



МАКРО EMC
Группа компаний Макро Групп

Модуль вычислительный

PicoS

Руководство по эксплуатации

МРЦН.467444.002РЭ

PicoS-T



PicoS



PicoS-1K



Содержание

Перечень сокращений.....	5
1 Описание и работа.....	7
1.1 Назначение изделия	7
1.2 Технические характеристики.....	10
1.3 Комплект поставки.....	13
2 Использование по назначению	14
2.1 Эксплуатационные ограничения	14
2.2 Карта интерфейсов модуля	14
2.3 Подключение модуля.....	16
2.3.1 Модуль версии PicoS-T, PicoS	16
2.3.2 Модуль версии PicoS-1K	16
2.4 Авторизация в ПО	17
2.5 Подключение к модулю по сетевому протоколу SSH.	17
2.5.1 Подключение по SSH через командную строку Windows.....	17
2.5.2 Подключение по SSH через командную строку Linux.....	18
2.6 Интерфейс USB	19
2.6.1 Общие сведения	19
2.6.2 Запуск USB-камеры на модуле PicoS-T, Pico-S	20
2.7 Интерфейс DIO.....	21
2.7.1 Общие сведения	21
2.7.2 Цоколёвка и схемотехника.....	21
2.7.3 Используемые сигналы подключения DIO	24
2.7.4 Реализация DIO в ОС Linux	25
2.7.5 Доступ к интерфейсу DIO из С.....	27
2.8 Интерфейс RS-232.....	28
2.8.1 Общие сведения.....	28
2.8.2 Цоколёвка и схемотехника.....	28
2.8.3 Используемые сигналы подключения RS-232	29
2.8.4 Доступ к RS-232 из командной строки	29
2.8.5 Доступ к RS-232 из С.....	30
2.9 Интерфейс RS-485.....	30
2.9.1 Общие сведения	30
2.9.2 Цоколёвка и схемотехника.....	30

2.9.3	Используемые сигналы подключения RS-485	31
2.9.4	Реализация интерфейса RS-485 в ОС Linux	32
2.9.5	Доступ к RS-485 из командной строки	32
2.9.6	Доступ к RS-485 из С.....	32
2.10	Интерфейс I2C	33
2.10.1	Общие сведения	33
2.10.2	Цоколёвка и схемотехника.....	33
2.10.3	Реализация интерфейса I2C в ОС Linux	34
2.10.4	Доступ к I2C из командной строки.	34
2.10.5	Доступ к I2C из С.	35
2.11	Интерфейс CAN.....	35
2.11.1	Общие сведения	35
2.11.2	Цоколёвка и схемотехника.....	35
2.11.3	Используемые сигналы подключения CAN.....	36
2.11.4	Реализация интерфейса CAN в ОС Linux.....	36
2.11.5	Доступ к CAN из командной строки.....	36
2.11.6	Доступ к CAN из С.....	37
2.12	Интерфейс SPI	37
2.12.1	Общие сведения	37
2.12.2	Цоколевка и схемотехника	37
2.12.3	Используемые сигналы подключения SPI.....	38
2.12.4	Доступ к SPI из командной строки.	39
2.12.5	Доступ к SPI из С.	39
2.13	Интерфейс PWM	39
2.13.1	Общие сведения	39
2.13.2	Цоколевка и схемотехника.....	40
2.13.3	Используемые сигналы подключения PWM.....	40
2.13.4	Доступ к PWM из командной строки.....	41
2.13.5	Доступ к PWM из С.	42
2.14	Интерфейс MIPI-CSI-2.....	42
2.14.1	Общие сведения	42
2.14.2	Цоколёвка и схемотехника.....	43
2.14.3	Используемые сигналы подключения MIPI-CSI-2.....	45
2.14.4	Доступ к интерфейсу MIPI-CSI-2 в системе Buildroot.....	46
2.14.5	Запуск видеосенсора в системе Buildroot	47
2.15	Интерфейс UART0	50
2.15.1	Общие сведения	50

2.15.2 Цоколёвка и схемотехника.....	50
2.15.3 Используемые сигналы подключения UART0	51
3 Возможные неисправности и методы их устранения.....	52
3.1 Особенности работы интегральной микросхемы 1892ВА018	52
3.2 Служба поддержки и полезные ссылки	53
3.2.1 Канал поддержки	53
3.2.2 Полезные ссылки.....	53
4 Условия хранения	54
5 Гарантия производителя.....	55
Приложение А – Список пакетов в сборке системы Buildroot.....	56
Приложение Б – Демо приложение SmartCamApp.....	65

Перечень сокращений

ВКС	– видеоконференцсвязь
ИИ	– искусственный интеллект
ИМС	– интегральная микросхема
ККМ	– контрольно-кассовая машина
ОС	– операционная система
ПО	– программное обеспечение
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 ОС (анг. simple boot loader)
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 PicoS-T МРЦН.467444.002, PicoS МРЦН.467444.002-01, PicoS-1К МРЦН.467444.002-02.

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.467444.002ТУ 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 «PicoS» (далее – модуль) является законченным модулем электронным в форм-факторе Pico-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 тактовой частотой до 2 ГГц, дополненный DSP (NPU) сопроцессором Elcore-50.

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

Модуль представлен в трех вариантах исполнения:

1. **PicoS-T** МРЦН.467444.002 – модуль вычислительный с термоpaketом (см. Рисунок 1.11);
2. **PicoS** МРЦН.467444.002-01 – модуль вычислительный без термопакета, с возможностью установки специализированного ПО для прототипирования двухдиапазонной камеры (см. Рисунок 1.12);
3. **PicoS-1K** МРЦН.467444.002-01 – модуль вычислительный без термопакета, без интерфейсов USB TYPE A, HDMI, с возможностью установки специализированного ПО для прототипирования однодиапазонной камеры (см. Рисунок 1.13).

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



Рисунок 1.11 – Внешний вид **PicoS-T** МРЦН.467444.002



Рисунок 1.12 – Внешний вид **PicoS** МРЦН.467444.002-01



Рисунок 1.13 – Внешний вид **PicoS-1K** МРЦН.467444.002-02

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

- рабочие станции начального уровня;
- тонкие клиенты;
- банкоматы;
- торговые терминалы;
- и т.д.

Основные сферы применения модуля показаны на рисунке 1.14.



Рисунок 1.14 – 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		
		PicoS-T	PicoS	PicoS-1K
1.	Общие характеристики			
1.1	Габаритные размеры, мм	112×79×43,7	100×72,4× 29,7	100×70×11
1.2	Процессор	СКИФ (1892ВА18)		
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	Пассивное (с тер.п-ом)	Пассивное (без термопакета)	
1.7	Аппаратное отключение камеры и звука	Да		
1.8	Функционирование под операционными системами	RedOS, AltLinux, Пользовательская Linux (Buildroot)		
1.9	Часы реального времени с элементом резервного электропитания	Да		
2.	Память			
2.1	ОЗУ (тип LPDDR4), ГБ	8 (2x4)*		
2.2	ПЗУ (тип eMMC), ГБ	16*		
2.3	ПЗУ (тип qSPI), МБ	16		
3.	Сетевые интерфейсы			
3.1	Ethernet 10/100/1000M, RJ-45, шт.	1	-	
3.2	Ethernet 10/100M, 8pin, шт.	-	1	
3.3	Модуль Wi-Fi (m.2 2230 KEY-AE), шт.	1		
4.	Видео интерфейсы			
4.1	HDMI 1.4, шт.	1	-	
4.2	eDP, 30pin, шт.	1		
4.3	Интерфейс подключения камеры MIPI-CSI-2, 22pin, шт.	2		
5.	Прочие интерфейсы			
5.1	SSD накопитель (m.2 2280 KEY-M), шт.	1		
5.2	Слот microSD, шт.	1		
5.3	USB 3.0 (Type-A), шт.	2	-	
5.4	USB 2.0 (Type-C), (совмещен с одним имеющимся USB A) шт.	1		
5.5	Аудио выход стерео, шт.	1		
5.6	Аудио вход моно, шт.	1		
5.7	Дискретные входы и выходы			
	Универсальный вход/выход (TTL 3,3 В), шт.	8		
	Оптоизолированный вход (Opto DI), шт.	4		
	Оптоизолированный выход (Opto DO), шт.	4		

Продолжение таблицы 1.2

№ п/п	Характеристика	Значение		
		PicoS-T	PicoS	PicoS-1K
5.8	Внешний интерфейс I2C, шт.		1	
5.9	Внешний интерфейс CAN 2.0B, шт.		1	
5.10	Внешний интерфейс RS-485, шт.		1	
5.11	Внешний интерфейс RS-232, шт.		1	
5.12	Внешний интерфейс SPI, шт.		1	
5.13	Внешний интерфейс PWM, 2 канала, шт.		1	
Примечание – характеристики и их значения могут быть изменены без уведомления				

Габаритные размеры модуля представлены на рисунках 1.21, 1.22 и 1.23.

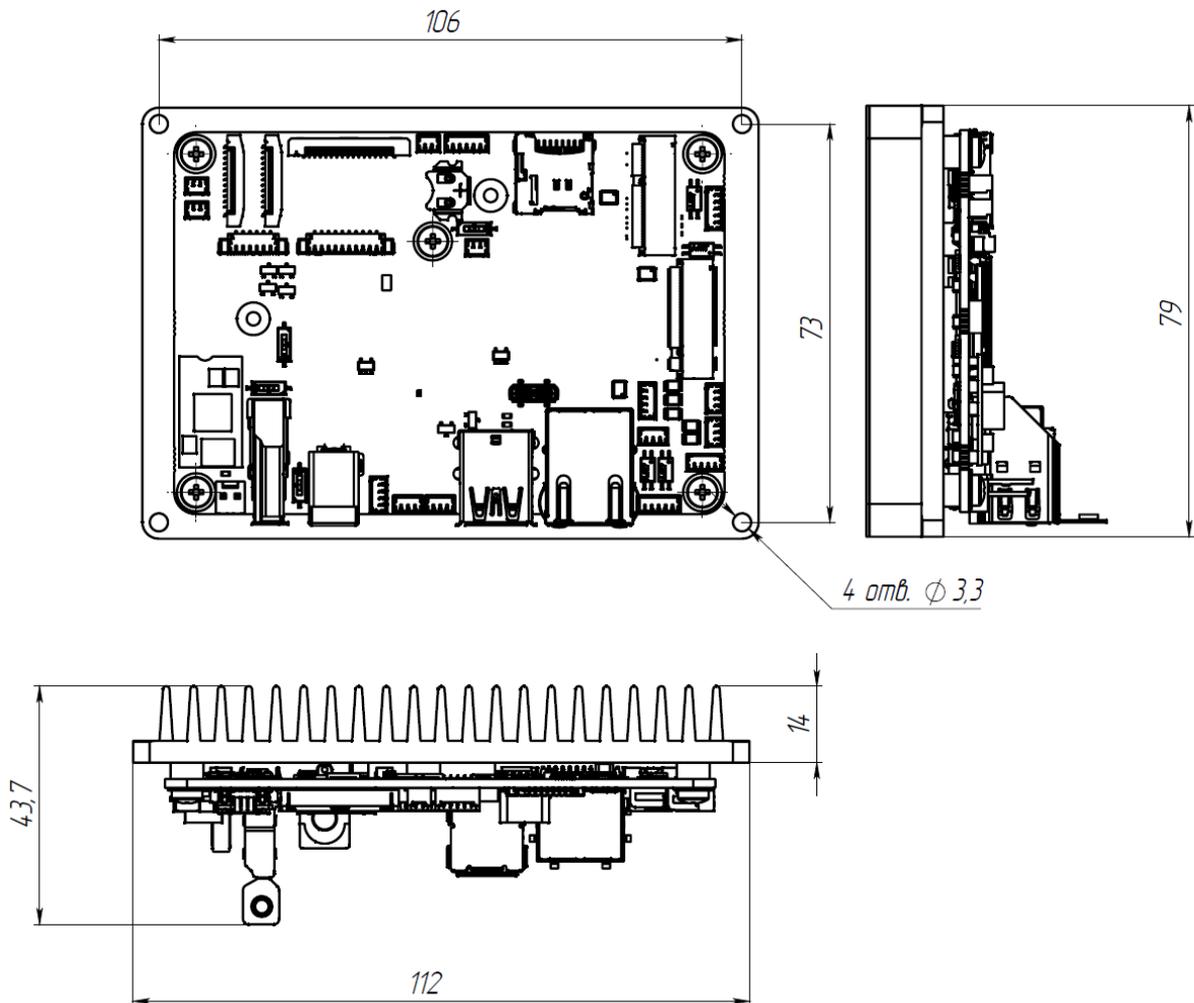


Рисунок 1.21 – Габаритные размеры PicoS-T

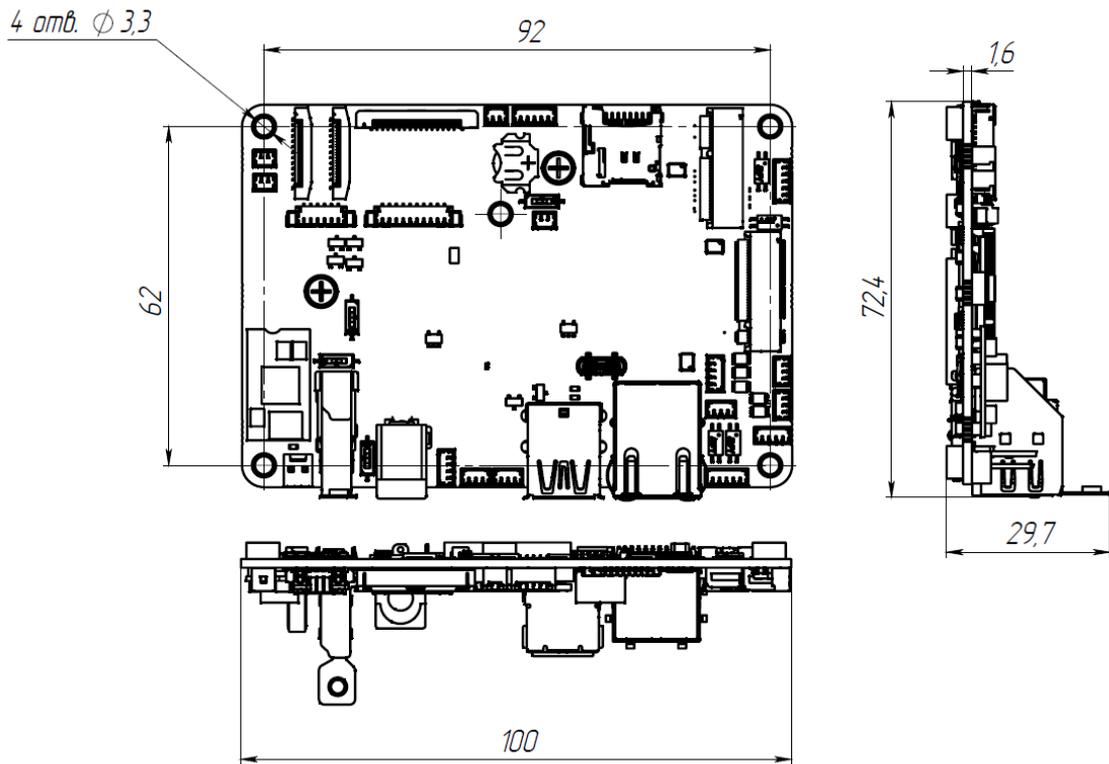


Рисунок 1.22 – Габаритные размеры PicoS

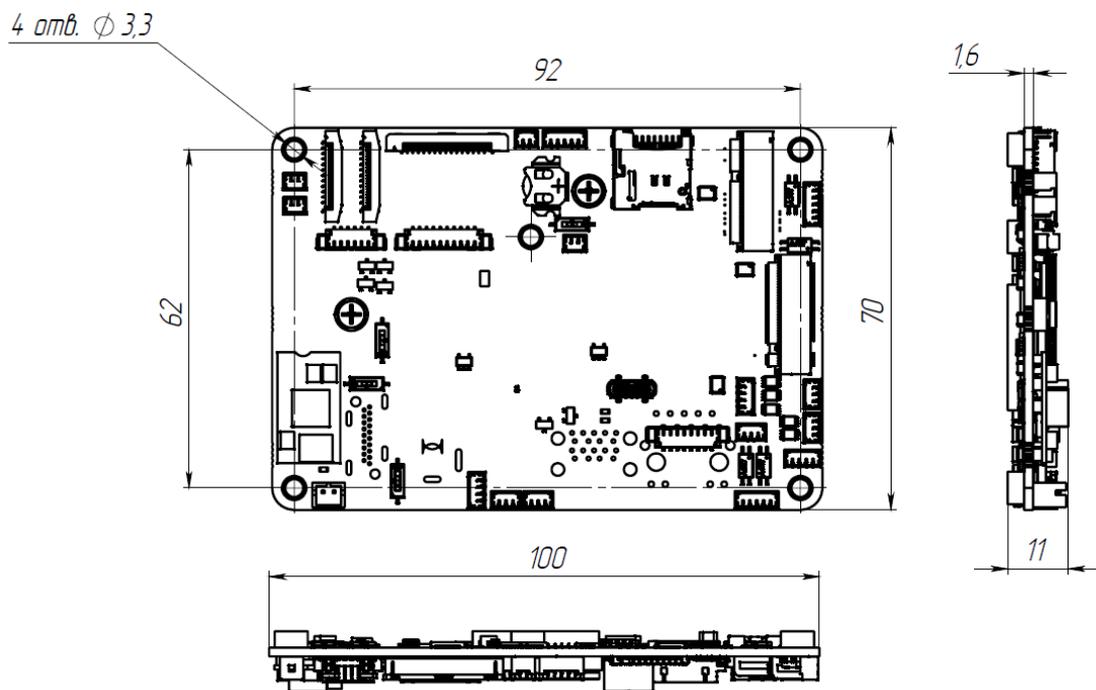


Рисунок 1.23 – Габаритные размеры PicoS-1K

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	PicoS-T или PicoS или PicoS-1K
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
<p>* – ПО представлено «как есть», исключительно для демонстрации возможностей модуля и доступно для скачивания с сайта компании ООО «Макро ЕМС»</p> <p>** – Описание пакетов в сборке Buildroot указано в приложение А</p>			

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 ПО 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 PicoS-T, PicoS интерфейсы

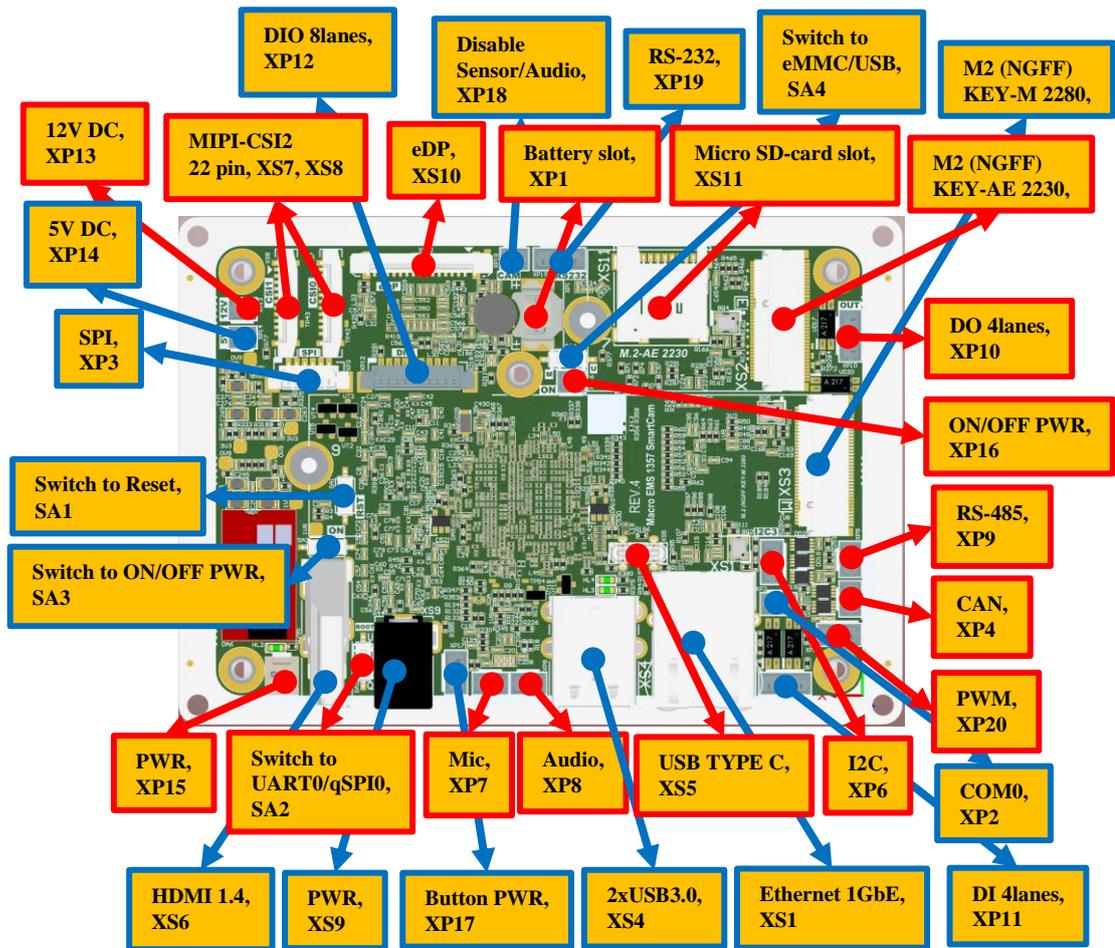


Рисунок 2.2.1 – Интерфейсы модуля PicoS-T, PicoS

- Подключите рабочую станцию в разъем «COM0» (XP2) через преобразователь UART-USB;
- При необходимости установите MicroSD карту и SSD в предназначенные для них слоты, а также подключите Ethernet кабель в разъем XP5;
- Подключите блок питания к разъему 12V DC (XP15);
- Войдите в консоль интегральной микросхемы через программу Minicom. Для этого в ОС на рабочей станции выполните следующие действия:
 - установите программу minicom;
 - войдите в консоль программы Terminal;
 - наберите в командной строке консоли Terminal команду:

minicom -D /dev/ttyUSBx -b 115200

Вместо ttyUSBx ввести инициализированный в ОС преобразователь UART-USB, например ttyUSB0.

2.4 Авторизация в ПО

ПО AltLinux

Логин	–	root
Пароль	–	elvees

ПО Buildroot

Логин	–	root
Пароль	–	отсутствует

2.5 Подключение к модулю по сетевому протоколу SSH.

2.5.1 Подключение по SSH через командную строку Windows.

2.5.2.1. 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.

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

2.5.2.3. 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.

2.5.2.4. 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 policycoreutils

sudo apt-get install net-tools

2.5.2.3. Проверяем 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 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 2 портов на модуле используется прямое подключение к ИМС 1892BA018. В интегральной микросхеме используются интерфейсы USB0 и USB1.

Внимание! Интерфейс USB1 (верхний порт относительно платы) подключен параллельно с разъемом USB Type-C (XS5).

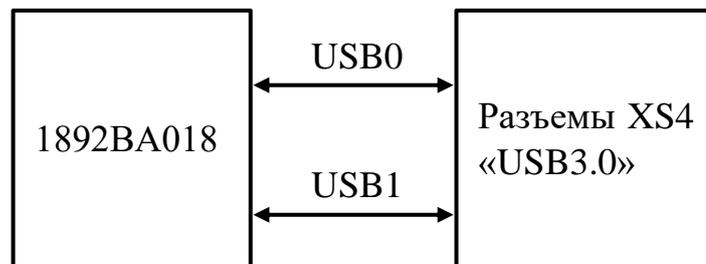


Рисунок 2.6.1 – Структурная схема интерфейсов USB

Используемые сигналы подключения

- Сигналы USB2.0;
- Сигналы USB3.0.

2.6.2 Запуск USB-камеры на модуле PicoS-T, Pico-S

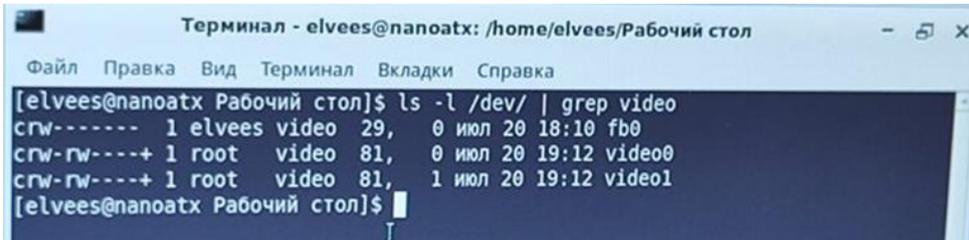
В свободный USB порт модуля подключите веб-камеру.

Убедитесь, что веб-камера определилась в ОС Alt Linux, для этого выполните следующие действия:

- Откройте «Терминал» в ОС Altlinux.
- 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 веб-камеры

Для вывода видеопотока с веб-камеры набрать команду в командной строке программы «Терминал» и нажать 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 веб-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 тремя разъемами: цифровыми входами Opto DI - «INPUT», цифровыми выходами Opto DO - «OUTPUT» и цифровыми входами/выходами GPIO - «DIO».

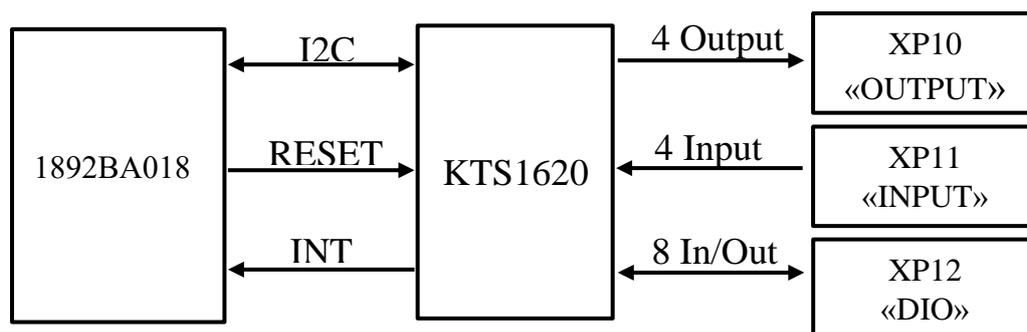


Рисунок 2.7.1 – Структурная схема интерфейсов DIO

2.7.2 Цоколёвка и схемотехника

Для подключения входных сигналов к разъему Opto DI используется разъем XP11. Тип используемого разъема – SCT1251WV-5P (Scondar). Для подключения к разъему XP11 необходимо использовать кабельный разъем SCT1251MH-5P (Scondar).

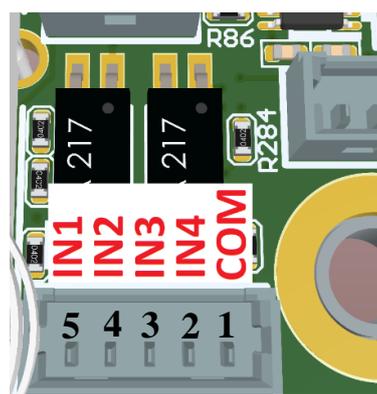


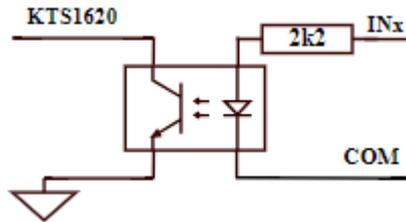
Рисунок 2.7.2.1 – Цоколёвка (разводка выводов) разъема Opto DI

Таблица 2.7.2.1 – Выводы разъема Opto DI

Номер вывода	Наименование вывода	Описание	Тип
1	COM	Общий вход	Opto
2	INPUT4	Вход №4	Opto

Номер вывода	Наименование вывода	Описание	Тип
3	INPUT3	Вход №3	Opto
4	INPUT2	Вход №2	Opto
5	INPUT1	Вход №1	Opto

Схема подключения цифрового входа Opto DI



Ограничения входных сигналов:

Напряжение.....24 В DC

Ток.....3-10 mA

Для подключения выходных сигналов к разъему Opto DO используется разъем XP10. Тип разъема – SCT1251WV-5P (Scandar). Для подключения к разъему XP10 необходимо использовать кабельный разъем SCT1251MH-5P (Scandar).

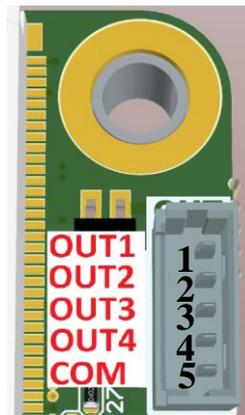


Рисунок 2.7.2.2 – Цоколёвка (разводка выводов) разъема Opto DO

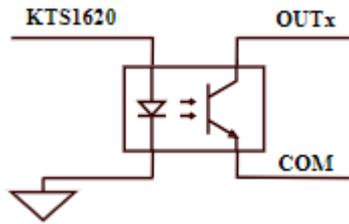
Таблица 2.7.2.2 – Выводы разъема Opto DO

Номер вывода	Наименование вывода	Описание	Тип
1	OUTPUT1	Выход №1	Opto
2	OUTPUT2	Выход №2	Opto
3	OUTPUT3	Выход №3	Opto
4	OUTPUT4	Выход №4	Opto
5	COM	Общий выход	Opto

Схема подключения цифрового выхода DO

Ограничения выходных сигналов:

Напряжение.....24 В DC



Ток.....50 mA

Для подключения цифровых входных/выходных сигналов к разъему цифровых входов/выходов GPIO используется разъем XP12. Тип разъема – SCT1251WV-10P (Scondar). Для подключения к разъему XP12 необходимо использовать кабельный разъем SCT1251MH-10P (Scondar).

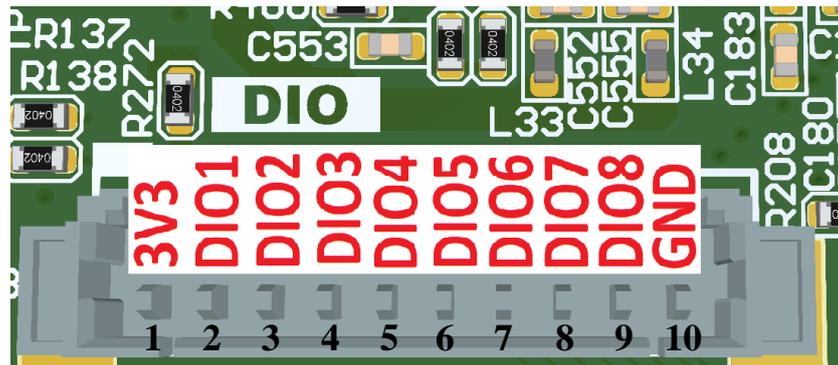
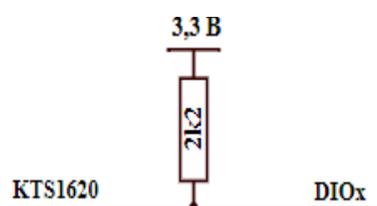


Рисунок 2.7.2.3 – Цоколёвка (разводка выводов) разъема Opto GPIO

Таблица 2.7.2.3 – Выводы разъема Opto GPIO

Номер вывода	Наименование вывода	Описание	Тип
1	+3.3В	Напряжение питания	Питание
2	DIO1	Цифровой вход/выход №1	TTL 3,3 В
3	DIO2	Цифровой вход/выход №2	TTL 3,3 В
4	DIO3	Цифровой вход/выход №3	TTL 3,3 В
5	DIO4	Цифровой вход/выход №4	TTL 3,3 В
6	DIO5	Цифровой вход/выход №5	TTL 3,3 В
7	DIO6	Цифровой вход/выход №6	TTL 3,3 В
8	DIO7	Цифровой вход/выход №7	TTL 3,3 В
9	DIO8	Цифровой вход/выход №8	TTL 3,3 В
10	GND	Земля	Земля

Схема подключения цифрового выхода DIO



Ограничения цифровых входов/выходов:

Напряжение.....3,3В DC

Ток.....5-10 мА

Таблица 2.7.2.4 – 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]	INPUT2	P1[0]	OUTPUT1	P2[0]	DIO1
P0[1]		P1[1]	OUTPUT2	P2[1]	DIO2
P0[2]	INPUT1	P1[2]	OUTPUT3	P2[2]	DIO3
P0[3]	INPUT3	P1[3]	OUTPUT4	P2[3]	DIO4
P0[4]	INPUT4	P1[4]	-	P2[4]	DIO5
P0[5]	-	P1[5]	-	P2[5]	DIO6
P0[6]	-	P1[6]	-	P2[6]	DIO7
P0[7]	-	P1[7]	-	P2[7]	DIO8

2.7.3 Используемые сигналы подключения DIO

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

Сигнал RESET (сигнала сброса-активный уровень 0). При подаче сигнала логический 0 на время более 20 мс 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 – Соответствие сигналов микросхемы KTS1620 и выводов ИМС 1892BA018

Наименование сигнала	Выход ИМС 1892BA018	Описание
RESET	GPIO0_PORTD_6	Сигнал сброса из ИМС 1892BA018
INT	GPIO1_PORTA_6	Сигнал прерывания в ИМС 1892BA018

Прерывание выводов интерфейса DIO не реализовано на аппаратном уровне.

2.7.4 Реализация DIO в ОС Linux

2.7.4.1 Активация драйвера

В ОС 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 ОС 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 – Соответствие внешних сигналов интерфейса DIO и адреса вывода микросхемы KTS1620 экспортированного в ОС Linux

Внешний сигнал, с выводов Opto DI	Адрес вывода микросхемы в ОС Linux	Внешний сигнал, на выходы Opto DO	Адрес вывода микросхемы в ОС Linux	Внешний сигнал, с/на выходы GPIO	Адрес вывода микросхемы в ОС Linux
INPUT1	426	OUTPUT1	433	DIO1	440
INPUT2	424	OUTPUT2	434	DIO2	441
INPUT3	427	OUTPUT3	435	DIO3	442
INPUT4	428	OUTPUT4	436	DIO4	443
-	-	-	-	DIO5	444
-	-	-	-	DIO6	445
-	-	-	-	DIO7	446
-	-	-	-	DIO8	447

2.7.4.3 Пример использования внешнего сигнала

В качестве примера используется внешний сигнал, поступающий с вывода интерфейса opto DI (INPUT2).

Для экспорта данного вывода в командной строке Linux наберите команду 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 воспользоваться 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 воспользоваться 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
```

В качестве примера использования интерфейса DIO есть скрипт «`gpio_test_pico.sh`». В ОС данный файл расположен: `/usr/local/bin/`

Скрипт «`gpio_test_pico.sh`» опрашивает состояние всех кнопок на тестирующей плате «Test_DIO_rev2». При нажатии на кнопку:

- На линиях DIO происходит свечение соответствующего светодиода.
- На линиях Opto DI/DO происходит триггерное переключение.

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 1892ВА018 используется интерфейс UART3 (COM3).

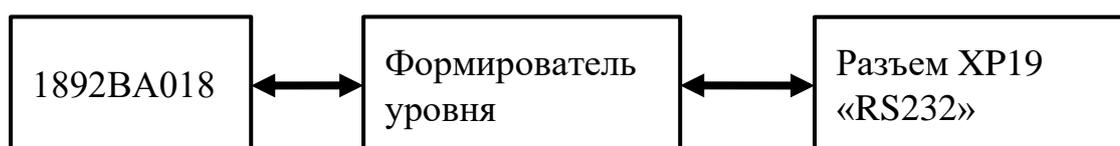


Рисунок 2.8.1 – Структурная схема интерфейса RS-232

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;
- Соответствует требованиям стандарта EIA/TIA-232-F.

2.8.2 Цоколёвка и схемотехника

Для подключения сигналов RS-232 к плате используется разъем XP19. Тип разъема XP19 – SCT1251WV-5P (Scondar). Для подключения к разъему XP19 необходимо использовать кабельный разъем SCT1251MH-5P (Scondar).

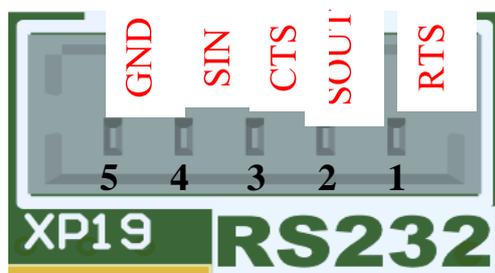


Рисунок 2.8.2 – Цоколёвка (разводка выводов) разъема RS-232

Таблица 2.8.2 – Выводы разъема интерфейса RS-232

Номер вывода	Наименование вывода	Описание
1	RTS	Выход запроса передачи данных
2	SOUT	Выход передатчика
3	CTS	Вход разрешения терминалу передавать данные
4	SIN	Вход приемника
5	GND	Земля

2.8.3 Используемые сигналы подключения RS-232

Таблица 2.8.3 – Соответствие сигналов интерфейса RS-232 и выводов ИМС 1892BA018

Номер вывода	Вывод интерфейса RS-232	Порты вывода ИМС 1892BA018	Описание
1	RTS	GPIO0_PORTA_7	Сигнал RTS из ИМС 1892BA018
2	SOUT	GPIO0_PORTB_1	Выход данных TX из ИМС 1892BA018
3	CTS	GPIO0_PORTA_2	Сигнал CTS в ИМС 1892BA018
4	SIN	GPIO0_PORTB_0	Вход данных RX в ИМС 1892BA018

2.8.4 Доступ к RS-232 из командной строки

Интерфейс RS-232 является стандартным серийным портом. В операционной системе 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 С

Для написания программы на языке С связанной с интерфейсом RS-232 достаточно стандартной библиотеки языка С (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).

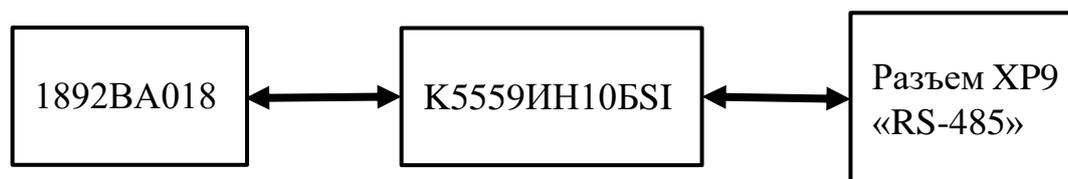


Рисунок 2.9.1 – Структурная схема интерфейса RS-485

Параметры интерфейса

Соответствует требованиям стандарта EIA/TIA-RS-485.

2.9.2 Цоколёвка и схемотехника

Для подключения сигналов RS-485 к плате используется разъем ХР9. Тип разъема ХР9 – SCT1251WV-3P (Scondar). Для подключения к разъему ХР9 необходимо использовать кабельный разъем SCT1251MH-3P (Scondar).

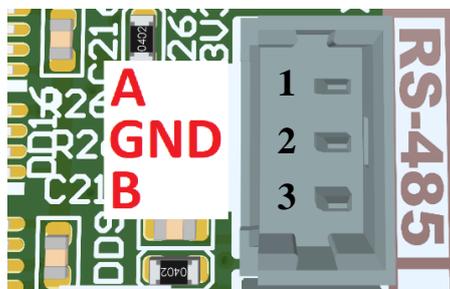


Рисунок 2.9.2 – Цоколёвка (разводка выводов) разъема RS-485

Таблица 2.9.2 – Выводы разъема интерфейса RS-485

Номер вывода	Наименование вывода	Описание
1	A	Прямой вход/выход
2	GND	Земля
3	B	Инвертирующий 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 инвертирующий 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 инвертирующего входа В с ИМС 1892ВА018. При подаче сигнала в логический «0» на время более 0,7 мкс микросхема Миландр К5559ИН10БСИ переход 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 2.9.3 – Соответствие сигналов интерфейса RS-485 и выводов 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 ОС Linux

В ОС реализован драйвер. Драйвер использует стандартный API ядра Linux для интерфейса RS485. Исходный код драйвера находится в: *drivers/tty/serial/8250/8250_dw.c*

Драйвер собран монолитно в ядре ОС и не требует дополнительной загрузки.

2.9.5 Доступ к RS-485 из командной строки

Интерфейс RS-485 является стандартным серийным портом. В операционной системе 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.

В ИМС 1892ВА018 используется интерфейс I2C-3.



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 Цоколёвка и схемотехника

Для подключения устройства по интерфейсу I2C используется разъем XP6. Тип используемого разъема - SCT1251WV-4P (Scondar). Для подключения к разъему XP6 необходимо использовать кабельный разъем SCT1251MH-4P (Scondar).

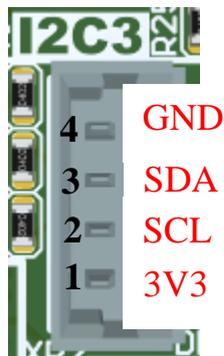


Рисунок 2.10.2 – Цоколёвка (разводка выводов) разъема I2C

Таблица 2.10.2 – Выводы разъема интерфейса I2C

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

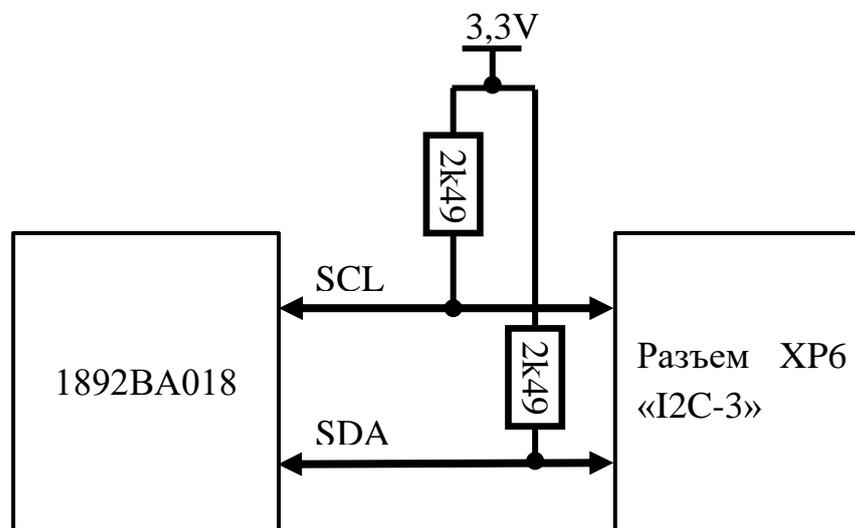


Рисунок 2.10.3 – Схема подключения интерфейса I2C

2.10.3 Реализация интерфейса I2C в ОС 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 Цоколёвка и схемотехника

Для подключения устройства по интерфейсу CAN используется разъем XP4. Тип используемого разъема - SCT1251WV-3P (Scondar). Для подключения к разъему XP4 необходимо использовать кабельный разъем SCT1251MH-3P (Scondar).

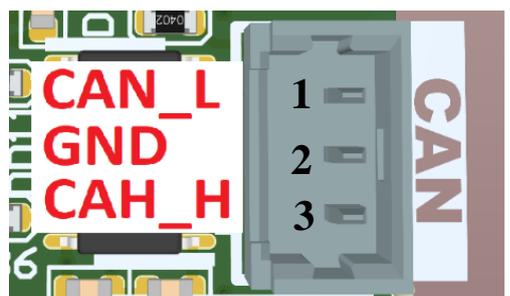


Рисунок 2.11.2 – Цоколёвка (разводка выводов) разъема CAN

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	MFBSPO_LDAT6	Вход данных в микросхему ИМС 1892BA018
CAN_TX	MFBSPO_LDAT7	Выход данных из микросхемы ИМС 1892BA018

2.11.4 Реализация интерфейса CAN в ОС Linux

Драйвер использует стандартный 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 настройки установились и шина 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 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 С.

Для написания программы на языке С связанной с интерфейсом CAN достаточно стандартной библиотеки языка С (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.

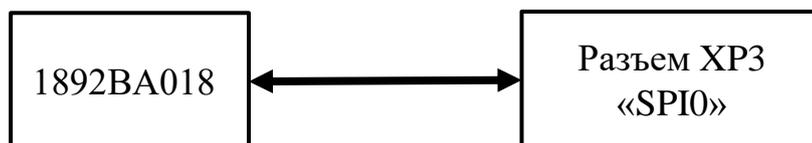


Рисунок 2.12.1 – Структурная схема интерфейса SPI

2.12.2 Цоколевка и схемотехника

Для подключения устройства по интерфейсу SPI используется разъем XP3. Тип используемого разъема - SCT1251WV-6P (Scondar). Для

подключения к разъему XP3 необходимо использовать кабельный разъем SCT1251MH-6P (Scondar).



Рисунок 2.12.2 – Структурная схема интерфейса SPI

Таблица 2.12.1 – Выводы разъема интерфейса SPI

Номер вывода	Наименование вывода	Описание
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 2.12.2 – Соответствие сигналов интерфейса SPI и ИМС 1892BA018

Номер вывода	Вывод интерфейса SPI	Порты вывода ИМС 1892BA018	Описание
2	CS	GPIO0_PORTC_5	Выход данных TX из ИМС 1892BA018
4	MOSI	GPIO0_PORTC_1	Выход данных из ИМС 1892BA018
5	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 *mtd_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 PWM

2.13.1 Общие сведения

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

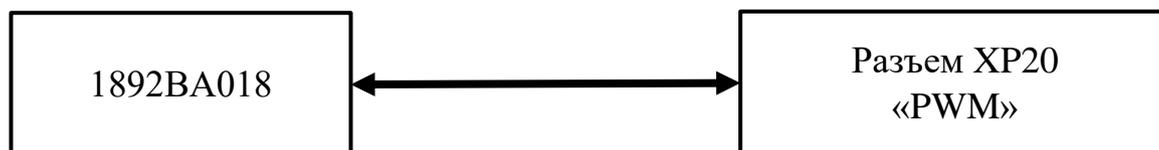


Рисунок 2.13.1 – Структурная схема интерфейса PWM

2.13.2 Цоколевка и схемотехника

Для подключения устройства по интерфейсу PWM используется разъем XP20. Тип используемого разъема - SCT1251WV-4P (Scondar). Для подключения к разъему XP20 необходимо использовать кабельный разъем SCT1251MH-4P (Scondar).

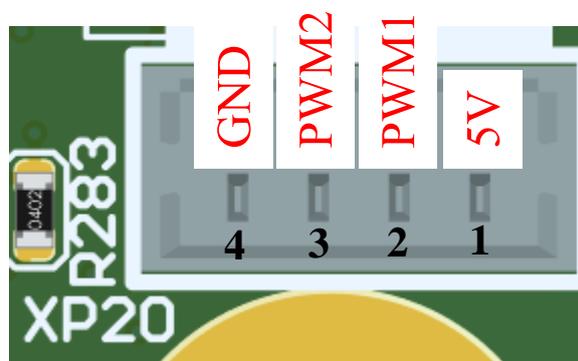


Рисунок 2.13.2 – Структурная схема интерфейса PWM

Таблица 2.13.1 – Выводы разъема интерфейса PWM

Номер вывода	Наименование вывода	Описание
1	5V	Питание
2	PWM1	Сигнальная линия аппаратной широко-импульсной модуляции из ИМС 1892BA018
3	PWM2	Сигнальная линия программной широко-импульсной модуляции из ИМС 1892BA018
4	GND	Земля

2.13.3 Используемые сигналы подключения PWM

PWM1 – аппаратный сигнал широко-импульсной модуляции. Высокий уровень сигнала по напряжению составляет - 5В, низкий – 0В. Диапазон частоты сигнала может быть задан от 0,5 кГц до 30 кГц.

PWM2 – программный сигнал широк-импульсной модуляции. Высокий уровень сигнала по напряжению составляет - 5В, низкий – 0В. Диапазон частоты сигнала может быть задан от 0,5 кГц до 30 кГц.

Таблица 2.13.2 – Соответствие сигналов интерфейса PWM и ИМС 1892ВА018

Номер вывода	Вывод интерфейса SPI	Порты вывода ИМС 1892ВА018	Описание
2	PWM1	GPIO1_PORTD_7	Аппаратный ШИМ из ИМС 1892ВА018
3	PWM2	GPIO0_PORTD_4	Программный ШИМ из ИМС 1892ВА018

2.13.4 Доступ к PWM из командной строки.

Для экспорта выводов PWM1 и PW2 в ОС в командной строке Terminal набрать команды ниже. Каждая команда должна заканчиваться нажатием клавиши «Enter».

```
export PWM1=452
```

```
export PWM2=455
```

```
echo $PWM1 >/sys/class/gpio/export
```

```
echo $PWM2 >/sys/class/gpio/export
```

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

Направление на выход:

```
echo out >/sys/class/gpio/gpio$PWM1/direction
```

```
echo out >/sys/class/gpio/gpio$PWM2/direction
```

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

```
cat /sys/class/gpio/gpio$PWM1/value
```

```
cat /sys/class/gpio/gpio$PWM2/value
```

Установить значение «1» на нем можно командой:

```
echo 1 >/sys/class/gpio/gpio$PWM1/value
```

```
echo 1 >/sys/class/gpio/gpio$PWM2/value
```

Или значение «0»:

```
echo 0 >/sys/class/gpio/gpio$PWM1/value
```

```
echo 0 >/sys/class/gpio/gpio$PWM2/value
```

2.13.5 Доступ к PWM 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 PWM 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

2.14 Интерфейс MIPI-CSI-2

2.14.1 Общие сведения

Для реализации интерфейса MIPI-CSI-2 используется прямое подключение к MIPI-CSI линиям ИМС 1892BA018. В интегральной микросхеме используются интерфейсы CSI0 и CSI1.

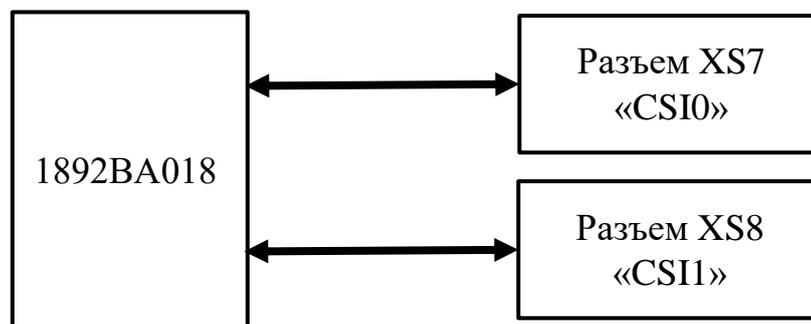


Рисунок 2.14.1 – Структурная схема интерфейса MIPI-CSI-2

2.14.2 Цоколёвка и схемотехника

Для подключения видеосенсора по интерфейсу MIPI-CSI-2 используется разъемы XS7 (CSI0) и XS8 (CSI1). Тип используемого разъема FCZ-022-VC50-99/RT (E-tec Interconnect). Для подключения к разъему XS7-XS8 необходимо использовать пленочный шлейф FFC 22 pin, тип А с шагом контактов 0,5 мм.

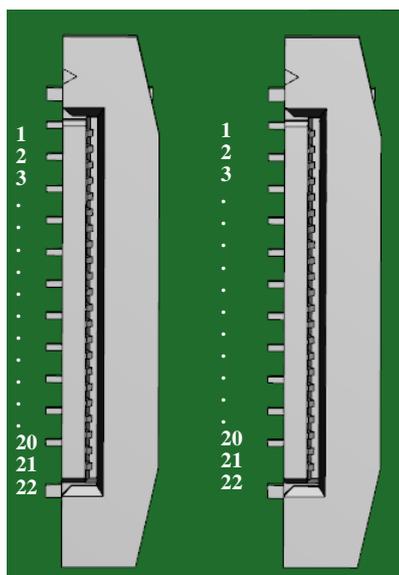


Рисунок 2.14.2.1 – Цоколёвка (разводка выводов) разъема MIPI-CSI-2

Таблица 2.14.1 – Выводы разъема 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

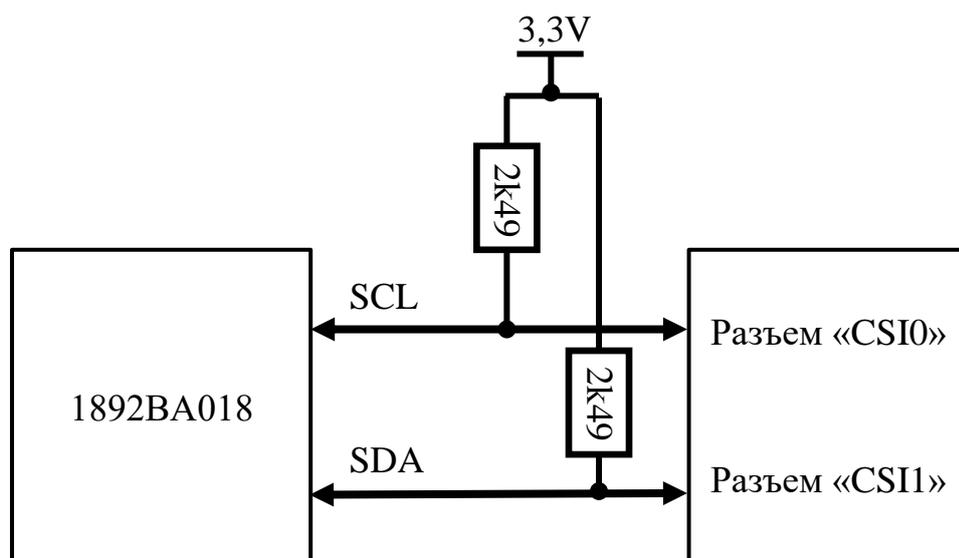


Рисунок 2.14.2.2 – Схема подключения шины управления I2C для MIPI-CSI-2

2.14.3 Используемые сигналы подключения MIPI-CSI-2

Шина управления видеосенсора подключенного к разъему CSI0. Видеосенсор управляется по шине I2C (i2c_1). Скорость шины I2C зависит от подключаемого видеосенсора. Уровень сигнала по напряжению составляет 3,3 В. На шине управления установлены подтягивающие резисторы 2,49 кОм.

Сигнал MIPI0_PWRen – сигнал включения видеосенсора подключенного к разъему CSI0. При подаче сигнала логический «0» из ИМС 1892BA018 видеосенсор включается. При подаче сигнала логическая «1» видеосенсор выключается. Уровень напряжения сигнала MIPI0_PWRen составляет 3,3В. На выводе сигнальной линии установлен подтягивающий резистор 2,49 кОм.

Шина управления видеосенсора подключенного к разъему CSI1. Видеосенсор управляется по шине I2C (i2c_2). Скорость шины I2C зависит от подключаемого видеосенсора. Уровень сигнала по напряжению составляет 3,3 В. На шине управления установлены подтягивающие резисторы 2,49 кОм.

Сигнал MIPI1_PWRen - сигнал включения видеосенсора подключенного к разъему CSI1. При подаче сигнала логический «0» из ИМС 1892BA018 видеосенсор включается. При подаче сигнала логическая «1» видеосенсор

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

Таблица 2.14.2 – Соответствие сигналов интерфейса MIPI-CSI-2 и ИМС 1892ВА018

Наименование сигнала	Порты вывода процессора ИМС 1892ВА018	Описание
MIPI0_PWRen	GPIO1_PORTD_0	Сигнал включения видеосенсора из ИМС 1892ВА018, подключенного к разъему «CSI0» (XS7).
MIPI1_PWRen	GPIO0_PORTC_6	Сигнал включения видеосенсора из ИМС 1892ВА018, подключенного к разъему «CSI1» (XS8).

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

В ОС Buildroot реализована передача видео с сенсора 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 сенсора изображения с использованием ISP поддерживается плагином gst-felix для пакета GStreamer. Данным плагином поддерживается элемент felixsrc – захват видео с сенсора. Элементом felixsrc не поддерживается работа двух видеосенсоров одновременно.

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

Таблица 2.14.3 – Расположение установочных файлов поддерживаемых сенсоров в системе Buildroot, разъем подключения на модуле и рекомендуемые режимы работы сенсора

Модель сенсора	Расположение установочного файла	Разъем модуля	Режимы работы сенсора*
DS-CIMX327-22	/etc/felix/imx327/imx327.cfg	CSI0 (XS7)	0, 1
DS-CIMX335-22	/etc/felix/imx335/imx335.cfg /etc/felix/imx335/imx335-noir.cfg	CSI0 (XS7)	5, 6, 7, 8
DS-CIMX415-22	/etc/felix/imx415/imx415.cfg	CSI0 (XS7)	0, 4, 5, 7, 8, 9, 11, 12
DS-CIMX662-22	/etc/felix/imx662/imx662.cfg /etc/felix/imx662/imx662-noir.cfg	CSI0 (XS7)	0, 1, 2

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

2.14.5 Запуск видеосенсора в системе Buildroot

Мы рекомендуем использовать сенсоры серии: DS-CIMX335-22, DS-CIMX415-22, DS-CIMX327-22, DS-CIMX662-22. Рекомендованные сенсоры могут быть подключены только к разъему «CSI0» (XS7 на плате модуля).

Для начала работы с видеосенсором выполните следующие действия:

1. Подключите видеосенсор к модулю используя разъем «CSI0» (XS7) через шлейф 22pin-to-22pin тип А с шагом 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 видеосенсор инициализирован в системе. Запустите утилиту `felix-sensor-test` набрав команду и нажав клавишу «Enter»:

felix-sensor-test

Запущенная утилита проверит подключение всех поддерживаемых платформой сенсоров и выведет их статус. Если сенсор 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 пред словом mode является порядковым номером режима работы сенсора.

Запустите захват видео с сенсора.

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-max-time=10000
exposure-auto-min-time=16 exposure-auto-priority=1 awb-enable=true awb-
algorithm=pid awb-mode=high-lum ! 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
```

Пример запуска видеотрансляции с сенсора 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=10000          exposure-auto-min-time=16  awb-enable=true    awb-algorithm=pid
awb-mode=high-lum ! video/x-raw,format=BGR,width=1920,height=1080 ! queue
! fpsdisplaysink video-sink="kmssink driver-name=mali-dp max-latency=-1 force-
modesetting=true" -v 2>&1
```

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

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.

2.15 Интерфейс UART0

2.15.1 Общие сведения

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.

Для подключения используется интерфейс процессора UART0(COM0). На интерфейсе UART0(COM0) реализована консоль интегральной микросхемы.



Рисунок 2.15.1 – Структурная схема интерфейса MIPI-CSI-2

2.15.2 Цоколёвка и схемотехника

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

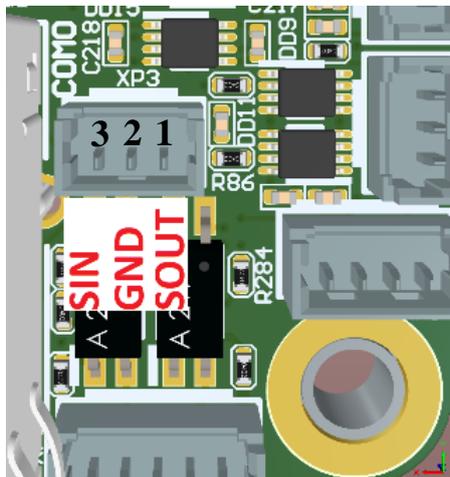


Рисунок 2.15.2 – Цоколёвка (разводка выводов) разъема UART0

Таблица 2.15.1 – Выводы разъема интерфейса UART0

Номер вывода	Наименование вывода
1	SOUT
2	GND
3	SIN

2.15.3 Используемые сигналы подключения UART0

Таблица 2.15.2 – Соответствие сигналов интерфейса MIPI-CSI-2 и 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 1892ВА018

Работа интерфейсов модуля базируется на работе микросхемы 1892ВА018 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
HDMI 1.4 Срыв синхронизации, в результате чего возникают искажение или мерцание экрана	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). <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». Решение 2: Перезапустите модуль

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 системы Buildroot

Общее описание пакетов в сборке Buildroot

Установленные библиотеки располагаются в пути */usr/lib/* в системе модуля. Сборка системы 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>
<code>enable unit test libraries</code>	<code>audioresample</code>
<code>enable command-line parser</code>	<code>rawparse</code>
<code>enable tracing subsystem</code>	<code>subparse</code>
<code>enable gst-debug trace support</code>	<code>tcp</code>
<code>enable plugin registry</code>	<code>typefind</code>
<code>install tools</code>	<code>videotestsrc</code>
<code>gst1-plugins-base→</code>	<code>videorate</code>
<code>app</code>	<code>videoscale</code>
<code>audioconvert</code>	<code>volume</code>

audomixer	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→
---	---

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

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→

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

<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 видеосенсора с нейросетевой обработкой по распознаванию лица и оружия.

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

Модель сенсора	Режимы работы сенсора*
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 сенсора взят из утилиты `felix-sensor-test`.

Запуск приложения SmartCamApp на модуле.

1. Подключите видеосенсор к модулю используя разъем «CSI0» (XS7) через шлейф 22pin-to-22pin тип А с шагом 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_сенсора>
```

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

```
app_rtsp -s 662
```

В результате запуска приложения начнётся видеотрансляция сенсора с обработкой нейросети. В правом нижнем углу видеотрансляции будет

отображаться FPS и разрешение. Чтобы прервать работу приложения нажмите комбинацию клавиш «Ctrl» + «C».

