**ОПИСАНИЕ**

программного обеспечения

для отображения данных от РЛС и гидроакустических систем

к проекту «Унифицированный ЭКМА»

автор ведущий инженер

Свиридов С. А.

АО «АМЭ»

2017

Оглавление

[1. ПО для отображения информации от РЛС 3](#_Toc488854016)

[1.1 Программа эмулятор РЛС. 3](#_Toc488854017)

[1.1.1. Установка программы. 4](#_Toc488854018)

[1.1.2. Запуск программы. 4](#_Toc488854019)

[1.2. Программа архивации данных РЛС. 6](#_Toc488854020)

[1.2.1. Управление программой. 7](#_Toc488854021)

[1.2.2. Длительность записи 8](#_Toc488854022)

[1.2.3. Шаблон имени файла и путь к каталогу 8](#_Toc488854023)

[1.2.4. Ретрансляция данных 9](#_Toc488854024)

[1.2.5. Примеры 9](#_Toc488854025)

[1.3. Программа отображения данных РЛС 11](#_Toc488854026)

[1.3.1. Примеры. 13](#_Toc488854027)

[2. ПО для отображения информации от гидроакустических систем 14](#_Toc488854028)

[2.1. Программа эмулятор гидроакустических сигналов. 14](#_Toc488854029)

[2.2. Программа архивации гидроакустических данных. 16](#_Toc488854030)

[2.2.1. Управление программой. 17](#_Toc488854031)

[2.2.2. Длительность записи 18](#_Toc488854032)

[2.2.3. Шаблон имени файла и путь к каталогу 18](#_Toc488854033)

[2.2.4. Ретрансляция данных 19](#_Toc488854034)

[2.2.5. Примеры 19](#_Toc488854035)

[2.3. Программа отображения гидроакустических данных. 21](#_Toc488854036)

[2.3.1. Примеры. 23](#_Toc488854037)

[4. Минимальные системные требования 24](#_Toc488854038)

[3. Информация от разработчика 24](#_Toc488854039)

# 1. ПО для отображения информации от РЛС

ПО для отображения гидроакустической информации состоит из трех программ:

1. Программа эмулятор первичной информации РЛС**;**
2. Программа архивации данных от РЛС **rlsarchiver;**
3. Программа отображения данных от РЛС **rlsviewer**.

**Функциональная схема взаимодействия программ**

Программа отображения данных РЛС

Архив

UDP

Программа архивации данных РЛС

Эмулятор РЛС

UDP

## 1.1 Программа эмулятор РЛС.

Программа эмулятор РЛС предназначена для эмуляции радиолокационной станции для тестирования ПО. Программа формирует пакеты данных первичной радиолокационной информации и отправляет их по сети, так же как это делает РЛС. Для формирования пакетов данных используются данные из файлов, которые были записаны на реальной РЛС. По сути, программа представляет собой проигрыватель файлов, записанных ранее.

Данная программа является сторонней разработкой.

### 1.1.1. Установка программы.

Скопировать папку MCS в корень диска C.

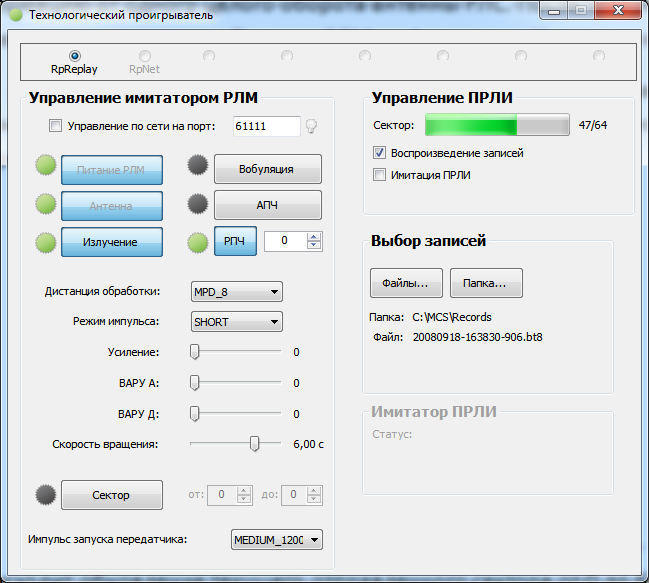
Если вы используете 32-битную версию windows, то необходимо исправить файл **lm.reg** – заменить все «\SOFTWARE\Wow6432Node\MCS» на «\SOFTWARE\MCS».

Запустить **register.bat** и убедиться, что библиотеки успешно зарегистрированы, а данные из файла **lm.reg** внесены в реестр.

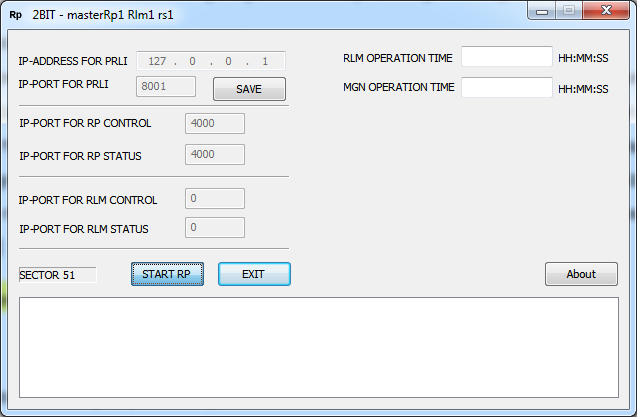
### 1.1.2. Запуск программы.

При запуске приложений необходимо соблюдать последовательность, описанную ниже. Если одно из приложений пришлось закрыть, то лучше перезапускать все остальные приложения тоже.

Запустить программу RpPlayerEx.exe. В разделе «Выбор записей» нажать кнопку «Файлы…» и выбрать несколько файлов записей, расположенных в папке «**Records»**. Каждый файл \*.bt8 содержит информацию от одного целого оборота антенны РЛС. После чего нажать последовательно кнопки «Питание РЛМ», «Антенна», «Излучение».



Запустить ярлык «RpServer.exe –rp1 –rs1». Это программа трансляции первичной информации (ПРЛИ) по сети. Данная информация транслируется по протоколу UDP на заданные адрес/порт.



Проконтролировать, что в поле «Sector» обоих приложений происходит обновление текущего отправленного сектора от 0 до 63. Чтобы изменить адрес/порт трансляции, необходимо остановить трансляцию, нажав кнопку «START RP» в программе RpServer. После чего ввести новые адрес и порт в поля «IP-ADDRESS FOR PRLI» и «IP-PORT FOR PRLI».

Возможно для корректной работы потребуется установить дополнительные компоненты Microsoft Visual Studio, расположенные в папке «**Redist**» (vcredist\_x86.exe и vcredist\_2010\_x86.exe).

Также, для работы RpPlayerEx.exe требуется .Net Framework 3.5 sp3 (входит в состав Windows 7 по умолчанию).

## 1.2. Программа архивации данных РЛС.

Программа **rlsarchiver** предназначена для сохранения данных от РЛС в архив.

Источником данных для программы служит радиолокационная станция, которая собирает радиолокационные данные, обрабатывает их и передает по сетям TCP/IP, используя протокол UDP. В тестовом режиме в качестве источника данных служит программа эмулятор РЛС. Описание программы эмулятора см. в разделе 1.1.

Программа **rlsarchiver** также ретранслирует полученные от источника данные на другой сетевой адрес/порт для отображения в программе **rlsviewer.**

Программа **rlsarchiver** может быть запущена с параметрами, определяющими режим работы.

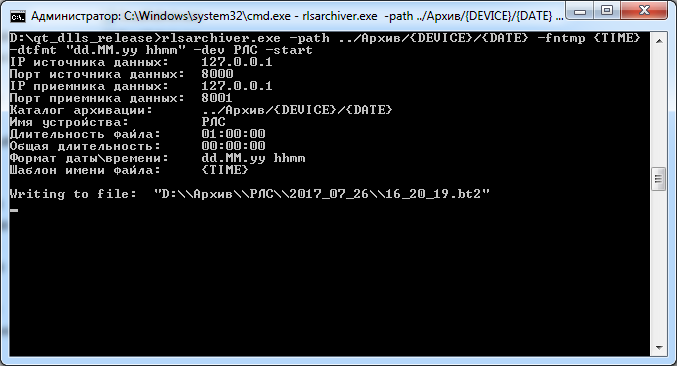
|  |  |  |  |
| --- | --- | --- | --- |
| **Параметр** | **Описание** | **Значения** | **Значение по умолчанию** |
| -ip | IP адрес для прослушивания данных. |  | 127.0.0.1 |
| -p, -port | Порт для приема данных по протоколу UDP от устройства. | 1-65535 | 8000 |
| -path,  -path\_to\_archive | Путь к каталогу архива. |  | ./archive |
| -dev, -device\_name | Имя устройства для подстановки в имя файла. |  | rls |
| -fd, -file\_duration | Длительность одного файла архивных данных в формате 'HHMMSS'. |  | 010000  (один час) |
| -td, -total\_duration | Общая длительность записи архивных данных в формате 'HHMMSS'. |  | 000000  (не ограничена) |
| -dtfmt,  -date\_time\_format | Формат даты времени для подстановки в имя файла. |  | ddMMyy\_hhmmss |
| -fntmp,  -file\_name\_template | Шаблон имени файла. |  | {DEVICE}\_{DATETIME} |
| -out\_ip | IP адрес для ретрансляции данных. |  | 127.0.0.1 |
| -out\_port | Порт для ретрансляции данных. | 1-65535 | 8001 |
| - autostart | Архивация начнется автоматически при запуске |  |  |

Для параметров path\_to\_archive и file\_name\_template допустимо использование регулярных выражений {DEVICE}, {DATETIME}, {DATE}, {TIME}. Описание их использования приведено в разделе 1.2.2.

У каждого параметра есть значение по умолчанию. Если какой-то параметр не был указан при запуске, то будет использовано его значение по умолчанию.

Нельзя использовать в шаблоне имени файла, пути к каталогу и формате времени/даты символы, недопустимые в именах файлов ( \ / : ? \* < > “ |).

После запуска программы, в окне терминала будет выведена информация о текущих параметрах.



Если при запуске программы какой-либо параметр был введен неверно, программа прерывается и на экран выводится справка. Также справку можно вызвать набрав -help в качестве параметра.

### 1.2.1. Управление программой.

Программа **rlsarchiver** является консольным приложением. Поэтому управление приложением происходит при помощи команд вводимых в окне терминала.

**start** – запускает архивацию. Если в командной строке был введен параметр -autostart, то архивация начнется автоматически после запуска программы.

**stop** – остановить архивацию. Программа при этом продолжает работать. После ввода команды start, архивация продолжится.

**quit** – остановить архивацию и завершить программу.

### 1.2.2. Длительность записи

Программа сохраняет архив в виде отдельных файлов с указанной длительностью. То есть, запись в один файл производится в течение указанного времени. После истечения этого времени, открывается новый файл, и запись продолжается в него. И так далее. Такая система позволяет в последующем быстро находить нужные данные.

Если задано общее время архивации (total\_duration), то архивация прекращается после превышения этого времени.

### 1.2.3. Шаблон имени файла и путь к каталогу

Программа сохраняет файлы с определенными именами. Имя файла определяется параметром file\_name\_template. В шаблоне имени файла можно использовать регулярные выражения {DEVICE}, {DATETIME}, {DATE}, {TIME} для определения имен создаваемых файлов. При создании нового файла, регулярные выражения заменяются соответствующим значением.

{DEVICE} – имя устройства, определяемое параметром device\_name;

{DATETIME} – текущие дата и время;

{DATE} – текущая дата;

{TIME} – текущее время.

Выражения даты и времени заменяются соответствующими им значениям согласно формату, определяемым параметром date\_time\_format.

При задании пути к каталогу архива, также можно использовать вышеуказанные регулярные выражения. После замены регулярных выражений соответствующим значением, файлы архива сохраняются в этот каталог. Если каталог не существует, то он создается.

### 1.2.4. Ретрансляция данных

Программа считывает данные, передаваемые устройством (или программой эмулятором) из порта, определенного параметром port и сохраняет эти данные в архив. Однако, после считывания из порта, эти данные становятся недоступны для других программ. Особенность работы сетевой карты, такова, что полученные данные могут быть прочитаны только одной программой. После этого сетевой буфер очищается. Поэтому чтобы иметь возможность, также отображать данные в реальном времени на дисплее, программа **rlsarchiver**, после сохранения в архив ретранслирует полученные данные на другой адрес/порт. Таким образом, она становится прозрачной для сетевых данных и программа **rlsviewer** может получать и отображать данные в реальном времени.

Для указания адреса и порта для ретрансляции служат параметры out\_ip и out\_port.

### 1.2.5. Примеры

**>> rlsarchiver -ip 192.168.75.32 -fd 023000 -td 480000 -dtfmt “dd.MM.yy hhmm”**

Программа будет получать данные с адреса 192.168.75.32. Архивные файлы будут сохраняться в тот же каталог, где располагается программа, в подкаталог archive. Каждый файл будет записываться в течение 2 часов 30 минут. Общая длительность записи составит 48 часов (двое суток) и после этого будет остановлена. Определен новый формат даты/времени. Остальные параметры имеют значения по умолчанию.

Таким образом, если запись была начата 27 июля 2017 года в 10 часов 20 минут, то в каталоге archive будут сформированы следующие файлы:

rls\_26.07.17 1020.bt2

rls\_26.07.17 1250.bt2

…

rls\_26.07.17 2250.bt2

rls\_27.07.17 0120.bt2

…

rls\_28.07.17 0720.bt2

rls\_28.07.17 0950.bt2

Причем, последний файл будет иметь длительность всего 30 минут, так как после закончится общее время записи.

**>> rlsarchiver -path ../Архив/{DEVICE}/{DATE}/ -dtfmt “dd\_MM\_yyyy hhmm” -fntmp {TIME} -dev РЛС**

Каталог архива «**Архив**» будет располагаться на один уровень выше каталога программы. В каталоге «**Архив»** будет создан подкаталог с заданным именем устройства «**РЛС».** В каталоге **«РЛС»** в свою очередь, на каждый новый день будет создаваться подкаталог с новой датой. Общая длительность записи не ограничена. Определен новый формат даты/времени. Определен новый шаблон имени файла. Задано имя устройства. Остальные параметры имеют значения по умолчанию.

Таким образом, если запись была начата 27 июля 2017 года в 16 часов 20 минут, то в каталоге «**Архив/РЛС»** будут создан подкаталог «**27\_07\_2017»,** в которомбудут сформированы следующие файлы:

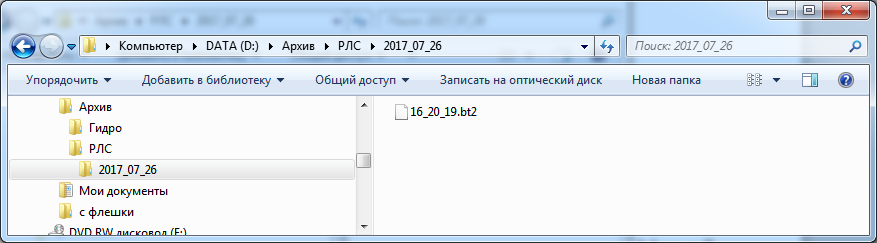
1620.bt2

1720.bt2

1820.bt2

…

и так далее.



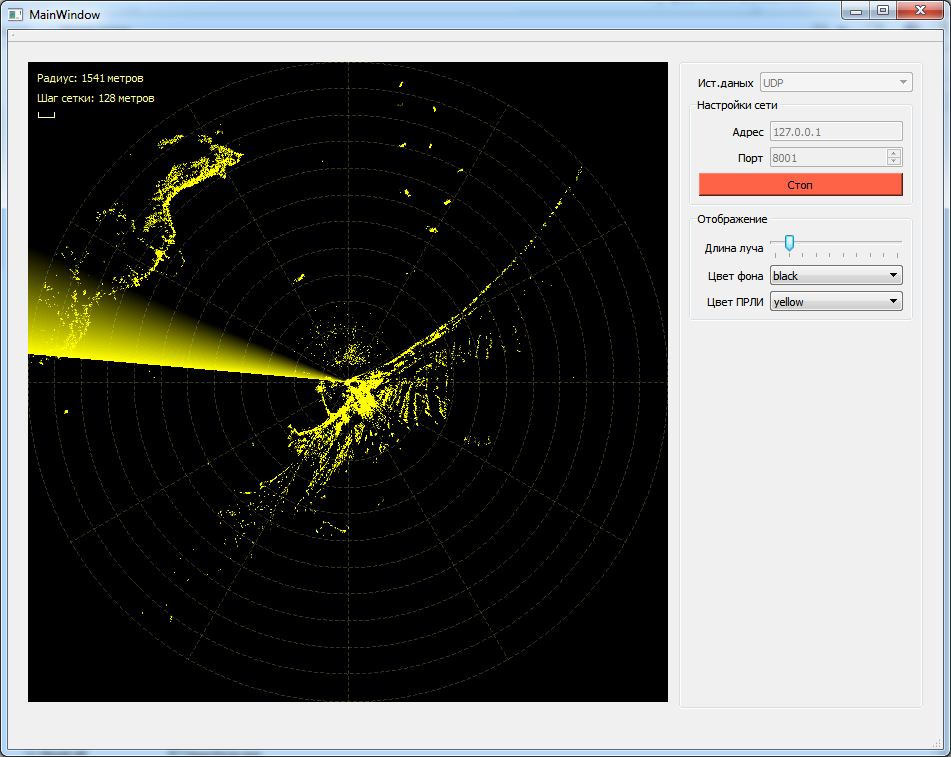
В 00 часов 20 минут 28 июля в каталоге «**Архив/РЛС»** будет сформирован новый подкаталог «**28\_07\_2017»**. Запись продолжится в этот подкаталог. И так далее.

## 1.3. Программа отображения данных РЛС

Программа **rlsviewer** предназначена для отображения данных от РЛС. Программа может отображать данные в двух режимах: в режиме реального времени или из архива.

В режиме отображения данных в реальном времени источником данных служит радиолокационная станция, которая собирает радиолокационные данные, обрабатывает их и передает по сетям TCP/IP, используя протокол UDP. В тестовом режиме в качестве источника данных служит программа эмулятор РЛС. Описание программы эмулятора см. в разделе 1.1.

В режиме отображения архивной информации программа считывает данные из файлов архива, сформированных программой для архивации **rlsarchiver,** и выводит эти данные на дисплей.



Программа **rlsviewer** может быть запущена с параметрами, определяющими режим работы.

|  |  |  |  |
| --- | --- | --- | --- |
| **Параметр** | **Описание** | **Значения** | **Значение по умолчанию** |
| -s,  -source | Источник данных для вывода на дисплей: сеть или архив. | udp, archive | udp |
| -ip | IP адрес для прослушивания данных |  | 127.0.0.1 |
| -p,  -port | Порт для приема данных по протоколу UDP | 1-65535 | 8001 |
| -path,  -path\_to\_archive | Путь к каталогу архива |  | ./archive |
| -bc, -bcolor | Цвет фона дисплея |  | #000000 |
| -dc, -dcolor | Цвет данных на дисплее |  | #FFFF00 |
| -display,  -display\_point\_count | Размер дисплея в точках. По умолчанию 640x640. |  | 640 |
| -line,  -line\_point\_count | Количество точек линейки данных РЛС, выводимых на дисплей. Каждая линейка соответствует одному лучу. Максимум 1460 точек на один луч. | 300 - 1460 | 1200 |
| -autostart | Автоматический запуск отображения при запуске. | Нет |  |
| -nocontrols | Не отображать элементы управления. | Нет |  |

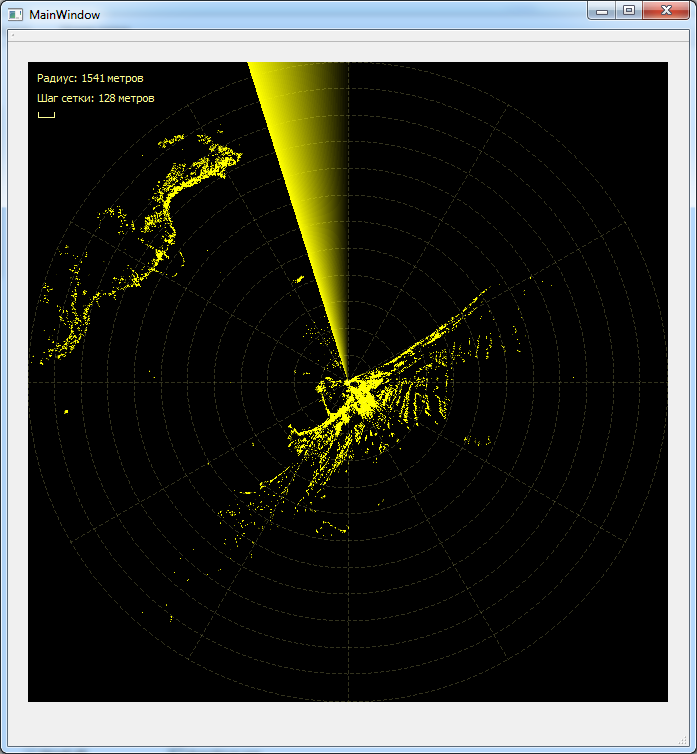
Параметры autostart и nocontrols имеют эффект, только если параметр source=udp.

Параметр line\_point\_count по сути определяет масштаб выводимого изображения. Чем меньше количество точек, тем крупнее изображение.

У каждого параметра есть значения по умолчанию. Если какой-то параметр не был указан при запуске, то будет использовано его значение по умолчанию.

При закрытии, программа сохраняет текущие параметры в файле rlsviewer.ini. Если программа запускается без параметров командной строки, то используются параметры из файла ini. Параметры из файла ini используются, только если не было введено ни одного параметра в командной строке. Если же был введен хоть один параметр в командной строке, то используются значения параметров командной строки.

### 1.3.1. Примеры.

**>> rlsviewer -autostart -ip 192.168.75.132 -display 600 -line 400 -nocontrols**

Программа после запуска автоматически начнет прослушивать адрес 192.168.75.132. Размер дисплея будет установлен 600x600 точек. На экран будут выводиться 400 первых точек линейки. Элементы управления будут скрыты. Все остальные параметры будут иметь значение по умолчанию.

**>> rlsviewer -source archive -autostart -path /home/user/archive**

Программа запустится в режиме отображения данных из архива. Путь к каталогу задает параметр path. Пользователю необходимо выбрать файлы для отображения и нажать кнопку «Старт». Параметр autostart в данном примере не имеет эффекта.

# 2. ПО для отображения информации от гидроакустических систем

ПО для отображения гидроакустической информации состоит из трех программ:

1. Программа эмулятор гидроакустических сигналов **gaemulator;**
2. Программа архивации гидроакустических данных **gaarchiver;**
3. Программа отображения гидроакустических данных **gaviewer**.

**Функциональная схема взаимодействия программ**

Программа отображения данных гидроакустики

Архив

UDP

Программа архивации данных гидроакустики

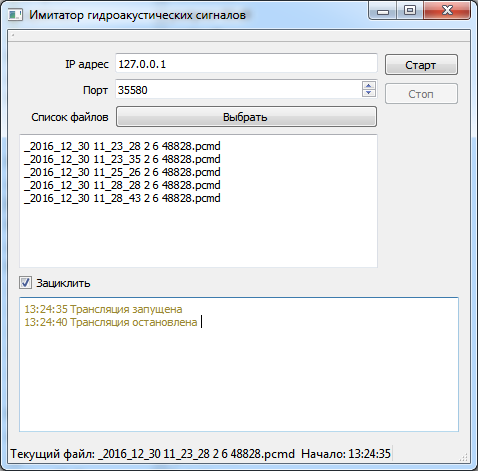
Эмулятор Гидроакустической системы

UDP

## 2.1. Программа эмулятор гидроакустических сигналов.

Программа **gaemulator** предназначена для эмуляции звукоулавливающей аппаратуры. Программа формирует пакеты данных и отправляет их по сети, так же как это делает звукоулавливающая аппаратура. Для формирования пакетов данных используются данные из файлов, которые были записаны на реальной аппаратуре. По сути, программа представляет собой проигрыватель файлов, записанных ранее.

После запуска программы, необходимо указать адрес и порт, на которые необходимо транслировать данные, то есть адрес/порт которые слушает программа приемник. Также необходимо выбрать файл, которые необходимо проигрывать. После этого нажать кнопку «Старт». Программа начнет читать файлы по порядку и транслировать их на указанный адрес/порт. Если установлена галочка «Зациклить», то файлы будут воспроизводиться по кругу, без остановки.



В строке состояния отображается имя файла, который транслируется в текущий момент и время начала его трансляции.

Нажатие кнопки «Стоп» прекращает трансляцию.

## 2.2. Программа архивации гидроакустических данных.

Программа **gaarchiver** предназначена для сохранения данных гидроакустики в архив.

Источником данных для программы служит звукоулавливающая аппаратура, которая улавливает звуковой сигнал, обрабатывает его и передает его по сети TCP по протоколу UDP. В тестовом режиме в качестве источника данных служит программа эмулятор **gaemulator**. Описание данной программы см. в разделе 2.1.

Программа **gaarchiver** также ретранслирует полученные от источника данные на другой сетевой адрес/порт для отображения в программе **gaviewer.**

Программа **gaarchiver** может быть запущена с параметрами, определяющими режим работы.

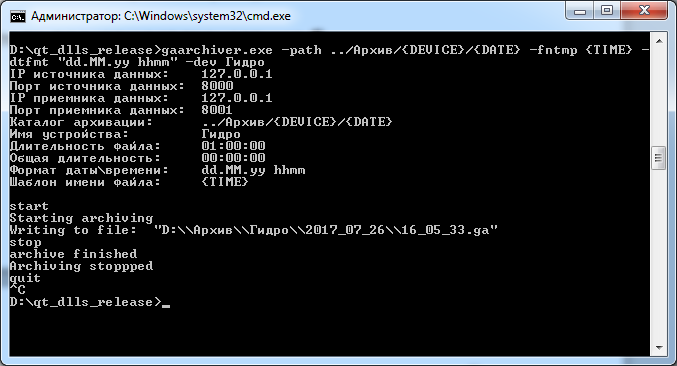
|  |  |  |  |
| --- | --- | --- | --- |
| **Параметр** | **Описание** | **Значения** | **Значение по умолчанию** |
| -ip | IP адрес для прослушивания данных. |  | 127.0.0.1 |
| -p, -port | Порт для приема данных по протоколу UDP от устройства. | 1-65535 | 9000 |
| -path,  -path\_to\_archive | Путь к каталогу архива. |  | ./archive |
| -dev, -device\_name | Имя устройства для подстановки в имя файла. |  | ga |
| -fd, -file\_duration | Длительность одного файла архивных данных в формате 'HHMMSS'. |  | 010000  (один час) |
| -td, -total\_duration | Общая длительность записи архивных данных в формате 'HHMMSS'. |  | 000000  (не ограничена) |
| -dtfmt,  -date\_time\_format | Формат даты времени для подстановки в имя файла. |  | ddMMyy\_hhmmss |
| -fntmp,  -file\_name\_template | Шаблон имени файла. |  | {DEVICE}\_{DATETIME} |
| -out\_ip | IP адрес для ретрансляции данных. |  | 127.0.0.1 |
| -out\_port | Порт для ретрансляции данных. | 1-65535 | 9001 |
| -autostart | Архивация начнется автоматически при запуске |  |  |

Для параметров path\_to\_archive и file\_name\_template допустимо использование регулярных выражений {DEVICE}, {DATETIME}, {DATE}, {TIME}. Описание их использования приведено в разделе 2.2.2.

У каждого параметра есть значение по умолчанию. Если какой-то параметр не был указан при запуске, то будет использовано его значение по умолчанию.

Нельзя использовать в шаблоне имени файла, пути к каталогу и шаблоне времени/даты символы, недопустимые в именах файлов ( \ / : ? \* < > “ |).

После запуска программы, в окне терминала будет выведена информация о текущих параметрах.



Если при запуске программы какой-либо параметр был введен неверно, программа прерывается и на экран выводится справка. Также справку можно вызвать набрав -help в качестве параметра.

### 2.2.1. Управление программой.

Программа gaarchiver является консольным приложением. Поэтому управление приложением происходит при помощи команд вводимых в окне терминала.

**start** – запускает архивацию. Если в командной строке был введен параметр -autostart, то архивация начнется автоматически после запуска программы.

**stop** – остановить архивацию. Программа при этом продолжает работать. После ввода команды start, архивация продолжится.

**quit** – остановить архивацию и завершить программу.

### 2.2.2. Длительность записи

Программа сохраняет архив в виде отдельных файлов с указанной длительностью. То есть, запись в один файл производится в течение указанного времени. После истечения этого времени, открывается новый файл, и запись продолжается в него. И так далее. Такая система позволяет в последующем быстро находить нужные данные.

Если задано общее время архивации (total\_duration), то архивация прекращается после превышения этого времени.

### 2.2.3. Шаблон имени файла и путь к каталогу

Программа сохраняет файлы с определенными именами. Имя файла определяется параметром file\_name\_template. В шаблоне имени файла можно использовать регулярные выражения {DEVICE}, {DATETIME}, {DATE}, {TIME} для определения имен создаваемых файлов. При создании нового файла, регулярные выражения заменяются соответствующим значением.

{DEVICE} – имя устройства, определяемое параметром device\_name;

{DATETIME} – текущие дата и время;

{DATE} – текущая дата;

{TIME} – текущее время.

Выражения даты и времени заменяются соответствующими им значениям согласно формату, определяемым параметром date\_time\_format.

При задании пути к каталогу архива, также можно использовать вышеуказанные регулярные выражения. После замены регулярных выражений соответствующим значением, файлы архива сохраняются в этот каталог. Если каталог не существует, то он создается.

### 2.2.4. Ретрансляция данных

Программа считывает данные, передаваемые устройством (или программой эмулятором) из порта, определенного параметром port и сохраняет эти данные в архив. Однако, после считывания из порта, эти данные становятся недоступны для других программ. Особенности работы сетевой карты, таковы, что полученные данные могут быть прочитаны только одной программой. После этого сетевой буфер очищается. Поэтому чтобы иметь возможность, также отображать данные в реальном времени на дисплее, программа **gaarchiver**, после сохранения в архив ретранслирует полученные данные на другой адрес/порт. Таким образом она становится прозрачной для сетевых данных и программа **gaviewer** может получать и отображать данные в реальном времени.

Для указания адреса и порта для ретрансляции служат параметры out\_ip и out\_port.

### 2.2.5. Примеры

**>> gaarchiver -ip 192.168.75.32 -fd 023000 -td 480000 -dtfmt “dd.MM.yy hhmm”**

Программа будет получать данные с адреса 192.168.75.32. Архивные файлы будут сохраняться в тот же каталог, где располагается программа, в подкаталог archive. Каждый файл будет записываться в течение 2 часов 30 минут. Общая длительность записи составит 48 часов (двое суток) и после этого будет остановлена. Определен новый формат даты/времени. Остальные параметры имеют значения по умолчанию.

Таким образом, если запись была начата 27 июля 2017 года в 10 часов 20 минут, то в каталоге archive будут сформированы следующие файлы:

ga\_26.07.17 1020.ga

ga\_26.07.17 1250.ga

…

ga\_26.07.17 2250.ga

ga\_27.07.17 0120.ga

…

ga\_28.07.17 0720.ga

ga\_28.07.17 0950.ga

Причем, последний файл будет иметь длительность всего 30 минут, так как после закончится общее время записи.

**>> gaarchiver -path ../Архив/{DEVICE}/{DATE}/ -dtfmt “dd\_MM\_yyyy hhmm” -fntmp {TIME} -dev Гидроакустика**

Каталог архива «**Архив**» будет располагаться на один уровень выше каталога программы. В каталоге «**Архив»** будет создан подкаталог с заданным именем устройства «**Гидроакустика».** В каталоге **«Гидроакустика»** в свою очередь, на каждый новый день будет создаваться подкаталог с новой датой. Общая длительность записи не ограничена. Определен новый формат даты/времени. Определен новый шаблон имени файла. Задано имя устройства. Остальные параметры имеют значения по умолчанию.

Таким образом, если запись была начата 27 июля 2017 года в 10 часов 20 минут, то в каталоге «**Архив/Гидроакустика»** будут создан подкаталог «**27\_07\_2017»,** и в этом каталоге будут сформированы следующие файлы:

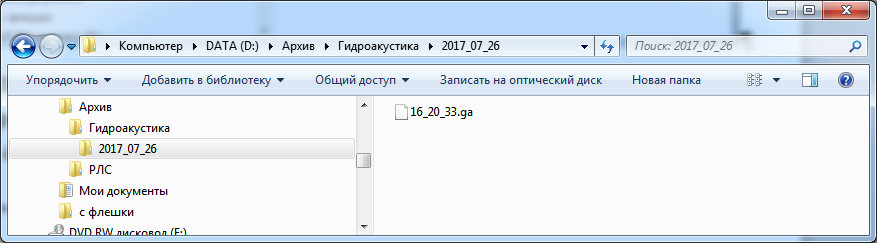
1020.ga

1120.ga

1220.ga

…

и так далее.



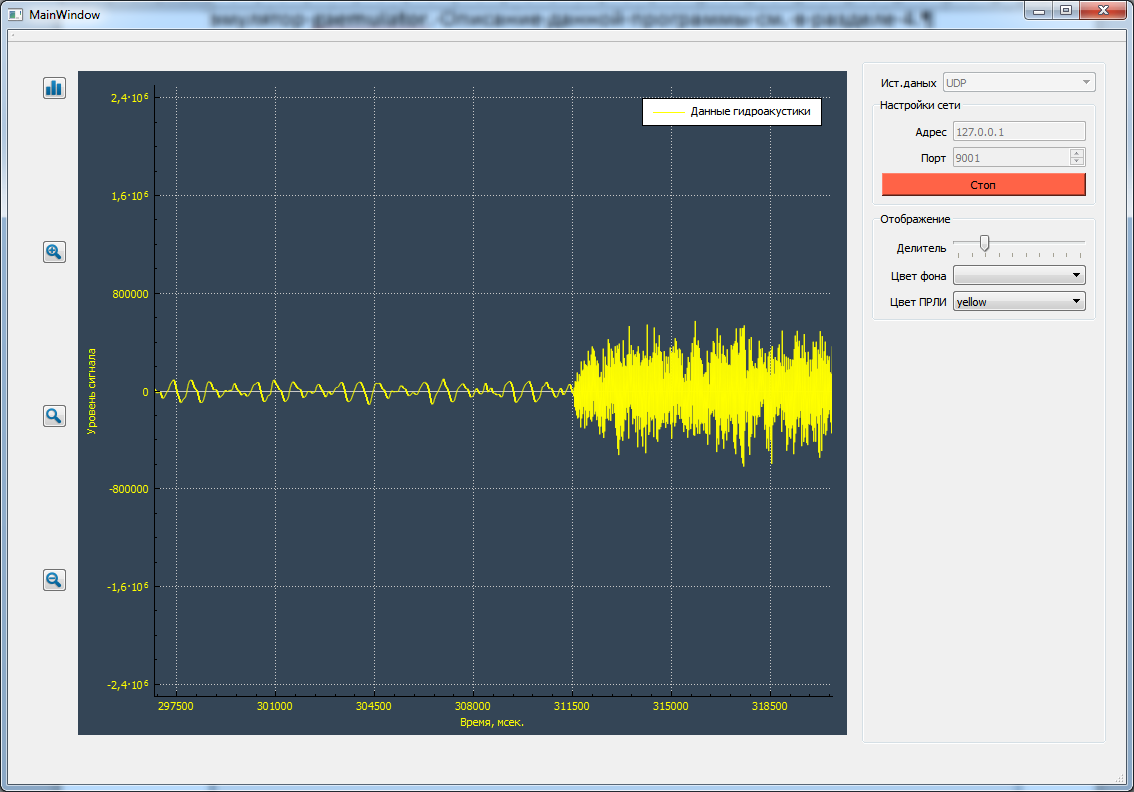
В 00 часов 20 минут 28 июля в каталоге «**Архив/Гидроакустика»** будет сформирован новый каталог «**28\_07\_2017»**. Запись продолжится в этот каталог. И так далее.

## 2.3. Программа отображения гидроакустических данных.

Программа **gaviewer** предназначена для отображения данных гидроакустики. Программа может отображать данные в двух режимах: в режиме реального времени или из архива.

В режиме отображения данных в реальном времени источником данных служит звукоулавливающая аппаратура, которая улавливает звуковой сигнал, обрабатывает его и передает его по сети TCP по протоколу UDP. В тестовом режиме в качестве источника данных служит программа эмулятор **gaemulator**. Описание данной программы см. в разделе 4.

В режиме отображения архивной информации программа считывает данные из файлов архива, сформированных программой для архивации **gaarchiver,** и выводит эти данные на дисплей.



Программа **gaviewer** может быть запущена с параметрами, определяющими режимы работы.

|  |  |  |  |
| --- | --- | --- | --- |
| **Параметр** | **Описание** | **Значения** | **Значение по умолчанию** |
| -s,  -source | источник данных для вывода на дисплей: сеть или архив. | udp, archive | udp |
| -ip | IP адрес для прослушивания данных |  | 127.0.0.1 |
| -p,  -port | Порт для приема данных по протоколу UDP | 1-65535 | 9001 |
| -path,  -path\_to\_archive | Путь к каталогу архива |  | ./archive |
| -bc, -bcolor | Цвет фона дисплея |  | #000000 |
| -dc, -dcolor | Цвет данных на дисплее |  | #FFFF00 |
| -buffer,  -buffer\_point\_count | Размер буфера отображения в точках. | 96000, 48000, 24000, 16000, 12000, 8000, 6000, 4000, 2000 | 48000 |
| -autostart | Автоматический запуск отображения при запуске. |  |  |
| -autoscale | Автомасштаб по оси y. |  |  |
| -nocontrols | Не отображать элементы управления. |  |  |

Параметры autostart и nocontrols имеют эффект, только если параметр source=udp.

У каждого параметра есть значения по умолчанию. Если какой-то параметр не был указан при запуске, то будет использовано его значение по умолчанию. При закрытии программа сохраняет текущие параметры в файле gaviewer.ini. Если программа запускается без параметров, то используются параметры из файла ini.

Параметры из файла ini используются только если не было введено ни одного параметра в командной строке. Если же был введен хоть один параметр в командной строке, то используются значения параметров командной строки.

Параметр buffer определяет значение делителя. Делитель определяет какое количество данных выводится на экран за единицу времени. Так при значении параметра buffer = 48000, делитель будет равен 1 и на экран будет выводиться сигнал за 1 секунду. При значении buffer = 24000 – 0,5 секунды. И так далее.

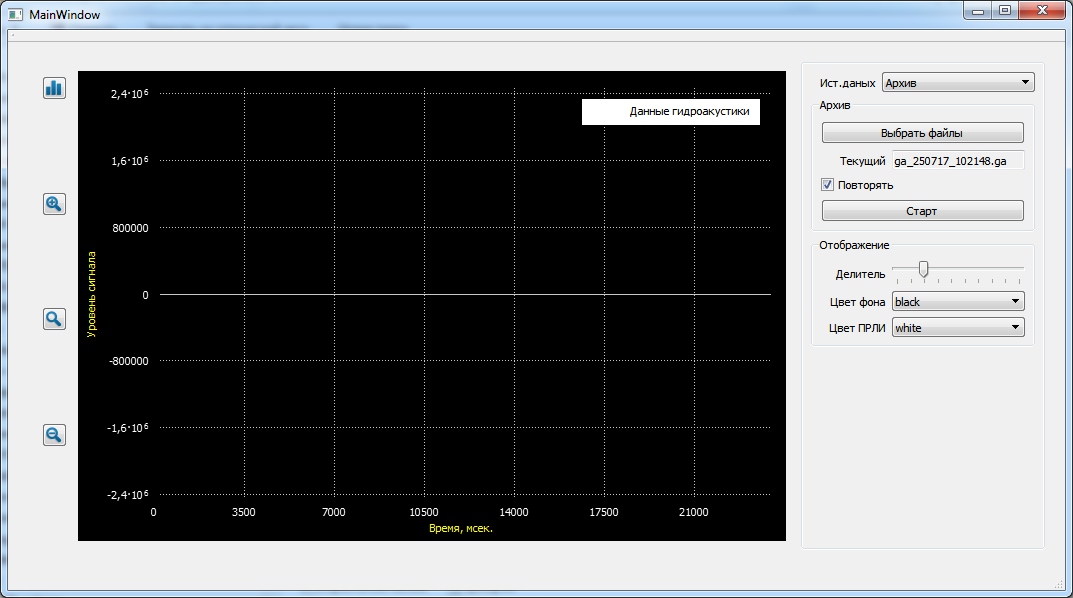
### 2.3.1. Примеры.

**>> gaviewer -autostart -ip 192.168.75.132 -buffer 24000**

Программа после запуска автоматически начнет прослушивать адрес 192.168.75.132. Размер буфера дисплея будет установлен равным 24000 точек. Все остальные параметры будут иметь значение по умолчанию.

**>> gaviewer -source archive -autostart -path /home/user/archive**

Программа запустится в режиме отображения данных из архива. Путь к каталогу задает параметр path. Пользователю необходимо выбрать файлы для отображения и нажать кнопку «Старт». Параметр autostart в данном примере не имеет эффекта.



# 4. Минимальные системные требования

Процессор Intel i3 2 ГГц.

Память 2 Гб.

Сетевая карта 100 мбит/сек.

Видеокарта с поддержкой OpenGL.

Так как все приложения работают с большими объемами данных в режиме реального времени, то для наилучших результатов аппаратная часть компьютера, на котором производится запуск ПО должна иметь определенный запас производительности.

# 3. Информация от разработчика

Все приложения, за исключением программы эмулятора РЛС написаны с использованием среды разработки Qt 5.5

Используемые компиляторы:

Windows - MinGW 5.3.0 32 bit

Astra Linux – GCC 64 bit