

**КОМУНІКАЦІЙНИЙ ПРОТОКОЛ
WIALON IPS v.2.0**

Версія	Дата	Зміни
2.0	10.2014	У пакет L (пакет логіну) додано поле «версія протоколу». У пакети L, SD, D, M, I, IT, T, US, UC додано поле контрольної суми CRC16. Для пакетів AL, ASD, AD, AM, AI, AIT, AT доданий код помилки перевірки контрольної суми.

Вхідні дані TCP протоколу

Всі дані приходять в текстовому форматі і являють собою пакет наступного виду:
#TP#msg\r\n

#	стартовий байт
TP	тип пакету, опис всіляких типів зазначено в табл. 1
#	роздільник
msg	повідомлення
\r\n	кінцевик пакету

**Типи пакетів
(таблиця 1)**

Тип	Опис	Відправник
L	пакет логіна	обладнання
AL	відповідь на пакет логіна	сервер
D	пакет з даними	обладнання
AD	відповідь на пакет з даними	сервер
P	пинговий пакет	обладнання
AP	відповідь на пинговий пакет	сервер
SD	скорочений пакет з даними	обладнання
ASD	відповідь на скорочений пакет	сервер
B	пакет з чорним ящиком	обладнання
AB	відповідь на пакет з чорним ящиком	сервер
M	повідомлення для водія	обладнання/сервер
AM	відповідь на повідомлення від водія	сервер
US	пакет з новою прошивкою	сервер
UC	пакет з файлом конфігурації	сервер



Пакет логіна

#L#protocol_version;imei;password;crc16\r\n

;	роздільник
protocol_version	Версія протоколу. Поле повинно містити значення
imei	2.0
password	унікальний ID трекара, IMEI або серійний номер
crc16	пароль для доступу до пристрою, при відсутності передається NA

У відповідь на пакет логіна сервер посилає команду AL:

"1" – у разі успішної авторизації на сервер;
"0" – при відмові від сервера підключення;
"01" – при помилці перевірки пароля;
"10" – при помилці перевірки контрольної суми.

Приклади:

#AL#1\r\n #AL#0\r\n

Скорочений пакет з даними

#SD#date;time;lat1;lat2;lon1;lon2;speed;course;height;sats;crc16\r\n

date	дата у форматі DDMMYY UTC, при відсутності передається NA
time	час в форматі HHMMSS UTC, при відсутності передається NA
lat1;lat2	широта (5544.6025;N), при відсутності передається NA;NA
lon1;lon2	довгота (03739.6834;E), за відсутності передається NA;NA
speed	швидкість, ціле число, км/год, при відсутності передається NA
course	курс, ціле число, градуси, при відсутності передається NA
height	висота, ціле число, в метрах, при відсутності передається NA
sats	кількість супутників, ціле число, при відсутності передається NA
crc16	контрольна сума (див. Додаток 1)

Якщо поля date і time містять NA, використовується поточний час сервера.

У відповідь на скорочений пакет з даними сервер посилає команду ASD:

"-1" – помилка структури пакета
"0" – некоректне час
"1" – пакет успішно зафіксувався
"10" – помилка отримання координат
"11" – помилка отримання швидкості, курсу або висоти
"12" – помилка отримання кількості супутників
"13" – помилка перевірки контрольної суми.

Приклади:

#ASD#1\r\n #ASD#0\r\n #ASD#10\r\n #ASD#11\r\n #ASD#12\r\n

Пакет з даними

#D#date;time;lat1;lat2;lon1;lon2;speed;course;height;sats;hdop;inputs;outputs;adc;ibutton;params;crc16\r\n

date	дата у форматі DDMMYY UTC, при відсутності передається NA
time	час в форматі HHMMSS UTC, при відсутності передається NA
lat1;lat2	широта (5544.6025;N), при відсутності передається NA;NA
lon1;lon2	довгота (03739.6834;E), за відсутності передається NA;NA
speed	швидкість, ціле число, км/год, при відсутності передається NA
course	курс, ціле число, градуси, при відсутності передається NA
height	висота, ціле число, в метрах, при відсутності передається NA
sats	кількість супутників, ціле число, при відсутності передається NA
hdop	зниження точності, дробове число, при відсутності передається NA
inputs	цифрові входи, кожен біт числа відповідає одному входу, починаючи з молодшого, ціле число, при відсутності передається NA
outputs	цифрові виходи, кожен біт числа відповідає одному виходу, починаючи з молодшого, ціле число, при відсутності передається NA
adc	аналогові входи, дробові числа, через кому. Нумерація виходу починається з одиниці; Якщо аналогових входів немає - передається порожній рядок.
ibutton	Приклад: 14.77,0.02,3.6 код ключа водія, рядок довільної довжини. При відсутності ключа передається NA
params	набір додаткових параметрів через кому. Кожен параметр являє собою конструкцію NAME:TYPE:VALUE NAME – довільна рядок, довжиною не більше 15 байт; TYPE –тип параметра, 1 –int/long long, 2 –double, 3 –string VALUE – значення в залежності від типу Для передачі тривожної кнопки використовується параметр першого типу з ім'ям «SOS», значення 1 означає натиснення тривожної кнопки. Для передавання текстового повідомлення використовується параметр третього типу (string) з іменем text». Цей параметр можна використовувати для передавання текстового повідомлення від водія, яке може містити координати і інші параметри. Приклади: count1:1:564,fuel:2:45.8,hw:3:V4.5 SOS:1:1
crc16	контрольна сума (див. Додаток 1)

Якщо поля date і time містять значення NA, використовується поточний час сервера.

У відповідь на пакет з даними сервер посилає команду ОГОЛОШЕННЯ:

"-1" – помилка структури пакета

"0" – некоректне час

"1" – пакет успішно зафіксувався

"10" – помилка отримання координат

"11" – помилка отримання швидкості, курсу або висоти

"12" – помилка отримання кількості супутників або hdop

Служить для підтримки активного TCP-з'єднання з сервером, а також для перевірки працездатності каналу.

У відповідь на пінговий пакет сервер посилає команду AR.

71 7.0571 Приклад:

```
09 10-409 #AP#\r\n
```

Пакет з чорного ящика

```
47 11,347 #B#msg|crc16\r\n
```

Пакет з чорним ящиком призначений для передачі повідомлень одночасно.

"msg" являє собою кілька тіл скорочених чи повних пакетів (без зазначення типу), розділених між собою символом '|'.
 14-053
 11 6-5911

"crc16" контрольна сума (див. Додаток 1)

Приклад:

```
#B#date;time;lat1;lat2;lon1;lon2;speed;course;height;sats|date;time;lat1;lat2;lon1  
;lon2;speed;course;height;sats|date;time;lat1;lat2;lon1;lon2;speed;course;height;s  
ats|crc16\r\n
```

У відповідь на пакет з чорним ящиком сервер посилає команду АВ, де вказується кількість зафіксованих повідомлень:

Приклади: #AB#3\r\n #AB#0\r\n

або порожній рядок, яка означає помилку перевірки контрольної суми:
#AB#\r\n

Повідомлення для водія

```
#M#msg;crc16\r\n
```

Служить для відправки текстового повідомлення водієві.

"msg" - безпосередньо текст повідомлення. Повідомлення може відправлятися як сервером, так і обладнанням

"crc16" контрольна сума (див. Додаток 1)

У відповідь на повідомлення для водія сервер посилає команду AM:

"1" –повідомлення прийнято

"0" – помилка прийняття повідомлення

"01" – помилка перевірки контрольної суми

Приклади: #AM#1\r\n #AM#0\r\n

Пакет с новою прошивкою

Служить для надсилання нової прошивки на трекер.

#US#sz;crc16\r\nBIN

sz	розмір бінарних даних пакету (наприклад, 51200 байт)
crc16	контрольна сума (див. Додаток 1)
BIN	прошивка в бінарному вигляді

Пакет с файлом конфігурації

Служить для відправки файлу конфігурації на трекер.

#UC#sz;crc16\r\nBIN

sz	розмір файла конфігурації, байт
crc16	контрольна сума (див. Додаток 1)
BIN	вміст файла конфігурації

Поле crc16 повинно містити значення контрольної суми в шістнадцятковому вигляді у форматі big-endian, без провідних нулів зліва, наприклад:

AA13BB, що в десятковому форматі являє собою число 11146171.

У пакетах L, SD, D, B, M для розрахунку crc16 береться частина пакета між #TP# полем crc16.

Приклад пакету:

```
#SD#date;time;lat1;lat2;lon1;lon2;speed;course;height;sats;crc16\r\n
```

В даному випадку crc16 розраховується для наступної частини пакета:

```
date;time;lat1;lat2;lon1;lon2;speed;course;height;sats;
```

Приклад пакету:

```
#B#date;time;lat1;lat2;lon1;lon2;speed;course;height;sats|date;time;lat1;lat2;lon1;lon2;speed;course;height;sats|crc16\r\n
```

В даному випадку crc16 розраховується для наступної частини пакета:

```
date;time;lat1;lat2;lon1;lon2;speed;course;height;sats|date;time;lat1;lat2;lon1;lon2;speed;course;height;sats|
```

У пакетах I, US, UC, T для розрахунку crc16 береться поле BIN.

Приклад пакету:

```
#I#51200;0;1;070512;124010;sample.jpg;crc16\r\nBIN
```

В даному випадку crc16 розраховується для поля BIN.

Приклад коду на мові C для розрахунку crc16:

```
static const unsigned short crc16_table[256] =
{
    0x0000,0xC0C1,0xC181,0x0140,0xC301,0x03C0,0x0280,0xC241,0xC601,0x06C0,0x0780,0xC741,
    0x0500,0xC5C1,0xC481,0x0440,0xCC01,0x0CC0,0x0D80,0xCD41,0x0F00,0xCFC1,0xCE81,0x0E40,
    0x0A00,0xCAC1,0xCB81,0x0B40,0xC901,0x09C0,0x0880,0xC841,0xD801,0x18C0,0x1980,0xD941,
    0x1B00,0xDBC1,0xDA81,0x1A40,0x1E00,0xDEC1,0xDF81,0x1F40,0xDD01,0x1DC0,0x1C80,0xDC41,
    0x1400,0xD4C1,0xD581,0x1540,0xD701,0x17C0,0x1680,0xD641,0xD201,0x12C0,0x1380,0xD341,
    0x1100,0xD1C1,0xD081,0x1040,0xF001,0x30C0,0x3180,0xF141,0x3300,0xF3C1,0xF281,0x3240,
    0x3600,0xF6C1,0xF781,0x3740,0xF501,0x35C0,0x3480,0xF441,0x3C00,0xFCC1,0xFD81,0x3D40,
    0xFF01,0x3FC0,0x3E80,0xFE41,0xFA01,0x3AC0,0x3B80,0xFB41,0x3900,0xF9C1,0xF881,0x3840,
    0x2800,0xE8C1,0xE981,0x2940,0xEB01,0x2BC0,0x2A80,0xEA41,0xEE01,0x2EC0,0x2F80,0xEF41,
    0x2D00,0xEDC1,0xED81,0x2C40,0xE401,0x24C0,0x2580,0xE541,0x2700,0xE7C1,0xE681,0x2640,
    0x2200,0xE2C1,0xE381,0x2340,0xE101,0x21C0,0x2080,0xE041,0xA001,0x60C0,0x6180,0xA141,
    0x6300,0xA3C1,0xA281,0x6240,0x6600,0xA6C1,0xA781,0x6740,0xA501,0x65C0,0x6480,0xA441,
    0x6C00,0xACC1,0xAD81,0x6D40,0xAF01,0x6FC0,0x6E80,0xAE41,0xAA01,0x6AC0,0x6B80,0xAB41,
    0x6900,0xA9C1,0xA881,0x6840,0x7800,0xB8C1,0xB981,0x7940,0xBB01,0x7BC0,0x7A80,0xBA41,
    0xBE01,0x7EC0,0x7F80,0xBF41,0x7D00,0xBDC1,0xBC81,0x7C40,0xB401,0x74C0,0x7580,0xB541,
    0x7700,0xB7C1,0xB681,0x7640,0x7200,0xB2C1,0xB381,0x7340,0xB101,0x71C0,0x7080,0xB041,
    0x5000,0x90C1,0x9181,0x5140,0x9301,0x53C0,0x5280,0x9241,0x9601,0x56C0,0x5780,0x9741,
    0x5500,0x95C1,0x9481,0x5440,0x9C01,0x5CC0,0x5D80,0x9D41,0x5F00,0x9FC1,0x9E81,0x5E40,
    0x5A00,0x9AC1,0x9B81,0x5B40,0x9901,0x59C0,0x5880,0x9841,0x8801,0x48C0,0x4980,0x8941,
    0x4B00,0x8BC1,0x8A81,0x4A40,0x4E00,0x8EC1,0x8F81,0x4F40,0x8D01,0x4DC0,0x4C80,0x8C41,
    0x4400,0x84C1,0x8581,0x4540,0x8701,0x47C0,0x4680,0x8641,0x8201,0x42C0,0x4380,0x8341,
    0x4100,0x81C1,0x8081,0x4040
};
unsigned short crc16 (const void *data, unsigned data_size)
{
    if (!data || !data_size)
        return 0;
    unsigned short crc = 0;
    unsigned char* buf = (unsigned char*)data;
    while (data_size--)
        crc = (crc >> 8) ^ crc16_table[(unsigned char)crc ^ *buf++];
    return crc;
}
```