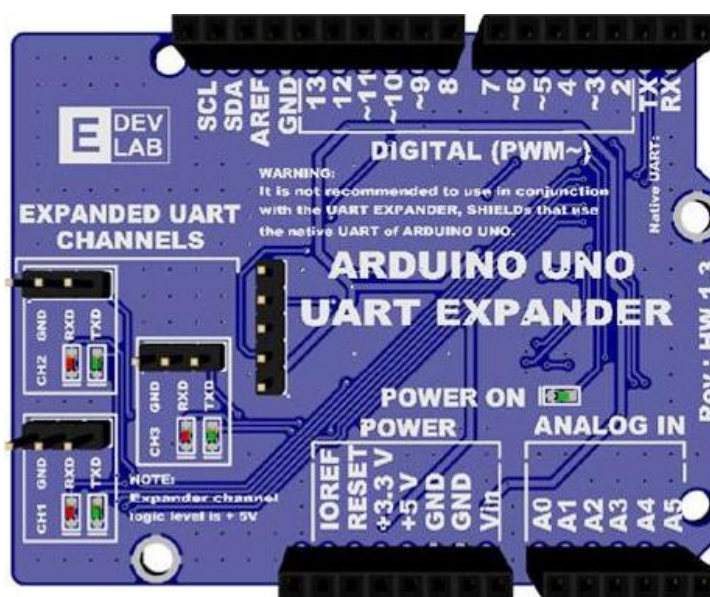


# Расширитель портов UART для ARDUINO UNO



## Руководство пользователя (Краткое)



## Лист регистраций изменений

Дата	Номер	Содержание
17.02.2019	1	Первый выпуск



## Содержание

2. ТЕРМИНЫ И АББРЕВИАТУРЫ .....	3
3. ВВЕДЕНИЕ .....	4
4. ОПИСАНИЕ УСТРОЙСТВА .....	5
4.1. ИНДИКАТОРЫ ПРИЕМА И ПЕРЕДАЧИ ДАННЫХ ПО КАНАЛУ УАПП .....	5
4.2. ШТЫРЬКОВЫЙ РАЗЪЕМ КАНАЛА УАПП .....	6
4.3. ИНДИКАТОР РЕЖИМА РАБОТЫ «Вкл/Выкл» .....	6
4.4. КОНТАКТЫ АППАРАТНОГО СБРОСА НАСТРОЕК .....	6
5. БЫСТРЫЙ СТАРТ .....	8
6. ПРОТОКОЛ ВЗАИМОДЕЙСТВИЯ С ARDUINO UNO .....	9

## 2. Термины и аббревиатуры

UART – См. УАПП;

УАПП – Универсальный асинхронный приемопередатчик;

АЕСР – Arduino Uno UART Expander Communication Protocol  
(Протокол взаимодействия с *расширителем портов* УАПП Arduino Uno);



### 3. Введение

*Расширитель портов UART для Arduino Uno (Arduino UNO UART Expander – далее расширитель портов)* предназначен для увеличения количества универсальных асинхронных приемопередатчиков (УАПП – UART) изделия *Arduino UNO* версии 3e и других совместимых с ней.

#### **Ключевые особенности:**

- Три расширенных канала УАПП;
- Индивидуальные настройки для каждого *расширенного канала* УАПП:
  - Возможность настройки скорости и формата данных;
  - 512 байтный буфер по приему данных;
  - 512 байтный буфер по передаче данных;
  - Световая индикация текущего режима работы;
  - Возможность сохранения текущих настроек в энергонезависимой памяти.

## 4. Описание устройства

На рисунке 1 изображен *расширитель портов*. Расположение и назначение контактов в *штырьковом разъеме ARDUINO UNO* расширителя портов соответствует расположению и назначению контактов в штырьковом разъеме изделия *Arduino UNO* версии 3e и других совместимых с ней.

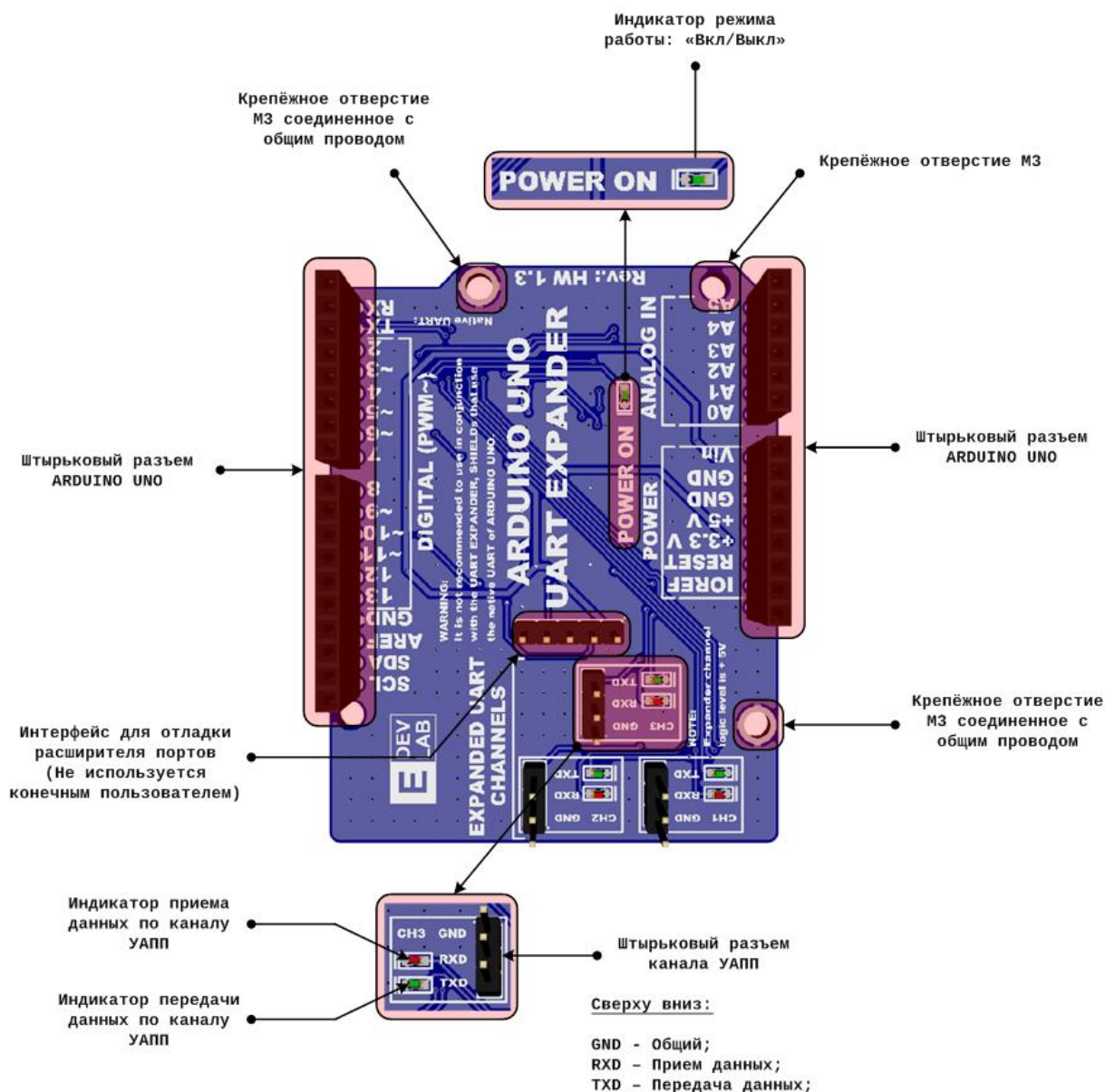


Рис. 1 – Расширитель портов.

### 4.1. Индикаторы приема и передачи данных по каналу УАПП

*Индикаторы приема и передачи данных по каналу УАПП* отображают текущее состояние канала УАПП. Если канал выключен, то оба светодиода – выключены. При отсутствии приема и передачи в канале, зелёный светодиод – включен, а красный – выключен. При



приеме данных, мигает красный светодиод, а при передаче – зелёный. Частота вспышек светодиодов равна 1,7 Гц.

#### 4.2. Штырьковый разъем канала УАПП

*Штырьковый разъем канала УАПП* состоит из трех контактов. Каждый из контактов промаркирован, как показано на рисунке 1. Назначение контактов отражено в маркировке. Каждый канал УАПП работает с логическими уровнями +5 В.

#### 4.3. Индикатор режима работы «Вкл/Выкл»

*Индикатор режима работы «Вкл/Выкл»* отражает текущее состояние расширителя портов. В нормальном режиме работы он светиться ровным зелёным светом. При стирании настроек из энергонезависимой памяти индикатор в течение 3 секунд будет мигать с частотой 3 Гц. Он будет мигать с этой же частотой, если при загрузке *расширитель портов* обнаружит поврежденные настройки в энергонезависимой памяти.

#### 4.4. Контакты аппаратного сброса настроек

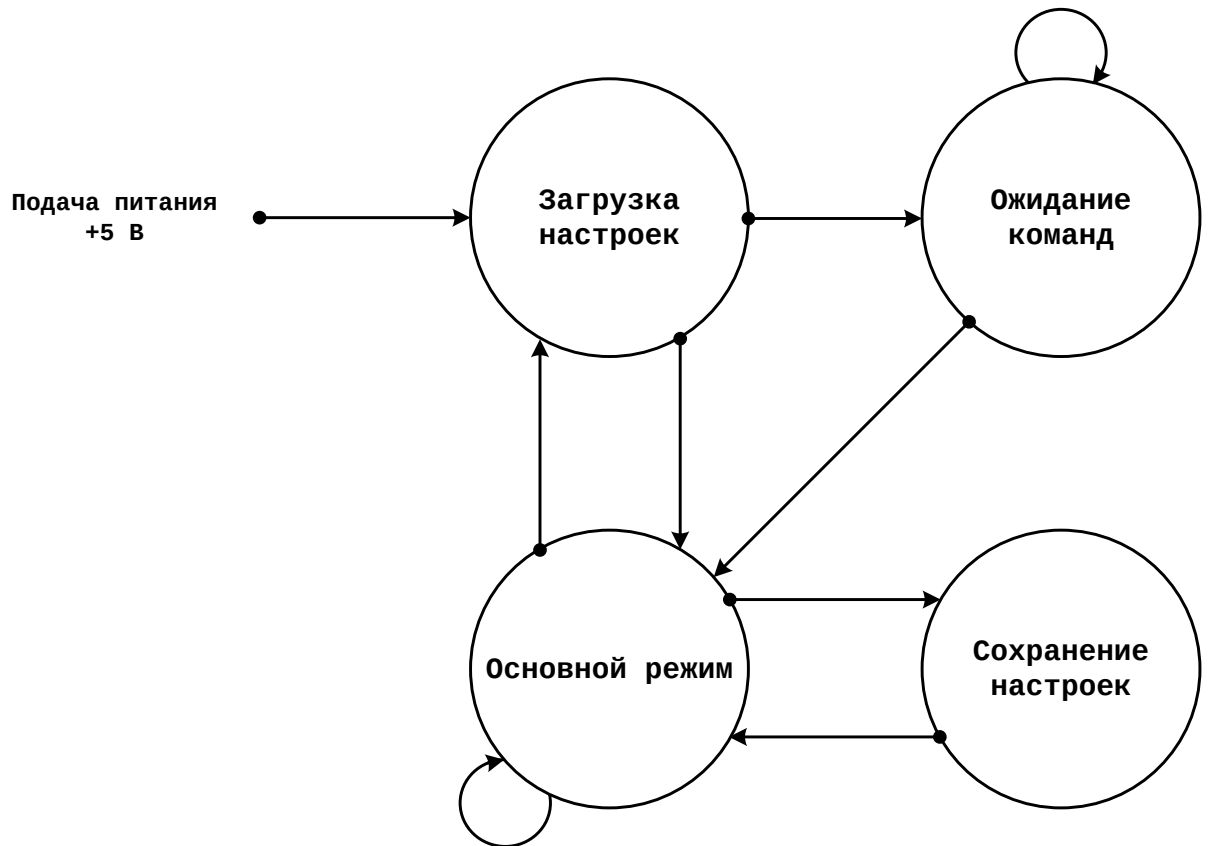
*Контакты аппаратного сброса настроек* предназначены для восстановления настроек по-умолчанию в случае, если пользователем были заданы некорректные значения и перепрограммировать устройство с помощью протокола *AESR* не удастся.

После подачи электропитания устройство включает зеленый светодиод индикатора режима работы и переходит в режим ожидания команды со стандартными настройками (9600, 1 стоп бит без контроля четности). В режиме ожидания устройство находится 200 мс. Если за это время от хоста не поступило команды с новыми настройками УАПП для родного порта Arduino, то устройство проверяет, есть ли в энергонезависимой памяти корректные сохраненные настройки. Если настройки обнаружены, то устройство загружает их. Если устройство обнаруживает не корректные настройки, оно очищает энергонезависимую память и записывает туда стандартные настройки, после чего осуществляет световую индикацию этого, загружает сохраненные настройки и переходит в основной режим работы.

В основном режиме работы при стандартных настройках расширенные каналы УАПП выключены и требуют соответствующей активации пользователем.

Если в устройстве загружены настройки которые пользователь не знает или у него не получается программно сбросить их, то необходимо выполнить аппаратный сброс настроек. Для этого

необходимо включить устройство, подав питание на него. Дождаться когда оно полностью проинициализируется, на это указывает горящий ровным светом зеленый индикатор питания устройства.





## 5. Быстрый старт

**Примечание:** Во избежание получения ложной информации от расширителя портов, включенные каналы устройства не должны «висеть в воздухе».



## 6. Протокол взаимодействия с ARDUINO UNO

Взаимодействие между *расширителем портов* и Arduino UNO осуществляется посредством двухуровневого протокола АЕСР (Arduino uno uart Expander Communication Protocol). Первый его уровень (ATL – АЕСР Transport Layer) является транспортным и отвечает за передачу сообщений между устройствами. Второй (ADL – АЕСР Data Layer) - уровень данных. Отвечает за передачу команд и данных бизнес логики (Рисунок 2).

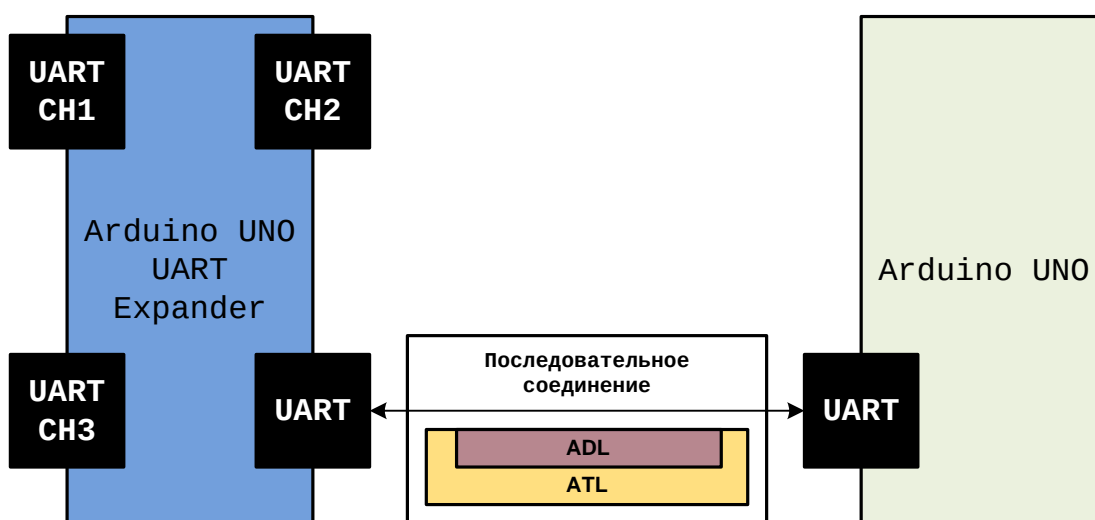


Рис. 2 – Схема двухуровневого протокола АЕСР

### 6.1. Структура кадра ATL

На рисунке 3 изображена структура кадра ATL. Описание полей приведено в таблице 1. В кадре ATL агрегируются данные получаемые *расширителем портов* по каналам CH1 – CH3 и передаются Arduino UNO для дальнейшей обработки. В нём так же, агрегируются данные передаваемые Arduino UNO *расширителю портов* для дальнейшей передачи их в соответствующий канал УАПП. Этот же кадр используется для передачи сообщений с настройками и индикации состояний *расширителя портов*.

**Примечание:** Реализация ATL-уровня доступна на сайте [www.edevlab.ru](http://www.edevlab.ru).

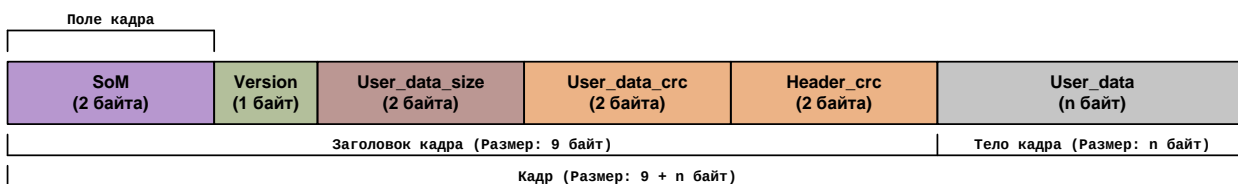


Рис. 3 – Структура кадра ATL

Таблица 1 – Структура кадра ATL

Группа	Смещение (байт)	Поле	Размер (байт)	Описание
Заголовок кадра	0	SoM	2	Маркер начала пакета кадра. Значение: 0xA55A
	2	Version	1	Версия пакета кадра. Значение: 0x01
	3	User_data_size	2	Размер поля «User_data»
	5	User_data_crc	2	Контрольная сумма тела кадра, рассчитанная по стандарту CRC-CCITT16. Значение порождающего полинома 0x1D0F
	7	Header_crc	2	Контрольная сумма заголовка кадра (не включая поле «Header_crc»), рассчитанная по стандарту CRC-CCITT16. Значение порождающего полинома 0x1D0F
Тело кадра	9	User_data	n	Массив пользовательских данных. Максимальным размером 128 байт.

## 6.2. Структура кадра ADL

На рисунке 4 изображена структура кадра ADL. Описание полей приведено в таблице 2. В кадре ADL может передаваться несколько разных типов сообщений (См. таблицу 3).

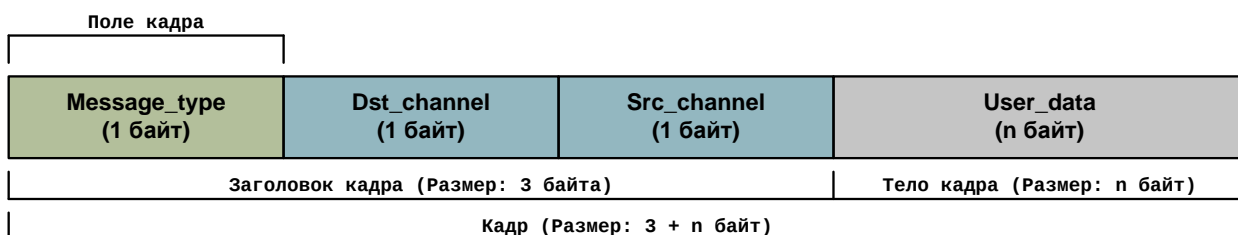


Рис. 4 – Структура кадра ADL



Таблица 2 – Структура кадра ADL

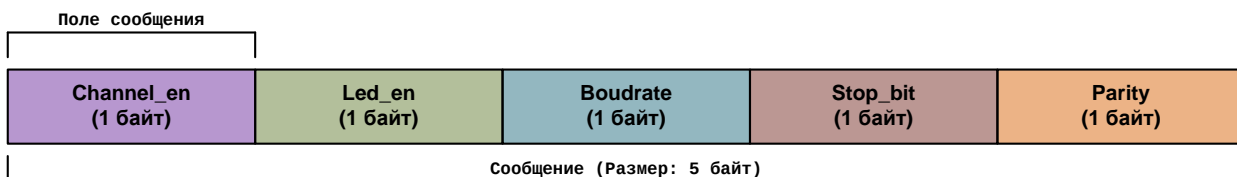
Группа	Смещение (байт)	Поле	Размер (байт)	Описание
Заголовок кадра	0	Message_type	1	Тип сообщения. См. таблицу 3.
	1	Dst_channel	1	Идентификатор получателя пакета. 0x00 – Arduino UNO; 0x01 – Расширитель портов. Канал 1; 0x02 – Расширитель портов. Канал 2; 0x03 – Расширитель портов. Канал 3
	2	Src_channel	1	Идентификатор отправителя пакета. Допустимые значения поля совпадают со значениями для поля «Dst_channel»
Тело кадра	3	User_data	n	Массив пользовательских данных. Максимальным размером 125 байт. Здесь располагаются тела сообщений из таблицы 3.

Таблица 3 – Типы сообщений

№ п/п	Тип сообщения	Значение	Описание
0	SET_SETTINGS	0x01	Тип сообщения, используемый для задания настроек канала расширителя портов.
1	GET_SETTINGS	0x02	Тип сообщения, используемый для чтения настроек канала расширителя портов.
2	DATA	0x03	Тип сообщения, используемый для передачи данных.
3	ACKNOWLEDGE	0x04	Тип сообщения, используемый для передачи подтверждения получения сообщений других типов.
4	INDICATION	0x05	Тип сообщения, используемый для передачи уведомляющих сообщений.

### 6.2.1. Структура сообщений типа «SET\_SETTINGS» и «GET\_SETTINGS»

На рисунке 5 изображена структура сообщений типа «SET\_SETTINGS» и «GET\_SETTINGS». Описание полей сообщений приведено в таблице 4.



**Рис. 5** – Структура сообщений типа «SET\_SETTINGS» и «GET\_SETTINGS»

**Таблица 4** – Структура сообщений типа «SET\_SETTINGS» и «GET\_SETTINGS»

Смещение (байт)	Поле	Размер (байт)	Описание
0	Channel_en	1	Канал включен/выключен: 0x00 – Канал выключен; 0x01 – Канал включен
1	Led_en	1	Индикация включена/выключена: 0x00 – Индикация выключена; 0x01 – Индикация включена
2	Boudrate	1	Скорость передачи данных (бит/с): 0x01 - 600; 0x02 - 1200; 0x03 - 2400; 0x04 - 4800; 0x05 - 9600; 0x06 - 14400; 0x07 - 19200; 0x08 - 28800; 0x09 - 38400; 0x0A - 56000; 0x0B - 57600; 0x0C - 115200
3	Stop_bit	1	Количество стоп-бит: 0x01 – 1; 0x02 – 1,5; 0x03 – 2
4	Parity	1	Чётность: 0x00 – выключена; 0x01 – нечетный; 0x02 – чётный

### 6.2.2. Структура сообщения типа «DATA»

На рисунке 6 изображена структура сообщения типа «DATA». Описание полей сообщений приведено в таблице 5.

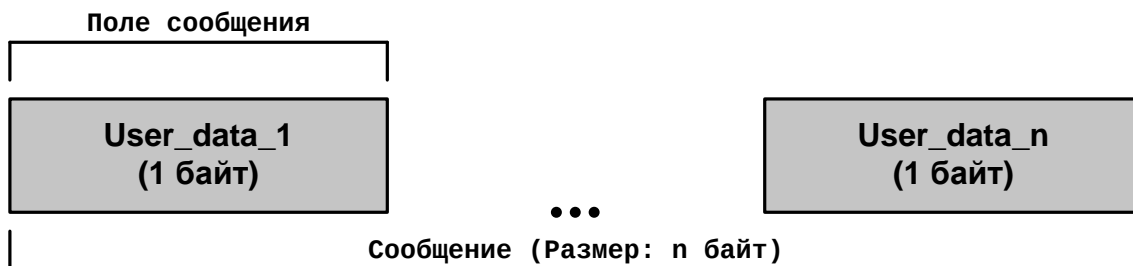


Рис. 6 – Структура сообщения типа «DATA»

Таблица 5 – Структура сообщения типа «DATA»

Смещение (байт)	Поле	Размер (байт)	Описание
0	User_data_1	1	Байт данных пользователя.
...			
n - 1	User_data_n	1	Байт данных пользователя. Где n равно максимум 125.

### 6.2.3. Структура сообщения типа «ACKNOWLEDGE»

На рисунке 7 изображена структура сообщения типа «ACKNOWLEDGE». Описание полей сообщений приведено в таблице 6.

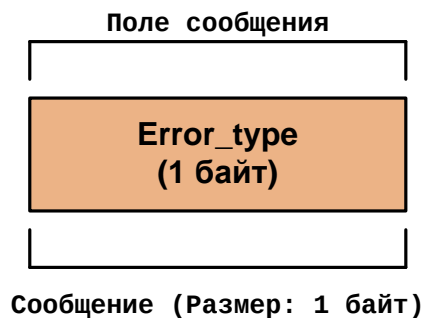


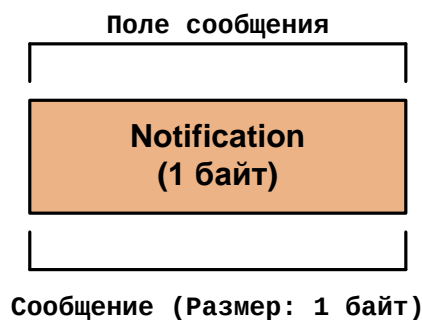
Рис. 7 – Структура сообщения типа «ACKNOWLEDGE»

**Таблица 6** – Структура сообщения типа «ACKNOWLEDGE»

Смещение (байт)	Поле	Размер (байт)	Описание
0	Error_type	1	Типы возвращаемых сообщений: 0x00 – Сообщение принято успешно; 0x01 – Ошибка: Неверный тип сообщения; 0x02 – Ошибка: Неверный идентификатор отправителя или получателя сообщения; 0x03 – Ошибка: Количество передаваемых данных превышает максимально допустимое; 0x04 – Ошибка: Недопустимое значение параметра; 0x05 – Ошибка: Неизвестная ошибка

#### 6.2.4. Структура сообщения типа «INDICATION»

На рисунке 8 изображена структура сообщения типа «INDICATION». Описание полей сообщений приведено в таблице 7.


**Рис. 8** – Структура сообщения типа «INDICATION»

**Таблица 7** – Структура сообщения типа «INDICATION»

Смещение (байт)	Поле	Размер (байт)	Описание
0	Notification	1	Тип возвращаемого сообщения: 0x01 – Произошла ошибка контроля чётности