



Всеволод Нестеров (КОМПЭЛ)

## ПРОТОКОЛ SIMPLICITY



Данная статья посвящена применению *SimpliciTI* в различных возможных топологиях.

Гибкость и универсальность нового протокола *SimpliciTI* от компании *Texas Instruments* позволяют разработчику использовать его в приложениях охранной и пожарной автоматики, домашней автоматизации и счетчиках.

Протокол *SimpliciTI* был разработан компанией для небольших беспроводных сетей, имеющих не более 100 узлов. С его помощью можно организовывать сети по топологиям «звезда», «звезда» с ретрансляцией и «точка-точка».

Благодаря малому размеру кода требования к аппаратной части довольно низки, из-за чего себестоимость сети получается довольно незначительной. Протокол разрабатывался с учетом того, что устройства большую часть времени должны быть в «спящем» режиме, благодаря этому потребляемый элементами сети ток достаточно мал. Гибкость и универсальность протокола позволяет использовать для построения сети различные микроконтроллеры и приемопередатчики разных частотных диапазонов. Перечисленные достоинства дают возможность применения *SimpliciTI* в таких областях как охранная и пожарная автоматика, счетчики газа, воды, электросчетчики, домашняя автоматизация и многих других. Пример топологии сети на базе протокола *SimpliciTI* показан на рисунке 1.

Для начала работы с протоколом можно приобрести отладочную плату микроконтроллера MSP430 с интерфейсной платой трансиверов CC1100 или CC2500 (рисунок 2).

Можно также использовать отладочный набор eZ430-RF2500 с трансивером CC2500 (рисунок 3).

Рассмотрим пример реализации протокола *SimpliciTI* на базе этого набора. Этот набор поставляется с уже загруженным программ-

ным обеспечением *SimpliciTI* двух типов узлов: точки доступа и конечного устройства. Пример называется *Wireless Sensor Monitor*. Точка доступа имеет стационарное питание и управляет сетью. Конечное устройство имеет батарейное питание и основную часть времени находится в спящем режиме. Раз в секунду оно просыпается, измеряет температуру окружающей среды,

передает данные на точку доступа и снова засыпает. Точка доступа подключена через USB-интерфейс к компьютеру и с помощью программы *Network Visualizer GUI* можно просмотреть информацию о температуре с каждого датчика. Точка доступа также может измерять температуру окружающей среды, кроме того, она постоянно тестирует сеть на факт появления новых устройств.

Блок-схема алгоритма работы точки доступа приведена на рисунке 4.

Исходный код этой программы для точки доступа содержится в файле *demo\_AP.c*.

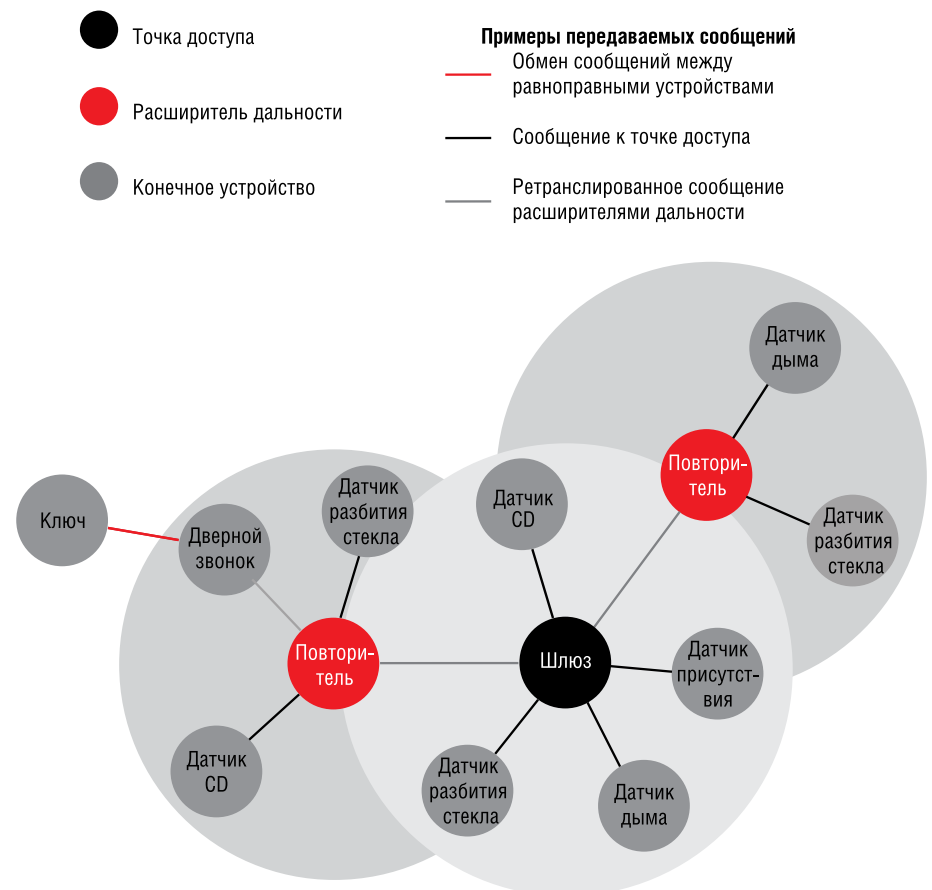


Рис. 1. Пример топологии протокола *SimpliciTI*

Таблица 1. Описание структуры фрейма

Поле	Описание	Комментарии
Preamble	Радиосинхронизация	Заполняется трансивером
Sync	Радиосинхронизация	Заполняется трансивером
Length	Длина пакета в байтах	Заполняется программно при передаче. Частично фильтруется при приеме.
MISK	Различные данные	Различно для разных трансиверов, может не заполняться
DSTADDR	Адрес получателя	Заполняется программно. Фильтруется в зависимости от трансивера.
SRCADDR	Адрес отправителя	Заполняется программно.
PORT	Биты шифрования – 6-7 Номер порта приложения – биты 5-0	Заполняется программно. Номера портов 0x20 – 0x3F зарезервированы за приложением пользователя, 0-1F – за сетевым уровнем
Device info	Отправитель/получатель и возможности платформы	Заполняется программно.
TRACTID	Идентификатор транзакции	Заполняется программно. Порядок зависит от контекста, каждый раз увеличивается.
APP Payload	Данные приложения	$0 \leq n \leq 52$ (50 байт для некоторых трансиверов, которые берут данные о 64 байтовом буфере FIFO для получения информации о сигнале)
FSC	Контрольная сумма	Обычно CRC прибавляется самим трансивером

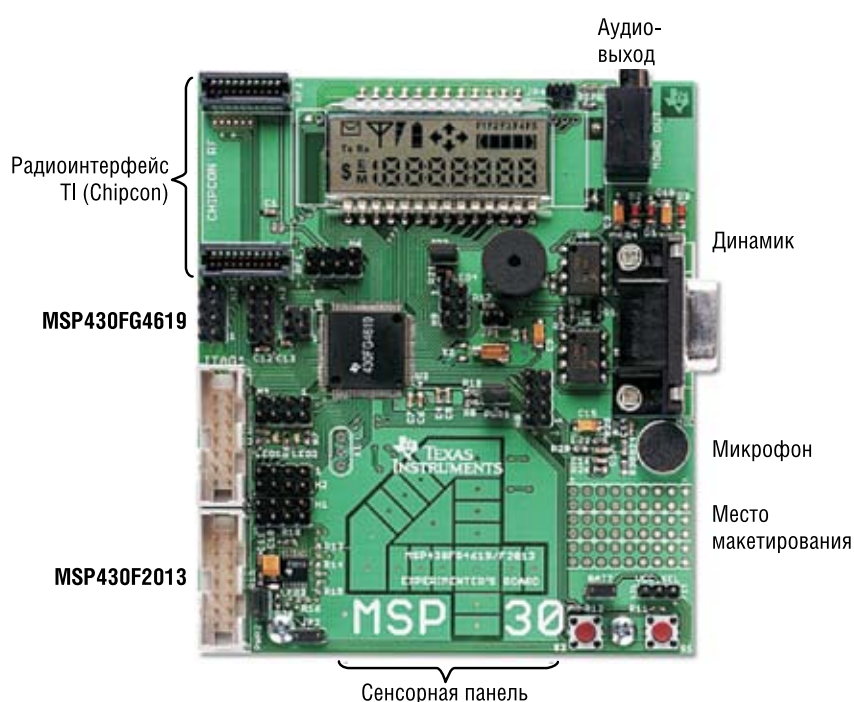


Рис. 2. Отладочная плата MSP430

На этапе инициализации функция `BSP_Init()` инициализирует светодиоды, функция `MCU_init()` инициализирует процессор, далее выдается сообщение на COM-порт, и функция `SMPL_init(sCB)` инициализирует трансивер и протокол `SimpliciTI`. Параметр `sCB` представляет собой указатель на функцию обратного вызова, которая будет выполняться в обработке прерываний при получении точкой доступа пакета. Эта функция определяет источник посылки пакетов в зависимости от идентификатора канала. Если идентификатор канала имеет нулевое значе-

ние, это означает, что происходит запрос на подключение. В случае принятия запроса точка доступа устройству очередной идентификатор из диапазона от `0x01` до `0x1D`. Пример `Wireless Sensor Monitor` версии 1.03 позволяет подключить максимум 8 конечных устройств к точке доступа.

В зависимости от идентификатора канала функция обратного вызова `sCB` определяет и увеличивает значения соответствующих семафоров `sPeerFrameSem` или `sJoinSem` для того, чтобы обработать их в теле основной програм-

мы. В программе также используется семафор `sSelfMeasureSem`, который устанавливается таймером `B`, чтобы каждую секунду измерять напряжение питания и температуру точки доступа. Таким образом, после инициализации логикой программы управляют три семафора. Функция обратного вызова `sCB` возвращает определенное значение. Если это значение больше нуля, значит, функция обработала принятый пакет и освободила память. В коде `demo_AR.c` функция возвращает нулевое значение, потому что оставляет принятый пакет в памяти для его обработки самим приложением. Это сделано для того, чтобы уменьшить время, проведенное в обработке этого прерывания. Когда устройство работает со многими конечными устройствами, важно иметь код небольшого размера в обработке, чтобы минимизировать риск потерь пакетов.

Блок-схема алгоритма работы конечного устройства представлена на рисунке 5.

Инициализация конечного устройства похожа на инициализацию точки доступа, однако есть некоторые отличия: адрес точки доступа генерируется случайным образом и передается в функцию инициализации `SMPL_init()`. Таким образом, если конечное устройство по какой либо причине перезапустится, оно получит тот же самый адрес, по которому его запо-



Рис. 3. Отладочный набор eZ430-RF2500

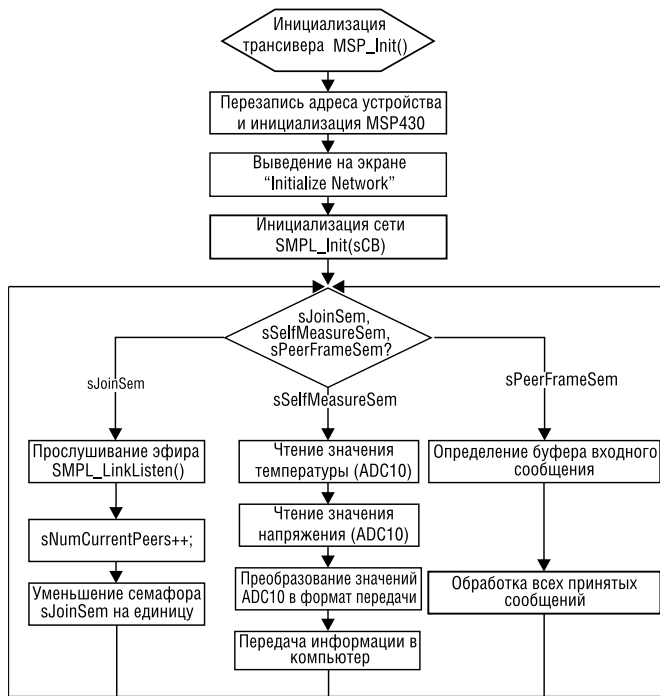


Рис. 4. Блок-схема алгоритма работы дочки доступа

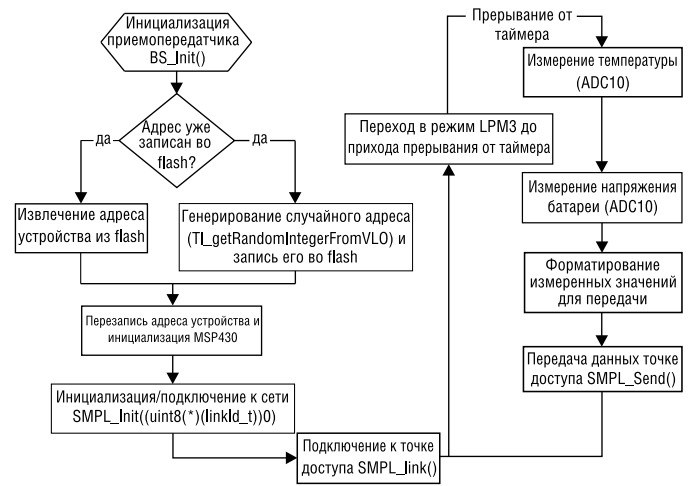


Рис. 5. Блок-схема алгоритма работы конечного устройства

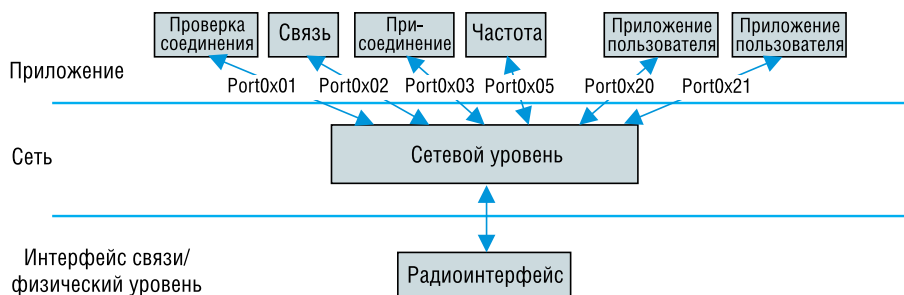


Рис. 6. Программные уровни протокола SimpliciTI

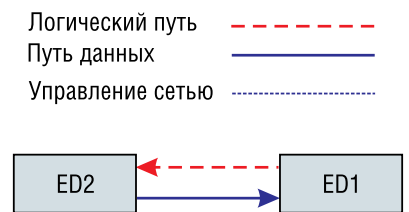


Рис. 7. Соединение точка-точка (ED1,2 - конечное устройство 1,2)

нила точка доступа, и для точки доступа оно не будет новым конечным устройством. После инициализации конечное устройство подключается к точке доступа, переходит в режим пониженного энергопотребления, из которого

оно раз в секунду пробуждается по прерыванию таймера, измеряет напряжение питания и температуру, передает на точку доступа и опять засыпает.

Более подробно с работой сети на примере Wireless Sensor Monitor

можно познакомиться в документе slaa378b.pdf, который можно скачать с сайта Texas Instruments.

Концептуально протокол SimpliciTI состоит из трех программных уровней, которые показаны на рисунке 6.

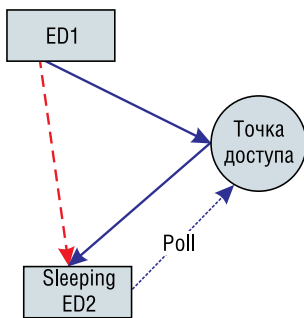


Рис. 8. Соединение точка-точка с сохранением и перенаправлением данных

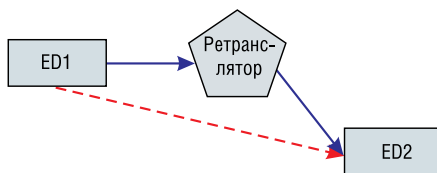


Рис. 9. Соединение точка-точка через ретранслятор

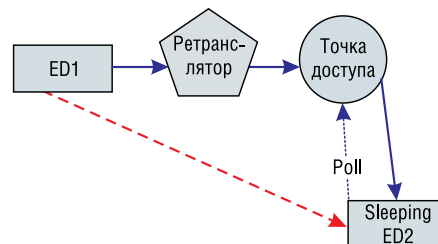
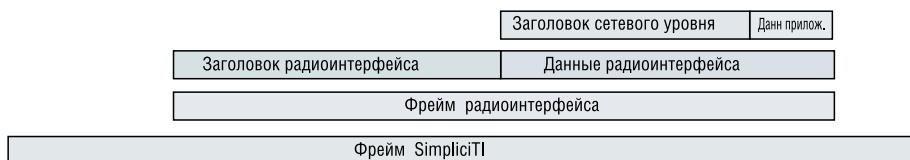


Рис. 10. Соединение точка-точка с сохранением и перенаправлением данных через ретранслятор и точку доступа

Преамбула	Поле синхр.	Длина	Разл. данные	Адрес получателя	Адрес отправителя	Порт	Информация об устройстве	Идентификат. транзакции	Данные приложения	Контрольная сумма
RD*	RD*	1	RD*	4	4	1	1	1	n	RD*



\*RD – зависит от радиоинтерфейса, заполняется им, либо самим трансивером

Рис. 11. Структура фрейма протокола SimpliciTI

В этой архитектуре пользователю достаточно только работы с уровнем приложения пользовате-

ля для построения законченного устройства. Поддержка связи обеспечивается простой установкой

API-символов, используемых для инициализации сети, чтения и записи сообщений. Эта архитектура отличается от 7-уровневой сетевой архитектуры OSI. В SimpliciTI нет уровня PHY или MAC/LLC: данные принимаются непосредственно от радиоинтерфейса уже сформированными, таким образом, радиоинтерфейс выполняет функции этих уровней. Также в SimpliciTI нет уровня представления, где его функции берет на себя сетевой уровень, и нет транспортного уровня, в случае его необходимости его можно реализовать в уровне приложений.

Возможные топологии сети SimpliciTI показаны на рисунках 7, 8, 9, 10.

Структура фрейма протокола SimpliciTI представлена на рисунке 11.

Описания структуры фрейма приведено в таблице 1.

Более детальную информацию по протоколу SimpliciTI можно получить, скачав с сайта [www.ti.com](http://www.ti.com) пакет, который содержит компоненты протокола, документацию и примеры.

# Сетевой протокол SimpliciTI™

для небольших беспроводных сетей с низким энергопотреблением и малым объемом передачи данных

- Предназначен для диапазонов от 433 до 2400 МГц
- Топологии: «точка – точка», «звезда»
- Увеличение дальности с помощью повторителя
- Использование минимальных ресурсов микроконтроллера
- Всего пять команд управления
- Для устройств с батарейным питанием
- Простота внедрения
- Низкая стоимость

www.compel.ru

Ответственный за направление в КОМПЭЛе – Мария Рудяк

Получение технической информации, заказ образцов, поставка – e-mail: [wireless.vesti@compel.ru](mailto:wireless.vesti@compel.ru)