



BLOG

ARDUINO

BLYNK

AUTOMAÇÃO

EMBARCADOS

ELETRÔNICA

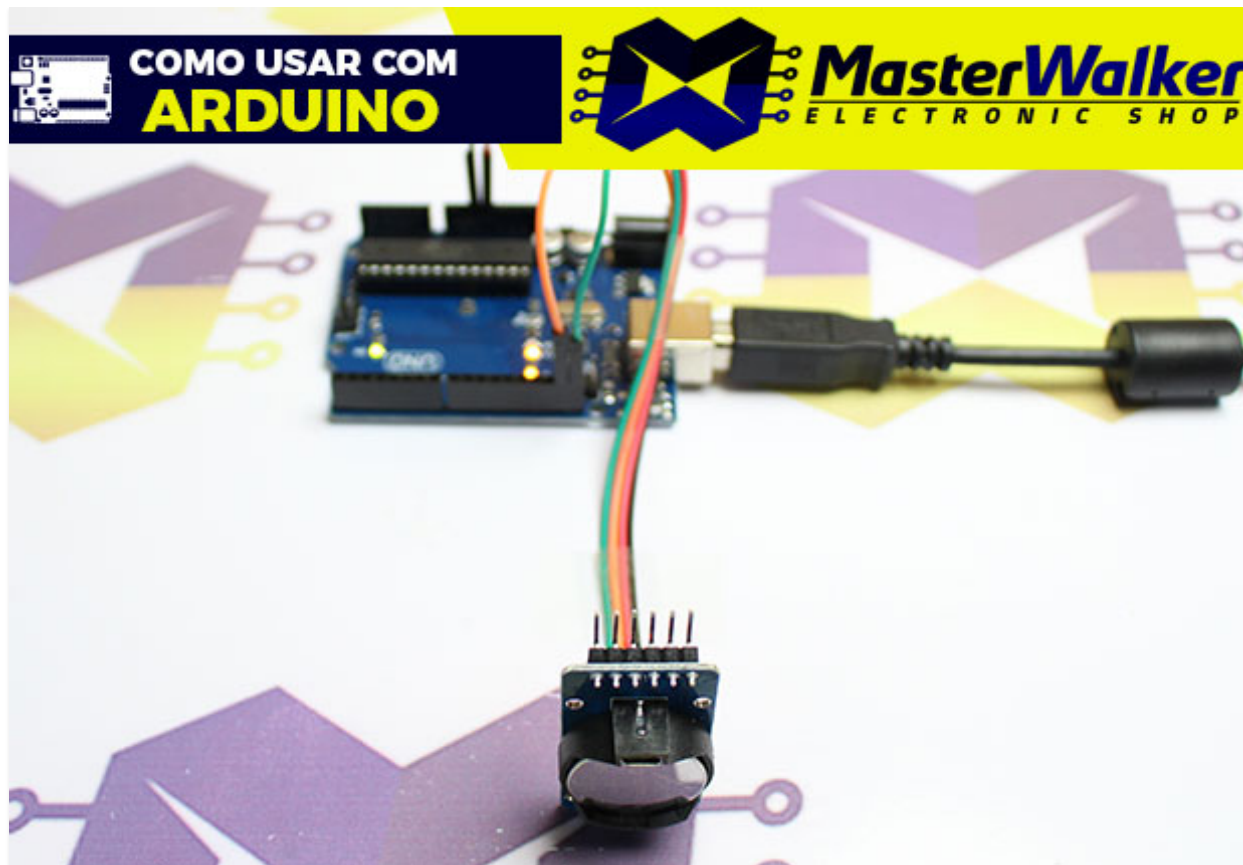
OUTROS

LOJA VIRTUAL



[🏠 Home](#) ▶ [ARDUINO](#) ▶ Como usar com Arduino – Módulo Real Time Clock RTC...





 Euler Oliveira  ARDUINO

Como usar com Arduino – Módulo Real Time Clock RTC DS3231

Pesquisar



Conheça o Wemos D1 Mini: pequena no tamanho e imensa nas funcionalidades



Descomplicando a pinagem do NodeMCU



Você sabe classificar os sensores de temperatura? Isto é importante para você minimizar os custos do seu projeto!

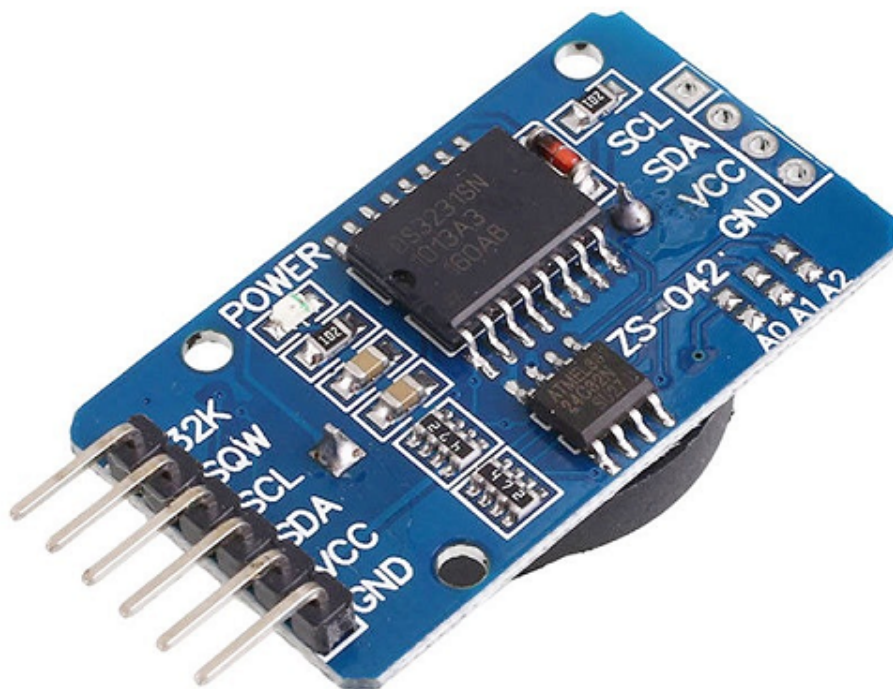


Sua casa estará incompleta se você não fizer este despertador com Arduino! Vamos montar? É simples e rápido!



Conhecendo o Módulo RXC6 Receptor Wireless Superheterodino RF 433MHz





– Descrição:

O **Módulo Real Time Clock RTC DS3231** é um relógio em tempo real (RTC) de alta precisão e baixo consumo de energia. O módulo possui um sensor de temperatura embutido e possui também um cristal oscilador para ampliar sua

Assine nossa NewsLetter! É rapidinho!



Clique aqui e inscreva-se

Assine nossa NewsLetter! É rapidinho!

Assinar

Nosso Site é Seguro



exatidão. O mesmo é capaz de fornecer informações de hora, minutos, segundos, dia, data, mês e ano.

Para o funcionamento do **Módulo Real Time Clock RTC DS3231** é necessário além da alimentação da placa, uma bateria CR2032 para poder permitir que os dados fiquem armazenados mesmo com a falta de energia. Além disso, o módulo é capaz de detectar falhas de energia e consequentemente acionando a bateria para evitar que os dados sejam perdidos. O módulo também é capaz de ajustar automaticamente o final dos meses que possuem menos de 31 dias, faz a correção de ano bissexto e trabalha tanto no formato de 12 horas quanto 24 horas.

– Especificações e características:

- Controlador: DS3231
- Tensão de operação: 3,3V – 5VDC
- Interface: I2C
- Computa horas / minutos / segundos / dias da semana / mês / meses / anos (de 2000 a 2099)
- Consome menos de 500nA no modo bateria com oscilador em funcionamento
- Memória: AT24C32 (capacidade de 32Kb que podem ser usados também como RAM estendida ao microcontrolador)
- Possui circuito de detecção de falha de energia
- Interface: I2C
- Bateria compatível: CR2032
- Sensor de temperatura: ~ 3° celsius de exatidão
- Temperatura de operação: 0 a 40° celsius

– Aplicações:



Projetos com Arduino ou outras plataformas microcontroladas em que seja necessário obter informações das horas, dia e data.

– Proposta da prática:

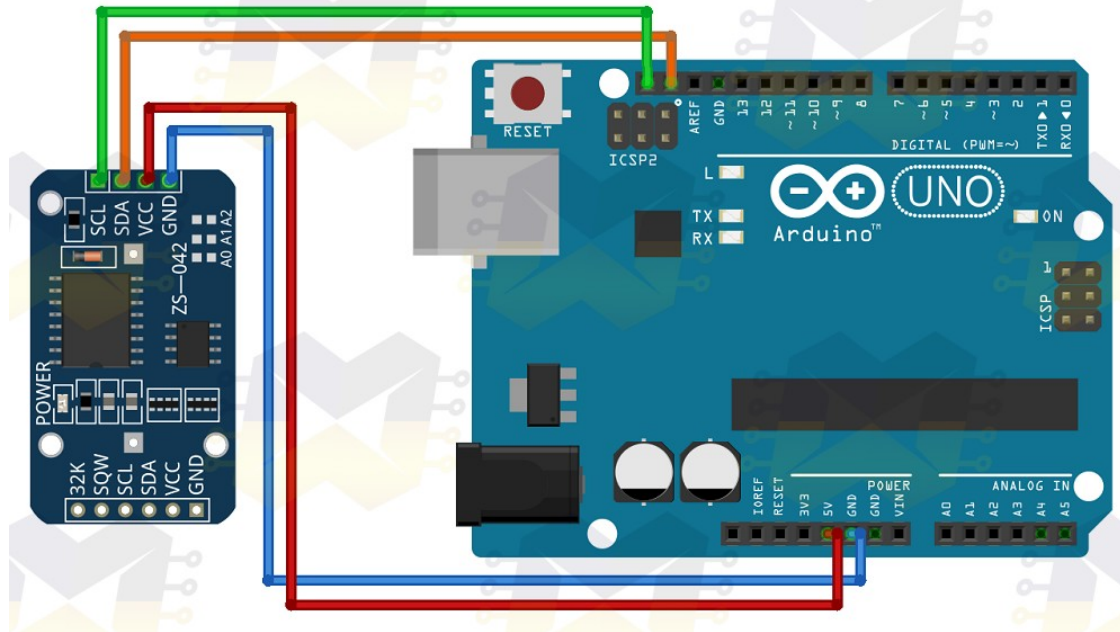
Utilizar o Módulo Real Time Clock RTC DS3231 em conjunto com o Arduino e obter informações de horas, dia e data atual. Estas informações serão exibidas no monitor serial do ambiente de programação do Arduino.

– Lista dos itens necessários:

- 01 – [Arduino com Cabo USB](#)
- 01 – [Módulo Real Time Clock RTC DS3231](#)
- 04 – [Cabos Jumper macho-fêmea](#)

– Esquema de ligação da prática





– Tutorial de instalação e configuração do ambiente de programação do Arduino:

[Arduino – Instalação e Configuração da IDE no Windows](#)

– Biblioteca:

[Download RTCLib](#)

– Importando bibliotecas para o ambiente de programação do Arduino:

[Arduino – Importando bibliotecas para a IDE](#)

– Código:



OBS: se o RTC estiver sendo ligado pela primeira vez, ficou sem energia, esgotou a bateria ou você precisa de ajustar as informações do mesmo, remova o comentário da linha 18 ou 19 para que as informações sejam atualizadas. A linha de código de número 18 captura as informações no momento em que o sketch for compilado e define no RTC. Na linha de código de número 19 você pode inserir as informações manualmente. Lembre-se de usar somente uma das linhas e deixar uma delas comentada.

```
1 #include <Wire.h> //INCLUSÃO DA BIBLIOTECA
2 #include "RTClib.h" //INCLUSÃO DA BIBLIOTECA
3
4 RTC_DS3231 rtc; //OBJETO DO TIPO RTC_DS3231
5
6 //DECLARAÇÃO DOS DIAS DA SEMANA
7 char daysOfTheWeek[7][12] = {"Domingo", "Segunda", "Terça", "Quarta", "Quint
8
9 void setup(){
10   Serial.begin(9600); //INICIALIZA A SERIAL
11   if(! rtc.begin()) { // SE O RTC NÃO FOR INICIALIZADO, FAZ
12     Serial.println("DS3231 não encontrado"); //IMPRIME O TEXTO NO MONITOR SE
13     while(1); //SEMPRE ENTRE NO LOOP
14   }
15   if(rtc.lostPower()){ //SE RTC FOI LIGADO PELA PRIMEIRA VEZ / FICOU SEM ENE
16     Serial.println("DS3231 OK!"); //IMPRIME O TEXTO NO MONITOR SERIAL
17     //REMOVA O COMENTÁRIO DE UMA DAS LINHAS ABAIXO PARA INSERIR AS INFORMAÇÕ
18     //rtc.adjust(DateTime(F(__DATE__), F(__TIME__))); //CAPTURA A DATA E HOR
19     //rtc.adjust(DateTime(2018, 9, 29, 15, 00, 45)); //(ANO), (MÊS), (DIA),
20   }
21   delay(100); //INTERVALO DE 100 MILISSEGUNDOS
22 }
23
24 void loop () {
25   DateTime now = rtc.now(); //CHAMADA DE FUNÇÃO
26   Serial.print("Data: "); //IMPRIME O TEXTO NO MONITOR SERIAL
27   Serial.print(now.day(), DEC); //IMPRIME NO MONITOR SERIAL O DIA
28   Serial.print('/'); //IMPRIME O CARACTERE NO MONITOR SERIAL
29   Serial.print(now.month(), DEC); //IMPRIME NO MONITOR SERIAL O MÊS
30   Serial.print('/'); //IMPRIME O CARACTERE NO MONITOR SERIAL
31   Serial.print(now.year(), DEC); //IMPRIME NO MONITOR SERIAL O ANO
32   Serial.print(" / Dia: "); //IMPRIME O TEXTO NA SERIAL
33   Serial.print(daysOfTheWeek[now.dayOfTheWeek()]); //IMPRIME NO MONITOR SE
34   Serial.print(" / Horas: "); //IMPRIME O TEXTO NA SERIAL
35   Serial.print(now.hour(), DEC); //IMPRIME NO MONITOR SERIAL A HORA
```



```
36 Serial.print(':'); //IMPRIME O CARACTERE NO MONITOR SERIAL
37 Serial.print(now.minute(), DEC); //IMPRIME NO MONITOR SERIAL OS MINUTOS
38 Serial.print(':'); //IMPRIME O CARACTERE NO MONITOR SERIAL
39 Serial.print(now.second(), DEC); //IMPRIME NO MONITOR SERIAL OS SEGUNDOS
40 Serial.println(); //QUEBRA DE LINHA NA SERIAL
41 delay(1000); //INTERVALO DE 1 SEGUNDO
42 }
```

– Resultado final:

Gostou desta prática? Então deixa seu comentário, dúvida ou sugestão aí embaixo!

Loja online: <https://www.masterwalkershop.com.br>

Fan page no Facebook: <https://www.facebook.com/masterwalkershop>

Nos ajude a espalhar conhecimento clicando no botão de compartilhar (**f Like**) que está mais abaixo.



Obrigado e até a próxima!

Seu feedback é muito importante! Que tal dar uma nota para esta postagem?!
Faça sua avaliação aqui embaixo.

Postagem anterior: [Como usar com Arduino – Sensor Indutivo PNP de Proximidade LJ12A3-4-Z/BY](#)

Próxima postagem: [Como usar com Arduino – Módulo Adaptador I2C para Display LCD \(16X2 / 20X4\)](#)

f Like

Postagem feita por



Euler Oliveira

Formado em Engenharia da Computação, Técnico em Eletroeletrônica, Fundador e Diretor da MasterWalker Electronic Shop. Se preocupa com o aprendizado dos seus clientes e daqueles interessados em aprender sobre Arduino e plataformas embarcadas em geral.

Postagens relacionadas





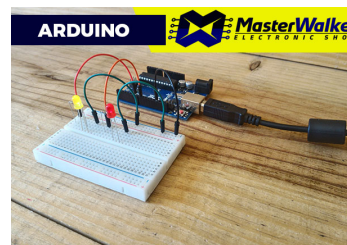
Como usar com Arduino – Sensor de Temperatura LM35

- Descrição: O Sensor de Temperatura LM35 possui alta precisão...



Como usar com Arduino – Controle Remoto Infravermelho

- Descrição: O Controle Remoto Infravermelho é um dispositivo utilizado para controlar...



Arduino – Utilizando a função millis

Nesta prática você vai aprender a utilizar a função millis...

Existem 13 comentários para esta postagem



LUCAS A AZEVEDO

23 de dezembro de 2018 às 16:11

Obrigado por compartilhar o seu conhecimento, me ajudou muito.

Responder ao comentário



Euler Oliveira Autor

24 de dezembro de 2018 às 05:59

Olá Lucas.

Por nada!



Responder ao comentário

**judenilson**

2 de junho de 2019 às 11:26

Poxa Euler Valeu pelo compartilhamento de informações!
Perdi um tempão atrás de alguma explicação sobre como fazer o relógio continuar contando mesmo após desligamento do arduino, pois o método ajuste é sempre chamado quando o arduino reinicia, reiniciando também o relógio para o primeiro horário de quando foi compilado.
Ninguém menciona esse método lostPower!
Infelizmente documentação para as bibliotecas usadas no Arduino são horríveis!
Mas graças a Deus e a Internet existem pessoas como você que estão dispostas a ajudar.
Só pra deixar registrado, estou usando o módulo num ESP32. 😊
Abraço.

Responder ao comentário

**Euler Oliveira** Autor

2 de junho de 2019 às 18:19

Olá Judenilson.

Agradeço pelo seu comentário e reconhecimento!

Vlw!!!

Responder ao comentário



**Pedro**

31 de agosto de 2019 às 00:48

Como eu poderia criar um alarme com esses comandos?

[Responder ao comentário](#)**Euler Oliveira** Autor

31 de agosto de 2019 às 18:07

Olá Pedro.

Agradeço pelo comentário!

Recomendo que pesquise por projetos similares a este que pretende montar, desta forma, você conseguirá as informações que precisa.

[Responder ao comentário](#)**FABIO APARECIDO BORTOLOTTI**

5 de junho de 2020 às 09:53

Mas devo ligalo em 5v ou 3.3v ouvi dizer que ele pode explodir a bateria se essa não for recarregável

[Responder ao comentário](#)

**Giovani**

6 de julho de 2020 às 19:41

Muito obrigado pela ajuda Euler. Não querendo abusar, como faço para ajustar o dia da semana? Forte abraço!

[Responder ao comentário](#)**Marco Antonio**

9 de julho de 2020 às 21:26

Olá, qual o procedimento para que o horário apareça como 20:00 e não 20:0 ? Parabéns pela aula, foi de grande ajuda

[Responder ao comentário](#)**Euler Oliveira** Autor

10 de julho de 2020 às 11:52

Olá Marco Antonio.

Agradeço pelo comentário!

Você deve verificar as funções da biblioteca utilizada no código e fazer a correção.

[Responder ao comentário](#)**Felipe**



28 de agosto de 2020 às 19:16

Sou novo em programação. Poderia me explicar porque essa matriz declarada no início, `daysOfTheWeek[7][12]`, tem 7 linhas e 12 colunas se são apenas 7 dias da semana no total?
E porque ela foi declarada como `char` se o conteúdo é `String` e não um carácter único?

[Responder ao comentário](#)



Fábio Da Silva Gregório

29 de outubro de 2020 às 09:57

Obgd pela dicas ,será Muito útil

[Responder ao comentário](#)



Luis Claudio Lorenzon

4 de fevereiro de 2021 às 20:04

Parabéns pela simplicidade e clareza nas explicações e nos comentários do código. Ajudou muito! Os códigos postados sobre Arduino + RTC por outros não mostram todos detalhes e o código não compila. Muito obrigado!

[Responder ao comentário](#)

Deixe um comentário

O seu endereço de e-mail não será publicado. Campos obrigatórios são marcados com *



Publicar comentário

Assine nossa NewsLetter! É rapidinho!

Assinar

Siga as nossas Mídias Sociais

Nosso Site é Seguro



© Copyright | Blog MasterWalker Shop

