

Дослідження мікроконтролерів та міні комп'ютерів у IoT пристроях.

Виконав:

Чернацький А.Ю.

Науковий керівник:

Харченко К.В.

- **Мета**

Дослідити можливості використання мікроконтролерів та міні комп'ютерів у IoT пристроях

- **Об'єкт дослідження**

Мікроконтролери та міні комп'ютери у IoT пристроях

- **Предмет дослідження**

Методи реалізації та протоколи взаємодій мікроконтролерів та міні комп'ютерів як IoT пристроїв

Задачі

1. Провести систематизацію, порівняння поширених мікроконтролерів та міні ПК, а також виконати їх навантажувальне тестування
2. Провести аналіз протоколів взаємодії IoT пристроїв
3. Дослідити можливість паралельних обчислень на IoT пристроях
4. Розробити зразок IoT системи
5. Надати практичні рекомендації щодо реалізації систем IoT

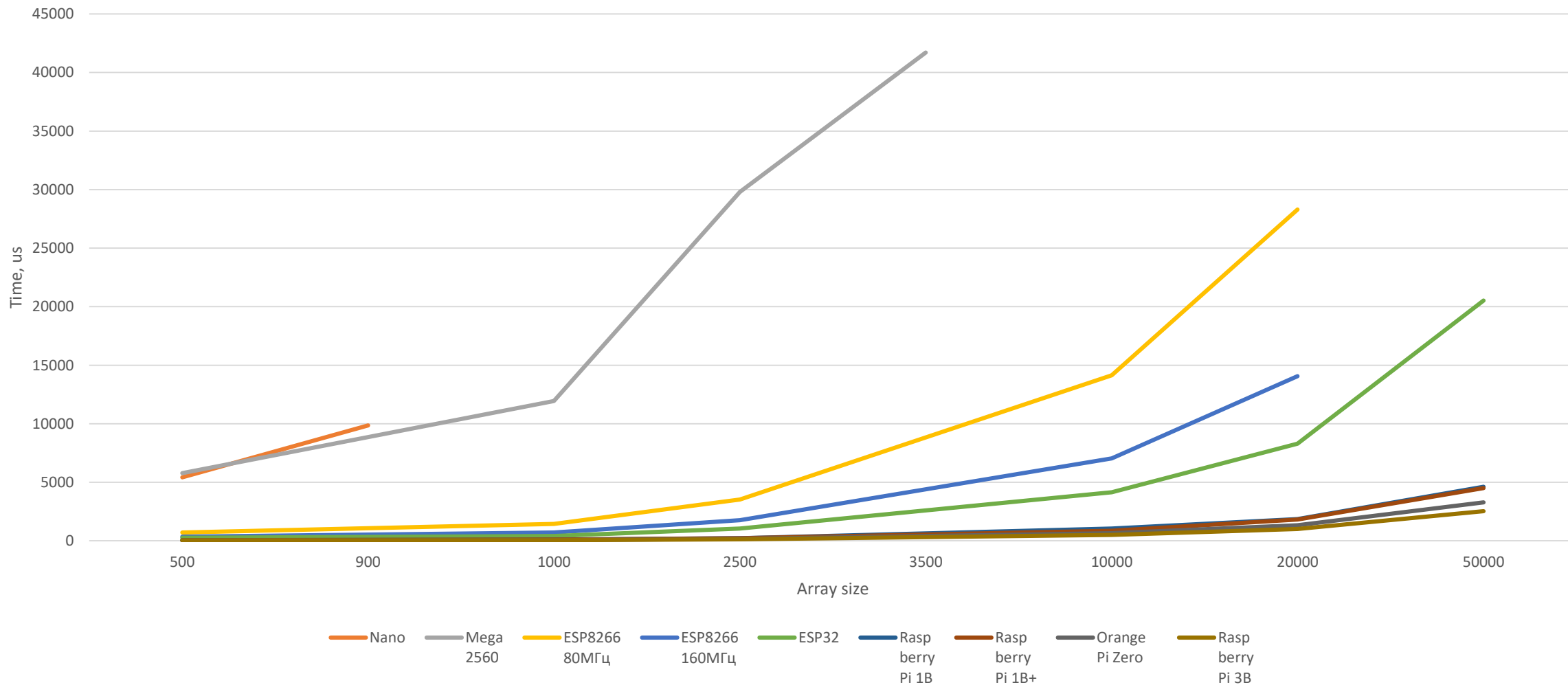
Характеристики мікроконтролерів та міні комп'ютерів

Модель	Чіп	Частота	RAM	ROM	Підключення до локальної мережі	Цифрові Входи / Виходи	Ціна
Arduino Uno	ATmega328	16 МГц	2 КБ	32 КБ	-	14	\$4
Arduino Mega	ATmega2560	16 МГц	8 КБ	256 КБ	-	54	\$8
ESP8266	Tensilica Xtensa L106	80/160 МГц	96 КБ	0.5-16 МБ	Wi-Fi	11	\$3
ESP32	Tensilica LX6	2x 240 МГц	520 КБ	16 МБ	Wi-Fi	32	\$8
Raspberry pi B+	BCM2835	700 МГц	512 МБ	-/microSD	RJ-45	40	\$38
Orange Pi Zero	Alwinner H2+	4x 1.2 ГГц	512 МБ	-/microSD	RJ-45/Wi-Fi	26	\$14
Raspberry pi 3B	BCM2837	4x 1.2 ГГц	1 ГБ	-/microSD	RJ-45/Wi-Fi	40	\$42

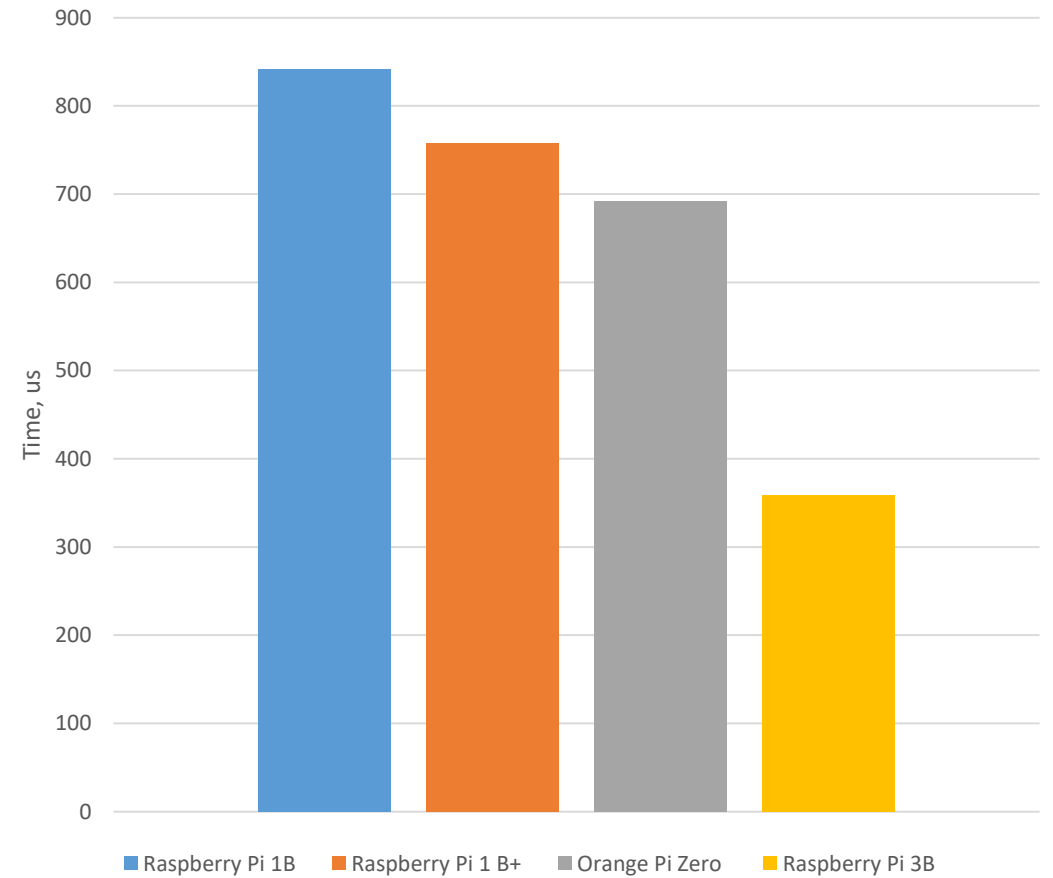
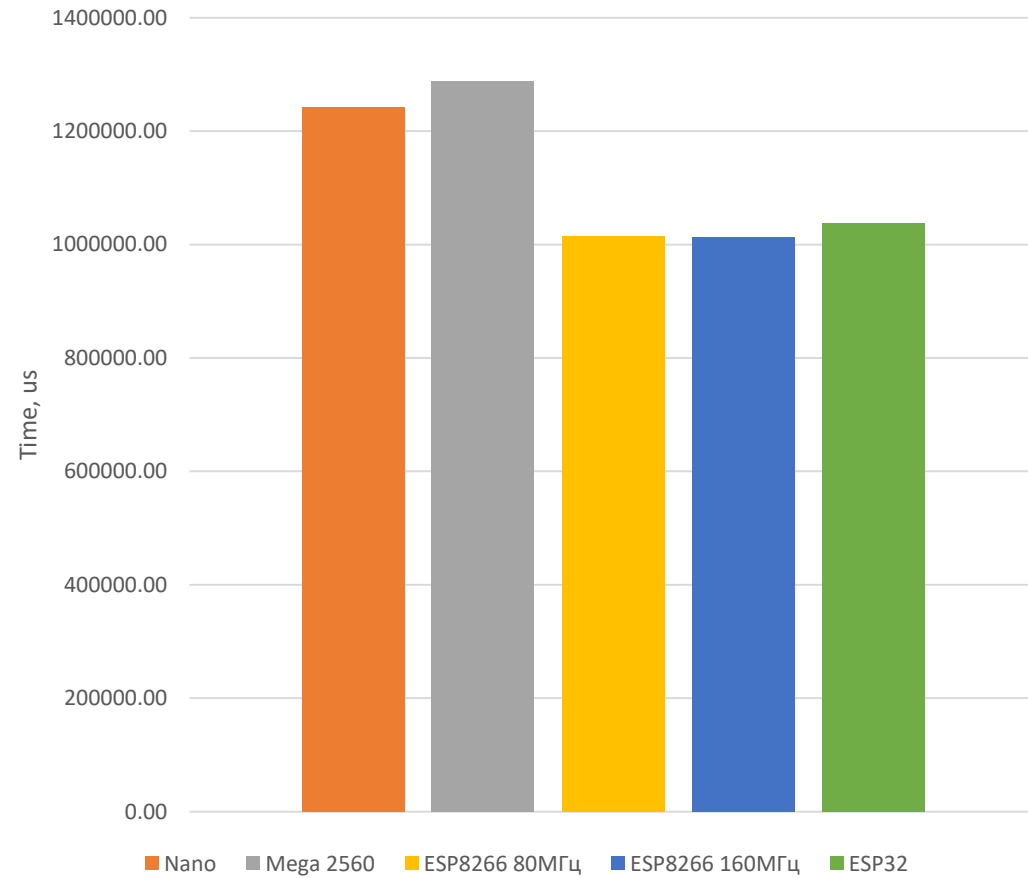
Навантажувальне тестування

- Види тестування
 1. Складання елементів масиву
 2. Виконання Get запитів
- Мови програмування
 1. C++
 2. Java
 3. Python

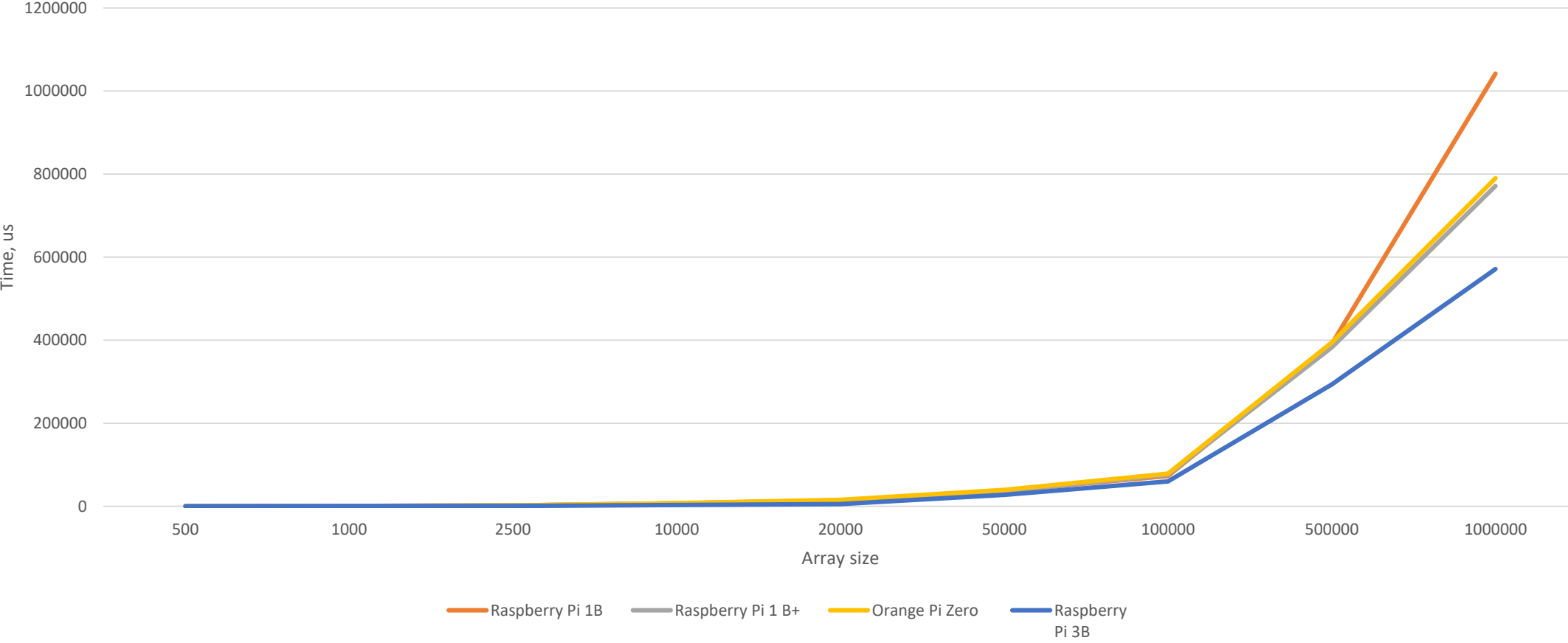
Складання елементів масиву на мові C++



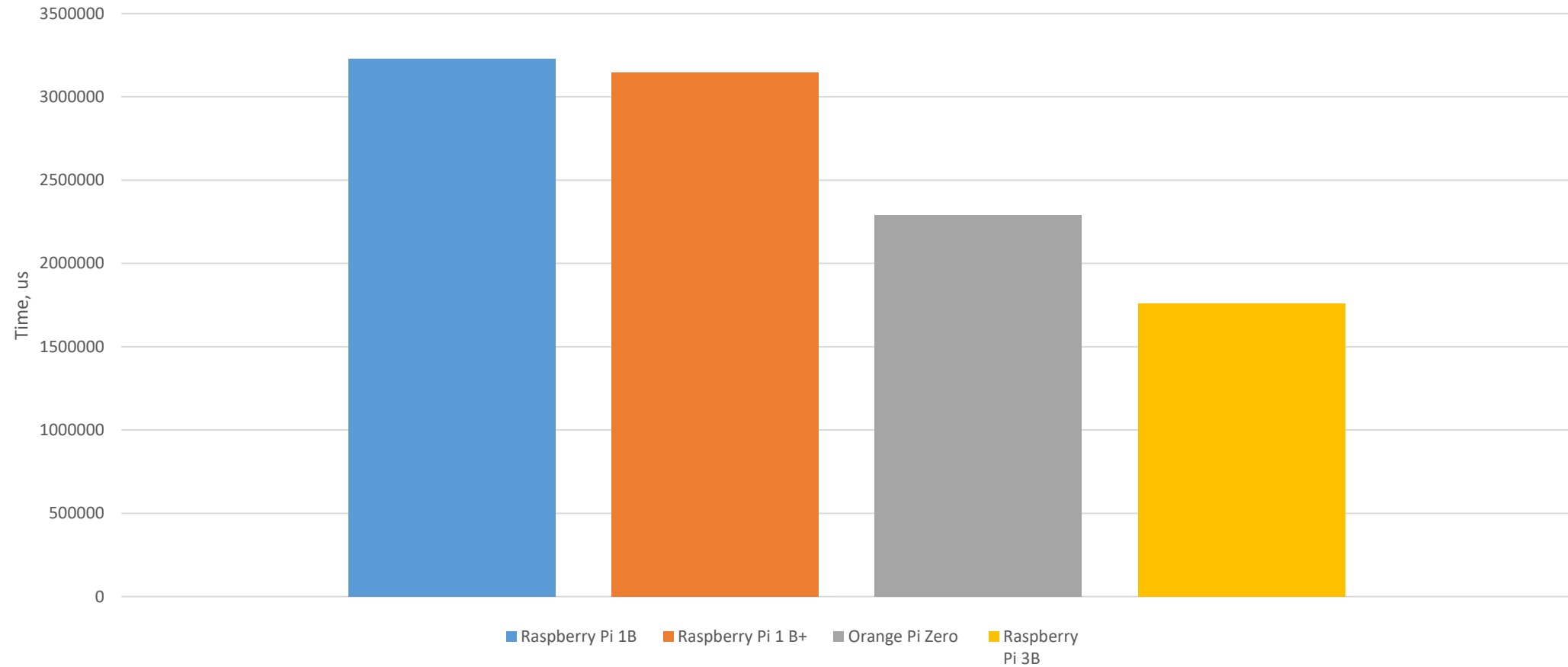
Виконання Get запитів на мові C++



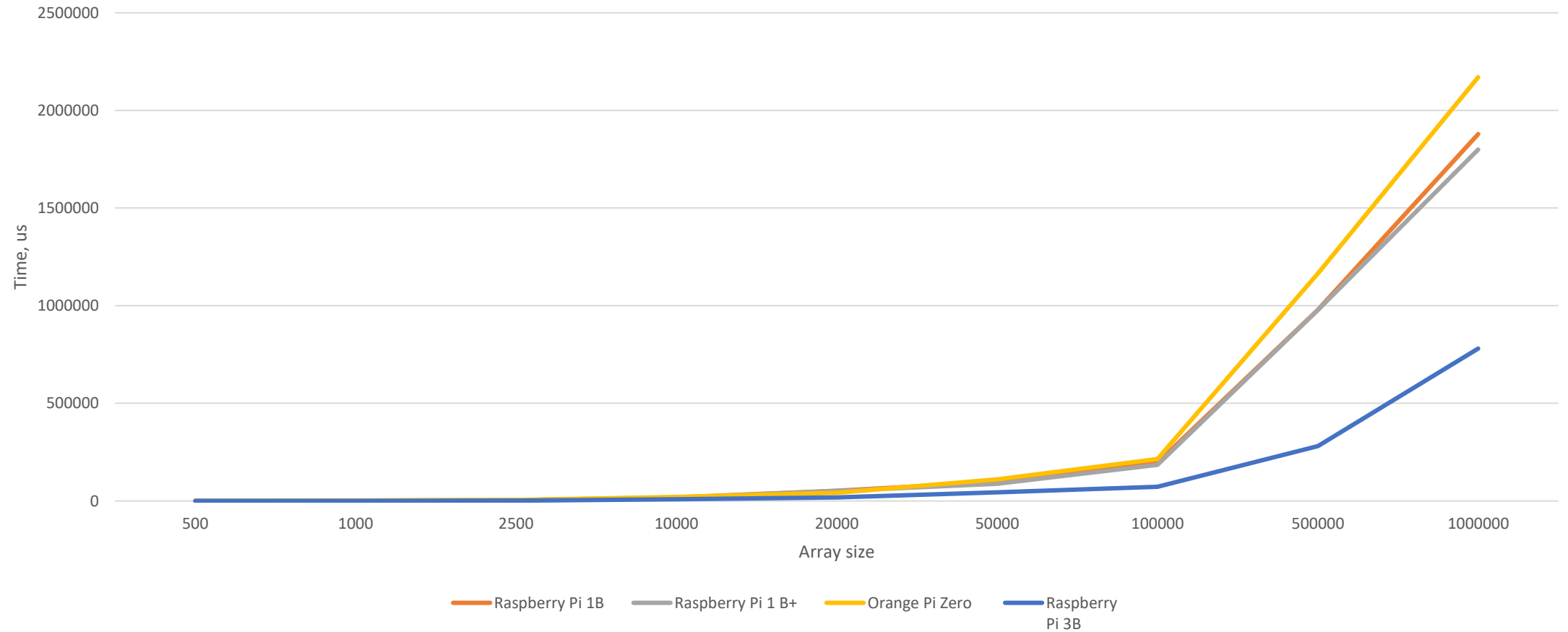
Складання елементів масиву на мові Java



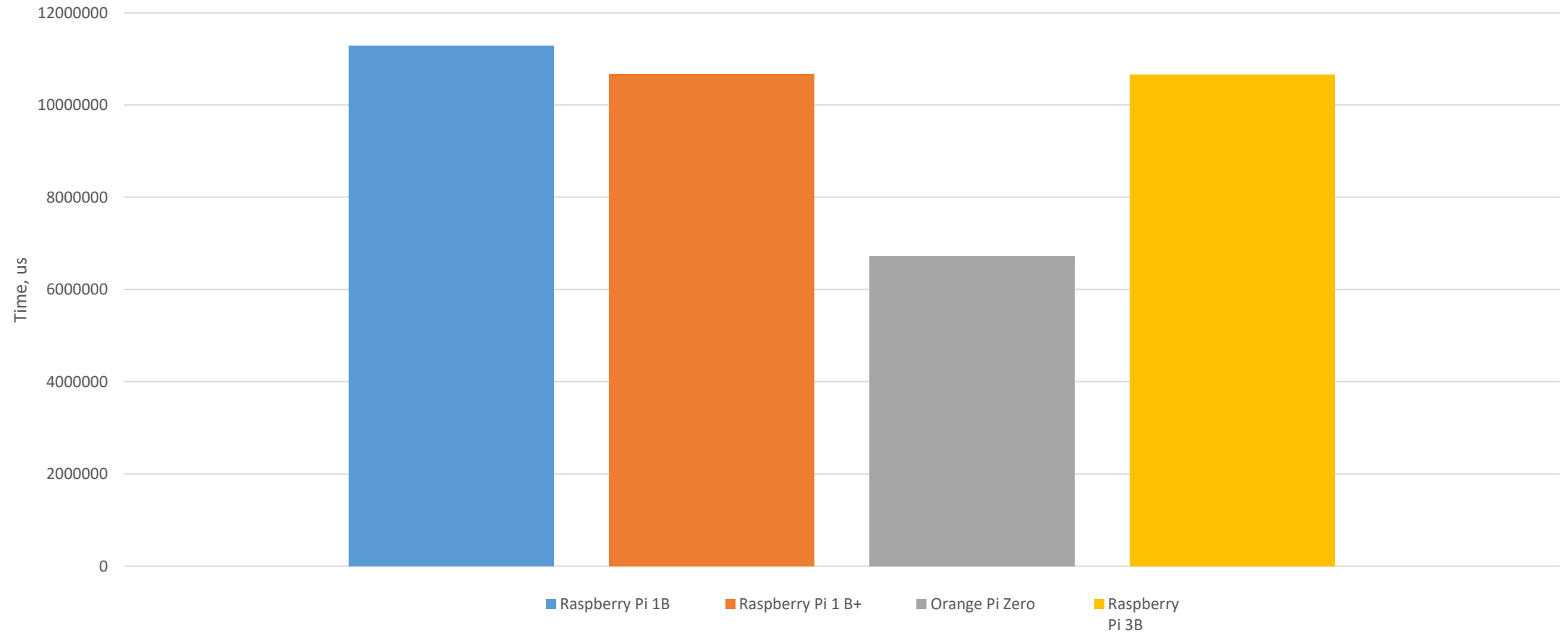
Виконання Get запитів на мові Java



Складання елементів масиву на мові Python



Виконання Get запитів на мові Python



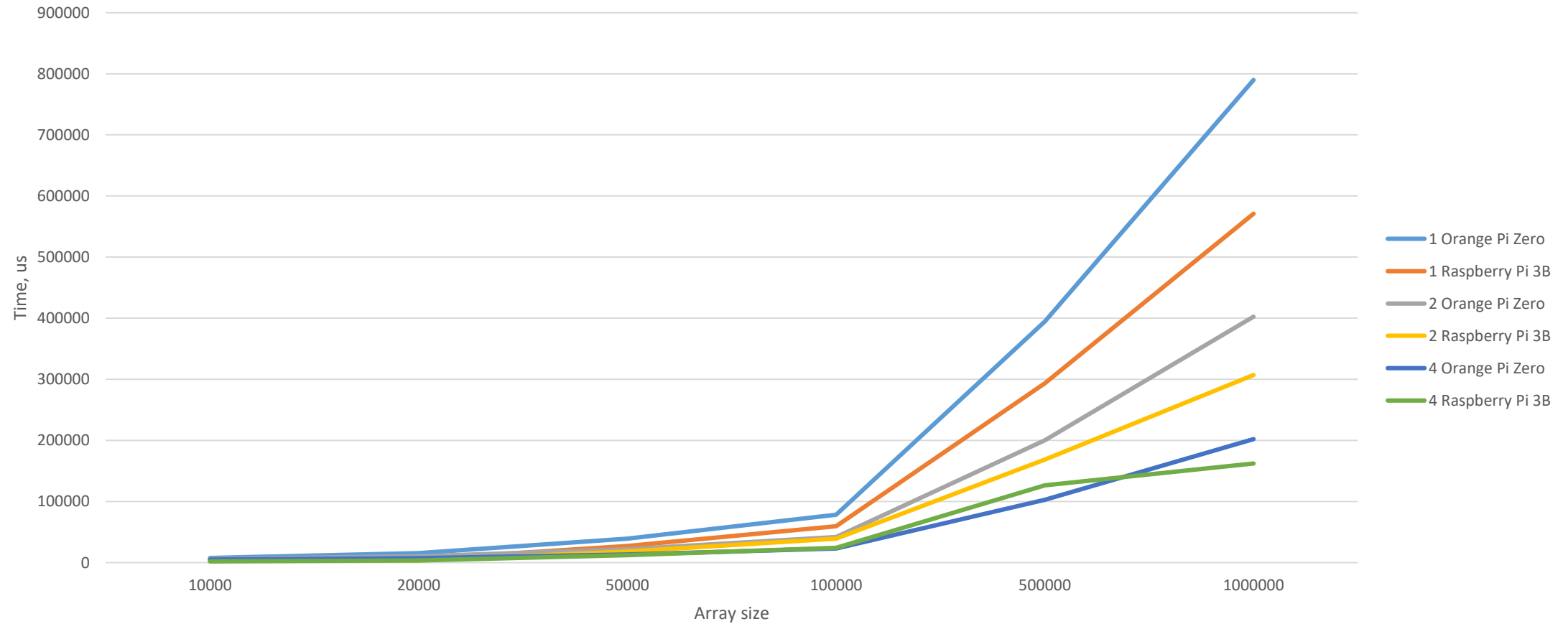
Засоби взаємодії

Технології	Транспортний протокол	Модель	Підтримка аутентифікації	Обмеження представлення контенту
Socket	udp, tcp	клієнт-сервер	-	-
REST	tcp	клієнт-сервер	+	+
MQTT	tcp	видавець-передплатник	+	-
SNMP	udp	менеджер-агент	+	+

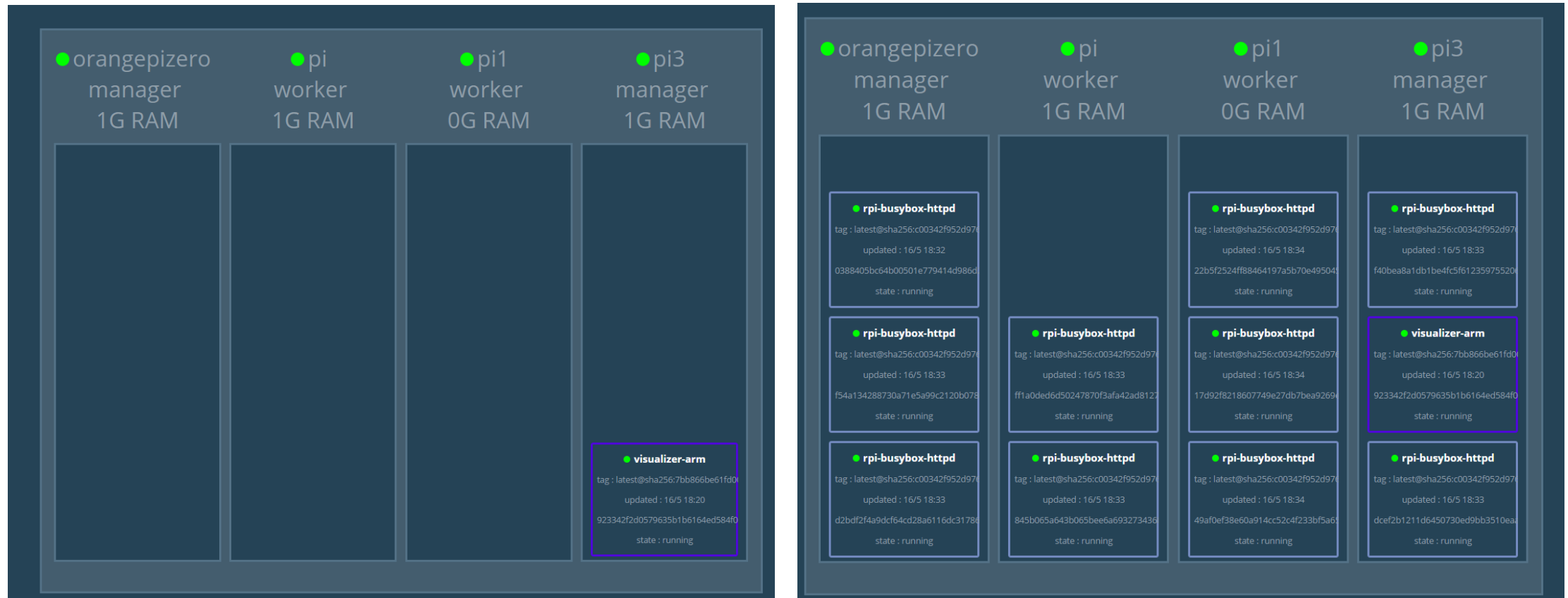
Паралельні обчислення

- Використання потоків
- Побудова кластера Docker Swarm

Багато поточність Java

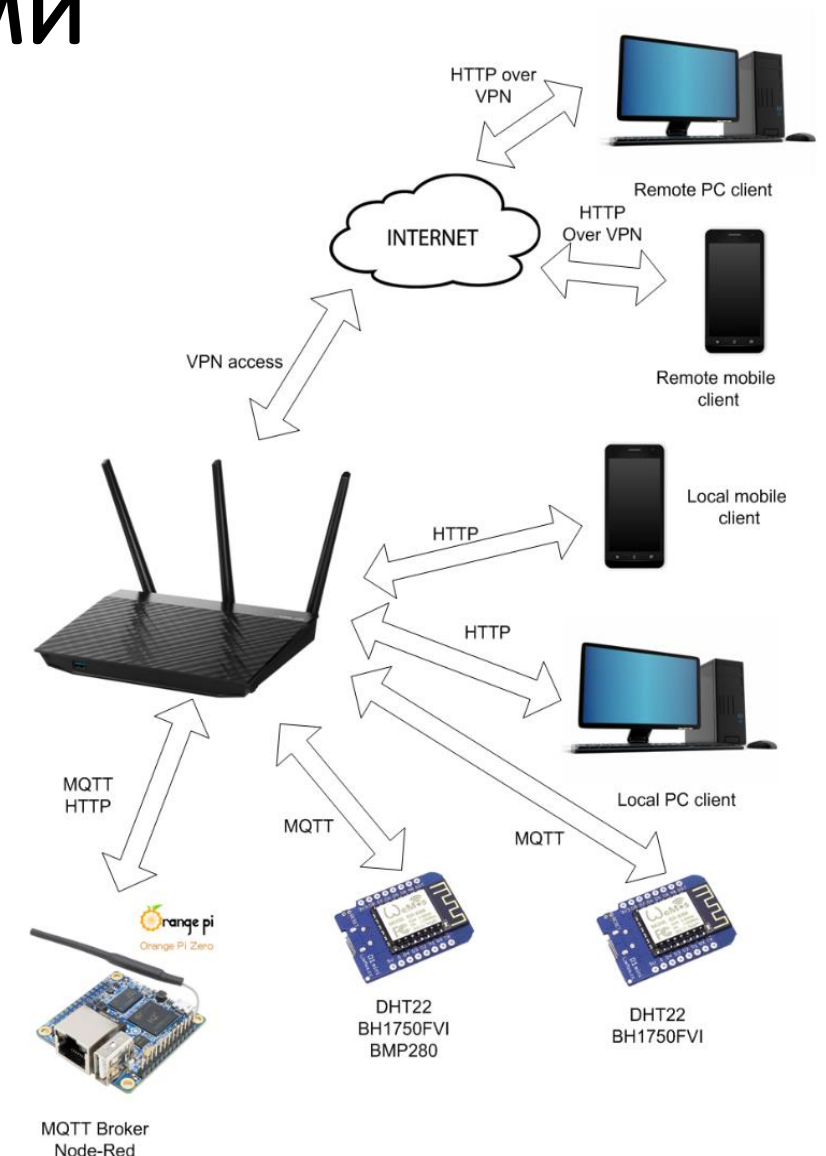


Кластер Docker swarm

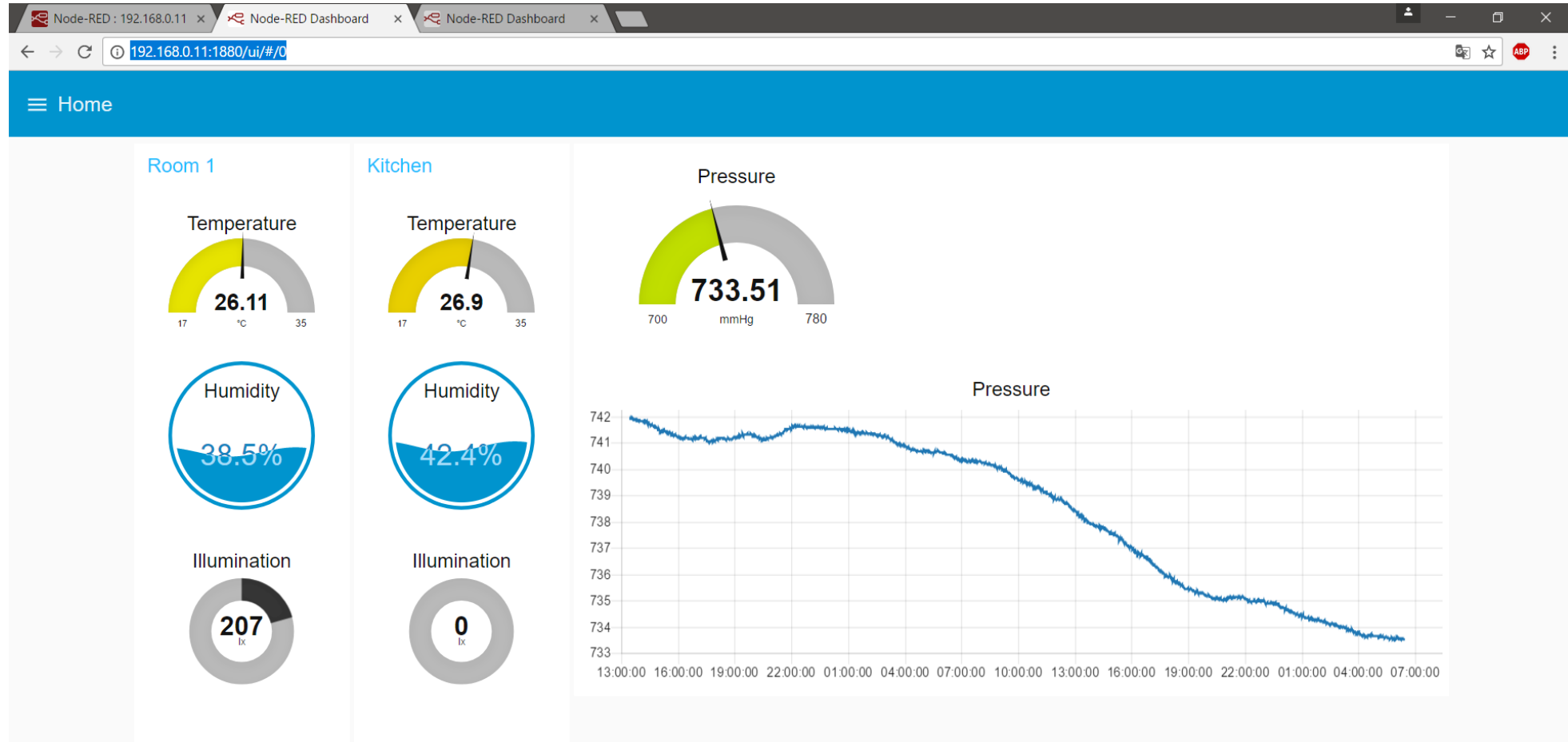


Зразок IoT системи

- Основа для датчиків: ESP8266
- Хаб пристрій: Orange Pi Zero
- Протокол взаємодії: MQTT
- Візуалізація: Node-Red
- Спосіб розгортання: Docker Compose



Node-Red



Розробка стартап-проекту

Згідно до проведених досліджень:

- існує можливість ринкової комерціалізації проекту (попит присутній, ринок зростає)
- існують перспективи впровадження з огляду на потенційні групи клієнтів, бар'єри входження не є високими, проект має дві значні переваги перед конкурентами (інтеграція зі сторонніми сервісами та простоту користування)
- необхідно реалізувати веб-додаток, мобільний додаток хаб з MQTT та датчики
- подальша імплементація є доцільною
- межа встановленої ціни на товар 5000 грн за базовий комплект

Висновки

- Оптимальним мікроконтролером для датчиків є ESP8266
- Оптимальним міні ПК для використання в якості хаба або міні сервера є рішення від Orange Pi, менша ціна та більш актуальні версії ОС
- Для виконання паралельних обчислень можлива побудова кластера Docker Swarm на міні ПК або використання декількох потоків
- Найбільш оптимальною мовою програмування для міні ПК є Java, коректна робота з декількома потоками, зручність розробки ПЗ
- Технологією взаємодії доцільно обирати MQTT або REST