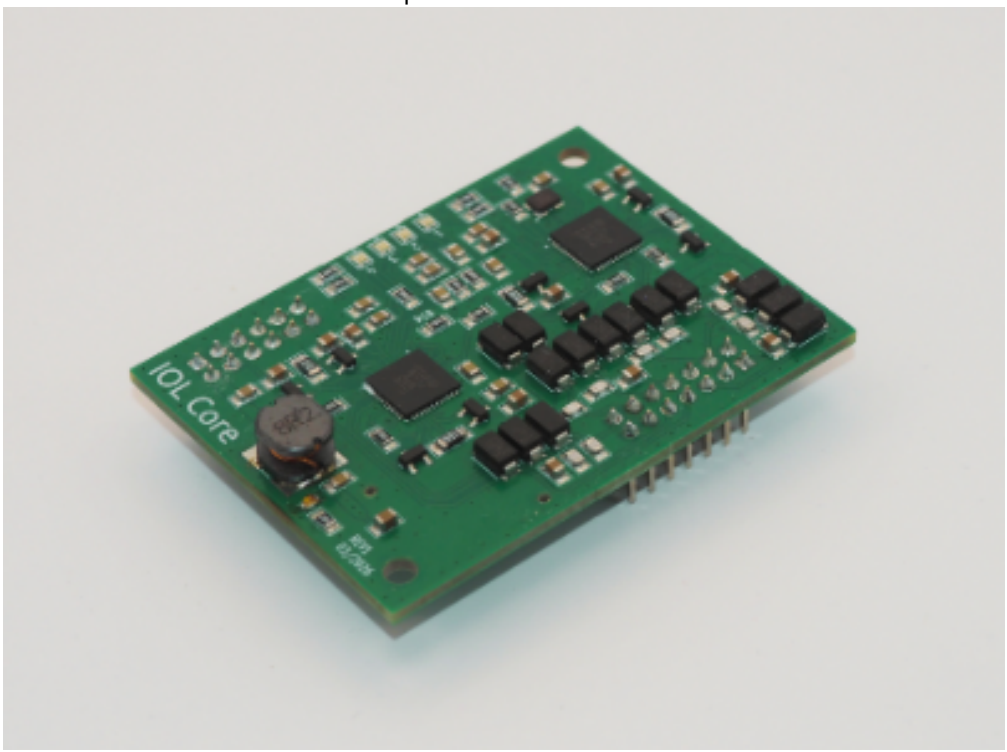
4 Counter / Trigger Engine

Hardware manual

www.pinetek-networks.com



Pinetek Networks GmbH, Bachstelzenweg 8, 88677 Markdorf
info@pinetek-networks.com

2026-11-22

Table of Contents

4 Counter / Trigger Engine 5
Counter Values and Labels 5
Counter Values 5
Counter Config 5
Set Counter Label 5
Counter Reset 6
Reset Counter 6
Reset All Counters 6
Trigger Configuration 6
Get Trigger Config 6
Set Trigger Config 7
Engine Control 7
Start 7
Stop 7
Status 7

4 Counter / Trigger Engine

The Pinebox provides **8 counters** (indices `0..7`) and **8 triggers** (indices `0..7`). Triggers monitor IO-Link process data values and increment or decrement a target counter when a configured condition is met.

The counter/trigger engine must be explicitly started and stopped.

Counter Values and Labels

Counter Values

`GET /counter/values`

Returns the current value of all 8 counters.

Response

```
{
  "counter0": 42,
  "counter1": 0,
  "counter2": 0,
  "counter3": 0,
  "counter4": 0,
  "counter5": 0,
  "counter6": 0,
  "counter7": 17
}
```

Counter Config

`GET /counter/config`

Returns the label configured for each counter.

Response

```
{
  "counter0": { "label": "Part count" },
  "counter1": { "label": "Reject count" },
  "..."
}
```

Set Counter Label

`POST /<num>/counter/config/set`

Sets the label for counter `<num>`.

Request body

```
{ "label": "Reject count" }
```

Response

```
{
  "error": 0,
```

```
  "error_string": "no_error",
  "label": "Reject count"
}
```

Counter Reset

Reset Counter

`POST /<num>/counter/reset`

Resets counter `<num>` to `0`.

Response: Error envelope

Reset All Counters

`POST /counter/reset`

Resets all 8 counters to `0` in a single call.

Response: Error envelope

Trigger Configuration

Get Trigger Config

`GET /<num>/trigger/config/get`

Returns the full configuration of trigger `<num>`.

Response

```
{
  "enabled": true,
  "port": 0,
  "pd_index": 2,
  "mode": "rising_edge",
  "threshold": 100,
  "filter_ms": 10,
  "target_counter": 0,
  "operation": "increment"
}
```

Field	Type	Description
`enabled`	bool	Whether this trigger is active
`port`	int (0–3)	IO-Link port to monitor
`pd_index`	int	Index of the PD-in variable to monitor
`mode`	string	Trigger condition (see table below)
`threshold`	number	Threshold value for `threshold_above` / `threshold_below` modes
`filter_ms`	uint32	Debounce time in milliseconds
`target_counter`	int (0–7)	Counter to affect when the trigger fires
`operation`	string	`increment` or `decrement`

Trigger modes

Mode	Description
`rising_edge`	Fires on low→high transition
`falling_edge`	Fires on high→low transition
`both_edges`	Fires on any transition
`threshold_above`	Fires when value rises above `threshold`
`threshold_below`	Fires when value falls below `threshold`

Set Trigger Config

`POST /<num>/trigger/config/set`

Sets the configuration of trigger `<num>`. The request body uses the same fields as the GET response above.

Response: Error envelope

Engine Control

Start

`POST /counter/start`

Starts the counter/trigger engine. Triggers begin evaluating IO-Link process data.

Response: Error envelope

Stop

`POST /counter/stop`

Stops the counter/trigger engine. Counter values are retained.

Response: Error envelope

Status

`GET /counter/status`

Returns the current run state of the counter/trigger engine.

Response

```
{ "running": true, "error": "none" }
```