

# Модуль спектрального анализа

## Назначение модуля

Модуль предназначен для обработки данных акселерометров или других датчиков за определенные интервалы времени для построения спектра колебаний и нахождения в полуавтоматическом режиме основных форм колебания и вычисления их параметров их пиков. Различает одиночные и двойные пики.

## Технические требования

Операционная система: Ubuntu 20/22, Linux OS 64 битная. PHP версии 7.3, модуль работы с PostgreSQL и ClickHouse.

## Требования к вычислительным ресурсам

1 ядро поддерживающее 64 битные вычисления, 500 Мбайт на жестком диске для временных данных.

## Описание работы

Структурно работу модуля можно поделить на две части: сбор, подготовку данных и отправку готовых данных, которую выполняет PHP скрипт «index.php». Обработка данных, формирование спектров и поиск основных форм колебаний выполняет программа RealTimeSpectrum. Модуль может быть вызван или из системы или из консоли в контейнере docker «php». Для запуска работы модуля запускается PHP скрипт «index.php». Блок-схема работы модуля представлена на рисунке.



Вначале работы скрипта «index.php» загружаются конфигурационные данные модуля из файла «config.php», содержащего данные к подключению к базам данных, он находится на директории выше модуля и файл конфигурации модуля «config.php», лежащего в директории модуля. Далее выбираются шины типа «an-d3» и «com-an-d3», шины для подключения акселерометров. В выбранных шинах выбираются устройства типа «and\_3». Далее просматриваются выбранные устройства и создается массив каналов этих устройств, у которых код равен «0.x» — ось X, «0.y» — ось Y, «0.z» — ось Z, «0.a» — модуль ускорения, обозначается через A.

serif;;#000000;;inherit>После формирования массива, обрабатываем оси. Проверяем, или есть настройка для каждого канала в конфигурационном файле модуля «config.php». Если данных настройки нет, по</font> <font inherit/inherit;;#000000;;inherit>обработка данной оси пропускается.</font> <font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>Если канал сконфигурирован, то запрашивается данные по этому каналу в интервале «\$time\_interval» до времени запрошенного в начале запуска скрипта, переменная «\$timestamp».</font> <font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>Далее проверяется и создаются папки для выходных файлов программы RealTimeSpectrum «/var/www/html/public/Spectr».</font> <font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>Данные программой RealTimeSpectrum обрабатываются в двух вариантах. Первый — для полноценной спектральной мощности, данные записываются в директорию «/data», второй — для цветовой спектрограммы, данные записываются в директорию «/data\_partial». Для спектрограммы спектральная мощность строится с меньшим частотным разрешением для увеличения быстродействия. Далее данные записываются в директории с именем равным id устройства, далее id канала, после год, и последняя директория месяц.</font> <font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>Для примера формирования пути, если id устройства 67, id канала 433, 2022 год и месяц сентябрь 9-й месяц, тогда путь будет выглядеть для первого случая «/var/www/html/public/Spectr/data/67/433/2022/09», для второго случая «/var/www/html/public/Spectr/data\_partial/67/433/2022/09».</font> <font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>Далее формируются конфигурационные файлы «ofs\_options.ini» и «Config.path» для работы программы RealTimeSpectrum и записываются полученные данные в бинарном виде для обработки. Далее данные обрабатываются программой RealTimeSpectrum и полученные данные размещаются как указано выше, и полученные данные по пикам основных форм колебаний отправляются в базу данных. Обработка, формирование файлов и отправка данных выполняются для двух случаев подряд, так как для них используются одни и те же данные и должны использоваться одни и те же данные, что бы результаты были одинаковыми.</font> <font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>Рисунок. Блок-схема модуля.</font> <font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>Основные файлы модуля</font> <font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>- config.php - файл настроек программы обработки спектров, должен быть в gitignore. config\_example.php - пример файла настроек программы обработки спектров</font> <font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>-</font> <font inherit/inherit;;#000000;;inherit>Config.path - динамический файл настроек датчиков для работы программы обработки спектров RealTimeSpectrum, формируется в процессе работы модуля, должен быть в gitignore.</font> <font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>- ofs\_options.ini - динамический файл общих настроек работы программы обработки спектров</font> <font inherit/inherit;;#000000;;inherit>RealTimeSpectrum, формируется в процессе работы модуля, должен быть в gitignore.</font> <font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>- .gitignore - содержит список файлов и директорий, которые должны игнорироваться и не попадать в индекс</font> <font inherit/inherit;;#000000;;inherit>git</font> <font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>-</font> <font inherit/inherit;;#000000;;inherit>RealTimeSpectrum - программа для обработки данных датчиков и формирования спектров и поиска пиков

основных форм колебания.

index.php - PHP скрипт вызываемый для обработки спектров. Непосредственно в нем берутся данные из базы данных, подготавливаются для обработки и вызывается программа RealTimeSpectrum. Скрипт формируют пути, куда записывается спектр сформированный RealTimeSpectrum и отправляет результаты определения частот и декрементов колебаний в базу данных.

data\_processing.sh - bash скрипт для запуска модуля через cron в контейнере.

source - директория с временными рабочими файлами модуля

index\_check.php - вспомогательный PHP скрипт, позволяет проверить сколько в системе установлено устройств типа АНД-3

data\_processing\_test.sh - вспомогательный bash скрипт для запуска index\_check.php в контейнере

## RealTimeSpectrum

Программа RealTimeSpectrum обрабатывает данные сформированные акселерометрами и строит спектры колебания. Физический принцип, на котором основана работа программы, состоит в получении исходных данных ускорений с акселерометров от сборщика данных. Далее с помощью преобразования Фурье формируется спектральная мощность колебаний. Метод определения основных форм колебаний и их декрементов является полуавтоматическим. Для его работы необходимо задать интервал частот, в котором необходимо искать пик колебаний, и так же указать ищется ли одинарный или двойной пик. Далее специальным методом с помощью метода наименьших квадратов аппроксимируется пик или два пика в зависимости от настроек, и по данным аппроксимации вычисляются частота максимума пика и по ширине на полувысоте аппроксимированного пика — декремент колебаний.

## Настройка модуля

Настройка модуля производится в файле config.php, все параметры подробно описаны в config\_example.php, его содержание приведено ниже

`$api_token = 'A8WQ0N4zYPQvxbBMvPVz2ZnhGGyfgjhbh';` *api токен для доступа по rest*

`$time_interval = 40960;` *Интервал в секундах от текущего времени, за который запрашиваются данные с clickhouse*

`$length_zap_full = 8192;` *Длина одной реализации для построения одного спектра*

`$length_zap_spectrograms = 1024;` *Длина одной реализации для построения цветовой спектрограммы, так как вопрос ещё в производительности то длина уменьшена для быстрого рендера картинки*

`serif;;#000000;;inherit>$number_records_full=2;` Количество реализаций, по которым происходит усреднение для построения одного спектра `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$number_records_spectrograms=16;` Количество реализаций, по которым происходит усреднение для построения цветовой спектрограммы `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$basic_waveforms=5;` Количество основных форм колебания, которые обрабатываются скриптом PHP и отправляются в базу данных `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>` если данных на *i*-ю форму нет после работы программы построения спектра, то она заполняется нулями `</font>` Настройка каналов для обработки `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>` В массив `$device_code` добавляется код устройства, к которому принадлежит канал `</font>` в `$min_frequency_all` и `<font inherit/inherit;;#000000;;inherit>` в `$max_frequency_all` `</font>` `<font inherit/inherit;;#000000;;inherit>` ключем `</font>` `<font inherit/inherit;;#000000;;inherit>` равным `</font>` `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>` коду устройства указывается минимальная и максимальная граница интервалов, в которых нужно искать пики основных форм колебания. В `$double_frequency_all` `<font inherit/inherit;;#000000;;inherit>` указывается количество пиков для поиска `</font>` `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$device_code=array('433', '434', '501', '502');` `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$min_frequency_all['433']='0.3 0.6 1.3';` `</font>` `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$max_frequency_all['433']='0.35 0.8 1.6';` `</font>` `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$double_frequency_all['433']='1 1 1';` `</font>` `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$min_frequency_all['434']='0.3 0.6 1.3';` `</font>` `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$max_frequency_all['434']='0.35 0.8 1.6';` `</font>` `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$double_frequency_all['434']='1 1 1';` `</font>` `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$min_frequency_all['501']='0.3 0.6 1.3';` `</font>` `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$max_frequency_all['501']='0.35 0.8 1.6';` `</font>` `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$double_frequency_all['501']='1 1 1';` `</font>` `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$min_frequency_all['502']='0.3 0.6 1.3';` `</font>` `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$max_frequency_all['502']='0.35 0.8 1.6';` `</font>` `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$double_frequency_all['502']='1 1 1';` `</font>` Более подробное описание что есть что `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>` `$device_code=array('433');` `$device_code` - набор кодов устройств, где устройствово означает одну ось датчика, например 001X. Т.е. устройство с кодом 433 это датчик с номером 001, ось X. `</font>` Для устройства 433 указываем интервалы, в которых нужно искать пики колебаний. Интервалы: `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>$min_frequency_all['433']='0.3 0.6 1.3';` `</font>` `$max_frequency_all['433']='0.35 0.8 1.6';` `<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>` Они означают 3 интервала, в которых искать пики. Интервал 1 0.3-0.35; интервал 2 0.6-0.8; интервал 3 1.3-1.6; `</font>` И количество пиков, которые необходимо искать `$double_frequency_all['433']='1 1 1';` `<font 14px/Arial,Helvetica,sans-`

serif;;#000000;;inherit> Здесь указаны три интервала, в них искать по одному пику. Варианты значений 1 либо 2.

Файл для работы программы обработки спектров, общие настройки

serif;;#000000;;inherit>\$file\_options='number\_records = '.\$number\_records.' Количество реализаций, по которым будет строиться

аппроксимация

спектра и находится параметры колебаний сооружения

length\_zap = '.\$length\_zap.' Длина одной реализации в количестве отсчетов, должно быть кратно  $2^n$ , 2 в степени n, где n целое число

number\_of\_records = 30 Сколько в одном файле реализаций

sampling\_time = 0.1 Время дискретизации датчиков в секундах

time = «22\_01\_10\_23\_59\_43\_021» Время, с которого считывать данные, данные до этой временной ометки будут игнорироваться, если указать 00\_00\_00\_00\_00\_00\_000, то будут обрабатываться все данные

time\_start = «2022\_02\_12\_07\_21\_00\_729» Время, с которого считывать данные, данные до этой временной ометки будут игнорироваться, если указать 00\_00\_00\_00\_00\_00\_000, то будут обрабатываться все данные

time\_end = «'»\$year«'»\_ckgedit\_QUOT«'»\$month«'»\_ckgedit\_QUOT«'»\$day«'»\_ckgedit\_QUOT«'»\$hour«'»\_ckgedit\_QUOT«'»\$minute«'»\_ckgedit\_QUOT«'»\$second«'»\_000

big\_files = 0 Параметр, отвечающий за то, что мы берем большие файлы данных за сутки. 1 - мы берем, другое значение, не берем

debug\_mode = 0 Режим отладки, выводятся промежуточные данные обработки, по умолчанию отключено

processe\_all\_files = 0 Обработка всех файлов сразу, по умолчанию обрабатывается начиная с самого последнего

rusian = 1 Включение русского языка, по умолчанию включен

file\_size\_for\_processing = 1 Устанавливаем ограничение обработки файлов за один запуск

time\_shift=0.245733333333333 Сдвиг по времени в секундах Пропуск времени между соседними спектрами при обработке всех файлов сразу, величина в секундах.

detector\_option\_file = «Config.path» Имя файла с настройками для датчиков.

fast\_processing = 0

file\_format=4 Формат

файлов

данных

monitoring\_horizont=1'» Файл для работы программы обработки спектров, настройка датчиков

\$file\_options\_conf='{

limitation\_standart\_deviation=0 Среднеквадратичное

отклонение

`serif;;#000000;;inherit>limitation_ejection=50</font> Выброс в процентах от диапазона  
<font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>measuring_range=3600 Тип</font>  
<font inherit/inherit;;#000000;;inherit>датчика (диапазон) AN3600, AN7200</font> <font  
14px/Arial,Helvetica,sans-serif;;#000000;;inherit>axis='.$detector_name.'</font> Номер  
датчика и его оси <font 14px/Arial,Helvetica,sans-  
serif;;#000000;;inherit>namefile=»'.dirname(</font> FILE__).'./source/« Относительный  
путь к данным <font 14px/Arial,Helvetica,sans-  
serif;;#000000;;inherit>savadata=»'.$dir.'/</font> Относительный путь, куда сохранять  
результаты моделирования <font 14px/Arial,Helvetica,sans-  
serif;;#000000;;inherit>datatemperature=»'./« Относительный путь, откуда читать  
данные по температуре</font> <font 14px/Arial,Helvetica,sans-  
serif;;#000000;;inherit>min_frequency ='.$min_frequency.'</font> Минимальная частота  
интервалов, в которых ищутся пики, в Гц, пример 0.4 0.8 2.3 <font  
14px/Arial,Helvetica,sans-serif;;#000000;;inherit>max_frequency ='.$max_frequency.'  
Максимальная частота интервалов, в которых ищутся пики, в Гц. Пример 0.6 1.1  
2.5</font> <font 14px/Arial,Helvetica,sans-serif;;#000000;;inherit>double_frequency  
= '.$double_frequency.'</font> Количество пиков в интервале, допустимые значения 1 -  
один пик, 2 - два пика в интервале <font 14px/Arial,Helvetica,sans-  
serif;;#000000;;inherit>signal_level=0;</font> <font 14px/Arial,Helvetica,sans-  
serif;;#000000;;inherit>}';</font>`