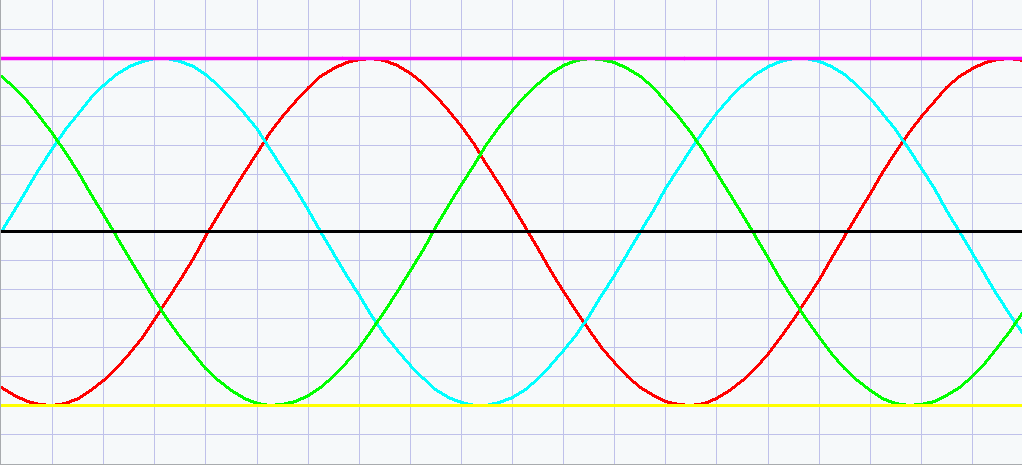
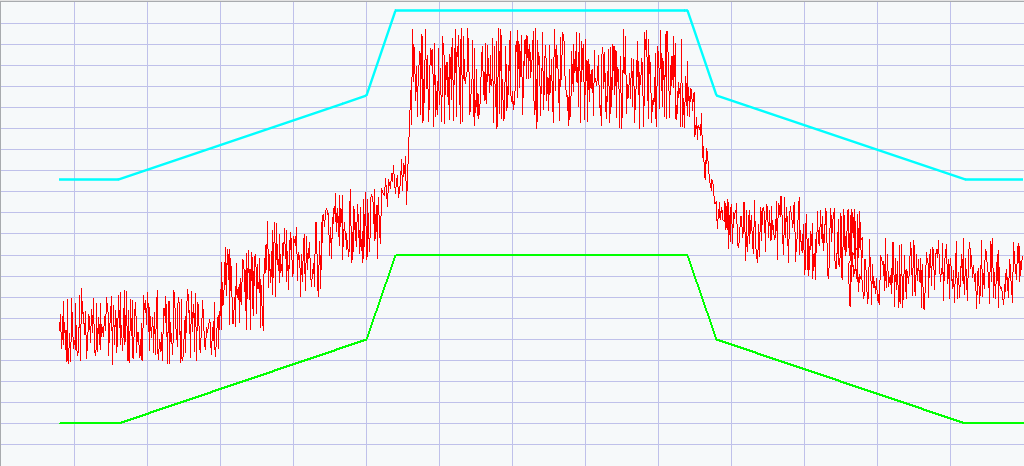
Презентация программы MOKO Graph





|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | |  | | |  | | |
| **Iteration** | x | y | **Iteration** | x | y | **Iteration** | x | y |
|  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | |  | | |  | | |
| **Iteration** | x | y | **Iteration** | x | y | **Iteration** | x | y |
|  |  |  |  |  |  |  |  |  |

Screenshots:

Скриншот №1

Скриншот №2

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | |  | | |  | | |
| **Iteration** | x | y | **Iteration** | x | y | **Iteration** | x | y |
|  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | |  | | |  | | |
| **Iteration** | x | y | **Iteration** | x | y | **Iteration** | x | y |
|  |  |  |  |  |  |  |  |  |

Screenshots:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | |  | | |
| **Iteration** | x | y | **Iteration** | x | y |
|  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | |  | | |
| **Iteration** | x | y | **Iteration** | x | y |
|  |  |  |  |  |  |

Screenshots:

**Документация**

Этот раздел посвящен функциональным возможностям плагина MOKO Graph и взаимодействию с ним через API.

MOKO Graph разработан для обработки численных данных и наглядной визуализации полученных данных. Плагин имеет главное окно и вспомогательное окно с минималистичным дизайном, которые фокусируют внимание пользователя на графическое представление данных. В нижней части программы расположены кнопки для использования основных возможностей программы и индикация состояния плагина.

Пользователь имеет возможность посредством команд, UI или окна настроек добавлять координаты для графика, совершать различные операции над ними, делать скриншоты.

Также API плагина позволяет получать определенную информацию для последующей обработки в скрипте или отчёте. Для использования API MOKO Graph используется библиотека graph.py.

– Добавление линии

*graph.add\_line(name,arroy,arrox,linewidth,color,visible)*

:param str name: Имя линии

:param str arroy: Координаты линии по оси ординат

:param str arrox: Координаты линии по оси абсцисс

:param str linewidth: Толщина линии

:param str color: Цвет линии

:param str visible: Видимость линии

:return None

Пример команды:

*name = "Plot 1"*

*arrox = [0, 1, 2, 3, 4]*

*arroy = [0, 1, 2, 3, 4]*

*linewidth = 1*

*color = "00FFFF"*

*visible = "Yes"*

*graph.add\_line(name, arroy, arrox, linewidth, color, visible)*

После выполнения данной команды в дропдауне появилась линия Plot 1. В окне Settings можно увидеть параметры данной линии. Однако линия не отражается на графике.

– Отображение данных на графике

*graph.write\_graph()*

:return None

Результатом выполнения данной команды является появление линии Plot 1 на графике.

– Изменение параметров уже добавленной линии

*graph.change\_line(numline,name,\*arroy,\*arrox,linewidth,color,visible)*

:param int numline: Номер линии

:param str name: Имя линии

:param int \*arroy: Координаты линии по оси ординат

:param int \*arrox: Координаты линии по оси абсцисс

:param int linewidth: Толщина линии

:param str color: Цвет линии

:param str visible: Видимость линии

:return None

Пример команды:

*name = "Plot 1"*

*numline = 1*

*arrox = [0, 1, 2, 3, 4]*

*arroy = [6, 0, 1, 2, 3]*

*linewidth = 5*

*color = "00FFFF"*

*visible = "Yes"*

*graph.change\_line(numline, name, arroy, arrox, linewidth, color, visible)*

После выполнения данной команды параметры линии Plot 1 изменятся.

– Удаление линии

*graph.delete\_graph(\*numline)*

:param int \*numline: Номер линии или номера нескольких линий

Номера нескольких линий передаются списком. Также, функция принимает единственное строковое значение “All”, которое приведёт к удалению всех линий.

Пример команды:

*numline = [0, 1, 2, 3]*

*graph.delete\_line(numline)*

*graph.delete\_line(‘All’)*