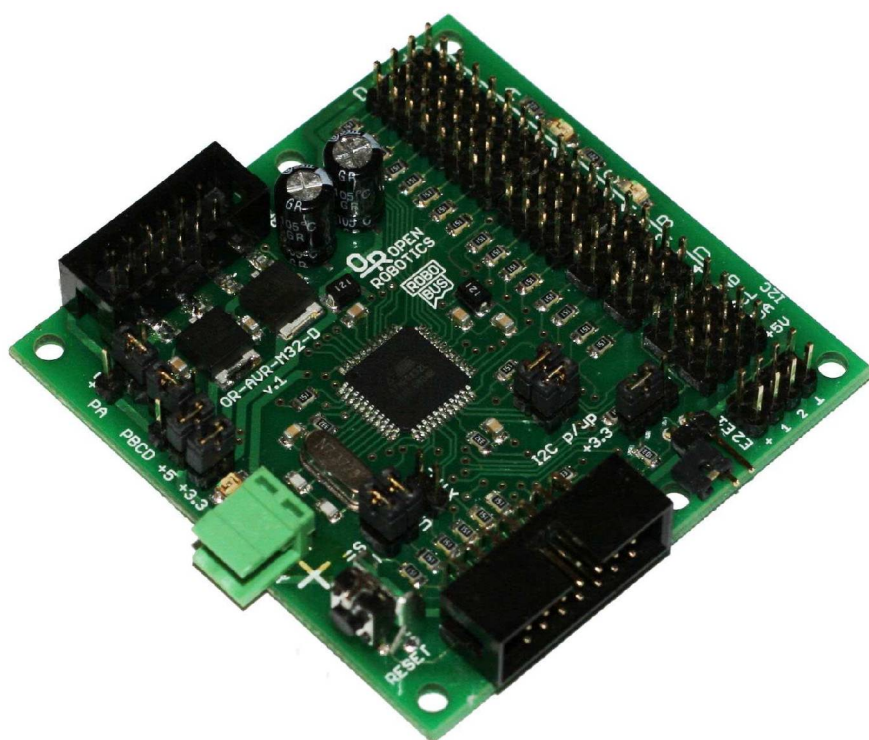


Руководство пользователя  
**OR-AVR-M32-D**

Open Robotics team

04.02.2010

Последняя версия документа доступна по адресу:  
<http://roboforum.ru/wiki/OR-AVR-M32-D>



## Содержание

1. Описание и основные характеристики.....	3
2. Расположение разъемов на плате контроллера.....	4
3. Питание контроллера.....	5
Питание с использованием встроенных стабилизаторов напряжения.....	5
Питание с использованием внешних стабилизаторов напряжения.....	5
4. Разъем RoboBus.....	6
5. Порты I2C.....	7
6. Порты GPIO.....	8
Управление питанием портов.....	9
АЦП.....	9
7. UART интерфейс.....	10
8. Подключение драйвера двигателей.....	11
Использование выводов управления двигателями в качестве GPIO линий.....	12
9. Подключение энкодеров.....	13
10. Габаритные размеры и крепежные отверстия.....	14
11. Загрузка прошивки.....	15
12. Использование прошивки ORFA.....	16
Работа с ORFA при использовании ПК или ноутбука.....	16
Формат обмена данными шлюза UART<=>I2C.....	17
Команды настройки шлюза UART<=>I2C.....	18
Работа с I2C-устройствами через шлюз (порты RoboI2C и RoboBus).....	18
Драйвер низкоскоростного цифрового ввода/вывода (порты GPIO).....	19
Драйвер АЦП (порты GPIO).....	20
Драйвер управления сервоприводами (порты GPIO).....	21
Драйвер управления коллекторными двигателями (порт RoboMD2).....	22
Драйвер SPI-интерфейса (порт RoboBus).....	23
Драйвер интроспекции.....	24

## 1. Описание и основные характеристики

Универсальный контроллер OR-AVR-M32-D (далее - контроллер) предназначен для управления устройствами мобильного робота. Контроллер имеет в своем составе все необходимое (за исключением драйвера двигателей) для управления распространенными электронными устройствами, применяемыми в робототехнике, такими как сервоприводы, ИК-дальномеры, контактные и бесконтактные бамперы, датчики линии, различные устройства с интерфейсами SPI и I<sup>2</sup>C.

В составе контроллера имеются стабилизаторы напряжения на 5 и 3.3 В, которые могут быть использованы для питания внешних устройств.

Основой контроллера является микроконтроллер (МК) ATmega32L, выпускаемый компанией Atmel <http://www.atmel.com/>.

**Таблица 1. Основные характеристики контроллера OR-AVR-M32-D**

Микроконтроллер:	ATmega32L
Тактовая частота:	7,3728 МГц
Объем памяти программ (Flash):	32 КиБ
Объем ОЗУ (RAM):	2 КиБ
Объем EEPROM:	1 КиБ
Габаритные размеры:	66(д)х66(ш)х16(в) мм
Логические уровни:	3,3 В
Диапазон напряжений питания:	6-16 В
Максимальный ток, потребляемый контроллером:	20 мА
Максимальный ток внешней нагрузки по линии 5 В:	до 0,8 А (подробнее см. в разделе "Питание контроллера")
Максимальный ток внешней нагрузки по линии 3,3 В:	до 0,8 А (подробнее см. в разделе "Питание контроллера")
Количество выводов общего назначения:	16
в том числе каналов АЦП:	8
Цифровые интерфейсы:	UART, SPI, I <sup>2</sup> C
Стандарты разъемов:	RoboBus <a href="http://www.roboforum.ru/wiki/RoboBUS">http://www.roboforum.ru/wiki/RoboBUS</a> RoboGPIO <a href="http://www.roboforum.ru/wiki/RoboGPIO">http://www.roboforum.ru/wiki/RoboGPIO</a> RoboI2C <a href="http://www.roboforum.ru/wiki/RoboI2C">http://www.roboforum.ru/wiki/RoboI2C</a> RoboMD2 <a href="http://www.roboforum.ru/wiki/RoboMD2">http://www.roboforum.ru/wiki/RoboMD2</a>

## 2. Расположение разъемов на плате контроллера

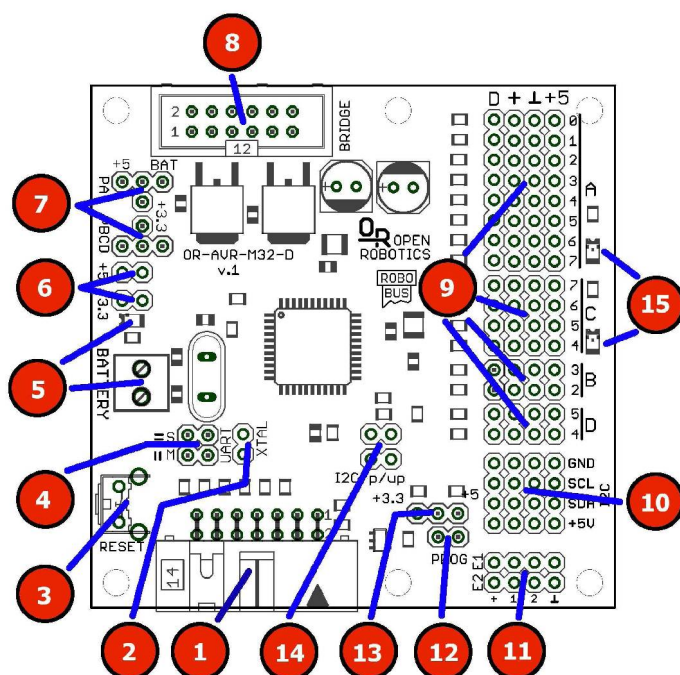


Рис. 1. Схема расположения разъемов и элементов управления на плате контроллера OR-AVR-M32-D

1. Разъем RoboBus
2. Джемпер включения внешнего тактового сигнала
3. Кнопка "Сброс"
4. Джемперы выбора режима работы UART
5. Разъем питания и индикатор питания
6. Джемперы включения встроенных стабилизаторов напряжения
7. Джемперы выбора напряжения питания портов GPIO
8. Разъем RoboMD2 для подключения драйвера коллекторных двигателей
9. Порты GPIO
10. Разъемы RoboI2C
11. Разъемы для подключения энкодеров
12. Джемпер отключения режима программирования
13. Джемпер выбора напряжения питания энкодеров
14. Джемперы подтяжки шины I<sup>2</sup>C
15. Светодиоды

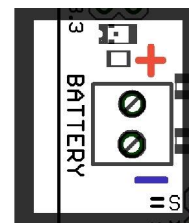
### 3. Питание контроллера

Питание контроллера может осуществляться от источника постоянного тока напряжением 6-16 вольт с использованием встроенных стабилизаторов напряжения, или от стабилизированного источника постоянного тока напряжением 3,3 вольт.



*Всегда проверяйте полярность при подключении источника питания!*

*Неверная полярность может привести к выходу из строя контроллера и подключенных к нему устройств. В первой партии контроллеров, неправильно нанесена маркировка разъема питания. Контакт, к которому подключается “минус” источника питания, ошибочно обозначен как “+”. Правильная полярность указана на рисунке справа.*



#### Питание с использованием встроенных стабилизаторов напряжения

В этом режиме, для питания используются встроенные стабилизаторы напряжения 5 В и 3,3 В. Эти напряжения выдаются на разъем RoboBus (см. раздел “Разъем RoboBus” далее) и разъемы подключения внешних устройств, а напряжение 3,3 В, кроме того, используется для питания микроконтроллера. Для включения этого режима, необходимо установить перемычки +5V и +3V3, и подать внешнее питание от 6 до 16 вольт через разъем питания.



*При использовании встроенных стабилизаторов, допустимая нагрузка по линиям +5V и +3,3 В зависит от напряжения внешнего источника питания. Зависимость приведена в таблице:*

**Таблица 2. Зависимость допустимого тока от напряжения внешнего питания**

Напряжение внешнего источника питания, В	Допустимый ток по линии «3,3 В», А	Допустимый ток по линии «5 В», А
6	0,8	0,8
9	0,35	0,5
12	0,25	0,3
16	0,16	0,18

#### Питание с использованием внешних стабилизаторов напряжения

Этот режим питания целесообразно использовать, когда мощности встроенных стабилизаторов недостаточно для питания внешних устройств. В этом случае, для питания используются внешние стабилизаторы напряжения 5 В и 3,3 В.



*Если питание осуществляется от внешних стабилизаторов, убедитесь, что перемычки +5V и +3V3 сняты, иначе возможен выход из строя контроллера и внешнего стабилизатора.*

Для включения этого режима, необходимо снять перемычки +5V и +3V3, и подать стабилизированные напряжения +5 вольт и +3,3 вольта через соответствующие контакты разъёма RoboBus (см. раздел “Разъём RoboBus” далее). Напряжение +5 вольт необходимо, только если оно требуется для питания внешних устройств, для самого контроллера нужно только +3,3 вольта.

## 4. Разъем RoboBus

Контроллер оснащён разъемом шины RoboBus. RoboBus - открытый стандарт системной шины, предназначенной для взаимодействия электронных модулей в робототехнике. Подробная информация о стандарте RoboBus - на странице <http://www.roboforum.ru/wiki/RoboBus>

На разъем RoboBus выведены интерфейсы I2C, UART, SPI, 3 вывода общего назначения и питающие напряжения +5 В и +3,3 В. Разъем может использоваться не только для взаимодействия модулей, но и для программирования контроллера (см. одноименный раздел).

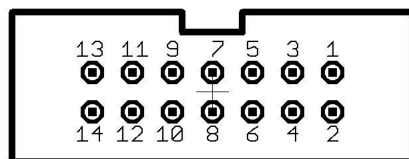


Рис. 2: Схема расположения контактов разъема RoboBus



Нумерация показана для разъема “папа”, установленного на печатную плату, вид с торца платы.



Контакты разъема RoboBus подключены к выводам микроконтроллера через ограничительные резисторы сопротивлением 150 Ом.

Таблица 3. Назначение контактов разъема RoboBus

№	Название	Интерфейс	Вывод МК	Примечание
1	GND			Земля
2	3,3V			Питание +3,3 В
3	RESET		RESET	Ввод сброса
4	MOSI	SPI	MOSI (PORTB.5)	Используется для программирования МК
5	MISO	SPI	MISO (PORTB.6)	Используется для программирования МК
6	SCK	SPI	SCK (PORTB.7)	Используется для программирования МК
7	GP2/XTAL1		PORTD.6	Вход\выход общего назначения или вход тактового сигнала
8	GP1		PORTD.7	Вход\выход общего назначения
9	GP0		PORTB.4	Вход\выход общего назначения
10	SDA	I <sup>2</sup> C	SDA (PORTC.1)	
11	SCL	I <sup>2</sup> C	SDA (PORTC.0)	
12	TXD	UART		UART передача данных (от контроллера)
13	RXD	UART		UART прием данных (в контроллер)
14	5V			

## 5. Порты I<sup>2</sup>C

Помимо разъёма RoboBus, устройства с интерфейсом I<sup>2</sup>C можно подключать к специальным разъёмам RoboI<sup>2</sup>C. Это позволяет использовать кабель меньшей ширины, что удобно для подключения подвижных устройств, например вращающийся ультразвуковой сонар.

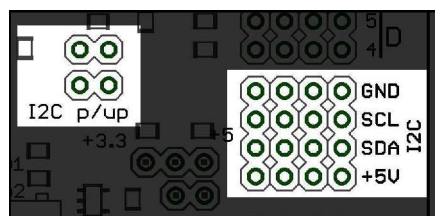


Рис. 3: Разъёмы RoboI<sup>2</sup>C

В соответствии со стандартом I<sup>2</sup>C, сигнальные линии должны быть подтянуты к напряжению питания через резисторы. На плате контроллера имеются 2 джампера, обозначенные I<sup>2</sup>C p/up. Для использования I<sup>2</sup>C, эти джамперы следует установить. Если используется несколько контроллеров, соединённых шиной RoboBus, джамперы следует установить только на одном контроллере.

На контроллере имеются 4 разъёма RoboI<sup>2</sup>C. Назначение контактов разъёма приведено в таблице. Нумерация контактов - сверху вниз.

Таблица 4. Назначение контактов разъёма RoboI<sup>2</sup>C

№	Название
1	GND
2	SCL
3	SDA
4	+5V

## 6. Порты GPIO

Контроллер имеет 16 выводов общего назначения (GPIO), сгруппированных в 4 порта, в соответствии с названиями портов микроконтроллера, к которым они подключены: A, B, C, D. Обозначения портов и номера контактов нанесены на печатную плату. Контакты разъемов портов GPIO расположены в соответствии со стандартом RoboGPIO <http://roboforum.ru/wiki/RoboGPIO>.

Назначение контактов показано на рисунке с помощью цветовой маркировки, контакты нумеруются слева направо от 1 до 4. Каждому выводу порта соответствует 4 контакта разъема: сигнал, выбираемое напряжение питания (+5В, +3,3В, или не стабилизированное питание внешнего источника), земля, и фиксированное напряжение +5В. При этом можно выбрать один уровень напряжения для 2 линии порта A (выделено синим) и другой уровень напряжения для 2 линии портов B, C, D (выделено зеленым).

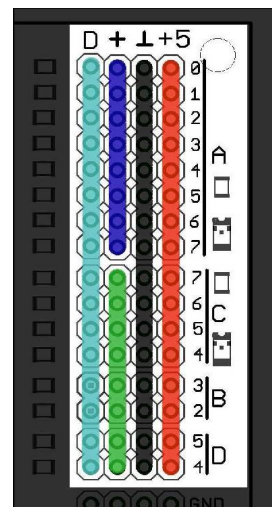


Рис. 4: Порты GPIO

Контакты на разъемах расположены таким образом, что позволяют напрямую подключать стандартные модельные рулевые машинки (серво, servo, RC servo). При этом контакт разъема серво, по которому подается управляющий сигнал, должен быть подключен к контакту 1 разъема RoboGPIO.



Контакты разъема RoboGPIO подключены к выводам микроконтроллера через ограничительные резисторы сопротивлением 150 Ом.



К выводам PORTC.5 и PORTB.3 подключены светодиоды индикации. При подключении внешних устройств к этим выводам светодиоды могут загораться.



Выводы PORTA.0, PORTA.1, PORTC.6, PORTC.7, PORTD.4 и PORTD.5 используются также для управления драйвером двигателя. При подключении драйвера двигателя к разъему RoboMD2, эти выводы нельзя использовать для управления внешними устройствами. (PORTA.0, PORTA.1, PORTC.6, PORTC.7 можно использовать, если в подключенном драйвере двигателей не используются функции датчика тока и сигнал перегрузки - см. раздел «Подключение драйвера двигателей»).



## Управление питанием портов

На контакт 4 разъёмов RoboGPIO подаётся напряжение +5В, либо от встроенного стабилизатора, либо от внешнего.

На контакт 2 разъёмов RoboGPIO можно дополнительно подать напряжение питания: 5В, 3,3В или напряжение внешнего источника питания. Выбор напряжения производится не отдельно для каждого вывода порта, а для групп выводов: отдельно для порта A и отдельно - портов B, C, D. Для выбора напряжения питания предназначены две группы контактов, обозначенные PA и PBCD. Первый выбирает напряжение для порта A, второй - для портов B, C и D.

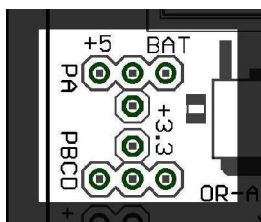


Рис. 5: Джамперы выбора питания портов

Контакты выбора напряжения расположены в виде буквы Т. Выбор осуществляется установкой джампера между центральным и одним из крайних контактов. Положение Vbat обычно используют при подключении серво. В этом положении, на контакты 2 соответствующей группы портов подается не стабилизированное напряжение от внешнего источника питания.

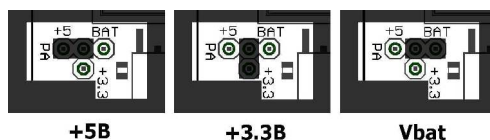


Рис. 6: Варианты питания порта A

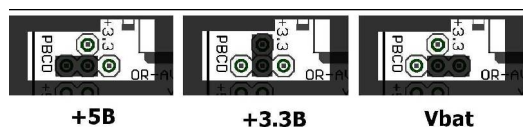


Рис. 7: Варианты питания портов B, C, D

## АЦП

Вывода порта A, помимо цифрового ввода-вывода, могут использоваться как входы аналогово-цифрового преобразователя (АЦП). Это позволяет подключать датчики, выдающие информацию в аналоговом виде, например переменные резисторы, фоторезисторы, датчики Sharp, акселерометры, и т. д.

## 7. UART интерфейс

UART (Universal Asynchronous Receiver-Transmitter, универсальный асинхронный приёмопередатчик) - один из самых распространённых последовательных интерфейсов связи. Контроллер OR-AVR-M32-D позволяет использовать UART на скоростях обмена до 115200 бит/с.

Для связи через UART используются контакты 12(TXD) и 13(RXD) шины RoboBus. В контроллере предусмотрены 2 варианта подключения UART. В первом варианте, контроллер считается ведущим (Master), передавая данные по линии TXD и принимая - по линии RXD. Второй вариант - подчинённый (Slave), контроллер принимает данные по линии TXD и передаёт - по линии RXD. Для выбора режима предназначена группа джамперов, обозначенная UART.

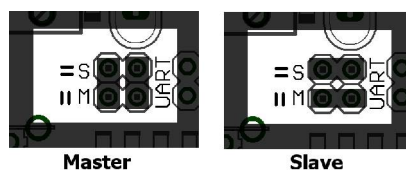


Рис. 8: Конфигурация UART

## 8. Подключение драйвера двигателей

Контроллер может управлять одним или двумя электродвигателями при подключении внешнего драйвера. Для подключения драйверов коллекторных двигателей на контроллере имеется разъём стандарта RoboMD2 (<http://roboforum.ru/wiki/RoboMD2>).

К контроллеру можно подключить любой драйвер одного или двух двигателей, который управляется двумя сигналами на каждый канал - выбор направления и регулирование скорости методом ШИМ. Драйвер должен работать с логическими уровнями 3,3 В.

Кроме сигналов управления двигателем, через разъём RoboMD2 драйвер двигателей может передавать контроллеру информацию о своём состоянии, например показания датчиков тока или сигнал перегрузки. Для этих целей на каждый канал выделено по 2 контакта, один из которых подключён к входу АЦП.

На рисунке приведена нумерация контактов разъёма RoboMD2 (вид с верхней стороны печатной платы).

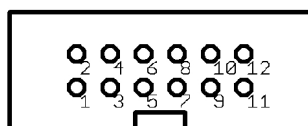


Рис. 9: Нумерация контактов разъёма RoboMD2

Таблица 5: Назначение контактов разъёма RoboMD2

№	Название	Вывод МК	Примечание
1	PWM2	PORTD.4	Цифровой сигнал управления скоростью вращения двигателя №2 (ШИМ)
2	PWM1	PORTD.5	Цифровой сигнал управления скоростью вращения двигателя №1 (ШИМ)
3	OVLД1	PORTC.7	Цифровой сигнал перегрузки двигателя №2
4	OVLД2	PORTC.6	Цифровой сигнал перегрузки двигателя №1
5	DIR2	PORTB.1	Цифровой сигнал управления направлением вращения двигателя №2
6	DIR1	PORTB.0	Цифровой сигнал управления направлением вращения двигателя №1
7	SENSE2	PORTA.1	Аналоговый сигнал датчика тока двигателя №2
8	SENSE1	PORTA.0	Аналоговый сигнал датчика тока двигателя №1
9,10	+VBAT		“+” источника питания
11,12	GND		Земля

### Использование выводов управления двигателями в качестве GPIO линий

Выводы микроконтроллера PORTD.4, PORTD.5, PORTC.6, PORTC.7, PORTA.0 и PORTA.1 подключены как к контактам разъёма RoboMD2, так и к выводам GPIO. При подключении драйвера двигателей к разъёму RoboMD2, эти выводы нельзя использовать в качестве линий общего назначения. Однако в контроллере предусмотрена возможность отключения контактов PORTC.6, PORTC.7, PORTA.0 и PORTA.1 от разъёма RoboMD2, если применённый драйвер двигателей не использует сигналы датчиков тока и/или перегрузки, или эти сигналы не нужны. Для отключения этих контактов с обратной стороны платы под разъёмом RoboMD2 расположены джамперы, выполненные в виде контактных площадок на плате. Для отключения выводов необходимо удалить припой, соединяющий контактные площадки.

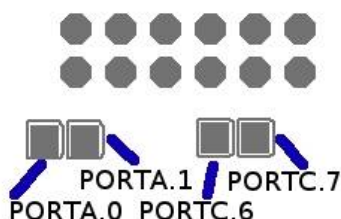


Рис. 10: Джамперы конфигурации RoboMD2

## 9. Подключение энкодеров

Для удобства подключения квадратурных энкодеров на плате контроллера предусмотрены специальные разъёмы. На разъёмы выводится напряжение для питания энкодеров, если оно требуется. Напряжение питания можно выбрать (5 вольт или 3,3 вольт) с помощью джампера.

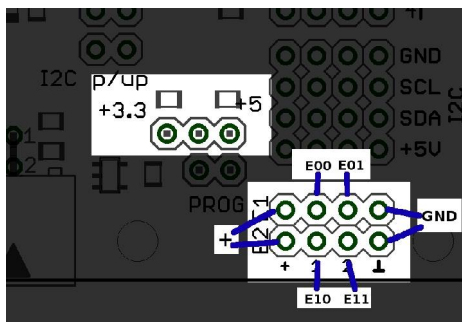


Рис. 11: Разъёмы энкодеров

Таблица 6: Назначение контактов разъёмов энкодеров

Контакт разъема	Вывод МК	Примечание
+		Напряжение питания
GND		Земля
E00	PORTD.2	Энкодер двигателя №1, канал А
E01	PORTC.2	Энкодер двигателя №1, канал В
E10	PORTD.3	Энкодер двигателя №2, канал А
E11	PORTC.3	Энкодер двигателя №2, канал В

Если энкодеры не используются, разъёмы можно использовать для подключения других устройств к соответствующим выводам микроконтроллера.

## 10. Габаритные размеры и крепежные отверстия

Для крепления на работе в печатной плате контроллера имеются 6 отверстий под M3 диаметром 3,2 мм. На рисунке размеры указаны в миллиметрах [дюймах].

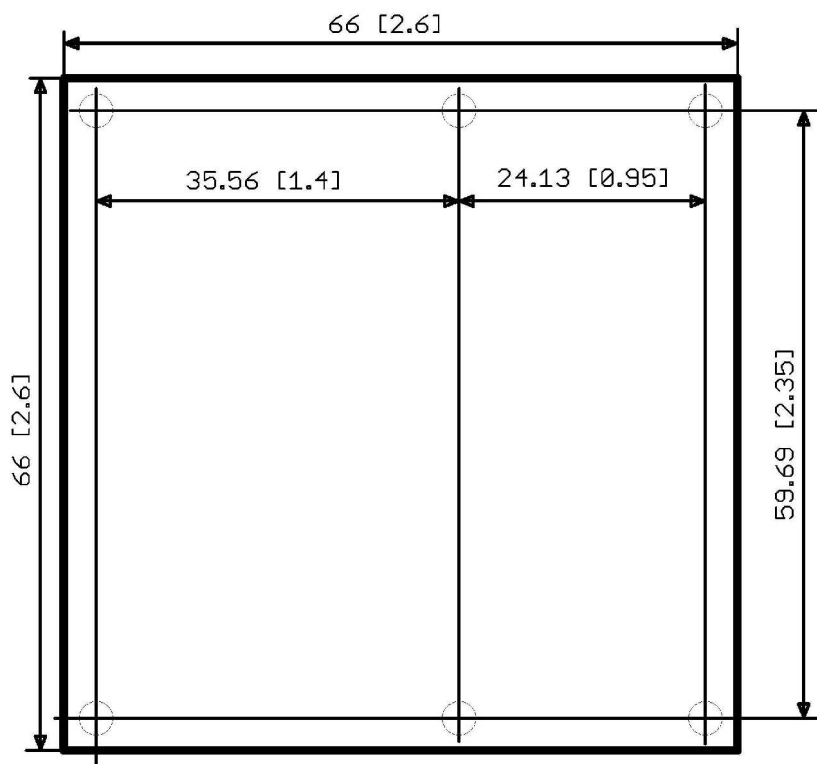


Рис. 12: Габаритные размеры и крепёжные отверстия

## 11. Загрузка прошивки

Для загрузки прошивки обычным программатором необходимо снять джампер отключения от программирования PROG, подключить программатор к контроллеру через разъем RoboBus, подать питание на контроллер (если программатор не обеспечивает его питание) и загрузить прошивку. Подробный порядок действий по прошивке контроллера смотрите в инструкции к программатору.

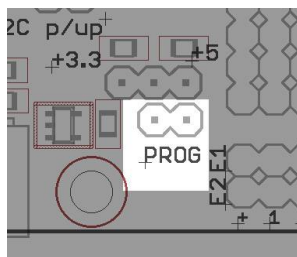


Рис. 13: Джампер отключения от программирования

## 12. Использование прошивки ORFA

Для управления контроллером через UART-порт (например, с ПК, ноутбука, КПК или мобильного телефона) разработана специальная прошивка, одна из последних версий которой доступна для скачивания на странице контроллера: <http://roboforum.ru/wiki/OR-AVR-M32-D>. Прошивка принимает команды через UART-порт, выполняет их на контроллере и выдаёт в этот же порт результат выполнения.

Кроме того в ORFA заложена возможность соединения нескольких ORFA-контроллеров по шине RoboBus (I<sup>2</sup>C), при этом всеми ими можно будет управлять через единый UART-порт, что полезно при создании сложных конструкций, в которых требуется подключение значительного количества периферии.

Прошивка позволяет работать с:

1. Портами GPIO в режиме АЦП (например, читать информацию с ИК-дальномеров Sharp);
2. Портами GPIO в цифровом режиме (низкоскоростной цифровой вход / выход);
3. Портами GPIO в режиме управления модельными сервоприводами (servo, серво, RC-servo);
4. Портом RoboMD2 в режиме управления подключенным драйвером коллекторных двигателей;
5. SPI и I<sup>2</sup>C-устройствами;

С точки зрения внутреннего устройства прошивка состоит из шлюза UART $\rightleftharpoons$ I<sup>2</sup>C и набора драйверов, обрабатывающих команды, обращенные непосредственно к контроллеру. Шлюз принимает команды поступающие через UART, команды настройки шлюза обрабатывает сам, команды направленные к внешним I<sup>2</sup>C-устройствам перенаправляет на шину I<sup>2</sup>C, а команды направленные к адресу I<sup>2</sup>C, закрепленному (специальной командой L настройки шлюза, см. ниже) за самим контроллером — на выполнение соответствующему драйверу.

### Работа с ORFA при использовании ПК или ноутбука

На ПК или ноутбуке UART-порту соответствует понятие «последовательный COM порт». Для работы с таким портом может быть использована как собственная программа, так и любой терминал для работы с COM-портом, например, HyperTerminal встроенный в Microsoft Windows (вызывается через меню Пуск \ Стандартные \ Связь \ HyperTerminal).



## Формат обмена данными шлюза UART<=>I<sup>2</sup>C

Все основные запросы к контроллеру унифицированы с протоколом I<sup>2</sup>C. Это позволяет единообразно обращаться к контроллеру и любым другим I<sup>2</sup>C-устройствам подключенным к нему. Все команды текстовые (для управления контроллера с ПК в самом простом случае будет достаточно любого терминала). Перевод строки означает окончание команды и передачу её на исполнение. Все числа передаются в шестнадцатиричном формате (например, десять будет выглядеть как «0A»)



*Все команды должны оканчиваться переводом строки, который означает окончание команды и передачу её на исполнение!*

Таблица 7. Формат обмена данными шлюза UART<=>I<sup>2</sup>C

Команда	Запрос	Ответ	Примечание
Получить версию протокола ORFA	V	V1.0	
Сбросить порт I <sup>2</sup> C	X	X	
Установить I <sup>2</sup> C-адрес локального контроллера	L<addr>	L<addr>	addr — uint8 (&0xfe)
Установить скорость обмена по I <sup>2</sup> C в КГц	C<freq>	C<freq>	freq — uint16 (100 или 400), КГц
Прочитать регистр	R<addr><reg>[<len>]	SWASR<rdata>P	addr — uint8 reg — uint8 len — uint8, по умолчанию 1 rdata — массив uint8
Записать в регистр	W<addr><reg><data>	SWA (A) + P	addr — uint8 reg — uint8 data — массив uint8 A = Ack
Выполнить I <sup>2</sup> C запрос	S<adr+w><data> S<adr+r><len>	SW (A) + SR<rdata>	adr+w — uint8 (=adr & 0xfe) adr+r — uint8 (=adr   0x01) len — uint8 data — массив uint8 rdata — массив uint8

Работа с драйверами с помощью I<sup>2</sup>C запросов организована через концепцию регистров, часто применяемую в I<sup>2</sup>C-устройствах — ячеек памяти, в которые можно записывать значения и из которых можно их читать. Не дожидаясь конца обработки команды, которую мы передали. Это позволяет разгрузить шину во время обработки команды для работы с другими устройствами.

Для удобной работы с регистрами предусмотрены специальные команды их чтения и записи.

## Команды настройки шлюза UART<=>I<sup>2</sup>C

Для работы непосредственно со шлюзом используется 4 команды: V, X, L<addr> и C<freq>. Примеры использования этих команд приведен ниже (здесь и далее знаками «<» и «>» будет обозначаться направление передачи по UART – в контроллер и от него, соответственно).

Пример отправки запроса версии протокола и получения ответа:

```
< v
> v1.0
```

Пример отправки команды установки I2C-адреса контроллера в 0x20 и получения ответа:

```
< I20
> I20
```

Пример отправки команды установки скорости работы по I2C шине в 100КГц и получения ответа:

```
< C0064
> C0064
```

Пример отправки команды сброса I2C шины и получения ответа:

```
< x
> x
```

## Работа с I<sup>2</sup>C-устройствами через шлюз (порты RoboI2C и RoboBus)

Шлюз UART<=>I<sup>2</sup>C позволяет работать с любым I<sup>2</sup>C устройством используя запросы вида:

(S<adr+w><data>|S<adr+r><len>)+P

На каждый переданный устройству байт будет в ответ получен символ A, а в ответ на каждый запрос нескольких байт будут получены эти несколько байт. Формат ответа на такой запрос:

(SW(A)+|SR<rdata>)+P

## Драйвер низкоскоростного цифрового ввода/вывода (порты GPIO)

Драйвер позволяющий использовать GPIO порты в режимах цифрового логического входа или цифрового логического выхода. Количество управляемых портов  $N=4$  (по количеству выводов МК подключенных к GPIO-портам). Количество регистров, используемых драйвером  $N*2$ . Далее будем считать, что {rr} - первый регистр драйвера, а {rr+n} - (n+1)-й регистр драйвера.

Первые N регистров, это чтение/запись значения, оставшиеся N регистров - чтение/запись режима порта. Каждой из 8 линий порта соответствует 1 бит. При работе в режиме входа запись в значение порта включает (1) или выключает (0) подтягивающие резисторы контроллера.

Таблица 8. Порядок работы

Действие	Регистр*	Отправляемая в регистр или читаемая из него последовательность байт
Установить режим работы линий ввода\вывода порта	Порт A - 0x06 Порт B - 0x07 Порт C - 0x08 Порт D - 0x09	<p>xx - устанавливаемая маска, какие линии GPIO порта включить на вход (0), а какие на выход (1)</p> <p>Пример* задания 2-ой и 3-ей линиям порта PORTB режима работы "цифровой вход", а остальным битам этого порта режим "цифровой выход":</p> <pre>&lt;S40 07 F3 P &gt;SWAAP</pre> <p><i>Примечание*: Отдавая такую команду, считаем, что ранее командой L40 установили шлюзу локальный адрес 0x40.</i></p>
Установить уровень на выходе цифрового порта или включить/выключить подтягивающий резистор на входе цифрового порта	Порт A - 0x02 Порт B - 0x03 Порт C - 0x04 Порт D - 0x05	<p>xx - устанавливаемая маска, на какие линии GPIO порта подать высокий логический уровень или включить подтягивающий резистор (1), а на какие подать низкий логический уровень или выключить подтягивающий резистор (0)</p> <p>Пример* Установить порт A0 контроллера в логический выход и выставить на нём значение 1:</p> <pre>&lt;S40 06 01 S40 02 01 P &gt;SWAASWAAP</pre> <p><i>Примечание*: Отдавая такую команду, считаем, что ранее командой L40 установили шлюзу локальный адрес 0x40.</i></p>
Прочитать уровень на входе цифрового порта	Порт A - 0x02 Порт B - 0x03 Порт C - 0x04 Порт D - 0x05	<p>xx - читаемая маска, на каких линиях GPIO порта какой логический уровень сейчас присутствует или какой логический уровень мы выставили.</p> <p>Пример* Установить порт A0 контроллера в логический вход и прочитать его значение:</p> <pre>&lt;S40 06 00 S40 02 S41 01 P &gt;SWAASWASR01P</pre> <p><i>Примечание*: Отдавая такую команду, считаем, что ранее командой L40 установили шлюзу локальный адрес 0x40.</i></p>

**Примечание\*** Указаны номера регистров действующей на момент написания этого текста версии прошивки ORFA. Чтобы не было опасности изменения своего ПО при смене номеров этих регистров можно использовать драйвер инстропекции (см. далее).

## Драйвер АЦП (порты GPIO)

Драйвер позволяющий управлять блоком Аналого-Цифрового Преобразователя (далее АЦП). Пока поддерживается только 8 входов АЦП (весь порт А любого контроллера). Количество регистров, используемых драйвером - 2. Далее будем считать, что {сс} - первый регистр драйвера, {тт} - второй регистр.

В действующей на момент написания этого текста версии прошивки ORFA, эти регистры имеют номера 0x0E и 0x0F соответственно. Чтобы не было опасности изменения своего ПО при смене номеров этих регистров можно использовать драйвер инспекции (см. далее).

Первый регистр - конфигурационный, в него пишутся 2 байта всегда:

1. Настройки АЦП (используются младшие 3 бита):

- биты 0..1:
  - ◆ 00 - External reference
  - ◆ 01 - AVCC
  - ◆ 10,11 - Internal reference
- бит 2 - режим точности АЦП:
  - ◆ 0 - 8-битный;
  - ◆ 1 - 11-битный;

2. Маска включаемых в АЦП каналов (8 бит - соотв. 8 линиям порта А);

Второй регистр для чтения результатов - из него читаются 8 байт, если режим 8-битный или 16 байт, если режим 16-битный. Можно читать меньше, тогда будем знать состояние первых нескольких линий.



*ВНИМАНИЕ!!! Чтение линий АЦП происходит в автоматическом циклическом режиме, поэтому после конфигурирования АЦП необходимо сделать паузу в 10мс перед чтением значений.*

Пример 1\*. Включение PORTA.2 в 8-битный режим АЦП с внутренним опорным напряжением.

<S 40 0E 03 04 P

>SWAAAP

S40 0E 03 04 - отправим в устройство 0x40 номер регистра 0E и два байта конфигурации 03 – режим внутреннего источника опорного напряжения, 8 бит, 0x04 — включить АЦП только по 2 линии порта, P - конец запроса. Ответ: SW - начали запись в устройство, AAA - подтверждения записи 3 байт в устройство, P - конец ответа.

Пример 2\*. Читаем 0-3 каналы АЦП.

<S 40 0F S 41 04 P

>SWASRFFD7D7D6P

S40 0F - отправим в устройство 0x40 номер регистра 0E, после чего S 41 04 – прочитаем теперь из устройства четыре байта значений из АЦП, P - конец запроса. Ответ: SW - начали запись в устройство, A - подтверждения записи 1 байта в устройство, SR<aa><bb><cc><dd> - возвращенные значения на входах АЦП, P - конец ответа.

---

*Примечание\*: Отдавая такую команду, считаем, что ранее командой L40 установили шлюзу локальный адрес 0x40.*

## Драйвер управления сервоприводами (порты GPIO)



Модельные сервоприводы (RC-servo) управляются прямоугольными сигналами частотой 50Гц, уровнем 3-6 В и скважностью (временем, которое на линии высокое напряжение) от 1 до 2мс. Поэтому не зная характеристик конкретного подключенного сервопривода часто приходится говорить о положении сервопривода, указывая лишь скважность управляющего сигнала. Например, 1200мкс. Это означает, что если сервопривод поворачивается на угол  $+75^\circ$  при управляющих сигналах 1мс и 2мс соответственно, тогда 1200мкс соответствует примерно  $-75^\circ + 2 \cdot 75^\circ \cdot (1200 - 1000) / (2000 - 1000) = -45^\circ$ .

Для контроллера драйвер управления сервоприводами имеет два регистра, через первый осуществляется настройка, на какие порты подключены сервоприводы, а через второй определяется положение каждого из включенных сервоприводов.

В действующей на момент написания этого текста версии прошивки ORFA, эти регистры имеют номера 0x10 и 0x11 соответственно. Чтобы не было опасности изменения своего ПО при смене номеров этих регистров можно использовать драйвер интроспекции (см. далее).

Таблица 8. Порядок работы

Действие	Регистр	Отправляемая в регистр последовательность байт
Определить, на какие порты подключены сервоприводы	0x10	<p>M1 M2 двухбайтовая маска, какие сервоприводы включить</p> <p>Пример* включения 2-3 порта для управления сервами:  <code>&lt;S40 10 0C 00 P</code>  <code>&gt;SWAAAP</code></p> <p>Примечание*: Отдавая такую команду, считаем, что ранее командой L40 установили шлюзу локальный адрес 0x40.</p>
Установить положение сервоприводов	0x11	<p>[{aa} {bb}{cc}]* — установить положения сервоприводов (сервопривод {aa} в положение 0x{bb}{cc} мкс).</p> <p>Пример* установки положения 2-го сервопривода в 1200мкс (0x04B0), а 3-го сервопривода в положение 1700мкс (0x06A4):  <code>&lt;S40 11 02 04B0 03 06A4 P</code>  <code>&gt;SWAAAP</code></p> <p>Примечание*: Отдавая такую команду, считаем, что ранее командой L40 установили шлюзу локальный адрес 0x40. А также включили 2 и 3 сервопривод, например, командой из вышеуказанного примера.</p>

Таблица 9. Соответствие между номерами серво и портами ввода-вывода

Номер серво	Порт МК
0..7	PORTA.0 – PORTA.7
8..11	PORTC.7 – PORTC.4
12..13	PORTB.3 – PORTB.2
14..15	PORTD.5 – PORTD.4

## Драйвер управления коллекторными двигателями (порт RoboMD2)



Коллекторные двигатели постоянного тока управляются подачей фиксированного напряжения в постоянном или ШИМ-режиме в одном из двух возможных направлений (в прямом и обратном). В ШИМ-режиме напряжение подаётся прямоугольными импульсами, ширина которых к длине шага называется скважность. При скважности 100% на двигатели подаётся постоянное напряжение. ШИМ повсеместно используется вместо регулировки напряжения на двигателе для того, чтобы минимизировать неэффективные потери (повысить КПД) и нагрев частей драйвера. С помощью регулировки скважности можно в некотором диапазоне добиться управления скоростью вращения двигателя.

В этом драйвере всегда 4 регистра. Далее будем считать, что начальный регистр этого драйвера - AA, второй регистр BB, третий CC, четвертый DD.

В действующей на момент написания этого текста версии прошивки ORFA, эти регистры имеют номера 0x0C-0x0F соответственно. Чтобы не было опасности изменения своего ПО при смене номеров этих регистров можно использовать драйвер интроспекции (см. далее).

Таблица 10. Порядок работы

Действие	Регистр	Отправляемая в регистр последовательность байт
Определить скважность ШИМ двигателя №1	0x0C	<p>M1 – ШИМ от 0 (остановлен) до 255 (полный ход)</p> <p>Пример* включения двигателя №1 на полный ход:  <code>&lt;S40 0C FF P</code>  <code>&gt;SWAAAP</code></p> <p>Примечание*: Отдавая такую команду, считаем, что ранее командой L40 установили шлюзу локальный адрес 0x40.</p>
Определить скважность ШИМ двигателя №2	0x0D	<p>M2 – ШИМ от 0 (остановлен) до 255 (полный ход)</p> <p>Пример* включения двигателя №2 на 50% мощности:  <code>&lt;S40 0D 80 P</code>  <code>&gt;SWAAAP</code></p> <p>Примечание*: Отдавая такую команду, считаем, что ранее командой L40 установили шлюзу локальный адрес 0x40.</p>
Определить направление вращения двигателя №1	0x0E	<p>D1 – направление, вперед (0) или назад (1)</p> <p>Пример* включения двигателя №1 на обратное направление:  <code>&lt;S40 0E 01 P</code>  <code>&gt;SWAAAP</code></p> <p>Примечание*: Отдавая такую команду, считаем, что ранее командой L40 установили шлюзу локальный адрес 0x40.</p>
Определить направление вращения двигателя №2	0x0F	<p>D2 – направление, вперед (0) или назад (1)</p> <p>Пример* включения двигателя №1 на прямое направление:  <code>&lt;S40 0F 00 P</code>  <code>&gt;SWAAAP</code></p> <p>Примечание*: Отдавая такую команду, считаем, что ранее командой L40 установили шлюзу локальный адрес 0x40.</p>

## **Драйвер SPI-интерфейса (порт RoboBus)**

Приносим извинения, описание драйвера пока не готово.

## Драйвер интроспекции

Специальный драйвер, всегда закрепленный за регистром 0x00. Позволяет узнать какие вообще драйверы присутствуют в устройстве, какой версии эти драйвера и какие регистры им назначены.

Типы запросов (регистр, данные...):

- 00 00 — запросить количество драйверов, после чего нужно прочитать 1 байт;
- 00 nn — запросить информацию о драйвере nn, после чего нужно прочитать 6 байт;

Таблица 11. Структура информации в ответе на запрос 00 nn

Байт	Название	Комментарий
0..1	UID	Уникальный идентификатор драйвера.  Может быть получен у команды проекта «Open Robotics». Таблица уже выданных идентификаторов: <a href="http://roboforum.ru/wiki/ORFA/UID">http://roboforum.ru/wiki/ORFA/UID</a> .  Для личного использования зарезервирован диапазон адресов с первым байтом 0xFF.
2..3	Major Version, Minor Version	Старший и младший номер версии.
4	Start Register	Номер начального регистра драйвера.
5	Count of registers	Количество регистров драйвера.  То есть все регистры драйвера это интервал: (Start Register) .. (Start Register + Count of registers - 1)

### Пример 1\*. Определяем сколько драйверов в устройстве

```
<S40 00 00 S41 01 P  
>SWAASR06P
```

S40 00 00 - отправим в устройство 0x40 номер регистра 00 и тип запроса 00 - получить количество драйверов, S41 01 - получить ответ 1 байт, P - конец запроса. Ответ: SW - начали запись в устройство, AA - подтверждения записи 2 байт в устройство, SR - начали получать ответ от устройства, 06 - получили 1 байт ответа (количество драйверов), P - конец ответа.

### Пример 2\*. Определяем информацию о драйвере №2

```
<S40 00 02 S41 06 P  
>SWAASR003101001002P
```

S40 00 02 - отправим в устройство 0x40 номер регистра 00 и тип запроса 02 - получить информацию о драйвере 0x02, S41 01 - получить ответ 6 байт, P - конец запроса. Ответ: SW - начали запись в устройство, AA - подтверждения записи 2 байт в устройство, SR - начали получать ответ от устройства, 003101001002 - получили 6 байт ответа (информация о драйвере - 0x0031 - идентификатор тип драйвера, 0x01.0x00 - версия драйвера, 0x10 - номер первого регистра драйвера, 0x02 - количество регистров драйвера), P - конец ответа.

---

*Примечание\*:* Отдавая такую команду, считаем, что ранее командой L40 установили шлюзу локальный адрес 0x40.