

Jogo de Mira a Laser

Ponto de Controle 4

Pedro Calile - 15/0144989

Victor Santos- 10/48473

RESUMO

O ponto de controle 4 tem como intuítos principais expor o refinamento do protótipo em linguagem C, interrupções e sub-rotina do código em Assembly, acrescentando temporização do sistema bem como recursos de automação ao sistema visando os resultados esperados para o projeto final da disciplina Microprocessadores e Microcontroladores.

INTRODUÇÃO

Proposta do Projeto

A escolha do Jogo de *Mira à Laser* vem da diversão e desafio que esse jogo propõe. Neste jogo o participante controla uma arma que emite laser em um alvo posto à frente do jogador. Dependendo do local em que atinja o feixe de laser, existe uma pontuação atribuída que se soma às chances anteriores. O jogo tem como vitorioso o jogador que vencer o oponente, caso seja jogado em dupla, ou ultrapassar seu próprio recorde.

Além da ideia lúdica existente no jogo, trazemos a proposta de inovar o jogo com sensores que medirão tremulações e aspectos fisiológicos do jogador para fazer análises e medições. Esses dados serão coletados e serão de grande valia para obter diagnóstico e desenvolvimento de terapias específicas e individualizadas de pacientes com doenças com sintomas onde a coordenação motora é prejudicada.

Serão utilizados os recursos de conhecimentos adquiridos na disciplina, portanto tendo em mente o jogo a ser projetado, a implementação necessitará de conhecimentos de operação de hardware e software.

DESENVOLVIMENTO

Descrição do Hardware

O microcontrolador utilizado no projeto será o MSP430, que consiste em um microcontrolador de propósito geral de baixo consumo de potência, desenvolvido pela *Texas Instruments* na década de 1990. Devido às características de baixíssimo consumo de energia, alto desempenho e baixo custo, o microcontrolador MSP430 torna-se extremamente popular e indicado para implementação do *videogame* proposto.

Serão utilizados dois MSP430, pois serão necessários 2 comandos, um para a arma que aciona/atira o laser e registrada os movimentos, e outro para o alvo que registra os sinais luminosos.

Além dos MSPs, serão utilizados:

- Protoboard: plataforma base para montagem dos circuitos;
- Demultiplexador CI 74154: conduzir ou não um transistor;
- Transistores (pnp): permitir quando alimentado com sinal baixo em cada coluna da matriz;

- Shift Register CI 74LS164;
- 4 LDRs ;
- Arma de Laser;
- Base de Madeira para suporte dos sensores de luz (LDRs);
- Push Button para atirar o laser,;
- Jumpers: conexões.

Especificações do LDR:

O **LDR** (Resistor Dependente da Luz) é um tipo especial de resistor que apresenta uma mudança em sua característica de resistência elétrica quando submetido à ação da luz. Apresentam um valor de resistência elevada em um ambiente escuro, e quando expostos à luz, têm a condutividade aumentada. Ou seja, oferecem baixa resistência elétrica quando iluminados.

Os LDRs são fabricados com materiais de alta resistência, como por exemplo o Sulfeto de Cádmio (CdS) ou o Sulfeto de Chumbo (PbS). Esses materiais possuem poucos elétrons livres quando colocados em ambiente escuro, e liberam elétrons quando há incidência de luz sobre eles, aumentando sua condutividade. Chamamos a esse efeito de **Fotocondutividade**. Quando cessa a incidência de luz sobre o componente, os elétrons retornam à camada de valência e a resistência do material volta a aumentar.

No geral, os LDRs possuem resistências entre 1 M Ω e 10 M Ω no quando em ambiente escuro, e resistências que podem chegar a menos de 100 Ω quando iluminados. A resistência do LDR não varia instantaneamente quando suas condições luminosas variam. Existe uma latência (tempo) que decorre até que a resistência atinja um valor estável. Expressamos esse tempo de recuperação em K Ω /s, sendo um valor típico cerca de 200K Ω /s na passagem de um ambiente claro para um escuro, e muito mais rápido no sentido inverso - em torno de apenas 10 ms na transição do escuro para o iluminado. Esse é um fator importante que deve ser levado em consideração ao criar projetos que utilizem LDRs para, por exemplo, acionar dispositivos de acordo com o nível de iluminação ambiente.

Descrição do Software

Seguindo o fluxograma da figura 1, foi levantado três códigos em linguagem C voltado para a operação do microcontrolador MSP, linguagem a qual aborda o funcionamento do protótipo objeto deste trabalho. O primeiro código refere-se a verificação e comando do jogo que tem sua operação por meio do MSP Master. Neste código indica ao usuário se a seção desejada do alvo foi atingida.

O segundo código trás o feedback por meio do MSP Slave ao MSP Master quanto ao acerto do alvo, seguindo o princípio de que quando acertado o alvo a resistência do LDR muda de valor e o MSP Slave detecta essa variação indicando que foi acertado o alvo corretamente.

O terceiro código aborda a temporização do LASER, uma vez que o mesmo não pode ficar aceso por mais do que um determinado intervalo de tempo visando seguir regras e o escopo do projeto.

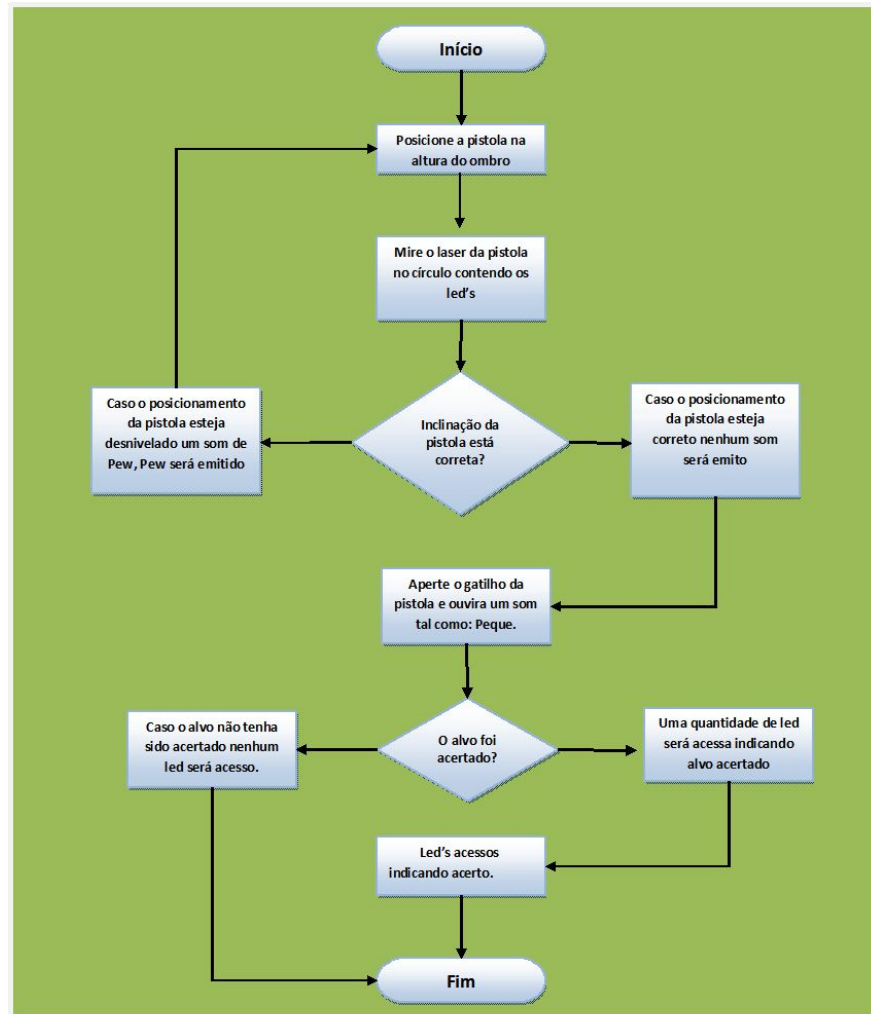


Figura 1 - Fluxograma de funcionamento do jogo

A figura 2 e a figura 3 referem-se ao código que indica qual alvo foi interceptado pela luz infravermelha da pistola. O código funciona de acordo com a variação da resistência do LDR.

```
mira
#include <msp430g2553.h>
#define P1IN BIT0 // LDR conectado ao P1.0
#define P1IN BIT1 // right sensor to P1.1

int main(void)
{
    WDCTL = WDTPW | WDTHOLD; // Parar tempo
    P1DIR |= BIT2|BIT3|BIT4|BIT5; // definindo P1.2, P1.3, P1.4 e P1.5 como saída, posição do LDR.
    P1DIR &= ~BIT0|~BIT1; // Definindo P1.0 P1.1 como entrada.
    P1OUT = 0;

    for(;;)
    {
        if((P1IN & 0x04) == 0x01)
        {
            P1OUT |= BIT2;
            P1OUT &= ~(BIT3|BIT4|BIT5);
        }
        if((P1IN & 0x04) == 0x02)
        {
            P1OUT |= BIT4;
            P1OUT &= ~(BIT2|BIT3|BIT5);
        }
        if((P1IN & 0x04) == 0x03)
        {
            P1OUT |= BIT3;
            P1OUT &= ~(BIT2|BIT4|BIT5);
        }
    }
}
```

Figura 2 - Código detector do LDR

```
mira
for(;;)
{
    if((P1IN & 0x04) == 0x01)
    {
        P1OUT |= BIT2;
        P1OUT &= ~(BIT3|BIT4|BIT5);
    }
    if((P1IN & 0x04) == 0x02)
    {
        P1OUT |= BIT4;
        P1OUT &= ~(BIT2|BIT3|BIT5);
    }
    if((P1IN & 0x04) == 0x03)
    {
        P1OUT |= BIT3;
        P1OUT &= ~(BIT2|BIT4|BIT5);
    }

    if((P1IN & 0x04) == 0x04)
    {
        P1OUT |= BIT5;
        P1OUT &= ~(BIT2|BIT3|BIT4);
    }
}
return 0;
}
```

Figura 3 - Código detector dos LDRs

A figura 3, refere-se ao código de detecção dos LDRs. O código funciona de modo que quando o luz infravermelha da pistola acerta um dos LDRs ocorre uma variação de resistência do LDR e o MSP SLAVE detecta essa variação e indica ao MSP MASTER que houve e a onde ocorreu essa variação.

```
Detecta_laser_projeto

// Sensor LDR

int recvPin = 0;
int wait = 1000;
int val = 0;    // Variavel que recebe freg.

void setup()
{
    // Iniciacao da interface serial
    Serial.begin(9600);
}

void loop()
{
    // leitura do analogic Pin
    val = analogRead(recvPin);

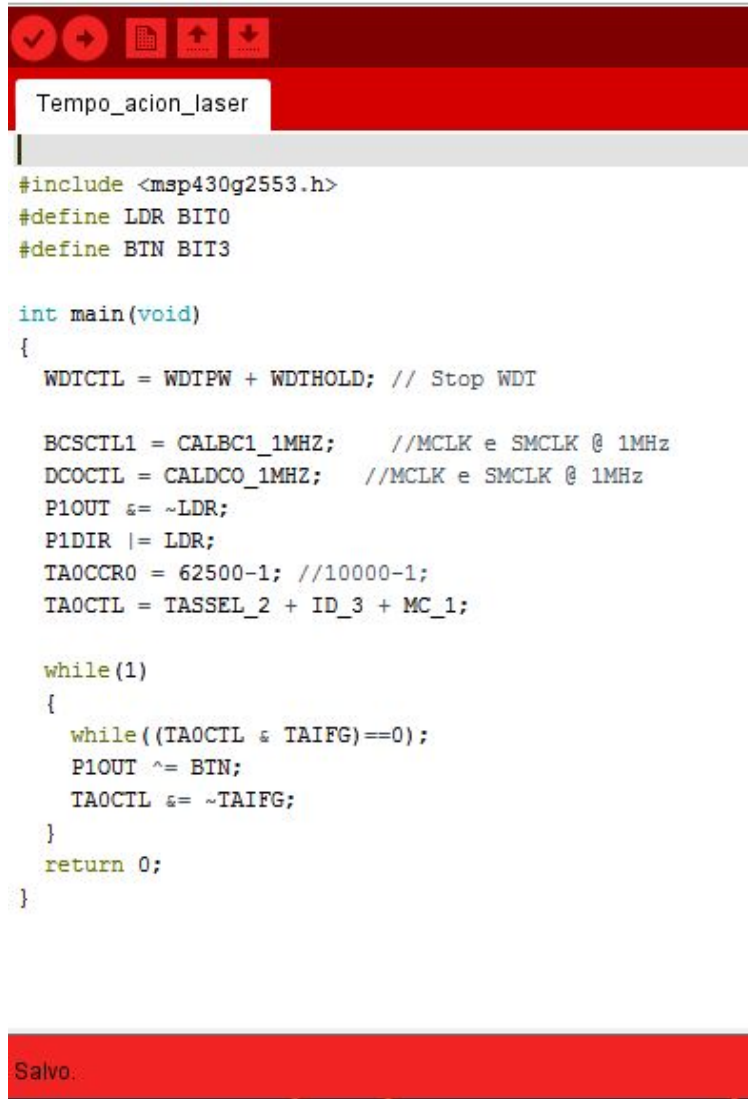
    // Saida da deteccao do valor

    Serial.print("DETECT : ");
    Serial.println(val);

    // Espera a proxima leitura.
    delay(wait);
}
```

Compilação terminada.

Figura 4. Detecta variação do LDR



```
Tempo_acion_laser

#include <msp430g2553.h>
#define LDR BIT0
#define BTN BIT3

int main(void)
{
    WDTCTL = WDTPW + WDTHOLD; // Stop WDT

    BCSCTL1 = CALBC1_1MHZ;    //MCLK e SMCLK @ 1MHz
    DCOCTL = CALDCO_1MHZ;    //MCLK e SMCLK @ 1MHz
    P1OUT &= ~LDR;
    P1DIR |= LDR;
    TA0CCR0 = 62500-1; //10000-1;
    TA0CTL = TASSEL_2 + ID_3 + MC_1;

    while(1)
    {
        while((TA0CTL & TAIFG)==0);
        P1OUT ^= BTN;
        TA0CTL &= ~TAIFG;
    }
    return 0;
}
```

Salvo.

Figura 5 - Acionamento do tempo de funcionamento do LASER.

- Cada arma possui seis tiros seguros. Não há recarga. Esses tiros podem ser disparados sem perigo.
- Cada jogador pode fazer seis acertos. Após fazer seis acertos, o jogador será eliminado.
- Cada tiro que um jogador disparar após o sexto aumentará a possibilidade de uma falha catastrófica da arma. No caso dessa falha catastrófica da arma, o jogador será eliminado.

RESULTADOS

Nesse patamar de desenvolvimento foi alcançado um protótipo de um alvo confeccionado em madeira de forma circular com 4 secções e uma pistola fabricada em material PVC, de acordo com a figura 6. Em cada seção de alvo consta um LDR para captura da luz infravermelha e na pistola está montado um circuito de um LASER do qual emite a luz infravermelho para o alcance do alvo.

Os experimentos ainda não demonstram os resultados almejados uma vez que ainda é preciso casar os códigos e finalização da maquete e do circuito da pistola para poder realizar os testes de alvo e coordenação motora, objeto deste projeto e protótipo.

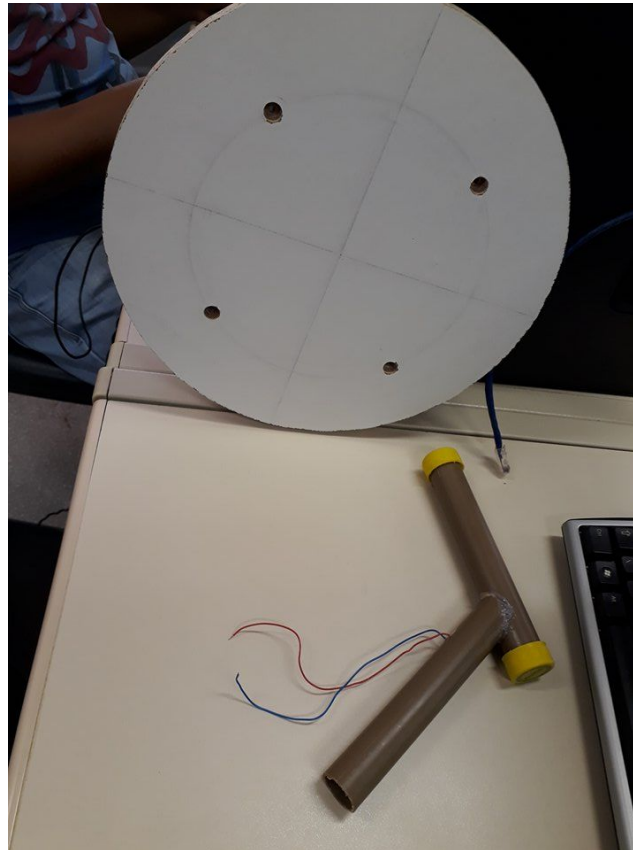


Figura 6 - Prévia das estruturas do alvo e arma laser.

CONCLUSÃO

Os desafios previstos para finalização e conclusão efetiva do projeto concentram-se na implementação serial do código entre os 2 MSPs, juntamente com que a jogabilidade sejam executadas sem erros. Os componentes da implementação de *hardware* estão em fase final de montagem, então espera-se tranquilidade referente à montagem dos circuitos e da interface com o jogador.

Por fim fica o entusiasmo de projetar este jogo e o otimismo que esse projeto seja um verdadeiro sucesso, mesmo tendo noção dos desafios e complicações que serão observadas nesta etapa final do projeto até a apresentação final.

REFERENCIAS BIBLIOGRÁFICAS

<http://www.feis.unesp.br/Home/departamentos/engenhariaeletrica/microcontroladores-pic-1.pdf>
http://www.ti.com/llds/ti/microcontrollers_16-bit_32-bit/msp/ultra-low_power/msp430frxx_fram/overview.page
<http://www.alldatasheet.com/view.jsp?Searchword=Msp430&gclid>
http://www.fade.up.pt/rpcd/_arquivo/artigos_soltos/vol.3_nr.1/1.5.investigacao.pdf