# ThermalDIAG – Rules

Conditions, filters and actions for programmable measurement and monitoring sequences.

| | | | |
|---|---|---|---|
| Software basis | **V1.440** | Date | **2026-11-24** |
| Language | **English** | | |

## Contents

1. Conditions: sources and comparisons
2. Filters: when may the rule trigger?
3. Numeric calculation and NumX variables
4. All 20 actions and their execution modes
5. Execution and observation
6. Time Delta and MQTT calculation

This description explains the Rules functions of **ThermalDIAG V1.440, dated 2026-11-24**.

Each rule has a name, a **severity of 1–10**, a **cooldown**, an order and an enabled state. Conditions use **AND** (all must be true) or **OR** (at least one must be true). A rule list defines the required behaviour without classic program code.

# 1. Conditions: sources and comparisons

Available sources depend on the configured devices and project variables.

| Source | Available values |
|---|---|
| AX8 thermal measurements | Spot temperature; Box minimum, maximum and average temperature. |
| Camera and Modbus I/O | Configured numeric measurements, digital states and supported rising/falling edges; counter-capable Modbus inputs also expose counter value and increment. |
| MQTT devices | Numeric and Boolean signals, including configured OpenMV and Shelly values; communication-lost status is also available. |
| OPC UA variables | Readable Boolean, Int32 and Double variables; String variables support text equality and inequality. |
| TUF-2000M flow meter channels | Configured Modbus channels for flow rate, velocity, net/positive/negative total volume, measurement volume, measurement-running and measurement-valid states. |
| Numeric variables | Project variables Num1–Num100, usable as values, references, thresholds and calculation results; an OpenMV-N6 camera result stored in NumX can be checked by another rule. |
| System variables | `timer_sec` (Unix seconds), `local_minute_of_day` (0–1439), `local_day_of_week` (1 = Monday, 7 = Sunday); CPU usage, CPU temperature and RAM usage. |
| Constants | Numeric thresholds and references, including a left-hand constant for Delta; text constants for OPC UA String comparisons. |

| Condition | Evaluation |
|---|---|
| Direct comparison | Compare compatible values with >, ≥, <, ≤, = or ≠; digital-state and OPC UA text comparisons use = or ≠. |
| Delta | Compare the absolute difference, `abs(A − B)`, with a constant or NumX threshold using >, ≥, < or ≤. Supported sources include numeric variables, system variables, constants, measurements, counters and digital values as 0/1. |
| Temperature change | Detect a rise or fall in an AX8 Spot/Box over **1–899 seconds** using live samples; no threshold time or hysteresis. |
| Digital edge | Detect a supported camera/Modbus rising or falling transition; the sampling interval is selected to suit the pulse duration. |
| Communication lost | For MQTT sources: direct =/≠ comparison with 0 (healthy) or 1 (stale/offline); no hysteresis. |
| Vision inspection status | Check the status of the Vision job started by a selected rule action. |

# 2. Filters: when may the rule trigger?

Filters decide whether true conditions may open or reactivate an incident. There is one filter of each type per rule. Disabled filters are ignored, and **all enabled filters must pass**. Evaluation order is **Schedule → Previous incident window → Count window**.

| Filter | Purpose |
|---|---|
| Schedule | Allow triggering on selected weekdays within daily local-time windows, using the project time zone. Unchecked weekdays are blocked while the filter is enabled. |
| Previous incident window | Require another rule in the same project to have opened/reactivated an incident, or not to have done so, within the selected time window of up to **24 hours**. |
| Count window | Require a selected number of separate false-to-true condition activations within a time window. Continuously true conditions count once; the count resets after triggering. |

**Condition stability and freshness:** Threshold time requires a supported condition to stay true for a configured duration. A NumX duration is rounded down and frozen when the true candidate starts; NumX hysteresis is read again at each evaluation. Hysteresis is available only for supported numeric comparisons. MQTT and OPC UA sources use direct comparisons and Delta.

Evaluation uses fresh, valid readings. IPC CPU/RAM values have a freshness threshold of **15 seconds**. Local time values use the project timezone. Project text variables serve the Set text action, while OPC UA String sources serve text conditions.

## 3. Numeric calculation and NumX variables

A calculation action can assign a value or calculate with **+, −, ×, ÷, MOD and FMOD** and store the result in NumX. Sources include constants, NumX, system variables, AX8 temperatures, compatible numeric/digital Modbus values and counters, numeric OPC UA values and numeric or Boolean MQTT signals. Boolean values use 0/1. System variables are read as sources; results are stored in numeric project variables.

MOD uses rounded integer magnitudes; FMOD uses unrounded magnitudes. A zero divisor is rejected. Constants support up to **six decimal places**. The final result is rounded to the target precision using **ROUND_HALF_UP**. Measurement operands are captured for the queued action job; an earlier calculation action can update a variable used by a later action.

## 4. All 20 actions and their execution modes

In ordered execution mode, rules and their actions run in the configured order. The next rule reads updated values once the synchronous actions finish. Vision inspections run in the background.

**The three execution modes of an action**

The execution mode determines whether an action starts **when the rule becomes active, while it remains active or when its active state ends**.

Here, “active” means that the state monitored by the rule is present. This is separate from whether the rule is enabled in the settings.

**A – When the rule becomes active**

The action starts when the conditions are met, the filters allow activation and the rule changes from inactive to active.

The action is started once for that activation. While the rule remains continuously active, mode A does not start it again. If the active state later ends and the rule subsequently becomes active again, the action starts again.

*Example: When an overtemperature alarm begins, save a snapshot and switch on an indicator light.*

**W – While the rule is active**

The action can execute repeatedly while the rule remains active. This mode is suitable for operations that need to be repeated regularly during the monitored state.

Repetition in mode W continues while the rule is active. After the active state ends, this mode does not start any further repetitions.

*Example: While the alarm is active, a calculation repeatedly updates a numeric variable.*

**C – When the active state of the rule ends**

The action starts when a previously active rule returns to the inactive state, for example because the condition that caused the alarm is no longer met.

This is a single start associated with the end of that active state. **The required closing action must be configured separately: mode C does not automatically undo the actions executed when the rule became active.**

*Example: When the overtemperature alarm ends, switch off the indicator light.*

Different actions in the same rule can use different execution modes. Only the modes supported by an action can be selected for it.

The Mode column lists the available choices; it does not mean that all modes execute automatically at the same time.

| Action | Description | Mode |
|---|---|---|
| Create event | Record an event with an optional message and tags. | A |
| Email | Send a notification to configured recipients. | A |
| Voice | Speak a configured English, German or Italian message, with optional numeric placeholders and delay. | A, C |
| Snapshot | Save an image from a supported camera assigned to the selected location. For AX8 cameras, a Snapshot action can select the image mode for the capture. | A |
| Set output | Switch a permitted camera, Modbus or MQTT/Shelly output ON/OFF, optionally after a delay. | A, C |
| Cancel pending output | Cancel a waiting output action for the selected target without forcing it OFF. | A, C |
| Set impulse | Generate a timed pulse or pulse sequence with duration, pause, repetition and optional delay. | A |
| Stop impulse | Request a running pulse sequence to finish after at most one further pulse and cancel older waiting work for the target. | A, C |
| Camera torch | Switch supported camera illumination ON/OFF, optionally after a delay. | A, C |
| Cancel pending torch | Cancel a waiting camera-light action without switching the light OFF. | A, C |
| Numeric calculation | Assign or calculate a value and store it in NumX; use numeric and Boolean MQTT signals as sources. | A, W, C |
| Reset DI counter | Reset a digital-input counter that supports and permits resetting. | A |
| OPC UA write | Write a configured, correctly typed constant to a permitted OPC UA variable. | A, C |
| Start OpenMV-N6 camera check | Start an inspection on the assigned OpenMV-N6 camera using configured result and send-time NumX variables. | A |
| Get OpenMV-N6 camera check status | Query the current OpenMV-N6 camera inspection and publish its validated result into the configured NumX variable. | A, W, C |
| Set text | Replace a project text variable with configured language-specific text values. | A, W, C |
| Set display colors | Change the foreground and/or background color of a selected display cell. | A, W, C |

| Action | Description | Mode |
|---|---|---|
| Start TUF-2000M flow meter measurement | Start a measurement session using a fresh meter-counter boundary reading and a configured maximum permitted delay. | A, C |
| Stop TUF-2000M flow meter measurement | Stop the measurement session using a fresh boundary reading and retain the measured volume and session status. | A, C |
| Start Vision inspection | Start an inspection in the background and map selected numeric results to NumX variables. | A, C |

**OpenMV-N6 camera:** The Start action begins the inspection; Get status subsequently retrieves the validated result.

**TUF-2000M flow meter:** The actions define the start and end of a software measurement session using the meter totalizer. Pump control and management of the device counter are configured separately.

## 5. Execution and observation

Cooldown defines the wait before reactivation. Actions are processed individually in the configured order. Delayed jobs are scheduled; cancellation removes waiting jobs. Stop impulse ends the sequence in a controlled way according to its pulse parameters.

Device capabilities, write permissions, enabled runtime and licensing still apply. The **Incidents** view shows activations, filter blocking and action results.

## 6. Time Delta and MQTT calculation

**Time Delta:** Delta uses the absolute difference, `abs(A − B)`, including `timer_sec`. With `timer_sec` and NumX, the result is `abs(timer_sec − NumX)` regardless of operand order.

**MQTT calculation operands:** Numeric signals and Boolean signals as 0/1 are supported. The fresh, valid reading is matched to its configured source and retained for the queued action job.