



Akademia Górniczo-Hutnicza  
im. Stanisława Staszica w Krakowie.  
Wydział Informatyki, Elektroniki i Telekomunikacji

Sensory w aplikacjach wbudowanych

**SPRAWOZDANIE**

Rok I, Systemy wbudowane, Elektronika i Telekomunikacja IIst.

Temat: Moduł z sensorami oparty na procesorze STM32F103, zgodny z wyprowadzeniami Mikrobus™

Zespół:

1. Mateusz Kozyra
2. Mirosław Wiącek
3. Radosław Sajdak

Oceny indywidualne:

- 1.
- 2.
- 3.

Data oddania sprawozdania:

Ocena sprawozdania:

Uwagi prowadzącego zajęcia:

Informacje dodatkowe:

## Spis treści

<b>1. Wstęp</b>	3
1.1. Mikrobus™	3
1.2. STM32F103	3
1.3. STS30	3
1.4. BMP280	3
1.5. MQ2	3
1.6. Makefile	3
1.7. Github	3
<b>2. Wykonanie projektu</b>	5
2.1. Projekt płytki	5
2.1.1. Schemat	5
2.1.2. Layout	5
2.1.3. Popelnione błędy oraz wykonane przeróbki	5
2.2. Oprogramowanie	6
2.2.1. Biblioteki peryferiów	6
2.2.2. API	6
2.2.3. Niewykorzystane możliwości	6
2.2.4. System kontroli wersji	6
<b>3. Prezentacja działania</b>	7
<b>4. Wnioski</b>	9

# **1. Wstęp**

Krótki opis co w ogóle zostało zrobione, po co itp.

## **1.1. Mikrob<sup>TM</sup>**

Co to microbus i po co to jest

## **1.2. STM32F103**

Krótki opis wybranego procka

## **1.3. STS30**

Krótki opis sensora Mirka

## **1.4. BMP280**

Krótki opis sensora Matiego

## **1.5. MQ2**

Krótki opis sensora Matiego

## **1.6. Makefile**

Napisać, że pisaliśmy na dwóch różnych płytkach, więc mieliśmy do tego Makefile. Krótko jak działa

## **1.7. Github**

Założenia korzystania z systemu kontroli wersji. Krótko o tym co to jest



## **2. Wykonanie projektu**

### **2.1. Projekt płytki**

#### **2.1.1. Schemat**

Tutaj wkleję schemat, opiszę krótko jak zdecydowałem się go podzielić. Warto wspomnieć o tym, że korzystałem przy projektowaniu z dokumentacji producentów, a i tak się pomyliłem. Trzeba też koniecznie powiedzieć, że komponenty doбираłem samodzielnie na podstawie dostępności na Mouserze i dokumentacji. Koniecznie napisać, o konwerterze do debugu kodu.

#### **2.1.2. Layout**

Pokazać warstwy elektryczne. Zdjęcia wydrukowanych płytek. Jakie problemy się pojawiły (1 raz z Kicadem, więc np. okazało się, że domyślnie ścieżki ma całkiem szerokie. Podobnie viasy). Brak miejsca na wyprowadzenia SWD

#### **2.1.3. Popelnione błędy oraz wykonane przeróbki**

- Pokazać odcięcie LDO konwertera
- Bramka wisząca w powietrzu
- Zła przetwornica 5V ( $V_{in} > V_{out}$ )
- Żle wsadzony mosfet. Pokazać możliwe przeróbki (OPAMP, rezystory, zdjęcia z prób i cięć).
- Bypass przetwornicy
- Ręczne lutowanie SWD do procka

## **2.2. Oprogramowanie**

### **2.2.1. Biblioteki peryferiów**

Krótko opisać jakie były założenia przy tworzeniu peryferiów (rozbicie drzewka kodu dla porządku, ustandaryzowane kody błędów, osobne libki na wszystko itp.)

### **2.2.2. API**

Założenia API. Jak działa, wrzucić grafy, listę komend(?), mechanizmy rejestracji komend jako coś, co pozwala łatwo rozbudować soft do innej aplikacji.

### **2.2.3. Niewykorzystane możliwości**

Nie wiem, tutaj mógłbym może napisać że zrobiłem kozak timery, ale w sumie to nie wiadomo po co bo API jest bezobsługowe xD No i może o tym, że Mati nie wiadomo po co pisał na F4...

### **2.2.4. System kontroli wersji**

IMHO spoko też napisać, o tym, że korzystaliśmy z tego gita mocno, dbaliśmy o porządek, robiliśmy review itp. To jednak zjadło kupę czasu.

---

### **3. Prezentacja działania**

Wrzucić tutaj fotki z cutecoma i pokazać, że działa. Może jakieś zdjęcia że przed initem lampka nie świeci, a po świeci xD





## 4. Wnioski

No na pewno napisać, że inaczej bym PCBka zrobił. Że dużo było czytania dokumentacji, trzeba było sobie przypomnieć sporo z elektroniki, a i tak pojawiły się błędy. Na pewno, że 0805 są za duże. Że robienie review pomaga obu stronom czegoś się nauczyć. Że sensory na PCB muszą być przemyślane (MQ + temperatury). Że mimo tego, że N osób widziało schematy i layout, to błędy przeszły produkcję i dużo czasu straciło się na debug zasilaczami, oscyloskopami itp.. Ale trzeba też powiedzieć, że robiliśmy projekt 4fun i dużo się nauczyliśmy.