



АО «МАССА-К»

Россия, 195277, Санкт-Петербург, Пироговская наб., 15, лит.А www.massa.ru

Протокол «МАССА-К: S4000»

Описание протокола обмена

РУКОВОДСТВО ПРОГРАММИСТА

Редакция 0.2

2026



Оглавление

Введение	3
1. Обмен данными с весами	3
2. Общее описание протокола.....	3
2.1. Указатель на ресурс (URL)	3
2.2. Заголовки (headers)	3
2.3. Тело запроса (body).....	3
2.4. Ответ весов	4
3. Описание команд протокола	5
3.1. Поиск подключенных весов (requestMassaK)	5
3.2. Проверка состояния весов (get_deviceStatus)	5
3.3. Загрузка таблицы базы данных в весы (set_имяТаблицы)	6
3.4. Выгрузка таблицы базы данных из весов (get_имяТаблицы)	6
3.5. Очистка таблицы базы данных в весах (clear_имяТаблицы)	7
4. Структура таблиц базы данных в формате JSON	7
4.1. Таблица товаров (packTable).....	8
4.2. Таблица операторов (operatorTable).....	9
4.3. Таблица отчетов (reportTable).....	10

Введение

Данный документ содержит описание протокола обмена «МАССА-К: S4000» весов S4000 (далее: весов) с внешними устройствами.

Примечание: весы S4000 состоят из терминала S4000 и подключенного к нему модуля взвешивающего.

1. Обмен данными с весами

Обмен данными с весами производится по интерфейсу Wi-Fi. Протокол предусматривает обязательный ответ весов на принятую команду. Отсутствие ответа означает отсутствие связи. Настройка параметров связи осуществляется в меню весов.

Инициатором обмена всегда является ПК (или другое аналогичное устройство).

2. Общее описание протокола

Протокол «МАССА-К: S4000» основан на протоколах передачи данных UDP и HTTP.

Протокол UDP используется для поиска весов, подключенных к сети. Протокол «МАССА-К: S4000» содержит одну UDP-команду, представляющую собой широковещательный (broadcast) запрос, направляемый одновременно всем весам. Принявшие запрос весы отправляют определенный ответ на адрес запрашивающего устройства, таким образом формируется список подключенных весов. Описание UDP-команды представлено в [п.3.1](#).

Остальные команды протокола «МАССА-К: S4000» представляют собой HTTP-запросы, содержащие указатель на ресурс (URL), заголовки, тело запроса (необязательное поле). Для отправки запросов и передачи данных используются методы POST, GET и DELETE. Кодировка всех текстовых данных должна производиться в UTF-8, кроме случаев, указанных отдельно.

2.1. Указатель на ресурс (URL)

HTTP-запрос отправляется по определенному URL (Uniform Resource Locator). Каждый URL состоит из следующих частей:

`http://<IP-адрес>:<порт>/<действие>`

Пример:

`http://192.168.1.81:5001/set_packTable`

Для некоторых запросов после URL могут быть указаны дополнительные параметры. Они отделяются от основного URL-адреса знаком вопроса (?), записываются в формате пар «ключ=значение», и разделяются знаком амперсанда (&). Пример URL с параметрами:

`http://<IP-адрес>:<порт>/get_reportTable?fromDateTime=2025-05-15%2016:28:34&toDateTime=2025-05-16%2012:00:00`

Примечание: знаки пробелов в указании URL должны быть представлены в виде последовательности символов %20.

2.2. Заголовки (headers)

Заголовки позволяют указывать дополнительные параметры HTTP-запросов. Могут включать данные о типе контента, формате принятого ответа и других параметрах, необходимых для обработки запроса. Протокол «МАССА-К: S4000» работает с используемыми всеми HTTP-клиентами стандартными заголовками (например, Content-Type).

2.3. Тело запроса (body)

Протокол «МАССА-К: S4000» при отправке данных по умолчанию поддерживает в качестве тела запроса как отправку прикрепленного файла (Content-Type=multipart/form-data, в поле заголовков указывать не требуется), так и отправку данных в формате JSON непосредственно в теле запроса (в этом случае в поле заголовков необходимо указать Content-Type=application/json).

2.4. Ответ весов

На каждый запрос протокола, весы возвращают код ответа. Общие коды ответов для всех запросов:

- 400 (Bad Request) – HTTP-запрос имеет неправильный формат.
- 404 (Not Found) – указанный URL не поддерживается протоколом «МАССА-К: S4000».
- 405 (Method Not Allowed) – указанный URL поддерживается протоколом «МАССА-К: S4000», но указан неверный метод.
- 500 (Internal Error) – при обработке запроса на стороне весов произошла ошибка.

Специфические для каждой команды ответы указаны в описании соответствующих команд.

Кроме того, если запрашивались данные и команда обработана успешно - весы возвращают данные в описанном формате в теле ответа.

3. Описание команд протокола

3.1. Поиск подключенных весов (requestMassaK)

Широковещательный UDP-запрос:

Представляет собой текстовое сообщение:

```
requestMassaK
```

UDP-ответ от весов:

Представляет собой текстовое сообщение:

```
responseMassaK:<код весов>
```

где <код весов> - код весов в текстовом формате длиной до 10 символов. Пример UDP-ответа:

```
responseMassaK:2808228C01
```

Примечание: для текстовых сообщений в запросе и ответе используется кодировка ASCII.

Примечание 2: если терминал не находится в составе весов, в качестве кода весов в ответе будет указано значение 0.

3.2. Проверка состояния весов (get_deviceStatus)

HTTP-запрос:

```
GET http://<IP-адрес>:<порт>/get_deviceStatus
```

HTTP-ответ от весов:

200	OK	Данные успешно выгружены	Данные в JSON-формате
-----	----	--------------------------	-----------------------

В случае ответа **200 (OK)**, в теле ответа в JSON-формате передаются данные о весах.

Структура данных весов:

№	Параметр	Тип	Описание	Допустимые значения
1	code	Строка	Код весов	До 10 символов

Примечание: если терминал не находится в составе весов, в качестве кода весов в ответе будет указано значение 0.

Пример выгружаемого файла с данными о весах в JSON-формате:

```
{
  "code": "2808228C01",
}
```

3.3. Загрузка таблицы базы данных в весы (set_имяТаблицы)

HTTP-запрос:

POST	http://<IP-адрес>:<порт>/set_имяТаблицы	Данные в JSON-формате
------	---	-----------------------

где *имяТаблицы* соответствует названию таблицы в базе данных весов.

Описание структуры таблиц базы данных см. в [п.4](#).

Пример HTTP-запроса:

POST	http://<IP-адрес>:<порт>/set_packTable	Данные в JSON-формате
------	--	-----------------------

HTTP-ответ от весов:

200	OK	Данные были успешно загружены
400	Bad Request	Неверный формат данных, либо данные содержат неверные значения
404	Not Found	Указанная таблица не поддерживается весами

3.4. Выгрузка таблицы базы данных из весов (get_имяТаблицы)

HTTP-запрос:

GET	http://<IP-адрес>:<порт>/get_имяТаблицы	Параметры запроса (см. Примечание)
-----	---	------------------------------------

где *имяТаблицы* соответствует названию таблицы в базе данных весов.

Описание структуры таблиц базы данных см. в [п.4](#).

Примечание: поле «Параметры запроса» используется для выгрузки таблиц, в которых записи формируются в процессе работы весов с указанием даты/времени для каждой записи; для выгрузки остальных таблиц поле «Параметры запроса» должно быть пустым.

Примеры HTTP-запросов:

GET	http://<IP-адрес>:<порт>/get_packTable
-----	--

GET	http://<IP-адрес>:<порт>/get_reportTable?fromDateTime=2025-05-15%2012:00:00&toDateTime=2025-05-16%2012:00:00
-----	--

HTTP-ответ от весов:

200	OK	Данные успешно выгружены	Данные в JSON-формате
400	Bad Request	Неверный формат данных запроса, либо запрошенная таблица не поддерживается весами.	
404	Not Found	Указанная таблица не поддерживается весами	

В случае ответа **200 (OK)**, в теле ответа передается таблица базы данных в JSON-формате.

Описание структуры таблиц базы данных см. в [п.4](#).

3.5. Очистка таблицы базы данных в весах (clear_имяТаблицы)

HTTP-запрос:

DELETE	http://<IP-адрес>:<порт>/clear_имяТаблицы
--------	---

где *имяТаблицы* соответствует названию таблицы в базе данных весов.

Пример HTTP-запроса:

DELETE	http://<IP-адрес>:<порт>/clear_packTable
--------	--

HTTP-ответ от терминала:

200	OK	Данные были успешно очищены
400	Bad Request	Указанная таблица не поддерживается весами

4. Структура таблиц базы данных в формате JSON

Таблицы базы данных, передаваемые в веса и принимаемые из весов, имеют единую структуру в формате JSON. В начале таблицы указывается ее имя, далее следует набор записей (одна запись соответствует одной строке в таблице базы данных). Каждая запись состоит из нескольких полей с указанием их наименования и значения:

<pre>{ "имяТаблицы": [{ "id": значениеПоляId, "наименованиеПоля1": значениеПоля1, "наименованиеПоля2": значениеПоля2, ... "наименованиеПоляN": значениеПоляN }, ... { "id": значениеПоляId, "наименованиеПоля1": значениеПоля1, "наименованиеПоля2": значениеПоля2, ... "наименованиеПоляN": значениеПоляN }] }</pre>

где:

- *имяТаблицы* соответствует названию таблицы в базе данных весов;
- *id* – имя поля, содержащего идентификатор записи;
- *значениеПоляId* – значение идентификатора записи, число;
- *наименованиеПоля1* .. *наименованиеПоляN* – наименование поля в указанной таблице;
- *значениеПоля1* .. *значениеПоляN* – значение соответствующего поля. Если поле содержит текстовые данные – значение поля заключается в кавычки.

Весы поддерживают следующие таблицы:

- *packTable* – таблица товаров (см. [п.4.1](#));
- *operatorTable* – таблица операторов (см. [п.4.2](#));
- *reportTable* – таблица отчетов (см. [п.4.3](#)).

4.1. Таблица товаров (packTable).

Содержит список товаров с указанием весовых требований к их фасовке.

Поддерживаемые HTTP-запросы:

POST	http://<IP-адрес>:<порт>/set_packTable	Данные в JSON-формате
GET	http://<IP-адрес>:<порт>/get_packTable	
DELETE	http://<IP-адрес>:<порт>/clear_packTable	

Структура записи таблицы товаров:

№	Параметр товара	Тип	Описание	Допустимые значения
1	id	Целое число	Идентификатор записи	От 0 до 2147483647
2	party	Строка	Номер заказа/партии	До 16 символов
3	code	Строка	Код товара	До 16 символов
4	name	Строка	Наименование товара	До 64 символов
5	minGr	Целое число	Минимальный вес в граммах	От 0 до 2147483647
6	maxGr	Целое число	Максимальный вес в граммах	От 0 до 2147483647
7	tareGr	Целое число	Вес предустановленной тары товара в граммах	От 0 до 2147483647

Пример таблицы товаров packTable в JSON-формате:

```
{
  "packTable": [
    {
      "id": 1,
      "party": "23",
      "code": "120",
      "name": "Картофель мытый 1 кг",
      "minGr": 1000,
      "maxGr": 1030,
      "tareGr": 100
    },
    {
      "id": 2,
      "party": "1734",
      "code": "1358",
      "name": "Свекла новый урожай 1 кг",
      "minGr": 1000,
      "maxGr": 1050,
      "tareGr": 120
    }
  ]
}
```

4.2. Таблица операторов (operatorTable).

Содержит список операторов, каждый из которых может зарегистрироваться на весах для проведения фасовки товаров.

Поддерживаемые HTTP-запросы:

POST	http://<IP-адрес>:<порт>/set_operatorTable	Данные в JSON-формате
GET	http://<IP-адрес>:<порт>/get_operatorTable	
DELETE	http://<IP-адрес>:<порт>/clear_operatorTable	

Структура записи таблицы операторов:

№	Параметр	Тип	Описание	Допустимые значения
1	id	Целое число	Идентификатор записи	От 0 до 2147483647
2	code	Строка	Код оператора (табельный номер)	До 16 символов
3	name	Строка	Имя оператора	До 64 символов
4	pin	Строка	PIN-код оператора	До 10 цифр

Пример таблицы операторов operatorTable в JSON-формате:

```
{
  "operatorTable": [
    {
      "id": 1,
      "code": "52",
      "name": "Иванов Иван Иванович",
      "pin": "12345"
    },
    {
      "id": 2,
      "code": "53",
      "name": "Павлов Павел Павлович",
      "pin": "54321"
    }
  ]
}
```

4.3. Таблица отчетов (reportTable)

Содержит список всех операций фасовки товаров с указанием конкретного оператора, товара и требований к его фасовке, зарегистрированного веса товара, даты/времени проведения операции.

Поддерживаемые HTTP-запросы:

GET	http://<IP-адрес>:<порт>/get_reportTable	Параметры запроса (см. Примечание)
DELETE	http://<IP-адрес>:<порт>/clear_reportTable	

Примечание: в качестве параметров запроса может быть указан временной диапазон, за который необходимо получить данные из весов. Временной диапазон описывается следующими ключами запроса:

№	Ключ	Тип	Описание	Допустимые значения
1	fromDateTime	Строка	Дата/время начала диапазона	ГГГГ-ММ-ДД ЧЧ:ММ:СС
2	toDateTime	Строка	Дата/время конца диапазона	ГГГГ-ММ-ДД ЧЧ:ММ:СС

В этом случае HTTP-запрос будет выглядеть следующим образом:

GET	http://<IP-адрес>:<порт>/get_reportTable	?fromDateTime=2025-05-15%2012:00:00&toDateTime=2025-05-16%2012:00:00
-----	--	--

При отсутствии параметра **toDateTime**, данные будут переданы от **fromDateTime** до текущего момента. При отсутствии параметра **fromDateTime**, данные будут переданы от первой существующей записи до **toDateTime**. При отсутствии обоих параметров, либо пустом поле «Параметры запроса», данные будут переданы за все время по всем имеющимся записям.

Структура записи таблицы отчетов:

№	Параметр	Тип	Описание	Допустимые значения
1	id	Целое число	Идентификатор записи	От 1 до 50000
2	number	Целое число	Номер регистрации	Уникальное значение, не ограничено
3	datetime	Строка	Дата/время регистрации	ГГГГ-ММ-ДД ЧЧ:ММ:СС
4	scalesCode	Строка	Код весов	До 10 символов
5	operatorCode	Строка	Код оператора (табельный номер)	До 16 символов
6	operatorName	Строка	Имя оператора	До 64 символов
7	packParty	Строка	Номер заказа/партии	До 16 символов
8	packCode	Строка	Код товара	До 16 символов
9	packName	Строка	Наименование товара	До 64 символов
10	weightGr	Целое число	Вес товара в граммах	От 0 до 2147483647
11	minGr	Целое число	Минимальный вес в граммах	От 0 до 2147483647
12	maxGr	Целое число	Максимальный вес в граммах	От 0 до 2147483647
13	tareGr	Целое число	Вес тары в граммах	От 0 до 2147483647

Пример таблицы отчетов reportTable в JSON-формате:

```
{
  "reportTable": [
    {
      "id": 1,
      "number": 1234,
      "dateTime": "2025-05-15 16:28:34",
      "scalesCode": "2808228C01",
      "operatorCode": "52",
      "operatorName": "Иванов Иван Иванович",
      "packParty": "23",
      "packCode": "120",
      "packName": "Картофель мытый 1 кг",
      "weightGr": 1012,
      "minGr": 1000,
      "maxGr": 1030,
      "tareGr": 100
    },
    {
      "id": 2,
      "number": 1235,
      "dateTime": "2025-05-15 16:30:17",
      "scalesCode": "2808228C01",
      "operatorCode": "52",
      "operatorName": "Иванов Иван Иванович",
      "packParty": "23",
      "packCode": "120",
      "packName": "Картофель мытый 1 кг",
      "weightGr": 1008,
      "minGr": 1000,
      "maxGr": 1030,
      "tareGr": 100
    }
  ]
}
```