



Neste tutorial apresentaremos como utilizar o módulo RTC juntamente com o Arduino

Instalando módulo RTC no Arduino

A maioria dos **microcontroladores** funcionam independente do tempo real. A maioria das coisas que desenvolvemos com **microcontroladores** não necessitam disso. Mas, o que fazer quando é preciso saber o tempo real? Por exemplo, como computar eventos ligados aos horários em que ocorreram, ou como criar um relógio. Para resolver isso o **Módulo de Real Time Clock (RTC)** foi desenvolvido.

Neste tutorial iremos mostrar como utilizar o módulo **RTC** juntamente com o **Arduino**. Para esta comunicação serão necessários os seguintes itens:

Arduino UNO R3

Jumper Premium M/F

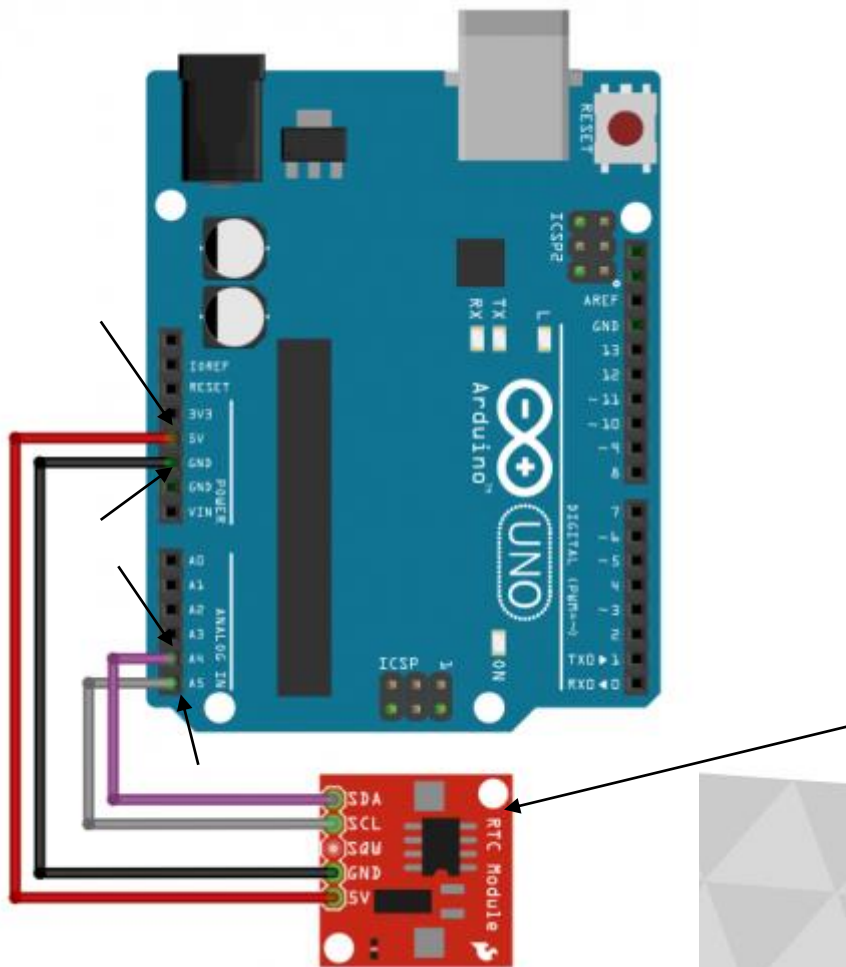
Módulo de Real Time Clock (RTC)

Entenda como o módulo funciona. O **DS1307** é um chip que funciona através do protocolo de comunicação **I2C** que retorna uma informação de 7 bytes. Sendo assim, ao receber a informação enviada pelo **módulo** é necessário transformá-la para **decimal**. Como a informação enviada está no padrão **BCD (Binary Code Decimal)**, vamos entender um pouco melhor como é este padrão.

O padrão **BCD** é uma codificação de cada dígito de um número em **4** dígitos binários. Vamos usar como exemplo o número **157**:

Em **BCD**: 157 = 0001 0101 0111

Em **Binário**: 157 = 10011101



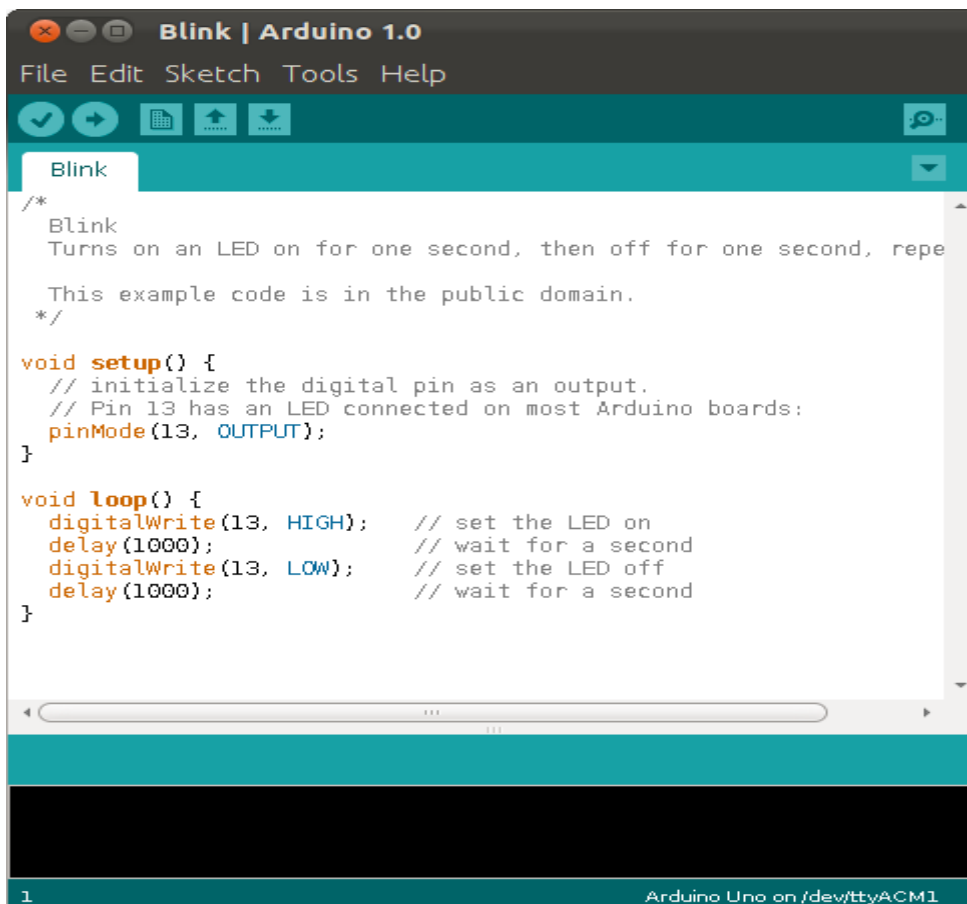
Agora que sabemos como receberemos a informação do módulo, precisamos de uma função (para facilitar durante a programação) que faça esta conversão de **BCD** para **decimal** quando recebermos informações do módulo e de **decimal** para **BCD** para quando enviarmos informações para o módulo. Estas funções já estão inclusas no código abaixo.

Assim como todos os relógios, este chip atrasa por conta de interferências externas como temperatura e campos magnéticos. Em seu manual, consta que o atraso pode chegar em 1 minuto por mês. Porém se este atraso for muito, existem módulos mais precisos como, por exemplo, o **DS3234** que atrasa cerca de 1 minuto por ano.

Bem, no código abaixo é apresentado, de uma forma bem simplificada, uma maneira de se programar o horário real e exibi-lo no monitor serial. Como todas as vezes que abre-se o monitor serial o Arduino reseta, se mantivermos a mesma programação no **Arduino** o módulo **RTC** receberá o horário definido na função **setDateTime()**. Como solucionar este problema? Simples, basta acrescentar **//** na linha da função **setDateTime()**; dentro do **void setup()** para que ela fique comentada e seja ignorada pelo programa e programar novamente o **Arduino**.

//Utilizar IDE Arduino 1.0 ou superior

Veja abaixo um exemplo do gráfico do software:



```
Blink | Arduino 1.0
File Edit Sketch Tools Help
Blink
/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeatedly.
 * This example code is in the public domain.
 */

void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards:
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH); // set the LED on
  delay(1000);           // wait for a second
  digitalWrite(13, LOW); // set the LED off
  delay(1000);           // wait for a second
}

1 Arduino Uno on /dev/ttyACM1
```

```
#include "Wire.h"
#define DS1307_ADDRESS 0x68
byte zero = 0x00;

void setup(){
  Wire.begin();
  Serial.begin(9600);
  setDateTime(); //Necessário configurar na função "setDateTime()"
}

void loop(){
  printDate();
  delay(1000);
}

void setDateTime(){
  byte segundo = 00; //0-59
  byte minuto = 00; //0-59
  byte hora = 19; //0-23
  byte diasemana = 2; //1-7
  byte dia = 3; //1-31
  byte mes = 12; //1-12
```



```
byte ano = 13; //0-99

Wire.beginTransmission(DS1307_ADDRESS);
Wire.write(zero);

Wire.write(decToBcd(segundo));
Wire.write(decToBcd(minuto));
Wire.write(decToBcd(hora));
Wire.write(decToBcd(diasemana));
Wire.write(decToBcd(dia));
Wire.write(decToBcd(mes));
Wire.write(decToBcd(ano));

Wire.write(zero);

Wire.endTransmission();

}

byte decToBcd(byte val){
// Conversão de decimal para binário
return ( val/10*16) + (val%10) ;
}

byte bcdToDec(byte val) {
// Conversão de binário para decimal
return ( val/16*10) + (val%16) ;
}

void printDate(){

Wire.beginTransmission(DS1307_ADDRESS);
Wire.write(zero);
Wire.endTransmission();

Wire.requestFrom(DS1307_ADDRESS, 7);

int segundo = bcdToDec(Wire.read());
int minuto = bcdToDec(Wire.read());
int hora = bcdToDec(Wire.read() & 0b111111); //Formato 24 horas
int diasemana = bcdToDec(Wire.read()); //0-6 -> Domingo - Sábado
int dia = bcdToDec(Wire.read());
int mes = bcdToDec(Wire.read());
int ano = bcdToDec(Wire.read());

//mostra a data e a hora. Ex.: 20/06/16 17:30:20

Serial.print(dia);
Serial.print("/");
Serial.print(mes);
Serial.print("/");
```



```
Serial.print(ano);  
Serial.print(" ");  
Serial.print(hora);  
Serial.print(":");  
Serial.print(minuto);  
Serial.print(":");  
Serial.println(segundo);  
  
}
```

Pronto, até o próximo tutorial galera.

