**Universidade Santa Cecília**

Gabriel Rodrigues

**Sensor de Luminosidade (LDR)**

Santos – SP

2021

**Sumário**

1 Introdução.........................................................................................................3

2 Circuito..............................................................................................................4

3 Desenvolvimento..............................................................................................5

4 Código...............................................................................................................6

4.1 Código............................................................................................................7

5 Sensor...............................................................................................................8

**Introdução**

O projeto do sensor de luminosidade tem como intuito informar através do acionamento de um LED que alguma pessoa, animal, objeto etc... Passou por determinado local. Podendo ser usado em diversos locais.

**Circuito**

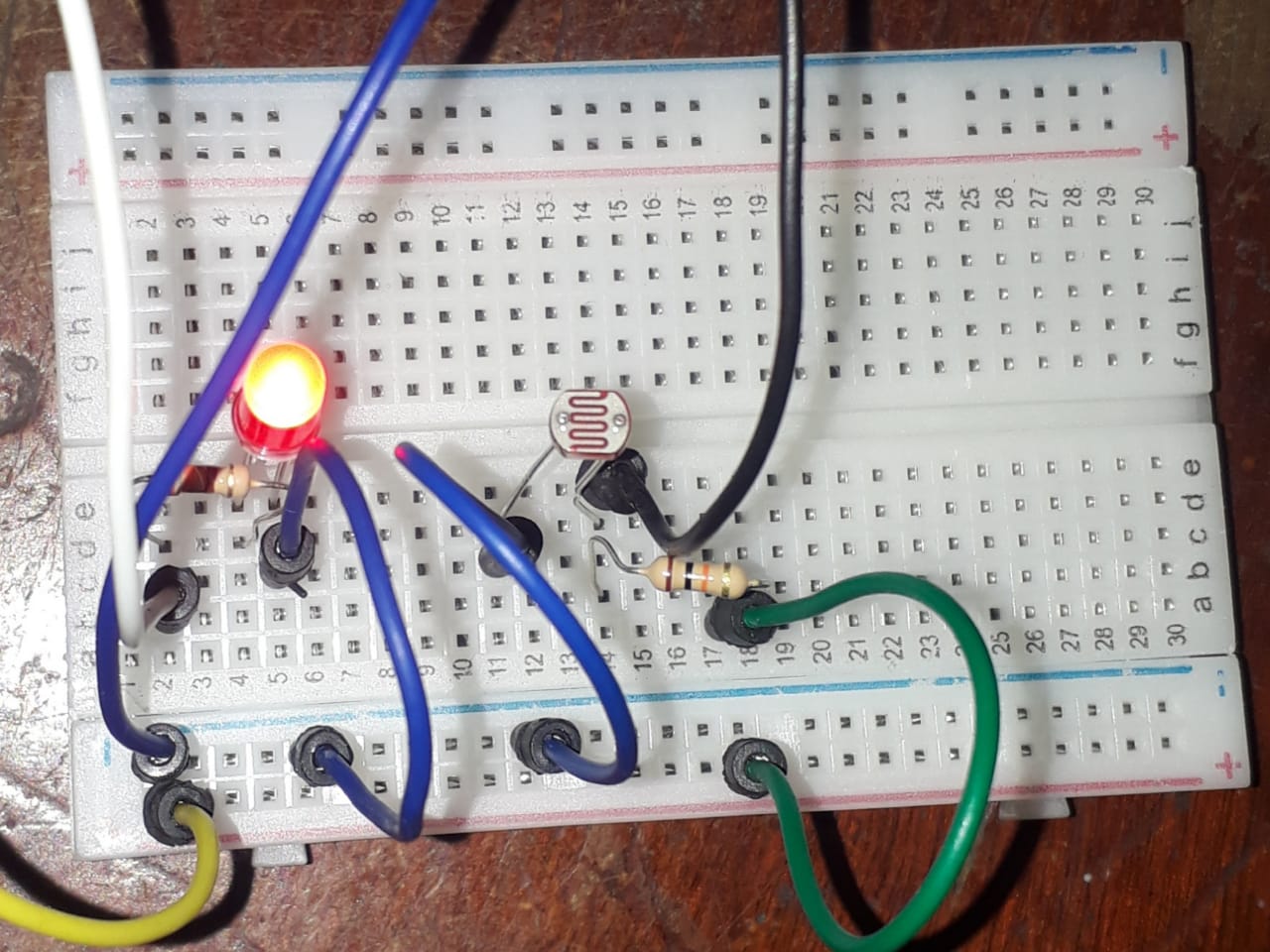
****

Figura 1 – Circuito do Projeto

Foi desenvolvido todo o circuito e simulação inicial do projeto no Protheus, após a simulação realizada montagem do circuito na Protoboard. O circuito desenvolvido no Protheus foi usado como base para a criação do circuito no SimulIDE.

**Componentes**

* Sensor de Luminosidade
* LED – Vermelhor
* Resistor 330 Ω
* Resistor 10k Ω
* Arduino UNO
* Protoboard

**Desenvolvimento**

Para o desenvolvimento do código foi usado à lógica do funcionamento do sensor e acionamento do LED.

A lógica é simples, enquanto estiver uma fonte de luz focada no sensor (laser) vai fazer com que o Led permaneça desligado. Essa lógica foi usado para desenvolver o código em Assembly no SimulIDE.

Para realizado o projeto foi necessário usar a conversão Analógica em Digital, ou seja, vai ser realizado a conversão para fazer com que o microcontrolador entenda esses valores que será 0 ou 1 (High ou Low). O sinal analógico é gerado pelo o sensor de luminosidade, esse sensor retornar um sinal analógico.

Os valores analógicos vão de 0 até 1024 que são as faixas, então temos 10 - bits de resolução. Como o microcontrolador trabalha com 8 bit será necessário usar 2 posições de memórias.

Foi definida qual a porta analógica que seria utilizada e configurado os registros.

* ADMUX - selecionar qual é a entrada
* ADCSRA – Controla o estado de acordo com o bit (funções de controle)
* ADC – ADCH e ADCL (HIGH e LOW)
* DIDR – Quais serão os registros que serão utilizados (No meu caso ADC0 – C0)

**Código**

Start – Foi realizada a configuração para que não seja iniciado de forma padrão, garantindo que o Stack Pointer (espaço de memória R/W temporário, funciona durante a execução do código) não inicie na posição da memória RAM.

Setup - Portas e registrador auxiliar (r16).

ConfigADC0 - Realização da configuração da porta analógica C0 (Conversão analógica).

Primeiro foi alterada a referência interna padrão da tensão que é 1,1V para 5V. (REFS1, REFS0, ADLAR, MUX3, MUX2, MUX1 e MUX0 igual a 0). Feito isso foi colocado no registro ADEMUX pela instrução STS (Guarda de formar direta no registro).

Segundo foi habilitado o ADC e iniciado ADPS2, ADPS1, ADPS0 como 1,fazendo isso eu tenho o fator de divisão igual a 128.

Comando SHIFT “<<”.

ADEN – Interrupção, ao terminar de converte o valor analógico é feito uma interrupção.

Após fazer essas configurações foi colocado no registro ADCSRA.

Loop – Iniciado a conversão analógica através do bit ADSC (Começar a conversão).

Ler\_adc0 – Confere se ADSC está em 1, enquanto estiver em 1 ele vai permanecer nessa Label. Quando o bit for igual a 0 ele pode sair dessa label, esse “escape” é feito através do SBRC.

LER\_H\_L\_ADC0 – É armazenado os valores do ADC (ADCH – HIGH e ADCL - LOW) no registro r24 e r25.

LED\_ON\_LED\_OFF\_ADC0\_value – Nessa label é a saída do código, de acordo com as configurações que foram feitas nessa Label vai fazer com que o valor que foi convertido faça o Led acender.

De acordo com o valor em ADCH se for > 128 vai fazer com o que o BRSH chame a Label LED\_ON, essa label faz com que o Led acenda. Isso seguindo a lógica inicial “Se a luz estiver focada no sensor o Led permanece apagado”. Então temos um if/else. O se não faz com que o led permaneça apagado, a linha em que está à instrução “BRSH LED\_ON” é pulada, e a instrução CBI vai limpar a PB1 e vai para instrução RJMP(desvio incondicional) que faz que volte para o Loop fazendo todo o processo novamente.

**Funcionamento do Sensor LDR**

O sensor de luminosidade LDR, conhecido como fotoresistor. Ele funciona de acordo com a exposição a luz, de acordo com a luminosidade em que ele é exposto faz com que a resistência dele diminua, a resistência é alterada conforme a luz.

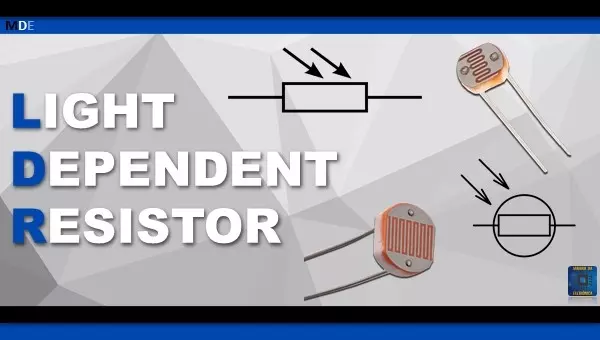


Figura 2 – Sensor de Luminosidade (LDR)

O projeto foi realizado nas seguintes ferramentas:

* SimulIDE
* Microchip Studio
* Arduino IDE
* Protheus

**FONTES CONSULTADAS**

**MICROCHIP**, ATmega328. Disponível em: <https://www.microchip.com >. Acesso em: 30 Maio. 2021.

**TECDICAS**, ATmega328 Registradores. Disponível em: < https://tecdicas.com/programando-o-atmega328p-com-registradores/>. Acesso em: 30 Maio. 2021.

**UNICAMP**, Introdução Assembly. Disponível em: <https://www.ic.unicamp.br/~ducatte/mc404/2009/docs/beginner\_pt.pdf>. Acesso em: 25 Mar. 2021.

**WR KITS**, Assembly. Disponível em: <https://www.youtube.com/channel/UCazAvTtoRlOrFDWDJDB2DKQ>. Acesso em: 30 Maio. 2021.