

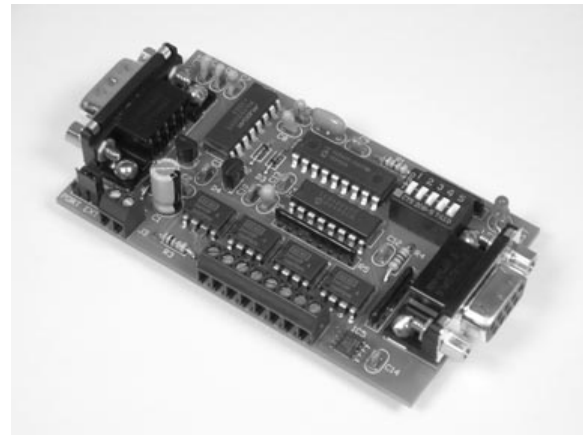
Thermocouple Input Module

FEATURES

- 4 thermocouple input channels.
- DIP switch addressable; stack up to 32 modules on the same port for 128 inputs.
- Accepts J, K, T, E (isolated) thermocouples.
- Each channel individually selectable to read temperature in Fahrenheit or Celsius.
- Temperature conversions use high-order polynomial equations eliminating look-up table interpolation errors.
- Incorporates on-board temperature sensor; no ice-point reference necessary.
- Thermocouple inputs protected to ± 40 volts.
- Software calibrated; no trim-pots to adjust.
- All user configuration and calibration data stored in non-volatile memory.
- Industry standard RS-232 interface. Meets all EIA/TIA-232E and V.28 specifications.
- Wide power supply range (8 to 30 VDC).
- Screw-terminal connectors used on all inputs.

DESCRIPTION

Connects to the RS-232 serial port of a PC, laptop, or other host. Attach up to four separate thermocouples of various types and use the host to read the temperatures of each. Simple command strings sent from the host will initiate a conversion on an input channel which in turn will, read the thermocouple voltage in μV , convert it to temperature, then add the cold junction reference temperature. The results returned to the host will be equal to the absolute temperature of the thermocouple tip and can be listed in degrees Celsius or Fahrenheit. This module is ideal for wide range temperature measurement in harsh industrial environments using a PC or laptop for data collection.



SPECIFICATIONS

Thermocouple Inputs	4 differential channels
Temperature Range	-210°C to +1372°C -346°F to +2502°F (see Table 4)
Resolution	1°C or 1°F selectable
Accuracy	$\pm (0.1\% + 2 \text{ degrees})$
Over Voltage Range	Each input protected from -40V to +40V
Calibration	Gain and zero offset corrected in firmware
Processor	PIC16F628
Clock	4 MHz
Communications	9600 Baud, N, 8, 1
Power Requirements	+8 to +30 VDC
Current Draw	18 to 20 mA
Operating Temperature	0°C to +80°C
Board Dimensions	3.5" x 2.0" x 0.7"
Weight	2.0 oz

TABLE 1: ADDRESS SETTING

HEADER CHARACTER ASCII (HEX)	DIP SWITCH SETTING 1=on, 0=off 1 2 3 4 5
A (41)	0 0 0 0 0
B (42)	0 0 0 0 1
C (43)	0 0 0 1 0
D (44)	0 0 0 1 1
E (45)	0 0 1 0 0
F (46)	0 0 1 0 1
G (47)	0 0 1 1 0
H (48)	0 0 1 1 1
I (49)	0 1 0 0 0
J (4A)	0 1 0 0 1
K (4B)	0 1 0 1 0
L (4C)	0 1 0 1 1
M (4D)	0 1 1 0 0
N (4E)	0 1 1 0 1
O (4F)	0 1 1 1 0
P (50)	0 1 1 1 1
a (61)	1 0 0 0 0
b (62)	1 0 0 0 1
c (63)	1 0 0 1 0
d (64)	1 0 0 1 1
e (65)	1 0 1 0 0
f (66)	1 0 1 0 1
g (67)	1 0 1 1 0
h (68)	1 0 1 1 1
i (69)	1 1 0 0 0
j (6A)	1 1 0 0 1
k (6B)	1 1 0 1 0
l (6C)	1 1 0 1 1
m (6D)	1 1 1 0 0
n (6E)	1 1 1 0 1
o (6F)	1 1 1 1 0
p (70)	1 1 1 1 1

STACKABLE DATA MODULES

All modules in this series incorporate two EIA/TIA-232E serial ports which communicate at 9600 baud, no parity, 8 data bits and 1 stop bit. DB9 connectors are jumpered to satisfy hardware handshaking. The port labeled "HOST" is configured as a DCE device and should be connected to a PC's serial port. The port labeled "SLAVE" is a DTE device and can be left open, or connected to another module's host port. Up to 32 modules can be chained together in this fashion to form a network. Either plugged together end to end, or separated by a cable. Because a module contains two individual bi-directional ports which pass data through, it also acts as a repeater, extending the total allowable length of the RS-232 communications line.

A modem can serve as the host for remote operation, but since a modem uses a DCE port, a "null modem" adapter must be placed between the modem and the data module's host port. A gender changer may also be required. In addition, any hardware/software flow control must be disabled in the terminal program.

Each module in a network should be set to a different address using the on-board 32-position DIP switch. A module will only respond to data packets that begin with its' own unique header character, which is determined by this DIP switch setting. See Table 1. Data packets transmitted by a module will also begin with this header character. The host PC can use the header character to address each individual module in a network, and to identify a module which is talking.

COLLISION CONTENTION

The utilization of the communications line can be thought of more as a single, bi-directional, data bus, operated in a multi-drop mode rather than a standard RS-232 data link. A transmission from a data module travels in both directions, upstream to the host, and downstream to signal other modules that it has seized the line. Before transmitting, a module will listen to the communications line and wait for quiescence. After a silent period equal to the length of one byte, the waiting module will send its data packet using a Carrier Sense Multiple Access with Collision Detection communications protocol. See the application note (AN100) at the back of this manual for more details.

COMMAND SET

The host PC communicates with the Thermocouple Input Module using a command set comprised of standard ASCII character strings as depicted in Table 2. Some commands are used to setup and configure the individual channels while others are used to poll for data. A detailed description of each command follows.

READ - Reads the current value of the specified input channel. The results returned to the host will be equal to the absolute temperature of the thermocouple sensor which is attached to the input terminals of that channel. The temperature will be displayed in the degree units (Fahrenheit or Celsius) which had previously been selected by using the UNITS command. The acquisition time will be approximately 70 mS. The communications port will be disabled until acquisition is complete and the results are returned to the host.

TYPE - Sets the thermocouple type for a specific input channel. Each channel must be set to the same type as the thermocouple which is attached to its inputs. Choices are J, K, T or E

and is determined by the types of metals which are used to construct the thermocouple sensor. Those metals consist of the following:

- J = Iron (+), Constantan (-)
- K = Chromel (+), Alumel (-)
- T = Copper (+), Constantan (-)
- E = Chromel (+), Constantan (-)

Consult the thermocouple manufacturer to determine which type or types you are using. It is perfectly acceptable to mix different thermocouple types on the same WTTCI board provided that each channel has its type value set accordingly.

UNITS - Sets the degree units for a specific thermocouple input channel. Choices are Fahrenheit or Celsius and will determine the format of the temperature data which is returned to the host when issuing the READ command for that channel. Note, this also sets the degree units used in the CALIBRATE command string.

CALIBRATE - Takes the actual measured values of two temperature points and computes the calibration coefficients for a specific

TABLE 2: COMMAND SET

TITLE	COMMAND	DESCRIPTION
READ	R <i>chn</i>	Read the current value of a thermocouple input channel. <i>chn</i> = A-D. Returns the thermocouple temperature listed in the degree units which has been selected by UNITS.
TYPE	T <i>chn value</i>	Sets the thermocouple type for a specific input channel. <i>chn</i> = A-D, <i>value</i> = J, K, T or E, listed in accordance with ANSI conventions. Default = J. If <i>value</i> omitted, reads the current setting. (Note 3)
UNITS	U <i>chn value</i>	Sets the degree units for a specific thermocouple input channel. <i>chn</i> = A-D, <i>value</i> = F or C. F = Fahrenheit, C = Celsius, default = F. If <i>value</i> omitted, reads the current setting. (Note 3)
CALIBRATE	C <i>chn t1 - t2</i>	Takes the actual measured values of two temperature points and computes the calibration coefficients for a given thermocouple input channel. <i>chn</i> = A-D, <i>t1</i> is the measured temperature when a 1000° calibration signal is applied to the input, <i>t2</i> is the measured temp when a 100° calibration signal is applied to the input. (Note 3, 4)
ERROR	?	This character will be returned after an invalid command or variable.
RESET	!	This character will be returned after a power-on reset, or brownout.

Note 1: All command strings sent to the data module should be preceded with the header character (see Table 1), and terminated with a carriage return. All responses from the data module will also appear in this format.

Note 2: Any spaces shown above in the listing of the command strings are for clarity only. They should not be included in the actual transmission from the host, nor expected in a response from the data module.

Note 3: After successful execution, this command will be echoed back to the host in the same format as received.

Note 4: The data module has been calibrated at the factory, it is not necessary to perform this operation prior to use.

TABLE 3: TERMINAL / CONNECTOR DESCRIPTION

NAME	TYPE	ELECTRICAL SPECS	COMMENTS:
HOST	DB9 (female)	EIA/TIA-232E Standard	RS-232 serial port configured as DCE. Connects to host PC. Hardware handshake jumpered.
SLAVE	DB9 (male)	EIA/TIA-232E Standard	RS-232 serial port configured as DTE. Can be connected to another data module's HOST port for networking.
Power Source	Jumper	N/A	Power source selection jumper. Selects either external, or port powered. (Note 1)
+	Screw Term	+8 to +30 VDC	External unregulated power supply input.
-	Screw Term	GND	External power supply ground.
A - D	Screw Term	Range = +2 to +3V (biased @ 2.5) Max input = -40V to +40V	Input channels which can accept isolated or ungrounded thermocouples only.
Note 1: Selecting "port powered" will draw from the power supply source of an upstream data module. Caution, the COM port of a PC or laptop does not supply enough current to serve as the power supply source.			

thermocouple input channel. To calibrate, apply a precision 1000° calibration signal to the input channel of the WTTCI. Use the READ command to measure the temperature and record the average reading of multiple samples. Then repeat this process using a 100° calibration signal. Include the results in the CALIBRATE command string using a hyphen to separate the two numbers. The WTTCI will use this data to calculate the gain and offset coefficients particular to that channel and store it in non-volatile memory. If necessary, repeating this process will zero in the calibration accuracy, and a deviation of 1 degree or less across the full temperature range can be achieved.

IMPORTANT, be sure to set the TYPE and UNITS for the channel appropriately before calibrating. Both should be set to the same as the calibration signal source. Once calibrated, the TYPE and UNITS can then be changed to any of the other choices without having to recalibrate. Note, the module has been calibrated at the factory, it is not necessary to perform this operation prior to use.

ERROR - Any data string sent from the host containing the correct header character but an invalid command or variable will be responded to with this error indicator.

RESET - Upon power-up or any other reset condition, this indicator is transmitted to the

host. Note, all user configuration and calibration data is stored in non-volatile memory. Therefore, a reset or loss of power will not corrupt these settings.

OPERATION

To hook the data module to a host PC, use a standard RS-232 cable with male and female DB9 connectors on opposite ends. This cable should be wired straight through (pin to pin) with no crossover of the data lines. In other words, not a null modem cable. Connect a suitable DC power source to the + and - terminals of the data module. It is highly recommended to use an ungrounded AC adapter such as that which is available from Weeder Technologies. This will provide isolation and prevent ground loops which are commonly created if the power supply and computer are grounded at different points.

When the data module is first powered up, the red LED will flash briefly. This indicates that the on-board microcontroller has booted up, successfully completed its internal diagnostic test, and has transmitted the reset character to the host to signal that it is up and running. The red LED will also flash anytime the module receives or transmits any data packet, thus making it a valuable diagnostic tool when troubleshooting communications problems.

An easy-to-use Windows™ software package called "ModCom" is available and can be downloaded from Weeder Technologies' web site. This program will allow the user to quickly set up custom buttons which transmit commands, custom windows that poll for data, and a variety of other screen objects such as slider controls, event counters & timers, bar-graph level indicators, button selection arrays, signal lamps, and more. In addition, conditional statements can be set up to take action when specific events or conditions are met, sequences can be written and then called by other screen objects during run-time, and data can be logged to a file automatically at user-defined intervals.

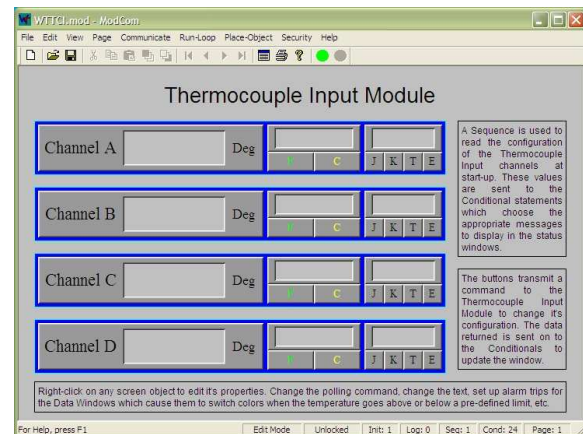
Once ModCom is installed and running, go to the <Communicate> menu item at the top of the screen and click on <Send/Receive>. A dialog box will pop up which you can use to type in the commands from Table 2, transmit them directly to the data module, and see the response coming back. Use this dialog box to familiarize yourself with the command set and to experiment with the various features supported by the module. The experience gained here is significant since these are the same command strings you will use when setting up the other objects in ModCom.

To read thermocouple temperatures, start with the sample application "WTTCI.mod" which can be found in the ModCom subfolder called "Samples". After this file is opened, it will appear as shown in Figure 1. To start the main run-loop, click on the green toolbar button at the top of the screen. At this time the Data Windows will begin polling for data and show the temperature readings of the thermocouples which are attached to each input channel of the WTTCI. Note, any input channel which is left open, may eventually float up to saturation and display a reading of "9999", but this is not guaranteed.

You can use the buttons to the right of each Data Window to change the display format from Fahrenheit to Celsius or vice versa, and to select the thermocouple types. The selections you make with these buttons will be stored in the WTTCI's memory and remain there even if power is cycled.

Each Data Window can be customized by the user for appearance or even set up to signal an alarm condition by changing colors if the temperature exceeds a predefined high or low limit. To do this, first halt the run-loop by clicking

FIGURE 1: MODCOM APPLICATION



on the red toolbar button at the top of the screen. Then right-click on the object to pull up its properties dialog box. Within this box are a number of options which can be modified by the user. For more details, read through the help files which can be accessed by clicking on the <Help> menu item at the top of the screen.

The temperature shown in each Data Window can also be logged to a file. Go to the <Run-Loop> menu item at the top of the screen and click on <Data Logging>. Within this dialog box you can specify the data log file name, set the time interval between log events, and select those screen objects that you wish to log to file. The contents of the file can be viewed or printed by clicking on the corresponding button shown in this dialog box, or by clicking on a toolbar button at the top of the main screen.

TEMPERATURE CONVERSIONS

The temperature versus voltage relationship of the output of a typical thermocouple is not linear. Therefore, simply reading the voltage and multiplying it by a scaling factor will not convert it to temperature, or at least not with any degree of accuracy over a broad range. Because of this, a very common design approach for converting the voltage to temperature is to use a lookup table of known voltages that represent various temperatures, and then use interpolation for those voltages that fall in-between the table listings. This is fine for low precision applications but because of the gaps in the lookup table, interpolation errors can not be avoided and thus limits the reliability of the system.

The WTTCI however uses a different approach. The signal from the thermocouple is amplified,

TABLE 4: TEMPERATURE RANGE

TYPE	CELSIUS	FAHRENHEIT
J	-210° to +1200°	-346° to +2192°
K	-200° to +1372°	-328° to +2502°
T	-200° to +400°	-328° to +752°
E	-200° to +1000°	-328° to +1832°

converted to a digital format, and then subjected to a high-order polynomial equation using 32-bit floating point math. The result is a voltage to temperature conversion with an accuracy of 0.1°C across the entire range of temperatures. Built in to the WTTCI's firmware is the polynomial coefficients published by the United States National Institute of Standards (NIST) which are needed for voltage to temperature conversions for each of the four thermocouple types that the module supports. When a READ command is issued by the host, the appropriate NIST coefficients are extracted and plugged into the mathematical equation mentioned above.

TEMPERATURE RANGE

The range of temperatures which can be measured by the WTTCI and its thermocouple sensors is dependant on two factors, the physical limitations of the thermocouple itself, and the mathematical boundaries inherent to the polynomial equation that is used to calculate the temperature. The former specifications can be obtained from the thermocouple manufacturer and will vary depending on the form of weld used to make the junction, and the type of insulating material used to cover the wires. As for the limits of the polynomial equation used for temperature conversions, refer to Table 4.

THERMOCOUPLE GROUNDING

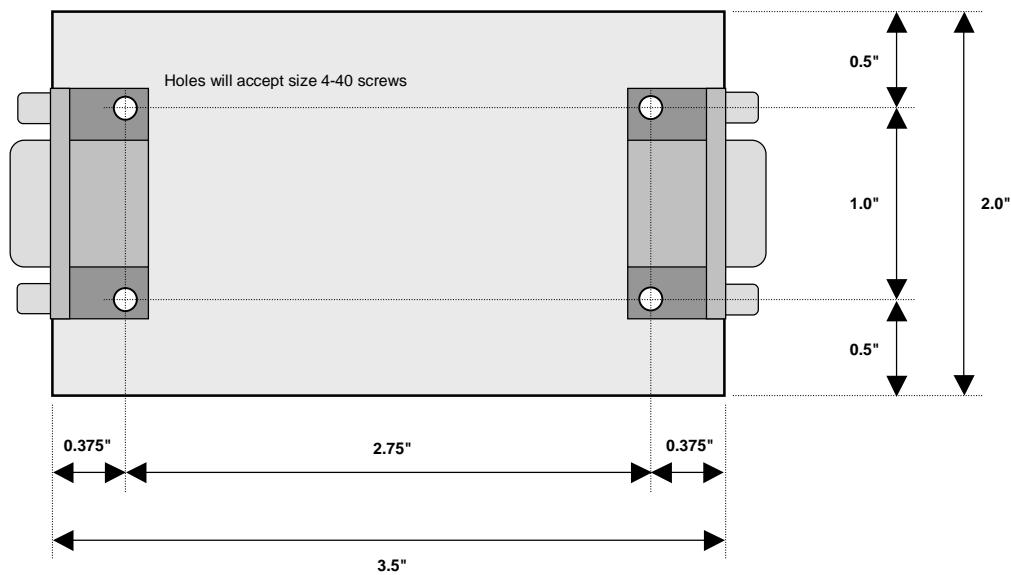
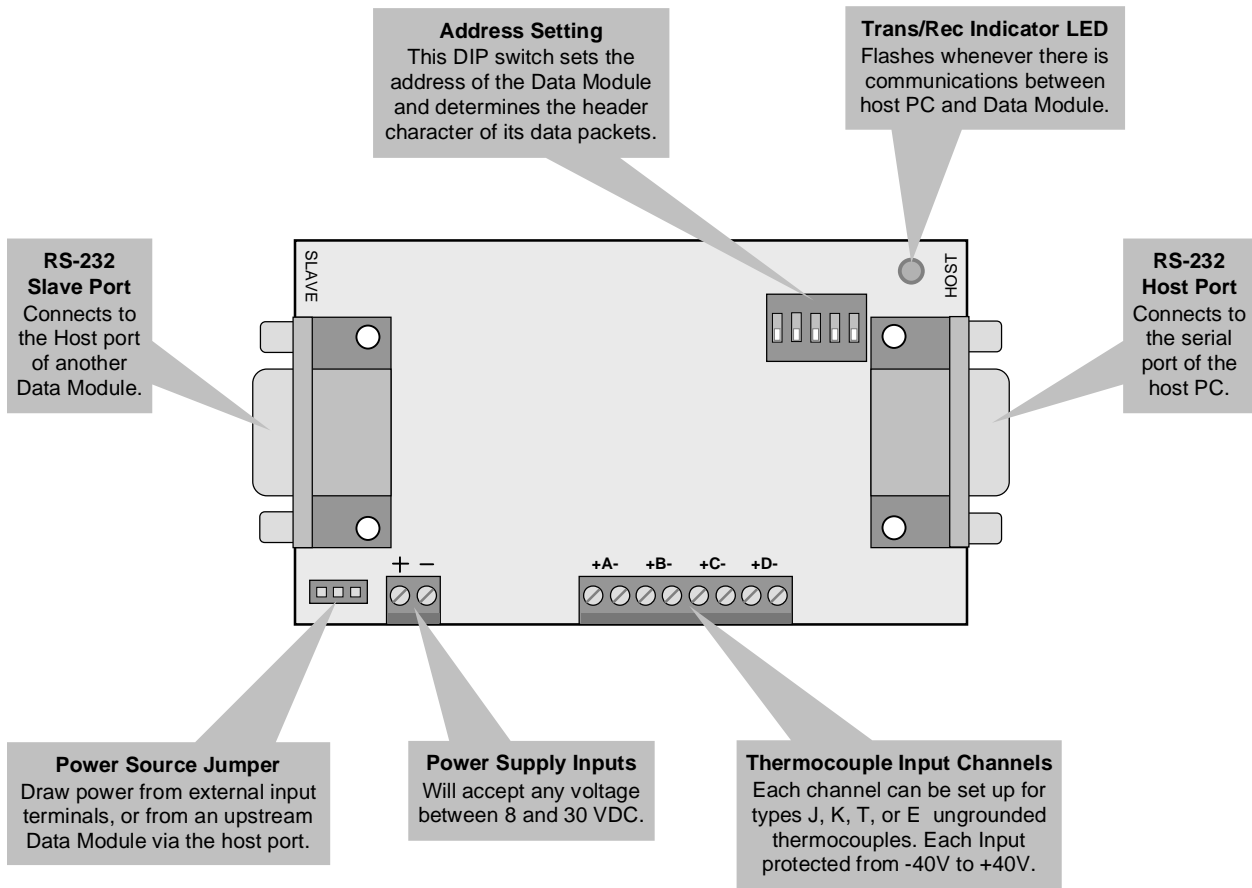
Although the WTTCI benefits from a wide range of input voltages that it can safely handle before damage results (± 40 volts), the common mode voltage range is quite small. When attached to a thermocouple that is electrically isolated, this poses no problem because the wires of the thermocouple will float to the bias point which is midway in the common mode region. However, if the thermocouple tip or junction is grounded (whether accidentally or intentionally) to the housing which it is mounted to, and that housing has a voltage potential beyond the common

mode range of the WTTCI's inputs, errors in temperature readings will result. Care must be observed to be certain that the thermocouple tip does not make electrical contact with any conductive surfaces it may be mounted to.

When using a prefabricated temperature probe, one where the thermocouple sensor is enclosed in a metal tube for instance, most are offered in a choice of configurations, "grounded", "ungrounded", or "exposed" junctions. Avoid those which are listed as "grounded", unless of course, you plan on mounting the probe to a structure or framework which is either non-conductive, or in itself electrically isolated. Consult the probe manufacturer for details on how to identify those units which are grounded and which are not. Note, some thermocouple probes have a metal shield or braid covering the length of wires, and if this probe uses a "grounded" junction configuration, the metal braid will also be grounded to the thermocouple tip and may cause problems if it comes into contact with any metal structure or framework.

If wishing to use an existing thermocouple mounted in the field which you are uncertain of its grounding state, perform the following steps to determine if it is isolated or at least within the common mode range of the WTTCI.

Attach the thermocouple wires to an input channel of the WTTCI and apply power to the module. Using a voltmeter, connect the black probe to power supply ground, and the red probe directly to the screw terminals which are holding the thermocouple wires. You should read a voltage of +2.5V plus or minus 0.5V. If not, the thermocouple tip is grounded and the voltage potential of that ground is outside the range of the WTTCI's inputs. In this case, the thermocouple must be removed from its original mounting and re-mounted in a way which keeps it electrically isolated from the frame that it is attached to, or replaced with a thermocouple probe using an "ungrounded" configuration.



Thermocouple Input Module
WTTCI-M

