

# **ИНСТРУКЦИЯ ПО ЗАПУСКУ ТЕСТОВ НА МОДУЛЯХ НА БАЗЕ 1892ВМ14Я**

**Версия  
11.03.2022**

## ОГЛАВЛЕНИЕ

<b>1</b>	<b>О документе</b>	<b>3</b>
<b>2</b>	<b>Подготовка модуля к тестированию</b>	<b>4</b>
<b>3</b>	<b>Тесты</b>	<b>5</b>
3.1	Audio . . . . .	5
3.2	Bluetooth . . . . .	6
3.3	CAN . . . . .	7
3.4	CPU . . . . .	9
3.5	CPUfreq . . . . .	10
3.6	Ethernet . . . . .	11
3.7	GPU EGL . . . . .	12
3.8	GPU Qt5 EGLFS . . . . .	13
3.9	HDMI-audio . . . . .	13
3.10	I2C . . . . .	14
3.11	IEEE 802.15.4 . . . . .	15
3.12	LEDs . . . . .	15
3.13	NAND . . . . .	16
3.14	PWM . . . . .	21
3.15	RAM . . . . .	22
3.16	RTC . . . . .	23
3.17	SD/MMC . . . . .	24
3.18	SDMA . . . . .	25
3.19	SPI FLASH . . . . .	25
3.20	System Sleep . . . . .	26
3.21	UART . . . . .	27
3.22	USB . . . . .	30
3.23	VPIN . . . . .	31
3.24	VPOUT . . . . .	31
3.25	VPU . . . . .	33
3.26	Watchdog timer . . . . .	34
3.27	Wi-Fi . . . . .	35

## 1. О ДОКУМЕНТЕ

Документ описывает тесты из состава дистрибутива ОС GNU/Linux на базе Buildroot 3.1 для модулей на базе 1892ВМ14Я.

Поддерживаются следующие модули:

- Салют-ЭЛ24Д1 r1.3;
- Салют-ЭЛ24Д1 r1.4;
- Салют-ЭЛ24Д1 r1.5;
- Салют-ЭЛ24Д1 r1.5 с установленным навигационным радиомодулем RF2Chan v2;
- Салют-ЭЛ24Д2 r1.1;
- Салют-ЭЛ24ОМ1 r1.1 с установленным Салют-ЭЛ24ПМ1 r1.1 или Салют-ЭЛ24ПМ1 r1.2;
- Салют-ЭЛ24ОМ1 r1.2 с установленным Салют-ЭЛ24ПМ1 r1.2, Салют-ЭЛ24ПМ2 r1.0 или Салют-ЭЛ24ПМ2 r1.1.

Описание каждого теста состоит из следующих секций:

1. «Применимость» — содержит список модулей, для которых применим данный тест.
2. «Назначение теста» — содержит назначение теста.
3. «Алгоритм теста» — содержит краткое описание алгоритма теста.
4. «Подготовка к тестированию» — содержит инструкции по подготовке и настройке дополнительного оборудования (секция может отсутствовать если отсутствует дополнительное оборудование).
5. «Запуск теста» — команда, которую необходимо ввести в терминал модуля для запуска теста.
6. «Ожидаемое время исполнения» — время, через которое ожидается успешное завершение теста.
7. «Пример результата теста» — пример вывода сообщений при успешном завершении теста<sup>1</sup>.
8. «Критерий прохождения теста» — список критериев, при выполнении которых тест считается пройденным успешно.

---

<sup>1</sup> При запуске тестов по терминалу UART сообщения ядра Linux выводятся в терминал. По умолчанию при запуске тестов через терминал SSH сообщения ядра Linux не выводятся. Для вывода сообщений ядра Linux в терминал SSH необходимо после логина выполнить команду `dmesg -n8`.

## 2. ПОДГОТОВКА МОДУЛЯ К ТЕСТИРОВАНИЮ

Перед запуском тестов необходимо подготовить модуль согласно главе «Запуск модуля» документа «Дистрибутив ОС GNU/Linux на базе Buildroot для 1892ВМ14Я. Руководство программиста».

В терминале модуля перед запуском необходимо установить регулятор масштабирования тактовой частоты ядер CPU в значение *performance*:

```
echo performance > /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor
```

Установку регулятора необходимо выполнять после каждой перезагрузки операционной системы.

## 3. ТЕСТЫ

### 3.1 Audio

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24ОМ1.

Назначение теста:

Для проверки записи звука через линейный вход Line In и воспроизведения звука через линейный выход Line Out модуля.

Алгоритм теста:

1. Записать исходный звуковой сигнал, поступающий на линейный вход Line In модуля, с линейного выхода Line Out ПЭВМ.
2. Воспроизвести записанный звуковой сигнал через линейный выход Line Out модуля.
3. Сравнить воспроизведенный звуковой сигнал с исходным.

Подготовка к тестированию:

1. Соединить кабелем линейный вход Line In модуля и линейный выход Line Out ПЭВМ.
2. Подключить наушники к линейному выходу Line Out модуля.

**Предупреждение:** При подключении наушников к линейному выходу Line Out модуля уровень громкости может быть низким.

3. Настроить модуль на запись с линейного входа Line In.

**Внимание:** По умолчанию модуль Салют-ЭЛ24ОМ1 настроен на запись с микрофона. Для настройки модуля на запись с линейного входа Line In выполнить команду `amixer -q cset name='Capture Mux' 1` в терминале модуля.

4. Настроить ПЭВМ (ОС, аудиоплеер) на воспроизведение звука через линейный выход Line Out.
5. Подготовить аудиофайл для воспроизведения на ПЭВМ.
6. Включить воспроизведение аудиофайла через линейный выход Line Out ПЭВМ.

Запуск теста:

```
arecord -f dat -d 30 test.wav  
aplay test.wav
```

Ожидаемое время исполнения: 60 секунд.

Критерий прохождения теста:

Тест считать успешно пройденным, если звуковой сигнал, воспроизведенный через линейный выход Line Out модуля, соответствует аудиофайлу, воспроизведенному на ПЭВМ.

## 3.2 Bluetooth

Применимость: Салют-ЭЛ24ОМ1.

Назначение теста:

Для проверки работоспособности Bluetooth. Для теста необходимы дополнительные устройства:

- Bluetooth-адаптер.

Алгоритм теста:

На модуле и ПЭВМ включается обнаружение устройства, производится сканирование устройств в зоне радиовидимости.

Подготовка к тестированию:

1. В терминале ПЭВМ выполнить команду для установки пакета BlueZ:

```
sudo yum install bluez
```

2. В терминале ПЭВМ выполнить:

```
sudo setcap 'cap_net_raw,cap_net_admin+eip' `which bluetoothctl`
```

3. Подключить Bluetooth-адаптер к ПЭВМ.

4. В терминале модуля выполнить:

```
btattach -B /dev/ttyS1 -P bcm &
```

Команда подключает последовательное устройство через UART HCI к стеку BlueZ на модуле.

5. В терминалах ПЭВМ и модуля выполнить команду и считать MAC-адреса ПЭВМ и модуля в поле Controller:

```
bluetoothctl
```

Команда запускает утилиту интерактивного управления интерфейсом Bluetooth. Дальнейшие команды вводятся в терминале утилиты.

6. В терминалах ПЭВМ и модуля выполнить:

```
power on
```

Команда конфигурирует на ПЭВМ и на модуле интерфейс hci0.

7. В терминалах ПЭВМ и модуля выполнить:

```
discoverable on
```

Команда включает обнаружение устройств.

Запуск теста:

В терминалах ПЭВМ и модуля выполнить:

```
scan on
```

Команда запускает поиск доступных устройств в зоне радиовидимости.

Ожидаемое время исполнения: около 15 секунд.

Пример результата теста:

```
Device AA:AA:AA:AA:AA:AA Name: mcom02
```

Критерий прохождения теста:

- в выводе команды в терминале модуля присутствует строка, содержащая MAC-адрес ПЭВМ,
- в выводе команды в терминале ПЭВМ присутствует строка, содержащая MAC-адрес модуля.

## 3.3 CAN

### 3.3.1 Тест передачи данных

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для проверки работоспособности шины CAN используется утилита `canfdtest`. Для теста необходимо дополнительное устройство — модуль Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, или Салют-ЭЛ24ОМ1 (далее — ответный модуль).

Алгоритм теста:

Проверяемый модуль генерирует кадры данных, передает их по шине и сравнивает с полученными. Ответный модуль ретранслирует принятые кадры, увеличивая значение каждого байта данных на единицу.

Подготовка к тестированию:

1. Подключить проверяемый и ответный модули по шине CAN. Для подключения соединить контакты *CAN-H*, *CAN-L* и *GND* разъемов CAN модулей (*XP1* для модулей Салют-ЭЛ24Д1; *XP2* для модулей Салют-ЭЛ24Д2, *XP6* для модулей Салют-ЭЛ24ОМ1).

**Предупреждение:** На модуле Салют-ЭЛ24Д1 r1.3 назначение выводов разъема CAN отличается от общепринятого: см. «Модуль Салют-ЭЛ24Д1 rev. 1.3. Руководство пользователя»

2. Подготовить проверяемый и ответный модули согласно главе «Подготовка модуля к тестированию».
3. В терминале ответного модуля выполнить команду: `canfdtest can0`.

Запуск теста:

```
timeout -t 15 -s KILL canfdtest -g -l 1024 can0
```

Ожидаемое время исполнения: 15 секунд.

**Внимание:** После завершения теста необходимо прервать выполнение команды `canfdtest` на ответном модуле нажатием `Ctrl+C` в терминале.

Пример результата теста:

```
interface = can0, family = 29, type = 3, proto = 1
```

```
Test messages sent and received: 1024
```

Критерий прохождения теста:

Тест считать успешно пройденным при выводе строки: `Test messages sent and received: 1024`.

### 3.3.2 Тест пропускной способности

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для проверки пропускной способности шины CAN используется утилита `canbusload`. Для теста необходимо дополнительное устройство — модуль Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, или Салют-ЭЛ24ОМ1 (далее — ответный модуль).

Алгоритм теста:

Проверяемый модуль генерирует и передает по шине кадры случайных данных, одновременно прослушивая шину для измерения пропускной способности. Ответный модуль подтверждает принятые кадры данных в соответствии со стандартом CAN.

Подготовка к тестированию:

1. Подключить проверяемый и ответный модули по шине CAN. Для подключения соединить контакты *CAN-H*, *CAN-L* и *GND* разъемов CAN модулей (*XP1* для модулей Салют-ЭЛ24Д1; *XP2* для модулей Салют-ЭЛ24Д2, *XP6* для модулей Салют-ЭЛ24ОМ1).

2. Подготовить проверяемый и ответный модули согласно главе «Подготовка модуля к тестированию».

Запуск теста:

```
timeout -t 15 -s KILL canbusload -t -b -e can0@125000 &  
timeout -t 15 -s KILL cangen -g 0 -p 1 can0
```

Ожидаемое время исполнения: 15 секунд.

Пример результата теста:

```
canbusload 2016-08-16 17:36:18 (exact bitstuffing)  
can0@125000 1171 110794 52344 88% |XXXXXXXXXXXXXXXXXX...|
```

Критерий прохождения теста:

Загрузка шины (пятое число в последней строке) должна быть более 80%.

### 3.4 CPU

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для проверки производительности двух ядер процессора СнК в целочисленных задачах используется утилита Coremark.

При выполнении не должно быть запущено других приложений и подключено устройств USB.

Запуск теста:

```
coremark
```

Ожидаемое время исполнения: 20 секунд.

Пример результата теста:

```
2K performance run parameters for coremark.  
CoreMark Size      : 666  
Total ticks       : 13311  
Total time (secs): 13.311000  
Iterations/Sec    : 6010.066862  
Iterations        : 80000  
Compiler version  : GCC5.2.0  
Compiler flags    : -D_LARGEFILE_SOURCE -D_LARGEFILE64_SOURCE -D_FILE_OFFSET_  
↳BITS=64 -Os  
Parallel PThreads : 2  
Memory location   : Please put data memory location here  
                   (e.g. code in flash, data on heap etc)  
seedcrc           : 0xe9f5  
[0]crclist        : 0xe714
```

(continues on next page)

(продолжение с предыдущей страницы)

```
[1]crclist      : 0xe714
[0]crcmatrix   : 0x1fd7
[1]crcmatrix   : 0x1fd7
[0]crcstate    : 0x8e3a
[1]crcstate    : 0x8e3a
[0]crcfinal    : 0x25b5
[1]crcfinal    : 0x25b5
Correct operation validated. See readme.txt for run and reporting rules.
CoreMark 1.0 : 6010.066862 / GCC5.2.0 -D_LARGEFILE_SOURCE -D_LARGEFILE64_SOURCE
-D_FILE_OFFSET_BITS=64 -Os / Heap / 2:PThreads
```

Критерий прохождения теста:

Значение поля `Iterations/Sec` должно быть не менее 5000.

### 3.5 CPUfreq

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для проверки производительности ядер процессора СнК при различных частотах используется скрипт `msom02-cpufreq-test.sh`.

При выполнении не должно быть запущено других приложений и подключено устройств USB.

Алгоритм теста:

1. Для governor'a `ondemand` скрипт считывает значения регистра `APLL` до запуска `coremark` (минимальное) и после (максимальное), сравнивает с ожидаемыми и выводит результат на экран.
2. Для governor'a `userspace` скрипт последовательно устанавливает каждое из доступных значений частоты, считывает значение регистра `APLL`, вычисляет значение частоты с помощью поля `coremark Iterations/Sec`, сравнивает с ожидаемыми и выводит результат на экран.
3. Если результат, полученный с помощью прямого чтения регистра `APLL`, совпадает со значением, установленным `CPUFreq`, на экран выводится сообщение "`APLL test successful`".
4. Если результат, полученный с помощью `Coremark`, отклоняется от значения, установленного `CPUFreq`, не более, чем на 5%, на экран выводится сообщение "`Coremark test successful`".

Запуск теста:

```
msom02-cpufreq-test.sh
```

Ожидаемое время выполнения: около 2 минут.

Пример результата теста:

```
Governor: ondemand
Frequency read from APLL: 312000, minimum, idle
APLL test successful
Frequency calculated from coremark results: 834016.
Coremark test successful
Frequency read from APLL: 816000, maximum, loaded
APLL test successful

Governor: userspace
Frequency read from APLL: 312000
APLL test successful
Frequency calculated from coremark results: 318116.
Coremark test successful
Frequency read from APLL: 408000
APLL test successful
Frequency calculated from coremark results: 416114.
Coremark test successful
Frequency read from APLL: 816000
APLL test successful
Frequency calculated from coremark results: 834552.
Coremark test successful

TEST PASSED
```

Критерий прохождения теста:

Тест считать успешно пройденным при выводе строки TEST PASSED.

## 3.6 Ethernet

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для измерения скорости ввода-вывода через Ethernet используется утилита `iperf`. Для теста необходимо дополнительное устройство — ПК с ОС Linux и пакетом `iperf` версии 2.0.5.

Подготовка к тестированию:

1. Подключить модуль в сеть Ethernet.
2. Подключить ПК в сеть Ethernet общую с модулем.
3. Выяснить текущий IP-адрес модуля:
  1. В терминале модуля ввести команду `ifconfig eth0`.
  2. Считать IP-адрес модуля из поля `inet` вывода команды.

Запуск теста:

На модуле запустить сервер:

```
iperf --server
```

На ПК запустить клиент:

```
iperf --time=5 --client адрес-сервера
```

где адрес-сервера — значение IP-адреса модуля.

Ожидаемое время исполнения: не более 5 секунд.

Пример результата теста:

```
[ ID] Interval      Transfer    Bandwidth
[  3] 0.0-5.0 sec   54 MBytes  85.4 Mbits/sec
```

Критерий прохождения теста:

Значение поля Bandwidth не менее 80 Mbits/sec для модулей Салют-ЭЛ24Д1, Салют-ЭЛ24Д2; не менее 340 Mbits/sec для модулей Салют-ЭЛ24ОМ1.

### 3.7 GPU EGL

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для измерения производительности графического ускорителя используется утилита cube из состава тестов драйвера Mali.

Подготовка к тестированию:

1. Подключить HDMI-монитор к соответствующему разъёму модуля.
2. Для модулей Салют-ЭЛ24Д1 или Салют-ЭЛ24Д2 выполнить загрузку драйвера voutfb:

```
modprobe voutfb
```

Запуск теста:

```
cd /opt/mali/cube
./cube
```

Ожидаемое время исполнения: бесконечно. Для завершения теста необходимо нажать Ctrl+C.

Пример результата теста:

```
FPS:      75.1
FPS:      75.5
FPS:      76.1
FPS:      75.6
FPS:      75.9
```

Критерий прохождения теста:

1. Средняя частота отрисовки кадров больше 75 кадров в секунду.
2. На HDMI-мониторе наблюдать вращающийся объект.

### 3.8 GPU Qt5 EGLFS

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для проверки работоспособности приложений Qt на графическом ускорителе используется утилита `helloworld` из состава пакета `Qt5base`.

Подготовка к тестированию:

Подключить HDMI-монитор к соответствующему разъёму модуля.

Запуск теста:

```
export QT_QPA_EGLFS_INTEGRATION=eglfs_mali
/usr/lib/qt/examples/opengl/helloworld/helloworld
```

Ожидаемое время исполнения: бесконечно. Для завершения теста необходимо нажать `Ctrl+C`.

Критерий прохождения теста:

На HDMI-мониторе наблюдать вращающийся трехмерный логотип Qt.

### 3.9 HDMI-audio

Применимость: Салют-ЭЛ24ОМ1.

Назначение теста:

Для проверки воспроизведения звука через интерфейс HDMI на мониторе.

Алгоритм теста:

1. Загрузить аудиофайл на модуль.
2. Воспроизвести аудиофайл через интерфейс HDMI модуля.

Подготовка к тестированию:

1. Соединить кабелем HDMI выход модуля и монитор.
2. Загрузить на модуль аудиофайл.
3. Определить идентификатор соответствующий аудиокарте HDMI. Идентификатор аудиокарты это значение поля `card` в выводе команды:

```
aplay -l
```

Запуск теста:

```
aplay -D plughw:<card_id>,0 file.wav,
```

где <card\_id>— идентификатор аудиокарты HDMI, file.wav — загруженный аудио-файл.

Ожидаемое время исполнения: 35 секунд.

Критерий прохождения теста:

Тест считать успешно пройденным, если звуковой сигнал, воспроизведенный на HDMI-мониторе, соответствует загруженному аудиофайлу.

### 3.10 I2C

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2.

Назначение теста:

Для тестирования скорости чтения и записи на шине I2C СнК используется утилита `i2cdump`.

Алгоритм теста:

Утилита считывает 256 регистров HDMI-контроллера, подключенного на частоте 100 КГц, через контроллер I2C1 СнК и выводит время прохождения теста. Скорость работы контроллера I2C1 на модуле — 100 Кб/с.

Размер переданных транзакций в тесте составляет:

$9 \text{ (бит на кадр)} * 3 \text{ (кадра — 2 с адресом, 1 с данными)} * 256 \text{ (количество регистров, считываемое утилитой)} * 256 \text{ (запусков утилиты)} = 1\,769\,472 \text{ бит.}$

Для вычисления скорости необходимо разделить размер переданных транзакций на время прохождения теста.

Запуск теста:

```
time sh -c 'for i in `seq 1 256`; do i2cdump -y 1 0x4c b >/dev/null ; done'
```

Ожидаемое время исполнения: около 30 секунд.

Пример результата теста:

```
real    0m 28.68s
user    0m 0.33s
sys     0m 2.31s
```

Критерий прохождения теста:

Число  $(1769472 / \langle \text{real-time} \rangle) / 100000$  больше 0.5, где <real-time> — значение поля `real` из вывода теста в терминал. Т.о. реальная пропускная способность I2C больше 50% от теоретической. Например, для данного теста  $1769472 / 28.68 / 100000 = 0.61 > 0.5$

### 3.11 IEEE 802.15.4

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2.

Назначение теста:

Для проверки связи по стандарту IEEE 802.15.4 (нижние уровни для семейства протоколов ZigBee) используется утилита `ping6`. Для теста необходимо дополнительное устройство — модуль Салют-ЭЛ24Д1, или Салют-ЭЛ24Д2 (далее — ответный модуль).

Подготовка к тестированию:

1. Установить антенны на разъемы *ZigBee* проверяемого и ответного модулей.
2. Подготовить проверяемый и ответный модули согласно главе «Подготовка модуля к тестированию». При подготовке модулей настроить сетевой интерфейс 6LoWPAN, задав для проверяемого модуля адрес `fc00::1` (подробнее см. главу «Настройка 6LoWPAN» документа «Дистрибутив ОС GNU/Linux на базе Buildroot для 1892BM14Я. Руководство программиста»). Для ответного модуля по умолчанию задан адрес `fc00::2`.

Запуск теста:

```
ping6 -I lowpan0 -c 5 fc00::2
```

Ожидаемое время исполнения: около 5 секунд.

Пример результата теста:

```
PING fc00::2 (fc00::2): 56 data bytes
64 bytes from fc00::2: seq=0 ttl=64 time=35.127 ms
64 bytes from fc00::2: seq=1 ttl=64 time=36.519 ms
64 bytes from fc00::2: seq=2 ttl=64 time=33.841 ms
64 bytes from fc00::2: seq=3 ttl=64 time=36.526 ms
64 bytes from fc00::2: seq=4 ttl=64 time=33.849 ms

--- fc00::2 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 33.841/35.172/36.526 ms
```

Критерий прохождения теста:

Значение поля `packet loss` (процент потери пакетов) равно 0%.

### 3.12 LEDs

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24ОМ1.

Назначение теста:

Для проверки работоспособности светодиодов используется утилита `ledtest`.

Алгоритм теста:

- выключить все светодиоды на 1 с;

- для каждого светодиода выполнить: включить светодиод, через 1 с выключить светодиод;
- включить все светодиоды;
- выключить все светодиоды через 1 с;
- перевести все пины светодиодов в состояние как после включения модуля.

Запуск теста:

```
ledtest
```

Ожидаемое время исполнения: около 10 секунд.

Критерий прохождения теста:

Светодиоды работают согласно алгоритму утилиты.

## 3.13 NAND

### 3.13.1 mtd\_pagetest

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для технологического тестирования NAND-флеш памяти модуля используется модуль ядра `mtd_pagetest`.

**Предупреждение:** Запуск данного теста приведёт к потере данных на чипе NAND.

Запуск теста:

```
modprobe mtd_pagetest dev=0
```

Ожидаемое время исполнения: около 3 часов.

Пример результата теста:

```
[ 138.955327]
[ 138.956834] =====
[ 138.962592] mtd_pagetest: MTD device: 0
[ 138.966489] mtd_pagetest: MTD device size 4294967296, eraseblock
size 1048576, page size 4096, count of eraseblocks 4096, pages per
eraseblock 256, OOB size 224
[ 138.980874] mtd_test: scanning for bad eraseblocks
[ 138.985713] mtd_test: block 90 is bad
[ 138.989372] mtd_test: block 91 is bad
[ 138.993520] mtd_test: block 4092 is bad
[ 138.997351] mtd_test: block 4093 is bad
[ 139.001180] mtd_test: block 4094 is bad
```

(continues on next page)

(продолжение с предыдущей страницы)

```
[ 139.005028] mtd_test: block 4095 is bad
[ 139.008862] mtd_test: scanned 4096 eraseblocks, 6 are bad
[ 139.014261] mtd_pagetest: erasing whole device
[ 149.217664] mtd_pagetest: erased 4096 eraseblocks
[ 149.222367] mtd_pagetest: writing whole device
[ 149.653784] mtd_pagetest: written up to eraseblock 0
[ 258.841520] mtd_pagetest: written up to eraseblock 256
[ 368.921588] mtd_pagetest: written up to eraseblock 512
[ 479.057226] mtd_pagetest: written up to eraseblock 768
[ 589.401764] mtd_pagetest: written up to eraseblock 1024
[ 700.129340] mtd_pagetest: written up to eraseblock 1280
[ 810.430958] mtd_pagetest: written up to eraseblock 1536
[ 920.510587] mtd_pagetest: written up to eraseblock 1792
[ 1030.427427] mtd_pagetest: written up to eraseblock 2048
[ 1140.465299] mtd_pagetest: written up to eraseblock 2304
[ 1250.483171] mtd_pagetest: written up to eraseblock 2560
[ 1360.513440] mtd_pagetest: written up to eraseblock 2816
[ 1470.596688] mtd_pagetest: written up to eraseblock 3072
[ 1580.443776] mtd_pagetest: written up to eraseblock 3328
[ 1690.302049] mtd_pagetest: written up to eraseblock 3584
[ 1800.225171] mtd_pagetest: written up to eraseblock 3840
[ 1908.080495] mtd_pagetest: written 4096 eraseblocks
[ 1908.085295] mtd_pagetest: verifying all eraseblocks
[ 1909.169019] mtd_pagetest: verified up to eraseblock 0
[ 2183.178217] mtd_pagetest: verified up to eraseblock 256
[ 2459.348834] mtd_pagetest: verified up to eraseblock 512
[ 2735.518683] mtd_pagetest: verified up to eraseblock 768
[ 3011.688729] mtd_pagetest: verified up to eraseblock 1024
[ 3287.857647] mtd_pagetest: verified up to eraseblock 1280
[ 3564.028262] mtd_pagetest: verified up to eraseblock 1536
[ 3840.199111] mtd_pagetest: verified up to eraseblock 1792
[ 4116.369468] mtd_pagetest: verified up to eraseblock 2048
[ 4392.540322] mtd_pagetest: verified up to eraseblock 2304
[ 4668.710759] mtd_pagetest: verified up to eraseblock 2560
[ 4944.881290] mtd_pagetest: verified up to eraseblock 2816
[ 5221.052029] mtd_pagetest: verified up to eraseblock 3072
[ 5497.222882] mtd_pagetest: verified up to eraseblock 3328
[ 5773.393804] mtd_pagetest: verified up to eraseblock 3584
[ 6049.564345] mtd_pagetest: verified up to eraseblock 3840
[ 6320.337449] mtd_pagetest: verified 4096 eraseblocks
[ 6320.342356] mtd_pagetest: crosstest
[ 6320.347278] mtd_pagetest: reading page at 0x0
[ 6320.352387] mtd_pagetest: reading page at 0xffbfff00
[ 6320.358056] mtd_pagetest: reading page at 0x0
[ 6320.363145] mtd_pagetest: verifying pages read at 0x0 match
[ 6320.368752] mtd_pagetest: crosstest ok
[ 6320.372538] mtd_pagetest: erasescrosstest
[ 6320.376463] mtd_pagetest: erasing block 0
[ 6320.383845] mtd_pagetest: writing 1st page of block 0
```

(continues on next page)

(продолжение с предыдущей страницы)

```
[ 6320.390027] mtd_pagetest: reading 1st page of block 0
[ 6320.395832] mtd_pagetest: verifying 1st page of block 0
[ 6320.401103] mtd_pagetest: erasing block 0
[ 6320.408484] mtd_pagetest: writing 1st page of block 0
[ 6320.414695] mtd_pagetest: erasing block 4091
[ 6320.422533] mtd_pagetest: reading 1st page of block 0
[ 6320.428294] mtd_pagetest: verifying 1st page of block 0
[ 6320.433571] mtd_pagetest: erasescrosstest ok
[ 6320.437755] mtd_pagetest: erasetest
[ 6320.441263] mtd_pagetest: erasing block 0
[ 6320.448644] mtd_pagetest: writing 1st page of block 0
[ 6320.454844] mtd_pagetest: erasing block 0
[ 6320.462224] mtd_pagetest: reading 1st page of block 0
[ 6320.467983] mtd_pagetest: verifying 1st page of block 0 is all 0xff
[ 6320.474312] mtd_pagetest: erasetest ok
[ 6320.478064] mtd_pagetest: finished with 0 errors
[ 6320.482755] =====
```

Критерий прохождения теста:

В предпоследней строке вывода указано 0 ошибок: «mtd\_pagetest: finished with 0 errors».

### 3.13.2 mtd\_speedtest

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для измерения скорости ввода-вывода NAND-флеш памяти используется модуль ядра mtd\_speedtest.

**Предупреждение:** Запуск данного теста приведёт к потере данных на чипе NAND.

Запуск теста:

```
modprobe mtd_speedtest dev=0 count=128
```

Ожидаемое время исполнения: около 8 минут

Пример результата теста:

```
[ 83.724711]
[ 83.726215] =====
[ 83.731956] mtd_speedtest: MTD device: 0    count: 128
[ 83.737128] mtd_speedtest: MTD device size 4294967296, eraseblock
size 2097152, page size 8192, count of eraseblocks 2048, pages per
eraseblock 256, 00B size 640
[ 83.772247] mtd_test: scanning for bad eraseblocks
```

(continues on next page)

(продолжение с предыдущей страницы)

```
[ 83.777084] mtd_test: scanned 128 eraseblocks, 0 are bad
[ 84.360363] mtd_speedtest: testing eraseblock write speed
[ 174.750650] mtd_speedtest: eraseblock write speed is 2900 KiB/s
[ 174.756573] mtd_speedtest: testing eraseblock read speed
[ 217.020337] mtd_speedtest: eraseblock read speed is 6203 KiB/s
[ 217.599805] mtd_speedtest: testing page write speed
[ 307.963326] mtd_speedtest: page write speed is 2901 KiB/s
[ 307.968726] mtd_speedtest: testing page read speed
[ 350.338617] mtd_speedtest: page read speed is 6187 KiB/s
[ 350.915537] mtd_speedtest: testing 2 page write speed
[ 441.039789] mtd_speedtest: 2 page write speed is 2908 KiB/s
[ 441.045375] mtd_speedtest: testing 2 page read speed
[ 483.391012] mtd_speedtest: 2 page read speed is 6191 KiB/s
[ 483.396497] mtd_speedtest: Testing erase speed
[ 483.969965] mtd_speedtest: erase speed is 460710 KiB/s
[ 483.975117] mtd_speedtest: Testing 2x multi-block erase speed
[ 484.551902] mtd_speedtest: 2x multi-block erase speed is 458293 KiB/s
[ 484.558342] mtd_speedtest: Testing 4x multi-block erase speed
[ 485.134180] mtd_speedtest: 4x multi-block erase speed is 459096 KiB/s
[ 485.140632] mtd_speedtest: Testing 8x multi-block erase speed
[ 485.715748] mtd_speedtest: 8x multi-block erase speed is 460710 KiB/s
[ 485.722199] mtd_speedtest: Testing 16x multi-block erase speed
[ 486.298030] mtd_speedtest: 16x multi-block erase speed is 459901 KiB/s
[ 486.304571] mtd_speedtest: Testing 32x multi-block erase speed
[ 486.879625] mtd_speedtest: 32x multi-block erase speed is 460710 KiB/s
[ 486.886164] mtd_speedtest: Testing 64x multi-block erase speed
[ 487.459118] mtd_speedtest: 64x multi-block erase speed is 461521 KiB/s
[ 487.465656] mtd_speedtest: finished
[ 487.469200] =====
```

Критерии прохождения теста:

1. Скорость записи eraseblock должна быть больше 2100 KiB/s.
2. Скорость чтения eraseblock должна быть больше 5100 KiB/s.
3. Скорость стирания должна быть больше 270000 KiB/s.

### 3.13.3 UBIFS

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для проверки работоспособности файловой системы UBIFS на чипе NAND модуля используется утилита `fiо`. Для создания, монтирования ФС и запуска утилиты используется скрипт `ubitest.sh`.

Алгоритм теста:

1. Форматировать устройство в том UBI с помощью утилиты `ubiformat`.
2. Создать раздел UBIFS максимального размера на устройстве.

3. Монтировать раздел в точку монтирования /mnt/ubifs.
4. Записать и верифицировать указанный в аргументах объём данных с помощью утилиты fio.
5. Очистить раздел и размонтировать его.

**Предупреждение:** Запуск данного теста приведёт к потере данных на чипе NAND.

Запуск теста:

```
ubitest.sh /dev/mtd0 800
```

Ожидаемое время исполнения: около 3 минут.

Пример результата теста:

```
1..3
[ 102.134366] ubi0: attaching mtd0
[ 107.408961] ubi0: scanning is finished
[ 107.426138] ubi0: attached mtd0 (name "arasan_nfc", size 2048 MiB)
[ 107.432337] ubi0: PEB size: 524288 bytes (512 KiB), LEB size: 516096 bytes
[ 107.439205] ubi0: min./max. I/O unit sizes: 4096/4096, sub-page size 4096
[ 107.445997] ubi0: VID header offset: 4096 (aligned 4096), data offset: 8192
[ 107.452961] ubi0: good PEBs: 4081, bad PEBs: 15, corrupted PEBs: 0
[ 107.459131] ubi0: user volume: 0, internal volumes: 1, max. volumes count: 128
[ 107.466352] ubi0: max/mean erase counter: 10/7, WL threshold: 4096, image
↳sequence number: 1771060986
[ 107.475572] ubi0: available PEBs: 4012, total reserved PEBs: 69, PEBs reserved
↳for bad PEB handling: 65
[ 107.484968] ubi0: background thread "ubi_bgt0d" started, PID 176
[ 107.547562] UBIFS (ubi0:0): default file-system created
[ 107.553590] UBIFS (ubi0:0): background thread "ubifs_bgt0_0" started, PID 182
[ 107.818682] UBIFS (ubi0:0): UBIFS: mounted UBI device 0, volume 0, name "full_
↳ubi"
[ 107.826275] UBIFS (ubi0:0): LEB size: 516096 bytes (504 KiB), min./max. I/O
↳unit sizes: 4096 bytes/4096 bytes
[ 107.836209] UBIFS (ubi0:0): FS size: 2064900096 bytes (1969 MiB, 4001 LEBs),
↳journal size 33546240 bytes (31 MiB, 65 LEBs)
[ 107.847259] UBIFS (ubi0:0): reserved for root: 4952683 bytes (4836 KiB)
[ 107.853891] UBIFS (ubi0:0): media format: w4/r0 (latest is w4/r0), UUID
↳07FF4EB6-C3B4-4FA1-B1A3-D149F334F5E8, small LPT model
ok 1 - Create and mount UBIFS partition
ok 2 - Write and verify with fio
[ 496.469464] UBIFS (ubi0:0): un-mount UBI device 0
[ 496.474210] UBIFS (ubi0:0): background thread "ubifs_bgt0_0" stops
[ 496.531859] ubi0: detaching mtd0
[ 496.537660] ubi0: mtd0 is detached
[ 496.544388] ubi0: attaching mtd0
[ 501.832063] ubi0: scanning is finished
```

(continues on next page)

(продолжение с предыдущей страницы)

```
[ 501.851497] ubi0: attached mtd0 (name "arasan_nfc", size 2048 MiB)
[ 501.857688] ubi0: PEB size: 524288 bytes (512 KiB), LEB size: 516096 bytes
[ 501.864594] ubi0: min./max. I/O unit sizes: 4096/4096, sub-page size 4096
[ 501.871396] ubi0: VID header offset: 4096 (aligned 4096), data offset: 8192
[ 501.878353] ubi0: good PEBs: 4081, bad PEBs: 15, corrupted PEBs: 0
[ 501.884549] ubi0: user volume: 1, internal volumes: 1, max. volumes count: 128
[ 501.891783] ubi0: max/mean erase counter: 11/7, WL threshold: 4096, image
↳sequence number: 1771060986
[ 501.901009] ubi0: available PEBs: 0, total reserved PEBs: 4081, PEBs reserved
↳for bad PEB handling: 65
[ 501.910324] ubi0: background thread "ubi_bgt0d" started, PID 192
[ 501.915863] UBIFS (ubi0:0): background thread "ubifs_bgt0_0" started, PID 194
[ 502.314263] UBIFS (ubi0:0): UBIFS: mounted UBI device 0, volume 0, name "full_
↳ubi"
[ 502.321852] UBIFS (ubi0:0): LEB size: 516096 bytes (504 KiB), min./max. I/O
↳unit sizes: 4096 bytes/4096 bytes
[ 502.331782] UBIFS (ubi0:0): FS size: 2064900096 bytes (1969 MiB, 4001 LEBs),
↳journal size 33546240 bytes (31 MiB, 65 LEBs)
[ 502.342829] UBIFS (ubi0:0): reserved for root: 4952683 bytes (4836 KiB)
[ 502.349444] UBIFS (ubi0:0): media format: w4/r0 (latest is w4/r0), UUID
↳07FF4EB6-C3B4-4FA1-B1A3-D149F334F5E8, small LPT model
[ 506.869158] UBIFS (ubi0:0): un-mount UBI device 0
[ 506.873897] UBIFS (ubi0:0): background thread "ubifs_bgt0_0" stops
[ 506.897651] ubi0: detaching mtd0
[ 506.904611] ubi0: mtd0 is detached
ok 3 - Test master node corruption after remount
TEST PASSED
```

Критерии прохождения теста:

Последняя строка вывода равна «TEST PASSED».

## 3.14 PWM

Применимость: Салют-ЭЛ24ОМ1.

Назначение теста:

Для проверки работоспособности PWM и возможности изменения яркости подсветки LCD на модуле.

Алгоритм теста:

1. Вывести тестовое изображение на LCD-монитор.
2. Установить яркость подсветки 95 процентов.
3. Установить яркость подсветки 5 процентов.

Подготовка к тестированию:

1. Подключить к разъему XS7 LCD-монитор EJ070NA-01J.

2. Настроить вывод изображения на LCD-монитор EJ070NA-01J согласно пункту «Вывод изображения на LCD-монитор» главы «Настройка ОС» документа «Дистрибутив ОС GNU/Linux на базе Buildroot для 1892BM14Я. Руководство программиста».

Запуск теста:

1. Вывести тестовое изображение на LCD-монитор:

```
nohup ffmpeg -f lavfi -i testsrc=duration=30.0:size=1024x600 -f fbdev -pix_
↔fmt bgra /dev/fb0 > /dev/null 2>&1 &
```

2. Установить яркость подсветки на 95 процентов:

```
echo 95 > /sys/class/backlight/backlight/brightness
```

3. Установить яркость подсветки на 5 процентов:

```
echo 5 > /sys/class/backlight/backlight/brightness
```

Критерий прохождения теста:

При вводе `echo 95 ...` яркость LCD-экрана должна отличаться от яркости при вводе команды `echo 5 ....`

## 3.15 RAM

### 3.15.1 Memtester

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для технологического тестирования динамической памяти модуля используется утилита `memtester`. С данными аргументами выделяется участок ОЗУ DDR размером в 512 МиБ.

Запуск теста:

```
memtester 512M 1
```

Ожидаемое время исполнения: не более 60 минут.

Пример результата теста:

```
Loop 1/1:
Stuck Address      : ok
Random Value       : ok
Compare XOR        : ok
Compare SUB        : ok
Compare MUL        : ok
Compare DIV        : ok
Compare OR         : ok
Compare AND        : ok
```

(continues on next page)

(продолжение с предыдущей страницы)

```
Sequential Increment: ok
Solid Bits           : ok
Block Sequential     : ok
Checkerboard         : ok
Bit Spread           : ok
Bit Flip             : ok
Walking Ones        : ok
Walking Zeroes      : ok
8-bit Writes         : ok
16-bit Writes       : ok
Done.
```

Критерий прохождения теста:

Каждый алгоритм тестирования завершился успешно: строка завершается словом "ок".

### 3.15.2 Ramspeed

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для измерения скорости чтения и записи в динамическую память модуля используется утилита `gamspeed`.

Запуск теста:

```
gamspeed -b 3
```

Ожидаемое время исполнения: менее 6 минут.

Пример результата теста:

```
RAMspeed (GENERIC) v2.6.0 by Rhett M. Hollander and Paul V. Bolotoff,
2002-09

8Gb per pass mode

INTEGER Copy:      425.15 MB/s
INTEGER Scale:    415.07 MB/s
INTEGER Add:      391.94 MB/s
INTEGER Triad:    386.52 MB/s
---
INTEGER AVERAGE: 404.67 MB/s
```

Критерий прохождения теста:

Значение поля "AVERAGE" составляет не менее 400 MB/s.

### 3.16 RTC

Применимость: Салют-ЭЛ24ОМ1.

Назначение теста:

Для проверки установки и чтения времени на микросхеме RTC.

Подготовка к тестированию:

Необходимо установить батарейку CR2032 в отсек GB1 на модуле Салют-ЭЛ24ОМ1.

Алгоритм теста:

1. Установить текущее время в RTC.
2. Переключить питание на модуле.
3. Сверить время на модуле и текущее время.

Запуск теста:

1. Установить системное время модуля в соответствии с текущим временем:

```
date -s "yyyy-mm-dd hh:mm:ss"
```

2. Сохранить системное время модуля в микросхеме RTC:

```
hwclock -f /dev/rtc0 -w
```

3. Отключить питание на модуле. Подождать 5 с. Включить питание на модуле.

4. Считать время на модуле:

```
date
```

5. Вычислить разницу между текущим временем и временем на модуле.

Критерий прохождения теста:

Время на модуле отличается от текущего времени менее чем на 2 с.

### 3.17 SD/MMC

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для измерения скорости чтения SD/MMC-карты используется утилита `hdparm`.

Запуск теста:

```
hdparm -t /dev/mmcblk0p1
```

Ожидаемое время исполнения: около 3 секунд.

Пример результата теста:

```
Timing buffered disk reads: 34 MB in 3.16 seconds = 10.76 MB/sec
```

Критерий прохождения теста:

Скорость чтения (последнее число в последней строке вывода) не менее 8.00 MB/sec для модулей Салют-ЭЛ24Д1, Салют-ЭЛ24Д2; не менее 18.00 MB/sec для модулей Салют-ЭЛ24ОМ1.

### 3.18 SDMA

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Проверка работоспособности SDMA.

Запуск теста:

```
dmatestcontig.sh
```

Ожидаемое время исполнения: около 5 секунд.

Пример результата теста:

```
[ 81.562775] dmatestcontig: 2000 iteration by 4194304 bytes takes 6689554 nsecs: [
↔1195 MiB/s
TEST PASSED
```

Критерий прохождения теста:

Вывод строки TEST PASSED.

### 3.19 SPI FLASH

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для тестирования скорости чтения флеш-памяти, подключенной по SPI, используется утилита `mtd_debug`. Утилита считывает заданную, смещением и размером — область памяти. В данном тесте читается вся область SPI флеш-памяти.

Запуск теста:

```
time mtd_debug read /dev/mtd/by-name/spi0.0 0 4194304 /dev/null
```

Пример результата теста:

```
Copied 4194304 bytes from address 0x00000000 in flash to /dev/null
real    0m 1.47s
user    0m 0.00s
sys     0m 0.03s
```

Критерий прохождения теста:

1. Тест завершается без ошибок.

2. Значение в поле `real`— менее 2 секунд.

## 3.20 System Sleep

Применимость: Салют-ЭЛ24ОМ1.

Назначение теста:

Для тестирования перехода в состояние сна используется утилита `rtcwake`.

Алгоритм теста:

1. Настройка таймера RTC модуля на срабатывание через заданный интервал;
2. Перевод ОС в состояние `standby`;
3. Срабатывание таймера RTC модуля;
4. Пробуждение ОС и вывод из состояния `standby`.

Запуск теста:

```
rtcwake -s 2 -m standby && echo TEST PASSED
```

Ожидаемое время исполнения: 3 секунды.

Пример результата теста:

```
rtcwake: assuming RTC uses UTC ...
rtcwake: wakeup from "standby" using /dev/rtc0 at Mon Jun  4 10:58:44 2018
PM: Syncing filesystems ... done.
Freezing user space processes ... (elapsed 0.001 seconds) done.
Freezing remaining freezable tasks ... (elapsed 0.002 seconds) done.
PM: suspend of devices complete after 147.175 msecs
PM: late suspend of devices complete after 1.274 msecs
PM: noirq suspend of devices complete after 1.215 msecs
Disabling non-boot CPUs ...
CPU1: shutdown
Enabling non-boot CPUs ...
CPU1 is up
PM: noirq resume of devices complete after 0.766 msecs
PM: early resume of devices complete after 0.788 msecs
PM: resume of devices complete after 303.958 msecs
Restarting tasks ... done.
TEST PASSED
```

Критерий прохождения теста:

Тест считать успешно пройденным при выводе строки `TEST PASSED`.

## 3.21 UART

### 3.21.1 RS-232

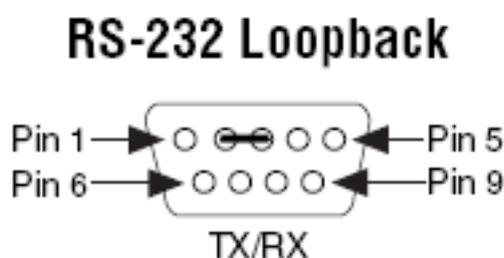
Применимость: Салют-ЭЛ24Д1.

Назначение теста:

Для тестирования скорости порта RS-232 подключенного к UART-контроллеру СнК через RS-232-трансмисмиттер используется утилита `uart_test_client`.

Подготовка к тестированию:

Необходимо замкнуть контакты порта RS-232 (UART3, вилка XP2) модуля в режиме loopback, см. рисунок 3.1:



**Рисунок 3.1. Включение режима loopback порта RS-232 на модулях Салют-ЭЛ24Д1**

Запуск теста:

```
uart_test_client -p /dev/ttyS2 -t 128 -b 115200
```

Ожидаемое время исполнения: 10 секунд.

Пример результата теста:

```
UART port configuration:
{
  'baudrate': 115200,
  'bytesize': 8,
  'dsrdtr': False,
  'interCharTimeout': None,
  'parity': 'N',
  'rtscts': False,
  'stopbits': 1,
  'timeout': 2,
  'writeTimeout': None,
  'xonxoff': False}
total transferred data bytes : 128 KiB
elapsed time : 11.358398 sec
estimated baudrate (with stopbits): 103856 b/sec
performance efficiency: 90.153559 %
```

Критерий прохождения теста:

Значение поля `performance efficiency` составляет более 90%.

### 3.21.2 RS-485 (Салют-ЭЛ24Д1, Салют-ЭЛ24Д2)

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2.

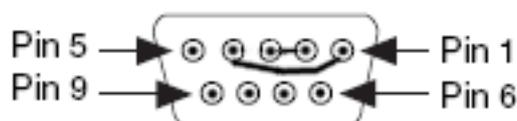
Назначение теста:

Для тестирования скорости порта RS-485 подключенного к UART-контроллеру СнК через RS-485-трансммиттер используется утилита `uart_test_client`.

Подготовка к тестированию:

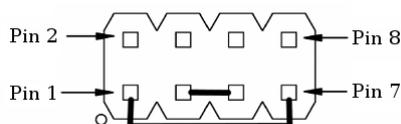
Необходимо замкнуть контакты порта RS-485 на тестируемом модуле:

- для модулей Салют-ЭЛ24Д1 — розетка XS6, см. рисунок 3.2;



**Рисунок 3.2. Включение режима loopback порта RS-485 на модуле Салют-ЭЛ24Д1**

- для модуля Салют-ЭЛ24Д2 — вилка XP1, см. рисунок 3.3.



**Рисунок 3.3. Включение режима loopback порта RS-485 на модуле Салют-ЭЛ24Д2**

Запуск теста:

```
uart_test_client -p /dev/ttyS1 -t 128 -b 115200
```

Ожидаемое время исполнения: 10 секунд.

Пример результата теста:

```
UART port configuration:
{
  'baudrate': 115200,
  'bytesize': 8,
  'dsrdtr': False,
  'interCharTimeout': None,
  'parity': 'N',
  'rtscts': False,
  'stopbits': 1,
  'timeout': 2,
  'writeTimeout': None,
  'xonxoff': False}
total transferred data bytes : 128 KiB
elapsed time : 11.358398 sec
```

(continues on next page)

(продолжение с предыдущей страницы)

```
estimated baudrate (with stopbits): 103856 b/sec  
performance efficiency: 90.153559 %
```

Критерий прохождения теста:

Значение поля `performance efficiency` составляет более 90%.

### 3.21.3 RS-485 (Салют-ЭЛ24ОМ1)

Применимость: Салют-ЭЛ24ОМ1.

Назначение теста:

Для тестирования скорости порта RS-485 подключенного к UART-контроллеру СнК через RS-485-трансмиссивтер в полудуплексе используется утилита `uart_test_client`.

Подготовка к тестированию:

Подключить ответное устройство к порту RS-485 (разъём ХР6). В качестве ответного устройства может применяться другой модуль Салют-ЭЛ24ОМ1 или ПЭВМ с переходником USB-RS-485. Полярность подключения указана на разъёме модуля Салют-ЭЛ24ОМ1. На ответном устройстве запустить:

```
uart_test_server -p <serial-port> -t 128 --marker --rs485-half-duplex
```

Где `<serial-port>` — имя последовательного порта в системе. Для модуля Салют-ЭЛ24ОМ1 порт RS-485 имеет имя `/dev/ttyS3`. Для ПЭВМ имя порта зависит от используемого переходника USB-RS-485, например `/dev/ttyUSB0`.

Запуск теста:

```
uart_test_client -p /dev/ttyS3 -t 128 --marker --rs485-half-duplex
```

Ожидаемое время исполнения: 20 секунд.

Пример результата теста:

```
UART port configuration:  
{  
  'baudrate': 115200,  
  'bytesize': 8,  
  'dsrdtr': False,  
  'interCharTimeout': None,  
  'parity': 'N',  
  'rtscts': False,  
  'stopbits': 1,  
  'timeout': 2,  
  'writeTimeout': None,  
  'xonxoff': False}  
total transferred data bytes : 128 KiB  
elapsed time : 11.358398 sec  
estimated baudrate (with stopbits): 103856 b/sec  
performance efficiency: 90.153559 %
```

Критерий прохождения теста:

Значение поля `performance efficiency` составляет более 90%.

## 3.22 USB

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для измерения скорости передачи данных по USB используется утилита `hdparm`. Для тестирования необходимо дополнительное устройство — USB-флеш-накопитель (или внешний USB диск) со скоростью чтения не менее 25 МиБ/с.

Подготовка к тестированию:

Подключить дополнительное устройство к модулю. При подключении возможен вывод сообщений от ядра Linux в терминал модуля:

```
[ 4631.440832] usb 1-1.1: new high-speed USB device number 3 using dwc2
[ 4631.562841] usb-storage 1-1.1:1.0: USB Mass Storage device detected
[ 4631.569617] scsi host0: usb-storage 1-1.1:1.0
[ 4632.571577] scsi 0:0:0:0: Direct-Access    A-DATA    USB Flash Drive  0.00 PQ:0
↪0 ANSI: 2
[ 4632.584023] sd 0:0:0:0: [sda] 31582415 512-byte logical blocks: (16.1 GB/15.0
↪GiB)
[ 4632.592024] sd 0:0:0:0: [sda] Write Protect is off
[ 4632.597252] sd 0:0:0:0: [sda] Asking for cache data failed
[ 4632.602741] sd 0:0:0:0: [sda] Assuming drive cache: write through
[ 4633.972193] sd 0:0:0:0: [sda] Attached SCSI removable disk
```

Эти сообщения являются стандартными и не влияют на результат прохождения теста.

Запуск теста:

```
hdparm -t /dev/sda
```

Ожидаемое время исполнения: около 3 секунд.

Пример результата теста:

```
Timing buffered disk reads: 88 MB in 3.01 seconds = 29.19 MB/sec
```

Критерий прохождения теста:

Последнее число в последней строке вывода (скорость чтения) составляет:

- для модулей Салют-ЭЛ24Д1, Салют-ЭЛ24Д2 — не менее 25.00 MB/sec;
- для модулей Салют-ЭЛ24ОМ1 с установленным Салют-ЭЛ24ПМ1 — не менее 5.00 MB/sec;
- для модулей Салют-ЭЛ24ОМ1 с установленным Салют-ЭЛ24ПМ2 — не менее 25.00 MB/sec.

## 3.23 VPIN

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для тестирования VPIN и видеосенсора, подключенного по интерфейсу MIPI CSI2.

Подготовка к тестированию:

См. документ «Инструкция по захвату видео с последовательного сенсора на модулях на базе микросхемы 1892ВМ14Я».

Запуск теста:

См. документ «Инструкция по захвату видео с последовательного сенсора на модулях на базе микросхемы 1892ВМ14Я».

Критерий прохождения теста:

На HDMI-мониторе появилось изображение с видеосенсора.

## 3.24 VPOUT

### 3.24.1 HDMI

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2.

Назначение теста:

Для тестирования работоспособности framebuffer-устройства VPOUT используется утилита `mcom02-vpoutfb-test.sh`.

Алгоритм теста:

Для каждого из списка стандартных разрешений:

1. Утилита выставляет разрешение.
2. Запускает проигрыватель с выводом видео на HDMI-монитор.

Подготовка к тестированию:

Подключить HDMI-монитор к соответствующему разъёму модуля.

Запуск теста:

```
modprobe vpoutfb
mcom02-vpoutfb-test.sh
```

Ожидаемое время выполнения: около 1 минуты.

Пример результата теста:

```
1..14
ok 1 - Resolution changed to 640x480
ok 2 - Playing video on resolution 640x480
```

(continues on next page)

(продолжение с предыдущей страницы)

```
ok 3 - Resolution changed to 720x480
ok 4 - Playing video on resolution 720x480
ok 5 - Resolution changed to 800x600
ok 6 - Playing video on resolution 800x600
ok 7 - Resolution changed to 1024x768
ok 8 - Playing video on resolution 1024x768
ok 9 - Resolution changed to 1280x720
ok 10 - Playing video on resolution 1280x720
ok 11 - Resolution changed to 1366x768
ok 12 - Playing video on resolution 1366x768
ok 13 - Resolution changed to 1920x1080
ok 14 - Playing video on resolution 1920x1080
```

Критерии прохождения теста:

1. Все строки вывода, за исключением первой, начинаются с `ok`.
2. После каждого вывода сообщения `Playing video` на HDMI-монитор выводится видео с отсчётом от 0 до 4.

### 3.24.2 LCD

Применимость: Салют-ЭЛ24ОМ1.

Назначение теста:

Для проверки работоспособности LCD-мониторов.

Алгоритм теста:

1. Настроить LCD-монитор EJ070NA-01J;
2. Отобразить тестовое изображение на LCD-мониторе EJ070NA-01J;
3. Настроить LCD-монитор AT070TN92;
4. Отобразить тестовое изображение на LCD-мониторе AT070TN92.

Запуск теста:

1. Подключить к разъему XS7 LCD-монитор EJ070NA-01J;
2. Выполнить команду для определения id коннектора LCD-монитора:

```
modeprint vout-drm
```

Определить секцию коннектора для этого монитора среди LVDS коннекторов. В качестве критерия поиска использовать значение разрешения 1024x600. Это значение соответствует полю `mode` искомой секции коннектора. Значение поля `id` этой секции соответствует id коннектора.

3. Вывести тестовое изображение на LCD-монитор:

```
modetest -M vout-drm -s <id>:1024x600
```

где `<id>` — id коннектора из предыдущего пункта.

4. Подключить к разъему XS8 LCD-монитор AT070TN92;
5. Выполнить команду для определения id коннектора LCD-монитора:

```
modeprint vout-drm
```

Определить секцию коннектора для этого монитора среди LVDS коннекторов. В качестве критерия поиска использовать значение разрешения 800x480. Это значение соответствует полю `mode` искомой секции коннектора. Значение поля `id` этой секции соответствует id коннектора.

6. Вывести тестовое изображение на LCD-монитор:

```
modetest -M vout-drm -s <id>:800x480
```

где `<id>` — id коннектора из предыдущего пункта.

Критерий прохождения теста:

На обоих LCD-мониторах наблюдалось тестовое изображение.

## 3.25 VPU

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для измерения производительности и тестирования функциональности блока аппаратного сжатия видео используется утилита `fc-avico`.

Алгоритм теста:

1. Сгенерировать тестовое видео.
2. Конвертировать тестовое видео в формат M420.
3. Сжать тестовое видео с помощью VPU.
4. Проверить, что скорость сжатия больше 30 кадров в секунду.

Подготовка к тестированию:

1. Подключить HDMI-монитор к соответствующему разъёму модуля.
2. Для модулей Салют-ЭЛ24Д1 или Салют-ЭЛ24Д2 выполнить загрузку драйвера `voutfb`:

```
modprobe voutfb
```

Запуск теста:

```
fc-avico
```

Ожидаемое время исполнения: первый запуск — около 15 секунд, повторные запуски — около 1 секунды.

Пример результата теста:

```
Input #0, yuv4mpegpipe, from 'm420.y4m':
Input #0, yuv4mpegpipe, from 'm420.y4m':
Duration: 00:00:02.000000:00:02.00, start: , start: 0.0000000.000000,
↪bitrate: , bitrate: 276481 kb/s276481 kb/s

Stream #0:0   Stream #0:0: Video: rawvideo (I420 / 0x30323449),
↪yuv420p(progressive), 1280x720: Video: rawvideo (I420 / 0x30323449),
↪yuv420p(progressive), 1280x720, , 25 fps, 25 tbr, 25 fps, 25 tbn, 25 tbr, 25
↪tbc2 25 tbc
Card: avico
Control: h264_i_frame_qp_value = 28
Control: h264_p_frame_qp_value = 28
Control: h264_chroma_qp_index_offset = 0
Control: video_gop_size = 60
Output size: 85 KiB
Total time in main loop: 0.8 s (66.5 FPS)
Card: avico
Control: h264_i_frame_qp_value = 28
Control: h264_p_frame_qp_value = 28
Control: h264_chroma_qp_index_offset = 0
Control: video_gop_size = 60
Output size: 85 KiB
Total time in main loop: 0.7 s (66.8 FPS)
Checking results for thread #0
Checking results for thread #1
TEST PASSED
```

Критерий прохождения теста:

В конце вывода теста присутствует строка TEST PASSED.

После окончания работы тест сохранит сжатое видео в файлах /tmp/fc-avico/encoded\_0.264 и /tmp/fc-avico/encoded\_1.264. Файл можно проиграть на HDMI-мониторе с помощью команды:

```
ffmpeg -re -i /tmp/fc-avico/encoded_0.264 -pix_fmt bgra -f fbdev /dev/fb0
```

## 3.26 Watchdog timer

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для тестирования работоспособности сторожевого таймера используется команда echo.

Алгоритм теста:

1. Включить сторожевой таймер командой echo.

2. Не выполнять сброс сторожевого таймера в течение 15с.
3. Убедиться, что произошла перезагрузка СнК.

Запуск теста:

```
echo 1 > /dev/watchdog
```

Ожидаемое время исполнения: 15-20 секунд.

Критерий прохождения теста:

В течение 15—20 секунд после запуска теста произошла перезагрузка СнК.

## 3.27 Wi-Fi

Применимость: Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1.

Назначение теста:

Для измерения скорости и тестирования функциональности модуля Wi-Fi AP6210 используется утилита `iperf`. Для теста необходимы дополнительные устройства:

- ПК с ОС Linux и пакетом `iperf` версии 2.0.5;
- Wi-Fi-роутер с поддержкой IEEE 802.11g и с возможностью подключения ПК через Ethernet;
- антенна Wi-Fi.

Подготовка к тестированию:

1. Подключить антенну Wi-Fi к разъёму *XW1* модуля.
2. Для модулей Салют-ЭЛ24ОМ1 установить переключатель *XP4* в положение Wi-Fi.
3. Для модулей Салют-ЭЛ24ОМ1 настроить загрузку операционной системы из eMMC согласно инструкции:
  1. Перейти в режим монитора U-Boot (более подробно см. документ «Загрузчик U-Boot для 1892ВМ14Я. Руководство программиста»).
  2. Выбрать загрузку ОС из eMMC:

```
setenv boot_targets "legacy_mmc0 mmc0"  
saveenv  
boot
```

4. Настроить Wi-Fi-роутер: включить Wi-Fi с шифрованием WPA2, указать имя сети (далее — «имя\_сети»), включить DHCP-сервер, подключить ПК кабелем Ethernet к роутеру.
5. На модуле создать файл конфигурации `wpa_supplicant.conf`, выполнив в консоли:

```
wpa_passphrase имя_сети > /etc/wpa_supplicant.conf
```

Приложение `wpa_passphrase` будет ожидать ввода пароля без вывода сообщений. Ввести пароль от сети, нажать `Enter`.

6. Установить сетевое соединение `wlan0`:

```
wpa_supplicant -B -i wlan0 -c /etc/wpa_supplicant.conf
```

После команды наблюдать вывод строки `Successfully initialized wpa_supplicant`.

7. Получить сетевой адрес по DHCP:

```
udhcpc -i wlan0 -n
```

8. Считать IP-адрес модуля в сети Wi-Fi:

```
ifconfig wlan0
```

В поле `inet addr` будет указан IP-адрес модуля в сети Wi-Fi (далее — «адрес\_сервера»).

Запуск теста:

На модуле запустить сервер:

```
iperf --server
```

На ПК запустить клиент:

```
iperf --time=5 --client адрес_сервера
```

Ожидаемое время исполнения: не более 7 секунд.

Пример результата теста:

[ ID]	Interval	Transfer	Bandwidth
[ 3]	0.0-5.0 sec	7.75 MBytes	12.9 Mbits/sec

Критерий прохождения теста:

Для модулей Салют-ЭЛ24Д1, Салют-ЭЛ24Д2, Салют-ЭЛ24ОМ1 значение поля `Bandwidth` составляет не менее 10.00 Mbits/sec. В случае, если значение поля `Bandwidth` ниже ожидаемого, то необходимо настроить Wi-Fi-роутер на использование другого канала и повторить тестирование.