

(<https://www.dfrobot.com/product-879.html>)

## Introduction

The DS1307 Real Time Clock developed by one of our designers: Waiman. The module comes fully assembled and pre-programmed with the current time (ok, so it's our current time - MST). The included Lithium coin cell battery (CR1225 41mAh) will run the module for a minimum of 9 years (17 years typical) without external 5V power. The DS1307 is accessed via the I2C protocol.

Except the DS1307 real time clock, the module also integrate the I2C EEPROM chip(24C32) and the DS18B20 sensor interface. The I2C EEPROM makes it possible to save the sensor data easily without spending too much microcontroller source. And the DS18B20 interface with the pull-up resistor embeded will be helpful to extend temperature monitoring feature for your project.

## Specification

- Working voltage: 5v
- Two wire I2C interface
- Hour : Minutes : Seconds AM/PM
- Day Month, Date - Year
- Leap year compensation
- Accurate calendar up to year 2100
- Consumes Less than 500nA in Battery-Backup
- Battery included
- 1Hz output pin
- 56 Bytes of Non-volatile memory available to user

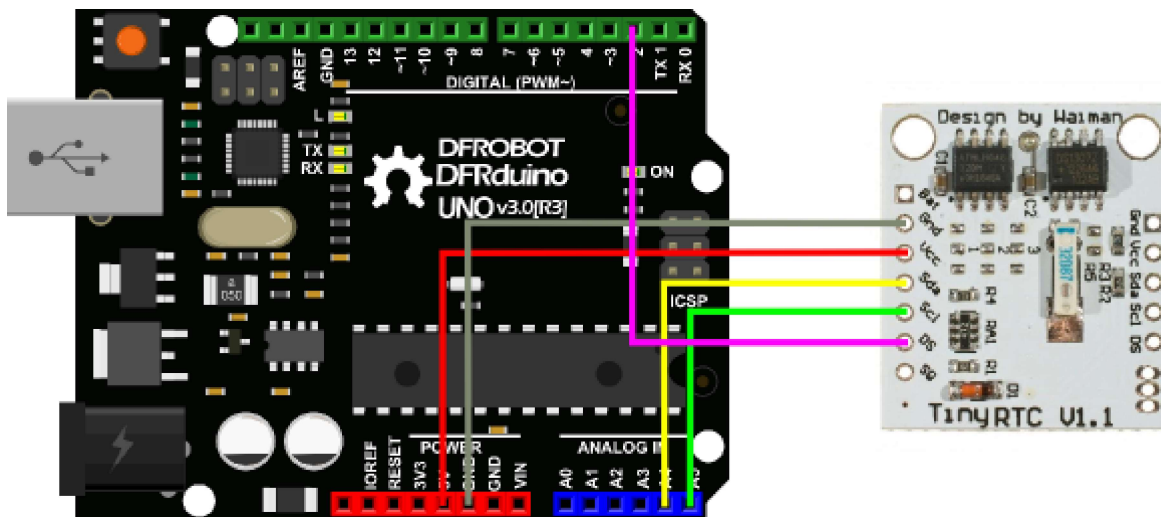
- 512 bytes of non-volatile memory available to user
- provides 32,768 bits(4KB) of serial electrically erasable and programmable read only memory (EEPROM)
- Embed DS18B20 temperature sensor interface with the pull-up resistor
- Dimensions: 28mm x 25mm x 8mm

## Application

- Real time monitoring system
- Timer

## Tutorial

### Connection diagram



#### Wiring instruction

GND——GND

SDA——A4

DS——D2

### Sample Code

Please download and install the Arduino library (<https://www.dfrobot.com/image/data/DFR0151/V1.1/Arduino%20library.zip>) first. how to install library (<https://www.arduino.cc/en/Guide/Libraries#.UxU8mdzF9H0>)

```

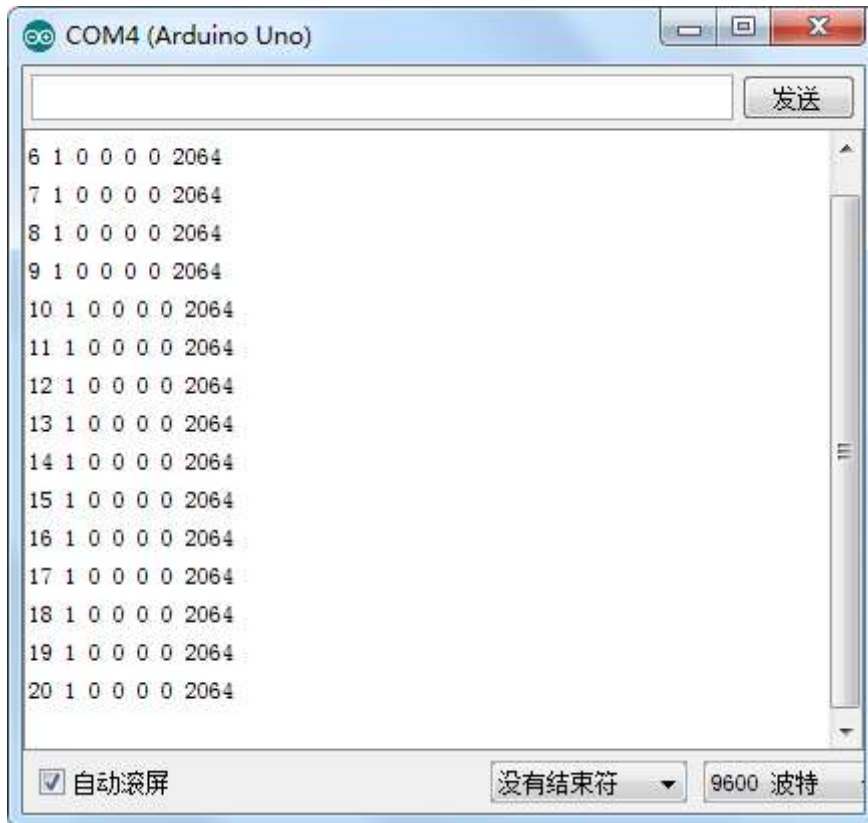
#include <Wire.h>
#include <DS1307.h>
/*
about set time:
format: year,month,day,week,hour,min,sec
example: 14,03,25,02,13,55,10   2014.03.25 tuesday 13:55:10
*/
String comdata = "";
int mark=0;
//store the current time data
int rtc[7];
//store the set time data
byte rr[7];
//light pin
int ledPin = 13;
//initial light
void setup()
{
  DDRC |= _BV(2) | _BV(3); // POWER:Vcc Gnd
  PORTC |= _BV(3); // VCC PINC3
  pinMode(ledPin, OUTPUT);
  //initial baudrate
  Serial.begin(9600);
  //get current time
  RTC.get(rtc, true);
  //if time is wrong reset to default time
  if (rtc[6] < 12) {
    //stop rtc time
    RTC.stop();
    RTC.set(DS1307_SEC, 1);
    RTC.set(DS1307_MIN, 27);
    RTC.set(DS1307_HR, 01);
    RTC.set(DS1307_DOW, 7);
    RTC.set(DS1307_DATE, 12);
    RTC.set(DS1307_MTH, 2);
    RTC.set(DS1307_YR, 12);
    //start rtc time
    RTC.start();
  }
  //RTC.SetOutput(LOW);
  //RTC.SetOutput(HIGH);
  //RTC.SetOutput(DS1307_SQW1HZ);
  //RTC.SetOutput(DS1307_SQW4KHZ);
  //RTC.SetOutput(DS1307_SQW8KHZ);
  RTC.SetOutput(DS1307_SQW32KHZ);
}

void loop()
{

```

```
int i;
//get current time
RTC.get(rtc, true);
//print current time format : year month day week hour min sec
for (i = 0; i < 7; i++)
{
    Serial.print(rtc[i]);
    Serial.print(" ");
}
//blink the light
Serial.println();
digitalWrite(ledPin, HIGH);
delay(500);
digitalWrite(ledPin, LOW);
delay(500);
//
int j = 0;
//read all the data
while (Serial.available() > 0)
{
    comdata += char(Serial.read());
    delay(2);
    mark = 1;
}
//if data is all collected,then parse it
if (mark == 1)
{
    Serial.println(comdata);
    Serial.println(comdata.length());
    //parse data
    for (int i = 0; i < comdata.length() ; i++)
    {
        //if the byte is ',' jump it,and parse next value
        if (comdata[i] == ',')
        {
            j++;
        }
        else
        {
            rr[j] = rr[j] * 10 + (comdata[i] - '0');
        }
    }
    comdata = String("");
    RTC.stop();
    RTC.set(DS1307_SEC, rr[6]);
    RTC.set(DS1307_MIN, rr[5]);
    RTC.set(DS1307_HR, rr[4]);
    RTC.set(DS1307_DOW, rr[3]);
    RTC.set(DS1307_DATE, rr[2]);
    RTC.set(DS1307_MTH, rr[1]);
    RTC.set(DS1307_YR, rr[0]);
    RTC.start();
    mark = 0;
}
}
```

## Result



## Trouble shooting

Thank the guy to share this great way to update DS1307 clock through computer, in this way, you could get the precise time.

Guide Link (<http://www.lucadentella.it/en/2013/11/27/rtcsetup/>) Download library

(<https://github.com/adafruit/RTCLib>) Download sketch & pc\_software

(<https://github.com/lucadentella/RTCSetup>)

More question and cool idea, visit DFRobot Forum (<https://www.dfrobot.com/index.php?route=DFblog/blogs>)

## More

- Real Time Clock Module (DS1307) V1.0 ([https://www.dfrobot.com/wiki/index.php?title=Real\\_Time\\_Clock\\_Module\\_DS1307\\_SKU:DFR0151\\_](https://www.dfrobot.com/wiki/index.php?title=Real_Time_Clock_Module_DS1307_SKU:DFR0151_))
- Arduino Library (<https://www.dfrobot.com/image/data/DFR0151/V1.1/Arduino%20library.zip>)
- Schematic (<https://www.dfrobot.com/image/data/DFR0151/V1.1/TinyRTC%20V1.1%20sch.pdf>)
- DS1307 Datasheet

(<https://www.dfrobot.com/image/data/DFR0151/V1.1/DS1307%20datasheet.pdf>)

- Forum discussion about EEPROM (<https://www.dfrobot.com/forum/viewtopic.php?f=8&t=935&p=4818#p4818>)
- Youtube: How to control a Real Time Clock. RTC using the Arduino Leonardo. (<https://youtu.be/aPOblVGm1us>)



Get **Real Time\_Clock Module DS1307 V1.1 SKU:DFR0151**

(<https://www.dfrobot.com/product-879.html>) from DFRobot Store or **DFRobot Distributor**.

(<https://www.dfrobot.com/index.php?route=information/distributorslogo>)

Category: DFRobot (<https://www.dfrobot.com/>) > Sensors & Modules

(<https://www.dfrobot.com/category-156.html>) > Modules (<https://www.dfrobot.com/category-211.html>)> RTC Modules (<https://www.dfrobot.com/category-232.html>) category: Product Manual

(category\_\_Product\_Manual) category: DFR Series (category\_\_DFR\_Series) category: Modules

(category\_\_Modules) category: Source (category\_\_Source) category: Diagram

(category\_\_Diagram) category: DFRobot (category\_\_DFRobot)