Designer's Corner

MOST CHEFS WOULD ARGUE THAT BARBECUING IS MORE ART THAN SCIENCE. NEVERTHELESS, JOHN INTRODUCED ATMEGA8-BASED TECHNOLOGY TO THE PROCEDURE. THE RESULT? AN AUTOMATIC TEMPERATURE CONTROLLER THAT ELIMINATES THE NEED FOR CONSTANT VIGILANCE WHEN SLOW COOKING.

Reprinted with permission of
Circuit Cellar
Issue 154
May 2003

Automatic Temperature Controller:

Designing a Data Logger for a Slow Cooker

By: John Moyer

A traditional barbecue typically involves cooking tough cuts of meat (e.g., pork ribs, pork shoulder, and beef brisket) over a wood or charcoal fire at a low temperature (200° to 250°F). The meat is cooked for long periods of time until it's tender (e.g., 5 to 6 hours for ribs and 10 to 20 hours for brisket or pork shoulder). Usually, steel offset cookers, modified 55-gallon drums, and even in-ground pits are used.

Another barbecue tradition requires someone to be present for the 5 to 20 hours that the meat is cooking. This person must add fuel so the fire doesn't go out and regulate the cooking temperature. Although modern technology can't shorten barbecue-cooking times, it can eliminate the need for someone to be present to maintain the cooking temperature.

Ceramic Cookers

Photo 1 shows a different kind of cooker. The cooker's thick ceramic walls do such a good job of insulating that a single load of lump charcoal fuel can maintain barbecue-cooking temperatures for more than 24 hours This neatly solves the problem of having to add fuel during a cooking session, but there's still the issue of temperature regulation. The firebox is at the bottom of the cooker, and the temperature is adjusted using the lower and upper draft openings. Additional air moving through the cooker produces a bigger fire and higher temperature.

After it's set for a given temperature, a ceramic cooker does a remarkable job of maintaining that temperature. However, if internal or external conditions shift too much (e.g., changes in the weather or wind, or ash buildup in the firebox), the draft settings must be adjusted in order to keep the temperature from changing. To have true hands-off temperature control, you need a closed-loop process control system. In such a system, the actual cooker temperature is monitored and used to adjust the airflow to maintain the desired temperature.

Project Goals

The original goal for this project was to implement a simple closed-loop temperature controller using an 8-bit microprocessor. Modeled after a kitchen oven, I wanted the controller to have a knob for setting the desired temperature. A temperature probe would monitor the ceramic cooker's actual temperature, and the controller would adjust the airflow to keep the desired and actual temperatures as close as possible.

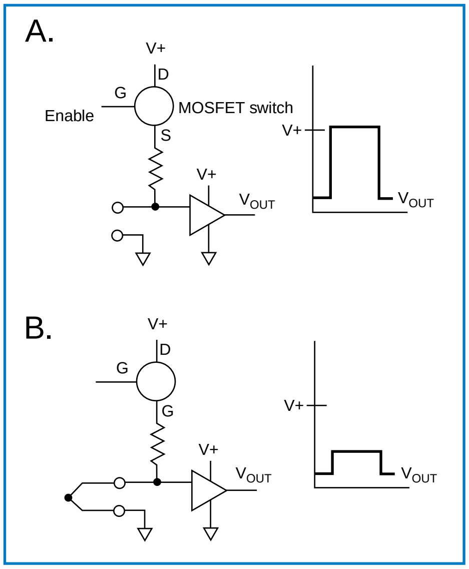

Figure 1a: With no thermocouple connected, a test voltage applied to the op-amp produces a maximum output voltage. b: The low impedance of a connected thermocouple drastically reduces the test voltage, producing a much lower output voltage.

Being microprocessor controlled, I thought it would be easy to add other functionality, such as additional probes to monitor the food temperature, and another control to indicate the desired final food temperature. An alarm could ring when the food reached a desired temperature or if an error condition occurred. An RS-232 port would allow cooker and food temperatures to be logged, either to a local computer or perhaps a web site for remote access. Photo 2 shows the front panel of my final controller implementation.

As for the control system's accuracy, my intention was make it at least as good as a kitchen oven. Temperature tolerances only needed to be within ±5°, because barbequing is more of an art than a science. Temperature is just one factor to be consider when you're trying to determine if the food is ready to be taken off the fire.

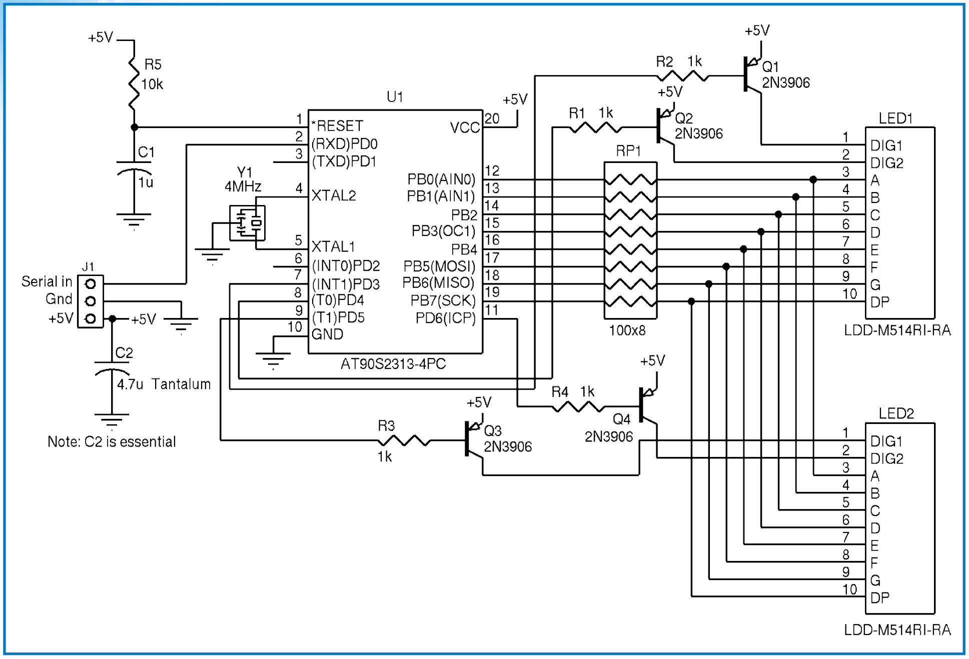

Figure 2: The goal is to build a system that is comparable to or better than a kitchen oven. Atmel's ATmega8 microcontroller is at the heart of the controller.

Temperature Acquisition

Temperatures are measured by taking an analog measurement from a temperature probe. Converting it to a digital value using the micro's A/D converter, and then converting the raw measurement to an actual temperature achieves this.

The temperature probe I used is a thermocouple, which is constructed simply by connecting two dissimilar types of metal wire. When this junction is heated, a voltage proportional to the temperature can be measured at the other end of the wires. Different types of metals have been standardized, and their voltage-versus-temperature properties characterized. This project uses a type-K thermocouple.

Generally, type-K can be used for temperatures up to 2300°F, although the insulating materials used in the low-cost probes limit the upper temperature to approximately 500°F for Teflon and 900° for glass braid. Photo 3 shows typical thermocouple configurations. Although thermocouples are simple, rugged, and inexpensive, they have some properties that must be accommodated.

Thermocouple Properties

First, the output voltage of a type-K thermocouple is only about 20 µV (i.e., 0.00002 V) per 1°F. This must be amplified in order to use the microprocessor's A/D converter, but most amplifiers aren't suitable. General-purpose op-amps typically have input offset voltage errors of 1 to 5 mV, which corresponds to a temperature error of hundreds of degrees! The precision op-amp used instead has a maximum offset voltage error of only 5 µV, or less than a quarter-degree error. The amplifier circuit is configured for a gain of 111; therefore, by using a 2.5-V reference with the A/D converter, you can measure temperatures from 32° to roughly 1000°F with approximately 1° resolution.

Of course, if you were to amplify the thermocouple signal, you'd also amplify any noise signals in this not too carefully constructed circuit. You can minimize the effects of noise by taking multiple (32 to 256) A/D readings and averaging them to produce a single temperature measurement.

The second thermocouple problem is that output is nonlinear. You cannot perform a simple division to convert voltage to temperature. This is easily dealt with by storing the type-K voltage-to-temperature table in the controller software. [1] The 0° to 1000°F temperature range can be divided into approximately linear regions. You can use linear interpolation to determine a temperature within a given region.

The final thermocouple problem is that output is relative, not absolute. The voltage at the probe end of the thermocouple where the two dissimilar wires connect is relative to the voltage at the circuit end, where the thermocouple wires connect to the voltage measurement circuit. If the probe and circuit ends have the same temperature, the measured voltage will be zero. If the probe end is hotter, a positive voltage will be measured. If it's colder, a negative voltage will be measured.

Therefore, in order to know the actual temperature at the probe end, you need to know the temperature of the circuit end. Historically, this was accomplished by keeping the circuit end in an ice bath so its temperature remained 32°. Consequently, the connection between the thermocouple and measuring circuit came to be known as the "cold junction." The process of using the junction temperature to calculate the actual probe temperature is referred to as cold junction compensation.

Instead of using ice, which is somewhat impractical for a barbecue application, you can use an IC temperature sensor to directly measure the actual temperature of the cold junction. For this to work, the sensor and physical connection point where the thermocouple wires meet the copper circuit wires must be the same temperature (i.e., isothermal). A difference in temperature will directly affect the accuracy of the temperature measurement. One method, which seems to work acceptably well, is shown in Photo 4.

Multiple Thermocouples

After you have the basic setup to measure one channel, it's easy to add an analog multiplexer to select from one of several channels. However, in order to prevent false alarms from the alarm circuit, you need to know if a thermocouple is actually connected to a channel. The circuit shown in Figure 1 allows the microprocessor to determine this.

Temp Measurement

You must complete several steps to make a single temperature measurement. First, select the channel to be read by setting the analog multiplexer to one of the four thermocouples. Then, see if there is actually a thermocouple connected by turning on the missing thermocouple test voltage and taking an ADC measurement. Remember that a single measurement is actually the average of 32 to 256 readings.

If the result is the maximum A/D value, then there is no thermocouple connected to that channel, so report a temperature value of zero. Otherwise, turn off the missing thermocouple test voltage, and measure the actual thermocouple output.

Using the A/D reading, the amplifier gain (111), and the A/D voltage reference value (2.5 V), compute the actual thermocouple voltage. A 10-bit A/D reading will be in the 0- to 1023-V range, so the actual voltage is equal to the following:

$$\frac{\text{A/D reading} \times 2.5}{1024 \times 111}$$

Next, take and average multiple A/D readings of the cold junction temperature sensor, and convert that voltage to the cold junction temperature. Using the cold junction temperature and the type-K voltage/temperature table, compute the type-K voltage that's equivalent to the cold junction temperature. Add the thermocouple voltage to the cold junction voltage. Using the type-K table, convert the summed thermocouple and cold junction voltages to the actual thermocouple temperature.

Measurement Accuracy

There are several sources for error in this process, including the inherent inaccuracies of the thermocouple and cold-junction temperature sensor, how isothermal the cold junction and sensor really are, the accuracies of the A/D converter and voltage reference, and gain errors in the amplifier. A good reality check is to test the temperature of boiling water, because 212°F is actually within the range of the typical barbecue-cooking temperatures. The circuit and setup described here yielded a boiling water temperature of 213°F, which is good enough for barbecue purposes.

Design Limitation

The circuit has a design limitation. When the thermocouple probe is at a lower temperature than the cold junction, the voltage produced is negative. However, the circuit, as configured, is single-ended, so it treats a negative voltage as if it were 0 V.

Consequently, if the probe is colder than the cold junction, the computed temperature will be the cold junction temperature, not the actual probe temperature. There are several ways to correct this, but none were used, because I was only concerned with high temperatures, not low ones.

Airflow Control

Now you're capable of measuring the cooker's temperature, but how can you control it? Adjusting the lower or upper draft openings to regulate the airflow would require a motorized mechanism. A simpler approach is to use a small fan.

Photo 1: The door at the bottom opens and closes, and the cap at the top turns to regulate the cooker temperature, which is monitored with a dial thermometer.

Little airflow is required for barbecue-cooking temperatures. The 40-mm fan that I used only puts out 10 cubic feet per minute, but can produce cooker temperatures higher than 400°F. Although bigger fans can produce higher temperatures, they're unnecessary for barbequing, and you don't want to cook the electronics, which are mounted only a few inches from the firebox. The fan speed can be controlled by the microprocessor using one of its PWM outputs from fully on to off in 255 steps.

So little airflow is needed that air leaks can prevent low temperatures from being reached! There must be a good seal between the cooker's lid and base. Also, you must prevent air from leaking in through the opening when the fan is off. The simple flapper valve, which is shown in Photo 5, accomplishes this; it's constructed from a piece of aluminum pie pan that's epoxy-glued to a small metal hinge. When the fan is on, it opens the aluminum flap. The flap shuts when the fan is off, sealing the opening.

Controller Electronics

Figure 2 shows the schematic for the controller. The microprocessor is an ATmega8, which has 8 KB of flash memory for program storage and 1 KB of RAM. The on-board oscillator is used at its default setting of 1 MHz, so a crystal or clock circuit isn't needed. Four of its six A/D channels are used for the thermocouple amplifier output, LM34 temperature sensor, and potentiometers needed for the cooker and food temperature settings. Digital lines are used for the LCD, which has an HD44780-type interface, fan, analog multiplexer channel select, piezoelectric alarm, missing thermocouple detection circuit, and RS-232 connections.

The ATmega8 can be in-circuit programmed. Power is supplied through an unregulated 12-V wall wart or 12-V sealed lead-acid battery. Regulators are needed to convert this to 5 V for the electronics, a 2.5-V 0.2% precision reference voltage, and 12 V for the fan. The transistor that controls the fan must be rated according to the fan's current requirements. The fan that I used draws 100 mA, so a 2N7000 MOSFET that can handle 200 mA is adequate. With a backlit LCD and the fan turned on, the current draw is only 120 mA, so a 7-Ah battery can power the controller for several days.

The RS-232 port is used primarily for data logging. Because the barbecue is usually outdoors and the computer indoors, a MAX232 and low data rate (i.e., 2400 bps) are used to allow for the greatest possible distance between the controller and data logger. Using six-conductor telephone cable with RJ11 to RS-232 modular connectors, distances of 100¢ are easily achieved. The alarm signal is also brought out on the RS-232 cable, so the outdoor on-board alarm can be muted at night, and a smaller piezo alarm that's connected to the end of the cable can be placed indoors where someone is likely to hear it.

Photo 6 shows the electronics, which are mounted on the lid of a standard doublewide outdoor "wet location" electrical box. The box itself is mounted to the lower draft door of the cooker. The cover supplied with the box allows the controller to stay outdoors and mounted on the cooker at all times. Alternate packaging schemes are possible, such as mounting the fan on the draft door, with the electronics further away in a

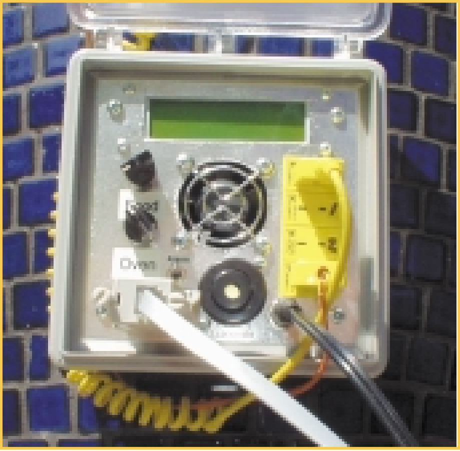

Photo 2: The LCD shows the actual and desired cooker temperatures, and up to three food temperatures. You can mute the piezoelectric alarm. The cooker temperature can be set from 180° to 400°F in 5° steps, and the food alarm can be set from 100° to 220°F.

separate box. Or you could choose to mount nothing on the cooker itself and pipe the air to the cooker through a metal pipe or flex tubing.

Controller Software

The software was written with the ImageCraft ICCAVR C compiler. Approximately half of the 8 KB of flash memory has been used so far. The ATmega8's flash memory is in-circuit programmed through an Atmel standard ISP header. The ImageCraft IDE provides a programmer. In addition, stand-alone programming software is widely available on the Internet. ImageCraft also offers a free 30-day unrestricted trial period for its IDE and compiler.

The software continuously reads the thermocouple temperatures, cooker, and alarm settings, and then displays them on the LCD. The alarm is sounded if there is no cooker temperature probe, the cooker temperature is too high or too low, or the temperature of the food has reached the alarm temperature. A simple software state machine is used to prevent "temp too low" alarms during the time the cooker is coming up to temperature, set-point changes, and when the lid is opened during cooking. The settings and temperature readings are sent to the RS-232 port once per second so that a terminal capture program, such as HyperTerminal, can be used for data logging.

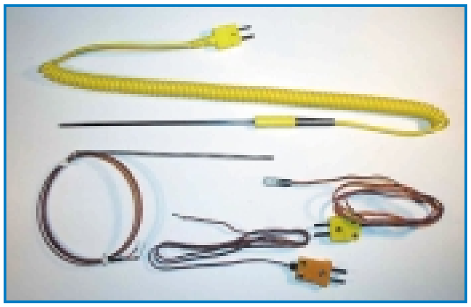

Photo 3: Beaded-type thermocouples respond too quickly for use in an oven. Adding a washer provides thermal mass. The slender stainless probes are used for food temperatures. The thicker probe is for oven temperatures.

The temperature reading and settings are used to implement the controller's main function (i.e., the process-control algorithm). Once per second, the desired and actual cooker temperatures are used to determine the fan's speed. Note that the controller was designed with two control strategies in mind, On/Off and PID.

On/Off Control

If the cooker temperature is below the desired level, the simplest control algorithm is to turn on the fan fully. Turn it off if the cooker temperature is above the desired value. The resulting cooker temperature will tend to oscillate around the desired value, because the thermal inertia in the cooker will cause the temperature to continue to rising after the fan is turned off and falling after the its turned on. As long as the average temperature is what you want and the swings aren't too big, this is perfectly acceptable. After all, this is how most kitchen ovens work, and the goal is to be as good as one of those.

PID Control

A more sophisticated approach is to set the fan to the exact speed needed to reach a given temperature, and then incrementally adjust the speed to maintain that temperature as conditions change. The PID algorithm uses proportional, integral, and derivative calculations to do just that.

A measure of current conditions, the proportional part of the calculation is based on the size of the error (i.e. the difference between the desired temperature and the actual temperature). The integral portion of the calculation is based on the sum of all the previous error values; it's a measure of any longer-term error trend. The derivative part of the calculation is based on the change in the error, not the temperature, and it measures how quickly the error is changing.

Each of the three components can be given a different weighting factor. If the weighting factors are chosen correctly—through a process called "tuning"—the PID algorithm will adjust the fan speed to bring the actual temperature to the desired temperature with no overshoot, and it will maintain that temperature with little or no oscillation even if conditions change. Parallax's Industrial Control Student Guide Version 1.1 provides a straightforward introduction to the PID algorithm.

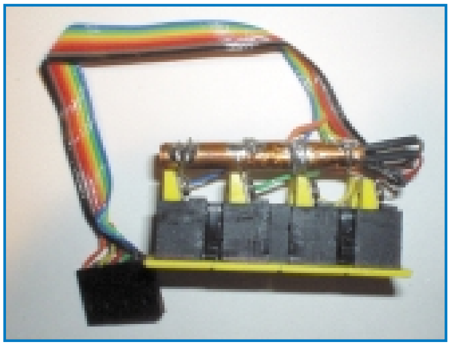

Photo 4: An LM34 (inside the 0.25 inch copper tube) measures the tube's temperature, which is approximately isothermal with the thermocouple connector.

Although the hardware was designed with a variable fan speed so that PID could be used, I ran the initial tests with an On/Off algorithm. The results are shown in Figure 3. The goal was to build a device that is as good as an indoor oven. But, even with a simple On/Off control, the ceramic cooker is considerably better. So, the final controller uses On/Off control, and the more complex PID algorithm isn't needed.

Photo 5: You don't want the fire to go out completely; therefore, when shut, the valve limits but doesn't completely restrict airflow through the fan's opening.

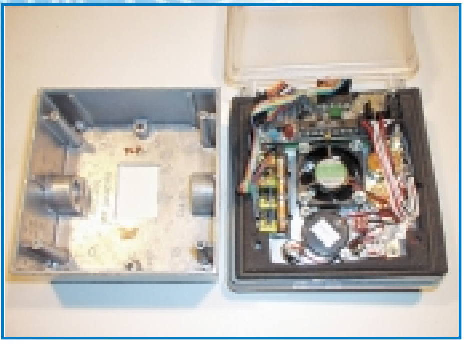

Photo 6: The electronics are assembled on a piece of double-plane prototype board and mounted over the LCD. The weatherproof gasket and cover come with the electrical box.

Using the System

Photo 7 shows the ceramic cooker with the automatic temperature controller in a typical setup. The meat, rubbed with secret barbecue spices, is on the top grill with a drip pan underneath it. Beneath the drip pan is a foil-covered pizza stone, which prevents the bottom of the meat from charring. A full load of hardwood lump charcoal is under the pizza stone. I added chunks of hickory and cherry wood to add a smoked flavor.

After the temperature probes are inserted and the lump lit, close the lid. The upper draft control is opened a half turn, and the cooker temperature and food alarms are set. As soon as the alarm sounds, you can check the meat to see if it's done. You aren't required to monitor or adjust the cooker temperature, although some barbecue styles require you to periodically baste and turn the meat.

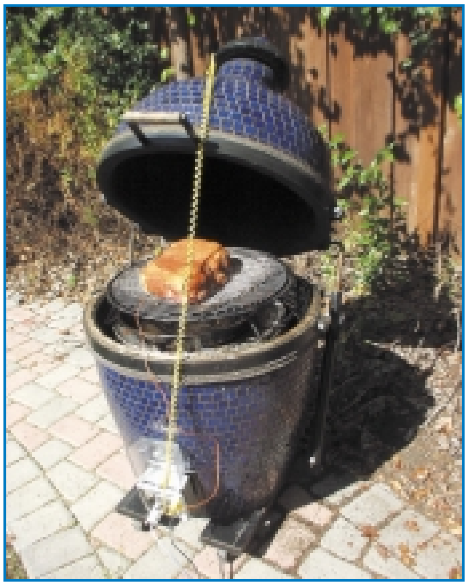

Photo 7: A typical setup includes food, oven and food temperature probes, and the controller.

Figure 4 shows the log of an actual cooking session. The meat required approximately 17 hours to reach a final temperature of 192°F. The cooker temperature was set at 225°F. It stayed within 3° of that setting more than 98% time, and remained within 1° more than 85%.

Who's Tending the Fire?

Traditionalists may question the idea that real barbecue can be achieved without an all-day or all-night fire-tending vigil. But modern technology can make this cooking style accessible to those who otherwise wouldn't have the time for it.

The system is useful for more than barbequing. After the fire is lit, the cooker is as easy to use as a kitchen oven but has the advantage of better temperature control. In addition, it makes it easy to monitor food temperature, and it gives you the ability to add a smoked flavor to anything you cook. ■

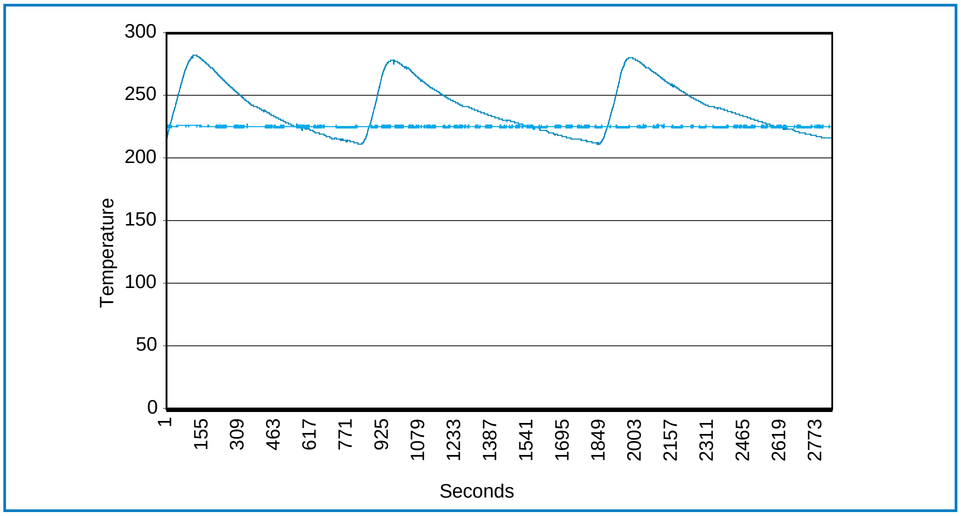

Figure 3: Compare the temperature values over a 45-min. period for an indoor kitchen oven (dark blue) and the automatically controlled ceramic cooker (light blue). In this instance, both are set at 225°. As you can see, the ceramic cooker shows much smaller temperature swings and a more accurate average temperature.

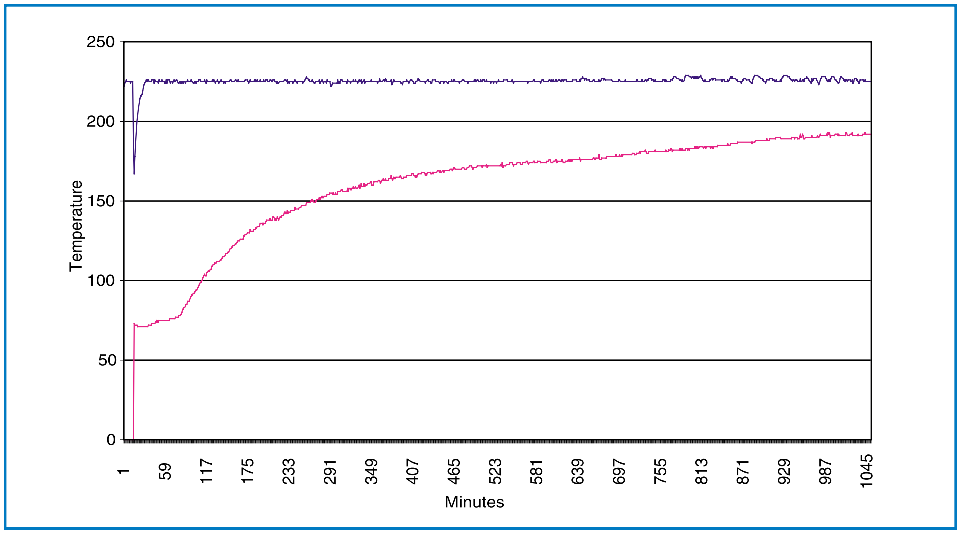

Figure 4: The data log shows a stable ceramic cooker temperature (in blue) and a slowly rising meat temperature (in red). The initial temperature dip occurred when I opened the lid to the preheated cooker to add the food.