



ION8-1324 Programming Guide

Web interface, PLC program, PID, wireless mirroring and software update

Contents

- 1. Ways to access the web interface
- 2. Using the web interface
- 3. PLC programming: methods and instruction list
- 4. Visual chain (block) editor
- 5. Ladder editor
- 6. Using the three methods together, with examples
- 7. Using the PID blocks
- 8. Troubleshooting

1. Ways to access the web interface

All settings and programs of the ION8-1324 are managed through the device's own web interface. No extra software or drivers are needed; a browser on a computer, tablet or phone is enough.

Note: The web interface is currently shown in Turkish. This guide gives each on-screen label in quotation marks next to its English meaning.

1.1 Connecting to the device's own wireless network

When first powered on, the device broadcasts its own wireless access point.

Setting	Value
Network name (SSID)	ION8_1324_XXXXXX (XXXXXX = 6 characters unique to the device, printed on the label)
Password (default)	Supplied with the device
Address	http://192.168.4.1
Name-based address (default)	http://ion8-1324.local (when connected through a network)

1. Select ION8_1324_XXXXXX in the wireless network list and enter the password.
2. Type http://192.168.4.1 in the browser.
3. Choose a page from the menu on the left (DI, DO, AI, AO, PLC Programı, PID, Device Info).

Important: The default password is publicly known. After installation, change the access point password in **Device Info → WiFi Settings** (8–63 characters). The change takes effect after the device restarts.

1.2 Joining your existing wireless network

The **WiFi Settings** section of **Device Info** offers two modes.

WiFi Mode	Meaning
AP Only	Only the device's own network is broadcast.
AP + STA	In addition to its own network, the device joins the existing network entered in Network SSID and Network Password . Only 2.4 GHz networks are supported. Once connected, the address obtained is shown in STA IP .

- **mDNS Hostname:** the name used to reach the device as name.local. Lowercase letters, digits and hyphens (up to 32 characters, no hyphen at the start or end). If left empty, the default ion8-1324 is used.
- **AP Yayın Süresi (saniye)** (access point broadcast time, seconds): the device's own network stays on for this long and switches off if no client connects. **0 = never switches off** (default).
- Wireless setting changes take effect after the device restarts. The reason for a failed connection (e.g. network not found, wrong password) is shown in the **Disc Reason** row.

2. Using the web interface

This chapter describes the interface layout and the settings other than PLC and PID: mirroring and wireless mode, input type, software update and factory reset.

2.1 General layout

- **Top bar:** live connection status and the device build signature (date-time in YYYY.MM.DD.HH:MM.SS form). If the device powers off or the link drops, the bar shows "Bağlantı Yok" (No connection); stale values are not presented as live. Quote this signature in support requests.
- **Left menu:** the DI, DO, AI and AO pages show live input/output values. **PLC Programı** (PLC Program) holds instruction-list programs, **PID** the independent PID loops, and **Device Info** the network, wireless mirroring and software update settings.

2.2 Remote I/O mirroring and wireless mode

Two ION8-1324 units can be linked wirelessly. Each unit then sends its own digital and analog inputs to its peer and applies the digital and analog output commands coming from the peer (two-way **mirroring**). Inputs and outputs are shared between two panels without running cable.

2.2.1 Pairing

The settings are in the wireless mirroring section of the **Device Info** page.

1. Open the web interface of both units. Select **P2P Master** on one and **P2P Slave** on the other. Master/Slave only decides which unit establishes the link; data flows both ways. When the mode changes, the wireless and I/O side restarts briefly; the web interface stays open.
2. On the Slave unit press **Eşleştir** (Pair); the pairing window opens.
3. On the Master unit press **Eşleştir**, choose the Slave's address from the list and connect.
4. Once linked, a signal-strength bar and value (dBm) appear. **Eşi Sil** (Delete peer) removes the pairing; **Yeniden Eşleştir** (Re-pair) pairs with a new peer.

The third option, **BRIDGE**, is for connecting the device to a network gateway and is not selected for a two-unit mirror; the factory mode is gateway, and for unit-to-unit mirroring the units are set to Master/Slave mode.

The default mapping is one-to-one: the **DI0...DI7** inputs of one unit go to **DO16...DO23** on the partner unit, and **AI4...AI7** go to **AO12...AO15** on the partner unit. The mapping can be changed on the receiving unit and a channel can be switched off. Outputs used by PWM, step/direction or the PLC program are excluded from mirroring automatically once the function is enabled. Counter pulses are regenerated as pulses on the partner unit.

If the link is lost: when the Safe State Time (factory value 1.5 s) expires, the digital and analog outputs mirrored from the partner unit go to 0. When the partner unit runs again, the link returns by itself and the mirrored values are applied again (typically 5–10 s).

2.2.2 Wireless link mode

The **Master** unit decides the mode; the selection applies only while that unit is Master. Each mode offers a different balance between range and delay. **Mesafe** (Range) is the default.

Mode	Typical delay	Advantage	Disadvantage
Mesafe Range (default)	~33 ms	Longest range, most resistant to noise. Holds the link best over long distances, behind panel sheet metal and in electrically noisy sites.	Highest delay; fast-signal mirroring is typically ~4–6 ms later than other modes.
Dengeli Balanced	~29 ms	Range close to Range mode, lower delay.	May drop earlier than Range mode in very noisy or distant installations.

Hız Fast	~28 ms	Low, stable delay; suited to nearby units in the same hall.	Range is noticeably shorter; link quality falls with obstacles and noise.
Çok hızlı Very fast	~27 ms	Lowest delay.	Shortest range; only for units in the same panel or very close.

Delay: time from an input change on one unit to the corresponding output change on the other (typical, including the 10 ms input filter). Actual values depend on the installation environment.

The **Güncelleme aralığı** (update interval) list offers 7.5 / 10 / 15 / 20 / 30 ms. A larger interval increases delay (at 30 ms typically ~38–48 ms). Recommended interval: 10 ms for Range, 7.5 ms for the other modes. Save the selection with **Kaydet ve uygula** (Save and apply). The **Şu anki bağlantı** (Current link) row shows the mode and interval actually in use.

Which mode? If the units are in different panels or far apart, or in an electrically noisy environment, keep **Range** (default). If they are close in the same hall and delay matters, try **Fast**; inside the same panel, try **Very fast**. Observe link quality in the field with the faster modes.

2.2.3 Routing: which output a peer's input writes to

In the **Giriş Tipi ve Aktarım** (Input type and routing) table in the same section, two choices are made for each input (DIO...DI7):

- **This unit's input type:** how this input is sent to the peer (Section 2.3).
- **Peer DI → this unit's output:** which digital output (DO16...DO23) of this unit the peer's input is written to; **Aktarılmaz** (not routed) can also be chosen. An output can be assigned to only one input. Because a conflict is known where the output is located, this choice is made on the **receiving** unit.

Mirroring does not write to these outputs and the table shows a warning: an output used by the PLC program, an enabled PWM output, a PWM direction pin, an output driving motion direction. The counters in the table (sent, received from peer, blocked, dropped, etc.) help monitor the pairing.

2.3 Pulse counting and input type

For each mirrored digital input the **input type** has two options.

Input type	Behavior
Mechanical contact / relay – low speed (default)	Only the level is mirrored. Contact bounce is suppressed by a 10 ms stable-level filter; a state is not passed on until it has been unchanged for at least 10 ms.
Pulse-counting sensor (UI: Yüksek hızlı sensör – pulse sayımı)	The number of pulses on the input is sent to the peer; the peer completes, at its output, any pulses lost to communication delay. Counts such as length measurement therefore do not lose pulses.

Caution: If the pulse-counting type ("Yüksek hızlı sensör") is selected for a mechanical contact, contact bounce produces false counts. Use this type only with clean-output sensors (e.g. proximity sensor, encoder). The pulse-counting mode requires current software on both units.

With pulse completion, pulse counts mirrored from the remote unit are delivered complete at the peer output. The local pulse counter input is monitored on the **Counter 0** page (DI4 pulse input; counter value, scale, scaled value, frequency and filtered frequency).

2.4 Wireless software update

The device software is updated wirelessly from the web interface; no cable, programmer or extra software is needed. The update renews all of the device's software in a single step.

2.4.1 Steps

1. Save the update package you received from Ionia (a file with the `.ion8ota` extension) on your computer.
2. Connect the computer to the device's network (Chapter 1) and go to the update (OTA) section of the **Device Info** page.
3. Choose the package file; its name and size are shown.
4. Press **Yükle ve Güncelle** (Load and Update). The progress bar shows the upload and transfer stages as a percentage; the update typically takes about 80 seconds.
5. When "Tamamlandı" (Completed) appears the device restarts; wait until it has restarted, then reopen the page and confirm that the build signature in the top bar has changed.

Caution: During the update do not cut power to the device or close the browser page. When the update starts, mirroring stops; digital outputs go to 0, analog outputs to 0 V, and PID and PLC outputs go to their safe values. Factory reset and program changes are rejected during the update. Therefore update while the machine is stopped; update two mirrored units one after the other, separately, and check afterwards that the pairing has been re-established.

The package is verified on the device; a corrupt or unsuitable file is rejected and a "Reddedildi" (Rejected) message appears. A rejected package does not affect the device's current software.

2.4.2 Factory reset

There are two ways: press the middle button and hold it for **10 seconds** while powering the device, or use the **Factory Reset** button on the **Device Info** page of the web interface. The operation is rejected while a software update, motion or PID is running.

Scope	
Returns to default	Wireless settings (including the access point), pairing and operating mode, timeouts, input filter, company and note fields.
Kept	Calibration, analog input/output offset and scale values, power-up counter, installed software version.

After the operation, set up the operating mode, pairing and program again.

3. PLC programming: methods and instruction list

On the ION8-1324 the PLC program can be written in three ways. All three describe the same program and you can switch between them at any time. The device compiles the program and runs it by scanning repeatedly.

Method	How you work	When it fits
Instruction list (IL)	Text is written line by line (LD I0, = 017 ...). It is on the Program tab of the PLC Program page (Chapter 3).	Short logic, quick changes, copy-paste, documenting a program.
Visual chain (blocks)	Blocks are placed on a canvas and wires are drawn from output to input (Chapter 4).	Data flow, analog processing, scaling, ramp and timing chains.
Ladder	Drawn like a relay diagram with rungs, contacts and coils (Chapter 5).	Users used to relay logic; seal-in and interlock circuits.

This chapter covers the instruction list, common naming and run rules. The visual methods are in Chapters 4 and 5; using the three together is in Chapter 6.

3.1 Input and output naming

In programs and PID settings every input/output is referred to by a short **identifier**. Naming follows the Allen-Bradley style: inputs start with **I**, outputs with **O**. Identifiers are written in capitals.

Identifier	What it is	Note
I0 ... I7	Digital inputs	Same as DIO...DI7 on screen (DI3 is I3 in a program). Read only.
O16 ... O23	Digital outputs	Same as DO16...DO23 on screen (DO17 is O17). Read and write.
AI4 ... AI7	Analog inputs	AI4 and AI4.raw: raw value; AI4.scaled: scaled value; AI4.filt: filtered value. Read only.
A012 ... A015	Analog outputs	Writable. Range 0 ... 10000 mV (0–10 V). If the link to the paired unit is lost, the mirrored analog output goes to 0 V when the Safe State Time (factory value 1.5 s) expires.
B0 ... B255	Boolean (on/off) internal memory	For intermediate results. Old notation: M.
F0 ... F255	Decimal number memory	Old notation: D.
N0 ... N255	16-bit integer memory	A bit is read/written as N5.3 (bits 0...15).
T1, T1.q, T1.et	Timer	T1 and T1.q: output state; T1.et: elapsed time.
C0, C0.q, C0.cv	Counter	C0 and C0.q: output state; C0.cv: counter value.
PID0.u, .pv, .sp	PID loop values	PID output, measurement and setpoint; read from the program (see Chapter 7).

Note: Older programs may use X for digital inputs and Y for digital outputs (X0, Y16); this notation is still accepted. Use I and O in new programs.

Tags. In the **Etiketler** (Tags) tab of the PLC page you can give an identifier a meaningful name, a unit and a display scale, e.g. I3=WireBreak or AI4=Pressure,bar,0:10000:0:16 (raw range 0...10000 shown as 0...16 bar). Tags only affect what is displayed, not how the program runs. A name cannot contain spaces or ; = , ! # @ and cannot start with a digit.

Output ownership. Outputs reserved for motion (PWM) cannot be written from a program. A program that writes to DO16...DO19 assigned to PWM, or to DO20...DO23 selected as direction pins, gets a **channel conflict** error; change the PWM setting or use other outputs. Outputs managed by a running PLC program cannot be written by hand from the DO page (the Live tab shows the label "PLC").

3.2 Page sections

Tab	Function
Canlı (Live)	Live state of inputs, outputs and internal values. DI/DO refresh 5 times per second, others about once per second. Values can be written, with confirmation, only to outputs that no program or motion function owns. Very short pulses show as "darbe x n" (pulse x n). Quality indicator: blank = good; bad, stale or forced values are flagged.
Program	Numbered-line editor and the buttons Denetle (Check), Yükle (Load), Çalıştır (Run), Durdur (Stop), Karttakini getir (Fetch from device), Örnek ekle (Add example), plus a command cheat sheet.
Sürümler (Versions)	Every program loaded is kept as a version (latest 8 by default). The difference between two versions can be viewed; Bu sürüme dön (Return to this version) loads an old version as a new version.
Etiketler (Tags)	Names, units and display scales for identifiers (Section 3.1).
Uzman (Expert)	Detailed information for diagnostics and support.

The **Visual chain**  button on the Program tab opens the same program in the block and ladder editors (Chapters 4 and 5).

The top bar shows the program kind (none / default / user), state (**Çalışıyor** Running, **Durdu** Stopped, **Boşta** Idle), version number, node count, any error, and write authority (**Yetkili** Authorized / **Salt okunur** Read only). While read only, programs cannot be loaded or changed.

3.3 Writing rules

- Commands are separated by line breaks or `;`. A comment starts with `//` and runs to the end of the line. `;` does not start a comment.
- Commands are written in capitals.** A lowercase `ld` gives a "command not found" error.
- A program may be at most 4096 bytes (comments included). Upper limits: 96 nodes, 64 symbols.
- Each command works on an **accumulator** (a single intermediate value) and a stack 8 levels deep. There are no jumps or labels; the program runs once from top to bottom on every scan.
- Boolean (on/off) and numeric values cannot be mixed: `LD I0` is boolean, `LD AI4.scaled` is numeric; `GT` cannot be applied to a boolean.
- An output can be written with `=` only once. A second `=` to the same output is a "double coil" error. Several `SET`/`RST` for the same output are allowed; `RST` takes priority.
- Reading an internal memory or timer that is never written anywhere gives a "not written" error.

3.4 Command summary

Command	Description
<code>LD</code> , <code>LDI</code> , <code>LDP</code>	Load the value; load its inverse; load the rising edge.
<code>AND</code> , <code>ANI</code> , <code>ANDP</code>	Series (AND); AND with the inverse; AND with the rising edge.
<code>OR</code> , <code>ORI</code> , <code>ORP</code>	Parallel (OR); OR with the inverse; OR with the rising edge.
<code>ANB</code> , <code>ORB</code> , <code>NOT</code>	Block AND, block OR, invert.
<code>=</code> (or <code>OUT</code>)	Write the result to an output or internal memory.

SET, RST	Set the output to 1 and hold; set it to 0. RST C0 resets a counter.
TON, TOF, TP	Timer: TON T1 500. The time is in milliseconds, must be a fixed number, 1 ... 86 400 000 ms (24 hours).
CTU, CTD, CTUD	Counter: CTU C0 3; CTUD C2 2 I2 (the second boolean input sets the count direction).
ADD, SUB, MUL, DIV	Arithmetic. Division by zero gives an invalid result; the analog output goes to its safe value.
GT, LT, GE, LE, EQ, NE	Comparison (>, <, ≥, ≤, =, ≠); the result is boolean.
BLK	Calls a ready-made function block (Section 3.8).

3.5 Example programs

The programs below are ready examples. Replace the example 0 outputs with those of your installation (see the output ownership note in Section 3.1).

Start-stop (latching)

```
// Start-stop latching
LD I0
OR O17
ANI I1
= O17
```

When I0 is pressed O17 turns on and holds itself even after I0 is released. I1 (stop) drops it. If I0 and I1 arrive at the same time, stop wins. The output itself (OR O17) is read as the holding contact: an output read **before** the command that writes it returns its value from the previous scan.

Delayed switch-on

```
LD I3
ANI I6
TON T1 500
= O21
```

While I3 is active and I6 is not, O21 turns on after 500 ms; if I3 drops or I6 arrives, O21 drops immediately. The timer starts from the scan in which the input is first seen.

Counter

```
LD I0
CTU C0 3
= O18
LD I1
RST C0
LD C0.cv
= AO12
```

Each rising edge of I0 increments the counter; on the third pulse O18 turns on. I1 resets the counter. The counter value is written to AO12 (mV). CTD counts down; its Q output turns on when the counter value is less than or equal to the preset (the preset is given as a negative number).

SET / RST

```
LD I2
SET O22
LD I3
RST O22
```

When I2 arrives O22 turns on and stays on after I2 is gone; I3 drops it. If both arrive together, **RST** wins.

Analog threshold and analog output

```
LD AI4.scaled
GT 2000
= 019
```

```
LD AI4.scaled
DIV 2
= A012
```

The first turns O19 on when the scaled value of AI4 exceeds 2000 (the threshold is written in AI4's scaled unit). The second writes half of the AI4 value to AO12.

3.6 Timer and counter details

- **TON**: the output turns on when the input has been continuously active for the set time. **TOF**: after the input drops the output is held for the set time; if the input was never active the output is 0. **TP**: produces a fixed-length pulse on the edge.
- **T1** (or **T1.q**) is the output state, **T1.et** the elapsed time. The Live tab also shows **.pt** (preset) and **.rt** (remaining).
- **CTU**: Q becomes 1 when the counter value is greater than or equal to the preset; the counter keeps counting after reaching the preset. **CTUD** counts both ways (the second boolean input counts down). **RST C0** resets the counter.

3.7 Error messages

The **Denetle** (Check) button, or **Yükle** (Load), compiles the program on the device. On error the line is flagged; the message has the form **command number, symbol, reason**. Examples (the reason text appears in Turkish on screen):

Message	Cause and remedy
1 ld komut yok	Command written in lowercase. Write LD .
4 016 çift bobin	Second = to the same output (double coil). Use a single = , combine conditions with OR , or use SET / RST .
"yazılmadı" (not written)	The internal memory or timer being read is never written. Add the command that writes it first.
süre en az 1 ms, en çok 24 saat (time at least 1 ms, at most 24 h)	The time must be a fixed number between 1 ms and 24 hours (86 400 000 ms).
... sayısal bekleniyor (... numeric expected)	Boolean and numeric values mixed (e.g. GT on a boolean).

3.8 Function blocks (BLK)

Functions without a dedicated command are called as **BLK FUNCTION instance name=value ...** (52 blocks; the accumulator, if set, is the first input; blocks can be chained). **PID** is a separate service, not a block (Chapter 7); a program can read **PID0.u**, **PID0.pv** and **PID0.sp**.

Example: AI4 0...4000 scaled linearly to AO12 0...10000 mV (0–10 V); input 2000 gives 5000 mV. All blocks are on the **cheat sheet** (Program tab).

```
LD AI4.scaled
BLK SCALE B1 in_min=0 in_max=4000 out_min=0 out_max=10000
= A012
```

3.9 Running, stopping and persistence

3.9.1 Loading and running

1. On PLC Programı → **Program**, write the program or insert a sample with **Örnek ekle**.
2. Use **Denetle** to check for errors.
3. Press **Yükle**. If a program is running, the interface shows the line differences between the active and the new program and asks for confirmation.
4. The loaded program **starts running automatically**.

Caution: When a program is loaded or run, timers, counters and latched outputs are reset. Load at a moment when the machine is not affected by this.

- **Çalıştır** and **Durdur** start and stop the program. A program cannot be run if the device has none.
- If you load or delete a program, or return to an older version, while a PID loop is running, the interface asks whether the PID should be stopped; if you confirm, the PID is stopped and the action is retried once.
- **Karttakini getir** reads the program stored on the device into the editor.

3.9.2 Persistence of Stop

The program keeps its last run/stop state. If power is cut and restored while the program is running, the program starts again by itself. Counter values inside the program are reset by a power interruption; do not rely on the program to keep a count that must survive power loss. **Durdur** is persistent: if power is cut and restored on a stopped device, the program starts up **stopped**; press **Çalıştır** to start it again.

Saved PID settings are likewise kept at power-up and the PID loop runs by itself (Section 7.3).

The loaded program and its version history are stored permanently in the device; a power cut does not erase them.

3.9.3 Motion (traverse) chain

A ready motion chain is provided; its settings are on the separate **TRV** and **SHZ** pages of the web interface (not on the PLC Program page). It is intended for applications such as the edge reversal (traverse) used in winding machines. Check the settings against your own machine before production use.

4. Visual chain (block) editor

In the visual chain editor the program is built from **blocks** joined by **wires**, without typing commands. The same program is shown both as blocks and as an instruction list; the two always describe one program.

4.1 Opening the editor and the screen layout

The **Visual chain**  button on the PLC Program page opens the editor. The top bar holds the program selector, **Load example...**, **Validate**, **Upload...**, **Read**, and the **Live I/O**, **Embed layout in IL** and **Simulation** options. The instruction-list text is on the left, the block canvas in the middle, and the **Properties** and **Blocks** tabs on the right. The **Block / Ladder** mode selector sits below the canvas.

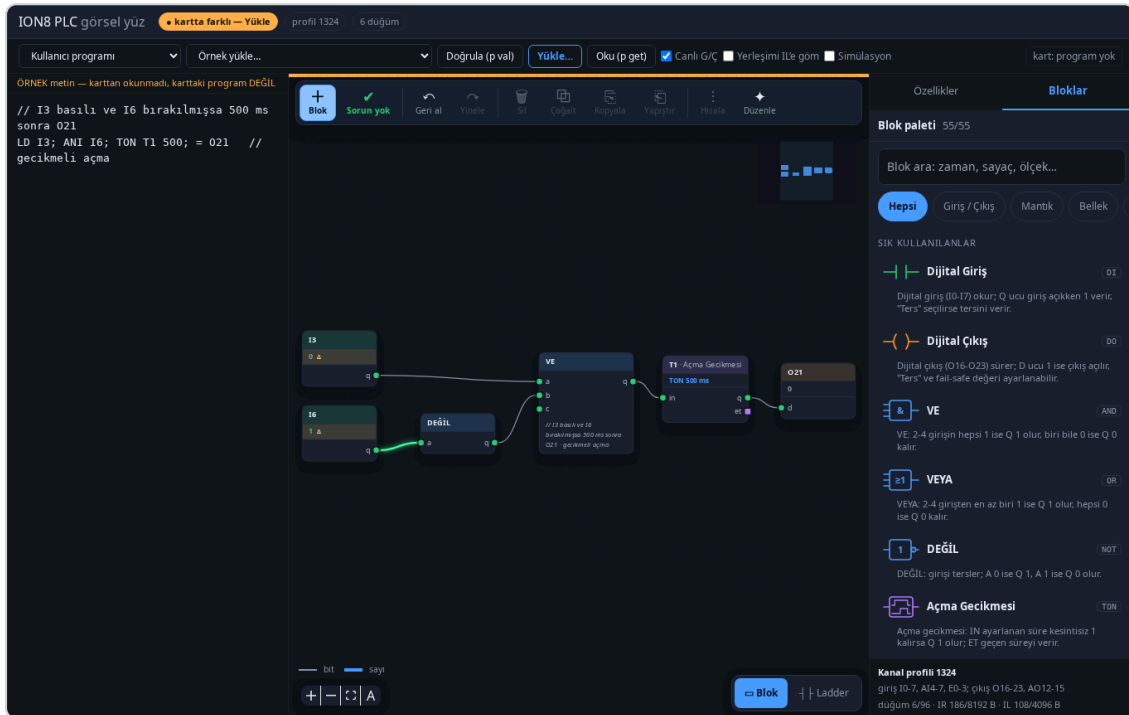


Figure 4.1 Visual chain editor: "if I3 is pressed and I6 released, O21 after 500 ms" (input → AND → on-delay → output). Screen text is shown in the device language.

Item	Function
Status badges	"program differs on device": the program on screen differs from the one running on the device. "profile 1324": channel profile. "N nodes": number of blocks in the program.
Counter strip	Bottom right: nodes (max. 96), IR (8192 bytes) and IL (4096 bytes) usage; watch it to see when a limit is near.
Channel profile	Channels available on this model: inputs I0–7, AI4–7, E0–3; outputs O16–23, AO12–15.
Toolbar	Block (add), status indicator (No issues / error count), Undo , Redo , Delete , Duplicate , Copy , Paste , Align , Arrange (automatic layout).
Canvas tools	Bottom left: zoom in / out / fit / readable scale; top right: mini map.

4.2 Adding blocks: palette and search

The **Blocks** tab on the right groups 55 blocks into categories: **Input/Output**, **Logic**, **Memory**, **Timing** and others. Type part of a block name in the search box to narrow the list; frequently used blocks are collected under **Favorites**. Drag a block onto the canvas or add it by touch; the **Block** button on the toolbar opens the same palette.

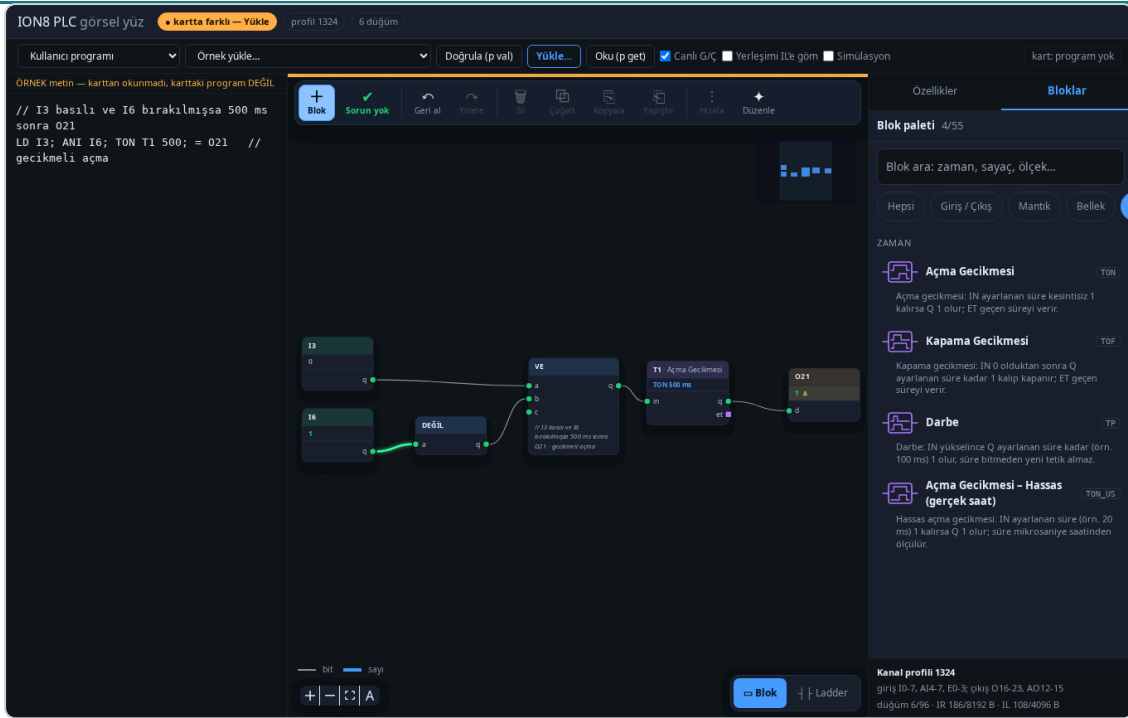


Figure 4.2 Blocks tab: category buttons and timing blocks (on-delay / off-delay, pulse, counters).

4.3 Connecting wires

Draw a wire from a block's output point to another block's input point. Wires are coloured by data type: **grey = bit** (0/1), **blue = number**. A connection of mismatched types is not accepted; to feed a bit into a number input, use a converter or comparator block. With **Live I/O** on, wire values are shown live: an energised wire is bright green.

To free an input, select the block and press **Disconnect** next to that input in the **Properties** panel. The input is then empty and the toolbar indicator reports the error count.

4.4 Properties panel

Click a block and the **Properties** tab shows its details:

- **Object explorer:** inputs (I) and outputs (O) used in the program; click one to select the related block.
- **Identity:** the block type and, for timer/counter/memory blocks, the **Name**, i.e. the internal memory identifier (e.g. T1).
- **Settings:** block-specific parameters (e.g. **Time** and unit, preset, threshold).
- **Inputs:** the source of each input and **Disconnect**. Wiring changes take effect immediately.

Setting changes are committed with **Apply** and discarded with **Cancel**. **Duplicate**, **Copy** and **Delete** at the bottom of the panel act on the block.

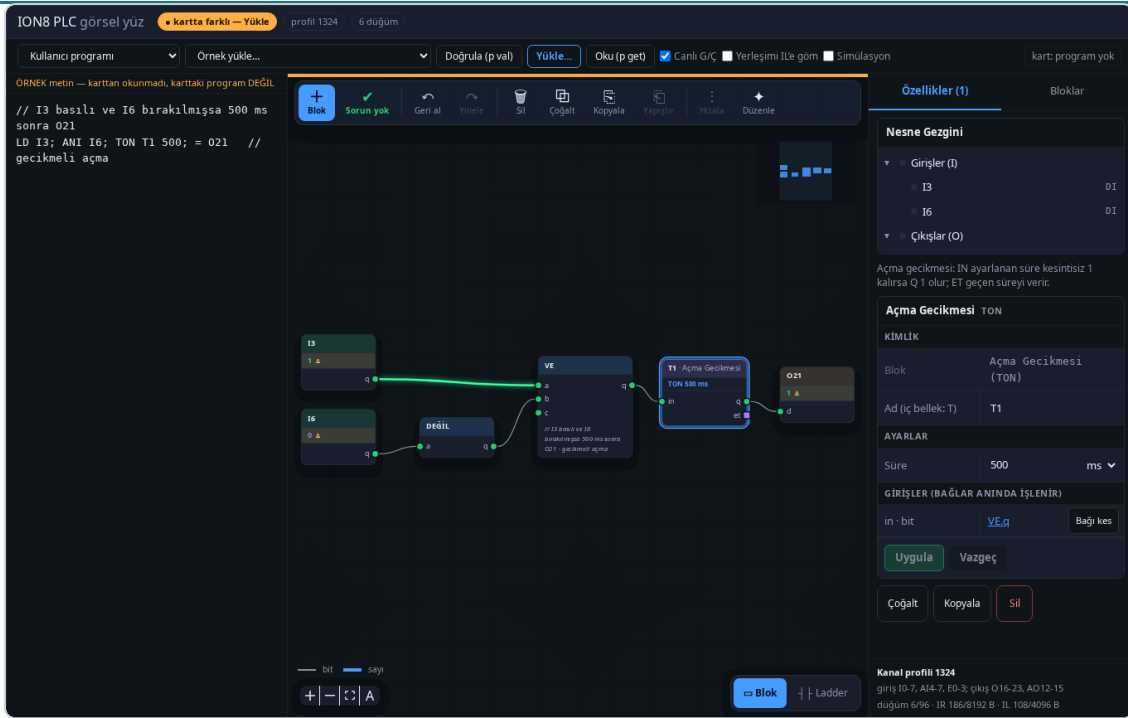


Figure 4.3 Selected on-delay block (T1, 500 ms) and the Properties panel: identity, setting, input connection.

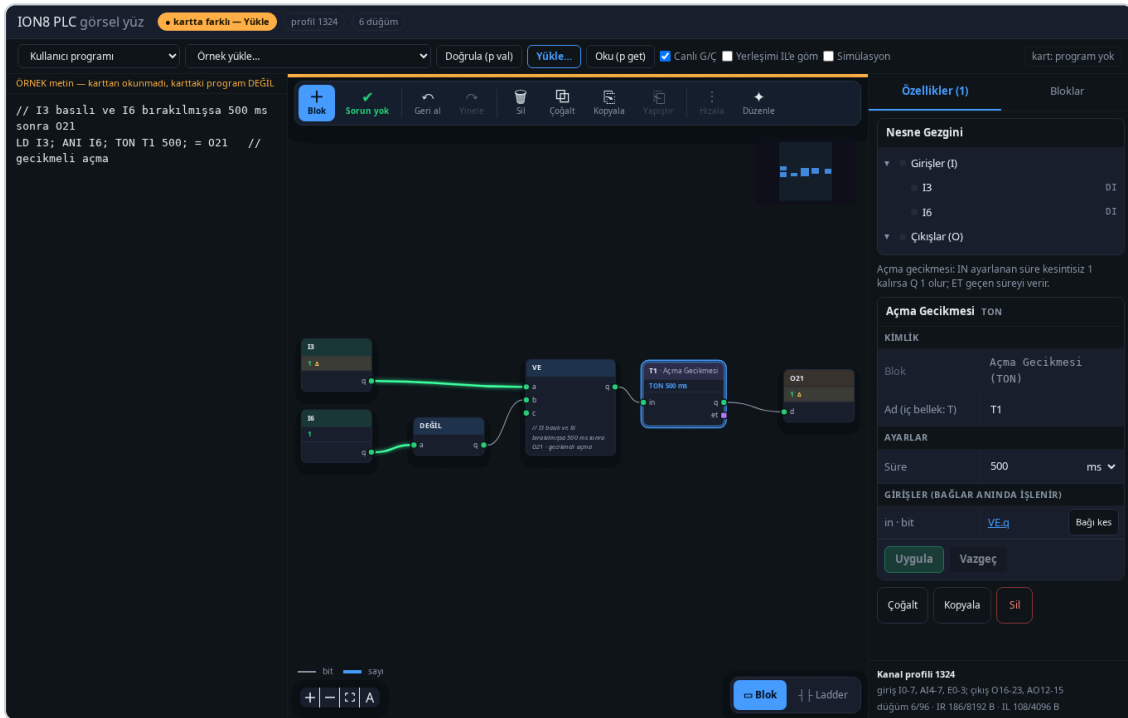


Figure 4.4 Properties panel of another block: setting fields depend on the block type.

4.5 Editing operations

Operation	What it does
Undo / Redo	Reverts or reapplies the latest edits.
Duplicate	Creates a copy of the selected block, with its settings, next to it.
Copy / Paste	Copies several blocks (with the wires between them) to the clipboard; pasting adds them with new identifiers.
Delete	Deletes the selected block(s) and their wires.

Align	Lines up the selected blocks.
Arrange	Re-lays out the whole chain left to right to reduce wire crossings. It does not change the program, only the view.

4.6 Memory and feedback

Memory blocks such as timers, counters and latches keep their value across program scans. When a value must be fed back to an input (as in a seal-in), use the **Previous Scan Value** block: it returns the value of what it is connected to from the **previous scan**, so the loop is built without a closed wire.

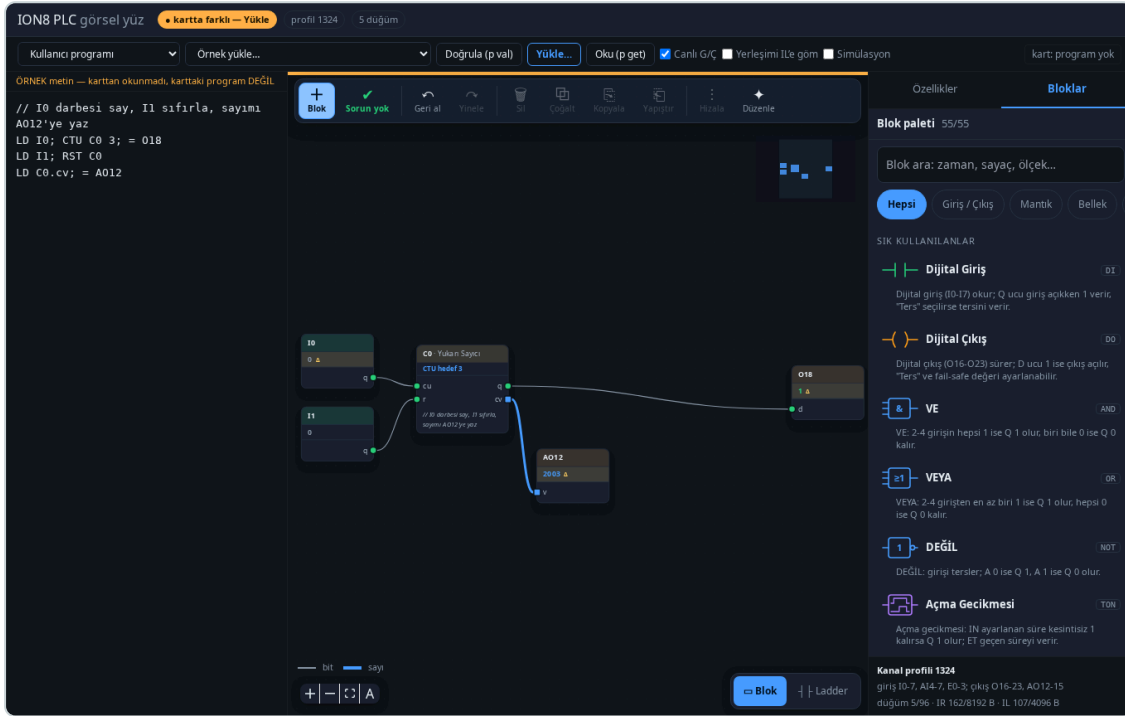


Figure 4.5 A chain built with a counter block: count input, reset and output.

4.7 Analog and numeric blocks

Number wires (blue) connect analog inputs (AI), analog outputs (AO), comparators, and function blocks such as scaling and ramp (BLK, Section 3.8). For example, scaling an AI input into a threshold block and wiring the threshold's bit output to a DO output gives "pull the relay in when the measurement exceeds the threshold".

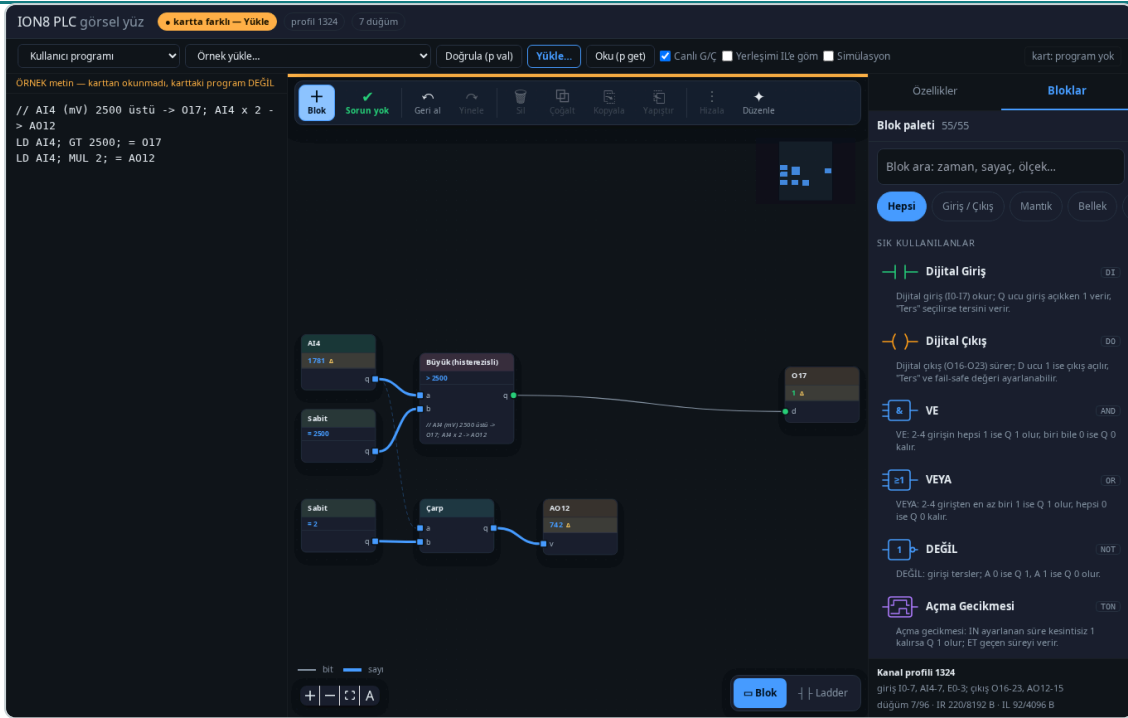


Figure 4.6 Analog input → scaling → threshold → output chain; blue wires carry numbers, grey wires carry bits.

4.8 Reading errors and warnings

When the toolbar indicator reads **No issues**, the program can be compiled. When there are errors it shows a count and marks the block or line concerned. A line in the instruction list that cannot be converted to a block appears on the canvas as a **Text block**: its text is preserved and the program can still be uploaded, but that line cannot be edited as a block. The warning text states the problem (e.g. "GT expects a number": a bit wire was connected to a comparator).

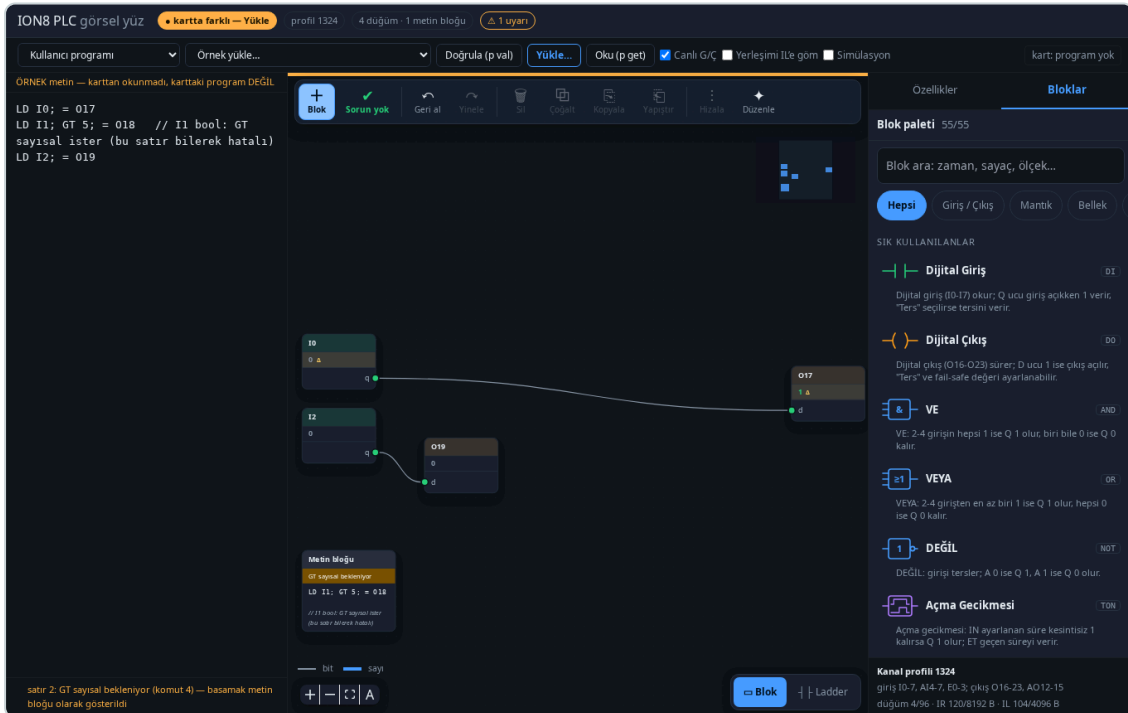


Figure 4.7 A deliberately faulty line: type-mismatch warning and Text block.

5. Ladder editor

The ladder view draws the same program as rungs, contacts and coils, for users familiar with relay panel diagrams. Choose **Ladder** on the mode selector; the program is drawn automatically from the instruction list.

5.1 Rung, contact and coil

A program consists of numbered **rungs**, top to bottom. On each rung the conditions (contacts) are on the left and the output (coil or box) on the right; conditions are in series, and **Branch** adds parallel paths. Each program ends with an **(End)** marker. The energised path is shown in green, live.

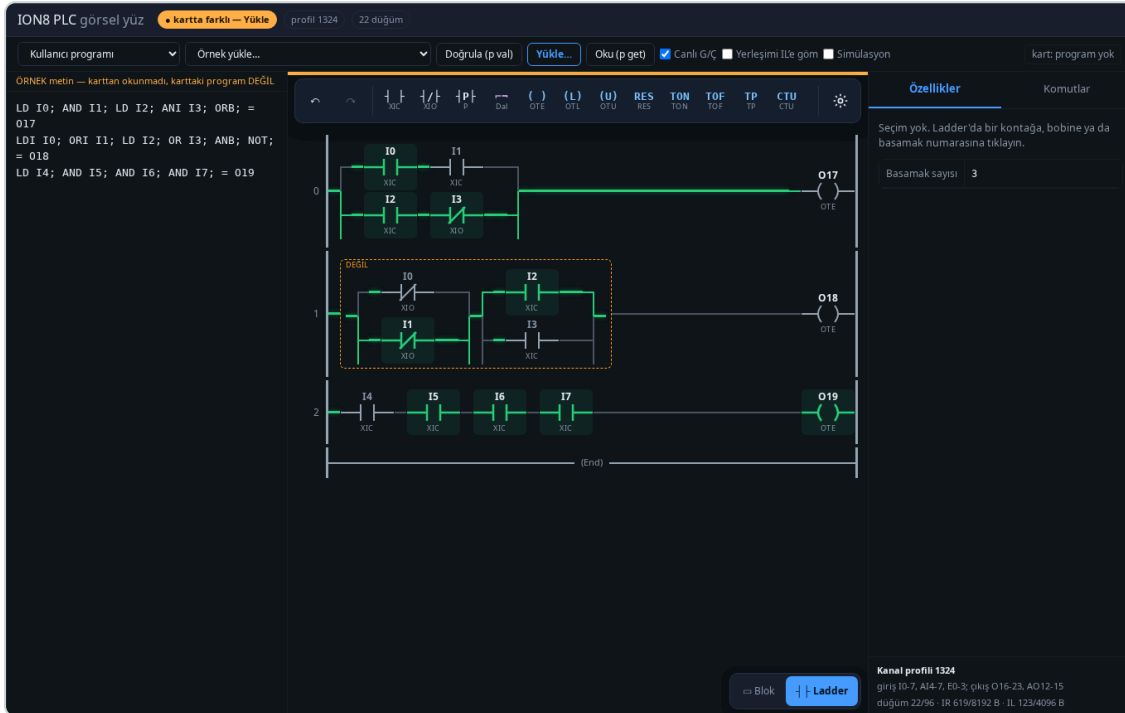


Figure 5.1 Ladder view: numbered rung, contacts, timer box (EN / DN outputs) and output coil.

Tool	Symbol	Meaning (instruction list equivalent)
XIC	$\begin{array}{ c } \hline \\ \hline \end{array}$	Normally open contact: passes when the address is 1 (LD / AND).
XIO	$\begin{array}{ c } \hline / \\ \hline \end{array}$	Normally closed contact: passes when the address is 0 (LDI / ANI).
P	$\begin{array}{ c } \hline P \\ \hline \end{array}$	Rising edge: passes for one scan when the address goes 0→1.
Branch	parallel path	Parallel condition (OR, ORB, ANB groups).
OTE	()	Output coil: 1 while the condition is true, otherwise 0 (=).
OTL / OTU	(L) (U)	Latch / unlatch (SET / RST).
RES	RES	Resets a timer or counter.
TON / TOF / TP	box	On-delay / off-delay / pulse. The box gives EN (enabled) and DN (done) outputs.
CTU	box	Up counter; count and reset conditions come from separate rungs.

5.2 Adding contacts and coils

Pick an item on the top toolbar and place it on the rung. When you click a contact or coil, a **context bar** opens below the canvas: contact **Type** (XIC/XIO/P), **Address** (e.g. I3, O21, T1), **+ Branch** (add a parallel path), **Move**, **Copy**, **Paste**, **Delete** and close (×). The identifier typed in the address box is checked at once; an unavailable channel is rejected.

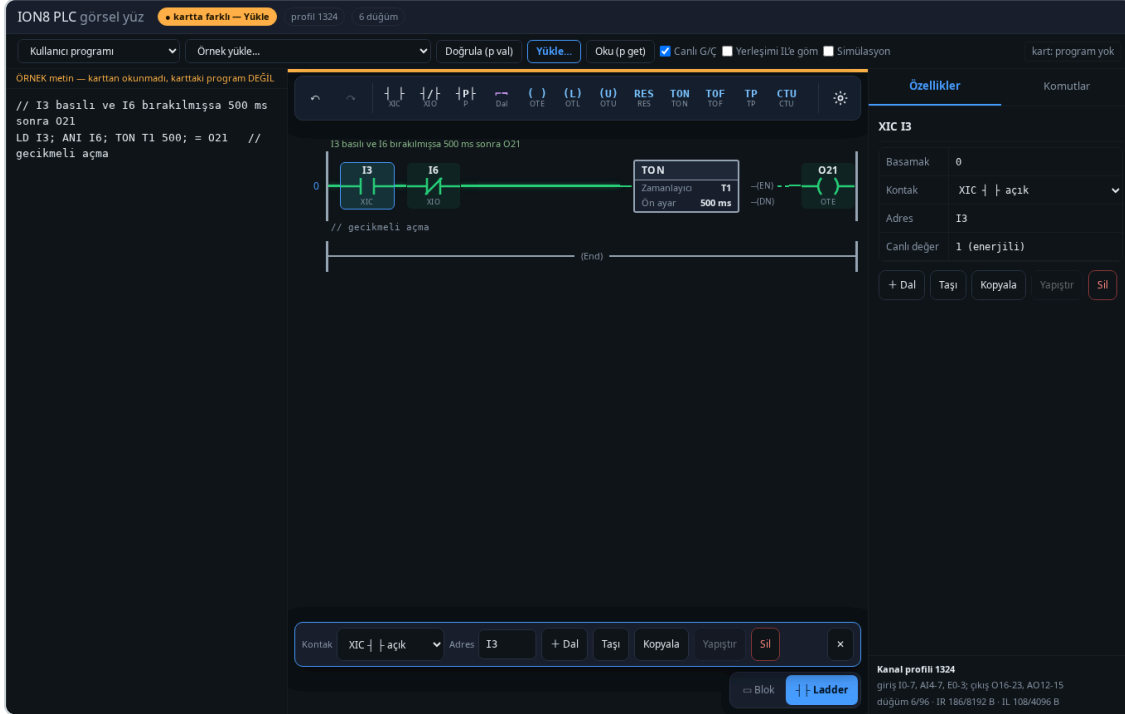


Figure 5.2 Selected contact (XIC I3): context bar below and the Properties panel on the right (rung, contact type, address, live value).

The **Properties** tab on the right shows the selected item's **Rung** number, **Contact** type, **Address** and **Live value** (e.g. "1 (energised)").

5.3 Commands tab

The **Commands** tab on the right lists the commands usable in ladder, in groups: **Conditions** (XIC, XIO, P, comparisons GRT, LES, GEQ, LEQ, EQU, NEQ), **Branch**, **Outputs** (OTE, OTL, OTU, RES, TON, TOF, TP, CTU, CTD) and **Numeric rung** commands. Click a command to add it to the selected rung.

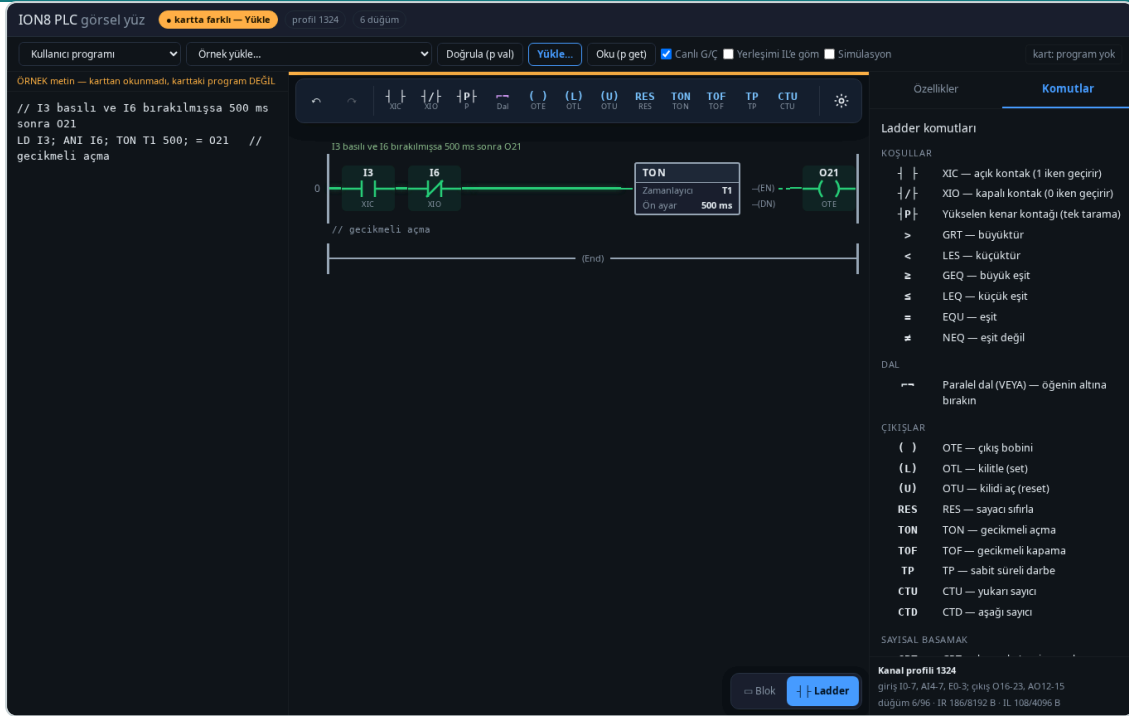


Figure 5.3 Commands tab: condition, branch, output and numeric command groups.

5.4 Seal-in, timer and counter examples

Start-stop (seal-in). I0 starts, I1 stops; O17's own contact is connected in parallel with I0 using **Branch**. The stop contact (XIO I1) is in series, so stop takes priority.

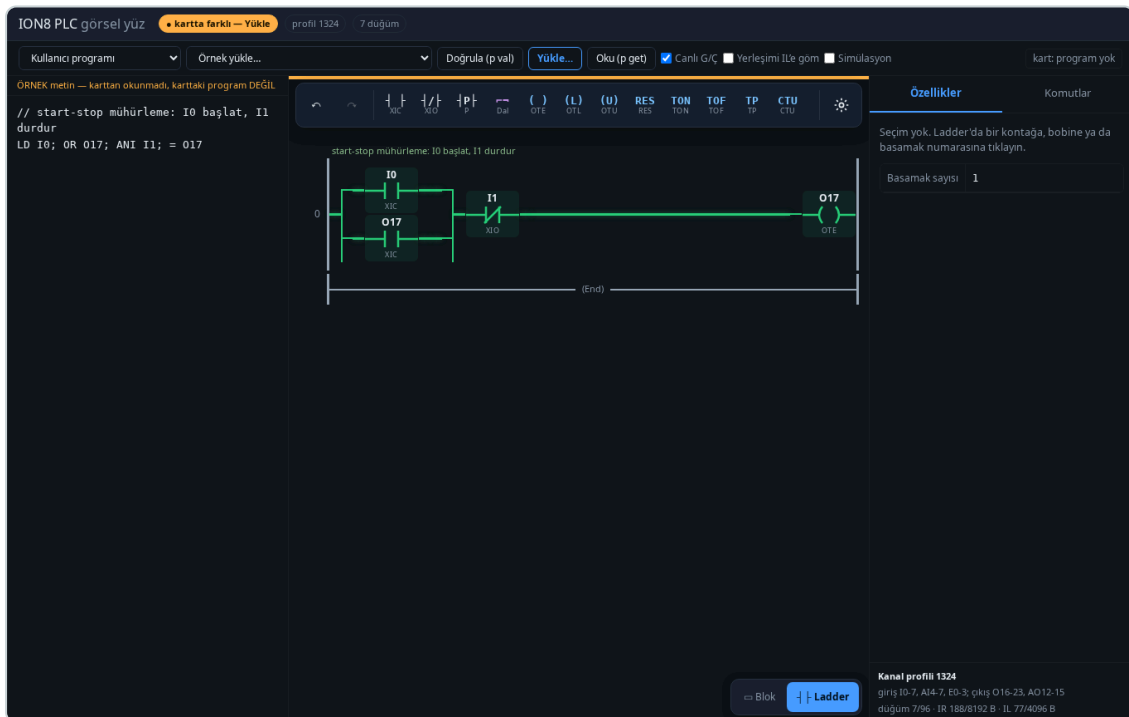


Figure 5.4 Seal-in rung: I0 || O17 parallel path, XIO I1 and the O17 coil.

On-delay. The conditions feed the timer box input; when the time has elapsed, the box's **DN** output drives the coil.

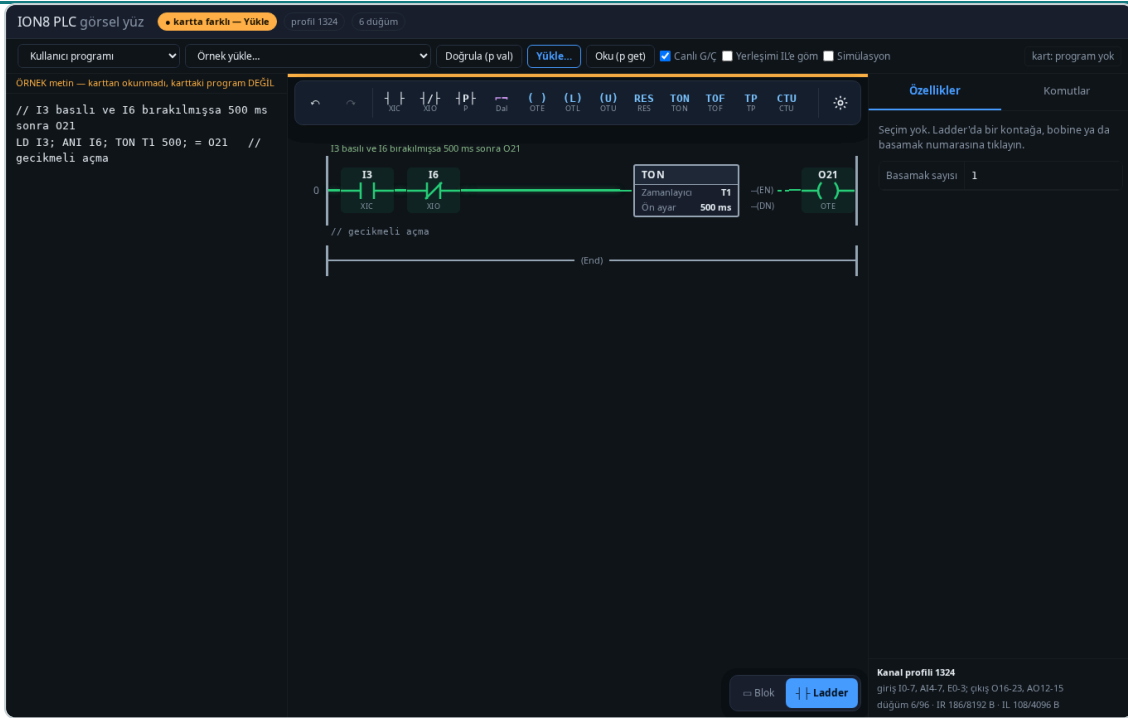


Figure 5.5 TON box: timer identifier, preset and EN / DN outputs.

Counter. The CTU box increments on each rising edge at its count input; when it reaches the preset, DN becomes 1. A RES rung resets the counter.

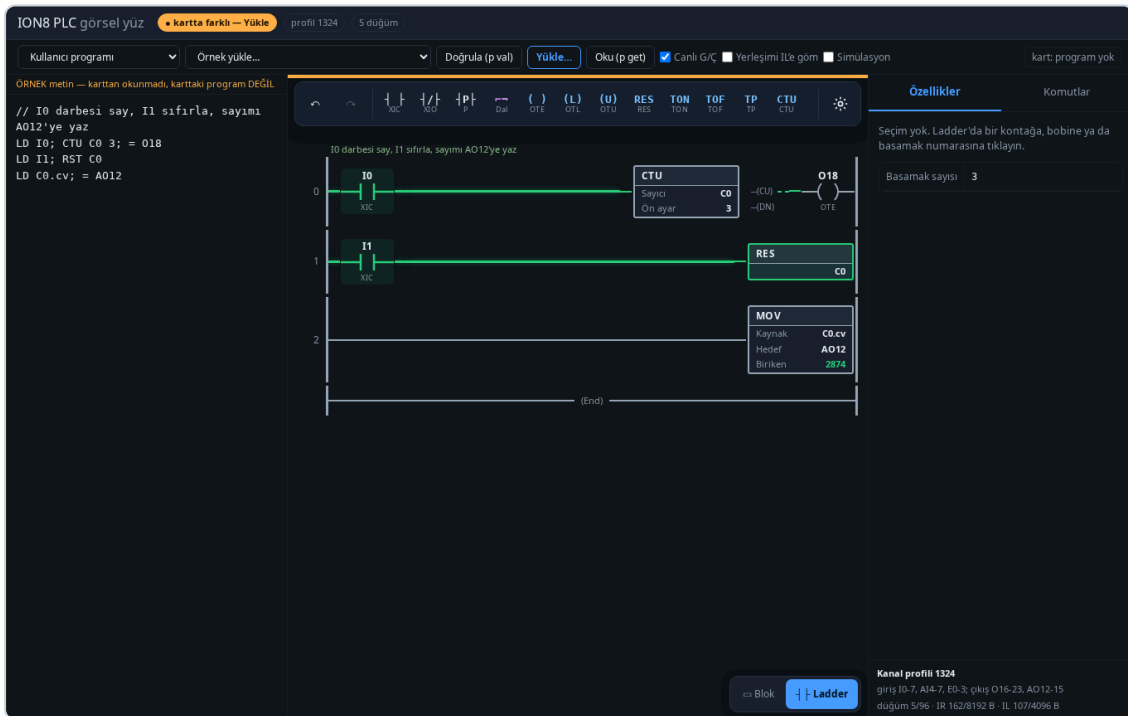


Figure 5.6 Counter (CTU) box and reset rung.

5.5 Constructs that cannot be shown as ladder

The ladder view does not cover the whole instruction list. Lines that cannot be converted (e.g. some function blocks) are shown as a **text rung** with their text preserved; edit them in Block mode or in the instruction list. Complex shared conditions are drawn as **ORB / ANB** groups, marked with "NOT" and group frames.

6. Using the three methods together, with examples

6.1 Moving between instruction list, blocks and ladder

The three views describe one program: a change in the instruction list appears in blocks and ladder, and a change in blocks or ladder appears in the instruction list. Suggested flow:

1. Open a ready example with **Load example...**, or fetch the device's current program with **Read**.
2. Make the change in the view you are most comfortable with (instruction list for short logic, blocks for data flow and analog, ladder for relay logic).
3. Fix errors until the toolbar indicator reads **No issues** (Sections 4.8, 3.7).
4. Have the device's compiler check it with **Validate**, then send it with **Upload...**

Embed layout in IL. When selected, block positions on the canvas are added to the instruction list as comments, so the layout is kept when the program is opened on another computer. When not selected, the layout is regenerated with **Arrange**.

6.2 Validate, Upload, Read

Button	Function
Validate	Checks the program with the device's compiler without sending it; shows the result (errors and warnings).
Upload...	Writes the program to the device and runs it; it is stored as a new version. Write permission is required.
Read	Brings the program running on the device into the editor. If the program on screen differs from the device's, a "program differs on device" badge appears at the top.
Live I/O	Shows the live state of inputs and outputs on the blocks and rungs.
Simulation	Runs the program with virtual input values you can try on screen, without uploading it to the device.

6.3 Other tabs of the PLC page

The **Program** tab shows the instruction list, **Live** the instantaneous values, **Tags** the names given to identifiers, and **Expert** diagnostics (Section 3.2).

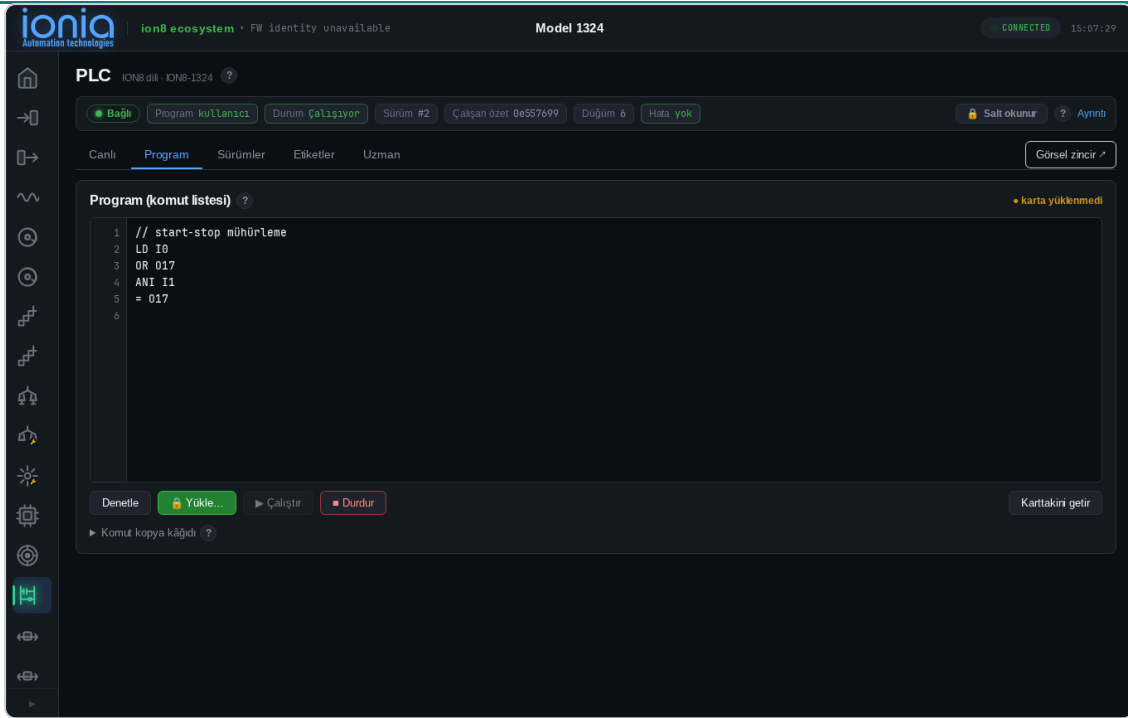


Figure 6.1 Program tab: instruction list editor with line numbers.

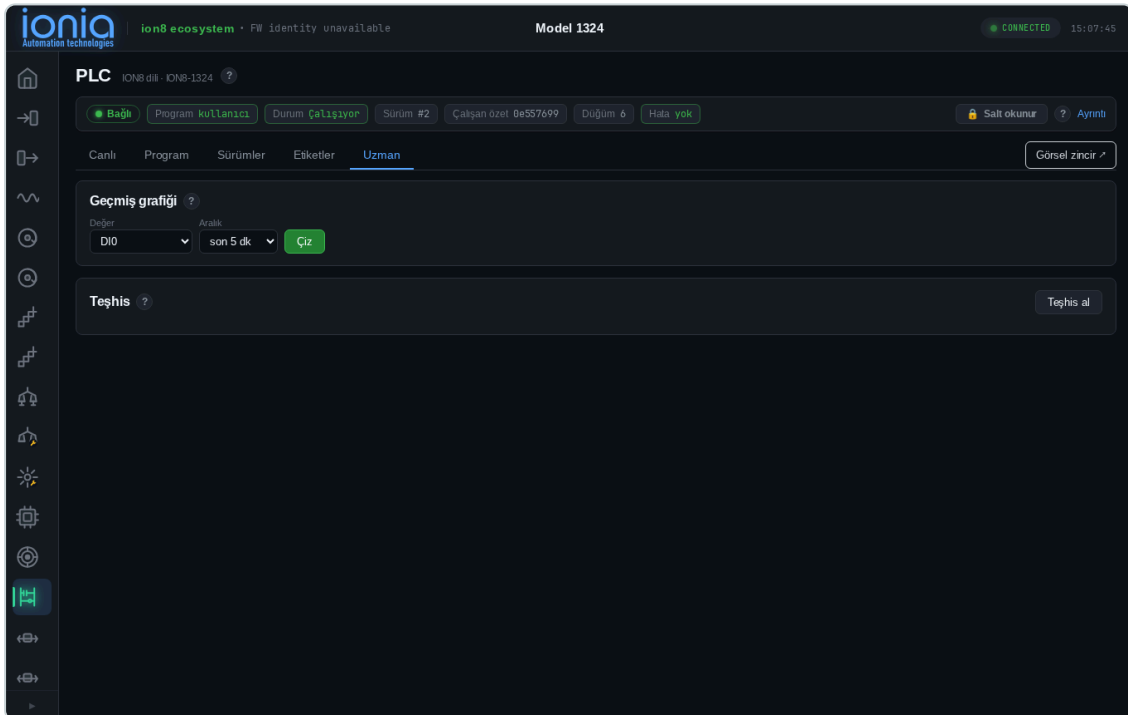


Figure 6.2 Live tab: instantaneous state of inputs, outputs and internal values.

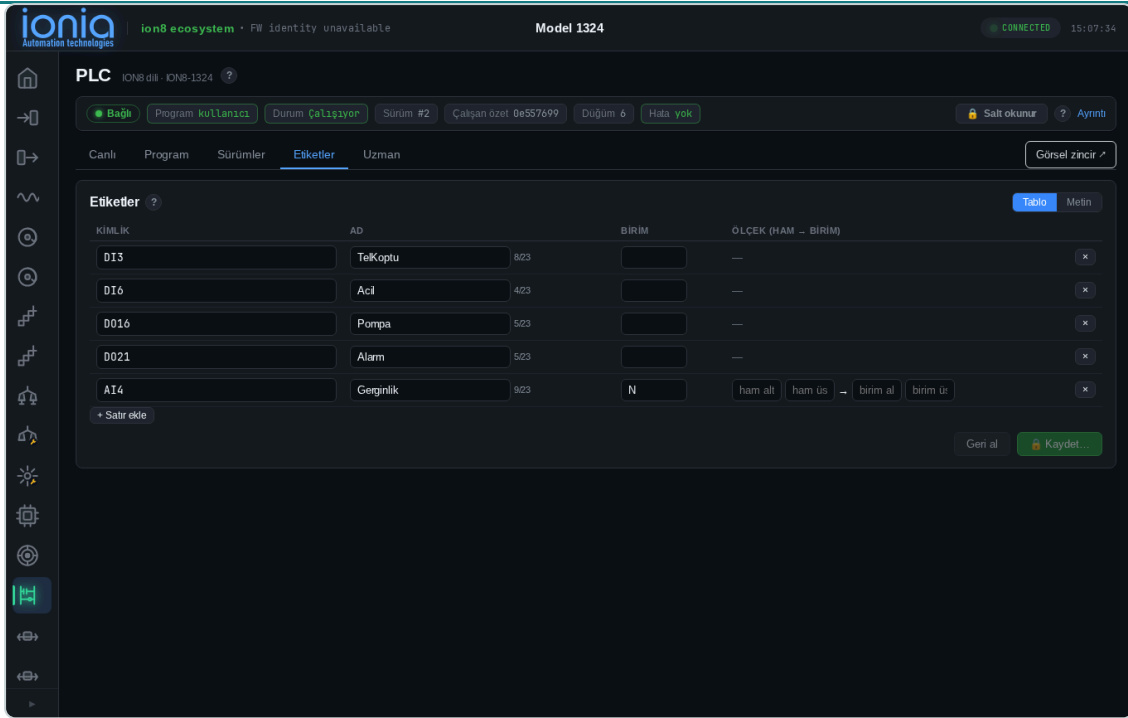


Figure 6.3 Tags tab: assigning names, units and scales to identifiers.

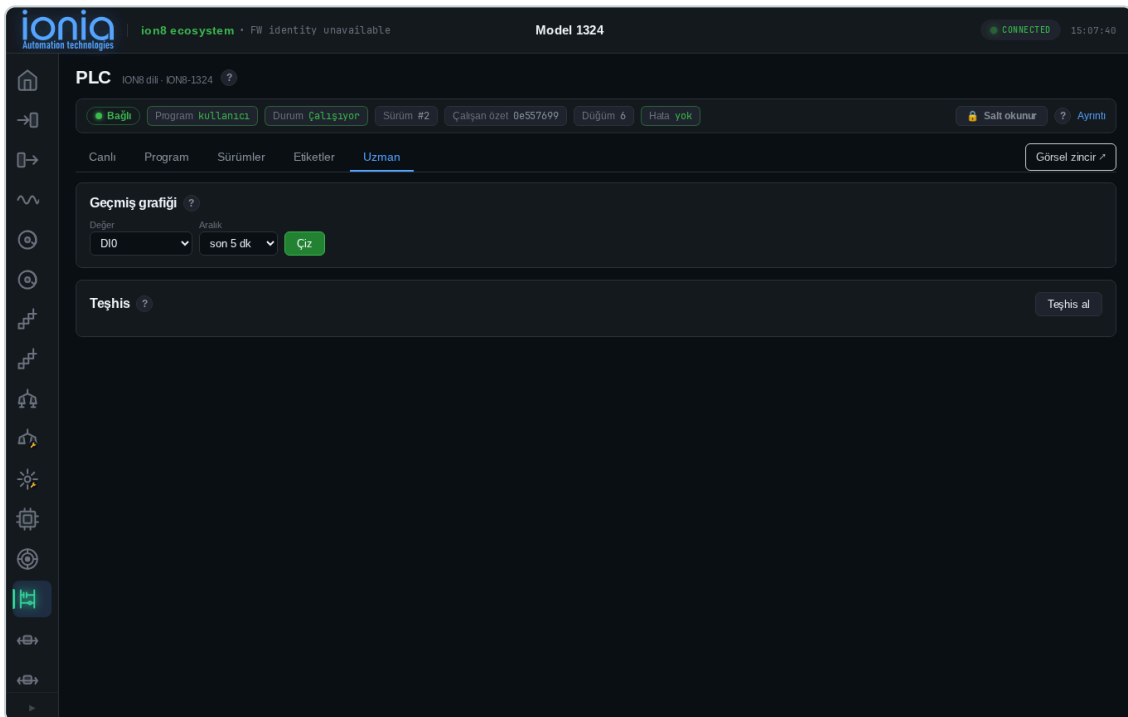


Figure 6.4 Expert tab: diagnostic and support information.

6.4 Step by step: delayed start for a pump

Goal: energise the pump relay O21 500 ms after I3 (run switch) is pressed while I6 (emergency interlock) is released.

1. Select **On-delay (TON)** from **Load example...**, or type the following lines into the instruction list:

```
LD I3  
ANI I6  
TON T1 500  
= O21
```

2. In Block mode, see the chain: I3 and the inverted I6 feed an AND block, the AND output feeds timer T1, and T1's output feeds O21.
3. Click T1, change **Time** in the Properties panel and press **Apply**.
4. Switch to Ladder and check the same rung: XIC I3, XIO I6, TON box, OTE O21.
5. Send it with **Validate** and **Upload...**; with **Live I/O** on, watch the input and output states.

Safety. Before driving a real pump or motor, verify the output channels and emergency-stop circuits in your own installation. The program is not a substitute for a safety function.

7. Using the PID blocks

The ION8-1324 has four independent PID loops (PID0 ... PID3). PID runs independently of the PLC program: saved and enabled loops run even if there is no program.

7.1 Loop structure

On the **PID** page each loop is shown as a signal chain:

Measurement → Scale → Setpoint and error → PID → Layers → Limits → Output

Select the loop and node in the object browser on the left; the property table on the right shows that node's settings. (On-screen node names: Ölçüm, Ölçek, Hedef ve sapma, PID, Katmanlar, Sınırlar, Çıkış.)

7.2 Settings

Node	Setting	Description
Measurement	Measurement (PV) source	The controlled quantity. Choose an identifier from the list (e.g. AI4) or enter the address to read manually.
	staleness time (ms)	If the measurement is not refreshed within this time it counts as a fault (at least 100 ms, default 500 ms).
	treat frozen measurement as fault	If the raw value does not change at all during the staleness time (sensor/cable fault), a fault is declared.
Scale	value = raw × multiplier + offset	Multiplier and offset applied to the measurement; the setpoint has its own multiplier/offset.
Setpoint and error	Setpoint (SP) source	Fixed value (written from the page) or another source. For cascade, choose the output of an outer loop (e.g. PID0.u).
	setpoint low/high limit, setpoint ramp (/s)	Range and rate of change of the setpoint. Low = high = 0: no limit. Ramp 0: none.
	reverse action	Select when the output must increase as the measurement increases (cooling, discharge).
	deadband, setpoint weight b, derivative setpoint weight c	Deadband: the output does not change while the error is below this value. b (0–1): a small value reduces overshoot on a setpoint step. c: if 0, the derivative acts on the measurement only (recommended).
PID	Kp	Proportional gain.
	Ti (s)	Integral time; 0 = no integral.
	Td (s)	Derivative time; 0 = no derivative.
	calculation interval (ms)	How often the loop is calculated, 5 ... 60000 ms (default 10 ms). A larger interval reduces overshoot in slow processes. Analog inputs refresh about every 50 ms, so 50 ms and above is sufficient.
	derivative filter N, tracking time Tt	Derivative filter time = Td / N. Tt is the tracking time that prevents the integral from winding up at an output limit; 0 = automatic.
Layers	Feed-forward, gain schedule	Feed-forward is added directly to the output (e.g. line speed). With the gain schedule on, Kp/Ti/Td come from a 3-point table that depends on a variable; values between points are linear.

Limits	output low/high limit	Range of the output.
	safe value	Value the output takes on a fault.
	output rate limit (/s)	Largest change of the output per second; 0 = none.
Output	Output target	A writable identifier (AO12...AO15 or PWM1...PWM3 duty/frequency) or a manual address. PWM0 is reserved for motion. PID output cannot be written to a digital output. If "none" is chosen the output stays only in the PID's own output value (a PLC program or another loop reads it).

If a manual address is used for a source or target, the data type (f32, i16, u16, i32, u32), scale and offset are also selected.

7.3 Applying and saving settings

1. Edit the node properties; changes are first held as pending, not yet applied.
2. **Uygula** (Apply): the device validates the setting as a whole. If it is invalid, nothing changes.
3. **Uygula ve Kaydet** (Apply and Save), or **Tümünü kaydet** (Save all) at the bottom of the page: the setting is stored permanently. **Kaydı sil** (Delete saved) removes the saved setting.

Gains (Kp, Ti, Td), mode and setpoint are written immediately. Saved and enabled loops **start by themselves** at power-up; no PLC program is needed. Saved PID settings (Kp, Ti, Td, setpoint, output limits and enabled state) are restored unchanged after a power cut and the loop continues running by itself. Saving does not disturb the control loop. If a motion is in progress the save is postponed and the page shows **kayıt bekliyor** (save pending); it is written when the motion ends.

7.4 Operating mode and safety

- **Instance:** Enabled / Off. **Mode:** Automatic / Manual. In manual mode the **manual output** value is applied within the output limits. Switching between automatic and manual causes no jump at the output.
- **integral:** **Reset** and **Freeze** buttons.
- The status field shows Unconfigured, Tuning test, Disabled, Safety, Manual, Automatic, Off or Waiting; faults are written as text.
- **Safety:** if the measurement is stale or bad, the output goes to the safe value (or stays at its last value if **hold last output on fault** is selected). If a PLC program owns the same analog output, the PID does not write to it.

7.5 Trend, smart tuning and diagnostics

- **Trend:** 1, 2 or 5 minute chart of measurement, setpoint and output.
- **Smart tuning:** the wizard performs a short test (you choose the target, amplitude, permitted measurement deviation, measurement low/high limits and maximum duration), derives a process model and suggests gains; it compares simulated curves of the suggested and previous gains (curves are computed in the browser; on the real process, noise and limits may change the result). In closed-loop mode the loop stays in automatic and the line is not stopped; the amplitude is in measurement units, not output units.
- Applied gains run but are **not permanent**; press **Kaydet** (Save) if you are satisfied, otherwise the device returns to the saved values after a restart.
- **Diagnostics:** internal state of the loop and faults.

Recommendation: For the first trial set the output limits and safe value according to the safe range of the process, run the loop first in **Manual** to confirm the measurement reads correctly, then switch to **Automatic**.

8. Troubleshooting

Symptom	Likely cause and remedy
The device network (ION8_1324_...) is not visible	Check that the device is powered. If an access point broadcast time is set the network may have switched off when it expired (default 0: never); restart the device.
Access point password forgotten	The password cannot be changed with the middle button. Return the device to factory settings with the middle button (Section 2.4.2): the password returns to the factory value and calibration is kept. Then set up network, pairing and operating mode again and change the password.
Connected to the network but the page does not open	Check that the address is <code>http://192.168.4.1</code> and that the computer has not switched automatically to another network (e.g. mobile data). The address on an existing network is shown in STA IP ; also try <code>http://ion8-1324.local</code> .
Does not join the existing wireless network	Only 2.4 GHz networks are supported. The Disc Reason row shows the cause: network not found, wrong password, etc. After changing settings make sure the device was restarted.
"Bağlantı Yok" in the top bar	The link to the device is lost. Check your network connection and the device power; values on screen are not current in this state.
"Salt okunur" badge; program cannot be loaded	No write authority. The authority state is shown in the top bar; contact Ionia for support about authority.
Program does not load, a line is flagged	Examine the error message using Section 3.7: lowercase command, double =, memory never written, bad time.
"Kanal çakışması" (channel conflict) error	The program writes to an output reserved for PWM. Change the PWM setting or use another output in the program.
Çalıştır button is inactive	The device has no program. Load one first with Yükle .
Program does not run after restart	The program had been stopped with Durdur ; stop is persistent. Press Çalıştır .
Cannot write to an output from the DO page	The output belongs to a PLC program or to motion/PWM. The label shows "PLC" or "Hareket / PWM".
PID output stays constant	Check the status field: it may be Safety (measurement stale/bad), Manual, Off or Disabled. Check the measurement source and staleness time; check whether the output limits are narrow.
PID setting is not saved ("kayıt bekliyor")	The save is postponed while a motion is in progress; it is written when the motion ends.
Mirroring does not work	One unit must be Master and the other Slave (not BRIDGE); repeat the pairing steps. Check the signal bar; if weak move the units closer or select Range mode. Make sure the target output is not used by another function (Section 2.2.3).
Too many pulses counted on a mirrored input	"Yüksek hızlı sensör" may be selected for a mechanical contact input. Set the type to "Mechanical contact / relay" (Section 2.3).
Update "Reddedildi" (Rejected)	The package is corrupt or not suitable for this device. Download the file again; if the problem persists contact Ionia.

In support requests, quoting the build signature (top bar) and, if available, the diagnostic output in the **Uzman** tab of the PLC page speeds up the process.