

Precision Diode Temperature Sensing using ADM1034

March 13, 2008

Contents

1 Introduction	1
1.1 Advantages of Diode temperature sensors	2
Disadvantages	3
2 Demo Board Hardware	3
3 Initialisation	3
4 Temperature Format	4
4.1 Reading with the I2C2PC Adaptor	4
4.2 Converting the Hex Data to Temperatures	5
4.3 Extending the Number of Sensors with Bus Switches	5
5 Matlab Example	5
6 Results	6
6.1 Room Temperature Measurements	7
6.2 Interchanging Sensors	9
6.3 Noise	9
6.4 Self Heating and Internal Sensor	9
6.5 Summary of Results	9
7 Drawings	9
7.1 Circuit Diagrams	10
7.2 PCB Drawings	11

1 Introduction

The Analog Devices www.analog.com ADM1034 temperature sensor and fan controller IC is an easy way to make an I2C based temperature measurement system using

external diode sensors¹.

The I2C2PC www.i2cchip.com/pdfs/i2c2pc.pdf adaptor provides an easy way for a PC to interface to the I2C bus, and works with any programming language and any OS. The example below shows datalogging on a Windows PC using Matlab for analysis and plotting. The associated command and Matlab files are in TemperatureSensor_ADM1034.zip www.i2cchip.com/pdfs/TemperatureSensor_ADM1034.zip

- 2 channels of external sensing using diodes + 1 internal sensor channel
- -64 - 191°C range
- 0.03C resolution.
- Internal averaging to improve noise. (0.1°C rms measured)
- 3 current level measurement technique to eliminate series resistance errors
- Internal comparators allow direct over/under temperature indication without any software.
- Simple circuit, small package.
- Specified at 3.3V, operates to 5.5V
- 4 hardware I2C Addresses: Up to 24 external sensor channels can be read directly from a single I2C2PC adaptor. (Use PCS9545 4 way bus switches to read up to 384)

Analog Devices (and others) make related devices. The ADT7466 is a similar device with an almost compatible footprint, but somewhat lower specs.

1.1 Advantages of Diode temperature sensors

We have compared different temperature sensing techniques elsewhere. The external diode temperature sensor has:

- Most readily available sensor: any common bipolar transistor is suitable, available anywhere, any time.
- Huge choice of packages: From less than 1mm² to packages with screw holes to bolt on.
- Lowest possible sensor cost: Sensors are so cheap they are disposable, and can be freely glued, soldered, cast in place
- Interchangablity is reasonable: No calibration needed for many jobs
- Reasonable temperature range: -64 - 191 °C
- Only a passive device on the sensor cable, no active, powered, digital devices: robust and simple to make.

¹When we say diode sensors, we normally use a bipolar transistor with base and collector connected.

²DFN1006 package is 1mm x 0.6mm x 0.5mm eg BC847BLP
<http://www.diodes.com/datasheets/ds30525.pdf>

- Negligible self heating of sensor (<10uW sensor heat) (compared to internal sensor IC's)

Disadvantages

- Capacitance limits cable length from ADM1034 to Diode to ~20m to 40m max using Cat5 cable.

2 Demo Board Hardware

A demo board was made up with an RJ45 connector. Header pins select the 4 I2C addresses.

See Schematics and Drawings at the end. These are not a stock item, but if requested, either bare pcbs, or built up boards can be supplied. Contact sales@i2cchip.com



3 Initialisation

At power on the sensor will be automatically updating its temperature registers 8 times a second, with averaging enabled for noise reduction. The only setup required is to setup the block read size for the number of temperature values you want to read.

```
S A0 00 06 P //setup block read size (6 bytes) for temperature blocks.  
SA0C0R06P //read the 3 temp sensor values
```

If you want to enable hardware threshold levels these must be initialised. There are two outputs that can be used for comparison: THRESH and ALERT/INT. THRESH can only be used to signal over temperature. In our demo board there is an LED on this. The ALERT output can be programmed with separate upper and lower limits. While normally used as an interrupt out to the bus, it can also just be used as a comparator output for an LED or alarm.

The chip also has a calibration offset register which can be set during initialisation.

The file InitADMx4.dat initialises 4 chips on a common I2C with addresses 0xA0, 0xA2, 0xA4, 0xA6

4 Temperature Format

The data is stored in 2 bytes, and is stored LSB first. It must be read LSB *then* MSB as there is a data lock mechanism to prevent updates between byte reads.

Data is most easily read in blocks. The ADM1034 has a non-intuitive block read mechanism.

- First you must write the number of bytes that will be read, into the block length register (reg 0x00). This only needs to be done when it changes (i.e. in the initialisation above)
- To do a block read, you must address the register with bit 7 set (so to block read reg 0x40 you address 0xC0)
- When you do a block read, the first byte returned will be the block length value. So you must read 1 byte more than the data length, and ignore this byte when processing the data
- Write the register address with bit 7 set, then do a repeated start and read the block of data.

4.1 Reading with the I2C2PC Adaptor

This assumes that you are familiar with the I2C2PC's simple I2C commands.

So to read 3 temperature value (6 bytes) from the device at I2C bus address 0xA0, you must read 7 bytes

```
S A0 C0 R 07 P .
```

Adaptor replies with "0670510850504F" ie number of bytes (06), then 3 pairs of data values 7051, 0850, 504F. As described above, LSB is first, so the 16 bit values are actually 0x5170, 0x5008, 0x4F50

If you only want to read the external diode sensors then just read 4 bytes, from register 0x42 (0xC2 with bit 7 set)

```
S A0 C2 R 05 P .
```

To read data from two chips (addr=0xA0,0xA2), with a comma delimiter between data.

```
S A0 C0 R 07 P , S A2 C0 R 07 P .
```

The read commands for 4 sensors are in Read3TempsADMx4.dat. Corresponding files for 1,2,3 sensors are also given.

The batch file "LogADMData.bat" contains the two Realterm commandlines that initialise the system, and then run the logging.

4.2 Converting the Hex Data to Temperatures

The datasheet formula for calculating temperature is a little misleading. This following formula works.

Temperature = (MSB-64) + (LSB and 248) / 256)

$$Temperature = (MSB - 64) + \left(\frac{LSB \& 248}{256} \right)$$

Again, note that the LSB byte is first, and MSB second. So 0x5170 = 16.4375°

4.3 Extending the Number of Sensors with Bus Switches

By using the Bus Switch www.i2cchip.com/pdfs/BusSwitch_MUX3.pdf based on the PCA 9545 you can extend the number of channels. Using a bus switch means that only 4 sensors are on each I2C cable, and so bus speed and noise are not compromised by enormous cable runs. It also make for an easy to debug system.

- ADM1034 has 4 addresses, giving 8 external sensors per I2C bus segment
- Bus Switch has 4 outputs, giving 32 sensors per Bus Switch PCB
- Bus Switch has 4 addresses, so 4 bus switches can share one upstream I2C bus, for 128 sensors per upstream I2C
- I2C2PC adaptor has 3 separate I2C busses, for 384 sensors, using just a single level of bus switching.
- If you want to use 2 levels of bus switching, then the sky's the limit!

When using large switched arrays it is a good idea to use the reset pin to the multiplexor array, so that one faulty downstream sensor bus cannot lock up the multiplexor array. The reset pin is used to force deselection of the whole array, and then the next active bus is selected through the bus.

5 Matlab Example

This example will use our Realterm program Realterm <http://www.i2cchip.com/realterm> to control the I2C2PC adaptor, and collect the data into a text file. The text file will be analysed and plotted with Matlab.

As the data logging process is so simple, and is done by Realterm I have made a simple batch file "LogADMDData.bat". Only two lines do the work. The commandlines are quite self evident.

- Realterm dumps an initialisation file to configure the I2C2PC adaptor, and to configure the ADM1034. It does this once at the start only, and quits

- Realterm is started a second time. It goes into an endless loop sending the command file every 5 seconds, and capturing the return data to a capture file.
- Realterm will add a timestamp to the start of each line, before the data. (Matlab time format chosen)
- Realterm goes into a loop. Every 5 seconds it sends a file with commands to read the temperatures.
- The capture file can be read and processed while this goes on. "PlotTemps.m" will plot the data, while it is being collected.
- Beware of using Microsoft programs (ie excel), that try to lock the files they access. You cannot read a file while it is being written with excel.

Two lines from a captured data file for a 4 channel system is below. The first number is the Matlab datetime value, followed by the hex data from the I2C2PC adaptor.

```
39350.9159514,0670510850504F,06B0513050384F,064052D0500850,0650514850
```

Matlab has no trouble reading the hex data. (Excel and Openoffice can also read hex numbers with standard functions)

There are two Matlab examples, one is for a system with a single ADM1034, the other has 4 of them.

Under a Unix operating system a simple bash script can perform the same functions as Realterm does above. See Linux Logging with Linksys Router <http://www.i2cchip.com/linksys.html> for an old example of a logging system running on Linux.

You can also use Realterm to talk to a telnet port. In this way you could use your Windows box to log data from a remote machine with a serial port redirector program. Under unix SREDIR is a simple redirector.

6 Results

Four ADM1034's were logged simultaneously at room temperature, VCC=3.3V.

All were obtained from the same supplier. #1&2 were from adjacent positions on the tape, #3,4 came several weeks later, and were supplied loose. They likely came from the same reel.

All were mounted on separate pcbs, with 100uF of bypass capacitance. BC337 NPN transistors were used as sensors. These were taken randomly from the same order. The external sensors were connected with 0.5m of UTP cable. They were bundled together, with the reference thermistor, in foam. The pcbs (and therefore the internal sensors) were left open to air.

A thermometrics EC95, 10k, 0.2° accuracy thermistor was used as a reference.

The ADM1034's were running in default, ie continuous operation, averaging, reading all 3 channels in round-robin. The data was collected every second.

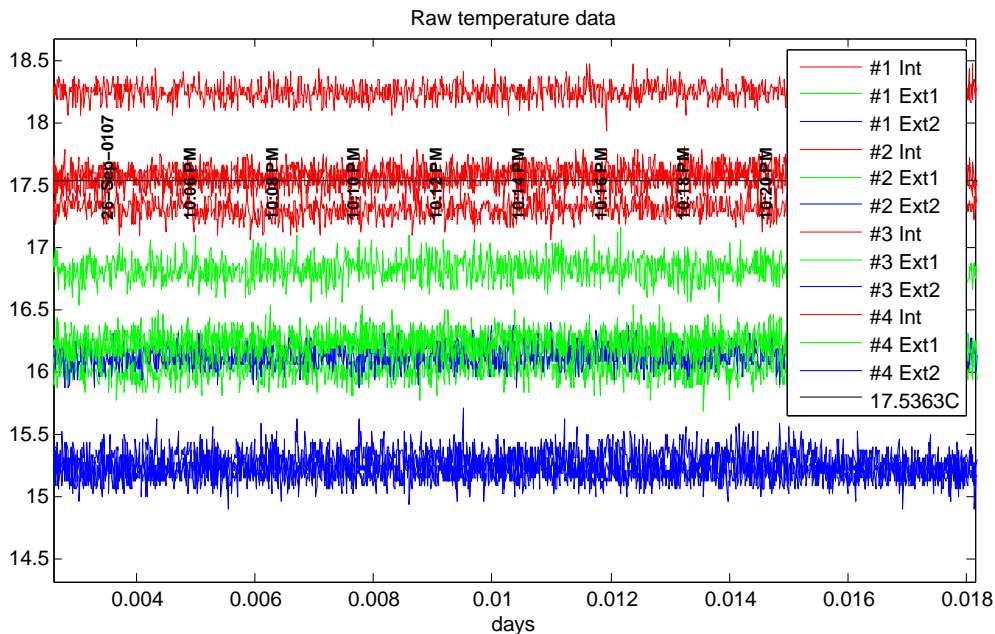
6.1 Room Temperature Measurements

Temperature during the measurement was around 17.5° by thermistor.

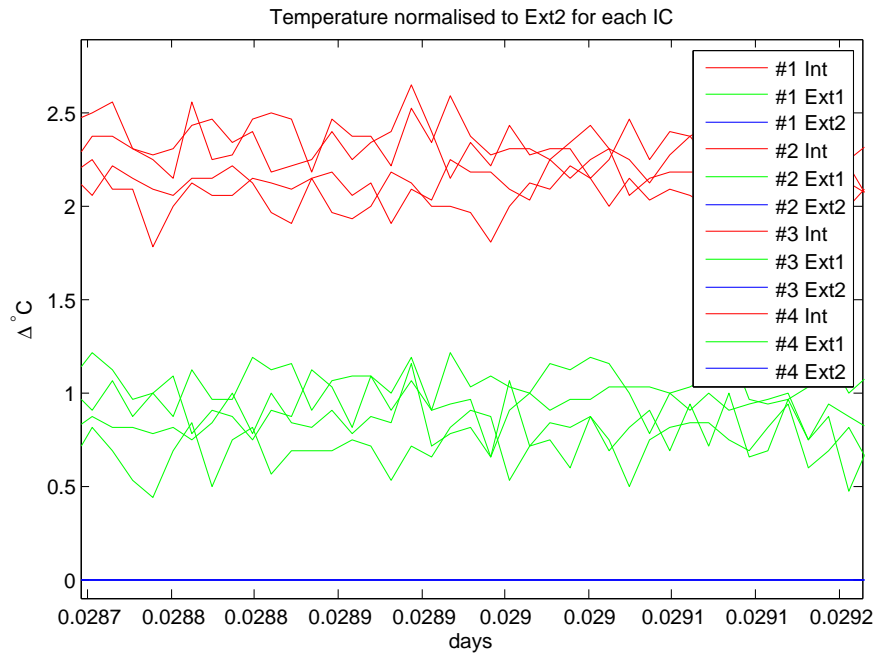
Immediately striking is the grouping of the corresponding channels on the different chips. This is especially so, since it appears that one of the chips is calibrated noticeably different to the other three.

Sensors were selectively heated, and disconnected to verify that the graph is displaying the expected sensors, and both chips.

Some of the difference between internal and external is expected due to self heating. However no systematic difference between the two external diodes is expected.



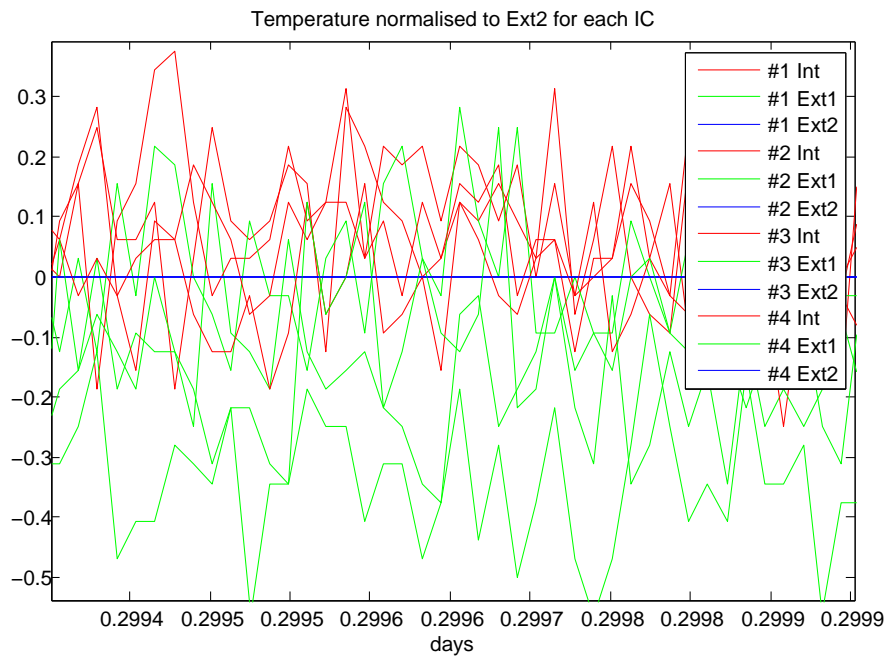
Below, we have normalised for each chips calibration error, by subtracting the Ext2 value for each IC. (So the blue Ext2 line is always 0). Here we see what appears to be a suspiciously systemic 1°C difference between the channels. Note that the Int sensor (red) is expected to be 0.35°C high due to self heating^{6.4}



So if we normalise with both the calibration error and the postulated °C difference between channels we get the graph below.

The conclusion is that

$$ActualTemp = INT + 0C^{\circ} = EXT1 + 1^{\circ}C = EXT2 + 2^{\circ}C$$



6.2 Interchanging Sensors

The external sensors were swapped between IC's. ie swap(cyan,green); swap(magenta,blue).

The effect was $<0.2^{\circ}$.

<complete>

6.3 Noise

The *rms* noise at room temperature was calculated for the 3 channels of two sensors.

Noise	IC	Internal	External D1	External D2
$^{\circ}\text{C rms}$	1	0.0686	0.1004	0.1088
$^{\circ}\text{C rms}$	2	0.0675	0.0893	0.0958

6.4 Self Heating and Internal Sensor

The internal sensor will have systematically higher temperatures due to the dissipation within the IC. This is dependent on the thermal resistance of the pcb design. Under continuous operation we can make some estimates of the range we expect.

Max: At VCC 5V with rated current drain of 3mA = 15mW, and a $T_{jA}=150\text{C/W}$, that $dT=2.25^{\circ}\text{C}$.

Min: At VCC 3V with rated current drain of 3mA = 9mW, and a $T_{jC}=39\text{C/W}$, that $\Delta T=0.35^{\circ}\text{C}$. Note that this error can be changed by altering the sample rate and averaging settings.

So far the experiments don't allow an estimate of the actual self heating error.

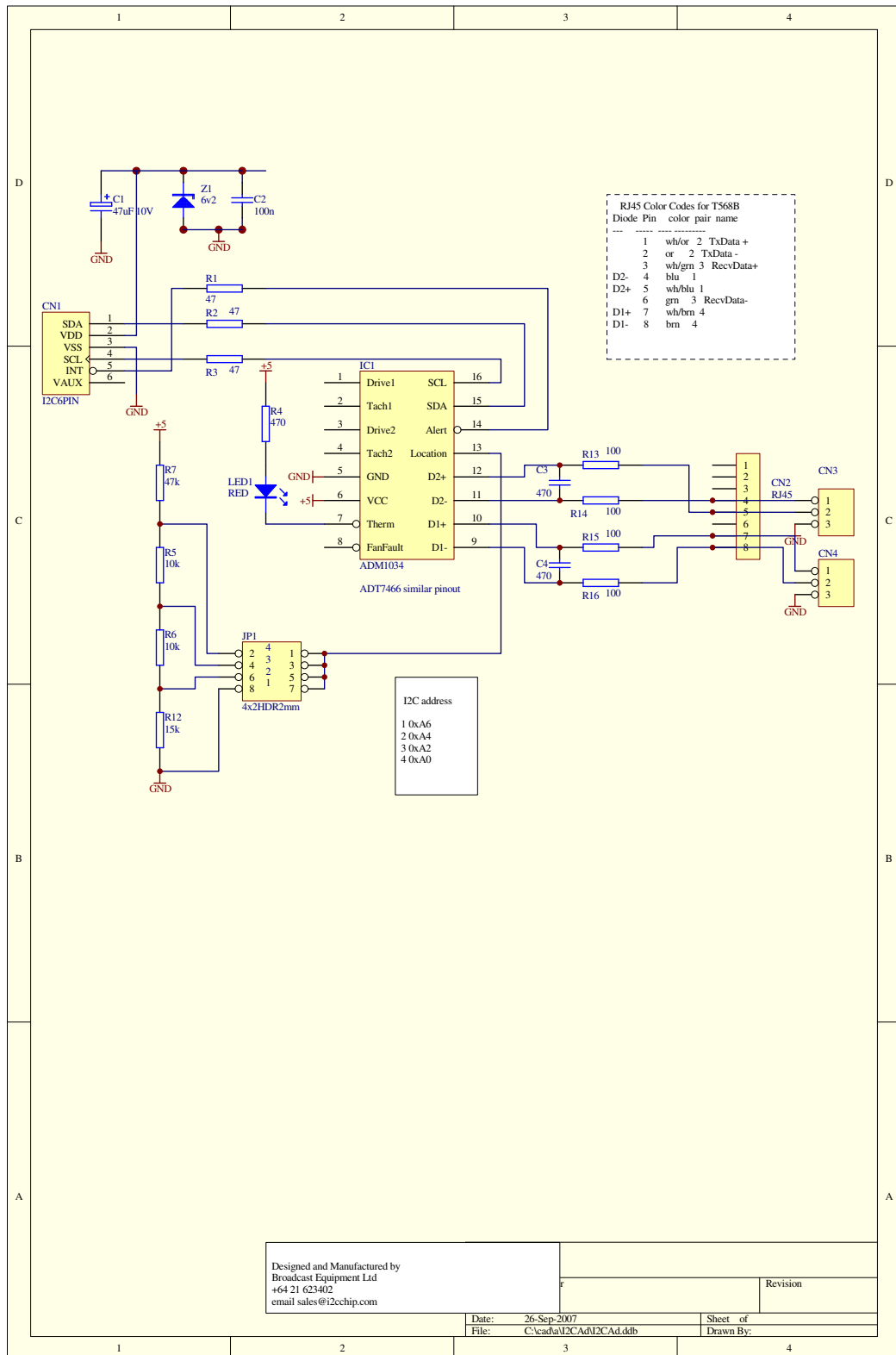
6.5 Summary of Results

Based on room temperature testing of 4 samples, likely from the same production lot, there appears to be a systematic error of 1° between successive channels in the same chip.

If this postulated error is removed, the correlation between sensors appears excellent. Noise performance is also good for integrated sensors

7 Drawings

7.1 Circuit Diagrams



Designed and Manufactured by
 Broadcast Equipment Ltd
 +64 21 623402
 email sales@i2chip.com

Date: 26-Sep-2007
 File: C:\eada\I2CA\I2CAD.ddb

Revision
 Sheet of
 Drawn By:

7.2 PCB Drawings

