



Общество с ограниченной ответственностью

Тэк Про Индастриз

Limited Liability Company

Tech Pro Industries

Руководство
Программное обеспечение
«Lazarus SoftW RS CT»

Москва, 2023 г.

Содержание

1 Установка и запуск	3
2 Функциональное назначение	5
2.1 Классы решаемых задач	5
3 Описание логической структуры.....	6
3.1 Алгоритм функционирования	6
4 Вызов и загрузка.....	8
4.1 Способ вызова программы с носителя данных	8
4.2 Типовое применение в дуплексном режиме.....	9
4.3 Типовое применение в полудуплексном режиме.....	10
Приложение А	11

1 Установка и запуск

1) Для скачивания ПО Lazarus SoftW RS CT, на персональном компьютере в браузере перейдите на страницу ООО «Тэк Про Индастриз» во вкладку Программное обеспечение, по ссылке: <https://techproin.com/lazarussoftware/>

2) На открывшейся странице находится вся краткая информация о ПО Lazarus SoftW RS CT. Под изображением находится ссылка для скачивания архива с дистрибутивом Lazarus SoftW RS CT, нажмите на нее (рисунок 1). Введите пароль для скачивания архива.

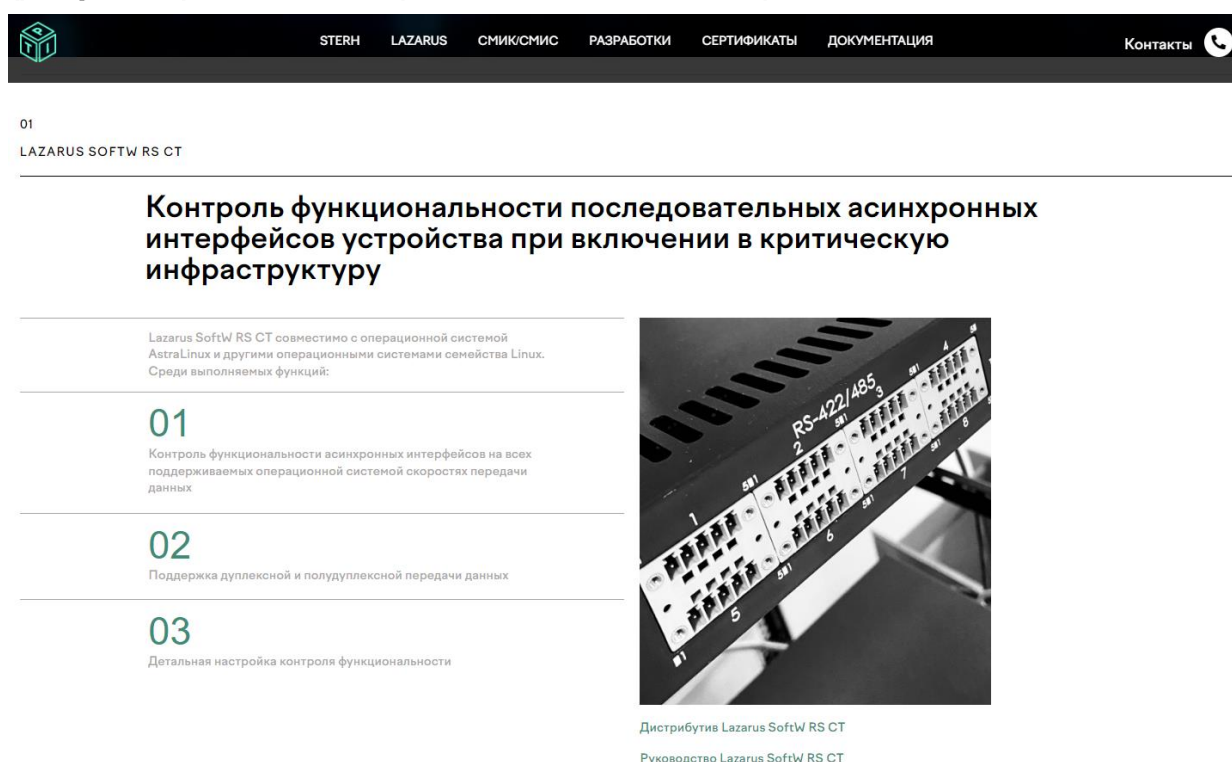


Рисунок 1 – Страница скачивания ПО Lazarus SoftW RS CT

3) Распакуйте скачанный архив с ПО Lazarus SoftW RS CT, на персональном компьютере.

4) Убедитесь, что программно-аппаратный комплекс Lazarus IC и персональный компьютер находятся в одной локальной сети.

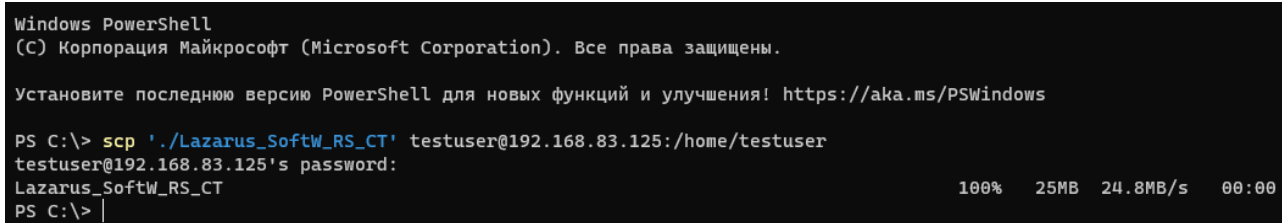
5) Определите IP-адрес Lazarus IC.

6) Для переноса ПО Lazarus SoftW RS CT на Lazarus IC необходимо в месте расположения ПО открыть окно терминала и перенести файл командой “scp”*.

*Перенос командой scp: `<scp './Lazarus_SoftW_RS_CT' testuser@192.168.83.125:/home/tetsuser>`, где `<testuser>` является именем пользователя в ОС Lazarus IC, `<192.168.83.125>` - его IP-адресс, а `</home/tetsuser>` - директория куда переносится файл.

7) При выполнении переноса командой scp необходимо будет ввести пароль от Lazarus IC.

Пример переноса в окне терминала приведен на рисунке 2.



```
Windows PowerShell
(С) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

Установите последнюю версию PowerShell для новых функций и улучшения! https://aka.ms/PSWindows

PS C:\> scp './Lazarus_SoftW_RS_CT' testuser@192.168.83.125:/home/testuser
testuser@192.168.83.125's password:
Lazarus_SoftW_RS_CT
PS C:\> |
```

Рисунок 2 – Перенос установочного файла ПО Lazarus SoftW RS CT на программно-аппаратный комплекс Lazarus IC

2 Функциональное назначение

2.1 Классы решаемых задач

2.1.1 Программное обеспечение решает следующие классы задач:

- Контроль функциональности последовательных асинхронных интерфейсов, включая UART, EIA/TIA-232E, EIA/TIA-422-B, EIA/ TIA-485-A устройства при включении в критическую инфраструктуру, без использования вспомогательного оборудования;
- Контроль функциональности последовательных асинхронных интерфейсов на всех поддерживаемых ОС скоростях передачи данных в дуплексном и полудуплексном режимах;
- Контроль функциональности последовательных асинхронных интерфейсов со следующими параметрами передачи данных: 8 бит в одном передаваемом символе, 1 стоп-бит, без контроля чётности, без управления потоком.

2.1.2 Программное обеспечение может применяться в автоматизированном тестировании для контроля качества функционирования программно-аппаратного обеспечения.

3 Описание логической структуры

3.1 Алгоритм функционирования

3.1.1 Функциональная схема взаимодействия и связей программы в составе тестового стенда представлена на рисунке 3.

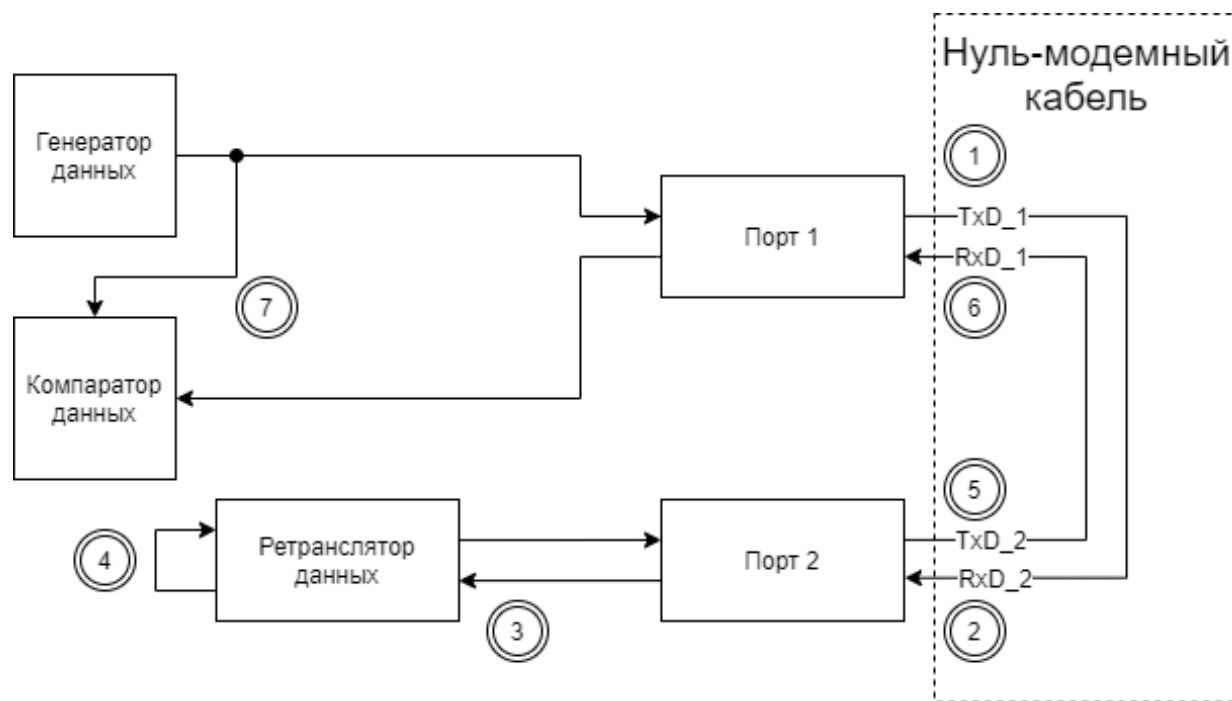


Рисунок 3 – Функциональная схема Lazarus SoftW RS CT

3.1.2 Этапы прохождения информации пронумерованы и отмечены на функциональной схеме. Номера этапов соответствуют младшим номерам следующих подразделов.

3.1.2.1 В цикле генератор данных формирует пакеты случайных тестовых данных заданной при помощи входного параметра длины и передаёт в последовательный Порт 1;

3.1.2.2 Данные по нуль-модемному кабелю поступают на вход Порта 2;

3.1.2.3 Данные принимаются с Порта 2 порциями, размер которых определяется входным параметром, и поступают на вход ретранслятора данных;

3.1.2.4 Ретранслятор данных перенаправляет полученный массив данных на передачу в Порт 2;

3.1.2.5 Производится обратная передача массива данных;

3.1.2.6 Через нуль-модемный кабель массив данных поступает на вход Porta 1;

3.1.2.7 Полученные из Porta 1 массивы данных аккумулируются в пакет и, при достижении полного размера пакета, результирующий пакет передаётся компаратору данных для сравнения с исходным пакетом тестовых данных. Если расхождения в данных не обнаружено, то цикл повторяется. Если обнаружено расхождение в данных или кадр не получен в течение ожидаемого времени, фиксируется ошибка;

3.1.2.8 Программа может завершить работу либо при достижении максимального количества ошибок, либо при получении указанного количества пакетов, либо по истеканию максимального времени работы. Опции завершения программы конфигурируются через параметры командной строки. Так же программа может быть прервана Пользователем.

4 Вызов и загрузка

4.1 Способ вызова программы с носителя данных

4.1.1 Программа может запускаться как с жёсткого диска машины, так и с других носителей. Для запуска программы у пользователя должны быть права root, либо пользователь должен состоять в группе dialout.

4.1.2 Исполняемый файл принимает следующие параметры командной строки:

--help – описание основных параметров;

```
testuser@astra-arm64:~$ sudo ./Lazarus_SoftW_RS_CT --help
General options:
--help                produce help message
--help-extended       produce help message for extended options
--device arg          name of serial device under test
--baud_rate arg       serial device baud rate
--packet_size arg (=2048) size of test data packets
```

--help-extended – описание дополнительных параметров;

```
testuser@astra-arm64:~$ sudo ./Lazarus_SoftW_RS_CT --help-extended
Extended options:
--device_timeout arg (=1000)    serial device data packet read timeout, ms
--loopback arg (=dev/null)      name of loopback serial device under test
--loopback_chunk_size arg (=32) size of loopback data chunks
--loopback_timeout arg (=200)   loopback device data chunk read timeout, ms
--errors_max arg (=1)          maximal number of errors to stop
--num_of_packets arg (=10000)   number of packets
--test_time arg                test time, s
```

--device – имя устройства последовательного порта отправителя тестовых данных;

--loopback – имя устройства последовательного порта компонента ретранслятора данных (петли). Значение данного параметра можно установить в **/dev/null**, при этом компонент ретранслятора данных работать не будет, в дуплексном режиме данные могут перенаправляться с выхода на вход порта при помощи специального тестового кабеля, значения параметров **loop_back_chunk_size** и **loopback_timeout** не будут оказывать эффекта;

--baud_rate – скорость передачи данных;

--num_of_packets – максимальное количество пакетов, после отправки которых программа закончит работу;

```
testuser@astra-arm64:~$ sudo ./Lazarus_SoftW_RS_CT --device=/dev/ttyUSB0 --loopback=/dev/ttyUSB2 --baud_rate=921600 --num_of_packets=500
Test start time: Tue Nov 7 17:41:51 2023
Testing...
[=====] 100 %
500 packets were sent
Test elapsed time: 0h 0m 20s 439ms
```

--packet_size – размер тестовых пакетов данных в байтах;

--test_time – максимальное время работы программы, с. Этот параметр невозможно использовать совместно с num_of_packets – возможно использование только одного из этих параметров.

```
testuser@astra-arm64:~$ sudo ./Lazarus_SoftW_RS_CT --device=/dev/ttyUSB0 --loopback=/dev/ttyUSB2 --baud_rate=921600 --packet_size=5 --test_time=20
Test start time: Tue Nov 7 17:46:08 2023
Testing...
[=====] 100 %
100 packets were sent
Test elapsed time: 0h 0m 20s 70ms
```

--device_timeout – время ожидания данных на порте отправителя тестовых данных, мс (пока не будет отправлено 10000 пакетов);

```
testuser@astra-arm64:~$ sudo ./Lazarus_SoftW_RS_CT --device=/dev/ttyUSB0 --loopback=/dev/ttyUSB2 --baud_rate=921600 --device_timeout=100
Test start time: Thu Nov 9 12:18:46 2023
Testing...
[=====] 100 %
10000 packets were sent
Test elapsed time: 0h 6m 48s 837ms
```

--loopback_chunk_size – размер массива данных, которые получает за раз компонент ретранслятора данных (пока не будет отправлено 10000 пакетов);

```
testuser@astra-arm64:~$ sudo ./Lazarus_SoftW_RS_CT --device=/dev/ttyUSB0 --loopback=/dev/ttyUSB2 --baud_rate=921600 --loopback_chunk_size=10000
Test start time: Thu Nov 9 12:33:14 2023
Testing...
[=====] 100 %
10000 packets were sent
Test elapsed time: 0h 33m 28s 22ms
```

--loopback_timeout – время ожидания данных на порте компонента ретранслятора данных, мс (пока не будет отправлено 10000 пакетов);

```
testuser@astra-arm64:~$ sudo ./Lazarus_SoftW_RS_CT --device=/dev/ttyUSB0 --loopback=/dev/ttyUSB2 --baud_rate=921600 --loopback_timeout=100
Test start time: Thu Nov 9 13:09:40 2023
Testing...
[=====] 100 %
10000 packets were sent
Test elapsed time: 0h 6m 48s 622ms
```

--errors_max – максимальное количество ошибок, при превышении которого программа закончит работу (или пока не будет отправлено 10000 пакетов);

```
testuser@astra-arm64:~$ sudo ./Lazarus_SoftW_RS_CT --device=/dev/ttyUSB0 --loopback=/dev/ttyUSB2 --baud_rate=921600 --errors_max=1
Test start time: Thu Nov 9 13:19:10 2023
Testing...
[=====] 100 %
10000 packets were sent
Test elapsed time: 0h 6m 49s 110ms
```

4.2 Типовое применение в дуплексном режиме

4.2.1 В дуплексном режиме работают интерфейсы UART, EIA/TIA-232E, EIA/TIA-422-B.

4.2.2 Для работы в дуплексном режиме значение параметра `loopback_chunk_size` должно быть меньше значения параметра `packet_size`.

4.3 Типовое применение в полудуплексном режиме

4.3.1 В полудуплексном режиме могут работать все интерфейсы: UART, EIA/TIA-232E, EIA/TIA-422-B, EIA/ TIA-485-A, причём EIA/ TIA-485-A работает исключительно в этом режиме.

4.3.2 Для работы в полудуплексном режиме значение параметра `loopback_chunk_size` должно равно значению параметра `packet_size`. Таким образом ретранслятор данных будет получать пакет данных целиком, а затем отправлять эти данные обратно.

Приложение А

(справочное)

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ

В настоящем документе используются следующие сокращения и обозначения:

Сокращение	Расшифровка
ПО	Программное обеспечение
UART	Universal Asynchronous Receiver Transmitter. Универсальный асинхронный последовательный приёмопередатчик
USART	Universal Synchronous Receiver Transmitter. Универсальный синхронный последовательный приёмопередатчик
TxD	Transmitted Data. Передаваемые данные, линия передачи данных.
RxD	Received Data. Принимаемые данные, линия приёма данных.
ПАК	Программно-аппаратный комплекс
ОС	Операционная система