



Internship Project: Oleksandra Baga
Eptecon Berlin, 01.03.2018 - 18.07.2018

Noise Detector

The Internet of Things
with ESP32

INTERNSHIP PROJECT: OLEKSANDRA BAGA
EPTECON BERLIN, 01.03.2018 - 18.07.2018

Is your office too loud?

Noise in open plan offices seems to result in higher levels of stress and lower task motivation, according to a new studies of environmental psychologists. These findings suggest that even moderately noisy open offices might contribute significantly to health problems such as heart disease (due to elevated levels of epinephrine, a stress hormone) and musculoskeletal problems¹



1. <https://www.sciencedaily.com/releases/2001/01/010125080258.htm>

Bild: <https://www.pexels.com/photo/adult-business-choices-choosing-515169/>

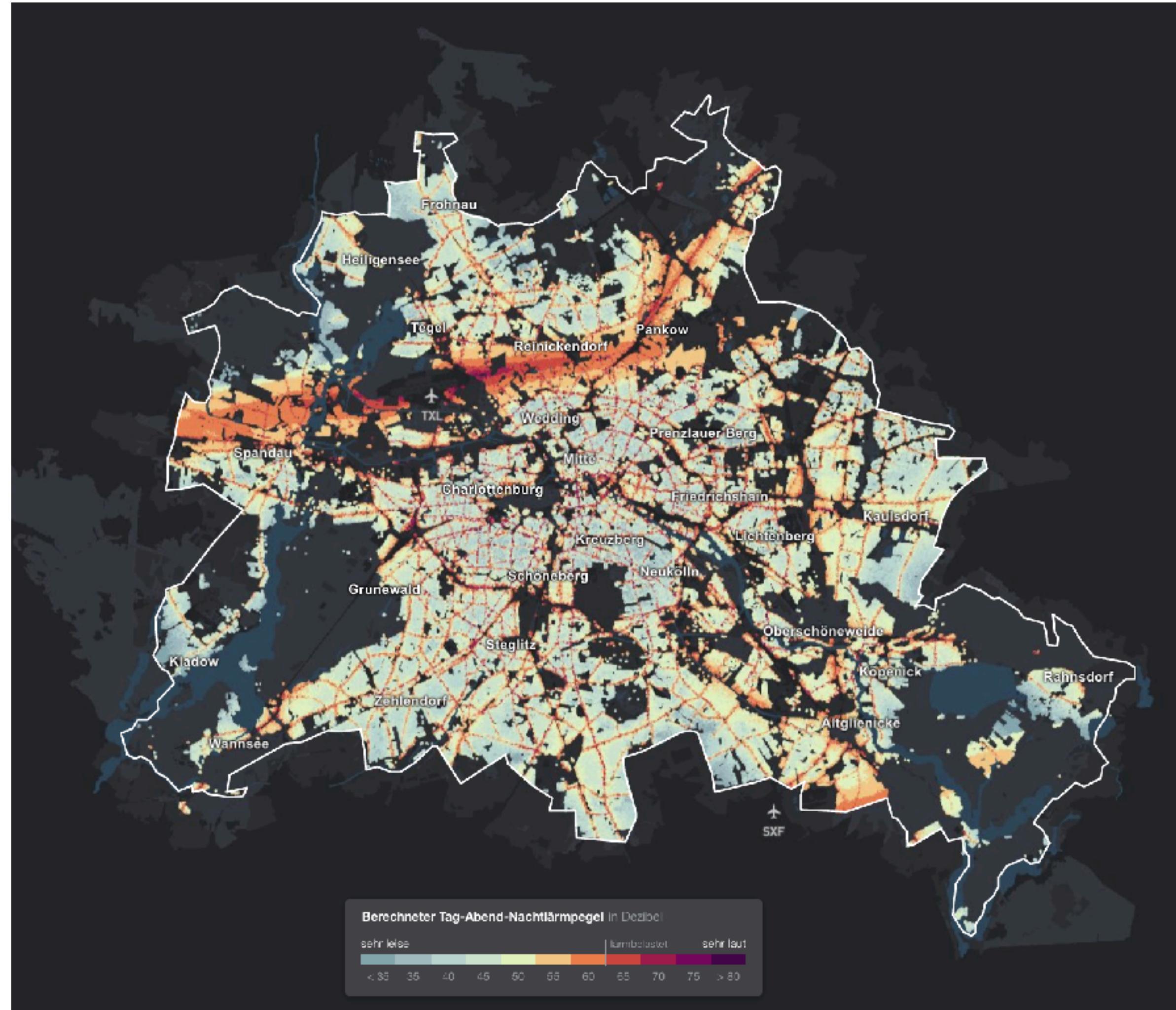


Bild: <https://interaktiv.morgenpost.de/laermkarte-berlin/>

2. <https://www.bbc.com/news/technology-41015486>

Noise in
Berlin city
Berlin never
sleeps

How loud is your street?

Living in a big city near the popular touristic areas or the main roads often means living with unacceptable noise levels for years. The Noise Detector can be used in public places for a data collection and environment improvement on the government level. The similar project was started in Barcelona² and the citizens of the popular Plaza del Sol finally got the night silence after they presented measurements of levels of a night noise to local authorities, all thanks to a new sensor.

Low noise is healthy!

NOISE DETECTOR HELPS TO WORK IN HEALTHY ENVIRONMENT

When the red LED turns on co-workers know that it is time to lower their voice. When the sound notification appears you will definitely understand that it is time to calm down.

Noise Detector is a simple and efficient way of lowering noise levels in the office or public places. It is easy to move from one place to another and get measurements in the new space.

DETECTED NOISE LEVEL APPEARS ONLINE IN THE CLOUD SERVICE

No software installation is required. Just turn on and connect to your WiFi Network. Immediately after connecting, the device starts to send the measured noise level to a cloud service enabling analytics for your working environment, even without invasive notifications.

The red LED and a sound notification make noise visible: your colleagues automatically lower their voices when the light turns red and sound notification turns on.

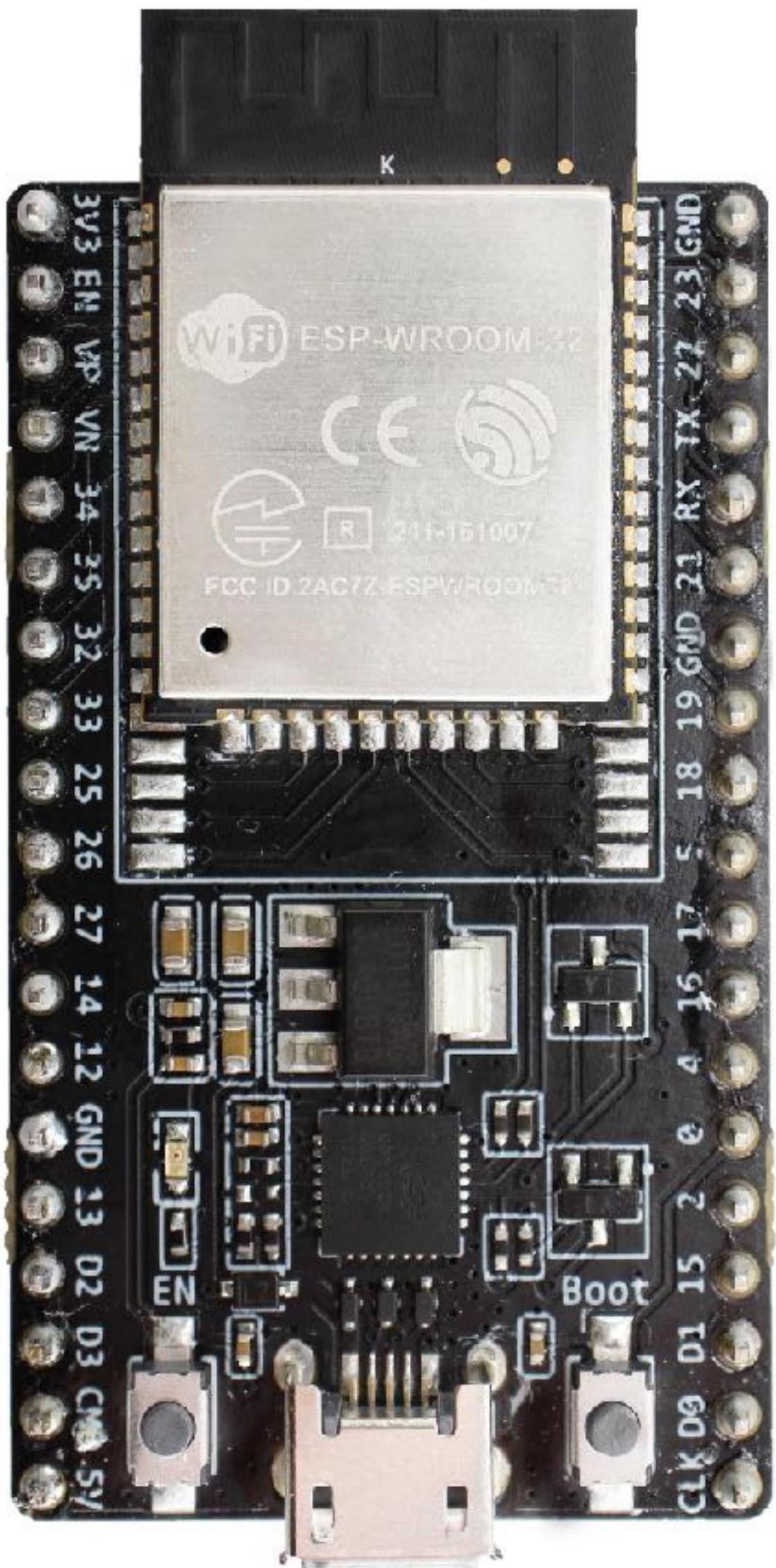
CONNECTED SOLUTIONS BY EPTECON.COM

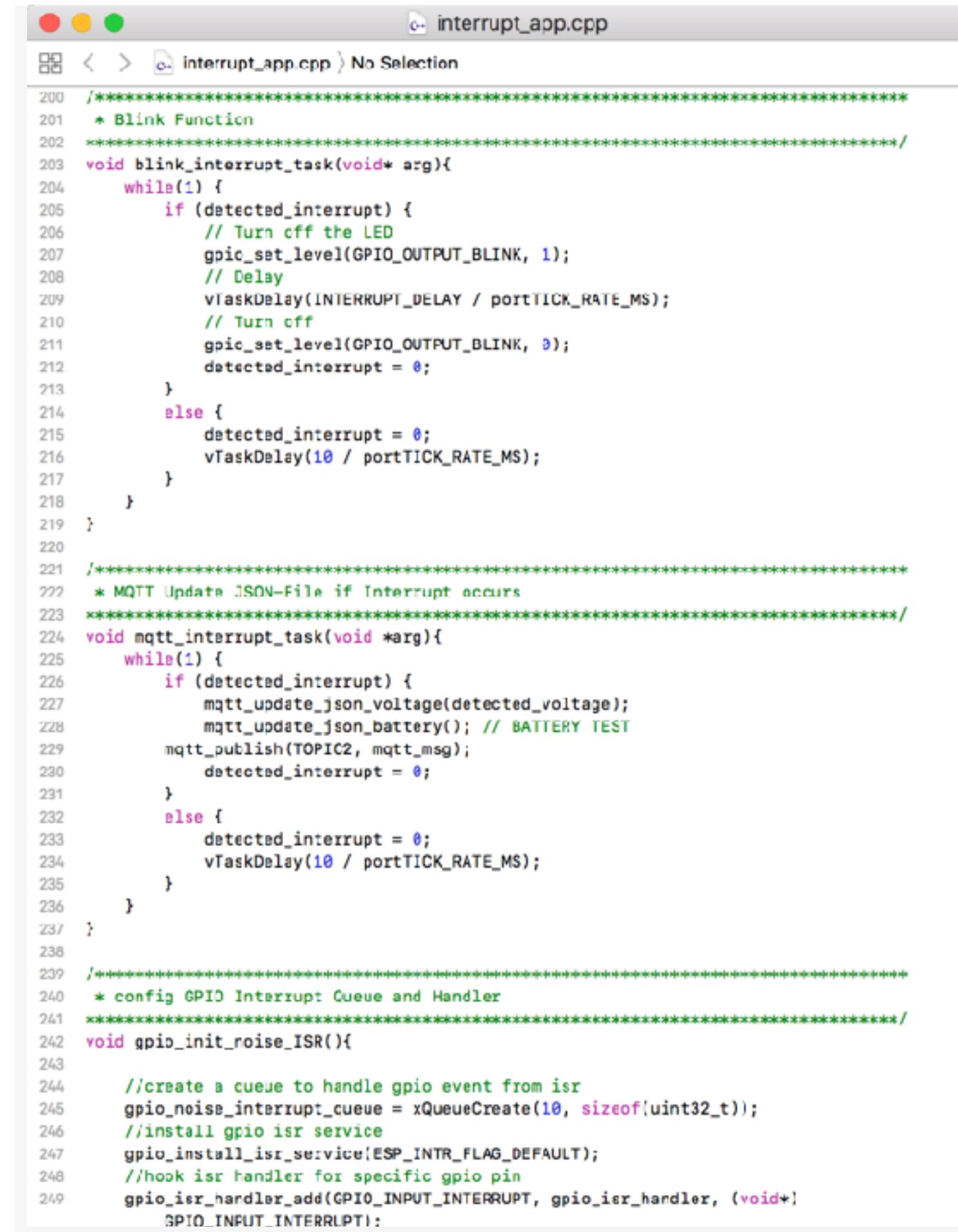
Based on ESP32

**LOW-COST, LOW-POWER SYSTEM ON A CHIP WITH
WI-FI & DUAL-MODE BLUETOOTH**

ESP32 can perform as a complete standalone system and interface with other systems to provide Wi-Fi and Bluetooth functionality through its SPI / SDIO or I2C / UART interfaces.

ESP32-DevKitC is used for Noise Click Project as an entry-level development board. All its GPIO pins are exposed and this allows programming of all necessary functions and connect with sound sensor, LED and buzzer sub-systems.





```
interrupt_app.cpp
=====
200 ****
201 * Blink Function
202 ****
203 void blink_interrupt_task(void* arg){
204     while(1) {
205         if (detected_interrupt) {
206             // Turn off the LED
207             gpic_set_level(GPIO_OUTPUT_BLINK, 1);
208             // Delay
209             vTaskDelay(INTERRUPT_DELAY / portTICK_RATE_MS);
210             // Turn off
211             gpic_set_level(GPIO_OUTPUT_BLINK, 0);
212             detected_interrupt = 0;
213         }
214         else {
215             detected_interrupt = 0;
216             vTaskDelay(10 / portTICK_RATE_MS);
217         }
218     }
219 }
220 ****
221 * MQTT Update JSON-File if Interrupt occurs
222 ****
223 void mqtt_interrupt_task(void *arg){
224     while(1) {
225         if (detected_interrupt) {
226             mqtt_update_json_voltage(detected_voltage);
227             mqtt_update_json_battery(); // BATTERY TEST
228             mqtt_publish(TOPIC2, mqtt_msg);
229             detected_interrupt = 0;
230         }
231         else {
232             detected_interrupt = 0;
233             vTaskDelay(10 / portTICK_RATE_MS);
234         }
235     }
236 }
237 ****
238 * config GPIO Interrupt Queue and Handler
239 ****
240 void gpio_init_noise_ISR(){
241
242     //create a queue to handle gpio event from isr
243     gpio_noise_interrupt_queue = xQueueCreate(10, sizeof(uint32_t));
244     //install gpio isr service
245     gpio_install_isr_service(ESP_INTR_FLAG_DEFAULT);
246     //hook isr handler for specific gpio pin
247     gpio_isr_handler_add(GPIO_INPUT_INTERRUPT, gpio_isr_handler, (void*)
248         GPIO_INPUT_INTERRUPTPI);
```

CONNECTED SOLUTIONS BY EPTECON.COM

Written in C/C++

WITH FREE PROFESSIONAL REAL TIME OPERATING SYSTEM FOR MICROCONTROLLERS

In order to fulfil its goal of noise level monitoring, Noise Detector has to perform several parallel tasks, sending information to the cloud and producing active notifications to coworkers. FreeRTOS (free realtime operating system) allowed to implement this functionality using the Espressif toolchain.

FreeRTOS was created in C with many additional packages and drivers written in C++ and therefore these languages were picked among others for Noise Click development.

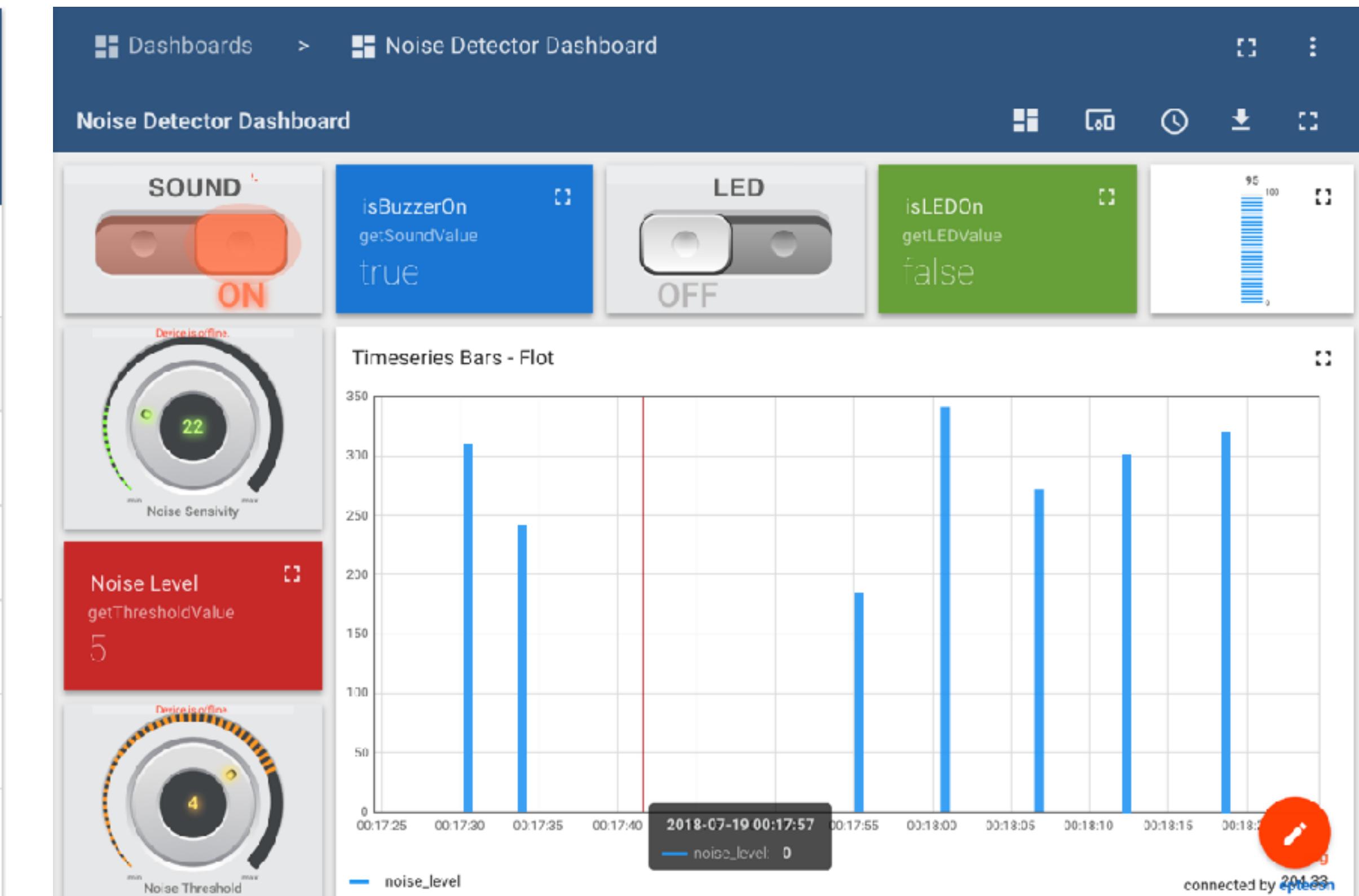
Cloud GlueLogics

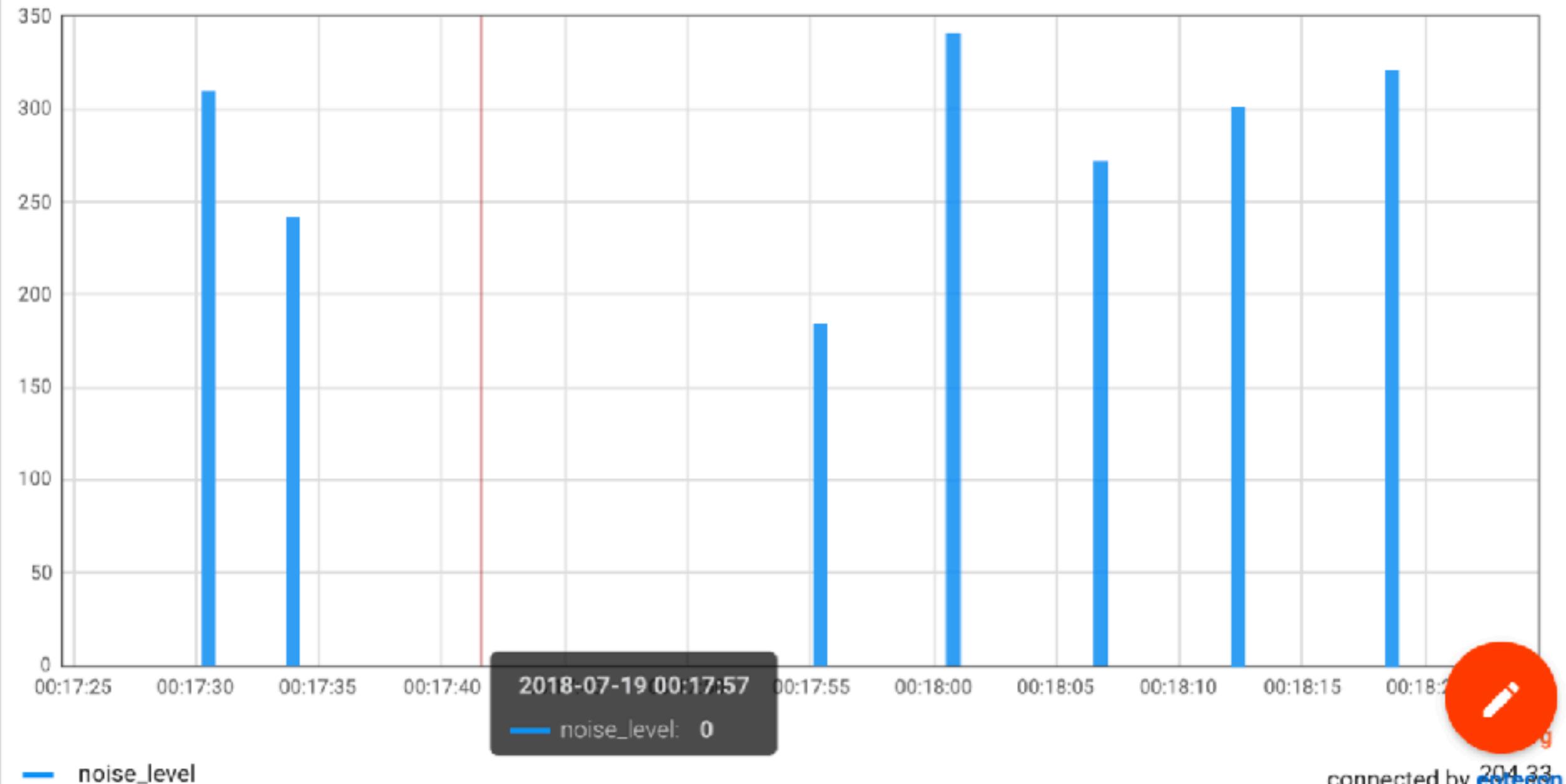
1 telemetry unit selected

SHOW ON WIDGET

<input type="checkbox"/>	Last update time	Key	Value
<input checked="" type="checkbox"/>	2018-07-19 00:18:06	getBatteryValue	95
<input type="checkbox"/>	2018-07-19 00:18:06	getLEDValue	false
<input type="checkbox"/>	2018-07-19 00:18:06	getSoundValue	true
<input type="checkbox"/>	2018-07-19 00:18:06	getThresholdValue	5
<input type="checkbox"/>	2018-07-19 00:18:06	noise_level	272

Page: 1 Rows per page: 5 1 - 5 of 5 < >





CONNECTED SOLUTIONS BY EPTECON.COM

Online Settings

It is easy to set reference noise value, LED and sound settings on a device. You don't need to reprogram device by connecting to your computer. Just send new settings from GlueLogics Cloud to a Noise Detector using Subscribe Button.

INTERNET OF THINGS

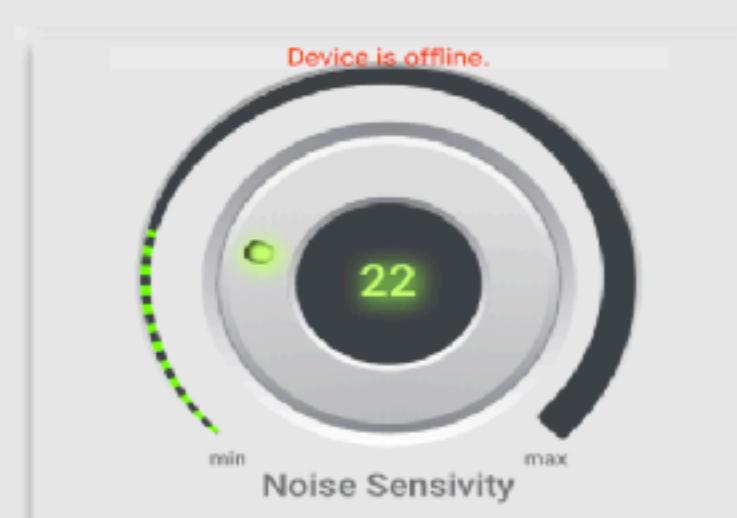
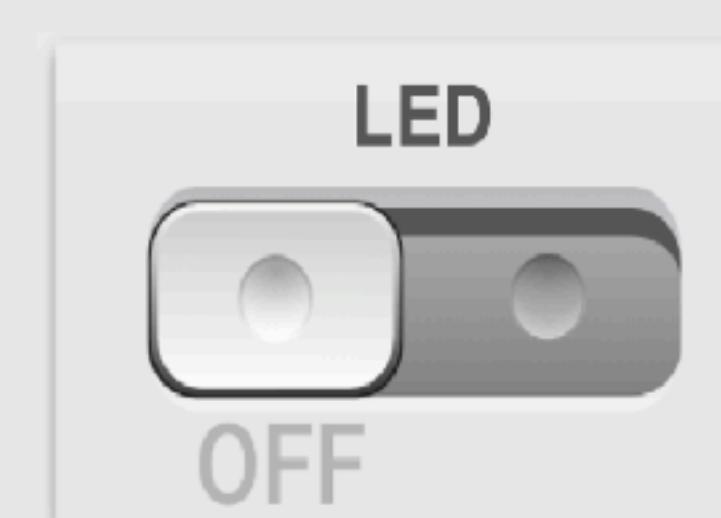
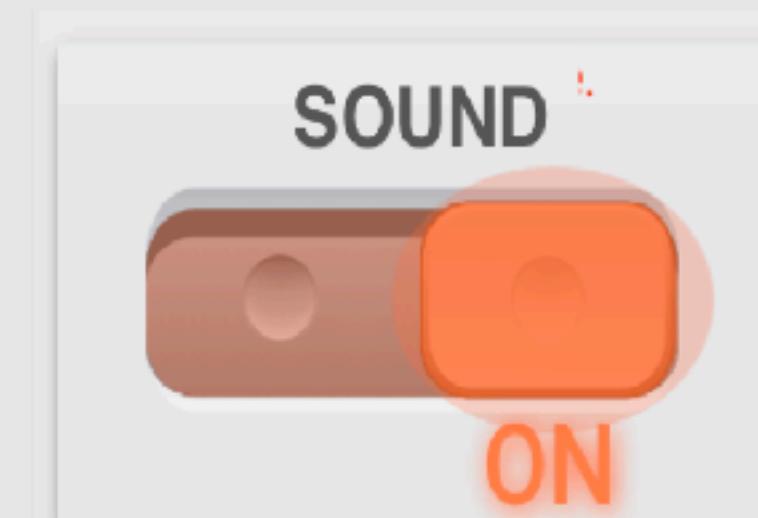
Online Noise Level

Noise level published to cloud with easy to use graphical widgets. The online values could be used for obtaining noise level remotely for direct monitoring or analytical purposes with further environmental improvement.

isBuzzerOn
getSoundValue
true

isLEDOn
getLEDValue
false

Noise Level
getThresholdValue
5

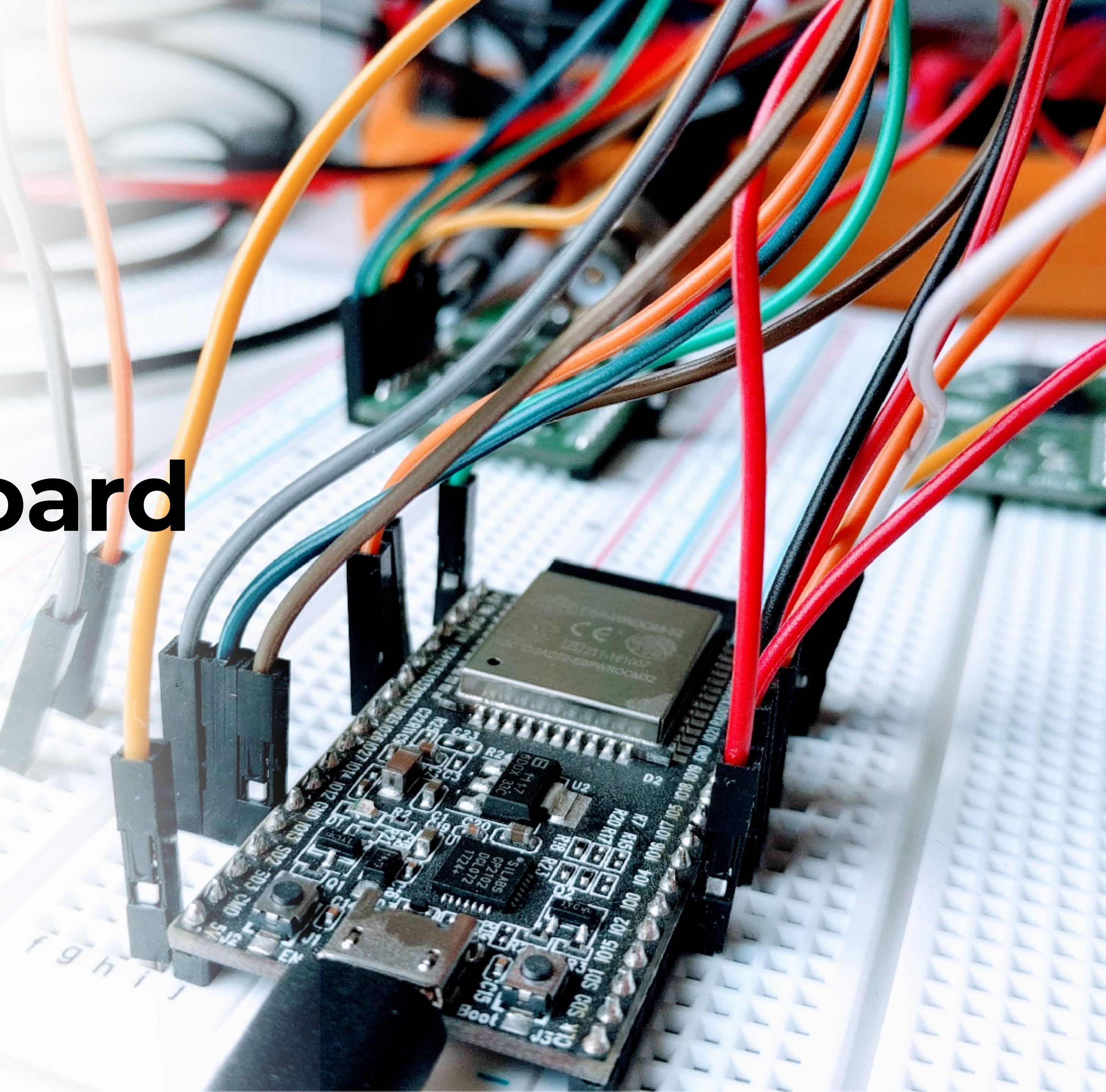


INTERNSHIP PROJECT: OLEKSANDRA BAGA
EPTECON BERLIN, 01.03.2018 - 18.07.2018

Project started on the breadboard

At first Noise Detector was connected and programmed using breadboard that is a great instrument for prototyping and requires absolutely no soldering.

For a prototype ESP32-DevKitC was chosen together with additional boards from Mikroelektronika: an add-on board with noise detecting circuitry and board with magnetic buzzer transducer.

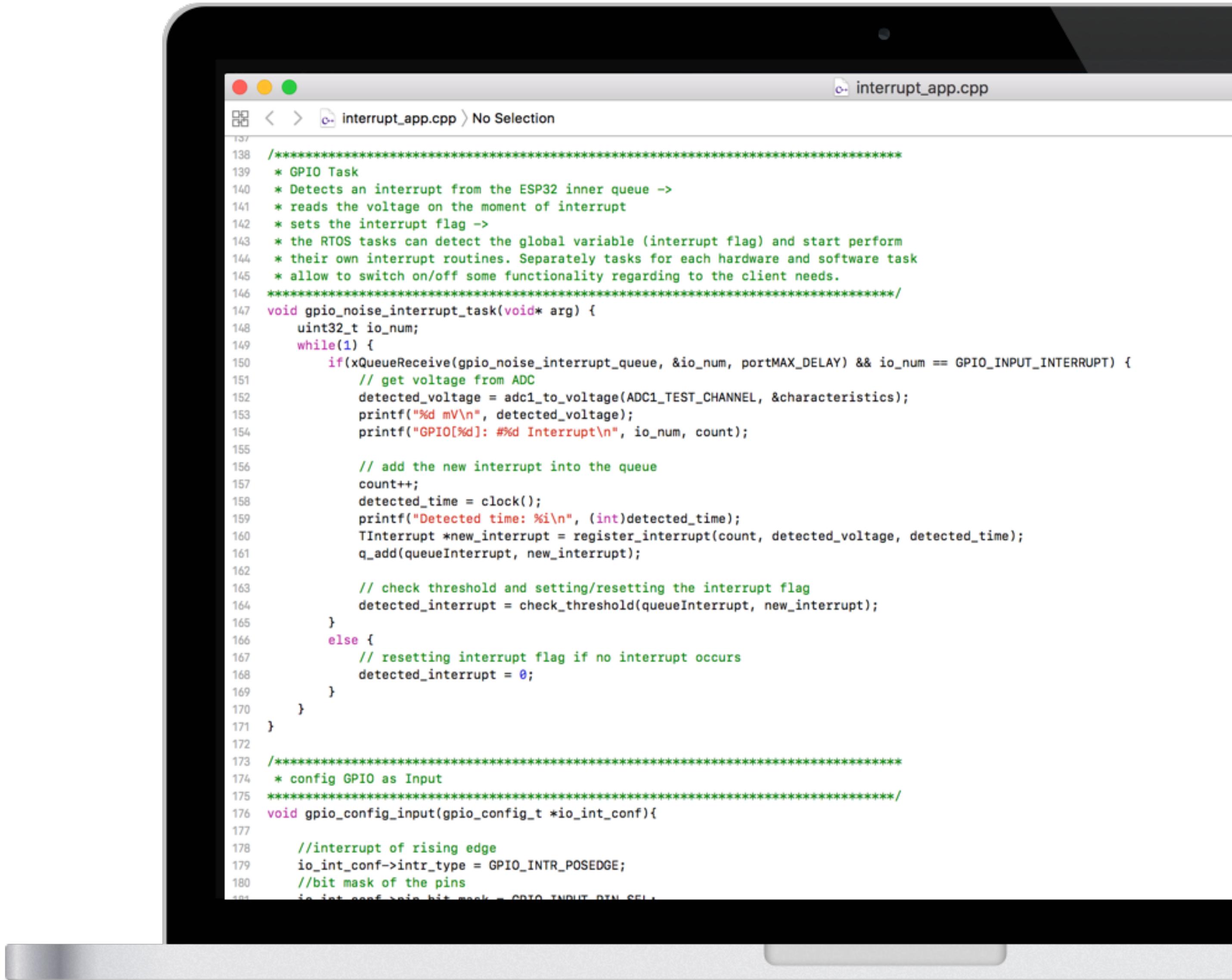


INTERNSHIP PROJECT: OLEKSANDRA BAGA
EPTECON BERLIN, 01.03.2018 - 18.07.2018

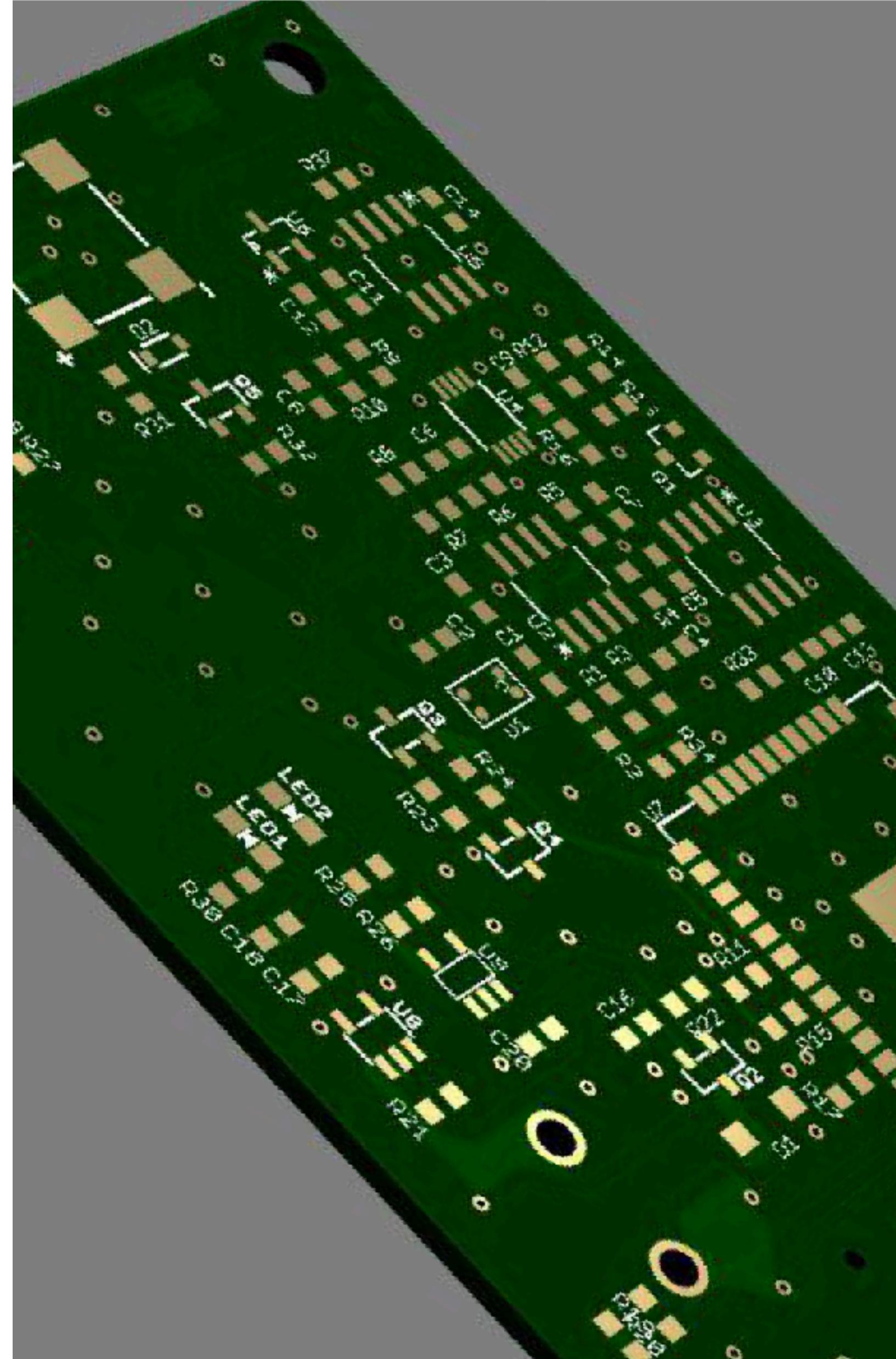
Code written in spring 2018

Initially only C programming language was used. The additional C++ modules were imported when WiFi connectivity and HTTPS Server were added.

Noise Detector doesn't use standard Espressif App for transferring SSID and password for the WiFi connection. User will is not required to install any additional software to start using Noise Detector.



```
interrupt_app.cpp
interrupt_app.cpp > No Selection
137 ****
138 * GPIO Task
139 * Detects an interrupt from the ESP32 inner queue ->
140 * reads the voltage on the moment of interrupt
141 * sets the interrupt flag ->
142 * the RTOS tasks can detect the global variable (interrupt flag) and start perform
143 * their own interrupt routines. Separately tasks for each hardware and software task
144 * allow to switch on/off some functionality regarding to the client needs.
145 ****
146 void gpio_noise_interrupt_task(void* arg) {
147     uint32_t io_num;
148     while(1) {
149         if(xQueueReceive(gpio_noise_interrupt_queue, &io_num, portMAX_DELAY) && io_num == GPIO_INPUT_INTERRUPT) {
150             // get voltage from ADC
151             detected_voltage = adc1_to_voltage(ADC1_TEST_CHANNEL, &characteristics);
152             printf("%d mV\n", detected_voltage);
153             printf("GPIO[%d]: #%d Interrupt\n", io_num, count);
154
155             // add the new interrupt into the queue
156             count++;
157             detected_time = clock();
158             printf("Detected time: %i\n", (int)detected_time);
159             TInterrupt *new_interrupt = register_interrupt(count, detected_voltage, detected_time);
160             q_add(queueInterrupt, new_interrupt);
161
162             // check threshold and setting/resetting the interrupt flag
163             detected_interrupt = check_threshold(queueInterrupt, new_interrupt);
164         }
165         else {
166             // resetting interrupt flag if no interrupt occurs
167             detected_interrupt = 0;
168         }
169     }
170 }
171 ****
172 * config GPIO as Input
173 ****
174 void gpio_config_input(gpio_config_t *io_int_conf){
175
176     //interrupt of rising edge
177     io_int_conf->intr_type = GPIO_INTR_POSEDGE;
178     //bit mask of the pins
179     io_int_conf->pin_bit_mask = GPIO_TINPUT_BTN_SEL;
180
181 }
```



INTERNSHIP PROJECT: OLEKSANDRA BAGA
EPTECON BERLIN, 01.03.2018 - 18.07.2018

Noise Detector PCB Design

PCB DESIGN IN EAGLE IS A TWO-STEP PROCESS



CIRCUIT DESIGN

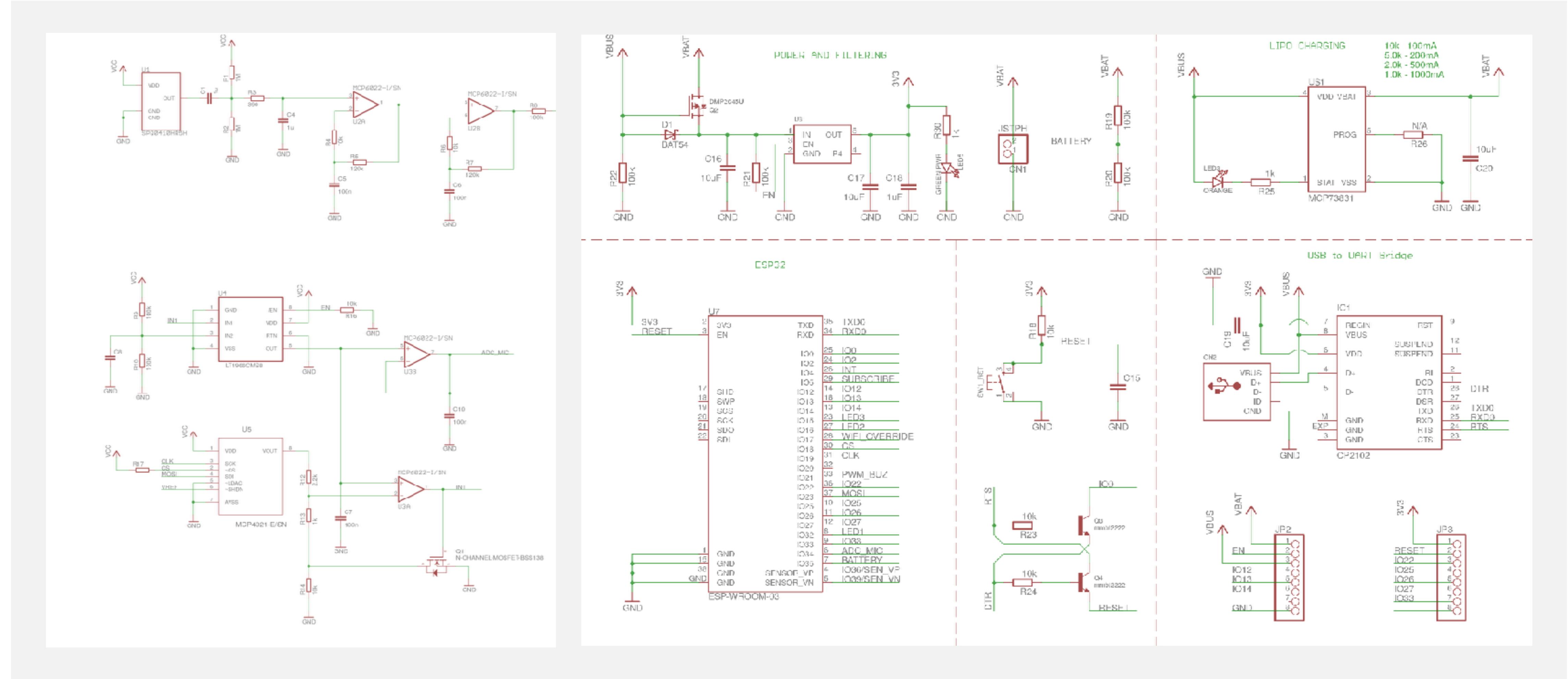
Circuit design is about electronically connecting the components. Added parts to the schematic sheet need to be wired together.



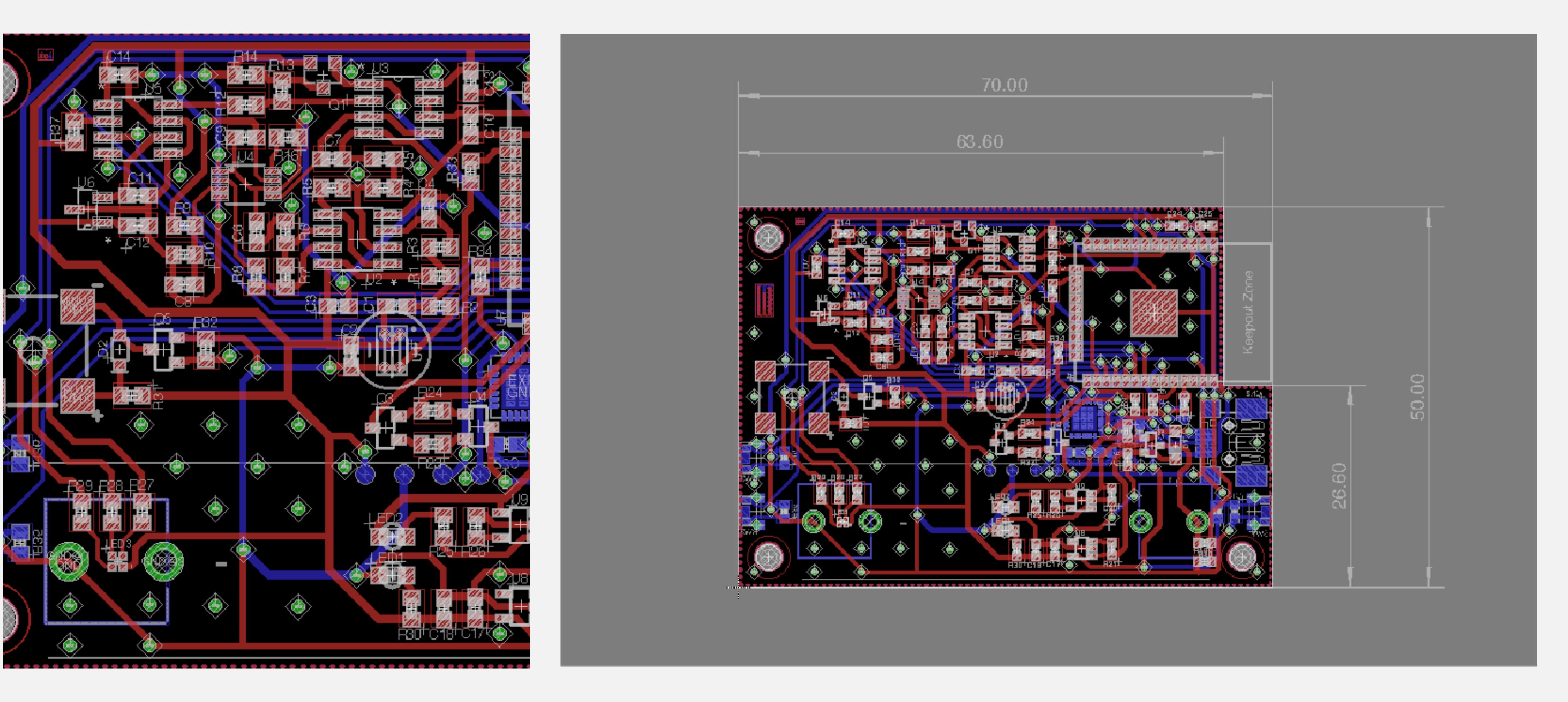
BOARD LAYOUT

PCB composition is all about copper connections between components. A thin layer of copper is where our electric signals pass through.

Noise Detector Circuit Design



Noise Detector Circuit Design

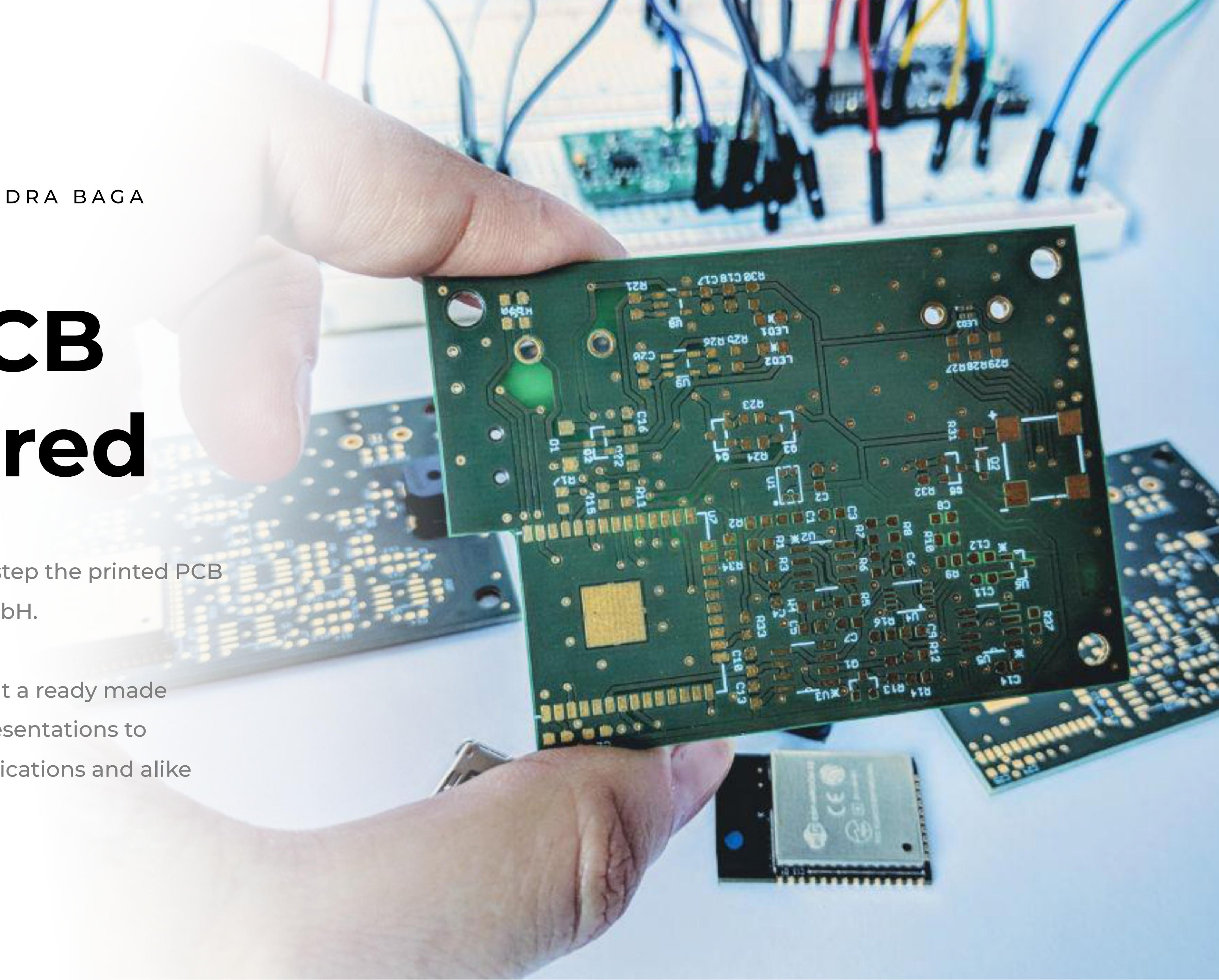


INTERNSHIP PROJECT: OLEKSANDRA BAGA
EPTECON BERLIN, 01.03.2018 - 18.07.2018

Printed PCB was soldered

To avoid extra costs on the prototyping step the printed PCB was soldered and tested by Eptecon GmbH.

Noise Detector now isn't a mere idea, but a ready made device prototype that can be used in presentations to potential customers, crowdfunding applications and alike





**DIPL.-ING.
ERWIN PRIZKAU**

EPTECON FOUNDER
ENGINEERING, PRODUCT
DEVELOPMENT AND TECHNOLOGY
CONSULTING

Internship adviser



**PROF. DR.
CHRISTIAN FORLER**

PROFESSOR AT BEUTH
UNIVERSITY OF APPLIED
SCIENCES

Internship supervisor



OLEKSANDRA BAGA

COMPUTER ENGINEERING
(B.ENG) STUDENT
AT BEUTH UNIVERSITY OF
APPLIED SCIENCES

Intern

FIND MORE ON THE WEBSITE:

WWW. EPTECON.COM