

ОПИСАНИЕ ПРОТОКОЛА V1310CICADA

1. Структура HTTP- пакета при передаче

При передаче данных используется метод POST. Данные передаются в текстовом виде. Пакет имеет следующую структуру:

POST http://<HOST>/<SCRIPT> HTTP/1.1\r\n

Host: <HOST>\r\n

Content-Type: application/x-www-form-urlencoded\r\n

Content-Length: <LENGTH>\r\n

\r\n

IMEI=<IMEI>&Строка с данными 1&Строка с данными 2

.....

&Строка с данными N

Где:

<HOST> - IP-адрес и название сервера

<SCRIPT> - полный путь к скрипту и его название

<LENGTH> - длина передаваемых данных

<IMEI> - IMEI устройства

2. Структура строк с данными

В строках с данными передаётся информация, необходимая для вычисления местоположения по базовым станциям, текущие координаты по GPS, состояние внутренних датчиков устройства и его настройки. В пакете всегда присутствуют строки +COPS, #INFO. Если есть сигнал GPS, в пакете могут отсутствовать строки с информацией о базовых станциях (#MONI). Если сигнал GPS отсутствует, то в пакете могут отсутствовать строки с GPS-данными (\$GPGGA, \$GPVTG, \$GPZDA, \$GPGSA).

2.1. Передача данных о базовых станциях

В этих строках основными являются параметры MCC, MNC, LAC, CELLID, TA.

Выдача MCC+MNC выполняется с помощью строки

&COPS=0,2,"MCC+MNC"

Например:

&COPS=0,2,"25506"

Где:

MCC = 255, MNC = 06.

Передача данных о базовых станциях (LAC, CELLID, TA) выполняется с помощью строки:

&BSMn=<bsic> <LAC> <CELLID> <arfcn> <dBm> <C1value> <C2value>

<TA> <qual> <netname>

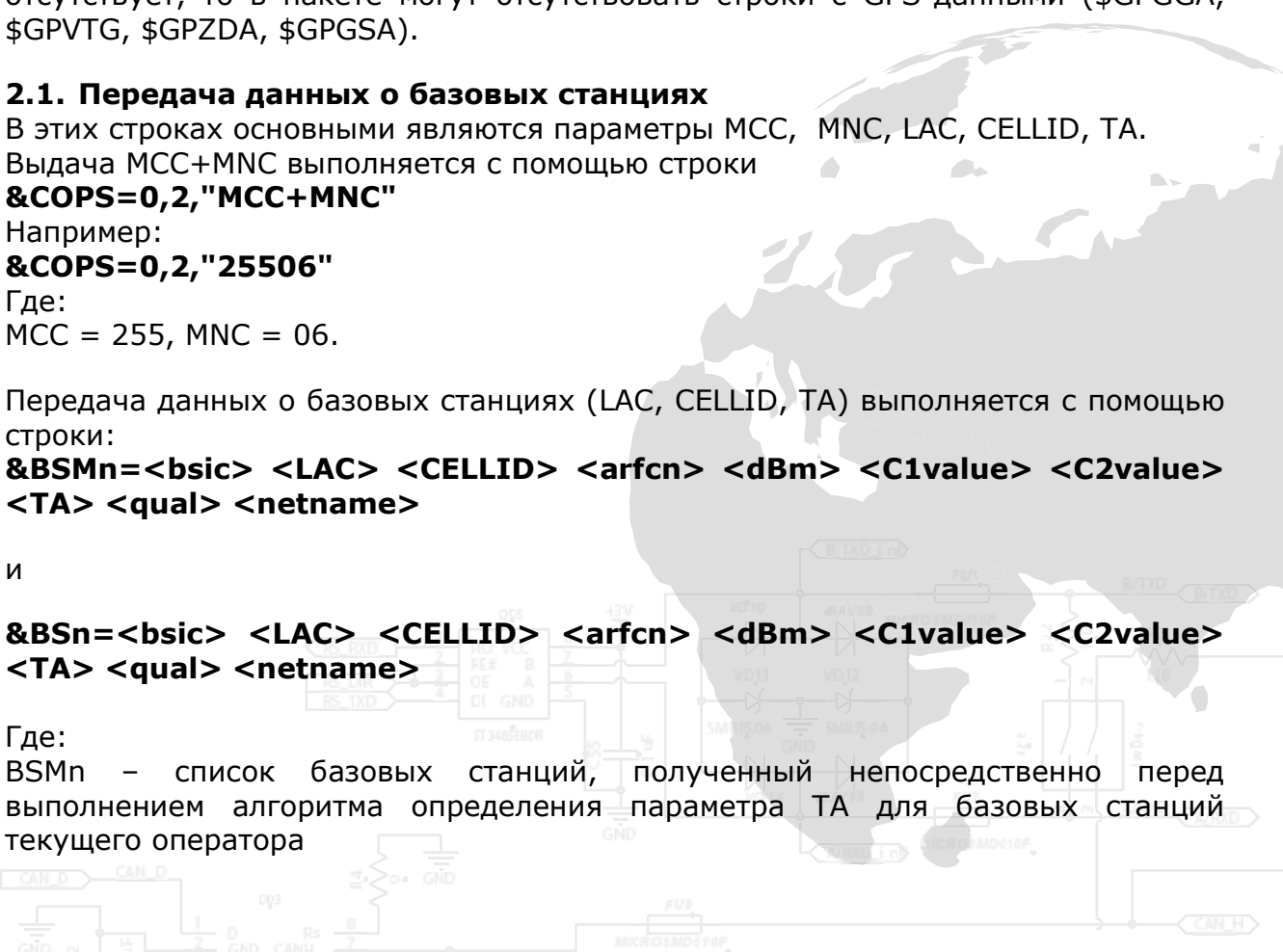
и

&BSn=<bsic> <LAC> <CELLID> <arfcn> <dBm> <C1value> <C2value>

<TA> <qual> <netname>

Где:

BSMn - список базовых станций, полученный непосредственно перед выполнением алгоритма определения параметра TA для базовых станций текущего оператора



BSn – список базовых станций с обновлёнными параметрами TA
n – индекс базовой станции 0-6

Например:

&BS0=56 2AA4 0E95 696 -86dbm 24 24 1 0 UA 06

Где:

LAC = 2AA4 (передается в HEX)

CELLID = 0E95 (передается в HEX)

TA = 1 (передается в DEC)

Возможна передача до 7ми строк с информацией о базовых станциях.

При получении строк BSMn поле **<netname>** присутствует только в строке BSM0. В строках BSM1-BSM6 это поле может отсутствовать.

В строках BS0-BS6 это поле присутствует всегда.

Все поля отделены друг от друга одним пробелом.

2.2. Передача GPS-данных

В качестве GPS-данных передаются строки GPGGA, GPVTG, GPZDA и GPGSA.

Подробное описание всех GPS-строк можно найти в **"The NMEA 0183 Protocol"**

Пример строк с GPS-координатами:

&GPGGA=084303.000,5025.1979,N,03025.6670,E,1,8,1.03,172.2,M,27.9,M,,*57

&GPVTG=309.71,T,,M,0.09,N,0.17,K,A*3E

&GPZDA=084303.000,20,05,2010,,*5E

&GPGSA=A,3,05,28,07,19,13,,,,,,,,,1.73,1.46,0.92*0D

2.3. Передача состояния внутренних датчиков и параметров устройства

Устройство может передавать на сервер ряд внутренних датчиков и параметров.

Датчики или параметры могут быть числовые или строковые.

Все строковые значения берутся в кавычки "...".

Пустые строковые значения передаются как "".

Общий формат передачи датчиков выглядит так:

&sens[ID]=x

Общий формат передачи параметров выглядит так:

&par[ID]=x

Где:

sens – передача датчиков

par – передача параметров

ID - индекс датчика или параметра, представляет собой 4 символа == ID параметра. Т.к. ID датчиков или параметров 3значное число, то ID будет всегда записываться как 0xxx

x – значение датчика или параметра

Пример передачи некоторых датчиков и параметров:

&sens[0100]=1&sens[0200]="GPS MAYAK VER

1.25"&par[0201]=300&par[0504]="www.google.com"

В данном примере отправляется:

источник сброса - датчик, ID=100, значение 1,

версия ПО - датчик, ID=200, значение "GPS MAYAK VER 1.25",

таймаут ожидания GPS-координат — параметр, ID=201, значение 300 секунд

название сервера - параметр, ID=504, значение "www.google.com"

В ответе, передаваемом устройством на сервер, будут присутствовать значения всех датчиков и параметров, описанные выше.

3. Структура ответа

После передачи пакета на сервер, ожидается ответ о подтверждении приёма данных со следующей структурой:

HTTP/1.1 200 OK\r\n

Content-Length: X\r\n

\r\n

STATUS=RECEIVED&setparam

xxxx

yyyy&setparam

xxxx

yyyy&cmd0&cmdx\r\n

Где:

X – длина данных

STATUS=RECEIVED – статус того, что сервер принял данные

&setparam xxxx yyyy – строка настройки параметров

&cmd0, &cmdx – дополнительные команды

\r\n – признак завершения строки с командами (если они присутствуют)

В строке настройки параметров xxxx – ID параметра, yyyy – значение параметра.

Команда &setparam отделяется от ID параметра xxxx одним пробелом, ID параметра xxxx отделяется от значения параметра yyyy одним пробелом.

ID параметра (смотри описание в таблице **ID параметров**) представляет собой 4 символа == ID параметра. Т.к. ID параметров 3значное число, то xxxx будет всегда записываться как 0xxx

yyyy – значение параметра. Числовая или строковая величина. **Строковые величины не берутся в кавычки "...". Если необходимо установить пустой строковый параметр, то записывается в виде "&setparam 0xxx ", т.е. признаком того что данный строковый параметр пустая строка является пробел и отсутствие данных после пробела.**

Устройство распознаёт следующие дополнительные команды – "loadparam", "saveparam", "cpureset", "BOOT host,port,path".

"loadparam" – принудительная загрузка параметров из энергонезависимой памяти в оперативную память

"saveparam" – принудительное сохранение параметров из оперативной памяти в энергонезависимую память

"cpureset" – немедленная перезагрузка процессора

"BOOT host,port,path" – команда загрузки обновления ПО

В процессе работы устройство автоматически загружает все параметры из энергонезависимой памяти в оперативную и сохраняет все изменения в параметрах после каждой успешной передачи данных. Поэтому с целью сбережения ресурса энергонезависимой памяти нет необходимости в дополнительном сохранении параметров с помощью команды "saveparam".

Пример типичного ответа:

HTTP/1.1 200 OK\r\n

Content-Length: X\r\n

\r\n

STATUS=RECEIVED

Данные приняты.

Пример ответа с установкой параметров GPS COORD TIMEOUT, HOST, PORT а также с необязательной командой сохранения изменений в энергонезависимую память:

```
HTTP/1.1 200 OK\r\n
Content-Length: X\r\n
\r\n
STATUS=RECEIVED&setparam          0201          300&setparam          0504
biakom.com&setparam 0200 80&saveparam\r\n
```

После отправки пакета подтверждения приёма сервер должен автоматически разорвать соединение с устройством.

4. Команда обновления ПО – “BOOT host,port,path”

Для обновления ПО предусмотрена команда BOOT. Эта команда содержит IP-адрес или название сервера, на котором находится файл обновления ПО, порт сервера и путь к файлу. Параметры команды отделены от ключевого слова BOOT одним пробелом. Друг от друга параметры отделяются запятыми. Длина параметров **host** и **path** не должна превышать 46 символов, **port** – 6 символов. Для возможности загрузки обновления ПО сервер должен поддерживать передачу файлов по частям (partial get).

Пример ответа с командой обновления ПО:

```
HTTP/1.1 200 OK\r\n
Content-Length: X\r\n
\r\n
STATUS=RECEIVED&setparam          0201          300&setparam          0504
biakom.com&setparam                0200
biakom.com,80,pr/gps_mayak_1_32.bin\r\n
```

