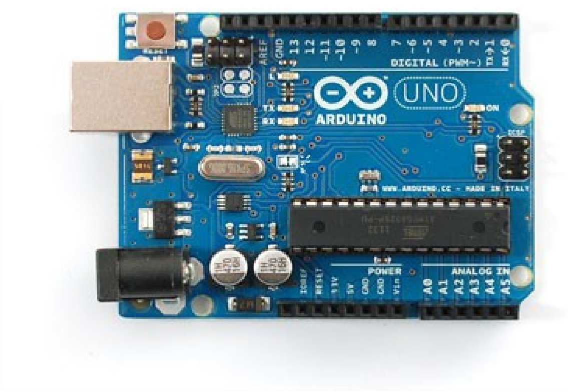
Using an Arduino Uno with a Temperature Sensor and a 16x2 LCD: Time-Lapse Temperature Readings with High and Low Readings

by

Dave Mawdsley, member, DACS Linux SIG

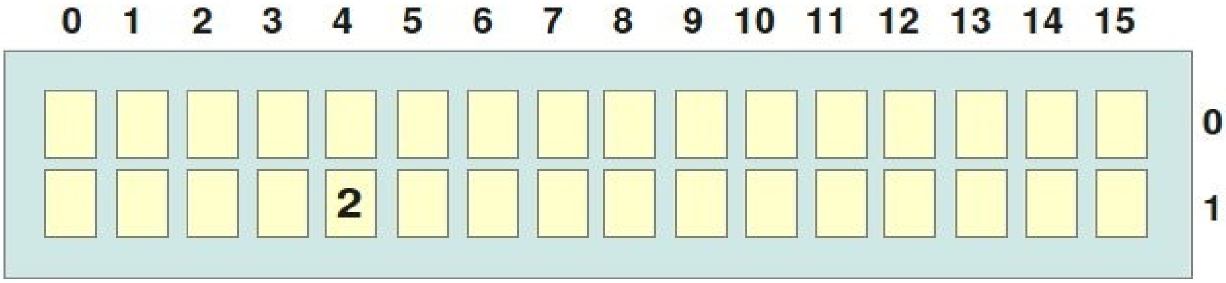
February 15, 2012

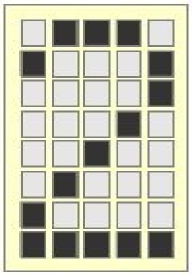


Parts List:

Arduino Uno	Amazon.com	$ 26.45
Standard LCD 16x2	Adafruit.com	$ 13.82
TMP36 (Analog Temperature Sensor)	Adafruit.com	approx. $ 2.00

The Geometry of the 16x2 LCD





```
byte two = {
   B01110,
   B10001,
   B00001,
   B00010,
   B01000,
   B10000,
   B11111
};
```

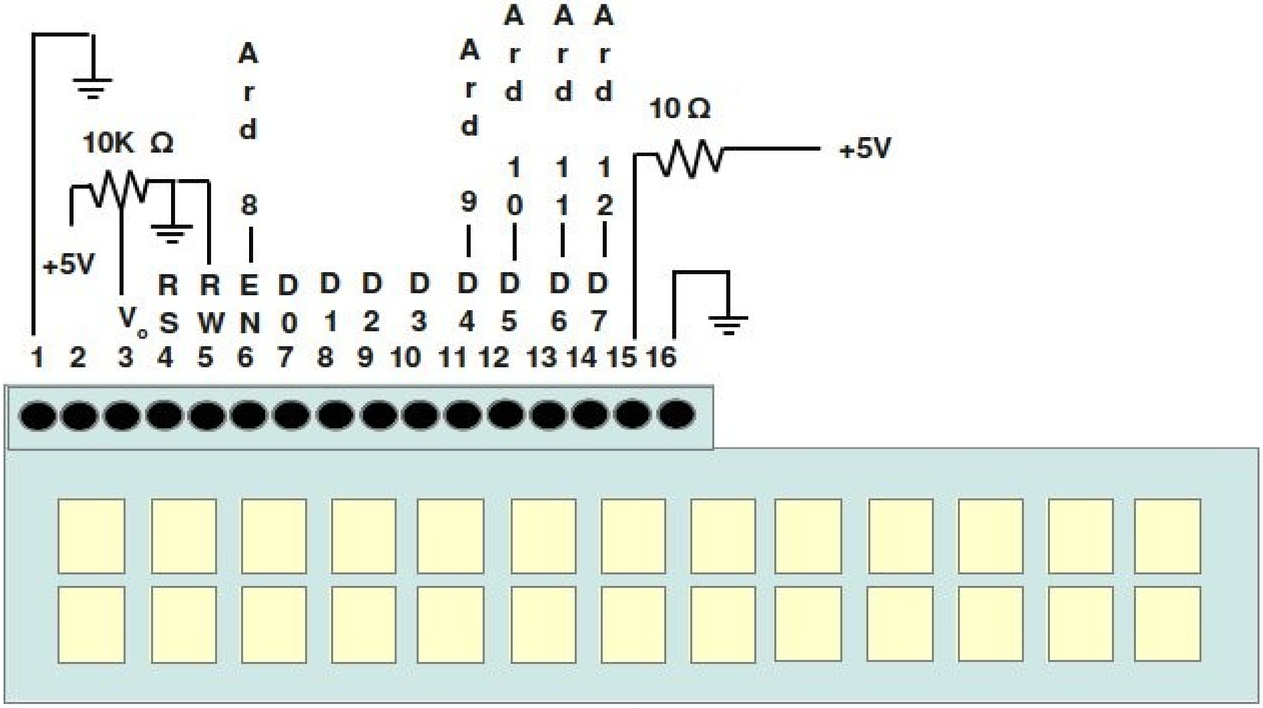
Include the following inside the braces:

```
void setup() {
...

lcd.createChar(1, two);
...
}
```

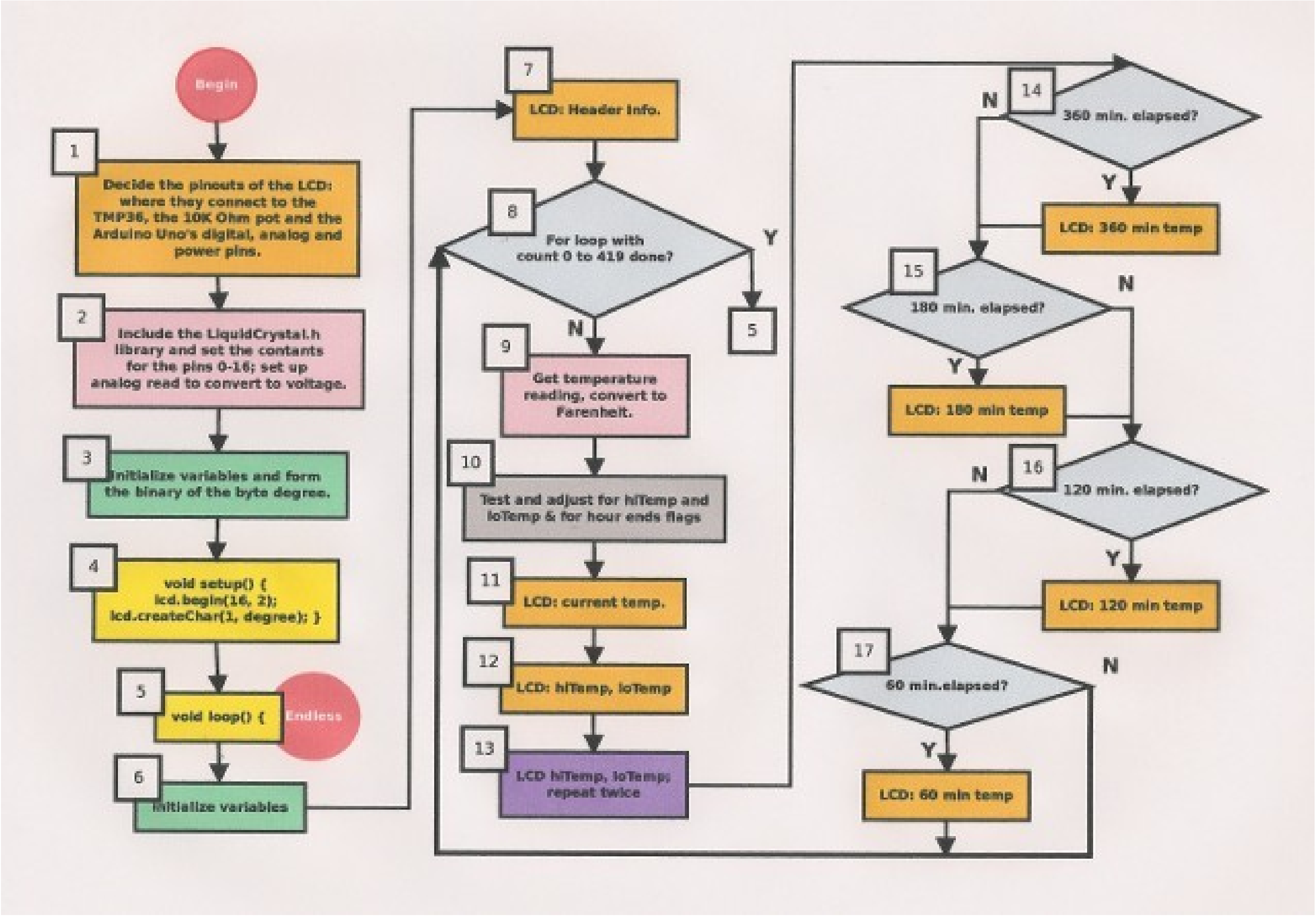
Include the following inside the braces:

```
void loop() {
...

lcd.setCursor(4, 1);
lcd.write(1);
...
}
```

Connections Between the LCD and the Arduino Uno



The Flowchart of the Program



Begin
1
Decide the pinouts of the LCD: where they connect to the TMP36, the 10K Ohm pot and the Arduino Uno's digital, analog and power pins.
2
Include the LiquidCrystal.h library and set the contants for the pins 0-16; set up analog read to convert to voltage.
3
Initialize variables and form the binary of the byte degree.
4
void setup() { lcd.begin(16, 2); lcd.createChar(1, degree); }
5
void loop() {
Endless
6
Initialize variables
7
LCD: Header Info.
8
For loop with count 0 to 419 done?
Y
5
N
9
Get temperature reading, convert to Farenheit.
10
Test and adjust for hiTemp and loTemp & for hour ends flags
11
LCD: current temp.
12
LCD: hiTemp, loTemp
13
LCD hiTemp, loTemp; repeat twice
14
N
360 min. elapsed?
Y
LCD: 360 min temp
15
180 min. elapsed?
N
Y
LCD: 180 min temp
16
N
120 min. elapsed?
Y
LCD: 120 min temp
17
60 min.elapsed?
N
Y
LCD: 60 min temp

The Program Code

```
/*
  6 Hour Temperature Sequence with high and low using a 16x2 LCD as of 02/13/12

  Pinouts of the Standard LCD 16x2 TC1602A-01T:
    16:    GND
    15:    in series with 10 Ohm (LCD protection) and then to +5V (Backlight)
          (Resistor for voltage drop:  R = (5-4.1)/0.100 ohms = 9 ohms.  Use 10 Ohms.)
    14:    D7 to Arduino Digital 12
    13:    D6 to Arduino Digital 11
    12:    D5 to Arduino Digital 10
    11:    D4 to Arduino Digital 9
    10-7:  No Connections
    6:     EN (enable) to Arduino Digital 8
    5:     RW (read=1/write=0) to GND via wiring ... stay in write mode.
    4:     RS (register select) to Arduino Digital 7
```

```
   3:     Vo to 10K Ohm pot Center  (Digit Contrast Vout)
   2:     +5V and to 10K Ohm pot Left
   1:     GND and to 10K Ohm pot Right

  Pinouts of the TMP36 Precision Temperature Sensor: (flat side of sensor facing away)
   right   pin to +5V
   center  Vo to Arduino Analog 0
   left    pin to GND

  Pinouts of the 10K Ohm pot for LCD backlight contrast
   left    pin to +5V
   center  pin to LCD Vo (pin 3)
   right   pin to GND
*/

#include <LiquidCrystal.h>          // include the LiquidCrystal library

const int RS = 7;                   // LCD RS uses Arduino Digital 7
const int EN = 8;                   // LCD EN uses Arduino Digital 8
const int D4 = 9;                   // LCD D4 uses Arduino Digital 9
const int D5 = 10;                  // LCD D5 uses Arduino Digital 10
const int D6 = 11;                  // LCD D6 uses Arduino Digital 11
const int D7 = 12;                  // LCD D7 uses Arduino Digital 12

LiquidCrystal lcd(RS, EN, D4, D5, D6, D7);  // initialize LiquidCrystal lcd(RS, Enable, D4, D5, D6, D7)

int temperaturePin = 0;    // Resolution: 10mV / degree celsius with 500mV offset (for negative temps)
float temp0;                        // initialize temp0
float temp60;                       // initialize temp60
float temp120;                      // initialize temp120
float temp180;                      // initialize temp180
float temp360;                      // initialize temp360

float getVoltage(int pin) {         // getVoltage() – returns the voltage on the analog input
   return (analogRead(pin) * .004887586);  // converting from a 0 to 1023 digital range to 0 to 5 volts
}

byte degree[8] = {                  // shape of the degree symbol
   B00110,
   B01001,
   B01001,
   B00110,
   B00000,
   B00000,
   B00000,
   B00000
};

void setup() {
   lcd.begin(16, 2);                      // set up the LCD's columns and rows of the LCD
   lcd.createChar(1, degree);             // create degree symbol from the binary
}

void loop() {
   int flag60 = 0;                        // intialize flags
   int flag120 = 0;
   int flag180 = 0;
   int flag360 = 0;
   float loTemp = 100;                    // initialize loTemp, hiTemp
   float hiTemp = 0;

   lcd.clear();                           // clear the display
   lcd.setCursor(0, 0);                   // set cursor to column 0, line 0 (1st row)
   lcd.print("MadMod Computing");             // print on the 1st line of the LCD
   lcd.print("6hr Temperatures");           // print on the 2nd line of the LCD
   delay(5000);                           // delay 5 sec.

   for (int count = 0; count <= 419; count++) {        // loop 420 times (6 hr. 59 min.)
       temp0 = getVoltage(temperaturePin);     // get voltage reading from the TMP36 start temp.
       temp0 = (((temp0 - 0.5) * 100) * 1.8) + 32;      // Each 10mV is per degree Celsius
                                                     // with 500mV offset. Farenheit:  F = 1.8 * C + 32
       if (temp0 < loTemp)
           loTemp = temp0;

       if (temp0 > hiTemp)
           hiTemp = temp0;

       if(count == 60 || count == 120 || count == 180 || count == 360) {
           temp360 = temp180;             // shift variables at the hour choices
           temp180 = temp120;
           temp120 = temp60;
           temp60 = temp0;
       }

       lcd.clear();                       // clear the display
lcd.setCursor(0, 0);             // set cursor to column 0, line 0 (1st row)
lcd.print(" temp ");             // print on the 1st line of the LCD
lcd.print(temp0);                // print the temperature
lcd.write(1);                    // write degree symbol
lcd.setCursor(13, 0);            // set cursor to column 13, line (1st row)
lcd.print(count);                // print minute count of loop.

for (int count1 = 0; count1 <= 1; count1++) {
   lcd.setCursor(0, 1);          // set cursor to column 0, line 1 (2nd row)
   lcd.print(" hi temp ");
   lcd.print(hiTemp);            // print high temperature on the 2nd line
   lcd.write(1);
   delay(6000);                  // delay 6 sec.

   lcd.setCursor(0, 1);
   lcd.print(" lo temp ");
   lcd.print(loTemp);            // print low temperature on the 2nd line
   lcd.write(1);
   delay(6000);
}

lcd.setCursor(0, 1);
lcd.print(" last minute   ");    // print msg on the 2nd line
delay(8000);

if (count >= 60) {
   flag60 = 1;                   // set flag at 60 min.
   lcd.setCursor(0, 1);
   lcd.print("60 min. ");
   lcd.print(temp60);            // print temperature at 60 min.
   lcd.write(1);
   delay(8000);
}

if (count >= 120) {
   flag120 = 1;                  // set flag at 120 min.
   lcd.setCursor(0, 1);
   lcd.print("120 min. ");
   lcd.print(temp120);           // print temperature at 120 min.
   lcd.write(1);
   delay(8000);
}
```

```
        if (count >= 180) {
            flag180 = 1;                        // set flag at 180 min.
            lcd.setCursor(0, 1);
            lcd.print("180 min. ");
            lcd.print(temp180);                 // print temperature at 180 min.
            lcd.write(1);
            delay(8000);
        }

        if (count >= 360) {
            flag360 = 1;                        // set flag at 360 min.
            lcd.setCursor(0, 1);
            lcd.print("360 min. ");
            lcd.print(temp360);                 // print temperature at 360 min.
            lcd.write(1);
            delay(8000);
        }

        if (flag360 == 0 && flag180 == 1 && flag120 == 1 && flag60 == 1)    // at at least 180 min.
            delay(4000);                        // delay 4 sec.
        if (flag360 == 0 && flag180 == 0 && flag120 == 1 && flag60 == 1)    // at at least 120 min.
            delay(12000);                       // delay 12 sec.
        if (flag360 == 0 && flag180 == 0 && flag120 == 0 && flag60 == 1)    // at at least 60 min.
            delay(20000);                       // delay 20 sec.
        if (flag360 == 0 && flag180 == 0 && flag120 == 0 && flag60 == 0)    // at before 60 min.
            delay(28000);                       // delay 28 sec.
    }
}
```