



Fakulta elektrotechnická
Katedra aplikované elektroniky a telekomunikací

BAKALÁŘSKÁ PRÁCE

Herní platforma s LCD displejem řízená mikrokontrolérem

Autor práce: Ondřej Lufinka
Vedoucí práce: Ing. Matouš Bartl

Plzeň 2012

ZÁPADOČESKÁ UNIVERZITA V PLZNI
Fakulta elektrotechnická
Akademický rok: 2011/2012

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Ondřej LUFINKA**
Osobní číslo: **E09B0298P**
Studijní program: **B2612 Elektrotechnika a informatika**
Studijní obor: **Elektronika a telekomunikace**
Název tématu: **Herní platforma s LCD displejem řízená mikrokontrolérem**
Zadávací katedra: **Katedra aplikované elektroniky a telekomunikací**

Z á s a d y p r o v y p r a c o v á n í :

Cílem práce je navrhnout hardware pro jednoduchou kapesní hru. Jako zobrazovač je (pro snadnou dostupnost a ovladatelnost) doporučen barevný LCD displej z mobilního telefonu Nokia 6100. Návrh proveďte dle zásad pro zařízení s nízkou spotřebou. Pro demonstraci naprogramujte jednoduchou hru.

1. Prostudujte potřebnou literaturu a navrhnete koncepci zařízení.
2. Prostudujte nabídku dobře dostupných součástek a konkretizujte zapojení zařízení. Při volbě procesorové platformy se zaměřte na nabídku firmy Texas Instruments.
3. Realizujte funkční vzorek.
4. Naprogramujte jednoduchou hru využívající pokročilejší grafiky jako "formule".
5. Změřte parametry zařízení, zejména odběr, v různých režimech.

Rozsah grafických prací: **podle doporučení vedoucího**

Rozsah pracovní zprávy: **20 - 30 stran**

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

Student si vhodnou literaturu vyhledá v dostupných pramenech podle doporučení vedoucího práce.

Vedoucí bakalářské práce:

Ing. Matouš Bartl

Katedra aplikované elektroniky a telekomunikací

Konzultant bakalářské práce:

Ing. Matouš Bartl


Katedra aplikované elektroniky a telekomunikací

Datum zadání bakalářské práce: **17. října 2011**

Termín odevzdání bakalářské práce: **3. června 2012**


Doc. Ing. Jiří Hammerbauer, Ph.D.
děkan




Doc. Dr. Ing. Vjačeslav Georgiev
vedoucí katedry

V Plzni dne 17. října 2011

Abstrakt

Tato bakalářská práce se zabývá návrhem herní platformy postavené na mikroprocesoru MSP430F5529 od firmy Texas Instruments. Popisuje kompletní stavbu hardwaru od základních zapojení pro oživení mikrokontroléru po veškeré vstupní a výstupní periférie vhodné k využití procesoru jako jádra pro aplikaci v oblasti herních konzolí. Obsažena jsou veškerá schémata zapojení a desky plošných spojů. Dále je řešen návrh programového vybavení mikropočítače. Prostřednictvím vývojových diagramů a přiložených kompletních zdrojových kódů je popsán postup inicializace procesoru do provozuschopného stavu a dále vlastní program uživatelského rozhraní včetně jednoduchých funkčních her. Práci lze dále rozvíjet tvorbou složitějšího softwarového vybavení a komplexnějšího využití vstupních a výstupních bloků. Například jejich větší kombinování v rámci jedné aplikace. Na závěr je přiloženo několik měření soustředěných na spotřebu energie procesoru a celého zařízení.

Klíčová slova

herní platforma, herní konzole, mikroprocesor, mikropočítač, MSP430, MSP430F5529, 3D grafika

Abstract

Lufinka, Ondřej. *Microprocessor controlled gaming platform with LCD display* [*Herní platforma s LCD displejem řízená mikrokontrolérem*]. Pilsen, 2012. Bachelor thesis (in Czech). University of West Bohemia. Faculty of Electrical Engineering. Department of Applied Electronics and Telecommunications. Supervisor: Matouš Bartl

This paper deals with the project of gaming platform based on the microprocessor MSP430F5529 by Texas Instruments company. At first it describes the complete construction of hardware. It starts with basic circuits to make the microprocessor operational and it also shows the other peripherals good for device from the field of game consoles. All electrical schematics and printed circuit boards are attached. Then the software design is discussed via flow charts and complete source codes. There is the solution of the initialization of the microcomputer and also the user interface including simple games. The thesis can be still developed by the creation of new firmware with more sophisticated applications and by more complex usage of the input output peripherals. For example by combining more of them in one application. At the end there are several simulations focused on the power consumption of the microprocessor itself and the whole device.

Keywords

gaming platform, game console, microprocessor, microcomputer, MSP430, MSP430F5529, 3D graphics

Prohlášení

Předkládám tímto k posouzení a obhajobě bakalářskou práci, zpracovanou na závěr studia na Fakultě elektrotechnické Západočeské univerzity v Plzni.

Prohlašuji, že jsem svou závěrečnou práci vypracoval samostatně pod vedením vedoucího bakalářské 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 bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 270 trestního zákona č. 40/2009 Sb.

Také prohlašuji, že veškerý software, použitý při řešení této bakalářské práce, je legální.

V Plzni dne 6. června 2012

Ondřej Lufinka

.....

Podpis

Obsah

Seznam obrázků	vi
Seznam tabulek	vii
Seznam symbolů a zkratek	viii
1 Úvod	1
2 Návrh hardwaru pro herní platformu	2
2.1 Mikroprocesor MSP430F5529	2
2.2 Napájecí obvody	3
2.2.1 Spínané zdroje TPS61220	3
2.2.2 Obvody pro zapínání a vypínání konzole	5
2.2.3 Filtr pro oddělení analogového napájení	6
2.3 Programovací rozhraní	7
2.3.1 Programovací konektor JTAG	7
2.3.2 Konektor USB	7
2.4 Vstupní periférie	8
2.4.1 Akcelerometr MMA7260	8
2.5 Výstupní periférie	9
2.5.1 Displej	9
2.5.2 D/A převodník TLV5638	10
2.5.3 Zesilovač LM386 s připojeným reproduktorem	11
2.5.4 Vibrační motorek	12
3 Software herní platformy	13
3.1 Popis souboru main.c	13
3.2 Inicializace procesoru	16
3.2.1 Hodinový systém	16
3.2.2 Rozhraní sériových komunikací	16
3.2.3 Vstupní a výstupní brány	16
3.2.4 Čítače	17
3.3 Hra Ping Pong	17

3.4 3D grafika	21
4 Měření spotřeby herní konzole	26
5 Závěr	28
Reference, použitá literatura	29
Přílohy	30
A Desky plošných spojů	30
A.1 Horní a spodní deska plošných spojů včetně potisku	30
B Fotografie hotového výrobku	37

Seznam obrázků

2.1	Princip činnosti step-up měniče.	3
2.2	Spínaný zdroj TPS61220 pro 3,3 V.	5
2.3	Obvody pro zapínání a vypínání konzole	6
2.4	Filtr pro oddělení analogového napájení.	6
2.5	Programovací konektor JTAG.	7
2.6	Konektor USB.	8
2.7	Akcelerometr MMA7260.	8
2.8	Displej.	9
2.9	Signál PWM.	10
2.10	D/A převodník TLV5638.	10
2.11	Zesilovač LM386 s připojeným reproduktorem.	11
2.12	Vibrační motorek.	12
3.1	Vývojový diagram funkce main.	14
3.2	Obrazovka se základním menu a kurzorem.	15
3.3	Ukázka činnosti 4-bitového čítače se směrem čítání nahoru.	17
3.4	Obrazovka hry Ping Pong.	18
3.5	Vývojový diagram funkce hry Ping Pong v přerušení čítače	18
3.6	Vývojový diagram hlavní funkce hry Ping Pong	19
3.7	Podstata metody ray-casting.	21
3.8	Efekt rybího oka.	22
3.9	Vývojový diagram hlavní funkce pro vykreslování 3D grafiky	22
3.10	Obrazovka 3D grafiky.	23
A.1	Horní strana dolní desky plošných spojů herní platformy.	31
A.2	Horní potisk dolní desky plošných spojů herní platformy.	32
A.3	Dolní strana dolní desky plošných spojů herní platformy (zrcadleně).	33
A.4	Dolní potisk dolní desky plošných spojů herní platformy (zrcadleně).	34
A.5	Horní strana horní desky plošných spojů herní platformy. Dolní není zob- razena. Obsahuje pouze propojení ovládacích prvků s konektory.	35
A.6	Horní potisk horní desky plošných spojů herní platformy.	36
B.1	Hotové a osazené desky plošných spojů.	37

Seznam tabulek

4.1	Naměřené hodnoty spotřeby při režimu nečinnosti.	26
4.2	Naměřené hodnoty spotřeby zařízení při plném provozu.	27

Seznam symbolů a zkratek

RISC	Reduced Instruction Set Computer. Procesor s redukovanou instrukční sadou.
LPM	Low Power Mode. Režim nízké spotřeby.
SRAM	Static Random Access Memory. Statická paměť s náhodným přístupem.
JTAG	Joint Test Action Group. Standardizovaná architektura pro programování FLASH pamětí aj.
USB	Universal Serial Bus. Univerzální sériová sběrnice.
LCD	Liquid Crystal Display. Displej s tekutými krystaly.
SPI	Serial Peripheral Interface. Sériové periferní rozhraní standardizované pro komunikaci mezi logickými obvody.
PWM	Pulse Width Modulation. Pulzně šířková modulace.
LED	Light Emitting Diode. Dioda emitující světlo.
DAC	Digital to Analog Converter. Digitálně analogový převodník.
WDT	Watch Dog Timer.
PLL	Phase Lock Loop. Fázový závěs.
MSB	Most Significant Bit. Nejvýznamnější bit.
LSB	Least Significant Bit. Nejméně významný bit.
PSP	Play Station Portable.

1

Úvod

Tématem bakalářské práce je herní platforma s LCD displejem řízená mikrokontrolérem. Zabývá se kompletním návrhem jak po stránce hardwaru, tak i programovým vybavením použitým ke správné funkčnosti procesoru pro kontrolování běhu zařízení. Nakonec je přiloženo několik měření spotřeby mikroprocesoru i celé konzole.

Cíl projektu spočívá v sestavení funkčního vzorku umožňujícího uživateli ovládnout několika jednoduchých her prostřednictvím vstupních prvků, kterými jsou například klasická tlačítka nebo případně další možnosti jako joystick nebo akcelerometr. Hlavním výstupem je zobrazování na barevném grafickém LCD displeji, konkrétně je v případě této herní konzole doporučen displej z mobilního telefonu NOKIA 6100 díky snadné dostupnosti na trhu a existenci dobrých návodů s postupy pro oživení a funkcemi k vykreslování základních objektů. Dále je možno pro výstupy využít vibrační motorek nebo zabudovaný reproduktor. Základním stavebním kamenem zařízení je mikrokontrolér, jehož výběr je zaměřený na sortiment firmy Texas Instruments. Jejich procesory se vyznačují nízkou spotřebou, což vyhovuje záměru vytvoření přenosné kapesní herní platformy, a mají dostupné poměrně slušné katalogové listy.

Z hlediska členění je bakalářská práce strukturována do tří kapitol. První pojednává o jednotlivých hardwarových blocích, které dohromady tvoří funkční koncept herní platformy. Začíná stručnou charakteristikou procesoru, který je hlavní řídicí a výpočetní jednotkou zařízení, a postupně přechází přes napájecí zdroje a programovací rozhraní na další bloky, tvořící vstupní a výstupní periférie. Každá součást je vysvětlena na výtahu z kompletního elektrického schématu, které je případně možné najít v přílohách bakalářské práce. Ty dále obsahují i všechny použité desky plošných spojů. Následující kapitola popisuje prostřednictvím vývojových diagramů a ukázek zdrojových kódů software použitý pro funkčnost procesoru. Probírají se hlavní řídicí část programu, inicializace, neboli postup jak uvést mikroprocesor do provozuschopného stavu, a jednotlivé hry běžící na této herní platformě. Poslední kapitola obsahuje tabulky měření zabývajících se spotřebou elektrické energie.

2

Návrh hardwaru pro herní platformu

Kapitola popisuje jednotlivé bloky hardwaru herní konzole postupně od použitého mikroprocesoru, přes spínané zdroje, programovací rozhraní až po jednotlivé vstupní a výstupní periférie. Konzole je rozdělena do dvou oboustranných desek plošných spojů oddělených konektory a distančními sloupky. Na spodní jsou veškeré součástky zajišťující funkčnost zařízení, na horní se nachází ovládací prvky a displej. Na obrázcích v této kapitole jsou znázorněné jednotlivé části schématu a kompletní návrh je poté k dispozici v samostatné příloze na konci práce. Desky plošných spojů jsou k nahlédnutí na obrázcích v příloze A.1.

2.1 Mikroprocesor MSP430F5529

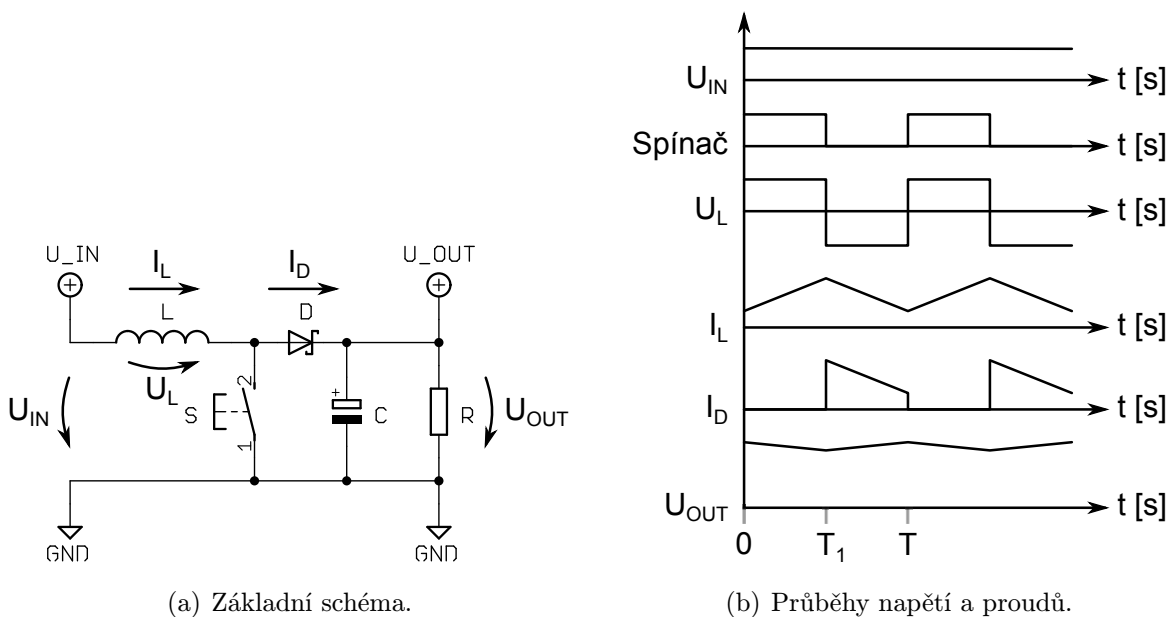
Celá herní konzole je založená na jednočipovém mikropočítači MSP430F5529 od firmy Texas Instruments. Jedná se o 16-bitový mikrokontrolér s architekturou RISC (Reduced Instruction Set Computer) vyznačující se nízkou spotřebou, jako ostatně většina produktů této firmy. Podporuje několik režimů pro další snižování spotřeby při nečinnosti, tzv. LPM (Low Power Mode). Disponuje pamětí typu FLASH o velikosti 128 kB a pamětí SRAM (Static Random Access Memory) s velikostí 10 kB. Obsahuje vstupy pro dva různé krystaly, přičemž nejčastěji se jeden využije pro taktování procesoru (hodnoty řádově MHz) a druhý pro měření reálného času (32,768 kHz). Vnitřní rozvod hodin lze dále upravovat pomocí vestavěného fázového závěsu. V popisované herní konzoli je využit krystal o frekvenci 4 MHz, která je nadále násobená na hodnotu 16 MHz. Ta slouží jako základní hodinový kmitočet procesoru. Dále máme k dispozici 4 16-bitové čítače, dvě sběrnice pro sériovou komunikaci, 12-bitový A/D převodník, podporu programování přes JTAG (Joint Test Action Group) nebo přímo přes USB (Universal Serial Bus), 63 vstupně/výstupních vývodů, atd. Úplný popis procesoru a jeho periférií je k nalezení v katalogovém listu uvedeném v seznamu literatury pod označením [1].

2.2 Napájecí obvody

Pro napájení zařízení jsou použity dva spínané zdroje TPS61220. Vstupní napětí 3V (nebo 2,4 V v případě použití nabíjecích článků) je generované dvěma bateriemi typu AAA. První zdroj vytváří napětí 3,3 V pro mikroprocesor a veškeré další číslicové a logické obvody a druhý zdroj dodává napětí 5V, které je použito k napájení zesilovače reproduktoru, vibračního motorku a joysticku. Od digitálního rozvodu 3,3 V je poté ještě filtrem odděleno napájení analogové pro D/A převodník a analogové části samotného mikroprocesoru. Dále jsou popsány jednotlivé části napájecích obvodů.

2.2.1 Spínané zdroje TPS61220

Spínané zdroje jsou obecně elektrická zapojení pro zvyšování nebo snižování hodnoty stejnosměrného napětí. V případě této herní konzole jsou použity pouze zvyšovací měniče, neboli step-up konvertory. Na schématu s grafem (obr. 2.1) je v principu vysvětlena jejich funkčnost.



Obr. 2.1: Princip činnosti step-up měniče.

Perioda fungování měniče je rozdělena vždy do dvou částí. Od nuly do doby T_1 je sepnutý spínač S. Na indukčnosti L je vstupní napětí a roste jí protékající proud, čímž se v ní akumuluje energie. Přes diodu D žádný proud neteče, protože je v tomto případě v závěrném směru. Výstupní napětí pozvolna klesá z důvodu vybíjení kondenzátoru C do zátěže R. Od doby T_1 do T dojde k rozepnutí spínače S a začne klesat proud indukčností. Tím se obrátí polarita jejího napětí a v sérii se sečte s napětím zdroje. Proud diodou D je stejný jako v indukčnosti L a přenáší energii z cívky a zdroje do kondenzátoru C, což způsobí pozvolný růst výstupního napětí. Napětí na výstupu je vždy vyšší než na vstupu právě díky sečtení obou napětí zdroje a indukčnosti. Pokud bychom vzali

v úvahu použití ideálního kondenzátoru C, nebude docházet ke kolísání výstupního napětí a lze ho vyjádřit v závislosti na vstupním pomocí následujících výpočtů. Kolísání je však i v reálném případě velmi malé. Uvažujeme také ideální diodu D, na které nedochází k úbytku napětí.

Rozdíl nárůstu proudu indukčností po dobu sepnutého spínače lze vyjádřit ze vstupního napětí, které je i napětím na indukčnosti, podle vztahu z rov. 2.1 (po úpravě z rov. 2.2).

$$\Delta I_{L \text{ sepnuto}} = \frac{1}{L} \int_0^{T_1} U_{IN} dt \quad (2.1)$$

$$\Delta I_{L \text{ sepnuto}} = \frac{T_1 U_{IN}}{L} \quad (2.2)$$

Po dobu rozepnutého spínače platí ve smyčce tvořené vstupním napětím, napětím na indukčnosti L a výstupním napětím vztah z rov. 2.3 a rozdíl poklesu proudu indukčností v této době lze poté vyjádřit vztahem z rov. 2.4 (po úpravě z rov. 2.5).

$$U_{OUT} - U_{IN} = -L \frac{di_{L \text{ rozepnuto}}}{dt} \quad (2.3)$$

$$\Delta I_{L \text{ rozepnuto}} = -\frac{1}{L} \int_{T_1}^T (U_{OUT} - U_{IN}) dt \quad (2.4)$$

$$\Delta I_{L \text{ rozepnuto}} = -\frac{(T - T_1)(U_{OUT} - U_{IN})}{L} \quad (2.5)$$

Na indukčnosti platí zákon zachování energie, která se vypočítá podle rov. 2.6, a proto se součet změn proudu na indukčnosti L při sepnutém a při rozepnutém spínači S musí rovnat nule (rov. 2.7).

$$W = \frac{1}{2} LI^2 \quad (2.6)$$

$$\frac{T_1 U_{IN}}{L} - \frac{(T - T_1)(U_{OUT} - U_{IN})}{L} = 0 \quad (2.7)$$

Nakonec se již jen vyjádří velikost výstupního napětí v závislosti na vstupním a na poměru dob sepnutí a rozepnutí spínače (rov. 2.8).

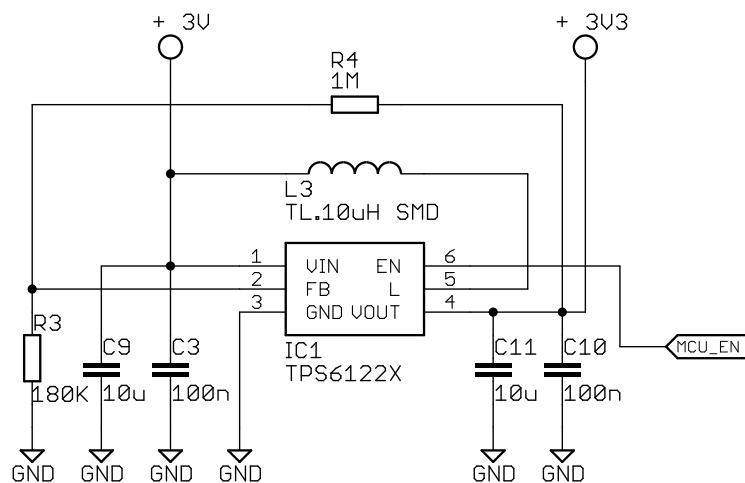
$$U_{OUT} = U_{IN} \frac{T}{T - T_1} \quad (2.8)$$

Z rovnice plyne, že například při nastavení doby T_1 na polovinu celkové periody se výstupní napětí rovná dvojnásobné hodnotě napětí vstupního.

Obvody konkrétních spínaných zdrojů použitých v této práci jsou již integrované a podpůrné součástky jsou doplněny podle katalogového listu uvedeného v seznamu literatury pod označením [2]. Kondenzátory C3 a C9 tvoří skupinu vstupních blokovacích kondenzátorů, C10 a C11 jsou výstupní blokovací kondenzátory, indukčnost L3 s feromagnetickým jádrem zastává úlohu akumulárního prvku, který je nezbytnou součástí každého spínaného zdroje, a rezistory R3 a R4 slouží k volbě výstupního napětí, které je dáno podle rov. 2.9. Integrovaný obvod je nastaven jako zvyšovací spínaný zdroj, jak již bylo řečeno, a vstupní napětí musí tedy být vždy nižší než výstupní.

$$R_4 = R_3 \left(\frac{V_{OUT}}{V_{FB}} - 1 \right) \quad [\Omega] \quad (2.9)$$

Vstup EN umožňuje mikropočítači řídit zapnutí a vypnutí zdroje. Spínaný zdroj je funkční, pokud je na vstupu EN logická "1". Na obr. 2.2 je příslušné schéma spínaného zdroje pro 3,3 V. Zdroj pro 5 V je zcela identický až na rezistory R3 a R4, které mají poté hodnoty 180k a 1M6.

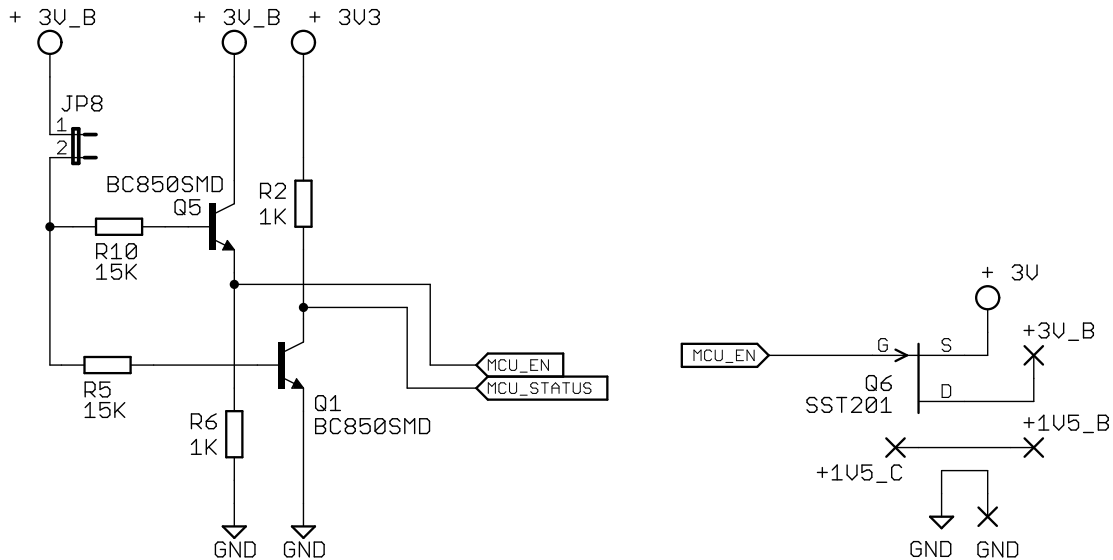


Obr. 2.2: Spínaný zdroj TPS61220 pro 3,3 V.

2.2.2 Obvody pro zapínání a vypínání konzole

Pro zapínání a vypínání platformy slouží obvod z obr. 2.3. V pravé části obrázku je mezi vývody GND a +1V5_B umístěna první baterie a mezi vývody +1V5_C a +3V_B druhá baterie. Na konektor JP8 je připojené tlačítko, které při stisknutí v době vypnutého zařízení vyvolá přes tranzistor Q5 logickou "1" ve větvi MCU_EN. Tato větev je vyvedena na vstup G tranzistoru Q6, čímž se k obvodu připojí AAA baterie. Dále je připojena ke vstupu EN spínaného zdroje pro 3,3 V a tento zdroj je tedy stiskem tlačítka aktivován. Tím dojde k inicializaci mikroprocesoru, který má přes svůj výstupní vývod také přístup k větvi MCU_EN a může se tedy programově postarat o napětí v této větvi, aby uživatel nemusel držet tlačítko stisknuté po celou dobu chodu zařízení.

Při opětovném stisku tlačítka tentokrát v době činnosti zařízení dojde k uzemnění větve MCU_STATUS, přes tranzistor Q1. Ta je připojená na vstup mikrokontroléru, který zruší dodávku napětí do větve MCU_EN a po uvolnění tlačítka uživatelem dojde k vypnutí konzole.



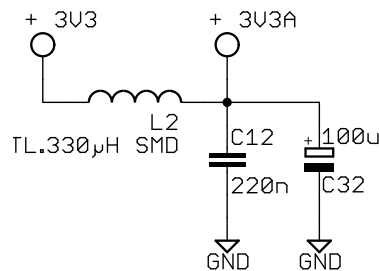
(a) Obvod pro zapínání a vypínání pomocí jednoho tlačítka.

(b) Zdroj herní konzole - dvě baterie typu AAA.

Obr. 2.3: Obvody pro zapínání a vypínání konzole

2.2.3 Filtr pro oddělení analogového napájení

Filtr oddělující analogovou část napájení od digitální (obr. 2.4) se používá z důvodu šumu na napájecím rozvodu číslicových obvodů. Šum vzniká impulsním odběrem proudu z tohoto rozvodu a bez filtru by negativně ovlivňoval správnou funkčnost obvodů analogových, které jsou díky požadované přesnosti hodnot napětí, s nimiž pracují, na jakékoli rušení velmi náchylné. Filtr se skládá z typické kombinace indukčnosti L2 a kondenzátorů C12 a C32 tvořící dolní propust.



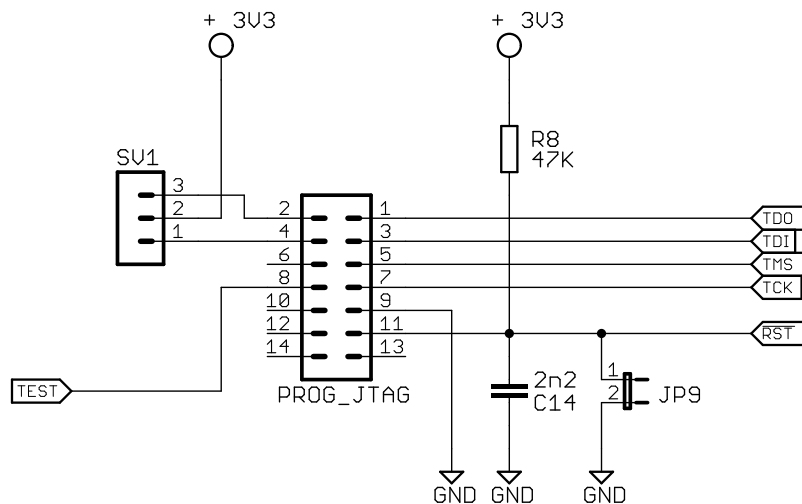
Obr. 2.4: Filtr pro oddělení analogového napájení.

2.3 Programovací rozhraní

Mikroprocesor MSP430F5529 podporuje klasické programování přes konektor typu JTAG, ale také programování přímo přes USB.

2.3.1 Programovací konektor JTAG

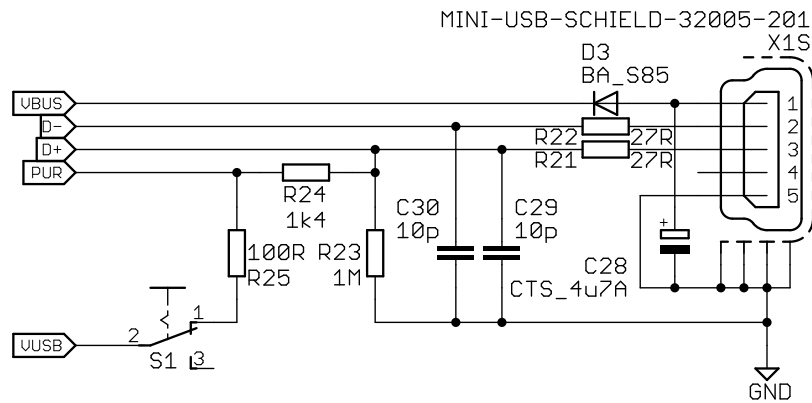
Programovací konektor JTAG je typické rozhraní pro programování mikroprocesorů. Součástky a propojení vývodů na obr. 2.5 s mikropočítačem jsou navrženy podle strany 119 katalogového listu uvedeného v seznamu literatury pod označením [3]. Komunikace probíhá sériově po 4 vodičích (Test Data Out, Test Data In, Test Mode Select a Test Clock). Přes rezistor R8 je na vstup RST přiváděna logická "1", která je nutná pro běh procesoru. Na konektoru JP9 je tlačítko pro reset procesoru a konektor SV1 slouží k přepínání režimu programování s interním nebo externím napájením.



Obr. 2.5: Programovací konektor JTAG.

2.3.2 Konektor USB

Připojení konektoru USB je dáno konstrukcí mikrokontroléru a je také kompletně vytvořeno podle strany 119 katalogového listu uvedeného v seznamu literatury pod označením [3]. Schéma je znázorněno na obr. 2.6. Vodičem VBUS je k mikroprocesoru přiváděné napájení přes ochrannou diodu, D- a D+ slouží k sériové komunikaci a vstup PUR označuje procesoru aktivitu na sběrnici. Přepínačem S1 lze uvést mikroprocesor do stavu k programování (sepnuto) nebo do stavu pouze pro komunikaci přes toto rozhraní (rozepnuto). Stav pro komunikaci není dále u této herní platformy využit.



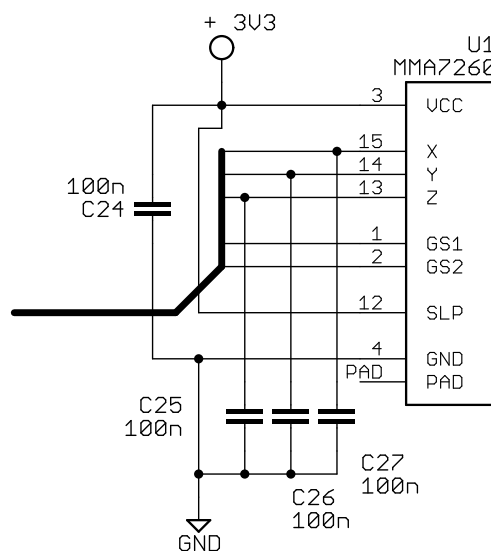
Obr. 2.6: Konektor USB.

2.4 Vstupní periférie

Obvody pro uživatelské vstupy tvoří skupina tlačítek, z nichž 4 jsou umístěny shora v klasickém rozložení pro ovládání 4 směrů a dvě zezadu. Po jednom na každé straně. Dále je na zařízení umístěn joystick a akcelerometr. Rozložení ovládacích prvků umístěných shora je patrné na obr. A.6, umístění zadních tlačítek a akcelerometru lze najít na obr. A.2. Podrobněji je dále popsán pouze akcelerometr. Zapojení tlačítek a joysticku je zřejmé z kompletního schématu v samostatné příloze.

2.4.1 Akcelerometr MMA7260

Použitý akcelerometr reaguje ve třech osách na dynamickou akceleraci a také na naklonění vůči gravitačnímu poli země. Schéma je na obr. 2.7.



Obr. 2.7: Akcelerometr MMA7260.

Poloha akcelerometru je indikována napětím na výstupech X, Y a Z, které jsou přivedené na analogové vstupy mikroprocesoru. Napětí 1,65 V udává nulové naklonění a

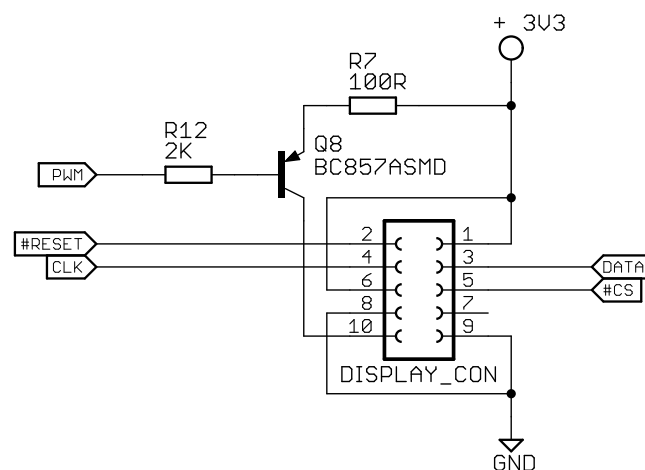
zrychlení pohybu v dané ose. Vstupy GS1 a GS2 slouží k nastavení citlivosti. Možné hodnoty v závislosti na logických úrovních jsou k nalezení v katalogovém listu pod označením [4] v seznamu literatury. Vstup SLP, který umožňuje uvedení obvodu do režimu spánku, je zde natrvalo připojen k napájení a zmíněný režim tedy není v tomto případě podporován. Kondenzátor C24 zabraňuje kolísání napájení a zbylé kondenzátory vyhlazují průběhy napětí na výstupech jednotlivých os, aby nedocházelo ke zbytečnému šumu.

2.5 Výstupní periférie

Mezi výstupní periférie na herní platformě jsou zařazeny displej, D/A převodník, zesilovač s reproduktorem a vibrační motorek. Displej je umístěn na horní desce plošných spojů (obr. A.6), zbytek bloků na spodní (obr. A.2). Společně zajišťují odezvu k uživateli. Součástí výstupu je dále osm světlo emitujících diod připojených na celý jeden port procesoru (k nahlédnutí na kompletním schématu v samostatné příloze). Slouží k testování funkčnosti jednotlivých bloků kódu při vývoji a dále nejsou popisovány.

2.5.1 Displej

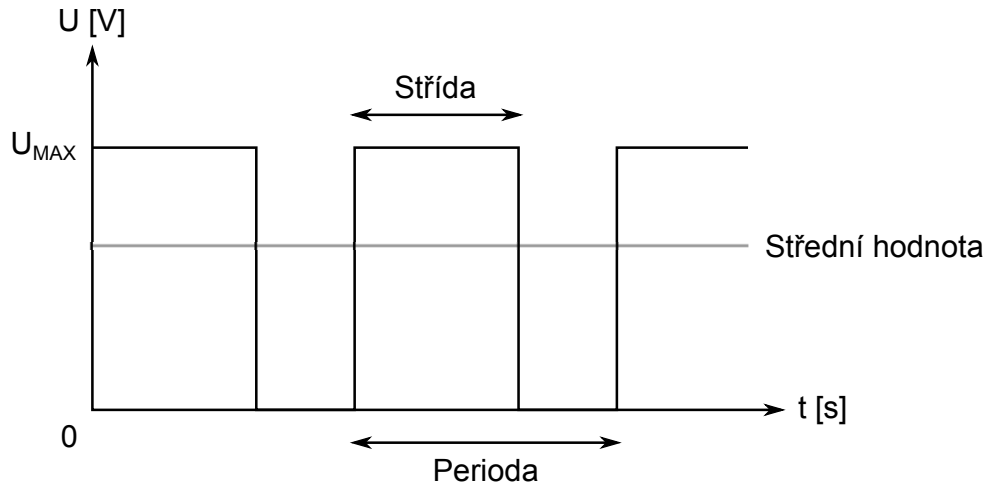
K herní konzoli je připojený barevný LCD (Liquid Crystal Display) s rozlišením 132 x 132 bodů a podporuje 8, 12, nebo 16-bitové kódování barev. Schéma jeho propojení s procesorem najdeme na obr. 2.8.



Obr. 2.8: Displej.

Přístupuje se k němu přes sériovou komunikaci s rozhraním SPI (Serial Peripheral Interface), která využívá obecně tři vodiče. U tohoto konkrétního displeje je ale umožněn pouze zápis pomocí vodičů CLK (synchronizační hodiny) a DATA (datové bity od procesoru k displeji), jak lze vidět ze schématu zapojení. Čtení z displeje není jeho řadičem podporováno a příslušný datový vodič tedy není vyveden. Dále lze pomocí logického výstupu mikrokontroléru #RESET uvést displej do výchozího stavu a výstup #CS oznamuje

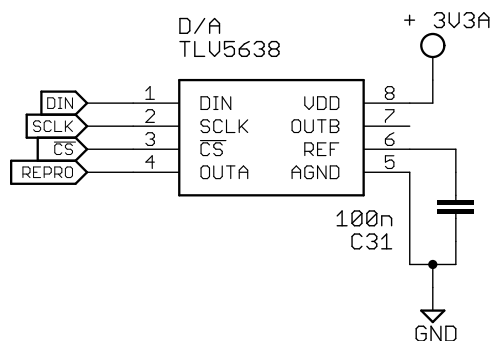
displeji zahájení nebo ukončení přenosu jednoho datového slova, které je v tomto případě dlouhé 9 bitů. Přes tranzistor Q8 je na vstup pro podsvícení displeje přivedeno napájení upravené signálem pulzně šířkové modulače (PWM), kterým lze řídit jeho střední hodnotu a tím intenzitu podsvícení. Displej je původně určený pro mobilní telefon NOKIA 6100 a návod k jeho obsluze je uveden v seznamu literatury pod označením [5].



Obr. 2.9: Signál PWM.

2.5.2 D/A převodník TLV5638

Pro zvukový výstup konzole je použit 12-bitový D/A převodník TLV5638 (schéma na obr. 2.10), za který je následně připojen zesilovač a reproduktor. Převodník slouží ke konverzi signálu v digitální podobě z procesoru na analogový, zpracovávaný reproduktorem. Komunikujeme s ním opět pomocí sériové komunikace s rozhraním SPI přes dva vodiče SCLK (synchronizační hodiny) a DIN (datové bity od procesoru k displeji). Tentokrát jsou datová slova dlouhá 16 bitů.

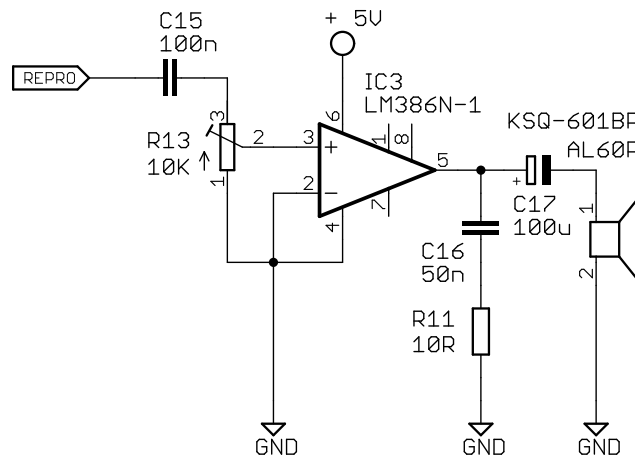


Obr. 2.10: D/A převodník TLV5638.

Pro napájení je použit analogový rozvod 3,3 V, oddělený od napájení digitálních obvodů filtrem. Důvodem je šum na digitálním rozvodu, jenž by nepříznivě ovlivňoval správnou funkčnost převodníku. Pomocí vstupu CS určujeme začátek a konec přenosu datového slova po sériové lince. Převodník má dva oddělené analogové výstupy, přičemž herní konzole využívá pouze jeden (OUTA). Ten dále vede na vstup zesilovače. Využita je vnitřní reference převodníku (možnosti jsou 1,024 V nebo 2,048 V), která je na vývodu REF oddělena od země kondenzátorem pro lepší stabilitu referenčního napětí. Detailní charakteristiky a zapojení převodníku jsou k nalezení v katalogovém listu uvedeném v seznamu literatury pod označením [6].

2.5.3 Zesilovač LM386 s připojeným reproduktorem

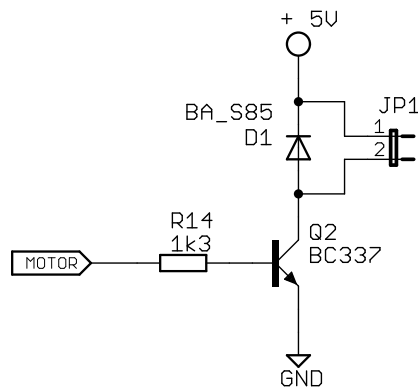
Jako integrovaný tranzistorový zesilovač je použit obvod LM386, jehož celé zapojení je znázorněno na obr. 2.11. Napájený je ze spínaného zdroje s výstupem 5 V. Na vstup je přes oddělující kondenzátor C15 a potenciometr R13 pro regulaci hlasitosti, který je zapojený jako klasický napěťový dělič, přiveden signál z A/D převodníku a jeho frekvence udává výšku výsledného tónu. Na výstup je v paralelní kombinaci připojen Boucherotův člen, zabraňující rozkmitání zesilovače, a přes oddělující kondenzátor reproduktor se vstupní impedancí 16 Ω . Katalogový list integrovaného obvodu je uveden v seznamu literatury s označením [7].



Obr. 2.11: Zesilovač LM386 s připojeným reproduktorem.

2.5.4 Vibrační motorek

Uživatelské výstupy doplňuje vibrační motorek na obr. 2.12. Napájen je 5 V a řízen proudem do báze výkonového tranzistoru Q2. Nemůže být připojen přímo na výstup mikroprocesoru, protože ten by nepokryl jeho proudový odběr. Dioda D1 má ochrannou funkci. Motor má totiž induktivní charakter a indukčnost při poklesu procházejícího proudu (po zavření tranzistoru Q2 a jejího odpojení od napájení) obrátí svoji polaritu a chová se jako zdroj. Musí tedy být umožněno její vybití, jinak dochází na indukčnosti k mnohonásobnému přepětí.



Obr. 2.12: Vibrační motorek.

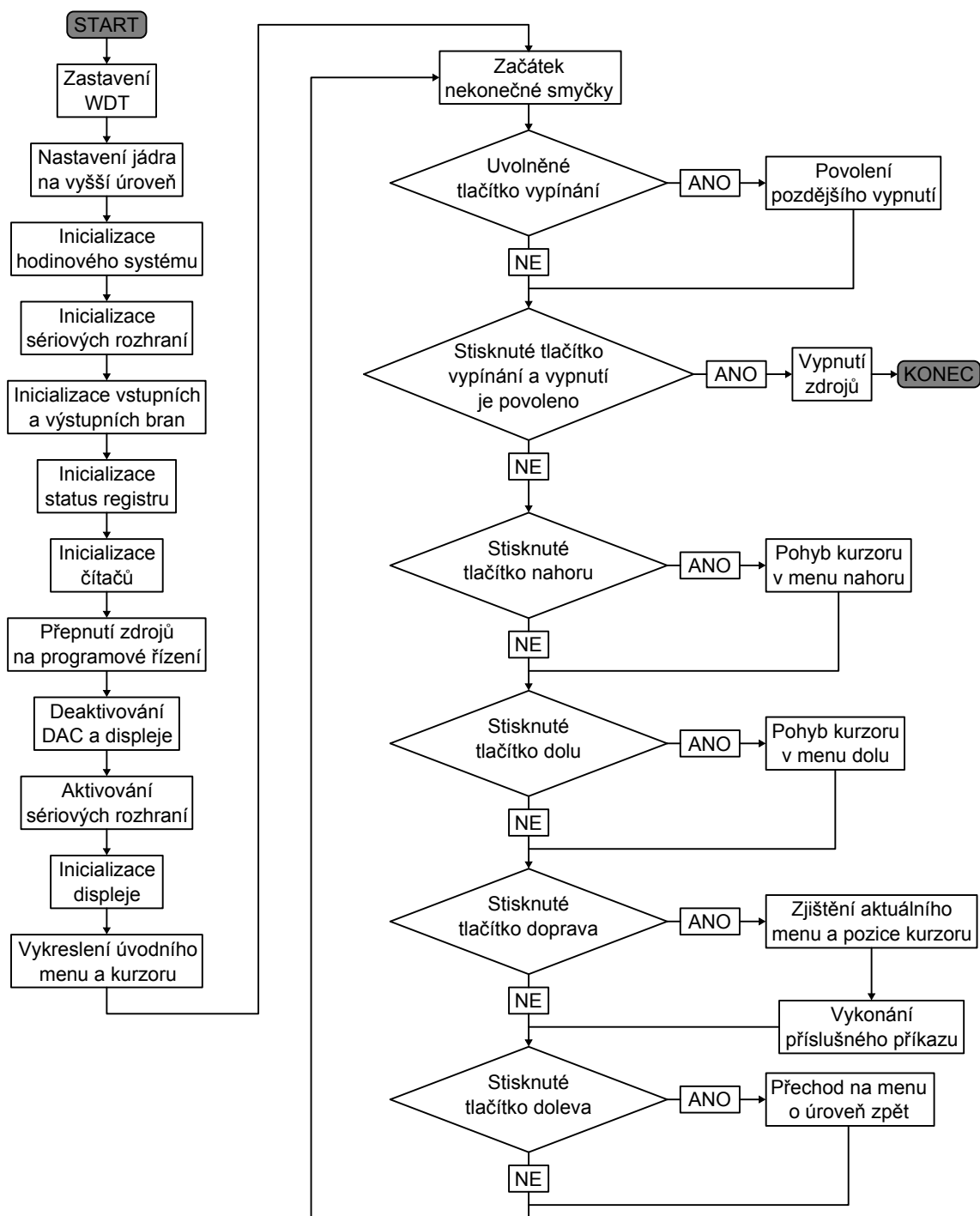
3

Software herní platformy

Následující kapitola se věnuje programovému vybavení vyrobeného zařízení. Ve vývojových diagramech přehledně zobrazuje strukturu hlavní funkce spouštěné při startu běhu procesoru a funkce pro jednotlivé hry fungující na konzoli. Dále se zabývá inicializačními funkcemi pro jednotlivé bloky mikrokontroléru. Veškeré použité zdrojové kódy (hlavní funkce, inicializace, hry, obsluha vstupních a výstupních periférií) jsou z důvodu velkého rozsahu k nalezení na CD-ROM nosiči přiloženému k bakalářské práci. Základem pro tvorbu kódu je katalogový list uvedený v seznamu literatury pod označením [8], který popisuje registry procesoru a postupy jejich nastavení podle požadované funkčnosti.

3.1 Popis souboru main.c

Soubor main.c obsahuje funkci stejného názvu, tedy main, neboli hlavní. Ta musí být součástí každého programu kontrolujícího činnost mikrokontroléru. Jedná se o funkci, která je spuštěna vždy po startu, a po průběhu prvotních inicializačních procesů se dostane do smyčky, která nesmí nikdy skončit, jinak se procesor ocitne v nekontrolovaném stavu. Nelze poté říci, co se bude v jeho blocích odehrávat a jaké hodnoty budou na vstupních a výstupních bránách. V krajních případech může vést k zablokování procesoru a podobným nežádoucím problémům. Ukončena může být pouze se současným vypnutím zdrojů a tedy celého hardwaru. Dále je součástí tohoto souboru blok obsluhující přerušování od čítače (vysvětleno dále v kapitole 3.2.4), který je vykonáván v přesných časových krocích. Konkrétně je zde nastavena frekvence 1 kHz tak, že čítač procesoru pracujícího na frekvenci 16 MHz přeteče jednou za 16000 kroků. Obsluhuje PWM pro kontrolní LED (Light Emitting Diode) a funkce jednotlivých her určené pro čítač (popsané v pozdějších kapitolách 3.3 a 3.4), pokud jsou spuštěné. Následující vývojový diagram na obr. 3.1 se zabývá funkcí main. Vysvětluje ji obecně bez vlastní syntaxe, kterou lze případně najít v příloze na CD-ROM. Jednotlivé vlastní naprogramované příkazy jsou popsány dále v kódu inicializačních funkcí (kapitola 3.2) nebo je lze najít v přiložených zdrojových kódech. Například funkce pro vykreslování různých tvarů a znaků na displej nejsou v textu podrobněji rozebírány. Vysvětlení mají v komentářích samotných souborů.



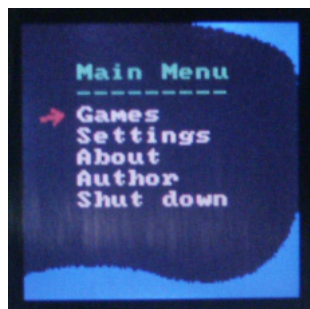
Obr. 3.1: Vývojový diagram funkce main.

Po startu procesoru tedy nejprve dojde k zastavení WDT (Watch Dog Timer). Ten funguje, jak již název vypovídá, jako hlídač pes. Pravidelně kontroluje, jestli nedošlo při běhu programu k zaseknutí bez dalšího východiska a řeší ho například opakovaným startem. Vyžaduje ale v kódu pravidelně se opakující bloky, které mu sdělují, že zmíněný stav nenastal. Navíc jeho činnost není v této herní konzoli nezbytná, protože program není až zas tak rozsáhlý, a proto není využit.

Z důvodu vyšší základní frekvence procesoru (16 MHz) je nutné na tento stav nejprve připravit vlastní jádro a nastavit je na příslušnou úroveň. Posloupnost příkazů je sestavena podle příkladů a ukázek kódu přímo od výrobce.

Následují inicializace jednotlivých bloků. Postupně hodinového systému - pracovní frekvence procesoru, sériových rozhraní pro komunikaci s displejem a D/A převodníkem, výstupních a vstupních bran podle připojených vnějších periférií, status registru - povolení globálního přerušení od libovolných bloků procesoru, čítačů pro generování PWM signálu pro podsvícení displeje a kontrolní LED a pro pravidelné vyvolávání přerušení zmíněné výše při popisu souboru main.c.

Poté dojde k přepnutí zdrojů na softwarové řízení, aby uživatel nemusel držet tlačítko napájení po celou dobu běhu programu. Deaktivují se displej a D/A převodník a aktivují se sériové komunikace, aby následně mohla proběhnout inicializace displeje. Veškeré funkce pro displej jsou vytvořené podle jeho návodu uvedeném v seznamu literatury pod označením [5] nebo z něj převzaté (např. knihovny textových znaků). Poté už se jen vykreslí zahajovací menu s kurzorem na základní pozici a řízení je předáno nekonečné smyčce.



Obr. 3.2: Obrazovka se základním menu a kurzorem.

Základním kamenem smyčky je cyklus while s počáteční podmínkou "1", která zaručuje nekonečný průběh. Stará se o uživatelské vstupy. Tlačítko pro vypínání a zapínání v případě uvolnění povolí následné vypnutí a poté při opětovném stisku vypne zdroj a tedy celé zařízení. Bez tohoto opatření by došlo k vypnutí ihned po zapnutí konzole. Uživatel by tlačítko musel uvolnit rychleji než se procesor dostane do nekonečné smyčky, což nemusí vždy platit. Tlačítka dolů a nahoru umožňují pohyb kurzoru v menu, tlačítko vlevo vrátí menu o krok zpět a tlačítko doprava vybere podle příslušné nabídky přechod na další menu, nastavení daného parametru, nebo spuštění požadované hry.

3.2 Inicializace procesoru

3.2.1 Hodinový systém

Nastavení hodinového systému začíná zvolením příslušných vývodů za vstupy pro krystaly a uvedením obou dvou krystalů (hlavní XT2 s frekvencí 4 MHz a hodinkový XT1 s frekvencí 32,768 kHz) do aktivního stavu. Součástí kontrolní smyčky rozvodu hodin je fázový závěs (PLL). Jeho výstup se pro začátek přidělí všem hodinovým rozvodům procesoru (hlavní - MCLK, sekundární - SMCLK a analogový - ACLK). Hlavní je nadále použit pro základní synchronizaci operací procesoru, sekundární pro různé periférie jako například čítače a sériové linky a analogový pro měření přesného času, pokud je požadováno. Nejprve se na vstup PLL nastaví vnitřní oscilátor a děličkou ve zpětné vazbě se zvedne výstupní frekvence PLL na 16 MHz, což je nyní hodnota synchronizační frekvence celého procesoru. Poté je nutné počkat na stabilizaci krystalů a celého PLL a až tehdy se vstup PLL přepne na krystal 4 MHz. Pokud bychom přepínali na krystaly dříve, procesor se povětšinou vůbec nerozběhne, protože nemá stabilní pracovní frekvenci. Změnou hodnot děliček opět nastavíme výstup na 16 MHz, což je požadovaná hodnota hlavní frekvence. Nakonec připojíme hodinkový krystal na analogový rozvod hodin (ACLK), aby ho bylo možné využít pro měření času.

3.2.2 Rozhraní sériových komunikací

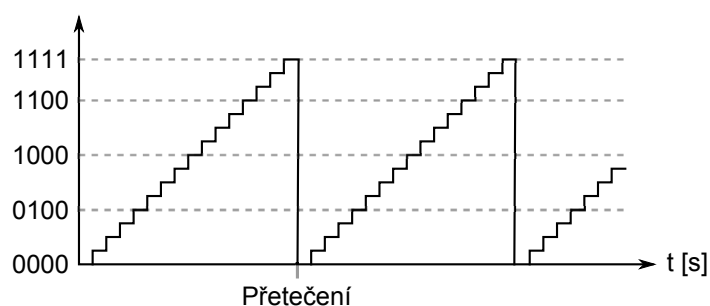
Inicializace pro obě sériová rozhraní je v podstatě identická, proto je zde uveden jen jeden příklad. Začíná se poskytnutím příslušných vývodů sériové lince. Zastaví se veškerá její aktivita a poté se provede konfigurace jednotlivých parametrů (fáze, případná inverze logické "0" a "1", počátek vysílání slova - od MSB (Most Significant Bit) nebo od LSB (Least Significant Bit), počet bitů ve slově, režim master nebo slave - nastavení, které zařízení na obou koncích linky synchronizuje komunikaci - u této herní konzole vždy procesor, což umožňuje režim master, druh sériové linky - u tohoto zařízení vždy SPI, synchronní nebo asynchronní režim - využít v obou případech synchronní). Parametry jsou nastavené podle potřeby zařízení, se kterým daná linka komunikuje. Dále určíme zdroj hodin, které udávají frekvenci synchronizačních pulsů a tedy celé komunikace. Použit je sekundární rozvod hodin (SMCLK), který má frekvenci, stejně jako hlavní rozvod, 16 MHz. Tuto frekvenci již dále nedělíme. Nakonec se zamezí modulaci přenosu, která není v konzoli potřebná.

3.2.3 Vstupní a výstupní brány

Brány procesoru, které tvoří skupina vždy osmi vývodů, lze nastavit zápisem do příslušného registru jako vstupní (logickou "0") nebo výstupní (logickou "1"). Veškeré brány jsou nastaveny podle okolních periférií připojených k procesoru podle toho, jestli jsou procesorem ovládané, nebo procesoru informace dodávají.

3.2.4 Čítače

V herní konzoli jsou využity dva čítače. První generuje přerušování při každém přetečení s frekvencí 1 kHz (zmněno v kapitole 3.1) a PWM signál pro kontrolní LED. Druhý pak vytváří PWM signál pro regulaci podsvícení displeje. Nastavení jsou shodná až na část pro konfiguraci přerušování, která u inicializace druhého čítače chybí, a na využití různých vývodů PWM signálů. To ale není pro demonstraci postupu důležité. Je zde tedy vysvětlena posloupnost příkazů pro správnou funkčnost prvního čítače. Nejprve se předají příslušné vývody PWM modulátoru a poté je nastaven hlavní registr čítače. Umožňuje vybrat zdroj synchronizačních hodin (zde SMCLK), případně děličku hodin (není využita) a směr čítání (v tomto případě od nuly k přetečení čítače). Další možnosti čítání jsou například opačně, nebo střídavě nahoru a dolů. Princip činnosti čítače je na obr. 3.3. V procesoru máme k dispozici 16-bitové čítače s 65536 stavy, ale pro zjednodušení a názornost postačí zobrazit 4-bitový, tedy se 16 stavy.

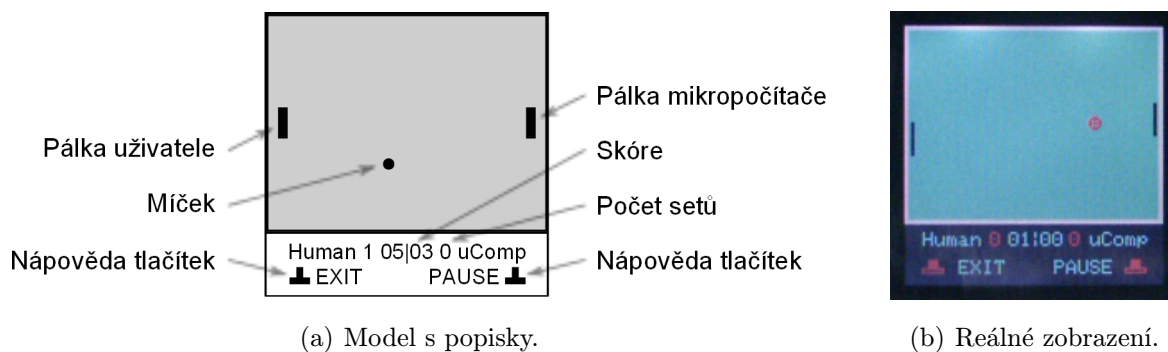


Obr. 3.3: Ukázka činnosti 4-bitového čítače se směrem čítání nahoru.

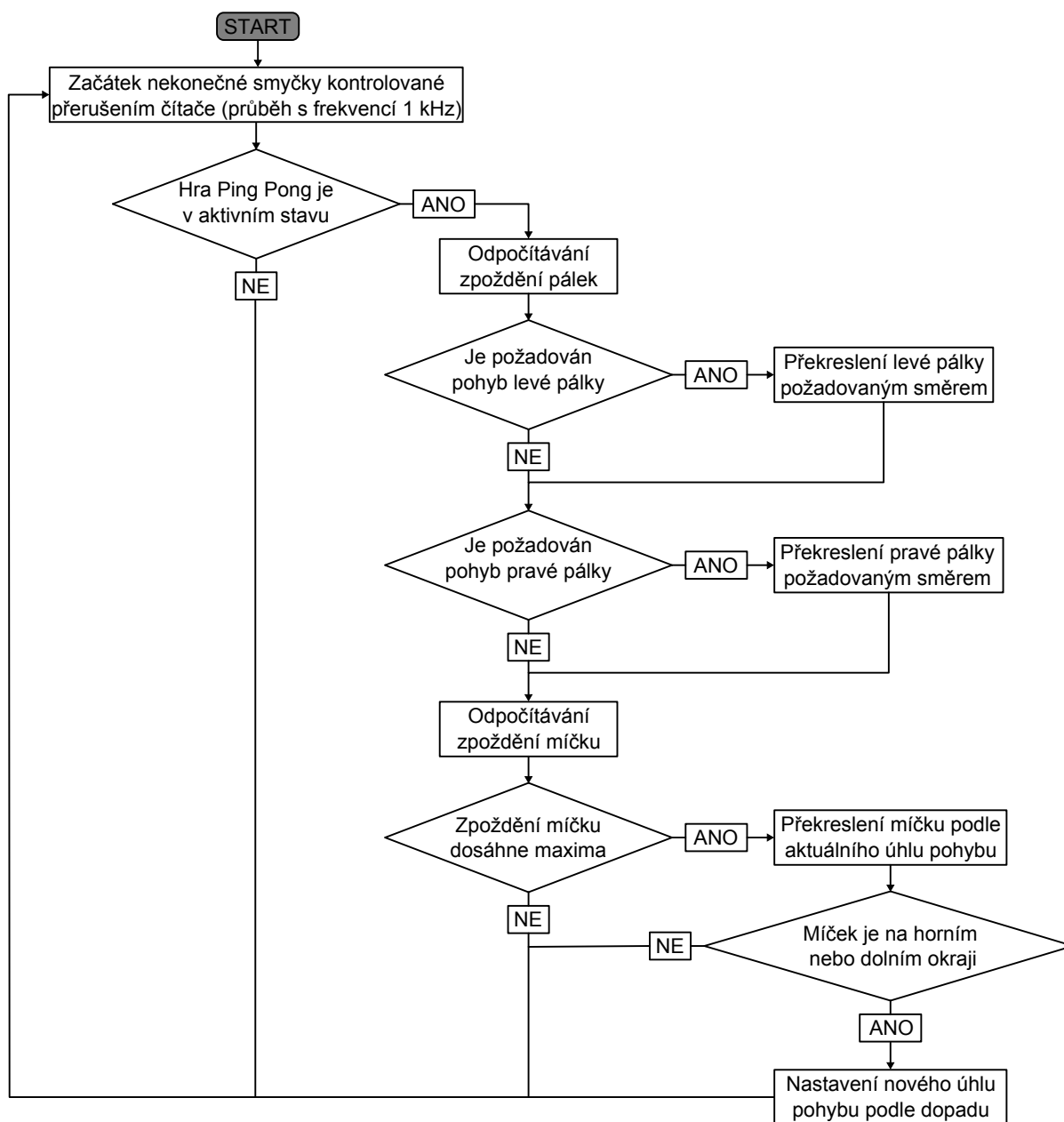
Poté povolíme vznik přerušování při každém přetečení čítače. To znamená při stavu, kdy se dosáhne maximální hodnoty a začne se počítat opět od nuly. V dalším kroku zvolíme tuto maximální hodnotu. Nakonec je vybrán blok čítače, který mimo jiné umožňuje tvorbu PWM signálu na zvoleném vývodu (pro podsvícení displeje nebo pro kontrolní LED), a je nakonfigurován tak, aby generoval tento signál s určenou střídou, jejíž hodnota je dána příslušným registrem.

3.3 Hra Ping Pong

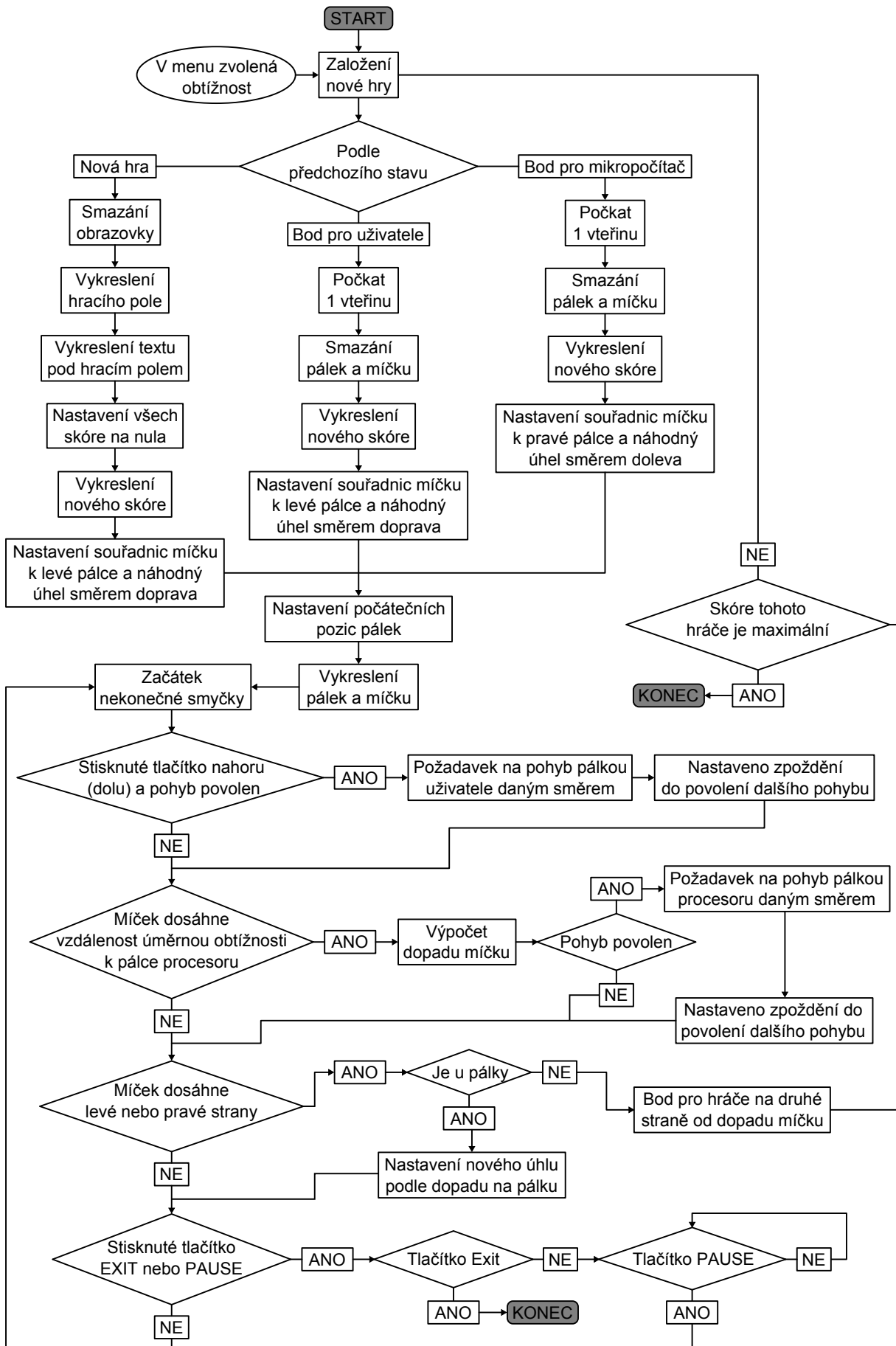
První ze dvou her v konzoli je klasický 2D Ping Pong. Při pohledu shora patří levá páčka uživateli a pravou ovládá umělá inteligence. Ta je založená na počítání souřadnic dopadu míčku a je do ní vložen určitý faktor náhody, aby nebyl průběh hry ze strany mikropočítače stále stejný. Pod hrací částí obrazovky se počítá skóre a při dosažení 2 setů, které se hrají do 11 bodů, na jedné či druhé straně se na obrazovku vypíše vítěz a hra se ukončí. Dole se poté zobrazuje nápověda pro funkce tlačítek umístěných na zadní straně. Na obr. 3.4 je znázorněna obrazovka při spuštění hry a následující vývojové diagramy (obr. 3.5 a obr. 3.6) vysvětlují fungování celé hry podle jejího zdrojového kódu.



Obr. 3.4: Obrazovka hry Ping Pong.



Obr. 3.5: Vývojový diagram funkce hry Ping Pong v přerušení čítače



Obr. 3.6: Vývojový diagram hlavní funkce hry Ping Pong

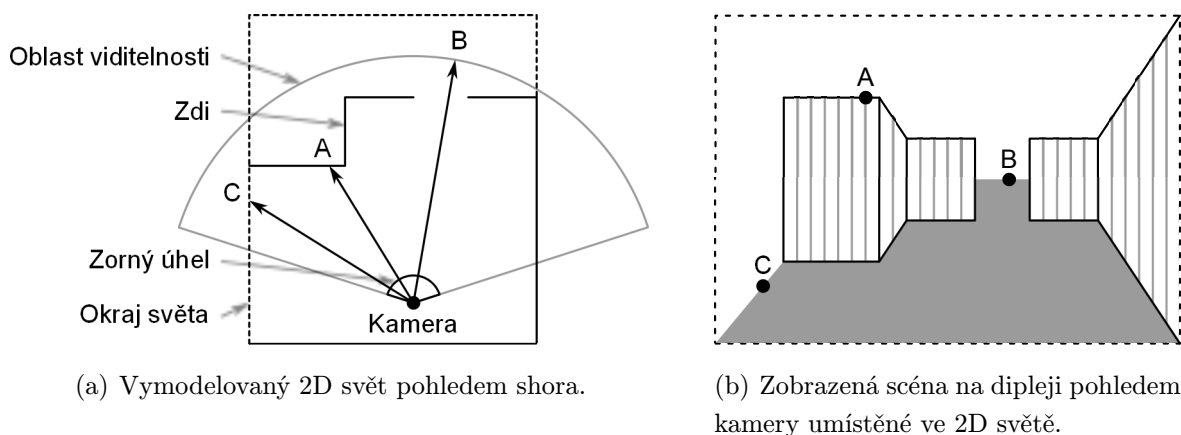
V přerušení čítače se vykonávají části hry požadující přesné opakování v čase. Nejprve se testuje, zda je vůbec hra v aktivním stavu, aby nedocházelo k překreslování obrazovky například při přerušení běhu uživatelem nebo při čekání na nové zahájení po chybě jednoho z hráčů. Pokud ano vykonávají se s frekvencí 1 kHz následující kroky nezávisle na stavu hlavní funkce. Odpočítává se zpoždění páčky uživatele i procesoru pro zmenšení jejich rychlosti tak, aby se míček nedal pokaždé stihnout. Pokud je vyžadován pohyb některé z pálek, což má na starost zjistit hlavní funkce hry, provede se její překreslení. Poté se odpočítává také zpoždění pro míček (opět aby nebyl jeho pohyb příliš rychlý a nepostřehnutelný). Pokud dosáhne maximální hodnoty, míček se přemístí na displeji v závislosti na aktuálním úhlu určujícím směr jeho pohybu a zpoždění se vynuluje. Nakonec se ještě otestuje, jestli nenastalo dosažení horního nebo dolního okraje obrazovky. V kladném případě se přepočítá úhel pohybu podle zákona odrazu.

V hlavní funkci nejprve dochází k vytvoření nového prostředí v závislosti na předchozím stavu. V případě, že se spouští zcela nová hra, musí se smazat obrazovka, vykreslit hrací pole a informace pod ním, vynulovat a vypsát skóre a zadat souřadnice míčku k levé páлке, od které se vždy po startu začíná. Pokud hra již probíhala a dochází k novému rozehrání po chybě na jedné či druhé straně, počká se 1 vteřinu, smažou se pouze páčky a míček, vykreslí se nové skóre a nastaví se počáteční pozice míčku na straně vítěze poslední výměny. Nakonec se ještě vždy zadají startovní pozice pálek a spolu s míčkem se nakreslí na plochu. Následně se již hra rozběhne (nastaví se její aktivní stav a tím se povolí funkce v přerušení čítače) a řízení se předá nekonečné smyčce.

Ve smyčce se kontrolují vstupy od uživatele, obsluhuje se umělá inteligence pravé páčky a dochází ke zjišťování konce výměny. Pokud je tedy stisknuté jedno z tlačítek pro pohyb levé páčky, nastaví se požadavek na změnu pozice příslušným směrem, která je následně vykonána v nejbližším kroku přerušení čítače. Zadá se také zpoždění než může dojít k dalšímu pohybu. Při dosažení vzdálenosti míčku od pravé páčky takové, že odpovídá zvolené obtížnosti (čím vyšší obtížnost, tím větší vzdálenost - mikroprocesor stihne vybrat více míčků), se vypočte podle úhlu a této vzdálenosti souřadnice dopadu míčku na pravou stranu. Aktivuje se požadavek na změnu místa pravé páčky a ta se s přerušením čítače začne přemísťovat směrem k zjištěné pozici. Opět s určitým časovým zpožděním mezi jednotlivými kroky než se znovu pohyb povolí. Dále se v momentu, kdy míček dosáhne levého nebo pravého okraje, zjišťuje jeho pozice vůči příslušné páлке. Pokud se nachází před pálkou, dojde k jeho odrazu nastavením nového úhlu. Ten se v tomto případě nerovná úhlu dopadu, ale je vybrán z dostupných možností podle vzdálenosti od místa, na jakém na páčku narazil, ke středu páčky. Když je naopak míček zcela mimo páčku, zvýší se skóre protějšího hráče a dojde ke spuštění hry od začátku tak, jak popisuje předchozí odstavec. Uživatel má ještě možnost hru přerušit stisknutím pravého zadního tlačítka, čímž se hra dostane do smyčky čekající na opětovný stisk (až po uplynutí určité doby, jinak by se hra pořád dokola přerušovala a spouštěla s nejednoznačným výsledkem), nebo ukončit pomocí levého zadního tlačítka.

3.4 3D grafika

Druhá hra je spíše než samotnou hrou ukázkou 3D grafiky v mikroprocesorových aplikacích. Umožňuje procházení 3D světa, podobného tomu, který známe z klasické první tří rozměrné hry Wolfenstein3D. Modelování 3D prostředí je založené na metodě ray-casting, neboli paprskové vykreslování. Každé vytvoření scény na obrazovku vychází z předem definovaného 2D světa (pohled shora), ve kterém neuvažujeme výšku objektů, ze souřadnic kamery (bod, ze kterého se díváme do fiktivního světa) a směru jejího pohledu. Dále je pevně zvolen zorný úhel. V tomto případě 130° . Pokud máme tyto vstupní parametry, můžeme přistoupit k samotnému výpočtu obrazu, který bude následně vykreslen na displej. Vše podstatné je znázorněno na obr. 3.7.



(a) Vymodelovaný 2D svět pohledem shora.

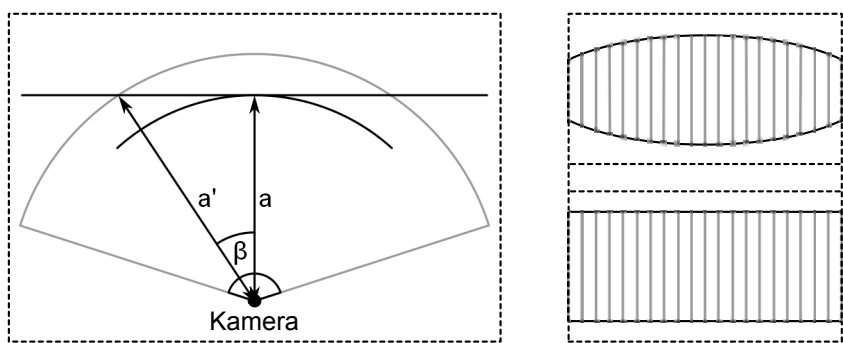
(b) Zobrazená scéna na displeji pohledem kamery umístěné ve 2D světě.

Obr. 3.7: Podstata metody ray-casting.

Z pozice kamery postupně vysíláme paprsky přes celý rozsah zorného úhlu a vždy testujeme vzdálenost, ve které se narazí na zeď (případ A na obrázku). Podle zjištěné délky dráhy se poté vypočte výška zdi pro daný úhel. Pokud paprsek putuje až do rozsahu viditelnosti, který si předem zvolíme, a na žádnou zeď nenarazí, přiřadíme tomuto úhlu nulovou výšku zdi (případ B). Nadále je ještě nutné ošetřit stav, kdy se vylétne mimo oblast definovaného 2D světa bez kontaktu se zdí dříve než se dosáhne rozsahu viditelnosti. Pro něj také platí přiřazení nulové výšky zdi (případ C).

Po tomto prozkoumání našeho 2D světa máme vytvořené pole výšky zdí v závislosti na zorném úhlu. Na první pohled se může zdát, že by nyní stačilo vykreslit jednotlivé úsečky na obrazovku. Musí se ovšem ošetřit ještě jeden problém a tím je efekt rybiho oka, který je názorně patrný z obr. 3.8. Při pohledu na přímou zeď dostaneme vyslání paprsku pod úhlem β vzdálenost a' , což způsobuje zakřivení. Aby byla zeď vodorovná, je nutné naměřit vzdálenost a ve všech bodech zdi. Toho docílíme jednoduchou korekcí všech naměřených vzdáleností podle vztahu v rov. 3.1, kde a' je vždy naměřená vzdálenost, a opravená vzdálenost a β aktuální úhel od osy pohledu. Tato korekce platí vždy, nezávisle na poloze a směru zdi.

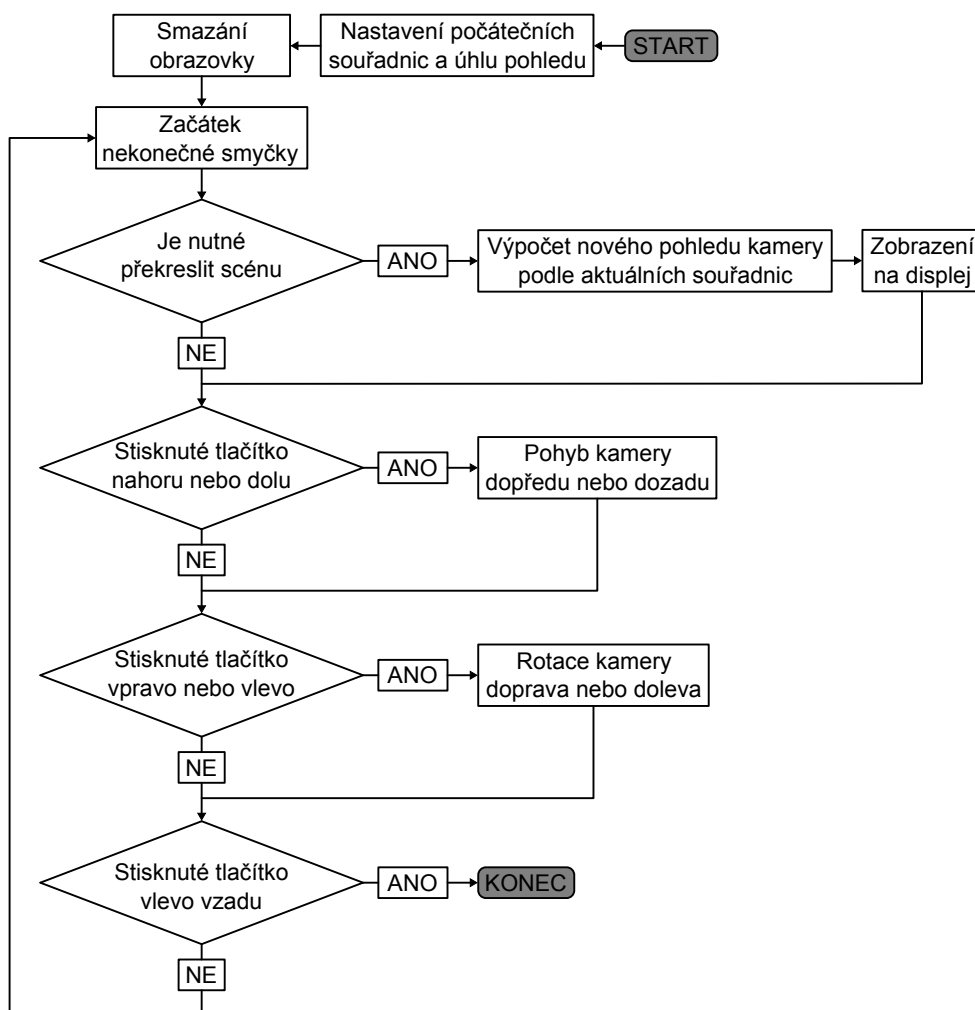
$$a = a' \cos(\beta) \quad (3.1)$$



(a) Vymodelovaný 2D svět.

(b) Pohled z kamery (nahore se zakřivením, dole po korekci).

Obr. 3.8: Efekt rybího oka.



Obr. 3.9: Vývojový diagram hlavní funkce pro vykreslování 3D grafiky

Na vývojovém diagramu z obr. 3.9 je vysvětlen průběh programu při vykreslování 3D grafiky podle vstupů od uživatele. Nejprve se zadají počáteční souřadnice kamery a smažou se z obrazovky předchozí údaje. Program poté dospěje do nekonečné smyčky. Každým průběhem se kontrolují vstupy od uživatele, kterými jsou tlačítka pro pohyb ve virtuálním světě a možnost ukončení aplikace a návrat do menu. Dále se zjišťuje, jestli nenastala od předchozího kroku změna v umístění nebo směru pohledu kamery. V případě, že k ní došlo, vypočítá se nová scéna podle výše popsanych postupů, která je následně vyobrazena na displej. Kreslí se pouze body, které se liší od předchozího pohledu, z důvodu plynulosti přechodů.



Obr. 3.10: Obrazovka 3D grafiky.

Na závěr sekce o 3D grafice je ještě přiložen použitý algoritmus pro výpočet scény k zobrazení na displej z pozice ve vymodelovaném 2D prostředí. Funkci se dodávají za parametry umístění kamery a úhel směru jejího pohledu. Postup je rozdělen do dvou identických částí, kdy se postupně zkoumají pro každý vyslaný paprsek v zorném úhlu (zajišťuje cyklus FOR na řádce 4 zobrazeného kódu) zvlášť zdi kolmé na směr osy x a na směr osy y (část pro osu x začíná na řádce 7 a pro osu y na řádce 41). Každý z obou bloků probíhá tak, že se nejprve nastaví do proměnných x a y počáteční pozice kamery. Následně se posune souřadnice zkoumané osy tak, aby se narazilo na nejbližší hranici mřížky 2D světa, kde každá buňka má 64×64 bodů, ve směru dané souřadnice a dopočítá se druhá souřadnice podle úhlu vyslaného paprsku. Poté se mění vždy o daný krok souřadnice zkoumané osy a dopočítává se souřadnice té druhé opět podle úhlu aktuálního vyslaného paprsku (smyčky WHILE na řádcích 24 a 58 pro oba jednotlivé bloky). Krok je pokaždé o 64 podle rozměru jedné buňky. Tím je zaručeno, že se žádná buňka nepřeskočí, že se na zeď narazí vždy na její hranici a navíc se minimalizuje počet nutných kroků. Každým průběhem smyčky se testuje vylétnutí paprsku mimo rozsah světa (za vzdálenost je dosazena v tomto případě hodnota vyšší než je nejvyšší možná dosažená ve 2D světě) a náraz na zeď (vypočte se vzdálenost ze změn obou souřadnic od počáteční polohy) a při splnění jedné z podmínek se smyčka ukončí. Nakonec se porovnají obě vzdálenosti a použije se ta kratší, která je upravena redukcí efektu rybího oka a přepočtena na výšku zdi pro daný úhel. Kratší se musí vzít z toho důvodu, že každý blok zkoumá pouze zdi kolmé na svoji osu a předem nelze říci, které zdi se dosáhne dříve.

```

1 void TransformView(unsigned int x0, unsigned int y0, signed int viewAngle0)
2 {
3     step = 0;
4     for(viewAngleStep=viewAngle0-65;viewAngleStep<=viewAngle0+65;viewAngleStep++) {
5         viewAngle = (viewAngleStep + 360) % 360;
6
7         x = x0;
8         y = y0;
9
10        if(viewAngle % 180 == 90) z = 60000;
11        else {
12            if(viewAngle < 90 || viewAngle > 270) {x -= ((x0 % 64) + 1); dir = 1;}
13            else {x += (64 - (x0 % 64)); dir = 2;}
14
15            if(viewAngle > 180) {
16                if(dir == 1) y -= (tangAngle[viewAngle % 180] * ((x0 % 64) + 1) / 64);
17                if(dir == 2) y -= (tangAngle[viewAngle % 180] * (64 - (x0 % 64)) / 64);
18            }
19            else {
20                if(dir == 1) y += (tangAngle[viewAngle % 180] * ((x0 % 64) + 1) / 64);
21                if(dir == 2) y += (tangAngle[viewAngle % 180] * (64 - (x0 % 64)) / 64);
22            }
23
24            while(1) {
25                if(x < 0 || x > 3200 || y < 0 || y > 3200) {z = 60000; break;}
26
27                if(world2DMatrix[x/64][y/64] == 1) {
28                    if(dir == 1) z = (64 * (x0 - x)) / sinAngle[(viewAngle + 90) % 180];
29                    if(dir == 2) z = (64 * (x - x0)) / sinAngle[(viewAngle - 90)];
30                    break;
31                }
32
33                if(dir == 1) x -= 64;
34                if(dir == 2) x += 64;
35
36                if(viewAngle >= 180) y -= tangAngle[viewAngle % 180];
37                else y += tangAngle[viewAngle];
38            }
39        }
40
41        x = x0;
42        y = y0;
43
44        if(viewAngle % 180 == 0) zNext = 60000;
45        else {
46            if(viewAngle > 180) {y -= ((y0 % 64) + 1); dir = 1;}
47            else {y += (64 - (y0 % 64)); dir = 2;}
48
49            if(viewAngle < 90 || viewAngle > 270) {
50                if(dir == 1) x -= tangAngle[(450 - viewAngle) % 180] * ((y0 % 64) + 1) / 64;
51                if(dir == 2) x -= tangAngle[(450 - viewAngle) % 180] * (64 - (y0 % 64)) / 64;
52            }
53            else {
54                if(dir == 1) x += tangAngle[(450 - viewAngle) % 180] * ((y0 % 64) + 1) / 64;
55                if(dir == 2) x += tangAngle[(450 - viewAngle) % 180] * (64 - (y0 % 64)) / 64;
56            }
57
58            while(1) {
59                if(x < 0 || x > 3200 || y < 0 || y > 3200) {zNext = 60000; break;}
60
61                if(world2DMatrix[x/64][y/64] == 1) {
62                    if(dir == 1) zNext = (64 * (y0 - y)) / sinAngle[viewAngle % 180];
63                    if(dir == 2) zNext = (64 * (y - y0)) / sinAngle[viewAngle];
64                    break;
65                }
66
67                if(dir == 1) y -= 64;
68                if(dir == 2) y += 64;
69
70                if(viewAngle < 90 || viewAngle > 270) x -= tangAngle[(450 - viewAngle) % 180];
71                else x += tangAngle[(450 - viewAngle) % 180];
72            }
73        }

```

```
74
75     if(zNext < z) z = zNext;
76     z = z * sinAngle[(step + 25)] / 64;
77
78     if(z < 6400) {
79         viewMatrix[step] = (64 * 100) / z;
80         if(viewMatrix[step] > 132) viewMatrix[step] = 132;
81     }
82     else {
83         viewMatrix[step] = 0;
84     }
85
86     step++;
87 }
88 }
```

Celé toto opatření s rozčleněním do dvou bloků zajišťuje daleko menší počet kroků, než kdyby se testoval každý paprsek jen jeho příslušným směrem a ne rozdělený do směrů os, protože v takovém případě nelze provádět kroky po 64 bodech, ale vždy po jednom (aby se narazilo na zeď se správnou vzdáleností k jejímu okraji). Zrychlení algoritmu je přibližně 32 krát. Krok je sice 64 krát větší, ale musí se provést dva výpočty pro každý paprsek. Oba kódy byly otestované i při reálném běhu aplikace na konzoli a procházení světa s použitím druhého algoritmu je opravdu daleko plynulejší. Omezením je pak spíše než výpočetní algoritmus odezva samotného displeje.

4

Meření spotřeby herní konzole

Poslední kapitola se věnuje zjištění spotřeby elektrické energie mikrokontroléru při současném běhu základních podpůrných periférií (jakýsi nečinný, ale ne vypnutý režim) a poté i celé herní platformy za plného provozu. Měření nečinného režimu probíhá při zapnutém zdroji 3,3 V pro digitální obvody se současným vypnutím ostatních nepotřebných bloků. Stále ovšem běží například krystaly, které vypnout nelze. Dále jsou problémem použité spínané zdroje, které i při logické nule na vstupu EN (tedy v případě, kdy by měli být zcela vypnuté) nemají díky své konstrukci na výstupu nulové napětí a díky tomu dodávají určité napětí i ostatním zařízením za ně připojených, čímž zvyšují odběr elektrického proudu. Spotřeba celého zařízení je poté brána za stavu plného provozu obou zdrojů (na 3,3 V i 5 V) a dalších hardwarových bloků. Tabulky obsahují jednotlivá měření s několika opakováními, z nichž jsou následně vypočítané průměrné hodnoty. Zaznamenáno bylo vždy deset hodnot pro každý režim v intervalech po jedné minutě.

Číslo měření	Odebíraný elektrický proud [mA]
1	13,72
2	13,68
3	13,70
4	13,69
5	13,71
6	13,73
7	13,72
8	13,71
9	13,70
10	13,69
Průměr	13,71

Tab. 4.1: Naměřené hodnoty spotřeby při režimu nečinnosti.

Číslo měření	Odebíraný elektrický proud [mA]
1	164,7
2	164,4
3	164,9
4	165,1
5	165,4
6	164,4
7	164,5
8	163,5
9	164,7
10	165,0
Průměr	164,66

Tab. 4.2: Naměřené hodnoty spotřeby zařízení při plném provozu.

Z hodnoty celkové spotřeby při obou zdrojích v aktivním stavu plyne, že vezmeme-li v úvahu použití baterií typu AAA s kapacitou 1000 mAh, vydrží zařízení v nepřetržitém provozu až přibližně 6 hodin. Při srovnání obou výsledků je dále vidět, že odběr při režimu nečinnosti je řádově menší, než odběr celého zařízení, což také odpovídá předpokladům. Nejhoršími bloky na spotřebu energie jsou například podsvícení displeje nebo zesilovač.

Z časových důvodů nebyly implementovány všechny zásady pro podporu nízké spotřeby a může to být zajímavým tématem dalšího rozvoje práce. Možnosti jsou například uspávání procesoru v době nečinnosti a dále vypínání zdroje napětí o hodnotě 5 V při neaktivním stavu zvukového výstupu a vibračního motorku a tím se přiblížit spotřebě nečinného režimu. Významným faktorem uspořené energie by bylo také nahrazení spínaných zdrojů jinými, které by měly vylepšenou konstrukci s podporou nulového napětí na výstupu, což současně, jak již bylo zmíněno, neobsahují. Vypínání zdroje pro 5 V by pak bylo mnohem efektivnější. Tato nedokonalost zdrojů je již v současné době částečně řešena odpojováním baterií od zařízení přes MOSFET tranzistor (k nahlédnutí na obr. 2.3) alespoň při zcela vypnutém zařízení. Před tímto vylepšením byl změřen elektrický proud odebíraný konzolí při vypnutí obou zdrojů 0,5 mA. Byl vybrán tranzistor s nízkým elektrickým odporem mezi hradly DRAIN (D) a SOURCE (S), aby na něm nevznikal příliš velký nežádoucí úbytek napětí. Toto opatření zamezí zbytečnému vybíjení baterií v době nevyužívání zařízení bez nutnosti jejich vyjímání.

5

Závěr

V bakalářské práci jsem se zabýval návrhem přenosné herní platformy na způsob PSP (Play Station Portable). Důraz byl kladen především na rozměry vhodné k ruční manipulaci, na spotřebu a tím i výdrž celého zařízení a na funkčnost po stránce hrátelnosti. Nízkou spotřebu podporuje zejména procesor od firmy Texas Instruments, který se řadí mezi nízkoodběrové součástky. Programové vybavení obsahuje prozatím jednu hru a ukázkou tří rozměrné grafiky.

Aktuální stav řešení problému ke dni publikace práce je tedy hotový koncept herní konzole, umožňující ovládání zařízení pomocí několika tlačítek s výstupem na barevný grafický displej. Při startu je uživatel uveden do hlavního menu, kde má možnost výběru z několika položek. Těmi jsou seznam her, nastavení, informace o autorovi, údaje o projektu a vypnutí konzole. Obsažena je funkční hra Ping Pong, kde uživatel změří síly s naprogramovanou umělou inteligencí s možností zvolení několika obtížností, které se liší schopností protivníka stíhat jednotlivé výměny. Dále je k dispozici ukáзка aplikace pro vykreslování 3D prostředí pomocí mikrokontroléru prostřednictvím procházení jednoduchého labyrintu. Možnosti jsou rotace pohledu a pohyb dopředu nebo dozadu. Tímto jsou splněny požadavky zadání.

Hardwarový návrh nadále počítá s dalšími vstupními a výstupními prvky (popsané v kapitole 2), které mohou být společně se snižováním spotřeby zmíněné v poslední kapitole zajímavými možnostmi k doplnění při rozvoji práce v budoucnosti. Konkrétně jsou připravené vývody pro vibrační motorek s joystickem, jejichž pozdější dodání je tedy s konzolou plně kompatibilní a není tudíž žádným problémem. Pouze otázkou vylepšení programového vybavení mikrokontroléru jsou oživení akcelerometru a D/A převodníku pro zvukový výstup, které jsou již na desce plošných spojů fyzicky přítomny.

Po stránce hardwarové je tedy zařízení téměř kompletní a další směry ve vývoji jsou proto především softwarového charakteru. Jednak je možnost využití zbylých periférií, jak již bylo řečeno, nebo například jejich komplexnější zapojení u jednotlivých aplikací. Místo v paměti procesoru je stále poměrně nevyužité (celkově máme k dispozici 128 kB a momentálně je obsazeno přibližně 10 kB) a nejperspektivnější oblastí je proto z mého hlediska vývoj dalších her, který je v podstatě při ohledu na výkon mikrokontroléru neomezený.

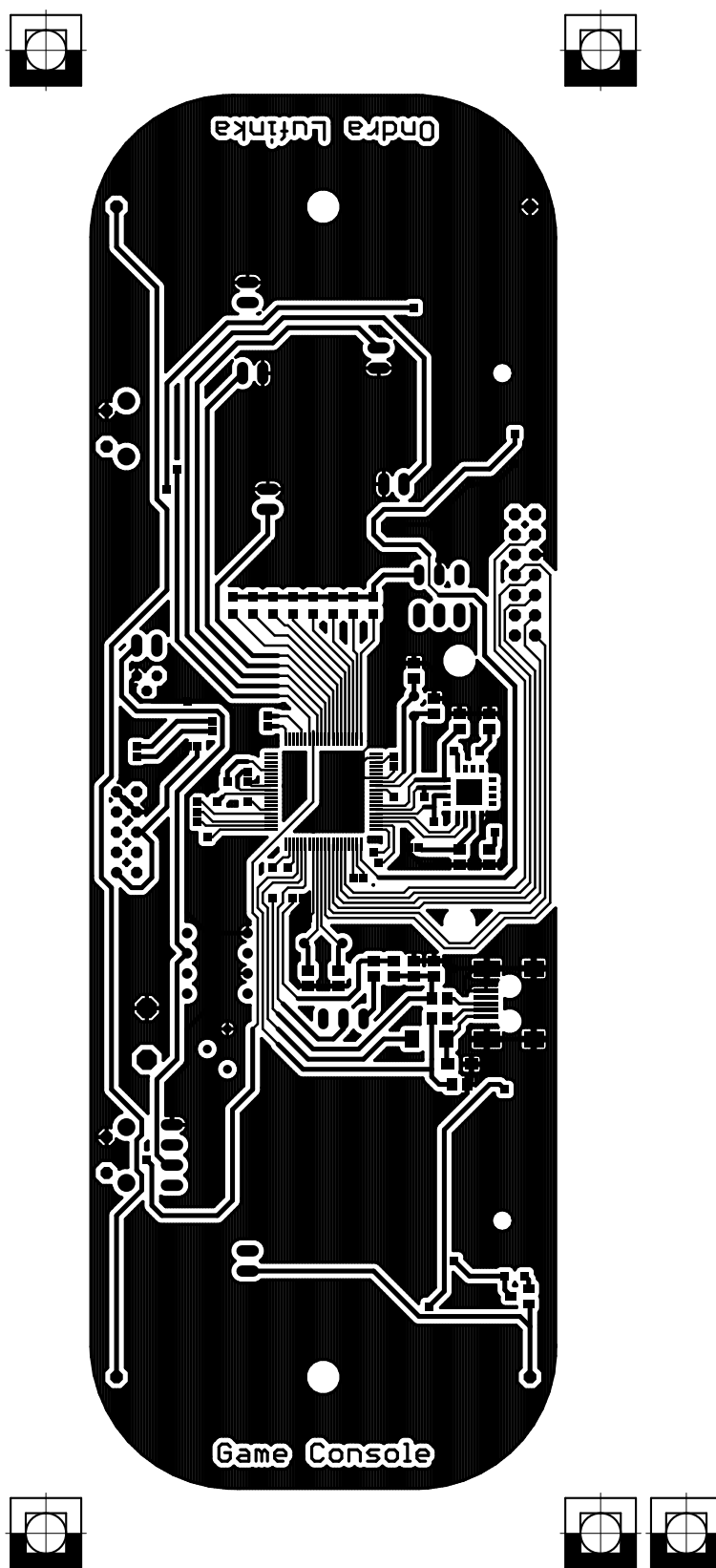
Literatura

- [1] Texas Instruments. *Mixed Signal Microcontroller* [online]. Dallas, Texas: Copyright ©2012, Texas Instruments Incorporated, březen 2009. Poslední změna únor 2012 [cit. 26.5.2012]. Dostupné z: <http://www.ti.com/lit/ds/symlink/msp430f5529.pdf>
- [2] Texas Instruments. *Low Input Voltage Step-Up Converter In 6 Pin SC-70 Package* [online]. Dallas, Texas: Copyright ©2011, Texas Instruments Incorporated, leden 2009 [cit. 26.5.2012]. Dostupné z: <http://www.ti.com/lit/ds/symlink/tps61220.pdf>
- [3] Texas Instruments. *MSP430 Hardware Tools* [online]. Dallas, Texas: Copyright ©2011, Texas Instruments Incorporated, květen 2009. Poslední změna prosinec 2011 [cit. 26.5.2012]. Dostupné z: <http://www.ti.com/lit/ug/slau278h/slau278h.pdf>
- [4] Freescale Semiconductor. *±1.5g - 6g Three Axis Low-g Micromachined Accelerometer* [online]. Chandler, Arizona: ©Freescale Semiconductor, Inc. 2005, duben 2005 [cit. 26.5.2012]. Dostupné z: <http://www.alldatasheet.com/datasheet-pdf/pdf/103487/MOTOROLA/MMA7260Q.html>
- [5] James P. Lynch. *Nokia 6100 LCD Display Driver* [online]. Grand Island, New York: 7.7.2007 [cit. 26.5.2012]. Dostupné z: <http://www.sparkfun.com/tutorial/Nokia%206100%20LCD%20Display%20Driver.pdf>
- [6] Texas Instruments. *2.7-V TO 5.5-V Low-Power Dual 12-Bit Digital-To-Analog Converter With Internal Reference And Power Down* [online]. Dallas, Texas: Copyright ©2012, Texas Instruments Incorporated, červen 1999. Poslední změna leden 2004 [cit. 26.5.2012]. Dostupné z: <http://www.ti.com/lit/ds/slas225c/slas225c.pdf>
- [7] National Semiconductor. *LM386 Low Voltage Audio Power Amplifier* [online]. ©2000 National Semiconductor Corporation, leden 2000 [cit. 26.5.2012]. Dostupné z: <http://www.alldatasheet.com/datasheet-pdf/pdf/8887/NSC/LM386.html>
- [8] Texas Instruments. *MSP430x5xx/MSP430x6xx Family User's Guide* [online]. Dallas, Texas: Copyright ©2011, Texas Instruments Incorporated, červen 2008. Poslední změna prosinec 2011 [cit. 26.5.2012]. Dostupné z: <http://www.ti.com/general/docs/lit/getliterature.tsp?baseLiteratureNumber=SLAU208&track=no>

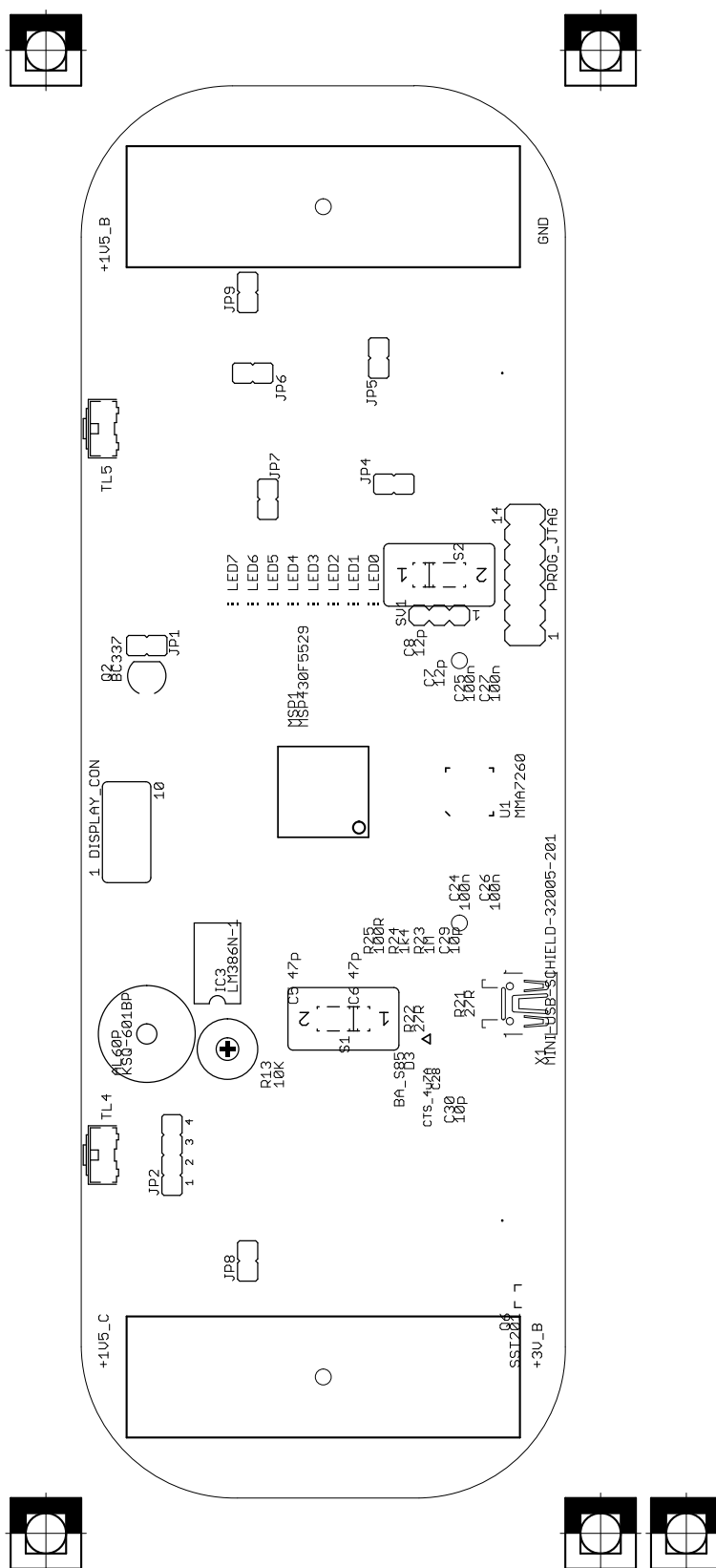
Příloha A

Desky plošných spojů

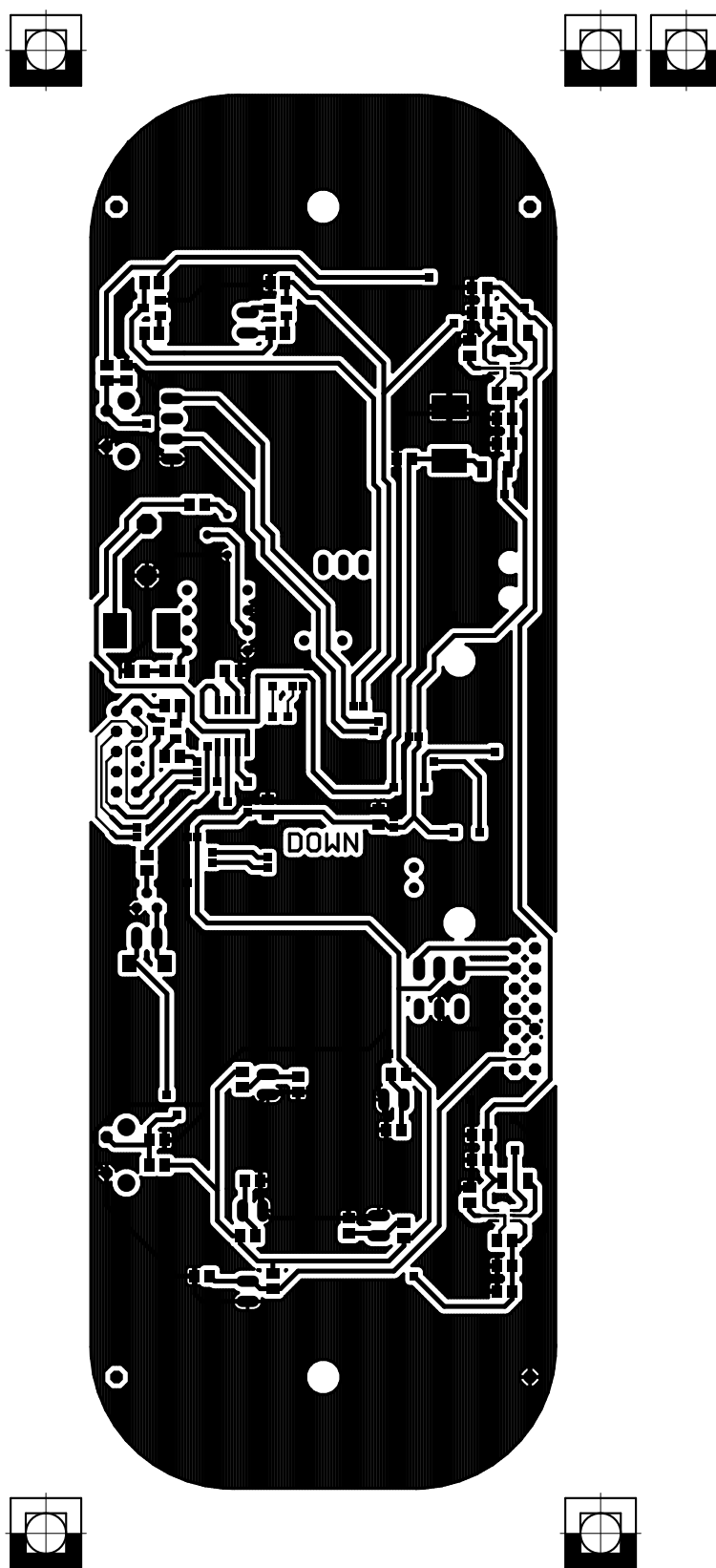
A.1 Horní a spodní deska plošných spojů včetně potisku



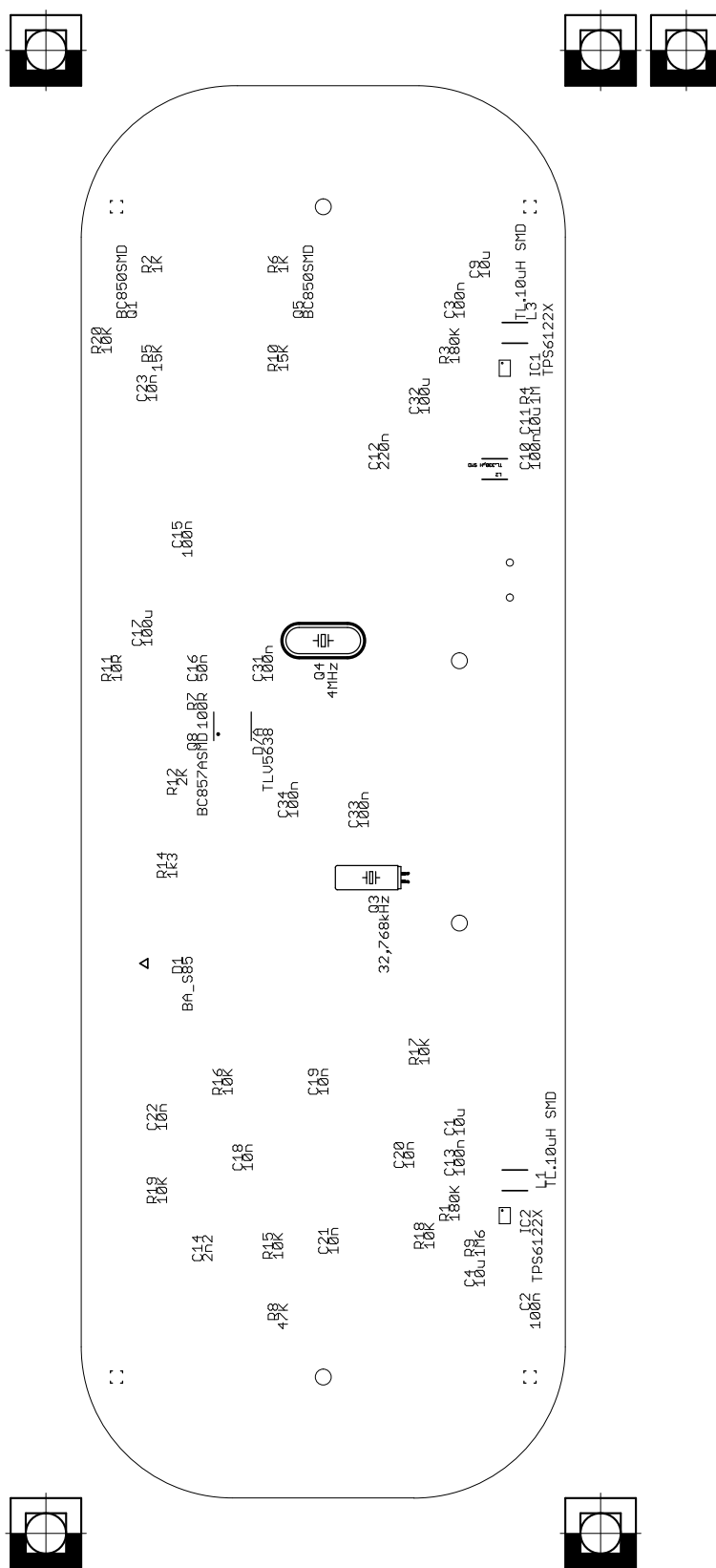
Obr. A.1: Horní strana dolní desky plošných spojů herní platformy.



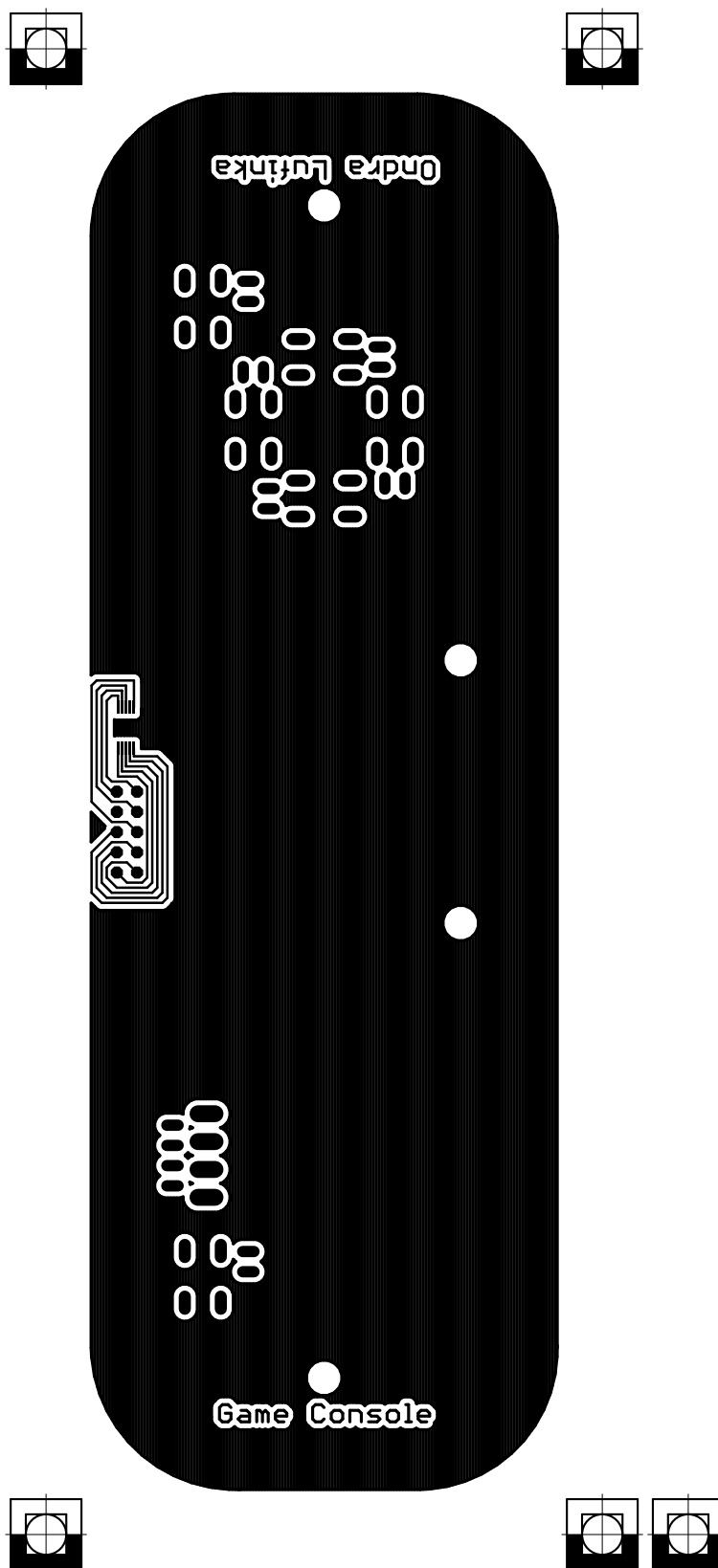
Obr. A.2: Horní potisk dolní desky plošných spojů herní platformy.



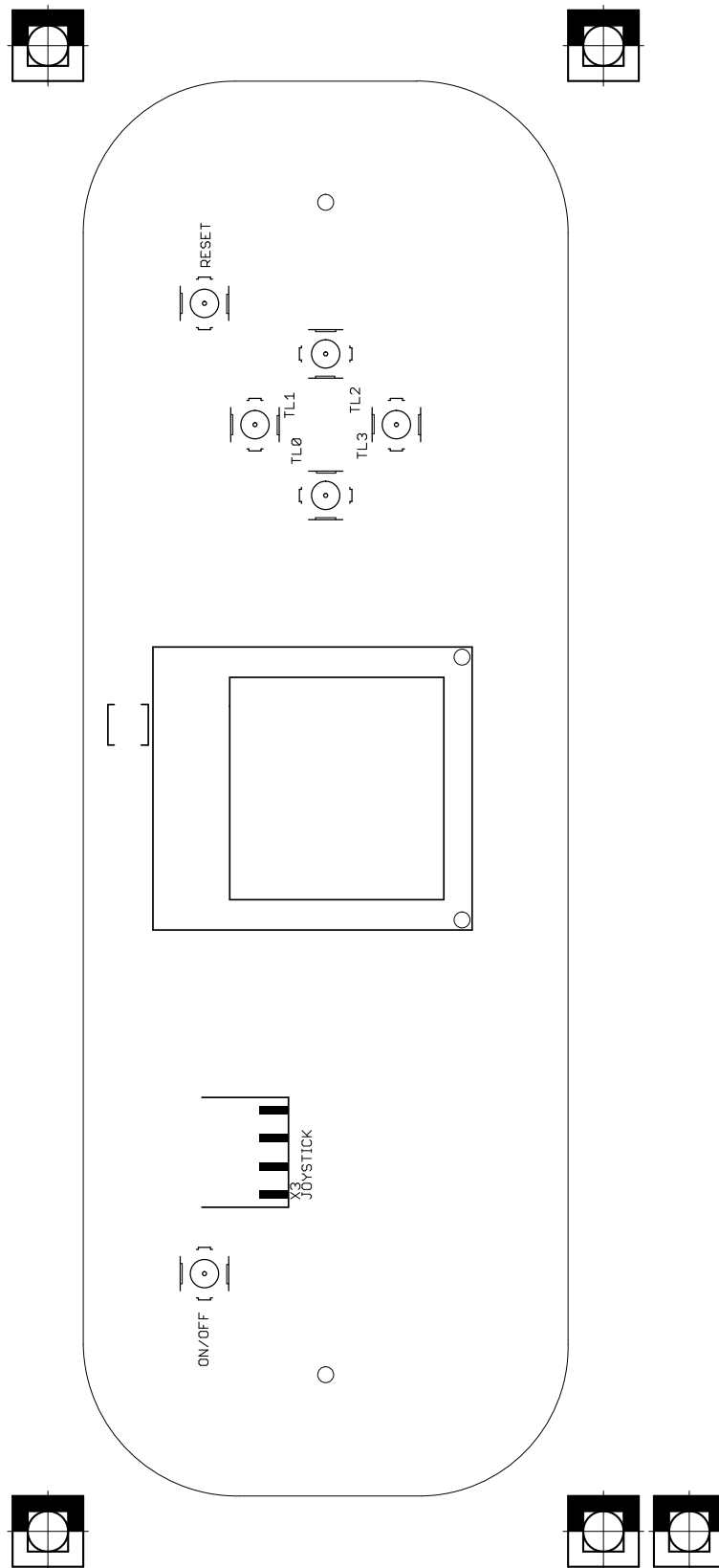
Obr. A.3: Dolní strana dolní desky plošných spojů herní platformy (zrcadleně).



Obr. A.4: Dolní potisk dolní desky plošných spojů herní platformy (zrcadleně).



Obr. A.5: Horní strana horní desky plošných spojů herní platformy. Dolní není zobrazena. Obsahuje pouze propojení ovládacích prvků s konektory.



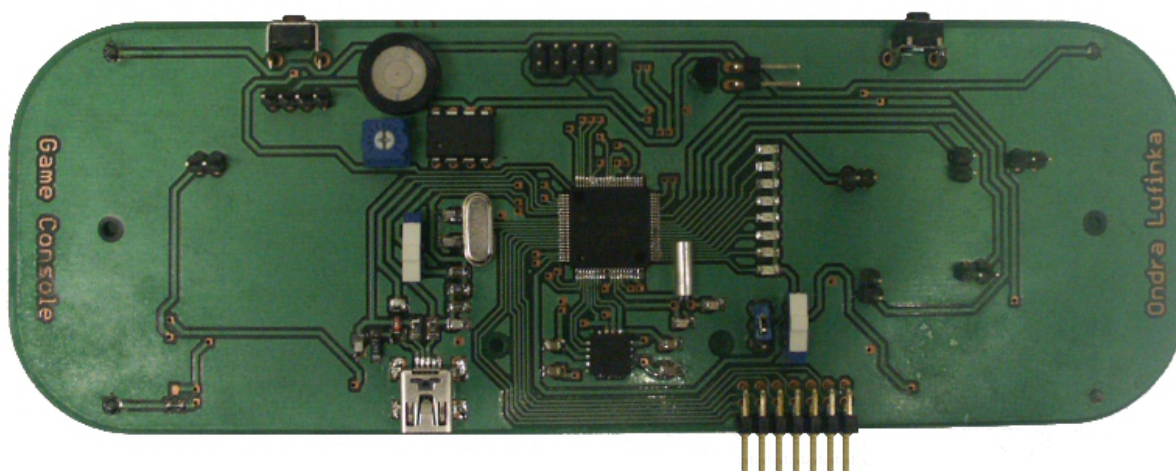
Obr. A.6: Horní potisk horní desky plošných spojů herní platformy.

Příloha B

Fotografie hotového výrobku



(a) Horní deska s displejem a ovládacími prvky.



(b) Dolní deska s mikroprocesorem a dalším hardwarem.

Obr. B.1: Hotové a osazené desky plošných spojů.