```
/*
  "blinky" the one-eyed clock
 Version beta 2
 John Boxall August 2010
 http://tronixstuff.wordpress.com/projects > blinky
 Creative Commons Attribution-Share Alike 2.5 Australia
 DS1307/i2c timekeeping based on code by Maurice Ribble
 17-4-2008 - http://www.glacialwanderer.com/hobbyrobotics
 */

#include "Wire.h"
#define DS1307_I2C_ADDRESS 0x68
int red = 9; // LEDs connected to these pins as you might want to PWM them to alter brightness
int green = 10;
int blue = 11;
// 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 setDateDs1307(byte second,        // 0-59
byte minute,        // 0-59
byte hour,          // 1-23
byte dayOfWeek,     // 1-7
byte dayOfMonth,    // 1-28/29/30/31
byte month,         // 1-12
byte year)          // 0-99
{
  Wire.beginTransmission(DS1307_I2C_ADDRESS);
  Wire.send(0);
  Wire.send(decToBcd(second));    // 0 to bit 7 starts the clock
  Wire.send(decToBcd(minute));
  Wire.send(decToBcd(hour));
  Wire.send(decToBcd(dayOfWeek));
  Wire.send(decToBcd(dayOfMonth));
  Wire.send(decToBcd(month));
  Wire.send(decToBcd(year));
  Wire.send(0x10); // sends 0x10 (hex) 00010000 (binary) to control register - turns on square wave
  Wire.endTransmission();
}
void getDateDs1307(byte *second,
byte *minute,
byte *hour,
byte *dayOfWeek,
byte *dayOfMonth,
byte *month,
byte *year)
```

```
{
  // Reset the register pointer
  Wire.beginTransmission(DS1307_I2C_ADDRESS);
  Wire.send(0);
  Wire.endTransmission();
  Wire.requestFrom(DS1307_I2C_ADDRESS, 7);
  *second     = bcdToDec(Wire.receive() & 0x7f);
  *minute     = bcdToDec(Wire.receive());
  *hour       = bcdToDec(Wire.receive() & 0x3f);  // Need to change this if 12 hour am/pm
  *dayOfWeek  = bcdToDec(Wire.receive());
  *dayOfMonth = bcdToDec(Wire.receive());
  *month      = bcdToDec(Wire.receive());
  *year       = bcdToDec(Wire.receive());
}

void blinkLED(int colour, int ondelay, int offdelay, int blinks)
// blinks LED on pin 'colour' for 'blinks' times with on and off delay of 'ondelay', 'offdelay'
// colour: 9 is red, 10 is green, 11 is blue
{

  for (int a=0; a<blinks; a++)
  {
    digitalWrite(colour, HIGH);
    delay(ondelay);
    digitalWrite(colour, LOW);
    delay(offdelay);
  }
}

void blinkTime()
// blinks the time
{
  byte second, minute, hour, dayOfWeek, dayOfMonth, month, year;
  float aa;
  int bb;
  getDateDs1307(&second, &minute, &hour, &dayOfWeek, &dayOfMonth, &month, &year);

  // convert hours from 24 to 12 hour time
  if (hour==0)
  {
    hour=12;
  }
  if (hour>12)
  {
    hour=hour-12;
  }
  blinkLED(9, 500, 500, hour); // blink hours in red
  blueGlow(1,10);
  aa=minute;
  aa=aa/10;
  bb=int(aa); // find the value of tens of minutes (0~5)
  if (bb>0)
```

```
  {
   blinkLED(10, 500, 500, bb); // blink tens of minutes
  }
  if (bb==0) // but if the time is something like 03:02?
  {
   blinkLED(11, 200, 200, 1); // blink blue quickly for zero
  }
  aa=minute % 10; // find modulo of minutes to get single minutes
  bb=aa;
  if (bb>0)
  {
   blinkLED(9, 500, 500, bb); // blink tens of minutes
  }
  if (bb==0)
  {
   blinkLED(11, 200, 200, 1); // blink blue quickly for zero
  }
}

void whiteGlow(int n, int d)
{
  for (int nn=0; nn<n; nn++)
  {
    for (int a=0; a<=255; a++)
    {
     analogWrite(red, a);
     analogWrite(green, a);
     analogWrite(blue, a);
     delay(d);
    }
    for (int a=255; a>=0; --a)
    {
     analogWrite(red, a);
     analogWrite(green, a);
     analogWrite(blue, a);
     delay(d);
    }
  }
}

void blueGlow(int n, int d)
{
  for (int nn=0; nn<n; nn++)
  {
    for (int a=0; a<=255; a++)
    {
     analogWrite(blue, a);
     delay(d);
    }
    for (int a=255; a>=0; --a)
    {
      analogWrite(blue, a);
```

```
      delay(d);
    }
  }
}

void setup()
{
  byte second, minute, hour, dayOfWeek, dayOfMonth, month, year;
  Wire.begin();
  second = 0;
  minute = 17;
  hour = 4;
  dayOfWeek = 6; // these values are moot, but need to store something
  dayOfMonth = 28;
  month = 5;
  year = 10;
  setDateDs1307(second, minute, hour, dayOfWeek, dayOfMonth, month, year); // every time
blinky has new batteries, it will start from midnight/midday
  pinMode(red, OUTPUT);
  pinMode(green, OUTPUT);
  pinMode(blue, OUTPUT);
}

void loop()
{
  whiteGlow(1, 10); // glow white - announces that the time will now be shown
  delay(1000); // give people a second to focus on blinky
  blinkTime();
  delay(10000); // wait ten seconds
}
```