

Semana 4: Regulador de tensão, Display de 7 Segmentos, Opto Acopladores e LCD 16×2

Prof. Irineu Lopes Palhares Junior

FCT/UNESP,
irineu.palhares@unesp.br

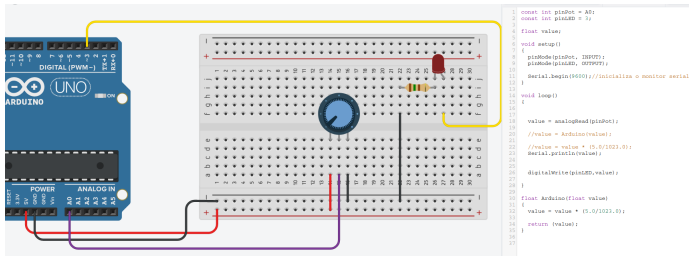


- Regulador de tensão
- Display de 7 segmentos
- Opto Acopladores
- Display LCD 16×2

Regularador de tensão



Montagem no tinkercad do regulador de tensão

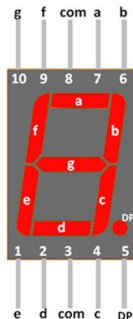


Programação do regulador de tensão

```
const int pinPot = A0; const int pinLED = 3;
float value;
void setup()  pinMode(pinPot, INPUT); pinMode(pinLED,
OUTPUT);
Serial.begin(9600); //inicializa o monitor serial
void loop()
value = analogRead(pinPot);
//value = Arduino(value);
//value = value * (5.0/1023.0); Serial.println(value);
digitalWrite(pinLED,value);
float Arduino(float value)  value = value * (5.0/1023.0);
return (value);
```

Display de 7 segmentos

O Display de 7 Segmentos Arduino é um componente que não possui mais do que 8 LEDs e cada LED está separado por segmento que pode ser nomeado como a, b, c, d, e, f, g, DP conforme mostrado na imagem abaixo.



Display de 7 segmentos de ânodo comum (AC) – A exibição comum do ânodo é comumente chamada de exibição AC. Neste tipo, o pino comum no display de 7 segmentos é conectado a todos os oito pinos positivos dos LEDs. Portanto, para fazer esse tipo de display funcionar, devemos conectar o pino no Vcc (+ 5V) e aterrar o pino do segmento necessário para ligá-lo.

Conectando ao arduino

Para começar, vamos conectar um dos pinos 3 ou 8 comuns ao pino 5V no Arduino (se você estiver usando um display de ânodo comum de 7 segmentos) ou ao pino GND no Arduino (se estiver usando um display de cátodo comum de 7 segmentos).

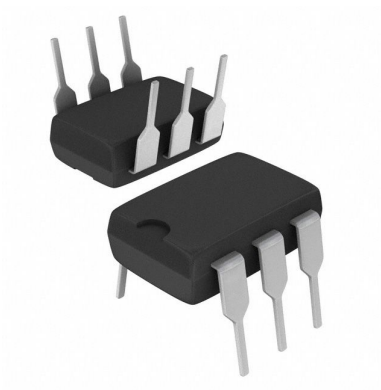
Os 4 pinos restantes na posição superior são conectados ao pino digital 2 ao pino digital 5. Os outros 4 pinos na posição inferior com ponto decimal são conectados ao pino digital 6 a 9.

Tarefa 1

Usando a atividade 1, sobre regulador de tensão, construa um circuito com o display de 7 segmentos de modo a mostrar o valor da tensão na tela.

Opto Acopladores 4N35

Os optoacopladores permitem que um circuito controle outro, mais ou menos como um transistor ou um relé também permitem. Contudo, diferente desses dois, o acoplador permite evitar completamente o contato elétrico entre o circuito controlador e o circuito controlado.



Display LCD 16×2

Módulos de display LCD (Liquid Crystal Display) de caracteres alfanuméricos são interfaces de comunicação visual muito úteis e atraentes. Eles se encontram em quase todos os aparelhos domésticos, eletroeletrônicos, automóveis, instrumentos de medição etc.



Ligação dos pinos do LCD ao Arduino

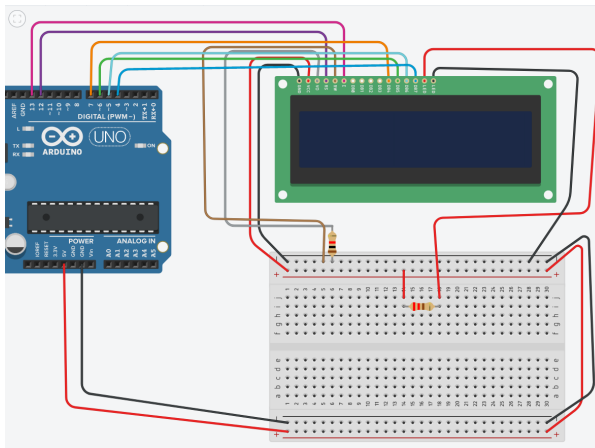
- O pino 1 (VSS) é ligado ao GND e o pino 2 (VDD) é ligado ao 5V. Utilize o GND e 5V do Arduino Uno.
- O pino 3 (V0) é responsável pelo ajuste do contraste do display, para isso coloque o resistor de 1K ligado ao GND, que será ajustado na medida certa para não saturar ou apagar as letras e números do display.
- O pino 4 (RS) é responsável pelo registrador de seleção do sinal, é ligado no pino 12 do Arduino Uno.
- O pino 5 (R/W) representa o sinal de escrita e leitura do display, é ligado ao GND.
- O pino 6 (E) habilita ou desabilita o sinal, é ligado no pino 13 do Arduino Uno.

Continuação da ligação dos pinos do LCD

- Os pinos de 7 ao 10 não são utilizados, pois o Data Bus será setado como 4 bits.
- O pino 11 (DB4) é ligado no pino 7 do Arduino Uno.
- O pino 12 (DB5) é ligado ao pino 6 do Arduino Uno.
- O pino 13 (DB6) é ligado ao pino 5 do Arduino Uno.
- O pino 14 (DB7) é ligado ao pino 4 do Arduino Uno.

Os pinos 15 (LED+) e 16 (LED-) são responsáveis por fornecer energia aos LEDs que ficam ao fundo do display para você enxergar o que está escrito nele. O pino 16 é ligado ao GND e o pino 15 ao vcc com um resistor de 220Ω .

Montagem do circuito e programação no tinkercad



```
Text
1 #include<LiquidCrystal.h>
2
3 LiquidCrystal lcd(12, 13, 7, 6, 5, 4); //Pinagem do LCD
4 int Number = 6, i;
5
6 void setup()
7 {
8   lcd.begin(16, 2); //inicia o lcd 16x2
9
10 }
11
12
13 void loop()
14 {
15   //lcd.println(Number);
16   //lcd.blink();
17
18   lcd.clear(); //limpa a tela
19   lcd.setCursor(3,0);
20   lcd.print("Atividade");
21
22   for (i=0; i<=8; i++)
23   {
24     lcd.scrollDisplayLeft(); //movimentar para esquerda
25     delay(500);
26   }
27
28   //lcd.clear();
29   //lcd.noCursor(); //começa em (0,0)
30   //lcd.print("Atividade");
31   //delay(1000);
32 }
33
34
35 Serial Monitor
```

Algumas funções da biblioteca LiquidCrystal.h

- Adicionar a biblioteca (cabeçalho): `LiquidCrystal.h`
- Setar os pinos: `LiquidCrystal lcd(RS, E, DB4, DB5, DB6, DB7);`
- Inicializar o LCD: `lcd.begin(16,2);`
- Limpar a tela: `lcd.clear();`
- Setar onde a escrita irá aparecer: `lcd.setCursor(Coluna, Linha);`
- Deslizar escrita para a esquerda: `lcd.scrollDisplayLeft();`

Tarefa 2

Refazer a tarefa 1 com o display LCD.