**Členové týmu**

* Vít Jánoš (zodpovědný za obsluhu sedmisegmentového displeje)
* Vojtěch Kudela (zodpovědný za obsluhu maticové klávesnice a správu GitHub)
* David Matějček (zodpovědný za komparátor a testování)
* Antonín Putala (zodpovědný za generátory, PWM modulátor a demo nahrávky)

**Teoretický popis a vysvětlení**

**Zvuk** je podélné mechanické vlnění. Zvuková vlna působí výchylku od atmosférického tlaku. Aby byl zvuk pro člověka slyšitelný, musí být o kmitočtu 20 Hz až 20 kHz a větší výchylku než práh slyšení *P*0 = 2·10-5 Pa. Dlužno podotknout, že citlivost lidského sluchu je frekvenčně závislé, práh slyšení i kmitočtový rozsah se liší dle jednotlivce. Lidské ucho vnímá hlasitost logaritmicky, proto se hlasitost vyjadřuje v decibelové míře.

*L*P je hladina akustického tlaku.

Periodické zvukové signály se nazývají **tóny**. Tyto tóny mají svou základní frekvenci, která je označovaná jako výška. Podle výšky také nese tón název. V hudební teorii se používá označení oktáva pro zdvojnásobení kmitočtu. Pokud zdvojnásobíme kmitočet tónu, ponese stejný název pouze bude ležet ve vyšší oktávě. Jako základní tón se užívá komorní A s kmitočtem *f* = 440 Hz. Oktáva se v evropské hudební teorii rozděluje na 12 dílů. Ty se nazývají půltóny. Každému půltónu je přiřazen kmitočet. Tento kmitočet se určí jako součin základního kmitočtu a poměru pro konkrétní půltón. Pokud změníme frekvenci základního tónu, změní se kmitočty všech půltónů.

Princip, podle kterého jsou přiřazovány poměry k půltónům nazýváme **ladění**. V minulosti byl užívány různé typy ladění, které měly své výhody i neduhy (např.: Pythagorejské, Středotónové, Parejovo…). V současném době se používá téměř výhradně rovnoměrně temperované ladění. V rovnoměrně temperovaném ladění je poměr mezi dvěma po sobě jdoucími půltóny .

Z tónu A se tedy získá A# vynásobením .

A tímto způsobem by bylo pokračováno pro zbývající půltóny.

**Tab. 1 Tabulka kmitočtů jednotlivých tónů**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Název | Kmitočet [Hz] | Název | Kmitočet [Hz] | Název | Kmitočet [Hz] |
| A | 440,0 | C# | 554,4 | F | 698,5 |
| A# | 466,2 | D | 587,3 | F# | 740,0 |
| B\* | 493,9 | D# | 622,3 | G | 784,0 |
| C | 523,3 | E | 659,3 | G# | 830,6 |

\* V českých a německých zemích je zvykem nazývat tento tón H. Oproti tomu jako B je označován tón A#. Zde je uváděné pojmenování obvyklé v anglosaských zemích.

Bylo by samozřejmě možné pokračovat dál, nicméně následoval by kmitočet přesně dvojnásobný oproti základnímu tónu, tedy bylo dosaženo celé oktávy.

Hudební tón není tvořen pouze základním kmitočtem, nýbrž též vyššími harmonickými složkami (alikvótními tóny), což jsou celistvé násobky základního kmitočtu. Tóny stejné výšky s odlišným obsahem vyšších harmonických složek se liší co do **barvy tónu**. To lze demonstrovat na rozdílném zvuku jednotlivých hudebních nástrojů. Vhodnou kombinací vyšších harmonických lze věrně napodobit libovolný hudební nástroj. Odlišná barva tónu je demonstrována na třech základních průbězích: obdélníkovém, trojúhelníkovém a pilovém.

Generace zvukových signálů pomocí FPGA působí problém, protože zvukový signál je spojitý, zatímco FPGA má pouze dvoustavové výstupy. Řešením je užití **pulzně šířkové modulace** (PWM). PWM je modulace, při které se střída výstupního obdélníkového signálu mění v závislosti na okamžité hodnotě modulovaného signálu. Velká výchylka okamžitého signálu se projeví větší střídou, menší výchylka menší střídou. Hlavním benefitem je 1bitový výstup, který může být přímo zaveden do reproduktoru, Pokud je potřeba získat původní signál stačí užít dolní propust.

PWM modulátor je v principu komparátor. Jako referenční průběh je nutné použít pilový průběh. Je nutné, aby byl kmitočet referenčního byl alespoň o řád větší než nejvyšší kmitočet modulačního signálu. Aby zařízení bezchybně fungovalo je nutné, aby měl v každém čase referenční průběh větší maximální hodnotu a menší minimální hodnotu než modulační signál. Jinak by měl výstup střídu 100 %, respektive 0 %, což není možno. Kmitočet použitého je pouhých 10 kHz, což je nicméně v této konkrétní aplikaci postačující, avšak nebylo by, pakliže by byl rozsah rozšířen směrem k vyšším tónům.

Obsah obrázku text, diagram, řada/pruh, Písmo

Popis byl vytvořen automaticky

**Obr. 1 Princip PWM modulátoru – převzato z 2**

Je pochopitelné, že modulační i referenční průběh je nutné uvnitř FPGA vyjádřit pomocí číslicových hodnot. Signál je vyjádřen vzorky, jímž je přiřazena 8bitová číselná hodnota. Protože tyto průběhy mohou nabývat též záporných hodnot, je hodnota 127 (x“7F“) užívána coby nulová úroveň, vyšší hodnoty jsou kladné, nižší záporné.

**Hardwarový popis a demo aplikace**

Popiš co umí periferie atd co ukazují.

**Softwarový popis**

Obsah obrázku text, diagram, Plán, schématické

Popis byl vytvořen automaticky

**Obr. 2 Schéma zapojení DISPLAY\_AND\_GEN\_AND\_DEMO**

Celé zařízení je možno si rozdělit do tří velkých bloků: [DISPLAY\_AND\_GEN](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/display_and_gen.vhd), [DEMO\_MAIN](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/demo_main.vhd) a [MATRIX\_CONTROL](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/matrix_control.vhd). Každý z nich obsluhuje odlišnou část zařízení. [DISPLAY\_AND\_GEN](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/display_and_gen.vhd) má podřízenou obsluhu displeje, generaci signálů, PWM modulátor a obsluhu tlačítek volume a mode. Obsahuje mimo jiné blok MAIN\_LOOP, který umožňuje přechod zařízení mezi jednotlivými módy. Blok [DEMO\_MAIN](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/demo_main.vhd) obsahuje informace o frekvenci a hlasitosti tří demo nahrávek a pomocnou logiku pro vytvoření příslušného signálu. [MATRIX\_CONTROL](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/matrix_control.vhd) má za úkol obsluhovat maticovou klávesnici a předávat ostatním blokům informaci o tom, které tlačítko bylo stisknuto. Obvod [CODER\_1\_FROM\_16](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/coder_1_from_16.vhd) je pomocný blok, který zajišťuje rozsvícení odpovídající diody, vždy při stisku klávesy. O něm budou bližší informace c souvislosti s blokem [MATRIX\_CONTROL](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/matrix_control.vhd). Všechny tyto bloky jsou zastřešeny design sourcem [DISPLAY\_AND\_GEN\_AND\_DEMO](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/display_and_gen_and_demo.vhd), který je přímo podřízen TOP\_LEVELU.

1. **Blok** [**DISPLAY\_AND\_GEN**](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/display_and_gen.vhd)

Obsah obrázku text, diagram, řada/pruh, Vykreslený graf

Popis byl vytvořen automaticky

**Obr. 3 Schéma zapojení DISPLAY\_AND\_GEN**

Blok DISPLAY\_AND\_GEN zastřešuje dva velké bloky SOUND\_GENERATOR a DISP\_CON Mimo to obsahuje logiku, která nastavuje reset komparátoru. Komparátor je resetován, pokud není nic stisknuto a zařízení nehraje demo nahrávku nebo pokud je reset komparátoru nařízen blokem DISP\_CON.

**1.1. Blok** [**SOUND\_GENERATOR**](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/sound_generator.vhd)

Úkolem SOUND\_GENERATOR je generovat PWM modulovaný signál, který obsahuje zvukovou informaci. Tento signál je buďto vygenerován uvnitř tohoto bloku nebo brán zvenčí vstupy pro dema.

Obsah obrázku diagram, text, Plán, Technický výkres

Popis byl vytvořen automaticky

**Obr. 4 Schéma zapojení SOUND\_GENERATOR**

Blok obsahuje tři generátory, které generují jednotlivé průběhy, [osmikanálový multiplexer](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/mux3bit.vhd), převodník [boot\_to\_freq](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/boot_to_freq.vhd), generátor pulzů [clock\_enable](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/clock_enable.vhd) a samotný [PWM modulátor](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/PWM_mod.vhd). [**Osmikanálový multiplexer**](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/mux3bit.vhd) umožňuje přepínání vstup PWM modulátoru mezi třemi střídavými průběhy a třemi demo nahrávkami. Je adresován signálem mode. 4bitový signál freq obsahuje informaci o frekvenci, nicméně v této podobě s ním nemohou generátory pracovat, proto je tato hodnota [**převodníkem boot\_to\_freq**](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/boot_to_freq.vhd) převedena na číselnou hodnotu, která odpovídá desetinásobku frekvence v Herzích. Protože tlačítka maticové klávesnice nejdou po sobě, bylo nutné zajistit jejich uspořádání softwarově, a to plní tento převodník. Převádí 4bitovou vstupní hodnotu na 16bitový výstup dle následující tabulky.

**Tab. 2 Převodní tabulka boot\_to\_freq**

|  |  |  |
| --- | --- | --- |
| **inpu (hex)** | **output** | **odpovídá** |
| 0 | x"0898" | A |
| 1 | x"09A5" | B |
| 2 | x"0A38" | C1 |
| 3 | x"0B78" | D1 |
| 4 | x"0E74" | F#1 |
| 5 | x"091B" | A# |
| 6 | x"0AD4" | C#1 |
| 7 | x"0C27" | D#1 |
| 8 | x"1235" | A#1 |
| 9 | x"1039" | G#1 |
| A | x"0DA4" | F1 |
| B | x"0CE0" | E1 |
| C | x"1470" | C2 |
| D | x"134B" | B1 |
| E | x"0F50" | G1 |
| F | x"1130" | A1 |

**1.1.1.** [**Osmikanálový multiplexer**](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/mux3bit.vhd)

Design source nese jméno mux3bit podle počtu bitů adresového vstupu. Multiplexer je elektronický přepínač, který v závislosti na adresových vstupech připojí k výstupu vždy pouze jeden vstup. Použitá rutina umožňuje pracovat s N-bitovými vstupy. Protože je pracováno s osmibitovým signálem, je i užitý multiplexer osmibitový. Blok realizuje funkci popsanou tabulkou:

**Tab. 3 Tabulka popisující funkci osmikanálového multiplexeru**

|  |  |
| --- | --- |
| sel | output |
| "000" | in\_0 |
| "001" | in\_1 |
| "010" | in\_2 |
| "011" | in\_3 |
| "100" | in\_4 |
| "101" | in\_5 |
| "110" | in\_6 |
| "111" | in\_7 |

**1.1.2 Blok** [**PWM\_mod**](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/PWM_mod.vhd)

Obsah obrázku text, diagram, řada/pruh, snímek obrazovky

Popis byl vytvořen automaticky

**Obr. 5 Schéma zapojení PWM\_mod**

Samotný PWM generátor se skládá z generátoru pilového průběhu a [komparátoru](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/kompareNbit.vhd). Pilový průběh vytváří bloky [CLOCK\_ENABLE](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/clock_enable.vhd) a [SIMPLE\_COUNTER](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/simple_counter.vhd). Oba bloky byly napsány a otestovány v rámci počítačového cvičení, a tedy zde nejsou podrobně popsány. Pro úplnost je pouze uvedeno, že blok [CLOCK\_ENABLE](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/clock_enable.vhd) jednou za nastavený počet cyklů pulz, který setrvá do příchodu další náběžné hrany. Blok [SIMPLE\_COUNTER](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/simple_counter.vhd) je pak vzestupný binární N-bitový čítač (v tomto případě osmibitový), který inkrementuje svůj stav vždy po příchodu hodinového signálu, je-li jeho vstup en v logické jedničce. Protože je jemu předřazený blok [CLOCK\_ENABLE](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/clock_enable.vhd), bude měnit svůj obsah vždy při příchodu pulzu z tohoto bloku. Poté, co čítač dosáhne maximální hodnoty, přeteče a začne čítat od nuly. Výsledkem je tedy periodicky se opakující nárůst číselné hodnoty, která reprezentuje pilový průběh. Protože [CLOCK\_ENABLE](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/clock_enable.vhd) generuje pulz co 40 hodinových pulzů a čítač čítá do 255 (28 ‒ 1) je frekvence pilového signálu přibližně 10 kHz.

Druhou neméně důležitou částí je komparátor, jenž je obsažen v design sourcu [kompareNbit](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/kompareNbit.vhd). Vzhledem na počet bitů modulačního signálu musí být i komparátor osmibitový. Jedná se o komparátor řízený hodinovým signálem s asynchronním resetem aktivním v nule, který je označen en. Protože zbývající bloky mají reset aktivní v jedničce, je ke společnému resetovacímu vstupu připojen přes invertor. Komparátor má dva vstupy „a“ a „b“. Jako vstup „a“ je připojen referenční pilový průběh, jako vstup „b“ modulační signál. Rutina umožňuje pracovat až se třemi výstupy: menší, větší a rovná se; nicméně pro aplikaci komparátoru coby funkčního bloku PWM modulátoru je bohatě postačující výstup jediný, konkrétně pak byl užit výstup b\_greater.

**1.1.3. Generátory tvarových průběhů**

**2. Blok** [**DEMO\_MAIN**](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/demo_main.vhd)

**3. Blok** [**MATRIX\_CONTROL**](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/matrix_control.vhd)

**3.1. Blok** [**CODER\_1\_FROM\_16**](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/coder_1_from_16.vhd)

Tento blok není přímo součástí [MATRIX\_CONTROL](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/matrix_control.vhd), nicméně svou činností s ním přímo souvisí. Jeho úkolem je zajistit, aby při stisknu libovolné klávesy byla rozsvícena právě jedna LED dioda, která odpovídá výšce hraného tónu. Jak již bylo poznamenáno, na výstupu [MATRIX\_CONTROL](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/matrix_control.vhd) nejsou číselné hodnoty seřazeny tak, aby nejvyšší číslo odpovídalo nejvyšší frekvenci. Proto je nezbytné, nejen tyto čtyřbitové číselné hodnoty seřadit, ale i převést je do kódu 1 z 16. Dále musí kodér zajistit, aby v případě, že není stisknuto žádné z tlačítek, nesvítila žádná dioda, proto také blok obsahuje vstup any, který je připojen na výstup any [MATRIX\_CONTROL](https://github.com/VojtaKudela/BPC-DE1-topic_4/blob/main/MAIN/Project_PWM.srcs/sources_1/new/matrix_control.vhd). Blok realizuje funkci popsanou tabulkou:

**Tab. 4 Převodní tabulka rutiny CODER\_1\_FROM\_16**

|  |  |  |
| --- | --- | --- |
| bcd (hex) | any | ncode (hex) |
| X | 0 | x"0000" |
| x"0" | 1 | x"8000" |
| x"1" | 1 | x"2000" |
| x"2" | 1 | x"1000" |
| x"3" | 1 | x"0400" |
| x"4" | 1 | x"0040" |
| x"5" | 1 | x"4000" |
| x"6" | 1 | x"0800" |
| x"7" | 1 | x"0200" |
| x"8" | 1 | x"0004" |
| x"9" | 1 | x"0010" |
| x"A" | 1 | x"0080" |
| x"B" | 1 | x"0100" |
| x"C" | 1 | x"0001" |
| x"D" | 1 | x"0002" |
| x"E" | 1 | x"0020" |
| x"F" | 1 | x"0008" |

**Simulace komponentů**

Write descriptive text and put simulation screenshots of your components.

**Instrukční list**

Write an instruction manual for your application, including photos and a link to a short app video.

**Reference**

1. [Online VHDL Testbench Template Generator (lapinoo.net)](https://vhdl.lapinoo.net/testbench/)
2. Analogová elektronika 2 – doc. Ing Roman Šotner, PhD.
3. [Pulsně šířková modulace (dhservis.cz)](http://www.dhservis.cz/psm.htm)
4. [František Fuka: Hudba z geekovsko-matematického hlediska - DevFest Praha 2012 (youtube.com)](https://www.youtube.com/watch?v=RQuRFCE5NzI)
5. Do you hear a people sing? – Claude-Michel Schönberg
6. Für Elise – Ludwig van Beethoven
7. Entertainer – Scott Joplin
8. [Microsoft PowerPoint - aud\_prednaska\_1 [režim kompatibility] (vut.cz)](https://moodle.vut.cz/pluginfile.php/827361/mod_resource/content/3/aud_prednaska_1.pdf)