

УДК 621.391
<http://doi.org/10.18799/29495407/2026/1/116>
Шифр специальности ВАК: 2.2.9
Научная статья



Простой обмен данными между смартфоном и микроконтроллером STM32F103 с помощью Bluetooth-модуля HC-06

П.Д. Шендрик, И.В. Трубин✉, В.Г. Трубин

Новосибирский Государственный Технический Университет, Российская Федерация, г. Новосибирск

✉i.trubin@corp.nstu.ru

Аннотация. Рассматривается практический пример организации беспроводного обмена данными между мобильным устройством и микроконтроллером STM32F103. Основное внимание уделяется настройке популярного Bluetooth-модуля HC-06 с использованием персонального компьютера и его интеграции с микроконтроллером. Приводится подробная схема подключения модуля к STM32F103. В программной части представлена реализация простой задачи «Echo+1» на языке ANSI C для микроконтроллера с прямым управлением периферией через управляющие регистры без использования стандартных библиотек и операционной системы. Описанный подход демонстрирует базовые принципы взаимодействия встраиваемых систем с внешними устройствами по Bluetooth и может служить основой для более сложных проектов в области интернета вещей и удаленного управления.

Ключевые слова: STM32F103, Bluetooth, HC-06, USART, микроконтроллер, обмен данными, AT-команды, беспроводная связь, низкоуровневое программирование

Для цитирования: Шендрик П.Д., Трубин И.В., Трубин В.Г. Простой обмен данными между смартфоном и микроконтроллером STM32F103 с помощью Bluetooth-модуля HC-06. *Известия Томского политехнического университета. Промышленная кибернетика*, 2026, Т. 4, № 1, С. 15–22. <http://doi.org/10.18799/29495407/2026/1/116>

Конфликт интересов: отсутствует.

UDC 621.391
<http://doi.org/10.18799/29495407/2026/1/116>
Scientific paper

Simple data exchange between a smartphone and STM32F103 microcontroller using the HC-06 Bluetooth module

P.D. Shendrik, I.V. Trubin✉, V.G. Trubin

Novosibirsk State Technical University, Novosibirsk, Russian Federation

✉i.trubin@corp.nstu.ru

Abstract. The article discusses a practical example of organizing wireless data exchange between a mobile device and STM32F103 microcontroller. The main focus is on configuring the popular HC-06 Bluetooth module using a personal computer and integrating it with the microcontroller. The article provides a detailed diagram of how to connect the module to the STM32F103. The software section demonstrates the implementation of a simple "Echo+1" task in ANSI C with direct control of the peripherals through control registers, without using standard libraries or an operating system. The described approach demonstrates the basic principles of interaction between embedded systems and external devices via Bluetooth and can serve as a foundation for more complex projects in the field of the Internet of Things and remote control.

Keywords: STM32F103, Bluetooth, HC-06, USART, microcontroller, data exchange, AT commands, wireless communication, low-level programming

For citation: Shendrik P.D., Trubin I.V., Trubin V.G. Simple data exchange between a smartphone and STM32F103 microcontroller using the HC-06 Bluetooth module. *Bulletin of the Tomsk Polytechnic University. Industrial Cybernetics*, 2026, vol. 4, no. 1, pp. 15–22. <http://doi.org/10.18799/29495407/2026/1/116>

Conflict of interest: none.

Введение

Современные тенденции развития электроники и встраиваемых систем неразрывно связаны с расширением функциональных возможностей устройств за счет интеграции беспроводных интерфейсов связи. Одной из востребованных задач является организация удаленного доступа и управления микроконтроллерными устройствами с помощью мобильных приложений. Это открывает широкие перспективы для создания систем «умного дома», персональных носимых устройств и промышленной телеметрии. Перенос интерфейса настройки и управления на смартфон позволяет отказаться от использования громоздких пультов и специализированных дисплеев, делая взаимодействие с устройством более интуитивным и удобным для конечного пользователя.

Для решения подобных задач отлично подходит связка недорогого и распространенного микроконтроллера STM32F103 (далее «МК») с Bluetooth-модулем HC-06 (далее «HC-06») [1]. Несмотря на

свою простоту, HC-06 является полноценным UART-мостом, позволяющим «прозрачно» передавать данные между МК и любым устройством, поддерживающим Bluetooth Classic (SPP-профиль). В данной статье рассматривается полный цикл подготовки такого беспроводного канала: от начальной конфигурации модуля HC-06 с помощью компьютера до написания программы для МК и ее тестирования со смартфона. В качестве демонстрационного примера реализована программа, выполняющая функцию приёма и передачи символа. Для программирования используется минимальный набор средств: язык ANSI C и прямая работа с регистрами микроконтроллера. Такой подход позволяет обойтись без громоздких библиотек и операционных систем, а вся необходимая информация содержится в официальной документации [2] и справочном руководстве [3] для МК, что обеспечивает полный контроль над аппаратурой и глубокое понимание происходящих процессов.

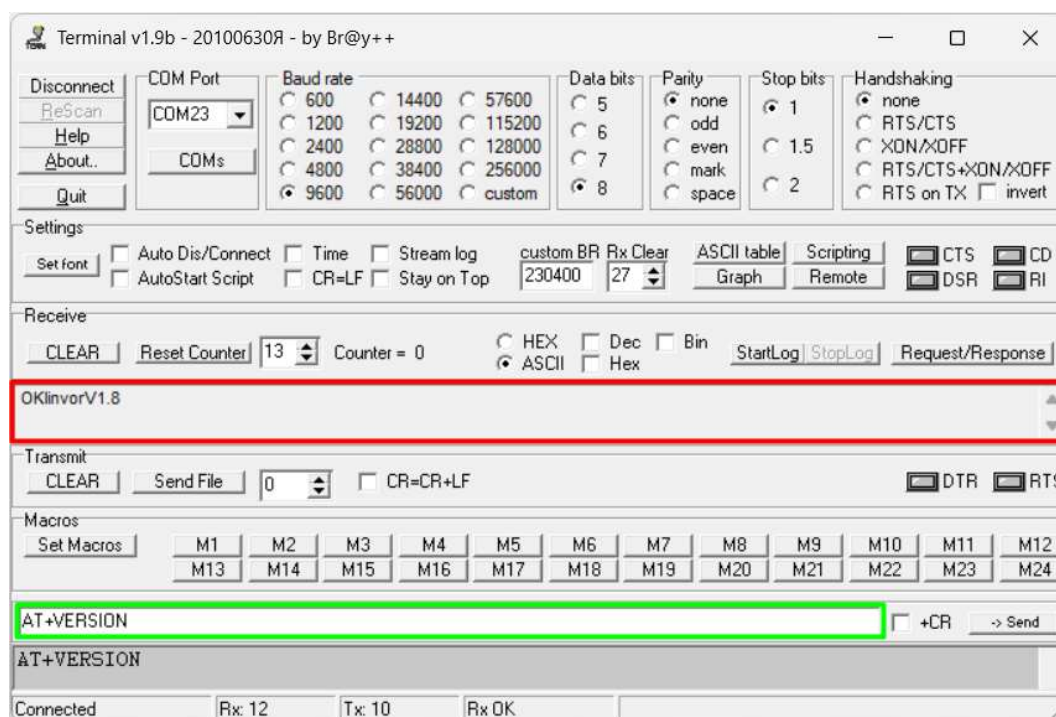


Рис. 1. Обмен данными в приложении Terminal (окна приёма и передачи данных)

Fig. 1. Data exchange in the Terminal application (windows for receiving and transmitting data)

Настройка Bluetooth-модуля HC-06

Перед использованием Bluetooth-модуля HC-06 настроим его с помощью компьютера. Для этого можно воспользоваться TTL-преобразователем USB-TO-UART на основе микросхемы CH340 (далее «CH340») [4] и программой для работы с виртуальными последовательными портами (так называемыми COM-портами) Terminal v1.9b (рис. 1).

Для общения с модулем откройте Terminal v1.9b и посмотрите список доступных COM-портов («стрелка вниз», рис. 2). В системе уже могут быть подключенные устройства, как на рис. 2, а также список может быть пустым, если доступные порты не обнаружены. Соберите схему (рис. 3, 4), подключите её к компьютеру и обновите список доступных COM-портов нажатием на «ReScan». Выберите появившейся новый порт и подключитесь к нему, нажав на «Connect».

После подключения можно приступить к конфигурированию модуля. Управление параметрами HC-06 выполняется с помощью специальных текстовых команд – AT-команд (от англ. Attention). Несмотря на кажущуюся простоту, этих команд достаточно для выполнения всех необходимых настроек: изменения имени устройства, установки ПИН-кода, выбора скорости передачи данных и запроса версии программного обеспечения Bluetooth-модуля. Полный перечень поддерживаемых модулем HC-06 AT-команд и их синтаксис представлены в таблице.

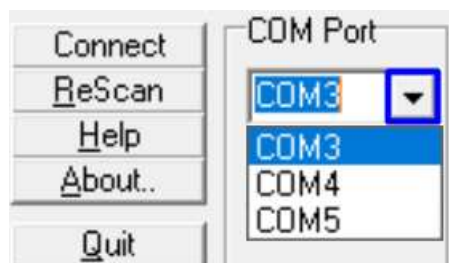


Рис. 2. Выбор COM-порта для связи
 Fig. 2. Selecting a COM port for connection

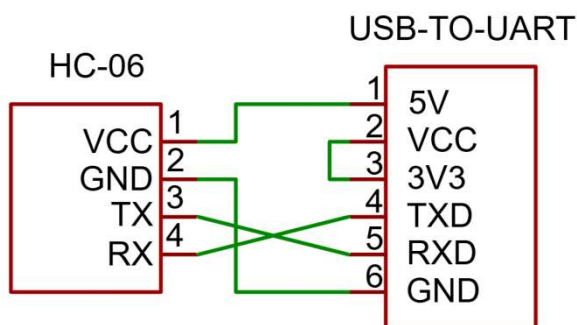


Рис. 3. Схема подключения HC-06 к компьютеру
 Fig. 3. Connecting the HC-06 to a computer



Рис. 4. Фото схемы подключения HC-06 к компьютеру
 Fig. 4. Photo of the HC-06 connection diagram to a computer

Таблица. AT-команды для Bluetooth-модуля HC-06

Table. AT commands for the HC-06 Bluetooth module

Назначение Purpose	AT-команда AT-command	Параметры Parameters	Ответ от модуля Response from the module
Проверка связи Checking the connection	AT	-	OK
Изменение скорости Speed change	AT+BAUDx	BAUD1 - 1200	OK1200
		BAUD2 - 2400	OK2400
		BAUD3 - 4800	OK4800
		BAUD4 - 9600	OK9600
		BAUD5 - 19200	OK19200
		BAUD6 - 38400	OK38400
		BAUD7 - 57600	OK57600
		BAUD8 - 115200	OK115200
		BAUD9 - 230400	OK230400
Изменение имени Name change	AT+NAMEname	name - новое имя, до 20 символов name - new name, up to 20 characters	OKsetname
		xxxx - новый ПИН-код xxxx - new PIN code	OKsetPIN
Изменение пароля Password change	AT+PINxxxx	xxxx - новый ПИН-код xxxx - new PIN code	OKsetPIN
Без проверки на чётность Without checking for parity	AT+PN	Для версий выше V1.5	OK None
Проверка на нечётность Checking for odd numbers	AT+PO	For versions higher than V1.5	OK Odd
Проверка на чётность Checking for parity	AT+PE		OK Even
Версия модуля Module version	AT+VERSION	-	OKlinvorV1.8

Прежде чем приступить к изменению параметров, необходимо знать настройки модуля HC-06. По умолчанию модуль имеет следующие установки: скорость передачи – 9600 бит/с (бод), 8 бит данных, 1 стоповый бит без контроля чётности; имя устройства – «linvor»; ПИН-код для сопряжения с другим устройством – «1234». В зависимости от задачи эти параметры могут быть изменены. Например, для

совместной работы с микроконтроллером *STM32F103* удобно установить скорость 9600 бит/с и задать собственное имя и ПИН-код.

Выставим необходимые настройки передачи данных (рис. 5) и проверим связь, отправив команду «AT», нажатием на «→ Send» (рис. 6). В ответ модуль должен отправить «OK» (рис. 7), это означает, что соединение установлено.

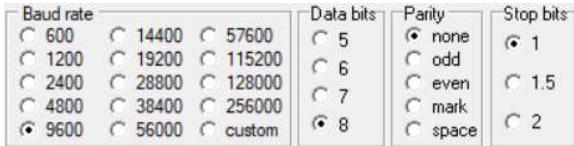


Рис. 5. Настройка параметров передачи данных
 Fig. 5. Configuring data transmission parameters



Рис. 6. Окно передачи данных
 Fig. 6. Data transfer window

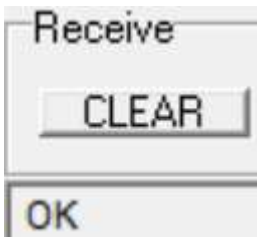


Рис. 7. Окно приёма данных
 Fig. 7. Data reception window

Если ответ от модуля *HC-06* не был получен, а схема собрана правильно, скорее всего, модуль настроен иначе, поэтому стоит прибегнуть к перебору различных скоростей из таблицы. Если всё ещё не удалось найти нужную скорость передачи данных, необходимо попробовать перебрать их ещё раз, предварительно включив добавление специальных управляющих символов в конец сообщения: «+CR (Carriage Return)» или «CR=CR+LF (Carriage Return+Line Feed)».

После получения ответа на скорости передачи данных, которая отличается от 9600 бит/с, следует установить скорость обмена 9600 бит/с, отправив команду «AT+BAUD4».

При установленном соединении можно проверить версию устройства, отправив «AT+VERSION», и перейти к настройке имени и ПИН-кода. Для задания нового имени устройства воспользуемся командой «AT+NAMEHC-06», а для установки ПИН-кода – «AT+PIN1234».

Схема подключения

После настройки *Bluetooth*-модуля, перейдём к схеме подключения (рис. 8, 9).

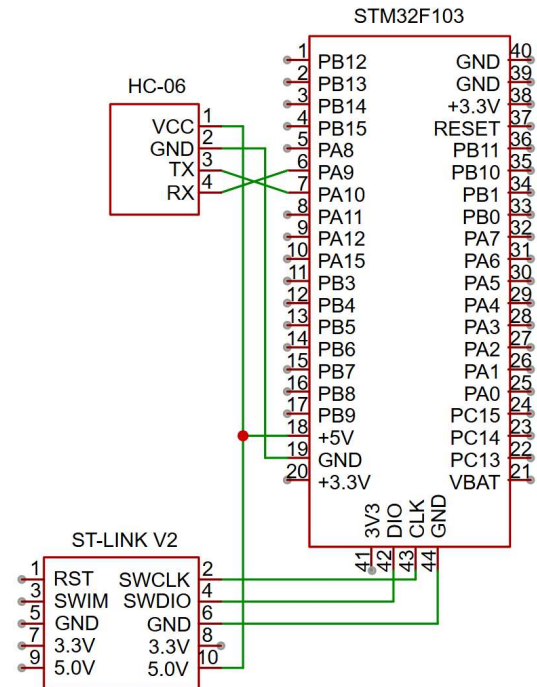


Рис. 8. Схема подключений *HC-06* к МК
 Fig. 8. *HC-06* connection diagram to MK

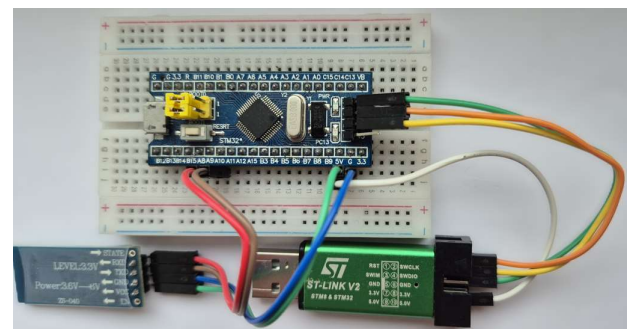


Рис. 9. Фото полностью собранной схемы
 Fig. 9. Photo of the fully assembled circuit

Программный код

Рассмотрим простой обмен данными между смартфоном и микроконтроллером *STM32F103* на примере решения типовой задачи «Echo+1». Данная программа должна выполнять следующие задачи:

1. ожидает приём символа (ASCII код) по интерфейсу USART1 [5, 6] от Bluetooth-модуля *HC-06*;
2. после получения символа увеличивает его ASCII код на 1 и отправляет его обратно в смартфон;
3. после отправления изменённого символа дополнительно отправляет «!» и символ переноса строки «\n».

Всю инициализацию МК выполним в отдельном файле «*main_init.c*».

Блок кода 1. Файл *main_init.c*

```
// === Инициализация микроконтроллера ===
static __INLINE void main_init() {
// Буфер предвыборки разрешён, 2 цикла за-
держки для FLASH, см. PM0075
FLASH->ACR = FLASH_ACR_PRFTBE |
FLASH_ACR_LATENCY_2;
// HCLK = SYSCLK
RCC->CFGR |= RCC_CFGR_HPRE_DIV1;
// PCLK1 = SYSCLK / 2 (APB1)
RCC->CFGR |= RCC_CFGR_PPRE1_DIV2;
// PCLK2 = SYSCLK (APB2)
RCC->CFGR |= RCC_CFGR_PPRE2_DIV1;
// Выбираем работу от PLL
RCC->CFGR |= RCC_CFGR_SW_PLL;
// SYSCLK = 72 MHz
RCC->CFGR |= RCC_CFGR_PLLMULL9;
// Тактирование PLL от кварца
RCC->CFGR |= RCC_CFGR_PLLSRC_HSE;
RCC->CR |= RCC_CR_HSEON; //Запускаем HSE
while (!(RCC->CR & RCC_CR_HSERDY)) {};
RCC->CR |= RCC_CR_PLLON; // Запускаем PLL
while ((RCC->CR & RCC_CR_PLLRDY) == 0) {};
// Ожидаем, когда PLL станет источником так-
тирования
while ((RCC->CFGR & RCC_CFGR_SWS) !=
RCC_CFGR_SWS_1) {};
// Отключаем внутренний RC-генератор
//RCC->CR &= ~RCC_CR_HSION;
// JTAG-DP запрещён and SW-DP разрешён
AFIO->MAPR = (AFIO_MAPR_SWJ_CFG_JTAGDISABLE);

// ===== DWT =====
// Разрешаем использовать счётчик
SCB_DEMCR |= 0x01000000;
// Обнуляем значение счётного регистра
DWT_CYCNT = 0;
// Запускаем счётчик
DWT_CONTROL |= 1;
}
```

Для приёма и передачи символов настроим по-следовательный интерфейс *USART1*.

Блок кода 2. Файл *main_init.c*

```
// ===== USART1 =====
// Настройки по умолчанию: 8 инф. бит,
// 1 стоповый бит, контроля чётности нет
// Включаем тактирование порта A
RCC->APB2ENR = RCC_APB2ENR_IOPAEN;
// Включаем тактирование USART1
RCC->APB2ENR |= RCC_APB2ENR_USART1EN;
// PA9 (TX1) AFIO Push-Pull, 10MHz
// PA10 (RX1) HiZ, 10MHz
GPIOA->CRH &= ~(GPIO_CRH_MODE9 |
GPIO_CRH_CNF9);
GPIOA->CRH &= ~(GPIO_CRH_MODE10 |
GPIO_CRH_CNF10);
GPIOA->CRH |= (GPIO_CRH_MODE9_0 |
GPIO_CRH_CNF9_1 | GPIO_CRH_CNF10_0);
// PCLK2 / Baud = 72000000 / 9600 бод
USART1->BRR = 7500;
// Включаем USART, передатчик и приемник
USART1->CR1 = USART_CR1_UE | USART_CR1_TE |
USART_CR1_RE;
}
```

Для удобства введём вспомогательный файл «*main_functions.c*» с функциями задержки, приёма и передачи данных.

Блок кода 3. Файл *main_functions.c*

```
// Константы для функций задержки
#define DWT_CYCNT *(volatile uint32_t *)0x0001004
#define DWT_CONTROL *(volatile uint32_t *)0x0001000
#define SCB_DEMCR *(volatile uint32_t *)0x000EDFC

/* Описание:
Функция задержки в мс от 1 мс до 59 с
(дальше переполнение)
Входные параметры:
uint32_t ms - время задержки в мс */
static __INLINE void delay_ms(uint32_t ms) {
static volatile uint32_t n;
n = ms * 72000;
DWT_CYCNT = 0;
while (DWT_CYCNT < n);
}

/* Описание:
Функция отправки символа по USART1
Входные параметры:
uint8_t Symbol - беззнаковое 8 битное число */
void Tx1(uint8_t Symbol) {
while ((USART1->SR & USART_SR_TXE) == 0) {};
USART1->DR = Symbol;
}

/* Описание:
Функция приёма символа по USART1
Выходные
параметры:
Возвращает байт формата uint8_t */
uint8_t Rx1() {
while ((USART1->SR & USART_SR_RXNE) == 0)
{};
return USART1->DR;
}
```

После всех подготовительных действий рассмотрим главный файл «*main.c*». Вначале выполняется подключение необходимых заголовочных файлов. Основная логика реализована в функции *main()*, которая сначала выполняет инициализацию МК с помощью функции из «*main_init.c*», затем объявляет переменную *msg* для приёма и передачи данных, после чего выполняется бесконечный цикл *while(1)*, где и происходит основная работа программы.

Блок кода 4. Файл *main.c*

```
// Описание регистров МК
#include "System/stm32f10x.h"
// Файл с функциями
#include "main_functions.c"
// Файл с инициализацией микроконтроллера
#include "main_init.c"

// ===== MAIN =====
int main(void) {
main_init(); // Инициализация МК
}
```

```
// Задержка для нормализации питания МК
delay_ms(100);
// Байт данных для общения
volatile uint8_t msg = 0;

// ===== ГЛАВНЫЙ ЦИКЛ =====
while (1) {
    // Принимаем данные (ASCII код)
    msg = Rx1();
    Tx1(msg + 1); // Передаем данные + 1
    Tx1('!'); // Передаем '!'
    Tx1('\n'); // Переносим строку
}
}
```

Проверка кода

Для программирования МК воспользуемся программатором *ST-LINK V2*. При его отсутствии можно воспользоваться технологией программирования микроконтроллеров с помощью встроенного загрузчика по *USART* [7].

После подключения собранной схемы к компьютеру и программирования МК светодиод на *HC-06* должен постоянно мигать (примерно 3,5 Гц) – модуль в режиме ожидания подключения.

Для проверки связи с модулем на смартфоне воспользуемся приложением *Bluetooth Monitor Serial* (рис. 10). Запустим приложение и нажмём кнопку «*Search*», чтобы найти доступные *Bluetooth*-устройства. Когда поиск завершится, в списке появится модуль *HC-06*. Выберем его и нажмём «*CONNECT*» для установки соединения.



Рис. 10. Поиск устройства во вкладке «*Connection*» приложения *Bluetooth Monitor Serial*

Fig. 10. Finding a device in the «*Connection*» tab of the *Bluetooth Monitor Serial* application

Если подключение происходит впервые, то приложение запросит ПИН-код (1234) для установки соединения (рис. 11), после чего сообщит об успешном подключении (рис. 12). После установки соединения светодиод должен перестать мигать и начать светиться постоянно.

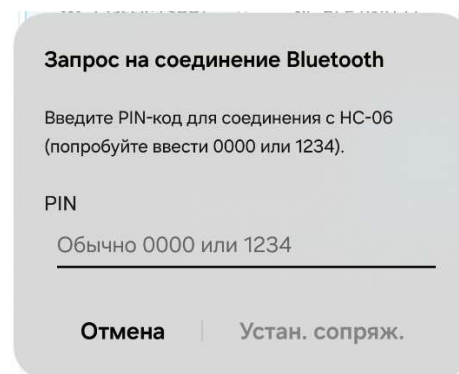


Рис. 11. Запрос на соединение Bluetooth
Fig. 11. Bluetooth connection request

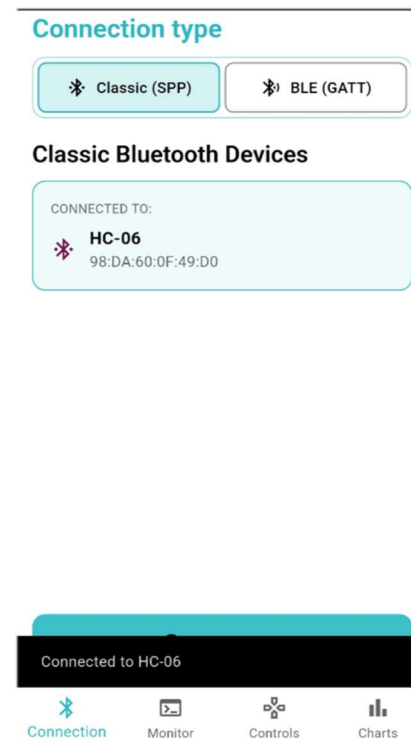


Рис. 12. Успешное подключение к *HC-06* в приложении *Bluetooth Monitor Serial*

Fig. 12. Successful connection to the *HC-06* in the *Bluetooth Monitor Serial* application

Для обмена данными переходим во вкладку «*Monitor*» (рис. 13) и отправляем символы. Отправленные символы отображаются после «>>», а полученные – после «<<».

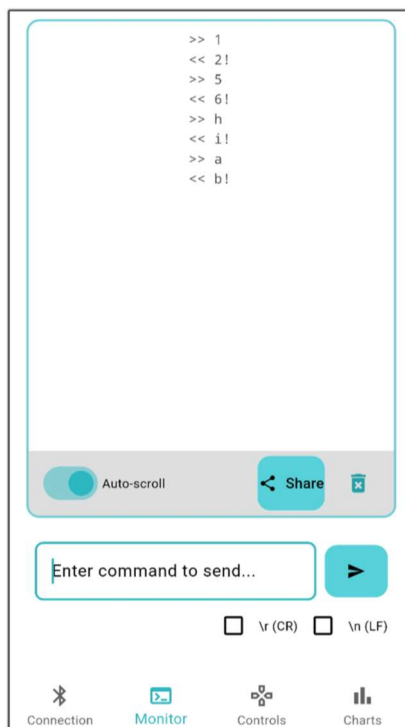


Рис. 13. Обмен данными во вкладке «Monitor» приложения Bluetooth Monitor Serial

Fig. 13. Data exchange in the «Monitor» tab of the Bluetooth Monitor Serial application

Заключение

Bluetooth-модуль *HC-06* – это простое и недорогое решение для добавления беспроводного интерфейса в проекты на базе микроконтроллеров. Он не требует сложной обвязки, легко подключается и настраивается, позволяя создавать устройства с удалённым управлением по радиоканалу без лишних усилий.

Представленный в статье пример является базовым, однако он закладывает фундамент для разработки более сложных и функциональных проектов, таких как удаленное управление нагрузками (реле, двигатели), сбор данных с датчиков и передача их на мобильное устройство. Используемый подход к программированию «на регистрах» позволяет разработчику тонко настраивать производительность системы и эффективно использовать ресурсы микроконтроллера.

Для Bluetooth-модулей семейства *HC* (*HC-04*, *HC-05*, *HC-06* и другие) существует множество приложений для операционной системы *Android*, что существенно упрощает тестирование и отладку разрабатываемых устройств. При этом наличие моделей с разными функциональными возможностями позволяет разработчику выбрать наиболее подходящий модуль под конкретную задачу.

СПИСОК ЛИТЕРАТУРЫ

1. Ескин А.В., Жмудь В.А., Трубин В.Г. Беспроводной удлинитель последовательного порта на базе радиоканала Bluetooth. *Автоматика и программная инженерия*, 2013, № 2 (4), С. 42–47. EDN: VIUNQB
2. Документация на микроконтроллер *STM32F103*. URL: <http://inverter48.ru/datasheet/mcu/STM32F103C8T6.pdf?ysclid=mlxph17uay118692408> (дата обращения: 22.01.2026).
3. *Reference manual STM32F101xx, STM32F102xx, STM32F103xx, STM32F105xx and STM32F107xx advanced Arm®-based 32-bit MCUs*. URL: https://robocraft.ru/files/datasheet/STM32-RM0008_Reference_manual.pdf?ysclid=mlxpgah3jc324540634 (дата обращения: 22.01.2026).
4. *USB to Serial Port Chip CH340*. URL: <https://www.wch-ic.com/download/file?id=79> (дата обращения: 16.01.2026).
5. Близнюк А.Е., Трубин В.Г., Саблина Г.В. Представление и отображение информации при работе с микроконтроллерами *STM32*. *Автоматика и программная инженерия*, 2022, № 3 (41), С. 60–78. EDN: GGMTQO
6. Жмудь В.А., Трубин И.В., Трубин М.В. Обмен данными между компьютером и микроконтроллером *STM32F100* по последовательному интерфейсу связи *RS-232*. *Автоматика и программная инженерия*, 2015, № 1 (11), С. 45–51. EDN: VJOEZZ
7. Программирование микроконтроллеров *STM32F10x* с помощью встроенного загрузчика по *USART*. А.Е. Близнюк, В.А. Жмудь, М.В. Трубин, В.Г. Трубин. *Программная инженерия*, 2022, Т. 13, № 3, С. 132–141. DOI: 10.17587/prin.13.132-141 EDN: OJTMKC

Информация об авторах

Павел Дмитриевич Шендрик, студент, кафедра автоматки, Новосибирский государственный технический университет, Российская Федерация, 630073, г. Новосибирск, пр. Карла Маркса, 20; pasha.shendrick@yandex.ru

Игорь Витальевич Трубин, старший преподаватель, кафедра защиты информации, Новосибирский государственный технический университет, Российская Федерация, 630073, г. Новосибирск, пр. Карла Маркса, 20; i.trubin@corp.nstu.ru; <https://orcid.org/0009-0001-1466-6001>

Виталий Геннадьевич Трубин, старший преподаватель, кафедра автоматки, Новосибирский государственный технический университет, Российская Федерация, 630073, г. Новосибирск, пр. Карла Маркса, 20; trubin@corp.nstu.ru

Поступила: 07.02.2026

Принята: 13.03.2026

Опубликована: 30.03.2026

REFERENCES

1. Eskin A.V., Zhmud V.A., Trubin V.G. Wireless serial port extender based on Bluetooth radio channel. *Automation and Software Engineering*, 2013, no. 2 (4), pp. 42–47. (In Russ.) EDN: VIUNQB
2. *Documentation for the STM32F103 microcontroller*. (In Russ.) Available at: <http://inverter48.ru/datasheet/mcu/STM32F103C8T6.pdf?ysclid=mlxph17yay118692408> (accessed: 22 January 2026).
3. *Reference manual STM32F101xx, STM32F102xx, STM32F103xx, STM32F105xx and STM32F107xx advanced Arm®-based 32-bit MCUs*. Available at: https://robocraft.ru/files/datasheet/STM32-RM0008_Reference_manual.pdf?ysclid=mlxpgah3jc324540634 (accessed: 22 January 2026).
4. *USB to Serial Port Chip CH340*. Available at: <https://www.wch-ic.com/download/file?id=79> (accessed: 16 January 2026).
5. Bliznyuk A.E., Trubin V.G., Sablina G.V. Presentation and display of information when working with stm32 microcontrollers. *Automation and Software Engineering*, 2022, no. 3 (41), pp. 60–78. (In Russ.) EDN: GGMTQO
6. Zhmud V.A., Trubin I.V., Trubin M.V. Data exchange between a computer and an STM32F100 microcontroller via an RS-232 serial communication interface. *Automation and software Engineering*, 2015, no. 1 (11), pp. 45–51. (In Russ.) EDN: VJOEZL
7. Bliznyuk A.E., Zhmud V.A., Trubin M.V., Trubin V.G. Programming of STM32F10X microcontrollers using the built-in usart bootloader. *Programmnyaya ingeneriya*, 2022, vol. 13, no. 3, pp. 132–141. DOI: 10.17587/prin.13.132-141 (In Russ.) EDN: OJTMKC

Information about the authors

Pavel D. Shendrik, Student, Novosibirsk State Technical University, 20, Karl Marks avenue, Novosibirsk, 630073, Russian Federation; pasha.shendrick@yandex.ru

Igor V. Trubin, Senior Lecturer, Novosibirsk State Technical University, 20, Karl Marks avenue, Novosibirsk, 630073, Russian Federation; i.trubin@corp.nstu.ru; <https://orcid.org/0009-0001-1466-6001>

Vitaly G. Trubin, Senior Lecturer, Novosibirsk State Technical University, 20, Karl Marks avenue, Novosibirsk, 630073, Russian Federation; trubin@corp.nstu.ru

Received: 07.02.2026

Revised: 13.03.2026

Accepted: 30.03.2026