

```

/*
Exercise 10.1
Two channel on/off timer

tronixstuff.com/tutorials > Chapter 10
based on code by Maurice Ribble
17-4-2008 - http://www.glacialwanderer.com/hobbyrobotics
and John Boxall - http://tronixstuff.com
CC by-sa 2.5 Australia

*/
// these arrays store timer data
// on hour, on minute, off hour, off minute, day of week [1-7 Sun>Sat, 8 = daily, 9 = weekends, 10 =
weekdays], on/off (0 off 1 on)
int t1[7]= {
  0,0,0,0,0,0,0};
int t2[7]= {
  0,0,0,0,0,0,0};
int dd=1000; // used for delay in show timer data

#include "Wire.h"
#define DS1307_I2C_ADDRESS 0x68
#include <LiquidCrystal.h> // we need this library for the LCD commands
LiquidCrystal lcd(12,11,5,4,3,2);

// 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 ds1307
// 2) Starts the clock
// 3) Sets hour mode to 24 hour clock

// Assumes you're passing in valid numbers

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();
}

// Gets the date and time from the ds1307
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);

    // A few of these need masks because certain bits are control bits
    *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());
}

int readdial(int rangemax, int dialpin)
// rangemax is the number of values in your range, e.g. if you want 0~9, set rangemax to be '10'
// dialpin is the analog pin number connected to the potentiometer to read
{
    int kv=0;
    int kr=0;
    int kb=0;
    float a=0;
    float rd=0;
    rd=1023/rangemax;
    kb=analogRead(dialpin); // read potentiometer connected to analog pin 1
    a=kb/rd;
    kr=int(a);
    if (kr>rangemax)
    {
        kr=rangemax;
    }
    return kr;
}

void displaymenu()
// display the menu options, selectable by using the knob
{
    delay(300); // for debounce
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Turn knob slowly");
    lcd.setCursor(0,1);
    lcd.print("to select option");
    while (digitalRead(8)==LOW)
    {
        if (readdial(6,1)==0) {
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("Show timer data");
        }
        else if (readdial(6,1)==1) {
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("Set timer 1");
        }
        else if (readdial(6,1)==2) {
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("Set timer 2");
        }
        else if (readdial(6,1)==3) {
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("Timers on/off");
        }
        else if (readdial(6,1)==4) {

```

```

    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Set time and day");
}
else if (readdial(6,1)==5) {
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("exit menu");
}
delay(100); // stop screen flicker
}

switch(readdial(6,1))
{
case 0:
    displaytimerdata(); // done, untested
    break;
case 1:
    msetalarm1();
    break;
case 2:
    msetalarm2();
    break;
case 3:
    timeronoff(); // done, untested
    break;
case 4:
    settimeday(); // done, untested
    break;
}
// if the knob is 5, that is for return to clock display. function will end and return to clock by
default
}

void displaytimerdata()
// scrolls through settings for the timers
{
    lcd.clear(); // clear LCD screen
    lcd.setCursor(0,0);
    lcd.print(" Timer One: ");
    if (t1[5]==0)
    {
        lcd.print("Off");
    }
    else if (t1[5]==1)
    {
        lcd.print("On");
    }
    lcd.setCursor(0,1);
    lcd.print("On: ");
    if (t1[0]<10)
    {
        lcd.print("0");
    }
    lcd.print(t1[0]);
    if (t1[1]<10)
    {
        lcd.print("0");
    }
    lcd.print(t1[1]);
    lcd.print("h");
    delay(dd);
    lcd.setCursor(0,1);
    lcd.print(" ");
    lcd.setCursor(0,1);
    lcd.print("Off: ");
    if (t1[2]<10)
    {
        lcd.print("0");
    }
    lcd.print(t1[2]);
    if (t1[3]<10)

```

```
{
  lcd.print("0");
}
lcd.print(t1[3]);
lcd.print("h");
delay(dd);
lcd.setCursor(0,1);
lcd.print("          ");
lcd.setCursor(0,1);
lcd.print("Freq: ");
switch(t1[4])
{
case 1:
  lcd.print("Sundays");
  break;
case 2:
  lcd.print("Mondays");
  break;
case 3:
  lcd.print("Tuesdays");
  break;
case 4:
  lcd.print("Wed'days");
  break;
case 5:
  lcd.print("Thursdays");
  break;
case 6:
  lcd.print("Fridays");
  break;
case 7:
  lcd.print("Saturdays");
  break;
case 8:
  lcd.print("Daily");
  break;
case 9:
  lcd.print("Weekends");
  break;
case 10:
  lcd.print("Weekdays");
  break;
}
delay(dd);
lcd.clear(); // clear LCD screen
lcd.setCursor(0,0);
lcd.print(" Timer Two: ");
if (t2[5]==0)
{
  lcd.print("Off");
}
else if (t2[5]==1)
{
  lcd.print("On");
}
lcd.setCursor(0,1);
lcd.print("On: ");
if (t2[0]<10)
{
  lcd.print("0");
}
lcd.print(t2[0]);
if (t2[1]<10)
{
  lcd.print("0");
}
lcd.print(t2[1]);
lcd.print("h");
delay(dd);
lcd.setCursor(0,1);
lcd.print("          ");
lcd.setCursor(0,1);
```

```
lcd.print("Off: ");
if (t2[2]<10)
{
  lcd.print("0");
}
lcd.print(t2[2]);
if (t2[3]<10)
{
  lcd.print("0");
}
lcd.print(t2[3]);
lcd.print("h");
delay(dd);
lcd.setCursor(0,1);
lcd.print("          ");
lcd.setCursor(0,1);
lcd.print("Freq: ");
switch(t2[4])
{
case 1:
  lcd.print("Sundays");
  break;
case 2:
  lcd.print("Mondays");
  break;
case 3:
  lcd.print("Tuesdays");
  break;
case 4:
  lcd.print("Wed'days");
  break;
case 5:
  lcd.print("Thursdays");
  break;
case 6:
  lcd.print("Fridays");
  break;
case 7:
  lcd.print("Saturdays");
  break;
case 8:
  lcd.print("Daily");
  break;
case 9:
  lcd.print("Weekends");
  break;
case 10:
  lcd.print("Weekdays");
  break;
}
delay(dd);
}

void timeronoff()
// allows user to turn timers on or off
{
  delay(300); // for debounce
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Turn knob slowly");
  lcd.setCursor(0,1);
  lcd.print("to select option");
  while (digitalRead(8)==LOW)
  {
    if (readdial(2,1)==0) {
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print(" Timer 1: Off ");
    }
    else if (readdial(2,1)==1) {
      lcd.clear();
    }
  }
}
```

```

        lcd.setCursor(0,0);
        lcd.print(" Timer 1: On      ");
    }
    delay(100); // stop screen flicker
}
switch(readdial(2,1))
{
case 0:
    t1[5]=0;
    break;
case 1:
    t1[5]=1;
    break;
}
delay(300); // for debounce
while (digitalRead(8)==LOW)
{
    if (readdial(2,1)==0) {
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print(" Timer 2: Off      ");
    }
    else if (readdial(2,1)==1) {
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print(" Timer 2: On      ");
    }
    delay(100); // stop screen flicker
}
switch(readdial(2,1))
{
case 0:
    t2[5]=0;
    break;
case 1:
    t2[5]=1;
    break;
}
}

void settimeday()
// allows user to set current time and day of week, e.g. 1234h, Sunday
{
    msethours();
    delay(300); // for debounce
    msetminutes();
    delay(300); // for debounce
    msetdayofweek();
    delay(300); // for debounce
}

void msethours()
// allows user to select hour value
{
    byte second, minute, hour, dayOfWeek, dayOfMonth, month, year;
    delay(300); // for debounce
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Select hour:0~23");
    while (digitalRead(8)==LOW)
    {
        lcd.setCursor(8,1);
        lcd.print("      ");
        lcd.setCursor(8,1);
        lcd.print(readdial(24,1));
        delay(100);
    }
    getDateDs1307(&second, &minute, &hour, &dayOfWeek, &dayOfMonth, &month, &year);
    hour=readdial(24,1);
    setDateDs1307(second, minute, hour, dayOfWeek, dayOfMonth, month, year);
}

```

```
void msetminutes()
// allows user to select minute value
{
  byte second, minute, hour, dayOfWeek, dayOfMonth, month, year;
  delay(300); // for debounce
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Set minute:0~59");
  while (digitalRead(8)==LOW)
  {
    lcd.setCursor(8,1);
    lcd.print(" ");
    lcd.setCursor(8,1);
    lcd.print(readdial(60,1));
    delay(100);
  }
  getDateDs1307(&second, &minute, &hour, &dayOfWeek, &dayOfMonth, &month, &year);
  minute=readdial(60,1);
  setDateDs1307(second, minute, hour, dayOfWeek, dayOfMonth, month, year);
}

void msetdayofweek()
// allows user to set day of week
{
  byte second, minute, hour, dayOfWeek, dayOfMonth, month, year;
  delay(300); // for debounce
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print(" Set day of week");
  while (digitalRead(8)==LOW)
  {
    lcd.setCursor(0,1);
    lcd.print(" ");
    lcd.setCursor(0,1);
    switch(readdial(7,1))
    {
      case 1:
        lcd.print("Sunday");
        break;
      case 2:
        lcd.print("Monday");
        break;
      case 3:
        lcd.print("Tuesday");
        break;
      case 4:
        lcd.print("Wednesday");
        break;
      case 5:
        lcd.print("Thursday");
        break;
      case 6:
        lcd.print("Friday");
        break;
      case 7:
        lcd.print("Saturday");
        break;
    }
    delay(100);
  }
  getDateDs1307(&second, &minute, &hour, &dayOfWeek, &dayOfMonth, &month, &year);
  dayOfWeek=readdial(7,1);
  setDateDs1307(second, minute, hour, dayOfWeek, dayOfMonth, month, year);
}

void msetalarm1()
// allows user to select timer one on/off times and calendar options
{
  // set on hour
  delay(300); // for debounce
  lcd.clear();
```

```
lcd.setCursor(0,0);
lcd.print("T1 On hour: 0~23");
while (digitalRead(8)==LOW)
{
  lcd.setCursor(8,1);
  lcd.print(" ");
  lcd.setCursor(8,1);
  lcd.print(readdial(24,1));
  delay(100);
}
t1[0]=readdial(24,1);
// set on minute
delay(300); // for debounce
lcd.clear();
lcd.setCursor(0,0);
lcd.print("T1 On mins: 0~59");
while (digitalRead(8)==LOW)
{
  lcd.setCursor(8,1);
  lcd.print(" ");
  lcd.setCursor(8,1);
  lcd.print(readdial(60,1));
  delay(100);
}
t1[1]=readdial(60,1);
// set off hour
delay(300); // for debounce
lcd.clear();
lcd.setCursor(0,0);
lcd.print("T1 Off hour:0~23");
while (digitalRead(8)==LOW)
{
  lcd.setCursor(8,1);
  lcd.print(" ");
  lcd.setCursor(8,1);
  lcd.print(readdial(24,1));
  delay(100);
}
t1[2]=readdial(24,1);
// set off minute
delay(300); // for debounce
lcd.clear();
lcd.setCursor(0,0);
lcd.print("T1 Off min: 0~59");
while (digitalRead(8)==LOW)
{
  lcd.setCursor(8,1);
  lcd.print(" ");
  lcd.setCursor(8,1);
  lcd.print(readdial(60,1));
  delay(100);
}
t1[3]=readdial(60,1);
// set calendar options
// daily?
delay(300); // for debounce
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Daily? ");
while (digitalRead(8)==LOW)
{
  lcd.setCursor(1,1);
  lcd.print(" ");
  lcd.setCursor(1,1);
  if (readdial(2,1)==0)
  {
    lcd.print("Yes");
  }
  else if (readdial(2,1)==1)
  {
    lcd.print("No");
  }
}
```

```
    delay(100);
}
if (readdial(2,1)==0)
{
    t1[4]=8;
}
delay(300); // for debounce
// weekend only?
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Weekend only? ");
while (digitalRead(8)==LOW)
{
    lcd.setCursor(1,1);
    lcd.print("      ");
    lcd.setCursor(1,1);
    if (readdial(2,1)==0)
    {
        lcd.print("Yes");
    }
    else if (readdial(2,1)==1)
    {
        lcd.print("No");
    }
    delay(100);
}
if (readdial(2,1)==0)
{
    t1[4]=9;
}
delay(300); // for debounce
// weekday only?
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Weekday only? ");
while (digitalRead(8)==LOW)
{
    lcd.setCursor(1,1);
    lcd.print("      ");
    lcd.setCursor(1,1);
    if (readdial(2,1)==0)
    {
        lcd.print("Yes");
    }
    else if (readdial(2,1)==1)
    {
        lcd.print("No");
    }
    delay(100);
}
if (readdial(2,1)==0)
{
    t1[4]=10;
}
delay(300);
// just a day of the week?
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Single day only?");
while (digitalRead(8)==LOW)
{
    lcd.setCursor(1,1);
    lcd.print("      ");
    lcd.setCursor(1,1);
    if (readdial(2,1)==0)
    {
        lcd.print("Yes");
    }
    else if (readdial(2,1)==1)
    {
        lcd.print("No");
    }
}
```

```
    delay(100);
}
if (readdial(2,1)==0)
{
    delay(300); // for debounce
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" Set day of week");
    while (digitalRead(8)==LOW)
    {
        lcd.setCursor(0,1);
        lcd.print("                ");
        lcd.setCursor(0,1);
        switch(readdial(7,1))
        {
            case 1:
                lcd.print("Sunday");
                break;
            case 2:
                lcd.print("Monday");
                break;
            case 3:
                lcd.print("Tuesday");
                break;
            case 4:
                lcd.print("Wednesday");
                break;
            case 5:
                lcd.print("Thursday");
                break;
            case 6:
                lcd.print("Friday");
                break;
            case 7:
                lcd.print("Saturday");
                break;
        }
        delay(100);
        t1[4]=readdial(7,1);
    }
}
}

void msetalarm2()
// allows user to select timer two on/off times and calendar options
{
    // set on hour
    delay(300); // for debounce
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("T2 On hour: 0~23");
    while (digitalRead(8)==LOW)
    {
        lcd.setCursor(8,1);
        lcd.print("    ");
        lcd.setCursor(8,1);
        lcd.print(readdial(24,1));
        delay(100);
    }
    t2[0]=readdial(24,1);
    // set on minute
    delay(300); // for debounce
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("T2 On mins: 0~59");
    while (digitalRead(8)==LOW)
    {
        lcd.setCursor(8,1);
        lcd.print("    ");
        lcd.setCursor(8,1);
        lcd.print(readdial(60,1));
        delay(100);
    }
}
```

```
}
t2[1]=readdial(60,1);
// set off hour
delay(300); // for debounce
lcd.clear();
lcd.setCursor(0,0);
lcd.print("T2 Off hour:0~23");
while (digitalRead(8)==LOW)
{
  lcd.setCursor(8,1);
  lcd.print(" ");
  lcd.setCursor(8,1);
  lcd.print(readdial(24,1));
  delay(100);
}
t2[2]=readdial(24,1);
// set off minute
delay(300); // for debounce
lcd.clear();
lcd.setCursor(0,0);
lcd.print("T2 Off min: 0~59");
while (digitalRead(8)==LOW)
{
  lcd.setCursor(8,1);
  lcd.print(" ");
  lcd.setCursor(8,1);
  lcd.print(readdial(60,1));
  delay(100);
}
t2[3]=readdial(60,1);
// set calendar options
// daily?
delay(300); // for debounce
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Daily? ");
while (digitalRead(8)==LOW)
{
  lcd.setCursor(1,1);
  lcd.print(" ");
  lcd.setCursor(1,1);
  if (readdial(2,1)==0)
  {
    lcd.print("Yes");
  }
  else if (readdial(2,1)==1)
  {
    lcd.print("No");
  }
  delay(100);
}
if (readdial(2,1)==0)
{
  t2[4]=8;
}
delay(300); // for debounce
// weekend only?
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Weekend only? ");
while (digitalRead(8)==LOW)
{
  lcd.setCursor(1,1);
  lcd.print(" ");
  lcd.setCursor(1,1);
  if (readdial(2,1)==0)
  {
    lcd.print("Yes");
  }
  else if (readdial(2,1)==1)
  {
    lcd.print("No");
  }
}
```

```
    }
    delay(100);
}
if (readdial(2,1)==0)
{
    t2[4]=9;
}
delay(300); // for debounce
// weekday only?
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Weekday only? ");
while (digitalRead(8)==LOW)
{
    lcd.setCursor(1,1);
    lcd.print("      ");
    lcd.setCursor(1,1);
    if (readdial(2,1)==0)
    {
        lcd.print("Yes");
    }
    else if (readdial(2,1)==1)
    {
        lcd.print("No");
    }
    delay(100);
}
if (readdial(2,1)==0)
{
    t2[4]=10;
}
delay(300);
// just a day of the week?
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Single day only?");
while (digitalRead(8)==LOW)
{
    lcd.setCursor(1,1);
    lcd.print("      ");
    lcd.setCursor(1,1);
    if (readdial(2,1)==0)
    {
        lcd.print("Yes");
    }
    else if (readdial(2,1)==1)
    {
        lcd.print("No");
    }
    delay(100);
}
if (readdial(2,1)==0)
{
    delay(300); // for debounce
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" Set day of week");
    while (digitalRead(8)==LOW)
    {
        lcd.setCursor(0,1);
        lcd.print("          ");
        lcd.setCursor(0,1);
        switch(readdial(7,1))
        {
            case 1:
                lcd.print("Sunday");
                break;
            case 2:
                lcd.print("Monday");
                break;
            case 3:
                lcd.print("Tuesday");
```

```
        break;
    case 4:
        lcd.print("Wednesday");
        break;
    case 5:
        lcd.print("Thursday");
        break;
    case 6:
        lcd.print("Friday");
        break;
    case 7:
        lcd.print("Saturday");
        break;
    }
    delay(100);
    t2[4]=readdial(7,1);
}
}
}

void checktimer1()
// checks to see if need to turn timer 1 on or off
{
    byte second, minute, hour, dayOfWeek, dayOfMonth, month, year;
    getDateDs1307(&second, &minute, &hour, &dayOfWeek, &dayOfMonth, &month, &year);
    // if daily check on or off
    if (t1[4]==8)
    {
        if (hour==t1[0] && minute==t1[1])
        {
            digitalWrite(9, HIGH);
            t1[6]=1;
        }
        if (hour==t1[2] && minute==t1[3])
        {
            digitalWrite(9, LOW);
            t1[6]=0;
        }
    }
    // if weekdays check on or off
    if (t1[4]==10)
    {
        if (dayOfWeek>1 && dayOfWeek<7) // if it's between Monday and Friday inclusive
        {
            if (hour==t1[0] && minute==t1[1])
            {
                digitalWrite(9, HIGH);
                t1[6]=1;
            }
            if (hour==t1[2] && minute==t1[3])
            {
                digitalWrite(9, LOW);
                t1[6]=0;
            }
        }
    }
    // if weekend check on or off
    if (t1[4]==9)
    {
        if (dayOfWeek==1 || dayOfWeek==7) // it is Sunday or Saturday
        {
            if (hour==t1[0] && minute==t1[1])
            {
                digitalWrite(9, HIGH);
                t1[6]=1;
            }
            if (hour==t1[2] && minute==t1[3])
            {
                digitalWrite(9, LOW);
                t1[6]=0;
            }
        }
    }
}
```

```
}
// if only one day per week
if (t1[4]<8)
{
  if (t1[4]==dayOfWeek)
  {
    if (hour==t1[0] && minute==t1[1])
    {
      digitalWrite(9, HIGH);
      t1[6]=1;
    }
    if (hour==t1[2] && minute==t1[3])
    {
      digitalWrite(9, LOW);
      t1[6]=0;
    }
  }
}
}

void checktimer2()
// checks to see if need to turn timer 2 on or off
{
  byte second, minute, hour, dayOfWeek, dayOfMonth, month, year;
  getDateDs1307(&second, &minute, &hour, &dayOfWeek, &dayOfMonth, &month, &year);
  // if daily check on or off
  if (t2[4]==8)
  {
    if (hour==t2[0] && minute==t2[1])
    {
      digitalWrite(10, HIGH);
      t2[6]=1;
    }
    if (hour==t2[2] && minute==t2[3])
    {
      digitalWrite(10, LOW);
      t2[6]=0;
    }
  }
  // if weekdays check on or off
  if (t2[4]==10)
  {
    if (dayOfWeek>1 && dayOfWeek<7) // if it's between Monday and Friday inclusive
    {
      if (hour==t2[0] && minute==t2[1])
      {
        digitalWrite(10, HIGH);
        t2[6]=1;
      }
      if (hour==t2[2] && minute==t2[3])
      {
        digitalWrite(10, LOW);
        t2[6]=0;
      }
    }
  }
  // if weekend check on or off
  if (t2[4]==9)
  {
    if (dayOfWeek==1 || dayOfWeek==7) // it is Sunday or Saturday
    {
      if (hour==t2[0] && minute==t2[1])
      {
        digitalWrite(10, HIGH);
        t2[6]=1;
      }
      if (hour==t1[2] && minute==t1[3])
      {
        digitalWrite(10, LOW);
        t2[6]=0;
      }
    }
  }
}
```

```
    }
  }
  // if only one day per week
  if (t2[4]<8)
  {
    if (t2[4]=dayOfWeek)
    {
      if (hour==t2[0] && minute==t2[1])
      {
        digitalWrite(10, HIGH);
        t2[6]=1;
      }
      if (hour==t2[2] && minute==t2[3])
      {
        digitalWrite(10, LOW);
        t2[6]=0;
      }
    }
  }
}

void setup()
{
  byte second, minute, hour, dayOfWeek, dayOfMonth, month, year;
  Wire.begin();
  second = 0;
  minute = 23;
  hour = 23;
  dayOfWeek = 4;
  dayOfMonth = 19;
  month = 5;
  year = 10;
  // setDateDs1307(second, minute, hour, dayOfWeek, dayOfMonth, month, year); // only do this once on
  first use
  lcd.begin(16, 2); // tells Arduino the LCD dimensions
  lcd.setCursor(0,0);
  lcd.print("tronixstuff.com"); // print text and move cursor to start of next line
  lcd.setCursor(0,1);
  lcd.print("* Exercise 10.1 ");
  delay(5000);
  lcd.clear(); // clear LCD screen
  lcd.setCursor(0,0);
  lcd.print("* Please check *"); // print text and move cursor to start of next line
  lcd.setCursor(0,1);
  lcd.print(" * timer data * ");
  delay(2000);
  lcd.clear(); // clear LCD screen
  pinMode(8, INPUT); // for button
  pinMode(9, OUTPUT); // to circuit 1
  pinMode(10, OUTPUT); // to circuit 2
}

void loop()
{
  byte second, minute, hour, dayOfWeek, dayOfMonth, month, year;
  getDateDs1307(&second, &minute, &hour, &dayOfWeek, &dayOfMonth, &month, &year);
  lcd.clear(); // clear LCD screen
  lcd.setCursor(0,0);
  lcd.print("Now: ");
  if (hour<10)
  {
    lcd.print("0");
  }
  lcd.print(hour, DEC);
  if (minute<10)
  {
    lcd.print("0");
  }
  lcd.print(minute, DEC);
  lcd.print("h ");
  switch(dayOfWeek){
  case 1:
```

```
    lcd.print("Sun");
    break;
case 2:
    lcd.print("Mon");
    break;
case 3:
    lcd.print("Tue");
    break;
case 4:
    lcd.print("Wed");
    break;
case 5:
    lcd.print("Thu");
    break;
case 6:
    lcd.print("Fri");
    break;
case 7:
    lcd.print("Sat");
    break;
}
lcd.setCursor(0,1);
lcd.print("Status 1[");
if (t1[6]==0)
{
    lcd.print("_] 2[");
}
else if (t1[6]==1)
{
    lcd.print("!] 2[");
}

if (t2[6]==0)
{
    lcd.print("_]");
}
else if (t2[6]==1)
{
    lcd.print("!]");
}

if (digitalRead(8)==HIGH)
    // has the user pressed the button? If so, display the menu
    {
        delay(200); // for debounce
        displaymenu();
    }

// check for on/off
checktimer1();
checktimer2();
delay(200); // to stop screen flicker
}
```