

Soutěž v programování 36. ročník Školní rok 2021/2022

Zadání krajského kola – programování mikrořadičů

Zařízení pro vysílání Morseovky

Komponenty

pasivní bzučák

signalizační LED

2 tlačítka či přepínače

rezistory pro zapojení výše uvedených periférií

nepájivé kontaktní pole

propojovací kabely

mikrokontroler (doporučené Arduino, nesmí obsahovat OS)

Zadání

Sestavte signalizační zařízení (dále v textu Zařízení) pro vysílání morseovky přepínatelné do tří režimů: AUDIO, VISUAL a DEBUG.

Nastavte komunikaci mezi PC a Zařízením po sériové lince UART rychlostí 9600 baudů za sekundu. Zařízení vypíše na sériovou linku zprávu `Ready...` a počká na vstup ze sériové linky. Z ní načte zprávu (posloupnost znaků ukončená znakem konce řádku). V průběhu zadávání nic nevypisuje, jakmile přijde znak konce řádku, vygeneruje v morseově kódu celou zprávu dle aktivního režimu

- **DEBUG:** vypíše výstup na sériovou linku: tečka `.` (ASCII znak 2E), čárka `,` (2D), konec znaku `|` (7C).
- **VISUAL:** vyblíká zprávu na LED (použijte pokud možno interní). Délka svícení LEDky tečka 200ms, čárka 500ms, 500ms mezi signály tečky a čárky, 1000ms mezi písmeny
- **AUDIO:** vypípá zprávu na bzučáku (stejně délky tónu jako u VISUAL režimu)

Ve zprávě se budou zpracovávat znaky **A-Z** (malá i velká písmena bez diakritiky) v morseově kódu dle ITU-R M.1677-1, ostatní znaky se budou ignorovat (a nezpůsobí chybu).

Po vypsání zprávy se opět vypíše zpráva `Ready...` a čeká se na další vstup. Volbu režimu realizujte tlačítky nebo přepínači (vč. ošetření zákmitu) a signalizujte LEDkami (jinými než výstupními v režimu VISUAL).

Zpracujte dokumentaci, jak se mezi režimy na vašem zařízení přepíná, jak se ze signalizačních LED pozná, který režim je aktivní a jak komunikovat se zařízením přes sériovou linku. Přiložte grafické schéma vašeho zapojení.

Pokyny k implementaci

Identifikátory pojmenovávejte jednotně pouze česky, pouze slovensky, nebo pouze anglicky.

Výpis výstupu v morseovce realizujte voláním jedné obecné funkce `transmit(message)` příp. `vysilej(zprava)`, kde parametrem je zpráva načtená ze sériové linky.

Pro získání bodů navíc (viz testovací scénář) naprogramujte zařízení vytvořením následujících funkcí:

- `dot()` příp. `tecka()` pro výpis tečky
- `dash()` příp. `carka()` pro výpis čárky
- `end_of_symbol()` příp. `konec_symbolu()` zavolá se po výpisu tečky nebo čárky
- `end_of_letter()` příp. `konec_pismena()` zavolá se po výpisu písmena
- `end_of_message()` příp. `konec_zpravy()` zavolá se po vypsání zprávy

Např. pro implementaci režimu DEBUG bychom mohli implementovat funkce takto:

```
void debug_dot() { Serial.print("."); }
void debug_dash() { Serial.print("-"); }
void debug_end_of_symbol() { }
void debug_end_of_letter() { Serial.print("|"); }
void debug_end_of_message() { Serial.println(); }
```

Na tyto funkce se pak v kód bude odkazovat, např. pokud uživatel zvolil režim DEBUG, nastavíme funkce kódem ve smyslu

```
typedef void(*Fn)();
Fn dot, dash, end_of_symbol, end_of_letter, end_of_message;

...

if(debug) {
    dot = &debug_dot;
    dash = &debug_dash;
    end_of_symbol = &debug_end_of_symbol;
    end_of_letter = &debug_end_of_letter;
    end_of_message = &debug_end_of_message;
}
```

Funkce `transmit(message)` příp. `vysilej(zprava)` nesmí být závislá na žádném hardware dle vzoru *Inversion of Control*. Jinými slovy přepínání pinů při vysílání nesmí být mimo funkce `dot`, `dash` a `end_of_*`.

Požadovaný výstup práce

- Funkční sestavený obvod.
- Dokumentace v textovém souboru `readme.txt`, kde bude popsáno, jak se mění režimy v zařízení a jak komunikovat se zařízením přes sériovou linku, jak jste vyřešili zákmit tlačítek, případně výčet věcí, které jste nezprovoznili nebo jejichž chování se liší od zadání.
- Schéma zapojení v souboru `schematics.png` (ideálně v programu, který umí rovné dotažené paralelní čáry a standardní elektronické značky, např. Eagle, Dia či online aplikace falstad.com, circuitlab.com)

Po dokončení práce ponechte na svém PC či notebooku

- připojené zařízení s otevřeným oknem pro komunikaci se sériovou linkou
- otevřené okno se zdrojovým kódem
- otevřené okno se souborem `readme.txt`
- otevřené okno se souborem `schematics.png`

Zajistěte, aby se notebook do příchodu porotce nezamknul. Pokud nechcete dát svůj notebook takto k dispozici (potřebujete odjet dříve, máte v notebooku tajná data CIA a podobně), domluvte se s porotci na testování na jejich strojích: vyzkoušejte, že komunikace UART u nich funguje (jde hlavně kompatibilitu ovladačů) a domluvte se na předání zdrojového kódu a uvedených dvou souborů (e-mail, sdílený disk či jinak).

Doufáme, že se Vám zadání líbí, přejeme mnoho tvůrčí inspirace a příjemnou práci

Porota a organizátoři