**F541**

**Prof. Davi Mendez Soares**

**Arduino- 5**

Há três tipos de memorias no Arduino.

1-Flash memory (program space), aonde o programa (sketch) é armazenado..

2-SRAM (static random access memory), aonde o programa manipula as variáveis quando roda..

3-EEPROM, aonde o programador armazena informação.

Flash memory and EEPROM memoryes são não voláteis. SRAM é volátil.

O ATmega328 chip do Arduino Uno tem:

Flash 32k bytes (of which .5k is used for the bootloader)

SRAM 2k bytes

EEPROM 1k byte

Os programas escritos, ocupam espaço na memoria do microcontrolador, que é pequena. Portanto, ao escrever programas, use o menor numero possível de variáveis necessárias para seu objetivo. Na tabela abaixo listamos alguns tipos de variáveis usadas nos programas.

*Tabela: Tipos de dados e memoria ocupada.*

**Data type RAM Number Range**

void keyword N/A N/A

boolean 1 byte 0 to 1 (True or False)

byte 1 byte 0 to 255

char 1 byte -128 to 127

unsigned char 1 byte 0 to 255

int 2 byte -32,768 to 32,767

unsigned int 2 byte 0 to 65,535

word 2 byte 0 to 65,535

long 4 byte -2,147,483,648 to 2,147,483,647

unsigned long 4 byte 0 to 4,294,967,295

float 4 byte -3.4028235E+38 to 3.4028235E+38

double 4 byte -3.4028235E+38 to 3.4028235E+38

string 1 byte + x Arrays of chars

array 1 byte + x Collection of variables

**Projeto 5**

Vamos escrever o programa: “LED Chase Effect” Neste, apenas um Led ascende em sequencia de ida e volta, com um tempo definido de ascendimento, em uma matriz de 10Leds.

// Project 5 - LED Chase Effect

byte **ledPin**[] = {4, 5, 6, 7, 8, 9, 10, 11, 12, 13}; // Cria array para os pinos LED

int **ledDelay=**65; // atraso entre mudanças

int **direction** = 1;

int **currentLED** = 0;

unsigned long **changeTime**;

void setup() {

for (int x=0; x<10; x++) { // faz pinos output

pinMode(ledPin[x], OUTPUT); }

changeTime = millis();

}

void loop() {

if ((millis() - **changeTime**) > **ledDelay**) { // tempo em ms para mudar

changeLED();

**changeTime** = millis();

}

}

void changeLED() {

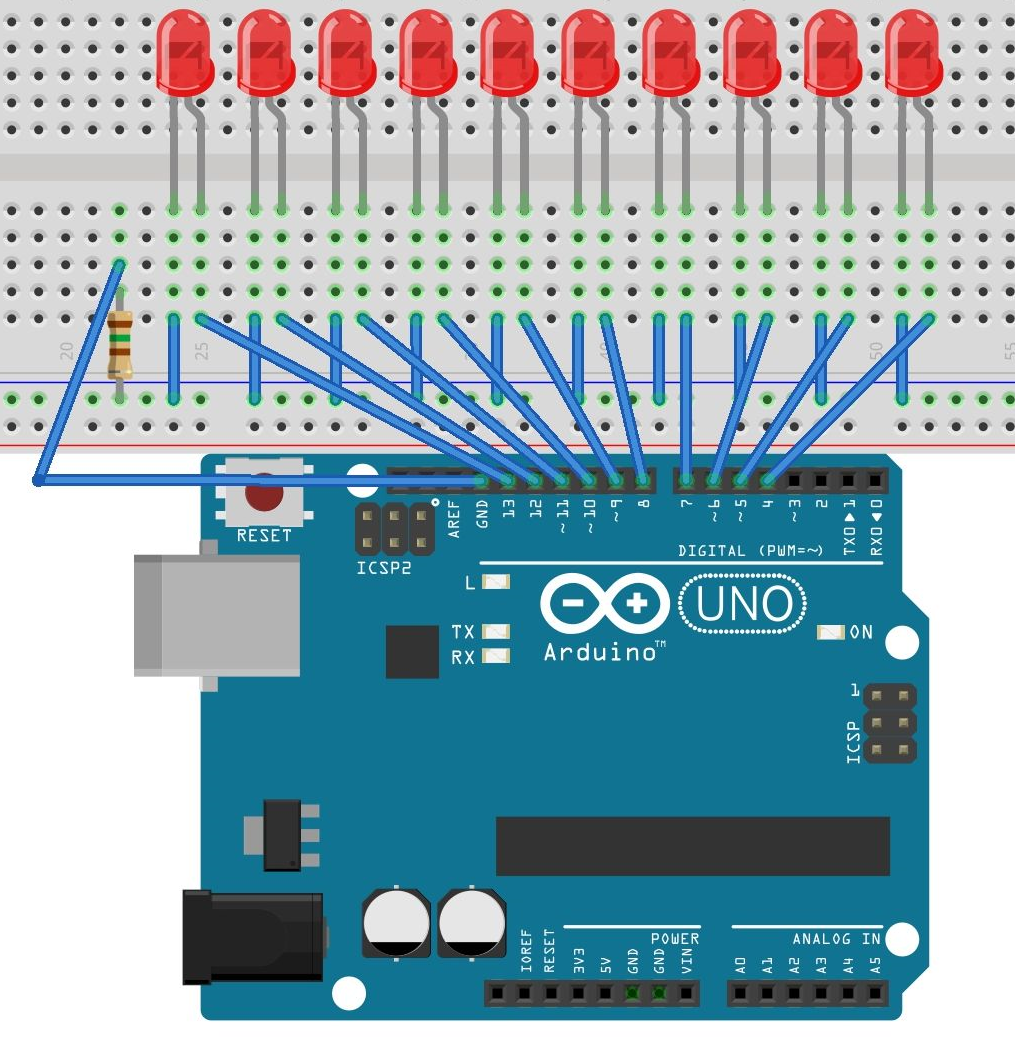
for (int x=0; x<10; x++) { // desliga todos os LED's

digitalWrite(ledPin[x], LOW);

}

digitalWrite(ledPin[currentLED], HIGH); // turn on the current LED

currentLED += direction; // increment by the direction value

Circuito em protoboard.

// change direction if we reach the end

if (currentLED == 9) {direction = -1;}

if (currentLED == 0) {direction = 1;}

}

No programa:

**byte ledPin**[] = {4, 5, 6, 7, 8, 9, 10, 11, 12, 13};

É a declaração de um ARRAY de 10 elementos. Se x=ledPin[4], x=8. Se x=ledPin[6], x=10 e assim por diante.

Se analisamos o programa, entenderemos todos os passos. Observamos que a velocidade de circulação do led aceso depende do valor de ledDelay que foi colocado em 65ms.