Design and Implementation of a 433MHz RF IoT Gateway

Alex Rodríguez Navas

Bachelor's Thesis

Degree in Informatics Engineering / Major in Computer Engineering

Thesis Director **Enric Xavier Martin Rull**

GEP Tutor **Paola Lorenza Pinto**

Barcelona School of Informatics UPC / FIB

Barcelona January 2023

Table of Contents

1 Introduction
- Context
- Test subjects
- Core concepts

2 Problem
- Definition
- Value proposal
- Aimed scope
- Objectives

3 Management
- Planning
- Budget
- Risks
- Regulations

4 Foundation
- Initial research
- Proposed system
- Taken approach
- Hardware components

5 Implementation
- Schematic
- Printed circuit board
- Casing
- Packaging
- Firmware

6 Results
- Demonstration
- Conclusions
- Future work

Introduction

Context

- Every second 200 new IoT devices connect to the Internet for the first time
- Connections are expected to triple by 2025 generating up to 11 trillion dollars in economic value
- However, only 0.06% of devices that could be connected to the Internet actually are

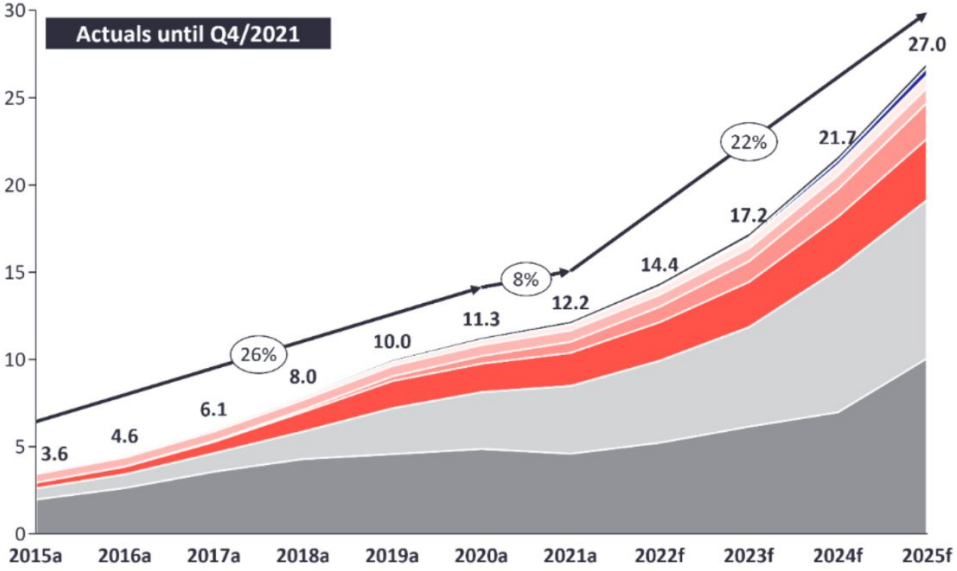


Figure 1: Number of global active IoT connections in billions by year.
Source: Mohammad Hasan [2].

Test subjects

- There exists a huge portion of those not-so-old dorky devices that can be smartified by hooking them up to the Internet
- From sensors: weather stations or motion detection alarms; to actuators: wireless switches or motorized shutters
- The common denominator of these devices is the usage of the 433MHz radio frequency band for data communication

Figure 5: Splenssy open-close alarm.
Source: Compiled by the author.

Figure 6: SatChef wireless socket.
Source: Compiled by the author.

Figure 7: Master Otello wireless motorized blinds.
Source: Compiled by the author.

Core concepts

- Internet of Things (IoT) concept of connecting anything to the Internet building up a network where all devices collect and share data
- Amplitude Modulation (AM) analog modulation technique that represents data on a continuous range of voltages [0, Vmax], varying the amplitude signal
- Amplitude-Shift Keying (ASK) digital modulation technique that represents data on a discrete range of voltages {0, Vmax} (assuming two levels), varying the amplitude signal too
- Gateway any device that connects separate networks by translating communications from one protocol to another

Problem

Definition

- Belonging to the Internet provides:
  - Location transparency
  - Time transparency
  - Access transparency
- 433MHz sensors and actuators are unconnected:
  - Manual single-purpose actions
  - Information with limited monitoring
  - Important enough and used daily

How can we bring these RF devices to the Internet?

433MHz radio frequency to Internet gateway

Value proposal

- Existing solutions? Yes but...
  - Expensive
  - Vendor lock-in
  - Closed source
  - Dependant

- Proposed solution:
  - Affordable
  - Vendor agnostic
  - Open source
  - Standalone

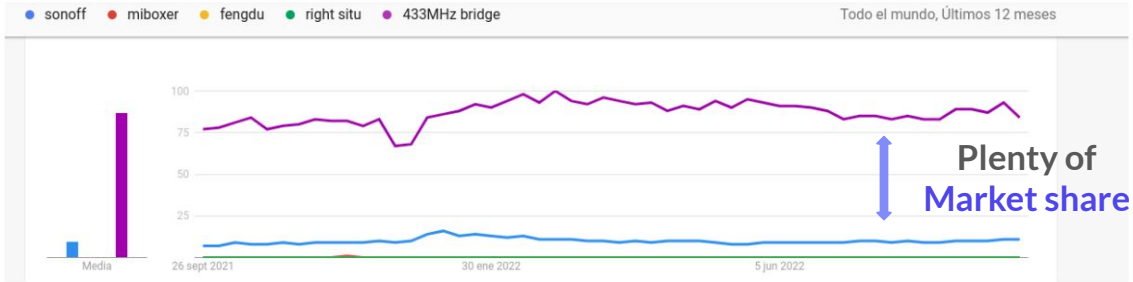


Figure 2: Percentages of global searches about Sonoff, Miboxer, Fengdu, Right Situ and 433MHz bridges on Google. *Source: Google Trends* [11].

Aimed scope

- **Inside:**
  - Electrical schematic
  - Printed circuit board
  - Casing
  - Packaging
  - Firmware
- **Outside:**
  - Mass manufacturable
  - User manuals
  - Regulatory compliance
  - Electrical certifications

End to end **retail** electronic product **prototype**

However curated up to the complexity, economic budget and time deadline

Objectives

- **Electrical schematic:**
  - Design without any ERC errors
- **Printed circuit board:**
  - Design without any DRC errors
  - Manufactured sample without any electrical flaws
- **Casing:**
  - Design without any measurement errors
  - Manufactured sample without any mechanical flaws
- **Packaging:**
  - Design without any styling errors
  - Manufactured sample without any printing flaws
- **Firmware:**
  - Encode and transmit commands to actuators
  - Receive and decode data from sensors
  - Receive and respond to HTTP requests
  - Maintain the current real time
  - Trigger commands based on a schedule
  - Enforce role based access control
  - Built-in web-based dashboard
  - Encrypt communications through TLS
  - Ensure firmware integrity and encrypt flash
  - ...

29 objectives

Management

Planning

September - January

522 hours

37 tasks

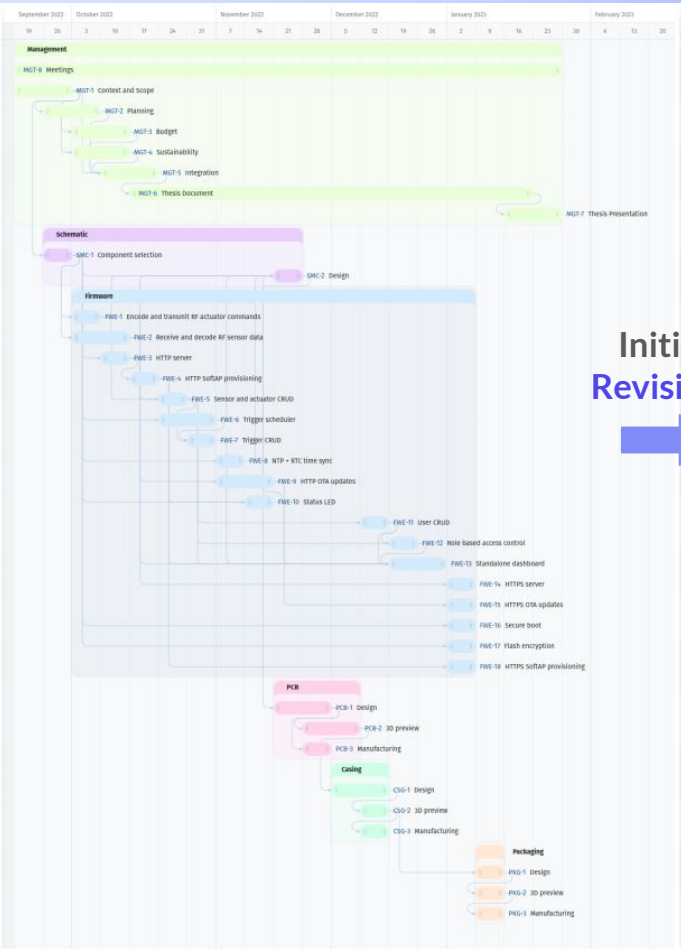


Figure 3: Gantt chart of the tasks. *Source: Compiled by the author.*

Initial

Revisited

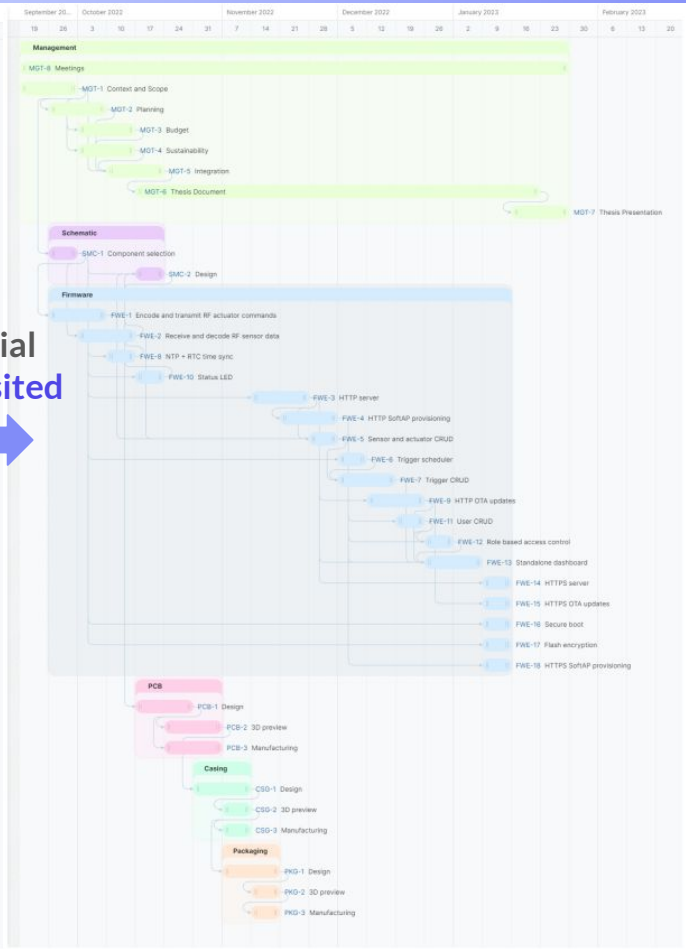


Figure 4: Revisited Gantt chart of the tasks. *Source: Compiled by the author.*

Budget

Hypothetical

ID	NAME	COST
STF	Staff	16,823.94
MAT	Material	50.12
GEN	Generic	439.87
INC	Incidental	374.53
CGY	Contingency	3,537.69
TOTAL		21,226.15

Table 7: Cost in euros of staff, materials, generics, incidences and contingencies through all the project. *Source: Compiled by the author.*

Reality

250 €

ID	NAME	HOURS				COST
		PM	CE	WD	IE	
MGT	**Management**	**197**	**24**	**24**	**24**	**8,205.12**
MGT-1	Context and Scope	18				583.20
MGT-2	Planning	18				583.20
MGT-3	Budget	7				226.80
MGT-4	Sustainability	7				226.80
MGT-5	Integration	4				129.60
MGT-6	Thesis Document	83				2,689.20
MGT-7	Thesis Presentation	36				1,166.40
MGT-8	Meetings	24	24	24	24	2,599.92
FWE	**Firmware**	**0**	**193**	**30**	**0**	**5,879.10**
FWE-1	Encode and transmit RF		30			810.00
FWE-2	Receive and decode RF		30			810.00
FWE-3	HTTP server		18			486.00
FWE-4	HTTP SoftAP provision		15			405.00
FWE-5	Sensor & actuator CRUD		12			324.00
FWE-6	Trigger scheduler		12			324.00
FWE-7	Trigger CRUD		6			162.00
FWE-8	NTP + RTC time sync		12			324.00
FWE-9	HTTP OTA updates		18			486.00
FWE-10	Status LED		4			108.00
FWE-11	User CRUD		6			162.00
FWE-12	Role based access control		10			270.00
FWE-13	Standalone dashboard			30		668.10
FWE-14	HTTPS server		4			108.00
FWE-15	HTTPS OTA updates		4			108.00
FWE-16	Secure boot		4			108.00
FWE-17	Flash encryption		4			108.00
FWE-18	HTTPS SoftAP provision		4			108.00
SMC	**Schematic**	**0**	**24**	**0**	**0**	**648.00**
SMC-1	Component selection		12			324.00
SMC-2	Design		12			324.00
PCB	**PCB**	**0**	**36**	**0**	**0**	**972.00**
PCB-1	Design		18			486.00
PCB-2	3D preview		12			324.00
PCB-3	Manufacturing		6			162.00
CSG	**Casing**	**0**	**0**	**0**	**24**	**639.84**
CSG-1	Design				12	319.92
CSG-2	3D preview				6	159.96
CSG-3	Manufacturing				6	159.96
PKG	**Packaging**	**0**	**0**	**0**	**18**	**479.88**
PKG-1	Design				6	159.96
PKG-2	3D preview				6	159.96
PKG-3	Manufacturing				6	159.96
TOTAL		**197**	**276**	**54**	**66**	**16,823.94**

Table 3: Cost in euros and hours dedicated to project tasks by profile. *Source: Compiled by the author.*

ID	NAME	UNITS	COST		LIFE TIME	AMORTIZATION	
			x1	x1000		MONTHLY	THESIS
COMPONENTS			**35.95**	23.39			
E3W	ESP32-S3-WROOM-2-N32R8V	1	7.34	7.34	120	0.06	0.27
QR4	QAM-RX10-433	1	4.38	2.46	120	0.03	0.13
QT4	QAM-TX2-433	1	5.51	2.63	120	0.04	0.18
W31	W3127	2	1.90	0.88	120	0.03	0.13
D33	DS3231SN	1	4.06	2.80	120	0.03	0.13
CFK	CLX6F-FKC	1	0.34	0.26	120	0.01	0.04
BC8	BC847B	4	0.16	0.06	120	0.01	0.04
USB	USB4085	1	0.75	0.62	120	0.01	0.04
UC6	USBLC6-2SC6	1	0.65	0.29	120	0.01	0.04
LDL	LDL1117S33R	1	1.38	0.79	120	0.01	0.04
MP0	MP008293	1	0.11	0.05	120	0.01	0.04
MFM	MF-MSMF075/24	1	0.32	0.30	120	0.01	0.04
1N5	1N5819HW-7-F	1	0.53	0.10	120	0.01	0.04
KMR	KMR421G-LFS	2	0.72	0.72	120	0.01	0.04
R85	RESISTORS SMD 0805	19	0.01	0.01	120	0.01	0.04
C85	CAPACITORS SMD 0805	12	0.09	0.01	120	0.01	0.04
I85	INDUCTORS SMD 0805	4	0.86	0.50	120	0.02	0.09
SERVICES			**18.20**	3.85			
APB	Man. PCB (AISLER)	1	9.56	3.11	120	0.08	0.36
PPK	Man. PACKAGING (PACKLY)	1	8.64	0.74	120	0.07	0.31
TOOLS			**676.55**				
E3D	ESP32-S3-DEVKIT1C-1-N32R8V	1	17.97		120	0.15	0.67
AA3	ACER ASPIRE 3	1	349.00		60	5.81	26.14
AN8	AN8009	1	50.59		120	0.42	1.89
Y93	YIHUA 926 III	1	49.99		60	0.83	3.73
ER3	ENDER 3	1	209.00		60	3.48	15.66
TOTAL			**730.70**			**11.16**	**50.12**

Table 4: Cost in euros of project hardware components, services and tools needed along with their lifetime in months and amortization in euros. *Source: Compiled by the author.*

ID	NAME	COST
ELE	Electricity	7.87
INT	Internet	135.00
EWR	Electronic workshop rent	105.00
CWR	Coworking rent	192.00
SWE	Software	0.00
TOTAL		439.87

Table 5: Cost in euros of electricity, internet, space rent and software through all the project. *Source: Compiled by the author.*

Risks

- **Components unavailability**

  →Preselect several alternatives

- **Bugs or hardware malfunction**

  →Testing portion in every task

- **Planning failing**

  →Move low priority tasks to future work (starting with the security ones)

- **Blocking complexity**

  →Move to future work

ID	NAME	IMPLICATION	PROB. %	COST	
				FULL	BUDGET
BDK	Broken development kit	Buy again	5	17.97	0.89
BPC	Broken computer	Buy again	1	349.00	3.49
BMU	Broken multimeter	Buy again	1	50.59	0.50
BSI	Broken soldering iron	Buy again	1	49.99	0.49
B3D	Broken 3D printer	Buy again	5	209.00	10.45
BHW	Broken components (all)	Buy again	10	42.14	4.21
UHW	Unavailable components (all)	Find and buy alternatives	33	50.00	16.50
LSW	Software libraries limitations	Find alternatives and refactor (15h)	5	405.00	20.25
BSW	Software bugs	Find and fix (15h)	20	405.00	81.00
FPB	PCB flaw	Fix and remake (10h)	15	270.00	40.50
FCS	Casing flaw	Fix and remake (8h)	10	213.28	21.32
FPK	Packaging flaw	Fix and remake (6h)	10	159.96	15.99
IEX	Inexperience	Learn (10h)	20	253.10	50.62
DEA	Cannot reach deadline	Do extra work (20h)	20	541.65	108.33
TOTAL					374.53

Table 6: Cost in euros and their probability of the possible incidences through all the project.
Source: Compiled by the author.

It is better to have an **MVP** than **nothing**

Regulations

- **Hardware:**
  - Restriction of hazardous substances (RoHS)
  - Electromagnetic compatibility (EMC)
  - Radio equipment directive (RED)
  - Conformitè europëenne (CE)
  - Waste electrical and electronic equipment (WEEE)
  - General product safety directive (GPSD)
  - Common charger directive (CCD)

- **Software:**
  - General data protection regulation (GDPR)
  - Cyber resilience act (CRA)

Foundation

Initial research

GQRX

Universal Radio Hacker

How can we communicate with the 433MHz sensors and actuators?

There is **no common** protocol

Reverse engineer each one

Parameterized generic **abstraction**

RTL-SDR scanner

Initial research - frequency analysis

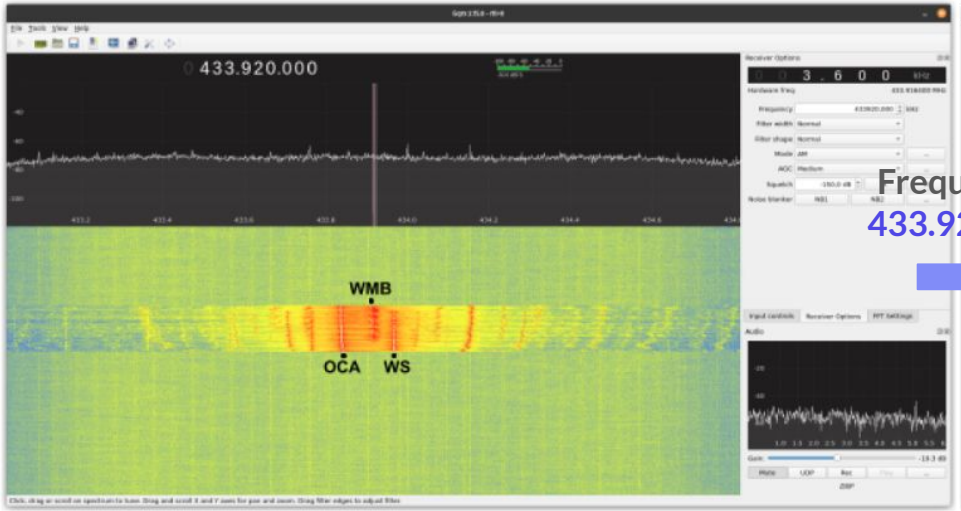


Figure 8: Radio frequency heatmap of the test subjects. Marked by the initials “OCA” refers to the open-close alarm, “WMB” refers to the wireless motorized blinds and “WS” refers to the wireless socket. *Source: Compiled by the author.*

Frequency
433.92MHz

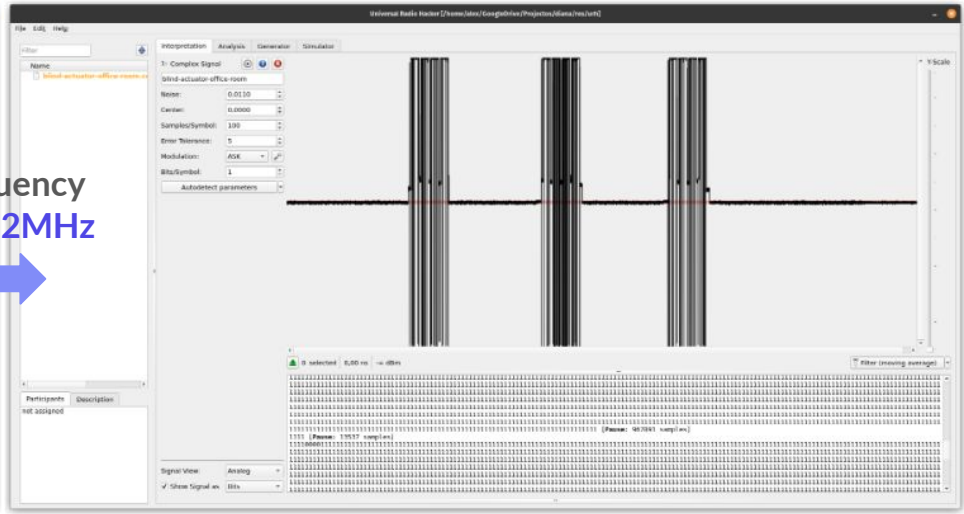


Figure 9: Scanned signals graphic of the wireless motorized blinds controller.
Source: Compiled by the author.

Initial research - modulation analysis

Modulation
ASK

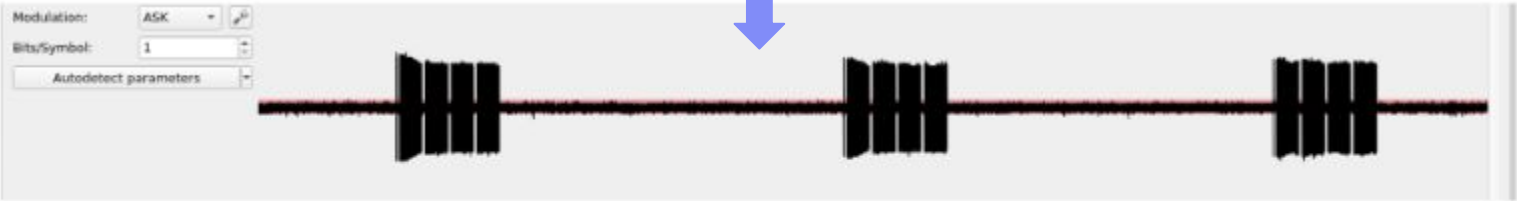


Figure 10: Scanned signals graphic of the wireless motorized blinds controller using ASK modulation. *Source: Compiled by the author.*

Initial research - phase analysis

3 phases

- Sync
  →Beginning of a new packet
- Preamble
  →Tuning of the error tolerance and base pulse width
- Data
  →Encoded sequence of bits

Figure 11: Decomposition of the "up" command signal of the wireless motorized blinds

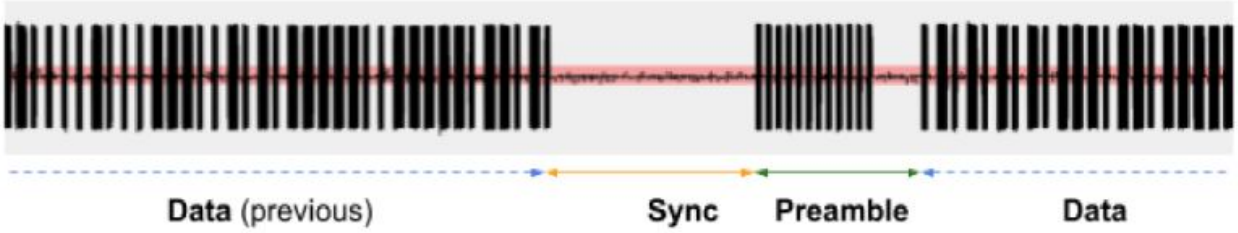


Figure 12: Mixed decomposition of two "up" command signals of the wireless motorized blinds controller. *Source: Compiled by the author.*

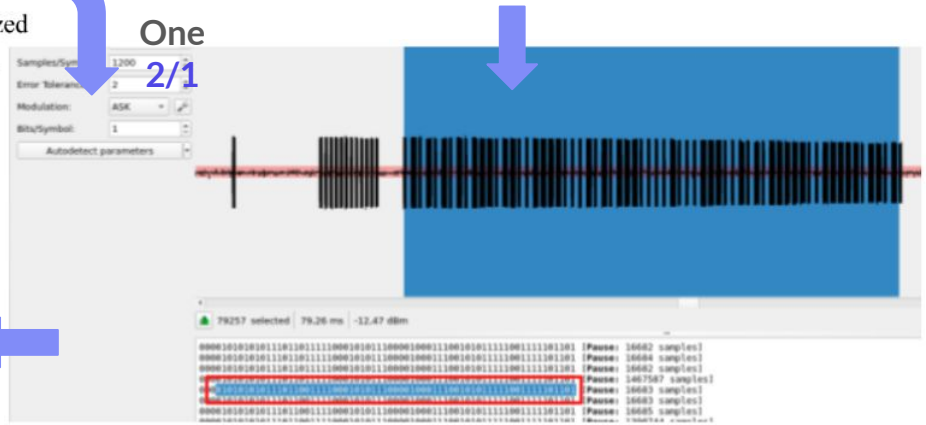
Initial research - data analysis

Figure 13: Data pattern decomposition of the “up” command signal of the wireless motorized blinds controller. On the left the pattern for a “one” bit and on the right for a “zero” bit. *Source: Compiled by the author.*

Zero
1/2

One
2/1

Base pulse width
400µs



0x557607EE11CAF9F6
64 bit-long

Figure 14: Decoded bits of the “up” command signal of the wireless motorized blinds controller (marked by a red rectangle). *Source: Compiled by the author.*

Initial research - data analysis

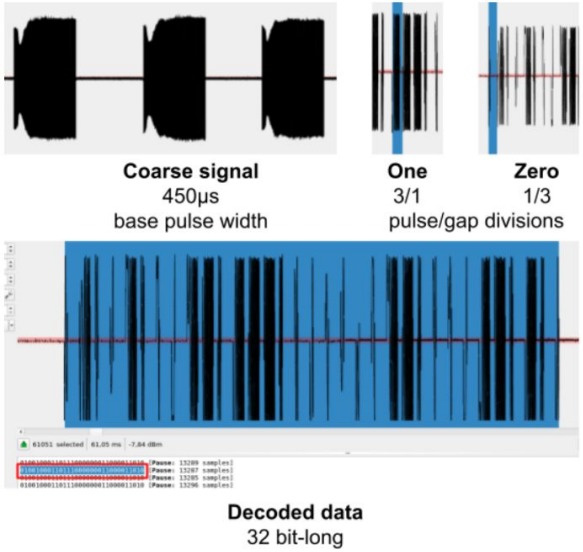


Figure 15: Summarized communication protocol analysis of the open-close alarm.
Source: Compiled by the author.

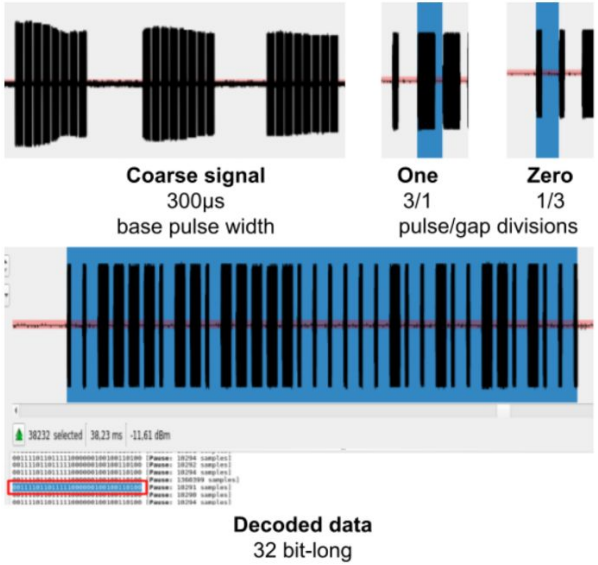


Figure 16: Summarized communication protocol analysis of the wireless socket controller.
Source: Compiled by the author.

- **Open-close alarm**
  - Open: 0x48DC061A
  - Closed: 0x48DC0609
- **Wireless socket**
  - On: 0x3DBE049A
  - Off: 0x3DBE0892
- **Wireless motorized blinds**
  - Roll up: 0x557607EE11CAF9F6
  - Stop: 0x55797BEE11CAF9F9
  - Roll down: 0x557D3BEE11CAF9FD

Proposed system

- Electronic device formed by a Wi-Fi-capable microcontroller, running a RTOS, with a pair of 433MHz-tuned antennas along with a transmitter and receiver modules
- Multiple users, using a built-in web app, interacturate with the gateway through the Internet, intranet or directly
- Gateway communicates with the 433MHz devices using a generic protocol with the parameters required for each one
- Gateway maintains the time using a built-in real-time clock module with an Internet clock synchronization

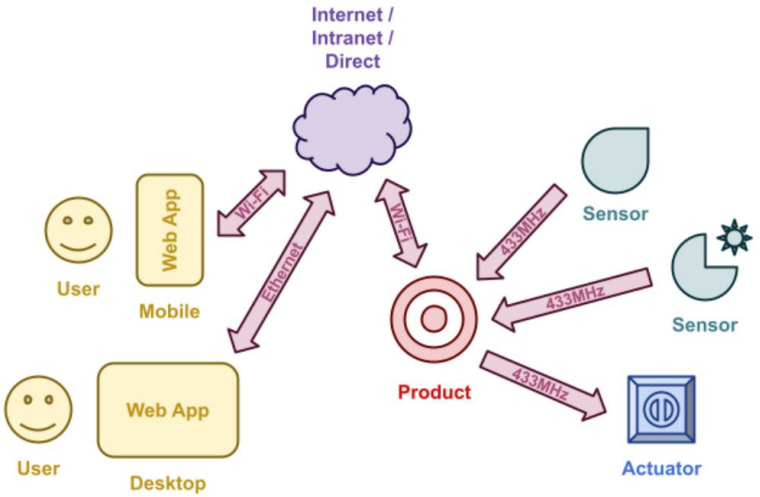


Figure 17: Conceptual diagram of the system. *Source: Compiled by the author.*

Taken approach

- Code name Diana Dot
- Aesthetic bulls-eye
- Not bigger than the palm of a hand
- Powered through a 5V / 1A USB-C cable
- Not preserve the time on a power reset
- Visual cue for internet connectivity and additional feedback
- Initial Internet credentials provisioning using a built-in captive portal

Hardware components

- Dual core 32-bit LX7 240MHz
- Flash 32MB, SRAM 512KB, PSRAM 8MB
- Integrated Wi-Fi
- System on a Chip
- USB JTAG controller
- 33 GPIOs, all PWM and INTR

Figure 30: ESP32-S3-WROOM-2-N32R8V.
Source: Compiled by the author.

Microcontroller

NAME	UNITS	COST		VOLTAGE		CURRENT	
		x1	x1000	OPERATING	MAX	DRAWN	MAX
ESP32-S3-WROOM-2-N32R8V	1	7.34	7.34	3.3	3.6	355.0	1,500.0
QAM-RX10-433	1	4.38	2.46	5.0	5.5	6.0	600.0
QAM-TX2-433	1	5.51	2.63	5.0	12.0	15.0	850.0
W3127	2	1.90	0.88		46.0		920.0
DS3231SN	1	4.06	2.80	3.3	6.0	0.7	1,000.0
CLX6F-FKC	1	0.34	0.26	5.0	5.0	35.0	100.0
BC847B	4	0.16	0.06		45.0	10.0	200.0
USB4085	1	0.75	0.62		20.0		3,000.0
USBLC6-2SC6	1	0.65	0.29		15,000.0	0.15	2,500.0
LDL1117S33R	1	1.38	0.79		18.0	0.5	1,200.0
MP008293	1	0.11	0.05	5.0	5.0	20.0	75.0
MF-MSMF075/24	1	0.32	0.30		24.0		1,500.0
1N5819HW-7-F	1	0.53	0.10		40.0	0.05	1,500.0
KMR421G-LFS	2	0.72	0.72		32.0		50.0
RESISTOR 0Ω	3	0.01	0.01		300.0		3,500.0
RESISTOR 22Ω	2	0.01	0.01		300.0		3,500.0
RESISTOR 330Ω	3	0.01	0.01		300.0		3,500.0
RESISTOR 1kΩ	4	0.01	0.01		300.0		3,500.0
RESISTOR 4.7kΩ	2	0.01	0.01		300.0		3,500.0
RESISTOR 5.1kΩ	3	0.01	0.01		300.0		3,500.0
RESISTOR 10kΩ	2	0.01	0.01		300.0		3,500.0
CAPACITOR 0.1µF	5	0.01	0.01		100.0		25,000.0
CAPACITOR 1µF	2	0.01	0.01		25.0		6,250.0
CAPACITOR 4.7µF	1	0.02	0.01		10.0		2,500.0
CAPACITOR 10µF	3	0.3	0.01		16.0		4,000.0
CAPACITOR 22µF	1	0.05	0.01		6.3		1,575.0
INDUCTOR 6.8nH	2	0.31	0.9		49.0		800.0
INDUCTOR 12nH	2	0.63	0.30		42.0		700.0

Table 9: Bill of materials, electrical characteristics and absolute maximum ratings. Cost in euros, voltage in volts and current in milliamperes. *Source: Compiled by the author.*

54 components

Hardware components

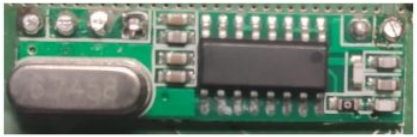


Figure 31: QAM-RX10-433.
Source: Compiled by the author.

Receiver

Figure 32: QAM-TX2-433.
Source: Compiled by the author.

Transmitter

Antennas

Figure 33: W3127.
Source: Compiled by the author.

Real Time Clock

Figure 34: DS3231SN.
Source: Compiled by the author.

RGB LED

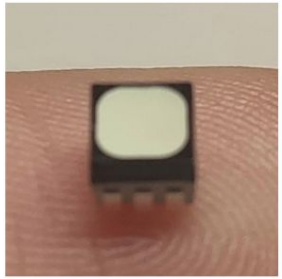


Figure 35: CLX6F-FKC.
Source: Compiled by the author.

Implementation

Schematic - conceptual

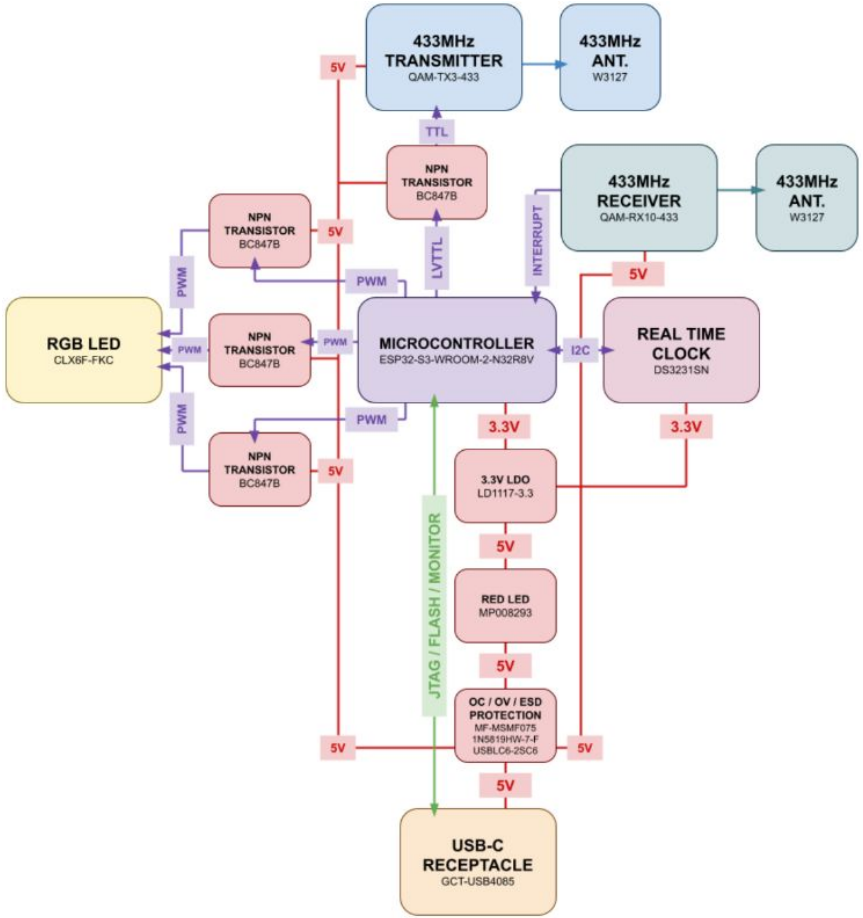


Figure 18: Conceptual diagram of the schematic. *Source: Compiled by the author.*

Schematic - design

3 problems

- Floating emitters

  →Add pull-down resistors

- Microcontroller random boot

  →Remove offender Capacitor (C3)

- Transmission antenna failing

  →Remove offender Inductor (L2)

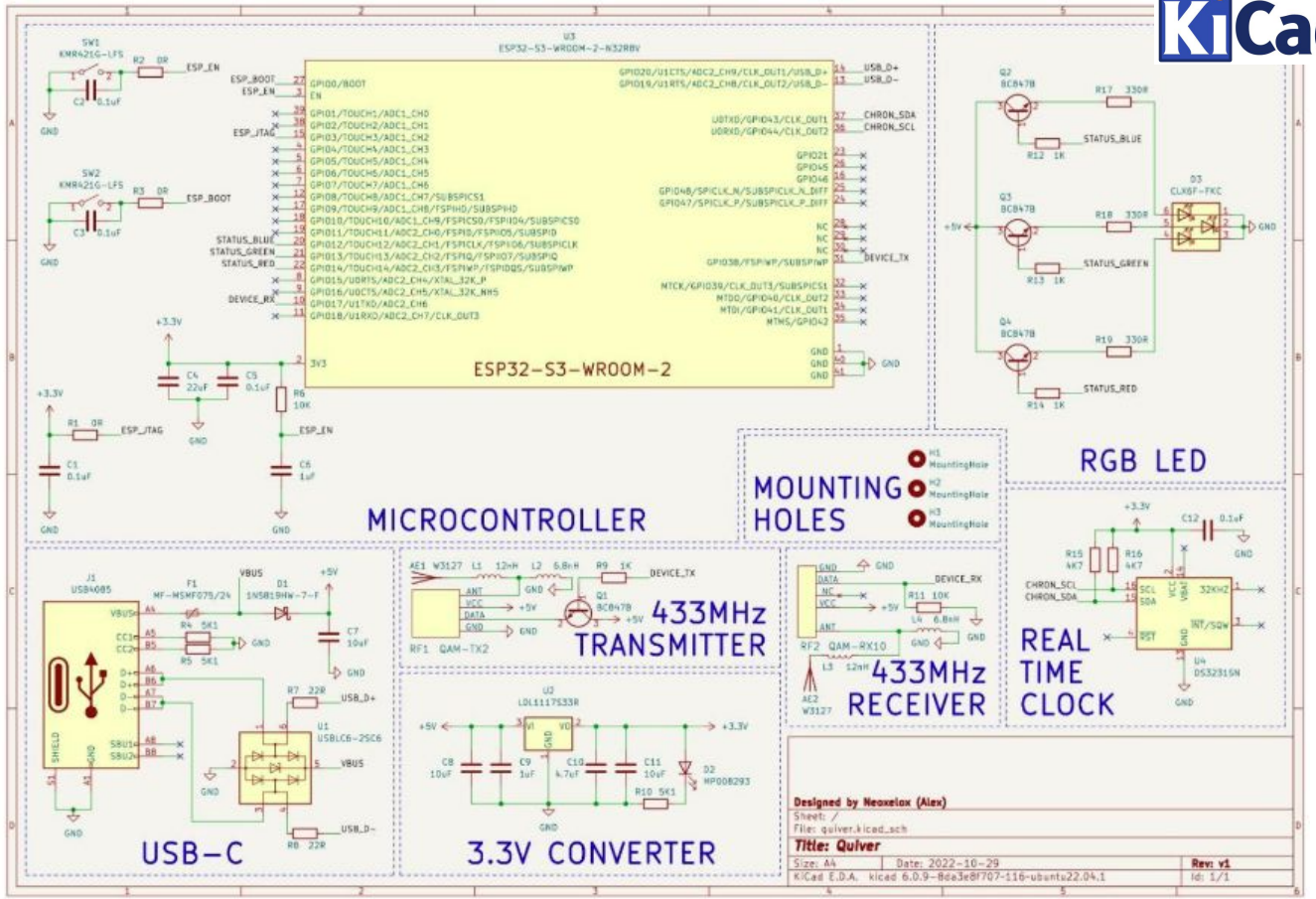


Figure 70: Complete electrical schematic. *Source: Compiled by the author.*

PCB
- conceptual

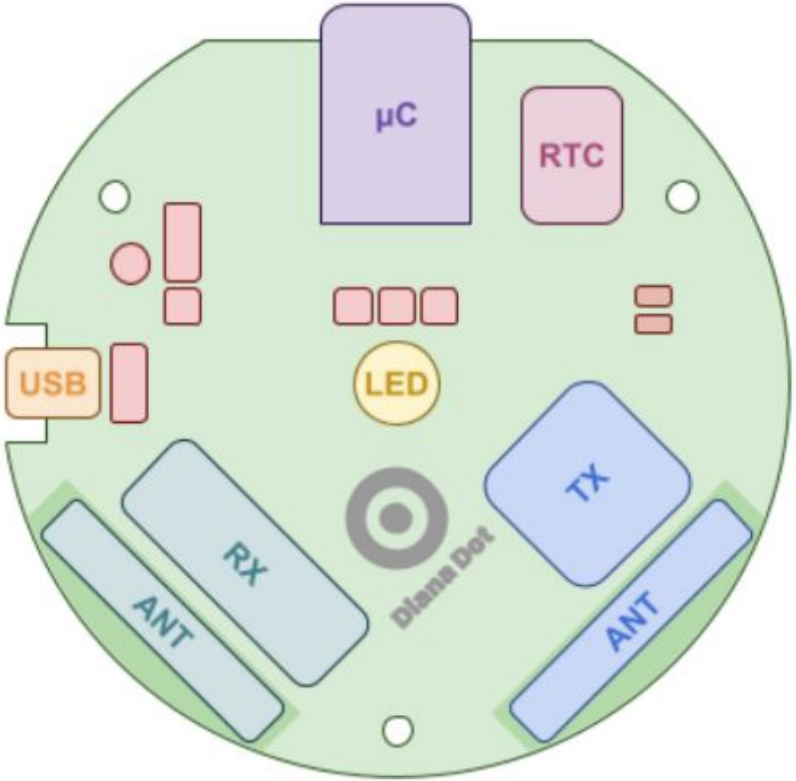


Figure 19: Conceptual diagram of the printed circuit board. *Source: Compiled by the author.*

PCB
- design

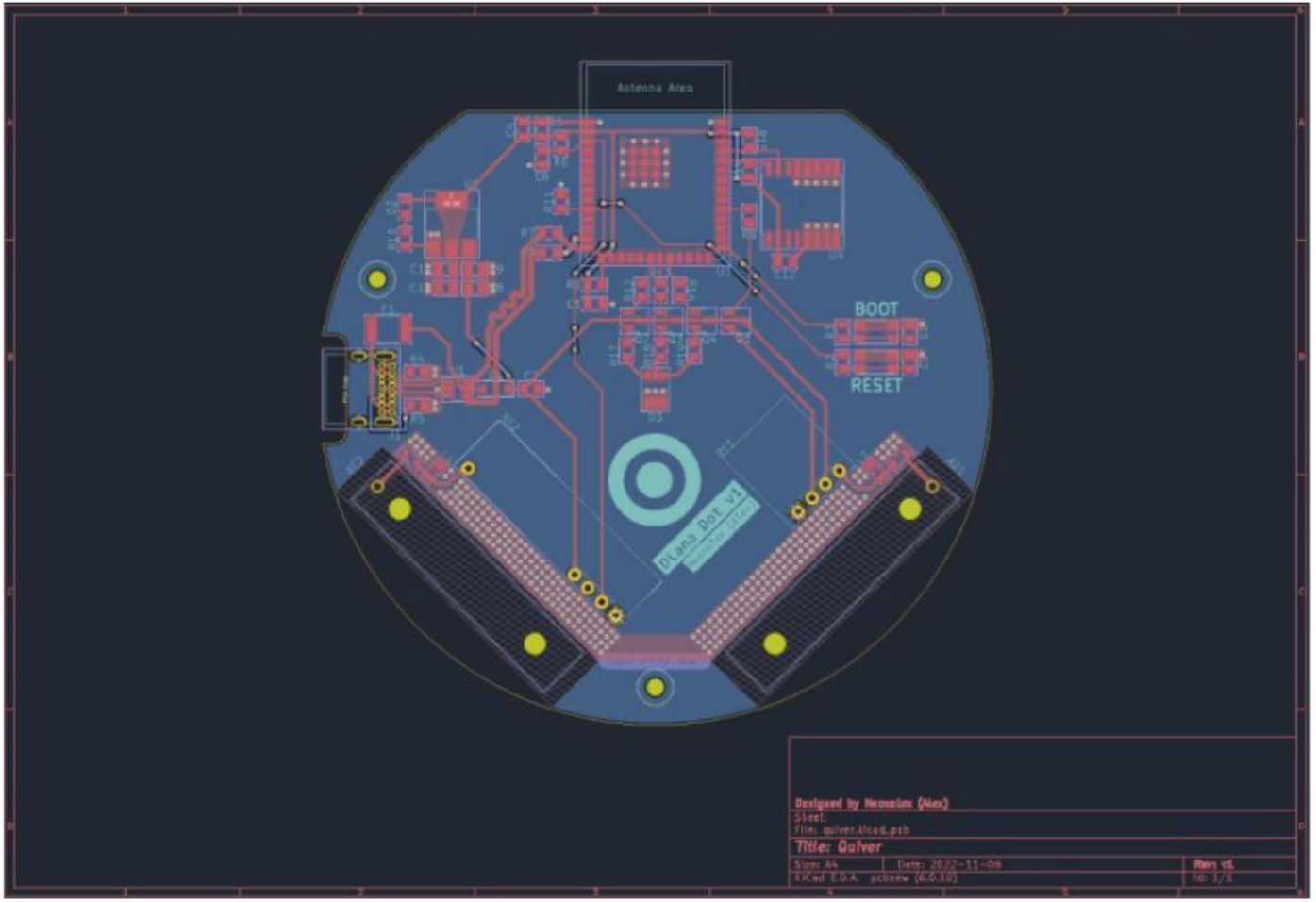


Figure 105: Complete PCB design. *Source: Compiled by the author.*

PCB - preview

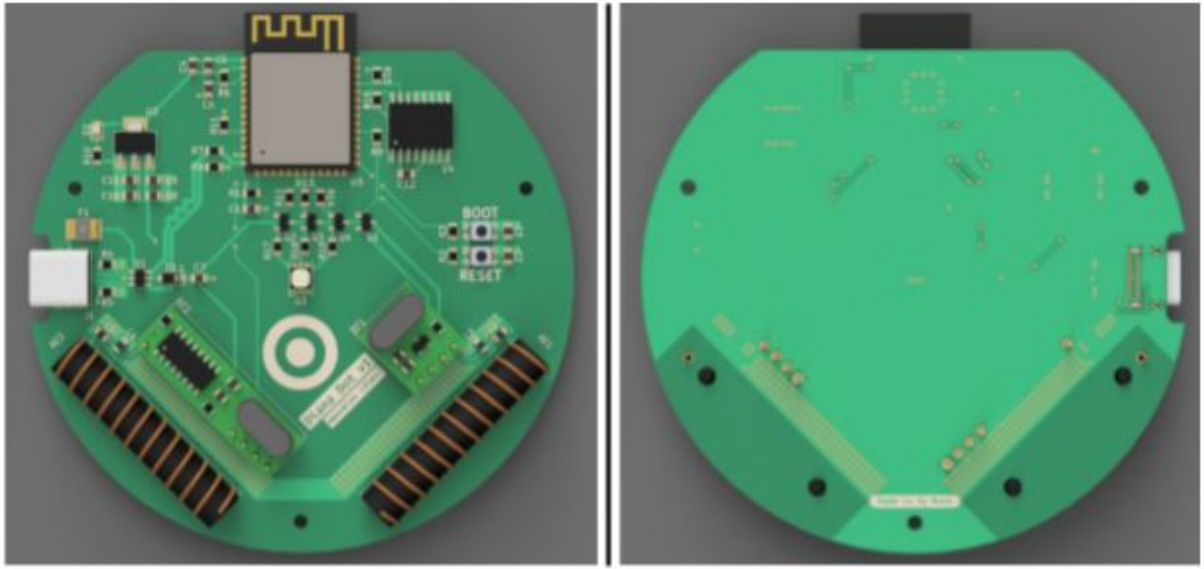


Figure 92: PCB 3D render top and bottom view. *Source: Compiled by the author.*

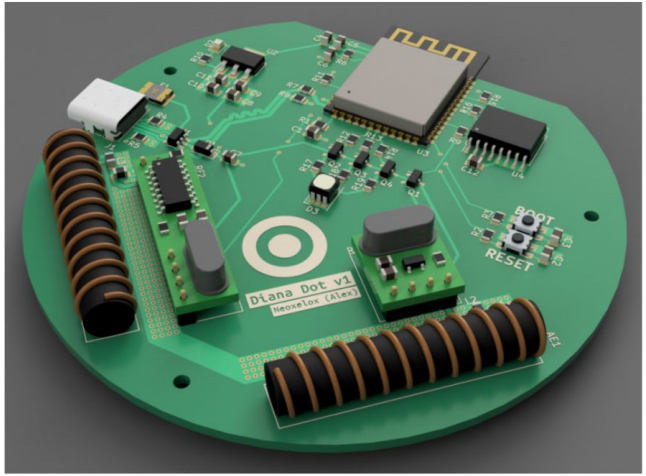


Figure 106: Complete PCB 3D render. *Source: Compiled by the author.*

AISLER

PCB - manufacturing

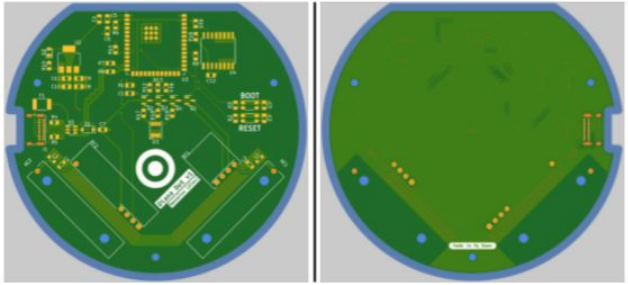


Figure 94: Aisler PCB preview top and bottom views. *Source: Compiled by the author.*

Panelized

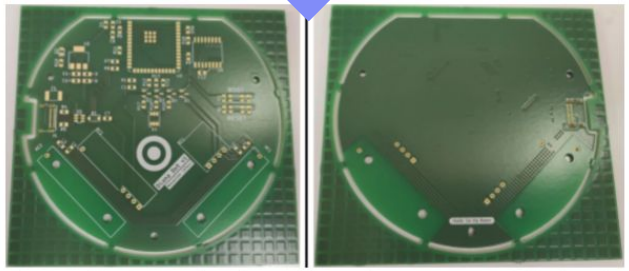


Figure 95: Aisler PCB manufactured top and bottom views. *Source: Compiled by the author.*

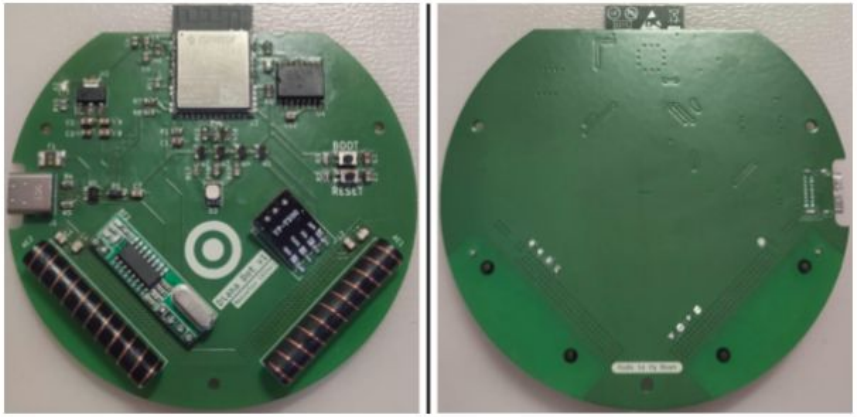


Figure 100: PCB with THT components soldered top and bottom views.
Source: Compiled by the author.

Soldered
THT

Soldered
SMD

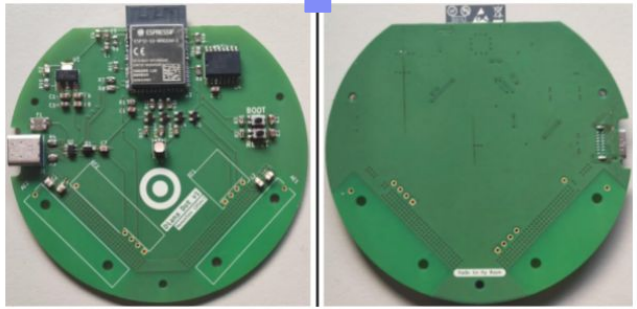


Figure 99: PCB with SMD components soldered top and bottom views.
Source: Compiled by the author.

PCB
- results

2 problems

- Reversed transmitter

  →Solder module reversed

- Transmission antenna failing

  →Add antistatic layer

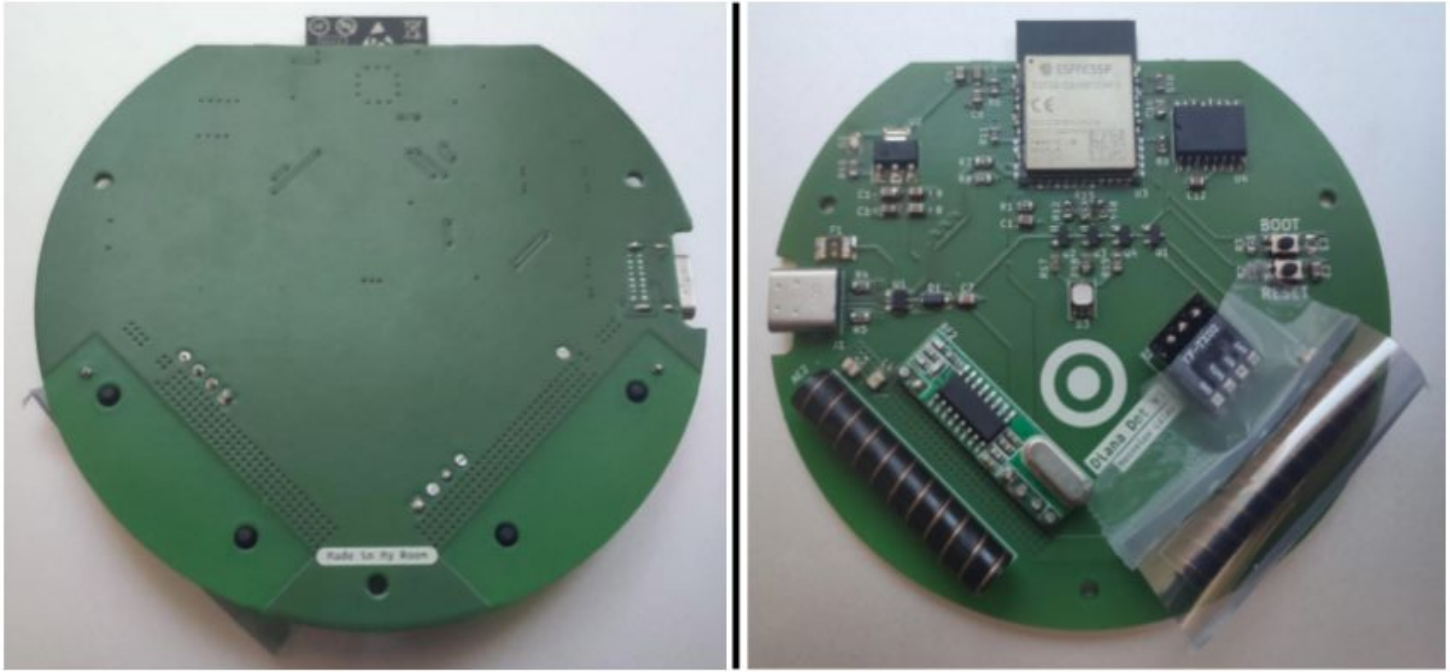


Figure 107: Complete PCB manufactured. *Source: Compiled by the author.*

Casing - conceptual

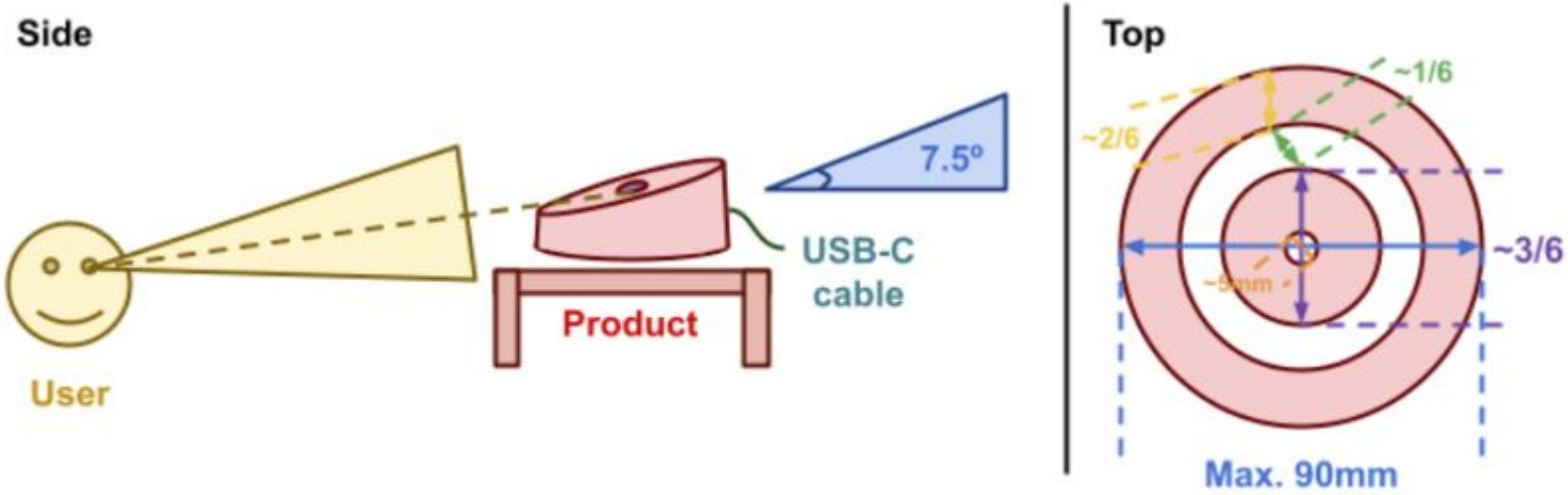


Figure 20: Conceptual diagram of the casing. *Source: Compiled by the author.*

Casing - design

AUTODESK®
FUSION 360™

Figure 109: Lid casing part. *Source: Compiled by the author.*

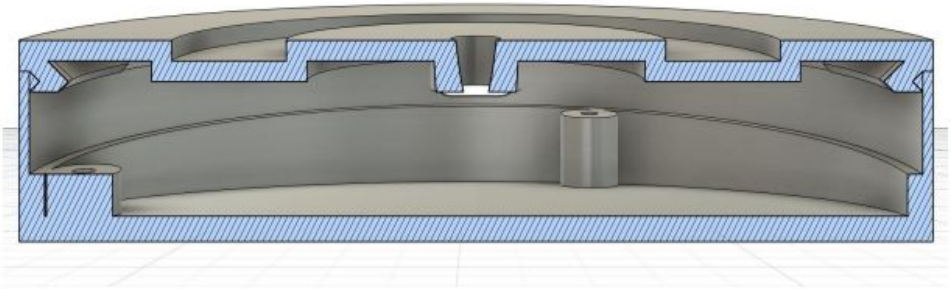


Figure 110: Snap fit closing casing part. *Source: Compiled by the author.*

Figure 108: Body casing part. *Source: Compiled by the author.*

Casing - preview

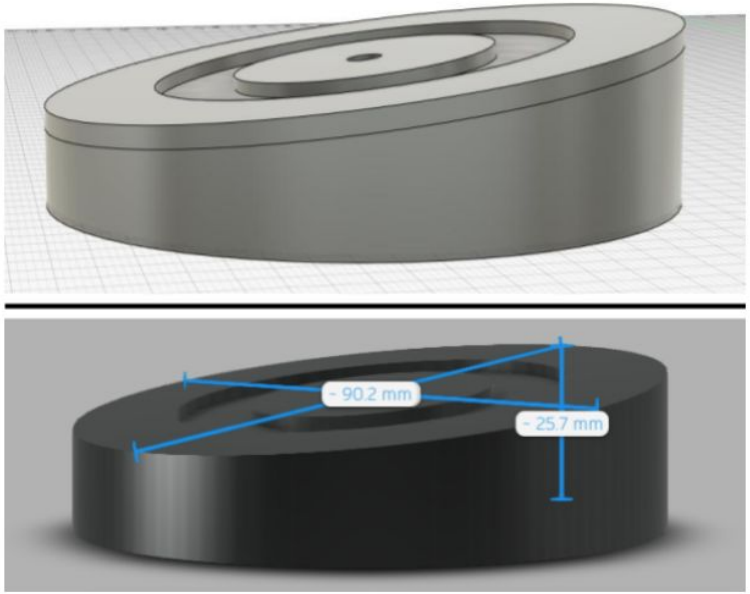


Figure 121: Complete casing design. *Source: Compiled by the author.*

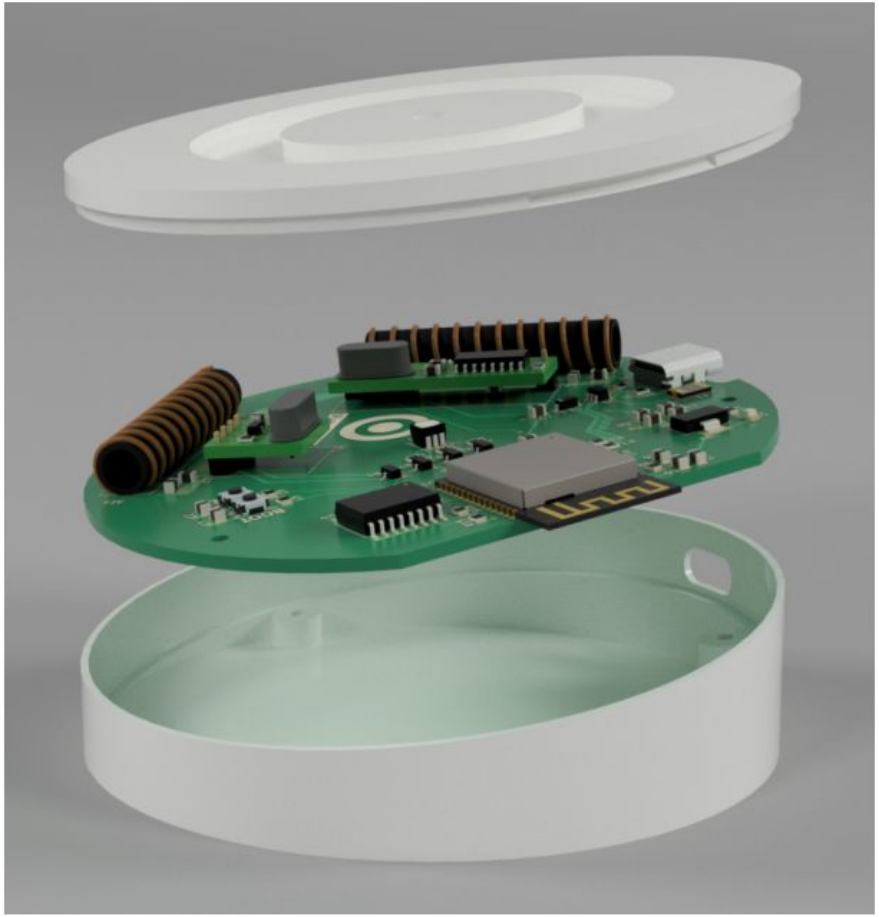


Figure 122: Complete casing 3D render. *Source: Compiled by the author.*

Casing - manufacturing

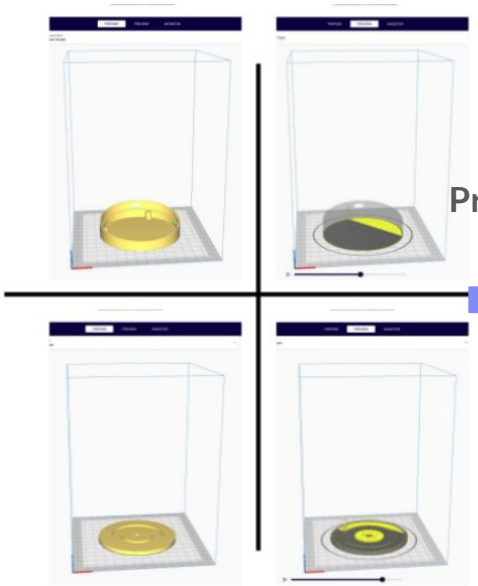


Figure 113: Casing slicing. *Source: Compiled by the author.*

Printed 3D

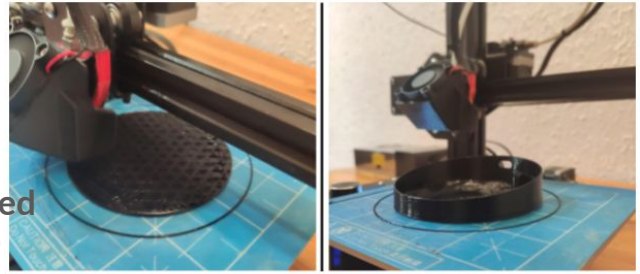


Figure 114: Casing body printing. *Source: Compiled by the author.*

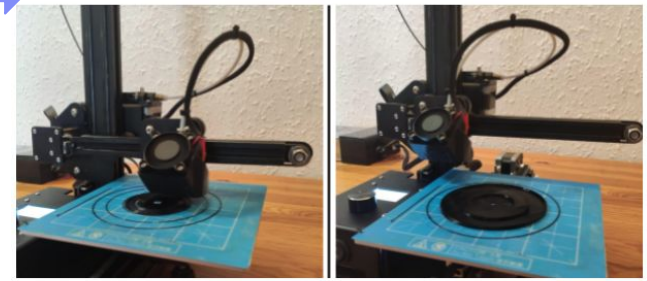
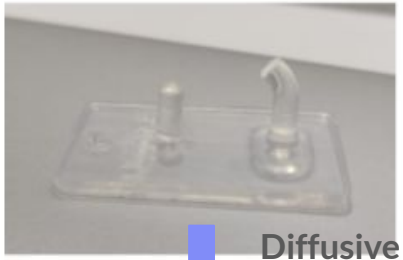


Figure 115: Casing lid printing. *Source: Compiled by the author.*



Diffusive Plastic lens

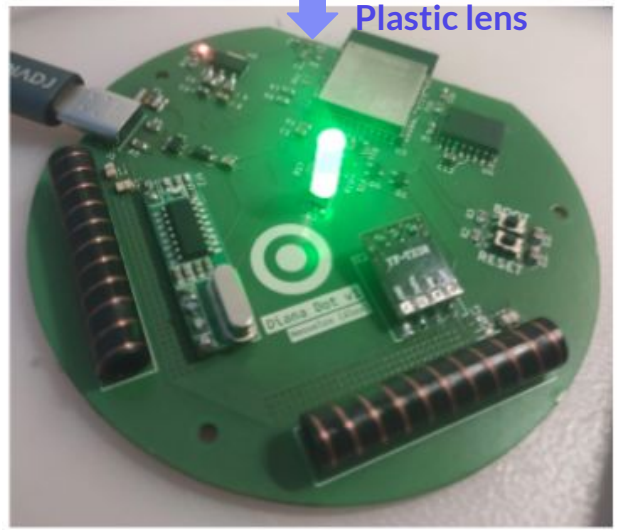


Figure 117: Casing RGB LED plastic lens. *Source: Compiled by the author.*

Casing - results

Figure 119: Testing of the casing. *Source: Compiled by the author.*

Figure 123: Complete casing manufactured. *Source: Compiled by the author.*

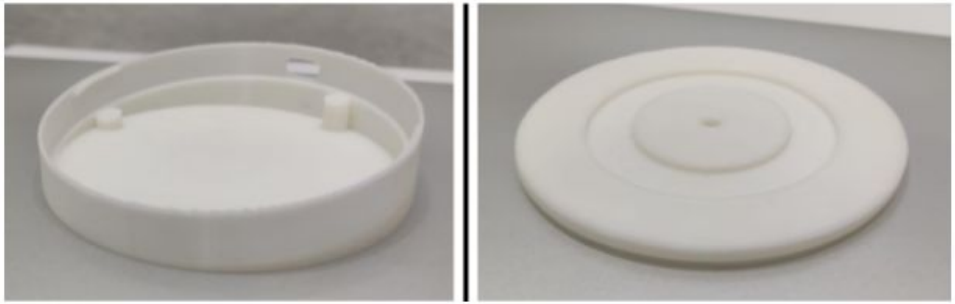


Figure 116: Casing body and lid manufactured. *Source: Compiled by the author.*

Packaging - conceptual

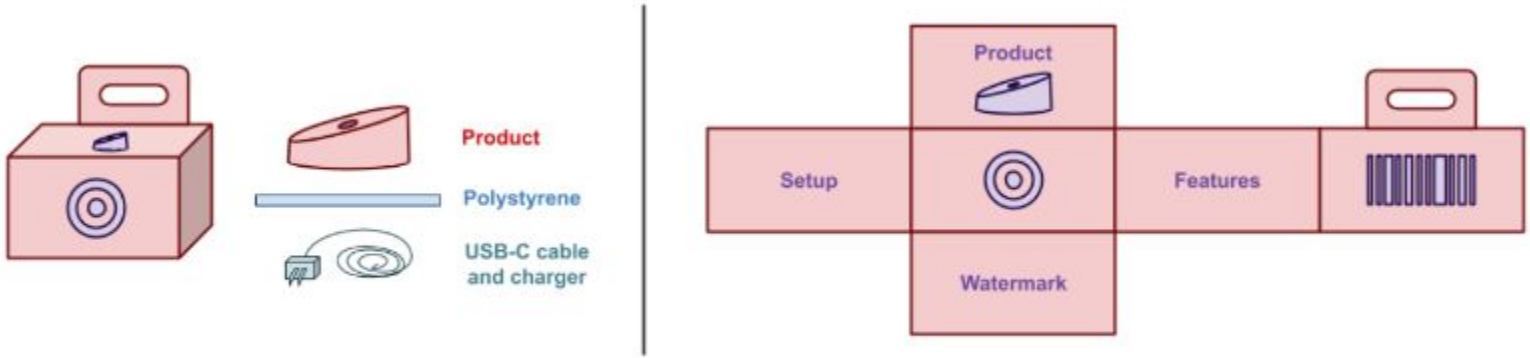


Figure 21: Conceptual diagram of the packaging. *Source: Compiled by the author*.

Packaging - design

Figma

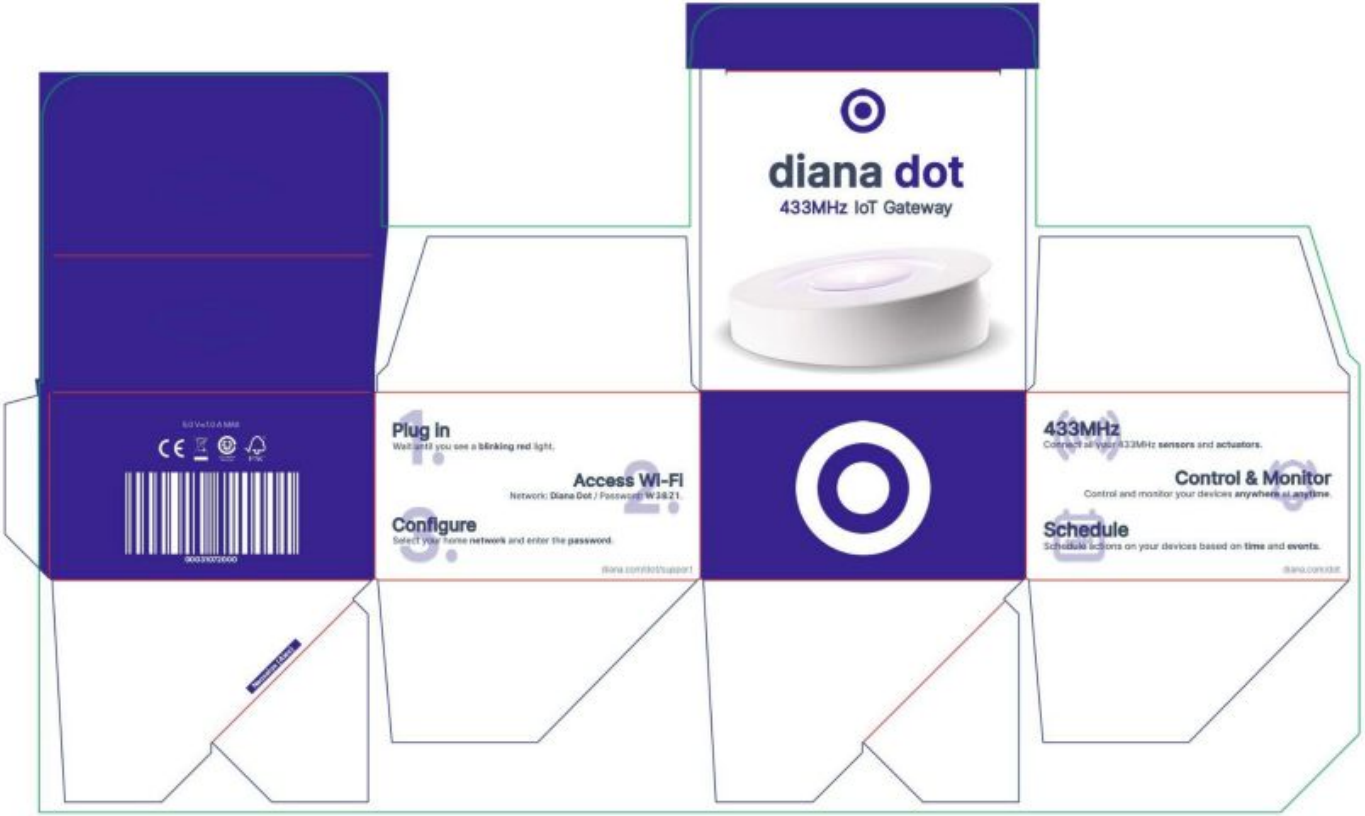


Figure 142: Complete packaging design. *Source: Compiled by the author.*

Packaging - preview

Figure 132: Packaging 3D render left and right side views. *Source: Compiled by the author.*

Figure 133: Packaging 3D render back view. *Source: Compiled by the author.*

packly

Packaging - manufacturing

Figure 135: Packly packaging manufactured. *Source: Compiled by the author.*

Bottom Mount

Add Separator / Charger

Figure 138: Packaging polystyrene separator. *Source: Compiled by the author.*

Hanging holes Fold

Figure 137: Packaging hanging holes. *Source: Compiled by the author.*

Packaging - results

Figure 144: Complete packaging manufactured. *Source: Compiled by the author.*

Figure 140: Testing of the packaging. *Source: Compiled by the author.*

Firmware - conceptual

Backend

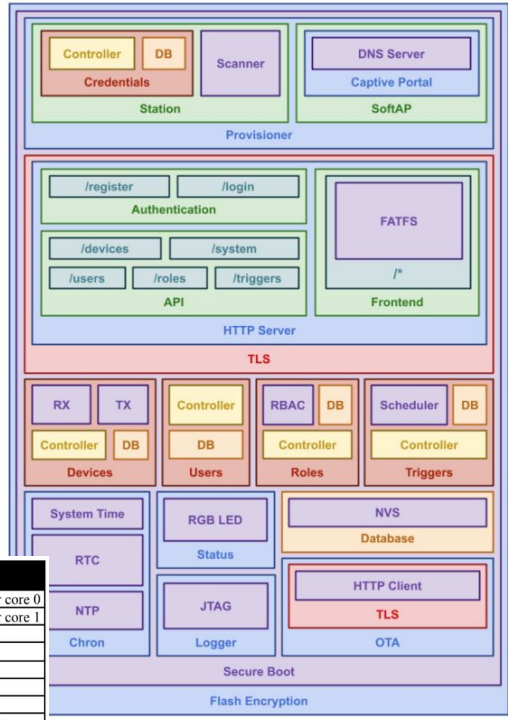


Figure 22: Conceptual diagram of the backend firmware. *Source: Compiled by the author.*

NAME	PRIORITY	CORE	BUILT-IN	COMMENT
ipc0	24	0	YES	Inter-process communication signaling for core 0
ipc1	24	1	YES	Inter-process communication signaling for core 1
wifi	23	0	YES	Wifi driver
esp-timer	22	0	YES	High resolution hardware timer
sys-evt	20	0	YES	System events handler
tiT	18	0	YES	LwIP TCP/IP driver
capdns	16	ANY	NO	Captive portal DNS server
Provisioner	11	ANY	NO	Provisioner Wi-Fi connection retrier
Receiver	10	0	NO	Receiver packet manager
Transmitter	10	1	NO	Transmitter packet manager
httpd	9	ANY	YES	HTTP server
Chron	8	ANY	NO	RTC & NTP time synchronizer
Trigger	7	ANY	NO	Trigger scheduler
Status	6	ANY	NO	Status RGB LED controller
main	1	0	YES	Firmware main entrypoint
Tmr Svc	1	0	YES	Low resolution software timer
IDLE	0	0	YES	Inactive resource liberator for core 0
IDLE	0	1	YES	Inactive resource liberator for core 1

Table 8: Summary of the firmware tasks running on the operating system. *Source: Compiled by the author.*

Code lines

+15K

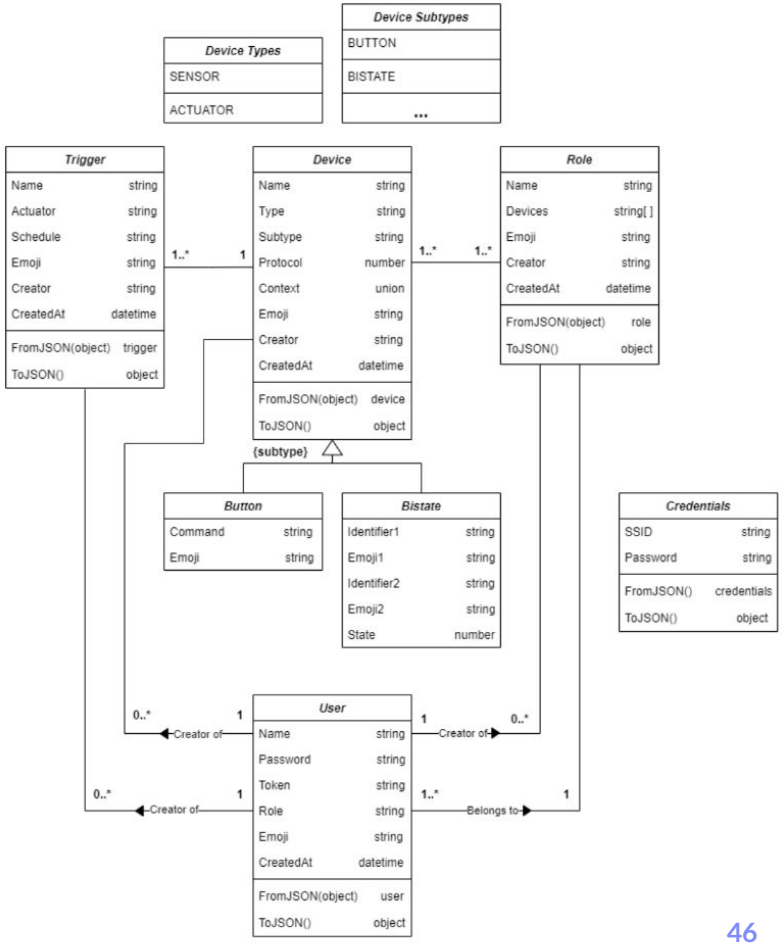


Figure 23: UML diagram of the backend models. *Source: Compiled by the author.*

Firmware - design

Visual Studio Code

ESPRESSIF

GitHub backend neoxelox/bow

GitHub frontend neoxelox/arrow

NAME		COMMENT
BUILD		Contains the intermediate C object files and the final firmware binary
COMPONENTS		Contains the source code for the different software components
	CHRON	Contains the source code for its software component
	DATABASE	
	DEVICE	
	LOGGER	
	PROVISIONER	
	ROLE	
	SERVER	
	STATUS	
	TRIGGER	
	USER	
DOCS		Contains HTTP REST API documentation
MAIN		Contains the source code for the main entrypoint
SCRIPTS		Contains build, lint and monitoring scripts written in Python
STATIC		Contains the embedded frontend web application
	FRONTEND.GZ	
VENDOR		Contains the source code for the different third-party libraries used
	CAPDNS	Captive portal DNS server by Espressif
	CCRONEXPR	Cron expression parsing by Staticlibs
	DS3231	DS3231 driver by UncleRus
	I2CDEV	I2C driver by UncleRus
CMAKELISTS.TXT		CMake definitions for compiling and linking the binary
PARTITIONS.CSV		Partitions definitions for the microcontroller
SDKCONFIG		Configuration definition for the toolchain

Table 11: Backend firmware project structure. *Source: Compiled by the author.*

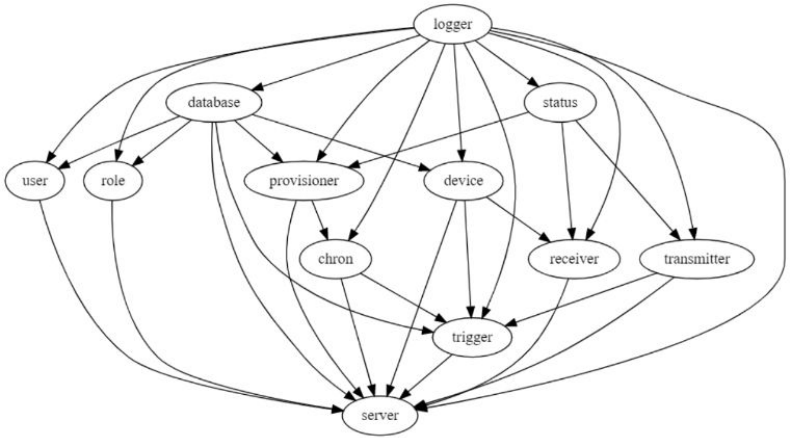


Figure 145: Backend firmware dependency graph. *Source: Compiled by the author.*

Firmware - design

```
class Protocol
{
public:
    struct Data
    {
        uint8_t Zero[MAX_DATA_ZERO_PULSES]; // Divisions
        uint8_t One[MAX_DATA_ONE_PULSES];   // Divisions
    };

    uint32_t Width;                         // Microseconds
    uint8_t Sync[MAX_SYNC_PULSES];          // Divisions
    uint8_t Preamble[MAX_PREAMBLE_PULSES]; // Divisions
    Data Data;
};
```

```
class Packet
{
public:
    uint8_t Protocol;
    char Data[MAX_DATA_PULSES/MAX_DATA_BIT_PULSES + 1];
};
```

```
static const Protocol PROTOCOLS[] = {
    {400, {1, 42}, {1, …(21)…, 1, 10}, {{1, 2}, {2, 1}}}, // Wireless motor. blinds
    {300, {1, 34}, {}, {{1, 3}, {3, 1}}},                 // Wireless socket
    {450, {1, 29}, {}, {{1, 3}, {3, 1}}},                 // Open-close alarm
}
```

Firmware - design

Transmitter

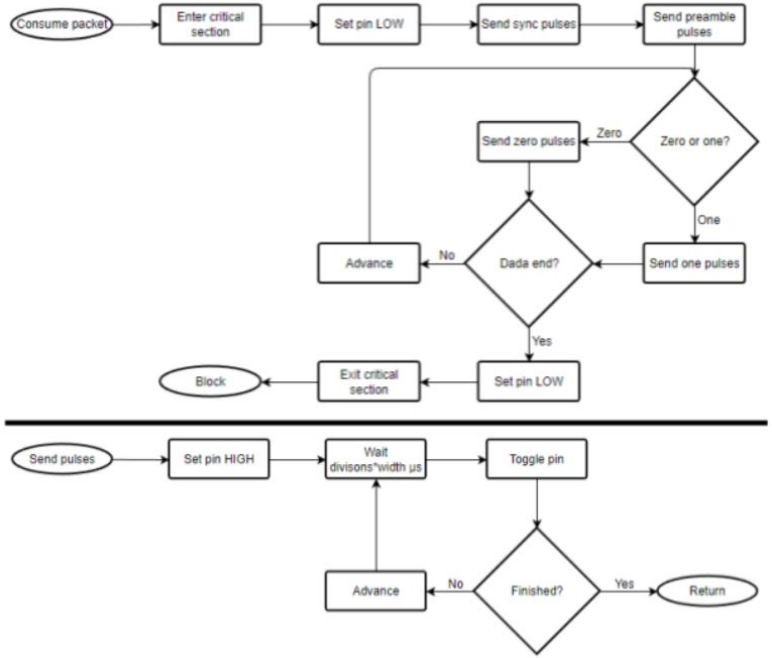


Figure 147: Device transmitter algorithm. *Source: Compiled by the author.*

Receiver

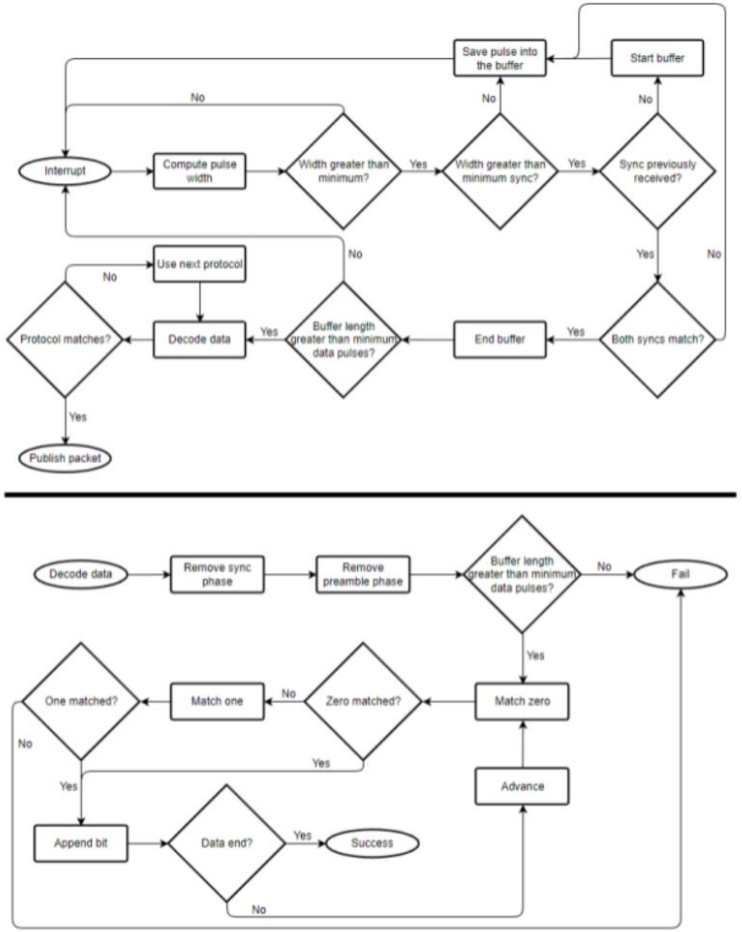


Figure 146: Device receiver algorithm. *Source: Compiled by the author.*

Firmware
- conceptual

Frontend

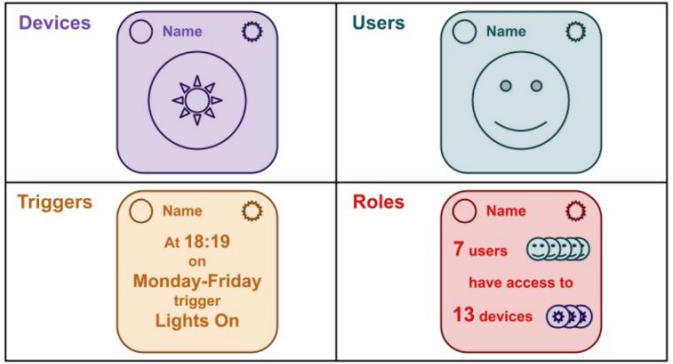


Figure 26: Conceptual diagram of the frontend devices, triggers, users and roles sections.
Source: Compiled by the author.

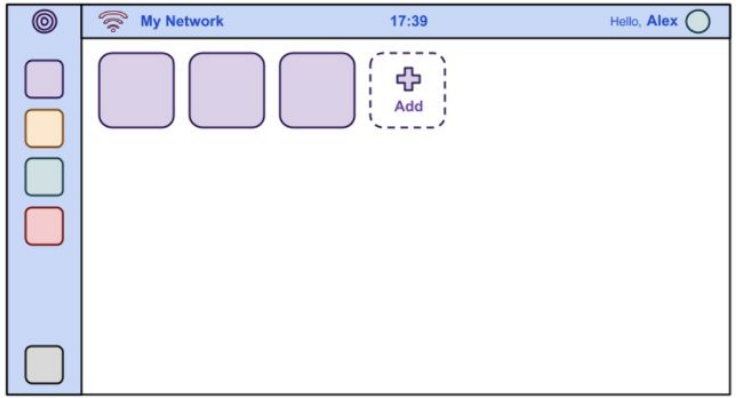


Figure 25: Conceptual diagram of the frontend dashboard page.
Source: Compiled by the author.

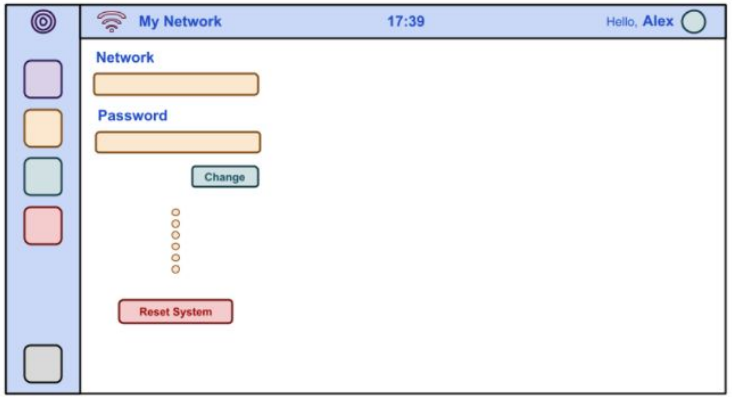


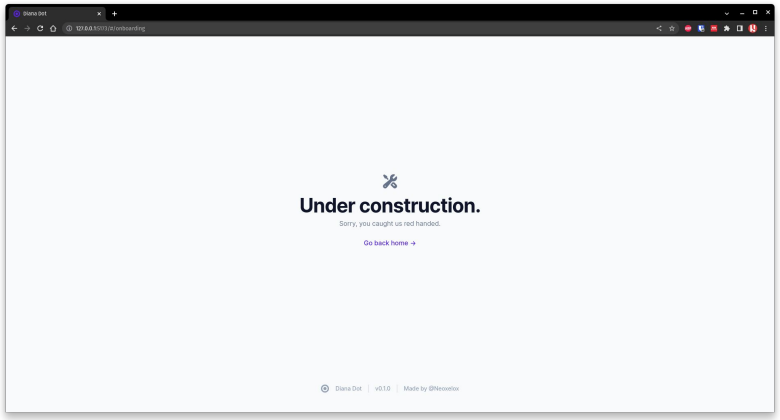
Figure 27: Conceptual diagram of the frontend system page.
Source: Compiled by the author.

Firmware - results

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL

D (28126) device: Rx | Data: 01001000110111000000011000011010 | Protocol: 3
D (28136) device: Received data from Alarm
D (28286) device: Rx | Data: 01001000110111000000011000011010 | Protocol: 3
D (28286) device: Received data from Alarm
D (28436) device: Rx | Data: 01001000110111000000011000011010 | Protocol: 3
D (28436) device: Received data from Alarm
D (62206) trigger: Actuator Lights-On triggered by Wake
D (62356) device: Tx | Data: 00111101101111100000010010110100 | Protocol: 2
D (68416) device: Rx | Data: 00111101101111100000111010101010 | Protocol: 2
D (71206) device: Rx | Data: 00111101101111100000101110010000 | Protocol: 2
D (71306) device: Rx | Data: 00111101101111100000101110010000 | Protocol: 2
D (72026) device: Rx | Data: 00111101101111100000101010010010 | Protocol: 2
D (72126) device: Rx | Data: 00111101101111100000101010010010 | Protocol: 2
D (122216) trigger: Actuator Lights-Off triggered by Sleep
D (122376) device: Tx | Data: 00111101101111100000100010010100 | Protocol: 2
```

Figure 157: Complete backend firmware. *Source: Compiled by the author.*

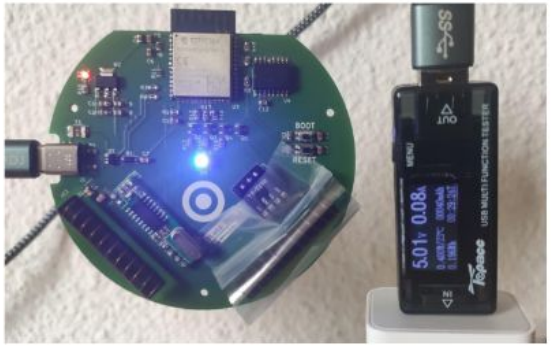


Results

Demonstration

Live!

Figure 158: Complete Thesis project. *Source: Compiled by the author.*



40mAh
0.4W

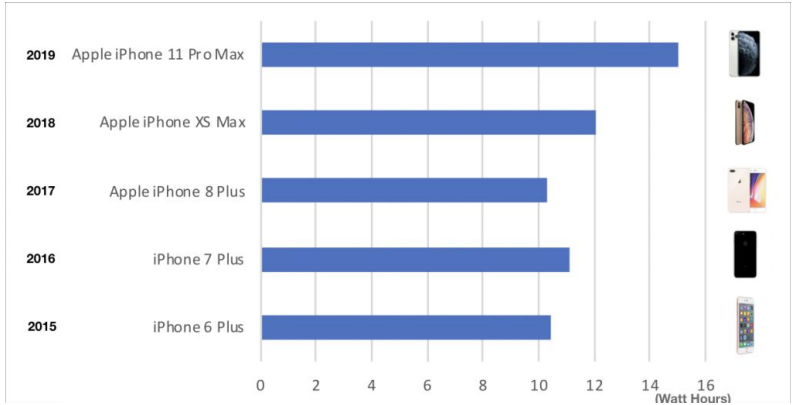


Figure 159: Power consumption of the device. *Source: Compiled by the author.*

ene 12 15:30
Update - bow - Visual Studio Code
Send
Status:
Size:
Time:
Body
Response
Headers
Cookies
Form-encode
Graphql
Binary
Format
As it is 15:30 th

Conclusions

- Electrical schematic:
  - ✓ 2/2 objectives completed
- Printed circuit board:
  - ✓ 3/3 objectives completed
- Casing:
  - ✓ 3/3 objectives completed
- Packaging:
  - ✓ 3/3 objectives completed

- Firmware:
  - ✓ 15/18 objectives completed
  - ○ OTA updates very complex, time consuming and zero rewarding ➡ USB-C JTAG works flawlessly
  - ○ Built-in web dashboard partially achieved ➡ frontend interface not finished
  - ○ TLS layers very complex and too resource intensive for the microcontroller ➡ degrades other functionalities

26/29 objectives achieved

Future work

- Finish objectives
  →Built-in web application interface
  →Over the air updates
- Polish designs
  →Fix flaws presented in the designs themselves
- Reduce power consumption
  →Switch off selectively different parts of the device
- Add features
  →RTC battery with automatic charging system
  →Google Home and Alexa integrations
  →More device protocols
  →...

Questions?

Thanks

Any more questions contact me

alex.rodriguez.navas at estudiantat.upc.edu