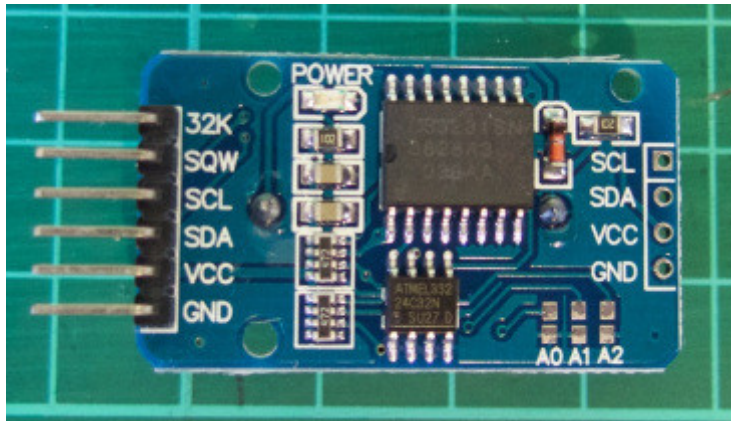




TUTORIÁL – UŽÍVÁNÍ HODIN REÁLNÉHO ČASU DS1307 A DS3231 S ARDUINEM

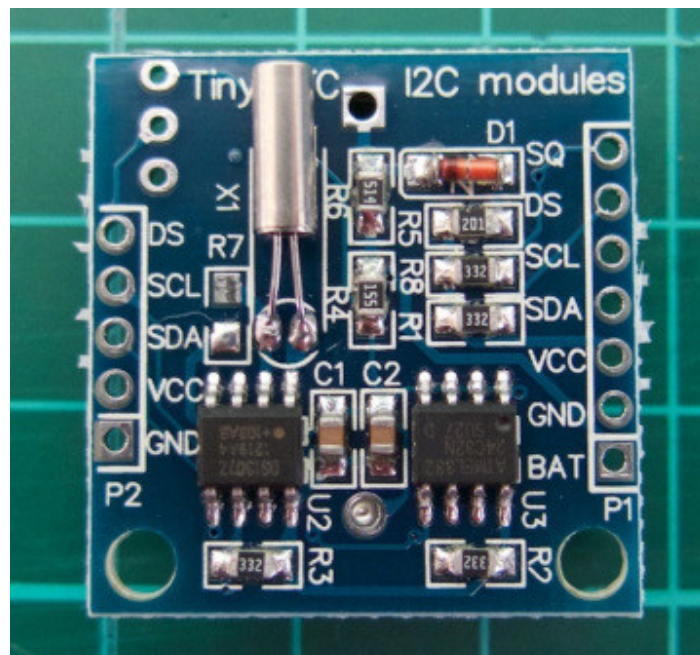
🕒 23.1.2015 👤 ANTONÍN HANDL 📁 ČLÁNKY, EXPERT, NÁVODY, POKROČILÝ

🔧 ARDUINO, DS1307, DS3231, RTC



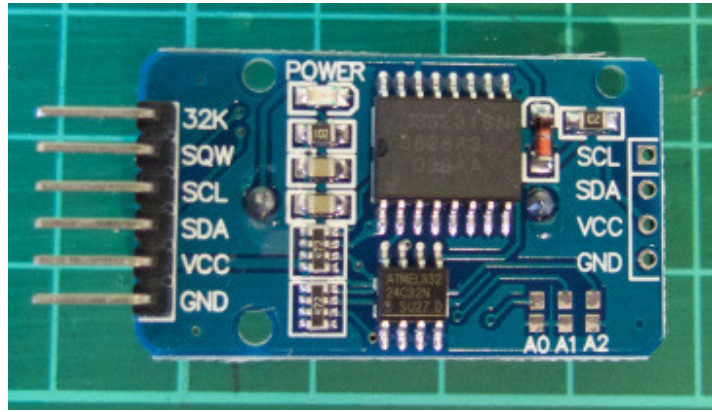
Server Tronixstuff před nedávnem zveřejnil návod, jak se dají s Arduinem ovládat moduly reálného času. Jeho první díl vám dnes přinášíme.

Podíváme se na dva moduly, Maxim DS1307 (Součástí například [tohoto kitu se senzory](#)):



Maxim DS1307

většina stránek na internetu, i my využíváme na stránkách cookies. V souladu s nařízením EU vás o tom tímto informujeme.



DS3231

Základní rozdíl mezi oběma integrovanými obvody hodin reálného času se týká přesnosti sledování času. Modul DS1307 funguje výborně, nicméně teplota okolí může ovlivnit frekvenci oscilátoru, který řídí vnitřní počítač DS1307.

To se může jevit jako problém, ve skutečnosti to ale znamená, že se hodiny posunou přibližně o pět minut v měsíci. Modul [DS3231](#) je výrazně přesnější, protože má vnitřní oscilátor, který neovlivňují vnější faktory a proto má odchylky nanejvýš několik minut ročně. Pokud užíváte modul DS 1307, nemusíte si zoufat, přes uvedený nedostatek jde o hodnotný kus, který vám bude dobře sloužit.

Oba moduly dostanete od výrobce Tronixlabs s vestavěnou baterií, ta je ale spíše z levnějších a déle než dva měsíce byste na ni neměli spoléhat. Pokud instalujete modul v rámci dlouhodobějšího projektu, je vhodnější zakoupit novou baterii typu CR2032 nebo CR2025 a tou v modulu nahradit baterii od výrobce.

Krom toho, že sledují čas a datum, mají tyto moduly také malou EEPROM, „budík“ (resp. má ho pouze DS3231) a schopnost generovat čtvercové vlny různých frekvencí. O tom všem pojedná náš druhý tutoriál.

Připojování modulu k Arduinu

Oba modely používají sběrnici I2C, která umožňuje velmi snadné připojení. Pokud si nejste v práci s ní a s Arduinem dostatečně jistí, podívejte se na příslušné návody (kapitoly [20](#) a [21](#)) nebo na sedmnáctou kapitolu mé knihy „[Arduino Workshop](#)“. (Poznámka redakce – v češtině si můžete o I2C sběrnici přečíst více [ZDE](#)).

Dále musíte identifikovat, které piny na Arduinu nebo kompatibilní desce sběrnice I2C užívá – jsou to piny značené SDA (data) a SCL (hodiny). Na Arduinu a kompatibilních deskách jde o piny A4 a A5 pro data a pro hodiny:

Připojení k Arduinu Uno

Pokud užíváte Arduino Mega, půjde o piny D20 a D21 pro data a pro hodiny:

Připojení k Arduinu Mega

Na přístrojích kompatibilních s Pro Mini půjde o piny A4 a A5 pro data a pro hodiny. Jsou paralelní k hlavním pinům, jak vidíte níže:

Připojení k ProMini

Modul DS1307

Jestliže používáte modul DS1307, potřebujete kabely připájet k desce nebo na pár pinů, abyste mohli používat jumper kabely. Poté připojíte piny SCL, SDA, Vcc a GND k Arduinu.

Modul DS3231

Připojování tohoto modulu je snadné, protože piny jsou na desce továrně instalovány. Stačí vést kabely opět z SCL a SDA do Arduina a znovu z Vcc modulu a GND pinů do 5V nebo 3,3 V a GND desky. Ty jsou nicméně duplikovány na druhé straně kvůli pájení vlastních kabelů.

Oba moduly mají potřebné pull-up rezistory, takže nemusíte dodávat vlastní. Jako v případě každého nástroje připojeného ke sběrnici I2C, snažte se používat co nejkratší SDA a SCL kabely.

Čtení a zapisování času v modulu hodin reálného času

Jakmile modul úspěšně připojíte, potřebujete si stáhnout následující sketch. Komentáře a funkce se sice týkají pouze DS3231, ale kód funguje také pro DS1307.

```
#include "Wire.h"
#define DS3231_I2C_ADDRESS 0x68
// Convert normal decimal numbers to binary coded decimal
byte decToBcd(byte val){
    return( (val/10*16) + (val%10) );
}
// Convert binary coded decimal to normal decimal numbers
byte bcdToDec(byte val){
    return( (val/16*10) + (val%16) );
}
```

```
void setup() {
    Wire.begin();
    Serial.begin(9600);
    // set the initial time here:
    // DS3231 seconds, minutes, hours, day, date, month, year
    // setDS3231time(30,42,21,4,26,11,14);
}

void setDS3231time(byte second, byte minute, byte hour, byte dayOfWeek,
byte dayOfMonth, byte month, byte year){
    // sets time and date data to DS3231
    Wire.beginTransmission(DS3231_I2C_ADDRESS);
    Wire.write(0); // set next input to start at the seconds register
    Wire.write(decToBcd(second)); // set seconds
    Wire.write(decToBcd(minute)); // set minutes
    Wire.write(decToBcd(hour)); // set hours
    Wire.write(decToBcd(dayOfWeek)); // set day of week (1=Sunday,
7=Saturday)
    Wire.write(decToBcd(dayOfMonth)); // set date (1 to 31)
    Wire.write(decToBcd(month)); // set month
    Wire.write(decToBcd(year)); // set year (0 to 99)
    Wire.endTransmission();
}

void readDS3231time(byte *second, byte *minute, byte *hour, byte
*dayOfWeek, byte *dayOfMonth, byte *month, byte *year){
    Wire.beginTransmission(DS3231_I2C_ADDRESS);
    Wire.write(0); // set DS3231 register pointer to 00h
    Wire.endTransmission();
    Wire.requestFrom(DS3231_I2C_ADDRESS, 7);
    // request seven bytes of data from DS3231 starting from register 00h
    *second = bcdToDec(Wire.read() & 0x7f);
    *minute = bcdToDec(Wire.read());
    *hour = bcdToDec(Wire.read() & 0x3f);
    *dayOfWeek = bcdToDec(Wire.read());
    *dayOfMonth = bcdToDec(Wire.read());
    *month = bcdToDec(Wire.read());
    *year = bcdToDec(Wire.read());
}

void displayTime(){
    byte second, minute, hour, dayOfWeek, dayOfMonth, month, year;
    // retrieve data from DS3231
    readDS3231time(&second, &minute, &hour, &dayOfWeek, &dayOfMonth, &month
, &year);
    // send it to the serial monitor
    Serial.print(hour, DEC);
    // convert the byte variable to a decimal number when displayed
    Serial.print(":");
    if (minute<10){
        Serial.print("0");
    }
    Serial.print(minute, DEC);
    Serial.print(":");
    if (second<10){
```

```
        Serial.print("0");
    }
    Serial.print(second, DEC);
    Serial.print(" ");
    Serial.print(dayOfMonth, DEC);
    Serial.print("/");
    Serial.print(month, DEC);
    Serial.print("/");
    Serial.print(year, DEC);
    Serial.print(" Day of week: ");
    switch (dayOfWeek) {
        case 1:
            Serial.println("Sunday");
            break;
        case 2:
            Serial.println("Monday");
            break;
        case 3:
            Serial.println("Tuesday");
            break;
        case 4:
            Serial.println("Wednesday");
            break;
        case 5:
            Serial.println("Thursday");
            break;
        case 6:
            Serial.println("Friday");
            break;
        case 7:
            Serial.println("Saturday");
            break;
    }
}

void loop() {
    displayTime(); // display the real-time clock data on the Serial
    Monitor,
    delay(1000); // every second
}
```

Není právě krátký, můžete si jej ale rozdělit na několik částí, se kterými se vám bude snadno pracovat.

První část se týká knihovny *Wire*, která se používá pro komunikaci přes sběrnici I2C, a definování adresy sběrnice pro RTC v podobě 0x68. Následují dvě funkce, které konvertují desítková čísla do BCD (binárního kódu) a naopak. Je to nutné, protože RTC ICs fungují v BCD, ne v desítkovém systému.

Funkce *setDS3231time()* slouží k nastavení hodin. Je to snadné, zkrátka zadejte hodnoty od roku po sekundu a RTC od zadaného času začne počítat. Pokud například chcete nastavit

středu 26. Listopadu 2014, 21:42 a 30 vteřin, použijte:

```
setDS3231time(30,42,21,4,26,11,14);
```

Čas se nastavuje ve čtyřadvacetihodinovém rámci, kdy čtvrtým parametrem je „den v týdnu“, který značí čísla 1 až 7 (od neděle po sobotu). Tyto parametry jsou pro údaje v bytech, pokud chcete nahrazovat vlastními proměnnými.

Jakmile zprovozníte funkci, je na místě k ní přidat `//` (zakomentovat ji) a znovu nahrát kód, aby neresetovala čas po každém spuštění programu.

Čtení času z RTC je rovněž jednoduché, celý proces snadno sledujete pomocí funkce `displayTime()`. Potřebujete definovat sedm bytových proměnných, abyste mohli ukládat data z RTC a ta potom vkládáte do funkce `readDS3231time()`.

Když máte například tyto proměnné:

```
byte second, minute, hour, dayOfWeek, dayOfMonth, month, year;
```

... obnovit se současnými daty z RTC je můžete pomocí:

```
readDS3232time(&second, &minute, &hour, &dayOfWeek, &dayOfMonth, &month, &year);
```

Pak můžete užívat proměnné, jak se vám bude hodit, např. posílat čas a datum do sériového monitoru jako sketch, který jsme použili jako příklad nebo konvertovat data do přijatelné podoby z celé řady výstupních nástrojů.

Pro ověření, že vše funguje, jak má, zadejte správný časový údaj do ukázkového sketchu, nahrajte jej, zakomentujte funkci `setDS3231time()` a znovu ji nahrajte. Následně otevřete sériový monitor a měli byste vidět běžící displej s aktuálním časem a datem, například:

Sketch: data a časy

Nyní máte softwarové nástroje pro nastavení dat a jejich opětovné získávání z modulu RTC a my doufáme, že nyní chápete, jak tyto nepříliš nákladné moduly používat.

Více se o konkrétních RTC – DS1307 a DS3231 – dozvíte na webových stránkách výrobce.

Přeloženo z <http://tronixstuff.com/2014/12/01/tutorial-using-ds1307-and-ds3231-real-time-clock-modules-with-arduino/> a mírně upraveno.

[About](#)[Latest Posts](#)

Antonín Handl

Působí v časopise PLAV – měsíčník pro světovou literaturu, kde nyní jako zástupce šéfredaktora odpovídá především za propagaci a správu sociálních sítí. Příležitostně překládá odbornou literaturu a soustavně se věnuje ilustraci a grafickým návrhům. Zajímá se o open source a DIY projekty, crowdfunding a alternativní marketing.

25 COMMENTS ON “TUTORIÁL – UŽÍVÁNÍ HODIN REÁLNÉHO ČASU DS1307 A DS3231 S ARDUINEM”

Piter

15.3.2018 at 21:10

Dobrý den.Po nahrání kodu se veškeré hodnoty zobrazují v jednom řádku.A hlavně jsou všechny špatně třeba datum je samé 165/165/165 a čas je něco podobného.Prosím o radu.Děkuji

Pro vložení odpovědi na komentář se musíte přihlásit

Pat

11.2.2018 at 20:44

Dobrý den, mohl bych poprosit o radu?

Chtěl bych zapsat následující podmínku pro časový spínač – v časech 6:30 – 8 hod nebo 17 – 20 hod se bude vykonávat určitá funkce.

Vše mi funguje správně když použiji jen hodiny. Ve chvíli, kdy chci zapsat minuty není podmínka správná.

Funkce se začne vykonávat v 6:30, ale v 7:00 skončí a začne zase v 7:30...

```
if (((hour >= 6 && minute >= 30) && hour = 17 && hour < 20))
```

Pro vložení odpovědi na komentář se musíte přihlásit

Zbyšek Voda

11.2.2018 at 20:55

Šel bych na to jinak.

Zavedeme si něco jako timestamp – ten bude udávat počet minut od začátku dne.

Potom třeba 1:00 by bylo převedeno na 60, 2:00 na 120, ...

Převod by byl následující:

```
int timestamp = hours * 60 + minutes;
```

Potom 6:30 odpovídá timestamp 390, 8:00 timestamp 480 a první podmínka bude tedy:

```
if(timestamp >= 390 && timestamp <= 480) {...}
```

Obdobně i pro druhou podmínku.

Popsal jsem to srozumitelně?

Pro vložení odpovědi na komentář se musíte přihlásit

Pat

12.2.2018 at 13:06

Dobrý den, jak jednoduché :) Vždy se hodí nový pohled. Srozumitelné to je, vyzkouším a dám vědět. Každopádně děkuji za rychlou odpověď.

Pro vložení odpovědi na komentář se musíte přihlásit

David

2.11.2016 at 13:38

Dobrý den, potřebuji poradit kde v programu se nastavuje čas? Zkoušel jsem to v tom to řádku ale hlásí to chybu. void setDS3231time(byte second, byte minute, byte hour, byte dayOfWeek, byte dayOfMonth, byte month, byte year){

Pro vložení odpovědi na komentář se musíte přihlásit

Zbyšek Voda

3.11.2016 at 9:21

Dobrý den, čas se nastavuje funkcí setDS3231time. Jakou chybu vám to hlásí?

Pro vložení odpovědi na komentář se musíte přihlásit

mfly

22.12.2016 at 19:03

Dobrý den,

jak mám nastavit hodiny? Nastavuji ve funkci setDS3231time, ale absolutně bez

reakce. Mohu poprosit o kousek kódu s konkrétním nastavením určitého času?
Děkuji

Pro vložení odpovědi na komentář se musíte přihlásit

Zbyšek Voda

25.12.2016 at 16:59

Dobrý den, spíš prvně nasdílejte Váš kód :)
Voláte tu funkci setDS3231time?

Pro vložení odpovědi na komentář se musíte přihlásit

mfly

27.12.2016 at 19:59

Díky za reakci. Již jsem rozchodil. Zadával jsem hodnoty do špatné funkce. Ale teď řeším jiný problém s podmínkou a časem – spuštění na základě času. Mohu poprosit o email nebo interní komunikaci? Nasdílel bych Vám kód a byl bych rád za radu. Můj email – mfly@seznam.cz

Díky

Vili

29.10.2016 at 23:56

Prosím, kde mám zrušit, resp pridať koment // pri inicializácii údajov?
Kompilácia mi zlyháva.

```
// set the initial time here:  
// DS3231 seconds, minutes, hours, day, date, month, year  
//setDS3231time(30,42,21,4,26,11,14);  
}  
void setDS3231time(byte second, byte minute, byte hour, byte dayOfWeek, byte  
dayOfMonth, byte month, byte year)
```

Pro vložení odpovědi na komentář se musíte přihlásit

Vili

30.10.2016 at 10:31

Po preinštalovaní knižnice DS3231 bol problém odstránený.

Pro vložení odpovědi na komentář se musíte přihlásit

Sakul

10.2.2016 at 3:55

Také jsem testoval DS1307 a je dle mého názoru naprosto nepoužitelný. Od obvodu reálného času bych očekával alespoň minimální přesnost. Ale dle testů nemá tento obvod problém ujet za 24 hodin o 10 sekund.

Poté co jsem otestoval DS3231 mi nesmí DS1307 do baráku. DS3231 je dostatečně přesný pro použití do hodin a navíc má i integrovaný teploměr, takže paráda. Mohu jen doporučit.

Pro vložení odpovědi na komentář se musíte přihlásit

haberturdeur

9.9.2015 at 19:43

Když se pokusím o kompilaci vyhodí mi to tyto chybové hlášky, prosím nevíte co s tím?

Arduino: 1.6.5 (Windows XP), Vývojová deska: „Arduino/Genuino Uno“

Volby pro sestavení se změnily; sestavuji vše znovu

sketch_sep09a.ino: In function ,void setDS3231time(byte, byte, byte, byte, byte, byte, byte):

sketch_sep09a:23: error: ,DS3231_I2C_ADDRESS' was not declared in this scope

sketch_sep09a.ino: In function ,void readDS3231time(byte*, byte*, byte*, byte*, byte*, byte*, byte*):

sketch_sep09a:36: error: ,DS3231_I2C_ADDRESS' was not declared in this scope
,DS3231_I2C_ADDRESS' was not declared in this scope

(nehledě na to (pravděpodobně moje nechápavost) kam mám napsat ty časové údaje)

Pro vložení odpovědi na komentář se musíte přihlásit

Zbyšek Voda

9.9.2015 at 19:47

Dobrý den, zkuste to znovu. Měli jsme chybu v tom, že první dva řádky byly napsané na jednom.

Pro vložení odpovědi na komentář se musíte přihlásit

Kolbis

12.2.2015 at 8:24

Jen dvě poznámky.

1. Modul s DS3231 předpokládá, že se jako záložní článek použije dobíjecí varianta. Obsahuje tedy dobíjecí odpor a diodu. Pokud se použije klasický nedobíjecí článek CR2032, je vhodné ten dobíjecí obvod přerušit.
2. Co se týče automatické změny letní/zimní čas. Existuje na to knihovna a funguje úplně super. Hledejte <https://github.com/JChristensen/Timezone>

Pro vložení odpovědi na komentář se musíte přihlásit

David

15.2.2015 at 8:24

Dobrý den, potřeboval bych nahradit dobíjecí baterii obyčejnou – potřebuji, aby RTC mělo i po připojení k napájení co nejmenší odběr. Můžu poprosit, nevíte přesně, který odpor a diodu „vyhodit“? Jedná se mi o DS3231.

Díky David

Pro vložení odpovědi na komentář se musíte přihlásit

Kolbis

16.2.2015 at 9:04

Mrkni na ten obrázek s modulem DS3231. Vpravo nahoře nad nápisem SCL je odpor a vlevo vedle nápisu SCL dioda. Stačí odletovat jeden z těchto prvků nebo přerušit spoj mezi nimi. Pokud Ti jde opravdu o co nejmenší odběr, pak bych odletoval ještě LED diodu signalizující přítomnost napájení a pokud nepoužíváš EEPROM 24C32, tak bych vyhodil i ten.

Pro vložení odpovědi na komentář se musíte přihlásit

ewrc

31.1.2015 at 21:42

no a je ticho :(

Pro vložení odpovědi na komentář se musíte přihlásit

jiri pospisil

27.1.2015 at 23:05

Pro precho na zimni nebo letni cas se nikdy nepouzi lva pulnoc ale tusim 2 hodina ranni. Potom jsou podminky jednoduche.

Pro vložení odpovědi na komentář se musíte přihlásit

ewrc

28.1.2015 at 22:57

áno,čas sa posúva z2:00na 3:0 a nazad, no to nerieši môj problém, pri aplikovaní hour++ je proste 23:00 +1 = 24:00 čo je naprd, a tak by som uvítal nejaký návod alebo časť kodu ako to vyriešiť. Preložených tutorialov na RTC je na nete hafo, no nikto nerieši práve tento automatický posun času ktorý stále aspoň do roku 2016 platí. no možno by stálo za to upraviť knižnicu napr.. RTCLib a potom v programe len zadať časové podmienky zmeny času (no ja si nato netrúfam).
proste uvítam akúkoľvek pomoc a nielen ja.
stále čakám ako to vyriešiť.
ďakujem

Pro vložení odpovědi na komentář se musíte přihlásit

Jiří Patera

11.4.2015 at 10:28

Potíž přechodu na letní/zimní čas jste asi už vyřešil, ale přeci jen... Pokud máte informaci o dnu v týdnu. Tu buď máte nebo si ji zařídíte vlastním zálohovaným počítadlem dní. Pak je to velmi jednoduché.
Na letní čas se přechází poslední neděli v březnu a na zimní čas se přechází poslední neděli v říjnu. Pokud nám to odpůrci letního času nebo EU nezmění.
Snadno si otestuji, zda aktuální neděle je tou poslední v zajímavém měsíci.
Pokud je to v březnu, v 02:00:00 přestavím hodiny na 03:00:00 (s ev. kompenzací času, pokud jsem nějaký čas prošvihl).
Pokud je to v říjnu, při průchodu v 03:00:00 otestuji a pak překlopím flag. Pokud flag byl původně vynulovaný, přestavím hodiny na 02:00:00 (s ev. kompenzací, pokud jsem nějaký čas prošvihl).
Pozorný čtenář už pochopil, že flag je tu nutný jako ochrana proti zacyklení přestavování. Jeho hodnota true označuje opakující se úsek času od 2 do 3hod,, kdy už je nastaven čas zimní.
Algoritmus neřeší ty mamlasy, kteří ručně nastavují hodiny v době, kdy je flag nahozený.

Pro vložení odpovědi na komentář se musíte přihlásit

ewrc

26.1.2015 at 13:14

ano každý tutorial je poučný, sám používám DS3231 a knižnici RTCLib.h,
no ako posunúť automaticky čas letný- zimný sa je problém dopátrať nieke, na nete je toho málo. Skúšal som cez rozne podmienky, pridať ako hour++, prípadne rovno prestaviť čas na DS3231 aj to čiastočne fungovalo len to ma lo malé muchy
(miesto 00:00 bolo 24:00 a nenabehol nový den, alebo posunutie času naspäť atd...) proste doteraz som to nevyriešil a zatiaľ si neviem rady. a tak by som privítal ako nato.. -tutorial . (čo najjednoduchšie riešenie, nejak to oblafnúť)
ďakujem
PS časový údaj slúži pre logger ako datová pečiatka a tým že je to mimo civilizácie není to možné prestaviť ručne.

Pro vložení odpovědi na komentář se musíte přihlásit

McArduRap

26.1.2015 at 12:27

Rád bych se vyjádřil k překladu: „To se může jevit jako problém, ve skutečnosti to ale znamená, že hodiny vypadnou přibližně na pět minut v měsíci.“ Myslím, že nejde o výpadek, ale o časový posun (clocks are off / clock drift / skew) až pět minut za měsíc.

Pokud jde o mé vlastní zkušenosti s jedním exemplářem DS1307 čipu, zrychluje se tento měsíčně o cca 2 minuty při běžných pokojových teplotách (18 až 26 °C). Je to nepříjemné, ale dá se to pořešit denními korekcemi aktuálního času v danou fixní dobu o cca 2 vteřiny. Naopak příjemných je těch pár desítek bajtů statické paměti navíc.

Pro vložení odpovědi na komentář se musíte přihlásit

Zbyšek Voda

26.1.2015 at 18:09

Děkuji. Máte pravdu. Opraveno.

Pro vložení odpovědi na komentář se musíte přihlásit

Aleš Filip

26.1.2015 at 9:20

Jenom bych chtěl upozornit, že DS1307 stojí dost za ho.... Hodiny se dokážou rozejít za 14 dní až o 5 minut. Mám jich několik, protože

jsem myslel, že je první kus vadný. Nepomůže ani výměna krystalu, ani přidání kondenzátorů ke krystalu. DS3231 je už v pohodě. Takže pozor, kupovat DS1307 jsou vyhozené peníze.

Pro vložení odpovědi na komentář se musíte přihlásit

Napsat komentář

Pro přidávání komentářů se musíte nejdříve přihlásit.

HLEDAT



PŘIHLÁSIT SE | [INFO@ARDUINO.CZ](mailto:info@arduino.cz) | [ARDUINO.CZ](http://arduino.cz) 2014-
2018
POWERED BY [HWKITCHEN.CZ](http://hwkitchen.cz)

