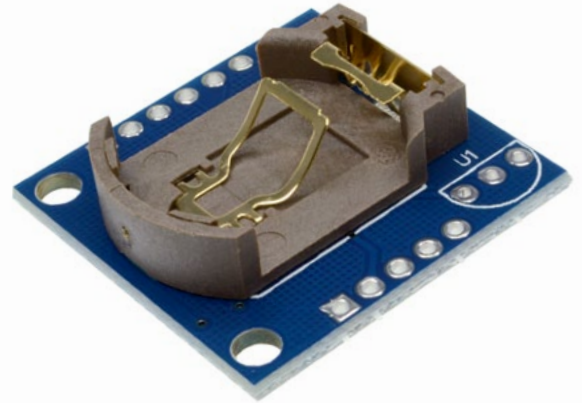


## ARD 2 **Arduino Compatibles**

Controllers, Shields, Modules & Sensors

### **DS1307 Real Time Clock Module** ARD2-2092

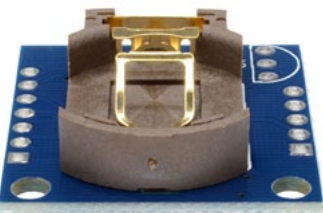
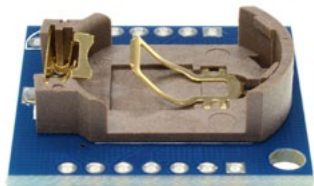
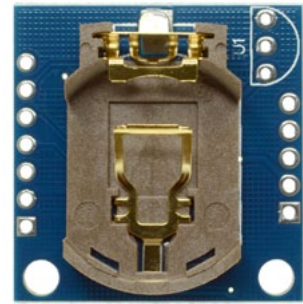
- **Track time even when module is unpowered**
- **Based on the DS1307 chip**
- **Keeps time for 5 years or more**
- **56-Byte, Battery-Backed, General-Purpose RAM with Unlimited Writes**
- **Programmable Square-Wave Output Signal**



#### Description

The DS1307 serial real-time clock (RTC) is a lowpower, full binary-coded decimal (BCD) clock/calendar plus 56 bytes of NV SRAM. It allows your microcontroller to keep track of time even if it is reprogrammed or power is lost. Address and data are transferred serially through an I2C, bidirectional bus. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The end of the month date is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with AM/PM indicator. The DS1307 has a built-in power-sense circuit that detects power failures and automatically switches to the backup supply. Timekeeping operation continues while the part operates from the backup supply.

**Note:** The DS1307 is not a high-precision device and may lose or gain up to 2 seconds per day.



#### Specifications

<b>Operating Voltage</b>	5V
<b>Main Chip</b>	DS1307
<b>Interface Type</b>	Serial I/O interface
<b>Dimensions</b>	27mm x 28mm x 8.4mm
<b>Weight</b>	40g
<b>Operating Temperature</b>	0°C – +70°C
<b>TTL Compatible</b>	Vcc=5V

#### Pinout

Module	Arduino	Function
VCC	5V	Power Supply
GND	GND	Ground Connection
SCL	D6	Serial Clock
SDA	D7	Data I/O
DS	D8	Programmable Square Wave Output (optional)

**Test Code**

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

RTC_DS1307 RTC;

void setup () {
  Serial.begin(57600);
  Wire.begin();
  RTC.begin();

  if (! RTC.isrunning()) {
    Serial.println("RTC is NOT running!");
    // following line sets the RTC to the date & time this sketch was compiled
    //RTC.adjust(DateTime(__DATE__, __TIME__));
  }
}

void loop () {
  DateTime now = RTC.now();

  Serial.print(now.year(), DEC);
  Serial.print('/');
  Serial.print(now.month(), DEC);
  Serial.print('/');
  Serial.print(now.day(), DEC);
  Serial.print(' ');
  Serial.print(now.hour(), DEC);
  Serial.print(':');
  Serial.print(now.minute(), DEC);
  Serial.print(':');
  Serial.print(now.second(), DEC);
  Serial.println();

  Serial.print(" since 1970 = ");
  Serial.print(now.unixtime());
  Serial.print("s = ");
  Serial.print(now.unixtime() / 86400L);
  Serial.println("d");

  // calculate a date which is 7 days and 30 seconds into the future
  DateTime future (now.unixtime() + 7 * 86400L + 30);

  Serial.print(" now + 7d + 30s: ");
  Serial.print(future.year(), DEC);
  Serial.print('/');
  Serial.print(future.month(), DEC);
  Serial.print('/');
  Serial.print(future.day(), DEC);
  Serial.print(' ');
  Serial.print(future.hour(), DEC);
  Serial.print(':');
  Serial.print(future.minute(), DEC);
  Serial.print(':');
  Serial.print(future.second(), DEC);
  Serial.println();

  Serial.println();
  delay(3000);
}
```