

### Linux での USB-CDC デバイスの認識

Linux では USB-CDC クラスのデバイスは OS が自動的に認識し、ドライバが適用されます。Windows のように特別なドライバは不要です。

USB-CDC 機器を接続した直後に root のアカウントで Terminal を開き、dmesg コマンドを実行します。

```

Terminal
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
root@kona:~# dmesg
[ 4362.576991] usb 1-2.1: new high-speed USB device number 5 using xhci_hcd
[ 4362.988457] usb 1-2.1: New USB device found, idVendor=098f, idProduct=2205
[ 4362.988487] usb 1-2.1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[ 4362.988530] usb 1-2.1: Product: DCS-1104B
[ 4362.988556] usb 1-2.1: Manufacturer: TEXIO
[ 4362.988582] usb 1-2.1: SerialNumber: GJP180497
[ 4363.376707] cdc_acm 1-2.1:2.0: This device cannot do calls on its own. It is not a modem.
[ 4363.378076] cdc_acm 1-2.1:2.0: ttyACM0: USB ACM device
[ 4363.381822] usbcore: registered new interface driver cdc_acm
[ 4363.381859] cdc_acm: USB Abstract Control Model driver for USB modems and ISDN adapters
    
```

機器をつないだ後のdmesgコマンドの表示の最後に表示されます。

利用するポート名

dmesg コマンドの実行結果の最後に追加した USB-CDC デバイスが ttyACM+番号または ttyUSB+番号で表示されます。このポート番号はシリアルポートとして認識され、番号は記憶されます。他の機器と重複することはありませんが、機器が増えるにしたがって番号が増加するので注意してください。

追加されたポートのパーミッションは専用の dialout になっていますので、他のユーザーから利用できません。ttyACM0 の場合

```
Chmod 666 /dev/ttyACM0
```

のコマンドで読書き可能へ変更が必要です。

## Linux でのシリアルポートの通信

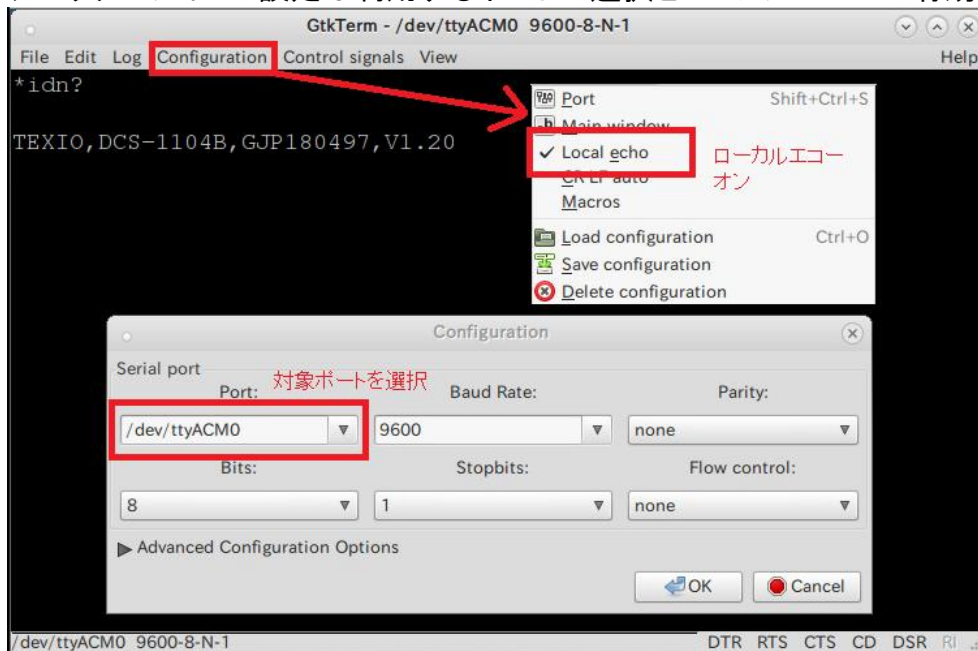
通信テストは Windows の通信アプリと同等のプログラムを用意して行います。例として GtkTerm の使用例を以下に示します。

### 1. GtkTerm のインストール

Linux のアプリケーション管理ツールで GtkTerm をインストールしてください。apt または yum などによるインストールでも問題ありません。インストール後に GtkTerm を実行してください。

### 2. GtkTerm の設定

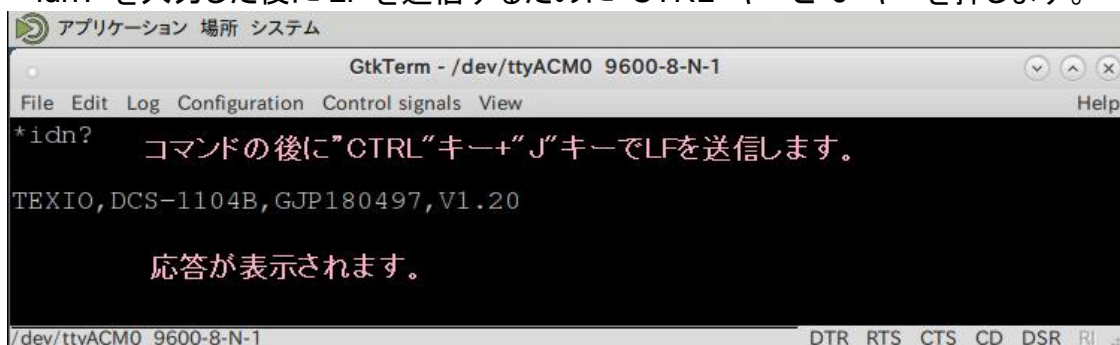
アプリケーションの設定は利用するポートの選択とローカルエコーの有効化を行います。



起動後に表示される Configuration ダイアログの Port で先ほど認識した ttyACM のポートを選択し OK をクリックします。次にアプリケーションの Configuration メニューをクリックし Local echo をクリックしてチェックを入れます。

### 3. 通信テスト

“\*idn?”を入力した後に LF を送信するために“CTRL”キーと“J”キーを押します。



コマンドが認識されると応答が表示されます。

応答が無い場合や本体にエラーが出る場合は LF を数回送信後に再度コマンドを送信して下さい。エラーは\*CLS コマンドでクリアされます。

以上で通信の確認が完了です。

実際の制御プログラムについては、C++や Python などを利用してください。

商用アプリとしてはナショナルインスツルメンツ社の LabVIEW があります。