

ShIoTiny

*Контроллер для тех, кто не любит программировать, но
любит рисовать*

Описание узлов и краткая справка по редактору

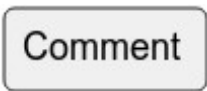
2019

Оглавление

1	Описание узлов (нод).....	3
1.1	Просто комментарий.....	3
1.2	Базовая логика (Basic logic).....	3
1.3	Простейшие входы и выходы (One-bit digital inputs, One-bit digital outputs).....	6
1.4	Аналого-цифровой преобразователь АЦП, он же ADC.....	8
1.5	Связь по протоколу MQTT.....	11
1.5.1	Сервер, порт, авторизация.....	12
1.5.2	Префикс.....	13
1.5.3	Безопасность (SSL).....	13
1.6	Связь по протоколу Rlink UDP.....	14
1.7	Датчики температуры и влажности.....	16
1.8	Математические функции.....	17
1.9	Функции работы со временем.....	18
1.10	Функции управления прохождением событий.....	20
1.11	Функции сетевого сервиса.....	21
	Приложение А Вкладка состояния ShIoTiny (Control and status).....	22
	Приложение Б Вкладка настройки сети и режимов работы ShIoTiny (Networking).....	23
	Приложение В Работа с редактором ElDraw.....	24

1 Описание узлов (нод)

1.1 Просто комментарий



Информация, комментарии (Comment)		
		Это просто комментарий. Он ничего не делает. Чтобы написать свой комментарий — перетащите узел на схему и кликните по нему мышкой.






1.2 Базовая логика (Basic logic)

Базовая логика — это то самое, что в цифровой схемотехнике называют «логическими элементами». Они являются основой основ цифровой логики. Опиерируют эти элементы только двумя цифрами 0 и 1, то есть бинарными данными.

Ниже приведена таблица, которая ясно дает понять — что будет у элемента на выходе при тех или иных значениях на входах.

Помимо базовых элементов, в этот раздел внесены ещё и регистры, триггеры и мультиплексоры. В них используются не только бинарные но и целые/вещественные входы и выходы.


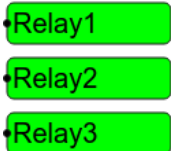
Узел (нода)	Событие	Примечание															
Основные логические элементы (Basic logic). Входы и выходы — бинарные.																	
		Логический элемент НЕ (отрицание) <div><table><tr><th>Вход</th><th>Выход</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table></div>	Вход	Выход	0	1	1	0									
Вход	Выход																
0	1																
1	0																
		Логический элемент И <div><table><tr><th>Вход1</th><th>Вход2</th><th>Выход</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table></div>	Вход1	Вход2	Выход	0	0	0	0	1	0	1	0	0	1	1	1
Вход1	Вход2	Выход															
0	0	0															
0	1	0															
1	0	0															
1	1	1															

		Логический элемент ИЛИ <table border="1" data-bbox="801 230 1134 495"><tr><th>Вход 1</th><th>Вход 2</th><th>Выход</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	Вход 1	Вход 2	Выход	0	0	0	0	1	1	1	0	1	1	1	1					
Вход 1	Вход 2	Выход																				
0	0	0																				
0	1	1																				
1	0	1																				
1	1	1																				
		Логический элемент ИСКЛЮЧАЮЩЕЕ ИЛИ <table border="1" data-bbox="801 582 1125 848"><tr><th>Вход 1</th><th>Вход 2</th><th>Выход</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	Вход 1	Вход 2	Выход	0	0	0	0	1	1	1	0	1	1	1	0					
Вход 1	Вход 2	Выход																				
0	0	0																				
0	1	1																				
1	0	1																				
1	1	0																				
		Триггер RS <table border="1" data-bbox="679 938 1252 1240"><tr><th>Вход S</th><th>Вход R</th><th>Выход Q</th><th>Примечание</th></tr><tr><td>0</td><td>0</td><td>Q_{t-1}</td><td>Хранение</td></tr><tr><td>0</td><td>1</td><td>0</td><td>Установка 0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>Установка 1</td></tr><tr><td>1</td><td>1</td><td>Q_{t-1}</td><td>Хранение</td></tr></table> <p>Q_{t-1} — предыдущее состояние</p>	Вход S	Вход R	Выход Q	Примечание	0	0	Q_{t-1}	Хранение	0	1	0	Установка 0	1	0	1	Установка 1	1	1	Q_{t-1}	Хранение
Вход S	Вход R	Выход Q	Примечание																			
0	0	Q_{t-1}	Хранение																			
0	1	0	Установка 0																			
1	0	1	Установка 1																			
1	1	Q_{t-1}	Хранение																			
Основные логические элементы (Basic logic) Входы и выходы разнотипные																						
		Регистр. Вход D и выход Q — <i>целые/вещественные</i> . C — <i>бинарный вход</i> <table border="1" data-bbox="679 1491 1252 1688"><tr><th>Вход D</th><th>Вход C</th><th>Выход Q</th><th>Примечание</th></tr><tr><td>X</td><td>1</td><td>X</td><td>Установка Q</td></tr><tr><td>X</td><td>0</td><td>X_{t-1}</td><td>Хранение</td></tr></table> <p>Когда $C \neq 0$, $Q=D$. Когда $C=0$, данные на выходе D не изменяются (хранение)</p>	Вход D	Вход C	Выход Q	Примечание	X	1	X	Установка Q	X	0	X_{t-1}	Хранение								
Вход D	Вход C	Выход Q	Примечание																			
X	1	X	Установка Q																			
X	0	X_{t-1}	Хранение																			
		Мультиплексор. Входы A , B и выход — <i>целые/вещественные</i> . b/a — <i>бинарный вход</i> .																				

		Вход A	Вход B	Вход b/a	Выход	Примечание
		A	B	0	A	
		A	B	1	B	

1.3 Простейшие входы и выходы (One-bit digital inputs, One-bit digital outputs)

«Простейшие», это значит имеющие два состояния — включено/выключено, замкнуто-разомкнуто. Такие входы и выходы еще называют «бинарными».

Узел (нода)	Событие	Примечание
Простейшие входы и выходы (One-bit digital inputs, One-bit digital outputs)		
	ДА	Бинарный выход. Дискретные входы, рассчитанные на подключение датчика типа «сухой контакт». То есть — кнопки, выключателя, геркона и т.п. На выходе узла Input «0», если контакты разомкнуты и «1», если контакты замкнуты. У ShIoTiny есть три абсолютно идентичных входа типа «сухой контакт» - Input1, Input2 и Input3 .
		Бинарный вход. Реле — это один из видов выхода типа «сухой контакт». То есть металлические провода, которые могут быть замкнуты или разомкнуты по нашему желанию. У ShIoTiny есть три абсолютно идентичных переключающих реле, рассчитанных на нагрузку до 250В 1А. Если на вход узла Relay подать «1», то реле включится, если подать «0» — то выключится.

Лирическое отступление первое. Про «сухие контакты». «Сухой контакт» — это контакт, который не имеет собственного источника напряжения. То есть просто два любых металлических проводника, которые можно замкнуть между собой и разомкнуть их. Под это определение попадают масса датчиков — кнопки, выключатели, поплавковые датчики уровня жидкости, геркона (датчики магнитного поля) и так далее. Очевидно, что все эти датчики, можно подключать ко входам контроллера ShIoTiny. Но электрических схемах нормально разомкнутые «сухие контакты» обычно обозначаются так, как показано на рисунке.



Нормально разомкнутые — это значит, разомкнутые при отсутствии внешнего воздействия — кнопка не нажата, выключатель не включен, у геркона нет поблизости магнита...

Есть также и нормально замкнутые «сухие контакты». На электрических схемах они обычно обозначаются так, как показано на рисунке.

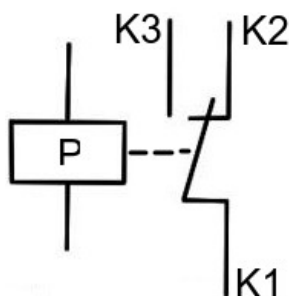


В противоположность нормально разомкнутым, нормально замкнутые «сухие контакты» в отсутствии внешнего воздействия — замкнуты.

Какие именно контакты использовать — решать вам. И те и другие «сухие контакты» можно спокойно подключать ко входам Input контроллера ShIoTiny.

Лирическое отступление второе. Про реле. Реле бывают разные — нет, не по цвету, а по назначению и характеристикам. Бывают включающие, выключающие, поляризованные, тактовые... Много их, в общем.

В контроллере ShIoTiny установлены три переключающих реле. Переключающие, это значит, что у них есть три контакта: один общий, один нормально разомкнутый с общим и один — нормально замкнутый с общим. Условно это показано на рисунке ниже.



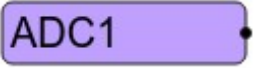
Когда питание на обмотку реле не подается (контроллер вообще выключен или на вход узла Relay подается 0) — замкнуты контакты K1 и K2. Такие контакты реле, которые замкнуты в отсутствие питания на его обмотке и называются «нормально замкнутые». Если подать питание на обмотку реле (в нашем случае это значит, подать 1 вход узла Relay), то контакты K1 и K2 разомкнутся, а контакты K1 и K3 — замкнутся. Контакты K1 и K3 называются «нормально разомкнутыми», потому что в отсутствие питания на обмотке реле они разомкнуты.

Надеюсь, что теперь все ясно с контактами и реле.

1.4 Аналого-цифровой преобразователь АЦП, он же ADC

Аналого-цифровой преобразователь — основа основ цифровых измерений.

Что он делает? На вход АЦП подается напряжение в диапазоне от 0 до 1В. На выходе АЦП мы получаем число, пропорциональное входному напряжению. То есть, если на вход подать 0В — то с выхода АЦП получим 0. Если на вход АЦП подать 1В — то на выходе получим число 1023.

Аналого-цифровой преобразователь АЦП, он же ADC		
	ДА	Вещественный выход. Аналого-цифровой преобразователь (аппаратный). Значение на выходе вычисляется по формуле: $X = k \cdot u_{\text{вх}} + b$, где $u_{\text{вх}}$ — напряжение на входе ADC (от 0 до 1В); k — диапазон (ADC range) и b -смещение (ADC offset). Если напряжение больше 1В, то на выходе устанавливается значение NaN (не число).

Число 1023 — максимальное, потому что АЦП ShIoTiny — 10разрядный. То есть имеет выходной диапазон значений от 0 до $2^{10}-1$. Если мы подадим на вход АЦП напряжение больше 1Вольта, то с выхода будет считываться значение с 1 в разряде ошибки переполнения. В случае ошибки, аппаратный АЦП выставляет значение 1024 в своем выходном регистре.

Все это касается аппаратного АЦП. Узел АЦП немного более сложен. Во-первых, он пересчитывает измеренное значение к диапазону 0..1. Во-вторых, он масштабирует это значение заданным коэффициентом **k** и сдвигает его на величину **b**. Зачем это надо? Очень просто: нам удобнее пересчитать входное напряжение сразу в привычные нам величины и работать уже с ними, а не с абстрактными «попугаями» - числом 0..1023.

Кроме того, если к АЦП подключен какой-то аналоговый датчик, например температуры или уровня жидкости — то можно сразу пересчитать измеряемое им значение в градусы или сантиметры.

Итак, как работает узел АЦП? Рассмотрим по шагам.

Шаг первый. Аппаратный АЦП измеряет входное напряжение и получает число 0..1024. (Помним, что 1024 — это переполнение АЦП!)

Если определено переполнение АЦП (подано напряжение выше 1В на вход), то на выходе модуля выставляется значение NaN (не число) и цикл изменений завершается.

Иначе, если все в порядке и никаких переполнений нет, то измеренное значение (0..1023) приводится к диапазону 0..1 (то есть число, считанное с АЦП просто делится на 1023). Это приведенное значение и называется $u_{\text{вх}}$ в наших формулах.

Шаг второй. Вычисляется выходное значение узла АЦП по формуле: $X = k \cdot u_{\text{вх}} + b$. Обратите внимание, что коэффициенты k и b можно задавать больше единицы, меньше единицы, положительными или отрицательными. Так мы можем пересчитывать измеренное значение практически в любой нужный нам линейный диапазон.

Шаг третий. Если полученное значение X отличается от предыдущего измеренного значения X_{t-1} на величину, превышающую заданный процент диапазона k , то генерируется событие. О событиях мы поговорим в другом месте. Пока достаточно знать, что узел АЦП не реагирует на изменение измеренной величины X , пока изменение не станет больше заданного процента диапазона k .

Как нам задать все эти величины? Очень просто. Перетащите узел АЦП на схему и ткните на него мышкой. Появится диалог настройки АЦП.

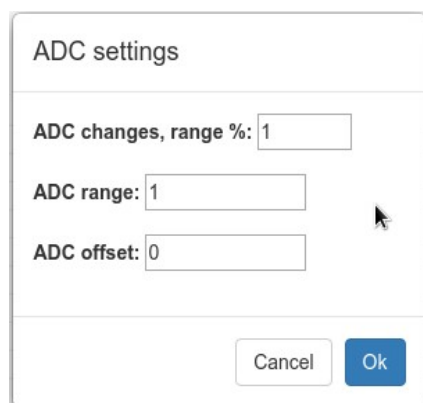


Рис. 1: Диалог настройки узла АЦП

Коэффициент k вводится в полк **ADC range** (диапазон). Коэффициент b вводится в полк **ADC offset** (смещение).

Минимальный процент изменения величины, при которой узел АЦП начинает реагировать на эти изменения задаётся в поле **ADC changes, range**.

Для чего нам надо задавать процент изменения? Да чтобы слишком часто не реагировать на события от АЦП.

Запутались? Я понимаю, сам запутался!

Для наглядности приведу пример. Например, мы ходим измерять напряжение в диапазоне от 0 до 18 вольт. Но на вход АЦП нам можно подавать

только не более 1Вольта. Как быть? Конечно же — ставим на вход делитель. На рисунке это показано. Слева, на вход делителя подается напряжение в диапазоне от 0 до 18Вольт. Делитель его делит на 18 и подает на вход АЦП.

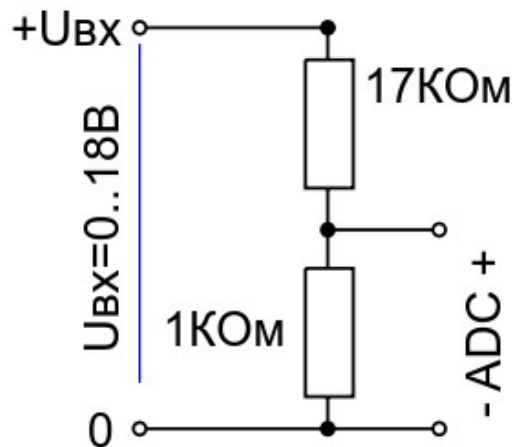


Рис. 2: Делитель напряжения на 18

Все, задача деления напряжения решена — на входе АЦП у нас теперь не больше 1Вольта. Но мы хотим, чтобы с выхода узла АЦП считывалось число 18 при 18Вольтах напряжения на входе делителя! Зачем нам какой-то 1Вольт?

Этому горю несложно помочь. Заходим в диалог настройки АЦП и устанавливаем k (**ADC range**) равным 18; устанавливаем b (**ADC offset**) равным 0.

Остается поле **ADC changes, range**. Каким его выбрать? Смотрите сами. Если это поле выставить в 1%, то АЦП будет реагировать на изменение напряжения $\frac{18 \text{ Вольт}}{100\%} \cdot 1\%$ или 0.18Вольта, а если 10%, то на изменение напряжения $\frac{18 \text{ Вольт}}{100\%} \cdot 10\%$ или 1.8Вольта. Выбор за вами.

Меньше 1% и больше 100% выставить нельзя. Обычно, для измерения напряжения питания достаточно реагировать на 5%-е изменение этого напряжения.

1.5 Связь по протоколу MQTT

MQTT (англ. message queuing telemetry transport) — упрощенный сетевой протокол, работающий поверх TCP/IP, ориентированный для обмена сообщениями между устройствами по принципу издатель-подписчик.

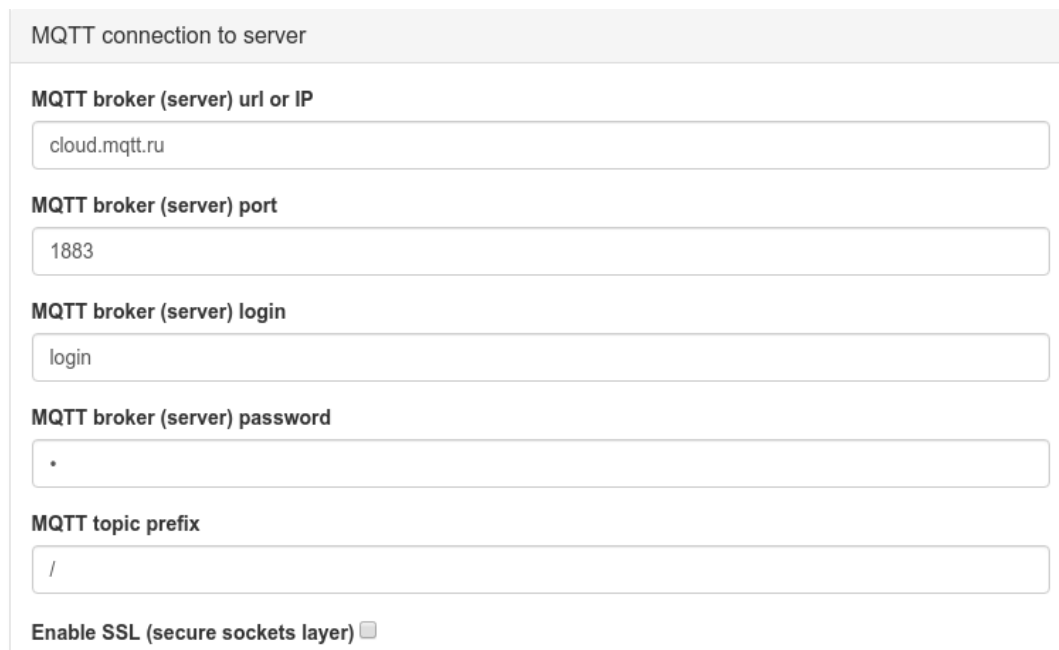
MQTT это святой грааль IoT (интернета вещей). На него сегодня молятся и уже кое-где начали приносить жертвы:)

Суть MQTT проста: одно устройство подписывается на некоторые данные (говорит серверу — какие данные ему нужны), а другое устройство — публикует эти данные. Разумеется, чтобы устройство могло публиковать данные или подписываться на них, оно должно сначала авторизоваться на сервере. Узлы для работы с протоколом MQTT показаны в таблице.

Публикация и подписка данных по протоколу MQTT (Mqtt)												
<div>MQTT publish topic</div>		<p>Вещественный вход. Узел публикует на MQTT-брокере входное значение. topic — название темы. Его можно изменить, кликнув мышкой по названию темы (topic) на узле MQTT publish на схеме. Настройка подключения к MQTT-брокеру производится на вкладке Networking.</p> <table><tr><td>Вход</td><td>Публикуемое значение</td></tr><tr><td>Целое число N</td><td>Число N</td></tr><tr><td>Вещественное число X</td><td>Число X</td></tr><tr><td>Бесконечность (Inf)</td><td>Строка «Inf»</td></tr><tr><td>Не-число (NaN)</td><td>Строка «NaN»</td></tr></table>	Вход	Публикуемое значение	Целое число N	Число N	Вещественное число X	Число X	Бесконечность (Inf)	Строка «Inf»	Не-число (NaN)	Строка «NaN»
Вход	Публикуемое значение											
Целое число N	Число N											
Вещественное число X	Число X											
Бесконечность (Inf)	Строка «Inf»											
Не-число (NaN)	Строка «NaN»											
<div>MQTT describe topic</div>	ДА	<p>Вещественный выход. Узел подписывается на заданную тему на MQTT-брокере и возвращает значение этой темы при новой публикации. topic — название темы. Его можно изменить, кликнув мышкой по названию темы (topic) на узле MQTT describe на схеме. Настройка подключения к MQTT-брокеру производится на вкладке Networking. При приеме с сервера нового значения указанной темы, генерируется событие для пересчета состояния узлов, связанных с выходом узла MQTT describe. Принимаются значения: число, бесконечность (Inf) и не-число (NaN).</p>										

Подобно рассказывать про MQTT-протокол я здесь не могу, но расскажу про настройку связи ShIoTiny с MQTT-сервером.

Чтобы публиковать свои данные и получать команды, контроллеру ShIoTiny необходимо установить соединение с MQTT-брокером. Для настройки соединения с MQTT-брокером, зайдите на вкладку «Сеть» («**Networking**») и сосредоточьтесь на разделе «Соединение с MQTT-сервером» («**MQTT connection to server**»). В нем мы можем настроить параметры доступа контроллера ShIoTiny к MQTT-серверу (брокеру).



MQTT connection to server

MQTT broker (server) url or IP

cloud.mqtt.ru

MQTT broker (server) port

1883

MQTT broker (server) login

login

MQTT broker (server) password

•

MQTT topic prefix

/

Enable SSL (secure sockets layer) ☐

Рис. 3: Параметры подключения к MQTT-серверу

Прежде всего, необходимо указать URL или IP-адрес MQTT-сервера и порт, к которым контроллер ShIoTiny будет присоединяться.

1.5.1 Сервер, порт, авторизация

Параметры подключения к MQTT-брокеру (серверу) вводятся соответственно в поля ввода «Адрес MQTT-брокера (сервера)» («**MQTT broker (server) url or IP**») и «Порт MQTT-брокера (сервера)» («**MQTT broker (server) port**»).

Для авторизации на MQTT-брокере вводится имя пользователя и пароль в поля «Имя пользователя MQTT-брокера» («**MQTT broker (server) login**») и «Пароль MQTT-брокера» («**MQTT broker (server) password**»).

Если авторизация на MQTT-брокере не требуется, то поля ввода имени пользователя и пароля для входа на MQTT-брокер не заполняются (сотрите все, что в них написано и оставьте эти поля пустыми).

1.5.2 Префикс

Префикс MQTT-параметров — это строка, добавляемая к названию темы при публикации и подписке на MQTT-брокере. Чтобы установить MQTT-префикс для вашего контроллера, его надо просто ввести в поле ввода **«Префикс темы MQTT»** (**«MQTT topic prefix»**). Префикс всегда начинается со слеша («/»)! Если вы не введете слэш в поле ввода — он добавится автоматически. В префиксе нельзя использовать символы «#» и «+». Других ограничений нет.

Если вы публикуете параметр **«status»** (или подписываетесь на него), а ваш префикс задан как **«/shiotiny/»**, то на брокере этот параметр будет опубликован под именем **«/shiotiny/status»**. Если у вас задан пустой префикс, то все параметры на брокере будут начинаться со слеша («/»): **«status»** будет публиковаться как **«/status»**.

1.5.3 Безопасность (SSL)

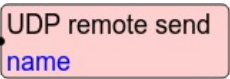
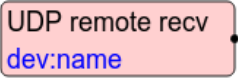
Для обеспечения безопасности, имеется возможность работать через «защищенные соединения» по технологии **SSL (Secure Sockets Layer** — уровень защищенных сокетов).

Если ваш MQTT-брокер использует SSL, то вы должны установить переключатель **«Разрешить SSL (защищенные соединения)»** (**«Enable SSL (secure sockets layer)»**) на вкладке **«Сеть»** (**«Networking»**).

1.6 Связь по протоколу Rlink UDP

Протокол Rlink UDP служит для обмена данными между устройствами в пределах локальной сети. Обмен данными ведется при помощи широковещательной рассылки (multicast). Разумеется, если вы создадите локальную сеть при помощи VPN, то можете обмениваться данными между устройствами, расположенными хоть на разных концах Земли.

Это значит, что все устройства «слышат» любой пакет, переданный одним из них.

Обмен данными протоколу Rlink UDP		
		Вещественный вход. Передача значения по протоколу Rlink UDP с заданным именем. name — имя передаваемого значения. Его можно изменить, кликнув нему мышкой, когда узел UDP remote send находится в схеме. Имя передаваемого значения не должно содержать двоеточий!
	ДА	Вещественный выход. Прием значения по протоколу Rlink UDP с заданным именем устройства и именем принимаемого значения. Имя устройства (dev) отделяется от имени параметра (name) двоеточием. Если имя устройства не задано (введено только имя принимаемого значения - name), то будут приниматься значения с указанным именем от любого устройства. Изменить имя устройства и значения можно кликнув по нему мышкой, когда узел UDP remote recv находится в схеме. Имя устройства и имя значения не должно содержать двоеточий! Двоеточия служат для отделения имени устройства от имени значения.

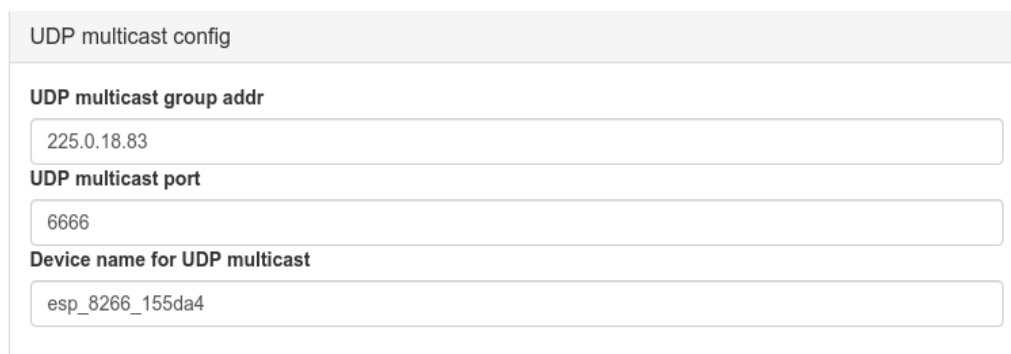
Каждое значение, которое отправляется по протоколу Rlink UDP помимо собственно значения, включает в себя имя устройства (**dev**) и имя значения (**name**).

Имя устройства задается одно для всех значений, а имя значения — непосредственно в узле.

Таким образом, разные устройства могут публиковать значения под одинаковыми именами и путаницы не произойдет.

Чтобы воспользоваться обменом данными по протоколу Rlink UDP, необходимо предварительно настроить параметры сети на вкладке **Networking**, раздел **UDP multicast config**.

Заметим, что обычно, параметры по умолчанию изменять не надо, за исключением имени устройства.



The image shows a web-based configuration interface titled "UDP multicast config". It contains three labeled input fields. The first field, "UDP multicast group addr", contains the text "225.0.18.83". The second field, "UDP multicast port", contains the text "6666". The third field, "Device name for UDP multicast", contains the text "esp_8266_155da4".

Рис. 4: Параметры передачи и приема данных по протоколу RLinkUDP

Здесь задаются три параметра:

- **адрес групповой рассылки** в пределах 224.0.0.1 — 239.255.255.254 (UDP multicast group addr);
- **порт групповой рассылки** в пределах 1000 — 65535 (UDP multicast port);
- **название (имя) устройства** (не более 16 символов) (Device name for UDP multicast). По умолчанию — имя устройства содержит серийный номер платы. Если вам захочется чего-то более красивого — введите свое имя. Имя устройства не должно содержать двоеточий!

1.7 Датчики температуры и влажности

Датчики температуры и влажности воздуха, подключаются к специальному входу DHT. Разумеется, подключить можно только один из датчиков, не более.

Датчики температуры и влажности		
	ДА	Вещественный выход. Выход Т-температура воздуха Выход Н-влажность воздуха
	ДА	Вещественный выход. Выход Т-температура воздуха Выход Н-влажность воздуха
	ДА	Вещественный выход. Выход Т-температура воздуха Выход Н-влажность воздуха
	ДА	Вещественный выход. Выход Т-температура воздуха Выход Н-влажность воздуха

Если датчик неисправен или физически отключен, то с выходов узлов датчиков Т и Н считывается значение NaN (не-число).

1.8 Математические функции

Самые простые и понятные функции. Даже пояснять ничего почти не надо. Если кто-то их не понял — то вам нужен учебник арифметики для 2-го класса средней школы.

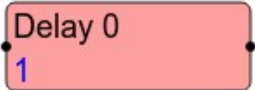
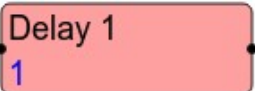
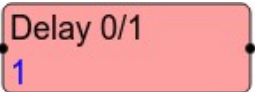

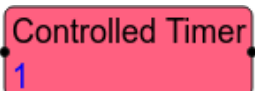

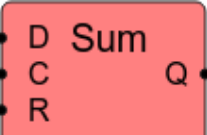
Математические функции (Math functions)		
		Вещественный выход. Константа. Любое число. Для изменения константы перетащите узел на схему и ткните на него мышкой.
		Вещественный выход. Константа «не-число». Это особая константа, которая не равна ни одному числу и даже самой себе.
		Вещественный выход. Бесконечность. Константа, которая больше любого числа.
		Вещественные вход и выход. Проверка, является ли значение на входе «не-числом». Если является — то на выходе 1. Иначе на выходе 0.
		Вещественные вход и выход. Проверка, является ли значение на входе бесконечностью. Если является — то на выходе 1. Иначе на выходе 0.
		Вещественные входы и выход. Выход — сумма чисел на входах $A + B$.
		Вещественные входы и выход. Выход — разность чисел на входах $A - B$.
		Вещественные входы и выход. Выход — произведение чисел на входах $A * B$.
		Вещественные входы и выход. Выход — частное от деления чисел на входах A/B .
		Вещественные входы и бинарный выход. Если $A=B$, то на выходе 1, иначе на выходе 0.
		Вещественные входы и бинарный выход. Если $A \neq B$, то на выходе 1, иначе на выходе 0.
		Вещественные входы и бинарный выход. Если $A > B$, то на выходе 1, иначе на выходе 0.
		Вещественные входы и бинарный выход. Если $A < B$, то на выходе 1, иначе на выходе 0.



1.9 Функции работы со временем

Нет, машину времени, несмотря на название этих функций, мы не построим.

Но все системы управления работают с какими-то процессами. А процесс — это функция, аргументом которой является время.

Так что фактически — это функции для измерения времени, счета, суммирования и задержки.

Функции работы со временем		
	ДА	Бинарные вход выход. Фильтр «задержка нуля». При изменении входа с 0 на 1 — немедленно выдает на выход 1. При изменении входа с 1 на 0 — выдает на выход 0 только при прошествии заданного времени. Время задержки задается в десятых долях секунды (0.1сек) или децисекундах.
	ДА	Бинарные вход выход. Фильтр «задержка единицы». При изменении входа с 1 на 0 — немедленно выдает на выход 0. При изменении входа с 0 на 1 — выдает на выход 1 только при прошествии заданного времени. Время задержки задается в десятых долях секунды (0.1сек) или децисекундах.
	ДА	Бинарные вход выход. Фильтр «задержка изменения состояния». При изменении входа с 0(1) на 1(0) — выдает на выход 1(0 только при прошествии заданного времени. Время задержки задается в десятых долях секунды (0.1сек) или децисекундах.
	ДА	Бинарный выход. Таймер. Выдаст на выход попеременно нули и единицы с заданным периодом. Период задается в десятых долях секунды (0.1сек) или децисекундах.
	ДА	Бинарные вход и выход. Управляемый таймер. Выдает на выход попеременно нули и единицы с заданным периодом, если на входе 1. Если на вход подать 0, то перестает считать. Период задается в десятых долях секунды (0.1сек) или децисекундах.
		Бинарные входы и целый выход. Счетчик. Считает количество переходов с 0 на 1 на входе С. Подача 1 на вход R — обнулит счетчик. На выходе Q — количество посчитанных импульсов.
		Бинарные входы С и R, вещественные вход D и выход Q. Сумматор. Суммирует все числа, пришедшие на вход D. Суммирование происходит при изменении с 0 на 1 на входе С.

		<p>Подача 1 на вход R — обнуляет счетчик.</p> <p>На выходе Q — сумма всех чисел, пришедших на вход D, пока R=0.</p>
		<p>Бинарные вход и выход.</p> <p>Формирователь импульса. При изменении входного значения с 0 на 1, на выходе формируется импульс единичный заданной длительностью.</p> <p>Длительность импульса задается в десятых долях секунды (0.1сек) или децисекундах.</p>
		<p>Бинарные вход и выход.</p> <p>Таймаут актуальности значения. При поступлении на вход <i>любого</i> события, на выходе появляется значение 1.</p> <p>Если в течении заданного времени не приходит нового события, то выходное значение сбрасывается в 0.</p> <p>Узел применяется в основном при обработке данных, принятых от других устройств (совместно с узлами UDP и MQTT).</p>


1.10 Функции управления прохождением событий

Узлов работы с данными у нас множество. А как же события? Разумеется, прохождением событий от узла к узлу тоже можно управлять.

Функции управления прохождением событий (Barriers and filters)		
	ДА	Вход и выход — вещественные. Позволяет прохождение событий не чаще, чем раз в заданный интервал времени. (Фильтр частоты событий). При поступлении события на вход, данные со входа копируются на выход. Затем прохождение событий блокируется на заданное время.
		Вход и выход — вещественные. Генерирует событие на выходе при переходе входа С из 0 в 1. Выход всегда равен входу D.
		Входы и выход — вещественные. Позволяет прохождение событий, если A=B. Выход всегда равен входу D.
		Входы и выход — вещественные. Позволяет прохождение событий, если A≠B. Выход всегда равен входу D.
		Входы и выход — вещественные. Позволяет прохождение событий, если A>B. Выход всегда равен входу D.
		Входы и выход — вещественные. Позволяет прохождение событий, если A<B. Выход всегда равен входу D.

1.11 Функции сетевого сервиса

ПОКА НЕ РЕАЛИЗОВАНЫ!

	<p>Бинарный входы и бинарный выход. Пингует заданный адрес.</p> <p>По переходу входа из 0 в 1 — посылает пинг-пакет. По переходу входа из 1 в 0 — считает, что тайм-аут истёк.</p> <p>Выход: 0 — ответа на пинг нет, 1 — ответ на пинг есть.</p>
---	---

Приложение А Вкладка состояния ShIoTiny (Control and status)

Это основная вкладка, на которую вы попадаете при входе на страницу устройства.

Здесь отображается общее состояние устройства: его название, соединения по WiFi и соединение с MQTT-брокером.

Кроме того, отображаются данные включенных в настоящее время периферийных устройств: бинарных входов, реле, АЦП и датчика температуры и влажности.



Рис. 5: Вкладка текущего состояния ShIoTiny (Control and status)

Данные отключенных устройств не отображаются (выделены серым цветом и надписью NoData).

При включении узлов в схему, связанные с ними устройства автоматически включаются. При удалении узлов из схемы, связанные с ними устройства также автоматически отключаются.

Приложение Б Вкладка настройки сети и режимов работы ShIoTiny (Networking)

Вкладка настройки сети и режимов работы ShIoTiny позволяет настроить:

Основной режим работы; работу в режиме точки доступа WiFi; работу в режиме станции WiFi; соединение с MQTT-брокером; обмен широковещательным пакетами по локальной сети.

ShIoTiny networking and MQTT config Control and status EIDraw

Network mode and access

Name (SSID)

ssid

Password

AP Name (SSID)

apssid

AP Password

AP IP address

192.168.4.1

ShIoTiny mode:

Station mode

Scan WiFi

Lock Web in Station mode ☐

MQTT connection to server

MQTT broker (server) url or IP

cloud.mqtt.ru

MQTT broker (server) port

1883

MQTT broker (server) login

login

MQTT broker (server) password

*

MQTT topic prefix

/

Enable SSL (secure sockets layer) ☐

UDP multicast config

UDP multicast group addr

225.0.18.83

UDP multicast port

6666

Device name for UDP multicast

esp_8266_155da4

Save

Cancel

Reset to default

Рис. 6: Вкладка настройки сети и режимов работы ShIoTiny (Networking)

Приложение В Работа с редактором EIDraw

Главная рабочая вкладка — редактор схем-программ.

Заметьте, что переход на другую вкладку из редактора, уничтожает все ваши не сохраненные изменения! Перед тем, как закрыть редактор или перейти на другую вкладку — сохраните изменения на диск или в устройство!

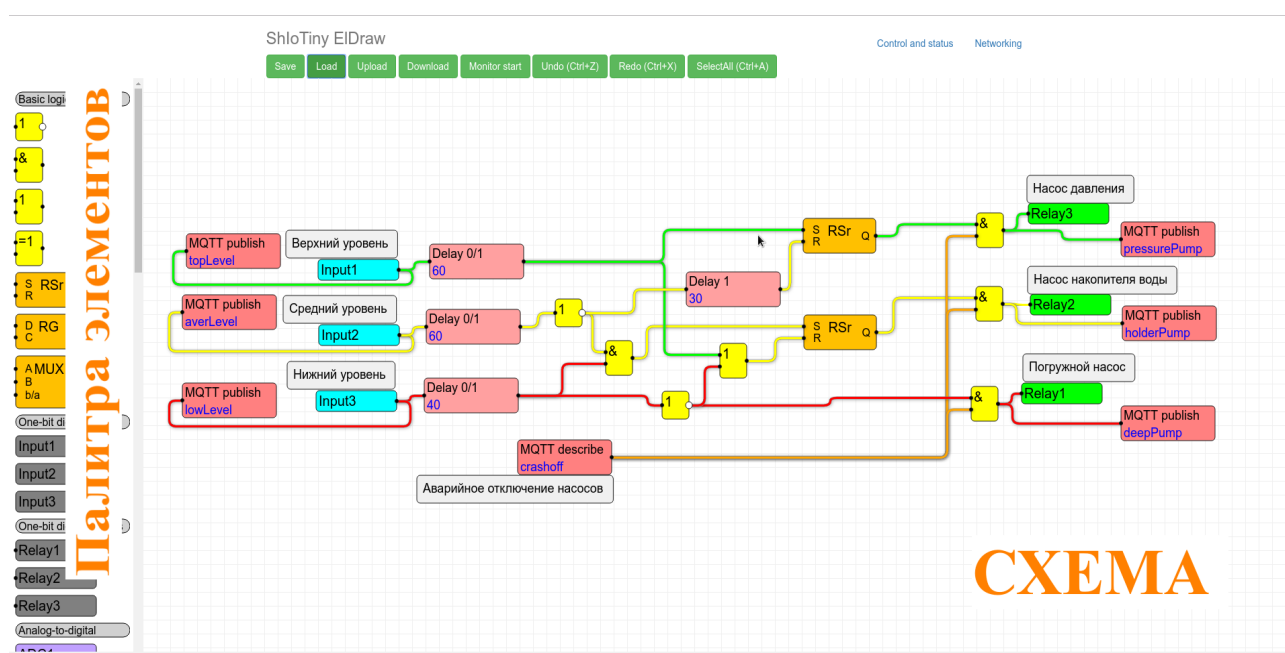


Рис. 7: Редактор схем-программ EIDraw.

Слева — палитра элементов-узлов.

Справа — схема-программа (схема).

Сверху — кнопки управления загрузкой-выгрузкой и редактированием.

Кнопки работы с диском.

Save — сохранить схему на ваш компьютер.

Load — загрузить схему с вашего компьютера в редактор (не на устройство!).

Кнопки работы с устройством ShIoTiny.

Upload — загрузить схему из редактора на устройство.

Чтобы программа была загружена на устройство необходимо, чтобы все входы всех элементов были соединены!

Если какой-либо вход не присоединен, будет выдано предупреждение и схема не загрузится на устройство!

Для удаления схемы из устройства загрузите в устройство пустую схему (не содержащую ни одного элемента-узла).

Download - загрузить схему из устройства в редактор. Вообще схема, если она имеется в устройстве, загружается автоматически. Но если вы что-то наисправляли-поменяли, то загрузка схемы из устройства позволяет все ваши исправления отменить.

Monitor Start/Monitor Stop — включение-выключение режима монитора. Он очень полезен для отладки схем. При включенном мониторе возле каждого входа и выхода показывается его реальное состояние.

Советуем перед каждым запуском монитора нажимать кнопку Download для синхронизации схемы и программы.

Кнопки редактирования.

Undo (CTRL+Z) — отменить действие редактирования.

Redo (CTRL+X) — вернуть действие редактирования.

Select All (CTRL+A) — выделить все, что есть на схеме.

Работа с редактором.

Для того, чтобы вставить узел в схему — наведите курсор мыши на нужный узел в палитре элементов; затем нажмите левую клавишу мыши. После этого, удерживая нажатой левую клавишу мыши перетащите этот узел из палитры элементов на схему.

Для того, чтобы выделить узел или связь наведите курсор мыши на нужный узел или связь в схеме и нажмите левую клавишу мыши.

Для того, чтобы выделить группу узлов и/или связей, наведите курсор мыши на произвольную точку схемы, свободную от узлов и связей и нажмите левую клавишу мыши. Затем, удерживая нажатой левую клавишу мыши,

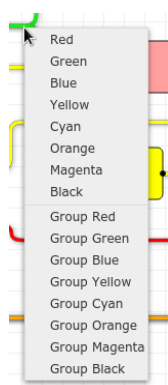
перемещайте курсор мыши, пока не выделите прямоугольную область, включающую нужные вам узлы и связи.

Для того, чтобы выделить несколько произвольных узлов и связей, нажмите клавишу SHIFT. Затем, удерживая клавишу SHIFT, поочередно наведите курсор мыши на каждый из выделяемых элементов и связей и при наведении на каждый из них нажмите однократно левую клавишу мыши.

Для удаления узлов и связей выделите их любым способом и нажмите клавишу DEL.

Для перемещения узлов и связей выделите их любым способом. Затем наведите курсор мыши на любой выделенный узел или связь и нажмите левую клавишу мыши. После этого, удерживая нажатой левую клавишу мыши перемещайте всю выделенную группу узлов и связей, одиночный элемент или связь.

Для того, чтобы раскрасить связи между узлами наведите курсор мыши на нужную связь и нажмите правую клавишу мыши. Появится выпадающее меню, как на рисунке ниже.



Выберете курсором нужный цвет связи или группы связей и нажмите левую клавишу мыши.

Если выбрана одиночная раскраска — изменит цвет только выбранная связь.

Если выбрана раскраска группы связей — изменят цвет все связи, подключенные к тому же выходу, что и выбранная связь.

Таблица цветов приведена ниже.

№	Пункт меню	Цвет	Примечание
1	Red	Красный	Одиночная связь
2	Green	Зеленый	Одиночная связь

3	Blue	Синий	Одиночная связь
4	Yellow	Желтый	Одиночная связь
5	Cyan	Голубой	Одиночная связь
6	Orange	Оранжевый	Одиночная связь
7	Magenta	Сиреневый	Одиночная связь
8	Black	Черный	Одиночная связь
9	Group Red	Красный	Группа связей
10	Group Green	Зеленый	Группа связей
11	Group Blue	Синий	Группа связей
12	Group Yellow	Желтый	Группа связей
13	Group Cyan	Голубой	Группа связей
14	Group Orange	Оранжевый	Группа связей
15	Group Magenta	Сиреневый	Группа связей
16	Group Black	Черный	Группа связей

Раскраска связей служит для удобства пользователей и никак не влияет на работу схемы-программы.

Для того, чтобы переместить сегмент связи, выделите нужную связь. Затем подведите курсор мыши к углу нужной сегмента связи. На углах сегментов линии связи при ее выделении появляются метки, как показано на рисунке ниже.

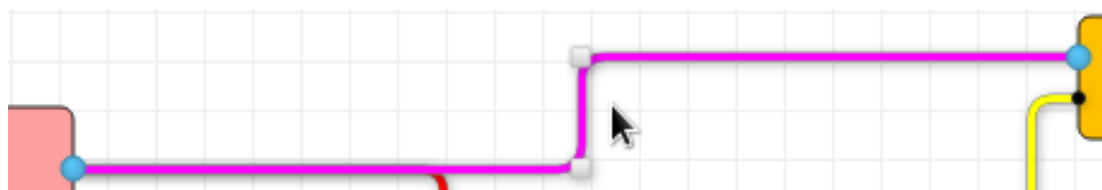


Рис. 8: При выделении линии связи, на углах ее сегментов появляются метки

При наведении на эти метки курсор мыши изменяет форму, что и показано на рисунке ниже.

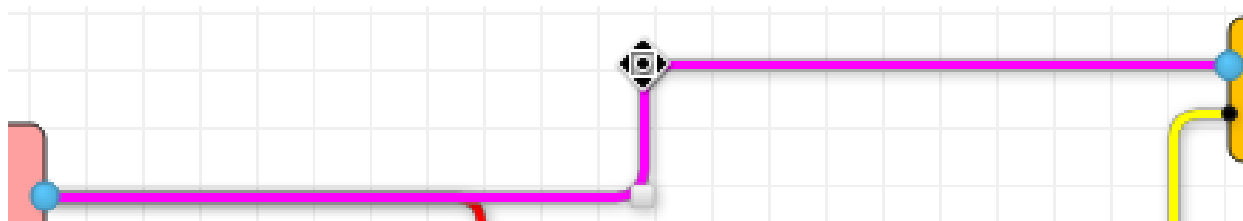


Рис. 9: При наведении на метки выделенной линии связи, курсор мыши изменяет форму

Наведя курсор, на нужный угол сегмента линии связи, нажмите левую клавишу мыши.

После этого, удерживая нажатой левую клавишу мыши перемещайте сегмент линии связи в нужное место.