



«Пермский Государственный Национальный  
Исследовательский Университет»

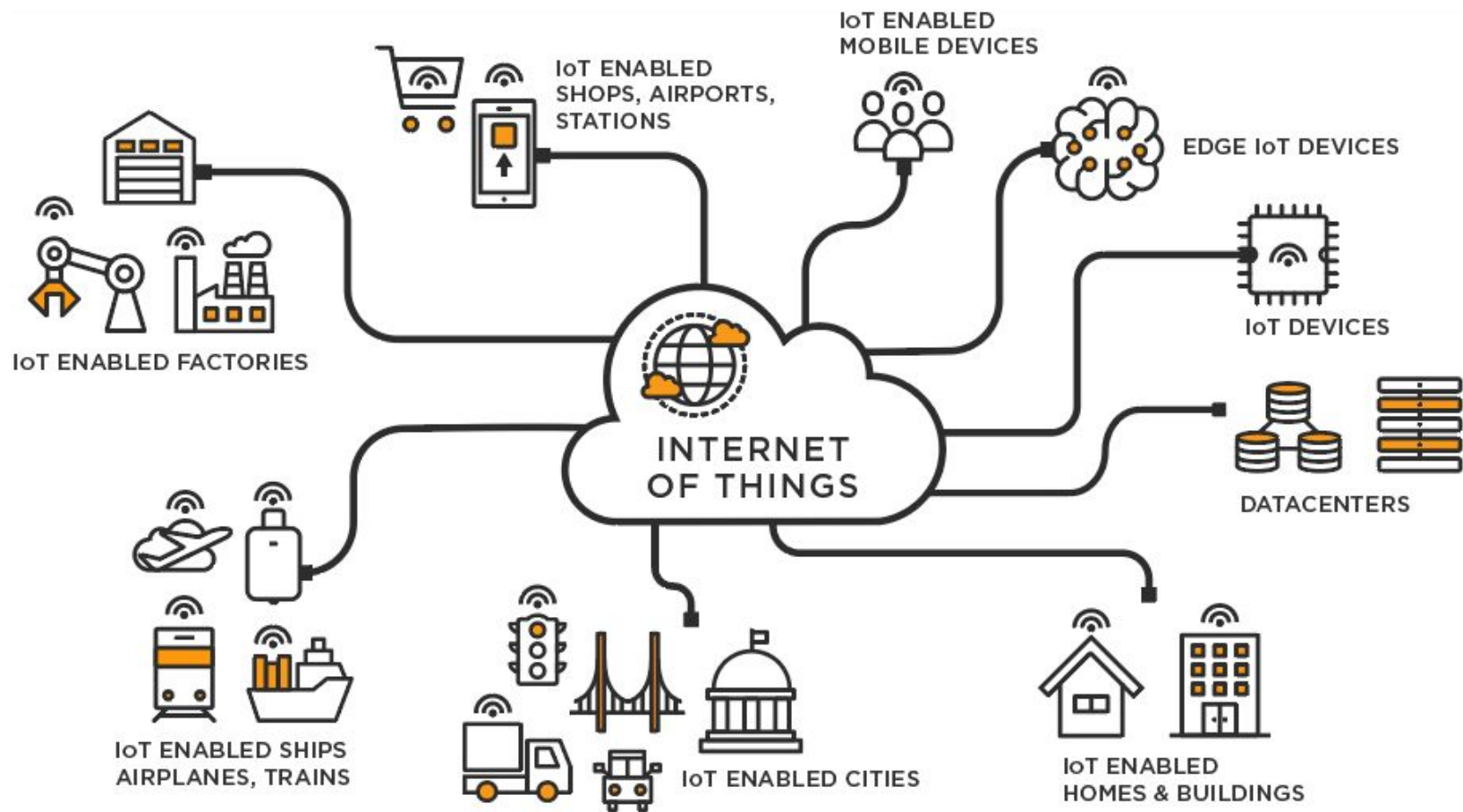
Кафедра математического обеспечения  
вычислительных систем

# **Разработка средств автоматизации программирования устройств Интернета вещей на базе платформы SciVi**

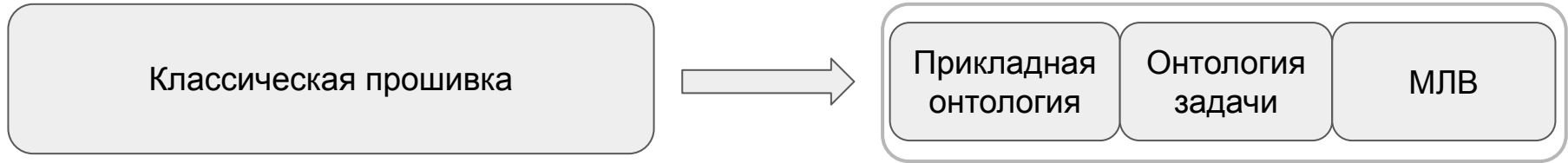
**Выпускная квалификационная работа бакалавра**

Работу выполнил: студент группы  
ПМИ-1,2-2019 4 курса механико-  
математического факультета  
Лукьянов Александр Михайлович

Научный руководитель:  
Кандидат физико-математических наук,  
доцент кафедры МОВС  
Рябинин Константин Валентинович



# Онтологически управляемые периферийные вычисления (Ontology-Driven Edge Computing, ODEC)

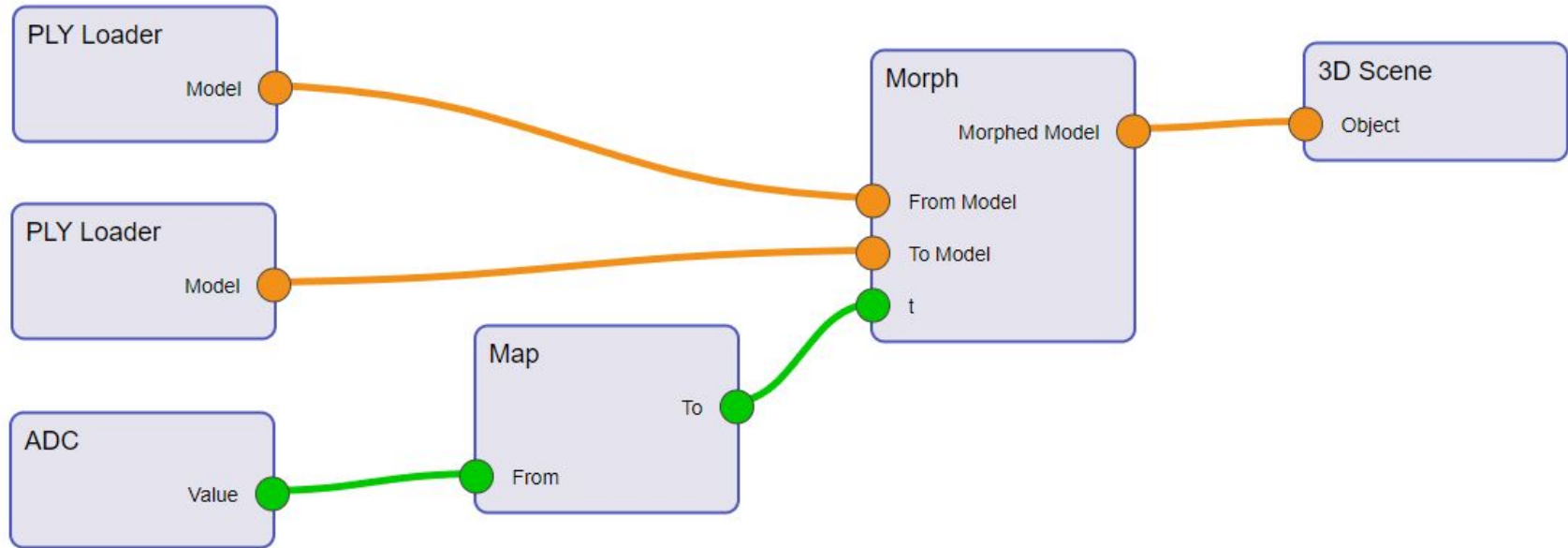


- Управление периферийными устройствами только с помощью онтологий
- Создание унифицированных интерфейсов
- Объединение в общую сеть разнородных устройств
- Быстрое перепрограммирование и реконфигурация
- Упрощение мониторинга работы устройств
- **Автоматизация процесса программирования устройств и снижение необходимого уровня квалификации разработчика**



ADVANCED SCIENTIFIC VISUALIZATION AND VISUAL ANALYTICS

# Редактор диаграмм потоков данных SciVi



Объект исследования:

автоматизация периферийных вычислений.

Предмет исследований:

средства платформы SciVi для организации онтологически управляемых периферийных вычислений.

## Цель ВКР:

Создание комплексного решения по автоматизации программирования устройств Интернета вещей на базе платформы SciVi путём реализации недостающей функциональности в рамках концепции онтологически управляемых периферийных вычислений

# Недостающая функциональность платформы SciVi

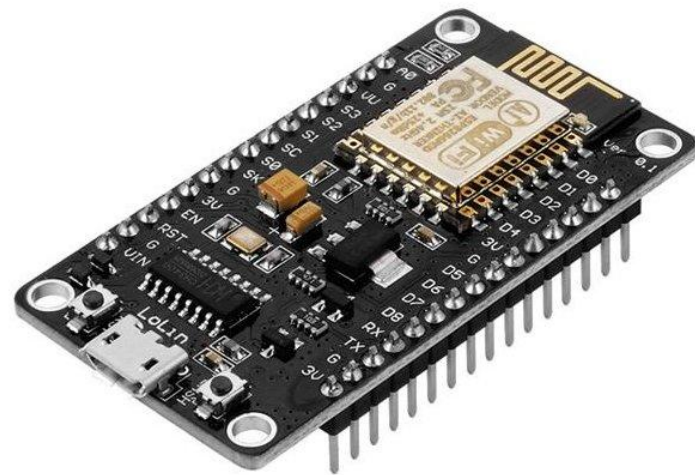
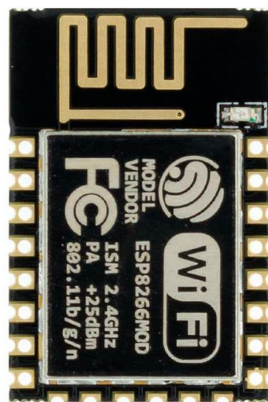
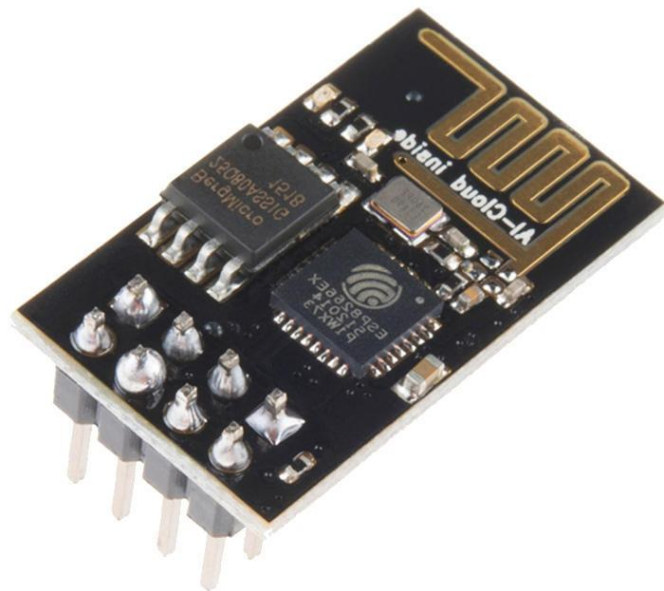
- Автоматическое обнаружение периферийных устройств в локальной сети
- Возможности эффективного взаимодействия с энергонезависимой памятью периферийных устройств



# Задачи:

1. Провести анализ литературы по тематике Интернета вещей и онтологически управляемых периферийных вычислений.
2. Изучить принципы функционирования платформы визуальной аналитики SciVi.
3. Провести анализ литературы и популярных решений в областях автоматизации обнаружения устройств Интернета вещей в локальной сети и взаимодействия с их энергонезависимой памятью.
4. Спроектировать и разработать программные решения для автоматизации обнаружения устройств Интернета вещей в локальной сети и взаимодействия с их энергонезависимой памятью с учётом особенностей подхода онтологически управляемых периферийных вычислений и платформы SciVi.
5. Интегрировать разработанные решения в платформу SciVi и провести комплексное тестирование средств автоматизации программирования устройств Интернета вещей на базе этой платформы.

# Целевые устройства. ESP8266



# Менеджер энергонезависимой памяти (EEPROM) устройств Интернета вещей

# Требования к программному модулю

- 1. Наличие возможности сохранять и считывать данные произвольной структуры из EEPROM.**
- 2. Высокоуровневый интерфейс.**
- 3. Обращение к хранимым данным по именам, а не физическим адресам.**
- 4. Минимизация количества операций записи**

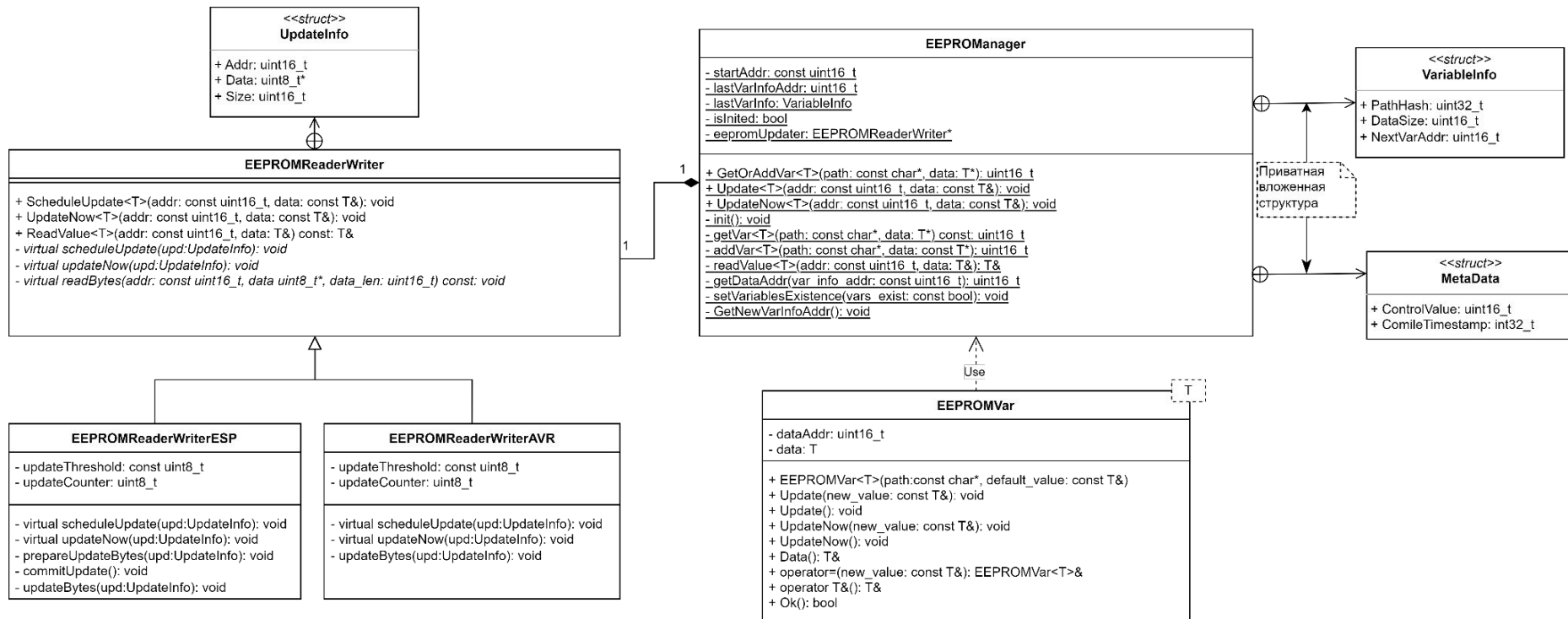
# Популярные решения

- Стандартные библиотеки
- EEManager
- EEPROMEx
- EEPROMWearLevel

# Общие недостатки

- **Взаимодействие с данными только через их адреса**
- **Отсутствие оптимизации для EPS8266**
- **Хранение данных только базовых типов**

# Диаграмма классов библиотеки

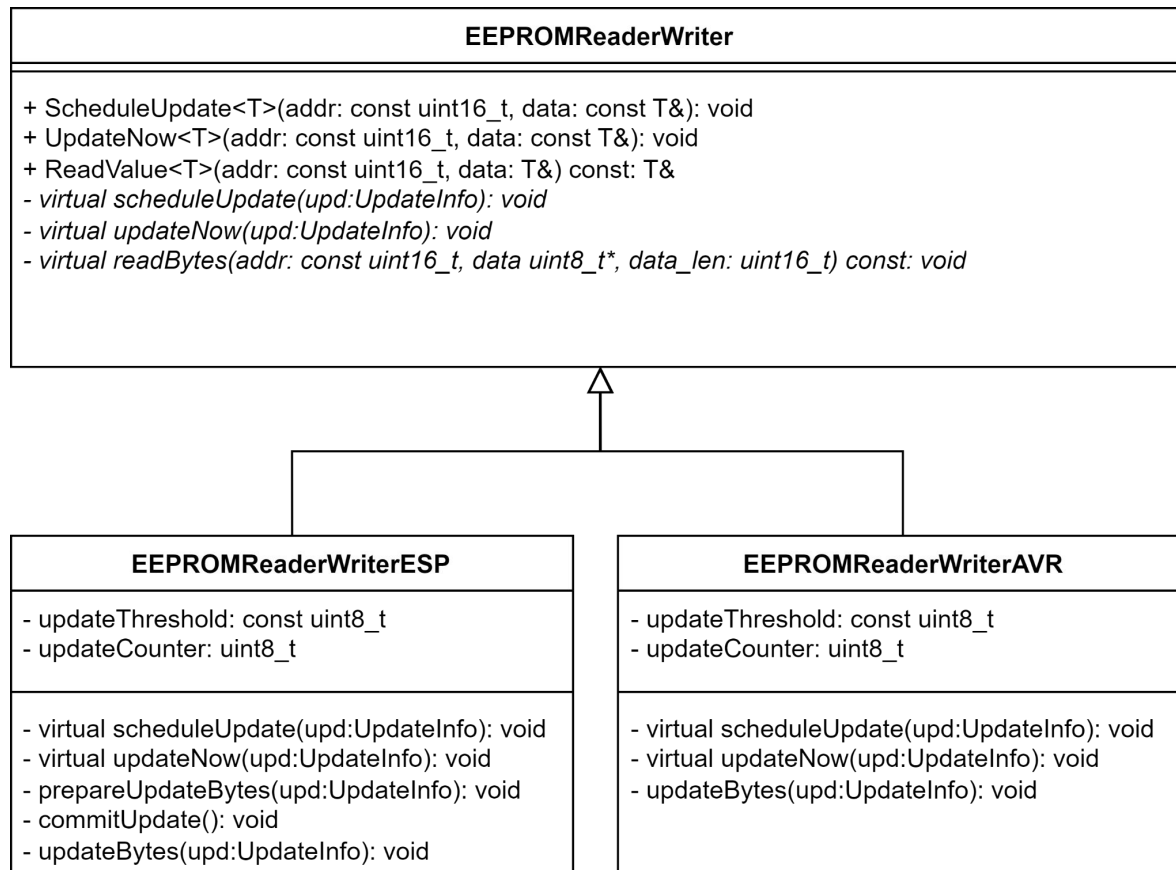


# Класс чтения/записи

## EEPROMReaderWriter

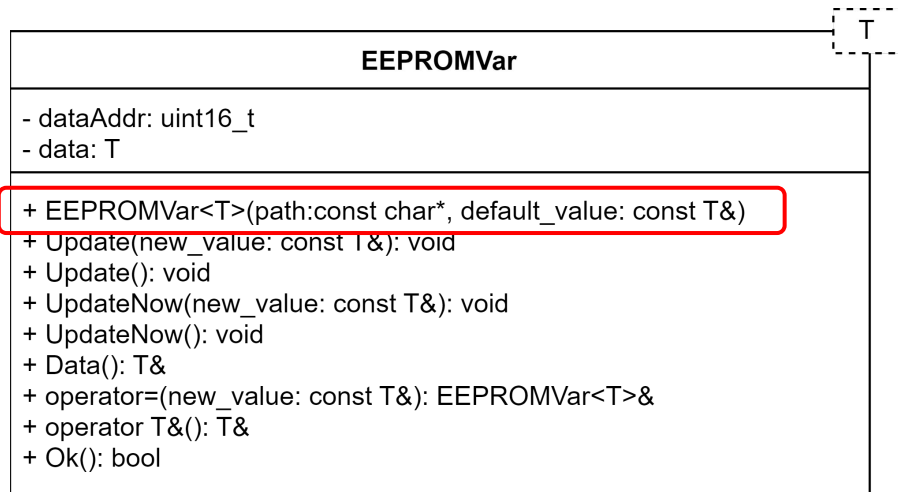
```
+ ScheduleUpdate<T>(addr: const uint16_t, data: const T&): void  
+ UpdateNow<T>(addr: const uint16_t, data: const T&): void  
+ ReadValue<T>(addr: const uint16_t, data: T&) const: T&  
- virtual scheduleUpdate(upd:UpdateInfo): void  
- virtual updateNow(upd:UpdateInfo): void  
- virtual readBytes(addr: const uint16_t, data uint8_t*, data_len: uint16_t) const: void
```

# Класс чтения/записи





# Интерфейс библиотеки



```
1 EEPROMVar<float> var("float_var", 0.5);  
2  
3 var = 10.;  
4  
5 float x = var;
```

# Интерфейс библиотеки

EEPROMVar
- dataAddr: uint16_t - data: T
+ EEPROMVar<T>(path:const char*, default_value: const T&) + Update(new_value: const T&): void + Update(): void + UpdateNow(new_value: const T&): void + UpdateNow(): void + Data(): T& + operator=(new_value: const T&): EEPROMVar<T>& + operator T&(): T& + Ok(): bool

```
1 EEPROMVar<float> var("float_var", 0.5);  
2  
3 var = 10.;  
4  
5 float x = var;
```

# Преимущества разработанной библиотеки

- **Автоматическая адресация на основе имён**
- **Автоматическое приведение из/к типу хранимых данных**
- Адаптируемость для различных устройств
- Оптимизация количества операций записи для ESP8266

Разработанная библиотека полностью удовлетворяет поставленным требованиям для использования в рамках платформы SciVi

Средство автоматизации обнаружения периферийных устройств в локальной сети

# Анализ протоколов самоидентификации

Критерий	NetBIOS	DNS-SD	mDNS	SSDP	SLP
Возможность обмена дополнительными данными	+	-	-	+	-
Возможность поиска по именам	+	+	+	+	+
Возможность поиска по типам	-	+	-	+	+
Возможность поиска с помощью предикатов	-	-	-	-	+
Возможность поиска не только устройств, но и сервисов	-	+	-	+	+
Наличие дополнительных механизмов защиты	-	-	-	-	+
Наличие механизмов уведомления о вхождении устройства в сеть	+	-	-	+	+
Бинарный протокол представления данных	-	+	+	-	+

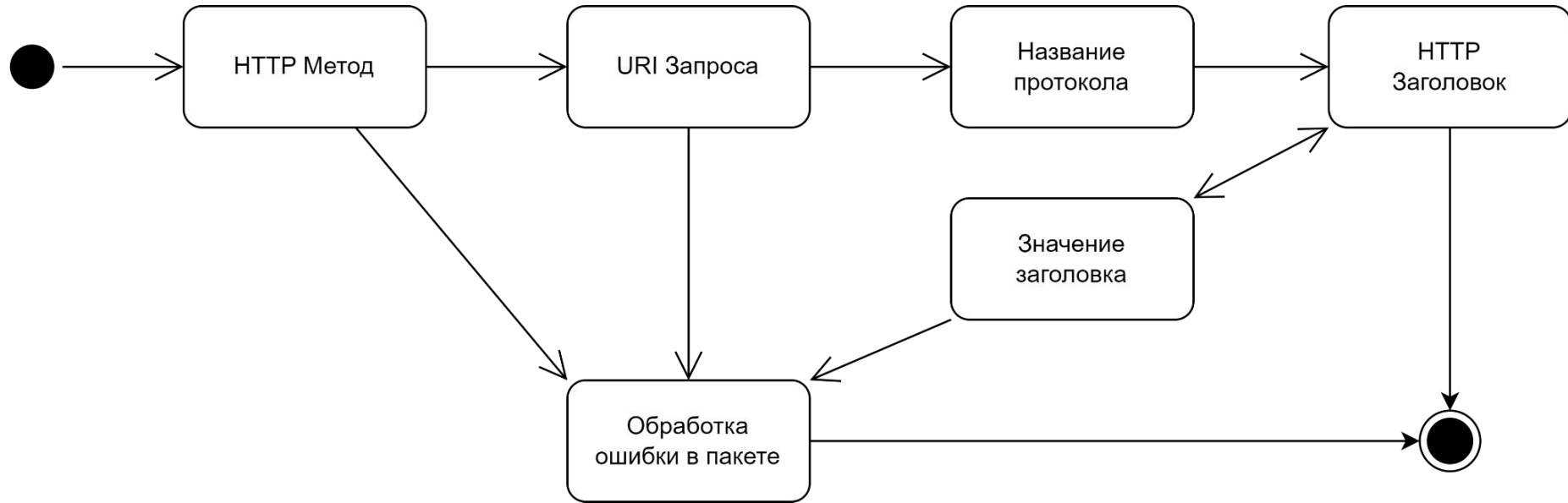
# Протокол SSDP. Преимущества

- Возможность дополнения процесса поиска передачей дополнительных данных
- Поиск конкретных устройств
- Поиск всех устройств некоторого типа
- Поиск всех устройств, предоставляющих определённый сервис
- Автоматические уведомления о вхождении устройства в сеть

# Недостатки существующей реализации

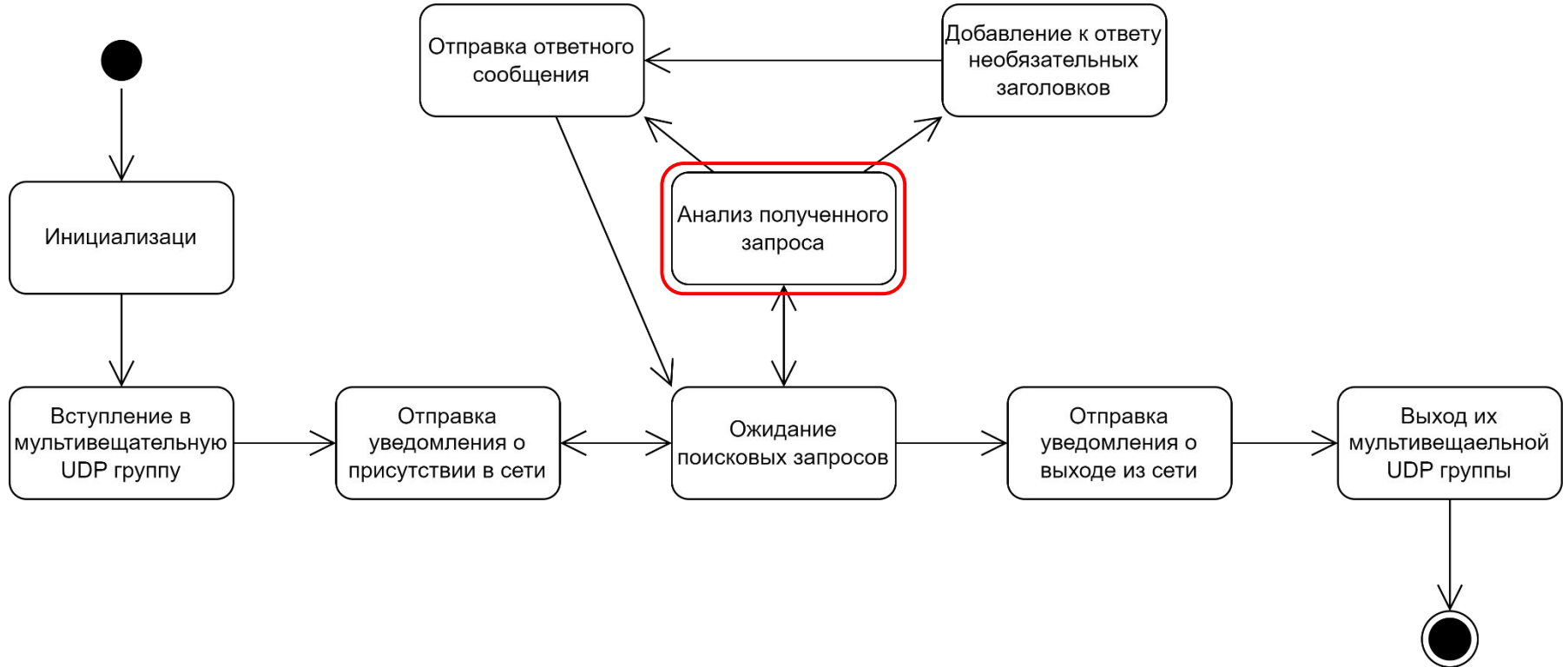
- Отсутствие возможностей для расширения протокола
- Реализация устаревшей версии протокола
- Нарушения стандарта SSDP
- Реализованы не все возможности протокола

# Этапы синтаксического разбора поисковых пакетов





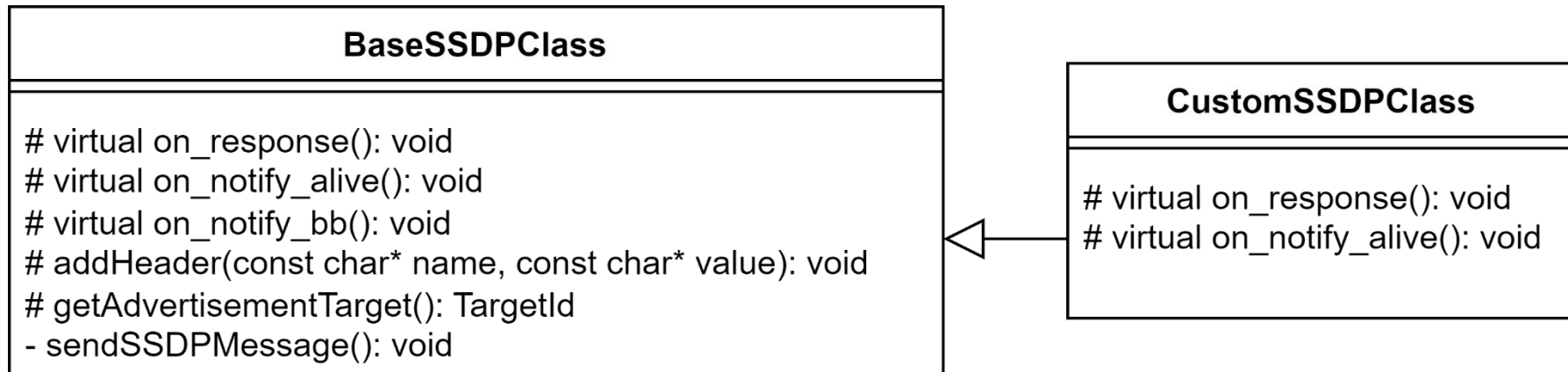
# Диаграмма рабочего цикла объекта SSDP



# Диаграмма рабочего цикла объекта SSDP



# Расширение протокола



# Достигнутые результаты

- Изучены принципы работы сетей Интернета вещей и подхода онтологически управляемых периферийных вычислений
- Исследована платформа SciVi. Выявлены её недостающие компоненты, необходимые для комплексного решения на её основе задачи автоматизации программирования устройств Интернета вещей
- Проанализированы распространённые существующие решения аналогичных задач
- Недостающие компоненты были спроектированы, реализованы, успешно прошли тестирование и частично интегрированы в платформу SciVi
- **Новая функциональность, добавленная в платформу SciVi в рамках данной работы, позволила вывести её средства программирования устройств Интернета вещей на новый уровень автоматизации**

# Апробация результатов работы

- Доклад “Разработка онтологически управляемого протокола сетевой самоидентификации устройств интернета вещей” на VII студенческой научно-практической конференции “Математическое и программное обеспечение информационных и интеллектуальных систем” (ПГНИУ, 28-29 апреля 2022 г.). Получена грамота 1-ой степени
- Одноимённая была опубликована в сборнике статей “Актуальные проблемы математики, механики и информатики 2022”
- Доклад “Разработка средств автоматизации программирования устройств Интернета вещей на базе платформы SciVi” на VIII студенческой научно-практической конференции “Математическое и программное обеспечение информационных и интеллектуальных систем” (ПГНИУ, 24 мая 2023 г.). Получена грамота 3-ей степени

# Спасибо за внимание!

2. [Интернет вещей](#)
3. [ODEC](#)
4. [SciVi](#)
5. [Графический редактор SciVi](#)
6. [Объект и предмет исследования](#)
7. [Цель работы](#)
8. [Подцели](#)
9. [Задачи](#)
10. [Микроконтроллер ESP8266](#)
11. [Библиотека менеджера EEPROM](#)
12. [Требования к менеджеру EEPROM](#)
13. [Популярные решения](#)
14. [Диаграмма классов менеджера](#)
15. [Абстрактный класс чтения/записи](#)
16. [Реализация классов чтения/записи](#)
17. [Внешний интерфейс менеджера](#)
19. [Преимущества разработанного решения](#)
20. [Автоматизации обнаружения устройств](#)
21. [Анализ протоколов самоидентификации](#)
22. [Протокол SSDP](#)
23. [Существующая реализация SSDP](#)
24. [Синтаксический анализ запросов](#)
25. [Диаграмма рабочего цикла](#)
27. [Средства расширения протокола](#)
28. [Достигнутые результаты](#)
29. [Апробация результатов](#)