Making Yoghurt with the Portlet

Or basically using the Portlet to control a defined temperature curve

Summary: This is a brief overview of how to use the Portlet to create a defined temperature profile. In this example, we make Yoghurt. The code for programming the Arduino is pasted in Appendix A

Table of Contents

Overview of Making Yoghurt

The basic steps in making yoghurt are:

1. Heat the milk to 185 degrees F
2. Pastuerize the milk (185 degrees F for 30 minutes)
3. Cool the milk down to 110 degrees F
4. Pitch the yoghurt culture and let it culture around 100 degrees for 7 hours
5. Put your yoghurt in the fridge!

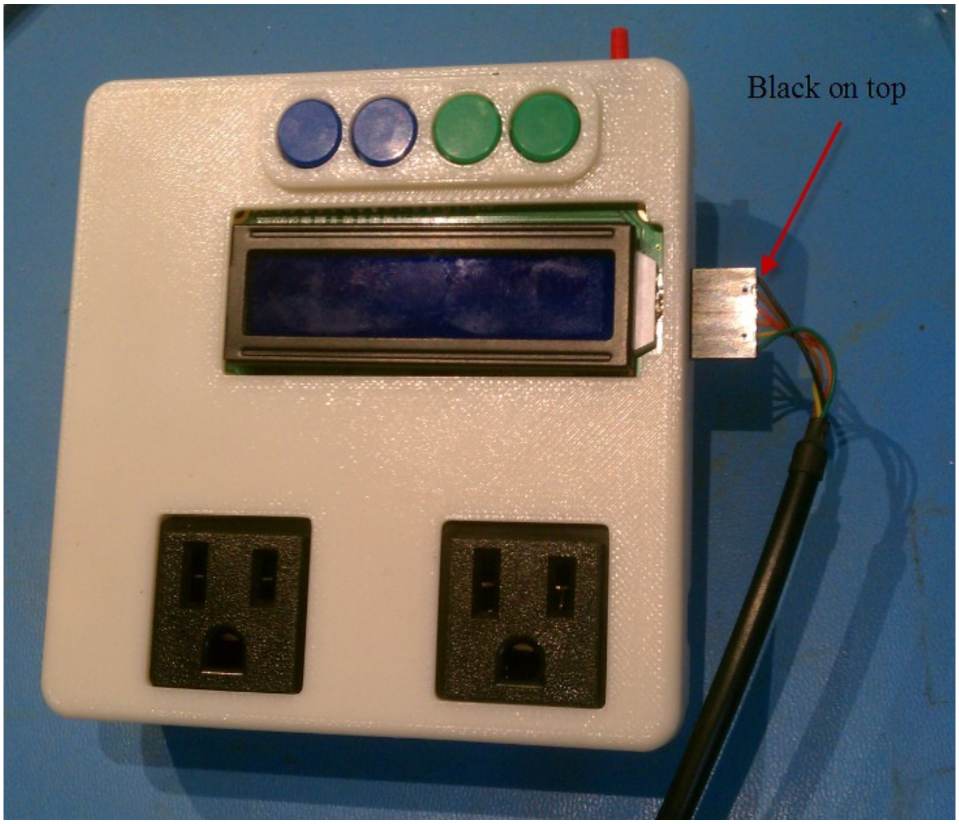
To achieve this we'll use the following

- A crockpot. Standard issue kitchen fare. Located in GoodWill stores all over, and can even be purchased for brand new at very reasonable prices.
- A Portlet. This will be the primary controller for the whole operation.
- Milk. Any milk will do. If you want non-fat yoghurt, you should start with non-fat milk. If you want tasty, creamy yoghurt, it's best to start with whole milk.
- Yoghurt. Pick a brand that you want to try and copy. This will provide your starter culture for all the little bacteria. Use a new cup if possible; you want to minimize the chance of introducing foreign bacteria into your culture. Only a spoonful is necessary.

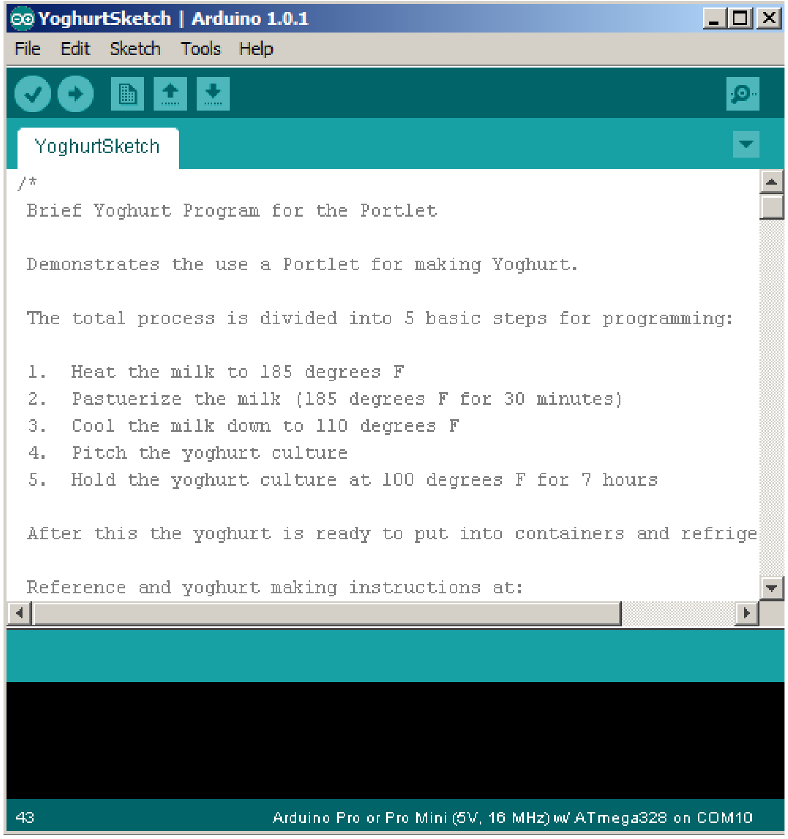
Programming the Portlet

Before you start, it's best to get the Portlet all ready and programmed. The Portlet is based off the popular Arduino platform and uses the Arduino IDE for programming. Here we give a brief example of how to program the Arduino, so if you want to learn more you should head over to the excellent Arduino home page at http://www.arduino.cc/

First off, plug your programming cable into the programming interface of the Portlet. The interface is the six header pins coming off of the Arduino Pro Mini build into the device. There is an access port in the covers on the right side of the Portlet. If you are using the Sparkfun 5V programming cable, the black wire goes on top for the correct orientation.

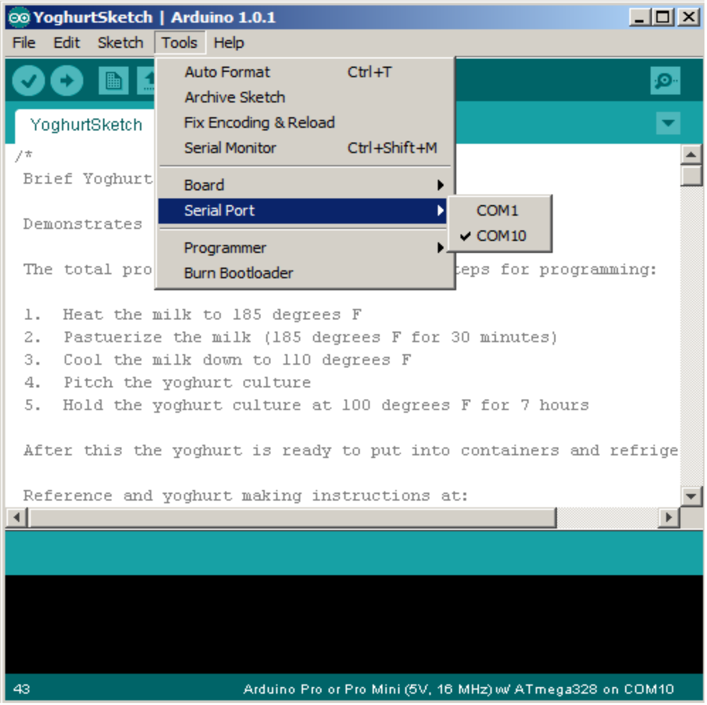


Open up the Arduino IDE and copy and paste the Yoghurt Sketch Program attached into the IDE.



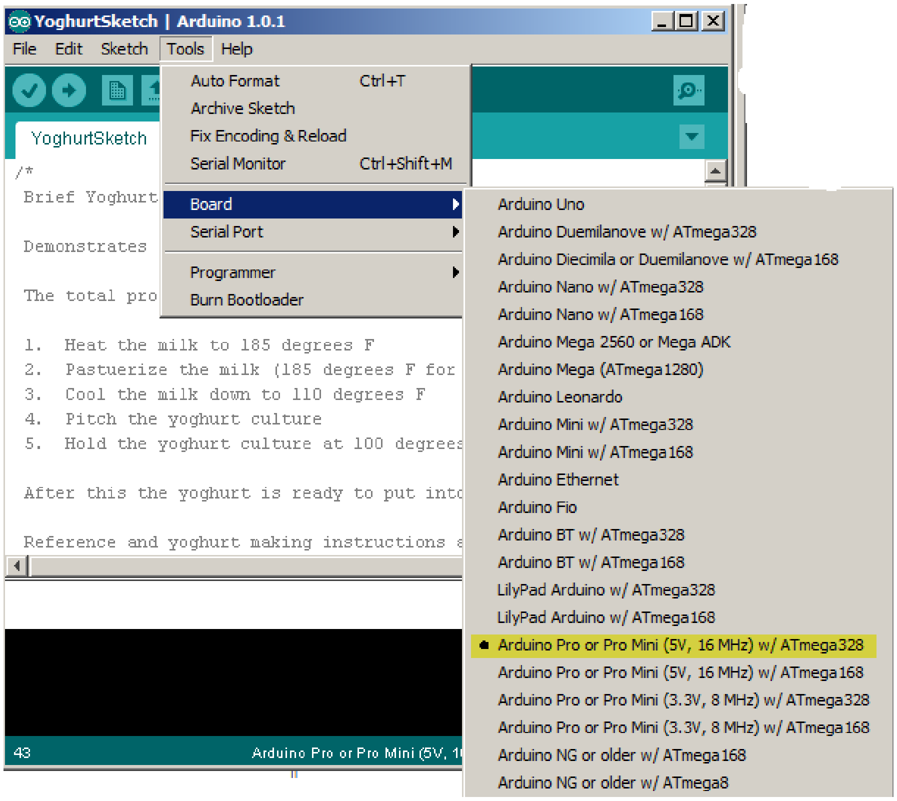
```
/*
 Brief Yoghurt Program for the Portlet

 Demonstrates the use a Portlet for making Yoghurt.

 The total process is divided into 5 basic steps for programming:

 1.  Heat the milk to 185 degrees F
 2.  Pasteurize the milk (185 degrees F for 30 minutes)
 3.  Cool the milk down to 110 degrees F
 4.  Pitch the yoghurt culture
 5.  Hold the yoghurt culture at 100 degrees F for 7 hours

 After this the yoghurt is ready to put into containers and refrige

 Reference and yoghurt making instructions at:
```

Make sure you have the correct serial port selected for your interface and for the board choose the “Arduino Pro or Pro Mini (5V, 16MHz) w/ATmega328”



```
/*
 Brief Yoghurt

 Demonstrates

 The total pro                          eps for programming:

 1.  Heat the milk to 185 degrees F
 2.  Pasteurize the milk (185 degrees F for 30 minutes)
 3.  Cool the milk down to 110 degrees F
 4.  Pitch the yoghurt culture
 5.  Hold the yoghurt culture at 100 degrees F for 7 hours

 After this the yoghurt is ready to put into containers and refrige

 Reference and yoghurt making instructions at:
```

43   Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328 on COM10


YoghurtSketch | Arduino 1.0.1
File
Edit
Sketch
Tools
Help
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Serial Monitor Ctrl+Shift+M
Board
Serial Port
Programmer
Burn Bootloader
YoghurtSketch
Arduino Uno
Arduino Duemilanove w/ ATmega328
Arduino Diecimila or Duemilanove w/ ATmega168
Arduino Nano w/ ATmega328
Arduino Nano w/ ATmega168
Arduino Mega 2560 or Mega ADK
Arduino Mega (ATmega1280)
Arduino Leonardo
Arduino Mini w/ ATmega328
Arduino Mini w/ ATmega168
Arduino Ethernet
Arduino Fio
Arduino BT w/ ATmega328
Arduino BT w/ ATmega168
LilyPad Arduino w/ ATmega328
LilyPad Arduino w/ ATmega168
● Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328
Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168
Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega328
Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega168
Arduino NG or older w/ ATmega168
Arduino NG or older w/ ATmega8

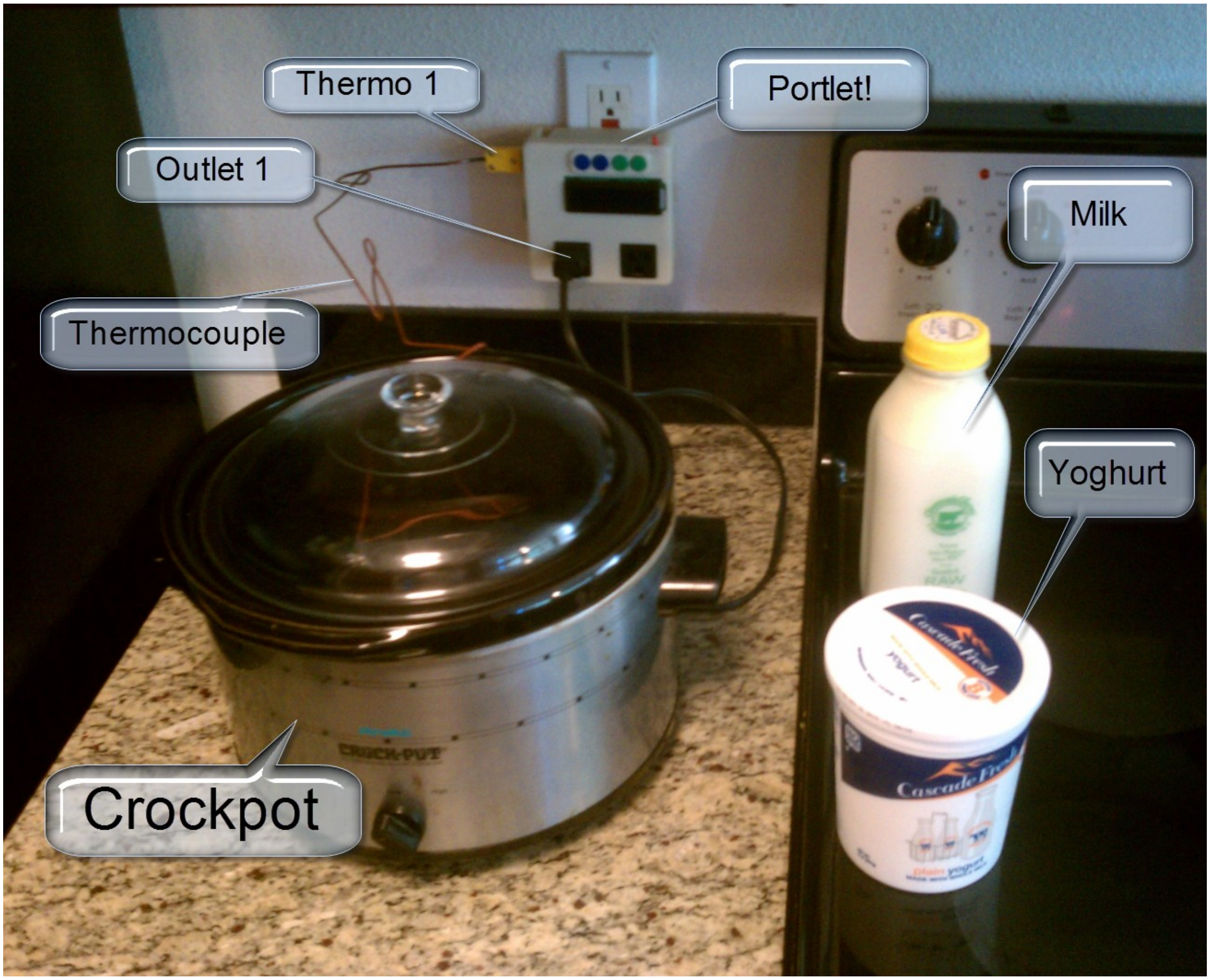
```
/*
 Brief Yoghurt

 Demonstrates

 The total pro

 1.  Heat the milk to 185 degrees F
 2.  Pasteurize the milk (185 degrees F for
 3.  Cool the milk down to 110 degrees F
 4.  Pitch the yoghurt culture
 5.  Hold the yoghurt culture at 100 degrees

 After this the yoghurt is ready to put into

 Reference and yoghurt making instructions a
```

43   Arduino Pro or Pro Mini (5V, 1

The Setup

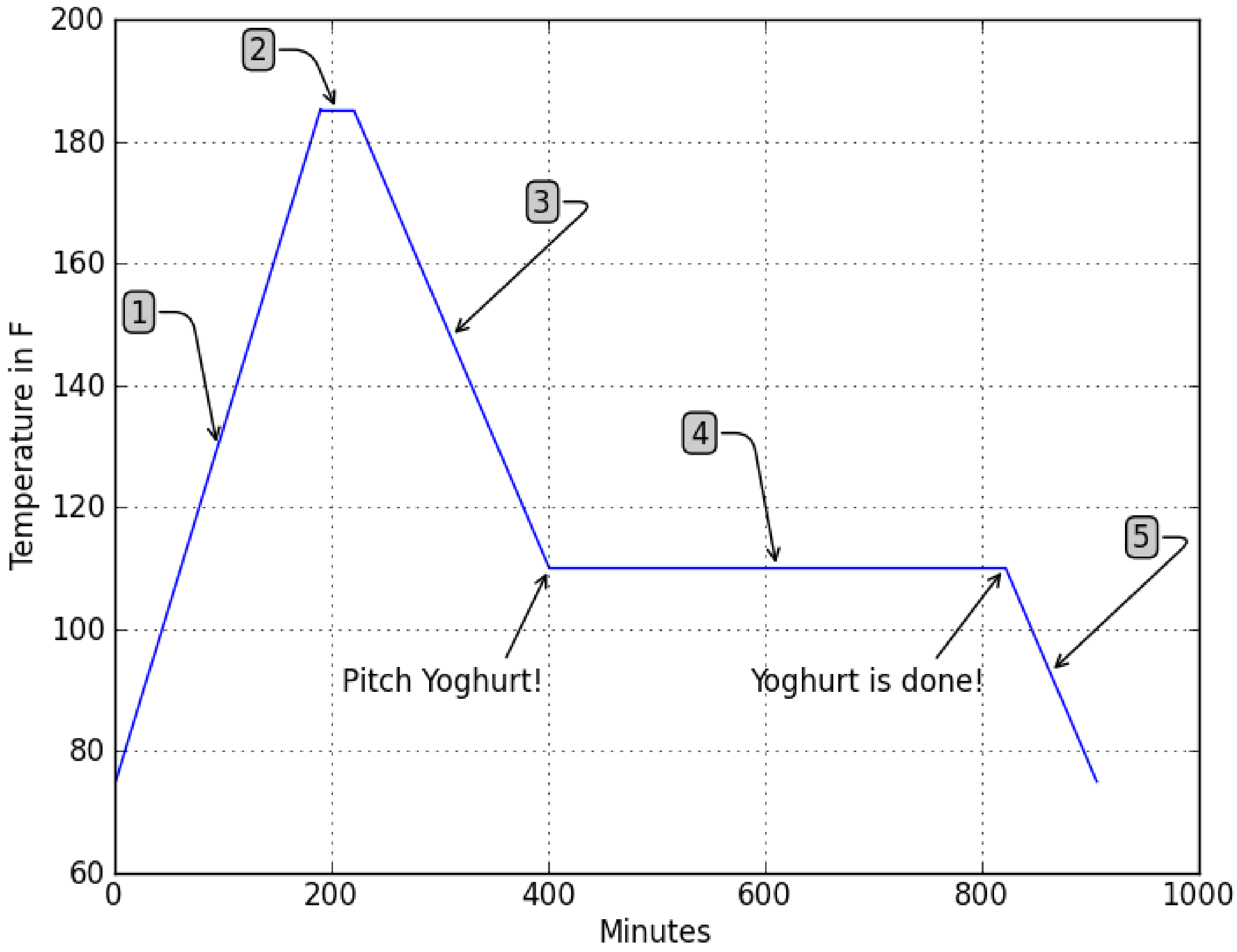
Here you see the crockpot plugged into the Portlet, which in turn is plugged into the wall outlet. The milk (raw Claravale milk is shown) and the yoghurt are on hand. The milk will get poured into the crockpot right away but since you won't be needing the yoghurt for some time, you should keep that in the fridge until ready.

The program supplied below expects that you are using Outlet 1 and Thermo 1. Make sure you have the crockpot and the thermocouple plugged into the right sockets!



Defining the temperature curve

If we are to graph this process, the basic temperature profile looks like:



And a more detailed explanation of the steps would be:

1. Here we are trying to get to 185 as quickly as possible. Depending on your crockpot, this curve may look different; steeper or more shallow. That's okay. We aren't really concerned with how quickly we get there, we just want to spend some time at 185 to make sure the batch is pasteurized and maybe we've broken down the proteins a little bit.
2. Now we spend some time at 185. The code for maintaining the temperature is very simple but adequate for this purpose. Ten minutes at 185 is recommended to make sure pasteurization is complete, but some yoghurt manufacturers will actually hold it there much longer to get a creamier yoghurt. It's a good place to experiment!
3. Now that pasteurization is complete, we want the temperature to go down to about 110 so we can start the culturing process. Some people will take their pot of milk and put it in ice because they don't want to wait around forever. But since the Portlet is automating this whole process, we can just let it turn the crockpot off and wait until the desired temperature has been reached. With the program attached it will notify us that it has arrived there by flashing the backlight on the LCD screen. This indicates it is time to dump a few spoonfuls of yoghurt into the pot and start the culturing process.
4. Culturing bacteria is very simple. You've given it a clean environment without any competition by pasteurizing the batch and it has plenty of nutrients in the mil, so now all it wants is a warm place to do its' thing. So we set the Portlet to maintain a temperature of about 110°F and let it sit for about 7 hours. The Portlet has a timer that counts down the time and notifies you when that time has elapsed. Once that time is over, you have yoghurt!
5. We don't want to let the yoghurt sit and culture forever so at this point we've turned off the heat and are letting the batch cool down to room temperature. This will not stop the culturing process, but the colder temperature will at least slow it down. By now you should have put the yoghurt into the fridge. The Portlet does not have the ability to do that for you (not yet anyways...)

Make Yoghurt!

Now let's get started!

1) Fill the crockpot with milk.

As much milk as you put into the crockpot is how much yoghurt you'll have the end. Since you are probably reading this because you really like yoghurt, go ahead and fill it up!

2) Position everything

The crockpot will be taking up some counter space for a few hours so make sure it's somewhere out of the way. **Make sure the crock pot is turned on to high.** The Portlet will be controlling the temperature by turning the outlet on or off, but it can't turn the crockpot on for you!

Then make sure the end of the thermocouple is sitting in the milk. That's the only way it will be able to get a nice temperature reading.

3) Start!

Follow the prompts on the LCD screen of the Portlet. It will tell you to hit button 1 (the one on the left) to get started.

It will then heat the milk up to 185, displaying the temperature along the way.

Then it sits at 185 for the set period of time, displaying the number of minutes.

Then it cools down 110 displaying the temperature along the way.

4) Pitch the yoghurt!

Once at 110, it will notify you that it is time to "pitch" the yoghurt. This is a fancy way of saying that you should now dump a few spoonfuls of yoghurt into your batch. Then hit the button to tell Portlet you are ready to start timing.

For the culturing period, the Portlet will maintain a nice culturing temperature while displaying the number of minutes.

5) Refrigerate your yoghurt!

After culturing, the yoghurt is ready to eat! But it's not very good warm. So you should probably stick it in the fridge for a while first.

Appendix A: Sample Arduino Code for Making Yoghurt

```
/*
 Brief Yoghurt Program for the Portlet

 Demonstrates the use a Portlet for making Yoghurt.

 The total process is divided into 5 basic steps for programming:

 1. Heat the milk to 185 degrees F
 2. Pastuerize the milk (185 degrees F for 30 minutes)
 3. Cool the milk down to 110 degrees F
 4. Pitch the yoghurt culture
 5. Hold the yoghurt culture at 100 degrees F for 7 hours

 After this the yoghurt is ready to put into containers and refrigerate.

 Reference and yoghurt making instructions at:

 www.makeyourownyogurt.com

 It is expected that Outlet 1 and Thermo 1 are being used

  The LCD circuit:
 * LCD RS pin to digital pin 11
 * LCD Enable pin to digital pin 12
 * LCD D4 pin to digital pin 7
 * LCD D5 pin to digital pin 8
 * LCD D6 pin to digital pin 9
 * LCD D7 pin to digital pin 10
 * LCD R/W pin to ground

 */
```

```
// include the library code for the LCD:
#include <LiquidCrystal.h>

// initialize the library with the numbers of the interface pins for LCD control
LiquidCrystal lcd(11, 12, 7, 8, 9, 10);

// define the switches and outputs

const int button1 = 2;
const int button2 = 3;
//buttonThree and buttonFour are hooked to Analog
const int buttonThree = A2;
const int buttonFour = A3;

const int thermo1 = A0;
const int thermo2 = A1;

const int outlet1 = 6;
const int outlet2 = 5;

//the LCD backlight is used as feedback to the user
const int backlight = 13;


void setup() {
  // set up the LCD's number of columns and rows:
  lcd.begin(16, 2);

  //define buttons as inputs and outlets as outputs
  pinMode(button1, INPUT);
  pinMode(button2, INPUT);

  pinMode(outlet1, OUTPUT);
  pinMode(outlet2, OUTPUT);
  //enable serial for troubleshooting
  Serial.begin(9600);

}



int convert_degrees(int analog_value){
  int temperature = 0;

  //NOTE: IF YOU WANT TO FINE TUNE YOUR TEMPERATURE READOUTS
  //THIS IS THE PLACE TO TO DO IT!
  int offset = 0;

  //the circuit changes 10mV/degree Celsius (hopefully)
  //Analog inputs measure 0 to 5V
  //0 to 5V is a range of 0 to 500 degrees Celsius

  //UNCOMMENT THIS LINE FOR CELSIUS
  //temperature = map(analog_value, 0, 1023, 0, 500) + offset;

  //0 to 500 degrees Celsius is 32 to 932 degrees Farenheit
  //UNCOMMENT THIS LINE FOR FARENHEIT
  temperature = map(analog_value, 0, 1023, 32, 932) + offset;

  return temperature;
}

//turn the outlet on until target temperature is reached
void heat_up(int target_temp){
  //while temp is less than target, keep outlet on
  while(target_temp > convert_degrees(analogRead(thermo1))){
    digitalWrite(outlet1, HIGH);
      //reset LCD and print current operation
    lcd.clear();
    lcd.home();
    lcd.print("Heating to ");
    lcd.print(target_temp);
    //set cursor for printing temp
    lcd.setCursor(0,1);
    //print current temperature
    lcd.print("Temp:");
    //clear the old temp from the screen
    lcd.setCursor(8, 1);
    lcd.print("     ");
    lcd.setCursor(8, 1);
    //print current temp
    lcd.print(convert_degrees(analogRead(thermo1)));
    //pause for a bit so the screen doesn't flicker
    delay(200);
  }
  //turn outlet off before exiting
  digitalWrite(outlet1, LOW);
}

//turn the outlet off until it cools to target temperature
void cool_down(int target_temp){
  //make sure outlet is off
  digitalWrite(outlet1, LOW);
  //wait for temperature to drop below target
  while(target_temp < convert_degrees(analogRead(thermo1))){
      //clear lcd and print current operation
    lcd.clear();
    lcd.home();
    lcd.print("Cooling to: ");
    lcd.print(target_temp);
    lcd.setCursor(0, 1);
    //print current temperature
    lcd.print("Temp:");
    //clear the old temp from the screen
    lcd.setCursor(8, 1);
    lcd.print("     ");
    lcd.setCursor(8, 1);
    //print current temp
    lcd.print(convert_degrees(analogRead(thermo1)));
    //pause for a bit so the screen doesn't flicker
    delay(200);
  }
}

//signalling the user to pitch the yoghurt bacteria
void pitch_yoghurt(){
  //clear lcd and print current operation
  lcd.clear();
  lcd.home();
  lcd.print("Pitch Yoghurt");
  lcd.setCursor(0, 1);
  lcd.print("1 to continue");
  //flash the backlight to get attention
  int time_elapsed = millis();
  while(digitalRead(button1) == LOW){
    if ((millis()-time_elapsed) > 1000){
      digitalWrite(backlight, HIGH);
    }
    if ((millis()-time_elapsed) > 2000){
      digitalWrite(backlight, LOW);
      time_elapsed = millis();
    }
  }
}

//routine to maintain the temperature at specified temp and for duration time
```

```
void maintain_temp(int target_temp, int time_in_min){
  //only check the temperature once per minute
  int num_loops = time_in_min;
  for(int x = 0; x < num_loops; x++){
    if (convert_degrees(analogRead(thermo1)) < target_temp){
      digitalWrite(outlet1, HIGH);
    }
    else {
      digitalWrite(outlet1, LOW);
    }
    //clear lcd and print current operation
    lcd.clear();
    lcd.home();
    lcd.print("Maintain temp");
    lcd.setCursor(0, 1);
    //display number of minutes
    lcd.print("Num min:  ");
    lcd.print(x);
    lcd.print("   ");
    //pause 1 minute each cycle
    delay(60000);
  }
  //turn outlet off before exiting
  digitalWrite(outlet1, LOW);
}



void loop() {
  // prepare LCD
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Press 1  ");
  lcd.setCursor(0, 1);
  lcd.print("to Start  ");

  //do nothing loop until a button is pressed
  while(digitalRead(button1) == LOW);

  //heat milk to 185 degrees
  heat_up(185);
  //pastuerize milk for 30 minutes
  maintain_temp(185, 30);
  //cool the milk down to 100 degrees
  cool_down(100);
  //request that the user pitch the culture
  pitch_yoghurt();
  //culture the bacteria at 100 degrees for 7 hours
  maintain_temp(100, 420);

  //Signal success
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("ALL DONE!");
  lcd.setCursor(0, 1);
  lcd.print("Refrigerate");

  //either press button1 or reset to do it again
  while(digitalRead(button1) == LOW);
}
```

A more detailed explanation of each of the temperature curves is

The idea is to get to the desired temperature as quickly as possible