

# Стандарты разработки для языков МЭК от PLCopen

Конференция пользователей 2018  
Осинский Алексей, ОВЕН

**1**

**Вступление**

**2**

**Руководство по языку SFC**

**3**

**Руководство по созданию библиотек**

**4**

**Руководство по кодированию**

**5**

**Заключение**

## **Предпосылки стандартизации у нас**

- **«Кто в лес, кто по дрова»**
- **Отсутствие сопроводительной документации**
- **Нет предрелизного анализа кода**
- **Большие затраты на поиск ошибок**

## **Предпосылки перевода**

- **Разработка корпоративного стандарта**
- **В русскоязычном сегменте МЭК программистов такого нет**
- **Большое количество свободного времени**

# PLCopen

- Международная некоммерческая организация
- Популяризирует стандарт МЭК 61131-3
- Облегчает использование стандарта МЭК 61131-3

1

Вступление

2

Руководство по языку SFC

3

Руководство по созданию библиотек

4

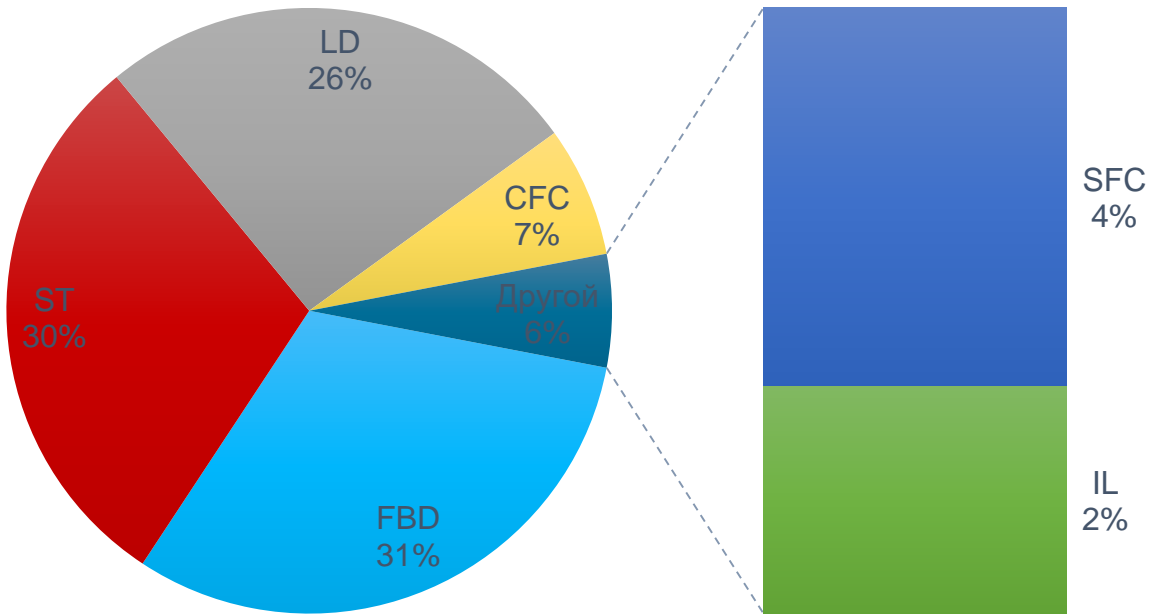
Руководство по кодированию

5

Заключение

Сколько людей  
использует SFC?

Статистика\* использования языков МЭК на постсоветском  
пространстве

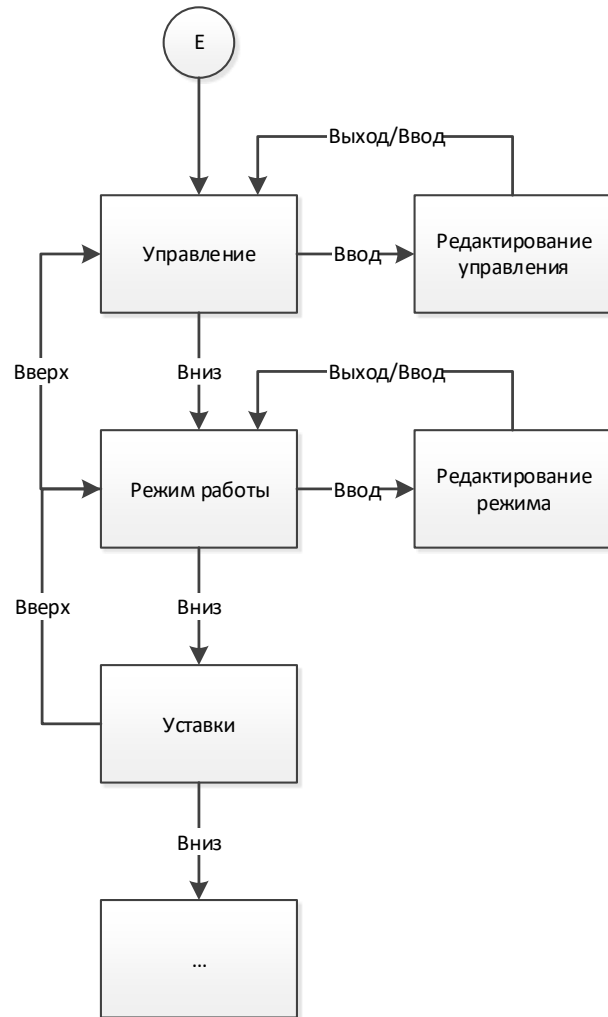


## Немного о SFC

- Используется для разработки конечных автоматов
- Выразительный графический язык
- Использовать может даже не программист
- Программы на SFC легко пишутся в процессе согласования требований к ПО



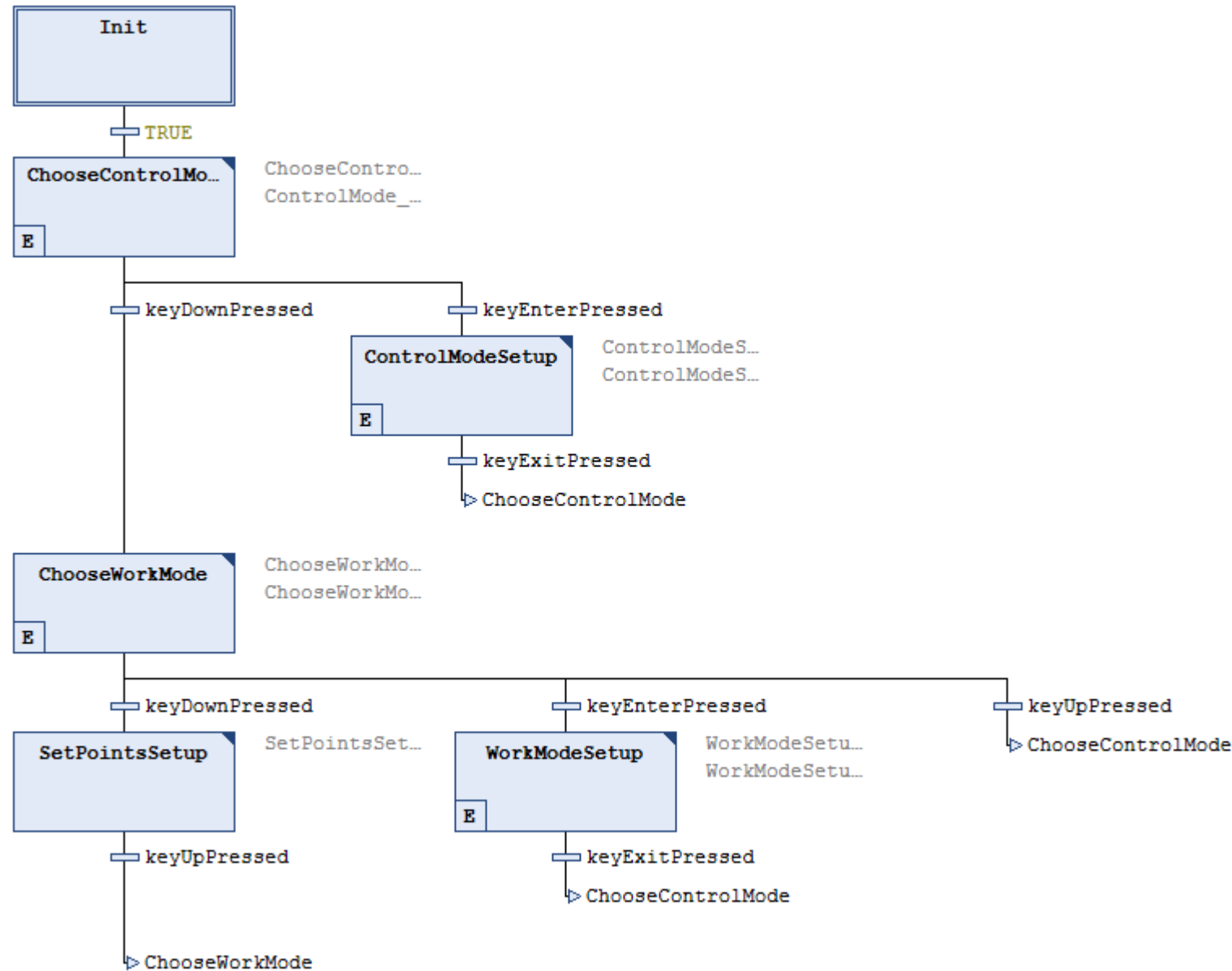
## Как мы раньше делали?



```

CASE eCurrentScreen OF
  CHOOSE_CONTROL_MODE:
    IF keyDownPressed THEN
      eCurrentScreen := CHOOSE_WORK_MODE;
    ELSIF keyEnterPressed THEN
      eCurrentScreen := SETUP_CONTROL_MODE;
    END_IF
    ...
  SETUP_CONTROL_MODE:
    ...
  CHOOSE_WORK_MODE:
    ...
    ...
END_CASE
  
```

## То же решение на SFC



## О чем документ?

- Назначение языка
- Описание элементов языка
- Ограничения языка
- Особенности реализации на разных платформах

1

Вступление

2

Руководство по языку SFC

3

Руководство по созданию библиотек

4

Руководство по кодированию

5

Заключение

## О чем документ?

- Описание стандартных интерфейсов
- Описание поведения блоков с выбранным интерфейсом
- Заготовки кода (ST, SFC)

## О чем документ?

- **Уже работали с библиотеками САА**
  - CAA File
  - CAA DTutil
  - CAA Net Base Services
  - CAA SerialCom
  - ...

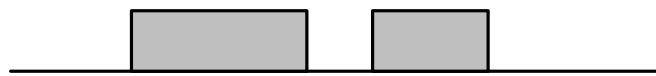
## Зачем мне использовать это?

- Сходу становится понятно, как этот ФБ работает
- Возможность сконцентрироваться на решении задачи
- Экономит время как разработчику, так и пользователю

Обзор моделей поведения

Работа по уровню

Разрешение на работу



Активность блока



Работа по фронту

Разрешение на работу



Активность блока

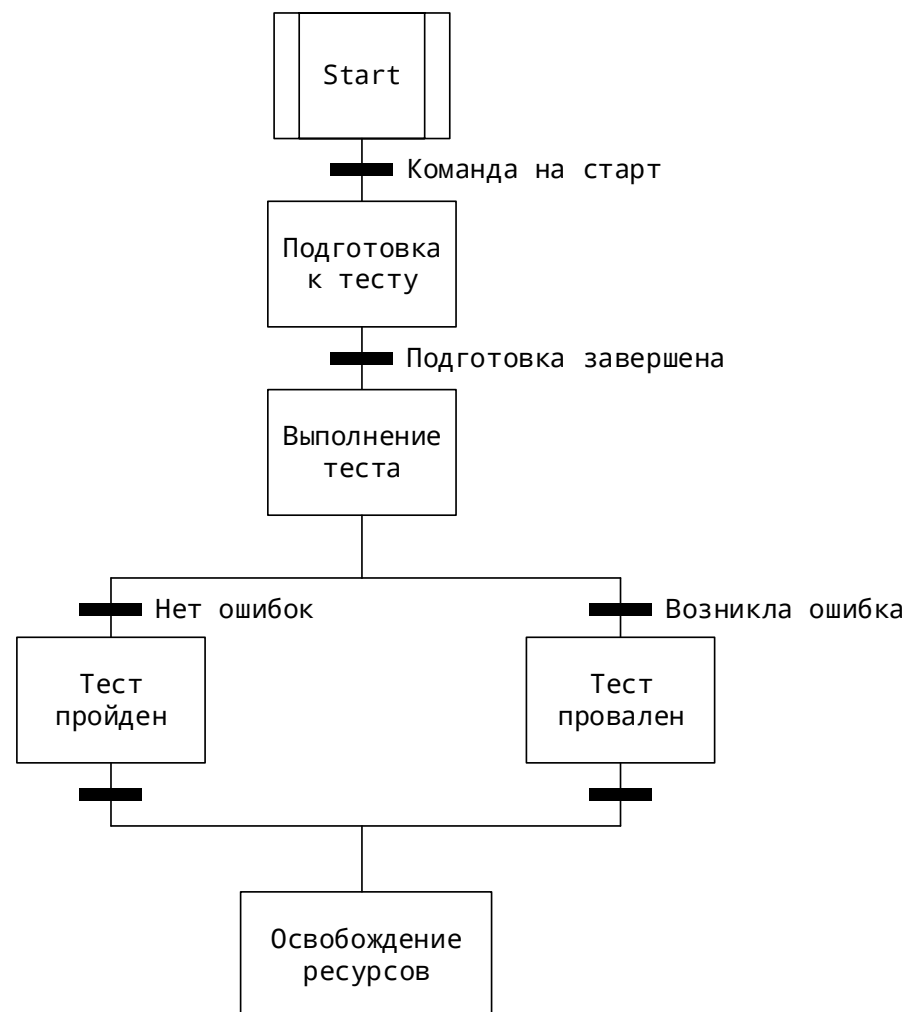




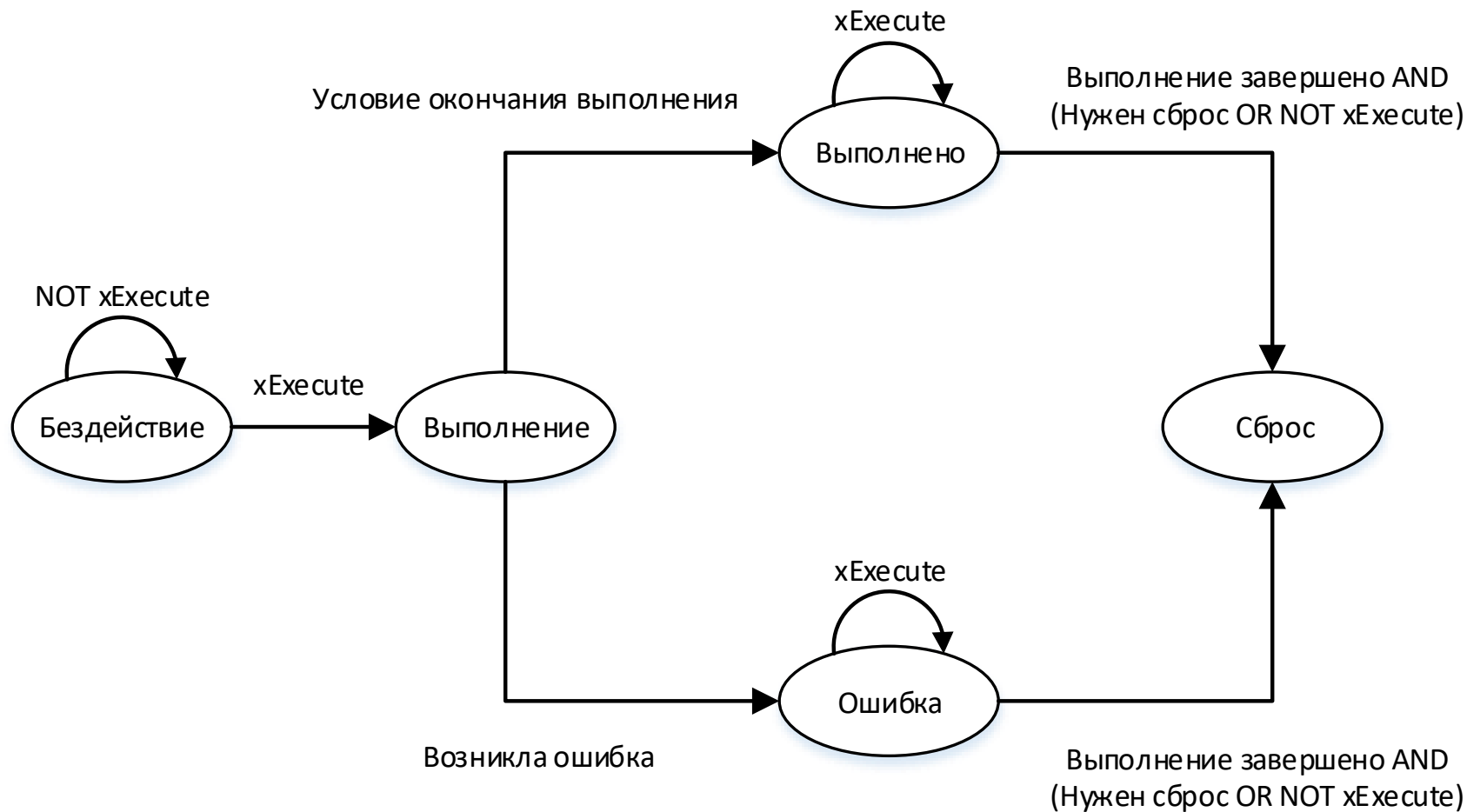
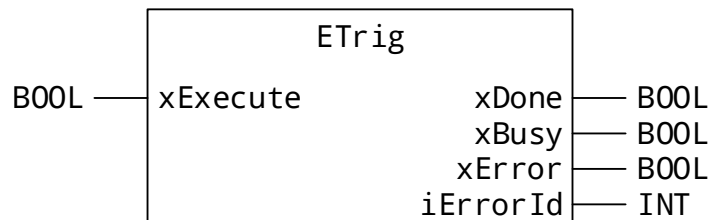
## Обзор моделей поведения

- Возможность досрочно прервать работу блока
- Назначить таймаут выполнения задачи
- Ограничить время выполнения блока в цикле

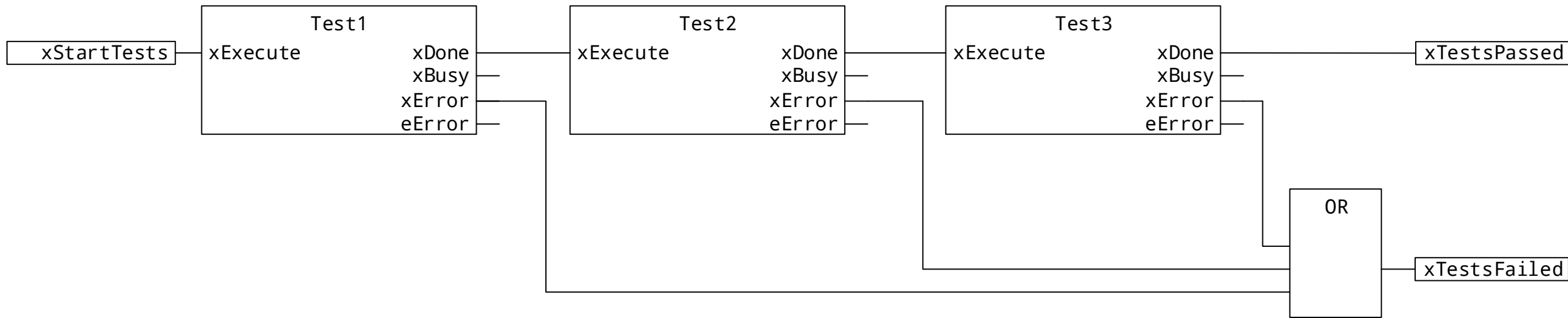
## Стандартный ход теста



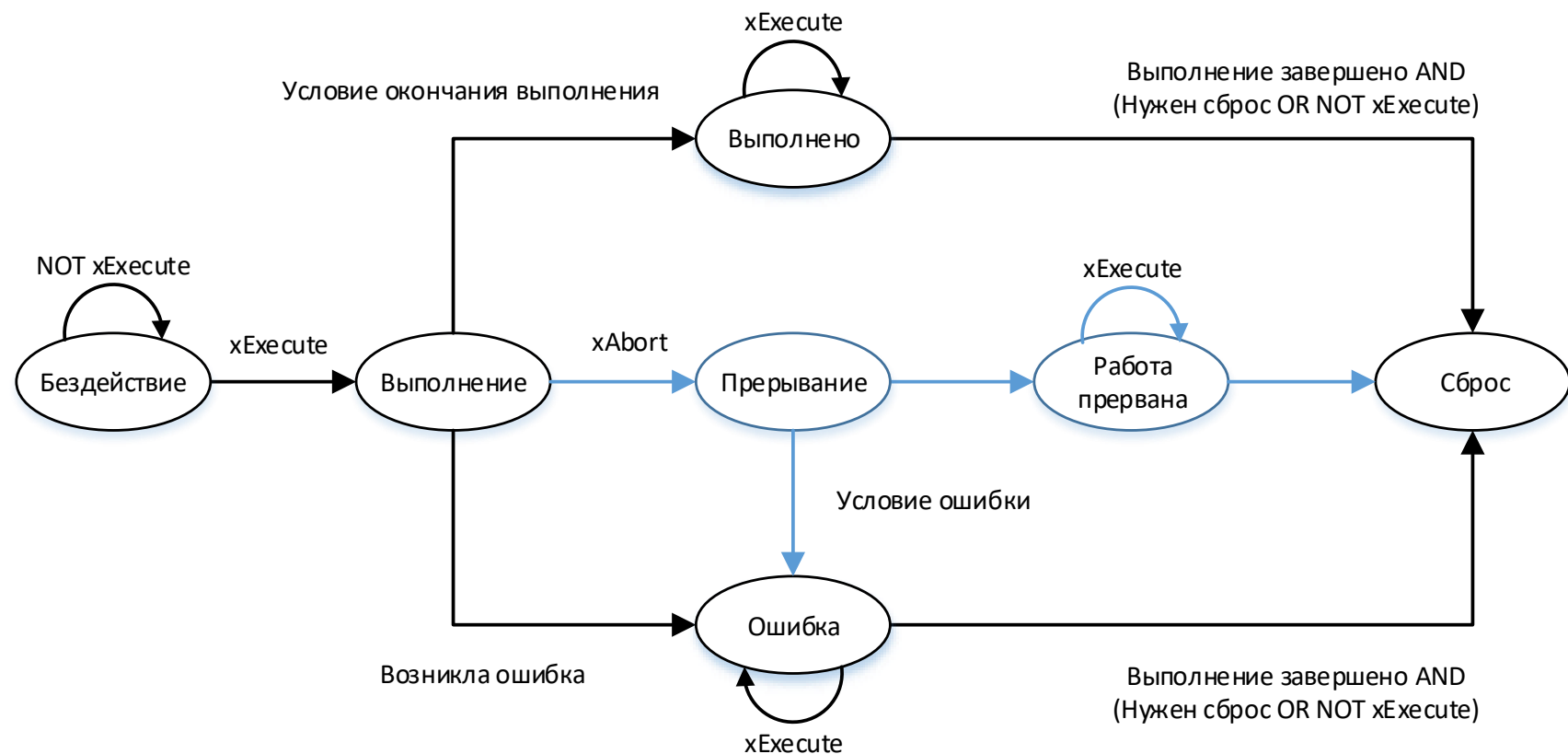
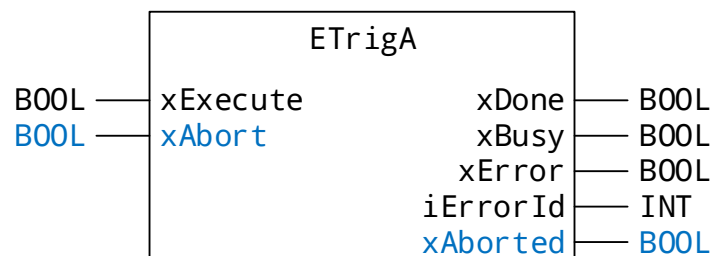
## Модель поведения ETrig



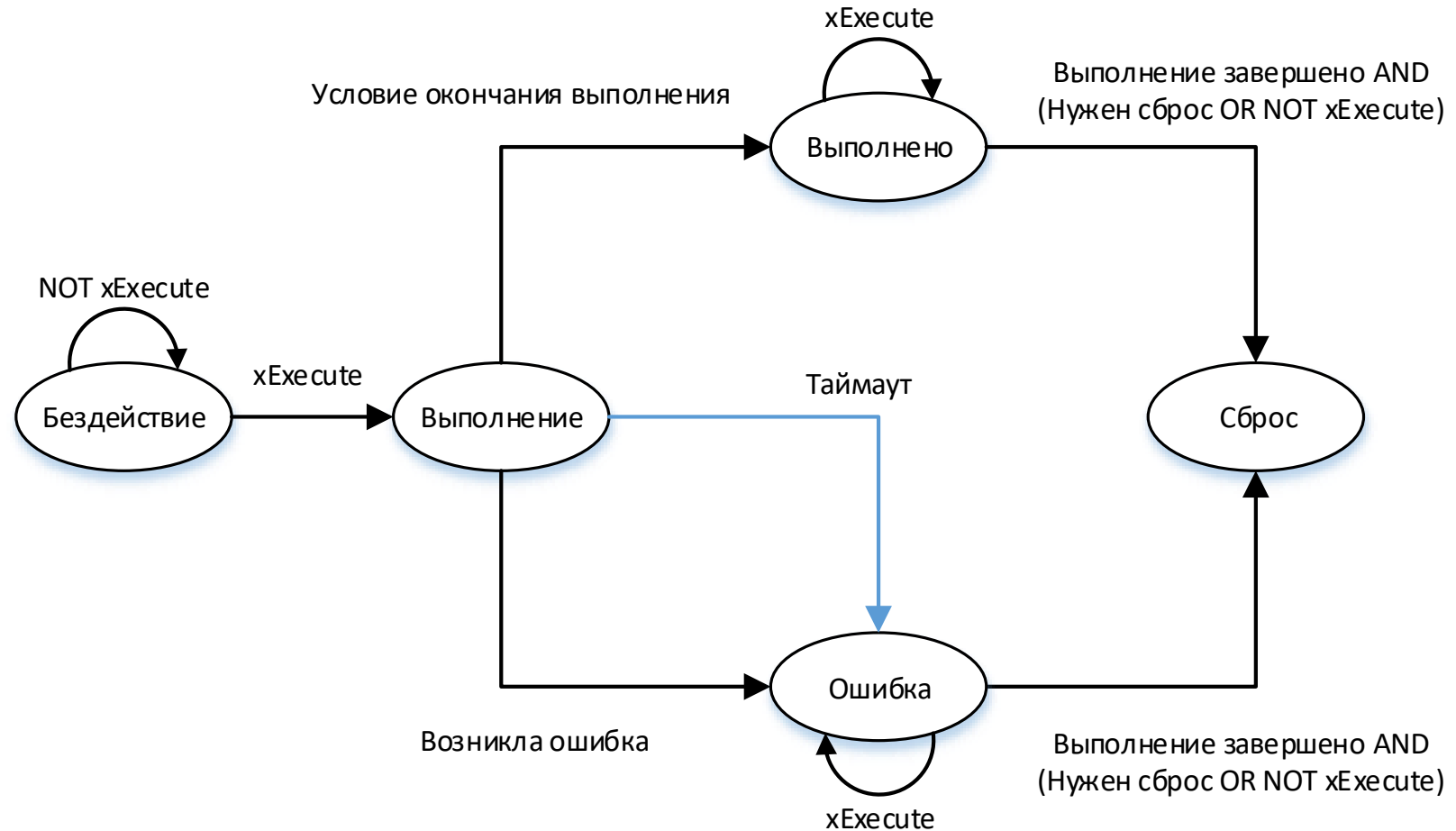
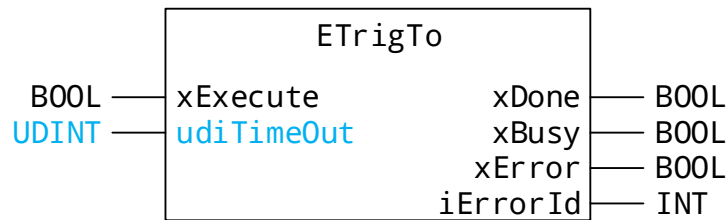
## Тестирующая программа, основанная на ETrig



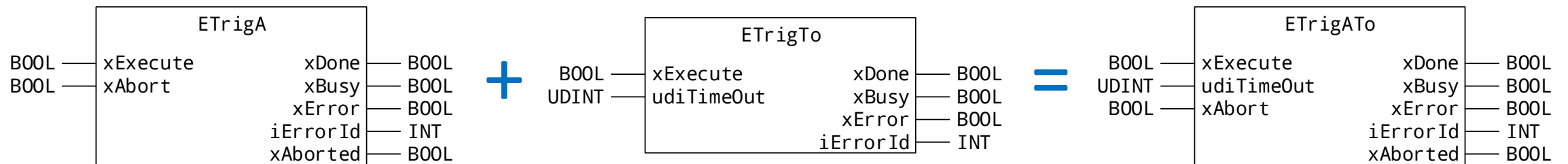
## Модель поведения ETrigA



## Модель поведения ETrigTo



## Комбинирование различных моделей поведения



1

Вступление

2

Руководство по языку SFC

3

Руководство по созданию библиотек

4

Руководство по кодированию

5

Заключение



## О чем документ?

- Набор рекомендаций для повышения качества кода
- Пояснения к рекомендациям
- Языки, для которых применима рекомендация
- Примеры применения

## Зачем стандартизировать код?

- **Если работаете в команде**
  - Уменьшение затрат на анализ кода
  - Уменьшение количества ошибок
  - Переносимость кода между платформами
  
- **Если работаете самостоятельно**

## Возражения против стандартизации кода

- «... я работаю сам ...»
- «... да в моем коде и так все понятно ...»
- «... я уже привык по-другому...»
- «... следование им отвлекает от кодирования ...»

## Что мы взяли себе?

- Применяйте префиксы

```
uiCounterDifference := uiCounter1 - uiCounter2;
```

## Что мы взяли себе?

- Комментарии должны описывать назначение кода

```
// Если a равно b  
IF a = b THEN  
...  
END_IF
```



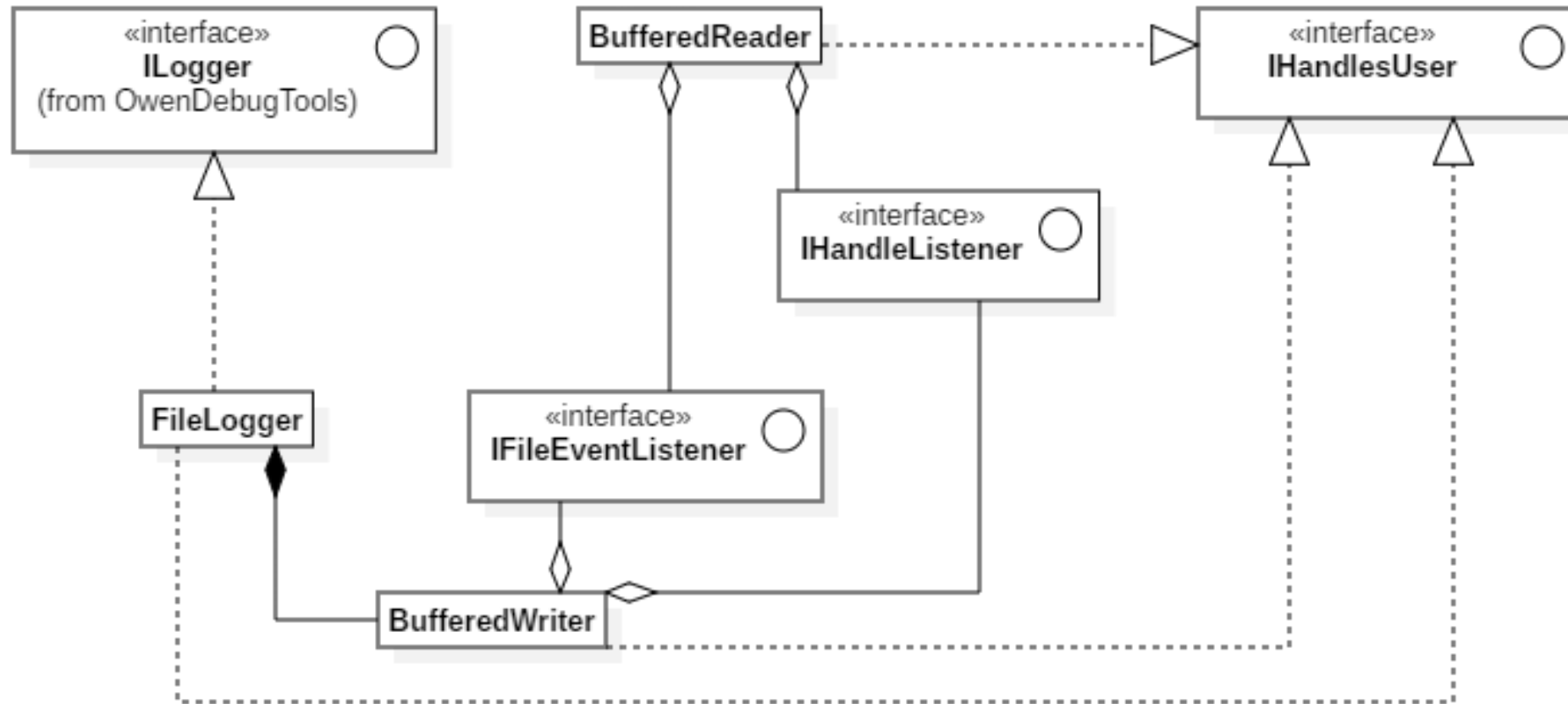
```
// Если достигли уставки  
IF a = b THEN  
...  
END_IF
```

## Что мы взяли себе?

- Комментарии должны описывать назначение кода
- Все объекты должны быть прокомментированы
- Комментарии не должны содержать код
- Используйте однострочные комментарии

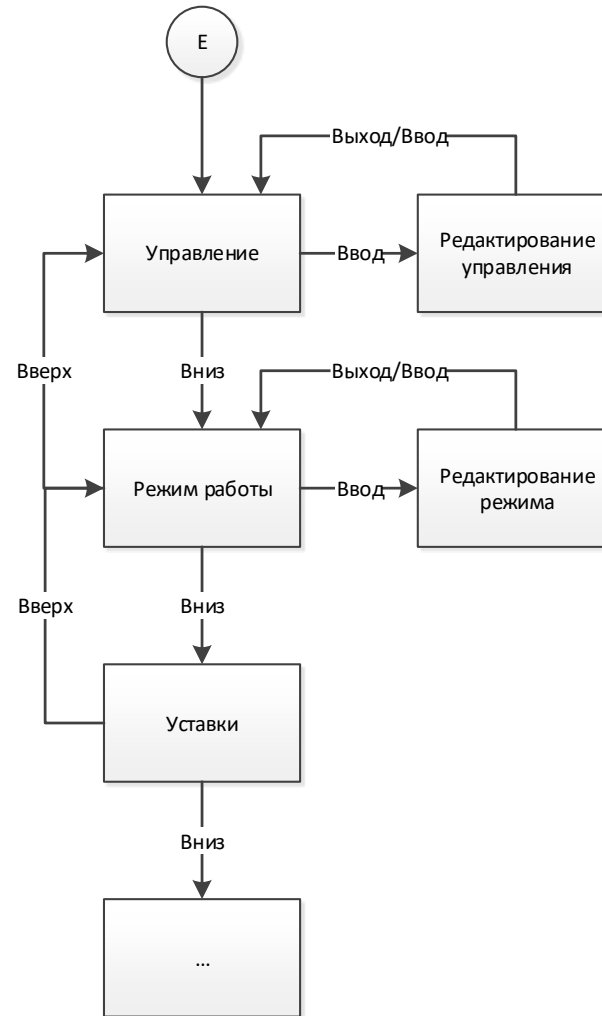
## Результаты проектирования

### Диаграммы классов



## Результаты проектирования

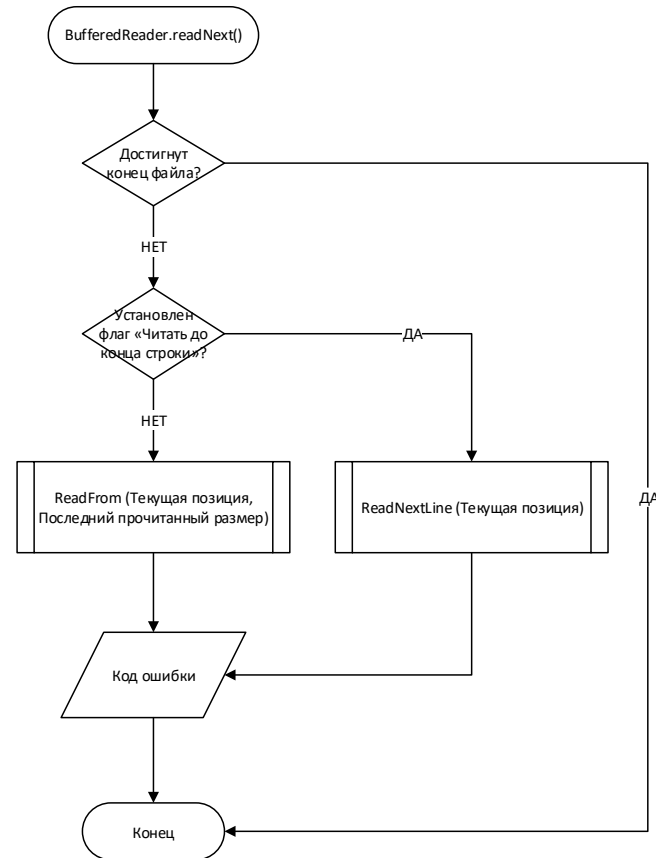
### Диаграммы состояний





## Результаты проектирования

### Алгоритмы работы блоков



## Что мы взяли себе?

**POU должен иметь одну точку выхода**

```
IF xErrorReason1 THEN
    // Обработка ошибки 1
    RETURN;
END_IF
IF xErrorReason2 THEN
    // Обработка ошибки 2
    RETURN;
END_IF

// Полезная работа здесь
```



```
IF xErrorReason1 THEN
    // Обработка ошибки 1
    ;
ELSIF xErrorReason2 THEN
    // Обработка ошибки 2
    ;
ELSE
    // Полезная работа здесь
END_IF
```

## Что мы взяли себе?

- Не используйте прямую адресацию
- Осуществляйте обработку ошибок
- Значения с плавающей точкой не должны проверяться на равенство и неравенство
- Разумно используйте глобальные переменные
- Используйте для переменных наиболее подходящий тип данных
- Используйте явные преобразования типов

1

Вступление

2

Руководство по языку SFC

3

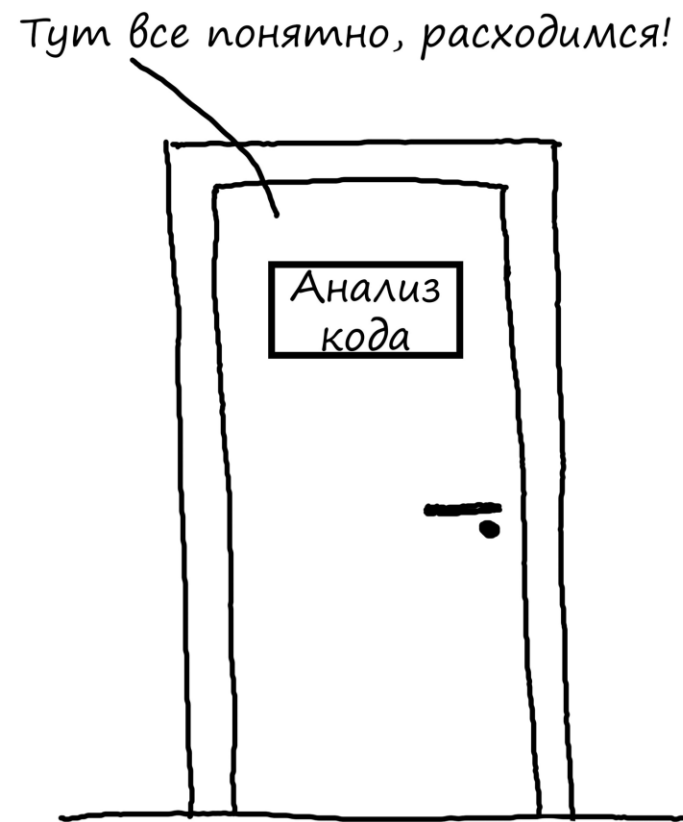
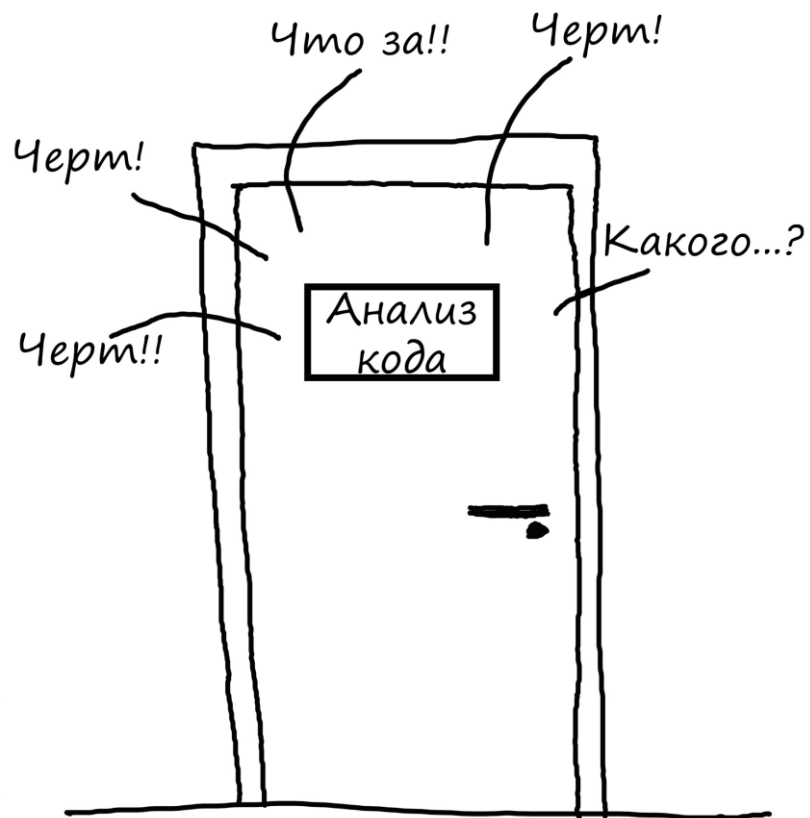
Руководство по созданию библиотек

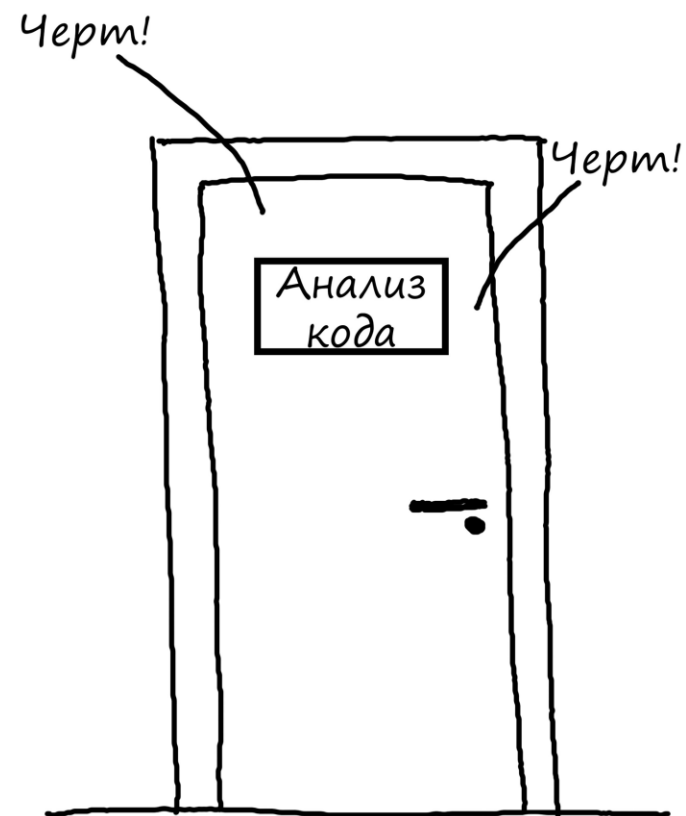
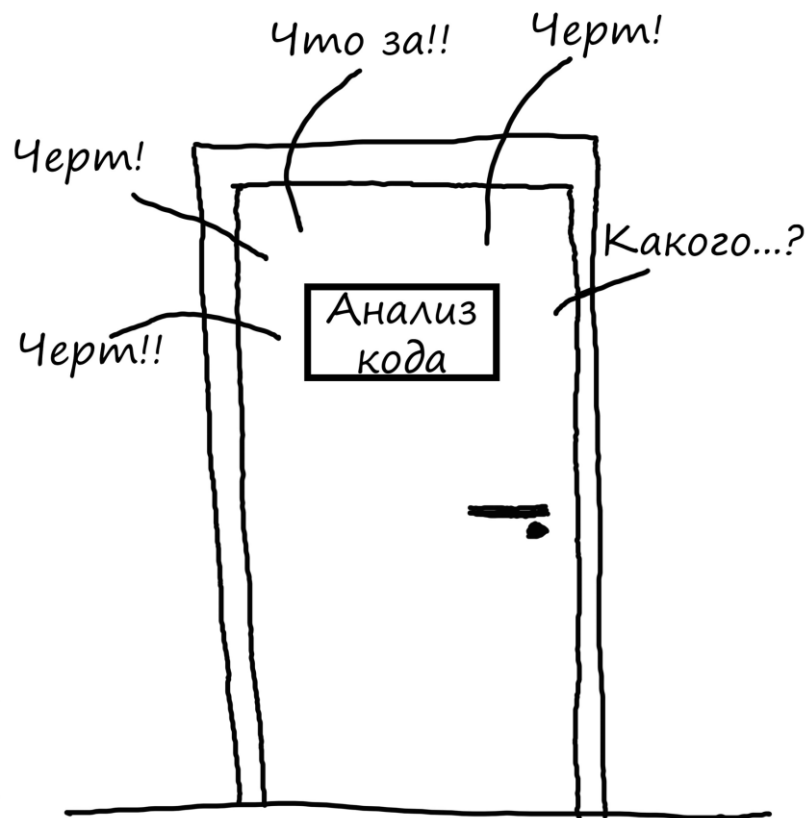
4

Руководство по кодированию

5

Заключение





## Где все это взять?

Документы:

<https://oscat-ru.weebly.com/>

Документы в оригинале:

<http://www.plcopen.org/>

Заметили ошибку?

<mailto:OscatLibRu@gmail.com>

Пообщаться на околопрограммистские темы:

<mailto:a.osinskiy@owen.ru>

Skype: a.osinskiy\_owen





**Спасибо за внимание.**

**CODESYS® is a registered trademark of 3S-Smart Software Solutions GmbH. Technical specifications are subject to change. Errors and omissions excepted. No reproduction or distribution, in whole or in part, without prior permission.**

**Note: Not all CODESYS features are available in all territories. For more information on geographic restrictions, please contact [sales@codesys.com](mailto:sales@codesys.com).**