



ASFAN

FR5 Digital Twin & PLC Studio

User Manual: a virtual Fairino FR5 V6 collaborative robot with a software PLC, for training, programming and cell design. No robot, PLC or other hardware needed.

Version 0.3.0 · English

Product	FR5 Digital Twin + FR5 PLC Studio (desktop application for Windows)
Robot model	Fairino FR5 V6, 6 axes (original URDF / SRDF description)
PLC notations	Siemens · Schneider Electric · Mitsubishi Electric · Rockwell (Allen-Bradley) · Omron
Edition	October 2026

Contents

- 1** Introduction
- 2** Getting started
- 3** The main window
- 4** Navigation menu: all pages
- 5** The 3D view
- 6** Moving the robot by hand
- 7** Teaching points and drag teach
- 8** Programming the robot
- 9** Work cells (environments)
- 10** Safety settings
- 11** Inputs / outputs and the PLC link
- 12** PLC Studio
- 13** Tutorial: robot + PLC cell
- 14** Display options: text size and language
- 15** Keyboard shortcuts
- 16** Saving, files and troubleshooting
- 17** Support

1 Introduction

FR5 Digital Twin is a complete virtual copy of the **Fairino FR5 V6** six-axis collaborative robot. It runs entirely on your computer. You can jog the robot, teach points, write and run programs, build production cells and test safety settings without any real hardware.

FR5 PLC Studio is the companion application: a software PLC with ladder-logic editing in the notation of five major PLC brands. Its inputs and outputs are wired live to the robot twin, so you can design and test a full robot + PLC automation cell on one PC.

What you can do

Learn robot programming · prepare programs before going on site · train operators and students · demonstrate cells to customers · test PLC/robot handshakes.

What it is not

The twin does not connect to a real controller (simulation only) and does not check self-collision. PLC Studio is not vendor software; brand names only identify the notation that is modelled.

Main features

- Accurate kinematics from the robot's original URDF/SRDF description, with joint limits and named poses.
- Joint jog, Cartesian (TCP) jog and inverse kinematics, plus drag teach directly in the 3D view.
- Programming in two modes: **Graphical** (command list) and **Coding** (Lua 5.4, Fairino-style commands).
- Ready-made work cells: production line, machine tending (CNC), palletizing and MIG welding.
- Safety: workspace box, joint soft limits, speed limits and floor clearance, with a safety stop.
- Virtual control box I/O (DI/DO, CI/CO, tool I/O) and a live link to PLC Studio.
- ASFAN interface: light theme, adjustable text size and a full Arabic / English interface.

2 Getting started

Starting the applications

- Open **FR5 Digital Twin** from its desktop or Start-menu shortcut.
- Open **PLC Studio** from its own shortcut, from **File > FR5 PLC Studio** (**Ctrl+2**) or with the **Open PLC Studio** button on the I/O page. Return to the robot with **File > FR5 Digital Twin** (**Ctrl+1**).
- Both windows remember their size and position.

Free trial and license

- The program works with **all features for 10 days** without a license. At each start the license window shows the days left. Press **Continue the free trial** to open the program.
- To get a license, press **Request a license on WhatsApp** in the license window. A message with your **Machine ID** is prepared for ASFAN.
- After payment you receive a license key (**FR5DT1-...**) or a **.lic** file. Paste the key, or use **Load license file**, then press **Activate** and **Open the program**.
- Licenses are monthly, yearly or until a set date. The program reminds you 7 days before the end. To renew, paste the new key.
- You can view the license or enter a new key at any time from **Help > License....**

Do not change the computer date: if the date is set back, the program stops until the correct date is restored.

The application menu

Menu	Items
File	FR5 Digital Twin (Ctrl+1) · FR5 PLC Studio (Ctrl+2) · Open the saved work folder · Exit
View	Reload · Force reload · Reset zoom / Zoom in / Zoom out · Full screen
Help	License... · Check for updates... · About FR5 Digital Twin (version, license, Machine ID, credits)

Tip: the menu bar is hidden automatically. Press **Alt** to show it.

3 The main window

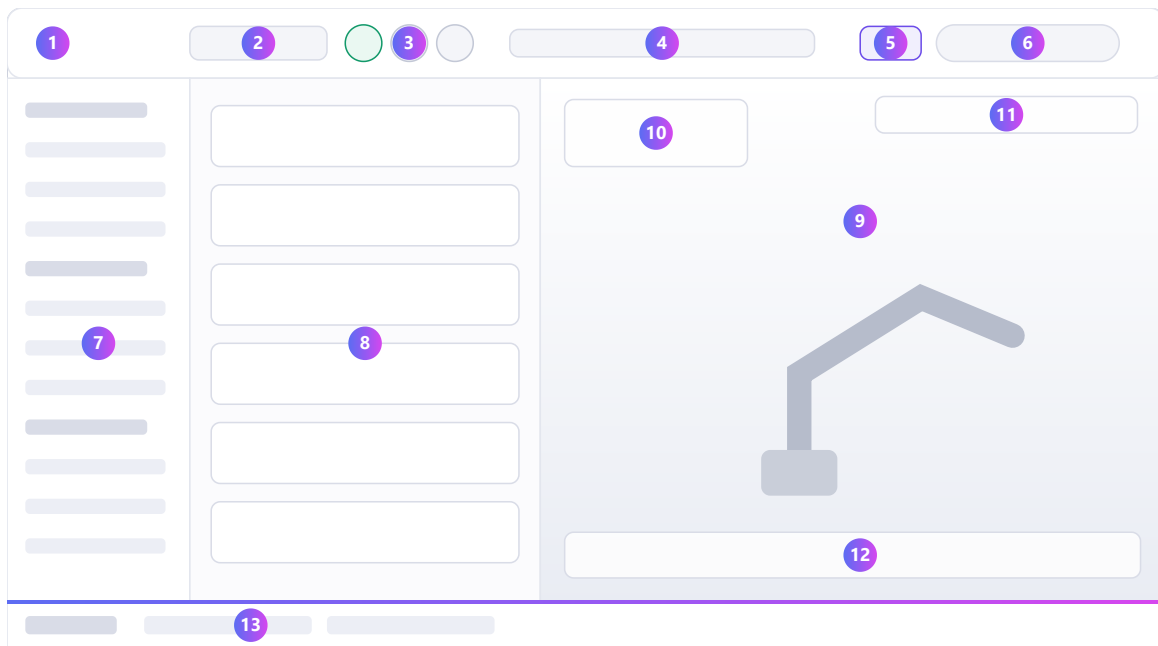


Figure 1: Main window of FR5 Digital Twin

#	Element	Use
1	Menu button	Shows or hides the navigation menu (7).
2	Program name	The active program (for example <code>main.lua</code>). Click it to open the program.
3	Run · Stop · Pause	Starts the active program, stops program and motion, or pauses and resumes.
4	State · Speed · Mode · Alarm	Program state (Stopped / Running / Paused), speed override in % (click to change), mode (Manual / Auto / Drag) and active alarms.
5	SIM / LIVE	Data source. SIM is the built-in simulation. LIVE is shown but unavailable because no controller is connected.
6	A- · A+ · Language	Text size and the Arabic/English switch.
7	Navigation menu	All pages, grouped as Initial, Program, Manual and Status (see chapter 4).
8	Side panel	Content of the selected page. Hide or show it with <code>Ctrl+B</code> .
9	3D view	The robot and its work cell (see chapter 5).
10	TCP readout	Live tool position X, Y, Z (mm) and orientation R, P, Y (°) in the world frame.
11	View toolbar	Drag teach, fit robot to view, frames/axes, collision overlay, floor grid and panel toggle.
12	Joint readouts	Current angle of J1 to J6, plus the floor clearance of the lowest point of the arm.
13	Footer	ASFAN contact: e-mail and WhatsApp.

4 Navigation menu: all pages

Group	Page	What it contains
Initial	Overview	Robot model summary, TCP pose, joint positions (bars show the position inside the URDF limits) and model notices.
	Environment	Choose the work cell (chapter 9), conveyor speed, part feed and cell signals.
	Safety	Workspace box, joint soft limits, speed limits and floor clearance (chapter 10).
	Robot Groups	Planning group, kinematic chain, joint table (origin, axis, limits) and SRDF information.
	Named Poses	Reference poses: Home, Zero, Upright, Tool down, Pick approach, Side reach, Folded. Click to move.
	Settings	Angle unit (degrees / radians), motion interpolation, IK solver information and application details.
Program	Coding	Lua text editor with syntax check, examples and the Lua API reference.
	Graphical	Build programs from a list of commands without typing code.
	Points	Teach, preview, re-teach and delete named points.
Manual	Joint Jog	Slider and value for each joint J1 to J6.
	TCP / Cartesian	Move the tool in X, Y, Z and rotation with inverse kinematics.
Status	I/O	Control-box and tool inputs/outputs, CI/CO functions and the PLC link (chapter 11).
	Diagnostics	Package files, meshes, FK cross-check, last IK result and renderer status.

5 The 3D view

Action	Mouse / key
Rotate (orbit)	Left-drag
Pan	Right-drag
Zoom	Mouse wheel
Hide / show the side panel	Ctrl+B

View toolbar

- **Drag teach (hand icon):** move the arm with the mouse and record points (chapter 7).
- **Fit robot to view:** frames the robot when you have zoomed or panned away.
- **Frames / axes:** shows the coordinate frames (X red, Y green, Z blue). Distances are in metres with Z pointing up.
- **Collision overlay:** shows the collision meshes used by the floor-clearance check.
- **Floor grid:** shows or hides the measuring grid.

On-screen information

- **TCP readout** (top corner): live tool position and orientation.
- **Joint readouts** (bottom): angles of J1 to J6.
- **Floor clearance:** lowest point of the moving arm above the floor safety limit, and which link it is.
- A red **SAFETY STOP** banner appears when a move is refused (chapter 10).

If the 3D view does not start: the PC needs WebGL. Update the graphics driver or enable hardware acceleration, then use **View > Reload**.

6 Moving the robot by hand

Joint Jog (Manual › Joint Jog)

- Each joint J1 to J6 has a slider limited to its URDF range.
- Sliders move the robot instantly. A typed value (confirm with `Enter`, cancel with `Esc`) moves smoothly according to the interpolation setting.
- Values outside the limits are clamped automatically.

TCP / Cartesian (Manual › TCP / Cartesian)

- Enter a target X, Y, Z (mm) and Roll, Pitch, Yaw (°) in the world frame (robot base), then press **Solve & move**.
- Use the `+` / `-` jog buttons to move or rotate the tool step by step along each axis.
- Inverse kinematics keeps the solution closest to the current joints and respects the URDF limits. Moves are in joint space, not straight lines.
- If a target is out of reach, the tool stays at the last reachable pose and a message explains why.

Named poses and Home

On **Initial › Named Poses**, click a pose (Home, Zero, Upright...) to move there. In a work cell, **Move robot home** uses a safe route from anywhere in the cell.

Manual control is locked while a program runs. Press **Stop** in the top bar to take over.

7 Teaching points and drag teach

Points page (Program > Points)

- 1 Move the robot to the wanted position (jog, TCP or drag teach).
- 2 Type a point name (letters, digits and `_`) and press **Teach point**, or press `R`.
- 3 The point stores the joint angles and the TCP pose. Use **Preview** to show it, **Move** to go there, **Re-teach** to update it, or **Delete**.

Points become variables in programs, for example `PTP(P1, 30)` or `Lin(P2, 120)`. **Add example points** creates a ready-made set.

Drag teach

- 1 Click the hand icon in the view toolbar. The mode changes to **DRAG**.
- 2 Drag the tool in the 3D view to move the arm by hand.
- 3 Press **Record point** (or `R`) at each position. Recent points are shown as chips you can click.
- 4 Press `Esc` or click the hand icon again to leave drag teach.

Drag teach is locked while a program is running.

8 Programming the robot

A program is a list of motion and I/O commands, written in **Lua 5.4** with Fairino-style functions. You can edit the same program in either mode and switch at any time with the **Coding / Graphical** button.

Graphical mode (Program > Graphical)

- 1 Pick a command on the left: Motion (PTP, LIN, MoveJ, MoveL, Home, Named pose, Joint speed, Linear speed), logic (Loop, While, If, Else, End, Wait, Pause, Var, Print, Comment, Lua) or I/O (Set DO, Wait DI).
- 2 Fill in its fields. **Use current** takes the robot's present position.
- 3 Press **Add**. The line is inserted after the selected line, or replaces it.
- 4 Click a line to edit it, or double-click it to run only that line.

The toolbar has save, copy / cut / paste, delete (`Del`), move a line (`Alt+↑ / ↓`), undo / redo and syntax check.

Coding mode (Program > Coding)

Type Lua code directly. **Check syntax** validates the program. **Import / Export .lua file** exchanges programs with other PCs. **Examples...** loads a ready program.

Running a program

- ► **Run** starts the program. **Pause** holds it, and pressing again resumes. **Stop** stops program and motion.
- The **Speed %** override in the top bar scales every program move.
- Messages from `print()` and errors appear in the console under the program.

Lua command reference

Command	Description
<code>PTP(point [, speed%])</code>	Joint move to a taught point.
<code>Lin(point [, mm/s])</code>	Straight-line TCP move to a taught point.
<code>MoveJ(j1, ..., j6 [, speed%])</code>	Joint move to absolute angles (°).
<code>MoveL(x, y, z, rx, ry, rz [, mm/s])</code>	Straight-line move to an explicit world-frame pose (mm, °).
<code>MoveHome([speed%])</code> <code>MovePose(name [, speed%])</code>	Move to Home, or to a named pose such as "zero". In a work cell, <code>MoveHome()</code> uses a safe route.
<code>WaitMs(ms)</code> · <code>Pause()</code>	Wait for a time (pausable) · hold the program until Resume.
<code>SetSpeed(%)</code> <code>SetLinearSpeed(mm/s)</code>	Default speeds for later moves (initially 25 % and 100 mm/s).
<code>GetJoints()</code> · <code>GetTCP()</code>	Current joint angles (°) · current TCP x, y, z (mm), rx, ry, rz (°).

<code>SetDO(id, 0 1)</code> <code>SetToolDO(id, 0 1)</code>	Control-box output 0–15 (8–15 = CO0–CO7) · tool output 0–1. GetDO / GetToolDO read them back.
<code>GetDI(id)</code> · <code>GetToolDI(id)</code>	Input state 0 or 1 (control box 0–15, 8–15 = CI0–CI7 · tool 0–1).
<code>WaitDI(id, 0 1 [, maxtime [, opt]])</code>	Wait for an input (WaitToolDI for the tool). maxtime 0 waits forever. On timeout, opt 0 stops with an error and opt 1 continues.
<code>print(...)</code>	Write to the console.

Example: pick & place square

```
-- Pick & place square (tool pointing down)
SetSpeed(30)
SetLinearSpeed(120)
MoveHome()
MoveJ(0, -90, -90, -90, 90, 0)
for _ = 1, 2 do
    MoveL(497, -102.1, 377, 180, 0, -90)
    WaitMs(300)
    MoveL(597, -102.1, 377, 180, 0, -90)
    MoveL(597, 47.9, 377, 180, 0, -90)
    MoveL(497, 47.9, 377, 180, 0, -90)
end
print("Cycle complete")
```

Built-in examples (Examples...)

Pick & place square · Relative square (variables) · Joint tour · Pick & place with taught points · Production line · Machine tending · Palletizing · Welding · PLC cell (I/O handshake).

The examples for a work cell expect that cell to be loaded on the Environment page.

9 Work cells (environments)

On **Initial > Environment**, choose the scene around the robot. Stop the program before you change the environment.

Work cell	Description	Main signals
Default	Robot only, on the studio floor with the measuring grid. Best for jogging, teaching and checking kinematics.	None
Production line	Parts arrive on the infeed conveyor and stop at a photo-eye. The robot picks each part with the gripper and places it on the outfeed conveyor.	DI1 part at pick position · tool DO0 gripper · tool DI0 grip sensor
Machine tending	The robot loads a CNC cutting table, moves clear, starts the cycle (the gantry cuts round the part) and then unloads the finished part.	DI1 blank ready · DO1 cycle start · DI2 cycle done
Palletizing	Cartons arrive on the infeed conveyor and are stacked on a pallet in 2 layers of 4. Use Swap pallet when it is full.	DI1 carton ready · DI2 pallet full
Welding	A MIG torch welds two fillet seams on a T-joint clamped on the welding table, at a 45° work angle.	DI1 clamped · DO2 arc on · DI3 arc established · DO3 unload

Cell controls include conveyor speed, a new part every N seconds, **Clear cell**, **Swap pallet** and the counters (fed, picked, placed, shipped, machined...).

10 Safety settings

Every move is checked before it happens: sliders, jog, IK, drag teach and programs. A refused move stops the robot at its last safe pose. Moves back toward safety are always allowed.

Setting	Description
Workspace box (TCP)	Keeps the tool inside a box (X/Y/Z min–max in mm, world frame), shown as an orange box in the 3D view.
Joint soft limits	Tighter min/max limits than the URDF for each joint.
Speed limits	Max joint speed (% of the URDF velocity) and max TCP speed (mm/s). They cap every program move before the speed override is applied.
Floor clearance	Stops before the arm touches the floor (floor height and margin in mm), using the collision meshes. The base and J1 are exempt.
Reset safety settings	Restores all safety values to their defaults.

SAFETY STOP: a red banner shows which part would hit (gripper, torch, held part...). Clear it with the button in the banner, then move away from the limit.

The simulation does not check self-collision or collisions between the robot and the machines. Always validate programs on the real robot at reduced speed.

11 Inputs / outputs and the PLC link

The **Status** > **I/O** page shows the virtual control box of the FR5:

Group	Channels	Notes
Digital inputs	DI0–DI7	General inputs. Click to simulate by hand when not wired to the PLC.
Configurable inputs	CI0–CI7	Each can have a function (program start, stop, pause, resume, clear fault...), triggered on a rising edge (0 → 1).
Digital outputs	DO0–DO7	Set by programs (SetDO) or by hand.
Configurable outputs	CO0–CO7	Report the robot state (ready, running, moving, fault, paused...).
Tool I/O (flange)	Tool DI0–1 · DO0–1	Gripper, torch and sensors at the flange.

PLC link

- When PLC Studio is open, it links to the twin automatically. The status shows **LINKED** and the PLC mode (RUN / STOP).
- PLC outputs drive robot inputs (DI/CI). Robot outputs (DO/CO) feed PLC inputs.
- Everything stays on this computer. If one application closes, the other sees its signals drop to 0.

12 PLC Studio

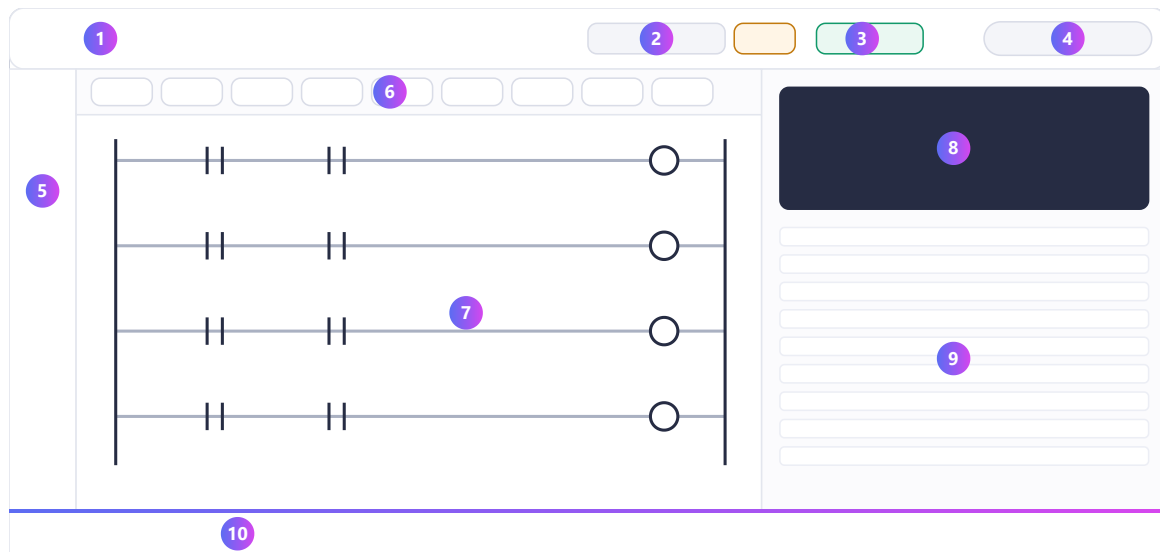


Figure 2: PLC Studio window

#	Element	Use
1	Project · CPU	Project name and the selected PLC brand/CPU (click to change).
2	RUN / STOP	Starts the PLC (cold restart) or stops it. In STOP, inputs are read but all outputs and lamps stay off.
3	Diagnostics · Panel	Error count ("No errors") and the floating operator panel.
4	Robot link	Link state with the FR5 twin (for example FR5 V6 · IDLE), plus text size and language.
5	Sections	Ladder program · Tags · Text listing · PLC hardware · Operator panel · I/O wiring · Robot link · Project & PLC.
6	Instruction toolbar	New rung, instruction picker, contacts, OR branch, coils, timers and counters.
7	Ladder editor	Networks (rungs) with comments. Live power flow is shown while the PLC is in RUN.
8	PLC view	Picture of the selected PLC with live input/output LEDs.
9	I/O list	Every input and output with its address, tag and state. Click to operate or to find it in the ladder.
10	Footer	ASFAN contact.

Supported PLC notations

Brand	CPUs	Editor notation
-------	------	-----------------

Siemens	SIMATIC S7-1200 (CPU 1214C, 1215C) · S7-1500 (CPU 1511-1 PN)	TIA Portal · LAD
Schneider Electric	Modicon M221 (TM221CE24T, CE40T) · M241 (TM241CE40T)	EcoStruxure Machine Expert – Basic
Mitsubishi Electric	MELSEC iQ-F FX5U-32MT/ES, FX5U-64MT/ES, FX5UC-32MT/D	GX Works3 · Ladder
Rockwell (Allen-Bradley)	MicroLogix 1100 (1763-L16BBB) · 1400 (1766-L32BXB)	RSLogix 500 · Ladder
Omron	SYSMAC CP1L-EM40DT-D · CP2E-N30DT-D, N40DT-D	CX-Programmer · Ladder

Creating a project

- 1 Choose **New project...**, then select the PLC brand and CPU.
- 2 Start from the **Robot-cell demo** (panel, handshake with the FR5 and a parts counter) or an **empty program**.
- 3 The project is saved automatically. Use **Export project file** to keep copies or move it to another PC, and **Open project file...** to load one.
- 4 **Change brand / CPU...** keeps the program and shows it in the notation of the new brand.

Editing ladder logic

- Select an element in a rung to edit it, or use the + slots to insert. **Instruction...** () opens the full library.
- Instructions: NO/NC contacts, rising/falling edge, one-shot, compare (EQU...LEQ), output, negated, set and reset coils, timers (TON, TOF, TP, retentive), counters (CTU, CTD), MOVE and math (ADD, SUB, MUL, DIV).
- Operands: type an address in the brand's format (for example `%I0.0`, `X0`, `I:0/0`) or a tag name. Timer values are in milliseconds.
- **Tags** gives addresses symbolic names. **Show tag names instead of addresses** switches the display.
- **Text listing** shows the program as read-only instruction text, useful for reviews and for re-entering the logic in the vendor tool.

Operator panel and wiring

- **Operator panel:** push-buttons (active while held), selector switches and lamps wired to the PLC. It can float over the ladder (**Panel** button).
- **I/O wiring:** connects PLC inputs/outputs to the panel and to the robot signals (DO/CO → PLC inputs, PLC outputs → DI/CI).
- **Project & PLC:** name, PLC hardware, scan cycle time and scan statistics.

13 Tutorial: robot + PLC cell

This exercise runs the ready-made demo in which the PLC starts and supervises the robot program through the I/O handshake.

- 1 Open **PLC Studio** and create a project from the **Robot-cell demo** (any brand).
- 2 Open the **FR5 robot twin**. It links automatically and the robot link shows **LINKED**.
- 3 In the twin, open **Program > Coding > Examples...** and load **PLC cell (I/O handshake)**.
- 4 In PLC Studio, switch the PLC to **RUN**.
- 5 On the operator panel, turn **Auto_Mode** on, make sure **Door_Closed** is on, then press **Start_PB**.
- 6 Switch **Part_Sensor** on: the robot receives "part ready" and runs its cycle. The PLC counts finished parts, and the batch lamp lights after 5 parts.
- 7 Try **Stop_PB**, opening the door (pause), **Start_PB** again (resume) and **Reset_PB** (clear fault / new batch).

Watch both windows: LEDs in the PLC view, live power flow in the ladder and the I/O page in the twin all update together.

14 Display options: text size and language

Button	Function
A-	Makes all interface text smaller (down to 85 %).
A+	Makes all interface text larger (up to 140 %). The current size is shown between the buttons.
عربي / English	Switches the interface between English and Arabic instantly, without restarting and without stopping the simulation. In Arabic the layout is right-to-left.

- Both settings are remembered on this computer.
- Technical terms (TCP, URDF, PLC, DI/DO), Lua commands, ladder mnemonics and tag names stay in English, as on the real equipment. Ladder diagrams and code always read left to right.
- The footer shows the ASFAN contact details. Clicking **WhatsApp** opens a chat. Clicking the **e-mail** address copies it so you can paste it into your mail program.

15 Keyboard shortcuts

FR5 Digital Twin		PLC Studio	
Ctrl+1 / Ctrl+2	Twin / PLC Studio window	Insert	Instruction picker
Ctrl+B	Hide / show side panel	F5 / F6	NO / NC contact
R	Record point (teach)	Shift+F5	Parallel contact (OR)
Esc	Leave drag teach · cancel input	Ctrl+Enter	New rung
Enter	Apply a typed value	Ctrl+D	Duplicate rung
Ctrl+S	Save program	Enter	Edit selected element
Ctrl+Z / Ctrl+Y	Undo / redo	Del	Delete element
Del	Delete program line	Alt+← / Alt+→	Move element
Alt+↑ / Alt+↓	Move line up / down	Ctrl+Z / Ctrl+Y	Undo / redo

16 Saving, files and troubleshooting

Where is my work saved?

- Programs, points, settings and PLC projects are saved automatically on this computer.
- **File > Open the saved work folder** opens the folder (also shown in **Help > About**). Back it up to keep a copy.
- Use **Export .lua file** and **Export project file** to move work to another PC.

Updates

- The program checks for a new version shortly after start and every 4 hours. When one exists it shows **Update now / Later**.
- **Update now** downloads and verifies the update, closes the program, installs it and reopens the program. Your license and saved work are kept.
- To check yourself, use **Help > Check for updates....** An internet connection is needed.

Troubleshooting

Problem	Solution
The 3D view is empty or shows a WebGL error	Update the graphics driver or enable hardware acceleration, then use View > Reload.
The robot does not move	Check for a SAFETY STOP banner or a running program (manual control is locked). Press Stop, clear the safety stop, then move away from the limit.
Program error	Use Check syntax. The console shows the line and the reason.
PLC and twin not linked	Open both applications on the same PC. Only one PLC Studio window is linked at a time.
PLC outputs stay off	Switch the PLC to RUN. In STOP all outputs and lamps are off.
The window stopped unexpectedly	Choose Reload in the message. Your saved work is kept.
Old interface after an update	Press Ctrl+Shift+R (Force reload).
"Windows protected your PC" when installing	Click More info, then Run anyway.
License window says "Computer date is wrong"	Set the correct date and time in Windows, then reopen the program.

17 Support

For training, support, licences or automation projects, contact ASFAN:



ASFAN

E-mail: info@asfanco.com

WhatsApp: +962 77 614 0404

Credits

- FR5 V6 robot model (URDF, SRDF and meshes): Fairino, from the frcobot_ros package, unmodified.
- Conveyor model: "Simple rubber conveyor" by scailman (sketchfab.com), licensed CC BY 4.0.
- CNC cutting table model: "Pulselaser CNC CO2 IE522".
- Brand names only identify the PLC notations that are modelled. PLC Studio is not vendor software.