Robot1 – User Manual

The BORUNTE robot control program, with HC controller over a network, includes: main window, poses, cycle, inputs and outputs, box packing, telemetry · an example of a configured "simple-pod" program (9 poses).

Robot1 connects to the robot controller via an Ethernet cable and provides the operator with everything needed for standard operation: positions, cycle through positions, response to sensors (inputs X), valve control and gripping (outputs Y), arranging parts in boxes in rows and layers, loading the finished program onto a USB drive for the teach pendant, and monitoring the robot as it operates according to its program.

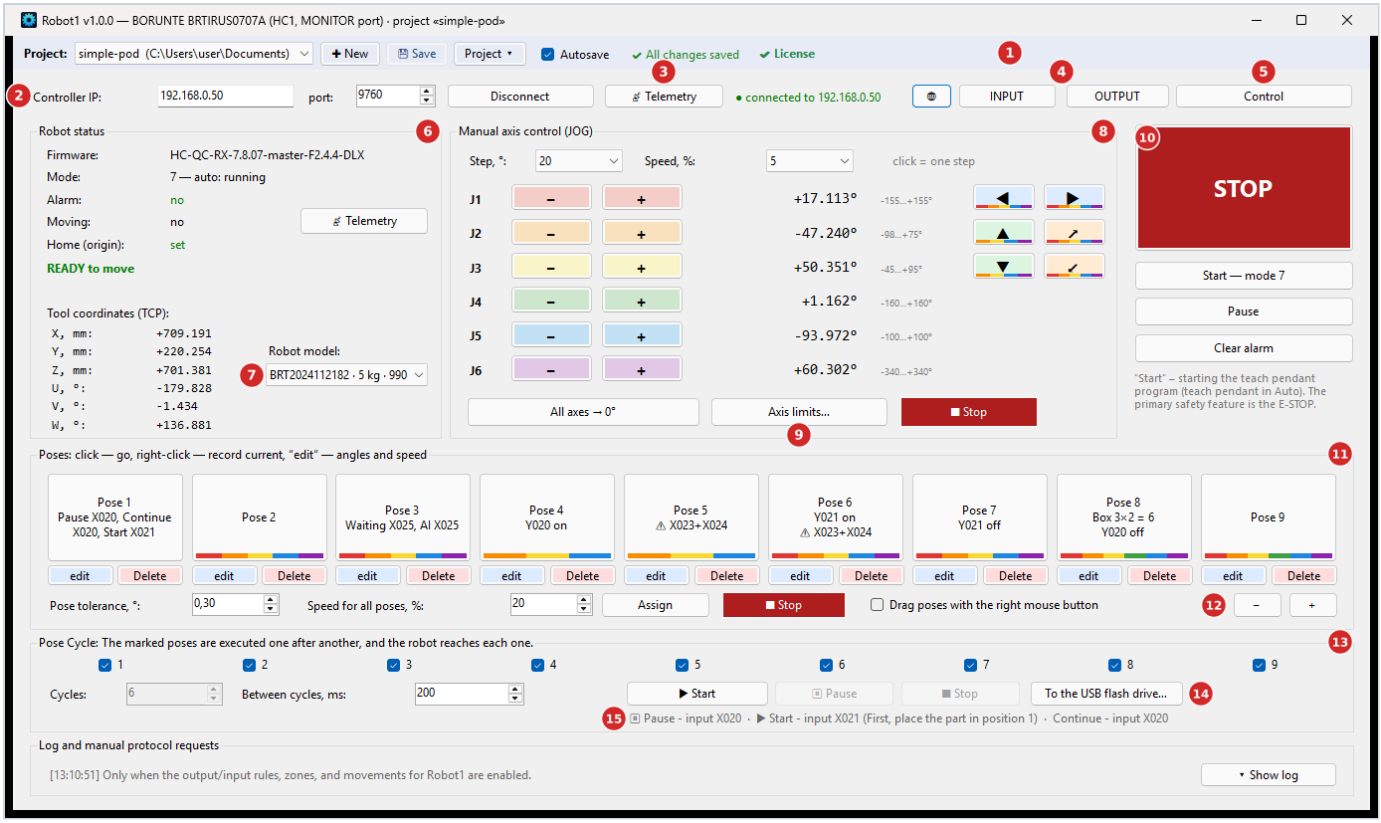
Computer	**Ethernet**	**HC Controller**	**Inputs: Sensors, buttons**	**Outlets Y**
Robot1	192.168.0.50 : 9760	teach pendant in Auto mode		Vacuum, valves

There are two ways to work with the robot: they can be combined:

- **Robot1 controls the robot itself** – "Pose Cycle" in the main window. The remote control has initiated a program for sending remote commands (mode 7 – "auto: executing"), Robot1 sequentially sends poses, monitors inputs, activates outputs, and places parts in the box. Everything is working here: pausing and continuing with input, protecting parts, disabling axes, and AI reception.
- **The robot operates according to its program.** The "Save to USB..." button records the same cycle using the teach pendant program (archive `HCBackupRobot_…`), which is then loaded onto the control panel via USB. The robot no longer requires a computer; the **Telemetry** window displays what the robot is doing and allows you to specify the part number, send a signal via memory M, and write to registers Addr-800...809.

1. Main window: zones
2. Project
3. Connection
4. Robot status
5. Robot model
6. Telemetry – robot operating according to its program
7. Manual control of axes (JOG)
8. Axis limits
9. STOP, Start, Pause, Emergency Reset
10. Positions
11. "Position" window
12. Cycle through positions and "To USB drive..."
13. Outputs – OUTPUT
14. Inputs – INPUT
15. Packaging
16. Journal
17. "Control" button – only after training
18. Safety
19. **Part 2. The "simple-pod" Program: 9 Poses, Entries, and Exits**

1. Main window: zones



The main window is labeled "Robot1", belonging to the "simple-pod" project. The numbers correspond to the zones listed in the table below.

№	Zone	For what purpose
1	Project	Selection, creation, and saving of a project (positions, input/output rules, packaging, limits). Automatic saving, license.
2	Connection	IP address and port of the controller, "Connect/Disconnect" button.
3	Telemetry	Window for observing the robot, which operates according to a program from the USB drive (section 6).
4	INPUT / OUTPUT	Input and output pages for X and Y: lights, manual ON/OFF, all rules (sections 13, 14).
5	Control	Cameras, AI, 3D vision, and teach pendant software – **available only after training** (Section 17).
6	Robot status	Programming, mode, error, movement, zero, standby line, tool coordinates (TCP).
7	Robot model	Which Borunte robot is connected: factory limits and head position calculation depend on the model.
8	JOG	Manual operation: each axis moves in increments, and the head moves in a straight line while holding the part (color-coded buttons with arrows).
9	Axis limits...	Software-defined axis angles, saved sets, "Setup".
10	STOP and modes	Large "STOP" button, "Start - Mode 7", "Pause", "Emergency Reset".
11	Positions	Pose buttons: Left-click - move, Right-click - record, "Edit" - angles, speed, box, OUTPUT, INPUT.
12	General pose settings	Pose acceptance, speed for all poses, "■ Stop", "Move poses with the right mouse button", "- / +" – number of poses.

13	Pose cycle	Which positions in the cycle, how many cycles, pause between cycles, "▶ Start", "⏸ Pause", "■ Stop".
14	To the USB flash drive...	Record the cycle using the teach pendant program on a USB flash drive.
15	Input hints	Which inputs pause, start, and continue the cycle.

2. Project

The project consists of a single file `.r1proj` located in the `Documents\Robot1\Проекты` folder. This file contains all the settings for a specific task: poses and their speeds, the cycle, input and output rules, packing instructions, axis limits, and the robot model.

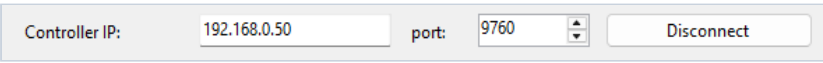
- **Project:** – a list of projects; selecting a different project immediately loads its settings.
- **+ New** – a blank project; **Save** – save now; **Project ▾** – "Save as...", "Rename...", "Delete project to trash...", "Open project file...", "Open project folder", "Undo unsaved changes".
- **Automatic Saving** – every change is automatically saved (the "Auto-saved at 12:02:00" timestamp is visible).
- **✓ License** – The license is active. The program functions in trial mode without a license key.

> For each new part or section, create a separate project – this prevents the settings of one operation from affecting another. Before making major changes, create a copy: "Project ▾" → "Save as...". The project file can be transferred to another computer and opened there using "Open project file...".

Tips

- **Project List** – select a different project – the program will reset all settings: INPUT, OUTPUT, Control, TCP, limits, poses, and 3D camera.
- **+ New** – a project with default settings: no rules, restrictions, TCP classes, or saved configurations.
- **Auto-save** – enabled: any changes (INPUT, OUTPUT, Control, TCP, limits, poses, 3D camera) are immediately saved to the current project. Disabled: changes are accumulated, and the "Save" function saves them to the project. When switching projects or exiting the program, it will ask the user what to do with the accumulated changes.
- 🌐 — the interface language. Translations are stored in the database robot1.db (the i18n table), and the selected language is remembered until the next launch.

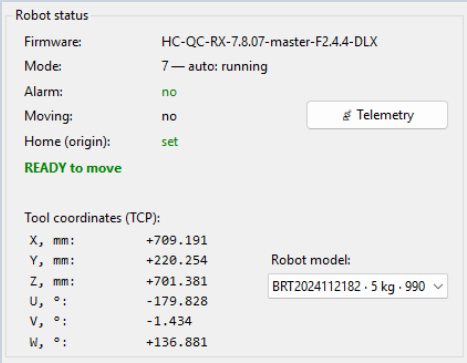
3. Connection



IP controller, port, connection button

- **IP address of the controller** – the robot's address on the network (for this robot `192.168.0.50`), which is set in the network settings on the remote control. **port** – `9760`.
- **Connect** – after connecting, the right side displays the text "● connected to 192.168.0.50", and the button changes to "Disconnect". Robot1 queries the controller every 0.3 seconds.
- The computer must be on the same network (e.g., 192.168.0.10, with a subnet mask of 255.255.255.0). Detailed instructions on configuring the network and the remote control program are provided in the separate instruction manual "BORUNTE – Remote Control via Network".

4. Robot status

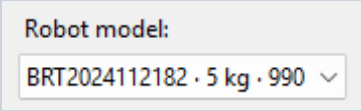


"Robot Status" block

- **Firmware** – the version of the controller (here, HC-QC-RX-7.8.07-master-F2.4.4-DLX).
- **Mode** – what the controller does (see the table below).
- **Error** – "no" means green; otherwise, the error code is red. When an error occurs, the cycle is paused.
- **Movement** – indicates whether the robot is currently moving.
- **Zero (origin)** – indicates whether the zero axis is active. If the zero axis is active, the robot will not move.
- **Status Line** – the red "NOT READY: ..." indicates what needs to be done (e.g., "press Start – Mode 7"); the green indicates that the robot is receiving commands.
- **Tool coordinates (TCP)** – X, Y, Z in mm and angles U, V, W – the tool's position as determined by the controller.
- **Telemetry** – This refers to the same window as the button at the top (section 6).

Mode	What it means	Accepts movement from a computer
1 — Manual	Using the teach pendant in manual mode, working with the teach pendant	No
2 — Auto: Stopped	Using the teach pendant in Auto, the program is not running	No — press "Start — Mode 7"
7 — Auto: Running	The remote control program (or the program from the USB drive) is running	**yes** (for remote command execution)
3 — Stop, 8 — Step-by-step, 9 — One cycle	Stopping, debugging step-by-step, running the program once	No

5. Robot Model



Model Selection

The list includes all 6-axis Borunte robots with the same HC controller and remote, with the same dimensions, payload, and reach: BRTIRUS0401A, 0707A, 0805A, BRT2024112182, 1510A, WD1506A, 1820A, 2030A, 2110A, WD2206A, 2550A, 2520B, 3030A, 3511A. The connected robot is **BRT2024112182** (this is BRTIRUS0805A: 5 kg, 990 mm, the dimensions of the links were measured on the robot itself).

The following depends on the model:

- **factory limits for axes** – the limits within the "Axes Limits..." window and the labels on the right of the corners in JOG; software limits are automatically trimmed to the selected model.
- **Calculating head position** – for colored JOG buttons with arrows, vertical posture, check-taking and packing into a box (rows, layers, top-down approach).

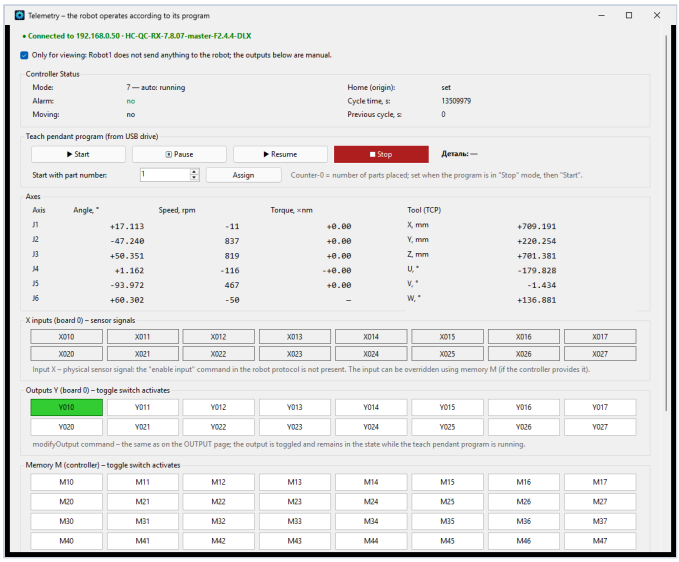
> When connecting Robot1, it automatically compares the calculation with the controller's coordinates. If the wrong model is selected, but a different one is suitable, it will switch to it and log the result, for example, "Robot model BRT2024112182 matches the controller: 0.8 mm".

Tips

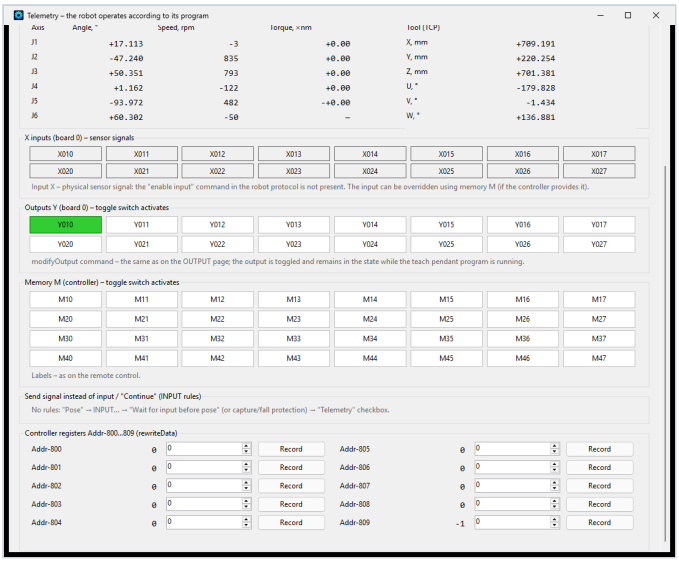
- **Robot Model** – 6-axis Borunte robots with HC controller: dimensions for calculations (box, vertical travel, grip verification), factory limits and axis speeds. This is not the correct model; the calculations differ from the robot by tens of millimeters. When connected, the program compares the models with the controller's coordinates and selects the appropriate one.

6. Telemetry – The robot operates according to its programmed instructions

Telemetry – a display showing the status and performance data of the robot while it is operating **Not under control Robot1**...and according to its own program, which is stored on the control panel (for example, loaded via a "To USB drive..." button). In this case, the logic and algorithms reside within the robot itself: it waits for inputs, checks sensors, counts parts, and arranges them in designated compartments. Robot1 It simply observes and, when necessary, transmits values to the robot via [a specific interface/method] **Memory M**, **Counter-0** and **Address registers Addr-800...809** — The robot's program reads them automatically.



Upper section: status, teach pendant program, axes, inputs, outputs, M memory



Lower section: M memory, input signal, Addr-800...809 registers

Block	What it does
Only for viewing	Checkmark (enabled by default): Robot1 does not send any commands to the robot – the OUTPUT/INPUT rules, restricted areas, and main window movements are disabled. You control the outputs, M, and registers within this window.
Controller Status	Mode, error, movement, zero, cycle time and previous cycle time.
Teach pendant program (from USB drive)	▶ **Start**, ⏸ **Pause**, ▶ **Continue**, ■ **Stop** – control the program running on the remote. "Detail: N" – the number of items placed. **Start with item number** + "Set" – record the Counter-0 counter (= number of items placed): the program will continue placing items starting from the specified box compartment. This is set when the program is stopped ("Stop"), then "Start".
Axes	Angle of each axis, speed (RPM), torque (as a percentage of nominal), and tool coordinates (TCP).
X inputs	Sensors on the 0 (X010–X027) circuit board. The input is a physical signal: it is not possible to activate the input via the network; it can be replaced with memory M.
Y outputs	Clicking the button toggles the output (green = on). It works while the teach pendant program is running.
Memory M	Clicking the button toggles the controller's memory bit (as on the remote control: M41 = point 25). The robot program can wait for M instead of an input. M30 and M34 are occupied by the robot program – do not interfere.
Send a signal instead of an input / "Continue"	Buttons for INPUT pose rules with the "... from Telemetry" checkbox: "wait for input before pose," protection against grabbing and falling. The administrator sends a signal via the Robot1 button and the program on the flash drive (via the selected memory M).
Addr-800...809 registers	Current values and "Write". Numbers are transmitted to the robot program here: part number, layer, recipe, operating mode. The written value is read back (verified on this robot).

> The Telemetry window does not move the robot. However, the Y and M output buttons directly control the real robot – ensure it is safe before pressing (valve, gripper, program signal).

Tips

- **Only observe** — Robot1 does not send anything to the robot: the OUTPUT/INPUT rules, restricted areas, and movements from the main window are disabled. You control the exits in the window yourself.
- **Memory M** – labels – as on the remote control (M41 = point 25).
- **Address registers Addr-800...809** – the value stored in these registers is read back.

7. Manual axis control (JOG)

- **Step, °** – the number of degrees (or millimeters for buttons with arrows) that one click corresponds to. **Speed, %** – the speed of manual movement; start with 1–5 %.
- **J1...J6 "–" / "+":** Rotate one axis by one step. Each axis has its own color (rainbow): J1 is red, J2 is orange, J3 is yellow, J4 is green, J5 is blue, and J6 is purple; the same colors are used in the stripe on the position buttons. On the right, you see the current angle and the limits of the axis (e.g., -155...+155°). Values exceeding the limits are truncated to the boundary, and movement is stopped at the boundary.
- **Arrow buttons** – The head moves in a straight line, one step at a time, in millimeters. All axes hold the part in its original position (it does not rotate or tilt). The color scheme is: group – straight line, arrow – direction (as can be seen, standing behind the robot facing the box):
  - Blue ◀ ▶ (J1 line) – left/right within the row (equivalent to "left of the first" in the box);
  - Green ▲ ▼ (left column) – strictly **up** / **down**;
  - Orange (↗ ↙ - right column) indicates: **further** away from the robot / **closer** to the robot.

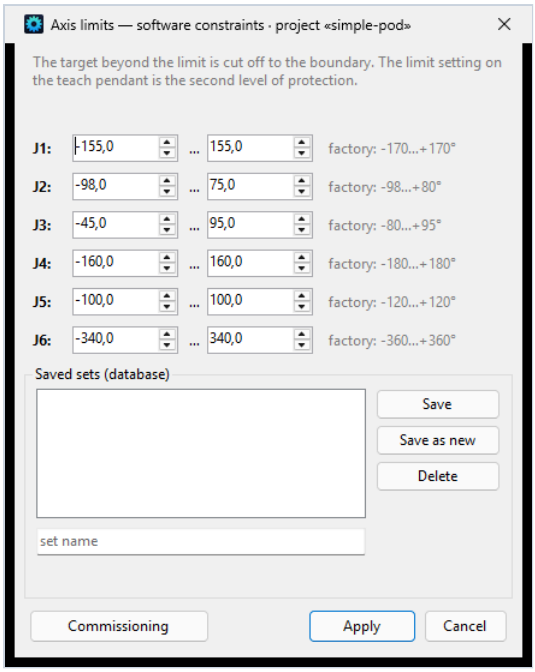
  The colored bar at the bottom of the button with the arrow indicates the axes that will rotate during the movement (the colors correspond to the axes J1...J6); the sequence is determined based on the robot's current position.
- **All axes → 0°** — set all axes to zero (available when it is safe to do so).
- ■ **Stop** – to halt manual movement.

> The buttons with arrows allow you to precisely position the part above the desired location and record the position using the right-click function – the part will remain parallel to the already laid pieces.

Tips

- ◀ ▶ **along a row** – the head moves along a straight line parallel to the base axis (X or Y) in increments of the "Step" field, mm, at the same height: ◀ left, ▶ right (as in the box). The part does not rotate or tilt – it remains parallel to the laid-down parts.
- ▲ ▼ **Up / Down** – The head is positioned vertically, with a step of 1 mm from the "Step" field; X and Y values remain unchanged (kinematics of the selected robot model).
- ↗ ↙ **further / closer** – head aligned along the other axis of the base – either from the robot or towards it – in a step of "Step", mm; height remains the same.

8. Axis limits



"Axis Limits..." window

Software-defined angle limits represent the first line of defense. Any target (JOG, pose, cycle, entry rule, box) outside the defined limits is truncated to the **boundary**. To the right, in gray, are the factory limits for the selected model ("factory: -170...+170°"): software limits are not set beyond these. The factory limits set on the remote control represent the second line of defense.

- **J1...J6** — lower and upper limit, in degrees.
- **Saved Sets** – "Save," "As New," "Delete" with set name: different restrictions for different areas.
- **Commissioning limits** – These are the reduced, safe limits used for the initial startup.
- **Apply** – enable for the project; **Cancel** – close without making any changes.

In the "simple-pod" project: J1 -155...+155, J2 -98...+75, J3 -45...+95, J4 -160...+160, J5 -100...+100, J6 -340...+340°.

Tips

- **Limits** – are checked within the program: the target area is cropped to the boundary before being sent to the robot. Factory limits set on the controller (adjustable via the remote) serve as a secondary layer of protection.
- **List of Sets** – click – load set values into fields; change fields – "Save" will update the selected set, "As New" will create a separate entry.

9. STOP, Start, Pause, Emergency Reset

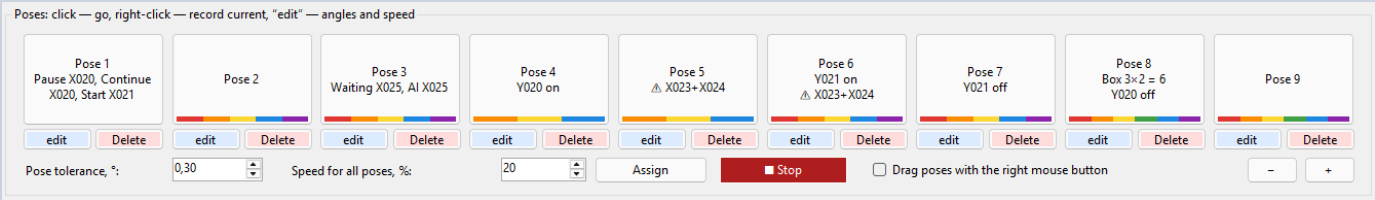
- **STOP** (large red) – immediate halt of movement and the cycle.
- **Start — Mode 7** — Initiate the remote control program (the teach pendant should be in Auto mode with the program open). After this, the status bar will turn green.
- **Pause** – To halt the operation of the teach pendant (pause command over the network).
- **Resetting an accident** – resetting the controller after the cause of the accident has been eliminated.

> **The primary safety feature is the hardware E-STOP button,** located on the control panel and at the work station. The buttons displayed on the screen are dependent on the network and computer system and do not replace the emergency stop function.

Tips

- **Start** – initiates the teach pendant program remotely: the teach pendant must be in Auto mode with the program open. The primary safety feature is the hardware E-STOP.

10. Positions



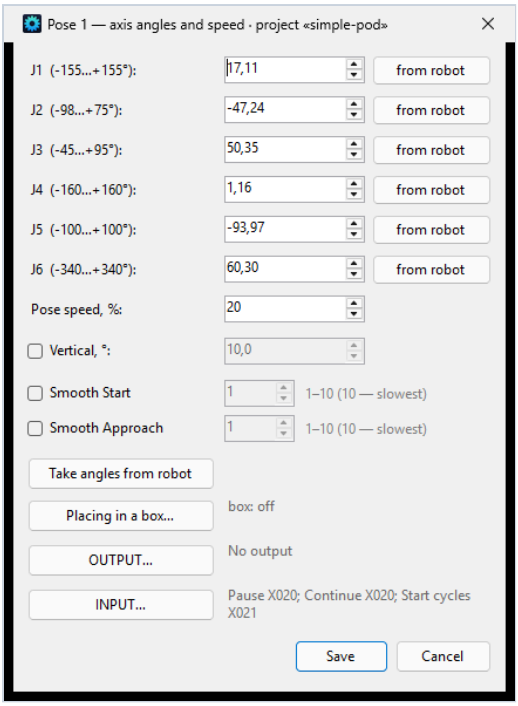
"Presentation Positions" block: buttons with labels for rules, "edit", "extend", and general settings

- **A colored bar at the bottom of the pose button** – the axes that move into this pose (the angle is greater than that of the previous pose; for pose 1, it is from the last); the colors are the same as the axes buttons J1...J6. The vertical pose is J2, J3, J5. Buttons under the pose: blue "remove" – angles and speed, red "delete" – delete the pose.
- **Move to Pose by Button** – The robot moves to this pose (at the pose speed). If the pose has exits, they are activated when the robot reaches the pose.
- **Record robot pose angles for the current pose** – Record the current angles of the robot in the current pose.
- **rare** – pose: angles, speed, vertical pose, smoothness, box, OUTPUT, INPUT (section 11). **remove** – delete pose.
- **Button labels** – what is configured for the "Y020 on" pose: "exit in pose, waiting for X025", "⚠ X023+X024" – fall protection, "3x2 box = 6" – stacking, "pause X020, continue X020, start X021" – general entry rules, set from this pose window.
- **Acceptable pose deviation, °** – how close (along each axis) the robot must get for the pose to be considered achieved and for its associated actions to be triggered. The specified value is 0.30°.
- **Speed for all positions, % + Set** — sets the same speed for all positions simultaneously.
- **■ Stop** – to bring the movement to a halt, assuming a resting position.
- **Enable drag-and-drop with the right mouse button** – This allows you to drag and drop the "right" button for each pose using the mouse, and to change the order: click and hold the right button on a pose and drag it to a different location. The numbers in the input and output rules will move along with the pose. Clicking the right button without dragging still saves the pose. This feature is disabled by default to prevent accidental movement of the pose.
- **– / +** — Remove the last pose (with confirmation) or add a new one: by default, at the end; you can insert it earlier, which will shift the subsequent poses forward by +1; the new pose takes the angles from the previous one.

Tips

- **Pose Buttons** – Left-click: Move to pose, Right-click: Save current pose, "Edit": Set angles and speed.
- **Move poses with the right mouse button** – enabled – hold down the right mouse button on a pose and drag it to the desired location: the poses will swap places; disabled – clicking the right mouse button still saves the current pose.
- **Position Tolerance** – This indicates that the robot has reached the target position, with each axis within the specified tolerance and movement complete. The same tolerance values are also displayed on the OUTPUT page. 0.5° – This represents a margin of error: after the axis stops, its position is indicated in hundredths of a degree.
- «+» **— Add pose** — Default number — the one that comes after the last; it can be set earlier, in which case the poses numbered from this point will shift forward by +1. The new pose takes its angles from the previous one.

11. "Edit" position window



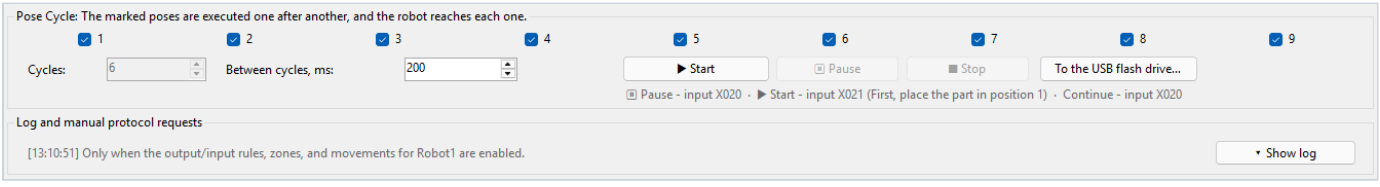
Pose 1 – Axis Angles and Speed

- **J1...J6** – joint angles, in parentheses – axis limits. **from the robot** – to take the angle of one axis from the robot; **Take angles from the robot** – all at once.
- **Pose speed, %** – the speed at which the pose changes.
- **Vertical, °** – The position is determined by raising (+) or lowering (-) from the point where the robot is positioned: the head moves strictly vertically, and the part does not tilt.
- **Smooth start 1-10** – a slow start to the movement towards the pose; **Smooth approach 1-10** – a slow approach (10 being the slowest). The pose speed can be maintained at a high level.
- **Arranging in a box...** – position the item within the box (section 15). On the right, you will see "box: off" or "3x2".
- **OUTPUT...** – These are the movements that occur in this position (Section 13).
- **INPUT...** – This section lists the specific rules for entering this position, as well as the general rules (Section 14). On the right, you'll find a brief summary, for example: "Pause X020; Continue X020; Initiate cycles X021".
- **Save / Cancel**.

Tips

- **from the robot** – to set the angle at which this robot's axis is currently oriented; the other axes remain unchanged.
- **Vertical** – the J2, J3, and J5 axes operate as a single unit: the position is determined by the angle of elevation from the robot's starting point (+10 degrees = 10° upward, -10 degrees = 10° downward). The head moves strictly vertically, and the tool's angle does not change; J1, J4, and J6 remain in their original positions. The angle refers to the rotation of the forearm of J3.
- **Smooth Start** – the first 20% of the journey to the robot pose is slow, with the remaining portion at the speed of the pose. 1 represents a slightly slower speed, while 10 represents 10% of the normal speed, without any pauses.
- **Smooth Approach** – the final 20% of the journey, where the robot approaches slowly and gently. 1 represents a slightly slower speed, while 10 represents 10% of the maximum speed.

12. Cycle through Poses and "To Flash Drive..."



Cycle through Poses and Log

- **Positions 1...9** – these are the positions that make up the cycle; the robot moves through these positions one after another, reaching each one.
- **Cycles** – the number of times the route is traversed. If the route includes a "packing" operation (i.e., placing items in a box), the field is locked and set to the number of spaces in the box (in the 3 x 2 example, this would be 6).
- **Between cycles, ms** – a pause after the final pose before the next round (as specified in the 200 ms project).
- **▶ Start** – to begin a cycle; after a pause, the button continues from the same position. **⏸ Pause** – to stop, without resetting the position in the cycle. **■ Stop** – to stop and reset: the next start will begin from the

beginning.

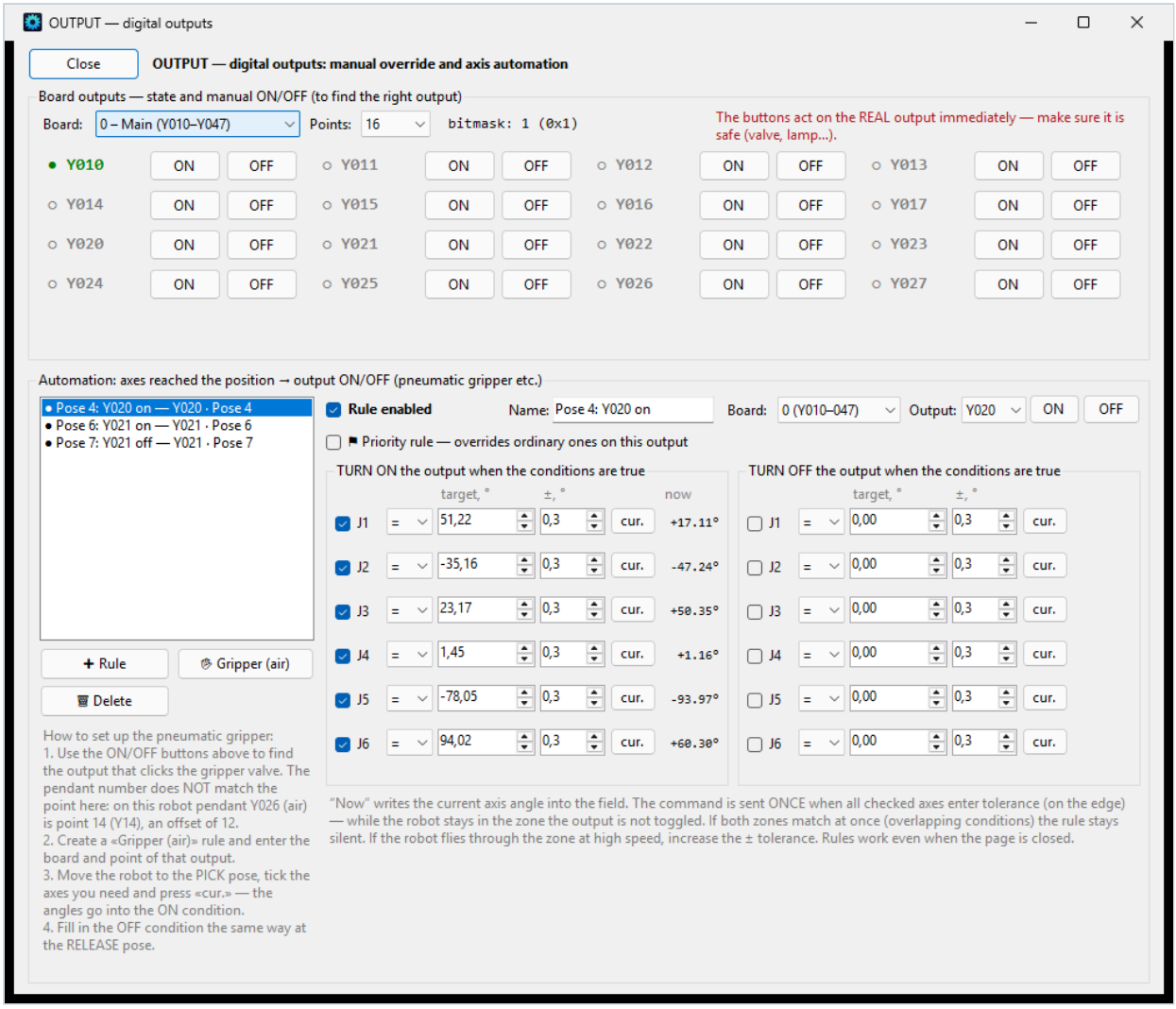
- **Button-based prompts** – cycle inputs: "⏸ Pause – input X020 · ▶ Start – input X021 (initially in position 1) · Continue – input X020".
- **To the USB drive…** – record the cycle using the teach pendant program: specified poses × cycles, pose outputs, parts counter box, input waiting, sensor checks, start with input. On the remote: "IMP USB", "one cycle" mode. The robot for unloading is not needed. Anything entered into the program is written to the log.

> The "Pause" and "Continue" functions, as well as the axis locking and AI-based input features, only work within the Robot1 loop – they are not included in the program on the flash drive (as indicated in the log).

Tips

- **To a flash drive…** – record this cycle using the teach pendant program on a flash drive: marked positions in sequence × number of cycles. Archive HCBackupRobot_… is loaded via "IMP USB"; "one cycle" mode – circles and stop until the next Start. A robot for recording is not needed.

13. Outputs – OUTPUT



OUTPUT Page: Outputs of the board and "Axis arrived at position → output"

Manual Control of Outputs

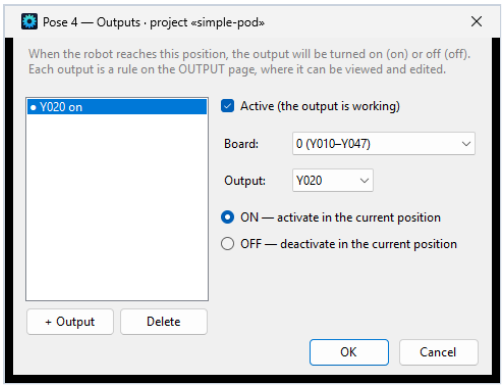
Above, you'll see all the outputs of the selected board (Y010–Y027 **on** board 0), each with its associated lamps and ON/**OFF** buttons. These buttons directly control the actual output, allowing you to identify which output is responsible for controlling the gripper valve, activating the blower, turning on a lamp, etc.

Axis Automation

Rule: *The output will be ON or OFF when the axes reach the specified angles.*

- **Rule enabled**, **Name**, **Fee**, **Output**, ON/OFF buttons for verification.
- **ENABLE output when the condition is true** – Axes marked J1...J6: condition (=, ≥, ≤), target angle in degrees, tolerance ±. **text** records the current angle of the axis.
- `**DISABLE the output when the condition is true**` – this applies to disabling as well.
- The command is executed once when the axes enter the zone; while the robot is in the zone, the output does not fluctuate. If the robot passes through the zone at high speed, increase the tolerance.
- **The primary rule** overrides the standard rules for the same output.
- **gripper (air)** – a pre-configured rule for a vacuum gripper: ON in the grasping position, OFF in the releasing position (the configuration order is written from left to right on the page).

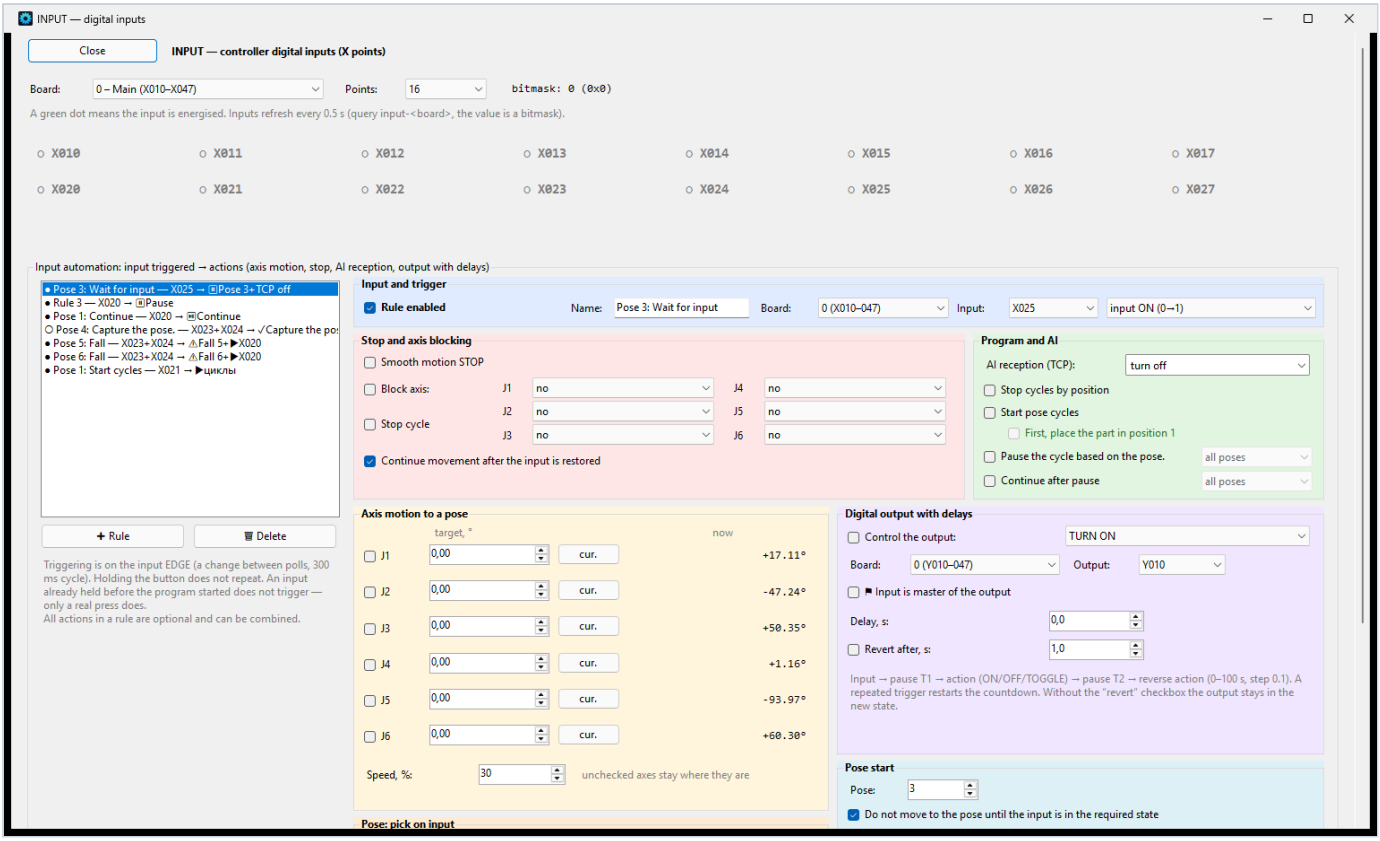
Duplication: OUTPUT... in the pose window



Pose 4 – Outputs

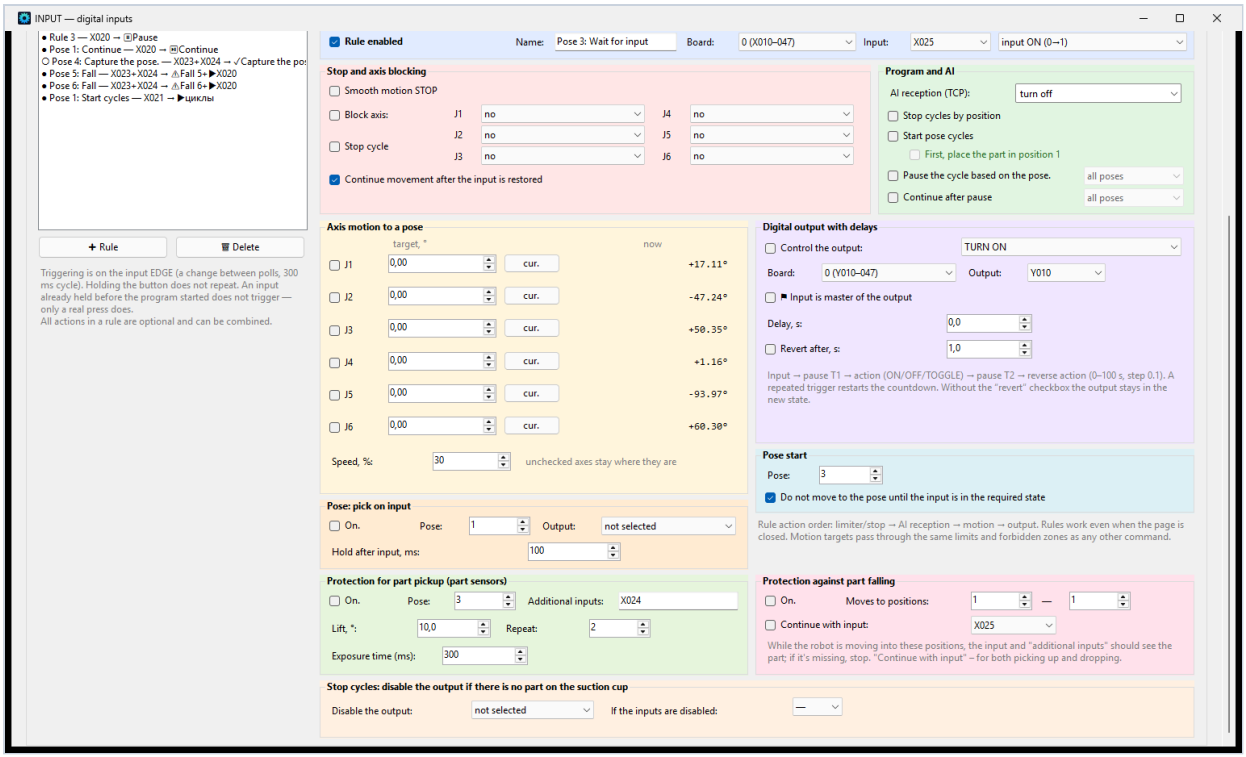
The "**OUTPUT...**" button in the pose window displays and modifies **the same rules** as the OUTPUT page, but specifically for that pose: it controls the output, enabling or disabling it **ON - enable in the pose** or **OFF - disable in the pose**, and indicates whether it is "Active." "+ Output" adds a new output, and "Delete" removes one. A rule created here is visible on the OUTPUT page with the name "Pose 4: Y020 on," and vice versa. The outputs for the pose are activated when the robot reaches that pose.

14. Inputs – INPUT



INPUT page: input channels and "input triggered → actions" rules list

Above are the input lamps for the selected board (X010–X027): a green dot indicates an active input. Below are the rules. A rule is triggered by the **edge** of the input (the transition between scans, 300ms clock): holding the repeat button does not trigger the input, which is pressed until the program starts. All actions in the rule are optional and can be combined; the order of execution is: limit/stop → AI reception → movement → output. The rules work even when the page is closed.

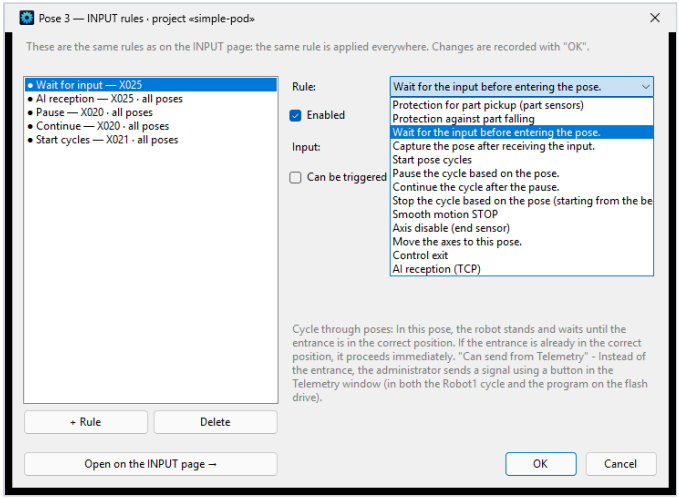


Lower part of the INPUT page: pick-up by input, protection of parts, stop cycles with output disable

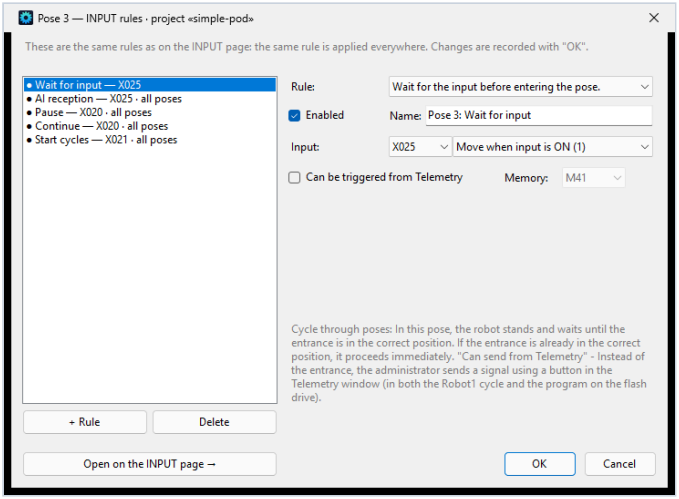
Block	Parameters and logic
Input and triggering	Rule enabled, Name, Cost, Input (X...), front: "input ENABLED (0→1)" or "input DISABLED (1→0)".
Smooth STOP of movement	Input stops the robot's movement.
Axis disable (end sensor)	For each axis J1...J6: no / down (−) / up (+) / down and up. If the input is activated, the robot stops and does not move in the prohibited direction; it can move in the opposite direction. The cycle pauses. **Stop cycle** – instead of pausing, the cycle stops completely. **Continue movement after input recovery** – if the input is restored, the cycle continues automatically; if disabled, it waits for "▶ Continue" or the "continue" input.
Program and AI	**AI-based admission (TCP)**: do not change / enable / disable - receiving commands from the camera/AI. **Stop looping poses** - Stop and reset the cycle; if the cycle is already stopped, the robot moves to position 1 (which can then be used to restart the cycle). **Initiate pose sequences** (+ "first, into position 1") - functions like "▶ Start"; during the cycle and when paused, the button press is ignored. **Pause the cycle based on poses** - like " ⏸ Pause"; "all poses" or only in the selected pose. **Continue after the break** — such as "▶ Continue"; "all poses" or only when the cycle is in the selected pose. The "pause" and "resume" functions work together on a single input **switch**Press once to pause, press again to resume.
Axis movement to a position	Marked axes move to specified angles ("curr." - take current), speed %; unmarked axes remain in place.
Discrete output with delays	Control the output: ON / OFF / SWITCH, along with the input. Input → T1 pause (delay) → action → T2 pause (return) → reverse action; 0–100 ms, step 0.1. Re-triggering restarts the countdown. "Input is primary for output" overrides the OUTPUT rules on this output.

Initiating the pose (= "Waiting for the pose to be initiated")	Pose N + "Do not move to pose until the input is in the correct position": Before this pose, the cycle waits for the input level. Once in the correct position, it moves immediately.
Pose: Execute based on input	While the robot is moving to pose N, the input is triggered → stop axes, output ON (gripper), "Stand after input, ms" – the pose is considered complete. To pick up a part whose position is not known (the robot lowers until the sensor touches).
Protection for part pickup (part sensors)	On/Off, Position, Additional Input (second sensor), Lift Angle (°), Number of Repetitions, Exposure Time. The "Protected Position" becomes **a verification lift**: the robot takes the part in the previous position, waits for the exposure time, lifts vertically to the specified angle (≈ mm indicated), and looks at the sensors. Everyone sees the part – proceed; no – return to the previous position, take it again and lift again; attempts are exhausted – pause.
Protection against part falling	Enable, "Moving to Position: from ... to ...", "Continue with Input". While the robot is moving the part to these positions, the input and "additional inputs" should see the part. If even one is missing, the cycle immediately stops, and the cycle is paused. "Continue with Input" applies to both picking up and dropping.
Stop cycles: disable the output if there is no part on the suction cup	When stopping the cycle with input, disable the output (e.g., air) only if all specified sensors for the part are disabled – this prevents dropping the part.

Duplication: INPUT... in the position window



List of "Rule" in the INPUT position window – all 13 types

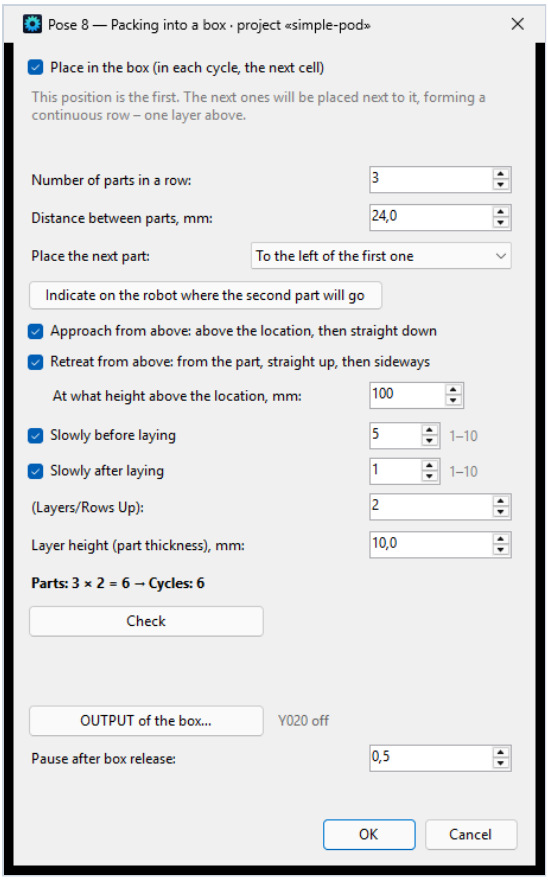


Position 3 – rule "Wait for Input before Position" X025

The `**INPUT...**` button in the pose window applies the **same rules** as the INPUT page: one rule per object, and changes are visible in both locations (and are saved when you click "OK"). The "Rule" list displays all the settings from the INPUT page: protection against grabbing, protection against falling, wait for input before executing the pose, grab on input, run cycles, cycle pause, continue after pause, stop cycles, smooth stop, axis lock, axis movement to this pose, output control, and AI-assisted execution.

- The rules for **this position** (entering, escaping, defenses, and movement into this position) are explained simply: "Wait for the opening – X025."
- **General** rules (start, stop, pause, continue, prohibition, exit, AI) are visible in the window for each pose with the label "· all poses". The "pause" and "continue" functions can only be used "within this pose".
- If one input has multiple actions (e.g., start and stop on one button), then each action has its own copy of the rule.
- The rules created from the "window pose" are associated with the button for this pose in the main window; **Open on the page INPUT** → – navigate to the rule on the full page.

15. Packing into a box



Position 8 – packing into a box

The "laying down" position refers to the **location of the first piece**. In each cycle, the robot places the piece in the next cell: first filling a row, then the next layer above. Each piece is placed in the same orientation as in the "laying down" position – the robot calculates the angles for each cell based on the kinematics of the chosen model and places it directly, not diagonally.

- **Place in a box** – turn on.
- **Number of components in the row** and **Distance between components, mm**.
- **Place the next part:**: to the right of the first / to the left of the first / closer to the robot / further from the robot (as you can see, standing behind the robot facing the box).
- **Show the robot where the second one will go** – the robot will slowly (10%) position itself above the first, then above the second cell. If it goes to the wrong place, choose a different side.
- **Top-down approach** – the robot first moves to the specified height above the target location, and then descends precisely downwards.
- **Moving upwards** – after laying it down in a straight, upward direction, then moving it to the side – this prevents the part from sliding along the template.
- **What is the height above the box edges and placed parts, in mm?** — slightly more than the box edges and the placed parts.
- **Slowly before laying out 1–10** – the last 20% of the path is slow; **Slowly after laying out 1–10** – the first 20% of the path is slow (10 – 10% of the speed of the position).
- **Layers (from top to bottom)** and **Layer height (part thickness), mm**.
- "Details: 3 x 2 = 6 → cycles: 6" – The number of cells determines the number of cycles. **Check** – This calculates all cells: does the robot pick up the items, or does it exceed the limits?
- **OUTPUT boxes...** - These are output signals that are activated in **each** cell when the robot reaches that location (e.g., "Y020 off" means to release the part). A pause is included after the output signal is activated to allow the valve to fully release.

> The program works both in the Robot1 loop and in the program on the flash drive: in this case, the cells follow the Counter-0 counter, so in Telemetry, you can "start with part number #".

Tips

- **The first item** – this position is the first item in the box. The others will lie next to it at the same height and orientation; the row is full, and the next layer will be placed above. You can check the position using the "Show on robot" button.
- **Position** – to the right/left – as you can see, when standing facing the robot with your back to the box: the parts are laid down in a straight line along the base axis. Closer/further – along the other axis of the base, either towards or away from the robot.
- **Display on Robot** – The robot will slowly (10%) position itself above this pose, then above the location of the second part, at a height of "above the location" (at least 50 mm). If it doesn't position itself correctly, choose a different side and check again.
- **Moving Up** – After placing the part, the robot first moves it strictly upwards from the part (to the approach height) and then moves to the next position – it does not drag or pull the part or suction cup sideways.
- **Top-Down Approach** – The robot moves horizontally to position itself above the desired height, then descends precisely downwards, places the part, and then precisely ascends back up – without touching the box or the stacked parts. It also functions with a program stored on a USB drive.
- **Distance from the edge** – the robot moves downwards by the specified number of mm; ensure that the edges of the box and the already placed parts are slightly higher.
- **Slowly before/after placement** – the final 20% of the journey to the destination (the first 20% from the destination) should be slow: the item should gently settle into place, and the suction should release smoothly. 1 = slightly slower, 10 = 10% of the speed of movement; the speed of movement can be maintained at a high level.

16. Log

At the bottom of the main window, you'll find: **Protocol Log and Manual Requests:** Each event, such as an entry, exit, pause, or stop, is recorded as a line with a timestamp, for example, " ↯ Entry X020 triggered → rule...", "Cycle through poses: Pose 3 is waiting for entry X025", "⚠ PART LOST on the way to pose 5 - no signal X024". **Show Log** expands the log. If you encounter any unclear stops, first check the log – it will tell you what happened and what to do.

17. "Control" button – only after training

The "Control" button functions require specialized training.

Behind the "Control" button are tools for configuring complex systems: industrial machine vision (cameras from **SICK** and **HIKROBOT**), **3D cameras (stereoscopic vision)**, receiving coordinates from AI and cameras via TCP, and teach pendant routes and programs. Incorrect configuration of these tools can cause the robot to move to unintended locations. Robot configuration with industrial and 3D vision is only performed by a trained specialist. This manual does not describe these functions; please contact the supplier for training and configuration.

18. Safety

- Before starting – no one should be in the work area, the enclosure should be closed, and the E-STOP should be checked.
- Try a new position and cycle at speeds of 1–5%; for the working speed, do so after a complete run without errors.
- The output (OUTPUT) and telemetry buttons should function immediately: before pressing, check that they are connected to the output.
- The project's axis limits – should not be wider than necessary; when changing the robot, select its model.
- In case of an emergency: E-STOP → cause → "Emergency Reset" → "Start – Mode 7" → continue the cycle.

Part 2. "simple-pod" program: 9 positions, inputs and outputs

The "simple-pod" project – the robot picks up a part, moves it through an intermediate operation, and places it in a 3 × 2 box (three in a row, two layers). The BRT2024112182 (BRTIRUS0805A) robot, all 9 positions in the cycle, each position at 20%, 6 cycles (= 6 places in the box), 200 ms between cycles, position tolerance of 0.30°.

Program inputs and outputs

The robot receives instructions via the following inputs: **part arrived**, **part grasped/held**, **pause/continue**, **start**. The robot's outputs control the grasping and second operation.

Input / Output	The meaning of the program	Where it operates	What it does
X021	**Launch**	All positions (set from position 1)	Starts a cycle through positions when the cycle is not running; the robot first moves to position 1. During the cycle and when paused, the "skip" button is ignored.
X020	**Pause / Continue**	All positions (set from position 1)	The cycle continues – pressing **pauses** (the position within the cycle is not reset). The cycle is paused – pressing **continues** from the same position. The same input continues the cycle after the safety mechanism is triggered.
X025	**The part has arrived.**	Position 3	Before position 3, the robot stands still and waits for X025 to be turned on. Turning on X025 also disables the AI receiver (TCP).
X023 + X024	**Part in grasp** (two sensors)	Positions 5 and 6 (position 4 - rule is disabled)	Fall protection: while the robot is moving the part to positions 5 and 6, both sensors must see the part. If either sensor loses sight of the part, the stop axis is activated, the cycle pauses, and the user can continue with X020 or "▶ Continue".
Y020	**gripper** (vacuum / valve)	Position 4 - ON, Position 8 - OFF	Turns on in position 4 - the part is taken. Turns off in each box - the part is released, pause for 0.5 seconds.
Y021	**The second operation**	Position 6 – ON, Position 7 – OFF	It turns on when the robot reaches position 6, and turns off in position 7. What is connected to Y021 – according to the circuit diagram.

Algorithm for one cycle

Start X021 → Position 1	**1 → 2** original, approach	**Waiting for X025** part; detail; component She arrived	**3** focusing on details	**4** Taking Y020 ON	**5** elevation; rise; ascent; lift; promotion; increase; raise ⚠ X023+X024	**6** Operation Y021 ⚠	**7** Withdrawal Y021 OFF	**8** box Y020 OFF	**9** home = pose 1

1. The operator presses **X021**. If the robot is not in position 1, it first moves to position 1. Cycle: positions 1...9 × 6 times.
2. **Pose 1 → Pose 2** – from the starting position to the position where the measurement is taken.

3. Before reaching **position** 3, the robot waits for the signal "part arrived" (**X025**). Once it receives that signal, it proceeds to position 3 above the part.
4. **Step 4** – lowering onto the part with a smooth approach; arrived – **Y020 ON**, part gripped.
5. **Position 5** – Lifting with detail. While moving, **X023 and X024** must be lit (otherwise, the stop and pause functions will be activated).
6. **Position 6** – transfer to the second operation, fall protection remains the same; arrived – **Y021 ON**.
7. **Position 7** – disengagement; **Y021 OFF**.
8. **Position 8** – Box: Robot moves its cell 100 mm downwards (slowly 5), **Y020 OFF**, 0.5 s, then moves it 100 mm upwards (slowly 1).
9. **Position 9** – returning to the original (the same angles as position 1). A 200ms pause – the next cycle, the next cell.
10. After the 6th cycle, the box is full: "Complete: cycles – 6". A new start – again X021.

At any time, pressing **X020** will pause the cycle, and the next press of **X020** will resume from the same position.

▶ Video of the "simple-pod" program – will be added (link and screenshots from the video).

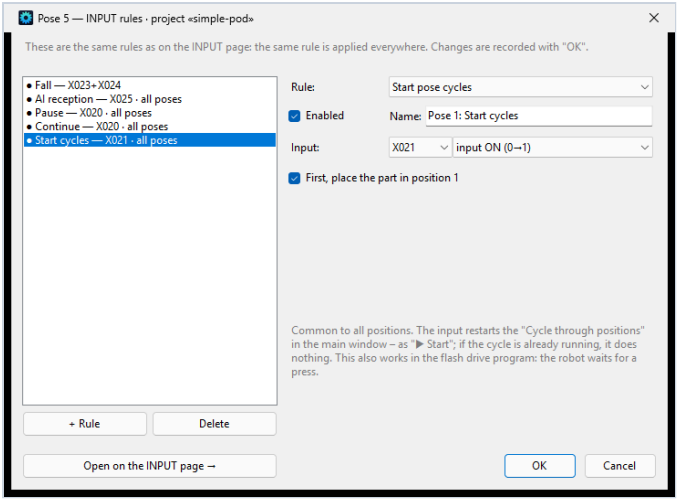
Positions in order

Position	J1	J2	J3	J4	J5	J6	Speed	What is configured
1	17,11	−47,24	50,35	1,16	−93,97	60,30	20 %	Initial; general rules: pause X020, continue X020, start X021
2	43,05	−31,51	24,88	1,42	−83,63	86,00	20 %	Approach to the picking location
3	51,22	−35,57	26,59	1,44	−81,07	94,10	20 %	on the detail; **waits for X025**; AI reception is disabled for X025
4	51,22	−35,16	23,17	1,45	−78,05	94,02	20 %	capture; smooth access 5; **Y020 ON**; capture protection - disabled
5	51,22	−36,22	30,42	1,42	−84,23	94,18	20 %	rise; ⚠ **X023+X024**, continue X020
6	27,59	−4,97	−11,15	1,30	−74,45	70,32	20 %	second operation; **Y021 ON**; ⚠ **X023+X024**, continue X020
7	16,96	−35,57	29,41	1,16	−84,65	59,94	20 %	withdrawal; **Y021 OFF**
8	−35,48	−68,12	31,82	0,06	−55,09	8,02	20 %	**3 x 2 = 6**; in each cell, **Y020 OFF**
9	17,11	−47,24	50,35	1,16	−93,97	60,30	20 %	Return to the initial state

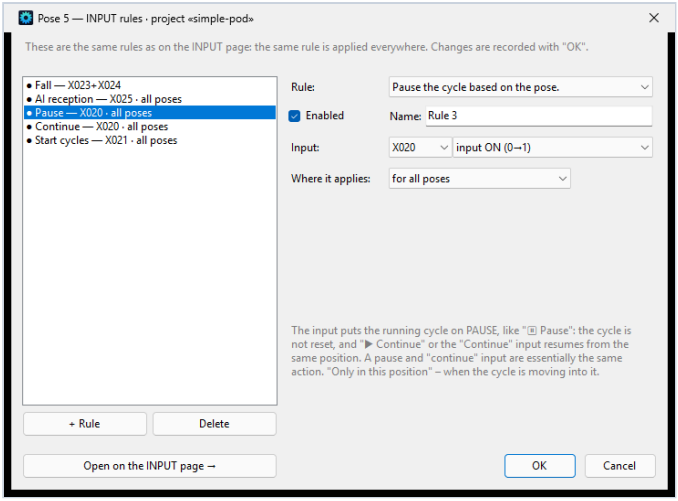
The pose angles are recorded as follows: move the robot (using the jog function or teach pendant) → **right-click on the pose button**; or, alternatively, select "read" → "read angles from the robot"; or, enter the numbers directly into the "edit" window.

Pose 1 – initial, start, pause, and "continue"

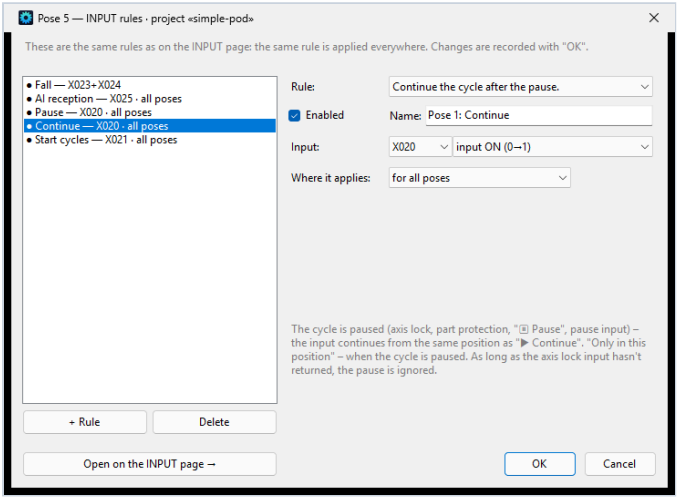
The robot is positioned here before deployment and returns to this position in pose 9. Three **general** rules ("all poses") are defined from window 1, so they are labeled on button 1: "pause X020, continue X020, start X021."



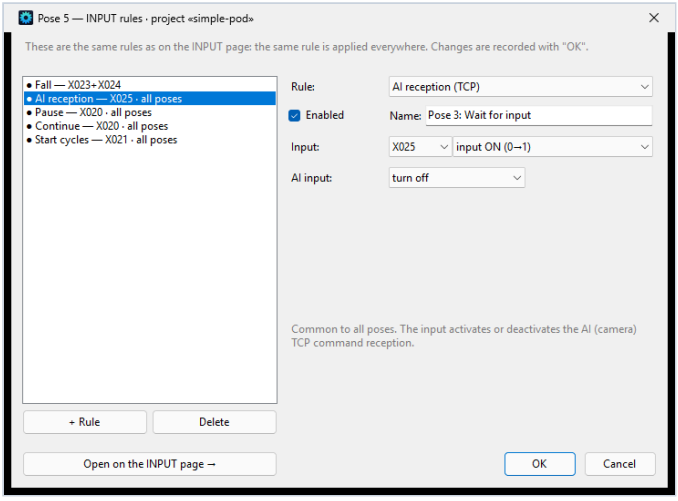
"Run cycles by pose": X021, input ON (0→1), "first to pose 1"



"Pause cycle by pose": X020, "for all poses"



"Continue cycle after pause": X020, "for all poses"



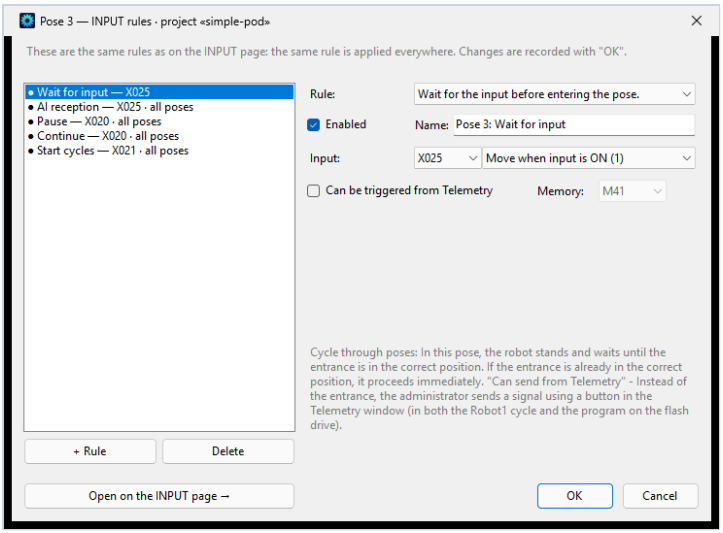
"AI reception (TCP)": X025 → turn off (rule "Pose 3: wait for input")

How to configure: "Position 1 → **INPUT...** → "+ Rule" → Select rule type from the "Rule" list → Input (X021 or X020) → Front "INPUT ON (0→1)" → "Applies to: All positions" → OK. The "Pause" and "Continue" buttons are on the same input (X020) – pressing once pauses, pressing again continues.

Pose 2 – approach

Intermediate pose: from initial position to the point of contact, so that the robot moves to the part in a clear and safe way, without touching the tooling. There are no inputs or outputs.

Pose 3 – above the part, waiting for "part arrived" (X025)



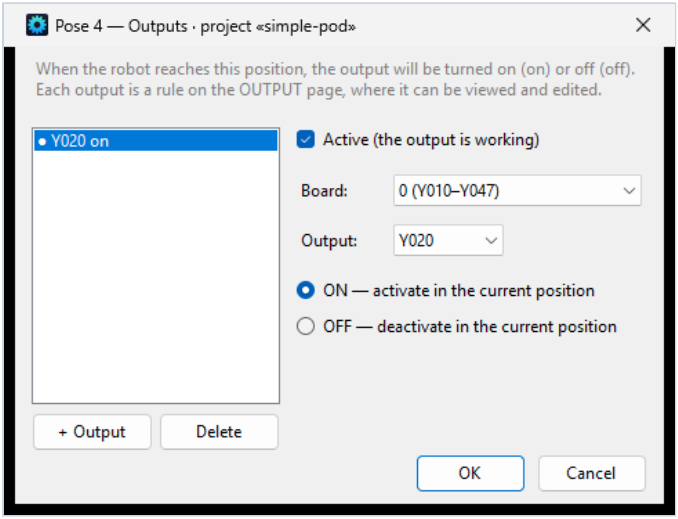
Pose 3 – "Wait for entry before pose" X025

Before moving to pose 3, the cycle checks for **X025**. If no signal is detected, the robot remains in pose 2, and the cycle displays "Pose 3 awaiting input X025". If a signal is detected, the robot proceeds to pose 3.

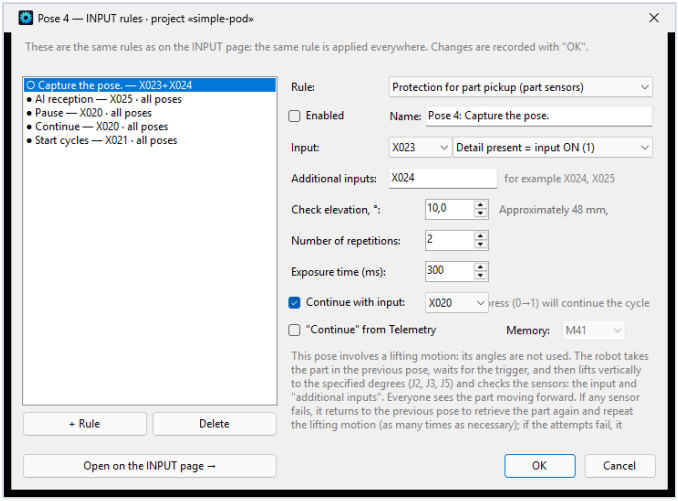
How to configure: "Set position 3 → INPUT... → 'Wait for input before position' → X025, 'Activate when input is ON (1)'". "Can be triggered via Telemetry" – trigger the signal using a button (via memory M41) instead of a sensor.

This rule has a setting called "**AI Function: Disable**". When input X025 is activated, the TCP command reception from the camera is disabled. The button label reads: "Waiting for X025, AI function X025".

Pose 4 – Capture, Y020 ON



Pose 4 – OUTPUT: Y020, ON – Activate in pose



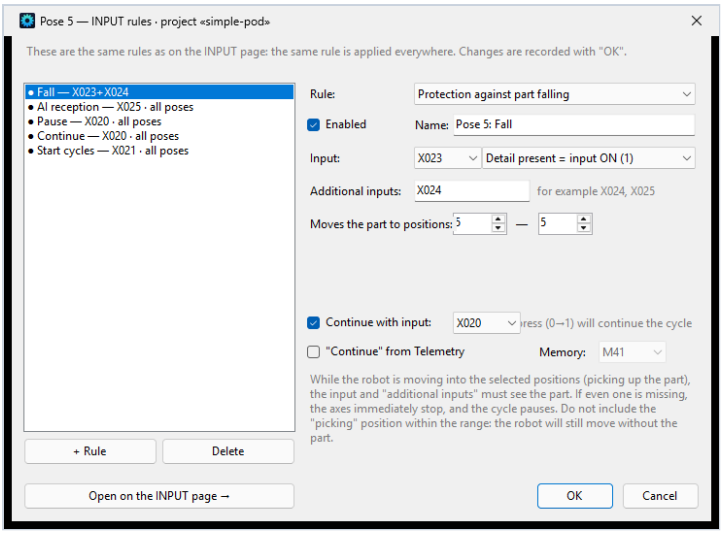
Pose 4 – "Capture protection" X023+X024 (○ Off)

The robot gently lowers itself onto the part with a **smooth approach** (slow end of travel). It has reached its position (all axes within 0.3 degrees) – **Y020 ON**: the gripper is holding the part. The "Y020 on" button is activated.

How configured: "Pose 4 → 'Smooth Entry' 5; OUTPUT... → '+ Exit' → payment 0, Y020, 'ON - activate in pose 4'. On the OUTPUT page, this rule 'Pose 4: Y020 on' with the angles of pose 4 and a tolerance of ±0.3° is configured.

> **The "Pose 4: Grab" protection is currently disabled (○).** Parameters: X023 + X024 sensors, 10° lift (≈ 48 mm), 2 repetitions, 300ms exposure, continue from X020. With this protection, **the lift itself becomes the verification process** (its angles are not used). Therefore, if you enable it, place it in the **after** grab position – specifically, in position 5, not position 4: the robot will grab the part in position 4, lift it by 10°, check X023 and X024, and if it fails, return to position 4 and try again.

Pose 5 – Lift with part, fall protection

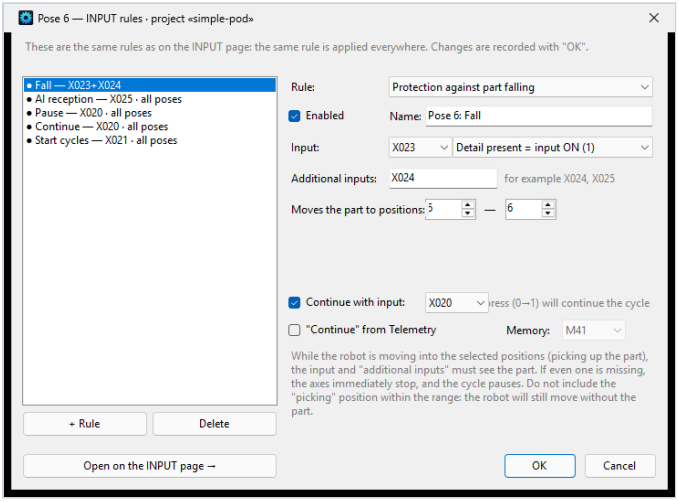


Pose 5 – "Part fall protection"

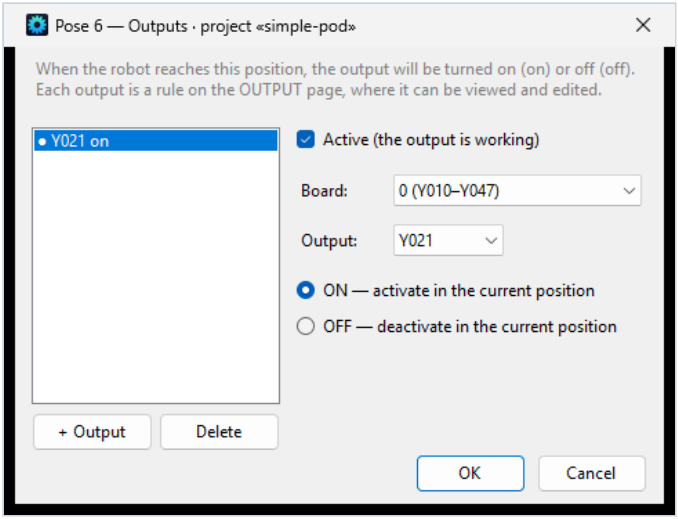
While the robot is moving the part to position 5, both sensors **X023** and **X024** should be able to see the part. If either one fails, the system immediately stops, and the log displays "⚠ PART LOST on the way to position 5 - no signal from X024". The operator removes the part, checks the suction cup, and presses **X020** (or "▶ Continue").

How to configure: "Position 5 → INPUT... → 'Part Drop Protection' → Input X023, 'Part present = INPUT ON (1)', 'Additional inputs: X024', 'Part moves from position 5 to 5', 'Continue with input: X020'."

Pose 6 – Second operation, Y021 ON



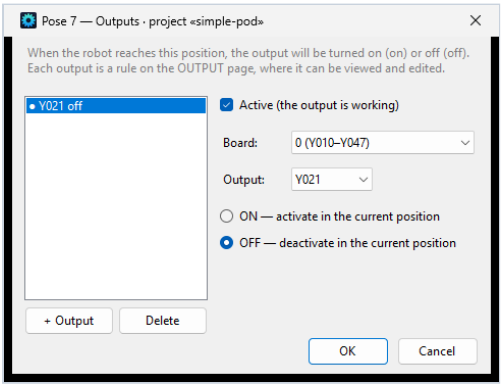
Pose 6 – Fall protection X023+X024, from 6 to 6, continue X020



Position 6 – OUTPUT: Y021 ON

Moving the component to the second operation – the fall protection mechanism works the same way as in position 5. Arrived – **Y021 ON**.

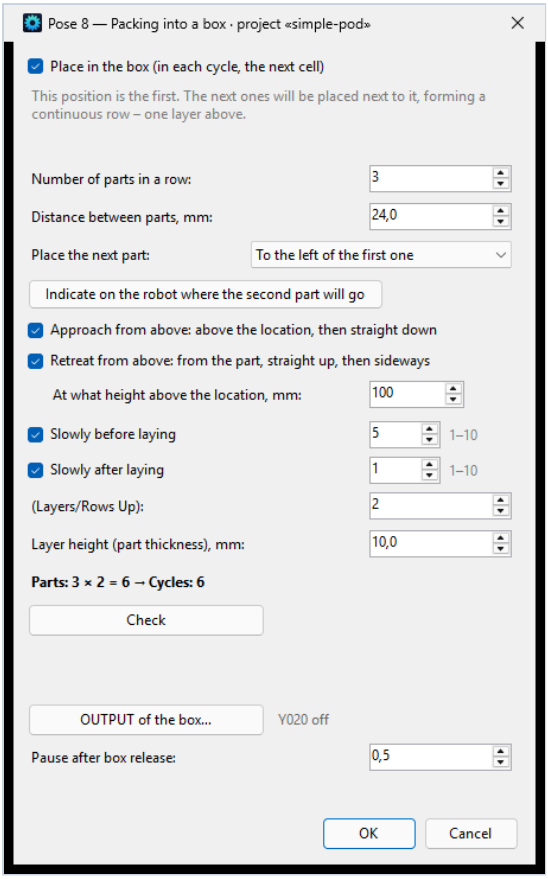
Position 7 – Removal, Y021 OFF



Position 7 – OUTPUT: Y021, OFF

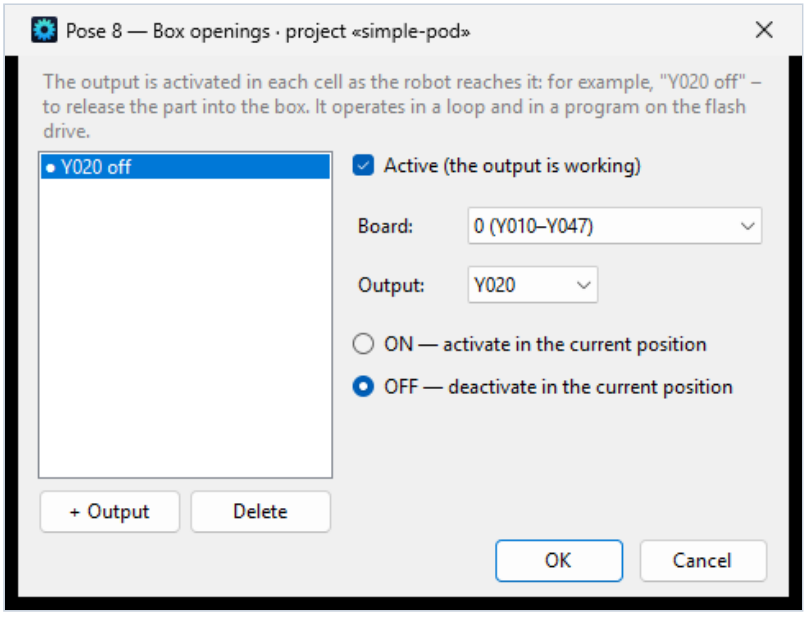
The robot is moving away from operation 2; it arrives at **Y021 OFF**. On the OUTPUT page, this rule "Posture 7: Y021 off" (condition: DISABLE corners of posture 7, ±0.3°) is defined. There are no fall protection measures on the path to postures 7 and 8 – if desired, expand "Transport part to postures" to 5...8.

Position 8 – Placement in 3 x 2 box



Position 8 – Box parameters

Parameter	Meaning
Place in box	Yes
Number of parts in a row	3
Distance between parts	24 mm
Place the next component	To the left of the first one
From above	Yes
Moving upwards	Yes
Height above the surface	100 mm
Slowly before laying	5
Slowly after laying	1
Layers (rows upwards)	2
Height of the layer (thickness of the component)	10 mm
Components / cycles	3 × 2 = 6
Output box	Y020 off
Pause after exit	0.5 seconds



Output box: Y020, OFF – in each cell

Where each part will go

Cycle	Layer	Position in the row	Offset from position 8
1	1 (bottom)	1	0 – refers to posture 8
2	1	2	24 mm to the left
3	1	3	48 mm to the left
4	2 (upper)	1	10 mm up
5	2	2	24 mm to the left, 10 mm up
6	2	3	48 mm to the left, 10 mm up

In each cell: the robot moves 100 mm above the starting point → straight down, the end of the path moves slowly (5) → **Y020 OFF**, the part is released → 0.5 seconds → straight up 100 mm, the beginning moves slowly (1) → position 9.

How to set it up: Place the part in the position of the first part (bottom layer, first position), enter posture 8 (right-click → "Edit" → "Place in box..." → set parameters as in the table → "Show on robot where the second will go" (check the side) → "Verify" → "OUTPUT box..." → "+ Output" → Y020, OFF → OK.

Pose 9 – return to the original position

The same angles as Pose 1. After Pose 9, a 200 ms pause, followed by the next cycle with a new box cell.

If the cycle has stopped

What is visible	The reason	What to do
"Position 3 awaiting input X025"	"No signal: part arrived"	"Place part / check X025 sensor (light on INPUT page)"
"Pause" after pressing X020	Operator initiated pause	Press X020 again or "▶ Continue"
"⚠ PART LOST en route to position 5 (6)"	X023 or X024 missing – part fell or moved	Remove part, check suction and sensors → Press X020 or "▶ Continue"
"Cycle through positions not completed – press skipped"	X021 was pressed during the cycle or pause	Wait for the cycle to complete or "■ Stop", then X021
"accident... - cycle paused"	controller malfunction	eliminate the cause → "Emergency Stop" → "Start - Mode 7" → "▶ Continue"
"NOT READY: Mode 2"	remote control program not running	"Start - Mode 7"

"simple-pod" program on a USB drive

The "Save to USB..." button records the same cycle using the teach pendant program. The following data is entered into the program: positions and speeds, Y020/Y021 in positions, the 3x2 box according to the part counter, the X025 wait before position 3, the X023 + X024 check upon arrival in positions 5 and 6, and the X021 entry (initially into position 1). **This data is not included** (it only works in the Robot1 cycle): the pause and "continue" function of X020, and the AI-controlled function. For the teach pendant: "IMP USB" → Load → Auto → Start → The robot waits for the X021 entry. To observe and start from a specific part, use the Telemetry window.