

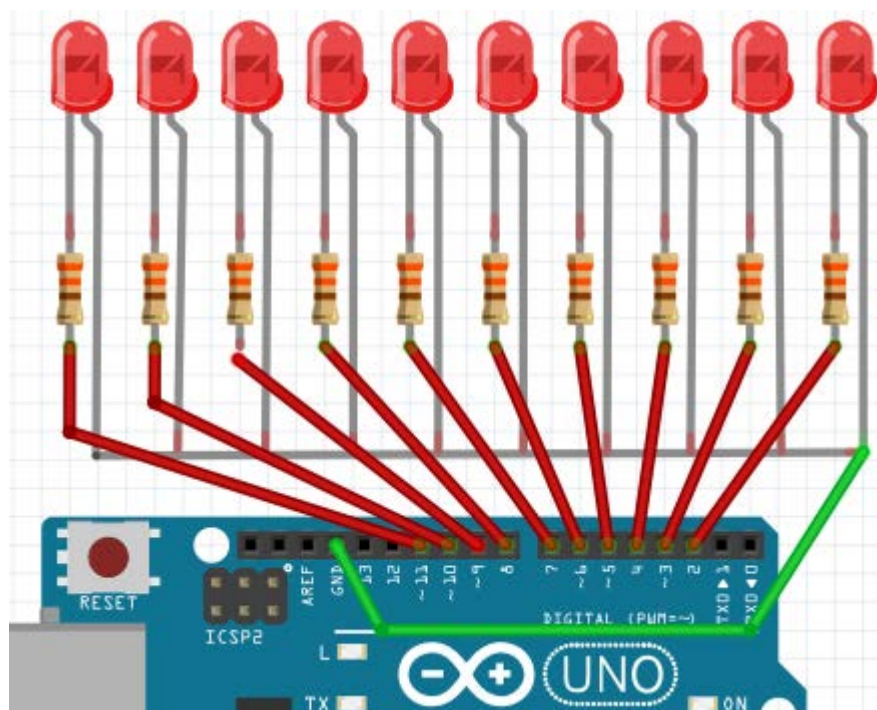
Loop (sequencial) com 10 leds

Nesta experiência construiremos um pisca com 10 leds que se desloca, a princípio, alternadamente da esquerda para a direita.

Os anodos dos leds (positivo) estão ligados nos pinos 2, 3, 4, 5, 6, 7, 8, 9, 10 e 11 da placa Arduino.

Os catodos estão ligados ao terra (GND) através de um resistor de 330 ohms.

Veja o diagrama das ligações na figura abaixo:



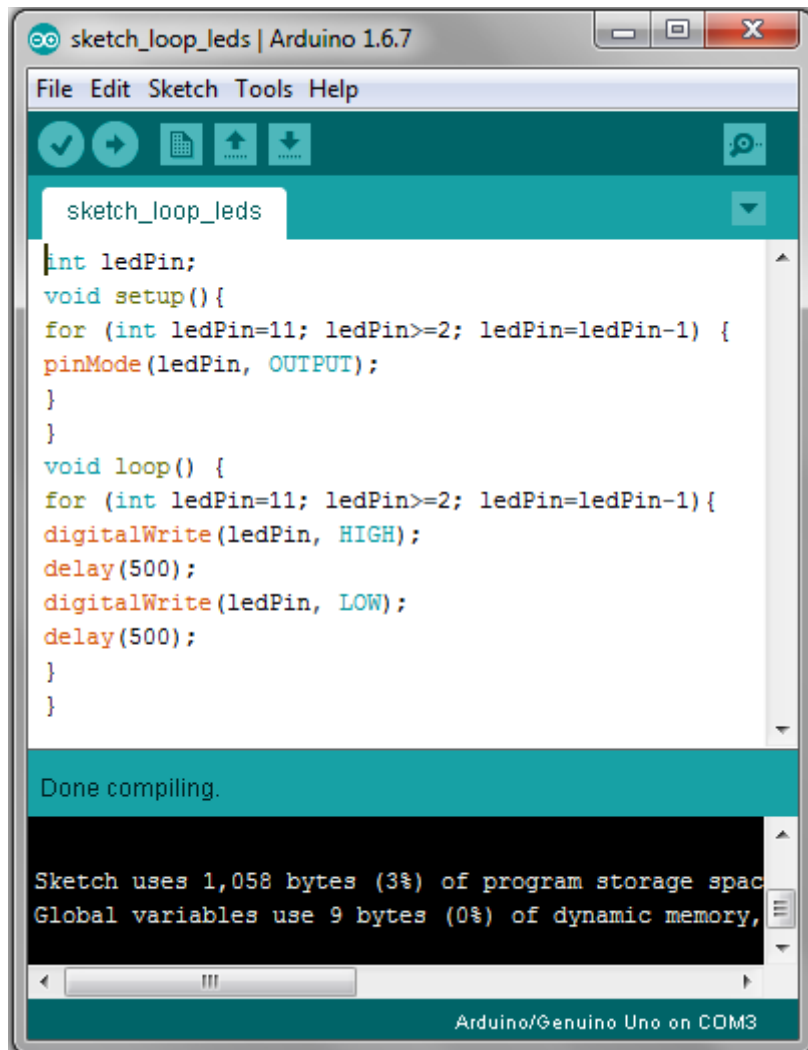
Programação:

A programação foi desenvolvida com laços de repetição, conforme os códigos abaixo:

```
int ledPin; //Identifica o pino (poderia ser dado outro nome, por exemplo, "pinoLed")
void setup(){
for (int ledPin=11; ledPin>=2; ledPin=ledPin-1) { //Quando o intervalo entre os pinos for entre 2 e 11, ou seja,
"maior ou igual a 2" e igual a 11, decrementa 1, uma vez que esses pinos estão fisicamente identificados
como 2, 3, 4, 5, 6, 7, 8, 9, 10 e 11.
OBS: o decremento de 1 pode ser escrito também como " ledPin--"
pinMode(ledPin, OUTPUT); } //Variável que define esses pinos como saída.
}
void loop() {
for (int ledPin=11; ledPin>=2; ledPin=ledPin-1){
```

```
digitalWrite(ledPin, HIGH);//manda corrente para o led - acende
delay(500); //permanece meio segundo aceso
digitalWrite(ledPin, LOW);//corta a corrente no led - apaga
delay(500); } //permanece meio segundo apagado (o processo se repete)
}
```

A figura a seguir mostra o *sketch* do Arduino:



```
sketch_loop_leds | Arduino 1.6.7
File Edit Sketch Tools Help
sketch_loop_leds
int ledPin;
void setup() {
  for (int ledPin=11; ledPin>=2; ledPin=ledPin-1) {
    pinMode(ledPin, OUTPUT);
  }
}
void loop() {
  for (int ledPin=11; ledPin>=2; ledPin=ledPin-1){
    digitalWrite(ledPin, HIGH);
    delay(500);
    digitalWrite(ledPin, LOW);
    delay(500);
  }
}
Done compiling.
Sketch uses 1,058 bytes (3%) of program storage space.
Global variables use 9 bytes (0%) of dynamic memory,
Arduino/Genuino Uno on COM3
```

➔ Após o upload, os leds acendem e apagam deslocando na sequência da “esquerda para a direita” e após um tempo de meio segundo o processo reinicia, graças aos códigos inseridos em “void loop()”.

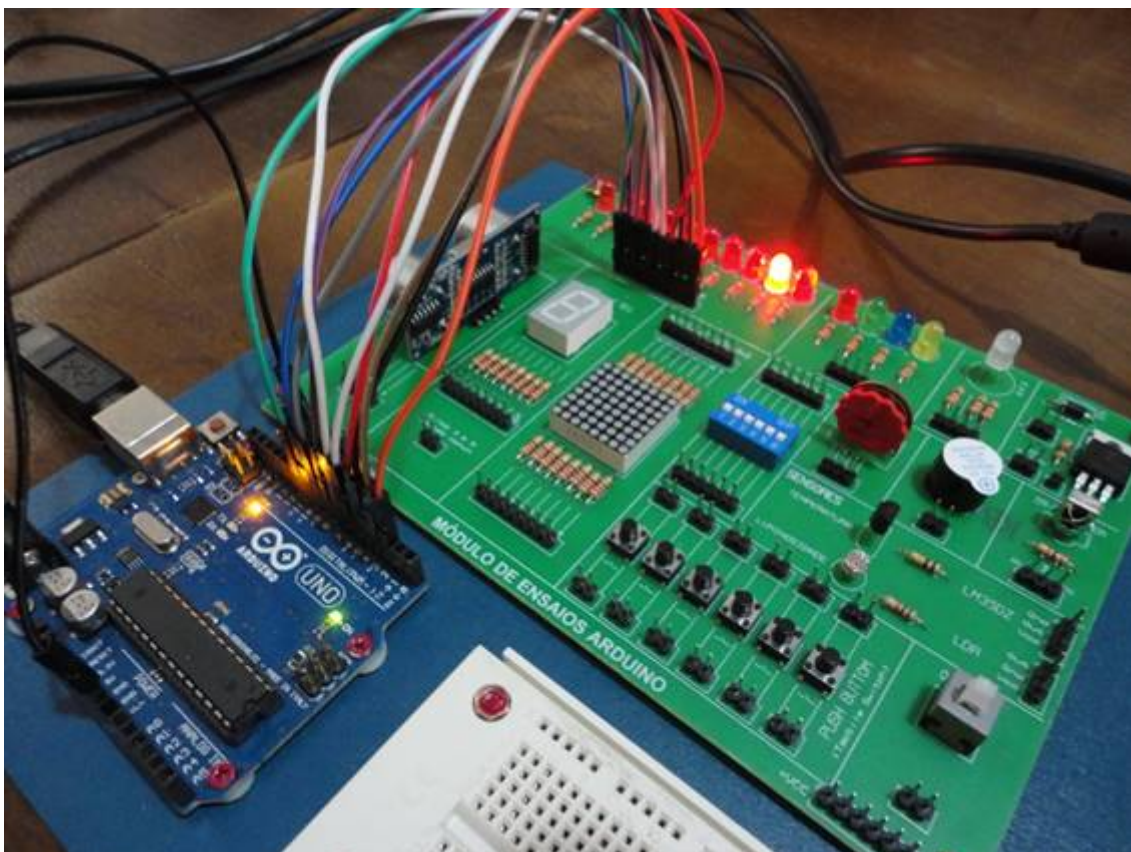
➔ Podemos incrementar a programação, fazendo com que depois de terminada a sequência de deslocamento da esquerda para a direita, imediatamente inicie o deslocamento no sentido contrário. Basicamente basta repetir os códigos em “void loop()”, modificando o atributo de ledPin de -1 (decremento) para +1 (incremento) e invertendo a ordem dos pinos.

```

int ledPin;
void setup(){
for (int ledPin=11; ledPin>=2; ledPin=ledPin-1) {
pinMode(ledPin, OUTPUT);
}
}
void loop() {
for (int ledPin=11; ledPin>=2; ledPin=ledPin-1){
digitalWrite(ledPin, HIGH);
delay(500);
digitalWrite(ledPin, LOW);
delay(500);
}
for (int ledPin=2; ledPin<=11; ledPin=ledPin+1){
digitalWrite(ledPin, HIGH);
delay(500);
digitalWrite(ledPin, LOW);
delay(500);
}
}
}

```

A figura a seguir mostra o circuito montado no Módulo de Ensaio.



Utilizando o mesmo circuito podemos sofisticar a programação com laços de repetição e vetores, que resulta no mesmo efeito.

Programação com vetores, deslocamento da esquerda para a direita:

```

byte ledPinos[] = {11,10,9,8,7,6,5,4,3,2}; //define a numeração dos pinos (cria um array)
void setup(){
  for(int pino=0; pino <=10;pino++) // define a quantidade de pinos {
    pinMode(ledPinos[pino],OUTPUT); }
}
void loop(){
  for(int pino=0; pino <=10;pino++) // Define que os 10 leds no processo sequencial. Se por exemplo,
  "pino=7" apenas os 3 leds da extrema direita serão incluídos no processo sequencial{
    digitalWrite(ledPinos[pino],HIGH);
    delay(200); // velocidade de deslocamento
    digitalWrite(ledPinos[pino],LOW);
    delay(200); }
}

```

Conclui-se que o efeito é o mesmo porém, com uma programação mais sofisticada.

```

sketch_loop_leds2 | Arduino 1.6.7
File Edit Sketch Tools Help
sketch_loop_leds2
byte ledPinos[] = {11,10,9,8,7,6,5,4,3,2};
void setup(){
  for(int pino=0; pino <=10;pino++) {
    pinMode(ledPinos[pino],OUTPUT);
  }
}
void loop()
  for(int pino=0; pino <=10;pino++){
    digitalWrite(ledPinos[pino],HIGH);
    delay(200);
    digitalWrite(ledPinos[pino],LOW);
    delay(200);
  }
}
Done Saving.
Sketch uses 1,118 bytes (3%) of program storage space.
Global variables use 19 bytes (0%) of dynamic memory,
14 Arduino/Genuino Uno on COM3

```