



Геликс Беспроводные Системы

Программируемый логический контроллер и
регистратор данных

Геликс-1

GPS/ГЛОНАСС + I/O версия

Телематический программируемый логический контроллер и
регистратор данных

Геликс-2

GPS/ГЛОНАСС + GSM/GPRS + I/O версия

Руководство Администратора

Версия 2.40

Листов 183

Информация представленная в этом документе конфиденциальна, является собственностью компании «Геликс Беспроводные Системы» (далее «Геликс») и не может быть раскрыта/передана иным лицам и компаниям без согласия компании «Геликс» за исключением сотрудников, задействованных в проекте. При этом раскрывающая Сторона несет ответственность за соблюдение такими третьими лицами и сотрудниками положений о неразглашении. Любое использование материалов этого документа (перепечатка, цитирование, копирование, дублирование, сканирование, фотографирование и проч.) без письменного согласия компании «Геликс» попадает под ответственность, оговоренную Федеральными законами РФ.

1 ноября 2011 г.

Оглавление

1. ВВЕДЕНИЕ	10
2. ТЕРМИНЫ.....	11
3. ОБЪЕКТНО-ОРИЕНТИРОВАННАЯ МОДЕЛЬ.....	16
3.1. ОБЪЕКТ «СИСТЕМА».....	17
3.2. ОБЪЕКТ «GPS»	19
3.3. ОБЪЕКТ «ВЫХОД».....	21
3.4. ОБЪЕКТ «ЗОНА ОХРАНЫ»	21
3.5. ОБЪЕКТ «АНАЛОГОВЫЙ ВХОД»	22
3.6. ОБЪЕКТ «ГЕО-ЗОНА»	23
3.7. ОБЪЕКТ «ПОЛЬЗОВАТЕЛЬ»	24
3.8. ОБЪЕКТ «СООБЩЕНИЕ»	24
3.9. ОБЪЕКТ «GSM/GPRS-МОДУЛЬ»	26
3.10. ОБЪЕКТ «ТАЙМЕР»	29
3.11. ОБЪЕКТ «УСЛОВИЕ»	31
3.12. ОБЪЕКТ «ЖУРНАЛ СОБЫТИЙ»	32
3.13. ОБЪЕКТ «МАКРО КОМАНДА»	34
3.13.1. Правила построения макро команды	35
3.13.3. Методы оптимизации сценария управления	37
3.14. ОБЪЕКТ «КЛЮЧЕВОЕ СЛОВО»	37
3.15. ОБЪЕКТ «СЧЕТЧИК»	38
3.16. ОБЪЕКТ «КАМЕРА»	38
3.17. ОБЪЕКТ «ВРЕМЕННАЯ ЗОНА».....	40
3.18. ОБЪЕКТ «ПОЛЬЗОВАТЕЛЬСКИЕ ФЛАГИ».....	40
3.19. ТАБЛИЦА ЗНАЧЕНИЙ КОНСТАНТ	41
3.19.1. Объекты.....	41
3.19.2. Условие	47
4. &-КОМАНДЫ	53
4.1. ПРАВИЛА ПОСТРОЕНИЯ &-КОМАНД	53
4.2. ОБЩИЕ КОМАНДЫ.....	55
4.2.1. Выдать информацию о системе.....	55
4.2.2. AT-команды.....	56
4.2.3. &set.[<psw>].id=[<system_id>][,<config_id>]	57
4.2.4. &get.[<psw>].id.....	57
4.2.5. &get.[<psw>].version[=all option]	58
4.2.6. &set.[<psw>].psw="<new_psw>"	58
4.2.7. &set.[<psw>].sys=[<gm>],[<sp>],[<es>],[<nr>],[<pr>],[<dl>],[<mm>],[<sz>],[<ts>],[<dz>],[<bc>]	58
4.2.8. &set.[<psw>].lock={Scenario All Unlock}	61
4.2.9. &set.[<psw>].save	61
4.2.10. &get.[<psw>].system[=cfg]	62
4.2.11. &restart.[<psw>]	63
4.3. РАБОТА С ДИСКРЕТНЫМИ ВХОДАМИ	64
4.3.1. &set.[<psw>].zone=<mask>,<mode>,[<arm>][,<NC>].....	64
4.3.2. &get.[<psw>].zone.....	65
4.4. РАБОТА С ДИСКРЕТНЫМИ ВЫХОДАМИ.....	66
4.4.1. &set.[<psw>].relay=<status>[,<mask>][,<type>]	66

4.4.2. &get.[<psw>].relay	67
4.5. РАБОТА С ДАТОЙ И ВРЕМЕНЕМ	67
4.5.1. &get.[<psw>].date	67
4.5.2. &set.[<psw>].date=<DDMMYY>,<HHMMSS>	68
4.6. РАБОТА С ЖУРНАЛОМ СОБЫТИЙ	68
4.6.1. &req.[<psw>].confirm[=<N>]	70
4.6.2. &req.[<psw>].first=[{f c w n}][,x]	71
4.6.3. &req.[<psw>].last=[<N>][,{h x}][,reverse]	71
4.6.4. &req.[<psw>].new=[{? <N>},{h x}][,reverse]	73
4.6.5. &req.[<psw>].pos[=x]	73
4.6.6. &req.[<psw>].range=<Start>,[<N>][,{h x}][,reverse]	74
4.6.7. &req.[<psw>].time=[<ddmmy>],[<hhmmss>],[<N>][,hexmode]	75
4.7. НАСТРОЙКА РАБОТЫ ПОРТА TE	76
4.7.1. &set.[<psw>].baud=<baud_index>[,Rightnow]	76
4.7.2. &set.[<psw>].echo=[<period>],[<filter>][,<mode>]	76
4.8. УПРАВЛЕНИЕ РЕЖИМОМ АВТОМАТИЧЕСКОЙ ЗАПИСИ ДАННЫХ В LOG	78
4.8.1. &set.[<psw>].log=[{t1}],[{F}],[{sp}],[{nS}],[{af}],[{df}],[{ns}],[{se}],[{ct}],[{lm}],[{pp}],[{t2}]	78
4.8.2. &add.[<psw>].log={Byte Word Double None}[,<n_reg>]	81
4.8.3. &get.[<psw>].log=[Full][,Hex]	81
4.9. РАБОТА С ОБЪЕКТОМ ТАЙМЕР	82
4.9.1. &add.[<psw>].timer=<num>	82
4.9.2. &get.[<psw>].timer[=?]	83
4.9.3. &set.[<psw>].timer=<id>,<mode>,<delay>,<Nrepeat>	83
4.9.4. &del.[<psw>].timer[=<id>]	84
4.10. РАБОТА С ОБЪЕКТОМ «УСЛОВИЕ»	84
4.10.1. &add.[<psw>].condition=<num>	85
4.10.2. &set.[<psw>].condition=<id>,<obj>,<obj_id>,<status>[,not]	85
4.10.3. &get.[<psw>].condition	85
4.10.4. &del.[<psw>].condition[=<id>]	86
4.11. РАБОТА С ОБЪЕКТОМ МАКРО	86
4.11.1. &add.[<psw>].macro=<num>	87
4.11.2. &get.[<psw>].macro=[<begin>[,<number>]]	87
4.11.3. &set.[<psw>].macro={id},{en},{st},{sid},{se},{dt},{did},{de}[,<condition>[,not]]	88
4.11.4. &set.[<psw>].macro=<id>,run[,any]	88
4.11.5. &del.[<psw>].macro[=<id>]	89
4.12. РАБОТА С ОБЪЕКТОМ ПОЛЬЗОВАТЕЛЬ	89
4.12.1. &add.[<psw>].user={type},{mode},{address},{port}[,ResetFirst]	90
4.12.2. &get.[<psw>].user[={current <user_id>}]	93
4.12.3. &del.[<psw>].user	93
4.13. РАБОТА С ОБЪЕКТОМ СООБЩЕНИЕ	93
4.13.1. &add.[<psw>].message=<message>	97
4.13.2. &get.[<psw>].message[=?]	97
4.13.3. &del.[<psw>].message	97
4.14. КОМАНДЫ РАБОТЫ С КЛЮЧЕВЫМИ СЛОВАМИ	98
4.14.1. &add.[<psw>].key=<keyword>	100
4.14.2. &del.[<psw>].key	100
4.15. КОМАНДЫ ТЕСТИРОВАНИЯ И ОТЛАДКИ СИСТЕМЫ	100
4.15.1. &erase.[<psw>].cfg	100
4.15.2. &erase.[<psw>].log	101
4.15.3. &restart.[<psw>]	101
4.15.5. &test.[<psw>].watchdog	102
4.16. НАСТРОЙКА РАБОТЫ УСТРОЙСТВА С НАВИГАЦИОННЫМ ПРИЁМНИКОМ	103

4.16.1. &set.[<psw>].nmea=[<mask >],[<baud_index>],[<pwrMode>],[<hdopMax>]	103
4.16.2. &get.[<psw>].nmea	104
4.17. РАБОТА С GSM	105
4.17.1. &get.[<psw>].gsm=[imei]	106
4.17.2. &set.[<psw>].gsm=["(pin)"],[mode],[ussd0],[ussd1],[fs],[<copsId>]	107
4.17.3. Настройка точки входа в GPRS-сеть	109
4.17.4. &get.[<psw>].temperature={B G I}	111
4.18. РАБОТА С ГЕО-ЗОНАМИ	112
4.18.1. &add.[<psw>].geo=<N>	112
4.18.2. &get.[<psw>].geo	113
4.18.3. &set.[<psw>].geo={id},{st},{lat},{long},{diff},{C R}	113
4.18.4. &del.[<psw>].geo	114
4.19. РАБОТА С АНАЛОГОВЫМИ ВХОДАМИ	114
4.19.1. &get.[<psw>].analog={? id}	115
4.19.2. &set.[<psw>].analog={id},{min},{max}	115
4.20. РАБОТА С ПРОГРАММНЫМИ СЧЁТЧИКАМИ	116
4.20.1. &set.[<psw>].cnt=<num>	116
4.20.2. &set.[<psw>].cnt={mask},{enable},{increment}	117
4.20.3. &get.[<psw>].cnt	118
4.21. РАБОТА С ВРЕМЕННЫМИ ЗОНАМИ (ОПЦИЯ)	118
4.21.1. &set.[<psw>].tzone={tzone_id},{enabled},{tz_beg},{tz_end}	119
4.21.2. &get.[<psw>].tzone	119
4.22. РАБОТА С ГЕО-ОДОМЕТРОМ	119
4.22.1. &set.[<psw>].odometer={value} disable enable	120
4.23. РАБОТА ИНТЕРФЕЙСОМ RS485	121
4.23.1. &set.[<psw>].cmd={baud_index},{msg_id},{fractal_mode}	121
4.23.2. &get.[<psw>].cmd	122
4.24. РАБОТА С ПОЛЬЗОВАТЕЛЬСКИМИ ФЛАГАМИ	122
4.24. РЕЖИМ ПРОГРАММИРОВАНИЯ УСТРОЙСТВА	123
4.24.1. &set.[<psw>].programmode={1 0}	123
5. ПРИМЕРЫ ПРОГРАММИРОВАНИЯ	124
5.1. УПРАВЛЕНИЕ РЕЖИМОМ РЕГИСТРАЦИИ ПО ВНЕШНЕМУ СИГНАЛУ	124
5.1.2. Задача	124
5.1.3. Макро программа	124
5.1.4. Пример программирования	124
5.2. ЗАПИСЬ ГЕО-ТОЧКИ В ЛОГ ПО ВНЕШНЕМУ СИГНАЛУ	124
5.2.1. Возможные области применения	124
5.2.2. Задача	124
5.2.3. Макро программа	124
5.2.4. Пример программирования	125
5.3. ИЗМЕНЕНИЕ СКОРОСТИ ЗАПИСИ ИСТОРИИ ПО ВНЕШНЕМУ СИГНАЛУ	125
5.3.2. Задача	125
5.3.3. Макро программа	125
5.3.4. Пример программирования	125
5.4. КРАТКОВРЕМЕННОЕ ВКЛЮЧЕНИЕ РЕЛЕ ПО ВНЕШНЕМУ СИГНАЛУ	125
5.4.1. Задача	125
5.4.2. Макро программа	126
5.4.3. Пример программирования	126
5.5. ПЕРИОДИЧЕСКОЕ ВКЛЮЧЕНИЕ/ВЫКЛЮЧЕНИЕ РЕЛЕ	126
5.5.1. Возможные области применения	126
5.5.2. Задача	126

5.5.3. Исходные объекты	126
5.5.4. Макро программа	126
5.5.5. Пример программирования	126
5.6. LED ТЕСТ	128
5.6.1. Задача	128
5.6.2. Программирование	128
5.6.3. Описание	128
5.7. LED ТЕСТ ПО СИГНАЛУ «ТРЕВОГА»	129
5.7.1. Задача	129
5.7.2. Программирование	129
5.8. LED ТЕСТ ПРИ ВЫПОЛНЕНИИ «УСЛОВИЯ»	130
5.8.1. Задача	130
5.8.2. Программирование	131
5.9. Послать SMS-сообщение по внешнему сигналу	131
5.9.1. Задача	131
5.9.2. Программирование	131
5.9.3. Пример диалога в режиме «отладка»	132
5.10. Передача навигационных данных по GPRS-каналу	132
5.10.1. Задача	132
5.10.2. Программирование	133
5.10.3. Отладка	134
5.11. Регистрация данных расхода топлива	135
5.11.1. Задача	135
5.11.2. Программирование	135
5.11.3. Отладка	136
5.12. Связь по BLUETOOTH-каналу	136
5.12.1. Задача	137
5.12.2. Программирование	137
5.12.3. Отладка	138
5.13. Аварийное соединение с резервным сервером	139
5.13.1. Задача	139
5.13.2. Программирование	139
5.14. Учёт времени работы силового агрегата (двигателя)	140
5.14.1. Задача	140
5.14.2. Решение	140
5.15. Звуковое подтверждение авторизированной карты доступа	141
5.15.1. Задача	141
5.15.2. Решение	141
5.16. Работа с датчиком расхода топлива LLS-12	142
5.16.1. Задача	142
5.16.2. Проверка режима работы ДУЖ-12	142
5.16.3. Решение 1	142
5.16.4. Решение 2	143
5.17. Изменение периода записи ЛОГ в зависимости от скорости	144
5.17.1. Задача	144
5.17.2. Решение	145
5.18. Передача фото изображения на IP-сервер	145
5.18.1. Задача	145
5.18.2. Решение	145
5.18.3. Варианты модернизации	146
5.19. Работа с резервным(и) IP-сервером	147
5.19.1. Задача	147

5.19.2. Решение.....	147
5.20. ГОЛОСОВОЙ ВЫЗОВ ПРИ АКТИВНОМ СОЕДИНЕНИИ С IP-СЕРВЕРОМ	147
5.20.1. Задача.....	147
5.20.2. Решение.....	148
6. ПРОБЛЕМЫ И РЕШЕНИЯ, ВОПРОСЫ И ОТВЕТЫ.....	150
7. ТЕХНИЧЕСКАЯ ПОДДЕРЖКА	158
 ПРИЛОЖЕНИЕ 1. МОДУЛЬ РАЗРАБОТЧИКА	 160
1.1. Принципиальная схема	161
ПРИЛОЖЕНИЕ 2. КАРТА FLASH	162
ПРИЛОЖЕНИЕ 3. НАСТРОЙКИ ПО УМОЛЧАНИЮ.....	163
3.1. Блок конфигурации	163
ПРИЛОЖЕНИЕ 4. ОБНОВЛЕНИЕ ПО И НАСТРОЕК	165
4.1. Обновление ПО (FIRMWARE)	165
4.2. Пакетное обновление конфигурации	166
ПРИЛОЖЕНИЕ 5. ВЫЧИСЛЕНИЕ CRC.....	168
ПРИЛОЖЕНИЕ 6. ФОРМАТЫ ДАННЫХ	169
6.1. Предложение формата PGIO	169
6.2. Предложения формата NMEA	171
ПРИЛОЖЕНИЕ 7. ЗАРЕЗЕРВИРОВАННЫЕ СЛОВА.....	172
ПРИЛОЖЕНИЕ 8. СООБЩЕНИЯ	173
8.1. OK, END, ERROR и сообщения об ошибках	173
8.2. Коды ошибок GSM/GPRS	174
8.3. Отладочные сообщения.....	176
ПРИЛОЖЕНИЕ 9. ИНДИКАТОР ОПЦИЙ ПРОШИВКИ	180
ПРИЛОЖЕНИЕ 10. ЗАМЕТКИ АДМИНИСТРАТОРА.....	183

История документа

Дата	Произведённые изменения	Версия	Автор
30.06.04	Документ создан	1.0	М.Сол.
23.09.05	Вторая версия <i>Устройства</i> (firmware v2.x15).	2.0	
30.09.05	Команда &req.last=,hex возвращает Декущий пакет данных в 16-ричном формате. Изменение значения события TIMER_RUNi (см. Описание объекта <i>Таймер</i> и команду &set.timer). События OBJ_GSM: ALERT_XXX и CALL_MONITOR.	2.1	
20.12.05	Добавлено описание команд &req.confirm и &req.new Добавлены блок-схемы управления записи и эхо (команды &set.log и &set.echo) Устройство версии 2006г. – изменены реакции управления дискретными выходами и индексы выходов в команде &set.relay .	2.2	
09.01.06	Добавлено описание фильтра количества видимых спутников, параметр Nsat в команде &set.log	2.3	
04.02.06	Режим «удержания» GPRS канала связи (см. второй параметр команды &add.user).	2.4	
12.03.06	Режим энергосбережения (рекция SYS_SPEED объекта SYSTEM). Управление скоростью пакетного вывода данных (команда &set.system). Добавлен параграф Техническая поддержка . Дополнения в Приложении 9 Отладочные сообщения .	2.5	
29.03.06	Добавлены реакции для объекта OBJ_LOG . Добавлены комментарии в разделе Пример построения макро команды . Новая команда сброса системных настроек при включении питания (раздел 4. Команды). Сокращены параметры команды &get.gsm . Новое событие SYS_TIMEOUT. Добавлено описание зарезервированных слов, см. Приложение 7.	2.6	
22.08.06	Добавлено описание дифференциального режима записи данных в ЛОГ. Дополнено описание объекта Условие и таблицы констант для него. В дополнение к режиму перенаправления потока данных в GSM порт введен режим копирования данных. Реакция SYS_FORWARD2GSM переименована в SYS_DATA2GSM (объект OBJ_SYSTEM). Введена поддержка 16-ричного режима ECHO (см. третий параметр команды &set.echo) Добавлены описания событий, реакций и соотв. Константы объекта OBJ_SYSTEM .	2.7	
25.08.06	Добавлено описание объекта и значения констант для OBJ_KEY.	2.8	
17.10.06	Изменения значений констант объекта OBJ_LOG. Добавлено событие GSM_OFFLINE объекта OBJ_GSM . Новый параметр в команде &set.log . Новая команда &req.pos . Добавлена реакции MAKELINKNOW и CHANGESIM объекта OBJ_GSM. Добавлены события объекта OBJ_GPS и константы Условия дл него. Описание команды &get.temperature .	2.9	
25.10.06	Команда &restart . Дополнения в описании реакций SYSTEM_RESTART (объект OBJ_SYSTEM), DROP_LINE и POWER_RESET (объект OBJ_GSM).	2.10	
01.11.06	Параграф « Работа с программными Счетчиками ». Описание объекта OBJ_COUNTER и констант для него.	2.11	
18.11.06	Расширена интерпретация событий SMS_RECEIVED и ALERT_DATA. Новый параметр в команде &get.user .	2.12	
13.12.06	Добавлены тэги в ключевых словах (см. &add.key). Добавлена команда &add.log . Добавлено событие LOG_CONFIRMED.	2.13	
04.01.07	Добавлено описание объекта OBJ_CAMERA.	2.14	

Дата	Произведённые изменения	Версия	Автор
14.02.07	Добавлена возможность управления завершением сеанса IP-связи с Пользователем. Команда &add.user (см. параметр <mode>). Изменения в командах &set.cnt и &get.cnt	2.15	
10.03.07	Добавлен двоичный режим возврата данных &set.echo . Новый параметр команды &set.id для установки идентификатора пользовательских настроек. Настройки точки выхода в GPRS-сеть для устройств с GSM/GPRS-модулем Wavcom GR64 (новая версия приборов с марта 2007 года).	2.16	
07.04.07	Новый бит параметра <mode> в команде &add.user при работе с Web- серверами. Переработано «Приложение 4». Изменено толкование параметра команды &get.macro . Новый пример программирования, Регистрация данных расхода топлива .	2.17	
11.04.07	Пример программирования – Связь по каналу Bluetooth . Управление режимами возврата данных по событию (объект OBJ_LOG)	2.18	
13.08.07	Добавлено событие GPRS_LINKED объекта OBJ_GSM .	2.19	
08.09.07	Добавлено описание кодов ошибок выполнения команд AT*E2IPA и AT*E2IPO, параграф 8.2.	2.20	
01.12.07	Добавлено описание идентификатора реакции GSM_DROPLINE объекта OBJ_GSM .	2.21	
25.12.07	Добавлен Пример 14 «Учет времени работы двигателя».	2.22	
12.01.08	Добавлено описания работы с временными зонами . Работа с беспроводными картами доступа, новый тип пользователя (0).	2.23	
11.02.08	Добавлены примеры программирования « Работа с датчиком расхода топлива » и « Изменение периода записи ЛОГ в зависимости от скорости »	2.24	
05.03.08	Новый параметр команды &set.macro =<id>,run пользовательского запуска макроса. Новый параметр команды &set.log определяющий время ожидания слова-подтверждения получения данных Пользователем.	2.25	
25.03.08	Новый параметр команды &get.macro	2.26	
23.04.08	Рестарт модема по событию (см. описание реакции GSM_RESTART).	2.27	
04.04.08	Новая реакция OBJ_GSM .<message_id>.ISSUE_ATCOMAND Новое событие OBJ_SYSTEM .SET_WAITING.END_PRINT Новая реакция OBJ_MACRO .<macro_id>.RUN Новый параметр команды &get.macro [=<begin>[,<number>]] Новый параметр команды &set.macro	2.28	
12.06.08	Новая реакция OBJ_LOG .RESET_MILAGE.0 Новая реакция OBJ_LOG .READ_LAST_VALID_POS.0	2.29	
05.08.08	Добавлены два параграфа в разделе « Проблемы и решения » Развернуто описание работы с журналом событий . Новый параграф Работа с гео-одометром	2.30	
12.08.08	Добавлено событие CAM_SNAPSHOT_READY и реакция CAM_GET_PICTURE объекта OBJ_CAMERA Добавлен Решение 2 примера « Работа с датчиком расхода топлива » и новый пример « Передача фото изображения на IP-сервер » Добавлен объект пользовательских флагов OBJ_FLAGS , значения констант работы с этим объектом и описание Работа с пользовательскими флагами Исправлены значения констант ECHO_ALL, ECHO_VALID, ECHO_OFF Добавлено описание реакций управления режимом записи в ЛОГ (см. описание объекта OBJ_LOG): реакции REC_DISABLE и REC_ENABLE.	2.31	

Дата	Произведённые изменения	Версия	Автор
15.01.09	Добавлены реакции OBJ_ZONE.<zone_id>.ZONE_SET_ALARM, OBJ_GSM.0.GSM_MAKELINKNEXT, реакция блокировки записи ЛОГ. См. таблицу значений констант . Исправлен макрос 4 в примере « 5.13. Аварийное соединение с резервным сервером » Исправлено максимальное значение параметра <dz>, см. команду &set.system . Новое событие LOG_READCOMPLETED	2.32	
02.03.09	Добавлено описание режима программирования Устройства и параметра /s для &-команд .	2.33	
30.03.09	Расширен перечень тэгов разрешенных к использованию в составе шаблона сообщения.	2.34	
10.07.09	Пример голосового вызова из активной IP-сессии	2.35	
13.08.09	Изменен смысл значения идентификатора события GSM_RING. Изменен формат команд &req.new (параметр reverse больше не поддерживается) и &req.confirm (поддерживается только один параметр)	2.36	
16.03.10	Команда &set.baud имеет второй параметр Righnow, наличие которого изменяет скорость порта без ожидания подтверждения. В объект Условие добавлены проверка режима IP-соединения OBJ_GSM.GSM_LINKED.0, проверка режима записи OBJ_GPS.0.LOG_SET, а также бит «сложного» условия. Новая команда &get.log. Поддержка режима регистрации штрих-кода, новый параметр в команде &set.system. Новое событие OBJ_LOG.3.STORED . Команды: &set.debug, &set.program. Новый параметр команды &get.gsm . Добавлено описание новых тэгов. Новые команды &set.cmd и &get.cmd . Управление режимом ГЛОНАСС-GPS, см. команду &set.nmea .	2.37	
22.06.11	Добавлен и расширен раздел Индикатор опций прошивки	2.38	
05.07.11	Управление режимом навигационного приёмника «автомобиль/пешеход» (только для версии VENUS). Параметр <lm> команды &set.log и параметры <es> и <sz> команды &set.system . Добавлен параметр <id> в команды &del.macro , &del.condition и &del.timer .	2.39	
31.10.11	Ограничение доступа к командному интерфейсу &set.system , расширенный набор параметров команды &set.cnt , опция Increment/Decrement При работе с дискретными входами в режиме «счетчик»	2.40	

1. Введение

- Геликс-1** (устройство сбора и передачи данных) программируемый логический контроллер производящий регистрацию навигационных и цифровых данных, значений дискретных и аналоговых сигналов, а также управление дискретными и цифровыми выходами. Модель контроллера с индексом –G (Геликс-1G) оборудована встроенным навигационным приёмником SiRF-III. Модель контроллера с индексом –SAT (Геликс-1SAT) интегрирована со спутниковым модемом серии SAT-200 Inmarsat-D+.
- Геликс-2** (абонентское устройство сбора и передачи данных) телематический программируемый логический контроллер производящий регистрацию навигационных и цифровых данных, значений дискретных и аналоговых сигналов, а также управление дискретными и цифровыми выходами и встроенным GSM/GPRS- модулем. Модель контроллера с индексом –G (Геликс-2G) оборудована встроенным навигационным приёмником SiRF-III. Модель контроллера с индексом –S (например, Геликс-2GS) оборудована вторым держателем SIM-карты.

[1] *Книга I. Руководство по Эксплуатации* содержит описание компонент, режимы работы и подключение *Устройства*.

[2] *Книга II. Руководство Администратора* (программиста) содержит описание командного интерфейса, модуля тестирования и отладки. Книга II предназначена для специалистов разработчиков конечных решений производящих настройку/программирование *Устройства*.

[3] *Книга III. Тест План* содержит описания операционных обстановок, задач тестов и их пошаговое выполнение, ожидаемый и действительный результаты. Приведены примеры работы командного интерфейса, граничные значения и значения параметров команд по умолчанию. Документ совмещает два типа тест планов: тест план разработчика (development test) и оценочный тест план (evaluation test plan).

[4] Приложение. Структура ЛОГ.

[5] Приложение. Пользовательские регистры.

[6] Приложение. Работа с датчиками уровня и расхода топлива.

[7] Приложение. Сигнальные зоны.

[8] Приложение. Сигнальный процессор.

[9] AT Commands Interface Guide. Sierra Wireless.

ВНИМАНИЕ! Разделы, параграфы и термины, помеченные знаком **(GSM)** относятся только к GSM версии *Устройства*, контроллер Геликс-2.

Руководство может быть изменено производителем без уведомления пользователя.

2. Термины

Активное событие такое *Событие* системы, которое задано объявлено в *Макро*.

База данных См. *Журнал событий*.

Верные координаты навигационные данные, в которых отсутствует признак «предупреждение навигационного приёмника» (см. формат [RMC](#)) или индикатор качества не «ноль» и значение параметра HDOP менее “48”.

Временная зона определяет пару значений «начало» и «окончание» временного отрезка суточного времени, в момент достижения которых генерируется системное *событие*. *Временную зону* иначе можно назвать генератором *событий* по времени суток.

Гео-зона множество географических точек расположенных внутри предопределённой географической области (*Гео-зоны*) с заданным центром и имеющей форму круга радиусом (R) или квадрата со стороной ($R*2$). Понятие *Гео-зона* дополняет событийную модель *Устройства*, в которой вырабатывается *событие* при входе ТС в *Гео-зону* или выходе ТС из *Гео-зоны*.

Дискретный вход См. *Зона охраны*.

Дискретный выход *Объект* системы, соответствующий релейному выходу. Управление состоянием любого дискретного выхода может производиться по команде или *событию*. Количество дискретных выходов определяется версией *Устройства*.

Журнал событий массив записей фиксированного формата, производимых *Устройством* по *событию* и хранящийся в энергонезависимой памяти FLASH (см. раздел Журнал событий). Журнал событий, иначе можно назвать «история», или «LOG», или база данных событий (БД).

Зона охраны *Объект* системы, соответствующий дискретному входу и выполняющий функции охранного шлейфа (зоны охраны). Состояние Зоны охраны может изменяться по команде, *событию* или изменению внешних условий. При изменении внешних условий объект Зона охраны генерирует соответствующее *событие*. Количество Зон охраны определяется версией *Устройства*.

Зона скорости два значения SOG в узлах, определяющие диапазон скоростей *ТС*, при пересечении границ которого производится генерирование системного События OBJ_GPS.<where>.GPS_SPEED. Значение <where> определяет положение действительной скорости *ТС* «меньше» минимального значения *зоны скорости*, «внутри» *зоны скорости*, или «больше» максимального значения *зоны скорости*.

История событий См. *Журнал событий*.

Команда (или &-команда) посылается пользователем или диспетчерской системой по каналу *TE*, GSM SMS/CSD или GPRS и интерпретируется

Устройством в зависимости от содержания команды для выполнения таких действий, как «настройка», «управление» или «запрос» на выдачу данных. &-команда начинается символом “логическое и” (код 26h) и должна завершаться символами <Enter> (коды 0Dh и 0Ah). См. *параметр команды, сервисная команда*.

Ключевое слово *Объект* системы, задающий последовательность букв, знаков и/или цифр (отличная от любого зарезервированного слова), получение по командному интерфейсу которой генерирует системное событие. Ключевое слово может содержать тэг для «захвата» числа.

Конфигурация совокупность настроек *Устройства*, производимых администратором/оператором, включающая параметрические (см. команды &set) и функциональные настройки (см. команды &set и описания объектов *Пользователь, Сообщение, Условие, Таймер и Макро*).

Макро предопределённая инструкция, задающая правило управления *Устройством* по *Событию*. Инициатором исполнения *Макро* всегда является *Активное Событие*. Объект *Макро* может быть множественным (см. *Реакция, Активное событие, Объект, Условие, Событие*).

Новая запись пакет данных в *ЛОГ* с несброшенным флагом «прочитано». Флаг «прочитано» может быть установлен для каждого пакета при получении его в режиме «старт-стоп» или по команде &req.confirm.

Объект виртуальный процессор, отвечающий за обслуживание определённого компонента системы и обладающий именем (типом), уникальным номером и набором свойств. Например, объект «*дискретный выход*» объединяет понятие релейный выход *Устройства* и подключенную нагрузку, исполнительный механизм (например, реле).

Очередь заявок стек команд типа «первый вошел, первый вышел» (FIFO), в который записываются реакции типа «соединиться с *Пользователем*», «отправить *Сообщения*» или «исполнить &-команду» принятую по каналу SMS. Работа с очередью заявок производится с помощью *Макро*. Очередь заявок хранит номер *Сообщения*, номер *Пользователя*, номер *Макро-инструкции* (по которой произошла запись команды в стек) и номер *Условия* этой *Макро-инструкции*. Первая команда из стека будет исполнена, стек сдвинут и команда из стека удалена при выполнении двух условий: GSM-модем свободен и *Условие* верно. Работа с *Макро* предусматривает управление очередью заявок. Например, «очистить очередь заявок» и «записать команду в начало очереди заявок».

Пакет данных информация предопределенной структуры совмещающая идентификационный номер, дату, время, состояния входов, выходов, навигационные и служебные данные представленные в [4]. См. также *Текущий пакет данных* и *Пользовательский регистр*.

Параметр команды параметр команды (см. определение &-команда) представляется

ASCII-символом, буквой латинского алфавита или цифрой в зависимости от его назначения описанного в [1]. Если параметр представляется в 16-ричном виде, то его значения должно содержать чётное количество цифр (диапазон 0..9, A..F).

ПЛК программируемый логический контроллер.

Пользователь *Объект* системы, идентифицирующий точку доступа, командный порт или информационный центр. Объект *Пользователь* определяет номер телефона/модема или IP адрес сервера в сети Интернет на который *Устройство* может отправлять *Сообщение* или с которого может получать команды на исполнение. Объект *Пользователь* может быть множественным. См. *Текущий Пользователь*.

Пользовательский регистр элемент (регистр) оперативной памяти *Устройства* предназначенный для оперативного хранения данных принятых по последовательному интерфейсу или результата подсчёта импульсов на дискретном входе (см. *Счетчик*). Информационная ёмкость пользовательского регистра составляет 4 байта. При регистрации данных значение пользовательского регистра может быть представлено в завершении *Пакета данных* [4]. Информационная ёмкость добавленного значения настраивается и может составлять 4, 2 или 1 байт (команда [&add.log](#)). Максимальное количество пользовательских регистров фиксировано (см. команду [«?»](#)).

Порог скорости предопределённая администратором величина скорости в узлах (SOG). Устройство генерирует *Событие* в случае если действительная скорость движения транспортного средства превышает значение *пороговой скорости* на 5 узлов или становится строго ниже.

Прошивка (или firmware) микропрограммный двоичный код с инструкциями работы микроконтроллера/процессора (синоним «операционная система»). Поставляется технической поддержкой производителя *Устройства* в виде 16-ричного (.HEX) кодированного файла (см. раздел «[Версии программного обеспечения](#)» в этом документе и «Обновление firmware» в Книга II. Руководство администратора).

Реакция внутренний сигнал системы, который вырабатывается самой системой в результате обработки предопределённых правил, *Макро* инструкций.

Регистр см. термин *Пользовательский регистр*

Сервисная команда команда управления режимом запуска *Устройства* при включении питания (см. раздел [&-команды](#)).

Событие внутренний сигнал системы, который вырабатывается по факту изменения состояния *Объекта*. Например, системный *Объект* «Зона охраны» при замыкании вырабатывает *Событие* «Замкнуто». Не каждый *объект* в системе может вырабатывать *Событие*.

Сообщение *Объект* системы, содержащий текст и/или *Tэг(u)*.

Состояние (состояние объекта) цифровое значение, определяемое в момент

обращения к объекту запроса по команде или по *событию*.

Структура данных массив данных фиксированного формата [4].

Сценарий иерархия *объектов* (*Макро*, *Таймер*, *Пользователь*, *Сообщение* и проч.) настраивающая *Устройство* для выполнения задач поставленных пользователем системы.

Счетчик *Объект* системы, значение которого может изменяться на единицу (+1 или -1) или сброшено в «ноль» по *Событию*. В зависимости от версии прошивки информационная емкость *Счетчика* может быть один, два или четыре байта.

Таймер *Объект* системы, который создается для генерирования одиночного, серии или бесконечной последовательности *Событий* с заданным промежутком времени. *Событию* могут быть нумерованные с целью передачи номера *События* из последовательности. Объект *Таймер* может быть множественным. Объектом *Таймер* можно управлять по *Событию*.

ТЕ терминальная программа или программа пользователя запущенная на компьютере, производящая управление и получение данных от *Системы* по одному из каналов связи (последовательный порт, инфракрасный порт или Bluetooth).

Текущий пользователь это такой *Пользователь*, к которому произведено (или было произведено) подключение, вызов или послано/принято сообщение. Значение *Текущего Пользователя* меняется на новое при каждой новой активности с объектом *Пользователь* при приёме вызова или нового установленного соединения с *Пользователем*. Исключением является случай при активном IP-соединении, при котором значение (номер) *Текущего пользователя* не изменяется, если был отправлен или принят вызов Voice, CSD или SMS.

Текущий пакет данных это такой пакет данных, значения полей которого формируются каждую секунду и еще не записаны в *ЛОГ*. Порядковый номер такого пакета данных всегда на единицу больше, чем номер последнего записанного пакета данных в *ЛОГ*.

ТС транспортное средство.

Тэг предопределенная текстовая последовательность фиксированной длины в составе *Сообщения*. Используется для формирования текста на основании соответствующего *Тэгу* системного значения (или состояния объекта) на момент передачи сообщения.

Условие настраиваемая компонента *Макро* инструкции, определяющая дополнительное условие его выполнения. *Условие* позволяет производить опрос действительного состояния компонент *Устройства* на момент исполнения *Макро*.

Устройство это электронный прибор со встроенным микропроцессором для сбора и хранения навигационных данных, событий и программного

управления, в основу которого положен принцип объектно-ориентированного *Макро*-программирования и управления по *Событию*. В зависимости от исполнения *Устройство* может поддерживать работу с внешним навигационным приёмником, встроенными таймером реального времени (RTC), GSM-модулем и иными компонентами.

&-команда посылается пользователем или диспетчерской системой по каналу *TE*, GSM SMS/CSD или GPRS и которая интерпретируется *Устройством* в зависимости от содержания команды для выполнения таких действий как «настройка», «управление» или «запрос» на выдачу данных.

LOG См. *Журнал событий*.

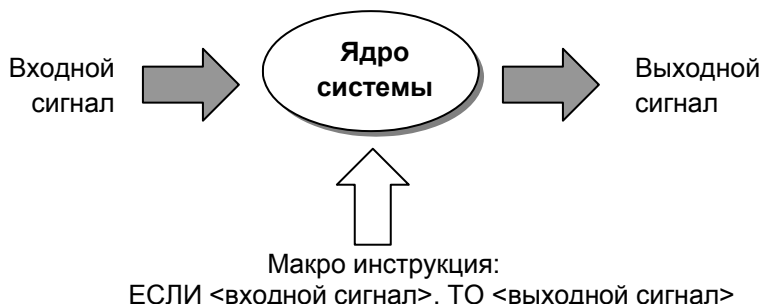
SOG (speed on the ground) скорость над поверхностью Земли.

3. Объектно-ориентированная модель

Объектно-ориентированная модель, положенная в основу разработки описываемой системы, имеет ряд преимуществ по сравнению с традиционными системами, работающими по жёстко заданной логике:

1. Простота настроек сложных функциональных зависимостей между входными и выходными сигналами, а также сигналами от внутренних элементов системы, таких как таймер, источник навигационных данных и др. Как следствие, это позволяет быстрее адаптировать логику работы системы по требованию заказчика, а также создавать новые решения. С другой стороны, изменение логики работы системы не требует знаний языков программирования, а значит, работу по адаптации системы может производить руководитель проекта.
2. Микропрограммное ядро системы (firmware) настроено и отлажено для интерпретации макро команд, задаваемых администратором системы. Следовательно, изменение логики обработки событий (макро команд) не требует затрат на перепрограммирование ядра и его тестирование. Тестированию подлежат коды макро команд открытые для администратора системы.

Понятие объектно-ориентированная система непосредственно связано с термином «управление по событию».



Абстрактно:

- 1). Ядро системы генерирует сигнал на выходе только в случае, если входной сигнал объявлен в макро инструкции. Если входной сигнал не объявлен в макро инструкции, то ядро системы не реагирует на него (игнорирует). Таким образом, можно говорить об активных *событиях* (входных сигналах) системы, задаваемых администратором системы.
- 2). Входной сигнал подаётся ядру системы от элемента системы. Например, Зона или Таймер. Назовём элемент системы термином *объект*.
- 3). Термин «входной сигнал» переименуем в «*событие*» и установим правило, что *событие* – это факт изменения состояния *объекта* системы. Таким образом, система всегда имеет в распоряжении полный набор *событий* от всех *объектов* системы, но реагирует только на *активные события*.

В сложной системе с множеством входных сигналов требуется идентификация входных сигналов по признакам:

- Тип объекта, от которого поступило входное сообщение.
- Идентификатор объекта (порядковый номер).
- Тип входного сообщения.

Указанную триаду получает ядро системы с приходом *события*. *Макро команда* должна определять эту триаду в первом параметре «ЕСЛИ».

Для идентификации выходных реакций тоже используется триада аналогичной структуры:

- Тип объекта, который подлежит управлению.
- Идентификатор объекта (порядковый номер).
- Тип выходного сообщения или команда.

Указанную триаду следует использовать во втором параметре *макро команды* «ТО».

Далее в этом разделе представлено описание *объектов*, работа с которыми производится с помощью макро-команд. Значения идентификаторов, событий и реакций приведены в разделе [3.17. Таблица значений констант](#) в этом документе.

3.1. Объект «Система»

Имя объекта	Описание
OBJ_SYSTEM	Объект соответствует системе.
Идентификатор	Описание
<command code>	Совместно с сообщением SYSTEM_CMD значение идентификатора соответствует типу введенных данных: CMD_GELIX - &-команда CMD_AT - AT-команда CMD_OK - ОК-команда CMD_UNKNOWN - другая команда
Событие	Описание
SYSTEM_RUN	Сообщение генерируется при запуске системы с задержкой в 2-3 секунды после включения питания.
SYSTEM_CMD	Сообщение генерируется после исполнения &- команды, принятой по командному интерфейсу. Тип команды передается значением идентификатора.
SYS_PRINT	Сообщение генерируется по факту окончания вывода пакета данных. Значение идентификатора соответствует типу процесса возврата данных, который претерпел завершение:

	<p>STRING – окончание возврата строки;</p> <p>REPLY – окончание возврата ответа исполнения процессором &-команды;</p> <p>ENDPRINT2TE – окончании возврата сообщения в порт TE согласно реакции OBJ MSG.<msg id>.PRINT2TE.</p>
SYS_CMD_OK	<p>Сообщение генерируется по факту получения ключевого слова OK по командному интерфейсу. Значение идентификатора соответствует числу следующему за ключевым словом и равно 0 (ноль) для ключевого слова OK или OK0. Максимальное значение равно 255 (например, OK255).</p> <p>Сообщение не генерируется, если <i>Устройство</i> ожидает от активного <i>Пользователя</i> подтверждения о получении «новых» данных. В этом случае рекомендуется использовать сообщение LOG_CONFIRMED объекта OBJ LOG.</p>
SYS_SYNCTIME	Сообщение генерируется после синхронизации таймера реального времени верными значениями даты и времени принятыми от навигационного приёмника.
SYS_TIMEOUT	Сообщение генерируется при возврате данных из ЛОГ в HEX режиме, в случае если <i>Устройство</i> не приняло подтверждения “<Enter>” или ключевого слова “OK<Enter>” от получателя данных (см. команды &req с ключом hex и описание тэгов %RC и %NC)
SYS_RESTART	Сообщение генерируется перед программным перезапуском <i>Устройства</i> (software reset).
Реакция	Описание
SYSTEM_RESTART	Иницирует программный перезапуск <i>Устройства</i> (software reset). Если реакция исполнена при активном соединении с IP-сервером, то IP-канал связи предварительно будет закрыт.
SYS_DATA2GSM	Перенаправляет (FWRD) или копирует (COPY) поток данных возвращаемых <i>Устройством</i> в канал GSM/GPRS. Значение идентификатора определяет режим «перенаправление» или «копирование». В режиме COPY <i>Устройство</i> производит копирование передаваемых данных в порт TE.
SYS_DATA2GSM_Q	Аналогично реакции SYS_DATA2GSM, но с предварительным сбросом <i>очереди заявок</i> на соединение и возврата сообщений.
SYS_TE2GSM_MIRROR	Устанавливает режим «сквозной» передачи данных между портами TE и GSM/GPRS. При включенном режиме: исполнение любых команд блокировано и включается режим перенаправления данных SYS_DATA2GSM/FWRD. Оба режима автоматически отключаются при разрыве соединения или выключении GSM-модуля. Значение идентификатора определяет действие «включить» или

	«выключить» режим. Если в установленном режиме «сквозной» передачи данных <i>Устройство</i> приняло три символа «плюс» (+++), то генерируется событие SYSTEM_DATA2GSM
SYS_SPEED	Производит управление скоростью работы процессора. Значение идентификатора определяет режим энергосбережения: от «0» до «2» для GSM версии; иначе – от «0» до «3». В режиме «0» снижение потребления электроэнергии не производится и <i>Устройство</i> работает на полной скорости (FULL SPEED). Режим «3» позволяет снизить энергопотребление в три раза, но при этом максимальная скорость обмена TE-порта составляет 9600 бит/сек и скорость программных таймеров, а также работа LED индикатора, снижается примерно в 24 раза.
SYS_SYNCTIME	Устанавливает флаг принудительной синхронизации встроенного таймера реального времени с глобальным временем. Флаг сбрасывается при первом получении валидных данных от навигационного приемника, значения даты и времени которых записываются в таймер реального времени. Значение идентификатора игнорируется.

3.2. Объект «GPS»

Имя объекта	Описание
OBJ_GPS	Объект соответствует навигационному приёмнику.
Идентификатор	Описание
INVALID	Координаты не верные (см. RMC формат, буква V в третьем параметре).
VALID_LAST*	Последняя верная координата (см. RMC формат, буква S в третьем параметре). * ВНИМАНИЕ! Признак “S” доступен только для приёмников производства FALCOM.
VALID	Верная координата (см. RMC формат, буква A в третьем параметре).
VALID_MASK	Объединяет оба типа навигационных данных VALID и VALID_LAST
Событие	Описание
GPS_READY4REC	Событие генерируется с периодом соответствующим значению параметра <t1> (&set.log) и при выполнении одного из условий: действительная скорость движения выше порогового значения <sp>, установлен «дифференциальный» режим записи, значение <t2> больше нуля (&set.log). Значение идентификатора указывает на качество принятых данных.
GPS_READY4ECHO	NMEA последовательность готова на вывод TE. Частота повторения этого события определяется командой &set.echo .

	Значение идентификатора указывает на качество принятых данных.
GPS_OFFLINE	Событие генерируется при не получении ожидаемого пакета NMEA предложений за определённый период времени (см. описание команды &set.nmea). Значение идентификатора равно 0.
GPS_RECEIVED	Событие генерируется при получении NMEA пакета данных от навигационного приёмника. Значение идентификатора указывает на качество принятых координат.
GPS_SPEED	Событие генерируется при превышении значения <i>порога скорости</i> и <i>зоны скорости</i> определённого третьим восьмым параметрами команды &set.system (SOG) или если скорость ТС строго меньше значения <i>порога скорости</i> . Идентификатор соответствует величине скорости «меньше» (SPEED_LESS), «больше» (SPEED_OVER) или «внутри <i>зоны скорости</i> » (SPEED_MIDDLE).
GPS_SPEEDEVENT	(не используется)
GPS_FIXCHANGED	Событие генерируется при изменении значения параметра FIX. Значение идентификатора равно новому состоянию навигационного сигнала (см. INVALID, VALID LAST* или VALID)
Реакция	Описание
GPS_POWER	Производит управление питанием встроенного навигационного приёмника (опция). Значение идентификатора определяет режим питания: «включить», «выключить» или «рестарт» (кратковременное выключение питания встроенного навигационного приёмника). При выключении питания по событию: - текущий пакет данных сохраняет последнее значение координат; - значения битов «флага состояние» fix indicator mask (03h) и признаков валидной координаты (A0h) текущего пакета данных, а также значения «количество видимых спутников» текущего пакета данных равны 0; - события GPS_READY4ECHO, GPS_READY4REC и GPS_RECEIVED не генерируются.
GPS_SEND_MSG	Отправить команду в последовательный интерфейс навигационного приёмника (опция). Команда определяется шаблоном сообщения с номером указанным в значении идентификатора объекта.
GPS_SET_BAUD	Установить скорость обмена последовательного интерфейса <i>Устройства</i> , к которому подключен навигационный приёмник (опция).
GPS_SET_PINNING	Устанавливает time position pinning (опция, только для версии GPS с приёмником VENUS). См. параметр <pp> команды &set.log

Термины объекта OBJ_GPS при работе с объектом Условие:

Состояние	Описание
GPS_INVALID	Данные принимае от навигационного приёмника невалидные.
GPS_VALIDLAST	Последний валидный пакет данных принятый от навигационного приёмника (только для GPS-приёмника производства FALCOM).
GPS_VALID	Данные принимае от навигационного приёмника валидные.
GPS_ALL_FILTERS	Аналогично GPS_VALID, но с дополнительной проверкой на одновременное выполнение условий: HDOP меньше или равно установленного порогового значения (см. &set.nmea), значение скорости движения SOG больше порогового значения (см. &set.log), значение YY в поле дата (год) больше или равно году выпуска прошивки, количество видимых спутников (значение NSAT) больше или равно порогового значения (см. &set.log).

3.3. Объект «Выход»

Имя объекта	Описание
OBJ_RELAY	Соответствует программируемым выходам системы.
Идентификатор	Описание
RELAY_ID1	Идентификатор выхода №1
RELAY_ID2	Идентификатор выхода №2
RELAY_ID3	Идентификатор выхода №3
RELAY_ID4	Идентификатор выхода №4
128+<bitmask>	Значение <bitmask> в бит-позиционном коде, младший бит соответствует первому выходу.
Событие	Описание
нет	
Реакция	Описание
RELAY_ON	Включить (переключить выход в состояние «замкнуто»)
RELAY_OFF	Выключить (переключить выход в состояние «разомкнуто»)

3.4. Объект «Зона охраны»

Имя объекта	Описание
OBJ_ZONE	Объект соответствует понятию «зона охраны» (Alarm input) или «дискретный вход».
Идентификатор	Описание
ZONE_ID1	Идентификатор Зоны охраны. Иначе, номер дискретного

ZONE_ID2	
ZONE_ID3	
ZONE_ID4	
Событие	Описание
ZONE_OPEN	Происходит при разрыве <i>Зоны</i> охраны.
ZONE_CLOSED	Происходит при замыкании <i>Зоны</i> охраны.
ZONE_ALARM	Происходит при изменении состояния <i>Зоны</i> охраны, которое отличается от заданного нормального состояния, NO (normal open) или NC (normal closed).
Состояние	Описание
ZONE_ENABLED	<i>Зона</i> включена. В этом состоянии объект <i>Зона</i> может генерировать событие ZONE_OPEN или ZONE_CLOSED в зависимости от изменения сигнала на входе <i>Устройства</i> .
ZONE_DISABLED	<i>Зона</i> выключена. В этом состоянии объект не генерирует.
ZONE_ARMED	Состояние «на охране».
ZONE_DISARMED	Состояние «снято с охраны».
ZONE_OPEN	<i>Зона</i> разомкнута
ZONE_CLOSED	<i>Зона</i> замкнута
NORMAL_OPEN	Нормальное состояние разомкнуто.
NORMAL_CLOSED	Нормальное состояние замкнуто.
Реакция	Описание
ENABLE	Включить объект <i>Зона</i> .
DISABLE	Выключить объект <i>Зона</i> . Система игнорирует любые события от объекта в этом состоянии.
ARM	Поставить <i>Зону</i> на охрану.
DISARM	Снять <i>Зону</i> с охраны.
CONFIRM	Подтвердить сигнал «тревога». <i>Зона</i> остаётся в режиме «на охране».

3.5. Объект «Аналоговый вход»

Имя объекта	Описание
OBJ_ADC	Объект соответствует понятию «аналоговый вход».
ADC_LESS	Применяется только в определении параметров объекта <i>Условие</i> . Объект <i>Условие</i> (ADC_LESS,<adc_id>,<value>) принимает значение ИСТИНА на момент обращения, если уровень напряжения на аналоговом входе меньше указанного значения <value>, <i>Состояние</i> этого объекта.

ADC_MORE	Применяется только в определении параметров объекта <i>Условие</i> . Объект <i>Условие</i> (ADC_MORE,<adc_id>,<value>) принимает значение ИСТИНА на момент обращения, если уровень напряжения на аналоговом входе больше указанного значения <value>, <i>Состояние</i> этого объекта.
Идентификатор	Описание
{1 2 3 4}	Идентификатор определяет номер аналогового входа <i>Устройства</i> . Нумерация аналоговых входов начинается с единицы.
Событие	Описание
ADC_LESS	Происходит при первом понижении уровня входного сигнала ниже порогового «Мин».
ADC_MIDDLE	Происходит при первом условии соответствующим уровнем входного сигнала более или равно пороговому значению «Мин» и менее или равно пороговому значению «Макс».
ADC_MORE	Происходит при первом превышении уровня входного сигнала порогового значения «Макс».
Состояние	Описание
ADC_LESS	Уровень входного сигнала ниже порогового «Мин».
ADC_MIDDLE	Уровень входного сигнала выше (или равен) порогового «Мин» и ниже (или равен) порогового «Макс».
ADC_MORE	Уровень входного сигнала выше порогового «Макс».
Реакция	Описание
нет	

3.6. Объект «Гео-зона»

Имя объекта	Описание
OBJ_GEO	Объект соответствует понятию «гео-зона» (Geo fence).
Идентификатор	Описание
<geo_id>	Порядковый номер <i>Гео-зоны</i> в системном списке.
Событие	Описание
GEO_IS_IN	Объект «вошёл» в гео-зону с номером <geo_id>.
GEO_IS_OUT	Объект «вышел» из гео-зоны с номером <geo_id>.
Состояние	Описание
GEO_IS_IN	Состояние объекта в или вне гео-зоны с заданным <geo_id>.

GEO_ENABLED	Состояние гео-зоны «включено» разрешает устройству генерировать сравнивать координаты объекта с гео-зоной.
GEO_RECT	Тип гео-зоны «квадрат»
GEO_ROUND	Тип гео-зоны «круг»
Реакция	Описание
нет	

3.7. Объект «Пользователь»

Имя объекта	Описание
OBJ_USER	Этот объект определяет номер телефона или модема абонента, IP-адрес или имя хоста в сети Интернет или номер беспроводной карты доступа (см. раздел «Работа с объектом Пользователь»), а также описание событий SMS_RECEIVED, ALERT_DATA и ALERT_VOICE для Объекта GSM/GPRS-модуль).
Идентификатор	Описание
<user_id>	Номер <i>Пользователя</i> в системном списке
Событие	Описание
USER_ACCESS	Карта доступа внесена в поле считывателя. Код карты доступа сравнивается с адресами из системного списка <i>Пользователей</i> . Идентификатор соответствует номеру <i>Пользователя</i> найденного в системном списке или «0» если не найден.
Реакция	Описание
нет	

3.8. Объект «Сообщение»

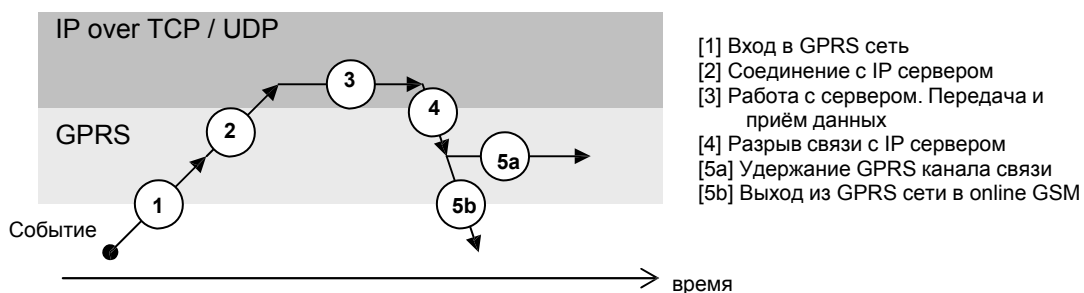
Имя объекта	Описание
OBJ_MESSAGE	Используется для передачи предопределённой структуры данных или значения переменной в указанный порт или <i>Пользователю</i> (см. объект OBJ_USER)
Идентификатор	Описание
<msg_id>	Значение идентификатора объекта определяет номер шаблона сообщения из списка (см. команды &add.message и &get.message). В составе макроопределения значение <msg_id> может иметь установленным старший бит (+128), который означает «высокий приоритет» сообщения. Сообщение с установленным битом «высокий приоритет» записывается в начало очереди

	сообщений.
Событие	Описание
Нет	
Реакция	Описание
<user_id>	Значение <i>Реакции</i> соответствует номеру объекта <i>Пользователь</i> в системном списке. Нумерация <i>Пользователей</i> начинается с единицы. Максимальное количество определяется версией прошивки и не может быть больше 127.
PRINT	Возвращает сообщение с номером <msg_id> в ТЕ-порт <i>Устройства</i> с учётом режима перенаправления потока данных в IP (или CSD) канал GSM-модуля (см. реакцию SYS_DATA2GSM для OBJ_SYSTEM).
PRINT2TE	Возвращает сообщение с номером <msg_id> только в ТЕ- порт <i>Устройства</i> . Режим перенаправления потока данных не учитывается.
PRINT2RS485	Возвращает сообщение с номером <msg_id> только в порт RS485 <i>Устройства</i> (если аппаратно таковой порт поддерживается). Режим перенаправления потока данных не учитывается.

В зависимости от значения объекта *Пользователь Устройство* инициирует одно из следующих соединений:

- GPRS** объект *Пользователь* типа IP сервер (4 или 5) или хост (6 или 7)
CSD объект *Пользователь* типа модем (3)
SMS объект *Пользователь* типа телефон (1 или 2)

Работа по отправлению сообщений с *Пользователем* следует поочерёдному выполнению трёх этапов 1) соединение, 2) отправление сообщения и 3) закрытие сеанс связи. Таким образом, при завершении передачи данных соединение с *Пользователем* автоматически прерывается. Для *Пользователей* типа 4 и выше после разрыва связи с сервером GPRS-соединение может остаться активным (см. вариант 5a на схеме ниже). В этом случае в настройках *Пользователя* должен быть установлен флаг «удержание GPRS».



Для типа *Пользователя* типа Web-сервер (см. второй параметр команды [&add.user](#)) предполагается, что после обработки сообщения (например, соответствующего методу HTTP GET) сервер вернёт в *Устройство* какое-либо ключевое слово (например, OK1) и самостоятельно закроет сеанс IP-связи.

Устройство, приняв ключевое слово, может обработать его и запустить соответствующий макрос.

При отправлении сообщения на Web-сервер, в состав которого входит тэг %RC, а также, если верно утверждение «значение параметра <nr> отлично от нуля (см. команду [&set.system](#))», то при получении от сервера ключевого слова OK в переданных пакетах записей будет снят флаг «новая запись». Таким образом, при производится досылка записей ЛОГ на сервер.

3.9. Объект «GSM/GPRS-модуль»

(GSM)

Имя объекта	Описание
OBJ_GSM	Объект соответствует встроенному GSM/GPRS-модулю <i>Устройства</i> .
Идентификатор	Описание
*	Значение идентификатора зависит от типа системного сообщения или управляющей реакции.
Событие	Описание
GSM_OK	Сообщение посылается с идентификатором равным «нуль» при успешном выполнении AT-команды. Идентификатор отличный от «нуля» означает: 1 – GSM-модуль выключен 3 – GPRS-соединение разорвано 4 – CSD-соединение разорвано 5 – «голосовое» соединение разорвано
GSM_ERROR	Сообщение посылается в случае задержки ответа GSM/GPRS-модуля на исполнение AT-команды.
CONNECT	Сообщение посылается при подсоединении GSM-модуля к линии типа IP-сервер, хост или модем. Значение идентификатора равно номеру <i>Пользователя</i> , к которому было произведено соединение.
NOCARRIER	Сообщение посылается если при обрыве IP/GPRS или CSD соединения (no carrier). Значение идентификатора равно номеру <i>Пользователя</i> , к которому было произведено соединение. Режим перенаправления данных (см. реакцию SYS_TE2GSM_MIRROR объекта OBJ_SYSTEM) сбрасывается.
INIT_END	Сообщение посылается по факту окончания инициализации GSM/GPRS-модуля после его включения. Инициализация GSM/GPRS-модуля выполняется каждый раз после включения питания GSM/GPRS-модуля.
SIMPIN	Сообщение посылается, если PIN верен или SIM-карта не защищена паролем. Значение идентификатора соответствует результату исполнения команды AT+CPIN? (требуется ввод PIN,

	PIN не требуется или введен корректно, SIM-карта заблокирована).
SMS_RECEIVED	Принято SMS-сообщение. Значение идентификатора соответствует номеру найденного <i>Пользователя</i> в списке, номер которого совпадает с номером отправителя сообщения. Идентификатор будет равен «0», если <i>Пользователь</i> не найден в списке или принято сообщение от найденного <i>Пользователя</i> в списке с приоритетом «0». Если режим авторизации выключен, то номер телефона пославшего сообщение будет добавлен в список <i>Пользователей</i> с приоритетом «0» (не активен). Принятое сообщение игнорируется, если включен режим авторизации и приоритет <i>Пользователя</i> пославшего сообщение равен «0». Идентификатор в этом случае будет равен «0».
ALERT_DATA	Входящий вызов типа «данные» (data call) опознан. Событие ALERT_DATA генерируется однократно по факту первого сигнала вызова (см. событие RING). Идентификатор принимает значение, соответствующее номеру найденного <i>Пользователя</i> в списке, производившего вызов. Идентификатор будет равен «0», если <i>Пользователь</i> не найден в списке или принят вызов от <i>Пользователя</i> с приоритетом «0». Если режим авторизации выключен, то номер телефона звонящего <i>Пользователя</i> будет добавлен в список <i>Пользователей</i> с приоритетом «0» (не активен). Входящий вызов сбрасывается, если включен режим авторизации и приоритет звонящего <i>Пользователя</i> равен «0» или <i>Пользователь</i> не найден в списке. Идентификатор в этом случае будет равен «0».
ALERT_VOICE	Входящий голосовой вызов опознан. Событие ALERT_VOICE генерируется однократно по факту первого сигнала входного вызова (см. событие RING). Значение идентификатора равно нулю.
CALL_MONITOR	(только для модема GR64) Событие генерируется однократно при входящем или исходящем вызове. Значения идентификатора соответствует: <div style="margin-left: 40px;"> 0 Idle 2 Connecting 3 Active (connection between A and B) 4 Hold 5 Waiting 7 Busy </div>
RING	Событие генерируется при входящем вызове периодически до момента сброса или приёма входящего вызова. Значение идентификатора соответствует порядковому номеру сигнала RING. Нумерация сигнала RING начинается с единицы.
MSG_SENT_OK	SMS сообщение отправлено. Значение идентификатора равно номеру <i>Пользователя</i> , получателя сообщения.

GPRS_LINKED	Подсоединение к сети GPRS завершено успешно (см. сообщение ERR_GPRS). Значение идентификатора равно номеру <i>Пользователя</i> , к которому производилась попытка соединения.
REGIST	Посылается <i>Системой</i> в результате регистрации в GSM- сети. Значение идентификатора соответствует сети GSM- оператора: 2 «home network» или 3 «roaming».
GSM_OFFLINE	Посылается <i>Системой</i> в результате выхода из GSM-сети. Значение идентификатора указывает что «потеря» GSM-оператора произошла впервые или оператор отсутствует по истечении определенного интервала времени.
ERR_IP_LINK	Сообщение посылается в при неудачной попытке подсоединения к указанному хосту или IP серверу. Значение идентификатора равно номеру <i>Пользователя</i> , к которому была произведена попытка соединения.
ERR_GPRS	Сообщение посылается в при неудачной попытке подсоединения к сети GPRS (см. сообщение GPRS_LINKED). Значение идентификатора равно номеру <i>Пользователя</i> , к которому производилась попытка соединения.
POWER_OFF	Сообщение генерируется при выключении модема. Перед этим событием вырабатывается событие GSM_OFFLINE
Реакция	Описание
DROP_LINE	При значении идентификатора равно “0” эта реакция соответствует команде «сбросить вызов» (ATH для соединения с <i>Пользователем</i> типа «телефон», или DTR для <i>Пользователя</i> типа «модем», или разорвать IP- соединение для <i>Пользователя</i> типа «сервер» или «хост»). В случае активного IP-соединения с <i>Пользователем</i> , у которого установлен флаг «удержание GPRS-канала», исполнение этой реакции разорвёт только IP-связь, а при повторном исполнении этой реакции (т.е. IP-канал закрыт и GPRS активен) IP-стек будет деактивизирован и <i>Устройство</i> «завершит» связь с GPRS-сетью. Значение идентификатора может быть равно “1”. В этом случае при активном IP-соединении модем будет переведён в командный режим.
PICKUP_LINE	Соответствует команде «принять вызов» (ATA). Рекомендуется исполнять по событию входящего вызова (например, OBJ_GSM.<ring_id>.RING).
POWER_CTRL	Изменяет состояние GSM/GPRS-модуля по питанию. Идентификатор объекта соответствует одному из действий «выключить», «включить» или «выключить и включить». Примечание: при выключении питания генерируется системное событие GSM_OFFLINE. Если выключение модема выполняется при активном соединении с IP- сервером, то IP-канал связи предварительно закрывается. По окончании инициализации производится генерируется событие INIT_END и производится

	подключение к сети GSM.
CHANGESIM	В версии с двумя SIM-держателями производит переключение на другую SIM-карту. Идентификатор объекта соответствует номеру SIM-карты или команде типа «переключиться на другую SIM-карту». При этом модем претерпевает процесс выключения и включения по питанию с последующей инициализацией AT-командами.
MAKELINK	При выполнении условия «модем свободен» команда соединения с <i>Пользователем</i> (GPRS, CDS или Voice в зависимости от типа <i>Пользователя</i>) добавляется в <i>очередь заявок</i> . Идентификатор объекта соответствует номеру <i>Пользователя</i> . Например, с целью восстановления IP-канала связи рекомендуется использовать с регулярно повторяющимся процессе по событию OBJ_GPS.<any>.RECEIVED или OBJ_TIMER.<id>.RUN.
MAKELINKNOW	Аналогично реакции MAKELINK, но без проверки условия «модем свободен». При исполнении такого макроса заявка на соединение записывается в очередь заявок и исполняется когда модем становится "свободным", GSM-или GPRS-online.
MAKELINKNEXT	Аналогично реакции MAKELINKNOW, но без указания номера пользователя из списка. Соединение производится со следующим пользователем из списка (см. ответ на команду &get.user) относительно текущего, к которому было произведено соединение. Поиск следующего производится по принципу «петля» и только для Пользователей типа «IP-сервер» или «Имя домена».
POWER_RESET	Иницирует рестарт GSM/GPRS-модуля по питанию с последующей инициализацией AT-командами и рестартом процессора модема (Reset Wireless CPU). См. примечание к реакции POWER_CTRL.
ISSUE_ATCOMAND	Посылает AT-команду модему согласно шаблона сообщения с номером в значении идентификатора. AT-команда в шаблоне должна быть без префикса AT (например, &add.me=+CGDCONT=1,"IPV4","internet.mts.ru")

3.10. Объект «Таймер»

Имя объекта	Описание
OBJ_TIMER	Объект соответствует программному таймеру системы.
Идентификатор	Описание
<timer_id>	Уникальный идентификатор объекта <i>Таймер</i> . Нумерация объектов начинается с «1».
Событие	Описание
TIMER_RUN_i	Только для типа TIMER_LIMITED. Генерируется при запуске

	таймера на новый цикл (первое событие имеет значение 1, второе событие 2 и так далее до значения <repeat>).
TIMER_RUN	Только для типа TIMER_NONSTOP . Происходит при запуске таймера с переходом на новый цикл отсчета времени (значение события всегда равно 1).
TIMER_STOP	Происходит один раз после окончания работы таймера (см. диаграмму работы таймера ниже).
Состояние	Описание
TIMER_STOP	Состояние выключенного или окончившего работу таймера (значение 0). В этом состоянии объект <i>Таймер</i> не генерирует событий.
TIMER_NONSTOP	Состояние соответствует бесконечному циклу работы таймера; без остановки, но с заданным параметром времени задержки. При каждом новом цикле выдаётся событие TIMER_RUN .
TIMER_LIMITED	Состояние соответствует ограниченной работе таймера с количеством циклов запуска заданного параметром <repeat>.
<delay>	Произвольно задаваемое значение в диапазоне от 1 до 65534. Определяет задержку в секундах генерирования события TIMER_RUN .
<repeat>	Только для таймера типа TIMER_LIMITED . Произвольно задаваемое значение в диапазоне от 1 до 65534. Определяет количество циклов <i>Таймера</i> . Значение <repeat> не влияет на работу <i>Таймер</i> типа TIMER_NONSTOP
Реакция	Описание
TIMER_RUN	Запустить таймер.
TIMER_STOP	Остановить таймер.

3.11. Объект «Условие»

Объект типа «условие» позволяет узнать текущее значение некоторых системных переменных или флагов состояний объектов системы. Объект типа «условие» определяет условное выражение, которое может учитываться при исполнении *Макро* инструкции. Объект типа «условие» всегда имеет одно из значений (истина или ложь) в момент исполнения того *Макро*, в котором оно объявлено.

Перед запуском *Макро* инструкции система всегда проверяет, определён ли объект «условие» для этого *Макро* или нет. Если объект «условие» определён и его значение «истина», то такая *Макро* инструкция выполняется системой. Иначе *Макро* инструкция исполнена не будет.

Объект «условие» задаётся тройкой или четверкой параметров:

Тип объекта, Идентификационный номер объекта, Состояние
Тип объекта, Идентификационный номер объекта, Состояние, NOT

Четвертый параметр позволяет выполнять операцию логического «НЕ» и инвертировать результат запрошенного состояния объекта.

Перечень состояний объекта *Условие* с соответствующими цифровыми значениями представлен в разделе [3.17. Таблица значений констант](#) в этом документе.

Комментарии к параметрам объекта *Условие* представлены в таблице ниже.

Тип объекта	(*) ID или параметр	Запрос на проверку значения (value)
OBJ_ZONE	Идентификатор(ы) зоны или дискретного входа	«охрана» (ZONE_ARM) «тревога» (ZONE_ALARM) «снята с охраны» (ZONE_DISARM) «замкнуто» (ZONE_CLOSED) «разомкнуто» (ZONE_OPEN)
OBJ_RELAY	Идентификатор дискретного выхода	«включено» (RELAY_ON) «выключено» (RELAY_OFF)
OBJ_GPS	(не используется)	«неверные данные» (GPS_INVALID) «верные данные» (GPS_VALID) «фильтры режима записи» (см. &set.log)
OBJ_TIMER	Идентификатор объекта Таймер	«таймер запущен» (TIMER_RUN) «таймер остановлен» (TIMER_STOP)
OBJ_SYSTEM	CHECK_SYSNEWREC	Параметр не используется. Производится проверка условия <количество новых записей> больше или равно <value> (см. команду &set.system =,,,<value>)
	CHECK_CUSTUMNEWREC	<количество новых записей> больше или равно <value>
	CHECK_SYSSTATUS	Результат побитового умножения <value> & <system_status>

	CHECK_SYSTEMPRINT	Параметр не используется. Возвращается «истина» если устройство производит потоковый вывод данных из ЛОГ
OBJ_GEO		
OBJ_USER		
OBJ_GSM	Не строгое сравнение	<gsm_status> & <value> != 0
	Строгое сравнение	(<gsm_status> & <value>) == <value>
	Проверка баланса	<gsm_ballance> меньше <value>
	Проверка индекса GSM-оператора	0 не известный оператор 1 MTS 2 Beeline 3 Megafon
OBJ_ADC	ID алогового входа	Значение сигнала на входе «меньше», «больше» или «между» установленных триггеров MAX и MIN (см. команду &set.analog)
OBJ_LESS	ID алогового входа	Результат сравнения: уровень сигнала на аналоговом входе меньше значения <value>
OBJ_MORE	ID алогового входа	Результат сравнения: уровень сигнала на аналоговом входе больше значения <value>

* - к значению идентификатора может быть добавлен признак «сложного» или «совмещённого» условия. При этом объект *Условие* принимает значение ИСТИНА, если выполняется основное условие определённое его параметрами в случае активного IP-соединения.

3.12. Объект «Журнал событий»

Имя объекта	Описание
OBJ_LOG	Объект соответствует встроенной энергонезависимой памяти системы (<i>журнал событий</i> или <i>LOG</i>).
Идентификатор	Описание
WRITE	Производит запись текущих данных в <i>журнал событий</i> . Значение реакции не используется (0)
SET_READ_MODE	Устанавливает режим возврата пакета данных (см. ниже ECHO, READ_LOG и READ_LOG_REV), который определяется битами значения реакции: бит 01h равен «0» HEX (аналогично &set.echo=,08); бит 02h равен «1» ожидание подтверждения (аналогично ключу “H” команды &req.last); бит 04h равен «1» запрос «новых записей» (аналогично ключу “H” команды &req.new); бит 08h равен «1» аналогично ключу “X” команд &req . Важно, чтобы команда SET_READ_MODE имела тоже событие (например, как для READ_LOG) и предшествовала в списке

	<p>макро инструкций команде ECHO, READ_LOG или READ_LOG_REV. Например:</p> <p>Макро_n <событие1> OBJ_LOG.SET_READ_MODE.<режим> Макро_{n+1} <событие1> OBJ_LOG.SET_READ_MODE.READ_LOG</p>
ECHO	Возвращает текущий пакет данных (аналогично &req.last без параметра)
READ_LOG	Возвращает <N> записей ЛОГ. Значение <N> определяется пятым параметром команды &set.system . Записи ЛОГ возвращаются в порядке убывания их порядковых номеров.
READ_LOG_REV	Аналогично READ_LOG, но возврат данных производится в порядке увеличения номеров записей ЛОГ.
CONFIRM_READ	Устанавливает флаг «прочитано» для указанного количества записей начиная с последней (или первой) переданной записи в направлении следования номеров записей согласно установке произведённой по реакции SET_READ_MODE. Количество записей определяется значением поля «реакция». Если значение реакция равно нулю, то флаг «прочитано» будет установлен для всех записей в ЛОГ.
CONFIRM_READ_OK	Аналогично CONFIRM_READ. Используется только в макросе с исходным событием OBJ_SYSTEM.255.CMD_OK, где количество «подтверждаемых» записей определяется числом следующим за ключевым словом OK. Например, OK10<Enter>.
SET_LOG_PERIOD	Устанавливает интервал записи данных аналогично команде &set.log=<t1> . Значение реакции соответствует интервалу в секундах и должно принимать значение 5 или больше.
SET_LOG_STATUS	Устанавливает режим записи данных в ЛОГ. Значение идентификатора определяет режим записи: (2)REC_OFF, (3)REC_ALL, (1)REC_VALID, (4)REC_DISABLE или (5)REC_ENABLE. Режим записи REC_DISABLE отключает любую запись данных в ЛОГ (периодическое, по событию и т.п). Режим записи REC_ENABLE восстанавливает рабочий режим записи до отключения произведённым REC_DISABLE. При изменении режима записи реакциями REC_DISABLE и REC_ENABLE производится сохранение изменений аналогично действию команды &set.save .
SET_ECHO_PERIOD	Устанавливает интервал вывода данных аналогично команде &set.echo=<period> . Значение реакции соответствует интервалу в секундах и должно больше нуля.
SET_ECHO_STATUS	Управляет режимом автоматического вывода данных <i>TE</i> (аналогично команде &set.echo). Значение идентификатора определяет режим вывода данных <i>TE</i> :

	(2) ECHO_OFF, (3) ECHO_ALL или (1) ECHO_VALID.
READ_LAST_VALID_POS	Возвращает последний пакет валидных данных.
RESET_MILAGE	Сбрасывает значение гео-одометра.
SET_LOG_PERIOD2	Устанавливает интервал записи данных аналогично команде &set.log =,,,,,,,<t2>. Значение реакции не может быть нулевым и соответствует интервалу в секундах <t2>*10.
Событие	Описание
LOG_RESTART	Событие генерируется в случае полного заполнения журнала событий и начала записей событий снова в первую страницу. Значение реакции равно «0».
LOG_NEWREC_OVER	Событие генерируется при превышении количества «новых» записей пороговому значению <nrg>, которое устанавливается командой &set.system . Если значение параметра <nrg> равно нулю, событие не генерируется.
LOG_RECORDED	Событие генерируется при записи пакета данных в ЛОГ в случае, если значение параметра <nrg> равно нулю или действительное значение новых записей не больше <nrg> (см. &set.system). Значение идентификатора соответствует признаку валидности навигационных данных. Событие также генерируется со значением идентификатора BARCODE, если ТЕ-порт принял строку и включен режим сканирования штрих кода (см. &set.system).
LOG_CONFIRMED	Событие генерируется при подтверждении получения пакета новых записей (см. описание команды &req.confirm и реакции CONFIRM_READ объекта OBJ_LOG).
LOG_READCOMPLETED	Событие генерируется по факту окончания возврата данных из ЛОГ (например, в случае возврата сообщения End при исполнении команд &req).
LOG_SPEEDEVENT	Событие генерируется с частотой зависящей от скорости движения. Чем выше значение действительной скорости движения (SOG), тем чаще генерируется событие SPEEDEVENT.
Реакция	Описание
*	См. описание значения идентификатора.

3.13. Объект «Макро команда»

Имя объекта	Описание
MACRO	Объект Макрос. Задаёт пару триад типа <Объект>, <Идентификатор> и <Событие>. Первая триада соответствует входному событию, поступившему в результате изменения состояния какого-либо объекта. Вторая триада определяет

	требуемую реакцию системы на объект для выполнения соответствующего действия. Дополнительным (не обязательным) параметром является идентификатор объекта « <u>условие</u> ».	
	Триада 1 : ИНТЕРПРЕТИРУЕМОЕ СОБЫТИЕ	
	Объект	См. раздел 3.17. Таблица значений констант
	ID	
	Событие	
	Триада 2 : СИСТЕМНАЯ РЕАКЦИЯ	
	Объект	См. раздел Таблица значений констант
	ID	
	Реакция	
	ДОПОЛНИТЕЛЬНЫЙ ПАРАМЕТР: УСЛОВИЕ	
	<id>	Идентификационный номер объекта «условие». Если параметр не указан значение <id> равно нулю, что соответствует понятию «условие не задано». Если условие не задано, то Макро инструкция будет исполняться согласно логике её определения.
Идентификатор	Описание	
<ID>	Уникальный идентификатор объекта Макро.	
Событие	Описание	
нет		
Реакция	Описание	
ENABLE	Включает макроинструкцию с заданным в идентификаторе номером.	
DISABLE	Выключает макроинструкцию с заданным в идентификаторе номером. Макро инструкции не исполняется, если она отключена.	
RUN	Запускает макроинструкцию с номером определённым значением идентификатора.	

3.13.1. Правила построения макро команды

Построение макро команды следует общепринятому построению сентенции “ВОПРОС-ОТВЕТ” и опирается на объявленные понятия объектно-ориентированной модели системы.

Например, предложение типа «если зона 1 замкнута, то запретить запись данных в ЛОГ» в терминах объектно-ориентированной модели будет выглядеть следующим образом:

OBJ_ZONE,ZONE_ID1,ZONE_CLOSE,OBJ_LOG,OFF,LOG_STATUS
событиереакция

Триада, выделенная синим цветом, соответствует *активному событию* системы. Красным цветом выделена *реакция* системы. Для составления такого (или подобного) предложения используется [Словарь терминов](#) и материал представленный в описании [Объектов](#).

Для перевода макро команды в термины командного интерфейса, следует заменить обозначения полей триад командами с соответствующими цифровыми значениями. Перевод мнемонических имён объектов, событий и реакций в цифровые значения производится согласно таблице в разделе [Таблица значений констант](#).

В завершении, согласно [Таблица значений констант](#), запишем представленный выше пример макро команды в терминах командного интерфейса *Устройства*:

```
&set.macro=1,1,1,1,1,10,5,2
```

■ Напомним, что перед работой (установкой параметров) соответствующая макро команда должна быть создана (см. команду [&add.macro](#)).

Первый параметр команды [&set.macro](#) определяет идентификатор (порядковый номер макроса), а второй параметр – состояние объекта (макрос 1-включен или 0-выключен).

Далее следуют шесть параметров разделённых запятыми и соответствующим двум триадам, *активное событие* (в примере это 1,1,1) и *реакция* (10,5,2). Цифровые взяты из [Таблица значений констант](#).

При необходимости сентенцию ВОПРОС-ОТВЕТ можно усложнить добавлением понятия *Условие*. Например, требуется выполнить подсоединение к IP серверу, если таковое не установлено. В качестве регулярного события можно выбрать событие GPS_RECEIVED объекта OBJ_GPS, которое вырабатывается по умолчанию каждую секунду. Тогда определение макро в терминах объектно-ориентированной модели будет выглядеть так:

```
OBJ_GPS,ANY,GPS_READY,OBJ_GSM,<user_id>,LINK IF( <condition_id> )
```

Где <condition_id> идентификатор объекта *Условие*, который соответствует состоянию требуемого компонента *Устройства*. В этом случае в команде [&set.macro](#) добавляется девятый параметр, идентификатор объекта *Условие*.

Например:

```
&set.macro=1,1,4,255,4,12,2,4,1
```

Тип GSM-соединения (голосовой вызов, CSD или GPRS/IP) зависит от настроек объекта *Пользователь*, индекс которого указан в макро определении (*Пользователь.2* в примере выше).

- Проверку исполнения макро команд следует производить при включенном режиме отладка (см. третий параметр команды [&set.echo](#)).
- В целях отладки создаваемых алгоритмов управления запустить макро команду на исполнение можно, например, с помощью ключевого слова ОКЗ (событие OBJ_SYSTEM.3.CMD_OK) или по внешнему сигналу с дискретного входа (событие OBJ_ZONE.<zone_id>.ZONE_ON).
- Макро команды «исполняются» в порядке следования их номеров (от младшего к старшему). Для ускорения работы с макро командами к идентификатору объекта системного события надо прибавить 128. Это рекомендуется использовать в случае, если данное событие (триада) последнее в списке макросов. Старший бит 128 при этом означает "выход из цикла интерпретации макросов" (break). Например:

```
macro 1,1;1,1,1,1;0,0,0 ; OBJ_ZONE.1.CLOSED
```

```
macro 2,1;129,2,1;0,0,0      ; OBJ_ZONE.2.CLOSED (break)
macro 3,1;1,2,1;0,0,0      ; OBJ_ZONE.2.CLOSED
macro 4,1;1,4,1;0,0,0      ; OBJ_ZONE.3.CLOSED
```

макрос 3 (при событии от замыкания входа 2) работать не будет. Следовательно, данный пример неверен и его следует исправить на:

```
macro 1,1;1,1,1;0,0,0      ; OBJ_ZONE.1.CLOSED
macro 2,1;1,2,1;0,0,0      ; OBJ_ZONE.2.CLOSED
macro 3,1;129,2,1;0,0,0    ; OBJ_ZONE.2.CLOSED (break)
macro 4,1;1,4,1;0,0,0      ; OBJ_ZONE.4.CLOSED
```

3.13.3. Методы оптимизации сценария управления

- 1) исключение "зеркальных" условий за счет использования флага "NOT" в определении макроса.
- 2) исключение условия типа "IP-соединение установлено" при соединении с сервером по реакции 12.<userId>.4, так как такая проверка производится автоматически.
- 3) использование реакции 9.<id>.<action> управления флагом макроса "включен"/"выключен", а также для запуска "соседнего" макроса по событию. Эта возможность позволит создавать "разветвлённые" сценарии управления (см. Руководство администратора).
- 4) в настройках макроса использовать флаг "break" - это позволит сократить время интерпретации событий, а также сократить количество макросов. Иными словами опция "break" может трактоваться как "условный выход из цикла обработки события".

3.14. Объект «Ключевое слово»

Имя объекта	Описание
OBJ_KEY	Используется для создания пользовательских протоколов взаимодействия с внешним контроллерным устройством (например, внешний модем, адаптер связи, интеллектуальный датчик и т.п.)
Идентификатор	Описание
<key_id>	Значение идентификатора объекта определяет порядковый номер ключевого слова из списка ключевых слов.
Событие	Описание
KEY_FOUND	Событие генерируется в случае совпадения принятой строки символов с каким-либо ключевым словом из ранее созданного списка (см. команду &add.key). Строка символов должна оканчиваться символом Enter (код 0Dh) и не совпадать с командами описанными в Главе 4 этого документа.
Реакция	Описание
нет	

3.15. Объект «Счетчик»

Имя объекта	Описание
OBJ_COUNTER	Используется для подсчета событий и генерирования реакций в зависимости от значения объекта. Имеет два пользовательских триггера MIN и MAX. Количество объектов ограничено и зависит от версии <i>Устройства</i> .
Идентификатор	Описание
<key_id>	Определяет порядковый номер объекта.
Событие	Описание
CNT_MIN	Генерируется если значение счетчика достигает значения триггера MIN
CNT_MAX	Генерируется, если значение счетчика достигает значения триггера MAX
Реакция	Описание
CNT_RESET	Сбрасывает значение счетчика в ноль
CNT_INCR	Увеличивает значение счетчика на 1.
CNT_DECR	Уменьшает значение счетчика на 1.
CNT_SETMIN	Устанавливает значение счетчика равным значению триггера MIN.
CNT_SETMAX	Устанавливает значение счетчика равным значению триггера MAX.

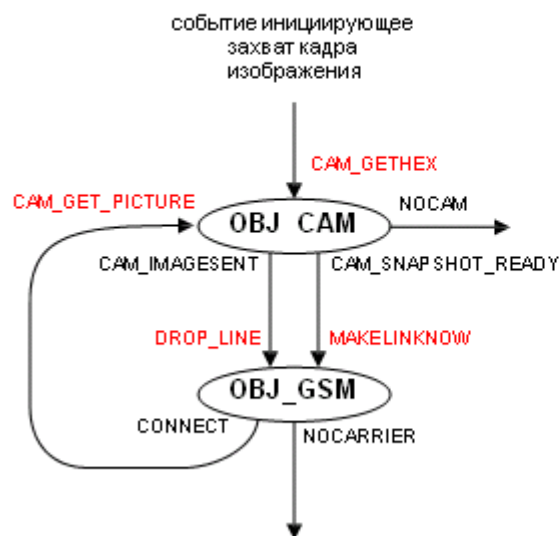
3.16. Объект «Камера»

Имя объекта	Описание
OBJ_CAMERA	Используется для передачи кадра изображения в GSM/GPRS канал связи.
Идентификатор	Описание
<resolution>	Определяет разрешение (resolution) изображения в пикселях по горизонтали и вертикали.
Событие	Описание
CAM_SNAPSHOT_READY	Сообщает о завершении сохранения кадра изображения в буфер <i>Устройства</i> , если отсутствует IP-канал связи с сервером. При использовании дополнительной макро-инструкции, это событие может инициировать соединение с IP-сервером.
CAM_IMAGESENT	Генерируется после по завершении передачи изображения кадра.

CAM_ERROR	Генерируется в случае ошибки передачи образа изображения из камеры в <i>Устройство</i> , при этом <i>Устройство</i> передает текстовое сообщение “NOERR“ в IP-канал, если таковой открыт.
CAM_NOCAM	Генерируется в случае отсутствия соединения с камерой, камера неисправна, камера не включена по питанию или несоответствие скорости TE-порта с настройками последовательного порта камеры. <i>Устройство</i> передает текстовое сообщение “NOCAM“ в IP-канал, если таковой открыт.
Реакция	Описание
CAM_GETBIN	Иницирует захват кадра изображения и передачу в IP-канал изображения в двоичном формате. Разрешение изображения определяется значением идентификатора. IP-канал должен быть предварительно открыт.
CAM_GETHEX	Иницирует захват кадра изображения и передачу в IP-канал изображения в 16-ричном формате. Разрешение изображения определяется значением идентификатора. IP-канал должен быть предварительно открыт.
CAM_GET_PICTURE	Иницирует чтение и передачу кадра изображения из буфера камеры в IP-канал. Чтение буфера производится если с сервером установлено IP- соединение и буфер не пуст.

Два варианта сценариев работы с объектом OBJ_CAM.

Вариант 1. Связь с IP-сервером отсутствует



Вариант 2. Связь с IP-сервером установлена



Здесь красным цветом обозначены управляющие реакции; черным – системное событие от объекта; овалом обозначен сам объект; стрелкам соответствуют макро-инструкции связывающие событие и реакцию.

3.17. Объект «Временная зона»

Имя объекта	Описание
OBJ_TIMEZONE	Используется как генератор одиночного события в заданное время суток.
Идентификатор	Описание
<tzzone_id>	Номер временной зоны.
Событие	Описание
TZONE_EVENT	Генерируется по достижении времени определённого командой &set.tzone .
Реакция	Описание
TZONE_ENABLE	Включает генератор события по времени для временной зоны с номером заданным в идентификаторе.
TZONE_DISABLE	Выключает генератор события по времени для временной зоны с номером заданным в идентификаторе.

3.18. Объект «Пользовательские флаги»

Имя объекта	Описание
OBJ_FLAGS	Используется как элемент памяти событий.
Идентификатор	Описание
<flag_id>	Номер флага (1-16).
Событие	Описание
-	-
Реакция	Описание
FLAG_CLEAR	Сбрасывает значения всех пользовательских флагов в ноль.
FLAG_SET	Устанавливает значение пользовательского флага в единицу. Номер флага указывается значением идентификатора.
FLAG_RESET	Сбрасывает значение пользовательского флага в ноль. Номер флага указывается значением идентификатора.
FLAG_INVERT	Инвертирует значение пользовательского флага. Номер флага указывается значением идентификатора.

3.19. Таблица значений констант

3.19.1. Объекты

Объект – это сущность телематической системы имеющая наименование, свойства определяющие изменения внутренних состояний в зависимости от внешних воздействий и выходные реакции, которые генерируются при изменении внутреннего состояния. Внешним воздействием может быть управляющая реакция поступившая от другого *Объекта*, которая определена *Макро* определением (см. описание объекта [Макро команда](#)).

При определении *Макро* команды допускается использовать значение 255 в поле идентификатора и/или события объекта, которое в поле идентификатора означает «любой идентификатор», а в поле событие - «любое событие». Например: OBJ_ZONE.255.ALARM означает «тревога в любой зоне» или OBJ_ZONE.255.255 означает «любое событие от любой зоны».

Таблица 3.2.1. Значения констант объектов

Объект	Идентификатор	Событие (от объекта)	Реакция (управление объектом)	Значение
OBJ_ZONE				1
	{1 2 4 8}	ZONE_OPEN		0
		ZONE_CLOSED		1
		ZONE_ALARM		32
			ZONE_DISABLE	0
			ZONE_ENABLE	1
			ZONE_SET_ARM	16
			ZONE_SET_ALARM	32
			ZONE_DISARM	48
			ZONE_CONFIRM	64
OBJ_RELAY				2
	{1 2 3 4} 128+<bitmask>		RELAY_ON	1
			RELAY_OFF	0

Объект	Идентифи- катор	Событие (от объекта)	Реакция (управление объектом)	Значение
<u>OBJ_GPS</u>				3
	{0 1 2 255}	GPS_READY4REC		1
	{0 1 2 255}	GPS_READY4ECHO		2
	0 NOW	GPS_OFFLINE		3
	{0 1 2 255}	GPS_RECEIVED		4
	1 SPEED_LESS 2 SPEED_OVER	GPS_SPEED		5
	0	GPS_SPEEDEVENT		6
	0 INVALID 1 VALID_LAST* 2 VALID	GPS_FIXCHANGED		7
	0 POWER OFF 1 POWER ON 2 POWER RESET		GPS_POWER (опция, только в версии со встроенным GPS- приёмником)	1
	<msg_id>		GPS_SEND_MSG	2
	<baud_index>		GPS_SET_BAUD	3
	<pp>		GPS_SET_PINNING	4
<u>OBJ_TIMER</u>				4
	<timer_id> или 255 ANY	TIMER_RUN принимает значение от 1 до N_REPEAT для таймера типа 20h. Для таймера типа 10h событие TIMER_RUN равно всегда 1.		1..N или 1
		TIMER_STOP		0
			TIMER_RUN	1
			TIMER_STOP	0

Объект	Идентифи- катор	Событие (от объекта)	Реакция (управление объектом)	Значение
<u>OBJ SYSTEM</u>				5
	0 COMPLETED	SYS_RUN		1
	0 CMD_UNKNOWN 2 CMD_GELIX 3 CMD_AT 4 CMD_OK	SYS_CMD		2
	1 STRING 2 REPLY 4 ENDPRINT2TE	SYS_PRINT		3
	0..254 (до трёх десятичн. цифр, следующих за OK) или 255 ANY	SYS_CMD_OK		4
	0 TIME_OK	SYS_SYNCTIME		5
	1 LOG_READ	SYS_TIMEOUT		6
	0 RUN	SYS_RESTART		7
	0 (received three +++)	SYS_DATA2GSM		8
	0 ON		SYS_RESTART	1
	FWRD COPY 0 RESET RESET 1 SET RESET 2 RESET SET 3 SET SET		SYS_DATA2GSM см. также FWRD_SYS2GSM и COPY_SYS2GSM табл. 3.3.2. Объект «Условие»	2
	1 ENABLE 0 DISABLE		SYS_TE2GSM_MIRROR	3
	0 FULL SPEED 1 10 MHz 2 1 MHz -- 3 LOW POWER (только Геликс-1)		SYS_SPEED (опция)	4
	0 RUN		SYS_SYNCTIME	5
			SYS_DATA2GSM_Q аналогично SYS_DATA2GSM, но с предварительным сбросом очереди заявок на соединение	12
<u>OBJ GEO</u>				6
	<geo_id>	GEO_IS_OUT		0
		GEO_IS_IN		1
<u>OBJ MSG</u>				7
	<message_id> + бит 80h HIGH_PRIORITY только для PRINT		Послать сообщение пользователю (GSM)	<user_id>
			PRINT	0
			PRINT2TE	128
			PRINT2RS485	129

Объект	Идентификатор	Событие (от объекта)	Реакция (управление объектом)	Значение
OBJ_MACRO				9
	<macro_id>		DISABLE	0
			ENABLE	1
			RUN	2
OBJ_LOG				10
	0 OK	LOG_RESTARTED		1
	0 OK	LOG_NEWREC_OVER		2
	0 INVALID 1 LAST_VALID 2 VALID 3 BAR_CODE 255 ANY	LOG_RECORDED		3
	1 OK	LOG_CONFIRMED		4
	0 нет новых записей 1 новые записи есть	LOG_READCOMPLETED		5
	1 SET_LOG_PERIOD		<period.seconds>	>=5
	0	GPS_SPEEDEVENT		6
	2 SET_LOG_STATUS		REC_ALL	3
			REC_VALID	1
			REC_OFF	2
			REC_DISABLE	4
			REC_ENABLE	5
	3 SET_ECHO_STATUS		ECHO_ALL	3
			ECHO_VALID	1
			ECHO_OFF	2
	4 SET_ECHO_PERIOD		<period.seconds>	>0
	5 WRITE		N/A	0
	6 SET_READ_MODE		Значение бит : 01h NMEA(1) HEX(0) 02h HEX StartStop(1) Auto(0) 04h NEW(1) ALL(0) 08h Don't confirm (1)	0..15
	7 ECHO		N/A	0
	8 READ_LOG		Кол-во записей <N> Ноль возвращает все	{<N> 0}
	9 READ_LOG_REV		Кол-во записей <N> Ноль возвращает все	{<N> 0}
	10 CONFIRM		Кол-во записей <N> или «0»	{<N> 0}
	11 CONFIRM_OK		N/A	0
	12 READ_LAST_VALID_POS			0
	13 MILAGE_CONTROL		MILAGE_RESET	0
			MILAGE_STOP	1
			MILAGE_START	2

Объект	Идентифи- катор	Событие (от объекта)	Реакция (управление объектом)	Значение
<u>OBJ KEY</u>				11
	<key_id> или 255 ANY	KEY_FOUND		1
<u>OBJ GSM</u> (GSM)				12
	1 POWER_OFF 3 IP_LINK_CLOSED 4 CSD LINK ENDED 5 VOICE LINK END	GSM_OK		1
	101 TIMEOUT	GSM_ERROR		2
	<user_id> или 255 ANY	GSM_CONNECT		5
	<user_id> или 255 ANY	GSM_NOCARRIER		6
	0	GSM_INIT_END		7
	0 SIM PIN 1 PIN READY 2 BLOCKED	GSM_SIMPIN		9
	<user_id> или 255 ANY	GSM_SMS_RECEIVED		11
	<user_id> или 255 ANY	GSM_ALERT_VOICE		12
	<user_id> или 255 ANY	GSM_ALERT_DATA		13
	0 Disconnected 2 Connecting 3 Link established 4 Hold 5 Waiting 7 Busy	GSM_CALL_MON отключено в версиях с модемом Q64. См. события: GSM_CONNECT и GSM_NOCARRIER		14
	<user_id> или 255 ANY	GSM_MSG_SENT_OK		15
	<user_id> или 255 ANY	GSM_GPRS_LINKED		16
	2 local network 3 roam 255 ANY	GSM_REGIST		18
	5 TIMEOUT 10 MIN 4 TIMEOUT 20 MIN 3 TIMEOUT 30 MIN 2 TIMEOUT 40 MIN 1 TIMEOUT 50 MIN 0 TIMEOUT 60 MIN	GSM_OFFLINE		19
	<user_id> или 255 ANY	GSM_ERR_IP_LINK		20
	0	GSM_TEMPC		21
	<user_id> или 255 ANY	GSM_ERR_GPRS		22
	<n_ring> или 255 ANY	GSM_RING		23

		0 unknown 1 MTS 2 Beeline 3 Megafon	GSM_COPS		28
		<msg_id>	GSM_USER_AT_SENT		64
		0 DROP LINE 1 COMMAND MODE		DROP_LINE	0
		0		PICKUP_LINE	1
		2 POWER RESET 1 POWER ON 0 POWER OFF		POWER_CTRL	2
		0 SELECT SIM0 1 SELECT SIM1 2 CHANGE_SIM		SIMCHANGE	3
		<user_id>		MAKELINK	4
		0		POWER_RESET	5
		<user_id>		MAKELINKNOW	6
		0		GETBALLANCE	7
		<message_id>		ISSUE_AT_CMD	8
		0		MAKELINKNEXT	9
	<u>OBJ_ADC</u>				13
		<adc_id>	ADC_LESS		1
			ADC_OVER		2
			ADC_MIDDLE		3
	<u>OBJ_CAMERA</u>				16
		0	CAM_SNAPSHOT_READ		0
		0	CAM_IMAGESENT		1
		0	CAM_ERROR		2
		0	CAM_NOCAM		3
		0		GET_PICTURE	0
		1 80x64 2 160x128 3 320x240 (def.) 4 640x480		CAM_GETBIN	1
				CAM_GETHEX	2

Объект	Идентификатор	Событие (от объекта)	Реакция (управление объектом)	Значение
OBJ_COUNTER				17
	<counter_id>	CNT_MIN		1
		CNT_MAX		2
			CNT_RESET	0
			CNT_INCR	1
			CNT_DECR	2
			CNT_SETMIN	3
			CNT_SETMAX	4
OBJ_TIMEZONE				18
	<tzone_id>			1..4
		TZONE_BEGIN		1
		TZONE_END		2
			TZONE_ENABLE	1
OBJ_FLAGS				21
	<flag_id> от 1 до 16		FLAG_RESETBIT	2
			FLAG_SET	1
			FLAG_CLEAR	0
			FLAG_INVERT	3

3.19.2. Условие

Объект *Условие* позволяет применять условные выражения при исполнении макроинструкции. Основные константы объекта *Условие* представлены в таблице ниже.

Таблица 3.3.2. Значения констант объекта «Условие»

Тип объекта	Идентификационный номер или режим	Состояние	Значение
OBJ_ZONE			1
	<zone_id> := {1 2 4 8}	ZONE_ARM	10h
		ZONE_ALARM	20h
		ZONE_DISARM	30h
		ZONE_CLOSED	1
OBJ_RELAY			2
	<relay_id> := {1 2 3 4}	RELAY_ON	1
		RELAY_OFF	0
OBJ_GPS			3
	0	GPS_INVALID	0
		GPS_VALIDLAST	1
		GPS_VALID	2
		GPS_ALL_FILTERS	3
	1 REAL_SPEED	SPEED_LESS	1

	(см. команду &set.sys параметры <es> и <sz>)	SPEED_OVER	2
		SPEED_MIDDLE	4
	2 REAL_SPEED LESS	SOG меньше <value>	<value>
	3 REAL_SPEED OVER	SOG больше <value>	<value>
OBJ_TIMER			4
	<timer_id>	TIMER_RUN	1
		TIMER_STOP	0

Тип объекта	Идентификационный номер или режим	Состояние	Значение
<u>OBJ SYSTEM</u>			5
	1 <i>CHECK_SYSNEWRREC</i>	N/A	0
	2 <i>CHECK_CUSTUMNEWREC</i>	Количество новых записей	0..FFFEh
	3 <i>CHECK_SYSSTATUS</i>	SYS_CMDREADY	1h
		SYSTEM_SYNC_RTC	2h
		SYS_ISSUEIPLINK	4h
		SYS_GPSINIT	8h
		SYS_COLDSTART	10h
		SYS_WATCHDOGSTART	20h
		SYS_MESSAGESEND	40h
		SYS_PSW_PROTECTED	200h
		SYS_PRINTED	800h
		LOG_HEXMODE	1000h
		LOG_HEXNEXT	2000h
		LOG_HEXPREV	4000h
	4 <i>CHECK_SYSTEMPRINT</i>	N/A	0
	5 <i>проверка состояния перенаправления потока данных (GSM)</i>	FWRD_SYS2GSM	10h
		COPY_SYS2GSM	20h
		FWRD_GSM2TE	40h
		FWRD_SMS2TE	80h
	6 <i>проверка состояния интерпритации сообщения с тэгом %RC</i>	NO_WAITING	0
	7 <i>проверка состояния интерпритации сообщения с тэгом %RC</i>	WATING	0
<u>OBJ GEO</u>			6
	<geo_id>	GEO_IN	80h
		GEO_OUT	0
<u>OBJ USER</u>			8
	<user_id>	ACTIVE	0 или 1..0Fh

Тип объекта	Идентификационный номер или режим	Состояние	Значение
<u>OBJ_GSM</u> (GSM)			12
	<код операции сравнения> 0 не строгое (A&B)>0 1 строгое (A&B)==B где А – текущее значение состояния объекта GSM, В – значение параметра «Состояние»	PIN_FALSE	1h
		ROAM	2h
		ONLINE	4h
		POWER_STATUS	8h
		CSD_LINK_ISSUED	10h
		IP_LINK_ISSUED	20h
		VOICE_LINK_ISSUED	40h
		LINKED	80h
		DROP_LINK_ISSUED	100h
		ISSUEPOWERON	200h
		ATCOMMAND	400h
		INIT	800h
		WAIT4ANSWER	1000h
		READY2SEND	2000h
		SMSREAD	4000h
		SENDSMS	8000h
	2 сравнение: текущий баланс меньше значения <balance>	<balance>	0..FFFEh
	3 проверка номера текущей СИМ-карты	GSM_SIM1	1
		GSM_SIM2	2
	4	GSM_RINGING	0
	5 проверка номера оператора (опция)	0 (MTS), 1 (Beeline) или 2 (Megafon)	0, 1 или 2
	6 проверка соединения с IP-сервером	GSM_LINKED	0
	7 проверка состояния GSM модуля (опция)	Signal level GPRS link error GPRS linked IP linked SIM ID Roaming	0700.. 0705 0706 0707 0F0F 1010 2020
<u>OBJ_ADC</u>			13
	<adc_id>	ADC_LESS	1
		ADC_MORE	2
		ADC_MIDDLE	3
ADC_LESS			14
	<adc_id>	<value>	0..3FFh
ADC_MORE			15
	<adc_id>	<value>	0..3FFh
<u>OBJ_COUNTER</u>			16
	<counter_id>	CNT_ZERO	1
		CNT_LESS	2
		CNT_MORE	3
		CNT_MIDLE	4

Тип объекта	Идентификационный номер или режим	Состояние	Значение
<u>OBJ TIMEZONE</u>			18
	<tzzone_id>	TIMEZONE_IN	1
		TIMEZONE_OUT	2
<u>OBJ FLAGS</u>			21
	<код операции сравнения> 0 – сравнение (A&B)>0 1 – сравнение (A&B)=B 2 – сравнение A=B где A текущее значение пользовательских флагов, B – значение <value>	<value>	0..FFFEh

Значения завершающиеся буквой “h” представлены 16-ричным числом.

4. Команды

После включения *Устройство* находится в состоянии трёхсекундного ожидания ввода сервисной команды.

Сервисная команда может быть введена в *TE*-порт в первые несколько секунд после включения питания *Устройства*.

Чтобы произвести сброс ситемных настроек необходимо:

- Подключить *TE* порт *Устройства* к RS232 порту компьютера (115200 8-N-1 управление потоком выключено).
- Запустить терминальную программу на компьютере и открыть соединение с COM-портом, к которому подключено *Устройство*.
- Включить источник питания *Устройства*.
- После того как *Устройство* вернёт слово **Init** и продолжает возвращать точки необходимо нажать три раза клавишу **Escape**. Про этом *Устройство* вернёт слово "default" (включение с настройками *Устройства* по умолчанию). Например:

```
Init.....  
default
```
- Иначе, при включении питания *Устройства* можно нажать и удерживать клавишу **Escape** до тех пор пока *TE* не вернет слово default.

После завершения инициализации *Устройства* терминальная программа вернёт приглашение к диалогу (знак больше) и будет находиться в состоянии ожидания ввода команды, которые описаны в этом разделе.

Включение *Устройства* с настройками по умолчанию способом описанным выше сохраняет настройки во FLASH ранее сохранённые командой &set.save

Полный перечень сервисных команд:

- | | |
|--------|--|
| Escape | включение <i>Устройства</i> с настройками по умолчанию |
| D | включение режима «отладка» |
| G | сброс всех настроек и удаление данных ЛОГ (форматирование FLASH) |

4. &-команды

4.1. Правила построения &-команд

Команда встроенного интерпретатора системы начинается с символа «&» (ampersand). Исключением является системная команда «?» (Возврат информации о системе) и АТ-команды, передаваемые на исполнение GSM/GPRS-модулю.

Структура &-команды:

<класс>.<пароль>.<тип>=<параметр1>,<параметр2>...[/s]

Если системный пароль не установлен администратором (см. команду [&set.psw](#)), то поле <пароль> опускается и &-команда выглядит так:

<класс>.<тип>=<параметр1>,<параметр2>...[/s]

Для успешной интерпретации команд всегда необходимо указывать «класс» и «тип» &-команд. После знака равно указываются параметры команды, назначение которых определяется для каждой команды индивидуально. Некоторые &-команды не требуют указания параметров.

Классы &-команд:

- Установить параметр системы или объекта (&set)
- Запросить данные (&req)
- Вернуть настройки объекта (&get)
- Создать или добавить объект (&add)
- Удалить объект (&del)

Интерпретатор &-команд может обрабатывать «короткие» команды, в которых класс команды задаётся минимум двумя буквами, а тип – одной. Например:

&re.l=10	равно &req.last=10
&se.e	равно &set.echo
&se.sa	равно &set.save

Как правило, для определения класса команды достаточно двух букв. Для определения типа команды достаточно одной или двух букв.

При успешном исполнении &-команды *Устройство* возвращает **OK**, а в случае ошибки – **ERROR**. Если администратором был определён пароль, который не был указан в &-команде или пароль указан неверно, то *Устройство* возвращает **PSW ERR**.

- Если &-команда завершается ключом “/s”, то после её исполнения произведённые изменения автоматически сохраняются (аналогично исполнению команды [&set.save](#)). Ключ “/s” рекомендуется использовать в случае исполнения одиночных команд передаваемых по каналу SMS или GPRS.

- 16-ричный параметр &-команды должен иметь длину кратную двум.
- Команда не выполняется и остается в очереди на исполнение, пока *Устройство* не завершит возврат запрошенных данных в 16-ричном режиме или возврат таких данных не будет прерван (см. команды [&req](#) с параметром или Н или Х).
- Для передачи на исполнение контроллеру команда должна завершаться символом Enter (коды 0Dh и 0Ah).
- Последовательность символов, принятая по командному интерфейсу (*TE* или GSM/GPRS) и совпадающая с любым зарегистрированным словом представленным в Приложении 8 этого документа может нарушить логику работы *Устройства*.

В зависимости от версии прошивки некоторые &-команды могут быть отключены. Перечень поддерживаемых команд согласовывается с заказчиком на этапе отладки проекта.

4.2. Общие команды

4.2.1. Выдать информацию о системе

Команда ? (знак вопроса) или [&help](#) возвращает информацию об устройстве.

Например:

> ?	имя <i>Устройства</i> и номер версии <i>firmware</i>
GELIX-1 v3.724	доступные системные ресурсы
/8IO/GPS/RTC/WD/upgr	идентификатор <i>Устройства</i> (см. команду &set.id)
id 00001001	пароль (не установлен в этом примере)
psw no	режим записи данных в <i>журнал событий</i> (см. команду &set.log)
log x60,off,0,4,0,20,2,0,60	настройки вывода <i>TE</i> (см. команду &set.echo)
echo x5,all,17	номер последней записи ЛОГ
last id 20036	размер структур записи или пакета данных (байт)
size 38,57	информационная ёмкость ЛОГ (количество записей)
flash 446976	размер свободной памяти (байт)
free heap 1635	скорость обмена по последовательному порту с GPS-приемником, интервал выдачи NMEA предложений, маска форматов возвращаемых NMEA-предложений (см. команду &set.nmea)
gps 4800,poll 1s,06	тоже, но для версии <i>Устройства</i> с поддержкой внешнего навигационного приёмника ГЛОНАСС/GPS (опция, &set.nmea)
glo 4800,poll 1s,06	состояние GSM-модуля (см. &get.gsm)
gsm pwrON,pinSET,lnLOC,ipNO,-	количество объектов типа <i>Пользователь</i> (&add.user)
users 0	количество объектов <i>Сообщение</i> (&add.message) и
msg 0,0	количество сообщений в очереди на отправление
macros 0	количество объектов типа <i>Макро</i> (&add.macro)
timers 0	количество объектов типа <i>Таймер</i> (&add.timer)
conditions 0	количество объектов типа <i>Условие</i> (&add.condition)
key 0	количество объектов <i>Ключевое слово</i> (&add.key)
counter 4x16b	количество программных счетчиков и их разрядность (опция)
timezone 4	количество объектов <i>Временная зона</i> (опция)
speedzone 4	количество объектов <i>Зона скорости</i> (опция)
register W,2,16	количество объектов <i>Пользовательский Регистр</i> , опция (&add.log)
>	

Опции прошивки могут дополнять или исключать друг-друга. Опции включаются в релиз прошивки при согласовании проекта с разработчиком системы мониторинга. Вернуть ключ поддерживаемых опций можно по команде [&get.system](#) в значении ключа option.

Системные ресурсы, поддерживаемые версией *firmware*, выводятся во второй строке и обозначают следующее:

/8IO	поддержка восьми входов/выходов (4 входа + 4 выхода)
/4A	поддержка четырёх аналоговых входов
/TS	поддержка датчика температуры Gelix-TS (опция)
/GPS	поддержка GPS-приёмника (GPS версия)
/GSM	GSM/GPRS-версия, SonyEricsson GR47
/GR64	GSM/GPRS-версия, Wavecom GR64
/2SIM	поддержка второй SIM-карты (опция)
/RTC	поддержка таймера реального времени (RTC)
/485	поддержка интерфейса RS485 и соотв. Протоколов (опция)
/WD	системный watchdog включен
/CAM	поддержка цифровой фото камеры Gelix-CAM (опция)
/IZOT	поддержка таксометра ИЗОТ (опция)
/dll	поддержка динамически загружаемых программных модулей (опция)
/upgr	поддержка режима обновления <i>firmware</i> по каналу связи <i>TE</i> и GSM (если доступен)

Возможные значения состояния встроенного GSM-модуля:

pwr { ON OFF }	GSM-модуль «включен» или «выключен».
pin { ERR SET NO }	состояние защиты SIM паролем (PIN), «PIN определен и неверен», «PIN определен и верен», «PIN не определен».
In { LOC ROA }	состояние GSM, «локальный оператор» или «в зоне роуминга».
ip { NO YES }	состояние IP-канала, соединение с сервером «не установлено» или «установлено».
GPRS	состояние GPRS-канала «открыт» или «в сети Интернет».
x.xx	значение текущего баланса. Если запрос на состояние баланса отключен возвращается “-“ (знак «минус»).

4.2.2. AT-команды

Командный интерфейс *TE* позволяет производить оперативное управление встроенным в *Устройство* GSM/GPRS-модулем с помощью AT-команд.

При исполнении AT-команды может быть нарушен ранее созданный и отлаженный процесс взаимодействия микроконтроллера со встроенным GSM/GPRS-модулем.

AT-команда должна начинаться символом ^, который определяет принадлежность вводимой информации к классу AT-команд. Вся последовательность букв, знаков и цифр (максимум 80) следующих за символом ^ будет передана на исполнение GSM/GPRS-модулю после ввода символа Enter.

Пользовательская АТ-команда передается на исполнение GSM/GPRS-модулю только при включенном режиме «Отладка» (см. параметр `<mode>` команды [&set.echo](#)).

Результат исполнения АТ-команды возвращается в порт *TE*.

Например:

```
> ^AT+CGDONT?
+CGDCONT: 1,"IP","internet.mts.ru","",0,0
OK
>
```

Как правило, необходимостью ввода пользовательских АТ-команд является настройка аудио тракта GSM-модуля.

АТ-команды не исполняются, если они были переданы по каналу GPRS или CSD, а также если с выключенным режимом «Отладка».

АТ-команда может не исполниться, если GSM/GPRS-модуль занят или установлена связь по голосовому каналу или открыта IP-сессия. При этом *TE* вернёт сообщение BUSY

4.2.3. &set.[<psw>].id=[<system_id>][,<config_id>]

Первый параметр команды устанавливает идентификационный номер *Устройства* (10-тичное число от 0 до 99999999). Второй – идентификационный номер пользовательских настроек (16-ричное число от 0 до FFFF).

Например:

```
> &set.id=12086,0000
OK
>
```

Идентификатор пользовательских настроек не влияет на логику работы *Устройства* и может быть возвращен командой `&get.id`.

4.2.4. &get.[<psw>].id

Команда возвращает идентификационный номер *Устройства*.

Например:

```
> &get.id
id 12086,0000
OK
>
```

4.2.5. &get.[<psw>.]version[=all|option]

Команда возвращает номер текущей версии firmware и/или код индикаторов [опций](#).

Например:

```
> &get.ver=all
GELIX-2 v4.774 (build 4870)
opt Q18-0284F807
OK
>
```

4.2.6. &set.[<psw>.]psw="<new_psw>"

Команда устанавливает (или удаляет) пароль необходимый для защиты исполнения пользовательских &-команд. Произведённые настройки сохраняются в системных настройках *Устройства*.

<psw> старый пароль. Если пароль не установлен, то поле «старый пароль» не указывается.

<new_psw> новый пароль в кавычках. Максимум 8 символов.

Пример 1. Пароль не установлен, установить пароль:

```
> &set.psw="1234"
OK
>
```

Пример 2. Пароль установлен (1234), изменить пароль:

```
> &set.1234.psw="qwer"
OK
>
```

Пример 3. Пароль установлен (1234), удалить пароль:

```
> &set.1234.psw=""
OK
>
```

4.2.7. &set.[<psw>.]sys=[<gm>],[<sp>],[<es>],[<nr>],[<pr>],[<dl>],[<mm>],[<sz>],[<ts>],[<dz>],[<bc>]

Команда устанавливает значения системных настроек.

<gm> 16-ричное значение, младшие два бита которого управляют режимом работы GSM-модуля:

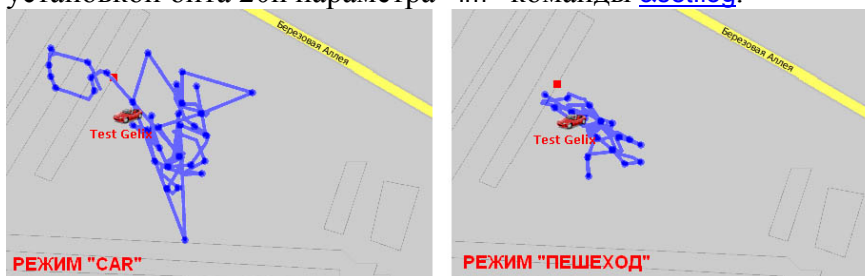
01 - определяет состояние «включено» GSM/GPRS-модуля при включении питания *Устройства*. Если значение равно «1», то при включении устройства GSM/GPRS-модуль будет включен и инициализирован (по умолчанию параметр равен 1). Если значение равно «0», то GSM/GPRS-модуль не включится. Дальнейшее управление

состоянием «включено» можно производить с помощью макроса или битом 04 (см. ниже);

02 - включает встроенный механизм мониторинга состояния GSM/GPRS-модуля, при котором производится регулярная проверка работоспособности GSM/GPRS-модуля и его перезапуск (reset modem), если модуль не отвечает на AT- команды или его состояние не соответствует требуемому. По умолчанию режим отключен.

04 - позволяет оперативно управлять состоянием GSM/GPRS- модуля включено/выключено, которое определяется значением младшего бита. Например, команда &set.sys=4 выключит GSM- модуль, а команда &set.sys=5 включит его.

- <sp>** минимальная скорость (SOG) транспортного средства, ниже которой запись в *ЛОГ* не производится (по умолчанию значение равно 0). Аналогично параметру <sp> команды [&set.log](#).
- <es>** пороговое значение скорости (SOG) транспортного средства для управления режимом навигационного приёмника «автомобиль/пешеход» (только для версии GPS-приёмника типа VENUS) и генерирования системных событий GPS_SPEED. Включение режима производится установкой бита 20h параметра <lm> команды [&set.log](#).



Если действительная скорость ТС превышает пороговое значение <es>+<sz> (см. описание <sz> ниже), то производится переключение навигационного приёмника в режим «автомобиль» и *Устройство* генерирует системное событие OBJ_GPS(3).SPEED_OVER(2).GPS_SPEED(5). Если действительная скорость транспортного средства становится меньше значения <es>, то навигационный модуль переключается в режим «пешеход» и *Устройство* генерирует системное событие OBJ_GPS(3).SPEED_LESS(1).GPS_SPEED(5). Указанные события генерируются только один раз после выполнения соответствующего условия. Указанные события не генерируются, если включена поддержка работы с объектом OBJ_SPEEDZONE.

- <nr>** пороговое количество «новых» записей, при превышении которого *Устройство* вырабатывает системное событие OBJ_LOG.0.NEWREC_OVER. Событие не генерируется, если значение параметра <nr> равно нулю.
- <pr>** количество записей возвращаемых в одном пакете данных в режиме «ожидания подтверждения» (см. команды [®](#) режим Нех, тэг [%RC](#) и реакцию [OBJ_LOG.READ_LOG.0](#)).
- <dl>** задержка вывода данных в порт TE (мсек, по умолчанию 0). Может быть необходима для повышения надёжности передачи данных по каналам IrDA, CSD, GPRS и Bluetooth.
- <mm>** 16-ричное значение двух индексов объекта *Сообщение*. Младшие 4 бита соответствуют номеру сообщения, которое возвращается в порт *TE*

после включения питания *Устройства* (например, с целью инициализации внешнего контроллера).

Старшие 4 бита соответствуют номеру сообщения, которое возвращается в порт *TE* перед перезапуском *Устройства* (см. описание [команд](#) работы с *Сообщениями*). При определении параметр должен иметь две 16-ричные цифры. Например:

[&set.system](#) = ,,,,,,01

✎ При подключении внешнего контроллера к TE-порту *Устройства* необходимо отключить режимы «эхо» и «отладка» (см. команду [&set.echo](#)).

<sz> зона скорости (см. параметр **<es>**) для события OBJ_GPS(3).SPEED_OVER(2).GPS_SPEED(5). Параметр не работает, если включена поддержка работы с объектом OBJ_SPEEDZONE или если сброшен бит 20h параметра **<lm>** команды [&set.log](#).

<ts> тип подключенного датчика температуры (1, 2, 3 или 4), опция:

1 – для датчика типа DS18B201;

2 – для датчика типа DS18S203;

3 – для датчика типа DS1820;

4 – для устройства считывания кода iButton и eMarine.

<dz> управление частотой опроса состояний дискретных входов с разрешением 1 мс. Если значение равно «0» (по умолчанию), то задержки опроса нет и опрос состояний входов производится на максимальной частоте. Максимальное значение параметра равно 254. Если значение равно 255, то при включении *Устройства* значение параметра сбрасывается в значение по умолчанию равное 2.

<bc> управление режимом поддержки сканера штрих-кода (Bar Code reader). Значение параметра равное 1 включает поддержку сканера штрих-кода. Значение 0 – выключает. Сканер штрих-кода должен возвращать последовательность байт завершающуюся кодами <0Dh> и <0Ah>. При этом *Устройство* полученную строку пытается интерпретировать как &-команду, потом как ключевое слово и, если нет совпадений, то производит запись следующей структуры в ЛОГ:

<record_id> <time> <date> <0xFF> <bar_code> <0x00>

при этом генерируется событие OBJ_LOG.3.RECORDED(3). Например:

&set.sys = ,,,,,,1

Например, требуется чтобы после включения питания *Устройство* возвращало в TE-порт команду инициализации внешнему контроллеру (согласно шаблона сообщения №1) подключенному к TE-порту:

> &set.sys = ,,,,,,01

OK

>

Пример 2, требуется выключить питание GSM-модуля:

> &set.sys = 4

OK

>

Пример 3, требуется включить питание GSM-модуля:

> &set.sys = 7

OK

>

4.2.8. &set.[<psw>].lock={Scenario|All|Unlock}

Команда устанавливает или снимает защиту командного интерфейса. Значение первого параметра является литерой S, A или U (от первой буквы слова Scenario, All или UnlockAll, соответственно):

Scenario блокировка &-команд изменяющих пользовательский сценарий управления Устройством;

All блокировка исполнения всех &-команд;

Unlock снятие блокировки всех &-команд.

Например, требуется запретить исполнение команд &set, &get, &add и &del для объектов Макро, Таймер и Условие:

```
> &set.lock=S
OK
>
```

Пример 2, запретить исполнение всех &-команд:

```
> &set.lock=A
OK
>
```

Пример 3, разрешить исполнение всех &-команд:

```
> &set.lock=Unlock
OK
>
```

4.2.9. &set.[<psw>].save

Команда сохраняет ранее произведённые системные настройки (см. команды &add, &set и &del).

Например:

```
> &set.save
OK
>
```

Запись конфигурации должна производиться с целью сохранения настроек в следующей сессии работы системы (после отключения от цепи питания). Сохранение настроек системы производится в сектор данных встроенной памяти FLASH (см. [Приложение 2, Карта FLASH и настройки по умолчанию](#)).

✓ Некоторые &-команды при исполнении не требуют последующего сохранения изменений командой &set.save, так как исполнение таких команд сопровождается непосредственной записью новых установок во FLASH. К таким &-командам относятся команды: [&set.psw](#) (установить/снять системный пароль), [&set.gsm](#) (только первый параметр, изменение SIM PIN), работа с шаблонами сообщений и работа с ключевыми словами.

4.2.10. &get.[<psw>.]system[=cfg]

Команда без параметра возвращает состояния модулей и внутренних флагов *Устройства*. Например:

```
> &get.system
fwr 3334
msg 18.11.2010
sys 3,0,0,2,5,0,02,4,0,100 (0800)
fla 0000
opt 1C00F8C7
gsm 06,1024,00 pw pi dcd
gps 5
led 222
OK
>
```

где

- fwr** версия файла прошивки (firmware). Возвращается в версиях прошивки до июня 2011.
- msg** дата создания файла системных сообщений.
- sys** текущие настройки произведенные командой [&set.system](#)
- fla** текущие значения пользовательских флагов.
- opt** битовая маска подключенных [опций прошивки](#). Начиная с июня 2011 года форма представления опций прошивки дополнена префиксом определяющим тип оборудования Устройства “<GSM><GPS><FLASH>”.
- gsm** дополнительная информация работы модуля GSM: первое 16-ричное число соответствует настройкам второго параметра команды [&set.gsm](#); второе 16-ричное число отображает состояние работы GSM/GPRS-модуля в соответствии с Таблицей [Значения констант объекта Условие](#); Биты третьего 16-ричного числа соответствуют текущим режимам перенаправления потока данных, устанавливаемым реакциями SYS_DATA2GSM и SYS_TE2GSM_MIRROR объекта OBJ_SYSTEM; pw (power pin on); pi (power indicator); dcd (pin status); ri (ring status).
- gps** флаг состояния навигационных данных и включения устройства по битам: 8h - GPS_VALID, 4h – GPS_VALID_LAST, 2h – SYSTEM_WACHDOG_RESTART, 1h – SYSTEM_RESTART.
- led** соответствует коду светового индикатора (состояния работы GSM, записи в ЛОГ и GPS соответственно, см. параграф «LED индикатор» в Книге I. Руководство по эксплуатации).

Если при исполнении в команде указан параметр “**cfg**” (или одна буква “**c**”), то *Устройство* вернёт *TE* образ блока конфигурации сохранённой во FLASH. Вывод данных образа блока конфигурации производится в 16-ричном представлении.

Перед исполнением команды &get.syste=cfg рекомендуется включить режим «Приём в файл» или «Capture text» терминальной программы, обеспечив при этом

сохранение принятых данных в файл. Размер принятого файла составляет 10 Кбайт. Сохраненный образ конфигурации может быть использован для быстрой настройки иных приборов в процессе производства/клонирования (см. [Приложение 4. Обновление Firmware](#)).

Пример ниже содержит фрагмент образа конфигурации (или пакетного файла настроек):

```
> &get.sys=cfg
>>>>C3233FFFFFFFFFFFFFFFFF000000FFFE0B00000001000007D0001E000000030ACA00
05060114B11500350200000000FFFF85FFFFFFFFFFFFFFFFF00F и т.п.
```

❓ Не допускайте изменений содержимого файла настроек.

📖 Файл настроек не несёт информации об идентификаторе *Устройства* (команда [&set.id](#)) и SIM PIN (первый параметр команды [&set.gsm](#)) в GSM версии *Устройства*, т.е. при обновлении системных настроек с помощью пакетного файла настроек идентификатор *Устройства* и SIM PIN не изменяются.

📖 Если установлен системный пароль (см. [&set.psw](#)), то при получении образа конфигурации *Устройство* принимающее пакетный файл проверяет на совпадение свой системный пароль с системным паролем в принятом пакетном файле и, в случае его совпадения, производит обновление системных настроек согласно принятым.

📖 Файл настроек можно послать Устройству в порт TE или открытый IP-канал.

4.2.11. &restart[.<psw>]

Команда производит перезапуск *Устройства*.

Например:

```
> &res
restart...
```

❓ В случае исполнения команды [&restart](#) в состоянии активного IP-соединения Устройство предварительно закрывает соединение с IP- сервером.

Пример исполнения команды [&restart](#) при включенном режиме «отладка» (см. комментарии справа, [debug] – отладочное сообщение):

```
> &res
-DTR low

OK

restart...

OK
*E2IPC
```

[debug] переключиться в командный режим
[debug] переключение в командный режим завершено

ответ на исполнение команды [&restart](#)
[debug] закрыть IP-стек

```
*E2IPC: 000
OK
```

```
.....
```

```
[debug] модем завершил сеанс IP-связи
[debug] исполнение уоманды *E2IPC
завершено успешно
выключение GSM-модуля и перезапуск
Устройства
```

Пример исполнения команды [&restart](#) при выключенном режиме «отладка» (см. комментарии справа):

```
> &res
restart...
OK
.....
```

```
ответ на исполнение команды &restart
выключение GSM-модуля и перезапуск
Устройства
```

4.3. Работа с дискретными входами

Настройками *Устройства* (команда [&set.zone](#)) можно определить требуемый режим работы любого дискретного входа:

1. Сигнальный режим – в этом случае изменение состояния сигнала на входе генерирует системное Событие;
2. *Зона охраны* – датчики подключенные параллельно цепи дискретного входа образуют *Зону охраны*. Дополнительные настройки *Зоны охраны* «нормально замкнута» и «поставлена на охрану» определяют условие, при котором может генерироваться системное Событие «тревога». Подтверждение состояния «тревога» и его сброс производятся по Событию;
3. Счётный режим – режим посчёта «+1» импульсов (сигналов) на дискретном входе. Значение счётчика хранится в пользовательском регистре, нумерация которого соответствует порядковому номеру входа.

Поддержка режима Decrement «-1» счётчика является опцией (см. [Приложение 9. Индикатор опций прошивки](#)). При этом команда [&set.zone](#) заменена на команду [&set.cnt](#).

4.3.1. [&set.\[<psw>\].zone=<mask>,<mode>,\[<arm>\]\[,<NC>\]](#)

Управляет состояниями дискретных входов, *Зон охраны*.

<mask> - определяет идентификатор *Зоны*, управляемый следующими параметрами команды. Идентификатор *Зоны* определяется одним 16-ричным числом от 00 до 0F, биты которого соответствуют номерам входов (младший бит – вход 1, старший бит – вход 4). Например, маска входов 1 и 2 имеет значение 03.

<mode> - определяет состояние входа и может иметь одно из следующих значений:

- 1 включить *Зону*. В этом случае изменение состояния на входе (или входах, см. параметр **<mask>**) будет генерировать

- соответствующее системное событие (ZONE_ON или ZONE_OFF).
- 0 выключить Зону определённую параметром <mask>. В этом режиме события объекта OBJ_ZONE не генерируются и входы сигналы могут изменять состояние соответствующего счётчика (см. раздел 4.20. [Работа с программными счётчиками](#)).
 - 2 подтвердить сигнал «тревога» в Зоне (или Зонах), соответствующей значению «единица» в параметре <mask>.

✓ Объект OBJ_ZONE (дискретный вход) не генерирует системных событий, если вход отключен (disabled). При этом соответствующий пользовательский регистр работает в «счётном» режиме (см. также игнорирование *эффектадребезга контактов*, команда [&set.system](#)).

<arm> - управляет режимом Зоны охраны (Зона должна быть включена, см. параметр <mode>):

- 1 поставить Зону на охрану. В этом состоянии Зона может генерировать событие ZONE_ALARM, если состояние не входе отлично от нормального.
- 0 снять Зону с охраны.

<NC> - задаёт нормальное состояние Зоны («normal open» или «normal closed», Зона должна быть включена, см. параметр <mode>):

- 1 нормально замкнуто (normal closed, NC)
- 0 нормально разомкнуто (normal open, NO)

Проверить настроечные параметры и состояния Зон можно командой **&get.zone**.

Например, поставить Зону 1 на охрану с нормальным состоянием NO (normal open):

```
> &set.zone=01,,1,0
OK
>
```

4.3.2. &get.[<psw>].zone

Команда возвращает состояние Зон в формате:

```
zone <enabled>,<armed>,<NC>;<alarmed>;<status>
```

где первый параметр есть десятичное число, остальные – шестнадцатеричные в позиционном битовом представлении Зон в порядке веса (младший разряд – Зона 1, старший – Зона 4):

- <enabled>** - Зона включена (1) или выключена (0);
- <armed>** - Зона поставлена на охрану (1) или (снята (0);
- <NC>** - нормальное состояние замкнуто (1) или разомкнуто (0);
- <alarmed>** - состояние «тревога» (1) или нет тревоги (0);
- <status>** - текущее состояние Зоны, замкнуто (1) или разомкнуто (0).

Например:

```
> &get.zone  
zone 0F,00,00;00;00  
>
```


В этом примере первый параметр ответа возвращает состояние всех зон «включено» и далее – «снята с охраны», «нормальное состояние разомкнуто», «нет тревоги» и «разомкнуто» (нет сигнала).

4.4. Работа с дискретными выходами

Термин «работа с дискретным выходом» подразумевает работу с подключенной нагрузкой. Такое допущение уместно, так как состояние дискретного выхода *Устройства* определяет состояние реальной цепи на выводах (выходах) *Устройства*, а также подключенной нагрузки (лампа, электромагнитное реле и т.п.).

4.4.1. &set.[<psw>].relay=<status>[,<mask>][,<type>]

Команда управляет текущими состояниями выходов («замкнуть» или «разомкнуть»), а также устанавливает нормальное состояние выходов («замкнуто» или «разомкнуто»).

 Нормальное – это такое состояние дискретного выхода, которое *Устройство* должно установить после включения питания.

Команда [&set.relay](#) может изменять состояние одного и одновременно нескольких выходов в зависимости от значения первого параметра <status>.

status параметр «состояние» является обязательным и определяет новое состояние выхода (см. третий параметр «тип»). Задаётся 16-ричным числом, каждый бит которого соответствует выходу (младший бит – младший номер выхода). Одна 16-ричная цифра адресует четыре выхода. Значение бита «1» включает выход (цепь замкнута), значение «0» - выключает выход (цепь разомкнута).

mask параметр «маска» является необязательным. Маска используется для выборочного управления состояниями выходов *Устройства*. Задаётся 16-ричным числом, каждый бит которого соответствует выходу (младший бит – младший номер выхода) Если значение бита **mask** равно «00», то состояние выхода не изменяется. По умолчанию (если параметр **mask** не задан) значение маски равно «FF».

type параметр «тип» позволяет указать, какое состояние выхода подлежит изменению: «текущее» (1), «нормальное» (2) или сразу оба (3). Значение параметра «тип» по умолчанию равно «1».

Примеры:

> &set.relay=08	включить выход 4, остальные -
выключить	
ОК	
> &set.relay=08,08	включить только выход 4
ОК	
> &set.relay=02,,1	включить выход 2, остальные -
выключить	
ОК	
> &set.relay=03,,2	установить нормальное состояние
«замкнуто» для выходов 1 и 3	
ОК	
>	

4.4.2. &get.[<psw>].relay

Возвращает нормальное и текущее состояния дискретных выходов *Устройства* в формате:

`relay <нормальное состояние>,<текущее состояние>`

Состояние выхода отображается 16-ричным числом, значения битов которого определяют состояние соответствующего выхода (младший бит – младший выход).

Например:

```
> &get.relay relay 05,00
ОК
>
```

4.5. Работа с датой и временем

Устройство, оборудованное таймером реального времени (RTC), поддерживает две дополнительные команды &get.date и &set.date необходимые для возврата и установки даты и времени соответственно.

При отключенном питании более чем на 7-10 дней системные дата и время могут быть сброшены.

- ✓ Режим синхронизации с глобальным временем. После включения питания *Устройства* системные дата и время автоматически устанавливаются согласно глобальному (GMT) времени при поступлении первой валидной координаты.

4.5.1. &get.[<psw>].date

Возвращает дату и время часов реального времени в формате:

`DDMMYY, HHMMSS`

где первое число соответствует дате в порядке следования день (DD), месяц (MM) и год (YY). Второе число – часы (HH), минуты (MM) и секунды (SS).

4.5.2. **&set.[<psw>].date=<DDMMYY>,<HHMMSS>**

Устанавливает дату и время таймера реального времени (RTC). При исполнении команда удаляет текущие дату и время и записывает новые значения, которые сохраняются автоматически. Данные будут сохранены до тех пор пока достаточно энергии во встроенной в *Устройство* батарее питания для таймера реального времени.

Первый параметр команды **&set.date** задаёт дату в порядке:

- DD - день
- MM - месяц
- YY – год

Второй параметр задаёт время в порядке:

- HH - часы
- MM - минуты
- SS – секунды

Например:

```
> &set.date=221204,184556      установить дату и время
ОК
>
```

ВНИМАНИЕ! При вводе значений даты и времени *Устройство* не производит проверку достоверности данных. После исполнения команды **&set.date** рекомендуется проверить произведённые установки командой **&get.date**.

- Системные дата и время автоматически синхронизируются с глобальными каждый раз после включения *Устройства*.

4.6. Работа с журналом событий

Несколько правил исполняемых *Устройством* в отношении записи данных в журнал событий (LOG) и возврата данных:

3. Запись данных в журнал событий (LOG) производится в порядке поступления, т.е. каждая "новая" запись может добавляться в LOG

- регулярно (согласно настройкам произведёнными командой [&set.log](#)) или по событию согласно управляющей реакции объекта [OBJ LOG](#).
4. Каждая запись имеет свой порядковый номер, который увеличивается на 1.
 5. Запись LOG производится по кругу, т.е. при заполнении LOG удаляется самая старая запись.
 6. Реализовано два метода возврата данных: «прямой» (в порядке уменьшения порядковых номеров записей) и «реверсивный» (обратный прямому). Оба метода доступны через командный интерфейс (команды [&req](#) с ключом rev или без него), через *макро команды* (реакции объекта [OBJ LOG](#)), а также через тэги входящие в состав шаблона сообщения (%RC, %NC и т.п.).
 7. Автоматический возврат данных (в TE-порт, IP- или CSD-канал или SMS) определяется ТОЛЬКО МАКРОКОМАНДАМИ и частично настройками [&set.echo](#), в случае если включен режим перенаправления потока данных.

Команды запроса (request) данных из *журнала событий* позволяют вернуть данные записанные в LOG по различным критериям:

- Вернуть запись, соответствующую запуску *Устройства* ([&req.first](#)).
- Вернуть <N> последних записей или все записи ([&req.last](#)).
- Вернуть записи из указанного диапазона ([&req.range](#)).
- Вернуть записи по дате и времени ([&req.time](#)).
- Вернуть *новые записи* ([&req.new](#)).
- Сбросить флаг *новая запись* ([&req.confirm](#)).

Параметры команд могут определять режим возврата данных:

Пакетный режим вывода данных – это режим, в котором при чтении *LOG* данные возвращаются последовательно до тех пор, пока все востребованные записи не будут прочитаны или процесс чтения/вывода не будет прерван пользователем (Escape).

Старт-стопный режим – это режим возврата предопределённой порции данных (одной или нескольких записей) с последующим ожиданием ответа-подтверждения получения этой порции данных от получателя. При этом пользователь может управлять процессом возврата данных, используя следующие коды:

Enter подтверждение приёма и возврат следующего пакета записей, если таковой содержится в ЛОГ. Если возвращенная запись является последней из списка запрошенных, то *TE* возвращает **ОК**. Иначе *Устройство* будет ожидать следующей команды от пользователя.

Пробел *TE* повторяет возврат предыдущей записи.

Escape прервать процесс возврата данных. При этом *TE* вернёт **End**

✍ Режим ожидания ответа-подтверждения устанавливается по

окончании возврата пакета данных.

- режим ожидания ответа-подтверждения сбрасывается в любом из нижеперечисленных случаев:
 - получение ответа-подтверждения от сервера или исполнения реакции CONFIRM;
 - окончании времени ожидания ответа-подтверждения (см. [&set.system](#)). В этом случае ТЕ возвращает сообщение TIMEOUT и генерируется событие [OBJ_SYSTEM\(5\).\(1\).TIMEOUT\(6\)](#)
 - окончании сеанса связи с сервером (NO CARRIER или по событию)
 - выключении питания GSM модуля
 - программном рестарте устройства в том числе и по команде Watchdog
 - исполнении пользователем команды [&req](#)
 - инициировании передачи нового пакета данных

Формат данных в старт-стопном режиме представлен в документе [4].

Возвращаемые записи могут следовать прямому и обратному порядку. В прямом порядке следования записи возвращаются по убыванию порядковых номеров. В обратном – с возрастанием порядковых номеров.

Формат возвращаемых данных определяется настройками, произведенными командой [&set.echo](#).

- Порядок следования форматов данных: **PGIO, GPGGA, GPRMC**.
-

4.6.1. [&req.<psw>.\[confirm\[=<N>\]](#)

Устанавливает флаг «прочитано» для записи(ей) данных в *ЛОГ*. Параметр <N> указывает количество записей *ЛОГ*, в которых необходимо установить флаг «прочитано». Если параметр <N> не указан или равен нулю, то флаг «прочитано» устанавливается для всех записей *ЛОГ*.

Команду [&req.confirm](#) рекомендуется использовать после исполнения команды [&req.new](#) с равным значением параметра.

Например:

```
&req.new=10
&req.confirm=10
```

Заметим, что исполнение команды [&req.confirm](#) после исполнения, например, команды [&req.last](#) также произведёт установку флага «прочитано» для количества записей указанного в параметре этой команды. Однако этот вариант использовать не рекомендуется.

Команда возвращает ОК в случае успешного выполнения или End, если достигнут флаг «конец ЛОГ» или «новых» записей в *ЛОГ* не существует.

Например:

```
> &req.confirm
End
>
```

4.6.2. &req.[<psw>].first=[f|c|w|n][,x]

Возвращает первую запись из ЛОГ согласно параметра, определяющего критерий поиска первой записи:

- f** первая запись из *журнала событий* (запись с наименьшим порядковым номером).
- c** запись последнего включения (cold start).
- w** запись последнего перезапуска системы по причине срабатывания watchdog.
- n** первая новая запись.

Значение параметра по умолчанию (если параметр не указан) равно “**f**”.

Если определён второй параметр, то команда возвращает пакет данных в 16-ричном виде.

Пример:

```
> &req.first
$PGIO,c,00011305,070604,011449,00*39
$GPGGA,011449,5548.5403,N,3738.4200,E,1,03,0.0,0.0,M,,,,,00*00
$GPRMC,011449,A,5548.5403,N,3738.4200,E,000.0,000.30,070604,0.0,E*7E
OK
>
```

Возврат данных завершается выводом **OK** в случае успешного поиска по заданному критерию или **End**, если запись не найдена.

Например, вернуть последнюю запись, произведённую после перезапуска *Устройства* по причине срабатывания watchdog:

```
> &re.f=w
End
>
```

4.6.3. &req.[<psw>].last=[<N>][,{h|x}][,reverse]

Возвращает указанное количество последних записей из *ЛОГ* в порядке от убывания номеров записей.

Если параметр **<N>** не указан, то будет возвращен текущий пакет данных. Если параметр **<N>** равен 0, то будут возвращены все записи из *ЛОГ*. Если значение параметра **<N>** равно 1 или более, то будет возвращено указанное количество записей из *ЛОГ*. Максимальное значение параметра **<N>** определяется объемом *ЛОГ*.

Пример ниже иллюстрирует запрос на чтение последних трех записей в составе PGIO, GPGGA и GPRMC с выводом идентификатора *Устройства* в составе предложения GPIO:

```
> &set.echo=,,97
```

```
OK
> &re.l=3
$PGIO,1001,_,00005580,221205,061627,00,00,,,,*59
$GPGGA,061627,5547.8728,N,03735.6138,E,1,08,8,179.6,M,14.5,M,0.0,0000*77
$GPRMC,061627,A,5547.8728,N,03735.6138,E,3.4,311.0,221205,,*13
$PGIO,1001,_,00005579,221205,061556,00,00,,,,*5A
$GPGGA,061556,5547.8364,N,03735.5681,E,1,08,8,182.3,M,14.5,M,0.0,0000*79
$GPRMC,061556,A,5547.8364,N,03735.5681,E,3.5,82.2,221205,,*26
$PGIO,1001,_,00005578,221205,061526,00,00,,,,*5C
$GPGGA,061526,5547.8517,N,03735.5966,E,1,07,14,176.5,M,14.5,M,0.0,0000*45
$GPRMC,061526,A,5547.8517,N,03735.5966,E,3.0,218.7,221205,,*14
OK
>
```

Вывод данных завершается **OK**, если все запрошенные записи были найдены и возвращены, или **End**, если процесс вывода записей остановлен по причине «нет записи» или по прерыванию пользователя вводом Escape.

ВНИМАНИЕ! Команда **&req.last** выводит данные в форматах, определённых командой **&set.echo**. Если команда **&set.echo** была исполнена с нулевым значением в младшем разряде третьего параметра (например, **&set.echo=5,all,90**), то команда **&req.last** вернёт только «OK», без пакетов данных!

Второй параметр N или X команды **&req.last** является не обязательным и используется для возврата данных в 16-ричном представлении. Параметр X включает старт-стопный режим. Следующий пример показывает вывод в режиме «старт-стоп»:

```
> &re.l=3,h
$х0000000C70BD098E034E9862023A03558506001007FE009110*7D      <Enter>
$х0000000B70B3098E034E9868023A03658505001A07E4009110*02      <Enter>
$х0000000A70A9098E034E9875023A0399850500110768009110*08
OK
>
```

В этом примере, для вывода следующей строки пользователь (хост) вводил символ Enter как указано справа.

- При вводе символа Enter в возвращенной записи устанавливается флаг «прочитано» (см. команду **&req.new**).
- Исполнение команды **&req.last** с параметром X возвращает данные непрерывным потоком. При этом по окончании возврата данных не требуется ввода символа подтверждения (Enter).

Третий параметр REVERSE команды **&req.last** является не обязательным и используется для возврата записей из ЛОГ в порядке следования номеров записей (от младшего к старшему). Например:

```
> &set.echo=, ,11
OK
> &re.l=3, ,r
$PGIO,_,00005578,221205,061526,00,00,,,,*5C
$PGIO,_,00005579,221205,061556,00,00,,,,*5A
$PGIO,_,00005580,221205,061627,00,00,,,,*59
OK
>
```


4.6.4. &req.[<psw>].new[=?|<N>,{h|x}]

Возвращает *новые записи* из *ЛОГ* в порядке возрастания порядковых номеров.

Режим вывода *новых записей* определяется двумя параметрами:

- нет параметра вернуть все *новые записи*
- ? вернуть количество *новых записей*
- <N> вернуть указанное количество *новых записей*
- h режим возврата *новых записей* в 16-ричном представлении с ожиданием подтверждения. Количество возвращенных записей с ожиданием подтверждения определяется пятым параметром команды [&set.system](#)
- x режим возврата *новых записей* в 16-ричном представлении без ожидания подтверждения

Возврат данных оканчивается возвратом ОК, если запрошенное количество пакетов данных возвращено или End, если обнаружен флаг «конец ЛОГ» или нет *новых записей*.

Например:

```
> &req.new=?
2
ОК
> &req.new
$PGIO,_,00008089,251205,110709,00,00,,,*,48
$GPGGA,110709,5548.2554,N,03735.6411,E,1,04,13,138.8,M,14.5,M,0.0,0000*42
$GPRMC,110709,A,5548.2554,N,03735.6411,E,2.5,318.1,251205,,*,14
$PGIO,_,00008088,251205,110639,00,00,,,*,4B
$GPGGA,110639,5548.2532,N,03735.6501,E,1,04,65,141.0,M,14.5,M,0.0,0000*47
$GPRMC,110639,A,5548.2532,N,03735.6501,E,0.9,241.0,251205,,*,14
End
> &req.new=2
$PGIO,_,00008089,251205,110709,00,00,,,*,48
$GPGGA,110709,5548.2554,N,03735.6411,E,1,04,13,138.8,M,14.5,M,0.0,0000*42
$GPRMC,110709,A,5548.2554,N,03735.6411,E,2.5,318.1,251205,,*,14
$PGIO,_,00008088,251205,110639,00,00,,,*,4B
$GPGGA,110639,5548.2532,N,03735.6501,E,1,04,65,141.0,M,14.5,M,0.0,0000*47
$GPRMC,110639,A,5548.2532,N,03735.6501,E,0.9,241.0,251205,,*,14
ОК
>
```

4.6.5. &req.[<psw>].pos[=x]

Возвращает пакет данных соответствующий последней валидной координате при выполнении условия NSAT больше или равно значению <nSat>. Где значение <nSat> устанавливается командой [&set.log](#).

Параметр «X» является необязательным и служит для возврата данных в 16-ричном представлении.

Возврат данных завершается ОК если последняя валидная координата существует или End, если таковой нет с момента включения *Устройства*. Пример 1, валидной координаты на существует:

```
> &req.pos  
End
```

Пример 2:

```
> &req.pos  
$PGIO, ,00000400,140906,185210,01*35  
$GPGBA,185210,5548.5329,N,3738.4109,E,1,04,0.0,0.0,M,,,,,00*0F  
$GPRMC,185210,A,5548.5329,N,3738.4109,E,002.31,015.13,140906,0,E*11  
OK  
>
```

4.6.6. &req.[<psw>].range=<Start>,<N>[,<h|x>][,<reverse>]

Возвращает <N> записей, начиная с записи с номером <Start> в порядке убывания порядковых номеров. Параметр <N> определяет количество возвращаемых записей. Например:

```
> &re.r=400,2  
$PGIO, ,00000400,140604,175513,01*35  
$GPGBA,175513,5548.5329,N,3738.4109,E,1,04,0.0,0.0,M,,,,,00*0F  
$GPRMC,175513,A,5548.5329,N,3738.4109,E,002.31,015.13,140604,0,E*11  
$PGIO, ,00000399,140604,175508,01*38  
$GPGBA,175508,5548.5340,N,3738.4208,E,1,04,0.0,0.0,M,,,,,00*08  
$GPRMC,175508,A,5548.5340,N,3738.4208,E,002.38,009.24,140604,0,E*46  
OK  
>
```

Возврат данных завершается **OK**, если поиск удовлетворяет заданному критерию, или **End**, если процесс вывода записей остановлен по причине «нет записи».

Третий параметр **H** или **X** является не обязательным и служит для возврата данных в 16-ричном представлении. Если указан параметр **H**, то возврат данных производится в старт-стопном режиме (подробнее см. 4.6 описание «старт-стопный режим» и в документе «Книга II. Руководство администратора»):

- h** старт-стопный режим вывода *новых записей* в 16-ричном представлении с ожиданием подтверждения (**hex**)
- x** пакетный режим вывода *новых записей* в 16-ричном представлении без ожидания подтверждения (**hex**)

Следующий пример показывает вывод данных в режиме «старт-стоп»:

```
> &re.r=11,3,h  
$x0000000B70B3098E034E9868023A03658505001A07E4009110*02 <Enter>  
$x0000000A70A9098E034E9875023A0399850500110768009110*08 <Enter>  
$x00000009709F098E034E9879023A03AA8545001A0766009110*01  
OK  
>
```

В этом примере, для вывода следующей строки пользователь вводил символ Enter.

Четвёртый параметр **REVERSE** команды &req.range является не обязательным и используется для возврата записей из *ЛОГ* в порядке следования номеров записей (от младшего к старшему). Пример вывода предложений PGIO в режиме «Реверс»:

```
> &set.echo=, ,91  
OK  
> &req.range=46,5,,r  
$PGIO,_,00000046,210105,082141,00,00*45  
$PGIO,_,00000047,210105,082142,00,00*47
```

```
$PGIO,_,00000048,210105,082143,00,00*49
$PGIO,_,00000049,210105,082144,00,00*4F
$PGIO,_,00000050,210105,082145,00,00*46
OK
>
```

4.6.7. &req.[<psw>].time=[<ddmmyy>],[<hhmmss>],[<N>],[hexmode]

Запрос на вывод **<N>** записей из истории, начиная с указанной даты и времени. Если дата не указана, то *Устройство* возвращает **<N>** записей последнего дня с указанного времени. Если время не указано, то вывод начинается с конца суток (23:59:59). Если не указан параметр **<N>**, то будет выведена только одна запись.

Например:

```
> &re.time=140604,180203,2
$PGIO, ,00000482,140604,180203,01*33
$GPGGA,180203,5548.5389,N,3738.4127,E,1,03,0.0,0.0,M,,,,,00*02
$GPRMC,180203,A,5548.5389,N,3738.4127,E,002.73,013.5,140604,0,E*7C
$PGIO, ,00000481,140604,180158,01*3D
$GPGGA,180158,5548.5385,N,3738.4136,E,1,03,0.0,0.0,M,,,,,00*03
$GPRMC,180158,A,5548.5385,N,3738.4136,E,002.57,001.49,140604,0,E*40
OK
>
```

Возврат данных завершается **OK**, если поиск удовлетворяет заданному критерию, или **End**, если процесс вывода записей остановлен по причине «нет записи».

Возврат данных может начинаться идентификатором *Устройства* (см. третий параметр команды [&set.echo](#)).

Четвёртый параметр является не обязательным и может включать старт-стопный режим вывода данных:

- h** старт-стопный режим вывода *новых записей* в 16-ричном представлении с ожиданием подтверждения (**hex**)
- x** пакетный режим вывода *новых записей* в 16-ричном представлении без ожидания подтверждения (**hex**)

Следующий пример показывает вывод 16-ричных пакетов данных в режиме «старт-стоп»:

```
> &re.t=140604,180203,2,h
$х000001E270B3098E034E9868023A03658505001A07E4009100*02      <Enter>
$х000001E0709F098E034E9879023A03AA8545001A0766009100*01
OK
>
```

В этом примере, для вывода следующей строки пользователь вводил символ Enter.

4.7. Настройка работы порта TE

4.7.1. &set.[<psw>].baud=<baud_index>[,Rightnow]

Команда позволяет изменять скорость обмена порта *TE* связи с *Устройством*.

Значение параметра **<baud_index>** определяет индекс скорости обмена согласно таблице:

Индекс	5	6	7	8	9	10	11
бит/сек	9600	14400	19200	28800	38400	57600	115200

Пример диалога при работе с *Устройством* с помощью терминальной программы:

```
> &set.baud=5
set TE baud 9600 bps and press Enter
(изменить настройки порта и нажать клавишу Enter)
New baud OK
>
```

На изменение настроек порта связи на стороне внешнего терминального устройства (компьютера) отведено 15 секунд. Если в течение 15 секунд *Устройство* не получит подтверждения (код клавиши Enter) на новой скорости, то скорость обмена данными с портом *TE* будет восстановлена в прежнее значение и TE-порт вернет сообщение **timeout** на прежней скорости. Например:

```
> &set.baud=5
set TE baud 9600 bps and press Enter
(настройки не изменены, нет кода Enter в течение 15 секунд)
timeout
>
```

Второй необязательный параметр Rightnow (или только одна буква “R” или “r”) указывает на то, что команда будет исполнена без ожидания подтверждения. В этом случае TE-порт возвращает OK на новой скорости порта. Параметр Rightnow рекомендуется использовать в случае, если команда передается *Устройству* по IP-каналу.

4.7.2. &set.[<psw>].echo=[<period>],[<filter>],[<mode>]

Эта команда предназначена для установки периода автоматического вывода данных в порт *TE* и настройки фильтра вывода данных командами **&req.last**, **&req.date**, **&req.first** и **&req.range**. Параметры разделяются запятыми и являются не обязательными для ввода. Пропущенный параметр сохраняет текущие значения.

Первый параметр **<period>** устанавливает период вывода данных *TE*. По умолчанию он равен 5 периодам вывода NMEA данных навигационным приёмником.

Второй параметр **<mode>** может быть одним из трёх значений “off”, “all” или “valid” (допускается ввод только первой буквы, соответственно “o”, “a” или “v”):

off выключить возврат данных
valid возвращать только данные с «верными» (валидными)

координатами
all возвращать все данные вне зависимости от признака
валидности (по умолчанию)

Значение третьего параметра **<mode>** представляется двумя 16-ричными цифрами, позиции битов которого определяют:

- 01 PGIO
- 02 GGA
- 04 RMC
- 08 16-ричный режим возврата данных
- 10 режим «эхо» возврата введённого символа
- 20 режим отладки
- 40 режим «прямого» возврата в *TE*-порт NMEA- предложений принятых *Устройством* от навигационного приемника.. Включение этого режима автоматически блокирует возврат пакетов данных установленных командой [&set.echo](#)
- 80 режим, при котором идентификационный номер *Устройства* возвращается в составе пакета данных

В случае активного соединения с IP-сервером и отключенных режимов «режим «эхо» возврата введённого символа» и «режим «эхо» возврата введённого символа» команды введённые в *TE*-порт устройства после исполнения возвращают ответ только в IP-канал GSM-модуля. Такой режим рекомендуется использовать при подключении к *TE*-порту *Устройства* внешнего контроллера. Например, датчика уровня жидкости.

В режиме с отключенным «эхо» (бит 10h в значении параметра **<mode>** равен нулю) *Устройство* не возвращает отладочных сообщений и информационной справки после включения питания. Отключение режима «эхо» рекомендуется производить при эксплуатации прибора совместно с подключенным внешним контроллером к порту *TE Устройства*.

По умолчанию значение **<mode>** равно “17” (соответствует: «эхо» включено, возврат данных производится в форматах PGIO, GGA и RMC).

Например, для включения возврата только двух протоколов GGA и PGIO значение параметра **<mode>** должно быть равно “13”, а при необходимости возврата идентификатора *Устройства* в составе предложения PGIO значение бита 80 параметра **<mode>** следует установить в “1”.

Пример 1, настроить вывод PGIO и RMC предложений не зависимо от признака достоверности координат каждый раз, когда навигационный приёмник сформирует пакет данных:

```
> &set.echo=1,all,15
OK
>
```

Пример 2, не изменять период вывода на *TE* и выводить только «верные» координаты в формате GGA:

```
> &set.echo=v,12
OK
>
```

Пример 3, выводить идентификационный номер в составе предложения PGIO:

```
> &set.echo=, , 97
OK
> &req.last=2
$PGIO,1001, , 00000482,140604,180203,01*33
$GPGGA,180203,5548.5389,N,3738.4127,E,1,03,0.0,0.0,M,,M,, , 00*02
$GPRMC,180203,A,5548.5389,N,3738.4127,E,002.73,013.5,140604,0,E*7C
$PGIO,1001, , 00000481,140604,180158,01*3D
$GPGGA,180158,5548.5385,N,3738.4136,E,1,03,0.0,0.0,M,,M,, , 00*03
$GPRMC,180158,A,5548.5385,N,3738.4136,E,002.57,001.49,140604,0,E*40
OK
>
```

На стадии настройки *Устройства* и работы с командным интерфейсом вручную не рекомендуется сбрасывать флаг [10] эхо-возврата вводимых символов.

ВНИМАНИЕ! Не допускайте исполнение команды с нулевым значением параметра **<mode>** в младшем разряде. Иначе команды чтения записей (например, **&req.last**) не будут возвращать данные!

4.8. Управление режимом автоматической записи данных в LOG

4.8.1. &set.[<psw>].log=[<t1>],[<F>],[<sp>],[<nS>],[<af>],[<df>],[<ns>],[<se>],[<ct>],[<lm>],[<pp>],[<t2>]

Устанавливает режимы работы модуля регистрации навигационных данных.

<t1> (time factor 1) - определяет интервал записи данных в *журнал событий* кратного периода возврата NMEA-предложений навигационным приёмником при скорости выше пороговой **<sp>**. Допустимые значения 5..65535. Значение по умолчанию 60.

<F> (Filter) - настраивает фильтр записи и может быть одним из четырёх значений “off”, “all”, “valid” или “disabled” (допускается ввод только первой буквы слова, “o”, “a”, “v” или “d” соответственно, значение по умолчанию “all”):

- off** выключить запись в *ЛОГ*;
- all** записывать все пакеты данных (вне зависимости от валидности);
- valid** записывать только верные (достоверные) данные;
- diff** записывать данные по критерию превышено «отклонение от курса» (см. параметр **<df>**);
- dis** запись отключена (см. реакцию OBJ_LOG.SET_LOG_STATUS.REC_DISABLE).

<sp> (speed) - устанавливает скорость в узлах относительно поверхности земли (SOG), ниже которой пакеты данных не записываются в *ЛОГ*. Допустимые значения 0..127. Значение по умолчанию «0», анализ скорости движения не производится.

<nS> (number of Satellites) - определяет минимальное количество видимых навигационных спутников, ниже которых запись в *ЛОГ* не производится. Допустимые значения: 0..15 (значение по умолчанию 4).

<af> (acceleration factor) - определяет пороговое значение изменения скорости движения ТС (узлы) от момента последней записи данных в *ЛОГ*, при достижении которой производится внеочередная запись данных в *ЛОГ*. Значение параметра **<af>** не влияет на процесс периодической регистрации данных, а позволяет производить запись только дополнительных пакетов данных. Допустимые значения 0, 3..100 (опция). Значение по умолчанию 0 (запись в *ЛОГ* при изменении скорости выключена).

<df> (deviation factor) - определяет чувствительность механизма записи внеочередного пакета данных в *ЛОГ* при изменении направления движения, «дифференциальный» режим записи. Меньшему значению **<df>** соответствует более частая запись данных в *ЛОГ* на поворотах. Период записи во время стоянки определяется значением параметра **<t1>**, минимальное значение которого в этом режиме равно 15. Диапазон допустимых значений параметра **<df>** в пределах от 5 до 255. По умолчанию значение параметра равно 20 (изменять не рекомендуется).

<ns> (null speed) - определяет интервал времени в секундах движения ТС со скоростью ниже пороговой **<sp>**, по достижении которого производится внеочередная запись в *ЛОГ* (опция). Если значение параметра **<ns>** равно нулю, то внеочередная запись при остановке *ТС* не производится. Допустимые значения 0, 3..255. По умолчанию значение равно 10.

<se> (speed event) - определяет частоту генерации события [OBJ LOG.0.LOG_SPEEDEVENT](#) (опция). Допустимые значения 0, 10..255. По умолчанию значение равно 0, событие не генерируется.

<ct> (confirm timeout) - определяет время ожидания в секундах ключевого слова от *Пользователя*, подтверждающего успешное получение данных. Допустимые значения 5..254. Значение по умолчанию значение равно 60.

<lm> (log mode) - управляет дополнительными режимами работы модуля приёма и анализа навигационных данных (опции). Бит 01h определяет режим работы программного гео-одометра: 1 (включен) или 0 (выключен, по умолчанию). Бит 20h управляет режимом переключения режима «пешеход» и «автомобиль» зависящий от скорости движения, см. параметры **<es>** и **<sz>** команды [&set.system](#) (по умолчанию режим включен; только для версии приёмника типа VENUS, см. [Индикатор опций прошивки](#)).

<pp> (position pinning) - управляет режимом «time position pinning» (опция). Допустимые значения этого параметра зависят от версии встроенного навигационного приёмника. В версии GPS VENUS допустимые значения: 1 (включен) или 0 (выключен), по умолчанию режим выключен. В версии GPS ET-316: 0 (выключен), 1..127 время в секундах (time position pinning).

<t2> (time factor 2) - определяет интервал записи данных в *журнал событий* кратного периода вывода NMEA-предложений навигационным приёмником при скорости ниже пороговой **<sp>**. Допустимые значения 0, 5..65535. Значение по

умолчанию 0, запись данных в ЛОГ при скорости ниже пороговой <sp> отключена.

Пакет данных *ЛОГ* записанный в результате выполнения дополнительного условия содержит соответствующий флаг-признак в поле <MacroID> [4].



В «дифференциальном» режиме записи *Устройство* производит анализ направления движения транспортного средства и записывает данные в случае превышения угла направления движения с момента предыдущей записи данных. При этом поле <MacroID> имеет соответствующий признак [4]. Данный режим работает совместно с «периодическим» режимом записи (см. параметр <t1> команды [&set.log](#)). Следует помнить, что в случае стоянки или малой скорости движения значение «направление движения» возвращаемое навигационным приёмником меняется. Поэтому при установке «дифференциального» режима записи рекомендуется устанавливать значение параметра <sp> (фильтр скорости) больше нуля.

Пример 1. Включить запись в *журнал событий* только данных с верными координатами каждые 5 циклов приёма NMEA-предложений от навигационного приёмника и при условии, что количество видимых спутников более трёх:

```
> &set.log=5,valid,0,4
OK
>
```

Пример 2. Отключить запись данных в *журнал событий*, настройки интервала записи (событие) оставить без изменения:

```
> &set.log=,off
OK
>
```

Устройство принимает NMEA предложения до тех пор, пока не будут получены все необходимые форматы предложений, которые задаются параметром <mask> (см. [&set.nmea](#)). По факту получения всех форматов NMEA предложений *Устройство* увеличивает внутренний счётчик и, в случае его превышения значению <period> (см. команду [&set.log](#)), вырабатывает внутреннее сообщение GPS_RD4REC, при этом принятый пакет данных может быть записан в *ЛОГ* (см. второй параметр команды [&set.log](#)).

ВНИМАНИЕ! Если навигационный приёмник не возвращает данных в формате GGA (например, InmarsatD+) *Устройство* необходимо настроить командой [&set.log](#) с параметром <nSat> равным нулю. Иначе запись данных в ЛОГ производиться не будет, так как RMC- предложение не содержат информации о количестве видимых спутников.

4.8.2. &add.[<psw>]log={Byte|Word|Double|None}[,<n_reg>]

При необходимости размер структуры данных LOG может быть увеличен на требуемое количество пользовательских регистров или восстановлен в первоначальное значение с помощью команды [&add.log](#). Первый параметр команды определяет размер пользовательского регистра:

- Byte** один байт;
- Word** одно слово (два байта);
- Double** двойное слово (четыре байта);
- None** восстановление размера структуры LOG в исходное значение.

Второй параметр определяет количество пользовательских регистров. Значение 0 (ноль) удаляет пользовательские регистры из состава структуры ЛОГ. Параметр действителен только в версиях firmware с поддержкой пользовательских регистров.

Команда возвращает ОК (или ERROR в случае если ошибка в параметре). Например:

```
> &add.log=w
OK
>
```

Команда [&add.log](#) отключена в версии firmware, в которой нет поддержки пользовательских регистров.

4.8.3. &get.[<psw>]log=[Full][,Hex]

Команда возвращает значения <N> пользовательских регистров в формате:

<reg_id>:<value>

где <reg_id> - порядковый номер пользовательского регистра, <value> - значение регистра. Исполнение команды завершается возвратом ОК или End, если пользовательские регистры не определены.

Количество пользовательских регистров <N> и их информационная ёмкость определяются командой [&add.log](#).

Если задан параметр Full команда возвращает значения всех пользовательских регистров максимальной информационной ёмкости (DOUBLE WORD). Выполнение команды завершается возвратом ОК.

Если задан второй параметр Hex, то значения пользовательских регистров возвращаются в 16-ричном виде. По умолчанию, если параметр Hex не определён, значения пользовательских регистров возвращаются в 10-чном представлении.

Команда &get.log отключена в версии firmware, в которой нет поддержки пользовательских регистров (см. [Индикатор опций прошивки](#)).

4.9. Работа с объектом Таймер

Программируемый таймер предназначен для генерирования внутренних системных событий, необходимых для управления процессами, распределёнными по времени.

Работа с таймером поддерживается тремя командами: *добавить таймер* (один или несколько), *установить параметры таймера* и *удалить все таймеры*.

Работа с таймерами производится в оперативной памяти и сразу вступают в силу (начинают работать. См. также [&set.programmode](#)). При этом, по окончании настроек, произведённые изменения требуется сохранить во FLASH командой [&set.save](#). В оперативной памяти объект типа *Таймер* занимает 9 байт.

ВНИМАНИЕ! Максимально количество объектов типа [Таймер](#), [Условие](#) и [Макро](#) ограничено свободным пространством сектора конфигурации в памяти FLASH и свободной оперативной памяти.

Объект Таймер может быть одним из двух типов - с «бесконечным» числом повторений и «конечным», т.е. с заданным числом повторений.

Объект *Таймер* с «бесконечным» числом повторений периодически генерирует событие `TIMER_RUN` с периодом `<delay>`.

Объект *Таймер* с «конечным» числом повторений генерирует событие `TIMER_RUNi` с периодом `<delay>` и количеством повторений, определяемым параметром `<Nrepeat>`. Генерируемое событие `TIMER_RUNi` принимает значения от 1 до `<Nrepeat>` включительно. В завершении работы, через `<delay>` секунд после последнего события `TIMER_RUNi` (где $i := <Nrepeat>$) такой таймер генерирует событие `TIMER_END`.

Состояние признака «таймер запущен» может быть определено при настройке таймера командой [&set.timer](#) и записывается в конфигурацию устройства (см. [&set.save](#)). Такой таймер начинает работать автоматически при запуске *Устройства* по питанию или после исполнения команды [&restart](#).

4.9.1. [&add.\[<psw>\].timer=<num>](#)

Добавляет указанное количество объектов типа *Таймер* (*см. примечание ниже). Количество объектов задаётся параметром `<num>`. Созданный объект имеет параметры по умолчанию:

- `<status>` 0 (состояние стоп)
- `<delay>` время задержки 1 секунда
- `<repeat>` количество повторов 1

Возвращает число соответствующее количеству объектов указанного типа.

Например:

```
> &add.timer=2
2
OK
>
```

4.9.2. &get.[<psw>].timer[=?]

Возвращает информацию всех объектов типа Таймер в формате:

```
timer <id>:<status>,<delay>,<repeat>
timer <id>:<status>,<delay>,<repeat>:<cur_delay>,<cur_repeat>
```

Параметры описаны в параграфе &set.timer.

Исполнение команды с параметром «знак вопроса» дополнительно выводит текущие значения счетчиков для каждого объекта *таймер*. Например:

```
> &get timer=?
timer 1:21,5,3:4,2
OK
>
```

В этом примере *таймер* ID1 был создан для генерирования трёх событий с интервалом 5 секунд. Его состояние на момент исполнения команды &get.timer было активно (*таймер* запущен), до следующего события осталось 4 секунды, ранее уже было генерировано 2 события.

4.9.3. &set.[<psw>].timer=<id>,<mode>,<delay>,<Nrepeat>

Устанавливает параметры объекта *Таймер* с идентификационным номером, указанным в первом параметре команды <id>.

Описание параметров:

- <mode>** параметр задаётся и отображается 16-ричным числом и определяет режим работы объекта *Таймер*:
 - 80** в режиме повторения событие TIMER_END не генерируется;
 - 20** режим повторения, количество повторов запуска таймера указывается третьим параметром <Nrepeat>;
 - 10** режим «non-stop». Значение параметра <Nrepeat> игнорируется.Младший бит параметра <mode> соответствует требуемому состоянию таймера. Состояние таймера восстанавливается после запуска *Устройства*:
 - 00** таймер в состоянии «остановлен»;
 - 01** таймер в состоянии «запущен».
- <delay>** период события TIMER_RUN в секундах (максимум 65534).
- <Nrepeat>** количество повторов (максимум 65534), если установлен бит 20 параметра <mode>.

Для возврата значений параметров объекта *Таймер* используйте команду [&get.timer](#).

Команда возвращает ОК в случае успешного исполнения или ERROR, если *Таймер* с указанным номером не существует.

4.9.4. &del.[<psw>.]timer[=<id>]

Удаляет объект типа Таймер с номером указанным в первом параметре. Если параметр <id> не указан, удаляются все объекты.

Команда возвращает ОК в случае успешного исполнения или End, если объектов нет.

Например:

```
> &del.timer=1
OK
> &del.ti
End
>
```

4.10. Работа с объектом «Условие»

Объект типа *Условие* позволяет определять дополнительное условие для исполнения *Макро* инструкции.

Работа с *Условиями* поддерживается тремя командами: *добавить условие* (одно или несколько), *установить параметры условия* и *удалить все условия*.

Работа с *Условиями* производится в оперативной памяти и сразу вступают в силу (начинают работать. См. также &set.programmode). При этом, по окончании настроек, произведённые изменения требуется сохранить во FLASH командой [&set.save](#). Размер требуемой памяти для одного объекта типа *Условие* **4 байта**.

Условие с порядковым номером равным «0» всегда возвращает значение «истина». *Условие* с несуществующим порядковым номером возвращает значение «ложь». Последнее необходимо помнить при создании *Макро* инструкции, в которой присутствует указатель (номер) на требуемое *Условие*. Условие с этим номером должно быть создано (см. команду [&add.condition](#)) и означено командой &et.condition.

ВНИМАНИЕ! Максимально количество объектов типа [Таймер](#), [Условие](#) и [Макро](#) ограничено свободным пространством сектора конфигурации в памяти.

Объект *Условие* для OBJ_USER (OBJ_USER.<user_id>.<req_status>) принимает значение ИСТИНА при выполнении условия **<req_status> <= <user_status>**, где:

<user_status> состояние объекта *Пользователь* с номером <user_id> (см. команду [&add.user](#)), а **<req_status>** - запрошенное состояние этого объекта. При работе с *Макро* инструкциями подобные

условные выражения можно использовать для выполнения каких-либо действий в зависимости от «статуса» пользователя.

4.10.1. &add.[<psw>.]condition=<num>

Добавляет количество объектов типа *Условие* указанное в параметре <num>.

Возвращает общее количество объектов в памяти *Устройства* и **ОК**.

Например:

```
> &add.condition=2
2
ОК
>
```

4.10.2. &set.[<psw>.]condition=<id>,<obj>,<obj_id>,<status>[,not]

Устанавливает параметры объекта типа «условие» в порядке следования:

- <id> идентификационный номер объекта «условие» (десятичное число)
- <obj_type> тип объекта подлежащего проверке (десятичное число)
- <obj_id> идентификационный номер объекта (десятичное число)
- <obj_status> проверяемое состояние объекта (16-ричное число)
- Not дополнительный параметр, указывающий на выполнение операции логического «НЕ» над результатом.

Не все объекты системы являются допустимыми в определении объекта *Условие*. Перечень допустимых объектов при работе с условными выражениями макро инструкций представлен в [Таблице значений констант объекта *Условие*](#). Команда возвращает **ОК** в случае успешного выполнения или **ERROR**, если объект *Условие* с указанным идентификационным номером не найден.

Например:

```
> &set.condition=1,1,1,0,not
ОК
> &set.condition=100,1,2,0
ERROR
>
```

4.10.3. &get.[<psw>.]condition

Возвращает TE все объекты типа *Условие*. Формат возвращенных данных:

```
if <ConditionID>: <ObjType>,<ObjID>,<ObjStatus> [not] <real status>
```

где

ConditionID идентификационный номер объекта типа *Условие*

ObjType тип объекта, состояние которого подлежит проверке
 ObjID идентификационный номер объекта проверки
 ObjStatus запрашиваемое состояние
 not флаг отрицания (логическое НЕ) настройки объекта *Условие*
 real status действительное состояние объекта *Условие* на момент
 выполнения команды &get.condition

Например:

```
> &get.condition if 1:1,1,0 true
if 2:1,2,32 not false
OK
>
```

Первое условие в примере соответствует выражению:

Если Зона 1 разомкнута

Второе условие –

Если Зона 2 в состоянии «нет тревоги»

4.10.4. &del.[<psw>].condition[=<id>]

Удаляет объект типа Условие с номером указанным в первом параметре. Если параметр <id> не указан, удаляются все объекты.

Команда возвращает OK в случае успешного исполнения или End, если объектов нет.

Например:

```
> &del.condition=1
OK
> &del.co
End
>
```

4.11. Работа с объектом Макро

Работа с *Макро* инструкциями поддерживается тремя &-командами: добавить Макро (или Макросы), установить параметры Макро и удалить все Макросы.

Устройство генерирует системные события и исполняет объявленные пользователем *Макро* инструкции, если системное событие совпадает с объявленным событием в *Макро* инструкции. Работа с *Макро* инструкциями производится в оперативной памяти. Исполнение Макро инструкций может быть приостановлено на время настройки *Устройства* командой &set.programmode. При этом, по окончании настроек, произведённые изменения требуется сохранить во FLASH командой [&set.save](#).

В оперативной памяти один объект типа *Макро* занимает **11 байт**.

ВНИМАНИЕ! Максимально количество объектов типа [Таймер](#), [Условие](#) и [Макро](#) ограничено свободным пространством сектора конфигурации в памяти FLASH и объёмом свободной оперативной памяти (free heap).

4.11.1. &add.[<psw>.]macro=<num>

Добавляет указанное количество объектов типа *Макро*. Количество объектов задаётся параметром <num>. Созданный объект по умолчанию имеет значения всех параметров равным 0 (ноль).

Возвращает общее количество объектов указанного типа. Например:

```
> &add.macro=7
7
ОК
>
```

4.11.2. &get.[<psw>.]macro[=<begin>[,<number>]]

Возвращает информацию объекта(ов) *Макро* в формате:

```
macro <id>:<en>,<st>,<sid>,<se>;<dt>,<did>,<de>
```

или

```
macro <id>:<en>,<st>,<sid>,<se>;<dt>,<did>,<de> if.<co>
```

где

<id> уникальный идентификационный номер (>0);

<en> состояние включено (1) или выключено (0);

Три параметра определяющих объект-источник:

<st> тип объекта (source object type);

<sid> идентификационный номер объекта (src id);

<se> тип сообщения (source event).

Три параметра определяющих объект-приёмник:

<dt> тип объекта (destination object type);

<did> идентификационный номер объекта;

<de> тип сообщения (destination event / reaction).

Дополнительный параметр <co> возвращается только в случае, если при определении макро был указан порядковый номер объекта *Условие*.

Если определён параметр <begin> команда возвращает все *Макро*-инструкции начиная с указанного в параметре номера. Сообщение ERROR возвращается если *Макро* с указанным номером не существует. По умолчанию (если параметр не задан) значение параметра равно 1.

Второй параметр `<number>` определяет количество возвращаемых макроинструкций. По умолчанию (если параметр не задан) значение параметра равно 255.

Например, создано три макроса. Команда [&get.macro](#) без параметров возвращает все макросы:

```
> &get.macro
macro 1:1,4,1,1;12,0,0
macro 2:1,1,1,1;7,1,1 if.2
macro 3:0,0,0,0;0,0,0
OK
>
```

Следующий пример показывает исполнение команды [&get.macro](#) с двумя параметрами:

```
> &get.macro=2,1
macro 2:1,1,1,1;7,1,1 if.2
OK
>
```

4.11.3. **&set.[<psw>].macro=<id>,<en>,<st>,<sid>,<se>,<dt>,<did>,<de> [,<condition> [,not]]**

Устанавливает параметры объекта *Макро* с идентификационным номером, указанным в первом параметре команды `<id>`.

<id> уникальный идентификационный номер (>0);

<en> состояние включено (1) или выключено (0);

Значения следующих семи параметров соответствуют перечню возвращаемых данных команды [&get.macro](#)

Дополнительный параметр `not` следует использовать, если определён номер условия `<condition>`. Параметр `not` инвертирует значение условия на момент исполнения макроса. Использование этого параметра позволяет сократить количество объектов *Условие*.

4.11.4. **&set.[<psw>].macro=<id>,run[,any]**

Запускает на исполнение макрос с номером `<id>`. Если в макросе с номером `<id>` определено условие, то макрос будет исполнен если значение условия на момент исполнения команды истинно.

Если определён третий параметр **any**, то макрос с номером `<id>` будет исполнен в любом случае, вне зависимости от результата проверки условия.

Команда возвращает ОК если макрос найден и исполнен, иначе – ERROR.

4.11.5. &del.[<psw>.]macro[=<id>]

Удаляет объект типа Макро с номером указанным в первом параметре <id>. Если параметр <id> не указан, удаляются все объекты.

Команда возвращает OK в случае успешного исполнения или End, если объектов нет.

Например:

```
> &del.macro=1
OK
> &del.ma
End
>
```

4.12. Работа с объектом Пользователь

Объект *Пользователь* предназначен для определения типа и адреса исходящего соединения (например, послать SMS-сообщение по предопределённому адресу или произвести соединение с IP-сервером и определённым IP-портом), а в случае входящего вызова объект *Пользователь* позволяет идентифицировать абонента.

При включении питания Устройство может автоматически произвести подключение к IP-серверу и передать свой идентификатор, если в его настройках:

- не определены макросы (см. [Работа с объектом Макро](#));
- определён один Пользователь типа IP-сервер (см. [Работа с объектом Пользователь](#)).

Формат строки возвращаемой Устройством в IP-порт при подключении к серверу:

ID=<imei>,VER=<version><Enter>

или, если в версии прошивки отключена [опция](#) работы с IMEI модема:

ID=<sysId>,VER=<version><Enter>

Ответ на вопрос «как послать сообщение» описан в разделе [Работа с объектом Сообщение](#).

В отличие от принципов работы с объектами *Таймер*, *Макро* и *Условие* работа с объектом *Пользователь* имеет сокращенный набор команд: добавить (add), вернуть (get) и удалить (delete). В связи с этим, для редактирования (изменения) значения параметра какого-либо объекта *Пользователь* необходимо удалить все объекты (команда &del), а затем создать объекты заново с требуемыми параметрами (команда &add).

Можно определить несколько объектов *Пользователь*. Каждый объект *Пользователь* имеет свой порядковый номер, на который могут ссылаться такие объекты, как *Макро* и *Условие*.

Поддерживаемые типы объекта *Пользователь*: «Телефон», «Модем», «IP-сервер TCP» или «IP-сервер UDP», «Имя домена TCP» и «Имя домена UDP».

- Объект типа Телефон и Модем имеют настраиваемые параметры определяющие доступ и абонентский номер. Доступа «0» соответствует состоянию «не активен». Если включен режим авторизации (см. команду [&set.gsm](#)), то входящий вызов или SMS-сообщение принятое от такого *Пользователя* игнорируется. *Пользователь* считается зарегистрированным, если его номер телефона содержится в системном списке *Пользователей* (см. команду [&get.user](#));
- Объект типа IP-сервер или Домен (IP/TCP или IP/UDP) настроечными параметрами имеют: режим удержания GPRS- канала связи (включен/выключен), Web-сервер (да/нет), наличие HTTP- заголовка в ответе от Web-сервера (да/нет), IP-адрес или имя домена и номер IP-порта.

ВНИМАНИЕ! При включенном режиме авторизации (см. команду [&set.gsm](#)) *Устройство* игнорирует входящие звонки и SMS-сообщения, если телефонный номер абонента найден в системном списке и имеет состояние «не активен» или номер абонента не зарегистрирован. Если режим авторизации выключен и номер абонента, от которого принято сообщение (или вызов) не известен, то номер абонента может быть добавлен в список объектов *Пользователь* с состоянием «не активен».

4.12.1. &add.[<psw>].user=(type),[<mode>],[<address>],[<port>][,ResetFirst]

Добавляет объект типа *Пользователь* (OBJ_USER) в системный список. *Устройство* поддерживает работу с семью типами пользователей, указываемом в первом параметре **<type>**:

- 0** Номер беспроводной карты доступа (12 цифр от 0 до 9).
- 1** Номер телефона пользователя в формате **[+]<макс. 11 цифр номера телефона>**. Этот тип пользователей применяется для голосовой связи и отправки SMS-сообщений.
- 3** Номер телефона цифрового модема в формате **[+]<макс. 11 цифр номера телефона>**. Этот тип пользователей используется для связи с удаленным модемом и передачи/приема данных по каналу GSM CSD.
- 4** IP-адрес сервера с номером IP-порта. Этот канал связи может иницироваться *Устройством* для передачи/приема данных по протоколу TCP/IP. По умолчанию номер порта равен 80.
- 5** IP-адрес сервера с номером IP-порта. Этот канал связи может иницироваться *Устройством* для передачи/приема данных по протоколу UDP/IP. По умолчанию номер порта равен 80.
- 6** Имя домена (domain name) с номером IP-порта. Этот канал связи может иницироваться *Устройством* для передачи/приема данных по протоколу TCP/IP. По умолчанию номер порта равен 80.
- 7** Имя домена (domain name) с номером IP-порта. Этот канал связи может иницироваться *Устройством* для

передачи/приема данных по протоколу UDP/IP. По умолчанию номер порта равен 80.

Второй параметр **<mode>** целое число и его значение зависит от типа *Пользователя*.

Для *Пользователя* типа 0 (беспроводная карта доступа) этот параметр является не обязательным. *Устройство* может производить автоматическое добавление номера беспроводной карты при внесении карты в поле считывателя, если параметр *mode* равен «auto». Например:

```
> &add.user=0,auto
slide card now...
-468997001100
1
OK
>
```

Исполнение команды `&add.user=0,auto` сопровождается приглашением «slide card now...». После этого необходимо внести карту в поле считывателя. ТЕ вернёт построчно 12-ти значный номер карты, номер нового *Пользователя* и ОК.

- В отладочном режиме (см. третий параметр команды [&set.echo](#)) внесение беспроводной карты доступа в поле считывателя сопровождается возвратом ТЕ номера карты.

Для *Пользователя* типов 1 и 3 значение параметра **<mode>** определяет состояние *Пользователя* «включен/выключен». Значение 0 (ноль) соответствует состоянию «пользователь отключен». При включенном режиме авторизации (см. второй параметр команды [&set.gsm](#)) *Устройство* игнорирует входящие SMS-сообщение и вызов для такого *Пользователя*.

Для *Пользователя* типа 4 и выше значение параметра **<mode>** производит управление сеансом связи с IP-сервером. Назначение бит этого параметра:

- 01 значение “1” этого бита аналогично биту [08]. Этот бит использовался в прошивках версий до 2009 года.
 - 02 значение “1” этого бита указывает на то, чтобы после отправления сообщения ждать разрыва IP-соединения со стороны IP- сервера например, если это Web-сервер. По умолчанию этот бит устанавливается в единицу для порта 80.
 - 04 значение “1” этого бита указывает на то, что Web-сервер на принятый запрос методом HTTP GET не возвращает HTTP-заголовок.
 - 08 значение “1” этого бита включает режим «удержания» GPRS-соединения после разрыва связи с IP-сервером (по умолчанию 0).
- При отправлении сообщения на Web-сервер (порт 80 или иной порт со установленным битом 2 в значении параметра **<mode>**), *Устройство* может интерпретировать принимаемые от Web-сервера данные как &-команды, прошивку или пакет конфигурации (см. пример шаблона сообщения удовлетворяющего требованиям метода HTTP GET).

Третий параметр **<address>** определяет адрес пользователя согласно указанного типа в параметре **<type>**:

- номер беспроводной карты доступа (тип 0)
- номер телефона или модема в общепринятом формате 8<код города><семь цифр номера> или +7<10 цифр номера> (тип 1 и 3)
- IP-адрес сервера в формате XX.XX.XX.XX, где XX - это десятичное число от 0 до 255. Например: 64.148.0.12 (тип 4 и 5)
- имя домена в формате www.gwe.ru (тип 6 и 7)

Четвертый параметр **<port>** определяет номер IP-порта и может быть указан только для *Пользователей* типа 4 и выше. Номер IP-порта задается десятичным числом от 0 до 65535. Если номер порта не определен, то действует правило «по умолчанию» и номер порта принимается равным 16100 для пользователя типа 3 и 80 для пользователей типа 4 и выше.

Пятый параметр **ResetFirst** (или одна буква R) необязательный и может быть использован для предварительной очистки списка *Пользователей* перед добавлением нового *Пользователя*.

Список *Пользователей* может быть очищен, если задан параметр ResetFirst и в поле <address> или <port> обнаружена ошибка и команда вернула сообщение ERROR.

В случае успешного добавления *Пользователя* в список команда возвращает идентификационный номер нового *Пользователя* и **OK**. Нумерация *Пользователей* в системном списке производится автоматически при создании объекта и начинается с единицы.

Например:

```
> &add.user=1,1,80951112233
1
OK
> &add.user=3,1,+79161112244
2
OK
> &add.user=4,8,64.148.0.12,1080
3
OK
> &add.user=6,,mydomain.ru
4
OK
> &add.user=0,,468997001100
5
OK
>
```

В зависимости от версии прошивки (firmware) максимальное количество объектов *Пользователь* может быть ограничено 768 байтами. Длина третьего параметра <address> не может превышать 63 буквы/символа.

4.12.2. &get.[<psw>].user[={current|<user_id>}]

Команда возвращает список *Пользователей* из системного списка. Если указан параметр *current*, то команда вернет значения настроек для текущего *Пользователя*. Если задан параметр *<user_id>* (число соответствующее номеру *Пользователя* в списке), то команда вернет настройки *Пользователя* с указанным номером.

В завершении вывода команда возвращает OK или End, если список *Пользователей* пуст или *Пользователь* не найден.

Например:

```
> &get.user
user 1,1,1,+70951112233 user 2,8,,64.8.17.172,154
OK
> &ge.u=c
End
>
```

4.12.3. &del.[<psw>].user

Удаляет все объекты типа *Пользователь* из системного списка.

В завершении вывода команда возвращает OK.

Например:

```
> &del.user
OK
>
```

4.13. Работа с объектом Сообщение

Объект *Сообщение* определяет информацию, которая передается Пользователю или в последовательный интерфейс связи с внешним устройством или модемом. В зависимости от получателя передаваемая информация трактуется как «информационное сообщение», или «команда запроса данных», или «команда управления» внешним устройством.

Сообщение (или шаблон сообщения) – это объект *Устройства* представляемый текстовой ASCII-строкой содержания определяемого администратором. *Сообщение* может содержать буквы латинского алфавита, символы, цифры и специальные ключевые слова (*тэги*). Тэг автоматически заменяется на соответствующую информацию перед отправлением. Тэг начинается символом процент (%), за которым следуют две заглавные буквы латинского алфавита, определяющие его тип и содержание, либо три десятичные цифры.

Набор поддерживаемых тэгов определяется версией прошивки *Устройства*. Тэги разработанные под заказ в этом окументе не представлены.

Общий перечень *тэгов*:

- %SI** - идентификационный номер системы (см. команду [&set.id](#)).
- %VR** - версия *firmware* (например, 3.715). Тэг возвращает номер версии и сборки в одном из двух форматов в зависимости от прошивки. Например: «4.764 (build 1033)» или «4.764.1033».
- %RI** - идентификационный номер последней записи *журнала событий*
- %DA** - системная дата в формате день, месяц, год (DDMMYY).
- %LA** - географическая широта в диапазоне от 0° до 89° 59.9999
- %SN** - направление географической широты, север или юг
- %LO** - географическая долгота в диапазоне от 0° до 179° 59.9999
- %WE** - направление географической долготы, запад или восток
- %TI** - системное время в формате HHMMSS.
- %###** - цифровое значение кода ### (от 00 до 255).
- %U#** - значение объекта *Пользователь* с номером # (от 1 до 9 или «ноль» для текущего пользователя).
- %ZS** - текущее состояние *Зон* охраны (замкнуто/разомкнуто).
- %ZA** - состояние *Зон* охраны («тревога»).
- %AD** - состояния аналоговых входов в 16-ричном виде
- %CN** - состояния программных счетчиков в 16-ричном виде
- %CC** - контрольная сумма строки до тэга %CC (см. алгоритм вычисления контрольной суммы в «Приложение 5»).
- %GB** - возвращает значение баланса на момент инициализации GSM/GPRS модуля (отлык в GSM-версии *Устройства*).
- %HX** - 16-ричное представление структуры пакета данных последней валидной координаты. Возвращаемый пакет данных может содержать идентификатор *Устройства* (см. [&set.id](#)), если определён режим командой [&set.echo](#) (бит 80h третий параметр).
- %LC** - аналогично тэгу %HX, но с добавлением в конце пакета данных контрольной суммы (AA).
- %HC** - аналогично тэгу %HX, но с добавлением в конце пакета данных символа «звездочка» и контрольной суммы (*AA).
- %HR** - 16-ричное представление текущего пакета данных с добавлением в конце пакета данных контрольной суммы (AA)
- %HH** - 16-ричное представление структуры текущего пакета данных с добавлением в конце пакета данных символа «звездочка» и контрольной суммы (*AA). Возвращаемый пакет данных может содержать идентификатор *Устройства* (см. [&set.id](#)), если определён режим командой [&set.echo](#) (бит 80h третий параметр)
- %BR** - двоичное представление структуры текущего пакета данных с добавлением в конце пакета данных байта контрольной суммы. Возвращаемый пакет данных может содержать идентификатор *Устройства* (см. [&set.id](#)), если определён режим командой [&set.echo](#) (бит 80h третий параметр).
- %LR** - 16-ричное представление последнего записанного пакета данных в ЛОГ с контрольной суммой (AA). Аналогично исполнению команды [&req.last=1.x](#)
- %RC** - 16-ричное представление <N>-последних «новых» записей

ЛОГ следующих в порядке возрастания их порядковых номеров (значение <N> устанавливается командой [&set.sys](#)) с ожиданием ключевого слова "OK<Enter>", подтверждающим получение данных. Время ожидания подтверждения устанавливается командой [&set.sys](#). См. также описание события [OBJ_SYSTEM.1.SYS_TIMEOUT](#). Команда [&req](#) или повторный возврат сообщения с тэгом %RC сбрасывает режим ожидания подтверждения. При отсутствии «новых» записей в ЛОГ тэг %RC возвращает текущий пакет данных аналогично работе тэга %HR.

- %NC** - аналогично тэгу %RC, но возврат записей ЛОГ производится в порядке возрастания их порядковых номеров.
- %LN** - количество символов следующих за тэгом %LN
- %LF** - символ перевод строки (код 0Ah)
- %CR** - символ возврат каретки (код 0Dh)
- %SM** - номер текущей SIM-карты (1 или 2, только в версии *Устройства* с поддержкой двух SIM-карт)
- %IO** - NMEA предложение формата PGIO
- %GG** - NMEA предложение формата GGA
- %RM** - NMEA предложение формата RMC
- %HD** - фактор ухудшения точности плановых координат (HDOP)
- %HG** - высота антенны приёмника над/ниже уровня моря (HGEO)
- %HM** - геоидальное различие - различие между земным эллипсоидом WGS-84 и геоидом (HMET)
- %SU** - количество используемых спутников (00-12, может отличаться от числа видимых)
- %CO** - истинное направление курса в градусах (COG)
- %SP** - скорость над поверхностью в узлах (SOG in knots)
- %AS** - среднее значение скорости с момента последней записи в ЛОГ
- %CH** - 16-ричное представление структуры 3.1 (см. Приложение 3.1 в этом документе)
- %CN** - значения программных счетчиков разделенных запятой
- %GO** - значение гео-одометра
- %II** - IMEI модема (или «нули», если модем не включен)
- %SM** - номер текущей СИМ-карты (только в версии с двумя СИМ-держателями)
- %NR** - количество "новых" записей в ЛОГ
- %LS** - размер пакета данных в байтах. Пакет данных в 16-ричном представлении имеет размер в 2 раза больше. При этом, добавление контрольной суммы увеличивает размер пакета еще на два. Например, размер пакета данных возвращаемый тэгом %HC будет равен $\%LS * 2 + 2$ символов.
- %NS** - количество байт передаваемых тэгом %RC.
- %OP** - 16-ричное представление включенных опций прошивки (см. [&get.sys](#)).
- %OF** - 16-ричное представление значений пользовательских флагов (см. описание объекта [OBJ_FLAGS](#)).
- %CH** - 16-ричное представление структуры конфигурации.
- %R#** - 16-ричное представление значения пользовательского

регистра с индексом # (одна или две десятичных цифры). Значение индекса # должно быть в пределах от 0 до максимального количества пользовательских регистров минус один (например, %R15)

%P# - десятичное представление IP-порта *Пользователя* с номером # (см. команду [&get.user](#)). Номер *Пользователя* # представляется десятичной цифрой от 1 до 9.

%PC - десятичное представление IP-порта текущего *Пользователя*.

%U# - IP-адрес *Пользователя* с номером #. Номер *Пользователя* # представляется десятичной цифрой от 1 до 9.

%UC - IP-адрес текущего *Пользователя*.

%RN - количество пользовательских регистров созданных командой [&add.log](#)

%RS - размер пользовательских регистров созданных командой [&add.log](#)

Представление данных возвращаемых тэгами %RC, %LS, %NR и %NS зависит от настроек третьего параметра команды [&set.echo](#) (может быть 16-ричным или двоичным). Передача байт следует правилу «старший вперёд».

Например, используя следующий шаблон сообщения

```
Station No %SI - ALARM ZONE 1 at %TI:%DA
```

адресат примет сообщение

```
Station No 1001 - ALARM ZONE 1 at 235100:050405
```

где время в формате HHMMSS и дата – DDMMYY.

В отличие от принципов работы с объектами *Таймер*, *Макро* и *Условие* работа с объектом *Сообщение* имеет сокращенный набор команд: добавить (add), вернуть (get) и удалить (delete). В связи с этим, для редактирования (изменения) значения параметра какого-либо объекта *Сообщение*, необходимо предварительно удалить все объекты (команда &del), а затем создать новые объекты с требуемыми параметрами (команда &add).

Для того чтобы отправить сообщение необходимо определить получателя (см. команды работы с объектом *Пользователь*). В зависимости от типа *Пользователя*-получателя Устройство будет использовать соответствующий канал передачи данных SMS, CSD или GPRS. Важно заметить, если сообщение посылается на Web-сервер (см. бит 2 второго параметра команды [&add.user](#)), то шаблон сообщения должен следовать требованиям метода HTTP GET или PUT (см. пример ниже), а скрипт принимающий запрос с параметрами должен отвечать ключевым словом OK или OK с последующей цифрой. При отправлении сообщения по каналу GPRS *Пользователям* отличным от Web-сервер *Устройство* разрывает соединение с сервером сразу по окончании передачи сообщения без ожидания ответа от сервера.

Пример шаблона сообщения для передачи RMC-предложения (тэг %RM) *Пользователю* 1 (тэг %U1) на Web-сервер и отвечающего требованиям метода HTTP GET:

```
GET /cgi-bin/pos.php?id=%SI&data=%RM HTTP/1.0%LFHost: %U1%LF%LF
```


4.13.1. &add.[<psw>.]message=<message>

Добавляет объект типа [Сообщение](#) в системный список. Параметром объекта является текст с ASCII символами, буквами, цифрами и/или знаками.

Команда возвращает идентификационный номер нового сообщения в системном списке и ОК. Нумерация сообщений начинается с единицы.

Пример 1:

```
> &add.me=Alarm, main door (zone 1)
1
OK
>
```

Пример 2 (сообщение соответствующее требованиям метода HTTP GET):

```
> &add.me=GET /test.php?data=%RC HTTP/1.0%LFHost: 10.10.10.10%LF%LF
2
OK
>
```

ВНИМАНИЕ! Общая длина всех сообщений не может превышать 1024 символа (зависит от версии firmware) из расчета, что длина одного сообщения составляет <количество символов в сообщении> + 1.

4.13.2. &get.[<psw>.]message[=?]

Возвращает строки со значениями объектов типа *Сообщение* из системного списка в порядке их следования (формат: **msg <id>,<message>**). Строка заканчивается символами «переход в начало строки» и «переход на след. строку».

Команда с заданным параметром «?» или «s» (string) возвращает форматированные сообщения с заменой *тэгов* на соответствующие данные на момент исполнения команды. Параметр ? (или S) может, например, использоваться с целью отладки сообщений перед отправкой сообщения *Пользователю*.

Например:

```
> &get.message
msg 1,Elevator broken on %SI
msg 2,Alarm main door (zone 1) at %DA:%TI OK
>&get.message=?
msg 1,Elevator broken on 1001
msg 2,Alarm main door (zone 1) at 190705:231420
OK
>
```

4.13.3. &del.[<psw>.]message

Команда удаляет все ранее созданные сообщения.

Возвращает ОК.

Например:

```
> &del.message  
ОК  
>
```

4.14. Команды работы с ключевыми словами

Ключевые слова помогают настраивать *Устройство* для работы с внешними контроллерами и создавать пользовательский командный интерфейс защищённый паролем.

При получении строки по командному интерфейсу начинающейся с заданного ключевого слова может быть выработано системное событие OBJ_KEY.<key_id>.KEY_FOUND, где <key_id> номер найденного ключевого слова в списке ключевых слов.

Максимальное количество ключевых слов определяется их общим объемом 255 байт из расчета <длина_ключевого_слова>+2 байта для одного ключевого слова.

Сравнение принятой последовательности и ключевого слова чувствительно к регистру.

Ключевое слово может содержать тэг, определяющий наличие некоторого цифровой величины, значение которой необходимо преобразовать и записать в LOG или сравнить с паролем системы. Тэг начинается символом «процент» с последующими двумя буквами и числом: **%{B|W|D|P}{I|H}<item>**. Первая буква определяет допустимый диапазон значений «захватываемой» величины для ее записи в ЛОГ (B, W или D) или сравнения с системным паролем (P):

- B** (Byte) «захват» одного байта
- W** (Word) «захват» слова (два байта)
- D** (Double) «захват» двойного слова (четыре байта)
- P** (Double) сравнение числа с системным паролем ([&set.psw](#))

Вторая буква (I или H) определяет форму представления «захватываемой» величины:

- I** (unsigned Int) десятичное представление (0..9)
- H** (unsigned Hex) 16-ричное представление (0..9, A..F)
- <item>** десятичное число, определяющее порядковый номер величины в последовательности символов принятой строки (нумерация начинается с нуля).

Критерием выбора первой буквы (B, W или D) тэга является диапазон чисел интересующей величины. Если значение величины лежит в диапазоне от 0 до

25.4, то первая буква тэга должна быть В (byte). Если в диапазоне от 0 до 6553.5 – буква W (word). Если в диапазоне от 0 до 429496729.5 – буква D (double).

Тэг %P# или %P# (где # десятичная цифра от 0 до 9) используется для «захвата» числа и сравнения его с системным паролем определённым командой [&set.psw](#). Системный пароль в это случае должен иметь числовое представление в диапазоне от 0 до 99999999. Системное событие OBJ_KEY.<key_id>.KEY_FOUND будет иметь место только в том случае, если значение системного пароля совпало с «захваченным» числом.

Пример 1. При необходимости «захвата» 16-ричного значения величины (в примере выделено красным) входящей в состав предложения передаваемого датчиком уровня жидкости ДУЖ-12:

```
F=F164!U=0000 t=7A D32=00000000 N=030A.1
```

ключевое слово может иметь следующее представление:

```
F=%WH6
```

Ниже представлено пояснение нумерации 16-ричных величин к данному примеру:

```
F=F164!U=0000 t=7A D32=00000000 N=030A.1
| | | | | | |
0 1 2 3 4 5 6
```

Пример 2. «Захват» значения расхода топлива, передаваемого датчиком расхода ПОРТ-1:

```
Время работы: 1234-12
Текущий расход: 12340 г/ч
Полный расход: 12345678 г
```

можно использовать ключевое слово с тэгом %DI3:

```
Время работы%D13
```

Пример 3. «Захват» значения текущего баланса, возвращаемого от внешнего модема:

```
+CUSD: 0,"Vash balans: 6.92 E. Blagodarim za svoevremennuju oplatu..."
```

можно использовать ключевое слово с тэгом %W11:

```
+CUSD: %W11
```

При использовании тэга в составе ключевого слова размер структуры LOG должен быть увеличен на соответствующее число байт (см. команду [&add.log](#)).

Пример 4. Управление *Устройством* по каналу SMS, CSD или GPRS защищенное паролем. Ключевое слово должно содержать тэг %P# (или %P#):

```
> &add.key=OPEN DOOR%PI0
1
OK
> &add.key=CLOSE DOOR%PH3
2
OK
>
```

В случае предопределённого пароля (например, 1111) и получением строки по командному интерфейсу будет произведено системное событие KEY_FOUND только для входящих строк (команд):

OPEN DOOR 1111
CLOSE DOOR 457

событие OBJ_KEY.1.KEY_FOUND
событие OBJ_KEY.2.KEY_FOUND

Внимание! Тэг входящий в состав ключевого слова должен завершать ключевое слово. Например: PARAM=%WIO

Не разрешается использовать ключевые слова полностью или частично совпадающие с зарезервированными словами представленными в Приложении 8 этого документа, включая ключевое слово OKxx.

4.14.1. &add.[<psw>.]key=<keyword>

Добавляет новое ключевое слово в список *ключевых слов*. Команда возвращает порядковый номер (индекс) нового ключевого слова и ОК в случае успешного добавления ключевого слова в список или ERROR если превышено допустимый объём ключевых слов.

Например:

```
> &add.key=CONNECTION DONE
1
OK
>
```

4.14.2. &del.[<psw>.]key

Удаляет все ранее созданные ключевые слова.

Возвращает ОК.

Например:

```
> &del.key
OK
>
```

4.15. Команды тестирования и отладки системы

4.15.1. &erase.[<psw>.]cfg

Очищает блок конфигурации во FLASH Очистка блока конфигурации производится без удаления текущих настроек в оперативной памяти (объекты *Макро, Условие, Таймер*).

После перезапуска *Устройство* будет работать с настройками «по умолчанию» (см. [Приложение 2. Карта FLASH и настройки по умолчанию](#)).

4.15.2. &erase[.<psw>.]log

Очищает всё пространство FLASH, соответствующее журналу событий (LOG).

Внимание! После удаления всех записей в журнале событий нумерация новых записей ЛОГ начинается с единицы.

4.15.3. &restart[.<psw>]

Команда инициирует программный перезапуск *Устройства* (SoftOn). При перезапуске *Устройство* вырабатывает все сигналы аппаратной и программной инициализации своих компонент, Flash, GPS, GSM и настройку *Объектов* (аналогично включению питания).

Например, в версии *Устройства* с поддержкой ГЛОНАСС/GPS приёмника:


```
&restart

Init glonass... gsm.....

GELIX-2 v4.747
/8IO/4A/GSM/2SIM/RTC/WD/CNT/upgr id 00001001
psw no
log x60,valid,0,4,0 echo x5,off,15
last id 1 size 28,57 flash 523648 free heap 1379
glo 4800,poll 1s,06
gsm pwrON,pinNO,lnLOC,ipNO,06,,1.0 user 1
msg 1,0 macro 3 timer 1 condition 1

>
```

Устройство перезапущено.

 В версии firmware с поддержкой Bluetooth адаптера исполнение команды &restart может сопровождаться выводом в порт *TE* служебных команд для настройки Bluetooth адаптера.

4.15.4. &dump[.<psw>]=<page>

Возвращает содержимое страницы FLASH, номер которой указывается в первом параметре.

В одной строке TE отображает шестнадцать байт в 16-ричном и текстовом виде.

Например:

```
> &dump=0
0000:00 00 00 00 00 00 00 00 00 31 2E 31 2E 30 54 00 00 : .....1.1.0T..
0000:01 00 00 00 00 04 D2 00 0A 00 05 31 71 31 00 00 00 : .....1q1...
0000:02 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF : .....
0000:03 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF : .....
0000:04 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF : .....
0000:05 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF : .....
0000:06 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF : .....
0000:07 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF : .....
0000:08 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF : .....
0000:09 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF : .....
0000:10 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF : .....
0000:11 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF : .....
0000:12 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF : .....
0000:13 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF : .....
0000:14 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF : .....
0000:15 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF : .....
OK
>
```

Адрес первого отображаемого байта выводится в первой колонке слева. Адрес разделён двоеточием на две части. Первая часть адреса соответствует номеру страницы (256 байт). Вторая часть – смещение относительно начала страницы.

4.15.5. &test.[<psw>.]watchdog

Эмулирует эффект «зависания» firmware специальным запуском бесконечного цикла типа:

```
while(1) { }
```

Системный Watchdog *Устройства* производит автоматический перезапуск системы (system_reset) через определённое время (см. Спецификацию) и восстанавливает нормальный режим работы с настройками, ранее сохранёнными командой **&set.save**.

4.16. Настройка работы Устройства с навигационным приёмником

4.16.1. &set.[<psw>].nmea=[<mask >],[<baud_index>],[<pwrMode>],[<hdopMax>]

Настраивает формат сообщений, выводимых навигационным приёмником, и изменяет настройки Порт А Устройства (скорость обмена порта связи с навигационным приёмником). Оба параметра не являются обязательными.

✎ Действие параметров <mask> и <pwrMode> не распространяется при работе с внешним навигационным приёмником (например, ГЛОНАСС/GPS).

<mask> - определяет NMEA форматы данных, достаточные для определения факта того, что все навигационные данные получены Устройством. Значение представляется двумя 16-ричными символами (0..9, A..F), позиции битов которого соответствуют NMEA-предложениям возвращаемых навигационным приёмником. Младшие три бита (под маской 0Eh) определяют набор NMEA-предложений передаваемых навигационным приёмником. Значение 1 в старшем бите (под маской 80h) определяет режим работы с внешним навигационным приёмником (например, ГЛОНАСС/GPS-приёмником Геликс-GL):

NMEA	Значение параметра <mask>
GGA	02
RMC	04
GLO	80

Например, для включения только одного предложения RMC и работе с приёмником GPS значение параметра <mask> должно быть “04”. По умолчанию значение <mask> равно “06” (GGA и RMC).

✎ Параметр <mask> необходимо использовать только для согласования используемого навигационного приёмника с Устройством, даже если навигационный приёмник не поддерживает команд SiRF.

☞ Устройство принимает NMEA предложения до тех пор, пока не будут получены все необходимые форматы предложений, которые определены параметром <mask>. По факту получения указанного комплекта NMEA-предложений Устройство генерирует событие OBJ_GPS.<status>.GPS_RECEIVED.

<baud_index> - индекс скорости обмена последовательного порта с навигационным приёмником:

Индекс	Скорость обмена, бит/сек
2	1200
3	2400
4	4800
5	9600
6	14400
7*	19200

* использовать не рекомендуется.

Пример команды для установки скорости обмена порта GPS в 9600 бит/сек:

```
> &set.nmea=,5  
OK  
>
```

Изменение параметров порта всязи с навигационным приёмником может привести к потере приёма навигационных данных и, как следствие, прекращению записи данных в *журнал событий* и возврата данных *TE*.

<pwrMode> - производит управление режимом питания встроенного навигационного модуля. Допустимые значения: 0 (выключить) или 1 (включить). По умолчанию навигационный модуль включен.

<hdop> - устанавливает пороговое значение уровня горизонтального дифференциального отклонения, который принимается от навигационного приёмника данных. По умолчанию значение **<hdop>** равно 255 (максимальное значение). Данные принятые от навигационного приёмника с действительным значением HDOP превышающим **<hdop>** считаются невалидными.

Таблица для оценки пороговых значений HDOP:

HDOP	Точность	Описание
1	Идеальная	Рекомендуется к использованию в системах, требующих максимально возможную точность во всё время их работы.
2-3	Отличная	Достаточная точность для использования результатов измерений в достаточно чувствительной аппаратуре и в программах.
4-6	Хорошая	Рекомендуемый минимум для принятия решений по полученным результатам. Результаты могут быть использованы для достаточно точных навигационных указаний.
7-8	Средняя	Результаты можно использовать в вычислениях, однако рекомендуется озаботиться повышением точности, например, выйти на более открытое место.
9-20	Ниже среднего	Результаты могут использоваться только для грубого приближения местоположения.
21-50 и более	Плохая	Выходная точность ниже половины футбольного поля. Обычно такие результаты должны быть отброшены.

Практика показывает, удовлетворительным геометрическим фактором является выполнение условия GDOP < 10.

4.16.2. &get.[<psw>].nmea

Команда возвращает текущие настройки GPS-модуля, порта связи и режим обработки данных:

- Скорость обмена по порту GPS;
- Период возврата NMEA-предложений навигационным приемником;
- Маска (16-ричное число) используемых NMEA-предложений. Определяет типы NMEA-предложений, необходимых для определения того факта, что все навигационные данные приняты *Устройством* успешно.

Пример исполнения команды в версии *Устройства* с поддержкой GPS-приёмника:

```
> &get.nmea
gps 4800,poll 1s,06,0
OK
>
```

Пример исполнения команды в версии *Устройства* с поддержкой ГЛОНАСС/GPS-приёмника:

```
> &get.nmea
glo 4800,poll 1s,06,0
OK
>
```

Подробнее см. описание команды [&set.nmea](#).

4.17. Работа с GSM

(GSM)

Описание настроек точки выхода в GPRS-сеть представлено в параграфе [4.17.3. Настройка точки входа в GPRS-сеть](#) ниже.

В версии *Устройства* с GSM-модулем типа GR64 можно ускорить работу интерфейса SIM-карты с помощью команды ^AT+WFM=1,"SIMSPEEDENH". Для некоторых SIM-карт старых образцов GSM-модуль может не поддерживать работу с SIM-картой в этом режиме. В этом случае требуется вернуть настройки в обычный режим командой ^AT+WFM=0,"SIMSPEEDENH". Произведенные изменения необходимо сохранить командой ^AT&W. Вернуть текущие настройки этого режима можно командой ^AT+WFM=2,"SIMSPEEDENH". Например:

> ^AT+WFM=1,"SIMSPEEDENH"	включить "ускоренный" режим
OK	
> ^AT+WFM=2,"SIMSPEEDENH"	вернуть тек. режим
+WFM: "SIMSPEEDENH",1,1	режим "ускорения" включен
OK	
> ^AT+WFM=0,"SIMSPEEDENH"	выключить "ускоренный" режим
OK	
> ^AT+WFM=2,"SIMSPEEDENH"	вернуть тек. режим
+WFM: "SIMSPEEDENH",0,1	режим "ускорения" выключен
OK	
>	

4.17.1. &get.[<psw>.]gsm[=imei]

Возвращает режимы работы модуля GSM согласно таблице ниже. Если определен параметр `imei`, то в версии прошивки с поддержкой тэга `%I` команда вернет IMEI встроенного GSM-модуля.

№	Комментарий	Значение
1	Состояние GSM/GPRS-модуля	Выключен pwrOFF
		Включен pwrON
2	Состояние SIM PIN	SIM-карта не защищена pinNO
		SIM-карта защищена, PIN не верен pinERR
		SIM-карта защищена, PIN верен pinYES
3	Состояние GSM	Оператор не найден InOFF
		Локальный оператор InLOC
		В зоне роаминга InROA
4	Состояние IP	Нет соединения ipNO
		Соединение установлено ipYES
5	Режим обработки входящего вызова и SMS	<mode>
6	Состояние GPRS*	Не активен
		Активен GPRS
7	Баланс	Десятичное число с точностью до 1/10 xx.x

* Значения нет при не активном соединении к сети GPRS.

<mode> 16-ричное число соответствует режиму обработки входящего вызова и SMS-сообщения. Режим **<mode>** представляется одним байтом по битам:

- 01 управление режимом авторизации входящего голосового вызова (CHECK_USER_CALL). Если бит установлен, то номер вызывающего абонента будет проверяться на наличие в списке объектов *Пользователь*. Если номер абонента найден в списке *Пользователей* с состоянием «включен», то производится обработка вызова. Если номер не найден или состояние найденного *Пользователя* равно «отключен», то принятый вызов будет игнорирован и сброшен (по умолчанию бит 01h не установлен).
- 02 управление режимом SMS_COMMAND или SMS_FWRD2TE. Если бит установлен, то входящее SMS-сообщение будет интерпретироваться как &-команда (перечень команд представлен в этом разделе). Если данный бит не установлен (режим SMS_FWRD2TE), то текст принятого SMS-сообщения будет передан в порт *TE* (по умолчанию бит 02h установлен).
- 04 управление режимом SMS_REPORT. Если этот бит установлен, то после получения &-команды от абонента по SMS-каналу и ее исполнения результат исполнения &-команды возвращается абоненту по SMS (по умолчанию бит 04h установлен).
- 08 завершать передачу сообщения *TE* символами завершения и перехода на новую строку (CR+LF). Этот режим работает в

- режиме SMS_FWRD2TE (см. описание бита 02h выше).
- 10 включение режима авторизация принятых SMS-сообщений CHECK_USER_SMS (по умолчанию бит 10h не установлен).
 - 20 (бит не используется)
 - 40 добавлять телефонный номер нового абонента в список *Пользователей* при входящем вызове или SMS-сообщении (по умолчанию бит 40h не установлен). При этом режим авторизации должен быть включен (см. описание бита 01h выше).
 - 80 «ускоренный» режим работы GSM-модема по последовательному интерфейсу с SIM-картой (режим FASTSIM или <fs>, только для модема GR64).

Сочетание нескольких режимов равно сумме соответствующих кодов (01+04=05 или 04+08=0C). По умолчанию режим обработки входящих SMS сообщений равен “2” (интерпретировать SMS-сообщение как команду).

Пример 1 (PIN не определён, режимы обработки SMS 01 + 02, контроль баланса отключен):

```
> &get.gsm
pwrON,pinNO,lnLOC,ipNO,03,, -
OK
>
```

Пример 2 (тоже, но GPRS активен, баланс 12.8 единиц):

```
> &get.gsm
pwrON,pinNO,lnLOC,ipNO,83,GPRS,12.8
OK
>
```

Пример 3:

```
> &get.gsm=imei
```

4.17.2. &set.[<psw>].gsm=[“<pin>”],[<mode>],[<ussd0>],[<ussd1>],[<fs>],[<copsId>]

Устанавливает режимы работы встроенного модуля GSM/GPRS-связи.

Параметр **<pin>** должен быть установлен в соответствии с SIM PIN используемой в Устройстве СИМ-карты.

ВНИМАНИЕ! В версии *Устройства* с двумя SIM-держателями обе SIM-карты должны иметь одинаковый PIN- код. *Устройство* проверяет установленный PIN-код в процессе инициализации модема. *Устройство* не производит изменений PIN-кода SIM-карты. Чтобы изменить PIN SIM-карты, необходимо использовать мобильный телефон или соответствующие АТ-команды [9].

Если параметр **<pin>** не указан или длина его значения менее четырёх символов, то PIN код не производит авторизации установленной SIM-карты.

При изменении SIM PIN (команда [&set.gsm](#)) новое значение **<pin>** автоматически сохраняется в энергонезависимой памяти *Устройства*.

Параметр **<mode>** определяет режим обработки входящих SMS-сообщений (см. описание команды [&get.gsm](#)).

Параметр **<ussd0>** определяет номер сервиса GSM-оператора связи возвращающего текущий баланс для основной SIM-карты. Параметр **<ussd1>** используется для дополнительной SIM-карты. Значение параметра **<ussd_>** может быть в диапазоне от 0 до 99. При этом USSD-запрос будет посылаться GSM-оператору каждый раз при инициализации GSM/GPRS-модуля на номер ***1xx#**, где **xx** – число определённое в параметре **<ussd_>**. Например, для отправления USSD запроса на номер ***102#** (основная SIM-карта) необходимо исполнить команду:

```
> &set.gsm=,,2
OK
>
```

Для того чтобы отключить автоматическое определение баланса необходимо исполнить команду [&set.gsm](#) со значением 128 в параметре **<ussd>**.

Например (отключение контроля баланса для дополнительной SIM-карты):

```
> &set.gsm=,,,128
OK
>
```

Параметр **<fs>** (Fast SIM) поддерживается только для *Устройств* версии с модемом GR64 и позволяет включить или отключить «быстрый» режим работы модема с SIM-картой. По умолчанию «быстрый» режим работы с SIM-картой отключен (значение параметра **<fs>** равно “0”). Работа с некоторыми SIM-картами может сопровождаться ошибкой при включении устройства и инициализации модема. В этом случае необходимо исполнить команду:

```
&set.gsm=,,,1
```

Параметр **<copsld>** определяет номер шаблона *Сообщения* содержащее имя (имена) оператора GSM-связи, при подключении к которому сбрасывается флаг-индикатор «роуминг» (см. описание констант [Условие](#) для объекта [OBJ GSM](#)).

Например, сеть Билайн-GSM в Калининградской области (код сети 250-28) может именоваться «Extel-GSM», «Rus-Extel» или «Rus-28». В этом случае следует определить шаблон *Сообщения* ([&add.message](#)) и произвести привязку этого сообщения к модулю GSM ([&set.gsm](#)):

```
> &add.message=EXTEL-GSM,RUS-EXTEL,RUS-28
2
OK
> &set.gsm=,,,2
OK
>
```

При подключении к GSM-сети *Устройство* будет производить запрос имени оператора связи и, получив имя оператора, поиск первого слова имени (до знака пробел) в тексте объекта *Сообщения* с номером **<copsld>**. Если слово не найдено, то устанавливается бит ROAM объекта [Условие](#) для [OBJ GSM](#).

4.17.3. Настройка точки входа в GPRS-сеть

Для успешного подключения к сети GPRS необходимо знать точку доступа (GPRS APN), имя пользователя (login) и пароль (password) в соответствии с оператором связи, который предоставляет услугу доступа к сети GPRS.

✍ Для настройки доступа к сети GPRS обратитесь в службу поддержки оператора связи и произведите настройки командами AT+CGDCONT и AT*ENAD как описано ниже.

Таблица параметров GPRS настроек на Январь 2011 г.

Управление доступом в сеть GPRS			
Оператор GSM	Точка доступа	Имя	Пароль
Beeline	internet.beeline.ru	beeline	beeline
Megafon Москва	internet.msk	gdata	gdata
MTS	internet.mts.ru	mts	mts

Для подключения к сети GPRS в зависимости от используемой SIM-карты (оператора связи) необходимо произвести настройки встроенного GSM/GPRS-модуля с помощью AT-команд.

GSM/GPRS-модуль сохраняет настройки автоматически после исполнения AT-команды. Примеры настроек APN первой SIM-карты для операторов связи в г.Москва на март 2007 года:

BeeLine (SIM1)

```
^AT+CGDCONT=1,"IP","internet.beeline.ru"  
^AT*ENAD=1,"GPRS1","beeline","beeline",1,0
```

для версии с модулем GR64:

```
^AT+CGDCONT=1,"IPV4","internet.beeline.ru"  
^AT*ENAD=1,"GPRS1","beeline","beeline",1,0
```

Megafon Москва (SIM1)

```
^AT+CGDCONT=1,"IP","internet.msk"  
^AT*ENAD=1,"GPRS1","gdata","gdata",1,0
```

для версии с модулем GR64:

```
^AT+CGDCONT=1,"IPV4","internet.msk"  
^AT*ENAD=1,"GPRS1","gdata","gdata",1,0
```

MTS (SIM1)

```
^AT+CGDCONT=1,"IP","internet.mts.ru"  
^AT*ENAD=1,"GPRS1","mts","mts",1,0
```

для версии с модулем GR64:

```
^AT+CGDCONT=1,"IPV4","internet.mts.ru"  
^AT*ENAD=1,"GPRS1","mts","mts",1,0
```

В версии *Устройства* с двумя SIM-картами настройка GPRS APN первой SIM-карты производится аналогично командам описанным выше. Точка доступа в GPRS-сеть для второй SIM-карты производится заменой цифры «1» в AT-командах на цифру «2».

Пример настроек APN второй SIM-карты для операторов связи в г.Москва на март 2007 года:

BeeLine (SIM2)

```
^AT+CGDCONT=2,"IP","internet.beeline.ru"  
^AT*ENAD=2,"GPRS2","beeline","beeline",1,0
```

или для версии с модулем GR64:

```
^AT+CGDCONT=2,"IPV4","internet.beeline.ru"  
^AT*ENAD=2,"GPRS2","beeline","beeline",1,0
```

Megafon Москва (SIM2)

```
^AT+CGDCONT=2,"IP","internet.msk"  
^AT*ENAD=2,"GPRS2","gdata","gdata",1,0
```

или для версии с модулем GR64:

```
^AT+CGDCONT=2,"IPV4","internet.msk"  
^AT*ENAD=2,"GPRS2","gdata","gdata",1,0
```

MTS (SIM2)

```
^AT+CGDCONT=2,"IP","internet.mts.ru"  
^AT*ENAD=2,"GPRS2","mts","mts",1,0
```

или для версии с модулем GR64:

```
^AT+CGDCONT=2,"IPV4","internet.mts.ru"  
^AT*ENAD=2,"GPRS2","mts","mts",1,0
```

Несоблюдение порядка и значений цифр (=1 или =2, "GPRS1" или "GPRS2") приведёт к ошибке подключения к сети GPRS и, следовательно, невозможности передачи данных по каналу GPRS.

Для версий *Устройств* с модемом GR64 необходимо произвести сохранение произведенных настроек исполнением команды **^AT&W**.

Проверить текущие настройки точек доступа к сети GPRS можно командами:

```
^AT+CGDCONT?  
^AT*ENAD?
```

Например:

```
> ^AT+CGDCONT?  
+CGDCONT: 1,"IP","internet.mts.ru","",0,0  
> ^AT*ENAD?  
*ENAD: 1,"GPRS 1","mts",,1,0,0,1,0  
>
```

Дополнительно, см. параграф [AT-команды](#) в этом документе.

4.17.4. &get.[<psw>].temperature[={B|G|1}]

Возвращает текущее значение температуры в градусах Цельсия от датчика температуры установленного на плате, внутри GSM/GPRS-модуля (только в GSM-версии *Устройства* с модемом типа GR64) или внешнего датчика «Геликс-TS». Значение параметра команды определяет опрашиваемый датчик температуры:

- BOARD** температуры на поверхности платы;
- GSM** температура внутри модема (опция);
- 1** внешний датчик «Геликс-TS».

По умолчанию (если параметр не задан) команда возвращает значение температуры от датчика «Геликс-TS».

Пример 1 (вернуть значение температуры внутри GSM/GPRS-модуля):

```
> &get.te=gsm
28
ОК
>
```

Пример 2 (вернуть значение температуры на поверхности платы):

```
> &get.te=board
28.7
ОК
>
```

Значение температуры от датчика «Геликс-TS» возвращаются четырёхзначным 16-ричным числом, значение бит которого в зависимости от типа датчика определяют:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
n/a						E	S	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰	2 ⁻¹

где

- E** бит ошибки (равен «1», если датчик не подключен или неисправен);
- S** знаковый бит (равен «1», если значение меньше нуля);
- n/a** старшие биты 10..15 информации не несут и равны нулю.

Пример 3 (вернуть значение температуры от датчика «Геликс-TS»):

```
> &get.te
0037
ОК
>
```

Значению 0037h соответствует температура 27.5°C (см. таблицу преобразования ниже).

Для вычисления абсолютного значения температуры при S=1 над целой частью (биты от 2⁶ до 2⁰) необходимо выполнять операции «побитового инвертирования плюс 1».

Значение дробной части результата (.5 или .0) определяется значением младшего бита («1» или «0» соответственно).

- Тип цифрового датчика температуры определяется командой [&set.system](#).

Примеры вычисления значения температуры для «Геликс-TS»		
Температура	Двоичное представление	16-ричное представление
+85.0°C	0000 0000 1010 1010	00AA
+25.0°C	0000 0000 0011 0010	0032
+0.5°C	0000 0000 0000 0001	0001
0.0°C	0000 0000 0000 0000	0000
-0.5°C	0000 0001 1111 1111	01FF
-25.0°C	0000 0001 1100 1110	01CE
-55.0°C	0000 0001 1001 0010	0192
n/a	0000 0010 0000 0000	0200

Команда **&get.temperature=gsm** не исполняется и вернёт BUSY, если GSM/GPRS-модуль в процессе инициализации, исполнения команды или установки сеанса связи по каналу GPRS, IP или CSD.

- Показания температуры на поверхности платы могут иметь более высокое значение по сравнению с температурой внутри GSM/GPRS-модуля.

4.18. Работа с Гео-Зонами

- Работа с *Гео-Зонами* относится к дополнительным возможностям и может не поддерживаться в некоторых версиях *firmware Устройства*.

Максимальное количество *Гео-Зон* определяется версией *firmware* и по умолчанию равно 30.

4.18.1. &add.[<psw>].geo=<N>

Создает/добавляет указанное количество объектов *Гео-Зона* в системный список. Вновь созданный объект отключен (disabled) и имеет нулевые значения во всех параметрах. Команда возвращает ОК если нет ошибки.

Например:

```
> &add.geo=2
2
OK
>
```


Если при исполнении команды текущее количество *Гео-Зон* плюс параметр <N> будет превышать максимально-допустимое количество *Гео-Зон*, то команда вернёт ERROR.

4.18.2. &get.[<psw>].geo

Возвращает текущие значения для всех объектов *Гео-Зона* в формате:

geo <id>:<enabled>,<latitude>,<longitude>,<radius>,C

где

- <id> - идентификационный (порядковый номер) объекта;
- <enabled> - состояние объекта: включено (1) или выключено (0);
- <latitude> - географическая широта (0000.0)
- <longitude> - географическая долгота (00000.0)
- <radius> - радиус зоны в минутах и десятых долях минуты (00.0)
- C - флаг «формы» гео-зоны: (C) круг, (R) квадрат

Например:

```
> &get.geo
geo 1:1,5548.2,03735.6,0.5,C geo 2:1,5548.2,03735.6,2.0,C OK
>
```

4.18.3. &set.[<psw>].geo=<id>,[<st>],[<lat>],[<long>],[<diff>],[{C|R}]

Устанавливает значение объекта *Гео-Зона* с номером <id> в соответствии с параметрами команды:

- <status> - состояние объекта: включено (1) или выключено (0).
- <latitude> - географическая широта центра Гео-Зоны в формате ГГММ.С, где ГГ - градусы, ММ – минуты, С – десятая доля минуты.
- <longitude> - географическая долгота центра Гео-Зоны в формате ГГГММ.С, где ГГГ - градусы, ММ – минуты, С – десятая доля секунд.
- <diff> - расстояние в минутах с десятой долей от центра до границы Гео-Зоны в формате ММ.С, где ММ – минуты, С – десятая доля минуты. Максимальное значение параметра <diff> равно 20.0. Минимальное – 0.1.
- C или R - флаг «формы» Гео-Зоны: «круг» (Circle) или «квадрат» (Rectangle).

Например:

```
> &set.geo=1,1,5548.2,03735.6,2.0
OK
> &set.geo=1,1,,,5.2
OK
>
```

При определении и работе с объектами *Гео-Зона*, следует учитывать, что множественное создание *Гео-Зон* является независимым для каждого определения

географической области. С точки зрения событийной модели *Устройства*, множественные *Гео-Зоны* являются тождественными.

Параметры <latitude>, <longitude> и <diff> должны соответствовать требуемому формату. Например, команда выдаст сообщение об ошибке ERROR если параметр был введён без десятичной точки или без десятой доли.

Пример 1 (нет десятичной точки в параметре <latitude>):

```
> &set.geo=1,1,5548,3735.6,2.4
ERROR
>
```

Пример 2 (значение <diff> вне допустимого диапазона):

```
> &set.geo=1,1,5548.2,3735.6,21.4
ERROR
>
```

4.18.4. &del.[<psw>].geo

Удаляет все ранее созданные объекты *Гео-Зона*. При исполнении команды &del.geo уничтожаются копии объектов *Гео-Зона* сохраненные во Flash командой

```
&set.save.
```

Например:

```
> &del.geo
OK
>
```

4.19. Работа с Аналоговыми входами

Версия *Устройства* с поддержкой команд работы с аналоговыми сигналами имеет дополнительные команды для настройки триггеров «Мин» и «Макс», а также команды возврата настроек и текущих значений уровней сигналов на аналоговых входах *Устройства*.

Аналоговые триггеры дополняют событийную модель *Устройства* тремя событиями (см. рис. 4.19):

- ADC_LESS** уровень сигнала на входе стал ниже значение триггера «Мин»
- ADC_MIDDLE** уровень сигнала на входе превысил значение триггера «Мин», но ниже значения триггера «Макс»
- ADC_OVER** уровень сигнала на входе превысил значение триггера «Макс»

4.19.1. &get.[<psw>].analog[=?|<id>]]

Возвращает текущие состояния аналоговых входов или значения пороговых уровней «Мин» и «Макс» в формате 0.00 вольт. Параметр определяет тип выводимых данных:

- ? вернуть пороговые уровни в порядке следования «Макс», «Мин» и по порядку следования номеров входов (от 1 до 4) следующих через запятую.
- <id> вернуть текущее состояние аналогового входа с номером <id>. Если параметр <id> равен нулю, то команда вернёт значение встроенного датчика температуры в градусах Цельсия.
- если параметр не задан, то команда возвращает текущие значения уровней сигнала на всех аналоговых входах в порядке следования их номеров от 1 до 4 и следующих через запятую.

Пример 1:

```
> &get.analog=?  
30.00,10.00,30.00,30.00  
0.00,3.50,0.00,0.00  
OK  
>
```

В этом примере первая строка возвращаемых параметров соответствует триггеру «Макс» для аналоговых входов в порядке их следования, а вторая строчка – триггеру «Мин». Выполнение команды завершается выводом ОК.

Пример 2:

```
> &get.analog=2  
5.48  
OK  
>
```

В этом примере команда возвращает значение уровня сигнала на втором входе. Выполнение команды завершается выводом ОК.

Пример 3:

```
> &get.analog  
13.08,5.48,0.00,0.00  
OK  
>
```

В этом примере команда возвращает значение уровня сигнала на каждом входе *Устройства* в порядке следования входов (от 1 до 4). Выполнение команды завершается выводом ОК.

Первый аналоговый вход аппаратно соединён с выводом питания *Устройства*.

4.19.2. &set.[<psw>].analog=<id>,[<min>],[<max>]

Устанавливает значения пороговых уровней «Мин» и «Макс» для указанного аналогового входа в формате 0.00 вольт в диапазоне от 0 до 30В с точностью до

0.03В. Неуказанный параметр «Мин» или «Макс» не изменяет значения порогового уровня.

Значения параметров «Мин» и «Макс» установленные командой [&set.analog](#) могут отличаться от значений полученных исполнением команды [&get.analog](#) в последнем разряде на 3 (три) единицы.

Например:

```
> &set.analog=1,3.00
OK
> &set.analog=2,,18.50
OK
> &set.analog=3,3.00,18.50
OK
>
```

Значения по умолчанию для «Мин» равно 0.00, для «Макс» - 50.00.

4.20. Работа с программными Счётчиками

Счётчик предназначен для подсчёта фронтов сигнала на дискретном входе *Устройства*. В версии прошивки с отключенной опцией Decrement команда [&set.cnt](#) с расширенным набором параметров отключена, см. команду настройки режима работы дискретного входа [&set.zone](#).

Значение программного счётчика хранится в пользовательском регистре, информационная ёмкость которого определяется командой [&add.log](#). Значения пользовательских регистров определённых этой командой входят в состав структуры данных ЛОГ.

4.20.1. &set.[<psw>].cnt=<num>

Команда определяет количество пользовательских регистров работающих в режиме «счётчик». Нумерация пользовательских регистров соответствует номерам дискретных входов *Устройства* и начинается с единицы. Исключением является пользовательский регистр для дискретного входа в режиме Decrement (см. [&set.cnt](#) с расширенным набором параметров)

Например, определить счётный режим для первого дискретного входа:

```
&set.cnt=1
&set.cnt=01,0
&add.log=D,1
```

количество счётных входов
выключить генератор События для входа 1
добавить к структуре ЛОГ 1 регистр
ёмкостью 4 байта

4.20.2. &set.[<psw>].cnt=<mask>,<enable>][,<increment>]

Команда определяет режим дискретных входов *Устройства*. Справедливо для версии прошивки с включенной опцией 2000-0000-0000 (см. раздел [Индикатор опций прошивки](#)).

<mask> - 16-ричное число, бит-позиционный код дискретного входа (значения от 00 до 0F). Установленное (1) значение бита определяет принадлежность дискретного входа к изменению режима определяемого значениями следующих параметров.

<enable> - управление состоянием дискретного входа: 1 - «вход включен» или 0 - «вход выключен». При изменении сигнала на входе в состоянии «включен» генерируется событие [OBJ_ZONE](#). В состоянии «выключен» дискретный вход является «счетным». В этом случае при изменении сигнала на входе изменяется значение пользовательского регистра в соответствии со значением параметра <increment>.

<increment> - управление режимом работы счётного входа, значение 1 – Increment «+1» или значение 0 – Decrement «-1». Если все входы установлены в режим Increment, то пользовательские регистры соответствуют номерам дискретных входов. Режим работы первого дискретного входа должен быть Increment. При наличии дискретного входа в режиме Decrement уменьшается значение пользовательского регистра предшествующего номеру дискретного входа с накоплением (см. примеры ниже).

Пример 1 – один регистр ёмкостью 4 байт; дискретный вход 1 в режиме Increment:

```
&add.log=D,1
&set.cnt=01,0,1          вход 1, регистр 1 (Increment)
&set.cnt=1
```

Пример 2 – два регистра ёмкостью 4 байта; дискретные входы 1 и 2 в режиме Increment:

```
&add.log=D,2
&set.cnt=03,0,1          вход 1, регистры 1 и 2 (Increment)
&set.cnt=2
```

Пример 3 – один регистр ёмкостью 2 байта; дискретные входы 1 и 2 в режиме Increment и Decrement, соответственно:

```
&add.log=W,1
&set.cnt=01,0,1          вход 1, регистр 1 (Increment)
&set.cnt=02,0,0          вход 2, регистр 1 (Decrement)
&set.cnt=1
```

Пример 4 – два регистра ёмкостью 2 байта; дискретные входы 1, 2, 3 и 4 в режиме Increment, Decrement, Increment и Decrement, соответственно:

```
&add.log=W,2
&set.cnt=01,0,1          вход 1, регистр 1 (Increment)
&set.cnt=02,0,0          вход 2, регистр 1 (Decrement)
&set.cnt=04,0,1          вход 3, регистр 2 (Increment)
&set.cnt=08,0,0          вход 4, регистр 2 (Decrement)
&set.cnt=2
```

Пример 5 – два регистра ёмкостью 2 байта; дискретные входы 1, 2, 3 и 4 в режиме Increment, Increment, Decrement и Decrement, соответственно:

```
&add.log=W,2
```

<code>&set.cnt=01,0,1</code>	вход 1, регистр 1 (Increment)
<code>&set.cnt=02,0,1</code>	вход 2, регистр 2 (Increment)
<code>&set.cnt=04,0,0</code>	вход 3, регистр 2 (Increment)
<code>&set.cnt=08,0,0</code>	вход 4, регистр 1 (Increment)
<code>&set.cnt=2</code>	

4.20.3. `&get.[<psw>].cnt`

Команда возвращает настройки произведённые `&set.cnt` в формате:

```
counters <num>  
<enabled>;<increment>;<status>
```

где

`<num>` - количество счётчиков созданных командой [&set.cnt=<num>](#);
`<enabled>` - бит-позиционный код дискретных входов в состоянии «включен» (1) или «выключен» (0), 16-ричное число от 00 до 0F;
`<increment>` - бит-позиционный код дискретных входов в состоянии «Increment» (1) или «Decrement» (0), 16-ричное число от 00 до 0F;
`<status>` - бит-позиционный код текущего состояния дискретных входов, 16-ричное число от 00 до 0F.

Например:

```
> &get.cnt  
2  
0C;03;00  
OK  
>
```

4.21. Работа с временными зонами (опция)

Временные зоны позволяют генерировать системное событие `OBJ_TIMEZONE.<tz_id>.TZONE_BEGIN` или `OBJ_TIMEZONE.<tz_id>.TZONE_END` с течением времени суток. Соответствующее системное событие также генерируется, если при включении питания *Устройства* системное GMT время «входит в» или «находится вне» диапазона временной зоны с номером `<tz_id>`.

Временная зона может быть отключена или включена по событию с использованием *макро*. Отключенная временная зона с течением времени не генерирует системных событий.

Состояния временной зоны (а именно, достигло ли реальное время установленного периода определённой временной зоны) на момент исполнения *макро* инструкции может быть возвращено объектом *Условие*. При этом значение состояния временной зоны с номером `<tz_id>` может быть равно 1 (реальное время внутри диапазона времени временной зоны) или 2 (вне диапазона времени).

Настройки параметров временных зон изменённые командой:

```
&set.tzone
```

сохраняются командой [&set.save](#).

4.21.1. &set.[<psw>].tzone=<tzone_id>,<enabled>,<tz_beg>,<tz_end>

Устанавливает (или изменяет) временную зону с номером <tzone_id>.

Параметр <enabled> позволяет отключить (значение параметра равно 0) или включить (значение параметра равно 1) генерирование событий с течением времени.

- Значение флага <enabled> может быть изменено по событию с помощью макро команды (см. [таблицу значений констант](#), объект OBJ_TIMEZONE).

Вводимое значение времени <tz_beg> и <tz_end> должно соответствовать формату
HHMMSS

где HH – часы (00..23), MM – минуты (00..59) и SS – секунды (00..59).

- Если значения <tz_beg> и <tz_end> равны, то по достижении системного времени равным <tz_beg> будет сгенерировано событие TIMEZONE_BEGIN, а через одну секунду – событие TIMEZONE_END.
- Значение <tz_beg> может быть больше значения <tz_end>.

Например, требуется определить временную зону соответствующую периоду рабочего дня с 8 часов утра до 6 часов вечера (080000 и 180000, соответственно):

```
> &set.tzone=1,1,080000,180000
OK
>
```

4.21.2. &get.[<psw>].tzone

Возвращает настройки временных зон в виде:

```
<tzone_id>,<enabled>,<tz_beg>,<tz_end>
```

где <tzone_id> идентификатор временной зоны, <enabled> признак временная зона включена (1) или выключена (0), границы временной зоны <tz_beg> и <tz_end> разделёнными запятой.

Например:

```
> &get.tzone
1,1,080000,180000
2,0,not valid
3,0,not valid
4,0,not valid
OK
>
```

4.22. Работа с гео-одометром

Гео-одометр – это микропрограммный алгоритм *Устройства* производящий подсчет пройденного расстояния (пробега) на основе геоинформационных данных принятых от навигационного приёмника.

В версии прошивки *Устройства* с поддержкой гео-одометра его значение записывается в ЛОГ и является накапливаемым, т.е. при включении питания *Устройства* значение гео-одометра продолжает отсчёт пройденного расстояния.

С помощью командного интерфейса, а также с помощью макро определений значение гео-одометра может быть изменено.

Например, определение нового значения гео-одометра:

```
&set.odo=15800
```

или сброс значения гео-одометра по событию

```
OBJ_SYSTEM.<ok2>.CMD_OK -> OBJ_LOG.ODOMETER_CONTROL.ODO_RESET
```

Значение гео-одометра возвращается в составе PGIO предложения или в составе объекта Сообщение, шаблон которого содержит тэг %GO.

Пример предложения PGIO (значение гео-одометра выделено красным и подчеркнуто):

```
$PGIO,,00000438,010808,110905,00,00,23.19,00.00,00.00,00.00,,FFFF,04,0,18.1  
41*04
```

4.22.1. &set.[<psw>].odometer={<value>|disable|enable}

В зависимости от первого параметра команда &set.odo производит сброс значения гео-одометра, выключение его работы или включение.

Значение первого параметра <value> может быть в пределах от 0 до 1000000 включительно.

Например, установить новое значение гео-одометра равным 15800 узлов:

```
> &set.odo=15800  
OK  
>
```

или «остановить» работу гео-одометра (при этом значение гео-одометра не изменяется):

```
> &set.odo=disable  
OK  
>
```

- При исполнении команды &set.odo (равно как и при сбросе значения гео-одометра по событию с помощью макро определения) в ЛОГ Устройства записывается две новых записи равных текущему пакету данных. Одна запись перед изменением значения гео-одометра, вторая – после изменения значения гео-одометра. При этом значение параметра <macro_id> в структуре ЛОГ равно 254.

4.23. Работа интерфейсом RS485

Устройство может быть оборудовано последовательным интерфейсом RS485. Работа с этим интерфейсом и подключенными к нему датчиками определяется командой [&add.message](#) и командой &set.cmd представленной в этом разделе.

4.23.1. &set.[<psw>].cmd=<baud_index>,<msg_id>,<fractal_mode>

Устанавливает режимы работы интерфейса RS485 и режимы работы с датчиками подключаемыми к последовательным интерфейсам RS485/232.

Значение параметра **<baud_index>** определяет индекс скорости обмена согласно таблице:

Индекс	1	2	3	4	5	6	7
бит/сек	300	600	1200	4800	9600	14400	19200

Значение параметра равное нулю соответствует утверждению «последовательный интерфейс RS485 отключен».

<msg_id> - определяет номер шаблона сообщения, содержимое которого задаёт количество команд и команды запроса данных внешнего датчика. Описание структуры команды запроса и примеры определения шаблона сообщения для датчиков с поддержкой интерфейсов Omnicomm и Modbus RTU представлены в документе [6].

<fractal_mode> - определяет режим работы модуля приёма и преобразования дробной части числа принятого по последовательному интерфейсу RS232. Биты управления режимом преобразования:

- 00 захват дробной части отключен. Любое исходное число принятое с десятичной точкой интерпретируется как целое, дробная часть игнорируется.
- 01 если бит установлен, то захват дробной части исходного числа включен.
- 02 если бит установлен, то включается «адаптивный» режим захвата дробной части исходного числа. Если исходное число содержит знак после десятичной точки, то производится преобразование одного знака после десятичной точки. Если исходное число не имеет десятичной точки, то результат преобразования такого числа соответствует исходному значению. Если «адаптивный» режим выключен (значение бита 02 равно нулю), то результат преобразования всегда производится с точностью до 1/10. Например:

Режим	Исходное число	Результат преобразования (значение пользовательского регистра)
00	101	101
	101.3	101
01	101	1010
	101.3	1013

03	101	101
	101.3	1013

Значение параметра <fractal_mode> равное 02 является недопустимым.

Команда возвращает OK или ERROR в случае обнаруженной ошибки в формате команды или в значении параметра.

Например:

```
> &set.cmd=5,0,1
OK
> &set.cmd=, ,2
ERROR
>
```

4.23.2. &get.[<psw>].cmd

Команда возвращает настройки произведённые командой [&set.cmd](#). Порядок следования возвращённых параметров и их назначение соответствует описанию команды [&set.cmd](#).

Например:

```
> &get.cmd
5,0,1
OK
>
```

4.24. Работа с пользовательскими флагами

Пользовательские флаги – это массив логических переменных, значение которых устанавливается по событию, с помощью *Макро* инструкций.

Использование пользовательского флага позволяет запомнить факт случившегося системного события, тем самым сохранить соответствующее текущее состояние *Устройства* для того, чтобы учитывать это состояние в будущем.

Синтаксис

[OBJ_FLAGS](#).<bitweight>.1

Например, если *Устройство* приняло запрос от IP-сервера о необходимости произвести сеанс голосовой связи, то необходимо установить пользовательский флаг и разорвать IP-соединение. По событию подключения к сети GSM (online) и, если установлен пользовательский флаг, *Устройство* производит голосовой вызов абонента. По окончании разговора (или если вызов прерван) сбросить пользовательский флаг. Далее, *Устройство* по любому периодическому событию и, если пользовательский флаг сброшен, устанавливает связь с IP-сервером.

4.24. Режим программирования Устройства

Программирование (настройка) *Устройства* производится вручную, с помощью &-команд, путём обновления пакетным файлом настроек (см. команду [&get.sys=cfg](#)) или с помощью программы конфигулятора gConfigurator.

Режим программирования временно блокирует исполнение *Макро* инструкций и работу программных таймеров.

- Значение режима программирования (включен или выключен) при исполнении команды [&set.save](#) во FLASH не сохраняется.
- По умолчанию значение режима программирования «выключен».

4.24.1. &set.[<psw>].programmode={1|0}

Команда производит управление режимом программирования *Устройства*. Значения параметра определяет режим:

- 1 - включить режим программирования;
- 0 - выключить режим программирования.

Например:

```
> &set.prog=1
ОК
>
```

- Режим программирования автоматически отключается через 60 секунд после ввода последней &-команды.

5. Примеры программирования

Каждый пример программирования *Устройства* представленный в этом разделе предполагает, что на момент программирования *Устройство* имеет настройки по умолчанию:

5.1. Управление режимом регистрации по внешнему сигналу

5.1.2. Задача

1. Включение регистрации навигационных данных с заданным интервалом времени при замыкании зоны 1 (кнопка нажата).
2. Выключение регистрации данных при размыкании зоны 1.

5.1.3. Макро программа

Макро 1: OBJ_ZONE,ZONE_ID1,ZONE_OPEN -> OBJ_LOG,LOG_STATUS,REC_OFF
Макро 2: OBJ_ZONE,ZONE_ID1,ZONE_CLOSED -> OBJ_LOG,LOG_STATUS,REC_ALL

Триада слева, выделенная синим цветом, соответствует *активному событию* системы. Красным цветом выделена триада соответствующая *реакции* системы.

5.1.4. Пример программирования

&add.macro=2	Создать два объекта типа Макро
&set.macro=1,1,1,1,0,10,2,2	Запретить запись в LOG
&set.macro=2,1,1,1,1,10,2,1	Разрешить запись в LOG
&set.save	Записать конфигурацию

5.2. Запись гео-точки в ЛОГ по внешнему сигналу

5.2.1. Возможные области применения

1. Фиксация прибытия в пункт назначения.
2. Запоминание местоположения отвечающего заданному критерию.

5.2.2. Задача

Записать навигационные и состояние дискретных данных в «историю» по факту нажатия кнопки 2 (Зона 2).

5.2.3. Макро программа

Макро 1: OBJ_ZONE,ZONE_ID2,ZONE_CLOSED -> OBJ_LOG,LOG_WRITE,0

Триада, выделенная синим цветом, соответствует *событию* системы. Красным цветом выделена *Реакция*. Заметим, что *Устройство* игнорирует значение реакции в триаде *Реакция*.

5.2.4. Пример программирования

```
> &add.macro=1                      Создать объект типа Макро
1
ОК
> &set.macro=1,1,1,2,1,10,5,0      Записать текущие данные в LOG
ОК
>
```

5.3. Изменение скорости записи истории по внешнему сигналу

5.3.2. Задача

1. Нормальная скорость записи навигационных данных составляет 60 секунд.
2. Требуется ускорить процесс снятия и записи навигационных данных в ЛОГ с периодом 5 секунд при замыкании зоны 2.
3. При размыкании зоны 2, необходимо восстановить начальное значение по п.1.

5.3.3. Макро программа

```
Макро 1: OBJ_ZONE,ZONE_ID2,ZONE_OPEN -> OBJ_LOG,SET_PERIOD,60
Макро 2: OBJ_ZONE,ZONE_ID2,ZONE_CLOSED -> OBJ_LOG,SET_PERIOD,5
```

Триада, выделенная синим цветом, соответствует *активному событию* системы. Красным цветом выделена триада *реакции* системы. Примечание: период записи не может быть установлен со значением менее 5 секунд.

5.3.4. Пример программирования

```
> &add.macro=2                      Создать два Макро
2
ОК
> &set.ma=1,1,1,1,0,10,1,60
ОК
> &set.ma=2,1,1,1,1,10,1,5
ОК
> &set.save                        Записать конфигурацию
ОК
>
```

5.4. Кратковременное включение реле по внешнему сигналу

5.4.1. Задача

1. При замыкании зоны 1 требуется включить реле 1 на 10 секунд.

2. Размыкание зоны 1 не должно влиять на состояние релейного выхода 1.

5.4.2. Макро программа

```
Макро 1: OBJ_ZONE,ZONE_ID1,ZONE_CLOSED -> OBJ_RELAY,RELAY_ID1,RELAY_ON
Макро 2: OBJ_ZONE,ZONE_ID1,ZONE_CLOSED -> OBJ_TIMER,TIMER_ID1,TIMER_RUN
Макро 3: OBJ_TIMER,TIMER_ID1,TIMER_STOP -> OBJ_RELAY,RELAY_ID1,RELAY_OFF
```

Триада, выделенная синим цветом, соответствует *активному событию* системы.
Красным цветом выделена *реакция* системы.

5.4.3. Пример программирования

```
&add.timer=1
&add.macro=3
&set.timer=1,20,10,1
&set.macro=1,1,1,1,1,2,2,1
&set.macro=2,1,1,1,1,4,1,1
&set.macro=3,1,4,1,0,2,2,0
&set.save
```

Записать конфигурацию

5.5. Периодическое включение/выключение реле

5.5.1. Возможные области применения

Система оповещения по событию.

5.5.2. Задача

1. При замыкании Зоны 1 требуется регулярно включать реле 2 на одну секунду с периодом повторения 2 секунды.
2. Размыкание Зоны 1 должно остановить процесс по п.1.

5.5.3. Исходные объекты

Таймер 1: TIMER_NONSTOP, Период повторения 2 сек.

Таймер 2: TIMER_LIMITED, Период повторения 1 сек, кол-во повторений 0.

5.5.4. Макро программа

```
Макро 1. OBJ_ZONE,ZONE_ID1,ZONE_CLOSED -> OBJ_TIMER,TIMER_ID1,TIMER_RUN
Макро 2. OBJ_TIMER, TIMER_ID1,TIMER_RUN -> OBJ_TIMER,TIMER_ID2,TIMER_RUN
Макро 3. OBJ_TIMER, TIMER_ID1,TIMER_RUN -> OBJ_RELAY,RELAY_ID2,RELAY_ON
Макро 4. OBJ_TIMER,TIMER_ID2,TIMER_END -> OBJ_RELAY,RELAY_ID2,RELAY_OFF
Макро 5. OBJ_ZONE,ZONE_ID1,ZONE_OPEN -> OBJ_TIMER,TIMER_ID1,TIMER_STOP
```

5.5.5. Пример программирования

```
> &del.timer
```

Удалить все объекты типа Таймер

OK	
> &del.macro	Удалить все объекты типа Макро
OK	
> &add.timer=2	Создать два объекта Таймер
2	
OK	
> &add.macro=5	Создать пять объектов Макро
5	
OK	
> &set.timer=1,10,2,0	Настроить первый таймер
OK	
> &set.timer=2,20,1,0	Настроить второй таймер
OK	
> &set.macro=1,1,1,1,1,4,1,1	Настроить макросы...
OK	
> &set.macro=2,1,4,1,1,4,2,1	
OK	
> &set.macro=3,1,4,1,1,2,3,1	
OK	
> &set.macro=4,1,4,2,0,2,3,0	
OK	
> &set.macro=5,1,1,1,0,4,1,0	
OK	
> &set.save	Сохранить произведённые настройки
OK	
>	

Используйте буфер обмена для быстрого ввода команд:

```
&del.ti
&del.ma
&add.ti=2
&add.ma=5
&set.ti=1,10,2,0
&set.ti=2,20,1,0
&set.ma=1,1,1,1,1,4,1,1
&set.ma=2,1,4,1,1,4,2,1
&set.ma=3,1,4,1,1,2,3,1
&set.ma=4,1,4,2,0,2,3,0
&set.ma=5,1,1,1,0,4,1,0
&set.save
```

5.6. LED тест

LED тест позволяет проверить работоспособность исполнительных устройств подключённых к выходам *Устройства*, а также изучить принципы программирования с использованием объектов *Макро* и *Таймер*.

5.6.1. Задача

По сигналу от входа 1 включать каждый выход периодически один за другим на 1 секунду (принцип «бегущая волна»).

5.6.2. Программирование

Для быстрого ввода и настройки *Устройства* повторите приведённую ниже последовательность команд в терминальной программе:

```
&del.ma
&del.ti
&add.ti=1
&add.ma=12
&set.ti=1,20,1,4

&set.ma=1,1,1,1,1,2,2,0
&set.ma=2,1,1,1,1,4,1,1

&set.ma=3,1,4,1,255,2,2,0
&set.ma=4,1,4,1,255,2,3,0
&set.ma=5,1,4,1,255,2,6,0
&set.ma=6,1,4,1,255,2,7,0

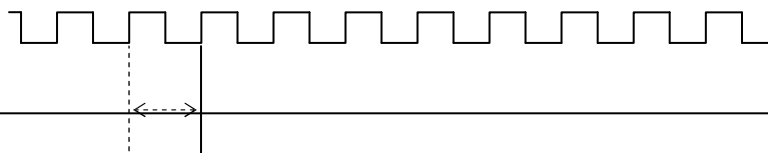
&set.ma=7,1,4,1,1,2,2,1
&set.ma=8,1,4,1,2,2,3,1
&set.ma=9,1,4,1,3,4,6,1
&set.ma=10,1,4,1,4,2,7,1
&set.ma=11,1,4,1,0,4,1,1

&set.ma=12,1,4,3,0,4,4,1

&set.save
```

5.6.3. Описание

При замыкании Зоны 1 выключается Выход 1 (см. Макро 1) и запускается Таймер 1 (Макро 2), который генерирует системные сообщения RUN1, RUN2, RUN3 и RUN4 каждую секунду, которые включают дискретные выходы 1, 2, 3 и 4 соответственно (см. Макро 7 – 10). По завершении отсчёта четырёх интервалов Таймер 1 заканчивает работу и запускается заново (Макро 11). Предварительное выключение дискретных выходов с началом каждого периода Таймера 1 обеспечивается макросами 3 – 6.



1 сек

Вход 1

Вых
од 1Вых
од 2Вых
од 3Вых
од 4

Следует заметить, что в представленном примере повторное замыкание *Зоны 1* может запустить Таймер 1 (см. Макро 2) до того момента как «бегущая волна» завершит свой цикл. При этом алгоритм начнет работу сначала.

5.7. LED тест по сигналу «тревога»

Этот тест наглядно демонстрирует принципы работы *Зоны*, поставленной или снятой с охраны. Прототипом теста является [LED тест](#) представленный выше. Необходимо лишь произвести несколько изменений.

5.7.1. Задача

При замыкании шлейфа Зоны 2 поставить Зону 1 на охрану. При размыкании Зоны 2 – снять Зону 1 с охраны.

По сигналу «тревога» в Зоне 1 запустить [LED тест](#) «бегущая волна», представленный выше.

5.7.2. Программирование

Повторите программирование, как описано в предыдущем тесте [LED тест](#). Измените настройки Макро 1 следующей командой:

```
> &set.ma=2,1,1,1,32,4,1,1
ОК
>
```

Теперь при замыкании Зоны 1 [LED тест](#) не запускается, т.к. Зона 1 не поставлена на охрану.

Убедитесь, что перед выполнением следующих действий Зона 2 разомкнута.

Добавим два макро, настроим их и сохраним произведённые изменения:

```
&add.macro=2  
&set.ma=13,1,1,2,0,1,1,48  
&set.ma=14,1,1,2,1,1,1,16  
&set.save
```

Макро 14 снимет Зону 1 с охраны при размыкании Зоны 2. Макро 15 производит обратное действие – поставит Зону 1 на охрану при замыкании Зоны 2.

Замкните Зону 2 и выполните команду **&get.zone**, чтобы убедиться, что Зона 1 находится в режиме «охрана». Замкните Зону 1 для имитации ситуации «тревога». По сигналу «тревога» в Зоне 1 (см. Макро 2) запустится [LED тест](#).

Подтвердить (или снять) состояние «тревога» в Зоне 1 можно командой

```
&set.zone=1,confirm
```

или, например, по замыканию Зоны 3 с помощью макроса

```
&set.macro=16,1,3,1,1,1,96
```

5.8. LED тест при выполнении «условия»

Условные выражения встречаются практически во всех задачах связанных с охраной объектов любой сложности.

При описании задач мы часто употребляем выражения типа «при условии», «если», «пока» и т.п. Например:

- Включить сирену, если две Зоны оказались в состоянии «тревога».
- Периодически включать/выключать сигнальную лампу пока сигнал «тревога» остается не подтверждённым.
- Открывая дверь включить свет, если в комнате темно.

Употребление подобных выражений в тексте поставленной задачи требует применения условных выражений при настройке *Устройства*. Система позволяет определять произвольные условия в рамках созданных объектов для условного выполнения *Макро* инструкций.

Поставим простую задачу (см. пример 3 параграфом выше).

5.8.1. Задача

При замыкании Зоны 1 включить Выход 1, если Зона 2 замкнута.

5.8.2. Программирование

Согласно требованию имеем условие:

Зона 2 замкнута

или тоже, но в терминах объектно-ориентированной модели:

OBJ_ZONE, ZONE_ID2, ZONE_CLOSE

Последовательность выполнения команд может быть следующей:

```
&del.macro  
&del.condition  
&add.condition=1  
&set.condition=1,1,2,1  
&add.macro=2  
&set.macro=1,1,1,1,0,2,2,0  
&set.macro=2,1,1,1,1,2,2,1,1
```

Напомним, что в настройке Макро 2 девятый параметр определяет идентификационный номер объекта *Условие* (т.е. условие ID1 должно быть определено, что и было сделано выше).

5.9. Послать SMS-сообщение по внешнему сигналу

Работа с SMS сообщениями требует предварительного определения нескольких параметров:

- текст сообщения (см. объект «[Сообщение](#)»)
- номер телефона получателя сообщения (см. объект «[Пользователь](#)»)
- событие, по которому должно быть послано сообщение (см. перечень событий системы в таблице терминов [Объектно-ориентированной модели](#))

5.9.1. Задача

По сигналу «Зона 1 замкнута» послать SMS сообщение «Door open at %TI-%DA», где ключевые слова %TI и %DA должны быть заменены данными, соответствующими времени и дате события замыкания.

5.9.2. Программирование

Предварительно удалим ранее созданные объекты (настройки системы):

```
&del.message  
&del.user  
&del.macro
```

Создаем объект *Сообщение*:

```
&add.me=Door open at %TI-%DA
```

Создаем объект *Пользователь* (тип 1, приоритет 1):

```
&add.user=1,1,+70951112233
```

Создаем *Макро*:

```
&add.macro=1
```

Предварительно напишем программу управления в терминах объектно-ориентированной системы:

```
OBJ_ZONE,1,ZONE_CLOSE,OBJ_MSG,Message1,User1
```

Используем [таблицу констант](#) для перевода написанного предложения из терминов объектно-ориентированной системы в представление команды настройки *Макро*:

```
&set.macro=1,1,1,1,1,7,1,1
```

Теперь при замыкании Зоны 1 пользователь телефона +70951112233 получит SMS сообщение типа:

```
Door open at 152426-090405
```

5.9.3. Пример диалога в режиме «отладка»

-macro.1 (1,1,1)	-- вызов макроса по событию 1.1.1
-msg.1 usr.1 if.0	-- чтение сообщения
+CMGS="89104287707"	-- команда модему отправить SMS
> Door open at 152426-090405	-- передача модему текста сообщения.1
+CMGS: 15	-- модем подтвердил передачу сообщения
OK	

5.10. Передача навигационных данных по GPRS-каналу

Для обеспечения передачи данных по каналу GPRS необходимо использование IP-сервера, подключенного к сети Интернет и имеющего статический IP-адрес.

Почему статический IP-адрес? В этом случае достаточно только один раз настроить Устройство, а именно адрес IP-сервера (см. команду [&add.user](#)), чтобы Устройство успешно инициировало соединение с сервером. Динамический IP-адрес может быть изменен при повторном включении сервера и его подключении к сети Интернет, что потребует повторных настроек Устройства.

5.10.1. Задача

Посылать навигационные данные с периодом 5 секунд в формате NMEA RMC, а также состояние входов/выходов *Устройства* на IP сервер центральной станции мониторинга и управления.

Автоматически восстанавливать соединение при обрыве связи.

Записывать навигационные данные в журнал событий с периодом 1 минута.

5.10.2. Программирование

Для решения поставленной задачи используем следующие возможности *Устройства*:

- Настройка периодического вывода навигационных данных ([&set.echo](#)).
- Настройка периодической записи навигационных данных ([&set.log](#)).
- Перенаправление потока данных «в канал связи GSM/GPRS-модуля».
- Соединение с IP сервером по *событию*.

Существует несколько способов выполнения поставленной задачи с использованием различных алгоритмов программирования. Мы остановимся на самом простом способе с точки зрения количества команд.

В качестве события, инициирующего связь с IP сервером, выберем событие **GPS_RD4ECHO** (пакет навигационных данных готов к выводу в порт *TE*), которое генерируется *Устройством* согласно настройкам произведёнными командой [&set.echo](#). Следует заметить, что это событие генерируется даже если вывод данных отключен ([&set.echo=,off](#)).

В терминах объектно-ориентированной модели программа выглядит следующим образом:

```
Макро 1. OBJ_GPS,VALID,GPS_RD4ECHO -> OBJ_GSM,User1,GSM_LINK      IF(1)
Макро 2. OBJ_GSM,User1,GSM_CONNECT -> OBJ_SYSTEM,FORWARD,SYS_DATA2GSM
```

где

User.1 порядковый номер объекта Пользователь 1 (см. [&add.user](#)).
IF(..) порядковый номер объекта *Условие*. Использование условного выражения позволит исключить попытку соединения с IP сервером, если IP было установлено ранее.
Объект *Условие* выглядит так: **OBJ_GSM,0,IP_CONNECTION – NOT**
Флаг **NOT** позволит проверить факт отсутствия IP-соединения с сервером в момент исполнения *Макро* инструкции.

Используем таблицу констант для перевода написанного предложения из терминов объектно-ориентированной системы в представление команды настройки *Макро*:

```
&set.ma=1,1,3,2,2,12,1,4,1
&set.ma=2,1,12,1,5,5,1,2
```

Первый макрос инициирует соединение с IP сервером при готовности вывода пакета данных в порт *TE*. Второй макрос переключает поток данных в канал GSM/GPRS по факту подключения к IP серверу.

Пример команд программирования и настройки *Устройства*:

```
&del.condition
&add.co=1
&set.co=1,12,0,C9A0,not
&del.user
&add.user=4,,10.10.10.10,555
&del.macro
&add.macro=2
&set.ma=1,1,3,2,2,12,1,4,1
&set.ma=2,1,12,1,5,5,1,2
```

```
&set.log=300,valid  
&set.echo=5,all,5
```

Теперь периодический вывод «всех» пакетов данных (с верными и ложными навигационными данными, согласно второму параметру команды [&set.echo](#)) будет направлен в порт связи GSM/GPRS и IP-сервер получит эти данные после подсоединения. В случае «обрыва» связи с приходом очередного пакета данных *Устройство* восстановит связь с IP сервером, и последующие пакеты данных будут снова передаваться в канал связи.

5.10.3. Отладка

Включить режим «отладка» командой [&set.echo=,35](#). Если режим «отладка» сохранён (команда [&set.save](#)), то после перезапуска *Устройства TE* вернёт :

```
GELIX-2 v4.747  
/8IO/4A/GPS/GSM/2SIM/RTC/WD/CNT/upgr id 00001023  
psw no  
log x30,all,0,0,0 echo x30,off,17 last id 742  
size 38,57 flash 523648 free heap 1404  
gps 4800,poll 1s,06  
gsm pwrON,pinNO,lnLOC,ipNO,06,,1.0 user 1  
msg 2,0 macro 0 timer 3 condition 2  
> AT+CIND?  
+CIND: 5,1,0,0,0,0,0  
OK  
  
-macro.1  
AT+E2IPA?  
*E2IPA: 0  
  
OK AT+E2IPA=1,1  
OK  
  
AT+E2IPO=1,"10.10.10.10",555  
CONNECT  
-macro.2
```

Теперь исполнение команды «?» в строке **gsm pwrON,pinNO,lnOFF,ipYES** сообщит, что IP соединене установлено. Как это происходило рассмотрим по порядку исполнения команд (см. пример лога выше):

AT+CIND? – команда запрашивает текущее состояние сети GSM. Возвращает

```
+CIND: 5,1,0,0,0,0,0  
OK
```

Первый параметр - качество связи с GSM, второй - флаг соединения (0) нет оператора, идёт поиск; (1) – подсоединение к локальному оператору или (2) роаминг, GSM ONLINE; остальные параметры игнорируются.

AT+E2IPA? - запрос на тек. состояние GPRS соединения. Возвращает *E2IPA: 0 если нет соединения и OK.

AT+E2IPA=1,1 - команда установить GPRS соединение согласно профайла № 1 (первая SIM-карта). Возвращает *E2IPA: 000 (в версии с модемом GR64) и OK если соединение установлено. Иначе – ошибка, например *E2IPA: 255 (см. коды ошибок в Приложении 7 этого документа), В случае ошибки следует проверить и

произвести настройки GPRS APN согласно описанию в параграфе 4.17 (команда [&set.gsm](#)) этого документа или повторить попытку соединения позже.

AT*E2IPO=1,"10.10.10.10",555 - установить соединение с IP-сервером по адресу 10.10.10.10 порт 555 используя профайл № 1 (первая SIM-карта). Возвращает CONNECT, если соединение установлено или NO CARRIER, если ошибка или соединение разорвано сервером.

5.11. Регистрация данных расхода топлива

Качественный учет расхода топлива во время эксплуатации транспортного средства или иного механизма позволяет экономить расходные средства за счет получения более точных данных о расходе топлива, а также выявить признак возможных неполадок силового агрегата.

С целью учета расхода топлива в основную магистраль подачи топлива устанавливается первый датчик расхода топлива. Если силовой агрегат имеет дренажную систему, то в неё устанавливается второй датчик расхода топлива.

В нашем примере будем ориентироваться на импульсные датчики расхода топлива (например, VZO-4, ДРТ-5 или аналогичные). В этом случае импульсный выход датчика расхода топлива подсоединяется к дискретному (релейному) входу *Устройства* согласно схеме подключения представленной в «Книга I. Руководство по эксплуатации» раздел «Подключение».

5.11.1. Задача

Производить регистрацию данных расхода дизельного топлива по пути следования транспортного средства.

5.11.2. Программирование

Для решения поставленной задачи используем *Устройство* с прошивкой поддерживающей работы с объектом *Счетчик* (OBJ_COUNTER).

Учитывая, что реальный расход топлива с дренажной системой вычисляется получением разницы прямого и обратного потоков, необходимо использовать один программный счетчик и два события от дискретных входов, к которым подключены датчики расхода.

По событию от первого входа (макро 1), к которому подключен датчик прямой подачи топлива будем производить увеличение счетчика, по событию от второго входа (макро 1), к которому подключен датчик обратного тока топлива – уменьшение значения этого же счетчика. Таким образом значение счетчика будет соответствовать реальному расходу топлива.

Макро 1. OBJ_ZONE.1.ZONE_ON →OBJ_COUNTER.1.INCREMENT

Макро 2. OBJ_ZONE.2.ZONE_ON →OBJ_COUNTER.1.DECREMENT

5.11.3. Отладка

Проверить работоспособность *Устройства* в задаче учёта расхода топлива можно выполнением следующих шагов:

1. Вернуть текущее значения счётчика (команда [&req.last](#) или [&get.cnt](#)).
2. Подсоединить первый датчик к первому дискретному входу *Устройства* и выполнить п.1. Значение первого счетчика должно увеличиться.
3. Отсоединить первый датчик и подсоединить второй датчик к второму дискретному входу *Устройства*. Значение первого счетчика должно уменьшиться.
4. Создать условие, при котором *Устройство* произведет запись данных в ЛОГ (например, подключить ГЛОНАСС/GPS-приёмник и исполнить команду [&set.log=5,all,0,0](#)). Перезапустить *Устройство* сбросом по питанию и вернуть значение счётчика командой [&get.cnt](#). Значение первого счётчика должно быть восстановлено согласно его значению сохранённым в ЛОГ до перезапуска *Устройства* (вернуть последнюю запись из ЛОГ можно по команде [&req.last=1](#)).

```
> &ge.cnt
cur 39,0
OK
> &re.la=1
$PGIO,c,00016199,070407,082719,00,00,0.00,0.00,0.00,0.00,,,39,0*7F
OK
>
```

Значение счётчика восстановлено при включении питания.

5.12. Связь по Bluetooth-каналу

Для работы с *Устройством* по радио каналу Bluetooth необходимо подсоединить адаптер Геликс-ВТ к ТЕ-порту *Устройства*. (см. «Книга I. Руководство по эксплуатации», приложение «Подключение»).

Вариант настроек *Устройства* для работы с Геликс-ВТ версии OSA описан в этой главе ниже.

5.12.1. Задача

Обеспечить прием/передачу данных пары «Устройство-Компьютер» по каналу Bluetooth с использованием адаптера Геликс-BT версии OSA.

5.12.2. Программирование

Для программирования *Устройства* с целью поддержки протокола взаимодействия с адаптером Геликс-BT выберем необходимые команды управления встроенного в него модуля OSA и возвращаемые сообщения:

№	Сообщения*	Команды*	Коментарий
1		BTE=0	Отключить «эхо»
2		BTLN=<строка>	Установить сетевое имя модуля
3		BTEXT	Переключиться в режим «данные»
4		###	Переключиться в режим «команды»
5	CONNECTION DONE		Установлено BT-соединение
6	DISCONNECT		BT-соединение прервано
* <i>Принятая команда и возвращенное сообщение от модуля OSA завершаются символом «конец строки»</i>			

При включении питания *Устройство* возвращает в *TE*-порт команды №1 и 2. Сетевое имя адаптера будет определяться по шаблону «G2.<id>», где <id> идентификационный номер *Устройства* определяемый командой [&set.id](#). При выключении питания *Устройства* необходимо прервать возможно установленный режим приема-передачи данных и переключить Bluetooth-адаптер в командный режим (команда №4). Для этого создадим два шаблона сообщений и настроим *Устройство* так чтобы сообщение №1 возвращалось в *TE*-порт при включении питания, а сообщение №2 – перед программным restart:

```
&del.message
&add.me=%LF%CRBTE=0%LF%CRBTLN=G2.%SI%LF%CR
&add.me=###
&se.sys=,,,,,21
```

Следующее сообщение необходимо возвращать в *TE*-порт для переключения Bluetooth-адаптера при подсоединении к нему по каналу Bluetooth:

```
&add.me=%LF%CRBTEXT%LF%CR
```

Bluetooth-адаптер возвращает ключевые слова №5 и №6. Для работы с ними создаем соответственно ключевые слова в настройках *Устройства*:

```
&del.key
&add.key=DISCONNECT
&add.key=CONNECTION DONE
```

При получении ключевого слова DISCONNECT *Устройство* должно переключить Bluetooth-адаптер в командный режим (###), а при получении ключевого слова CONNECTION DONE – в режим передачи данных (команда VTEXIT). Эту работу будут выполнять соответствующие макросы:

```
&del.macro  
&add.macro=2  
&set.macro=1,1,11,1,1,7,2,0  
&set.macro=2,1,11,2,1,7,3,0
```

После этого необходимо сохранить настройки.

```
&set.save
```

5.12.3. Отладка

Проверку работоспособности *Устройства* необходимо производить в два этапа:

а). Подключить *Устройство* к COM-порту компьютера и запустить терминальную программу. Включить питание *Устройства*. Убедиться, что TE-порт *Устройства* перед словом Init возвращает сообщение 1. Например:

```
BTE=0  
BTIN=G2.1001  
  
Init...  
gps.....  
gsm.....  
  
GELIX-2 v4.752  
/8IO/4A/GPS/GSM/2SIM/RTC/WD/upgr  
id 00009999  
psw NO  
log x5,off,0,0,0 echo x5,off,31  
last id 2410  
size 42,36  
flash 519168  
free heap 1354  
gps 4800,poll 1s,06  
gsm pwrOFF  
user 3  
msg 1,0  
macro 3  
timer 0  
condition 1  
key 0  
counter 2  
>
```

Ввести слово CONNECTION DONE и нажать Enter. TE-порт должен вернуть VTEXIT. Ввести слово DISCONNECT и нажать Enter. TE-порт должен вернуть ###. Исполнить команду [&restart](#). TE-порт должен вернуть ###.

б). Подключить Bluetooth-адаптер Геликс-ВТ и подключить *Устройство* к цепи питания. С помощью Bluetooth-адаптера со стороны компьютера произвести поиск Bluetooth-окружения и подсоединиться к устройству с именем, например, G2.1001. Подключить терминальную программу к виртуальному порту Bluetooth и

исполнить команду [&help](#). Терминальная программа должна вернуть информацию об *Устройстве*.

Со стороны компьютера произвести многократные отключения и подключения к *Устройству* G2.1001 в Bluetooth-окружении используя:

- отключение Bluetooth-адаптера от цепи питания.
- удаление Bluetooth-адаптера на расстояние превышающее допустимое для качественного приема/передачи данных или помещение металлического листа на «невидимой линии» между двумя Bluetooth-адаптерами.

Критерием работоспособности является успешное подсоединение к устройству G2.1001 в Bluetooth-окружении и исполнение &-команд.

5.13. Аварийное соединение с резервным сервером

5.13.1. Задача

Подключение к резервному IP-серверу в случае многократных ошибок подсоединения главному IP-серверу.

5.13.2. Программирование

Предположим, что принцип работы главного IP-сервера с *Устройством* основан на том, что для получения данных в *Устройство* посылается команда запроса данных (например, [&req.new](#)), результат исполнения которой возвращается на IP- сервер. Предположим, что отсутствие команды запроса данных от IP-сервера в течение одного часа является аварийной ситуацией и этот факт требует переключения связи на второй сервер.

```
&add.user=4,,10.10.10.10,21000,r
&add.user=4,,10.10.11.11,21000
&add.ti=1
&set.ti=1,11,3600
&add.ma=2
&set.ma=1,1,3,255,2,12,1,4
&set.ma=2,1,12,255,5,5,1,2
&set.ma=3,1,5,2,2,4,1,1
&set.ma=4,1,4,1,1,12,130,4
&set.save
```

где в последнем макросе код 130 равен 128+2: 128 (80h) означает: поставить заявку на соединение с сервером с высшим приоритетом, т.е. впереди иных заявок в очереди (за это отвечает бит 80h). Номер резервного *Пользователя* указывается в младших битах и равен в этом примере 2.

Заметим, что таймер 1 имеет состояние «запущен» на момент сохранения настроек и, следовательно, при включении *Устройства* этот таймер будет запущен автоматически.

Первый макрос предназначен для подключения к главному серверу (user 1). Второй макрос производит перенаправление потока данных от *Устройства* в IP-канал после подключения к любому серверу. После установления IP-связи сервер может отправлять *Устройству* &-команды на исполнение.

5.14. Учёт времени работы силового агрегата (двигателя)

5.14.1. Задача

Необходимо знать, когда у машины было включено зажигание, то есть время когда двигатель был запущен и остановлен.

5.14.2. Решение

Задача выполнима с помощью любого дискретного входа.

Например, подсоединяем вход 1 к цепи IGNITION и "масса" замка зажигания (подробнее см. раздел «Подключение» в «Книга I. Руководство по эксплуатации»). В этом случае состояние первого входа (значение "1") соответствует включенному зажиганию, а состояние "0" выключенному.

Напомним, что состояния всех входов (и выходов) записываются в *ЛОГ* синхронно с навигационными данными.

Однако, в таком решении при достаточно редкой записи в *ЛОГ* или отсутствии таковой (что определяется настройками произведёнными командой [&set.log](#)) информация о времени активного сигнала IGNITION может оказаться не точной. Для решения этой проблемы следует создать два правила (или макроса), которые записывают внеочередную точку в *ЛОГ* при включении и при выключении зажигания. Для примера с первым входом это делается так:

```
&set.ma=1,1,1,1,1,10,5,0  
&set.ma=2,1,1,1,0,10,5,0
```

Номера макросов в Вашем сценарии могут быть другими. Но и в этом решении есть недостаток, связанный с возможным "дребезгом" сигнала IGNITION при включении и выключении зажигания. Ликвидировать дребезг можно двумя способами. Первый достигается увеличением периода опроса дискретных входов (см. команду [&set.sys=.....<delay>](#) где единица измерения параметра <delay> определяется в мсек). При необходимости фильтрации "частого" сигнала IGNITION исчисляемого многократными поворотами ключа зажигания в течение несколько секунд необходимо использовать два [программных таймера](#). Один таймер для ликвидации дребезга при включении, второй - при выключении зажигания. Делается это так:

```
&set.ti=1,20,5,1  
&set.ti=2,20,5,1  
&set.ma=1,1,1,1,1,4,1,1  
&set.ma=2,1,1,1,0,4,2,1  
&set.ma=3,1,4,1,0,10,5,0
```

```
&set.ma=4,1,4,2,0,10,5,0
```

Если в Вашем сценарии нет таймеров (кроме указанных в примере выше), то два последних макроса можно заменить одним:

```
&set.ma=3,1,4,255,0,10,5,0
```

Одна рекомендация, в транспортном средстве с питанием 24В используйте дополнительный резистор от 1 до 2 кОм, который устанавливается последовательно в цепь дискретного входа. Это необходимо для снижения тока потребления входом *Устройства*, что наиболее важно при выключенном зажигании.

5.15. Звуковое подтверждение авторизованной карты доступа

5.15.1. Задача

Необходимо подать звуковой сигнал владельцу беспроводной карты доступа об успешной идентификации.

5.15.2. Решение

Коричневый провод считывателя беспроводной карты доступа должен быть подключен к дискретному выходу 1 согласно рекомендациям параграфа Подключение в документе «Книга I. Руководство по эксплуатации».

Пример настроек *Устройства* представлен ниже. В этом примере предполагается, что Пользователь 1 создан и его тип равен «0» (тип «беспроводная карта доступа»):

```
&del.timer  
&add.timer=1  
&set.timer=1,20,1,1  
&del.macro  
&add.macro=4  
&set.macro=1,1,8,1,1,4,1,1  
&set.macro=2,1,8,1,1,2,1,1  
&set.macro=3,1,4,1,0,2,1,0
```

Первый макрос запускает таймер 1 при внесении карты в поле считывателя. Также, при внесении карты в поле считывателя, замыкается дискретный выход 1, который включает звуковой сигнал встроенный в считыватель.

Через одну секунду, по окончании работы таймера 1, событие OBJ_TIMER.1.TIMER_END выключает звуковой сигнал.

При внесении незарегистрированной карты доступа (кода такой карты нет в системном списке *Пользователей*) генерируется событие OBJ_USER.0.USER_ACCESS, которое игнорируется макро сценарием представленным в этом примере.

5.16. Работа с датчиком расхода топлива LLS-12

5.16.1. Задача

Необходимо производить учёт уровня топлива в баке с использованием датчика LLS-12.

5.16.2. Проверка режима работы ДУЖ-12

В первую очередь необходимо убедиться в работоспособности датчика ДУЖ-12. Подключаем датчик к COM-порту компьютера. Запускаем программу Hyperterminal (COM-порт должен иметь настройки 19200bps 8N1 управление потоком выключено) и посылаем в порт строку DP (в старых версиях датчика строка должна завершаться символом <Enter>):

```
DP<Enter>
```

Датчик должен вернуть строку и повторять её каждую секунду. Например:

```
F=F164!U=0000 t=7A D32=00000000 N=030A.1  
F=F164!U=0000 t=7A D32=00000000 N=030A.1  
F=F164!U=0000 t=7A D32=00000000 N=030A.1
```

Если датчик не возвращает данные, то необходимо проверить соединение датчика с COM-портом компьютера, а также наличие напряжения питания подаваемое на датчик LLS-12. В случае затруднения обратитесь в техническую поддержку «Геликс» или «Омником».

5.16.3. Решение 1

Исполняем две команды (добавить сообщение и установить номер "startup" сообщения):

```
> &add.me=DP%LF  
1  
OK  
> &set.system=,,,,,01  
OK  
>
```

В этом случае при старте *Устройство* будет возвращать в ТЕ-порт сообщение №1, тем самым настраивая датчик LLS-12 на периодический возврат данных уровня топлива. После этой команды датчик начинает возвращать предложение типа: F=F164!U=0000 t=7A D32=00000000 N=031A.1 (в этом примере значение параметра N есть показания датчика уровня).

Создаем ключевое слово, где "F=" есть неизменная часть предложения возвращаемая датчиком LLS-12 и определяем требуемый тэг для "захвата" значения, %WN6 (W - два байта, N - 16-ричное число, 6 - шестое по порядку слева):

```
> &add.key=F=%WN6  
1  
OK  
>
```

где

%D - double word
%W - word
%B - byte

вторая буква после символа % может быть

H - hex
D - decimal

Число в тэге следующее после H (или D) есть порядковый номер "захватываемого" значения. В примере ниже указана нумерация 16-ричных чисел для предложения возвращаемого датчиком LLS-12 (любой знак кроме десятичной цифры и букв от A до F является разделителем):

```
F=F164!U=0000 t=7A D32=00000000 N=031A.1  
0 1 2 3 4 5 6 7
```

или нумерация десятичных чисел для того же предложения (любой знак кроме десятичной цифры от 0 до 9 является разделителем):

```
F=F164!U=0000 t=7A D32=00000000 N=031A.1  
0 1 2 3 4 5 6
```

Сохраним настройки командой &set.save

Данные полученные *Устройством* в результате анализа тэга ключевого слова записываются в текущий пакет данных и могут быть возвращены устройством в составе предложения \$PGIO (значение между значениями температуры и первым счетчиком).

Например (выделено красным):

```
$PGIO,_,00000003,030307,161654,00,00,10.40,0.00,0.00,0.00,,31A1,0,0,0,0*0c
```

Заметим, что если исходное значение имело десятичную точку, то в предложении \$PGIO эта точка не присутствует. "захватывается" все число, поэтому выбор тэга %D, %W или %B необходимо производить с учетом максимально допустимого значения "захватываемого" числа.

Если датчик не подключен или не возвращает своих показаний в порт, то

Устройство в составе предложения PGIO не будет возвращать параметр соответствующий уровню топлива, при этом значение уровня записываемое в ЛОГ равно FFFFh.

Например (значение уровня отсутствует):

```
$PGIO,c,00000004,030307,163103,00,00,10.40,0.00,0.00,0.00,,,0,0,0,0*42
```

5.16.4. Решение 2

Замечено, что в процессе эксплуатации *Устройство* может перестать принимать данные от датчика уровня топлива. Это может быть связано с бросками напряжения в цепи питания. С целью восстановления потока данных возвращаемых датчиком необходимо повторно послать ему команду DP, тем самым включить режим периодического возврата данных.

С помощью настроек представленных в этом параграфе *Устройство* сможет отслеживать прекращение потока принимаемых данных и посылать в ТЕ-порт команду DP.

Убедимся, что шаблон сообщения с командой DP и требуемое ключевое слово существуют (см. [Решение 1](#) выше):

```
> &ge.me
msg 1,DP%LF
OK
> &ge.key
key 1, F=%WH6
OK
>
```

(!) Тэг %LF необходим только при работе со старой версией датчика LLS-12.

Добавим один таймер [c] и определим тип (непрерывный) и время срабатывания таймера [d], например, в десять раз превышающим период возврата данных датчиком:

```
> &add.ti=1
1
OK
> &set.timer=1,11,10
OK
>
```

Создадим два макроса [e]:

```
> &add.ma=2
2
OK
> &set.ma=1,1,11,1,1,4,1,1
OK
> &set.ma=2,1,4,1,1,7,1,128
OK
>
```

Команда [f] определяет макро 1, который запускает таймер 1 (4.1.1) каждый раз с приёмом ключевого слова 1 (11.1.1). Макро 2 [g] – отправляет в ТЕ-порт ключевое слово 1 (7.1.128) каждый раз когда таймер 1 генерирует событие RUN (4.1.1).

Таким образом, если *Устройство* перестало принимать строки начало которых совпадает с ключевым словом 1, то через 10 секунд (и каждые 10 секунд) ТЕ-порт Устройства будет возвращать команду DP.

5.17. Изменение периода записи ЛОГ в зависимости от скорости

5.17.1. Задача

Необходимо изменять значение периода записи ЛОГ в зависимости от скорости движения ТС. При скорости меньше 20 узлов период записи 60 секунд, а при скорости больше 20 узлов – 120 секунд.

5.17.2. Решение

Определяем порог скорости «переход» которого генерирует событие OBJ_GPS.SPEED_LESS.GPS_SPEED или OBJ_GPS.SPEED_OVER.GPS_SPEED в зависимости от вектора изменения скорости ТС.

```
&set.system=, ,20
```

Создаём два макроса. Первый будет устанавливать период записи 120 секунд при превышении порога скорости, а второй макрос – период равным 60 секунд, если скорость ТС стала ниже заданного порога скорости.

```
&set.macro=1,1,3,2,5,10,1,120  
&set.macro=2,1,3,1,5,10,1,60
```

Сохраняем настройки командой &set.save

5.18. Передача фото изображения на IP-сервер

5.18.1. Задача

С помощью цифровой фотокамеры необходимо произвести захват изображения с последующей передачей по GPRS на IP-сервер в двоичном JPEG-формате и с разрешением 320x240.

5.18.2. Решение

Создадим *Пользователя* (адрес IP-сервера) [a]. Заметим, для более оперативного подключения к IP-серверу во втором и последующих сеансах, а также для экономии трафика определим тип соединения «с удержанием GPRS канала»:

```
> &add.user=4,1,201.22.54.17,1002,reset [a]  
1  
OK  
>
```

Согласимся, что в качестве события инициирующего захват фотоизображения будет событие «Зона 1 замкнута». В связи с возможнымдребезгом контактов при замыкании создадим таймер [b] с одиночным интервалом срабатывания 1 сек [c], который будет запускаться по событию «Зона 1 замкнута» [d], а окончание работы таймера будет инициировать захват фотоизображения камерой [e]:

```
> &add.ti=1 [b]  
1  
OK  
> &set.ti=1,20,1,1 [c]  
OK  
>
```

Представление макро определений в терминах мнемособытий:

```
OBJ_ZONE.1.CLOSED -> OBJ_TIMER.1.RUN [d]  
OBJ_TIMER.1.RUN -> OBJ_CAMERA.0.CAM_CAPTURE_BIN [e]
```

Здесь в реакции макро [e] значение идентификатора указывает на индекс разрешения захватываемого изображения (значение по умолчанию равно 320x240 pch).

Успешный захват изображения и сохранение его образа в буферной памяти *Устройства* завершается событием CAM_SNAPSHOT_READY, по которому следующее макро [f] инициирует соединение с Пользователем 1.

```
OBJ_CAMERA.0.CAM_SNAPSHOT_READY -> OBJ_GSM.<user1>.MAKELINKNOW      [f]
OBJ_GSM.<user1>.CONNECT -> OBJ_CAMERA.0.CAM_GET_PICTURE              [g]
```

По окончании соединения [g] *Устройство* начинает возвращать в открытый IP-канал образ изображения.

Успешная передача образа фото изображения завершается событием CAM_IMAGESENT (ошибка событием CAM_ERROR), по которому, в случае необходимости, можно закрыть IP-канал с помощью макро [h] и [i].

```
OBJ_CAMERA.0.CAM_IMAGESENT -> OBJ_GSM.0.BREAKLINK      [h]
OBJ_CAMERA.0.CAM_ERROR -> OBJ_GSM.0.BREAKLINK          [i]
```

Используем таблицу значений констант для формирования последовательности &-команд определяющих макросы с [d] по [i].

```
&add.ma=6
&set.ma=1,1,1,1,1,4,1,1
&set.ma=2,1,4,1,0,16,0,1
&set.ma=3,1,16,0,0,12,1,6
&set.ma=4,1,12,1,5,16,0,0
&set.ma=5,1,16,0,1,12,0,0
&set.ma=6,1,16,0,2,12,0,0
```

5.18.3. Варианты модернизации

1. Использование режима постоянного IP-соединения с сервером. В этом случае событие CAM_SNAPSHOT_READY не генерируется, а захваченный и сохранённый в буфере памяти образ фотоизображения сразу передаётся в открытый IP-канал. Для этого макросы 5 и 6 необходимо отключить:

```
&set.ma=5,0
&set.ma=6,0
```

В этом случае передача только первого образа изображения будет инициировать соединение с сервером. Передача второго и последующих образов будет передана без задержки на соединение с сервером.

2. Захват и передача фото изображения по сигналу от сервера. В этом случае необходимо реализовать режим постоянного соединения с IP-сервером с восстановлением IP-канала после случайной потери связи с сервером. Восстановление IP-канала можно реализовать заменой макро 1 например так:

```
OBJ_GPS.<any>.RD2ECHO -> OBJ_GSM.<user1>.MAKELINK
```

Сигналом от сервера может служить зарезервированное ключевое слово, например OK22, приняв которое Устройство должно инициировать захват и сохранение образа фото изображения:

```
OBJ_SYSTEM.<ok22>.CMDOK -> OBJ_CAMERA.0.CAM_CAPTURE_BIN
```

В терминах &-команд эти два макро определяются так:

```
&set.ma=1,1,4,255,2,12,1,4  
&set.ma=2,1,5,22,4,16,0,1
```

5.19. Работа с резервным(и) IP-сервером

5.19.1. Задача

В системе мониторинга с несколькими IP-серверами (IP-адрес или имя домена) в случае ошибки соединения с сервером производить соединение со следующим сервером из списка.

5.19.2. Решение

Создадим двух *Пользователей* и определим типы *Пользователей* как 4 (IP-адрес) и 6 (имя домена) согласно условию задачи:

```
> &add.user=4,1,201.22.54.17,1002,reset [a]  
1  
OK  
> &add.user=6,1,www.testgps.ru,1002 [a]  
2  
OK  
>
```

Создадим один (первый) макрос:

```
> &add.ma=1 [b]  
1  
OK  
>
```

И определим реакцию «соединиться со следующим пользователем» по событию «ошибка соединения с любым пользователем»:

```
> &set.ma=1,1,12,255,20,12,0,9 [c]  
OK  
>
```

Заметим, в системном Событии 12.255.20 значение идентификатора указывает на «любой» *Пользователь* (255). Это означает, что управляющая Реакция (12.0.9) будет исполнена при подключении к любому *Пользователю*.

5.20. Голосовой вызов при активном соединении с IP-сервером

5.20.1. Задача

В системе мониторинга в режиме непрерывной передачи данных на IP-сервер необходимо при нажатии кнопки «Вызов» произвести:

- регистрацию местоположения
- передачу на IP-сервер текущего местоположения

- голосовой вызов требуемого абонента

5.20.2. Решение

Создадим двух *Пользователей* типа «Адрес IP-сервера» (с включенным режимом «удержания GPRS-канала») и «Номер телефона»:

```
&add.user=4,8,10.10.0.54,60100  
&add.user=1,1,88121112233
```

В качестве события инициирующего IP-соединение с сервером возьмём периодическое событие от навигационного приёмника с игнорированием признака достоверности принятых данных:

```
OBJ_GPS.<any>.RD4ECHO -> OBJ_GSM.<user1>.MAKELINK
```

Напомним, что период генерирования этого события определяется первым параметром команды [&set.echo](#).

Традиционно, по факту соединения с IP-сервером необходимо установить режим перенаправления потока данных:

```
OBJ_GSM.<user1>.CONNECTED -> OBJ_SYS.SET.FWRD2GSM
```

После того как соединение с сервером установлено взаимодействие *Устройства* с сервером может производиться например в порядке:

- Запрос-ответ
- Периодический возврат данных (см. [&set.echo](#))
- Передача данных по событию в соответствии с шаблоном сообщения
- Комбинация выше перечисленных методов

В этом примере мы не станем уточнять способы реализации выше перечисленных методов, а перейдем к выполнению поставленной задачи. Именно, вызов абонента (в нашем примере это абонент №2) по факту замыкания кнопки «Вызов». Согласимся, что кнопка «Вызов» подсоединена к дискретному входу №4 (см. раздел «Подключение» в документе [1]).

```
OBJ_GSM.<user1>.CONNECTED -> OBJ_LOG.WRITE  
OBJ_GSM.<user1>.CONNECTED -> OBJ_MSG.1.SEND2TE  
OBJ_ZONE.<zone4>.CLOSED -> OBJ_TIMER.1.RUN IF(1)  
OBJ_ZONE.<zone4>.CLOSED -> OBJ_GSM.DISCONNECT IF(1)  
OBJ_GSM.<any_network>.REGISTER -> OBJ_GSM.<user2>.MAKELINKNOW IF(2)  
OBJ_GSM.<any>.CALL_MONITOR -> OBJ_TIMER.1.STOP
```

В представленном сценарии управления *Устройством* используется объект OBJ_TIMER.<id1> для сохранения факта нажатия кнопки «Тревога». Сохранение этого факта необходимо для последующего «позволения» инициировать голосовой вызов к абоненту №2 после того, как соединение с IP-сервером будет прервано. В связи с этим указанный объект должен быть типа NONSTOP.

Заметим, что с целью экономии баланса SIM-карты *Устройство* не разрывает связь с GPRS-сетью.

Использование объекта [OBJ_TIMER](#) в качестве флага состояния может быть заменено на объект [OBJ_FLAG](#). Для этого необходимо убедиться, что Ваша версия прошивки имеет включенной эту опцию.

По умолчанию (т.е. при включении питания *Устройства*) таймер №1 должен иметь состояние STOP с периодом, например, 30 сек. Остановка работы таймера №1 производится после набора номера вызываемого абонента (событие CALL_MONITOR).

В сценарии управления *Устройством* также используется объект «[Условие](#)».

Первое условие при нажатии кнопка «Тревога» разрешает:

- запуск таймера №1
- разрыв соединения с IP-сервером, если уоединение с сервером установлено (&set.co=1,[12,1,A0](#)).

Второе условие проверяет состояние и разрешает:


- вызов абонента №2 (голосовой вызов), если таймер №1 запущен (&set.co=2,4,4,1).


Представленный выше метод управления *Устройством* не является рекомендуемым и его требуется испытать перед эксплуатацией на объекте мониторинга.

6. Проблемы и решения, вопросы и ответы



В этом разделе приняты следующие обозначения:

Симптом обозначен восклицательным знаком в начале параграфа.





Параграф, помеченный символом , содержит признаки возможной проблемы.

Решение проблемы представлено параграфе начинающимся символом .



1. Нет связи, TE не отображает эхо-символ при вводе или отображает не читаемые символы

-  Настройки последовательного порта не соответствуют требуемым.
-  Установите настройки *TE* 115200 бит/с, 8 бит, чётность - нет, 1 стоп бит, контроль потока – нет.



2. TE не выводит NMEA предложений, хотя команда «?» сообщает «echo x1,all,7»

-  Настройки GPS-порта не соответствуют настройкам подключенного навигационного приёмника.
-  Установите настройки порта *GPS* в соответствии с рекомендациями Руководства Пользователя навигационного приёмника (см. команду [&set.nmea](#), параметр <baud_index>).
-  Устройство не принимает ожидаемый набор NMEA-предложений от навигационного приёмника.
-  Необходимо произвести соответствующую настройку работы Устройства с навигационным приёмником командой [&set.nmea](#) (см. параметр <mask>).

3. При включении Устройства (или после исполнения команды &restart), пропали ранее произведённые настройки объектов Макро и Таймер

-  Произведённые настройки не были сохранены (см. [&set.save](#)).
-  Необходимо восстановить ранее произведённые настройки (см. команды [&set](#) и [&add](#)) и сохранить изменения командой [&set.save](#).

4. Количество объектов Макро и Таймер не соответствуют созданным (см. команду «?»). Настройки Макро и/или Таймер не действительны (см. команды [&get.macro](#) и [&get.timer](#))

-  Причина связана с прежним обновлением firmware, версия которого имеет изменённый формат данных нулевого сектора (блок конфигурации) и, как следствие, сбой в настройках Устройства и объектов Макро, Условие и Таймер. В последних версиях прошивки этот симптом невозможен за счёт встроенной проверки.
-  Решить проблему можно выполнением команд - очистить сектор конфигурации системы (команда [&erase.cfg](#)) и перезагрузка (команда

[&restart](#)).

номером *99***1# (команда [&add.user=3,1,*99***1#](#)). Также, очень вероятно, что следующее соединение к GPRS-сети будет успешным, если модем соединяется к другой БС, что происходит при движении транспортного средства. Можно также попробовать произвести рестарт модема управляющей реакцией OBJ_GSM(12).RESTART(2).POWER_CONTROL(2).

8. В каких случаях встроенный GSM-модем может оказаться выключенным после включения питания *Устройства*?

- Модем может временно отключиться по причине не исполнения требуемой АТ-команды, но после 3-5 секунд он должен включиться самостоятельно и повторить инициализацию сквозь настройку АТ-командами аналогично тому, как это происходит после включения питания *Устройства*. При этом бит 1 параметра <gm> (см. [&get.system](#)) не сбрасывается.
- ✎ Если указанный бит установлен и модем остается выключенным после 30 секунд, то требуется исполнить команду [&get.system](#) и сообщить о неисправности и результат исполнения этой команды в службу технической поддержки.
- Модем не включается после включения питания *Устройства* при "первом" включении *Устройства* во время его производства. В этом случае команда [&help](#) возвращает знаки вопроса вместо информации о системных настройках.
- ✎ В этом случае необходимо послать в *TE*-порт файл системных сообщений (`_text_messages.hex`). Файл можно загрузить с сайта.
- Модем не будет включен после включения *Устройства* если бит 1 параметра <gm> (см. [&get.system](#)) равен нулю.
- ✎ В этом случае необходимо исполнить команду [&set.system=1](#).

9. Ошибка обновления прошивки с сервера. По окончании обновления прошивки с WEB-сервера *TE* возвращает сообщение об ошибке (FAILED).

- Устройство принимает искаженные данные или «лишние» символы.
- ✎ Есть три решения этой проблемы:
 1. Файл прошивки на WEB-сервере должен иметь расширение BIN.
 2. Вместо протокола HTTP1.1 необходимо использовать протокол HTTP1.0, который не разбивает передаваемые данные на блоки (невозможна докачка файлов, данные возвращаются единым целым).
 3. Переписываем php-скрипты, выдающие файлы, чтобы в них явно переприсваивались заголовки, что возвращаться будет поток единым целым с размером возвращаемого файла.

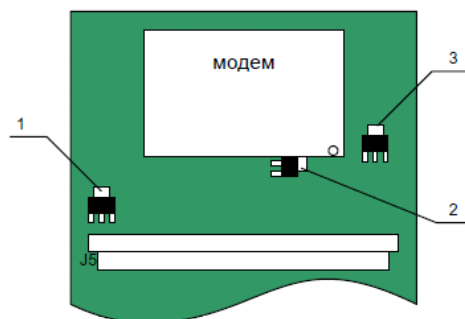
Резюме: модем не распаковывает пакеты данных от сервера, это делается WEB-браузерами.

10. Как получить образ пакета конфигурации, если *Устройство* постоянно «занято» рабочими процессами с GPRS.

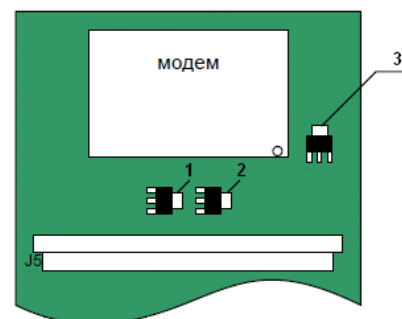
- ✎ Для получения образа пакета настроек (&get.sys=cfg) нет необходимости вынимать СИМ-карту или «ловить» момент исполнения какой-либо &- команды. Достаточно включить питание *Устройства* с предварительно нажатой клавишей Escape (например в терминальной программе MS HyperTerminal) - в этом случае *Устройство* не будет загружать из flash свои настройки, а запустится с настройками по умолчанию, в котором можно исполнить команду &get.sys=cfg. Для возврата в "боевой" режим достаточно произвести restart устройства командой [&res](#) или сбросом по питанию.
- ✎ Второй способ – исполнить команду &set.programm=1, тем самым приостановить исполнение сценария управления и исполнить команду &get.sys=cfg. Действие команды [&set.programm](#) прекратиться автоматически через 60 секунд после ввода последней &-команды или по команде &set.programm=0.

11. Есть ли на плате контрольные точки для первичной диагностики и каковы напряжения в этих точках?

- ✎ Устройство с РСВ (печатными платами) версий до 4.51 включительно на винтах 6 и 10 винтового клеммника J5 имеют выходное напряжение $+5V \pm 10\%$. Ещё три контрольные точки расположены на плате на стороне модема и служат для питания процессора 2.5V (точка 1) и 3.3V (точка 2), а также модема 3.75V (точка 3). Измерение производить относительно вывода GND (винт 2-J5)



PCB v4.35 и младше



PCB v4.5x

12. Приборы установлены на транспортные средства и настроены так, чтобы установить соединение с IP-сервером и периодически передавать данные в NMEA-формате. Как произвести удалённую настройку всех приборов для передачи данных описанных в руководстве «Пример реализации передачи данных на Web-сервер методом HTTP GET»?

- ✎ Здесь используется способ настройки *Устройства* путём отправления ему файла пакетных настроек (см. команду [&get.sys=cfg](#)). Для этого согласно рекомендациям упомянутого документа необходимо настроить *Устройство* "на столе" и произвести испытания установив его в автомобиль. Потом необходимо подключить это *Устройство* к компьютеру и исполнить команду [&get.sys=cfg](#), записав при этом возвращаемые данные в файл. Ещё раз произвести проверку, послав полученный файл настроек этому тестовому *Устройству* по каналу IP от сервера. Если всё работает, тогда можно отправлять по IP-каналу файл настроек каждому *Устройству* Вашей системы.
- Для информации: при загрузке пакетного файла настроек идентификатор *Устройства* и SIM PIN не изменяются.
- ✎ Принцип работы сервера может быть следующим: если сервер после соединения с *Устройством* принимает данные в формате NMEA, то сервер вместо ответа-подтверждения возвращает *Устройству* файл настроек.

13. Прибор постоянно подключен к цепи питания и периодически передает данные на IP-сервер по каналу GPRS. Однако, с течением времени IP-сервер перестаёт принимать данные. С подключением терминала к прибору выяснилось, что команда «выйти в GPRS» *E2IPA=1,1 не производит соединения с сетью GPRS и завершается ошибкой. Перезапуск прибора по команде &reset или по событию (реакция OBJ_GSM(12).RESTART(2).POWER_CONTROL(2)) не помогают. Что делать чтобы восстановить работоспособность модема?

- Не исключено, что в процессе работы *Устройства* GSM-сеть может оказаться загруженной настолько, что GPRS-услуга оказывается временно не доступной. Замечено, что после восстановления GPRS-связи встроенный модем *Устройства* не всегда способен производить успешное подключение к GPRS-сети.
- ✎ Рекомендуем производить перезапуск модема по событию. В качестве события инициирующего перезапуск можно взять событие от таймера, период которого равен минимум двухкратной сумме «период передачи данных на сервер» и «время соединения» (последнее значение определяется таймаутами сети, состоит из времени подключения к сети GPRS и времени IP-подключения к серверу и равно минимум 3 минуты). Таймер должен запускаться с получением ключевого слова от сервера. Например слова-подтверждения. Пример настроек:

```
&set.timer=1,11,1200
&set.macro=1,1,10,1,4,4,1,1
```

```
таймер №1 с периодом 20 минут
OBJ_LOG.CONFIRMED -> OBJ_TIMER.1.RUN
```

&set.macro=2,1,4,1,1,12,0,5

OBJ_TIMER.1.RUN -> OBJ_GSM.RESTART

14. Повторили пример из документации, но сервер не принимает данные.

- В некоторых примерах настроек *Устройства* (см. файл Gelix2-IP-sample.pdf) соединение с IP-сервером и передача данных инициируется событием RD4ECHO, RD2REC, NEWREC_OVER или иным событием объекта [OBJ_LOG](#). Однако, следует помнить, что все эти события основаны на приёме данных принимаемых от GPS-приёмника. Следовательно, если GPS-приёмник не работает или устройство имеет специфические настройки (например, запись в ЛОГ отключена), то указанные события не генерируются.
- ✦ Проверить работоспособность GPS приёмника можно включением перенаправления потока данных принимаемых от GPS-приёмника в порт *TE*. Режим перенаправления потока данных включается командой [&set.echo](#) в третьем параметре (&set.echo=,40). После исполнения команды &set.echo=,40 порт *TE* должен периодически (каждую секунду) построчно возвращать предложения \$GPGGA и \$GPRMC.
- ✦ Если периодического возврата данных не происходит – GPS приёмник выключен или неисправен. В версии прошивок 4.x65 (build 1362) и выше проверить состояние питания GPS-приёмника можно командой «?» (см. параметр off в строке gps).
- ✦ Периодический возврат данных производится, но событие (например, RD2REC или NEWREC_OVER) не генерируется в случае, если включен фильтр «валидности» или фильтр «скорости» (см. команду [&set.log](#)), но GPS-приёмник не зафиксировал валидные координаты. Проверить «валидность» принимаемых координат можно по признаку валидности в предложении \$GPRMC. Если второй параметр предложения \$GPRMC равен «V», то данные «не валидны». Значение параметра «A» означает данные «валидные».

15. Передача данных на Web-сервер производится корректно, но модем перезапускается после завершения сеанса связи.

- С одной стороны, при получения запроса (например, HTTP GET) Web-сервер возвращает HTTP-заголовок *Устройству*. Режим возврата HTTP-заголовка может быть отключен. С другой стороны, *Устройство* после отправления GET-запроса ждёт от сервера HTTP-заголовок, приняв заголовок игнорирует его и интерпретирует следующие за заголовком строки как команды или ключевые слова. По умолчанию режим ожидания *Устройством* HTTP-заголовка и его игнорирование включены. В случае, если Web-сервер не возвращает HTTP-заголовок, а *Устройство* согласно настройкам ожидает его, то при разрыве соединения (NO CARRIER) производится перезапуск модема и восстановление внутренних флагов состояния, так как ожидаемый HTTP- заголовок не был принят в данном сеансе связи.
- ✦ Рекомендуется согласовать настройки Web-сервера и *Устройства*.

Настройки Web-сервера производятся изменением соответствующих режимов (например, в файле http.conf), а настройки *Устройства* командой настройки пользователя ответственного за данное IP-соединение (см. второй параметр, бит 4 команды [&add.user](#)).

16. Как узнать действительную настройку скорости обмена ТЕ-порта *Устройства*?

- Настройка скорости обмена ТЕ-порта *Устройства* сохраняется в первой странице памяти FLASH в байте со смещением 16 от начала страницы (т.е. относительно нулевого адреса).
- ✓ Вернуть настройку скорости обмена порта ТЕ можно без потери действительных настроек *Устройства* последовательностью действий:
 - включить *Устройство* в режиме «по умолчанию» на скорости 115200 бит/сек (с удержанием клавиши Escape при включении питания);
 - включить режим «отладка» (например, [&set.echo=,37](#))
 - исполнить команду &dump
 - в 16-ом байте (нумерация начинается с нуля) младшие четыре бита вернуть действительную настройку скорости обмена ТЕ-порта 16-ричной цифрой (см. таблицу в описании команды [&set.baud](#));
 - изменить настройки СОМ-порта терминальной программы и перезапустить *Устройство* по питанию.

Например:

Индекс скорости обмена ТЕ-порта

```
&dump
0000:00  5134FFFFFFFFFFFFFFFF1C5C000000FF :Q4.....\....
0000:01  000019D1000000000000F00000006FF :.....
0000:02  0064060134A10700300700110102FFFF :.d..4...0.....
0000:03  FFFFFFF00000000000000004D4FFFF0001 :.....
0000:04  BF0271FFFFFFFFFFFFFFFFF0A0002 :..q.....
0000:05  090202820A0001070101810C01060902 :.....
0000:06  028203FF02070202820CFF0604010180 :.....
0000:07  0CFF0609020180040100050001820401 :.....
0000:08  FF0C000582040100040101800C011409 :.....
0000:09  0200820C0007090201800C0007090202 :.....
0000:10  820C0007070302820CFF0C0C0000800C :.....
0000:11  030C078403800CFF0C070502800C6502 :.....e.
0000:12  0C0005800258210006050200018C0100 :....x!.....
0000:13  02FFFFFFFFFFFFFFFFFFFFFFFFFFFF :.....
0000:14  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF :.....
0000:15  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF :..... OK
>
```

17. Согласно рекомендациям отключили отладочный режим (&set.echo=,18) и ТЕ-порт перестал отображать данные возвращаемые на IP-сервер. Возможно ли в этом режиме производить контроль отправляемых *Устройством* данных через ТЕ-порт?

- ТЕ-порт *Устройства* производит «зеркальное» отображение данных отправляемых в IP-канал в нескольких случаях: режим «отладка» включен или включен режим перенаправления потока данных типа «COPY2GSM». Обычно, в примерах настроек *Устройства*, мы приводим макрос типа

OBJ_GSM.<user_id>.CONNECTED => OBJ_SYSTEM.SET.FWRD2GSM, который перенаправляет поток данных от устройства в IP-канал GSM-модема. При этом TE-порт “зеркально” возвращает данные только в случае, если включен режим “отладка”.

18. При включении питания TE-порт *Устройства* не возвращает справку и не отображает введенные с терминальной программы коды клавиш. Почему?

- Настройка скорости обмена TE-порта *Устройства* не соответствует настройкам терминальной программы.
- Версия прошивки Вашего *Устройства* поддерживает работу с датчиком НПП «Сенсор». TE-порт *Устройства* с такой прошивкой имеет два режима: «настройка» и «работа».
- ✓ Изменить настройку COM-порта терминальной программы на 115200 бит/с и включить *Устройство* с нажатой клавишей Escape на клавиатуре компьютера.

19. В ходе установки датчика LLS-AF, возник вопрос. При подаче на аналоговый вход напряжения 13.1 Вольт Геликс возвращает на Виалон значение «256», а при напряжении 20 Вольт (максимум для выхода LLS-AF) - «390». Однако, в описании прибора сказано 30 Вольт максимальное допустимое напряжение на входе.

- В описании (см. документ «Книга – I. Руководство по эксплуатации», раздел «Основные технические характеристики») указано максимальное значение на аналоговом входе 30В. Однако, физически электрические цепи аналогового входа *Устройства* могут выдерживать напряжение до 51.2 В. Этот запас сделан для увеличения надёжности работы. Поэтому максимальному значению 1023 после преобразования соответствует 51.2В. Вашему значению 390 соответствует 19.5В, а значению 256 - 12.8В.
- Заметим, что существует некоторое фиксированное отклонение показаний АЦП от реального напряжения на входе *Устройства* из-за разброса электрических параметров компонент, которое составляет около $\pm 10\%$. Проверить это можно например, одновременным соединением всех аналоговых входов на источник питания *Устройства* (винт 1-J5). В результате нашего теста результат (исполнение команды &get.analog) такой: 13.49,13.44,13.59,13.39. Где первому значению соответствует напряжение на первом аналоговом входе (напряжение источника питания).
- Заметим, команда &get.analog в некоторых прошивках может быть отключена. В этом случае текущее значение сигналов можно узнать по команде [&req.last](#) или из последнего пакета данных принятого сервером.

Значения аналоговых сигналов записываются в ЛОГ "как есть", т.е. без преобразования и каких-либо поправок.

20. Согласно рекомендациям отключили отладочный режим (&set.echo=„18) и TE-порт перестал отображать данные возвращаемые IP-серверу. Возможно ли в этом режиме производить контроль отправляемых устройством данных через TE-порт?

- ✓ TE-порт устройства производит «зеркальное» отображение данных

отправляемых в IP-канал в нескольких случаях: режим «отладка» включен или включен режим перенаправления потока данных типа COPY2GSM. Обычно, в примерах настроек устройства, мы приводим макрос типа OBJ_GSM.<user_id>.CONNECTED => OBJ_SYSTEM.SET.FWRD2GSM, который перенаправляет поток данных от устройства в IP-канал GSM-модема. При этом TE-порт «зеркально» возвращает данные только в случае, если включен режим «отладка».

- ✎ Заменить реакцию в макросе OBJ_GSM.<user_id>.CONNECTED => OBJ_SYSTEM.SET.FWRD2GSM на OBJ_SYSTEM.SET.COPY2GSM. В этом случае TE-порт устройства будет возвращать данные передаваемые в IP-порт вне зависимости от состояния режима “отладка”.

Поясните отладочные сообщения возвращаемые TE при исполнении &-команды принятой Устройством по каналу SMS.

+CMTI: "ME",1	-- сообщение модема «принято СМС»
+CMGR=1	-- чтение СМС из буферной памяти модема
+CMGR: "REC UNREAD", "+79104287707", "11/07/13,13:03:04+16"	
-user found	-- результат авторизации пользователя (см. бит 10 параметра <mode>, &set.gsm)
&ge.id	-- текст принятого сообщения (&-команда)
OK	-- ответ модема на команду +CMGR
+CMGD=1,4	-- удаление SMS из памяти модема
OK	
+CMGS="89104287707"	-- вернуть SMS-ответ Пользователю (см. бит 04 параметра <mode>, &set.gsm)
> id 00001001	-- исполнение принятой команды &get.id
OK	-- ответ модема на команду +CMGS
+CMGS: 20	-- модем подтвердил доставку
OK	

7. Техническая поддержка

С целью оперативного решения проблем и перед обращением в службу поддержки необходимо обратиться к материалу описанному в [Разделе 6](#). Если вопрос остался без ответа, то необходимо послать email по адресу support@gwe.ru, в котором сообщается:

- текущая версия прошивки Устройства и настроечные параметры (к письму следует прикрепить файл с результатами исполнения команд [&help](#), [&get.system](#) и [&get.system=cfg](#)).
- описание операционной обстановки и последовательности действий для воспроизведения ситуации.
- лог-файл, полученный с терминальной или иной программы при включенном режиме «отладка» Устройства (см. команду [&set.echo](#)) за время достаточное для обнаружения неисправности.

Целью передачи указанной информации является воспроизведение проблемы в лаборатории службы поддержки, устранение проблемы и подготовка рекомендаций для её решения.

Современные технологии передачи данных позволяют производить техническую поддержку и обучение персонала удалённо, исключая непосредственную передачу Устройства в службу поддержки. Принцип работы с Устройством в Вашем офисе из офиса технической поддержки представлен на рисунке 7.

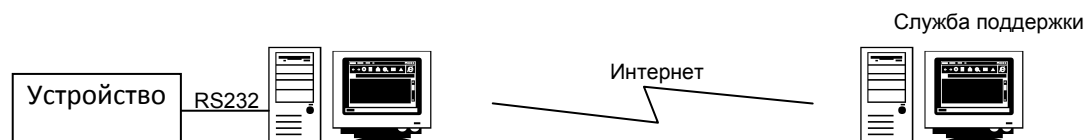


Рис.7. Схема удаленной технической поддержки

Для обеспечения удалённой работы службе технической поддержки с Устройством в Вашем офисе необходимо:

- подключить Ваш компьютер к сети Интернет*
- подключить Устройство к последовательному порту компьютера
- запустить программу IP-COM-Server**
- связаться с службой поддержки – Вам сообщат номер IP-порта связи.
- Сообщить в техническую службу поддержки IP-адрес Вашего компьютера.
- IP-адрес должен быть постоянным на всё время оказания технической поддержки. При наличии в Вашей сети Firewall необходимо настроить для перенаправления потока данных на Ваш компьютер.

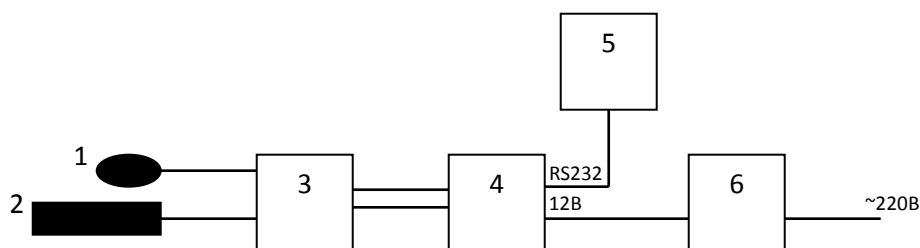
** программу IP-COM-Server можно скачать из директория download сайта разработчика или получить по e-mail от нашей службы поддержки.

Приложение 1. Модуль разработчика

Модуль разработчика «Геликс-КИТ» разработан для быстрого подключения *Устройства* к компьютерному устройству по ИК каналу и предназначен для проведения демонстраций, обучения или настройки *Устройства*.

Состав рабочего стенда для проведения работ с помощью модуля Геликс-КИТ:

1. GPS-антенна
2. GSM-антенна
3. Устройство
4. Модуль Геликс-КИТ
5. Адаптер питания 220/12В
6. Персональный компьютер

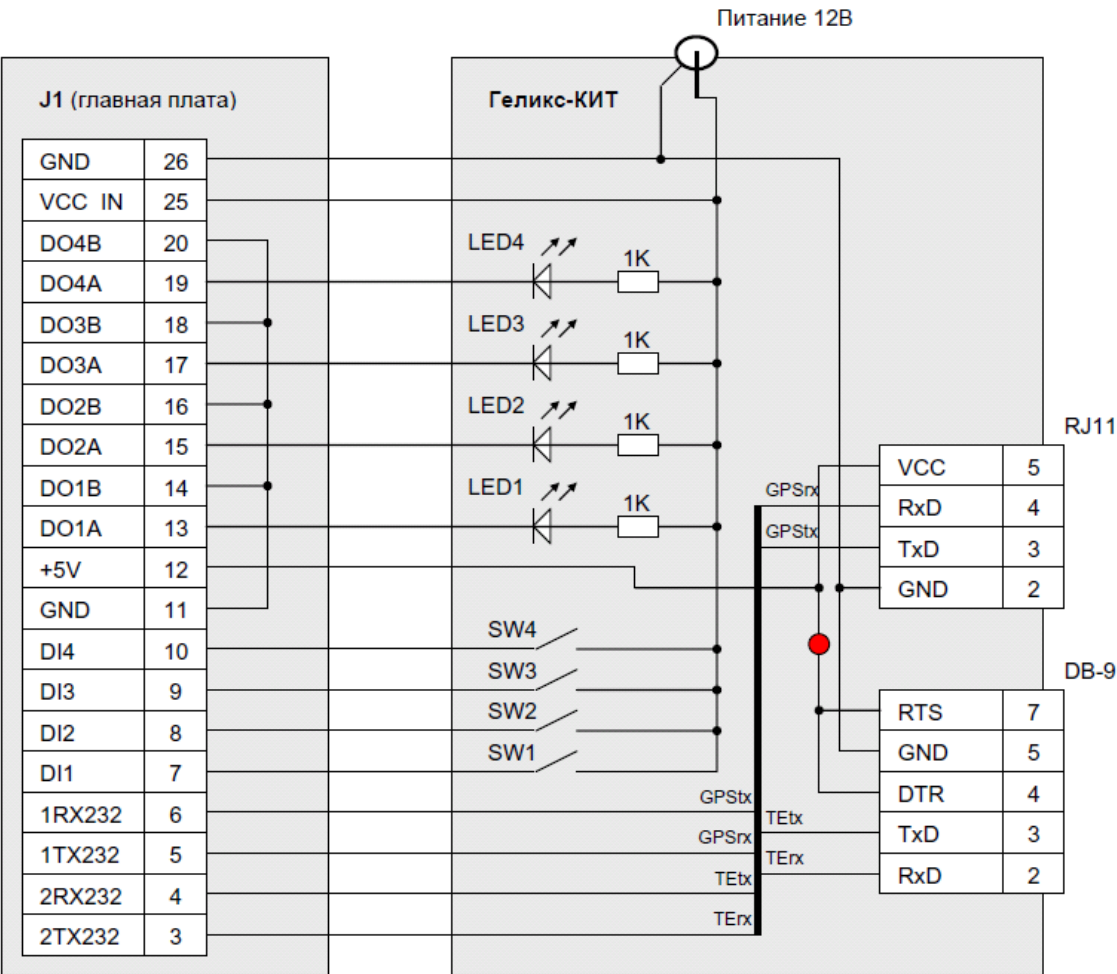


Для получения данных и управления *устройством* компьютерное устройство должно быть оборудовано портом RS232. В качестве компьютерного устройства может быть:

- Desktop PC
- Notebook
- PDA
- Мобильный телефон
- Микропрограммное устройство разработчика

Компьютерное устройство должно быть оснащено соответствующим программным обеспечением. Например, для настройки *Устройства* с помощью компьютера под управлением операционной системы Windows можно использовать терминальную программу MS Hyper Terminal, Telnet или Norton Telemax.

1.1. Принципиальная схема



* Для полного «гашения» индикаторов состояния выходов параллельно каждому LED может быть установлен резистор номиналом 1 кОм (на схеме не показан).

Приложение 2. Карта FLASH

Рисунок справа наглядно представляет страничную архитектуру встроенной FLASH.

1. Блок конфигурации записан в странице 0 адресного пространства FLASH, занимает две страницы (512 байт) и хранит параметры настройки *Устройства*, необходимые для работы прибора после включения питания. Значения параметров блока конфигурации, принятые по умолчанию, приведены в Приложении 3. Формат блока конфигурации приведен в Приложении 6.
2. Если *Устройство* поддерживает работу с *гео-зонами*, то информация о *гео-зонах* записывается в странице 2.
3. Записи *журнала событий* начинаются в странице 100.

Cfg (0-1)
Geo (2)
User (3-5)
Msg (6-9)
Key (10)

LOG
(100..)

GSM версия *Устройства* отличается наличием дополнительных данных для настроек объектов *Пользователь* и *Сообщение* (см. схему справа):

FAT

Приложение 3. Настройки по умолчанию

3.1. Блок конфигурации

№	Описание	Команда настройки	Значение по
1	Пароль администратора	&set.psw	нет
2	Номер версии firmware	нет	4.765 ¹⁾
3	Идентификационный номер <i>Устройства</i>	&set.id	1001
4	Период вывода данных <i>TE</i>	&set.echo	5
5	Перечень доступных NMEA форматов для записи в <i>LOG</i>	Нет	GPGGA+ GPRMC
6	Режимы и форматы NMEA для вывода <i>TE</i>	&set.echo	5 секунд all PGIO+ GPGGA+ GPRMC
7	Режимы записи <i>LOG</i> (период записи, режим записи, фильтр SOG, фильтр количества видимых спутников)	&set.log	60 секунд valid 0 miles/h 4 sat
8	Настройки GPS-приёмника и <i>Устройства</i>	&set.nmea	4800, 1 сек.
9	Настройки объекта <i>Zone</i> (все <i>Зоны охраны</i>)	&set.zone	Enabled NormalOpen Disarmed
10	Настройки объекта <i>Relay</i> (все выходы)	&set.relay	Enabled Open
11	Настройки интерпретации входящих SMS ²⁾	&set.gsm	02
12	PIN-код и GPRS-точка доступа (APN)	&set.gsm	не определен
13	Количество объектов <i>Macro</i>	&add.macro	0
14	Количество объектов <i>Timer</i>	&add.timer	0
15	Количество объектов <i>Условие</i>	&add.condition	0
16	Количество объектов <i>Пользователь</i> ²⁾	&add.user	0
17	Количество объектов <i>Сообщение</i> ²⁾	&add.message	0
18	Количество объектов <i>Гео-Зона</i>	&add.geo	0
19	Период проверки <i>Гео-зон</i> ³⁾		x3 NMEA
20	Пороговый уровень «Мин» ³⁾ на аналог.входе	&set.analog	0.00
21	Пороговый уровень «Макс» ³⁾ на аналог.входе	&set.analog	30.00
22	Init-сообщение	&set.system	Отключ.
23	End-сообщение	&set.system	Отключ.
24	USSD-запрос баланса ²⁾	&set.gsm	Отключ.
25	Режим питания GSM-модуля	&set.system	Вкл.
26	Контроль зависания IP-канала ²⁾	&set.system	Вкл.
27	Номер конфигурации пользователя ³⁾	&set.id	0
28	Количество записей в пакете данных передаваемых по тэгу %RC и %NC	&set.system	5
29	Тип датчика температуры Геликс-TS ³⁾	&set.system	1

30	Время ожидания приёма 128 байтной порции файла прошивки по CSD- или IP-каналу ³⁾	&set.log	120 сек
31	Период опроса состояний дискретных входов	&set.system	2 мсек
32	Режим гео-адометра ³⁾	&set.log	Вкл.
33	Порог скорости «остановка»	&set.log	5 узлов/ч

- 1) значение может быть изменено производителем
 2) только в GSM версии *Устройства*
 3) опция

Приложение 4. Обновление ПО и настроек

4.1. Обновление ПО (firmware)

Новая версия firmware загружается в *Устройство* с помощью терминальной или иной программы с компьютера в последовательный порт, к которому подключено *Устройство*. Обновлённая версия firmware поставляется в виде текстового файла (.hex) или двоичного (.bin).

ВНИМАНИЕ! Изменять содержимое файла firmware не допускается, иначе работоспособность *Устройства* не гарантируется!

Для обновления firmware необходимо выполнить следующие шаги:

1. Скопируйте файл firmware на диск компьютера.
2. Подключите *Устройство* к последовательному порту компьютера.
3. Запустите терминальную программу, которая поддерживает режим передачи файла в ASCII режиме (например, Norton Telex или MS HyperTerminal). Убедитесь, что *Устройство* «на линии» и отвечает, например, на ввод команды «?». *TE* в этом случае вернёт системную информацию.
4. В окне терминальной программы выберите меню, соответствующее команде «Отправить ASCII файл» или «Отправить текстовый файл». В новом окне выберите файл по п.1 и нажмите клавишу Enter.
5. Сообщение *TE* «**Upload run**» свидетельствует об успешном начале загрузки firmware, процесс ввода которого отображается точками.

Процесс загрузки firmware отображается на световом индикаторе состояния кодом «31» (см. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ, раздел «Световой индикатор состояния»).

В завершении ввода firmware *TE* выведет количество принятых байтов и результат загрузки.

Например:

```
Upload run
.....
25341 bytes, OK
```

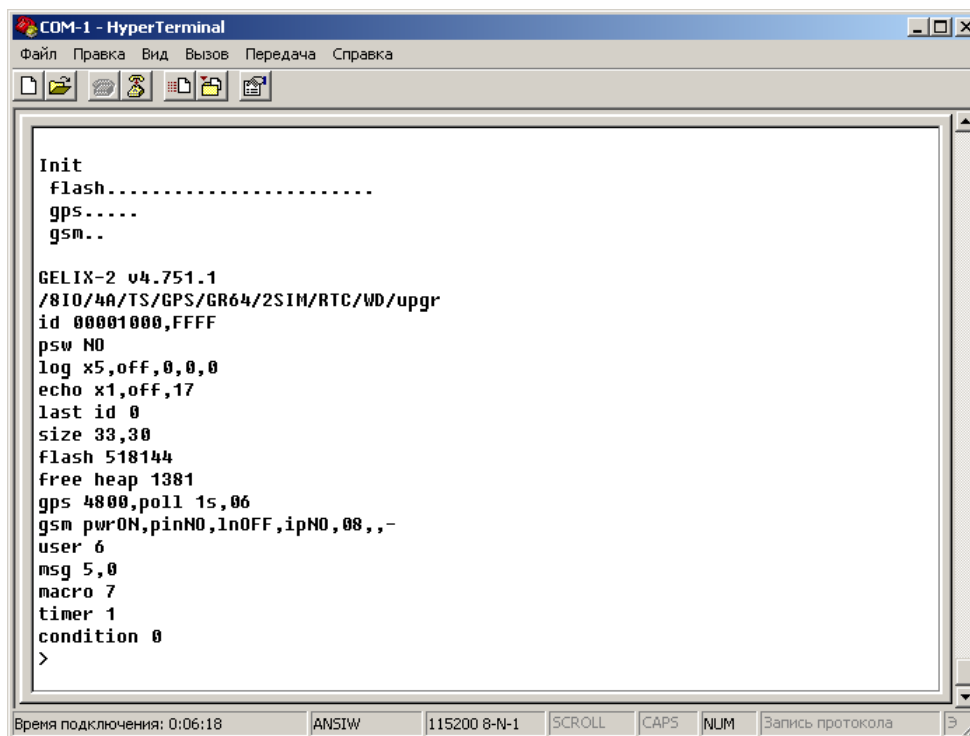
в случае успеха или

```
Upload run
.....
25341 bytes, FAIL
```

если в процессе загрузки firmware была обнаружена ошибка.

Если обновление firmware завершено успешно, то *Устройство* автоматически будет запущено под управлением уже новой версии firmware. При этом

Устройство произведёт очистку (форматирование) разделов FLASH отведенных для ЛОГ:



```
COM-1 - HyperTerminal
Файл  Правка  Вид  Вызов  Передача  Справка

Init
flash.....
gps.....
gsm..

GELIX-2 v4.751.1
/8IO/4A/TS/GPS/GR64/2SIM/RTC/WD/upgr
id 00001000,FFFF
psw NO
log x5,off,0,0,0
echo x1,off,17
last id 0
size 33,30
flash 518144
free heap 1381
gps 4800,poll 1s,06
gsm pwrON,pinNO,lnOFF,ipNO,08,,,-
user 6
msg 5,0
macro 7
timer 1
condition 0
>
```

При обновлении прошивки идентификатор *Устройства* и SIM PIN не сохраняются.

Обновление прошивки (firmware) по каналу GPRS или CSD производится аналогично прямому соединению по COM-порту. В этом случае серверу достаточно послать в предварительно открытый канал данных содержимое HEX-файла прошивки.

4.2. Пакетное обновление конфигурации

Пользовательские настройки *Устройства* могут быть сохранены на компьютере для последующего их «клонирования» на иные *Устройства* в процессе производства или удаленного администрирования объектами миниторинга.

Процесс пакетного обновления конфигурации состоит из двух этапов:

1. Вернуть и сохранить пакет настроек *Устройства* в файл на компьютере (см. команду [&get.system=cfg](#)). Обычно это производится в лаборатории по окончании тестирования системы «Устройство-Сервер».
2. Послать файловый пакет настроек в *Устройство* в открытый канал данных (COM-порт, GPRS IP- или GSM CSD-канал данных). Приняв пакет настроек

Устройство обновит их и произведет программный перезапуск (&restart) уже с новыми настройками.

Приложение 5. Вычисление CRC

Вычисление контрольной суммы (CRC) производится для всех символов исходной строки между символами \$ и *, не включая последних. Значение CRC (два 16-ричных символа) добавляется в конец исходной строки после символа '*' (звездочка). Красным цветом (также подчеркнуто) выделен фрагмент кода для версий прошивки до 2008 года включительно.

```
u8_t int2hex(u8_t);  
// параметр: str - указатель на строку (заканчивается символом '\0')  
// возвращает: указатель на строку str с добавленными ASCII-байтами CRC  
u8_t* make_chsum( u8_t* src )  
{  
    u8_t chksm=0;  
    u8_t *s=src;  
    while( s && *s && *s!='*' && *s!='\r' )  
    {  
        if( *s!='$' && *s!='x' ) chksm = chksm ^ *s++;  
    }  
    if( *s!='*' ) *s++ = '*';          // добавить символ * если его нет  
    *s++ = int2hex( chksm>>4 );  
    *s++ = int2hex( chksm );  
    *s = '\0';  
    return src;  
}  
  
// параметр: байт (используется 4 младших бита)  
// возвращает: байт-символ 16-ричного представления  
u8_t int2hex( u8_t val )  
{  
    u8_t c = val&0x0F;  
    if( c<10 ) return '0'+c;  
    return 'A' + (c-10);  
}
```


Приложение 6. Форматы данных

6.1. Предложение формата PGIO

Предложение формата PGIO определено разработчиком *Устройства* и отображает флаг запуска системы, уникальный номер записи в *журнале*, дату, время, и состояние входов/выходов:

1 2
3
4
5 6 7
8 9 10
11
\$PGIO,C,nnnnnnnn,ddmmyy,hhmmss,io,a,A1,A2,A3,A4,TS,LLS,C1,C2,C3,C4*AA

1. флаг запуска системы
2. порядковый номер записи
3. дата
4. время
5. состояние I/O
6. состояние тревога
7. состояние сигналов на аналоговых входах

В предложении формата PGIO первому параметру может предшествовать идентификационный номер *Устройства* (см. команду [&set.id](#) и [&set.echo](#)).

Например:

\$PGIO,id,C,nnnnnnnn,ddmmyy,hhmmss,io,a,A1,A2,A3,A4,*AA

Флаг запуска системы [1] отображается одним символом и показывает причину запуска системы: включение в цепь питания или перезапуск системы по причине Watchdog.

Символ в поле [1] устанавливается при старте и сохраняется до тех пор, пока не будет произведена запись в *журнал* событий. После того как была произведена запись в *журнал* событий, флаг запуска системы [1] сбрасывается (см. таблицу ниже). Для различия предложений, возвращаемых *TE*, флаг запуска системы имеет два типа – данные, возвращаемые по запросу Оператора (см. команды [&req](#)) и автоматически возвращаемые данные согласно настройкам *Устройства* (см. команду [&set.echo](#)):

Описание флага	Режим вывода данных	
	По запросу <i>TE</i>	Автоматически
Включение в цепь питания	c	C
Перезапуск Watchdog	w	W
«Сброшенный» флаг	–	“пробел”

Порядковый номер записи [2] увеличивается на единицу по факту записи данных в ЛОГ. Порядковый номер записи может оставаться без изменений, если:

- Отключена запись данных в ЛОГ (команда [&set.log=,off](#))

- Запись данных в ЛОГ включена командой **&set.log=,valid**, но принимаемые навигационные данные не достоверны (invalid GPS data).

Дата [3] представлена в порядке следования день (DD), месяц (MM) и год (YY).

Время [4] представлено в порядке следования часы (HH), минуты (MM) и секунды (SS).

Состояние I/O [5] отображается двумя 16-ричными числами с позиционным битовым кодированием *Зоны*. Первые четыре бита (первое число) соответствуют выходам *Устройства*, вторые четыре бита (второе число) – входам *Устройства*. Нумерация входов/выходов начинается справа в порядке следования бит.

Значение	1	0
Выход	Замкнут	Разомкнут
Вход (Зона)	Замкнут	Разомкнут

2f 0010 1111	Выход 2 замкнут, все Зоны замкнуты
69 0110 1001	Выходы 1 и 2 замкнуты, Зоны 1 и 4 замкнуты

Состояние [6] показывает, какие *Зоны* находятся в состоянии «тревога». Отображение состояния производится одной 16-ричной буквой с позиционным битовым кодированием *Зон*. Каждый бит отображает состояние соответствующей *Зоны*:

1 - состояние «тревога»
0 – нет «тревоги».

Нумерация *Зон* начинается справа в порядке следования бит. Например:

1 – тревога в *Зоне* 1
5 – тревога в *Зонах* 1 и 3

Пример сообщения:

```
$PGIO,  
,00002894,120608,092937,00,00,24.53,00.00,00.00,00.00,,FFFF,04,0,0.0*0B
```

Состояние сигналов на аналоговых входах в вольтах (от 0.00 до 52.50). В версии *Устройства* без поддержки аналоговых входов значения поля [7] отсутствуют и запятые следуют одна за другой.

Значение датчика температуры (опция, отсутствует если датчик не подключен).

Значения датчика уровня жидкости или значения переданного внешним контроллером (опция).

Четыре значения программных счётчиков разделённых запятой (опция).

Состояние связи с базовой станцией GSM (опция).

Номер макроса и признак источника записи (опция).

Значение средней скорости с момента предыдущей записи (опция).

Значение пройденного расстояния (опция).

Предложение оканчивается символом «звездочка» и следующим за ним двумя 16-ричными цифрами контрольной суммы (см. «Приложение 5» в этом документе).

6.2. Предложения формата NMEA

NMEA-0183 описывает стандарт передачи данных GPS. NMEA предложения формата GGA, GLL, GSA, GSV, RMC и VTG несут разнородную информацию о времени, дате, координатах, курсе и других параметрах местонахождения системы и видимых спутниках.

NMEA – это формат передачи сообщений между корабельными приборами. Он включает в себя систему сообщений для обмена информацией между навигационными GPS приёмниками и потребителями навигационной информации. Все команды и сообщения передаются в текстовом ASCII виде и начинаются с \$GP. В конце строки сообщения должны быть символы <CR><LF>. В последнем поле сообщения может быть указана контрольная сумма текущего сообщения, начинающаяся с разделителя *. Контрольная сумма – это 8-ми битная (исключающая ИЛИ) всех символов сообщения, включая пробелы, расположенных между разделителями \$ и *, не включая последних. Шестнадцатеричный результат переводится в два ASCII символа (0-9, A-F).

Приложение 7. Зарезервированные слова

Зарезервированные слова принимаются/передаются *Устройством* по командному интерфейсу в процессе работы с внешними устройствами по интерфейсам *TE* и GSM/GPRS.

Слова описанные ниже (за исключением ОК) не рекомендуется отправлять в *Устройство* по командному интерфейсу, иначе связь с *Устройством* или его работоспособность может быть временно нарушена.

Ключевое слово	Описание
ERROR	
+CME ERROR	
+CMS ERROR	
CONNECT	
NO CARRIER	
+CPIN: SIM PIN	
+CPIN: READY	
+CMGR:	
+CMGS:	
Content:	Ответ Web-сервера
+CSCA:	
+CIND:	
Connection:	Ответ Web-сервера
*E2IPO:	
*E2IPA:	
*E2OTR:	
*E2IPRH:	
*ECAV:	
+CMTI:	
+CUSD:	
OK<num>	Хост посылает <i>Устройству</i> . При этом генерируется системное событие OBJ_SYSTEM.<num>.CMD_OK (если <num> не указан, значение <num> равно нулю)

Приложение 8. Сообщения

8.1. OK, End, ERROR и сообщения об ошибках

Сообщения OK, ERROR и End, а также сообщения об ошибках возвращаются *Устройством* по каналам TE, GSM SMS, CSD и GPRS. Таблица (см. ниже) служит для определения причин, вызывающих ошибку, и способов их устранения.

Текст сообщения	Комментарий	Устранение
OK	&-команда исполнена	
End	Возврат данных завершен по причине отсутствия данных в ЛОГе или данные не найдены в ЛОГе согласно заданного критерия поиска.	
ERROR	Ошибка исполнения &-команды. Ошибка параметра или неверная команда. Обращение к несуществующему объекту.	Проверить синтаксис команды. Проверить наличие объекта с соответствующим номером (например при исполнении команды &set.macro=<id>, макро с номером <id> должно быть предварительно создано).
ERROR	Неверная &-команда.	См. раздел Команды в этом документе.
ERR101	Ошибка форматирования FLASH	Обратиться в службу поддержки разработчика.
ERR102	Перезапуск Устройства по причине Watchdog.	Обратиться в службу поддержки разработчика.
ERR103	Не достаточно оперативной памяти.	Обратиться в службу поддержки разработчика.
ERR105	Пустая команда.	См. раздел Команды в этом документе. Обратиться в службу поддержки разработчика.
ERR106	Ошибка SIM PIN или отсутствует SIM карта.	Проверить SIM PIN с помощью мобильного телефона. Установить SIM карту в держатель и защелкнуть держатель.
ERR107	Ошибка в имени доступа к Интернет (APN)	Обратиться к провайдеру GPRS услуг и узнать настройки входа в сеть GPRS. Исполнить AT-команду AT+CGDCONT (см. параграф Настройка входа в GPRS сеть).
ERR108	Ошибка в имени пользователя (login) или пароле доступа в Интернет (APN)	Обратиться к провайдеру GPRS услуги и узнать настройки доступа в GPRS сеть. Исполнить AT-команду AT*ENAD (см. параграф Настройка входа в GPRS сеть).

ERR110	Неопознанная ошибка	Обратиться в службу поддержки разработчика.
ERR111	Датчик температуры не подсоединен или подсоединен неверно. TE выводит это сообщение при обращении к датчику температуры.	Проверить подсоединение или заменить датчик(и) температуры.
ERR112	Ошибка обращения к датчику температуры. Устройство выводит это сообщение при несовпадении контрольной суммы полученных данных.	Проверить подсоединение или заменить датчик(и) температуры.
Текст сообщения	Комментарий	Устранение
ERR113	Попытка отправить AT-команду GSM-модулю на исполнение при выключенном режиме «Отладка».	Включить режим «Отладка» и повторить исполнение AT-команды.
ERR114	Входящий вызов не опознан. Не включена услуга АОН.	Обратиться в службу поддержки клиентов GSM оператора. Включить услугу автоматического определения номера (АОН).
ERR9xx	Ошибка размера структуры пакета данных	Обратиться в службу поддержки разработчика.
FAILED	Ошибка загрузки кода firmware	Изменить параметры соединения. Повторить загрузку кода firmware.
BUSY	GSM/GPRS-модуль занят.	Повторить AT-команду через несколько секунд. Исполнить команду &get.s=? и убедиться, что модуль в состоянии «готов» (0004 или 0006). При повторении ошибки обратиться в службу поддержки разработчика.
TIMEOUT	Превышено время ожидания данных при обновлении firmware или подтверждения от оператора после попытки изменения настроек порта связи.	Перед обновлением кода firmware закрыть все рабочие приложения на компьютере, уменьшить скорость обмена и повторить процесс обновления.
NO NETWORK SERVICE	Не связи с оператором GSM. Сообщение возвращается после включения модема или при выходе из зоны покрытия сети GSM.	Проверить не заблокирована ли SIM-карта, баланс должен быть положительным, переместить GSM-антенну в зону покрытия сети, после включения питания подождать 5-30 секунд.

8.2. Коды ошибок GSM/GPRS

Коды ошибок выполнения команды **AT+E2IPA** для модема **GR64** (M2M IP Activate, установить соединение с GPRS сетью).

<err> Description
000 No Error

008 Operator Determined Barring
 024 MBMS bearer capabilities insufficient for the service
 025 LLC or SMDCP failure(A/Gb mode only)
 026 Insufficient resources
 027 Missing or unknown APN
 028 Unknown PDP address or PDP type
 029 User authentication failed
 030 Activation rejected by GGSN
 031 Activation rejected, unspecified
 032 Service option not supported
 033 Requested service option not subscribed
 034 Service option temporarily out of order
 035 NSAPI already used (not sent)
 036 Regular deactivation
 037 QoS not accepted
 038 Network failure
 039 Reactivation required
 040 Feature not supported
 041 Semantic error in the TFT operation
 042 Syntactical error in the TFT operation
 043 Unknown PDP context
 044 PDP context without TFT already activated
 045 Multicast group membership time-out
 046 Semantic errors in packet filter(s)
 047 Syntactical errors in packet filter(s)
 081 Invalid transaction identifier value
 095 Semantically incorrect message
 096 Invalid mandatory information
 097 Message type non-existent or not implemented
 098 Message type not compatible with the protocol state
 099 Information element non-existent or not implemented
 100 Conditional IE error
 101 Message not compatible with the protocol state
 111 Protocol error, unspecified
 112 APN restriction value incompatible with active PDP context

Коды ошибок выполнения команды **AT+E2IPO** для модема **GR64** (AT+E2IPO IP Open/Connect, установить IP-соединение).

<err>	Description
0	IP OK – No Error
1	IP Host not found
2	GPRS Context is configured incorrectly, out of range, or doesn't exist.
3	Command aborted.
8	Open failed
9	Bind failed
10	Connect failed
11	listen failed

12	Accept failed
13	Close failed
14	Send to failed
15	Setopt failed
16	Unkown internal error
17	Fatal Error
18	Unknown error
19	destination unreachable
20	Not connected
21	Address in use
22	Connection aborted
23	Already Connected
24	Connection refused
25	Connection Timeout
32	Socket not found
33	Undefined operation
34	Bad socket type
35	Incompatible socket type
36	Too many sockets
37	Bad domain
38	Already bound
39	Already connected
40	Not bound and connected
41	No data
42	Send buffer full
43	Bad address parameter
44	Bad backlog parameter
45	Unknown socket type
64	DNS error - server not found
65	DNS error - socket error
66	DNS error - too many queries
67	DNS error - bad parameter
68	DNS error - bad address name
69	DNS error - bad host name
70	DNS error - bad response from server
71	DNS error - server unreachable
72	DNS error - timeout waiting for answer
252	IP Error Receiving IP Data
253	IP Error Max number of retries
254	IP Error GPRS Connection Lost
255	IP Error General Failure

8.3. Отладочные сообщения

Отладочные сообщения выводятся в *TE*-порт при включенном режиме «отладка» (см. третий параметр команды [&set.echo](#)). Вывод отладочных сообщений производится только в *TE*-порт. Отладочное сообщение начинается со знака «-» (минус), если это результат работы процессора *Устройства* или иначе, если это

АТ-команда, которая передается в GSM/GPRS-модуль или полученное от него (подробнее см. документ «GR47/GR48 AT Commands Manual»). АТ-команды передаваемые GSM/GPRS-модулю в отладочном режиме возвращаются *TE* (см. таблицу ниже). Например команда “АТ+СІND?” будет возвращена *TE* как “+СІND?” с соответствующим ответом. например:

```
+CIND?
+CIND: 4,1,0,0,0,0,0
OK
```

Таблица отладочных сообщений *TE*-порта *Устройства*

Сообщение TE	Комментарии
+CIND?	АТ-команда к GSM/GPRS-модулю для проверки состояния линии связи, если нет GPRS соединения.
+CIND: N,S,0,0,0,0,0	Сообщение от GSM/GPRS-модуля о состоянии подключения к сети GSM: N – качество сигнала от базовой станции (значение от 0 до 5, где 0 – нет связи); S – флаг подключения к сети GSM (0 – нет подключения, 1 – подключение установлено).
*E2IPA?	АТ-команда к GSM/GPRS-модулю для проверки состояния линии связи при активном GPRS соединении.
RING	Сообщение от GSM/GPRS-модуля: принят вызов от абонента
*ECAV:	Сообщение от GSM/GPRS-модуля. Параметры следующие за этим сообщением указывают на одно из событий: поступил вызов от абонента, вызов принят, вызов сброшен
H	АТ-команда к GSM/GPRS-модулю для сброса входящего вызова
A	АТ-команда к GSM/GPRS-модулю для приёма вызова
+CMGD	АТ-команда к GSM/GPRS-модулю для удаления входящего SMS сообщения из ME памяти. <i>Устройство</i> игнорирует вероятную ошибку выполнения команды, если в ME памяти нет записанных сообщений

Сообщение TE	Комментарии
*E2IPA=1,<i>action</i> *E2IPA=2,<i>action</i>	АТ-команда к GSM/GPRS-модулю для активирования (параметр <i><action></i> равен 1) или деактивирования (параметр <i><action></i> равен 0) встроенного IP-стека. Иначе, «выход в» или «выход из» сети GPRS. Первый параметр (цифра 1 или 2) соответствует номеру активной SIM-карты.
*E2IPRH="host_name"	Команда посылается в GSM/GPRS-модуль для преобразования имени хоста в IP-адрес.
*E2IPO=1,"ip_addr",port *E2IPO=2,"ip_addr",port	АТ-команда к GSM/GPRS-модулю для соединения с сервером по IP-адресу и номеру порта. Первый параметр (цифра 1 или 2) соответствует номеру активной SIM-карты.
CONNECT	Ответ от GSM/GPRS-модуля означает, что соединение с удалённым сервером (или модемом) установлено.
NO CARRIER	Ответ от GSM/GPRS-модуля означает, что соединение с удалённым сервером (или модемом) разорвано сервером (или модемом).
*E2OTR?	АТ-команда к GSM/GPRS-модулю с запросом на чтение показания встроенного термометра.
-?msg	Выводится при исполнении макро команды, если объект <i>Сообщение</i> с указанным номером не определён.
-?user	Выводится при исполнении макро команды, если объект <i>Пользователь</i> с указанным номером не определён.
-timeout	Превышено время ожидания ответа на АТ команду.
-issue msg	Сообщение будет отправлено серверу, модему или на телефон пользователя.
-sms #####> text	Вывод SMS сообщения с содержанием text .
-new user:	Вызов от незарегистрированного пользователя.
-user found	Вызов от зарегистрированного пользователя.
-sms sent ok	SMS сообщение отправлено
-DTR low	Аппаратный сброс сеанса IP- или CSD-связи. Сброс завершается выводом сообщения «-up».
-up	см. описание сообщения –DTR low
-gsm on	GSM/GPRS-модуль включается
-gsm off	GSM/GPRS-модуль выключается
-macro.N (obj.id.event)	Макро командой N зарегистрировано системное сообщение obj.id.event .

-ignored

а). Принятое NMEA-предложение игнорировано из-за ошибки контрольной суммы или отсутствия данного предложения в реестре разрешенных.

б). Принятая по SMS-каналу &-команда игнорируется из-за неактивного отправителя сообщения (см. описание режима обработки SMS-сообщения команды [&get.gsm](#)).

Приложение 9. Индикатор опций прошивки

Внимание! Версии прошивок до июня 2011 года возвращают код индикатора опций без префикса "<GSM><GPS><FLASH>".

Действительный код индикаторов опций прошивки возвращаются по команде [&get.system](#) в формате:

```
opt <GSM><GPS><FLASH>--<options>
```

или по команде [&get.version](#) с параметром all или option в формате:

```
<GSM><GPS><FLASH>--<options>
```

где

<GSM> - тип используемого GSM/GPRS-модема, одна буква: (Q)64 или G(R)64

<GPS> - тип используемого GPS-приёмника, одна буква: (V)enus или (E)T316

<FLASH> - объем встроенной FLASH-памяти, одна цифра: (8)МБ или 5(1)2Кб

Значения бит <options>	Наименование
0000-0000-0001	В составе структуры ЛОГ байт <macroId> - номер макро определения, которое инициировало запись данных в ЛОГ (1).
0000-0000-0002	В составе структуры ЛОГ значения параметров HGeo и HMET по два байта каждый (1).
0000-0000-0004	В составе структуры ЛОГ один байт значений состояний входов-выходов (1).
0000-0000-0008	В составе структуры ЛОГ два байта для значения показаний датчика температуры 1wirebus (1).
0000-0000-0070	Информационная ёмкость и количество счетных регистров в составе структуры ЛОГ: 2x32 бит (1); 4x32 бит (2); 2x16 бит (3); 4x16 бит (4); 1x32бит (5); не используется (6 и 7) (1).
0000-0000-0080	В составе структуры ЛОГ один байт значения средней скорости с момента последней записи данных в ЛОГ (1).
0000-0000-0100	Режим ГЛОНАСС/GPS (см. &set.nmea) (1).
0000-0000-0200	В составе структуры ЛОГ четыре байта значения программного гео-одометра (1).
0000-0000-0400	Режим поддержки расширенной модели событий (1).
0000-0000-0800	Поддержка тэгов Ключевых Слов (1).
0000-0000-1000	В составе структуры ЛОГ пять байт упакованных значений аналоговых сигналов (1).
0000-0000-2000	В составе структуры ЛОГ байт состояния GSM-модуля (1).
0000-0000-4000	В составе структуры ЛОГ значения параметров HDOP и NSAT информационной ёмкостью один байт каждый (1).
0000-0000-8000	В составе структуры ЛОГ значения параметров COG и SOG по два байта каждый (1).
0000-0001-0000	Поддержка режима работы с таксометром типа ИЗОТ (1).
0000-0002-0000	Поддержка протокола датчиков уровня НПП «СЕНСОР» (1).

0000-0004-0000	Поддержка режимов работы с пользовательскими Регистрами (1).
0000-0008-0000	(в разработке)
0000-0010-0000	Управление значением пользовательского Регистра по событию (1).
0000-0020-0000	Поддержка сигнального процессора (1).
0000-0040-0000	Поддержка дополнительного режима работы с дискретными входами – программный фильтр «дребезга» сигнала (1).
0000-0080-0000	Поддержка фильтра кратных «скачков» местоположения во время движения со скоростью ниже пороговой (1).
0000-0100-0000	Поддержка режима работы сканера штрих-кода с интерфейсом RS232 (1).
0000-0200-0000	Поддержка интерфейса RS485 (1).
0000-0400-0000	Поддержка Условия для объекта OBJ_GPS (1).
0000-0800-0000	Поддержка Условия для объекта OBJ_ZONE (1).
0000-1000-0000	Режим синхронизации передачи данных с тремя серверами. По умолчанию режим синхронизации передачи данных с одним сервером (1).
0000-2000-0000	Поддержка работы с адаптером CANlog M333 (1).
0000-4000-0000	Поддержка протокола Modbus RTU по интерфейсу RS485 (1).
0000-8000-0000	Поддержка протокола LLS/ДУЖ/УЗИ по интерфейсу RS485 (1).
0001-0000-0000	Поддержка режима возврата данных в формате NMEA GPIO.
0002-0000-0000	Поддержка режима возврата данных в формате NMEA GPGBA.
0004-0000-0000	Поддержка режима возврата данных в формате NMEA GPRMC.
0008-0000-0000	Поддержка режима внеочередной записи ЛОГ по событию "остановка" (параметр &set.log =,,,,,<null_speed_duration>).
0010-0000-0000	Поддержка режима управления тактовой частотой процессора по событию (см. объект OBJ_SYSTEM).
0020-0000-0000	Поддержка режима управления питанием встроенного GPS-приёмника по событию (см. объект OBJ_GPS).
0040-0000-0000	Поддержка режима возврата данных в двоичном представлении.
0080-0000-0000	Поддержка режима возврата данных в 16-ричном представлении.
0100-0000-0000	Поддержка проверки имени GSM-оператора связи (см. параметр <copid>, команда &set.gsm).
0200-0000-0000	Поддержка управления регистратора данных по событию (см. &set.log = <t1>,<F>,,,,,,,,<t2> и &set.echo =<period>,<Filter>) (1).
0400-0000-0000	Поддержка режима запроса кода IMEI и работы с тэгом %II
0800-0000-0000	Поддержка режима передачи данных по каналу CUSD
1000-0000-0000	Поддержка считывателей карт доступа eMarine и iButton по интерфейсу 1wirebus
2000-0000-0000	Поддержка работы со счётчиками в режиме Decrement (-1), &set.zone
4000-0000-0000	-
8000-0000-0000	-

Например:

opt QV1-0087-0284-F807

Приложение 10. Заметки администратора

Дата	Комментарий