

Bonusová úloha Jednoduchý souborový systém

Obsah

1. [1. úloha - Vytvoření souborového systému na EEPROM \(10 b\)](#)

1. [Formátování úložiště](#)
2. [Vytvoření souboru](#)
3. [Smazání souboru](#)
4. [Zobrazení souboru](#)
5. [Zjišťování velikosti souboru](#)
6. [Výpis seznamu souborů](#)
7. [Defragmentace](#)
8. [Tipy pro realizaci](#)

1. úloha - Vytvoření souborového systému na EEPROM (10 b)

Vaším úkolem bude navrhnout a implementovat jednoduchý souborový systém v paměti EEPROM.

Prostřednictvím sériové linky bude možné:

- Formátovat úložiště
- Vytvářet soubory
- Mazat soubory
- Zobrazovat soubory
- Zjišťovat velikost souborů
- Vypisovat seznam souborů
- Defragmentovat úložiště

Příkladem jednoduchého systému souborů může být využití tabulky, ve které budou uloženy informace o jednotlivých souborech. Maximální počet souborů bude omezen fixní hodnotou. Ta bude určovat počet řádků tabulky. V tabulce bude pro každý soubor uvedena jeho adresa, velikost, název a příznak, zda tento záznam odkazuje na soubor, nebo je momentálně nevyužitý (může být nahrazeno speciální hodnotou adresy). Maximální délka názvu bude omezena fixní hodnotou. Maximální délka souboru bude omezena rozsahem datového formátu pro její uložení.

Takto například může vypadat výsledná tabulka:

Adresa		Délka		Název					
88	10	's'	'o'	'u'	'b'	'o'	'r'	'1'	0
98	30	's'	'o'	'u'	'b'	'o'	'r'	'2'	0
128	30	'a'	'b'	'c'	0	'o'	'r'	'3'	0

Adresa		Délka		Název					
-1	10	's'	'o'	'u'	'b'	'o'	'r'	'g'	'g'
168	10	'a'	's'	'd'	'f'	0	0	0	0
-1	0	0	0	0	0	0	0	0	0
-1	0	0	0	0	0	0	0	0	0
-1	0	0	0	0	0	0	0	0	0

Uvedená tabulka umožňuje zapsat celkem 8 souborů. Maximální délka názvu souboru je 8 znaků (bajtů). Adresa souboru má 2 bajty, záporná hodnota značí, že tento řádek tabulky neodkazuje na žádný soubor. Velikost souboru má 1 bajt (případně 2 bajty, pokud bychom chtěli maximální délku souboru větší než 255 bajtů). Celkem tedy bude mít tabulka (v případě jednobajtové délky souboru) $8 \cdot (2+1+8) = 88$ bajtů. Samotné soubory lze tedy zapisovat od adresy 88.

Tabulka obsahuje 4 soubory:

- Soubor s názvem "soubor1" o délce 10 bajtů se nachází na adrese 88
- Soubor s názvem "soubor2" o délce 30 bajtů se nachází na adrese 98
- Soubor s názvem "abc" (který je zapsaný na místě smazaného souboru "soubor3") o délce 30 bajtů se nachází na adrese 128
- Soubor s názvem "asdf" o délce 10 bajtů zapsaný na adrese 168

Dále vidíme, že:

- Tabulka obsahovala soubor s názvem "soubor99", který byl smazán.
- Tabulka obsahuje celkem 4 volné pozice pro soubory (1 smazaný a 3 nikdy nezapsané).

Formátování úložiště

Formátování se provede zasláním příkazu *format* po sériové lince. Po provedení se odešle zpět informace o dokončení formátování.

Formátování musí uvést naimplementovaný systém souborů do stavu, kdy je validní a není v něm uložený žádný soubor.

Ve výše uvedeném příkladu to znamená nastavit položku adresa u všech záznamů v tabulce na zápornou hodnotu.

Vytvoření souboru

Vytvoření souboru se provede zasláním příkazu *create* po sériové lince v následujícím formátu:

```
create name length content
```

Example:

```
create file1 32 This is content of file 'file1'.
```

Prvních *length* bajtů zaslanych po mezeře za číslem je uloženo do souboru s názvem *name*. Pokud soubor s tímto názvem již existuje, tabulka souborů je plná, není k dispozici místo pro uložení souboru, nebo je použita neplatná syntaxe příkazu, odešlete zpět odpovídající chybovou hlášku. V opačném případě odešlete informaci o úspěšném provedení příkazu.

Ve výše uvedeném příkladu by se nejprve prohledaly názvy v tabulce. V případě, že by zde nebyl soubor se shodným názvem, soubor by se zapsal za poslední uložený, tedy na adresu 178. Pokud by byla velikost souboru menší, nebo rovna 10, pak by se uložil do volného místa za souborem "abc" na adresu 158.

Smazání souboru

Smazání souboru se provede zasláním příkazu *remove* po sériové lince v následujícím formátu:

```
remove name
```

Example:

```
remove file1
```

Pokud soubor s tímto názvem neexistuje, odešlete zpět odpovídající chybovou hlášku. V opačném případě odešlete informaci o úspěšném provedení příkazu.

Ve výše uvedeném příkladu by se prohledaly názvy v tabulce a v případě nalezení shody by se u záznamu nastavila adresa na zápornou hodnotu.

Zobrazení souboru

Zobrazení souboru se provede zasláním příkazu *read* po sériové lince v následujícím formátu:

```
read name
```

Example:

```
read file1
```

Pokud soubor s tímto názvem neexistuje, odešlete zpět odpovídající chybovou hlášku. V opačném případě odešlete obsah souboru po sériové lince.

Ve výše uvedeném příkladu by se prohledaly názvy v tabulce a v případě nalezení shody by se odeslalo tolik bajtů, kolik odpovídá délce uvedené u daného záznamu, počínaje bajtem na adrese uvedené u daného záznamu.

Zjišťování velikosti souboru

Zjištění velikosti souboru se provede zasláním příkazu *size* po sériové lince v následujícím formátu:

```
size name
```

Example:

```
size file1
```

Pokud soubor s tímto názvem neexistuje, odešlete zpět odpovídající chybovou hlášku. V opačném případě odešlete velikost nalezeného souboru.

Ve výše uvedeném příkladu by se prohledaly názvy v tabulce a v případě nalezení shody by se odeslala délka souboru u daného záznamu.

Výpis seznamu souborů

Vypsání seznamu souborů se provede zasláním příkazu *list* po sériové lince. Seznam bude vypsán zpět na sériovou linku. U každého záznamu bude napsaný název, velikost a adresa souboru.

Ve výše uvedeném příkladu by se prošly všechny záznamy v tabulce a informace o těch, které by obsahovaly nezápornou adresu, by se odeslaly.

Defragmentace

Defragmentace se provede zasláním příkazu *defrag* po sériové lince. Po provedení se odešle zpět odpověď o dokončení defragmentace.

Defragmentace v našem případě znamená jen urovnání záznamů (soubory nedělíme na fragmenty). Výsledkem operace by mělo být, že budou všechny neprázdné záznamy v tabulce souborů uloženy za sebou na začátku tabulky a taktéž všechny soubory budou v datovém prostoru na sebe přímo navazovat.

Ve výše uvedeném příkladu by výsledná tabulka souborů po provedení "defragmentace" vypadala takto:

Adresa		Délka		Název					
88	10	's'	'o'	'u'	'b'	'o'	'r'	'1'	0
98	30	's'	'o'	'u'	'b'	'o'	'r'	'2'	0
128	30	'a'	'b'	'c'	0	'o'	'r'	'3'	0
158	10	'a'	's'	'd'	'f'	0	'r'	'9'	'9'
-1	10	'a'	's'	'd'	'f'	0	0	0	0
-1	0	0	0	0	0	0	0	0	0
-1	0	0	0	0	0	0	0	0	0
-1	0	0	0	0	0	0	0	0	0

Soubory by byly samozřejmě odpovídajícím způsobem umístěné na úložišti.

Tipy pro realizaci

- Uvedená implementace je pouze příkladem, můžete si navrhnout vlastní souborový systém úplně jinak.
- Pokud vám zbyde čas, nebojte se přidat další funkcionalitu, můžete být odměněni bonusovými body.
- Všechny změny na souborovém systému provádějte ihned po zaslání příkazu. Pokud si je budete "cachovat" do operační paměti, tak o ně při ztrátě napájení/resetu přijdete.
- Snažte se minimalizovat opotřebení EEPROM. Kde to jde, využívejte funkce, které zapisují do paměti jen v situaci, kdy se nová hodnota liší od staré.
- Pokud máte vlastní Arduino a nechcete si opotřebovávat EEPROM, využijte simulátor. **Na cvičení ale úlohu předvedte na přípravku.**
 - Persistence paměti v simulátoru je omezená. EEPROM udrží hodnotu při stisknutí tlačítka Reset, ale vynuluje se při ukončení simulace. Pro testování si vytvořte funkce, které vám EEPROM zaplní testovacími daty.
- Pokud budete pracovat s názvem souboru jako se stringem v C, nezapomeňte, že pokud název souboru zabírá všechny dostupné bajty v tabulce, musíte si k němu ručně přidat zakončovací nulu. Alternativně můžete maximální délku souboru o 1 zkrátit a nulu na konci zapisovat do tabulky vždy.
- Při odesílání příkazu *create* dbejte na to, aby velikost souboru odpovídala odeslanému počtu bajtů. V opačném případě vám to bude při přijímání dělat problémy a bude se to velmi těžko detekovat.
- Názvy souborů a jejich obsah ukládejte beze změny přímo tak, jak přijdou po sériové lince. Naopak délku souborů (případně adresu) musíte při přijetí po sériové lince nejdříve převést ze stringu do integeru a při jejím odesílání po sériové lince zase opačně.
- Obsah souboru přijímejte tak, aby se korektně uložily i případné netisknutelné či bílé znaky. Název by naopak měly tvořit jen tisknutelné znaky, jinak bychom museli také posílat jeho délku.