repulsão geradas por eletroímãs e ímãs permanentes. A velocidade de rotação deste motor não depende de nada a não ser da força que o rotor tenha de fazer para girar. Desta forma, os pequenos motores de corrente contínua têm uma velocidade muito maior quando giram livremente do que quando girar fazendo algum tipo de esforço.

2.2.5. Display LCD 16X2 Shield com Teclado

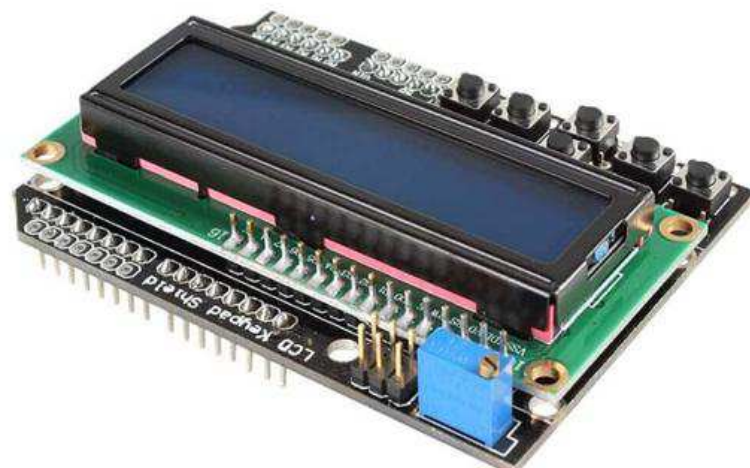


Figura 15 - Display Shield 16x2 com Teclado

O arduino LCD Keypad Shield é desenvolvido para placas compatíveis com Arduino, para fornecer uma interface amigável que permite aos usuários passar pelo menu, fazer seleções, etc. Ele consiste em um LCD retroiluminado de 1602 caracteres brancos. O teclado consiste em 5 teclas - selecione, para cima, para a direita, para baixo e para a esquerda. Para salvar os pinos digitais de entrada / saída, a interface do teclado usa apenas um canal ADC. O valor da chave é lido através de um divisor de tensão de 5 estágios.

Este display shield possui um controlador HD44780. Este display shield possui um controlador HD44780 com tensão de operação até 5V DC, com corrente de operação de 26mA e uma quantidade máxima de 6 botões, sendo um teclado com 5 botões para operação e traz o botão de reset do Arduino para o topo do escudo

Compatível com placas compatíveis Arduino e exibe até 16 caracteres por 2 linhas. Integrou o Controlador LCD de matriz matricial SPLC780D 16 COMs x 40 SEGs e o driver.

Ele é utilizado em projetos com Arduino ou outras plataformas microcontroladas em que seja necessário exibir Informações em tempo real, tais como temperatura, umidade, valor de tensão ou corrente, mensagens de erro e etc. Os botões disponíveis no shield possibilitam que opções sejam selecionadas para que a plataforma embarcada execute determinada ação.

3. METODOLOGIA

A plataforma automatizada é formada por uma estrutura de três andares e três recipientes para armazenar e misturar os materiais químicos a serem trabalhados. A plataforma também possui um Arduino Mega 2560, uma placa de rele, uma válvula solenoide, duas motos bomba utilizada em aquários e um motor utilizado geralmente em DVD, responsável pela mistura dos produtos.

A seguir serão apresentados e descritos com mais detalhes o desenvolvimento e funcionalidades do projeto.

3.1. PLATAFAFORMA DE OPERAÇÃO

3.1.1. Sistema

O sistema foi construído utilizando um microcontrolador Mega, responsável por coordenar todos os eventos operacionais no produto. A placa do arduino é acionada através do datashield keyboard, que após a acionar algum dos botões, o produto inicia a operação, enviando a informação ao arduino, que logo após envia a informação ao rele, que atraca a cada operação dos atuadores. Na figura 18 temos a estrutura desenvolvida em torno do Arduino.

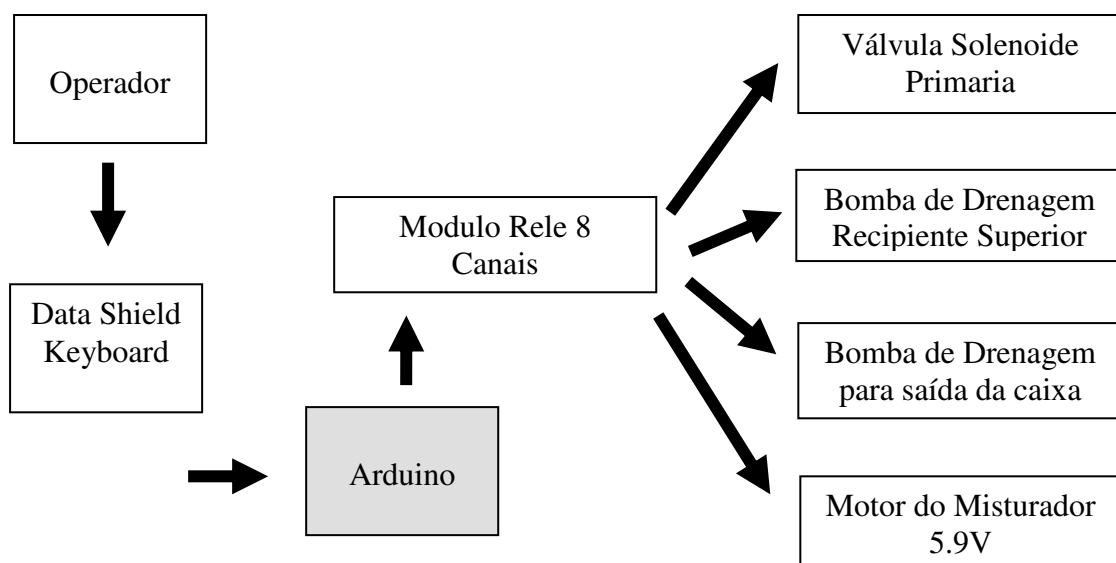


Figura 16 - Sistema de Operação do Arduino na Plataforma

O cérebro da plataforma é criado através do microcontrolador ATmega 2560 que coordenada e gerencia todos os periféricos ligados a ele, possibilitando o sistema de realizar as seguintes funções.

- Ligar a válvula solenoide primária para liberação da água;
- Ligar a moto bomba para drenagem do material químico;
- Ligar o motor para misturar os líquidos;
- Acionar a bomba secundaria para liberar o produto da caixa de mistura;

O código estruturado para este sistema, em sua essência, é um loop infinito onde e realizado o processo em uma sequência baseada no tempo, e com base nessas informações, são acionados seus atuadores para realizar tais funções.

3.1.2. Estrutura

3.1.2.1. Arquitetura do Misturador

Para realizarmos a montagem da estrutura do projeto, a qual atendesse a arquitetura de um misturador, separado através do material químico, água para iniciação do processo, misturador e canal para liberação do resíduo, utilizou uma estrutura feita de madeira, composta por três andares. A seguir na figura 19 etapas do processo de corte e montagem da primeira estrutura física do projeto.



Figura 17 - Corte para montagem da estrutura.

Após o corte da madeira e montagem da estrutura com os apoios, temos o esquemático pronto para início da montagem eletrônica do projeto, conforme a figura 20 a seguir.



Figura 18 - Esquemático pronto para instalação eletrônica

3.1.2.2. Instalação dos recipientes.

Nesta etapa, iniciamos a instalação dos recipientes nos devidos lugares da estrutura do projeto. Na parte superior do projeto (terceiro andar), foram instalados o adaptador responsável por receber o encaixe da mangueira que irá liberar a água ao sistema e o recipiente responsáveis pela armazenagem do material químico responsável pela dissolução do óleo na caixa de gordura, conforme a figura 21 apresentada na próxima página.



Figura 19 - Parte Superior do Projeto

Na parte do meio do projeto, ou seja, no segundo andar, foi implantado a caixa na qual onde será realizada toda a mistura entre água e material químico, conforme a figura 22.



Figura 20 - Caixa de Mistura

No terceiro andar, será o local responsável pela saída da mistura para dissolução do óleo. Neste local que será despejado toda a mistura realizada no esquemático do projeto. Neste caso utilizamos a instalação da moto bomba para realizar o bombeamento da mistura pronta, para direcionar a caixa de gordura conforme a figura 23 a seguir.



Figura 21 - Saída da Mistura para local destinado

3.1.2.3. Conexões

3.1.2.3.1. Recipiente Água x Recipiente Misturador

Para realizamos a conexão entre o recipiente onde haverá a armazenagem da água e o local responsável para receber e realizar a mistura, realizamos uma adaptação de um cano PVC utilizados em caixa em encanamentos residenciais, juntamente com um adaptador de saída de caixa d'água no qual é realizado a conexão com a válvula solenoide que será responsável pela liberação controlada da água no recipiente de mistura, e por ultimo adaptamos uma mangueira na saída da válvula que conduzira a água até o recipiente após a válvula ser atracada para liberação (abertura). A seguir, na figura 24, poderá ser observado a conexão realizada entre os recipientes.

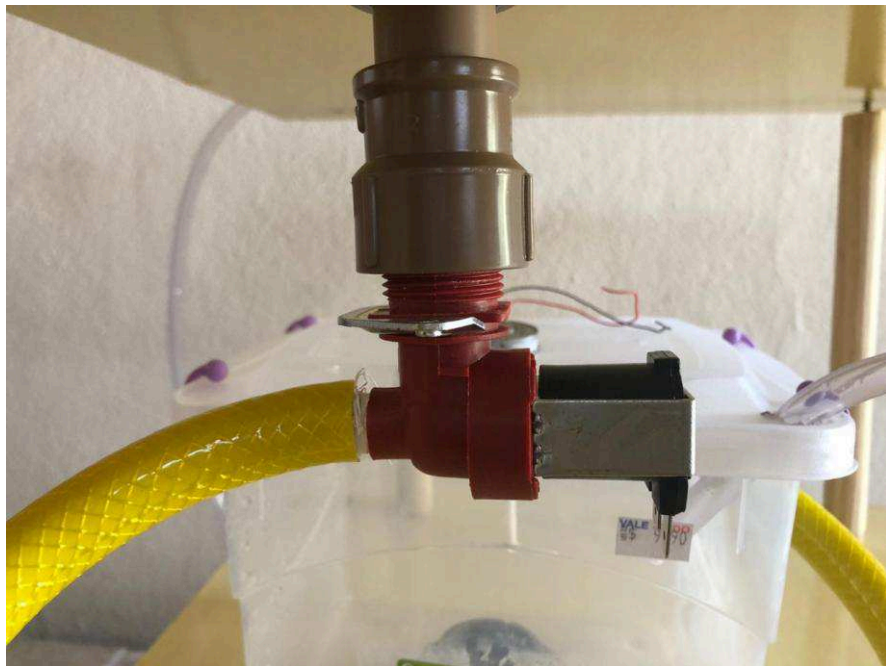


Figura 22 - Conexão com a Válvula Solenoide

3.1.2.3.2. Recipiente Armazenagem Material Químico x Recipiente Misturador

Para realizamos a conexão e envio do produto químico até o recipiente de mistura, utilizamos uma mangueira, adaptada na bomba de drenagem, que ficara submersa no recipiente. Após ser atracada, a bomba será responsável por drenar o material e enviar

diretamente para o recipiente de mistura através da mangueira de conexão, conforme podemos ver na figura 25.

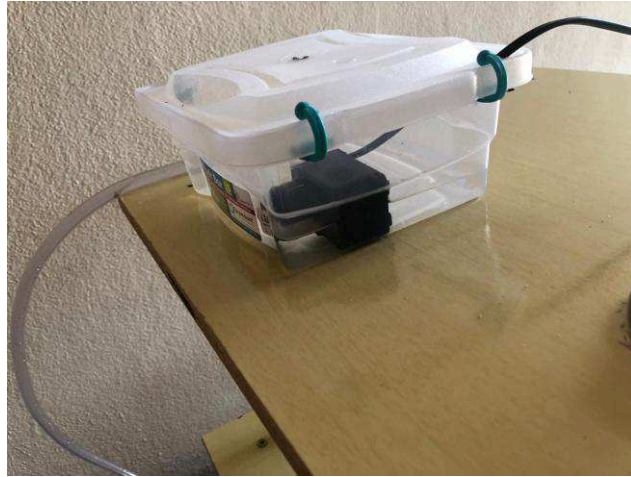


Figura 23 - Conexão de Produto químico à armazenagem.

3.1.2.3.3. Recipiente Misturador x Local para Liberação da Mistura

Após a finalização de toda mistura finalmente a mistura será liberada ao local apropriado e destinado para liberação. Para realizar a conexão entre ambos, utilizamos o procedimento semelhante a primeira conexão (recipiente de água x recipiente misturador. Foi instalado um adaptador de caixa de água na parte inferior do recipiente conectado a um cano PVC no qual foi instalado a moto bomba secundaria, responsável por bombear a mistura e levar ao lugar apropriado. Podemos verificar tal conexão na figura 26 a seguir.

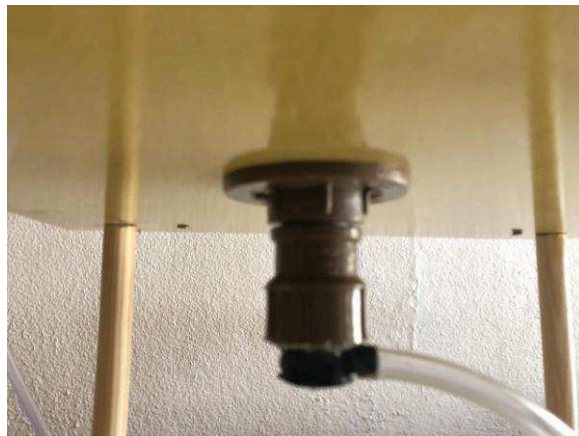


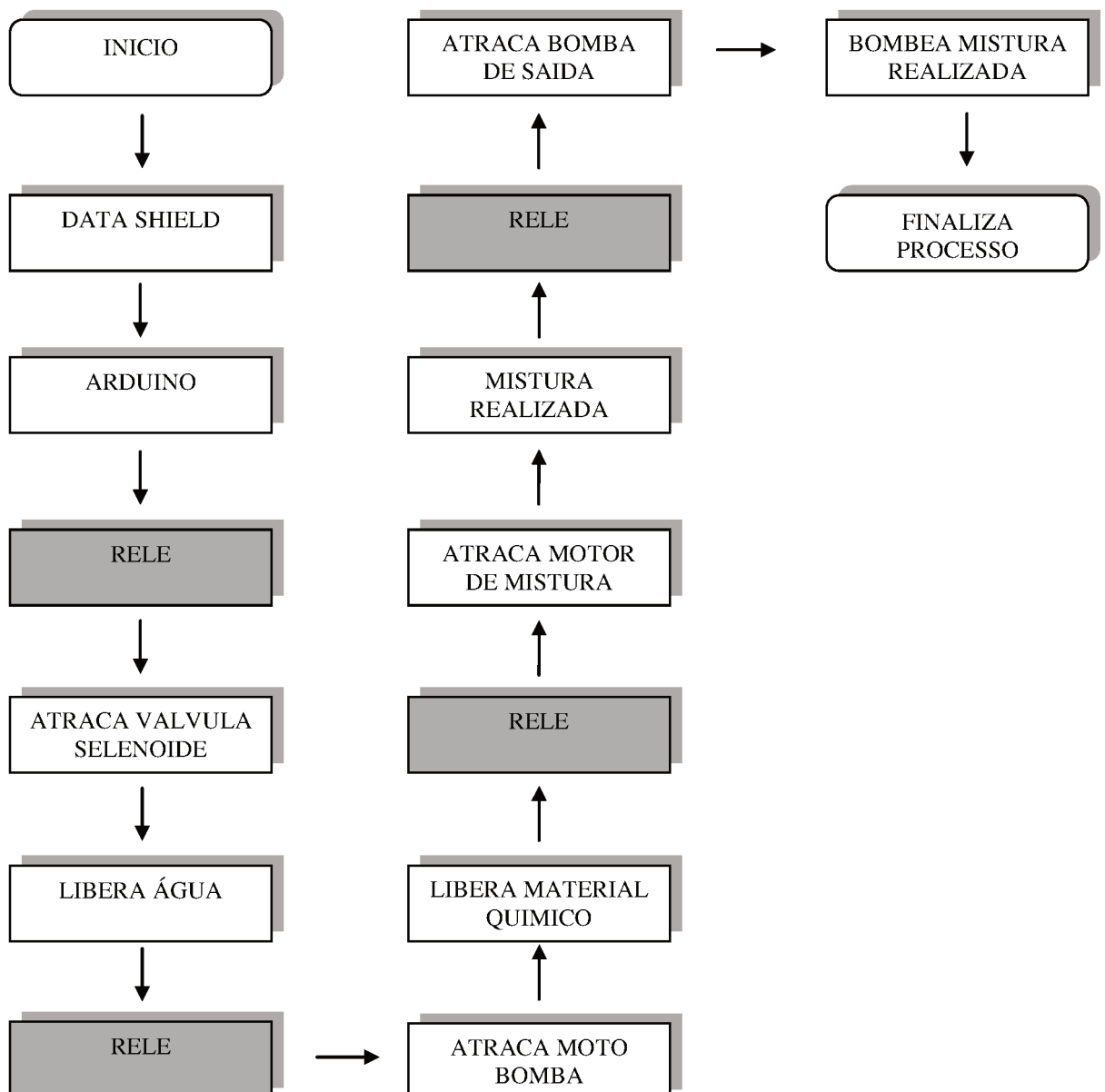
Figura 24 - Conexão entre misturador e saída para caixa de Gordura

3.2. FUNCIONAMENTO

O projeto baseou-se em uma estrutura responsável por realizar a mistura de um material químico, responsável pela dissolução do óleo antes da sua liberação a rede de esgoto.

Para darmos início no projeto, buscamos uma solução de baixo custo e que atendesse todas as solicitações necessárias, sendo elas liberação da água da água no recipiente, drenagem do material químico, no qual se encontrava uma densidade elevada ao comparada com a água, mistura dos materiais e por ultimo sua liberação.

Para realizar tais funções estabelecemos uma lógica de inicialização e funcionamento, conforme foi indicado pelo cliente necessário para realização com sucesso do procedimento, a mesma será representada a seguir através de um diagrama de processo.



O projeto consistiu em uma lógica que realizará um loop com pausa, responsável por uma sequência de comandos. O projeto inicialmente foi elaborado para dar sequência nos comandos apenas uma única vez após ser dado o START, ou seja, o misturador realizará apenas uma mistura, sendo necessário um novo START para uma nova mistura. Porém para melhorias e adaptações futuras o misturador poderá ser programado para realizar a função algumas vezes ao dia, conforme a necessidade do cliente e volume de resíduos para ser dissolvido, dispensando o START diário.

Foram realizados testes de operação, sendo verificado o tempo necessário para cada operação ser realizada. Levamos em consideração o volume do material químico e a quantidade de água necessário para realização do processo no protótipo criado. Os valores em relação ao tempo também poderão ser alterados de acordo com a solicitação do cliente, relacionado totalmente ao volume de material que terá que ser dissolvido.

3.2.1. Relação de capacidade do tanque

Para a melhoria do projeto, criamos uma programação relativa com a quantidade de resíduo para ser dissolvida no momento do processo, ou seja, a quantidade de resíduo que precisará ser dissolvida terá uma relação no tempo de atuação de cada atuador. A mesma foi dividida em três etapas. O operador poderá escolher a opção desejada diretamente no Display acoplado no arduino, clicando em select. Após isso, o sistema realiza o processo de forma automática.

3.2.1.1. Primeira Etapa – Pouco Resíduo

A etapa inicial é considerada a mais importante do protótipo, pois nela é feita uma relação com o tamanho do tanque, na qual as etapas posteriores, apenas deverão acrescentar os valores de tempo, baseado nos valores iniciais da primeira etapa. Caso o tanque (caixa de gordura) esteja com pouco resíduo para realizar o processo, utilizaremos a etapa um. A mesma aparecerá no visor descrito como “Pouco Resíduo”, nesta etapa o tanque entenderá que haverá a necessidade de pouca água, pouco resíduo e pouco tempo de mistura”. Após o operador, selecionar esta opção a informação é enviada ao arduino que por sua vez transmitirá para a placa de relé, que com o qual irá atracar os atuadores mantendo por pouco tempo e se

desligará, assim não haverá o desperdício do produto químico nem água. Podemos ver sobre a etapa na imagem a seguir.

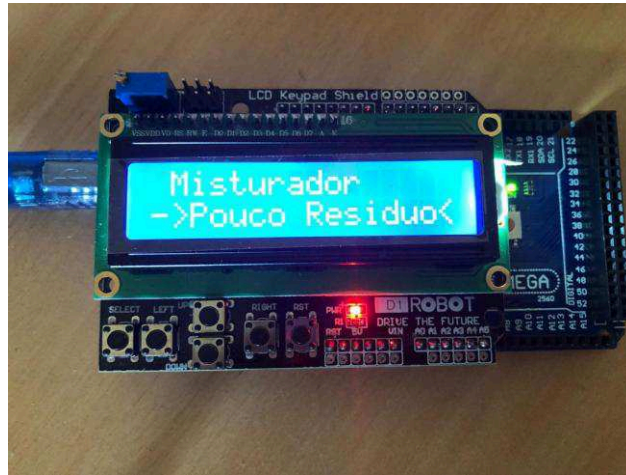


Figura 25 - Etapa 1 “Pouco Resíduo”

3.2.1.2. Segunda Etapa – Relativamente Cheio

Nesta etapa é entendido que o tanque esteja com uma quantidade de resíduo relativo, na qual necessitará de um tempo maior para mistura. Nestas condições o fabricante do produto, indica que deverá ser usada apenas uma pequena quantidade superior a primeira etapa e que deverá manter a mesma quantidade de água da etapa anterior. Nesta etapa será apresentada no visor como “Meio Residuo”. O sistema irá operar com 3 segundos a mais de produto químico e 5 segundos a mais de mistura. Devemos levar em consideração que ambos os valores foram escolhidos de acordo com testes realizados no protótipo do projeto.



Figura 26 - Etapa 2 “Meio Resíduo”

3.2.1.3. Terceira Etapa – Tanque Cheio

Na ultima etapa, consideramos que o tanque esteja perto do seu nível máximo, ou seja, possui uma grande quantidade de resíduo para ser diluída. Neste caso, teremos que optar por aumentar todos os tempos de operação, de acordo com testes realizados no protótipo. Realizamos uma alteração referente aos valores de operação, aumentando a liberação da água por mais três segundos, também realizamos uma alteração na liberação do produto químico, no qual é preciso uma quantidade superior de produto devido à quantidade de resíduo no tanque, aumentando para mais três segundos. Por ultimo, é realizada uma alteração no tempo de mistura, que também devido a quantidade superior de resíduo deveria ser superior às etapas anteriores, sendo aumentada para mais quinze segundos. Na ultima etapa é apresentada no visor como “Cheio”. Na próxima podemos conferir imagem da terceira e ultima etapa.



Figura 27 - Etapa 2 “Muito Resíduo”

As etapas mostradas anteriormente foram criadas e adaptadas de acordo com a utilização do protótipo atual, porém ambas podem ser alteradas, ou também acrescentadas novas etapas do protótipo, de acordo com a necessidade do cliente.

4. RESULTADOS

Após testes realizados podemos chegar aos resultados do projeto, no qual apresentou resultado eficiente pelo qual foi desenvolvido.

O protótipo conseguiu operar de forma pontual, realizando todas as funções as quais foram desenvolvidas e empregada no arduino. Foi realizado uma sequência de 4 testes, sendo o primeiro apenas na parte mecânica do projeto, verificando possíveis vazamentos nos encaixes e conexões. Após a verificação da parte mecânica seguimos os testes da parte eletrônica, testando separadamente cada componente, após todos os resultados positivos então começamos a ligações eletrônicas a cada componente testando a parte de programação.

Com o projeto totalmente conectado e programado para configurações iniciais, começamos a sequência de testes de operação, no quais foi possível definir o tempo de operação para cada fase do protótipo, bem como a operação de cada componente.

Após a sequência de testes, podemos chegar aos resultados positivos do projeto, apresentando total eficiência nas atividades de processamento e mistura no qual poderá ser empregado em qualquer ambiente onde haja a necessidade da dissolução de resíduos para caixas de gorduras.

O projeto também apresentou um ótimo custo benefício em referência aos concorrentes do ramo, tendo um gasto total de R\$250,00(duzentos e cinquenta reais).

5. CONCLUSÃO

O objetivo do projeto foi de automatizar um sistema de mistura e gerencia-los de forma fácil podendo ser operada por qualquer pessoa, até mesmo pelo próprio cliente ou consumidor. Para atingir o objetivo foram utilizados hardware e software open source de baixo custo de compra e fácil implementação no sistema.

O projeto mostrou-se eficaz em seu propósito, que foi permitir a operação automática do sistema de mistura sem o manuseio do misturador e dos produtos químicos utilizados de forma manual e perigosa ao operador.

A utilização do Arduino propiciou uma simples e fácil implementação com todos os dispositivos do sistema e obteve o mesmo resultado que teriam sido obtidos caso utilizássemos no projeto outros microcontroladores, que teriam que ser instalados e esquematizados com diversos outros componentes para atingir o propósito da plataforma Arduino.

Os resultados obtidos foram satisfatórios e o misturador se mostrou útil, podendo ser implementado em qualquer cliente, seja ele industrial, comercial, ou consumidores finais. Atingiu o objetivo de toda automação; propiciar comodidade, segurança, facilidade e rapidez no processo, bem como resultar em uma preservação futura ao meio ambiente, reduzindo o óleo liberado a rede de esgoto.

REFERÊNCIAS

ARDUINO, **Primeiros Passos.** Disponível em: <http://paginapessoal.utfpr.edu.br/gustavobborba/material/files/EL66J_Slides_arduino.pdf> Acesso em: 19 out.2018.

BANZI, MASSIMO. **Getting Arduino Started.** P.267, 2015

CORPORATION, ARDUINO, **Arduino With Headers.** Disponível em: <<https://store.arduino.cc/usa/arduino-leonardo-with-headers> > Acesso em: 21 out.2018.

CORPORATION, ATMEL, **DataSheet Arduino Mega.** Disponível em: <<https://sites.google.com/a/liberato.com.br/sistemas-microprocessados/home/microprocessadores---4323/10---arduino> > Acesso em: 21 nov.2018.

GIRAD, GIULIANA. **Veja os estragos causados pelo óleo de cozinha despejado no ralo da pia.** Disponível em: < <http://g1.globo.com/jornal-hoje/noticia/2010/09/veja-os-estragos-causados-pelo-oleo-de-cozinha-despejado-no-ralo-da-pia.html>> Acesso em: 22 set.2018.

LEMONS, MANUEL. **Conheça esta plataforma de hardware livre.** Disponível em: <<http://blog.fazedores.com/arduino-conheca-esta-plataforma-de-hardware-livre-e-suas-aplicacoes/>> Acesso em: 2 set.2018.

MCROBERTS, Michel. **Arduino básico.** Disponível em: <<http://www.novateceditora.com.br/livros/arduino/capitulo9788575222744.pdf>> Acesso em: 14 out.2018.

OLIVEIRA, GUILHERME, **Microprocessadores Arduino.** Disponível em:<<https://sites.google.com/a/liberato.com.br/sistemas-microprocessados-i/home/microprocessadores---4323/10---arduino> > Acesso em: 18 out.2018.

SHILOH, MICHAEL. **Getting Started with Arduino.** P.262, 2014.

SOARES, KARLA. **O que é um Arduino e o que pode ser feito com ele.** Disponível em: <<http://www.techtudo.com.br/noticias/noticia/2013/10/o-que-e-um-arduino-e-o-que-pode-ser-feito-com-ele.html>> Acesso em 05 de julho de 2015. Acesso em: 4 nov.2015.

SOUZA, FABIO, **Arduino Mega.** Disponível em: <<https://www.embarcados.com.br/arduino-mega-2560>> Acesso em: 21 nov.2018.

SOUZA, FABIO, **Arduino Microprocessadores.** Disponível em: <<https://www.embarcados.com.br/arduino-uno/>> Acesso em: 21 nov.2018.

THOMSEN, ADILSON, **Arduino With Headers.** Disponível em: <<https://www.filipeflop.com/blog/o-que-e-arduino>> Acesso em: 18 nov.2018.

THOMSEN, ADILSON, **O que é Arduino.** Disponível em: <<https://www.filipeflop.com/blog/o-que-e-arduino>> Acesso em: 19 nov.2018.

THOMSEN, ADILSON. **Controle Modulo Rele Arduino.** Disponível em: <<https://www.filipeflop.com/blog/controle-modulo-rele-arduino>> Acesso em: 20 nov.2018.

APÊNDICE A - CODIGO FONTE UTILIZADO NO PROJETO

```
// SISTEMA DE MISTURA CONTROLADO PELO ARDUINO
// DIEGO CHAGAS
// TRABALHO DE CONCLUSÃO DE CURSO
// DEPARTAMENTO DE ENGENHARIA ELETRICA UNITAU
```

```
#include <LiquidCrystal.h> // Inclui biblioteca "LiquidCristal.h"
```

```
LiquidCrystal lcd (8, 9, 4, 5, 6, 7); // Define Pinos do Display
```

```
// DEFINE AS VARIÁVEIS
```

```
int Menu = 0; // Inicializa valores para Menu
```

```
int estado = 0; // Inicializa valores para estado
```

```
//Porta ligada ao pino IN1 do modulo 8 Valvula Solenoide
```

```
int porta_rele1 = 52;
```

```
//Porta ligada ao pino IN2 do modulo 7 Bomba Agua Cima
```

```
int porta_rele2 = 50;
```

```
//Porta ligada ao pino IN3 do modulo 6 Motor da Mistura
```

```
int porta_rele3 = 48;
```

```
//Porta ligada ao pino IN4 do modulo 5 Bomba de Saida
```

```
int porta_rele4 = 46;
```

```
// INICIA O CODIGO
```

```
void setup() {
```

```
  lcd.begin(16, 2); // Estabelece caracteres do display
```

```
  clearPrintTitle();
```

```
}
```

```
void loop() {
```

```
MenuPrincipal();  
}
```

```
void MenuPrincipal() {  
    int ValEstado = 0; // Inicializa valores para ValEstado  
    int x = analogRead (0);  
    lcd.setCursor(0,0); // Posiciona Cursor  
  
    if (x < 80) { // Define valores máximos para UP em ValEstado = 1  
        }  
    else if (x < 200) {  
        ValEstado = 1;  
        }  
    else if (x < 400){ // Define valores máximos para Down em ValEstado = 2  
        ValEstado = 2;  
        }  
    else if (x < 600){ // Define ação nula para Left  
        }  
    else if (x < 800){ // Define valores máximos para Select em ValEstado = 3  
        ValEstado = 3;  
        }  
  
    if (Menu < 0 || Menu > 3) {  
        Menu = 0;  
        }  
  
    if (ValEstado != estado) {  
        if (ValEstado == 1) {  
            Menu = Menu - 1;  
            displayMenu(Menu);  
        } else if (ValEstado == 2) {  
            Menu = Menu + 1;  
            displayMenu(Menu);  
        } else if (ValEstado == 3) {
```

```
selectMenu(Menu);  
}  
estado = ValEstado;  
}  
delay(5);  
}
```

```
void displayMenu(int x) {  
  switch (x) {  
    case 1:  
      clearPrintTitle();  
      lcd.print ("->Pouco Residuo<-"); // Imprime na tela a opção do Menu  
      break;  
    case 2:  
      clearPrintTitle();  
      lcd.print ("->Meio Residuo<-"); // Imprime na tela a opção do Menu  
      break;  
    case 3:  
      clearPrintTitle();  
      lcd.print ("->Muito Residuo<-"); // Imprime na tela a opção do Menu  
      break;  
  }  
}
```

```
void clearPrintTitle() {  
  lcd.clear();  
  lcd.setCursor(0,0);  
  lcd.print(" Misturador ");  
  lcd.setCursor(0,1);  
}
```

```
void selectMenu(int x) {  
  switch (x) {  
    case 1:
```

```
pinMode(porta_rele1, OUTPUT);
pinMode(porta_rele2, OUTPUT);
pinMode(porta_rele3, OUTPUT);
pinMode(porta_rele4, OUTPUT);
digitalWrite(porta_rele1, LOW); //Liga rele 1
digitalWrite(porta_rele2, HIGH); //Desliga rele 2
digitalWrite(porta_rele3, HIGH); //Desliga rele 3
digitalWrite(porta_rele4, HIGH); //Desliga rele 4
delay(5000);
digitalWrite(porta_rele1, HIGH); //Desliga rele 1
digitalWrite(porta_rele3, HIGH); //Desliga rele 3
digitalWrite(porta_rele4, HIGH); //Desliga rele 4
digitalWrite(porta_rele2, LOW); //Liga rele 2
delay(8000);
digitalWrite(porta_rele1, HIGH); //Desliga rele 1
digitalWrite(porta_rele2, HIGH); //Desliga rele 2
digitalWrite(porta_rele4, HIGH); //Desliga rele 4
digitalWrite(porta_rele3, LOW); //Liga rele 3
delay(20000);
digitalWrite(porta_rele1, HIGH); //Desliga rele 1
digitalWrite(porta_rele2, HIGH); //Desliga rele 2
digitalWrite(porta_rele3, HIGH); //Desliga rele 3
digitalWrite(porta_rele4, LOW); //Liga rele 4
delay(10000);
digitalWrite(porta_rele1, HIGH); //Desliga rele 1
digitalWrite(porta_rele2, HIGH); //Desliga rele 2
digitalWrite(porta_rele3, HIGH); //Desliga rele 3
digitalWrite(porta_rele4, HIGH); //Desliga rele 4
break;
case 2:
pinMode(porta_rele1, OUTPUT);
pinMode(porta_rele2, OUTPUT);
pinMode(porta_rele3, OUTPUT);
pinMode(porta_rele4, OUTPUT);
```



```
digitalWrite(porta_rele1, LOW); //Liga rele 1
digitalWrite(porta_rele2, HIGH); //Desliga rele 2
digitalWrite(porta_rele3, HIGH); //Desliga rele 3
digitalWrite(porta_rele4, HIGH); //Desliga rele 4
delay(8000);
digitalWrite(porta_rele1, HIGH); //Desliga rele 1
digitalWrite(porta_rele3, HIGH); //Desliga rele 3
digitalWrite(porta_rele4, HIGH); //Desliga rele 4
digitalWrite(porta_rele2, LOW); //Liga rele 2
delay(14000);
digitalWrite(porta_rele1, HIGH); //Desliga rele 1
digitalWrite(porta_rele2, HIGH); //Desliga rele 2
digitalWrite(porta_rele4, HIGH); //Desliga rele 4
digitalWrite(porta_rele3, LOW); //Liga rele 3
delay(40000);
digitalWrite(porta_rele1, HIGH); //Desliga rele 1
digitalWrite(porta_rele2, HIGH); //Desliga rele 2
digitalWrite(porta_rele3, HIGH); //Desliga rele 3
digitalWrite(porta_rele4, LOW); //Liga rele 4
delay(25000);
digitalWrite(porta_rele1, HIGH); //Desliga rele 1
digitalWrite(porta_rele2, HIGH); //Desliga rele 2
digitalWrite(porta_rele3, HIGH); //Desliga rele 3
digitalWrite(porta_rele4, HIGH); //Desliga rele 4
break;
case 3:
pinMode(porta_rele1, OUTPUT);
pinMode(porta_rele2, OUTPUT);
pinMode(porta_rele3, OUTPUT);
pinMode(porta_rele4, OUTPUT);
digitalWrite(porta_rele1, LOW); //Liga rele 1
digitalWrite(porta_rele2, HIGH); //Desliga rele 2
digitalWrite(porta_rele3, HIGH); //Desliga rele 3
digitalWrite(porta_rele4, HIGH); //Desliga rele 4
```

```
delay(5000);
digitalWrite(porta_rele1, HIGH); //Desliga rele 1
digitalWrite(porta_rele3, HIGH); //Desliga rele 3
digitalWrite(porta_rele4, HIGH); //Desliga rele 4
digitalWrite(porta_rele2, LOW); //Liga rele 2
delay(8000);
digitalWrite(porta_rele1, HIGH); //Desliga rele 1
digitalWrite(porta_rele2, HIGH); //Desliga rele 2
digitalWrite(porta_rele4, HIGH); //Desliga rele 4
digitalWrite(porta_rele3, LOW); //Liga rele 3
delay(20000);
digitalWrite(porta_rele1, HIGH); //Desliga rele 1
digitalWrite(porta_rele2, HIGH); //Desliga rele 2
digitalWrite(porta_rele3, HIGH); //Desliga rele 3
digitalWrite(porta_rele4, LOW); //Liga rele 4
delay(10000);
digitalWrite(porta_rele1, HIGH); //Desliga rele 1
digitalWrite(porta_rele2, HIGH); //Desliga rele 2
digitalWrite(porta_rele3, HIGH); //Desliga rele 3
digitalWrite(porta_rele4, HIGH); //Desliga rele 4
break;
}
}
```