



ETEC JORGE STREET

**TRABALHO DE CONCLUSÃO DO CURSO TÉCNICO EM AUTOMAÇÃO
INDUSTRIAL**

ALIMENTADOR AUTOMÁTICO

**Camila Oliveira da Cunha
Davi Pichinin de Assis
Edimar Castellani Júnior
Kaique Lacerda Silva**

**Professor Orientador:
Renê Graminhani**

**São Caetano do Sul / SP
2017**

ALIMENTADOR AUTOMÁTICO

Trabalho de Conclusão de Curso
apresentado como pré-requisito para
obtenção do Diploma de Técnico em
Automação Industrial.

**São Caetano do Sul / SP
2017**

AGRADECIMENTOS

Agradecemos ao nossos familiares, professores, amigos e colegas de classe pelo apoio prestado durante o desenvolvimento do projeto e à escola por ceder laboratórios e espaços para concretização do projeto.

RESUMO

O projeto visa automatizar um sistema de alimentação de pets. O alimentador distribuirá, automaticamente, porções de rações secas e água conforme programado. Será microcontrolado por arduino atribuindo a distribuição de ração por sistema de despejo e água por sistema de vazão. Através de um painel LCD será possível verificar as refeições programadas, tamanho das porções e número de refeições servidas. O produto tem capacidade de, aproximadamente, 1kg de alimento seco e 2 litros de água.

Palavras-chave: Automatizar. Alimentação. Arduino.

LISTA DE FIGURAS

Figura 1- Arduino.....	11
Figura 2- Relé.....	12
Figura 3 - Válvula solenóide.....	13
Figura 4 - Sensor de água.....	14
Figura 5 - Fonte de alimentação.....	15
Figura 6 – Diagrama em blocos.....	16
Figura 7 – Fluxograma do Arduíno.....	17
Figura 8 – Croqui.....	18
Figura 9 – Compra de materiais.....	19
Figura 10 – Teste da Válvula Solenóide.....	20
Figura 11 – Teste da Válvula Solenóide e Sensor Nível de Água.....	20
Figura 12 – Teste do Sensor Nível de Água e Buzzer.....	21
Figura 13 – Início dos testes.....	22
Figura 14 – Teste Arduíno.....	22
Figura 15 – Teste do Relé.....	23
Figura 16 –Teste da Solenóide.....	23
Figura 17 – Display em funcionamento.....	24
Figura 18 – Teste da Programação.....	24
Figura 19 – Recipiente.....	25
Figura 20 – Montagem da Válvula Solenóide e Relé de nível.....	25
Figura 21 – Resultado obtido.....	26

SUMÁRIO

Introdução.....	9
1.Fundamentação Teórica.....	11
1.1. Arduino.....	11
1.2. Relé.....	12
1.3. Válvula Solenóide.....	13
1.4. Sensor nível de água.....	14
1.5. Fonte de alimentação.....	15
2. Planejamento do projeto.....	16
2.1. Descrição do projeto.....	16
2.2. Diagrama em blocos.....	16
2.3. Fluxograma do Arduino.....	17
2.4. Croqui.....	18
2.5. Cronograma.....	18
3. Desenvolvimento do projeto.....	19
3.1. Compra de materiais.....	19
3.2. Testes.....	20
3.3. Montagem do projeto.....	22
4. Resultados obtidos.....	26
Conclusão.....	27
Referências Bibliográficas.....	28
Anexo A.....	29

Introdução

Para a finalização do curso técnico em Automação Industrial surgiram, inicialmente, ideias sobre como simplificar ações do cotidiano. Com isso, pensamos em Alimentador Automático, Lixeira Automática e QR Code em Aeroportos.

O projeto escolhido foi o Alimentador Automático e, este, tem utilidade na alimentação de animais domésticos em casos de ausência do dono.

É um equipamento de programação compreensível, avisos sonoros na ausência de ração ou água e de fácil locomoção.

O consenso comum em realizar o projeto aconteceu devido à convivência dos integrantes com animais, compreendendo que no dia-a-dia os animais passam boa parte do tempo sozinhos, sem alimentar-se ou alimentando-se em excesso. E sabendo que um percentual de 20% de peso excessivo, a predisposição a graves problemas de saúde é elevada, é importante dar atenção à questão da alimentação, pois o estado nutricional influencia na qualidade de vida dos animais. Existe uma preocupação referente à ligação entre a nutrição e a saúde e sua contribuição para o bem-estar geral dos animais.

TEMA E DELIMITAÇÃO

O projeto é a confecção de um alimentador automático enfocado para automatizar o processo de alimentação de animais domésticos. É um equipamento de fácil programação para o usuário, locomoção sem grande esforço e opções para faixa etária e porte do animal, promovendo comodidade para o dono.

OBJETIVOS

O objetivo é desenvolver um protótipo microcontrolado de um alimentador automático para distribuição de ração seca e água.

Objetivo Geral

Desenvolver um projeto para alimentação de animais domésticos na ausência do dono.

Objetivos Específicos

Controlar, rigorosamente, a quantidade de ração despejada no comedouro e a quantidade de água no bebedouro.

JUSTIFICATIVA

Compreendendo que no dia-a-dia os animais passam boa parte do tempo sozinhos, sem se alimentar ou alimentando-se em excesso. E sabendo que um percentual de 20% de peso excessivo, a predisposição a graves problemas de saúde é elevada, é importante dar atenção à questão da alimentação, pois o estado nutricional influencia na qualidade de vida dos animais. Existe uma preocupação referente à ligação entre a nutrição e a saúde e sua contribuição para o bem-estar geral dos animais.

METODOLOGIA

A metodologia empregada neste projeto é a metodologia experimental, ou seja, a partir das ideias e pesquisas, construiu-se um protótipo funcional.

Fundamentação Teórica

Aqui serão descritas: a fundamentação teórica do projeto e a coleta de dados referente à pesquisa executada.

1.1. Arduíno

Foi utilizado o arduíno MEGA2560. Este tipo de Arduíno é mais indicado para projetos que exigem mais memória de esboço e memória RAM. São, ao todo, 54 pinos digitais, 16 entradas analógicas e um espaço maior para seu esboço.



Figura 1- Arduino – Fonte: O grupo

1.2. Relé

É um componente eletromecânico controlador de circuitos a partir de pequenas correntes e/ou tensões. O relé possui o seguinte funcionamento: a partir da circulação da corrente pela bobina, o campo magnético criado atrai contato(s) fechando ou abrindo circuitos. Se a corrente for interrompida, o campo magnético consequentemente também é, fazendo com que os contatos voltem à posição original. Podem ser NA (normalmente aberto), NF (normalmente fechado) ou ambos.



Figura 2 – Relé – Fonte: O grupo

1.3. Válvula solenóide

A Válvula Solenoide usada no projeto ($\frac{1}{2} \times \frac{1}{2}$) possui uma bobina interna em formato cilíndrico. No momento em que a corrente elétrica passa pela bobina, gera uma força na região central, que é responsável pelo acionamento do êmbolo, que se encontra na parte interna e é responsável pela abertura/fechamento do sistema, posicionado 180° em relação à entrada e saída de água.

A Solenoide 12V é normal fechada, ou seja, sempre que houver perda ou falta de energia o sistema permanecerá fechado impossibilitando a passagem de água. A válvula solenoide, usada no projeto, controlada por um arduino MEGA, terá a função de controlar a vazão de água.

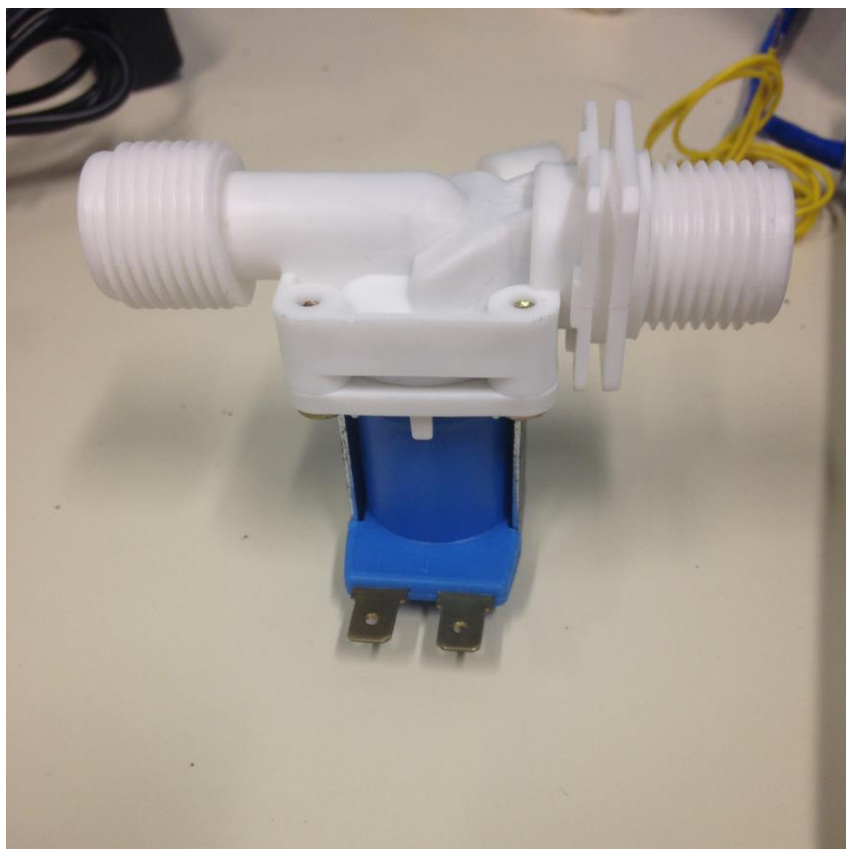


Figura 3 – Válvula Solenoide – Fonte: O grupo

1.4. Sensor nível de água

O sensor é feito de plástico, possui uma haste na qual percorre um material flutuante. Tem seu funcionamento baseado em contato reed switch e um flutuador magnético. O cilindro possui contatos normal aberto (NA) e normal fechado (NF). Conforme o posicionamento desse flutuador, ele abre ou fecha o contato reed switch. Sendo possível mandar sinais para sistemas de controles digitais (arduinos, microcontroladores), ou acionamento de CLPs, relés e até contadores para ligar ou desligar bombas de água etc.



Figura 4 – Sensor de água – Fonte: O grupo

1.5. Fonte de alimentação

A fonte de alimentação usada é padrão de entrada 100 – 240V e saída de 12V/1A. É composta, basicamente, de 4 componentes: um transformador de força (que pode aumentar ou diminuir a tensão), filtro capacitivo e/ou indutivo, circuito retificador e reguladores de tensão. Sua função é transformar a energia elétrica na forma de corrente alternada, para energia elétrica corrente contínua, que é o necessário para o arduino ser alimentado.



Figura 5 – Fonte de alimentação– Fonte: O grupo

2. Planejamento do projeto

2.1. Descrição do projeto

Os reservatórios terão sensores que mandarão informações para o ArduinoMEGA. Os recipientes terão programações prévias para a alimentação e hidratação adequada, contendo quantidades diárias necessárias. A partir de uma seleção feita pelo usuário, no LCD, os recipientes despejarão água e ração.

Através de um sensor de peso, o arduino receberá informações sobre a quantidade de ração que precisa despejar e, através de uma rosca dosadora, realizará a ação.

Através de dois sensores de água; um localizado no pote e outro na estrutura interna indicando os níveis de água. Quando a água estiver abaixo do nível do sensor, um buzzer será acionado para ciência do usuário, indicando no LCD. Já o sensor externo, servirá para que não transborde. A cada 2 horas, um motor de vibração acionará por determinado tempo para não ocasionar de a água ficar parada, evitando, assim, doenças ou infecções.

2.2. Diagrama em blocos

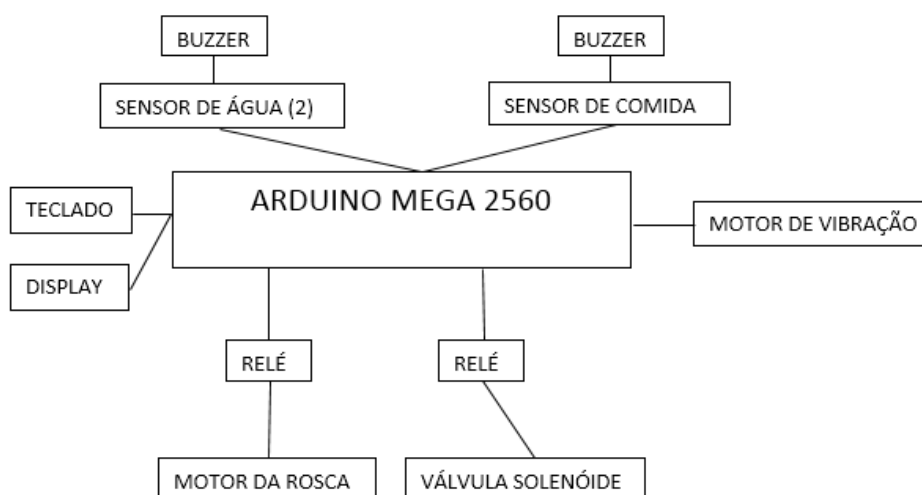


Figura 6 – Diagrama em blocos – Fonte: O grupo

2.3. Fluxograma do Arduino

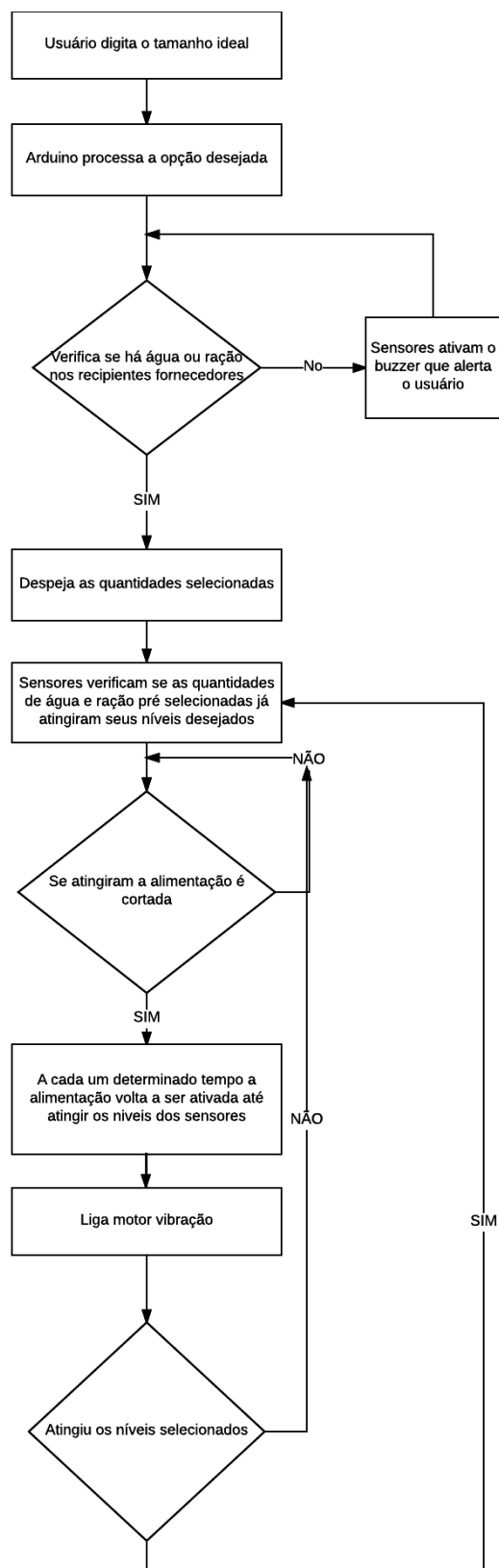


Figura 7 – Fluxograma do Arduino – Fonte: O grupo.

2.4. Croqui

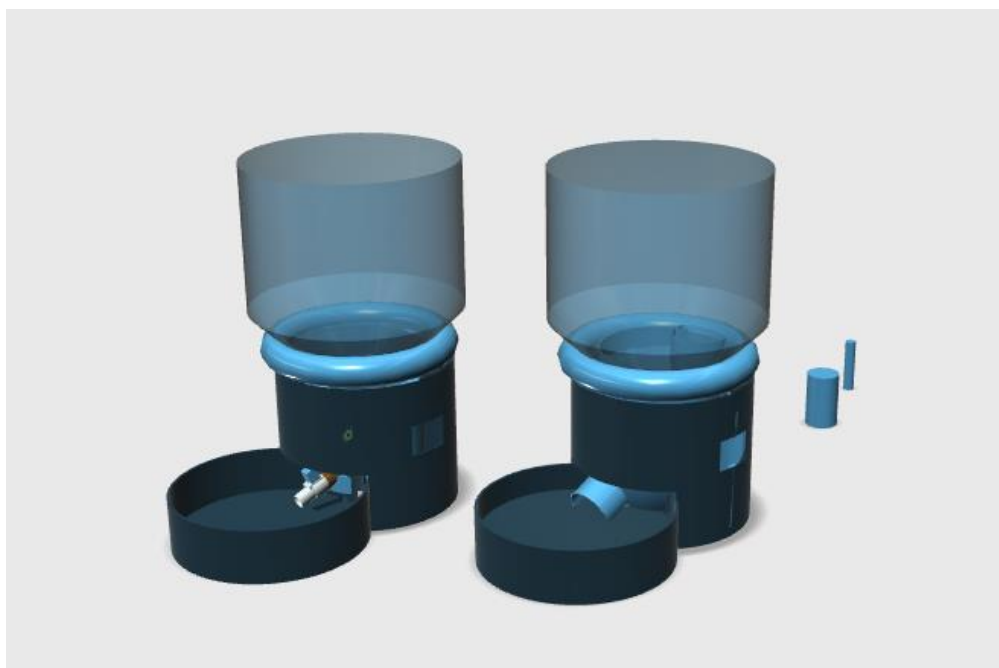


Figura 8 – Croqui – Fonte: O grupo

2.5. Cronograma

TAREFA/RESPONSÁVEL		07/fev	14/fev	21/fev	28/fev	07/mar	14/mar	21/mar	28/mar	04/abr	11/abr	18/abr	25/abr	02/mai	09/mai	16/mai	22/mai
Monografia	P																
Camila/Davi	R																
Compra de Materiais	P																
Davi/Edimar/Kaique	R																
Montagem e Teste do Hardware	P																
Camila/Davi/Edimar/Kaique	R																
Engenharia do Produto	P																
Kaique	R																
Produção de Software	P																
Kaique	R																
Montagem da Estrutura	P																
Davi/Edimar/Kaique	R																

Figura 9 – Cronograma – Fonte: O grupo

3. Desenvolvimento do projeto

Durante esta etapa, diversos testes foram realizados, separadamente, em partes compositoras do projeto para visualização de possíveis erros e/ou falhas no sistema. Depois de todos os testes, a junção foi feita e, conseqüentemente, o trabalho completo.

3.1. Compra de materiais

Para a compra de materiais, os preços foram previamente estudados pela internet e, também, por lojas físicas.



Figura 9 – Compra de materiais – Fonte: O grupo

3.2. Testes

Nesta etapa, alguns testes foram realizados. Exemplos:

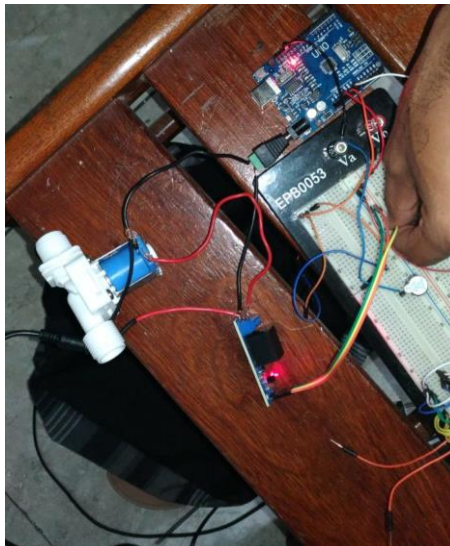


Figura 10 – Teste da Válvula Solenóide – Fonte: O grupo

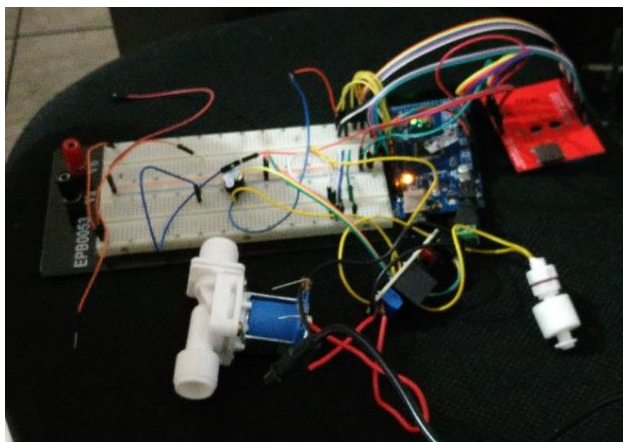


Figura 11 – Teste da Válvula Solenóide e Sensor Nível de Água – Fonte: O grupo

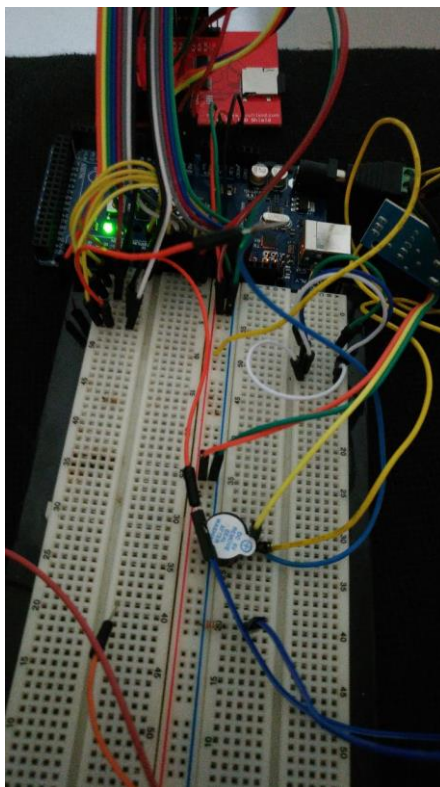


Figura 12 – Teste do Sensor Nível de Água e Buzzer – Fonte: O grupo

3.3. Montagem do projeto



Figura 13 – Início dos testes – Fonte: O grupo

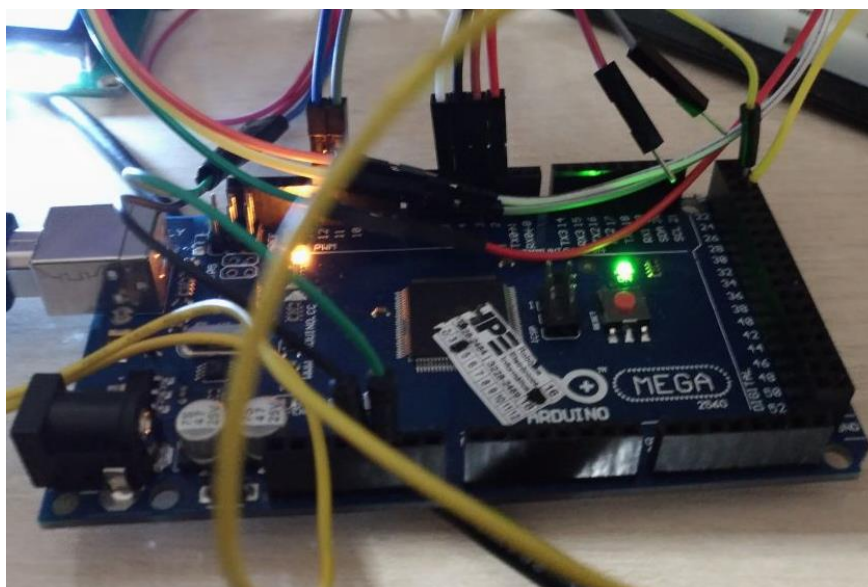


Figura 14 – Teste Arduino – Fonte: O grupo



Figura 15 - Teste Relé – Fonte: O grupo

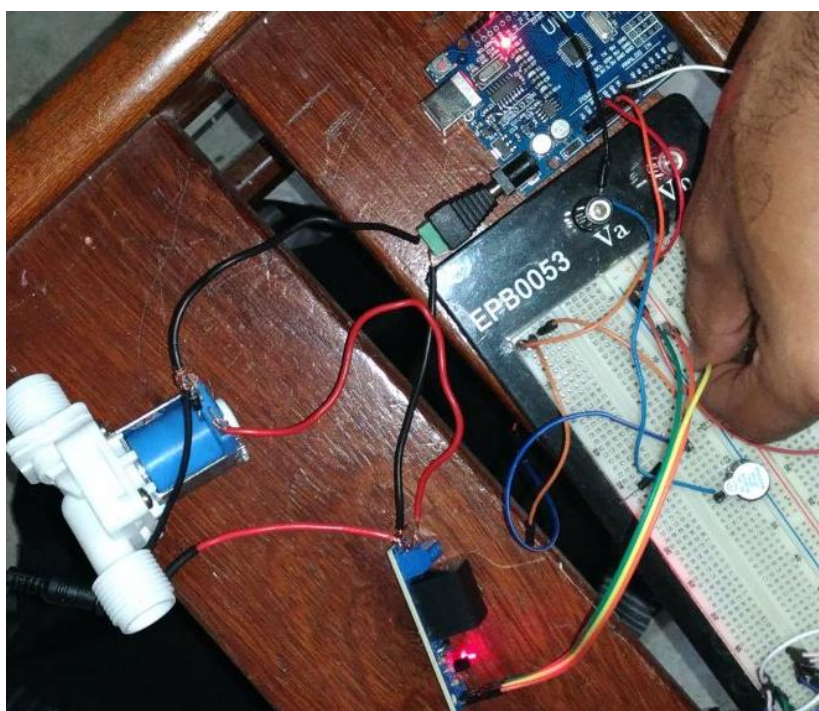


Figura 16 – Teste de Solenóide – Fonte: O grupo



Figura 17- Display em funcionamento – Fonte: O grupo

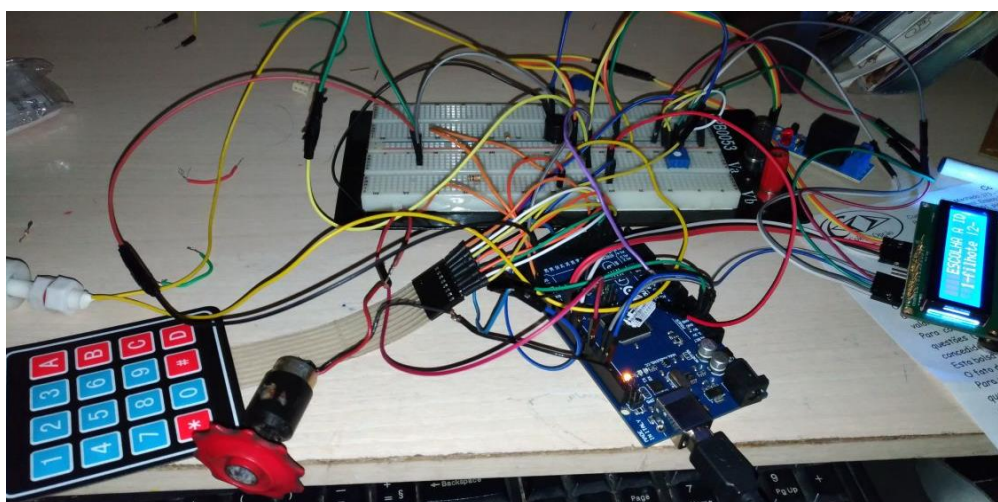


Figura 18 – Teste da Programação – Fonte: O grupo



Figura 19 – Recipiente – Fonte: O grupo



Figura 20 – Montagem da Válvula Solenóide e Relé de nível – Fonte: O grupo

4. Resultados Obtidos



Figura 21 - Resultado obtido – Fonte: O grupo

Conclusão

Uma experiência laboratorial completa. Desde à montagem até os problemas e as soluções, pudemos ver, de fato, em prática, tudo que outrora fora visto em teoria (sala de aula ou oficinas de experimentos) no decurso de dois anos letivos.

Referências

Introdução ao Arduino. Disponível em:
<<http://www.educatronica.com.br/arduino.intro.html>>. Acesso em: abril de 2017.

Introdução ao Arduino MEGA2560. Disponível em:
<<https://www.arduino.cc/en/Guide/ArduinoMega2560>>. Acesso em: abril de 2017.

Tudo sobre relés. Disponível em:
<<http://www.newtoncbraga.com.br/index.php/como-funciona/597-como-funcionam-os-reles?showall=1&limitstart=>>>. Acesso em: abril de 2017.

Anexo A – Programa do Arduino

```
//Carrega a biblioteca LiquidCrystal
#include <LiquidCrystal.h>
// teclado
#include <Keypad.h>

//SENSOR NIVEL DE AGUA POTE

int vazio = 23;//BUZZER
int agua;
int sensorSOLENOIDE;
int sensorQUANT;
int pinsensor2 = 30;
int vibra = 22;
int motor = 37;
int tempo = 36;

//Define os pinos que serão utilizados para ligação ao display
LiquidCrystal lcd (12, 11, 5, 4, 3, 2);
//lcd(23, 25, 37, 39, 41);

//\V teclado
const byte ROWS = 4; //four rows
const byte COLS = 4; //four columns
//define the cymbols on the buttons of the keypads
char hexaKeys[ROWS][COLS] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};
byte rowPins[ROWS] = {52, 50, 48, 46}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {53, 51, 49, 47}; //connect to the column pinouts of the keypad

//initialize an instance of class NewKeypad
Keypad customKeypad = Keypad( makeKeymap(hexaKeys), rowPins, colPins,
ROWS, COLS);
//teste

//solenóide
int solenoide;
//DESLIGARLIGAR RELE /SOLENOIDE
// void interrup(){
//digitalWrite(33, LOW); // DESLIGA A SOLENOIDE
//}

// SENSOR INTERNO + BUZZER
```

```

void alerta2(){
{

digitalWrite(vazio, HIGH);
delay(3000);

{
  lcd.clear();
  lcd.setCursor(4, 0);
  lcd.print("Atencao!!!");
  lcd.setCursor(0, 1);
  lcd.print("AGUA ESTA ACABANDO");
  for (int posicao = 0; posicao < 3; posicao++)
  {
    lcd.scrollDisplayLeft();
    delay(300);
  }
  //Rolagem para a direita
  for (int posicao = 0; posicao < 3; posicao++)
  {
    lcd.scrollDisplayRight();
    delay(300);
  }
  delay (5000);
}
} return 0;
}

void alerta(){
  digitalWrite(vazio, LOW);

  lcd.clear();
  //Posiciona o cursor na coluna 0, linha 0;
  lcd.setCursor(1, 0);
  //Envia o texto entre aspas para o LCD
  lcd.print(" ALIMENTADORA");
  lcd.setCursor(2, 1);
  lcd.print(" AUTOMATICA ");
  delay(500);
  digitalWrite(vazio, LOW);
}

void setup()
{
attachInterrupt(2,alerta2,RISING);//sensor nivel de agua
attachInterrupt(3,alerta,FALLING);

/// attachInterrupt(2,interrup,RISING);//sensor nivel de agua desliga solenoide

//Define o número de colunas e linhas do LCD

```

```

Serial.begin(9600);
lcd.begin(16, 2);

pinMode(22, OUTPUT); //MOTOR VIBRA
pinMode(23, INPUT); //SENSOR AGUA
pinMode(30, OUTPUT);
pinMode(33, OUTPUT);
pinMode(8, OUTPUT);
pinMode(tempo, INPUT);
pinMode(vazio, OUTPUT); //BUZZER

```

```

lcd.clear();
//principal
//Limpa a tela
lcd.clear();
//Posiciona o cursor na coluna 0, linha 0;
lcd.setCursor(1, 0);
//Envia o texto entre aspas para o LCD
lcd.print(" ALIMENTADORA");
lcd.setCursor(2, 1);
lcd.print(" AUTOMATICA ");
delay(500);
//Rolagem para a esquerda
for (int posicao = 0; posicao < 3; posicao++)
{
    lcd.scrollDisplayLeft();
    delay(300);
}
//Rolagem para a direita
for (int posicao = 0; posicao < 6; posicao++)
{
    lcd.scrollDisplayRight();
    delay(300);
}

```

```

int escolha = 0;
while(escolha == 10)
    escolha;
{
    //escolha
    //Limpa a tela
    lcd.clear();
    //Posiciona o cursor na coluna 0, linha 0;
    lcd.setCursor(4, 0);
    //Envia o texto entre aspas para o LCD

```

```

lcd.print("ESCOLHA A IDADE DO SEU PET: ");
lcd.setCursor(2, 1);
lcd.print("1-filhote | 2- Jovem | 3- Adulto");
delay (400);
//Rolagem para a esquerda
for (int posicao = 0; posicao < 26; posicao++)
{ lcd.scrollDisplayLeft();
  delay(600); }
//Rolagem para a direita
for (int posicao = 0; posicao < 26; posicao++) {
  lcd.scrollDisplayRight();
  delay(500); }
delay(5000);

char customKey = customKeypad.getKey();

if (customKey){
  Serial.println(customKey);
}
switch(customKey)

{
case 1://escolhendo filhote
int escolhafilhote;
switch(customKey)
{
  lcd.clear();
  //Posiciona o cursor na coluna 0, linha 0;
  lcd.setCursor(0, 0);
  //Envia o texto entre aspas para o LCD
  lcd.print("Seu filhote recebera 4");
  lcd.setCursor(0, 1);
  lcd.print("porções de ração no dia");
  delay (5000);
  lcd.clear();
  lcd.setCursor(0, 0);
  //Envia o texto entre aspas para o LCD
  lcd.print("ESCOLHA A QUANTIA:");
  lcd.setCursor(1, 1);
  lcd.print(" 1 | 2 | 3 | 4");
  {
    case 1:
      digitalWrite(8, HIGH);//LIGAR MOTOR ROSCA
      delay(50000);
      digitalWrite(8, HIGH);

    break;
    case 2:
    break;

```

```

    case 3:
    break;
    case 4:
    break;
  }}
  break;
case 2://pequeno
int escolhapequeno;

switch(customKey)
{
  lcd.clear();
  //Posiciona o cursor na coluna 0, linha 0;
  lcd.setCursor(0, 0);
  //Envia o texto entre aspas para o LCD
  lcd.print("Seu PET recebera 3");
  lcd.setCursor(0, 1);
  lcd.print("porções de ração no dia");
  delay (5000);
  lcd.clear();
  lcd.setCursor(0, 0);
  //Envia o texto entre aspas para o LCD
  lcd.print("ESCOLHA A QUANTIA:");
  lcd.setCursor(1, 1);
  lcd.print(" 1 | 2 | 3 | 4");
  {
    case 1:
    break;
    case 2:
    break;
    case 3:
    break;
    case 4:
    break;
  }
  break;
case 3://medio/grande
int escolhamedio;

switch(customKey)
{
  lcd.clear();
  //Posiciona o cursor na coluna 0, linha 0;
  lcd.setCursor(0, 0);
  //Envia o texto entre aspas para o LCD
  lcd.print("Seu PET recebera 3");
  lcd.setCursor(0, 1);
  lcd.print("porções de ração no dia");

```



```

    while(sensor1 < 2 ){
sensor1 ++;

int estado2 = digitalRead(sensor1);//MONITORAMETO
Serial.print("Estado sensor1 : ");//MONITORAMETO
Serial.println(estado2);//MONITORAMETO
    int estado3 = digitalRead(solenoid);//MONITORAMETO
Serial.print("Estado solenoide : ");//MONITORAMETO
Serial.println(estado3);//MONITORAMETO
int estado4 = digitalRead(vibra);//MONITORAMETO
Serial.print("Estado vibra : ");//MONITORAMETO
Serial.println(estado4);//MONITORAMETO

if(sensorSOLENOIDE == 0)
{

digitalWrite(33, HIGH);//liga/desliga solenoide
delay(10000);

}
else
{
    digitalWrite(33, LOW);//liga/desliga solenoide

}

digitalWrite(22, HIGH); // liga o motor VIBRA
delay(500);             // tempo
digitalWrite(22, LOW);  // desliga o motor VIBRA
delay(10000);           // tempo

}

//ATENÇÃO
//SENSOR NIVEL AGUA

//vibra
digitalWrite(vibra, HIGH); // liga o motor VIBRA
delay(5000);               // tempo
digitalWrite(vibra, LOW);  // desliga o motor VIBRA
delay(10000);              // tempo


    digitalWrite(33, LOW);//liga/desliga solenoide
delay(6000);
    digitalWrite(8, LOW);//liga/desliga solenoide
delay(6000);

```

```
//solenoide com sensor nivel de agua  
  lcd.clear();  
  lcd.print(" ALIMENTADORA");  
  lcd.setCursor(2, 1);  
  lcd.print(" AUTOMATICA ");  
  delay(500);  
}}}}
```