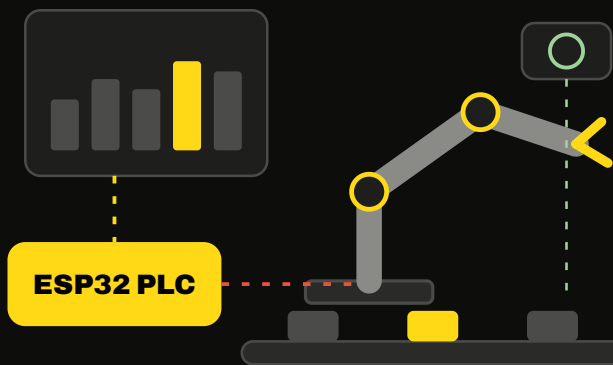


COBOT CELL · ESP32 PLC

One cobot, many references. One PLC to run it.

An ESP32 PLC 38R runs the program of each part and shows the output live.

[Read the interactive case study online ›](#)



Schematic illustration

10 kg

payload of the robotic arm

800 mm

reach of the arm

3

types of grippers, to adapt to each product

4

production figures on the screen: units per minute, sequence time, total and pending parts

THE CHALLENGE

More output from the line, with a robot that works next to people.

The manufacturer makes components, and its main customer is a direct supplier to an automotive assembly plant. It wanted to raise the efficiency of its production lines and cut downtime. Its answer was to invest in collaborative robots (cobots) that work alongside the operators on the lines. Each cobot had to be controlled, adapted to each reference and watched.

01

Many references

The line makes several references. Each one needs its own configuration of the cobot, and three types of grippers adapt the arm to the product.

02

Work next to people

The cobot has sensors that detect when an operator is nearby, and software that slows down or stops its movements.

03

One controller for the cell

The PLC controls the movements of the cobot, receives inputs from sensors and cameras, and talks to the production control system of the line.

04

Production in sight

The operator needs the state of the cobot and the production of each product in real time, on a screen at the cell.



Data for maintenance

The energy consumption is monitored, and the movements of the cobot and of its critical elements, those most likely to wear out, are saved for preventive maintenance.

HOW IT WORKS

Three stages: configure, produce, maintain.

Stage 1 · Configure

Each reference has its own program. The PLC loads it.

The cell makes several references. For each one, the PLC runs a customised program: the configuration of the cobot and the gripper that fits the product. Before the arm moves, the PLC reads its sensors and cameras.

- One program and one configuration per reference
- Three types of grippers for the products
- Inputs from sensors and cameras

Stage 2 · Produce

The screen shows the production of each product, live.

An external touch screen connected to the PLC shows the state of the cobot and the production of the current product: units per minute, time to run the sequence, total parts and pending parts. When an operator comes close, the cobot slows down by itself, and the screen shows the effect on the sequence time.

- State of the cobot in real time
- Units per minute and sequence time
- Total and pending parts of the order

Stage 3 · Maintain

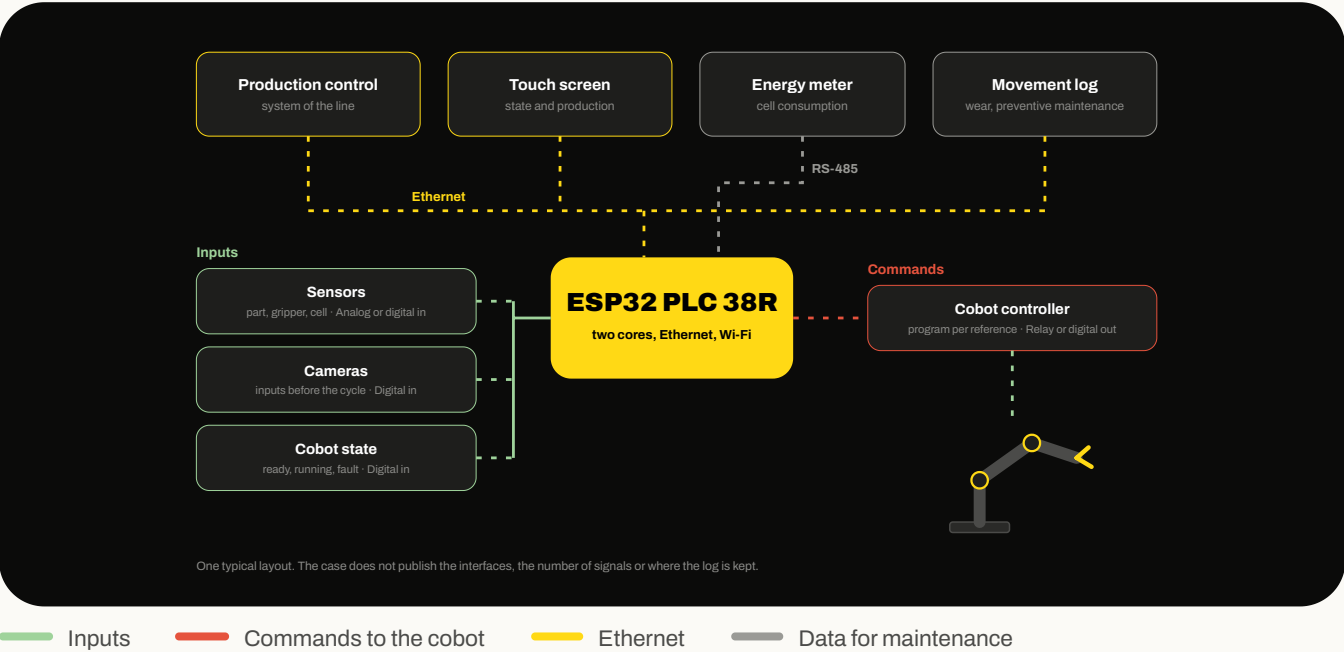
Energy and movements go to a log. Maintenance plans from it.

The PLC monitors the energy consumption of the cell and saves the movements of the cobot and of the elements most likely to wear out. With that log, maintenance can service a part before it fails. A typical way to do it: count the cycles of each wear element and warn before its service interval.

- Energy consumption of the cell
- Movements of the cobot and of the wear elements
- Preventive warnings from the counters

How does an ESP32 PLC control and monitor a cobot?

The ESP32 PLC 38R sits between the cobot and the line. It receives inputs from sensors and cameras and the state of the cobot, and controls the movements of the cobot through the program of each reference. It sends the production data to an external touch screen at the cell and talks to the production control system of the line. It also monitors the energy consumption and keeps a log of movements for maintenance. The case does not publish the interfaces; the diagram shows one typical way to wire them.



Signals and data

Signal	Between	Type	Purpose
Cobot state	Cobot controller → PLC	Digital in (typical)	Ready, running, done or fault, for the sequence and the screen
Program and start	PLC → cobot controller	Relay or digital out (typical)	Runs the program of the current reference
Cell sensors	Sensors → PLC	Analog or digital in	Part present, gripper state and other conditions of the cell
Camera result	Camera → PLC	Digital in or Ethernet (typical)	Inputs from the cameras before and during the cycle
Production data	PLC → touch screen	Ethernet (typical)	State, units per minute, sequence time, total and pending parts
Line data	PLC ↔ production control system	Ethernet (typical)	Reference to make, order size and counts back
Energy	Energy meter → PLC	RS-485, Modbus RTU (typical)	Energy consumption of the cell
Movement log	PLC → log	Ethernet (typical)	Movements and wear counters for preventive maintenance

HARDWARE USED

An ESP32 PLC for the cell. A touch screen beside it.



ESP32 PLC 38R

An industrial controller built on the dual-core ESP32, with Ethernet, Wi-Fi, Bluetooth LE, RS-485 and RS-232, for DIN rail. It is programmed in C++ with the Arduino IDE.

Role	Cobot control, inputs, production data, energy and log
Inputs	12: 8 analog/digital (0-10 V, 11-bit), 4 digital optoisolated (5 to 24 Vdc), 4 with interrupt
Outputs	22: 6 analog/digital/PWM (0-10 V, 12-bit), 16 relays (230 Vac 4 A / 24 Vdc 2 A, resistive)
Communications	Ethernet 10/100, Wi-Fi 2.4 GHz, Bluetooth LE, RS-485, RS-232, I2C, SPI
Processor	ESP32 dual core, up to 240 MHz
Built in	Real-time clock

The case names the ESP32 PLC 38R. Size it for your cell: count the signals from the cobot, the sensors and the cameras, and the outputs you need.

[See the ESP32 PLC 38R ›](#)



TouchBerry Pi Panel PC 10.1" FLEX

For the touch screen at the cell, one current option is an industrial Panel PC with a Raspberry Pi 4 and a capacitive touchscreen, connected to the PLC over Ethernet. It shows the state of the cobot and the production figures of each product.

Role	State of the cobot and production figures
Screen	10.1", capacitive touch
Computer	Raspberry Pi 4
Network	Gigabit Ethernet

The case uses an external touch screen connected to the ESP32 PLC and does not publish its model. The current TouchBerry Pi family has 7", 10.1" and 18.5" models.

[See the TouchBerry Pi 10.1" ›](#)

One program per reference

The configurations match each reference, so a change of product is a change of program, not a new cell.

Two cores, two jobs

The ESP32 has two cores: the process and the communications run apart, so each task runs as it should.

Connected to the line

Ethernet, Wi-Fi, Bluetooth LE and RS-485 link the PLC with the screen, the line system and the meters. It can also send data to cloud services for analysis and remote management.

SIMILAR CELL?

Tell us your robot, your references and what the line needs to see.

Tell us which cobot or robot you use, how many references it must handle, which sensors and cameras the cell has, and where the production data must go. We will tell you which controller fits and how to connect it.

READ IT ONLINE

industrialshields.com/case-study-cobot-automotive-components-esp32-plc

EMAIL

info@industrialshields.com

PHONE

[\(+34\) 938 760 191](tel:+34938760191)