Chapter 11: Wireless Communication – Bluetooth and Zigbee

tw rev. 26.8.16

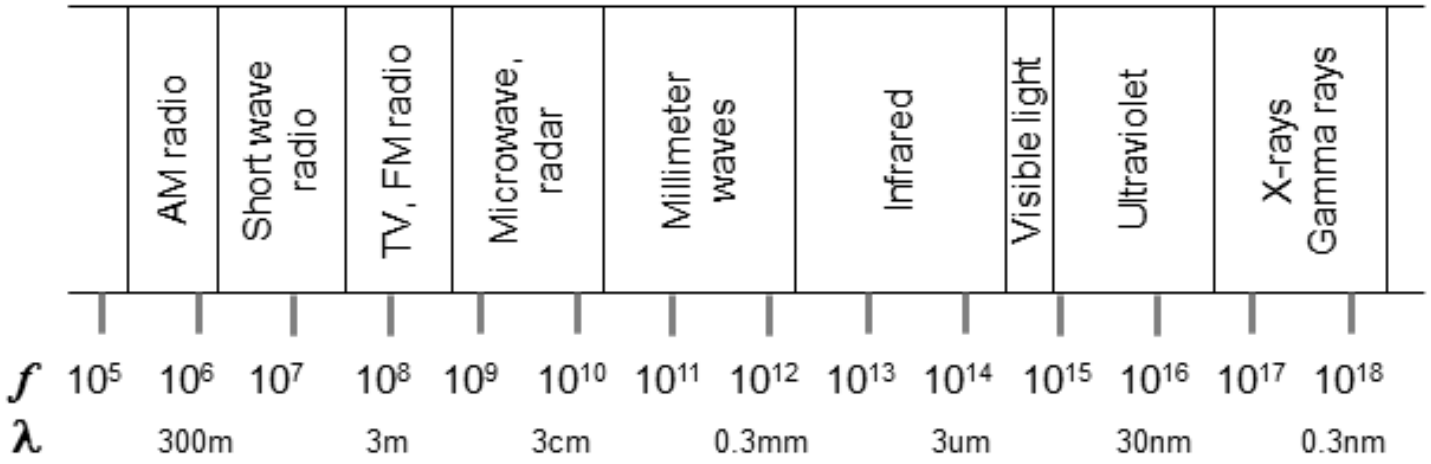
If you use or reference these slides or the associated textbook, please cite the original authors' work as follows:

Toulson, R. & Wilmshurst, T. (2016). Fast and Effective Embedded Systems Design - Applying the ARM mbed (2nd edition), Newnes, Oxford, ISBN: 978-0-08-100880-5.

Some Wireless Preliminaries 1

The electromagnetic spectrum shows where various wireless activities sit. Any frequency on the spectrum, from the lowest frequency up to visible light, can be used for data communication. Almost all of this is very strictly regulated by national and international agencies.

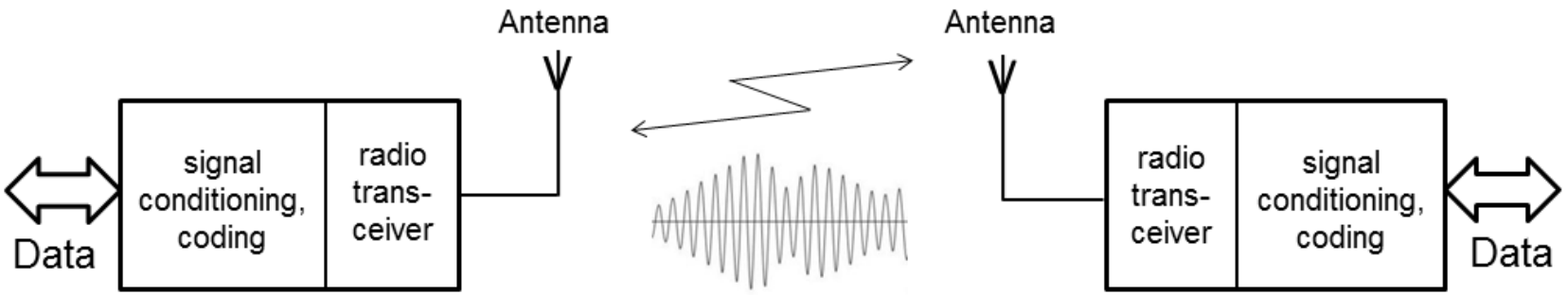
The International Telecommunication Union manages the allocation of the radio spectrum between different broadcasters and applications. It reserves certain frequency bands for Industrial, Scientific and Medical (ISM) applications. These are *unlicensed*, and some vary between countries. The 2.4 GHz band is however reserved for unlicensed use in all regions. It has become widely used, mainly for short-range, low power applications.



Some Wireless Preliminaries 2

While the spectrum represents the “pure” radio frequencies, they only become useful once they are carrying information. This is done by the process of *modulation*; the information to be carried is imprinted onto the carrier frequency, through one of a number of different techniques.

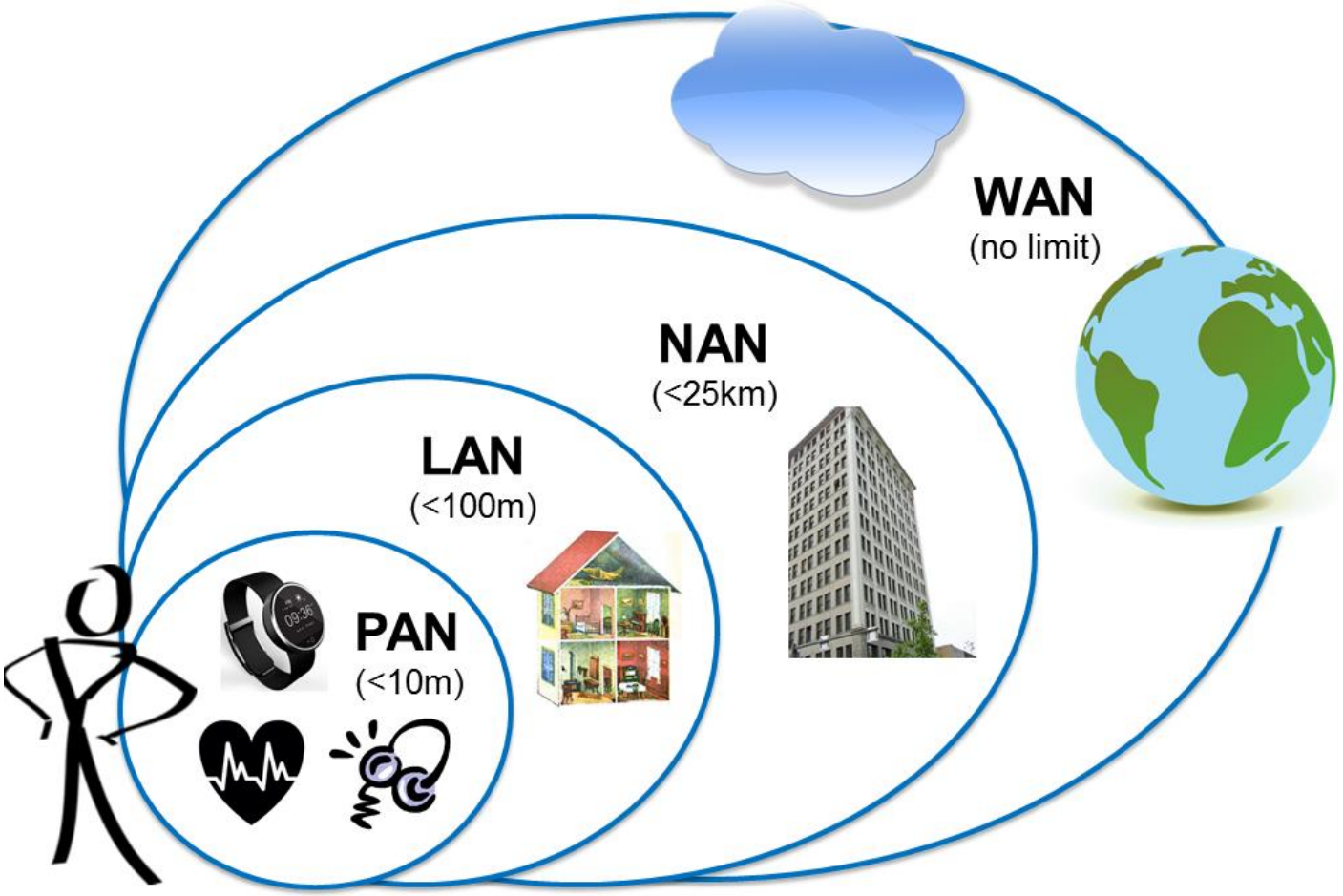
One effect of modulation is to cause fluctuations around the base frequency; thus if we say that a certain radio station can be found at the frequency of 103 MHz, in fact it is in a narrow band of frequencies centred on 103 MHz. The word *bandwidth* is used to define a range of frequencies, for example within which a particular transmission may be taking place.



Wireless Networks

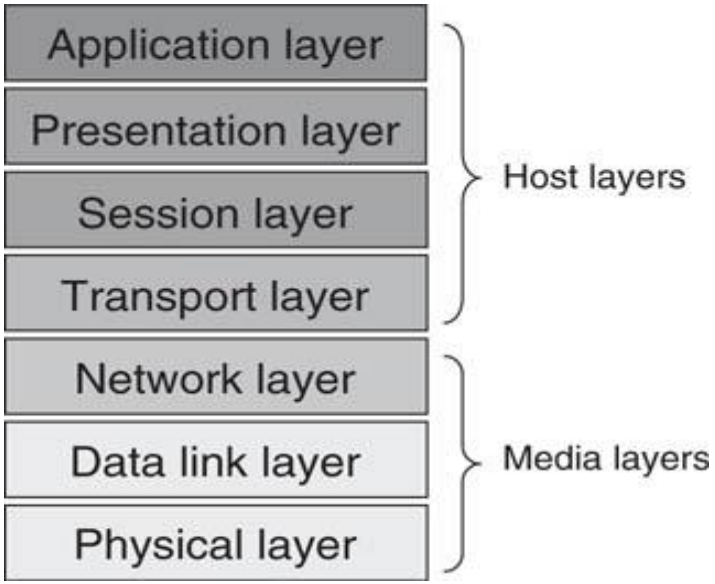
Networks can be divided into four categories:

- The Personal Area Network (PAN) usually relates to devices close to a person.
- The Local Area Network (LAN) typically applies to a single building.
- The Neighbourhood Area Network (NAN) could apply to a smart transport or smart energy system.
- The Wide Area Network (WAN) includes national or global systems, notably the Internet.



Protocols

With large networked systems, protocols can become very complicated, defining every aspect of the communication link. To aid in the process of defining a protocol, the International Organisation for Standardisation (ISO) devised a “protocol for protocols”, called the *Open Systems Interconnect* (OSI) model, as shown. Each layer of the OSI model provides a defined set of services to the layer above, and each therefore depends on the services of the layer below.



Protocols – IEEE Working Groups

The IEEE (the Institute of Electrical and Electronic Engineers) plays a major role in defining standards and protocols. It maintains a set of standards for Local Area Networks, allocated the number 802. A small number of these which are relevant to this and the next chapter are shown in the Table.

IEEE Working Group	Description
802.3	Ethernet
802.11	Wireless LAN, including Wi-Fi
802.15	Wireless PAN
802.15.1	Bluetooth
802.15.3	High-rate wireless PAN
802.15.4	Low-rate wireless PAN, e.g. Zigbee

Introducing Bluetooth

Bluetooth is a digital radio protocol, meant for PAN applications, and operating in the 2.4 GHz radio band. It provides wireless data links between devices such as mobile phones, wireless audio headsets, computer interface devices like mice and keyboards, and remote sensors.

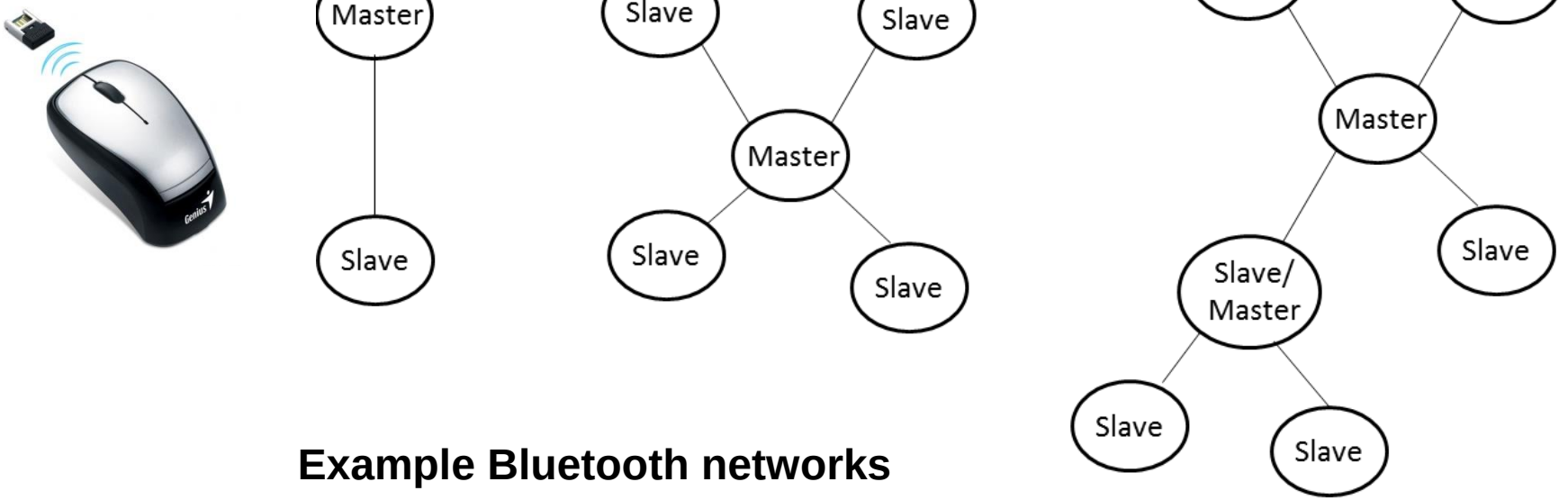
There are three Bluetooth classes, based on output power. Class 2 is the most common. The main characteristics are:

- The approximate communication range is up to 100 m for Class 1 Bluetooth devices, up to 10 m for Class 2 devices, and 1 m for Class 3.
- Bluetooth is relatively low power; devices of Classes 1 to 3 use around 100 mW, 2.5 mW and 1 mW respectively.
- Data rates up to 3 Mbps can be achieved. Recent higher data rate versions are being adopted.
- Up to 8 devices can be simultaneously linked, in a *piconet*. A Bluetooth device can belong to more than one piconet.
- Spread-spectrum frequency hopping is applied, with the transmitter changing frequency in a pseudo-random manner 1600 times per second.

Introducing Bluetooth

When Bluetooth devices detect one another, they determine automatically whether they need to interact. Each device has a unique Media Access Control (MAC) address which communicating devices can recognise and initialise interaction if required. This process follows three phases:

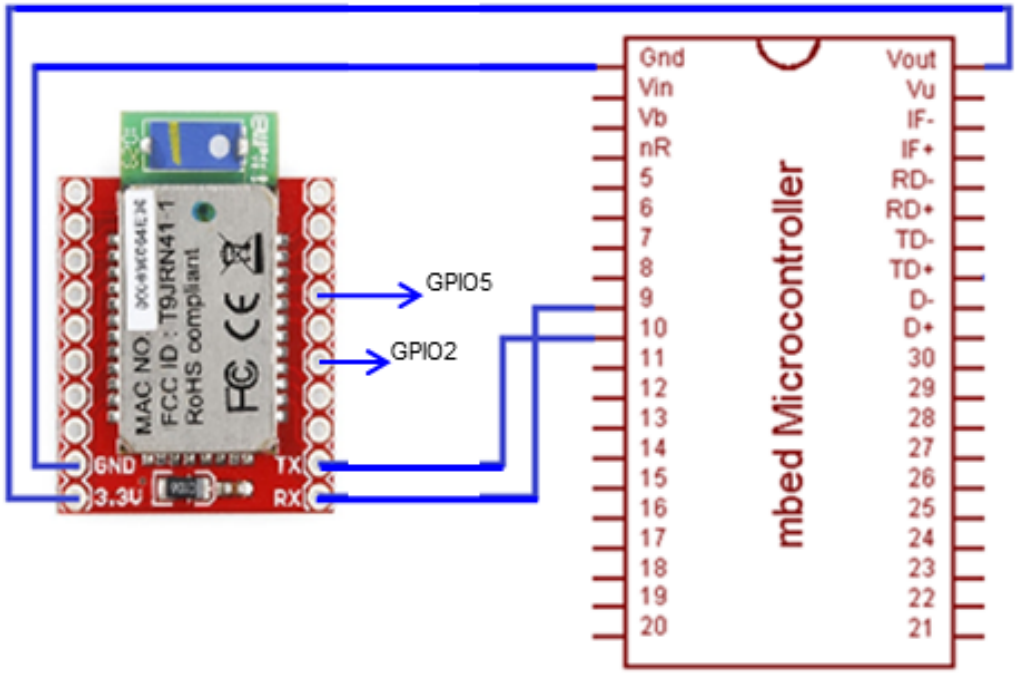
- Discovery - a slave module broadcasts its name and MAC address, seeking for devices to link to.
- Pairing - slave and master exchange identification and authentication data, exploring whether a link should be established.
- Connecting - initiated by the master, through which a link is finally established.



Example Bluetooth networks

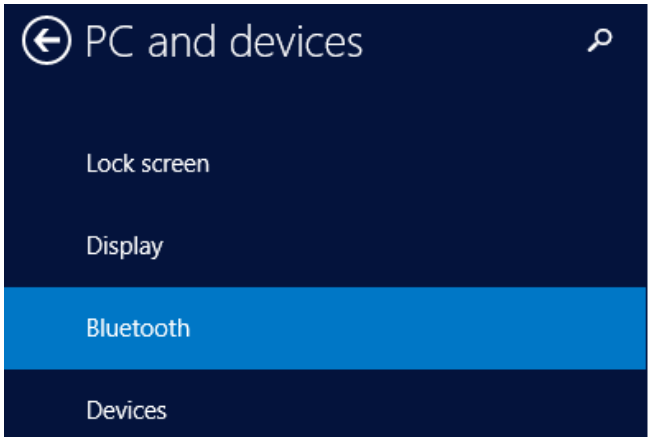
the RN-41 Bluetooth Module

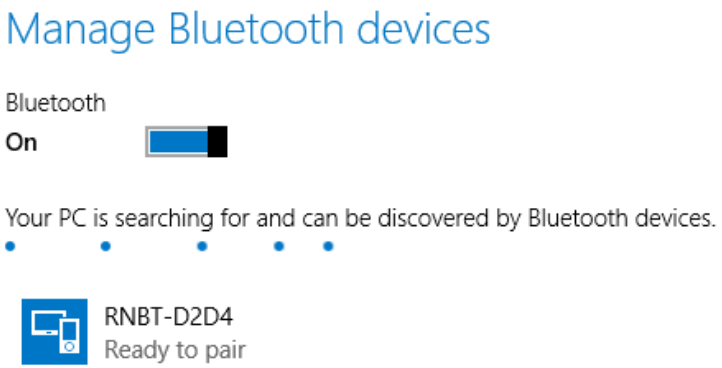
The simplest use of the RN-41 module, a Class 1 Bluetooth device, is to replace a wired serial link with Bluetooth. It defaults to Slave mode, and is immediately ready for this. For example, most laptop computers have Bluetooth capability installed, so it is possible to replace the USB cable to an mbed with a Bluetooth device and allow wireless communication to the laptop.



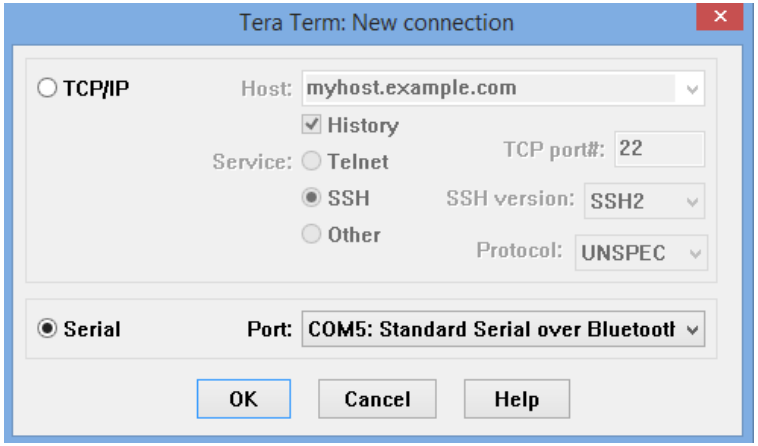
Linking the RN-41 to a PC

A PC with Bluetooth capability can link direct to a powered RN-41 module.

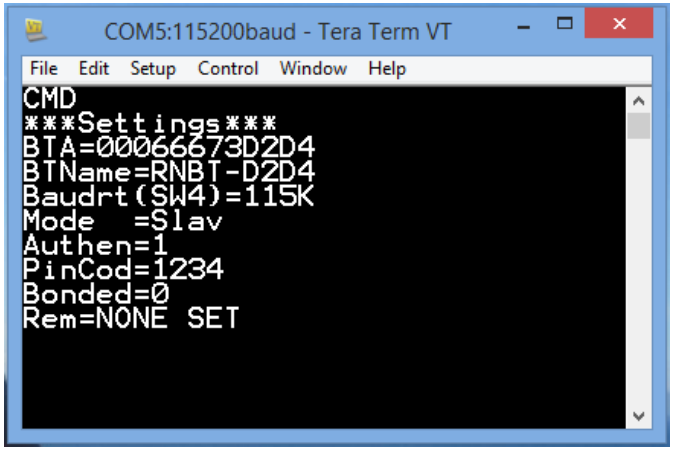




Setting up the PC link to the RN-41 module (display depends on PC and OS type)



Configuring Tera Term for the Bluetooth link



Tera Term display of RN-41 messages

Simple Bluetooth: sending mbed data to a PC

This program applies the mbed/RN-41 hardware shown earlier.

```
/* Program Example 11.1: Bluetooth serial test data program
   Data is transferred from mbed to PC via Bluetooth.       */

#include "mbed.h"
Serial rn41(p9,p10);      //name the serial port rn41
BusOut led(LED4,LED3,LED2,LED1);

int main() {
  rn41.baud(115200);     // set baud for RN41
  while (1) {
    for (char x=0x30;x<=0x39;x++){  // ASCII numerical characters 0-9
      rn41.putc(x);          // send test char data on serial to RN41
      led = x & 0x0F;              // set LEDs to count in binary
      wait(0.5);
    }
  }
}
```

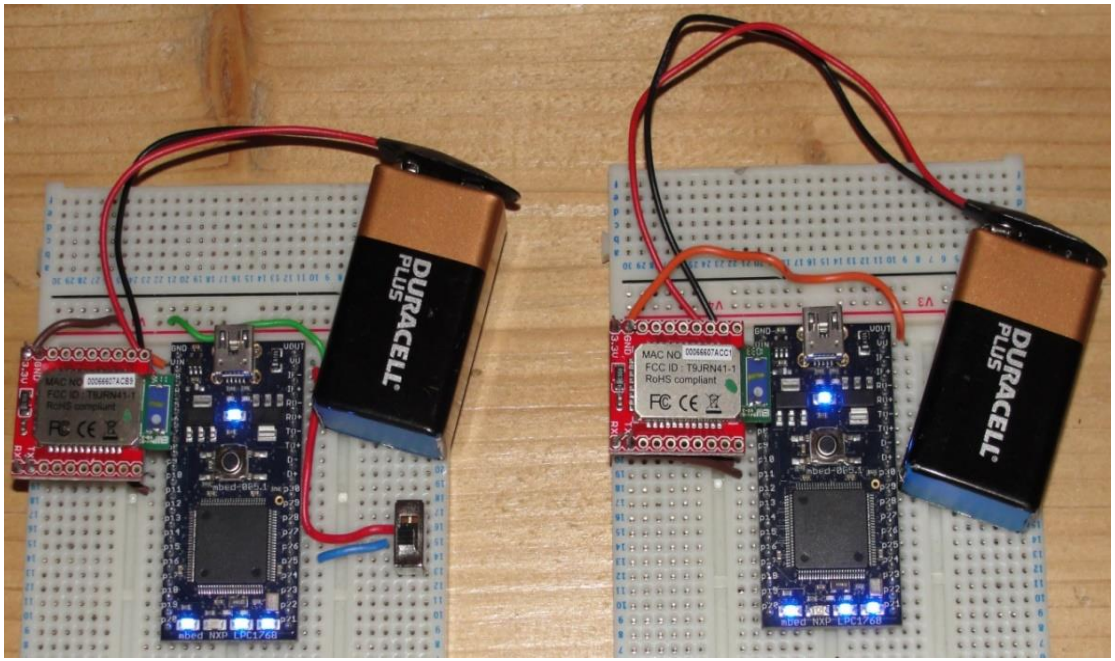
Simple Bluetooth: receiving Bluetooth data from a PC

This program sends a character string from mbed to PC, appearing on Tera Term if enabled. It then receives any data from the host PC keyboard; the remote mbed displays the lower four bits of the received byte on the on-board LEDs.

```
/* Program Example 11.2: Bluetooth serial test data program
   Data is transferred bidirectionally between mbed and PC via
Bluetooth.
                                                              */

#include "mbed.h"
Serial rn41(p9,p10);                    //name the serial port rn41
BusOut led(LED4,LED3,LED2,LED1);

int main() {
  rn41.baud(115200);              // setup baud rate
  rn41.printf("Serial data test: outputs received data to
LEDs\n\r");
  while (1) {
    if (rn41.readable()) {        // if data available
      char x=rn41.getc();         // get data
      led=x;                      // output LSByte to LEDs
    }
  }
}
```

More Advanced Bluetooth: Communicating Between two mbeds

By implementing Bluetooth it is possible to have two or more mbeds communicating wirelessly with each other. This configuration is a little more involved, as one of the RN-41 modules must be set as Master, using the commands available in its *Command mode.* Full details are given in the book.



Evaluating Bluetooth

- Bluetooth is an exciting technology that allows short range wireless communication. This has many valuable applications where wires are intrusive, expensive or difficult to install.
- Recent enhancements to Bluetooth have enabled streaming of high quality audio data and increased range, so the opportunities and applications for Bluetooth are continuously growing.
- An important new version of the protocol is Bluetooth Smart, previously called Bluetooth Low Energy (BLE); this is intended for low power applications. Space does not allow us to cover BLE in the book. It is interesting to note that some mbed enabled boards have this capability, and there is support information on the mbed web site

Introducing Zigbee

While Bluetooth is continuously pressed for ever increasing data rates, there are other applications where high data rates are not the main criterion of interest. Instead, some situations require a combination of low data rates, extreme low power, and continuous operation over months or years.

Zigbee is intended for low power systems, with low data rates. It applies and builds on the IEEE 802.15.4 Low-Rate WPAN standard (Table 11.1). Like Bluetooth it operates in the ISM bands of the radio spectrum.

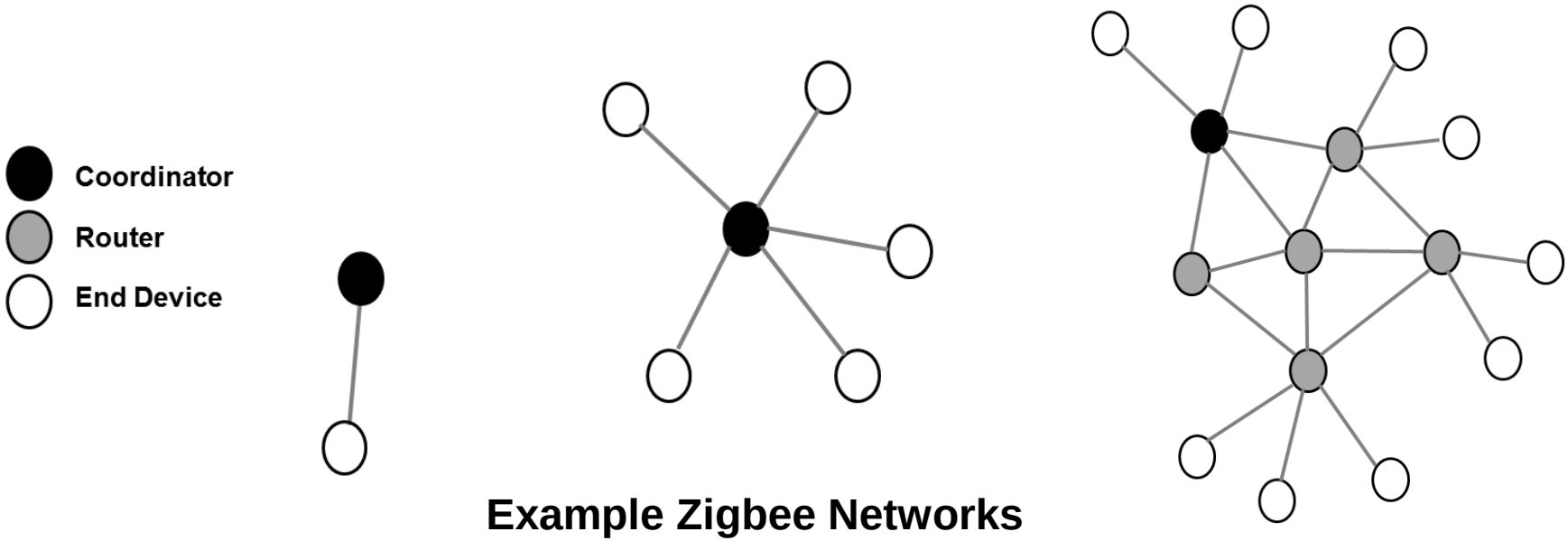
Zigbee has some similarities to Bluetooth, but aims to be simpler, cheaper, with smaller software overhead and with different target applications. Like Bluetooth, Zigbee devices apply spread spectrum communication.

Zigbee Device Types

The end device: the simplest, able to undertake simple measurement actions and pass back the data. It can only do this to its “parent”, i.e. the router or coordinator which allows it to join. It may spend a large part of its time in the Sleep mode.

The router: After joining a Zigbee PAN, it can receive and transmit data, and can allow further routers and end devices to join the network. It can also exercise a useful function (e.g. measurement). It cannot sleep.

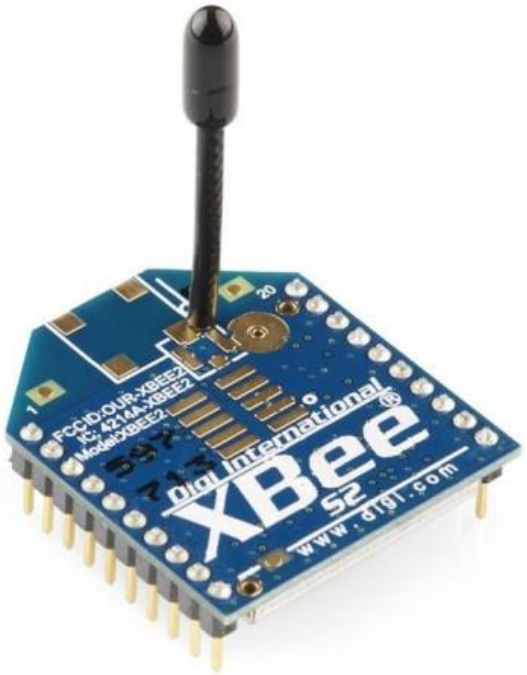
The coordinator: This is the most capable Zigbee device; there can be only one in a network. It launches a network by selecting a PAN ID, and a channel to communicate over. It can allow routers and end devices to join the network. It cannot sleep, so must normally be mains powered.



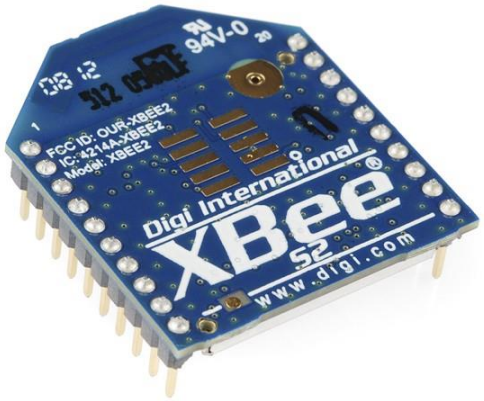
Example Zigbee Networks

Introducing XBee Wireless Modules

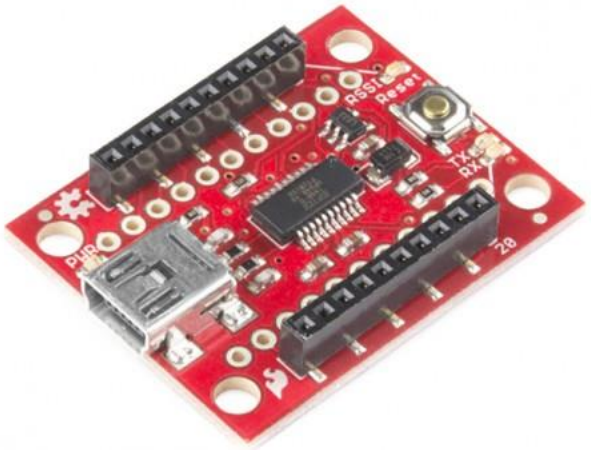
The XBee wireless modules, made by Digi International, can be used to rapidly configure Zigbee networks.



XBee ZB with whip antenna



XBee ZB with pcb antenna

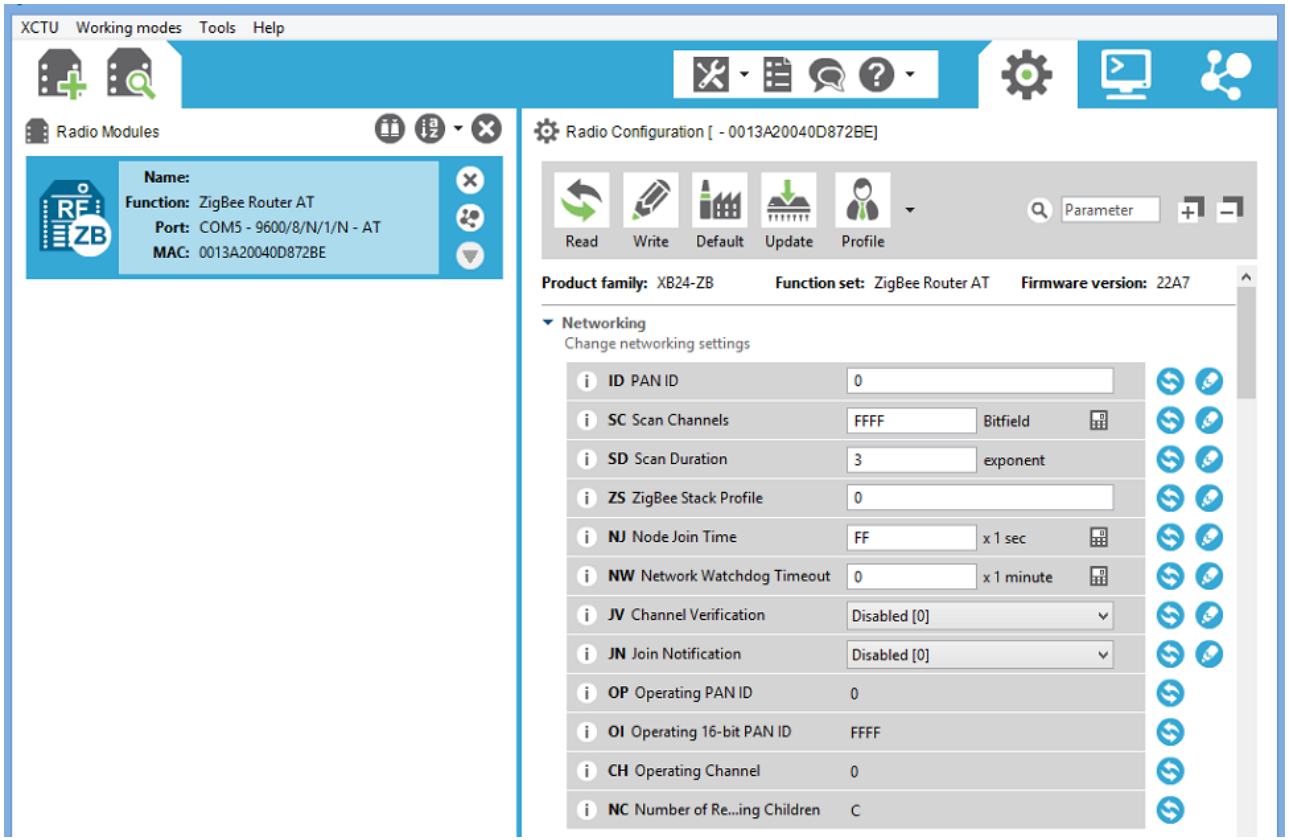


XBee Explorer USB interface

Linking to the XBee from a PC

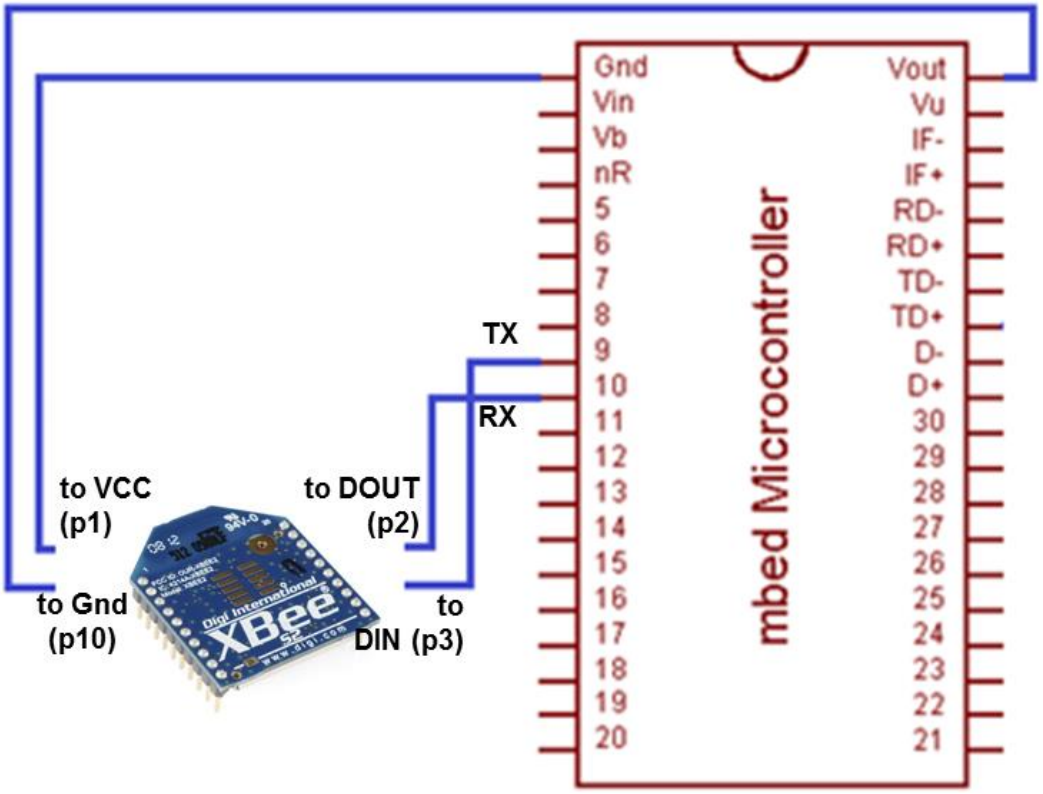
Unlike Bluetooth, PCs don't have Zigbee capability. This can be set up, with two extra pieces of kit. The first is a USB interface such as the Explorer, shown earlier, providing a USB link from PC to XBee. The second is the official interface software from Digi, XCTU, downloaded from the Digi web site. This allows diagnostic testing of an XBee, downloading of new firmware, plus access to remote XBee devices by radio link. Using XCTU, the XBee device can be configured as coordinator, router or end device.

Full details of practical work with Xbee and XCTU can be found in the book.



Implementing a Zigbee PC link to XBee and mbed

Connect a preconfigured (wth XCTU) router XBee in the circuit shown. Compile and download Programme Example 11.5 into the mbed. A coordinator XBee is placed in the USB Explorer, plugged into the host PC. With Tera Term or CoolTerm running, the terminal should continuously count from 0 to 9, with the mbed LEDs counting up in synchronism.

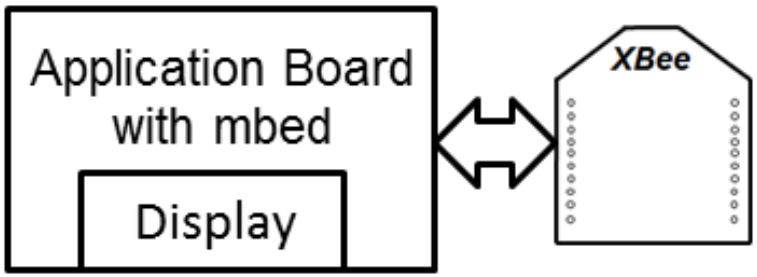


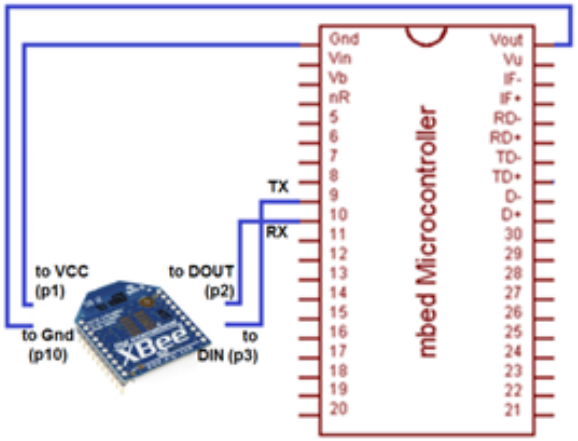
Implementing a Zigbee PC link to XBee and mbed

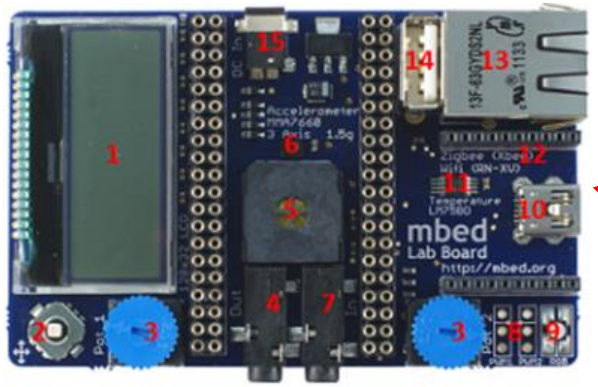
```
/* Program Example 11.5: Zigbee serial test data program
Data is transferred from mbed to PC via Zigbee.
  Requires a set of "paired" XBee modules.              */
#include "mbed.h"
Serial xbee(p9,p10);      //name the serial port xbee
BusOut led(LED4,LED3,LED2,LED1);

int main() {
  xbee.baud(9600);     // set baud rate for xbee
  while (1) {
    for (char x=0x30;x<=0x39;x++){  // ASCII numerical characters 0-9
      xbee.putc(x);          // send test char data on serial to XBee
      led = x & 0x0F;                // set LEDs to count in binary
      wait(0.5);
    }
  }
}
```

Implementing a Zigbee Link mbed to mbed

It is a simple step to set up an XBee to XBee link, with no PC intervention. The mbed app board has a very useful XBee socket. We use this as the coordinator. Full program details are given in the book.





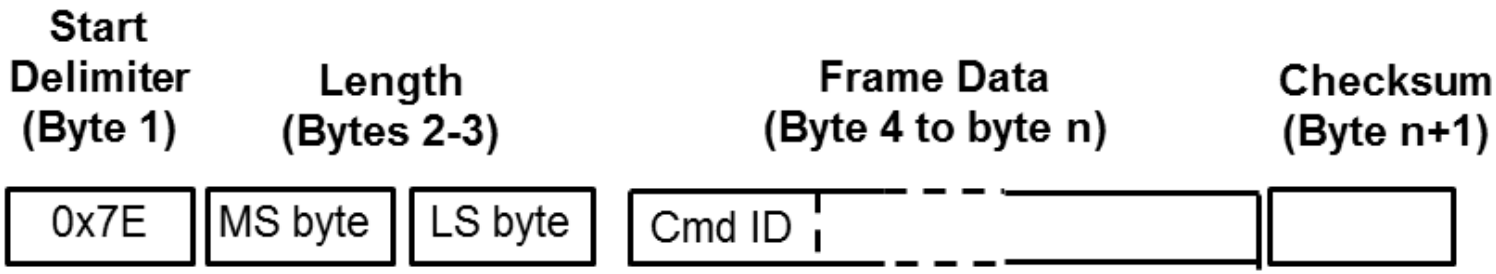


App board XBee socket

Introducing the XBee API

The XBee can operate in AT or API modes. It is the API mode which unlocks the power of Zigbee through the XBee. Using the API, the XBee can change the destination address dynamically, perform error checking, reconfigure remote radios, and exploit the remote XBee I/O capability.

In API mode all data is packaged into *frames,* which carry both the data itself, plus a range of ID, addressing, and error-checking capability.

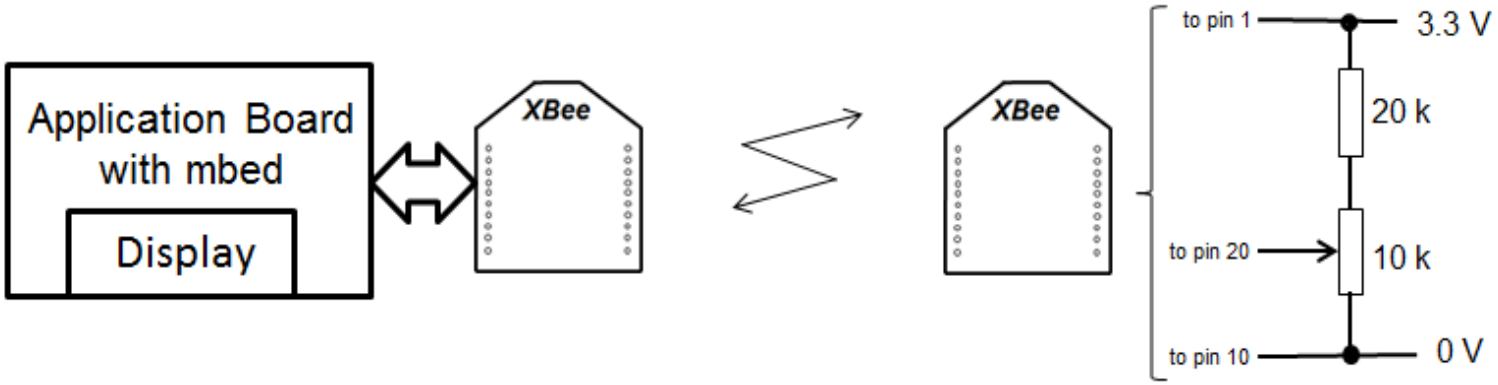


General XBee API data frame structure

Applying the XBee API

In this simple Zigbee link one XBee is set up as a router device, making analog measurements. Its pin 20, labelled AD0/DIO0, is configured as analog input; to play this simple role, it can remain in AT mode. It periodically transmits data values to an XBee coordinator, running in an mbed app board, in API mode.

The XBee ADC is a 10-bit device, and has an input range of 1.2 V. A potentiometer has been chosen to provide this simple analog source, as shown. This has itself been placed in a potential divider, through the addition of the 20 kΩ resistor, to match the ADC input range.



Diagnostic circuit for API trial

Chapter Review

- Wireless links exploit the characteristics of the electromagnetic spectrum, notably in radio, infra-red or visible light.
- A wide range of protocols and technologies exist to implement wireless links across personal, local, neighbourhood and wide area networks.
- Bluetooth is a complex yet effective protocol defined within the IEEE 802 group, which allows Bluetooth-enabled devices to connect and transfer data wirelessly, with potentially high data rates.
- The RN-41 module can be used to give an mbed Bluetooth capability.
- Zigbee is another important protocol defined within the IEEE 802 group, targeted towards low data rate, distributed measurement systems, and extreme low power.
- The XBee module provides Zigbee capability, which can be linked to the mbed. The XBee has its own on-board processing power and input/output capability, so can also readily act as a stand-alone device.