COOLAUTOMATION

Driver Suite

Crestron SIMPL Integration Guide

INCLUDED MODULES

- CoolAutomation Manager v1
- CoolAutomation Zone v1
- CoolAutomation Sensor v1
- CoolAutomation HOA Zone v1
- CoolPlug Helper v1

COMPATIBLE DEVICES

CoolMaster · CoolMasterNet* · CooLinkHub · CooLinkBridge · CoolMaster Edge
GS-SMT-HOA · GS-SMT-IO · SMT-131 · CoolPlug

Revision 1.1.2 · 2026-10-24 14:35 EDT

TCP/IP and RS-232 transport supported

Overview

Five release modules. Use one Manager per CoolAutomation device, one Zone per HVAC indoor unit, one Sensor per SMT-IO point, one HOA Zone per SMT-HOA hydronic zone, and the optional CoolPlug Helper only when commissioning CoolPlug settings. The Manager owns PRM queueing, parsing, polling, and child-module buses; the external TCP/IP Client or RS-232 symbol owns the wire.

Supports TCP/IP and RS-232 transport. Wire any Crestron TCP/IP Client or COM symbol to the Manager transport pins. Zone, Sensor, and HOA Zone modules filter their own UID from Manager broadcasts.

What 1.1.2 Fixes

- **Mode buttons:** With `Send_ON_With_Mode` high, one press turns an OFF unit on and selects the mode. The Zone checks the result and retries once if needed.
- **OFF button:** OFF takes priority over commands still waiting for the same Zone, so a queued ON cannot turn it back on. A command already sent to the device finishes first.
- **Debugging:** Normal Zone traffic no longer fills the trace window. Manager `Debug_FB` shows when Debug is on.

What 1.1.1 Fixed

- **Lost communication:** A device could stop answering commands even though TCP still showed connected. After five unanswered commands, Manager reconnects and starts communication again while `Connect` stays high.
- **Queue overflow:** Waiting commands could fill the Manager queue after replies stopped. The queue now has a size limit.
- **Mode and power:** A mode button configured to turn on a Zone now sends ON and the selected mode in order.
- **Setpoint display:** Old replies no longer make a newly requested temperature jump back on screen. If the command fails or times out, the display returns to the latest actual value.
- **Sensor updates:** Manager separates messages so the Sensor module can read each one reliably.

Install the five matched 1.1.2 modules together. Do not mix these Manager and child modules with older builds. For support, visit `https://support.coolautomation.com/hc/en-us/requests/new`.

Compatible Devices

CoolMaster · CoolMasterNet* · CooLinkHub · CooLinkBridge · CoolMaster Edge

PRM ASCII protocol · TCP port 10102 · RS-232 default 9600 8N1 · up to 16 simultaneous TCP connections

* CoolMasterNet is discontinued; CoolMaster is the current model. Both are supported -- protocol is backwards compatible.

NOTE

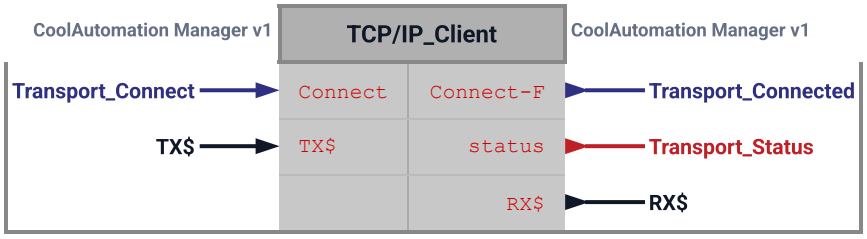
CoolMaster Edge first-time setup: run `set aserver enable 1` once from a terminal/service session before using TCP port 10102. The Manager cannot send this automatically because TCP is unavailable until the ASCII server is enabled.

Quick Start

1. Add Modules

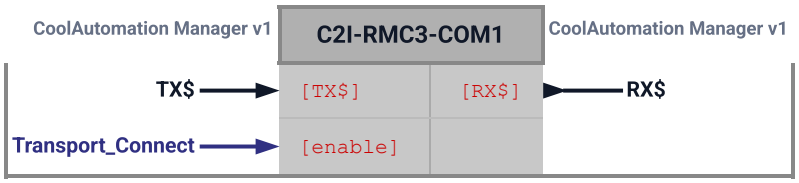
- Add **CoolAutomation Manager v1** -- one instance per gateway
- Add **CoolAutomation Zone v1** -- one instance per indoor unit
- Add **CoolAutomation Sensor v1** -- one instance per SMT-IO sensor point (optional)

2. Wire Transport (TCP/IP)



Set `Transport_Mode` parameter = `0` (TCP/IP mode). Set the TCP/IP Client IP address to the device IP and port `10102`.

2. Wire Transport (RS-232)



Set `Transport_Mode` parameter = `1` (RS-232 mode). Configure COM port to 9600 8N1 (default).

3. Wire the Bus

- Manager `To_Zone_Modules_Tx$` → every Zone `From_Manager$`
- Every Zone `To_Manager$` → Manager `From_Zone_Modules$`
- Manager `To_Sens_Tx$` → every Sensor `From_Manager$`

4. Set IDs and Parameters

- Each Zone: drive `Unit_UID$` with the PRM address (e.g., `L1.001`)
- For thermostat-style mode buttons, hold each Zone `Send_ON_With_Mode` input high; then a single mode press sends ON before the selected mode.
- Each HOA Zone: drive `Unit_UID$` with the HOA PRM address (e.g., `L2.011`)
- Each Sensor: drive `Sensor_UID$` with the PRM sensor address (e.g., `L3.001.001`)
- Manager: set `Poll_Interval_Sec` (default 45) and `Sens_Poll_Interval_Sec` (default 45)

NOTE

The 1.1.2 package contains five matched SIMPL+ module pairs and current help. It does not contain a newly compiled integrated demo program; compile and load your own SIMPL Windows program for the target system.

Architecture

Release Module Design

Manager broadcasts; Zone, Sensor, and HOA Zone modules subscribe. The Manager polls the device and forwards every state line onto the bus. Each child module filters by its own UID - no per-zone arrays, no shared unit state in the Manager. Modules are delivered as SIMPL+ user modules.



Bus Message Format

Manager-to-child records use LF (0x0A) framing and retained BUFFER_INPUTs. Child commands use the existing serial request interface.

Message	Direction	Description
`STATE:<UID>:<onoff>:<setT>:<roomT>:<fan>:<mode>:...`	Manager → Zone	Zone state update from ls/ls2 poll. Temperatures as tenths-integer (830 = 83.0°F).
`MANAGER:READY`	Manager -> helpers	Broadcast after startup set/line/ifconfig completes. CoolPlug Helper uses this to schedule delayed automatic readback.
`CMD:<UID>:<verb>[:<arg>]`	Zone → Manager	PRM request. modeon holds ON, mode and refresh in one ordered queue batch.
`SENSOR:<UID>:<type>:<value>`	Manager → Sensor	Sensor reading from sens poll.
`ZONEOK:<UID>:<verb>`	Manager → Zone	Command completed successfully for this UID.
`ZONEERR:<UID>:<verb>:<code>:<text>`	Manager → Zone	Command failed for this UID; code is PRM exit code.
`PASSTHRU:<raw>`	Zone → Manager	Raw PRM command sent through the Manager command queue.

NOTE Temperature values use tenths-integers: 830 = 83.0°F; 215 = 21.5°C.

CoolAutomation Manager v1

One instance per physical CoolAutomation device. Handles PRM handshake, command queueing, zone and sensor polling, and child-module bus broadcasts. The external TCP/IP Client or RS-232 symbol owns the physical transport; the Manager owns the protocol.

Parameters

Parameter	Type	Description
`Poll_Interval_Sec`	Time	HVAC poll interval in seconds. Default 45s; valid range 5s..3600s. In Fahrenheit mode Manager polls with ls for rounded CoolPlug/CooLinkHub setpoints; in Celsius mode it polls with ls2 for tenths precision.
`Sens_Poll_Interval_Sec`	Time	Sensor poll interval in seconds. Default 45s; valid range 5s..3600s.
`Transport_Mode`	Decimal list	0 = external TCP/IP Client. 1 = RS-232 direct serial transport. SIMPL parameter properties expose these choices as a list where supported.
`Inter_Command_Gap_Sec`	Time	Queue pacing between PRM commands. Values are seconds only, default to 2s, and clamp to 1s..10s. Lower values should be hardware-tested.

Digital Inputs

Signal Name	Type	Description
Connection Control		
`Transport_Connected`	DI	Assert when external TCP/IP Client reports connected. Manager begins init sequence on rising edge.
`Connect`	DI	Maintained high requests/maintains the transport session and protocol initialization. Low drops the request and disables reconnect/polling.
Polling & Refresh		
`Poll_Enable`	DI	High: enable HVAC polling at configured intervals. Low: suspend HVAC polling.
`Refresh_Names`	DI	Pulse to re-send props query and refresh unit name outputs on all zones.
`Sens_Enable`	DI	High: include sensor polling. ERROR:700 marks sensors unavailable/disabled and retries slowly while high.
`Reboot_Device`	DI	Pulse to send PRM reboot command. Device disconnects; Manager auto-reconnects.
Temperature Scale		
`Set_Scale_F`	DI	Pulse to configure device for Fahrenheit reporting. Stored in device NVRAM.
`Set_Scale_C`	DI	Pulse to configure device for Celsius reporting. Stored in device NVRAM.
`Swing_Enable`	DI	Required gate for Zone louver/swing commands. Hold high only on systems where louver control is desired/supported.
Diagnostics		
`Debug`	DI	High: enable console Trace messages plus Debug_TX$ and Debug_RX$ outputs. Increases string bus traffic significantly.

Analog Inputs

Signal Name	Type	Description
Transport		
`Transport_Status`	AI	TCP/IP Client status analog. Status 2 means connected. Ignored in RS-232 mode.
`Verbose_Errors`	AI	0 = suppress repeated identical errors. 1 = always output error text.

String Inputs

Signal Name	Type	Description
Bus Inputs		
`From_Zone_Modules$`	SI	CMD and PASSTHRU messages from all Zone module instances.
`RX$`	SI	Raw TCP receive data from external transport symbol (BUFFER_INPUT, 24 576 bytes).
`User_CMD$`	SI	Raw PRM ASCII command -- enqueued directly. For diagnostics and commissioning.

Digital Outputs

Signal Name	Type	Description
Connection & Status		
`Transport_Connect`	DO	Drive high to maintain TCP connection (wire to TCP/IP Client Connect input).
`Online_FB`	DO	High when transport is connected and the PRM session is active.
`Device_Info_Valid_FB`	DO	High after ifconfig response fully received and device strings populated.
`Polling_FB`	DO	High while an HVAC poll cycle is active.
`Temp_Scale_Is_F_FB`	DO	High when device is configured for Fahrenheit (from set response).
`Debug_FB`	DO	Mirrors the Manager Debug input; confirms whether routine Manager diagnostics are enabled.
Sensor State		
`Sens_Available_FB`	DO	High when at least one sensor reading has been received.
`Sens_Disabled_FB`	DO	High when sens polling was auto-disabled after ERROR:700 (no sensors on device).

Analog Outputs

Signal Name	Type	Description
Diagnostics & Counts		
`Last_Exit_Code`	AO	Most recent PRM exit code (0 = OK; non-zero = error).
`Queue_Depth_FB`	AO	Number of commands currently waiting in the send queue.
`Zone_Count_FB`	AO	Count of zone UIDs seen in the last completed HVAC poll cycle.
`Sensor_Count_FB`	AO	Count of sensor readings received in the last sens poll.

String Outputs

Signal Name	Type	Description
Device Information		
`Build_Info$`	SO	Module build timestamp string (e.g., "Build: 2026-05-12 10:30").
`Device_SN$`	SO	Device serial number from set response.
`Device_IP$`	SO	Device IP address from ifconfig response.
`MAC_Address$`	SO	Device MAC address from ifconfig response.
`Firmware_Version$`	SO	Firmware version string from set response.
`Active_Lines$`	SO	Comma-separated list of active bus lines (from line command).
Diagnostics		
`Last_Error$`	SO	Most recent error text string.
`Debug_TX$`	SO	Raw bytes sent to device (only when Debug is high).
`Debug_RX$`	SO	Raw bytes received from device (only when Debug is high).
Bus Outputs		
`To_Zone_Modules_Tx$`	SO	STATE:, ZONEOK, ZONEERR, MANAGER:READY, and helper/config messages broadcast to child modules.
`To_Sens_Tx$`	SO	SENSOR: messages broadcast to all Sensor instances.
`TX$`	SO	Raw PRM ASCII to send (wire to TCP/IP Client Send$ input).

CoolAutomation Zone v1

One instance per indoor unit. Drive the `Unit_UID$` string input with the PRM address (e.g., `L1.001`) and wire it to the Manager bus. The module handles all state tracking and command translation -- no other configuration required.

Parameters

Parameter	Type	Description
`Temp_Step_Tenths`	Integer	Setpoint step size as tenths-integer. Default 10 (= 1.0°). Use 5 for 0.5° steps.

Digital Inputs

Signal Name	Type	Description
Power		
`Power_On`	DI	Pulse to turn unit on.
`Power_Off`	DI	Pulse to turn unit off.
Mode Selection		
`Mode_Cool`	DI	Pulse to set operating mode to Cool.
`Mode_Heat`	DI	Pulse to set operating mode to Heat.
`Mode_Fan`	DI	Pulse to set operating mode to Fan only.
`Mode_Dry`	DI	Pulse to set operating mode to Dry (dehumidify).
`Mode_Auto`	DI	Pulse to set operating mode to Auto.
`Send_ON_With_Mode`	DI	Hold high for thermostat-style mode buttons. Zone sends ON, then the selected mode, and checks state. If needed it rechecks after ten seconds and retries once. An OFF press cancels the retry. Low sends mode only.
Fan Speed		
`Fan_Very_Low`	DI	Pulse to set fan speed to Very Low (if supported by unit).
`Fan_Low`	DI	Pulse to set fan speed to Low.
`Fan_Medium`	DI	Pulse to set fan speed to Medium.
`Fan_High`	DI	Pulse to set fan speed to High.
`Fan_Top`	DI	Pulse to set fan speed to Top.
`Fan_Auto`	DI	Pulse to set fan speed to Auto.
Louver		
`Louver_Auto`	DI	Set louver to Auto position.
`Louver_Horizontal`	DI	Set louver to Horizontal (fully open).
`Louver_Vertical`	DI	Set louver to Vertical (fully closed / downward).
`Louver_30_Deg`	DI	Set louver to 30° position.
`Louver_45_Deg`	DI	Set louver to 45° position.

Signal Name	Type	Description
`Louver_60_Deg`	DI	Set louver to 60° position.
`Louver_Stop`	DI	Stop louver at current position.
Temperature & Maintenance		
`Set_Temperature_Up`	DI	Increment setpoint by Temp_Step_Tenths.
`Set_Temperature_Dn`	DI	Decrement setpoint by Temp_Step_Tenths.
`Reset_Filter_Sign`	DI	Clear the filter maintenance reminder on this unit.

Analog Inputs

Signal Name	Type	Description
Direct-Set Controls		
`Mode`	AI	Set operating mode: 1=Cool, 2=Heat, 3=Fan, 4=Dry, 5=Auto.
`Fan_Speed`	AI	Set fan speed: 1=VLow, 2=Low, 3=Med, 4=High, 5=Top, 6=Auto.
`Set_Temperature`	AI	Set temperature setpoint as tenths-integer (e.g., 720 = 72.0°F).
`Feed_Temperature`	AI	External temperature override (from a room sensor) as tenths-integer.
`Louver`	AI	Set louver position: 0=None, 1=Auto, 2=Horizontal, 3=Vertical, 4=30°, 5=45°, 6=60°, 7=Stop.

String Inputs

Signal Name	Type	Description
Bus		
`Unit_UID$`	SI	PRM unit identifier, e.g., L1.001. Must match exactly.
`From_Manager$`	SI	Retained LF-framed STATE:, ZONEOK, and ZONEERR messages from Manager To_Zone_Modules_Tx$.

Digital Outputs

Signal Name	Type	Description
Power & Mode		
`Power_On_Fb`	DO	High when unit is on.
`Power_Off_Fb`	DO	High when unit is off.
`Mode_Cool_Fb`	DO	High when mode is Cool.
`Mode_Heat_Fb`	DO	High when mode is Heat.
`Mode_Fan_Fb`	DO	High when mode is Fan.
`Mode_Dry_Fb`	DO	High when mode is Dry.
`Mode_Auto_Fb`	DO	High when mode is Auto.
Fan Speed		
`Fan_Very_Low_Fb`	DO	High when fan speed is Very Low.

Signal Name	Type	Description
`Fan_Low_Fb`	DO	High when fan speed is Low.
`Fan_Medium_Fb`	DO	High when fan speed is Medium.
`Fan_High_Fb`	DO	High when fan speed is High.
`Fan_Top_Fb`	DO	High when fan speed is Top.
`Fan_Auto_Fb`	DO	High when fan speed is Auto.
Louver		
`Louver_Auto_Fb`	DO	High when louver is at Auto position.
`Louver_Horizontal_Fb`	DO	High when louver is at Horizontal.
`Louver_Vertical_Fb`	DO	High when louver is at Vertical.
`Louver_30_Deg_Fb`	DO	High when louver is at 30°.
`Louver_45_Deg_Fb`	DO	High when louver is at 45°.
`Louver_60_Deg_Fb`	DO	High when louver is at 60°.
`Louver_Stop_Fb`	DO	High when louver is stopped.
Status & Alerts		
`Filter_Sign_Fb`	DO	High when unit is signaling a filter maintenance reminder.
`In_Error_Fb`	DO	High when unit is in a fault or lockout state (PRM fail field non-OK).
`In_Demand_Fb`	DO	High when unit is actively heating or cooling.
`Command_Error_Fb`	DO	High when the most recent command returned an error.
`Feed_Error_Fb`	DO	High when the Manager returned a feed/zone-scoped error.

Analog Outputs

Signal Name	Type	Description
State Values		
`Mode_Fb`	AO	Current operating mode: 1=Cool, 2=Heat, 3=Fan, 4=Dry, 5=Auto.
`Fan_Speed_Fb`	AO	Current fan speed: 1=VLow, 2=Low, 3=Med, 4=High, 5=Top, 6=Auto.
`Louver_Fb`	AO	Current louver position: 0=None, 1=Auto, 2=Horizontal, 3=Vertical, 4=30°, 5=45°, 6=60°, 7=Stop.
`Set_Temperature_Fb`	AO	Current temperature setpoint as tenths-integer. The newest commanded value is held while stale replies are ignored; matching STATE confirms it. Command error or a 30-second timeout restores the latest raw value; temperature acknowledgements renew the window.
`Room_Temperature_Fb`	AO	Current room temperature as tenths-integer.
`Command_Error_Code`	AO	PRM exit code of the most recent failed command (0 = no error).
`Feed_Error_Code`	AO	Feed/zone-scoped error code from Manager.

String Outputs

Signal Name	Type	Description
Status & Diagnostics		
`Status_Fb$`	SO	Status string from Manager state. OK when healthy; CP01-CP05 CoolPlug faults are expanded into readable text.
`Unit_Name_Fb$`	SO	Unit name from props query (triggered by Refresh_Names on Manager).
`Command_Error_Text$`	SO	Text description of the most recent command error.
`Feed_Error_Text$`	SO	Text description of the most recent feed/zone error.
`To_Manager$`	SO	CMD: messages sent to Manager From_Zone_Modules$.

CoolAutomation Sensor v1

One instance per SMT-IO sensor point. Set `Sensor_UID` to the PRM sensor address (e.g., `L3.001.001`) and wire to Manager `To_Sens_Tx$`. Supports temperature, analog I/O, and digital I/O sensor types.

> **NOTE**
> Sensor polling is auto-disabled if the device returns ERROR:700 (no sensors attached). The Sens_Disabled_FB signal on the Manager will go high. Re-enabling requires pulsing Sens_Enable on the Manager after physical sensors are connected.

Sensor Types

Code	Type	Category	Feedback
1	**T**	*Temperature sensor*	Reports temperature as tenths-integer via Temperature_Fb.
2	**AIN**	*Analog input 0-10V*	Reports voltage via Analog_Voltage_Fb.
3	**IN**	*Digital input*	Contact closure -- Input_Open_Fb / Input_Closed_Fb.
4	**OUT**	*Digital output*	Relay -- Output_On_Fb / Output_Off_Fb.
5	**AOUT**	*Analog output 0-10V*	Reports current level via Output_Level_Fb. Writeable via Debug_Send_Level.

Digital Inputs

Signal Name	Type	Description
Debug Output Control		
`Debug_Output_On`	DI	Pulse to enable sensor debug writeback to device.
`Debug_Output_Off`	DI	Pulse to disable sensor debug writeback.
`Debug_Send_Level`	DI	Pulse to write the Debug_Output_Level value to the device output.

Analog Inputs

Signal Name	Type	Description
Configuration		
`Sensor_Type`	AI	Type filter: 1=T, 2=AIN, 3=IN, 4=OUT, 5=AOUT. Module ignores SENSOR: messages not matching this type.
`Debug_Output_Level`	AI	Output level to write on Debug_Send_Level pulse (analog output types only).

String Inputs

Signal Name	Type	Description
Bus		
`Sensor_UID$`	SI	Sensor identifier matching PRM sens output, e.g., L3.001.001.
`From_Manager$`	SI	Retained LF-framed SENSOR: messages from Manager To_Sens_Tx$.

Digital Outputs

Signal Name	Type	Description
Presence & Type		
`Seen_Fb`	DO	High after the first SENSOR: message matching this UID has been received.
`Type_T_Fb`	DO	High when sensor type is Temperature.
`Type_Ain_Fb`	DO	High when sensor type is Analog Input.
`Type_In_Fb`	DO	High when sensor type is Digital Input.
`Type_Out_Fb`	DO	High when sensor type is Digital Output.
`Type_Aout_Fb`	DO	High when sensor type is Analog Output.
Digital I/O State		
`Input_Open_Fb`	DO	High when digital input is open (type IN only).
`Input_Closed_Fb`	DO	High when digital input is closed (type IN only).
`Output_On_Fb`	DO	High when digital output is energized (type OUT only).
`Output_Off_Fb`	DO	High when digital output is de-energized (type OUT only).

Analog Outputs

Signal Name	Type	Description
Values		
`Type_Fb`	AO	Sensor type code: 1=T, 2=AIN, 3=IN, 4=OUT, 5=AOUT.
`Temperature_Fb`	AO	Temperature reading as tenths-integer (type T only).
`Analog_Voltage_Fb`	AO	Analog reading scaled (type AIN or AOUT).
`Output_Level_Fb`	AO	Current output level (type AOUT only).

String Outputs

Signal Name	Type	Description
Diagnostics		
`Type_Text_Fb$`	SO	Sensor type as text: "T", "AIN", "IN", "OUT", or "AOUT".
`Value_Raw_Fb$`	SO	Raw value string as received from device (for diagnostics).
`To_Manager$`	SO	Reserved for future Sensor→Manager messaging.

CoolAutomation HOA Zone v1

One instance per underfloor heating zone on a CoolAutomation SMT-HOA board. The SMT-HOA is an 8-zone hydronic/underfloor heating controller that communicates via Modbus RTU over the CooLinkBridge. HOA zones appear as standard PRM units on the bus (e.g., `L2.011`), so the same Manager module handles them alongside regular HVAC zones -- no separate gateway is required.

In addition to standard on/off and setpoint control (identical to Zone v1), the HOA Zone module writes thermostat configuration directly to the SMT-HOA board registers via routed HOACFG Modbus commands. Set all config inputs first, then hold `Write_Config` high for three seconds to commit the configuration. A short pulse does nothing; invalid values block writes.

NOTE

HOA zones have no fan speed, operating mode, or louver -- these fields are ignored in the PRM response. `In_Demand_Fb` reflects the relay output state (calling for heat). `In_Error_Fb` reflects the PRM status field.

UID Format

HOA zone UIDs use the standard PRM two-part format: `L2.011` = Line L2, Board 1, Zone 1. The three-digit suffix encodes both the Modbus slave ID (tens digit) and zone position within the board (units digit):

- `L2.011` → Line L2, Board 1, Zone 1 (Modbus slave 1, register base 30)
- `L2.015` → Line L2, Board 1, Zone 5 (Modbus slave 1, register base 70)
- `L2.021` → Line L2, Board 2, Zone 1 (Modbus slave 2, register base 30)
- `L2.028` → Line L2, Board 2, Zone 8 (Modbus slave 2, register base 100)

The module derives the Modbus slave ID and zone position automatically from the UID -- no separate parameter is needed. `Modbus_Slave_Fb` and `Zone_Position_Fb` confirm the decoded values after the UID is set.

Parameters

Parameter	Type	Description
`Temp_Step_Tenths`	Integer	Setpoint step for Up/Dn buttons, tenths-integer. Default 10 = 1.0°.

Digital Inputs

Signal Name	Type	Description
Power		
`Power_On`	DI	Pulse to turn zone on.
`Power_Off`	DI	Pulse to turn zone off.
Temperature		
`Set_Temperature_Up`	DI	Increment setpoint by Temp_Step_Tenths.
`Set_Temperature_Dn`	DI	Decrement setpoint by Temp_Step_Tenths.
Board Configuration		
`Write_Config`	DI	Hold high for three seconds to write validated config parameters via routed HOACFG register requests. A tap does nothing. Set all Config_* inputs first.

Signal Name	Type	Description
Set_Config_Scale_To_F	DI	Maintained level. High = Config_* write inputs are in ?F; Low = ?C (default). Module converts to ?C before Modbus write. Readback feedback is always in ?C because it comes from board registers.

Analog Inputs

Signal Name	Type	Description
Setpoint Control		
Set_Temperature	AI	Direct setpoint as tenths-integer. Same scale as the runtime device setting.
Temp_Step_Tenths	AI	Runtime override for setpoint step size. 0 uses the parameter default (10).
Board Configuration –– written after three–second Write_Config hold		
Config_Offset_Pos	AI	Positive sensor calibration offset -- tenths of selected scale (°C: 0-150 / °F: 0-270).
Config_Offset_Neg	AI	Negative sensor calibration offset magnitude -- tenths of selected scale. Net = Pos - Neg; drive one at a time.
Config_Diff	AI	Thermostat hysteresis (dead band) -- tenths of selected scale (°C: 5-500 / °F: 9-900). Default 5 = 0.5°C.
Config_Cutoff	AI	Floor sensor safety shutoff limit -- whole degrees (°C: 10-55 / °F: 50-131). Default 50°C / 122°F.
Config_Max_Cycles	AI	Max relay on/off cycles per hour (0-10). No scale conversion. Default 6.

String Inputs

Signal Name	Type	Description
Bus		
Unit_UID$	SI	HOA zone UID, e.g., L2.011. Must match the CooLinkBridge PRM address exactly.
From_Manager$	SI	Retained LF-framed STATE:, ZONEOK, and ZONEERR messages from Manager To_Zone_Modules_Tx$.

Digital Outputs

Signal Name	Type	Description
Power & Status		
Power_On_Fb	DO	High when zone is on.
Power_Off_Fb	DO	High when zone is off.
In_Demand_Fb	DO	High when the relay output is active (board is calling for heat).
In_Error_Fb	DO	High when PRM status field is non-OK (board fault or lockout).
Command_Error_Fb	DO	High when the most recent PRM command returned an error.
Config Write Status		
Config_Sent_Fb	DO	Latches high after the first successful Write_Config hold.
Config_Scale_Is_F_Fb	DO	High when Set_Config_Scale_To_F is asserted (config inputs interpreted as °F).

Signal Name	Type	Description
`Config_Scale_Is_C_Fb`	DO	High when config inputs are in °C (default). Inverse of Config_Scale_Is_F_Fb.

Analog Outputs

Signal Name	Type	Description
Runtime State		
`Room_Temperature_Fb`	AO	Current room temperature as tenths-integer (same scale as device).
`Set_Temperature_Fb`	AO	Current setpoint as tenths-integer. The newest commanded value is held while stale replies are ignored; matching STATE confirms it. Command error or a 30-second timeout restores the latest raw value; temperature acknowledgements renew the window.
`Command_Error_Code`	AO	PRM exit code of most recent failed command. 0 = no error.
Config Register Readback – always C units		
`Config_Offset_Pos_Fb`	AO	Actual board readback, positive offset magnitude in tenths °C.
`Config_Offset_Neg_Fb`	AO	Actual board readback, negative offset magnitude in tenths °C.
`Config_Diff_Fb`	AO	Actual board readback, thermostat hysteresis in tenths °C.
`Config_Cutoff_Fb`	AO	Actual board readback, cutoff limit in whole °C.
`Config_Max_Cycles_Fb`	AO	Actual board readback, max cycles per hour.
Commissioning		
`Modbus_Slave_Fb`	AO	Modbus slave ID decoded from UID tens digit. e.g., L2.023 → 2.
`Zone_Position_Fb`	AO	Zone position within board decoded from UID units digit. e.g., L2.023 → 3.

String Outputs

Signal Name	Type	Description
Status		
`Status_Fb$`	SO	"OK" or "ERROR:" `reflecting the PRM status field from the last STATE: message.`
`Command_Error_Text$`	SO	Text description of the most recent command error.
`To_Manager$`	SO	CMD: and HOACFG register requests sent to Manager From_Zone_Modules$.

Config Parameter Ranges

All config inputs follow the scale selected by `Set_Config_Scale_To_F`. The module converts F entries to C before writing Modbus registers. Readback values always report the board's C register values, even when the write values were entered in F. Values below show both the decimal meaning and the raw Crestron integer to drive:

Input	Type	Range	Description
`Config_Offset_Pos`	AI	0-150 (°C) / 0-270 (°F)	Positive sensor calibration offset in tenths of the selected scale. e.g. 50 = +5.0°C or +5.0°F depending on scale.
`Config_Offset_Neg`	AI	0-150 (°C) / 0-270 (°F)	Negative sensor calibration offset magnitude. Drive *one* of Pos/Neg at a time; net = Pos - Neg.
`Config_Diff`	AI	5-500 (°C) / 9-900 (°F)	Thermostat hysteresis (dead band) in tenths. Relay turns ON at setpoint - diff, OFF at setpoint. Default 5 = 0.5°C.
`Config_Cutoff`	AI	10-55 (°C) / 50-131 (°F)	Safety shutoff limit in whole degrees. Floor sensor shuts relay if this limit is exceeded. Default 50°C / 122°F.
`Config_Max_Cycles`	AI	0-10	Maximum relay on/off cycles per hour. Independent of temperature -- a secondary valve-wear guard. Default 6.

Value	°C decimal	Crestron °C	°F decimal	Crestron °F
Config_Diff -- hysteresis	tenths of selected scale			
Min / default	0.5°C	`5`	~0.9°F	`9`
1 degree	1.0°C	`10`	1.8°F	`18`
2 degrees	2.0°C	`20`	3.6°F	`36`
5 degrees	5.0°C	`50`	9.0°F	`90`
Config_Offset -- calibration	tenths of selected scale			
+/-0.5°	0.5°C	`5`	~0.9°F	`9`
+/-1.0°	1.0°C	`10`	~1.8°F	`18`
+/-5.0°	5.0°C	`50`	~9.0°F	`90`
Maximum	15.0°C	`150`	~27.0°F	`270`
Config_Cutoff -- safety limit	whole degrees of selected scale			
Board minimum	10°C	`10`	50°F	`50`
Typical slab	40°C	`40`	104°F	`104`
Default	50°C	`50`	122°F	`122`
Board maximum	55°C	`55`	131°F	`131`
Config_Max_Cycles	count -- same value in both scales			
No limit	`0`			
Default	`6`			
Maximum	`10`			

How Hysteresis Works

The SMT-HOA relay output is bang-bang (fully open or closed). `Config_Diff` defines the dead band around the setpoint:

- Relay turns **ON** when room temperature falls below *setpoint - differential*
- Relay turns **OFF** when room temperature rises back to *setpoint*

Example -- setpoint 24.0°C, diff 0.5°C (value 5): valve opens at 23.5°C, closes at 24.0°C.

Underfloor floor slabs have large thermal mass and move slowly through temperature. The default 0.5°C hysteresis is appropriate. `Config_Max_Cycles` (default 6/hr) is a secondary hard limit on valve cycling regardless of temperature, protecting zone valves from wear. A wider differential of 1.0-2.0°C gives longer on/off cycles and slightly less precision, which suits slab systems well.

NOTE

Set and validate every configuration input before holding Write_Config. Invalid values block all writes and report local error 900. Config write inputs follow Set_Config_Scale_To_F; readback values always use board C units.

CoolPlug Helper v1

Optional commissioning helper for one CoolPlug on a CooLinkHub or CooLinkHub Wireless line. Normal HVAC control stays in the standard Zone module. The helper exists only to read and set temperature-source and master/slave settings without making the programmer type raw PRM commands.

NOTE

Write buttons are deliberately simple: hold **`Write_Enable`** high, then pulse the desired source or master/slave button. Releasing **`Write_Enable`** blocks all writes again. Read_CoolPlug does not require Write_Enable.

NOTE

After the Manager reports startup ready, the helper waits briefly and performs one automatic readback. This uses the same staggered sequence as the manual read: **`plug <UID> set`**, then **`plug <UID> line`** after the first command completes.

Parameters

Parameter	Type	Description
`Default_Plug_UID$`	String	Default CoolPlug UID, e.g. L2.001 or L5.001. Plug_UID$ can override this at runtime.

Digital Inputs

Signal Name	Type	Description
Readback		
`Read_CoolPlug`	DI	Pulse to send **`plug <UID> set`**, then **`plug <UID> line`**, then one **`plug <UID> wpan`**. Updates temperature-source, serial number, master/slave, online/offline, and wireless feedback.
Write Gate		
`Write_Enable`	DI	Maintained level. Must be high while any write pulse occurs.
Room Temperature Source		
`Room_Source_Default`	DI	Pulse while Write_Enable is high: **`set tsrc R D`**.
`Room_Source_Indoor`	DI	Pulse while Write_Enable is high: **`set tsrc R I`**.
`Room_Source_Feed`	DI	Pulse while Write_Enable is high: **`set tsrc R F`**.
`Room_Source_External`	DI	Pulse while Write_Enable is high: **`set tsrc R E`**.
Control Temperature Source		
`Control_Source_Default`	DI	Pulse while Write_Enable is high: **`set tsrc A D`**.
`Control_Source_Indoor`	DI	Pulse while Write_Enable is high: **`set tsrc A I`**.
`Control_Source_Feed`	DI	Pulse while Write_Enable is high: **`set tsrc A F`**.
`Control_Source_External`	DI	Pulse while Write_Enable is high: **`set tsrc A E`**.
Master/Slave		
`Master_Mode_Auto`	DI	Pulse while Write_Enable is high: automatic master/slave selection.

Signal Name	Type	Description
`Master_Mode_Slave`	DI	Pulse while Write_Enable is high: force slave.
`Master_Mode_Master`	DI	Pulse while Write_Enable is high: force master.
`Reboot_Device`	DI	Pulse while Write_Enable is high after Boot_Required_FB latches. Sends `plug <UID> boot 2`, e.g. `plug L5.001 boot 2`. This targets the selected CoolPlug, not the hub. The plug may not return positive confirmation; timeout is reported as sent/no confirmation and Status_Text$ shows `Reboot sent - no confirmation`.

Digital Outputs

Signal Name	Type	Description
Command State		
`UID_Valid_FB`	DO	High when the selected UID is exact Lx.xxx format.
`Write_Enable_FB`	DO	Mirrors the write gate.
`Command_Busy_FB`	DO	High while waiting for Manager/device response.
`Command_OK_FB`	DO	Short pulse when command returns OK.
`Command_Error_FB`	DO	High when local validation or device command fails.
`Validation_Blocked_FB`	DO	Short pulse when a write is blocked by bad UID, missing Write_Enable, or invalid reboot state.
`Settings_Valid_FB`	DO	High after the helper has accepted or parsed at least one source/master setting.
`Boot_Required_FB`	DO	Latches high when a response line contains Boot Required.
`Plug_Online_FB`	DO	High after an OK response for the selected plug.
`Plug_Offline_FB`	DO	High after a device/Manager error for the selected plug.
`Wireless_Status_Valid_FB`	DO	High after a wireless readback line has been received and parsed.
Room Source Feedback		
`Room_Source_Default_FB`	DO	Current room source is Default.
`Room_Source_Indoor_FB`	DO	Current room source is Indoor return-air sensor.
`Room_Source_Feed_FB`	DO	Current room source is Feed temperature.
`Room_Source_External_FB`	DO	Current room source is External CoolPlug sensor.
Control Source Feedback		
`Control_Source_Default_FB`	DO	Current control source is Default.
`Control_Source_Indoor_FB`	DO	Current control source is Indoor return-air sensor.
`Control_Source_Feed_FB`	DO	Current control source is Feed temperature.
`Control_Source_External_FB`	DO	Current control source is External CoolPlug sensor.
Master/Slave Feedback		
`Master_Mode_Auto_FB`	DO	Current mode read/accepted as Auto.
`Master_Mode_Slave_FB`	DO	Current mode read/accepted as Slave.

Signal Name	Type	Description
`Master_Mode_Master_FB`	DO	Current mode read/accepted as Master.

Analog and String Outputs

Signal Name	Type	Description
Diagnostics		
`Command_Status`	AO	0 idle, 1 busy, 2 OK, 3 validation blocked, 4 device error.
`Command_Error_Code`	AO	900 for local validation; otherwise Manager/PRM error code.
`Wireless_Signal_Level`	AO	First numeric value found in the wireless readback line, when present. Use Wireless_Signal_Text$ as the raw truth.
`Plug_SN$`	SO	CoolPlug serial number from the `S/N` line in settings readback.
`Status_Text$`	SO	Plain text command state.
`Last_Command$`	SO	Last raw PRM command emitted through Manager.
`Last_Response$`	SO	Raw helper response lines received from Manager.
`Wireless_Signal_Text$`	SO	Raw wireless readback line from `plug <UID> wpan`.
`Last_Error$`	SO	Most recent validation/device error text.
`To_Manager$`	SO	CPHELPER messages wired to Manager From_Zone_Modules$.

Temperature Source Choices

Choice	Meaning
Default	CoolPlug chooses by priority: external sensor, feed command, wall thermostat if available, then indoor return-air sensor.
Indoor	Use the indoor unit return-air sensor.
Feed	Use the temperature sent by a CoolAutomation feed command.
External	Use the sensor physically connected to the CoolPlug.

PRM Error Codes

The Manager's **Last_Exit_Code** analog output holds the most recent code. Zone modules surface command-specific errors via **Command_Error_Code** and **Command_Error_Text$**. Common codes encountered during commissioning and operation:

Code	Meaning	Notes
0	**OK**	Command executed successfully.
1	**No UID**	Unit address not found. Check the child module UID string input or the target UID in the command.
2	**Not Strict UID**	UID must be exact format (e.g., L1.001 not L1.1).
3	**Bad Format**	Command syntax error.
4	**Failed**	Command execution failed.
101	**Unsupported Feature**	This command is not supported by the indoor unit type.
102	**Incorrect Indoor Type**	Indoor unit doesn't support this command or parameter.
103	**No ACK From Indoor**	Indoor unit not responding -- check HVAC wiring.
104	**Time Out on Receive**	No response from indoor unit within timeout period.
107	**Line Error**	Error on HVAC communication line.
108	**Feed Disabled**	Indoor unit can't accept ambient feed -- check unit switch position.
200	**UID Not Found**	Unit not in device database. Verify UID and device connection.
700	**No Sensors**	No SMT-IO sensors attached. Sensor polling auto-disabled.
701	**Sensor Error**	Sensor I/O error.

See **Appendix A** for the full PRM error code reference (codes 0-310, 700-701).

Appendix A –– PRM Error Code Reference

Complete error code table from CoolMaster Product Line PRM Rev 0.10, Section 6.1.2. Codes are returned by the device as `ERROR:N` on the PRM bus.

Code	Verbose	Description
0	**OK**	Command executed successfully.
1	**No UID**	Unit not found.
2	**Not Strict UID**	UID must be precise.
3	**Bad Format**	Command format is wrong.
4	**Failed**	Command execution failed.
5	**Line Unused**	Line is unused.
6	**Unknown Command**	Command is unknown.
7	**Bad HVAC Line**	Line number is wrong.
8	**Bad Function**	Wrong function.
9	**Bad Line Type**	Wrong line type definition.
10	**Bad Parameter**	Command parameter is wrong.
11	**OK, Boot Required**	Command will take effect after reboot.
12	**Wrong GPIO**	Wrong GPIO.
13	**SDDP Disabled**	SDDP module is disabled; enable it to proceed.
14	**Virtual Address In Use**	Virtual address already in use.
15	**Bad Property**	Wrong property.
16	**Number of Lines Exceeded**	Can't define more line types.
17	**Dip Switch State Incorrect**	Dip switch state is incorrect for defined line type.
18	**SDDP Not Initialized**	SDDP is enabled but Ethernet link is still down.
80	**ModBus Error 80**	No response from the addressee.
81	**ModBus Error 81**	Big timeout between bytes in received message.
82	**ModBus Error 82**	Small timeout between bytes in received message.
83	**ModBus Error 83**	Received message with internal timeout.
84	**ModBus Error 84**	Received message is too big.
85	**ModBus Error 85**	CRC error in received message.
86	**ModBus Error 86**	ModBus exception in response.
100	**Collision**	Collision in sent command on HVAC line.
101	**Unsupported Feature**	Unsupported command or parameter for this HVAC line.
102	**Incorrect Indoor Type**	Indoor unit doesn't support this command or parameter.
103	**No ACK From Indoor**	Indoor unit didn't acknowledge sent command.

Code	Verbose	Description
104	**Time Out on Receive**	No response from indoor unit.
105	**CS Error In Received Message**	Check sum error in received message.
106	**Line Init In Progress**	Can't show line status during initialization.
107	**Line Error**	Error on the HVAC line.
108	**Feed Disabled**	Indoor unit can't receive ambient temperature due to switch position.
150	**HDL Not Initialized**	HDL line not defined or Ethernet cable unplugged.
151	**HDL DB Overflow**	HDL Data Base full; can't add new configuration.
152	**HDL Eth Disabled**	HDL over Ethernet disabled.
200	**UID Not Found**	Specified indoor unit not found in Data Base.
201	**Strict UID Not Found**	Specified indoor unit not found by strict UID.
202	**Indoor Removed**	Indoor unit removed from Data Base.
203	**DB Overflow**	Indoor Units Data Base full; can't add new one.
204	**Group DB Overflow**	Group Data Base full; can't add new group.
205	**VA DB Overflow**	Virtual address Data Base full; can't assign new virtual address.
206	**FDBS Overflow**	Properties Data Base full; can't add new property.
250	**Link DB Overflow**	Link Data Base full; can't link new CoolPlug device.
251	**No CooHub Line**	CooHub line not defined; define it to proceed.
252	**Auto Visibility Failed**	Error adding visibility props during link creation.
253	**Link Already Exists**	CoolPlug device already linked; delete existing link first.
307	**KNX DB Overflow**	KNX Data Base full; can't add new group.
309	**KNX Not Connected**	No connection with KNX chip.
310	**KNX Line Not Started**	KNX line not defined.
700	**No Sensors**	No SMT-IO sensors attached. Sensor polling auto-disabled after first occurrence.
701	**Sensor Error**	Sensor I/O error.

Appendix B – CoolPlug PBUS Status Codes

CoolPlug zones can report CP status codes in the PRM fail/status field. The Zone module expands these in `Status_Fb$` and raises `In_Error_Fb`.

Code	Description
`CP01`	CoolPlug has no connection to the HVAC unit.
`CP02`	No communication with the CoolPlug device.
`CP03`	CoolPlug device is in boot mode.
`CP04`	CoolPlug is not properly configured.
`CP05`	CoolPlug cannot detect the HVAC unit.

Revision History

Rev	Date	Notes
`1.1.2`	October 2026	Improves mode selection when a Zone is off, gives OFF priority over waiting commands, and reduces routine trace messages.
`1.1.1`	October 2026	Restores lost connections, protects the command queue, and prevents old replies from changing a new setpoint.
`1.0.5`	October 2026	Improves messages between modules and keeps a new setpoint visible until the device responds.
`1.0.4`	September 2026	Prevents Zone string errors when a device sends a short or incomplete message.
`1.0.3`	August 2026	Adds louver and filter reset controls.
`1.0.2`	August 2026	Shows Manager trace messages only when Debug is enabled.
`1.0.1`	August 2026	Reduces empty poll messages and corrects Zone ID guidance.
`1.0.0`	August 2026	First public driver suite for CoolAutomation PRM devices.
`0.9.2`	July 2026	Adds simple CoolPlug setup controls.
`0.9.1`	July 2026	Adds setup and status guidance for HOA Zone, CoolPlug, and Edge.
`0.9.0`	May 2026	Initial test build of Manager, Zone, and Sensor.