# Система локального позиционирования

**Описание взаимодействия сервера и базовых станций**

Разработал: Д.С. Зенков

## Введение

### Термины и определения

* *RTLS –* система позиционирования объектов в режиме реального времени
* *Базовая станция (БС, «якорь»)* – устройство, которое взаимодействует с метками в процессе определения координат последних. Базовые станции имеют фиксированные координаты, относительно которых определяются координаты меток. Базовые станции располагаются так, чтобы в любой точке контролируемой территории метка могла «видеть» минимум три базовые станции.
* *Метка* — радиоэлектронное устройство, которые прикрепляются к контролируемым объектам и взаимодействуют с RTLS.
* *Серверное программное обеспечение*— программное обеспечение, обеспечивающее управление процессом измерений, расчет координат объектов, обработку и накопление данных.

### Назначение документа

Документ определяет состав команд и их формат. Протокол описанный в данном документе позволяет конфигурировать устройства системы позиционирования, а также получать локационные данные от БС.

## Формат и перечень сообщений

Данные, между датчиком и внешним устройством передаются в виде сообщений стандартного формата (Таблица 3.1). Сообщение передаётся байтами

Таблица 3.1

|  |  |  |  |
| --- | --- | --- | --- |
| Порядковый номер поля | Название поля | Размер поля, байт | Описание |
| 1 | Префикс | 1 | Поле является маркером начала сообщения.  Входящие от сервера сообщения от должны иметь префикс 94h, а исходящие сообщения от БС должны выдаваться программой с префиксом 9Dh. |
| 2 | Сетевой адрес | 1 | Поле содержит:  - для префикса 94h сетевой адрес получателя сообщения;  - для префикса 9Dh сетевой адрес отправителя сообщения.  Адрес со значением 255 считается широкополосным. |
| 3 | Код операции | 1 | Поле содержит:  - для префикса 94h код операции, которую должна выполнить БС;  - для префикса 9Dh код операции, на которую выдаётся ответ. |
| 4 | Данные | Зависит от кода операции | Состав данных и формат поля зависит от кода операции. |
| 5 | Контрольная сумма | 1 | Поле используется для контроля целостности данных.  Алгоритм вычисления описан в разделе 5. |

Перечень сообщений приведён в таблице 3.2.

Таблица 3.1

|  |  |  |
| --- | --- | --- |
| **№** | **Код операции** | **Описание операции** |
| 1 | 06h | Считать сетевые настройки |
| 2 | 07h | Записать сетевые настройки |
| 3 | 08h | Считать настройки радиомодуля |
| 4 | 09h | Записать настройки радиомодуля |
| 5 | 0Ah | Считать настройки якоря |
| 6 | 0Bh | Установить настройки якоря |
| 5 | 0Сh | Считать настройки метки |
| 7 | 0Dh | Установить настройки метки |
| 9 | 0Eh | Считать локационные данные (формат автовыдачи) |

## Описание команд

### Считать сетевые настройки (06h)

Команда предназначена для чтения следующих сетевых настроек:

|  |  |  |
| --- | --- | --- |
| № | Настройка | Значение по умолчанию |
| 1 | IP-адрес устройства | 192.168.1.50 |
| 2 | Маска подсети | 255.255.255.0 |
| 3 | IP-адрес сервера | 192.168.1.10 |
| 4 | Порт сервера | 3000 |

После получения команды БС отправляет ответ с текущими настройками.

*Формат команды от сервера*

|  |  |  |  |
| --- | --- | --- | --- |
| Смещение, байт | Размер поля, байт | Значение | Описание |
| 0 | 1 | 94h | Префикс |
| +1 | 1 | 00h…FFh | Сетевой адрес получателя. |
| +2 | 1 | 06h | Код операции. |
| +3 | 1 | 00h…FFh | Контрольная сумма. |

*Формат ответа устройства*

|  |  |  |  |
| --- | --- | --- | --- |
| Смещение, байт | Размер поля, байт | Значение | Описание |
| 0 | 1 | 9Dh | Префикс |
| +1 | 1 | 00h…FFh | Сетевой адрес получателя. |
| +2 | 1 | 06h | Код операции. |
| +3 | 4 | [b0-b3]: 00h…FFh | IP-адрес устройства: b0.b1.b2.b3 |
| +7 | 4 | [b0-b3]: 00h…FFh | Маска подсети: b0.b1.b2.b3 |
| +11 | 4 | [b0-b3]: 00h…FFh | IP-адрес сервера: b0.b1.b2.b3 |
| +15 | 2 | 0000h…FFFFh | Порт сервера |
| +17 | 1 | 00h…FFh | Контрольная сумма. |

### Записать сетевые настройки (07h)

Команда предназначена для установки следующих сетевых настроек: IP-адрес устройства, маска подсети, ip-адрес и порт сервера. После получения команды БС отправляет сообщение об успешной обработке команды, перенастривает сетевые параметры и производит перезагрузку сетевых служб.

*Формат команды от сервера*

|  |  |  |  |
| --- | --- | --- | --- |
| Смещение, байт | Размер поля, байт | Значение | Описание |
| 0 | 1 | 94h | Префикс |
| +1 | 1 | 00h…FFh | Сетевой адрес получателя. |
| +2 | 1 | 07h | Код операции. |
| +3 | 4 | [b0-b3]: 00h…FFh | IP-адрес устройства: b0.b1.b2.b3 |
| +7 | 4 | [b0-b3]: 00h…FFh | Маска подсети: b0.b1.b2.b3 |
| +11 | 4 | [b0-b3]: 00h…FFh | IP-адрес сервера: b0.b1.b2.b3 |
| +15 | 2 | 0000h…FFFFh | Порт сервера |
| +17 | 1 | 00h…FFh | Контрольная сумма. |

*Формат ответа устройства*

|  |  |  |  |
| --- | --- | --- | --- |
| Смещение, байт | Размер поля, байт | Значение | Описание |
| 0 | 1 | 9Dh | Префикс |
| +1 | 1 | 00h…FFh | Сетевой адрес получателя. |
| +2 | 1 | 07h | Код операции. |
| +3 | 1 | 00h | Команда выполнена успешно |
| 01h | Команда не может быть выполнена |
| +4 | 1 | 00h…FFh | Контрольная сумма. |

### Считать настройки радиомодуля (08h)

Команда предназначена для чтения следующих настроек радиомодуля:

|  |  |  |
| --- | --- | --- |
| № | Настройка | Значение по умолчанию |
| 1 | Сhannel number |  |
| 2 | Pulse Repetition Frequency |  |
| 3 | TX Preamble Length |  |
| 4 | rxPAC (Acquisition Chunk Size) |  |
| 5 | TX Preamble Code |  |
| 6 | RX Preamble Code |  |
| 7 | Non-std SFD |  |
| 8 | Data Rate |  |
| 9 | PHT Mode |  |
| 10 | SFD Timeout Value |  |

После получения команды БС отправляет ответ с текущими настройками.

*Формат команды от сервера*

|  |  |  |  |
| --- | --- | --- | --- |
| Смещение, байт | Размер поля, байт | Значение | Описание |
| 0 | 1 | 94h | Префикс |
| +1 | 1 | 00h…FFh | Сетевой адрес получателя. |
| +2 | 1 | 08h | Код операции. |
| +3 | 1 | 00h…FFh | Контрольная сумма. |

*Формат ответа устройства*

|  |  |  |  |
| --- | --- | --- | --- |
| Смещение, байт | Размер поля, байт | Значение | Описание |
| 0 | 1 | 9Dh | Префикс |
| +1 | 1 | 00h…FFh | Сетевой адрес получателя. |
| +2 | 1 | 06h | Код операции. |
| +3 | 1 |  | Сhannel number |
| +4 | 1 |  | Pulse Repetition Frequency |
| +5 | 1 |  | TX Preamble Length |
| +6 | 1 |  | rxPAC (Acquisition Chunk Size) |
| +7 | 1 |  | TX Preamble Code |
| +8 | 1 |  | RX Preamble Code |
| +9 | 1 |  | Non-std SFD |
| +10 | 1 |  | Data Rate |
| +11 | 1 |  | PHT Mode |
| +12 | 1 |  | SFD Timeout Value |
| +17 | 1 | 00h…FFh | Контрольная сумма. |

### Записать настройки радиомодуля (09h)

### Считать настройки якоря (0Ah)

### Установить настройки якоря (0Bh)

### Считать настройки метки

### Установить настройки метки

### Считать локационные данные (формат авто-выдачи данных) (0Eh)

Команда используется БС для выдачи данных на сервер. В случае, если период выдачи не равен нулю, данные выдаются автоматически, без запроса от сервера.

*Формат выдачи данных от БС*

|  |  |  |  |
| --- | --- | --- | --- |
| Смещение, байт | Размер поля, байт | Значение | Описание |
| 0 | 1 | 9Dh | Префикс |
| +1 | 1 | 00h…FFh | Сетевой адрес получателя. |
| +2 | 1 | 0Сh | Код операции. |
| +3 | 2 | 0000h…FFFFh | Идентификатор метки. |
| +5 | 2 | 0000h…FFFFh | Номер сообщения |
| +5 | 4 |  | Временная метка |
| +9 | 1 | 00h…FFh | Контрольная сумма. |

*Описание полей:*

* Идентификатор метки – имя метки, относительно которой измеряется и предается якорем временная метка.
* Номер сообщения – сквозной номер сообщения. Номер задается меткой и отправляется в пакете с blink-сигналом по радиосети.
* Временная метка – основная информация, по которой рассчитывается позиция метки. Временные метки засекаются якорями в момент получения blink-сигнала от метки. При известной разницы между временными метками от одной метки до 4-ех якорей и известном местоположении якоря, можно рассчитать местоположение метку.

## Алгоритм вычисления контрольной суммы

|  |
| --- |
| CRC8 |
| #define CRC8\_POLY 0x8B  U08 CRC\_08 (U08 init, U08\* pBuf, U16 len)  {  U08 crc = init;    while (len--)  {  crc = crc ^ \*pBuf++;  for (U08 i=0; i < 8; i++)  crc = (crc & 0x80) ? ((crc << 1) ^ CRC8\_POLY) : (crc << 1);  }    return crc;  } |

|  |
| --- |
| CRC16 |
| #define CRC16\_POLY 0x1021  U16 CRC\_16 (U16 init, U08\* pBuf, U16 len)  {  U16 crc = init;  while (len--)  {  crc ^= \*pBuf++ << 8;  for (U08 index = 0; index < 8; index++)  crc = crc & 0x8000 ? (crc << 1) ^ CRC16\_POLY : crc << 1;  }  return crc;  } |