SD-Modul mit AVR

SD-Karte am Z9001 - die Lösung von Kingstener 2012

Dieses kleine Zusatzgerät erlaubt das Laden und Speichern auf SD-Karte. Es sind keine Änderungen am Z9001 nötig!



Version mit Micro-SD-Card und ATMega328p, links mit Gehäuse (Bilder: Kingstener)



Anschluss Tape-Eingang

2024: unter arduino_nano gibt es eine Nachbau-Variante mit einem Arduino Nano.

Von Kingstener habe ich ein fertig aufgebautes SD-CARD-Interface für den Z9001 bekommen.

Benötigt wird i.W. nur ein Mikrocontroller ATMEGA32 und eine SD-CARD-Fassung (s. Bild). Die Hardware wird am User-Port und anstelle des Kassettenrecorders am Tape-Eingang angeschlossen und fungiert als digitaler Kassettenrecorder. Am Z9001 muss nichts verändert werden! Nach RESET wird automatisch ein kleines Terminalprogramm an den Tape-Eingang gesendet, was auf dem Z9001 gestartet wird und die weitere Kommunikation mit der SD-Karte über den USER-Port übernimmt. Im Video ist der Startvorgang bis hin zum Start eines BASIC-Programms zu sehen:

sd-modul-87.mp4

Downloads

- sd2kc.zip Software
- plaenesd2kc.zip Schaltung, siehe auch unten!

- msd2kc87_ki_2013.zip

komplette Unterlagen von kingstener 2013, ehem. http://www.kingsteners.homepage.t-online.de/, Seite ist seit 2023 nicht mehr online.

- msd2kc87_nano.zip

orig. von kingstener Arduino Nano -Anpassung

Das Terminalprogramm SDX

Das Terminalprogramm ist nur wenige Bytes groß. Die ganze Funktionalität steckt im Mikrocontroller.



links: Kommando-Übersicht, rechts: Verzeichniswechsel, Directory listen, Programm laden

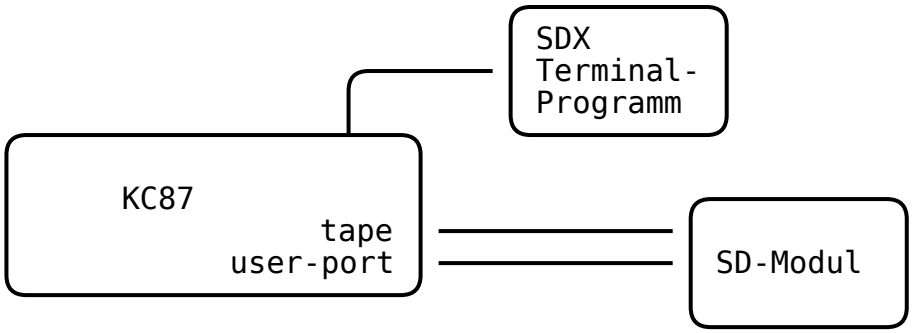
Das Terminal-Programm SDX steuert die Kommunikation zwischen SD-Modul und Computer.

Die SD-Karte (keine SDHC!) muss FAT12/FAT16-formatiert sein. Auf ihr sind die Programme „load87“, ggf. „SD2KC.BIN“ und beliebige Z9001-Programme im TAP-Format abgelegt. Unterverzeichnisse sind natürlich möglich. Das Modul wird am Userport und am Kassettenrekorderport angeschlossen.

Nach Einschalten des Z9001 erhält auch das Modul Spannung vom Z9001 und es wird das Programm „load87“ von der SD-Karte an den Kassettenanschluss ausgegeben. Am Z9001 / KC87 tippt man „SDX“ + Enter. Es erscheint „tape load“. Man drückt erneut Enter und das Programm wird geladen, indem die Datei „load87“ über den tape-Anschluss als Audiosignal gesendet wird. Ist man zu langsam und es erscheint „bad record“, muss man am Modul Reset drücken und es halt nochmal versuchen.

Das Programm SDX - das Terminalprogramm - startet automatisch. Das SD-Modul meldet sich mit „SD>“ und ist bereit für ein Kommando. Mit H gibt es eine Hilfe, mit L programmname wird ein Programm eingelesen. So einfach ist das!

Im Megamodul ist die Terminalsoftware SDX mit enthalten. Damit kann bei Nutzung des Megamoduls das Audiokabel auch komplett entfallen. Auch das Programm „load87“ muss nicht auf der SD-Karte vorhanden sein.



SD-OS

Januar 2025 habe ich neue Software programmiert.
Das System SD2KC.BIN Version 1.4k oder neuer ist voll kompatibel zu älteren Versionen. Ein SD-Modul mit dieser Firmware kann normal wie bisher mit SDX genutzt werden.

Neu sind:

- Unterstützung des KCC-Formats anstelle des TAP-Formats

TAP-Dateien werden dennoch weiterhin unterstützt! Sie können mit L geladen und mit

```
T gespeichert werden.
* Auch die Load-Datei "load87" kann nun im KCC-Format abgelegt werden
* Zusätzliche Kommandos zur Fremdsteuerung. Codes >= 0xF0 werden speziell
behandelt\\
s. Beschreibung SD.txt
```

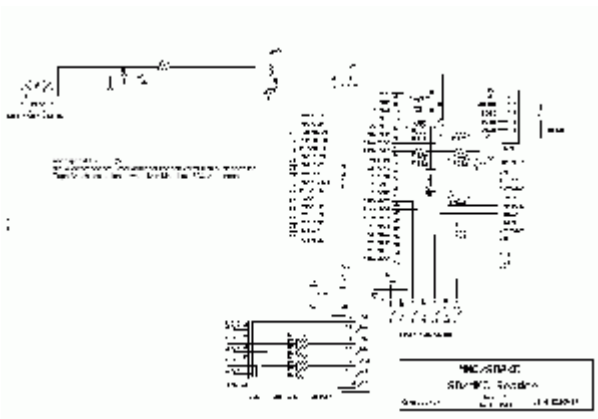
Mit dieser neuen Firmware und der KC-seitigen Software SD-OS ist ein nativer Betrieb des KC möglich, bei dem die normalen Load- und Save-Routinen der einzelnen Programme genutzt werden. Ein alternatives Laden/Speichern via Terminalprogramm SDX ist nicht nötig.

s. Beschreibung SD-OS.pdf

Hardware

Kingsteners Lösung baut auf dem SD-Interface von Arne Fitzenreiter http://www.fitzenreiter.de/kc85/kc85cf.htm auf; diese wiederum basiert auf dem SD-AVR-FAT-Treibern von Holger Klabunde http://www.holger-klabunde.de/avr/avrboard.htm#cf.

Erste Versionen des Moduls von 2008 basieren auf einem ATMEGA32, Kingstener hat das Modul dann entfrachtet und kleine Platinen entworfen, die direkt an die Seite des KCs angesteckt werden können. Es gab Module für normale SD-Karten und mit Micro-SD-Slot.



Bootloader

Der AVR-Mikrocontroller enthält einen 2K großen Bootloader.

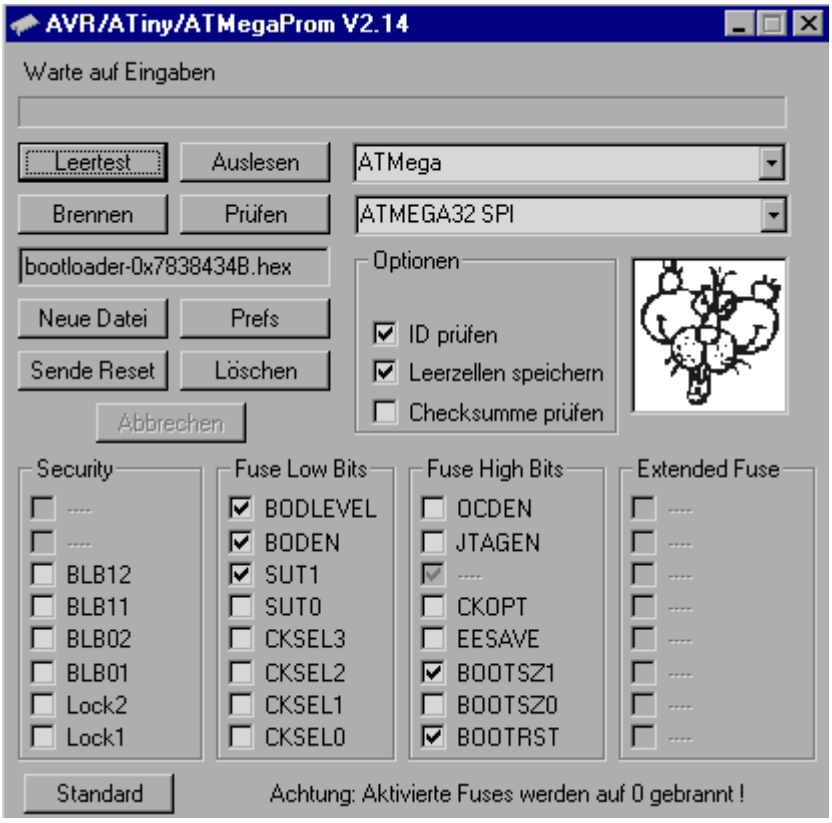
Befindet sich auf der SD-Karte eine Datei SD2KC.BIN mit neuerer Version als im Controller gespeichert, so kopiert der Bootloader diese Datei in den internen Flash als Firmware (= Betriebssystem) für sich und startet dieses. Damit ist auf einfachste Weise ein Update der Firmware möglich. Beim flashen blinkt dann normalerweise die LED.

Der hier verwendete Bootloader stammt aus dem mikrocontroller-Forum SD_Bootloader_für_AT_Mega, sd2iec_Bootloader_2008-02-23.zip. Er wurde minimal angepasst (main.c, makefile).

Initial muss der Controller deshalb den Bootloader (bootloader-0x7838434B.hex) einprogrammiert bekommen. Das geht mit einem externen AVR-Programmer oder einer selbstgebauten Lösung, wie etwa unter USB-Anschluss beschrieben. Wichtig ist das korrekte Setzen der Fuses:

Fuses: bootsz1, bootrst, keine crc Kontrolle[1)]

BOOTSZ1 = 0 und BOOTSZ0 = 1 (Boot Flash section size 2048 Byte = 1024 Word, Start Address 3C00h) (!!)
BOOTRST = 0: Ein Reset führt direkt zur Ausführung des Boot Loaders
Ext. Quarz (CKSEL = 1111, SUT=11) (SUT=10 lt. u.a. Angaben, vgl. Bild und Liste)



Holgi's AVR / ATMega Prommer, http://www.holger-klabunde.de/avr/avrboard.htm#AVR / ATMega Prommer

Mein AVR ausgelesen mit Wellon-Programmer VP-280:

```
CKSEL0:Disable CKSEL1:Disable CKSEL2:Disable CKSEL3:Disable SUT0:Disable
SUT1:Enable BODEN:Enable BODLEVEL:Enable BOOTRST:Enable BOOTSZ0:Disable
BOOTSZ1:Enable EESAVE:Disable CKOPT:Disable SPIEN:Disable JTAGEN:Disable
OCDEN:Disable Lock Bit1:Disable Lock Bit2:Disable Boot Lock Bit01:Disable
Boot Lock Bit02:Disable Boot Lock Bit11:Disable Boot Lock Bit12:Disable
```

Der Atmega-Microcontroller kann auch später im fertigen Gerät über das SPI-Interface programmiert werden!

Die SD-Karte

Als SD-Karten kommen normale MMC oder SD-Karten mit max. 2GB Größe in Betracht. Größere SDHC-Karten werden vom Bootloader nicht unterstützt. Es reichen bereits Karten mit 16 MB !

Der Bootlader verarbeitet nur FAT12 oder FAT16! Sicher sind aus Erfahrung Karten bis 512MB, darüber hinaus funktionieren verschiedene Karten bis etwa 4GB nur noch teilweise mit oder ohne Tricks oder gar nicht. Das SD2KC V1.2 dagegen verarbeitet FAT12/16/32, getestet mit diversen Micro-SD, SD und SDHC Karten bis 16GB.

Auf die mit FATxx formatierte SD-Karte müssen die beiden Programme

- load87
- SD2KC.BIN

ins Root-Verzeichnis.

Der Bootloader programmiert den AVR beim Start mit SD2KC.BIN, wenn die Version auf der SD-Karte neuer ist als die im AVR befindliche.

Z9001-Programme müssen in TAP-Format vorliegen. Sie können beliebig in Unterverzeichnissen der SD-Karte liegen.

Da nur 8+3-(DOS)-Filenamen angezeigt werden, sollte man die TAP-Dateien entsprechend umbenennen. Bei längeren Dateinamen wird sonst die Kurzform mit ~ im Namen angezeigt. Da der Z9001 keine Tilde auf der Tastatur hat, dient ^ als Tilden-Ersatz bei der Eingabe.
Statt der Endung TAP kann auch die originale Z9001-Dateityp genutzt werden. Ein kleines Perl-Programm **tap_rename.pl** benennt die TAP-Dateien um in 8.3-Dateinamen: Es wird der originale KC-Programmname aus der TAP-Datei geholt. Bei BASIC wird der Typ weggelassen.

SD2KC

Die Software für den ATMEL ATMEGA32 Mikrocontroller basiert i.W. auf der FAT-Engine von von Holger Klabunde http://www.holger-klabunde.de/avr/avrboard.htm#cf.

Aus Holgers Paket FATSingleOpt41.zip stammen unverändert die Quell-Dateien

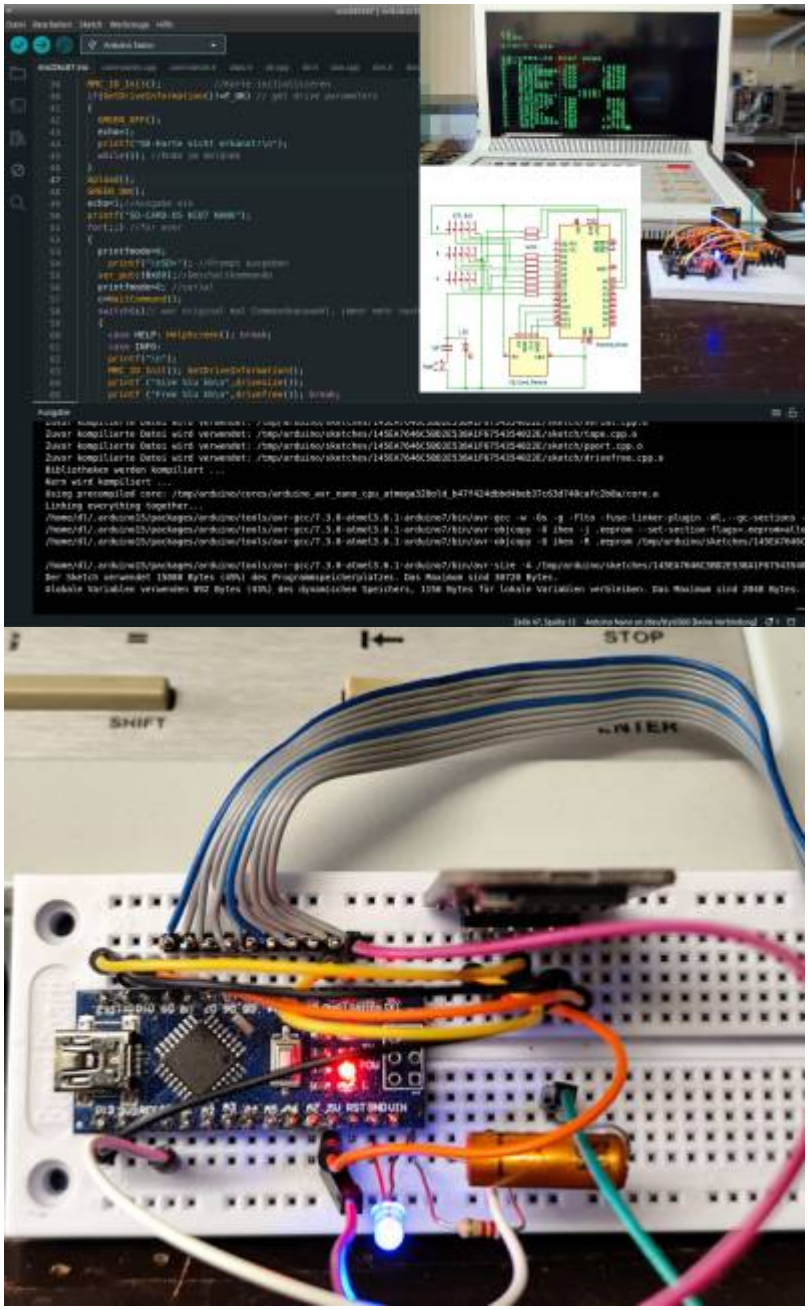
```
COMPACT.C COMPACT.H DIR.C DIR.H DOS.C DOS.H drivefree.c drivefree.h
DUMPSECT.C DUMPSECT.H FAT.C FAT.H FIND_X.C FIND_X.H LCD.C LCD.H mem-check.c
mem-check.h MMC_SPI.C MMC_SPI.H PRINTF.C PRINTF.H README.TXT READRAW.C
READRAW.H RENAME.C SERIAL.C SERIAL.H
```

sowie (ans konkrete Projekt angepasst) `DOSDEFS.H MYDEFS.H`

Von Kingstener und Arne Fitzenreiter stammen `COMMANDS.C COMMANDS.H KCFILE.C KCFILE.H KCSCF.C PPORT.C PPORT.H TAPE.C TAPE.H`

Arduino Nano

2024 hat Kingstener die Software für den Arduino Nano angepasst. Damit ist ein Nachbau auch möglich:



Unterlagen s. downloads

https://www.robotrontechnik.de/html/forum/thwb/showtopic.php?threadid=6232

1)

manche Brennprogramme bringen durch das nachträgliche Eintragen der CRC einen CRC-Fehler in Bin-File hinein. Wir brauchen aber genau die mit CRCGEN erzeugte CRC oder 00 00

From:
https://hc-ddr.hucki.net/wiki/ - **Homecomputer DDR**

Permanent link:
https://hc-ddr.hucki.net/wiki/doku.php/z9001/erweiterungen/sdcard2?rev=1737708807

Last update: **2025/01/24 08:53**