**Modules de transmission 868MHz**

**Description**

Module radio 868MHz permettant d’échanger des données entre deux ou plusieurs modules.

Ces modules doivent être simple à utiliser, ils doivent se comporter comme un port série, donc des caractères ascii envoyés en UART.

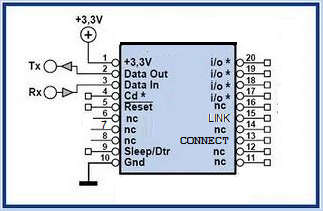
Possibilité de faire du broadcast.

**Caractéristiques:**

* Alimentation 3.3V
* Signaux UART 3.3V : RX, TX.
* Signal de statut de la connexion : LINK.
* Signal de requête de connexion : CONNECT.
* Signal de reset actif bas : RSTB.
* IOS génériques : GPIO0, GPIO1, GPIO2, GPIO3.
* module radio nRF905 868MHz.
* microcontrôleur PIC32MX.
* mémoire flash pour stocker un identifiant unique (UUID).
* L’antenne est directement implémenté sur le PCB.
* Le module sera connecté avec un connecteur mezzanine
* Connecteur compatible avec un Xbee.

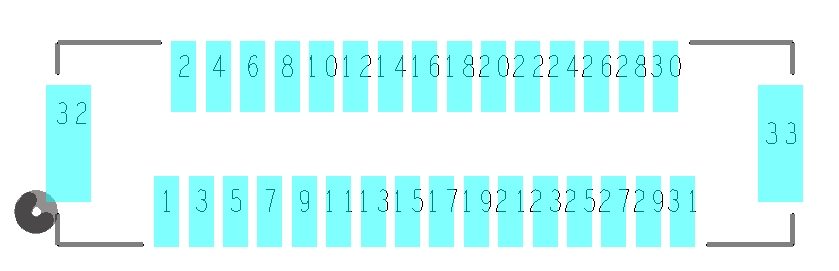
**PIN connecteur Xbee**

|  |  |
| --- | --- |
| **1** | **Vcc** |
| **2** | **Tx** |
| **3** | **Rx** |
| **4** | **CD** |
| **5** | **MCLR** |
| **9** | **SLEEP** |
| **10** | **GND** |
| **13** | **CONNECT** |
| **15** | **LINK** |
| **17** | **PDC/IO4** |
| **18** | **PGC/IO3** |
| **19** | **IO2** |
| **20** | **IO1** |

****

**Connecteur mezzanine**

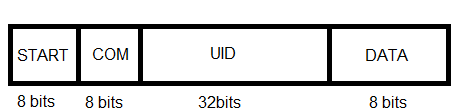
|  |  |
| --- | --- |
| **28,29** | **Vcc** |
| **1** | **Tx** |
| **2** | **Rx** |
| **5** | **CD** |
| **22** | **MCLR** |
| **4** | **SLEEP** |
| **30,31** | **GND** |
| **6** | **CONNECT** |
| **7** | **LINK** |
| **9** | **PDC** |
| **8** | **PGC** |
| **11** | **IO2** |
| **10** | **IO1** |

****

**Exemple d' utilisation**

**Communication UART:**

Trame UART envoyée et reçue.

****

* **Start** : Toujours 0xAA.
* **COM:** Commande pour configurer le module 1 Tx, 2 Rx, 0 Standard et 3 lectures UID.
* **UID:** Adresse du module avec qui on veut communiquer (0 = tout le monde).
* **DATA:** caractères affichables de la table ASCII non étendu.

**Mode Tx :**

1. Mettre un 0 sur l’entrée CONNECT pour indiquer au module que l’on veut le paramétrer et lui envoyer la commande 1 pour Tx.
2. Mettre l’entrée CONNECT à 1 et attendre que la patte LINK passe à 1.
3. Commencer à envoyer les data.

**Mode Rx :**

1. Mettre un 0 sur l’entrée connecte pour indiquer au module que l’on veut le paramétrer et luis envoyer la commande 2 pour Rx.
2. Mettre l’entrée CONNECT à 1 et attendre que la patte LINK passe à 1.
3. Attendre de recevoir les data .

**Mode standard (bidirectionnelle**)**:**

1. Mettre un 0 sur l’entrée CONNECT pour indiquer au module que l’on veut le paramétrer et lui envoyer la commande 0 pour standard.
2. Mettre l’entrée CONNECT à 1 et attendre que la patte LINK passe à 1.
3. Envoyer ou recevoir des data (tout est fait automatiquement).

**lecture UID:**

1. Mettre un 0 sur l’entrée CONNECT pour indiquer au module que l’on veut le paramétrer et lui envoyer la commande 3 pour lecture UID.
2. UID renvoyer en boucle toutes les 10 ms dans la case UID de la trame UART.

.

**Electrical characteristics**

**Recommended Operating Conditions:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Min** | **Typ** | **MAX** |
| **Supply Voltage** [V] | 3 | 3.3 | 3.6 |
| **Temperature** °C | -20 | 20 | 80 |

**CURRENT**

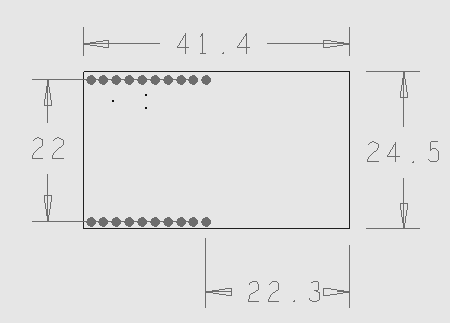
|  |  |  |  |
| --- | --- | --- | --- |
|  | **Min** | **Typ** | **MAX** |
| **Supply Current:**  **mode TX** | **-** | **14mA** | **34mA** |
| **Supply Current:**  **Mode Rx** | **-** | **11mA** | **13mA** |
| **Supply Current:**  **Standbye** | **4mA** | **6 mA** | **8mA** |
| **PIN Current: source/sink** | **-** | **-** | **10mA** |

**Device Times:**

|  |  |  |
| --- | --- | --- |
|  | **MIN** | **MAX** |
| **power UP** | **-** | **16ms** |
| **Standbye -> Rx mode** | **-** | **13ms** |
| **Standbye -> Tx mode** | **-** | **14ms** |
| **Tx mode -> Rx mode** | **-** | **13ms** |
| **Rx mode -> Tx mode** | **-** | **14ms** |
| **Connect time** | **5s** | **-** |
| **Send data to receive UART** | **12ms** | **25ms** |

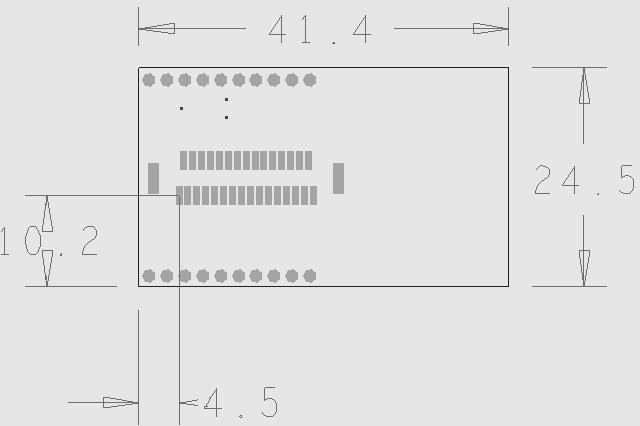
**Mechanical characteristics [mm]**

**Xbee connector**

****

**Pitch 1.5mm**

**Mezzanine Connector**

****

