

## /\* Códigos para Arduino \*/

```
1 //ARDUINO - LENDO A PORTA SERIAL
2
3 const int LED = 3;
4 char nextChar = 0,
5 lendo = 0;
6 String valor;
7
8 void setup() {
9     Serial.begin(9600);
10    pinMode(LED,OUTPUT);
11 }
12
13 void loop() {
14     if (Serial.available() > 0) {
15         // lê o byte disponível na porta serial:
16         nextChar = Serial.read();
17
18         if(nextChar == 'B') {
19             lendo = 1; //lendo <- true
20             valor = "";
21         } else if(nextChar == 'E') {
22             lendo = 0; //lendo <- false
23             analogWrite(LED,valor.toInt());
24             Serial.println(String("Potencia do LED: ") + valor);
25         } else if(lendo && nextChar >= '0' && nextChar <= '9') {
26             valor += nextChar;
27         }
28     }
29 }
30
```

```
1 //Arduino - Lendo a porta Serial (outro modo)
2
3 const int LED = 3;
4 char nextChar = 0;
5
6 void setup() {
7     Serial.begin(9600);
8     pinMode(LED,OUTPUT);
9
10 }
11
12 void loop() {
13     if (Serial.available() > 0) {
14         // lê o byte disponível na porta serial:
15         nextChar = Serial.read();
16
17         if(nextChar == 'B') {
18             //Lê o próximo inteiro vindo da serial
19             int valor = Serial.parseInt();
20             //Atenção: em caso de erro o valor lido será 0
21             analogWrite(LED,valor);
22             Serial.println(String("Potencia do LED: ") + valor);
23         }
24     }
25 }
26
```

```

1 //EXEMPLO DE INTERRUPÇÃO - ARDUINO
2
3 int led = 13; //Porta do LED
4 int interruptPort = 2; //porta da interrupção
5 int interruptNumber = 0; //ID da interrupção
6
7 //Variáveis modificadas por interrupções devem ser volatile
8 volatile int state = LOW;
9
10 void setup() {
11     pinMode(led, OUTPUT);
12     pinMode(interruptPort, INPUT_PULLUP);
13     attachInterrupt(interruptNumber, toggle, CHANGE);
14 }
15
16 void loop() {
17     // Qualquer processamento mais longo...
18 }
19
20 void toggle() {
21     state = !state;
22     digitalWrite(led, state);
23 }
24

```

```

1 //LENDO O SENSOR DHT-11
2
3 #include "DHT.h"
4 #define DHTPIN A1 // pino que estamos conectado
5 #define DHTTYPE DHT11 // DHT 11
6
7 DHT dht(DHTPIN, DHTTYPE); //Instanciação do objeto do sensor
8
9 void setup() {
10     Serial.begin(9600);
11     dht.begin();
12 }
13
14 void loop() {
15     // A leitura da temperatura e umidade pode levar 250ms!
16     float h = dht.readHumidity(); //Valor da umidade
17     float t = dht.readTemperature(); //Valor da temperatura
18
19     if (isnan(t) || isnan(h)) {
20         Serial.println("Erro ao ler do DHT");
21     } else {
22         Serial.print("Umidade: ");
23         Serial.print(h); Serial.print(" %\t");
24         Serial.print("Temperatura: ");
25         Serial.print(t); Serial.println(" °C");
26     }
27 }
28

```

```

1 //Lendo o JSON da porta Serial e mandando a luminosidade
2
3 #include <ArduinoJson.h>
4 const int LED = 3;
5 const int LUZ = A1;
6 const int TAMANHO = 200;
7
8 void setup() {
9     Serial.begin(9600);
10    Serial.setTimeout(10); //1000ms é muito tempo
11    pinMode(LED,OUTPUT);
12 }
13
14 void loop() {
15     if (Serial.available() > 0) {
16         //Lê o texto disponível na porta serial:
17
18         char texto[TAMANHO];
19         Serial.readBytesUntil('\n', texto, TAMANHO);
20
21         //Grava o texto recebido como JSON
22         StaticJsonBuffer<TAMANHO> jsonBuffer;
23         JsonObject& json = jsonBuffer.parseObject(texto);
24
25         if(json.success() && json.containsKey("led")) {
26             analogWrite(LED, json["led"]);
27         }
28     }
29
30     StaticJsonBuffer<TAMANHO> jsonBuffer;
31     JsonObject& json = jsonBuffer.createObject();
32     json["luz"] = analogRead(LUZ);
33     json.printTo(Serial); Serial.println();
34     delay(1000);
35 }

```