

Proposta de Projeto

Cinto de Ecolocalização para Cegos

Adriano Silva de Moraes
Faculdade do Gama
Universidade de Brasília - UnB
Gama-DF, Brasil

Hachid Habib Cury
Faculdade do Gama
Universidade de Brasília - UnB
Gama-DF, Brasil

I. JUSTIFICATIVA

Do total da população brasileira, 23,9% (45,6 milhões de pessoas) declararam ter algum tipo de deficiência. Entre as deficiências declaradas, a mais comum foi a visual, atingindo 3,5% da população brasileira. Entre os deficientes visuais 528.624 pessoas são incapazes de enxergar (cegos), de acordo com os dados do IBGE de 2010. Visando incluir da melhor maneira os deficientes visuais e resolver problemas que os portadores de deficiência visual enfrentam, como a mobilidade reduzida e a dependência de algum auxílio como cães guias, bengalas, parentes, amigos e etc.

O mercado oferece produtos similares ao proposto no projeto. As bengalas sensoriais funcionam com sensores ultrassônicos que possibilitam a identificação que obstáculos no caminho do usuário, o preço desse produto chega a 1350.00 €, ou seja R\$ 5895.00 como mostrado na figura 1.



Figura 1: Bengala sensorial comparação de preço.

O mesmo projeto já foi trabalhado por engenheiros como o brasileiro Rodrigo Azevedo. Ele é um dos responsáveis pelo projeto Bengala IoT, que visa criar um modelo replicável de uma bengala inteligente para deficientes visuais. De código aberto e sem fins lucrativos, conforme o criador do projeto, o pacote básico que é composto por dois sensores e um microcontrolador simples, poderia sair na faixa dos R\$ 100 ou R\$ 200.

II. OBJETIVOS

A dupla propõe a criação de um cinto eletrônico composto por sensores ultrassônicos e vibradores, que ofereçam ao usuário a capacidade de se orientar no ambiente sem os incômodos gerados pelas bengalas, que são comumente utilizadas, já que os produtos que utilizam a mesma tecnologia proposta possuem preços exorbitantes para a maioria da população cega.

III. BENEFÍCIOS

Os usuários devem tirar os mesmos proveitos que as ferramentas atuais, ou seja, saber com antecedência a distância dos obstáculos, evitando assim acidentes de locomoção, dar mais liberdade ao usuário para se locomover em público com mais autoconfiança, mas com o diferencial de utilizar um equipamento sutil e sem inconvenientes para tal.

IV. HARDWARE

Lista de materiais:

- MSP-EXP430G2553LP
- Quatro sensores de distância ultrassônicos (HC-SR04)
- Quatro vibracalls (Modelo F1030300112)
- Resistores 4.7 K Ω
- Resistores 10K Ω
- Quatro Transistores BCE549B
- Protoboard
- Jumpers
- Cinta elástica

Descrição do Hardware:

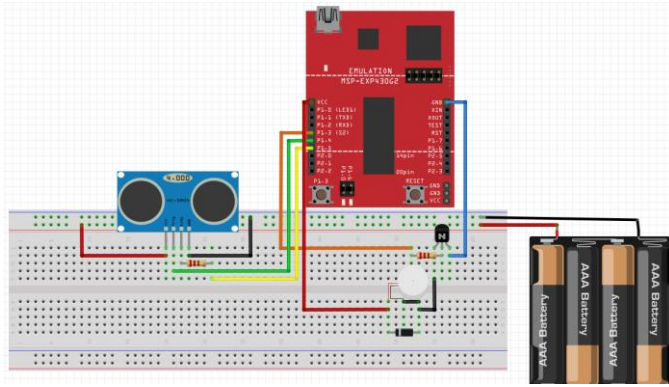


Figura 2: Esquemático do circuito apresentado.

A cinta elastica será a base para os quatro sensores, microcontrolador e para os demais dispositivos eletrônicos, conterá toda a circuitaria.

Os sensores ficarão distribuídos da seguinte maneira, 1 sensor apontado para cima de modo que identifique objetos a 50 cm do rosto do usuário, 1 sensor virado para baixo para que identifique obstáculos a 50 cm dos pés do mesmo e 2 sensores virados para frente, para que além de mostrar ao usuário obstáculos a 1 metro a sua frente, distingue em que lado se encontram, possibilitando assim ao utilizador da cinta de ecolocalização saber onde está a obstrução e a que distância se encontra do mesmo. Para isso, serão distribuídos vibracall's ao longo da cinta que funcionarão da seguinte forma, os vibracall da esquerda e da direita mostram objetos que estão na esquerda e à direita do usuário e sua proximidade por vibração (objetos mais pertos vibraram mais que objetos mais longes), o sensor de cima vibrará com força total quando identificar algo perto do rosto do usuário e o sensor de baixo terá dois modos de vibração, vibrará ininterruptamente para degraus, meu fios e etc, e múltiplas vibrações para buracos.

V. SOFTWARE

O código foi criado para ser compilado pelo software Code Composer Studio e possui 2 funções, a função Atraso_us é utilizada para criar atrasos em microsegundos e é utilizada no código para controle do tempo mínimo que Trigger do sensor ultrassônico fique em alto. A função SensorUlt foi criada para que o projetista informe a posição do Trigger e Echo no MSP430 e seu retorno é a distância em centímetros, para isso é necessário levar Trigger para o nível alto durante 10µs e esperar que Echo retorne em nível alto, com isso é utilizado o TIMER_A2 do MSP em modo de captura para ler o tempo do pulso em microsegundos. Com a leitura do pulso utiliza-se a seguinte equação para descobrir a distância do objeto a frente do sensor.

$$Distancia = \frac{TPulso \times V_{Som}}{2} \simeq \frac{TPulso}{58}$$

Já que a velocidade do som igual a:

$$V_{Som} = 0,0343 \frac{cm}{\mu s}$$

A main do programa possui um loop infinito que chama a função SensorUlt e joga seu retorno em uma variável inteira nomeada "distancia", o valor de distancia é comparado com faixas de distâncias e se seu valor encaixar na faixa é dado um valor para DUTY_CYCLE que com a utilização do TIMER_A1 em modo comparação é transmitido para a saída Vibracall e LED uma onda quadrada PWM. A execução da main fornece aos Led2 e ao motor vibracall uma intensidade maior com a diminuição da distância entre o sensor e o objeto, ou seja, quanto mais perto do objeto maior deve ser a vibração do motor e a luminosidade do led.

VI. DISCUSSÃO

Nessa etapa do projeto a dupla se deparou com a dificuldade na alimentação do circuito devido a capacidade do MSP430 de oferecer uma corrente máxima de 10 mA por pino, sendo que só um motor vibracall necessita de 58 mA, sendo então necessário a utilização de um circuito aparte de alimentação para os vibracall's.

Outra questão a ser reparada no código é que o ciclo de trabalho e a frequência da onda deve ser ajustada já que não é possível notar a diferença de vibração com a diminuição da distância entre o objeto teste e o sensor ultrassônico, esse efeito é facilmente visto no led.

VII. CONCLUSÃO

A evolução do trabalho ocorreu como desejado, já que o projeto já está sendo executado no software CCS e todos elementos do projeto estão em funcionamento, faltando apenas poucos ajustes. O próximo desafio será a união de todas componentes com suas devidas alimentações na cinta elástica.

VIII. REVISÃO BIBLIOGRÁFICA

- [1] Davies, J., MSP430 Microcontroller Basics, Elsevier, 2008.
- [2] BRASIL, IBGE. Pesquisa sobre quantidade de cegos no Brasil, Censo Demográfico, 2010. Disponível em: <www.ibge.gov.br>. Acesso em: 30/04/2018.
- [3] Bengala eletrônica Ultracane, comparativo de preço no mercado. Disponível em: <<https://www.megaserafim.pt/online/orientacao-e-mobilidade/41-bengala-electronica-ultracane.html>>. Acesso em: 03/06/2018.
- [4] Brasileiro cria bengala inteligente que usa sensores para guiar deficientes. Disponível em: <<https://www.tudocelular.com/tech/noticias/n119414/cpbr11-brasileiro-bengala-inteligente-iot.html>>. Acesso em: 03/06/2018.

IX. ANEXOS

```
#include <msp430.h>
#define LED (BIT6 + BIT3)
#define Trig0 BIT4
#define Echo0 BIT5

#define PERIODO 5000

unsigned int TempoI=0;
unsigned int TempoF=0;

//Função Criada para causar atraso em
microsegundos
void Atraso_us(volatile unsigned int us)
{
    TACCR0 = us-1;
    TACTL = TASSEL_2 + ID_0 + MC_1;

    while((TACTL & TAIFG)==0);
    TACTL = TACLR;
}

//Função criada para controlar sensores
ultrassônicos
int SensorUlt(volatile unsigned int Trig,
volatile unsigned int Echo)
{
    int distancia;

    P1DIR |= Trig;
    P1OUT &= ~Trig;
    P1DIR &= ~Echo;
    //Colocar Trig em alto durante 10 us
    P1OUT |= Trig;
    Atraso_us(10);
    // Zerando o TAR
    TACTL = TACLR;
    // Configurando clock do timer
    TACTL = TASSEL_2 + ID_0 + MC_2;
    // Configurando modo de captura
    TACCTL2 = CM_3 + CCIS_0 + CAP;
    P1OUT &= ~Trig;

    while((P1IN&Echo)==0);
    TempoI=TAR;
    while((P1IN&Echo)==Echo);
    TempoF=TAR;
    // Calculando a distancia em centimetros
    distancia = (TempoF - TempoI)/58;
    return distancia;
}void main(void)
{
    BCSCTL1 = CALBC1_1MHZ;
    DCOCTL = CALDCO_1MHZ;
```

```
WDTCTL = WDTPW + WDTOLD;

int distancia;

P1DIR |= LED;
P1SEL |= LED;
P1SEL2 &= ~LED;

int DUTY_CYCLE=2500;

TACCR0 = PERIODO-1;
TACCR1 = DUTY_CYCLE-1;
TACCTL1 = OUTMOD_7;
TACTL = TASSEL_2 + ID_0 + MC_1;

while(1){
    distancia= SensorUlt(Trig0, Echo0);

    TACCR0 = PERIODO-1;
    TACCR1 = DUTY_CYCLE-1;
    TACCTL1 = OUTMOD_7;
    TACTL = TASSEL_2 + ID_0 + MC_1;

    if(distancia<=10){
        P1SEL |= LED;
        DUTY_CYCLE= PERIODO-2;
    }
    else if (distancia<=30){
        P1SEL |= LED;
        DUTY_CYCLE= PERIODO*0.30;
    }
    else if (distancia<=50){
        P1SEL |= LED;
        DUTY_CYCLE= PERIODO*0.1;
    }
    else if (distancia<=100){
        P1SEL&= ~LED;
        P1OUT&=~LED;
    }else {
        P1SEL&= ~LED;
        P1OUT&=~LED;
    }
}
```