

ИНСТРУКЦИЯ ПО ЗАХВАТУ ВИДЕО С ПОСЛЕДОВАТЕЛЬНОГО СЕНСОРА НА МОДУЛЯХ НА БАЗЕ 1892ВМ14Я

Версия v3.1

01.11.2019

ОГЛАВЛЕНИЕ

1	О документе	3
2	Состав стенда	4
3	Подготовка стенда	5
4	Захват и вывод видео на HDMI-монитор	7
4.1	Последовательность действий	7
4.2	Поясняющая информация	7
5	Захват видео и RTSP-вещание	9
5.1	Последовательность действий	9
5.2	Поясняющая информация	9
6	Передача видео по SpaceWire	12
6.1	Последовательность действий	12
6.2	Поясняющая информация	12
7	Захват, обработка на DSP и вывод видео на HDMI-монитор	14
7.1	Последовательность действий	14
7.2	Поясняющая информация	14

1. О ДОКУМЕНТЕ

Документ содержит указания по настройке модулей на базе 1892ВМ14Я (далее — “СнК”) для захвата и вывода в реальном времени следующих форматов видео:

- 720р 30 FPS для модулей Салют-ЭЛ24Д1 и Салют-ЭЛ24Д2;
- 1080р 30 FPS для модулей Салют-ЭЛ24ОМ1 с установленным Салют-ЭЛ24ПМ1 или Салют-ЭЛ24ПМ2.

Документ применим к отладочным модулям (далее — модулям) следующих ревизий:

- Салют-ЭЛ24Д1 r1.3;
- Салют-ЭЛ24Д1 r1.4;
- Салют-ЭЛ24Д1 r1.5;
- 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.

Для модулей Салют-ЭЛ24Д1 и Салют-ЭЛ24Д2 захват видео выполняется с видеомодуля LINK OV2715 Rev 1.0, подключенного по последовательному интерфейсу MIPI CSI2 к соответствующему разъёму модуля.

Для модулей Салют-ЭЛ24ОМ1 захват видео выполняется с видеомодуля RPi Camera (B), подключенного по последовательному интерфейсу MIPI CSI2 к соответствующему разъёму модуля.

Варианты демонстрации захваченного видео:

- вывод видео на HDMI-монитор;
- RTSP-вещание на модуле, приём и вывод видео на ПЭВМ.
- передача видео по SpaceWire с одного модуля на другой, вывод видео на приёмном модуле.
- обработка на DSP и вывод видео на HDMI-монитор.

Для работы требуется прошивка SD/eMMC-карты, собранная из дистрибутива ОС GNU/Linux на базе Buildroot версии v3.1 для 1892ВМ14Я.

2. СОСТАВ СТЕНДА

Для захвата и вывода видео необходим стенд:

1. Комплект модуля: модуль, блок питания, SD-карта.
2. Видеомодуль LINK OV2715 (для модулей Салют-ЭЛ24Д1 и Салют-ЭЛ24Д2).
3. Видеомодуль RPi Camera (B) (для модулей Салют-ЭЛ24ОМ1).
4. Гибкий шлейф для подключения видеомодуля.
5. HDMI-монитор.
6. HDMI-кабель.
7. Ethernet-кабель.
8. ПЭВМ, удовлетворяющая требованиям:
 1. ПЭВМ должна удовлетворять требованиям из руководства программиста дистрибутива ОС GNU/Linux на базе Buildroot;
 2. ПЭВМ и модуль подключены к одной локальной Ethernet-сети.

Демонстрация передачи видео по SpaceWire выполняется на двух модулях Салют-ЭЛ24ОМ1 с установленным Салют-ЭЛ24ПМ2 (на модулях Салют-ЭЛ24Д1, Салют-ЭЛ24Д2 отсутствует разъём SpaceWire).

Для передачи видео по SpaceWire нужны дополнительно:

1. Второй комплект модуля.
2. Второй Ethernet-кабель.
3. Кабель SpaceWire (входит в состав комплекта модуля Салют-ЭЛ24ОМ1).

3. ПОДГОТОВКА СТЕНДА

Для подготовки стенда к захвату видео с видеомодуля необходимо:

1. Прошить SD-карту прошивкой, собранной из дистрибутива.
2. Установить SD-карту в модуль.
3. Для модулей Салют-ЭЛ24ОМ1 настроить загрузку операционной системы из SD-карты согласно инструкции (данные действия не требуются, если прошивка eMMC-карты уже содержит дистрибутив ОС GNU/Linux на базе Buildroot версии v3.0 или выше):
 1. Перейти в режим монитора U-Boot (см. документ “Загрузчик U-Boot для 1892ВМ14Я. Руководство программиста”).
 2. Установить переключатель *XP4* в положение *uSDcard*.
 3. Выбрать загрузку ОС из SD-карты:

```
setenv mmcdev 1
saveenv
boot
```

4. Подключить модуль к сети Ethernet-кабелем, используя соответствующий разъем модуля.
5. Подключить видеомодуль к соответствующему разъёму модуля через гибкий шлейф:
 - для Салют-ЭЛ24Д1 r1.3 — разъём XS1;
 - для Салют-ЭЛ24Д1 r1.4 — разъём XS1;
 - для Салют-ЭЛ24Д1 r1.5 — разъем XS1;
 - для Салют-ЭЛ24Д2 r1.1 — разъём XS1;
 - для Салют-ЭЛ24ОМ1 r1.1 — разъём XS9;
 - для Салют-ЭЛ24ОМ1 r1.2 — разъём XS9.

Пример подключения видеомодуля к Салют-ЭЛ24Д2 r1.1 представлен на рисунке 3.1.

Пример подключения видеомодуля к Салют-ЭЛ24ОМ1 r1.2 представлен на рисунке 3.2.

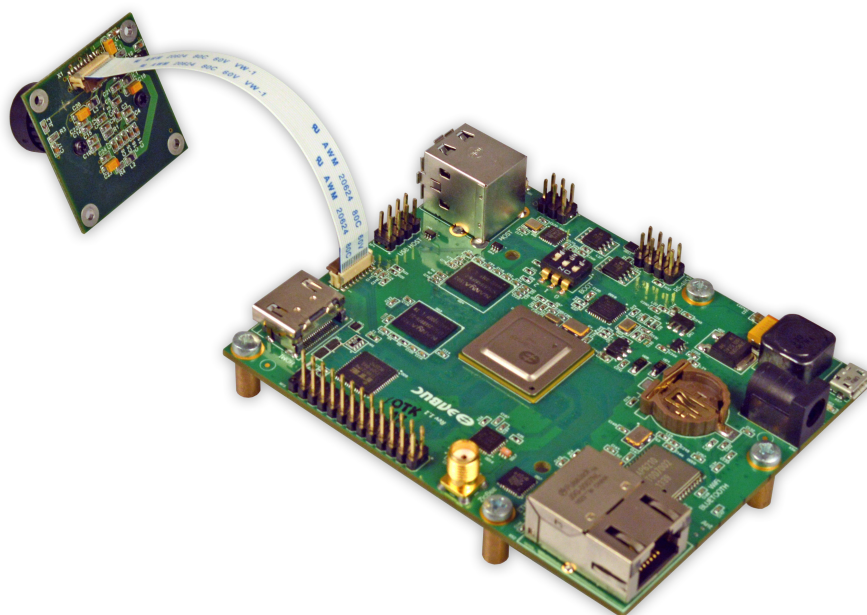


Рисунок 3.1. Пример подключения видеомодуля LINK OV2715 Rev 1.0 к модулю Салют-ЭЛ24Д2 r1.1

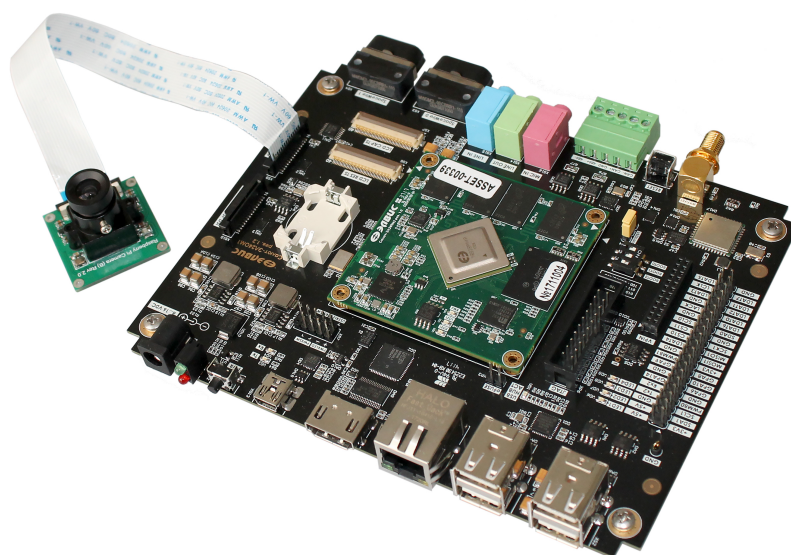


Рисунок 3.2. Пример подключения видеомодуля RPi Camera (B) к модулю Салют-ЭЛ24ОМ1 r1.2

4. ЗАХВАТ И ВЫВОД ВИДЕО НА HDMI-МОНИТОР

4.1 Последовательность действий

Для запуска демонстрации необходимо:

1. Подготовить модуль в соответствии с главой *Подготовка стенда*.
2. Подключить HDMI-монитор HDMI-кабелем к соответствующему разъёму модуля.
3. Подать питание на модуль.
4. На ПЭВМ открыть консоль, установить соединение по терминалу SSH с модулем.
5. Для модулей Салют-ЭЛ24Д1 и Салют-ЭЛ24Д2 на ПЭВМ в терминале SSH выполнить:

```
modprobe voutfb  
v4l2tofb -i 2 -w 1280 -h 720
```

6. Для модуля Салют-ЭЛ24ОМ1 на ПЭВМ в терминале SSH выполнить:

```
v4l2tofb -i 2 -w 1920 -h 1080
```

7. Наблюдать на экране HDMI-монитора, подключенного к модулю, видео, поступающее с видеомодуля.
8. Прервать исполнение программы, нажав Ctrl-C.

4.2 Поясняющая информация

Диаграмма потоков видеокадров представлена на рисунке 4.1. Захват видео с видеомодуля выполняет блок VPIN СнК. DMA блока VPIN записывает видеокадры в буфер видеоввода в ОЗУ DDR. CPU СнК копирует видеокадры из буфера видеоввода в буфер драйвера Frame buffer. DMA блока VPOUT СнК считывает видеокадры из буфера драйвера Frame buffer и выводит на параллельный порт подключенный к HDMI-контроллеру модуля.

В ОС Linux захват видео выполняется через драйвер VPIN, реализующий стандартный API интерфейс V4L2¹. Спецификация на драйвер находится в документе “Спецификация на драйвер V4L2 VINC блока ISP VPIN/VINC 1892BM14Я”.

Для модулей Салют-ЭЛ24Д1 и Салют-ЭЛ24Д2 по умолчанию драйвер VPOUT Frame buffer (voutfb) не загружается автоматически. Для загрузки драйвера на одну сессию (до перезагрузки ОС), необходимо выполнить:

```
modprobe voutfb
```

¹ <https://linuxtv.org/downloads/v4l-dvb-apis/uapi/v4l/v4l2.html>

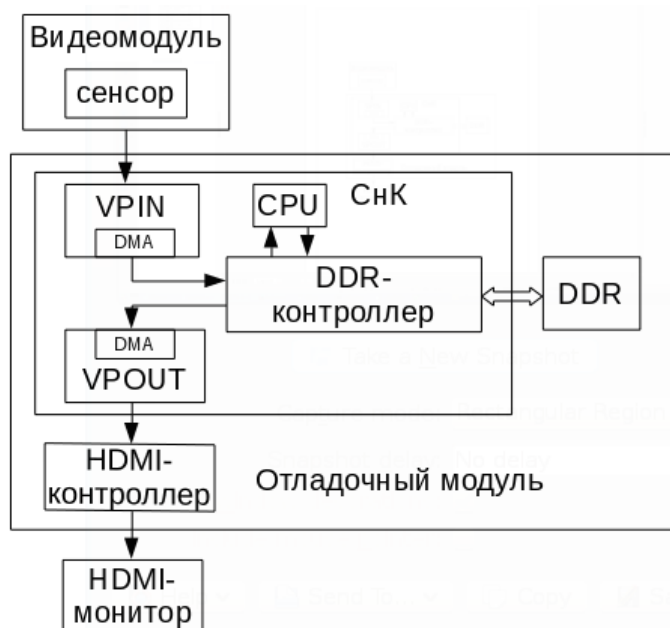


Рисунок 4.1. Диаграмма потоков видеок кадров

Процедура настройки автоматической загрузки драйвера описана в главе “Включение драйвера framebuffer voutfb” документа “Дистрибутив ОС GNU/Linux на базе Buildroot для 1892BM14Я. Руководство программиста”.

Для вывода видео на монитор используется утилита `v4l2tofb`. Утилита открывает устройство соответствующее входному интерфейсу, который задается ключом `-i`, устанавливает формат `BGR32`², принимает кадры от VPIN и выводит их на Frame buffer.

Соответствие между значением ключа `-i` и выбираемым входным интерфейсом следующие: 0 - PInterface0; 1 - PInterface1; 2 - SInterface0(CSI0); 3 - SInterface1(CSI1).

² <https://linuxtv.org/downloads/v4l-dvb-apis/uapi/v4l/pixfmt-rgb.html>

5. ЗАХВАТ ВИДЕО И RTSP-ВЕЩАНИЕ

5.1 Последовательность действий

Для запуска RTSP-вещания видео необходимо:

1. Подготовить модуль в соответствии с главой *Подготовка стенда*.
2. Подать питание на модуль.
3. На ПЭВМ открыть консоль, установить соединение по терминалу SSH с модулем.
4. В терминале SSH выполнить:

```
cap-enc -s <res> -f1 /dev/v4l/by-path/platform-37200000.vinc-video-index0 \
/dev/v4l/by-path/platform-37100000.codec-video-index0 | \
gst-rtsp-test-launch 'fdsrc blocksize=4116480 do-timestamp=true ! \
video/x-h264,stream-format=byte-stream,alignment=nal ! \
rtph264pay name=pay0 mtu=65507'
```

где <res> — необходимо заменить на разрешение кадра. Для модулей Салют-ЭЛ24Д1 и Салют-ЭЛ24Д2 установить разрешение кадра 1280x720. Для модуля Салют-ЭЛ24ОМ1 установить разрешение кадра 1920x1072.

5. Для приема и вывода видео на монитор на ПЭВМ с использованием [GStreamer](https://gstreamer.freedesktop.org)³ необходимо в консоли ПЭВМ выполнить:

```
gst-launch-1.0 -v rtspsrc location=rtsp://<board-ip>:8554/test ! \
rtph264depay ! queue ! h264parse ! avdec_h264 ! xvimagesink sync=false
```

где <board-ip> — необходимо заменить на внешний сетевой адрес модуля.

Для приема и вывода видео на монитор на ПЭВМ с использованием [FFmpeg](https://www.ffmpeg.org)⁴ необходимо в консоли ПЭВМ выполнить:

```
ffmpeg rtsp://<board-ip>:8554/test
```

6. Наблюдать на экране монитора, подключенного к ПЭВМ, видео, поступающее с видеомодуля.
7. Прервать исполнение программы, нажав Ctrl-C.

5.2 Поясняющая информация

Диаграмма потоков видеок кадров представлена на рисунке 5.1. Захват видео с видеомодуля выполняет блок VPIN СнК. DMA блока VPIN записывает видеок кадры в буфер видео-

³ <https://gstreamer.freedesktop.org>

⁴ <https://www.ffmpeg.org>

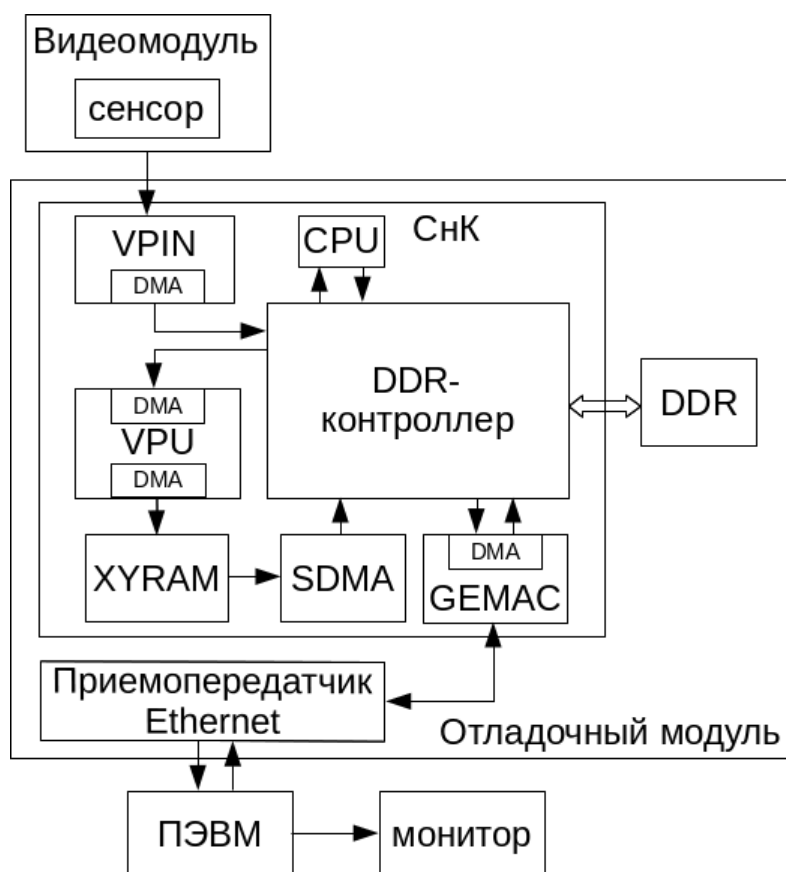


Рисунок 5.1. Диаграмма потоков видеокадров

ввода в ОЗУ DDR. Сжатие видео в соответствии со стандартом H.264⁵ (Constrained Base line Profile) выполняет блок VPU СнК. DMA блока VPU записывает исходные видеокадры в буфер в памяти VRAM СнК, а сжатые данные — в буфер в памяти XYRAM СнК. Блок SDMA СнК копирует сжатые данные из буфера в памяти XYRAM в буфер в ОЗУ DDR. CPU СнК передает данные в блок GEMAC СнК, который отправляет их в приемопередатчик Ethernet, соединенный с ПЭВМ. ПЭВМ принимает видео, декодирует его и выводит на монитор.

В ОС Linux захват видео выполняется через драйвер VPIN, реализующий стандартный API интерфейс V4L2. Спецификация на драйвер находится в документе “Спецификация на драйвер V4L2 VINC блока ISP VPIN/VINC 1892BM14Я”. Сжатие видео выполняется через драйвер VPU, реализующий стандартный API интерфейс V4L2. Описание драйвера находится в документе “Ядро Linux для 1892BM14Я. Руководство системного программиста”.

Для захвата и сжатия видео в соответствии со стандартом H.264 используется утилита `car-enc`. Утилита открывает устройство, соответствующее видеомодулю, и устройство, соответствующее VPU, устанавливает формат M420⁶, принимает кадры от VPIN, передает их VPU, принимает сжатые данные от VPU и записывает их в стандартный вывод.

Примечание: Утилита `car-enc` позволяет установить контролы V4L2 драйвера VPU с

⁵ <https://www.itu.int/rec/T-REC-H.264>

⁶ <https://linuxtv.org/downloads/v4l-dvb-apis/uapi/v4l/pixfmt-m420.html>

помощью опции `-c`.

Для RTSP-вещания используется утилита `gst-rtsp-test-launch` (GStreamer). Утилита с помощью элемента `fdsrc` считывает данные из стандартного ввода в буферы GStreamer, преобразовывает буферы GStreamer в пакеты RTP с помощью элемента `rtph264pay` и реализует работу RTSP-сервера с помощью библиотеки `gst-rtsp-server` GStreamer.

Примечание: В представленной команде временные метки буферов устанавливаются элементом `fdsrc` в момент получения буфера, а не в момент считывания изображения сенсором.

Свойство `blocksize` элемента `fdsrc` должно иметь значение, равное размеру `capture`-буфера, который вычисляется по следующей формуле:

$$S = W \times H \times 2$$

где W — ширина кадра, H — высота кадра. Например, при разрешении 1920x1072 размер `capture`-буфера равен 4116480 байт, следовательно, свойство `blocksize` элемента `fdsrc` необходимо установить в значение 4116480.

Примечание: `gst-rtsp-test-launch` по умолчанию использует RTSP-порт 8554. Другой порт может быть задан с помощью опции `-p`.

Прием, декодирование и вывод видео на монитор на ПЭВМ выполняется с использованием GStreamer или FFmpeg.

Примечание: При использовании GStreamer протокол передачи может быть задан с помощью свойства `protocols` элемента `rtspsrc`, например:

```
protocols="tcp+udp-mcast+udp"
```

GStreamer будет пытаться использовать протоколы в следующем порядке: UDP, unicast/UDP, multicast/TCP. Данный порядок не может быть переопределен.

При использовании FFmpeg протокол передачи может быть задан с помощью опции `-rtsp_transport`, например:

```
-rtsp_transport tcp+udp_multicast+udp
```

Стабильность декодирования на ПЭВМ может зависеть от используемых версий FFmpeg и GStreamer, драйверов аппаратного декодера или производительности CPU (при программном декодировании).

В случае потери одного пакета при передаче по UDP, на приёмнике становится невозможно восстановить все P-кадры до следующего I-кадра. В этом случае декодер может либо воспроизводить испорченные кадры или показывать последний неиспорченный кадр до приёма следующего I-кадра, поэтому рекомендуется использовать передачу по TCP.

6. ПЕРЕДАЧА ВИДЕО ПО SPACEWIRE

6.1 Последовательность действий

Для запуска передачи видео по SpaceWire необходимо:

1. Подготовить модуль-передатчик в соответствии с главой *Подготовка стенда*.
2. Подготовить модуль-приёмник в соответствии с главой *Подготовка стенда* пункты 1-4.
3. Подключить HDMI-монитор HDMI-кабелем к соответствующему разъёму модуля-приёмника.
4. Соединить модули кабелем SpaceWire, используя гнёзда XS3 на обоих модулях.
5. Подать питание на модули.
6. На ПЭВМ открыть две консоли, установить соединение по терминалу SSH или UART с модулями.
7. В терминале SSH модуля-передатчика выполнить запуск systemd-сервиса:

```
systemctl start spacewire-video-transmit
```

8. В терминале SSH модуля-приёмника выполнить запуск systemd-сервиса:

```
systemctl start spacewire-video-receive
```

9. Наблюдать на экране HDMI-монитора видео, поступающее с видеомодуля, подключенного к модулю-передатчику.
10. Прервать исполнение программы, выполнив остановку systemd-сервисов в терминале SSH модуля-передатчика:

```
systemctl stop spacewire-video-transmit
```

и в терминале SSH модуля-приёмника:

```
systemctl stop spacewire-video-receive
```

6.2 Поясняющая информация

В ОС Linux захват видео выполняется через драйвер VPIN, реализующий стандартный API интерфейс V4L2. Спецификация на драйвер находится в документе “Спецификация на драйвер V4L2 VINC блока ISP VPIN/VINC 1892BM14Я”. Сжатие видео выполняется через драйвер VPU, реализующий стандартный API интерфейс V4L2. Описание драйвера находится в документе “Ядро Linux для 1892BM14Я. Руководство системного программиста”.

Для захвата и сжатия видео в соответствии со стандартом H.264 используется утилита `car-enc`. Утилита открывает устройство, соответствующее видеомодулю, и устройство, соответствующее VPU, устанавливает формат M420⁷, принимает кадры от VPIN, передает их VPU, принимает сжатые данные от VPU и записывает их в стандартный вывод. Стандартный вывод перенаправляется на вход утилиты `swic-xfeg`, которая посредством использования Linux драйвера контроллера SpaceWire передаёт данные на модуль-приёмник по кабелю SpaceWire.

На модуле-приёмнике отключается бланкирование консоли. Данные, поступающие от модуля-передатчика по кабелю SpaceWire, читаются утилитой `swic-xfeg` и записываются в стандартный вывод. Стандартный вывод перенаправляется на вход утилиты `ffmpeg`, которая выводит видеопоток на HDMI-монитор.

⁷ <https://linuxtv.org/downloads/v4l-dvb-apis/uapi/v4l/pixfmt-m420.html>

7. ЗАХВАТ, ОБРАБОТКА НА DSP И ВЫВОД ВИДЕО НА HDMI-МОНИТОР

7.1 Последовательность действий

Примечание: Демонстрация может быть запущена только на модуле Салют-ЭЛ24ОМ с установленным Салют-ЭЛ24ПМ.

Для вывода видео на монитор необходимо:

1. Подготовить модуль в соответствии с главой *Подготовка стенда*.
2. Подключить HDMI-монитор HDMI-кабелем к соответствующему разъёму модуля.
3. Подать питание на модуль.
4. На ПЭВМ открыть консоль, установить соединение по терминалу SSH с модулем.
5. Загрузить модуль драйвера *delcore30m*. Для этого в терминале SSH выполнить:

```
modprobe -r avico
echo -n "37220000.dma" > /sys/bus/amba/drivers/dma-pl330/unbind
modprobe -r delcore30m
modprobe delcore30m
```

6. Для запуска демонстрации инверсии в терминале SSH выполнить:

```
delcore30m-inversiondemo -i 2
```

или запустить демонстрацию детектора движения:

```
delcore30m-dspdector -i 2
```

7. Наблюдать на экране HDMI-монитора, подключенного к модулю, обработанное видео, поступающее с видеомодуля.
8. Прервать исполнение программы, нажав Ctrl-C.

7.2 Поясняющая информация

Диаграмма потоков видеокадров представлена на рисунке 7.1. Перемещения видеокадров осуществляются с помощью утилит инверсии видеокадров *delcore30m-inversiondemo*, детекции движения *delcore30m-dspdector*, а также драйверов DSP, VPIN и DRM. Захват видео с видеомодуля выполняет блок VPIN СнК. DMA блока VPIN записывает видеокадры в буфер видеоввода в ОЗУ DDR. Блок SDMA СнК копирует видеокадр из буфера в

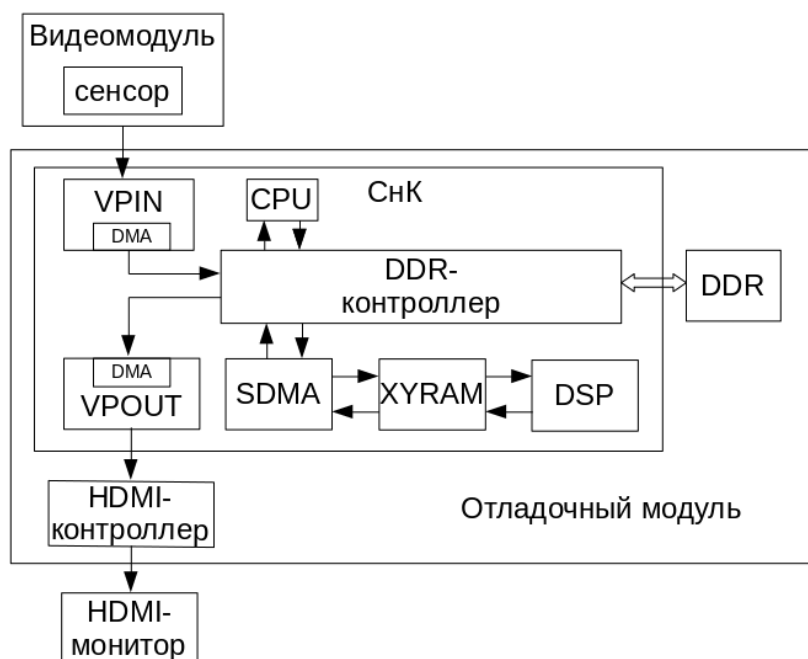


Рисунок 7.1. Диаграмма потоков видеокадров

памяти DDR в буфер памяти XYRAM. DSP СнК обрабатывает данные, записывая результат обработки в буфер памяти XYRAM, который копируется блоком SDMA СнК в буфер памяти DDR, соответствующий Frame buffer. DMA блока VPOUT СнК считывает видеокадры из буфера драйвера Frame buffer и выводит на параллельный порт подключенный к HDMI-контроллеру модуля.

Утилиты `delcore30m-inversiondemo` и `delcore30m-dspdectector` выполняют следующие действия:

1. Открывают устройство драйвера VINC, соответствующее входному интерфейсу, который задается ключом `-i` и выполняют следующие действия:
 - Устанавливают формат `BGR32`⁸.
 - Устанавливают файловые дескрипторы буферов, в которые будут сохраняться кадры от VPIN.

Спецификация на драйвер VINC находится в документе “Драйвер V4L2 vinc ISP VPIN/VINC 1892BM14Я. Руководство программиста”.

Значение ключа `i` может принимать следующие значения:

- 0 — PInterface0;
- 1 — PInterface1;
- 2 — SInterface0(CSI0);
- 3 — SInterface1(CSI1).

2. Открывают устройство драйвера `delcore30m` и выполняют следующие действия:

⁸ <https://linuxtv.org/downloads/v4l-dvb-apis/uapi/v4l/pixfmt-rgb.html>

- Выделяют ресурсы DSP и SDMA.
- Загружают исполняемый код DSP.
- Выделяют буфера памяти и загружает необходимые данные.
- Разбивают очередной видеокадр на тайлы.
- Настраивают цепочку SDMA.
- Создают задание для DSP.
- Запускают задание на DSP.
- Ожидают завершения задания на DSP.

Спецификация на драйвер *delcore30m* находится в документе “Драйвер *delcore30m*. Руководство программиста”.

3. Открывают устройство драйвера DRM и выполняют следующие действия:

- Настраивают выходное разрешение.
- Экспортируют два кадровых буфера, откуда будут считываться кадры, обработанные на DSP.

Модуль драйвера *delcore30m* по умолчанию выгружен. Для загрузки модуля драйвера *delcore30m* на одну сессию (до перезагрузки ОС), необходимо заранее выгрузить драйвер *pl330*.