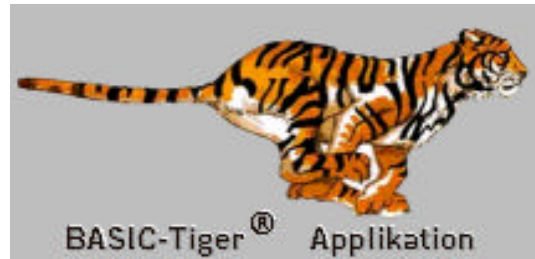


## Tiger an Excel®

Gunther Zielosko



### 1. Das Vorhaben

Im Rahmen der Reihe von Projekten, die die Zusammenarbeit von BASIC-Tiger® zum Thema haben, wollen wir im vorliegenden Applikationsbericht Daten vom Tiger direkt in eine Excel®-Tabelle schreiben. Wie immer benutzen wir dabei die serielle Schnittstelle zum Datentransfer zwischen den beiden Komponenten. Auch hier ist die Basis unserer Experimente das Programm TestPoint® der Firma Keithley, mit dem auf einfache Weise technisch orientierte Windows®-Programme erstellt und über Runtime-Module auch weitergegeben werden können, ohne daß auf dem Zielrechner TestPoint® installiert sein muß. Die Prozeduren zur Einrichtung derartiger Runtime-Module auf einem PC wurden im Applikationsbericht Nr. 31 „BASIC-Tiger® und PC-Programme“ ausführlich beschrieben und können hier analog benutzt werden. Die verwendeten Programmbezeichnungen passen wieder zueinander:

EXCEL_01.TST	das Runtime-Modul aus TestPoint®
EXCEL_01.TIG	das BASIC-Tiger®-Programm zur Erzeugung von Daten
EXCEL_01.XLS	die entstehende Excel®-Datei mit den aufgesammelten Daten

### 2. Das PC-Programm EXCEL\_01.TST

Das Programm sollte wiederum sehr übersichtlich und dennoch so universell wie möglich sein. Im Screenshot Bild 1 sind sämtliche Funktionen zu erkennen. Links die Einstellung der seriellen Schnittstelle Ihres PC, die über die Eingabefelder eingestellt werden (beim Start des Programms gibt es eine Voreinstellung, die ggf. geändert werden muß!). Mit dem Button „COM-Port setzen“ werden diese Einstellungen übernommen. Rechts erkennt man die Namen und Pfade des eigentlichen Programms EXCEL.EXE (das natürlich auf Ihrem Rechner vorhanden sein muß) und der Excel®-Datei EXCEL\_1.XLS, in die später die Daten kommen. Bei der Installation des Runtime-Programms EXCEL\_01.TST über die Datei SETUP.EXE wird ein Verzeichnis EXCEL\_01 und eine solche Datei bereits angelegt.

Wenn Sie die vorgeschlagenen Installationsparameter akzeptieren, brauchen Sie beim Start nichts mehr an dem EXCEL-System zu ändern. Durch Drücken des Buttons „Empfangen“ wird bereits die serielle Schnittstelle geöffnet und das Programm wartet auf Daten. Da es hier ein Time-Out von 100 s gibt, sollten Sie vor dem Start das Programm EXCEL\_01.TIG auf dem BASIC-Tiger® zum Laufen gebracht haben, das nun kontinuierlich die Meßwerte der 4 AD-Wandler sendet (analog zum Programm AD\_01.TIG aus dem Bericht). Wenn alles geklappt hat, wird durch den Button „Empfangen“ eine Excel®-Tabelle geöffnet, in die sofort Daten der Analog-Kanäle in die zugehörigen Spalten A, B, C, D geschrieben werden. Bei jeder neuen Messung wird eine neue Zeile geschrieben, wie in Bild 2 zu erkennen ist.

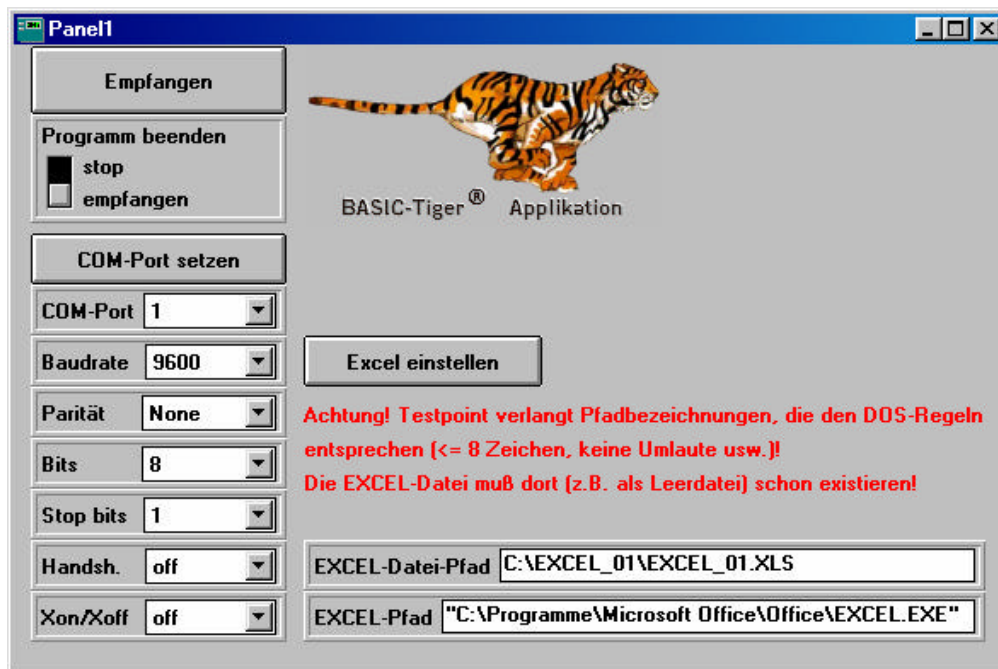


Bild 1 So sieht EXCEL\_01.TST aus

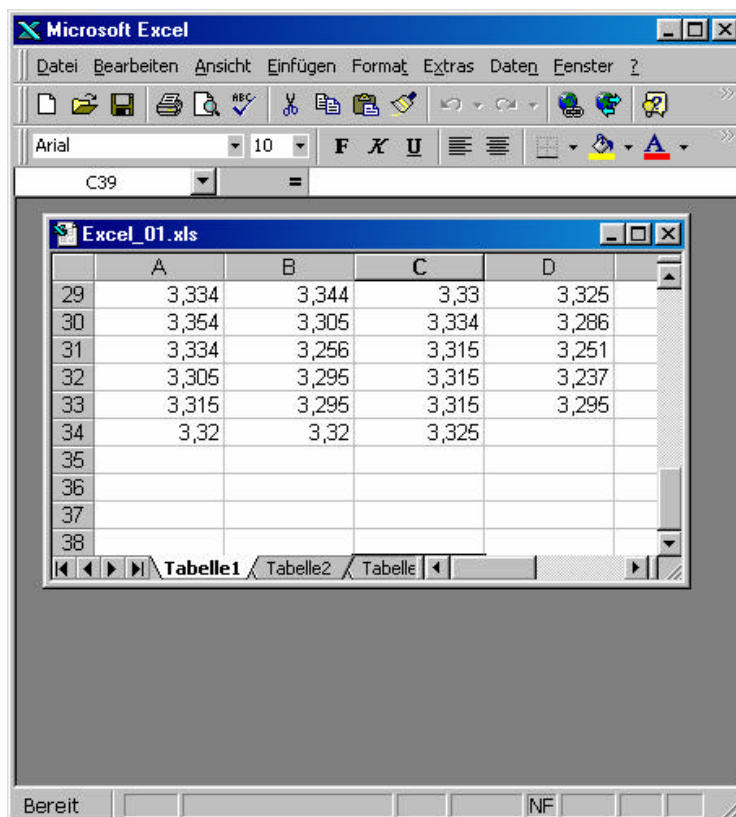


Bild 2 die Excel<sup>®</sup>-Tabelle EXCEL\_01 wird mit Daten gefüllt

Etwas knifflig sind die zeitlichen Abstimmungen. Wie schon erwähnt, wurde für die serielle Schnittstelle am PC ein Time-Out von 100 Sekunden eingestellt. Das macht Sinn, wenn trotz einer versehentlich falsch konfigurierten COM-Schnittstelle der Rechner irgendwann wieder einmal funktionieren soll. Ohne Time-Out würde er nämlich ewig auf ein Ereignis am COM-Port warten. Andererseits kann es bei zu lang eingestellten „Sendepausen“ (im BASIC-Tiger®-Programm) zur Fehlermeldung „Time-Out“ kommen. Stellen Sie also die Intervalle maximal so ein, daß Time-Out nicht überschritten wird. Zu kurze Meßfolgen werden ebenfalls nicht toleriert, da das Programm EXCEL\_01.TST auch hier eine Zwangspause von ca. 100 ms braucht. Diese Kompromisse wurden in dieser Applikation eingegangen, um ohne bidirektionalen Betrieb der seriellen Schnittstelle auszukommen. Solchen Fragen werden wir uns in einer der kommenden Anwendungen stellen.

### **3. Das BASIC-Tiger®-Programm EXCEL\_01.TIG**

Das Programm ist fast identisch mit dem bereits bekannten Programm AD\_01.TIG. Lediglich die Formatierung der Daten erfolgt etwas anders, da Excel® in der deutschen Version mit Punkten als Dezimalzeichen nichts anzufangen weiß – deshalb nehmen wir hier Kommas. Auch das Timing ist etwas anders, es geht etwas schneller, damit wir in der Excel®-Tabelle auch einen Fortschritt erkennen. Ansonsten nichts wirklich Neues...

### **4. Betrieb**

Es ist alles ganz einfach. Verbinden Sie BASIC-Tiger® und PC, laden Sie EXCEL\_01.TIG in den BASIC-Tiger® und EXCEL\_01.TST in den PC (wie's geht, wissen Sie ja...) und rufen sie letzteres auf. Stellen sie ggf. die Excel®-Verzeichnisse ein (Abschluß mit „Excel einstellen“), wählen Sie die COM-Schnittstelle am PC aus, am besten die, mit der Sie eben den BASIC-Tiger® programmiert haben (vergessen Sie aber nicht, danach Tiger-Basic zu beenden, da dies sonst die COM-Schnittstelle belegt!). Stellen Sie die Nummer des COM-Ports unter EXCEL\_01.TST ein und drücken Sie abschließend auf „Empfangen“. Wenn es nach 100 s immer noch keinen Kontakt zwischen beiden Rechnern gibt, kommt es zur o.a. Fehlermeldung. Wenn Sie Erfolg haben, wird Ihre erste Excel®-Tabelle gerade vom BASIC-Tiger® ausgefüllt. Damit Sie den ganzen Vorgang besser beobachten können, sollten Sie die Fenstergröße der Excel®-Tabelle neu einrichten. Sonst ist es möglich, daß Sie nicht alle Spalten sehen oder, wenn Excel® zu groß oder im Vollbild eingestellt ist, alles andere verdeckt wird. Außerdem brauchen Sie ja später noch die Bedienelemente von EXCEL\_01.TST...

Haben Sie irgendwann genug Daten und wollen das Programm beenden, schalten Sie den Schalter „Programm beenden“ auf „stop“. Damit wird zwar nicht das Programm beendet, aber die Datenübertragung gestoppt. Das Programm selbst beenden Sie, indem Sie den Button mit dem Kreuz rechts oben drücken. Die Excel®-Tabelle EXCEL\_01.XLS bleibt uns auch nach dem Programmende erhalten und kann natürlich unter einem anderen Namen irgendwohin abgespeichert werden.

Wenn Sie einen Neustart über „Empfangen“ versuchen, wird eine neue Excel®-Tabelle aufgerufen – das ist nicht so günstig. Besser ist es, das Programm ganz zu verlassen und wieder neu zu starten, was je sehr schnell geht.

Und wieder haben wir ein nützliches Tool für die Anwendung des BASIC-Tigers®. Vielleicht auch nur eine Anregung für eigene Entwicklungen mit TestPoint® ...