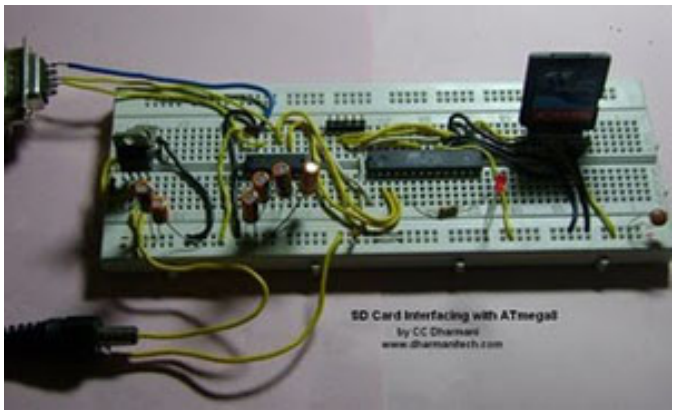
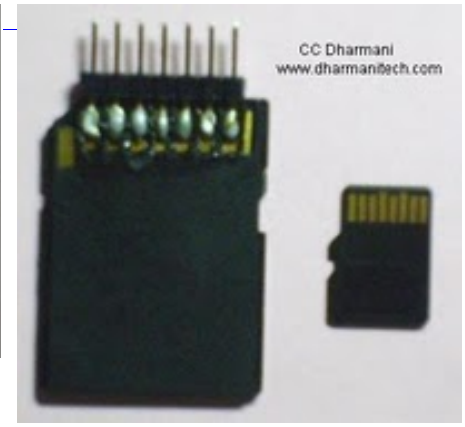
Popis pripojenia SD Karty k MCU typu AVR aj s implementáciou FAT32

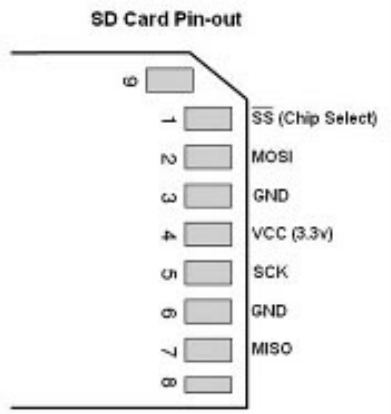
MicroSD karty sú dnes dostupné za veľmi nízke ceny, čo dáva možnosť doplniť akúkoľvek aplikáciu o veľkú pamäť. Nakoľko je kompatibilná s SPI zbernicou, pripijenie je veľmi jednoduché. Adaptéry pre SD karty sú taktiež veľmi ľahko dostupné a lacné, takže je možné realizovať daptér aj na skúšobnej doske len pripájkovaním k niekoľkým pinom.



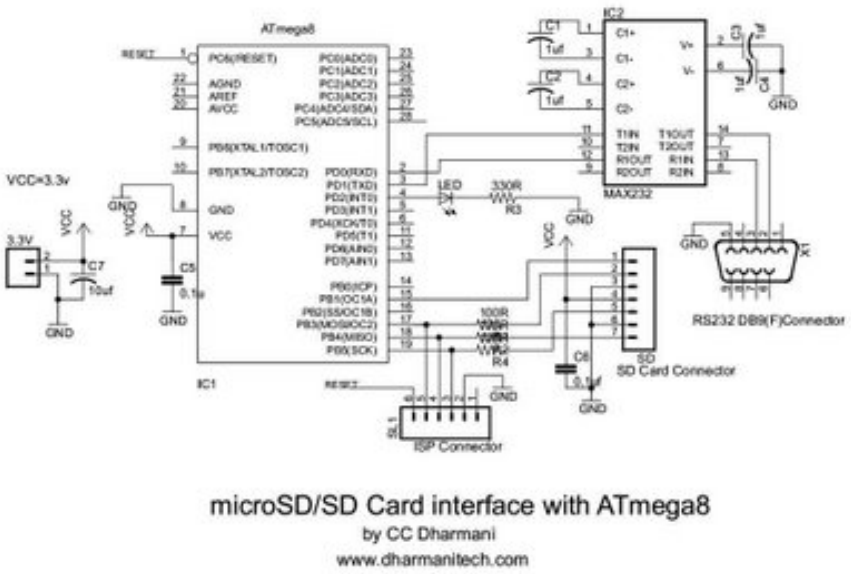
Nasledujúce obrázky popisujú zapojenie SD karty a konštrukciu adaptéra pomocou lámacej kolíkovej lišty pripájkovaním k MicroSD adaptéru.







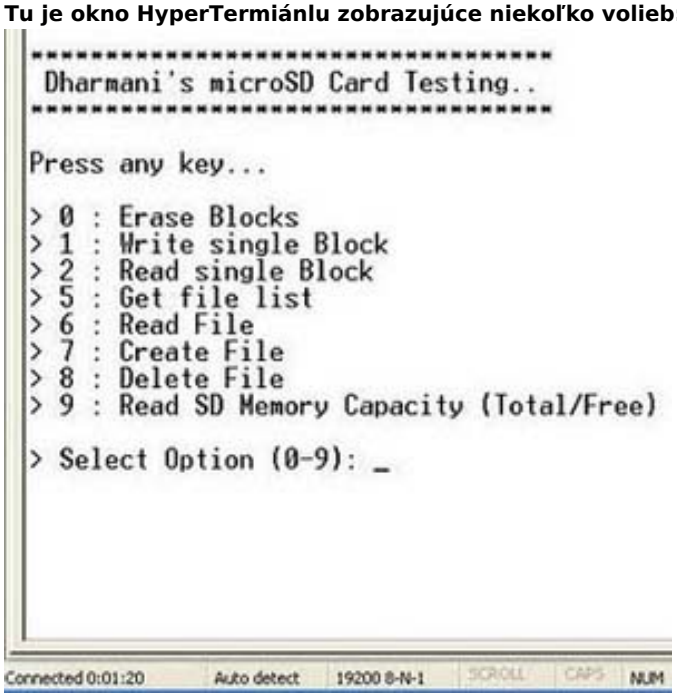
Použil som 1GB MicroSD karu od Sandisku (neskôr bol adaptér testovaný aj s inými kartami). Ako mikrokontrolér bola použitá AVR ATmega8 taktovaná interným oscilátorom s kmitočtom 8Mhz. MAX232 je použitý pre pripojenie k PC za účelom sledovania dát. Napájacím napätím 3,3V je napájaný procesor, MicroSD karta a aj MAX232 (aj napriek tomu, že MAX232 je navrhnutý pre napájanie 5V, funguje spoľahlivo aj pri 3,3V).
MicroSD karta je pripojená siedmimi pinmi. Datasheety ako aj veľa návrhárov odporúča datové vedenie ako aj nepoužité piny ošetriť pull-up rezistormi, avšak všetko spoľhlivo fungovalo aj bez nich, takže som ich nepoužil. Schéma je na nasledujúcom obrázku (doplnené 3.3.2009, bol pridaný ISP konektor a sériové rezistory k SD karte):



Cieľom tohoto projektu bolo vyskúšať pripojenie SD karty a pochopiť systém prenosu dát v RAW formáte ako aj prácu s FAT32. Začal som s prenosmi dát v RAW formáte, skúšal som zapísať nejaké data do ľubovoľných blokov MicroSD karty, čítnie blokov dát, čítanie a zápis viacerých blokov, mazanie viacerých blokov. To všetko v RAW formáte. Použil som RS232 pre zobrazenie dát načítaných mikrokontrolérom z SD karty. Mikrokontrolér odosielal dáta do HyperTerminálu. Podobne, dáta re zápis boli získavané z HyperTerminálu, písaním textu.

Keď som toto zvládol, naformátoval som SD karu pod WindowsXP (na FAT32) a zapísal na ňu nejaké textové súbory, adresáre a ďalšie súbory (všetko do koreňového adresára). Potom som napísal rutiny pre prácu s FAT32 - čítanie súborov, čítanie zoznamu súborov (opäť s použitím HyperTerminálu), nájdenie kapacity/voľného priestoru karty. Všetky tieto dáta boli mikrokontrolérom odosielané do HyperTerminálu.

Tu je okno HyperTermiánlu zobrazujúce niekoľko volieb:

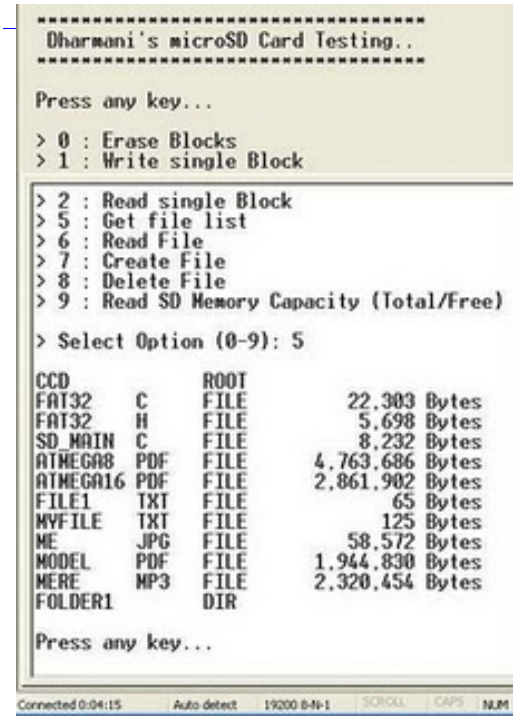


Voľby 0 až 4 predstavujú nízkoúrovňové funkcie pracujúce s datami v RAW formáte. Ak použijete rutiny 0, 1 alebo 3, bude treba pred prácou s FAT32 kartu preformátovať.
0: Vymaže zadaný počet blokov od zadaného bloku
1: Zapisuje dáta na zadanú adresu SD karty. Data sa zadávajú prostredníctvom HyperTerminálu cez klávesnicu
2: Číta dáta zo zadaného bloku. Dáta ú zobrazené v okne HyperTerminálu
3. Zapíše zadaný počet blokov od zadaného bloku
4. Prečíta zadaný počet blokov od zadaného bloku

V tomto prípade funkcie pre prácu s viacnásobnými blokmi spojenými s voľbami 3 a 4 sú zneprístupnené kvôli pamäťovej kapacite ATmega8, nakoľko tieto funkcie nie sú potrebné pre testovanie funkcií FAT32. Pre sprístupnenie volieb 3 a 4 môžete odstrániť makro definované v SD_routines.h (#define FAT_TESTING_ONLY). A ak použijete ATmega8, môžete odstrániť aj knižnicu FAT32..
Voľby 5 až 9 súvisia s FAT32. Podporované sú len krátke mená (8 znakov názov + 3 znaky prípona). Ak na kartu uložíte súbor s dlhým menom, bude týmito rutinami zobrazené len v krátkom formáte..
Na testovanie týchto volieb naformátujte kartu systémom FAT32 a uložte na ňu niekoľko adresárov a textových súborov (textové súbory možno jednoducho čítať a kontrolovať cez HyperTerminál).

5: Zobrazí zoznam dostupných adresárov a súborov s ich veľkosťou (z koreňového adresára karty)
6: Prečíta zvolený súbor a zobrazí jeho obsah prostredníctvom HyperTerminálu
7: Vytorí súbor s zadaným menom, obsah sa zadá prostredníctvom HyperTerminálu
8: Zmaže existujúci súbor s zadaným menom
9: Zobrazí celkovú kapacitu a voľný priestor na karte (použije sektor s FSinfo na karte)

Nasledujúce obrázky zobrazujú okno HyperTerminálu po použití volieb 5 a 9:





Poznámka: Nastavenie HyperTerminálu 19200 bd, No parity, Flow Control 'none'.

Tento projekt vyžaduje len minimum komponentov a dá sa jednoducho vzrealizovať v domácich podmienkach, a preto veselo skúšajte!

Zdrojové kódy doplnené do verzie 2.0:

- pridaná podpora pre SD karty, ktoré majú prvý sektor ako MBR namiesto boot sektora
- pridané funkcie createFile a deleteFile
- opravená chyba pri čítaní súborov na vzdialených sektoroch, tatiež opravená chyba v čítaní názvov 8+3 (kvôli omylu pôvodne 7+3)
- FSinfo sektor použitý na uloženie počtu voľných clusterov a prvého voľného clustera pre rýchlejší prístup k súborom
- vylepšené zobrazenie freeMemory (pôvodne to trvalo viac ako 30 sekúnd) za použitia sektora FSinfo. FSinfo sektor je teraz zmenený po každom vytvorení, alebo zmazaní súboru
- Veľkosť súboru zobrazená decimálne, pôvodne bolo v hex
- Raw SD funkcie pre prácu s viacnásobnými blokmi, ktoré nie sú potrebné pre prácu s FAT32 zablokované makrom FAT32_TESTING_ONLY aby sa získal priestor pre funkcie createFile a deleteFile (môžete ich aktivovať, ak použijete procesor s viac ako 8k flash), momentálne je pamäť plná na 99.9%

- hodinový kmitočet zdvihnutý z 1Mhz na 8 MHz, rýchlosť SPI (po inicializácii SD karty): 4MHz oproti pôvodným 500K a komunikačná rýchlosť 19200 bd (pre HyperrTerminál) oproti pôvodným 4800 bd

Zmeny vo verzii 2.1:

- odstránená chyba, ktorá znemožnila vytvorenie súboru, ak ich počet v koreňovom adresári prekročil 32x8
- koreňový adresár bol zbytočne zväčšený o jeden cluster po každom vytvorení nového súboru, opravené v novšej verzii
- taktiež opravený pevný počet 8 sektorov na cluster, teraz podpora aj iného počtu sektorov

Zmeny vo verzii 2.2 (iba pre ATmega32):

- pridaná možnosť AppendFile. Funkcia 'createFile' nahradená funkciou writeFile, ktorá najprv skontroluje existenciu zadaného mena súboru, ak takéto meno neexistuje, vytvorí nový súbor a zapíše data, ale ak súbor už existuje, otvorí ho a pripojí zadané dáta.

Extra updaty vo verziách 2.1 a 2.2:

- odstránená chyba pri čítaní dlhých mien
- súbory FAT32.c a .h opravené 13.9.09, odstránená chyba, ktorá obmedzovala veľkosť súboru na 32MB (vďaka Kun-Szabo Martonovi, ktorý na to upozornil)

Prenosová rýchlosť: 1 RAW datový blok (512 bytov) sa prečíta alebo zapíše za 4.15ms (123.37 KByt/s) pri aktuálnej rýchlosti SPI 4 MHz. Ak je k dispozícii viac ako 8k pamäti, môžete deklarovať funkcie SPI_receive() a SPI_transmit() ako 'inline'. To zvýši rýchlosť prenosu na 140 KByt/s. Prenosovú rýchlosť možno ďalej zvýšiť použitím 16MHz kryštálu (8 MHz SPI). Čítanie súborov z FAT32 prebieha rýchlosťou 78 až 91 KByt/s.

Download zdrojových kódov V2.1 (v formáte pre AVR-GCC, pre ATmega8):
Nasledujúce súbory boli kompilované s použitím winAVR (ver-20081205) v AVRStudiu, so zapnutou optimalizáciou (nastavené na -0s).
1. SD_main.c
2. SD_routines.c a SD_routines.h
3. FAT32.c a FAT32.h (V2.1)
4. SPI_routines.c a SPI_routines.h
5. UART_routines.c a UART_routines.h
6. Makefile
7. HEX súbor (V2.1)

Download zozipovaných zdrojových kódov upravených pre ATmega32. Prístupných všetkých 10 spomínaných volieb s pridanou možnosťou dopĺňania súborov. Posledná zmena 13.9.09, odstránený limit veľkosti súboru 32MB.
Download kompletný zdrojový kód pre ATmega32 (V2.2)

Referencie:
1. Špecifikácia FAT32 od Microsoftu
2. SanDisk SD mnuál v1.9
3. Špecifikácia SD-Simplified Physical Layer Ver 2.00
4. Dokument k použitiu SD kariet
5. Info o štruktúre FAT32, vrátane MBR
6. Jednoduchý popis štruktúry FAT32
7. Chanova knižnica FAT

Článok preložený so súhlasom autora.
Homepage projektu: http://www.dharmanitech.com

Preklad: Dali2