ПРОТОКОЛ

информационно-технического взаимодействия с датчиками в системе AURA

# Структуры сообщений

Таблица 1 – Общая структура

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Header | [Chunk] | [Chunk] | … | Checksum |

Checksum – контрольная сумма по стандарту CRC-16-IBM (Modbus RTU), 2 байта, вычисляется по всему сообщению

Таблица 2 – Структура заголовка (Header)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Тип | Размер, байт | Переменная | Инфо | Значение |
| 1 | uint | 4 | header | Идентификатор протокола | 0x41525541  ('A', 'U', 'R', 'A') |
| 2 | ushort | 2 | counter | Номер пакета | 0 .. 65535 |
| 3 | uchar | 1 | distance | Количество пересылок,  0 – напрямую | 0 .. 255 |
| 4 | uchar | 1 | flags | Флаги побитно, биты:  0 – требует подтверждения  1 – данные от датчика  2 – *whoami* пакет |  |
| 5 | uint | 4 | source | UID источника пакета |  |
| 6 | uint | 4 | destination | UID адресата пакета |  |
| 7 | uint[4] | 16 | path | до 4 UID ретрансляторов пакета |  |
|  |  | **32** |  | **Общий размер** |  |

Таблица 3 – Структура данных (Chunk)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Тип | Размер, байт | Переменная | Инфо | Значение |
| 1 | ushort | 2 | ID | ID данных (функция) |  |
| 2 | ushort | 2 | type | Тип данных |  |
| 3 | ushort | 2 | payload\_size | Размер полезной нагрузки в байтах | 0 .. 65535 |
| 4 | uchar[] | payload\_size | data | Полезная нагрузка |  |

Таблица 4 – Базовые типы данных:

|  |  |  |
| --- | --- | --- |
| type ID | Тип | Размер, байт (payload\_size) |
| 1 | char | 1 |
| 2 | uchar | 1 |
| 3 | short | 2 |
| 4 | ushort | 2 |
| 5 | int | 4 |
| 6 | uint | 4 |
| 7 | float | 4 |
| 8 | double | 8 |
| 9 | String | N (ascii, нет \0 на конце) |

# Описание взаимодействия

Нумерация ведется каждым устройством, создающим, но не пересылающим, пакет.

Минимальным сообщением является структура заголовка.

Создание пакета устройством:

* **distance** = 0
* указать **source** и **destination**

Для датчиков **destination** не обязателен

Ретрансляция пакета:

* **distance** увеличить на 1
* добавить в **path** свой UID

Получение пакета датчиком:

* проверить **destination**, если совпадает с собственным, то принять

Оповещение датчиками:

Каждое конечное устройство (датчик, ручка, …) регулярно и при включении питания рассылают *whoami* пакеты:

* бит *whoami* в поле flags установить в 1
* указать в **source** свой UID

Это позволит контроллерам быть в курсе подключенных устройств, новых или вышедших из строя.

С помощью *whoami* пакетов каждый контроллер и расширитель строит карту путей, т.е. соответствие адресата набору расширителей. С помощью этой карты можно не заботиться на уровне сервера о расчете пути для создании пакета, он рассылается всем соседям, и каждый расширитель сам поймет, что делать.