



# АО «МАССА-К»

---

Россия, 194044, Санкт-Петербург, Пироговская наб., 15, лит.А [www.massa.ru](http://www.massa.ru)

## «МАССА-К: Драйвер 100» Описание драйвера весов



**РУКОВОДСТВО ПРОГРАММИСТА**

---

## Оглавление

<b>1. Введение .....</b>	<b>3</b>
<b>2. Поддерживаемые модели весов .....</b>	<b>3</b>
<b>3. Описание драйвера .....</b>	<b>3</b>
<b>4. Объекты DLL-библиотеки драйвера .....</b>	<b>3</b>
<b>5. Перечень сокращений и условных обозначений.....</b>	<b>3</b>
5.1. Типы данных реквизитов объекта .....	3
5.2. Доступность свойств объекта .....	3
<b>6. Свойства объекта Scales.....</b>	<b>4</b>
6.1. Connection .....	4
6.2. Weight.....	4
6.3. TareWeight .....	4
6.4. Division.....	4
6.5. Stable .....	4
6.6. NetIndication .....	4
6.7. ZeroIndication.....	4
6.8. ScalesID .....	5
6.9. Name.....	5
6.10. IpAddress .....	5
6.11. Mask.....	5
6.12. PortEthernet .....	5
6.13. Gateway .....	5
6.14. IpAddressWiFi.....	5
6.15. MaskWiFi .....	6
6.16. PortWiFi.....	6
6.17. GatewayWiFi .....	6
6.18. SSID .....	6
6.19. Key.....	6
6.20. ApIpAddressWiFi .....	6
6.21. ApSSID .....	6
6.22. ApKey.....	7
<b>7. Методы объекта Scales .....</b>	<b>7</b>
7.1. OpenConnection .....	7
7.2. CloseConnection.....	7
7.3. ReadWeight .....	7
7.4. SetTare .....	7
7.5. GetTare .....	7
7.6. SetZero.....	7
7.7. GetName .....	8
7.8. GetEthernet .....	8
7.9. GetWiFi .....	8
7.10. GetWiFiAP .....	8
7.11. SetName.....	8
7.12. SetEthernet.....	8
7.13. SetWiFi .....	9
7.14. SetWiFiAP .....	9
<b>8. Результат выполнения методов (коды ошибок). .....</b>	<b>9</b>
<b>9. Примеры работы с драйвером.....</b>	<b>10</b>
9.1. Примеры для Delphi.....	10

## 1. Введение

Данный документ является описанием драйвера электронных весов (Драйвер 100) производства «МАССА-К».

## 2. Поддерживаемые модели весов

Драйвер позволяет производить обмен данными со всеми моделями весов, поддерживающими [протокол 100](#).

Обмен осуществляется по интерфейсам RS-232, USB, Ethernet и Wi-Fi.

## 3. Описание драйвера

Драйвер включает в себя DLL-библиотеку, содержащую COM-объект с набором свойств и методов, и визуальную утилиту настройки параметров связи в весах. Драйвер может быть подключен к любым системам и средам программирования, поддерживающим технологии ActiveX и OLE-Automation (Delphi, Microsoft Visual Studio, 1С: Предприятие и т.п.).

Драйвер позволяет:

- удаленно получать результаты взвешивания со всех поддерживаемых моделей весов;
- удаленно производить настройку параметров связи и установку тары и >0< на весах.

Перед использованием драйвера его необходимо установить. Для этого, запустить файл программы установки драйвера "[Driver100Setup vX.X.X.X.exe](#)" и следовать инструкциям, появляющимся на экране.

Одновременно с драйвером устанавливается визуальная утилита "Настройка весов 100", позволяющая устанавливать параметры связи весов непосредственно с компьютера. Описание работы утилиты содержится в ее справке.

## 4. Объекты DLL-библиотеки драйвера

DLL-библиотека драйвера содержит объект Scales, который используется для получения массы с весов, установки тары и >0<, настройки и получения параметров связи.

Описание свойств и методов объекта, с помощью которых производится взаимодействие программы пользователя с весами, приведено ниже.

## 5. Перечень сокращений и условных обозначений

### 5.1. Типы данных реквизитов объекта

Обозначение	Описание
LONG	Целое 32-битное число со знаком
BSTR	Текстовая строка

### 5.2. Доступность свойств объекта

Обозначение	Описание
R	Свойство доступно только для чтения
RW	Свойство доступно для чтения и записи

## 6. Свойства объекта Scales

### 6.1. Connection

**Тип:** BSTR

**Доступ:** RW

**Описание:** задает тип соединения (номер COM-порта или IP-адрес:порт).

**Пример:**

```
Scales.Connection := 'COM1';  
Scales.Connection := '192.168.1.245:5001';
```

### 6.2. Weight

**Тип:** LONG

**Доступ:** R

**Описание:** масса, считанная с весов, в делениях со знаком.

**Пример:**

```
i := Scales.Weight;
```

### 6.3. TareWeight

**Тип:** LONG

**Доступ:** RW

**Описание:** масса тары, устанавливаемая на весах или полученная с весов, в граммах.

**Пример:**

```
Scales.TareWeight := 1000;
```

### 6.4. Division

**Тип:** LONG

**Доступ:** R

**Описание:** цена деления в значении массы. Принимаемые значения:

0 – 100 мг,

1 – 1 г,

2 – 10 г,

3 – 100 г,

4 – 1 кг.

**Пример:**

```
i := Scales.Division;
```

### 6.5. Stable

**Тип:** LONG

**Доступ:** R

**Описание:** признак стабилизации массы. Принимаемые значения:

0 – нестабильна,

1 – стабильна.

**Пример:**

```
i := Scales.Stable;
```

### 6.6. NetIndication

**Тип:** LONG

**Доступ:** R

**Описание:** признак индикации <NET>. Принимаемые значения:

0 – нет индикации,

1 – есть индикация.

**Пример:**

```
i := Scales.NetIndication;
```

### 6.7. ZeroIndication

**Тип:** LONG

**Доступ:** R

**Описание:** признак индикации >0<. Принимаемые значения:

0 – нет индикации,

1 – есть индикация.

**Пример:**

```
i := Scales.ZeroIndication;
```

## 6.8. ScalesID

**Тип:** BSTR

**Доступ:** R

**Описание:** идентификатор (номер) весов. Устанавливается на заводе.

**Пример:**

```
id := Scales.ScalesID;
```

## 6.9. Name

**Тип:** BSTR

**Доступ:** RW

**Описание:** имя весов, задается пользователем.

**Пример:**

```
Scales.Name := 'Фасовка';
```

## 6.10. IpAddress

**Тип:** BSTR

**Доступ:** RW

**Описание:** IP-адрес весов в сети Ethernet. Задается пользователем. Для использования динамической адресации задается равным 0.

**Пример:**

```
Scales.IpAddress := '192.168.1.245';
```

## 6.11. Mask

**Тип:** BSTR

**Доступ:** RW

**Описание:** маска подсети Ethernet. Задается пользователем. Для использования динамической адресации задается равной 0.

**Пример:**

```
Scales.Mask := '255.255.255.0';
```

## 6.12. PortEthernet

**Тип:** LONG

**Доступ:** RW

**Описание:** номер TCP-порта для подключения к весам в сети Ethernet. Задается пользователем.

**Пример:**

```
Scales.PortEthernet := 5001;
```

## 6.13. Gateway

**Тип:** BSTR

**Доступ:** RW

**Описание:** IP-адрес межсетевого шлюза сети Ethernet. Задается пользователем. Для использования динамической адресации задается равным 0.

**Пример:**

```
Scales.Gateway := '192.168.1.2';
```

## 6.14. IpAddressWiFi

**Тип:** BSTR

**Доступ:** RW

**Описание:** IP-адрес весов в сети Wi-Fi. Задается пользователем. Для использования динамической адресации задается равным 0.

**Пример:**

```
Scales.IpAddressWiFi := '192.168.1.245';
```

#### 6.15. MaskWiFi

**Тип:** BSTR

**Доступ:** RW

**Описание:** маска подсети Wi-Fi. Задается пользователем. Для использования динамической адресации задается равной 0.

**Пример:**

```
Scales.MaskWiFi := '255.255.255.0';
```

#### 6.16. PortWiFi

**Тип:** LONG

**Доступ:** RW

**Описание:** номер TCP-порта для подключения к весам в сети Wi-Fi. Задается пользователем.

**Пример:**

```
Scales.PortWiFi := 5001;
```

#### 6.17. GatewayWiFi

**Тип:** BSTR

**Доступ:** RW

**Описание:** IP-адрес межсетевого шлюза сети Wi-Fi. Задается пользователем. Для использования динамической адресации задается равным 0.

**Пример:**

```
Scales.GatewayWiFi := '192.168.1.2';
```

#### 6.18. SSID

**Тип:** BSTR

**Доступ:** RW

**Описание:** идентификатор сети Wi-Fi. Задается пользователем.

**Пример:**

```
Scales.SSID := 'SampleWifiNetwork';
```

#### 6.19. Key

**Тип:** BSTR

**Доступ:** RW

**Описание:** пароль доступа к сети Wi-Fi. Задается пользователем.

**Пример:**

```
Scales.Key := 'Password12345678';
```

#### 6.20. ApIpAddressWiFi

**Тип:** BSTR

**Доступ:** RW

**Описание:** IP-адрес весов в режиме точки доступа в сети Wi-Fi. Задается пользователем. Для отключения режима точки доступа задается равным пустой строке.

**Пример:**

```
Scales.ApIpAddressWiFi := '192.168.1.113';
```

#### 6.21. ApSSID

**Тип:** BSTR

**Доступ:** R

**Описание:** идентификатор сети Wi-Fi весов в режиме точки доступа. Устанавливается на заводе. Имеет значение "Mk\_хххххх", где хххххх - идентификационный номер весов.

**Пример:**

```
ScalesAccessPointSSID := Scales.ApSSID;
```

## 6.22. ApKey

**Тип:** BSTR

**Доступ:** R

**Описание:** пароль доступа к сети Wi-Fi весов в режиме точки доступа. Устанавливается на заводе. Имеет значение "MassaK2017".

**Пример:**

```
ScalesAccessPointKey := Scales.ApKey;
```

## 7. Методы объекта Scales

### 7.1. OpenConnection

**Описание:** подключиться к весам.

**Возвращаемое значение:** код ошибки, см. Табл. 8.1.

**Используемые свойства:** Connection.

**Модифицируемые свойства:** –.

**Пример:**

```
Scales.Connection := '192.168.1.245:5001';  
res := Scales.OpenConnection;
```

### 7.2. CloseConnection

**Описание:** отключиться от весов.

**Возвращаемое значение:** код ошибки, см. Табл. 8.1.

**Используемые свойства:** –.

**Модифицируемые свойства:** –.

**Пример:**

```
res := Scales.CloseConnection;
```

### 7.3. ReadWeight

**Описание:** получить массу с весов.

**Возвращаемое значение:** код ошибки, см. Табл. 8.1.

**Используемые свойства:** –.

**Модифицируемые свойства:** Weight, Division, Stable, NetIndication, ZeroIndication.

**Пример:**

```
res := Scales.ReadWeight;
```

### 7.4. SetTare

**Описание:** установить тару на весах.

**Возвращаемое значение:** код ошибки, см. Табл. 8.1.

**Используемые свойства:** TareWeight.

**Модифицируемые свойства:** –.

**Примечание:** если TareWeight = 0, в весах производится тарирование текущим весом.

**Пример:**

```
Scales.TareWeight := 100;  
res := Scales.SetTare;
```

### 7.5. GetTare

**Описание:** получить тару, установленную на весах.

**Возвращаемое значение:** код ошибки, см. Табл. 8.1.

**Используемые свойства:** –.

**Модифицируемые свойства:** TareWeight.

**Пример:**

```
res := Scales.GetTare;
```

### 7.6. SetZero

**Описание:** установить >0< на весах.

**Возвращаемое значение:** код ошибки, см. Табл. 8.1.

**Используемые свойства:** –.

**Модифицируемые свойства:** –.

**Примечание:** в ряде весовых устройств команда не поддерживается.

**Пример:**

```
res := Scales.SetZero;
```

### 7.7. GetName

**Описание:** получить имя весов.

**Возвращаемое значение:** код ошибки, см. Табл. 8.1.

**Используемые свойства:** –.

**Модифицируемые свойства:** ScalesID, Name.

**Пример:**

```
res := Scales.GetName;
```

### 7.8. GetEthernet

**Описание:** получить из весов параметры связи сети Ethernet.

**Возвращаемое значение:** код ошибки, см. Табл. 8.1.

**Используемые свойства:** –.

**Модифицируемые свойства:** ScalesID, IPAddress, Mask, PortEthernet, Gateway.

**Пример:**

```
res := Scales.GetEthernet;
```

### 7.9. GetWiFi

**Описание:** получить параметры связи весов в сети Wi-Fi.

**Возвращаемое значение:** код ошибки, см. Табл. 8.1.

**Используемые свойства:** –.

**Модифицируемые свойства:** ScalesID, IPAddressWiFi, MaskWiFi, PortWiFi, GatewayWifi, SSID, Key.

**Пример:**

```
res := Scales.GetWiFi;
```

### 7.10. GetWiFiAP

**Описание:** получить параметры связи весов как точки доступа в сети Wi-Fi.

**Возвращаемое значение:** код ошибки, см. Табл. 8.1.

**Используемые свойства:** –.

**Модифицируемые свойства:** ScalesID, ApIPAddressWiFi.

**Пример:**

```
res := Scales.GetWiFiAP;
```

### 7.11. SetName

**Описание:** записать имя в весы.

**Возвращаемое значение:** код ошибки, см. Табл. 8.1.

**Используемые свойства:** Name.

**Модифицируемые свойства:** –.

**Пример:**

```
Scales.Name := 'Scales MK-A';  
res := Scales.SetName;
```

### 7.12. SetEthernet

**Описание:** записать в весы параметры связи сети Ethernet.

**Возвращаемое значение:** код ошибки, см. Табл. 8.1.

**Используемые свойства:** IPAddress, Mask, PortEthernet, Gateway.

**Модифицируемые свойства:** –.

**Пример:**

```
Scales.IPAddress := '192.168.1.245';  
Scales.Mask := '255.255.255.0';  
Scales.PortEthernet := 5001;  
Scales.Gateway := '192.168.1.2';
```

```
res := Scales.SetEthernet;
```

### 7.13. SetWiFi

**Описание:** записать параметры связи в весы.

**Возвращаемое значение:** код ошибки, см. Табл. 8.1.

**Используемые свойства:** IpAddressWiFi, MaskWiFi, PortWiFi, GatewayWifi, SSID, Key.

**Модифицируемые свойства:** –.

**Пример:**

```
Scales.IpAddressWiFi := '192.168.1.245';  
Scales.MaskWiFi := '255.255.255.0';  
Scales.PortWiFi := 5001;  
Scales.GatewayWiFi := '192.168.1.2';  
Scales.SSID := 'SampleWifiNetwork';  
Scales.Key := 'Password12345678';  
res := Scales.SetWiFi;
```

### 7.14. SetWiFiAP

**Описание:** записать параметры связи в весы для работы в режиме точки доступа.

**Возвращаемое значение:** код ошибки, см. Табл. 8.1.

**Используемые свойства:** ApIpAddressWiFi.

**Модифицируемые свойства:** –.

**Пример:**

```
Scales.ApIpAddressWiFi := '192.168.1.113';  
res := Scales.SetWiFiAP;
```

## 8. Результат выполнения методов (коды ошибок).

Все методы возвращают код ошибки, список возможных значений представлен в Табл. 8.1.

Табл. 8.1

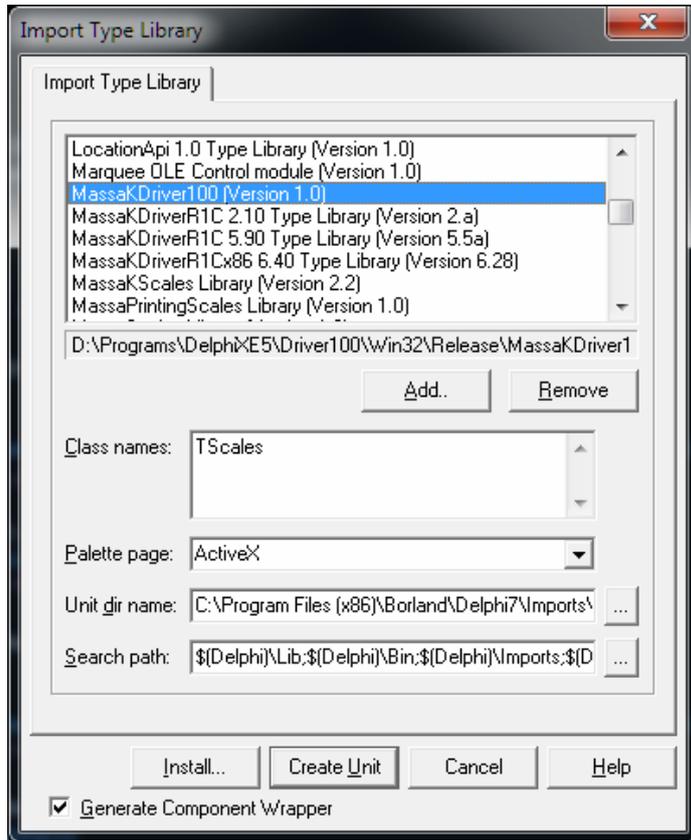
Код ошибки	Описание
0	Ошибок нет
1	Связь с весами не установлена
2	Ошибка обмена данными с весами
3	Весы не готовы к передаче данных
4	Параметр не поддерживается весами
5	Установка параметра невозможна
7	Невозможно выполнить команду / Команда не поддерживается
8	Нагрузка на весовом устройстве превышает НПВ
9	Весовое устройство не в режиме взвешивания
10	Ошибка входных данных
11	Ошибка сохранения данных
16	Интерфейс WiFi не поддерживается
17	Интерфейс Ethernet не поддерживается
21	Установка >0< невозможна из-за наличия нагрузки на платформе
23	Нет связи с модулем взвешивающим
24	Установлена нагрузка на платформу при включении весового устройства
25	Весовое устройство неисправно

## 9. Примеры работы с драйвером

### 9.1. Примеры для Delphi

#### 9.1.1. Подключение библиотеки типов к проекту

1. Создайте новый проект.
2. В главном меню выберите пункт Project \ Import Type Library \
3. В появившемся окне выберите в списке пункт MassaKDriver100 (Version X.X) и нажмите кнопку Create Unit (см. рисунок). Остальные настройки следует оставить по умолчанию.



4. В результате будет сформирован модуль MassaKDriver100\_TLB и автоматически подключен к проекту.
5. В главном модуле программы добавьте в секцию **implementation** следующие строки:

```
uses MassaKDriver100_TLB;  
var Scales: TScales;
```

6. Объект готов к использованию.

#### 9.1.2. Создание экземпляра объекта Scales

Перед началом работы необходимо создать экземпляр COM-объекта, посредством которого будет осуществляться обмен с весами с помощью его методов и свойств. Для этого необходимо вписать в текст программы строку:

```
Scales := TScales.Create(nil);
```

### 9.1.3. Подключение к весам через USB/COM-порт

В качестве параметра подключения необходимо указать требуемый COM-порт, в примере используется порт COM1. Пример:

```
var res: integer;
begin
    // .....
    Scales.Connection:='COM1';
    res:=Scales.OpenConnection;
    if res=0 then begin
        // подключение произведено успешно
    end else begin
        // ошибка подключения
    end;
    // .....
end;
```

### 9.1.4. Подключение к весам через Wi-fi/Ethernet (LAN)

В качестве параметра подключения необходимо указать IP-адрес, в примере используется адрес 192.168.1.245 и порт 5001. Пример:

```
var res: integer;
begin
    // .....
    Scales.Connection:='192.168.1.245:5001';
    res:=Scales.OpenConnection;
    if res=0 then begin
        // подключение произведено успешно
    end else begin
        // ошибка подключения
    end;
    // .....
end;
```

### 9.1.5. Отключение от весов

Пример:

```
var res: integer;
begin
    // .....
    res:=Scales.CloseConnection;
    if res=0 then begin
        // отключение произведено успешно
    end else begin
        // ошибка отключения
    end;
    // .....
end;
```

### 9.1.6. Функция получения массы с весов

Перед получением массы с весов необходимо последовательно выполнить п.п. 9.1.2 и 9.1.3 (или 9.1.4, в зависимости от типа подключения весов). Пример функции получения массы:

```
function GetWeight: string;
var res: integer;
    s: string;
begin
    Result:='0.000';
    res:=Scales.ReadWeight;
    if res=0 then begin
        if Scales.Stable=1 then begin
            // если полученная масса стабильна
            s:=IntToStrDef(Scales.Weight, 0);
            try
                Insert(s, Result, Length(s)-(4-Scales.Division));
            except
                // обработка исключения
            end;
        end else begin
            // если полученная масса нестабильна
        end;
    end else begin
        // если выполнение функции завершено с ошибкой
    end;
end;
```