

Popis komunikačního protokolu, pro komunikaci procesorů v mp3 přehrávači.

Řídící procesor má na svých portech připojenou klávesnici pro řízení přehrávače a display pro zobrazování údajů (o přehrávaném souboru, listování složkami..)

Obslužný procesor je připojen k mp3 dekodéru a přes IDE sběrnici k pevnému disku.

Aby byla umožněna spolupráce obou zařízení, komunikují spolu oba procesory USARTem.

Řídící procesor musí vždy po odeslání příkazu počkat na odpověď! Některé příkazy musejí po sobě následovat v přesném pořadí. Příkaz nebude proveden, dokud nebudou odeslány všechny parametry. Pokud obslužný procesor na příkaz dlouho nereaguje, lze příkaz stornovat posláním min. osmi nulovými byty. Pokud ani to nepomůže, musí se procesor restartovat.

Toto je zatím jen návrh, ne popis, tak to skutečně pracuje...

Např. pokud má příkaz jeden 4 bytový parametr, jsou po USARTu odeslána data:

číslo příkazu, 4. byt parametru (nejnižší), 3., 2. byt, 1 byt (nejvyšší byt)

01h – vrat' vlastnosti disku (bez parametrů)

Musí být 1. příkaz ve vzájemné komunikaci. Pokud je v odpovědi nastaven 7. bit, tak nelze znova tento příkaz volat.

Parametry:

žádné

Odpověď:

- 1. byte obsahuje vlastnosti disku připojeného k řadiči. Byty mají následující význam:

```
0. bit = 1 -> disk je přítomen, zapnutý a funkční :-)  
1. bit = 1 -> podpora LBA (pokud není nastaven bit2, tak pouze LBA28  
-> disk do 120GB)  
2. bit = 1 -> podpora LBA 48 (disk je zřejmě větší jak 120GB)  
3. bit = 1 -> velikost sektoru je větší než 256 !!!  
-> (s takovým diskem program neumí)  
4. bit -> 0  
5. bit -> 0  
6. bit -> 0  
7. bit = 1 -> disk je pro nás použitelný (LBA, sektor o 512B)
```

Po řídící procesor je významný zvláště 7. bit. (Pokud je nastaven, může mp3 přehrávač pokračovat v práci.) Pokud není žádný disk k procesoru připojen, tak procesor na žádný příkaz nereaguje! Proto by měl řídící procesor vypsat něco jako „SEARCH ATA...” a poté čekat na odpověď příkazu 01h.

- 2. – 41. byte - model disku (40 ASCII znaků obsahující jméno disku)

např.:

„SAMSUNG SV0511D

“

02h – vrat' oddíl se systémem FAT32

Každý disk musí být rozdělen alespoň na jeden oddíl. Program v obslužném procesoru umí pracovat pouze s oddíly typu 0Bh nebo 0Ch (FAT32 s LBA adresací). Program dokáže najít až čtyři oddíly se systémem souborů FAT32, takový oddíl však musí být vždy na prvním záznamu MBR, či BR rozšířeného oddílu. (Jeden primární oddíl s FAT32 a poté vždy jeden logický oddíl v rozšířeném oddílu...)

Parametry:

● číslo požadovaného oddílu (0-3)

Vrátí adresu a jmenovku jednoho z nalezených oddílů se systémem souborů FAT32. Hodnota parametru může být v rozsahu 0-3, víc oddílů program nevyhledává.

Odpověď:

● 1. byte – signalizace oddílu

Pokud tento byte se rovná 00h, tak požadovaný záznam neobsahuje adresu oddílu. Pokud se rovná FFh, tak záznam obsahuje adresu oddílu se systémem FAT32 a jmenovku tohoto oddílu.

● 2. - 5. byte – adresa prvního sektoru oddílu

2. byte je nejnižší část adresy a 5. byte nejvyšší část adresy. Jedná se o LBA adresu sektoru, na kterém se nachází spouštěcí záznam svazku se systémem souborů FAT32.

● 6. - 16. byte – 11 ASCII znaků obsahujících jmenovku svazku

03h – nastav oddíl

Načte spouštěcí záznam vybraného oddílu se systémem FAT32. Parametr pro tento příkaz lze zjistit voláním příkazu 02h. Pokud jako odpověď je úspěch, nelze již znova tento příkaz (ani předchozí) volat. (Do restartu procesoru.)

Parametry:

● adresa požadovaného oddílu (4byty)

Adresa prvního sektoru oddílu, na kterém se nachází spouštěcí záznam FAT32. (Adresa by měla odpovídat jednomu ze záznamů získaných příkazem 02h.)

Odpověď:

● 1 byte signalizující úspěch

Pokud je nastaven 4. bit vráceného bytu, znamená to, že spouštěcí záznam svazku byl korektně načten. Pokud ne, došlo k některé z chyb:

- adresa sektoru neobsahuje spouštěcí záznam svazku s FAT32
- velikost sektoru není 512B
- cluster FAT je větší jak 16KB (více jak 32 sektorů na cluster)
- svazek neobsahuje dvě kopie FAT tabulky

04h – vrat' velikost clusteru

parametry:

- bez parametru

odpověď:

- velikost clusteru

vrátí jeden byte s velikostí FAT clusteru (kolik sektorů má cluster) Vždy se jedná o exponenty 2 (1, 2, 4, 8, 16, 32, 64, 128).

05h – vrat' záznam v adresáři

Záznamy v adresáři nemusejí být v systému FAT(32) nijak řazeny (podle abecedy)!

parametry:

- 1. až 4. byte je číslo prvního clusteru adresáře
- 5. až 6. byte = číslo záznamu v adresáři

Maximální počet záznamů, které lze tedy pomocí tohoto příkazu vyčíst je tedy 65536. Protože většina záznamů obsahuje záznam o dlouhém jméně některého souboru, je skutečný maximální počet zjištěných souborů a adresářů o něco menší.

odpověď:

- první byte signalizuje stav odpovědi:

- 00h => záznam v adresáři je prázdný (následujících 15 bytů je k ničemu)
- 01h => záznam je adresář
- 02h => záznam je soubor
- 06h => záznam je soubor s příponou MP3
- 80h => ukazatel na záznam (5. a 6. byte parametru) je mimo rozsah (Více souborů v adresáři není.)
- 81h => zadaný cluster neobsahuje začátek adresáře

- 2. až 9. byte --> 8 znaků dlouhé jméno

("DOSovský" tvar - např. složka "dokumenty" = "DOKUME~1")

- 10..12 byte -> u souboru tři znaky přípony

(u adresářů většinou mezery - 20h 20h 20h) (Pokud se jedná o adresář, a záznam o příponě se nerovná 0x202020 (tri mezery), tak skutečné jméno adresáře je : ADRESAR.PŘÍ)

- 13..16 byte -> 1. cluster souboru či adresáře
-

06h – vrat' číslo dalšího clusteru v alokačním řetězci

parametry:

- 1. až 4. byte je číslo clusteru od kterého chcete zjistit pokračování.

Program nekontroluje, zda číslo nepřevyšuje počet clusterů ve FAT!!! Pokud byl zadán parametr mimo tento rozsah, nevrátí program platné informace!

odpověď:

- 1. byte signalizuje stav výsledku:

- 00h => další 4 byty odpovědi obsahují číslo následujícího clusteru v alokačním řetězu
- FFh => předaný cluster je poslední v řetězu, nebo je prázdný. (další čtyři byty odpovědi obsahují neplatnou hodnotu)

- 2. až 5. byte obsahují číslo clusteru, který následuje v alokačním řetězci za předaným clusterem

07h – čti cluster

(Přečte a odešle všechna data obsažená v požadovaném clusteru.)

parametry:

- 1. až 4. byte je číslo libovolného clusteru (na kterém se třeba nachází soubor).

odpověď:

- velikost_clusteru * 512 bytů dat obsažených v požadovaném clusteru (pokud soubor v tomto clusteru zabírá třeba jen 100B a velikost clusteru je 8KB, odešle všech 8KB.)
-

08h – zjisti velikost souboru

Tento příkaz není pro přehrávání mp3 moc důležitý, je přidám pouze pro účely ladění programu.

Program nekontroluje zda požadovaný záznam obsahuje soubor, to lze zjistit voláním příkazu 05h

parametry:

- 1. až 4. byte je číslo prvního clusteru adresáře
- 5. až 6. byte = číslo záznamu v adresáři

odpověď:

- první byte signalizuje stav odpovědi:
 - 00h => velikost záznamu byla přečtena
 - 80h => ukazatel na záznam (5. a 6. byte parametru) je mimo rozsah (Více souborů v adresáři není.)
 - 81h => zadaný cluster neobsahuje začátek adresáře
- 4 byty reprezentující velikost souboru v bytech.

09h – hledej záznamy seřazeně podle abecedy

Ve FATce nemusejí být záznamy v adresáři nikterak řazeny (podle abecedy), proto byl přidán tento příkaz, který umožňuje získávání záznamů z adresáře seřazeně.

parametry

- 4 byty – první cluster adresáře
- 2 byty – číslo záznamu
Od kterého hledáme buď následující či předchozí záznam.
- 1 byt – parametr hledání

```
0. bit = 0 > pokud hledáme soubory, tak pouze MP3
    1 > pokud hledáme soubory, vrací všechny soubory
1. bit = 0 > hledáme klasicky (nejdříve adresáře, poté soubory)
    1 > hledáme pouze adresáře, nebo soubory
2. bit = 0 > je-li nastaven 1. bit, hledáme pouze adresáře
    1 > je-li nastaven 1. bit, hledáme pouze soubory
3. bit = 0 > hledáme podle čísla záznamu buď následující či předchozí záznam
    1 > hledáme první záznam v adresáři
4. bit = 0 > hledáme NÁSLEDUJÍCÍ záznam
    1 > hledáme PŘEDCHOZÍ záznam
```

I když hledáme pouze soubory, je jako první záznam v adresářích, které nejsou kořenové, vrácen vždy adresář „..“ odkazující na nadřazený adresář.

odpověď:

- první byte signalizuje stav odpovědi:
 - 00h => žádný další (předchozí) záznam již neexistuje (nebo byl zadán neplatný druhý parametr)
 - 01h => záznam je adresář
 - 02h => záznam je soubor
 - 06h => záznam je soubor s příponou MP3
 - 81h => zadaný cluster neobsahuje začátek adresáře
- 2. až 9. byte --> 8 znaků dlouhé jméno
("DOSovský" tvar - např. složka "dokumenty" = "DOKUME~1")
- 10..12 byte -> u souboru tři znaky přípony
(u adresářů většinou mezery - 20h 20h 20h) (Pokud se jedná o adresář, a záznam o příponě se nerovná 0x202020 (tri mezery), tak skutečné jméno adresáře je : ADRESAR.PŘI)
- 13..16 byte -> 1. cluster souboru či adresáře
- 17..18 byte -> číslo vyčteného záznamu

*Pokud jsou vyhledávány záznamy v jiném adresáři než kořenovém (ROOT), je **vždy** za první záznam považován adresář „..“ což prakticky je odkaz na nadřazený adresář.*

80h – hraj mp3

parametry:

- 1. až 4. byte je číslo prvního clusteru adresáře
- 5. až 6. byte = číslo záznamu v adresáři

odpověď:

- první byte signalizuje stav odpovědi:
 - 06h => soubor se nejspíše podařilo začít přehrávat (nekontroluje zda jsou mp3 data platná)
 - 00h, 01h nebo 02h => **záznam neobsahuje soubor s příponou MP3**
 - 80h => ukazatel na záznam (5. a 6. byte parametru) je mimo rozsah (Více souborů v adresáři není.)
 - 81h => **zadaný cluster neobsahuje začátek adresáře**

81h – vrat' stav přehrávání

Příkaz je bez parametru

odpověď:

- Dva byty indikující stav přehrávání. Bity mají následující význam:

```
PREH_STAV0:
    0. bit = 0 => stop nebo pauza (nic nehraje)
              1 => play (nejaky soubor je prehravan)
    1. bit = 0 => není zadný soubor k prehravani (nastaveni bitu play nema
zadny
                ucinek)
              1 => je soubor pripraven k prehravani (muze byt ale pauza - nic
nehraje)
    2. bit = 1 => je nastaven, kdyz se zmeni prehravany soubor (bez volani
prikazu)
    3. bit = 0 => do doby, nez prijde prikaz na dotaz STAVU... (81h)
nastaveni
                po skonceni souboru se pokracuje v prehravani (zavisi na
                nastaveni
                repeatu)
              1 => po skonceni prehravani souboru se nic neprehrava
                (ceka se na prijeti prikazu k prehravani)
    4. bit = 0 => repeat off
              1 => repeat on
    5. bit = 0 => repeat adresare
              1 => repeat souboru (po skonceni prehravani souboru se ten samy
soubor
                zacne prehravat znova)
    6. bit = 0 =>
              1 => rezervovano (pro nahodny vyber, nebo jiny rezim
prehravani...)
    7. bit = 0 =>
              1 => rezervovano
PREH_STAV1:
    0. bit = 0 => normalni prehravani
              1 => previjeni mp3 dopredu
    1. bit = 0 =>
              1 => rezervovano (pro previjeni dozadu)
    2. bit = 0 => zpusob previjeni,
              0 => previji se rychle, potichu
              1 => previji se pomalu, vzdy se po nejake dobe kratky usek souboru
prehraje
    3. bit = 0 => normalni nastaveni zvuku
              1 => zvýraznění basu a vysek (moznost dekodery vs1001 "bass/treble
enhancer")
    4. bit = 0 => NO MUTE
              1 => MUTE
    5,6,7 => rezervovano
```

Je-li nastaveno, že po skončení přehrávání souboru se pokračuje v přehrávání, a začne se automaticky přehrávat další soubor (podle nastavení repeatu) nastaví se 2. bit. Adresu souboru, který se právě přehrává

lze zjistit voláním příkazu 83h.

● 2 byty obsahující odehraný čas mp3 v sekundách

Tato informace je získávána z registru dekodéru. Čas je platný i v případě rychlého převíjení. Pouze v případě neplatných dat, či skoků v souboru neobsahuje tento záznam pravdivou informaci.

82h – nastav hlasitost

parametry

- 1.byte: hlasitost pro levý kanál
- 2. byte: hlasitost pro pravý kanál

Hodnota 0 je maximální hlasitost, 255 minimální hlasitost.

Výchozí hodnota po startu přehrávače je 0x4040.

Odpověď:

- vždy FFh
-

83h – vrať informace o přehrávaném souboru

Program netestuje zda se vůbec nějaký soubor přehrává. To lze zjistit testováním 1. bitu 1. bytu odpovědi na příkaz 81h.

bez parametru

odpověď:

- 2 byty získané z registru AUDATA dekodéru vs1001. Bližší informace o jejich obsahu najdete na straně 26 datasheetu k vs1001. (Obsahuje informace o datovém toku, vzorkovací frekvenci a počtu kanálů.)
- 2 byty získané z registru HDAT0
- 2 byty získané z registru HDAT1
- 4 byty – LBA adresa adresáře, který se právě přehrává
- 2 byty – číslo záznamu, který se právě přehrává

Registry HDAT0, HDAT0 obsahují informace získané z hlavičky mp3 souboru. Jejich bližší popis najdete na straně 27 datasheetu k vs1001. Nejdůležitější informaci obsahují asi bity 15 až 5 registru HDAT1. Pokud tyto byty obsahují konstantu 2047, jsou přehrávaná data platná. (Mp3 hraje.)

84h – nastav stav přehrávání

parametry:

- 2 byty nastavující vlastnosti přehrávání

```
PREH_STAV0:
  0. bit = 0 => pauza
           1 => play (pokud není připraven soubor k přehrávání, nemá
nastavení           tohoto bitu žádný účinek)
  1. a 2. bit nelze příkazem nastavit (mohou mít libovolnou hodnotu)
  3. bit = 0 => po skončení souboru se pokračuje v přehrávání (zavisi na
nastavení           repeatu)
           1 => po skončení přehrávání souboru se nic nepřehrává
                (čeká se na přijetí příkazu k přehrávání)
  4. bit = 0 => repeat off
           1 => repeat on
  5. bit = 0 => repeat adresare
           1 => repeat souboru (po skončení přehrávání souboru se ten samý
soubor              začne přehrávat znovu)
  6. bit - rezervováno (pro nahodný vyber, nebo jiný režim přehrávání...)
  7. bit - rezervováno
PREH_STAV1:
  0. bit = 0 => normalní přehrávání
           1 => převíjení mp3 dopředu
  1. bit = 0 =>
           1 => rezervováno (pro převíjení dozadu)
  2. bit = způsob převíjení,
           0 => převíjí se rychle, potichu
           1 => převíjí se pomalu, vždy se po nějaké době kratký úsek souboru
                přehraje
  3. bit = 0 => normalní nastavení zvuku
           1 => zvýraznění basu a vysek (možnost dekodéru vs1001 "bass/treble
                enhancer")
  4. bit = 0 => NO MUTE
           1 => MUTE
  5,6,7 => rezervováno
```

I když výrobce dekodéru píše, že by měl být čas MP3 správně vypočítáván i během převíjení, u některých souborů tomu tak není. (Při začátku převíjení se registr s odehraným časem vynuluje.) Zatím se mi nepodařilo zjistit, zda tato chyba je u dekodéru vs1001k běžná, či se jedná o chybu mého firmwaru. Při pomalém převíjení je poměr přehraných sektorů ku přeskočeným 5:20. Při pomalém převíjení může docházet k velmi hlasitým defektům výstupního signálu! Nedoporučuji převíjet pomalu při maximální hlasitosti.

V nejnovější verzi firmwaru (0.9) zatím nefunguje nastavení opakování složky.

odpověď:

- vždy FFh