

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

SEMESTRÁLNÍ PRÁCE



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV MIKROELEKTRONIKY

DEPARTMENT OF MICROELECTRONICS

ELEKTRONICKÁ HRA LOGIC

ELECTRONIC GAME LOGIC

SEMESTRÁLNÍ PRÁCE

SEMESTRAL THESIS

AUTOR PRÁCE

AUTHOR

Renata Zemanová

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. Jiří Háze, Ph.D.

BRNO 2020

Semestrální práce

bakalářský studijní program **Mikroelektronika a technologie**

Ústav mikroelektroniky

Studentka: Renata Zemanová

ID: 211251

Ročník: 3

Akademický rok: 2020/21

NÁZEV TÉMATU:

Elektronická hra Logic

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s principem hry Logic. Navrhněte a realizujte elektronickou verzi této hry s různými úrovněmi obtížnosti a v praxi ověřte funkčnost vašeho řešení.

DOPORUČENÁ LITERATURA:

Podle pokynů vedoucího práce

Termín zadání: 21.9.2020

Termín odevzdání: 18.12.2020

Vedoucí práce: doc. Ing. Jiří Háze, Ph.D.

doc. Ing. Jiří Háze, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor semestrální práce nesmí při vytváření semestrální práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

PROHLÁŠENÍ

Prohlašuji, že svou semestrální práci na téma „Název studentské práce“ jsem vypracoval samostatně pod vedením vedoucího semestrální práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené semestrální práce dále prohlašuji, že v souvislosti s vytvořením této semestrální práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autora

Obsah

Úvod	15
1 Pravidla hry	17
2 Koncept	19
3 Použitý hardware	21
3.1 Procesor	21
3.2 LED	21
4 Návrh DPS	23
4.1 Napájení	23
4.2 Nabíjecí obvod	23
4.3 Měření baterie	24
4.4 Stepdown	24
4.5 Ochrana proti přepólování	24
4.6 Zapínání LED	24
4.7 Programovací převodník	24
5 Ovládání hry	25
6 Oživování	27
7 Ovládání hry	29
Závěr	31
Literatura	33
Seznam symbolů, veličin a zkratk	35
Seznam příloh	37
A Některé příkazy balíčku <code>thesis</code>	39
A.1 Příkazy pro sazbu veličin a jednotek	39
A.2 Příkazy pro sazbu symbolů	39
B Druhá příloha	41
C Příklad sazby zdrojových kódů	43
C.1 Balíček <code>listings</code>	43

Seznam obrázků

B.1 Alenčino zrcadlo	41
--------------------------------	----

Seznam tabulek

4.1	Nastavení nabíjecího proudu rezitorem R_{PROG}	23
A.1	Přehled příkazů	39

Seznam výpisů

C.1	Ukázka sazby zkratk	43
C.2	Příklad Schur-Cohnova testu stability v prostředí Matlab.	44
C.3	Příklad implementace první kanonické formy v jazyce C.	45

Úvod

Úvod studentské práce, např. . .

Tato práce se věnuje oblasti **DSP!** (**DSP!**), zejména jevům, které nastanou při nedodržení Nyquistovy podmínky pro *vzorkovací kmitočet* (f_{vz}).¹

Šablona je nastavena na *dvoustranný tisk*. Pokud máte nějaký závažný důvod sázet (a zejména tisknout) jednostranně, nezapomeňte si přepnout volbu **twoside** na **oneside**!

¹Tato věta je pouze ukázkou použití příkazů pro sazbu zkratk.

1 Pravidla hry

Logic je desková hra pro dva hráče. Jeden hráč určí hledanou kombinaci, dále bude označován jako hráč A, a druhý tuto kombinaci za pomoci logických úvah a vyhodnocení hráčem A hledá, dále bude označován jako hráč B.

Hráči si určí herní pozice. Hráč A vybere barvy a následně je v určitém pořadí vloží do zadávacího pole a zakryje je, aby tuto kombinaci spoluhráč neviděl. Hráč B je v této chvíli otočen. Hráč B následně zvolí náhodnou kombinaci barev a pozic. Po ukončení tahu nechá hráče A, aby tah vyhodnotil. Hráč A vyhodnotí tah následujícím způsobem. Pokud hráč B vložil správnou barvu na správnou pozici, tak vloží do vyhodnocovací sekce černý kolík. Pokud vložil barvu, která se v zadání vyskytuje, ale vložil ji na nesprávnou pozici, tak vloží bílý kolík. Pokud zůstanou některé pozice neobsazené, tak to znamená, že daná barva se v zadání nevyskytuje. Poté začne hráč B na základě vyhodnocení a svých všech předchozích tahů hledat kombinaci.

Cílem hry je nalézt správnou kombinaci dříve, než skončí plocha herního pole.

Těžší varianta spočívá v možnosti absence barvy - v zadání je mezera.

2 Koncept

Celá DPS je navržena tak, aby se svým vzhledem co nejvíce podobala deskové hře. Ve spodní části DPS jsou tlačítka, která slouží pro ovládání hry. Hráč pomocí tlačítek vybírá pozici a určuje barvu LED na dané pozici. Po ukončení tahu hra vyhodnotí kombinaci zvolenou hráčem a hráč může pokračovat dalším tahem. Herní pole proto obsahuje sekci s herními LED a s vyhodnocovacími LED.

DPS je navržena pro výrobu i strojní osazení u firmy JLCPCB. Výrobní podklady proto musely být navrženy v souladu s jejich výrobními možnostmi [citace].

3 Použitý hardware

3.1 Procesor

Byl zvolen procesor ESP32-PICO.

3.2 LED

Byly zvoleny programovatelné LED WS2812C. Tento typ LED je určen pro přenosná zařízení, díky jejich nízké spotřebě oproti typu WS2812B. Tyto LED jsou řízeny programově. K procesoru jsou připojeny všechny jedním datovým pinem. Z procesoru jsou těmto LED předávána data za sebou v RGB formátu. Každá LED má definovaný počet bytů, ve kterých jsou data právě pro jednu LED. Pokud LED registruje další data, tak je pošle dál. Tímto způsobem se všechna data dostanou ke všem LED. V programu musí být tedy zachováno pořadí LED.

Porovnání WS2812B s WS2812C

4 Návrh DPS

DPS je navržena v programu KiCad a má 4 vrstvy. Vnitřní vrstvy slouží pro napájení. V jedné vnitřní vrstvě je po celé její ploše rozlité GND a ve druhé vnitřní vrstvě jsou rozlita jednotlivá napájecí napětí. Vrchní a spodní strana slouží pro signálové dráhy. Všechny SMD součástky jsou umístěny na vrchní vrstvě, protože firma JL-CPCB osazuje pouze z jedné strany. THT součástky jsou připraveny na ruční pájení, proto je kolem nich ponecháno více místa.

Kondenzátory u procesoru musí být co nejbližší k jeho pouzdru, aby se do procesoru dostalo co nejméně rušení.

4.1 Napájení

Byly zvoleny baterie 18650.

4.2 Nabíjecí obvod

Pro nabíjecí obvod byl zvolen čip TP4056. Jeho zapojení bylo převzato z datasheetu [citace]. Velikost rezistoru R_{PROG} se volí podle nabíjecího proudu. Tabulka rezistorů R_{PROG} .. [citace]

Tab. 4.1: Nastavení nabíjecího proudu rezitorem R_{PROG}

R_{PROG} [kOhm]	Nabíjecí proud [mA]
10	130
5	250
4	300
3	400
2	580
1,66	690
1,5	780
1,33	900
1,2	1000

Nabíjení baterií by mělo probíhat při 0,5C, tudíž pro 18650 je to cca 0,5 A. Proto byl zvolen rezitor R_{prog} 2 kOhm.

4.3 Měření baterie

4.4 Stepdown

4.5 Ochrana proti přepólování

4.6 Zapínání LED

4.7 Programovací převodník

Jako nabíjecí a zároveň programovací konektor slouží USB. Procesor má pouze programovací rozhraní RS-232. Aby mohlo programování probíhat z USB, musí DPS obsahovat převodník z USB na RS-232. Byl zvolen převodník CP2102.

Byl zvolen protože na ESP32 devkitu C je použit právě tento převodník. Byl zapojen podle doporučeného zapojení v datasheetu [citace].

5 Ovládání hry

6 Oživování

Po dodání DPS z výroby byly zapájeny THT komponenty (pouzdro na baterii, USB konektor, tlačítka a vypínač). Po zapojení baterií 18650 do pouzdra se rozsvítí zelená LED D1, která indikuje přítomnost napájecího napětí. Po připojení k počítači přes USB se rozsvítí LED D9. Ta signalizuje nabíjení baterií. Pokud se rozsvítí LED D8, znamená to, že baterie jsou nabité.

7 Ovládání hry

První verze je připravena jako hra pro jednoho hráče.

Po zapnutí DPS stikneme tlačítko "Nová hra". V této chvíli se vygeneruje zadání, které není vidět, a na první herní LED se rozblíká kurzor. Kurzorem lze pohybovat pomocí tlačítek "Šipka vpravo" a "Šipka vlevo". Barvy LED se nastavují tlačítky ve spodní části DPS, které jsou barevně označené. Po ukončení tahu stiskneme tlačítko "Vyhodnocení". Proběhne vyhodnocení a zobrazí se na vyhodnocovacích LED. Kurzor se posune na první LED v dalším řádku. Po zadání správné kombinace barev a jejich pozic se rozsvítí zadání a hra je u konce. Pro novou hru stiskneme tlačítko "Nová hra" a pro ukončení tlačítko "Konec". Při stisku tlačítka "Konec" zhasnou všechny herní, vyhodnocovací LED i LED pro zadání. Poté je DPS připravena pro vypnutí vypínačem. V této fázi lze zároveň stisknout tlačítko "Nová hra".

Závěr

Byla vytvořena elektronická hra Logic. Tato hra vychází z původní deskové hry. Knoflíky jsou nahrazeny LED a ovládání je pomocí tlačítek.

Literatura

- [1] VUT v Brně: *Úprava, odevzdávání a zveřejňování vysokoškolských kvalifikačních prací na VUT v Brně* [online]. Směrnice rektora č.2/2009. Brno: 2009, poslední aktualizace 24. 3. 2009 [cit. 23. 10. 2015]. Dostupné z URL: [<https://www.vutbr.cz/uredni-deska/vnitрни-predpisy-a-dokumenty/smernice-rektora-f34920/>](https://www.vutbr.cz/uredni-deska/vnitрни-predpisy-a-dokumenty/smernice-rektora-f34920/).
- [2] ČSN ISO 690 (01 0197) *Informace a dokumentace – Pravidla pro bibliografické odkazy a citace informačních zdrojů*. 40 stran. Praha: Český normalizační institut, 2011.
- [3] ČSN ISO 7144 (010161) *Dokumentace – Formální úprava disertací a podobných dokumentů*. 24 stran. Praha: Český normalizační institut, 1997.
- [4] ČSN ISO 31-11 *Veličiny a jednotky – část 11: Matematické znaky a značky používané ve fyzikálních vědách a v technice*. Praha: Český normalizační institut, 1999.
- [5] BIERNÁTOVÁ, O., SKŮPA, J.: *Bibliografické odkazy a citace dokumentů dle ČSN ISO 690 (01 0197) platné od 1. dubna 2011* [online]. 2011, poslední aktualizace 2. 9. 2011 [cit. 19. 10. 2011]. Dostupné z URL: [<http://www.citace.com/CSN-ISO-690.pdf>](http://www.citace.com/CSN-ISO-690.pdf)
- [6] *Pravidla českého pravopisu*. Zpracoval kolektiv autorů. 1. vydání. Olomouc: FIN PUBLISHING, 1998. 575 s. ISBN 80-86002-40-3.
- [7] WALTER, G. G.; SHEN, X. *Wavelets and Other Orthogonal Systems*. 2. vyd. Boca Raton: Chapman & Hall/CRC, 2000. 392 s. ISBN 1-58488-227-1
- [8] SVAČINA, J. Dispersion Characteristics of Multilayered Slotlines – a Simple Approach. *IEEE Transactions on Microwave Theory and Techniques*, 1999, vol. 47, no. 9, s. 1826–1829. ISSN 0018-9480.
- [9] RAJMIC, P.; SYSEL, P. Wavelet Spectrum Thresholding Rules. In *Proceedings of the International Conference Research in Telecommunication Technology*, Žilina: Žilina University, 2002. s. 60–63. ISBN 80-7100-991-1.

Seznam symbolů, veličin a zkratk

Šířka levého sloupce Seznamu symbolů, veličin a zkratk je určena šířkou parametru prostředí `acronym` (viz řádek 1 výpisu zdrojáku na str. 43)

KolikMista pouze ukázka vyhrazeného místa

DPS Deska plošného spoje

LED Light Emitting Diode

THT Through-hole technology (Vývodová technologie součástek)

f_{vz} vzorkovací kmitočet

Seznam příloh

A	Některé příkazy balíčku <code>thesis</code>	39
A.1	Příkazy pro sazbu veličin a jednotek	39
A.2	Příkazy pro sazbu symbolů	39
B	Druhá příloha	41
C	Příklad sazby zdrojových kódů	43
C.1	Balíček <code>listings</code>	43
D	Obsah přiloženého CD	47

A Některé příkazy balíčku thesis

A.1 Příkazy pro sazbu veličin a jednotek

Tab. A.1: Přehled příkazů pro matematické prostředí

Příkaz	Příklad	Zdroj příkladu	Význam
<code>\textind{...}</code>	β_{\max}	<code>\$_\beta_\textind{max}\$</code>	textový index
<code>\const{...}</code>	U_{in}	<code>\$_\textconst{U}_\textind{in}\$</code>	konstantní veličina
<code>\var{...}</code>	u_{in}	<code>\$_\textvar{u}_\textind{in}\$</code>	proměnná veličina
<code>\complex{...}</code>	\textit{u}_{in}	<code>\$_\textcomplex{u}_\textind{in}\$</code>	komplexní veličina
<code>\vect{...}</code>	\mathbf{y}	<code>\$_\textvect{y}\$</code>	vektor
<code>\mat{...}</code>	\mathbf{Z}	<code>\$_\textmat{Z}\$</code>	matice
<code>\unit{...}</code>	kV	<code>\$_\textunit{kV}\$</code> či <code>\unit{kV}</code>	jednotka

A.2 Příkazy pro sazbu symbolů

- `\E`, `\eul` – sazba Eulerova čísla: e ,
- `\J`, `\jmag`, `\I`, `\imag` – sazba imaginární jednotky: j , i ,
- `\dif` – sazba diferenciálu: d ,
- `\sinc` – sazba funkce: sinc ,
- `\mikro` – sazba symbolu mikro stojatým písmem¹: μ ,
- `\uppi` – sazba symbolu π (stojaté řecké pí, na rozdíl od `\pi`, což sází π).

Všechny symboly jsou určeny pro matematický mód, vyjma `\mikro`, jenž je použitelný rovněž v textovém módu.

¹znak pochází z balíčku `textcomp`

B Druhá příloha



Obr. B.1: Zlepšené Wilsonovo proudové zrcadlo.

Pro sazbu vektorových obrázků přímo v \LaTeX je možné doporučit balíček `TikZ`. Příklady sazby je možné najít na `\TeX`ample. Pro vyzkoušení je možné použít programy `QTikz` nebo `TikzEdt`.

C Příklad sazby zdrojových kódů

C.1 Balíček listings

Pro vysázení zdrojových souborů je možné použít balíček `listings`. Balíček zavádí nové prostředí `lstlisting` pro sazbu zdrojových kódů, jako například:

```
\section{Balíček lstlistings}
Pro vysázení zdrojových souborů je možné použít
  balíček \href{https://www.ctan.org/pkg/listings}%
  {\texttt{listings}}.
Balíček zavádí nové prostředí \texttt{lstlisting} pro
  sazbu zdrojových kódů.
```

Podporuje množství programovacích jazyků. Kód k vysázení může být načítán přímo ze zdrojových souborů. Umožňuje vkládat čísla řádků nebo vypisovat jen vybrané úseky kódu. Např.:

Zkratky jsou sázeny v prostředí `acronym`:

```
6 \begin{acronym}[KolikMista]
```

Šířka textu volitelného parametru `KolikMista` udává šířku prvního sloupce se zkratkami. Proto by měla být zadávána nejdelší zkratka nebo symbol. Příklad definice zkratky f_{vz} je na výpisu C.1.

Výpis C.1: Ukázka sazby zkratek

```
25 \acro{symfvz} % název
26 [\ensuremath{f_{\text{vz}}}] % symbol
27 {vzorkovací kmitočet} % popis
```

Ukončení seznamu je provedeno ukončením prostředí:

```
26 [\ensuremath{f_{\text{vz}}}] % symbol
27 {vzorkovací kmitočet} % popis
28 %%% esymfvz
29
30 \end{acronym}
```

Poznámka k výpisům s použitím volby jazyka czech nebo slovak:

Pokud Váš zdrojový kód obsahuje znak spojovníku `-`, pak překlad může skončit chybou. Ta je způsobená tím, že znak `-` je v českém nebo slovenském nastavení balíčku `babel` tzv. aktivním znakem. Přepněte znak `-` na neaktivní příkazem `\shorthandoff{-}` těsně před výpisem a hned za ním jej vraťte na aktivní příkazem `\shorthandon{-}`. Podobně jako to je ukázáno ve zdrojovém kódu šablony.

Na výpisu C.2 naleznete příklad kódu pro Matlab, na výpisu C.3 zase pro jazyk C.

Výpis C.2: Příklad Schur-Cohnova testu stability v prostředí Matlab.

```
1 %% Příklad testování stability filtru
2
3 % koeficienty polynomu ve jmenovateli
4 a = [ 5, 11.2, 5.44, -0.384, -2.3552, -1.2288];
5 disp('Polynom:'); disp(poly2str(a, 'z'))
6
7 disp('Kontrola pomocí kořenů polynomu:');
8 zx = roots(a);
9 if( all( abs( zx) < 1))
10     disp('System je stabilní')
11 else
12     disp('System je nestabilní nebo na mezí stability');
13 end
14
15 disp(' '); disp('Kontrola pomocí Schur-Cohn:');
16 ma = zeros( length(a)-1, length(a));
17 ma(1,:) = a/a(1);
18 for( k = 1:length(a)-2)
19     aa = ma(k,1:end-k+1);
20     bb = fliplr(aa);
21     ma(k+1,1:end-k+1) = (aa-aa(end)*bb)/(1-aa(end)^2);
22 end
23
24 if( all( abs( diag( ma.'))))
25     disp('System je stabilní')
26 else
27     disp('System je nestabilní nebo na mezí stability');
28 end
```

Výpis C.3: Příklad implementace první kanonické formy v jazyce C.

<code>// první kanonická forma</code>	1
<code>short fxdf2t(short coef[][5], short sample)</code>	2
<code>{</code>	3
<code>static int v1[SECTIONS] = {0,0}, v2[SECTIONS] = {0,0};</code>	4
<code>int x, y, accu;</code>	5
<code>short k;</code>	6
	7
<code>x = sample;</code>	8
<code>for(k = 0; k < SECTIONS; k++){</code>	9
<code>accu = v1[k] >> 1;</code>	10
<code>y = _sadd(accu, _smpy(coef[k][0], x));</code>	11
<code>y = _sshl(y, 1) >> 16;</code>	12
	13
<code>accu = v2[k] >> 1;</code>	14
<code>accu = _sadd(accu, _smpy(coef[k][1], x));</code>	15
<code>accu = _sadd(accu, _smpy(coef[k][2], y));</code>	16
<code>v1[k] = _sshl(accu, 1);</code>	17
	18
<code>accu = _smpy(coef[k][3], x);</code>	19
<code>accu = _sadd(accu, _smpy(coef[k][4], y));</code>	20
<code>v2[k] = _sshl(accu, 1);</code>	21
	22
<code>x = y;</code>	23
<code>}</code>	24
<code>return(y);</code>	25
<code>}</code>	26

D Obsah přiloženého CD

Nezapomeňte uvést, co čtenář najde na přiloženém médiu. Je vhodné okomentovat obsah každého adresáře, specifikovat, který soubor obsahuje důležitá nastavení, který soubor je určen ke spuštění atd. Také je dobře napsat, v jaké verzi software byl kód testován (např. Matlab 2010b).

Pokud je souborů hodně a jsou organizovány ve více složkách, je možné pro výpis adresářové struktury použít balíček `dirtree`.

```
/ ..... kořenový adresář přiloženého CD
├── logo ..... loga školy a fakulty
│   ├── BUT_abbreviation_color_PANTONE_EN.pdf
│   ├── BUT_color_PANTONE_EN.pdf
│   ├── FEEC_abbreviation_color_PANTONE_EN.pdf
│   ├── FEKT_zkratka_barevne_PANTONE_CZ.pdf
│   ├── UTKO_color_PANTONE_CZ.pdf
│   ├── UTKO_color_PANTONE_EN.pdf
│   ├── VUT_barevne_PANTONE_CZ.pdf
│   ├── VUT_symbol_barevne_PANTONE_CZ.pdf
│   └── VUT_zkratka_barevne_PANTONE_CZ.pdf
├── obrazky ..... ostatní obrázky
│   ├── soucastky.png
│   ├── spoje.png
│   ├── ZlepseneWilsonovoZrcadloNPN.png
│   └── ZlepseneWilsonovoZrcadloPNP.png
├── pdf ..... pdf stránky generované informačním systémem
│   ├── student-desky.pdf
│   ├── student-titulka.pdf
│   └── student-zadani.pdf
├── text ..... zdrojové textové soubory
│   ├── literatura.tex
│   ├── prilohy.tex
│   ├── reseni.tex
│   ├── uvod.tex
│   ├── vysledky.tex
│   ├── zaver.tex
│   └── zkratky.tex
├── sablona-obhaj.tex ..... hlavní soubor pro sazbu prezentace k obhajobě
├── sablona-prace.tex ..... hlavní soubor pro sazbu kvalifikační práce
└── thesis.sty ..... balíček pro sazbu kvalifikačních prací
```