

# Alimentador automático para animais de estimação

Ponto de controle 4

Arthur Torres Magalhães – 15/0006063  
Universidade de Brasília - UnB  
Brasília-DF - Brasil  
arthurtorres26@outlook.com

Jovelino Caetano Braz Junior – 14/0043641  
Universidade de Brasília - UnB  
Brasília-DF - Brasil  
jovelinocbjunior@gmail.com

**Resumo** — Este trabalho propõe a criação de um protótipo com o uso de conceitos de sistemas embarcados e do microcontrolador MSP430, para a implementação de um alimentador automático para animais de estimação, com custo muito menor em relação a outros produtos com a mesma finalidade.

**Palavras chaves** – MSP430; alimentador automático; animais de estimação.

## I. JUSTIFICATIVA

Segundo dados do IBGE em parceria com a Abinpet (Associação Brasileira da Indústria de Produtos para Animais de Estimação), a população total de animais de estimação no Brasil é a 4ª maior do mundo, com mais de 130 milhões de animais. Se forem contados somente a população total de cachorros, gatos e aves canoras e ornamentais (mais de 112 milhões), o país se torna o 2º maior no mundo [1].

Esses animais precisam ser alimentados de tempos em tempos e essa é uma tarefa aparentemente simples para os donos. Entretanto devido a correria e as ocupações do dia-a-dia, os donos desses animais podem acabar esquecendo de alimentá-los, causando um mal tanto nos animais, que ficam sem se alimentar, quanto nos donos que ficam com um mal-estar e sentimento de culpa ao descobrir que esqueceram de fazer algo tão básico. Além do mais, deixar de alimentar animais domésticos é considerado maus tratos com o animal, que por sua vez é crime [2]. Outro ponto a ser observado é que nem todos os animais tem noção do que devem comer, nem dos horários. Eles tendem a comer toda comida da tigela assim que ela é colocada, o que pode trazer problemas de peso e até de ansiedade neles.

Portanto, uma solução para isto está no projeto de um alimentador automatizado de ração para animais de estimação, capaz de encher o recipiente do animal em intervalos programados de tempo e de maneira remota, não permitindo que este animal fique sem acesso à comida caso o dono esqueça ou não esteja em casa no momento (esteja numa viagem, por exemplo).

## II. OBJETIVO

Elaboração de um projeto com protótipo de um alimentador automatizado para animais de estimação, que despeje ração em

intervalos regulares de tempo e de maneira remota, capaz de avisar ao dono quando o reservatório de ração está perto de ficar vazio.

## III. REQUISITOS

A. Disponibilizar ao usuário a possibilidade de determinar o intervalo de tempo em que serão despejadas as rações, assim como a quantidade desejada;

Tal funcionalidade será possível a partir de uma interface, que se comunicará com o MSP430 a partir de um módulo Wi-Fi. O usuário será capaz de determinar um intervalo de tempo (como um timer) onde a ração será despejada.

B. Despejar a quantidade de ração desejada pelo usuário nas horas programadas;

Tal função se torna possível pelo fato de que o tempo em que a “porta” do dispositivo está aberta, ou seja, o tempo em que a ração está sendo despejada, determina a vazão de ração. Consequentemente, ao se determinar o tempo de abertura da “porta”, pode-se determinar a quantidade de ração despejada.

C. Disponibilizar ao usuário a possibilidade de acionar o dispositivo através de um botão;

O usuário será capaz de ativar o sistema através de um botão físico no dispositivo, independentemente do tempo programado para o timer.

D. Disponibilizar ao usuário acesso remoto ao dispositivo;

O usuário será capaz de ativar o sistema através de uma interface via Wi-Fi, independentemente do tempo programado para o timer.

E. Disponibilizar ao usuário a informação de que o reservatório está perto de esvaziar;

O sensor ultrassônico medirá a distância do nível de ração para a tampa do reservatório. Com isso, a quantidade de ração no reservatório será medida. Quando esta quantidade atingir um limite crítico (inicialmente foi estabelecido 10% da quantidade), uma notificação é enviada ao usuário via Wi-Fi.

F. Ter uma alimentação de energia que dure dias.

O dispositivo deve ser capaz de se manter ligado por uma quantidade considerável de dias.

#### IV. TABELA DE MATERIAIS

Tabela 1. Materiais usados no projeto.

Quantidade	Equipamento	Marca
01	MSP430G2553LP	Texas Instruments
01	Módulo Wi-Fi – ESP8266	Espressif Systems
01	Servo Motor – SG90	Tower Pro
01	Sensor de Distância Ultrassônico HC-SR04	ElecFreaks
-	Jumpers	-
01	Protoboard	Hikari
01	Bateria 9V	-
01	Tabua de madeira	-
01	Cano PVC	-
01	Bocal	-

#### V. HARDWARE E SOFTWARE

O hardware do projeto engloba quatro pontos principais: a placa MSP430, o servo motor, o sensor ultrassônico e o módulo Wi-Fi. Ou seja, basicamente, envolve um microcontrolador, um sensor, um atuador e um responsável pela comunicação.

A placa MSP430 por ser um dos objetos de estudo da disciplina, será utilizada como um microcontrolador no projeto, ou seja, toda a programação lógica para resolução do problema estará nessa placa. Ela funciona como o “cérebro” do sistema. A versão utilizada será a MSP-EXP430G2553.



Fig. 1. MSP430G2553 da Texas Instruments.

O servo-motor escolhido para a aplicação foi o SG90. Sua escolha foi devido ao fato desse motor ser capaz de realizar movimentos entre 0° e 180° (suficientes para a realização do projeto) e ter uma programação de fácil entendimento, que permite modificar a angulação e a velocidade do movimento. Além disso, ele tem um torque de 1,2 kg.cm (4,8V) a 1,6 kg.cm (6,0V) e é mais acessível economicamente, custando menos de

R\$ 20,00. No projeto, ele realiza o movimento responsável por liberar comida do reservatório.



Fig. 2. Micro Servo SG90.

O módulo Wi-Fi por sua vez, é o responsável por fazer a comunicação entre o equipamento com o usuário através da internet, seja para que o usuário defina o timer, seja para que ele receba notificações sobre o estado de risco de falta de ração do reservatório.



Fig. 3. NodeMCU ESP8266.

Já o sensor ultrassônico HC-SR04 será usado no projeto para determinar a quantidade de ração contida dentro do reservatório. Terá como principal função retornar ao MSP430 o valor da distância do nível de ração até a tampa do reservatório e assim o usuário receberá um aviso indicando risco para a falta de ração. Esse aviso será determinado quando a ração no reservatório representar 10% do total do compartimento de ração.



Fig. 4. Sensor HC-SR04.

O software utilizado na integração dos hardwares com o microcontrolador foi o Code Composer. Conseguiu-se fazer o código do servo motor e do sensor ultrassônico, tal código, assim como os comentários explicando o que cada parte do código faz, pode ser visto no Anexo 1 deste relatório e em [3].

O fluxograma inicial, que define basicamente o funcionamento do projeto pode ser visto na Fig. 5 a seguir.

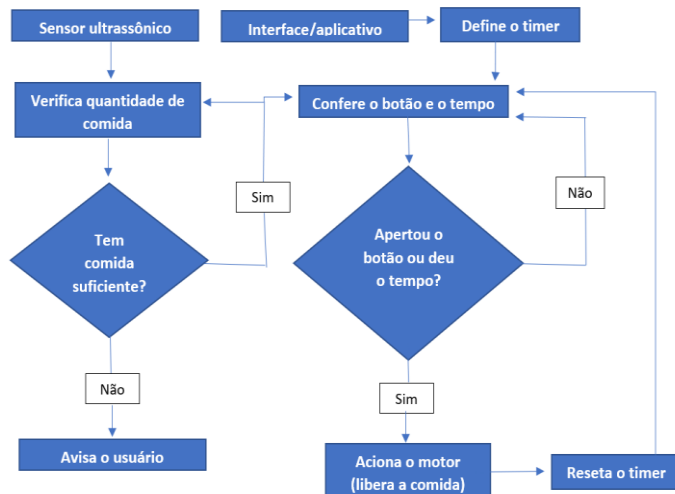


Fig. 5. Fluxograma do projeto.

## VI. ESTRUTURA

A estrutura foi montada utilizando uma tábua de madeira, uma haste de madeira para o reservatório e o cano PVC para o depósito de ração. A tabua foi desenhada e cortada para dar um design mais bonito e ter uma base bem firme e rente ao chão, para não ter queda do reservatório para nenhum dos lados. A haste de madeira foi colada e pregada com a base para com o uso de arame ser presa ou depósito de ração. Para controlar a vazão de ração pelo bocal, foi utilizado duas buchas de cano.



Fig. 6. Estrutura do projeto

É importante frisar que toda a construção da estrutura, foi feita com materiais recicláveis, de baixíssimo custo. Tendo em vista o escopo da disciplina.

## VII. BENEFÍCIOS

O uso de um alimentador automático beneficia pessoas com animais de estimação que não ficam muito tempo em casa, que estão constantemente viajando ou até mesmo aquelas pessoas que constantemente tem imprevistos que as impedem de alimentar seus animais na hora correta.

Também é muito útil para pessoas que tem animais com problemas alimentares (animais com sobrepeso, diabéticos...) e que precisam ser alimentados em intervalos certos de tempo, com quantidades controladas.

Além disso, um alimentador automático permite que o dono possa definir as quantidades e horários em que a ração será despejada, o que ajudaria na criação de uma rotina mais saudável para os animais. Ao fazer isso, o dono além de ter mais comodidade ainda pode economizar tempo.

## VIII. REVISÃO BIBLIOGRÁFICA

Atualmente, existem vários modelos de alimentadores automáticos para animais de estimação no mercado. Eles podem ser feitos especificamente para cães, gatos, pássaros ou até mesmo para peixes. Alguns desses modelos oferecem muitas funcionalidades, o que pode elevar bastante o preço. Dois exemplos desses modelos são:

### A. Alimentador eletrônico Hoison: babá robô para pet [4]

#### 1) Funcionalidades:

- a) Alimentador programável por aplicativo;
- b) Comedouro removível e lavável;
- c) Capacidade de até 2kg de ração;
- d) Liberação de ração
  - Automaticamente nos horários programados no aplicativo;
  - Remotamente por app, sem necessidade de programação;
  - Manualmente, pressionando o botão no topo do aparelho.

e) Necessária rede Wi-fi para estabelecer conexão com smartphone.

#### 2) Custo: R\$ 1.349,90.

#### IX.

### A. Alimentador Eletrônico Eatwell Plus 5 refeições – Amicus [5]

#### 1) Funcionalidades:

- a) Evita a exposição prolongada de alimentos no ambiente;
- b) Compartimentos de comida removível que facilita a higienização;
- c) Até cinco refeições programadas por hora;

2) *Custo: R\$ 549,90.*

Também existem projetos de alimentadores eletrônicos que são construídos de forma mais simples, porém com custo menor de fabricação e que viabilizam o mesmo objetivo dos demais. Não necessariamente estes projetos são comercializados. Um exemplo disso é o projeto feito através do uso do Arduino [6], que assim como diversos projetos encontrados, se enquadra melhor no escopo da disciplina.

## X. RESULTADOS

Neste ponto de controle quatro, os avanços foram significativos se comparados ao ponto de controle anterior. Os códigos do sensor ultrassônico e do servo motor, que até então foram desenvolvidos no Energia, foram refinados em linguagem C, através do Code Composer e testados com sucesso.

Além disso, a estrutura de suporte para o dispositivo foi montada com materiais de baixo custo. A estrutura, apesar de estar com as principais funcionalidades instaladas, ainda precisa de ajustes, após realizado vários testes, como por exemplo, um melhor suporte para o servo motor e tampas para o reservatório. Outro ponto a ser melhorado na estrutura é em relação a saída da ração, uma vez que após os testes realizados foi possível perceber que dependendo do tamanho, a comida

pode emperrar no gargalo. O resultado de diminuir a vazão de ração foi bem satisfatório, uma vez que o controle de saída de ração é de suma importância nesse projeto.

Por fim, ainda não conseguiu-se estabelecer a comunicação Wi-Fi para o dispositivo. Espera-se até a data final de entrega, que tal funcionalidade esteja funcionando.

## REFERÊNCIAS

- [1] IBGE. População de animais de estimação no Brasil – 2013. Disponível em: <http://www.agricultura.gov.br/assuntos/camaras-setoriais-tematicas/documentos/camaras-tematicas/insumos-agropecuarias/anos-antiores/ibge-populacao-de-animais-de-estimacao-no-brasil-2013-abinpet-79.pdf>. Acesso em 4 de setembro de 2018.
- [2] Brasil. Lei nº 9.605 de 12 de Fevereiro de 1998. Disponível em: [http://www.planalto.gov.br/ccivil\\_03/LEIS/L9605.htm](http://www.planalto.gov.br/ccivil_03/LEIS/L9605.htm). Acessado em 4 de setembro de 2018.
- [3] Github. Disponível: [https://github.com/arthurtorrs/Eletronica-Embarcada/tree/master/2\\_PCs/Codigos](https://github.com/arthurtorrs/Eletronica-Embarcada/tree/master/2_PCs/Codigos)
- [4] Petlove. Disponível em: [https://www.petlove.com.br/hoison?destaque=alimentador-eletronico-hoison-baba-robo-para-pet&sku=1571337&gclid=CjwKCAjw2rjcBRBuEiwAheKeL8s7U7LQdUQnehyGO5Im-zucs7HsaKT1QRsox7gYqfsjv2UOH1r8BoCNK0QAvD\\_BwE](https://www.petlove.com.br/hoison?destaque=alimentador-eletronico-hoison-baba-robo-para-pet&sku=1571337&gclid=CjwKCAjw2rjcBRBuEiwAheKeL8s7U7LQdUQnehyGO5Im-zucs7HsaKT1QRsox7gYqfsjv2UOH1r8BoCNK0QAvD_BwE). Acesso em 4 de setembro de 2018.
- [5] Petlove. Disponível em: <https://www.petlove.com.br/amicus?destaque=alimentador-eletronico-eatwell-plus-5-refeicoes---amicus-3102969&sku=3102969>. Acesso em 4 de setembro de 2018.
- [6] Arduino Project Hub. Disponível em: [https://create.arduino.cc/projecthub/circuito-io-team/iot-pet-feeder-10a4f3?ref=tag&ref\\_id=pets&offset=0](https://create.arduino.cc/projecthub/circuito-io-team/iot-pet-feeder-10a4f3?ref=tag&ref_id=pets&offset=0). Acesso em 3 de setembro de 2018.

## Anexo 1

### Código

```
#include <msp430.h>

#define LED BIT6

int miliseconds;
int distance;
long sensor;

void delay()
{
    P1OUT ^= BIT0;
    volatile unsigned long i;
    i = 29999;
    do (i--);
    while (i != 0);
}

int main(void)
{
    BCSCTL1 = CALBC1_1MHZ;
    DCOCTL = CALDCO_1MHZ;           // Clock de 1Mhz
    WDTCTL = WDTPW + WDTHOLD;       // Desliga o WatchdogTimer

    CCTL0 = CCIE;                   // Habilita a interrupção pelo CCR0
    CCR0 = 1000;                     // 1ms em 1mhz
    TACTL = TASSEL_2 + MC_1;         // SMCLK, upmode

    P1IFG = 0x00;                   // Limpa todas as flagas de interrupção
    P1DIR |= 0x01;                   // P1.0 como saída do LED
    P1OUT &= ~0x01;                  // Desliga o LED

    _BIS_SR(GIE);                   // Habilita a interrupcao global

    while (1)
    {
        P1IE &= ~0x01;              // Desabilita a interrupcao
        P1DIR |= 0x02;               // Pino do TRIGGER (P1.1) como saída
        P1OUT |= 0x02;               // Gera o pulso
        __delay_cycles(10);           // Delay de 10us
        P1OUT &= ~0x02;               // Para o pulso
        P1DIR &= ~0x04;               // Pino do ECHO (P1.2) como entrada
        P1IFG = 0x00;                 // Limpa a flag, por precaução
        P1IE |= 0x04;                 // Habilita a interrupção pelo pino do ECHO
        P1IES &= ~0x04;               // Define a borda de subida no pino ECHO
        __delay_cycles(30000);         // Delay de 30ms (depois desse tempo, o ECHO acaba se nenhum objeto for detectado)
        distance = sensor / 58;        // Converting o valor de ECHO para cm

        P1DIR |= BIT6;                // P1.6/TA0.1 É USADO PARA O PWM, QUE FUNCIONA COMO A SAÍDA -> servo 1
        P1OUT = 0;                    // LIMPA AS SAIDAS P1
        P1SEL |= BIT6;                // P1.6 SELECIONA TA0.1

        P1DIR &= ~BIT3;
        P1REN |= BIT3;
        P1OUT |= BIT3;
```

```

// COMO O CLOCK É DE 1MHz, E 1000ms EQUIVALEM A 1Hz, ENTAO SETANDO CCR0 PARA 20000
// TEMOS QUE 20000(1000000 / 1000 * 20) É UM PERIODO DE 20ms

TA0CCR0 = 20000-1;          // PERIODO DO PWM TA0.1
TA1CCR0 = 20000-1;          // PERIODO DO PWM TA1.1

// SETANDO 1500 É 1.5ms QUE É 0deg. (servo pos)
TA0CCR1 = 2000;             // CCR1 PWM duty cycle
TA1CCR1 = 2000;             // CCR1 PWM duty cycle

TA0CCTL1 = OUTMOD_7;        // CCR1 reset/set
TA0CTL = TASSEL_2 + MC_1;   // SMCLK, up mode
TA1CCTL1 = OUTMOD_7;        // CCR1 reset/set
TA1CTL = TASSEL_2 + MC_1;   // SMCLK, up mode

if (distance < 20 && distance != 0) {
    P1OUT |= 0x01;          // Se a distancia for menor que 20cm e diferente de 0, o LED acende
}
else {
    P1OUT &= ~0x01;          // Caso contrario, ele permanece desligado.
}

if((P1IN&BIT3)==0){
    delay();
    TA0CCR1 = 1000;
    TA1CCR1 = 2000;          //Ativa o servo (abre)
    delay();
    TA0CCR1 = 2000;
    TA1CCR1 = 1000;          //Ativa o servo (fecha)
}
}
}

#pragma vector=PORT1_VECTOR
__interrupt void Port_1(void)
{
    if (P1IFG & 0x04)        // Verifica se tem alguma interrupcao pendente
    {
        if (!(P1IES & 0x04)) // Verifica se tem uma borda de subida
        {
            TACTL |= TACLR;    // Se tem, limpa o timer A
            microseconds = 0;
            P1IES |= 0x04;      // Borda de descida
        }
        else
        {
            sensor = (long)microseconds * 1000 + (long)TAR; // Calcula o comprimento de ECHO
        }
        P1IFG &= ~0x04;        // Limpa a flag
    }
}

#pragma vector=TIMER0_A0_VECTOR
__interrupt void Timer_A (void)
{
    microseconds++;
}

```