

⊗ O Potenciômetro é similar a um resistor, serve para limitar o fluxo de corrente elétrica, a principal diferença é que sua resistência pode ser ajustada enquanto o resistor tem uma resistência fixa. O potenciômetro possui três terminais e um eixo giratório, para a variação da resistência. Quando compramos um potenciômetro a resistência informada é sua resistência máxima (R), podendo variar de 0 até R .



Potenciômetro 1: terminais 1 e 2 ligados, sua resistência varia entre 0 até R . Girando para a esquerda a resistência diminui e para a direita ela aumenta.

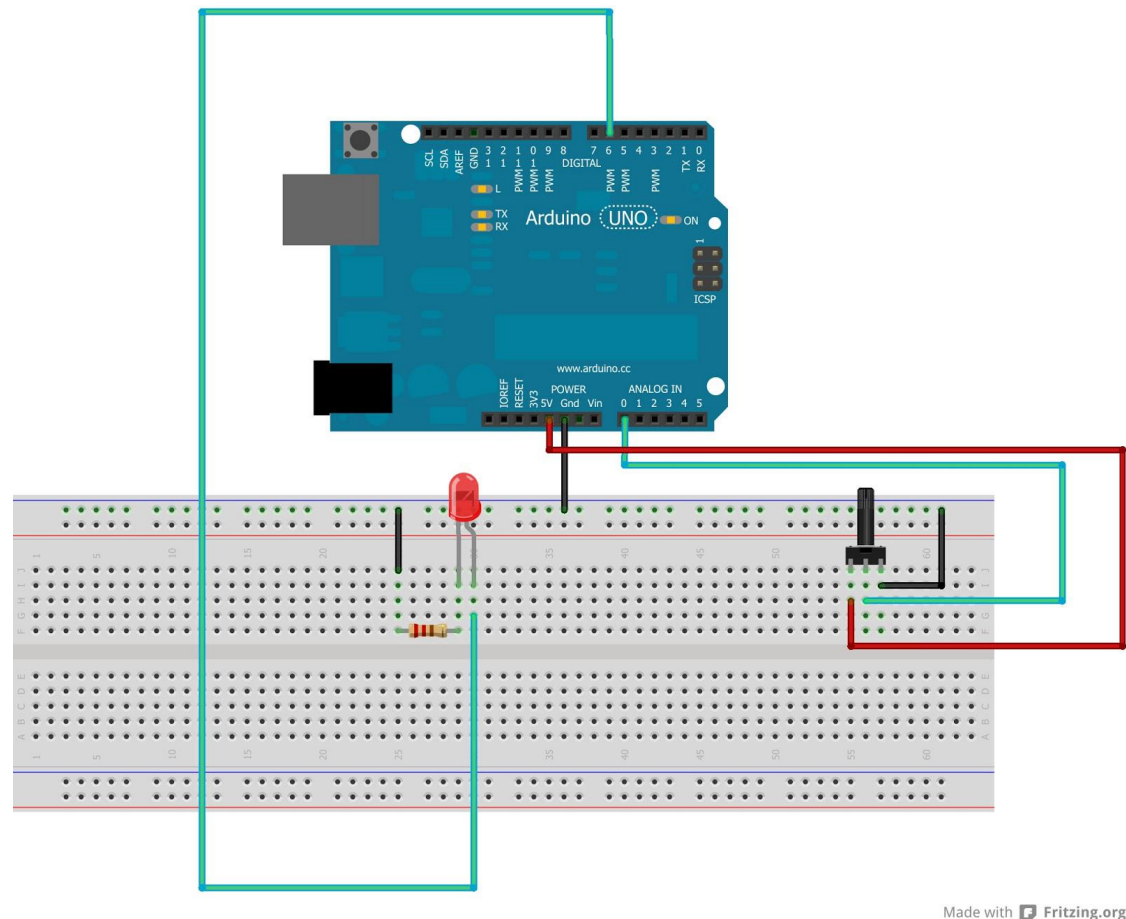
Potenciômetro 2: terminais 2 e 3 ligados, sua resistência varia entre 0 até R . Girando para a esquerda a resistência aumenta e para a direita ela diminui.

Potenciômetro 3: a resistência é fixa, em R . Mesmo girando sua resistência não varia.

As portas analógicas são outra forma que o microcontrolador tem para interagir com o mundo exterior e pela leitura e/ou geração de sinais contínuos, por exemplo, a temperatura do ambiente, o nível de água num tanque, a temperatura de um corpo, etc. As variáveis contínuas precisam ter uma 'adaptação' antes de entrar no microcontrolador, imagine que você vai medir a temperatura do ambiente e o range de temperatura definido pro você é de -10°C até $+50^{\circ}\text{C}$ e você tem internamente no microcontrolador um conversor analógico/digital de 8 bits (ou 256 valores diferentes, 2 elevado a 8 potência), desta forma, se você dividir o range total de temperatura pelo range do conversor ($60^{\circ}\text{C}/256$) você vai obter a resolução da captura da temperatura que seria de $0,23^{\circ}\text{C}$ ou por outro lado, você teria aproximadamente um incremento de 4 unidades por cada incremento de 1 grau Celsius. As portas analógicas de entrada dos microcontroladores geralmente estão ligadas aos canais do conversão A/D do microcontrolador e as portas analógicas de saída, que não são muito comuns, estão ligadas

a um conversor D/A.

⊗ A imagem abaixo mostra como foram feitas as ligações entre o arduino, o LED, o resistor e o potenciometro.



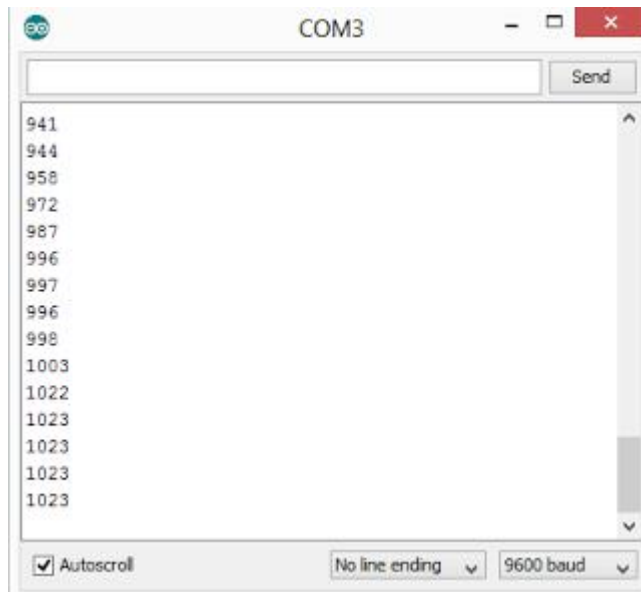
O potenciometro serve para controlar a intensidade do brilho do LED, o valor da intensidade será mostrado no monitor serial. Após ligados os componentes conforme a imagem acima, foi executado o código abaixo:

á á ô á
á ô
á
í

ô

ô

O valor da intensidade do LED era mostrado no monitor serial:



Algumas das funções utilizadas foram:

analogWrite: A saída será na porta indicada pela variável "led". Como vamos modificar o brilho do led é necessário utilizar uma porta Analógica em que podemos enviar valores de 0 a 255, ou seja, a saída pode variar de 0 V até 5 V, proporcionalmente ao valor passado. O valor recebido do potenciômetro varia de 0 a 1023, por isso é necessário que seja dividido por 4.

Serial.println: esta função irá imprimir no Serial Monitor o valor recebido pelo potenciômetro e fará o cursor pular uma linha, para que a próxima escrita seja feita em uma nova linha.

⊗ A corrente máxima por pino é de 40mA, mas a soma da corrente de todo o CI não pode ultrapassar 200mA. O resto da atividade já foi feita no último item do trabalho 1.

⦿ O circuito foi o mesmo do segundo item deste trabalho. A única mudança foram as ligações das ponteiros do osciloscópio no arduino. (jacaré - gcc, gancho - 5v). Conforme mudavamos o valor da intensidade através do potenciometro o formato de onda mudava.

⦿ Basicamente, um relé é um interruptor. Ou seja, tem um circuito de comando (ou primário), que quando alimentado por uma corrente, aciona um eletroímã que faz mudar de posição do outro par de contatos ligados a um outro circuito (circuito secundário ou comandado).

A grande vantagem do uso dos relés é que permite controlar circuitos de grande consumo, que exigem uma grande potência, através de sinais fracos, porque enquanto os platinados do relé podem suportar grandes correntes, o circuito de comando exige um sinal muito fraco.

Existe uma grande diversidade de relés disponíveis. A maioria são interruptores simples, e normalmente desligados. Isto quer dizer que quando o relé não recebe corrente no circuito de comando, não passa corrente no circuito comandado.

Existe ainda outro tipo de relé relativamente comum que comuta a corrente entre duas saídas diferentes.

⦿ Podemos controlar a velocidade de um motor variando a tensão aplicada a ele. Quanto menor a tensão, mais devagar o motor gira, e aumentando a tensão, ele passa a girar mais rápido. Para variar a tensão podemos usar um potenciômetro, funcionando como um divisor de tensão.

Porém, essa não é a técnica mais adequada para controlar a velocidade de motores de corrente contínua, pois é muito ineficiente - uma grande quantidade de energia é perdida na forma de calor dissipado pela resistência do potenciômetro. Uma forma melhor é usar a **técnica de PWM** para controlar a quantidade de energia que irá acionar o motor.

Para simplificar um pouco o circuito, iremos usar apenas o transistor para chavear o motor, sem o uso do relé. O transistor irá controlar a velocidade de rotação do motor por meio do controle de corrente que o atravessa, usando um sinal PWM fornecido pelo Arduino, por meio da função **analogWrite()**.

O ciclo ativo (duty cycle) do sinal PWM será, por sua vez, controlado por um potenciômetro, que poderemos ajustar para variar a velocidade do motor.

O resistor R1 está conectado ao pino 6 do Arduino.

O código é bem simples, no qual declaramos duas constantes, **MOTOR** e **POT** para ajustarmos os pinos de ligação do motor e do potenciômetro, respectivamente, e uma variável **valor** que irá receber o valor lido a partir do potenciômetro.

```
Const int MOTOR = 6;
const int POT = 0;
int valor = 0;
void setup() {
  pinMode(MOTOR, OUTPUT);
}
void loop() {
  valor = analogRead(POT);  valor =
  ap(valor, 0, 1023, 0, 255);
  analogWrite(MOTOR, valor);
}
```

Ajustamos o pino do motor para ser saída de sinal, e na função loop() realizamos a leitura do valor atual da tensão fornecida a partir do divisor de tensão formado com o potenciômetro. Esse valor é mapeado para o intervalo entre 0 e 255, que é o intervalo de valores válidos para o PWM (pois a entrada analógica fornece valor de tensão em resolução de 10 bits, portanto, entre 0 e 1023), e, finalmente, o valor lido e mapeado é enviado, via PWM, para o controle eletrônico do motor, via transistor no circuito.

Após montar e revisar o circuito, e criar o código e enviá-lo para o Arduino, basta ligar a fonte do motor, e girar o eixo do potenciômetro para ver o motor girar com maior ou menor velocidade.

Caso o motor não gire, ou não mude de velocidade ao rotacionar o potenciômetro, verifique tanto as ligações na matriz de contatos quanto o código-fonte do circuito.

A imagem a seguir mostra o circuito de controle de velocidade de motor DC com Arduino montado em matriz de contatos:

