



АВТОГРАФ серия X. ПРИМЕНЕНИЕ СКРИПТОВ И ОПИСАНИЕ ФУНКЦИЙ

РУКОВОДСТВО ПО ПРИМЕНЕНИЮ

ВЕРСИЯ
2.0



ОГЛАВЛЕНИЕ

Введение	3
Подготовка окружения	4
Компиляция и загрузка микропрограммы	5
Компиляция микропрограмм T.Скрипт через Visual Studio Code	8
Выполнение микропрограмм T.Скрипт в терминале АвтоГРАФ	10
Документация	10
Функции для работы с CAN-шиной	11
Функции для работы с текстовыми командами	16
Функции для работы с событиями	22
Функции для работы со входами	26
Различные встроенные функции	32
Функции для работы с навигационными приемниками	43
Функции для работы с параметрами устройств АвтоГРАФ	46
Функции для работы с последовательными портами	70
Функции для работы с сокетом	76
Функции для работы с файлами в SPI памяти	81
Функции для работы с файлами в SD/RAM памяти	91
Функции для работы с BLE	101
Стандартные функции Rawp	110
Стандартные функции Rawp для чисел с фиксированной запятой	124
Стандартные функции Rawp для чисел с плавающей запятой	134
Стандартные функции Rawp для строк	151
Стандартные функции Rawp для времени и таймера	169
Параметры устройства	179

Введение

В терминалах АвтоГРАФ-GX реализована технология исполнения микропрограмм, написанных пользователем, — Т.Скрипт:

- в АвтоГРАФ-LX — начиная с серийного номера **2909800**;
- в АвтоГРАФ-LX (E) — начиная с серийного номера **2823000**;
- в АвтоГРАФ-SX — начиная с серийного номера **2478000**;
- в АвтоГРАФ-GX — начиная с серийного номера **3115000**;
- в АвтоГРАФ-GX Wi-Fi — начиная с серийного номера **3029000**;
- в АвтоГРАФ-GX АКБ — начиная с серийного номера **3996500**;
- в АвтоГРАФ-GX Wi-Fi АКБ — начиная с серийного номера **3990200**.

В качестве языка программирования используется компилируемый язык Pawn (<https://www.compuphase.com/pawn/pawn.htm>). Написание исходного кода и компиляция выполняются в среде разработки Pawn IDE.

Использование микропрограмм Т.Скрипт позволяет:

- получить доступ к интерфейсам терминала АвтоГРАФ;
- формировать собственные телеметрические данные;
- управлять формированием стандартных телеметрических данных терминала;
- получить доступ к параметрам терминала (например, CAN параметры, навигационные параметры, дискретные параметры, флаги устройства и прочее);
- выполнять команды удаленной настройки (SMS- и серверные команды) непосредственно из микропрограммы.

Выполнение микропрограмм Т.Скрипт защищено от несанкционированного доступа к настройкам терминала: при установленном уровне защиты 1 скрипты терминала недоступны для изменения, а при установленном уровне защиты 2 — недоступны для считывания. Также выполнение микропрограмм Т.Скрипт защищено от несанкционированного доступа к памяти терминала. Это позволяет сохранить надежность работы основной микропрограммы терминала на высоком уровне. Подробнее о выполнении микропрограмм см. раздел [«Выполнение микропрограмм Т.Скрипт в терминале АвтоГРАФ»](#).

Подготовка окружения

Для создания и компиляции микропрограмм Т.Скрипт нужно подготовить окружение:

1. Загрузите архив с файлами скриптов *.inc* по ссылке https://i.tk-chel.ru/products/T_SCRIPT/t.script.zip и распакуйте его содержимое в любую директорию. Архив также содержит примеры скриптов, разработанных специалистами компании «ТехноКом».
2. Загрузите установочный файл для Pawn IDE по ссылке https://i.tk-chel.ru/products/T_SCRIPT/pawn-4.1.7152.exe.
3. Запустите загруженный файл *pawn-4.1.7152.exe* и выполните установку в любую директорию.
4. Запустите Pawn IDE (Quincy) через меню «Пуск» (каталог «Pawn»).
5. Выберите в главном меню Pawn IDE пункт «Tools», далее — пункт «Options...» (**Рис.1**) и укажите в поле «Include path» путь к файлам скриптов *.inc* от ТехноКом (**Рис.2**).

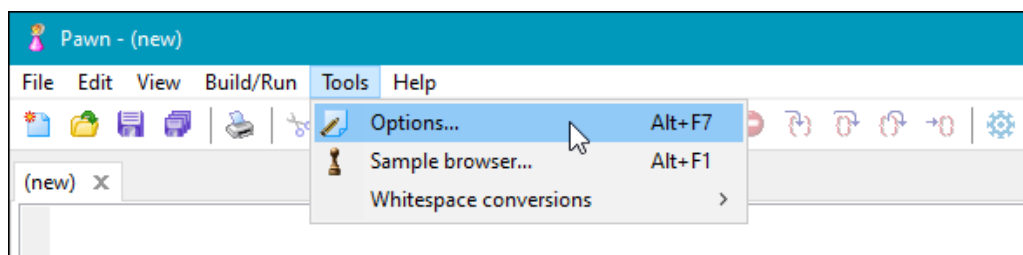


Рис.1. Переход к настройкам Pawn IDE

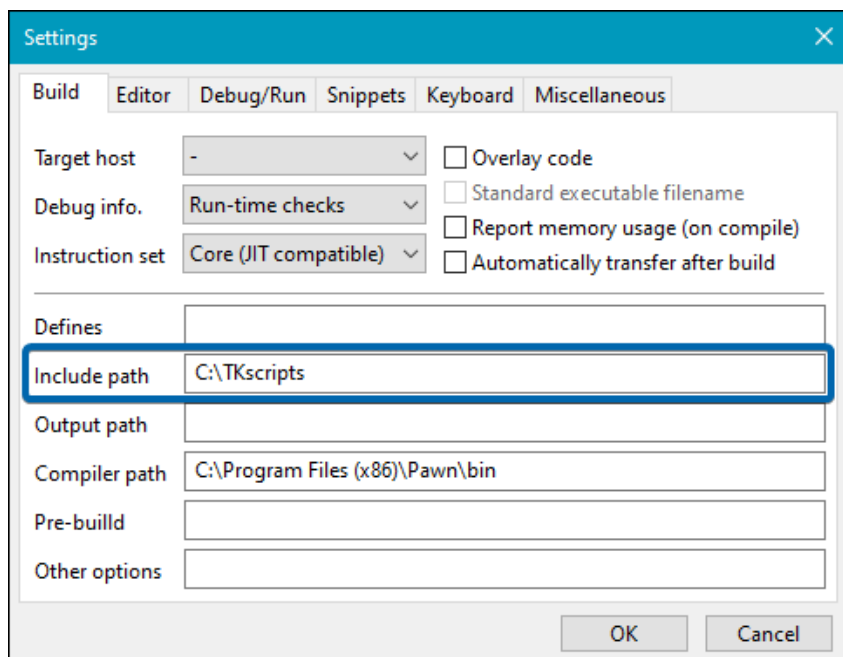


Рис.2. Указание пути к файлам скриптов от ТехноКом

Компиляция и загрузка микропрограммы

Пример использования возможностей микропрограмм Т.Скрипт, содержащийся в файле *tkExample.p*, демонстрирует следующие возможности:

- периодическую выдачу в отладку сообщений в debug интерфейс CDC и файлов логов терминала;
- формирование и сохранение пользовательской телеметрической информации;
- работу с интерфейсом RS232 в виде реализации функции ЭХО.

Для **компиляции и загрузки** микропрограммы в терминал выполните следующий порядок действий:

1. Запустите Pawn IDE.
2. Откройте файл с исходным текстом скрипта *tkExample.p* (Рис.3).

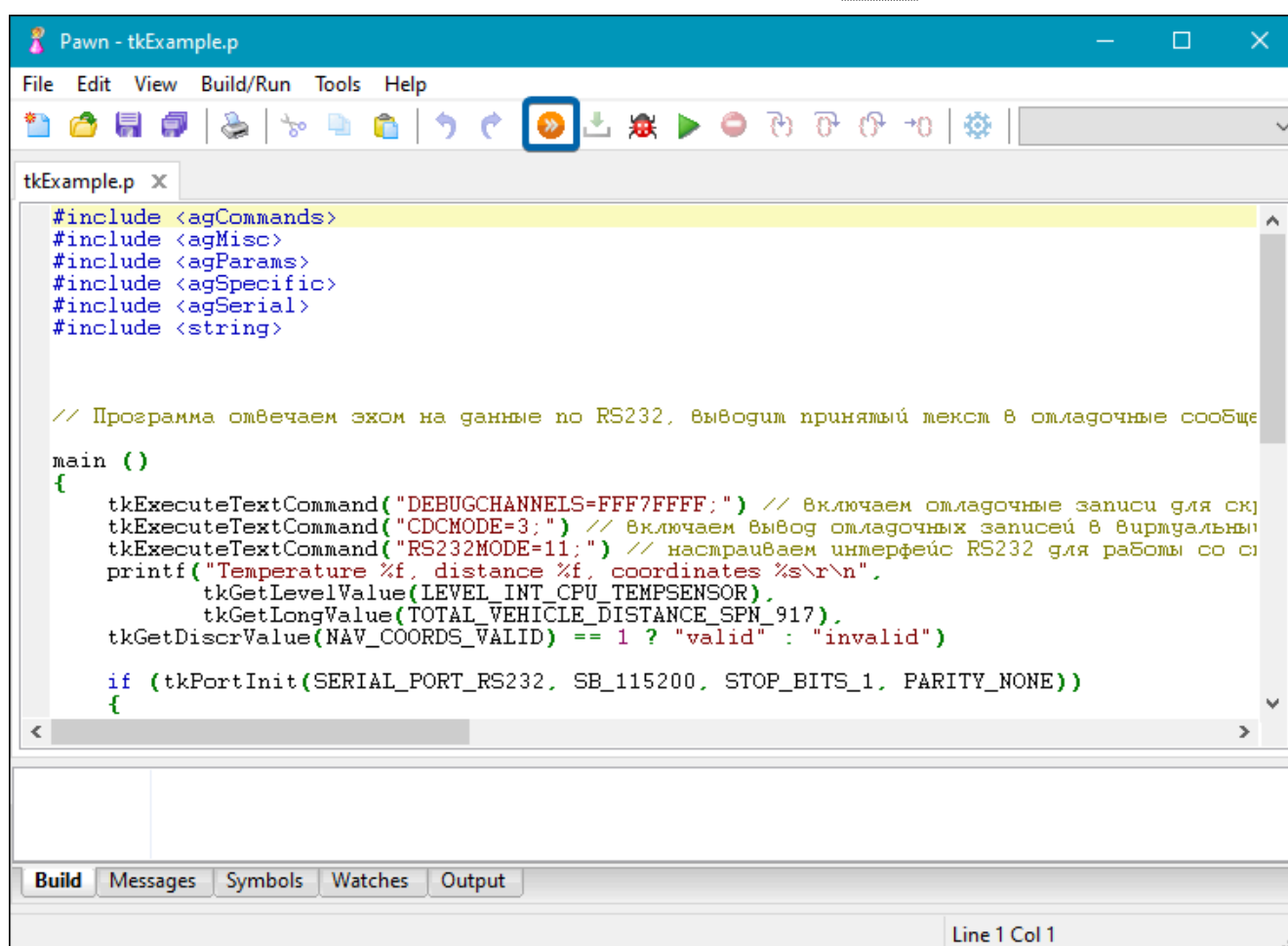


Рис.3. Исходный текст скрипта

3. Нажмите на клавиатуре клавишу F7 или кнопку «Compile Current Script» на панели инструментов Pawn IDE — в директории с файлом *tkExample.p* появится файл *tkExample.amx* (Рис.4).

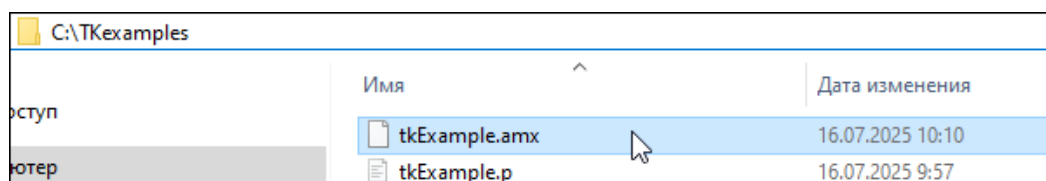


Рис.4. Скомпилированный файл

4. Скопируйте файл `tkExample.amx` в директорию `MCU\SCRIPTS` в терминале. Для этого воспользуйтесь файловым менеджером в конфигураторе АвтоГРАФ GSMConf 5.0 или отправьте файл через сервер.
5. Включите в терминале запись скриптов в журнал с помощью соответствующего чек-бокса в разделе «Логи» конфигуратора или с помощью команды `DEBUGCHANNELS=FFF7FFF;`
6. Разрешите работу скриптов в терминале с помощью переключателя в конфигураторе или команды `SCRIPTSENABLE=1;`
7. Проверьте работу скрипта через отладочный интерфейс CDC конфигуратора, выбрав режим «Вывод отладочной информации» (Рис.5).

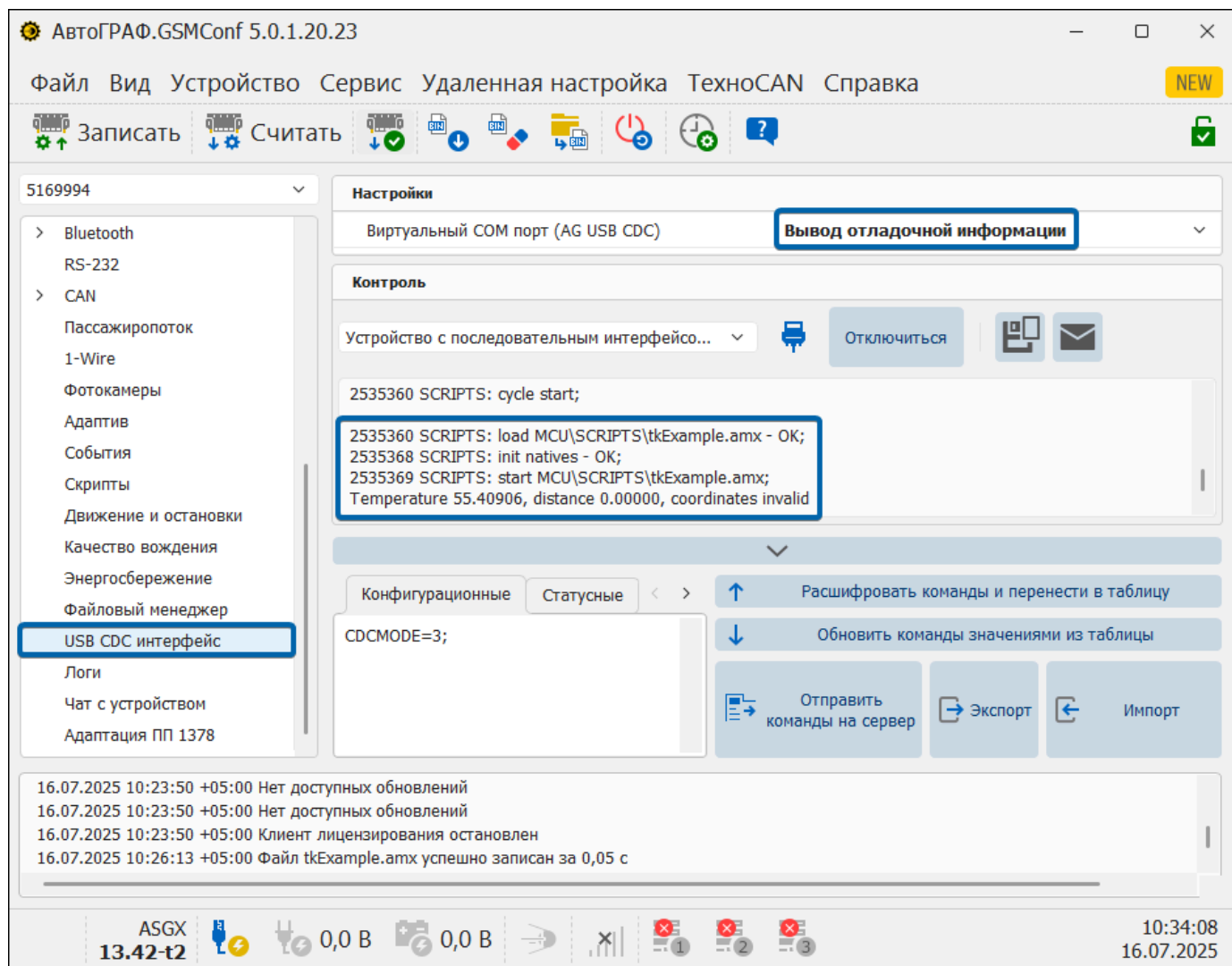


Рис.5. Проверка работы скрипта через отладочный интерфейс CDC

8. Удаленно проверьте наличие файла со скриптом с помощью команды `DIRTREE;`

Процесс выполнения можно проследить в лог-файлах терминала на сервере, если для терминала в конфигураторе включена опция «Передавать логи на сервер» (Рис.6).

Для **обновления** микропрограммы в терминале загрузите в него обновленный файл `.amx`. Если имя файла, имеющегося в терминале, совпадает с именем загружаемого файла, то файл в терминале будет перезаписан автоматически. Для исполнения перезаписанного файла `.amx` нужно отправить команду, которая перезапускает работу скриптов в терминале, — `SCRIPTSRESTART;`

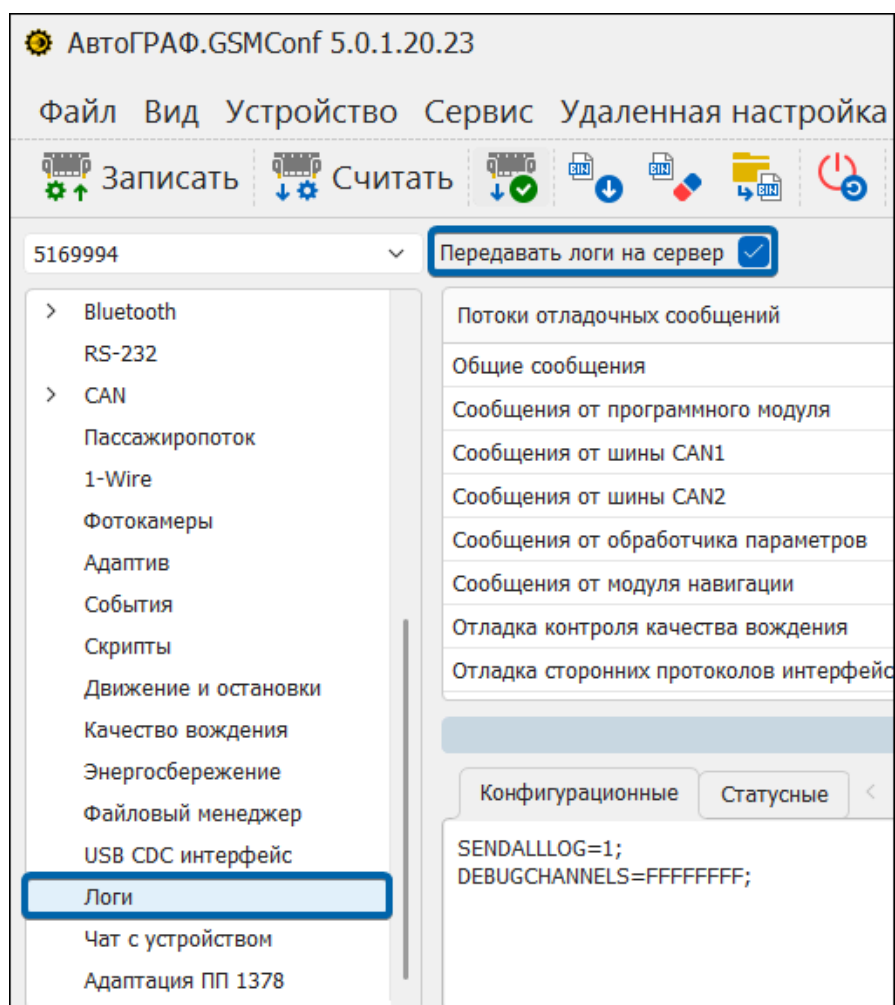


Рис.6. Включение передачи логов на сервер

Для **удаления** файлов со скриптами из терминала отправьте команду `FORMATMCU`. По этой команде все файлы в директории `MCU` будут удалены. Удаление файлов по отдельности не предусмотрено.

Компиляция микропрограмм T.Скрипт через Visual Studio Code

Такой способ компиляции микропрограмм T.Скрипт позволяет:

- проводить контроль целостности файлов загружаемых микропрограмм (путем автоматического запуска файла `amx_to_agx.exe` после компиляции);
- пользоваться расширением для подсветки синтаксиса в Visual Studio Code (например, Pawn Tools);
- пользоваться AI-ассистентом (например, GitHub Copilot).

Предварительно требуется загрузить архив с файлами для компиляции через Visual Studio Code по ссылке https://i.tk-chel.ru/products/T_SCRIPT/Pawn_VSCode.zip.

Для компиляции микропрограммы выполните следующий порядок действий:

1. Скопируйте папку `.vscode` в директорию с файлом скрипта.
2. Скопируйте файл `amx_to_agx.exe` в любую директорию (можно в папку с файлом скрипта).
3. Перейдите в скопированную папку `.vscode` и отредактируйте в ней файл `task.json` (Рис.7):
 - замените значение в строке `"command": "C:\\example\\pawnc.exe"` на путь до вашего компилятора;
 - замените значение в строке `"-iC:\\scripts\\inc"` на путь до папки с файлами скриптов `.inc`;
 - замените значение в строке `"command": "C:\\example\\amx_to_agx.exe"` на путь до скопированного файла `amx_to_agx.exe`.

```
1 {
2   "version": "2.0.0",
3   "tasks": [
4     {
5       "label": "build-normal",
6       "type": "shell",
7       "command": "C:\\example\\pawnc.exe",
8       "args": [
9         "${fileBasename}",
10        "-D${fileDirname}",
11        "-iC:\\scripts\\inc",
12        "-(+'"
13      ],
14      "presentation": {
15        "reveal": "always",
16        "panel": "shared",
17        "clear": true
18      },
19      "problemMatcher": [],
20      "group": "none"
21    },
22    {
23      "label": "convert-amx",
24      "type": "shell",
25      "command": "C:\\example\\amx_to_agx.exe",
26      "args": [
27        "${fileDirname}\\${fileBasenameNoExtension}.amx"
28      ],
```

Рис.7. Редактирование файла `task.json`

4. Если требуется, чтобы вместо файла `.agx` собирался файл `.amx`, то удалите строку `"convert-amx"` (**Рис.8**). Следует учитывать, что в таком случае не будет гарантии целостности собранного файла.

```
37 {
38   "label": "build-and-convert",
39   "dependsOrder": "sequence",
40   "dependsOn": [
41     "build-normal",
42     "convert-amx"
43   ],
```

Рис.8. Настройка сборки файла `.amx`

5. Запустите программу Visual Studio Code и откройте в ней папку с файлом скрипта.
6. Откройте файл скрипта, который нужно скомпилировать, и выполните сборку, выбрав в меню программы пункты `Terminal | Run Build Task...` или нажав на клавиатуре комбинацию клавиш `Ctrl+Shift+B`.

Примечание. Для удобства компиляции рекомендуется изменить комбинацию клавиш сборки (например, на `F5`): выберите в меню программы пункты `File | Preferences | Keyboard Shortcuts`, найдите команду «`Tasks: Run Build Task`» и назначьте ей нужную клавишу или комбинацию клавиш.

Выполнение микропрограмм Т.Скрипт в терминале АвтоГРАФ

Микропрограммы Т.Скрипт выполняются в отдельной задаче операционной системы реального времени (RTOS). Приоритет задачи Т.Скрипт ниже, чем у задач, выполняющих базовые функции прошивки АвтоГРАФ. Используемая RTOS вытесняющего типа работает по следующему принципу: если в процессе выполнения задачи Т.Скрипт появится задача с более высоким приоритетом в состоянии «готова к исполнению», то исполнение переключится на эту более приоритетную задачу.

В ПЗУ и ОЗУ терминалов АвтоГРАФ-GX (ASGX) под нужды выполнения микропрограмм Т.Скрипт выделен объем памяти в 1 Мбайт.

Документация

Настоящий документ сформирован на основании прошивки **13.47-a1**.

Оригинальная документация по языку программирования Pawn доступна по ссылке <https://www.compuphase.com/pawn/pawn.htm> или в директории с установленной Pawn IDE (...\\Pawn\\doc).

К ознакомлению рекомендованы следующие файлы:

- Pawn_Getting_Started.pdf — общее описание языка.
- Pawn_Language_Guide.pdf — подробное руководство по языку Pawn.
- Floating_Point_Support — описание библиотеки работы с числами с плавающей запятой.
- Fixed_Point_Support.pdf — описание библиотеки работы с числами с фиксированной запятой.
- String_Manipulation.pdf — описание библиотеки работы со строками.
- Time_Functions.pdf — описание библиотеки работы с временем и таймером.
- Arguments_Support.pdf — описание библиотеки работы с аргументами.

Списки и описание функций, реализованных в терминале АвтоГРАФ, приведены в последующих разделах.

Функции для работы с CAN-шиной

Список функций	Описание
tkCanInit	Инициализация CAN-шины.
tkCanSetFilter	Установка фильтра приема.
tkCanSend	Отправка сообщения в CAN-шину.
tkCanReceive	Прием сообщения из CAN-шины.

tkCanInit

Инициализация CAN-шины.

Для использования функции подключите файл:

```
#include <agCan.inc>
```

Формат функции:

tkCanInit(speed, active, port)

Параметры:

speed	Скорость шины (по умолчанию 250 кбит/с).
active	Разрешить отправку АСК в шину: <ul style="list-style-type: none">• true — разрешить;• false — запретить.
port	Номер CAN-шины: <ul style="list-style-type: none">• 0 — CAN_1 (по умолчанию);• 1 — CAN_2.

Возвращаемое значение:

Нет.

Пример вызова функции:

```
tkCanInit(500000, true, CAN_1)
```

tkCanSetFilter

Установка фильтра приема.

Для использования функции подключите файл:

```
#include <agCan.inc>
```

Формат функции:

tkCanSetFilter(id, mask, type, num_filter, port)

Параметры:

id	Идентификатор CAN-сообщения.
mask	Маска CAN-сообщения.
type	Тип используемых идентификаторов: <ul style="list-style-type: none">• 0 — STD (11-битные);• 1 — EXT (29-битные).
num_filter	Номер фильтра (0–31).
port	Номер CAN-шины: <ul style="list-style-type: none">• 0 — CAN_1 (по умолчанию);• 1 — CAN_2.

Возвращаемое значение:

Нет.

Пример вызова функции:

```
tkCanSetFilter(0x100, 0x700, 0, CAN_1);
```

tkCanSend

Отправка сообщения в CAN-шину.

Для использования функции подключите файл:

```
#include <agCan.inc>
```

Формат функции:

tkCanSend(CANMSG, port)

Параметры:

CANMSG[]	Сообщение, отправляемое в CAN-шину.
port	Номер CAN-шины: <ul style="list-style-type: none">• 0 — CAN_1 (по умолчанию);• 1 — CAN_2.

Возвращаемое значение:

Нет.

Пример вызова функции:

```
new canMessage[CANMSG];  
canMessage.id = 0x18DAEEF1;  
canMessage.idType = 1;  
canMessage.dataSize = 8;  
canMessage.data[0] = 0x02107E00;  
tkCanSend(canMessage);
```

tkCanReceive

Прием сообщения из CAN-шины.

Для использования функции подключите файл:

```
#include <agCan.inc>
```

Формат функции:

tkCanReceive(CANMSG, port)

Параметры:

CANMSG[]	Сообщение, получаемое из CAN-шины.
port	Номер CAN-шины: <ul style="list-style-type: none">• 0 — CAN_1 (по умолчанию);• 1 — CAN_2.

Возвращаемое значение:

Принято новое сообщение (1 — сообщение получено, 0 — сообщение не получено).

Пример вызова функции:

```
new can_message[CANMSG];  
tkCanReceive(can_message)
```

Функции для работы с текстовыми командами

Список функций	Описание
tkExecuteTextCommand	Выполнение текстовой команды.
tkReadScriptConfig	Чтение пользовательских настроек.
tkWriteScriptConfig	Запись пользовательских настроек.
tkReadScriptConfigSigned	Чтение пользовательских настроек (со знаком).
tkWriteScriptConfigSigned	Запись пользовательских настроек (со знаком).

tkExecuteTextCommand

Выполнение текстовой команды.

Для использования функции подключите файл:

```
#include <agCommands.inc>
```

Формат функции:

```
tkExecuteTextCommand(const command[], answer[], maxlength=sizeof answer)
```

Параметры:

command	Текст исполняемой текстовой команды.
answer	Буфер под ответ команды.
maxlength	Размер буфера под ответ команды.

Возвращаемое значение:

true — команда была выполнена, false — команда не была выполнена.

Примечание. Если текст ответа не требуется, то параметр **answer** можно не передавать.

Примечание. Отправленные этой функцией настройки автоматически не сохраняются в энергонезависимой памяти и могут быть потеряны при отключении питания устройства. Для сохранения настроек отправьте текстовую команду **SAVECONF** после завершения конфигурирования либо перезагрузите устройство.

Примечание. Чтобы избежать исчерпания ресурсов перезаписи внутренней флеш-памяти, вызов команды **SAVECONF** из Т.Скрипт ограничен по времени: не чаще 10 раз в минуту, 20 раз в час и 30 раз в день.

Пример вызова функции:

```
var answer[128];
if(tkExecuteTextCommand("GVERSION", answer)
{
    printf("Version %s\r\n", answer)
}
```

tkReadScriptConfig

Чтение пользовательских настроек.

Для использования функции подключите файл:

```
#include <agCommands.inc>
```

Формат функции:

tkReadScriptConfig(configId)

Параметры:

configId	Порядковый номер пользовательской настройки, 1...10.
-----------------	--

Возвращаемое значение:

Значение пользовательской настройки.

Примечание. Значение пользовательской настройки может быть записано функцией `tkWriteScriptConfig` или текстовой командой `SCRIPTCONFIGn`.

Пример вызова функции:

```
printf("Config 1 value %d", tkReadScriptConfig(1))
```

tkWriteScriptConfig

Запись пользовательских настроек.

Для использования функции подключите файл:

```
#include <agCommands.inc>
```

Формат функции:

tkWriteScriptConfig(configId, value)

Параметры:

configId	Порядковый номер пользовательской настройки, 1...10.
value	Значение пользовательской настройки, 0...4294967295 (0xFFFFFFFF).

Возвращаемое значение:

Нет.

Примечание. Значение пользовательской настройки может быть считано функцией `tkReadScriptConfig` или текстовой командой `GSCRIPTCONFIGn`.

Примечание. Отправленные этой функцией настройки автоматически не сохраняются в энергонезависимой памяти и могут быть потеряны при отключении питания устройства. Для сохранения настроек отправьте текстовую команду `SAVECONF` после завершения конфигурирования либо перезагрузите устройство.

Внимание! Не рекомендуется часто (чаще одного раза в сутки) менять значение пользовательских настроек. Частая смена значения может привести к исчерпанию ресурсов перезаписи внутренней флеш-памяти.

Примечание. Чтобы избежать исчерпания ресурсов перезаписи внутренней флеш-памяти, вызов команды `SAVECONF` из Т.Скрипт ограничен по времени: не чаще 10 раз в минуту, 20 раз в час и 30 раз в день.

Пример вызова функции:

```
tkWriteScriptConfig(1,100)
```

tkReadScriptConfigSigned

Чтение пользовательских настроек (со знаком).

Для использования функции подключите файл:

```
#include <agCommands.inc>
```

Формат функции:

tkReadScriptConfigSigned(configId)

Параметры:

configId	Порядковый номер пользовательской настройки, 1...10.
-----------------	--

Возвращаемое значение:

Знаковое 32-битное значение пользовательской настройки.

Примечание. Значение пользовательской настройки может быть записано функцией `tkWriteScriptConfigSigned` или текстовой командой `SCRIPTCONFIGnS` (если реализована запись знаковых значений).

Пример вызова функции:

```
printf("Config 1 value %d", tkReadScriptConfigSigned(1));
```

tkWriteScriptConfigSigned

Запись пользовательских настроек (со знаком).

Для использования функции подключите файл:

```
#include <agCommands.inc>
```

Формат функции:

tkWriteScriptConfigSigned(configId, value)

Параметры:

configId	Порядковый номер пользовательской настройки, 1...10.
value	Знаковое 32-битное значение пользовательской настройки, -2147483648...2147483647.

Возвращаемое значение:

Нет.

Примечание. Отправленные этой функцией настройки автоматически не сохраняются в энергонезависимой памяти и могут быть потеряны при отключении питания устройства. Для сохранения настроек отправьте текстовую команду `SAVECONF` после завершения конфигурирования либо перезагрузите устройство.

Внимание! Не рекомендуется часто (чаще одного раза в сутки) менять значение пользовательских настроек. Частая смена значения может привести к исчерпанию ресурсов перезаписи внутренней флеш-памяти.

Примечание. Чтобы избежать исчерпания ресурсов перезаписи внутренней флеш-памяти, вызов команды `SAVECONF` из Т.Скрипт ограничен по времени: не чаще 10 раз в минуту, 20 раз в час и 30 раз в день.

Пример вызова функции:

```
tkWriteScriptConfigSigned(1, -100);
```

Функции для работы с событиями

Список функций	Описание
<u>tkGetEventState</u>	Получение состояния события.
<u>tkGetEconomyState</u>	Получение состояния режима экономии.
<u>tickUntilReset</u>	Получение количества миллисекунд до плановой перезагрузки системы.

tkGetEventState

Получение состояния события.

Для использования функции подключите файл:

```
#include <agEvents.inc>
```

Формат функции:

```
tkGetEventState(eventNumber, eventData[])
```

Параметры:

eventNumber	Порядковый номер события, 1...16.
eventData	Массив структур, куда записываются данные о событии: <ul style="list-style-type: none">• .eventState — состояние события:<ul style="list-style-type: none">• 1 — сработка;• 0 — нет сработки.• .eventTrigger — ожидается действие по сработке:<ul style="list-style-type: none">• 1 — ожидается;• 0 — не ожидается.• .eventDeTrigger — ожидается действие по окончании события:<ul style="list-style-type: none">• 1 — ожидается;• 0 — не ожидается.• .sourceState — текущее состояние источника события (зависит от управляющих команд EVENTSOURCE и EVENTTYPE).• .eventTime — время нахождения в состоянии срабатывания, в миллисекундах.

Возвращаемое значение:

true — событие находится в состоянии сработки, false — событие не находится в состоянии сработки.

Примечание. Если расширенная информация о событии не требуется, то параметр **eventData** можно не передавать.

Пример вызова функции:

```
new eventData[.eventState, .eventTrigger, .eventDeTrigger, .sourceState, .eventTime]
tkGetEventState(1, eventData)
printf("Event 1 state %d, time %d\r\n", eventData.eventState, eventData.eventTime)
printf("Event 2 state %d\r\n", tkGetEventState(2))
```

tkGetEconomyState

Получение состояния режима экономии.

Для использования функции подключите файл:

```
#include <agEvents.inc>
```

Формат функции:

```
tkGetEconomyState(eventData[])
```

Параметры:

eventData	<p>Массив структур, куда записываются данные о режиме:</p> <ul style="list-style-type: none">• .eventState — состояние события:<ul style="list-style-type: none">• 1 — сработка;• 0 — нет сработки.• .eventTrigger — ожидается действие по сработке:<ul style="list-style-type: none">• 1 — ожидается;• 0 — не ожидается.• .eventDeTrigger — ожидается действие по окончании события:<ul style="list-style-type: none">• 1 — ожидается;• 0 — не ожидается.• .sourceState — текущее состояние источника события (зависит от управляющих команд ECONOMYSOURCE и ECONOMYTYPE).• .eventTime — время нахождения в состоянии экономии, в миллисекундах.
------------------	--

Возвращаемое значение:

true — устройство находится в режиме экономии, false — устройство не находится в режиме экономии.

Примечание. Если расширенная информация об экономии не требуется, то параметр **eventData** можно не передавать.

Пример вызова функции:

```
new economyData[.eventState, .eventTrigger, .eventDeTrigger, .sourceState, .eventTime]
tkGetEconomyState(economyData)
printf("Economy state %d, time %d\r\n", economyData.eventState, economyData.eventTime)
printf("Economy state %d\r\n", tkGetEconomyState())
```

tickUntilReset

Получение количества миллисекунд до плановой перезагрузки системы.

Для использования функции подключите файл:

```
#include <agEvents.inc>
```

Формат функции:

```
tickUntilReset ()
```

Параметры:

Нет.

Возвращаемое значение:

Время до следующей плановой перезагрузки системы, в миллисекундах.

Пример вызова функции:

```
if ( tickUntilReset() > 1000 ) {  
    // Работаем, если есть время до перезагрузки  
}
```

Функции для работы со входами

Список функций	Описание
tkGetInputDiscr	Получение состояния входа.
tkGetInputAnalog	Получение аналогового значения входа.
tkGetInputFrequency	Получение частоты цифрового импульсного сигнала на входе.
tkGetInputCounter	Получение счетчика импульсов на входе.
tkSetOutputValue	Установка значения на выходе.

tkGetInputDiscr

Получение состояния входа.

Для использования функции подключите файл:

```
#include <agInputs.inc>
```

Формат функции:

tkGetInputDiscr(input_name)

Параметры:

input_name	Номер входа. Может принимать значения, описанные константой input_name: <ul style="list-style-type: none">• 0 — INPUT_U_1;• 1 — INPUT_U_2;• 2 — INPUT_U_3;• 3 — INPUT_U_4;• 4 — INPUT_U_5;• 5 — INPUT_U_6;• 6 — INPUT_U_7;• 7 — INPUT_U_8.
-------------------	---

Возвращаемое значение:

true — на входе лог «1», false — на входе лог «0».

Пример вызова функции:

```
new bool: value = tkGetInputDiscr(INPUT_U_1);
```

tkGetInputAnalog

Получение аналогового значения входа.

Для использования функции подключите файл:

```
#include <agInputs.inc>
```

Формат функции:

tkGetInputAnalog(input_name)

Параметры:

input_name	Номер входа. Может принимать значения, описанные константой input_name: <ul style="list-style-type: none">• 0 — INPUT_U_1;• 1 — INPUT_U_2;• 2 — INPUT_U_3;• 3 — INPUT_U_4;• 4 — INPUT_U_5;• 5 — INPUT_U_6;• 6 — INPUT_U_7;• 7 — INPUT_U_8.
-------------------	---

Возвращаемое значение:

Напряжение на входе, в милливольтках.

Пример вызова функции:

```
new value = tkGetInputAnalog(INPUT_U_1);
```

tkGetInputFrequency

Получение частоты цифрового импульсного сигнала на входе.

Для использования функции подключите файл:

```
#include <agInputs.inc>
```

Формат функции:

tkGetInputFrequency(input_name)

Параметры:

input_name	Номер входа. Может принимать значения, описанные константой input_name: <ul style="list-style-type: none">• 0 — INPUT_U_1;• 1 — INPUT_U_2;• 2 — INPUT_U_3;• 3 — INPUT_U_4;• 4 — INPUT_U_5;• 5 — INPUT_U_6;• 6 — INPUT_U_7;• 7 — INPUT_U_8.
-------------------	---

Возвращаемое значение:

Частота сигнала (Float), в герцах.

Пример вызова функции:

```
new Float: freq = tkGetInputFrequency(INPUT_U_2);
```

tkGetInputCounter

Получение счетчика импульсов на входе.

Для использования функции подключите файл:

```
#include <agInputs.inc>
```

Формат функции:

```
tkGetInputCounter(input_name)
```

Параметры:

input_name	Номер входа. Может принимать значения, описанные константой input_name: <ul style="list-style-type: none">• 0 — INPUT_U_1;• 1 — INPUT_U_2;• 2 — INPUT_U_3;• 3 — INPUT_U_4;• 4 — INPUT_U_5;• 5 — INPUT_U_6;• 6 — INPUT_U_7;• 7 — INPUT_U_8.
-------------------	---

Возвращаемое значение:

Количество импульсов, накопленных на входе с момента последнего обнуления.

Пример вызова функции:

```
new count = tkGetInputCounter(INPUT_U_3);
```

tkSetOutputValue

Установка значения на выходе.

Для использования функции подключите файл:

```
#include <agInputs.inc>
```

Формат функции:

tkSetOutputValue(outputNum, value)

Параметры:

outputNum	Номер выхода. Может принимать значения, описанные константой output_name: <ul style="list-style-type: none">• 0 — OUTPUT_1;• 1 — OUTPUT_2;• 2 — OUTPUT_3.
value	Новое состояние выхода: <ul style="list-style-type: none">• true — выход замкнут (притянут к земле);• false — выход разомкнут.

Пример вызова функции:

```
tkSetOutputValue(OUTPUT_1, false);
```

Различные встроенные функции

Список функций	Описание
tkSwapBuf	Изменение порядка байтов в ячейках массива.
tkGetIntFromBuf	Получение числа (4-байтового int) из буфера по позиции.
tkGetIMEI	Получение IMEI устройства и сохранение в массив.
printfHex	Вывод массива в шестнадцатеричном формате.
tkCrc8	Вычисление CRC-8/MAXIM.
tkCrc16	Вычисление CRC-16.
tkCrc32	Вычисление CRC-32/JAMCRC.
tkCrc8_MinTrans	Вычисление CRC-8.
tkCrc16_MinTrans	Вычисление CRC-16/CCITT.
memset	Заполнение указанной области памяти заданным значением.

tkSwapBuf

Изменение порядка байтов в ячейках массива.

Для использования функции подключите файл:

```
#include <agMisc.inc>
```

Функция позволяет поменять байты местами внутри ячейки массива из big-endian в little-endian.

Формат функции:

tkSwapBuf(buf[], size)

Параметры:

buf[]	Имя модифицируемого массива.
size	Размер модифицируемого массива, должен быть кратен 4.

Возвращаемое значение:

Нет.

Пример вызова функции:

```
tkSwapBuf(buf, 16)
```

tkGetIntFromBuf

Получение числа (4-байтового int) из буфера по позиции.

Для использования функции подключите файл:

```
#include <agMisc.inc>
```

Формат функции:

tkGetIntFromBuf(buf{}, index)

Параметры:

buf	Буфер, из которого будет извлекаться число.
index	Индекс байта, начиная с которого число располагается в массиве.

Возвращаемое значение:

Число, извлеченное из буфера.

Пример вызова функции:

```
new buf{15} = {1,2,3,4,5,6,7,8,9,10,11,12,13,14,15};  
new index = 2;  
new ret = tkGetIntFromBuf(buf, index);
```

tkGetIMEI

Получение IMEI устройства и сохранение в массив.

Для использования функции подключите файл:

```
#include <agMisc.inc>
```

Формат функции:

tkGetIMEI(res[], resSize)

Параметры:

res	Массив, в который будет записан IMEI в виде строки символов.
resSize	Размер массива res (должен быть не менее 16).

Возвращаемое значение:

Нет.

Пример вызова функции:

```
new imei[16];  
tkGetIMEI(imei, sizeof imei);  
printf("imei: %s\r\n", imei);
```

printfHex

Вывод массива в шестнадцатеричном формате.

Для использования функции подключите файл:

```
#include <agMisc.inc>
```

Формат функции:

```
printfHex(buf[], size)
```

Параметры:

buf	Массив байтов для вывода в шестнадцатеричном формате.
size	Размер массива в байтах.

Возвращаемое значение:

Нет.

Пример вызова функции:

```
new data{5} = { 0xDE, 0xAD, 0xBE, 0xEF, 0x01 };  
printfHex(data, 5);
```

tkCrc8

Вычисление CRC-8/MAXIM.

Для использования функции подключите файл:

```
#include <agMisc.inc>
```

- Poly : $0x31 \ x^8 + x^5 + x^4 + 1$
- Init : 0x00
- Revert : true
- XorOut : 0x00
- Check : 0xA1 («123456789»)

Формат функции:

tkCrc8(crc, buf[], len)

Параметры:

crc	Начальное значение CRC.
buf	Массив байтов, для которого вычисляется CRC.
len	Количество байтов в массиве buf .

Возвращаемое значение:

CRC-8 значение.

Пример вызова функции:

```
new data{9} = "123456789";  
new crc_init = 0x00;  
new crc = tkCrc8(crc_init, data, 9);  
// Ожидаемый результат: 0xA1
```

tkCrc16

Вычисление CRC-16.

Для использования функции подключите файл:

```
#include <agMisc.inc>
```

- Poly : $0x8005 x^{16} + x^{15} + x^2 + 1$
- Init : $0xFFFF$
- Revert : true
- XorOut : $0x0000$
- Check : $0x4B37$ («123456789»)

Формат функции:

tkCrc16(crc, buf[], len)

Параметры:

crc	Начальное значение CRC.
buf	Массив байтов, для которого вычисляется CRC.
len	Количество байтов в массиве buf .

Возвращаемое значение:

CRC-16 значение.

Пример вызова функции:

```
new data{9} = "123456789";  
new crc_init = 0xFFFF;  
new crc = tkCrc16(crc_init, data, 9);  
// Ожидаемый результат: 0x4B37
```

tkCrc32

Вычисление CRC-32/JAMCRC.

Для использования функции подключите файл:

```
#include <agMisc.inc>
```

- Poly : 0x04C11DB7 $x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$
- Init : 0xFFFFFFFF
- Revert : true
- XorOut : 0x00000000
- Check : 0x340BC6D9 («123456789»)

Формат функции:

tkCrc32(crc, buf[], len)

Параметры:

crc	Начальное значение CRC.
buf	Массив байтов, для которого вычисляется CRC.
len	Количество байтов в массиве buf .

Возвращаемое значение:

CRC-32 значение.

Пример вызова функции:

```
new data{9} = "123456789";  
new crc_init = 0xFFFFFFFF;  
new crc = tkCrc32(crc_init, data, 9);  
// Ожидаемый результат: 0x340BC6D9
```

tkCrc8_MinTrans

Вычисление CRC-8.

Для использования функции подключите файл:

```
#include <agMisc.inc>
```

- Poly : $0x31 \ x^8 + x^5 + x^4 + 1$
- Init : 0xFF
- Revert : false
- XorOut : 0x00
- Check : 0xF7 («123456789»)

Формат функции:

tkCrc8_MinTrans(buf[], len)

Параметры:

buf	Массив байтов, для которого вычисляется CRC.
len	Количество байтов в массиве buf .

Возвращаемое значение:

CRC-8 значение.

Пример вызова функции:

```
new data{9} = "123456789";  
new crc = tkCrc8_MinTrans(data, 9);  
// Ожидаемый результат: 0xF7
```

tkCrc16_MinTrans

Вычисление CRC-16/CCITT.

Для использования функции подключите файл:

```
#include <agMisc.inc>
```

- Poly : $0x1021 \cdot x^{16} + x^{12} + x^5 + 1$
- Init : 0xFFFF
- Revert : false
- XorOut : 0x0000
- Check : 0x29B1 («123456789»)

Формат функции:

tkCrc16_MinTrans(crc, buf[], len)

Параметры:

crc	Начальное значение CRC.
buf	Массив байтов, для которого вычисляется CRC.
len	Количество байтов в массиве buf .

Возвращаемое значение:

CRC-16 значение.

Пример вызова функции:

```
new data{9} = "123456789";  
new crc_init = 0xFFFF;  
new crc = tkCrc16_MinTrans(crc_init, data, 9);  
// Ожидаемый результат: 0x29B1
```

memset

Заполнение указанной области памяти заданным значением.

Для использования функции подключите файл:

```
#include <agMisc.inc>
```

Формат функции:

```
memset(dest[], index=0, numbytes, value=0, maxlenh=sizeof dest)
```

Параметры:

dest[]	Массив, который требуется заполнить.
index	Номер байта в массиве dest[] , с которого требуется начать заполнение (по умолчанию 0). Необязательный параметр.
numbytes	Количество байтов для заполнения.
value	Значение для заполнения (0–255, по умолчанию 0). Необязательный параметр.
maxlength	Максимальный размер массива dest[] . Необязательный параметр.

Возвращаемое значение:

Количество заполненных байтов (целое число).

Пример вызова функции:

```
// Обнуление массива
new buffer[10];
memset(buffer, 0, sizeof(buffer)*4); // Умножаем на 4 (размер ячейки)

// Заполнение со смещением
new data[20];
memset(data, 5*4, 10, 0xFF); // Заполняет 10 байт значением 0xFF начиная с 5-й ячейки

// Частичное заполнение с контролем границ
memset(data, 0, 15, 0xAA, sizeof(data));
```

Функции для работы с навигационными приемниками

Список функций	Описание
tkGetPosition	Получение текущих навигационных данных.
tkSavePoint	Принудительное сохранение точки с координатами.

tkGetPosition

Получение текущих навигационных данных.

Для использования функции подключите файл:

```
#include <agNavigation.inc>
```

Формат функции:

tkGetPosition(position[])

Параметры:

position	Массив структур, куда записываются навигационные данные: <ul style="list-style-type: none">• .lat — широта в градусах;• .lon — долгота в градусах;• .height — высота над уровнем моря, в метрах;• .course — курс (направление) в градусах;• .speed — скорость в км/ч;• .hdop — коэффициент точности HDOP (горизонтальная погрешность);• .numSV — количество спутников.
-----------------	--

Возвращаемое значение:

true — навигационные данные достоверны, false — данные недостоверны.

Пример вызова функции:

```
new pos[Float: .lat, Float: .lon, Float: .height, Float: .course, Float: .speed, Float: .hdop, .numSV];
if (tkGetPosition(pos)) {
    printf("Lat: %f, Lon: %f, H: %f m, C: %f°, V: %f km/h, HDOP: %f, numSV: %d\r\n",
        pos.lat, pos.lon, pos.height, pos.course, pos.speed, pos.hdop, pos.numSV);
}
else {
    printf("Navigation data is unavailable\r\n");
}
```

tkSavePoint

Принудительное сохранение точки с координатами.

Для использования функции подключите файл:

```
#include <agNavigation.inc>
```

Формат функции:

tkSavePoint(secondsToWrite)

Параметры:

secondsToWrite	Количество предыдущих секунд, за которые можно записать координаты. По умолчанию 1 — текущая точка (максимум 5).
-----------------------	---

Возвращаемое значение:

Нет.

Пример вызова функции:

```
tkSavePoint(1);
```

Функции для работы с параметрами устройств АвтоГРАФ

Список функций	Описание
<u>tkGetLevelValue</u>	Запрос значения уровня.
<u>tkSetLevelValue</u>	Установка значения уровня.
<u>tkLevelValueUpdated</u>	Проверка актуальности уровневого параметра.
<u>tkGetDiscrValue</u>	Запрос дискретного параметра.
<u>tkSetDiscrValue</u>	Установка дискретного параметра.
<u>tkDiscrValueUpdated</u>	Проверка актуальности дискретного параметра.
<u>tkGetLongValue</u>	Запрос значения длинного параметра.
<u>tkSetLongValue</u>	Установка значения длинного параметра.
<u>tkLongValueUpdated</u>	Проверка актуальности длинного параметра.
<u>tkGetGenericFloatValue</u>	Запрос значения произвольного параметра как числа с плавающей запятой.
<u>tkGetGenericIntValue</u>	Запрос значения произвольного параметра как целочисленного.
<u>tkSetGenericFloatValue</u>	Установка значения произвольного параметра как числа с плавающей запятой.
<u>tkSetGenericIntValue</u>	Установка значения произвольного параметра как целочисленного.
<u>tkGenericValueUpdated</u>	Проверка актуальности произвольного параметра.
<u>tkWriteValue</u>	Запись пользовательских данных в бинарный файл устройства.
<u>tkWriteFloatValue</u>	Запись пользовательских данных типа float в бинарный файл устройства.
<u>tkWriteLongRecord</u>	Запись пользовательского буфера в бинарный файл устройства.
<u>tkSetDeviceFlag</u>	Установка состояния флага устройства.
<u>tkGetDeviceFlag</u>	Чтение текущего состояния битов устройства.
<u>tkWriteTkiaLongRecord</u>	Сохранение пользовательского буфера в запись ТКІА бинарного файла устройства.

Список групп параметров	Описание
<u>DeviceFlags</u>	Флаги (биты состояний) контроллера.
<u>ServerConnectionStatus</u>	Статус подключения к серверу.
<u>DangerZoneType</u>	Типы зон в системе приближения.

tkGetLevelValue

Запрос значения уровня.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

tkGetLevelValue(id)

Параметры:

id	Номер параметра из LevelId.
----	-----------------------------

Возвращаемое значение:

Значение уровня типа Float.

Примечание. Параметры LEVEL_RAM_PARAM_1...LEVEL_RAM_PARAM_10 можно использовать для передачи данных скрипту, меняя их с помощью текстовой команды SETLEVELVALUEn, отправленной через сервер или посредством SMS, AGL и событий.

Пример вызова функции:

```
new Float: value = tkGetLevelValue(VEHICLE_SPEED_SPN_84)
```

tkSetLevelValue

Установка значения уровня.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

tkSetLevelValue(id, value)

Параметры:

id	Номер параметра из <code>LevelId</code> .
value	Устанавливаемое значение уровня (Float).

Возвращаемое значение:

Нет.

Примечание. Параметр, записанный командой `tkSetLevelValue`, попадает в записи устройства с периодом и логикой того модуля, к которому этот параметр относится. Так, уровни, относящиеся к CAN, будут записаны с периодом записи CAN параметров, уровни, относящиеся к датчикам уровня топлива, будут записаны с периодом записи данных с ДУТ и так далее.

Некоторые уровневые параметры, например, `LEVEL_RAM_PARAM_n`, в записи устройства не попадают, но, как и другие параметры, могут использоваться для настройки адаптивов, в событиях и в Т.Скрипт.

Если требуется гарантированно записать определенное значение в определенный момент, то следует использовать функции `tkWriteValue` и `tkWriteFloatValue`.

Примечание. Некоторые параметры, например, напряжение аналогового входа, автоматически перезаписываются прошивкой устройства, и запись в такие параметры функцией `tkSetLevelValue` не имеет смысла.

Примечание. Параметры `LEVEL_RAM_PARAM_1...LEVEL_RAM_PARAM_10` можно использовать для передачи данных скрипту, меняя их с помощью текстовой команды `SETLEVELVALUEn`, отправленной через сервер или посредством SMS, AGL и событий.

Пример вызова функции:

```
tkSetLevelValue(VEHICLE_SPEED_SPN_84, 2.75)
```

tkLevelValueUpdated

Проверка актуальности уровневого параметра.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

tkLevelValueUpdated(id, bool: reset)

Параметры:

id	Номер параметра из <u>LevelId</u> .
reset	Флаг сброса актуальности после выполнения функции.

Возвращаемое значение:

true — значение параметра было обновлено после последнего сброса, false — нет.

Пример вызова функции:

```
tkLevelValueUpdated(VEHICLE_SPEED_SPN_84, true);
```

tkGetDiscrValue

Запрос дискретного параметра.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

tkGetDiscrValue(id)

Параметры:

id	Номер параметра из DiscrParamId.
----	----------------------------------

Возвращаемое значение:

Значение параметра в uint32_t.

Пример вызова функции:

```
new DiscrValue = tkGetDiscrValue(DISCR_RAM_PARAM_1)
```

tkSetDiscrValue

Установка дискретного параметра.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

tkSetDiscrValue(id, value)

Параметры:

id	Номер параметра из DiscrParamId.
value	Устанавливаемое значение параметра в uint32_t.

Возвращаемое значение:

Нет.

Примечание. Параметр, записанный командой `tkSetDiscrValue`, попадает в запись устройства с периодом и логикой того модуля, к которому этот параметр относится. Значительная часть дискретных параметров относится к шине CAN и будет записана с периодом записи CAN параметров.

Некоторые дискретные параметры, например, `DISCR_RAM_PARAM_n`, в записи устройства не попадают, но, как и другие параметры, могут использоваться для настройки адаптивов, в событиях и в Т.Скрипт.

Если требуется гарантированно записать определенное значение в определенный момент, то следует использовать функцию `tkWriteValue`.

Примечание. Некоторые параметры, например, состояние выхода, являются справочными и автоматически перезаписываются прошивкой устройства, и запись в такие параметры функцией `tkSetDiscrValue` не имеет смысла.

Примечание. Параметры `DISCR_RAM_PARAM_1...DISCR_RAM_PARAM_10` можно использовать для передачи данных скрипту, меняя их с помощью текстовой команды `SETDISCRVALUEn`, отправленной через сервер или посредством SMS, AGL и событий.

Пример вызова функции:

```
tkSetDiscrValue(DISCR_RAM_PARAM_1, 42)
```

tkDiscrValueUpdated

Проверка актуальности дискретного параметра.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

tkDiscrValueUpdated(id, bool: reset)

Параметры:

id	Номер параметра из DiscrParamId.
reset	Флаг сброса актуальности после выполнения функции.

Возвращаемое значение:

true — значение параметра было обновлено после последнего сброса, false — нет.

Примечание. Параметры DISCR_RAM_PARAM_1...DISCR_RAM_PARAM_10 можно использовать для передачи данных скрипту, меняя их с помощью текстовой команды SETDISCRVALUEn, отправленной через сервер или посредством SMS, AGL и событий.

Пример вызова функции:

```
tkDiscrValueUpdated(DISCR_RAM_PARAM_1, true);
```

tkGetLongValue

Запрос значения длинного параметра.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

tkGetLongValue(id)

Параметры:

id	Номер параметра из LongParamId.
----	---------------------------------

Возвращаемое значение:

Значение типа Float.

Пример вызова функции:

```
new Float: value = tkGetLongValue(TOTAL_VEHICLE_DISTANCE_SPN_917)
```

tkSetLongValue

Установка значения длинного параметра.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

tkSetLongValue(id, value)

Параметры:

id	Номер параметра из LongParamId.
value	Устанавливаемое значение длинного параметра (Float).

Возвращаемое значение:

Нет.

Примечание. Параметр, записанный командой `tkSetLongValue`, попадает в записи устройства с периодом записи CAN параметров.

Если требуется гарантированно записать определенное значение в определенный момент, то следует использовать функции `tkWriteValue` и `tkWriteFloatValue`.

Пример вызова функции:

```
tkSetLongValue(TOTAL_VEHICLE_DISTANCE_SPN_917, 2.75)
```

tkLongValueUpdated

Проверка актуальности длинного параметра.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

tkLongValueUpdated(id, bool: reset)

Параметры:

id	Номер параметра из LongParamId.
reset	Флаг сброса актуальности после выполнения функции.

Возвращаемое значение:

true — значение параметра было обновлено после последнего сброса, false — нет.

Пример вызова функции:

```
tkLongValueUpdated(TOTAL_VEHICLE_DISTANCE_SPN_917, true);
```

tkGetGenericFloatValue

Запрос значения произвольного параметра как числа с плавающей запятой.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

```
tkGetGenericFloatValue(id)
```

Параметры:

id	Номер параметра из GenericParamsId.
----	-------------------------------------

Возвращаемое значение:

Значение типа Float.

Пример вызова функции:

```
new Float: value = tkGetGenericFloatValue(GENERIC_CAN_PARAM_1)
```

tkGetGenericIntValue

Запрос значения произвольного параметра как целочисленного.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

tkGetGenericIntValue(id)

Параметры:

id	Номер параметра из GenericParamsId.
----	-------------------------------------

Возвращаемое значение:

Значение типа int.

Пример вызова функции:

```
new value = tkGetGenericIntValue(GENERIC_CAN_PARAM_6)
```

tkSetGenericFloatValue

Установка значения произвольного параметра как числа с плавающей запятой.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

tkSetGenericFloatValue(id, value)

Параметры:

id	Номер параметра из <code>GenericParamsId</code> .
value	Устанавливаемое значение произвольного параметра (Float).

Возвращаемое значение:

Нет.

Примечание. Параметр, записанный командой `tkSetGenericFloatValue` с **id** в диапазоне от `GENERIC_CAN_PARAM_1` до `GENERIC_CAN_PARAM_50`, попадает в записи устройства с периодом записи CAN параметров. Параметр, записанный командой `tkSetGenericFloatValue` с **id** в диапазоне от `GENERIC_MODBUS_PARAM_1` до `GENERIC_MODBUS_PARAM_100`, попадает в записи устройства с периодом записи параметров MODBUS.

При записи произвольный параметр будет преобразован с учетом настроек и коэффициентов, заданных в устройстве для соответствующего параметра, поэтому записанное значение может отличаться от установленного.

Если требуется гарантированно записать определенное значение в определенный момент, то следует использовать функции `tkWriteValue` и `tkWriteFloatValue`.

Пример вызова функции:

```
tkSetGenericFloatValue(GENERIC_CAN_PARAM_6, 1.5)
```

tkSetGenericIntValue

Установка значения произвольного параметра как целочисленного.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

tkSetGenericIntValue(id, value)

Параметры:

id	Номер параметра из <code>GenericParamsId</code> .
value	Устанавливаемое значение произвольного параметра.

Примечание. Параметр, записанный командой `tkSetGenericFloatValue` с **id** в диапазоне от `GENERIC_CAN_PARAM_1` до `GENERIC_CAN_PARAM_50`, попадает в записи устройства с периодом записи CAN параметров. Параметр, записанный командой `tkSetGenericFloatValue` с **id** в диапазоне от `GENERIC_MODBUS_PARAM_1` до `GENERIC_MODBUS_PARAM_100`, попадает в записи устройства с периодом записи параметров MODBUS.

Если требуется гарантированно записать определенное значение в определенный момент, то следует использовать функции `tkWriteValue` и `tkWriteFloatValue`.

Возвращаемое значение:

Нет.

Пример вызова функции:

```
tkSetGenericIntValue(GENERIC_CAN_PARAM_6, 0xface)
```

tkGenericValueUpdated

Проверка актуальности произвольного параметра.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

tkGenericValueUpdated(id, bool: reset)

Параметры:

id	Номер параметра из <u>GenericParamsId</u> .
reset	Флаг сброса актуальности после выполнения функции.

Возвращаемое значение:

true — значение параметра было обновлено после последнего сброса, false — нет.

Пример вызова функции:

```
tkGenericValueUpdated(GENERIC_CAN_PARAM_1, true);
```

tkWriteValue

Запись пользовательских данных в бинарный файл устройства.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Делает запись пользовательского значения в бинарный файл устройства.

Формат функции:

tkWriteValue(paramId, paramValue)

Параметры:

paramId	ID параметра, должен быть в диапазоне 0...65535 (0xffff).
paramValue	Значение параметра, 0...4294967295 (0xFFFFFFFF).

Возвращаемое значение:

Нет.

Примечание. Значение параметра **paramValue** будет записано в момент вызова функции в прочий числовой параметр с индексом ($0xFA0000 + \text{paramId}$).

Примечание. Функции `tkWriteValue` и `tkWriteFloatValue` записывают прочие числовые параметры с индексами в одинаковом диапазоне. Разделение по типу идет в ПО верхнего уровня.

Примечание. В диспетчерском ПО АвтоГРАФ.Pro значение параметра можно получить с помощью вызова функции `NDUInt(0xFA0000 + paramId)`

Пример вызова функции:

```
tkWriteValue(10, 556677)
```

tkWriteFloatValue

Запись пользовательских данных типа float в бинарный файл устройства.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Делает запись пользовательского значения в бинарный файл устройства.

Формат функции:

tkWriteFloatValue(paramId, Float:paramValue)

Параметры:

paramId	ID параметра, должен быть в диапазоне 0...65535 (0xffff).
paramValue	Значение параметра (Float).

Возвращаемое значение:

Нет.

Примечание. Значение параметра **paramValue** будет записано в момент вызова функции в прочий числовой параметр с индексом ($0xFA0000 + \mathbf{paramId}$) в формате IEEE 754 одинарной точности (32 бит).

Примечание. Функции tkWriteValue и tkWriteFloatValue записывают прочие числовые параметры с индексами в одинаковом диапазоне. Разделение по типу идет в ПО верхнего уровня.

Примечание. В диспетчерском ПО АвтоГРАФ.Pro значение параметра можно получить с помощью вызова функции NDFloat($0xFA0000 + \mathbf{paramId}$)

Пример вызова функции:

```
tkWriteFloatValue(11, 1.5)
```

tkWriteLongRecord

Запись пользовательского буфера в бинарный файл устройства.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Делает запись пользовательского бинарного массива или строки в бинарный файл устройства.

Формат функции:

tkWriteLongRecord (paramId, buf{}, bufSize, bool:pack=true)

Параметры:

paramId	ID параметра, должен быть в диапазоне 0...65535 (0xffff).
buf	Записываемые данные.
bufSize	Длина (размер) записываемых данных, 1...1530 байт.
pack	Флаг упаковки приемного буфера.

Возвращаемое значение:

Нет.

Примечание. Содержимое буфера **buf** будет записано в момент вызова функции в длинную запись с индексом (0xFA0000 + **paramId**).

Пример вызова функции:

```
tkWriteLongRecord(10, "Script started", 14, true)
```

tkSetDeviceFlag

Установка состояния флага устройства.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

tkSetDeviceFlag(flags, status)

Параметры:

flags	Битовая маска (см. DeviceFlags).
status	true — установить флаг, false — сбросить флаг.

Возвращаемое значение:

Нет.

Примечание. Флаги устройства записываются в каждую запись устройства, и могут быть использованы в настройках событий.

Примечание. Некоторые флаги, например, наличие основного питания, автоматически перезаписывается прошивкой устройства, и запись в такие параметры функцией `tkSetDeviceFlag` не имеет смысла.

Пример вызова функции:

```
tkSetDeviceFlag(DF_MASK_ALARM | DF_MASK_GSM, true);
```

tkGetDeviceFlag

Чтение текущего состояния битов устройства.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Формат функции:

tkGetDeviceFlag(flags)

Параметры:

flags	Битовая маска (см. DeviceFlags).
--------------	----------------------------------

Возвращаемое значение:

Набор установленных битов из параметра **flags**.

Пример вызова функции:

```
new state = tkGetDeviceFlag(DF_MASK_BORT | DF_MASK_USBCONNECT);  
if (state & DF_MASK_BORT) {}
```

tkWriteTkiaLongRecord

Сохранение пользовательского буфера в запись ТКІА бинарного файла устройства.

Для использования функции подключите файл:

```
#include <agParams.inc>
```

Сохраняет данные из пользовательского бинарного массива или строки в запись ТКІА бинарного файла устройства.

Формат функции:

tkWriteTkiaLongRecord (buf{}, bufSize, bool:pack=true))

Параметры:

buf	Записываемые данные.
bufSize	Длина (размер) записываемых данных, 1...1530 байт.
pack	Флаг упаковки приемного буфера.

Возвращаемое значение:

Нет.

Примечание. Содержимое буфера **buf** будет сохранено в длинную запись с индексом 0x50 в момент вызова функции.

Пример вызова функции:

```
tkWriteTkiaLongRecord("Script started", 14, true)
```

DeviceFlags

Флаги (биты состояний) контроллера.

DF_MASK_READ1 = 0x00000100	9 — данные отправлены на первый сервер (только в записях).
DF_MASK_READ2 = 0x00000200	10 — данные отправлены на второй сервер (только в записях).
DF_MASK_BORT = 0x00000400	11 — наличие основного питания.
DF_MASK_RESERV = 0x00000800	12 — наличие питания от внешнего аккумулятора.
DF_MASK_INANTOK = 0x00001000	13 — состояние антенны навигационного приемника.
DF_MASK_OUTANTOK = 0x00002000	14 — зарезервировано.
DF_MASK_USBCONNECT = 0x00004000	15 — подключено USB.
DF_MASK_ALARM = 0x00008000	16 — нажата тревожная кнопка.
DF_MASK_RPMCAN = 0x00010000	17 — есть обороты по CAN.
DF_MASK_ROAMING = 0x00020000	18 — контроллер находится в роуминге.
DF_MASK_LOADING = 0x00040000	19 — идет погрузка в транспортное средство.
DF_MASK_GSM = 0x00080000	20 — наличие GSM сигнала.
DF_MASK_ISSTAND = 0x00100000	21 — остановка.
DF_MASK_AKK_IN = 0x00400000	23 — наличие питания от внутреннего аккумулятора.
DF_MASK_READ3 = 0x00800000	24 — данные отправлены на третий сервер (только в записях).

ServerConnectionStatus

Статус подключения к серверу.

SCS_BEGINNING = 0	0 — начальное состояние подключения.
SCS_CLOSED = 1	1 — соединение закрыто.
SCS_CONNECTING = 2	2 — попытка подключения.
SCS_CONNECTED = 3	3 — соединение установлено.
SCS_DATA_SENT = 4	4 — данные на сервер отправлены.
SCS_ANSWER_OK = 5	5 — успешная передача данных на сервер.
SCS_ERROR_PASSWORD = 100	100 — пароль на сервере и контроллере не совпадает.
SCS_ERROR_NOT_SERVICED = 101	101 — контроллер не обслуживается на сервере.
SCS_ERROR_WRONG_CONFIGURATION = 102	102 — некорректная конфигурация сервера.
SCS_ERROR_WRONG_CHANNEL = 103	103 — передача данных по неправильному каналу.
SCS_ERROR_WRONG_PROTOCOL = 104	104 — протокол АвтоГРАФ (legacy) недоступен для комбинации данного сервера и контроллера.
SCS_ERROR_NOT_ANSWER_OK = 105	105 — нет ответа от сервера.

DangerZoneType

Типы зон в системе приближения.

DANGER_ZONE_SAFE = 0	0 — зеленая зона.
DANGER_ZONE_DANGER = 1	1 — красная зона.
DANGER_ZONE_WARNING = 2	2 — желтая зона.
DANGER_ZONE_UWB_DISABLED = 3	3 — UWB выключен.

Функции для работы с последовательными портами

Список функций	Описание
tkPortInit	Инициализация порта.
tkPortDeinit	Закрытие порта.
tkPortReadPackage	Чтение из порта пакетов, разделенных по временной задержке.
tkPortWrite	Запись данных в порт.
tkPortClearReadBuf	Очистка приемного буфера.

tkPortInit

Инициализация порта.

Для использования функции подключите файл:

```
#include <agSerial.inc>
```

Формат функции:

tkPortInit(index, speed, stopBits, parity)

Параметры:

index	Индекс открываемого порта: <ul style="list-style-type: none">• 0 — SERIAL_PORT_RS232 (RS-232);• 1 — SERIAL_PORT_RS485_1 (RS-485-1);• 2 — SERIAL_PORT_RS485_2 (RS-485-2).
speed	Скорость обмена данными: <ul style="list-style-type: none">• 0 — SB_AUTO (0 бит/с);• 9600 — SB_9600 (9600 бит/с);• 19200 — SB_19200 (19200 бит/с);• 38400 — SB_38400 (38400 бит/с);• 57600 — SB_57600 (57600 бит/с);• 115200 — SB_115200 (115200 бит/с).
stopBits	Количество стоп-битов: <ul style="list-style-type: none">• 0 — STOP_BITS_1 (1 стоп-бит);• 1 — STOP_BITS_2 (2 стоп-бита).
parity	Четность: <ul style="list-style-type: none">• 0 — PARITY_NONE (нет);• 1 — PARITY_EVEN (четный);• 2 — PARITY_ODD (нечетный).

Возвращаемое значение:

true — успешная инициализация, false — инициализация не удалась.

Примечание. После вызова функции `tkPortInit` работа с портом из основной прошивки прекращается. Чтобы работа с портом из основной прошивки продолжилась, нужно вызвать функцию `tkPortDeinit`.

Пример вызова функции:

```
tkPortInit(SERIAL_PORT_RS232, SB_115200, STOP_BITS_1, PARITY_NONE)
```

tkPortDeinit

Закрытие порта.

Для использования функции подключите файл:

```
#include <agSerial.inc>
```

Формат функции:

tkPortDeinit(index)

Параметры:

index	Индекс порта для закрытия: <ul style="list-style-type: none">• 0 — SERIAL_PORT_RS232 (RS-232);• 1 — SERIAL_PORT_RS485_1 (RS-485-1);• 2 — SERIAL_PORT_RS485_2 (RS-485-2).
--------------	--

Возвращаемое значение:

Нет.

Пример вызова функции:

```
tkPortDeinit(0);
```

tkPortReadPackage

Чтение из порта пакетов, разделенных по временной задержке.

Для использования функции подключите файл:

```
#include <agSerial.inc>
```

Функция читает последовательность байтов из указанного порта. Пакет определяется отсутствием данных в течение заданного интервала времени или размером буфера для получения пакета.

Формат функции:

tkPortReadPackage(index, buf, bufSize, firstByteTimeout, nextByteTimeout, bool:pack=true)

Параметры:

index	Индекс порта: <ul style="list-style-type: none">• 0 — SERIAL_PORT_RS232 (RS-232);• 1 — SERIAL_PORT_RS485_1 (RS-485-1);• 2 — SERIAL_PORT_RS485_2 (RS-485-2).
buf	Указатель на буфер для получения пакета.
bufSize	Размер передаваемого буфера в байтах.
firstByteTimeout	Время ожидания первого байта пакета, в миллисекундах.
nextByteTimeout	Время ожидания прихода bufSize байт, в миллисекундах.
pack	Приемный буфер запакован.

Возвращаемое значение:

Количество прочитанных байтов в пакете (0, если пакет не был получен).

Пример вызова функции:

```
new rxBuf{32};
new bytesRead = tkPortReadPackage(SERIAL_PORT_RS485_1, rxBuf, 32, 100, 100);
if (bytesRead > 0) {
    // Обработать rxBuf
}
```

tkPortWrite

Запись данных в порт.

Для использования функции подключите файл:

```
#include <agSerial.inc>
```

Функция передает указанный буфер данных в открытый последовательный порт.

Формат функции:

tkPortWrite(index, buf, bufSize, packed)

Параметры:

index	Индекс открытого порта: <ul style="list-style-type: none">• 0 — SERIAL_PORT_RS232 (RS-232);• 1 — SERIAL_PORT_RS485_1 (RS-485-1);• 2 — SERIAL_PORT_RS485_2 (RS-485-2).
buf[]	Указатель на буфер с данными для записи.
bufSize	Размер буфера в байтах.
packed	Указатель на то, запакован ли буфер с данными.

Возвращаемое значение:

Нет.

Пример вызова функции:

```
new txBuf{4};  
tkPortWrite(SERIAL_PORT_RS485_1, txBuf, 4);
```

tkPortClearReadBuf

Очистка приемного буфера.

Для использования функции подключите файл:

```
#include <agSerial.inc>
```

Функция удаляет все принятые данные из буфера открытого последовательного порта.

Формат функции:

tkPortClearReadBuf(index)

Параметры:

index	Индекс открытого порта: <ul style="list-style-type: none">• 0 — SERIAL_PORT_RS232 (RS-232);• 1 — SERIAL_PORT_RS485_1 (RS-485-1);• 2 — SERIAL_PORT_RS485_2 (RS-485-2).
--------------	---

Возвращаемое значение:

Нет.

Пример вызова функции:

```
tkPortClearReadBuf(SERIAL_PORT_RS485_1);
```

Функции для работы с сокетами

Список функций	Описание
<u>tkSocketOpen</u>	Открытие сокета и подключение к серверу.
<u>tkSocketSend</u>	Отправка данных в открытый сокет.
<u>tkSocketRecv</u>	Получение данных из открытого сокета.
<u>tkSocketClose</u>	Закрытие сокета.

tkSocketOpen

Открытие сокета и подключение к серверу.

Для использования функции подключите файл:

```
#include <agSocket.inc>
```

Формат функции:

tkSocketOpen (address[], port, timeout)

Параметры:

address	Адрес сервера для подключения (доменное имя или IP-адрес).
port	Порт для подключения.
timeout	Время ожидания соединения.

Возвращаемое значение:

1 — подключение установлено, 0 — подключиться не удалось.

Пример вызова функции:

```
#define PORT 2225
#define TIMEOUT 4000
new address{} = "office.tk-chel.ru";
if ( !tkSocketOpen(address, PORT, TIMEOUT) ) {
    // обработка ошибки открытия сокета
}
```

tkSocketSend

Отправка данных в открытый сокет.

Для использования функции подключите файл:

```
#include <agSocket.inc>
```

Формат функции:

tkSocketSend (src[], bytesToSend, &bytesSent, timeout)

Параметры:

src	Массив данных для отправки.
bytesToSend	Количество байтов, которое нужно отправить.
bytesSent	Выходной параметр: фактически отправленное количество байтов.
timeout	Время ожидания операции (в миллисекундах).

Возвращаемое значение:

1 — данные успешно отправлены, 0 — ошибка отправки.

Пример вызова функции:

```
new buffer{} = "Hello, server!";  
new sent = 0;  
if ( !tkSocketSend(buffer, 14, sent, 5000) ) {  
    // обработка ошибки отправки  
}
```

tkSocketRecv

Получение данных из открытого сокета.

Для использования функции подключите файл:

```
#include <agSocket.inc>
```

Формат функции:

tkSocketRecv (dest[], bufSize, &bytesRcvd, timeout)

Параметры:

dest	Буфер для приема данных.
bufSize	Размер приемного буфера в байтах.
bytesRcvd	Выходной параметр: фактически полученное количество байтов.
timeout	Время ожидания операции (в миллисекундах).

Возвращаемое значение:

1 — данные успешно получены, 0 — ошибка получения.

Примечание. Функция завершится, если был заполнен весь буфер или истекло время ожидания операции.

Пример вызова функции:

```
new recvBuf{256};  
new received = 0;  
if ( tkSocketRecv(recvBuf, 256, received, 5000) ) {  
    // обработка полученных данных  
} else {  
    // обработка ошибки приема  
}
```

tkSocketClose

Закрытие сокета.

Для использования функции подключите файл:

```
#include <agSocket.inc>
```

Формат функции:

tkSocketClose ()

Параметры:

Нет.

Возвращаемое значение:

1 — сокет успешно закрыт, 0 — ошибка закрытия.

Пример вызова функции:

```
if (!tkSocketClose()) {  
    // обработка ошибки закрытия сокета  
}
```

Функции для работы с файлами в SPI памяти

Список функций	Описание
tkNextFileSPI	Получение списка файлов в заданной папке.
tkFileSizeSPI	Получение размера файла.
tkFileReadSPI	Чтение данных из файла с указанным смещением.
tkFileWriteSPI	Запись данных в файл с указанным смещением.
tkFileDeleteSPI	Удаление файла.
tkFileSendSPI	Отправка файла на сервер.
tkFileSendStatusSPI	Получение статуса отправки файла.
tkFileReadMCU	Чтение данных из файла (из флэш памяти микроконтроллера) с указанным смещением.

Список групп параметров	Описание
FRESULT	Возможные результаты операций.

tkNextFileSPI

Получение списка файлов в заданной папке.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkNextFileSPI (rootDir[], lastFile[], res[], resSize, &resultOut)

Параметры:

rootDir	Полный путь до папки, в которой производится поиск (относительно SPI).
lastFile	Последний файл, найденный в процессе поиска.
res	Массив, в который будут записаны путь и имя найденного файла (упакованная строка).
resSize	Размер массива res в байтах.
resultOut	Переменная, в которую будет записан результат операции (см. FRESULT). Необязательный параметр.

Возвращаемое значение:

1 — файл найден, 0 — файлов больше нет.

Пример вызова функции:

```
new rootDir{} = "";  
new lastFile{} = "";  
new result{256};  
if ( tkNextFileSPI (rootDir, lastFile, result, 256) ) {  
    // обработка найденного файла  
}
```

tkFileSizeSPI

Получение размера файла.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkFileSizeSPI (filename[], &resultOut)

Параметры:

filename	Имя файла (относительно SPI).
resultOut	Переменная, в которую будет записан результат операции (см. FRESULT). Необязательный параметр.

Возвращаемое значение:

Размер файла в байтах, -1 — файл не найден или ошибка.

Пример вызова функции:

```
new filename{} = "test.txt";  
new size = tkFileSizeSPI (filename);  
if ( size > 0 ) {  
    // обработка размера файла  
}
```

tkFileReadSPI

Чтение данных из файла с указанным смещением.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkFileReadSPI (filename[], buf[], bufSize, offset = 0, &resultOut)

Параметры:

filename	Имя файла (относительно SPI).
buf	Буфер для данных.
bufSize	Размер буфера (ограничен 1024 байтами).
offset	Смещение в файле (по умолчанию 0).
resultOut	Переменная, в которую будет записан результат операции (см. FRESULT). Необязательный параметр.

Возвращаемое значение:

Количество прочитанных байт, 0 — ошибка чтения из файла.

Примечание. За один вызов функция может прочитать не более 1024 байт.

Пример вызова функции:

```
new filename{} = "data.bin";
new buffer{1024};
new bytesRead = tkFileReadSPI (filename, buffer, 1024, 0);
if ( bytesRead > 0 ) {
    // обработка прочитанных данных
}
```

tkFileWriteSPI

Запись данных в файл с указанным смещением.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkFileWriteSPI (filename[], buf[], bufSize, offset, maxFileLen, &resultOut)

Параметры:

filename	Имя файла (относительно SPI).
buf	Буфер с данными.
bufSize	Размер данных для записи (ограничен 1024 байтами).
offset	Смещение в файле.
maxFileLen	Размер создаваемого файла (ограничен размером общего доступного пространства для файлов на SPI Flash).
resultOut	Переменная, в которую будет записан результат операции (см. FRESULT). Необязательный параметр.

Возвращаемое значение:

Количество записанных байт, 0 — ошибка записи в файл.

Примечание. Особенности SPI Flash: если файл еще не существует, то место для него заранее выделяется через *maxFileLen*.

Примечание. За один вызов функция может записать не более 1024 байт.

Пример вызова функции:

```
new filename{} = "output.txt";
new data{} = "Hello, World!";
new bytesWritten = tkFileWriteSPI (filename, data, 13, 0, 1024);
if ( bytesWritten > 0 ) {
    // данные успешно записаны
}
```

tkFileDeleteSPI

Удаление файла.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkFileDeleteSPI (filename[], &resultOut)

Параметры:

filename	Имя файла (относительно SPI).
resultOut	Переменная, в которую будет записан результат операции (см. FRESULT). Необязательный параметр.

Возвращаемое значение:

1 — файл удален, 0 — ошибка удаления файла.

Примечание. Чтобы избежать исчерпания ресурсов перезаписи внутренней флеш-памяти, вызов функции `tkFileDeleteSPI` из Т.Скрипт ограничен по времени: не чаще 10 раз в минуту, 20 раз в час и 30 раз в день.

Пример вызова функции:

```
new filename{} = "temp.txt";  
if ( tkFileDeleteSPI (filename) ) {  
    // файл успешно удален  
} else {  
    // ошибка удаления файла  
}
```

tkFileSendSPI

Отправка файла на сервер.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkFileSendSPI (filename[], deleteAfter = 1)

Параметры:

filename	Имя файла (относительно SPI).
deleteAfter	1 — удалить файл после отправки, 0 — не удалять файл.

Возвращаемое значение:

1 — задача отправки поставлена, 0 — поставить задачу отправки не удалось.

Пример вызова функции:

```
new filename{} = "mydata.bin";  
if ( tkFileSendSPI (filename, 1) ) {  
    // задача отправки поставлена  
}
```

tkFileSendStatusSPI

Получение статуса отправки файла.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkFileSendStatusSPI (filename[], outStatus[.state, .filesize, .progress])

Параметры:

filename	Имя файла (относительно SPI).
outStatus	Массив структур, в который будет записан статус: <ul style="list-style-type: none">• .state — текущее состояние отправки:<ul style="list-style-type: none">• 0 — нет файлов на отправку;• 1 — есть файлы на отправку;• 3 — идет передача файла на сервер;• 4 — передача файла на сервер успешно завершена;• 5 — передача файла на сервер завершена с ошибкой.• .filesize — размер файла в байтах.• .progress — размер уже отправленного файла в процентах.

Возвращаемое значение:

Нет.

Пример вызова функции:

```
new filename{} = "mydata.bin";
new status[.state, .filesize, .progress];

tkFileSendStatusSPI(filename, status);
printf("state=%d filesize=%d progress=%d%%\r\n", status.state, status.filesize, status.progress);
```

tkFileReadMCU

Чтение данных из файла (из флэш памяти микроконтроллера) с указанным смещением.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkFileReadMCU (filename[], buf[], bufSize, offset = 0, &resultOut)

Параметры:

filename	Имя файла (относительно MCU).
buf	Буфер для данных.
bufSize	Размер буфера (ограничен 1024 байтами).
offset	Смещение в файле (по умолчанию 0).
resultOut	Переменная, в которую будет записан результат операции (см. FRESULT). Необязательный параметр.

Возвращаемое значение:

Количество прочитанных байт, 0 — ошибка чтения из файла.

Примечание. За один вызов функция может прочитать не более 1024 байт.

Пример вызова функции:

```
new filename{} = "data.bin";  
new buffer{1024};  
new bytesRead = tkFileReadMCU (filename, buffer, 1024, 0);  
if ( bytesRead > 0 ) {  
    // обработка прочитанных данных  
}
```

FRESULT

Возможные результаты операций.

FR_OK = 0	0 — Успешно.
FR_DISK_ERR	1 — Возникла критическая ошибка на низком уровне дискового ввода-вывода.
FR_INT_ERR	2 — Сбой утверждения (Assertion failed).
FR_NOT_READY	3 — Физический диск не может работать.
FR_NO_FILE	4 — Файл не найден.
FR_NO_PATH	5 — Путь не найден.
FR_INVALID_NAME	6 — Неверный формат имени пути.
FR_DENIED	7 — Доступ запрещен или каталог заполнен.
FR_EXIST	8 — Доступ запрещен.
FR_INVALID_OBJECT	9 — Неверный объект файла/каталога.
FR_WRITE_PROTECTED	10 — Физический диск защищен от записи.
FR_INVALID_DRIVE	11 — Неверный номер логического диска.
FR_NOT_ENABLED	12 — Для тома нет рабочей области.
FR_NO_FILESYSTEM	13 — Нет действительного тома FAT.
FR_MKFS_ABORTED	14 — Выполнение функции <code>f_mkfs()</code> прервано из-за проблемы.
FR_TIMEOUT	15 — Не удалось получить доступ к тому в течение заданного периода.
FR_LOCKED	16 — Операция отклонена в соответствии с политикой общего доступа к файлам.
FR_NOT_ENOUGH_CORE	17 — Не удалось выделить рабочий буфер для длинных имен файлов (LFN).
FR_TOO_MANY_OPEN_FILES	18 — Количество открытых файлов > <code>FF_FS_LOCK</code> .
FR_INVALID_PARAMETER	19 — Указанный параметр недействителен.

Функции для работы с файлами в SD/RAM памяти

Следующие команды относятся к работе с файловой системой на SD-карте и во внутренней оперативной памяти устройства (RAM). Если карта памяти установлена, то функции будут работать с ней. Если карта памяти не установлена, то работа с файловой системой будет осуществляться в RAM. Обратите внимание, что при потере питания RAM очищается. Проверка наличия карты памяти осуществляется при подаче питания на устройство.

Список функций	Описание
<u>tkNextFile</u>	Получение списка файлов в заданной папке.
<u>tkNextDir</u>	Получение списка папок, вложенных в заданную папку.
<u>tkFileSize</u>	Получение размера файла.
<u>tkFileRead</u>	Чтение данных из файла.
<u>tkFileWrite</u>	Запись данных в файл.
<u>tkFileDelete</u>	Удаление файла.
<u>tkFileSend</u>	Отправка файла на сервер.
<u>tkFileSendStatus</u>	Получение статуса отправки файла.
<u>tkIsSdCardInserted</u>	Проверка наличия карты памяти в устройстве.

tkNextFile

Получение списка файлов в заданной папке.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkNextFile (rootDir[], lastFile[], res[], resSize, &resultOut)

Параметры:

rootDir	Полный путь до папки, в которой производится поиск.
lastFile	Последний файл, найденный в процессе поиска.
res	Массив, в который будут записаны путь и имя найденного файла (упакованная строка).
resSize	Размер массива res в байтах.
resultOut	Переменная, в которую будет записан результат операции (см. FRESULT). Необязательный параметр.

Возвращаемое значение:

1 — файл найден, 0 — файлов больше нет.

Пример вызова функции:

```
new rootDir{} = "";  
new lastFile{} = "";  
new result{256};  
if ( tkNextFile (rootDir, lastFile, result, 256) ) {  
    // обработка найденного файла  
}
```

tkNextDir

Получение списка папок, вложенных в заданную папку.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkNextDir (rootDir[], lastDir[], res[], resSize, &resultOut)

Параметры:

rootDir	Полный путь до папки, в которой производится поиск.
lastDir	Последняя папка, найденная в процессе поиска.
res	Массив, в который будут записаны путь и имя найденной папки (упакованная строка).
resSize	Размер массива res в байтах.
resultOut	Переменная, в которую будет записан результат операции (см. FRESULT). Необязательный параметр.

Возвращаемое значение:

1 — папка найдена, 0 — папок больше нет.

Пример вызова функции:

```
new rootDir{} = "";  
new lastDir{} = "";  
new result{256};  
if ( tkNextDir (rootDir, lastDir, result, 256) ) {  
    // обработка найденной папки  
}
```

tkFileSize

Получение размера файла.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkFileSize (filename[], &resultOut)

Параметры:

filename	Имя файла.
resultOut	Переменная, в которую будет записан результат операции (см. FRESULT). Необязательный параметр.

Возвращаемое значение:

Размер файла в байтах, -1 — файл не найден или ошибка.

Пример вызова функции:

```
new filename{} = "test.txt";  
new size = tkFileSize (filename);  
if ( size > 0 ) {  
    // обработка размера файла  
}
```

tkFileRead

Чтение данных из файла.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkFileRead (filename[], buf[], bufSize, offset = 0, &resultOut)

Параметры:

filename	Полный или относительный путь.
buf	Буфер для данных.
bufSize	Размер буфера (ограничен 1024 байтами).
offset	Смещение в файле (по умолчанию 0).
resultOut	Переменная, в которую будет записан результат операции (см. FRESULT). Необязательный параметр.

Возвращаемое значение:

Количество прочитанных байт, 0 — ошибка чтения файла.

Примечание. За один вызов функция может прочитать не более 1024 байт.

Пример вызова функции:

```
new filename{} = "data.bin";  
new buffer{1024};  
new bytesRead = tkFileRead (filename, buffer, 1024, 0);  
if ( bytesRead > 0 ) {  
    // обработка прочитанных данных  
}
```

tkFileWrite

Запись данных в файл.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkFileWrite (filename[], buf[], bufSize, offset, &resultOut)

Параметры:

filename	Полный или относительный путь.
buf	Буфер с данными.
bufSize	Размер данных для записи (ограничен 1024 байтами).
offset	Смещение в файле.
resultOut	Переменная, в которую будет записан результат операции (см. FRESULT). Необязательный параметр.

Возвращаемое значение:

Количество записанных байт, 0 — ошибка записи в файл.

Примечание. За один вызов функция может записать не более 1024 байт.

Пример вызова функции:

```
new filename{} = "output.txt";
new data{} = "Hello, World!";
new bytesWritten = tkFileWrite (filename, data, 13, 0);
if ( bytesWritten > 0 ) {
    // данные успешно записаны
}
```

tkFileDelete

Удаление файла.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkFileDelete (filename[], &resultOut)

Параметры:

filename	Имя файла.
resultOut	Переменная, в которую будет записан результат операции (см. FRESULT). Необязательный параметр.

Возвращаемое значение:

1 — файл удален, 0 — ошибка удаления файла.

Пример вызова функции:

```
new filename{} = "temp.txt";  
if ( tkFileDelete (filename) ) {  
    // файл успешно удален  
} else {  
    // ошибка удаления файла  
}
```

tkFileSend

Отправка файла на сервер.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkFileSend (filename[], deleteAfter = 1)

Параметры:

filename	Имя файла.
deleteAfter	1 — удалить файл после отправки, 0 — не удалять файл.

Возвращаемое значение:

1 — задача отправки поставлена, 0 — поставить задачу отправки не удалось.

Пример вызова функции:

```
new filename{} = "mydata.bin";  
if ( tkFileSend (filename, 1) ) {  
    // задача отправки поставлена  
}
```

tkFileSendStatus

Получение статуса отправки файла.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkFileSendStatus (filename[], outStatus[.state, .filesize, .progress])

Параметры:

filename	Имя файла.
outStatus	Массив структур, в который будет записан статус: <ul style="list-style-type: none">• .state — текущее состояние отправки:<ul style="list-style-type: none">• 0 — нет файлов на отправку;• 1 — есть файлы на отправку;• 3 — идет передача файла на сервер;• 4 — передача файла на сервер успешно завершена;• 5 — передача файла на сервер завершена с ошибкой.• .filesize — размер файла в байтах.• .progress — размер уже отправленного файла в процентах.

Возвращаемое значение:

Нет.

Пример вызова функции:

```
new filename{} = "mydata.bin";
new status[.state, .filesize, .progress];

tkFileSendStatus(filename, status);
printf("state=%d filesize=%d progress=%d%%\r\n", status.state, status.filesize, status.progress);
```

tkIsSdCardInserted

Проверка наличия карты памяти в устройстве.

Для использования функции подключите файл:

```
#include <agFile.inc>
```

Формат функции:

tkIsSdCardInserted()

Параметры:

Нет.

Возвращаемое значение:

1 — карта памяти установлена, 0 — карта памяти не установлена.

Пример вызова функции:

```
if (tkIsSdCardInserted())  
{  
    printf("SD-Card is inserted\r\n");  
}
```

Функции для работы с BLE

Данные функции доступны на прошивке BLE-чипа версии AGBT-01.14 и выше. Для проверки версии прошивки и ее обновления используйте следующие управляющие команды:

- GBLEVERSION — запрос версии прошивки BLE-чипа.
- BLEUPDATE=1 — обновление BLE-чипа на последнюю стабильную версию.

Список функций	Описание
tkBleStart	Запуск BLE в режиме работы скриптов.
tkBleAddAddrsFromConf	Добавление в белый список адресов из конфигурации устройства.
tkBleAddAddrInWhiteList	Добавление MAC-адреса в белый список.
tkBleAddNameInWhiteList	Добавление маски имени в белый список.
tkBleClearWhiteList	Очистка белого списка устройств.
tkBleReceive	Прием Bluetooth-сообщения.
tkBleWriteRecord	Запись Bluetooth-данных в бинарный файл устройства.
tkBleSetRecordPeriod	Установка периода записи Bluetooth-данных.

tkBleStart

Запуск BLE в режиме работы скриптов.

Для использования функции подключите файл:

```
#include <agBle.inc>
```

Формат функции:

```
tkBleStart()
```

Возвращаемое значение:

Нет.

Пример вызова функции:

```
tkBleStart();
```

tkBleAddAddrsFromConf

Добавление в белый список адресов из конфигурации устройства.

Для использования функции подключите файл:

```
#include <agBle.inc>
```

Формат функции:

```
tkBleAddAddrsFromConf()
```

Возвращаемое значение:

Количество добавленных адресов.

Пример вызова функции:

```
new addr_counter = tkBleAddAddrsFromConf();
```

Примечание. Команда добавит датчики, адреса которых записаны в конфигурацию устройства.
(SCRIPTCONFSTRING=MAC1:AABBCCDDEE01,MAC2:AABBCCDDEE02,...,MAC8:AABBCCDDEE08;)

tkBleAddAddrInWhiteList

Добавление MAC-адреса в белый список.

Для использования функции подключите файл:

```
#include <agBle.inc>
```

Формат функции:

```
tkBleAddAddrInWhiteList(addr[BLEADDR])
```

Параметры:

addr	MAC-адрес устройства.
-------------	-----------------------

Возвращаемое значение:

Нет.

Пример вызова функции:

```
new mac_addr[BLEADDR];  
mac_addr.addr{0} = 0x1A;  
mac_addr.addr{1} = 0x2B;  
mac_addr.addr{2} = 0x3C;  
mac_addr.addr{3} = 0x4D;  
mac_addr.addr{4} = 0x5E;  
mac_addr.addr{5} = 0x6F;  
tkBleAddAddrInWhiteList(mac_addr);
```

Примечание. Доступно добавление до 8 адресов.

tkBleAddNameInWhiteList

Добавление маски имени в белый список.

Для использования функции подключите файл:

```
#include <agBle.inc>
```

Формат функции:

```
tkBleAddNameInWhiteList(const name[])
```

Параметры:

name	Маска имени устройства (строка).
------	----------------------------------

Возвращаемое значение:

Нет.

Пример вызова функции:

```
tkBleAddNameInWhiteList("abc");
```

Примечание. Доступно добавление до 8 масок.

Примечание. Чтобы фильтр сработал, рекламный пакет должен удовлетворять следующим условиям:

1. Он должен содержать структуру данных с типом `BLE_GAP_COMPLETE_LOCAL_NAME (0x09)`.
2. Значение в этой структуре должно начинаться с заданной маски (например, пакет с именем «abc123» будет принят по маске «abc»).

tkBleClearWhiteList

Очистка белого списка устройств.

Для использования функции подключите файл:

```
#include <agBle.inc>
```

Формат функции:

```
tkBleClearWhiteList()
```

Возвращаемое значение:

Нет.

Пример вызова функции:

```
tkBleClearWhiteList();
```

tkBleReceive

Прием Bluetooth-сообщения.

Для использования функции подключите файл:

```
#include <agBle.inc>
```

Формат функции:

```
tkBleReceive(msg[BLEMSG])
```

Параметры:

msg	Полученное сообщение.
------------	-----------------------

Возвращаемое значение:

Принято новое сообщение (1 — сообщение получено, 0 — сообщение не получено).

Пример вызова функции:

```
new ble_msg[BLEMSG];  
tkBleReceive(ble_msg);
```

tkBleWriteRecord

Запись Bluetooth-данных в бинарный файл устройства.

Для использования функции подключите файл:

```
#include <agBle.inc>
```

Формат функции:

```
tkBleWriteRecord(addr[BLEADDR], data{}, len)
```

Параметры:

addr	MAC-адрес устройства.
data	Данные.
len	Длина данных.

Возвращаемое значение:

Статус записи данных (1 — данные записаны, 0 — данные не записаны).

Пример вызова функции:

```
new ble_msg[BLEMSG];
if (tkBleReceive(ble_msg))
{
    tkBleWriteRecord(ble_msg.addr, ble_msg.data, ble_msg.len);
}
```

Примечание. Тип записи — 0xFA0001.

Примечание. Работает только для датчиков, адреса которых записаны в конфигурацию устройства и добавлены с помощью функции `tkBleAddAddrsFromConf()`.

tkBleSetRecordPeriod

Установка периода записи Bluetooth-данных.

Для использования функции подключите файл:

```
#include <agBle.inc>
```

Формат функции:

tkBleSetRecordPeriod(period)

Параметры:

period	Период записи, 5...300 с.
---------------	---------------------------

Возвращаемое значение:

Нет.

Пример вызова функции:

```
new period = 10;  
tkBleSetRecordPeriod(period);
```

Стандартные функции Pawn

Список функций	Описание
<u>printf</u>	Форматированный вывод текста в консоль сервера/клиента.
<u>heapspace</u>	Получение информации о количестве свободного пространства в секции стека/кучи скрипта.
<u>funcidx</u>	Проверка существования public-функции в скрипте.
<u>numargs</u>	Получение количества аргументов, переданных в текущую функцию.
<u>getarg</u>	Получение значения аргумента функции по его индексу.
<u>setarg</u>	Установка значения аргумента функции по указанному индексу.
<u>tolower</u>	Преобразование символа в нижний регистр.
<u>toupper</u>	Преобразование символа в верхний регистр.
<u>swapchars</u>	Перестановка байтов ячейки в обратном порядке.
<u>random</u>	Генерация псевдослучайного числа в заданном диапазоне.
<u>min</u>	Определение наименьшего из двух чисел.
<u>max</u>	Определение наибольшего из двух чисел.
<u>clamp</u>	Приведение числа к указанному диапазону.

printf

Форматированный вывод текста в консоль сервера/клиента.

Для использования функции подключите файл:

```
#include <console.inc>
```

Функция позволяет форматировать данные в строку (преобразовывать данные в строковый эквивалент и вставлять их в форматный текст) и выводить получившийся текст в консоль и в лог сервера.

Формат функции:

```
printf(const format[], {Fixed,}_:...)
```

Параметры:

format	Строка формата, определяющая вывод (аналогично стандартной printf в C).
...	Переменное количество аргументов, подставляемых в строку формата.

Возвращаемое значение:

Нет.

Спецификаторы формата:

- %d, %i — целое число (integer).
- %f — число с плавающей точкой (float).
- %s — строка (string).
- %x, %X — шестнадцатеричное число.
- %c — символ (character).
- %% — символ процента.

Пример вызова функции:

```
new hour, minute, second;  
gettime(hour, minute, second);  
printf("Текущее время: %d:%02d:%02d", hour, minute, second);
```

Примечание. В Rawn нет разницы между %d и %i.

Примечание. Для вывода чисел с плавающей точкой требуется явное приведение: `printf("%f", float(value))`.

Примечание. Максимальная длина выводимой строки — 1024 символа.

heap space

Получение информации о количестве свободного пространства в секции стека/кучи скрипта.

Для использования функции подключите файл:

```
#include <core.inc>
```

Формат функции:

heap space()

Возвращаемое значение:

Размер свободного пространства (в байтах) в секции стека/кучи.

Пример вызова функции:

```
new heap_free = heap space();  
printf("В секции стека/кучи свободно %d байт", heap_free);
```

funcidx

Проверка существования public-функции в скрипте.

Для использования функции подключите файл:

```
#include <core.inc>
```

Формат функции:

```
funcidx(const name[])
```

Параметры:

name	Название функции.
------	-------------------

Возвращаемое значение:

Индекс функции в таблице public-функций. И -1, если функции с указанным названием не существует.

Пример вызова функции:

```
new idx = funcidx("MyFunction");  
// Вывод: "Функция MyFunction() существует (ID 0)"  
printf("Функция MyFunction() %существует (ID %d)", (idx == -1) ? "не " : "", idx);
```

Примечание. Проверяемая функция должна быть объявлена с атрибутом public либо ее название должно начинаться с символа «@».

Примечание. Индексы public-функций начинаются с 0.

numargs

Получение количества аргументов, переданных в текущую функцию.

Для использования функции подключите файл:

```
#include <core.inc>
```

Формат функции:

```
numargs()
```

Возвращаемое значение:

Количество аргументов, переданных в текущую функцию (0, если аргументов нет).

Пример вызова функции:

```
new n = numargs();  
printf("Количество аргументов: %d", n);  
for (new i = 0; i < n; ++i)  
    printf(" %d", getarg(i));
```

getarg

Получение значения аргумента функции по его индексу.

Для использования функции подключите файл:

```
#include <core.inc>
```

Формат функции:

getarg(arg, index=0)

Параметры:

arg	Индекс аргумента (начиная с 0).
index	Индекс элемента, если указанный аргумент является массивом (опционально, по умолчанию 0).

Возвращаемое значение:

Значение запрошенного аргумента в виде целого числа.

Пример вызова функции:

```
PrintArray(a[], size = sizeof(a))
{
    printf("Размер массива: %d", size);
    // Выведем все элементы массива
    for (new i = 0; i < size; ++i)
    {
        // То же самое, что и "printf(" %d", a[i]);"
        // На этот раз мы работаем с массивом, поэтому параметр index нужно указывать
        printf(" %d", getarg(0, i));
    }
}
```

setarg

Установка значения аргумента функции по указанному индексу.

Для использования функции подключите файл:

```
#include <core.inc>
```

Формат функции:

getarg(arg, index=0, value)

Параметры:

arg	Индекс аргумента (начиная с 0).
index	Индекс элемента, если указанный аргумент является массивом (параметр можно пропустить с помощью знака «_»).
value	Значение, которое нужно сохранить в аргументе.

Возвращаемое значение:

1 — значение установлено, 0 — указан неправильный индекс аргумента.

Пример вызова функции:

```
// Увеличим значение всех аргументов функции на 1  
// Обратите внимание: в вызове setarg() параметр index пропущен (знак "_")  
for (new i = 0, n = numargs(); i < n; ++i)  
    setarg(i, _, getarg(i) + 1);
```

tolower

Преобразование символа в нижний регистр.

Для использования функции подключите файл:

```
#include <core.inc>
```

Формат функции:

tolower(c)

Параметры:

c	Символ, который нужно преобразовать в нижний регистр.
---	---

Возвращаемое значение:

Аналог символа **c** в нижнем регистре, если таковой существует. Иначе — исходное значение **c**.

Пример вызова функции:

```
static string[] = "THE QUICK FOX JUMPS OVER THE LAZY DOG";  
for (new i = 0; i < sizeof(string); ++i)  
    string[i] = tolower(string[i]);  
// Вывод: "the quick fox jumps over the lazy dog"  
print(string);
```

Примечание. Функция работает только с латинскими символами (A–Z).

Примечание. Функция не изменяет символы в нижнем регистре.

Примечание. Функция возвращает оригинальный символ, если он не является буквой верхнего регистра.

toupper

Преобразование символа в верхний регистр.

Для использования функции подключите файл:

```
#include <core.inc>
```

Формат функции:

toupper(c)

Параметры:

c	Символ, который нужно преобразовать в верхний регистр.
---	--

Возвращаемое значение:

Аналог символа **c** в нижнем регистре, если таковой существует. Иначе — исходное значение **c**.

Пример вызова функции:

```
static string[] = "the quick fox jumps over the lazy dog";  
for (new i = 0; i < sizeof(string); ++i)  
    string[i] = toupper(string[i]);  
// Вывод: "THE QUICK FOX JUMPS OVER THE LAZY DOG"  
print(string);
```

Примечание. Функция работает только с латинскими символами (A–Z).

Примечание. Функция не изменяет символы в верхнем регистре.

Примечание. Функция возвращает оригинальный символ, если он не является буквой нижнего регистра.

swapchars

Перестановка байтов ячейки в обратном порядке.

Для использования функции подключите файл:

```
#include <core.inc>
```

Формат функции:

swapchars(c)

Параметры:

c	Значение, байты которого нужно обменять местами.
---	--

Возвращаемое значение:

Значение аргумента c с байтами, переставленными в обратном порядке.

Пример вызова функции:

```
// Вывод: "swapchars(0x11223344): 0x44332211"  
new n = 0x11223344;  
printf("swapchars(0x%08x): 0x%08x", n, swapchars(n));
```

random

Генерация псевдослучайного числа в заданном диапазоне.

Для использования функции подключите файл:

```
#include <core.inc>
```

Формат функции:

random(max)

Параметры:

max	Верхняя граница диапазона для случайного числа.
-----	---

Возвращаемое значение:

Псевдослучайное число от 0 до (max – 1).

Пример вызова функции:

```
new r = random(4);
switch (r)
{
  case 0:
    print("Ноль");
  case 1:
    print("Один");
  case 2:
    print("Два");
  default:
    printf("%d", r);
}
```

Примечание. Если параметр **max** равен нулю, то функция примет это как отсутствие границ диапазона, в результате чего сгенерирует число от 0 до максимального значения, которое может вместить ячейка `RawN` (`cellmax`).

Примечание. Функция работает на основе линейного конгруэнтного метода с диапазоном и периодом равными 2^{31} . Генераторы псевдослучайных чисел на основе этого метода подвержены серийной корреляции и непригодны для решения задач, требующих высокого качества случайности чисел (например, в криптографии), однако подходят для многих других задач.

min

Определение наименьшего из двух чисел.

Для использования функции подключите файл:

```
#include <core.inc>
```

Формат функции:

min(value1, value2)

Параметры:

value1	Первое число для сравнения (integer, float).
value2	Второе число для сравнения (integer, float).

Возвращаемое значение:

Наименьшее значение между **value1** и **value2**.

Пример вызова функции:

```
// Вывод: "Из чисел 3 и 4 наименьшее - 3"  
new a = 3, b = 4;  
printf("Из чисел %d и %d наименьшее - %d", a, b, min(a, b));
```

max

Определение наибольшего из двух чисел.

Для использования функции подключите файл:

```
#include <core.inc>
```

Формат функции:

max(value1, value2)

Параметры:

value1	Первое число для сравнения (integer, float).
value2	Второе число для сравнения (integer, float).

Возвращаемое значение:

Наибольшее значение между **value1** и **value2**.

Пример вызова функции:

```
// Вывод: "Из чисел 3 и 4 наибольшее - 4"  
new a = 3, b = 4;  
printf("Из чисел %d и %d наибольшее - %d", a, b, max(a, b));
```

clamp

Приведение числа к указанному диапазону.

Для использования функции подключите файл:

```
#include <core.inc>
```

Формат функции:

clamp(value, min=cellmin, max=cellmax)

Параметры:

value	Ограничиваемое значение (integer, float).
min	Нижняя граница диапазона (по умолчанию cellmin).
max	Верхняя граница диапазона (по умолчанию cellmax).

Возвращаемое значение:

Значение аргумента **value**, если оно находится в диапазоне от **min** до **max**.
Если **value** < **min**, то возвращает значение **min**. Иначе (**value** > **max**) — значение **max**.

Пример вызова функции:

```
new limited = clamp(150, 0, 100); // Вернет 100  
new Float:angle = clamp(450.0, 0.0, 360.0); // Вернет 360.0  
new autoClamped = clamp(9999); // Вернет cellmax
```

Стандартные функции Prawn для чисел с фиксированной запятой

Список функций	Описание
<u>fixed</u>	Преобразование числа в формат с фиксированной запятой.
<u>strfixed</u>	Создание неизменяемой строки с фиксированным содержимым.
<u>fmul</u>	Умножение чисел с фиксированной запятой.
<u>fdiv</u>	Деление числа с фиксированной запятой.
<u>ffract</u>	Извлечение дробной части числа с фиксированной запятой.
<u>fround</u>	Округление числа с фиксированной запятой согласно указанному методу.
<u>fpower</u>	Возведение числа с фиксированной запятой в указанную степень.
<u>fsqroot</u>	Вычисление квадратного корня из числа с фиксированной запятой.
<u>fabs</u>	Получение абсолютного значения числа с фиксированной запятой.

fixed

Преобразование числа в формат с фиксированной запятой.

Для использования функции подключите файл:

```
#include <fixed.inc>
```

Формат функции:

Fixed:fixed(value)

Параметры:

value	Число для преобразования (целое, дробное или строка).
-------	---

Возвращаемое значение:

Значение с фиксированной запятой.

Пример вызова функции:

```
new Fixed:a = Fixed:fixed(10); // Целое число  
new Fixed:b = Fixed:fixed(3.14); // Дробное число  
new Fixed:c = Fixed:fixed("2.718"); // Строковое представление
```

strfixed

Создание неизменяемой строки с фиксированным содержимым.

Для использования функции подключите файл:

```
#include <fixed.inc>
```

Формат функции:

Fixed:strfixed(const string[])

Параметры:

string[]	Строка для фиксации (массив символов).
-----------------	--

Возвращаемое значение:

Указатель на неизменяемую строку.

Пример вызова функции:

```
new Fixed:message = strfixed("Hello World");  
printf("%s", _:message); // Вывод неизменяемой строки
```

fmul

Умножение чисел с фиксированной запятой.

Для использования функции подключите файл:

```
#include <fixed.inc>
```

Формат функции:

Fixed:fmul(Fixed:oper1, Fixed:oper2)

Параметры:

oper1	Первый множитель (число с фиксированной запятой).
oper2	Второй множитель (число с фиксированной запятой).

Возвращаемое значение:

Результат умножения двух чисел с фиксированной запятой.

Пример вызова функции:

```
new Fixed:result = Fixed:fmul(Fixed:1.5, Fixed:2.0); // вернет Fixed:3.0  
new Fixed:price = Fixed:fmul(Fixed:10.99, Fixed:0.85); // расчет скидки
```

fdiv

Деление числа с фиксированной запятой.

Для использования функции подключите файл:

```
#include <fixed.inc>
```

Формат функции:

Fixed:fdiv(Fixed:dividend, Fixed:divisor)

Параметры:

dividend	Делимое (число с фиксированной запятой).
divisor	Делитель (число с фиксированной запятой). Не должен быть равен 0.

Возвращаемое значение:

Результат деления числа с фиксированной запятой.

Пример вызова функции:

```
new Fixed:result = Fixed:fdiv(Fixed:3.0, Fixed:2.0); // вернет Fixed:1.5  
new Fixed:ratio = Fixed:fdiv(Fixed:100.0, Fixed:3.0); // вернет Fixed:33.333...
```

ffract

Извлечение дробной части числа с фиксированной запятой.

Для использования функции подключите файл:

```
#include <fixed.inc>
```

Формат функции:

Fixed:ffract(Fixed:value)

Параметры:

value	Число с фиксированной запятой для извлечения дробной части.
--------------	---

Возвращаемое значение:

Дробная часть числа в диапазоне [0.0, 1.0) с фиксированной точностью.

Пример вызова функции:

```
new Fixed:num = Fixed:3.14159;  
new Fixed:fraction = Fixed:ffract(num); // вернет Fixed:0.14159  
new Fixed:whole = num - fraction; // целая часть 3.0
```

fround

Округление числа с фиксированной запятой согласно указанному методу.

Для использования функции подключите файл:

```
#include <fixed.inc>
```

Формат функции:

```
fround(value, fround_method:method=fround_round)
```

Параметры:

value	Число с фиксированной запятой для округления.
method	Метод округления (по умолчанию fround_round — математическое округление).

Возвращаемое значение:

Округленное значение с фиксированной запятой.

Пример вызова функции:

```
new Fixed:num = Fixed:3.6;  
new Fixed:r1 = fround(num); // 4.0 (по умолчанию математическое округление)
```

fpower

Возведение числа с фиксированной запятой в указанную степень.

Для использования функции подключите файл:

```
#include <fixed.inc>
```

Формат функции:

Fixed:fpower(Fixed:value, exponent)

Параметры:

value	Основание степени (число с фиксированной запятой).
exponent	Показатель степени (целое число).

Возвращаемое значение:

Результат возведения в степень с фиксированной запятой.

Пример вызова функции:

```
new Fixed:base = Fixed:2.5;  
new Fixed:result1 = Fixed:fpower(base, 2); // 6.25 (2.5 в квадрате)  
new Fixed:result2 = Fixed:fpower(base, 3); // 15.625 (2.5 в кубе)  
new Fixed:result3 = Fixed:fpower(Fixed:10.0, -1); // 0.1 (10 в степени -1)
```

fsqroot

Вычисление квадратного корня из числа с фиксированной запятой.

Для использования функции подключите файл:

```
#include <fixed.inc>
```

Формат функции:

Fixed:fsqroot(Fixed:value)

Параметры:

value	Число с фиксированной запятой для извлечения корня. Не должно быть отрицательным.
--------------	---

Возвращаемое значение:

Квадратный корень числа с фиксированной запятой.

Пример вызова функции:

```
new Fixed:num1 = fixed(25.0);  
new Fixed:sqrt1 = Fixed:fsqroot(num1); // 5.0
```

```
new Fixed:num2 = fixed(2.0);  
new Fixed:sqrt2 = Fixed:fsqroot(num2); // ~1.4142
```

```
new Fixed:num3 = fixed(0.25);  
new Fixed:sqrt3 = Fixed:fsqroot(num3); // 0.5
```

fabs

Получение абсолютного значения числа с фиксированной запятой.

Для использования функции подключите файл:

```
#include <fixed.inc>
```

Формат функции:

Fixed:fabs(Fixed:value)

Параметры:

value	Число с фиксированной запятой для вычисления абсолютного значения.
--------------	--

Возвращаемое значение:

Абсолютное значение (модуль) числа с фиксированной запятой.

Пример вызова функции:

```
new Fixed:num1 = fixed(3.14);  
new Fixed:abs1 = Fixed:fabs(num1); // 3.14  
  
new Fixed:num2 = fixed(-2.71);  
new Fixed:abs2 = Fixed:fabs(num2); // 2.71  
  
new Fixed:num3 = fixed(0.0);  
new Fixed:abs3 = Fixed:fabs(num3); // 0.0
```

Стандартные функции Pwn для чисел с плавающей запятой

Список функций	Описание
float	Преобразование числа в формат с плавающей запятой.
strfloat	Преобразование строки в число с плавающей запятой.
floatmul	Умножение чисел с плавающей запятой.
floatdiv	Деление числа с плавающей запятой.
floatadd	Сложение чисел с плавающей запятой.
floatsub	Вычитание чисел с плавающей запятой.
floatfract	Извлечение дробной части числа с плавающей запятой.
floatround	Округление числа с плавающей запятой согласно указанному методу.
floatcmp	Сравнение двух чисел с плавающей запятой.
floatsgroot	Вычисление квадратного корня из числа с плавающей запятой.
floatpower	Возведение числа с плавающей запятой в указанную степень.
floatlog	Вычисление логарифма числа с плавающей запятой по указанному основанию.
floatsin	Вычисление синуса угла для чисел с плавающей запятой.
floatcos	Вычисление косинуса угла для чисел с плавающей запятой.
floattan	Вычисление тангенса угла для чисел с плавающей запятой.
floatabs	Получение абсолютного значения числа с плавающей запятой.

float

Преобразование числа в формат с плавающей запятой.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

Float:float(value)

Параметры:

value	Число для преобразования (целое, строка или число с фиксированной запятой).
--------------	---

Возвращаемое значение:

Значение с плавающей запятой.

Пример вызова функции:

```
new Float:a = Float:float(5); // Из целого числа  
new Float:b = Float:float("3.14"); // Из строки  
new Float:c = Float:float(fixed(2)); // Из числа с фиксированной запятой
```

strfloat

Преобразование строки в число с плавающей запятой.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

Float:strfloat(const string[])

Параметры:

string[]	Строка для преобразования (должна содержать числовое значение).
-----------------	---

Возвращаемое значение:

Число с плавающей запятой, соответствующее переданной строке.

Пример вызова функции:

```
new Float:val1 = Float:strfloat("3.1415"); // возвращает 3.1415  
new Float:val2 = Float:strfloat("-2.718"); // возвращает -2.718  
new Float:val3 = Float:strfloat("invalid"); // возвращает 0.0
```

floatmul

Умножение чисел с плавающей запятой.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

Float:floatmul(Float:oper1, Float:oper2)

Параметры:

oper1	Первый множитель (число с плавающей запятой).
oper2	Второй множитель (число с плавающей запятой).

Возвращаемое значение:

Результат умножения двух чисел с плавающей запятой.

Пример вызова функции:

```
new Float:a = Float:floatmul(3.0, 2.5); // возвращает 7.5  
new Float:b = Float:floatmul(-1.5, 4.0); // возвращает -6.0
```

floatdiv

Деление числа с плавающей запятой.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

Float:floatdiv(Float:dividend, Float:divisor)

Параметры:

dividend	Делимое (число с плавающей запятой).
divisor	Делитель (число с плавающей запятой). Не должен быть равен 0.0.

Возвращаемое значение:

Результат деления числа с плавающей запятой. В случае деления на ноль возвращает специальное значение INFINITY (типа Float).

Пример вызова функции:

```
new Float:a = Float:floatdiv(10.0, 2.0); // возвращает 5.0  
new Float:b = Float:floatdiv(1.0, 3.0); // возвращает 0.333...  
new Float:c = Float:floatdiv(5.0, 0.0); // возвращает INFINITY
```

floatadd

Сложение чисел с плавающей запятой.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

Float:floatadd(Float:oper1, Float:oper2)

Параметры:

oper1	Первое слагаемое (число с плавающей запятой).
oper2	Второе слагаемое (число с плавающей запятой).

Возвращаемое значение:

Результат сложения чисел с плавающей запятой.

Пример вызова функции:

```
new Float:result = Float:floatadd(2.5, 3.1); // возвращает 5.6  
new Float:sum = Float:floatadd(-1.2, 0.5); // возвращает -0.7
```

floatsub

Вычитание чисел с плавающей запятой.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

Float:floatsub(Float:oper1, Float:oper2)

Параметры:

oper1	Уменьшаемые (число с плавающей запятой).
oper2	Вычитаемое (число с плавающей запятой).

Возвращаемое значение:

Результат вычитания.

Пример вызова функции:

```
new Float:result1 = Float:floatsub(5.0, 2.5); // возвращает 2.5  
new Float:result2 = Float:floatsub(1.0, 1.1); // возвращает -0.1  
new Float:result3 = Float:floatsub(INFINITY, INFINITY); // возвращает NaN
```

floatfract

Извлечение дробной части числа с плавающей запятой.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

Float:floatfract(Float:value)

Параметры:

value	Число с плавающей запятой для извлечения дробной части.
-------	---

Возвращаемое значение:

Дробная часть числа в диапазоне [0.0, 1.0) или (-1.0, 0.0] для отрицательных чисел.

Пример вызова функции:

```
new Float:f1 = Float:floatfract(3.1415); // возвращает 0.1415  
new Float:f2 = Float:floatfract(-2.718); // возвращает -0.718  
new Float:f3 = Float:floatfract(5.0); // возвращает 0.0
```

floatround

Округление числа с плавающей запятой согласно указанному методу.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

```
floatround(Float:value, floatround_method:method=floatround_round)
```

Параметры:

value	Число с плавающей запятой для округления.
method	Метод округления (по умолчанию floatround_round — математическое округление).

Возвращаемое значение:

Округленное значение с плавающей запятой.

Пример вызова функции:

```
new Float:a = floatround(3.3); // 3.0 (по умолчанию математическое округление)
```

floatcmp

Сравнение двух чисел с плавающей запятой.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

floatcmp(Float:oper1, Float:oper2)

Параметры:

oper1	Первое число с плавающей запятой для сравнения.
oper2	Второе число с плавающей запятой для сравнения.

Возвращаемое значение:

- -1, если **oper1** < **oper2**.
- 0, если **oper1** = **oper2**.
- 1, если **oper1** > **oper2**.

Пример вызова функции:

```
new result1 = floatcmp(3.14, 3.14); // 0 (равны)  
new result2 = floatcmp(1.0, 1.0001); // -1  
new result3 = floatcmp(5.0, 4.9); // 1
```

floatsqrt

Вычисление квадратного корня из числа с плавающей запятой.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

Float:floatsqrt(Float:value)

Параметры:

value	Число с плавающей запятой для извлечения корня. Не должно быть отрицательным.
--------------	---

Возвращаемое значение:

Квадратный корень числа с плавающей запятой.

Пример вызова функции:

```
new Float:root1 = Float:floatsqrt(25.0); // 5.0  
new Float:root2 = Float:floatsqrt(2.0); // ~1.4142
```

floatpower

Возведение числа с плавающей запятой в указанную степень.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

Float:floatpower(Float:value, Float:exponent)

Параметры:

value	Основание степени (число с плавающей запятой).
exponent	Показатель степени (число с плавающей запятой).

Возвращаемое значение:

Результат возведения в степень с плавающей запятой.

Пример вызова функции:

```
new Float:res1 = Float:floatpower(2.0, 3.0); // 8.0  
new Float:res2 = Float:floatpower(4.0, 0.5); // 2.0 (квадратный корень)  
new Float:res3 = Float:floatpower(-1.0, 2.0); // 1.0
```

floatlog

Вычисление логарифма числа с плавающей запятой по указанному основанию.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

Float:floatlog(Float:value, Float:base=10.0)

Параметры:

value	Число с плавающей запятой для логарифмирования, больше 0.
base	Основание логарифма (число с плавающей запятой, больше 0 и не равно 1, по умолчанию 10.0).

Возвращаемое значение:

Значение логарифма с плавающей запятой.

Пример вызова функции:

```
new Float:log1 = Float:floatlog(100.0); // 2.0 (по основанию 10)  
new Float:log2 = Float:floatlog(8.0, 2.0); // 3.0 (логарифм по основанию 2)  
new Float:log3 = Float:floatlog(2.71828); // ~1.0 (натуральный логарифм)
```

floatsin

Вычисление синуса угла для чисел с плавающей запятой.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

Float:floatsin(Float:value, anglemode:mode=radian)

Параметры:

value	Угол для вычисления синуса (число с плавающей запятой).
mode	Единицы измерения угла (по умолчанию radian — радианы (0–2π)).

Возвращаемое значение:

Значение синуса с плавающей запятой в диапазоне [–1.0, 1.0].

Пример вызова функции:

```
new Float:sin1 = Float:floatsin(1.5708); // ~1.0 (sin(π/2) радиан)
```

floatcos

Вычисление косинуса угла для чисел с плавающей запятой.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

Float:floatcos(Float:value, anglemode:mode=radian)

Параметры:

value	Угол для вычисления косинуса (число с плавающей запятой).
mode	Единицы измерения угла (по умолчанию radian — радианы (0–2π)).

Возвращаемое значение:

Значение косинуса с плавающей запятой в диапазоне [–1.0, 1.0].

Пример вызова функции:

```
new Float:cos1 = Float:floatcos(0.0); // 1.0 (cos(0) радиан)
```

floattan

Вычисление тангенса угла для чисел с плавающей запятой.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

Float:floattan(Float:value, anglemode:mode=radian)

Параметры:

value	Угол для вычисления тангенса (число с плавающей запятой).
mode	Единицы измерения угла (по умолчанию radian — радианы (0–2π)).

Возвращаемое значение:

Значение тангенса с плавающей запятой.

Пример вызова функции:

```
new Float:tan1 = Float:floattan(0.7854); // ~1.0 (tan(π/4) радиан)
```

floatabs

Получение абсолютного значения числа с плавающей запятой.

Для использования функции подключите файл:

```
#include <float.inc>
```

Формат функции:

Float:floatabs(Float:value)

Параметры:

value	Число с плавающей запятой для вычисления абсолютного значения.
-------	--

Возвращаемое значение:

Абсолютное значение (модуль) числа с плавающей запятой.

Пример вызова функции:

```
new Float:abs1 = Float:floatabs(3.14); // 3.14  
new Float:abs2 = Float:floatabs(-2.71); // 2.71  
new Float:abs3 = Float:floatabs(0.0); // 0.0
```

Стандартные функции Pawk для строк

Список функций	Описание
<u>strlen</u>	Получение длины строки.
<u>strpack</u>	Копирование и упаковка строки-источника в строку-назначение.
<u>strunpack</u>	Распаковка упакованной строки в обычный строковый формат.
<u>strcopy</u>	Копирование содержимого одной строки в другую с контролем переполнения.
<u>strcat</u>	Объединение двух строк.
<u>strmid</u>	Копирование части содержимого одной строки в другую.
<u>strins</u>	Вставка подстроки в указанную позицию строки.
<u>strdel</u>	Удаление части строки между указанными позициями.
<u>strcmp</u>	Сравнение двух строк.
<u>strfind</u>	Поиск первого вхождения подстроки в строке.
<u>strval</u>	Преобразование строкового представления числа в целочисленное значение.
<u>valstr</u>	Преобразование целочисленного значения в строковое представление.
<u>ispacked</u>	Проверка вида строки (упакована или не упакована).
<u>strformat</u>	Форматирование строки согласно заданному шаблону.
<u>uudecode</u>	Декодирование строки из формата UUencode в бинарные данные.
<u>uuencode</u>	Кодирование бинарных данных в формат UUencode.
<u>memcpy</u>	Копирование данных из одного массива в другой.

strlen

Получение длины строки.

Для использования функции подключите файл:

```
#include <string.h>
```

Формат функции:

```
strlen(const string[])
```

Параметры:

string[]	Строка, длину которой требуется узнать.
-----------------	---

Возвращаемое значение:

Длина строки (целое число).

Примечание. Данную функцию не рекомендуется использовать на пустой строке: если строка не пустая, функция обойдет все символы в строке, чтобы найти символ конца строки «\0» и на основе его позиции узнать длину. Чем длиннее строка, тем больше времени будет затрачено на обход всех символов.

Пример вызова функции:

```
new len = strlen("Пример строки");  
// Вывод: "Длина строки: 13 символов"  
printf("Длина строки: %d символов", len);
```

strpack

Копирование и упаковка строки-источника в строку-назначение.

Для использования функции подключите файл:

```
#include <string.inc>
```

Формат функции:

```
strpack(dest[], const source[], maxlength=sizeof dest)
```

Параметры:

dest[]	Массив для записи упакованной строки.
source[]	Строка, которую требуется упаковать. Если строка уже упакована, то функция скопирует ее.
maxlength	Максимальная длина упакованной строки (по умолчанию размер массива dest[]). Необязательный параметр.

Возвращаемое значение:

Длина упакованной строки.

Пример вызова функции:

```
// Базовое использование  
new packed[32];  
strpack(packed, "Hello World");
```

```
// С указанием максимальной длины  
new small[10];  
strpack(small, "Very long string", sizeof small);
```

strunpack

Распаковка упакованной строки в обычный строковый формат.

Для использования функции подключите файл:

```
#include <string.inc>
```

Формат функции:

```
strunpack(dest[], const source[], maxlength=sizeof dest)
```

Параметры:

dest[]	Массив для записи распакованной строки.
source[]	Строка, которую требуется распаковать. Если строка уже распакована, то функция копирует ее.
maxlength	Максимальная длина распакованной строки (по умолчанию размер массива dest[]). Необязательный параметр.

Возвращаемое значение:

Длина распакованной строки.

Пример вызова функции:

```
// Распаковка упакованной строки
new packed[32], unpacked[32];
strpack(packed, "Compressed text");
strunpack(unpacked, packed);

// С ограничением длины вывода
new short[10];
strunpack(short, packed, sizeof short);
```

strcpy

Копирование содержимого одной строки в другую с контролем переполнения.

Для использования функции подключите файл:

```
#include <string.inc>
```

Формат функции:

```
strcpy(dest[], const source[], maxlenh=sizeof dest)
```

Параметры:

dest[]	Массив для копирования содержимого строки.
source[]	Строка, содержимое которой требуется скопировать.
maxlength	Максимальное количество копируемых символов (по умолчанию размер массива dest[]). Необязательный параметр.

Возвращаемое значение:

Количество скопированных символов (целое число).

Пример вызова функции:

```
// Базовое использование  
new buffer[32];  
strcpy(buffer, "Hello World");
```

```
// Копирование с ограничением длины  
new small[10];  
strcpy(small, "Very long string", sizeof small);
```

strcat

Объединение двух строк.

Для использования функции подключите файл:

```
#include <string.inc>
```

Формат функции:

```
strcat(dest[], const source[], maxlenh=sizeof dest)
```

Параметры:

dest[]	Строка, в конец которой требуется добавить содержимое строки source[] .
source[]	Строка, содержимое которой требуется добавить к строке dest[] .
maxlenh	Максимальное количество копируемых символов (по умолчанию размер массива dest[]). Необязательный параметр.

Возвращаемое значение:

Количество символов, добавленных в строку **dest[]** (целое число).

Примечание. Строка **dest[]** после объединения сохраняет исходный упакованный/неупакованный вид. Например, если перед вызовом функции строка была упакована, то она и после объединения останется упакованной.

Пример вызова функции:

```
// Базовое объединение строк
new text[32] = "Hello";
strcat(text, " World");
printf("%s", text); // "Hello World"
```

```
// Безопасное объединение с контролем длины
new buffer[10] = "Test";
strcat(buffer, "12345", sizeof buffer);
printf("%s", buffer); // "Test1234" (обрезано до размера буфера)
```

strmid

Копирование части содержимого одной строки в другую.

Для использования функции подключите файл:

```
#include <string.inc>
```

Формат функции:

strmid(dest[], const source[], start=0, end=cellmax, maxlength=sizeof dest)

Параметры:

dest[]	Строка, в которую требуется скопировать содержимое строки source[] .
source[]	Строка, часть содержимого которой требуется скопировать в строку dest[] .
start	Позиция начала копирования (нумерация начинается с 0).
end	Позиция завершения копирования. Копируются символы из диапазона от start до (end – 1).
maxlength	Максимальное количество копируемых символов (по умолчанию размер массива dest[]). Необязательный параметр.

Возвращаемое значение:

Количество скопированных символов (целое число).

Примечание. Вид строки **dest[]** (упакована или не упакована) зависит от вида строки **source[]**.

Пример вызова функции:

```
new result[32];
strmid(result, "Hello World", 6, 11);
printf("%s", result); // "World"
```

strins

Вставка подстроки в указанную позицию строки.

Для использования функции подключите файл:

```
#include <string.inc>
```

Формат функции:

bool: strins(string[], const substr[], index, maxlenh=sizeof string)

Параметры:

string[]	Строка, в которую требуется вставить подстроку substr[] .
substr[]	Подстрока для вставки в строку string[] .
index	Позиция начала вставки (0 — начало строки).
maxlength	Максимальный размер массива string[] . Необязательный параметр.

Возвращаемое значение:

true — вставка выполнена успешно, false — вставка невозможна.

Примечание. Строка **string[]** сохраняет исходный упакованный/неупакованный вид после вставки подстроки. Например, если перед вызовом функции строка была упакована, то и после вставки она останется упакованной.

Пример вызова функции:

```
// Вставка в середину строки
new text[32] = "Hello World";
bool: strins(text, "Beautiful ", 6);
printf("%s", text); // "Hello Beautiful World"
```

```
// Вставка с проверкой границ
new buffer[20] = "OpenAI";
bool: strins(buffer, "ChatGPT ", 0, sizeof buffer);
printf("%s", buffer); // "ChatGPT OpenAI" (если хватает места)
```

```
// Вставка в конец строки (аналогично strcat)
new str[32] = "Start";
bool: strins(str, "End", strlen(str));
printf("%s", str); // "StartEnd"
```

strdel

Удаление части строки между указанными позициями.

Для использования функции подключите файл:

```
#include <string.inc>
```

Формат функции:

bool: strdel(string[], start, end)

Параметры:

string[]	Строка, из которой требуется удалить символы.
start	Позиция начала удаления (0 — начало строки).
end	Позиция завершения удаления. Удаляются символы из диапазона от start до (end - 1).

Возвращаемое значение:

true — удаление выполнено успешно, false — удаление невозможно.

Пример вызова функции:

```
// Удаление части строки  
new text[32] = "Hello Beautiful World";  
bool: strdel(text, 6, 15);  
printf("%s", text); // "Hello World"
```

```
// Удаление с начала строки  
new str[32] = "PrefixData";  
bool: strdel(str, 0, 6);  
printf("%s", str); // "Data"
```

```
// Удаление до конца строки  
new buffer[32] = "MainTextExtra";  
bool: strdel(buffer, 8, strlen(buffer));  
printf("%s", buffer); // "MainText"
```

strcmp

Сравнение двух строк.

Для использования функции подключите файл:

```
#include <string.h>
```

Формат функции:

```
strcmp(const string1[], const string2[], bool:ignorecase=false, length=cellmax)
```

Параметры:

string1[]	Первая строка для сравнения.
string2[]	Вторая строка для сравнения.
ignorecase	Флаг регистронезависимого сравнения (по умолчанию false — учитывать регистр). Необязательный параметр.
length	Максимальное количество символов для сравнения. Необязательный параметр.

Возвращаемое значение:

- 0, если строки равны.
- 1, если на *i*-й позиции символы различаются и **string1[i] > string2[i]**.
- -1, если на *i*-й позиции символы различаются и **string1[i] < string2[i]**.

Пример вызова функции:

```
result = strcmp("1234", "1234");  
printf("result: %d", result); // "result: 0"  
  
result = strcmp("1234", "5678");  
printf("result: %d", result); // "result: -1" ('1' < '5')  
  
result = strcmp("efgh", "abcd");  
printf("result: %d", result); // "result: 1" ('e' > 'a')
```

strfind

Поиск первого вхождения подстроки в строке.

Для использования функции подключите файл:

```
#include <string.inc>
```

Формат функции:

```
strfind(const string[], const sub[], bool:ignorecase=false, index=0)
```

Параметры:

string[]	Строка, в которой требуется найти подстроку sub[] .
sub[]	Подстрока, которую требуется найти в строке string[] .
ignorecase	Флаг регистронезависимого поиска (по умолчанию false — учитывать регистр). Необязательный параметр.
index	Позиция начала поиска (по умолчанию 0). Необязательный параметр.

Возвращаемое значение:

Позиция первого вхождения подстроки **sub[]** в строку **string[]** (целое число). И -1, если подстрока не найдена.

Пример вызова функции:

```
// Простой поиск
new pos = strfind("Hello World", "World");
// pos == 6

// Регистронезависимый поиск
pos = strfind("Sample Text", "text", true);
// pos == 7

// Поиск с указанием начальной позиции
pos = strfind("ababab", "ab", false, 2);
// pos == 2

// Проверка наличия подстроки
if (strfind("main string", "sub") != -1) {
    print("Подстрока найдена");
}
```

strval

Преобразование строкового представления числа в целочисленное значение.

Для использования функции подключите файл:

```
#include <string.inc>
```

Формат функции:

```
strval(const string[], index=0)
```

Параметры:

string[]	Строка, которую требуется преобразовать в число.
index	Позиция в строке, с которой требуется начать разбор (по умолчанию 0). Необязательный параметр.

Возвращаемое значение:

Целочисленное значение. Если преобразование не выполнено, то возвращает 0.

Пример вызова функции:

```
new value = strval("42", 0);  
// value == 42
```

valstr

Преобразование целочисленного значения в строковое представление.

Для использования функции подключите файл:

```
#include <string.inc>
```

Формат функции:

valstr(dest[], value, bool:pack=true)

Параметры:

dest[]	Массив для записи строкового представления числа.
value	Целочисленное значение для преобразования.
pack	Флаг упаковки строки (по умолчанию true — сохранить как упакованную строку). Необязательный параметр.

Возвращаемое значение:

Количество записанных символов (целое число).

Пример вызова функции:

```
new str[10];  
valstr(str, 42);  
printf("%s", str); // "42"
```

ispacked

Проверка вида строки (упакована или не упакована).

Для использования функции подключите файл:

```
#include <string.inc>
```

Формат функции:

bool: ispacked(const string[])

Параметры:

string[]	Строка, которую требуется проверить.
-----------------	--------------------------------------

Возвращаемое значение:

true — строка упакована, false — строка не упакована.

Пример вызова функции:

```
new normalStr[] = "Hello";  
if (!ispacked(normalStr)) {  
    print("Строка не упакована");  
}
```

strformat

Форматирование строки согласно заданному шаблону.

Для использования функции подключите файл:

```
#include <string.inc>
```

Формат функции:

```
strformat(dest[], size=sizeof dest, bool:pack=true, const format[], {Fixed,Float,_}:...)
```

Параметры:

dest[]	Массив для записи отформатированной строки.
size	Максимальная длина отформатированной строки (по умолчанию размер массива dest[]). Необязательный параметр.
pack	Флаг упаковки строки (по умолчанию true — сохранить как упакованную строку). Необязательный параметр.
format[]	Строка формата (шаблон).
...	Переменные аргументы для подстановки в шаблон.

Возвращаемое значение:

Количество записанных символов (целое число).

Пример вызова функции:

```
// Базовое форматирование
new buffer[128];
strformat(buffer, sizeof(buffer), true, "Player: %s, Score: %d", "John", 100);

// Форматирование чисел
new float:value = 3.14159;
strformat(buffer, sizeof(buffer), false, "Value: %.2f", value);

// Создание упакованной строки
new packedStr[64 char];
strformat(packedStr, sizeof(packedStr), true, "ID:%04d", 42);
```

uudecode

Декодирование строки из формата UUencode в бинарные данные.

Для использования функции подключите файл:

```
#include <string.inc>
```

Формат функции:

```
uudecode(dest[], const source[], maxlength=sizeof dest)
```

Параметры:

dest[]	Массив для записи декодированных данных.
source[]	Строка в формате UUencode, которую требуется декодировать.
maxlength	Максимальный размер массива dest[] . Необязательный параметр.

Возвращаемое значение:

Количество декодированных байтов (целое число).

Пример вызова функции:

```
new data[128];
new decoded = uudecode(data, "begin 644 file.txt\nMOF%R96T@<FEP\n`nend");
if (decoded > 0) {
    printf("Декодировано %d байт", decoded);
}
```

uuencode

Кодирование бинарных данных в формат UUencode.

Для использования функции подключите файл:

```
#include <string.inc>
```

Формат функции:

uuencode(dest[], const source[], numbytes, maxlength=sizeof dest)

Параметры:

dest[]	Массив для записи закодированной строки.
source[]	Бинарные данные для кодирования.
numbytes	Количество байтов для кодирования.
maxlength	Максимальный размер массива dest[] . Необязательный параметр.

Возвращаемое значение:

Количество записанных символов (целое число).

Примечание. В конце строки с закодированными данными добавляется символ «\n».

Пример вызова функции:

```
// Кодирование бинарных данных
new data[10] = {0x54, 0x45, 0x53, 0x54, 0x00};
new encoded[128];
new result = uuencode(encoded, data, 5, sizeof(encoded));
if (result > 0) {
    printf("Закодированная строка: %s", encoded);
}
```

memcpy

Копирование данных из одного массива в другой.

Для использования функции подключите файл:

```
#include <string.h>
```

Формат функции:

`memcpy(dest[], const source[], index=0, numbytes, maxlength=sizeof dest)`

Параметры:

dest[]	Массив, в который требуется скопировать данные.
source[]	Массив, из которого требуется скопировать данные.
index	Номер байта в массиве source[] , с которого требуется начать копирование (по умолчанию 0). Необязательный параметр.
numbytes	Количество байтов для кодирования.
maxlength	Максимальный размер массива dest[] . Необязательный параметр.

Возвращаемое значение:

1 — копирование выполнено, 0 — копирование невозможно (указаны неправильные параметры).

Пример вызова функции:

```
// Копирование всего массива
new src[10] = {1, 2, 3, 4, 5};
new dest[10];
memcpy(dest, src, 0, sizeof(src));

// Частичное копирование с указанием размера буфера
new buffer[5];
memcpy(buffer, src, 2, 3 * 4, sizeof(buffer)); // Копирует 3 элемента (по 4 байта) начиная с индекса 2
```

Стандартные функции Rawp для времени и таймера

Список функций	Описание
<u>gettime</u>	Получение текущего времени с разбивкой на часы, минуты, секунды.
<u>settime</u>	Установка системного времени.
<u>getdate</u>	Получение текущей даты.
<u>setdate</u>	Установка даты.
<u>settimestamp</u>	Установка системного времени по Unix-временной метке.
<u>cvttimestamp</u>	Конвертация Unix-временной метки в компоненты даты и времени.
<u>settimer</u>	Создание и запуск таймера с заданным интервалом.
<u>tickcount</u>	Получение количества миллисекунд, прошедших с момента запуска системы.
<u>delay</u>	Приостановка выполнения текущего потока на указанное время.

gettime

Получение текущего времени с разбивкой на часы, минуты, секунды.

Для использования функции подключите файл:

```
#include <time.inc>
```

Формат функции:

```
gettime(&hour=0, &minute=0, &second=0)
```

Параметры:

&hour	Ссылка на переменную для получения часа (0–23). Необязательный параметр.
&minute	Ссылка на переменную для получения минут (0–59). Необязательный параметр.
&second	Ссылка на переменную для получения секунд (0–59). Необязательный параметр.

Возвращаемое значение:

Текущее Unix-время (`time_t`) в секундах. Часовая, минутная и секундная составляющие возвращаются по отдельности через параметры **hour**, **minute** и **second** соответственно.

Примечание. Возвращаемое функцией значение (Unix-время), а также значения, возвращаемые через параметры **hour**, **minute** и **second**, обозначают время в часовом поясе UTC+0 (GMT+0).

Пример вызова функции:

```
new h, m, s;  
new currentTime = gettime(h, m, s);  
printf("Время: %02d:%02d:%02d (%d mc)", h, m, s, currentTime);
```

settime

Установка системного времени.

Для использования функции подключите файл:

```
#include <time.inc>
```

Формат функции:

```
settime(hour=cellmin, minute=cellmin, second=cellmin)
```

Параметры:

hour	Часовая составляющая для установки (0–23). Необязательный параметр, при пропуске текущий час не изменяется.
minute	Минутная составляющая для установки (0–59). Необязательный параметр, при пропуске текущая минута не изменяется.
second	Секундная составляющая для установки (0–59). Необязательный параметр, при пропуске текущая секунда не изменяется.

Возвращаемое значение:

0.

Пример вызова функции:

```
// Установка полного времени  
settime(14, 30, 0); // Устанавливает 14:30:00
```

```
// Установка только часов  
settime(12); // Устанавливает 12:XX:XX (минуты и секунды не изменяются)
```

getdate

Получение текущей даты.

Для использования функции подключите файл:

```
#include <time.inc>
```

Формат функции:

```
getdate(&year=0, &month=0, &day=0)
```

Параметры:

&year	Ссылка на переменную для получения года (4 цифры). Необязательный параметр.
&month	Ссылка на переменную для получения месяца (1–12). Необязательный параметр.
&day	Ссылка на переменную для получения дня месяца (1–31). Необязательный параметр.

Возвращаемое значение:

Номер дня с начала года (отсчет начинается с 1, т. е. 1 января считается 1-м днем). Год, месяц и день возвращаются по отдельности через параметры **year**, **month** и **day** соответственно.

Примечание. Все возвращаемые функцией значения (как напрямую, так и через параметры **year**, **month** и **day**) основываются на часовом поясе UTC+0 (GMT+0).

Пример вызова функции:

```
// Получение полной даты
new y, m, d;
new ms = getdate(y, m, d);
printf("Дата: %04d-%02d-%02d, время с полуночи: %d мс", y, m, d, ms);

// Получение только года
new year;
getdate(year);
printf("Текущий год: %d", year);
```

setdate

Установка даты.

Для использования функции подключите файл:

```
#include <time.inc>
```

Формат функции:

```
setdate(year=cellmin, month=cellmin, day=cellmin)
```

Параметры:

year	Год для установки (4 цифры). Необязательный параметр, при пропуске текущий год не изменяется.
month	Месяц для установки (1–12). Необязательный параметр, при пропуске текущий месяц не изменяется.
day	День для установки (1–31). Необязательный параметр, при пропуске текущий день не изменяется.

Возвращаемое значение:

0.

Пример вызова функции:

```
// Установка полной даты  
setdate(2023, 12, 31); // Устанавливает 31 декабря 2023 года
```

```
// Установка только года  
setdate(2024); // Устанавливает 2024 год, месяц и день не изменяются
```

settimestamp

Установка системного времени по Unix-временной метке.

Для использования функции подключите файл:

```
#include <time.inc>
```

Формат функции:

```
settimestamp(seconds1970)
```

Параметры:

seconds1970	Количество секунд, прошедших с 1 января 1970 года.
--------------------	--

Возвращаемое значение:

Нет.

Пример вызова функции:

```
// Установка конкретной даты и времени (1 января 2023 00:00:00 UTC)  
settimestamp(1672531200);
```

cvttimestamp

Конвертация Unix-временной метки в компоненты даты и времени.

Для использования функции подключите файл:

```
#include <time.inc>
```

Формат функции:

`cvttimestamp(seconds1970, &year=0, &month=0, &day=0, &hour=0, &minute=0, &second=0)`

Параметры:

seconds1970	Количество секунд, прошедших с 1 января 1970 года.
&year	Ссылка на переменную для получения года (4 цифры). Необязательный параметр.
&month	Ссылка на переменную для получения месяца (1–12). Необязательный параметр.
&day	Ссылка на переменную для получения дня месяца (1–31). Необязательный параметр.
&hour	Ссылка на переменную для получения часа (0–23). Необязательный параметр.
&minute	Ссылка на переменную для получения минут (0–59). Необязательный параметр.
&second	Ссылка на переменную для получения секунд (0–59). Необязательный параметр.

Возвращаемое значение:

Нет.

Пример вызова функции:

```
// Полное преобразование метки времени
new y, m, d, h, mn, s;
cvttimestamp(1672531200, y, m, d, h, mn, s); // 1 января 2023 00:00:00
printf("Дата: %04d-%02d-%02d %02d:%02d:%02d", y, m, d, h, mn, s);

// Получение только даты
new year, month, day;
cvttimestamp(1672531200, year, month, day);
printf("Дата: %04d-%02d-%02d", year, month, day);

// Получение только времени
new hour, minute;
cvttimestamp(1672531200, .hour = hour, .minute = minute);
printf("Время: %02d:%02d", hour, minute);
```

settimer

Создание и запуск таймера с заданным интервалом.

Для использования функции подключите файл:

```
#include <time.inc>
```

Формат функции:

```
settimer(milliseconds, bool: singleshot=false)
```

Параметры:

milliseconds	Интервал срабатывания таймера в миллисекундах (1–2147483647).
singleshot	Режим работы таймера (по умолчанию false — периодический).

Возвращаемое значение:

ID созданного таймера.

Пример вызова функции:

```
// Создание периодического таймера (срабатывает каждую секунду)  
new Timer:periodic_timer = settimer(1000);
```

tickcount

Получение количества миллисекунд, прошедших с момента запуска системы.

Для использования функции подключите файл:

```
#include <time.inc>
```

Формат функции:

```
tickcount(&granularity=0)
```

Параметры:

&granularity	Ссылка на переменную для получения точности таймера, в миллисекундах. Необязательный параметр.
-------------------------	---

Возвращаемое значение:

Количество миллисекунд с момента запуска системы.

Пример вызова функции:

```
printf("С момента запуска системы прошло %d мс", tickcount());
```

delay

Приостановка выполнения текущего потока на указанное время.

Для использования функции подключите файл:

```
#include <time.inc>
```

Формат функции:

delay(milliseconds)

Параметры:

milliseconds	Время задержки в миллисекундах (0–2147483647).
---------------------	--

Возвращаемое значение:

Нет.

Пример вызова функции:

```
// Простая задержка на 1 секунду  
delay(1000);
```

Параметры устройства

Список групп параметров	Описание
DiscrParamId	Дискретные параметры.
GenericParamsId	Произвольные параметры.
LevelId	Уровневые параметры.
LongParamId	Длинные параметры.

DiscrParamId

Дискретные параметры.

Доступ к параметрам:

- сервер/SMS, чтение: команда GDISCRVALUE;
- сервер/SMS, запись: команда SETDISCRVALUE;
- Т.Скрипт, чтение: нативная функция `tkGetDiscrValue`;
- Т.Скрипт, запись: нативная функция `tkSetDiscrValue`.

Адаптивная обработка доступна для всех параметров. Возможность адаптивной записи отмечена для каждого параметра отдельно.

Все параметры могут служить источниками событий.

BRAKE_SWITCH_SPN_597	<p>1 — педаль тормоза.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, ACH, ГЛОНАРУС. • Выражение в ПО: CANBrake, DPFootbrake. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
PARKING_BRAKE_SWITCH_SPN_70	<p>2 — ручник.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, ACH, ГЛОНАРУС. • Выражение в ПО: CANHandbrake, DPHandbrake. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
OIL_PRESSURE_EMERGENCY_LAMP	<p>3 — аварийная лампа давления масла.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, ACH, ГЛОНАРУС. • Выражение в ПО: DPIPressLvOil. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
CHECK_ENGINE_LAMP	<p>4 — аварийная лампа неисправности двигателя.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, ACH, ГЛОНАРУС. • Выражение в ПО: DPICheckEngine. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

ABS_FAULT_LAMP	<p>5 — лампа неисправности ABS.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPIABS. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
BRAKE_FAULT_LAMP	<p>6 — лампа неисправности тормозов.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
ESP_FAULT_LAMP	<p>7 — лампа неисправности ESP.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPIESP. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
CRUISE_CONTROL_SPN_595	<p>8 — круиз-контроль.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANCruise, DPCruiseControl. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
CLUTCH_SWITCH_SPN_598	<p>9 — педаль сцепления.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANCoupling, DPclutch. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
ACCEL_PEDAL_LOW_IDLE_SW_SPN_558	<p>10 — режим холостого хода.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANIdlingMode. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

ACCEL_PEDAL_KICKDOWN_SW_SPN_559	<p>11 — режим kickdown.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANKickDownMode. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
TORQUE_MODE_SPN_899	<p>12 — режим крутящего момента двигателя.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANTorqueMode. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
PTO_STATE_SPN_976	<p>13 — режим РТО.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANPTOState. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
CRUISE_CONTROL_STATE_SPN_527	<p>14 — состояние круиз-контроля.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANCruiseState. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
IGNITION	<p>15 — зажигание.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPIgnition. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
IGNITION_KEY_IN_LOCK	<p>16 — ключ в замке зажигания.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPKeyIgnition. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

DYNAMIC_IGNITION_2	<p>17 — динамическое зажигание 2.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
WEBASTO	<p>18 — webasto.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPWebasto. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
ENGINE_RUN	<p>19 — двигатель работает.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPEngine. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
ADD_ENGINE_RUN	<p>20 — дополнительный двигатель работает.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPAddEngine. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
READY_FOR_MOVE	<p>21 — готов к движению.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPReady. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
ENGINE_ON_LNG	<p>22 — двигатель работает на СПГ.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPWorkOnGas. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEFT_FRONT_DOOR_OPEN	<p>23 — левая передняя дверь открыта.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPFrontLeftDoor. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
RIGHT_FRONT_DOOR_OPEN	<p>24 — правая передняя дверь открыта.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPFrontRightDoor. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEFT_REAR_DOOR_OPEN	<p>25 — левая задняя дверь открыта.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRearLeftDoor. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
RIGHT_REAR_DOOR_OPEN	<p>26 — правая задняя дверь открыта.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRearRightDoor. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LUGGAGE_BOOT_DOOR_OPEN	<p>27 — открыт багажник.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPTrunk. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
ENGINE_HOOD_OPEN	<p>28 — открыт капот.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DP Hood. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

CHARGER_WIRE_CONNECTED	<p>29 — провод зарядки подключен.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPChargeWire. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
BATTERY_CHARGE	<p>30 — зарядка аккумуляторной батареи включена.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPBatteryCharge. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AUTO_CLOSED	<p>31 — автомобиль закрыт.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPVehicleClosed. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AUTO_CLOSED_BY_STANDARD_RC	<p>32 — автомобиль закрыт при помощи штатного брелока.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPVehicleClosedRC. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
STANDARD_ALARM_ON	<p>33 — штатная сигнализация поставлена на охрану.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPFactoryAlarm. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
STANDARD_ALARM_EMULATION_ON	<p>34 — эмуляция штатной сигнализации активирована.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPEmulFactoryAlarm. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

STANDARD_RC_CL_SIG_SENT_ONCE	<p>35 — сигнал закрытия с помощью заводского ПДУ был отправлен.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPCloseSignalRC. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
STANDARD_RC_OPENING_SIGNAL_SENT	<p>36 — сигнал открытия с помощью заводского ПДУ был отправлен.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPOpenSignalRC. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
REPEAT_CLOSING_SIGNAL_SENT	<p>37 — сигнал перепостановки был отправлен.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRestoreSignal. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LUGGAGE_BOOT_OPENED_BY_RC	<p>38 — багажник был открыт ПДУ.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPTrunkOpenRC. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
CAN_SLEEP_MODE	<p>39 — CAN-модуль в спящем режиме.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPCANSleep. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
STANDARD_RC_CL_SIG_SNT_3_TIMES	<p>40 — сигнал закрытия с помощью заводского ПДУ был отправлен трехкратно.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPCloseSignalRCx3. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

AGB_PARKING_ON	<p>41 — АКПП в режиме «Парковка».</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPTnsP. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
GB_REVERSE_ON	<p>42 — КПП в режиме «Задний ход».</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPTnsR. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGB_NEUTRAL_ON	<p>43 — АКПП в режиме «Нейтраль».</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPTnsN. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGB_MOVE_ON	<p>44 — АКПП в режиме «Движение».</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPTnsD. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
PARKING_LIGHTS	<p>45 — парковочные огни включены.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPParkLights. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LOW_BEAM_HEADLIGHTS	<p>46 — ближний свет фар включен.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPLowLights. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

HIGH_BEAM_HEADLIGHTS	<p>47 — дальний свет фар включен.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPHighLights. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
REAR_FOG_LIGHTS	<p>48 — задние противотуманные фонари включены.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRearFoglights. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AIR_CONDITIONING	<p>49 — кондиционер включен.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPAirCondition. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AUTO_RETARDER	<p>50 — автоматический ретардер.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPAutoRetarder. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
MANUAL_RETARDER	<p>51 — ручной ретардер.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPHandRetarder. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
DRIVER_SEAT_BELT	<p>52 — ремень водителя пристегнут.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPDriverBelt. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

FRONT_PASSENGER_SEAT_BELT	<p>53 — ремень переднего пассажира пристегнут.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPFrontBelt. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
REAR_LEFT_PASSENGER_SEAT_BELT	<p>54 — ремень заднего левого пассажира пристегнут.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRearLeftBelt. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
REAR_RIGHT_PASSENGER_SEAT_BELT	<p>55 — ремень заднего правого пассажира пристегнут.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRearRightBelt. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
REAR_CENTER_PASSENGER_SEAT_BELT	<p>56 — ремень заднего центрального пассажира пристегнут.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRearCentralBelt. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
FRONT_PASSENGER_SEAT_BELT_PR	<p>57 — передний пассажирский ремень присутствует.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPFrontCentralBelt. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
ESP_OFF	<p>58 — ESP выключена.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPESP. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

STOP_LAMP	<p>59 — лампа STOP.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPISStop. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
COOLANT_EMERGENCY_LAMP	<p>60 — лампа температуры/уровня охлаждающей жидкости.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPPressLvCoolant. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
BATTERY_LAMP	<p>61 — индикатор отсутствия зарядки АКБ.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPIBattery. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
PARKING_BRAKE_LAMP	<p>62 — индикатор системы стояночного тормоза.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPIHandbrake. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AIRBAG_LAMP	<p>63 — индикатор подушки безопасности.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPIAirbag. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
EPS_FAULT_LAMP	<p>64 — индикатор отказа EPS (электроусилитель руля).</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPIEPS. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

WARNING_LAMP	<p>65 — индикатор предупреждения.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPIWarning. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
EXTERNAL_LIGHTING_FAULT_LAMP	<p>66 — индикатор неисправности внешних световых приборов.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPExtLightsMalf. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
TYRES_LOW_PRESSURE_LAMP	<p>67 — индикатор низкого давления в шинах.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPLowPressure. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
BRAKE_PADS_WEAR_LAMP	<p>68 — индикатор износа тормозных колодок.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPBrakePadWear. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LOW_FUEL_LEVEL_LAMP	<p>69 — индикатор низкого уровня топлива.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPLowFuelLevel. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
MAINTENANCE_LAMP	<p>70 — индикатор наступления времени технического обслуживания.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

GLOWPLUG_LAMP	<p>71 — индикатор калильных свечей.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPlGlowPlug. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
DPF_LAMP	<p>72 — лампа DPF (сажевый фильтр, FAP).</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPlFAP. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
EPC_LAMP	<p>73 — индикатор EPC (электронный контроль мощности).</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPlEPC. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
ENGINE_OIL_LOW_PRESSURE_LAMP	<p>74 — индикатор низкого давления масла в двигателе.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPlowOilPressure. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
ENGINE_OIL_HIGH_PRESSURE_LAMP	<p>75 — индикатор высокого давления масла в двигателе.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPlHighOilPressure. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
COOLANT_LOW_LEVEL_LAMP	<p>76 — индикатор низкого уровня охлаждающей жидкости.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPlowCoolantLevel. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

HYDRO_FILTER_LAMP	<p>77 — индикатор засорения фильтра масляной гидросистемы.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPHydrFltrClogged. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
HYDRO_OIL_FILTER_LAMP	<p>78 — индикатор засорения масляного фильтра гидросистемы.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPHydrClogged. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
HYDRO_LOW_PRESSURE_LAMP	<p>79 — индикатор низкого давления в гидросистеме.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPLowHydrPressure. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
HYDRO_LOW_LEVEL_LAMP	<p>80 — индикатор низкого уровня масла в гидросистеме.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPLowHydrLevel. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
HYDRO_HIGH_TEMPERATURE_LAMP	<p>81 — индикатор высокой температуры в гидросистеме.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPHighHydrTemper. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
HYDRO_HIGH_LEVEL_LAMP	<p>82 — индикатор перелива масла в баке в гидросистеме.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPHydrOilOver. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

AIR_FILTER_LAMP	<p>83 — индикатор засорения воздушного фильтра.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPAirFtrClogged. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
FUEL_FILTER_LAMP	<p>84 — индикатор засорения топливного фильтра.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPFuelFtrClogged. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
FUEL_WATER_LAMP	<p>85 — индикатор присутствия воды в топливе.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPWaterInFuel. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
BRAKE_FILTER_LAMP	<p>86 — индикатор засорения фильтра тормозной системы.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPBrakeFtrClogged. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
CATALYST_OVERHEAT_LAMP	<p>87 — индикатор перегрева катализатора.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_RIGHT_JOYSTICK_RIGHT	<p>88 — правый джойстик вправо.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRStickRight. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

AGRO_RIGHT_JOYSTICK_LEFT	<p>89 — правый джойстик влево.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRStickLeft. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_RIGHT_JOYSTICK_PUSH	<p>90 — правый джойстик вперед.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRStickFwd. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_RIGHT_JOYSTICK_PULL	<p>91 — правый джойстик назад.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRStickBack. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_LEFT_JOYSTICK_RIGHT	<p>92 — левый джойстик вправо.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPLStickRight. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_LEFT_JOYSTICK_LEFT	<p>93 — левый джойстик влево.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPLStickLeft. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_LEFT_JOYSTICK_PUSH	<p>94 — левый джойстик вперед.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPLStickFwd. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

AGRO_LEFT_JOYSTICK_PULL	<p>95 — левый джойстик назад.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPLStickBack. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_HYDRO_REAR_1	<p>96 — первый задний гидропривод.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRearHydrDrv1. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_HYDRO_REAR_2	<p>97 — второй задний гидропривод.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRearHydrDrv2. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_HYDRO_REAR_3	<p>98 — третий задний гидропривод.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRearHydrDrv3. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_HYDRO_REAR_4	<p>99 — четвертый задний гидропривод.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRearHydrDrv4. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_HYDRO_FRONT_1	<p>100 — первый передний гидропривод.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPFrontHydrDrv1. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

AGRO_HYDRO_FRONT_2	<p>101 — второй передний гидропривод.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPFrontHydrDrv2. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_HYDRO_FRONT_3	<p>102 — третий передний гидропривод.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPFrontHydrDrv3. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_HYDRO_FRONT_4	<p>103 — четвертый передний гидропривод.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPFrontHydrDrv4. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_THREE_POINT_HITCH_FRONT	<p>104 — передняя трехточечная система навески.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPFrontHitch. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_THREE_POINT_HITCH_REAR	<p>105 — задняя трехточечная система навески.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRearHitch. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_PTO_FRONT_SPN_3452	<p>106 — передний механизм отбора мощности.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPFrontPTO. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

AGRO_PTO_REAR_SPN_3453	<p>107 — задний механизм отбора мощности.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPRearPTO. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_MOWING	<p>108 — покос.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPMowing. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_THRESHING	<p>109 — молотья.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPThreshing. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_GRAIN_HOPPER_UNLOADING	<p>110 — разгрузка зерна из бункера.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPGrainUnloading. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_GRAIN_HOPPER_100_LOAD	<p>111 — зерновой бункер заполнен на 100 %.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPGrainTank100. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_GRAIN_HOPPER_70_LOAD	<p>112 — зерновой бункер заполнен на 70 %.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPGrainTank70. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

AGRO_GRAIN_HOPPER_OPEN	<p>113 — зерновой бункер открыт.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPGrainTankOpen. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_ULD_MECH_ACT_TUBE_AWAY	<p>114 — привод выгрузного механизма при сложенной выгрузной трубе включен.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPUUnloadingDrv. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_CLEANING_FAN_CTRL_DIS	<p>115 — управление вентилятором очистки отключено. 0b01 — отключено.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPCleaningFanCtrl. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_THRESHING_DRUM_CTRL_DIS	<p>116 — управление молотильным барабаном отключено. 0b01 — отключено.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPThreshingDrumCtrl. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_STRAW_WALKER_FAULT	<p>117 — соломотряс забит.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPStawRake. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
AGRO_THRESHING_DRUM_EXC_CLR	<p>118 — избыточный зазор под молотильным барабаном.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPClearanceUnderDrum. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

US_SALT_THROWER	<p>119 — распылитель соли (песка).</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPSaltSprayer. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
US_REAGENTS_POURING	<p>120 — разливка реагентов.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPReagentWatering. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
US_CONVEYOR_BELT	<p>121 — конвейерный ремень.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPConveyorBelt. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
US_SALT_THROWER_WHEEL_DRIVE	<p>122 — привод колеса солеразбрасывателя.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPSprDrvWheel. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
US_BRUSH	<p>123 — щетки.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPBrush. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
US_VACUUM_CLEANER	<p>124 — пылесос.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPVacuum. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

US_WATER_SUPPLY	<p>125 — подача воды.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPWaterSupply. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
US_WASHING_MACHINE	<p>126 — мощный аппарат высокого давления.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPWashingDevice. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
US_WATER_PUMP	<p>127 — насос подачи жидкости.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPLiquidPump. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
US_HOPPER_UNLOADING	<p>128 — выгрузка из бункера.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPBunkerUnloading. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
US_SALT_LOW_LEVEL_LAMP	<p>129 — индикатор низкого уровня соли (песка) в баке.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPLowSaltLevel. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
US_WATER_LOW_LEVEL_LAMP	<p>130 — индикатор низкого уровня воды в баке.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPLowWaterLevel. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

US_REAGENTS_USAGE	<p>131 — использование реагентов.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPReagentUsing. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
US_COMPRESSOR	<p>132 — компрессор.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPCompressor. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
US_WATER_VALVE	<p>133 — водяной клапан.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPWaterValve. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
US_CABIN_MOVED_UP	<p>134 — статус «Кабина перемещена вверх».</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPCabAtTop. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
US_CABIN_MOVED_DOWN	<p>135 — статус «Кабина перемещена вниз».</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DPCabAtBottom. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
EDDP_ACCELERATION	<p>136 — событие качества вождения: резкое ускорение.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.

EDDP_BREAKING	<p>137 — событие качества вождения: резкое торможение.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
EDDP_EXTRBREAKING	<p>138 — событие качества вождения: экстренное торможение.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
EDDP_RIGHTTURN	<p>139 — событие качества вождения: резкий поворот направо.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
EDDP_LEFTTURN	<p>140 — событие качества вождения: резкий поворот налево.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
EDDP_HOLE	<p>141 — событие качества вождения: неровность дороги (яма).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
EDDP_TILT	<p>142 — событие качества вождения: опрокидывание.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
EDDP_OVERTURN	<p>143 — событие качества вождения: переворот.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.

EDDP_ANY_EVENT	<p>145 — качество вождения: любое событие из контроля ускорений.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
EDDP_SPEEDPOROG1	<p>146 — превышен порог скорости 1.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
EDDP_SPEEDPOROG2	<p>147 — превышен порог скорости 2.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
EDDP_SPEEDPOROG3	<p>148 — превышен порог скорости 3.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
MOT_MOTION_FROM_NAV	<p>149 — признак движения по навигационному приемнику.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
MOT_MOTION_FROM_ACCEL	<p>150 — признак движения по акселерометру.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
MOT_FAST_MOTION_FROM_NAV	<p>151 — признак быстрого движения по навигационному приемнику.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.

TNS_CURRENT_STATE1	<p>152 — текущее состояние подключения к серверу 1 (см. ServerConnectionStatus).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
TNS_CURRENT_STATE2	<p>153 — текущее состояние подключения к серверу 2 (см. ServerConnectionStatus).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
TNS_CURRENT_STATE3	<p>154 — текущее состояние подключения к серверу 3 (см. ServerConnectionStatus).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
TNS_MAX_STATE1	<p>155 — максимальное состояние подключения к серверу 1 (см. ServerConnectionStatus).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
TNS_MAX_STATE2	<p>156 — максимальное состояние подключения к серверу 2 (см. ServerConnectionStatus).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
TNS_MAX_STATE3	<p>157 — максимальное состояние подключения к серверу 3 (см. ServerConnectionStatus).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.

<p>TKAM_1_OUT_1</p>	<p>158 — состояние выхода 1 ДУН 1.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(1). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_1_OUT_2</p>	<p>159 — состояние выхода 2 ДУН 1.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(1). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_2_OUT_1</p>	<p>160 — состояние выхода 1 ДУН 2.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(2). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_2_OUT_2</p>	<p>161 — состояние выхода 2 ДУН 2.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(2). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_3_OUT_1</p>	<p>162 — состояние выхода 1 ДУН 3.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(3). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_3_OUT_2</p>	<p>163 — состояние выхода 2 ДУН 3.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(3). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_4_OUT_1</p>	<p>164 — состояние выхода 1 ДУН 4.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(4). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_4_OUT_2</p>	<p>165 — состояние выхода 2 ДУН 4.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(4). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_5_OUT_1</p>	<p>166 — состояние выхода 1 ДУН 5.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(5). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_5_OUT_2</p>	<p>167 — состояние выхода 2 ДУН 5.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(5). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_6_OUT_1</p>	<p>168 — состояние выхода 1 ДУН 6.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(6). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_6_OUT_2</p>	<p>169 — состояние выхода 2 ДУН 6.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(6). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_7_OUT_1</p>	<p>170 — состояние выхода 1 ДУН 7.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(7). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_7_OUT_2</p>	<p>171 — состояние выхода 2 ДУН 7.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(7). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_8_OUT_1</p>	<p>172 — состояние выхода 1 ДУН 8.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(8). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_8_OUT_2</p>	<p>173 — состояние выхода 2 ДУН 8.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(8). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_9_OUT_1</p>	<p>174 — состояние выхода 1 ДУН 9.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(9). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_9_OUT_2</p>	<p>175 — состояние выхода 2 ДУН 9.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(9). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_10_OUT_1</p>	<p>176 — состояние выхода 1 ДУН 10.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(10). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_10_OUT_2</p>	<p>177 — состояние выхода 2 ДУН 10.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(10). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_11_OUT_1</p>	<p>178 — состояние выхода 1 ДУН 11.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(11). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_11_OUT_2</p>	<p>179 — состояние выхода 2 ДУН 11.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(11). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_12_OUT_1</p>	<p>180 — состояние выхода 1 ДУН 12.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(12). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_12_OUT_2</p>	<p>181 — состояние выхода 2 ДУН 12.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(12). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_13_OUT_1</p>	<p>182 — состояние выхода 1 ДУН 13.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(13). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_13_OUT_2</p>	<p>183 — состояние выхода 2 ДУН 13.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(13). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_14_OUT_1</p>	<p>184 — состояние выхода 1 ДУН 14.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(14). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_14_OUT_2</p>	<p>185 — состояние выхода 2 ДУН 14.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(14). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_15_OUT_1</p>	<p>186 — состояние выхода 1 ДУН 15.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(15). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_15_OUT_2</p>	<p>187 — состояние выхода 2 ДУН 15.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(15). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_16_OUT_1</p>	<p>188 — состояние выхода 1 ДУН 16.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut1(16). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_16_OUT_2</p>	<p>189 — состояние выхода 2 ДУН 16.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMOut2(16). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

NAV_COORDS_VALID	<p>190 — прием координат достоверен.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
IBUTTON_ID_READ_FLAG	<p>191 — считан идентификатор по iButton.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Устанавливается в 1 и снова сбрасывается в 0 в течение секунды после считывания.</p>
BLE_ID_READ_FLAG	<p>192 — считан идентификатор по BLE.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E) (S/N с 2904500/2810600), SX, GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Устанавливается в 1 и снова сбрасывается в 0 в течение секунды после считывания.</p>
CAN_ID_READ_FLAG	<p>193 — считан идентификатор по CAN.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Устанавливается в 1 и снова сбрасывается в 0 в течение секунды после считывания.</p>
MODBUS_ID_READ_FLAG	<p>194 — считан идентификатор по MODBUS.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Устанавливается в 1 и снова сбрасывается в 0 в течение секунды после считывания.</p>

POSITION_OF_DOORS_SPN_1821	<p>195 — положение дверей.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
RAMP_POSITION_SPN_1820	<p>196 — рампа/лифт для коляски.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
STATUS_2_OF_DOORS_SPN_3411	<p>197 — статус дверей.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
CURRENT_GEAR_SPN_523	<p>198 — текущая передача.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: VSGear. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
FUEL_TYPE_SPN_5837	<p>199 — используемое топливо.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
OUTPUT_OK_STATE_1	<p>200 — состояние выхода 1.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: Out(1), Out1. • Период записи: PERIODANALOG. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: На АвтоГРАФ-Mobile X значение параметра в запись не сохраняется.</p>

OUTPUT_OK_STATE_2	<p>201 — состояние выхода 2.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, ГЛОНАРУС, Mobile X. • Выражение в ПО: Out(2), Out2. • Период записи: PERIODANALOG. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: На АвтоГРАФ-Mobile X значение параметра в запись не сохраняется.</p>
OUTPUT_OK_STATE_3	<p>202 — состояние выхода 3.</p> <ul style="list-style-type: none"> • Доступен в: GX/GX WIFI, АСН, Mobile X. • Выражение в ПО: Out(3). • Период записи: PERIODANALOG. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: На АвтоГРАФ-Mobile X значение параметра в запись не сохраняется.</p>
AM_1_EVENT_STATE	<p>203 — состояние сработки события датчика Эскорт DU-BLE 1.</p> <ul style="list-style-type: none"> • Доступен в: ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: ТКAMPERIOD, ТКAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>
AM_2_EVENT_STATE	<p>204 — состояние сработки события датчика Эскорт DU-BLE 2.</p> <ul style="list-style-type: none"> • Доступен в: ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: ТКAMPERIOD, ТКAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>
AM_3_EVENT_STATE	<p>205 — состояние сработки события датчика Эскорт DU-BLE 3.</p> <ul style="list-style-type: none"> • Доступен в: ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: ТКAMPERIOD, ТКAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>

AM_4_EVENT_STATE	<p>206 — состояние сработки события датчика Эскорт DU-BLE 4.</p> <ul style="list-style-type: none">• Доступен в: ГЛОНАРУС.• Выражение в ПО: нет.• Период записи: TKAMPERIOD, TKAM.• Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER.• Адаптивная запись: нет.• Доступ через сервер/SMS: чтение.• Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>
AM_5_EVENT_STATE	<p>207 — состояние сработки события датчика Эскорт DU-BLE 5.</p> <ul style="list-style-type: none">• Доступен в: ГЛОНАРУС.• Выражение в ПО: нет.• Период записи: TKAMPERIOD, TKAM.• Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER.• Адаптивная запись: нет.• Доступ через сервер/SMS: чтение.• Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>
AM_6_EVENT_STATE	<p>208 — состояние сработки события датчика Эскорт DU-BLE 6.</p> <ul style="list-style-type: none">• Доступен в: ГЛОНАРУС.• Выражение в ПО: нет.• Период записи: TKAMPERIOD, TKAM.• Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER.• Адаптивная запись: нет.• Доступ через сервер/SMS: чтение.• Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>
AM_7_EVENT_STATE	<p>209 — состояние сработки события датчика Эскорт DU-BLE 7.</p> <ul style="list-style-type: none">• Доступен в: ГЛОНАРУС.• Выражение в ПО: нет.• Период записи: TKAMPERIOD, TKAM.• Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER.• Адаптивная запись: нет.• Доступ через сервер/SMS: чтение.• Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>

<p>AM_8_EVENT_STATE</p>	<p>210 — состояние сработки события датчика Эскорт DU-BLE 8.</p> <ul style="list-style-type: none"> • Доступен в: ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: ТКAMPERIOD, ТКAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>
<p>AM_9_EVENT_STATE</p>	<p>211 — состояние сработки события датчика Эскорт DU-BLE 9.</p> <ul style="list-style-type: none"> • Доступен в: ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: ТКAMPERIOD, ТКAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>
<p>AM_10_EVENT_STATE</p>	<p>212 — состояние сработки события датчика Эскорт DU-BLE 10.</p> <ul style="list-style-type: none"> • Доступен в: ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: ТКAMPERIOD, ТКAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>
<p>AM_11_EVENT_STATE</p>	<p>213 — состояние сработки события датчика Эскорт DU-BLE 11.</p> <ul style="list-style-type: none"> • Доступен в: ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: ТКAMPERIOD, ТКAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>

AM_12_EVENT_STATE	<p>214 — состояние сработки события датчика Эскорт DU-BLE 12.</p> <ul style="list-style-type: none">• Доступен в: ГЛОНАРУС.• Выражение в ПО: нет.• Период записи: ТКAMPERIOD, ТКAM.• Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER.• Адаптивная запись: нет.• Доступ через сервер/SMS: чтение.• Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>
AM_13_EVENT_STATE	<p>215 — состояние сработки события датчика Эскорт DU-BLE 13.</p> <ul style="list-style-type: none">• Доступен в: ГЛОНАРУС.• Выражение в ПО: нет.• Период записи: ТКAMPERIOD, ТКAM.• Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER.• Адаптивная запись: нет.• Доступ через сервер/SMS: чтение.• Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>
AM_14_EVENT_STATE	<p>216 — состояние сработки события датчика Эскорт DU-BLE 14.</p> <ul style="list-style-type: none">• Доступен в: ГЛОНАРУС.• Выражение в ПО: нет.• Период записи: ТКAMPERIOD, ТКAM.• Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER.• Адаптивная запись: нет.• Доступ через сервер/SMS: чтение.• Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>
AM_15_EVENT_STATE	<p>217 — состояние сработки события датчика Эскорт DU-BLE 15.</p> <ul style="list-style-type: none">• Доступен в: ГЛОНАРУС.• Выражение в ПО: нет.• Период записи: ТКAMPERIOD, ТКAM.• Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER.• Адаптивная запись: нет.• Доступ через сервер/SMS: чтение.• Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>

<p>AM_16_EVENT_STATE</p>	<p>218 — состояние сработки события датчика Эскорт DU-BLE 16.</p> <ul style="list-style-type: none"> • Доступен в: ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Внеочередная запись сохраняется при изменении значения данного параметра.</p>
<p>CAMERA_EVENT_STATE</p>	<p>219 — события камеры.</p> <ul style="list-style-type: none"> • Доступен в: GX/GX WIFI, АСН. • Выражение в ПО: числовой параметр с типом 0x000303. • Период записи: RS232CAMERAEVENTPERIOD1. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Флаги кратковременных событий могут устанавливаться и сбрасываться в течение одной секунды. Настоятельно рекомендуется использовать механизм адаптивной записи.</p>
<p>UWB_ANCHOR_DANGER_ZONE</p>	<p>220 — тег находится в опасной зоне (см. DangerZoneType).</p> <ul style="list-style-type: none"> • Доступен в: Mobile X (с модулем UWB). • Выражение в ПО: числовой параметр с типом 0x0000FD. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: нет. <p>ВНИМАНИЕ: Значение параметра сохраняется в бинарный файл только при перемещении в зону с другим уровнем опасности.</p>
<p>MOT_STOP_FROM_NAV</p>	<p>221 — признак остановки по навигационному приемнику.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
<p>INT_BAT_CHARGING</p>	<p>222 — признак зарядки внутренней батареи.</p> <ul style="list-style-type: none"> • Доступен в: Mobile X. • Выражение в ПО: внутреннее событие 145—ChargerStatChange. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: нет. <p>ВНИМАНИЕ: Значение параметра сохраняется в бинарный файл при изменении.</p>

<p>LLS_1_ERRORS</p>	<p>223 — ошибки ДУТ 1.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS1, TLLS(1). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>
<p>LLS_2_ERRORS</p>	<p>224 — ошибки ДУТ 2.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS2, TLLS(2). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>
<p>LLS_3_ERRORS</p>	<p>225 — ошибки ДУТ 3.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS3, TLLS(3). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>
<p>LLS_4_ERRORS</p>	<p>226 — ошибки ДУТ 4.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS4, TLLS(4). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>

<p>LLS_5_ERRORS</p>	<p>227 — ошибки ДУТ 5.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS5, TLLS(5). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>
<p>LLS_6_ERRORS</p>	<p>228 — ошибки ДУТ 6.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS6, TLLS(6). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>
<p>LLS_7_ERRORS</p>	<p>229 — ошибки ДУТ 7.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS7, TLLS(7). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>
<p>LLS_8_ERRORS</p>	<p>230 — ошибки ДУТ 8.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS8, TLLS(8). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>

<p>TKAM_1_ROLL_CNTR</p>	<p>231 — количество оборотов, сделанное датчиком угла наклона TKAM 1.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_2_ROLL_CNTR</p>	<p>232 — количество оборотов, сделанное датчиком угла наклона TKAM 2.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_3_ROLL_CNTR</p>	<p>233 — количество оборотов, сделанное датчиком угла наклона TKAM 3.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_4_ROLL_CNTR</p>	<p>234 — количество оборотов, сделанное датчиком угла наклона TKAM 4.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_5_ROLL_CNTR</p>	<p>235 — количество оборотов, сделанное датчиком угла наклона TKAM 5.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_6_ROLL_CNTR</p>	<p>236 — количество оборотов, сделанное датчиком угла наклона TKAM 6.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_7_ROLL_CNTR</p>	<p>237 — количество оборотов, сделанное датчиком угла наклона TKAM 7.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_8_ROLL_CNTR</p>	<p>238 — количество оборотов, сделанное датчиком угла наклона TKAM 8.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_9_ROLL_CNTR</p>	<p>239 — количество оборотов, сделанное датчиком угла наклона TKAM 9.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_10_ROLL_CNTR</p>	<p>240 — количество оборотов, сделанное датчиком угла наклона TKAM 10.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_11_ROLL_CNTR</p>	<p>241 — количество оборотов, сделанное датчиком угла наклона TKAM 11.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_12_ROLL_CNTR</p>	<p>242 — количество оборотов, сделанное датчиком угла наклона TKAM 12.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_13_ROLL_CNTR</p>	<p>243 — количество оборотов, сделанное датчиком угла наклона TKAM 13.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_14_ROLL_CNTR</p>	<p>244 — количество оборотов, сделанное датчиком угла наклона TKAM 14.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_15_ROLL_CNTR</p>	<p>245 — количество оборотов, сделанное датчиком угла наклона TKAM 15.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_16_ROLL_CNTR</p>	<p>246 — количество оборотов, сделанное датчиком угла наклона TKAM 16.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

DISCR_RAM_PARAM_1	<p>247 — произвольный дискретный параметр RAM 1.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.
DISCR_RAM_PARAM_2	<p>248 — произвольный дискретный параметр RAM 2.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.
DISCR_RAM_PARAM_3	<p>249 — произвольный дискретный параметр RAM 3.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.
DISCR_RAM_PARAM_4	<p>250 — произвольный дискретный параметр RAM 4.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.
DISCR_RAM_PARAM_5	<p>251 — произвольный дискретный параметр RAM 5.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.
DISCR_RAM_PARAM_6	<p>252 — произвольный дискретный параметр RAM 6.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.
DISCR_RAM_PARAM_7	<p>253 — произвольный дискретный параметр RAM 7.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.

DISCR_RAM_PARAM_8	<p>254 — произвольный дискретный параметр RAM 8.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.
DISCR_RAM_PARAM_9	<p>255 — произвольный дискретный параметр RAM 9.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.
DISCR_RAM_PARAM_10	<p>256 — произвольный дискретный параметр RAM 10.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.
ANY_POI_IN	<p>257 — признак нахождения в любой контрольной точке.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.

GenericParamsId

Произвольные параметры.

Доступ к параметрам:

- сервер/SMS, чтение: команда GENERICVALUE;
- сервер/SMS, запись: нет;
- Т.Скрипт, чтение: нативные функции `tkGetGenericFloatValue`, `tkGetGenericIntValue`;
- Т.Скрипт, запись: нативные функции `tkSetGenericFloatValue`, `tkSetGenericIntValue`.

Адаптивная обработка доступна для всех параметров.

Параметры MODBUS:

Общий формат: GENERIC_MODBUS_PARAM_n — произвольный параметр датчика Modbus n.

- Доступен в LX/LX(E), SX, GX/GX WIFI, АСН, ГЛОНАРУС.
- Выражение в ПО: MODBUSFloat(n, order), MODBUSInt(n, order).
- Период записи: MODBUSSENPERIOD.
- Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER.
- Адаптивная запись: да.
- Доступ через сервер/SMS: чтение.
- Доступ через Т.Скрипт: чтение и запись.

Примечание.

- *Запись сохраняется только при наличии новых данных с момента предыдущей записи.*
- *Описанное выше справедливо для параметров размером 2 или 4 байта. Параметры большего размера доступны только на GX/GX WIFI и АСН. Для таких параметров недоступен стандартный механизм адаптивной обработки. При необходимости адаптивной записи следует воспользоваться настройкой MODBUSSENLONGADAPTIVE. Доступ через сервер и SMS на чтение осуществляется при помощи MODBUSSENLONGCONTROL. Доступ к таким параметрам из Т.Скрипт невозможен. Для доступа из ПО необходимо составление пользовательских выражений.*

Параметры CAN:

Общий формат: GENERIC_CAN_PARAM_n — произвольный параметр CAN n.

- Доступен в SX, GX/GX WIFI, АСН, ГЛОНАРУС.
- Выражение в ПО: NDCANIntByType(type), где type — значение, заданное командой CANGENERICTYPEn.
- Период записи: CANPERIOD.
- Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER.
- Адаптивная запись: да.
- Доступ через сервер/SMS: чтение.
- Доступ через Т.Скрипт: чтение и запись.

Примечание. *Запись сохраняется только при наличии новых данных с момента предыдущей записи.*

GENERIC_MODBUS_PARAM_1	1 — произвольный параметр датчика Modbus 1.
GENERIC_MODBUS_PARAM_2	2 — произвольный параметр датчика Modbus 2.
GENERIC_MODBUS_PARAM_3	3 — произвольный параметр датчика Modbus 3.
GENERIC_MODBUS_PARAM_4	4 — произвольный параметр датчика Modbus 4.
GENERIC_MODBUS_PARAM_5	5 — произвольный параметр датчика Modbus 5.
GENERIC_MODBUS_PARAM_6	6 — произвольный параметр датчика Modbus 6.
GENERIC_MODBUS_PARAM_7	7 — произвольный параметр датчика Modbus 7.
GENERIC_MODBUS_PARAM_8	8 — произвольный параметр датчика Modbus 8.
GENERIC_MODBUS_PARAM_9	9 — произвольный параметр датчика Modbus 9.
GENERIC_MODBUS_PARAM_10	10 — произвольный параметр датчика Modbus 10.
GENERIC_MODBUS_PARAM_11	11 — произвольный параметр датчика Modbus 11.
GENERIC_MODBUS_PARAM_12	12 — произвольный параметр датчика Modbus 12.
GENERIC_MODBUS_PARAM_13	13 — произвольный параметр датчика Modbus 13.
GENERIC_MODBUS_PARAM_14	14 — произвольный параметр датчика Modbus 14.
GENERIC_MODBUS_PARAM_15	15 — произвольный параметр датчика Modbus 15.
GENERIC_MODBUS_PARAM_16	16 — произвольный параметр датчика Modbus 16.
GENERIC_MODBUS_PARAM_17	17 — произвольный параметр датчика Modbus 17.
GENERIC_MODBUS_PARAM_18	18 — произвольный параметр датчика Modbus 18.
GENERIC_MODBUS_PARAM_19	19 — произвольный параметр датчика Modbus 19.
GENERIC_MODBUS_PARAM_20	20 — произвольный параметр датчика Modbus 20.
GENERIC_MODBUS_PARAM_21	21 — произвольный параметр датчика Modbus 21.
GENERIC_MODBUS_PARAM_22	22 — произвольный параметр датчика Modbus 22.
GENERIC_MODBUS_PARAM_23	23 — произвольный параметр датчика Modbus 23.
GENERIC_MODBUS_PARAM_24	24 — произвольный параметр датчика Modbus 24.
GENERIC_MODBUS_PARAM_25	25 — произвольный параметр датчика Modbus 25.
GENERIC_MODBUS_PARAM_26	26 — произвольный параметр датчика Modbus 26.
GENERIC_MODBUS_PARAM_27	27 — произвольный параметр датчика Modbus 27.
GENERIC_MODBUS_PARAM_28	28 — произвольный параметр датчика Modbus 28.
GENERIC_MODBUS_PARAM_29	29 — произвольный параметр датчика Modbus 29.
GENERIC_MODBUS_PARAM_30	30 — произвольный параметр датчика Modbus 30.
GENERIC_MODBUS_PARAM_31	31 — произвольный параметр датчика Modbus 31.
GENERIC_MODBUS_PARAM_32	32 — произвольный параметр датчика Modbus 32.
GENERIC_MODBUS_PARAM_33	33 — произвольный параметр датчика Modbus 33.
GENERIC_MODBUS_PARAM_34	34 — произвольный параметр датчика Modbus 34.
GENERIC_MODBUS_PARAM_35	35 — произвольный параметр датчика Modbus 35.
GENERIC_MODBUS_PARAM_36	36 — произвольный параметр датчика Modbus 36.

GENERIC_MODBUS_PARAM_37	37 — произвольный параметр датчика Modbus 37.
GENERIC_MODBUS_PARAM_38	38 — произвольный параметр датчика Modbus 38.
GENERIC_MODBUS_PARAM_39	39 — произвольный параметр датчика Modbus 39.
GENERIC_MODBUS_PARAM_40	40 — произвольный параметр датчика Modbus 40.
GENERIC_MODBUS_PARAM_41	41 — произвольный параметр датчика Modbus 41.
GENERIC_MODBUS_PARAM_42	42 — произвольный параметр датчика Modbus 42.
GENERIC_MODBUS_PARAM_43	43 — произвольный параметр датчика Modbus 43.
GENERIC_MODBUS_PARAM_44	44 — произвольный параметр датчика Modbus 44.
GENERIC_MODBUS_PARAM_45	45 — произвольный параметр датчика Modbus 45.
GENERIC_MODBUS_PARAM_46	46 — произвольный параметр датчика Modbus 46.
GENERIC_MODBUS_PARAM_47	47 — произвольный параметр датчика Modbus 47.
GENERIC_MODBUS_PARAM_48	48 — произвольный параметр датчика Modbus 48.
GENERIC_MODBUS_PARAM_49	49 — произвольный параметр датчика Modbus 49.
GENERIC_MODBUS_PARAM_50	50 — произвольный параметр датчика Modbus 50.
GENERIC_MODBUS_PARAM_51	51 — произвольный параметр датчика Modbus 51.
GENERIC_MODBUS_PARAM_52	52 — произвольный параметр датчика Modbus 52.
GENERIC_MODBUS_PARAM_53	53 — произвольный параметр датчика Modbus 53.
GENERIC_MODBUS_PARAM_54	54 — произвольный параметр датчика Modbus 54.
GENERIC_MODBUS_PARAM_55	55 — произвольный параметр датчика Modbus 55.
GENERIC_MODBUS_PARAM_56	56 — произвольный параметр датчика Modbus 56.
GENERIC_MODBUS_PARAM_57	57 — произвольный параметр датчика Modbus 57.
GENERIC_MODBUS_PARAM_58	58 — произвольный параметр датчика Modbus 58.
GENERIC_MODBUS_PARAM_59	59 — произвольный параметр датчика Modbus 59.
GENERIC_MODBUS_PARAM_60	60 — произвольный параметр датчика Modbus 60.
GENERIC_MODBUS_PARAM_61	61 — произвольный параметр датчика Modbus 61.
GENERIC_MODBUS_PARAM_62	62 — произвольный параметр датчика Modbus 62.
GENERIC_MODBUS_PARAM_63	63 — произвольный параметр датчика Modbus 63.
GENERIC_MODBUS_PARAM_64	64 — произвольный параметр датчика Modbus 64.
GENERIC_MODBUS_PARAM_65	65 — произвольный параметр датчика Modbus 65.
GENERIC_MODBUS_PARAM_66	66 — произвольный параметр датчика Modbus 66.
GENERIC_MODBUS_PARAM_67	67 — произвольный параметр датчика Modbus 67.
GENERIC_MODBUS_PARAM_68	68 — произвольный параметр датчика Modbus 68.
GENERIC_MODBUS_PARAM_69	69 — произвольный параметр датчика Modbus 69.
GENERIC_MODBUS_PARAM_70	70 — произвольный параметр датчика Modbus 70.
GENERIC_MODBUS_PARAM_71	71 — произвольный параметр датчика Modbus 71.
GENERIC_MODBUS_PARAM_72	72 — произвольный параметр датчика Modbus 72.

GENERIC_MODBUS_PARAM_73	73 — произвольный параметр датчика Modbus 73.
GENERIC_MODBUS_PARAM_74	74 — произвольный параметр датчика Modbus 74.
GENERIC_MODBUS_PARAM_75	75 — произвольный параметр датчика Modbus 75.
GENERIC_MODBUS_PARAM_76	76 — произвольный параметр датчика Modbus 76.
GENERIC_MODBUS_PARAM_77	77 — произвольный параметр датчика Modbus 77.
GENERIC_MODBUS_PARAM_78	78 — произвольный параметр датчика Modbus 78.
GENERIC_MODBUS_PARAM_79	79 — произвольный параметр датчика Modbus 79.
GENERIC_MODBUS_PARAM_80	80 — произвольный параметр датчика Modbus 80.
GENERIC_MODBUS_PARAM_81	81 — произвольный параметр датчика Modbus 81.
GENERIC_MODBUS_PARAM_82	82 — произвольный параметр датчика Modbus 82.
GENERIC_MODBUS_PARAM_83	83 — произвольный параметр датчика Modbus 83.
GENERIC_MODBUS_PARAM_84	84 — произвольный параметр датчика Modbus 84.
GENERIC_MODBUS_PARAM_85	85 — произвольный параметр датчика Modbus 85.
GENERIC_MODBUS_PARAM_86	86 — произвольный параметр датчика Modbus 86.
GENERIC_MODBUS_PARAM_87	87 — произвольный параметр датчика Modbus 87.
GENERIC_MODBUS_PARAM_88	88 — произвольный параметр датчика Modbus 88.
GENERIC_MODBUS_PARAM_89	89 — произвольный параметр датчика Modbus 89.
GENERIC_MODBUS_PARAM_90	90 — произвольный параметр датчика Modbus 90.
GENERIC_MODBUS_PARAM_91	91 — произвольный параметр датчика Modbus 91.
GENERIC_MODBUS_PARAM_92	92 — произвольный параметр датчика Modbus 92.
GENERIC_MODBUS_PARAM_93	93 — произвольный параметр датчика Modbus 93.
GENERIC_MODBUS_PARAM_94	94 — произвольный параметр датчика Modbus 94.
GENERIC_MODBUS_PARAM_95	95 — произвольный параметр датчика Modbus 95.
GENERIC_MODBUS_PARAM_96	96 — произвольный параметр датчика Modbus 96.
GENERIC_MODBUS_PARAM_97	97 — произвольный параметр датчика Modbus 97.
GENERIC_MODBUS_PARAM_98	98 — произвольный параметр датчика Modbus 98.
GENERIC_MODBUS_PARAM_99	99 — произвольный параметр датчика Modbus 99.
GENERIC_MODBUS_PARAM_100	100 — произвольный параметр датчика Modbus 100.
GENERIC_CAN_PARAM_1	101 — произвольный параметр CAN 1.
GENERIC_CAN_PARAM_2	102 — произвольный параметр CAN 2.
GENERIC_CAN_PARAM_3	103 — произвольный параметр CAN 3.
GENERIC_CAN_PARAM_4	104 — произвольный параметр CAN 4.
GENERIC_CAN_PARAM_5	105 — произвольный параметр CAN 5.
GENERIC_CAN_PARAM_6	106 — произвольный параметр CAN 6.
GENERIC_CAN_PARAM_7	107 — произвольный параметр CAN 7.
GENERIC_CAN_PARAM_8	108 — произвольный параметр CAN 8.

GENERIC_CAN_PARAM_9	109 — произвольный параметр CAN 9.
GENERIC_CAN_PARAM_10	110 — произвольный параметр CAN 10.
GENERIC_CAN_PARAM_11	111 — произвольный параметр CAN 11.
GENERIC_CAN_PARAM_12	112 — произвольный параметр CAN 12.
GENERIC_CAN_PARAM_13	113 — произвольный параметр CAN 13.
GENERIC_CAN_PARAM_14	114 — произвольный параметр CAN 14.
GENERIC_CAN_PARAM_15	115 — произвольный параметр CAN 15.
GENERIC_CAN_PARAM_16	116 — произвольный параметр CAN 16.
GENERIC_CAN_PARAM_17	117 — произвольный параметр CAN 17.
GENERIC_CAN_PARAM_18	118 — произвольный параметр CAN 18.
GENERIC_CAN_PARAM_19	119 — произвольный параметр CAN 19.
GENERIC_CAN_PARAM_20	120 — произвольный параметр CAN 20.
GENERIC_CAN_PARAM_21	121 — произвольный параметр CAN 21.
GENERIC_CAN_PARAM_22	122 — произвольный параметр CAN 22.
GENERIC_CAN_PARAM_23	123 — произвольный параметр CAN 23.
GENERIC_CAN_PARAM_24	124 — произвольный параметр CAN 24.
GENERIC_CAN_PARAM_25	125 — произвольный параметр CAN 25.
GENERIC_CAN_PARAM_26	126 — произвольный параметр CAN 26.
GENERIC_CAN_PARAM_27	127 — произвольный параметр CAN 27.
GENERIC_CAN_PARAM_28	128 — произвольный параметр CAN 28.
GENERIC_CAN_PARAM_29	129 — произвольный параметр CAN 29.
GENERIC_CAN_PARAM_30	130 — произвольный параметр CAN 30.
GENERIC_CAN_PARAM_31	131 — произвольный параметр CAN 31.
GENERIC_CAN_PARAM_32	132 — произвольный параметр CAN 32.
GENERIC_CAN_PARAM_33	133 — произвольный параметр CAN 33.
GENERIC_CAN_PARAM_34	134 — произвольный параметр CAN 34.
GENERIC_CAN_PARAM_35	135 — произвольный параметр CAN 35.
GENERIC_CAN_PARAM_36	136 — произвольный параметр CAN 36.
GENERIC_CAN_PARAM_37	137 — произвольный параметр CAN 37.
GENERIC_CAN_PARAM_38	138 — произвольный параметр CAN 38.
GENERIC_CAN_PARAM_39	139 — произвольный параметр CAN 39.
GENERIC_CAN_PARAM_40	140 — произвольный параметр CAN 40.
GENERIC_CAN_PARAM_41	141 — произвольный параметр CAN 41.
GENERIC_CAN_PARAM_42	142 — произвольный параметр CAN 42.
GENERIC_CAN_PARAM_43	143 — произвольный параметр CAN 43.
GENERIC_CAN_PARAM_44	144 — произвольный параметр CAN 44.

GENERIC_CAN_PARAM_45	145 — произвольный параметр CAN 45.
GENERIC_CAN_PARAM_46	146 — произвольный параметр CAN 46.
GENERIC_CAN_PARAM_47	147 — произвольный параметр CAN 47.
GENERIC_CAN_PARAM_48	148 — произвольный параметр CAN 48.
GENERIC_CAN_PARAM_49	149 — произвольный параметр CAN 49.
GENERIC_CAN_PARAM_50	150 — произвольный параметр CAN 50.

LevelId

Уровневые параметры.

Доступ к параметрам:

- сервер/SMS, чтение: команда GLEVELVALUE;
- сервер/SMS, запись: команда SETLEVELVALUE;
- Т.Скрипт, чтение: нативная функция `tkGetLevelValue`;
- Т.Скрипт, запись: нативная функция `tkSetLevelValue`.

Адаптивная обработка доступна для всех параметров кроме напряжения антенны (LEVEL_GNS_ANT_VDD) и напряжения внешнего аккумулятора (LEVEL_A_EXT_BATTERY). Возможность адаптивной записи отмечена для каждого параметра отдельно.

LEVEL_LLS1	<p>1 — уровень топлива с датчика 1, единицы измерения датчика.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: LLS1, LLS(1). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS2	<p>2 — уровень топлива с датчика 2, единицы измерения датчика.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: LLS2, LLS(2). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS3	<p>3 — уровень топлива с датчика 3, единицы измерения датчика.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: LLS3, LLS(3). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS4	<p>4 — уровень топлива с датчика 4, единицы измерения датчика.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: LLS4, LLS(4). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_LLS5	<p>5 — уровень топлива с датчика 5, единицы измерения датчика.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: LLS5, LLS(5). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS6	<p>6 — уровень топлива с датчика 6, единицы измерения датчика.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: LLS6, LLS(6). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS7	<p>7 — уровень топлива с датчика 7, единицы измерения датчика.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: LLS7, LLS(7). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS8	<p>8 — уровень топлива с датчика 8, единицы измерения датчика.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: LLS8, LLS(8). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_TEMP1	<p>9 — температура с датчика 1, °С.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, ACH, ГЛОНАРУС. • Выражение в ПО: Temper1, Temper(1). • Период записи: TEMPPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
LEVEL_TEMP2	<p>10 — температура с датчика 2, °С.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, ACH, ГЛОНАРУС. • Выражение в ПО: Temper2, Temper(2). • Период записи: TEMPPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.

LEVEL_TEMP3	<p>11 — температура с датчика 3, °С.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: Temper3, Temper(3). • Период записи: TEMPPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
LEVEL_TEMP4	<p>12 — температура с датчика 4, °С.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: Temper4, Temper(4). • Период записи: TEMPPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
LEVEL_TEMP5	<p>13 — температура с датчика 5, °С.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: Temper5, Temper(5). • Период записи: TEMPPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
LEVEL_TEMP6	<p>14 — температура с датчика 6, °С.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: Temper6, Temper(6). • Период записи: TEMPPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
LEVEL_TEMP7	<p>15 — температура с датчика 7, °С.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: Temper7, Temper(7). • Период записи: TEMPPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
LEVEL_TEMP8	<p>16 — температура с датчика 8, °С.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: Temper8, Temper(8). • Период записи: TEMPPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.

LEVEL_INT_CPU_TEMPSENSOR	<p>17 — температура МК, °С.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: DevT. • Период записи: TEMPPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
LEVEL_GNS_ANT_VDD	<p>19 — напряжение антенны, В.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Параметр недоступен на устройствах с внутренней антенной. Для параметра недоступна адаптивная обработка.</p>
LEVEL_EXT_VDD	<p>20 — напряжение внешнего питания, В.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС. • Выражение в ПО: MainVolt. • Период записи: PERIODANALOG. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
LEVEL_A_IN_1	<p>21 — напряжение аналогового входа 1, В.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС. • Выражение в ПО: Avolt(1), A1Volt. • Период записи: PERIODCOUNT12, PERIODANALOG. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Значение данного параметра сохраняется в две разные записи бинарного файла. В запись, соответствующую параметру Avolt(1) в ПО, напряжение сохраняется в полном диапазоне (30 В) с периодом PERIODCOUNT12. Возможно внеочередное сохранение записи при помощи адаптивной обработки. В запись, соответствующую параметру A1Volt в ПО, напряжение сохраняется в ограниченном диапазоне (10 В) с периодом PERIODANALOG. Внеочередное сохранение данной записи при помощи адаптивной обработки не выполняется.</p>

<p>LEVEL_A_IN_2</p>	<p>22 — напряжение аналогового входа 2, В.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX (S/N с 2409400), GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС. • Выражение в ПО: Avolt(2), A2Volt. • Период записи: PERIODCOUNT12, PERIODANALOG. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Значение данного параметра сохраняется в две разные записи бинарного файла. В запись, соответствующую параметру Avolt(2) в ПО, напряжение сохраняется в полном диапазоне (30 В) с периодом PERIODCOUNT12. Возможно внеочередное сохранение записи при помощи адаптивной обработки. В запись, соответствующую параметру A2Volt в ПО, напряжение сохраняется в ограниченном диапазоне (24 В) с периодом PERIODANALOG. Внеочередное сохранение данной записи при помощи адаптивной обработки не выполняется.</p>
<p>LEVEL_A_EXT_BATTERY</p>	<p>23 — напряжение внешнего аккумулятора, В.</p> <ul style="list-style-type: none"> • Доступен в: GX/GX WIFI, АСН. • Выражение в ПО: BackupVolt. • Период записи: PERIODANALOG. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Для параметра недоступна адаптивная обработка.</p>
<p>LEVEL_A_INT_BATTERY</p>	<p>24 — напряжение внутреннего аккумулятора, В.</p> <ul style="list-style-type: none"> • Доступен в: GX/GX WIFI, АСН, LogistiX, Mobile X. • Выражение в ПО: числовой параметр с типом 0x0000FE. • Период записи: INTERNALBATPERIOD. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Значение параметра сохраняется в бинарный файл только на АвтоГРАФ-Mobile X.</p>
<p>VEHICLE_SPEED_SPN_84</p>	<p>25 — скорость, км/ч.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANSpeed. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
<p>ACCEL_PEDAL_SPN_91</p>	<p>26 — педаль акселератора, %.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANGaz. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

FUEL_LEVEL_1_SPN_96	<p>27 — уровень топлива 1, %.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANL1, CANL(1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
FUEL_LEVEL_2_SPN_96	<p>28 — уровень топлива 2, %.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANL2, CANL(2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
FUEL_LEVEL_3_SPN_96	<p>29 — уровень топлива 3, %.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANL3, CANL(3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
FUEL_LEVEL_4_SPN_96	<p>30 — уровень топлива 4, %.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANL4, CANL(4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
FUEL_LEVEL_5_SPN_96	<p>31 — уровень топлива 5, %.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANL5, CANL(5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
FUEL_LEVEL_6_SPN_96	<p>32 — уровень топлива 6, %.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANL6, CANL(6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

ADBLUE_LEVEL_SPN_1761	<p>33 — уровень AdBlue, %.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANLAB. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
RPM_SPN_190	<p>34 — обороты, об/бит.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANErpm, CANErpmRaw. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
OIL_PRESSURE_SPN_100	<p>35 — давление масла, кПа.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANPoil. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
OIL_TEMP_SPN_175	<p>36 — температура масла, °С.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANToil. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
COOLANT_TEMP_SPN_110	<p>37 — температура охлаждающей жидкости, °С.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANTcool. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
FUEL_TEMP_SPN_174	<p>38 — температура топлива, °С.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANTfuel. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

AMBIENT_AIR_TEMP_SPN_171	<p>39 — внешняя температура, °С.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANTair. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
CHARGER_AIR_TEMP_SPN_105	<p>40 — температура в коллекторе наддува, °С.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANTboost. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
ENGINE_AIR_INLET_PRES_SPN_106	<p>41 — давление воздуха на впуске, кПа.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANPboost. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
ENGINE_CHGR_BOOST_PRES_SPN_102	<p>42 — избыточное давление наддува, кПа.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANPmap. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_1_1	<p>43 — нагрузка на колесо 1 оси 1, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(1, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_1_2	<p>44 — нагрузка на колесо 2 оси 1, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(1, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_1_3	<p>45 — нагрузка на колесо 3 оси 1, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(1, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_1_4	<p>46 — нагрузка на колесо 4 оси 1, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(1, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_1_5	<p>47 — нагрузка на колесо 5 оси 1, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(1, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_1_6	<p>48 — нагрузка на колесо 6 оси 1, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(1, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_2_1	<p>49 — нагрузка на колесо 1 оси 2, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(2, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_2_2	<p>50 — нагрузка на колесо 2 оси 2, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(2, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_2_3	<p>51 — нагрузка на колесо 3 оси 2, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(2, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_2_4	<p>52 — нагрузка на колесо 4 оси 2, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(2, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_2_5	<p>53 — нагрузка на колесо 5 оси 2, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(2, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_2_6	<p>54 — нагрузка на колесо 6 оси 2, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(2, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_3_1	<p>55 — нагрузка на колесо 1 оси 3, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(3, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_3_2	<p>56 — нагрузка на колесо 2 оси 3, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(3, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_3_3	<p>57 — нагрузка на колесо 3 оси 3, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(3, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_3_4	<p>58 — нагрузка на колесо 4 оси 3, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(3, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_3_5	<p>59 — нагрузка на колесо 5 оси 3, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(3, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_3_6	<p>60 — нагрузка на колесо 6 оси 3, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(3, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_4_1	<p>61 — нагрузка на колесо 1 оси 4, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(4, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_4_2	<p>62 — нагрузка на колесо 2 оси 4, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(4, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_4_3	<p>63 — нагрузка на колесо 3 оси 4, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(4, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_4_4	<p>64 — нагрузка на колесо 4 оси 4, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(4, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_4_5	<p>65 — нагрузка на колесо 5 оси 4, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(4, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_4_6	<p>66 — нагрузка на колесо 6 оси 4, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(4, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_5_1	<p>67 — нагрузка на колесо 1 оси 5, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(5, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_5_2	<p>68 — нагрузка на колесо 2 оси 5, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(5, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_5_3	<p>69 — нагрузка на колесо 3 оси 5, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(5, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_5_4	<p>70 — нагрузка на колесо 4 оси 5, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(5, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_5_5	<p>71 — нагрузка на колесо 5 оси 5, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(5, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_5_6	<p>72 — нагрузка на колесо 6 оси 5, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(5, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_6_1	<p>73 — нагрузка на колесо 1 оси 6, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(6, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_6_2	<p>74 — нагрузка на колесо 2 оси 6, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(6, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_6_3	<p>75 — нагрузка на колесо 3 оси 6, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(6, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_6_4	<p>76 — нагрузка на колесо 4 оси 6, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(6, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_6_5	<p>77 — нагрузка на колесо 5 оси 6, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(6, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_6_6	<p>78 — нагрузка на колесо 6 оси 6, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(6, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_7_1	<p>79 — нагрузка на колесо 1 оси 7, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(7, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_7_2	<p>80 — нагрузка на колесо 2 оси 7, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(7, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_7_3	81 — нагрузка на колесо 3 оси 7, кг. <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(7, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_7_4	82 — нагрузка на колесо 4 оси 7, кг. <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(7, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_7_5	83 — нагрузка на колесо 5 оси 7, кг. <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(7, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_7_6	84 — нагрузка на колесо 6 оси 7, кг. <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(7, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_8_1	85 — нагрузка на колесо 1 оси 8, кг. <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(8, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_8_2	86 — нагрузка на колесо 2 оси 8, кг. <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(8, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_8_3	<p>87 — нагрузка на колесо 3 оси 8, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(8, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_8_4	<p>88 — нагрузка на колесо 4 оси 8, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(8, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_8_5	<p>89 — нагрузка на колесо 5 оси 8, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(8, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_8_6	<p>90 — нагрузка на колесо 6 оси 8, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(8, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_9_1	<p>91 — нагрузка на колесо 1 оси 9, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(9, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_9_2	<p>92 — нагрузка на колесо 2 оси 9, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(9, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_9_3	<p>93 — нагрузка на колесо 3 оси 9, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(9, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_9_4	<p>94 — нагрузка на колесо 4 оси 9, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(9, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_9_5	<p>95 — нагрузка на колесо 5 оси 9, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(9, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_9_6	<p>96 — нагрузка на колесо 6 оси 9, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(9, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_10_1	<p>97 — нагрузка на колесо 1 оси 10, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(10, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_10_2	<p>98 — нагрузка на колесо 2 оси 10, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(10, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_10_3	<p>99 — нагрузка на колесо 3 оси 10, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(10, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_10_4	<p>100 — нагрузка на колесо 4 оси 10, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(10, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_10_5	<p>101 — нагрузка на колесо 5 оси 10, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(10, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_10_6	<p>102 — нагрузка на колесо 6 оси 10, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(10, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_11_1	<p>103 — нагрузка на колесо 1 оси 11, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(11, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_11_2	<p>104 — нагрузка на колесо 2 оси 11, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(11, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_11_3	<p>105 — нагрузка на колесо 3 оси 11, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(11, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_11_4	<p>106 — нагрузка на колесо 4 оси 11, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(11, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_11_5	<p>107 — нагрузка на колесо 5 оси 11, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(11, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_11_6	<p>108 — нагрузка на колесо 6 оси 11, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(11, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_12_1	<p>109 — нагрузка на колесо 1 оси 12, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(12, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_12_2	<p>110 — нагрузка на колесо 2 оси 12, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(12, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_12_3	<p>111 — нагрузка на колесо 3 оси 12, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(12, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_12_4	<p>112 — нагрузка на колесо 4 оси 12, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(12, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_12_5	<p>113 — нагрузка на колесо 5 оси 12, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(12, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_12_6	<p>114 — нагрузка на колесо 6 оси 12, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(12, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_13_1	<p>115 — нагрузка на колесо 1 оси 13, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(13, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_13_2	<p>116 — нагрузка на колесо 2 оси 13, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(13, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_13_3	<p>117 — нагрузка на колесо 3 оси 13, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(13, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_13_4	<p>118 — нагрузка на колесо 4 оси 13, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(13, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_13_5	<p>119 — нагрузка на колесо 5 оси 13, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(13, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_13_6	<p>120 — нагрузка на колесо 6 оси 13, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(13, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_14_1	<p>121 — нагрузка на колесо 1 оси 14, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(14, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_14_2	<p>122 — нагрузка на колесо 2 оси 14, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(14, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_14_3	<p>123 — нагрузка на колесо 3 оси 14, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(14, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_14_4	<p>124 — нагрузка на колесо 4 оси 14, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(14, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_14_5	<p>125 — нагрузка на колесо 5 оси 14, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(14, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_14_6	<p>126 — нагрузка на колесо 6 оси 14, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(14, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_15_1	<p>127 — нагрузка на колесо 1 оси 15, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(15, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_15_2	<p>128 — нагрузка на колесо 2 оси 15, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(15, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_15_3	<p>129 — нагрузка на колесо 3 оси 15, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(15, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_15_4	<p>130 — нагрузка на колесо 4 оси 15, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(15, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_15_5	<p>131 — нагрузка на колесо 5 оси 15, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(15, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_15_6	<p>132 — нагрузка на колесо 6 оси 15, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(15, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_16_1	<p>133 — нагрузка на колесо 1 оси 16, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(16, 1). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_16_2	<p>134 — нагрузка на колесо 2 оси 16, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(16, 2). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_WHEEL_LOAD_16_3	<p>135 — нагрузка на колесо 3 оси 16, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(16, 3). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_16_4	<p>136 — нагрузка на колесо 4 оси 16, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(16, 4). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_16_5	<p>137 — нагрузка на колесо 5 оси 16, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(16, 5). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_WHEEL_LOAD_16_6	<p>138 — нагрузка на колесо 6 оси 16, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANAW(16, 6). • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
ENGINE_FUEL_RATE_SPN_183	<p>139 — расход топлива в единицу времени, л/ч.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANFInstant. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
ENGINE_THROTTLE_POS_SPN_51	<p>140 — положение дроссельной заслонки, %.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANChoker. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

ACTUAL_ENG_PCNT_TORQUE_SPN_513	<p>141 — действующий момент, %.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANTorquePercent. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
CRUISE_CONTROL_SET_SPEED_SPN_86	<p>142 — скорость круиз-контроля, км/ч.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANCruiseSpeed. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
NOMINAL_FRICT_PCNT_TRQ_SPN_514	<p>143 — номинальное трение — % крутящего момента, %.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANFrictionPercent. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
BATTERY_VOLTAGE_SPN_158	<p>144 — напряжение АКБ, В.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: BatteryVolt. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
BAROMETRIC_PRESSURE_SPN_108	<p>145 — абсолютное атмосферное давление, кПа.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANPair. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
ENGINE_LOAD_SPN_92	<p>146 — нагрузка на двигатель, %.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: CANELoad. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

BATTERY_CURRENT_SPN_114	<p>147 — ток АКБ, А.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, ACH, ГЛОНАРУС. • Выражение в ПО: BatteryAmp. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
PARTICULATE_FILTER	<p>148 — сажевый фильтр, %.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, ACH, ГЛОНАРУС. • Выражение в ПО: числовой параметр с типом 0xFBFF02. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_FREQUENCY_01	<p>149 — частота или ШИМ со входа 1.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: F1, F(1). • Период записи: PERIODCOUNT12. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Если и вход 1, и вход 2 работают в дискретном режиме, то данные, при их наличии, сохраняются в бинарный файл с периодом PERIODCOUNT12. При отсутствии данных запись сохраняется с периодом 10 минут. Период сохранения не зависит от STOPRECORDSMULTIPLIER и факта остановки. Если хотя бы один из этих входов настроен в аналоговом режиме, сохранение данных происходит с периодом PERIODCOUNT12 с учетом STOPRECORDSMULTIPLIER и факта остановки.</p>
LEVEL_FREQUENCY_02	<p>150 — частота или ШИМ со входа 2.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: F2, F(2). • Период записи: PERIODCOUNT12. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Если и вход 1, и вход 2 работают в дискретном режиме, то данные, при их наличии, сохраняются в бинарный файл с периодом PERIODCOUNT12. При отсутствии данных запись сохраняется с периодом 10 минут. Период сохранения не зависит от STOPRECORDSMULTIPLIER и факта остановки. Если хотя бы один из этих входов настроен в аналоговом режиме, сохранение данных происходит с периодом PERIODCOUNT12 с учетом STOPRECORDSMULTIPLIER и факта остановки.</p>

<p>LEVEL_FREQUENCY_03</p>	<p>151 — частота или ШИМ со входа 3.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: F3, F(3). • Период записи: PERIODCOUNT34. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Если и вход 3, и вход 4 работают в дискретном режиме, то данные, при их наличии, сохраняются в бинарный файл с периодом PERIODCOUNT34. При отсутствии данных запись сохраняется с периодом 10 минут. Период сохранения не зависит от STOPRECORDSMULTIPLIER и факта остановки. Если хотя бы один из этих входов настроен в аналоговом режиме, сохранение данных происходит с периодом PERIODCOUNT34 с учетом STOPRECORDSMULTIPLIER и факта остановки.</p>
<p>LEVEL_FREQUENCY_04</p>	<p>152 — частота или ШИМ со входа 4.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), GX/GX WIFI, ACH. • Выражение в ПО: F4, F(4). • Период записи: PERIODCOUNT34. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Если и вход 3, и вход 4 работают в дискретном режиме, то данные, при их наличии, сохраняются в бинарный файл с периодом PERIODCOUNT34. При отсутствии данных запись сохраняется с периодом 10 минут. Период сохранения не зависит от STOPRECORDSMULTIPLIER и факта остановки. Если хотя бы один из этих входов настроен в аналоговом режиме, сохранение данных происходит с периодом PERIODCOUNT34 с учетом STOPRECORDSMULTIPLIER и факта остановки.</p>
<p>LEVEL_FREQUENCY_05</p>	<p>153 — частота или ШИМ со входа 5.</p> <ul style="list-style-type: none"> • Доступен в: GX/GX WIFI, ACH. • Выражение в ПО: F5, F(5). • Период записи: PERIODCOUNT56. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Если и вход 5, и вход 6 работают в дискретном режиме, то данные, при их наличии, сохраняются в бинарный файл с периодом PERIODCOUNT56. При отсутствии данных запись сохраняется с периодом 10 минут. Период сохранения не зависит от STOPRECORDSMULTIPLIER и факта остановки. Если хотя бы один из этих входов настроен в аналоговом режиме, сохранение данных происходит с периодом PERIODCOUNT56 с учетом STOPRECORDSMULTIPLIER и факта остановки.</p>

<p>LEVEL_FREQUENCY_06</p>	<p>154 — частота или ШИМ со входа 6.</p> <ul style="list-style-type: none"> • Доступен в: GX/GX WIFI, АСН. • Выражение в ПО: F6, F(6). • Период записи: PERIODCOUNT56. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Если и вход 5, и вход 6 работают в дискретном режиме, то данные, при их наличии, сохраняются в бинарный файл с периодом PERIODCOUNT56. При отсутствии данных запись сохраняется с периодом 10 минут. Период сохранения не зависит от STOPRECORDSMULTIPLIER и факта остановки. Если хотя бы один из этих входов настроен в аналоговом режиме, сохранение данных происходит с периодом PERIODCOUNT56 с учетом STOPRECORDSMULTIPLIER и факта остановки.</p>
<p>LEVEL_FREQUENCY_07</p>	<p>155 — частота или ШИМ со входа 7.</p> <ul style="list-style-type: none"> • Доступен в: GX/GX WIFI (S/N с 3105200/3023200), АСН. • Выражение в ПО: F7, F(7). • Период записи: PERIODCOUNT78. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Если и вход 7, и вход 8 работают в дискретном режиме, то данные, при их наличии, сохраняются в бинарный файл с периодом PERIODCOUNT78. При отсутствии данных запись сохраняется с периодом 10 минут. Период сохранения не зависит от STOPRECORDSMULTIPLIER и факта остановки. Если хотя бы один из этих входов настроен в аналоговом режиме, сохранение данных происходит с периодом PERIODCOUNT78 с учетом STOPRECORDSMULTIPLIER и факта остановки.</p>
<p>LEVEL_FREQUENCY_08</p>	<p>156 — частота или ШИМ со входа 8.</p> <ul style="list-style-type: none"> • Доступен в: GX/GX WIFI (S/N с 3105200/3023200), АСН. • Выражение в ПО: F8, F(8). • Период записи: PERIODCOUNT78. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Если и вход 7, и вход 8 работают в дискретном режиме, то данные, при их наличии, сохраняются в бинарный файл с периодом PERIODCOUNT78. При отсутствии данных запись сохраняется с периодом 10 минут. Период сохранения не зависит от STOPRECORDSMULTIPLIER и факта остановки. Если хотя бы один из этих входов настроен в аналоговом режиме, сохранение данных происходит с периодом PERIODCOUNT78 с учетом STOPRECORDSMULTIPLIER и факта остановки.</p>
<p>LEVEL_FREQUENCY_RPM</p>	<p>158 — частота или ШИМ со входа RPM.</p> <ul style="list-style-type: none"> • Доступен в: GX/GX WIFI, АСН. • Выражение в ПО: FRPM. • Период записи: FREQPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.

<p>LEVEL_LLS_1_TEMPERATURE</p>	<p>159 — температура с ДУТ 1, °С.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS1, TLLS(1). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>
<p>LEVEL_LLS_2_TEMPERATURE</p>	<p>160 — температура с ДУТ 2, °С.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS2, TLLS(2). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>
<p>LEVEL_LLS_3_TEMPERATURE</p>	<p>161 — температура с ДУТ 3, °С.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS3, TLLS(3). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>
<p>LEVEL_LLS_4_TEMPERATURE</p>	<p>162 — температура с ДУТ 4, °С.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS4, TLLS(4). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>

<p>LEVEL_LLS_5_TEMPERATURE</p>	<p>163 — температура с ДУТ 5, °С.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS5, TLLS(5). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>
<p>LEVEL_LLS_6_TEMPERATURE</p>	<p>164 — температура с ДУТ 6, °С.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS6, TLLS(6). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>
<p>LEVEL_LLS_7_TEMPERATURE</p>	<p>165 — температура с ДУТ 7, °С.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS7, TLLS(7). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>
<p>LEVEL_LLS_8_TEMPERATURE</p>	<p>166 — температура с ДУТ 8, °С.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TLLS8, TLLS(8). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: В данный параметр в ПО попадает и значение температуры, и код ошибки. Значение меньше -110 является кодом ошибки. Значение -110 и выше является температурой.</p>

LEVEL_LLS_1_ANGLE	<p>167 — угол с ДУТ 1, градусы (0...180°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: ALLS(1). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_2_ANGLE	<p>168 — угол с ДУТ 2, градусы (0...180°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: ALLS(2). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_3_ANGLE	<p>169 — угол с ДУТ 3, градусы (0...180°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: ALLS(3). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_4_ANGLE	<p>170 — угол с ДУТ 4, градусы (0...180°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: ALLS(4). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_5_ANGLE	<p>171 — угол с ДУТ 5, градусы (0...180°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: ALLS(5). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_6_ANGLE	<p>172 — угол с ДУТ 6, градусы (0...180°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: ALLS(6). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_LLS_7_ANGLE	<p>173 — угол с ДУТ 7, градусы (0...180°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: ALLS(7). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_8_ANGLE	<p>174 — угол с ДУТ 8, градусы (0...180°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: ALLS(8). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_1_PITCH	<p>175 — угол тангажа с ДУТ 1, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: FLLS(1). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_2_PITCH	<p>176 — угол тангажа с ДУТ 2, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: FLLS(2). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_3_PITCH	<p>177 — угол тангажа с ДУТ 3, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: FLLS(3). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_4_PITCH	<p>178 — угол тангажа с ДУТ 4, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: FLLS(4). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_LLS_5_PITCH	<p>179 — угол тангажа с ДУТ 5, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: FLLS(5). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_6_PITCH	<p>180 — угол тангажа с ДУТ 6, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: FLLS(6). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_7_PITCH	<p>181 — угол тангажа с ДУТ 7, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: FLLS(7). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_8_PITCH	<p>182 — угол тангажа с ДУТ 8, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: FLLS(8). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_1_ROLL	<p>183 — угол крена с ДУТ 1, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: SLLS(1). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_2_ROLL	<p>184 — угол крена с ДУТ 2, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: SLLS(2). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_LLS_3_ROLL	<p>185 — угол крена с ДУТ 3, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: SLLS(3). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_4_ROLL	<p>186 — угол крена с ДУТ 4, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: SLLS(4). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_5_ROLL	<p>187 — угол крена с ДУТ 5, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: SLLS(5). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_6_ROLL	<p>188 — угол крена с ДУТ 6, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: SLLS(6). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_7_ROLL	<p>189 — угол крена с ДУТ 7, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: SLLS(7). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_8_ROLL	<p>190 — угол крена с ДУТ 8, градусы (-90°...90°).</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: SLLS(8). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_LLS_1_FREQUENCY	<p>191 — частота с ДУТ 1, Гц.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: Значение параметра в бинарный файл не сохраняется, но для параметра доступна адаптивная обработка. Параметр может служить источником события «Адаптив».</p>
LEVEL_LLS_2_FREQUENCY	<p>192 — частота с ДУТ 2, Гц.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: Значение параметра в бинарный файл не сохраняется, но для параметра доступна адаптивная обработка. Параметр может служить источником события «Адаптив».</p>
LEVEL_LLS_3_FREQUENCY	<p>193 — частота с ДУТ 3, Гц.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: Значение параметра в бинарный файл не сохраняется, но для параметра доступна адаптивная обработка. Параметр может служить источником события «Адаптив».</p>
LEVEL_LLS_4_FREQUENCY	<p>194 — частота с ДУТ 4, Гц.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: Значение параметра в бинарный файл не сохраняется, но для параметра доступна адаптивная обработка. Параметр может служить источником события «Адаптив».</p>

LEVEL_LLS_5_FREQUENCY	<p>195 — частота с ДУТ 5, Гц.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: Значение параметра в бинарный файл не сохраняется, но для параметра доступна адаптивная обработка. Параметр может служить источником события «Адаптив».</p>
LEVEL_LLS_6_FREQUENCY	<p>196 — частота с ДУТ 6, Гц.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: Значение параметра в бинарный файл не сохраняется, но для параметра доступна адаптивная обработка. Параметр может служить источником события «Адаптив».</p>
LEVEL_LLS_7_FREQUENCY	<p>197 — частота с ДУТ 7, Гц.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: Значение параметра в бинарный файл не сохраняется, но для параметра доступна адаптивная обработка. Параметр может служить источником события «Адаптив».</p>
LEVEL_LLS_8_FREQUENCY	<p>198 — частота с ДУТ 8, Гц.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись. <p>ВНИМАНИЕ: Значение параметра в бинарный файл не сохраняется, но для параметра доступна адаптивная обработка. Параметр может служить источником события «Адаптив».</p>
LEVEL_A_IN_3	<p>199 — напряжение аналогового входа 3, В.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX (S/N с 2409400), GX/GX WIFI, АСН, LogistiX, ГЛОНАРУС. • Выражение в ПО: Avolt(3). • Период записи: PERIODCOUNT34. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.

LEVEL_A_IN_4	<p>200 — напряжение аналогового входа 4, В.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), GX/GX WIFI, АСН. • Выражение в ПО: Avolt(4). • Период записи: PERIODCOUNT34. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
LEVEL_COUPLER_LOAD	<p>201 — нагрузка на сцепное устройство, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: TWCouplerLoad. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_CARGO_WEIGHT_SPN_181	<p>202 — вес груза, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: TWLoadWeight. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_TRAILER_WEIGHT_SPN_180	<p>203 — вес трейлера (прицепа), кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС. • Выражение в ПО: TWTrailerWeight. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_A_IN_5	<p>204 — напряжение аналогового входа 5, В.</p> <ul style="list-style-type: none"> • Доступен в: GX/GX WIFI, АСН. • Выражение в ПО: Avolt(5). • Период записи: PERIODCOUNT56. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
LEVEL_A_IN_6	<p>205 — напряжение аналогового входа 6, В.</p> <ul style="list-style-type: none"> • Доступен в: GX/GX WIFI, АСН. • Выражение в ПО: Avolt(6). • Период записи: PERIODCOUNT56. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.

LEVEL_NAV_SPEED	<p>206 — скорость с навигационного приемника, км/ч.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: SpeedRaw. • Период записи: PERIODWR. • Настройки, влияющие на период записи: STOPCOORDINATESMULTIPLIER, STOPTRANSMITMULTIPLIER, MODEMmPERIODSENDx, MODEMmPERIODROAMINGx, WIFIPERIODSEND. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
LEVEL_LLS_1_BAT_VOLT	<p>207 — напряжение батареи беспроводного ДУТ 1, В.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: VoltLLS(1). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_2_BAT_VOLT	<p>208 — напряжение батареи беспроводного ДУТ 2, В.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: VoltLLS(2). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_3_BAT_VOLT	<p>209 — напряжение батареи беспроводного ДУТ 3, В.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: VoltLLS(3). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_4_BAT_VOLT	<p>210 — напряжение батареи беспроводного ДУТ 4, В.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: VoltLLS(4). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_5_BAT_VOLT	<p>211 — напряжение батареи беспроводного ДУТ 5, В.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: VoltLLS(5). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_LLS_6_BAT_VOLT	<p>212 — напряжение батареи беспроводного ДУТ 6, В.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: VoltLLS(6). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_7_BAT_VOLT	<p>213 — напряжение батареи беспроводного ДУТ 7, В.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: VoltLLS(7). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_8_BAT_VOLT	<p>214 — напряжение батареи беспроводного ДУТ 8, В.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: VoltLLS(8). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_1_RSSI	<p>215 — RSSI беспроводного ДУТ 1.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: RSSILLS(1). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_2_RSSI	<p>216 — RSSI беспроводного ДУТ 2.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: RSSILLS(2). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_3_RSSI	<p>217 — RSSI беспроводного ДУТ 3.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: RSSILLS(3). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

LEVEL_LLS_4_RSSI	<p>218 — RSSI беспроводного ДУТ 4.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: RSSILLS(4). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_5_RSSI	<p>219 — RSSI беспроводного ДУТ 5.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: RSSILLS(5). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_6_RSSI	<p>220 — RSSI беспроводного ДУТ 6.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: RSSILLS(6). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_7_RSSI	<p>221 — RSSI беспроводного ДУТ 7.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: RSSILLS(7). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_LLS_8_RSSI	<p>222 — RSSI беспроводного ДУТ 8.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: RSSILLS(8). • Период записи: LLSPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
TKAM_1_ANGLE	<p>223 — угол с датчика угла наклона TKAM 1.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(1). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</p>

<p>TKAM_2_ANGLE</p>	<p>224 — угол с датчика угла наклона TKAM 2.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(2). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_3_ANGLE</p>	<p>225 — угол с датчика угла наклона TKAM 3.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(3). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_4_ANGLE</p>	<p>226 — угол с датчика угла наклона TKAM 4.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(4). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_5_ANGLE</p>	<p>227 — угол с датчика угла наклона TKAM 5.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(5). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_6_ANGLE</p>	<p>228 — угол с датчика угла наклона TKAM 6.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(6). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_7_ANGLE</p>	<p>229 — угол с датчика угла наклона TKAM 7.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(7). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_8_ANGLE</p>	<p>230 — угол с датчика угла наклона TKAM 8.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(8). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_9_ANGLE</p>	<p>231 — угол с датчика угла наклона TKAM 9.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(9). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_10_ANGLE</p>	<p>232 — угол с датчика угла наклона TKAM 10.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(10). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_11_ANGLE</p>	<p>233 — угол с датчика угла наклона TKAM 11.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(11). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_12_ANGLE</p>	<p>234 — угол с датчика угла наклона TKAM 12.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(12). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_13_ANGLE</p>	<p>235 — угол с датчика угла наклона TKAM 13.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(13). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_14_ANGLE</p>	<p>236 — угол с датчика угла наклона TKAM 14.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(14). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</p>
<p>TKAM_15_ANGLE</p>	<p>237 — угол с датчика угла наклона TKAM 15.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(15). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</p>
<p>TKAM_16_ANGLE</p>	<p>238 — угол с датчика угла наклона TKAM 16.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMAngle(16). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</p>
<p>TKAM_1_ROLL</p>	<p>239 — крен с датчика угла наклона TKAM 1.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(1). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</p>

<p>TKAM_2_ROLL</p>	<p>240 — крен с датчика угла наклона TKAM 2.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(2). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_3_ROLL</p>	<p>241 — крен с датчика угла наклона TKAM 3.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(3). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_4_ROLL</p>	<p>242 — крен с датчика угла наклона TKAM 4.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(4). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_5_ROLL</p>	<p>243 — крен с датчика угла наклона TKAM 5.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(5). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_6_ROLL</p>	<p>244 — крен с датчика угла наклона TKAM 6.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(6). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_7_ROLL</p>	<p>245 — крен с датчика угла наклона TKAM 7.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(7). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_8_ROLL</p>	<p>246 — крен с датчика угла наклона TKAM 8.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(8). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_9_ROLL</p>	<p>247 — крен с датчика угла наклона TKAM 9.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(9). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_10_ROLL</p>	<p>248 — крен с датчика угла наклона TKAM 10.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(10). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_11_ROLL</p>	<p>249 — крен с датчика угла наклона TKAM 11.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(11). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_12_ROLL</p>	<p>250 — крен с датчика угла наклона TKAM 12.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(12). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_13_ROLL</p>	<p>251 — крен с датчика угла наклона TKAM 13.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(13). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_14_ROLL</p>	<p>252 — крен с датчика угла наклона TKAM 14.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(14). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_15_ROLL</p>	<p>253 — крен с датчика угла наклона TKAM 15.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(15). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_16_ROLL</p>	<p>254 — крен с датчика угла наклона TKAM 16.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMRoulis(16). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_1_PITCH</p>	<p>255 — тангаж с датчика угла наклона TKAM 1.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(1). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_2_PITCH</p>	<p>256 — тангаж с датчика угла наклона TKAM 2.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(2). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_3_PITCH</p>	<p>257 — тангаж с датчика угла наклона TKAM 3.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(3). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_4_PITCH</p>	<p>258 — тангаж с датчика угла наклона TKAM 4.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(4). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_5_PITCH</p>	<p>259 — тангаж с датчика угла наклона TKAM 5.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(5). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_6_PITCH</p>	<p>260 — тангаж с датчика угла наклона TKAM 6.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(6). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_7_PITCH</p>	<p>261 — тангаж с датчика угла наклона TKAM 7.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(7). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_8_PITCH</p>	<p>262 — тангаж с датчика угла наклона TKAM 8.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(8). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_9_PITCH</p>	<p>263 — тангаж с датчика угла наклона TKAM 9.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(9). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_10_PITCH</p>	<p>264 — тангаж с датчика угла наклона TKAM 10.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(10). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_11_PITCH</p>	<p>265 — тангаж с датчика угла наклона TKAM 11.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(11). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_12_PITCH</p>	<p>266 — тангаж с датчика угла наклона TKAM 12.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(12). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_13_PITCH</p>	<p>267 — тангаж с датчика угла наклона TKAM 13.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(13). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_14_PITCH</p>	<p>268 — тангаж с датчика угла наклона TKAM 14.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(14). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_15_PITCH</p>	<p>269 — тангаж с датчика угла наклона TKAM 15.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(15). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_16_PITCH</p>	<p>270 — тангаж с датчика угла наклона TKAM 16.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: TKAMTangage(16). • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>BRAKE_AIR_PRESSURE_1_SPN_1087</p>	<p>271 — давление в первом тормозном контуре, кПа.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, ACH, ГЛОНАРУС. • Выражение в ПО: VSBC1Pressure. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
<p>BRAKE_AIR_PRESSURE_2_SPN_1088</p>	<p>272 — давление во втором тормозном контуре, кПа.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, ACH, ГЛОНАРУС. • Выражение в ПО: VSBC2Pressure. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

GROSS_VEHICLE_WEIGHT_SPN_1760	<p>273 — общий вес автомобиля, кг.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, ACH, ГЛОНАРУС. • Выражение в ПО: VSTotalWeight. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
LEVEL_A_IN_7	<p>274 — напряжение аналогового входа 7, В.</p> <ul style="list-style-type: none"> • Доступен в: GX/GX WIFI (S/N с 3105200/3023200), ACH. • Выражение в ПО: Avolt(7). • Период записи: PERIODCOUNT78. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
LEVEL_A_IN_8	<p>275 — напряжение аналогового входа 8, В.</p> <ul style="list-style-type: none"> • Доступен в: GX/GX WIFI (S/N с 3105200/3023200), ACH. • Выражение в ПО: Avolt(8). • Период записи: PERIODCOUNT78. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: да. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение.
LEVEL_AKB_CHARGE_PERCENT	<p>277 — процент заряда батареи.</p> <ul style="list-style-type: none"> • Доступен в: Mobile X. • Выражение в ПО: числовой параметр с типом 0x0000FE. • Период записи: INTERNALBATPERIOD. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: нет.
TKAM_1_RPM	<p>278 — частота вращения датчика угла наклона TKAM 1, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</p>

<p>TKAM_2_RPM</p>	<p>279 — частота вращения датчика угла наклона TKAM 2, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_3_RPM</p>	<p>280 — частота вращения датчика угла наклона TKAM 3, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_4_RPM</p>	<p>281 — частота вращения датчика угла наклона TKAM 4, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_5_RPM</p>	<p>282 — частота вращения датчика угла наклона TKAM 5, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_6_RPM</p>	<p>283 — частота вращения датчика угла наклона TKAM 6, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_7_RPM</p>	<p>284 — частота вращения датчика угла наклона TKAM 7, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_8_RPM</p>	<p>285 — частота вращения датчика угла наклона TKAM 8, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_9_RPM</p>	<p>286 — частота вращения датчика угла наклона TKAM 9, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_10_RPM</p>	<p>287 — частота вращения датчика угла наклона TKAM 10, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_11_RPM</p>	<p>288 — частота вращения датчика угла наклона TKAM 11, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_12_RPM</p>	<p>289 — частота вращения датчика угла наклона TKAM 12, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_13_RPM</p>	<p>290 — частота вращения датчика угла наклона TKAM 13, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>

<p>TKAM_14_RPM</p>	<p>291 — частота вращения датчика угла наклона TKAM 14, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_15_RPM</p>	<p>292 — частота вращения датчика угла наклона TKAM 15, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>TKAM_16_RPM</p>	<p>293 — частота вращения датчика угла наклона TKAM 16, об/мин.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС. • Выражение в ПО: нет. • Период записи: TKAMPERIOD, TKAM. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение. <p>ВНИМАНИЕ: <i>Запись сохраняется в бинарный файл только при наличии новых данных с момента предыдущего сохранения. Внеочередное сохранение данных выполняется при изменении состояния выходов датчика.</i></p>
<p>LEVEL_RAM_PARAM_1</p>	<p>294 — произвольный уровневый параметр RAM 1.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.
<p>LEVEL_RAM_PARAM_2</p>	<p>295 — произвольный уровневый параметр RAM 2.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.

LEVEL_RAM_PARAM_3	<p>296 — произвольный уровневый параметр RAM 3.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.
LEVEL_RAM_PARAM_4	<p>297 — произвольный уровневый параметр RAM 4.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.
LEVEL_RAM_PARAM_5	<p>298 — произвольный уровневый параметр RAM 5.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.
LEVEL_RAM_PARAM_6	<p>299 — произвольный уровневый параметр RAM 6.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.
LEVEL_RAM_PARAM_7	<p>300 — произвольный уровневый параметр RAM 7.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.
LEVEL_RAM_PARAM_8	<p>301 — произвольный уровневый параметр RAM 8.</p> <ul style="list-style-type: none"> • Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X. • Выражение в ПО: нет. • Период записи: нет. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение и запись • Доступ из Т.Скрипт: чтение и запись.

LEVEL_RAM_PARAM_9	302 — произвольный уровневый параметр RAM 9. <ul style="list-style-type: none">• Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X.• Выражение в ПО: нет.• Период записи: нет.• Адаптивная запись: нет.• Доступ через сервер/SMS: чтение и запись• Доступ из Т.Скрипт: чтение и запись.
LEVEL_RAM_PARAM_10	303 — произвольный уровневый параметр RAM 10. <ul style="list-style-type: none">• Доступен в: LX/LX(E), SX, GX/GX WIFI, ACH, LogistiX, ГЛОНАРУС, Mobile X.• Выражение в ПО: нет.• Период записи: нет.• Адаптивная запись: нет.• Доступ через сервер/SMS: чтение и запись• Доступ из Т.Скрипт: чтение и запись.

LongParamId

Длинные параметры.

Доступ к параметрам:

- сервер/SMS, чтение: команда GLONGVALUE;
- сервер/SMS, запись: нет;
- Т.Скрипт, чтение: нативная функция tkGetLongValue;
- Т.Скрипт, запись: нативная функция tkSetLongValue.

Адаптивная обработка недоступна.

TOTAL_FUEL_USED_SPN_250	<p>1 — суммарно использованное топливо, л.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, ACH, ГЛОНАРУС. • Выражение в ПО: CANFtotalAbs, CANFtotal. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
SERVICE_DISTANCE_SPN_914	<p>2 — пробег до ТО, км.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, ACH, ГЛОНАРУС. • Выражение в ПО: CANDmaint. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
ENGINE_HOURS_SPN_247	<p>3 — моточасы, ч.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, ACH, ГЛОНАРУС. • Выражение в ПО: CANEmhAbs, CANEmh. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.
TOTAL_VEHICLE_DISTANCE_SPN_917	<p>4 — полный пробег, м.</p> <ul style="list-style-type: none"> • Доступен в: SX, GX/GX WIFI, ACH, ГЛОНАРУС. • Выражение в ПО: CANDtotalAbs, CANDtotal. • Период записи: CANPERIOD. • Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER. • Адаптивная запись: нет. • Доступ через сервер/SMS: чтение. • Доступ из Т.Скрипт: чтение и запись.

TRIP_DISTANCE_SPN_918	5 — пробег за поездку, м. <ul style="list-style-type: none">• Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС.• Выражение в ПО: CANDaily.• Период записи: CANPERIOD.• Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER.• Адаптивная запись: нет.• Доступ через сервер/SMS: чтение.• Доступ из Т.Скрипт: чтение и запись.
CALCULATED_FUEL_CONSUMPTION	6 — потребление топлива, вычисленное по мгновенному расходу с прошлой записи, л. <ul style="list-style-type: none">• Доступен в: SX, GX/GX WIFI, АСН, ГЛОНАРУС.• Выражение в ПО: CANFcalc.• Период записи: CANPERIOD.• Настройки, влияющие на период записи: STOPRECORDSMULTIPLIER.• Адаптивная запись: нет.• Доступ через сервер/SMS: чтение.• Доступ из Т.Скрипт: чтение и запись.



ООО НПО «ТехноКом»

www.glonassgps.ru

30.03.2026

info@tk-nav.ru