# Memorandum

**To:** Alex Grichener
Dan Koller

**cc:** Kirk Crady, Geoff Ediss
Tony Kerr, Gene Lauria

**From:** John Effland

**Date:** 2003-10-09

**Revisions:**

| | | |
|---|---|---|
| 2003-10-03 | jee | Initial |
| 2003-10-07 | jee | Added `RackID` to `TempSensors` and `TempMeters` |
| 2003-10-09 | jee | Changed `TempSensor.TestDate` to simply `Date` |

**Subject:** Programming LakeShore 218 Temperature Monitors with Calibration Data from DT-670 Sensors

## 1. Summary

This memo describes a procedure for programming LakeShore 218S Temperature Monitors to use dip-test results for DT-670 temperature sensors. Also included is the schema for the database that holds calibration data for both LakeShore DT-470 and DT-670 series temperature sensors.

## 2. Procedure

To display the corrected temperature values on the front panel of LakeShore 218 Temperature Monitors, software will upload correction factors for the sensors into the temperature monitors. LakeShore DT-470 sensors can use their three-point "SoftCal" routine to correct the standard DT-470 curve near temperatures of 4K, 80K, and 300K. DT-670 sensors can't employ the "SoftCal" routine and instead must use modified "UserCurves," which consist of a number of voltage and temperature mappings suitably corrected for calibration points[1]. The temperature monitor will be configured so that each temperature sensor uses the appropriate corrected "UserCurve" to map sensor voltages to temperatures by linearly interpolating between the breakpoints.

The procedure to program the temperature monitor to display corrected temperatures when using DT-670 sensors is:

1. Obtain the table of breakpoints corresponding to temperatures *vs*. voltage for the for the DT-670 temperature sensor. The table only needs to be downloaded once and is graphed in Figure 1 from two different sources. The curves labeled "330" were downloaded from the LakeShore web site. The other curve is from a particular LakeShore 218 Temperature Monitor and is nearly the same as the "330" curve. The data for that curve is tabulated in Table 1. Because the curve from the temperature monitor contains

[1] See Attachment 1 for an e-mail from LakeShore describing their reluctance to use 3-point calibration with DT-670 temperature sensors.

more calibration points, it will be used as the reference curve for all DT-670 sensors. As discussed later, this curve is stored in the database and will be downloaded by software, corrected for the dip test data, and uploaded again into the temperature monitor.

2. From the temperature *vs.* voltage breakpoint curve obtained in 1 above, generate a 3rd order polynomial to approximate the measured points between 2K and 10K as shown in Figure 2. This equation, valid from 2K and 10K, is

$$T = -323.05V^3 + 1452.3V^2 - 2202.9V + 1133.4$$

where: T is sensor temperature in K, and
V is sensor voltage in volts

The curve labeled "Estimate Error" in Figure 2 shows the goodness-of-fit is about ±0.03K between temperatures obtained from the polynomial and the temperature-voltage pairs downloaded from the temperature monitor.

3. Measure the voltage output of the DT-670 temperature sensor when dipped in liquid helium using the 4-wire dip-testing probe.
4. With the 3rd order polynomial generated in 2 above, calculate the DT-670's temperature corresponding to the voltage measured from the liquid helium dip testing in 3 above.
5. Subtract the calculated temperature in 4 above from the boiling point of helium, after correcting the boiling point for atmospheric pressure using the curve in Figure 3.
6. Apply the correction factor obtained in 5 above to the DT-670 breakpoint data obtained in 1 above.
7. Upload the breakpoint data into the LakeShore 218 Temperature Monitor as a "User Curve" and configure the temperature monitor to use that "User Curve" with the appropriate sensor.

As described below, the temperature sensor calibration data will be stored in the SQL Server database to simplify software maintenance. The database simplifies changing sensors. When different temperature sensors are used with a particular temperature monitor, only database records need to be changed and the software can remain unchanged.

## 3. Database Schema

The Microsoft SQL Server database running on computer `wheat` holds calibration data for each temperature sensor. Software will to read records from the database, calculate a corrected "UserCurve," and upload that data into the temperature monitor.

Figure 4 shows the table design for the database, which consists of four related tables. Table `TempMeters` holds information about each temperature monitor and the field `RackID` holds information about which measurement rack contains the monitor. Other fields are identified in Table 2.

Each record in table `TempSensors` holds information about a particular sensor as defined in Table 3. Foreign key-field `fkMeterID` maps which sensors are used with the corresponding temperature meter record in table `TempMeters`. Field `RackID` also identifies the rack where the sensor is used. Field `MeterChan` in table `TempSensors` holds the requisite information about which particular sensor is assigned to a particular channel on the temperature monitor.

Tables `TempCalData` and `TempRefCurve` are two child tables associated with `TempSensors` and contain calibration and "UserCurve" information about the particular sensor record. Table `TempCalData` includes fields to hold the 3-

point calibration data, which is provided by LakeShore and also can be measured manually. Field assignments are itemized in this memo in Table 4. The 3-point calibration data is required for the temperature monitor's "SoftCal" routines when using DT-470 sensors and 3-point data are also provided by LakeShore for DT-670 sensors. The table `TempCalData` provides the means to store multiple calibration sets for a particular sensor and the software will generally always use the most recent data as determined by the `Test Date` field. An alternative would have been to include the temperature-voltage pairs in `TempSensors`, but then only a single calibration could be stored in the database and information about historical calibrations would be lost. Both three-point calibration data as well as single-point data (typically at 4.2K) can be stored in this table.

Table `TempRefCurve` holds the standard DT-670 temperature-voltage data pairs as shown in Table 5. A single reference curve is defined in this table by using a constant value in the field `CurveID`. For example, all records for the "330" data discussed above would have the same number in the `CurveID` field. The particular reference curve used for each sensor in table `TempSensors` is determined by the foreign-key value stored in the field `fkCurveID`. That field holds the value found in the field `CurveID` in table `TempRefCurve`.

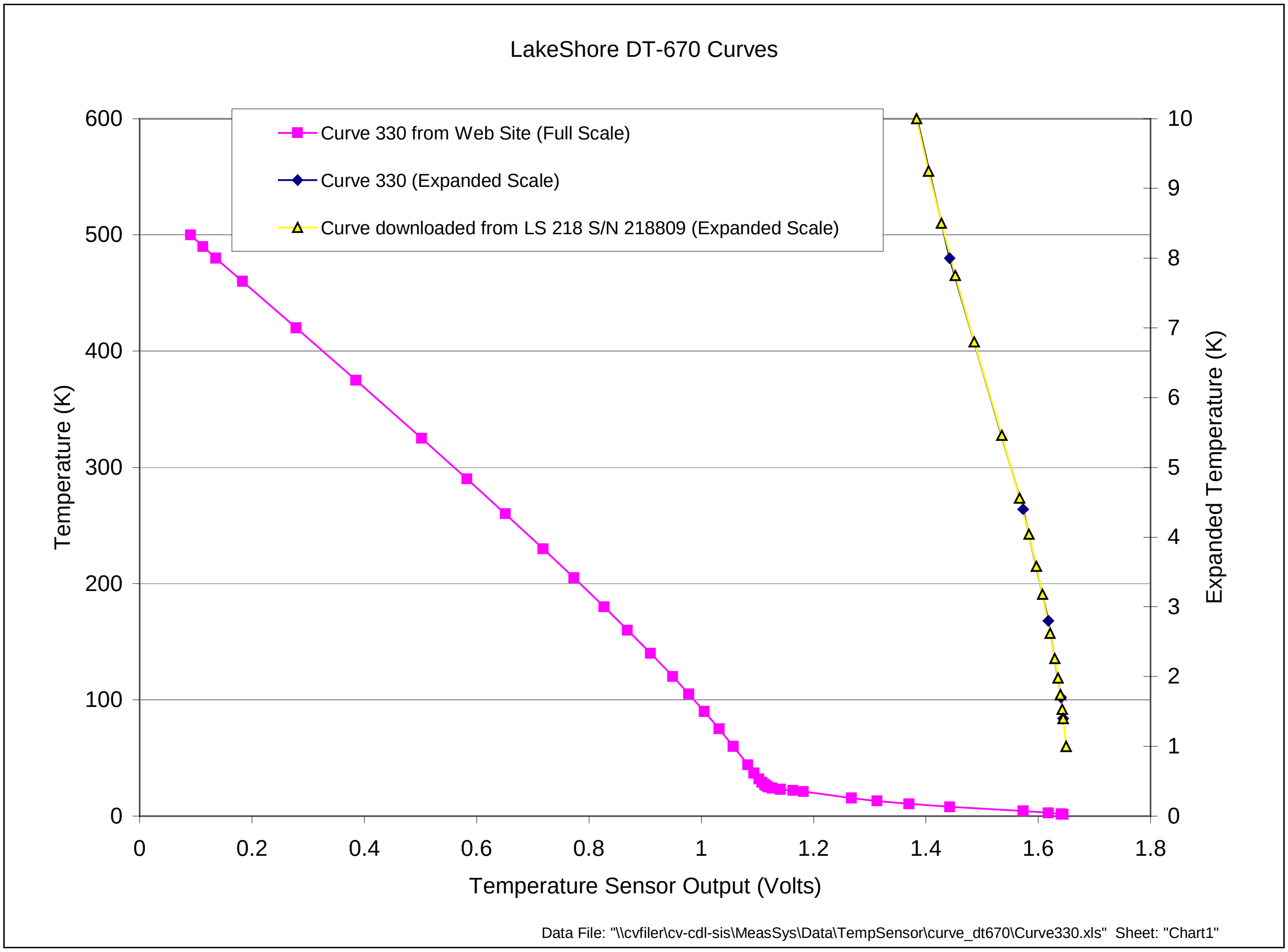



**Figure 1: LakeShore DT-670 Temperature Sensor Curves**

| Breakpoint Data from LakeShore 218S SN 218809 | | | | | |
|---|---|---|---|---|---|
| Breakpoint Index # | Sensor Voltage (V) | Sensor Temperature (K) | Breakpoint Index # | Sensor Voltage (V) | Sensor Temperature (K) |
| 1 | .07964 | 505. | 40 | 1.11204 | 27.3 |
| 2 | .09057 | 500. | 41 | 1.11414 | 26.5 |
| 3 | .11024 | 491. | 42 | 1.11628 | 25.8 |
| 4 | .13656 | 479.5 | 43 | 1.11853 | 25.2 |
| 5 | .17918 | 461.5 | 44 | 1.1209 | 24.7 |
| 6 | .26539 | 425.5 | 45 | 1.1234 | 24.3 |
| 7 | .34952 | 390. | 46 | 1.12589 | 24. |
| 8 | .4528 | 346. | 47 | 1.12913 | 23.7 |
| 9 | .51339 | 320. | 48 | 1.13494 | 23.3 |
| 10 | .56313 | 298.5 | 49 | 1.14495 | 22.8 |
| 11 | .60785 | 279. | 50 | 1.16297 | 22. |
| 12 | .64872 | 261. | 51 | 1.17651 | 21.3 |
| 13 | .68694 | 244. | 52 | 1.19475 | 20.2 |
| 14 | .72251 | 228. | 53 | 1.24208 | 17.1 |
| 15 | .75549 | 213. | 54 | 1.26122 | 15.9 |
| 16 | .78699 | 198.5 | 55 | 1.27811 | 14.9 |
| 17 | .81703 | 184.5 | 56 | 1.2943 | 14. |
| 18 | .84454 | 171.5 | 57 | 1.3107 | 13.2 |
| 19 | .86958 | 159.5 | 58 | 1.32727 | 12.4 |
| 20 | .89323 | 148. | 59 | 1.34506 | 11.6 |
| 21 | .91447 | 137.5 | 60 | 1.36423 | 10.8 |
| 22 | .93436 | 127.5 | 61 | 1.38361 | 10. |
| 23 | .9529 | 118. | 62 | 1.40454 | 9.3 |
| 24 | .97013 | 109. | 63 | 1.42732 | 8.5 |
| 25 | .98607 | 100.5 | 64 | 1.45206 | 7.8 |
| 26 | .99892 | 93.5 | 65 | 1.48578 | 6.8 |
| 27 | 1.01064 | 87. | 66 | 1.53523 | 5.5 |
| 28 | 1.02125 | 81. | 67 | 1.56684 | 4.6 |
| 29 | 1.03167 | 75. | 68 | 1.58358 | 4. |
| 30 | 1.04189 | 69. | 69 | 1.5969 | 3.6 |
| 31 | 1.05192 | 63. | 70 | 1.60756 | 3.2 |
| 32 | 1.06277 | 56.4 | 71 | 1.62125 | 2.6 |
| 33 | 1.07472 | 49. | 72 | 1.62945 | 2.3 |
| 34 | 1.0911 | 38.7 | 73 | 1.63516 | 2. |
| 35 | 1.09602 | 35.7 | 74 | 1.63943 | 1.7 |
| 36 | 1.10014 | 33.3 | 75 | 1.64261 | 1.5 |
| 37 | 1.10393 | 31.2 | 76 | 1.6443 | 1.4 |
| 38 | 1.10702 | 29.6 | 77 | 1.6495 | 1. |
| 39 | 1.10974 | 28.3 | | | |

**Table 1 : Sensor Breakpoint data for LakeShore DT670 Temperature Sensor from LakeShore 218 Temperature Monitor SN 218209**

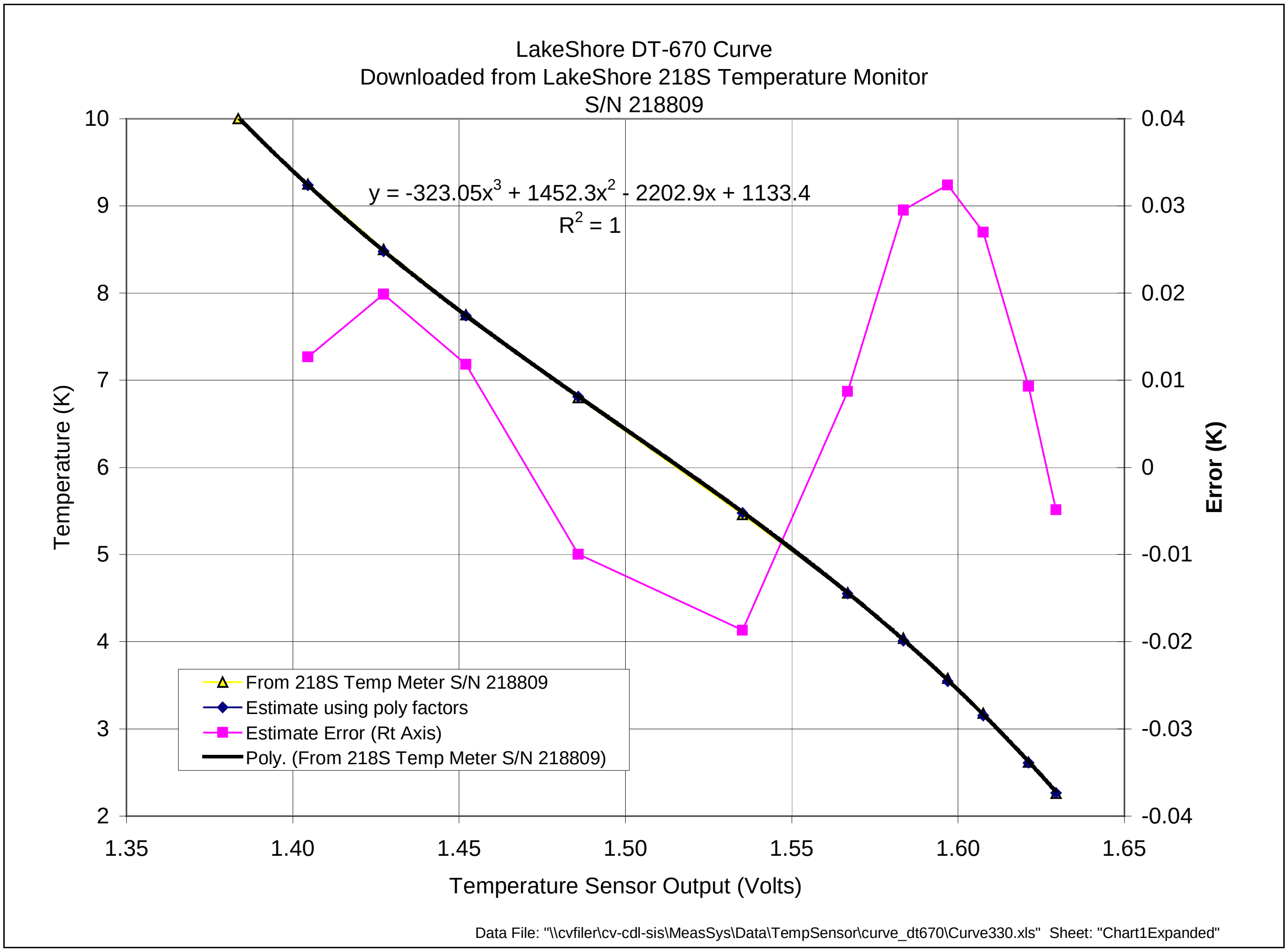



**Figure 2: Temperature Sensor Curve Polynomial Approximation**

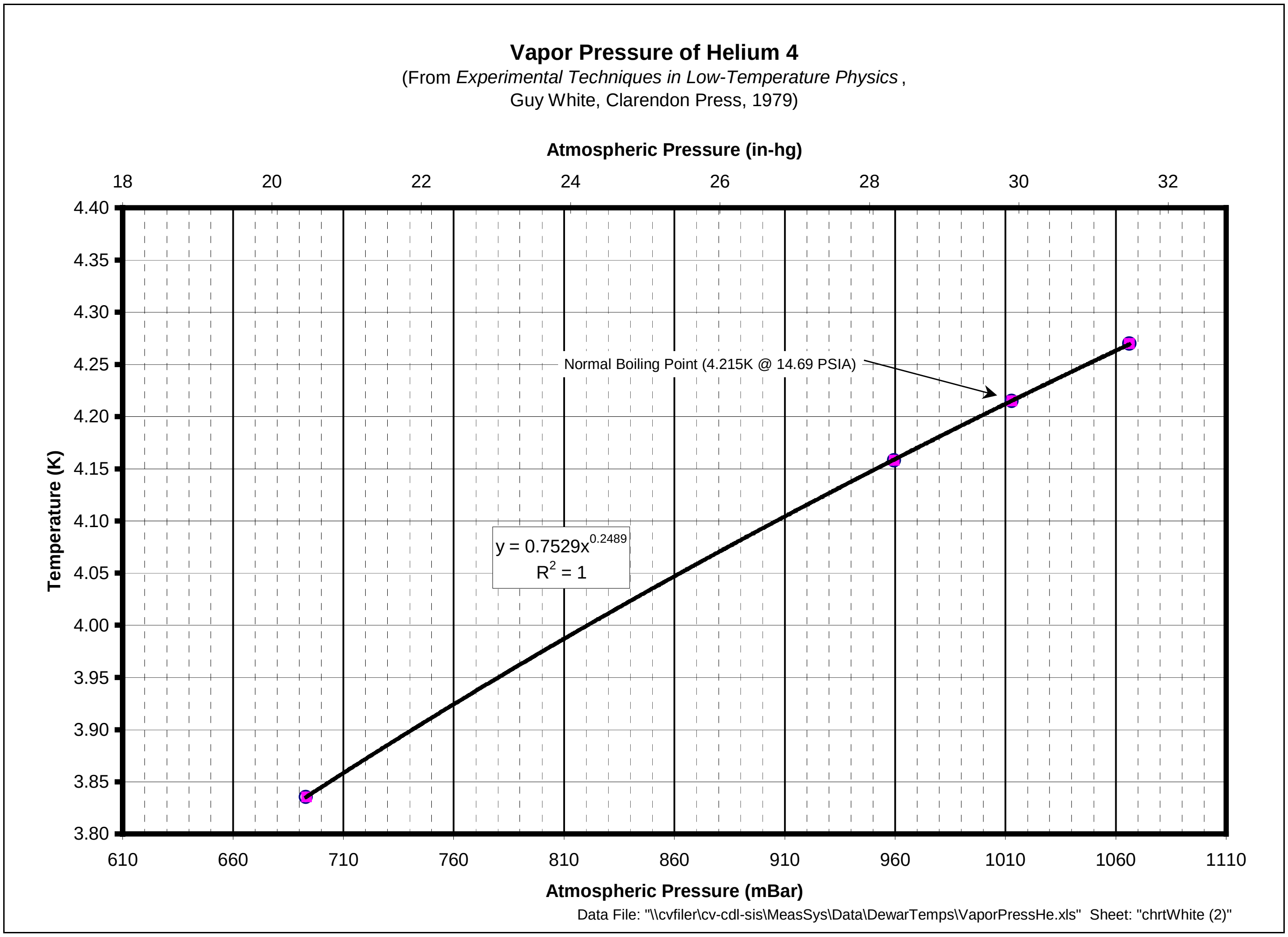
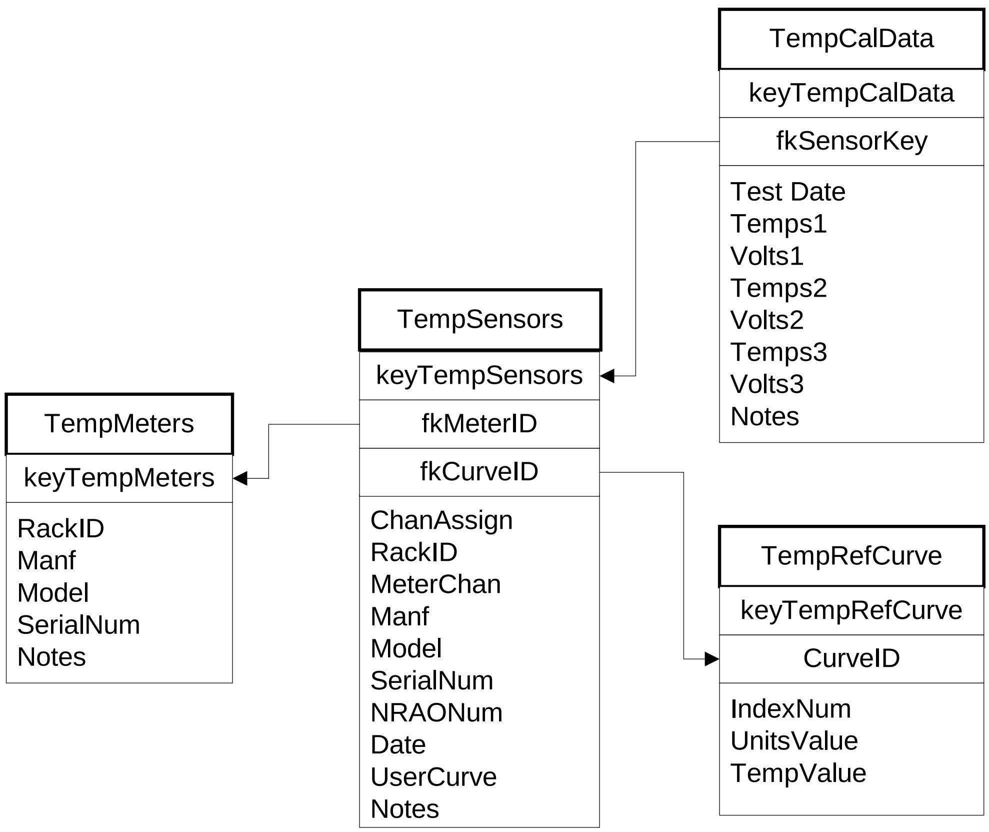



**Figure 3: Liquid Helium Boiling Point vs. Atmospheric Pressure**




| Rev | Date | Who | Description |
|---|---|---|---|
| 02 | 2003-10-09 | jee | Changed TempSensors.TestDate to Date, added ChAssing |
| 01 | 2003-10-07 | jee | Added RackID to TempSensors, renamed key fields |
| 00 | 2003-10-02 | jee | Initial |

| | | |
|---|---|---|
| **SIS Mixer Software** | Edit Date: 2003-10-9 11:55 | |
| Database Schema, LakeShore Temperature Sensors | | |
| Page 1 of 1 | Rev: 0 | Creator: jee |
| File: TempSensorCal.doc | | National Radio Astronomy Observatory |

**Figure 4: Database Schema for Temperature Sensors**

**Table 2: Table `TempMeters` Design**

| Field Name | Description |
|---|---|
| | |
| keyTempMeters | Unique index for each record |
| RackID | Holds the rack location of the meter |
| Manf | Meter manufacturer |
| Model | Model of meter |
| SerialNum | Meter serial number |
| Notes | General notes about this meter. |

**Table 3: Table `TempSensors` Design**

| Field Name | Description |
|---|---|
| | |
| keyTempSensors | Unique index for each record |
| fkMeterID | Relates this record to a particular temperature monitor stored in table `TempMeters` |
| fkCurveID | Relates this record to a set of records stored in table `TempRefCurve` |
| ChanAssign | Contains an integer that describes the location of sensor, such as:<br>0 - Room temp sensor<br>1 - Mixer1 sensor |
| RackID | Holds the rack location of the sensor |
| MeterChan | The channel in the temperature monitor where this sensor is installed. |
| Manf | Sensor manufacturer |
| Model | Sensor model number |
| SerialNum | Sensor serial number |
| NRAONum | Proprietary ID number assigned by NRAO. Used on early DT-470 sensors |
| Date | Date of dip test |
| UserCurve | `UserCurve` number assignment, used for DT-470 sensors |
| Notes | General notes about this sensor. |

**Table 4: Table `TempCalData` Design**

| Field Name | Description |
|---|---|
| | |
| keyTempCalData | Unique index for each record |
| fkSensorKey | Relates this record to a particular temperature sensor stored in table `TempSensors` |
| TestDate | Date of dip test |
| UserCurve | `UserCurve` number assignment, used for DT-470 sensors |
| Temps1<br>Volts1 | Temperature-voltage pair for 4.2K dip test. |
| Temps2<br>Volts2 | Temperature-voltage pair for 78K dip test. |
| Temps3<br>Volts3 | Temperature-voltage pair for 300K test. |
| Notes | General notes about this sensor. |

**Table 5: Table `TempRefCurve` Design**

| Field Name | Description |
|---|---|
| | |
| `keyTempRefCurve` | Unique index for each record |
| `CurveID` | Constant number for a particular temperature breakpoint dataset |
| `IndexNum` | Array element number used by `UserCurve` routine. |
| `UnitsValue` | Temperature sensor units in volts |
| `TempValue` | Corresponding temperature sensor value in K |

# Attachment 1: E-Mail from LakeShore concerning 3-Point Calibration of DT-670's

-----Original Message-----
From: Joe Yeager [mailto:jyeager@lakeshore.com]
Sent: Tuesday, 2003 September 30 09:36
To: jeffland@nrao.edu
Subject: softcal 670

John,

Jim forwarded your message to me. Let me see if I can help.
We do not have a softcal available for the DT-670 for a couple of reasons. One reason is that the existing bands (A,B, C, or D) allow selection of a sensor that is nearly as good as a softcal. Band D is nearly as good as a softcal except at 60-345K. The softcal is +/-0.15 while the Band D is +/- 0.25K. The band D is $182 while a softcal for the 470 diodes is $303. The only band that could be improved is the Band C ($161 and +/-1K).

Another reasons is that there is a problem implementing the softcal between 4 and 25K. In general the 670 all have a tighter grouping and allow a tighter tolerance band. But in the region between 4 and 25K, they are unpredictable within their tolerance.

For example, We specify a tolerance from 2 to 100K of +/-0.25 K for the A band. The largest deviation is around 15K so the tolerance at 4.2K is actually tighter than 0.25K. This is done to insure meeting tolerance specs between 2 and 25K.
The 670 have a better overall tolerance to the standard curve, but the 470 were more predictable below the knee. So, in fact, we can not do the soft-cal for the 670 below 25K. If you are only interested in T > 25 K I would recommend the Band D. You could do a linear correction with the 77K and 300K points.

However, if you are only interested in 4.2K you could still work this out with the 670. The key to the softcal is that all the diodes have a “pinning” point at 25K. If you draw a line from the corrected 4.2K reading to the standard curve value at 25K you will get something that will work around 4.2K. So use the 4.2K as the offset and the correction factor is the new curve based on the 4.2K point and the old 25K point. The biggest problem region is between 10-14 K. Still, the tolerance at 4.2K is better than we specific (since we specify a range of temperatures).

Let me know what you think. To push this further, we would need to know what temperature range and level of accuracy you need.

Thanks

Joe

Dr. Joe Yeager
Product Manager
Cryogenic Thermometry
Lake Shore Cryotronics, Inc.
jyeager@lakeshore.com