

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

SEMESTRÁLNÍ PRÁCE

Brno, 2020

Renata Zemanová



# 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** Renata Zemanová  
AUTHOR

**VEDOUCÍ PRÁCE** doc. Ing. Jiří Háze, Ph.D.  
SUPERVISOR

**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.

## **ABSTRAKT**

Cílem práce je navrhnut elektronickou hru Logic. Elektronická hra musí respektovat pravidla deskové hry. Dále by se hra měla i svým vzhledem podobat deskové hře. Elektronická hra je navržena jako jedna DPS, která obsahuje řídicí elektroniku i herní prvky. Byl vyroben prototyp hardwaru elektroniky, na němž byla otestována funkčnost zapojení.

## **KLÍČOVÁ SLOVA**

Elektronická hra Logic, pravidla deskové hry, ESP32-PICO, inteligentní LED WS2812C, USB, DPS

## **ABSTRACT**

The goal of this thesis is design an electronic game Logic. This electronic game should look like the board game and also respect the game's rules.

The electronic game is designed as one PCB that contains control electronics and game elements. A prototype of electronics hardware was made, on which the functionality of the connection was tested.

## **KEYWORDS**

Electronic game Logic, board game rules, ESP32-PICO, intelligent control LED WS2812C, USB, PCB

# Obsah

<b>Úvod</b>	<b>8</b>
<b>1 Pravidla deskové hry</b>	<b>9</b>
<b>2 Návrh elektroniky</b>	<b>10</b>
2.1 Řídicí elektronika . . . . .	10
2.2 Napájení . . . . .	11
2.3 Snižující měnič napájecího napětí pro mikrokontrolér . . . . .	12
2.4 Převodník z USB na RS-232 . . . . .	13
2.5 Herní prvky . . . . .	14
2.5.1 Rozdělení . . . . .	16
2.5.2 Převodník úrovně . . . . .	16
2.5.3 Zapínání napájení . . . . .	16
2.6 Spínací prvky . . . . .	17
2.7 Indikace přítomnosti napájecího napětí . . . . .	18
<b>3 Návrh DPS</b>	<b>20</b>
3.1 Vzhled DPS . . . . .	20
3.2 Funkční rozmístění součástek . . . . .	21
<b>4 Oživení DPS</b>	<b>23</b>
<b>5 Od prvního prototypu po finální verzi</b>	<b>24</b>
5.1 Napájení . . . . .	25
5.1.1 Verze 0.0 . . . . .	25
5.1.2 Verze 1.0 . . . . .	26
5.2 Vzhled DPS . . . . .	27
5.3 Oživení prototypů . . . . .	27
5.3.1 Verze 0.0 . . . . .	27
5.3.2 Verze 1.0 . . . . .	28
<b>6 Software</b>	<b>30</b>
6.1 Funkce init . . . . .	30
6.2 Funkce ovládání tlačítek . . . . .	31
6.3 Zadání . . . . .	33
<b>7 Způsob ovládání elektronické hry</b>	<b>35</b>
7.1 Hra pro jednoho hráče . . . . .	35
7.2 Hra pro dva hráče . . . . .	36

<b>8 Krabička</b>	<b>38</b>
<b>9 Kompletace</b>	<b>39</b>
<b>Závěr</b>	<b>40</b>
<b>Literatura</b>	<b>41</b>
<b>Seznam symbolů a zkratek</b>	<b>44</b>
<b>Seznam příloh</b>	<b>46</b>
<b>A Schéma zapojení DPS verze 1.0</b>	<b>47</b>
<b>B Výrobní podklady DPS verze 1.0</b>	<b>55</b>
<b>C Některé příkazy balíčku thesis</b>	<b>61</b>
C.1 Příkazy pro sazbu veličin a jednotek . . . . .	61
C.2 Příkazy pro sazbu symbolů . . . . .	61
<b>D Druhá příloha</b>	<b>62</b>
<b>E Příklad sazby zdrojových kódů</b>	<b>63</b>
E.1 Balíček listings . . . . .	63
<b>F Obsah elektronické přílohy</b>	<b>68</b>

# Seznam obrázků

1.1	Desková hra Logic [2] . . . . .	9
2.1	Blokové schéma elektroniky . . . . .	10
2.2	Schéma zapojení mikrokontroléru ESP32-PICO [4] . . . . .	11
2.3	Schéma zapojení konektoru USB-C [10] . . . . .	12
2.4	Schéma zapojení čipu SY8105 [16] . . . . .	12
2.5	Schéma zapojení převodníku z USB na RS-232 [3] . . . . .	14
2.6	Zapojení LED pro indikaci komunikace čipu CP2102 s mikrokontrolérem [17] . . . . .	15
2.7	Zapojení inteligentních LED WS2812C [18] . . . . .	15
2.8	Obvod pro zapínání napájení pro inteligentní LED . . . . .	17
2.9	Zapojení tlačítek . . . . .	18
2.10	Zapojení LED pro indikaci napájecího napětí . . . . .	19
3.1	Rozložení součástek kolem čipu SY8105 na DPS [16] . . . . .	22
5.1	Blokové schéma zapojení verze 0.0 . . . . .	24
5.2	Blokové schéma zapojení verze 1.0 . . . . .	24
5.3	Schéma zapojení čipu TP4056 [15] . . . . .	25
5.4	Verze 0.0 z výroby . . . . .	27
5.5	Oživená verze 0.0 s opravami . . . . .	28
5.6	3D pohled verze 1.0 DPS . . . . .	29
7.1	Vývojový diagram . . . . .	36
7.2	Vývojový diagram . . . . .	37
A.1	Blokové schéma DPS verze 1.0 . . . . .	47
B.1	Vrstva mědi TOP . . . . .	55
B.2	Vnitří vrstva mědi napájení . . . . .	56
B.3	Vnitří vrstva mědi GND . . . . .	57
B.4	Vrstva mědi BOTTOM . . . . .	58
B.5	Vrstva s popisky . . . . .	59
B.6	Celá DPS . . . . .	60
D.1	Alenčino zrcadlo . . . . .	62

# **Seznam tabulek**

2.1	Parametry čipu SY8105 [16]	13
2.2	Parametry čipu CP2102 [17]	13
2.3	Parametry inteligentních LED WS2812C [18]	15
2.4	Zapojení přepínačů ke strapping pins mikrokontroléru EPS32	18
5.1	Nastavení nabíjecího proudu rezitorem R13 [15].	26
C.1	Přehled příkazů	61

# **Úvod**

Tato práce se zabývá návrhem elektronické hry Logic. Tato hra by se měla co nejvíce podobat deskové hře a musí respektovat pravidla deskové hry. Hra Logic je určená pro všechny věkové kategorie hráčů. Podporuje a rozvíjí logické myšlení.

Hra musí být navržena jako přenosné zařízení, takže musí být kladen důraz na spotřebu jednotlivých použitých komponentů.

Hra Logic není komerčně dostupná v elektronické verzi. Existuje pouze ve verzi deskové hry pro dva hráče. Elektronická hra Logic je navržena pro jednoho hráče. Hra je jedinečná svými ovládacími prvky a způsobem hry.

V této práci je popsán návrh elektronické hry. Na začátku jsou popsána pravidla deskové hry, která jsou respektována i u elektronické hry. Elektronická hra Logic je tvořena DPS, která obsahuje veškeré potřebné komponenty pro funkci hry i pro její ovládání.

Na závěr je popsán způsob elektronické hry a její ovládání.

# 1 Pravidla deskové hry

Hra Logic je desková hra pro dva hráče [8]. Jeden hráč určí hledanou kombinaci, dále bude označován jako hráč A, a druhý tuto kombinaci za pomocí logických úvah a vyhodnocení hráčem A hledá, dále bude označován jako hráč B.

Hráči si určí herní pozice. Hráč A vybere barevné kolíky dle libosti a následně je v určitém pořadí vloží do zadávacího pole. Zadání zakryje stříškou, aby tuto kombinaci spoluhráč neviděl. Hráč B se v tuto chvíli nedívá. Hráč B následně zvolí libovolnou kombinaci barev a jejich pozic. Po ukončení tahu nechá hráče A, aby jeho 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 se dané barvy v zadání nevyskytují. Vyhodnocovací kolíky umisťuje od kraje, nejprve černé a pak bílé, aby hráči B nebylo jasné, kterých hracích kolíků se vyhodnocení týká [8]. Poté začne hráč B na základě vyhodnocení a svých všech předchozích tahů hledat správnou kombinaci.

Hráč B má maximálně 10 pokusů na zjištění správné kombinace. Po skončení hry hráč A odkryje stříšku a ukáže hledanou kombinaci. Hra skončí po určení správné kombinace, nebo po vypotřebování všech pokusů.

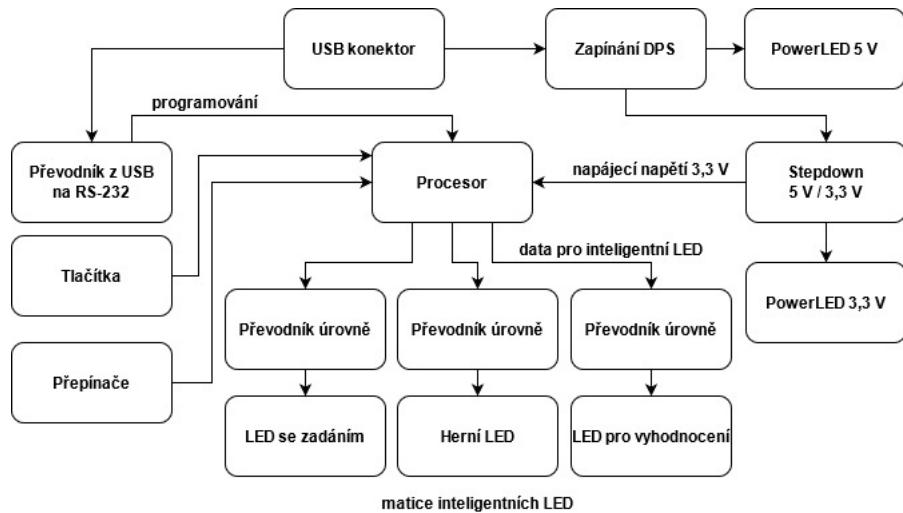
Hru lze hrát ve více variantách. Hráči se mohou domluvit, zda zadání může, nebo nesmí obsahovat volnou pozici. Zároveň existuje více variací této hry. Většinou se liší v délce hledané kombinace.



Obr. 1.1: Desková hra Logic [2].

## 2 Návrh elektroniky

V následující kapitole budou rozebrány všechny komponenty DPS Elektronické hry Logic. Návrh se skládá z řídící elektroniky, napájení, herních prvků a dalších potřebných komponent.



Obr. 2.1: Blokové schéma elektroniky.

### 2.1 Řídící elektronika

Jako řídící elektronika byl vybrán mikrokontrolér ESP32-PICO-D4. Hlavním důvodem pro výběr tohoto mikrokontroléru bylo, že je téměř totožný jako ESP32-WROOM, se kterým mám dlouhodobé zkušenosti. Tento mikrokontrolér má veškeré periferie, které jsou pro výrobu této hry zapotřebí.

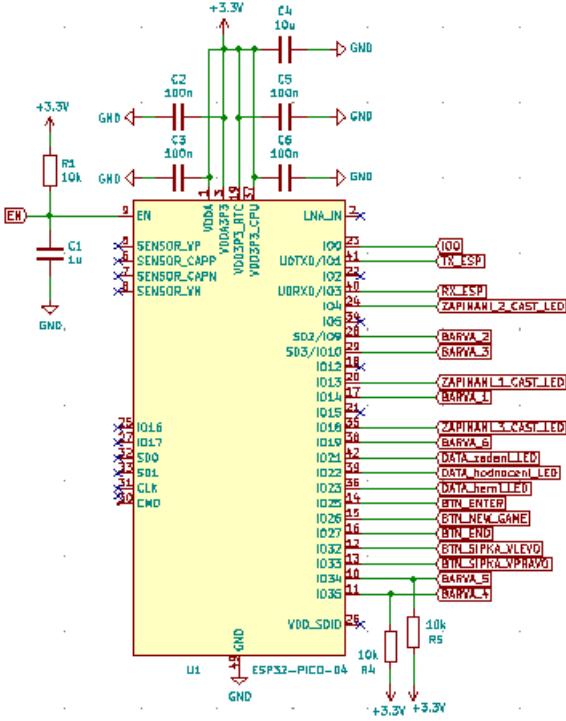
Mikrokontrolér ESP32-PICO také podporuje Arduino framework, díky kterému bylo programování značně zjednodušeno.

Mikrokontrolér ESP32-PICO obsahuje [4]:

- WiFi,
- Bluetooth,
- 32 GPIO pinů,
- dvoujádrový 32bitový procesor Xtensa LX6,
- 520 kB SRAM,
- 4 MB FLASH.

Napájecí napětí tohoto mikrokontroléru je od 3,0 do 3,6 V a průměrný odběr proudu je 80 mA [4]. ESP32-PICO má vyvedeno 32 GPIO pinů, které je možno softwarově nastavit jako vstupní nebo výstupní. Na tyto piny lze poté připojit různá

zařízení. Vstupním senzorem může být typicky tlačítko a výstupním indikátorem např. LED. Tato zařízení zprostředkovávají komunikaci mezi mikrokontrolérem a okolním světem.



Obr. 2.2: Schéma zapojení mikrokontroléru ESP32-PICO [4].

GPIO piny IO16 a IO17 nemohou být použity, protože ESP32-PICO má na těchto pinech připojenou flash paměť [4]. Pokud by na tento pin bylo připojeno nějaké zařízení, tak by mikrokontrolér ztratil přístup ke své paměti.

GPIO piny IO02, IO05, IO12 a IO15 slouží jako konfigurační piny při spouštění nebo resetu mikrokontroléru ESP32-PICO. Slouží například k výběru paměti, ze které se mikrokontrolér ESP32-PICO načítá. Při resetu musí být tyto piny ve správné konfiguraci.

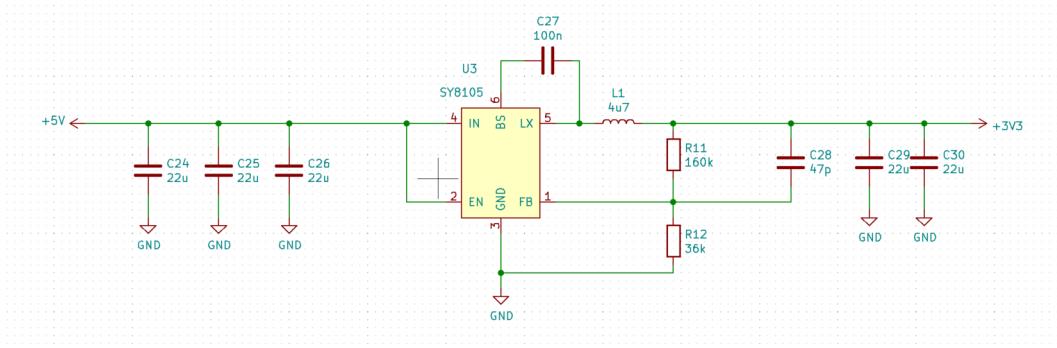
Pin MTDI je GPIO12 a pin MTDO je pin GPIO15 [4].

GPIO piny IO34 a vyšší jsou pouze vstupní [4]. Vstupní piny nemají softwarově zapojitelný pullup rezistor. Pokud je tedy zapotřebí pullup rezistor, musí se fyzicky zapojit.

## 2.2 Napájení

Napájení probíhá před USB konektorem. V dnešní době jsou 2 nejpoužívanější konektory – USB Micro a USB-C. Proto byly navrženy oba USB konektory. DPS ne-

podporuje funkci Quick Charge ani Power Delivery. DPS může být napájena pouze napětím 5 V. Každé jiné napětí by zbytečné komplikovalo návrh DPS. Konektor USB-C je tedy funkčně zapojen totožně jako USB Micro a oba konektory jsou na DPS pouze kvůli jejich tvaru.

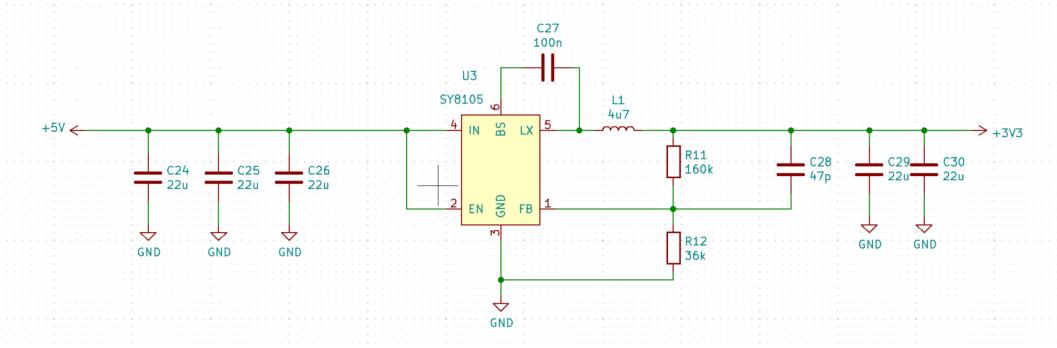


Obr. 2.3: Schéma zapojení konektoru USB-C [10].

## 2.3 Snižující měnič napájecího napětí pro mikrokontrolér

Mikrokontrolér ESP32-PICO má napájecí napětí 3,3 V. Napětí z powerbanky přes USB je 5 V. Proto je tedy zapotřebí zapojit měnič pro snížení napětí, který bude vytvářet z napájecího napětí 5 V napájecí napětí 3,3 V pro mikrokontrolér ESP32-PICO.

Jako měnič byl zvolen čip SY8105. Tento čip byl vybrán podle projektu RB0005-UniversalStepDown [14]. V tomto projektu zapojení funguje bezproblémově, a proto bylo rozhodnuto toto zapojení použít v mé práci.



Obr. 2.4: Schéma zapojení čipu SY8105 [16].

Výstupní napětí zapojení čipu SY8105 je závislé na poměru rezistorů R11 a R12 [16]. Hodnota rezistoru R11 byla zvolena  $160\text{ k}\Omega$ . Hodnota rezistoru R12 byla dopočítána dle následující rovnice.

$$R_{12} = \frac{0,6}{U_{OUT} - 0,6} \cdot R_{11} = \frac{0,6}{3,3 - 0,6} \cdot 160 \cdot 10^3 = 35,555\text{ k}\Omega \quad [16] \quad (2.1)$$

Hodnota rezistoru R12 byla zvolena nejbližší z vyráběné rezistorové řady E24 [13], tudíž  $36\text{ k}\Omega$ . Hodnota rezistoru se musí co nejvíce blížit vypočítané hodnotě, aby výstupní napětí nebylo příliš odlišné od požadovaného. Proto nebyl zvolen rezistor z častěji používaných řad E6 ani E12. Hodnota rezistoru z těchto řad by se již příliš odchylovala od vypočítané.

Tab. 2.1: Parametry čipu SY8105 [16]

Vstupní napětí	4,5–18 V
Maximální výstupní proud	5 A

## 2.4 Převodník z USB na RS-232

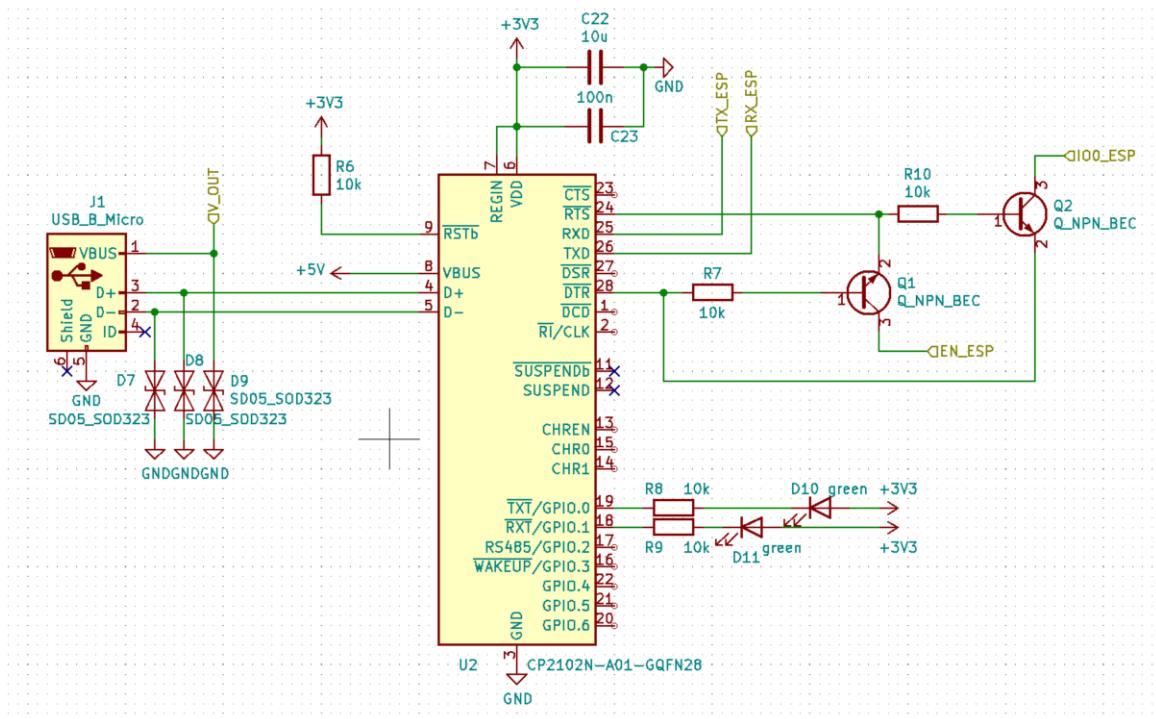
Mikrokontrolér ESP32-PICO používá jako komunikační rozhraní linku RS-232. Programování ale probíhá přes USB konektor, který toto rozhraní nemá. Proto bylo potřeba použít převodník z USB na rozhraní RS-232.

Velkou inspirací při výběru součástek byl kit ESP32-DEVKITC [3]. Tento kit obsahuje ESP32-WROOM a také převodník z USB na RS-232. Na tomto kitu je převodník realizován čipem CP2102. ESP32-WROOM a ESP32-PICO se liší pouze v drobnostech. Proto byl převodník CP2102 použit i při návrhu převodníku u elektronické hry Logic. Tento čip zároveň převádí logiku z 0–5 V na logiku 0–3,3 V [17].

Tab. 2.2: Parametry čipu CP2102 [17]

Typický odběr proudu	9,5 mA
Napájecí napětí	3–3,6 V

Čip CP2102 dokáže komunikovat velkým množstvím komunikačních rychlostí (300, 9600, 19200, 38400, 115200, 256000 Bd, atd) [17]. Počítač započne komunikaci



Obr. 2.5: Schéma zapojení převodníku z USB na RS-232 [3].

určitou rychlosťí a tento čip komunikaci zachytí, určí rychlosť a touto rychlosťí začne probíhat programování.

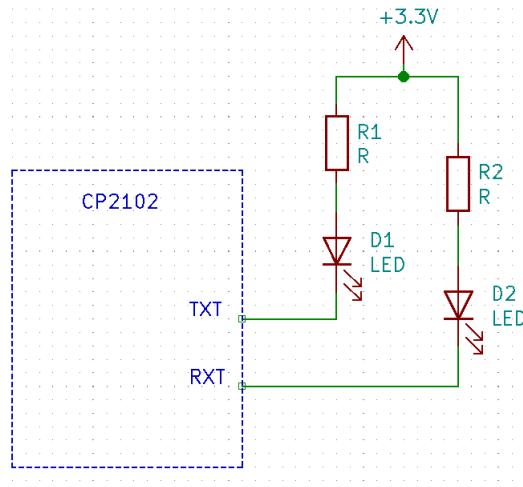
Z USB jsou signály D+ a D- připojeny k čipu CP2102. Tento čip signál z USB převede na signály RX a TX, které mají výstup na pinech RXD a TXD. Následně jsou tyto signály připojeny k mikrokontroléru ESP32-PICO. Signály RX a TX musí být překříženy – RX CP2102 je připojeno na TX ESP32-PICO a TX CP2102 je připojeno na RX ESP32-PICO.

LED D10 a D11 slouží k indikaci komunikace s mikrokontrolérem ESP32-PICO. Pokud je do mikrokontroléru nahráván program, tak LED D10 a D11 blikají.

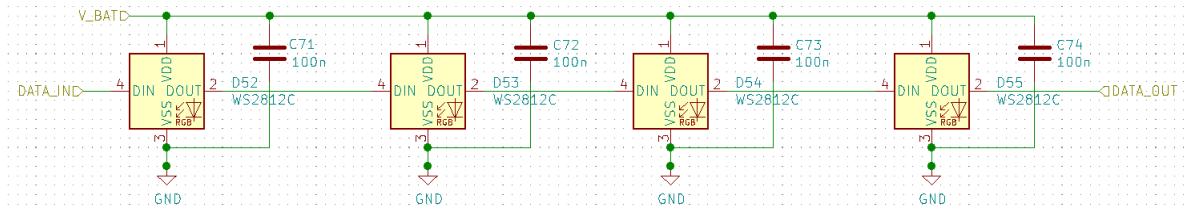
## 2.5 Herní prvky

Herními prvky elektronické hry Logic jsou inteligentní LED typu WS2812C. Tento typ inteligentních LED je určen pro přenosná zařízení díky jejich nízké spotřebě. Jsou také plně kompatibilní s typem WS2812B [18]. K těmto inteligentním LED existují knihovny, které usnadňují softwarovou práci s nimi.

Každá inteligentní LED má v sobě procesor, který slouží pro zpracování dat. Inteligentní LED WS2812C se zapojují za sebou přes piny DATA IN a DATA OUT.



Obr. 2.6: Zapojení LED pro indikaci komunikace čipu CP2102 s mikrokontrolérem [17].



Obr. 2.7: Zapojení inteligentních LED WS2812C [18].

Každá inteligentní LED převeze data z pinu DATA IN, která jsou pro ni určena, a zbytek pošle ven přes pin DATA OUT. K pinu DATA OUT je připojen pin DATA IN další inteligentní LED.

Tab. 2.3: Parametry inteligentních LED WS2812C [18]

Napájecí napětí	3,5–5,3 V
Výstupní napětí	$(VDD - 0,5) - (VDD + 0,5)$ V
Typický odběr proudu	5 mA
Klidový odběr proudu	0,3 mA

Ke každé inteligentní LED je připojen na napájení filtrační kondenzátor [18], aby LED svítily kontinuálně a nedostal se jim na napájení žádný šum.

## **2.5.1 Rozdělení**

Inteligentní LED jsou rozděleny do tří skupin. Skupina inteligentních LED pro zadání, skupina inteligentních LED pro herní pole a skupina inteligentních LED pro zobrazení vyhodnocení tahu. Skupina inteligentních LED pro zadání obsahuje 4 LED a skupiny pro herní pole a pro vyhodnocení každá 40 LED.

## **2.5.2 Převodník úrovně**

Inteligentní LED WS2812C pracují s logikou od 0 V do 5 V. Mikrokontrolér ESP32-PICO pracuje s logikou od 0 V do 3,3 V. Z tohoto důvodu byl mezi mikrokontrolér a inteligentní LED na datový signál pro inteligentní LED zapojen převodník úrovně, tzv. level shifter.

Převodník je realizován MOSFET tranzistorem a dvěma pull-up rezistory. Jeden rezistor je připojen k napájecímu napětí 3,3 V a druhý k napájecímu napětí 5 V. Jedná se o nejjednodušší způsob převodníku úrovně směrem nahoru. Tento způsob umožňuje komunikaci oběma směry. MOSFET tranzistor byl zvolen pro nulovou spotřebu, narozdíl od bipolárních tranzistorů.

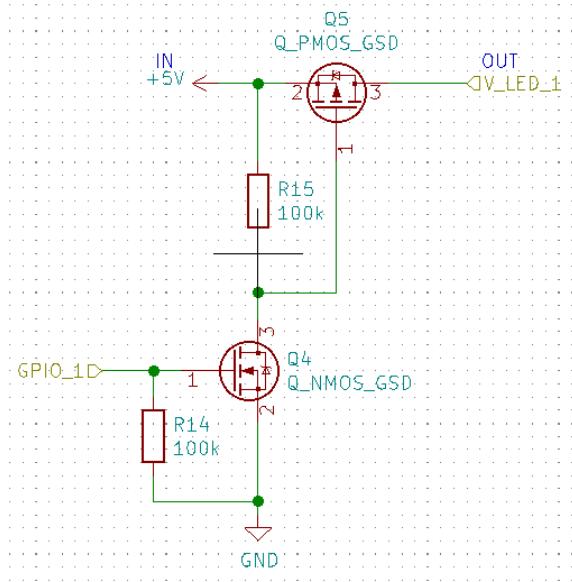
Pokud jsou obě zařízení v klidu (nekomunikují), tak je tranzistor zavřený a pull-up rezistory zajišťují, že na pinu mikrokoltroléru i inteligentní LED je logická 1. Pro mikrokontrolér ESP32-PICO je to tedy 3,3 V a pro inteligentní LED WS2812C je to 5 V. Jakmile započne mikrokontrolér komunikaci, to znamená, že svůj stav přepne do stavu logické nuly (0 V), tak rozdíl mezi gate a source tranzistoru stoupne a na výstupu tranzistoru bude také logická nula (0 V). Pokud by komunikace mohla probíhat i naopak (inteligentní LED by odesílala data do mikrokoltroléru), tak by byl princip totožný.

## **2.5.3 Zapínání napájení**

DPS je navrhována pro přenosnou aplikaci, a proto je potřeba zajistit její co nejnižší odběr.

Inteligentní LED WS2812C mají spotřebu 0,3 mA ve vypnutém stavu. Proto je herní pole dohromady s vyhodnocovacími LED rozděleno na 3 části. Tyto 3 části mají rozděleno napájení, které je postupně zapínáno dle potřeby. Do první části patří inteligentní LED se zadáním a první 4 čtverice inteligentních LED z herního pole a z vyhodnocení. Do druhé části patří další 3 čtverice inteligentních LED z herního a vyhodnocovacího pole. Do třetí části patří poslední 3 čtverice inteligentních LED z herního a z vyhodnocovacího pole.

Ke spínání slouží obvody s MOSFET tranzistory. MOSFET tranzistory byly zvoleny pro jejich nulovou spotřebu, narozdíl od bipolárních tranzistorů.



Obr. 2.8: Obvod pro zapínání napájení pro inteligentní LED.

Napájení inteligentních LED nelze spínat pouze jedním tranzistorem, protože logická 1 mikrokontroléru ESP32-PICO má hodnotu pouze 3,3 V. V tomto případě je zapotřebí spínat 5 V. Pokud by byl pro spínání použit jeden tranzistor, tak by bylo napájení vždy sepnuto.

GPIO pin mikrokontroléru ESP32-PICO je nastaven do logické 0, dokud není potřebné přivedení napájecího napětí dané skupině. Při logické nule na gate tranzistoru Q4 je tranzistor zavřený. Tranzistor Q5 v tomto okamžiku drží zavřený rezistor R15.

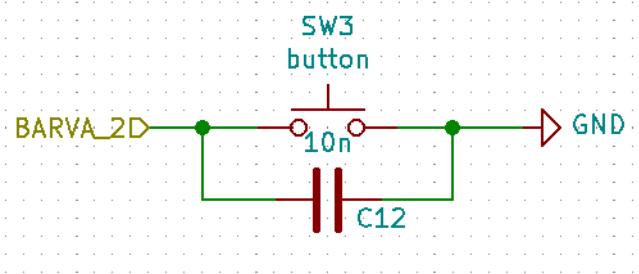
Když je GPIO pin mikrokontroléru přepnuto do logické 1, tak se tranzistor Q4 otevře. Otevřením tranzistoru Q4 je gate tranzistoru Q5 připojen ke GND a tím se otevře i tranzistor Q5, kterým je sepnuto napájecí napětí dané skupině inteligentních LED.

Rezistor R14 udržuje tranzistor Q4 zavřený při nestandardních stavech pinu mikrokontroléru, jako je např. při resetu procesoru.

## 2.6 Spínací prvky

Přepínač SW1 slouží pro zapínání celé DPS. Tento přepínač připojuje napájecí napětí 5 V z USB k celému zbytku DPS.

Tlačítka slouží pro ovládání hry. Ke každému tlačítku je připojen kondenzátor o hodnotě 100 nF. Tento kondenzátor slouží pro filtraci zákmitů při zmáčknutí tlačítka. Filtrace se proto nemusí řešit softwarově.



Obr. 2.9: Zapojení tlačítek.

Přepínače slouží pro volbu možnosti hry. První přepínač slouží pro volbu hry pro jednoho, nebo pro 2 hráče. Druhý slouží pro možnosti hry na 3 nebo 4 herní prvky a poslední přepínač slouží pro nastavení, zda se v zadání může, nebo nesmí vyskytnout mezera.

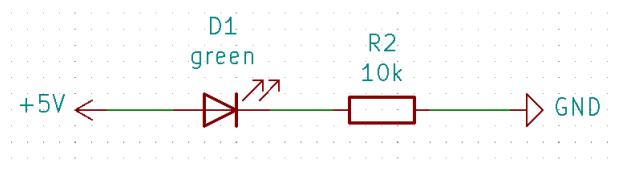
Z důvodu nedostatku pinů musely být přepínače připojeny k mikrokontroléru ESP32 přes konfigurační piny. U konfiguračních pinů musejí být dodržena pravidla pro připojovaná zařízení. Z těchto důvodů je jeden přepínač připojen přes pull-down rezistor a ostatní dva přepínače jsou připojeny přes pull-up rezistor.

Tab. 2.4: Zapojení přepínačů ke strapping pins mikrokontroléru EPS32

Přepínač	Pin mikrokontroléru	Typ připojení rezistoru
SW14	GPIO15	Pull-up
SW13	GPIO05	Pull-up
SW15	GPIO02	Pull-down

## 2.7 Indikace přítomnosti napájecího napětí

Pro indikaci přítomnosti napájecího napětí slouží LED D1 a D2. LED D1 indikuje přítomnost napájecího napětí 5 V a LED D2 indikuje přítomnost napájecího napětí 3,3 V. Obě LED mají zelenou barvu a jsou připojeny přes rezistor o hodnotě  $10\text{ k}\Omega$ .



Obr. 2.10: Zapojení LED pro indikaci napájecího napětí.

### 3 Návrh DPS

DPS je navržena v programu KiCad a její parametry jsou určeny pro výrobu i osazení ve firmě JLCPCB [5] [6]. Výrobní podklady proto musely být navrženy v souladu s jejich výrobními možnostmi [7].

DPS má 4 vrstvy. Vnitřní vrstvy slouží pro napájení a vnější pro signálové dráhy. V jedné vnitřní vrstvě je po celé její ploše polygon GND a ve druhé vnitřní vrstvě jsou polygony jednotlivých napájecích napětí. Součástky jsou rozloženy tak, aby pokud možno korespondovaly s polygony a napájecí napětí tak nemuselo být dále rozváděno zbytečně po celé DPS. Někdy však nemohlo být umístění přizpůsobeno a napájecí napětí tak muselo být přivedenou dráhou v horní nebo spodní vstvě, která jsou převážně určeny pro signálové dráhy.

Na vrchní straně jsou umístěny plošky pro osazení součástek, protože firma JLCPCB osazuje součástky pouze z jedné strany DPS.

Signálové dráhy jsou vedeny tenkou dráhou a napájecí dráhy jsou vedeny širší dráhou. V signálových drahách tečou zanedbatelné proudy, proto mohou být co nejtenčí. Výrobce umožňuje vyrobit u čtyřvrstvé DPS nejtenčí dráhu 0,09 mm [7]. Aby nebyly použity krajní hodnoty, byla zvolena šířka signálové dráhy 0,15 mm.

Dráhy, kterými je veden napájecí napětí, mají tloušťku 0,5 mm. Pro výpočet šířky dráhy byla použita online kalkulačka Printed Circuit Board Width Tool [1]. Proud dráhami při použití všech inteligentních LED je 420 mA a tloušťka mědi 35  $\mu$ A [7]. Doporučená šířka dráhy byla tedy 0,236 mm, a proto byla zvolena šířka dráhy 0,5 mm.

$$n = 84 \text{ ks}$$

$$I_{LED} = 5 \text{ mA}$$

$$I = I_{LED} \cdot n = 5 \cdot 84 = 420 \text{ mA} \quad (3.1)$$

Vnější vrstvy byly vyplněny polygonem GND. Zejména tak bylo učiněno kvůli výrobnímu procesu nanášení nepájivé masky. Tato maska může být velmi tenká zejména na plochách, kde jsou pouze tenké dráhy a žádné velké plochy. Vytvořením polygonu GND tak dopomáhá k vytvoření pevné a stejně tlusté nepájivé masky.

#### 3.1 Vzhled DPS

Vzhled DPS byl ovlivněn vzhledem deskové hry a výrobními možnostmi firmy JLCPCB. Firma JLCPCB vyrábí DPS o maximálních rozměrech  $10 \times 10 \text{ cm}$  [6]. Tato DPS tedy byla navržena na maximální vyrobitevnou velikost a poté byly do těchto rozměrů rozmístovány součástky.

Inteligentní LED WS2812C jsou rozděleny do 3 skupin, aby se hra co nejvíce podobala deskové hře. Zároveň dopomáhá orientaci ve hře. Také kvůli množství inteligenčních LED a jejich poměrně velkým rozměrům ( $5 \times 5$  mm) zabírají velkou část DPS [18]. Proto musely být inteligenční LED rozmístěny jako první.

Inteligentní LED pro zadání jsou umístěny v horní části DPS. V levém sloupci pod zadáním se nachází matice inteligenčních LED, které slouží jako herní pole, a v pravém sloupci je matice inteligenční LED pro vyhodnocení tahu. Matice reprezentují herní tahy, proto jsou v rozložení ( $4 \times 10$  LED).

Rozmístění tlačítek a přepínačů probíhalo dle rozmyšlenému způsobu ovládání elektronické hry. Byl také brán ohled na estetiku vzhledu. Proto byly tlačítka i inteligenční LED rozmístěny dle přesné mřížky. Následně byly ostatními obvody využívány zbylé prostor.

Dráhy s napájecím napětím by měly být co nejkratší a měly by být vedeny co nejvíce zpríma. Proto byla snaha o umístění součástek se stejným napájecím napětím k sobě. Ne vždy ale mohla být tato doporučení splněna. Někdy musel být proveden kompromis mezi vzhledem a funkčním zapojením. Dlouhé obvody byly rozmístěny v následujícím pořadí - USB konektory, měnič napětí, převodník z USB na RS-232 a ESP32-PICO. USB konektor musel být umístěn na okraj DPS. Proto byly jednotlivé části obvodu naskládány podél levého okraje DPS.

USB konektory jsou umístěny těsně vedle sebe. Tímto způsobem je ošetřeno, že za pomoci běžného USB kabelu, nepůjde zapojit napájení do obou konektorů zároveň.

Převodník úrovně je umístěn vždy těsně před vstupem signálu do první inteligenční LED daného řetězce.

Přepínač pro změnu hry pro jednoho a pro 2 hráče byl umístěn do mezery, mezi herní a vyhodnocovací inteligenční LED. Další 2 přepínače, jeden pro volbu hry na 3 nebo 4 herní prvky a druhý na volbu nastavení, zda se v zadání může, nebo nesmí vyskytnout mezera, jsou umístěny v pravém horním rohu.

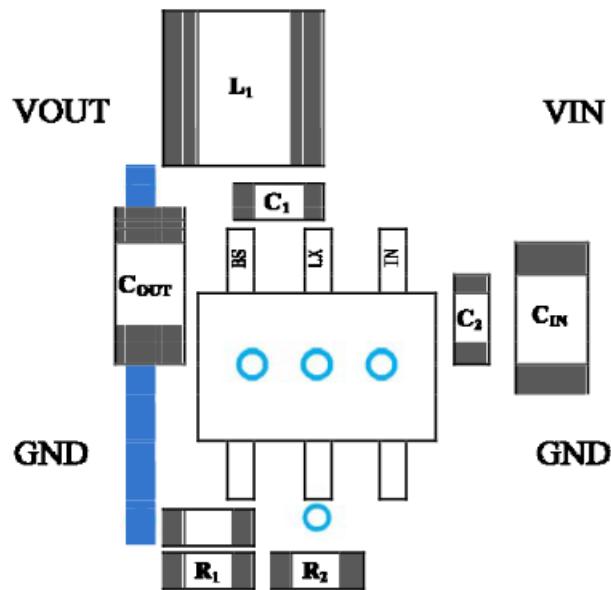
Obvody pro zapínání napájení jednotlivým sekčním inteligenčním LED jsou umístěny vždy na začátku dané části polygonu vlevo od herních inteligenčních LED.

## 3.2 Funkční rozmístění součástek

U některých součástek není jejich umístění na DPS lhostejné. Rozmístění součástek v některých případech může ovlivnit funkčnost obvodu. Rozmístění součástek často souvisí s rozkreslením ve schématu. Například kondenzátory filtrující napájení se umisťují co nejbliže přívodu napájení.

Rozložení součástek měniče napětí na DPS může velmi ovlivnit jeho funkčnost,

zejména pak jeho účinnost. Proto bylo rozložení a zapojení součástek převzato z datasheetu.



Obr. 3.1: Rozložení součástek kolem čipu SY8105 na DPS [16].

Signály D+ a D- od USB k čipu CP2102 jsou diferenciální páry, a proto tomu musely být přizpůsobeny i dráhy těchto signálů. Dráhy jsou vedeny vedle sebe a blízko u sebe po celou jejich délku.

Kondenzátory u mikrokontroléru ESP32-PICO a u čipu CP2102 musí být umístěny co nejblíže jejich pouzdru. Tyto kondenzátory slouží pro filtraci šumu na napájení. Stejná pravidla platí také pro filtrování kondenzátorů u inteligentních LED WS2812C.

Kondenzátory u tlačítek jsou také umístěny co nejblíže pouzdru tlačítka. Čím blíže pouzdru kondenzátor bude, tím lépe budou filtrovány zákmity při stisku tlačítka.

## 4 Oživení DPS

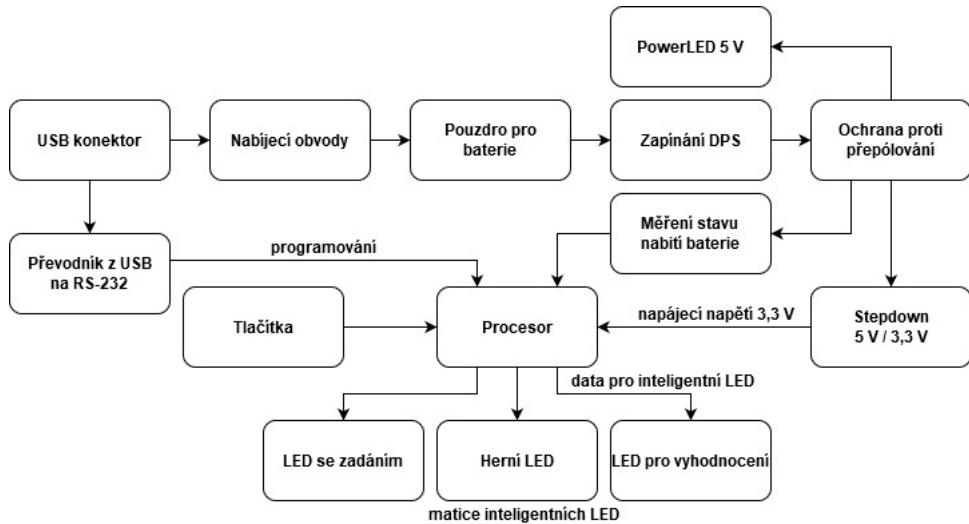
Při výrobě finální verze DPS elektronické hry Logic byly využity maximální možnosti osazení firmou JLCPCB. Pro oživení finální DPS je tedy zapotřebí pájení pouze vypínače, který není umístěn na DPS, ale je zasazen do krabičky a zapájen do DPS pomocí drátových prodlužek.

Při objednání finální DPS nastaly problémy s nedostatkem součástek, se kterým se v této době potýká mnoho návrhářů. Čip SY8105 neměla firma JLCPCB dlouho-době na skladě a nebylo možné čekat, než jej znova naskladní. Z tohoto důvodu byl na finální DPS objednán tento čip externě a doosazen ručně. Za klasických okolností by byl ale i tento čip osazen firmou JLCPCB.

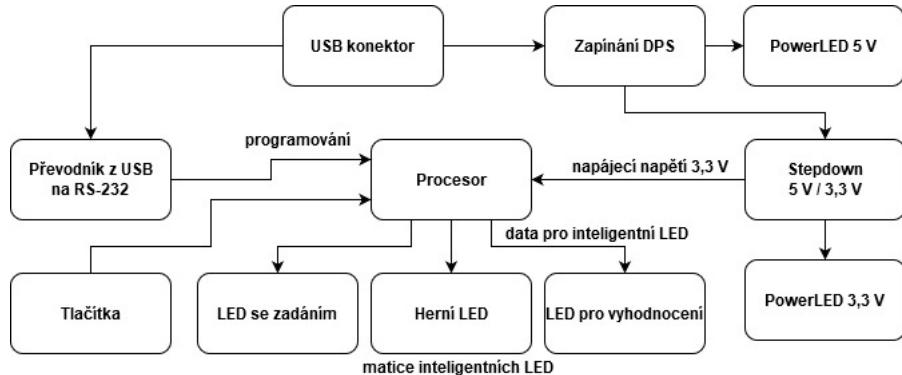
Po připojení DPS přes konektor USB-C nebo USB Micro k powerbance, nebo do počítače, se rozsvítí LED D1 a D2, které indikují přítomnost napájecího napětí. LED D1 značí přítomnost napětí 5 V z powerbanky a LED D2 značí správnou funkci zapojení měniče napětí, tedy přítomnost napětí 3,3 V.

## 5 Od prvního prototypu po finální verzi

Již byly navrženy 2 verze DPS elektronické hry Logic. Verze 0 byla pouze testovací, takže neobsahovala celé herní pole. Sloužila pro ověření funkčnosti daných zapojení různých čipů a zároveň pro testování softwaru.



Obr. 5.1: Blokové schéma zapojení verze 0.0.



Obr. 5.2: Blokové schéma zapojení verze 1.0.

Asi největším rozdílem mezi verzí 0 a 1 je způsob napájení. Verze 0 je napájena z baterií a verze 1 je napájena z USB. Díky této změně odpadly i mnohé části obvodu. Díky absenci baterií bylo možné postrádat nabíjecí obvody, ochranu proti přepólování a měření napětí na bateriích.

## 5.1 Napájení

### 5.1.1 Verze 0.0

V první verzi byly pro napájení použity baterie Li-Ion INR18650-29E od firmy Samsung. Tyto baterie mají jmenovité napětí 3,7 V, při plném nabití až 4,2 V. Kapacita jedné baterie je 2850 mAh [11]. Byly použity 2 články baterií, které byly zapojeny paralelně. Paralelním zapojením se celková kapacita zdvojnásobila.

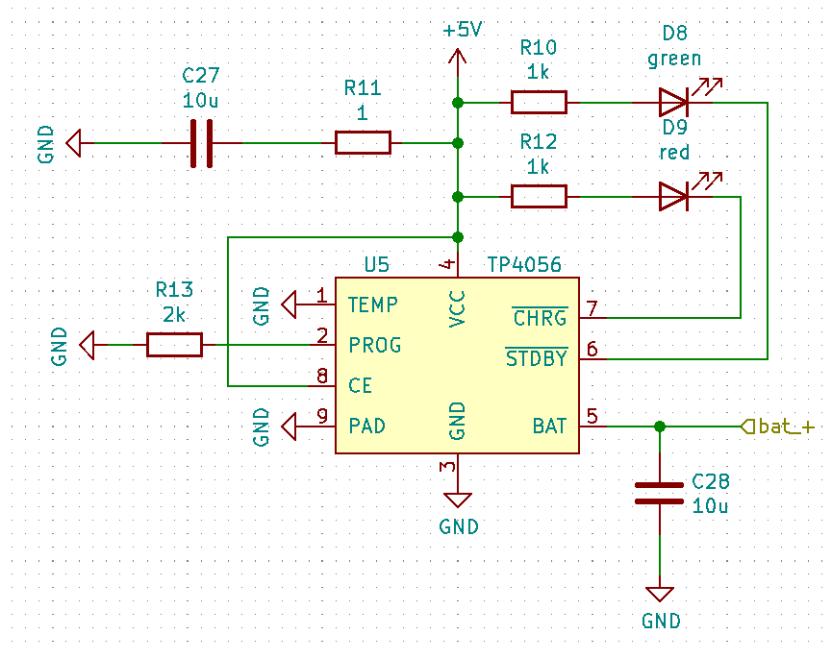
Baterie typu Li-Ion jsou náchylné na podvybití. Verze 0.0 proto obsahovala i měření napětí na bateriích. Pokud by došlo k podvybití, mohla by se bateriím zmenšovat kapacita, nebo by se mohly zničit.

Pro baterie bylo vybráno pouzdro na 2 články s THT montáží do DPS [12].

Baterie byly připevněny a zapájeny přímo do DPS, a proto musel být integrován i nabíjecí obvod. Nabíjení probíhalo přes konektor USB Micro.

Pro nabíjecí obvod byl zvolen čip TP4056. Tento obvod byl vybrán, protože je přímo určen pro nabíjení baterií typu Li-Ion. Existují moduly pro nabíjení těchto baterií, které mají integrovaný tento čip [9]. Tyto moduly jsou často používané, a proto bylo zapojení tohoto modulu převzato do této práce.

Vlastnosti čipu jsou přizpůsobeny bateriím Li-Ion. Nabíjení baterií probíhá do 4,2 V, což je maximální napětí na použitých bateriích [11] [15]. Díky tomu nemůže dojít ke zničení baterií nabíjením.



Obr. 5.3: Schéma zapojení čipu TP4056 [15].

Červená LED D9 indikuje nabíjení baterií a zelená LED D8 svítí, pokud jsou baterie nabité [15]. Pokud nesvítí ani červená ani zelená LED, znamená to, že baterie jsou podvybité, baterie nejsou vloženy a nebo je příliš vysoká, nebo nízká jejich teplota [15].

Rezistorem R13 se určuje nabíjecí proud baterií [15]. Aby baterie nabíjením nebyly poškozovány, měl by být nabíjecí proud maximálně 0,5 C (polovina kapacity baterie). Vybraná baterie má kapacitu 2850 mAh. Nabíjecí proud proto může být až 1425 mA. Jelikož je pro tuto hru důležitější kapacita baterie, než doba nabíjení, byl zvolen nabíjecí proud pouze 0,5 A. Čím menším proudem je baterie nabíjena, tím má delší životnost a pomaleji ztrácí svoji kapacitu.

Tab. 5.1: Nastavení nabíjecího proudu rezitorem R13 [15].

R13 [kΩ]	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

Byl zvolen rezistor o hodnotě 2 kΩ, kterým byl určen nabíjecí proud 580 mA.

### 5.1.2 Verze 1.0

Ve verzi 1.0 byl změněn způsob napájení. Napájení v této verzi neprobíhá přes baterie, ale pouze přes USB konektor, přes který ve verzi 0.0 probíhalo pouze nabíjení baterií.

Výhody

- absence drahých baterií,
- absence nabíjecího obvodu,
- absence hlídání stavu nabité baterie,
- absence ochrany proti přepólování (USB konektor je uzpůsoben svým tvarem, aby uživatel nemohl napájení přepólovat.),
- napájení inteligenčních LED napětím přímo z USB (Inteligenční LED mají menší odběr proudu.).

## Nevýhody

- předpokládá se, že uživatel vlastní powerbanku,
- výdrž závisí na kapacitě powerbanky.

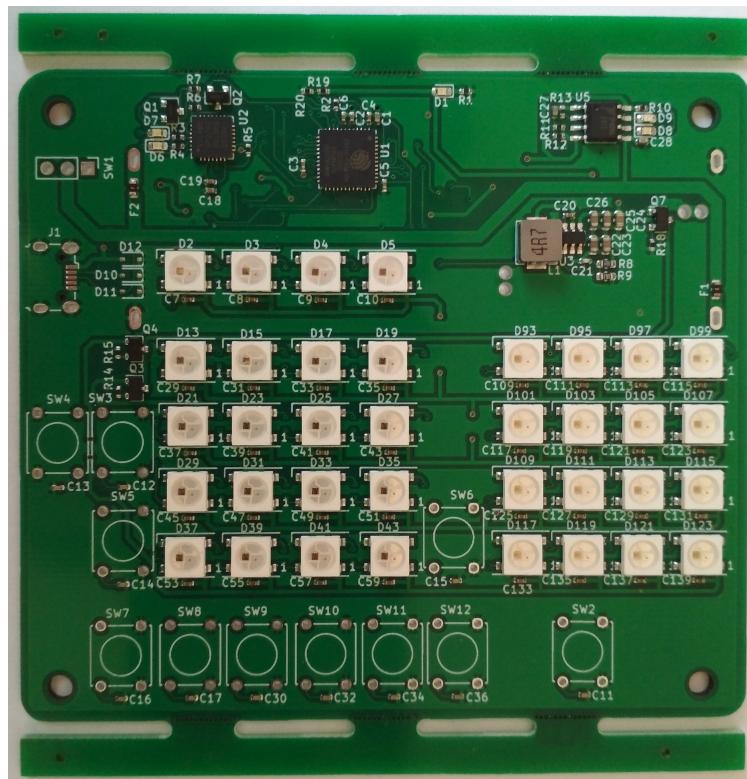
Byl zvolen konektor USB Micro, protože se jedná o nejrozšířenější USB konektor dnešní doby.

## 5.2 Vzhled DPS

### 5.3 Oživení prototypů

#### 5.3.1 Verze 0.0

DPS přijde z výroby ve stavu, kdy jsou osazeny pouze SMD komponenty.



Obr. 5.4: Verze 0.0 z výroby.

Po dodání DPS z výroby byly zapájeny THT komponenty (pouzdro na baterii, konektor USB micro, tlačítka a vypínač). Na testovací verzi byl místo vypínače osazen konektor, na který je pro zapnutí DPS potřeba nasunout propojku. Po zapojení baterií 18650 do pouzdra a zapnutí vypínače se rozsvítí zelená LED D1, která indikuje přítomnost napájecího napětí.

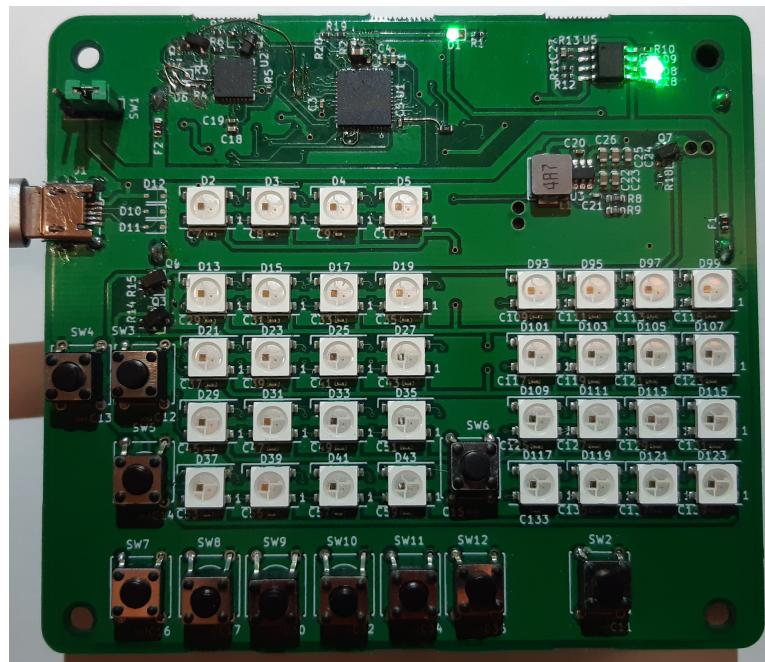
Po vložení baterií do pouzdra a zapnutí DPS bylo pomocí multimetru zjištěno, že ochrana proti přepólování baterií nepropouští napětí dále, i když jsou baterie v pouzdře umístěny správně. Bylo zjištěno, že pouzdro tranzistoru zajišťujícího ochranu proti přepólování neodpovídá navrženému pouzdro tranzistoru. Všechny MOSFET tranzistory byly stejného typu, takže musely být všechny otočeny.

Při zkoušce nabíjení bylo zjištěno, že červená LED indikující nabíjení baterií je přepólovaná. LED se povedla otočit a indikace nabíjení baterií z USB byla v pořádku.

Při programování byl zjištěn problém s komunikací po lince RS-232. Bylo zjištěno, že LED pro indikaci komunikace po lince RS-232 jsou špatně zapojeny, a proto byly odstraněny a nahrazeny zkratem. Jejich zapojení této komunikaci bránilo. Bipolární tranzistory byly taktéž osazeny ve špatném pouzdře a neodpovídaly navrženým pouzdrům.

Při programování a testování všech inteligentních LED WS2812C a tlačítka bylo zjištěno, že jedno tlačítko bylo zapojeno na pin, který nemá softwarově zapínatelný pullup rezistor. Tento rezistor byl dodatečně osazen.

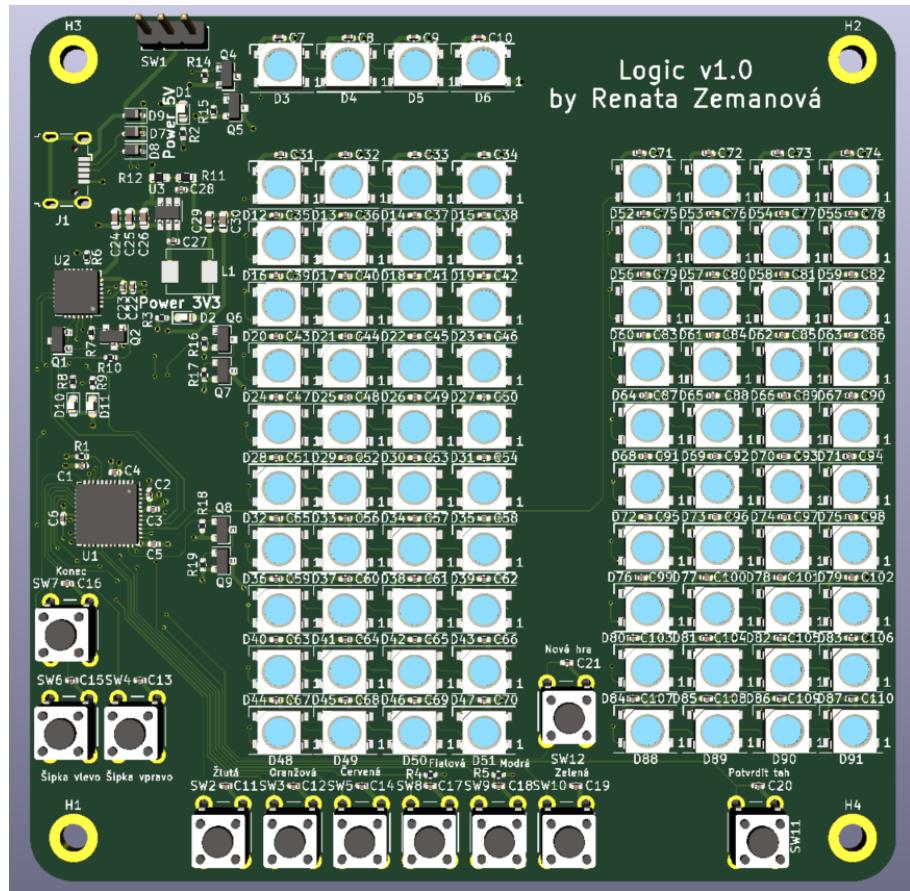
Po veškerých opravách byla DPS připravena na testování softwaru elektronické hry Logic.



Obr. 5.5: Oživená verze 0.0 s opravami.

### 5.3.2 Verze 1.0

DPS přijde z výroby ve stavu, kdy jsou osazeny pouze SMD komponenty.



Obr. 5.6: 3D pohled verze 1.0 DPS.

Poté je nutné ručně osadit THT součástky, tj. vypínač, tlačítka a konektor USB micro. Po připojení DPS přes USB konektor k powerbance, nebo do počítače, se rozsvítí LED D1 a D2, které indikují přítomnost napájecího napětí. LED D2 zároveň značí, že zapojení měniče napětí je funkční.

Tato firma v dnešní době velmi rychle rozšiřuje svůj sortiment k osazení. Od výroby prototypů začala tato firma osazovat nejen SMD součástky, ale také THT.

Všechny chyby z předchozí verze byly odstraněny.

# 6 Software

Software pro elektronickou hru Logic byl napsán v jazyce C++. Bylo využito skutečnosti, že mikrokontrolér ESP32-PICO podporuje Arduino framework. Pro psaní softwaru bylo použito vývojové prostředí Visual Studio Code. [citace](#)

Pro programování inteligentních LED bylo využito již existující knihovny Smart-Leds.h [citace](#) Tato knihovna velmi usnadňuje softwarovou práci s těmito inteligentními LED. Obsahuje například funkci *show*, která umožní zobrazit nastavené barvy na všech LED připojených k jednomu pinu mikrokontroléru ESP32-PICO.

## 6.1 Funkce `init`

Funkce `_init_` slouží pro inicializaci vstupně-výstupních pinů. Všechny piny, na kterých jsou připojena tlačítka, jsou nastaveny jako vstupní se zapnutým softwarovým pull-up rezistorem. Piny, na kterých jsou připojeny inteligentní LED, jsou nastaveny jako výstupní. Piny, kterými je ovládáno připojení napájecího napětí jednotlivých částí inteligentních LED, jsou nastaveny jako výstupní a mají přednastavenou počáteční hodnotu. První část je zapnuta, zbylé 2 jsou vypnuty. Zbylým dvěma částem je napájení zapínáno až ve chvíli, kdy mají dané inteligentní LED svítit.

Výpis 6.1: Funkce pro úvodní inicializaci hardwaru.

```

1 void _init_ (){
2     pinMode(LED_PIN_GAME , OUTPUT);
3     pinMode(LED_PIN_TASK , OUTPUT);
4     pinMode(LED_PIN_EVAL , OUTPUT);
5
6     pinMode(SET_POWER_LEDS_1_TO_4 , OUTPUT);
7     pinMode(SET_POWER_LEDS_5_TO_7 , OUTPUT);
8     pinMode(SET_POWER_LEDS_8_TO_10 , OUTPUT);
9
10    pinMode(SW_ENTER , INPUT_PULLUP);
11    pinMode(SW_RIGHT , INPUT_PULLUP);
12    pinMode(SW_LEFT , INPUT_PULLUP);
13    pinMode(SW_END , INPUT_PULLUP);
14    pinMode(SW_NEW_GAME , INPUT_PULLUP);
15    pinMode(SW_YELLOW , INPUT_PULLUP);
16    pinMode(SW_ORANGE , INPUT_PULLUP);
17    pinMode(SW_RED , INPUT_PULLUP);
18    pinMode(SW_PURPLE , INPUT_PULLUP);
19    pinMode(SW_BLUE , INPUT_PULLUP);
20    pinMode(SW_GREEN , INPUT_PULLUP);
21
22    digitalWrite(SET_POWER_LEDS_1_TO_4 , POWER_ON);
23    digitalWrite(SET_POWER_LEDS_5_TO_7 , POWER_OFF);
24    digitalWrite(SET_POWER_LEDS_8_TO_10 , POWER_OFF);
25
}
```

## 6.2 Funkce ovládání tlačítek

Zákmity tlačítka, které se dějí při stisku tlačítka, jsou řešeny hardwarovým způsobem pomocí filtračních kondenzátorů. Proto tento probém není řešen softwarově. Je ale možné, že hráč tlačítko drží po dobu delší, než trvá jeden programový cyklus. Došlo by tak k detekci více stisků.

Aby k této detekci nedocházelo, čeká se po detekci stisku na puštění tlačítka. Až poté je vykonáván příkaz. Tím je zaručeno, že se např. kurzor posune vždy jen o jedno pole. Pro zjednodušení tohoto zápisu byly vytvořeny funkce *is\_pressed(SW)* a *wait\_for\_btn\_release(SW)*. Argumentem obou funkcí je pin tlačítka, na které se dotazuje/čeká. Funkce *is\_pressed* má návratovou hodnotu typu bool. Typ bool nabývá pouze 2 hodnot. Těmito hodnotami jsou „true“ = 1 (pravda) a „false“ = 0 (nepravda). Pokud je tlačítko zmáčknuto, funkce vrátí „true“ pokud tlačítko zmáčknuto

není, funkce vrátí „false“ Návratová hodnota je poté vyhodnocována v podmínce, zda je konkrétní tlačítko zmáčknuto.

Výpis 6.2: Funkce pro detekci stisku tlačítka.

```

1  bool is_pressed(int btn){
2      if(!digitalRead(btn))
3          return true;
4      return false;
5  }

```

Výpis 6.3: Funkce čekající na puštění tlačítka.

```

1  void wait_for_btn_release(int btn){
2      while(!digitalRead(btn))
3          delay(1);
4  }

```

Tlačítka s barvami plní 2 funkce. Nejprve nastaví barvu dané LED a poté posunou kurzor o jedno pole vpravo. Tlačítka s šípkou posouvají kurzor daným směrem. Pro tyto 2 postupy vznikly funkce *set\_color(led, color)* a *shift\_cursor(led, direct, length)*. Funkce *set\_color* má 2 argumenty. Jedním argumentem je barva LED, kterou má nastavit a druhým je řetězec LED, kde má tuto barvu nastavit.

Výpis 6.4: Ukázka funkce nastavující barvu inteligentním LED.

```

1  void set_color (led_t &LED, Colors COLOR){
2      const int num_max = 32;
3      switch (COLOR){
4          case YELLOW:
5              LED.leds[LED.pos] = Rgb{num_max, num_max, 0};
6              break;
7          case RED:
8              LED.leds[LED.pos] = Rgb{num_max, 0, 0};
9              break;
10         case BLUE:
11             ...
12     }
13 }

```

Funkce *shift\_cursor* posune kurzor vpravo, nebo vlevo, dle předaného argumentu. Pokud se kurzor nachází na konci herního řádku a je předán argument pro posun kurzoru vpravo, je pozice přepočítána tak, aby se kurzor posunul na začátek herního řádku. Pokud se kurzor nachází na začátku herního řádku a je předán argument pro posun kurzoru vlevo, je pozice přepočítána tak, aby se kurzor posunul na konec herního řádku.

Výpis 6.5: Funkce posouvající kurzor.

```

void shift_cursor (led_t &LED, Direct DIR, int length){
    if((DIR == RIGHT) &&
        !((LED.pos + 1 + LINE_LENGTH - length) % LINE_LENGTH))
        LED.pos -= LINE_LENGTH - (LINE_LENGTH - length) - 1;
    else if(DIR == RIGHT)
        LED.pos += 1;
    else if((DIR == LEFT) &&
        (LED.pos == 0 || !(LED.pos % LINE_LENGTH)))
        LED.pos += LINE_LENGTH - (LINE_LENGTH - length) - 1;
    else if(DIR == LEFT)
        LED.pos -= 1;
}

```

## 6.3 Zadání

Vygenerování zadání a její následné uložení do inteligentních LED zajišťují funkce *generate\_task(array, length, diff)* a *set\_task(leds, array, length)*. Funkce generate task vygeneruje pomocí funkce *esp\_random* a matematické úpravy číslo od 0 do počtu barev. Každá barva je interpretována jiným číslem z tohoto rozsahu. Proměnná *diff* určuje, zda je může vygenerovat i černá (= mezera). Její hodnotu určuje stav přepínače. Proměnná *length* určuje, zda se vygeneruje zadání pro hru na 3 nebo 4 herní prvky. Její hodnota je také určována stavem přepínače. Zadání je uloženo do pole prvků, které je poté používáno pro vyhodnocení.

Výpis 6.6: Funkce pro vygenerování zadání.

```

void generate_task (Colors *task, int length, int diff){
    for(int i = 0; i < length; ++i)
        task[i] = Colors(esp_random() % (NUM_OF_COLORS - diff));
}

```

Funkce *set\_task* uloží hodnoty z pole, ve kterém je vygenerováno zadání, do inteligentních LED. Tam je uloženo a po zavolení funkce *show* je ve správnou chvíli zobrazeno.

Výpis 6.7: Funkce pro vygenerování zadání.

```
void set_task(led_t &leds, Colors* array_task, int length){  
    for (int i = 0; i < length; ++i){  
        leds.pos = i;  
        set_color(leds, array_task[i]);  
    }  
}
```

1  
2  
3  
4  
5  
6

funkce evaluate - vyhodnocení

## 7 Způsob ovládání elektronické hry

DPS obsahuje 3 přepínače SW13, SW14 a SW15. Těmito přepínači si hráč volí způsob hry.

Přepínač SW15 slouží k volbě hry pro jednoho, nebo pro dva hráče. Pokud hráč umístí přepínač do pozice ON, hra se přepne do módu pro dva hráče. Rozdíl mezi těmito variantami hry je popsán v následujících kapitolách, funkce dalších přepínačů takéž.

### 7.1 Hra pro jednoho hráče

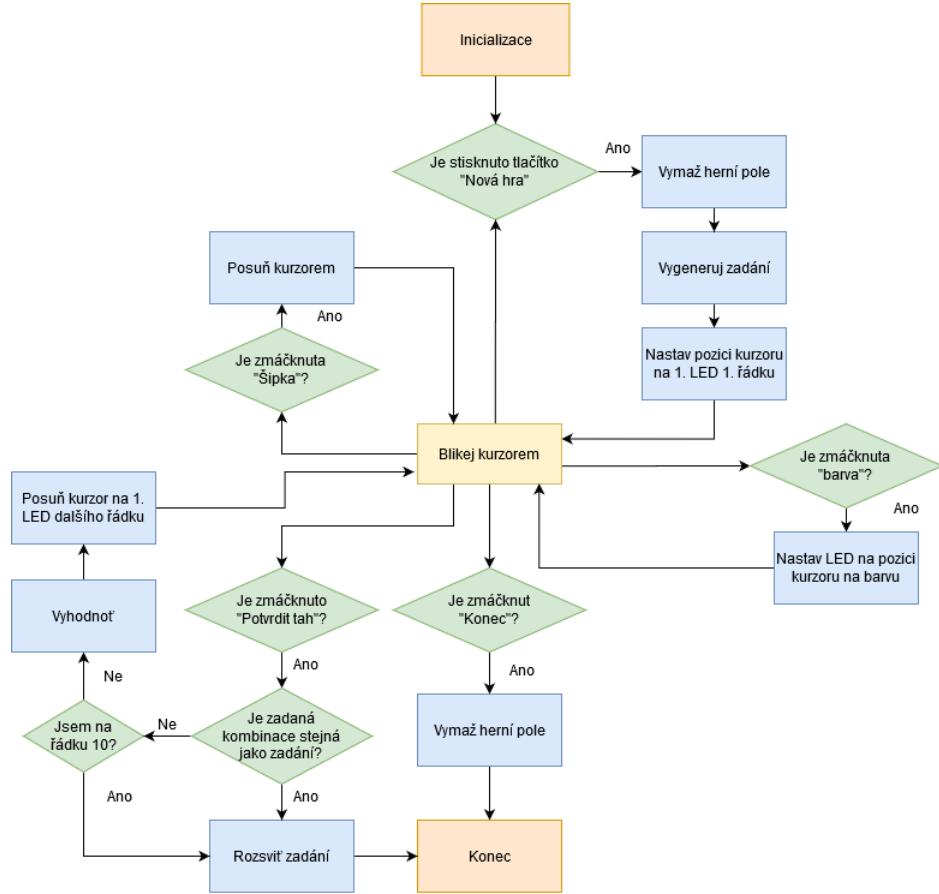
Tato verze hry pro jednoho hráče má stejná pravidla jako desková hra Logic. Funkci druhého hráče nahrazuje řídící elektronika - mikrokontrolér ESP32-PICO.

Po zapnutí DPS stiskneme tlačítko "Nová hra". V této chvíli se vygeneruje zadání, které není vidět, a první herní LED se rozblíká. Rozblíkání herní LED značí pozici kurzoru. Kurzorem lze pohybovat pomocí tlačítek "Šipka vpravo" a "Šipka vlevo". Barvy herních LED se nastavují tlačítky ve spodní části DPS. Tato tlačítka jsou označena danými barvami. Po zadání kombinace stiskneme tlačítko "Potvrdit tah". Po stisku tlačítka proběhne vyhodnocení a to se zobrazí na vyhodnocovacích LED. Červená barva ve vyhodnocení značí, že hráč má v barevné kombinaci stejnou barvu se zadáním na správné pozici. Žlutá barva znamená, že hráč má v barevné kombinaci stejnou barvu se zadáním, ale na chybné pozici. Další barvy nejsou ohodnoceny. Pozice vyhodnocení záměrně není shodná s pozicemi, kterých se hodnocení týká. Zleva jsou nejdříve umístěna všechna červená a až poté všechna žlutá hodnocení.

Po vyhodnocení se kurzor 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". Po stisku tlačítka "Konec" zhasnou všechny herní, vyhodnocovací LED i LED pro zadání. Poté je DPS připravena pro vypnutí vypínačem.

Přepínač SW14 slouží pro nastavení, zda zadání obsahuje 3 nebo 4 barvy. Pokud hráč umístí přepínač do pozice ON, tak tím zvolí variantu hry na 3 barvy. Pohyb kurzoru se omezí na první 3 pole. Poslední pole se ignoruje a také nikdy nesvítí.

Přepínač SW13 slouží pro nastavení, zda zadání může nebo nesmí obsahovat mezery. Pokud dá hráč přepínač do pozice ON, tak v zadání může být mezera. Není to ale nutností.



Obr. 7.1: Vývojový diagram.

## 7.2 Hra pro dva hráče

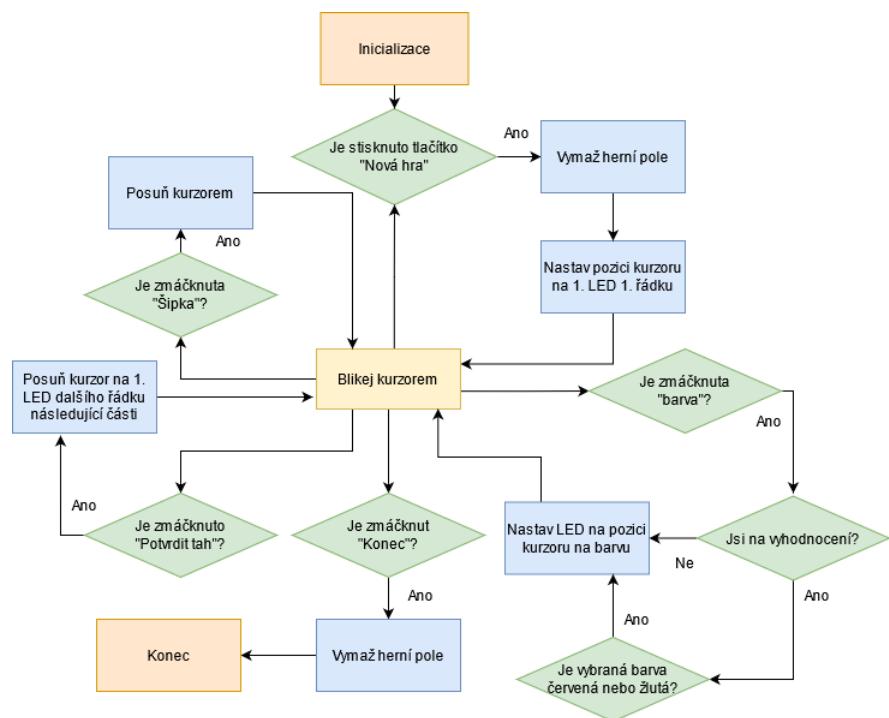
Varianta elektronické hry Logic pro dva hráče má totožná pravidla s deskovou hrou. Po spuštění hry a stisku tlačítka "Nová hra" . bliká kurzor v poli intelligentních LED se zadáním. První hráč zadává barvy pomocí barevných tlačítek a v poli LED se pohybuje pomocí šipek interpretovanými tlačítky. Hráč zadání zakryje a stiskne tlačítko „Potvrdit tah“ .

Po stisku zadání zůstane svítit a kurzor se přesune na první herní pozici. Druhý hráč hádá zadanou kombinaci a tuto kombinaci navolí pomocí barevných tlačítek. Kurzorem může taktéž pohybovat pomocí šipek interpretovanými tlačítky. Po zadání kombinace druhý hráč stiskne tlačítko „Potvrdit tah“ a kurzor se přesune na první pozici vyhodnocovacího pole.

První hráč má v tuto chvíli aktivní pouze šipky a červené a žluté barevné tlačítka. Červenou barvu zadává jako první a indikuje jí, že některá barva druhého hráče je obsažena v zadání, a to na totožné pozici. Poté zadává žlutou barvu, kterou indikuje, že některá z barev prvního hráče je obsažena v zadání, ale na jiné pozici, než kam

ji druhý hráč umístil. Vyhodnocení hráč záměrně neumisťuje na pozice, kterých se hodnocení týká. Zpravidla se zleva umisťují nejdříve všechna červená a až poté všechna žlutá hodnocení.

Po dokončení hodnocení je opět kurzor posunut na první pozici dalšího řádku herního pole. Hra dále pokračuje obdobným způsobem. Pokud druhý hráč nalezne správnou kombinaci, pak první hráč odkryje zadání pro možnou kontrolu a hra je u konce. Pro možnost další hry stiskne jeden z hráčů tlačítka „Nová hra“. Celé herní, vyhodnocovací pole i pole se zadáním je smazáno a kurzor je znova posunut na první pozici pole se zadáním.



Obr. 7.2: Vývojový diagram.

## **8 Krabička**

Krabička byla navržena v programu SolidWorks a vytiskána na 3D tiskárně. Krabička obsahuje 2 vyvýšené sloupky, do kterých jsou zlepeny matičky. Vnitřní a nižší jsou určeny pro přišroubování DPS pomocí montážních otvorů a šroubů M3. Vnější otvory jsou poté pro přišroubování vrchního krytu krabičky. V obou případech jsou použity šrouby M3 se zápustnou hlavou, protože mají podstatně nižší hlavičku, než šrouby klasické.

Krabička také obsahuje tzv. stříšku, která je také součástí deskové hry. Tato stříška slouží pro zakrytí zadání ve hře pro 2 hráče, kdy musí zadání svítit po celou dobu hry, aby druhý hráč byl schopen vyhodnocovat.

## **9 Kompletace**

# Závěr

Byla navržena elektronická hra Logic. Tato hra vychází z původní deskové hry. Byl také vyroben prototyp této hry.

V této práci byla uvedena pravidla deskové hry Logic. Dále byl popsán kompletní návrh elektronické verze této hry. Při výběru elektronických komponentů byl kladen důraz na spotřebu jednotlivých zařízení. Hru řídí mikrokontrolér ESP32-PICO. Herními prvky jsou inteligentní LED WS2812C, které jsou určeny pro přenosná zařízení. Hra Logic je integrována do jedné DPS, která řídí hru a zároveň jsou na ní umístěny herní prvky.

Při návrhu je kladen velký důraz na vizuální podobu s deskovou hrou Logic. Zároveň byla také zachována veškerá doporučená zapojení použitých čipů a modulů a rozmístění součástek na DPS.

Byl vyroben prototyp, na kterém byla otestována funkčnost zapojení. Při oživování prototypu byly odhaleny chyby, jejichž opravy jsou zpracovány do návrhu verze 1.0. Na prototypu byl následně otestován i software.

Dále bude projekt pokračovat v řešení softwaru. Bude vyrobena krabička na DPS a umístěn vypínač. Také budou vyzkoušeny další alternativy způsobu ovládání. DPS bude navržena, aby bylo možné hrát hru jak v konfiguraci pro jednoho hráče, tak v konfiguraci pro dva hráče.

Jsou také zvažovány další možnosti snížení spotřeby DPS, jako je například regulace jasu inteligentních LED v závislosti na okolním osvětlení.

# Literatura

- [1] ADVANCED CIRCUITS: *PCB Trace Width Calculator* Denver, Colorado: [cit. 09. 12. 2020]. Dostupné z URL:  
[<https://www.4pcb.com/trace-width-calculator.html>](https://www.4pcb.com/trace-width-calculator.html).
- [2] Agátin svět: *Dřevěná hra Logik* Praha: [cit. 08. 12. 2020]. Dostupné z URL:  
[<https://www.agatinsvet.cz/drevena-hra-logik/?gclid=Cj0KCQiA5bz-BRD-ARIsABjT4nhc98iKGbskL1JDi\\_EOA2-Yynl\\_DG03UGjG5w-4-1mWRE58qwTKp1oaAj9CEALw\\_wcB>](https://www.agatinsvet.cz/drevena-hra-logik/?gclid=Cj0KCQiA5bz-BRD-ARIsABjT4nhc98iKGbskL1JDi_EOA2-Yynl_DG03UGjG5w-4-1mWRE58qwTKp1oaAj9CEALw_wcB).
- [3] Espressif: *ESP32\_DevKitC\_v4* Poslední aktualizace 06. 12. 2017 [cit. 24. 11. 2020]. Dostupné z URL:  
[<https://dl.espressif.com/dl/schematics/esp32\\_devkitc\\_v4-sch.pdf>](https://dl.espressif.com/dl/schematics/esp32_devkitc_v4-sch.pdf).
- [4] Espressif Systems: *ESP32-PICO-D4* Poslední aktualizace 2019 [cit. 24. 11. 2020]. Dostupné z URL:  
[<https://www.espressif.com/sites/default/files/documentation/esp32-pico-d4\\_datasheet\\_en.pdf>](https://www.espressif.com/sites/default/files/documentation/esp32-pico-d4_datasheet_en.pdf).
- [5] Jean-Pierre Charras, Dick Hollenbeck, Wayne Stambaugh: *KiCad EDA* [cit. 07. 12. 2020]. Dostupné z URL:  
[<https://kicad.org>](https://kicad.org).
- [6] JLCPBCB: *JLCPCB* [cit. 23. 11. 2020]. Dostupné z URL:  
[<https://jlcpbc.com>](https://jlcpbc.com).
- [7] JLCPBCB: *PCB Capabilities* [cit. 23. 11. 2020]. Dostupné z URL:  
[<https://jlcpbc.com/capabilities/Capabilities>](https://jlcpbc.com/capabilities/Capabilities).
- [8] Klub deskových her Paluba: *Logik pravidla* [cit. 30. 11. 2020]. Dostupné z URL:  
[<http://testapp.hrejsi.cz/logik/pravidla.htm>](http://testapp.hrejsi.cz/logik/pravidla.htm).
- [9] Laskarduino: *Nabíječka Li-ion článku TP4056 microUSB* [cit. 04. 12. 2020]. Dostupné z URL:  
[<https://www.laskarduino.cz/nabijecka-li-ion-clanku-tp4056-microusb>](https://www.laskarduino.cz/nabijecka-li-ion-clanku-tp4056-microusb).
- [10] Microchip Technology Inc.: *AN1953, Introduction to USB Type-C* Poslední aktualizace 2015 [cit. 28. 04. 2021]. Dostupné z URL:  
[<https://cdn.sparkfun.com/assets/e/b/4/f/7/USB-C\\_Datasheet.pdf>](https://cdn.sparkfun.com/assets/e/b/4/f/7/USB-C_Datasheet.pdf).
- [11] NKON: *INR18650-29E* [cit. 01. 12. 2020]. Dostupné z URL:  
[<https://eu.nkon.nl/samsung-inr18650-29e.html>](https://eu.nkon.nl/samsung-inr18650-29e.html).

- [12] POSTAV ROBOTA: *Držák na baterie, 2 akumulátorové články 18650 do DPS* [cit. 01. 12. 2020]. Dostupné z URL:  
[<https://www.postavrobota.cz/Drzak-na-baterie-2-akumulatorove-clanky-18650-d.htm>](https://www.postavrobota.cz/Drzak-na-baterie-2-akumulatorove-clanky-18650-d.htm).
- [13] Radioklub OK1KVK: *Elektrotechnické řady hodnot E3, E6, E12, E24* Karlovy Vary [cit. 28. 04. 2021]. Dostupné z URL:  
 [<https://ok1kvk.cz/clanek/2011/elektrotechnicke-rady-hodnot-e3-e6-e12-e24/>](https://ok1kvk.cz/clanek/2011/elektrotechnicke-rady-hodnot-e3-e6-e12-e24/).
- [14] RoboticsBrno, Jan Mrázek: *RB0005-UniversalStepDown* Brno: Poslední aktualizace 30. 11. 2020 [cit. 04. 12. 2020]. Dostupné z URL:  
[<https://github.com/RoboticsBrno/RB0005-UniversalStepDown>](https://github.com/RoboticsBrno/RB0005-UniversalStepDown).
- [15] Satic Chipid: *TP4056 A Standalone Linear Li-lon Battery Charger with Thermal Regulation in SOP-8* [cit. 03. 12. 2020]. Dostupné z URL:  
[<https://static.chipdip.ru/lib/977/DOC002977110.pdf>](https://static.chipdip.ru/lib/977/DOC002977110.pdf).
- [16] SILERGY: *Applocation Note:AN\_SY8105* [cit. 24. 11. 2020]. Dostupné z URL:  
[<https://datasheet.lcsc.com/szlcsc/Silergy-Corp-SY8105ADC\\_C178247.pdf>](https://datasheet.lcsc.com/szlcsc/Silergy-Corp-SY8105ADC_C178247.pdf).
- [17] Silicon Laboratories Inc.: *CP2102N Data Sheet* USA: Poslední aktualizace květen 2016 [cit. 24. 11. 2020]. Dostupné z URL:  
[<https://www.silabs.com/documents/public/data-sheets/cp2102n-datasheet.pdf>](https://www.silabs.com/documents/public/data-sheets/cp2102n-datasheet.pdf).
- [18] Worldsemi: *WS2812C* Poslední aktualizace 06. 12. 2017 [cit. 24. 11. 2020]. Dostupné z URL:  
[<https://datasheet.lcsc.com/szlcsc/1810231210\\_Worldsemi-WS2812C\\_C114587.pdf>](https://datasheet.lcsc.com/szlcsc/1810231210_Worldsemi-WS2812C_C114587.pdf).
- [19] 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/vnitrni-predpisy-a-dokumenty/smernice-rektora-f34920/>](https://www.vutbr.cz/uredni-deska/vnitrni-predpisy-a-dokumenty/smernice-rektora-f34920/).
- [20] ČSN ISO 690 (01 0197) *Informace a dokumentace – Pravidla pro bibliografické odkazy a citace informačních zdrojů*. 40 stran. Praha: Český normalizační institut, 2011.
- [21] ČSN ISO 7144 (010161) *Dokumentace – Formální úprava disertací a podobných dokumentů*. 24 stran. Praha: Český normalizační institut, 1997.

- [22] 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.
- [23] BIERNÁTOVÁ, O., SKÚPA, 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>>
- [24] *Pravidla českého pravopisu*. Zpracoval kolektiv autorů. 1. vydání. Olomouc: FIN PUBLISHING, 1998. 575 s. ISBN 80-86002-40-3.
- [25] 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
- [26] 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.
- [27] RAJMÍC, 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 a zkratek

<b>Bd</b>	Baud - (jednotka modulační rychlosti)
<b>C</b>	Kapacita
<b>DPS</b>	Deska plošného spoje
<b>FLASH</b>	Typ paměti, která je trvalá - nesmaže se ani při ztrátě napájení
<b>GND</b>	Ground - pin, který má nulový potenciál, vůči němu jsou referencované všechny ostatní signály
<b>GPIO</b>	General Purpose Input Output - piny, které mohou být vstupní nebo výstupní
<b>kB</b>	Kilobajt (jednotka velikosti paměti)
<b>LED</b>	Light-Emitting Diode - dioda emitující světlo
<b>Li-Ion</b>	Lithium-iontový akumulátor (druh nabíjecí baterie)
<b>mAh</b>	Miliampérhodina (jednotka kapacity používaná hlavně u baterií)
<b>MB</b>	Megabajt (jednotka velikosti paměti)
<b>MOSFET</b>	Metal Oxide Semiconductor Field Effect Transistor - tranzistor řízený elektrickým polem
<b>RS-232</b>	Druh sériového komunikačního rozhraní
<b>RX</b>	Reciever - přijímač sériového rozhraní
<b>SMD</b>	Surface Mount Device - součástky určené pro povrchovou montáž
<b>SRAM</b>	Static Random Access Memory - rychlá statická paměť, která se smaže při ztrátě napájení
<b>THT</b>	Through-hole technology - vývodová technologie součástek
<b>TX</b>	Transciever - vysílač sériového rozhraní
<b>USB</b>	Universal Serial Bus - univerzální sériová sběrnice, která se používá pro připojení zařízení k počítači
<b>VDD</b>	Označení napájecího napětí

**Šířka levého sloupce Seznamu symbolů a zkratek** je určena šířkou parametru prostředí acronym (viz řádek 1 výpisu zdrojáku na str. 63)

**KolikMista** pouze ukázka vyhrazeného místa

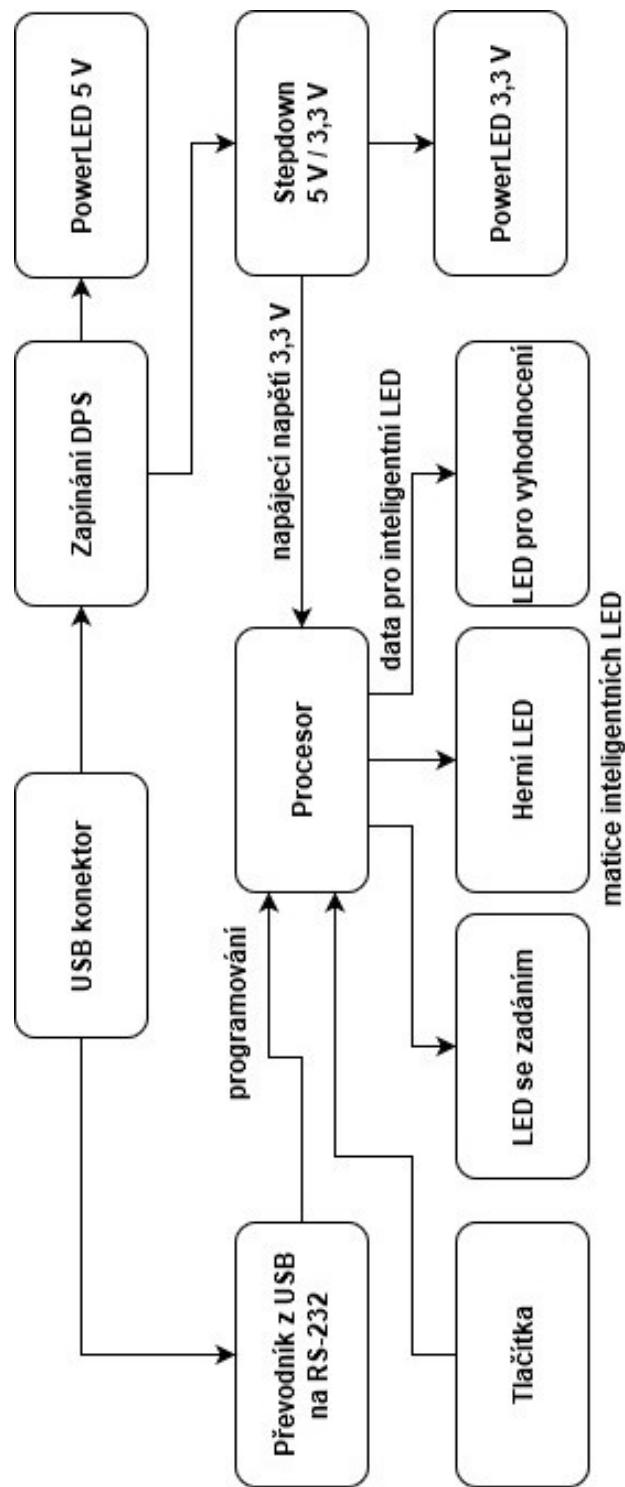
**DSP** číslicové zpracování signálů – Digital Signal Processing

$f_{vz}$  vzorkovací kmitočet

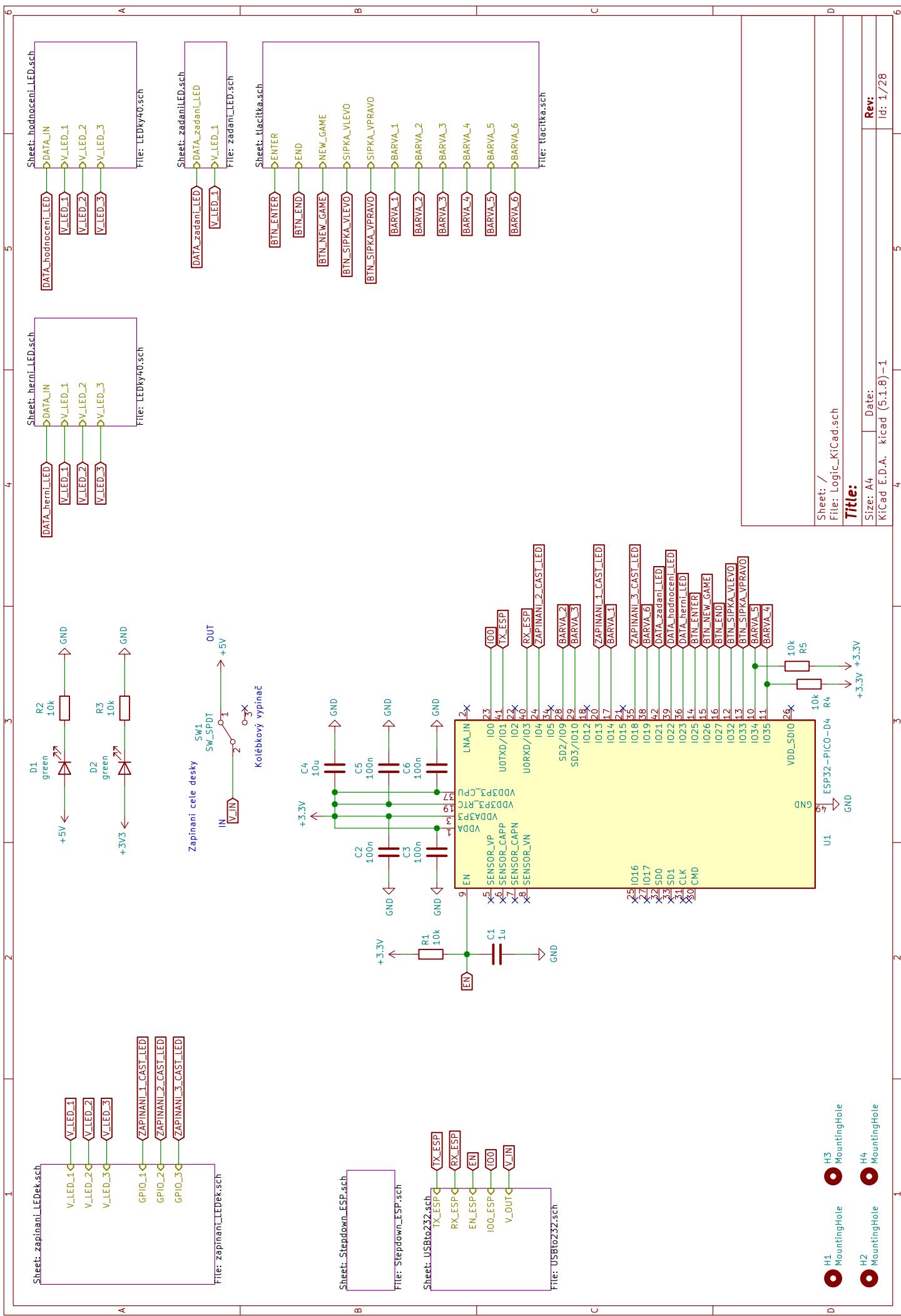
# **Seznam příloh**

<b>A Schéma zapojení DPS verze 1.0</b>	<b>47</b>
<b>B Výrobní podklady DPS verze 1.0</b>	<b>55</b>
<b>C Některé příkazy balíčku <i>thesis</i></b>	<b>61</b>
C.1 Příkazy pro sazbu veličin a jednotek . . . . .	61
C.2 Příkazy pro sazbu symbolů . . . . .	61
<b>D Druhá příloha</b>	<b>62</b>
<b>E Příklad sazby zdrojových kódů</b>	<b>63</b>
E.1 Balíček <i>listings</i> . . . . .	63
<b>F Obsah elektronické přílohy</b>	<b>68</b>

## A Schéma zapojení DPS verze 1.0



Obr. A.1: Blokové schéma DPS verze 1.0.

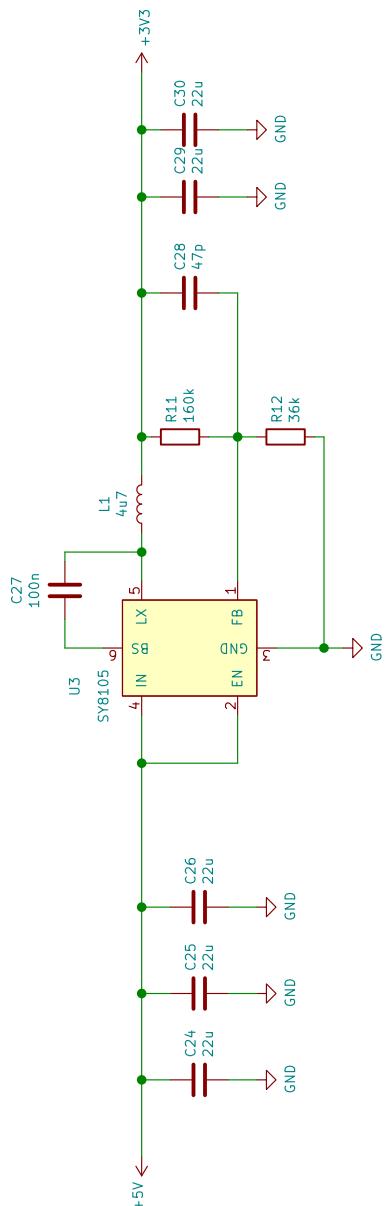


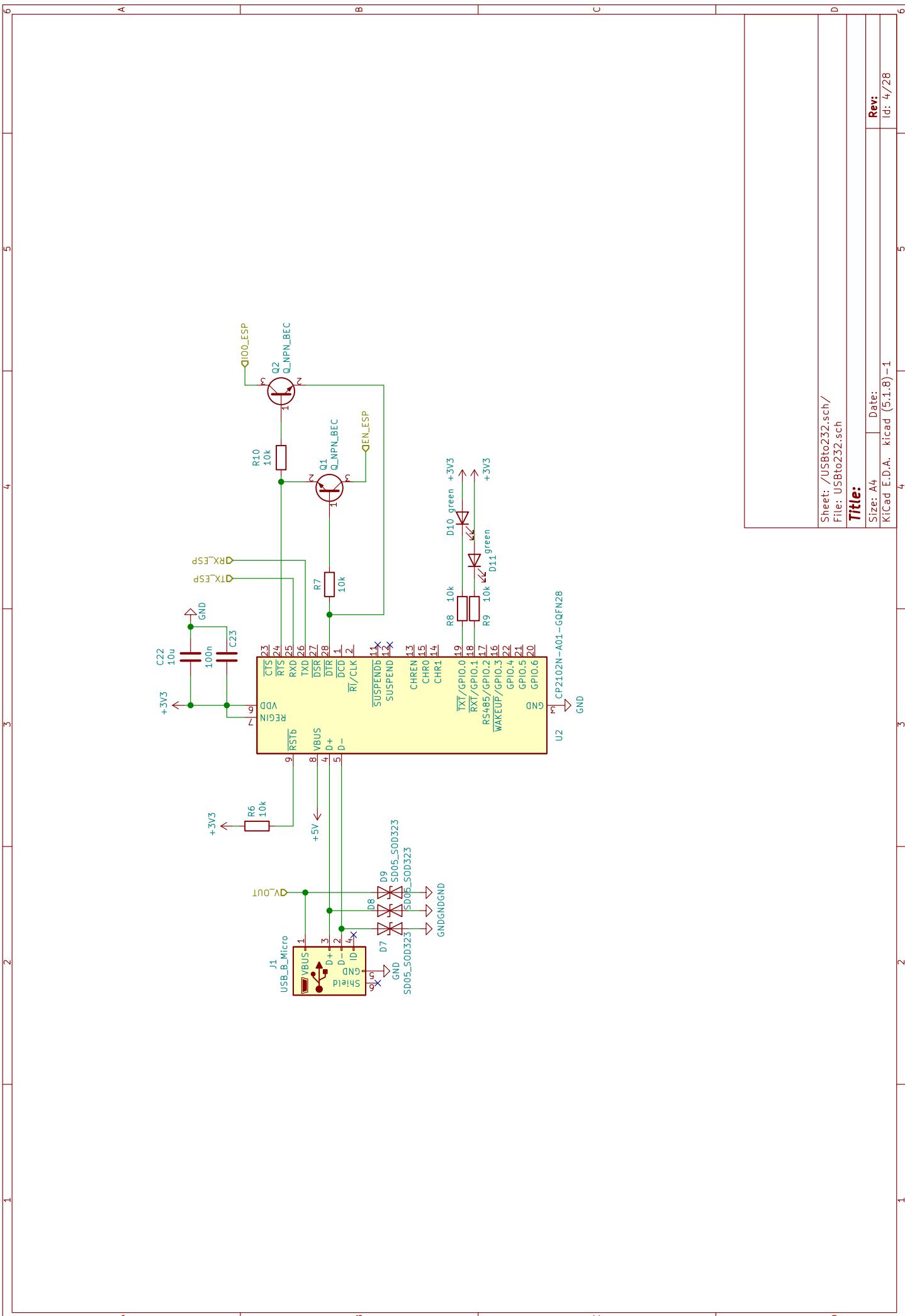
Sheet: /Stepdown\_ESP.sch/  
File: Stepdown\_ESP.sch

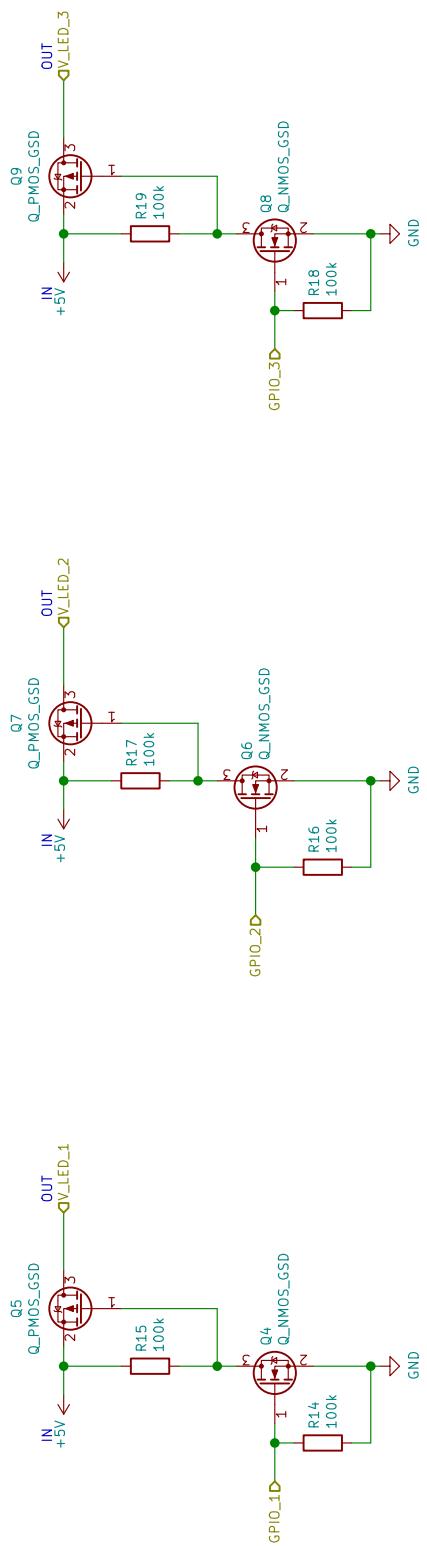
**Title:**

Size: A4  
KiCad E.D.A.      Date:  
Id: 5/28

**Rev:**





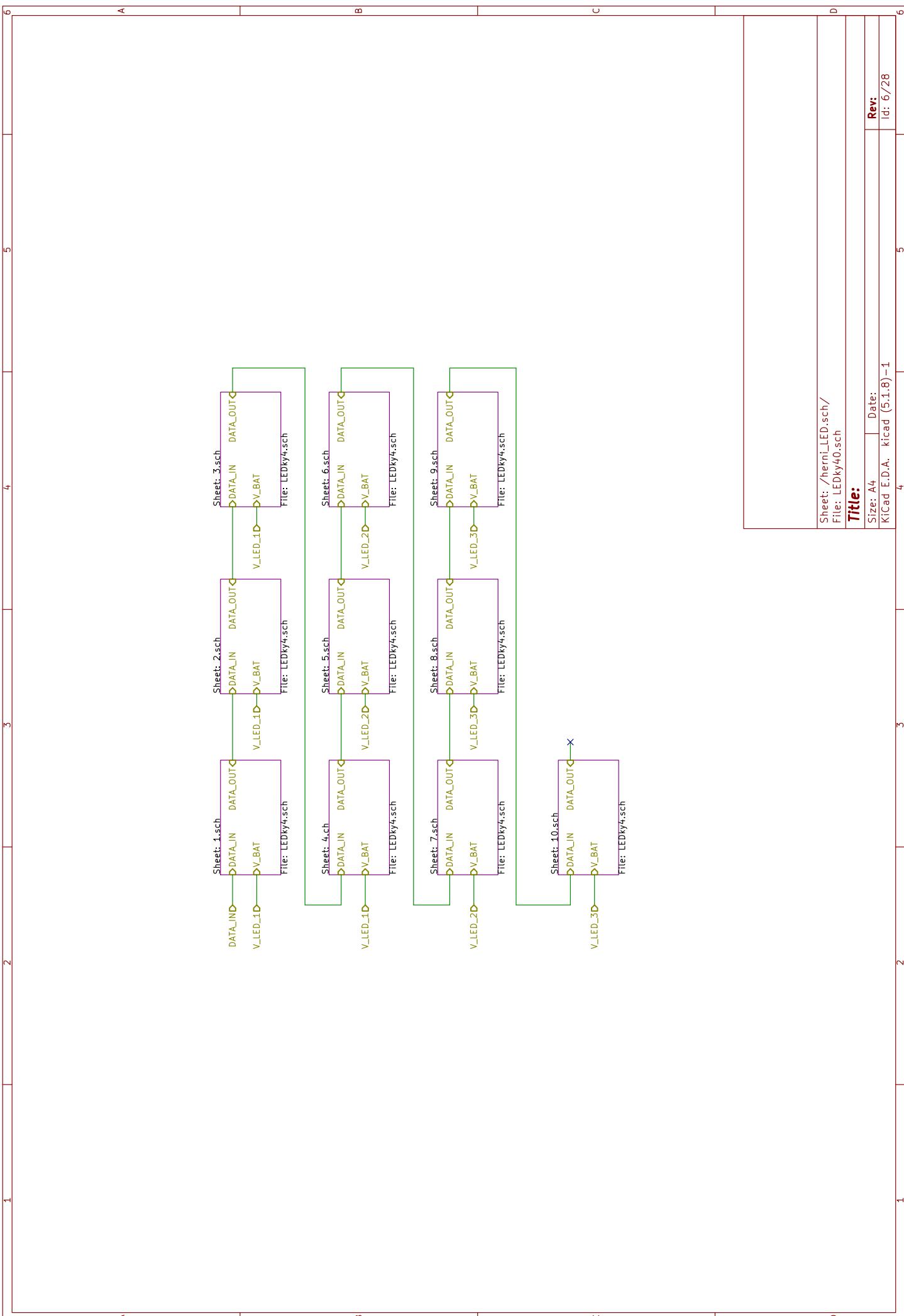


Sheet: /zapinani\_LEDek.sch/  
File: zapinani\_LEDek.sch

**Title:**

Size: A4  
KiCad E.D.A. Date:  
kicad (5.1.8)-1  
Id: 17/28

Rev:	Date:
5	4

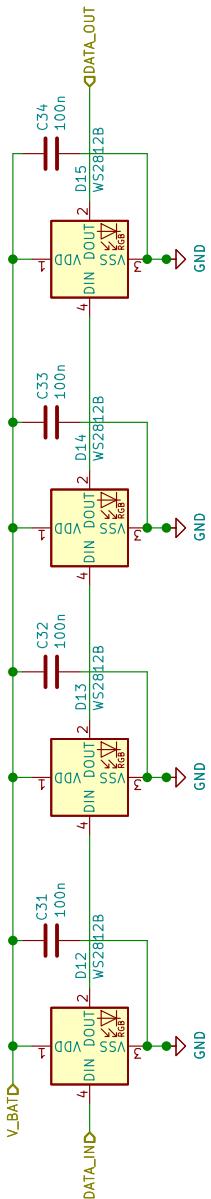


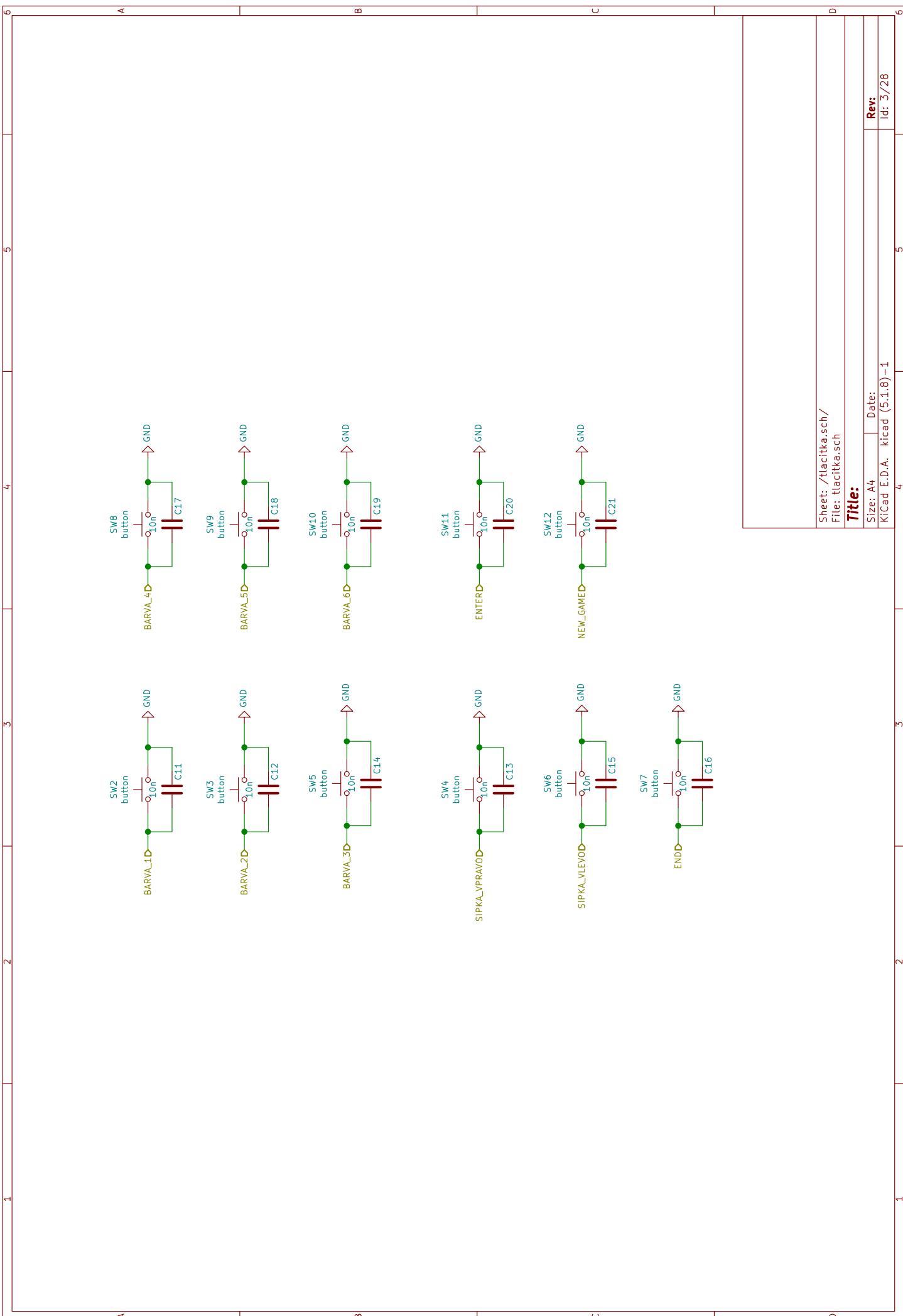
Sheet: /herni\_LED.sch/1.sch/  
File: LEDKy4.sch

**Title:**

Size: A4 Date:  
KiCad E.D.A. kicad (5.1.8)-1  
Id: 7/7/28

Rev: 5 5 5 5 5 5





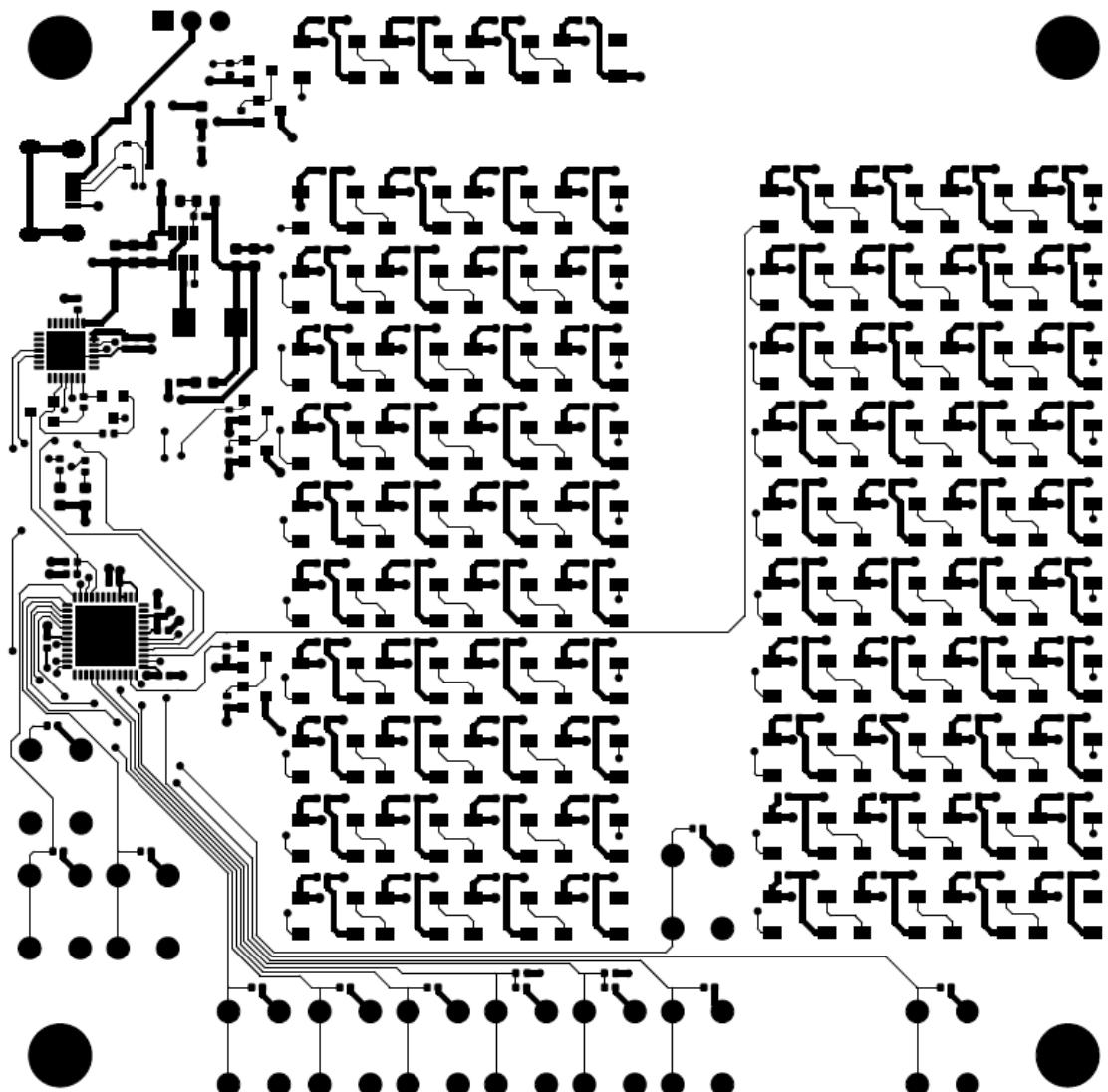
Sheet: /tacitka.sch/  
File: tacitka.sch

**Title:**

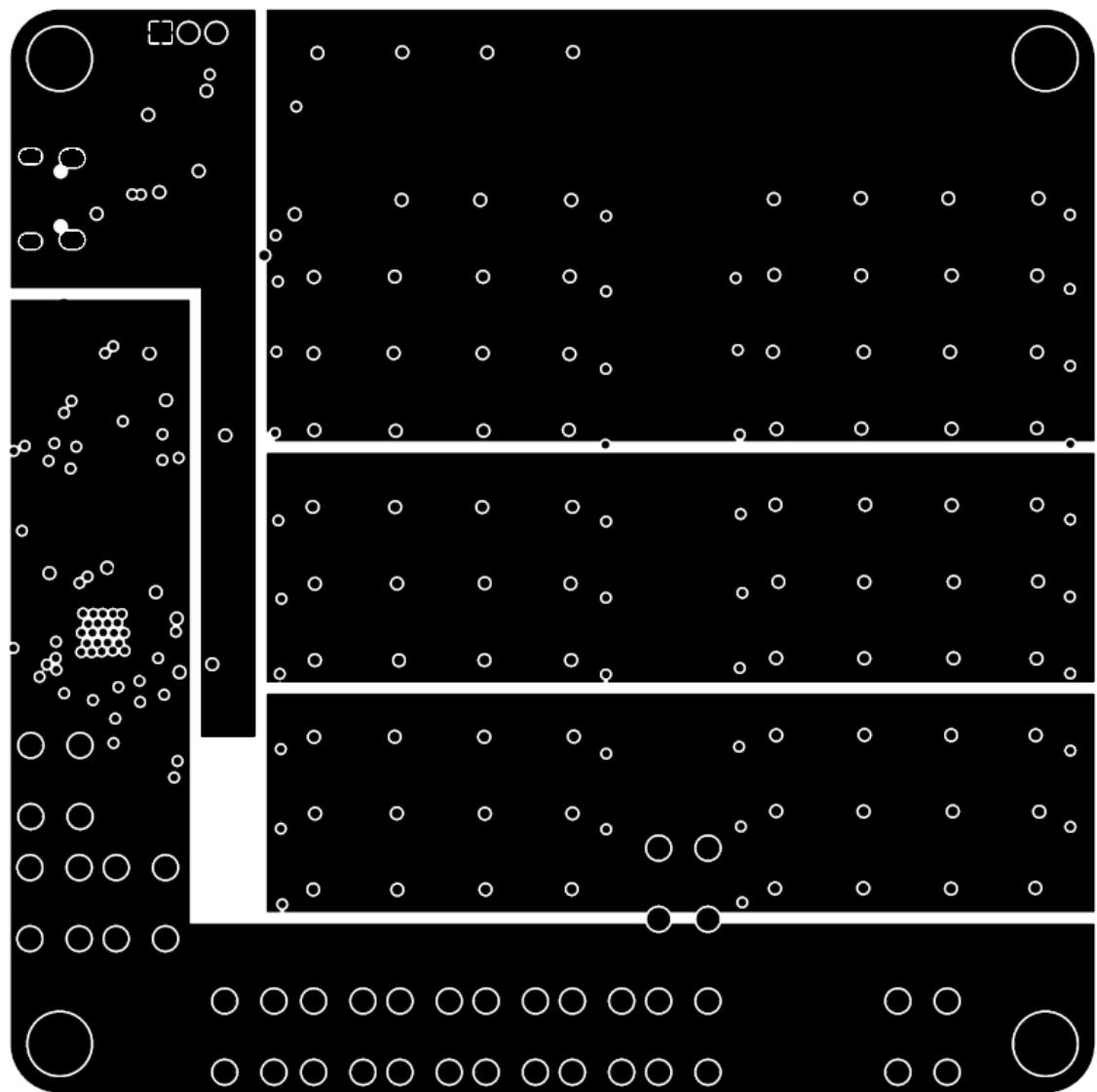
Size: A4  
KiCad E.D.A. Date:  
kicad (5.1.8)-1  
Id: 3/28

**Rew:**

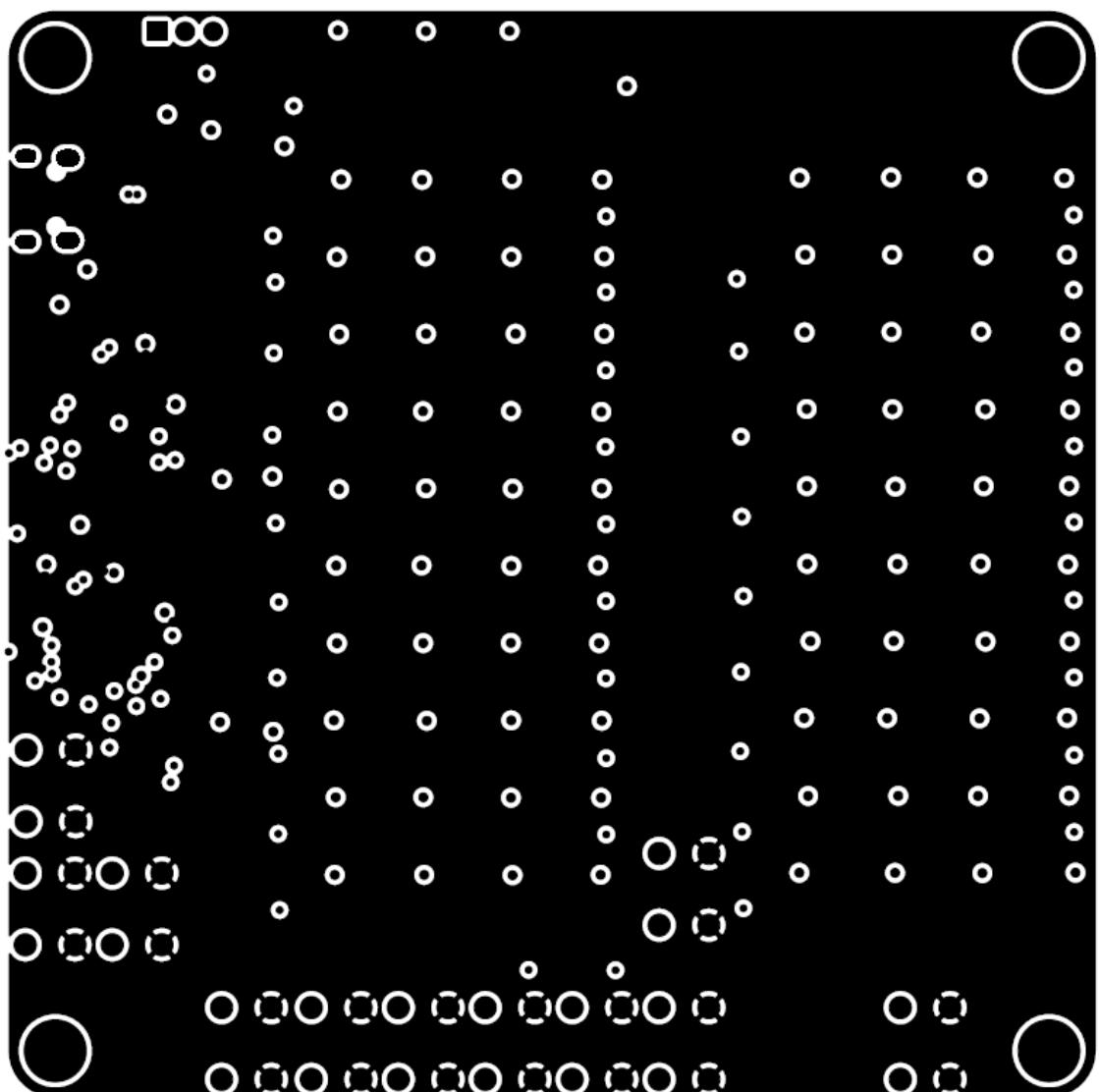
## B Výrobní podklady DPS verze 1.0



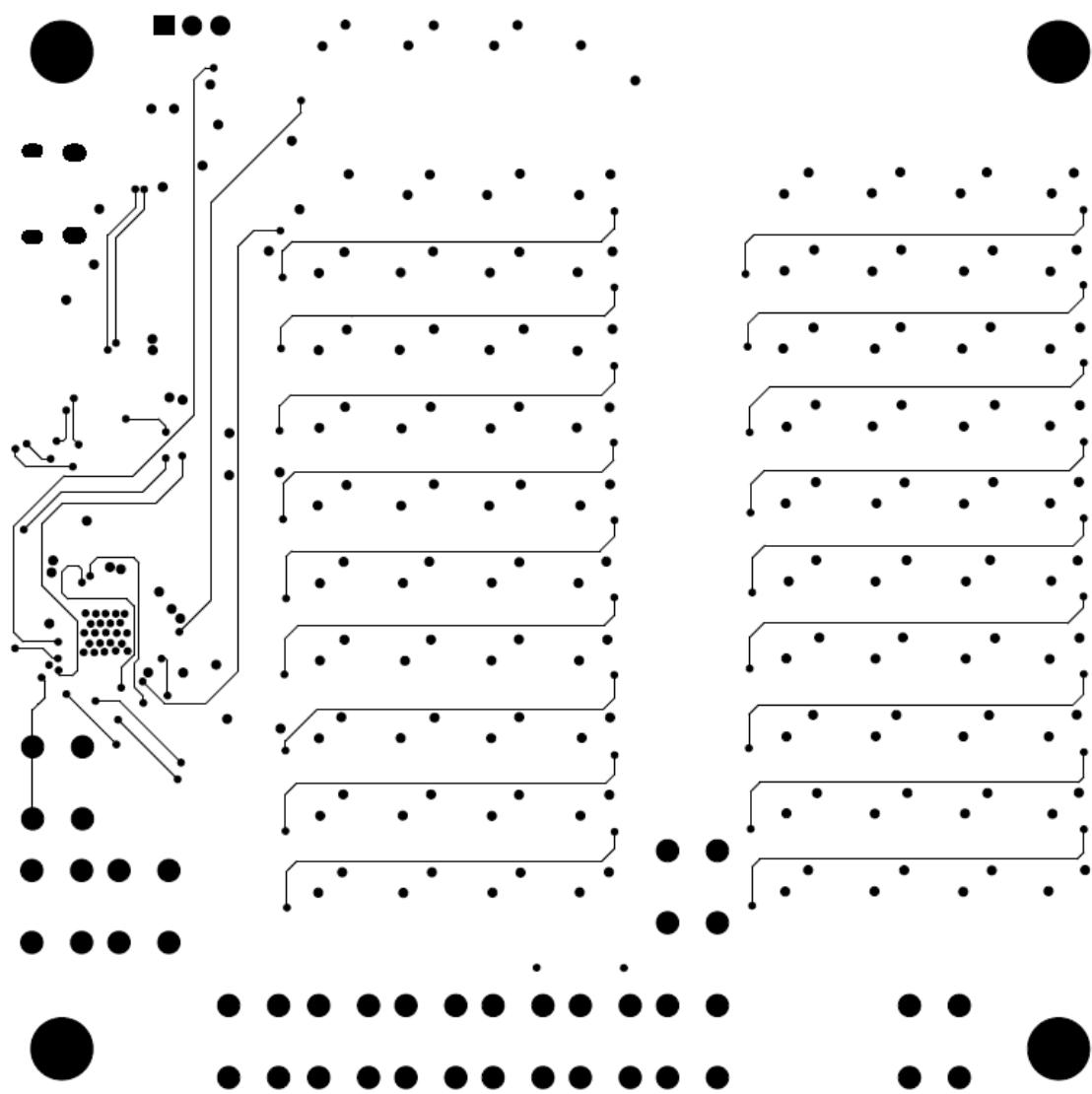
Obr. B.1: Vrstva mědi TOP.



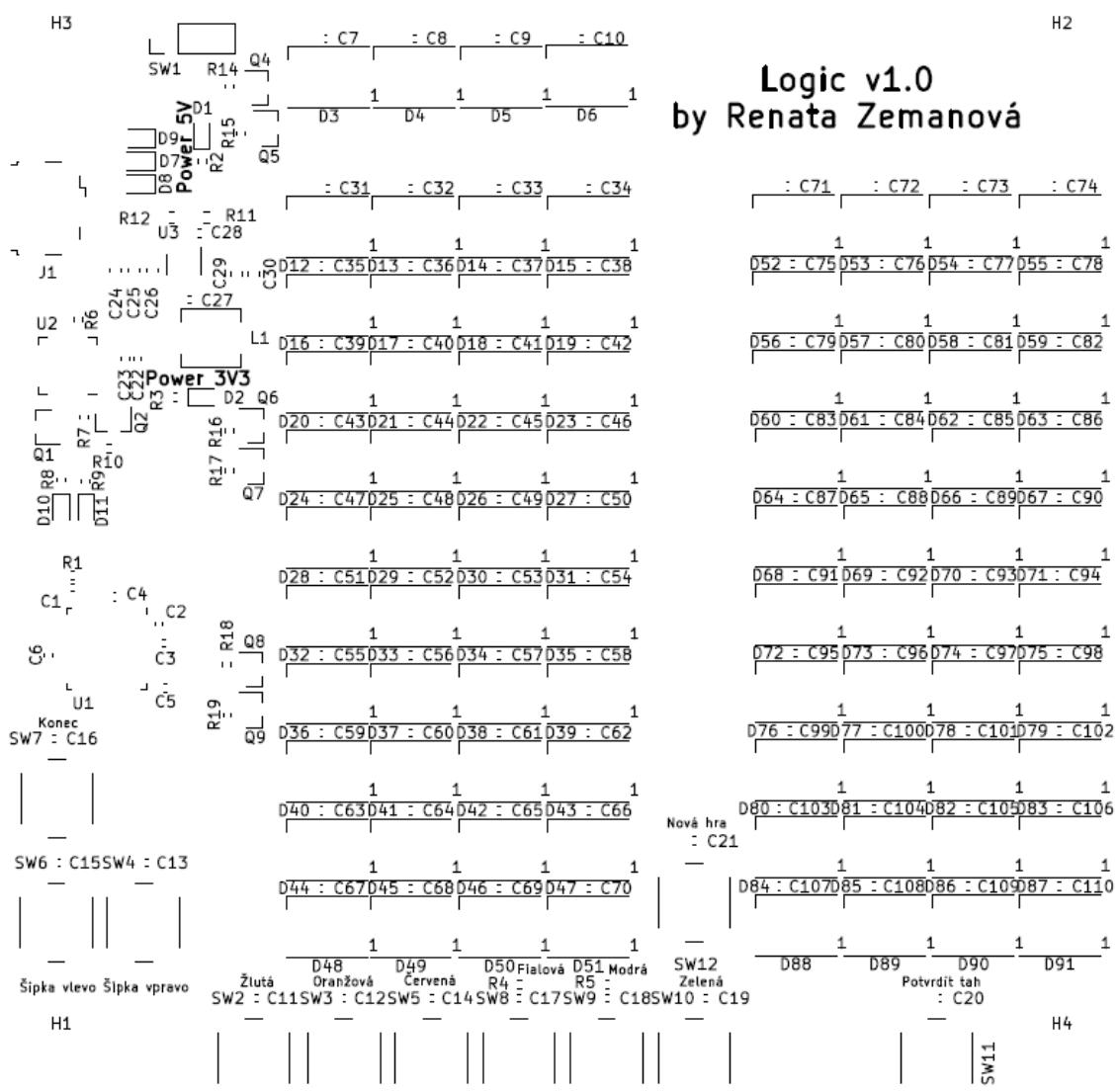
Obr. B.2: Vnitří vrstva mědi napájení.

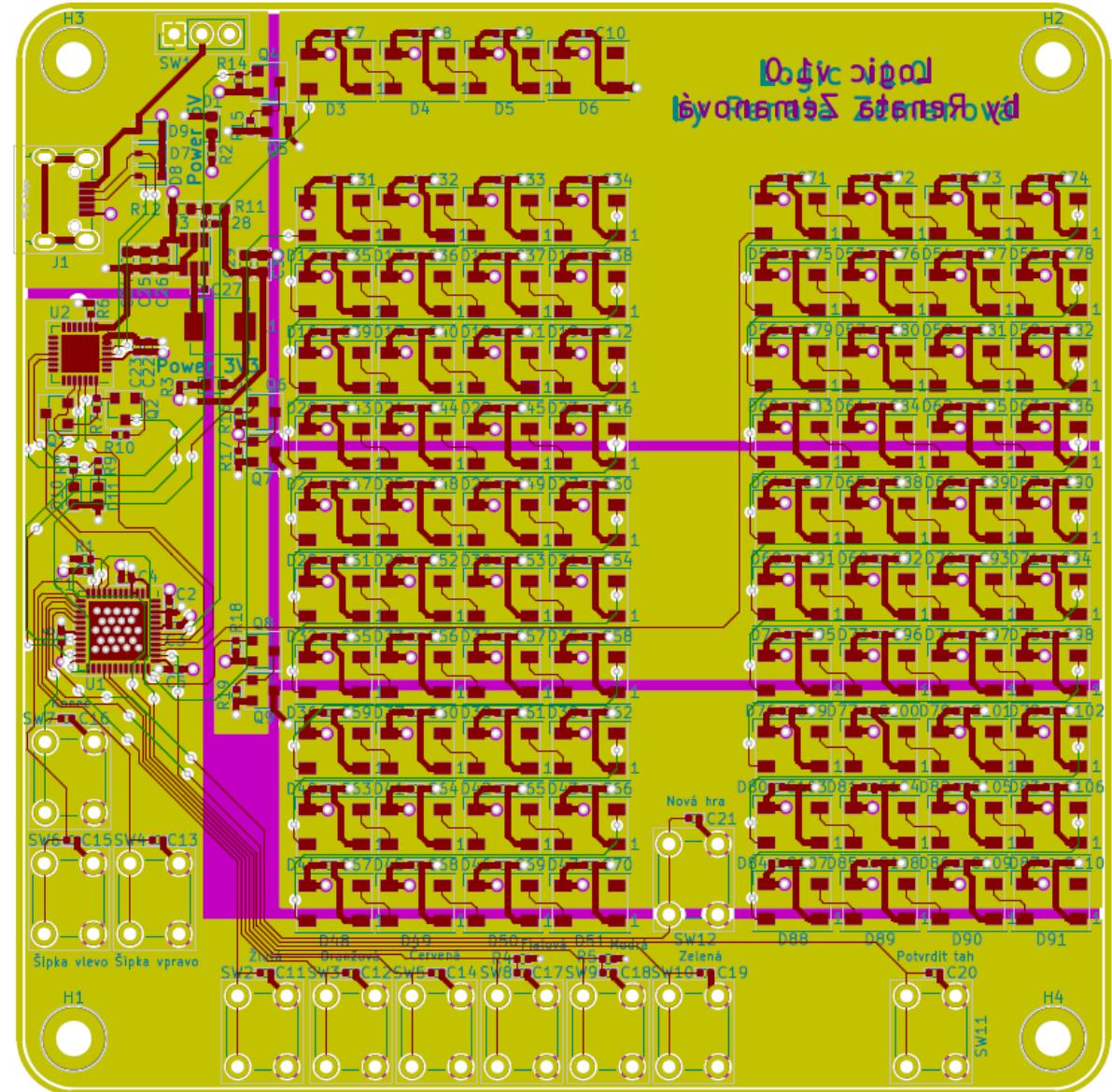


Obr. B.3: Vnitří vrstva mědi GND.



Obr. B.4: Vrstva mědi BOTTOM.





Obr. B.6: Celá DPS.

## C Některé příkazy balíčku thesis

### C.1 Příkazy pro sazbu veličin a jednotek

Tab. C.1: Přehled příkazů pro matematické prostředí

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

### C.2 Příkazy pro sazbu symbolů

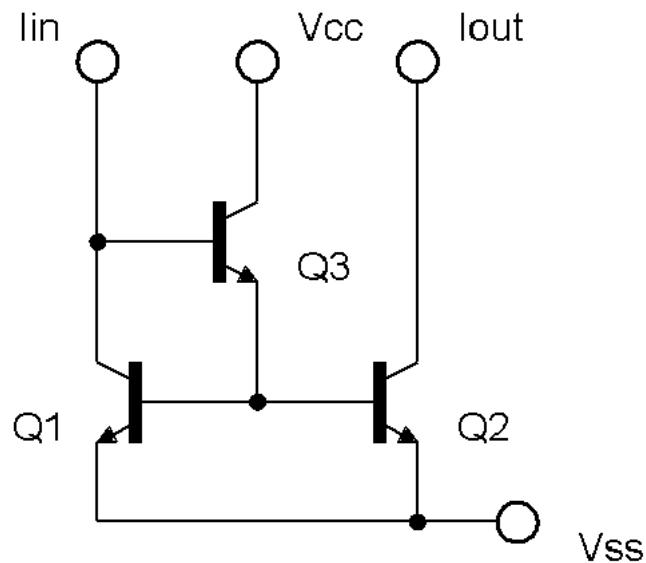
- `\E`, `\eul` – sazba Eulerova čísla: e,
- `\J`, `\jmag`, `\I`, `\imag` – sazba imaginární jednotky: j, i,
- `\dif` – sazba diferenciálu: d,
- `\sinc` – sazba funkce: sinc,
- `\mikro` – sazba symbolu mikro stojatým písmem<sup>1</sup>: μ,
- `\uppi` – sazba symbolu π (stojaté řecké pí, na rozdíl od `\pi`, což sází π).

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

---

<sup>1</sup>znak pochází z balíčku `textcomp`

## D Druhá příloha



Obr. D.1: Zlepšené Wilsonovo proudové zrcadlo.

Pro sazbu vektorových obrázků přímo v L<sup>A</sup>T<sub>E</sub>Xu je možné doporučit balíček TikZ. Příklady sazby je možné najít na T<sub>E</sub>Xample. Pro vyzkoušení je možné použít programy QTikz nebo TikzEdt.

# E Příklad sazby zdrojových kódů

## E.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 lstlisting}
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 nejdélší zkratka nebo symbol. Příklad definice zkratky  $f_{vz}$  je na výpisu E.1.

Výpis E.1: Ukázka sazby zkratky

```
75 \acro{symfvz} % název
76   [\ensuremath{f_{\text{\tiny \texttt{vz}}}}] % symbol
77   {vzorkovací kmitočet} % popis
```

Ukončení seznamu je provedeno ukončením prostředí:

```
26 \% Kilohm (jednotka odporu)
27 \acro{LED}
28   {Light-Emitting Diode - dioda emitující světlo}
29 \acro{Li-Ion}
30   {Lithium-iontový akumulátor (druh nabíjecí baterie)}
31 \% \acro{mA}
32 \% Milliampér (jednotka proudu)
33 \acro{mAh}
34   {Milliampérhodina (jednotka kapacity používaná hlavně u~baterií)}
35 \acro{MB}
36   {Megabajt (jednotka velikosti paměti)}
```

```

37 % \acro{mm}
38 % {Milimetr (jednotka délky)}
39 \acro{MOSFET}
40     {Metal Oxide Semiconductor Field Effect Transistor - tranzistor}
41 \acro{RS-232}
42     {Druh sériového komunikačního rozhraní}
43 \acro{RX}
44     {Reciever - přijímač sériového rozhraní}
45 \acro{SMD}
46     {Surface Mount Device - součástky určené pro povrchovou montáž}
47 \acro{SRAM}
48     {Static Random Acess Mamory - rychlá statické paměť, která se s}
49 \acro{THT}
50     {Through-hole technology - vývodová technologie součástek}
51 \acro{TX}
52     {Transciever - vysílač sériového rozhraní}
53 \acro{USB}
54     {Universal Serial Bus - univerzální sériová sběrnice, která se s}
55 % \acro{V}
56 % {Volt (základní jednotka napětí)}
57 \acro{VDD}
58     {Označení napájecího napětí}
59
60
61 %%%
62 \acro{zkTemp}    % název
63     [Šířka levého sloupce Seznamu symbolů a zkratek]
64     {je určena šířkou parametru prostředí \texttt{acronym} (viz řád}
65             % rozvinutí zkratky
66
67 \acro{zkDummy}
68     [KolikMista]
69     {pouze ukázka vyhrazeného místa}
70
71 \acro{DSP}      % název/zkratka
72     {číslicové zpracování signálů -- Digital Signal Processing}
73             % rozvinutí zkratky
74 %%% bsymfvz
75 \acro{symfvz}      % název

```

```
76      [\ensuremath{f_{\text{vz}}}] % symbol
77      {vzorkovací kmitočet}      % popis
78      %% esymfuz
79
80 \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 způsobena 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 vratte na aktivní příkazem `\shorthandon{-}`. Podobně jako to je ukázáno ve zdrojovém kódu šablony.

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

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

```
1 %% Priklad testovani stability filtrov  
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 korenů 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 mezi 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 mezi stability');  
28 end
```

Výpis E.3: Funkce init v jazyce C++.

```
void _init_ (){
    pinMode(LED_PIN_GAME, OUTPUT);
    pinMode(LED_PIN_TASK, OUTPUT);
    pinMode(LED_PIN_EVAL, OUTPUT);

    pinMode(SET_POWER_LEDS_1_TO_4, OUTPUT);
    pinMode(SET_POWER_LEDS_5_TO_7, OUTPUT);
    pinMode(SET_POWER_LEDS_8_TO_10, OUTPUT);

    pinMode(SW_ENTER, INPUT_PULLUP);
    pinMode(SW_RIGHT, INPUT_PULLUP);
    pinMode(SW_LEFT, INPUT_PULLUP);
    pinMode(SW_END, INPUT_PULLUP);
    pinMode(SW_NEW_GAME, INPUT_PULLUP);
    pinMode(SW_YELLOW, INPUT_PULLUP);
    pinMode(SW_ORANGE, INPUT_PULLUP);
    pinMode(SW_RED, INPUT_PULLUP);
    pinMode(SW_PURPLE, INPUT_PULLUP);
    pinMode(SW_BLUE, INPUT_PULLUP);
    pinMode(SW_GREEN, INPUT_PULLUP);

    digitalWrite(SET_POWER_LEDS_1_TO_4, POWER_ON);
    digitalWrite(SET_POWER_LEDS_5_TO_7, POWER_OFF);
    digitalWrite(SET_POWER_LEDS_8_TO_10, POWER_OFF);
}
```

## F Obsah elektronické přílohy

Elektronická příloha je často nedílnou součástí semestrální nebo závěrečné práce. Vkládá se do informačního systému VUT v Brně ve vhodném formátu (ZIP, PDF ...).

Nezapomeňte uvést, co čtenář v této příloze najde. Je vhodné komentovat obsah každého adresáře, specifikovat, který soubor obsahuje důležitá nastavení, který soubor je určen ke spuštění, uvést nastavení kompilátoru atd. Také je dobré napsat, v jaké verzi software byl kód testován (např. Matlab 2018b). Pokud bylo cílem práce vytvořit hardwarové zařízení, musí elektronická příloha obsahovat veškeré podklady pro výrobu (např. soubory s návrhem DPS v Eagle).

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

```
/..... kořenový adresář přiloženého archivu
  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
    ZlepseWilsonovoZrcadloNPN.png
    ZlepseWilsonovoZrcadloPNP.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
```