Anatomy of an EV charging system

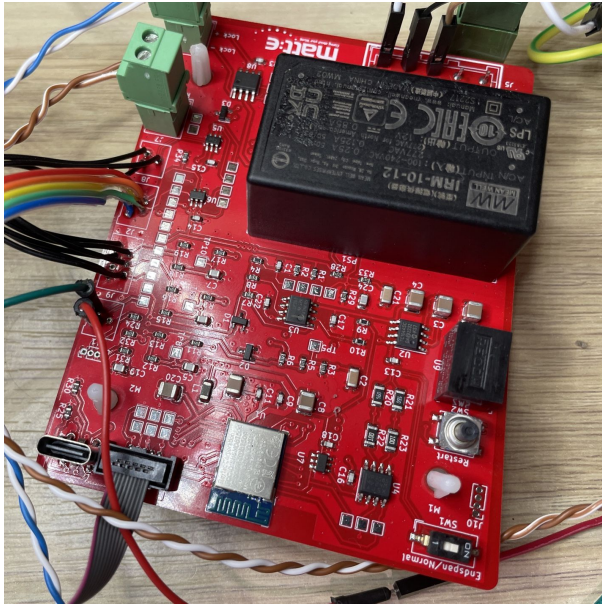
WRITTEN IN RUST

What is an EV charging system?

- An Electric Vehicle charger (EVSE)
  - What the car plugs into
  - firmware/software monitors temperatures, current flow, voltage, meters energy
- A mains board monitoring device
  - Lives in the switchboard
- An internet gateway
  - Lives in the switchboard
  - Manages WiFi/Ethernet/RS485
  - Secure
- An RS485 network that connects all of the above
  - 1km range, economic in cost, reliable
- A cloud service for managing/observing the system
  - Bridges a proxy with a VPN

What's in an EVSE?

- Firmware
  - Rust embedded
  - 100% async
    - 15 tasks
    - Sampling analog 10kHz
  - 140 KiB size
  - Encrypted RS485 and USB-serial
  - ed25519 verified firmware updates
- Software
  - no_std
  - DSPs, state machines, things we wanted to unit test/exercise
- We also did the hardware design
  - Firmware and hardware very tightly integrated



Major firmware/software dependencies

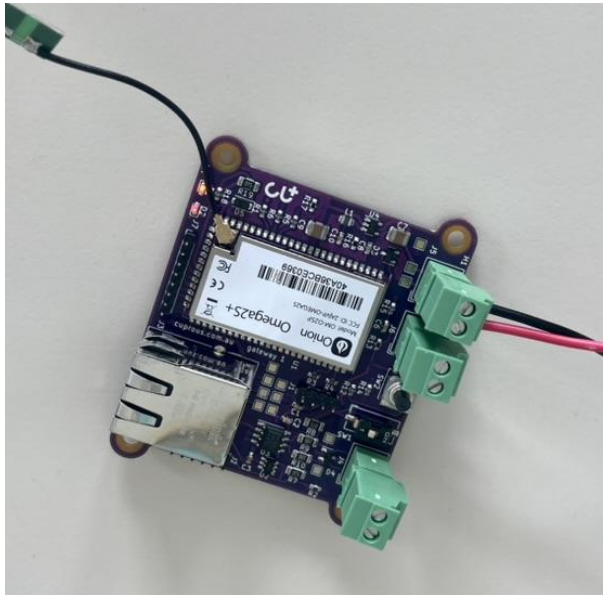
- Various Rust Crypto
- edfsm - event driven finite state machines
  - We open sourced this: https://crates.io/crates/edfsm
  - Provides a DSL for describing states, commands, events and transitions
- Embassy - async embedded executor and HALs
  - Contributed channels, saadc, firmware update verification, some UART, some USB, some PWM
- Flip-flop-protocol - half duplex protocol for commands and events
  - We open sourced this: https://github.com/titanclass/flip-flop-protocol
  - Discovery, encryption, broadcast, firmware update
- Serde/Postcard - serialisation

What's in a monitor board?

- Very similar to the EVSE
- Its function is to measure:
  - three phase current
  - three phase voltage
  - PV current
- Broadcasts this information which is consumed by the gateway controller

What's in an internet gateway?

- Hardware and software
- We built the hardware
  - Omega2S+, 500mHz, 128MiB, 32MB
  - Runs OpenWRT Linux
    - Customised
- Our software:
  - Configurator
    - Backend
    - Frontend
- Our client's software:
  - Controller
    - Backend
    - Frontend



Major std software dependencies - back

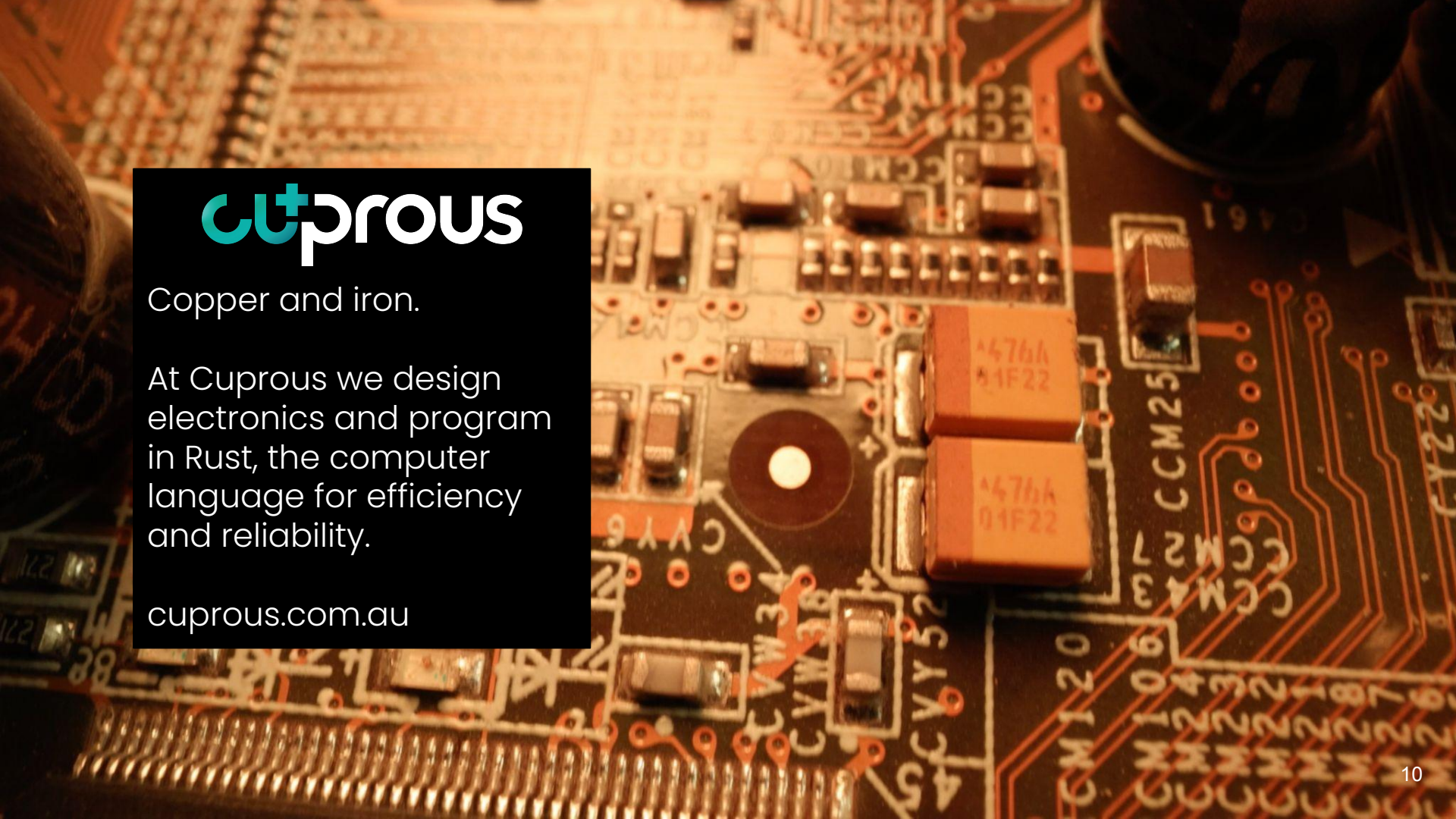
- Various Rust Crypto
- Chrono
- Clap
- Flip-flop-protocol
- immutable-chunkmap **
- Ciborium **, postcard, serde, serde-with
  - CBOR - a binary JSON
- Streambed-rs - event driven microservices
  - We open-sourced this: https://github.com/streambed/streambed-rs
  - Kafka-like commit log (32 KiB), secret store (45 KiB) - opt level z
  - Long history - evolved from our beginnings with Cisco
- Tokio
- Warp

Major std software dependencies - front

- Chrono
- Gloo
- immutable-chunkmap
- serde/json
- Yew
- Yew-chart - SVG charting primitives
    - We open-sourced this: https://crates.io/crates/yew-chart

Takeaways

- We contribute about 30% of our time to open-source
- Our entire stack is Rust
- Rust is more than ready for prime-time
- Rust is empowering
    - This has been my first firmware targeted at production


cuprous
Copper and iron.
At Cuprous we design electronics and program in Rust, the computer language for efficiency and reliability.
cuprous.com.au