

USB und BASIC-Tiger®

Gunther Zielosko

1. Grundlagen

Der Universal Serial Bus (USB) ist eine moderne Kommunikationsschnittstelle zunächst einmal für die PC-Welt. Seit der Einführung von Windows® 98 gehört er zur Standardausrüstung und ist dabei, die traditionellen Schnittstellen, wie serielle, parallele und PS2-Schnittstelle zu verdrängen. Ein Grund dafür ist, daß es mittlerweile für fast alle externen Komponenten eine USB-Variante gibt. Trotzdem war bis vor kurzem jeder PC mit der „neuen“ USB-Schnittstelle auch noch mit den „alten“ Ports ausgerüstet. Das ändert sich jetzt – ganz aktuelle Laptops z.B. haben teilweise schon keine serielle Schnittstelle mehr. Das sollte für die BASIC-Tiger®-Familie ein Alarmsignal sein. Was tun, wenn der neue Computer keinen Com-Port für den BASIC-Tiger® mehr hat? Der vorliegende Applikationsbericht bietet Lösungen an. Wir werden feststellen, daß der BASIC-Tiger® an der USB-Schnittstelle keinesfalls nur eine Notlösung bei fehlender COM-Schnittstelle ist, denn es ergeben sich daraus auch eine Reihe neuer Möglichkeiten. Zu den Vorteilen von USB gegenüber den traditionellen Schnittstellen am PC werden wir im folgenden Abschnitt einiges erfahren.

1.1. Die USB-Schnittstelle im Überblick

Wie schon der Name sagt, ist auch USB eine serielle Datenübertragung. Allerdings nicht zu vergleichen mit der Technik der bisherigen RS232-(COM-) Schnittstelle. Bei USB wurde an fast alles gedacht, was bei RS232 fehlt:

- Eine sehr hohe Datenübertragungsrate bis 10 MHz.
- Ein sehr einfacher und vor allem kleiner Steckverbinder mit nur 4 Anschlüssen (Bild 1 Typ-B) für die Hostseite (i.A. der PC). Es gibt noch eine zweite USB-Steckverbinder-Norm (Typ A, Bild 2), die in der Regel für Peripheriegeräte benutzt wird.



Bild 1 USB-Anschlußstecker Typ B für Host (PC) mit dem USB-Symbol



Bild 2 USB-Stecker Typ A für Peripherie-Geräte

- Die Möglichkeit, das angeschlossene Gerät automatisch mit Strom zu versorgen.

- Plug-and-Play-Fähigkeit, d.h. USB-Geräte dürfen prinzipiell bei laufendem PC und sogar bei laufendem Programm einfach eingesteckt werden, ohne daß es zu Abstürzen kommt oder gar ein neuer Bootvorgang nötig ist, um das Zubehör betreiben zu können – meist funktioniert das sogar...
- Über USB werden bereits beim Einstecken automatisch Informationen über das aktuelle Gerät an den PC geschickt, der es erkennt und dem Betriebssystem mitteilt, daß jetzt z.B. ein Scanner angeschlossen ist. Das geht sogar so weit, daß das eingesteckte Teil seinen Strombedarf meldet und dann auch „zuteilt“ bekommt.
- USB-Schnittstellen können sich sehr einfach „vermehren“. Kleine Einheiten, sogenannte HUB's, machen aus einer USB-Schnittstelle z.B. vier. So kann auch ein Laptop mit nur einer USB-Schnittstelle problemlos eine USB-Maus, einen USB-Drucker, eine USB-Digitalkamera und ... einen USB-BASIC-Tiger® betreuen.
- Es gibt mittlerweile fast keinen Gerätetyp mehr, den es nicht auch als USB-Version gibt. Neben den traditionellen Zubehörteilen, wie Maus, Drucker, Scanner, Joystick, Digitalkamera usw. gibt es auch solche, die sich bisher eher „innen“ befanden, wie Festplatten, TV- und Sound-Karten, Laufwerke für Wechselmedien (ZIP, LS120, Flashkarten aller Art) und vieles andere mehr.
- Man kann aus einer USB-Schnittstelle heute praktisch alle anderen traditionellen Schnittstellen „zaubern“, es gibt USB → RS232 – Konverter, USB → LPT – Konverter und USB → PS2 – Konverter. Leider wirkt der Zauber nicht umgekehrt, eine vorhandene COM-Schnittstelle wird nie ein USB-Port.

Zu den „Problemzonen“ der USB-Technik nur soviel:

Ein USB-Gerät mit den oben genannten Eigenschaften läßt sich nur realisieren, wenn auch das Peripheriegerät eine gehörige Intelligenz mitbringt. Es muß sich zu beim Einstecken von selbst zu erkennen geben, braucht neben der USB-Fähigkeit des PC auch spezielle gerätebezogene Software (Treiber) und vieles andere mehr. Leider ist es hier auch hardwareseitig nicht mit einem Pegelwandler-IC wie bei der RS-232-Schnittstelle getan.

Trotzdem ist zu erkennen, daß an USB einfach kein Weg mehr vorbeiführt. Welche Optionen bieten sich nun für das BASIC-Tiger®-System an, diesem Trend zu folgen? Zumindest der letzte Punkt der Aufzählung der positiven USB-Eigenschaften bietet eine einfache Lösung.

2. USB zu COM – Konverter

2.1. Fertige Lösungen – Schnittstellenwandler USB → RS232

Konverter, die aus einer USB-Schnittstelle eine RS232-Schnittstelle machen, sind die einfachste Variante, ein BASIC-Tiger®-System bei nicht vorhandenen oder „besetzten“ COM-Schnittstelle am PC zu betreiben. Betreiben heißt hier, mit Tiger-BASIC® Programme zu erstellen, zu editieren und in den Flash des BASIC-Tigers® zu laden. Kurz, für den Anwender muß alles so laufen, wie mit einer Standard-RS232-Schnittstelle. Genauso ist sicherzustellen, daß über diesen Schnittstellen-Konverter auch der Datenverkehr zwischen BASIC-Tiger® und

PC so funktioniert wie gewohnt, d.h. alle Einstellungen der Parameter auf der Seite des BASIC-Tigers® und auf der PC-Seite müssen ohne Einschränkungen gültig sein.

Solche USB → Seriell-Adapter gibt es standardmäßig in Kabelform (Bild 3). Eine Seite hat einen USB-Stecker, die andere einen meist 9-poligen SUB-D-Stecker. Auf dem PC muß Software installiert werden und, wenn alles klappt, hat man eine RS232-Schnittstelle mehr bzw. überhaupt eine. Zahlreiche Links zu Vertretern dieser Adapter-Kategorie finden Sie mit einer Suchmaschine im Internet z.B. mit „USB to serial converter“.

Soweit zu dieser Variante, die z.B. direkt mit dem Plug-and-Play-Lab zusammenarbeiten könnte. Für eigene Systeme braucht man dann natürlich BASIC-Tiger® mit eingebautem RS232-Pegelwandler oder einen solchen Baustein (MAX232 oder andere) extern.

2.2. Direkte Ankopplung USB → TTL

Warum nicht den BASIC-Tiger® (ohne eingebaute RS232-Schnittstelle) direkt, also ohne den zweimalige Pegelwandlung an USB anschließen? Diese Pegelwandlungen sind teuer, brauchen zusätzliche Energie und „bremsen“ den Datenfluß nicht unerheblich. Tolle Idee – gibt es solche Adapter? Ja, die Firma FTDI stellt Chips für alle möglichen USB-Anwendungen her, unter anderem den FT8U232AM. Dieser 7x7 mm große Baustein bringt alle Intelligenz für eine USB → Seriell-Wandlung mit (RS232, RS422, RS485), liefert TTL-Pegel, kann bei RS232 Datenraten von 300 Baud bis 920Kbaud (!) realisieren, unterstützt volles Hardware-Handshaking oder X-On/X-Off. Für uns als Bastler und Tüftler stellt Gigatechnology.com Pty Ltd ein kleines 32-poliges Entwicklungssystem mit dem FT8U232AM bereit, mit dem wir die Möglichkeiten des Bausteines einfach und unkompliziert austesten können (Bild 4).



Bild 3 typischer USB → RS232-Konverter in Kabelform

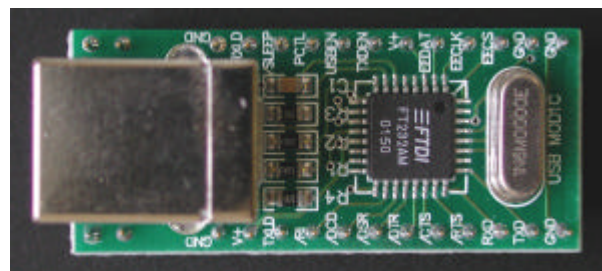


Bild 4 USB Plug and Play Serial Development Module der Firma Gigatechnology.com Pty Ltd

Informationen zum FT8U232AM <http://www.ftdichip.com/Documents/ft232r08.pdf>

Informationen über das Modul http://www.ftdichip.com/usb_mod1.htm

Auf den Internet-Seiten von FTDI finden sich viele weitere Informationen zu unserem Projekt, wie kostenlose Software (zur Bereitstellung einer „neuen“ COM-Schnittstelle usw.), Datenblätter, Distributoren und Applikationshinweise.

Alle unsere weiteren USB-Aktivitäten beziehen sich auf dieses Entwicklungssystem USBMOD1 mit dem FT8U232AM.

2.3. Das Entwicklungssystem USBMOD1 mit dem FT8U232AM

Eigentlich ist das Modul von Bild 4 wieder ein kleiner Schaltkreis mit 32 Anschlüssen im DIL-Raster 600 mil, so daß man es einfach in einen handelsüblichen Sockel stecken oder einlöten kann. Das Modul enthält alle zum Betrieb notwendigen Komponenten, weitere externe Bauteile sind nicht erforderlich. Als größtes Bauteil fällt die USB-Buchse auf, hier vom Typ A. Für eine Verbindung zum PC ist also ein Adapter-Kabel nötig, das auf einer Seite einen USB-Stecker Typ B (die meisten PC's haben Typ-B-Buchsen) und auf der anderen Seite einen vom Typ A hat. Warum aber hat das Modul so viele Pins? Es ist eben ein Entwicklungssystem – man hat Zugriff auf eine Menge von Funktionen, von denen einige notwendig und andere im Einsatzfall unwichtig sind. Hier eine Auflistung der einzelnen Pins und deren Funktionen:

RXD, TXD	sind die beiden wichtigsten Kommunikationspins für die serielle Schnittstelle, die bei SER1 am BASIC-Tiger® allein, bei SER0 mit weiteren Signalen zusammen benutzt werden.
/RXD, /TXD, /CTS	sind Signale, die nur bei SER0 gebraucht werden könnten.
/RI, /DCD, /DSR, /DTR	sind weitere Signale einer voll ausgebauten seriellen Schnittstelle, die im BASIC-Tiger®-System aber nicht verwendet werden.
TXLD, RXLD EECS, EECLK, EEDAT	sind LED-Treiber, die USB-Aktivitäten anzeigen können dienen zum Betrieb eines (optionalen) EEPROM's, zu diesen Möglichkeiten später.
PCTL	wenn dieses Pin auf Low liegt, wird das System über den USB-Anschluß mit Strom versorgt (sehr interessant!).
V+ (Ausgang!)	hier kommen 5 V (90 mA) heraus, wenn PCTL auf Low liegt und das System über USB an den PC angeschlossen ist. Mit dieser Eigenschaft kann man sich völlig neue Anwendungen ohne externe Betriebsspannung für den BASIC-Tiger® vorstellen.
SLEEP, USBEN, TXDEN GND	sind Pins zur Betriebsarten-Steuerung des Chips. Massepins.

Wir erkennen sofort, daß im einfachsten Fall nur 3 Pins für eine Kommunikation erforderlich sind (GND, RXD, TXD). Da unser FT8U232AM reine TTL-Pegel braucht und liefert, kommt als Partner nur ein BASIC-Tiger® ohne innere RS232-Schnittstelle in Betracht (ist ohnehin billiger!).

Unter Beachtung dieser Hinweise sieht die Beschaltung dann so aus:

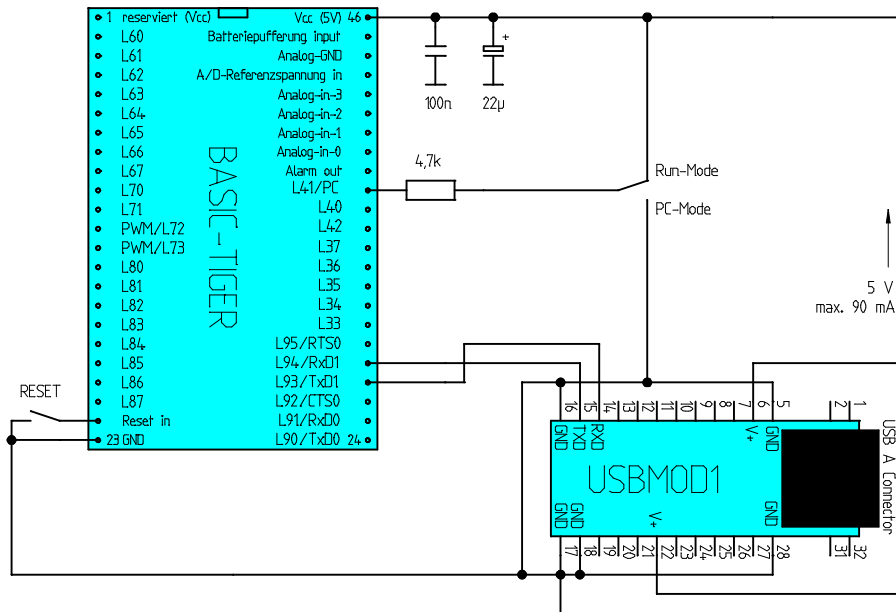


Bild 5 Einfachste Anschaltung des BASIC-Tigers® an das USB-Modul USBMOD1

Das BASIC-Tiger® System kann wie üblich benutzt werden, beachten Sie aber, daß das Plug-and-Play-Lab in jedem Falle RS232-Pegel braucht und liefert (-15 V / +15 V). Hier müssen Sie die RS232-Pegelwandler MAX232 entfernen und RXD bzw. TXD direkt an den BASIC-Tiger® anschließen. Wenn Sie die VCC-Selbstversorgung des BASIC-Tiger®-Systems über USB nicht nutzen wollen oder können (max. 90 mA!), trennen Sie die VCC-Verbindungen V+ des USBMOD1 von der übrigen VCC des BASIC-Tigers®!

3. Die Installation der Treibersoftware für das USBMOD1

Als erstes brauchen wir die Software zum Betrieb des Moduls. Über diese Internetadressen bekommen Sie Software und Installationshinweise über geeignete Treiber für Ihr Betriebssystem:

Link zu FTDI-Treibern:

<http://www.ftdichip.com/FTDriver.htm>

Installation der Treiber-Software:

<http://www.ftdichip.com/Documents/AN232-03.PDF>

Der Autor benutzte für Windows 98 den Treiber ohne PNP-Support. Laden Sie sich die entsprechende ZIP-Datei herunter und entpacken Sie diese z.B. auf eine Diskette.

Damit sind alle Vorbereitungen abgeschlossen und Sie können sich nun eine (oder eine zusätzliche) virtuelle COM-Schnittstelle einrichten.

Dank der „Power-Eigenversorgung“ des Modules ist es nach dem Auspacken und Anschließen des USB-Kabels (zur Erinnerung: Typ A nach Typ B) sofort betriebsbereit. Sie werden unter Windows 98 sofort nach dem Einstecken informiert, daß eine neue Hardware gefunden wurde. In anderen Betriebssystemen gibt es ähnliche Meldungen. Jetzt verfahren Sie so, wie es in der o.a. Beschreibung des Installationsvorganges steht. Im Resultat finden Sie nach dieser Installation im Geräte-Manager einen neuen COM-Port.

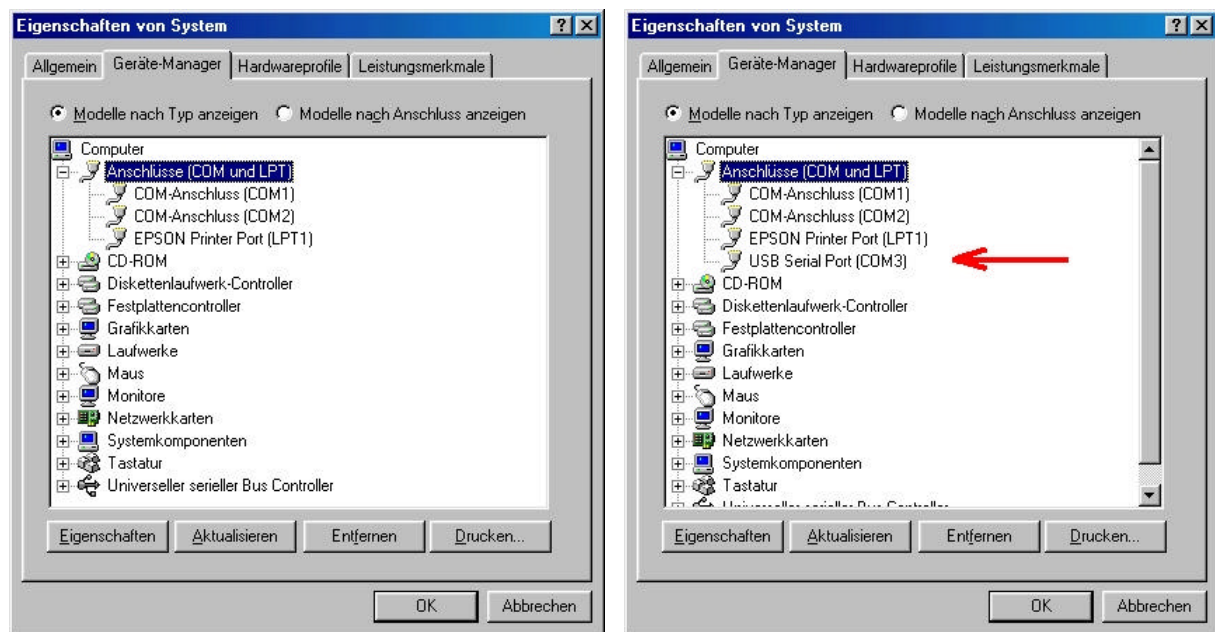


Bild 6 So sah Ihr Geräte-Manager bisher aus

Bild 7 nach dem Verbinden über USB-Kabel und dem Laden der Treiber haben Sie eine weitere COM-Schnittstelle

Toll – unser neuer COM-Port meldet sich im Geräte-Manager jedesmal beim Einstecken automatisch und verschwindet ebenso, wenn man ihn abzieht. Damit steht einer Nutzung mit dem BASIC-Tiger® nichts mehr im Wege.

4. Betrieb mit dem BASIC-Tiger®

Der BASIC-Tiger® wird zum Erstellen eines Programmes am PC generell über die SER1 mit dem PC verbunden. Das bleibt auch bei der Konvertierung zu USB so. Da wir mit dem USBMOD1 zunächst nur eine Schnittstelle bedienen können, liegt der Anschluß des BASIC-Tigers® via SER1 am USB-Modul und dann weiter über USB-Kabel an der USB-Schnittstelle des PC. Die Beschaltung nach Bild 5 ist demzufolge unmittelbar zu verwenden. Natürlich können über diese Verdrahtung später auch Daten vom PC zum BASIC-Tiger® und umgekehrt ausgetauscht werden.

Um zunächst allen Problemen mit fremder VCC und eventuellen falschen Pegeln am Plug-and-Play-Lab aus dem Wege zu gehen, ist der einfachste Weg zum Testen der neuen Technik ein eigenständiger Aufbau. Dieser enthält nur die absolut notwendigsten Komponenten des BASIC-Tigers®, diesen selbst und das USB-Modul. Alle diese Komponenten sind bereits im Bild 5 dargestellt. Eine Stromversorgung entfällt in diesem Falle – wir benutzen eine der hübschen Eigenschaften von USB, die Busversorgung von Peripheriebausteinen. Bild 8 zeigt einen einfachen Laboraufbau des Autors.



Bild 8 So einfach wird der USB-BASIC-Tiger®

Ihre Schaltung gemäß Bild 5 kann nun mit dem PC verbunden werden. Testen Sie sicherheitshalber, ob am Vcc-Pin des BASIC-Tigers® nun + 5 V anliegen.

Nun wird es spannend – wie wird unser kleines System mit der Tiger-BASIC®-Software zusammenarbeiten?

Wenn Sie die Treiber für den USB → Seriell-Wandler installiert haben, sagt Ihnen der Geräte-Manager die Nummer der neuen COM-Schnittstelle. Im Bild 7 ist das die COM3, bei Ihnen vielleicht eine andere. Starten Sie nun Tiger-BASIC® und stellen Sie über „Optionen“ und „Übertragung“ diese neue Schnittstellen-Nummer ein. Wenn Sie den Schalter auf PC-Mode gelegt und RESET betätigt haben, können Sie über „Anzeige“ und „Tiger-Status“ erfahren, ob der erste Versuch der Kommunikation geklappt hat. Wenn nein, sollten Sie nochmals prüfen, ob der PC-Mode eingeschaltet, RESET betätigt, die neue Schnittstelle ausgewählt wurde, Sie einen BASIC-Tiger® ohne interne RS232-Konverter benutzen und die Verdrahtung stimmt. Wenn ja, haben Sie schon fast gewonnen.

Laden Sie jetzt wie gewohnt das beigegefügte kleine Testprogramm SIN_TEST.TIG in den BASIC-Tiger® und lassen Sie es laufen (Run-Mode, RESET). Mit irgendeinem Terminal-Programm (neue Schnittstelle einstellen, 9600 Baud, 8 Bit, No Parity usw.) sollten Sie jetzt eine laufende Sinustabelle auf dem PC sehen.

Hurra, Wir haben jetzt einen BASIC-Tiger®, der über ein paar kleine Umwege direkt über USB kommunizieren kann!

5. Eine neue Welt

Bisher haben wir immer nur über die Anpassung einer vorhandenen seriellen Schnittstelle an die moderne USB-Schnittstelle gesprochen, sozusagen als Notbehelf. Dabei ist USB, wie wir schon erkannt haben, weit mehr als nur eine andere Kommunikationsschnittstelle. Wo liegen nun die konkreten Vorteile für uns BASIC-Tiger®-Nutzer? Die in diesem Applikationsbericht vorgestellte Technik bietet bereits:

- Eine Lösung für alle, die keine COM-Schnittstelle an ihrem PC mehr haben oder nicht mehr frei haben.
- Ein Option für einfache Module auf BASIC-Tiger®-Basis, die beim Einstecken des USB-Kabels sofort loslaufen und dabei kein eigenes Netzteil oder Batterien brauchen.
- Plug and Play – Fähigkeit eigenständiger Module dank USB. Das Modul über USB-Kabel an den PC anschließen, der innerer Power-On-Reset erspart sogar die Reset-Taste. Die Kommunikation läuft automatisch an...
- Eine hohe Übertragungsgeschwindigkeit (vom USB → Seriell-Wandler-System her 920 Kbaud). Inwiefern solche Datenraten mit dem BASIC-Tiger®-System auf dem Umweg über die SER1 und am PC erreichbar sind, muß im Einzelnen noch getestet werden.
- Keine Änderung von BASIC-Tiger®-Programmen oder im Tiger-BASIC® erforderlich, alles kann so bleiben wie bisher.
- Eine einfache Kontrolle der USB-Aktivitäten über simple LED's ohne Vorwiderstände an den Pins RXLD (Datenempfang) und TXLD (Senden von Daten) gegen Vcc ist möglich.

Schaut man sich aber die bei FTDI verfügbaren Hard- und Software-Komponenten genauer an, hat man gleich Visionen...

- Der winzige USB-Konverter FT8U232AM und ein paar passive Bauteile könnte in eine neue BASIC-Tiger®-Generation integriert werden, statt RS232-Wandler hätte diese dann USB-Wandler „inside“. Dabei könnten dann bereits vom Prozessor her höhere Datenübertragungen möglich werden (wäre es nicht ein Segen, wenn die Programmübertragung z.B. 10-mal schneller funktionierte?).
- Eine andere Alternative wäre ein Plug-and-Play-Lab mit integrierter USB-Schnittstelle. Mit Jumpfern, Schaltern oder automatisch könnte man je nach eingesetztem BASIC-Tiger® und ganz nach Bedarf wählen, welche Art der Kommunikation man haben möchte. Der zusätzliche Aufwand sind ein FT8U232AM sowie ein paar Kleinbauteile. Das macht das Plug-and-Play-Lab dann fit für das neue Jahrtausend...
- Es gibt die Option, mit dem FT8U232AM (auch am Modul USBMOD1) ein EEPROM 93C46 zu betreiben. Neben Texten (mit denen sich jedes USB-Modul meldet) lassen sich viele interessante Eigenschaften des Modules variieren. So kann z.B. die Stromabgabe der USB-Schnittstelle für die Anwendung neu festgelegt werden. Ohne EEPROM ist diese auf 90 mA begrenzt.
- FTDI bietet Treiber für nahezu alle Vorhaben an. Es ist ohne Probleme machbar, daß ein neues Tiger-BASIC® per DLL oder andere verfügbare Treiber direkt auf den USB-Baustein zugreift, also ohne den Umweg über die Installation einer eigenen Treiber-

Software. Damit würde beim Anwender z.B. neben der Auswahl der „klassischen“ COM-Schnittstellen auch direkt eine USB-Schnittstelle zur Verfügung stehen.

- Spinnt man diesen Gedanken weiter, könnte die neue verfügbare USB-Schnittstelle im BASIC-Tiger®-Menü mit einer weit höheren Geschwindigkeit „vorprogrammiert“ werden als 38 kBaud.
- Es besteht auch die Möglichkeit, zwei USB-Schnittstellen an einem BASIC-Tiger® zu realisieren. Damit wäre ein komplett auf USB zugeschnittener BASIC-Tiger® möglich.

Viel Erfolg beim Experimentieren!