

ОБНОВЛЕНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ПО РАДИОКАНАЛУ В БЕСПРОВОДНЫХ СЕТЯХ EMBER

Татьяна Кривченко, к.т.н., начальник отдела беспроводных технологий, ООО «ЭФО»

На начальной стадии разработки беспроводной сети, когда в отладочной системе присутствует небольшое количество устройств, программист не испытывает затруднений при их программировании. Однако при увеличении сети и разнесении отдельных узлов по территории становится неудобно, а иногда и невозможно бегать от узла к узлу, например, по разным этажам здания, с внутрисхемным программатором. Для решения этой проблемы компания Ember предлагает технологию обновления программного обеспечения узлов беспроводной сети по радиоканалу.

Основой системы удаленного программирования является предлагаемая компанией Ember программа-загрузчик, которая изначально заносится в память программ микроконтроллера беспроводного узла, например, кристалла EM250.

Программа-загрузчик Ember является автономной (standalone boot-loader), т.к. не имеет непосредственной связи с программным кодом приложения. Микроконтроллер сетевого узла может работать в двух основных режимах: под управлением загрузчика или под управлением приложения.

При работе под управлением загрузчика поддерживается возможность «подгрузки» программного кода приложения по последовательному каналу с использованием стандартного протокола XMODEM CRC или по радиоканалу, по протоколу низкого уровня стандарта 802.15.4. В случае, если код приложения не загрузился, микроконтроллер продолжает работать под управлением загрузчика и поддерживает возможность бесконечного количества повторных попыток «загрузки» до тех пор, пока код приложения не будет успешно записан в память микроконтроллера.

Помимо автономного загрузчика, в состав библиотеки EmberZNet входит набор утилит, API- и callback-функций, которые позволяют программному коду приложения как на целевом устройстве, так и на устройстве-источнике взаимодействовать с программой-загрузчиком. Эти утилиты и функции предоставляются в исходных кодах, хотя Ember не рекомендует изменять их.

Загрузчик Ember использует бинарный формат файлов «прошивки». Выбор этого формата обусловлен его компактностью, возможностью передачи в одном файле и кода программы, и данных, а также проверки целостности «прошивки» подсчетом контрольных CRC-сумм.

Перед обновлением программного обеспечения целевого узла его следует перевести в режим работы под управлением загрузчика. Переход в этот режим может осуществляться следующими способами:

- аппаратно — путем удержания определенного входа микроконтроллера в состоянии низкого логического уровня во время выхода микроконтроллера из состояния сброса;
- из приложения при помощи вызова определенной API-функции;
- удаленно по радиоканалу.

Разработчик беспроводной системы может заложить различные сценарии инициации обновления программного обеспечения удаленного узла. Рассмотрим вариант, когда все узлы уже работают в сети. Один из узлов, являющийся инициатором обновления программного обеспечения системы, при помощи широковещательной рассылки запрашивает версии «прошивок» удаленных узлов. В случае необходимости обновления программного обеспечения он дает команду требуемым удаленным узлам перейти в режим загрузки на том же частотном канале, на котором работает вся сеть.

Возможен и другой сценарий, когда удаленный узел не доступен по сети и не имеет информации, на каком частотном канале выполнять загрузку. Такое может произойти, например, при сбое предшествующей загрузки, или при инсталляции нового узла. В этом случае целевой узел по умолчанию переходит на загрузку программного кода приложения на 13 канале ZigBee.

В руководстве Ember выделяются следующие возможные варианты обновления программного обеспечения узлов:

- загрузка по последовательному каналу;
- транзитная пересылка по радиоканалу (Passthru);
- транзитная пересылка по радиоканалу с ретрансляцией (Multi-hop passthru);
- клонирование по радиоканалу (Cloning).

При переходе узла под управление загрузчика он постоянно опрашивает вход последовательного канала, и если получает символ «Enter», то переходит в режим загрузки по этому проводному каналу (см. рис. 1). В этом режиме со стороны персонального компьютера может использоваться стандартная терминальная программа, например, HyperTerminal, в окне которой в ответ на символ «Enter» пользователь увидит меню команд, которые программа-загрузчик готова принять по последовательному каналу и выполнить. При помощи этих команд можно считать текущую версию программного обеспечения, инициировать обновление «прошивки» узла, а затем передать управление вновь записанному программному коду приложения на целевом устройстве.

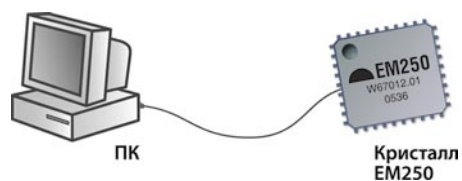


Рис. 1. Загрузка по последовательному каналу



Рис. 2. Транзитная пересылка по радиоканалу



Рис. 3. Транзитная пересылка с ретрансляцией

Если программа-загрузчик на целевом узле не получает входных данных по последовательному каналу, она готова принимать данные по радиоканалу.

Как было сказано выше, предлагается три схемы программирования целевого устройства по радиоканалу, которые представлены соответственно на рисунках 2, 3 и 4.

Во всех случаях устройство-источник и целевое устройство должны находиться в пределах радиуса действия друг друга. Устройство-источник передает «прошивку» целевому устройству, работающему под управлением загрузчика, используя упрощенный протокол MAC-уровня, который базируется на протоколе XMODEM CRC с 64-битными блоками данных, которые могут быть включены в один пакет стандарта 802.15.4.

Узел — источник, который передает код «прошивки», — сам не переходит в режим загрузки, а работает под управлением программного кода приложения, базирующегося на библиотеке EmberZNet и использующего предлагаемые компанией Ember загрузочные утилиты, API- и callback функции. Также не переходит в режим загрузки и все промежу-

точные узлы-ретрансляторы, которые работают как обычные роутеры сети ZigBee.

Транзитная пересылка (см. рис. 2) предполагает, что устройство-источник имеет проводную связь с персональным компьютером.

Как только устройство-источник определит, что целевое устройство находится в режиме загрузки, оно инициирует процесс загрузки, вызывая соответствующую API-функцию с указанием режима загрузки. При этом программный код приложения принимается устройством-источником из последовательного канала по протоколу XMODEM и транслируется целевому узлу по радиоканалу.

Транзитная пересылка с ретрансляцией (см. рис. 3) является расширением простой транзитной пересылки и используется, когда необходимо передать «прошивку» на устройство, расположенное достаточно далеко от компьютера.

В этом случае проводное соединение между компьютером и устройством-источником заменяют на цепочку беспроводных устройств. Устройство-шлюз получает «прошивку» по проводному последовательному каналу и затем использует стандартный сетевой протокол для пересылки «прошивки» по сети на устройство-источник. Устройство-источник преобразует формат сообщения высокого уровня Zigbee в низкоуровневый протокол, используемый автономным загрузчиком Ember на целевом узле.

Клонирование (см. рис. 4) удобно использовать в тех случаях, когда затруднительным оказывается

использование персонального компьютера для обновления программного обеспечения узлов сети, например, на объекте. Вместо этого можно занести последнюю версию программного обеспечения в один из узлов. Затем такой узел, используя библиотечные функции Ember, может сгенерировать код «прошивки», считывая содержимое своей памяти программ и данных, и загрузить его в целевое устройство.

Смешанные режимы. Разработчик имеет возможность реализовать в своей системе различные комбинации рассмотренных режимов. Например, при клонировании можно организовать передачу сгенерированной «прошивки» на целевое устройство с использованием промежуточных ретрансляторов. В режиме транзитной пересылки устройство-источник может получать «прошивку» из последовательного канала не только от персонального компьютера, но также, например, из последовательного ПЗУ.

Технология удаленного обновления программного обеспечения Ember предусматривает средства защиты беспроводной системы от несанкционированного обновления программного обеспечения. Устройство-источник и целевое устройство перед началом процесса загрузки обмениваются рядом сообщений, сравнивая версии программы загрузчика, идентификаторы производителя, кристалла, целевой платы, а также уровень радиосигнала. В этот же момент целевое устройство проверяет полномочия источника, сравнивая ключи шифрования.

Для освоения технологии удаленного обновления программного обеспечения беспроводных узлов сети компания Ember предоставляет демонстрационный проект приложения в исходных кодах Standalone bootloader demo, который показывает, как использовать загрузочные утилиты и функции библиотеки EmberZNET в конечном приложении. Этот демонстрационный проект реализует простой командный интерфейс, который позволяет вручную инициировать различные режимы загрузки программного обеспечения по радиоканалу, и может послужить как средством обучения, так и отладочным средством, позволяющим осуществлять обновление программного обеспечения удаленных узлов сети.

ЛИТЕРАТУРА

1. Using the Standalone Bootloader, Руководство Ember, 20.10.2006.

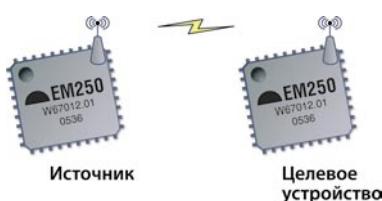


Рис. 4. Клонирование