



# CAN232

## Version 3

### マニュアル



February 2010  
Version 3.0

CAN232 Manual

このマニュアルではそのように明確に示されていない、著作権で保護された製品に対する記載があります。著作権©記号の欠如は、製品が保護されていないことを暗示してはなりません。更に、登録された特許と商標をこのマニュアルでは同様にはっきりと示しています。

この文書の情報は非常に慎重にチェックされ、信頼できると信じています。しかし、**LAWICEL 社**はいかなる不正確さに対しても責任を負いません。**LAWICEL 社**は、このマニュアルあるいはその関連製品の使用から生じた損害に対し、保証も与えず、何らの責任も認めません。**LAWICEL 社**は、前もっての通知なしに、ここに含まれる情報を変更する権利を保有します。

更に、**LAWICEL 社**は、ここに記載したハードウェアの不適切な使用、または不適切に組み付けたために起こる損害に対して保証を提供するものではなく、いかなる義務も負いません。最後に、**LAWICEL 社**は前もっての通知なしにハードウェアを変更し、設計する権利を保有し、そのようにすることに対する責任を認めません。

Copyright 2001-2003 **LAWICEL AB**

All rights reserved. Printed in Sweden.

翻訳、再版、放送、写真製版または同様の複製を含みます。  
LAWICEL社からの同意書なしに再生を行うことはできません。

**LAWICEL AB**  
Klubbgatan 3  
S-282 32 Tyringe  
SWEDEN  
Phone: +46 451 59877  
FAX: +46 451 59878  
<http://www.CAN232.com/>  
[info@CAN232.com](mailto:info@CAN232.com)

## 1.0 はじめに

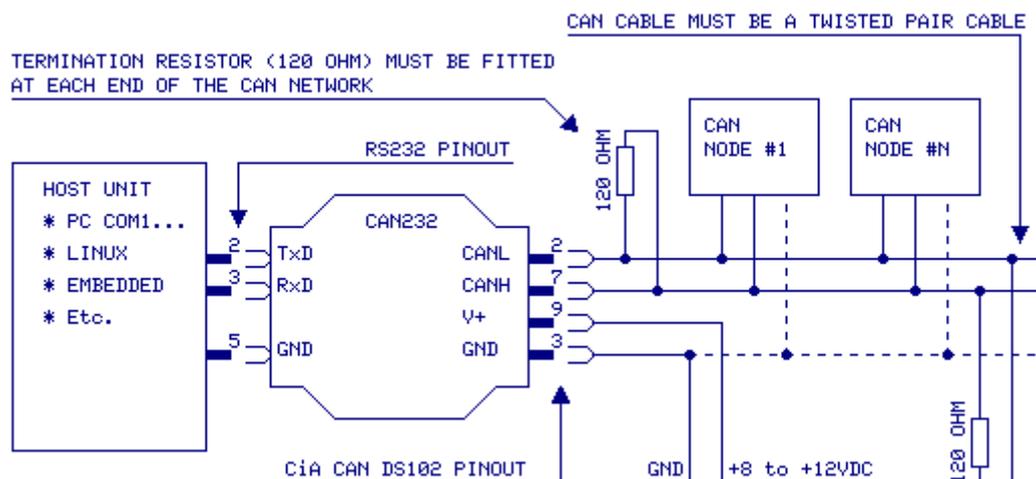
LAWICEL CAN232 は、低コストで使い易い dongle です。RS232 から CAN へのゲートウェイなので、ドライバーなしでもどんな OS とも一緒に使うことができます。DOS、Windows95/98/ME、NT4/2000/XP/Vista/Windows 7, Mac または Linux で動いているどんな PC にでも単純に接続し、標準の ASCII フォーマットでユニットと "talk" してください。また、既存のハードウェアを変えずに単純に CAN 接続を必要としている組み込みコンピュータと共に使うことができます。CAN232 は、29 ビット ID フォーマットと同じく 11 ビット ID フォーマットの両方を取り扱い、ビルトイン FIFO キュー、拡張 info/error 情報、わずかのコマンドによる簡単なパワーアップを取り扱います。CAN232 は、ボードの両側に小さな SMD 部品の最新技術を使用して、長さ 68mm、幅 31mm、厚さ 16mm しかなく、Atmel AVR, Philips SJA1000 CAN コントローラと FTDI Parallel FT245BM USB チップを陰の力として、高速バスの CAN フレームの小さいバーストを非常にフレキシブルに取り扱います。

CAN232 は、ニーズによって CAN to RS232 ユニットとしてカスタマイズできます。(即ち、既存 RS232 製品の変換するのに高価な CAN への変換、あるいは、RS232 等に関して、ネットワークを通常の長さ以上に拡張。) この文書は、ブートローダ経由で更新することができる CAN232 のバージョン 3 に関して記載されています。(CAN232 のバージョン V1324 またはそれ以上)

CAN232 マニュアルのオリジナルバージョン 1,2 は、www.CAN232.com で見つけることができます。このマニュアルのいくつかのコマンドは、古いオリジナルバージョンでは機能しません。加えて、ここで示された性能 (速度とバッファなど) は、新しいバージョン 3 にのみ有効です。

## 1.1 インストール

dongle の RS232 側 (DB9 メス) を、PC の COM シリアルポートに直接、またはケーブルを経てホストシステム (例えば、組み込みシステムなど) へ挿入します。dongle の CAN 側 (DB9 オス) は、CAN in Automation (CiA) DS102 プロファイルと同じピン配列となっており、CAN232 dongle は、CAN 側を経て、6~16VDC で給電されなければなりません。dongle は ESD 保護されており、電源の逆接は、CAN232 に損害を与えません、その代わりに、電源は CAN232 dongle を保護するために短絡されます。CAN232 dongle は、CAN ネットワークが負荷される程度 (即ち、ノード数など) に依存して、約 40-100mA を必要とします。\*下図は、CAN232 接続の簡単な図示です。



## 1.2 CAN232 の試験

CAN232 を前のページの 1.1 節の導入に従って PC の COM ポートにインストールし、電源を入れて試験して下さい。CAN232 がパワーを得ると、4 つの LED (赤と黄と 2 つの緑) が、セットアップされている RS232 の速度によって、数回素早く点滅します。もし、RS232 が 57,600 ボー (出荷時のデフォルト) なら、4LED 全てが 3 回点滅します (より高い RS232 速度では、より少ない回数で点滅する、詳細は U コマンド参照)。そして、Windows Terminal ソフトウェア (あるいは好きなターミナルソフトウェア) を起動し、57,600 ボー、8 データビット、パリティなし、1 ストップビット (もし、CAN232 が 57,600 ボーにセットされていれば) に設定し、また、あなたが入力するものをあなたが見られるようにローカルエコーをセットし、それを受信したときと、ラインの終わりに、改行を追加するようにチェックフラグをセットします。最後に、ハードウェアとソフトウェアハンドシェイクをオフにし、出力される CR (Ascii=13) に LF (Ascii=10) が追加されることを確認します。そして、接続していることを確認し、>ENTER< を押してください。新しい行を作成するでしょう、それから、V と>ENTER<を押してください。それが Vhhss を返答しプリントします。ここでは hh がハードウェアのバージョンで、ss がソフトウェアのバージョンです (例えば V1335)。CAN232 ユニットで完全な接続をし、CAN 速度を設定し、CAN ポートをオープンし、フレームを送受信できます。CAN コネクタ側の緑の LED が、CAN フレームがうまく送られるか、CAN232 ユニットを受信されたことを示すことに注意して下さい。そして RS232 側の緑 LED は RS232 のデータ受信で点滅します (V3 で新規に)。CAN フレームを送受信するために、少なくとも 2 ノード (CAN232 は、読み取り専用モードに設定されていなければ 1 ノードとして動作します。) を持たねばならないこと、CAN ケーブルネットワークが CANL と CANH ライン上の両端を 120 オームで終端されること、その上、ツイストペア CAN ケーブルを用いていることにご注意下さい。CAN232 は全フレームを受け入れるようにセットされているので、試験のためにフィルターなどのセットは必要ありません。CAN232 は、また、www.CAN232.com のサンプルプログラムでも試験できます。

### CAN232 テストの例

```
V[CR]          (バージョンたとえば V1324[CR]と返答されます。)  
S4[CR]        (CAN 速度 125Kbit に設定)  
O[CR]         (CAN チャンネルオープン、黄 LED が点灯します。)  
t1001AA[CR]   (16 進 ID=0x100、DLC=1、データ 1 バイト 0xAA を送信)
```

## 1.3 CAN232の限界

もちろん、CAN232が如何に多くのCANフレームを送受信できるかの限界があります。現在のバージョン (V13nn) を、125kビットCANビットレートとRS232の115,200ボーレートで、8データバイトの500スタンダード11ビットフレーム送信のスループットにより試験します。もちろん、ボトルネックは、秒当たり多くのフレームを処理できないRS232側とマイクロコントローラです。そして、CAN232は低速のCANに狙いを定められて、125kビットかそれ以下のCAN速度で非常にうまく作動しますが、それは1Mビット (しかし、バスロードをこの速度では高くできない、あるいは、例えばフィルターをフレームの一部を受け入れるようにセットしなければならぬ) まで使用できます。CAN232は送信と受信のための、ソフトウェアCAN FIFOを持っています。これらの送信FIFOは8フレーム (スタンダードあるいは拡張) を取り扱うことができ、一方、受信FIFOは32フレーム (スタンダードあるいは拡張) を取り扱うことができます。更に、CAN232は小さなRS232バッファしか持たないので、1度に1つか2つのコマンドだけを扱い、それは次のコマンドの送信前に、CAN232ユニットからの答え (OKである [CR] あるいはエラーの [BELL]) を待たなければならぬことを意味しております。

## 1.4 ドライバー設計ガイド

CAN232 は、ドライバーと共に納められません。多くの市販開発ツールが RS232 ASYNC LIB (Visual Basic, Delphi のような) を提供しているため、CAN232 ユニットと”talk”するための簡単なプログラムを書くことは容易です。最良の方法は CAN232 への全ての通信を扱い、アプリケーションに従って全てのメッセージを FIFO またはメールボックスに置くスレッドを作成することです。

古い **P** および **A** コマンドを使用する代わりに、新しい CAN232 の新しい”AUTO POLL/SEND”機能を用いることを強く勧めます。この新機能をオンにし旧機能を禁止にする方法についての、より多くの情報のため **X** コマンドを見て下さい。(この動作は EEPROM にセットされ、次の電源オンまで記憶されるので、設定するたびに、セットする必要はありません)。

この新機能はまた、**t** と **T** コマンドからの返答も変え、それはフレーム送信時、[CR] のみを返答する代わりに、コマンドに依存して **z**[CR] あるいは **Z**[CR] を返します。その理由は、CAN232 からのコマンド解析を簡単にするためです (すなわち、送信コマンドが CAN232 から送り出されたフレーム間で、良好になっていることの認識のため)。

常に、全ての前のコマンドあるいは CAN232 の待ち行列に入った文字 (電源投入時、何度も、待ち行列に間違った文字か、前のセッションからの古いもの) を空にするために、2-3 の [CR] を送信して、各セッション (プログラム起動時) を起動します、**V** コマンドで CAN バージョンをチェックし (正しい速度でユニットと通信していることを確認するため)、**s** あるいは **S** コマンドで CAN 速度を設定し、**O** で CAN ポートをオープンして、CAN232 が CAN フレームの送受信の両方に対し作動中となります。**t** あるいは **T** コマンドによりフレームを送信し、それが CAN 送信 FIFO 列に置かれるか、あるいは列が一杯かを見るために返答を待ちます。CAN バスから入ってくるフレームはすぐに RS232 へまたは、RS232 がいっぱいの際は FIFO 列へ送り出されます。そして、時々 (例えば、500-1000ms 毎あるいは CAN232 からエラーを戻された) エラーがあるか見るために **F** コマンドを送ります。もし、ユニットにコマンドを送った後に多くのエラー応答を得た場合、バッファを空にするために 2-3 の [CR] を送信し、コマンドを再び発行し、これが続く場合、通信エラーがある (例えば、CAN トランスミッターの損傷や電源障害など) ことを、プログラム内でユーザーあるいはアプリケーションに警告します。

www.CAN232.com ウェブサイトは無料の Delphi 用 RS232 LIB のように、多くのサンプルプログラムを提供します。これらのプログラムは使用が無料、あなたのニーズに適するように変えます。

## 1.5 バージョン情報

CAN232 のバージョンナンバーは、ハードウェアに対するものとソフトウェアに対するものの2つのバージョンから構成されます。これらの2つのバージョンナンバーは、V で始まり、ハードウェアに対する2文字とソフトウェアに対する2文字の5文字で1つの固有のバージョン文字列に組合せられています。例えば、バージョン V1324 は、それがハードウェアバージョン 1.3 で、ソフトウェアバージョン 2.4 であることを示します。もし、ハードウェアを更新すると、2つの最初の文字のバージョン数を増加し、コマンドの追加変更やバグ訂正をする場合は、ソフトウェアバージョンナンバーを増加します。CAN232 がこのマニュアルのコマンドをサポートしているかどうかを見るためには、CAN232 に V コマンドを送ることによって、どのバージョンを持っているかをチェックします (下記のコマンドでそれが如何に作動するかを見ます)。このマニュアルのどのコマンドも、バージョンナンバーを載せており、そのコマンドがどのバージョンで動作するかを示しています。(すなわち、コマンドは継続的に追加され、バージョン情報はそれらのコマンドがそこから利用できることを示しています。)

## 2.0 使用可能なCAN232 ASCIIコマンド:

注: CAN232への全てのコマンドは[CR] (Ascii=13) で終わらなければなりません。  
大文字と小文字は区別されません。

**Sn[CR]** 標準CANビットレートの設定、nは0-8。  
このコマンドはCANチャンネルがクローズの時のみ作動します。

S0	10Kbit	に設定
S1	20Kbit	に設定
S2	50Kbit	に設定
S3	100Kbit	に設定
S4	125Kbit	に設定
S5	250Kbit	に設定
S6	500Kbit	に設定
S7	800Kbit	に設定
S8	1Mbit	に設定

例: S4[CR]  
CANを125Kbitに設定します。  
戻り: OKに対しCR (Ascii 13) またはERROR (エラー) に対しBELL (Ascii 7)。

**sxxyy[CR]** BTR0/BTR1によるCANビットレートの設定、xxとyyは16進数。  
このコマンドはCANチャンネルがクローズの時のみ作動します。

xx	16進数のBTR0の値
yy	16進数のBTR1の値

例: s031C[CR]  
125Kbitに等しいBTR0=0x03とBTR1=0x1CでCANを設定します。  
戻り: OKに対しCR (Ascii 13) またはERROR (エラー) に対しBELL (Ascii 7)。

**O[CR]** CANチャンネルをノーマルモード (送信および受信) でオープン。  
このコマンドはCANチャンネルがクローズの時に、直前にSまたはsコマンドが  
セットされている時のみ作動します。

例: O[CR]  
CANチャンネルをオープンします。黄LEDが点灯。  
戻り: OKに対しCR (Ascii 13) またはERROR (エラー) に対しBELL (Ascii 7)。

**L[CR]** 読み取り専用モードでCANチャンネルオープン（受信）  
このコマンドはCANチャンネルがクローズで、直前にSまたはsコマンドがセットされている  
ときのみ作動します。

注記： CANフレームを送ることはできません。(t,T,r,R)

例： L[CR]  
CANチャンネルをオープンし、黄LEDが点滅します。

戻り： OKに対しCR (Ascii 13) またはERRORに対しBELL (Ascii 7)。

**C[CR]** CANチャンネルクローズ  
このコマンドはCANチャンネルがオープンの時のみ作動します。

例： C[CR]  
CANチャンネルをクローズします。黄LEDが消灯。

戻り： OKに対しCR (Ascii 13) またはERROR(エラー)に対しBELL (Ascii 7)。

**tiiidd...[CR]** スタンダード(11bit)CANフレームの送信  
このコマンドはCAN232がノーマルモードでオープンの時のみ作動します。

iii 16進数(000-7FF)のID  
l データ長(0-8)  
dd 16進数(00-FF)のバイト値、ddの数はデータ長に合わなければなりません  
もないとエラーが発生します。

例1： t10021133[CR]  
0x11と0x33の2バイトで、ID=0x100の11ビットCANフレームを送信します。

例2： t0200[CR]  
ID=0x20と0バイトで11ビットCANフレームを送信する。

戻り： Auto Poll が非作動（デフォルト）の時、CAN232 は OK ならば CR (Ascii 13)  
を応答し、エラーならば BELL (Ascii 7)を応答します。  
Auto Poll が作動（X コマンド参照）の時、CAN232 は OK ならば z[CR]を、  
エラーならば BELL (ascii 7) を応答します。

**Tiiiiiiiidd...[CR]** 拡張(29bit)CANフレームの送信  
このコマンドはCAN232がノーマルモードでオープンの時のみ作動します。

iiiiiii 16進数(00000000-1FFFFFFF)のID  
l データ長(0-8)  
dd 16進法(00-FF)のバイト値、ddの数はデータ長に合わなくてはなりません、  
さもないとエラーが発生します。

例1 : T0000010021133[CR]  
0x11と0x33の2バイトで、ID=0x100の29ビットCANフレームを送信します。

戻り : Auto Pollが非作動(デフォルト)の時、CAN232はOKならばCR(Ascii 13)  
を応答し、エラーならばBELL(Ascii 7)を応答します。  
Auto Pollが作動している(Xコマンド参照)時、CAN232はOKならばZ[CR]  
を、エラーならばBELL(Ascii 7)を応答します。

**riiii[CR]** スタンダード(11bit)フレームのRTRを送信  
このコマンドはCANチャンネルがノーマルモードでオープンの時のみ作動します。

iii 16進のID(000-7FF)  
l データ長(0-8)

例1 : r1002[CR]  
ID=0x100, データ長2byteの11bit RTRフレームの送信

戻り : Auto Pollがオフ(デフォルト)であれば、OKに対しCR(Ascii 13)またはERROR(エ  
ラー)に対しBELL(Ascii 7)。Auto Pollがオン(Xコマンド参照)であれば、  
OKに対しz[CR]またはERRORに対しBELL(Ascii 7)。

**Riiiiiiiil[CR]** 拡張(29bit)フレームのRTRを送信  
このコマンドはCANチャンネルがノーマルモードでオープンの時のみ作動します。

iiiiiii 16進のID(00000000-1FFFFFFF)  
l データ長(0-8)

例1 : R000001002[CR]  
ID=0x100, データ長2byteの29bit RTRフレームの送信

戻り : Auto Pollがオフ(デフォルト)であれば、OKに対しCR(Ascii 13)または  
ERROR(エラー)に対しBELL(Ascii 7)。Auto Pollがオン(Xコマンド参照)  
であれば、OKに対しZ[CR]またはERRORに対しBELL(Ascii 7)。

**P[CR]** FIFO内のCANフレームのポーリング（単一ポーリング）  
このコマンドはCANチャンネルがオープンの際のみ作動します。

**注意:** このコマンドはバージョンV1220からの新しいAUTO POLL/SEND下では動作しません。  
使用されれば、BELLを応答します。

例1 : P[CR]  
FIFOから1個のCANフレームを読み出します。

戻り : OKに対し、送信フレームと同じフォーマットのCANフレームと、終わりにCR (Ascii 13)。付属フレームがない場合、CRだけを返します。CANチャンネルがオープンしていない場合は、BELL (ascii 7) を返します。もし、TIME STAMP使用の場合は、最後のデータバイト (CRの前) の後に、ミリ秒の時間を返答します。さらに多くの情報については、Zコマンドを参照下さい。

**A[CR]** FIFO内のCAN フレームのオールポーリング（全ての未読フレーム）。  
このコマンドはCANチャンネルがオープンの際のみ作動します。

**注意:** このコマンドはバージョンV1220からの新しいAUTO POLL/SEND下では作動しません。使用されれば、BELLを応答します。

例1 : A[CR]  
FIFOから全てのCANフレームを読み出します。

戻り : OKに対して、CR (Ascii 13) で分離された、送信フレームと同じフォーマットのCANフレーム。全てのフレームが読み出されると、終わりにAとCR (Ascii 13)で終わります。もし、未読フレームが全くなければ、AとCRのみを返します。もし、CANチャンネルがオープンしていなければ、BELL (Ascii 7)を返します。TIME STAMP使用の場合は、最後のデータバイト (CRの前) の後に、ミリ秒の時間を返答します。さらに多くの情報については、Zコマンドを参照下さい。

**F[CR]** ステータスフラグ読み出し  
このコマンドはCANチャンネルがオープンの際のみ作動します。

例1 : F[CR]  
ステータスフラグを読みます。

戻り : OKに対し、Fと2バイトBCD16進数値プラスCR (Ascii 13)。もしCANチャンネルがオープンしていないと、BELL (Ascii 7)を返す。このコマンドはRED Error LEDをクリアします。下の利用可能なエラーをご参照下さい。

例えば、F01 [CR]

Bit 0	受信FIFOキューがフル
Bit 1	送信FIFOキューがフル
Bit 2	エラー警告 (EI)、SJA1000データシートをご覧ください
Bit 3	データオーバーラン (DOI)、SJA1000データシートをご覧ください
Bit 4	未使用
Bit 5	エラーパッシブ (EPI)、SJA1000データシートをご覧ください
Bit 6	アービトレーションロスト (ALI)、SJA1000データシートをご覧ください*
Bit 7	バスエラー (BEI)、SJA1000データシートをご覧ください**

\*アービトレーションロストは赤LED点滅を発生しません!

\*\*バスエラーは赤LEDを連続点灯します!

### Xn[CR]

受信フレームの Auto Poll/Send オンオフ設定

このコマンドはCANチャンネルがクローズの時のみ作動します。

この値はEEPROMに保存され、次回CAN232の電源オン時、戻されます。このコマンドは、この挙動を変えたい時以外に使用すべきではありません。CAN232用に書かれた古いプログラムとの互換性のために、初期設定でOFFにセットされています。ONにセットすると、PとAコマンドを使用不能にし、tとTコマンドからの応答を変えてしまいます(応答に関する更なる情報については、これらのコマンドを参照下さい。) この機能の設定と古いポーリングからのアップグレードを強くお勧めします。これにより、CAN232に送れるCANフレームの帯域幅を節約し、送れる数を増やします。この機能がセットされれば、CANチャンネルがオープンされるや、即座に、CANフレームはRS232上へ送られます。

例1 : X0[CR]  
Auto Poll/Send 機能をオフ (デフォルト) にします。

例2 : X1[CR]  
Auto Poll/Send 機能をオンにします。

戻り : OKに対して CR (Ascii 13)、エラーに対して BELL (Ascii 7)

### Wn[CR]

フィルターモードの設定

デフォルトでフィルターモード(0)に設定し、以前のCAN232と互換になる。

これが設定されるとシングルフィルターモードが可能になります。設定はEEPROMに保存され、次のスタートアップまで記憶されます。

このコマンドはCAN232が初期化され、まだオープンしていない時のみ作動します。

例1 : W0[CR]  
デュアルフィルターモードに設定(デフォルト)

例2 : W1[CR]  
シングルフィルターモードに設定

戻り : OK に対し CR (Ascii 13) または ERROR に対し BELL (Ascii 7)。

**Mxxxxxxx[CR]** Acceptance Code Register(SJA1000のACn Register)をセット  
このコマンドは、CANチャンネルが初期化され、まだオープンしていないときのみ作動します。

xxxxxxx 16進のAcceptance Code、LSB first、AC0、AC1、AC2、AC3  
詳細はPhilips SJA1000データシートをご覧ください。

例: M00000000[CR]  
Acceptance Codeを0x00000000にセットします。  
これは電源オン時の初期設定です、すなわち、全てのフレームを受信。

戻り: OKに対しCR (Ascii 13)あるいはエラーに対しBELL (Ascii 7)。

**mxxxxxxx[CR]** Acceptance Mask Register(SJA1000のAMn Register)をセット  
このコマンドは、CANチャンネルが初期化され、まだオープンしていないときのみ作動します。

xxxxxxx 16進のAcceptance Mask、LSB first、AM0、AM1、AM2、AM3  
詳細はPhilips SJA1000をご覧ください。

例: mFFFFFFFF[CR]  
Acceptance Maskを0xFFFFFFFFにセットします。  
これは電源オン時の初期設定です、すなわち、全てのフレームを受信。

戻り: OKに対しCR (Ascii 13)あるいはエラーに対しBELL (Ascii 7)。

### Acceptance CodeとMaskレジスタの設定

Acceptance Code RegisterとAcceptance Mask Registerは一緒に作動し、それらは2グループのメッセージをフィルタリングできます。これがどのように作動するかの詳細についてはSJA1000データシートをご覧ください。11ビットIDのものでは、この方法で単一IDをフィルタリングできますが、29ビットIDのものでは、IDのグループをフィルタリングできるだけです。下の例は0x300から0x3FFの全ての11ビットIDのものをだけを受信するようにフィルターをセットします。

コマンド	コメント
M00006000[CR]	AC0=0x00, AC1=0x00, AC2=0x60, AC3=0x00
m00001FF0[CR]	AM0=0x00, AM1=0x00, AM2=0x1F, AM3=0xF0

最初のコマンドは、セットされていない場合フィルター2に対して、2ビットを合わせるように告げ、(この場合、それは0x3nn、3に相当)。第2のコマンドはnnに気かけないように告げるので、それらを0からFFまでにできます、メモリー列に良く配置されていないので、読むのがそれほど簡単ではないが、フィルター1はオフになっています(AM0、AM1とAM3の下半分)。マスクの最後のバイトも0xF0の代わりに0xE0にすることができ、RTRビットをフィルタアウトし、RTRフレームを受け入れます。

**Un[CR]**

新しいボーレートでのUARTのセットアップ、 nは0-6  
このコマンドは、CANチャンネルがクローズしている時だけ作動します。  
この数値はEEPROMに保存され、次回CAN232の電源オン時、戻されます。

U0 • セットアップ 230400 ボー (baud) (作動は保証されません)  
U1 •• セットアップ 115200 ボー  
U2 ••• セットアップ 57600 ボー (出荷時のデフォルト)  
U3 •••• セットアップ 38400 ボー  
U4 ••••• セットアップ 19200 ボー  
U5 •••••• セットアップ 9600 ボー  
U6 ••••••• セットアップ 2400 ボー

例: U1[CR]  
UARTを 115200 ボーにセットアップ

戻り: OKに対して CR (Ascii 13)、エラーに対して BELL (Ascii 7)。

上記のドットは赤と緑のLEDがパワーアップされた時、何回点滅するかを示します。これは、RS232速度がどの位で構成されたかを知る簡単な方法です。

**V[CR]**

バージョン番号の取得、CAN232のハードウェアとソフトウェアの両方  
このコマンドは常に作動しています。

例: V[CR]  
バージョン番号を得ます。

戻り: OKに対し、Vとハードウェアバージョンの2バイトBCD値と、ソフトウェアの  
2バイトBCD値プラスCR (Ascii 13)。例えば、V1013[CR]

**N[CR]**

シリアル番号の取得  
このコマンドは常に作動します。

例: N[CR]  
シリアル番号を得ます。

戻り: OKに対しNとシリアル番号の4バイト値とCR(Ascii 13)。  
例えば、NA123[CR]

シリアル番号はその中に数値とアルファベットの両方を持っていることにご注意下さい。簡単な参照のために、シリアル番号をCAN232にも印刷しますが、例えば、CAN232をプログラムで認識するために使用することができますので、プログラムはそれが正しい方法で設定されていることを知ります(パラメータはEEPROMに保存されています)。

**Zn[CR]**

受信したフレームのTime Stamp ON/OFFセット

このコマンドは、CANチャンネルがクローズしているときだけ作動します。

この値はEEPROMに保存され、次に次回、CAN232に電源を入れるまで記憶されます。このコマンドは、次に挙動を変えたい時以外に使用すべきではありません。CAN232用に書かれた古いプログラムとの互換性のために、初期設定ではOFFにセットされています。ONにセットすることは、AとPコマンドか、Auto Poll/Sendがイネーブルの時に、CAN232からの送用に4バイトを追加します。Time Stampを使用すると、各メッセージはそれがCAN232に受信された時に、ミリ秒の時間を得ます、これは例えば、メッセージ間の時間を知るためのリアルタイムアプリケーションに使用できます。しかし、送られている各メッセージに4バイトを追加するので、この特性の使用によって、多分CAN232の帯域幅を減らすことに注意して下さい。(特に、VCPドライバーで)

Time StampがOFFならば、入ってくるフレームはこのように見えます:

t10021133[CR] (ID=0x100と2バイトを持つスタンダードフレーム)

Time StampがONであれば、入ってくるフレームはこのように見えます:

t100211334D67[CR] (ID=0x100と2バイトを持つスタンダードフレーム)

最後の4バイト0x4D67は、ミリ秒(もちろん16進数)でのこの仕様のメッセージに関するTime Stampであるにご注意下さい。CAN232のタイマーはゼロ0x0000で始まり、ループする前に0xEA5Fまで行き、0x0000へ戻ります。これは正確な60,000ms(すなわち、大部分のシステムで十分以上である1分)に一致します。

例1:           Z0[CR]  
                  Time Stamp特性をOFFにします。(初期設定)

例2:           Z1[CR]  
                  Time Stamp特性をONにします。

戻り:           OKIに対しCR (Ascii 13)あるいはエラーに対しBELL (Ascii 7)

**Qn[CR]** オートスタート機能(電源オンから)

このコマンドはCANチャンネルがオープンの際のみ作動します。

この機能は、CAN速度、フィルターを設定し、これらの設定で電源オンのたびに自動的にスタートさせたいときに使用して下さい。ロギングなど、CAN232を設定する事柄が無いときに最適です。

注記： 自動送信でだけ可能 (Xコマンドを参照)、CAN フレームは、CAN 側で受信したとき RS232 に自動的に送出されます。ポーリングなしではできません。

例 1： Q0[CR]  
オートスタート機能オフ(デフォルト)  
次の電源オンで、CAN232 は通常動作で設定のコマンドを待ちます。

例 2： Q1[CR]  
オートスタート機能をノーマルモードでオン  
フィルターなどは記憶され、次回電源オンで使用されます。

例 3： Q2[CR]  
オートスタート機能を読み取り専用モードでオン  
フィルターなどは記憶され、次回電源オンで使用されます。  
t, T, r, R コマンドは使用できません。

戻り： OK に対し CR (Ascii 13) または ERROR に対し BELL (Ascii 7)。