

```

/*
 * RTC Control v.01
 * Original <http://www.combustory.com> John Vaughters
 * Adapted by <http://www.fadishop.eu> Lleonard García Llop
 *
 * With this code you can set the date/time, retrieve the date/time
   and use the extra memory of an RTC DS3232 chip.
 * The program also sets all the extra memory space to 0xff.
 * Serial Communication method with the Arduino that utilizes a leading CHAR
   for each command described below.
 * Commands:
 * T(00-59)(00-59)(00-23)(1-7)(01-31)(01-12)(00-99) -
   T(sec)(min)(hour)(dayOfWeek)(dayOfMonth)(month)(year) -
   T Sets the date of the RTC DS3232 Chip.
 * Example to set the time for 02-Feb-09 @ 19:57:11 for the 3 day of the week
   use this command - T1157193020209
 * Q(1-2) - (Q1) Memory initialization (Q2) RTC - Memory Dump
 */

#include "Wire.h"
// Global Variables
int DS3232_I2C_ADDRESS = 0xD0 >> 1; // The address for arduino is the seven M

int command = 0; // This is the command char, in ascii form, sent fr
int i;
long previousMillis = 0; // will store last time Temp was updated
byte second, minute, hour, dayOfWeek, dayOfMonth, month, year;
byte test;

// 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) );
}

// 1) Sets the date and time on the ds3232
// 2) Starts the clock
// 3) Sets hour mode to 24 hour clock
// Assumes you're passing in valid numbers, Probably need to put in checks for

void setDateDs3232()
{

```

```

second = (byte) ((Serial.read() - 48) * 10 + (Serial.read() - 48));//
minute = (byte) ((Serial.read() - 48) * 10 + (Serial.read() - 48));//
hour = (byte) ((Serial.read() - 48) * 10 + (Serial.read() - 48));//
dayOfWeek = (byte) (Serial.read() - 48);
dayOfMonth = (byte) ((Serial.read() - 48) * 10 + (Serial.read() - 48));
month = (byte) ((Serial.read() - 48) * 10 + (Serial.read() - 48));
year = (byte) ((Serial.read() - 48) * 10 + (Serial.read() - 48));
Wire.beginTransaction(DS3232_I2C_ADDRESS);
Wire.send(0x30);
Wire.send(decToBcd(second)); // 0 to bit 7 starts the clock
Wire.send(decToBcd(minute));
Wire.send(decToBcd(hour)); // If you want 12 hour am/pm you need to set
// bit 6 (also need to change readDateDs3232 to 12)

Wire.send(decToBcd(dayOfWeek));
Wire.send(decToBcd(dayOfMonth));
Wire.send(decToBcd(month));
Wire.send(decToBcd(year));
Wire.endTransmission();
}

// Gets the date and time from the ds3232 and prints result
void getDateDs3232()
{
// Reset the register pointer
Wire.beginTransaction(DS3232_I2C_ADDRESS);
Wire.send(0x00);
Wire.endTransmission();

Wire.requestFrom(DS3232_I2C_ADDRESS, 7);

// A few of these need masks because certain bits are control bits
second = bcdToDec(Wire.receive() & 0x7f);
minute = bcdToDec(Wire.receive() & 0x7f);
hour = bcdToDec(Wire.receive() & 0x3f); // Need to change this if 12
dayOfWeek = bcdToDec(Wire.receive() & 0x07);
dayOfMonth = bcdToDec(Wire.receive() & 0x3f);
month = bcdToDec(Wire.receive() & 0x1f);
year = bcdToDec(Wire.receive() & 0x7f);

Serial.print(hour, DEC);
Serial.print(":");
Serial.print(minute, DEC);
Serial.print(":");
Serial.print(second, DEC);
Serial.print(" ");
Serial.print(month, DEC);
Serial.print("/");
Serial.print(dayOfMonth, DEC);
Serial.print("/");

```

```

Serial.print(year, DEC);

//Wire.endTransmission();
}

void setup() {
  Wire.begin();
  Serial.begin(9600);
}

void loop() {
  getDateDs3232();Serial.println("");
  delay(1000);
  if (Serial.available())      { // Look for char in serial que and process
    getDateDs3232();
    command =Serial.read();
    if (command == 84) { //If command = "T" Set Date
      setDateDs3232();
      getDateDs3232();
      Serial.println(" dummy ");
    }
    else if (command == 81) { //If command = "Q" RTC1307 Memory Function
      delay(100);
      if (Serial.available()) {
        command =Serial.read();
        if (command == 49) { //If command = "1" RTC1307 Initialize Memory
          // to 255 (0xff). Therefore 255 or 0 will be the initial value
          // that occurs when the RTC is initialized
          Wire.beginTransaction(DS3232_I2C_ADDRESS); // 255 will be the initial value
          Wire.send(0x08); // Set the register pointer to be just past the data
          for (i = 1; i <= 27; i++) {
            Wire.send(0xff);
            delay(100);
          }
          Wire.endTransmission();
          getDateDs3232();
          Serial.println(": RTC3232 Initialized Memory");
        }
        else if (command == 50) { //If command = "2" RTC1307 Memory Dump
          getDateDs3232();
          Serial.println(": RTC 3232 Dump Begin");
          Wire.beginTransaction(DS3232_I2C_ADDRESS);
          Wire.send(0x00);
          Wire.endTransmission();
          Wire.requestFrom(DS3232_I2C_ADDRESS, 64);
          for (i = 1; i <= 64; i++) {
            test =Wire.receive();

```

```
        Serial.print(i);
        Serial.print(":");
        Serial.println(test, DEC);
    }
    Serial.println(" RTC3232 Dump end");
}
}
}
Serial.print("Command: ");
Serial.println(command);    // Echo command CHAR
                            }// in ascii that was sent

        command = 0;// reset command
delay(1000);
}
//*****The End*****
```