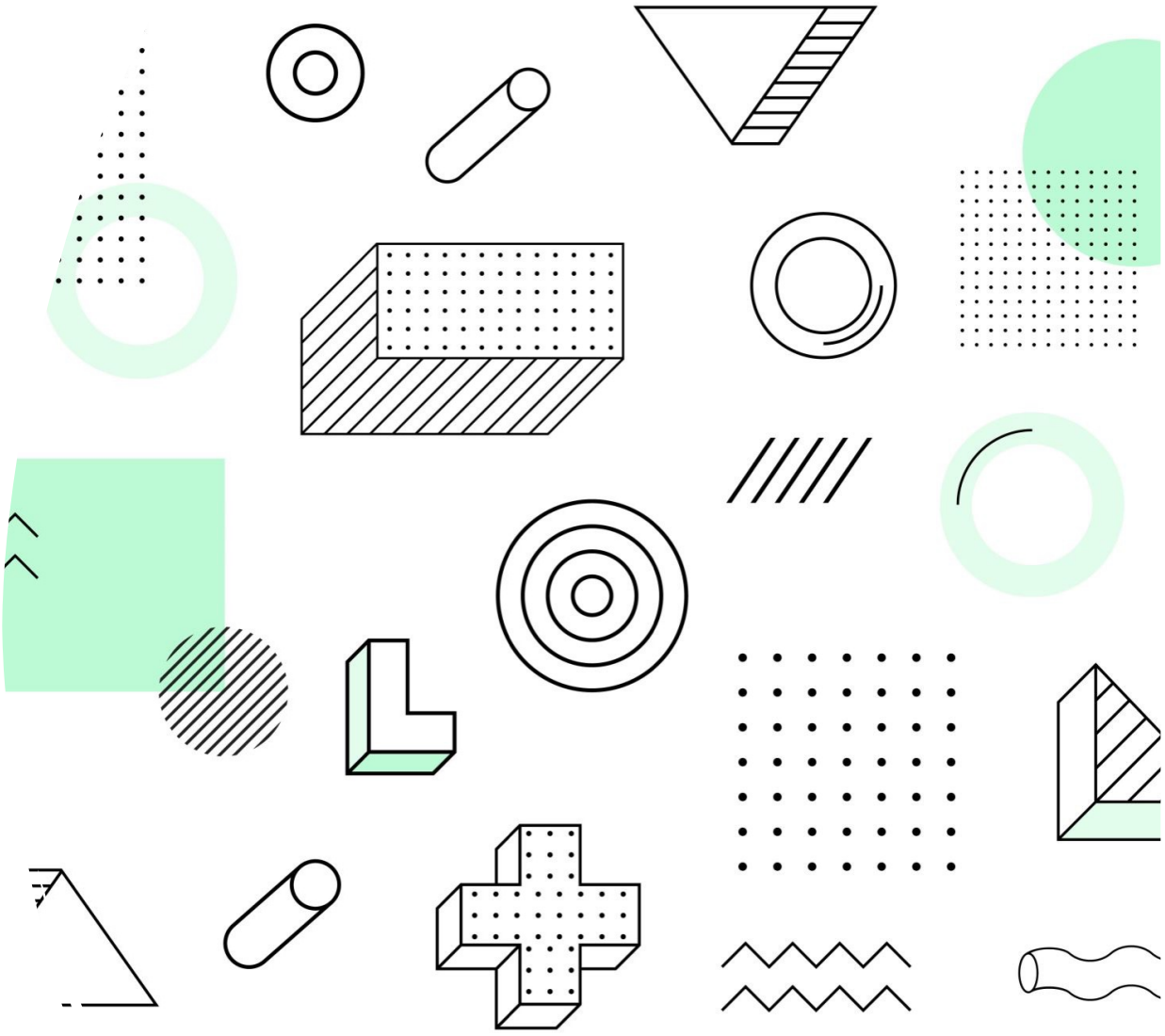
ESP32+LoRa Workshop

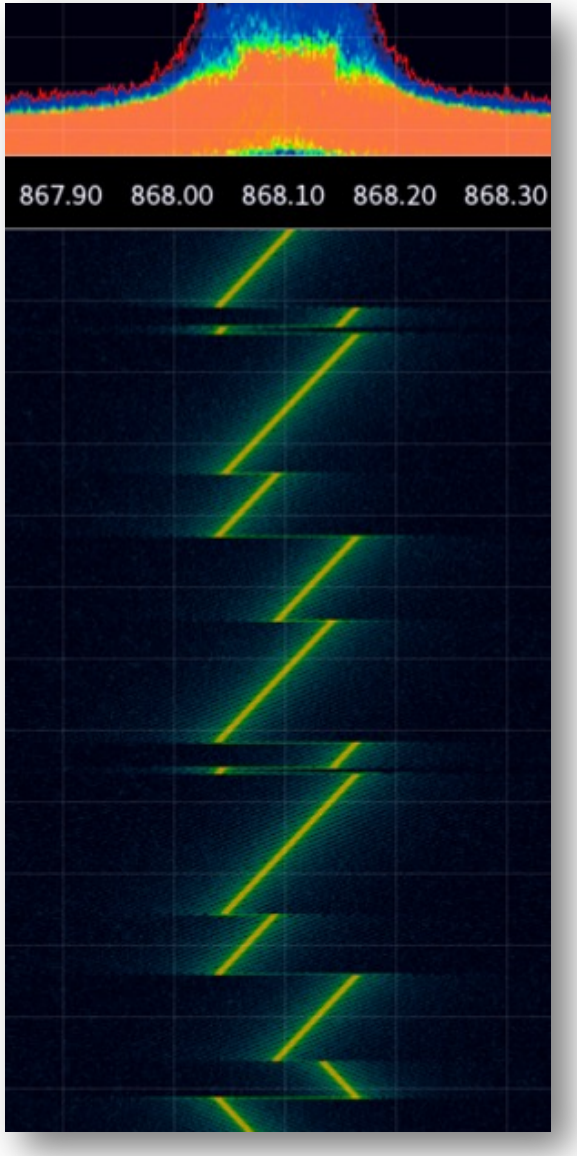
Hands-on Tutorial



LoRa Basics

- PHY for Low Power Wide Area Networks (LPWAN)
- Chirp Spread Spectrum (CSS)
- 433 MHz ISM, 868 MHz (EU) SRD, 915 MHz (US) ISM
- 0.3 kbit/s to 50 kbit/s
- Range ≈1-5km (city/urban), >10km (LoS, rural)
- EU 868 MHz: Tx Power and Duty Cycle restrictions!

Frequency (F)
Bandwidth (BW)
Spreading Factor (SF)
Coding Rate (CR)
Sync Word (SW)



EU 868 / Germany Legal Stuff

- “Short Range Device” (SRD) Band — 863-870 MHz

Subband (MHz)	TX Power	Duty Cycle
863,0–865,0	25 mW	≤0,1 %
865,0–868,0	25 mW	≤1 %
868,0–868,6	25 mW	≤1 %
868,7–869,2	25 mW	≤0,1 %
869,4–869,65	500 mW	≤10 %
869,7–870,0	5/25 mW	≤10% / ≤1%

https://www.bundesnetzagentur.de/SharedDocs/Downloads/DE/Sachgebiete/Telekommunikation/Unternehmen_Institutionen/Frequenzen/Allgemeinzuteilungen/FunkanlagenGeringerReichweite/2018_05_SRD_pdf.pdf?__blob=publicationFile&v=7

Heltec ESP32 WiFi LoRa v3

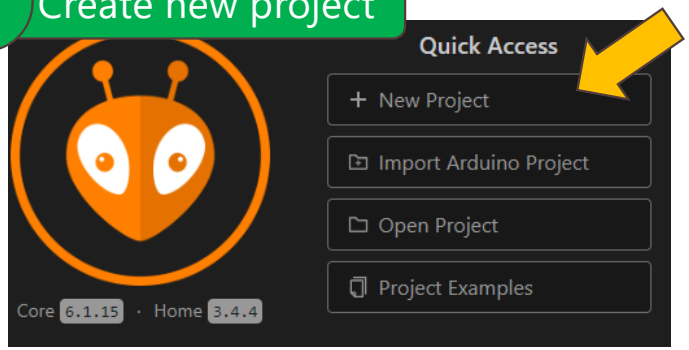
- 32bit dual-core microcontroller from Espressif (ESP32-S3FN8)
- WiFi (802.11 b/g/n), Bluetooth 5, LoRa (SX1262)
- USB-C, Battery Interface, 0.96“ 128x64 OLED display

- Development choices
  - C/C++: Espressif IoT Development Framework (ESP-IDF)
  - C/C++(alike): Arduino Core SDK (builds upon ESP-IDF)

More info: https://www.espboards.dev/blog/esp-idf-vs-arduino-core/

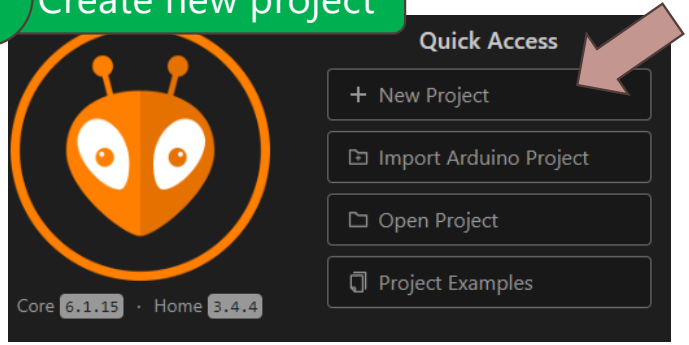
First Project: VSC + Platform.IO

1 Create new project



First Project: VSC + Platform.IO

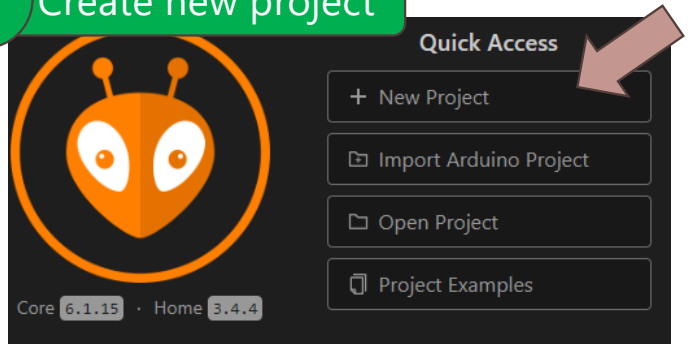
1 Create new project



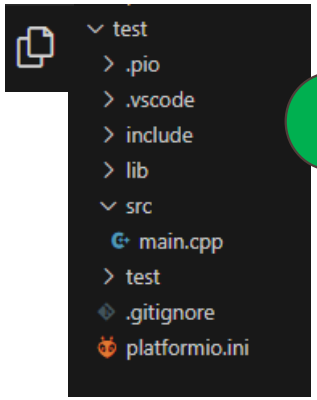
2 Select board & framework

First Project: VSC + Platform.IO

1 Create new project

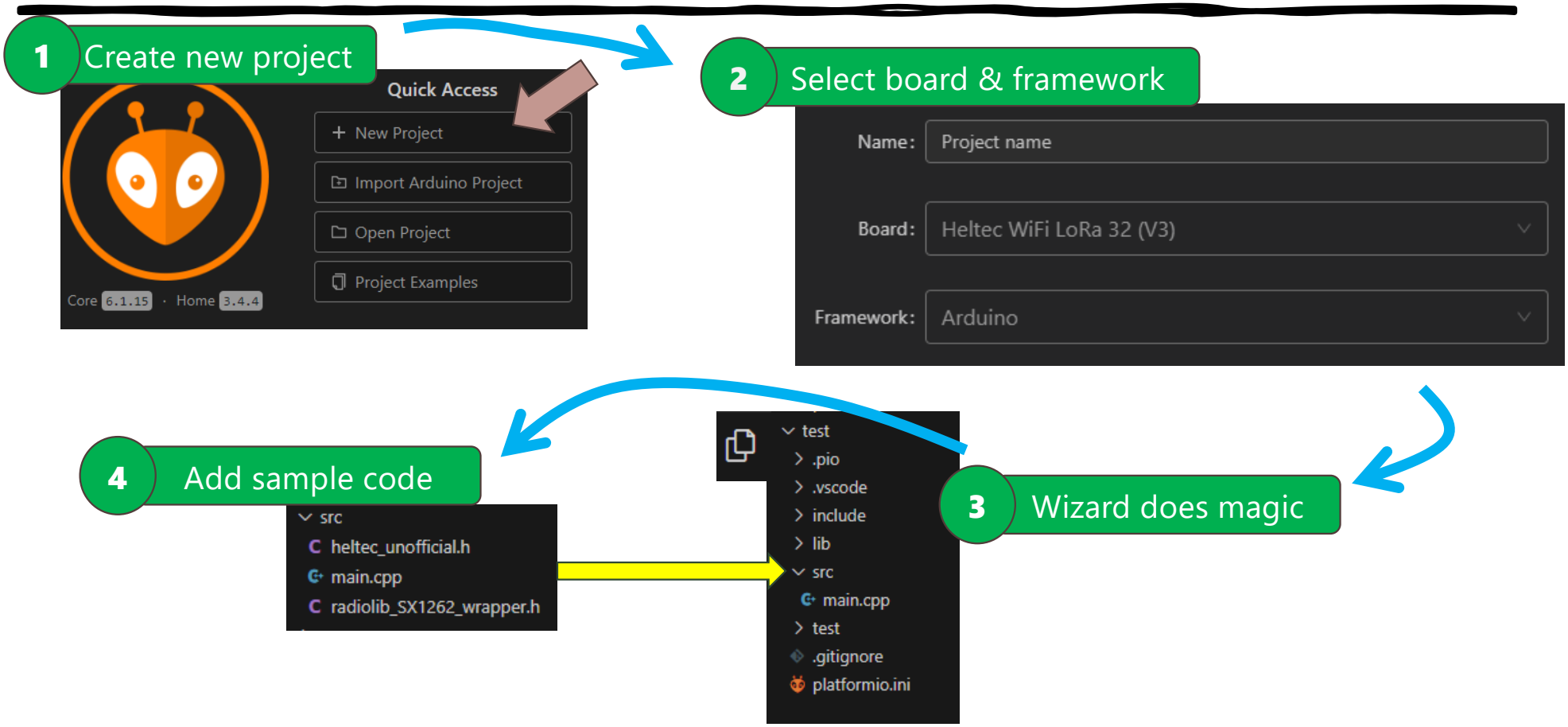


2 Select board & framework

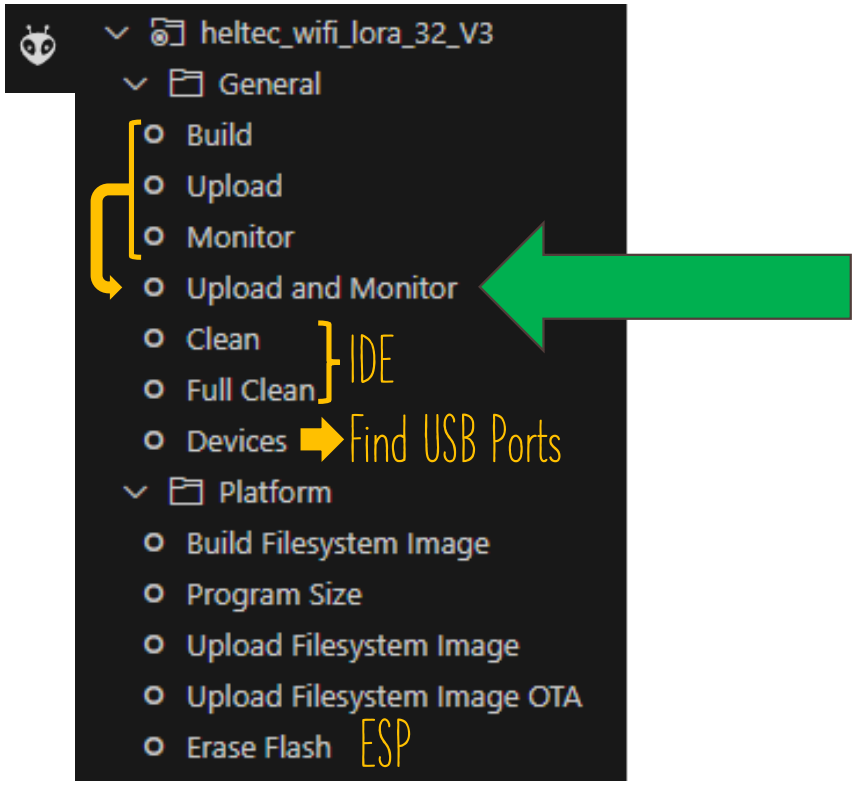


3 Wizard does magic

First Project: VSC + Platform.IO



Working with Platform.IO



Main Classes

main.cpp

- `setup()` → called first
- `loop()` → continuous call after finished setup
- `click_callback()` → PRG button
- `callback_lora_action()` → “something happens”

SX1262_wrapper.h

- Setup for SX1262 chip
- RadioLib
- Simple duty cycle check
- Send function

Pick your device ID

0x11
0x22
0x33
0x44
0x55
0x66
~~0x77~~
0x88
0x99
0xAA
0xBB
0xCC
0xDD
0xEE
~~0xFF~~

First Reception

- Setup device with given LoRa parameters and your ID
- Implement callback function
  → set flag for received msg
  → check received in `loop()`
- Receive message from sender *`0xC1`* (sent every 10 seconds)
- Put to console and/or display

Frequency: 866.5 Mhz
Bandwidth: 250.0 kHz
Spreading Factor: 10
Coding Rate: 4/5
Sync Word: 0x36

```
Serial.println(String(rssi));
display.drawString(0, 0, String(rssi));
```

Don't forget to put your radio back to listen after reception and sending!

```
radio.startReceive();
```

Gotta complete ‘em all!