
instantiations

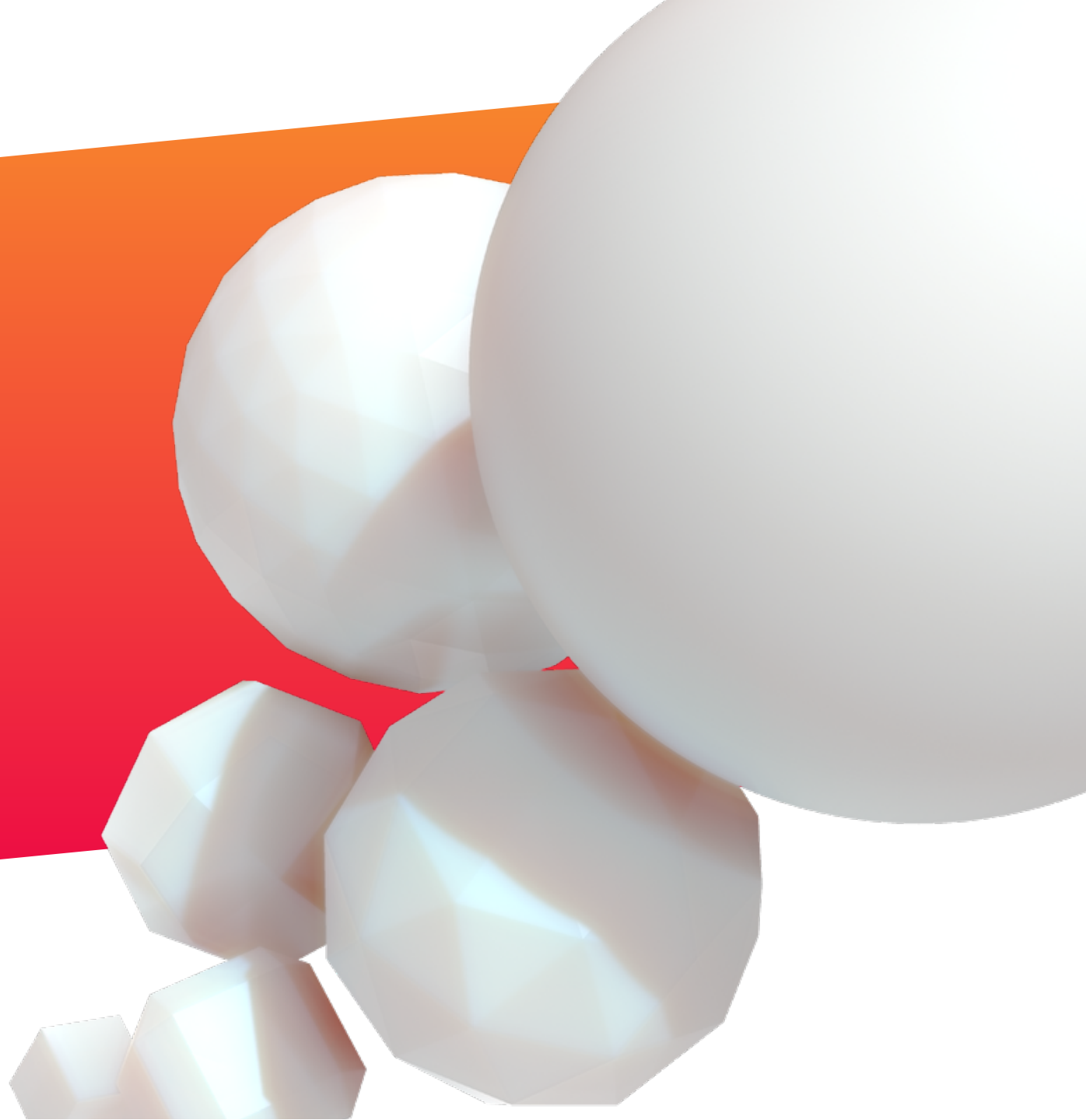
The IoT development experience can be better.

instantiations

instantiations

Taking IoT Programming to a Higher Level

with Instantiations and Toit



Seth Berman

President & CEO

instantiations

Kasper Lund

Co-Founder & CEO

Toit

Mariano Martinez Peck

Senior Software Engineer

instantiations

Instantiations, Inc.

- Founded in 1988 – Headquartered in Raleigh, NC
- Developer of the **VAST Platform**
  - A comprehensive integrated development environment (IDE) that works with a Smalltalk language runtime
  - Advancing both the IDE and Virtual Machine (VM)
- Provider of technology services
  - Custom software development
  - Consulting
  - Training
  - IoT
- Our software and services are leveraged by
  - Fortune 100/500 companies
  - Small and medium-sized businesses
  - Solo developers/entrepreneurs

instantiations

Why is a higher-level development experience better?

instantiations

Advantages of higher-level languages

- Deal with high-level abstractions (objects, functions, prototypes)
- More productive
- Memory-safe
- Easier to debug
- More human readable

VAST Platform

- A comprehensive integrated development environment (IDE) that works with a Smalltalk language runtime
- For Windows® and Linux® operating systems
- Intel®/AMD® and ARM® processor support (32/64-bit)
- Purely object-oriented
- Extremely productive and performant
- Almost 30 years of enterprise-proven stability
- Can be used for IoT orchestration or edge devices

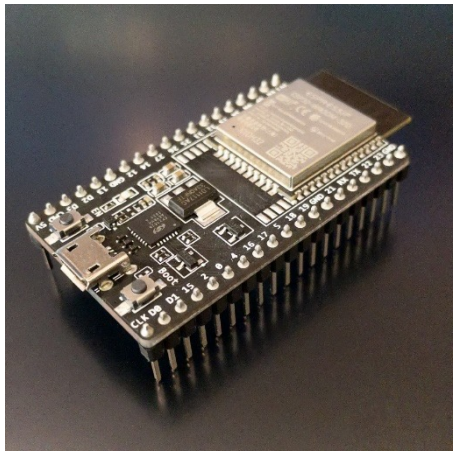
VAST Platform

What about microcontrollers?

Toit: The Programming Language

- An open-source, object-oriented language purpose-built for IoT programming
- Uses open-source hardware (ESP32) that's small and energy efficient, and available from dozens of reputable sources
- 20x faster than MicroPython

Toit



ESP32 Development Board

instantiations

Toit

with Kasper Lund, co-founder and CEO

instantiations

Kasper Lund, co-founder and CEO of Toit

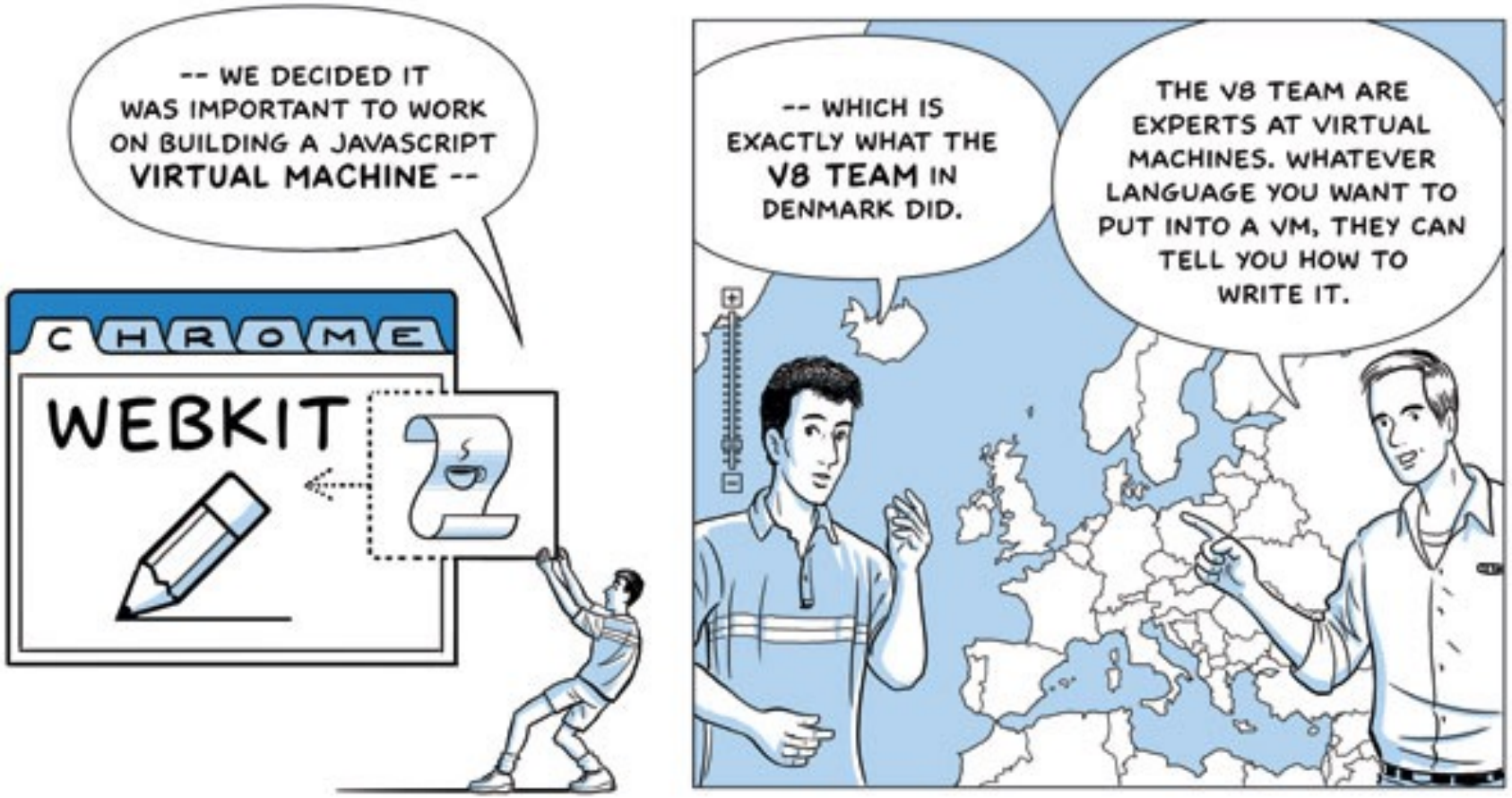
instantiations

I joined Google in 2006 to build something new …

instantiations

instantiations

I am located in Aarhus, Denmark



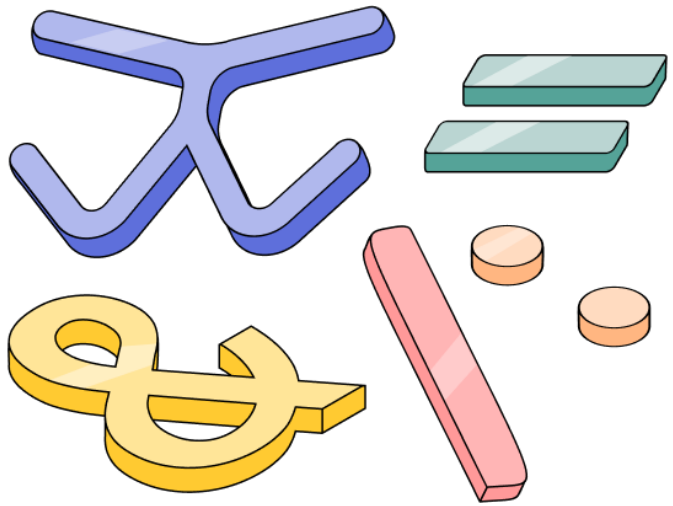
I have a deeply technical background

Flutter

I am driven by enabling others to be more productive

Virtual machines

Operating systems



Developer tooling

Programming languages

instantiations

I ❤️ Smalltalk!

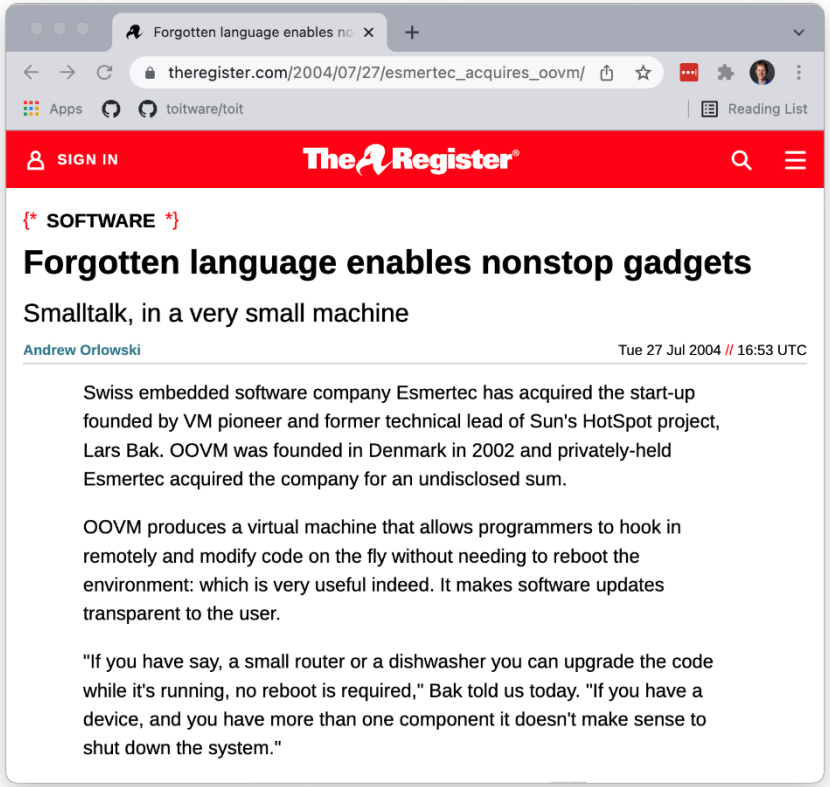
Revolutionizing Embedded Software

Kasper V. Lund and Jakob R. Andersen

Master's Thesis

Department of Computer Science
University of Aarhus
Denmark

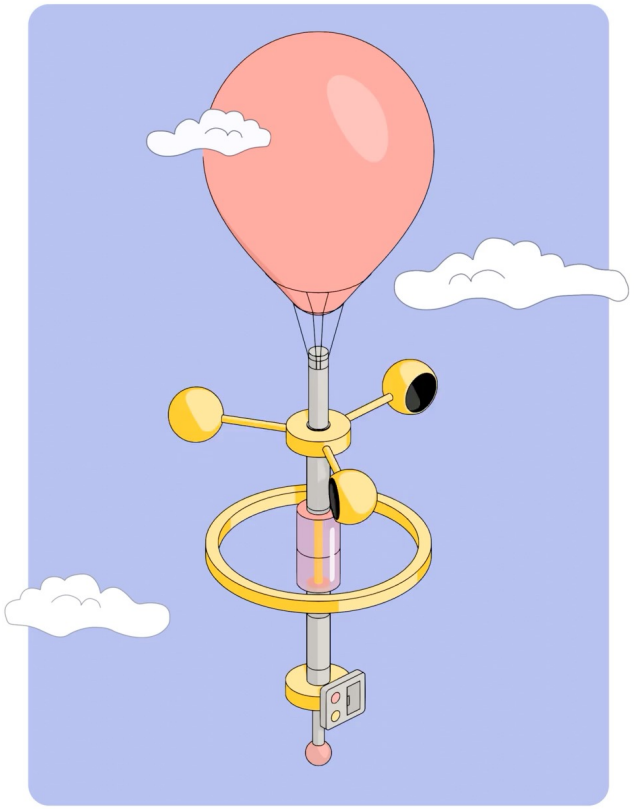
https://verdich.dk/kasper/RES.pdf



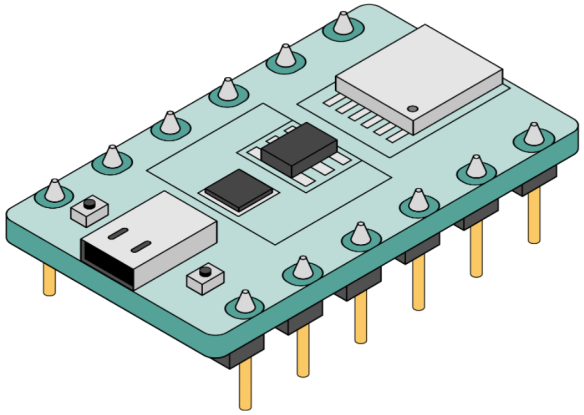
instantiations

I am excited to be presenting with Instantiations

- Worked with fantastic ex-Instantiations folks at Google
- Indirectly collaborated in the Dart and Flutter ecosystems
- Shared passion for developer experience and languages



I learned of the ESP32 back in 2018 ...



Powerful

Dual-core 240 MHz RISC CPU
520 KB RAM, 4MB+ Flash
Built in WiFi / Bluetooth

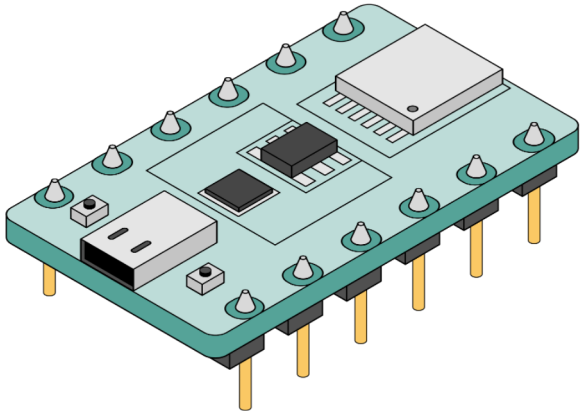
Runs on batteries

Practical drain in sleep mode is ~10 uA
Runs for years on AA batteries

Inexpensive

Development kit costs $10
Standalone chip costs less than $2

For a lot of interesting use cases, this is a ...



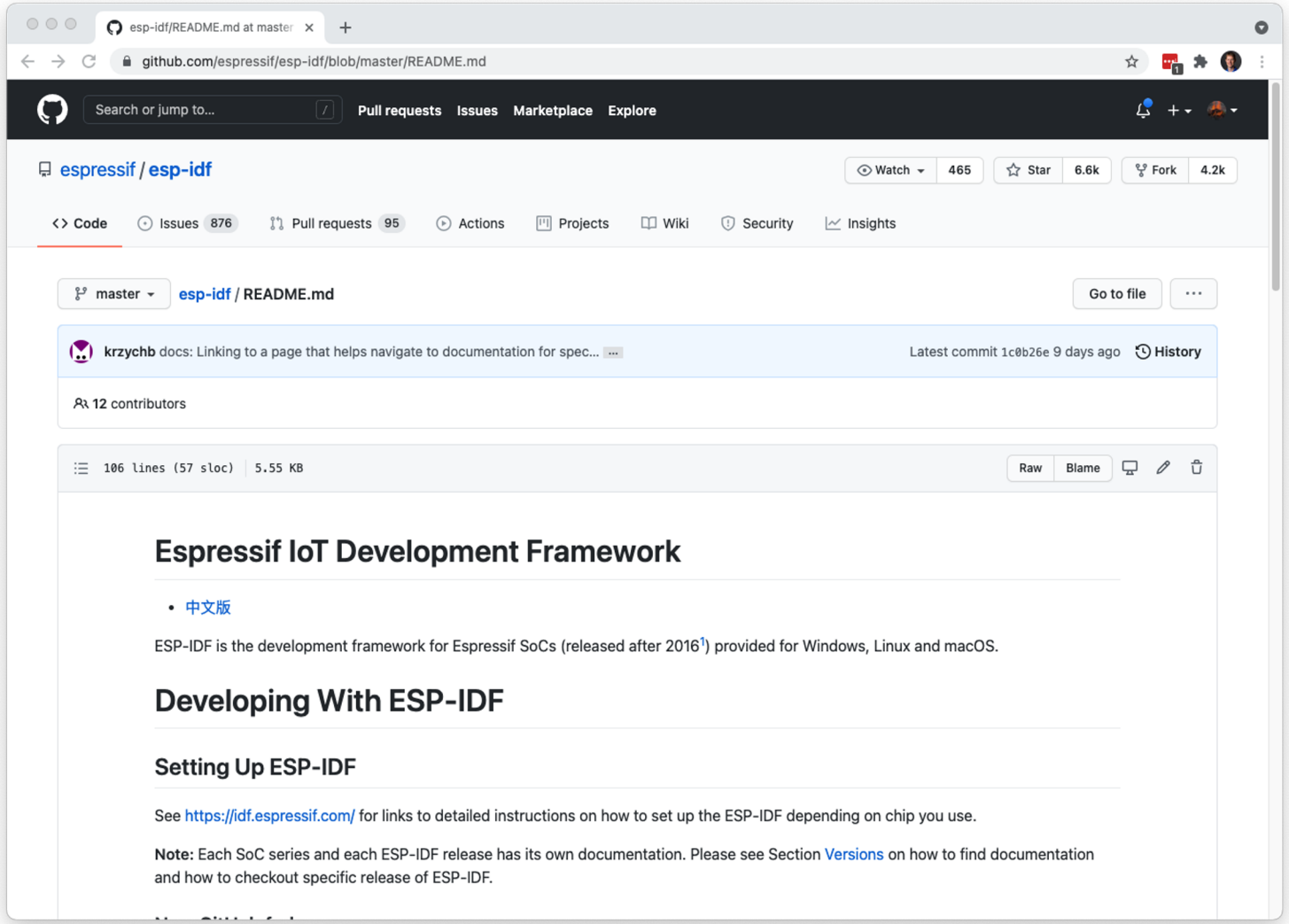
... compelling alternative to the Raspberry Pi

I am *just* a software engineer.

How can I write code for an ESP32?

instantiations

I found the ESP32 development framework on GitHub!



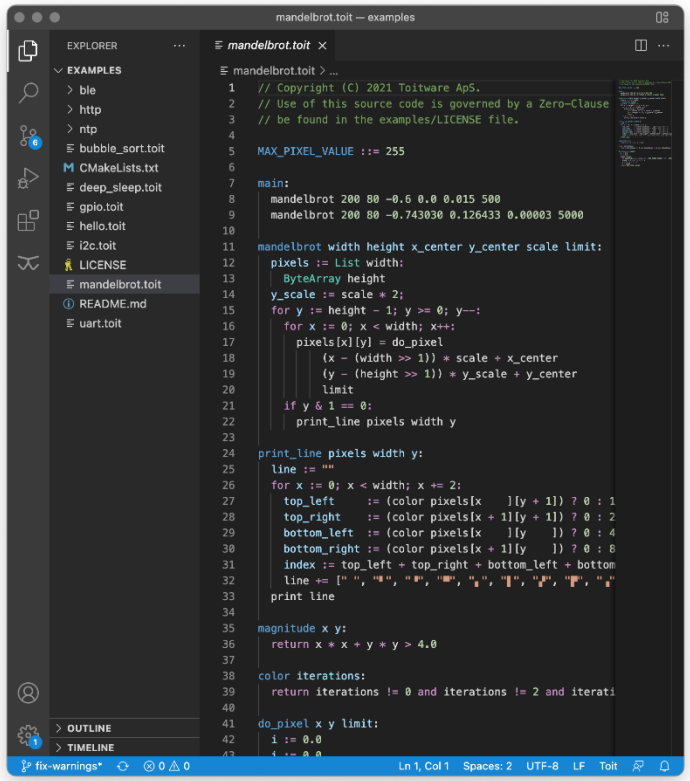
instantiations

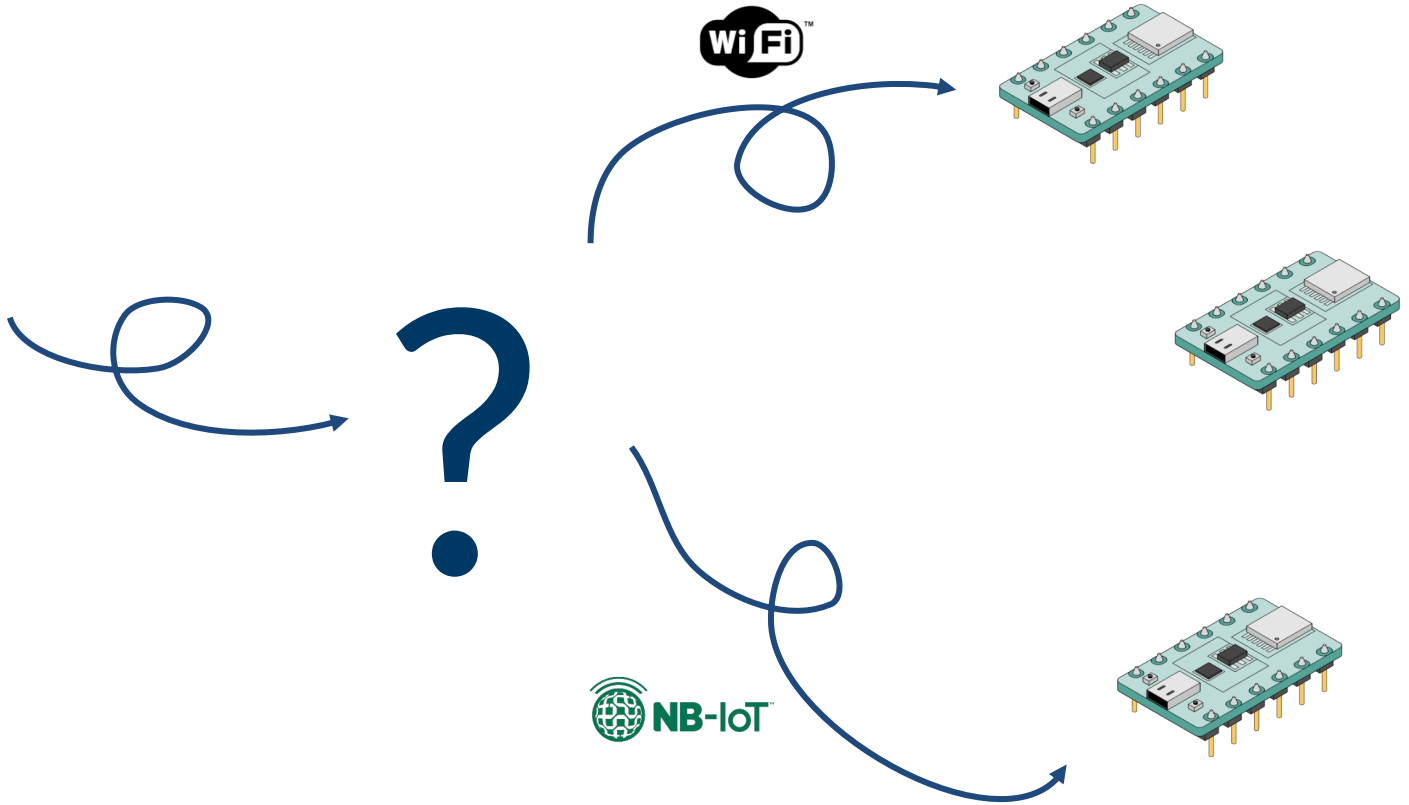
I still find that developing for microcontrollers sucks ...

- Thin, if any, separation between application, drivers, and OS
- Monolithic, close-knit system software tied to specific hardware
- Common source languages are C and assembly
- Application errors often result in crashing the entire device
- Development cycles are looooong

In spite of the hardware advances in microcontrollers,
the development experience just doesn't compare favourably
to server, desktop, or mobile development.

How can we enable and empower software engineers?

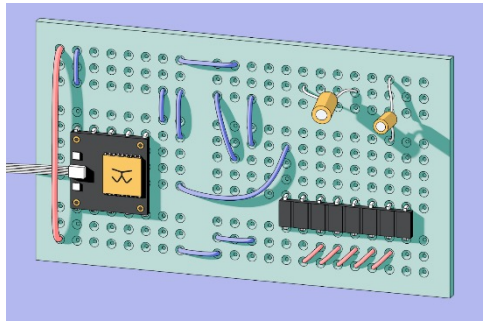




instantiations

https://toitlang.org/

Open source language implementation for microcontrollers

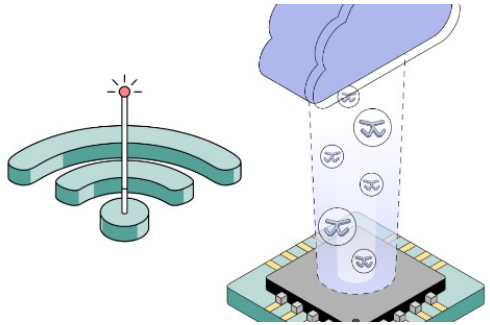


Productive and high level developer tooling for your ESP32s.
No C developers? No problem.

?

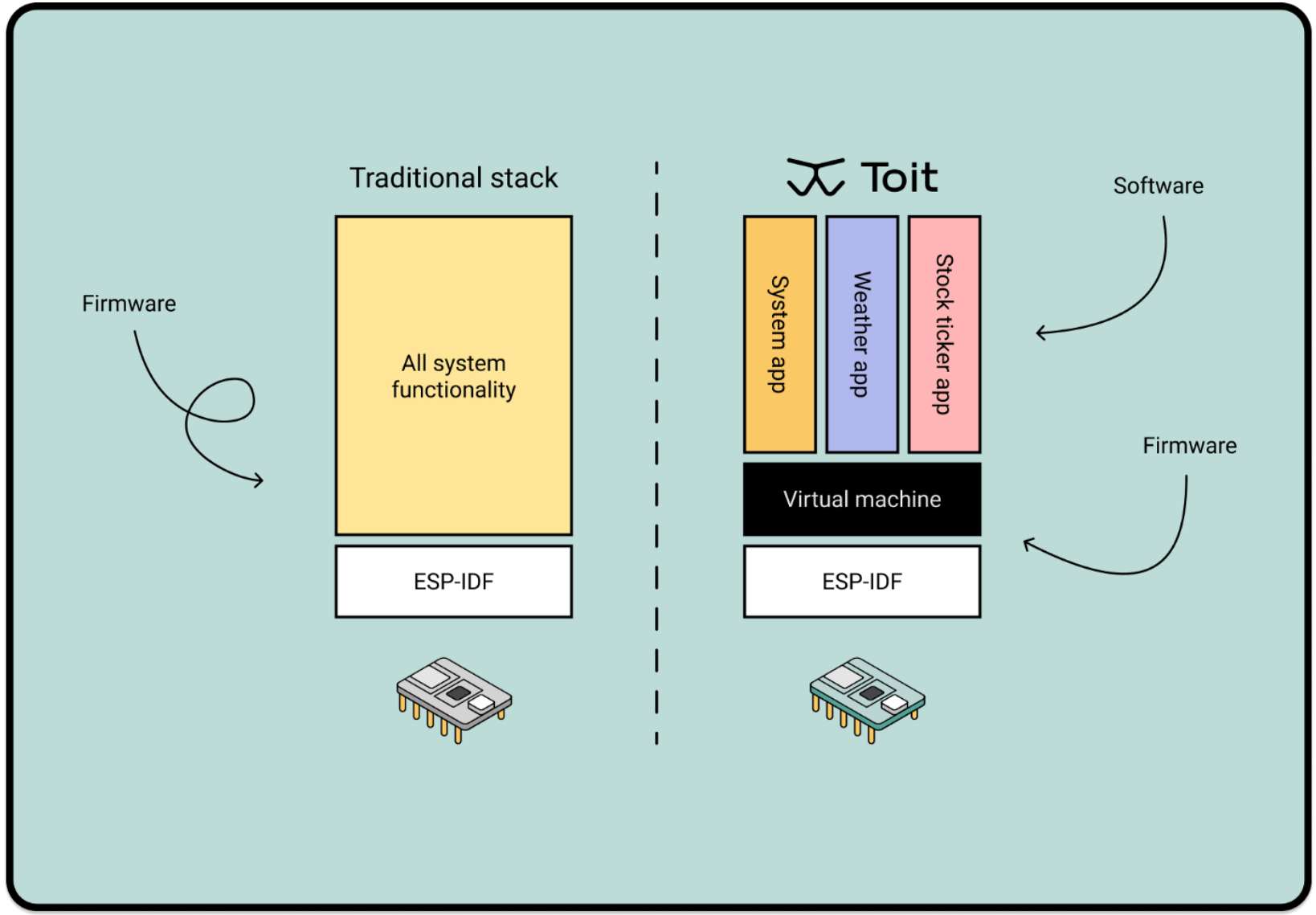
https://toit.io/

Cloud-managed containers on microcontrollers



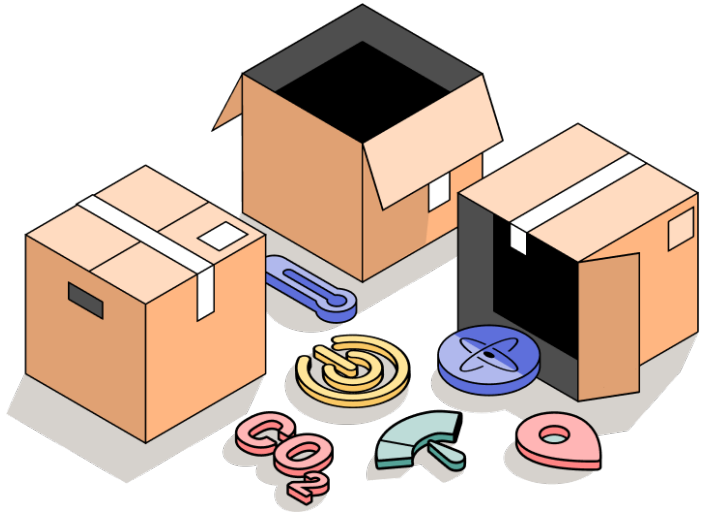
Cloud-managed environment for your ESP32 code, fully controlled through a rich programmatic API.

instantiations



We have been tackling this since 2018

- Innovate at the fundamental technology level so you don’t have to
- Build the foundation for the future of all microcontroller functionality
- Enable software engineers to use their skills on battery-operated devices



instantiations

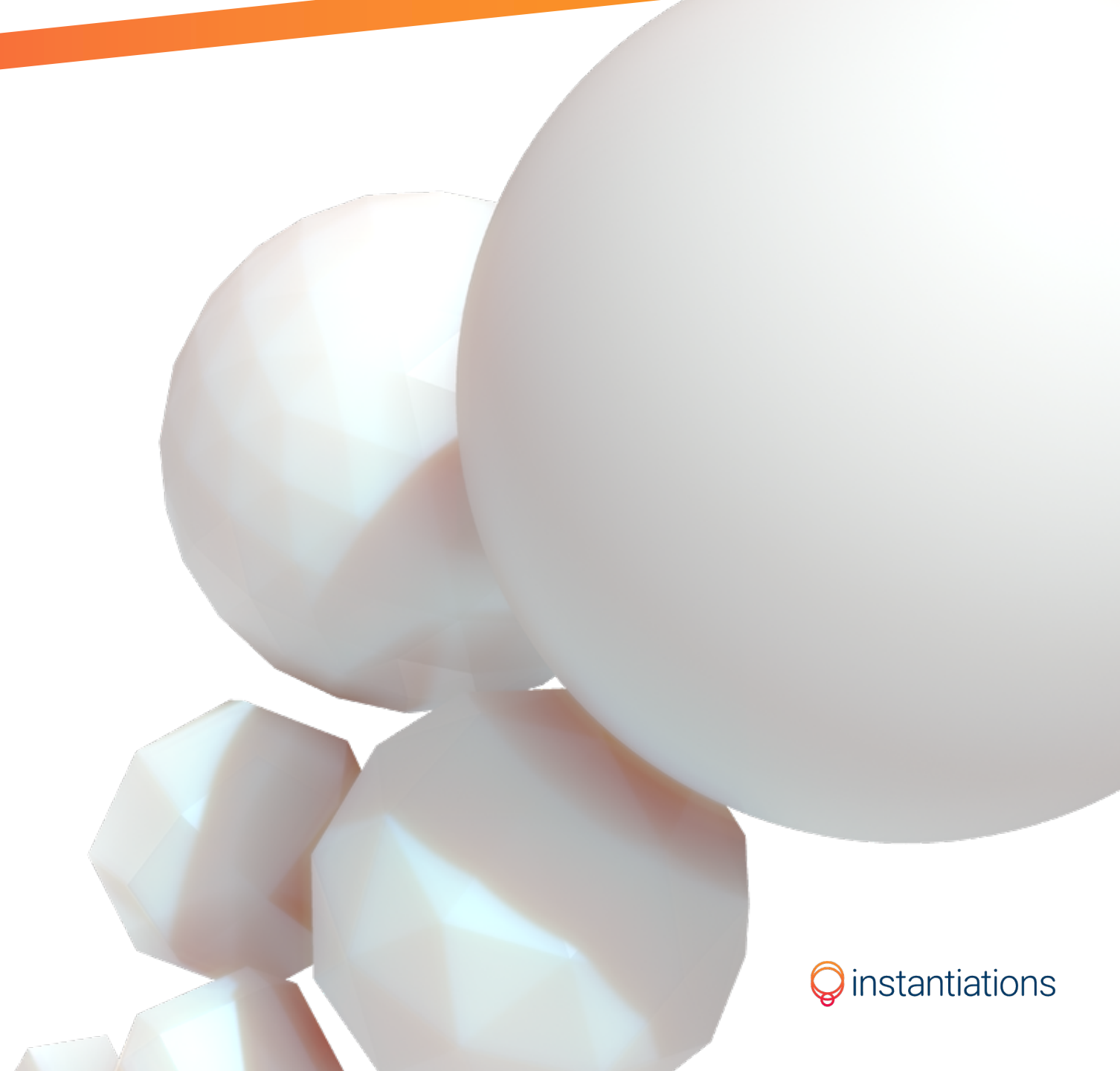
Hands-on demo with Toit

Jump-start your knowledge of the Toit language and platform

instantiations

Let’s get started

- Preparing for the demo
- Hardware and Toit setup
- ‘Hello World’ program
- ‘Date and Time Logger’ program
- ‘Traffic Light’ app
- ‘Fleet device update’ demo
- Preparing your ESP32 for home usage



instantiations

Preparing for the demo

instantiations

Preparing for the demo

Go to: instantiations.com/riot-dev-day-2022

1. Sign up for the Toit platform
2. Provide your Toit account email to us
3. Install Toit CLI
4. Install USB serial drivers
5. Confirm Chrome/Edge is up-to-date
6. **Download a PDF of the slides (now available)**

Hardware and Toit setup

instantiations

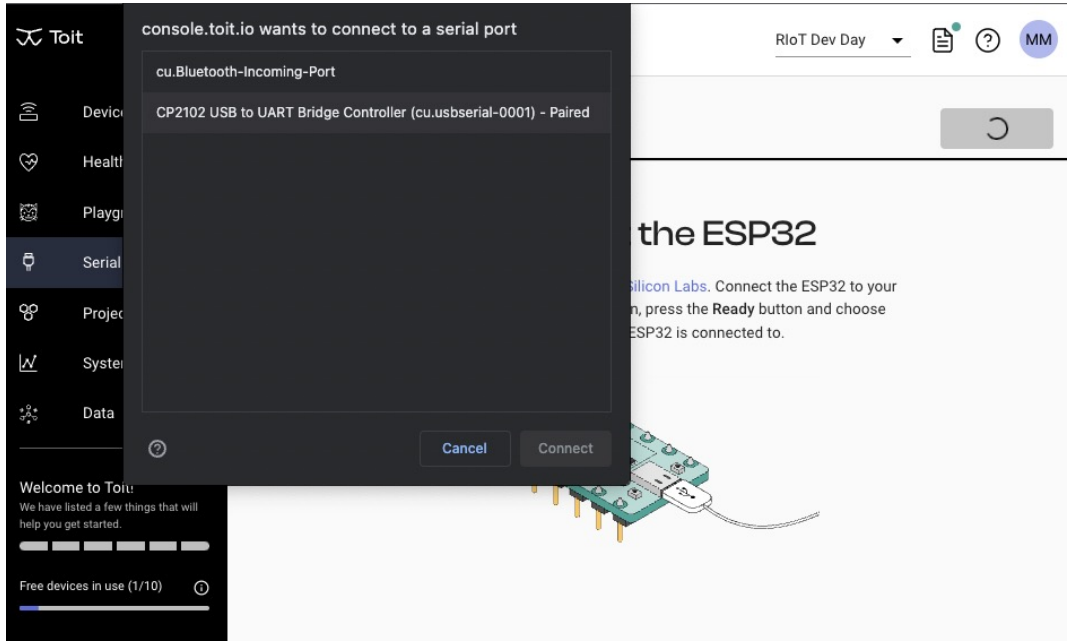
Unpack your Tech Demo Kit

Each kit includes:

- ESP32 Development Board
- Micro USB to USB-A cable
- GPIO Traffic Light

Provisioning the Hardware via Toit's Web Console

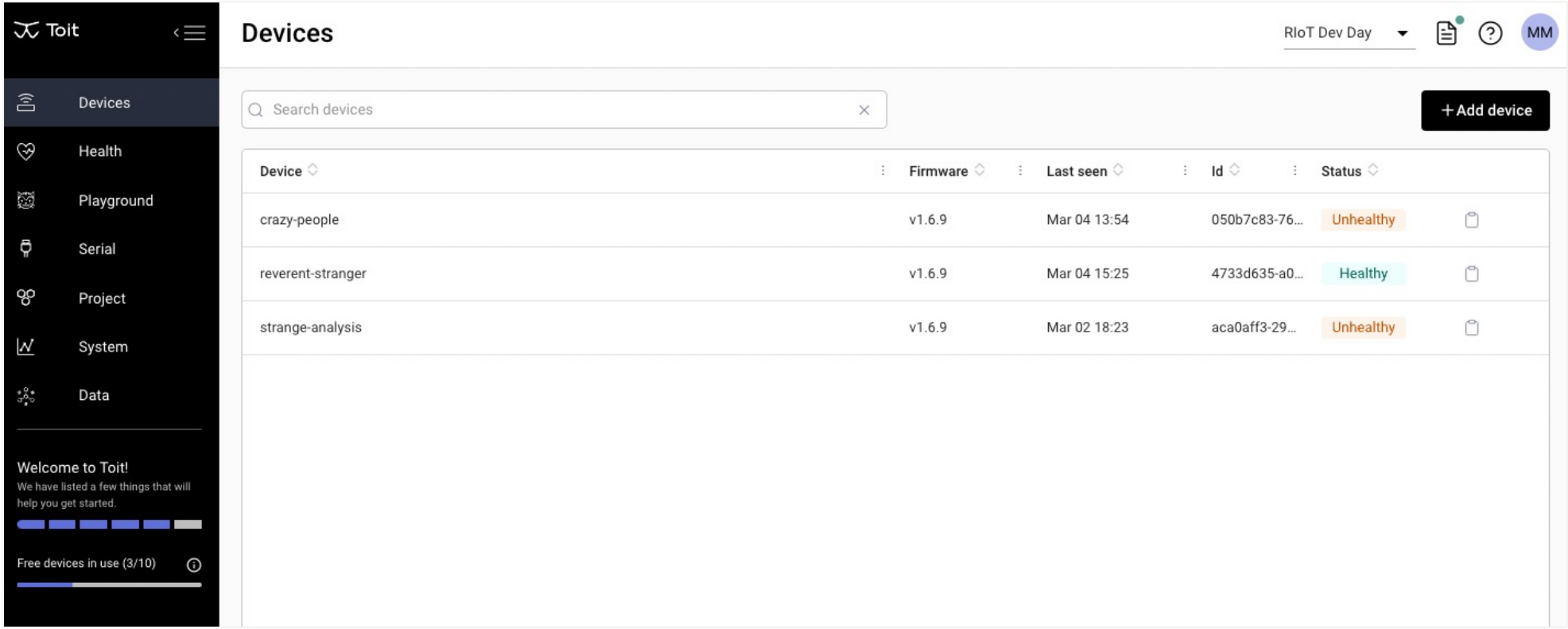
1. Sign in to https://console.toit.io
2. Select "RIoT Dev Day" project
3. Plug the ESP32 to your computer with USB cable
4. Go to the "Serial" Tab and provision the hardware with the "Ready >" button
5. SSID is "waketech" and no password



instantiations

Confirming the Setup

1. Go to the “Devices” tab
2. Your device should be up and running! (as status “Healthy”)



instantiations

Once your ESP32 is provisioned with Toit, it's very energy efficient and can run on any power source.

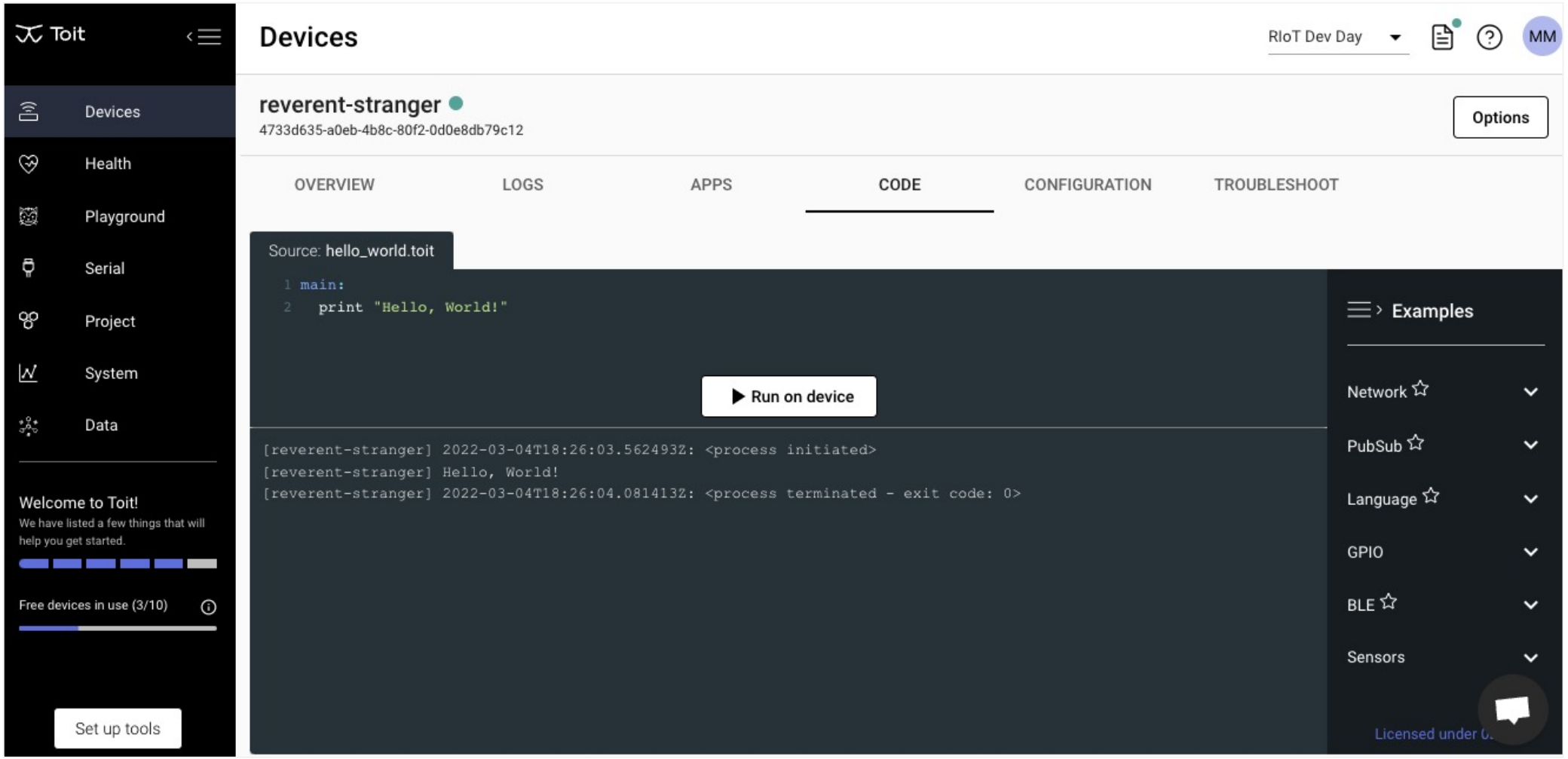
It no longer needs to be plugged into your computer. All further communications will be over-the-air.

Wi-fi, Cellular, and Ethernet are supported.

‘Hello World’ program

instantiations

‘Hello World’ Program from web console



instantiations

‘Date and Time Logger’ program

instantiations

Set default project and device

Set your ESP32 as the default one for you:

1. `$ toit project use "RIoT Dev Day"`
2. `$ toit device use <<device-name>>`

```
mariano@MacBook-Pro-2 ~/T/I/I/Toit
❯ toit devices
+--------------------------------------+---------------+----------+----------+
|              DEVICE ID               |     NAME      | LAST SEEN | FIRMWARE |
+--------------------------------------+---------------+----------+----------+
| 47b619e6-f610-48ac-ace7-be3e8ebefcde | final-square  | Now       | v1.1.3   |
| 6d0eba1e-3fc7-42db-ba78-b72c21177dde | relaxed-store | Now       | v1.1.3   |
+--------------------------------------+---------------+----------+----------+

mariano@MacBook-Pro-2 ~/T/I/I/Toit
❯ toit device use relaxed-store
relaxed-store is now your default device

mariano@MacBook-Pro-2 ~/T/I/I/Toit
❯
```

instantiations

Date and Time Logger “Program” from CLI

1. Grab code from:

    github.com/instantiations/devday2022-demo-toit

2. Save code in a new file named: dateAndTime.toit
   (if you didn’t clone it)

3. Run it:
   - `$ toit run dateAndTime.toit`

```
mariano@MacBook-Pro-3 ~/T/I/C/D/g/d/s/date-time-logger main : ?
❯ toit run dateAndTime.toit
2022-03-10T14:33:23.998812Z: <process initiated>
Time: 15:33
Date: 2022-03-10
2022-03-10T14:33:24.401103Z: <process terminated - exit code: 0>
```

instantiations

‘Traffic Light’ app

instantiations

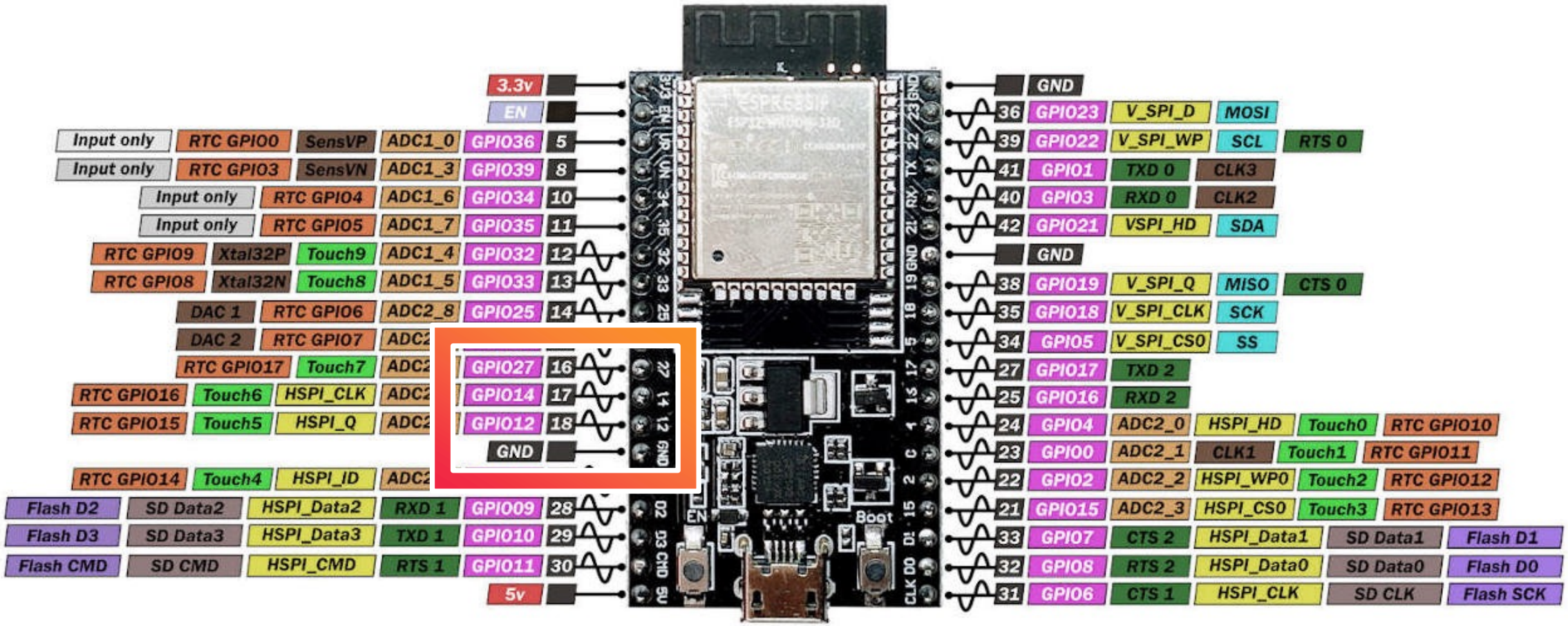
Application Code

1. Grab code from:

    github.com/instantiations/devday2022-demo-toit

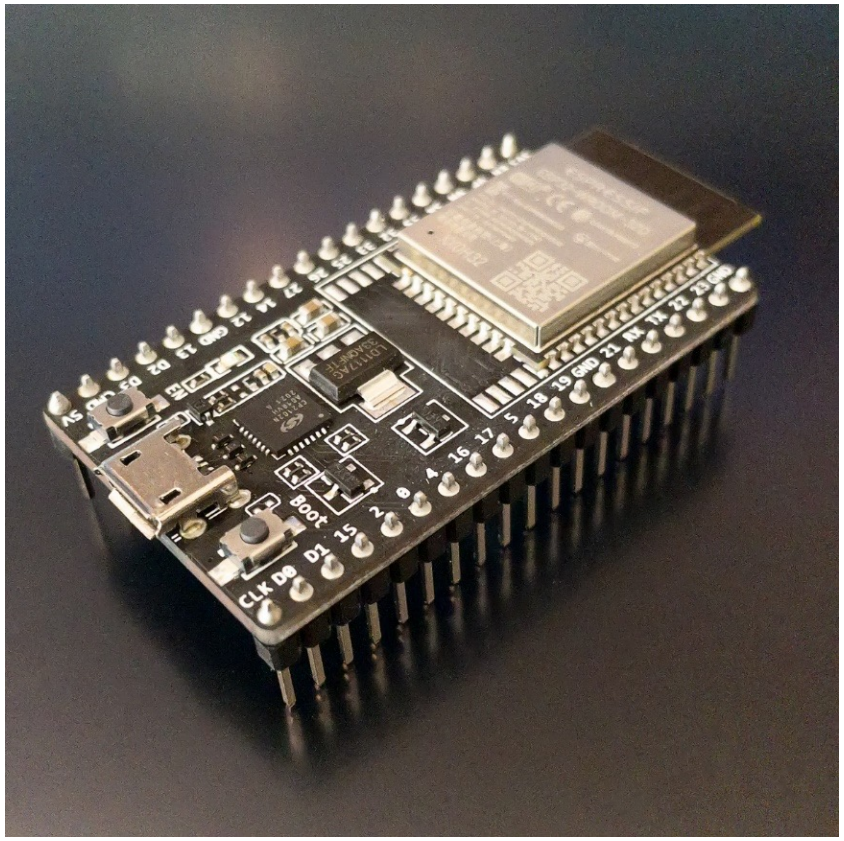
2. Save code in new files named: traffic-light.yaml and traffic-light.toit
   (if you didn't clone it)

ESP32 Dev Board Pin Layout

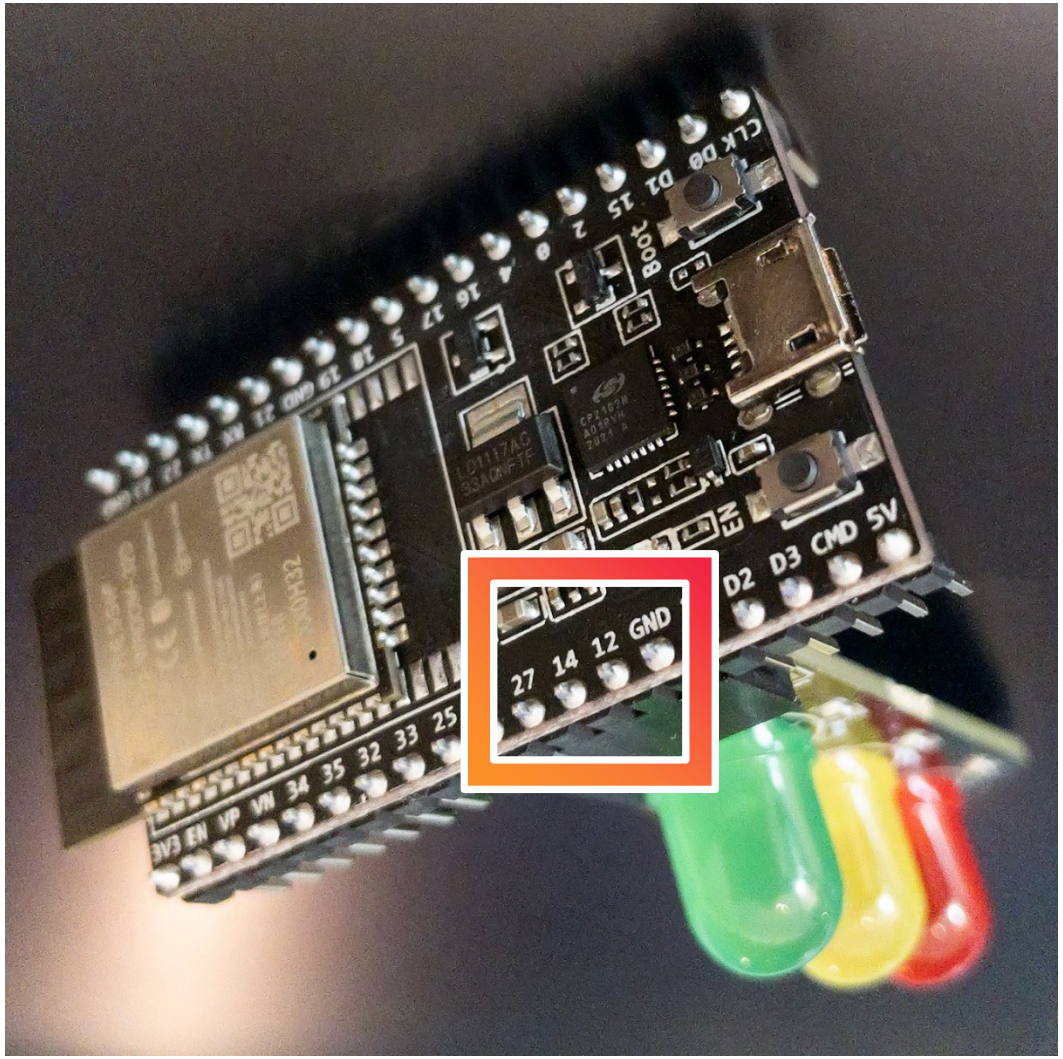


instantiations

Traffic Light Connection



ESP32 Development Board



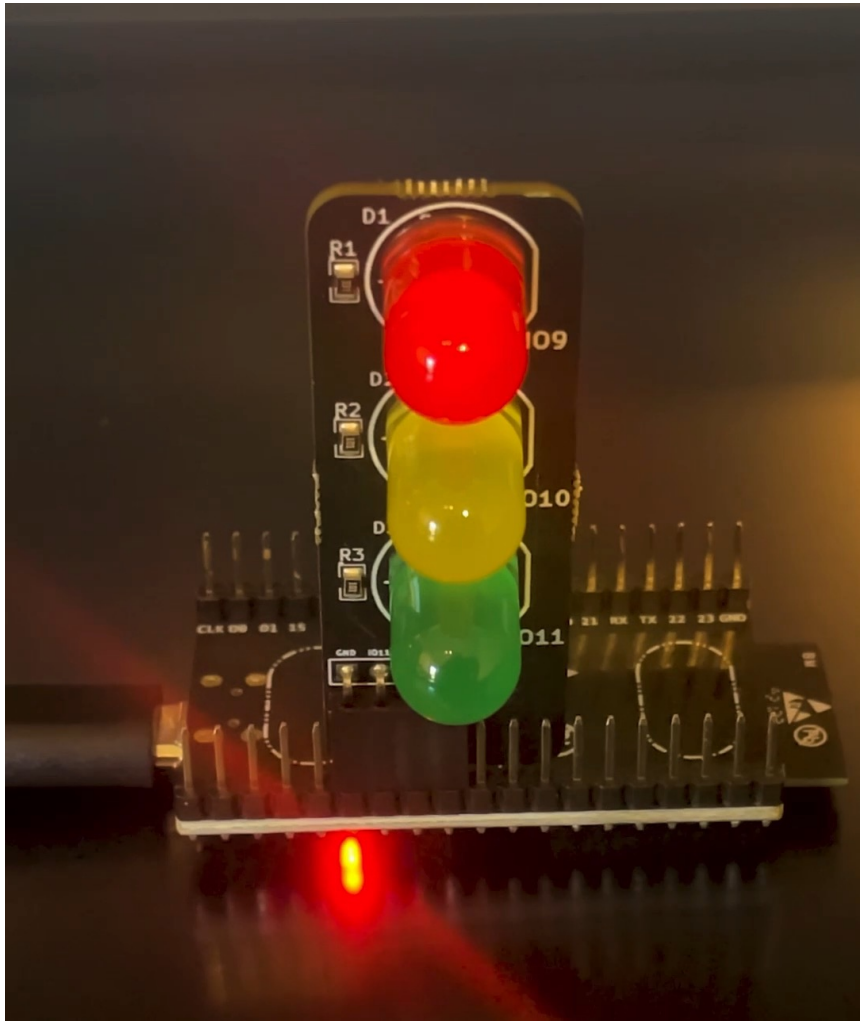
instantiations

Deploy the app

```
$  toit deploy traffic-light.yaml
```

```
❯ toit deploy traffic-light.yaml
Successfully deployed the job Traffic Light on 'reverent-stranger'
The job will be installed on 'reverent-stranger' next time it connects to the Toit console
```

The Result



instantiations

‘Fleet Device Update’ demo

instantiations

Updating our fleet of devices

1. Change code to make the lights switch faster

2. Update app on all devices over the air

```
$ toit fleet deploy traffic-light.yaml
```

```
mariano@MacBook-Pro-3 ~/T/I/I/Toit
 ❯ toit fleet deploy traffic-light.yaml
No device name prefix provided. Are you sure you want to proceed? [y/n] y
   ◦ Device 'crazy-people': Job 'Traffic Light' successfully deployed
   ◦ Device 'reverent-stranger': Job 'Traffic Light' successfully deployed
   ◦ Device 'strange-analysis': Job 'Traffic Light' successfully deployed
```

Preparing your ESP32 for home

…if you're not sticking around longer

instantiations

Preparing your ESP32 for home

1. Move device to “MyProject”
   1. `$ toit device unclaim`
   2. `$ toit project use "MyProject"`
2. Re-provision again your device (ensure it’s plugged in via USB)
   1. `$ toit serial provision --skip-identity-detection --device-name <<device-name>> -p wifi.ssid="waketech" -p wifi.password=""`
   2. `$ toit device -d <<device-name>> config connection -c "wifi:<<home-ssid>>:<<home-password>>" -c "wifi:waketech:"`
3. Your username will be removed from the “RIoT Dev Day” project after this session.

Additional resources

- https://toit.io/
- https://toitlang.org/
- https://docs.toit.io/
- https://pkg.toit.io/
- https://www.instantiations.com/news/bridging-the-gap-with-toit-a-rich-dev-experience-for-microcontrollers/

Hardware used in this demo:

- https://www.digikey.com/en/products/detail/espressif-systems/ESP32-DEVKITC-32D/9356990
- https://www.amazon.com/Pi-Traffic-Light-Raspberry-pack/dp/B00P8VFA42/
- https://www.amazon.com/Anker-3-Pack-Powerline-Micro-USB/dp/B015MJLEUS/

So, what's next?

instantiations

Thank you for attending!

Any more questions? Let's hear them.

Contact

General Inquiry
info@instantiations.com

Sales
sales@instantiations.com

Support
vast-support@instantiations.com

North America, Toll Free
855 476 2558

International
+1 503 263 0058

Seth Berman
President & CEO
Instantiations

sberman@instantiations.com
@sethloco78
/in/seth-berman

Kasper Lund
Co-Founder & CEO
Toit

kasper@toit.io
@toitware
/company/toitio/

Mariano Martinez Peck
Senior Software Engineer
Instantiations

mpeck@instantiations.com
@MartinezPeck
/in/mariano-martinez-peck