linaro™ Arm Solutions at Lightspeed

Testing ESPHome in the real world

Rémi Duraffort
Principal Tech Lead
Linaro

remi.duraffort@linaro.org

Who Am I?

- Rémi Duraffort
- Principal Tech Lead at Linaro

- OSS developer since 2007
  - VLC mdia player
  - v8 js engine
  - PRoot/CARE
  - LAVA, lavacli, meta-lava, lavafed, …
  - TuxRun, TuxSuite
  - KissCache
  - …

An home automation fairy tale

An home automation fairy tale

- Once upon a time .. [...] ... They lived happily ever after
  - In a large castle
  - With a lot of rooms & lights
    - And a large energy bill

An home automation fairy tale

- They decided to add home automation to their castle
  - One server
  - Multiple sensors & actuators

- They know DRAGons (Dropcam, Ring, Alexa, Google Home) are dangerous creatures.

An home automation fairy tale

- They wanted
  - OSS software
  - Not connected to any proprietary cloud
    - Installed locally in their castle network
      - They have ethernet in every room!

- They went with **Home Assistant** (for the server)

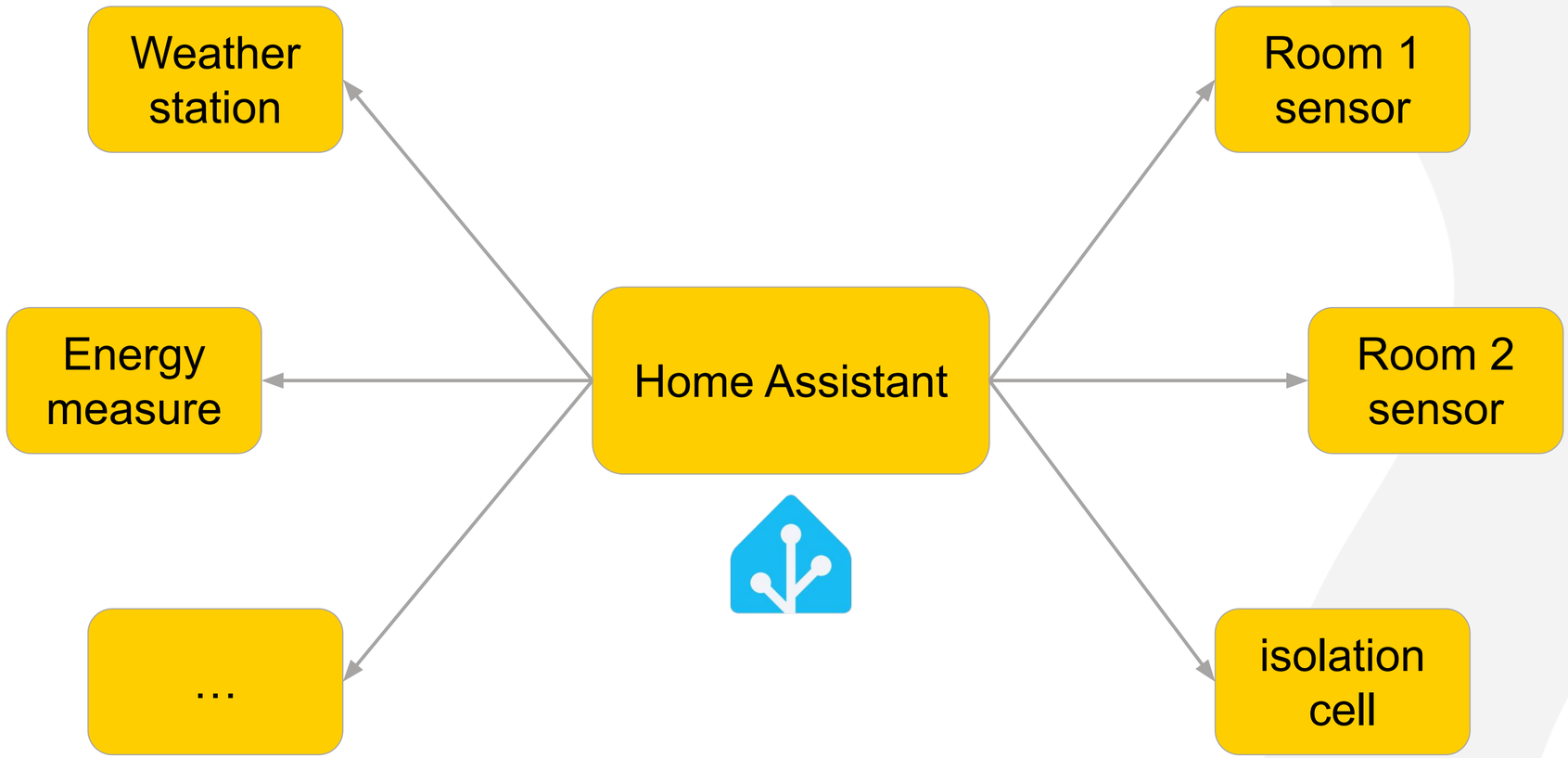
Home Assistant

Open source home automation

- Gather data from sensors in your house
  - Temperature, humidity, energy consumption, …
- Control devices
  - lamps, blinds, dungeons doors, drawbridge, …
- Create automation
  - If the drawbridge opens during the night … you are in trouble
- Show the data in dashboards
  - Data coming from sensors

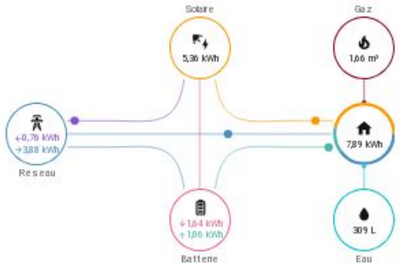
Your new castle

HA and ESPHome



Home Assistant

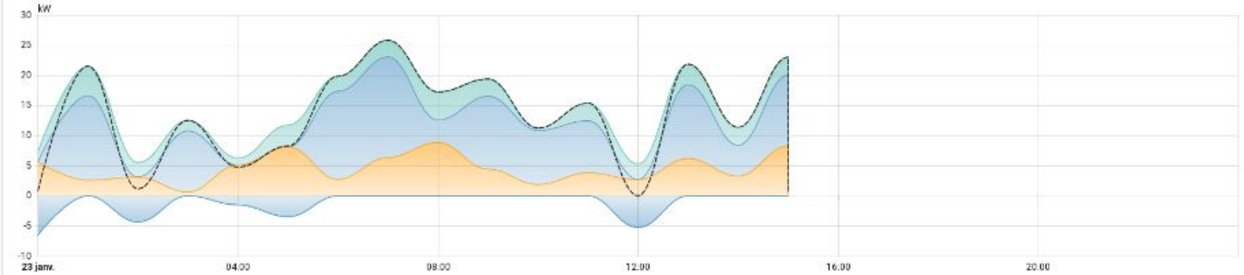
Distribution d'énergie



Sources

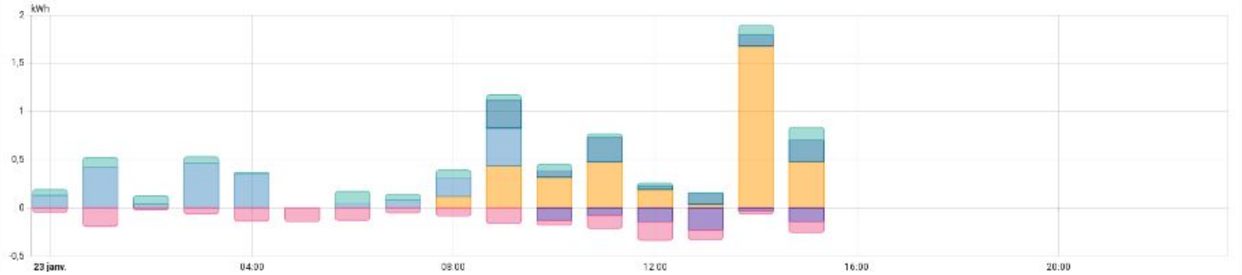
Source	Usage	Coût
Total production solaire	5,36 kWh	
Batterie totale	-0,58 kWh	
Total du réseau	3,12 kWh	0,63 $US
Total gaz	1,66 m³	0,65 $US
Total de la consommation d'eau	308,85 L	0,00 $US
Coûts totaux		1,28 $US

Sources d'alimentation



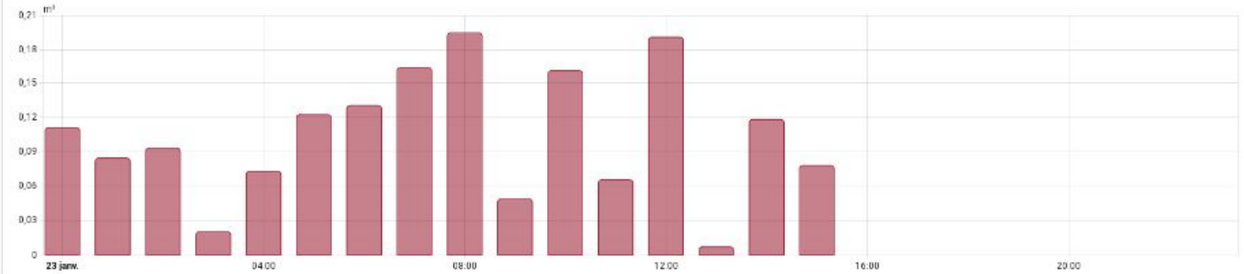
Consommation d'électricité

+7,89 kWh



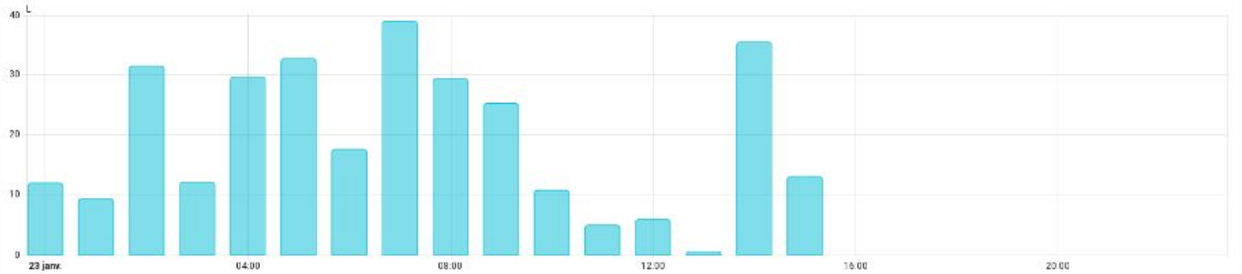
Consommation de gaz

1,66 m³

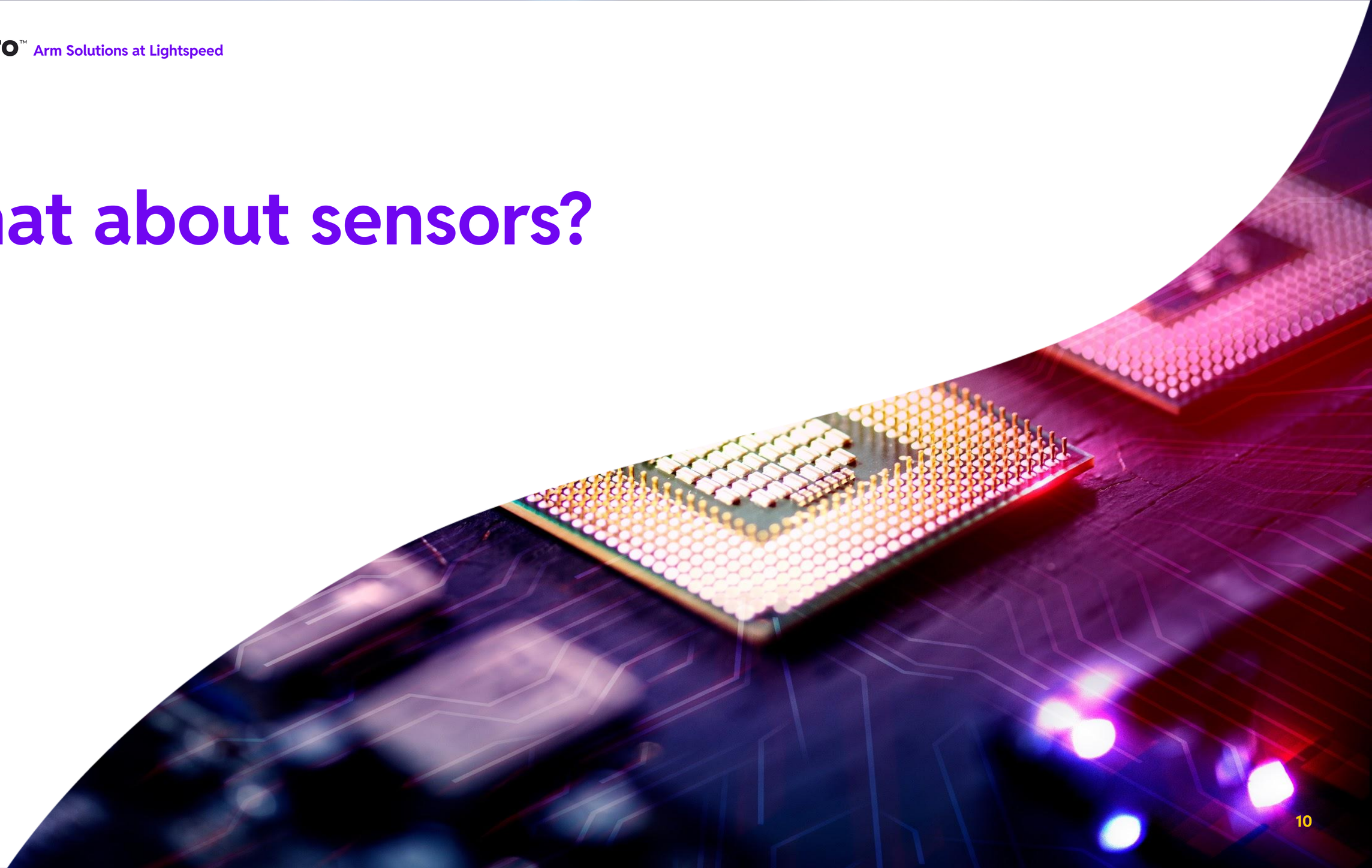


Consommation d'eau

308,85 L



What about sensors?



An home automation fairy tale

- They wanted
  - Cheap and customizable hardware
  - Wide support for sensors and actuators

- Well known brands

Espressif	ESP32 & ESP8266
Raspberry Pi	RP2040
Nordic Semiconductor	nRF52
Realtek	RTL87xx

An home automation fairy tale

- What about the firmware?
- They wanted
  - OSS software
  - Customizable
- But … they don’t know how to program
  - Remember: they are fairy tale characters!

ESPHome

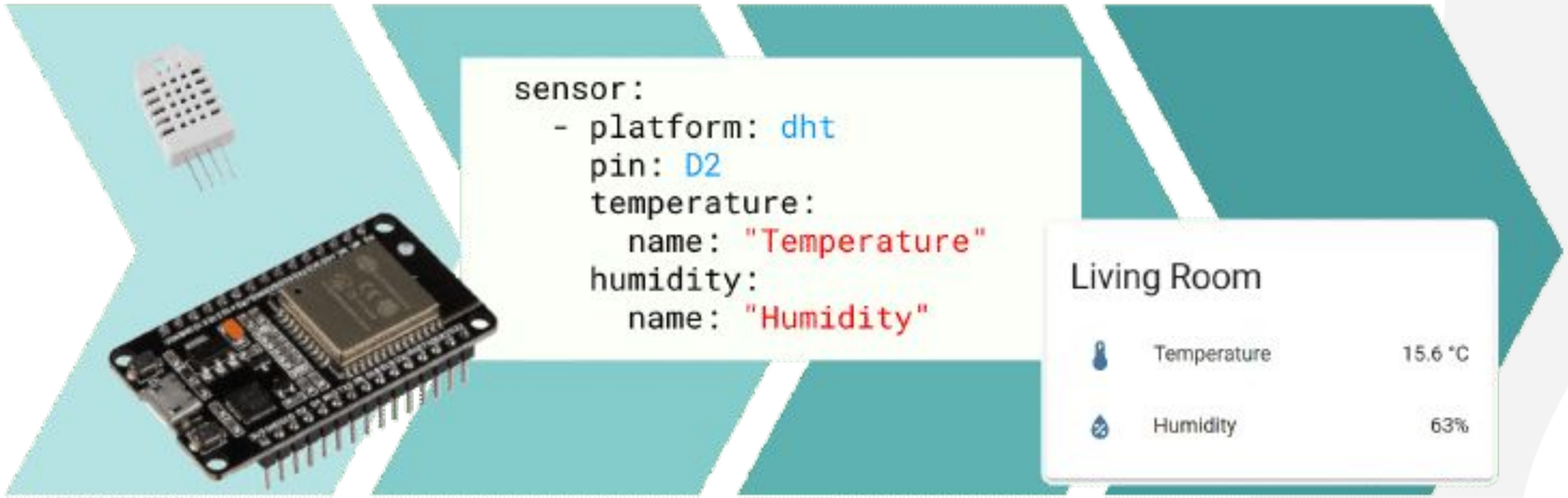
“Turn your ESP32, ESP8266, or RP2040 boards into powerful smart home devices with simple YAML configuration”

ESPHome

esphome.io

A tool to define and build board (ESP32, RPI2040, ...) firmwares

1. Describe the hardware in yaml
2. ESPHome generates the firmware



ESPHome configuration

- Firmware configuration
  - d1 mini (ESP8266)
  - Connect to wifi
  - AM2320 (temperature and humidity)
- Build

```
$ esphome compile config.yaml
```

- Flash

```
$ esphome upload --device /dev/ttyUSB0 config.yaml
```

Next update will use wifi (no more serial)

```yaml
esphome:
  name: livingroom_sensor
  platform: ESP8266
  board: d1_mini

wifi:
  ssid: "MyWiFi"
  password: "MyPassword"

ota:
  - platform: esphome
    password: !secret ota_password

sensor:
  - platform: am2320
    temperature:
      name: "Living Room Temperature"
    humidity:
      name: "Living Room Humidity"
    update_interval: 60s
```

An home automation fairy tale

“Everything Is AWESOME!!!“

The lego movie

“Until the next release”

The real world

Testing ESPHome

Testing ESPHome

How to ensure next version is still working well?

- Continuous Integration

Currently

- Build for a large set of platform
- Test interfaces/integrations
- but … do not run any test on real hardware

- Test ESPHome dev branch
  - On real hardware
  - Test the real world
    - need a sensor
    - a way to control the sensor value

Testing ESPHome

Board: wt32-eth01

I already have one

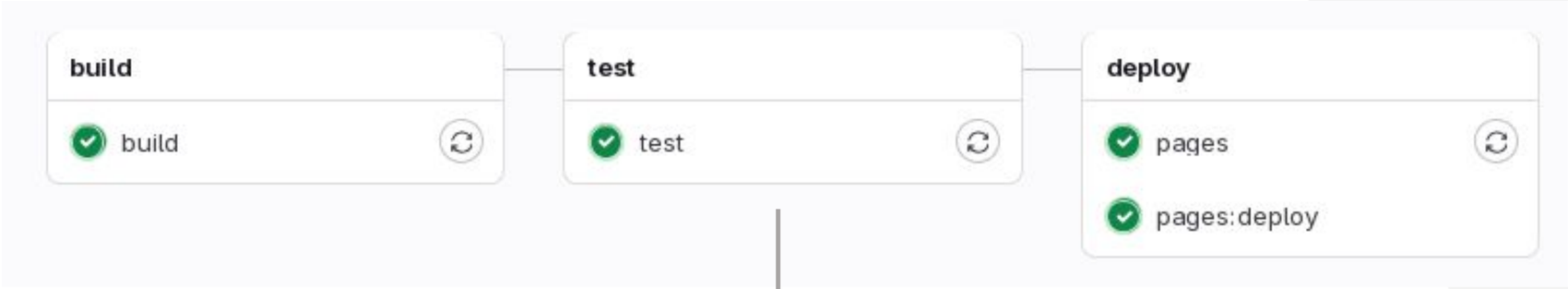
Sensor: BH1750

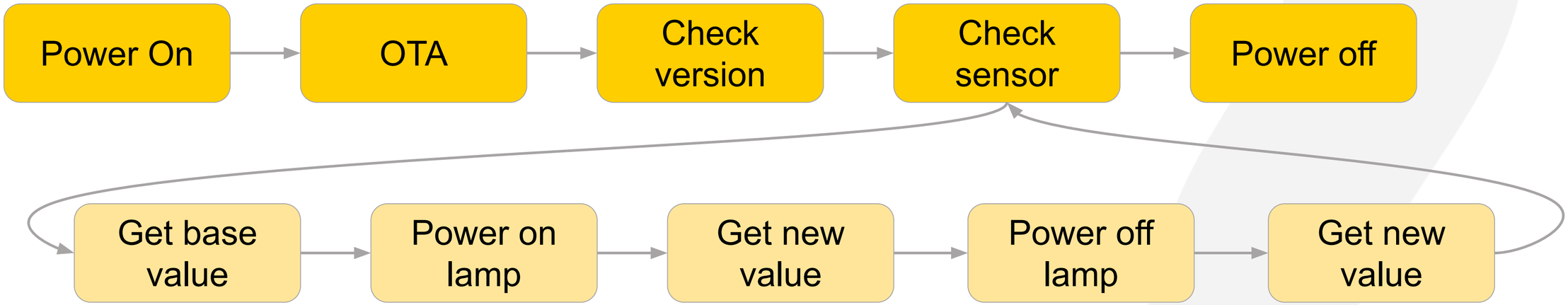
USB lamp: Micro LED USB

Powering on/off the lamp will change the sensor value

Gitlab pipeline

Test scenario





Board automation

Using LAVA on the LAA

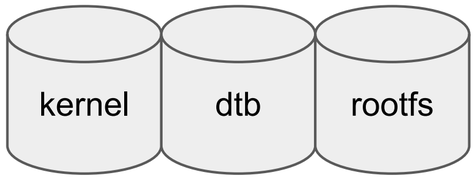
- **L**inaro **A**utomation **V**alidation **A**rchitecture
  - Automatically test your software on your hardware
  - Testing from bootloaders to OS or user-space applications

- Principles
  - Define how LAVA can connect to the board (serial, power commands, …)
  - Define the job as a yaml file: deploy, boot, test, …
  - LAVA run the tests for you

- Used by many projects
  - KernelCI: testing the Linux kernel
  - Civil Infrastructure Platform: Industrial Grade Linux
  - …

Board automation with LAVA

Testing with LAVA

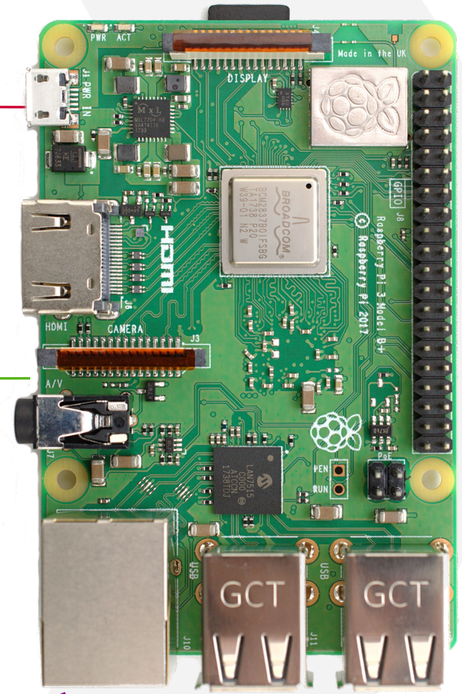


```
% power on board
% telnet localhost 2000
<enter>
=> dhcp
=> setenv serverip 10.3.1.1
=> […]
=> bootm 0x01000000 - 0x03f00000
[…]
raspberrypi3 login: root
# run-test.sh
[…]
% power off board
```

Power control

Serial relay

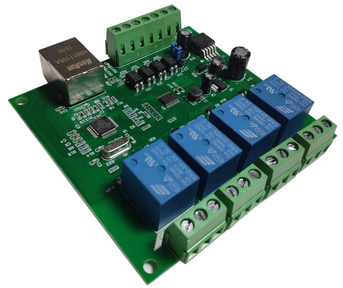
tftp&nfs server



Board automation

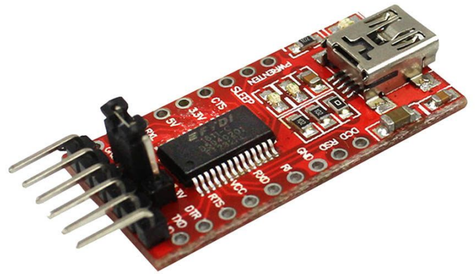
- Control the 5V power
  - Power on/off the board
- Control the USB power
  - Power on/off the lamp

→ Managed power socket and USB hub
  - Use web relays boards (ok for low voltage)



- Access to the serial
  - Flashing
  - Logging

→ serial via FTDI

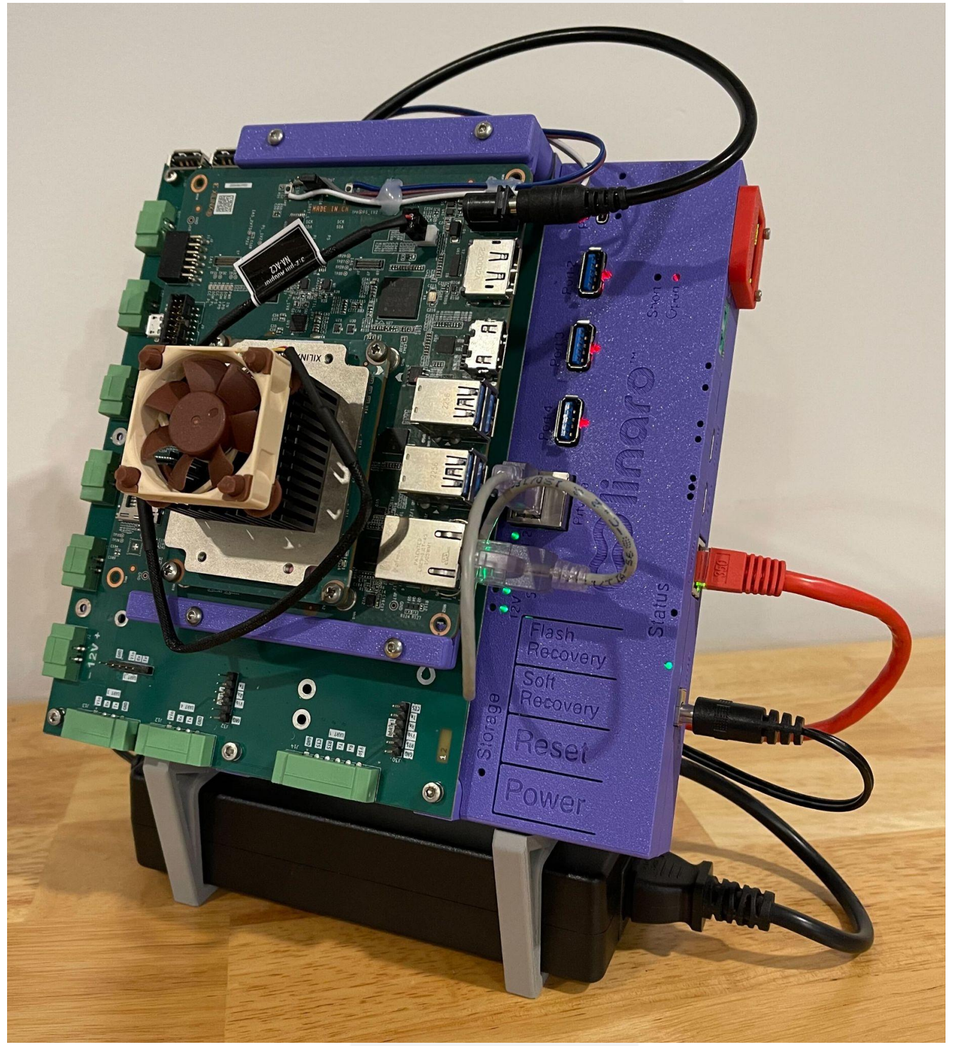


- Ethernet access (with dhcp)

→ switch and dhcp server

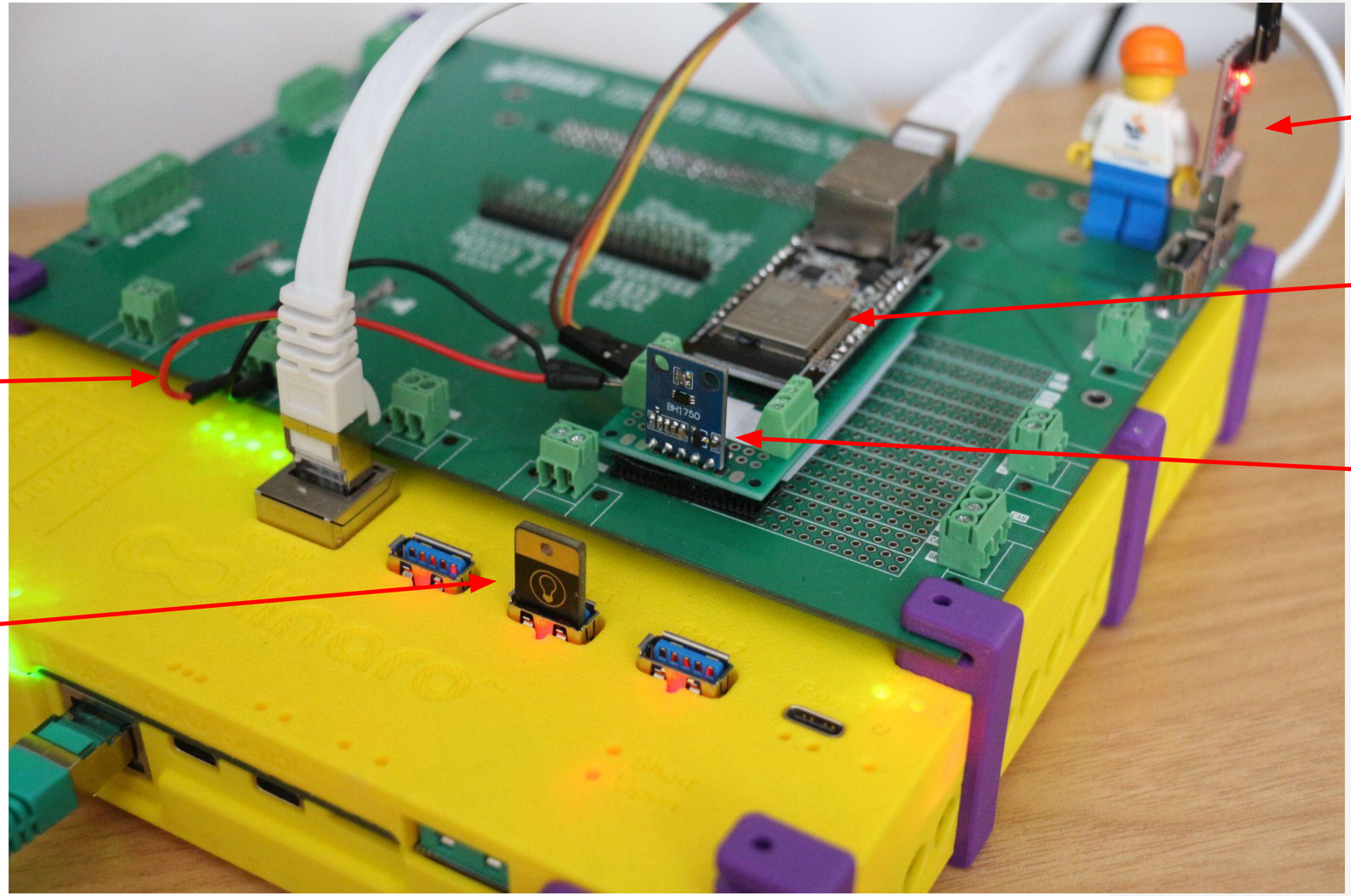
Linaro Automation Appliance

- All in one automation appliance
  - **Managed power rails:** 1v8, 3v3, **5v** and 12v
    - With power measurements
  - Integrated serials
  - **4 managed USB ports**
  - USB Mass storage emulation
  - Virtual buttons (GPIOs, Optocouplers)
  - Thermal probe
  - CAN, SPI, I^2C
  - 2 ethernet ports
    - **1 private ethernet** for the DUT
      - Fully isolated network for the DUT



Board automation

Using the LAA



Board automation

LAVA device configuration

```
{% set connection_command = 'tio /dev/ttyUSB0' %}

{% set hard_reset_command = [
    'laacli power 5v reset',
] %}

{% set power_on_command = [
    'laacli power 5v on',
] %}

{% set power_off_command = [
    'laacli power 5v off',
] %}
```

Board automation

LAVA job definition

```
actions:
- boot:
    method: minimal
    prompts:
      - "IP Address: 198.18.0.2"

- test:
    [...]
      steps:
        - uvx esphome upload --device 198.18.0.2 --file firmware.ota.bin testing-light-sensor.yaml

- boot:
    method: minimal
    reset: false
    prompts:
      - "IP Address: 198.18.0.2"

- test:
    [...]
      steps:
        - lava-test-case firmware-version --shell ./check-version {ESPHOME_VERSION}
        - lava-test-case lux-sensor --shell ./check-sensor 3
```

1/ Boot and wait for IP

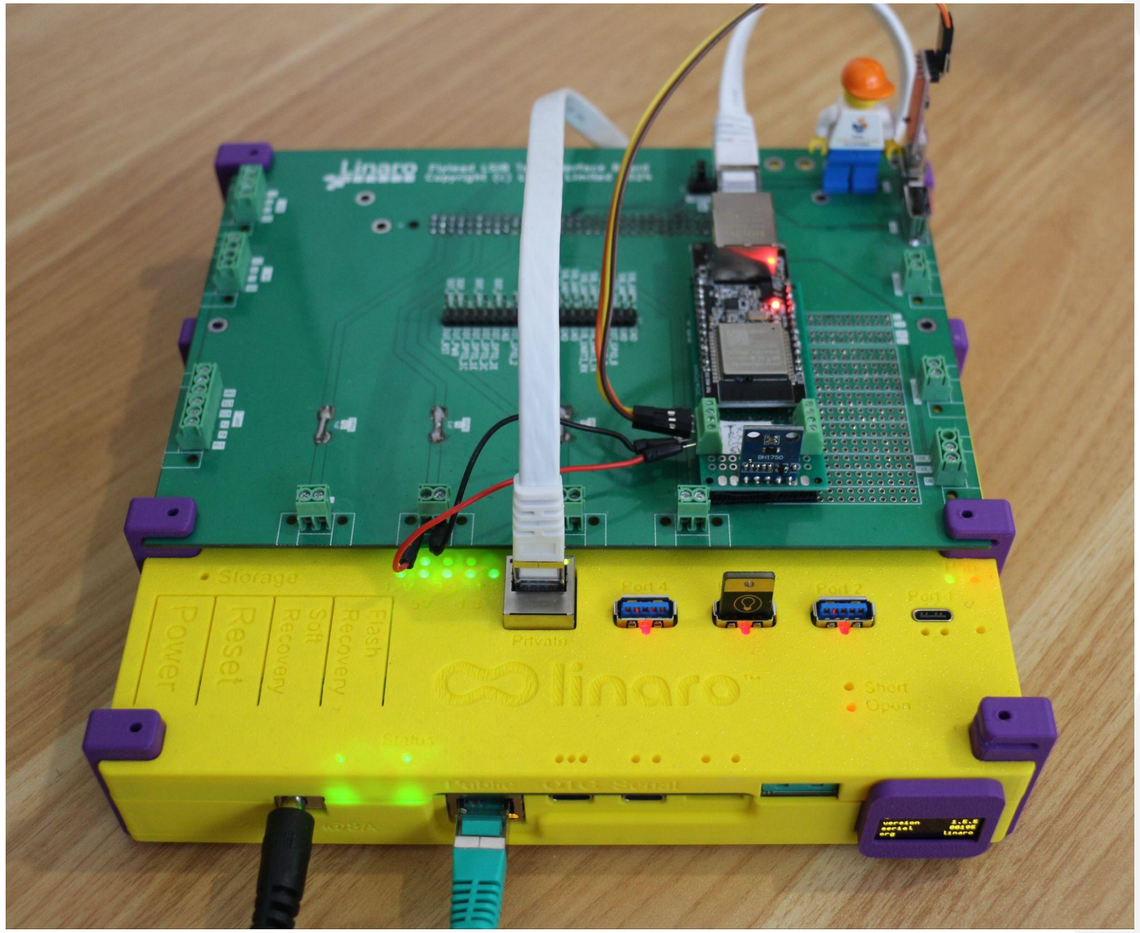
2/ Push OTA and reboot

3/ Wait for reboot and IP

4/ Check version and sensor

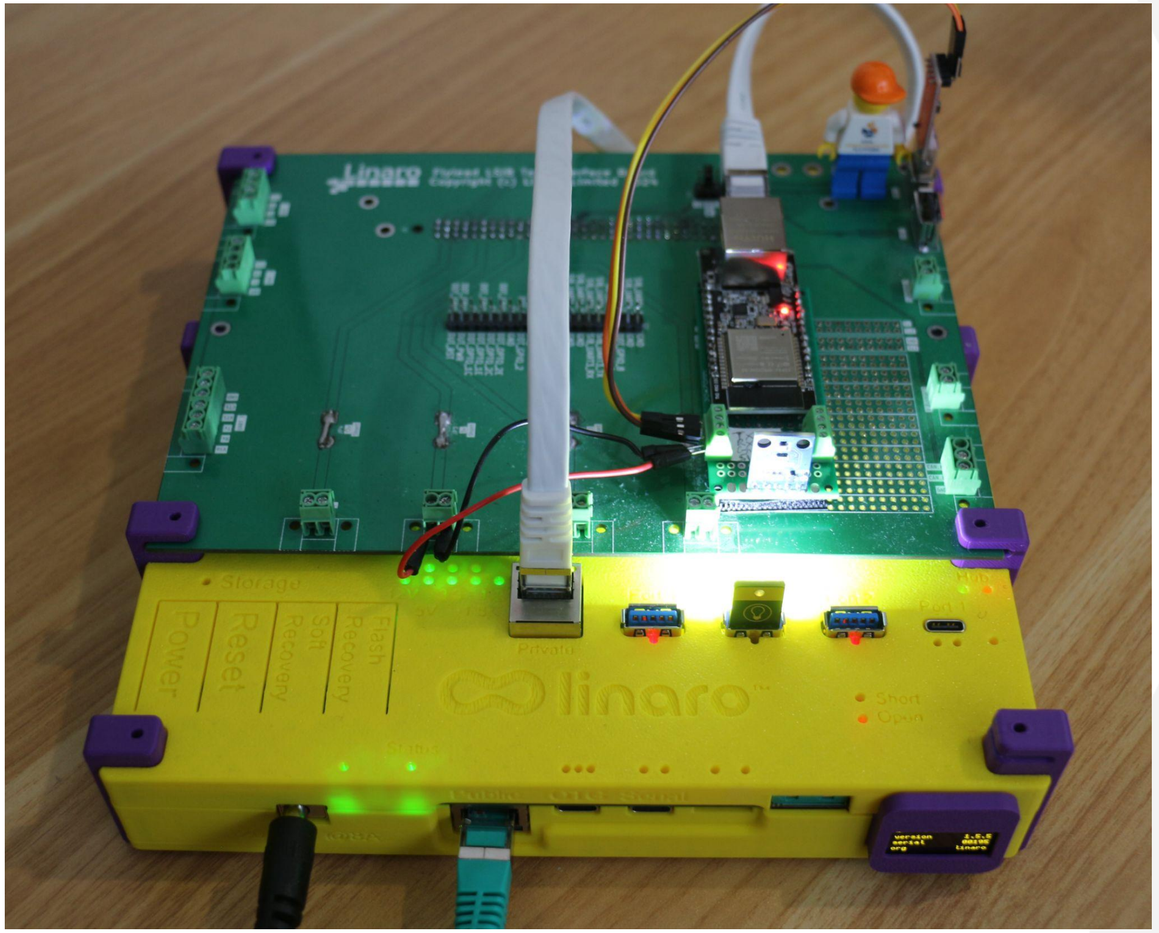
Testing ESPHome

Using LAVA on the LAA



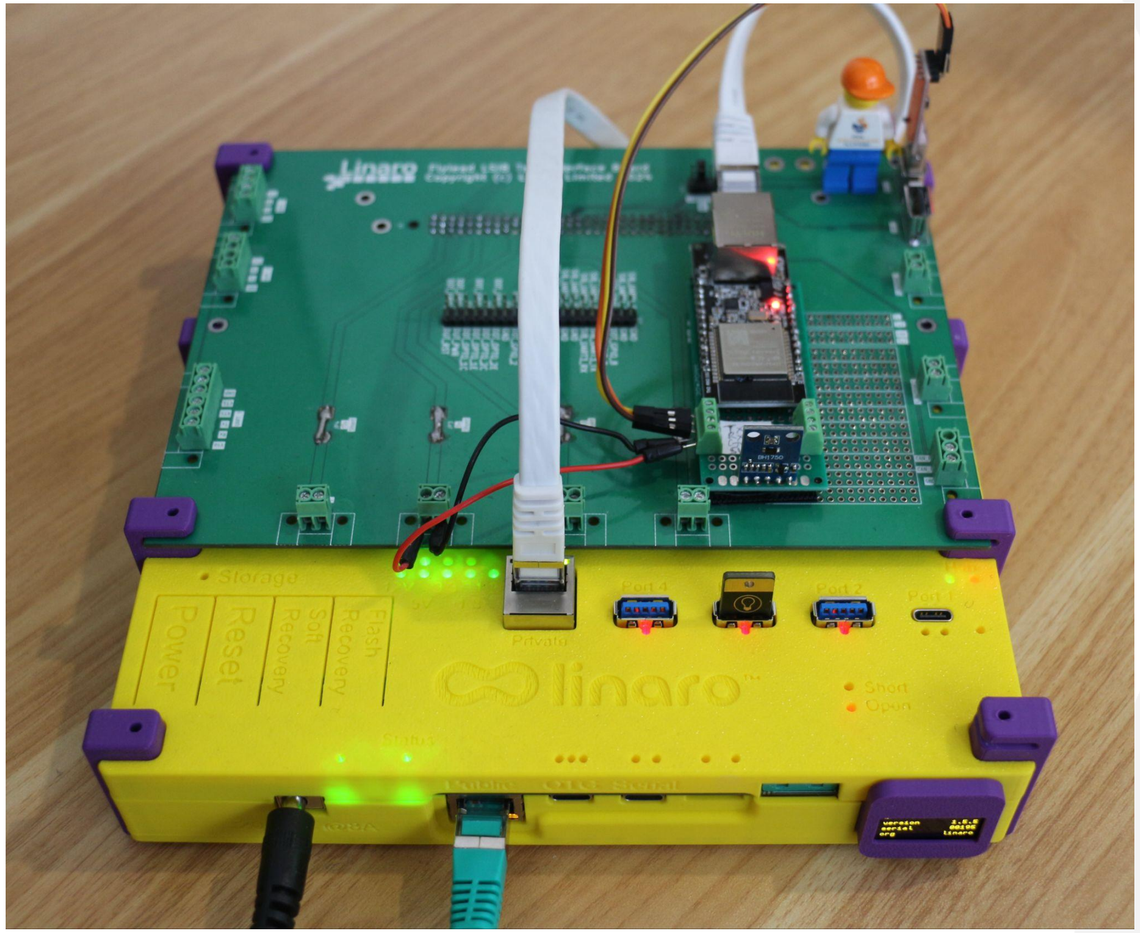
Testing ESPHome

Using LAVA on the LAA



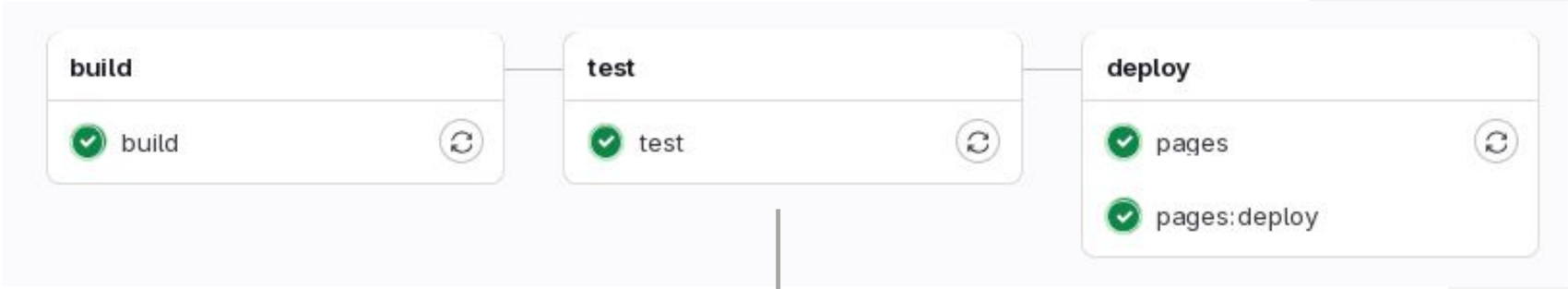
Testing ESPHome

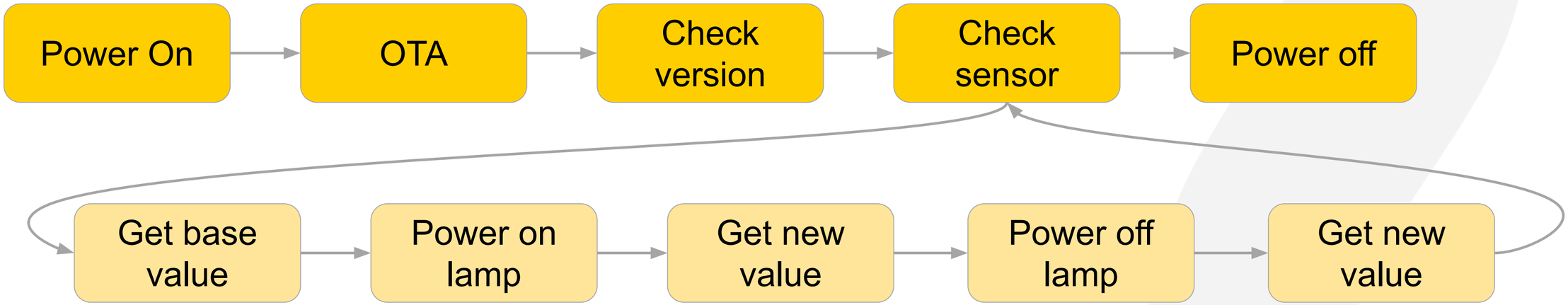
Using LAVA on the LAA



Gitlab pipeline

Test scenario





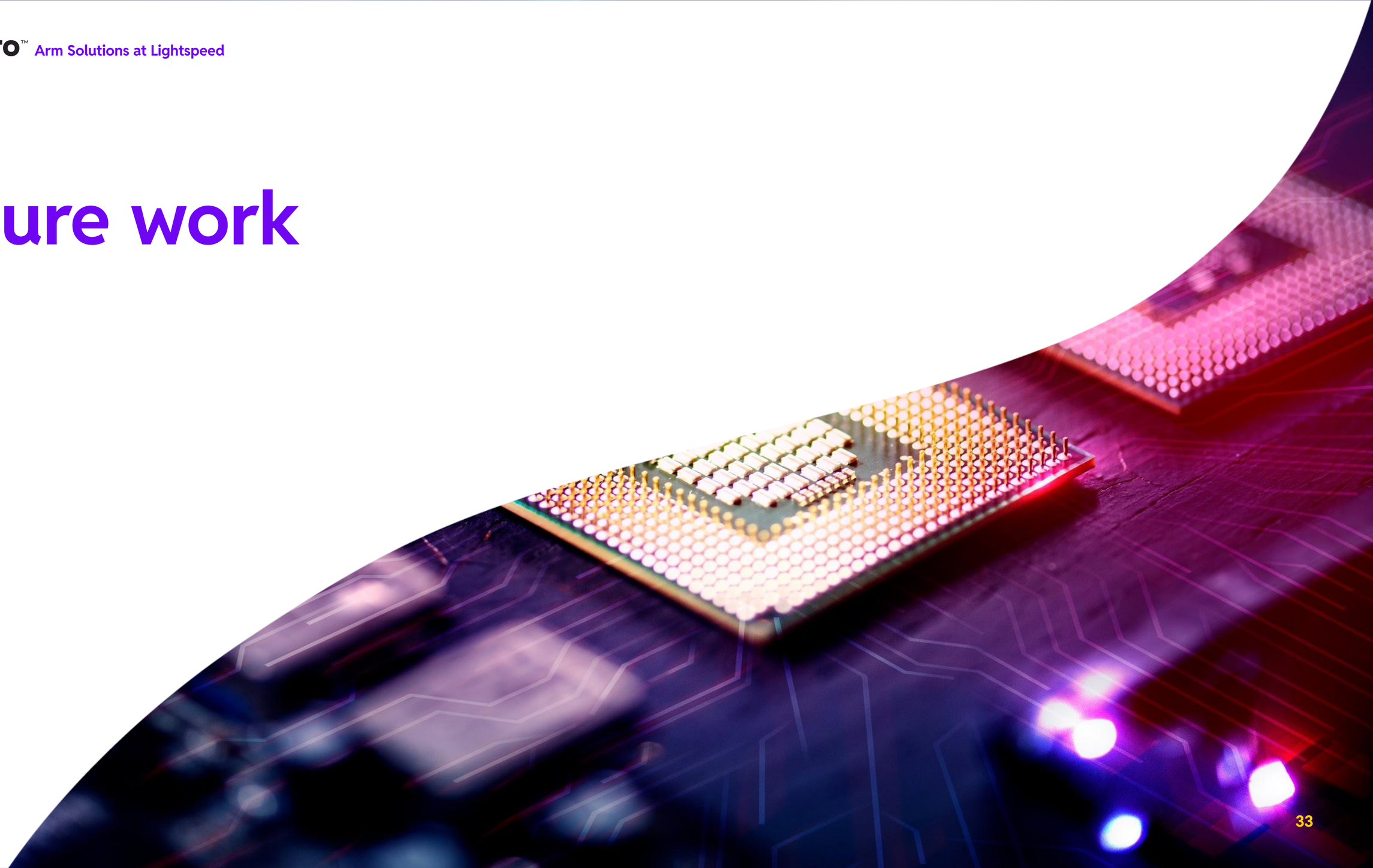
Results dashboard

See esphome-ci.lavacloud.io

Version	Commit	Tests results		
		Testing	OTA	Lux Sensor
2026.1.1-0179-gcfb61bc50	cfb61bc50	✓ 🔗	✓	✓
2026.1.1	44e624d7a	✓ 🔗	✓	✓
2026.1.0-0162-gddb762f8f	ddb762f8f	✓ 🔗	✓	✓
2026.1.0-0158-g5bbf9153c	5bbf9153c	✓ 🔗	✓	✓
2026.1.0-0138-g37eaf10f7	37eaf10f7	✓ 🔗	✓	✓
2026.1.0b3-0146-g0b60fd0c8	0b60fd0c8	✓ 🔗	✓	✓
2026.1.0b3-0116-g79ccacd6d	79ccacd6d	✓ 🔗	✓	✓
2026.1.0b3-0115-ge2319ba65	e2319ba65	✓ 🔗	✓	✓

Just™ a gitlab page

Future work



Future work

- Test OTA from/to released version
  - Fash the latest release
  - OTA to dev build
  - Check version and sensor
  - Rollback to latest release
- Test with more sensors
- Reach out to ESPHome developers

Links

- ESPHome: esphome.io
- Home Assistant: home-assistant.io

- LAVA: gitlab.com/lava/lava
- Linaro Automation Appliance: docs.lavacloud.io

- CI results: esphome-ci.lavacloud.io
  - CI configuration: gitlab.com/ivoire/esphome-ci
  - Pipelines: gitlab.com/ivoire/esphome

Thank you

remi.duraffort@linaro.org