



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ (ФИЛИАЛ)
ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО
ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ
«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
В Г. ТАГАНРОГЕ РОСТОВСКОЙ ОБЛАСТИ**

ПИ (филиал) ДГТУ в г. Таганроге

ОТЧЕТ

По дисциплине (модулю) Перспективные информационные технологии
(наименование учебной дисциплины (модуля))

на тему: “Схемы распределительных систем с IoT. Виды и типы.”

Выполнил обучающийся: Дахно К.К.
(Ф.И.О.)

Направление:

09.03.02 Информационные системы и технологии
Код направления (наименование)

Обозначение отчет 2251853 Группа ВО ИСиТ-4122
номер зачетки

Проверил Доцент Орда-Жигулина М.В.
должность (Ф.И.О.)

Отчет защищён _____
дата оценка подпись

Таганрог
2025

Лабораторная работа 3.1

Схемы распределительных систем с IoT. Виды и типы.

Вариант 1.

Цель лабораторной работы: Изучить принципы построения схем распределительных систем IoT в соответствии со стандартами ГОСТ, освоить инструменты Google Colab, PlantUML Editor и draw.io для визуализации архитектуры и приобрести навык создания структурных и функциональных диаграмм (Deployment и State) для IoT-системы.

Постановка задачи:

1. Изучить теоретические основы построения схем согласно ГОСТ 2.701 и соответствие диаграмм UML типам схем по ЕСКД.
2. Освоить работу с инструментом PlantUML.
3. Выполнить индивидуальное задание (Вариант №1): разработать Deployment-диаграмму и State-диаграмму для трехуровневой архитектуры с заданными характеристиками.

Используемые средства:

- Язык описания диаграмм: PlantUML.
- Среда разработки: PlantUML Web Editor.
- Стандарты: ГОСТ 2.701 (ЕСКД. Схемы. Виды и типы).

Ход работы:

1. Анализ индивидуального задания

В соответствии с вариантом №1, проектируемая система имеет следующую конфигурацию:

- Краевые устройства (Edge): 5 шт.

					ВО ИСuT-4122			
Изм.	Лист	№ докум.	Подпись	Дат				
Разраб.		Дахно К.К.			Перспективные информационные технологии	Лит.	Лист	Листов
Провер.		Орда-Жигулина					1	5
						ПИ (филиал) ДГТУ в г.Таганроге		
Н. Контр.		Орда-Жигулина						
Утверд.		Орда-Жигулина						

- Fog-узлы (Туман): 2 шт.
- Облачные серверы (Cloud): 1 шт.
- Характеристика системы: «Много Edge на Fog» (высокая плотность подключения датчиков к промежуточным узлам).

2. Разработка Deployment-диаграммы (Структурная схема)

Была спроектирована схема физического развертывания системы. Учитывая характеристику варианта, была реализована схема с неравномерной нагрузкой:

- Первый Fog-узел обслуживает 3 датчика (подключение по ZigBee).
- Второй Fog-узел обслуживает 2 датчика (подключение по Wi-Fi).
- Оба Fog-узла передают предварительно обработанные данные на единственный облачный сервер по каналу Ethernet (протокол MQTT).

Листинг 1 – Код Deployment-диаграммы

```
@startuml
skinparam shadowing false
skinparam defaultFontName Arial
skinparam nodesep 50
skinparam ranksep 50

title Deployment-диаграмма (Вариант 1: Нагруженная)

node "Краевой уровень (Edge)" {
    component "Датчик 1" as s1
    component "Датчик 2" as s2
    component "Датчик 3" as s3
    component "Датчик 4" as s4
    component "Датчик 5" as s5
}

node "Туманный уровень (Fog)" {
    component "Fog-узел 1" as f1
    component "Fog-узел 2" as f2
}

cloud "Облачный уровень (Cloud)" {
    component "Облачный Сервер" as cloud
}

' Распределение нагрузки (3 на одного, 2 на другого)
s1 --> f1 : ZigBee
s2 --> f1 : ZigBee
```

```

s3 --> f1 : ZigBee

s4 --> f2 : Wi-Fi
s5 --> f2 : Wi-Fi

' Передача в облако
f1 --> cloud : MQTT (Ethernet)
f2 --> cloud : MQTT (Ethernet)

@enduml

```

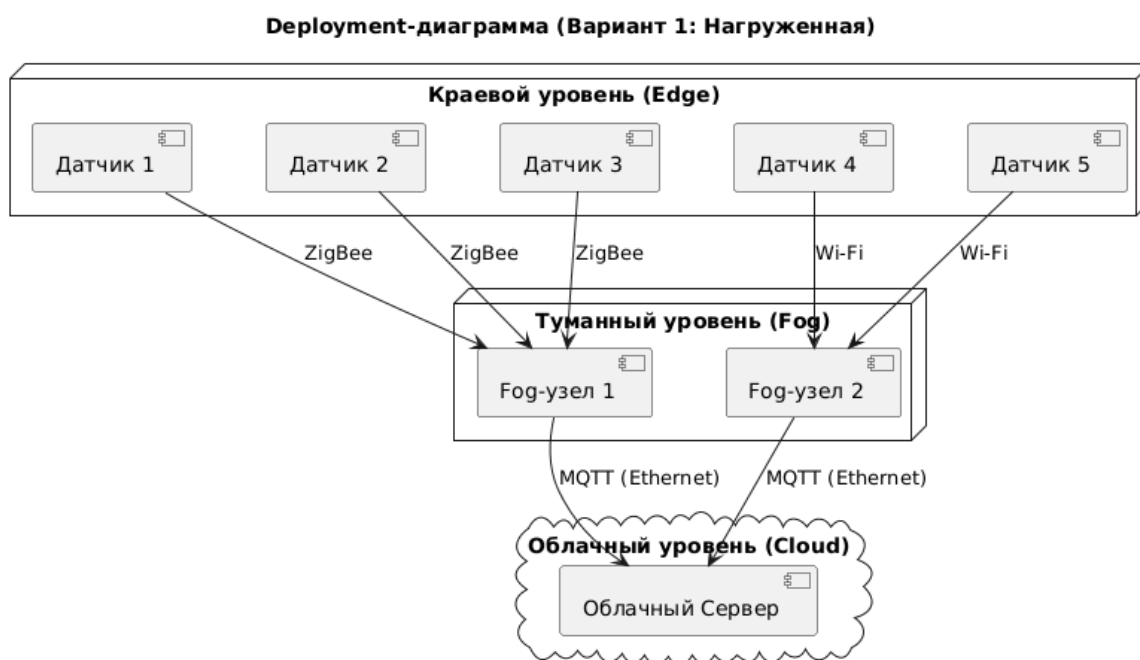


Рисунок 1 – Структурная схема распределенной системы

3. Разработка State-диаграммы (Схема состояний)

Для моделирования поведения Fog-узла в условиях высокой нагрузки была разработана диаграмма состояний. Логика работы предусматривает обработку ситуации переполнения очереди запросов от множества датчиков.

Основные переходы:

- Режим ожидания - Получение задачи: Инициация работы при поступлении сигнала.
- Получение - Перегрузка: Срабатывает, если входящий буфер переполнен (характерно для варианта с большим числом датчиков).
- Получение - Обработка - Передача: Штатный цикл работы.

Листинг 2 – Код State-диаграммы

```
@startuml
skinparam defaultFontName Arial
title State-диаграмма Fog-узла (Вариант 1)

[*] --> Режим_ожидания

state "Режим ожидания" as Режим_ожидания
state "Получение задачи" as Получение
state "Обработка данных" as Обработка
state "Передача результата" as Передача
state "Перегрузка (Очередь полна)" as Перегрузка

Режим_ожидания --> Получение : Поступили данные
Получение --> Обработка : Данные приняты
Получение --> Перегрузка : Буфер переполнен!

Обработка --> Передача : Расчет окончен
Передача --> Режим_ожидания : Успешная отправка

Перегрузка --> Режим_ожидания : Сброс/Очистка буфера

@enduml
```

State-диаграмма Fog-узла (Вариант 1)

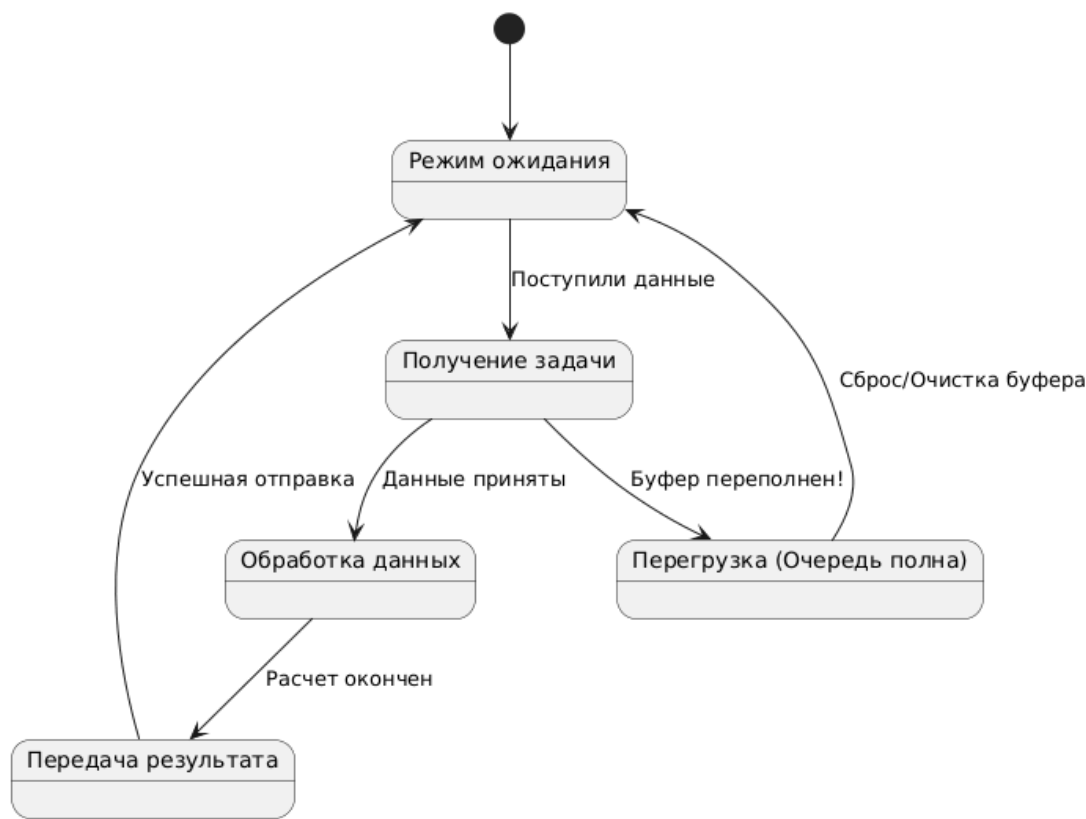


Рисунок 2 - Функциональная схема состояний Fog-узла

Вывод: В ходе выполнения лабораторной работы была спроектирована архитектура распределенной IoT-системы для варианта №1.

1. Построена Deployment-диаграмма, демонстрирующая физическую организацию сети с высокой концентрацией краевых устройств (5 Edge) на малом количестве туманных узлов (2 Fog). Схема наглядно показывает распределение потоков данных и используемые интерфейсы.
2. Построена State-диаграмма, описывающая жизненный цикл Fog-узла. В модель включено состояние «Перегрузка», что соответствует специфике задания (риск переполнения очереди из-за большого количества подключенных устройств).
3. Полученные навыки работы с PlantUML позволяют быстро создавать и модифицировать техническую документацию в соответствии с принципами «Infrastructure as Code».