

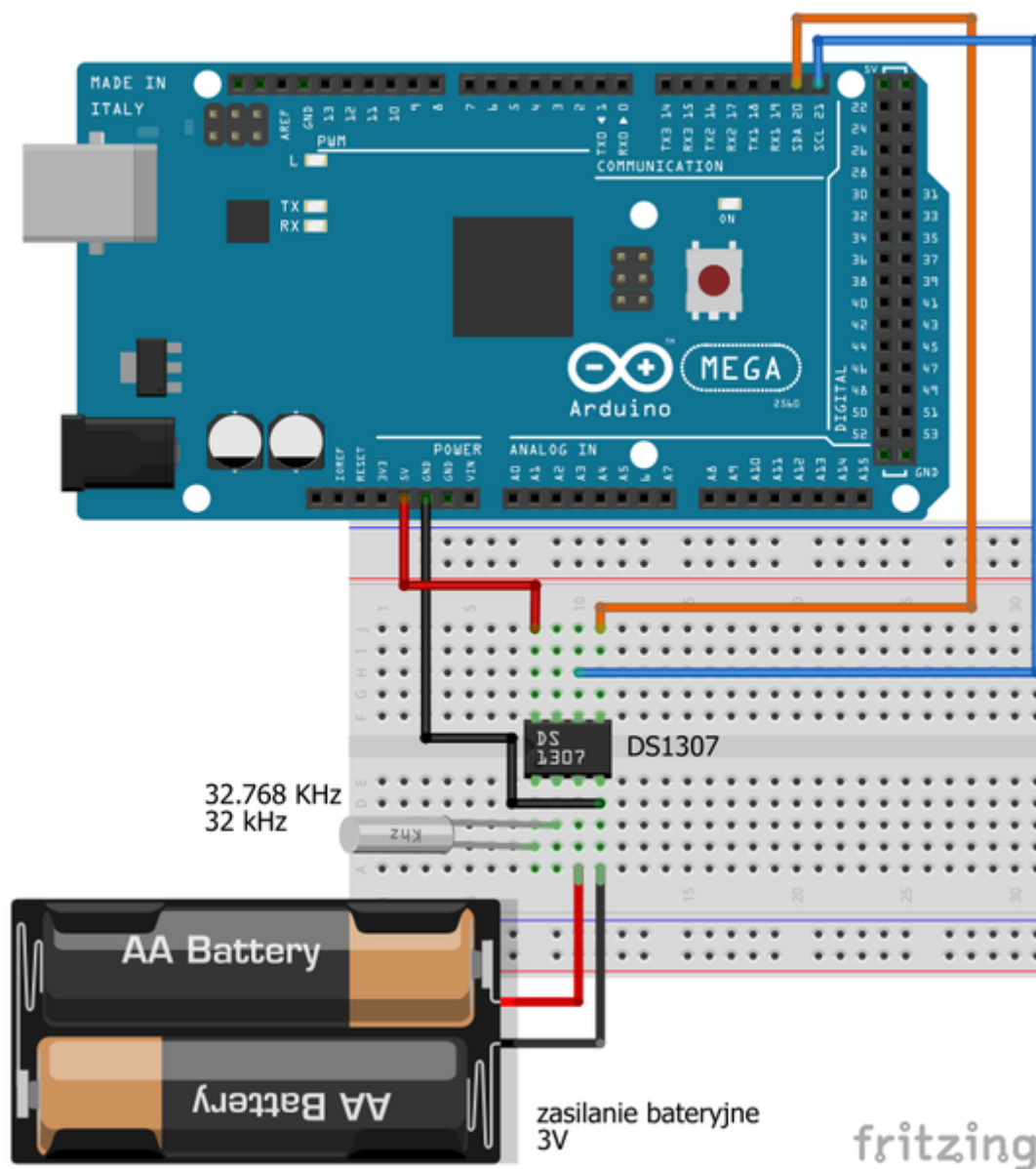


Nazwa implementacji: Budowa RTC w oparciu o DS1307

Autor: Krzysztof Bytow

Opis implementacji: Układ DS1307 jest to zegar czasu rzeczywistego (Real Time Clock) służy do odliczania czasu niezależnie od stanu układu (z reguły podtrzymywany z baterii, co umożliwi mu pracę gdy główny układ jest odcięty od zasilania).

1. Schemat połączeń:



SCL -> Pin 21
SDA -> Pin 20
VCC -> 5V
GND -> GND





Uwaga!!!

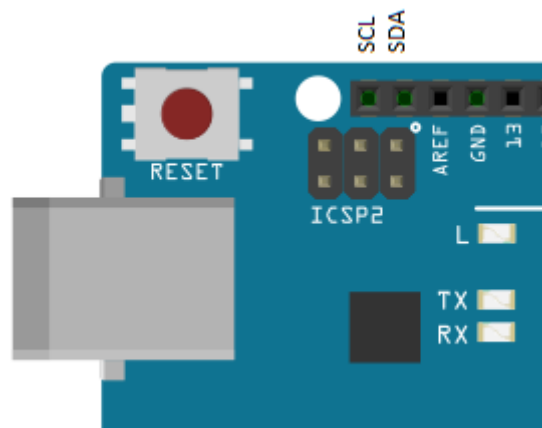
W Arduino UNO R3 można wykorzystać:

SCL -> A5 Analog Pin

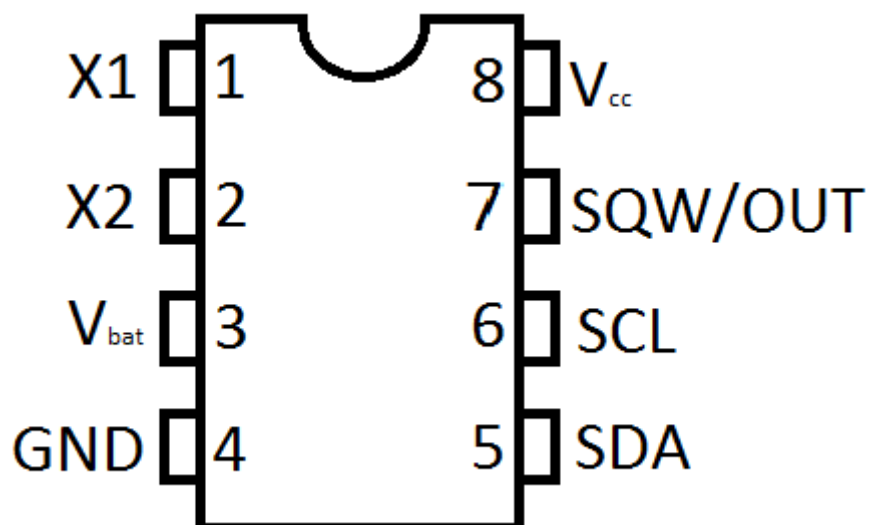
SDA -> A4 Analog Pin

lub

dotychczasowe porty SDA i SCL (ilustracja poniżej)



2. Wyprowadzenia układu DS1307 8-Pin DIP:



- V_{cc} – zasilanie (min. 4.5V max. 5.5V);
- SQW/OUT- wyj. sygnału prostokątnego
- SCL – sygnał zegara magistrali I2C;
- SDA – sygnał danych magistrali I2C;
- X1,X2 – kwarc 32.768 KHz;
- V_{bat} – podtrzymanie baterijne (min. 2V max. 3.5V);





- GND – masa

3. Elementy niezbędne do budowy układu:

- kwarc 32.768 KHz;
- DS1307;
- opcjonalne:
 - bateria podtrzymująca (3V np.: CR2032)
 - goldpin (2 piny) lub podstawka do baterii CR2032 lub koszyczek na baterie AA (2szt.)

W kwestii zasilania bateryjnego, można je pominąć lecz nie będzie zachowany czas (należy ponownie ustawić prawidłowy czas), innym rozwiązaniem jest użycie koszyczka na baterie AA (2szt. 1.5V) lub podstawki do baterii CR2032. W projekcie wykorzystałem ciut cieńszą odmianę, a dokładniej CR2016, z braku podstawki wykorzystałem złącze goldpin delikatnie spłaszczając górę.

4. Kod implementacji – do prawidłowej pracy niezbędna jest biblioteka RTC, przykładową można pobrać ze strony

więcej...

Przy pierwszym uruchomieniu należy usunąć komentarz w linijce:

```
//czas.adjust(DateTime(DATE, TIME));
```

wiersz ten ustawia czas na czas taki jak w komputerze. Po udanym wgraniu/ustawieniu czasu w DS1307, wiersz należy ponownie skomentować lub wykasować i wgrać do Arduino.

```
#include <Wire.h>
#include "RTClib.h"

RTC_DS1307 czas;

void setup ()
{
  Serial.begin(9600);
  Wire.begin();
  czas.begin();
  //czas.adjust(DateTime(__DATE__, __TIME__));
}

void loop ()
{
  DateTime now = czas.now();
  Serial.print(now.day(), DEC);
  Serial.print('.');
  Serial.print(now.month(), DEC);
  Serial.print('.');
  Serial.print(now.year(), DEC);
  Serial.print(' ');
  Serial.print(now.hour(), DEC);
```



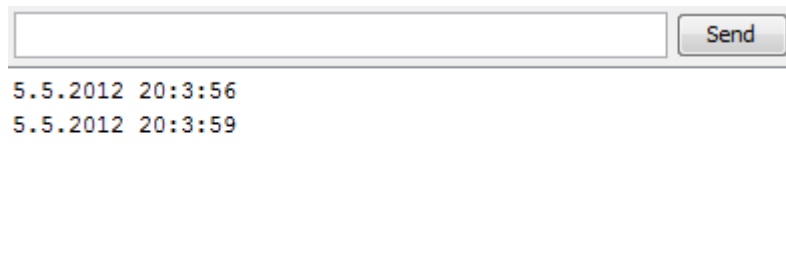


```
Serial.print(':');
Serial.print(now.minute(), DEC);
Serial.print(':');
Serial.print(now.second(), DEC);
Serial.println();
delay(3000);
}
```

Po wgraniu kodu do Arduino i uruchomieniu Monitora portu szeregowego zostaną zaprezentowane odczytane dane:



Odczyt ustawienia czasu:



5. Przykładowe zastosowanie – sterowanie diodą wbudowaną w Arduino Pin 13 – kod przykładowy, działanie sprowadza się do ustawienia w kodzie

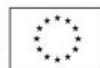
```
if(godzina=19 & minut>=30 & minut<=45)
```

czasu włączenia i wyłączenia diody. Dioda zostanie zapalona o 19:30, a zgaszona o 19:46

```
#include <Wire.h>
#include "RTClib.h"
int led = 13;
RTC_DS1307 czas;

void setup ()
{
  pinMode(led, OUTPUT);
  Wire.begin();
  czas.begin();
  //czas.adjust(DateTime(__DATE__, __TIME__));
}

void loop ()
{
  DateTime now = czas.now();
  int godzina=now.hour();
  int minut=now.minute();
  if(godzina=19 & minut>=30 & minut<=45)
  {
    digitalWrite(led, HIGH);
  }
  else
  {
```





```
digitalWrite(led, LOW);  
}  
delay(1000);  
}
```

