

Лабораторная работа 1

Передача информации по сети MiWi P2P

1. Подготовка отладочной платы

Убедитесь, что радиомодуль MRF49XA установлен в верхнюю часть разъема PICTailPlus на плате Explorer 16 .

2. Цель

В этой лабораторной работе Вы научитесь конфигурировать Ваше приложение для обмена сообщениями в радиосети MiWi.

3. Требования

Среда разработки : MPLAB IDE v8.60

Компилятор : MPLAB C30 C Compiler

Оборудование: Explorer 16, MRF49XA PICTail Card

Расположение файлов: C:\Microchip Solutions v2010-10-19\MiWi DE Demos\Node 2\

4. Задание

А) Сконфигурируйте приложение, и подключитесь к MiWi сети

Б) Передайте нешифрованное ширококестательное сообщение в сеть.

В) Передайте зашифрованное адресное сообщение на компьютер преподавателя

5. Процесс

Закройте все окна, оставшиеся от предыдущей работы.

Запустите MPLAB.

Откройте проект masters2010lab.mcp

В окне проекта в разделе “Header Files/Application” найдите файл “ConfigApp.h” и откройте его.

Установите в качестве используемого протокола протокол MiWi P2P.

Установите в качестве используемого радиомодуля модуль MRF49XA.

Установите уникальный 8-байтный адрес вашей платы. Для наглядности присвойте байтам адреса платы ASCII символы, образующие, например, Ваше имя или любую другую уникальную информацию

Сохраните файл “ConfigApp.h”.

В окне проекта в разделе “Source Files/Application” найдите файл “LabFeatureDemoNode2.c” и откройте его.

Найдите в файле строку:

//Masters 2010 MiWiLab: ActiveScan

Это место, куда необходимо добавить код программы для подключения к сети.

Добавьте в программу вызов этой функции таким образом, чтобы Ваша плата постоянно осуществляла активный поиск имеющейся сети. При обнаружении сети на любом из каналов плата должна установить этот канал в качестве рабочего и установить прямое подключение к этой сети.

Для поиска сети используйте функцию MiApp_SearchConnection:

/*****/

```
//Функция MiApp_SearchConnection осуществляет активное сканирование
// найденных соединени на всех каналах.
// Возвращаемоезначение - это общее количество найденных соединений.
// Данные о соединениях сохраняются в глобальной переменной
// ActiveScanResults.
// Первый параметр - максимальное время сканирования. Значение 10
// соответствует приблизительно 1 секунде, 9 - пол секунды,
// 11 - 2 секунды. Максимальное значение 14, что соответствует
// 16 секундам
// Второй параметр - маска каналов.Каждый бит, установленный в 1,означает
// разрешение сканирования соответствующего канала. 0 - запрет
/*****/
```

```
BYTE MiApp_SearchConnection(BYTE ScanDuration, DWORD ChannelMap);
```

Установите время сканирования 1 сек. И маску для всех каналов.

```
Для установки рабочего канала используйте функцию MiApp_SetChannel.
/*****/
// Функция MiApp_SetChannel назначает рабочи частотный канал
// Параметр – номер канала
/*****/
```

```
BOOL MiApp_SetChannel(BYTE Channel);
```

Установите номер канала, который получен в результате сканирования. В массиве структур ActiveScanResults - это поле channel.

Для установки режима подключения к сети используйте функцию MiApp_ConnectionMode

```
/*****/
// Функция MiApp_ConnectionMode устанавливает режим соединения.
// Допускаются следующие режимы:
// - ENABLE_ALL_CONN принимать все запросы на соединение
// - ENABLE_PREV_CONN принимать запросы на соединение только от известных узлов
// - ENABL_ACTIVE_SCAN_RSP не принимать запросы на соединения, но отвечать
// на запросы активного сканирования
// - DISABLE_ALL_CONN отклонять все запросы, в т.ч. запросы активного
// сканирования
/*****/
```

```
void MiApp_ConnectionMode(BYTE Mode);
```

Установите режим работы сети с приемом всех запросов.

Для подключения к сети используйте функцию MiApp_EstablishConnection.

```
/*****/
// Функция MiApp_EstablishConnection пытается установить новое
// соединение с другим устройством.
//
// Первый параметр - это индекс соединения в таблице результатов
// активного сканирования. Если активного сканирование не производилось,
// то установка значения 0xFF в качестве первого параметра позволяет
// искать соединение с любым доступным узлом
//
// Второй параметр - это решим соединения, либо прямое соединение
// (CONN_MODE_DIRECT), либо косвенное соединение (CONN_MODE_INDIRECT)
// Косвенное соединение означает соединение с узлом, который может
```

```
// не находиться в зоне прямой радиовидимости
//
// Функция возвращает индекс подключенного соединения в таблице соединений
/*****/
```

```
BYTE MiApp_EstablishConnection(BYTE ActiveScanIndex, BYTE Mode);
```

Установите режим прямого соединения с нулевым индексом записей таблицы результатов сканирования.

Для вывода сообщения об успешном подключении на ЖКИ используйте функцию LCDDisplay.
, например:

```
LCDDisplay((char *)" Found network on Channel %d", OperatingChannel, TRUE);
```

Убедитесь, что у Вас в проекте установлен тот тип программатора, который находится у Вас на столе.

Откомпилируйте проект и загрузите его в отладочную плату

Если все сделано правильно, то после подключения к сети на Вашей плате загорится левый светодиод .

Если Вы вставили в свой код вывод сообщения на ЖКИ об успешном подключении к сети, то Вы увидите это сообщение на ЖКИ.

После чего, на ЖКИ выведется сообщение о количестве переданных и принятых сообщений

Найдите в файле строку:

```
//Masters 2010 MiWiLab: Broadcast
```

В этом месте размещается часть кода, которая выполняется по нажатию на кнопку RD6.

Добавьте в это место программы вызов функций, обеспечивающих ширококестельную передачу сообщения по нажатию кнопки RD6 . В сообщении передайте несколько ASCII символов, и закончите символами перевода строки и возврата каретки.

Для перевода буфера передачи в исходное состояние используйте функцию MiApp_FlushTx().

Для заполнения очередного байта буфера используйте функцию MiApp_WriteData(char a).

Для передачи ширококестельного сообщения используйте функцию MiApp_BroadcastPacket.
BOOL MiApp_BroadcastPacket(BOOL SecEn);

Параметр SecEn определяет требуется ли шифровать пакет перед передачей.

Установите этот параметр для передачи пакета без шифрования

Откомпилируйте проект и загрузите его в отладочную плату

После подключения к сети на Вашей плате загорится светодиод D10, и на ЖКИ выведется сообщение об успешном подключении к сети.

Нажимайте на кнопку RD6, и убедитесь, что на экране преподавателя появляется передаваемое Вашей платой ширококестельное сообщение

Найдите в файле строку:

```
//Masters 2010 MiWiLab: Unicast
```

В этом месте размещается часть кода, которая выполняется по нажатию на кнопку RD7.

Добавьте в это место программы вызов функций, обеспечивающих зашифрованную адресную передачу сообщения по нажатию кнопки RD7 . В сообщении передайте несколько ASCII символов, и закончите символами перевода строки и возврата каретки.

Для перевода буфера передачи в исходное состояние используйте функцию `MiApp_FlushTx`.
Для заполнения очередного байта буфера используйте функцию `MiApp_WriteData`.

Для передачи широковещательного сообщения используйте функцию `BOOL MiApp_UnicastConnection(BYTE ConnectionIndex, BOOL SecEn)`;
Параметр `ConnectionIndex` определяет индекс соединения в таблице соединений.
Параметр `SecEn` определяет, требуется ли шифровать пакет перед передачей.

Передайте сообщение через нулевой индекс и с шифрованием пакета.

Откомпилируйте проект и загрузите его в отладочную плату

После подключения к сети на Вашей плате загорится светодиод D10, и на ЖКИ выведется сообщение об успешном подключении к сети.
Нажимайте на кнопку RD7, и убедитесь, что на экране преподавателя появляется передаваемое Вашей платой адресное шифрованное сообщение

6. Результат

Только что Вы научились как конфигурировать ваше приложения для работы с сетью MiWi. Также вы научились добавлять функции подключения к сети и передачи сообщений. Если у Вас осталось время, попробуйте осуществить адресную передачу, используя другую, альтернативную, функцию передачи сообщения.