Протокол обмена микроконтроллера STM32F071CBU6 c процессором NT1004

Версия 1.00

21-01-2016

# История изменений документа

|  |  |  |
| --- | --- | --- |
| **Дата** | **№ версии** | **Описание изменений** |
| 21.01.2016 | 1.00 | Начальное наполнение документа |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Требования к радиомодулю, протокол обмена

Процессор радиомодуля STM32F071CBU6 должен быть подключен к центральному процессору NT1004 при помощи интерфейса UART.

Требования к GUI центрального процессора NT1004

GUI должна предусматривать возможность выбора режимов работы радиомодуля

- режим обмена голосом

- режим обмена данными

- режим обмена поправками RTK

Возможность ввода частоты приема и частоты передачи в диапазоне 410МГц – 480 МГц, с шагом кратным 25 кГц.

Возможность выбора режима усилителя мощности: номинальная или повышенная

Уровень приема RSSI приемника

Громкость динамика

Чувствительность микрофона

Возможность выбора источника данных (файла) в режиме передачи данных

Возможность отображения текущего состояния (прием, передача, дежурный прием)

Возможность отображения ошибки

Возможность отображения (по запросу) версии ПО микроконтроллера STM

Что еще добавить?

Как должен работать в режиме RTK?

**Протокол обмена между процессорами**

Стандартный тип – UART.

Скорость обмена – до 115200 бит/с

Ведущим устройством на шине является процессор NT1004, ведомым – микроконтроллер STM радиомодуля.

Функции, которые должен обеспечивать интерфейс

- усыплять/будить радиомодуль

- задавать режим функционирования радиомодуля

- управлять параметрами радиомодуля

- получать параметры от радиомодуля

- ?

Для реализации предполагается следующий логический интерфейс:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| FEND | LEN | MSG | ID | Data……………….data | CRC | FEND |

FEND = C0. 1 байт-признак начала обмена

FEND = C0 1 байт-признак конца сообщения

Если в потоке данных (в заголовке, в теле или в контрольной сумме сообщения) встречаются байты, значения которых совпадают с управляющими кодами, производится подмена этих байт ESC-последовательностями (механизм байт-стаффинга, byte stuffing). Код FEND заменяется последовательностью <FESC>, <TFEND>, а код FESC – последовательностью <FESC>, <TFESC>, где TFEND = DCh (Transposed FEND), TFESC = DDh (Transposed FESC). Коды TFEND и TFESC являются управляющими только в ESC-последовательностях, поэтому при передаче данных они в подмене не нуждаются.

<FESC> = DBh

LEN – 1 байт содержащий значение количества байт данных поля data. Количество байт данных не должно превышать 128 байт

MSG – 1 байт, код сообщения.

Структура байта MSG:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Адрес | | | | Номер сообщения | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |

Адрес сообщения (4 бит)

«01» - микроконтроллер STM;

«10» - процессор NT1004.

Остальные значения зарезервированы для диагностики отдельных микросхем в светлом будущем.

Номер сообщения

Предназначен для последовательного нумерации отправляемых сообщений. Используется для избежания коллизий в случае перезапроса отправляемых сообщений.

ID – 1 байт, идентификатор сообщения

Предназначен для указания типа сообщения (команд, запросов, ответов) с указанием параметров сообщения.

CRC - Размер поля контрольной суммы сообщения: 8 бит.

XOR всех байт сообщения (за исключением старт - байта и стоп-байта)

Перечень идентификаторов сообщений с соответствующими описаниями приведен в таблице:

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор сообщения, (название) | Описание сообщение | Размер данных | Комментарий |
| 00  (NOP) | Пустая команда | 1 байт | Данные произвольные |
| 01  (NOP\_BACK) | Ответ на пустую команду | 1 байт | Возвращает байты данных команды NOP |
| 02  () | Установить режим | 4 байт | 1 байт  Два бита:  01 – голос  10 – данные  11 – данные RTK  3й бит - резерв  4й бит – мощность малая/полная  5й бит – спать/не спать  06-07 резерв  2 байт  3 бита громкость  3 бита чувствительность микрофона  2 бита - резерв  3,4 байт  Значение частоты, начиная с 410 МГц, заканчивая 480 МГц кратно 25 кГц. |
| 03  () | Ответ на команду Установить режим | 1 байт | Возвращает байт данных |
| 04 () | Передать фрейм | 128 байт | Передать данные на трансивер |
| 05 () | Ответ на команду  Передать фрейм | 1 байт |  |
| 06 () | Принять фрейм | 1 байт | Принять данные с трансивера |
| 07 () | Ответ на команду принять фрейм | 128 байт |  |
| 08 () | Запрос текущих параметров | 1 байт | 0й бит – однократный/асинхронный ответ на запрос. В случае выбора асинхронного запроса ведомое устройство самостоятельно инициирует ответ с требуемыми параметрами в случае их изменения (например уровень RSSI). Для выключения асинхронного режима ведущее устройство формирует повторный запрос точно такого же формата (?).  1-3 бит – код (порядковый номер) байта для возврата  000 все байты – см. сообщение 09  001 первый байт - см. сообщение 09  010 второй байт см сообщение 09  011 первый и второй байты – см. сообщение 09  100 третий и четвертый байт (текущая частота) – см. сообщение 09  101 пятый байт (значение RSSI) – см сообщение 09  110 шестой байт (2 бита (статус PTT, ошибки) – см сообщение 09  111 – пятый и шестой байт – см сообщение 09 |
| 09 () | Ответ на запрос текущих параметров | 2-7 байт (в зависимости от запроса) | 1й байт – повторяет байт запроса 08  2й байт (такой же как в сообщении «установить режим»):  Два бита:  01 – голос  10 – данные  11 – данные RTK  3й бит - резерв  4й бит – мощность малая/полная  5й бит – спать/не спать  06-07 резерв  3й байт (такой же как в сообщении «установить режим»)  3 бита громкость  3 бита чувствительность микрофона  2 бита – резерв  4,5 байт  Значение частоты  6й байт – значение RSSI  7й байт  2 бита (статус PTT)  01 прием  10 дежурный прием  11 передача  6 бит – код ошибки  0 – ошибки не зафиксировано  Остальные биты зарезервированы |
| 0A () | Запрос версии ПО | 0 байт |  |
| 0B () | Ответ на запрос версии ПО | 1 байт | Значение текущей версии ПО |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Перечень команд GUI (STM) – CC1120

В данном разделе приведены логические команды и функции, необходимые для работы с трансивером СС1120 при управлении с GUI. Общая логика выполнения команды должны быть следующей:

- при выборе

Такие же команды и функции должны выполняться с STM при управлении трансивером CC1120 без GUI.

Перечень команд приведен в таблице

|  |  |
| --- | --- |
| Наименование команды | Описание |
| Transiever\_Vreifity() |  |
| TX\_FIFO\_Control() |  |
| RX\_FIFO\_Control() |  |
| Manual\_calibration() |  |
| Reset() |  |
| Tx\_FIFO\_flush() |  |
| Rx\_FIFO\_flush() |  |
| TX() |  |
| Status() |  |
| IDLE() |  |
| MARC() |  |
| SFSTXON() |  |
| LQI\_VAL() |  |
| RX() |  |
| Rx\_FIFO\_read() |  |
| Записать конфигурацию  WRITE\_CONF |  |
| Прочитать конфигурацию  READ\_CONF |  |
| Записать частоту  WRITE\_FREQ |  |
| Прочитать частоту  READ\_FREQ |  |
| Записать регистр  WRITE\_REG |  |
| Прочитать регистр  READ\_REG |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Настройки микросхем радиоблока при обмене: