

Общество с ограниченной ответственностью

Тэк Про Индастриз

Limited Liability Company

Tech Pro Industries

Руководство
Программное обеспечение
«Lazarus SoftW HDLC СТ»

Москва, 2023 г.

Содержание

1	Установка и запуск	3
2	Функциональное назначение	3
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
	Приложение А	11

1 Установка и запуск

- 1) Для скачивания программного обеспечения Lazarus SoftW HDLC СТ, 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 необходимая информация о Lazarus SoftW HDLC СТ. Под изображением находится ссылка для скачивания архива с дистрибутивом, нажмите на нее (рисунок 1). Введите пароль для скачивания архива.

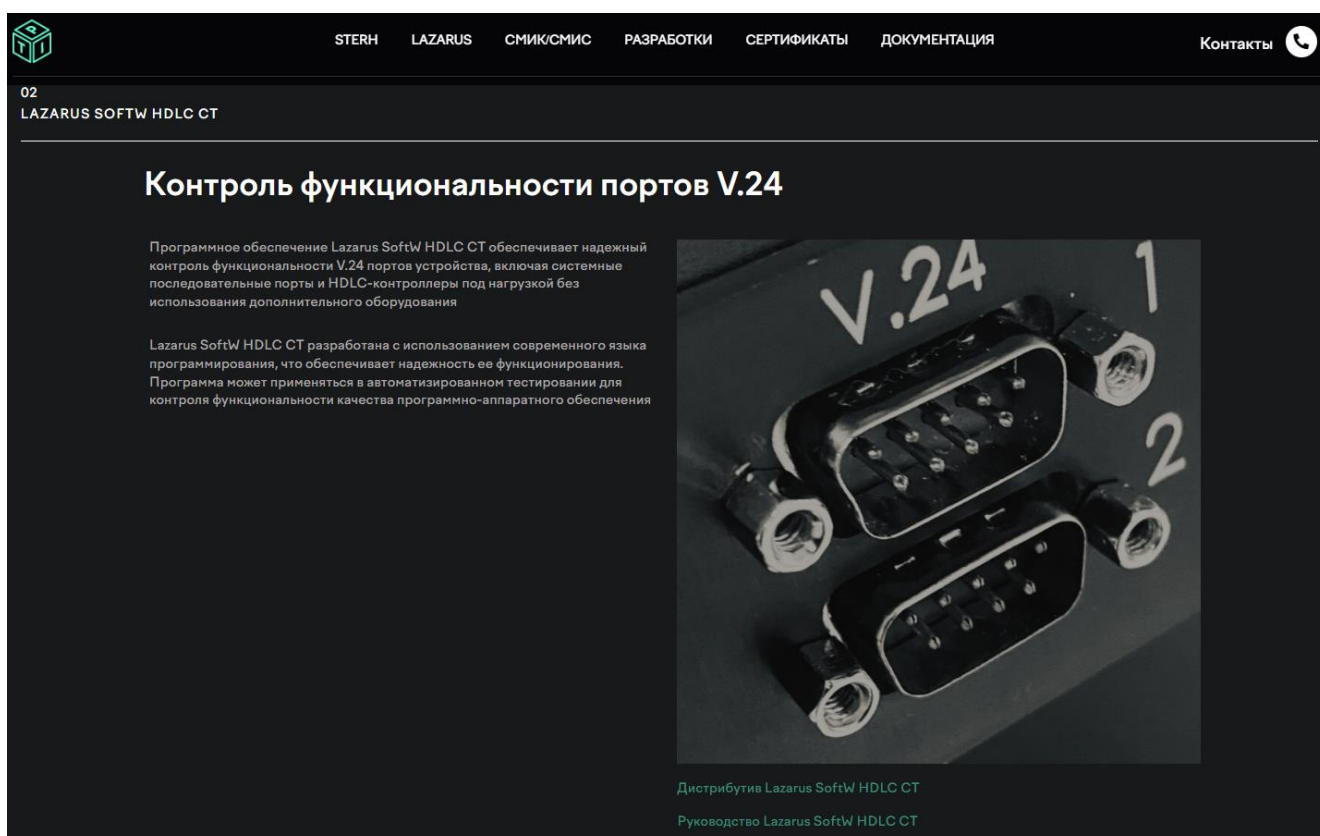


Рисунок 1 – Страница скачивания ПО Lazarus SoftW HDLC СТ

- 3) Распакуйте скачанный архив с Lazarus SoftW HDLC СТ, на персональном компьютере.
- 4) Убедитесь, что программно-аппаратный комплекс Lazarus IC и персональный компьютер находятся в одной локальной сети.
- 5) Для переноса Lazarus SoftW HDLC СТ на Lazarus IC необходимо в месте расположения программного обеспечения открыть окно терминала и перенести файл командой “scp”*.

*Перенос командой `scp`: `<scp './Lazarus_SoftW_HDLC_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.

- б) При выполнении переноса командой `scp` необходимо будет ввести пароль от Lazarus IC.

Пример команды для переноса в окне терминала приведен на рисунке 2.

```
C:\Users\Пользователь> scp Lazarus_SoftW_HDLC_CT testuser@192.168.83.125:/home/testuser
testuser@192.168.83.125's password:
Lazarus_SoftW_HDLC_CT                               100% 25MB 29.8MB/s 00:00
```

Рисунок 2 – Перенос установочного файла ПО Lazarus SoftW HDLC CT на Lazarus IC

2 Функциональное назначение

2.1 Классы решаемых задач

2.1.1 Продукт решает следующие классы задач:

- 1) Контроль функциональности портов V24, включая системные последовательные порты и HDLC-контроллеры, без использования дополнительного оборудования;
- 2) Вывод статистической информации об ошибках и отказах, обнаруженных в ходе тестирования.

2.1.2 Продукт может применяться в автоматизированном тестировании для контроля качества 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.

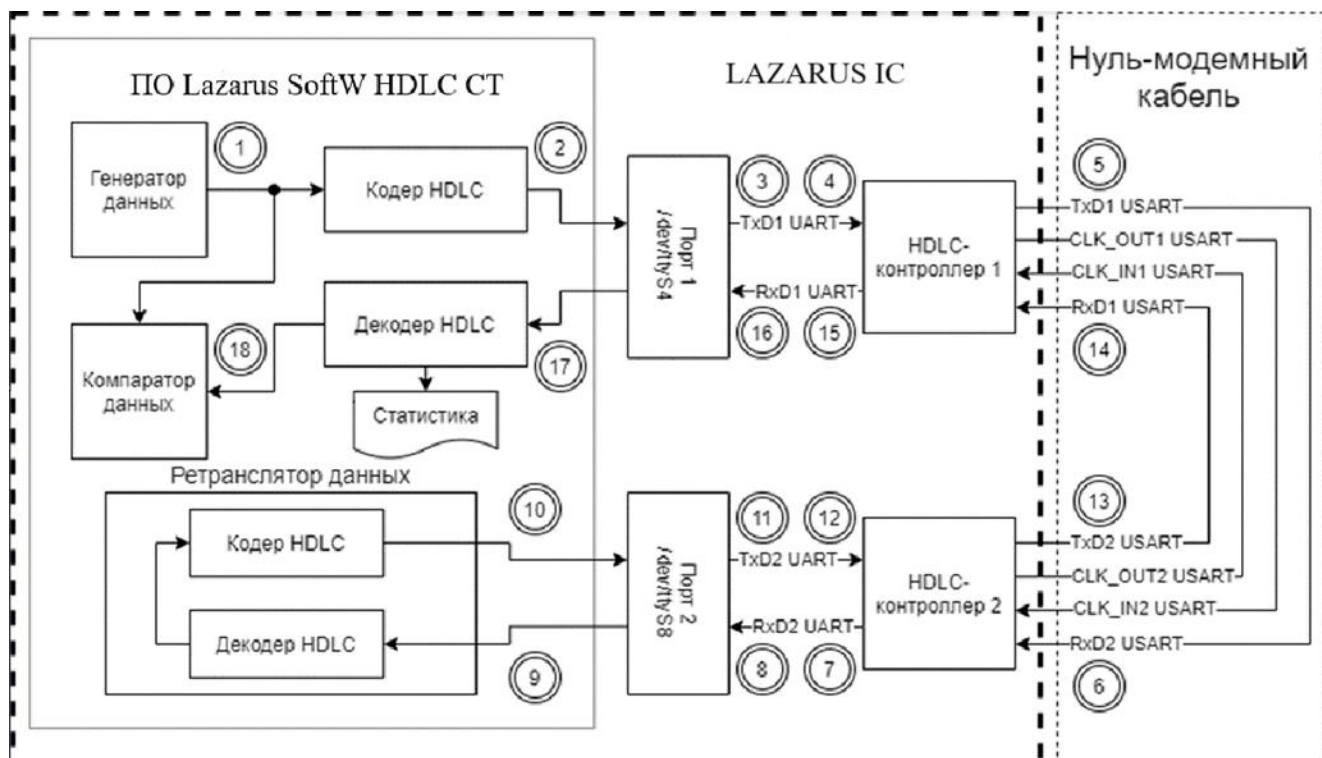


Рисунок 3 - Функциональная схема Lazarus SoftW HDLC CT

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 массив случайных тестовых данных случайной длины в диапазоне от 2 до 512 байтов и передаёт кодировщику HDLC для кодирования;

3.1.2.2 Закодированные тестовые данные передаются в последовательный Порт 1 для передачи;

3.1.2.3 Данные передаются по UART HDLC-контроллеру 1;

3.1.2.4 HDLC-контроллер 1 получает данные по UART;

3.1.2.5 HDLC-контроллер 1 синхронно передаёт полученные по UART данные в USART;

3.1.2.6 Через нуль-модемный кабель HDLC-контроллер 2 принимает данные по USART;

3.1.2.7 HDLC-контроллер 2 передаёт полученные по USART данные в UART Porta 2;

3.1.2.8 Последовательный Порт 2 принимает данные по UART;

3.1.2.9 Компонент ретранслятора данных получает порцию данных из Porta 2 и передаёт в декодер HDLC ретранслятора на обработку;

3.1.2.10 В случае успешного декодирования кадра HDLC, этот кадр повторно кодируется кодировщиком HDLC ретранслятора данных и перенаправляется на обратную передачу в Порт 2;

3.1.2.11 Порт 2 передаёт данные по UART HDLC-контроллеру 2;

3.1.2.12 HDLC-контроллер 2 получает данные по UART;

3.1.2.13 HDLC-контроллер 2 синхронно передаёт полученные по UART данные в USART;

3.1.2.14 Через нуль-модемный кабель HDLC-контроллер 1 принимает данные по USART;

3.1.2.15 HDLC-контроллер 1 передаёт полученные по USART данные в UART Porta 1;

3.1.2.16 Последовательный Порт 1 принимает данные по UART;

3.1.2.17 Полученные из Porta 1 данные поступают на вход декодера HDLC;

3.1.2.18 Если декодер раскодировал полный кадр HDLC, то данные кадра передаются в компаратор данных для сравнения с исходными тестовыми данными. Если расхождения в данных не обнаружено, то цикл повторяется. Если обнаружено расхождение в данных или кадр не получен в течение ожидаемого времени, фиксируется ошибка;

3.1.2.19 Программа может завершить работу либо при достижении максимального количества ошибок, либо при получении указанного количества кадров, 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. Перед началом использования программы надо выполнить команду `sudo su`, чтобы работать в режиме суперпользователя и вводить команды от него;

4.1.2 Исполняемый файл принимает следующие параметры командной строки:

--help – описание основных параметров;

```
testuser@astra-arm64:~$ sudo su
root@astra-arm64:/home/testuser# ./Lazarus_SoftW_HDLC_CT --help
General options:
--help                produce help message
--help-extended       produce help message for extended options
--device arg          name of serial device under test
--loopback arg        name of loopback serial device under test. It is
                      optionally can be set to /dev/null to use with some
                      external loopback device
```

--help-extended – описание дополнительных параметров;

```
root@astra-arm64:/home/testuser# ./Lazarus_SoftW_HDLC_CT --help-extended
Extended options:
--device_timeout arg (=1000)  serial device data read timeout, ms
--loopback_chunk_size arg (=32) size of loopback data chunks
--loopback_timeout arg (=200) loopback device data read timeout, ms. Do not
                              recommended to set value smaller then 20 ms
--errors_max arg (=0)        maximal number of errors to stop
--num_of_frames arg (=10000) number of frames
--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 качестве источника получаемых данных должен использоваться внешний программный продукт, значения параметров `loop_back_chunk_size` и `loopback_timeout` не будут оказывать эффекта;

(Если не задать других параметров, то процесс будет продолжаться до получения 10000 кадров. При этом каждые 5 минут будет формироваться мини-отчет, как на примере ниже)

```
root@astra-arm64:/home/testuser# ./Lazarus_SoftW_HDLC_CT --device=/dev/ttyS4 --loopback=/dev/ttyS8
Test start time: Tue Nov 14 14:48:09 2023
Testing...

=====
1263 frames were sent
Test elapsed time: 0h 5m 0s 86ms
Frames receiving total:
Overall:      1263
Malformed:   0
Error FCS:   0

=====
2519 frames were sent
Test elapsed time: 0h 10m 0s 318ms
Frames receiving total:
Overall:      2519
Malformed:   0
Error FCS:   0

=====
3805 frames were sent
Test elapsed time: 0h 15m 0s 177ms
Frames receiving total:
Overall:      3805
Malformed:   0
Error FCS:   0
```

--loopback_chunk_size – размер массива данных, которые получает за раз компонент ретранслятора данных;

--num_of_frames – максимальное количество кадров HDLC, после отправки которых программа закончит работу;

```
root@astra-arm64:/home/testuser# ./Lazarus_SoftW_HDLC_CT --device=/dev/ttyS4 --loopback=/dev/ttyS8 --loopback_chunk_size=100
--num_of_frames=1000
Test start time: Tue Nov 14 15:46:12 2023
Testing...

Stopping...

=====
1000 frames were sent
Test elapsed time: 0h 4m 32s 933ms
Frames receiving total:
Overall:      1000
Malformed:   0
Error FCS:   0
```

--loopback_timeout – время ожидания данных на порте компонента ретранслятора данных, мс;

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

```
root@astra-arm64:/home/testuser# ./Lazarus_SoftW_HDLC_CT --device=/dev/ttyS4 --loopback=/dev/ttyS8 --loopback_timeout=10
--test_time=300
Test start time: Tue Nov 14 15:58:37 2023
Testing...

=====
1279 frames were sent
Test elapsed time: 0h 5m 0s 205ms
Frames receiving total:
Overall:      1279
Malformed:    0
Error FCS:    0
```

--errors_max – максимальное количество ошибок, при превышении которого программа закончит работу;

```
root@astra-arm64:/home/testuser# ./Lazarus_SoftW_HDLC_CT --device=/dev/ttyS4 --loopback=/dev/ttyS8 --errors_max=1
--num_of_frames=1000
Test start time: Tue Nov 14 16:10:13 2023
Testing...

Stopping...

=====
1000 frames were sent
Test elapsed time: 0h 3m 46s 605ms
Frames receiving total:
Overall:      1000
Malformed:    0
Error FCS:    0
```

--device_timeout – время ожидания данных на порте отправителя тестовых данных, мс.

```
root@astra-arm64:/home/testuser# ./Lazarus_SoftW_HDLC_CT --device=/dev/ttyS4 --loopback=/dev/ttyS8 --device_timeout=100
--test_time=300
Test start time: Tue Nov 14 16:19:37 2023
Testing...

=====
1271 frames were sent
Test elapsed time: 0h 5m 0s 45ms
Frames receiving total:
Overall:      1271
Malformed:    0
Error FCS:    0
```

Приложение А

(справочное)

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ

В настоящем документе используются следующие сокращения и обозначения:

Сокращение	Расшифровка
ПО	Программное обеспечение
ОС	Операционная система
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.