Honeywell Panel Bus Driver for Tridium Niagara

User Guide

baudrate.io

niagara4

February 22, 2026

Contents

1 Introduction **2**

2 Requirements **3**

3 Quick Start **3**

4 Panel Bus Network **4**

5 Panel Bus Device **5**

6 Panel Bus Points **6**

7 Panel Bus Point Config **8**

7.1 Digital Inputs 9
7.2 Analog Inputs 9
7.3 Digital and Analog Outputs 9
7.4 Floating Outputs 9

8 FAQ **10**

8.1 I cannot discover any IO modules 10
8.2 Output points do not change the physical output 10
8.3 Temperature readings are incorrect 11

1 Introduction

Honeywell[1] Panel Bus is a protocol for IO modules for Honeywell controllers, such as Excel 800, Excel Web II, ComfortPoint Open Plant, LION, EAGLE, and EAGLEHAWK. The family of these IO modules includes multiple pluggable modules with just one type of IO (analog input, binary input, analog output, relay output, and floating / three-position output) or a mixture of IO.

Model	AI	BI	AO	RO	FO	Override
XF821A / CLIOP821A	8					
XF822A / CLIOP822A			8			
XFR822A / CLIOPR822A			8			X
XF823A / CLIOP823A		12				
XF824A / CLIOP824A				6		
XF824A / CLIOPR824A				6		X
XFR825A / CLIOPR825A					3	X
XF830A / CLIOP830A	8	12	8	6		
CPO-IO830 / CLIOP831A	8	12	8	6		

Panel Bus modules communicate over an RS-485 interface using 115.2 kbps baud rate. They are hot-pluggable – no need to disconnect power to swap them. All addressing is done using one HEX switch. Output modules are available with manual override switches / potentiometers for easy maintenance. Panel Bus modules cover all kinds of applications that require from tens to hundreds of IO points.

For a full description of Panel Bus refer to Panel Bus I/O Modules.

Tridium Niagara driver enables retrofitting Honeywell panels by replacing the main controller with JACE or any other Niagara-compatible device (Cylon Integra, Johnson Controls FX, Distech Controls EC-BOS, GC5 iSMA-B-MAC36NL etc.) and keeping all IO modules. This greatly reduces the amount of work and cost of the new controller IO. Panel Bus IO points will become available for control by Niagara logic on a standard wiresheet. Each IO point can also be configured, e.g., to set thermistor types for inputs or fail-safe value for outputs.

[1] All trademarks or registered trademarks are the property of their respective owners

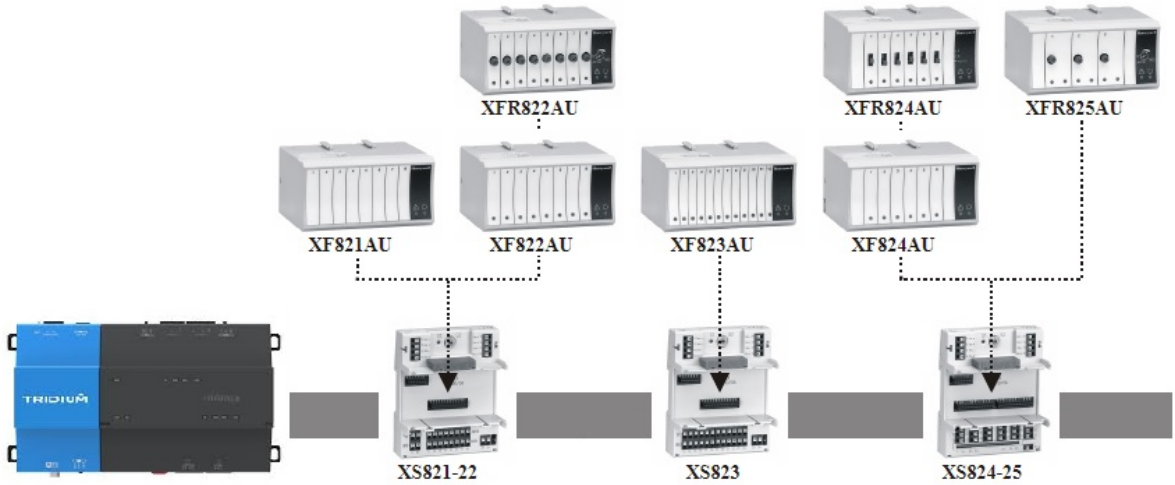

Figure 1: Panel Bus driver architecture

2 Requirements

- Niagara-powered device with software v4.8 or later, such as JACE or any OEM version
- JACE RS-485 port or serial-to-IP device with Baudrate Virtual Port driver
- Panel Bus driver module and license

3 Quick Start

1. Import panelBus-rt.jar module with all dependencies into both JACE and Workbench, restart *both*.
2. Add **Panel Bus Network** to the station.
3. Enter license code in **License** property under **Panel Bus Network**.
4. Set COM port under **Serial Config / Port Name**. Make sure port settings are correct (Panel Bus default settings are 115200/8/1/None).
5. **Discover** devices – you should be able to see IO modules connected to the COM port.
6. Add discovered devices to the station.
7. Open device **Points** extension and **Discover** points.
8. Add points to the station. Points should show their values and output signals can be changed.

4 Panel Bus Network

Panel Bus Network contains many standard Niagara properties, as well as a few driver-specific ones:

- **License** – the code that allows the driver to run on your host.
- **Tuning Policies / Default Policy / Read IO Fault Status** – whether the point status is read from the IO module and applied to the Niagara point. By default it is set to false.
- **Tuning Policies / Default Policy / Always Subscribed** – whether the point should always be subscribed: periodically read from the IO module.
- **Poll Scheduler** – specifies polling rates for periodic data reading:
  - Fast Rate: 5 seconds
  - Normal Rate: 10 seconds
  - Slow Rate: 30 seconds

Normally only **License** and **Serial Config / Port Name** need to be changed. It is possible to have multiple **Panel Bus Networks** in a Niagara station, each with its own COM port. Points in all networks are counted for licensing purposes.

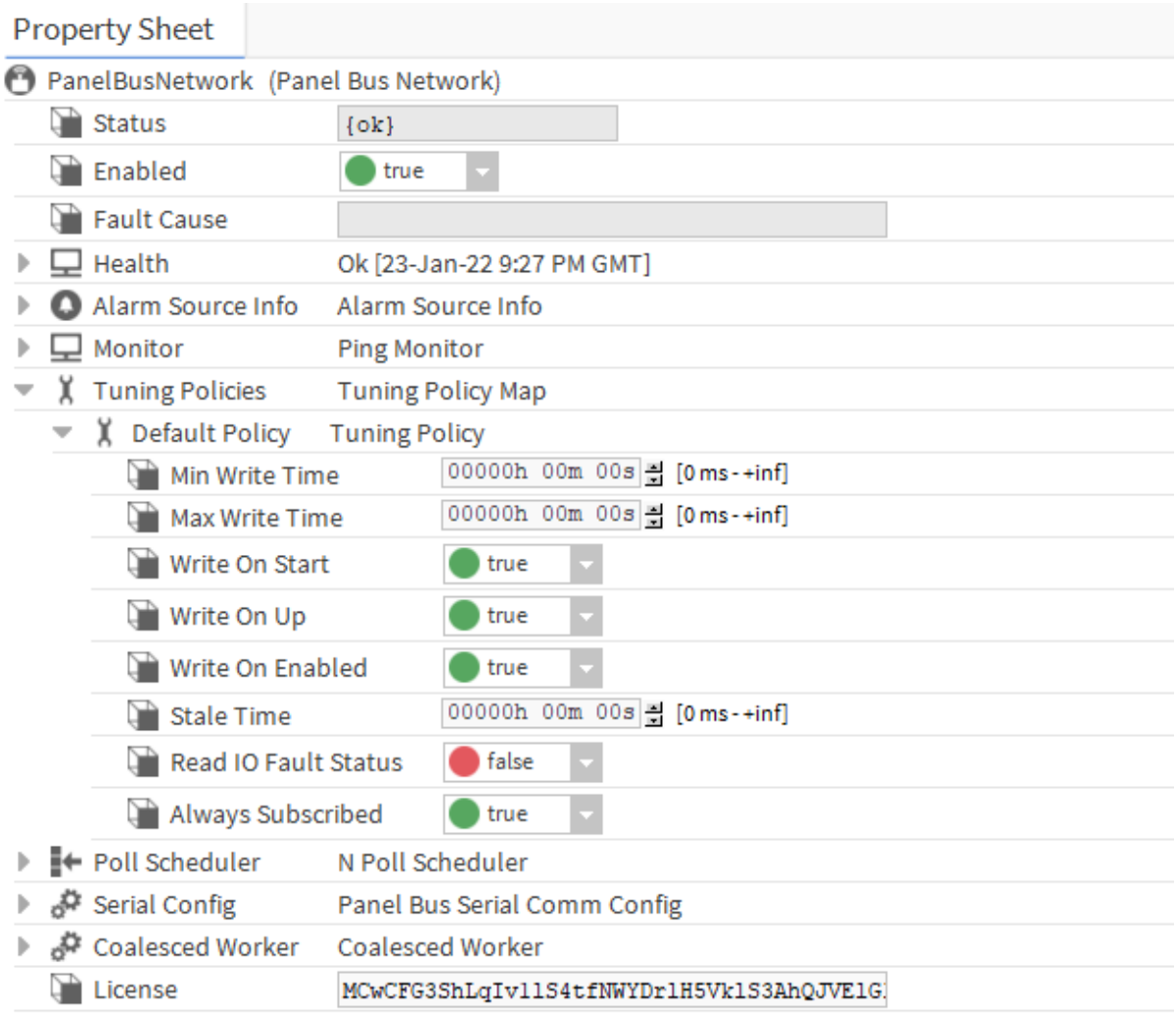

Figure 2: Panel Bus network properties

5 Panel Bus Device

Panel Bus IO modules are addressed using the *S2 HEX switch*, so each module has an address expressed as one hexadecimal digit from 0 to F. Each type of IO modules (AI, BI, AO, BO, mixed) is addressed separately. For example, it is possible to have one CLIOP821A module with address 0 and one CLIOP822A with address 0 on the same line.

Panel Bus Device contains the following driver-specific properties:

- **Model** – IO device model.

- **Address** – hexadecimal number set by the S2 switch on the IO module.
- **Serial** – device serial number, read only.
- **Firmware** – device firmware version, read only.
- **Poll Frequency** – frequency of point polling, either *Fast*, *Normal*, or *Slow*. Timings are specified in **Panel Bus Network / Poll Scheduler**.

Devices can be discovered automatically or added manually. In case of automatic discovery, no properties need to be changed.

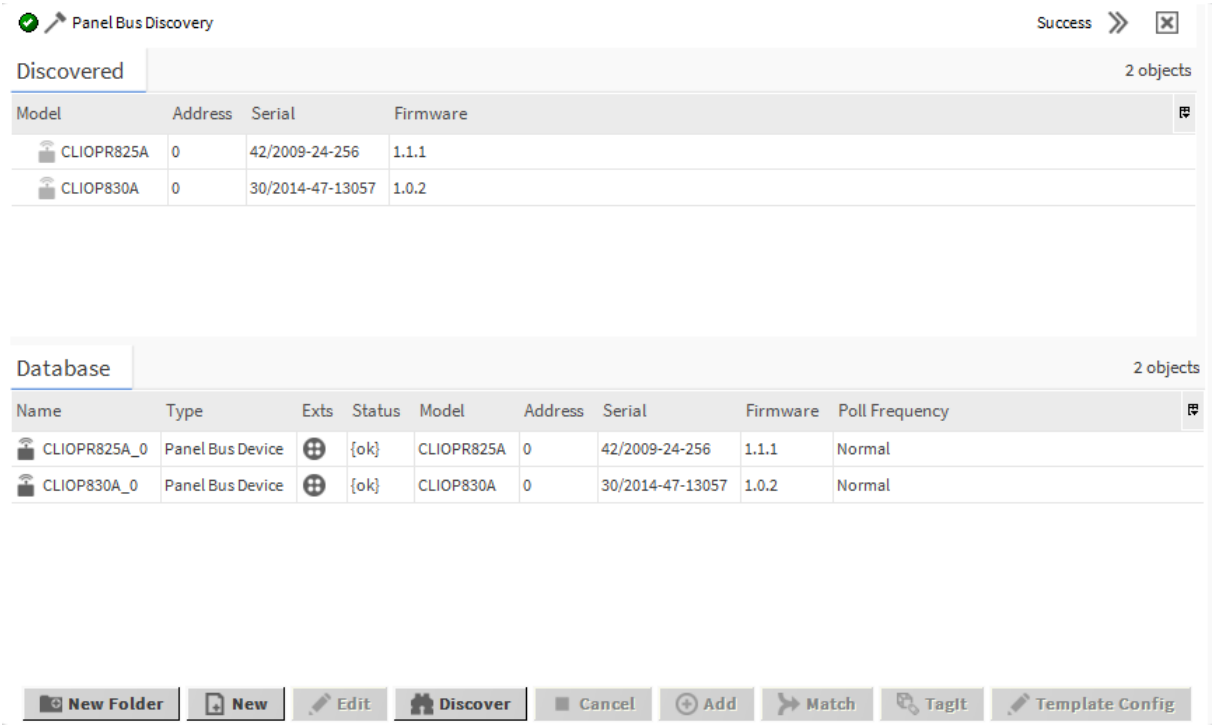

Figure 3: Panel Bus devices

6 Panel Bus Points

Panel Bus points are discovered automatically in **Panel Bus Point Manager**. Each point has the following properties:

- **Type** – Niagara point type, Boolean or Numeric Point for inputs and Boolean or Numeric Writable for outputs.
- **Facets** – point units.
- **Point Type** – Panel Bus IO type.
- **Address** – IO number.

- **IO Type** – Panel Bus IO configuration.

Point Type and **Address** are automatically assigned when points are added to the station from the discovery pane. **Type** is also automatically selected based on **Point Type**, but it can be changed if needed.

Analog Input is mapped into *Numeric Point* by default, but it can also be used as a digital input, then it is mapped into *Boolean Point*. **IO Type** specifies the analog input type: select thermistor curve for passive sensors or 0/2-10V for active ones.

Digital Input is mapped into *Boolean Point* by default, but it can also be used as a counter input, then it is mapped into *Numeric Point*.

Analog Output is mapped into *Numeric Writable* by default, but it can also be used as a digital output, then it is mapped into *Boolean Writable*. **IO Type** specifies the analog output type: 0-10V or 2-10V.

Relay Output is mapped into *Boolean Writable*.

Floating Output is mapped into *Numeric Writable*.

Point units are set automatically in **Facets** property. All temperatures are in degrees Celsius by default. For Fahrenheit degrees set **Facets** to Fahrenheit and **Proxy Ext / Device Facets** to Celsius – values will be recalculated automatically.

Proxy Ext / Tuning Policy Name assigns a *Tuning Policy* to each point. This policy determines when the point is written to, if the point is always subscribed, and if the open / short circuit of analog inputs is mapped to the point fault status.

All added points are polled by the driver in groups to minimize the traffic. Polling rate is specified in **Panel Bus Device / Poll Frequency** property.

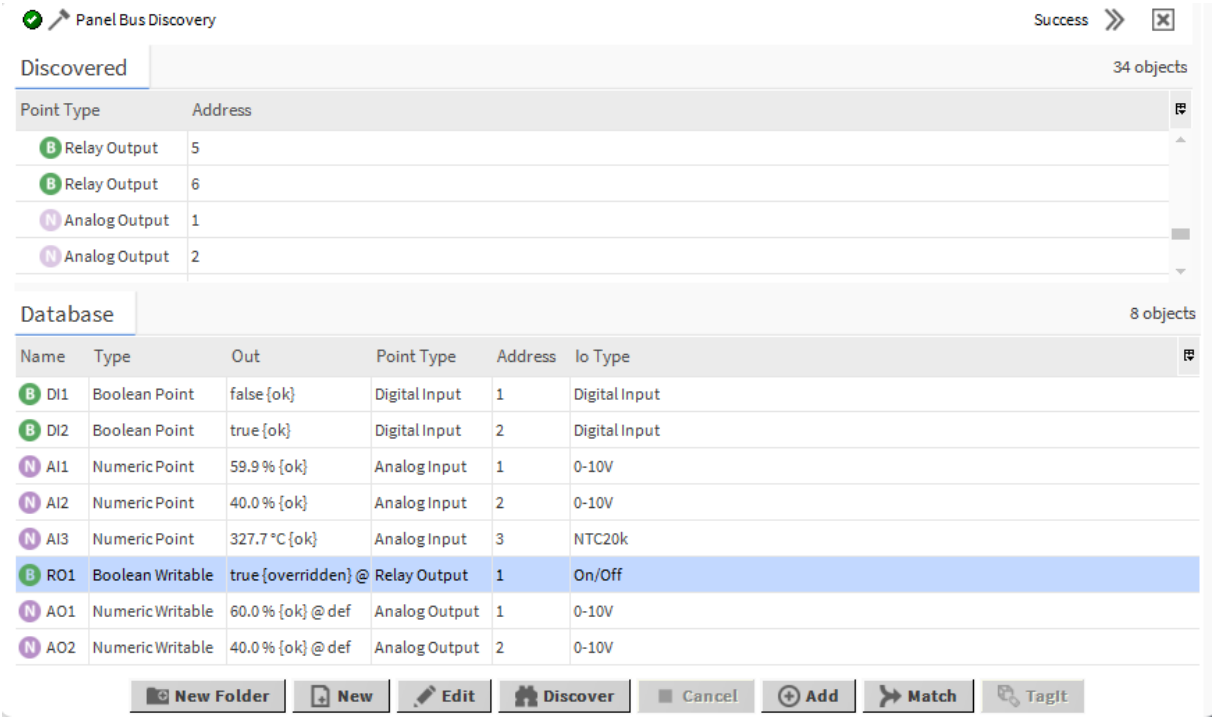

Figure 4: Panel Bus points

7 Panel Bus Point Config

Panel Bus points have extra configuration parameters that determine, for example, whether DI is normally open or closed and in which position RO remains in case of communication failure. In order to set these parameters, **PointConfig** component must be copied from panelBus palette under each point. **PointConfig** automatically displays parameters relevant to parent point type, see below. The driver checks if this component is available and writes its values into the Panel Bus module.

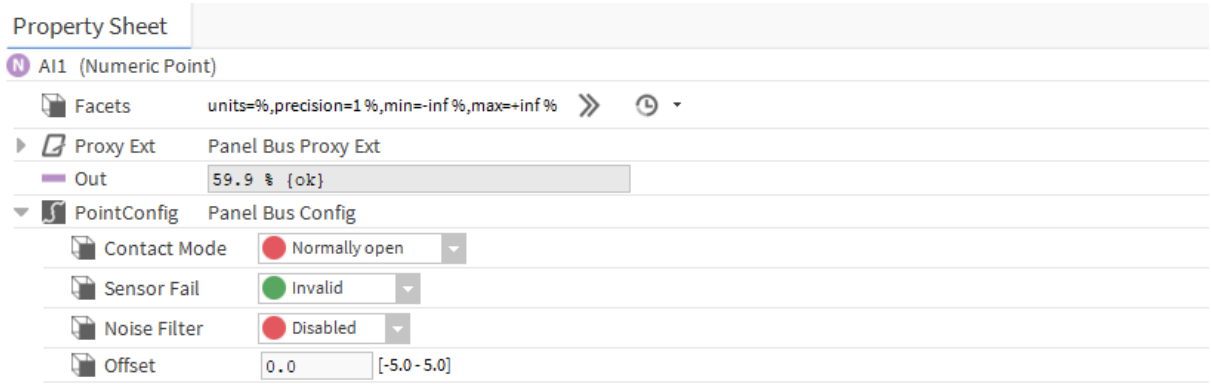

Figure 5: Panel Bus point config

7.1 Digital Inputs

- **Contact Mode** – *Normally closed* or *Normally open*.
- **LED Behavior** – IO LED color to be *Off / Yellow* or *Green / Red* (except for CLIOP830A / CLIOP831A modules).

7.2 Analog Inputs

- **Contact Mode** – [for digital input mode] *Normally closed* or *Normally open*.
- **Sensor Fail** – in case of sensor failure show XXX (*Invalid* option) or the last good value (*Last Position* option).
- **Noise Filter** – limits signal changes to filter out extra noise.
- **Offset** – an offset to correct the value. One can also use **ProxyExt / Conversion** property to add offset on Niagara side.

7.3 Digital and Analog Outputs

- **Direct Reverse** – reverses the output signal.
- **Safety Position** – output signal in case of communication fault: 0%, 50%, 100%, or keep the last value.

7.4 Floating Outputs

- **Safety Position** – output signal in case of communication fault.

- **End Switches** – if the motor has end switches, open and close relays will remain energized at the end positions.
- **Power Up Sync** – the motor will fully close on power-up for synchronization.
- **Sync 24h** – the motor will fully close every 24 hours for synchronization.
- **Sync Break** – if *No* option is selected, complete synchronization before moving to a new position.
- **Close Repeat** – after reaching 0%, the close relay will be energized for a short period to fully close its rubber seal.
- **Valve Exercising** – open valve once a week to assure reliable operation.
- **Open Runtime** – time from 0% to 100%.
- **Close Runtime** – time from 100% to 0%.
- **Sync Charge** – how long the relay is kept energized after reaching end position during synchronization.
- **Sync Level Open** – synchronize the motor to fully open position if its setpoint is higher than this setting; if the value is 255, disable this function.
- **Sync Level Close** – synchronize the motor to fully closed position if its setpoint is lower than this setting; if the value is 255, disable this function.
- **Min Stop Time** – minimal stop time before changing movement direction.
- **Min Run Percent** – minimal percentage change.

8 FAQ

8.1 I cannot discover any IO modules

- Verify the COM port name and settings (Panel Bus uses 115200 baud, 8 data bits, 1 stop bit, no parity).
- Check RS-485 wiring between the JACE and the IO modules.
- Ensure IO modules are powered and properly seated in the mounting rail.
- Check that the license is valid.

8.2 Output points do not change the physical output

- Verify the point type is writable (e.g., **Numeric Writable** or **Boolean Writable**).
- Check the **Safety Position** setting in **PointConfig** – if the module detects a communication fault, it may override the output.
- Ensure the manual override switch on the IO module is not engaged.

8.3 Temperature readings are incorrect

- Check the **IO Type** property on the analog input – ensure the correct thermistor curve is selected for the connected sensor.
- Verify the **Offset** value in **PointConfig** is correct.
- If converting between Celsius and Fahrenheit, set **Facets** to the desired unit and **Proxy Ext / Device Facets** to the source unit.