

XBee API-Tool in Python / Tkinter

Vor einiger Zeit hatte ich ein kleines Tool entwickelt, mit dem XBee's im API-Modus angesprochen werden können. Dieses Tool war in C geschrieben, sparsam im Speicherbedarf, hilfreich bei der Erforschung der XBee's, aber auch sehr spartanisch in seiner Bedienung.

Nun habe ich dieses Tool in Python neu/umgeschrieben und ihm mit tkinter eine grafische Nutzeroberfläche spendiert.

Was kann man mit dem Programm anfangen ?

Man kann AT-Kommandos, Remote-AT-Kommandos und DATA-Frames im API-Modus an die Xbee-Module senden. Das Programm nimmt die Eingabe entgegen, schnürt alles in das zugehörige API-Packet, versieht es mit einer Prüfsumme und versendet es.

Die Antwort der Gegenstellen wird formatiert ausgegeben, so dass man schnell die relevanten Informationen erkennen kann.

Wozu ist das gut ?

Man kann "lokale" und "remote" XBee's konfigurieren, IO-Pins einstellen, Parameter setzen oder auslesen usw. (Im AT-Modus ist das bekanntlich ausschließlich an der lokalen Station möglich.)

So kann man "spielerisch" die Funktionsweise der Module testen und zu verstehen lernen.

Als DATA-Frame können beliebige Datenstrukturen, die gemäß eigener Spezifikation aufgebaut sind, versendet und empfangen werden.

Dieser Modus ist nützlich, wenn man sich ein eigenes XBee-Netz aufgebaut hat und darin zu Testzwecken eingreifen oder debuggen muss.

Das Programm ist unter Python 3.2 entstanden, für die Serielle Schnittstelle muss "pyserial" installiert sein.

Der Bildschirm ist in drei Bereiche gegliedert, in zwei Fenstern, in denen die Kommunikation mit dem lokalen XBee-Modul ausgegeben wird und einen Eingabebereich mit drei Registern zur Eingabe von Local AT-, Remote AT- und Data-Frames.

In jedem Register werden 4 unabhängige Eingabefelder angeboten.

Zu jedem Eingabefeld kann eine eigene Zielstation ausgewählt und ein eigene Eingabe ausgeführt werden. (Bei den Local AT Frames fehlt die Auswahl der Zielstation logischerweise.)

In den Auswahlfelder der Stationen (ComboBoxen) werden die frei definierbaren Namen der Stationen angezeigt. Zusätzlich wird die aktuelle 16-Bit Netzwerkadresse vorangestellt.

Diese Adresse steht beim Programmstart auf "FF FE", wird aber durch die aktuelle Netzwerkadresse ersetzt, sobald diese identifiziert werden konnte.

Im Ausgabebereich oben rechts wird für jede Transaktion zwischen PC und XBee eine Zeile mit den vollständigen Daten des gesendeten bzw. empfangenen Frames in hexadezimaler Darstellung ausgegeben.

Die logischen Informationseinheiten sind zur besseren Lesbarkeit durch Kommata getrennt.

Diese Daten können bei Bedarf in einer Logdatei aufgezeichnet werden.

Im Ausgabebereich links werden detaillierte und formatierte Informationen zu jedem einzelnen Frame angezeigt.

In den Optionen.Einstellungen können Frametypen ausgewählt werden, deren Anzeige dann unterdrückt wird.

Weiterhin können in Optionen.Einstellungen.Tx-Frames einige Parameter für die Frames 0x10 und 0x17 eingestellt werden.

In Optionen.DeviceListe wird die Liste aller in der Datei "xbee_adr.cfg" angemeldeten Stationen angezeigt.

Die jeweilig aktuelle 16-Bit Netzwerkadresse steht rechts neben der konstanten 64-Bit Adresse.

Sobald von einer Station ein Frame mit einer 16-Bit Adresse empfangen wird, wird die Liste aktualisiert .

In der rechten Spalte wird "online" eingetragen, sobald eine Station nach dem Programmstart erkannt wird.

Geht eine Station später "verloren", dann wird die 16-Bit Adresse wieder auf "FFFE" gesetzt und in der rechten Spalte "offline" eingetragen.

Die Station mit der 16-Bit Adresse "0000" ist der Coordinator, entsprechende Hinweise werden eingetragen.

Ausserdem wird die Signalstärke beim letzten Empfang von Daten dieser Station ausgegeben.

Zu beachten ist, dass bei Frames, die über mehrere "Hops" eingehen, nicht die Signalstärke des Absenders sondern die des letzten Hops angezeigt wird.

Zwei Worte zur Eingabe.

Lokale AT- und Remote AT-Commands erwarten als erste beide Zeichen ein gültiges AT-Kommando (ein 'AT' darf vorangestellt werden). Großschreibung ist nicht signifikant.

Wird kein AT-Kommando erkannt, dann gibt es ein akustisches Signal, das Kommando wird nicht gesendet.

Hinter dem AT-Kommando können - durch Kommata getrennt - Zahlenfolgen eingegeben werden.

Direkt hinter dem AT-Kommando darf ein Komma stehen, muss aber nicht.

Die Eingabe erfolgt nach Belieben als Dezimal-, Hexadezimal- [0x...], Oktal- [0o...] oder Binärzahl [0b...].

Entsprechend des Wertes der Eingabe wird eine Bytefolge von 1, 2 oder 4 Byte erstellt.

Sobald eine Eingabe nicht ohne Fehler in eine Bytefolge umgewandelt werden kann, wird der gesamte Rest des Eingabestrings als Zeichenfolge übernommen.

```
dh0x12,0xab,0x3,0      -> DH,12AB0300
dh0,0b1101,0o77,0xae   -> DH,000D3FAE
dh,0b00,0b11,0o77,0xae -> DH,00033FAE
dh0,0b21,0o77,0xae     -> DH,00306232312C306F37372C30786165
                        ("0b21" kann nicht als Zahl interpretiert werden)
```

Um ein Zahl als Zeichenfolge zu übernehmen wird ein "\" vorangestellt.

```
ni\1000      -> NI,31303030
```

ohne "\" wird der String zur Bytefolge umgerechnet:

```
ni1000      -> NI,03E8
```

Um eine Zahl zwingend in eine Folge von 1, 2 oder 4 Byte zu formatieren, wird hinter die Zahl ein "[1,2,4]" angehängt.

```
dh0|4       -> DH,00000000
dh,1|2      -> DH,0001
dh256|1     -> DH,00
```

Zuletzt noch:

Im virtuellen Kreuzungspunkt des horizontalen und vertikalen Scrollbars gibt es einen kleinen Button mit einem "X". Der löscht den Inhalt des zugehörigen Fensters.

Bekannte Probleme:

Beim Starten überprüft das Programm, ob es die Seriellen Schnittstellen von 0 .. 9 öffnen kann.

Die Schnittstelle mit der niedrigsten Nummer wird zur Kommunikation geöffnet.

Unter Window7 werden bei mir aber Schnittstellen angezeigt, die aktuell physikalisch nicht mehr vorhanden sind ...

In diesem Falle muss man von Hand nachhelfen.

In der Datei "xbee.cfg" kann folgender Eintrag vorgenommen werden:

```
COMPORT : COMx      Wobei x die Schnittstelle [1..9] darstellt.
```

Unterschiedliche Fonts auf den Rechnern/Betriebssystemen können das Layout verändern.

In die Tiefen des ttk.Style() konnte ich leider bislang nicht eindringen.

Auf Netbooks mit Auflösung 1024/748 muss die Höhe der Editorfelder reduziert werden.

Dazu gibt es in der "xbee.cfg" Einträge, die angepasst werden können.

Ebenso müssen die Geräteadressen und Namen der verwendeten Xbee-Module in der "xbee.cfg" angemeldet werden.

Auf einem Netbook mit Windows7 und dauerte der Start des Programmes manchmal (nicht immer) seeehr lange.

PS.

Dies ist mein erstes umfangreicheres Programm in Python / tkinter - also bitte ich um Nachsicht, wenn einiges noch im Experimentierstadium ist und nicht immer elegant gelöscht wurde.

Konstruktive Anregungen werden natürlich gerne entgegengenommen.

Michael S.
20.12.12