



УНИВЕРСИТЕТ ИТМО

# Итоги изучения прошивки потенциостата

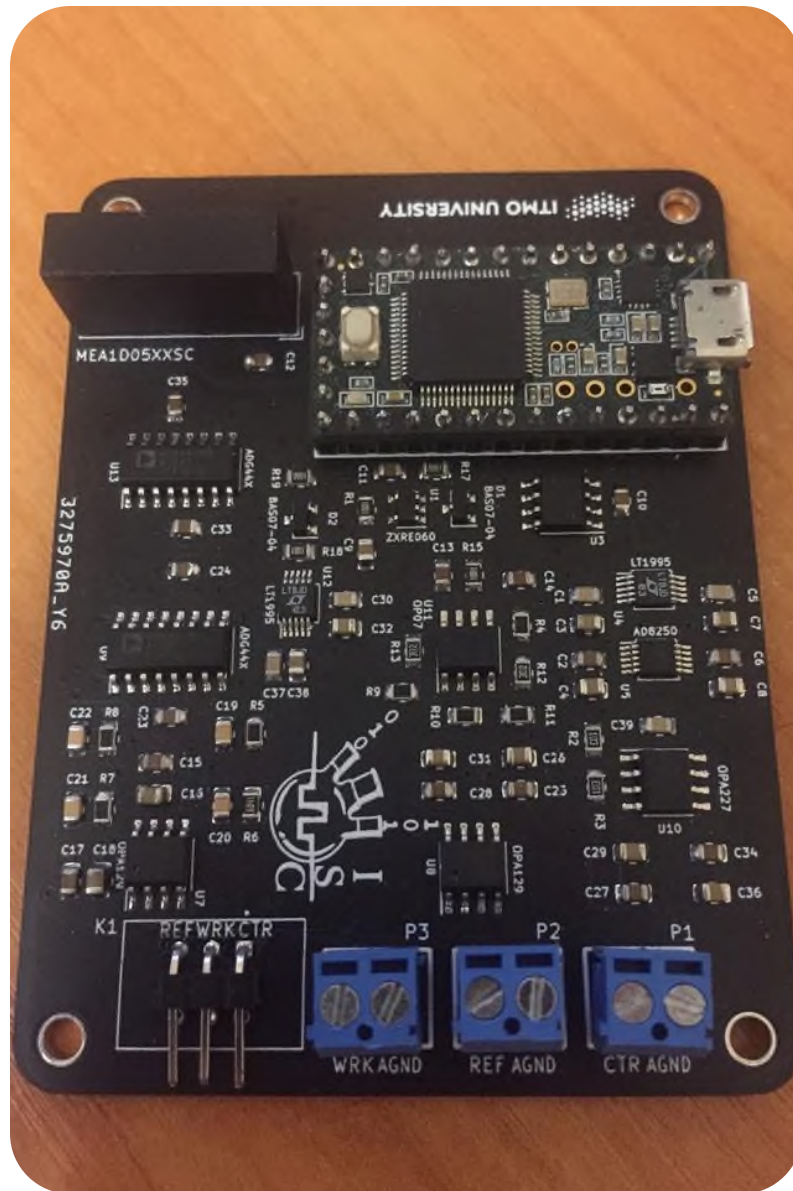
Ганькин Владимир  
Котлярова Софья

Панаёт Виктор  
Панаёт Роман

Санкт-Петербург, 2020-2021

1. Что такое потенциостат?
2. Схема потенциостата;
3. Функционал прошивки;
4. Составные части прошивки;
  - Модули, описывающие тесты, test;
  - Модули, описывающие общение с пользователем;
  - Константы, описывающие состояние электросхем и электродов потенциостата - constant and pin map;
  - Аналоговая подсистема;
  - Инициализация системы;
  - Мультиплексор;
  - Фильтр нижних частот;
  - Строковая презентация коэффициентов усиления;
  - Установление id девайса;
5. Принципы написания кода прошивки

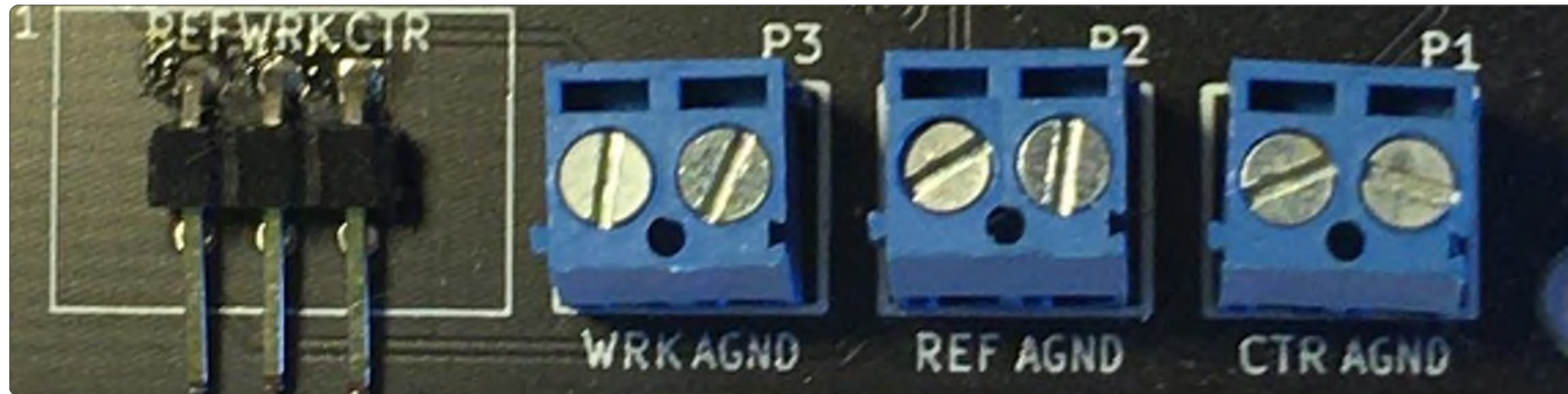
## Что такое потенциостат?



**Потенциостат** – электрическое устройство, служащее для поддержки потенциала или тока в отдельных точках независимо от их полярности.

Рисунок 1 - Внешний вид  
потенциостата





**Потенциальный электрод** - вывод или провод, с помощью которого прибор измеряет напряжение. Считается, что по этому проводу течет пренебрежимо малый ток.

### **Рабочий электрод**

- тот электрод, электрохимические процессы и явления на котором исследуются.

### **Токовый электрод**

- вывод или провод, по которому прибор передает в исследуемый объект рабочий ток

**Электрод сравнения** - нужен как точка отсчета абсолютного значения потенциала в трехэлектродной схеме, и как точка ввода в усилитель потенциостата сигнала обратной связи от электрохимической ячейки.

**Вспомогательный электрод** - второй токовый электрод в электрохимической ячейке, нужен для поляризации рабочего электрода, производство и поставка приборов для электрохимических исследований проще говоря, нужно как минимум два провода, чтобы пропустить через исследуемый объект электрический ток.

### TEENSY 3.2

- компактная платформа для разработки на базе микроконтроллера NXP MK20DX256VLH7 с вычислительным ядром ARM Cortex® M4

12-БИТНЫЙ ВЫХОД НАПРЯЖЕНИЯ С ЧЕТЫРЬМА ДОСТУПНЫМИ ДИАПАЗОНАМИ:  
 $\pm 1$ , 2, 5, 10V

МОЖЕТ БЫТЬ ЗАПРОГРАММИРОВАН ЧЕРЕЗ USB С ПОМОЩЬЮ ARDUINO IDE

УПРАВЛЕНИЕ ОСУЩЕСТВЛЯЕТСЯ ЧЕРЕЗ USB С ПОМОЩЬЮ ПРОСТЫХ СООБЩЕНИЙ JSON

16-БИТНОЕ ИЗМЕРЕНИЕ ТОКА С ЧЕТЫРЬМА ДОСТУПНЫМИ ДИАПАЗОНАМИ:  
 $\pm 1$ , 10, 100, 1000UA

**ПРОШИВКА ПОДДЕРЖИВАЕТ МНОГИЕ СТАНДАРТНЫЕ ВОЛЬТАМПЕРОМЕТРИЧЕСКИЕ МЕТОДЫ, ВКЛЮЧАЯ:**

- постоянное напряжение;
- циклическая вольтамперометрия;
- синусоидальная вольтамперометрия;
- вольтамперометрия линейной развертки;
- хроноамперометрия;
- многоступенчатый.



УНИВЕРСИТЕТ ИТМО

# Модули, описывающие тесты, test

**ps\_base\_test** – модуль для создания основы теста

Параметр	Ед. изм	Тип данных
Время молчания	мс.	uint_64
Напряжение во время молчания	В.	float

Наследуемые методы тестов:

- 1. Узнать исполнен ли тест
- 2. Узнать время исполнения теста
- 3. Сброс теста
- 4. Установление и получение значений:
  - Параметры конкретного теста;
  - Времени между выборками;
  - Имени теста;
  - Метода выборки теста (custom или generic);
- 5. Работа с JSON файлами:
  - Получить текущие параметры теста в JSON файле;
  - Установить параметры теста из JSON файла;
- 6. Мультиплексор:
  - Узнать совместим или не совместим;
  - Установить совместимость или не совместимость.



**ps\_constant\_test** – тест, где потенциал между рабочим и эталонным электродами остается постоянным. Как и другие испытания, испытание постоянным напряжением включает в себя период молчания, в течение которого выходное напряжение удерживается, и постоянное значение (напряжение молчания) в течение фиксированной продолжительности до начала испытания.

Параметр	Ед. изм	Тип данных
Время молчания	мс.	uint_64
Напряжение во время молчания	В.	float
Напряжение	В.	float
Продолжительность теста	мс.	uint_64

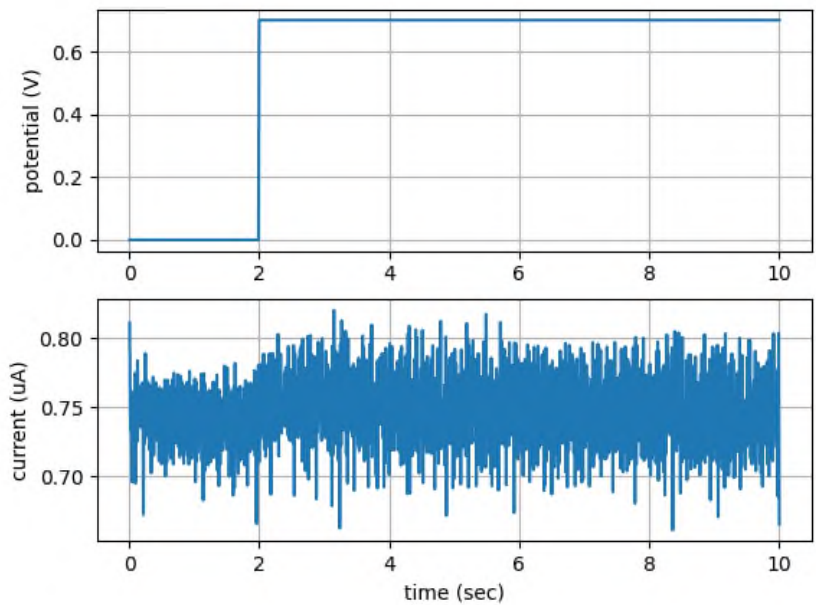


Рисунок 2 - Напряжение и сила тока от носителя во времени

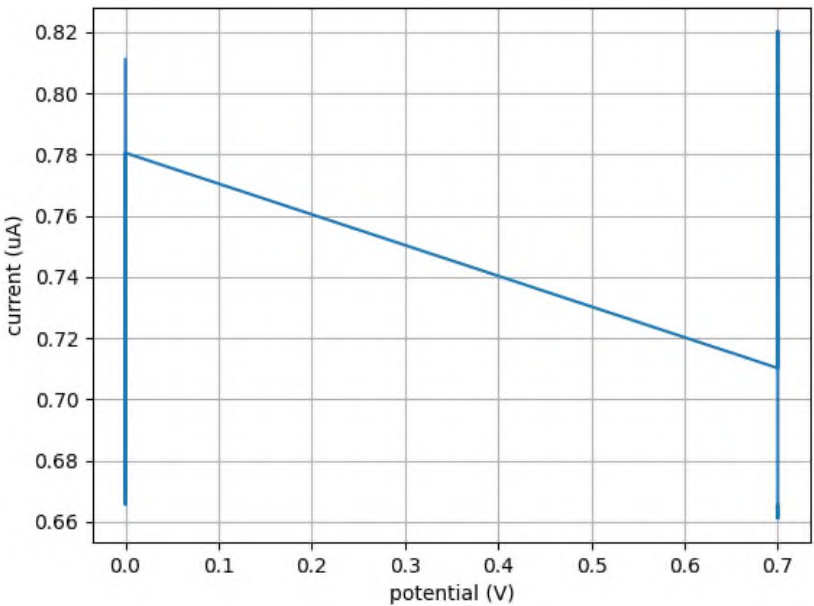


Рисунок 3 - Напряжение от носителя относительно силы тока



ps\_multistep\_test – тест, где каждое изменение напряжения задается шагом

Параметр	Ед. изм	Тип данных
Время молчания	мс	uint_64
Напряжение молчания	В	float
Шаг* - (продолжительность, напряжение)	(мс, В)	(uint_64, float)

\*макс кол-во шагов = 50

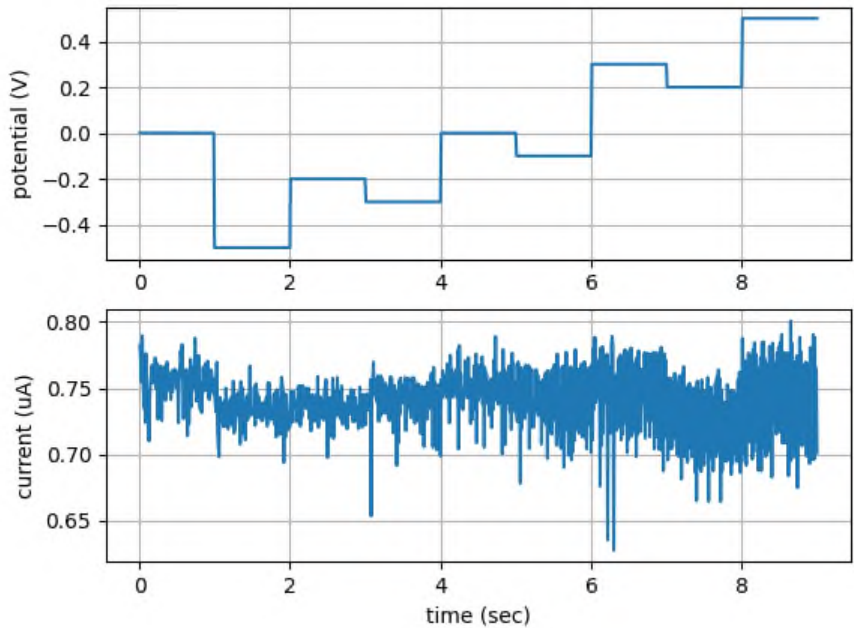


Рисунок 4 - Напряжение и сила тока от носителя во времени

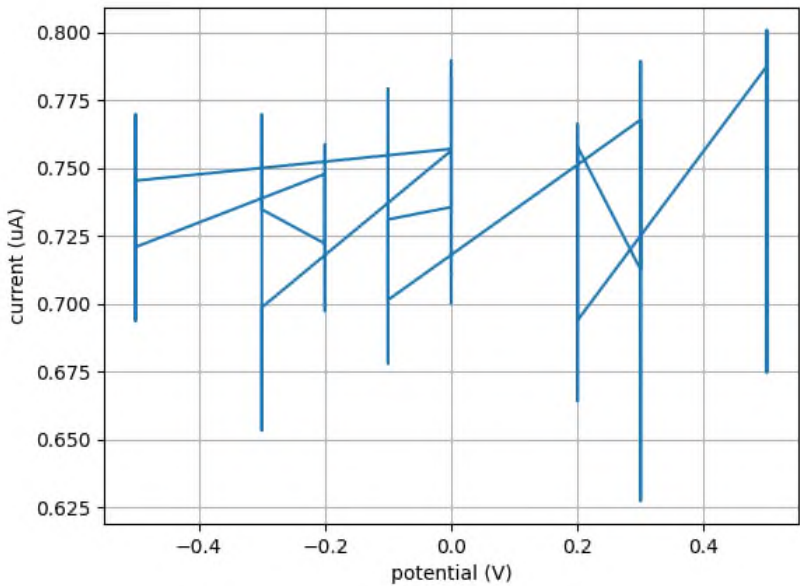


Рисунок 5 - Напряжение от носителя относительно силы тока

ps\_chronoamperometry\_test – тест, копирующий мультистеп тест, но имеющий два, и только два, шага

Параметр	Ед. изм	Тип данных
Время молчания	мс	uint_64
Напряжение молчания	В	float
Шаг* - (продолжительность, напряжение)	(мс, В)	(uint_64, float)

\*кол-во шагов = 2

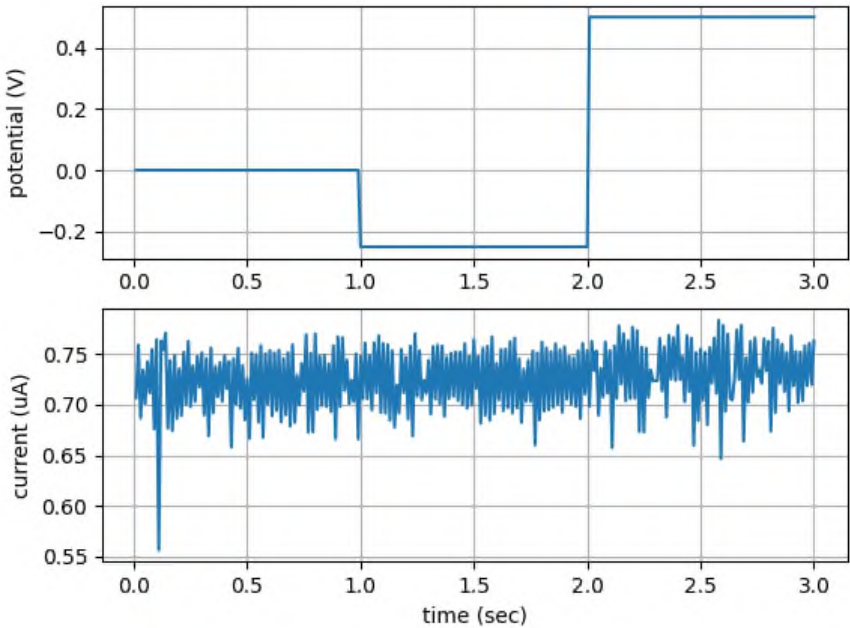


Рисунок 6 - Напряжение и сила тока от носителя во времени

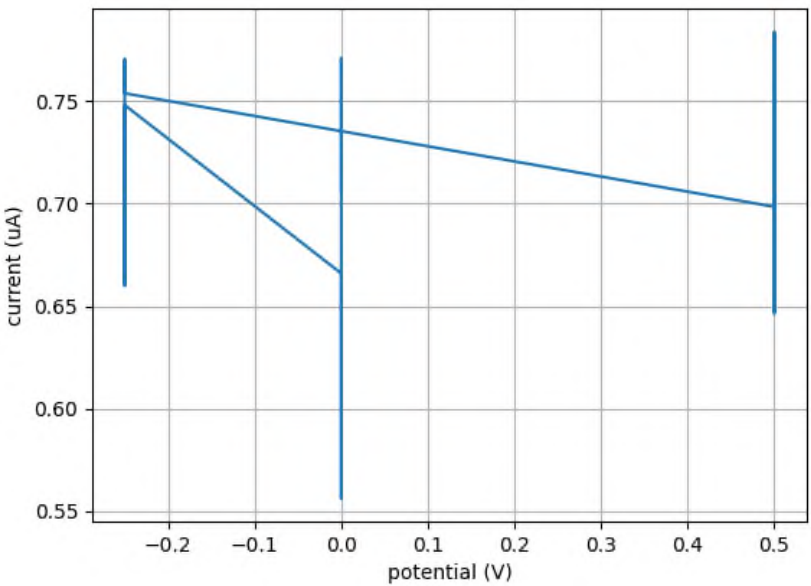


Рисунок 7 - Напряжение от носителя относительно силы тока

**ps\_periodic\_test** – модуль для создания основы периодических тестов. (Функция теста будет повторять свои значения через некоторый регулярный интервал аргумента). Здесь были добавлены новые важные параметры амплитуда, смещение по оси ординат, по оси абсцисс, сам период и кол-во циклов.

Параметр	Ед. изм	Тип данных
Время молчания	мс	uint_64
Напряжение молчания	В	float
Амплитуда	В	float
Смещение по Y	В	float
Период	мс	uint_64
Кол-во циклов	шт.	uint_32
Фазовый сдвиг (смещение по x)		float

**ps\_cyclic\_test** – наследник periodic\_test. Тест, где потенциал между рабочим и опорным электродами циклически наращивается вверх и вниз кусочно-линейным способом - в треугольной форме волны.

Параметр	Ед. изм	Тип данных
Время молчания	мс	uint_64
Напряжение молчания	В	float
Амплитуда	В	float
Смещение по у	В	float
Период	мс	uint_64
Кол-во циклов	шт.	uint_32
Фазовый сдвиг (смещение по х)		float



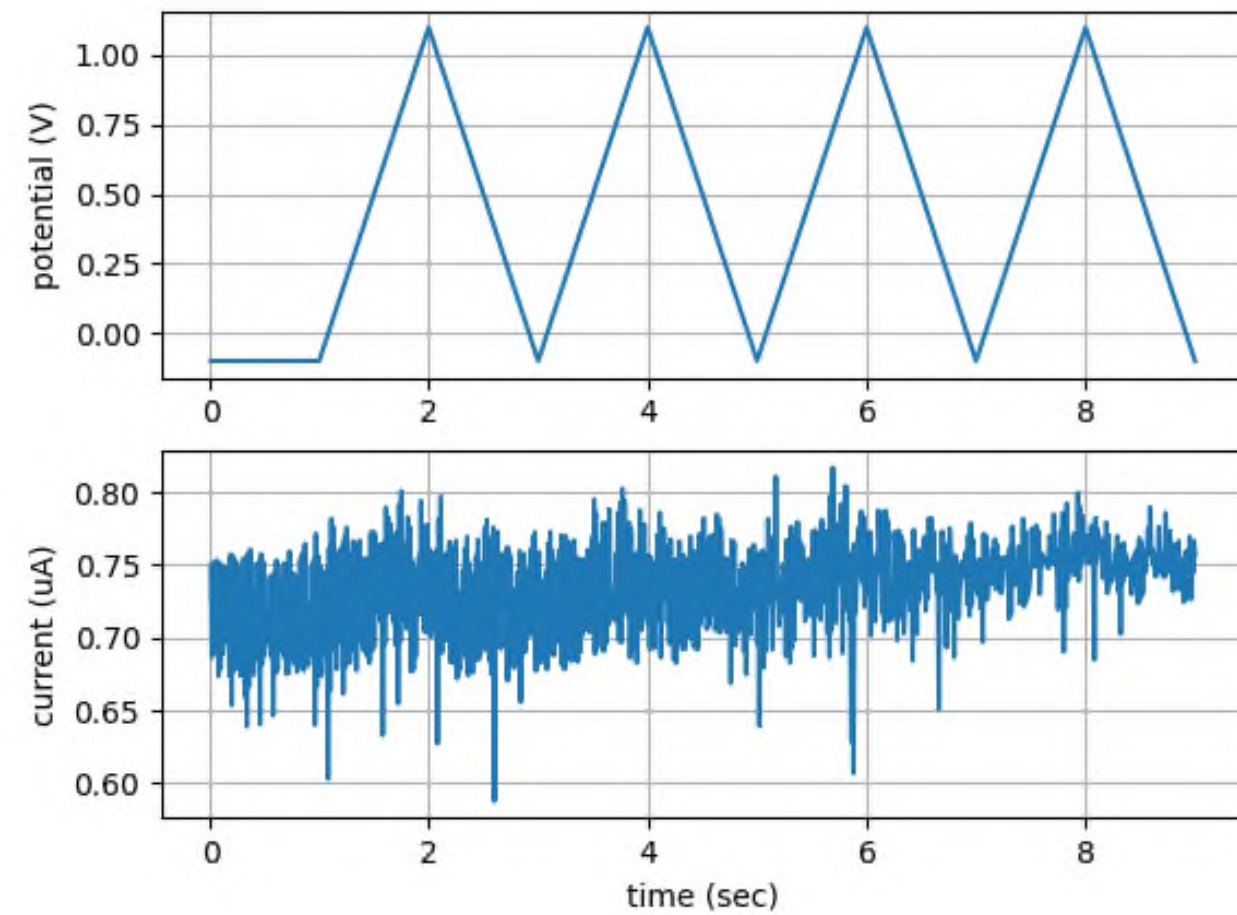


Рисунок 8 - Напряжение и сила тока от носителя во времени

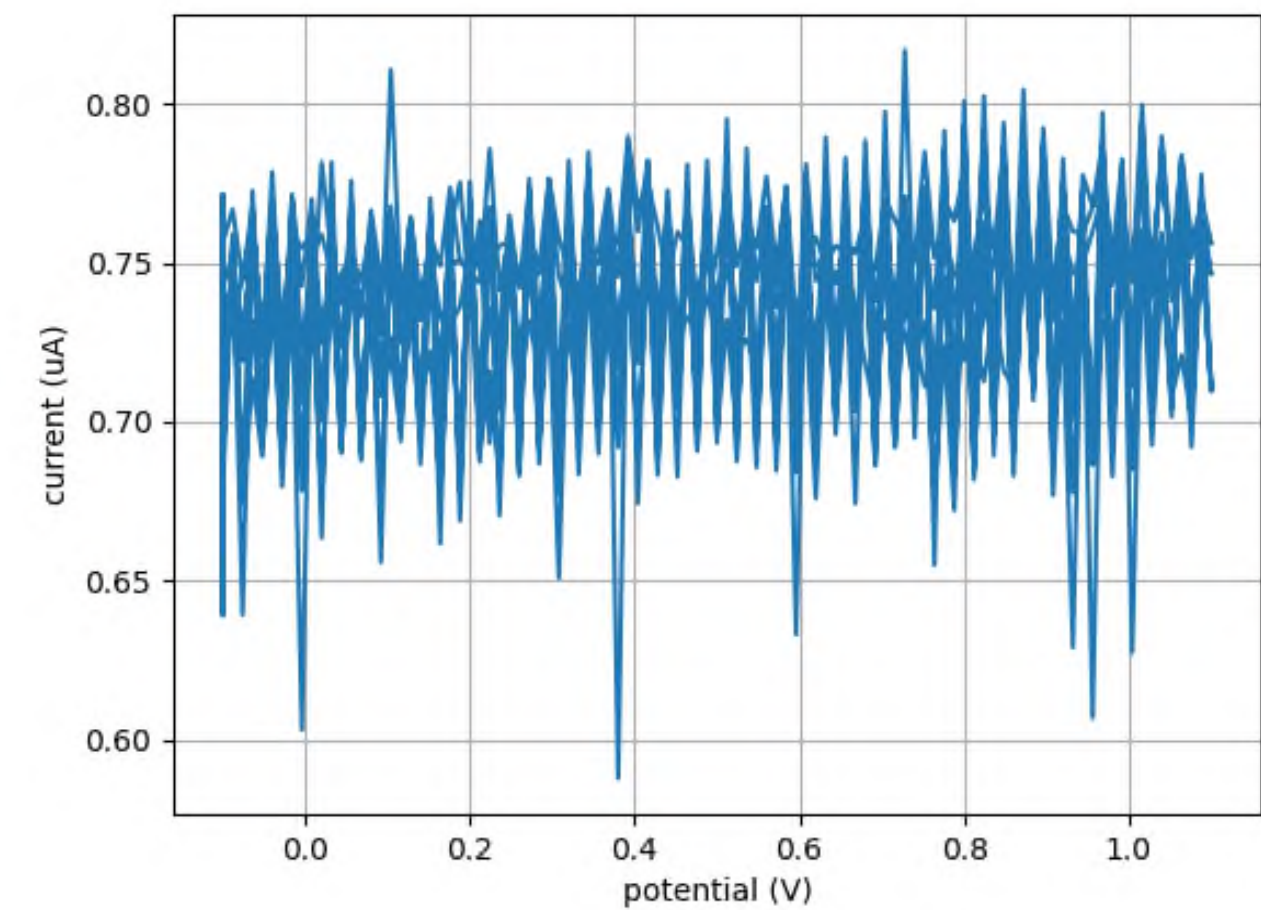


Рисунок 9 - Напряжение от носителя относительно силы тока

ps\_linearsweep\_test – тест линейной развертки

Параметр	Ед. изм	Тип данных
Стартовое значение	В.	float
Финальное значение	В.	float
продолжительность теста	Сек.	uint_64

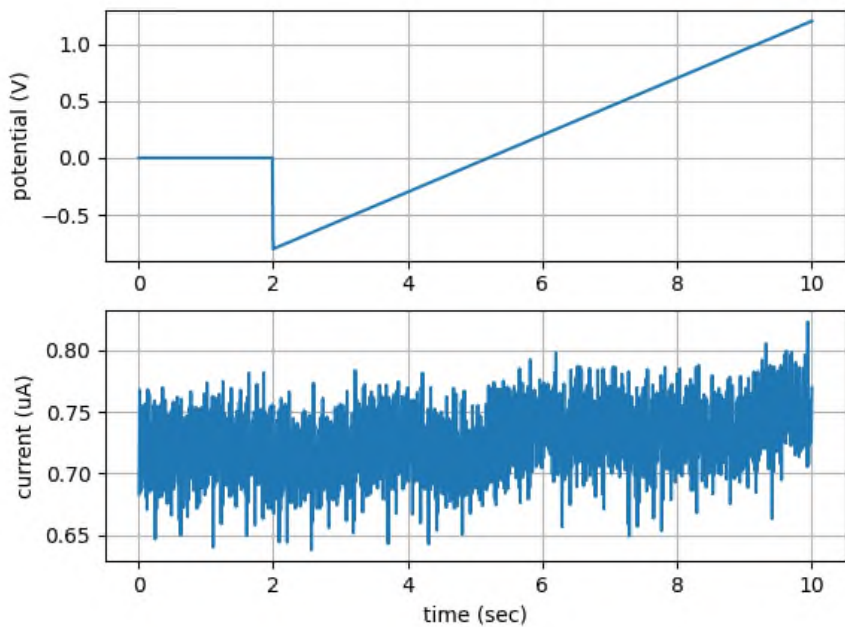


Рисунок 10 - Напряжение и сила тока от носителя во времени

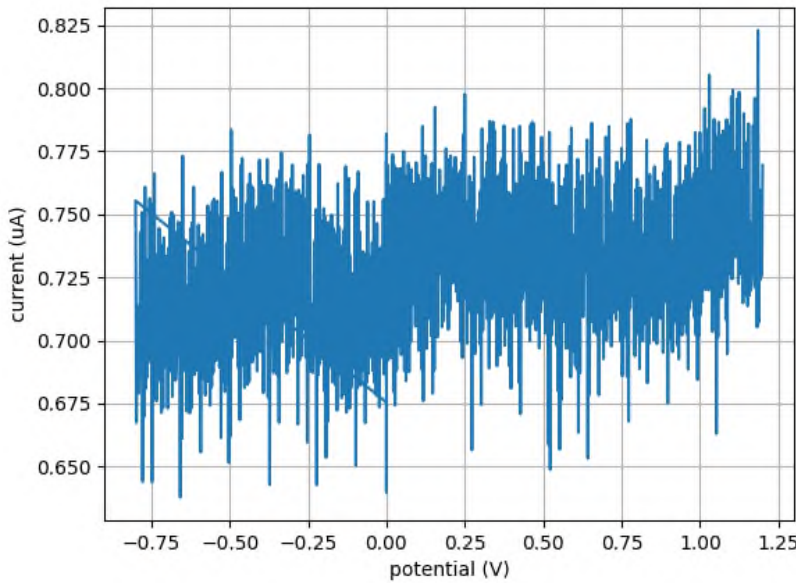


Рисунок 11 - Напряжение от носителя относительно силы тока

**ps\_sinusoid\_test** – наследник periodic\_test. Этот тест проверяет, является ли ток в потенциостате синусоидальным

Параметр	Ед. изм	Тип данных
Время молчания	мс	uint_64
Напряжение молчания	В	float
Амплитуда	В	float
Смещение по у	В	float
Период	мс	uint_64
Кол-во циклов	шт.	uint_32
Фазовый сдвиг (смещение по х)	В	float



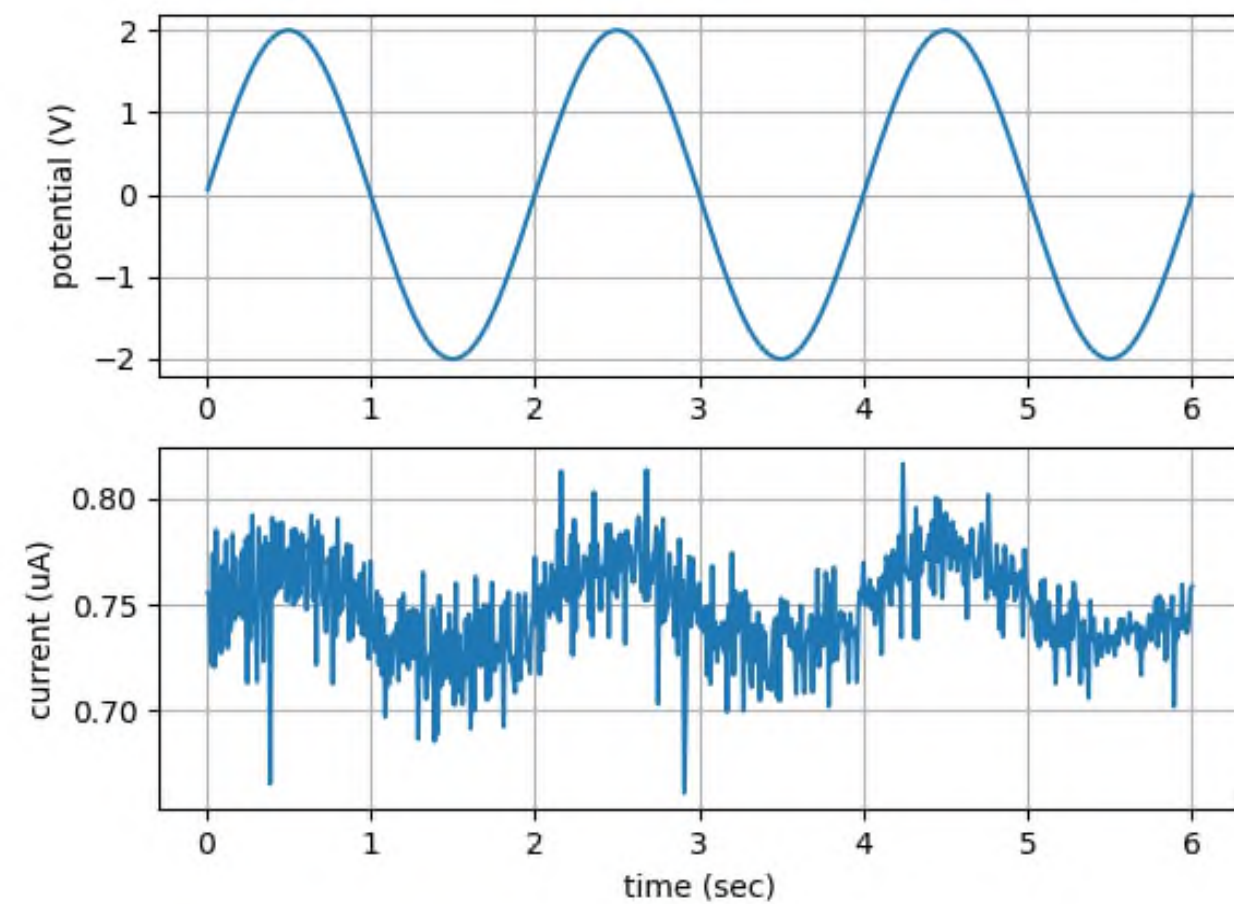


Рисунок 12 - Напряжение и сила тока от носителя во времени

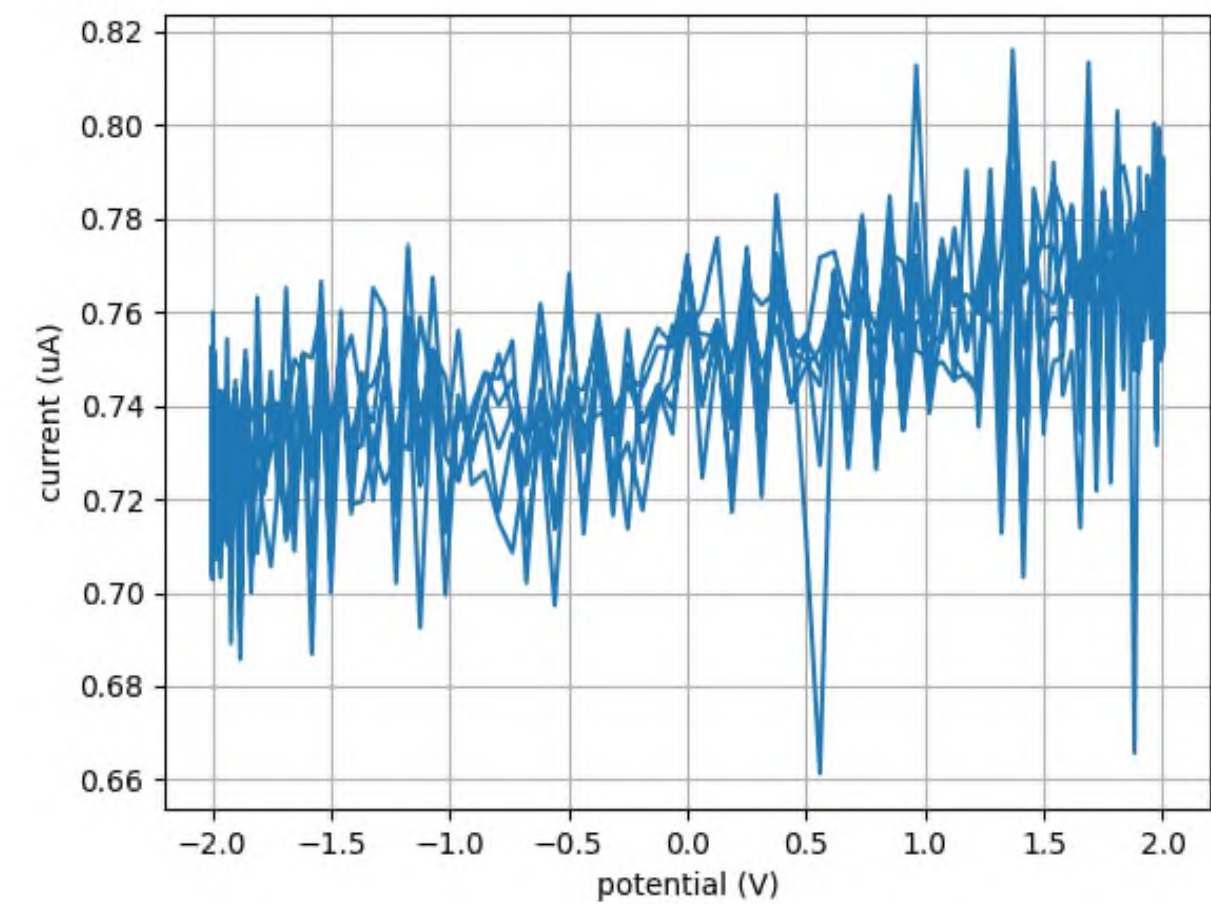


Рисунок 13 - Напряжение от носителя относительно силы тока



Тест прямоугольной вольтамперометрии. В эксперименте по вольтамперометрии прямоугольной формы измеряется ток на (обычно неподвижном) рабочем электроде, в то время как потенциал между рабочим электродом и электродом сравнения передается вперед и назад с постоянной частотой. Потенциальную форму волны можно рассматривать как наложение регулярной прямоугольной волны на нижележащую лестницу.

Поясняющее изображение:



Параметр	Ед. изм	Тип данных
Стартовое значение	В.	float
Финальное значение	В.	float
Шаг	(мс, В)	float
Амплитуда	В.	float
Окно	%	float
Продолжительность теста	Сек.	uint_64

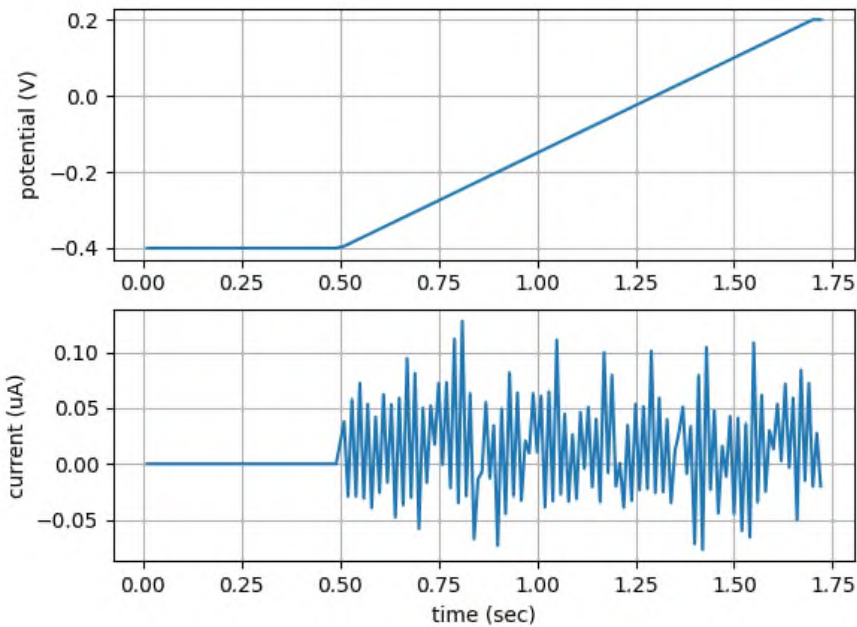


Рисунок 14 - Напряжение и сила тока от носителя времени

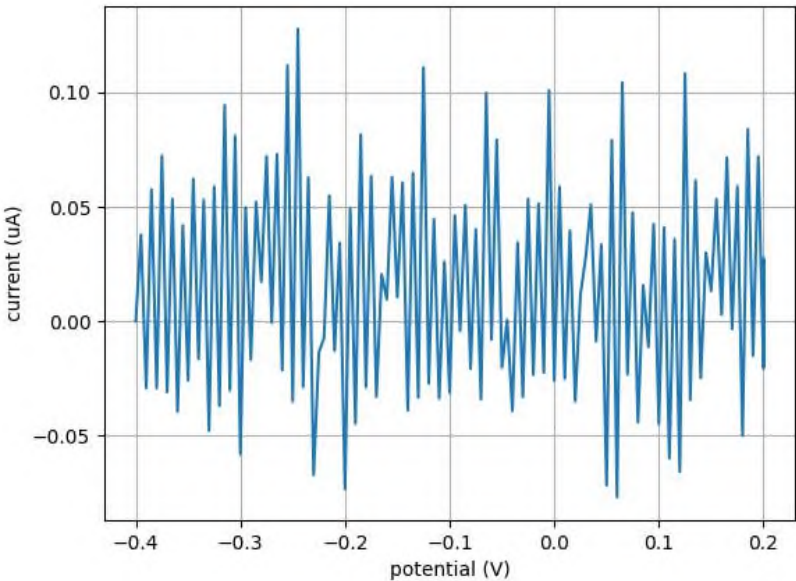


Рисунок 15 - Напряжение от носителя силы тока



УНИВЕРСИТЕТ ИТМО

# Модули, описывающие общение с пользователем

**ps\_command\_table** – реализация командной строки

### Методы:

- Table – Массив с зарегистрированными методами;
- Cleartable – очистить табло;
- Clear – очистить табло и клиента, создавшего табло;
- Size – размер табло;
- Maxsize – максимальный размер табло (установлен в коде);
- Registermethod – зарегистрировать метод в табло;
- Apply – получить результат команды. Прописаны следующие исключения: клиент не найден, команда не найдена.

**ps\_message\_parser** – класс Message parser переводит сообщение типа String в JSON объект;

**ps\_message\_sender** – класс Message sender отвечает за запись данных в JSON объект

**ps\_message\_receiver** – класс Message receiver отвечает за принятие сообщения. Состоит из ранее упомянутого циркулярного буфера, флага заполнения и счетчика принятых сообщений без переполнения и счётчика всех принятых сообщений

### Функционал:

- Считать и обнулять кол-во принятых сообщений;
- Посимвольно принимать строку;
- Копировать данные из буфера в строку.





УНИВЕРСИТЕТ ИТМО

# Константы, описывающие состояние электросхем и электродов потенциостата

### В целом константы можно разделить на 14 видов:

1. Версии - прошивки, вольтаж, железа у-ва;
2. Параметры командной строки - максимальное значение команд;
3. Параметры буферов (ударение поставьте сами) - Data, Serial, JSON Test Data, JSON Message Data;
4. Адрес ячейки памяти, где располагается id у-ва;
5. Параметры Serial, т. е. монитора порта - бодрейт (к-во бит в секунду);
6. Message ключи, т. е. ключи, которые компилятор будет ожидать встретить в наших JSON и из которых сам будет составлять JSON.
7. Названия команд, Перечислять отдельно смысла нет, у вас только уши завянут от моего иняза.
8. Интервалы для исходящего вольтаж;
9. Интервалы для электрода сравнения;
10. Интервалы для входящей силы тока; Про интервалы упомянуто в gains.
11. Таймеры и периоды в микрос. Время на выборку (SamplePeriod) - min. default, max. Дефолтное время на тест;
12. Параметры фильтра - частота среза;
13. Параметры тестов - макс. кол-во, макс кол-во шагов для мультитепа теста;
14. Параметры мультиплексора.



УНИВЕРСИТЕТ ИТМО

# Модули, описывающие амплитуды изменения электрических переменных

**Range** - класс-родитель. Задаёт атрибуты и методы обязательные для наследников, такие как - имя, минимальное и максимальное значение, коэф. усиления и максимальное целочисленное значение.

### Методы, позволяют узнать атрибуты класса:

- `valueToInt()` - приводит передаваемое значение `volt` к `int` типу + возвращает значение ограниченное  $0 \leq \text{value} \leq \text{maxInt\_}$ :
  - `IntType value = IntType(float(maxInt_)/(maxValue_ - minValue_)*(maxValue_ - volt));`
  - `intToValue()` - обратная по отношению к `valueToInt` операция.
- `Curr_range` - переопределяет атрибут коэф. усиления на интервал изменения силы тока. Возможные варианты в `gains`.
- `Volt_range` - переопределяет атрибут коэф. усиления на интервал изменения напряжения. Возможные варианты в `gains`. У этого класса существует два наследника для DAC и ADC (преобразователи цифры в аналог, аналога в цифру)



### Один из важнейших модулей

Подсистема для работы с вольтажом электродов, усилителем тока и напряжения и отдельно для работы с вольтажом электрода сравнения (только для второй версии железа).

Под железо второй версии расширен функционал. Под каждое железо своя настройка пинов по умолчанию. Настройка по умолчанию происходит при первой инициализации.

1

Настраиваем состояние системы по умолчанию – тестов нет, таймер не запущен, значения фильтра нижних частот тоже по умолчанию (частота среза – 200), период на выборку – 0.1с.

2

Далее в командном табло регистрируются методы по формуле обращения «command <команда>» и устанавливается текущий клиент. Их много, позволяют настраивать все элементы описанные выше. Кол-во методов и, след-но, функционал зависит от версии железа.

3

После включается аналоговая система с выходящим значением напряжения 0; Обнуляется циркулярный буфер приемщика сообщений.

4

Если версия устройства вторая, то производится автоматическая настройка электродов на вывод; Устанавливать состояние «присоединен» надо вручную в любом случае.

- Можно запустить или остановить тест в безопасном режиме, т.к. входящие данные теста и если они неправильного синтаксиса, то тест не запустится.
- Если мультиплексор включён, то тест должен его поддерживать, иначе теста не будет.
- Если на мультиплексоре заняты рабочие электроды, то тест также не запустится.
- При успешном прохождении проверок происходит подключение электродов к мультиплексору.

### Также реализованы такие вещи:

- Буфер для хранения выборок;
- Вывод системных сообщений;
- Установить колбэк таймера;
- Возможность изменения значений теста во время его исполнения.

В файле **ps\_muxplexer** реализован мультиплексор

**Мультиплéксор** — устройство, имеющее несколько сигнальных входов, один или более управляющих входов и один выход. Мультиплексор позволяет передавать сигнал с одного из нескольких входов на один выход; при этом выбор желаемого входа осуществляется подачей соответствующей комбинации управляющих сигналов.

Фильтр имеет пять порядков от 1 до 5. Чем выше порядок, тем сильнее фильтр. Соответственно порядок входит в параметры фильтра.

Ещё есть частота среза и значение, которое надо отфильтровать. Устанавливать можно как всё сразу, так и каждое по отдельности.

$$f = \frac{1}{2\pi RC} = \frac{1}{2\pi \tau}$$

Фильтр нижних частот работает так: он пропускает частотный спектр сигнала ниже определенной частоты, называемой частотой среза, и подавляет сигнал выше этой частоты.

В коде из этой формулы выводится RC, который нужен для расчета фильтрующего значения - alpha. alpha - дифференциал от rc по времени.



**ps\_gains** – методы для строковой презентации коэффициентов усиления. Используется в аналоговой подсистеме.

Чем выше коэффициент усиления, тем шире интервалы изменения напряжения и силы тока. У каждого параметра свой коэффициент.

Напряжения от 1 до 4 (+-1, 2, 4, 8, 10 V) или от 1 до 3 (+- 1, 2, 5, 10) в зависимости от вольтажа устройства.

У силы тока от 1 до 3 ( +-10, 100, 1000 миллиАмпер).

`ps_device_id_eeprom` – сохранение в EEPROM памяти ID устройства

**EEPROM** – память, к которой есть полный доступ из выполняющейся программы, т.е. можем во время выполнения читать и писать туда данные, и эти данные не сбрасываются при перезагрузке потенциостата и вообще микроустройства.

Для записи надо выбрать адрес ячейки памяти и предоставить JSON файл с ключом “`deviceId`” и типом данных `Int`.

Специальным методом можно и получить JSON файл с ID.

**sample:** объявляет класс `sample` с набором переменных (`uint64_t t`, `float volt`, `float curr`, `uint8_t chan`)

**return\_status:** сохраняет статус записывая в переменную `message` статусы тестов через запятую

**time\_utils:** методы для преобразования секунд в миллисекунды и обратно

**voltammetry:** создает экземпляры классов всех тестов, описывает методы для работы со всем параметрами тестов через json файлы



