РАЗДЕЛ II. ФИЗИКА

Смагулов Ж.К., Мырзахметова А.Ж.

Использование беспроводной связи для автоматизаций отопления жилого дома

Карагандинский Государственный Университет им Е.А.Букетова

(Казахстан, Караганда)

doi:10.18411/spc-12-03-2018-02

idsp: 000001:spc-12-03-2018-02

Аннотация

В настоящее время в связи с удорожанием и ограничением запасов ископаемого топлива, все чаще используют комбинированные системы для обогрева жилых и производственных зданий с использованием возобновляемых и не возобновляемых источников тепла. В данной работе представлен анализ и выбор беспроводной технологии для автоматической системы управления обогревом жилого дома от трех источников тепла: газового котла, солнечного теплообменника и ветрогенератора.

Ключевые слова: модуль, фрейм, приложение, сеть, топология

При проектировании системы беспроводной системы автоматического управления было использовано следующее оборудование: 2 беспроводных модуля XBeePro S2B компании Digi; контроллер Arduino MEGA 2560 с микроконтроллером ATMEGA2560; аналоговый датчик температуры LM335.

Модули XBee-Pro – это новые повышенной мощности беспроводные модули, которые доступны в двух вариантах – простом и программируемом. Новые модули (платформа S2B) имеют улучшенные технические характеристики – дальность связи на открытой местности достигает 3200 метров. Модули при их нагрузке на всю мощность употребляют меньше энергии (205 мА против 295 мА), так же имеют встроенный температурный датчик и защиту от перегрева и повышенного напряжения питания. Настройка ZigBee-части модуля выполняется через внутренний UART-интерфейс, что обеспечивает целостность ZigBee-стека и увеличивает общую стабильность работы устройства. Контроллер ArduinoMega собран на микроконтроллере ATmega2560. В микросхему входят 54 цифровых входов/выходов, 14 из которых могут быть применены в качестве выходов ШИМ. Контроллер Mega 2560 совместим со многими платами расширения, разработанными для контроллеров Uno или Duemilanove. В таблице 1 приведены краткие характеристики микроконтроллера ATmega2560.

Таблица 1

Краткие характеристики ATmega2560

Микроконтроллер	ATmega2580
Рабочее напряжение	5В
Входное напряжение(рекомендуемое)	7-12В
Входное напряжение(предельное)	6-20В
Цифровые Входы/Выходы	54(14 из которых могут работать как выходы ШИМ)
Аналоговые входы	16
Постоянный ток через вход/выход	40mA
Постоянный ток для выхода 3.3В	50mA
Флеш-память	256 КБ (из которых 8 КБ используется для разгрузчика)
ОЗУ	8 КБ
Энергонезависимая память	4КБ
Тактовая частота	16 MHz

Аналоговый датчик температуры - LM335 с диапазоном измерений от -40°C до $+100^{\circ}\text{C}$ и погрешностью в 1°C . LM335 - это стабилитрон с приведенным температурным коэффициентом напряжения ($\text{TKH} = 10 \text{ мВ/К}$) [1]. Задавая значение тока резистором через датчик в интервале от 0.45 мА до 5 мА , принимаем напряжение на устройстве, которое в десятках мВ представляет абсолютную температуру в градусах Кельвина.

Для разработки беспроводного канала в диапазоне 2,4 ГГц были использованы радиомодули XBee фирмы Digi. Радиомодули XBee зарекомендовали себя как надежные, недорогие и очень простые в освоении устройства. Встроенное программное обеспечение (802.15.4 / DigiMesh / ZigBee / SmartEnergy) позволяет построить как простейший радиосвязь между двумя точками (до 3.2 км), так и полноценную Mesh-сеть, состоящую из сотен узлов. Уникальный протокол DigiMesh с поддержкой спящих маршрутизаторов может оказаться крайне полезным при создании протяженных сетей с питанием от батарей. Радиомодули XBee не требуют для своей работы никаких внешних компонентов. Управление осуществляется с помощью AT-команд или API-фреймов, отсылаемых по UART, SP или USB-интерфейсу. Были изучены и сравнены оба режима передачи информации: AT-режим (прозрачный режим) или с помощью структурированных пакетов (API-режим). Режим AT-команд очень прост для понимания и оптимален при передаче непрерывных потоков данных в сетях с не сложной топологией. В AT-режиме радиомодуль XBee просто ретранслирует последовательные данные на другой приемный модуль XBee, согласно заданному в памяти адресу. Этот режим удобен в тех случаях, когда не нужно часто менять адреса назначения удаленного модуля, или в случае использования простой топологии сети такой, как «точка-точка». Режим API (Application Programming Interface) гораздо более гибок, особенно при управлении посредством внешнего контроллера, что позволяет отправлять и данные, и команды управления в общем потоке. Помимо этого, некоторые возможности в режиме AT-команд являются недоступными. Например, настроить или изменить параметры удаленного модуля есть возможность только в API-режиме. Также режим API предполагает обмен информацией с модулем с помощью блоков структурированных данных, называемых фреймами. API определяет правила формирования фреймов, в соответствии с которыми команды, ответы и статусные сообщения отправляются и принимаются из модуля [2]. Для настройки беспроводных модулей в тот или иной режим использовалась бесплатная и свободно-доступная программа X-CTU, разработанная специалистами компании Digi. Для создания беспроводной сети в режиме AT-команд с топологией «точка-точка» необходимо наличие одного координатора и роутера. Для настройки модулей необходимо:

- 1) подключить модули к ПК через интерфейс USB;
- 2) открыть программу X-CTU.
- 3) выбрать COM-порт подключенного модуля XBee.

Для настройки модуля в качестве координатора необходимо выполнить следующее: в параметре CE (CoordinatorEnable) устанавливаем значение 1;

в параметре DL (DestinationAddressLow) устанавливаем значение FFFF - для широковещательной передачи данных.

Два модуля (роутер и координатор) подключены к двум разным ПК, создавая сеть с простой топологией рис.1.

CoordinatorRouter

Рис.1 Сеть с модулями в AT-режиме.

Используя среду графического программирования LabVIEW, разработан интерфейс пользователя. Для связи среды графического программирования LABVIEW с контроллером Arduino было использовано специальное приложение LIFA. При работе в режиме API все входящие и отправляемые данные содержатся во фреймах, определяющих поведение модуля. Фреймы данных на передачу (получаемые через DOUT-output (вывод 2) включают: фрейм данных для передачи по RF-каналу; фрейм управления (эквивалент AT-командам). Фреймы приема данных (отправляемые через DIN- input (вывод 3) включают: фреймы приема RF-данных; ответ на команду; оповещения - перезапуск, активацию и деактивацию ассоциирования и т.д.

Режим API обеспечивает альтернативный способ настройки модуля и маршрутизации данных на уровне хост-приложения. Теперь для изменения параметров модуля, например, адреса назначения, не обязательно переходить в режим управления. Вместо этого хост-приложения должно отправить модулю фрейм с необходимыми параметрами и данными. Прием данных от модуля также осуществляется с помощью фреймов, содержащих ответы на команды, статусные пакеты, адресную информацию или данные.

Используемые модули XBeeSeries 2 поддерживают API фреймы, указанные в таблице 2.

Таблица 2

Наименования и значения API фреймов

В целях изучения структуры передачи данных и формирования фреймов в API-режиме с помощью программы X-CTU, изучена передача управляющей команды, назначенной для включения индикатора на удаленном модуле. Для включения индикатора на удаленном модуле был использован 17-фрейм – фрейм запроса удаленной команды. Данный фрейм используется, чтобы запросить или установить параметры модуля на отдаленном устройстве.

Структура фрейма подробно представлена в таблице 3.

Таблица 3

Структура 17 фрейма

Для отправки API-фрейма с помощью программы X-CTU, необходимо во вкладке «Terminal» записать фрейм и отправить на удаленный модуль.

После отправки API-фрейма практически сразу приходит ответный фрейм, в данном случае это фрейм 97 – фрейм ответ на удаленную команду. Данный фрейм указывает была ли команда выполнена или же показывает ошибки при передаче данных.

Разработана беспроводная сенсорная сеть (БСС), которая может служить в качестве модели для реализации любых сенсорных систем.

Прежде чем начать строить БСС, необходимо правильно настроить беспроводные модули. При настройках беспроводных модулей использовалась последняя версия программы X-CTU 5.2.8.6.

Топология разработанной БСС представлена на рисунке 2, полученной с помощью ПО X-CTU.

Рисунок 2 – Топология БСС

Для визуализации полученных значений температуры был использован программный продукт Processing. Processing – это среда разработки программного обеспечения с открытым исходным кодом, предназначенный для визуализации данных. В результате проведенного анализа и исследования режимов передачи данных беспроводных модулей Xbee можно сделать следующие выводы: использование передачи данных в прозрачном режиме имеет следующие особенности: простая конфигурация сети и настройка адресации; совместимость с любыми устройствами имеющими последовательный интерфейс; передача данных в общем потоке (поточная передача данных, также используемая в сотовых связях). К недостаткам АТ-режима можно отнести: медленный переход из прозрачного режима передачи данных в режим команд управления; не позволяет использовать сложные топологии сети; не оптимизирован для встраиваемых и сложных систем передачи и сбора информации. Работа в режиме API упрощает многие операции, например, такие как: прием и передача данных на множество адресов без перехода в режим управления; получение статуса об отправке данных для каждого из отправленных RF-пакетов, а так же другой дополнительной информации; идентификацию исходного адреса для каждого полученного пакета; конфигурацию сети и удаленных модулей «по воздуху» (overtheair-беспроводно); использование всех возможностей модулей [3].

Основным недостатком данного режима является сложность программирования формирования пакетов передачи данных (фреймов).

Таким образом, использование прозрачного режима является удобным, в случаях использования простой сети передачи данных и отсутствия необходимости частой конфигурации сети и адресов назначения. Для больших и сложных сетей, включающих узлы, назначенных на управление несколькими потоками, режим API является более оптимальным, т.к. позволяет проводить расширенную передачу данных, контролировать получаемые пакеты, обрабатывать события.

***

1. Mingming G., Liangshan S., Ning L. The Design of Wireless Smart Home System based on ZigBee // IACSIT Press. – 2012. – №53
2. Ю. Незнамов, В. Козаченко Перспективы использования беспроводных ZigBee – интерфейсов в электроприводе // Электронные компоненты. 2008. № 2 – С. 17–24;
3. Faludi R. Building Wireless Sensor Networks with ZigBee, XBees, Arduino, and Processing. // A practical guide to the ZigBee Mesh Networking Protocol. – USA: O'Reilly Media, Inc., 2011- 321p.