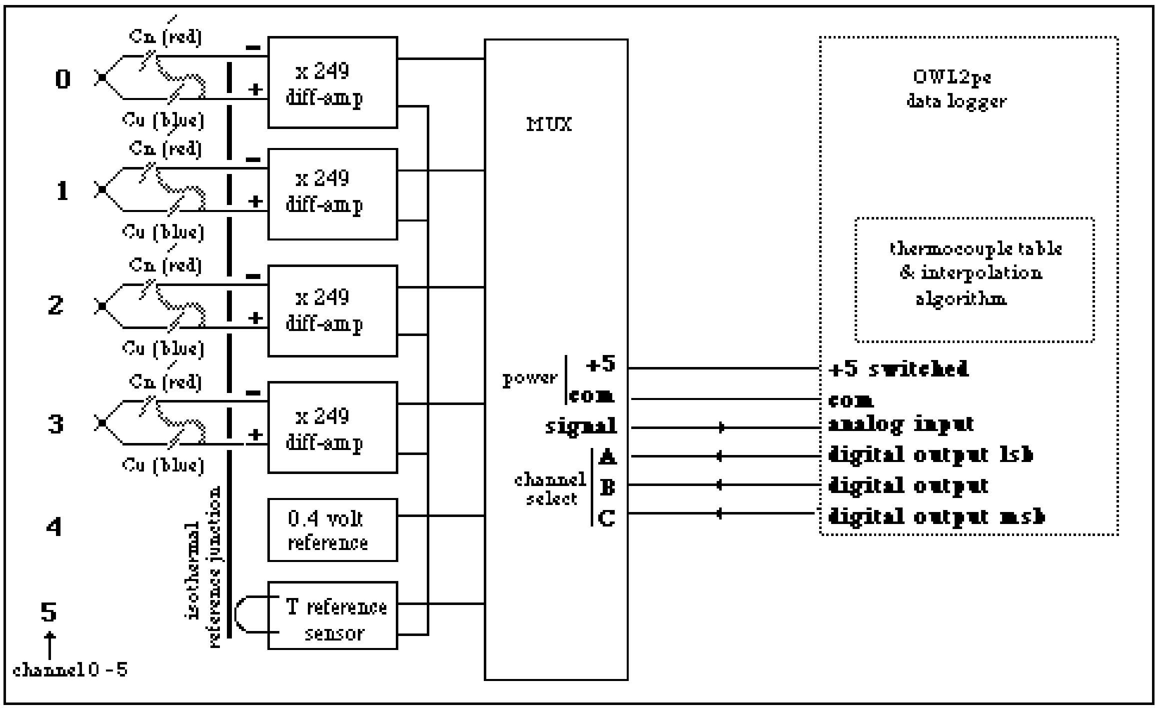
THC4b
Thermocouple conditioner/multiplexer

Thc4b is an amplifier + multiplexer for thermocouples. Thermocouples produce small signals in the range of 40 microvolts per degree, and the amplifier brings that up to a higher voltage level (100 to 250 times gain), while also rejecting noise and power line pickup. The multiplexer selects one of 4 input thermocouples under computer control and routes the amplified signal to the single analog output. The muliplexer can also select a precision reference voltage or a precision reference temperature. With this scheme, conversion of the signals to temperature of each thermocouple is accomplished in the OWL2pe (or other microcontroller) with the help of a lookup table. There is nothing in the THC4b hardware that restricts its use to one or the other type of thermocouple, so it can be used with any type with appropriate change of lookup table to use a different thermocouple or a different full scale range.



For biological applications, we recommend type T thermocouples, which have the lowest margin of error (due to the attainable purity of the metals copper and constantan), and perform well in either mildly reducing or mildly oxidizing situations. Type K thermocouples are also popular for general purpose applications and higher temperature.

Thc4b comes standard with four pairs of terminals where thermocouples can be attached. To gain access to the terminals, please remove the single screw holding on the bottom of the enclosure.
Type T:
  Blue=copper=positive (+)
  Red=constantan (nickel chromium allow) = negative (-)
Type K:
  Yellow=chromel=positive (+)
  Red=alumel=negative (-)
When attaching thermocouples to the terminals, please observe the polarity marks. In particular, note that the red wire is negative.

The 22 gage stranded thermocouple wire can be soldered together at the tip to make a measurement junction. Or, to make a very fine and light weight junction with fast response, separate out one of the smaller strands on each side and bring those out and solder them together at the tip. Remember that heat will flow up the wires, so try to keep a length of wire leading up to the tip in the medium where the measurement is to be made.

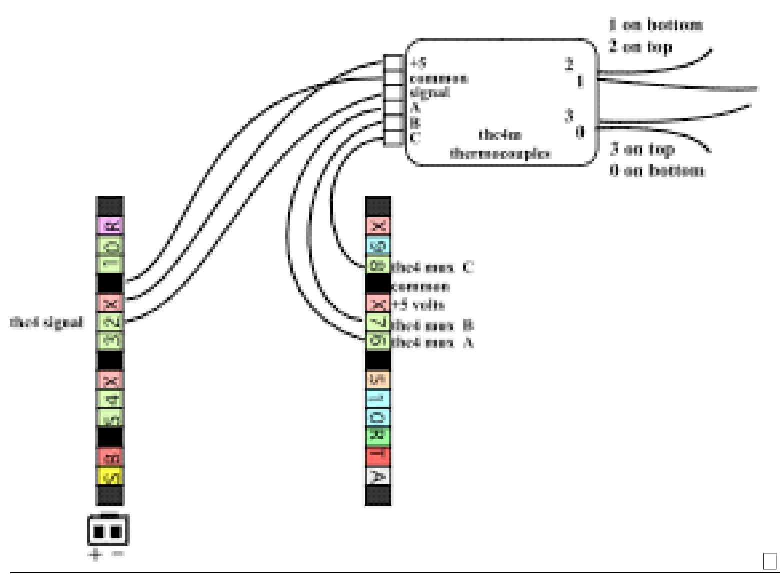
Thermocouples can also be welded, but carefully so that the properties of the metals will not be compromised. Welding is more suitable for use at higher temperatures or when the metals in solder would not be safe or appropriate. Thermocouples are available premade in a wide variety of shapes and also installed in various kinds of probe tips.

Always try to insulate the thermocouple tip, especially when it will be in contact with large wet or conductive objects. Direct contact with large objects tends to increase the noise signal, because the entire object acts like an antenna. The amplifiers in the Thc4 can reject some noise and power line pickup, but there is a limit, and the signals from thermocouples are small, at a level of around 40 microvolts per degree. A sign of poor insulation and large common mode signal is that the temperature on one channel will influence the readings on a neighboring channel.

The following are the Thc4 connections to the data logger. These connections can be made by opening the enclosure.

common -- signal and power common
power -- +5 volts switched (4.5 to 6 volts okay)
signal -- output signal from the multiplexer, goes to data logger analog input
range is zero to 2.5 volts, approximately, upper level depending on signal.
reference voltage is 0.4 volts (+/- 2 millivolts)
cold junction signal is 0.4 volts plus 10 millivolts per degree Fahrenheit (2.52 volts at 100 ' C)
A,B,C -- multiplexer channel select inputs, digital cmos levels weighted A=1, B=2, C=4
0 -- thermocouple signal #0
1 -- thermocouple signal #1
2-- thermocouple signal #2
3 -- thermocouple signal #3
4 -- reference voltage
5 -- cold junction temperature

The following diagram shows only one possible connection to theOWL2pe logger with the TB2-10 breakout board.



The OWL2pe can store the raw voltage signals from the Thc4, and offload the raw values to a spreadsheet (Excel for example) to do the calculations. Alternatively, the OWL2pe can perform the computations and then log the data directly as temperature values. The algorithm either way is as follows:

1) Read the reference voltage. It should be between 398 and 402 millivolts.
2) Read the reference temperature sensor value in millivolts and subtract the reference voltage (-400). This will be the temperature of the reference junction in degrees Fahrenheit *10. For example, reading is 1025 millivolts.

    "Fahrenheit temperature = 1025-400 = 625 ' for 62.5°F"

    Convert to units of 0.1 degrees Celsius using the formula, degC = degF * 5/9 - 178.

    Celsius temperature = 62.5 * 5/9 - 17.8 = 169 ' for 16.9 °C

    (in the OWL2pe these values are maintained as integers, 625 and 169 in the example)
3) Look up the reference junction temperature (in units of 1 °C) in the table, using the degrees Celsius value, and interpolate using the residual tenths of a degree. Example, the reference temperature is 16.9 °C. From the table, the output of a T thermocouple at 16 °C is 157 millivolts, and the output at 17 °C is 167 millivolts, and the result with interpolation is 166 millivolts.
4) Read the thermocouple voltages (amplified) in millivolts,
5) Add the millivolt compensation value determined in step (3) and subtract the offset (from step 1)

    Example: suppose the thermocouple channel reads 987 millivolts

    Then the compensated voltage is 987 +166 – 400 = .753 millivolts

    This is the voltage a thermocouple woul output with the reference junction at zero degrees Celsius.
6) Use that voltage to look the temperature up in the table. For example, in the table, 753 falls between 746 and 758, which correspond to 72 and 73 degrees Celsius. With interpolation, the temperature of that thermocouple is found to be 72.6 degree Celsius. Note that this is really a reverse lookup.
7) Average, and apply individual calibrations if necessary.

Thermocouples are nonlinear, so the easiest and fastest way to do the conversion in PBASIC is by the table lookup with interpolation. This is suited to the OWL2pe. It is necessary to have a thermocouple table prepared in memory. Usually this will be in the form of temperatures in degree Celsius increments from, say, -25 degrees Celsius up to, say, 102 degrees Celsius. Or, for a wider range with a type K couple, from -100 °C up to 1300 °C in 10 degree increments.

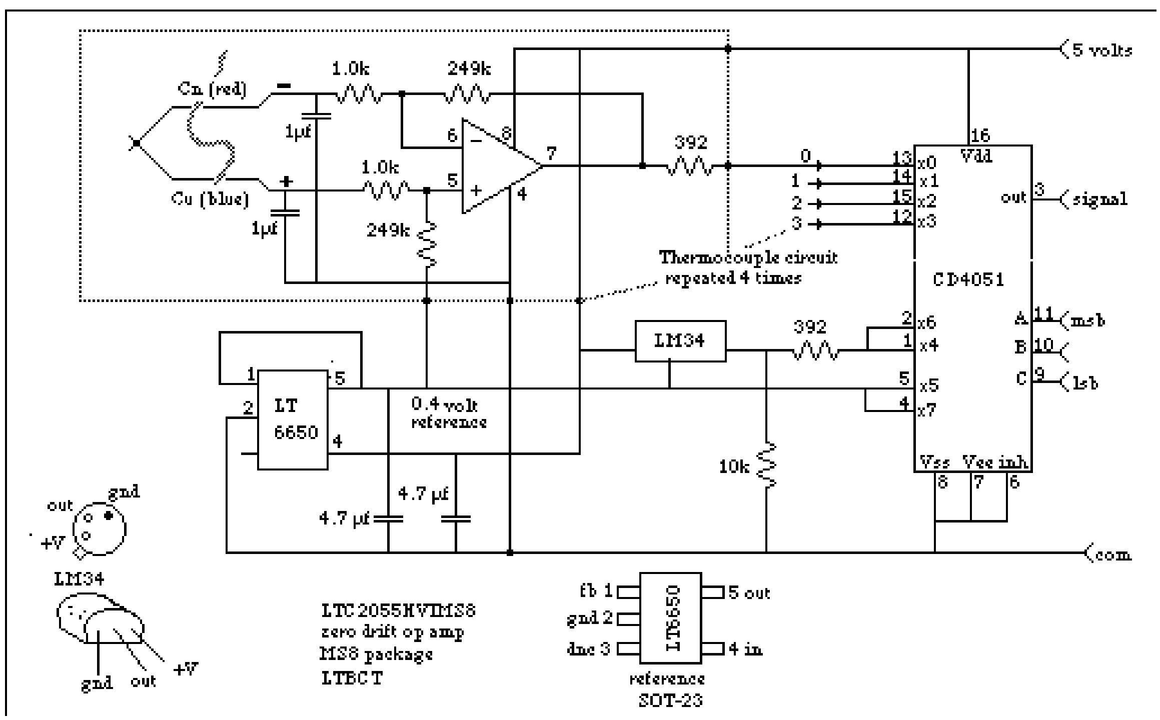
Thermocouples can be used to high temperature, which of course will need a table to match. The amplifier gain must be set to match the range desired for best accuracy. A gain of 75 will provide full scale up to the 1350 degree range of the type K thermocouple, while a gain of 100 provides improved resolution over a range from -10 degrees Celsius up to 900 degrees Celsius, with a resolution of about 0.3 degree. A gain of 250 is appropriate for -4 degrees to +350 degrees Celsius with a resolution of about 0.1 degree.

The cold junction reference temperature sensor is accurate to 1 degree Fahrenheit, and the gain of the thermocouple amplifier stage is accurate to within 1%. The largest effect on accuracy will be noise effects and the effects of uneven temperatures in the THC4b signal conditioner.

Note that when a thermocouple is at the same temperature as the reference junction, the thermocouple reading will be 400 millivolt, the thc4 constant reference voltage. Ths offset voltage allows the Thc4 accommodate negative temperatures using a single power supply.

The example program in PBASIC for the OWL2pe BASIC Stamp reads all 4 type T thermocouples and implements the above calculations. This routine with slight modifications is used in the data logging program as a subroutine that can be called from a main program and places the results in the data buffer where it can be logged at intervals.

Schematic diagram:



Alternative gains of x75 and x100 are available, in order to cover wider temperature ranges. Also, the offset can be modified from 0.4 volts to a higher value, so that lower cold temperatures can be monitored, or temperatures where the cold junction (the THC4b itself) will be kept at very cold temperatures.

Demo Program for BASIC Stamp 2
This is a program for use with type T thermocouples up to 102 degrees Celsius.

```
' demo for thermocouple math in thc4b
' reads cold junction temperature, reference voltage, and
' amplified thermocouple voltage
' after x249 amplifier (as in thc4b)
' maths compute thermocouple temperature
‘ via lookup and lookdown subroutines referencing a therm

ww VAR WORD   ' number entered by user
wx VAR WORD      ' table entry, computation
wy VAR WORD      ' table entry, computation
wz VAR WORD      ' interpolation computation
wj VAR WORD    ' index into table
coldref  VAR WORD ' thermocouple cold reference
tcold VAR coldref
vref VAR WORD
sign VAR BIT      ' 1 if value from table is less than user
value
temperature    VAR WORD
coldmv  VAR WORD
adch VAR NIB
thc4abc VAR NIB
cat VAR BYTE

Xpwr CON 12

tablesize CON 127
tablesize1 CON tablesize-1


' type T thermocouple table
' amplifier millivolts output, thermocouple voltage *249
' at temperatures from -25 degC to +102 degC
' first table entry is -25 degC
' last table entry is +102 degC
' 128 degC span for table lookup.

table    DATA WORD -234, WORD  -225, WORD -216, WORD -207
DATA  WORD -198, WORD -189, WORD -179, WORD -170
DATA  WORD -161, WORD -151, WORD -142, WORD -133
DATA  WORD -124, WORD -114, WORD -105, WORD -95
DATA  WORD -86, WORD -76, WORD -67, WORD -58
DATA  WORD -48, WORD -38, WORD -29, WORD -19, WORD -10
DATA  WORD 0   ' zero degrees, ice reference
DATA  WORD 10, WORD 19, WORD 29, WORD 39, WORD 49
DATA  WORD 58, WORD 68, WORD 78, WORD 87, WORD 97
DATA  WORD 107, WORD 117, WORD 127, WORD 137
```

```
DATA  WORD 147, WORD 157, WORD 167, WORD 177
DATA  WORD 187, WORD 197, WORD 207, WORD 217
DATA  WORD 228, WORD 237, WORD 247, WORD 257
DATA  WORD 267, WORD 277, WORD 288, WORD 298
DATA  WORD 308, WORD 319, WORD 329, WORD 339
DATA  WORD 349, WORD 360, WORD 370, WORD 381
DATA  WORD 391, WORD 401, WORD 412, WORD 422
DATA  WORD 433, WORD 443, WORD 454, WORD 464
DATA  WORD 475, WORD 486, WORD 496, WORD 507
DATA  WORD 517, WORD 528, WORD 539, WORD 550
DATA  WORD 560, WORD 571, WORD 582, WORD 593
DATA  WORD 604, WORD 614, WORD 625, WORD 636
DATA  WORD 647, WORD 658, WORD 669, WORD 680
DATA  WORD 691, WORD 702, WORD 713, WORD 724
DATA  WORD 735, WORD 746, WORD 758, WORD 769
DATA  WORD 780, WORD 791, WORD 802, WORD 813
DATA  WORD 825, WORD 836, WORD 847, WORD 858
DATA  WORD 870, WORD 881, WORD 892, WORD 904
DATA  WORD 915, WORD 927, WORD 938, WORD 949
DATA  WORD 961, WORD 973, WORD 984, WORD 996
DATA  WORD 1007, WORD 1019, WORD 1030, WORD 1042
DATA  WORD 1054, WORD 1065, WORD 1077, WORD 1084 ' 127th
word, +101degC
DATA WORD 1100
' end of type T thermocouple table
' could be extended to higher temperatures

' initialize the screen and pins
DEBUG CLS
AUXIO
HIGH Xpwr
HIGH ADpwr
MAINIO
Vref=400

' --- MAIN program loop
DO
  DEBUG HOME
  ADch=1
  thc4abc=2
  LOW thc4abc
  LOW thc4abc+1
  LOW thc4abc+2

FOR cat=5 TO 0    ' read the AD channels
  OUTS=OUTS & ~(7 << thc4abc) | (cat << thc4abc)
  PAUSE 70
  GOSUB adread
  DEBUG DEC ww,"   "
```

```
  SELECT cat
  CASE =4   ' the reference voltage
    vref=ww
  CASE = 5    ' the reference temperature
    ww=ww-vref
    coldref=ww*5/9-177
    coldref=coldref+1000 MIN 750 MAX 2020 - 1000   ' -25.0
minimum, 102 max

    DEBUG TAB,DEC ww," ",TAB, DEC coldref," "
    GOSUB coldjunction
    DEBUG TAB,SDEC wx," ",TAB,SDEC coldmv," ",TAB,SDEC wy ,"
"
  CASE <4   ' thermocouples
    GOSUB hotjunction
    DEBUG TAB,TAB,SDEC temperature,TAB,SDEC wx," ",TAB,SDEC
ww," ",TAB,SDEC wy," "
  ENDSELECT
  DEBUG CR
  NEXT
  PAUSE 300
LOOP

show1dec:
  DEBUG REP "-"\wx.BIT15,DEC ABS wx/10,".",DEC1 ABS wx
  RETURN

' This is a reverse table lookup, binary search algorithm
lookback:
  wy=DCD NCD tablesize / 2
  wj=wy-1
'  DEBUG CR,DEC wj,TAB,DEC wy
index_check1:
  DO
    READ wj MAX tablesize1 * 2 + table, WORD wx
    IF wx=ww THEN EXIT
    wy=wy/2
    IF ww+32768>wx+32768 THEN wj=wj+wy:sign=1 ELSE wj=wj-
wy:sign=0
'    DEBUG DEC wj,TAB, DEC wy,TAB, SDEC wx ,TAB,IBIN
sign,CR
  LOOP WHILE wy ' wy is halved each time through
index_check8:
  IF ww+32768>=wx+32768 THEN
    READ wj+1 MAX tablesize1 * 2,WORD wy
    wj=wj MAX tablesize1
  ELSE
    wy=wx
    wj = wj MIN 1 - 1
```

```
    READ wj * 2,WORD wx
  ENDIF
  ' exits with interval containing value
  ' wx  <= ww < wy
  ' wj is index of wx
  RETURN

coldjunction:
  READ (coldref+250/10)*2+table,WORD wx   ' table entries
  READ (coldref+260/10)*2+table,WORD wy
  coldmV=(coldref+1000//10) *(wy-wx)/10+wx  ' interpolation
  RETURN

hotjunction:
' enter with ww in millivolts
' Vref from Thc4 (~400 mV)
' coldmV millivolts at temperature of cold junction
  ww = ww - Vref + coldmV
  GOSUB lookback
  ' returns with wx < ww < wy, wj is index of wx
  ' now interpolate temperature  to 0.1 degree
  ' wj-25*10 is tenths of a degree, now fill in tenths
digit
  temperature = (ww-wx)*10 / (wy - wx) + (wj-25*10)
  RETURN

' --- Analog to digital converter -----------
' ---- AUXIO pin assignments ----
ADcs     PIN 4         ' select ADC active low
ADsdo    PIN 5         ' shiftin from ADC
ADsdi    PIN 6         ' for shiftout to ADC
ADpwr    PIN 7         ' X7 power to 4.096 reference, LM50
ADsck    PIN 14
ADread:      '
  AUXIO
  ww=0
  FOR wj=1 TO 8  ' average 8 samples from each probe
  LOW ADcs
  SHIFTOUT ADsdi,ADsck,MSBFIRST,[ADch<<8\12]   ' channel 0
  LOW ADsdi
  SHIFTIN  ADsdo,ADsck,MSBPRE,[wx\12]
  ww=ww+wx
  HIGH ADcs
  NEXT
  ww=ww/8
  MAINIO
RETURN
```