

## Using the Dataforth 5B37 non-linearized thermocouple isolation module with the MSXB027 expansion board

When a Dataforth 5B37 thermocouple isolation module is used with a Microstar Data Acquisition Processor (DAP), additional processing is necessary to linearize the output into meaningful temperature data. If a 5B47 linearized thermocouple isolation module is used instead of the 5B37, the linearization will be performed by the module and this tech note will not be necessary.

### THERMO for Linearization

All thermocouples follow a nonlinear voltage vs. temperature curve which differs depending on the thermocouple material used. DAPL provides a command, THERMO, which compensates for this curve. The input data is converted directly into tenths of a degree Celsius. Application 29 in the Applications Manual (Apps2000.hlp, Apps2000.pdf) explains the use of the THERMO command in detail.

### 5B37 Scale and Offset

THERMO cannot use values obtained from a 5B37 module directly because the module applies a scaling factor and offset as described in the Dataforth catalog Application Note AN501. AN501 also provides a formula used to convert from the voltage at the output of the 5B37 to the true thermocouple voltage:

$$V_t = (V_{out} / G) + V_{offset}$$

Where:

- $V_t$  = the thermocouple voltage
- $V_{out}$  = the 5B37 module's output voltage
- G = the gain applied by the 5B37 module
- $V_{offset}$  = offset voltage applied by the 5B37 module

The values for G and  $V_{offset}$  for different thermocouple types can be found in AN501.

### Converting 5B37 Values for use by THERMO

As can be seen in the Application Manual examples, the typical data used by the THERMO command are the raw thermocouple voltages amplified with a gain of 100 and then digitized using the DAP's A/D converters. A scale factor of 5000/32768 is applied in order to convert this digitized value into the "tens of microvolts" required by THERMO (see the DAPL Manual for the full description of the THERMO command). These typical gain and scale factors will be used to provide a formula for a backwards conversion between the 5B37 digitized output values and the corresponding integer values needed by THERMO.

1: Apply a gain of 100 to the formula above:

$$100 * V_t = 100 * (V_{out} / G) + 100 * V_{offset}$$

2: "Digitize" the voltages to 16-bit values assuming +/-5V full scale. A different full scale input range will result in a different value for A.

Let  $A = (65536/10)$  = voltage to digitized value conversion factor

$$A * 100 * V_t = A * 100 * (V_{out} / G) + A * 100 * V_{offset}$$

Let  $D_{THERMO} = A * 100 * V_t$  = The integer data expected by THERMO

Let  $D_{5B37} = A * V_{out}$  = The integer data provided by the 5B37

Let  $OFS = A * 100 * V_{offset}$  = The integer offset to be applied to the digitized 5B37 data

Note: if the DAP is configured for unipolar inputs, the calculated OFS value will need to be divided by two.

3: Rewrite the previous equation as:

$$D_{THERMO} = 100 * D_{5B37} / G + OFS$$

## DAPL Example

The following DAPL example illustrates how this conversion technique can be applied using a type K thermocouple module. The GAIN value is taken directly from Dataforth's Application Note AN501 and rounded to the nearest integer. The OFS value is the "Thermocouple neg. full scale in volts" value given by Dataforth multiplied by 100 and again by the voltage-to-integer conversion factor "A" (= 65536/10). The GAIN and OFS values necessary for other supported thermocouple types can be calculated using the values found in AN501. The LONG pipe PKL is used to extend the data to 32-bit integers to preserve the accuracy of intermediate calculations. The temperature is displayed directly as tenths of a degree.

```
RESET
```

```
PIPE PK, PTEMP  
PIPE PKL LONG
```

```
;Type K Constants  
CONST GAIN = 87  
CONST OFS = -2329  
CONST TYPE = 2
```

```
;Sample one Type-K 5B37 module  
IFDEF A 1  
  SET IP0 S0  
  TIME 10000  
END
```

```
PDEF B  
  PKL = IP0  
  PK = ((100 * PKL) / GAIN) + OFS  
  THERMO(PK, TYPE, 5000, 32767, PTEMP)  
  FORMAT (PTEMP)  
END
```