



UMCS

UNIWERSYTET MARII CURIE-SKŁODOWSKIEJ
W LUBLINIE

Wydział Matematyki, Fizyki i Informatyki

Kierunek: **informatyka**

Jan Bylina

nr albumu: 303827

Projekt oraz implementacja systemu gromadzenia rozproszonych danych z wykorzystaniem technologii LoRa

Design and implementation of the distributed data collection system
using LoRa technology

Praca licencjacka

napisana w Katedrze Oprogramowania Systemów Informatycznych

Instytutu Informatyki UMCS

pod kierunkiem **dr hab. Przemysław Stpiczyńskiego**

Lublin 2023

Spis treści

Wstęp	5
1 Rozdział — tutorial	7
1.1 Sekcja A	7
1.2 Sekcja B	8
2 Wykorzystane narzędzia, technologie i protokoły	9
2.1 Urządzenia	9
2.1.1 ESP32	9
2.1.2 Raspberry Pi Pico	10
2.1.3 STM32	10
2.2 Języki programowania i technologie	10
2.2.1 C++ for Arduino	10
2.2.2 C for STM32	10
2.2.3 MicroPython for Raspberry Pi Pico	10
2.2.4 Python for MQTT	10
2.3 Protokoły komunikacyjne	10
2.3.1 MQTT	11
2.3.2 LoRa	11
2.3.3 HTTP	11
2.4 Bazy danych i pozostałe technologie	11
2.4.1 InfluxDB 2	11
2.4.2 Docker	11
2.4.3 PlatformIO	11
3 Istniejące rozwiązania	13
3.1 LoRaWAN	13
3.1.1 The Things Network ?	13
3.1.2 ChirpStack ?	13
3.1.3 Loriot ?	13

3.2	Artykuły	13
3.3	Wpisy w sieci i blogach	13
4	Założenie i Implementacja	15
5	Wdrożenie i testy	17
6	Wnioski i perspektywy rozwoju	19
	Spis listingów	21
	Spis tabel	23
	Spis rysunków	25
	Bibliografia	25

Wstęp

Tu treść wstępu WSTĘP WSTEP —

Rozdział 1

Rozdział — tutorial

1.1 Sekcja A

W tabeli 1.1 widzimy przykład tabeli z nagłówkiem i odnośnikiem. Tabele tworzymy z nagłówkiem na górze oraz opcją `[t]`. Natomiast na rysunku 1.1 — widzimy przykład rysunku z nagłówkiem i odnośnikiem. Rysunki tworzymy z nagłówkiem pod spodem oraz opcją `[b]`. Rysunki powinny być w formacie PDF; jeśli to niemożliwe, to PNG (w wysokiej rozdzielczości); a ostatecznie JPG (jak tu). Jeśli chcemy sterować rozmiarem, to zwykle najwygodniej użyć `width=...`. Ponadto możemy odwoływać się do bibliografii

Jeśli chodzi o wzory, możemy złożyć je na kilka sposobów, w zależności od potrzeb — w tekście: $e = \lim_{n \rightarrow \infty} \left(1 + \frac{1}{n}\right)^n$, wyniesiony do osnej linii (warto zwrócić uwagę, że ten i kolejny są złożone nieco inaczej niż pierwszy):

$$e = \lim_{n \rightarrow \infty} \left(1 + \frac{1}{n}\right)^n,$$

a także wyniesiony z numerem:

$$e = \lim_{n \rightarrow \infty} \left(1 + \frac{1}{n}\right)^n. \tag{1.1}$$

Do tego oostatniego możemy się odwołać: (1.1). No i oczywiście listingi — listing 1.1 pokazuje, jak zrobić to w miarę poprawnie...

sekcja 232323



Rysunek 1.1: Przykładowy rysunek

Tabela 1.1: Przykładowa tabela

slkdjflsj	sdkskd	s;lkdsdk
slkjd	skljdsldj	skljdsjdsldj
sljkdslkjd	woieupowiepoweiwiewp	weoiw eppowie wpo

1.2 Sekcja B

```
1 tab[0:n] = dem[nRows][nCols]; //?  
2 #pragma acc data copy(tab [0:n], slope [0:n])
```

Listing 1.1: Jakież dwie linijki w C++ (z OpenACC)

Rozdział 2

Wykorzystane narzędzia, technologie i protokoły

2.1 Urządzenia

2.1.1 ESP32

ESP32 to jednoukładowy mikrokontroler, zaprojektowany i produkowany przez firmę Espressif Systems. Jego najważniejsze cechy to:

- energooszczędny procesor RISC o częstotliwości do 240 MHz
- 520 kB pamięci SRAM
- WiFi 802.11 b/g/n
- Bluetooth
- liczne interfejsy cyfrowe i analogowe, w tym:
 - 2x UART
 - 2x I2C
 - 2x SPI
 - 2x I2S
 - 2x CAN
 - 2x ADC
 - 2x DAC
 - 2x PWM
 - 2x LED PWM

- 2x Hall sensor
- 2x SDIO
- 2x Ethernet MAC
- 2x USB 2.0

[1]

• ...

Powstało wiele wersji tego układu, różniące się m.in. szybkością procesora, ilością pamięci flash, ilością pinów, ilością interfejsów cyfrowych i analogowych, a także możliwością pracy w trybie bezprzewodowym (WiFi) lub przewodowym (Ethernet)[2].

2.1.2 Raspberry Pi Pico

—

2.1.3 STM32

—

2.2 Języki programowania i technologie

2.2.1 C++ for Arduino

—

2.2.2 C for STM32

—

2.2.3 MicroPython for Raspberry Pi Pico

—

2.2.4 Python for MQTT

2.3 Protokoły komunikacyjne

—

2.3.1 MQTT

—

2.3.2 LoRa

—

2.3.3 HTTP

—

2.4 Bazy danych i pozostałe technologie

2.4.1 InfluxDB 2

—

2.4.2 Docker

—

2.4.3 PlatformIO

—

Rozdział 3

Istniejące rozwiązania

—

3.1 LoRaWAN

—

3.1.1 The Things Network ?

—

3.1.2 ChirpStack ?

—

3.1.3 Lorient ?

—

3.2 Artykuły

—

3.3 Wpisy w sieci i blogach

Rozdział 4

Założenie i Implementacja

Rozdział 5

Wdrożenie i testy

Rozdział 6

Wnioski i perspektywy rozwoju

Spis listingów

1.1	Jakieś dwie linijki w C++ (z OpenACC)	8
-----	---	---

Spis tabel

1.1	Przykładowa tabela	8
-----	------------------------------	---

Spis rysunków

1.1	Przykładowy rysunek	7
-----	-------------------------------	---

Bibliography

- [1] Espressif Systems. *ESP32 Datasheet*. 2023. URL: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf (visited on 04/16/2023).
- [2] Espressif Systems. *ESP32 SoCs*. 2023. URL: <https://www.espressif.com/en/products/socs> (visited on 04/16/2023).