



МАКРО EMC
Группа компаний Макро Групп

Модуль вычислительный
NanoS

Руководство по эксплуатации

МРЦН.467444.001РЭ



Содержание

Перечень сокращений.....	5
1 Описание и работа.....	7
1.1 Назначение изделия.....	7
1.2 Технические характеристики.....	10
1.3 Комплект поставки.....	12
2 Использование по назначению.....	13
2.1 Эксплуатационные ограничения.....	13
2.2 Карта интерфейсов модуля.....	13
2.3 Подключение модуля.....	14
2.4 Авторизация в системе.....	14
2.5 Подключение к модулю по сетевому протоколу SSH.....	14
2.5.1 Подключение по SSH через командную строку Windows.....	14
2.5.2 Подключение по SSH через командную строку Linux.....	15
2.6 Интерфейс USB.....	16
2.6.1 Общие сведения.....	16
2.6.2 Запуск USB-камеры на модуле.....	17
2.7 Интерфейс DIO.....	18
2.7.1 Общие сведения.....	18
2.7.2 Цоколевка и схемотехника.....	18
2.7.3 Используемые сигналы подключения DIO.....	21
2.7.4 Реализация DIO в системе Linux.....	21
2.7.5 Доступ к интерфейсу DIO из С.....	23
2.8 Интерфейс RS-232.....	24
2.8.1 Общие сведения.....	24
2.8.2 Цоколевка и схемотехника.....	24
2.8.3 Используемые сигналы подключения RS-232.....	25
2.8.4 Доступ к RS-232 из командной строки.....	25
2.8.5 Доступ к RS-232 из С.....	26
2.9 Интерфейс RS-485.....	26
2.9.1 Общие сведения.....	26
2.9.2 Цоколевка и схемотехника.....	26
2.9.3 Используемые сигналы подключения RS-485.....	27

2.9.4	Реализация интерфейса RS-485 в системе Linux.....	28
2.9.5	Доступ к RS-485 из командной строки.....	28
2.9.6	Доступ к RS-485 из C.....	28
2.10	Интерфейс I2C.....	29
2.10.1	Общие сведения.....	29
2.10.2	Цоколевка и схемотехника.....	29
2.10.3	Реализация интерфейса I2C в системе Linux.....	30
2.10.4	Доступ к I2C из командной строки.....	30
2.10.5	Доступ к I2C из C.....	31
2.11	Интерфейс CAN.....	31
2.11.1	Общие сведения.....	31
2.11.2	Цоколевка и схемотехника.....	31
2.11.3	Используемые сигналы подключения CAN.....	32
2.11.4	Реализация интерфейса CAN в системе Linux.....	32
2.11.5	Доступ к CAN из командной строки.....	32
2.11.6	Доступ к CAN из C.....	34
2.12	Интерфейс SPI.....	34
2.12.1	Общие сведения.....	34
2.12.2	Цоколевка и схемотехника.....	34
2.12.3	Используемые сигналы подключения SPI.....	35
2.12.4	Доступ к SPI из командной строки.....	35
2.12.5	Доступ к SPI из C.....	36
2.13	Интерфейс MIPI-CSI-2.....	36
2.13.1	Общие сведения.....	36
2.13.2	Цоколевка и схемотехника.....	36
2.13.3	Используемые сигналы подключения MIPI-CSI-2.....	39
2.13.4	Доступ к интерфейсу MIPI-CSI-2 в системе Buildroot.....	40
2.13.5	Запуск МЭЖ в системе Buildroot.....	41
2.14	Последовательный интерфейс UART0.....	45
2.14.1	Общее описание.....	45
2.14.2	Цоколевка и схемотехника разъема UART0.....	45
2.14.3	Используемые сигналы подключения UART0.....	46
3	Возможные неисправности и методы их устранения.....	47
3.1	Особенности работы интегральной микросхемы 1892BA018.....	47
3.2	Служба поддержки и полезные ссылки.....	48
3.2.1	Канал поддержки.....	48

3.2.2 Полезные ссылки.....	48
4 Условия хранения.....	49
5 Гарантия производителя.....	50
Приложение А – Список пакетов в сборке образа системы Buildroot.....	51
Приложение Б – Демо приложение SmartCamApp.....	60

Перечень сокращений

ВКС	–	видеоконференцсвязь
ИИ	–	искусственный интеллект
ИМС	–	интегральная микросхема
ККМ	–	контрольно-кассовая машина
МЭК	–	модуль электронной камеры
ОС	–	операционная система
РЭ	–	руководство по эксплуатации
ЭД	–	эксплуатационная документация
СнК	–	система на кристалле
DSP	–	цифровой сигнальный процессор (анг. digital signal processor)
NPU	–	нейронный процессор (анг. neural processing unit)
SBC	–	одноплатный компьютер (анг. single board computer)
SBL	–	примитивный загрузчик (анг. secondary bootloader)
TOPs	–	единица скорости вычислений процессора - триллион операций в секунду (анг. trillion operations per second)

Настоящее руководство по эксплуатации является руководящим документом для изучения устройства, функционирования, порядка и правил использования по назначению, при техническом обслуживании и хранении Модуля вычислительного NanoS-T МРЦН.467444.001, NanoS-T-W МРЦН.467444.001-01.

В связи с постоянным усовершенствованием изделия в его конструкцию и комплектацию могут быть внесены отдельные изменения, не влияющие на основные эксплуатационные характеристики, которые могут быть не отражены в настоящем РЭ.

Настоящее РЭ может быть уточнено и дополнено в установленном порядке.

Несоблюдение указаний по эксплуатации, техническому обслуживанию и правил техники безопасности, изложенных в настоящем Руководстве, может быть причиной возникновения ситуаций, связанных с причинением вреда здоровью.

ООО «Макро ЕМС» гарантирует соответствие качества модуля вычислительного требованиям технических условий МРЦН.467444.001ТУ при соблюдении потребителем мер безопасности, условий и правил хранения, транспортирования, эксплуатации и монтажа, установленных в ЭД на модуль.

Адрес изготовителя:

Российская Федерация, 196105, г. Санкт-Петербург,
ул. Свеаборгская, д.12, пом.3Н.

Телефон/факс: +7(812) 370-60-70

Электронная почта: contract@macrogroup.ru

ИНН 7810895610 КПП 781001001 Р/с 40702810206000003697

БИК 044030920 К/с 30101810000000000920

САНКТ-ПЕТЕРБУРГСКИЙ ФИЛИАЛ ПАО "ПРОМСВЯЗЬБАНК"

ОКПО 43468759 ОКВЭД 26.30, 27.90, 46.69.9, 47.78, 47.99, 72.1, 73.20.1

1 Описание и работа

1.1 Назначение изделия

Модуль вычислительный «NanoS» (далее – модуль) является законченным модулем электронным в форм-факторе Nano-ITX и нацелен для работы с классическими и нейросетевыми алгоритмами в равной степени.

Модуль базируется на СнК 1892ВА018, коммерческое название «СКИФ» от АО НПЦ «Элвис», представляющий собой четырёхъядерный процессор архитектуры Arm A53 с максимальной частотой до 2 ГГц дополненный DSP (NPU) сопроцессором Elcore-50.

Модуль работает под системами семейства Linux. На текущий момент проверена совместимость модуля с системами AltLinux, RedOS, Buildroot.

Модуль представлен в двух вариантах исполнения:

- **NanoS-T** МРЦН.467444.001 – модуль вычислительный с термопакетом (см. Рисунок 1.1.1);
- **NanoS-T-W** МРЦН.467444.001-01 – модуль вычислительный с термопакетом, без интерфейсов DIO, MIPI-CSI, CAN, RS485, внешнего I2C, внешнего SPI, нет разъема аппаратного отключения звука/видео, питание только с разъема 2.1x5.5 мм (см. Рисунок 1.1.2).

Подробные технические характеристики, различия в интерфейсах исполнений приведены в соответствующем подразделе РЭ (1.2 Технические характеристики).

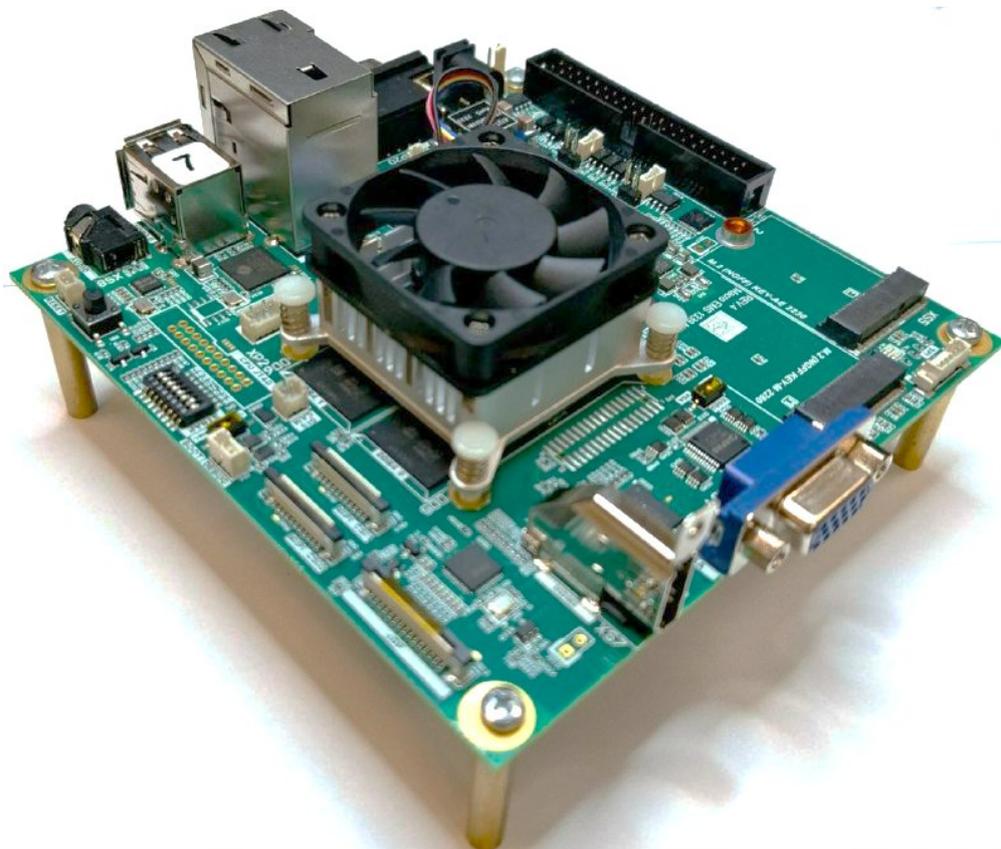


Рисунок 1.1.1 – Внешний вид NanoS-T МРЦН.467444.001

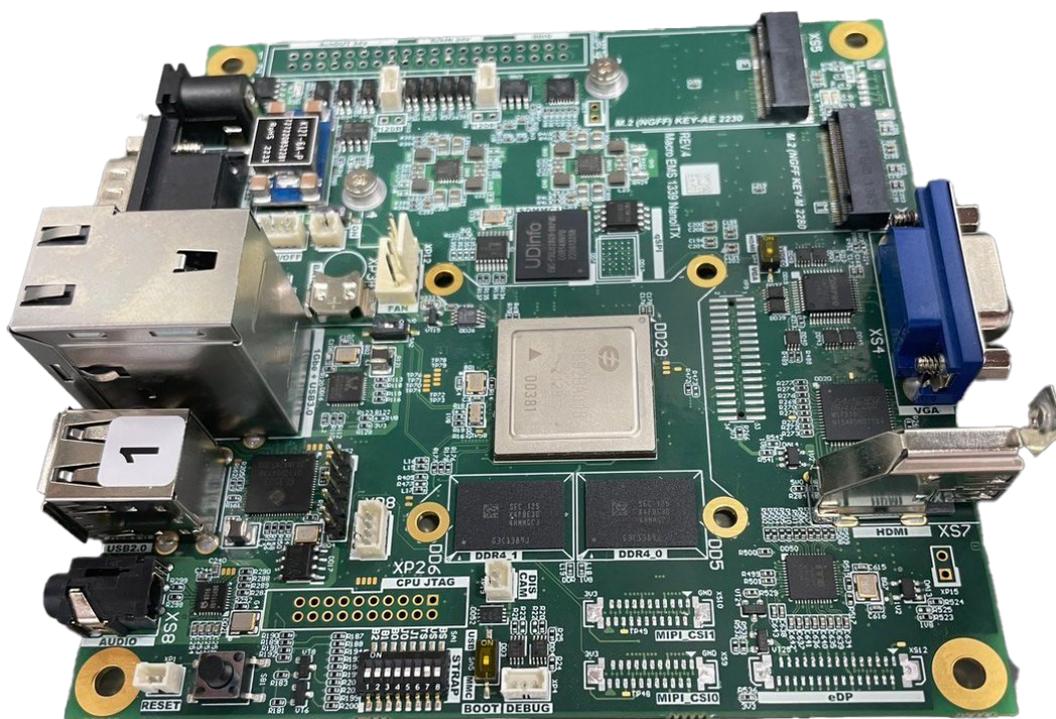


Рисунок 1.1.2 – Внешний вид NanoS-T-W МРЦН.467444.001-01

Модуль предназначен для использования в качестве универсального вычислителя для следующих устройств:

- рабочие станции начального уровня;
- умные камеры, камеры ВКС;
- терминалы самообслуживания;
- торговые терминалы.

Основные сферы применения модуля показаны на рисунке 1.3



Рисунок 1.1.3 – Основные сферы применения модуля

1.2 Технические характеристики

Основные технические характеристики указаны в таблице 1.2

Таблица 1.2 – Основные характеристики модуля вычислительного

№ п/п	Характеристика	Значение	
		NanoS-T	NanoS-T-W
1.	Общие характеристики		
1.1	Габаритные размеры, не более, мм	120×120×37	
1.2	Процессор	СКИФ (1892ВА018)	
1.3	Напряжение электропитания, В	12	
1.4	Ток потребления, не более, А	1,5	
1.5	Мощность потребления, не более, Вт	18	
1.6	Охлаждение	Активное	
1.7	Аппаратное отключение камеры и звука	Да	-
1.8	Аппаратное включение/отключение электропитание	Да	
1.9	Функционирование под операционными системами	RedOS, AltLinux, Пользовательская Linux (Buildroot)	
1.10	Часы реального времени с элементом резервного электропитания	Да	Да
2.	Память		
2.1	ОЗУ (тип LPDDR4), не менее, ГБ	8 (2x4)*	
2.2	ПЗУ (тип eMMC), не менее, ГБ	16*	
2.3	ПЗУ (тип qSPI), не менее, МБ	16	
3.	Сетевые интерфейсы		
3.1	Ethernet 10/100/1000M, RJ-45, шт.	1	1
3.2	Модуль Wi-Fi (m.2 2230 KEY-AE), шт.	1	1
4.	Видео интерфейсы		
4.1	HDMI 1.4, шт.	1	1
4.2	eDP, 30pin, шт.	1	-
4.3	VGA (DB15), шт.	1	1

Продолжение таблицы 1.2

5.	Прочие интерфейсы		
5.1	SSD накопитель (m.2 2280 KEY-M), шт.	1	1
5.2	USB 3.0 (Type-A), шт.	2	2
5.3	USB 2.0 (Type-A), шт.	2	2
5.4	Аудио выход стерео, шт.	1	1
	Дискретные входы и выходы		
	Универсальный вход/выход (TTL 3,3 В), шт.	8	-
5.5	Оптоизолированный вход (Opto DI), шт.	8	-
	Оптоизолированный выход (Opto DO), шт.	8	-
5.6	Интерфейс подключения камеры MIPI-CSI-2, 15pin, шт.	2	-
5.7	Внешний интерфейс I2C, шт.	1	-
5.8	Внешний интерфейс CAN 2.0B, шт.	1	-
5.9	Внешний интерфейс RS-485, шт.	1	-
5.10	Внешний интерфейс RS-232, шт.	1	-
5.11	Внешний интерфейс SPI, шт.	1	-
Примечание – характеристики и их значения могут быть изменены без уведомления			

Габаритные размеры модуля представлены на рисунке 1.21.

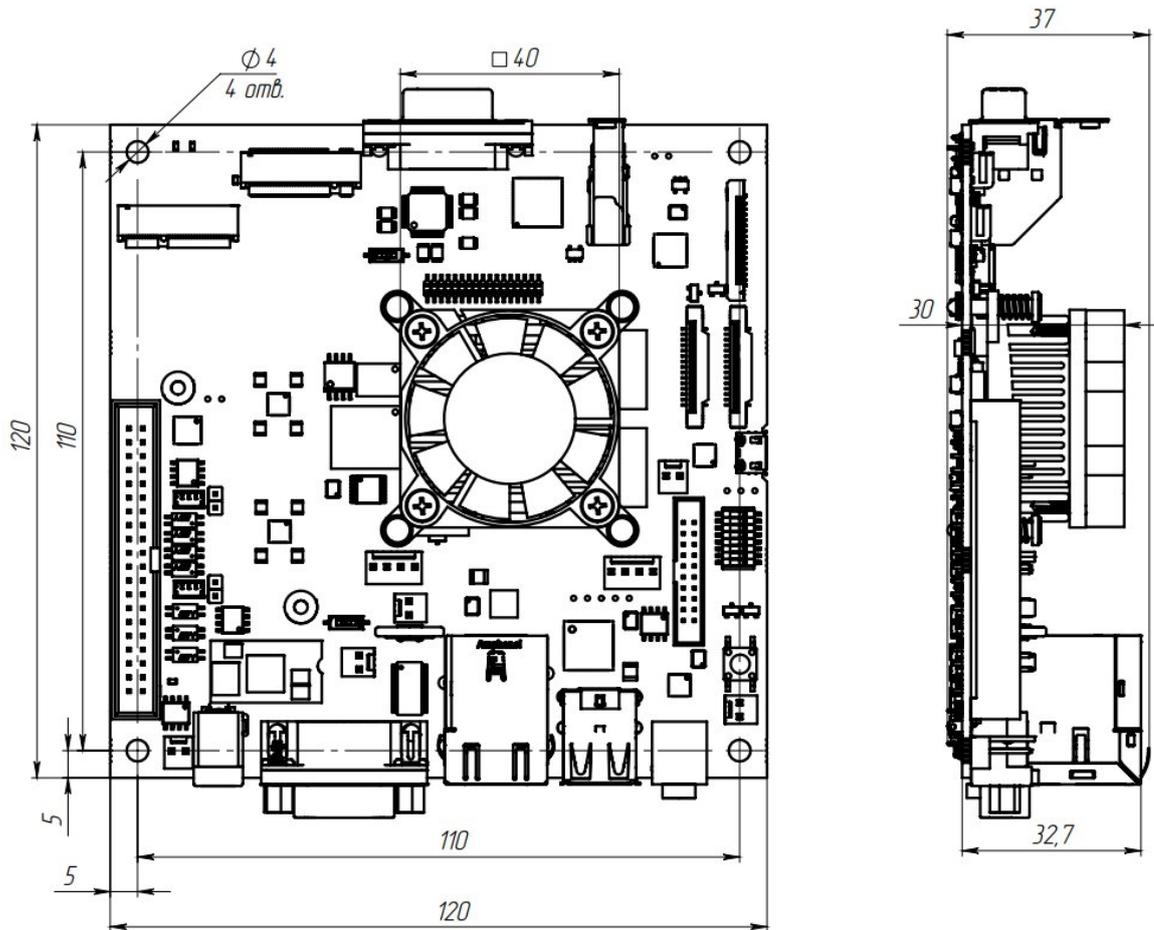


Рисунок 1.21 – Габаритные размеры модуля

1.3 Комплект поставки

Комплект поставки модуля приведен в таблице 1.3

Таблица 1.3 – Комплект поставки

№	Наименование	Кол-во	Примечание
1	Модуль вычислительный	1	NanoS-T или NanoS-T-W
2	Этикетка	1	
3	Руководство по эксплуатации	1	В электронном виде
4	Блок питания 12 В, 1.5А	1	Опционально
5	Wi-Fi модуль с антенной	1	Опционально
6	Программное обеспечение*	-	Сборка Buildroot** или AltLinux

* – ПО представлено «как есть», исключительно для демонстрации возможностей модуля и доступно для скачивания с сайта компании [ООО «Макро EMC»](#)

** – Описание пакетов в сборке **Buildroot** указано в [приложение А](#)

2 Использование по назначению

2.1 Эксплуатационные ограничения

Небрежное или неумелое обращение с модулем, а также нарушение техники безопасности могут привести к несчастным случаям и вызвать выход из строя оборудования.

ЗАПРЕЩАЕТСЯ!

- Проводить настройку модуля с использованием систем отличного от рекомендуемого;
- Проводить изменение конфигурации модуля механическим путем, а также заменой компонентов;
- Использовать модуль не по прямому назначению.

2.2 Карта интерфейсов модуля

На рисунке 2.2.1 изображены расположенные на плате модуля NanoS-T интерфейсы

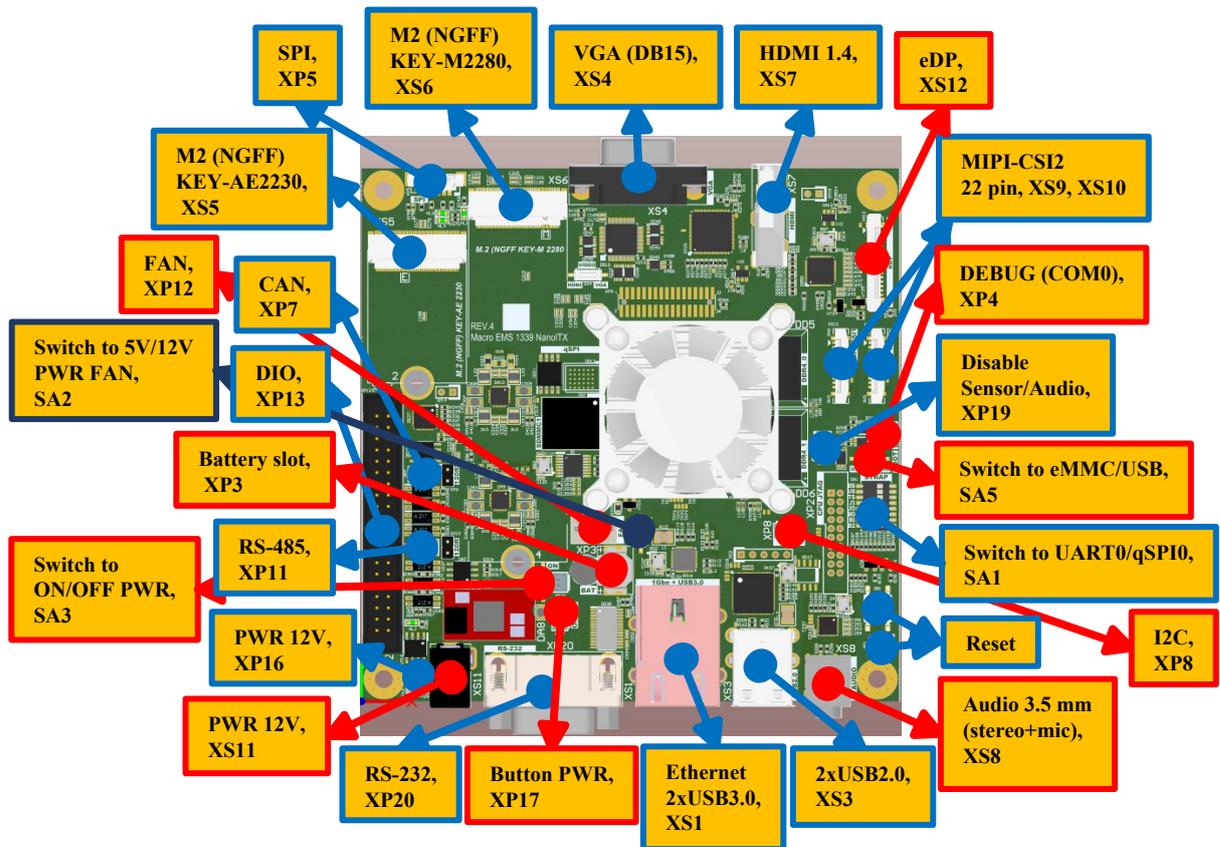


Рисунок 2.2.1 – Интерфейсы модуля

2.3 Подключение модуля

Для начала работы с модулем, выполните следующие действия:

- Подключите клавиатуру в USB разъем модуля;
- Подключите монитор (экран) в разъем VGA (XS4) или HDMI (XS7);
- При необходимости установите SSD накопитель, а также подключите Ethernet кабель в разъем XS1;
- Подключите блок питания в разъем XS11 (2.1x5.5 мм).

2.4 Авторизация в системе

AltLinux

Логин	–	root
Пароль	–	elvees

Buildroot

Логин	–	root
Пароль	–	отсутствует

2.5 Подключение к модулю по сетевому протоколу SSH

2.5.1 Подключение по SSH через командную строку Windows.

2.5.1.1 В Windows с 10-ой версии и старше по умолчанию утилита OpenSSH отключена, и чтобы приступить к выполнению команд, необходимо установить её в настройках:

- Откройте раздел «Параметры», затем раздел «Приложения»;
- Выберите подпункт «Дополнительные компоненты»;
- Найдите в списке «Клиент OpenSSH» и нажмите «Установить».

Если этой кнопки нет, значит, служба уже включена;

- После установки перезагрузите компьютер.

2.5.1.2 Откройте командную строку. Нажмите комбинацию клавиш "Win" + "R". В поисковике введите символы cmd и нажмите клавишу "Enter".

2.5.1.3 В командной строке введите команду и нажмите клавишу "Enter":

```
ssh username@ip-address
```

Вместо `username` введите логин администратора на модуле, по умолчанию в пользовательской Linux `username - root`. Вместо `ip-address` введите IP адрес модуля. Для того, чтобы узнать IP адрес модуля выполните последовательно пункты 2.3, 2.4. В командной строке Linux введите команду и нажмите клавишу «Enter»:

```
ifconfig
```

В выводе команды отобразятся активные сетевые интерфейсы на устройстве. В активном сетевом устройстве, например `end0`, в строке `inet` отобразится IP адрес устройства, например `192.168.98.34`.

2.5.1.4 После запуска команды в консоли система запросит подтверждение подключения, необходимо ввести слово «yes» и нажать клавишу «Enter».

2.5.2 Подключение по SSH через командную строку Linux.

2.5.2.1 Установите утилиту OpenSSH с помощью менеджера пакетов `apt-get`. Откройте консоль программы Терминал нажав комбинацию клавиш «Ctrl» + «Alt» + «Т». Наберите в консоли команду и нажмите клавишу «Enter»:

```
sudo apt-get install openssh-server
```

2.5.2.2 Устанавливаем дополнительные инструменты для работы с сетью и настройками безопасности. Запустите в консоли следующие команды:

```
sudo apt-get install polycoreutils
```

```
sudo apt-get install net-tools
```

2.5.2.3 Проверим статус SELinux, состояние должно быть disable. В открытой консоли наберите команду и нажмите клавишу «Enter»:

```
sestatus
```

Если отобразился статус enabled. Измените параметр SELINUX на SELINUX=disabled в файле конфигурации. Он расположен по следующему пути /etc/selinux/config.

2.5.2.4 Выполните аналогичные действия пунктов 2.5.1.3 - 2.5.1.4.

2.6 Интерфейс USB

2.6.1 Общие сведения

Для реализации 4 портов на модуле используется микросхема Genesys Logic GL3523-OTY30 (4-port hub). В интегральной микросхеме 1892BA018 используется интерфейс USB0.

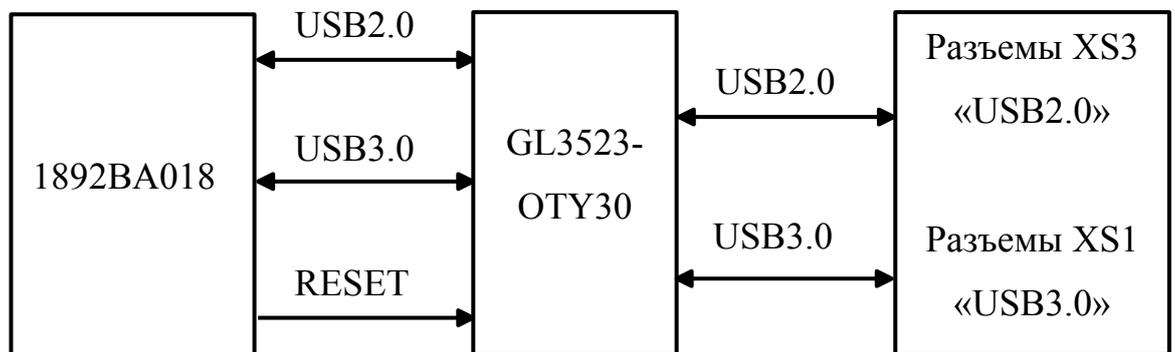


Рисунок 2.6.1 – Структурная схема интерфейса USB

Используемые сигналы подключения:

- Сигналы USB2.0;
- Сигналы USB3.0;
- Сигнал RESET. При подаче сигнала логическая 1 на время более 20мс микросхема GL3523-OTY30 переходит в состояние сброса. После подачи сигнала логический 0 микросхема выходит из состояния сброса и инициализирует свои регистры значениями по умолчанию.

Таблица 2.6.1 – Соответствие сигнала микросхемы GL3523-ОТУ30 и вывода ИМС 1892ВА018

Наименование сигнала	Вывод ИМС 1892ВА018	Описание
RESET	GPIO0_PORTB_0	Сигнал сброса микросхемы GL3523-ОТУ30 из ИМС 1892ВА018

2.6.2 Запуск USB-камеры на модуле

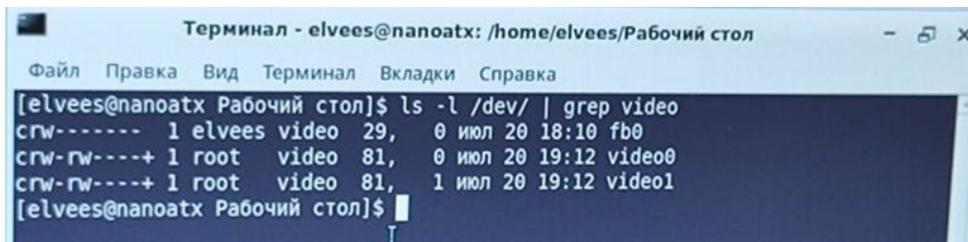
В свободный USB порт модуля подключите USB-камеру.

Убедитесь, что USB-камера определилась в системе Alt Linux, для этого выполните последовательно действия:

- Откройте Терминал в ОС Altlinux. Нажмите комбинацию клавиш «Ctrl» + «Alt» + «Т».
- Введите в консоли Терминал команду и нажмите клавишу «Enter»:

```
ls -l /dev/ | grep video
```

В результате выведутся адреса подключенной камеры, как показано на рисунке 2.6.2.



```

Терминал - elvees@nanoatx: /home/elvees/Рабочий стол
Файл Правка Вид Терминал Вкладки Справка
[elvees@nanoatx Рабочий стол]$ ls -l /dev/ | grep video
crw----- 1 elvees video 29, 0 июл 20 18:10 fb0
crw-rw----+ 1 root video 81, 0 июл 20 19:12 video0
crw-rw----+ 1 root video 81, 1 июл 20 19:12 video1
[elvees@nanoatx Рабочий стол]$

```

Рисунок 2.6.2 – Вывод адресов подключенной USB-камеры

Для вывода видеопотока с USB-камеры наберите команду в командной строке программы Терминал и нажмите клавишу «Enter»:

```
ffplay /dev/video0*
```

* – номер видео порта модуля в каталоге устройств (/dev) может отличаться от написанного видео порта в команде (video0). В случае если вывод не отобразился с порта video0, воспользуйтесь портом video1.

Прервать вывод видеопотока с USB-камеры можно в консоли Терминал, где была запущена команда. Нажмите комбинацию клавиш «Ctrl» + «C».

2.7 Интерфейс DIO

2.7.1 Общие сведения

Для реализации интерфейса DIO на модуле используется микросхема Kinetic Technologies KTS1620ERG-TR (24 ports IO expander). Вывод интерфейса DIO реализован разъемом XP13 - VH-40 (IDC-40MS).

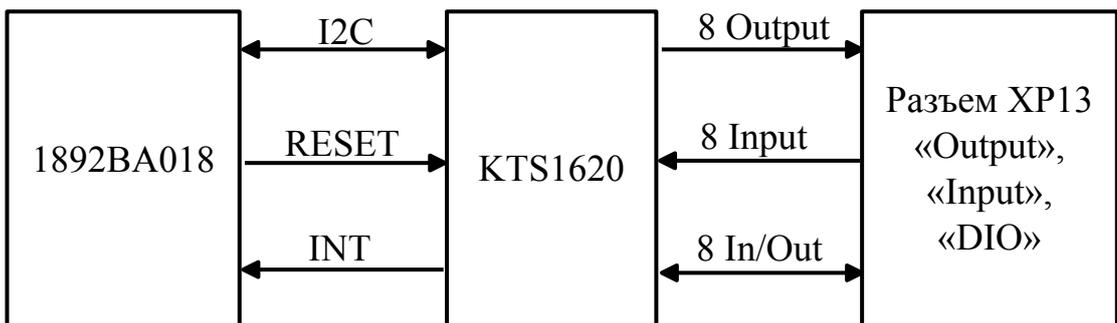


Рисунок 2.7.1 – Структурная схема интерфейса DIO

2.7.2 Цоколевка и схемотехника



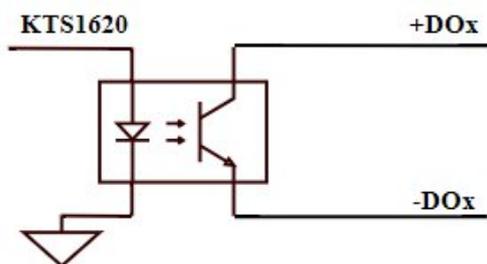
Рисунок 2.7.2 – Цоколевка разъема DIO

Для подключения к разъему XP13 необходимо использовать кабельный разъем IDC-40.

Таблица 2.7.1 – Цоколевка разъема XP13

Номер вывода	Наименование вывода	Описание	Тип
1	Output1 N	Инверсный выход №1	Opto
2	Output1 P	Прямой выход №1	Opto
3	Output2 N	Инверсный выход №2	Opto
4	Output2 P	Прямой выход №2	Opto
5	Output3 N	Инверсный выход №3	Opto
6	Output3 P	Прямой выход №3	Opto
7	Output4 N	Инверсный выход №4	Opto
8	Output4 P	Прямой выход №4	Opto
9	Output5 N	Инверсный выход №5	Opto
10	Output5 P	Прямой выход №5	Opto
11	Output6 N	Инверсный выход №6	Opto
12	Output6 P	Прямой выход №6	Opto
13	Output7 N	Инверсный выход №7	Opto
14	Output7 P	Прямой выход №7	Opto
15	Output8 N	Инверсный выход №7	Opto
16	Output8 P	Прямой выход №8	Opto
17	+12В	Напряжение питание	Питание
18	+5В	Напряжение питание	Питание
19	Input1	Вход №1	Opto
20	Input2	Вход №2	Opto
21	Input3	Вход №3	Opto
22	Input4	Вход №4	Opto
23	Input5	Вход №5	Opto
24	Input6	Вход №6	Opto
25	Input7	Вход №7	Opto
26	Input8	Вход №8	Opto
27	Input COM	Общий вход	Opto
28	Input COM	Общий вход	Opto
29	Reserved	Зарезервированный вывод	-
30	+3.3В	Напряжение питания	Питание
31	Ground	Земля	Земля
32	Ground	Земля	Земля
33	DIO1	Цифровой вход/выход №1	TTL 3,3 В
34	DIO2	Цифровой вход/выход №2	TTL 3,3 В
35	DIO3	Цифровой вход/выход №3	TTL 3,3 В
36	DIO4	Цифровой вход/выход №4	TTL 3,3 В
37	DIO5	Цифровой вход/выход №5	TTL 3,3 В
38	DIO6	Цифровой вход/выход №6	TTL 3,3 В
39	DIO7	Цифровой вход/выход №7	TTL 3,3 В
40	DIO8	Цифровой вход/выход №8	TTL 3,3 В

Схема подключения выходов Output_P/N

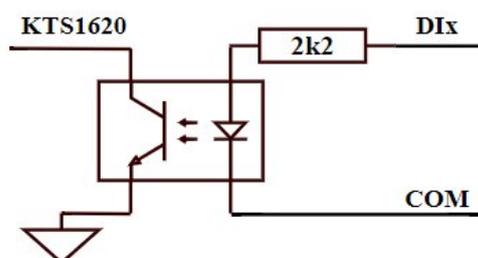


Ограничения выходных сигналов:

Напряжение.....24В DC

Ток.....50mA DC

Схема подключения входов Input

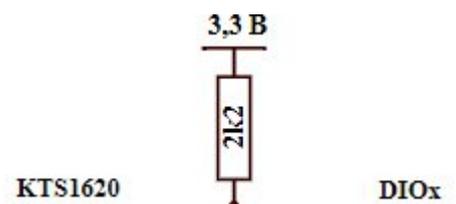


Ограничения входных сигналов:

Напряжение.....24В DC

Ток.....3-10mA DC

Схема подключения входов/выходов DIO



Ограничения цифровых входов/выходов:

Напряжение.....3,3В DC

Ток.....5-10mA DC

Таблица 2.7.2 – Соответствие выходов/входов микросхемы KTS1620 к внешним сигналам, поступающим/приходящим на выводы интерфейса DIO

Наим. вывода KTS1620	Внешний сигнал	Наим. вывода KTS1620	Внешний сигнал	Наим. вывода KTS1620	Внешний сигнал
P0[0]	Input5	P1[0]	Output5	P2[0]	DIO2
P0[1]	Input4	P1[1]	Output6	P2[1]	DIO1
P0[2]	Input2	P1[2]	Output3	P2[2]	DIO4
P0[3]	Output8	P1[3]	Output4	P2[3]	DIO3
P0[4]	Input6	P1[4]	Output1	P2[4]	DIO6
P0[5]	Input3	P1[5]	Output2	P2[5]	DIO5
P0[6]	Input1	P1[6]	Input7	P2[6]	DIO8
P0[6]	Output7	P1[6]	Input8	P2[6]	DIO7

2.7.3 Используемые сигналы подключения DIO

Микросхема KTS1620 управляется по шине I2C (i2c_0). Скорость шины I2C 100/400/1000кГц. Микросхема KTS1620 имеет 7-битный адрес 22h.

Сигнал RESET. При подаче сигнала логический 0 на время более 20мс микросхема KTS1620 переходит в состояние сброса. После подачи сигнала логическая 1 микросхема KTS1620 выходит из состояния сброса и инициализирует свои регистры значениями по умолчанию.

Сигнал INT (сигнал прерывания). При возникновении «событий» в микросхеме KTS1620 данный сигнал переходит в активное состояние: логический 0. Данный механизм требует настройки в KTS1620 в соответствии с описанием.

Таблица 2.7.3 – Соответствия сигналов микросхемы KTS1620 к выводам ИМС 1892BA018

Наименование сигнала	Вывод ИМС 1892BA018	Описание
RESET	GPIO0_PORTD_6	Сигнал сброса из ИМС 1892BA018
INT	GPIO1_PORTA_6	Сигнал прерывания в ИМС 1892BA018
Примечание – прерывание выводов интерфейса DIO не реализовано на аппаратном уровне.		

2.7.4 Реализация DIO в системе Linux

2.7.4.1 Активация драйвера

В системе Linux реализован драйвер `drivers/gpio/gpio-kts1620.c`, модуль драйвера находится в `/lib/modules/5.10.179/kernel/drivers/gpio/gpio-kts1620.ko`.

Драйвер активирован на модуле. В описание дерева устройства (dts) драйвер прописан следующим образом:

```
&i2c0 {
    gpio2: gpio@0x22 {
        compatible = "kinetic,kts1620x-gpio";
        reg = <0x22>;
```

```

status = "okay";
};
};

```

2.7.4.2 Доступ к интерфейсу DIO из командной строки

Необходимо выполнить экспорт ножек микросхемы KTS1620 в Linux для передачи/приема через выводы интерфейса DIO из командной строки или скрипта shell. Согласно приведенной таблицы в разделе схематики, каждая ножка микросхемы получает/передает внешний сигнал через выводы интерфейса DIO.

Таблица 2.7.4 – Соответствия внешних сигналов интерфейса DIO и адреса выводов микросхемы KTS1620 экспортированного в системе Linux

Внешний сигнал, с выводов Opto DI	Адрес вывода микросхемы в Linux	Внешний сигнал, на выходы Opto DO	Адрес вывода микросхемы в Linux	Внешний сигнал, с/на выходы GPIO	Адрес вывода микросхемы в Linux
Input1	430	Output1	436	DIO1	441
Input2	426	Output2	437	DIO2	440
Input3	429	Output3	434	DIO3	443
Input4	425	Output4	435	DIO4	442
Input5	424	Output5	432	DIO5	445
Input6	428	Output6	433	DIO6	444
Input7	438	Output7	431	DIO7	447
Input8	439	Output8	437	DIO8	446

2.7.4.3 Пример использования внешнего сигнала

В качестве примера используется внешний сигнал, поступающий с вывода интерфейса opto DI (Input5).

Для экспорта данного вывода в командной строке Linux наберите команды ниже. Каждая команда должна заканчиваться нажатием клавиши «Enter».

```
export PIN0=424
echo $PIN0 >/sys/class/gpio/export
```

Необходимо настроить направление вывода интерфейса. По умолчанию он установлен на вход. Для того чтобы задать направление вывода воспользуйтесь следующими командами:

Направление на выход:

```
echo out >/sys/class/gpio/gpio$PIN0/direction
```

Направление на вход:

```
echo in >/sys/class/gpio/gpio$PIN0/direction
```

Чтобы прочитать значение вывода интерфейса opto DI воспользуйтесь командой `cat`, представленной ниже:

```
cat /sys/class/gpio/gpio$PIN0/value
```

Если вывод микросхемы выставлен как выход, то установить значение «1» на нем можно командой:

```
echo 1 >/sys/class/gpio/gpio$PIN0/value
```

Или значение «0»:

```
echo 0 >/sys/class/gpio/gpio$PIN0/value
```

2.7.5 Доступ к интерфейсу DIO из C

Доступ осуществляется через new user-space GPIO API, которое использует *character device* `/dev/gpiochipX` и системные вызовы `open()`, `close()`, `ioctl()`, `poll()`, `read()`, `write()`.

Доступ к выводам интерфейса DIO можно получить с помощью библиотеки `libgpiod`. Данная библиотека предоставляет шесть инструментов командной строки:

gpiodetect - список всех присутствующих в системе gpiochips, их названия, метки и количество линий GPIO;

gpioinfo - список всех линий указанных gpiochips, их имена, потребители, направление, активное состояние и дополнительные флаги;

gpioget - чтение значений указанных линий GPIO;

gpioset - установить значения указанных линий GPIO;

gpiofind - найти имя gpiochip и смещение строки по имени строки;

gpiomon - ждать событий на линиях GPIO, указывать какие события смотреть, сколько событий нужно обработать перед выходом или если события должно быть сообщено в консоль.

2.8 Интерфейс RS-232

2.8.1 Общие сведения

В ИМС 1892BA018 используется интерфейс UART3 (COM3).



Рисунок 2.8.1 – Структурная схема интерфейса RS-232

Параметры интерфейса:

- Скорость передачи данных от 9600 Бит/с до 115200 Бит/с
- 8 бит данных.
- 1 стоп бит.
- Контроль четности не поддерживается.
- Управление потоком RTS, CTS, DTR, DSR, DCD, RI.
- Соответствует требованиям стандарта EIA/TIA-232-F

2.8.2 Цоколевка и схемотехника

Для подключения сигналов RS-232 к плате используется разъем XP20. Тип разъема XP20 – DB9 male. Для подключения к разъему XP20 используйте кабельный разъем DB9 female.

2.8.3 Используемые сигналы подключения RS-232

Таблица 2.8.1 – Соответствия сигналов интерфейса RS-232 и выводов ИМС 1892ВА018

Номер вывода	Вывод интерфейса RS-232	Порты вывода ИМС 1892ВА018	Описание
1	DCD	GPIO0_PORTA_4	Сигнал DCD в ИМС 1892ВА018
2	SOUT	GPIO0_PORTB_1	Выход данных TX из ИМС 1892ВА018
3	SIN	GPIO0_PORTB_0	Вход данных RX в ИМС 1892ВА018
4	DTR	GPIO0_PORTA_6	Сигнал DTR из ИМС 1892ВА018
6	DSR	GPIO0_PORTA_3	Сигнал DSR в ИМС 1892ВА018
7	RTS	GPIO0_PORTA_7	Сигнал RTS из ИМС 1892ВА018
8	CTS	GPIO0_PORTA_2	Сигнал CTS в ИМС 1892ВА018
9	RI	GPIO0_PORTA_5	Сигнал RI в ИМС 1892ВА018

2.8.4 Доступ к RS-232 из командной строки

Интерфейс RS-232 является стандартным серийным портом. В системе Linux данный порт соответствует файл-устройство телетайп. RS-232 располагается в каталоге устройств /dev/ с именем ttyS3.

Пример отправки строки на скорости 115200 из shell:

```
stty -F /dev/ttyS3 115200 raw
echo -e "hello world\r\n" >/dev/ttyS3
```

Пример приема строки из shell:

```
cat /dev/ttyS3
```

2.8.5 Доступ к RS-232 из C

Для написания программы на языке C связанной с интерфейсом RS-232 достаточно стандартной библиотеки языка C (libc). В данной библиотеке находятся заголовочные файлы: fcntl.h, termios.h необходимые для работы с данным интерфейсом.

2.9 Интерфейс RS-485

2.9.1 Общие сведения

Для реализации интерфейса RS-485 на модуле используется микросхема Миландр К5559ИН10БСИ (RS-485 driver). Нагрузочный резистор 120 Ом установлен на плате.

В ИМС 1892ВА018 используется интерфейс UART2(COM2).

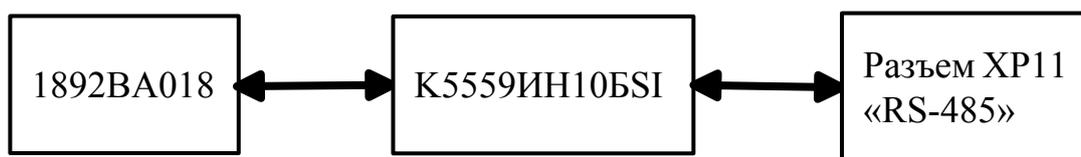


Рисунок 2.9.1 – Структурная схема интерфейса RS-485

Параметры интерфейса соответствуют требованиям стандарта EIA/TIA-RS-485.

2.9.2 Цоколевка и схемотехника

Для подключения сигналов RS-485 к плате используется разъем XP11. Тип разъема XP11 - SCT1251WV-3P (Scondar). Для подключения к разъему XP11 необходимо использовать кабельный разъем SCT1251MH-3P (Scondar). Для подключения нагрузочного резистора 120 Ом необходимо установить джампер MJ-O-6 (2,54мм) в разъем XP10.

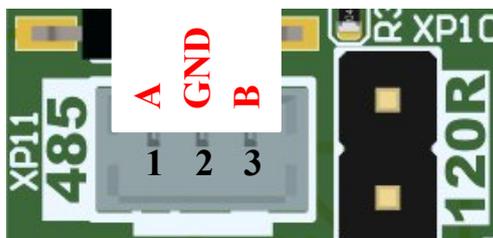


Рисунок 2.9.2 – Цоколевка разъема RS-485

Таблица 2.9.1 – Выводы разъема интерфейса RS-485

Номер вывода	Наименование вывода	Описание
1	A	Прямой вход/выход
2	GND	Земля
3	B	Инверсный вход/выход

2.9.3 Используемые сигналы подключения RS-485

Сигнал DE (активный высокий логический уровень) разрешает работу микросхемы Миландр К5559ИН10БСИ на передачу сигналов в прямой выход А и инверсный выход В с ИМС 1892ВА018. При подаче сигнала в активное состояние логическую «1» на время более 0,7 мкс микросхема Миландр К5559ИН10БСИ переходит в режим передатчика.

Сигнал RE (активный низкий логический уровень) разрешает работу микросхемы Миландр К5559ИН10БСИ на прием сигналов с прямого входа А и инверсного входа В с ИМС 1892ВА018. При подаче сигнала в логический «0» на время более 0,7 мкс микросхема Миландр К5559ИН10БСИ переходит в режим приемника.

Сигнал SIN поступает на вход в ИМС 1892ВА018 с выхода RO микросхемы Миландр К5559ИН10БСИ в режиме приемника.

Сигнал SOUT поступает на вход DI в микросхеме Миландр К5559ИН10БСИ из ИМС 1892ВА018 в режиме передатчика.

Таблица 2.9.2 – Соответствия сигналов интерфейса RS-485 к выводам ИМС 1892ВА018

Наименование сигнала	Порты вывода ИМС 1892ВА018	Описание
DE	GPIO0_PORTB_2	Выход управления микросхемы из ИМС 1892ВА018. Разрешение входа микросхемы в режиме передатчика.
RE	GPIO0_PORTB_3	Выход управления микросхемы из ИМС 1892ВА018. Разрешение выхода микросхемы в режиме приемника.
SIN	GPIO0_PORTB_7	Вход данных в ИМС 1892ВА018
SOUT	GPIO0_PORTD_0	Выход данных из ИМС 1892ВА018

2.9.4 Реализация интерфейса RS-485 в системе Linux

В системе Linux реализован драйвер. Драйвер использует стандартный API ядра Linux для интерфейса RS485. Исходный код драйвера расположен *drivers/tty/serial/8250/8250_dw.c*. Драйвер собран монолитно в ядре Linux и не требует дополнительной загрузки.

2.9.5 Доступ к RS-485 из командной строки

Интерфейс RS-485 является стандартным серийным портом. В системе Linux данный порт соответствует файл-устройство телетайп. RS-485 располагается в каталоге устройств */dev/* с именем *ttyS2*.

Пример отправки строки на скорости 115200 из shell:

```
stty -F /dev/ttyS2 115200 raw
echo -e "hello world\r\n" >/dev/ttyS2
```

Пример приема строки из shell:

```
cat /dev/ttyS2
```

2.9.6 Доступ к RS-485 из C

Для написания программы на языке C связанной с интерфейсом RS-485 достаточно стандартной библиотеки языка C (*libc*). В данной библиотеке

находятся заголовочные файлы: `linux/serial.h`, `sys/ioctl.h` необходимые для работы с данным интерфейсом.

2.10 Интерфейс I2C

2.10.1 Общие сведения

Для реализации внешнего интерфейса I2C на модуле используется прямое подключение к ИМС 1892ВА018 через конвертеры уровня 1,8В-3,3В. На плате установлены pull-up резисторы номиналом 2,49 кОм.

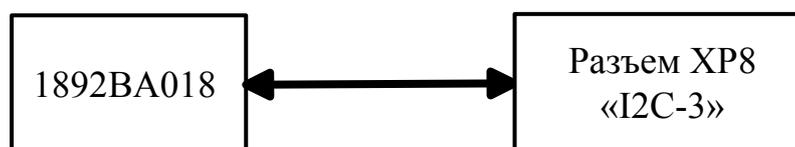


Рисунок 2.10.1 – Структурная схема интерфейса I2C

Параметры интерфейса:

- Скорость передачи данных от 10 кб/с до 500 кб/с;
- 8 бит данных.

2.10.2 Цоколевка и схемотехника

Для подключения устройства по интерфейсу I2C используется разъем XP8. Тип используемого разъема - SCT1251WV-4P (Scondar). Для подключения к разъему XP8 необходимо использовать кабельный разъем SCT1251MH-4P (Scondar).



Рисунок 2.10.2 – Цоколёвка разъема I2C

Таблица 2.10.1 – Выводы разъема интерфейса I2C

Номер вывода	Наименование вывода	Описание
1	3V3	Питание
2	SCL	Линия тактирования
3	SDA	Линия данных
4	GND	Земля

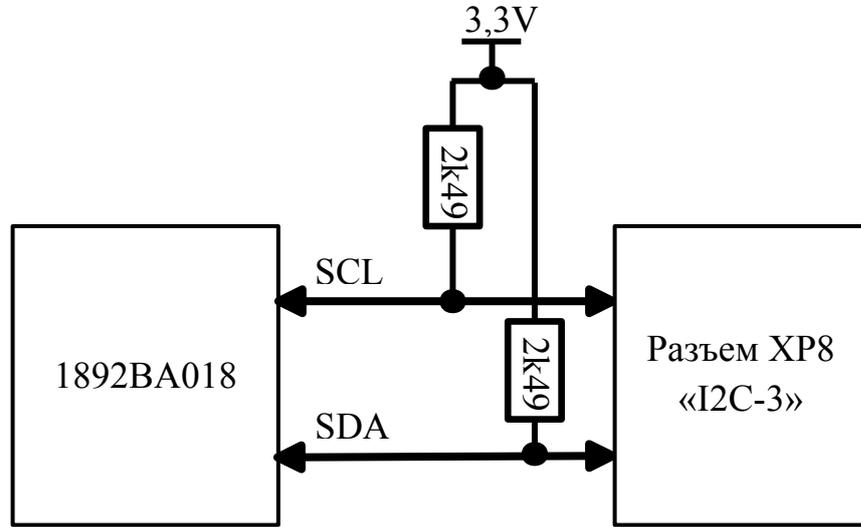


Рисунок 2.10.3 – Схема подключения интерфейса I2C

2.10.3 Реализация интерфейса I2C в системе Linux

Драйвер использует стандартный API ядра Linux шины I2C (`i2c_3`).

2.10.4 Доступ к I2C из командной строки

Шина I2C располагается в каталоге устройств `/dev/` с именем `i2c-3`. Для того чтобы записать или прочитать данные с шины `i2c` можно воспользоваться инструментами командной строки `i2cset` и `i2cget`.

Пример для записи данных на устройство по шине `i2c-3` из shell:

```
i2cset -y 3 0x20 0x01
```

где `-y` – отключение интерактивного режима;

`3` – номер шины `i2c-3`;

`0x20` – адрес устройства, инициализированного на шине `i2c-3`;

`0x01` – данные записанные в адрес устройства `0x20`.

Пример чтения данных с устройства по шине i2c-3 из shell:

```
i2cget -y 3 0x20
```

где -y – отключение интерактивного режима;

3 – номер шины i2c-3;

0x20 – адрес устройства, инициализированного на шине i2c-3.

2.10.5 Доступ к I2C из C

Для написания программы на языке C связанной с интерфейсом I2C достаточно стандартной библиотеки языка C (libc). В данной библиотеке находятся заголовочные файлы: linux/i2c.h, linux/i2c-dev.h, sys/ioctl.h необходимые для работы с данным интерфейсом.

2.11 Интерфейс CAN

2.11.1 Общие сведения

Для реализации интерфейса CAN на модуле используется микросхема Mornsun SCM3425ASA.

В ИМС 1892BA018 используется интерфейс MFBSP0.



Рисунок 2.11.1 – Структурная схема интерфейса CAN

2.11.2 Цоколевка и схемотехника

Для подключения устройства по интерфейсу CAN используется разъем XP7. Тип используемого разъема - SCT1251WV-3P (Scondar). Для подключения к разъему XP7 необходимо использовать кабельный разъем SCT1251MH-3P (Scondar). Для подключения нагрузочного резистора 120 Ом необходимо установить джампер MJ-O-6 (2,54мм) в разъем XP6.

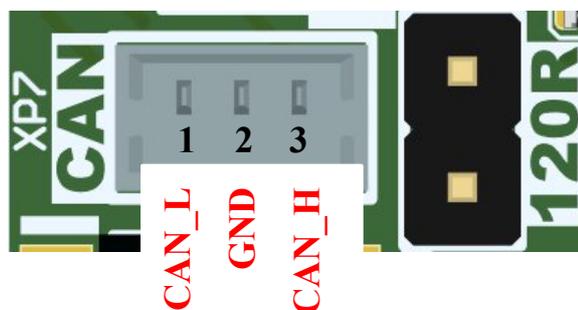


Рисунок 2.11.2 – Цоколевка разъема CAN

Таблица 2.11.1 – Выводы разъема интерфейса RS-485

Номер вывода	Наименование вывода	Описание
1	CAN_L	Линия низкого напряжения
2	GND	Земля
3	CAN_H	Линия высокого напряжения

2.11.3 Используемые сигналы подключения CAN

Таблица 2.11.2 – Соответствие сигналов интерфейса CAN и выводов ИМС 1892BA018

Наименование сигнала	Порты вывода ИМС 1892BA018	Описание
CAN_RX	MFBSP0_LDAT6	Вход данных в микросхему ИМС 1892BA018
CAN_TX	MFBSP0_LDAT7	Выход данных из микросхемы ИМС 1892BA018

2.11.4 Реализация интерфейса CAN в системе Linux

Драйвер использует стандартные API ядра Linux шины CAN (can0).

2.11.5 Доступ к CAN из командной строки

Шина CAN доступна в системе как сетевой интерфейс can0. Для того чтобы записать или прочитать данные с шины can можно воспользоваться утилитами *cansend* и *candump*. Если по каким-то причинам их нет, можно установить пакет *can-utils*.

Пример настройки интерфейса:

Для начала выключите интерфейс. Для этого в командной строке наберите команду и нажмите клавишу «Enter»:

```
ip link set dev can0 down
```

Установите скорость шины равную 125кб/с. Для этого в командной строке наберите команду и нажмите клавишу «Enter»:

```
ip link set can0 type can bitrate 125000
```

Включите интерфейс. Для этого в командной строке наберите команду и нажмите клавишу «Enter»:

```
ip link set dev can0 up
```

Проверьте, что параметры шины CAN установились в системе Linux и в рабочем состоянии. Для этого в командной строке наберите команду и нажмите клавишу «Enter»:

```
ip -details link show can0
```

В случае успешной установки параметров, шина отобразится с установленными параметрами. В противном случае отобразится вывод с ошибками.

Чтобы отправить по шине с именем «can0» данные – ABCDEF99 по адресу 123, в командной строке наберите команду и нажмите клавишу «Enter»:

```
cansend can0 123#ABCDEF99
```

Чтобы прочитать данные по шине «can0», в командной строке наберите команду и нажмите клавишу «Enter»:

```
candump can0
```

2.11.6 Доступ к CAN из C

Для написания программы на языке C связанной с интерфейсом CAN достаточно стандартной библиотеки языка C (libc). В данной библиотеке находятся заголовочные файлы: linux/types.h, linux/socket.h необходимые для работы с данным интерфейсом.

2.12 Интерфейс SPI

2.12.1 Общие сведения

Для реализации интерфейса SPI на модуле используется прямое подключение к ИМС 1892ВА018 через конвертеры уровня 1,8В-3,3В.



Рисунок 2.12.1 – Структурная схема интерфейса SPI

2.12.2 Цоколевка и схемотехника

Для подключения устройства по интерфейсу SPI используется разъем XP5. Тип используемого разъема - SCT1251WV-6P (Scondar). Для подключения к разъему XP5 необходимо использовать кабельный разъем SCT1251MH-6P (Scondar).

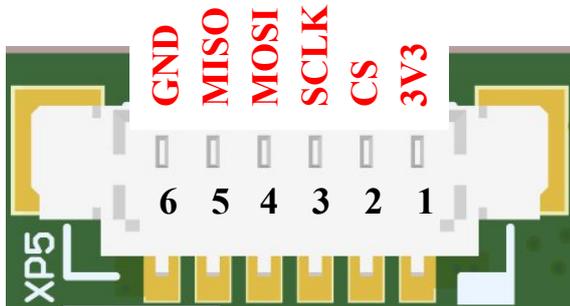


Рисунок 2.12.2 – Цоколевка разъема SPI

Таблица 2.12.1 – Выводы разъема интерфейса SPI

Номер вывода	Наименование	Описание
--------------	--------------	----------

	вывода	
1	3V3	Питание
2	CS	Сигнальная линия выбор ведомого
3	SCLK	Линия тактирования
4	MOSI	Сигнальная линия выхода ведущего устройства, вход ведомого устройства
5	MISO	Сигнальная линия входа ведущего устройства, выход ведомого устройства
6	GND	Земля

2.12.3 Используемые сигналы подключения SPI

Таблица 2.12.1 – Соответствия сигналов интерфейса SPI и выводов ИМС 1892BA018

Наименование сигнала	Порты вывода ИМС 1892BA018	Описание
CS	GPIO0_PORTC_5	Выход данных TX из ИМС 1892BA018
MOSI	GPIO0_PORTC_1	Выход данных из ИМС 1892BA018
MISO	GPIO0_PORTC_2	Вход данных в ИМС 1892BA018

2.12.4 Доступ к SPI из командной строки

Интерфейс SPI располагается в каталоге устройств /dev/ с именем `mtd1`. Для того чтобы записать или прочитать данные по данному интерфейсу можно воспользоваться утилитой `mtd-utils` в которой есть команды `mtd_debug write` и `mtb_debug read`.

Чтобы удостовериться что устройство подключенное по SPI инициализировалась в системе следует набрать команду в командной строке и нажать клавишу «Enter»:

```
mtdinfo /dev/mtd1
```

В результате успешной инициализации отобразится вывод с именем устройства, с типом памяти, размером и к допуску записи в память.

Чтобы отправить данные 0x01 по интерфейсу SPI в адрес 0x0000000 набрать команду в командной строке и нажать клавишу «Enter»:

```
mtd_debug write /dev/mtd1 0 0x01
```

Чтобы прочитать значение по адресу 0x0000000 в командной строке набрать команду и нажать клавишу «Enter»:

```
mtd_debug read /dev/mtd1 0
```

2.12.5 Доступ к SPI из C

Для написания программы на языке C связанной с интерфейсом SPI достаточно стандартной библиотеки языка C (libc). В данной библиотеке находятся заголовочные файлы: linux/serial.h, sys/ioctl.h необходимые для работы с данным интерфейсом.

2.13 Интерфейс MIPI-CSI-2

2.13.1 Общие сведения

Для реализации интерфейса MIPI-CSI-2 используется прямое подключение к MIPI-CSI линиям ИМС 1892BA018. В интегральной микросхеме используются интерфейсы MIPI_CSI0 и MIPI_CSI1.

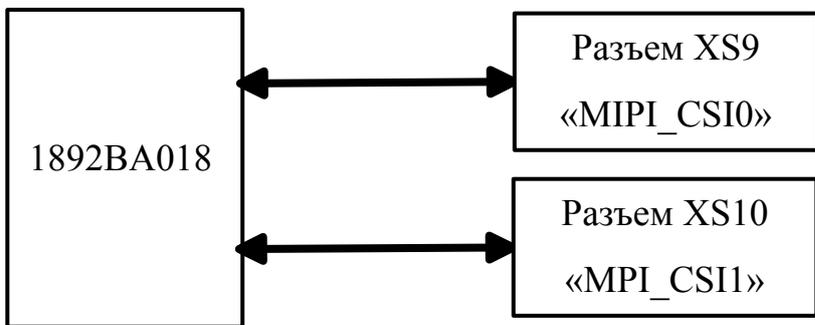


Рисунок 2.13.1 – Структурная схема интерфейса MIPI-CSI-2

2.13.2 Цоколевка и схемотехника

Для подключения МЭК по интерфейсу MIPI-CSI-2 используется разъемы XS9 (MIPI_CSI0) и XS10 (MIPI_CSI1). Тип используемого разъема – FCZ-022-VC50-99/RT (E-tec Interconnect). Для подключения к разъему XS9-

XS10 необходимо использовать стандартный «прямой» 22-контактный FFC-шлейф (тип А) с шагом контактов 0,5 мм.

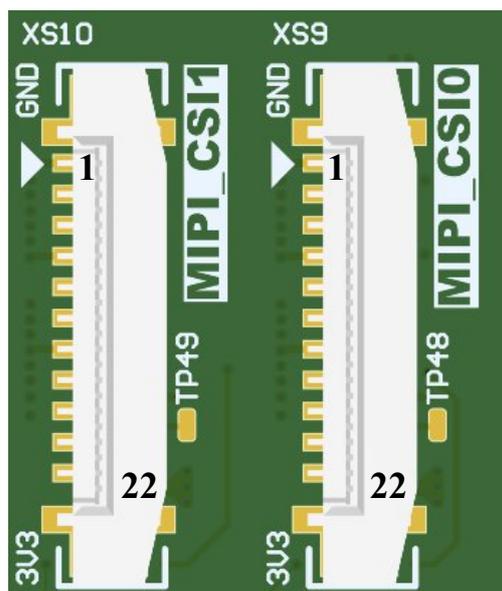


Рисунок 2.13.2 – Цоколевка разъемов MIPI-CSI-2

Таблица 2.13.1 – Выводы разъемов интерфейса MIPI-CSI-2

Номер вывода	Наименование сигнала	Описание
1	GND	Земля
2	CAM_D0_N	Линия видеоданных 0 отрицательный вывод
3	CAM_D0_P	Линия видеоданных 0 положительный вывод
4	GND	Земля
5	CAM_D1_N	Линия видеоданных 1 отрицательный вывод
6	CAM_D1_P	Линия видеоданных 1 положительный вывод
7	GND	Земля
8	CAM_CK_N	Линия тактового сигнала отрицательный вывод
9	CAM_CK_P	Линия тактового сигнала положительный вывод
10	GND	Земля
11	CAM_D2_N	Линия видеоданных 2 отрицательный вывод
12	CAM_D2_P	Линия видеоданных 2 положительный вывод
13	GND	Земля
14	CAM_D3_N	Линия видеоданных 3 отрицательный вывод
15	CAM_D3_P	Линия видеоданных 3 положительный вывод
16	GND	Земля
17	PWR_EN	Сигнальная линия включения/выключения питания камеры
18	LED/XCLK	-
19	GND	Земля
20	CAM_SCL	Линия тактирования по шине I2C
21	CAM_SDA	Линия данных по шине I2C
22	CAM_3V3	Линия питание сенсора +3,3 В

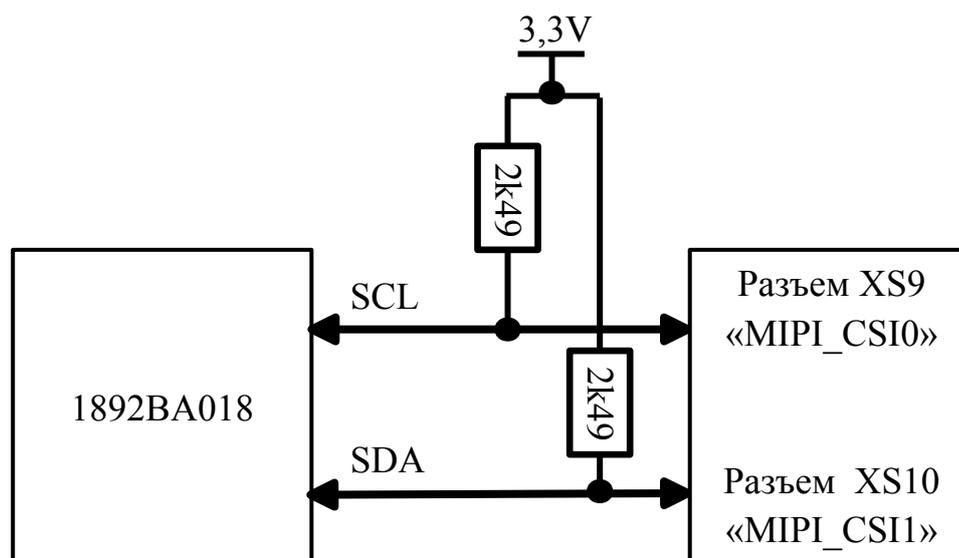


Рисунок 2.13.3 – Схема подключения шины управления I2C для MIPI-CSI-2

2.13.3 Используемые сигналы подключения MIPI-CSI-2

Шина управления МЭК, подключенного к разъему MIPI_CSI0 управляется по шине I2C (i2c_1). Скорость шины I2C зависит от подключенного МЭК. Уровень сигнала по напряжению составляет 3,3 В. На шине управления установлены подтягивающие резисторы 2,49 кОм.

Сигнал MIPI0_PWRen - сигнал включения МЭК, подключенного к разъему MIPI_CSI0. При подаче сигнала логический «0» из ИМС 1892BA018 МЭК включается. При подаче сигнала логическая «1» МЭК выключается. Уровень напряжения сигнала MIPI0_PWRen составляет 3,3 В. На выводе сигнальной линии установлен подтягивающий резистор 2,49 кОм.

Шина управления МЭК, подключенного к разъему MIPI_CSI1 управляется по шине I2C (i2c_2). Скорость шины I2C зависит от подключенного МЭК. Уровень сигнала по напряжению составляет 3,3 В. На шине управления установлены подтягивающие резисторы 2,49 кОм.

Сигнал MIPI1_PWRen - сигнал включения МЭК, подключенного к разъему MIPI_CSI1. При подаче сигнала логический «0» из ИМС 1892BA018 МЭК включается. При подаче сигнала логическая «1» МЭК выключается.

Уровень напряжения сигнала MIPI1_PWRen составляет 3,3 В. На выводе сигнальной линии установлен подтягивающий резистор 2,49 кОм.

Таблица 2.13.2 – Соответствия сигналов интерфейса MIPI-CSI-2 и выводов ИМС 1892BA018

Наименование сигнала	Порты вывода процессора ИМС 1892BA018	Описание
MIPI0_PWRen	GPIO0_PORTC_1	Сигнал включения МЭК из ИМС 1892BA018, подключенного к разъему «MIPI_CSI0» (XS9).
MIPI1_PWRen	GPIO0_PORTC_6	Сигнал включения МЭК из ИМС 1892BA018, подключенного к разъему «MIPI_CSI0» (XS10).

2.13.4 Доступ к интерфейсу MIPI-CSI-2 в системе Buildroot

В системе Buildroot реализована передача видео с МЭК с помощью мультимедийного фреймворка GStreamer. GStreamer поддерживает следующие аппаратные блоки:

- VPU ARM Mail-V61 (Video Processing Unit, видеопроцессор);
- ISP Felix v2505 (Image Signal Processor, процессор обработки изображений);
- GPU PowerVR Graphics Series8XE (Graphics Processing Unit, графический ускоритель).

Захват и обработка видео с МЭК с использованием ISP поддерживается плагином `gst-felix` для пакета GStreamer. Данным плагином поддерживается элемент `felixsrc` – захват видео с МЭК. Элементом `felixsrc` не поддерживается работа двух МЭК одновременно.

Конфигурационный файл взаимодействия модуля с МЭК располагается в файловой системе модуля по следующему пути `/etc/felix/boardcfd/default.cfg`. Перечень возможных установочных файлов (`setup-file`) в зависимости от используемого МЭК расположены в директории `/etc/felix/`.

Таблица 2.13.3 – Расположение установочных файлов поддерживаемых МЭК в системе Buildroot, разъем подключения на модуле и рекомендованные режимы работы МЭК

Модель МЭК	Расположение установочного файла	Разъем модуля	Режимы работы МЭК*
DS-CIMX327-22	/etc/felix/imx327/imx327.cfg	MIPI_CSI0 (XS9)	0, 1
DS-CIMX335-22	/etc/felix/imx335/imx335.cfg /etc/felix/imx335/imx335-noir.cfg	MIPI_CSI0 (XS9)	5, 6, 7, 8
DS-CIMX415-22	/etc/felix/imx415/imx415.cfg	MIPI_CSI0 (XS9)	0, 4, 5, 7, 8, 9, 11, 12
DS-CIMX662-22	/etc/felix/imx662/imx662.cfg /etc/felix/imx662/imx662-noir.cfg	MIPI_CSI0 (XS9)	0, 1, 2
* - данные режимы поддерживают вывод изображения на экран с разрешением Full HD с максимальным количеством кадров 30 (FPS). Порядковый номер режима работы МЭК взят из утилиты felix-sensor-test.			

2.13.5 Запуск МЭК в системе Buildroot

Мы рекомендуем использовать МЭК серии: DS-CIMX335-22, DS-CIMX415-22, DS-CIMX327-22, DS-CIMX662-22. Рекомендованные МЭК могут быть подключены только к разъему «MIPI_CSI0» (XS9 на плате модуля).

Для начала работы с МЭК выполните следующие действия:

- 1) Подключите МЭК в разъем модуля «MIPI_CSI0» (XS9) через стандартный «прямой» 22-контактный FFC-шлейф (тип А) с шагом контактов 0,5 мм.
- 2) Выполните последовательно действия пункта 2.3 [Подключение модуля](#).
- 3) Войдите в среду Linux. Наберите в командной строке логин **root** и нажмите клавишу «Enter».

Убедитесь, что МЭК инициализирован в системе. Запустите утилиту `felix-sensor-test` набрав команду в командной строке и нажав клавишу «Enter»:

felix-sensor-test

Запущенная утилита проверит подключение всех поддерживаемых платформой МЭК и выведет их статус. Если МЭК правильно определен системой и доступен для видеозахвата ответом на команду будут следующие строки:

```
...
18: IMX335 (v0x8806 imager 0)
mode 0: 2592x 1944 @60.00 10bit (total 275x4500 mipi_lane=4)
exposure=(3..1000000) flipping=horizontal|vertical
pixel rate 37.1250 Mpx/s, bit rate 92.8125 Mbits/s (per mipi lane)
mode 1: 2592x 1944 @30.00 12bit (total 550x4500 mipi_lane=4)
exposure=(7..1000000) flipping=horizontal|vertical
pixel rate 37.1250 Mpx/s, bit rate 111.3750 Mbits/s (per mipi lane)
```

В качестве примера приведена часть вывода утилиты. Цифра перед словом `mode` является порядковым номером режима работы МЭК.

В случае, если МЭК не определен системой, то для него ответом на команду `felix-sensor-test` будет следующим:

X: IMX335 – no modes display available

Запустите захват видео с МЭК.

Общий вид команды запуска видеотрансляции с выводом изображения на монитор с разрешением Full HD через интерфейс HDMI с постоянной частотой кадров:

```
gst-launch-1.0 felixsrc setup-file=<setup-file> sensor=<sensor> sensor-
mode=<sensor-mode> exposure-auto=true exposure-auto-priority=1 awb-
enable=true awb-algorithm=pid awb-mode=high-lum ! queue max-size-
buffers=1 ! video/x-raw,format=BGR,width=1920,height=1080 ! fpsdisplaysink
video-sink="kmssink driver-name=mali-dp max-lateness=-1 force-
modesetting=true" -v 2>&1
```

Пример запуска видеотрансляции с МЭК DS-CIMX662-22 с выводом изображения на монитор с разрешением Full HD через интерфейс HDMI:

```
gst-launch-1.0 felixsrc setup-file=/etc/felix/imx662/imx662.cfg
sensor=IMX662 sensor-mode=0 exposure-auto=true exposure-auto-max-
time=60000 exposure-auto-min-time=10 exposure-auto-priority=1 awb-
enable=true awb-algorithm=pid awb-mode=high-lum preenq-buffers=1 use-
dmabuf=true restart-on-error=true alloc-buffers=6 ! video/x-
raw,format=BGR,width=1920,height=1080 ! queue ! fpsdisplaysink video-
sink="kmssink driver-name=mali-dp max-lateness=-1 force-modesetting=true" -v
2>&1
```

Чтобы прервать видеотрансляцию нажмите комбинацию клавиш «Ctrl» + «С». После остановки команды в Терминале выведется FPS видеотрансляции (количество потерянных кадров, моментальное и среднее значения).

Для вывода свойств элемента felixsrc воспользуйтесь описанной ниже командой. У данных свойств будет описан тип значения, значение, установленное по умолчанию и диапазон возможных принимаемых значений.

```
gst-inspect-1.0 felixsrc
```

Значение максимальной и минимальной выдержки (exposure-auto-max-time, exposure-auto-min-time) для каждого МЭК свое. Желаемое максимальное время выдержки для МЭК можно рассчитать по формуле:

$$\text{exposure-auto-max-time} = 1000000 / \text{желаемое_FPS}$$

Параметр `sensor-mode` должен соответствовать разрешению устройства видеовывода. Для вывода доступных режимов устройства видеовывода можно воспользоваться командой:

```
modetest -M mali-dp -c
```

Для принудительного масштабирования захватываемого видео под устройство видеовывода можно задать разрешение видеопотока для вывода, например: `video/x-raw,format=BGRx,width=1920,height=1080`

Общий вид команды запуска потоковой передачи видео по протоколу RTSP:

```
gst-rtsp-test-launch "felixsrc setup-file=<setup-file> sensor=<sensor>
sensor-mode=<sensor-mode> alloc-buffers=10 buf-mode=query exposure-
auto=true awb-enable=true awb-algorithm=pid awb-mode=high-lum ! queue max-
size-buffers=1 ! video/x-raw,format=NV12 ! omxh264enc control-rate=constant
target-bitrate=10000000 ! rtph264pay name=pay0 pt=96"
```

Пример запуска сенсора DS-CIMX662-22 с потоковой передачей видео по протоколу RTSP:

```
gst-rtsp-test-launch "felixsrc setup-file= /etc/felix/imx662/imx662.cfg
sensor=IMX662 sensor-mode=0 alloc-buffers=10 buf-mode=query exposure-
auto=true awb-enable=true awb-algorithm=pid awb-mode=high-lum ! queue max-
size-buffers=1 ! video/x-raw,format=NV12 ! omxh264enc control-rate=constant
target-bitrate=10000000 ! rtph264pay name=pay0 pt=96"
```

В консоль процессорного модуля будет выведено сообщение:

```
stream ready at rtsp://127.0.0.1:8554/test
```

Для приёма и вывода видео с сенсора на ПК подайте в консоли Терминал команду `ffplay` в формате:

```
ffplay rtsp://<module-address>:8554/test
```

где `<module-address>` - это IP-адрес процессорного модуля.

2.14 Последовательный интерфейс UART0

2.14.1 Общее описание

Для реализации последовательного интерфейса на модуле используется прямое подключение к ИМС 1892ВА018 через конвертеры уровня 1,8В-3,3В. Для подключения необходимо использовать преобразователь UART-USB.

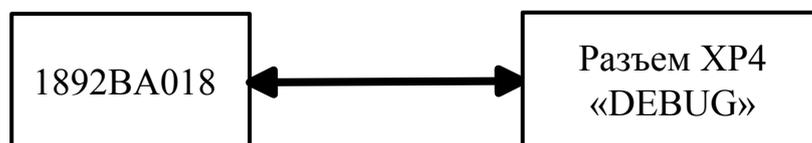


Рисунок 2.14.1 – Структурная схема интерфейса UART0

2.14.2 Цоколевка и схемотехника разъема UART0

Для подключения сигналов UART0 к плате используется разъем XP4. Тип разъема XP2 – SCT1251WV-3P (Scondar). Для подключения к разъему XP4 необходимо использовать кабельный разъем SCT1251MH-3P (Scondar).

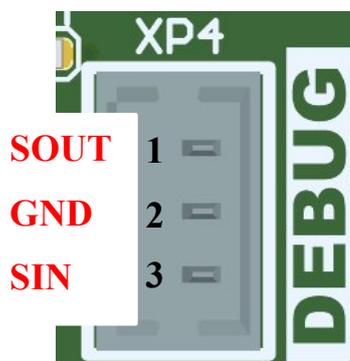


Рисунок 2.14.2 – Цоколевка разъема UART0

Таблица 2.14.1 – Цоколевка разъема XP5

Номер вывода	Наименование вывода
1	SOUT
2	GND
3	SIN

2.14.3 Используемые сигналы подключения UART0

Таблица 2.14.2 – Соответствия сигналов интерфейса UART0 и выводов ИМС 1892BA018

Номер вывода	Вывод интерфейса UART0	Порты вывода ИМС 1892BA018	Описание
1	SOUT	GPIO1_PORTB_6	Выход данных TX из ИМС 1892BA018
3	SIN	GPIO1_PORTB_7	Вход данных RX в ИМС 1892BA018

3 Возможные неисправности и методы их устранения

3.1 Особенности работы интегральной микросхемы 1892BA018

Работа интерфейсов модуля базируется на работе микросхемы 1892BA018 СнК «СКИФ». В данной версии модуля используются инженерные образцы данной микросхемы, которые обуславливают особенности его работы. Специалисты АО НПЦ «Элвис» планируют исправить аппаратную часть микросхемы в коммерческих версиях. В таблице 3 представлены особенности работы модуля (неисправности) на инженерных образцах микросхемы и способы обхода данных особенностей.

Таблица 3.1 – Неисправности и методы их решений

Неисправность	Методы устранения
<p>HDMI 1.4 Срыв синхронизации, в результате чего возникают искажение или мерцание экрана</p>	<p>Решение 1: <u>ПО Alt Linux</u> Запустить скрипт с рабочего стола «HDMI turning». В открывшейся консоли терминала с вопросом хорошего изображения картинка, нажимать клавишу «n» (нет), пока не появится хорошее изображение. Когда появится хорошее изображение на экране монитора, нажать клавишу «y» (да).</p> <p><u>ПО Buildroot</u> Запустить скрипт «mcom03-hdmi-setup.sh»: Набрать в командной строке системы Buildroot ./mcom03-hdmi-setup.sh и нажав клавишу «Enter». В результате запуска скрипта на экране монитора появится изображение (зелено-синее) для проверки стабильности вывода по интерфейсу. Нажимать клавишу «n», пока не появится хорошее изображение на экране монитора. Когда появится хорошее изображение на экране монитора, нажать клавишу «y».</p> <p>Решение 2: Перезапустите модуль</p>

3.2 Служба поддержки и полезные ссылки

3.2.1 Канал поддержки

Nano_Pico_ITX_support – ODM@macrogroup.ru

П р и м е ч а н и е – Для добавления в канал поддержки, пожалуйста, сообщите вашему менеджеру (от Макро Групп) ваш домен почты.

3.2.2 Полезные ссылки

- [Комплект для разработки ПО](#)
- [Готовые сборки образов ОС Linux](#)
- [Архив с готовыми образами пользовательской Linux и инструкции по сборке и установке собственного образа среды Linux](#)

Информация по установке и переустановке ОС находится на [странице продукта](#).

4 Условия хранения

Модуль вычислителя должен храниться в сухих проветриваемых помещениях при нормальных климатических условиях:

- температура окружающей среды от 15 °С до 35 °С;
- атмосферное давление от 645 до 795 мм рт.ст.;
- относительная влажность воздуха от 45 % до 75 %.

5 Гарантия производителя

ООО «Макро ЕМС» гарантирует соответствие качества модуля при соблюдении потребителем условий эксплуатации и хранения.

Гарантийный срок – 12 мес.

Срок службы – 36 мес.

ООО «Макро ЕМС» снимает свои гарантийные обязательства при наличии на изделии механических повреждений деталей модуля (в том числе следы замены, перепайки компонентов).

Приложение А – Список пакетов в сборке образа системы Buildroot

Общее описание пакетов в сборке Buildroot

Установленные библиотеки располагаются в пути `/usr/lib/` на файловой системе модуля. Сборка ОС Buildroot собиралась из внутренних и внешних пакетов, представленных в списке.

Внутренние пакеты – пакеты, установленные через Manager Packages.

Внешние пакеты – пакеты, установленные через сторонние ресурсы. В состав внешних пакетов входят:

Пакет `opencv_contrib`: `intensity_transform`, `line_descriptor`, `mcc`, `optflow`, `ovis`, `phase_unwrapping`, `plot`, `quality`, `rapid`, `text`, `videostab`, `viz`, `xfeatures2d`, `xobjdetect`, `xphoto`.

Пакет `nginx`: `nginx-rtmp-module`.

Условные обозначения в списке пакетов:

Жирный шрифт – разделы пакетов;

Символ “→” – переход на уровень ниже.

Таблица А - Список пакетов

Audio and video applications:	<code>videoconvert</code>
<code>alsa-utils</code>	<code>gio</code>
<code>ffmpeg</code>	<code>gio-typefinder</code>
<code>gstreamer 1.x→</code>	<code>playback</code>
enable unit test libraries	<code>audioresample</code>
enable command-line parser	<code>rawparse</code>
enable tracing subsystem	<code>subparse</code>
enable gst-debug trace support	<code>tcp</code>
enable plugin registry	<code>typefind</code>
install tools	<code>videotestsrc</code>
<code>gst1-plugins-base→</code>	<code>videorate</code>
app	<code>videoscale</code>
audioconvert	<code>volume</code>

auidomixer	alsa
audiorate	ogg
theora	jp2kdecimator
vorbis	jpegfromat
gst1-plugins-bayer2rgb-neon	librfb
gst1-plugins-good→	midi
jpeg	mpegdemux
png	mpegtsdemux
avi	mpegtsmux
isomp4	mpegpsmux
law	mxf
matroska	netsim
multifile	onvif
rtp	pcapparse
rtpmanager	pnm
rtsp	proxy
udp	rawparse
videobox	removesilence
videocrop	rist
videofilter	rtmp2
videomixer	rtp
wavenc	rtmp
wavparse	sdp
v4l2	segmentclip
gst1-plugins-bad →	siren
adpcmdec	smooth
aiff	speed
asfmux	subenc
audiobuffersplit	switchbin

<p> audiofxbad audiolateny audiomixmatrix audiovisuaalizers autoconvert bayer debugutils dvbsubenc dvbsuboverlay dvdspu frei0r gaudieffects geometrictransform gdp id3tag inter interlace ivtc mpv musepack v4l2grab v4l2loopback Compressors and decompressors: bzip2 Debugging, profiling and benchmark: fio gdb ramspeed </p>	<p> videofilters videoframe-audiolevel videoparsers videosignal vmnc y4m hls kmssink mpeg2enc musepack neon openh264 webp webrtc webrtcdsp gst1-plugins-ugly → mpeg2dec mjpegtools ubirename ubirmvol ubirsvol ubiupdatevol ubiblock MTD test tools Fonts, cursors, icons, sounds and themes: DejaVu fonts→ mono fonts sans fonts </p>
---	--

<p>stress-ng</p> <p>tinymembench</p> <p>Developments tools:</p> <p>make</p> <p>Filesystem and flash utilities:</p> <p>dosfstools</p> <p>fatlabel</p> <p>fsck.fat</p> <p>mkfs.fat</p> <p>e2fsprogs→</p> <p> debugfs</p> <p> e2image</p> <p> e4defrag</p> <p> fuse2fs</p> <p> resize2fs</p> <p>mtd, jffs2 and ubi/ubifs tools</p> <p>flashcp</p> <p>flash_lock</p> <p>flash_unlock</p> <p>mkfs.ubifs</p> <p>mtd_debug</p> <p>nanddump</p> <p>nandtest</p> <p>nandwrite</p> <p>sumtool</p> <p>mtdinfog</p> <p>ubiattach</p> <p>ubicrc32</p> <p>ubidetach</p>	<p>serif fonts</p> <p>sans condensed fonts</p> <p>serif condensed fonts</p> <p>font-awesome</p> <p>Liberation (free fonts) →</p> <p> mono fonts</p> <p> sans fonts</p> <p> serif fonts</p> <p>Graphic libraries and applications (graphic/text):</p> <p>sdl2</p> <p>Qt5→</p> <p> qt5base</p> <p> concurrent module</p> <p> gui module</p> <p> widgets module</p> <p> linuxfb support (\$ export QT_QPA_PLATFORM=linuxfb for activation)</p> <p> DBus module</p> <p> qt5connectivity</p> <p> qt5enginio</p> <p> qt5multimedia</p> <p>Hardware handling:</p> <p>Firmware→</p> <p> linux-firmware</p> <p> Video firmware→</p> <p> Lontium LT9611UXC HDMI transceiver firmware</p>
--	---

ubiformat ubihealthd ubimkvol ubinfo ubinize dbus evtest gpsd → NMEA hdparm hwdata→ install pci.ids install usb.ids i2c-tools lm-sensors→ sensors memtester parted pciutils smartmontools u-boot tools→ fw_printenv mcom03-utils usbutils Interpreter languages and scripting: nodejs→ NPM for the target python3 External python modules→	WiFi firmware→ Atheros 10k (QCA9377) Qualcomm Atheros 6174 Ethernet firmware→ Realtek 8169 Crypto→ gnutls libgcrypt libgpg-error nettle openssl support Filesystem→ libfuse Graphics→ bayer2rgb-neon cairo→ pdf support png support svg support fontconfig freetype harfbuzz jpeg support libdrm→ Install test programs libpng libsvg libsvg-cairo opencv4→
---	---

django	calib3d
pip	features2d
Libraries:	highgui
Audio/Sound→	gui toolkit (qt5)
alsa-lib→	imgcodecs
aload	imgproc
mixer	ml
pcm	objdetect
rawmidi	python
hwdep	shape
seq	stitching
ucm	videoio
alisp	video
old-symbols	opencv-contrib
libcuefile	gstreamer-1.x
libreplayagain	jpeg support
libvorbis	png support
webrtc-audio-processing	v4l support
Compression and decompression→	pixman
lzo	webp
zlib support	Hardware handling→
libaio	boost→
libgpod→	Layout (system)
install tools	boost-atomic
libiio→	boost-chrono
Local backend	boost-date_time
Install test programs	boost-filesystem
libnfc→	boost-locale
arygon driver	boost-log

pn532_uart driver	boost-regex
pn53x_usb driver	boost-system
build libnfc examples	boost-thread
libqmi	elfutils
libusb	gmp
libusb-compat	gobject-introspection
libv4l	libcap
tslib	libcap-ng
Javascript→	libffi
angularjs→	libglib2
External AngularJS plugins→	liblinear
angular-websocket	libpthread-stubs
vuejs	libtasn1
vuejs-router	Text and terminal handling→
JSON/XML→	fmt
expat	libfribidi
tinycl2	ncurses→
yaml-cpp	ncurses programs
Logging→	newt
spdlog	pcre
Multimedia→	popt
libass	readline
libmpeg2	slang
libogg	Miscellaneous:
libopenh264	collectd→
libtheora	misc plugins→
Networking→	logfile
c-ares	syslog
libndp	read plugins→

<p>libnice</p> <p>libnl</p> <p>liboping</p> <p>libpcap</p> <p>libneon</p> <p>nghttp2</p> <p>librtmp</p> <p>Other→</p> <p>bluez-utils→</p> <p> build OBEX support</p> <p> build CLI client</p> <p> build monitor utility</p> <p> build tools</p> <p> build audio plugins (a2dp and avrcp)</p> <p> build hid plugin</p> <p> build hog plugin</p> <p> build network plugin</p> <p>can-utils</p> <p>dhcpcd</p> <p>ethtool→</p> <p> enable pretty printing</p> <p>ifmetric</p> <p>iperf4</p> <p>iproute2</p> <p>iptables</p> <p>lrzsz</p> <p>modem-manager→</p> <p> QMI support</p>	<p>cpu</p> <p>interface</p> <p>memory</p> <p>ping</p> <p>write plugins→</p> <p>network</p> <p>write_log</p> <p>Netwrking applications:</p> <p> ngx_http_limit_req_module</p> <p> ngx_http_empty_gif_module</p> <p> ngx_http_browser_module</p> <p> ngx_http_upstream_ip_-</p> <p>hash_module</p> <p> ngx_http_upstream_-</p> <p>least_conn_module</p> <p> ngx_http_upstream_-</p> <p>keepalive_module</p> <p> ngx_http_upstream_-</p> <p>random_module</p> <p>nmap→</p> <p> install ncat</p> <p>openssh</p> <p> client</p> <p> server</p> <p> key utilites</p> <p>wget</p> <p>wireless tools→</p> <p> Install shared library</p> <p>wpa_supplicant</p>
--	---

<p>network-manager→</p> <ul style="list-style-type: none"> nmtui support modem-manager support <p>nginx→</p> <ul style="list-style-type: none"> nginx-rtmp-module <p>http server→</p> <ul style="list-style-type: none"> ngx_http_charset_module ngx_http_gzip_module ngx_http_ssi_module ngx_http_userid_module ngx_http_access_module ngx_http_auth_basic_module ngx_http_autoindex_module ngx_http_geo_module ngx_http_map_module ngx_http_split_clients_module ngx_http_referer_module ngx_http_rewrite_module ngx_http_proxy_module ngx_http_fastcgi_module ngx_http_uwsgi_module ngx_http_scgi_module ngx_http_memcached_module ngx_http_limit_conn_module <p>fsck</p> <p>hwclock</p> <p>mount/umount</p> <p>scheduling utilities</p>	<p>Shell and utilites:</p> <ul style="list-style-type: none"> bash <p>System tools:</p> <ul style="list-style-type: none"> coreutils <p>kmod→</p> <ul style="list-style-type: none"> kmod utilities <p>systemd→</p> <ul style="list-style-type: none"> enable pstore support enable hwdb installation enable myhostname NSS plugin enable network manager enable resolve daemon enable timedate daemon enable timesync daemon enable tmpfiles support enable vconsole tool <p>util-linux→</p> <ul style="list-style-type: none"> libblkid libfdisk libmount libsmartcols libuuid basic set agetty uuid <p>Text editors and viewers:</p> <ul style="list-style-type: none"> nano→ optimize for size
--	---

Приложение Б – Демо приложение SmartCamApp

Приложение SmartCamApp разработано для ознакомительных целей и демонстрирует запуск трансляции по HDMI с модуля электронной камеры с нейросетевой обработкой по распознаванию лица и оружия.

Таблица Б – МЭК и их режимы работы, поддерживаемые приложением SmartCamApp

Модель МЭК	Режимы работы МЭК*
DS-CIMX327-22	0
DS-CIMX335-22	5
DS-CIMX415-22	0
DS-CIMX662-22	0

* - данные режимы поддерживают вывод изображения на экран с разрешением Full HD с максимальным количеством кадров 30 (FPS). Порядковый номер режима работы МЭК взят из утилиты felix-sensor-test.

Запуск приложения SmartCamApp на модуле.

1. Подключите МЭК к модулю используя разъем «MIPI_CSI0» (XS9) через стандартный «прямой» 22-контактный FFC-шлейф (тип А) с шагом контактов 0,5 мм.
2. Выполните последовательно действия пункта 2.3 [Подключение модуля](#).
3. Войдите в среду Linux. Наберите в командной строке логин **root** и нажмите клавишу «Enter».
4. В командной строке среды Linux наберите команду и нажмите клавишу «Enter»:

```
app_rtsp -s <номер_подключенного_МЭК>
```

Пример запуска приложения SmartCamApp с МЭК DS-CIMX662-22:

```
app_rtsp -s 662
```

В результате запуска приложения начнётся видеотрансляция МЭЖ с обработкой нейросети. В правом нижнем углу видеотрансляции будет отображаться FPS и разрешение. Чтобы прервать работу приложения SmartCamApp нажмите комбинацию клавиш «Ctrl» + «С».

