

Production testing automation for open hardware ULX3S board

The Customer

ULX3S FPGA development board is designed and manufactured in Croatia as a result of cooperation between the Radiona.org makerspace, the [Faculty of Electrical Engineering and Computing \(FER\)](http://Faculty of Electrical Engineering and Computing (FER), University of Zagreb), University of Zagreb and [Končar-INEM](http://Koncar-INEM). The ULX3S board was designed by Radiona's member Davor Jadrijević in response to the need for a compact, robust and affordable development FPGA platform equipped with a balanced range of built-in additional components and expansion ports, while fully covering the needs of the "Digital Logic" course at FER. Following the successful Crowd Supply campaign, [Intergalaktik d.o.o.](http://Intergalaktik d.o.o) took over the further production and distribution of ULX3S boards.



The Challenge

For ULX3S boards testing, there was a so-called "self-test" firmware that checked the functionality of sections such as ADC, DAC, switches, keypad, SDRAM, RTC, LEDs, SD card, Flash memory, etc. The test results were displayed on a connected monitor and via the serial port. However, the self-test was used only after the PCBA contractor delivered the finished ULX3S boards, so all subsequently found problems could not be corrected immediately in production, but defective boards had to be returned for repair, which slowed down the production cycle and made it more expensive.

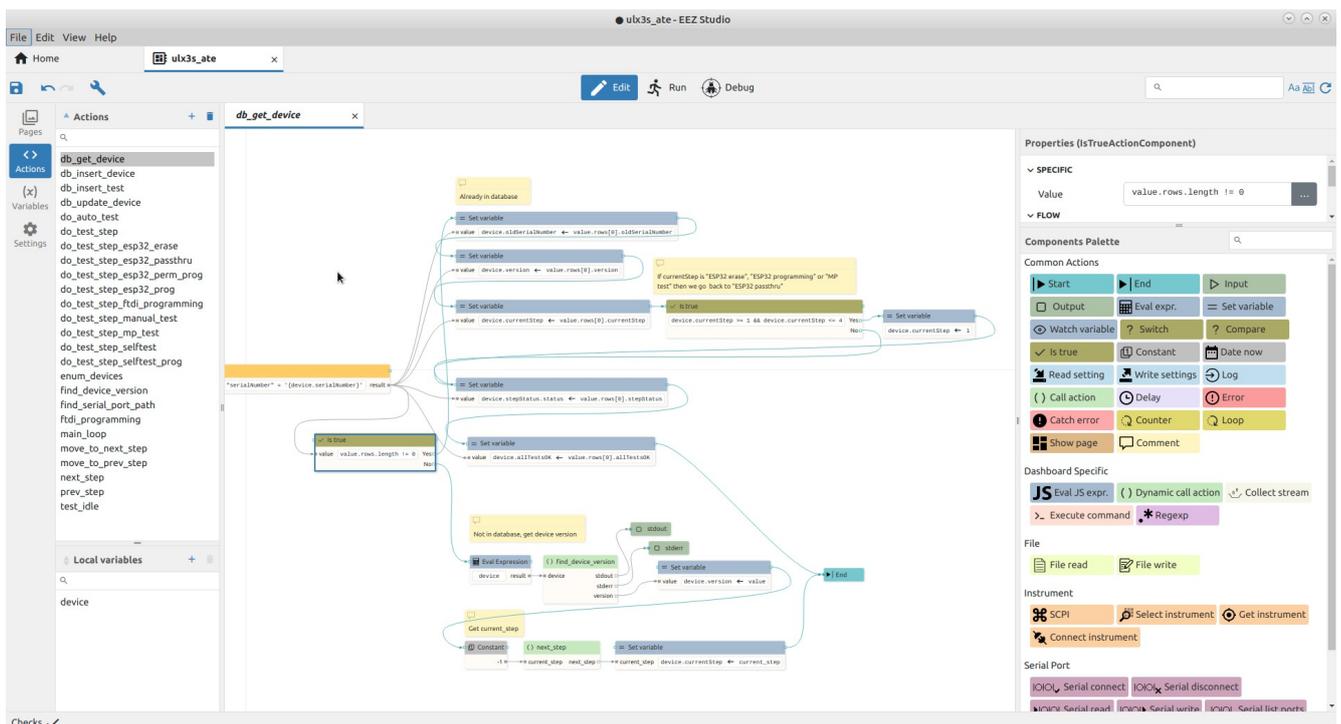
The use of the self-test procedure in production is a problem because it requires advanced knowledge of the ULX3S board, as well as certain preparatory actions. Additionally, even if the production tester is adequately trained, the self-test command line interface leaves more room for human error which increases the chance that the boards will not be properly tested.

A simpler, GUI-based test procedure is preferred with storing all relevant data and test results in one place in order to speed up the whole process and reduce the possibility of error due to the human factor.

The Solution

Selected EEZ Studio is an open source cross-platform (Linux, Mac, Windows) low-code development tool that enables programming and management of Test & Measurement (T&M) equipment using various interfaces and standardized SCPI commands.

Recently added flow editor, offers quick creation of test scenarios using flowchart like visual programming (i.e. *flow*).



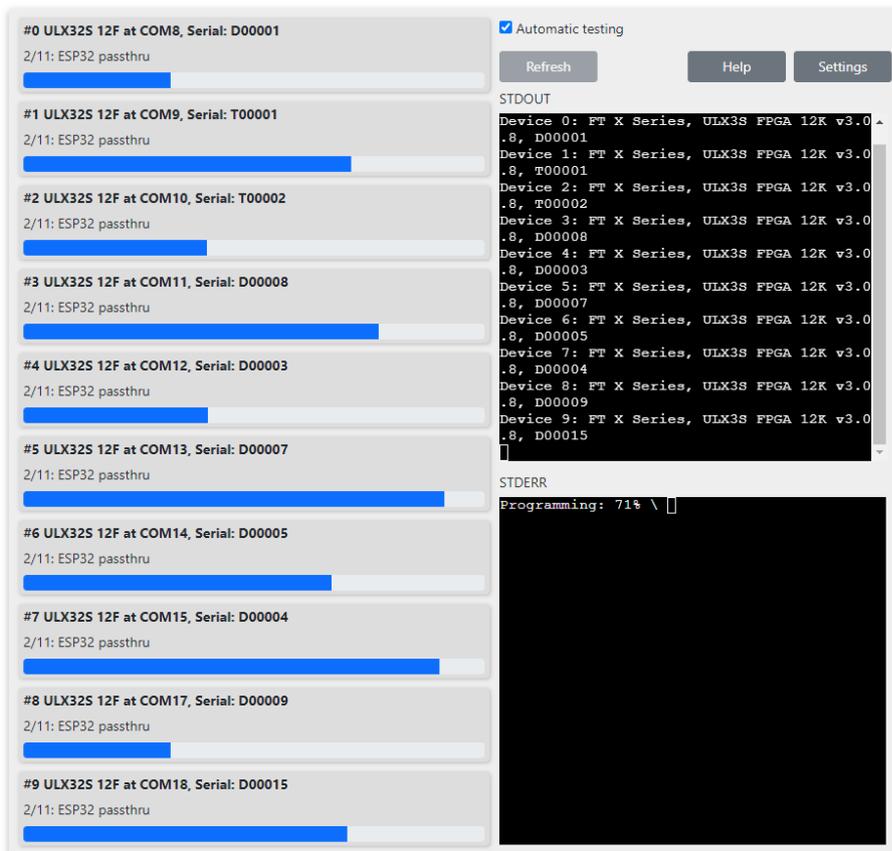
A fully featured visual debugger further shortens development time. The created *flow* takes care of the entire test procedure in 11 steps:

- FTDI programming
- ESP32 “passthru”
- ESP32 erasing
- ESP32 programming
- MicroPython testing
- Self-test programming
- Self-test execution
- Manual test LED’s
- Manual test display output
- ESP32 “passthru” flash
- Manual test US2

It is possible to test up to 10 boards simultaneously connected to a USB hub. The test procedure assigns a unique serial number to each ULX3S board, which is stored along with information about each test step in the PostgreSQL database for further analysis and debugging.

Furthermore, since manual interventions are required during the test (e.g. pressing all the buttons), the *flow* will also inform the tester when to do so. To better monitor the overall procedure, two text terminals (STDOUT and STDERR) have also been implemented.

The *flow* used for this project is 100% open source and is available on [GitHub](#) as well as EEZ Studio.



The Project Rollout

“The main advantages of using EEZ Studio is the simple way creation of an understandable GUI (i.e. dashboard) on the frontend side, and at the same time to create/debug easy to follow complete graphical flow on the backend” said Goran Mahovlić, CEO of Intergalaktik d.o.o. “By looking at a graphical representation of the flow, it is easier to get the whole picture of the test than by reading the script.”

“Another important advantage is that it is scalable, as from initial test case we can very easily extend the testing to get for example, speed grade sorting of products. Seamless integration with the database allowed us to store all test data and logs. Finally, it is completely open source so even adding functionality that are missing is also possible and feasible.”

The implementation highlights

- Discovery (enumeration) of devices connected to a USB hub
- Parallel testing of up to 10 boards
- Automatic/manual mode
- Boards registration data and test results stored in the PostgreSQL database

Action components:

- Serial port input/output communication (via USB)
- Execution of CLI commands with output and error processing
- RegExp based parsing
- PostgreSQL database
- XML document load, modification and save

Widgets:

- Terminal
- Markup document viewer (Help page)
- Various standard widgets (e.g. buttons, checkboxes, input fields, etc.)

Business results

Testing automation has enabled faster testing, more reliable ULX3S boards data collection as well as reduced testing costs. The boards delivered by the PCBA contractor do not require additional testing, which shortened the delivery time to end customers. Created EEZ Studio *flow* can be easily upgraded and modified to test future versions of ULX3S but also new [ULX4M](#).

