```
/*  Hedgehog heater control
start date:  7 Sept 13
Shattered97

Concept:  Use arduino board to control Power Tail II from makershed
to turn on or off a 110VAC ceramic heater and 5VDC heating pad.
TMP36 tempature sensors are used to test the environment then
either turn on or off the heaters.

RBG LCD Setup (Basic setup from Ardunio.cc)
http://www.arduino.cc/en/Tutorial/LiquidCrystal

 * LCD RS pin to digital pin 12
 * LCD Enable pin to digital pin 11
 * LCD D4 pin to digital pin 5
 * LCD D5 pin to digital pin 4
 * LCD D6 pin to digital pin 3
 * LCD D7 pin to digital pin 2
 * LCD R/W pin to ground
 * 10K resistor:
 * ends to +5V and ground
 * wiper to LCD VO pin (pin 3)
 * PWM digital pin 6 to LCD pin 17 (blue backlight) check lcd pinout
 * PWM digital pin 9 to LCD pin 16  (green backlight)
 * PWM digital pin 10 to LCD pin 18 (red backlight)
 * 5V to LCD pin 15 used to run backlight lightcolor
*/

//Include Files
#include <LiquidCrystal.h>
#define aref_voltage 3.3

//Initialize LCD Pins
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

//Variables
  int x = 0;
 //LCD Variables
```

```
  int green  =  1;//background color for LCD
  int red  =  2;//background color for LCD
  int blue  =  3;//background color for LCD
  int bgBlue = 6;//use PWM to set color
  int bgGreen = 9;//use PWM to set color
  int bgRed = 10;//use PWM to set color
 //TMP36Variables
  int TMP36_1  =  0;//TMP36 sensor 1 Left side of cage
  int TMP36_2  =  1;//TMP36 sensor 2 Right side of cage
  int TMP36_3  =  2;//TMP36 sensor 3 Center of cage
  int TMP36_4  =  3;//TMP36 sensor 4 Rock house temp
  const int arraySize  =  100;//size of array to store data stream fr
  int  readings[arraySize];//array to store readings to smooth readi
  float   left;//Variable to store data for TMP36 1
  float  right;//Variable to store data for TMP36 2
  float  center;//Variable to store data for TMP36 3
  float   averageTemp;//Variable to store average Temp from TMP36 1-3
  float   houseTemp;//Variable to store data for TMP36 4

void setup()
{
  //LCD Setup
  lcd.begin(16, 2); //setup the LCD's number of columns and rows
  lcd.print("Loading");
  pinMode(bgRed, OUTPUT);  //setup digital pin 10 for output
  pinMode(bgBlue, OUTPUT);  //setup digital pin 6 for output
  pinMode(bgGreen, OUTPUT);  //setup digital pin 9 for output
  //TMP36 Setup
  Serial.begin(9600);
  analogReference(EXTERNAL);
  pinMode(TMP36_1, OUTPUT);  //Temp Sensor 1 Left side of cage
  pinMode(TMP36_2, OUTPUT);  //Temp Sensor 2 Right side of cage
  pinMode(TMP36_3, OUTPUT);  //Temp Sensor 3 Center of cage
  pinMode(TMP36_4, OUTPUT);  //Temp Sensor 4 Rock house

  //PowerSwitch tail
  pinMode(8, OUTPUT);
}
```

```
void loop()
{
  //Get readings from Temp sensors average them and convert to F
   left = getReadings(0);//get readings from TMP36 1 smooth and conv
   right = getReadings(1);//get readings from TMP36 2 smooth and cor
   center = getReadings(2);//get readings from TMP36 smooth and conv
   averageTemp = (left + right + center) / 3;//find average of readi
   houseTemp = getReadings(3);//get reading from TMP36 4 smooth and

  //Check temp and determine if powerswitch tail is on or off
  if(averageTemp <= 75) //Turn powerswitch tail on
    {digitalWrite(8, HIGH);    }//Temp drops 75 Turn powerswitch tail
  else if(averageTemp > 78)
    {digitalWrite(8, LOW);    }//Temp reaches 78 Turn powerswitch tai

  //check temp and set background color
  if(averageTemp < 71)//Temp below 71F turn display red
   {   bgLCD(2);   }//Below 71 degrees background red
  else if(averageTemp >= 71)//Temp 71F and above turn display green
   {   bgLCD(1);   }// 71 degrees or above background green
   textLCD(left, center, right);//Send data to display
} //end main loop

//LCD Functions
void bgLCD(int color)
{
  /* use analogWrite(PIN, Value) to set color.  Value is 0 to 255.  Z
  on and 255 is off.  Any mixture of the value will result in a comb
  of the colors.
  PINS used:  Digital 6, 9, 10
  */
  switch (color)
  {
    case   1://Set Background to green
      analogWrite(bgGreen,200); //set to 200 so it is not so bright
      analogWrite(bgRed, 255);
      analogWrite(bgBlue, 255);
```

```
      break;
    case   2://Set Background to red
      analogWrite(bgGreen, 255); //set to 200 so it is not so bright
      analogWrite(bgRed, 200);
      analogWrite(bgBlue, 255);
      break;
    case   3://Set Background to blue
      analogWrite(bgGreen, 255);
      analogWrite(bgRed, 255);
      analogWrite(bgBlue, 200); //set to 200 so it not so bright
      break;
    default:  //debug code - no background color
      analogWrite(bgGreen, 255);
      analogWrite(bgRed, 255);
      analogWrite(bgBlue, 255);
      break;
  }//end switch
}//end bgLCD

//setup LCD display to display data
void textLCD(float  left,float  center,float right)
{
  /* Used to set up input from temp36 sensor to display the current t
  int left is left sensor temp, int center is center sensor temp, int
   sensor  temp.
  */
   float average = (left + right + center)/3;//Average Temp from al

  //round numbers to whole number
  //will not work for negative temp.  Should never reach negative tem
   left =floor(left  +  .5);
   right =floor(right + .5);
   center =floor(center + .5);
   average =floor(average + .5);
   houseTemp =floor(houseTemp + .5);//sensor inside of house

  //setup display
  lcd.clear();
```

```
  lcd.print("Avg:");
  lcd.print(average, 0);
  lcd.setCursor(7,0);
  lcd.print("House:");
  lcd.print(houseTemp,0);
  lcd.setCursor(0,1);
  lcd.print("L:");
  lcd.print(left, 0);
  lcd.setCursor(6,1);
  lcd.print("C:");
  lcd.print(center, 0);
  lcd.setCursor(12,1);
  lcd.print("R:");
  lcd.print(right, 0);
}

//TMP36 Functions

float getReadings(int SensorNum)
{
  /*  This is used to smooth the readings out taking an
  average over time (miliseconds).  This will take 100 readings
  store that in an array.  Then add up all the data and divide by
  100 to get an average reading.  This is done several times
  a second to smooth out sensor readings.
  */

  float temp  =  0;//ensure the starting point is zero;
  for (int x=0; x<arraySize; x++)
  {
    readings[x] =analogRead(SensorNum);  //fill the array with data
    temp = temp + readings[x];//find the sum
  }
  float result  =  temp/arraySize;//return the average
  float voltage = (result * 3.3)/1024;//figure the voltage use 3.3v
  float temperatureC = (voltage - 0.5) * 100;//convert to C based f
  float temperatureF = (temperatureC * 9 /5) + 32;//convert C to F
  return  temperatureF;//return temp in F
```

```
}//end     getReadings
```