

Easy Bee



Guida alla scelta e alla comprensione dei moduli XBee



Quest'opera è distribuita con licenza Creative Commons
Attribuzione - Non Commerciale - Non opere derivate 2.5 Italia



Autore: Giovanni Bernardo, © 2011

Indice

Introduzione.....	2
1. I moduli XBee : Cosa sono e cosa fanno?.....	3
2. Caratteristiche elettriche e meccaniche.....	5
3. Adattamento dei livelli logici.....	6
4. Il protocollo 802.15.4.....	8
5. Il protocollo ZigBee.....	9
5.1 Differenze tra Zigbee e 802.15.4	10
5.2 Le diverse versioni del protocollo Zigbee: ZNET 2.5 e Zigbee 2007.....	11
5.3 Compatibilità tra ZNET 2.5 e Zigbee 2007.....	11
6. Il protocollo DigiMesh.....	12
7. Serie 1 o Serie 2 ?.....	13
8. Tipi di antenna.....	15
9. Riconoscere il modello di XBee.....	16
10. Elenco modelli di XBee.....	17
11. Configurare gli XBee : XCTU.....	20
11.1 Aggiornamento delle revisioni firmware.....	20
11.2 Collegamento del modulo XBee con XCTU.....	22
11.3 Lettura/Scrittura Firmware e Configurazione.....	24
11.4 Alcuni parametri importanti.....	27
12. Un' ape in modalità API.....	28
13. Modalità comandi - Command mode.....	29
14. Links.....	30
14.1 Documentazione ufficiale DIGI.....	30
14.2 Software.....	30
14.3 Risorse in rete.....	30
14.4 Schede di interfacciamento per XBee.....	30
14.5 Schede correlate.....	30

Introduzione

I moduli XBee hanno un po' cambiato il modo di comunicare via wireless con i dispositivi embedded: il costo accessibile e le numerose caratteristiche messe a disposizione, tra le quali la possibilità di creare reti, ne fanno moduli davvero versatili.

Tuttavia l'esistenza di numerosi tipi di moduli XBee può rendere difficile la scelta e causa spesso alcune difficoltà nella messa in opera, sebbene alcuni determinati modelli siano stati realizzati proprio per essere utilizzati da subito in determinate applicazioni, senza necessità di riconfigurazione.

Questo documento non vuole assolutamente essere un manuale di utilizzo degli XBee ma è stato piuttosto concepito per aiutare a capire quali sono le differenze tra i vari moduli esistenti in commercio e, quindi, aiutare nella scelta del modulo più adatto alle proprie esigenze, senza tralasciare alcuni aspetti pratici importanti per cominciare con il piede giusto ed evitare da subito frustrazioni e perdite di tempo.

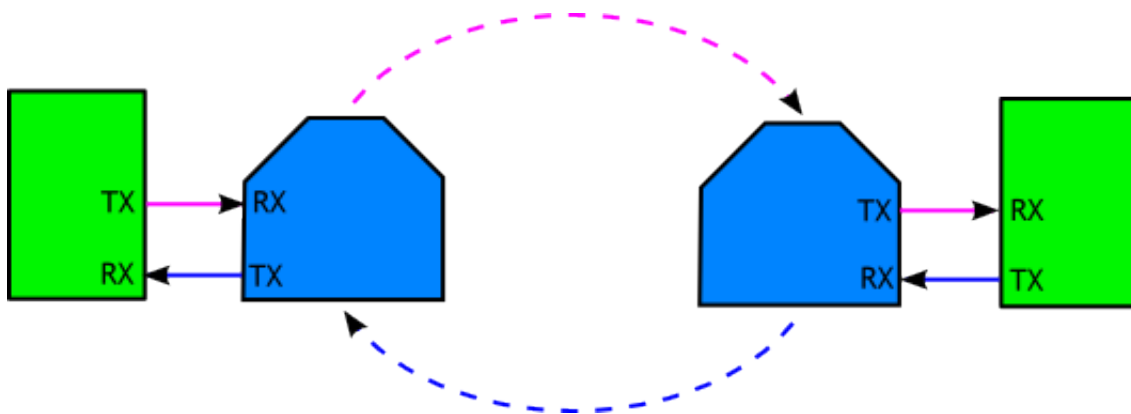
In molti, anche i più esperti, si trovano difatti spiazzati di fronte alla possibilità di scegliere tra moduli Serie 1, Serie 2, Serie 2B, ZNET 2.5, Zigbee, Digimesh, 802.15.4 ecc senza contare che la documentazione presente sul sito del produttore non sempre è facile da trovare e non sempre è di facile consultazione.

Nonostante abbia fatto tutto il possibile per realizzare questa guida in maniera accurata (contattando anche più volte la DIGI per chiedere delucidazioni), e semplice da leggere anche dai meno esperti, non escludo che vi possano essere alcuni errori. Per tale motivo consiglio di consultare sempre il manuale del modulo che avete tra le mani: a fine guida, spero, almeno sarete capaci di capire qual'è il manuale giusto per il modulo che avete tra le mani !

1. I moduli XBee : Cosa sono e cosa fanno?

Gli XBee sono essenzialmente dei moduli, prodotti dalla DIGI (www.digi.com - ex *Maxstream*) che consentono il trasferimento dati via RF (Radio Frequenza) sfruttando una normale comunicazione seriale. Questo vuol dire che per riuscire a trasferire dei dati, senza fili, è necessario soltanto saper utilizzare la periferica UART inclusa sul nostro microcontrollore preferito o la porta seriale del PC: è possibile difatti immaginare, nella loro configurazione più semplice, una coppia di XBee come un sostituto di un cavo seriale.

In breve: tutto ciò che il microcontrollore (o il computer o altro dispositivo in grado di effettuare una comunicazione seriale asincrona) invia sul pin RX di un modulo XBee, giunge ad un altro XBee sul pin TX e può quindi essere ricevuto dal dispositivo remoto:



Questo particolare modo di operare degli XBee viene anche detto *Modalità Trasparente* (*Transparent Mode*) e non è, come vedremo, l'unica modalità operativa degli XBee, tuttavia è forse quella più semplice ed utilizzata.

Gli XBee più diffusi operano alla frequenza di 2.4GHz, ma esistono anche XBee operanti a 900MHz e ad 868MHz ma non tutti sono utilizzabili in Italia come vedremo a breve.

Nota: Sebbene anche il WiFi operi a 2.4GHz, i due tipi di comunicazione non interferiscono tra loro in quanto utilizzano protocolli differenti.

Alcuni utenti, quando si trovano nella necessità di dover realizzare un'applicazione che comunichi senza fili, si trovano a dover fare una scelta tra i vari moduli wireless esistenti in commercio e spesso si fanno dei confronti con moduli RF a basso costo che operano sui 433MHz. Tali moduli costano sicuramente meno e sono utilizzabili per semplici applicazioni, le differenze tuttavia sono notevoli e alcune caratteristiche spesso vengono trascurate o sottovalutate:

- Un modulo XBee è bidirezionale : un singolo modulo è in grado sia di ricevere che di trasmettere. Molti moduli economici possono operare in una sola direzione.
- Un modulo XBee può essere indirizzato in maniera univoca. Ogni XBee ha difatti un numero seriale che lo distingue da tutti gli altri XBee. Tramite una semplice utility (XCTU, che [analizzeremo poi](#)) è difatti possibile fare in modo che due moduli comunichino soltanto tra di loro senza fare salti mortali nei nostri software. Questo consente di avere molti moduli XBee in una stanza senza che interferiscano tra loro.
- Nei moduli XBee sono già implementati nell'hardware tutti i controlli tipici di una comunicazione wireless, come ad esempio i controlli sugli errori di trasmissione, che garantiscono una comunicazione affidabile. Questo, sui moduli economici, siamo costretti a realizzarlo noi via software. Il modulo XBee esegue tutti questi controlli in maniera automatica senza l'intervento dell'utente e possiamo fare in modo che ritrasmetta i dati in automatico se questi non sono giunti a destinazione in maniera corretta!
- Sugli XBee è anche possibile assegnare diversi canali ai moduli: avendo più XBee in un'area si può assegnare ad alcuni moduli un canale e ad altri un canale differente: questo minimizza ancor più eventuali interferenze.

A tutto questo non abbiamo aggiunto che un modulo XBee ha anche a disposizione alcune periferiche come GPIO (porte di Input/Output programmabili), PWM e convertitori Analogico/Digitali che li rendono dispositivi adatti, tra le altre cose, a creare reti con nodi autonomi dotati di sensori senza ricorrere a collegamenti con un microcontrollore.

2. Caratteristiche elettriche e meccaniche

E' molto facile riconoscere un modulo XBee: la loro forma è particolare ed è stata copiata da molti produttori nella realizzazione di altri moduli (sia RF che non, come moduli Ethernet ad esempio), soprattutto nei confronti della piedinatura: è difatti possibile trovare in commercio molti moduli "Bee-like" che hanno una piedinatura compatibile con gli XBee, per cui tali moduli spesso possono essere tranquillamente utilizzati sulle schede di adattamento realizzate per gli XBee: in particolare i pin di alimentazione e i pin di comunicazione (TX/RX) si trovano nella stessa posizione.

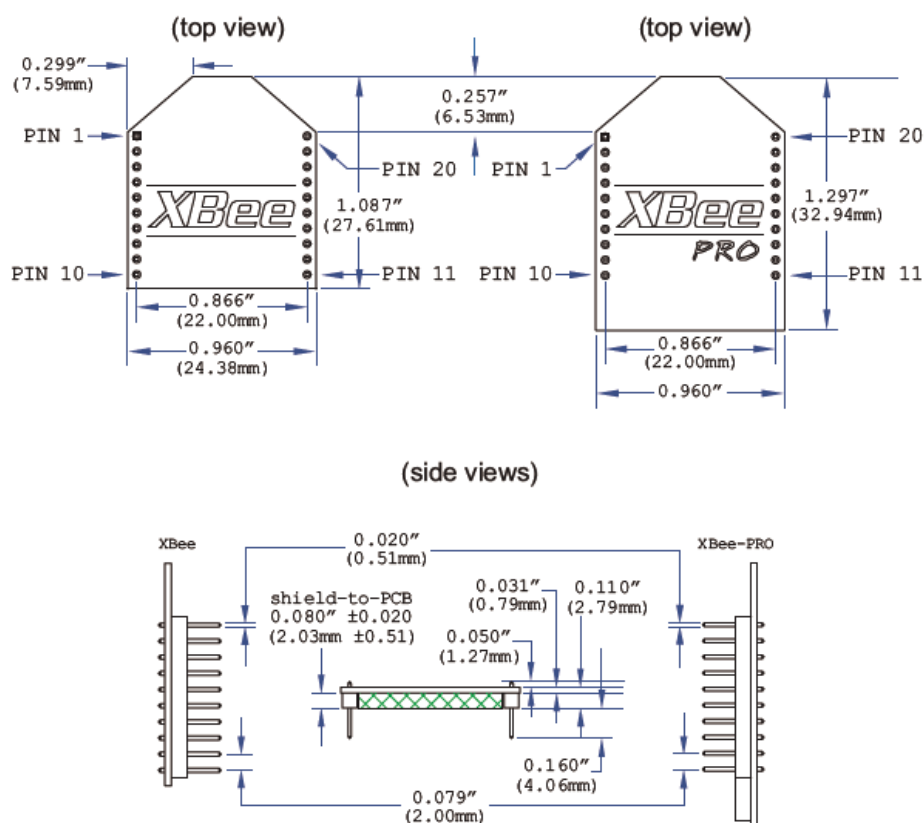
I moduli XBee possiedono una doppia fila di piedini con una spaziatura (pitch) di 2mm, quindi leggermente più stretta di quella "classica" da 2.54mm a cui molti utenti sono abituati con millefori e breadboard. Generalmente gli XBee vengono utilizzati su schede di adattamento che prevedono anche una regolazione di tensione (gli XBee funzionano a 3.3V - Fate riferimento all' [ultimo paragrafo](#) per un elenco di schede di adattamento semplici da utilizzare).

Prima di addentrarci in discussioni di carattere più esteso, incominciamo a fare una prima distinzione di base tra i vari moduli: gli XBee appartenenti alle varie famiglie si presentano quasi sempre in due varianti: il modello *Low Power* e il modello *PRO*.

I modelli Low Power, come lascia intuire la loro stessa denominazione, offrono consumi ridotti, distanza di comunicazione inferiore a fronte di un costo decisamente più basso. I modelli PRO sono anche identificati con la dicitura *Extended Range* perchè ricoprono una distanza superiore rispetto al corrispettivo Low Power, per contro hanno ovviamente un consumo superiore e anche un costo più elevato.

Generalmente la variante Low Power non ha nessuna denominazione che lasci intuire si tratti di un modello a bassa potenza mentre i modelli Extended Range sono chiaramente indicati come PRO, per cui: se nel nome del modulo manca la denominazione PRO, allora si tratta sicuramente di un modello Low Power.

Esternamente i moduli PRO si presentano leggermente più lunghi, oltre ad avere la dicitura "PRO" chiaramente stampata sulla parte superiore del corpo:



I moduli Low Power esistono in una sola variante: sono uguali in tutte le parti del mondo. I moduli PRO invece possono esistere in due varianti a seconda del paese a cui sono destinati a causa delle diverse legislazioni relative alle trasmissioni RF: negli Stati Uniti sono difatti concesse potenze di trasmissione superiori a quelle previste in altre parti del mondo, per cui i modelli PRO destinati all'Europa e ad altre parti del mondo

hanno la lettera J nella parte finale della sigla (tali modelli sono anche identificati come variante *International*, alcuni siti di vendita online indicano queste versioni come *versione Japan ...* per via della J).

In ogni caso, col beneficio del dubbio, dovrebbe essere possibile utilizzare anche in Europa le versioni non-J purchè si riduca la potenza di trasmissione agendo sulla configurazione.

3. Adattamento dei livelli logici

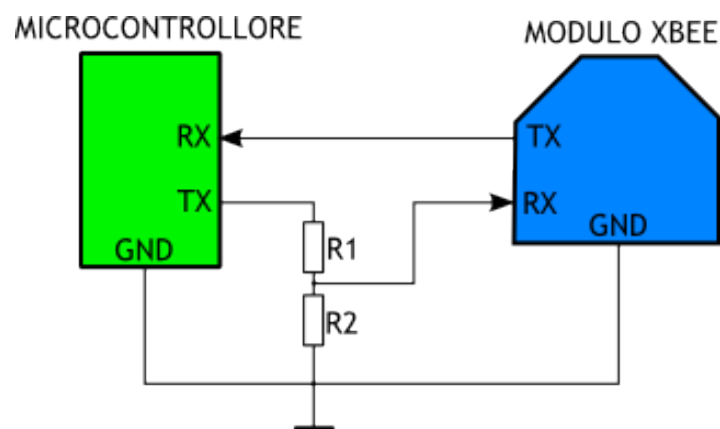
Abbiamo detto poco fa che i moduli XBee funzionano a 3.3V. Questo in apparenza può costituire un problema nel momento in cui si debbano interfacciare tali moduli con microcontrollori operanti a 5V; in realtà bastano semplicemente due resistenze.

Sulla linea TX dell'XBee (che va quindi collegata alla linea RX del nostro dispositivo), non è necessario eseguire nessun adattamento di tensione: le logiche a 5V riconoscono 3.3V come livello alto senza problemi.

Nota: Le logiche TTL a 5V hanno difatti una soglia indeterminata nel range 2 ÷ 1.5V. Le tensioni in questo range potrebbero essere riconosciute sia come livello alto che come livello basso, mentre al di sopra dei 2V vengono sicuramente riconosciute come livello alto.

Al contrario, la linea RX dell'XBee non può assolutamente ricevere una tensione di 5V altrimenti viene danneggiata (soltanto alcuni modelli molto particolari di XBee hanno i pin 5V-tolerant: questi sono gli XBee XSC a 900MHz -Serie 3- che tra l'altro sono approvati soltanto nel Nord America).

In questo caso basta un semplice **partitore di tensione** per fare in modo che i 5V (che identificano il livello logico alto) forniti dal nostro dispositivo sulla linea TX, giungano all'XBee come 3.3V senza causare danni:



Possiamo ad esempio utilizzare R1=12KΩ e R2=22KΩ oppure R1=10KΩ e R2=18KΩ.

L'utilizzo di un partitore di tensione è sicuramente la soluzione più semplice ed efficace. Ad ogni modo induce un certo *slew rate* ma questo non è un problema con le velocità di comunicazione massime consentite dagli XBee. Potrebbero però aversi errori di trasmissione solo nei casi in cui la lunghezza dei collegamenti tra dispositivo ed XBee fosse eccessiva, questo generalmente non accade mai.

Esistono comunque altri sistemi più o meno complessi per adattare i livelli logici, fate riferimento all'[ultimo paragrafo](#) in cui è presentato un link in cui si parla appunto del **level shifting** sugli xbee.

Sulle schede di adattamento che prevedono un collegamento su USB o su RS232, l'opzione del level shifting è obbligatoriamente di serie. Sulle schede di adattamento generiche che prevedono unicamente il regolatore di tensione, generalmente il partitore non viene inserito perchè si potrebbe scegliere di utilizzare il modulo XBee con logiche a 3.3V, che tra l'altro sono sempre più diffuse e destinate a soppiantare quelle operanti a 5V.

4. Il protocollo 802.15.4

Prima di capire la differenza tra le varie famiglie di XBee è necessario comprendere in che modo i moduli XBee implementano i diversi protocolli di comunicazione, perchè è proprio in base a tali caratteristiche che i moduli XBee vengono suddivisi.

Partiamo quindi con l'identificare il protocollo di comunicazione di partenza: **802.15.4**.

L' 802.15.4 è uno standard per le comunicazioni wireless definito dall' **IEEE** (www.ieee.org). L'IEEE è un'istituto di ingegneria che si occupa appunto di definire gli standard di comunicazione: sappiamo bene che la presenza di uno standard in un'apparato di comunicazione garantisce l'**interoperabilità** anche tra moduli realizzati da diversi produttori.

Lo standard 802.15.4 è stato concepito essenzialmente per soddisfare requisiti di basso consumo, ovvero:

- Velocità di trasferimento basse
- Connettività semplice
- Possibilità di operare con batterie

Le frequenze definite da questo standard sono tutte **frequenze ISM** (Industrial, Scientific and Medical): 868-868.8 MHz, 902-928 MHz e 2.400-2.4835 GHz.

Le frequenze sui 2.4GHz sono approvate in tutto il mondo : ecco perchè i moduli operanti in tale banda sono quelli aventi una diffusione maggiore.

La banda degli 868MHz è approvata soltanto in Europa e la banda 902-928MHz fin'ora può essere utilizzata soltanto negli USA, Canada ed alcuni stati minori che accettano i regolamenti dell'FCC (*Federal Communication Commission* - www.fcc.gov)

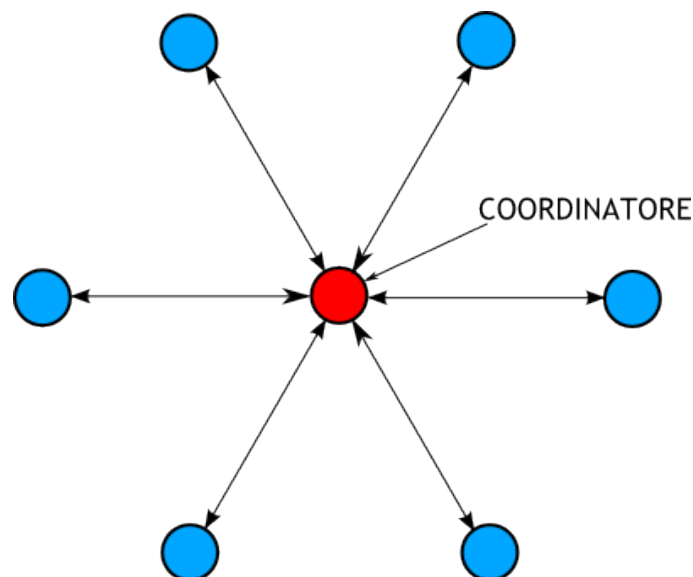
Riguardo alla banda di frequenza 2.400-2.4835 GHz, lo standard IEEE specifica che la comunicazione deve avvenire su canali distanziati tra loro a passi di 5MHz a partire da 2.405GHz fino ad arrivare a 2.480GHz e con una velocità di trasferimento massima di 250KBPs in aria.

A causa dell'**overhead** (ovvero quella parte di banda necessaria per trasmettere dati aggiuntivi richiesti dal protocollo, come il controllo di errore ecc.) la velocità a disposizione dell'utente per il trasferimento dei dati è generalmente di **115KBPs** massimo.

Lo standard 802.15.4 prevede una comunicazione **point-to-point** o **point-to-multipoint**. Una topologia di rete point-to-point prevede che due dispositivi A e B comunichino unicamente tra loro.

Nelle applicazioni point-to-multipoint è invece richiesto un **coordinatore** (un nodo centrale) che può ricevere e trasmettere da/verso tutti e si occupa in qualche modo di "avviare" la rete, mentre i nodi periferici (**end devices**) possono soltanto comunicare con il coordinatore.

Questa *topologia di rete* è anche chiamata "a stella" :



I moduli XBee identificati con la dicitura **XBee 802.15.4 OEM RF modules**, conosciuti formalmente con la denominazione **Serie 1**, sono in grado di eseguire comunicazioni point-to-point e point-to-multipoint e sono quindi conformi allo standard 802.15.4. In aggiunta tali moduli possono anche eseguire comunicazioni di tipo **peer-to-peer**

(modalità, quest'ultima, non prevista dalla specifica IEEE 802.15.4). Nella comunicazione peer to peer non c'è bisogno di un coordinatore e ogni nodo può comunicare con tutti i nodi vicini. Di default i moduli XBee Serie1 sono configurati per operare in modalità peer-to-peer *trasparente*. Cerchiamo ora di capire quali sono le differenze tra il protocollo 802.15.4 e il protocollo Zigbee.

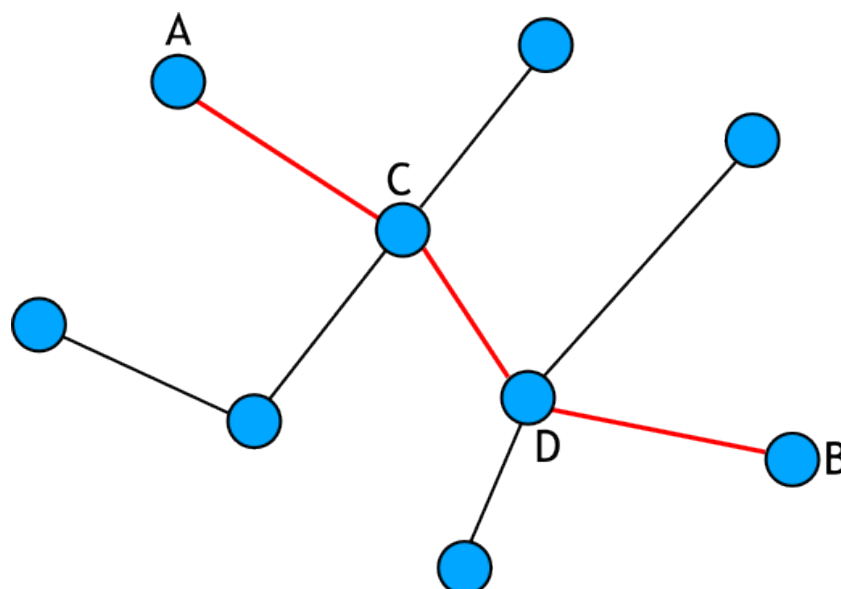
5. Il protocollo ZigBee

Il protocollo ZigBee è stato definito dalla **ZigBee alliance** (www.zigbee.org), il quale raggruppa diverse compagnie che hanno lavorato insieme per definire un protocollo semplice da utilizzare, a basso consumo e che possa soddisfare varie esigenze sia commerciali che industriali. Per tale motivo la ZigBee alliance ha scelto come base di partenza lo standard 802.15.4.

5.1 Differenze tra Zigbee e 802.15.4

A differenza dello standard 802.15.4, il protocollo ZigBee prevede un collegamento di tipo **mesh**. Un collegamento mesh è utile quando due moduli devono comunicare tra di loro ma la distanza che li separa è eccessiva: in questo caso altri moduli presenti nella rete, operanti come **router**, possono effettuare il *rilancio* del messaggio.

Si prenda ad esempio la figura seguente che simula la disposizione di alcuni moduli in una rete mesh:



Il nodo A (che nella fattispecie è un end device) ha bisogno di comunicare con il nodo B (altro end device), ma la distanza che li separa è eccessiva. In questo caso il messaggio può quindi essere inviato al router C, il quale lo invia a D che esegue infine il

trasferimento a B. Questo tipo di rete ha bisogno di un coordinatore per poter funzionare. Il coordinatore mette in opera la rete dopodichè agisce come un normale router, ciononostante nella rete possono essere presenti altri dispositivi operanti come semplici router e sicuramente ci saranno degli end devices.

Il protocollo ZigBee è strutturato in maniera tale che il trasferimento da A a B attraverso i router intermedi avviene in automatico senza interventi esterni: i nodi costituenti la rete “capiscono” l’esigenza e si organizzano affinché il messaggio possa giungere a destinazione. Se per caso un nodo viene rimosso, gli altri moduli si accorgono dell’evento e riorganizzano la rete sempre in maniera autonoma.

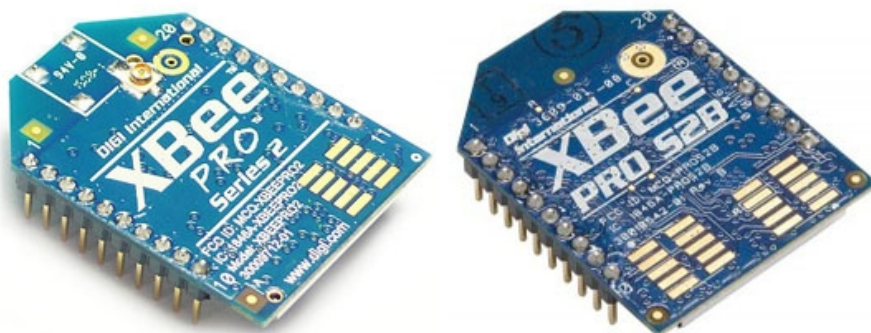
Il protocollo ZigBee inoltre si prende cura anche di fare in modo che il messaggio venga inviato nuovamente se c’è stato qualche errore di trasmissione (quest’ultima non è comunque una caratteristica esclusiva del protocollo Zigbee).

5.2 Le diverse versioni del protocollo Zigbee: ZNET 2.5 e Zigbee 2007

I moduli Xbee identificati con la dicitura **Xbee ZB Zigbee RF Modules**, anche noti come **Serie 2** o semplicemente **XBee ZB**, implementano il protocollo ZigBee di *ultima generazione*. Ho specificato di ultima generazione, perchè?

Alcuni modelli serie 2 vengono difatti indicati con la denominazione **XBee ZNET 2.5** : questa dicitura identifica in realtà una vecchia versione del firmware. I moduli Serie 2 ZNET 2.5 implementano lo stack *EmberZNet 2.5 Zigbee*. Questi modelli sono identificati dalla sigla iniziante per **XB24-B** (o **XBP24-B** per la variante PRO). I nuovi moduli Serie 2 ZB implementano invece lo stack *EmberZNet PRO 3.1 Zigbee* anche noto come **Zigbee 2007**.

I modelli PRO (solo quelli PRO!) che usano lo stack Zigbee 2007 esistono in due versioni hardware differenti: abbiamo i modelli con hardware Serie 2, il cui hardware è uguale a quello dei moduli XBee ZNET 2.5, e identificati dalla sigla iniziante per **XBP24-Z7** (il Z7 sta appunto per Zigbee 2007) e i modelli più recenti, con hardware **Serie 2B** (indicati con **S2B**) :



I modelli con hardware **Serie 2B** si distinguono per avere una doppia fila di pad sulla parte superiore e sono identificati dalla sigla iniziante per **XBP24BZ7** (senza trattini, con

la B che identifica appunto la serie 2B, seguita da Z7). I modelli PRO Serie 2B esistono anche in versione programmabile: hanno un processore aggiuntivo della Freescale che può essere programmato per contenere applicazioni definite dall'utente. I modelli con l'hardware S2B, inoltre, sono destinati a soppiantare i modelli Serie 2 PRO.

5.3 Compatibilità tra ZNET 2.5 e Zigbee 2007

I due stack non sono compatibili tra loro, per cui un modulo Serie 2 ZNET 2.5 non può comunicare con un modulo Serie 2 ZB.

Ad ogni modo abbiamo detto che i moduli ZNET2.5 e i moduli Serie 2 (non S2B) hanno lo stesso hardware ed è quindi possibile aggiornare un modulo ZNET 2.5 ad un modulo ZB.

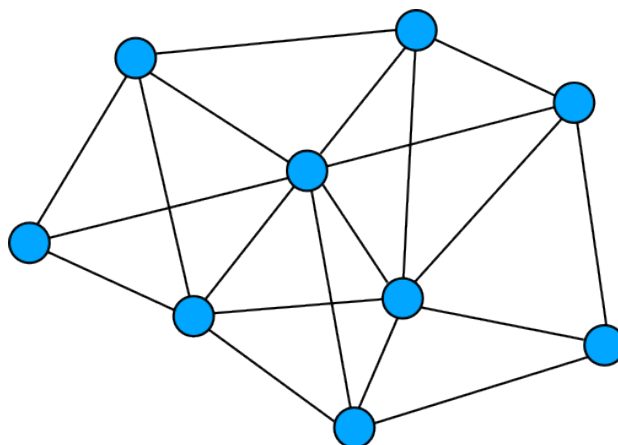
La DIGI ha preparato alcuni files (*ZNET 2.5 to ZB conversion kit*) che permettono di eseguire questo aggiornamento utilizzando il software [XCTU](#). In realtà non ci sarebbe nemmeno bisogno di scaricarli avendo a disposizione gli ultimi aggiornamenti: vedi [paragrafo 11](#) per una descrizione di XCTU e l' [ultimo paragrafo](#) per un link al download di tali files.

Per concludere: i modelli di XBee con la dicitura ZB nella sigla sono in grado di comunicare unicamente con dispositivi ZB, per cui un modulo Serie 2 non può comunicare con un modulo Serie 1 e viceversa, ma non può nemmeno comunicare con un modulo Serie 2 ZNET 2.5 ! Abbiamo però capito che quest'ultimo modulo è possibile aggiornarlo ad un Serie 2 ZB.

6. Il protocollo DigiMesh

A complicare ulteriormente le cose, esistono anche i moduli identificati come **Serie 1 DigiMesh**. Si tratta in realtà sempre di moduli Serie 1 ma forniti con un firmware differente che permette di implementare la topologia di rete DigiMesh.

Il protocollo DigiMesh è proprietario della DIGI, consente di creare delle reti Mesh di tipo Peer to Peer (ovvero senza coordinatori):



Nella topologia Digimesh ogni nodo può comunicare con tutti gli altri e di conseguenza non c'è bisogno di dispositivi che fungono da coordinatore.

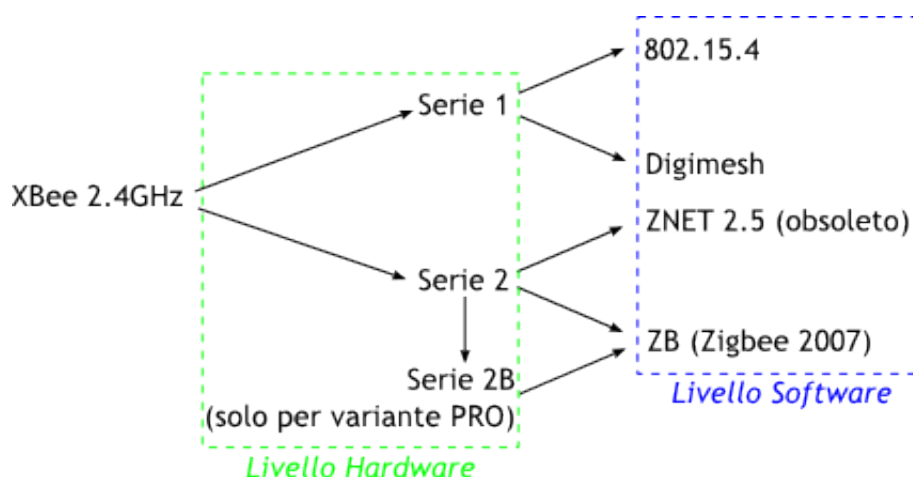
Gli XBee Serie 1 Digimesh sono quindi in tutto e per tutto XBee Serie 1: abbiamo detto che si tratta degli stessi moduli (stesso hardware) ma venduti con un firmware differente. E' difatti possibile fare in modo che un modulo Serie 1 Digimesh diventi un modulo Serie 1 802.15.4 e viceversa eseguendo un aggiornamento del firmware tramite XCTU.

C'è un però: sebbene gli XBee Serie 1 Digimesh appartengano comunque alla Serie 1, non tutti i modelli Digimesh sono compatibili con la Serie 1 802.15.4 e viceversa. Sul sito della DIGI è spiegato come verificare tale compatibilità: vedi [ultimo paragrafo](#) in cui è presente un link al sito della DIGI che spiega come fare a capire se è possibile eseguire questo aggiornamento.

7. Serie 1 o Serie 2 ?

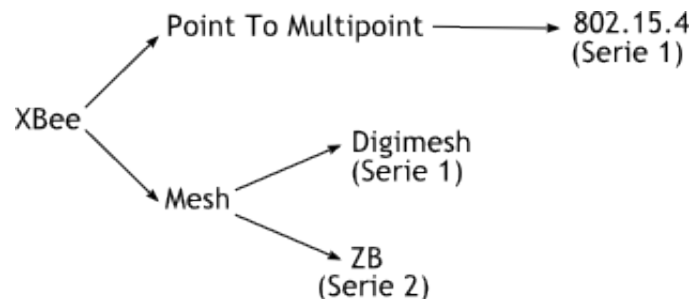
Abbiamo visto che gli XBee vengono comunemente identificati come “Serie1” o “Serie2”. Questa denominazione, in realtà, abbiamo capito che non è indicativa delle varie famiglie di XBee ma è piuttosto un riferimento all'*hardware* montato a bordo che permette loro di funzionare in un modo piuttosto che in un altro.

Faccio quindi qui una mia personale distinzione, basata sull'Hardware, sapendo che una stessa serie può esistere con differenti denominazioni dovute al firmware precaricato di serie: abbiamo capito che gli XBee Serie 1 esistono sia Digimesh che 802.15.4, gli XBee serie 2 esistono sia in versione ZNET 2.5 (obsoleta) che in versione ZB (e i nuovi moduli ZB PRO hanno invece l'hardware S2B):



La DIGI, invece, fa una distinzione basata sul protocollo di comunicazione piuttosto che sull'hardware: ha difatti suddiviso gli XBee in 2 grandi famiglie: **Point-To-MultiPoint** (alla quale appartengono i modelli Serie1) e **Mesh**. Ed è importante capire questa distinzione almeno per riuscire a districarsi nel loro sito.

La famiglia Mesh viene quindi ulteriormente suddivisa in due sottofamiglie: ZigBee (modelli Serie2 ZB, Serie 2B ZB e ZNET2.5 ormai in disuso e non più elencati sul sito) e Digimesh (modelli Serie1 con firmware digimesh):



Nei casi in cui si desidera realizzare una comunicazione point-to-point (ovvero: devono comunicare tra loro soltanto due dispositivi) è sicuramente conveniente scegliere i moduli appartenenti alla famiglia XBee 802.15.4 OEM RF modules (ovvero la Serie 1). Questa scelta è consigliata anche dalla stessa DIGI. Perché?

Questi moduli di default sono già impostati per la comunicazione in modalità trasparente (transparent mode): tolti dalla scatola possono essere messi subito in funzione senza eseguire complicati settaggi. In parole povere, i dati che vengono inviati sul pin RX di un modulo (cioè inviati dal TX del microcontrollore collegato al modulo), arrivano tal quali al pin TX di un altro modulo (e quindi sull'RX del dispositivo ad esso collegato), comportandosi di fatto come una prolunga seriale wireless proprio come abbiamo visto all'inizio di questo documento.

Per una comunicazione punto punto possono andar bene anche gli XBee serie 2 ma sono più difficili da utilizzare in quanto andrebbero opportunamente configurati dato che di serie sono impostati per lavorare in una rete Mesh secondo il protocollo Zigbee. Vedi [ultimo paragrafo](#) per un link con maggiori informazioni su come impostare i moduli Serie2 per una semplice comunicazione point-to-point.

Generalmente si sceglie un modulo Serie 2 soltanto nei casi in cui si desidera implementare una topologia di rete mesh secondo il protocollo Zigbee o si sceglie un modulo Serie 1 Digimesh qualora si desidera implementare una rete mesh senza coordinatori.

Molti credono, erroneamente, che la Serie 2 sia la sostituta della Serie 1 e che quindi i moduli della Serie 2 siano più potenti e/o più performanti: ciò è sbagliato. Semplicemente: i modelli Serie 1 permettono alcune topologie di rete e i modelli Serie 2 permettono la comunicazione secondo lo standard ZigBee. Si sceglie la Serie 1 o la 2 in base a ciò che bisogna realizzare.

Nota: Datasheet alla mano, i moduli Serie2 in realtà sono davvero leggermente più potenti di quelli della Serie1, ma se bisogna realizzare unicamente una comunicazione point-to-point, il costo più elevato dei moduli Serie 2 e la necessità di riconfigurazione, operazione tra l'altro non proprio intuitiva, non giustificano il piccolo aumento di potenza.

Quindi per applicazioni semplici in cui devono comunicare unicamente due dispositivi (es.: un robot con un sistema di controllo remoto) la soluzione più immediata, semplice ed economica è sicuramente un XBee Serie 1.

Ricordo ancora una volta che un modello Serie 1 (sia Digimesh che 802.15.4) non può comunicare con un modello Serie 2. I modelli Serie 1 Digimesh non sono tutti compatibili con la Serie 1 802.15.4. Un modello PRO di una serie può sempre comunicare col modello non-PRO (Low Power) appartenente alla stessa serie (Serie 1 con Serie 1, Serie 2 con Serie 2).

La variante PRO del modulo che abbiamo deciso di utilizzare si sceglie invece nei casi in cui si ha bisogno di una potenza di trasmissione (e quindi una distanza da coprire) maggiore. La potenza di trasmissione inoltre dipende anche dal tipo di antenna che il modulo monta, come vedremo tra poco. La variante PRO, inoltre, potrebbe anche avere altre caratteristiche leggermente differenti rispetto al corrispettivo Low Power (ad esempio è possibile settare la potenza di trasmissione) per cui bisogna sempre guardare il manuale per non avere sorprese.

8. Tipi di antenna

I vari modelli di XBee appartenenti ad una famiglia (sia le varianti Low Power che le varianti PRO) si distinguono tra loro anche in base al tipo di antenna. Antenne più performanti consentono di avere una maggiore potenza di trasmissione e di conseguenza una maggiore distanza coperta. I costi (e i consumi) ovviamente aumentano man mano che il tipo di antenna consente una copertura maggiore.

I modelli con minore potenza di trasmissione, ma più economici e soprattutto compatti, hanno un'antenna stampata su PCB o un'antenna a Chip:



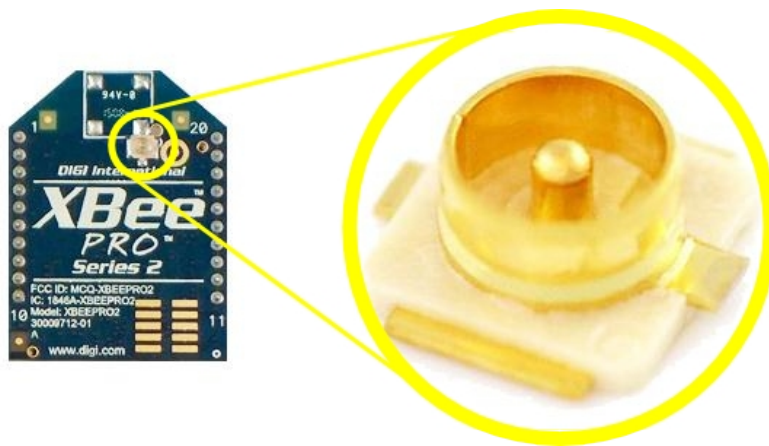
Oppure con antenna a filo (wire antenna):



Questi due modelli appena visti, con antenna integrata (*embedded antenna*), se devono essere rinchiusi all'interno di contenitori, questi non devono essere metallici. Nel caso in cui il contenitore da usare per il proprio progetto debba essere necessariamente metallico, bisogna utilizzare un modello con antenna esterna.

Abbiamo quindi i modelli che NON hanno un'antenna integrata e prevedono pertanto il collegamento di un'antenna esterna. Tali modelli consentono di raggiungere il massimo della potenza di trasmissione scegliendo opportunamente l'antenna da collegare. Il costo sale notevolmente in quanto è necessario acquistare l'antenna a parte. Questi modelli prevedono un **connettore U.FL** o un **connettore RPSMA**.

Il connettore U.FL è un connettore coassiale miniaturizzato con innesto a pressione. I connettori U.FL non sono progettati per essere rimossi ed innestati di continuo in quanto sono molto piccoli e abbastanza delicati:



Generalmente un'antenna con connettore U.FL una volta innestata non viene più rimossa. L'antenna collegata in questo modo necessita di essere agganciata ad un pannello in quanto non può essere montata direttamente sull'XBee.

Il connettore RP-SMA (scritto anche RPSMA) è anch'esso un connettore coassiale, ma di dimensioni decisamente maggiori, con fissaggio a vite:



Tale connettore è molto robusto e consente di avvitare l'antenna anche direttamente al corpo dell'XBee. E' possibile montare l'antenna anche su un pannello esterno, utilizzando un cavetto di prolunga.

9. Riconoscere il modello di XBee

E' possibile capire quale modello di XBee abbiamo tra le mani leggendo l'adesivo posto nella parte inferiore del modulo:



L'XBee nella foto, ad esempio, ha la sigla XBP24-ACI-001 che, capiremo tra poco, corrisponde ad un modulo XBee PRO Serie 1 con antenna a chip non destinato al mercato europeo e operante a 2.4GHz.

I moduli XBee hanno sempre la sigla che inizia per XB, segue una lettera P solo se si tratta di un modulo PRO. L'indicazione 24 ci fa capire che ci troviamo di fronte ad un modulo operante nella banda dei 2.4GHz (quelli operanti a 900MHz hanno scritto 09, quelli a 868MHz hanno 08).

Vi sono quindi una o più lettere che indicano il tipo di XBee: A=802.15.4, DM=DigiMesh, B=ZNET 2.5, Z7=Zigbee 2007, BZ7=Zigbee 2007 con hardware S2B.

Seguono 2 o 3 lettere che indicano il tipo di antenna: CI o CIT = antenna a Chip, PI o PIT = antenna stampata su PCB, WI o WIT = antenna a filo, UI o UIT = connettore U.FL, SI o SIT = connettore RPSMA. I modelli Serie 2B PRO programmabili hanno anche una B (o una A) dopo l'identificazione dell'antenna. Seguono quindi 3 numeri che indicano la funzione del firmware: 001 = coordinator AT, 002 = coordinator API, 003 = router/enddev API, 004= router/enddev AT. Vi è infine la J solo per i modelli PRO destinati all'Europa.

Il paragrafo successivo fornisce un elenco dei modelli di XBee presenti e passati.

10. Elenco modelli di XBee

In queste tabelle vengono elencati unicamente i modelli di XBee destinati all'Europa (quindi i modelli PRO hanno la J alla fine), operanti alla frequenza di 2.4GHz, con il relativo codice ufficiale della DIGI. In rosso sono indicati i modelli non più elencati sul sito della DIGI ma comunque diffusi dato che molti di loro sono ancora attualmente in produzione.

Serie 1

(XBee 802.15.4 OEM Low Power RF Modules)

Tipo Antenna	Codice DIGI
PCB *	XB24-API-001
Filo	XB24-AWI-001
Connettore U.FL	XB24-AUI-001
Connettore RPSMA	XB24-ASI-001

* Sostituisce il modello **XB24-ACI-001** (antenna a chip)

Serie 1 PRO

(XBee-PRO 802.15.4 OEM Extended Range RF Modules)

Tipo Antenna	Codice DIGI
Chip	XBP24-ACI-001J
Filo	XBP24-AWI-001J
Connettore U.FL	XBP24-AUI-001J
Connettore RPSMA	XBP24-ASI-001J

Serie 1 Digimesh

(XBee Digimesh Low Power RF Modules)

Tipo Antenna	Codice DIGI
PCB	XB24-DMPIT-250
Filo	XB24-DMWIT-250
Connettore U.FL	XB24-DMUIT-250
Connettore RPSMA	XB24-DMSIT-250

Serie 1 PRO Digimesh

(XBee-PRO Digimesh Extended Range RF Modules)

Tipo Antenna	Codice DIGI
Chip	XBP24-DMCIT-250J
Filo	XBP24-DMWIT-250J
Connettore U.FL	XBP24-DMUIT-250J
Connettore RPSMA	XBP24-DMSIT-250J

Serie 2 (ZNET 2.5)**(Obsoleta, aggiornabile a XBee ZB)**

Tipo Antenna	Codice DIGI
Chip	XB24-BCIT-004
Filo	XB24-BWIT-004
Connettore U.FL	XB24-BUIT-004
Connettore RPSMA	XB24-BSIT-004

Serie 2**(XBee ZB ZigBee Low Power RF Modules)**

Tipo Antenna	Codice DIGI
Chip	XB24-Z7CIT-004
Filo	XB24-Z7WIT-004
Connettore U.FL	XB24-Z7UIT-004
Connettore RPSMA	XB24-Z7SIT-004

Serie 2 PRO (ZNET 2.5)**(Obsoleta, aggiornabile a XBee ZB)**

Tipo Antenna	Codice DIGI
Chip	XBP24-BCIT-004
Filo	XBP24-BWIT-004
Connettore U.FL	XBP24-BUIT-004
Connettore RPSMA	XBP24-BSIT-004

Serie 2 PRO**(XBee ZB ZigBee Extended Range RF Modules - Soppiantata dalla Serie 2B PRO)**

Tipo Antenna	Codice DIGI
PCB	XBP24-Z7PIT-004J
Filo	XBP24-Z7WIT-004J
Connettore U.FL	XBP24-Z7UIT-004J
Connettore RPSMA	XBP24-Z7SIT-004J

Serie 2B PRO (Hardware S2B)**(XBee-PRO ZB S2B ZigBee Extended Range RF Modules)**

Tipo Antenna	Codice DIGI
PCB	XB24BZ7PIT-004J
Filo	XB24BZ7WIT-004J
Connettore U.FL	XB24BZ7UIT-004J
Connettore RPSMA	XB24BZ7SIT-004J

Serie 2B PRO Programmabile (*Hardware S2B*) (Programmable XBee-PRO Extended Range RF Modules)

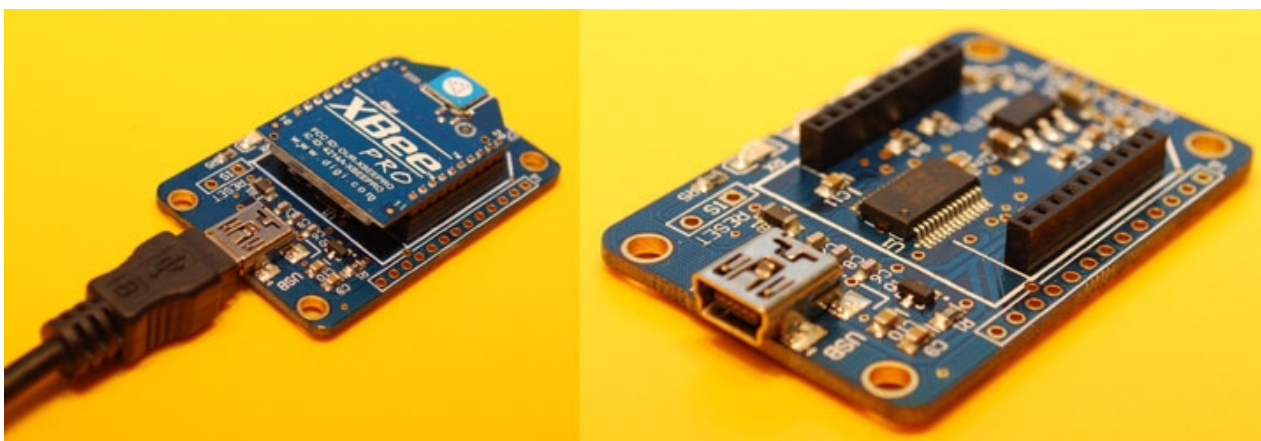
Tipo Antenna	Codice DIGI
PCB	XBP24BZ7PITB003J
Filo	XBP24BZ7WITB003J
Connettore U.FL	XBP24BZ7UITB003J
Connettore RPSMA	XBP24BZ7SITB003J

11. Configurare gli XBee : XCTU

XCTU è un'utility della DIGI per PC che permette di configurare gli XBee. Le funzioni di base sono quelle che prevedono il settaggio del BaudRate, l'aggiornamento del firmware, semplici test di comunicazione e cambio configurazione.

Anche se i modelli Serie1 sono di default già configurati per comunicare tra loro e quindi in un primo momento può sembrare inutile ricorrere ad una riconfigurazione, spesso è indispensabile ricorrere a questa utility per poter eseguire un aggiornamento del firmware o semplicemente per cambiare il baudrate o comunque fare delle prove di comunicazione qualora non riuscissimo a mettere in funzione il modulo o sorgano dei dubbi.

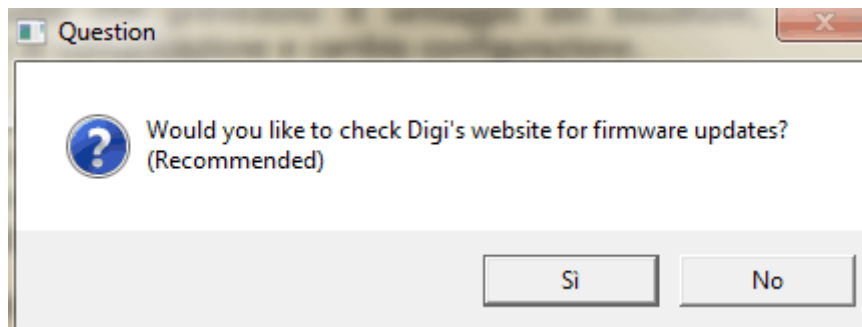
Per poter configurare l'XBee è necessario innanzitutto avere una scheda che permetta il collegamento dell'XBee al PC come la scheda [990002 XBee - USB Board](#):



XCTU può essere scaricato gratuitamente dal [sito della DIGI](#). In particolare per l'aggiornamento del firmware è richiesta un'interfaccia come questa che abbia collegate anche le altre linee di comunicazione piuttosto che unicamente le linee TX/RX.

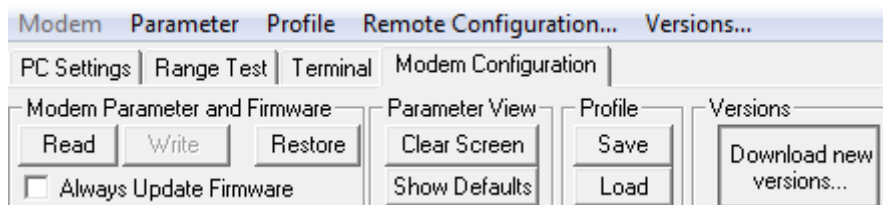
11.1 Aggiornamento delle revisioni firmware

La prima volta che XCTU viene installato, si presenta una finestra che richiede di controllare sul sito della DIGI se sono presenti aggiornamenti firmware:

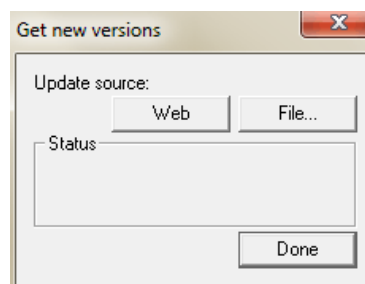


Questa operazione è importante e consente di scaricare i firmware per tutti i modelli di XBee: in questo modo qualsiasi tipo di XBee colleghiamo, XCTU sarà in grado, eventualmente, di aggiornarlo con il firmware più recente disponibile. Il processo di aggiornamento è molto lungo e può durare svariati minuti.

Se l'aggiornamento non va a buon fine questo passaggio può anche essere eseguito manualmente: basta scaricare il file ZIP presente nella pagina raggiungibile dal link [Firmware Updates](#). Si va quindi su XCTU, alla scheda "Modem Configuration" e si preme il pulsante "Download New Versions":



Viene quindi chiesto come eseguire l'aggiornamento, se in automatico via web o tramite il file ZIP che abbiamo scaricato manualmente:



L'aggiornamento tramite il file scaricato manualmente, comunque, non è mai completo ma prevede soltanto l'ultimo firmware relativo magari ad un solo tipo di XBee, è

possibile difatti sfogliare sul sito della DIGI le pagine relative ai vari tipi di XBee e cercare quindi il firmware relativo solo a quel tipo.

L'aggiornamento completo di tutti i firmware può essere eseguito soltanto via web o eventualmente scaricando a mano tutti i files dal [sito FTP della DIGI](#), ma quest'ultima possibilità non ve la consiglio assolutamente.

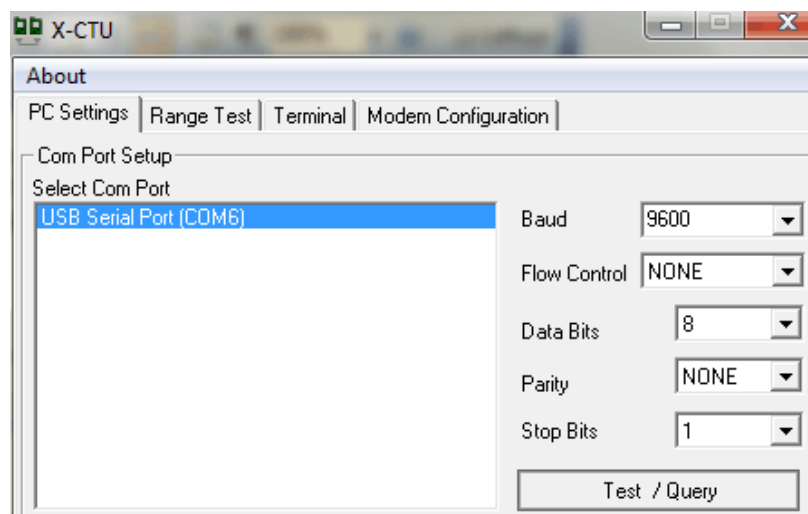
11.2 Collegamento del modulo XBee con XCTU

Collegando un adattatore USB al PC per la prima volta, viene generalmente richiesto un driver. La maggioranza di questi adattatori, compresa la [990002 XBee - USB Board](#), utilizza il chip FTDI232 e quindi i driver possono essere scaricati dal sito della FTDI (www.ftdichip.com) alla voce [Drivers/VCP Drivers](#). L'installazione crea in pratica una nuova porta COM sul pc.

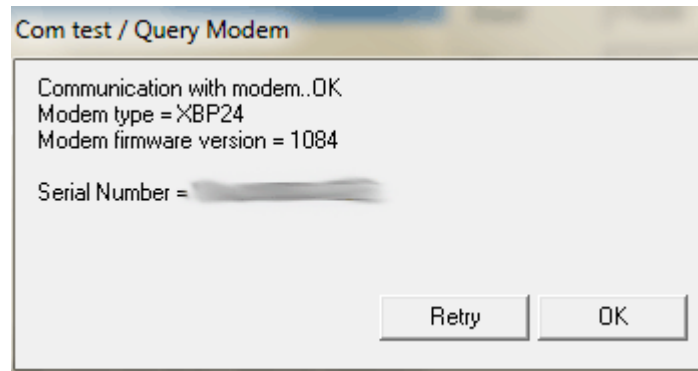
L'adattatore va collegato al PC (con sopra già innestato il modulo XBee) prima di avviare XCTU, altrimenti quest'ultimo non è in grado di rilevare la porta COM aggiuntiva.

Fate attenzione al collegamento del modulo: sulla [990002 XBee - USB Board](#) l'orientamento del modulo XBee è chiaramente indicato dalla serigrafia, tuttavia capita spesso che alcuni montino l'XBee al contrario. Quando il modulo è collegato correttamente e la scheda viene alimentata (collegata all'USB), il led contrassegnato con la scritta ASSOC deve lampeggiare.

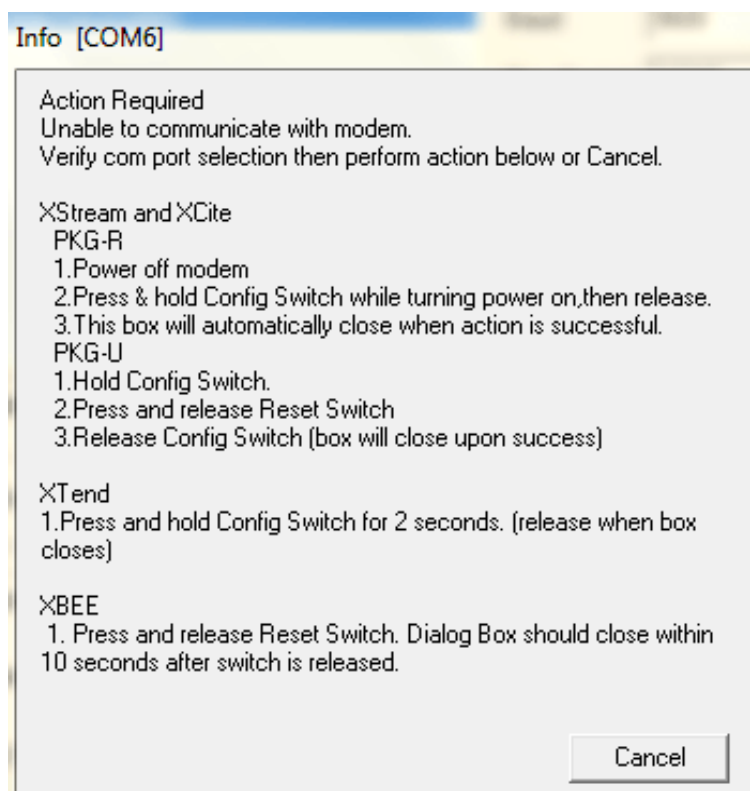
Una volta avviato XCTU bisogna eseguire una prova per vedere se il modulo è collegato correttamente. Si seleziona la porta COM dove è collegato l'adattatore e quindi il baudrate al quale è impostato il modulo. Il baudrate di default degli XBee è 9600, possiamo lasciare inalterati tutti gli altri settaggi. Dopo aver impostato questi due parametri è possibile premere il pulsante Test/Query:



Se è tutto ok, compare una finestra che avvisa della riuscita comunicazione con il modulo:



Altrimenti, se qualcosa non è andato per il verso giusto, si presenta una finestra simile:



I motivi più comuni che causano questo errore sono i seguenti:

- Non abbiamo selezionato la porta COM giusta
- Il modulo XBee è montato al contrario
- Il baudrate selezionato non è quello a cui il modulo XBee è impostato

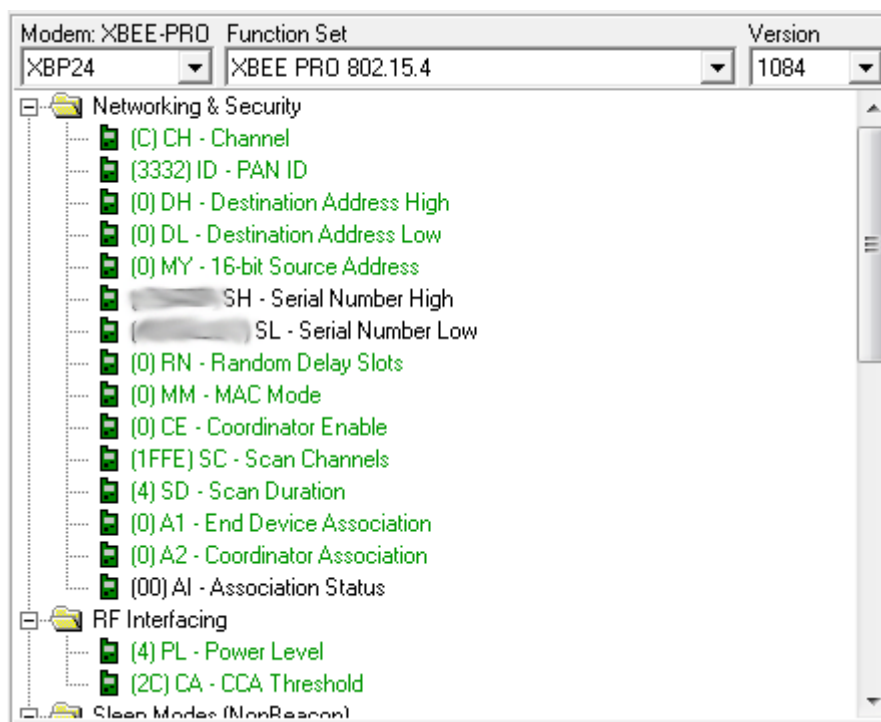
I primi due problemi si risolvono abbastanza facilmente, riguardo all'ultimo la cosa più semplice è quella di selezionare altri baudrate e fare quindi la prova con ognuno fino a quando il modulo risponde.

11.3 Lettura/Scrittura Firmware e Configurazione

Per eseguire tutte le impostazioni, compreso l'aggiornamento del firmware, si va sulla scheda *Modem Configuration* e si preme quindi il pulsante Read per leggere la configurazione attuale del modulo collegato:



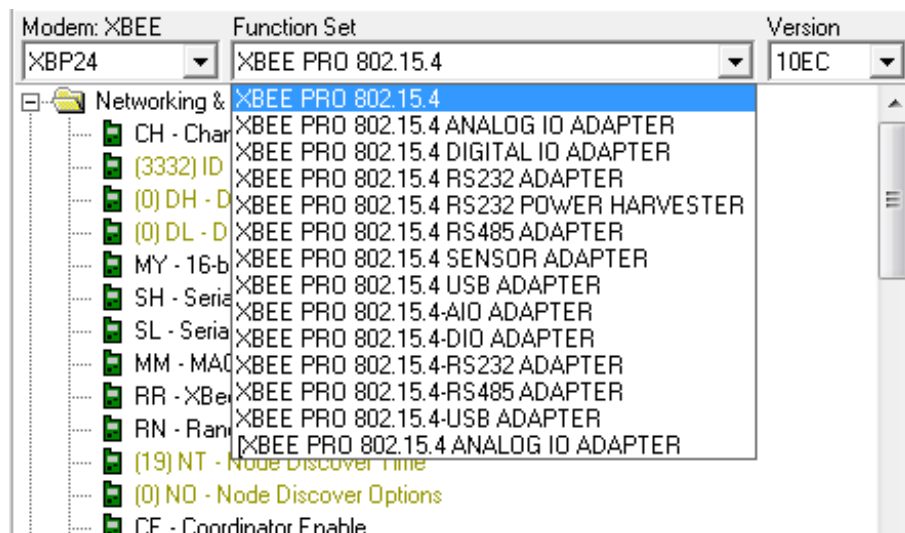
Compare la configurazione corrente del nostro XBee (indicato semplicemente come *modem* dal software), con il nome del set di funzioni caricato (Function Set) e la revisione del firmware (Version):



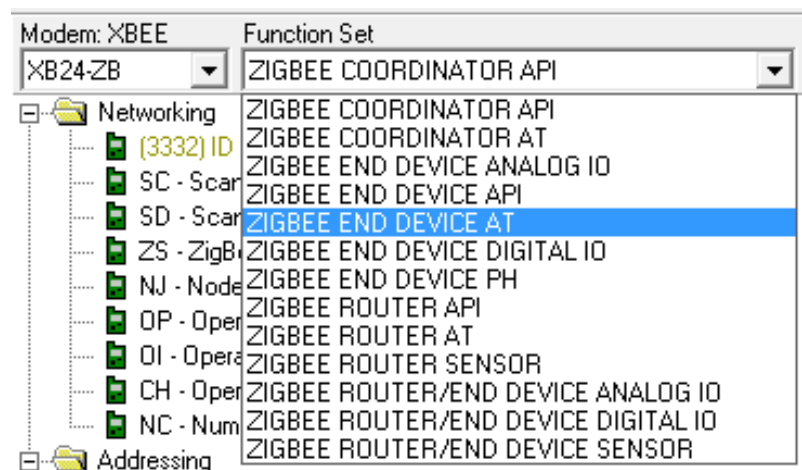
Da questa schermata si capisce che tipo di modulo abbiamo (Modem: XBEE-PRO, XBP24 = XBee PRO a 2.4GHz), quale set di funzioni è caricato sul modulo (XBEE PRO 802.15.4 in questo esempio) e quale revisione del firmware è presente (1084).

Nota: mi è capitato, personalmente, che alcuni XBee PRO vengano identificati semplicemente come XBee nella parte superiore (Modem: XBEE). In ogni caso la sigla XBP24 indica chiaramente un XBee PRO da 2.4GHz e il set di funzioni è quello di un XBee PRO.

Il set di funzioni, come dice il nome stesso, rappresenta in pratica la parte di firmware che consente al modulo XBee di operare in un certo modo: è possibile cambiare il set di funzioni per fare in modo che il modulo esegua compiti diversi:



In particolare, gli XBee Serie 2 hanno set di funzioni per operare come coordinatori, router o end device:

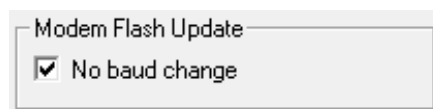


Scorrendo invece l'elenco presente sotto la scritta *Version* è possibile trovare firmware più aggiornati: il più recente è quello presente più in basso nella lista. Può accadere che una revisione di firmware più recente includa uno o più set di funzioni aggiuntivi rispetto alle versioni precedenti.

Quando andiamo a selezionare una diversa versione del firmware, un diverso set di funzioni o comunque vogliamo rendere definitive le impostazioni scelte sul nostro modulo XBee, bisogna premere il pulsante *Write* in alto a sinistra.

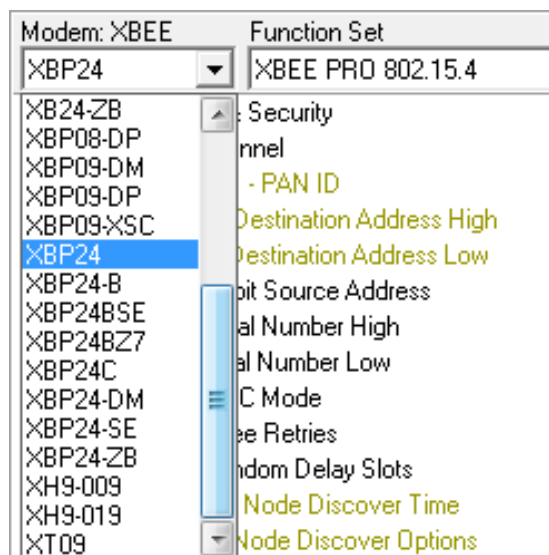
Quando eseguiamo questa operazione dopo aver scelto un diverso firmware, il led assoc smette di lampeggiare e durante la riprogrammazione i led TX RX mostrano l'attività in corso. Inutile dire che non bisogna staccare il cavo durante questa operazione ed è meglio eseguirla su un pc dotato di gruppo di continuità.

Se il baudrate di default non era 9600, dopo l'aggiornamento è normale che ci appaia la finestra che indica l'impossibilità di comunicare col modem dato che XCTU è ancora impostato con il baudrate precedente (la velocità di comunicazione del modulo è stata reimpostata dal nuovo firmware a 9600bps). In questo caso basta tornare sulla scheda *PC Settings* e impostare il baudrate a 9600 per comunicare nuovamente. Questo si può evitare se nel tab *PC Settings* selezioniamo l'opzione *No Baud Change*:

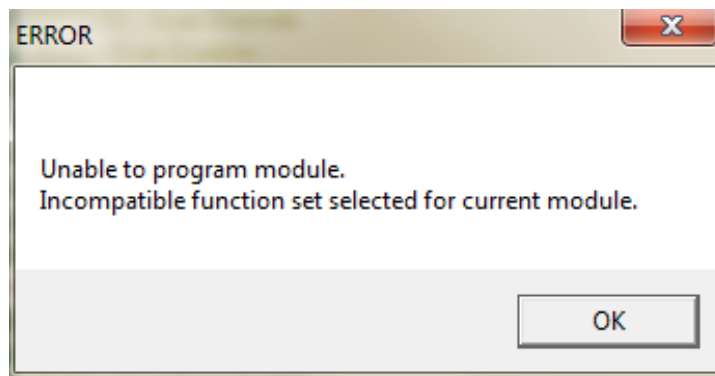


Nel caso avessimo cambiato alcuni settaggi che non permettono più al nostro modulo di operare correttamente, possiamo eseguire il ripristino della configurazione di fabbrica premendo il pulsante *Restore* dopo aver eseguito un *Read* della configurazione attuale.

Vediamo che ci è data anche la possibilità di cambiare il tipo di modem andando a scorrere l'apposita casella di selezione:



Questo può essere utile per capire ogni modulo che elenco di funzioni ha disponibili ma è ovvio che selezionando un modello di modem avente un hardware diverso da quello collegato e premendo il pulsante *write*, l'operazione non potrà mai andare a buon fine (!) dato che i set di funzioni sono strettamente correlati all'hardware del modulo:



Questa possibilità offerta da XCTU è tuttavia utile per eseguire alcuni particolari aggiornamenti.

Esempio: i moduli Serie 2 ZNET2.5 vengono indicati come XB24-B (o XBP24-B se si tratta della versione PRO). Per aggiornarli alla nuova versione dello stack Zigbee è possibile selezionare dalla casella a discesa il modello XB24-ZB (o XBP24-ZB per il PRO) e quindi scrivere il nuovo firmware: abbiamo capito che questo è possibile perchè in realtà i modelli XB24-B e XB24-ZB montano lo stesso hardware.

Stessa cosa per i modelli Serie 1 e Serie 1 Digimesh, un modello Serie 1, identificato ad esempio dalla sigla XB24, può diventare XB24-DM (Digi Mesh) e viceversa, in questo caso particolare, però, abbiamo visto che bisogna verificare l'effettiva compatibilità hardware come spiegato più in alto.

11.4 Alcuni parametri importanti

Per ogni set di funzioni abbiamo una serie di parametri che è possibile modificare cliccandovi sopra (parametro scritto in verde) e parametri di sola lettura. Ad esempio il numero seriale, diviso in due parti (*Serial Number High - SH* e *Serial Number Low - SL*) è un parametro di sola lettura.

Quando si vuole fare in modo che due moduli XBee comunichino soltanto tra di loro ignorando eventualmente altri moduli presenti nella zona limitrofa, basta fare in modo che i due parametri *Destination Address High* e *Destination Address Low* di un modulo siano impostati rispettivamente con il Serial Number High e Serial Number Low dell'altro modulo, così per entrambi i moduli. Ovvio che canale (*CH - Channel*) e *PAN ID* debbano avere lo stesso valore per entrambi i moduli.

Parametro da non sottovalutare è la voce *Interface Data Rate*. Se i moduli devono avere una comunicazione bidirezionale (ogni modulo deve cioè sia ricevere che trasmettere) è consigliabile non superare il valore di 57600 per non incorrere in frequenti errori di trasmissione causati dalla saturazione del buffer sugli XBee. Se una coppia di moduli deve funzionare soltanto in una direzione allora possono essere scelti valori più elevati.

Ovvio che gli XBee in una rete che devono comunicare tra loro, debbano tutti avere lo stesso baudrate.

Per ulteriori approfondimenti sull'utilizzo di XCTU vi rimando ai link elencati nel [paragrafo finale](#).

12. Un' ape in modalità API

La modalità API (*Application Programming Interface*) è un'alternativa alla modalità trasparente che consente, oltre che trasmettere e ricevere dati, di interagire ad un livello più basso con i moduli XBee consentendo, tra le altre cose, di:

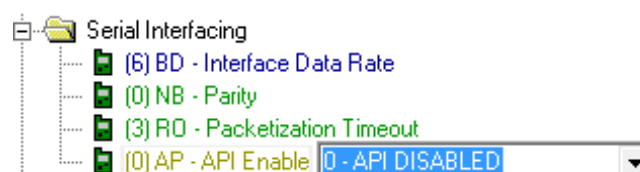
- Cambiare i parametri di configurazione (inviare comandi AT) senza entrare in [modalità comandi](#)
- Conoscere l' **RSSI** (*Received Signal Strength Indicator* - Indicatore di forza del segnale ricevuto)
- Ricevere una conferma di pacchetto dati consegnato correttamente per ogni pacchetto trasmesso o un'indicazione di consegna fallita
- Trasmettere dati a più destinatari
- Identificare l'indirizzo di chi ha trasmesso il pacchetto dati

La comunicazione in modalità API avviene inviando sulla linea seriale dei pacchetti di dati appositamente strutturati in *frames*. Un frame è costituito da un byte di start (0x7E), due byte che identificano la lunghezza del messaggio, seguono quindi i bytes di dati da inviare e infine 1 byte di checksum:



Si capisce quindi perchè la modalità trasparente abbia tale nome: i dati vengono inviati sulla seriale vengono inviati tal quale, in modalità API invece i dati vengono interpretati.

La modalità trasparente viene impostata, tramite XCTU, agendo sul parametro AP - API Enable. Il valore 0 identifica la modalità trasparente (API Disabled):



I valori 1 e 2 impostano rispettivamente la modalità API e la modalità API con i caratteri di escape (modalità, quest'ultima che si usa quando si ha la necessità di inviare dati che potrebbero interferire con la struttura del frame: se ad esempio tra i dati è necessario inviare il byte 0x7E -che viene usato per delimitare i frames- allora bisogna usare la modalità 2 ed utilizzare una particolare sequenza di escape per identificare il byte).

Fate riferimento al manuale dell'XBee in vostro possesso (link all' [ultimo paragrafo](#)) per capire come utilizzare la modalità API per le proprie applicazioni (paragrafi *API Operation*).

13. Modalità comandi - Command mode

La modalità comandi è quella normalmente utilizzata da XCTU per comunicare con il modulo XBee e permette di inviare comandi AT (lo standard utilizzato per tutti i modem).

Per entrare in modalità comandi, di default basta rispettare 3 semplici regole:

- Attendere almeno un secondo dall'ultimo dato ricevuto
- Inviare al modulo la stringa +++ (senza ritorno a capo!)
- Attendere un secondo

Il modulo deve rispondere con "OK".

Il secondo da attendere sia prima che dopo è definito dal parametro GT - Guard Times, di default impostato al valore 0x03E8 (1000 in decimale), mentre il carattere da inviare per 3 volte per entrare in modalità comandi è definito dal parametro CC - Command Sequence Character, impostato di default al valore 0x2B (codice ascii del +)



Se non vengono ricevuti comandi AT nel tempo definito dal parametro CT - AT Command mode timeout (di default impostato a 100 secondi), il modulo esce dalla modalità comandi e ritorna alla modalità attivata di default (trasparente o API). Si può uscire dalla modalità comandi anche manualmente inviando ATCN seguito dal ritorno a capo.

L'elenco dei comandi AT disponibili con gli XBee è elencato nel relativo manuale (link all' [ultimo paragrafo](#)) alla voce *Command Reference Tables*.

14. Links

14.1 Documentazione ufficiale DIGI

- [Tabella comparativa dei diversi moduli XBee](#)
- [Differenze tra 802.15.4 e ZigBee](#)
- [Datasheet XBee e XBee PRO 802.15.4 OEM RF Modules \(Serie 1\)](#)
- [Datasheet XBee e XBee PRO ZB Zigbee RF Modules \(Serie 2\)](#)
- [Datasheet XBee e XBee PRO Digimesh 2.4 RF Modules \(Serie 1 Digimesh\)](#)
- [Manuale d'uso XBee e XBee PRO 802.15.4 OEM RF Modules \(Serie 1\)](#)
- [Manuale d'uso XBee e XBee PRO ZB Zigbee RF Modules \(Serie 2\)](#)
- [Manuale d'uso XBee e XBee PRO Digimesh 2.4 RF Modules \(Serie 1 Digimesh\)](#)
- [Disegni dimensionali moduli XBee e XBee PRO](#)
- [Disegni dimensionali moduli XBee e XBee PRO variante RPSMA](#)
- [Manuale d'uso di XCTU](#)
- [Compatibilità tra Serie 1 Digimesh e Serie 1 802.15.4](#)
- [ZNET2.5 to ZB Conversion Kit](#)
- [Cos'è la modalità API e come funziona](#)

14.2 Software

- [XCTU](#)

14.3 Risorse in rete

- Robot-Italy Forum: [Semplice comunicazione tra due moduli XBee Serie 2](#)
- Settorezero.com : [Applicazione pratica con picmicro e settaggi XBee Serie 1](#)
- Rob Faludi : [Librerie per utilizzare la modalità API degli XBee con Processing](#)
- Andrew Rapp : [Librerie per utilizzare la modalità API degli XBee in Java](#)
- Rob Faludi : [XBee level shifting](#)

14.4 Schede di interfacciamento per XBee

- [990001 XBee - Simple Board](#) (interfacciamento generico con livelli TTL)
- [990002 XBee - USB Board](#) (interfacciamento su USB)
- [990006 Scheda seriale DTE per XBee](#) (interfacciamento RS232)
- [990102 1000Pads-Mini XB Board](#) (protoboard per XBee, completa di componenti)
- [990102L 1000Pads-Mini XB Bare Board](#) (protoboard per XBee, senza componenti)

14.5 Schede correlate

- [Convertitore di livello logico bidirezionale](#)
- [Convertitore di livello logico doppio](#)