

COMEDOURO AUTOMÁTICO PARA PETS

Sistemas Embarcados



Equipe

- Francisco Gilson Pereira Almeida Filho - 401066
- José Ícaro Santa Bernardes - 399037
- Lucas Pedrosa Valente - 398592



Sumário

- Apresentação.
- Introdução.
- Eletrônica.
- Simulação.
- Firmware.
- Conclusão.
- Upgrades futuros.
- Referências.



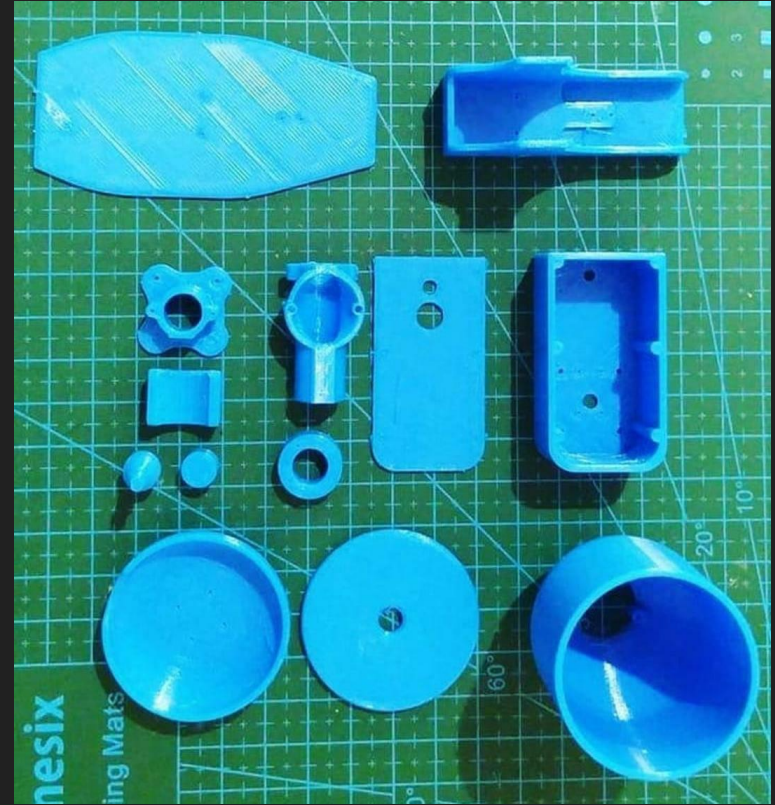
Introdução

- Esse trabalho apresenta o desenvolvimento, montagem e simulação de um Comedouro Automático para Pets.
- A parte estrutural do projeto foi produzida em uma impressora 3D com PLA azul.



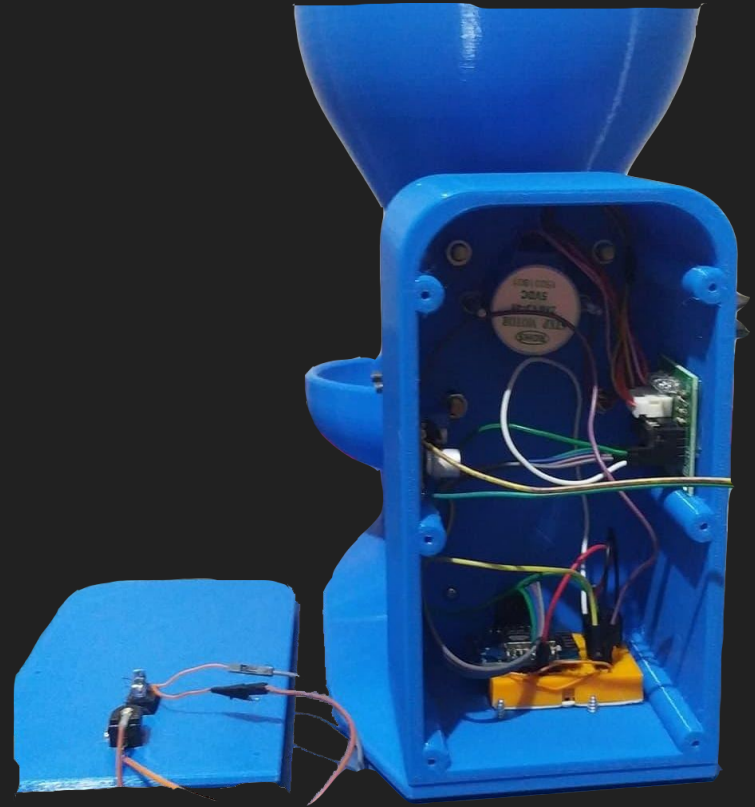
Introdução

- O design estrutural foi baseado em um modelo livre e adaptado para eletrônica em questão, o link para o modelo se encontra nas referências.
- No total foram 14 peças produzidas em mais de 30 horas de impressão.

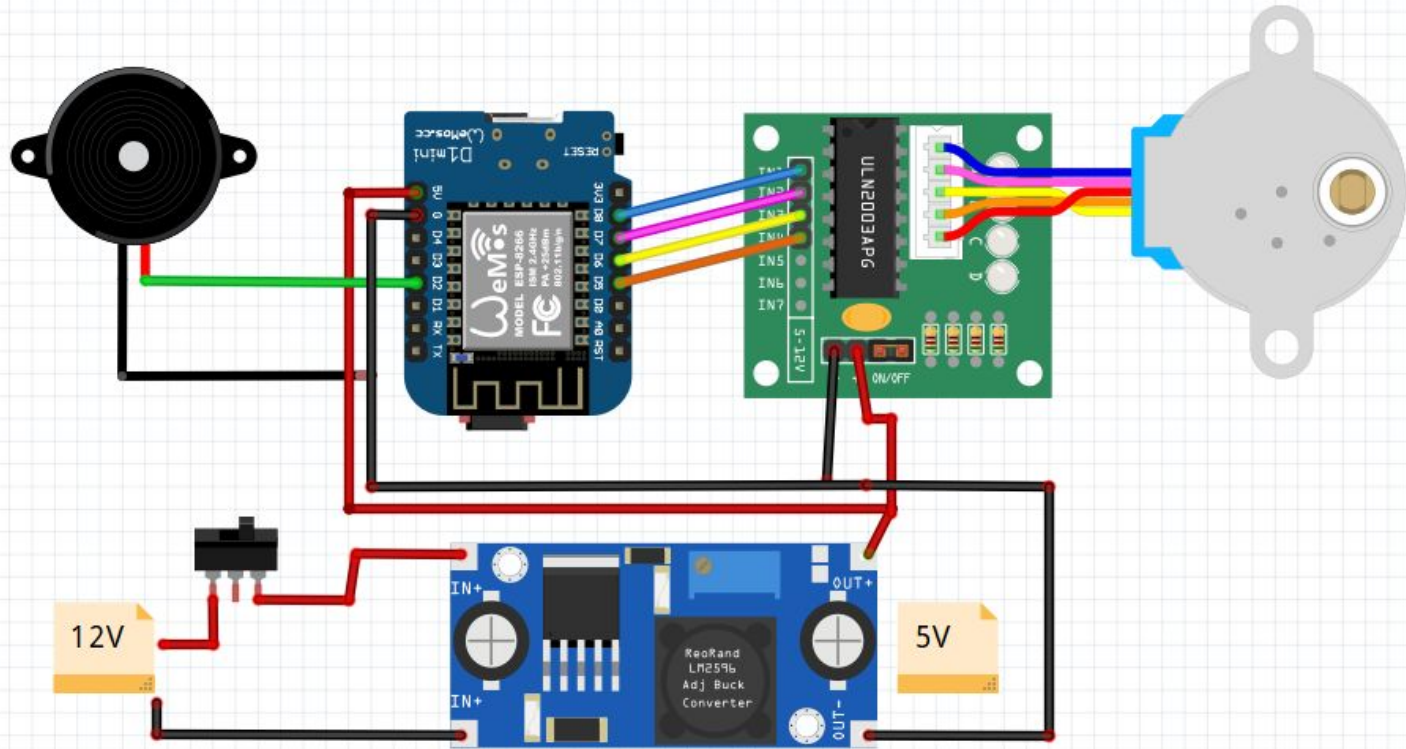


Eletrônica

- O projeto é controlado por um ESP12, um motor de passo 28BYJ-48 com driver ULN2003 que faz o controle de um barra rosqueada que é responsável por despejar a ração.
- Foi utilizado um buzzer passivo para efeitos sonoros.
- O comedouro tem capacidade máxima de 2 kg de armazenamento, dando uma autonomia de semanas para a alimentação de animais de pequeno porte.



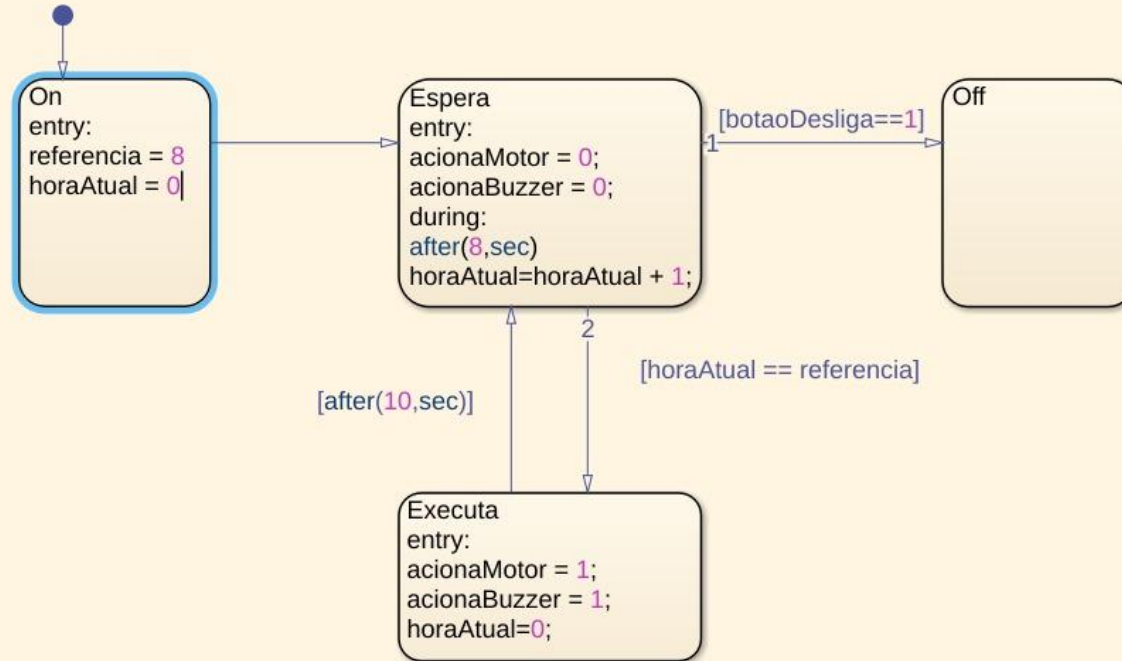
Eletrônica (Esquemático)



fritzing



Simulação Simulink



Firmware

```
void loop() {  
  
  //Musica antes de cair a comida  
  for (int thisNote = 0; thisNote < notes * 2; thisNote = thisNote + 2) {  
    divider = melody[thisNote + 1];  
    if (divider > 0) {  
      noteDuration = (wholenote) / divider;  
    } else if (divider < 0) {  
      noteDuration = (wholenote) / abs(divider);  
      noteDuration *= 1.5;  
    }  
    tone(buzzer, melody[thisNote], noteDuration * 0.9);  
    delay(noteDuration);  
    noTone(buzzer);  
  }  
  //Musica antes de cair a comida
```



Firmware

```
//LAÇO "for" QUE LIMITA O TEMPO EM QUE O MOTOR GIRA NO SENTIDO ANTI-HORÁRIO
for(int i = 0; i < 40; i++){ //PARA "i" IGUAL A 0, ENQUANTO "i" MENOR QUE 50 INCREMENTA "i"
    myStepper.step(-687); //GIRA O MOTOR NO SENTIDO ANTI-HORÁRIO
    Serial.println(i);
    delay(5

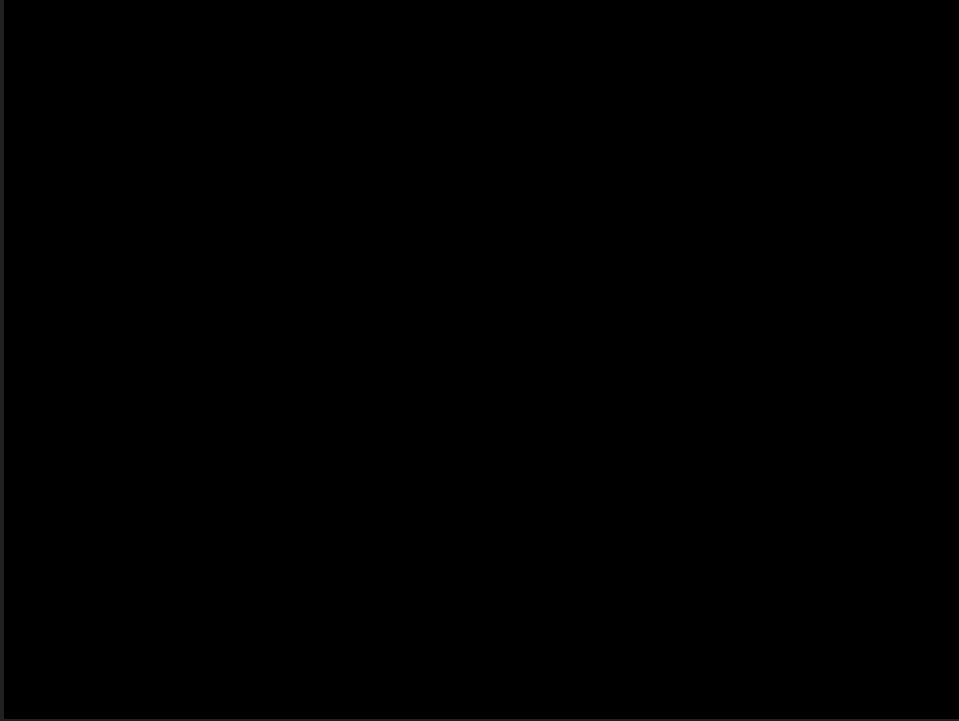
}

//Desligar o motor de passo
digitalWrite(15,0);
digitalWrite(14,0);
digitalWrite(13,0);
digitalWrite(12,0);
Serial.println("x");

//Rotina que espera 6 Horas
for(int y = 0; y < 21600; y++){
    delay(1000);
    Serial.println(y);
}
```



Resultados e Conclusão

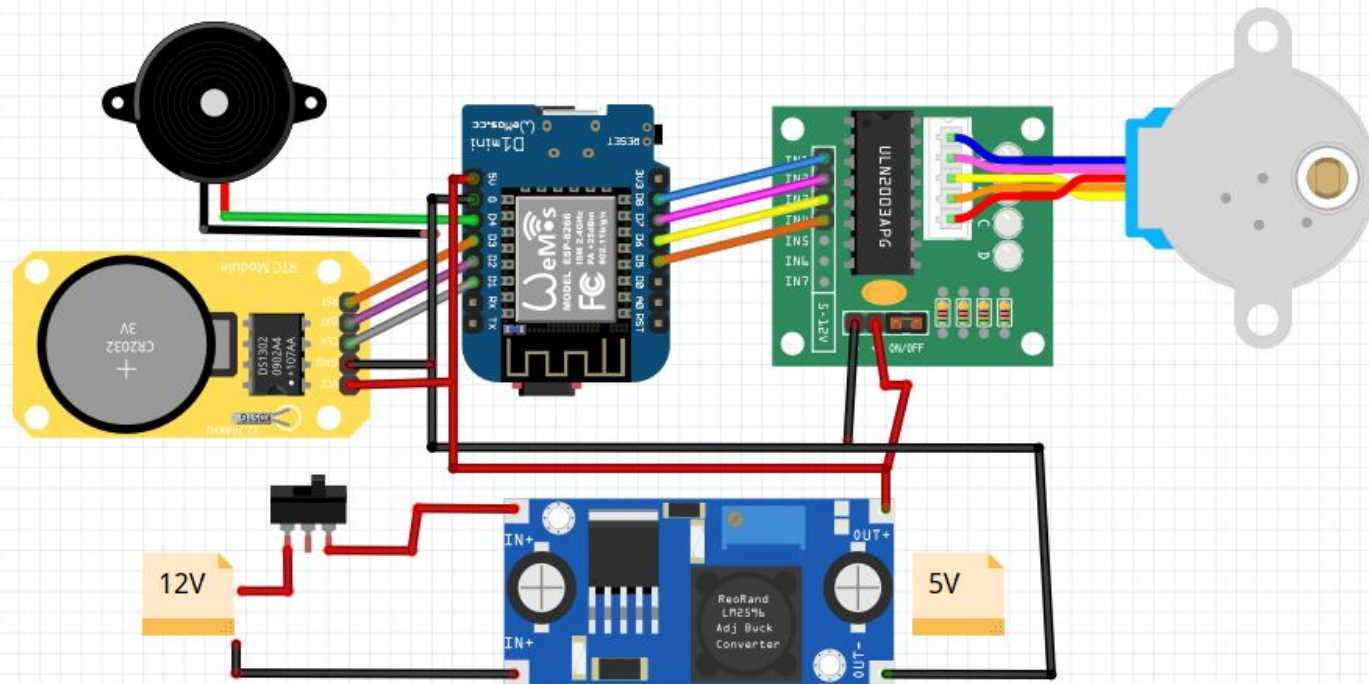


Upgrades Futuros

- Adicionar um módulo RTC para contagem do tempo mesmo com ausência de energia no ESP.
- Uso do módulo Wi-Fi do ESP para configurações de horários e quantidade de ração despejada, através de um aplicativo.
- Utilização do FreeRTOS para escalonamento das tarefas de despejo de ração e sons do buzzer.



Upgrades Futuros



Referências

1 - Fully automatic cat feeder. Disponível em: <https://www.thingiverse.com/thing:3623148>.

2 - Keyboardcat.ino. Song buzzer. Disponível em:

<https://github.com/robsoncouto/arduino-songs/blob/master/keyboardcat/keyboardcat.ino>





Perguntas?



Obrigado!