

```
// Microcontrolador - Arduino Leonardo
```

```
/*
```

Instruções:

Esse programa realiza o controle e processamento de dados do sistema da Torneira Automática.

O sistema é composto por diversas integrações para garantir a eficiência.

Segue algumas instruções básicas de utilização:

1) Para acionar o equipamento ele deve estar conectado á tomada e sua chave seletora acionada.

2) O microcontrolador está sendo energizado por uma fonte 24v é essencial os pinos GND e VCC não sejam invertidos pois isso queimará o equipamento. Caso seja necessário a manutenção as etiquetas e MANUAL do equipamento devem ser seguidos!!

3) Para a compilação do código o aparelho DEVE estar Desconectado da tomada, caso contrario haverá sobrecarga de tensão que resultará em danos na memória Flash do processador.

Caso isso ocorra por falta de atenção, acesse ArduinoIDE > Arquivo > Exemplos > ArduinoISP. O exemplo irá abrir e então selecione a placa e portar do Arduino Leonardo e compile o código.

4) Certifique-se de que todos os cabos estão bem conectados nos devidos lugares.

```
* -----
```

```
* |   INFRAVERMELHO   |
```

```
* -----
```

```
* |   Vcc   |   5V   |
```

```
* |   PINO   |   3   |
```

```
* |   Gnd   |   GND   |
```

```
* -----
```

```
* -----
```

```
* |   FLUXO   |
```

```
* -----
```

```
* |   Vcc   |   5V   |
```

```
* |   PINO   |   2   |
```

```
* |   Gnd   |   GND   |
```

```
* -----
```

```
* -----
```

```
* |   RELÉ   |
```

```
* -----
```

```
* |   Vcc   |   5V   |
```

```
* |   PINO   |   4   |
```

```
* |   Gnd   |   GND   |
```

```
* -----
```

```
* Note: You do not obligatorily need to use the pins defined above
```

Após esses passos você estará apto para realizar a compilação do código de funcionamento.

Se tudo estiver funcionando corretamente você já poderá utilizar a torneira automática.

Segue a programação.

```
*/
//Variáveis Globais e Definições -----
-----

int Infravermelho = 3; // Pino sensor infravermelho
int Solenoide = 4;      // Pino conectado ao relé
#define Fluxo 2          // Pino sensor de fluxo
bool torneiraLigada = false;
double Valor;
double Lmin;
volatile int contagem;
int contagem2 = 0;
int memoria;

// Definição de setup -----
-----
---
void setup() {
    pinMode(13, OUTPUT);
    pinMode(Solenoide, OUTPUT);
    //inicializacao do monitor serial
    Serial.begin(9600);
    pinMode(Fluxo, INPUT);          // Configura o sensor de fluxo como
    entrada
    attachInterrupt(digitalPinToInterrupt(Fluxo), Flow, RISING); //
    Configura a interrupção no sensor de fluxo
    //mensagem de inicializacao de depuração
    Serial.println("Setup finalizado");
}

// Contagem e informações para acionamento -----
-----
void loop() {
    memoria = digitalRead(Infravermelho);
    if(memoria == 0){
        contagem2++;
        Lmin = contagem2 * Valor;
        Serial.print("Quantidade de Acionamentos: ");
        Serial.println(contagem2);
        Serial.print("Quantidade de Litros por Minuto Gastos: ");
        Serial.println(Lmin);}
    if (Serial.available() > 0) {
        char comando = Serial.read();
        if (comando == 'a' && !torneiraLigada) {
            ligarTorneira();
        }
    }
}
```

```

    if(digitalRead(Infravermelho) == 0){
        ligarTorneira();
    }
    else {
        TorneiraDesligada();
        delay(50);}
}
// Condição de acionamento -----
-----

void ligarTorneira() {
    digitalWrite(Solenoide, 1);
    Serial.println("A Torneira Está Acionada e Encerrará Seu Ciclo em 10
seg");
    Serial.println(" ");
    delay(50);
    Fluxo1();
    Serial.print("Este Valor é o Ultimo Registrado por Fluxo: ");
    Serial.println(Valor);          //Imprime a variável Valor no Serial
    digitalWrite(Solenoide, 0);
    delay(50);}
// Condição de encerramento -----
-----

void TorneiraDesligada(){
    digitalWrite(Solenoide, 0);
    delay(50);}
// Void sesor de fluxo -----
-----

----
void Fluxo1(){
    digitalWrite(Infravermelho, HIGH); // Ativa o relé para abrir a torneira
    torneiraLigada = true;
    contagem = 0; // Reseta a contagem para iniciar a contagem de vazão
    interrupts(); // Habilita as interrupções
    delay(10000); // Espera 5 segundos
    Valor = (contagem * 2.25);          //Conta os pulsos no último segundo e
multiplica por 2,25mL, que é a vazão de cada pulso
    Valor = Valor * 60;                //Converte segundos em minutos,
tornando a unidade de medida mL/min
    Valor = Valor / 10000;              //Converte mL em litros, tornando a
unidade de medida L/min
    noInterrupts(); // Desabilita as interrupções
}
// Contagem de pulsos sensor de fluxo -----
-----

void Flow() {
    if (torneiraLigada) {

```

```
        contagem++; // Incrementa a contagem quando o sensor de fluxo  
detecta um pulso  
    }  
}
```