Allen-Bradley

Barrel Temperature Control Module

(Cat. No. 1746-BTM)

User Manual

Important User Information

Because of the variety of uses for the products described in this publication, those responsible for the application and use of this control equipment must satisfy themselves that all necessary steps have been taken to assure that each application and use meets all performance and safety requirements, including any applicable laws, regulations, codes and standards.

The illustrations, charts, sample programs and layout examples shown in this guide are intended solely for purposes of example. Since there are many variables and requirements associated with any particular installation, Allen-Bradley does not assume responsibility or liability (to include intellectual property liability) for actual use based upon the examples shown in this publication.

Allen-Bradley publication SGI-1.1, *Safety Guidelines for the Application, Installation, and Maintenance of Solid-State Control* (available from your local Allen-Bradley office), describes some important differences between solid-state equipment and electromechanical devices that should be taken into consideration when applying products such as those described in this publication.

Throughout this manual we use notes to make you aware of safety considerations:

ATTENTION: Identifies information about practices or circumstances that can lead to personal injury or death, property damage or economic loss.

Attention statements help you to:

- identify a hazard
- avoid the hazard
- recognize the consequences

Important: Identifies information that is critical for successful application and understanding of the product.

Summary of Changes

This manual has been revised extensively. Major changes include:

- The Quick Start (chapter 1) was removed and made a separate publication (1746-10.4), packaged with the BTM module.
- A detailed autotune procedure was added.
- The description and examples of implied decimal point were revised.
- The listings of channel data were revised.
- The revised sample program and this manual are now on the internet. You can access them from our website at: http://www.ab.com
  with extensions to:
  – program: /mem/appsys/prodinfo/applac/appla/BTMsw/
  – manual: /manuals (Application Systems Library, pub 1746-6.10)

We improved module performance with these firmware adjustments:

- Cool-only applications in non-barrel mode

Important: For an abbreviated outline to help you set up and operate this module for the first time, refer to the 1746-BTM Quick Start, publication 1746-10.4 (April 1998). It is currently packaged with the module.

Notes:

Using this Manual

This manual shows you how to use the Barrel Temperature Control Module (cat. no. 1746-BTM) in an Allen-Bradley SLC® system for barrel temperature control and other injection molding or extrusion-related temperature control applications. The manual explains how to install, program, calibrate, and troubleshoot the BTM module.

ATTENTION: Use the 1746-BTM module in a local I/O chassis only for barrel temperature control of injection molding applications or extruders. Any other applications are not supported.

Audience

You must be able to program and operate an Allen-Bradley SLC programmable controller to make efficient use of this module. In particular, you must know how to configure M0 and M1 files. For more information, see the appropriate SLC programming manual before you generate a program for this module.

System Compatibility

System compatibility involves data table use as well as compatibility with a local I/O chassis and SLC processor.

Data table

Communication between the module and processor is bi-directional. The processor transfers output data through the output image table to the BTM module and transfers input data from the BTM module through the input image table. The BTM module also requires M files for configuration and calibration values.

I/O chassis

You can use this module with 1746-A4, -A7, -A10, or -A13 chassis, provided there is an SLC controller in the chassis (local system). You can place the BTM module in any I/O slot except for the first slot which is reserved for the processor.

SLC Processor

The 1746-BTM module is compatible with any SLC processor that supports M0/M1 files, such as the SLC 5/05, SLC 5/04, SLC 5/03, and SLC 5/02 controllers.

Vocabulary

In this manual, we refer to:

- the barrel temperature control module as the "1746-BTM module," the "BTM module," or as the "module"
- the programmable controller, as the "SLC processor" or the "processor"
- a thermocouple as a "TC"
- a time-proportioned output as "TPO"
- the tuning-assisted process as "TAP"
- proportional-integral-derivative as "PID"
- cold-junction compensation as "CJC"

Table of Contents

Getting Started

Chapter 1

Temperature Control Using a BTM Module in an SLC System 1–1
Features of the Temperature Control Module 1–2
Module Outputs 1–2
Module Addressing 1–3
Response to Slot Disabling 1–4

Installing and Wiring

Chapter 2

Avoiding Electrostatic Damage 2–1
Compliance with European Union Directive 2–1
EMC directive 2–1
Low voltage directive 2–2
Determining Power Requirements 2–2
Planning for Sufficient Enclosure Depth 2–2
Choosing a Module Slot in a Local I/O Chassis 2–2
Installation Considerations 2–2
Installing the Module 2–3
Removing the terminal block 2–4
Wiring the Module 2–4
Cold junction compensation (CJC) 2–5
Wiring considerations 2–5
Preparing and Wiring the Cables 2–6

Configuring the Module

Chapter 3

Loop Operation Mode 3–1
Type of Loop Input 3–1
Enable Loop Alarms 3–1
TC Break Response 3–1
Loop Autotune Gains Level 3–1
Barrel/Non-Barrel Control 3–2
Barrel control 3–2
Non-barrel control 3–2
Switching the barrel control 3–2
Inner/Outer Zone Selection 3–2
High/Low CV Limits 3–3
TC Break Control 3–3
Standby Setpoint 3–3
Heat/Cool Minimum On-times 3–3
Heat/Cool TPO Period 3–4
PV Rate and Associated Alarm 3–4
High/Low Temperature and Deviation Alarms 3–4

Configuring the Module

Chapter 3 (continued)

Alarm Dead Band 3–4
Thermal Integrity Loss Detection (thermal runaway) 3–5
Heat/Cool Ramp Rates 3–5
Non-barrel Autotune Disturbance Size 3–5
Implied Decimal Point 3–5
Configuration Block, M1 File 4–6

Setting Autotune and Gains Values

Chapter 4

Sequence of Setting PID Gains 4–1
Autotuning the Loops 4–2
Fine-Tuning the Loops 4–3
Using the PID Equation 4–3
Entering Autotune/Gains Values with Implied Decimal Point 4–4
Autotune/Gains Block, M0 File 4–4

Control and Autotune a Loop

Chapter 5

Controlling a Loop 5–1
Autotune a Loop 5–2
Requirements for autotune 5–2
Items to check before autotune 5–2
Autotune barrel control applications 5–3
Autotune non-barrel control applications 5–4
Items to check if autotune was not successful 5–5
Using the Output Image Table 5–5
Operating Commands to Loops 1–4 5–5
Global Commands to All Loops 5–6

Monitoring Status Data

Chapter 6

Input Image Table 6–1
Status Values from Each Loop 6–1
Global Status From All Loops 6–2

Calibrating the Module

Chapter 7

About the Procedure 7–1
Calibration Codes and Status 7–1
Calibration Procedure 7–1

Troubleshooting the Module

Chapter 8

Troubleshooting with the LED Indicators 8–1
Locating Error Code Information 8–1

Sample Program

Chapter 9

Obtaining the Sample Program from the Internet 9–1
Configuring Your SLC Processor, Off-line 9–1
Using the Sample Program 9–2
Data Table Layout for the Sample Program 9–2
The Sample Program 9–3

Module Specifications

Appendix A

Electrical Specifications A–1
Physical Specifications A–1
Environmental Specifications A–2
Input Specifications A–2
Overall accuracy A–3
Thermocouple Resolution A–3

Loop Data

Appendix B

Configuration Values, M1 File B–2
Autotune/Gains Values, M0 File B–3
Loop Operating Commands B–3
Loop Status Values B–3
Global Commands to All Four Loops B–4
Global Status from All Four Loops B–4

Notes:

Getting Started

This chapter gives you information on:

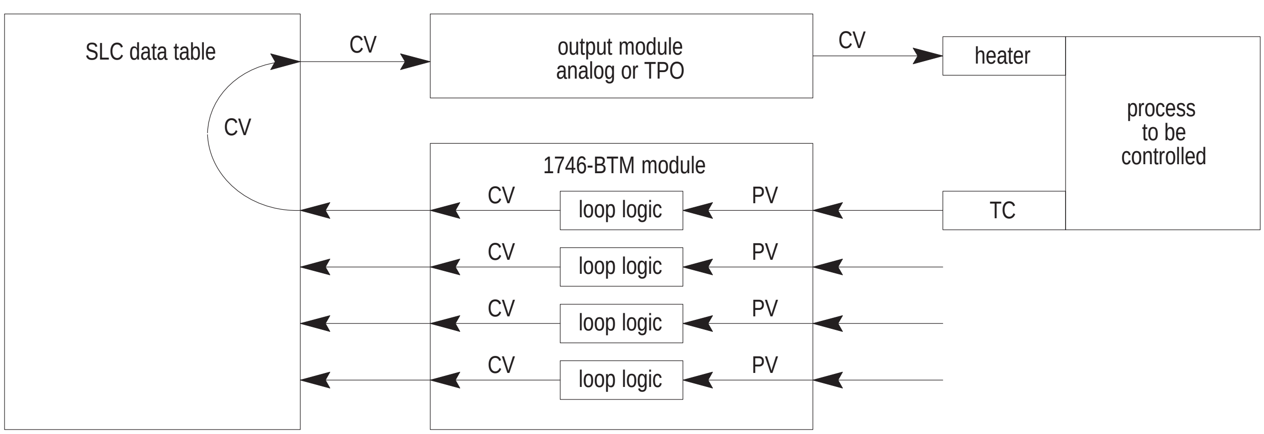
- the function of the temperature control module
- features of the temperature control module
- time-proportioned output (TPO)
- module addressing
- response to slot disabling

ATTENTION: Use the 1746-BTM module only for barrel temperature control for injection molding applications or extruders in a local I/O chassis. Any other applications are not supported.

Temperature Control Using a BTM Module in an SLC System

The temperature control module is an intelligent I/O module that can provide a maximum of 4 PID loops for temperature control. The module has 4 analog thermocouple (TC) inputs. Each analog input functions as the process variable (PV) for a PID loop. The PID algorithm and tuning-assisted-process (TAP) algorithm are performed on the module for each of the loops. The control-variable (CV) output of each loop, either analog output or time-proportioned output (TPO), is sent from the module to the SLC data table. Your application ladder logic must access the CV value in the data table and send the analog or TPO data to an output module to close the loop.

Figure 1.1
A 1746-BTM module with 4 PID logic channels, showing one complete PID loop



Features of the Temperature Control Module

The 1746-BTM module provides:

- 4 independent temperature control loops
- autotune PID loops (one loop or any combination of loops can be autotuned while other loops are running)
- a unique start-up algorithm to minimize overshoot
- an isolated thermocouple (J and K) input for each PID loop
- 16-bit analog-to-digital converter resolution (0.1° resolution)
- a heat CV signal (for each PID loop) as a numeric % value
- a cool CV signal (for each PID loop) as a numeric % value
- a heat CV signal (for each PID loop) as a TPO bit
- a cool CV signal (for each PID loop) as a TPO bit
- temperature values in °C or °F
- self-calibration (external reference required)
- user-selectable high and low alarms with dead band for hysteresis
- input open-circuit detection

Module Outputs

The BTM module sends the control variable (CV) for heating and/or cooling each loop to the SLC processor's input image table as both of:

- numeric value (current CV)
- time-proportioned output (TPO)

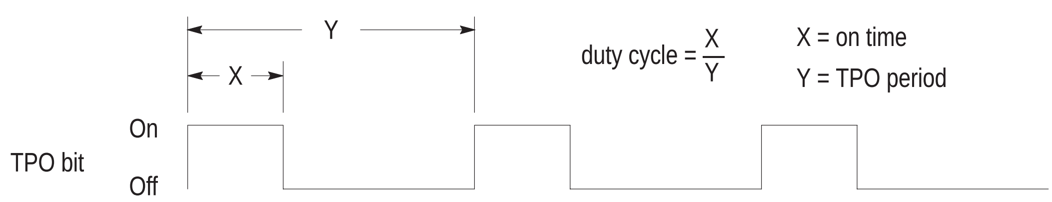
Current CV

Your ladder logic should read the numeric value (current CV), scale it, and send it to an analog output module to generate the control signal to an analog temperature control actuator. If using the sample program look for current CVs in N7:180-183 for loops 1-4 (more in chapter 9).

TPO

The module returns the heat TPO (bit 6) and cool TPO (bit 7) in input image table words 8-11 for loops 1-4. The sample program sends TPO signals to a digital output module to generate the control signal to a digital temperature control actuator. See sample program (chapter 9).

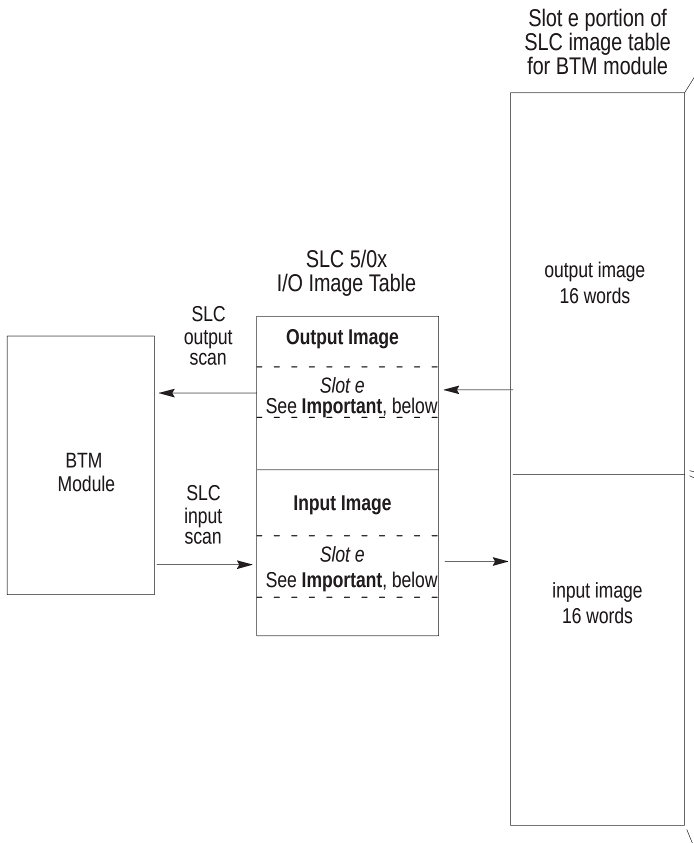
Figure 1.2
TPO timing diagram



Important: The TPO duty cycle (Y) must be considerable shorter in time than the system dead time. For additional information, refer to Autotune the Loop in chapter 5.

Module Addressing

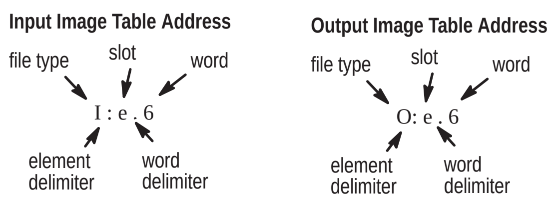
The following memory map shows you how the SLC processor's output and input image tables are defined for the module. For the sample program's Data Table Layout, see chapter 9.



Bit 15 Bit 0		Address
Loop 1 configuration data	word 0	O:e.0
Loop 2 configuration data	word 1	O:e.1
Loop 3 configuration data	word 2	O:e.2
Loop 4 configuration data	word 3	O:e.3
Loop 1 run setpoint value	word 4	O:e.4
Loop 2 run setpoint value	word 5	O:e.5
Loop 3 run setpoint value	word 6	O:e.6
Loop 4 run setpoint value	word 7	O:e.7
Loop 1 manual output value	word 8	O:e.8
Loop 2 manual output value	word 9	O:e.9
Loop 3 manual output value	word 10	O:e.10
Loop 4 manual output value	word 11	O:e.11
miscellaneous control bits	word 12	O:e.12
not used	word 13	O:e.13
not used	word 14	O:e.14
not used	word 15	O:e.15

Loop 1 temperature	word 0	I:e.0
Loop 2 temperature	word 1	I:e.1
Loop 3 temperature	word 2	I:e.2
Loop 4 temperature	word 3	I:e.3
Loop 1 configuration status	word 4	I:e.4
Loop 2 configuration status	word 5	I:e.5
Loop 3 configuration status	word 6	I:e.6
Loop 4 configuration status	word 7	I:e.7
Loop 1 control status and TPO	word 8	I:e.8
Loop 2 control status and TPO	word 9	I:e.9
Loop 3 control status and TPO	word 10	I:e.10
Loop 4 control status and TPO	word 11	I:e.11
If using the sample program, variables in words 12-15, including current CVs, are multiplexed and scanned into N7:170-188. See Data Table Layout in chapter 9.	word 12	I:e.12
	word 13	I:e.13
	word 14	I:e.14
	word 15	I:e.15

Important: When you enter the module ID in processor configuration (off-line), the processor automatically reserves the required number of I/O image table words. That section of the I/O image table (above) is designated by "slot e". Its location in the I/O image table is determined by the module's slot location "e" in the I/O chassis. Slot location "e" is a required addressing unit when referring to the module in ladder logic.



Response to Slot Disabling

By writing to the status file in your modular SLC processor you can disable any chassis slot. See your SLC programming manual for the slot disable/enable procedure.

ATTENTION: Always understand the implications of disabling the module before using the slot disable feature.

Input response

When the slot for this module is disabled, the module continues to update its inputs. However, the SLC processor does not read from a module whose slot is disabled. Therefore, inputs appearing in the processor image table remain in their last state, and the module's updated inputs are not read. When the processor re-enables the module slot, the current state of module inputs are read by the controller during the subsequent scan.

Output response

When the slot for this module is disabled, configuration words in the SLC processor's output image table are held in their last state and not transferred to the module. When the slot is re-enabled, output image table words are transferred to the module during the subsequent scan.

Installing and Wiring

This chapter gives you information about:

- avoiding electrostatic damage
- compliance with European Union directive
- determining the module's chassis power requirement
- planning for sufficient enclosure depth
- choosing a module slot in a local I/O chassis
- installing the module
- wiring the module

Avoiding Electrostatic Damage

Electrostatic discharge can damage semiconductor devices inside this module if you touch backplane connector pins. Guard against electrostatic damage by observing the following precautions:

ATTENTION: Electrostatic discharge can degrade performance or cause permanent damage. Handle the module as stated below.

- Touch a grounded object to rid yourself of charge before handling.
- Wear an approved wrist strap when handling the module.
- Handle the module from the front, away from the backplane connector. Do not touch backplane connector pins.

Compliance with European Union Directive

If this product is installed within the European Union or EEA regions and has the CE mark, the following regulations apply.

EMC directive

This apparatus is tested to meet Council Directive 89/336 Electromagnetic Compatibility (EMC) using a technical construction file and the following standards, in whole or in part:

- EN 50081-2 EMC – Generic Emission Standard, Part 2 – Industrial Environment
- EN 50082-2 EMC – Generic Immunity Standard, Part 2 – Industrial Environment

The product described in this manual is intended for use in an industrial environment.

Low voltage directive

This apparatus is also designed to meet Council Directive 73/23 Low Voltage, by applying the safety requirements of EN 61131–2 Programmable Controllers, Part 2 – Equipment Requirements and Tests.

For specific information that the above norm requires, see the appropriate sections in this manual, as well as the following Allen-Bradley publications:

- Industrial Automation Wiring and Grounding Guidelines, publication 1770-4.1
- Automation Systems Catalog

Determining Power Requirements

The module receives its power through the 1746 I/O chassis backplane from the modular +5V dc/ +24V dc chassis power supply. The maximum steady-state backplane current load of the module is:

5V dc amps	24V dc amps
0.110	0.085

When computing power supply requirements, add the values shown above to the requirements of all other modules in the SLC chassis to prevent overloading the chassis power supply.

Planning for Sufficient Enclosure Depth

The cable connector sticks out from the front of the module. The enclosure must provide room for a total of 8.2 inches (215 mm) from the back-panel to the connector.

Choosing a Module Slot in a Local I/O Chassis

Place your module in any slot of an SLC500 modular, or modular expansion chassis, except for the left-most slot (slot 0) reserved for the SLC processor or adapter modules.

Important: For proper operation, use this module with a local processor. The module is not designed to operate in a remote chassis.

Installation considerations

Most thermocouple-type applications require an industrial enclosure to reduce the effects of electrical interference. Thermocouple inputs are highly susceptible to electrical noises due to the small signal amplitudes (microvolt/°C). Isolate them from other input wiring and modules that radiate electrical interference.

Group your modules within the I/O chassis to minimize adverse effects from radiated electrical noise and heat. Consider the following conditions when selecting a slot location. Position the module *away from* modules that:

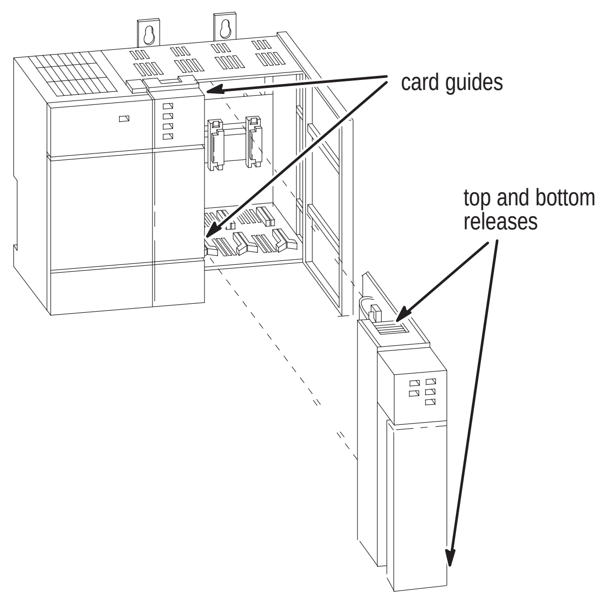
- connect to sources of electrical noise such as relays and ac motor drives
- generate significant heat, such as 32-point I/O modules

Installing the Module

Follow this procedure:

ATTENTION: Never install, remove, or wire modules with power applied to the chassis or devices wired to the module.

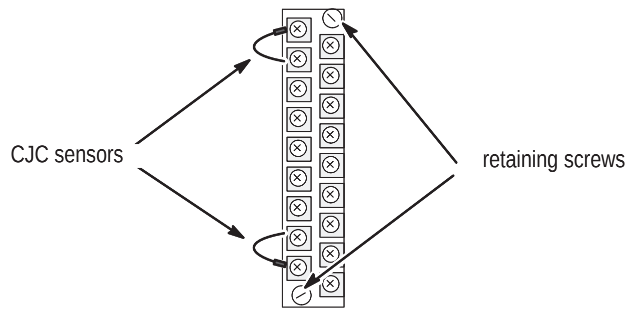
1. Align the circuit board of the thermocouple module with the card guides located at the top and bottom of the chassis.
2. Slide the module into the chassis until both top and bottom retaining clips are secured. Apply firm even pressure on the module to attach it to its backplane connector. Never force the module into the slot.
3. Cover unused slots with the card slot filler, catalog number 1746-N2.
4. To remove, press the releases at the top and bottom of the module, and slide the module out of the chassis slot.



Removing the terminal block

When installing the module, it is not necessary to remove the terminal block. But if you need to remove it, follow this procedure:

1. Alternately loosen the two retaining screws to avoid cracking the terminal block.
2. Grasp the terminal block at the top and bottom and pull outward and down. When removing or installing the terminal block be careful not to damage the CJC sensors.



3. Use the write-on label to identify the module and its location.

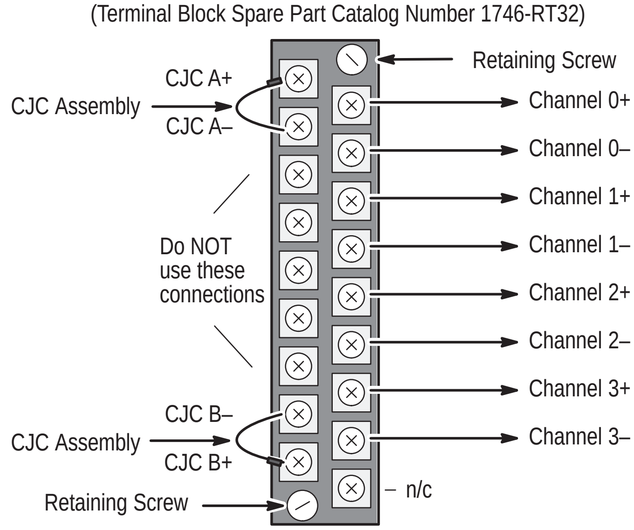
SLOT ____ RACK ____
***** MODULE _______________

Wiring the Module

The module's 18-position, removable terminal block has a terminal pin-out is shown below.

ATTENTION: Disconnect power to the SLC before attempting to install, remove, or wire the removable terminal wiring block.

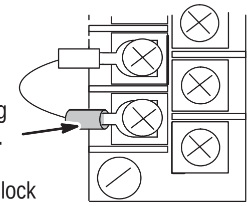
(Terminal Block Spare Part Catalog Number 1746-RT32)



Cold Junction Compensation (CJC)

ATTENTION: Do not remove or loosen the cold junction compensating thermistors located on the terminal block. *Both thermistors are critical to ensure accurate thermocouple input readings at each channel.* The module will not operate in the thermocouple mode if a thermistor is removed.

In case of accidental removal of one or both thermistors, replace them by connecting them across the CJC terminals located at the top and/or bottom left side of the terminal block. Always connect the red lug to the (+) terminal (to CJC A+ or CJC B+).



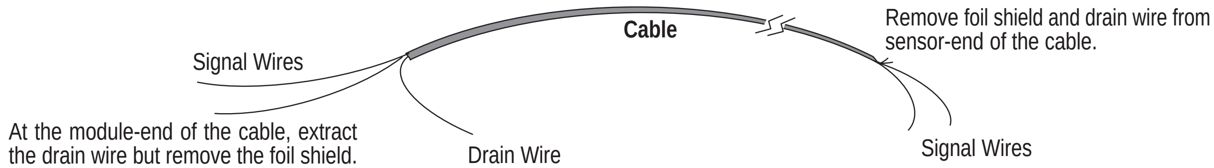
Wiring considerations

Follow the guidelines below when planning your system wiring.

- To limit the pickup of electrical noise, keep thermocouple and millivolt signal wires away from power and load lines.
- For high immunity to electrical noise, use Alpha 5121 (shielded, twisted pair) or equivalent wire for millivolt sensors; or use shielded, twisted pair thermocouple extension lead wire specified by the thermocouple manufacturer. Using the incorrect type of thermocouple extension wire or not following the correct polarity may cause invalid readings. See IEEE Std. 518, Section 6.4.2.7 or contact your sensor manufacturer for additional details.
- When trimming cable leads, minimize the length of unshielded wires.
- Ground the shield drain wire at only one end of the cable. The preferred location is at the I/O chassis ground (Figure 3.1).
- For maximum noise reduction, use 3/8 inch braid wire to connect cable shields to the nearest I/O chassis mounting bolt. Then connect the I/O chassis to earth ground (Figure 3.1). These connections are a requirement regardless of cable type.
- Tighten terminal screws. Excessive tightening can strip the screw.
- The open-circuit detector generates approximately 20 nano-amperes into the thermocouple cable. A total lead resistance of 25 ohms (12.5 one-way) will produce 0.5 μV of error.
- Follow system grounding and wiring guidelines found in your SLC 500 Modular Hardware Installation and Operation Manual, publication 1747-6.2.

Preparing and Wiring the Cables

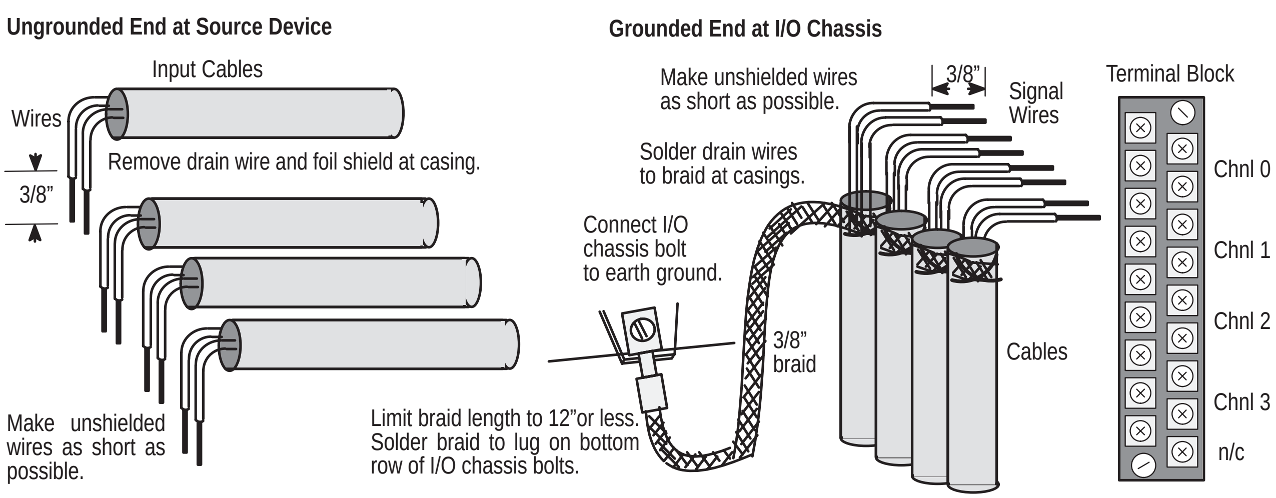
To prepare and connect cable leads and drain wires, follow these steps.



1. At each end of the cable, strip some casing to expose individual wires.
2. Trim signal wires to 5-inch lengths beyond the cable casing. Strip about 3/16 inch (4.76 mm) of insulation to expose the ends of the wires.
3. At the module-end of the cables:
   – extract the drain wire and signal wires
   – remove the foil shield
   – bundle the input cables with a cable strap
4. Connect drain wires together and solder them to a 3/8" wire braid, 12" long. Keep drain wires as short as possible.
5. Connect the 3/8" wire braid to the nearest chassis mounting bolt.
6. Connect the signal wires of each channel to the terminal block .
   Important: Only after verifying that your connections are correct for each channel, trim the lengths to keep them short. Avoid cutting leads *too* short.
7. At the source–end of cables from mV devices (Figure 3.1):
   – remove the drain wire and foil shield
   – apply shrink wrap as an option
   – connect to mV devices keeping the leads short

Important: If noise persists, try grounding the opposite end of the cable, instead. (Ground one end only.)

Figure 3.1
Cable Preparation and Connections to Minimize Electrical Noise Interference



Configuring the Module

You configure the module by setting words and bits for each loop in Configuration Block, N7:10-110, which your ladder logic uses to load the module's M1 file. We cover bit selections and word descriptions. Refer to the table at the end of this chapter for selections, units, and defaults.

Loop Operation Mode
(word 1, bits 0 & 1 for channel 1)

Use these bits to select how you want the loop to perform:

Mode of Loop Operation	01	00
monitor the loop to indicate temperature and alarms	0	0
perform PID loop control with temperature indication and alarms	0	1
disable the loop	1	0

Type of Loop Input
(word 1, bits 2-5, for channel 1)

Use these bits to select type J or K thermocouple:

TC	05	04	03	02
type J	0	0	0	0
type K	0	0	0	1

Enable Loop Alarms
(word 1, bit 6, for channel 1)

Set this bit to enable alarms for the designated loop.

TC Break Response
(word 1, bits 7 & 8, for channel 1)
(also see word 4, or O:e.8)

If the module detects a TC open wire for a loop in automatic mode, the module responds in one of these ways that you select:

TC Break Response	08	07
disables the loop	0	0
forces CV to TC Break Control value (word 4, below)	0	1
forces CV to manual % output (O:e.8 for loop 1)	1	0

Loop Autotune Gains Level
(word 1, bits 10,11 for channel 1)

You can change and download autotune gains level selection for any or all zones at any time. When changed, you must redownload the M1 file (configuration) followed by the M0 file (autotune/gains) so the module can recalculate PID values based on new loop autotune gains. You do not need to re-autotune.

Autotune Gain Level	11	10
low	0	0
medium	0	1
high	1	0
very high	1	1

Barrel/Non-Barrel Control
(word 1, bit12, for channel 1)

You select between barrel and non-barrel control.

Select:	for these applications:	12
barrel control	heat-only or heat/cool	0
non-barrel control	heat-only, cool-only, or heat/cool	1

Barrel control

Select barrel control for multiple-zone applications in which there is thermal conduction between the zones. Injection molding and extrusion are good example applications because they use multiple heater bands (zones) mounted on one thermal conductor (the metal barrel). The barrel conducts heat between different zones. If you select barrel control, also select between inner and outer zones (word 1, bit 13 for channel 1). A barrel loop is autotuned as the temperature rises from a cold start to a temperature setpoint during startup.

Non-barrel control

Select non-barrel control for applications with independent loops and no thermal conduction between zones. If you select non-barrel control, the inner/outer zone selection doesn't apply.

Switching the barrel control

For some applications, even though the loops are independent with no thermal conduction between zones, barrel control might provide better performance than non-barrel control. If a loop has any of these characteristics, you might want to use barrel control if the:

- time constant is greater than 100.0 secs
- loop has a problem of overshooting the setpoint
- loop output is saturating (CV is at 100%) for a significant duration

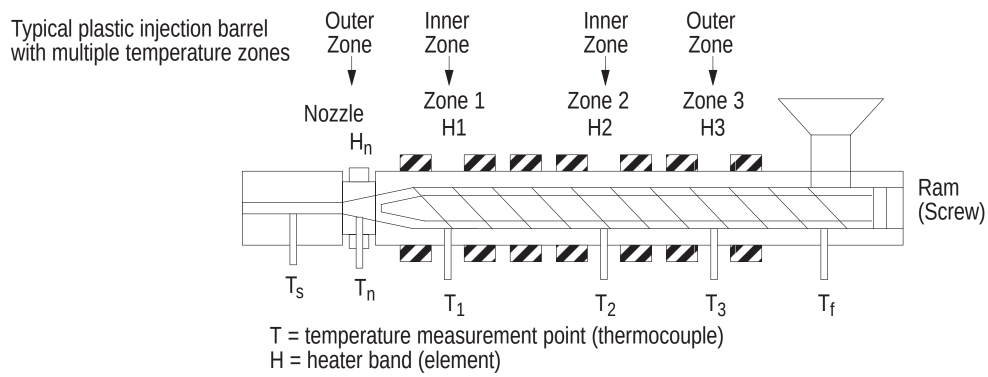
ATTENTION: If you switch a loop between non-barrel and barrel control, you must re-autotune the loop before operating it. If you don't re-autotune, the autotune values will be wrong for the application and the gains will be greatly distorted.

Inner/Outer Zone Selection
(word 1, bit 13, for channel 1)

If you make a selection for barrel control, you also must select whether the loop is an inner zone or outer zone.

Select:	for a zone:	13
inner	not at either end of the barrel	0
outer	at either end of the barrel	1

The PID gain calculation algorithm for an inner zone is slightly different than that for an outer zone to account for an inner zone being more affected by adjacent zones. The inner zones are treated as more of an integrating process than the outer zones.



If you change zone selection after autotune, you must re-autotune.

High/Low CV Limits
(words 2 & 3 for channel 1)

Use CV High and Low Limits to set up the loop mode:

For this loop mode:	CV Low:	CV High:
heat, only	0 %	100%
cool, only	–100%	0
heat/cool	–100%	+100%

TC Break Control
(word 4, or O:e.8 for channel 1), (also see word 1, bits 7 & 8,)

If a loop input circuit becomes open (open wire) the loop can not measure temperature. In automatic mode, the lack of temperature feedback makes it impossible to control the temperature. To guard against this condition, the BTM module provides TC break detection. When a break is detected, the module responds in one of these ways:

- disables the loop
- forces CV to this (TC Break Control) value (word 4 for loop 1)
- forces the CV to the manual %-output value (O:e.8 for loop 1)

Standby Setpoint
(word 5 for channel 1)

When not using the runtime setpoint (O:e.4 for loop 1), use this value to hold a lower temperature for faster warm up and/or optimum standby conditions.

Heat/Cool Minimum On-times
(words 6 & 8 for channel 1)

These values determine the minimum cycle time that loop TPO bits will turn ON. They are used to allow contactors time to close or pull in. If the contactor is energized for less than this value, the contactor will not close, but the attempt will count as a cycle.

For example, suppose you set the TPO period for 10 seconds and the minimum ON time to 1 second. Then if the module calculates a CV% of 10% or less, the TPO bit for that zone will not turn ON.

Heat/Cool TPO Period
(words 7 & 9 for channel 1)

When CV loop output is time-proportioned (TPO), use this value to set the interval between successive turn-ONs. For less than a 100% output level, the output goes OFF for the balance of the interval.

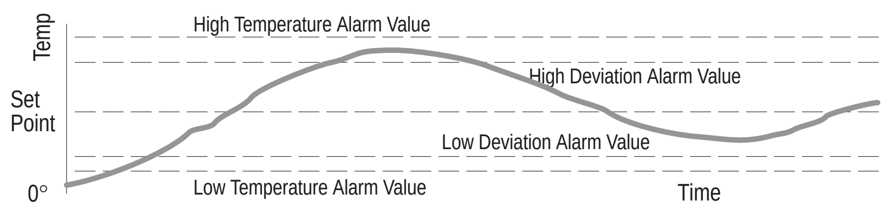
PV Rate and Associated Alarm
(word 10 and alarm bit I:e.4/05 for channel 1)

The PV Rate is a setpoint with an associated alarm that indicates when the PV is rising too rapidly. If the zone PV has risen more than this setpoint in one second, the module sets the PV rate alarm bit (I:e.4/05, loop 1).

High/Low Temperature and Deviation Alarms
(words 11-14 for channel 1)

In the configuration block (M1 file) you select values for the following temperature-level alarms:

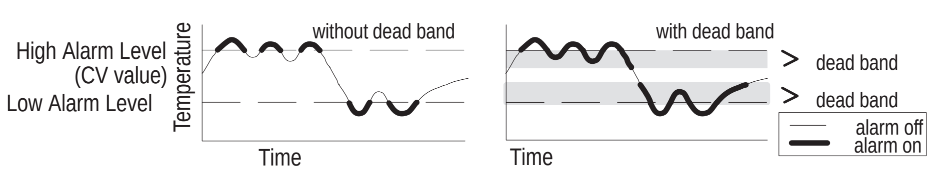
- low temperature alarm (word 11 for channel 1)
- high temperature alarm (word 12 for channel 1)
- low deviation alarm from the set point(word 13 for channel 1)
- high deviation alarm from the set point(word 14 for channel 1)



Alarm Dead Band
(word 15 for channel 1)

Once the temperature alarm bits are on, they are kept on until the temperature drops below the high alarm by the alarm dead-band value or rise above the low alarm by this value. The alarm dead band *applies to the CV value at the high and low temperature alarms and deviation alarm values.* The dead band provides a hysteresis effect.

- **Low Alarm With Dead Band** — When the temperature falls below the user-defined low alarm value, the low alarm bit is turned on. When the temperature rises above the level of the low alarm value but still below the level of the dead-band value, the low alarm bit remains on. Only when the temperature rises above the dead-band level will the alarm bit be turned off.
- **High Alarm With Dead Band** — When the temperature rises above the user-defined high alarm value, the high alarm bit is turned on. When the temperature falls below the level of the high alarm value but still above the level of the dead-band value, the high alarm bit remains on. Only when the temperature falls below the dead-band level will the alarm bit be turned off.



Important: The temperature passes thru the dead band before the alarm is turned on or off to provide stability to alarm indicators. Dead bands apply to CV and deviation alarms.

Thermal Integrity Loss Detection (thermal runaway) and Runaway Period

(words 16 & 17 for channel 1)

The loss of thermal integrity is detected when the loop, in automatic mode, is not responding to a CV at 100% Detecting the loss of thermal integrity requires an assumption of a minimum rate of change in the temperature input value (PV) when the output (CV) is at 100%. Examples of a loss of thermal integrity could be the failure of a heating-band contactor to close, or a sensor not in proper position to measure true temperature.

The values you enter in words 16 and 17 for loop 1 establish a minimum rate of change (°/min) in the temperature input (PV) that you allow when the the output (CV) is at 100% in automatic mode. The temperature change value you enter in word 16 divided by the period value you enter in word 17 is the thermal integrity rate.

Important: Once loss of thermal integrity is detected, you must clear this condition by disabling the affected loop and then re-enabling it.

Heat/Cool Ramp Rates

(words 18 & 19 for channel 1)

This value ramps the setpoint in steps to the new setpoint.

Non-barrel Autotune Disturbance Size

(word 20 for channel 1)

This is a pure %-output step function for performing autotune. It is added to the current output (%). It should be applied under steady-state conditions. The loop operating mode must be non-barrel.

Implied Decimal Point

Important: Because loop values are stored and reported in integer files, you must understand the meaning of implied decimal point (IDP). Otherwise, the magnitude of your intended value may be in error by as much as 1000, depending on the position of the IDP.

When entering or reading integer (counting number) values, the *range*, given in the associated table, tells you the implied decimal point. It is the number of digits *to the right* of the decimal point (for an example range of 0.0 thru 3276.7, the implied decimal point is 1). Also, you will probably need to use leading zeros when entering a value.

You read status values similarly. You must know the *range* of the value to read it correctly. For example, if reading a heat integral (0.0000 thru 3.2767), a display of 05000 would have a value of 0.5.

Parameter	Given Range	IDP*	Example
Thermal Runaway	0 thru 100°	0	If you want to store a value of 66°, enter 00066.
Standby Setpoint	0.0 thru 32767.7°	1	If you want to store a value of 660.0°, enter 06600.
TPO Period	0.00 thru 100.00 sec	2	If you want to store a value of 6%, enter 00600.
Cool Proportional	0.000 thru 32.767	3	If you want to store a value of 18, enter 18000.
Heat Integral	0.0000 thru 3.2767	4	If you want to store a value of 0.5, enter 05000.

*IDP indicates the number of digits *from* the *right* that locates the implied decimal point.

Configuration Block, M1 File, Loops 1–4 N7:10-110

Configuration block (M1 file) contains 101 words as listed below.
Important: Word numbers for loops 1-4 are in left-most columns.
For corresponding N7:xx address, add 10 to word the number.

Block Header (word 0 / N7:10) = 8801 (–30719 decimal)

Loops 1-4 / Word #							Set a Bit or Enter a Value															
1	2	3	4	Bit	To Configure	Bit Select or Range	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	26	51	76	0–1	Operation mode	Monitor; no PID control															0	0
						Control loop with PID															0	1
						Disable loop															1	0
				2–5	Input type	Type J											0	0	0	0		
						Type K											0	0	0	1		
				6	Alarm enable	Disable = 0; Enable = 1										x						
				7–8	TC break configuration	Disable PID loop (CV=0)								0	0							
						Use thermal runaway CV								0	1							
						Use manual mode CV								1	0							
				9	reserved																	
				10-11	Autotune gains	Low gains					0	0										
						Medium gains					0	1										
						High gains					1	0										
						Very high gains					1	1										
				12	Barrel control	Barrel = 0; Non-barrel = 1				x												
				13	Zone	Inner = 0; Outer = 1			x													
				14-15	reserved																	
2	27	52	77	0–15	High CV limit %	–100.00 thru +100.00%	default = +100.00%															
3	28	53	78	0–15	Low CV limit %	–100.00 thru +100.00%	default = 0.00%															
4	29	54	79	0–15	CV for TC break	–100.00 thru +100.00%	default =															
5	30	55	80	0–15	Standby setpoint	0.0 thru 3276.7°	default = 0.0															
6	31	56	81	0–15	Heat on time (min)	0.00 thru 100.00 sec	default = 0.00															
7	32	57	82	0–15	Heat TPO period	0.00 thru 100.00 sec	default = 5.00															
8	33	58	83	0–15	Cool on time (min)	0.00 thru 100.00 sec	default = 0.00															
9	34	59	84	0–15	Cool TPO period	0.00 thru 100.00 sec	default = 5.00															
10	35	60	85	0–15	PV alarm rate	–3276.8 thru +3276.7°/s	default = 0.0															
11	36	61	86	0–15	Low temp alarm	–3276.8– thru +3276.7°	default = +999.9															
12	37	62	87	0–15	High temp alarm	–3276.8 thru +3276.7°	default = +999.9															
13	38	63	88	0–15	Low deviation	–3276.8 thru +3276.7°	default = +999.9															
14	39	64	89	0–15	High deviation	–3276.8 thru +3276.7°	default = +999.9															
15	40	65	90	0–15	Alarm dead band	0.0 thru 10.0°	default = 0.0															
16	41	66	91	0–15	Thermal runaway	0 thru 100°	default = 5															
17	42	67	92	0–15	Runaway period	0 thru 100 minutes	default = 20															
18	43	68	93	0–15	Heat ramping	0 thru 100°/min	default = 0															
19	44	69	94	0–15	Cool ramping	0 thru 100°/min	default = 0															
20	45	70	95	0–15	Nonbarrel autotune disturb size (0.00–100.00%)		default = 10.00															
21	46	71	96	0–15	Startup aggressiveness factor (0 thru 100)		default = 0 for heat or cool, only; 25 for heat/cool															
>25	>50	>75	>99	reserved																		

Setting Autotune and Gains Values

This chapter shows you how to independently set the gains for each PID loop of the BTM module. This includes:

- setting PID gains
- autotuning the loops
- fine tuning the loops
- using the PID equation
- configuring the autotuning and gains block

Sequence of Setting PID Gains

Autotuning causes the module to measure the process dynamics and calculate PID gains.

↓

Reading the gains block from the module copies the gains generated by autotuning into the SLC files.

↓

Writing the gains block to the module overwrites any PID gains that had been in the module.

↓

Autotuning or writing the autotune block to the module causes the module to calculate PID gains and overwrite any PID gains that had been in the module.

Any time you cause successful autotuning of a loop, write an autotune block to the module, or write a gains block to the module, a new set of PID gains is established on the module.

At initial start-up, you must write the autotune block to the BTM module or perform autotuning. If you select autotuning, for any loop that is successfully tuned, the gains are calculated by the module. Gains you had sent to the module for a loop in any gains block previous to successful autotuning of the loop are superseded by the gains derived from autotuning. If you then read the gains block, it contains the gains derived from autotuning.

If autotuning is not successful for any loops (as indicated in the status block) the gains you had sent for those loops before autotuning will be used by the module.

Once autotuning is complete, you must read the gains block from the module to store it in SLC processor memory.

You can write the autotune and gains block either of these ways:

- Send autotune block to the module in words 1–24 (N7:121–144). This causes the module to calculate the PID gains. In this case, set the block header in word 0 (N7:120) to 880A hexadecimal.

  or

- Send PID gains only in words 25–48 (N7:145–168). This overwrites the current PID values in the module. In this case, set the block header in word 0 (N7:120) to 880B hexadecimal.

Important: When you download either an autotune or gains block, the BTM module’s PID algorithm requires time to adjust, proportional to the thermal mass of the system. This could cause a slow or unexpected system response.

The module's memory is volatile. Whenever power to the module is interrupted, you must again establish the gains. If you don't send an autotune block to the module, the module startup runs and the module operates in standard PID mode. Sending the autotune block establishes the start-up algorithm and the values the module uses to calculate the PID gains, causing the module to recalculate the PID gains. However, you can override the autotune gains by sending the gains block *after* the autotune block.

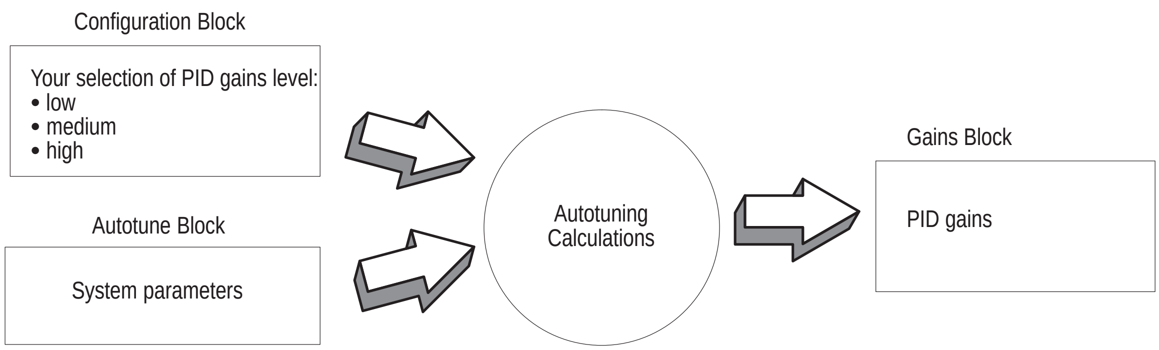
Important: You must initially download M0 and M1 files for the module to operate.

Autotuning the Loops

You select autotuning from the output image table block (chapter 5). For each loop, you must turn on the specific bit to enable autotuning for the corresponding loop. To trigger the start of autotuning, you must also cause a 0-to-1 transition of word 12, bit 1 of the output image table (see chapter 5).

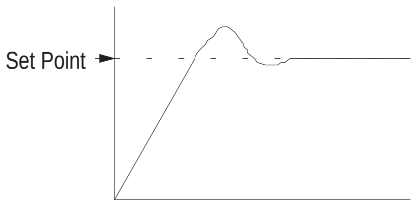
During autotuning, the module measures system parameters. At the end of autotuning, the module calculates PID gains based on these parameters and your selection of low, medium., or high PID gain level in the configuration block. When autotuning is complete, the PID gains calculated from autotuning are available in the gains block that you can read from the module.

Whenever you write autotune values to the module, it re-calculates PID gains based on measured system parameters stored in the autotune block and your selection of low, medium, or high PID gain level stored in the latest configuration block. If you had changed the level of PID gains selection in the configuration block in the mean time, the PID gains calculated would be different from those calculated originally.



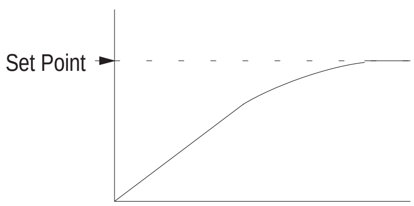
Fine-Tuning the Loops

After autotuning, you may want to fine-tune the loops by manually setting the gains. As you fine-tune a loop, first try adjusting the proportional gain; this will have the greatest impact. Your second choice for adjustment should be the integral gain. The derivative gain should be the last choice for fine-tuning a loop.



If the loop **over-shoots** the set point either at start-up or at a change of set point, you may be able to dampen the loop response by doing one or more of the following (in order of effectiveness):

1. decrease the proportional gain
2. decrease the integral gain
3. increase the derivative gain



If the loop is **slow** in reaching the set point either at start-up or at a change of set point, you may be able to improve the loop response by doing one or more of the following (in order of effectiveness):

1. increase the proportional gain
2. increase the integral gain
3. decrease the derivative gain

Using the PID Equation

The module provides dependant PID control action. Dependent control action can be represented by the equation:

$$CV = K_p\left[E + K_i\int_0^t E \cdot dt + K_d\frac{dE}{dt}\right]$$

Where:
CV = Control variable
K_p = Proportional gain (no units)
E = Error (SP–PV or PV–SP)
K_i = Integral gain (repeats/second)
K_d = Derivative gain (seconds)
t = Time

The module is capable of performing PID control by calculating the solution to an approximation of the PID equation. The approximation is represented by the equation:

$$CV = K_p\left(E + K_i\sum_0^t E \cdot \Delta t + K_d\frac{\Delta E}{\Delta t}\right)$$

Where:

$$\sum_0^t E \cdot \Delta t = E_1 \cdot \Delta t + E_2 \cdot \Delta t + E_3 \cdot \Delta t + \ldots$$

Entering Autotune/Gains Values with Implied Decimal Point

The autotune/gains block (M0 file) contains 49 words as listed below. For each gain value, you enter a 16-bit integer value.

Important: Because loop values are stored and reported in integer files, you must understand the meaning of implied decimal point (IDP). Otherwise, the magnitude of your intended value may be in error by as much as 1000, depending on the position of the IDP.

When entering or reading integer (counting number) values, the *range*, given in the associated table, tells you the implied decimal point. It is the number of digits *to the right* of the decimal point (for an example range of 0.0 thru 3276.7, the implied decimal point is 1). Also, you will probably need to use leading zeros when entering a value.

Parameter	Given Range	IDP*	Example
Cool Time Constant	0.0 thru 32767.7 sce	1	If you want to store a value of 660.0, enter 06600.
Heat Gain	0.00 thru 327.67°/sec	2	If you want to store a value of 100.00, enter 10000.
Cool Proportional	0.000 thru 32.767	3	If you want to store a value of 18, enter 18000.
Heat Integral	0.0000 thru 3.2767	4	If you want to store a value of 0.5, enter 05000.

*IDP indicates the number of digits *from* the *right* that locates the implied decimal point.

Gains/Autotune Block, M0 File, for Loops 1-4

Important: Word numbers for loops 1-4 are in left-most columns. For corresponding N7:xx address, add 120 to word the number.

Gains/Autotune (N7:120–168): Block Header (word 0 / N7:120) = 880B (–30709 decimal)

Loops 1–4				Autotune Values (N7:120–144)		
1	**2**	**3**	**4**	**To Configure**	**Range**	**Enter a Value**
1	7	13	19	Heat gain	0.00 thru 327.67°/sec	
2	8	14	20	Heat time constant	0.0 thru 3276.7 sec	
3	9	15	21	Heat dead time	0.0 thru 3276.7 sec	
4	10	16	22	Cool gain	0.00 thru 327.67°/sec	
5	11	17	23	Cool time constant	0.0 thru 3276.7 sec	
6	12	18	24	Cool dead time	0.0 thru 3276.7 sec	
Loops 1–4				**Gains Values (N7:145–168)**		
25	31	37	43	Heat proportional	0.000 thru 32.767 sec	
26	32	38	44	Heat integral	0.0000 thru 3.2767 rpts/sec	
27	33	39	45	Heat derivative	0.0 thru 3276.7 sec	
28	34	40	46	Cool proportional	0.000 thru 32.767 sec	
29	35	41	47	Cool integral	0.0000 thru 3.2767 rpts/sec	
30	36	42	48	Cool derivative	0.0 – thru 3276.7 sec	

Control and Autotune a Loop

This chapter explains how to:

- control loop operation
- autotune a loop

Because of the extensive re-write, change bars have been omitted.

Controlling a Loop

At initial start-up, you must write the M1 configuration block to establish the module's mode of control. Then, you must update the output image table any time you want to change the operating mode.

M1 Configuration File

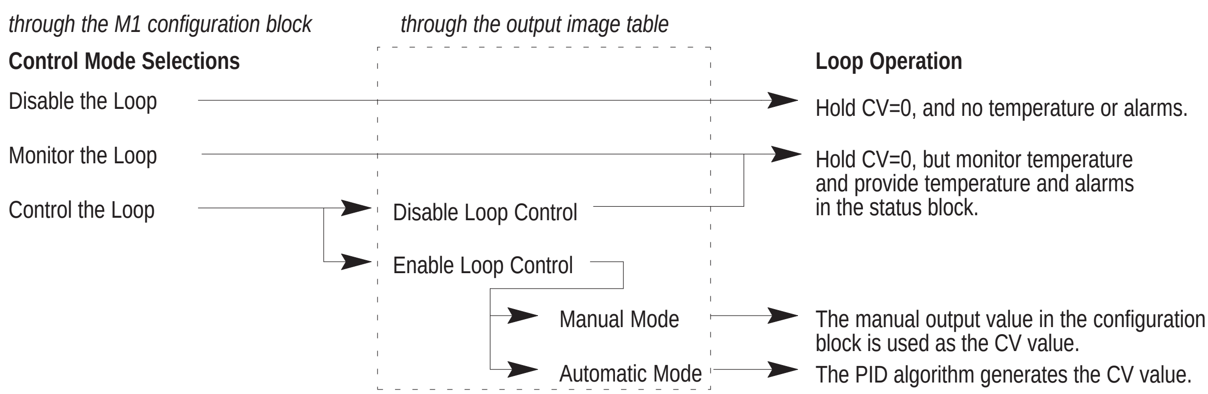
You select the loop control mode in the configuration file:

Words:	Bit 01	Bit 00	Lets you select:
1, 26, 51, 76 for loops 1–4	0	0	monitor the loop
	0	1	control the loop with PID
	1	0	disable the loop

Output Image Table

If you select control-the-loop mode, you control loop operation with these words and bits in the output image table (abbreviated list):

Words:	Bit:	Lets you:
0–3 loops 1–4	00	enable or disable the loop
	03	enable or disable autotune
4–7	n/a	enter run temperature setpoints
8–11	n/a	enter manual CV % output values
12 global for all loops	01	invoke autotune
	02	abort autotune
	03	reset error codes



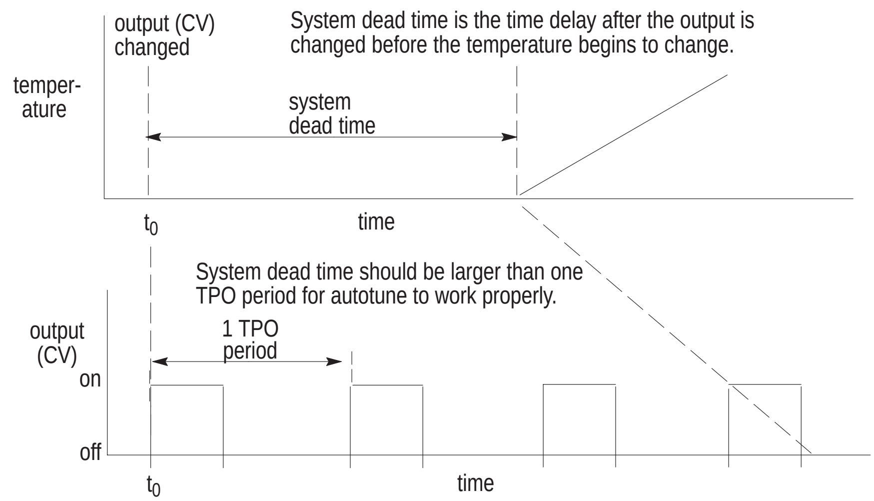
The BTM module uses the output image table to control loop operation. We list the words and bits at the end of this chapter.

Autotune a Loop

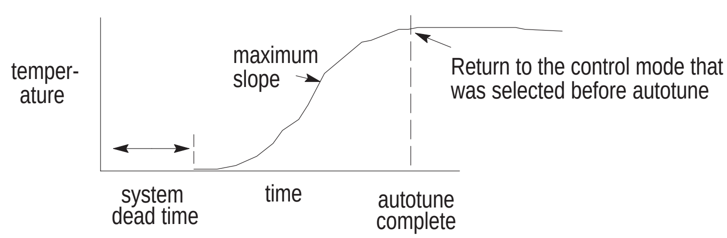
Use the following as a guide:

Requirements for Autotune

- Start autotune from a steady-state temperature. For best results, do a cold start. If the temperature fluctuates, autotune may not provide accurate results.
- The runtime setpoint for autotune must be at least 50º above current temperature or autotune will not start.
- Set the TPO period smaller than the system dead time. Autotune algorithm may calculate excessive gains if system dead time is less than the TPO period. This may cause the PV to overshoot.



- The autotune algorithm does not take the temperature to setpoint. When autotune is complete, the zones will return to the mode (auto or manual) that was selected before autotune.



Items to check before autotune

Make sure that each loop:

1. is properly configured with a valid M1 file and no errors
2. is set for barrel mode
3. is in manual mode and that run setpoints are selected
4. TPO period is set considerably smaller that system dead time
5. no alarm conditions could cause problems (such as a TC break)

Autotune barrel control applications

Autotune enables the module to compute PID values for optimum temperature control. The following autotune procedure is based on the ladder program presented in chapter 11. You must load the program and use this procedure to autotune the module.

For **barrel control**, better results are achieved when you autotune all loops associated with the barrel. After autotune, each zone will return to the mode (auto or manual) that was selected beforehand.

Important: For best results, start from room temperature (cold start).

1. Set initial conditions:
Configuration File N7:
N7:11 bits 00, 01 set for PID control
N7:36 bits 00, 01 set for PID control
N7:61 bits 00, 01 set for PID control
N7:86 bits 00, 01 set for PID control
remaining bits/words set for your application

Output image table words 0-3 bits 00-03 (O:e.x/xx) for loops 1-4
bit 00 = 1 enables PID control
bit 01 = 0 puts loop into manual mode
bit 02 = 1 uses runtime setpoint
bit 03 = 1 enables autotune

Set to zero Output image table words 8-11 (O:e.x) for loops 1-4
This zeros manual outputs to remove control signals from loops.

2. Download the M1 Configuration File by setting N7:0/00 = 1.

3. Download the M0 Autotune File by setting N7:0/01 = 1.

4. Verify that the M1 Configuration File downloaded:

Check input image words 4-7 bits 03, 04 for loops 1-4 to verify:
bit 03 = 1 module received a valid M1 file for the loop
bit 04 = 0 no parameter errors for the loop
If bit 04 (parameter error) is set for any loop, look for the error code in N7:185-188. Look up the error code in chapter 8 listing.

Check input image words 8-11 bits 00-02 for loops 1-4 to verify that the module:
bit 00 = 1 enabled PID control
bit 01 = 0 put loop into manual mode
bit 02 = 1 used runtime setpoint

5. Enter runtime temperature setpoints (at least 50°F above current) into output image words 4-7 **(O:e.x) for loops 1-4**.
Important: For implied decimal point, enter 2000 for 200°.

6. Invoke autotune. (Starts autotune for loops enabled in step 1.)
Set output image table word 12, bit 1 = 1 **(O:e.12/01**).
The module needs a 0-1 transition of this bit.

7. Verify autotune is in progress.
   Monitor input image word 8, bit 11 **(I:e.8/11**) for a 0-1 transition.

8. Reset the autotune enable bit.
   Reset output image table word 12, bit 1 = 0 **(O:e.12/01)**.

9. Enable each loop for automatic mode.
   This lets each loop control to run setpoint at autotune completion.

   Output image words 0-3 bit 01 (O:e.x/01) for loops 1-4
   bit 01 = 1 puts loop into automatic mode

10. Verify that autotune is complete and successful.
    Input image words 8-11 bits 03 and 04 (I:e.x/xx) for loops 1-4
    bit 03 = 1 autotune complete
    bit 04 = 1 autotune successful
    If bit 04 = 0 (not successful) for any loop, look for the error code in N7:185-188. Look up the error code in the chapter 10 listing.

11. Upload the autotune/PID gains block to the processor for storage.
    Set word N7:0 = 24.
    Following a power loss or module replacement, you can download the autotune/PID gains block to avoid repeating this procedure.

12. We suggest that you modify our ladder code (chapter 11) to set N7:0 = 3 at power up. This will automatically download M0 and M1 files to the module to start module operation.

Autotune non-barrel control applications

1. Enter a safe non-barrel autotune disturbance size in the M1 file.

   Disturbance size is the step output that the module uses to autotune. For example, if disturbance size is 15% and current CV is:
   - 0% when autotune is invoked, the CV changes to 15%
   - 10% when autotune is invoked, the CV changes to 25%

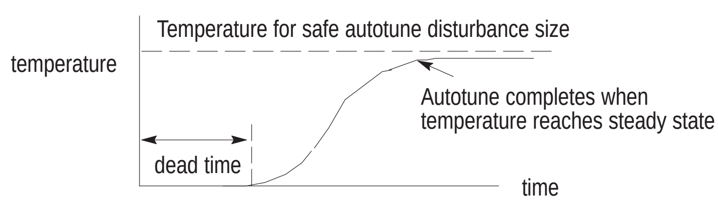
   Optimum disturbance lets temperature rise, then level off. If too large, temperature will not level off and autotune will be unsuccessful.

2. Make sure all zones have valid M1 files and no parameter errors.

3. Do a cold start or start from a *steady-state* temperature.
   a. If doing a cold start, invoke autotune *after* putting loop into manual mode and setting manual CV output to zero.
   b. If starting from a steady-state temperature, invoke autotune.

4. When autotune completes, upload the autotune and gains block.

5. Return the zone to auto mode.



Items to check if autotune was not successful

The module reports successful completion of autotune in status word 8, bits 03, and 04 in the input image table (I:e.8/03, 04). If not successful, look for autotune error codes in N7:185–188. These are described in chapter 8. Also check that:

1. You uploaded autotune/gains values after autotune completion.

2. You selected an optimum PID gains level (configuration block). If the loop responded too rapidly or slowly, change this selection.

Using the Output Image Table

The output image table contains 16 words as shown below. You must enter a 16-bit signed integer value for the run temperature setpoint and manual output.

- For a run temperature setpoint, the implied decimal point is **1** place from the right (causing the resolution to be **0.1**). For example, if you want a value of **499.9**, enter 04999.
- For the manual output, the implied decimal point is **2** places from the right (causing the resolution to be **0.01**). For example, if you want a value of **49.99%**, enter 04999.

Operating Commands to Loops 1–4

Loop / Word #							Set a Bit or Enter a Value															
1	**2**	**3**	**4**	**Bit**	**To Control**	**Bit Select or Range**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	1	2	3	0	Loop control	Disable = 0; Enable = 1																*x*
				1	Auto/manual mode	Manual = 0; Auto = 1															*x*	
				2	Setpoint select	Standby = 0*****; Run = 1******														*x*		
				3	Autotune enable	Disable = 0; Enable = 1													*x*			
				4	PID integral reset*******	Accume = 0; Reset = 1												*x*				
				5	Ramp enable	Disable = 0; Enable = 1											*x*					
				6	Ramp hold	Hold = 0; Don't hold = 1										*x*						
				7-15	reserved																	
4	5	6	7	0-15	Run temp setpoint	0.0 thru 3276.7°																
8	9	10	11	0-15	Manual output	–100.00 thru +100.00%																

*****For loops 1–4, standby setpoint is stored in M1, 5, 30, 50, 80, respectively ******entered below *******requires a 0-to-1 transition for each reset

Global Commands to All Loops

Word	Bit	To Control	Selected By	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
12	0	Temperature units	°F = 0; °C =1																
	1	Autotune invoke	Invoke = 1; None = 0																
	2	Autotune abort	Abort = 1; None = 0																
	3	Reset error codes	None = 0; Reset = 1																
	3-7	reserved																	
	8-10	Selection of Reported value (input image table) See **Important** below.	Current setpoint						0	0	1								
			Current error value						0	1	0								
			Current CV (loop output)						0	1	1								
			Current error code						1	0	0								
			Cold junction temperature						1	0	1								
			Firmware revision number						1	1	0								
	11	reserved																	
	12	M0 download request	None = 0; Download = 1																
	13	M1 download request	None = 0; Download = 1																
	14	M0 upload request	None = 0; Upload = 1																
	15	M1 upload request	None = 0; Upload = 1																
13	0-15	reserved																	
14	0-15	Calibration word																	
15	0-15	reserved																	

Important: The sample program returns all six variables. For their data table locations, refer to Data Table Layout in chapter 9.

Remember that the module returns the control variable (CV) of each loop to the input image table as both a numeric value (current CV) and a time-proportioned output (TPO). For additional information, refer to Module Output (chapter 2) and Data Table Layout (chapter 9).

Monitoring Status Data

This chapter describes status data reported by the BTM module in the input image table (16 words), applicable to the sample program.

Input Image Table

Implied Decimal Point

You must interpret the value of displayed 16-bit integer numbers. For temperature values reported in words 0–3, the implied decimal point is **1** place from the right (for a resolution to be **0.1**). For example, if 4999 is displayed, you must interpret it as **499.9**.

Status Values from Each Loop

1	2	3	4	Bit	Define	M1*	Indicated By	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	2	3	Current temperature (–3276.8° thru + 3276.7°)																			
4	5	6	7	0	Open circuit		0 = None; 1 = Error																
				1	Under range		0 = None; 1 = Error																
				2	Over range		0 = None; 1 = Error																
				3	Configuration		0 = None; 1 = Valid																
				4	Parameter value		0 = None; 1 = Error																
				5	PV rate alarm	10	0 = None; 1 = Alarm																
				6	Thermal runaway	16	0 = None; 1 = Alarm																
				7	High CV limit	2	0 = None; 1 = Alarm																
				8	Low CV limit	3	0 = None; 1 = Alarm																
				9	Low temperature	11	0 = None; 1 = Alarm																
				10	High temperature	12	0 = None; 1 = Alarm																
				11	Low deviation	13	0 = None; 1 = Alarm																
				12	High deviation	14	0 = None; 1 = Alarm																
				>15	reserved																		
8	9	10	11	0	Loop control		0 = No; 1 = Enabled																
				1	Loop mode		0 = Manual; 1 = Auto																
				2	Setpoint	5	0 Standby; 1 = Run																
				3	Autotune complete		0 = No; 1 = Yes																
				4	Autotune success		0 = No; 1 = Yes																
				5	Setpoint ramping		0 = No; 1 = Enabled																
				6	Heat TPO		0 = Off; 1 = On																
				7	Cool TPO		0 = Off; 1 = On																
				8-15	global status, see next page																		
12	13	14	15	Function and value of this word is set by O:e.12/bits 8–10. See table below.																			

***For reference, we give you the loop-1 word number of the alarm or limit value in the M1 file.**

Values reported in words 12–15 for loops 1-4 vary, depending the bit code reported in input image word I:e.8/bits 08–10. You must interprit the reported value according to the implied decimal point:

If I:e.8/10-09-08	reports:	implied decimal pt. is:	interprit:	as:
0 0 1 0 1 0 1 0 1	current setpoint, current error, or cold-junction temperature	1 decimal place (from the right)	04999	499.9
0 1 1	current CV (analog output)	2 decimal places	04999	49.99
1 0 0 1 1 0	error code, or firmware revision number	none	04999	4999

Global Status from All Loops

Word	Bit	Define	Indicated By	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
8 upper byte	8-10	Selection of Reported values See **Important** below.	Current setpoint						0	0	0								
			Current error value						0	0	1								
			Current CV (loop output)						0	1	0								
			Current error code						0	1	1								
			Cold junction temperature						1	0	0								
	11	Autotune progress	0 = None; 1 = In progress																
	12	Cold junction low	0 = None; 1 = Alarm																
	13	Cold junction high	0 = None; 1 = Alarm																
	14-15		reserved																
9 upper byte	8-11		reserved																
	12	M0 download	0 = None; 1 = Download																
	13	M1 download	0 = None; 1 = Download																
	14	M0 upload	0 = None; 1 = Upload																
	15	M1 upload	0 = None; 1 = Upload																

Important: The sample program returns all six variables. For their data table locations, refer to Data Table Layout in chapter 9.

Remember that the module returns the control variable (CV) of each loop to the input image table as both a numeric value (current CV) and a time-proportioned output (TPO). For additional information, refer to Module Output (chapter 2) and Data Table Layout (chapter 9).

Calibrating the Module

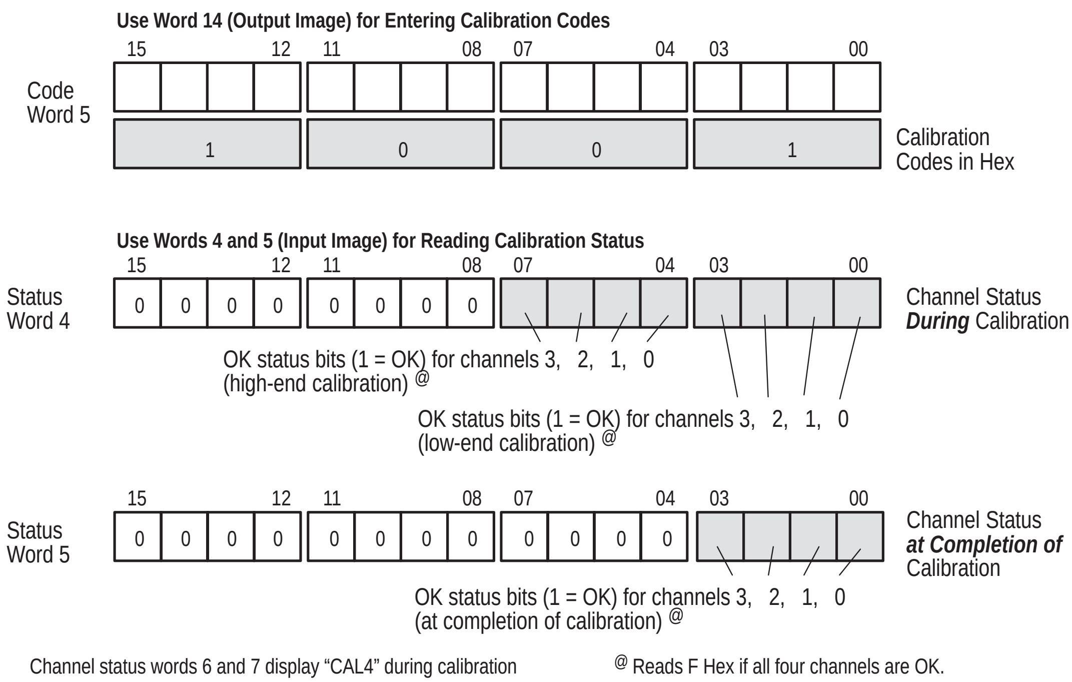
About the Procedure

Calibrate the module after the first 6 months of operation. Then check the calibration and recalibrate only if necessary once a year.

Use this procedure to store calibration values for each channel in EEPROM. Calibration sets channel accuracy at 0.05% of full range regardless of channel circuit tolerances. You can calibrate the input channels individually or in groups. The thermocouple/mV operation of all channels is suspended during calibration.

Calibration Codes and Status

Use the following format for entering calibration code words and reading calibration status bits. Enter calibration values in hexadecimal. You read channel status bits at different steps in the calibration procedure, one bit for each channel you are calibrating.



Calibration Procedure

To calibrate the module, you need a precision dc voltmeter and precision power supply that can display and maintain a calibration voltage to 1/1000 of a millivolt: at 0.000 mV and 90.000 mV.

For convenience, calibrate all four channels at the same time.

To prepare for the calibration:

- Remove the thermocouple leads from the input terminals of the channels that you want to calibrate.
- Switch the SLC processor to run mode so it can execute the calibration ladder logic.

To calibrate the module, follow this procedure:

1. With your programming terminal, enter calibration code 1001 Hex into output word 14.
2. Observe input words 0-3, 6 and 7.

   The module clears words 0–3. The module returns “CAL4” Hex in words 6 and 7.
3. Short circuit the pairs of input terminals for the channels you want to calibrate. Make the jumper as short as possible.
4. With your programming terminal, enter calibration code 1002 Hex into output word 14.
5. Observe bits 0-3 in input word 4.

   If all the channels you are calibrating see zero voltage, the module returns status-OK bits set, one bit for each channel (F Hex for all four channels). Otherwise, the module returns channel status bits set to zero.
6. Apply 90.000 mV to the pairs of input terminals, all in parallel, for the channels you are calibrating. Use short leads.
7. With your programming terminal, enter calibration code 1004 Hex into output word 14.
8. Observe bits 4-7 in status word 4.

   If all channels being calibrating see 90.000 mV, the module returns a status-OK bit set for each channel (F Hex for all four channels). Otherwise, the module returns channel status bits set to zero.
9. Remove the 90.000 mV calibration voltage.
10. With your programming terminal, enter calibration code 1008 Hex into output word 14.
11. Observe bits 0-3 in status word 5.

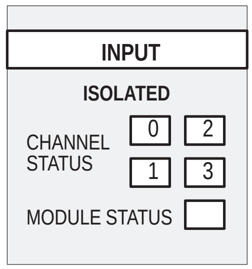
    After the module burns the calibration values into its EEPROM, it returns status-OK bits set, one bit for each channel (F Hex for all four channels). If the module could not complete the calibration of one or more channels, it returns a zeroed status bit for that channel (non-F Hex returned)
12. To end the calibration procedure, enter calibration code 0000 Hex into output word 14. During thermocouple/mV operation, word 14 must be zero (which is normal operation).

Troubleshooting the Module

This chapter provides troubleshooting guidelines.

Troubleshooting with LED Indicators

The front panel of the module contains five green LED indicators for channel status and one green LED indicator for module status.



LED:	When:	Indicates
Channel Status	On	channel is correctly configured when you enable the channel
	Flashing	channel fault condition
Module Status	On	self-check completed OK
	Flashing	communication occurring between SLC processor and BTM module

We present a table of indications, probable causes, and recommended action. See page 2 for a listing of error codes.

Indication:	Probable Cause:	Recommended Action:
all indicators are OFF	no power to module	check power to I/O chassis recycle power as necessary
	module performing self-check	wait until self check is complete
	module performing calibration	wait until calibration is complete
	possible short on the module LED failure	replace module
channel status indicator is ON	channel is correctly configured when you enable the channel	normal operation
	during calibration, the channel is properly configured for the high-end of millivolt range	normal calibration
channel status indicator is flashing	fault condition, such as open circuit or an under/over range condition	correct fault condition
	during calibration, the channel is properly configured for the low-end of millivolt range	normal calibration
module status indicator is ON	self-check is completed satisfactorily module is OK	normal power-up module waiting for channels to be enabled
module status indicator is flashing	communication occurring between SLC processor and the BTM module	normal operation

Locating Error Code Information

You configure the BTM module to report error codes by setting bits 10-8 to 100 in word 12 of the output image table. (See chapter 5.) The module reports error codes in input image table words 12–15. If using the sample program, error codes are reported in N7:185–188. Location of the 2-digit code in the error word determines its source:

Display of error code in N7:185–188 if using example program — **xxyy**

Autotune error code (xx) **Configuration error code** (yy)

Example: If 0024 is displayed, the error code is from the configuration block.

Code:	Description:
0	No error
1	Run setpoint is invalid
2	Manual control value is < –100% or > +100%
10	Heat system gain is less than 0
11	Heat time constant is less than 0
12	Heat dead time is less than 0
13	Cool system gain is less than 0
14	Cool time constant is less than 0
15	Cool dead time is less than 0
20	Heat proportional gain is less than 0
21	Heat integral gain is less than 0
22	Heat derivative gain is less than 0
23	Cool proportional gain is less than 0
24	Cool integral gain is less than 0
25	Cool derivative gain is less than 0
30	Loop operational mode is invalid
31	Thermocouple type is invalid
32	Maximum CV allowable is < –100%, > +100%, or max CV < min CV
33	Minimum CV allowable is < –100% or > +100%
34	Control value when TC break detected is< –100 or > +100%
35	Standby setpoint is invalid
36	Heat minimum TPO cycle time is < 0 or > 100
37	Heat maximum TPO cycle time is < 0, > 100, or min TPO > max TPO
38	Cool minimum TPO cycle time is < 0 or > 100
39	Cool maximum TPO cycle time is < 0, > 100, or min TPO > max TPO
40	PV rate alarm degrees per second is invalid
41	Low temperature alarm is invalid
42	High temperature alarm is invalid
43	Low deviation alarm is invalid
44	High deviation alarm is invalid
45	Temperature alarm deadband is < 0 or > 10
46	Runaway temperature change value is < 0– or > 100
47	Runaway period in minutes < 0 or > 100
48	Setpoint ramp rate is < 0
49	reserved
50	Nonbarrel autotune disturbance size is < 0% or greater than 100%
51	Startup aggressiveness factor is <0 or > 100
66	Autotune terminated because of TC break
67	Start conditions prevent heat autotune
68	Start conditions prevent cool autotune
69	Setpoint will be reached before autotuning is complete
70	Too much noise causing time constant to be 0
71	Very small gain
72	PV now higher than maximum PV; do not trust PID values

Sample Program

This chapter describes:

- Obtaining the sample program from the internet
- Configuring Your SLC processor, Off-line
- Using the Sample Program

Obtaining the Sample Program from the Internet

You can obtain the sample program from the Allen-Bradley website on the Internet and download it to your PC as an executable file.

To Access the Internet:

1. Access the Allen-Bradley website at: http://www.ab.com with extensions to:
   – program: /mem/appsys/prodinfo/applac/appla/BTMsw/
   – manual: /manuals (Application Systems Library, pub 1746-6.10)
2. Download the sample program to your PC.
3. Move the sample program into the subdirectory on your hard drive where your programming software looks for files.
   For example, with RSLogix500:
   C:\program files\rockwell software\rslogix 500 english\projects

Configuring Your SLC Processor, Off-line

This procedure uses RSLogix500 version 2.0 or later. For other types of programming software, the procedure and/or prompts may vary.

You must modify the I/O configuration to match your system layout and change associated addresses, offline.

1. Open the ladder file (for example, BTM.RSS) from the file pull-down window.
2. Select the I/O Configuration icon and launch it. Then select:
   A. Processor type (sample program is SLC 5/04).
   B. Module slot number (sample program is slot 1 in 4-slot chassis).
   C. Module ID is 10223 entered under "Other".
3. The sample program creates M0/M1 files:
   M1 = 101 words; M0 = 50 words
4. If you move the module to another slot:
   - Be sure to retain the files.
   - Search and replace all addresses that must be changed.
5. Save the file as a new file to preserve the original file for backup.
6. Load the ladder file into the SLC processor.

Using the Sample Program

The sample program uses files 2 and 5, with the BTM module in slot 1. If you move this logic to other program files or locate the module in another slot, you must search for and change addresses, accordingly.

What You Must Do

To make this program work in your application:

1. Assign an output module to slot 2.

If using:	Then you need:
TPO	8 outputs per BTM module (four outputs for heat TPO bits; four outputs for cool TPO bits). If your system is a heat-only, use 4 outputs for the heat TPO bits. Refer to sample program page 1 for heat/cool TPO bits.
current CV	an analog output module with at least four channels per BTM module

2. Write your own ladder logic to interface to the sample program.

If using:	Then write ladder logic to:	and refer to:
TPO	not required	n/a
current CV	1. read current CV in N7:180-183 for loops 1-4 2. scale current CV to match the analog output module and channel application	n/a
either one	1. manipulate words/bits in output image table to control: – an individual loop – all loops (global command) 2. read words/bits in the input image table to verify events	chapter 5 chapter 6

Data Table Layout for the Sample Program

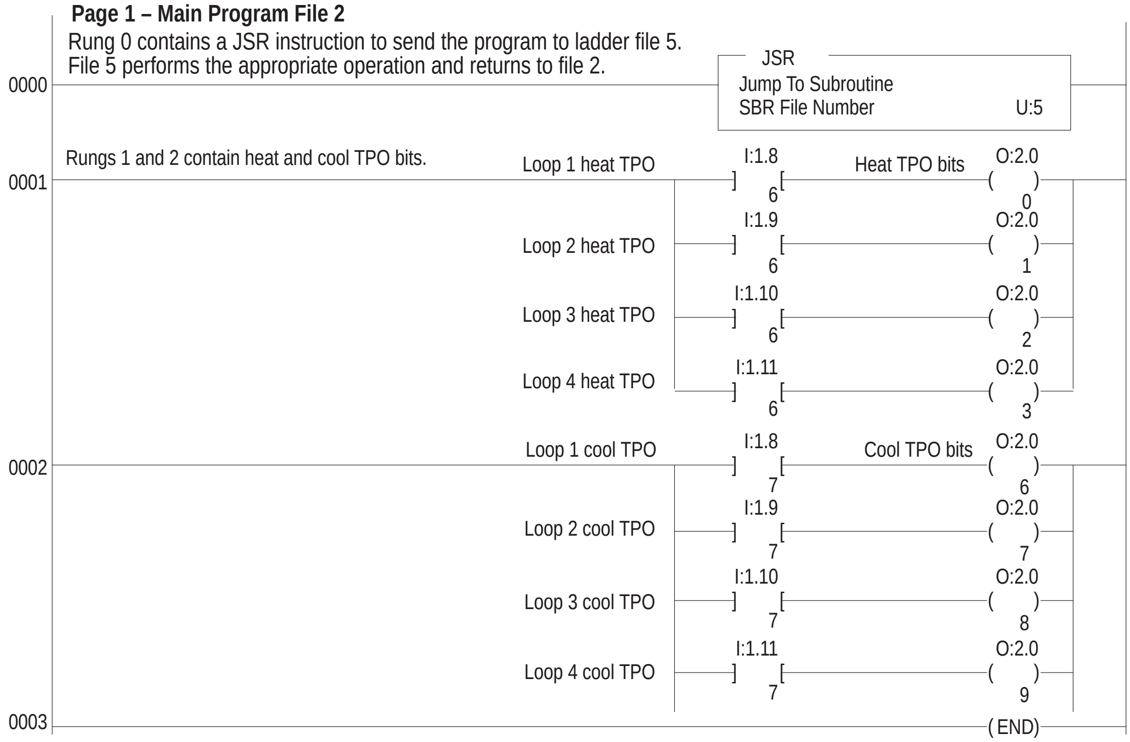
The sample program uses data table file N7. If you need to change this file address, re-address the sample program to the new file.

When you create File N7, allocate 200 elements/words as follows:

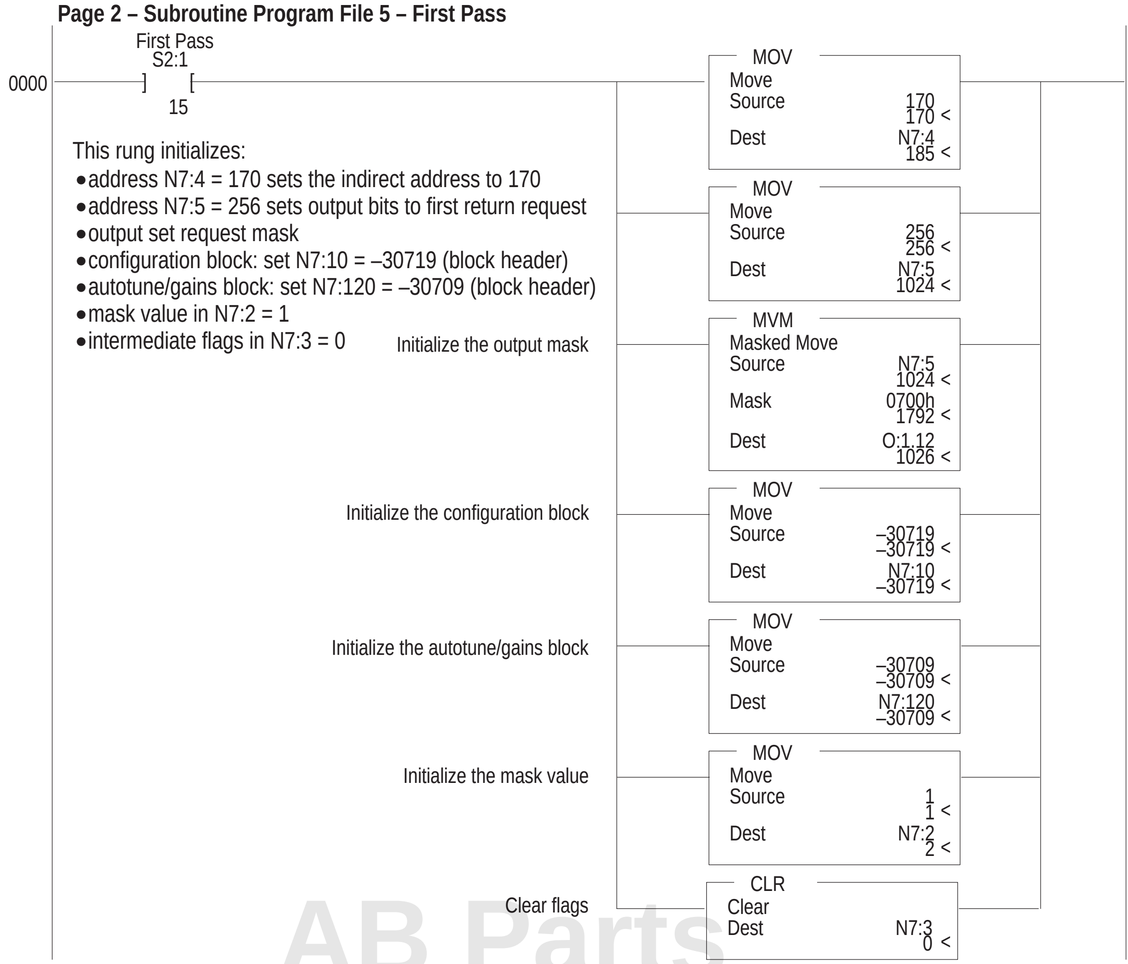
Word:	Definition:
N7:0	user request bits
N7:1–9	reserved
N7:10–110	configuration block
N7:111–119	reserved
N7:120–168	autotune/gains block
N7:169	reserved
N7:170–173	displays current setpoint for loops 1-4
N7:174	reserved
N7:175–178	displays current error for loops 1-4
N7:179	reserved
N7:180–183	displays current CV for loops 1-4
N7:184	reserved
N7:185–188	displays error code for loops 1-4
N7:189	reserved
N7:190–194	displays cold junction temperature
N7:195–198	displays firmware revision

Sample Program

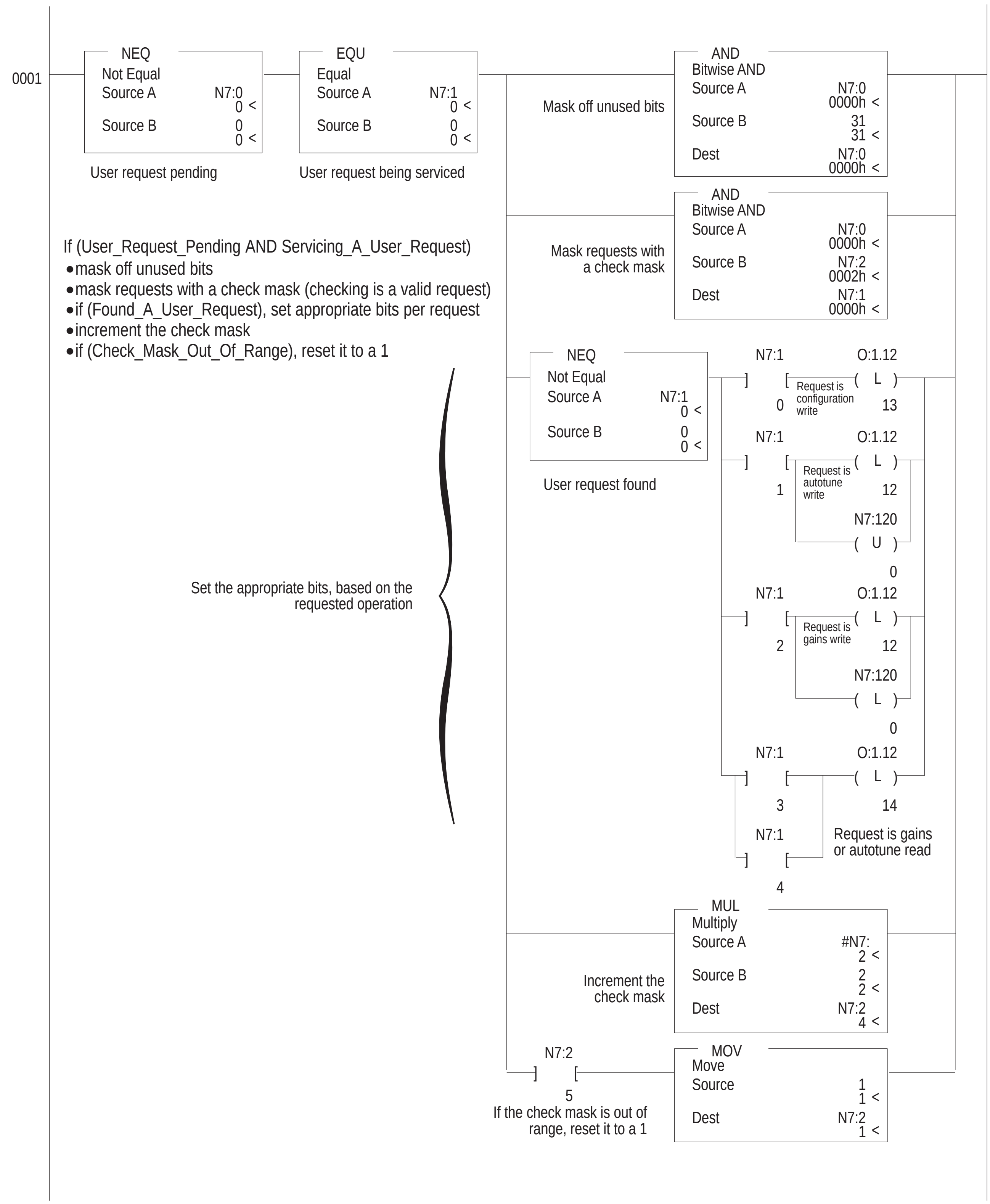
Page 1 – Main Program File 2



Page 2 – Subroutine Program File 5 – First Pass



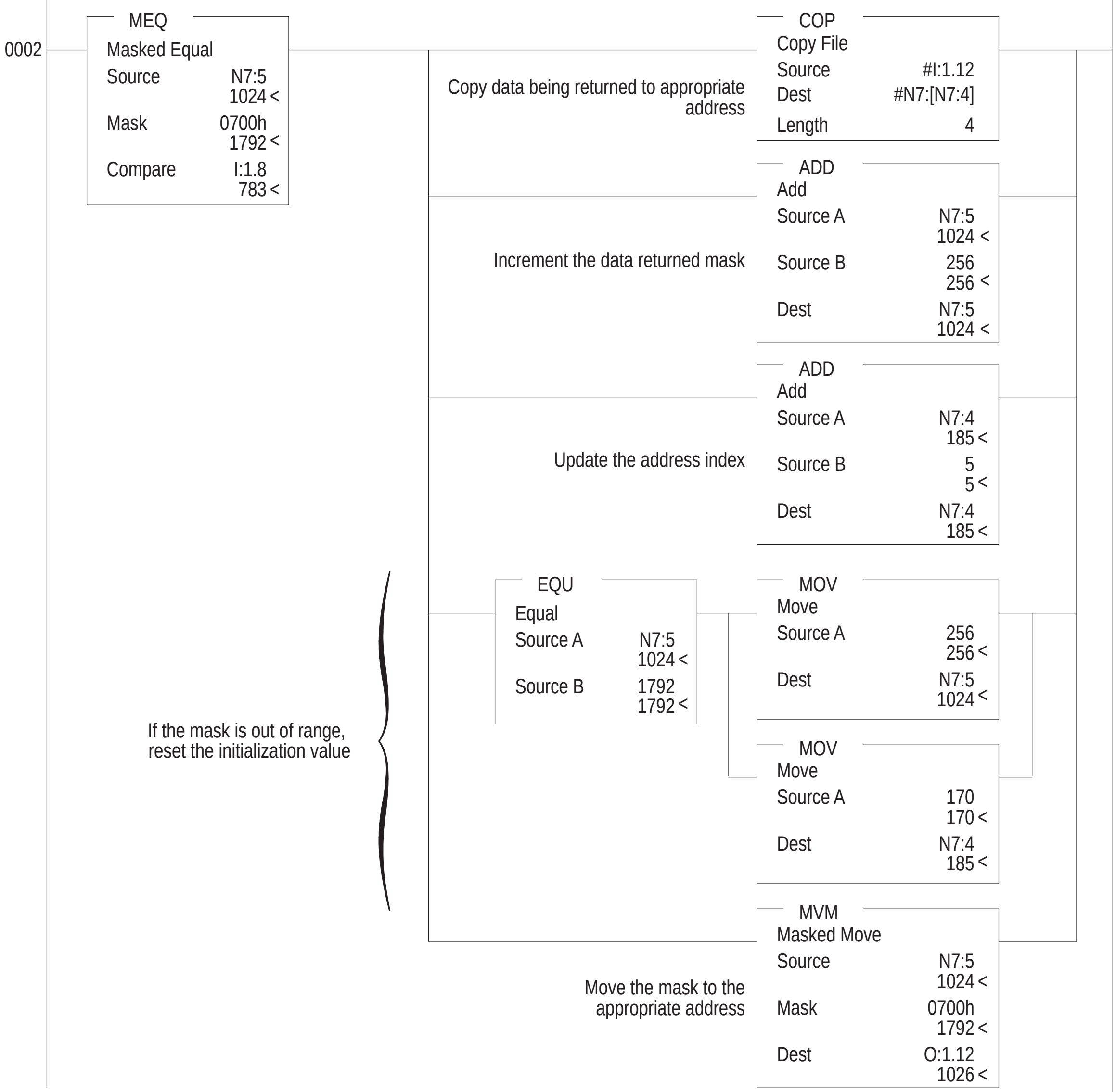
Page 3 – Subroutine Program File 5 – User Request is Pending while Servicing a User Request



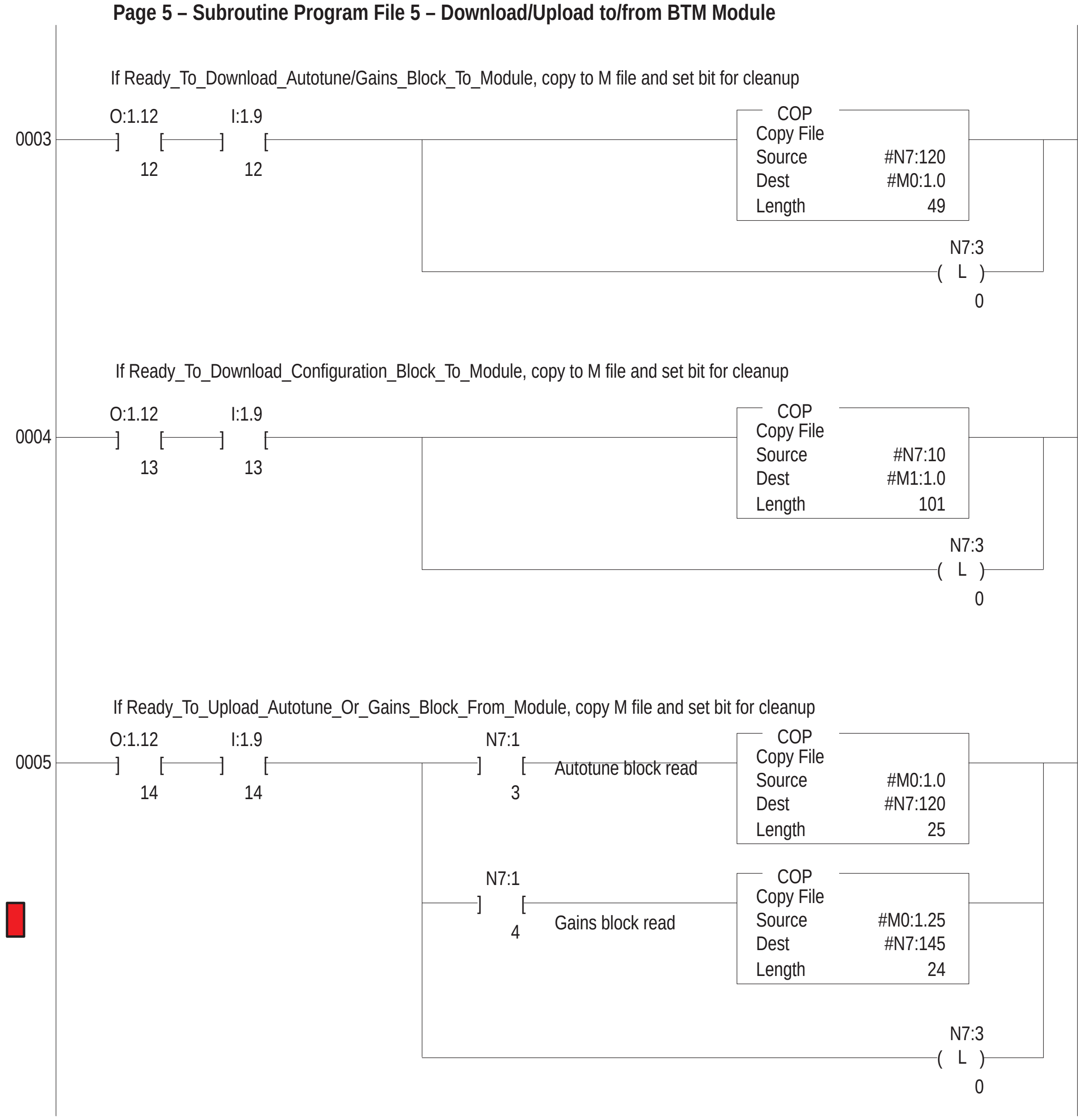
Page 4 – Subroutine Program File 5 – If Data_Return_Request = Data_Being_Returned

Important: If using an SLC 5/02 processor, you cannot use this rung, See last page for equivalent logic.

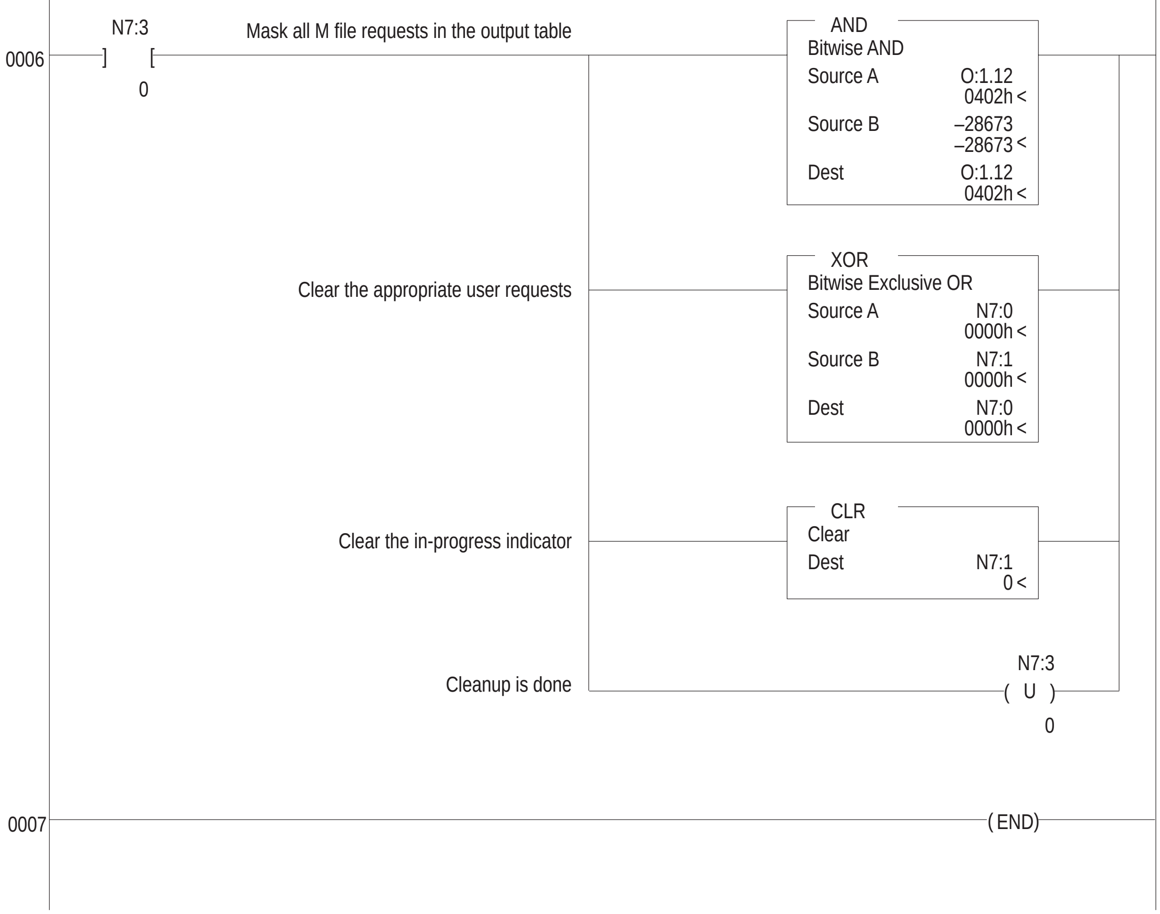
If (Data_Return_Request = Data_Being_Returned
- copy data being returned to the appropriate address
-)increment the Data_Returned_Mask
- update the address index
- if (Mask_Is_Out_Of_range) reset to the initialization values
- move the mask to the appropriate output location



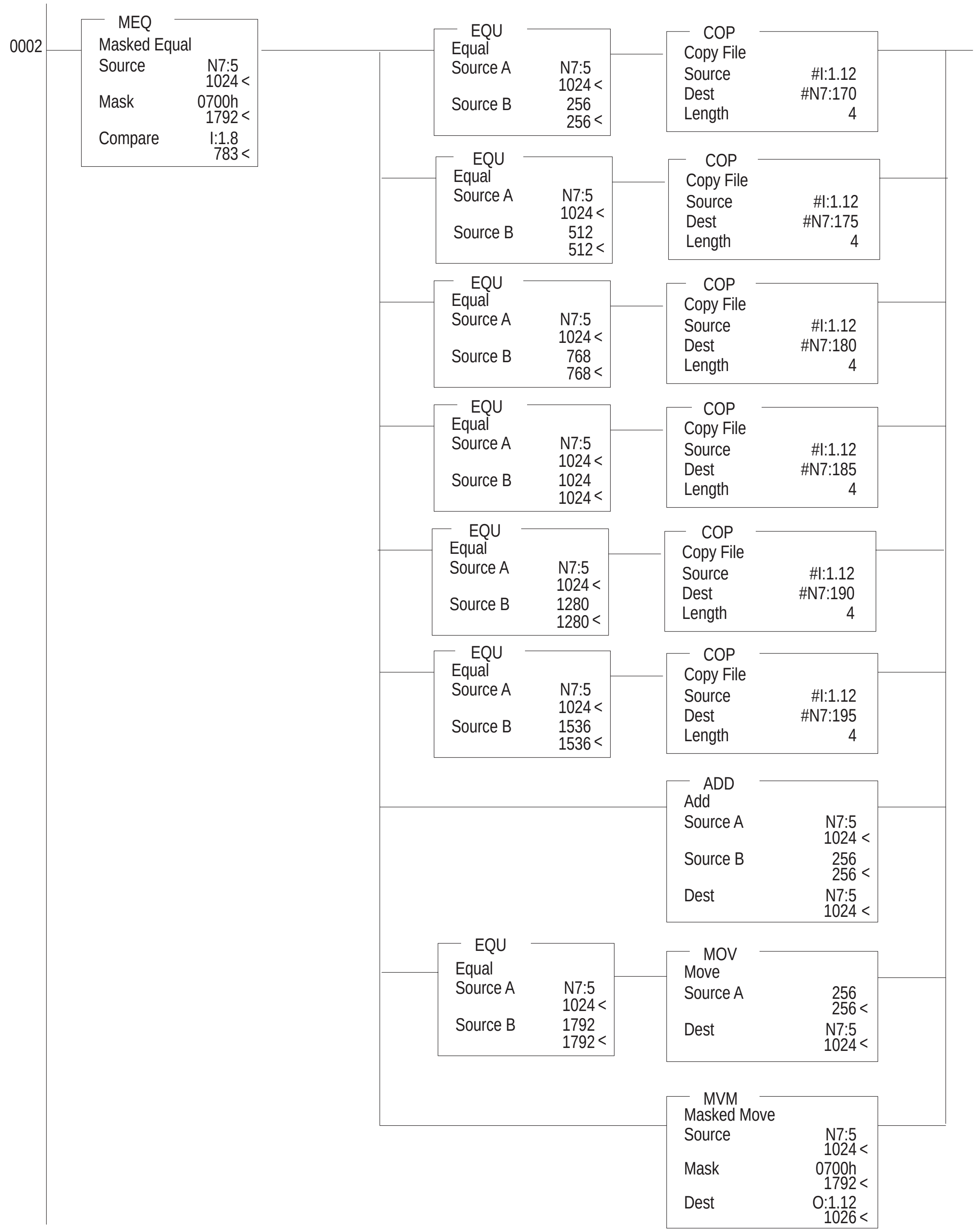
Page 5 – Subroutine Program File 5 – Download/Upload to/from BTM Module



Page 6 – Subroutine Program File 5 – If Cleanup_Reqired



Page 4a – For SLC 5/02 Processor
Subroutine Program File 5 – If Data_Return_Request = Data_Being_Returned



Module Specifications

This appendix lists specifications for the 1746-BTM barrel temperature control module and lists thermocouple requirements.

Electrical Specifications

Backplane current consumption	110 mA at 5V dc 85 mA at 24V dc
Backplane power consumption	0.6W maximum (0.55W @ 5V dc, 2W @ 24V dc)
Number of channels	4 (backplane and channel-to-channel isolated)
I/O chassis location	any I/O module slot except slot 0
A/D conversion method	sigma-delta modulation
Input filtering	analog filter with low pass digital filter
Normal mode rejection (between [+] input and [–] input)	greater than 50 dB at 50 Hz greater than 60 dB at 60 Hz
Common mode rejection (between inputs and chassis ground)	greater than 120 dB at 50/60 Hz (with 1K ohm imbalance)
Channel bandwidth (–3db)	8 Hz
Calibration	once every six months
Isolation	1000 V transient or 150 VAC continuous channel-to-channel or channel-to-backplane

Physical Specifications

LED indicators	5 green status indicators one for each of 4 channels one for module status
Module ID code	10223
Recommended cable: for thermocouple inputs for mV inputs	 shielded twisted pair thermocouple extension wire[1] alpha 5121 or equivalent
Maximum wire size	two 14 AWG wires per terminal
Maximum cable impedance	150 ohms maximum loop impedance, for <1LSB error
Terminal strip	removable, Allen-Bradley spare part catalog number 1746–RT32

[1] Refer to the thermocouple manufacturer for the correct extension wire.

Environmental Specifications

Operating temperature	0°C to 60°C (32°F to 140°F)
Storage temperature	–40°C to +85°C (–40°F to +185°F)
Relative humidity	5% to 95% (without condensation)
Agency Certification (when product or packaging is marked)	CSA Class 1 Div 2 Hazardous CE marked for all applicable directives

Input Specifications

Type of input (selectable)	Thermocouple Type J –210°C to 760°C (–346°F to 1400°F) Thermocouple Type K –270°C to 1370°C (–454°F to 2498°F) Millivolt (–50 mV dc to +50 mV dc) Millivolt (–100 mV dc to +100 mV dc)
Thermocouple linearization	IPTS–68 standard, NBS MN–125, NBS MN–161
Cold junction compensation	accuracy ±1.5°C, 0°C to 70°C (32°F to 158°F)
Input impedance	greater than 10MΩ
Temperature scale	0.1°C or 0.1°F
DC millivolt scale	0.01 mV
Open circuit detection leakage current	20 nA typical
Open circuit detection	upscale
Time to detect open circuit	0.5 seconds, typical
Input step response	0 to 99.9% (less 1 LSB) in 600 ms (worst case)
Input resolution	see input resolution graphs on following pages these graphs show the smallest measurable value based on combined hardware and software tolerances.
Display resolution	0.1°C/step or 0.1°F/step
Overall module accuracy @ 25°C (77°F)	see module accuracy table, page A–3
Overall module accuracy (0°C to 60°C, 32°F to 140°F)	see module accuracy table, page A–3
Overall module drift	see module accuracy table, page A–3
Module update time	less than 500 ms
Channel turn-off time	up to one module update time

Overall accuracy

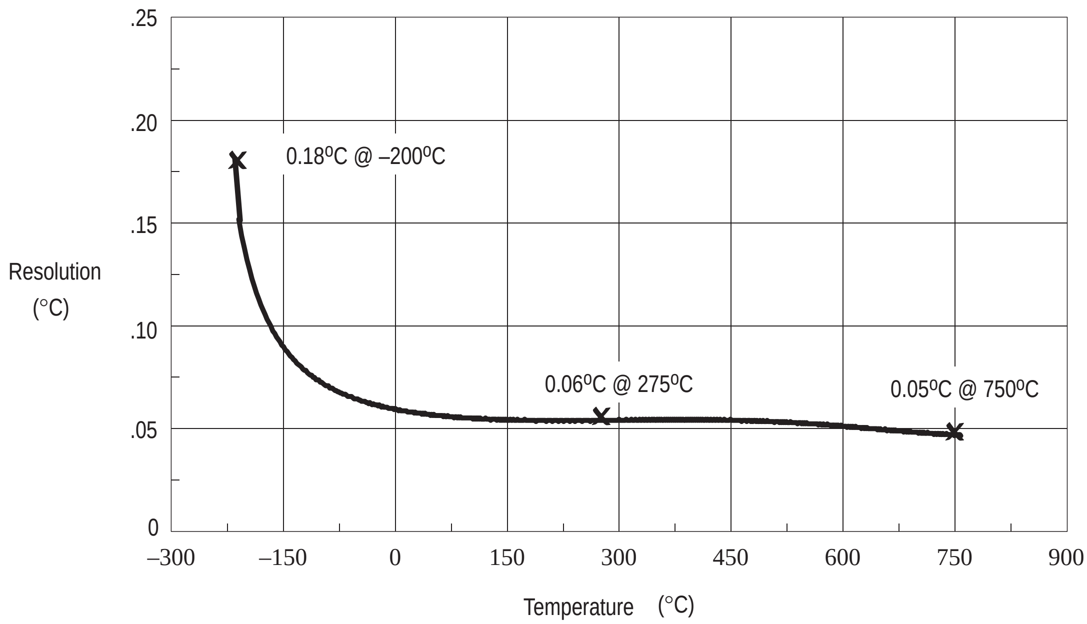
Overall accuracy to includes variances in cold-junction compensation, calibration, non-linearity, and resolution.

Input Type	Maximum Error① @ 25°C	Maximum Error① @ 77°F	Temperature Drift① (0°C–60°C)	See page
J	±1.06°C	±1.91°F	±0.0193°C/°C, °F/°F	A–3
K	±1.72°C	±3.10°F	±0.0328°C/°C, °F/°F	A–4
±50 mV	±30 μV	±30 μV	±1.0 μV/°C, ±1.8 μV/°F	—
±100 mV	±30 μV	±30 μV	±4.0 μV/°C, ±2.7 μV/°F	—

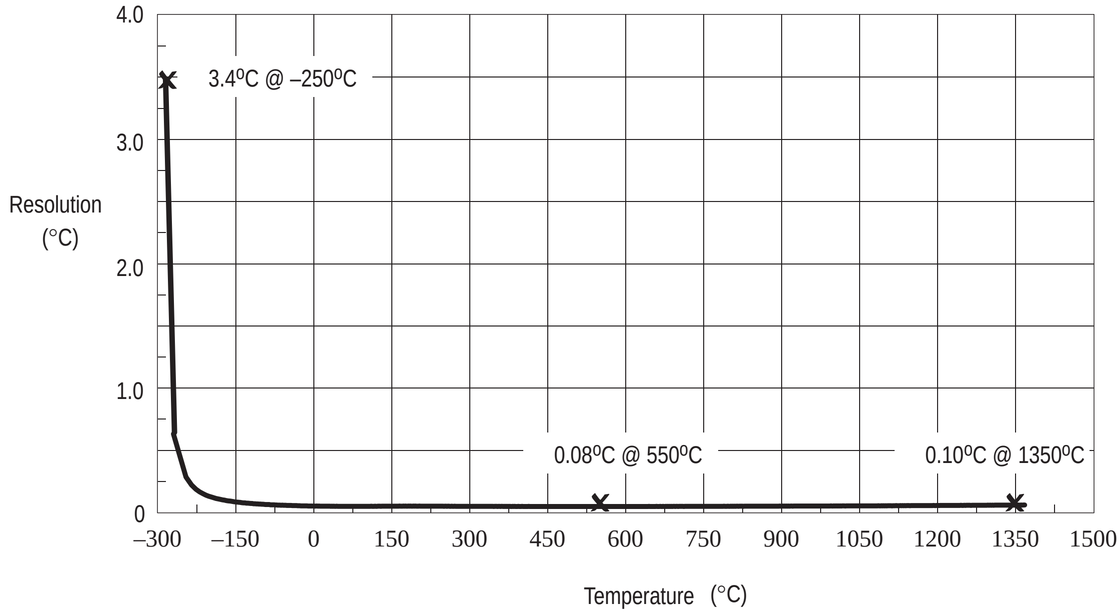
① Assumes the module terminal block temperature is stable.

Thermocouple Resolution

Type J Thermocouple
Iron vs. Copper–Nickel <Constantan>



Type K Thermocouple
NIckel-Chromium or Nickel-Aluminum



Loop Data

We present loop data in worksheet format that includes:

- Configuration Values (M1 file, N7:10–110)
- Autotune/Gains Values (M0 file N7:120–168)
- Loop Operating Commands
- Loop Status Values
- Global commands to all four loops
- Global status from all four loops

Left-hand columns list word numbers for each of the four loops.

Important: Because loop values are stored and reported in integer files, you must understand the meaning of implied decimal point (IDP). Otherwise, the magnitude of your intended value may be in error by as much as 1000, depending on the position of the implied decimal point.

When entering or reading integer (counting number) values, the *range*, given in the associated table, tells you the implied decimal point. You will probably need to use leading zeros when storing a value.

Parameter	Given Range	IDP*	Example
Thermal Runaway	0–100º	0	If you want to store a value of 66º, enter 00066.
Standby Setpoint	0.0 thru 32767.7º	1	If you want to store a value of 660.0º, enter 06600.
TPO Period	0.00 thru 100.00 sec	2	If you want to store a value of 6%, enter 00600.
High CV Limit	–100.00 thru +100.00	2	If you want to store a value of –09%, enter –00900.
Cool Proportional	0.000 thru 32.767	3	If you want to store a value of 18, enter 18000.
Heat Integral	0.0000 thru 3.2767	4	If you want to store a value of 0.5, enter 05000.

*IDP indicates the number of digits *from* the *right* that locates the implied decimal point.

You read status values similarly. You must know the *range* of the value to read it correctly. For example, if you are reading a temperature (such as standby), a display of 12345 would have a value of 1234.5º.

Configuration Values (M1 File, N7:10–110): Block Header (word 0 / N7:10) = 8801 (–30719 decimal)

Important: For the corresponding address N7:xxx, add 10 to the loop word numbers 1-99.

Loops 1-4 / Word #							Set a Bit or Enter a Value															
1	2	3	4	Bit	To Configure	Bit Select or Range	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	26	51	76	0–1	Operation mode	Monitor; no PID control															0	0
						Control loop with PID															0	1
						Disable loop															1	0
				2–5	Input type	Type J											0	0	0	0		
						Type K											0	0	0	1		
				6	Alarm enable	Disable = 0; Enable = 1										x						
				7–8	TC break configuration	Disable PID loop (CV=0)								0	0							
						Use thermal runaway CV								0	1							
						Use manual mode CV								1	0							
				9	reserved																	
				10-11	Autotune gains	Low gains					0	0										
						Medium gains					0	1										
						High gains					1	0										
						Very high gains					1	1										
				12	Barrel control	Barrel = 0; Non-barrel = 1				x												
				13	Zone	Inner = 0; Outer = 1			x													
				14-15	reserved																	
2	27	52	77	0–15	High CV limit %	–100.00 thru +100.00%	default = +100.00%															
3	28	53	78	0–15	Low CV limit %	–100.00 thru +100.00%	default = 0.00%															
4	29	54	79	0–15	CV for TC break	–100.00 thru +100.00%	default =															
5	30	55	80	0–15	Standby setpoint	0.0 thru 3276.7°	default = 0.0															
6	31	56	81	0–15	Heat on time (min)	0.00 thru 100.00 sec	default = 0.00															
7	32	57	82	0–15	Heat TPO period	0.00 thru 100.00 sec	default = 5.00															
8	33	58	83	0–15	Cool on time (min)	0.00 thru 100.00 sec	default = 0.00															
9	34	59	84	0–15	Cool TPO period	0.00 thru 100.00 sec	default = 5.00															
10	35	60	85	0–15	PV alarm rate	–3276.8 thru +3276.7°/s	default = 0.0															
11	36	61	86	0–15	Low temp alarm	–3276.8– thru +3276.7°	default = +999.9															
12	37	62	87	0–15	High temp alarm	–3276.8 thru +3276.7°	default = +999.9															
13	38	63	88	0–15	Low deviation	–3276.8 thru +3276.7°	default = +999.9															
14	39	64	89	0–15	High deviation	–3276.8 thru +3276.7°	default = +999.9															
15	40	65	90	0–15	Alarm dead band	0.0 thru 10.0°	default = 0.0															
16	41	66	91	0–15	Thermal runaway	0 thru 100°	default = 5															
17	42	67	92	0–15	Runaway period	0 thru 100 minutes	default = 20															
18	43	68	93	0–15	Heat ramping	0 thru 100°/min	default = 0															
19	44	69	94	0–15	Cool ramping	0 thru 100°/min	default = 0															
20	45	70	95	0–15	Nonbarrel autotune disturb size (0.00–100.00%)		default = 10.00															
21	46	71	96	0–15	Startup aggressiveness factor (0 thru 100)		default = 0 for heat or cool, only; 25 for heat/cool															
>25	>50	>75	>99	reserved																		

Autotune/Gains Values, M0 File (N7:120–168): Block Header N7:120 (word 0) = 880B (–30709 decimal)

Important: For the corresponding address N7:xxx, add 120 to the loop word numbers 1-48.

Loops 1–4				Autotune Values (N7:120–144)		
1	2	3	4	To Configure	Range	Enter a Value
1	7	13	19	Heat gain	0.00 thru 327.67°/sec	
2	8	14	20	Heat time constant	0.0 thru 3276.7 sec	
3	9	15	21	Heat dead time	0.0 thru 3276.7 sec	
4	10	16	22	Cool gain	0.00 thru 327.67°/sec	
5	11	17	23	Cool time constant	0.0 thru 3276.7 sec	
6	12	18	24	Cool dead time	0.0 thru 3276.7 sec	
Loops 1–4				**Gains Values (N7:145–168)**		
1	**2**	**3**	**4**	**To Configure**	**Range**	**Enter a Value**
25	31	37	43	Heat proportional	0.000 thru 32.767 sec	
26	32	38	44	Heat integral	0.0000 – 3.2767 rpts/sec	
27	33	39	45	Heat derivative	0.0 thru 3276.7 sec	
28	34	40	46	Cool proportional	0.000 thru 32.767 sec	
29	35	41	47	Cool integral	0.0000 thru 3.2767 rpts/sec	
30	36	42	48	Cool derivative	0.0 thru 3276.7 sec	

Loop Operating Commands (output image table)

Loop / Word #							Set a Bit or Enter a Value															
1	2	3	4	Bit	To Control	Bit Select or Range	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	2	3	0	Loop control	Disable = 0; Enable = 1																x
				1	Auto/manual mode	Manual = 0; Auto = 1															x	
				2	Setpoint select	Standby = 0; Run = 1														x		
				3	Autotune enable	Disable = 0; Enable = 1													x			
				4	PID integral reset	Accume = 0; Reset = 1												x				
				5	Ramp enable	Disable = 0; Enable = 1											x					
				6	Ramp hold	Hold = 0; Don't hold = 1										x						
				7-15	reserved																	
4	5	6	7	0-15	Run temp setpoint	0.0–3276.7°																
8	9	10	11	0-15	Manual output	–100.00 thru +100.00%																

Loop Status Values (input image table)

Loop / Word #							Read the Bit Status or Word Value															
1	2	3	4	Bit	Define	Indicated By	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	2	3	0-15	Current temperature																	
4	5	6	7	0	Open circuit	0 = No error; 1 = Error																
				1	Under range	0 = No error; 1 = Error																
				2	Over range	0 = No error; 1 = Error																
				3	Configuration	0 = No config; 1 = Valid																
				4	Parameter value	0 = No error; 1 = Error																
				5	PV rate alarm	0 = No alarm; 1 = Alarm																
				6	Thermal runaway	0 = No alarm; 1 = Alarm																
				7	High CV limit	0 = No alarm; 1 = Alarm																
				8	Low CV limit	0 = No alarm; 1 = Alarm																
				9	Low temperature	0 = No alarm; 1 = Alarm																
				10	High temperature	0 = No alarm; 1 = Alarm																
				11	Low deviation	0 = No alarm; 1 = Alarm																
				12	High deviation	0 = No alarm; 1 = Alarm																
				>15	reserved																	

Loop Status Values (continued)

Loop / Word #							Read the Bit Status or Word Value															
1	2	3	4	Bit	Define	Indicated By	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
8	9	10	11	0	Loop control	0 = Disabled; 1 = Enabled																
				1	Loop mode	0 = Manual; 1 = Auto																
				2	Setpoint	0 Standby; 1 = Run																
				3	Autotune complete	0 = No; 1 = Complete																
				4	Autotune success	0 = No; 1 = Success																
				5	Setpoint ramping	0 = Disabled; 1 = Enabled																
				6	Heat TPO	0 = Off; 1 = On																
				7	Cool TPO	0 = Off; 1 = On																
				8-15	global status, see last page																	
12	13	14	15	0-15	Set by word 12, bits 8–10, output image table																	

Global Commands to All Four Loops (output image table)

Word	Bit	To Control	Selected By	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
12	0	Temperature units	°F = 0; °C =1																
	1	Autotune invoke	Invoke = 1; None = 0																
	2	Autotune abort	Abort = 1; None = 0																
	3	Reset error codes	None = 0; Reset = 1																
	3-7	reserved																	
	8-10	Selection of Reported values	Current setpoint						0	0	1								
			Current error value						0	1	0								
			Current CV value						0	1	1								
			Current error code						1	0	0								
			Cold junction temperature						1	0	1								
			Firmware revision number						1	1	0								
	11	reserved																	
	12	M0 download request	None = 0; Download = 1																
	13	M1 download request	None = 0; Download = 1																
	14	M0 upload request	None = 0; Upload = 1																
	15	M1 uploard request	None = 0; Upload = 1																
13	0-15	reserved																	
14	0-15	Calibration word																	
15	0-15	reserved																	

Global Status from All Four Loops (input image table)

Word	Bit	Define	Indicated By	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
8 upper byte	8-10	Selection of Reported values	Current setpoint						0	0	0								
			Current error value						0	0	1								
			Current CV value						0	1	0								
			Current error code						0	1	1								
			Cold junction temperature						1	0	0								
	11	Autotune progress	0 = None; 1 = In progress																
	12	Cold junction low	0 = None; 1 = Alarm																
	13	Cold junction high	0 = None; 1 = Alarm																
	14-15	reserved																	
9 upper byte	8-11	reserved																	
	12	M0 download	0 = None; 1 = Download																
	13	M1 download	0 = None; 1 = Download																
	14	M0 upload	0 = None; 1 = Upload																
	15	M1 uploard	0 = None; 1 = Upload																

A

addressing, 1–3

alarms, 3-4
- dead band, 3–4
- loop enable, 3–1
- temperature deviation, 3–4

autotune
- disturbance size, non-barrel, 3-5
- gains level selection, 3-1
- loop and fine-tuning overview 4–2
- M0 file autotune/gains block
- PID equation, 4–3
- procedural overview, 4–1
- procedure, barrel control, 5-2
- procedure, non-barrel control, 5-4
- values, setting of, 4–4

B

barrel/non-barrel control, 3–2

C

cables, preparation & wiring, 2-6

calibration
- codes and status, 7–1
- procedure, 7–1

cold junction compensate CJC, 2–5

compatibility, system, P-1

configuration block, M1 file, 3-6, B-2

configuration:
- module, see loop configuration
- SLC processor, 9-1

control mode, selection of, 5–1

CV limits, 3-3

current CV, 1-2, 6-2, 9-2

D

data table, sample program, 9–2

dead band, 3–4

decimal point, implied 3–5, 4-4, B-1

deviation alarm, 3–4

disable module slot, 1–4

disturbance size, non-barrel, 3–5

download,
- required delay for autotune, 4-1
- required to operate module, 4-2

E

electrostatic damage, avoiding, 2–1

enclosure depth, 2–2

error codes, 8–1

European union directive, 2–1

F

features, of module, 1–2

files
- input image table, 6-1, B-3, B-4
- M0, autotune/gains, 4-4, B-3
- M1, configuration, 3-6, B-2
- output image table, 5-5, B-3, B-4

fine-tuning, loops, 4–3

first-scan logic, 9-2

G, H

gains
- block (M0 file), 4–4
- overview, 4–1
- PID equation, 4–3
- sequence to set, 4–1

grounding, 2-6

I, J, K

ID, of module, 9-1

I/O image table, addressing, 1-3

indicator lights, see LEDs

input devices, wired to module, 2–6

input image table, 6–1, B-3, B-4

installation, of module
- avoiding ESD, 2–1
- considerations, 2–2
- enclosure depth, 2–2
- module slot location, 2–2
- power requirements, 2–2
- procedure, 2–3

internet, access to, 9–1

L

ladder logic, see program, sample

LEDs, 8–1

loop autotune gains, 3–1

L (continued)

loop configuration, chapter 3
  alarm dead band, 3–4
  alarm enable, 3-1
  alarms, deviation & temp, 3–4
  autotune gain levels 3–1
  autotune disturbance size 3–5
  barrel/non-barrel control, 3–2
  configuration block (M1 file), 3–6
  CV limits, high/low, 3–3
  on-times, heat/cool, min, 3–3
  operating mode, 3–1
  PV rate & alarm, 3–4
  ramp rates, heat/cool 3–5
  standby setpoint, 3-3
  TC type, 3–1
  TC break control, 3–3
  TC break response, 3–1
  TPO period, heat/cool 3–4
  thermal integrity loss, 3–5
  zone, inner/outer, 3–2

M

M0 file, 4-4, B-3

M1 file, 3–6, B-2

module
  calibration, 7–1
  location in I/O chassis, 2–2
  outputs, 1–2, 5-2, 9-2
  specifications, A–1
  troubleshooting, 8–1

monitoring, status data
  input image table, 6–1

N

noise, electrical, reduction of, 2-6

O

on-time, min, 3-3

output image table, 5–5, B-3, B-4

output s, module, 1–2, 5-2, 9-2

overview, module 1–1

P

PID
  equation, 4–3
  gains, 4–1
  loop, 1–1

power requirements, 2–2

program, sample, chapter 9
  data table, 9-2
  internet access to, 9-1

R

ramp rates, heat/cool, 3-5

runaway, thermal, 3-5

S

SLC processor, configuration, 9-1

slot, for module
  addressing, 1-3
  location considerations, 2-2

slot disable, response to, 1-4

specifications A-1

standby setpoint, 3-3

status words, monitoring, 6–1

switching, barrel control, 3–2

T

TC (thermocouple) break control, 3–1, 3–3

TC resolution, A–3

TC type selection, 3–1

terminal block, removal, 2–4

thermal integrity loss, 3-5

TPO (time-proportioned output) 1-2, 9-2
  on-times, min, 3-3
  period, 3-4
  timing diagram, 5-2

troubleshooting
  autotune, 5–5
  error codes, 8–2
  LEDs, 8–1

V

vocabulary P–2

W

wiring, 2-4, 2-6
  considerations, 2–5

Z

zones, inner/outer 3–2

Allen-Bradley, a Rockwell Automation Business, has been helping its customers improve productivity and quality for more than 90 years. We design, manufacture and support a broad range of automation products worldwide. They include logic processors, power and motion control devices, operator interfaces, sensors and a variety of software. Rockwell is one of the world's leading technology companies.

Worldwide representation.

Argentina • Australia • Austria • Bahrain • Belgium • Brazil • Bulgaria • Canada • Chile • China, PRC • Colombia • Costa Rica • Croatia • Cyprus • Czech Republic • Denmark • Ecuador • Egypt • El Salvador • Finland • France • Germany • Greece • Guatemala • Honduras • Hong Kong • Hungary • Iceland • India • Indonesia • Ireland • Israel • Italy • Jamaica • Japan • Jordan • Korea • Kuwait • Lebanon • Malaysia • Mexico • Netherlands • New Zealand • Norway • Pakistan • Peru • Philippines • Poland • Portugal • Puerto Rico • Qatar • Romania • Russia–CIS • Saudi Arabia • Singapore • Slovakia • Slovenia • South Africa, Republic • Spain • Sweden • Switzerland • Taiwan • Thailand • Turkey • United Arab Emirates • United Kingdom • United States • Uruguay • Venezuela • Yugoslavia

Allen-Bradley Headquarters, 1201 South Second Street, Milwaukee, WI 53204 USA, Tel: (1) 414 382-2000 Fax: (1) 414 382-4444

1746-6.10 April 1998
Superseded 1746-6.10 March 1997
PN 955129-92