Сервер Коммуникаций ag-server.

# Данный сервер выполняет следующие задачи:

1. Осуществляет обмен информацией с контроллерами АСУД.
2. Обновляет базу состояний контроллеров и перекрестков.
3. Принимает команды от сервера приложений и отправляет их на контроллер.
4. Поддерживает в персистентном состоянии базу данных.
5. Производит первоначальное создание базы данных и наполнение ее первоначальными данными.
6. Выполняет логгирование изменения состояния контроллеров, а так-же ошибок во время функционирования программы.

Программа написана на языке Golang и функционально разбита на модули:

* main.go — Собственно главный модуль программы в котором производится первоначльная настройка всех модулей и из вызов.
* Binding — Пакет программ для описания и управления массивами привязки контроллеров.
* Comm — Пакет программ собственно для связи с контроллерами АСУД.
* Extcon — Пакет программ управления состоянием модулей сервера.
* Inspect — Пакет программ поддержания соответствия состояния контроллера заданными на перекрестке.
* Pudge — Пакет программ сопровождения базы данных системы.
* Setup — Пакет программ приема и настройки сервера.
* Test — Локальные тесты системы.
* Transport — Пакет программ транспортного протокола поверх TCP/IP для контроллеров АСУД.

# База Данных сервера.

База данных сервера построена на основе PostgreSQL. В состав базы данных входят следующие таблицы:

* Accounts — Данные пользователей с правами.
* Cross — Описание и настройки перекрестков.
* Devices — Состояние контроллеров АСУД.
* Log — Текущие логи контроллеров.
* Region — Справочник наименований регионов и районов.

Необходимо отметить, что в таблице cross ключ состоит из триады <код\_региона,код\_района,номер\_перекрестка>. Тем самым производится ограничение в номере перекрестка в один байт.

В общем случае, база данных и программное обеспечение системы позволяет одновременную эксплуатацию нескольких регионов на одном сервере. Предварительные испытания на имитаторах показало, что система без особенного напряжения выдерживает до десяти тысяч подключенных контролеров АСУД.

При первоначальном запуске сервере в режиме создания производится создание базы данных. Все необходимые скрипты должны быть расположены в каталоге data сервера. Они разделяюся на следующие типы:

creator.xml — Описывает коды и названия регионов системы, коды и наименования районов каждого региона, а также имени файла с первоначальными настройками и описаниями перекрестков.

Cross.sql — Программа на языке SQL для создания таблицы cross.

Devices.sql — Программа на языке SQL для создания таблицы devices.

Log.sql — Программа на языке SQL для создания таблицы log.

Region.sql — Программа на языке SQL для создания таблицы region.

Файлы с расширение mrk содержат описания перекрестков данного региона.

# Главный модуль.

Назначение данного модуля заключается в подготовке к запуске всех остальных модулей системы. Первоначально подготавливается модуль логгирования сервера, как правило файл логов сервера располагается в каталоге «~/log/ag-server». Имя файла лога имеет структуру «logYYYY-MM-DD.log». Естественно при наступлении новых суток предыдущий файл лога закрывается и создается с новым именем файла.

Затем производится загрузка настроечного файла сервера, который имеет следующий путь «~/setup/setup\_ag.json». В этом файле прописывается настройка подключения к базе данных PostgreSQL. Кроме этого указывается номер порта в котором производится прием данных с контроллерами АСУД. Дополнительно номера портов по которым ожидается прием команд и массивов привязки от сервера приложений. И естественно все необходимые тайм-ауты.

Если обнаружены ошибки на этом этапе сервер завершает свою работу с выдачей диагностического сообщения.

Затем вызывается модуль pudge в котором производится работа с базой данных.

Затем вызывается модуль comm который производит обмен с контроллерами АСУД.

Затем производится запуск модуля inspect который начинает контролировать состояние перекрестков.

После этого модуль начинает контролировать сигналы операционной системы и при получении сигнала на завершение работы передает его основным модулям сервера. Тем самым особо важные модули сервера успевают, как правило, оставаться с персистентными данными.

# Binding — Пакет программ для описания и управления массивами привязки контроллеров.

В данном пакете приведены типы данных и функции обработки массивов привязки.

### **Package files**

[anytime.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/anytime.go) [arrays.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/arrays.go) [ctrlio.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/ctrlio.go) [days.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/days.go) [plans.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/plans.go) [setupDK.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/setupDK.go) [statistic.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go) [timeuse.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/timeuse.go) [weeks.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/weeks.go) [year.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/year.go)

## type **[Arrays](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/arrays.go?s=122:683" \l "L1)**

Arrays масиссивы привязок

type Arrays struct {

SetupDK [SetupDK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetupDK)

SetDK [SetDK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetDK)

MonthSets [MonthSets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "MonthSets)

WeekSets [WeekSets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "WeekSets)

DaySets [DaySets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "DaySets)

SetCtrl [SetCtrl](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetCtrl)

SetTimeUse [SetTimeUse](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetTimeUse)

TimeDivice [TimeDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "TimeDevice) `json:"timedev"` //Настройки времени

StatDefine [StatDefine](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "StatDefine) `json:"defstatis"` // Описание настройки сбора статистики

PointSet [PointSet](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "PointSet) `json:"pointset"` //Точки сбора статистики

UseInput [UseInput](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "UseInput) `json:"useinput"` //Назначение входов для сбора статистики

}

### **func [NewArrays](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/arrays.go?s=727:751" \l "L15)**

func NewArrays() \*[Arrays](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "Arrays)

NewArrays создание нового

### **func (\*Arrays) [IsCorrect](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/arrays.go?s=1144:1179" \l "L31)**

func (ar \*[Arrays](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "Arrays)) IsCorrect() [error](http://localhost:6060/pkg/builtin/" \l "error)

IsCorrect проверяет правильность массивов

### **func (\*Arrays) [SetArray](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/arrays.go?s=2451:2512" \l "L82)**

func (ar \*[Arrays](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "Arrays)) SetArray(nom, nelem [int](http://localhost:6060/pkg/builtin/" \l "int), array [][int](http://localhost:6060/pkg/builtin/" \l "int)) [error](http://localhost:6060/pkg/builtin/" \l "error)

SetArray принимает массивы привязки на устройтсве

## type **[CtrlStage](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/ctrlio.go?s=335:768" \l "L10)**

CtrlStage один интервал контроля

type CtrlStage struct {

Nline [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"line"` //Номер строки

Start [StageTime](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "StageTime) `json:"start"` //Время начала контроля

End [StageTime](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "StageTime) `json:"end"` //Время конца контроля

TVPLen [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"lenTVP"` //Длительность секунд контроля ТВП

MGRLen [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"lenMGR"` //Длительность секунд контроля МГР

}

### **func (\*CtrlStage) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/ctrlio.go?s=836:883" \l "L19)**

func (s \*[CtrlStage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "CtrlStage)) Compare(ss \*[CtrlStage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "CtrlStage)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

## type **[DaySets](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/days.go?s=103:162" \l "L1)**

DaySets все суточные планы

type DaySets struct {

DaySets []\*[OneDay](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "OneDay) `json:"daysets"`

}

### **func [NewDaySet](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/days.go?s=766:791" \l "L28)**

func NewDaySet() \*[DaySets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "DaySets)

NewDaySet создание нового набора суточных планов

### **func (\*DaySets) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/days.go?s=358:402" \l "L13)**

func (ds \*[DaySets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "DaySets)) Compare(dd \*[DaySets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "DaySets)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*DaySets) [FromBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/days.go?s=520:569" \l "L18)**

func (ds \*[DaySets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "DaySets)) FromBuffer(buffer [][int](http://localhost:6060/pkg/builtin/" \l "int)) [error](http://localhost:6060/pkg/builtin/" \l "error)

FromBuffer загружает суточные планы из буфера

### **func (\*DaySets) [String](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/days.go?s=164:198" \l "L4)**

func (ds \*[DaySets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "DaySets)) String() [string](http://localhost:6060/pkg/builtin/" \l "string)

## type **[Level](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=279:610" \l "L4)**

Level один из двух уровней сбора

type Level struct {

TypeSt [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"typst"` //Признак статистики

Period [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"period"` //Период усреднения в минутах

Ninput [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"ninput"` //Колличество входов сбора статистики

Count [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"count"` //Число направлений

}

## type **[Line](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/days.go?s=1425:1518" \l "L53)**

Line структура одного периода времени

type Line struct {

PKNom [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"npk"`

Hour [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"hour"`

Min [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"min"`

}

## type **[MonthSets](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/year.go?s=94:160" \l "L1)**

MonthSets все месячные планы

type MonthSets struct {

MonthSets []\*[OneMonth](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "OneMonth) `json:"monthset"`

}

### **func [NewYearSets](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/year.go?s=393:422" \l "L9)**

func NewYearSets() \*[MonthSets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "MonthSets)

NewYearSets создает новый набор месячных карт

### **func (\*MonthSets) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/year.go?s=228:276" \l "L4)**

func (ms \*[MonthSets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "MonthSets)) Compare(mm \*[MonthSets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "MonthSets)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*MonthSets) [FromBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/year.go?s=696:747" \l "L19)**

func (ms \*[MonthSets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "MonthSets)) FromBuffer(buffer [][int](http://localhost:6060/pkg/builtin/" \l "int)) [error](http://localhost:6060/pkg/builtin/" \l "error)

FromBuffer переводит из массива кодов в недельный массив

## type **[OneDay](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/days.go?s=993:1148" \l "L38)**

OneDay Один день плана

type OneDay struct {

Number [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"num"`

Count [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"count"` //Счетчик переключений

Lines []\*[Line](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "Line) `json:"lines"`

}

### **func (\*OneDay) [IsEmpty](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/days.go?s=1990:2022" \l "L81)**

func (od \*[OneDay](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "OneDay)) IsEmpty() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

IsEmpty возвращает истину если данный недельный массив пустой

### **func (\*OneDay) [String](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/days.go?s=1150:1183" \l "L44)**

func (od \*[OneDay](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "OneDay)) String() [string](http://localhost:6060/pkg/builtin/" \l "string)

### **func (\*OneDay) [ToBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/days.go?s=2208:2242" \l "L91)**

func (od \*[OneDay](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "OneDay)) ToBuffer() [][int](http://localhost:6060/pkg/builtin/" \l "int)

ToBuffer переводит из суточного массива в буфер кодов

## type **[OneMonth](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/year.go?s=931:1010" \l "L29)**

OneMonth Одна строка недельных планов

type OneMonth struct {

Number [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"num"`

Days [][int](http://localhost:6060/pkg/builtin/" \l "int) `json:"days"`

}

### **func (\*OneMonth) [IsEmpty](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/year.go?s=1313:1347" \l "L45)**

func (om \*[OneMonth](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "OneMonth)) IsEmpty() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

IsEmpty возвращает истину если данный недельный массив пустой

### **func (\*OneMonth) [ToBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/year.go?s=1528:1564" \l "L55)**

func (om \*[OneMonth](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "OneMonth)) ToBuffer() [][int](http://localhost:6060/pkg/builtin/" \l "int)

ToBuffer переводит из недельного массива в буфер кодов

## type **[OneWeek](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/weeks.go?s=897:975" \l "L29)**

OneWeek Одна строка недельных планов

type OneWeek struct {

Number [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"num"`

Days [][int](http://localhost:6060/pkg/builtin/" \l "int) `json:"days"`

}

### **func (\*OneWeek) [IsEmpty](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/weeks.go?s=1274:1307" \l "L45)**

func (nm \*[OneWeek](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "OneWeek)) IsEmpty() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

IsEmpty возвращает истину если данный недельный массив пустой

### **func (\*OneWeek) [ToBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/weeks.go?s=1488:1523" \l "L55)**

func (nm \*[OneWeek](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "OneWeek)) ToBuffer() [][int](http://localhost:6060/pkg/builtin/" \l "int)

ToBuffer переводит из недельного массива в буфер кодов

## type **[Point](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=1082:1223" \l "L28)**

Point Описание одной точки

type Point struct {

NumPoint [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"num"` //Номер точки

TypeSt [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"typst"` //Признак статистики

}

## type **[PointSet](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=917:1033" \l "L22)**

PointSet Точки сбора статистики

type PointSet struct {

Points [][Point](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "Point) `json:"pts"` //Описание точек сбора статистики

}

### **func [NewPointSet](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=2506:2534" \l "L81)**

func NewPointSet() \*[PointSet](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "PointSet)

NewPointSet создание новой

### **func (\*PointSet) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=2083:2129" \l "L64)**

func (ps \*[PointSet](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "PointSet)) Compare(ss \*[PointSet](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "PointSet)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*PointSet) [FromBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=4384:4434" \l "L171)**

func (ps \*[PointSet](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "PointSet)) FromBuffer(buffer [][int](http://localhost:6060/pkg/builtin/" \l "int)) [error](http://localhost:6060/pkg/builtin/" \l "error)

FromBuffer заполнить из буфера

### **func (\*PointSet) [IsEmpty](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=1296:1330" \l "L34)**

func (ps \*[PointSet](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "PointSet)) IsEmpty() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

IsEmpty вернет истину если массив пустой

### **func (\*PointSet) [ToBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=3101:3137" \l "L113)**

func (ps \*[PointSet](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "PointSet)) ToBuffer() [][int](http://localhost:6060/pkg/builtin/" \l "int)

ToBuffer сохранить в буфер

## type **[SetCtrl](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/ctrlio.go?s=100:142" \l "L1)**

SetCtrl массив контроля входов

type SetCtrl struct {

Stage [][CtrlStage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "CtrlStage)

}

### **func [NewSetCtrl](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/ctrlio.go?s=1002:1028" \l "L24)**

func NewSetCtrl() \*[SetCtrl](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetCtrl)

NewSetCtrl создает новый набор контроля входов

### **func (\*SetCtrl) [FromBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/ctrlio.go?s=1648:1697" \l "L56)**

func (sc \*[SetCtrl](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetCtrl)) FromBuffer(buffer [][int](http://localhost:6060/pkg/builtin/" \l "int)) [error](http://localhost:6060/pkg/builtin/" \l "error)

FromBuffer заполнить из буфера

### **func (\*SetCtrl) [IsEmpty](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/ctrlio.go?s=1473:1506" \l "L46)**

func (sc \*[SetCtrl](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetCtrl)) IsEmpty() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

IsEmpty вернет истину если весь набор пустой

### **func (\*SetCtrl) [ToBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/ctrlio.go?s=2430:2465" \l "L82)**

func (sc \*[SetCtrl](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetCtrl)) ToBuffer() [][int](http://localhost:6060/pkg/builtin/" \l "int)

ToBuffer выгружает набор в буфер

## type **[SetDK](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/plans.go?s=228:302" \l "L3)**

SetDK наборы планов координации для обеих ДК перекрестка

type SetDK struct {

DK [][SetPk](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetPk) `json:"dk"` // Наборы для ДК1

}

### **func [NewSetDK](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/plans.go?s=2225:2247" \l "L46)**

func NewSetDK() \*[SetDK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetDK)

NewSetDK создание нового набора планов координации

### **func (\*SetDK) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/plans.go?s=370:410" \l "L8)**

func (sd \*[SetDK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetDK)) Compare(ss \*[SetDK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetDK)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*SetDK) [FromBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/plans.go?s=5235:5282" \l "L207)**

func (sd \*[SetDK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetDK)) FromBuffer(buffer [][int](http://localhost:6060/pkg/builtin/" \l "int)) [error](http://localhost:6060/pkg/builtin/" \l "error)

FromBuffer создает план координации из буфера возвращает ошибку

### **func (\*SetDK) [IsEmpty](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/plans.go?s=2672:2713" \l "L68)**

func (sd \*[SetDK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetDK)) IsEmpty(dk, pk [int](http://localhost:6060/pkg/builtin/" \l "int)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

IsEmpty если план координации для данного ДК нулеыой то истина

## type **[SetPk](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/plans.go?s=528:1474" \l "L13)**

SetPk набор планов координации перекрестка

type SetPk struct {

DK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"dk"` //Номер ДК

Pk [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"pk"` //Номер программы от 1 до 12

TypePU [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"tpu"` //Тип программы управления управления 0-ЛПУ (локальная) 1-ПК(координации)

RazLen [bool](http://localhost:6060/pkg/builtin/" \l "bool) `json:"razlen"` //Признак наличия разнодлительных фаз

Tc [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"tc"` //Время цикла программы

Shift [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"shift"` //Сдвиг начала цикла

LastType [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"lasttype"` //Тип переходной фазы при сдвиге

LastNumber [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"lastnumber"` //Номер переходной фазы при сдвиге

TwoT [bool](http://localhost:6060/pkg/builtin/" \l "bool) `json:"twot"` //Признак 2Т

Stages [][Stage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "Stage) `json:"sts"` //Фазы переключения

}

### **func [NewSetPk](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/plans.go?s=2391:2418" \l "L56)**

func NewSetPk(pk [int](http://localhost:6060/pkg/builtin/" \l "int)) [SetPk](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetPk)

NewSetPk новый план

### **func (\*SetPk) [FromBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/plans.go?s=5554:5601" \l "L221)**

func (st \*[SetPk](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetPk)) FromBuffer(buffer [][int](http://localhost:6060/pkg/builtin/" \l "int)) [error](http://localhost:6060/pkg/builtin/" \l "error)

FromBuffer создает план координации из буфера возвращает ошибку

### **func (\*SetPk) [ToBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/plans.go?s=3095:3128" \l "L89)**

func (st \*[SetPk](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetPk)) ToBuffer() [][int](http://localhost:6060/pkg/builtin/" \l "int)

ToBuffer выгружает в буфер

## type **[SetTimeUse](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/timeuse.go?s=144:353" \l "L1)**

SetTimeUse хранение настроек внешних входов

type SetTimeUse struct {

Uses [][Use](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "Use) `json:"uses"`

IntervalTE [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"ite"` //Интервал между ТЕ

Tuin [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"tuin"` //Т уср ИН

MGRNotWork [][int](http://localhost:6060/pkg/builtin/" \l "int) `json:"notwork"`

}

### **func [NewSetTimeUse](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/timeuse.go?s=790:822" \l "L24)**

func NewSetTimeUse() \*[SetTimeUse](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetTimeUse)

NewSetTimeUse создание нового описания

### **func (\*SetTimeUse) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/timeuse.go?s=639:688" \l "L19)**

func (s \*[SetTimeUse](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetTimeUse)) Compare(ss \*[SetTimeUse](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetTimeUse)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*SetTimeUse) [FromBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/timeuse.go?s=1585:1636" \l "L62)**

func (s \*[SetTimeUse](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetTimeUse)) FromBuffer(buffer [][int](http://localhost:6060/pkg/builtin/" \l "int)) [error](http://localhost:6060/pkg/builtin/" \l "error)

FromBuffer загружает из буфера

### **func (\*SetTimeUse) [IsEmpty](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/timeuse.go?s=1351:1386" \l "L49)**

func (s \*[SetTimeUse](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetTimeUse)) IsEmpty() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

IsEmpty вернет истину если весь набор пустой

### **func (\*SetTimeUse) [ToBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/timeuse.go?s=3723:3767" \l "L165)**

func (s \*[SetTimeUse](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetTimeUse)) ToBuffer(num [int](http://localhost:6060/pkg/builtin/" \l "int)) [][int](http://localhost:6060/pkg/builtin/" \l "int)

ToBuffer выгружает в буффер num номер массива на сервере

## type **[SetupDK](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/setupDK.go?s=81:803" \l "L1)**

SetupDK настройка ДК

type SetupDK struct {

DKNum [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"dkn"` //Номер ДК

TMaxF [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"tmaxf"` //Максимальное время ожидания смены фаз

TMinF [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"tminf"` //Минимальное время ожидания смены фаз

TmaxTmin [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"tminmax"` // Максимальное время ожидания включения фазы

DKType [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"dktype"` //Тип ДК

ExtNum [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"extn"` //Внешний номер ДК

Tprom [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"tprom"` //Максимальное время промежуточного такта

IsPreset [bool](http://localhost:6060/pkg/builtin/" \l "bool) `json:"preset"` // Прищнак наличия контроллера на линии

}

### **func [NewSetupDK](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/setupDK.go?s=1033:1059" \l "L16)**

func NewSetupDK() \*[SetupDK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetupDK)

NewSetupDK создает новый набор настройки ДК

### **func (\*SetupDK) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/setupDK.go?s=871:917" \l "L11)**

func (sdk \*[SetupDK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetupDK)) Compare(tdk \*[SetupDK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetupDK)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*SetupDK) [FromBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/setupDK.go?s=1180:1230" \l "L22)**

func (sdk \*[SetupDK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetupDK)) FromBuffer(buffer [][int](http://localhost:6060/pkg/builtin/" \l "int)) [error](http://localhost:6060/pkg/builtin/" \l "error)

FromBuffer переводит из массива кодов в структуру1

### **func (\*SetupDK) [IsEmpty](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/setupDK.go?s=2154:2188" \l "L54)**

func (sdk \*[SetupDK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetupDK)) IsEmpty() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

IsEmpty возвращает истину если данный массив пустой

### **func (\*SetupDK) [ToBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/setupDK.go?s=2263:2299" \l "L59)**

func (sdk \*[SetupDK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "SetupDK)) ToBuffer() [][int](http://localhost:6060/pkg/builtin/" \l "int)

ToBuffer переводит в буфер кодов

## type **[Stage](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/plans.go?s=1555:2133" \l "L27)**

Stage описание одной фазы плана координации

type Stage struct {

Nline [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"line"` //Номер строки

Start [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"start"` //Время начала фазы

Number [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"num"` //Номер фазы

Tf [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"tf"` //Тип фазы 0 -простая

// 1 - МГР

// 2 - 1ТВП

// 3 - 2ТВП

// 4 - 1,2ТВП

// 5 - Зам 1 ТВП

// 6 - Зам 2 ТВП

// 7 - Зам

// 8 - МДК

// 9 - ВДК

Len [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"len"` //Длительность секунд

Plus [bool](http://localhost:6060/pkg/builtin/" \l "bool) `json:"plus"` //Признак переноса времени на следующую фазу

}

## type **[StageTime](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/ctrlio.go?s=202:278" \l "L4)**

StageTime время в массиве контроля

type StageTime struct {

Hour [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"hour"`

Minute [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"min"`

}

## type **[StatDefine](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=111:220" \l "L1)**

StatDefine настройка сбора статистики

type StatDefine struct {

Levels [][Level](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "Level) `json:"lvs"` //Два уровня сбора статистики

}

### **func [NewStatDefine](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=2365:2397" \l "L74)**

func NewStatDefine() \*[StatDefine](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "StatDefine)

NewStatDefine создание новой

### **func (\*StatDefine) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=1927:1977" \l "L59)**

func (sd \*[StatDefine](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "StatDefine)) Compare(ss \*[StatDefine](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "StatDefine)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*StatDefine) [FromBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=3768:3820" \l "L149)**

func (sd \*[StatDefine](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "StatDefine)) FromBuffer(buffer [][int](http://localhost:6060/pkg/builtin/" \l "int)) [error](http://localhost:6060/pkg/builtin/" \l "error)

FromBuffer заполнить из буфера

### **func (\*StatDefine) [IsEmpty](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=683:719" \l "L12)**

func (sd \*[StatDefine](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "StatDefine)) IsEmpty() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

IsEmpty вернет истину если массив пустой

### **func (\*StatDefine) [ToBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=2775:2813" \l "L95)**

func (sd \*[StatDefine](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "StatDefine)) ToBuffer() [][int](http://localhost:6060/pkg/builtin/" \l "int)

ToBuffer сохранить в буфер

## type **[TimeDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/anytime.go?s=113:303" \l "L1)**

TimeDevice Описание времени устройства

type TimeDevice struct {

TimeZone [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"tz"` //Смещение от Гринвича

Summer [bool](http://localhost:6060/pkg/builtin/" \l "bool) `json:"summer"` // Есть ди переход на летнее время

}

### **func [NewTimeDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/anytime.go?s=556:588" \l "L10)**

func NewTimeDevice() \*[TimeDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "TimeDevice)

NewTimeDevice создает новое описание времени устройства

### **func (\*TimeDevice) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/anytime.go?s=371:421" \l "L5)**

func (td \*[TimeDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "TimeDevice)) Compare(tt \*[TimeDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "TimeDevice)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*TimeDevice) [FromBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/anytime.go?s=727:779" \l "L17)**

func (td \*[TimeDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "TimeDevice)) FromBuffer(buffer [][int](http://localhost:6060/pkg/builtin/" \l "int)) [error](http://localhost:6060/pkg/builtin/" \l "error)

FromBuffer переводит из массива кодов в структуру

### **func (\*TimeDevice) [IsEmpty](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/anytime.go?s=1306:1342" \l "L36)**

func (td \*[TimeDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "TimeDevice)) IsEmpty() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

IsEmpty возвращает истину если данный массив пустой

### **func (\*TimeDevice) [ToBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/anytime.go?s=1481:1519" \l "L44)**

func (td \*[TimeDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "TimeDevice)) ToBuffer() [][int](http://localhost:6060/pkg/builtin/" \l "int)

ToBuffer переводит из структуры в буфер кодов

## type **[Use](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/timeuse.go?s=379:571" \l "L9)**

Use один вход

type Use struct {

Name [string](http://localhost:6060/pkg/builtin/" \l "string) `json:"name"`

Type [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"type"`

Tvps [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"tvps"`

Dk [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"dk"`

Fazes [string](http://localhost:6060/pkg/builtin/" \l "string) `json:"fazes"`

Long [float32](http://localhost:6060/pkg/builtin/" \l "float32) `json:"long"`

}

## type **[UseInput](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=1527:1674" \l "L44)**

UseInput назначение входов для сбора статистики

type UseInput struct {

Used [][bool](http://localhost:6060/pkg/builtin/" \l "bool) `json:"used"` //Признаки использования входа для сбора статистики

}

### **func [NewUseInput](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=2641:2669" \l "L88)**

func NewUseInput() \*[UseInput](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "UseInput)

NewUseInput создание новой

### **func (\*UseInput) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=2235:2281" \l "L69)**

func (ui \*[UseInput](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "UseInput)) Compare(ss \*[UseInput](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "UseInput)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*UseInput) [FromBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=5018:5068" \l "L193)**

func (ui \*[UseInput](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "UseInput)) FromBuffer(buffer [][int](http://localhost:6060/pkg/builtin/" \l "int)) [error](http://localhost:6060/pkg/builtin/" \l "error)

FromBuffer заполнить из буфера

### **func (\*UseInput) [IsEmpty](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=1747:1781" \l "L49)**

func (ui \*[UseInput](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "UseInput)) IsEmpty() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

IsEmpty вернет истину если массив пустой

### **func (\*UseInput) [ToBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/statistic.go?s=3444:3480" \l "L130)**

func (ui \*[UseInput](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "UseInput)) ToBuffer() [][int](http://localhost:6060/pkg/builtin/" \l "int)

ToBuffer сохранить в буфер

## type **[WeekSets](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/weeks.go?s=95:154" \l "L1)**

WeekSets все недельные планы

type WeekSets struct {

WeekSets [][OneWeek](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "OneWeek) `json:"wsets"`

}

### **func [NewWeekSets](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/weeks.go?s=635:663" \l "L19)**

func NewWeekSets() \*[WeekSets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "WeekSets)

NewNedelSets создает новый набор недельных карт

### **func (\*WeekSets) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/weeks.go?s=222:268" \l "L4)**

func (ns \*[WeekSets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "WeekSets)) Compare(nn \*[WeekSets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "WeekSets)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*WeekSets) [FromBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/binding/weeks.go?s=390:440" \l "L9)**

func (ns \*[WeekSets](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "WeekSets)) FromBuffer(buffer [][int](http://localhost:6060/pkg/builtin/" \l "int)) [error](http://localhost:6060/pkg/builtin/" \l "error)

FromBuffer загружает недельный массив из буфера

# Comm — Пакет программ собственно для связи с контроллерами АСУД.

### **Package files**

[comarray.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/comm/comarray.go) [comm-serv.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/comm/comm-serv.go) [data-comm.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/comm/data-comm.go)

## func **[GetChanArray](http://localhost:6060/src/github.com/ruraomsk/ag-server/comm/comm-serv.go?s=652:703" \l "L16)**

func GetChanArray(id [int](http://localhost:6060/pkg/builtin/" \l "int)) (chan [CommandArray](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/comm/" \l "CommandArray), [bool](http://localhost:6060/pkg/builtin/" \l "bool))

GetChanArray возвращает канал для присылки массивов для данного устройства

## func **[StartListen](http://localhost:6060/src/github.com/ruraomsk/ag-server/comm/comm-serv.go?s=868:929" \l "L25)**

func StartListen(stop chan [int](http://localhost:6060/pkg/builtin/" \l "int), rq chan [int](http://localhost:6060/pkg/builtin/" \l "int), ans chan [string](http://localhost:6060/pkg/builtin/" \l "string))

StartListen основной вход сервер коммуникаций

## type **[CommandARM](http://localhost:6060/src/github.com/ruraomsk/ag-server/comm/data-comm.go?s=121:226" \l "L1)**

CommandARM Команды от Сервера АРМ

type CommandARM struct {

ID [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"id"`

Command [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"cmd"`

Params [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"param"`

}

## type **[CommandArray](http://localhost:6060/src/github.com/ruraomsk/ag-server/comm/data-comm.go?s=287:429" \l "L5)**

CommandArray Привязка от Сервера АРМ

type CommandArray struct {

ID [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"id"`

Number [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"number"`

NElem [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"nelem"`

Elems [][int](http://localhost:6060/pkg/builtin/" \l "int) `json:"elems"`

}

# Extcon — Пакет программ управления состоянием модулей сервера.

### **Package files**

[extcon.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/extcon/extcon.go)

## func **[BackgroundInit](http://localhost:6060/src/github.com/ruraomsk/ag-server/extcon/extcon.go?s=1101:1122" \l "L45)**

func BackgroundInit()

BackgroundInit инициализируем

## func **[BackgroundWork](http://localhost:6060/src/github.com/ruraomsk/ag-server/extcon/extcon.go?s=1689:1743" \l "L68)**

func BackgroundWork(step [time](http://localhost:6060/pkg/time/).[Duration](http://localhost:6060/pkg/time/" \l "Duration), stop chan [int](http://localhost:6060/pkg/builtin/" \l "int))

BackgroundWork обычно вызывается для обслуживания разного ПОсле выхода нет контекстов

## func **[SetTimerClock](http://localhost:6060/src/github.com/ruraomsk/ag-server/extcon/extcon.go?s=1450:1501" \l "L62)**

func SetTimerClock(step [time](http://localhost:6060/pkg/time/).[Duration](http://localhost:6060/pkg/time/" \l "Duration)) \*[time](http://localhost:6060/pkg/time/).[Ticker](http://localhost:6060/pkg/time/" \l "Ticker)

SetTimerClock порождает канал где приходят по времения сообщения

## type **[ExtContext](http://localhost:6060/src/github.com/ruraomsk/ag-server/extcon/extcon.go?s=134:238" \l "L3)**

ExtContext расширенный контекст

type ExtContext struct {

// contains filtered or unexported fields

}

### **func [NewContext](http://localhost:6060/src/github.com/ruraomsk/ag-server/extcon/extcon.go?s=536:585" \l "L19)**

func NewContext(name [string](http://localhost:6060/pkg/builtin/" \l "string)) (\*[ExtContext](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/extcon/" \l "ExtContext), [error](http://localhost:6060/pkg/builtin/" \l "error))

NewContext создает новый расширенный контекст только с командой завершения

### **func (\*ExtContext) [Done](http://localhost:6060/src/github.com/ruraomsk/ag-server/extcon/extcon.go?s=983:1027" \l "L40)**

func (ec \*[ExtContext](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/extcon/" \l "ExtContext)) Done() <-chan struct{}

Done return chan for cancel

### **func (\*ExtContext) [GetName](http://localhost:6060/src/github.com/ruraomsk/ag-server/extcon/extcon.go?s=893:931" \l "L35) [¶](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/extcon/" \l "ExtContext.GetName)**

func (ec \*[ExtContext](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/extcon/" \l "ExtContext)) GetName() [string](http://localhost:6060/pkg/builtin/" \l "string)

GetName return name context

# Inspect — Пакет программ поддержания соответствия состояния контроллера заданными на перекрестке.

### **Package files**

[inspect.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/inspect/inspect.go)

## func **[Start](http://localhost:6060/src/github.com/ruraomsk/ag-server/inspect/inspect.go?s=558:611" \l "L8)**

func Start(context \*[extcon](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/extcon/).[ExtContext](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/extcon/" \l "ExtContext), stop chan [int](http://localhost:6060/pkg/builtin/" \l "int))

Start главный модуль инспектора

# Pudge — Пакет программ сопровождения базы данных системы.

### **Package files**

[api-controller.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/api-controller.go) [api-pudge.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/api-pudge.go) [data-pudge.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go) [dev-api.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/dev-api.go) [gui-pudge.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/gui-pudge.go) [loadsave.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/loadsave.go)

## Variables

Works флаг готовности pudge

var Works [bool](http://localhost:6060/pkg/builtin/" \l "bool)

## func **[DeleteCross](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/api-pudge.go?s=1232:1270" \l "L54)**

func DeleteCross(region, area, id [int](http://localhost:6060/pkg/builtin/" \l "int))

DeleteCross Удаляет перекресток

## func **[SetController](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/api-pudge.go?s=3125:3158" \l "L120)**

func SetController(c \*[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller))

SetController Записывает новое состояние контроллера и если есть изменения то записывает его в лог

## func **[SetCross](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/api-pudge.go?s=2253:2276" \l "L91)**

func SetCross(c \*[Cross](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Cross))

SetCross обновляет состояние перекрестка

## func **[SetCrossNewDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/api-pudge.go?s=1889:1942" \l "L77)**

func SetCrossNewDevice(reg [Region](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Region), idevice [int](http://localhost:6060/pkg/builtin/" \l "int)) [error](http://localhost:6060/pkg/builtin/" \l "error)

SetCrossNewDevice деляет новую привязку контроллера

## func **[SetDefault](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=12675:12717" \l "L330)**

func SetDefault(c \*[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller), key [string](http://localhost:6060/pkg/builtin/" \l "string))

SetDefault Заполнить по умолчанию

## func **[Start](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/api-pudge.go?s=3787:3870" \l "L145)**

func Start(context \*[extcon](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/extcon/).[ExtContext](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/extcon/" \l "ExtContext), stop chan [int](http://localhost:6060/pkg/builtin/" \l "int), rq chan [int](http://localhost:6060/pkg/builtin/" \l "int), ans chan [string](http://localhost:6060/pkg/builtin/" \l "string))

Start главная процедура управления состоянием котроллеров

## type **[ArrayPriv](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=10466:10529" \l "L283)**

ArrayPriv собственно массив привязки

type ArrayPriv struct {

Number [int](http://localhost:6060/pkg/builtin/" \l "int)

NElem [int](http://localhost:6060/pkg/builtin/" \l "int)

Array [][int](http://localhost:6060/pkg/builtin/" \l "int)

}

### **func (\*ArrayPriv) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=10597:10644" \l "L290)**

func (a \*[ArrayPriv](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "ArrayPriv)) Compare(aa \*[ArrayPriv](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "ArrayPriv)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

## type **[Controller](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=10765:12463" \l "L295)**

Controller внутренне представление контроллера

type Controller struct {

ID [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"id"` // Уникальный номер контроллера

Name [string](http://localhost:6060/pkg/builtin/" \l "string) `json:"name"` //Имя перекрестка если привязан

StatusConnection [StatusConnection](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "StatusConnection) `json:"scon"` // Статус соединения

LastOperation [time](http://localhost:6060/pkg/time/).[Time](http://localhost:6060/pkg/time/" \l "Time) `json:"ltime"` // Время последней операции обмена с устройством

WriteToDB [bool](http://localhost:6060/pkg/builtin/" \l "bool) `json:"-"` //Если истина то еще не записана в БД

TexRezim [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"rezim"` //Технологический режим

Base [bool](http://localhost:6060/pkg/builtin/" \l "bool) `json:"base"` //Если истина то работает по базовой привязке

PK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"pk"` //Номер плана координации

CK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"ck"` //Номер суточной карты

NK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"nk"` //Номер недельной карты

StatusCommandDU [StatusCommandDU](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "StatusCommandDU)

DK [DK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "DK)

TMax [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"tmax"` //Максимальное время ожидания ответа от сервера в секундах

Model [Model](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Model)

Error [ErrorDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "ErrorDevice)

GPS [GPS](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "GPS)

Input [Input](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Input)

Statistics [][Statistic](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Statistic)

Arrays [][ArrayPriv](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "ArrayPriv) `json:"arrays"` //Файлы привязки

LogLines [][LogLine](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "LogLine)

}

### **func [GetController](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/api-pudge.go?s=1677:1723" \l "L69)**

func GetController(id [int](http://localhost:6060/pkg/builtin/" \l "int)) (\*[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller), [bool](http://localhost:6060/pkg/builtin/" \l "bool))

GetController возвращает копию Контроллера

### **func (\*Controller) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=12531:12580" \l "L325)**

func (c \*[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller)) Compare(cc \*[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*Controller) [IsConnected](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/api-controller.go?s=115:154" \l "L1)**

func (c \*[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller)) IsConnected() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

IsConnected возвращает на связи ли устройство

### **func (\*Controller) [ToList](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/gui-pudge.go?s=5210:5248" \l "L111)**

func (c \*[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller)) ToList() [][string](http://localhost:6060/pkg/builtin/" \l "string)

ToList для вывода в гуи

## type **[Controllers](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=1882:1930" \l "L29)**

Controllers возврат выбранных контроллеров

type Controllers struct {

Contrs [][Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller)

}

## type **[Cross](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=8960:10401" \l "L261)**

Cross описание перекрестка

type Cross struct {

Region [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"region"` //Регион

Area [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"area"` //Район

SubArea [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"subarea"` //подрайон

ID [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"id"` //Номер перекрестка

IDevice [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"idevice"` // Назначенное на перекресток устройство

Dgis [string](http://localhost:6060/pkg/builtin/" \l "string) `json:"dgis"` //Координаты перекрестка

ConType [string](http://localhost:6060/pkg/builtin/" \l "string) `json:"contype"` //Тип соединения устройства

NumDev [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"numdev"` //Номер устройства (УСДК,ДК-А,С12УСДК)

Name [string](http://localhost:6060/pkg/builtin/" \l "string) `json:"name"`

Phone [string](http://localhost:6060/pkg/builtin/" \l "string) `json:"phone"` //Телефон

StatusDevice [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"status"` // Статус устройства

WriteToDB [bool](http://localhost:6060/pkg/builtin/" \l "bool) `json:"-"` //Если истина то еще не записана в БД

PK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"pk"` //Номер плана координации

CK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"ck"` //Номер суточной карты

NK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"nk"` //Номер недельной карты

Statistics [][Statistic](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Statistic) `json:"statis"` //Накопленная статистика

Arrays [binding](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/).[Arrays](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/binding/" \l "Arrays) `json:"arrays"` //Файлы привязки

}

### **func [GetCross](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/api-pudge.go?s=528:577" \l "L21)**

func GetCross(region, area, id [int](http://localhost:6060/pkg/builtin/" \l "int)) ([Cross](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Cross), [bool](http://localhost:6060/pkg/builtin/" \l "bool))

GetCross возвращает копию перекрестка

### **func [NewCross](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=13216:13238" \l "L355)**

func NewCross() \*[Cross](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Cross)

NewCross создание нового описания перекрестка

## type **[DK](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=2199:3261" \l "L46)**

DK диагностика состояния по ДК

type DK struct {

RDK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"rdk"` //Режим ДК

FDK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"fdk"` //Фаза ДК

// от 1 до 8 номера рабочих фаз

// 9 промежуточный такт

// 10 желтое мигание

// 11 отключен светофор

// 12 кругом краснный

DDK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"ddk"` //Устройство ДК

EDK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"edk"` //Неисправность ДК

PDK [bool](http://localhost:6060/pkg/builtin/" \l "bool) `json:"pdk"` //Признак переходного периода ДК

EEDK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"eedk"` //дополнительный код неисправности

ODK [bool](http://localhost:6060/pkg/builtin/" \l "bool) `json:"odk"` //Открыта дверь ДК

LDK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"ldk"` //Номер фазы на которой сгорели лампы

FTUDK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"ftudk"` //Фаза ТУ ДК на момент передачи

TDK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"tdk"` //Время отработки ТУ в секундах

FTSDK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"ftsdk"` //Фаза ТС ДК

TTCDK [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"ttcdk"` //Время от начала фазы ТС в секундах

}

### **func (\*DK) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=3329:3362" \l "L68)**

func (d \*[DK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "DK)) Compare(dd \*[DK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "DK)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*DK) [Make](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/dev-api.go?s=1652:1693" \l "L84)**

func (d \*[DK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "DK)) Make(buffer [][byte](http://localhost:6060/pkg/builtin/" \l "byte), pos [int](http://localhost:6060/pkg/builtin/" \l "int))

Make устанавлиявает поля

### **func (\*DK) [Set](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/dev-api.go?s=1049:1089" \l "L53)**

func (d \*[DK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "DK)) Set(buffer [][byte](http://localhost:6060/pkg/builtin/" \l "byte), pos [int](http://localhost:6060/pkg/builtin/" \l "int))

Set устанавлиявает поля

### **func (\*DK) [ToList](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/gui-pudge.go?s=72:117" \l "L1)**

func (d \*[DK](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "DK)) ToList(result [][string](http://localhost:6060/pkg/builtin/" \l "string)) [][string](http://localhost:6060/pkg/builtin/" \l "string)

ToList для вывода в гуи

## type **[DataStat](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=7608:7806" \l "L224)**

DataStat статистика по канально

type DataStat struct {

Chanel [int](http://localhost:6060/pkg/builtin/" \l "int) //Номер канала

Status [int](http://localhost:6060/pkg/builtin/" \l "int) // Состояние 0-исправен 1-обрыв 2 - замыкание

Intensiv [int](http://localhost:6060/pkg/builtin/" \l "int) //Интенсивность

}

## type **[ErrorDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=4010:4404" \l "L88)**

ErrorDevice описание ошибок устройства

type ErrorDevice struct {

V220DK1 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Срабатывание входа контроля 220В DK1

V220DK2 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Срабатывание входа контроля 220В DK2

RTC [bool](http://localhost:6060/pkg/builtin/" \l "bool) // Неисправность часов RTC

TVP1 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Неисправность ТВП1

TVP2 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Неисправность ТВП2

FRAM [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Неисправность FRAM

}

### **func (\*ErrorDevice) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=4925:4976" \l "L126)**

func (e \*[ErrorDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "ErrorDevice)) Compare(ee \*[ErrorDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "ErrorDevice)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*ErrorDevice) [MakeError](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=4561:4599" \l "L105)**

func (e \*[ErrorDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "ErrorDevice)) MakeError() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

MakeError случайным образом создает ошибку

### **func (\*ErrorDevice) [ToList](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/gui-pudge.go?s=2120:2174" \l "L42)**

func (e \*[ErrorDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "ErrorDevice)) ToList(result [][string](http://localhost:6060/pkg/builtin/" \l "string)) [][string](http://localhost:6060/pkg/builtin/" \l "string)

ToList для вывода в гуи

## type **[GPS](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=5095:5402" \l "L131)**

GPS описание состояния модуля GPS устройства

type GPS struct {

Ok [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Исправно

E01 [bool](http://localhost:6060/pkg/builtin/" \l "bool) // Нет связи с приемником

E02 [bool](http://localhost:6060/pkg/builtin/" \l "bool) // Ошибка CRC

E03 [bool](http://localhost:6060/pkg/builtin/" \l "bool) // Нет валидного времени

E04 [bool](http://localhost:6060/pkg/builtin/" \l "bool) // Мало спутников

Seek [bool](http://localhost:6060/pkg/builtin/" \l "bool) // Поиск спутников после включения

}

### **func (\*GPS) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=5821:5856" \l "L162)**

func (g \*[GPS](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "GPS)) Compare(gg \*[GPS](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "GPS)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*GPS) [MakeError](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=5476:5506" \l "L141)**

func (g \*[GPS](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "GPS)) MakeError() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

MakeError порождает ошибки или испавность

### **func (\*GPS) [ToList](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/gui-pudge.go?s=2900:2946" \l "L59)**

func (g \*[GPS](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "GPS)) ToList(result [][string](http://localhost:6060/pkg/builtin/" \l "string)) [][string](http://localhost:6060/pkg/builtin/" \l "string)

ToList для вывода в гуи

## type **[Input](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=5973:6402" \l "L167)**

Input описание состояния входов устройства

type Input struct {

V1 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Неисправность входа 1

V2 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Неисправность входа 2

V3 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Неисправность входа 3

V4 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Неисправность входа 4

V5 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Неисправность входа 5

V6 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Неисправность входа 6

V7 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Неисправность входа 7

V8 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Неисправность входа 8

}

### **func (\*Input) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=6875:6914" \l "L204)**

func (i \*[Input](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Input)) Compare(ii \*[Input](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Input)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*Input) [MakeError](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=6476:6508" \l "L179)**

func (i \*[Input](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Input)) MakeError() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

MakeError порождает ошибки или испавность

### **func (\*Input) [ToList](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/gui-pudge.go?s=3568:3616" \l "L75)**

func (i \*[Input](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Input)) ToList(result [][string](http://localhost:6060/pkg/builtin/" \l "string)) [][string](http://localhost:6060/pkg/builtin/" \l "string)

ToList для вывода в гуи

## type **[LogLine](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=8601:8666" \l "L247)**

LogLine запись лога устройства

type LogLine struct {

Time [time](http://localhost:6060/pkg/time/).[Time](http://localhost:6060/pkg/time/" \l "Time)

Record [int](http://localhost:6060/pkg/builtin/" \l "int)

Info [int](http://localhost:6060/pkg/builtin/" \l "int)

}

### **func (\*LogLine) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=8734:8777" \l "L254)**

func (l \*[LogLine](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "LogLine)) Compare(ll \*[LogLine](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "LogLine)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

## type **[Model](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=3460:3799" \l "L73)**

Model Описание модели устройства

type Model struct {

VPCPD [int](http://localhost:6060/pkg/builtin/" \l "int) //Версия ПО платы ПСПД

VPBS [int](http://localhost:6060/pkg/builtin/" \l "int) //Версия ПО платы ПБС

C12 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Субблок С12

STP [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Разрешение накопление статистики по ТП

DKA [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Контроллер ДК-А

DTA [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Детектор транспорта

}

### **func (\*Model) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=3867:3906" \l "L83)**

func (m \*[Model](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Model)) Compare(mm \*[Model](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Model)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*Model) [ToList](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/gui-pudge.go?s=1402:1450" \l "L25)**

func (m \*[Model](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Model)) ToList(result [][string](http://localhost:6060/pkg/builtin/" \l "string)) [][string](http://localhost:6060/pkg/builtin/" \l "string)

ToList для вывода в гуи

## type **[Region](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=1492:1632" \l "L17)**

Region указатель на номер перекрестка

type Region struct {

Region [int](http://localhost:6060/pkg/builtin/" \l "int) //Код региона

Area [int](http://localhost:6060/pkg/builtin/" \l "int) //Код района

ID [int](http://localhost:6060/pkg/builtin/" \l "int) //Номер перекрестка

}

### **func [GetCrosses](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/api-pudge.go?s=829:855" \l "L34)**

func GetCrosses() [][Region](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Region)

GetCrosses возвращает все перекрестки

### **func (\*Region) [ToKey](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=1685:1716" \l "L24)**

func (r \*[Region](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Region)) ToKey() [string](http://localhost:6060/pkg/builtin/" \l "string)

ToKey создает строковый ключ

## type **[Statistic](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=6986:7400" \l "L209)**

Statistic статистика

type Statistic struct {

Period [int](http://localhost:6060/pkg/builtin/" \l "int) //Номер периода усреднения от начала суток

Type [int](http://localhost:6060/pkg/builtin/" \l "int) //Тип статистики 1-интенсивность скорость

TLen [int](http://localhost:6060/pkg/builtin/" \l "int) //Величина времения усреднения мин

Hour [int](http://localhost:6060/pkg/builtin/" \l "int) //Час окончания периода

Min [int](http://localhost:6060/pkg/builtin/" \l "int) //Минуты окончания периода

Datas [][DataStat](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "DataStat)

}

### **func (\*Statistic) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=7468:7515" \l "L219)**

func (s \*[Statistic](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Statistic)) Compare(ss \*[Statistic](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Statistic)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

## type **[StatusCommandDU](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=7846:8382" \l "L231)**

StatusCommandDU команды ДУ

type StatusCommandDU struct {

IsPK [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Назначен ПК

IsPKS [bool](http://localhost:6060/pkg/builtin/" \l "bool) // назначена карта выбора по времени суток

IsNK [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Назначена недельная карта

IsDUDK1 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //на 1 ДК есть команда ДУ

IsDUDK2 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //на 2 ДК есть команда ДУ

IsReqSFDK1 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Есть запрос на передачу фаз по 1 ДК СФДК

IsReqSFDK2 [bool](http://localhost:6060/pkg/builtin/" \l "bool) //Есть запрос на передачу фаз по 2 ДК СФДК

}

### **func (\*StatusCommandDU) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=8450:8509" \l "L242)**

func (s \*[StatusCommandDU](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "StatusCommandDU)) Compare(ss \*[StatusCommandDU](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "StatusCommandDU)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнивание истина если равны

### **func (\*StatusCommandDU) [Make](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/dev-api.go?s=103:151" \l "L1)**

func (s \*[StatusCommandDU](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "StatusCommandDU)) Make() (command [uint8](http://localhost:6060/pkg/builtin/" \l "uint8))

Make создает байт для передачи статуса команд ДУ

### **func (\*StatusCommandDU) [Set](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/dev-api.go?s=522:566" \l "L21)**

func (s \*[StatusCommandDU](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "StatusCommandDU)) Set(command [uint8](http://localhost:6060/pkg/builtin/" \l "uint8))

Set переносит из байта статуса команд ДУ в поле состояния

### **func (\*StatusCommandDU) [ToList](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/gui-pudge.go?s=4501:4559" \l "L96)**

func (s \*[StatusCommandDU](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "StatusCommandDU)) ToList(result [][string](http://localhost:6060/pkg/builtin/" \l "string)) [][string](http://localhost:6060/pkg/builtin/" \l "string)

ToList для вывода в гуи

## type **[StatusConnection](http://localhost:6060/src/github.com/ruraomsk/ag-server/pudge/data-pudge.go?s=1985:2010" \l "L34)**

StatusConnection статус соединения

type StatusConnection [int](http://localhost:6060/pkg/builtin/" \l "int)

const (

//Connected Ok

Connected [StatusConnection](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "StatusConnection) = [iota](http://localhost:6060/pkg/builtin/" \l "iota)

//NotConnected not Ok

NotConnected

//Undefine Undefine

Undefine

)

# Setup — Пакет программ приема и настройки сервера.

### **Package files**

[api-setup.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/setup/api-setup.go) [data-setup.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/setup/data-setup.go)

## Variables

Setuppath путь к файлу

var Setuppath [string](http://localhost:6060/pkg/builtin/" \l "string)

## func **[FromJSON](http://localhost:6060/src/github.com/ruraomsk/ag-server/setup/api-setup.go?s=1341:1375" \l "L49)**

func FromJSON(result [][byte](http://localhost:6060/pkg/builtin/" \l "byte)) [error](http://localhost:6060/pkg/builtin/" \l "error)

FromJSON приниает json

## func **[LoadSetUp](http://localhost:6060/src/github.com/ruraomsk/ag-server/setup/api-setup.go?s=325:358" \l "L8)**

func LoadSetUp(path [string](http://localhost:6060/pkg/builtin/" \l "string)) [error](http://localhost:6060/pkg/builtin/" \l "error)

LoadSetUp загружает настройки из json

## func **[ToJSON](http://localhost:6060/src/github.com/ruraomsk/ag-server/setup/api-setup.go?s=1208:1248" \l "L43)**

func ToJSON() (result [][byte](http://localhost:6060/pkg/builtin/" \l "byte), err [error](http://localhost:6060/pkg/builtin/" \l "error))

ToJSON переводит в Json

## func **[WriteSetUp](http://localhost:6060/src/github.com/ruraomsk/ag-server/setup/api-setup.go?s=724:747" \l "L24)**

func WriteSetUp() [error](http://localhost:6060/pkg/builtin/" \l "error)

WriteSetUp записывает в json настройки

## type **[CommServer](http://localhost:6060/src/github.com/ruraomsk/ag-server/setup/data-setup.go?s=406:1219" \l "L6)**

CommServer настройки для сервера коммуникации

type CommServer struct {

Port [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"port"` //Стартовый номер порта на прием

PortCommand [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"portc"` //Порт приема команд от сервера АРМ

PortArray [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"porta"` //Порт приема массивов привязки от сервера АРМ

TimeOutRead [time](http://localhost:6060/pkg/time/).[Duration](http://localhost:6060/pkg/time/" \l "Duration) `json:"read\_timeout"` //Таймаут на чтение если данные должны быть получены

TimeOutWrite [time](http://localhost:6060/pkg/time/).[Duration](http://localhost:6060/pkg/time/" \l "Duration) `json:"write\_timeout"` //Таймаут на запись если данные должны быть переданы

KeepAlive [time](http://localhost:6060/pkg/time/).[Duration](http://localhost:6060/pkg/time/" \l "Duration) `json:"time\_keep\_alive"` //Интервал времени в течении которого должен прийти keepalive от устройства

}

## type **[Controller](http://localhost:6060/src/github.com/ruraomsk/ag-server/setup/data-setup.go?s=2688:3182" \l "L42)**

Controller настройки имитатора

type Controller struct {

IP [string](http://localhost:6060/pkg/builtin/" \l "string) `json:"ip"`

GuiPort [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"guiport"`

Port [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"port"`

Step [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"step"` //Интервал времени в секундах для расчетов

KeepAlive [time](http://localhost:6060/pkg/time/).[Duration](http://localhost:6060/pkg/time/" \l "Duration) `json:"time\_keep\_alive"` //Интервал времени в течении которого должен прийти keepalive от устройства

Random [bool](http://localhost:6060/pkg/builtin/" \l "bool) `json:"random"`

}

## type **[DataBase](http://localhost:6060/src/github.com/ruraomsk/ag-server/setup/data-setup.go?s=2450:2635" \l "L33)**

DataBase настройки базы данных postresql

type DataBase struct {

Host [string](http://localhost:6060/pkg/builtin/" \l "string) `json:"host"`

Port [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"port"`

User [string](http://localhost:6060/pkg/builtin/" \l "string) `json:"user"`

Password [string](http://localhost:6060/pkg/builtin/" \l "string) `json:"password"`

DBname [string](http://localhost:6060/pkg/builtin/" \l "string) `json:"dbname"`

}

## type **[Pudge](http://localhost:6060/src/github.com/ruraomsk/ag-server/setup/data-setup.go?s=2010:2386" \l "L26)**

Pudge настройки подсистемы хранения состояние контроллеров

type Pudge struct {

StepSave [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"step"` //Интервал времени в секундах для сохранения состояния контроллеров

TableCross [string](http://localhost:6060/pkg/builtin/" \l "string) `json:"cross"` //Имя таблицы перекрестков

TableSave [string](http://localhost:6060/pkg/builtin/" \l "string) `json:"save"` //Имя таблицы где храним текущее состояние

}

## type **[Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/setup/data-setup.go?s=1282:1899" \l "L17)**

Server настройки для сервера армов

type Server struct {

Port [int](http://localhost:6060/pkg/builtin/" \l "int) `json:"port"` //Стартовый номер порта на прием

TimeOutRead [time](http://localhost:6060/pkg/time/).[Duration](http://localhost:6060/pkg/time/" \l "Duration) `json:"read\_timeout"` //Таймаут на чтение если данные должны быть получены

TimeOutWrite [time](http://localhost:6060/pkg/time/).[Duration](http://localhost:6060/pkg/time/" \l "Duration) `json:"write\_timeout"` //Таймаут на запись если данные должны быть переданы

KeepAlive [time](http://localhost:6060/pkg/time/).[Duration](http://localhost:6060/pkg/time/" \l "Duration) `json:"time\_keep\_alive"` //Интервал времени в течении которого должен прийти keepalive от устройства

}

## type **[Setup](http://localhost:6060/src/github.com/ruraomsk/ag-server/setup/data-setup.go?s=118:325" \l "L1)**

Setup общая структура для настройки всей системы

type Setup struct {

DataBase [DataBase](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/setup/" \l "DataBase)

Server [Server](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/setup/" \l "Server)

Pudge [Pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/setup/" \l "Pudge)

CommServer [CommServer](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/setup/" \l "CommServer)

Controller [Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/setup/" \l "Controller)

Location [string](http://localhost:6060/pkg/builtin/" \l "string) `json:"location"` //Локация временной зоны

}

Set переменная для хранения текущих настроек

var Set [Setup](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/setup/" \l "Setup)

# Test — Локальные тесты системы.

# Transport — Пакет программ транспортного протокола поверх TCP/IP для контроллеров АСУД.

### **Package files**

[message.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/message.go) [net-transport.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/net-transport.go) [subdevice.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go) [subserver.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go) [transport.go](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/transport.go)

## func **[GetMessagesFromDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/net-transport.go?s=184:265" \l "L2)**

func GetMessagesFromDevice(socket [net](http://localhost:6060/pkg/net/).[Conn](http://localhost:6060/pkg/net/" \l "Conn), hcan chan [HeaderDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderDevice), status \*[bool](http://localhost:6060/pkg/builtin/" \l "bool))

GetMessagesFromDevice принять сообщение

## func **[GetMessagesFromService](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/net-transport.go?s=1926:1994" \l "L52)**

func GetMessagesFromService(socket [net](http://localhost:6060/pkg/net/).[Conn](http://localhost:6060/pkg/net/" \l "Conn), hcan chan [HeaderServer](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderServer))

GetMessagesFromService прием сообщений от сервера

## func **[SendMessagesToDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/net-transport.go?s=3404:3484" \l "L90)**

func SendMessagesToDevice(socket [net](http://localhost:6060/pkg/net/).[Conn](http://localhost:6060/pkg/net/" \l "Conn), hout chan [HeaderServer](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderServer), status \*[bool](http://localhost:6060/pkg/builtin/" \l "bool))

SendMessagesToDevice передать сообщение на устройство

## func **[SendMessagesToServer](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/net-transport.go?s=4255:4321" \l "L116)**

func SendMessagesToServer(socket [net](http://localhost:6060/pkg/net/).[Conn](http://localhost:6060/pkg/net/" \l "Conn), hout chan [HeaderDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderDevice))

SendMessagesToServer передать сообщение на устройство

## type **[HeaderDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/transport.go?s=130:508" \l "L3)**

HeaderDevice Сообщение от устройства

type HeaderDevice struct {

ID [int](http://localhost:6060/pkg/builtin/" \l "int) //ID

TypeDevice [uint8](http://localhost:6060/pkg/builtin/" \l "uint8) //Тип устройства

Time [time](http://localhost:6060/pkg/time/).[Time](http://localhost:6060/pkg/time/" \l "Time) // Время сообщения

Number [uint8](http://localhost:6060/pkg/builtin/" \l "uint8) // Номер сообщения

Code [uint8](http://localhost:6060/pkg/builtin/" \l "uint8) // Код отправителя

Message [][uint8](http://localhost:6060/pkg/builtin/" \l "uint8) // Собственно сообщение без контрольной суммы

}

### **func [CreateHeaderDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/transport.go?s=586:648" \l "L13)**

func CreateHeaderDevice(id, tp, number, code [int](http://localhost:6060/pkg/builtin/" \l "int)) [HeaderDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderDevice)

CreateHeaderDevice создает заголовок устройства

### **func (\*HeaderDevice) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/transport.go?s=849:902" \l "L25)**

func (d \*[HeaderDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderDevice)) Compare(dd \*[HeaderDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderDevice)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare Сравнение

### **func (\*HeaderDevice) [MakeBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/transport.go?s=3217:3259" \l "L112)**

func (d \*[HeaderDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderDevice)) MakeBuffer() [][byte](http://localhost:6060/pkg/builtin/" \l "byte)

MakeBuffer создает буфер для передачи полностью упакованный со всеми КС

### **func (\*HeaderDevice) [Parse](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/transport.go?s=1929:1978" \l "L61)**

func (d \*[HeaderDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderDevice)) Parse(buffer [][byte](http://localhost:6060/pkg/builtin/" \l "byte)) [error](http://localhost:6060/pkg/builtin/" \l "error)

Parse разбор сообщения от устройства

### **func (\*HeaderDevice) [ParseMessage](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/message.go?s=2613:2663" \l "L123)**

func (d \*[HeaderDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderDevice)) ParseMessage() [][SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)

ParseMessage разбор буфера сообщений от устройства

### **func (\*HeaderDevice) [UpackMessages](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/message.go?s=2925:2986" \l "L135)**

func (d \*[HeaderDevice](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderDevice)) UpackMessages(subs [][SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) [error](http://localhost:6060/pkg/builtin/" \l "error)

UpackMessages записывает сообщения устройства для передачи

## type **[HeaderServer](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/transport.go?s=995:1336" \l "L30)**

HeaderServer Сообщение от сервера

type HeaderServer struct {

IDServer [int](http://localhost:6060/pkg/builtin/" \l "int) //ID Сервера 0xa7 0x8D

Time [time](http://localhost:6060/pkg/time/).[Time](http://localhost:6060/pkg/time/" \l "Time) // Время сообщения

Number [uint8](http://localhost:6060/pkg/builtin/" \l "uint8) // Номер сообщения

Code [uint8](http://localhost:6060/pkg/builtin/" \l "uint8) // Код отправителя

Message [][uint8](http://localhost:6060/pkg/builtin/" \l "uint8) // Собственно сообщение без контрольной суммы

}

### **func [CreateHeaderServer](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/transport.go?s=1408:1459" \l "L39)**

func CreateHeaderServer(num, code [int](http://localhost:6060/pkg/builtin/" \l "int)) [HeaderServer](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderServer)

CreateHeaderServer создает заголовок сервера

### **func (\*HeaderServer) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/transport.go?s=1771:1824" \l "L56)**

func (s \*[HeaderServer](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderServer)) Compare(ss \*[HeaderServer](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderServer)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare Сравнение

### **func (\*HeaderServer) [MakeBuffer](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/transport.go?s=4026:4068" \l "L138)**

func (s \*[HeaderServer](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderServer)) MakeBuffer() [][byte](http://localhost:6060/pkg/builtin/" \l "byte)

MakeBuffer создает буфер для передачи полностью упакованный со всеми КС

### **func (\*HeaderServer) [Parse](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/transport.go?s=2754:2803" \l "L95)**

func (s \*[HeaderServer](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderServer)) Parse(buffer [][byte](http://localhost:6060/pkg/builtin/" \l "byte)) [error](http://localhost:6060/pkg/builtin/" \l "error)

Parse разбор сообщения от сервера

### **func (\*HeaderServer) [ParseMessage](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/message.go?s=645:695" \l "L21)**

func (s \*[HeaderServer](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderServer)) ParseMessage() [][SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)

ParseMessage разбор буфера сообщений от сервера

### **func (\*HeaderServer) [UpackMessages](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/message.go?s=1103:1164" \l "L40)**

func (s \*[HeaderServer](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "HeaderServer)) UpackMessages(subs [][SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) [error](http://localhost:6060/pkg/builtin/" \l "error)

UpackMessages записывает сообщения сервера для передачи

## type **[SubMessage](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/message.go?s=135:193" \l "L1)**

SubMessage структура для хранения сообщений

type SubMessage struct {

Type [uint8](http://localhost:6060/pkg/builtin/" \l "uint8)

Message [][uint8](http://localhost:6060/pkg/builtin/" \l "uint8)

}

### **func (\*SubMessage) [Compare](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/message.go?s=477:526" \l "L16)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Compare(ss \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Compare сравнение сообщений

### **func (\*SubMessage) [Get0x01Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=822:890" \l "L22)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x01Device() (num, min, sec, emas, elem [int](http://localhost:6060/pkg/builtin/" \l "int))

Get0x01Device читает субсообщение

### **func (\*SubMessage) [Get0x01Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=468:508" \l "L9)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x01Server() [int](http://localhost:6060/pkg/builtin/" \l "int)

Get0x01Server получение параметров команды -1 ошибки

### **func (\*SubMessage) [Get0x02Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=782:823" \l "L21)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x02Server() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Get0x02Server получение параметров команды -1 ошибки

### **func (\*SubMessage) [Get0x04Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=1617:1681" \l "L48)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x04Device() (hour, min, day, month [int](http://localhost:6060/pkg/builtin/" \l "int))

Get0x04Device записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Get0x04Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=1104:1148" \l "L34)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x04Server() [2][bool](http://localhost:6060/pkg/builtin/" \l "bool)

Get0x04Server получение параметров команды -1 ошибки

### **func (\*SubMessage) [Get0x05Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=1508:1548" \l "L49)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x05Server() [int](http://localhost:6060/pkg/builtin/" \l "int)

Get0x05Server получение параметров команды -1 ошибки

### **func (\*SubMessage) [Get0x06Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=1822:1862" \l "L61)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x06Server() [int](http://localhost:6060/pkg/builtin/" \l "int)

Get0x06Server получение параметров команды -1 ошибки

### **func (\*SubMessage) [Get0x07Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=2350:2414" \l "L72)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x07Device() (hour, min, day, month [int](http://localhost:6060/pkg/builtin/" \l "int))

Get0x07Device записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Get0x07Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=2136:2176" \l "L73)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x07Server() [int](http://localhost:6060/pkg/builtin/" \l "int)

Get0x07Server получение параметров команды -1 ошибки

### **func (\*SubMessage) [Get0x09Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=8478:8546" \l "L305)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x09Device() (st [pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[Statistic](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Statistic), err [error](http://localhost:6060/pkg/builtin/" \l "error))

Get0x09Device записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Get0x09Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=2450:2490" \l "L85)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x09Server() [int](http://localhost:6060/pkg/builtin/" \l "int)

Get0x09Server получение параметров команды -1 ошибки

### **func (\*SubMessage) [Get0x0ADevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=9573:9634" \l "L341)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x0ADevice(st \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[Statistic](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Statistic)) [error](http://localhost:6060/pkg/builtin/" \l "error)

Get0x0ADevice сообщение статистики

### **func (\*SubMessage) [Get0x0AServer](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=2764:2804" \l "L97)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x0AServer() [int](http://localhost:6060/pkg/builtin/" \l "int)

Get0x0AServer получение параметров команды -1 ошибки

### **func (\*SubMessage) [Get0x0BDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=10617:10676" \l "L379)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x0BDevice(lg \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[LogLine](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "LogLine)) [error](http://localhost:6060/pkg/builtin/" \l "error)

Get0x0BDevice сообщение статистики

### **func (\*SubMessage) [Get0x0BServer](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=3078:3121" \l "L109)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x0BServer() [2][int](http://localhost:6060/pkg/builtin/" \l "int)

Get0x0BServer получение параметров команды -1 ошибки

### **func (\*SubMessage) [Get0x0FDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=2665:2726" \l "L84)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x0FDevice(c \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller)) [error](http://localhost:6060/pkg/builtin/" \l "error)

Get0x0FDevice изменяет состояние контроллера по команду

### **func (\*SubMessage) [Get0x10Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=4765:4826" \l "L156)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x10Device(c \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller)) [error](http://localhost:6060/pkg/builtin/" \l "error)

Get0x10Device изменяет состояние контроллера по команду

### **func (\*SubMessage) [Get0x11Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=5915:5976" \l "L194)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x11Device(c \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller)) [error](http://localhost:6060/pkg/builtin/" \l "error)

Get0x11Device изменяет состояние контроллера по команду

### **func (\*SubMessage) [Get0x12Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=4203:4264" \l "L137)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x12Device(c \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller)) [error](http://localhost:6060/pkg/builtin/" \l "error)

Get0x12Device изменяет состояние контроллера по команду

### **func (\*SubMessage) [Get0x13Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=11334:11395" \l "L405)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x13Device(ar \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[ArrayPriv](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "ArrayPriv)) [error](http://localhost:6060/pkg/builtin/" \l "error)

Get0x13Device сообщение массива привязки

### **func (\*SubMessage) [Get0x1CDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=12060:12106" \l "L431)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x1CDevice() (num [int](http://localhost:6060/pkg/builtin/" \l "int))

Get0x1CDevice читает субсообщение

### **func (\*SubMessage) [Get0x1DDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=12319:12380" \l "L441)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x1DDevice(c \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller)) [error](http://localhost:6060/pkg/builtin/" \l "error)

Get0x1DDevice изменяет состояние контроллера по команду

### **func (\*SubMessage) [Get0x32Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=3462:3531" \l "L124)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x32Server() (ip [string](http://localhost:6060/pkg/builtin/" \l "string), port [int](http://localhost:6060/pkg/builtin/" \l "int), err [error](http://localhost:6060/pkg/builtin/" \l "error))

Get0x32Server получение параметров команды -1 ошибки

### **func (\*SubMessage) [Get0x33Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=4119:4159" \l "L146)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x33Server() [int](http://localhost:6060/pkg/builtin/" \l "int)

Get0x33Server получение параметров команды -1 ошибки

### **func (\*SubMessage) [Get0x34Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=4460:4501" \l "L158)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x34Server() [bool](http://localhost:6060/pkg/builtin/" \l "bool)

Get0x34Server получение параметров команды -1 ошибки

### **func (\*SubMessage) [Get0x35Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=4782:4830" \l "L171)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Get0x35Server() ([int](http://localhost:6060/pkg/builtin/" \l "int), [bool](http://localhost:6060/pkg/builtin/" \l "bool))

Get0x35Server получение параметров команды -1 ошибки

### **func (\*SubMessage) [GetArray](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=5146:5195" \l "L184)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) GetArray() ([int](http://localhost:6060/pkg/builtin/" \l "int), [int](http://localhost:6060/pkg/builtin/" \l "int), [][int](http://localhost:6060/pkg/builtin/" \l "int))

GetArray возвращает номер и массив от сервера

### **func (\*SubMessage) [GetCodeCommandServer](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=202:249" \l "L1)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) GetCodeCommandServer() [int](http://localhost:6060/pkg/builtin/" \l "int)

GetCodeCommandServer возвращает номер команды или ноль если это массив привязки

### **func (\*SubMessage) [Set0x00Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=202:238" \l "L1)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x00Device()

Set0x00Device записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x01Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=460:525" \l "L9)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x01Device(num, min, sec, emas, elem [int](http://localhost:6060/pkg/builtin/" \l "int))

Set0x01Device записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x01Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=5598:5641" \l "L198)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x01Server(num [int](http://localhost:6060/pkg/builtin/" \l "int))

Set0x01Server записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x02Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=5890:5935" \l "L207)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x02Server(flag [bool](http://localhost:6060/pkg/builtin/" \l "bool))

Set0x02Server записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x03Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=6229:6265" \l "L219)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x03Server()

Set0x03Server записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x04Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=1223:1284" \l "L36)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x04Device(hour, min, day, month [int](http://localhost:6060/pkg/builtin/" \l "int))

Set0x04Device записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x04Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=6487:6536" \l "L227)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x04Server(dk1, dk2 [bool](http://localhost:6060/pkg/builtin/" \l "bool))

Set0x04Server записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x05Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=6842:6885" \l "L244)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x05Server(num [int](http://localhost:6060/pkg/builtin/" \l "int))

Set0x05Server записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x06Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=7149:7192" \l "L253)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x06Server(num [int](http://localhost:6060/pkg/builtin/" \l "int))

Set0x06Server записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x07Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=1953:2014" \l "L60)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x07Device(hour, min, day, month [int](http://localhost:6060/pkg/builtin/" \l "int))

Set0x07Device записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x07Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=7464:7507" \l "L262)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x07Server(num [int](http://localhost:6060/pkg/builtin/" \l "int))

Set0x07Server записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x09Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=8042:8097" \l "L292)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x09Device(st \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[Statistic](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Statistic))

Set0x09Device записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x09Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=7767:7810" \l "L271)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x09Server(num [int](http://localhost:6060/pkg/builtin/" \l "int))

Set0x09Server записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x0ADevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=8917:8978" \l "L319)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x0ADevice(st \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[Statistic](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Statistic)) [error](http://localhost:6060/pkg/builtin/" \l "error)

Set0x0ADevice сообщение статистики

### **func (\*SubMessage) [Set0x0AServer](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=8064:8107" \l "L280)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x0AServer(num [int](http://localhost:6060/pkg/builtin/" \l "int))

Set0x0AServer записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x0BDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=10293:10346" \l "L367)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x0BDevice(lg \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[LogLine](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "LogLine))

Set0x0BDevice сообщение статистики

### **func (\*SubMessage) [Set0x0BServer](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=8361:8405" \l "L289)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x0BServer(m, n [int](http://localhost:6060/pkg/builtin/" \l "int))

Set0x0BServer записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x0FDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=3221:3282" \l "L102)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x0FDevice(c \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller)) [error](http://localhost:6060/pkg/builtin/" \l "error)

Set0x0FDevice изменяет состояние сообщения по состоянию контроллера

### **func (\*SubMessage) [Set0x10Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=5323:5378" \l "L170)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x10Device(c \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller))

Set0x10Device изменяет состояние контроллера по команду

### **func (\*SubMessage) [Set0x11Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=6947:7002" \l "L222)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x11Device(c \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller))

Set0x11Device изменяет состояние контроллера по команду

### **func (\*SubMessage) [Set0x12Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=3708:3769" \l "L119)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x12Device(c \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller)) [error](http://localhost:6060/pkg/builtin/" \l "error)

Set0x12Device изменяет состояние сообщения по состоянию контроллера

### **func (\*SubMessage) [Set0x13Device](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=10975:11030" \l "L391)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x13Device(ar \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[ArrayPriv](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "ArrayPriv))

Set0x13Device сообщение массива привязки

### **func (\*SubMessage) [Set0x1CDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=11827:11870" \l "L422)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x1CDevice(num [int](http://localhost:6060/pkg/builtin/" \l "int))

Set0x1CDevice записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x1DDevice](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subdevice.go?s=13375:13430" \l "L469)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x1DDevice(c \*[pudge](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/).[Controller](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/pudge/" \l "Controller))

Set0x1DDevice изменяет состояние контроллера по команду

### **func (\*SubMessage) [Set0x32Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=8641:8702" \l "L299)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x32Server(ip [string](http://localhost:6060/pkg/builtin/" \l "string), port [int](http://localhost:6060/pkg/builtin/" \l "int)) [error](http://localhost:6060/pkg/builtin/" \l "error)

Set0x32Server записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x33Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=9360:9403" \l "L331)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x33Server(min [int](http://localhost:6060/pkg/builtin/" \l "int))

Set0x33Server записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x34Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=9663:9707" \l "L340)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x34Server(rez [bool](http://localhost:6060/pkg/builtin/" \l "bool))

Set0x34Server записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [Set0x35Server](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=9942:10002" \l "L351)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) Set0x35Server(interval [int](http://localhost:6060/pkg/builtin/" \l "int), ignor [bool](http://localhost:6060/pkg/builtin/" \l "bool))

Set0x35Server записывает субсообщение для команды с номером в имени

### **func (\*SubMessage) [SetArray](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/subserver.go?s=10220:10282" \l "L362)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) SetArray(num [int](http://localhost:6060/pkg/builtin/" \l "int), nelem [int](http://localhost:6060/pkg/builtin/" \l "int), array [][int](http://localhost:6060/pkg/builtin/" \l "int))

SetArray возвращает номер и массив от сервера

### **func (\*SubMessage) [ToString](http://localhost:6060/src/github.com/ruraomsk/ag-server/transport/message.go?s=233:271" \l "L6)**

func (s \*[SubMessage](http://localhost:6060/pkg/github.com/ruraomsk/ag-server/transport/" \l "SubMessage)) ToString() [string](http://localhost:6060/pkg/builtin/" \l "string)

ToString вывод в строку