

Встроенное программирование BLE-модуля BT4502

от HopeRF

В статье представлен новый миниатюрный микропотребляющий BLE SoC CMT4502 от CMOSTEK и выполненный на его основе модуль HM-BT4502B от HopeRF. Подробно описано подключение выводов модуля в режиме обнаружения, передачи данных и сна. Приведен пример настройки терминальной Android-программы для обмена данными с UART модуля. Дана методика создания собственного программного обеспечения в среде Keil uVision5 и загрузки его в модуль. Статья может быть полезной разработчикам устройств для «умного дома» и построения на их основе IoT-сети.

Сергей Гаевский
info@nuvoton.com.ua

Разработка современной электроники немислима без использования беспроводных технологий малого радиуса действия, обеспечивающих взаимосвязь различных устройств на небольших расстояниях. Разнообразные сетевые архитектуры («точка-точка», «звезда», Mesh, ширококвещательная передача и т. п.) могут строиться на базе различных радиотехнологий нижнего уровня, реализующих физический канал передачи данных с необходимыми характеристиками. Важнейшими требованиями к подобным системам обычно становятся энергосбережение и совместимость (стандартизация различных сетевых уровней). Одно из лучших решений, удовлетворяющих данным критериям, — Bluetooth Low Energy (BLE).

Эта технология прочно вошла в мир «Интернета вещей» и, несмотря на свою молодость, успела завоевать огромную популярность среди IoT-разработчиков. В сравнении с технологией классического Bluetooth технология BLE спроектирована с учетом максимального энергосбережения: BLE-устройства могут годами работать от миниатюрной батареи, постоянно оставаясь на связи. Для обеспечения такой экономии была полностью изменена концепция протокола: периферийные устройства являются серверными и подавляющее большинство времени находятся в режиме сна. Периодически, в течение очень короткого времени, устройства передают специальный ширококвещательный пакет (Advertising) на так называемых «рекламных» выделенных радиоканалах. Для этого используются канал 0 (2402 МГц), канал 12 (2424 МГц) и канал 39 (2480 МГц), расположенные между основными каналами Wi-Fi, обычно загруженными пользователями, что позволяет даже при малом уровне мощности быть «услышанными» другими устройствами. Значительный разнос частот между «рекламными» радиоканалами уменьшает эффект

интерференции в помещениях, приводящий к появлению «мертвых» зон приема. Таким образом, на «рекламных» каналах BLE-устройства сообщают о себе другим BLE-устройствам, находящимся в зоне радиовидимости.

В отличие от «классического» Bluetooth устройства BLE предоставляют так называемые сервисы, объединяемые в профили. Обычно IoT-разработчики используют профиль, объединяющий сервисы UART (последовательного порта, аналогично SPP в «классическом» Bluetooth). Для обмена данными с конкретным BLE-чипсетом приложение должно получить от него идентификаторы характеристик (UUID) необходимого сервиса с помощью встроенного протокола GATT. Как вариант, можно получить нужные UUID из документации на чипсет и на этапе разработки добавить их в список UUID, поддерживаемых приложением.

Сегодня на рынке доступно несколько десятков BLE-чипсетов, отличающихся по ключевым характеристикам (размеру корпуса, энергопотреблению, излучаемой мощности) и по возможностям встроенного программного обеспечения нижнего уровня, реализующего поддерживаемые сервисы и определяющего их UUID. Также большое значение для разработчика имеет возможность устанавливать встроенное программное обеспечение верхнего уровня, обеспечивающее пользовательскую логику работы чипсета в составе модуля или готового устройства, включающую сетевые протоколы и интерфейс. Обычно производитель предоставляет среду разработки (SDK) и интерфейс нижнего уровня (API), с помощью которых пользователь может создавать программное обеспечение под конкретную решаемую им задачу. Скомпилированный бинарный код записывается во Flash-память чипсета с помощью специальной загрузочной утилиты и выполняется в виде отдельной задачи, управляя работой чипсета путем

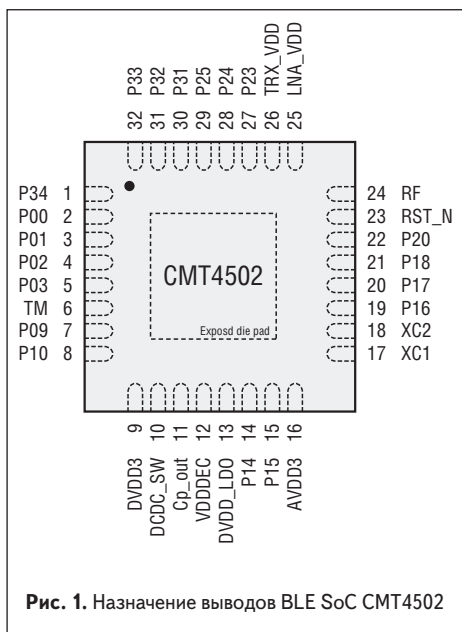


Рис. 1. Назначение выводов BLE SoC CMT4502

взаимодействия с «ядром», непосредственно работающим с аппаратными ресурсами кристалла.

Компания SMOSTEK — разработчик специализированных радиочастотных SoC, использующихся в продукции компании HoreRF, представила BLE-чипсет CMT4502 [1] — усовершенствованный аналог более раннего чипсета PNY6212 с аппаратной поддержкой стандарта BLE 5.0 (рис. 1). Новый чипсет обеспечивает чувствительность -97 дБм на скорости 1 Мбит/с и -103 дБм на скорости 125 кбит/с.

Широкий диапазон допустимых питающих напряжений (1.8–3,6 В) позволяет использовать данный чипсет в устройствах с различными элементами питания. Максимальная выходная мощность составляет +10 дБм, что эквивалентно 10 мВт. При этом ток потребления в режиме постоянного приема составляет 6,7 мА, а в режиме передачи — 7 мА. Максимальная скорость передачи полезных данных может достигать 1,6 Мбит/с с учетом задержки подключения. Предусмотрена аппаратная поддержка функций SIG-Mesh, обеспечивающая эффективную работу в составе Mesh-сети с сохранением низкого энергопотребления.

Для удобства инсталляции чипсета в пользовательское устройство радиочастотная часть имеет внутренний переключатель и обеспечивает работу с 50-омной антенной, подсоединенной непосредственно к единственному выводу без дополнительных цепей согласования импеданса.

Чипсет выполнен в миниатюрном корпусе QFN32 и сохраняет работоспособность в промышленном диапазоне температур -40...+125 °С.

Реализация BLE-стека и пользовательских приложений обеспечивается интегрированным в чипсет Cortex M0 микроконтроллером, содержащим 512 кбит Flash-памяти, 128 кбит предварительно запрограммированного ПЗУ для системных функций и 138 кбит ОЗУ с возможностью сохранения данных в режиме сна. Периферия представлена восемью каналами 12-битного АЦП, шестью каналами ШИМ, четырьмя каналами I²S, двумя каналами I²C и SPI, интерфейсами UART и JTAG (рис. 2).

Подробное техническое описание встроенного микроконтроллера доступно пользователю [2], но радиointерфейс и его регистры являются закрытыми.

Компания HoreRF представила на рынок BLE-модуль HM-BT4502B [3], базирующийся на вышеописанном чипсете и включающий необходимую обвязку: цепи питания, кварцевый резонатор с требуемой точностью установки частоты и элементы защиты линий ввода/вывода (рис. 3). Размеры модуля 15×11 мм при толщине 2 мм. Встроенная антенна расположена на верхней стороне платы, выво-

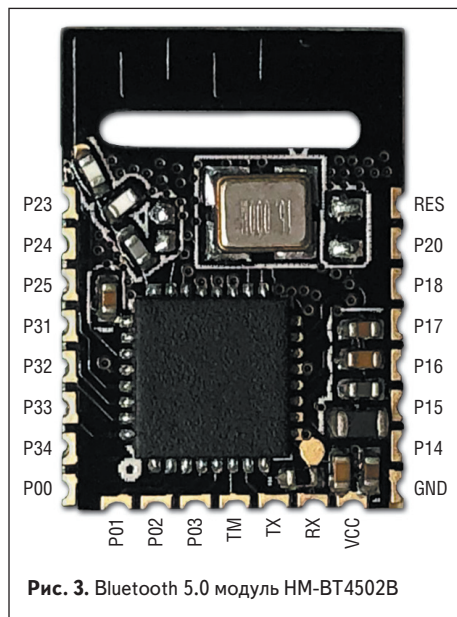


Рис. 3. Bluetooth 5.0 модуль HM-BT4502B

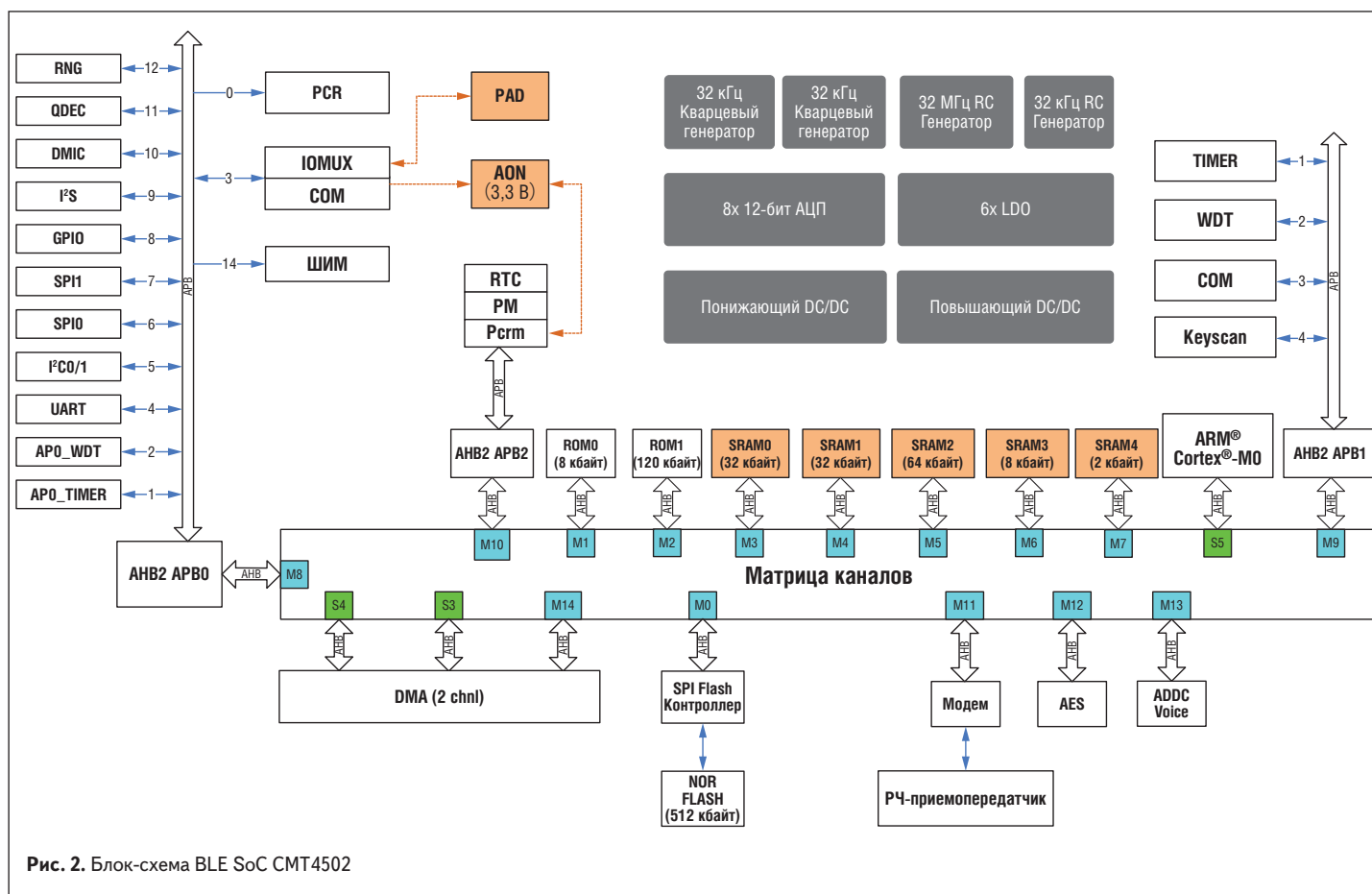


Рис. 2. Блок-схема BLE SoC CMT4502

ды — на остальных трех сторонах. Монтаж модуля осуществляется на материнскую плату пользовательского устройства.

Характеристики модуля определяются используемым чипсетом и программным обеспечением, реализующим верхний уровень протокола управления и обмена данными между модулем и пользовательским устройством. При напряжении питания 3 В ток в режиме низкого энергопотребления составляет 4 мкА, а максимальный ток в короткие периоды приема и передачи пакетов не превышает 8 мА. Подключение внешнего микроконтроллера реализуется через двухпроводный дуплексный интерфейс UART, включающий линии RX и TX и по умолчанию работающий на скорости 115 200 бод. Дополнительно для контроля режимов энергосбережения предусмотрены три аппаратные линии: PND, WKU и INT. Программно модуль управляется с помощью специальных команд, начинающихся с префикса TTM: (аналог AT+ в GSM-модулях) и содержащих последующую строку команды, дополненную символами возврата каретки <CR=0x0A>, перевода строки <LF=0x0D> и обязательным завершающим нулем <0x00>. Набор команд предоставляет широкие возможности по настройке модуля перед началом его работы. Можно изменить BLE-имя, MAC-адрес, установить интервал подключения 20–2000 мс (по умолчанию используется значение 30 мс), интервал отправки широковещательного оповещения (200 мс — 5 с) и назначить его содержимое, установить

требуемую выходную мощность передатчика в интервале –20...+10 дБм и т. п. Из списка следует выделить команду установки кода безопасности TTM:PID-XXXX<CR><LF><0>, позволяющую изменить PIN-код, требуемый при первичном подключении внешнего BLE-устройства к модулю. По умолчанию этот код равен 0000. В таком случае внешнее устройство имеет возможность подключаться к модулю автоматически без аутентификации. После смены пользовательского PIN-кода внешнее устройство должно запросить новый PIN-код у оператора или выбрать из списка, предоставляемого производителем устройства, включающего BLE-модуль. Подробно команды управления модулем приведены в техническом описании [4].

Большинство информационных выводов чипсета подключено к контактным площадкам модуля, расположенным с трех сторон мини-платы с шагом 1,25 мм (рис. 3).

Некоторые выводы имеют специальные альтернативные функции: P15 — АЦП0, P20 — АЦП1, P31 — ШИМ2, P33 — ШИМ3 и т. д. К выводам P16 и P17 может быть подключен кварцевый резонатор на частоту 32 768 Гц для повышения точности отсчета временных интервалов в режиме сна. Особую роль играют следующие выводы: P0 — вход сигнала PowerDown (PDN), P14 — вход сигнала WakeUp (WKU) и P18 — выход сигнала INT.

Входы PDN и WKU имеют внутреннюю подтяжку к напряжению питания (VCC) и для активации указанных сигналов должны быть установлены в нулевой уровень внешним микроконтроллером. Для корректного старта модуля в момент подачи питания вывод PDN должен быть заземлен, а вывод WKU — отключен. Через 100 мс после подачи питания вывод WKU необходимо также заземлить. В таком состоянии модуль отвечает на команды управления и доступен для обнаружения другими BLE-устройствами.

Для обмена данными с модулем удобно использовать Android-приложение Serial Bluetooth Terminal [6]. После обнаружения Bluetooth-устройства из меню настроек телефона и удачного подключения BLE-имя модуля (по умолчанию CMT4501-XXXXXXXX, где XXXXXXXX — четыре последних байта MAC-адреса) появляется на вкладке приложения Bluetooth LE.

Длинным нажатием на появившемся имени модуля необходимо вызвать меню и выбрать EDIT для настройки используемого сервиса. Чипсет BT4502 является сравнительно новым на рынке, поэтому, к сожалению, далеко не все приложения (в том числе и Serial Bluetooth Terminal в его текущей редакции) имеют предустановленные настройки для реализации обмена данными через UART. Вследствие этого коды UUID необходимо задать согласно даташиту на чипсет, выбрав из списка, формируемого GATT-запросом. Считав список UUID нажатием на одноименную кнопку, выбираем сервис со значением 0000ffe0 в первом поле, а затем подключаем UUID для чтения и записи данных нажатием соответствующих кнопок. Результат настройки показан на рис. 4.

После выполненных настроек можно подключаться к устройству из основного меню терминала и обмениваться данными с UART в обоих направлениях. Допускается применение произвольных бинарных данных, передача которых производится пакетами. Максимальная длина данных в одном пакете составляет 240 байт. При подключении внешнего устройства модуль выводит в UART сообщение TTM:CONNECT<CR><LF><0>, а при отключении — сообщение TTM:DISCONNECT<CR><LF><0>. Во время активного подключения можно снизить энергопотребление модуля, отсоединив вывод PDN. При этом данные, отправленные внешним устройством на модуль, будут успешно приняты и выведены в UART, но данные, отправленные в UART, не будут передаваться на внешнее устройство. При обратном подключении вывода PDN на «землю» двухсторонний обмен данными возобновляется. При отключении вывода WKU от «земли» модуль переходит в режим сна, соединение с внешним устройством разрывается. Также необходимо отметить, что при отключении от «земли» вывода PDN модуль становится недоступным для обнаружения (не передает пакеты представления). Выход модуля INT используется для пробуждения микроконтроллера из режима сна. При получении пакета данных от удаленного устройства модуль генерирует смену уровня от высокого к низкому, а затем через настраиваемое время задержки (по умолчанию 500 мкс) отправляет принятые данные в UART. Таким образом, и внешний микроконтроллер, и BLE-модуль могут при необходимости «будить» друг друга, большую часть времени находясь в режиме низкого энергопотребления.

Для расширения возможностей использования модуля компания NoreRF предоставляет свою среду разработки (SDK) [6], разрешающую пользователю создавать собственное программное обеспечение верхнего уровня и устанавливать его в модуль, заменяя описанный выше заводской функционал управления TTM-командами и работы с UART. Архитектура базируется на операционной системе реального времени OSAL. Большинство компонентов представлено в виде исходного кода, за исключением библиотеки работы с радиотрансивером, поставляемой в виде предварительно скомпилированных бинарных lib-файлов.

Среда разработки не требует дополнительной индивидуальной настройки и использует Keil uVision V5 для компиляции проекта. Последняя версия SDK хорошо документирована и включает широкий набор примеров, сгруппированных по функционалу. Каждый пример является самодостаточным Keil-проектом, готовым к компиляции.

Список примеров, имеющихся в SDK, предусматривает реализацию центрального устройства с возможностью подключения к другим модулям, различные варианты реализации сетей BLE-Mesh, примеры работы с периферией встроеного в чипсет микроконтроллера, а также множество реализаций протоколов верхнего уровня: различные

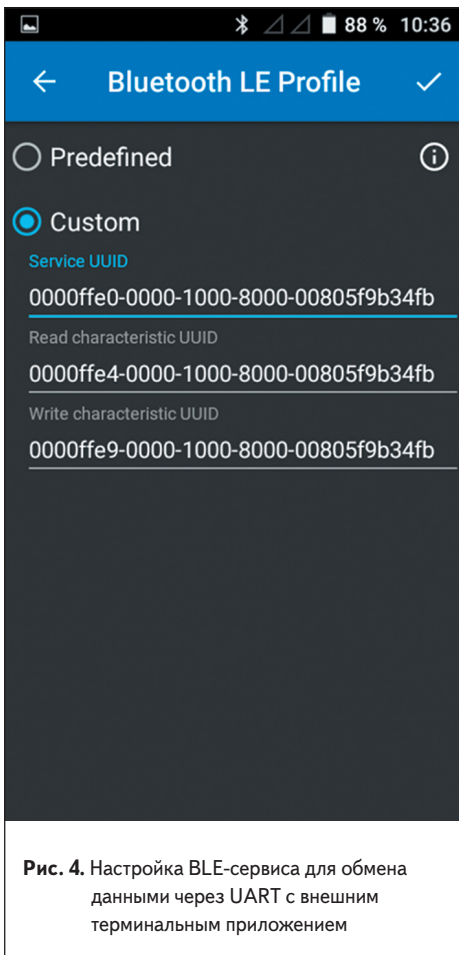


Рис. 4. Настройка BLE-сервиса для обмена данными через UART с внешним терминальным приложением

маяки, часы, сенсоры и другие приложения, на базе которых можно разработать практически все необходимые устройства, использующиеся в IoT.

После успешной компиляции Keil генерирует HEX-файл пользовательской прошивки, готовый для загрузки в модуль. Загрузка осуществляется через UART-интерфейс с помощью специальной утилиты [7], работающей в среде Windows (рис. 5). Компьютер необходимо предварительно подключить к UART модуля, используя переходник FTDI USB — UART, обеспечивающий 3-V уровни сигналов.

Для активации внутреннего загрузчика перед подачей питания на модуль необходимо подсоединить вывод ТМ модуля (описано выше) к шине VCC. В прошивочной утилите двойным щелчком по полям загружаемых файлов нужно выбрать hex-файл пользовательской прошивки и файл загрузчика *ota_dualbank_noFCT.hex* из SDK-папки *CMT4502.SDK2.0.0exampleOTAOTA_internal_flashbin*. После завершения прошивки необходимо отключить вывод ТМ для нормального старта модуля при следующем его включении. ■

Литература

1. CMT4502: Bluetooth Low Energy (BLE) System on Chip. www.hoperf.com/ic/rf_BLE/CMT4502.html
2. CMT4502 Technical reference manual. www.hoperf.com/data/upload/portal/20200320/
3. HM-BT4502B Bluetooth Low Energy (BLE) Pass-through Module. [www.hoperf.com/modules/Bluetooth/HM-BT4502\(-1\).html](http://www.hoperf.com/modules/Bluetooth/HM-BT4502(-1).html)
4. HM-BT4502B Bluetooth Low Energy Pass-through Module Specification. www.microchip.ua/wireless/HM-BT4502B.pdf
5. Serial Bluetooth Terminal. www.play.google.com/store/apps/details?id=de.kai_morich.serial_bluetooth_terminal&hl=ru
6. CMT4502 SDK. 2.0.0. www.hoperf.com/data/upload/back/20190628/CMT4502%20SDK%202.0.0.rar
7. CMOSTEK CMT4502 firmware down-load tool. www.microchip.ua/wireless/cmt4502_tool.rar

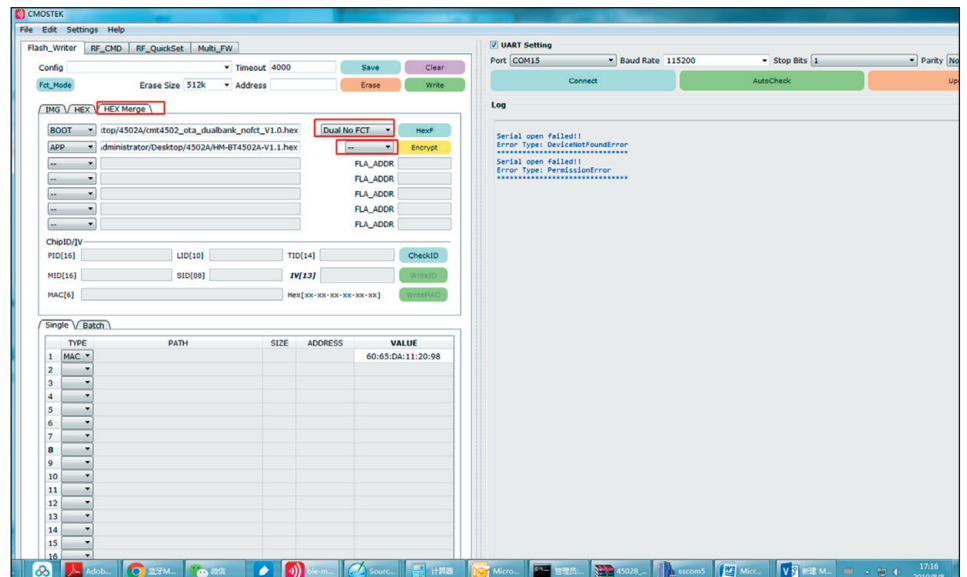


Рис. 5. Утилита от CMOSTEK для загрузки пользовательского кода в модуль HM-BT4502B