

Black Box für Kraftfahrer

Gunther Zielosko

1. Warum wir einen neuen GPS-Logger bauen

In den Berichten 063 und 064 haben wir uns bereits ausführlich mit dem Thema GPS-Logger im Auto beschäftigt. Warum also erneut darüber nachdenken?

Also – ein neues Auto des Autors erforderte zunächst eine neue mechanische Konstruktion. Dabei fiel auf, dass die bisherigen Lösungen einige Nachteile hatten, die wir nun mit besseren technischen Möglichkeiten beheben wollen.

Als erstes benutzen wir jetzt einen moderneren GPS-Empfänger, der deutlich weniger Strom benötigt, mit 16 Kanälen genauer arbeitet und viel besser an unsere Aufgabenstellung angepasst ist.

Zweitens werden wir die veraltete SmartMedia-Karte (sie ist kaum noch erhältlich!) aus dem Bericht 064 zu den Akten legen und die moderneren SD-Karten als Speichermedium benutzen. Damit erhöhen wir die Kapazität um ein Vielfaches (SmartMedia-Karte maximal 128 MB, SD-Karten hier bis 2GB nutzbar!). Ein weiterer Vorteil ist, dass sich die Ansteuerung des gewählten SD-Adapters wesentlich gegenüber dem SmartMedia-Adapter vereinfacht.

Drittens werden wir das Konzept so optimieren, dass nur noch ein preisgünstigerer und kleinerer Tiny-Tiger-Economy erforderlich ist.

Als weitere Verbesserung werden wir eine Schaltung benutzen, die nur noch eine Betriebsspannung benötigt, nämlich die an Zigarettenanzünder-Steckdose übliche Spannung, die sich erst in der Stellung Zündung einschaltet. Damit entfällt der bisher notwendige Anschluss „Dauerplus“.

Insgesamt werden wir bei verbesserten Eigenschaften unseres neuen GPS-Loggers also Kosten, Energie und Platz sparen – doch wohl ein Anreiz zur Neukonstruktion.

2. Die neue GPS-Mouse LC-N-102

Im Projekt Bericht 064 benutzten wir eine LC-GPS-Mouse LC-HI-203 von Landolt Computer in Maintal. Nun steigen wir um auf die LC-GPS-Mouse LC-N-102:

http://www.landolt.de/info/lc-gps_n102.htm

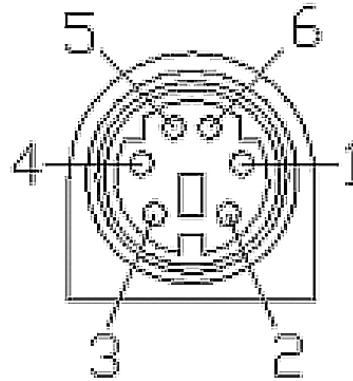


Bild 1 LC-N-102 im Größenvergleich mit Economy-Tiger (im Bild die ähnliche Mouse LC-N-101)

Bild 2 Als Standardanschluss wird hier im Gegensatz zur LC-HI-203 eine PS2-Buchse verwendet

Die Vorteile gegenüber der bisherigen Variante in Kürze:

	LC-GPS-Mouse LC-HI-203	LC-GPS-Mouse LC-N-102
Kanäle	12	16
Stromaufnahme	Ca. 115 mA	Ca. 35 mA
Datenausgabe seriell	Nur RS232	RS232 und TTL gleichzeitig
Maße	58x40x20	57x55x20

Anders als die LC-GPS-Mouse LC-HI-203 liefert die LC-GPS-Mouse LC-N-102 also ihre Daten gleichzeitig sowohl im RS232- und im TTL-Format. Letzteres ist natürlich optimal für den Betrieb mit einem kleinen Tiger, der ohne externen Pegelwandler ebenfalls nur das TTL-Format versteht. Das vereinfacht die Zusammenschaltung beider Teile erheblich.

Die bei dieser GPS-Maus automatisch vorhandene zusätzliche echte RS232-Ausgabe kann man aber sehr gut für weitere Geräte nutzen, z.B. ein Navigationssystem, Laptop mit Karten-Software usw.

3. Der SD-Card Adapter 1 von Wilke Technology

Auf das moderne Speichermedium SD-Karte hat Wilke Technology mit sogar zwei Adaptern für das BASIC-Tiger-System reagiert:

Der SD-Card Adapter 1 wurde entwickelt, um ältere Projekte, die mit dem SmartMedia-Adapter arbeiteten, nahezu ohne Aufwand in Projekte mit SD-Karten zu überführen.

Der SD-Card Adapter 2 ist hard- und softwareseitig eine separate Lösung, um auch die Ansteuerung von Grafik-Displays zu ermöglichen. Dies ist beim SD-Card Adapter 1 nicht möglich.

In unserem aktuellen Projekt wählen wir den SD-Card Adapter 1, um Hardware und Software aus dem Bericht 064 nicht komplett neu entwickeln zu müssen. Obwohl wir wegen der vielen neuen Eigenschaften unseres GPS-Loggers einiges zu ändern haben, wird im Laufe der Beschreibung deutlich, wie einfach die Umsetzung von SmartMedia auf SecureDigital sein kann.

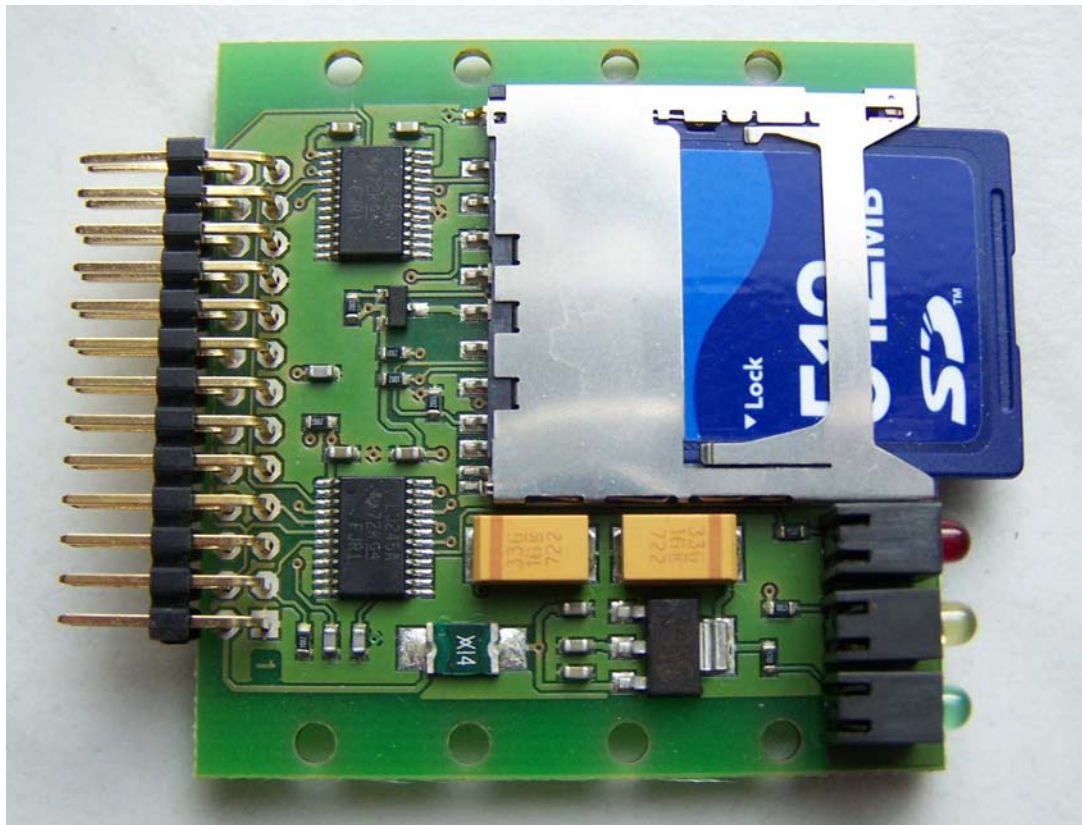


Bild 3 der SD-Card Adapter 1

Der SD-Card Adapter 1 hat äußerlich gleiche Abmessungen, Befestigungs-Bohrungen und den gleichen 26-poligen Pfostenstecker. Das Geniale an diesem Steckverbinder ist, wie wir später sehen werden, dass auch die Verbindungen zum Tiger praktisch identisch wie beim SMC-Adapter liegen. Hinweis: Es funktioniert nur die neueste Hardware-Version PCB 06. 2100.V1.4 dieses Adapters, die vorherigen Versionen 1.2 und 1.3 konnten vom Autor nur durch einen PullDown Widerstand von 10k Ohm am DataOut Pin der SD-Card-Aufnahme zum Laufen gebracht werden. Die Versionsnummer findet man auf der Bauelemente-Seite direkt vor dem Kartenslot.

4. Das Konzept

Wie im Bericht 064 soll unser KFZ-GPS-Logger „vollautomatisch“ arbeiten, d.h. der Fahrer kümmert sich nicht um das Aus- und Einschalten, im Betrieb sind keine Aktivitäten erforderlich und das System kann vollkommen unauffällig irgendwo im Auto eingebaut werden. Lediglich die GPS-Mouse muss einigermaßen waagrecht angeordnet werden und einen möglichst freien „Blick“ zum Himmel haben, d.h. Blech und metallisierte Scheiben (Sonnenschutz) schränken die Funktion eist erheblich ein.

Alle paar Monate im normalen KFZ-Betrieb oder bei einem besonderen Vorkommnis werden die SD-Karte entnommen, die Daten auf den heimischen PC übertragen, die SD-Karte wieder frei gemacht und die Daten ggf. ausgewertet.

Die Tiger-Software wird im Betriebsfall (Zündung eingeschaltet, GPS-Empfänger angeschlossen und gültige Daten vorhanden) auf der SD-Karte ein Verzeichnis für jeden Tag anlegen, das Datum wird aus dem GPS-Datum ermittelt:

Verzeichnisname: JJMMTT Jahr, Monat, Tag

Dann wird eine Datei in diesem Verzeichnis erzeugt, die ihren Namen aus der GPS-Zeit ableitet (Startzeit der Aufzeichnung, Achtung – das ist die so genannte UTC-Zeit und nicht die Lokalzeit!):

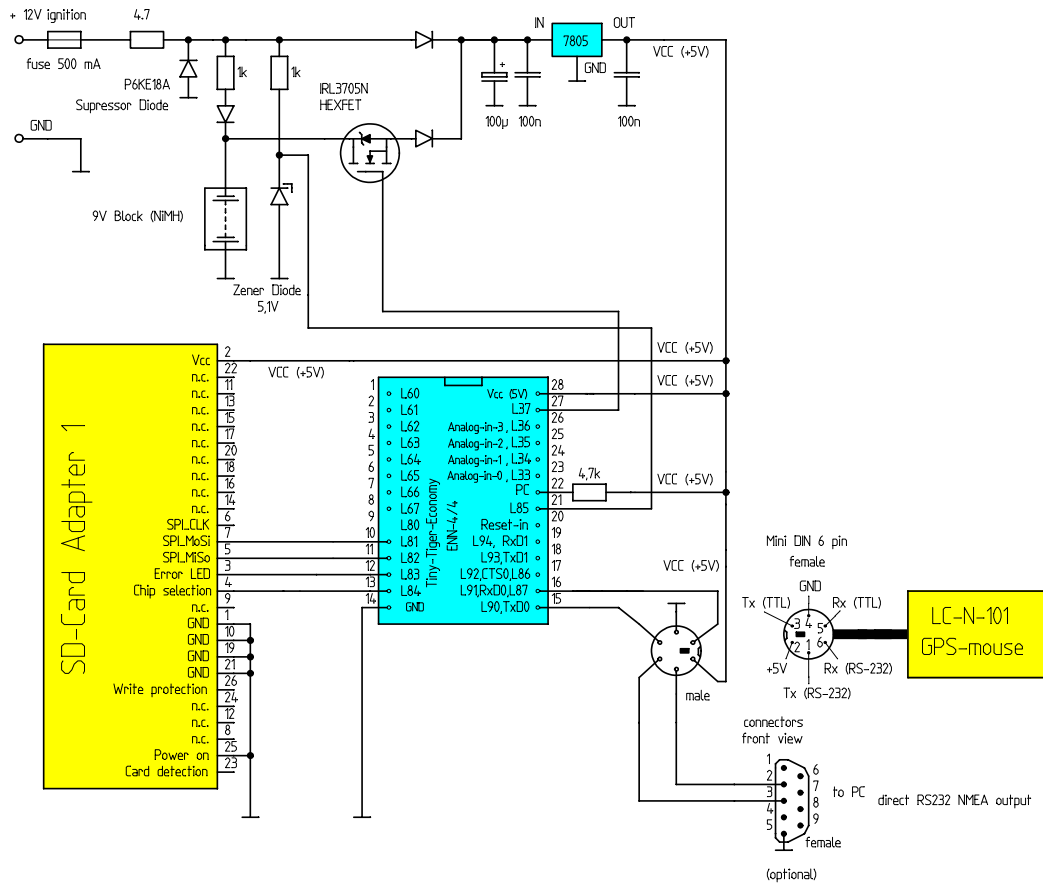
Dateiname: hhmmss.txt Stunde, Minute, Sekunde

Damit wird jedesmal unverwechselbar eine Datei auf der SD-Karte erzeugt, die später als normale Textdatei mit NMEA-Format gelesen und mit vielen Programmen komfortabel ausgewertet werden kann.

Dabei ist für den BASIC-Tiger-Programmierer die Auswahl der NMEA-Protokolle weitgehend freizügig, d.h. im Programm können sehr einfach NMEA-Datensätze zusätzlich ausgewählt oder weggelassen werden.

5. Die komplette Schaltung

Bevor wir zu Details der Schaltung und deren Funktion kommen, hier zunächst das Gesamtschaltbild:



GPS-Logger mit SD-Card-Adapter 1 und LC-N-101 GPS Mouse

Bild 4 die komplette Schaltung

Die obige Pin-Konfiguration ist die Standardbelegung für den SD-Card Adapter 1. Leider gibt es in der Dokumentation zum SD-Card Adapter von Wilke Technology einige Unstimmigkeiten, die schnell zu Irritationen führen können. So wurde z.B. die Standard-Pinzuordnung in zwei Dokumenten unterschiedlich angegeben (Bilder 5 und 6). Nur die Definition aus der In der Memory Cards – Tiger Basic API entsprechen der Zielstellung, genau dieselben Verbindungen zu benutzen wie beim SmartMedia-Adapter. Natürlich kann man die Pin-Konfiguration in der Software oder durch Löten ändern, besser wäre aber eine durchgängig einheitliche Dokumentation.

SD-Card Adapter 1

Version of Product 1.4

6 Used Tiger Pins

Pin No	direction	signal	description
L80	output	SPI_clk	SPI clock line
L81	input	SPI_MiSo	SPI data in
L82	output	SPI_MoSi	SPI data out
L83	output	Err_LED	Sign a error
L84		not use	-
L85		not use	-
L86	output	OE	Chip selection
L87		not use	-

Note: This table is only true for using our BTI-ADAP-xxSMC Interface! You can do your own pin mapping for your application.

Memory Cards – Tiger Basic API

```

.....
'
' Pin Description for SD Card Adapter 1
'
'.....
#define SD_ADAPTER1
' SPI configuration
#define SD_SPI_PORT 8
' MISO (Tiger - In, Card - Out)
#define SD_SPI_DATA_IN_PIN 2
' MoSI (Tiger - Out, Card - In)
#define SD_SPI_DATA_OUT_PIN 1
' clock
#define SD_SPI_CLOCK_PIN 0
'
' Features controlled over X-Port
' X-Port Address for Control Lines
#define SD_XP_ADDRESS 00F8h
' Offset to X-Port Address for Power Line (if not necessary, must be set to
zero)
#define SD_XP_POWER_OFFSET 1
'
#define SD_XP_ADDRESS
' Bit of X-Port for Card Detect Line
#define SD_XP_SDCARD_DETECT 2
' Bit of X-Port for Write Protect Line
#define SD_XP_WRITE_PROTECT 4
' Bit of X-Port for Power Line
#define SD_XP_POWER 6
' Bit of X-Port for Error Indication Line (unused in this configuration)
' #define SD_XP_ERROR_INDICATOR 1
' Bit of X-Port for Chip Select Line (unused in this configuration)
' #define SD_XP_CHIP_SELECT 0
#endif ' SD_XP_ADDRESS
'
' Features controlled over Tiger-Pins
' Port Address for Control Lines
#define SD_I_PORT 8
#define SD_I_PORT
' Pin for Card Detect Line (unused in this configuration)
' #define SD_I_SDCARD_DETECT 2
' Pin for Write Protect Line (unused in this configuration)
' #define SD_I_WRITE_PROTECT 4
' Pin for Power Line (unused in this configuration)
' #define SD_I_POWER 6
.....

```

Documentation

Bild 5 Im Datenblatt liegen:

L81 an SPI_MiSo
L82 an SPI_MoSi

Bild 6 In der Memory Cards – Tiger Basic API liegen:

L81 an SPI_MoSi
L82 an SPI_MiSo

Ähnliche Irritationen ergeben sich bei Pin 25 des SD-Card Adapter 1. Ursprünglich (beim SMC-Adapter) wird hier standardmäßig das Pin L86 zum Steuern der SMC_LED (LED leuchtet beim Kartenzugriff) benutzt. In der Dokumentation zum SD Card Adapter 1 sollte deutlich gemacht werden, dass dieses Pin 25 nun ständig an Masse zu liegen hat. Jetzt ist nämlich Pin 25 (Power on) kein simples Signal mehr, sondern aktiviert die Funktion des Adapters überhaupt nur bei Low (dauernd GND oder logisches Signal).

6. Zur Schaltungsfunktion

Grundlage für die Spannungsversorgung ist das KFZ-Bordnetz (z.B. an der Buchse für den Zigarettenanzünder). In der Regel liegt dieser Punkt nur auf +12 V, wenn die Zündung eingeschaltet ist, wovon wir hier ausgehen. Wird die Zündung eingeschaltet, gelangt die Bordspannung über die Sicherung 500 mA und einen Schutzwiderstand 4,7 Ω an eine Supressor-Diode (18 V), die unsere Elektronik vor hohen Spannungsspitzen schützt.

Über 1 kΩ und eine Diode wird ein 9 V Blockakku (NiMH wegen Memory-Effekt) mit wenigen mA aufgeladen, solange das Auto fährt. Der Ladestrom sollte nicht höher als der maximale Strom bei Erhaltungsladung gewählt werden, um den Akku nicht zu überladen. Der Widerstand kann je nach Fahrzeugbenutzung

angepasst werden, wird das Auto häufig über längere Strecken benutzt, kann man den Ladestrom kleiner einstellen. Bei nur kurzen Fahrstrecken und seltenen Fahrten ist ein etwas höherer Strom zu empfehlen, um den Akku genügend zu laden.

Gleichzeitig wird an dieser Stelle mit einem Vorwiderstand 1 k Ω und einer Zener-Diode von 5,1 V ein logisches Signal für Pin L85 des Economy-Tigers abgeleitet (Erkennung Zündung ein/aus).

Der 5 V – Spannungsregler wird im Betrieb von der Bordspannung über eine Diode gespeist und erzeugt $V_{cc} = 5\text{ V}$ für die gesamte Schaltung. Nach Abschaltung der Bordspannung (Zündung aus) müssen aber noch restliche Daten auf die SD-Karte geschrieben und die aktive Datei einschließlich Ordner ordnungsgemäß geschlossen werden. Das macht erforderlich, dass V_{cc} noch für einige Sekunden stehen bleibt. Für diese Aufgabe haben wir die schon erwähnte 9 V – Blockbatterie vorgesehen. Diese übernimmt, ebenfalls über eine Diode entkoppelt, kurzzeitig die Stromversorgung der Schaltung. Schließlich dient der HEXFET IRL3705N zur endgültigen Abschaltung der Betriebsspannung für die gesamte Elektronik. Dieser Leistungs-FET kann durch normale Logiksignale entweder sehr niederohmig (High am Gate) oder sehr hochohmig (Low am Gate) werden, was direkt durch Pin L37 des Economy-Tigers gesteuert wird.

Die Anschaltung des GPS-Empfängers ist simpel. Bei Verwendung anderer Empfänger mit echter RS232-Schnittstelle muss darauf geachtet werden, dass noch ein MAX232 zur Pegelwandlung erforderlich wird. Bei höherer Stromaufnahme des Empfängers muss ggf. ein anderes Stromversorgungskonzept (z.B. Schaltregler oder eine leistungsfähige Kühlung) verwendet werden.

Die Anbindung des SD-Card Adapter 1 ist denkbar einfach, der Aufwand gegenüber der SmartMedia-Variante ist deutlich geringer.

7. Software

Bei den vielen Ergänzungen, Verbesserungen und Anpassungen des Tiger-Basic in den letzten Jahren bleiben Kompatibilitätsprobleme nicht aus. Um den modernen SD-Card Adapter 1 nutzen zu können, sind einige neuere Include-Dateien erforderlich. Leider genügt es dabei nicht, diese Dateien einfach in das Tiger-Basic-System einzubinden. Für unsere Anwendung müssen diese Dateien teilweise angepasst werden:

1. folgende INC-Dateien wurden vom Autor in den Include-Ordner eingefügt, benutzt und erprobt (hier als Kopie einer ZIP- Datei):

Name	Größe	Gepackte Größe	Geändert am
fs_coinc.INC	3516 B	1377 B	2008-10-10 14:42
fs_fat_d.inc	11 K	2990 B	2009-06-23 16:24
fs_fat_j.inc	83 K	17 K	2009-05-12 20:03
fs_dat_j.inc	19 K	4945 B	2009-04-22 17:02
SD_CARD_SPI_D.INC	5910 B	2171 B	2008-10-21 10:52
fs_smc_d.inc	6738 B	2075 B	2008-10-21 10:51
fs_smc_j.inc	48 K	10 K	2008-10-21 10:50
fs_hal_d.INC	6067 B	1692 B	2008-10-21 10:50
SD_CARD_SPI_I.INC	45 K	8830 B	2008-10-10 14:42
sd_card_pins_d.inc	9588 B	1476 B	2008-10-10 14:42
fs_inx_j.inc	42 K	8485 B	2008-10-10 14:42
fs_inx_d.inc	4675 B	1663 B	2008-10-10 14:42
fs_fmt_j.INC	9 K	2470 B	2008-10-10 14:42
fs_fat_sdcard_j.inc	9 K	3011 B	2008-10-10 14:42
fs_fat_sdcard_d.inc	1843 B	748 B	2008-10-10 14:42
fs_ecc_j.INC	8231 B	2381 B	2008-10-10 14:42
fs_dat_d.INC	1853 B	780 B	2008-10-10 14:42
fs_conf.INC	2646 B	1137 B	2008-10-10 14:42

2. In der Datei sd_card_pins_d.inc wurden folgende Veränderungen vorgenommen:

Die gelb gekennzeichneten Definitionen wurden auskommentiert.

```

'' Features controlled over X-Port
'' X-Port Address for Control Lines
#define SD_XP_ADDRESS 00F8h
'' Offset to X-Port Address for Power Line (if not necessary, must be zero)
#define SD_XP_POWER_OFFSET 1

#ifdef SD_XP_ADDRESS
'' Bit of X-Port for Card Detect Line
#define SD_XP_SDCARD_DETECT 2
'' Bit of X-Port for Write Protect Line
#define SD_XP_WRITE_PROTECT 4
'' Bit of X-Port for Power Line
#define SD_XP_POWER 6
'' Bit of X-Port for Error Indication Line (unused in this configuration)
'' #define SD_XP_ERROR_INDICATOR 1
'' Bit of X-Port for Chip Select Line (unused in this configuration)
'' #define SD_XP_CHIP_SELECT 0
#endif '' SD_XP_ADDRESS
    
```

Die gelb gekennzeichnete Definition wurde von „6“ auf „4“ geändert, um problemlos einen Economy-Tiger einsetzen zu können.

```

'' Features controlled over Tiger-Pins
'' Port Address for Control Lines
#define SD_I_PORT 8
#ifdef SD_I_PORT
'' Pin for Card Detect Line (unused in this configuration)
    
```



```
' #define SD_I_SDCARD_DETECT          2
' Pin for Write Protect Line (unused in this configuration)
' #define SD_I_WRITE_PROTECT         4
' Pin for Power Line (unused in this configuration)
' #define SD_I_POWER                 7
' Pin for Error Indication Line
#define SD_I_ERROR_INDICATOR          3
' Pin for Chip Select Line
#define SD_I_CHIP_SELECT              4          vorher 6
#endif ' SD_I_PORT
```

3. Die Datei „fs_coinc.inc“ sollte bei der Verwendung des SD Card Adapters 1 am Anfang (nach dem umfangreichen Kommentar-Block) so aussehen

```
#ifndef __FS_COINC_INC
#define __FS_COINC_INC

' Activate SD/MMC Card Support
#define SD_CARD
' Activate Smart Media Card Support
' #define SMC_CARD
```

4. Die ansonsten erforderlichen Definitionen und Programmdetails finden Sie im kommentierten Programm „GMOUSE_SD02.TIG“ selbst. Interessant ist, dass alle aus früheren Programmteilen übernommenen Zuordnungen zum SMC-Adapter unverändert übernommen werden konnten. Die Anweisung: **#define SD_ADAPTER1** leitet alle diese Zuordnungen nun hardware- und softwaremäßig auf den SD Card Adapter 1 um. Ein genialer Trick von Wilke Technology...

8. Auswertungen

Die Software im Economy-Tiger erzeugt echte NMEA-Datensätze, die ganz nach Bedarf noch angepasst werden können. Zur Zeit werden die Datensätze: \$GPRMC \$GPGGA aufgezeichnet.

Somit können die meisten PC-Programme, die NMEA-Daten benötigen, mit diesen Daten etwas anfangen. Man kann z.B. mit Fugawi sofort die gefahrenen Strecken auf der Karte ansehen und auswerten. Es gibt weiterhin eine Vielzahl von kleineren Programmen, die die NMEA-Daten nach verschiedenen Kriterien auswerten oder umsetzen können (z.B. in KML-Daten für Google Earth). Genauso interessant könnten die Textdateien selbst sein. Wer die NMEA-Daten interpretieren kann, liest problemlos Datum, Uhrzeit, Position, Höhe, Geschwindigkeit usw. direkt aus.

Ein anderer Weg ist ebenfalls möglich. Genauso wie wir jetzt die Daten einer GPS-Mouse mit einem Tiger auf SD-Karte schreiben, könnte ein Tiger-Programm diese Daten umgekehrt aus dem gespeicherten NMEA-File auslesen und

zeitgenau über RS232 an einen PC senden. Damit wird jede Autofahrt später in Echtzeit nachvollziehbar.

Viel Spaß bei Nachbau und Einsatz der „Black Box“!