

Bakalářská práce



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická

Jednoduché laboratorní generátory signálu s mikrořadiči STM32

Kateřina Pravdová

Vedoucí práce: doc. Ing. Jan Fischer, CSc.
Studijní program: Elektronika a komunikace
Květen 2022

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Pravdová** Jméno: **Kateřina** Osobní číslo: **495660**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra elektromagnetického pole**
Studijní program: **Elektronika a komunikace**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Jednoduché laboratorní generátory signálu s mikrořadiči STM32

Název bakalářské práce anglicky:

Simple Laboratory Signal Generators Based on STM32 Microcontroller

Pokyny pro vypracování:

Navrhněte a s využitím mikrořadičů řady STM 32 realizujte jednoduché signálové a impulsní generátory pro použití v laboratorní výuce na ČVUT- FEL.

Pro generaci spojitých signálů s prostřednictvím interních převodníků DAC využijte mikrořadiče STM32G431, případně též STM32F303. S mikrořadiči STM řady G031 a F042 realizujte generátory impulsů s využitím kanálů PWM. Ovládání a indikace nastavení generátorů bude prostřednictvím kanálu UART a terminálové PC aplikace. Pro lokální ovládání a nastavování parametrů generátorů se orientujte na využití tlačítek, enkodéru a potenciometru. Pro indikaci stavu využijte též RGB LED.

Seznam doporučené literatury:

- [1] Yiu, J.: The Definitive Guide to ARM Cortex -M0 and Cortex-M0+ processors
- [2] STMicroelectronics: RM0444 STM32G0x12 Reference manual
- [3] STMicroelectronics: DS12992 STM32G031 Data

Jméno a pracoviště vedoucí(ho) bakalářské práce:

doc. Ing. Jan Fischer, CSc. katedra měření FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **02.02.2022**

Termín odevzdání bakalářské práce: **20.05.2022**

Platnost zadání bakalářské práce: **30.09.2023**

doc. Ing. Jan Fischer, CSc.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Studentka bere na vědomí, že je povinna vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

Datum převzetí zadání

Podpis studentky

Poděkování

Ráda bych poděkovala svému vedoucímu bakalářské práce doc. Ing. Janu Fischerovi, CSc., za odborné vedení práce a mnoho cenných rad.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracovala samostatně a že jsem uvedla veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze, 20. května 2022

.....

Abstrakt

Tato práce se zabývá návrhem digitálních generátorů. Jsou navrženy funkční, PWM a sinusové generátory, které jsou implementovány na mikrokontroléry STM32G431, STM32G030, STM32F303 a STM32F042. Funkční generátor využívá princip přenosu vzorků z paměti na DA převodník. Pro dosažení vysoké přesnosti frekvence a nízkého zkreslení signálu je na něm uplatněn optimalizační algoritmus. Impulsní generátor používá časovač v PWM módu. Harmonický generátor je vytvořen pomocí PWM signálu s pulsy o proměnných střídacích, který je následně filtrován. Pro filtraci je použit RC článek 1. řádu. Jako ovládací prvky jsou použity potenciometry a tlačítka. V práci je také popsána USB a UART komunikace, pomocí kterých jsou měněny parametry generovaných signálů.

Klíčová slova: STM32, STM32G431, STM32G030, STM32F042, generátor, funkční generátor, PWM

Vedoucí práce: doc. Ing. Jan Fischer, CSc.

Abstract

This work deals with the design of digital generators. Functional, PWM and sine generators are implemented on STM32G431, STM32G030, STM32F303 and STM32F042 microcontrollers. The function generator uses the DA converter with timer and DMA. An optimization algorithm is implemented in order to achieve high frequency accuracy and low signal distortion. The pulse generator uses a timer in PWM mode. The sine generator is created by using a PWM signal with pulses of variable duty, which is then filtered. A low pass RC circuit is used for filtration. Potentiometers and buttons are used as controls. The work also describes USB and UART communications, which are used to change the parameters of the generated signals.

Keywords: STM32, STM32G431, STM32G030, STM32F042, generator, functional generator, PWM

Title translation: Simple laboratory signal generators with STM32 microcontrollers

Obsah

| | |
|--|-----------|
| 1 Úvod | 1 |
| 2 Rozbor zadání | 3 |
| 3 Funkční generátor | 5 |
| 3.1 Nastavení frekvence signálu | 6 |
| 3.2 Výpočet hodnot v tabulce vzorků pro generaci signálu | 6 |
| 3.3 Implementace optimalizačního algoritmu pro zvýšení přesnosti frekvence | 7 |
| 3.4 USB komunikace | 9 |
| 3.5 Manuální ovládání parametrů signálu tlačítky | 9 |
| 3.6 Proces bootování v mikrokontrolérech | 9 |
| 4 Interaktivní PWM generátor | 13 |
| 4.1 RGB LED | 14 |
| 4.2 Manuální ovládání parametrů signálu trimrem | 15 |
| 4.3 UART komunikace | 16 |
| 4.4 Řešení nedostatečného množství pinů | 18 |
| 5 Dvoukanálový generátor | 21 |
| 5.1 Metody generace burst signálu | 21 |
| 5.2 Použití generátoru s Dataplotterem | 23 |
| 5.3 Terminálový výstup – přepínání mezi stránkami | 24 |
| 6 Harmonický generátor | 25 |
| 6.1 Frekvence signálu Harmonického generátoru | 26 |
| 6.2 Návrh filtru typu dolní propust | 26 |
| 6.3 Zamezení šumu a tlumení signálu při použití neměnného RC filtru | 27 |
| 6.4 Změna amplitud dvou signálů pomocí trimru | 28 |
| 7 Zhodnocení dosažených výsledků práce | 29 |

| | |
|--|-----------|
| 8 Závěr | 33 |
| Literatura | 35 |
| A Zdrojové kódy jednotlivých generátorů | 37 |
| B Uživatelské manuály jednotlivých generátorů | 39 |
| B.1 Uživatelský manuál Funkčního generátoru | 39 |
| B.2 Uživatelský manuál Interaktivního PWM generátoru | 43 |
| B.3 Uživatelský manuál Dvoukanálového PWM generátoru | 47 |
| B.4 Uživatelský manuál Harmonického PWM generátoru | 53 |

Obrázky

| | |
|---|----|
| 2.1 Generace signálu pomocí DA převodníku | 3 |
| 3.1 Schéma DMA generátoru | 5 |
| 3.2 Rozdílný průběh signálu při odlišném počtu vzorků v tabulce | 7 |
| 3.3 Přibližné schéma algoritmu pro určení optimálního počtu vzorků..... | 8 |
| 3.4 Blokové schéma obsluhy přerušení | 10 |
| 4.1 Princip PWM generátoru s proměnnou střídou | 13 |
| 4.2 Repräsentace bitů v RGB LED | 15 |
| 4.3 Datový rámec RGB LED | 15 |
| 4.4 Barevný přechod v RGB LED | 16 |
| 4.5 Schéma algoritmu měnícího citlivost ADC pro ochranu před šumem | 17 |
| 4.6 Datový rámec UART komunikace | 17 |
| 5.1 Příklad burst signálu | 21 |
| 5.2 Generace PWM signálu použitím PWM bloku časovače a DMA..... | 22 |
| 5.3 Parametry burst signálu | 23 |
| 5.4 Příklad použití Dvoukanálového generátoru s Dataplotterem | 23 |
| 5.5 Stránky Dvoukanálového generátoru | 24 |
| 6.1 Generace harmonického signálu využitím PWM s modulací střídy..... | 25 |
| 6.2 Dolní propust 1. řádu | 27 |
| 6.3 Amplitudová charakteristika dolní propusti 1. řádu..... | 27 |
| 6.4 Rozdílný šum signálu o frekvenci 616 Hz na mikrokontrolérech F042 (vlevo) a G030 (vpravo) | 28 |
| 7.1 Funkční generátor | 29 |
| 7.2 Interaktivní PWM generátor | 30 |

| | |
|--------------------------------------|----|
| 7.3 Dvoukanálový PWM generátor | 31 |
| 7.4 Harmonický generátor | 31 |

Tabulky

| | |
|--|----|
| 4.1 Počet dostupných pinů a potřebných periférií pro Interaktivní PWM generátor na mikrokontroléru STM32G030J6 | 18 |
| 4.2 Alternativní funkce pinů Interaktivního PWM generátoru | 18 |
| 6.1 Možné hodnoty frekvencí signálu Harmonického generátoru na mikrokontrolérech STM32F042 a STM32G030 | 26 |



Kapitola 1

Úvod

Cílem této práce je navrhnout a realizovat několik digitálních generátorů, které by měly být použity v laboratorní výuce. Měly by sloužit k realizaci různých experimentů, k demonstraci vlastností signálů a k testování jiných laboratorních přístrojů. Jeden z generátorů by měl být navíc vhodný k použití na akcích pro děti pořádaných skupinou Witches.

Ve výuce na Katedře měření na FEL ČVUT se již využívá široká škála tzv. SDI přístrojů, což jsou jednoduché softwarově definované přístroje pro laboratorní experimenty a distanční výuku. Mezi ně patří osciloskop LEO, universální STM32 čítač, logický analyzátor ELA a další. Hlavní motivací pro vývoj těchto přístrojů je, aby měl každý student k dispozici své vlastní zařízení. V dnešní době výkonných procesorů je jednodušší dát každému studentovi procesor, na který si nahraje potřebnou funkci, než analogový obvod.

Generátory by měly být realizovány na různých mikrokontrolérech STM32. Od typu mikrokontroléru se budou odvíjet funkce generátorů tak, aby byl naplno využit jejich potenciál. Generátory by měly být levné a lehce sestavitelné. Proto budou vytvořeny buď pomocí mikrořadiče STM32 na nepájivém poli, nebo přímo na vývojové desce. Pro každý generátor by měl být vytvořen uživatelský manuál, aby ho mohl kdokoli používat bez speciální instruktáže.

Jedním z možných použití generátoru jsou testovací signály pro experimenty nebo pro logický analyzátor. Také by mělo být možné generátorem impulsních burst signálů testovat stroboskopické vzorkování osciloskopu. Dalším možným uplatněním by mohla být například demonstrace změny amplitudy sinusového signálu, který je generován pomocí pulsně-šířkové modulace, pouhou změnou střídy PWM signálu.

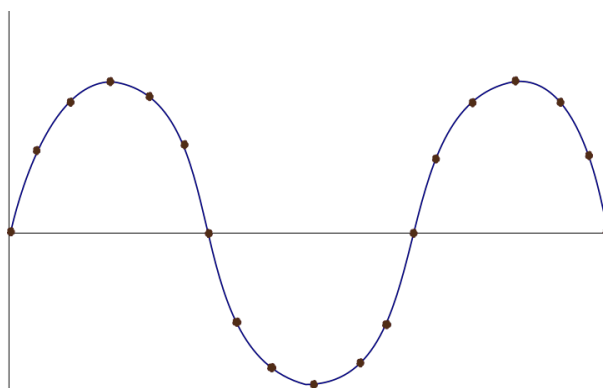
Kapitola 2

Rozbor zadání

Zadáním této práce je návrh a realizace několika jednoduchých signálových a impulsních generátorů na mikrořadičích STM32. Hlavním požadavkem je spolehlivost a jednoduché ovládání, které je důležité, aby bylo možné generátory použít při laboratorní výuce nebo při promo akcích pořádaných fakultou.

Úkolem je implementovat generátory na mikrokontroléry STM řady G431, G030 a F042, které se používají na Katedře měření. Jeden z generátorů by také měl být implementován na Nucleo STM32F303. Nucleo STM32F303 je totiž považováno za hlavní SDI přístroj a všechny funkce, které se implementují na jiné mikrokontroléry, by měly být k dispozici i pro F303. Mikrokontroléry STM32G431 a F303 jsou zástupci zařízení se dvěma DA převodníky.

Signál, který je generován pomocí DA převodníku, je složen z diskretních vzorků (viz obr. 2.1), které jsou na něj jeden po druhém zasílány. Důležitým parametrem je počet vzorků na periodu. Kvůli některým omezujícím faktorům, jako je např. maximální hodinová frekvence, na které mikrokontrolér pracuje, musí být signály o vyšších frekvencích složeny z menšího počtu vzorků. To s sebou přináší několik problémů, kterými je potřeba se zabývat.



obr 2.1: Generace signálu pomocí DA převodníku

Zařízení s DA převodníky bývají ovšem dražší a mívají větší pouzdra. Pro mnoho aplikací navíc není nutné generování libovolných signálů a častokrát je postačující obdélníkový signál. Pro generaci obdélníkového signálu je možné použít čítač v PWM módu, který má k dispozici i většina menších mikrokontrolérů. Výsledný impulsní generátor na malém čipu by potom mělo být možné vložit do obvodu a použít jako časovač, stejně jako oblíbený integrovaný obvod NE555. Časovač vytvořený pomocí mikrokontroléru má však podstatně více možností.

V některých případech je vhodné mít k dispozici kromě obdélníkového signálu ještě alespoň

sinusový signál. Ten je možné generovat pomocí PWM signálu filtrovaného dolní propustí. Jedním filtrem ovšem není možné generovat signály v širokém rozsahu frekvencí. Tento rozsah je možné některými způsoby zvětšit, nicméně možnosti jeho rozšíření mohou být limitovány např. velikostí paměti mikrokontroléru.

Z výše uvedeného vyplývá, že je vhodné vytvořit následující generátory:

- Funkční generátor sinusového, trojúhelníkového, obdélníkového a rampového signálu
- Interaktivní generátor obdélníkového signálu
- Dvoukanálový generátor obdélníkového signálu
- Sinusový generátor s filtrem

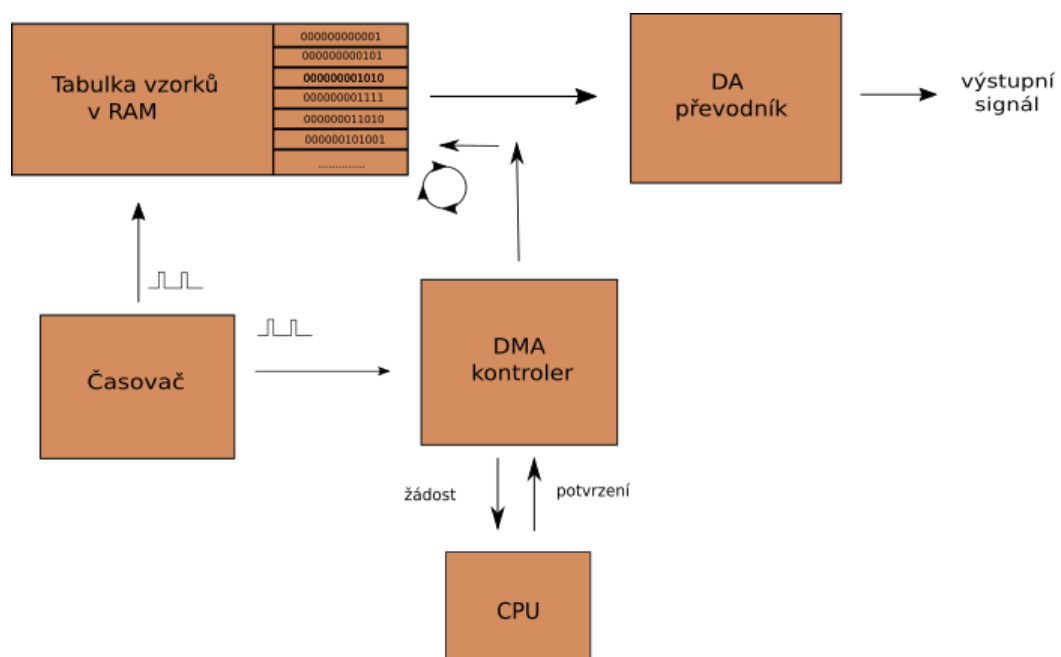
Aby bylo možné modifikovat parametry generovaných signálů, je potřeba, aby se dalo s mikrokontroléry komunikovat buď přes počítač, nebo manuálně. Některé mikrokontroléry mají USB komunikační rozhraní, některé disponují pouze rozhraním USART. Pro manuální změnu je možné použít buď tlačítka, nebo trimery. V některých případech má pouzdro mikrokontroléru dostatečný počet pinů pro kombinaci manuálního ovládání a ovládání přes PC, v některých je potřeba řešit nedostatek pinů různými způsoby.

Kapitola 3

Funkční generátor

Nejlepší digitální funkční generátory využívají přímou digitální syntézu (DDS). Takové generátory dokáží generovat signály s vysokou přesností frekvence a zároveň mají dobrou stabilitu. Na druhou stranu potřebují velké množství výpočetního výkonu. Další nevýhodou je, že pro implementaci DDS generátoru je kromě čipu potřeba ještě dodatečný hardware [1]. Pokud chceme vytvořit generátor pouze na jediném čipu, je potřeba najít jinou metodu.

Jednodušší variantou oproti DDS je použití analogově digitálního převodníku, tabulky vzorků a časovače. Tuto metodu je navíc možné udělat ještě šetrnější ke spotřebě výpočetního výkonu použitím přímého přístupu do paměti (DMA). Tabulku vzorků je možné uložit do flash paměti nebo do RAM. Flash paměť poskytuje prostor pro uložení většího množství vzorků než RAM. Nicméně přenášení vzorků z RAM na DA převodník bude trvat kratší dobu než přenos z flash paměti.



obr 3.1: Schéma DMA generátoru

Generátor založený na přímém přístupu do paměti (DMA) se skládá z řídicího mikroprocesoru s DMA kontrolérem, časovače a digitálně-analogového převodníku [4]. Do datové paměti je uložena tabulka hodnot vzorků jedné periody signálu, který má být generován. DMA kontrolér pošle centrální

procesorové jednotce žádost o použití sběrnice a čeká na potvrzení. Po potvrzení jsou číselné hodnoty v časových intervalech, který určuje časovač, přímo přenášeny na DAC viz obr. 3.1. V případě periodického signálu se adresy vzorků cyklicky opakují.

3.1 Nastavení frekvence signálu

Výstupní frekvenci DMA generátoru určuje několik faktorů. Mezi tyto faktory patří frekvence časovače (f_{tim}), počet vzorků signálu v tabulce (n_{vzr}) a hodnota auto-reload registru (ARR). Výsledná frekvence je potom

$$[h]f_{sig} = \frac{f_{tim}}{ARR \cdot n_{vzr}}. \quad (3.1)$$

V mikrořadičích, pro které by měl být Funkční generátor implementován, je 16 různých časovačů [2]. Časovač TIM 2 je 32bit časovač obecného účelu, pro který je možné nastavit systémové hodiny, prescaler a auto-reload čítač. Zdrojem systémových hodin může být vysokorychlostní externí oscilátor pracující na frekvencích od 4 do 48 MHz nebo interní RC oscilátor, přičemž oba mohou být použity s fázovým závěsem. RC oscilátor má jednu určitou frekvenci a pomocí fázového závěsu je možné nastavit až 476 různých frekvencí. Prescaler je 32bitový čítací obvod, jenž se nachází za oscilátorem. Prescaler čítá kmity oscilátoru do nastavené hodnoty a tím dělí jeho frekvenci požadovanou hodnotou. Frekvenci časovače tedy určuje frekvence oscilátoru, fázový závěs a prescaler.

Použitý časovač má dále k dispozici čítač periody. Ten čítá cykly časovače a v momentě, kdy dojde do nastavené hodnoty, se spustí daná událost – přenos vzorků z tabulky do DA převodníku. Hodnota čítání se nastavuje pomocí auto-reload registru, který má, stejně jako prescaler, 32 bitů [3].

Hodnoty prescaleru i ARR jsou jediné celočíselné, přičemž prescaler může být v rozmezí 1 až 65 536 a hodnota ARR mezi 1 a 4 294 967 295. Pokud je frekvence systémových hodin 144 MHz, perioda přenášení vzorků pomocí DMA může být až 500 hodin. Minimální frekvence je tedy pro reálný případ neomezená. Prakticky je omezená celočíselným datovým typem, ve kterém je uložena, na 1 Hz. Maximální frekvence je odvozena v kapitole 3.3.

3.2 Výpočet hodnot v tabulce vzorků pro generaci signálu

Funkční generátor by měl generovat harmonický, trojúhelníkový, rampový a obdélníkový signál. Pro generaci harmonického signálu je nutné použít funkci sinus [14]. K hodnotě sinu se přičte 1, protože je potřeba, aby byly hodnoty kladné. Při využití 12bitového digitálně-analogového převodníku je maximální hodnota, kterou do něj lze zapsat, 4096. Maximální hodnota signálu bude tedy 4096/2 a minimální 0. Hodnoty vzorků harmonického signálu s nejvyšší možnou amplitudou a s nulovým offsetem se tedy vypočítají podle vzorce:

$$v[i] = \frac{4096}{2} \left(\sin\left(\frac{2 \cdot \pi \cdot i}{n}\right) + 1 \right), \quad (3.2)$$

kde $v[i]$ je hodnota i -tého vzorku a n je celkový počet vzorků. Maximální hodnota vzorků odpovídající nejvyšší možné hodnotě DA převodníku se rovná napájecímu napětí. Amplituda harmonického signálu při napájecím napětí 3,3 V (které je používané nejčastěji) je tedy 1,65 V.

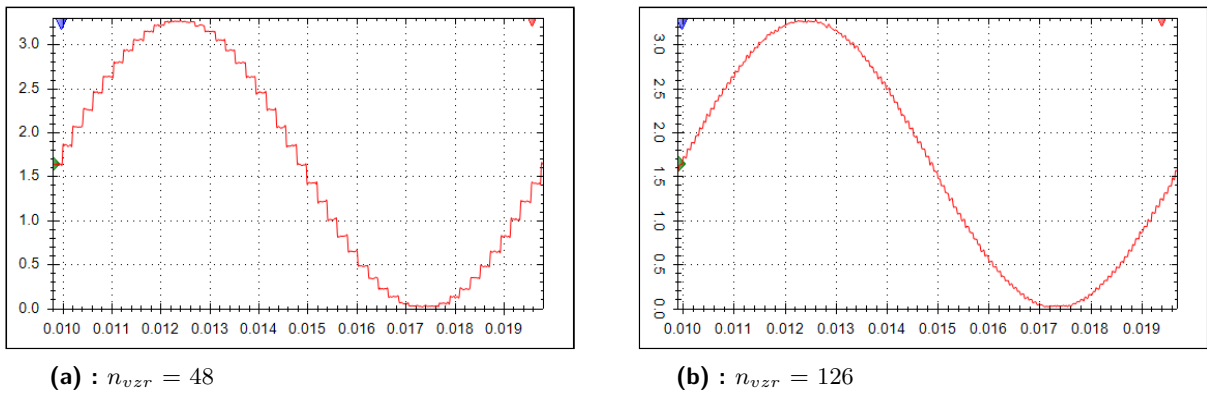
Pro trojúhelníkový signál platí:

$$v[i] = 4096 \cdot \left(1 - \frac{1}{2} \cdot \left|i - \frac{n_{vzr}}{2}\right|\right). \quad (3.3)$$

Pro rampový signál platí:

$$v[i] = \begin{cases} 0, & (0, \frac{n_{vzr}}{4}) \\ \frac{2048 \cdot 4}{n_{vzr}} \cdot (i - \frac{n_{vzr}}{4}), & i \in (\frac{n_{vzr}}{4}, \frac{n_{vzr}}{2}) \\ 2048, & (\frac{n_{vzr}}{2}, \frac{3 \cdot n_{vzr}}{4}) \\ 2048 - \frac{2048 \cdot 4}{n_{vzr}} \cdot (i - \frac{3 \cdot n_{vzr}}{4}), & i \in (\frac{3 \cdot n_{vzr}}{4}, n_{vzr}). \end{cases} \quad (3.4)$$

Čím delší je tabulka vzorků, tím má daný signál menší zkreslení viz obr. 3.2.



obr 3.2: Rozdílný průběh signálu při odlišném počtu vzorků v tabulce

Pro přehlednost ovládání bylo určeno, že amplituda U_m i offset O_{ff} budou mít hodnoty 1 až 100. Pro změnu amplitudy je potřeba vynásobit každý vzorek hodnotou $\frac{U_m}{100}$ u rampového signálu, nebo hodnotou $\frac{U_m}{100}$ u ostatních signálů. Pro změnu offsetu je nutné přičíst ke každému vzorku hodnotou $\frac{O_{ff}}{100}$.

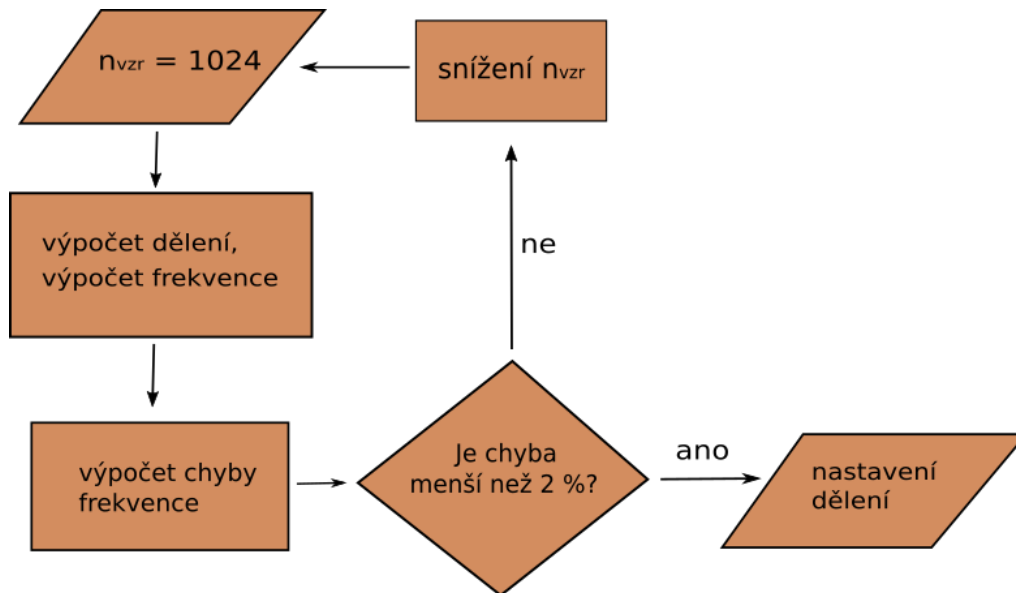
3.3 Implementace optimalizačního algoritmu pro zvýšení přesnosti frekvence

Generátorem, který je založený na přímém přenosu vzorků z paměti na DAC, nelze vygenerovat libovolnou frekvenci. To je způsobeno tím, že hodnota prescaleru i ARR má konečné rozlišení. V případě mikrokontrolérů STM32G431 jsou tyto hodnoty dokonce celočíselné. Frekvence je navíc podle vzorce 3.1 dělená počtem vzorků v tabulce, což ještě více snižuje její přesnost. Je-li f_{exp} očekávaná frekvence, pak relativní chyba frekvence je

$$e = \frac{\left|f_{exp} - \frac{f_{osc}}{n_{psc} \cdot ARR}\right|}{f_{exp}}. \quad (3.5)$$

Zkrácení tabulky sice zvýší přesnost frekvence, ale zároveň způsobí větší zkreslení signálu. Je nutné najít mezi těmito dvěma faktory rovnováhu a to vede k použití optimalizačního algoritmu. Přibližně

schéma algoritmu je na obrázku 3.3. Podobný algoritmus prezentoval Jiří Hladík na POSTEL 2017 [4].



obr 3.3: Přibližné schéma algoritmu pro určení optimálního počtu vzorků

Při určování počtu vzorků je potřeba také myslet na reálné parametry mikrokontroléru. Mezi zásadní parametry pro generaci signálu pomocí DAC patří za prvé omezená frekvence vysokorychlostní sběrnice (AHB), která poskytuje hodinový signál digitálně-analogovém převodníku. Za druhé je podstatná doba přenosu vzorků z paměti na DAC [14]. A za třetí sem patří počet převodů, který je za jednu sekundu schopný udělat DA převodník.

Frekvence AHB mikrokontrolérů STM32G431 je až 170 MHz. Pokud by měl signál délku 16 vzorků, což je minimální délka, pro kterou je průběh signálu rozpoznatelný, mohl by dosáhnout frekvence až 10,6 MHz.

Daný mikrokontrolér má dva DA převodníky, které mají dohromady čtyři kanály. Dva kanály mají rychlost převodu 15 MS/s a dají se použít pro periferie na čipu, avšak druhé dva kanály, které je možné připojit i k externímu pinu, mají rychlost převodu pouze 1 MS/s. Z rychlosti převodu DA převodníku (v_{DAC}) vyplývá následující podmínka pro všechny generované signály:

$$f_{sig} \cdot n_{vzr} \leq v_{DAC} \quad (3.6)$$

O tuto podmínku by měl být doplněn algoritmus na obrázku 3.3. Pokud ovšem nedochází v generovaném signálu k rychlým skokovým změnám, nebude mít nedodržení 3.6 pro $v_{DAC} = 1$ MS/s přílišný vliv na výsledný průběh signálu. Z měření vyplývá, že součin $f_{sig} \cdot n_{vzr}$ pro signály generované Funkčním generátorem může být maximálně 4,8 MS/s. Signál délky 16 vzorků by pro splnění nerovnice mohl mít frekvenci nejvýše 300 000 Hz. Tuto frekvenci tedy lze považovat za maximální možnou frekvenci generátoru.

3.4 USB komunikace

Kromě komponent na obrázku 3.1 musí mít mikrořadič, na který má být Funkční generátor implementován, k dispozici ještě komunikační rozhraní, jímž ho bude možné propojit s počítačem. Díky tomu bude možné přijímat data o požadovaných parametrech a vysílat informace o aktuálních hodnotách parametrů do počítače. Mikrořadič STM32G431KB, který jsem pro implementaci použila, disponují několika komunikačními periferiemi – FD CAN, I2C, USART, SPI, USB 2.0 full-speed, IRTIM [2].

Z komunikačních rozhraní, která nabízí použitý mikrořadič, je pro implementaci Funkčního generátoru nejvýhodnější USB. Kromě toho, že je nejrychlejší, je to navíc nejrozšířenější standardizovaná sběrnice a je tedy možné ji bez jakýchkoliv převodníků připojit ke každému počítači. Vysoká kompatibilita USB bude podle trendů současného vývoje platit i v budoucnu. Velká část zařízení využívajících napájecí napětí do 5 V může být navíc přes USB i napájena.

USB komunikace je typu single master a probíhá prostřednictvím 8bytových paketů [7]. Komunikace je vedena v rámci trvajících přesně 1 ms. Nevýhoda implementace této komunikace je v tom, že je poměrně složitá, a proto se většinou používají již vytvořené knihovny. Full speed USB vyžaduje přesný 48MHz oscilátor, který u použitého mikrořadiče zajišťuje buď hlavní fázový závěs, externí oscilátor, nebo interní 48MHz oscilátor s automatickým přizpůsobením [2].

USB používá dva páry vodičů. První pár slouží k napájení a druhé dva vodiče přenášejí diferenciální signál DATA+ a DATA-. DATA+ je potřeba připojit k zařízení přes pull-up rezistor o hodnotě 1,5 kΩ. V poslední řadě je nutné pro připojení k PC použít USB konektor.

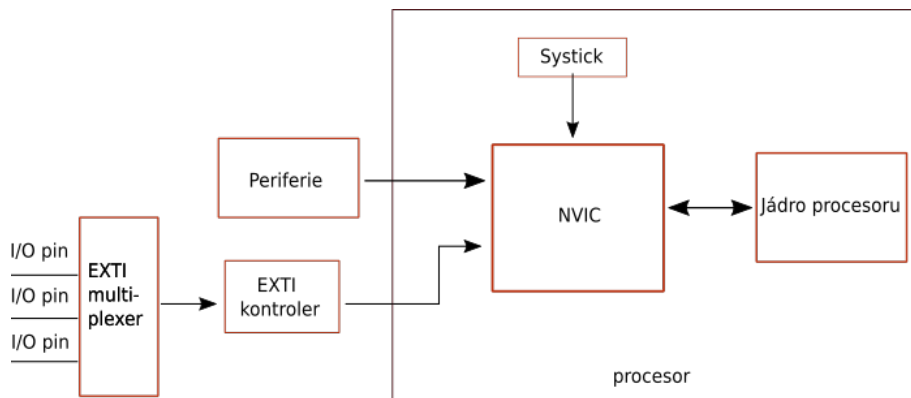
Při implementaci Funkčního generátoru na mikrokontroléry STM324G31RB a STM32F303RE není výhodné použití USB komunikace. Tyto mikrokontroléry se totiž nacházejí na vývojových deskách, na kterých je zároveň k dispozici UART-USB převodník. V tomto případě je tedy uživatelsky přívětivější použít UART komunikaci, která je popsána v kapitole 4.3.

3.5 Manuální ovládání parametrů signálu tlačítka

V některých situacích uživatel nemá k dispozici počítač s terminálovou aplikací. Proto je užitečné, aby měl generátor k dispozici tlačítka na skokové snižování a zvyšování frekvence signálu, případně na změnu průběhu signálu. Přítomnost tlačítek může být výhodná, např. pokud není k dispozici počítač, ze kterého by mohl být generátor ovládán. Vyrobený generátor na STM32G431KB má k dispozici dvě tlačítka. Obsluha tlačítek se nejčastěji implementuje pomocí přerušení. Obsluhu přerušení zajišťuje NVIC kontrolér. Mikrokontroléry STM32G431 mají k dispozici 71 maskovatelných kanálů přerušení a 17 přerušovacích linek Cortex-M4 procesoru [2]. Mají také rozšířený ovladač přerušení (EXTI) obsahující 39 detektorů hran. Právě periferie EXTI se používá pro detekci náběžných hran na pinech. Přepínání mezi jednotlivými piny zprostředkovává EXTI multiplexor. Schéma obsluhy přerušení je na obr 3.4.

3.6 Proces bootování v mikrokontrolérech

Pro práci s mikrokontroléry je dobré vědět, co se děje ihned po jejich resetu neboli co se děje před tím, než se na nich začnou vykonávat námi požadované instrukce. Pokud navíc požadujeme, aby měl mikrokontrolér dva módy, ve kterých bude pracovat, jsou tyto informace nezbytné. Také je nutné znát způsoby, kterými je možné do mikrokontroléru nahrát program.



obr 3.4: Blokové schéma obsluhy přerušení

Reset mikrokontroléru se aktivuje automaticky po jeho připojení k napájecímu napětí. Také je možné ho spustit krátkým připojením reset pinu k zemi. Toto jsou dva nejpoužívanější způsoby resetu, nicméně reset je také možné spustit uživatelsky instrukcí, která skočí na adresu resetu [15]. Předtím je však nutné vypnout hodiny periferií, fázové závěsy a přerušení. Při přerušení podmínky a současném hardwarovém resetu nebo pomocí příkazu Go to je možné z boot módu vystoupit.

Mikrokontroléry STM32 mají dva typy paměti – flash a statickou (SRAM) paměť. Pro uložení jednoho bitu ve flash paměti je potřeba méně místa než pro uložení jednoho bitu v SRAM. Proto bývá flash paměť větší než SRAM, na druhou stranu je flash paměť ze své podstaty pomalejší nežli SRAM. Zásadní je, že flash paměť je energeticky nezávislá a informace v ní zůstávají i po odpojení napájení, zatímco SRAM je energeticky závislá a při každém spuštění se do ní musí data znovu nahrát. SRAM dokáže udržet obsah i při napětí kolem 2 V. Po odpojení napájení se obsah z SRAM díky pomalému poklesu náboje na blokovacích kondenzátorech vymaže až po několika sekundách. Reset mikrokontroléru se tedy nemusí podařit, pokud mikrokontrolér odpojíme od napájení a ihned ho opět připojíme. Toto je velmi důležité pro přepínání módů Interaktivního PWM generátoru.

Těsně po resetu mikrokontroléru se začnou postupně vykonávat tyto kroky: mikrokontrolér si vezme první instrukci z první (nulté) adresy. Na té je uložena adresa vrcholu zásobníku. Na druhé adrese je ukazatel na začátek instrukcí, které obsluhují reset. Tato adresa je uložena do program counteru [9]. Program counter vždy ukazuje na následující provedenou instrukci, začnou se tedy vykonávat instrukce resetu. Nejprve se znepřístupní přerušení, aby se nemohla během resetu spouštět, a proměnné z .data sekce, což jsou hlavně globální proměnné, se přesunou z flash paměti do paměti RAM. Dále se rezervuje .bss segment pro neinicilizované proměnné, inicializuje se stack segment podle instrukcí vzniklých v linku pre-kompilací [10]. Nakonec se znovu povolí přerušení a zavolá se funkce main. Prostředí, ve kterém kódy tvoříme, většinou samy startup soubory se všemi těmito instrukcemi vytvoří.

Jak již bylo řečeno, adresa reset instrukcí je uložena do program counteru. Tato adresa může být buď ve hlavním bloku flash paměti, ve vestavěné SRAM, nebo v systémové paměti. Pro výběr módů slouží pin BOOT0 nebo nBOOT0 bit a nBOOT1 ve FLASH_OPTR registru [11]. STM32G030 mají tento proces podobný s tou výjimkou, že se nastavuje ještě BOOT_SEL bit.

Pro nahrání nového programu do flash paměti lze využít buď In-circuit programming, nebo In-application programming. In-circuit programming používá JTAG nebo SWD a přeprogramuje celou flash paměť. In-application programming mění obsah jen části flash paměti, a to za běhu programu. Pro naprogramování může použít libovolnou komunikační periferii, kterou má mikrokontrolér k dispozici. SWD je sériový synchronní debug, který používá dva piny: SWDIO, po kterém se posílají

data, a SWCLK, který zajišťuje synchronizaci. Stejně jako JTAG používá protokol JTAG. JTAG je oproti němu zprostředkován po pěti pinech.

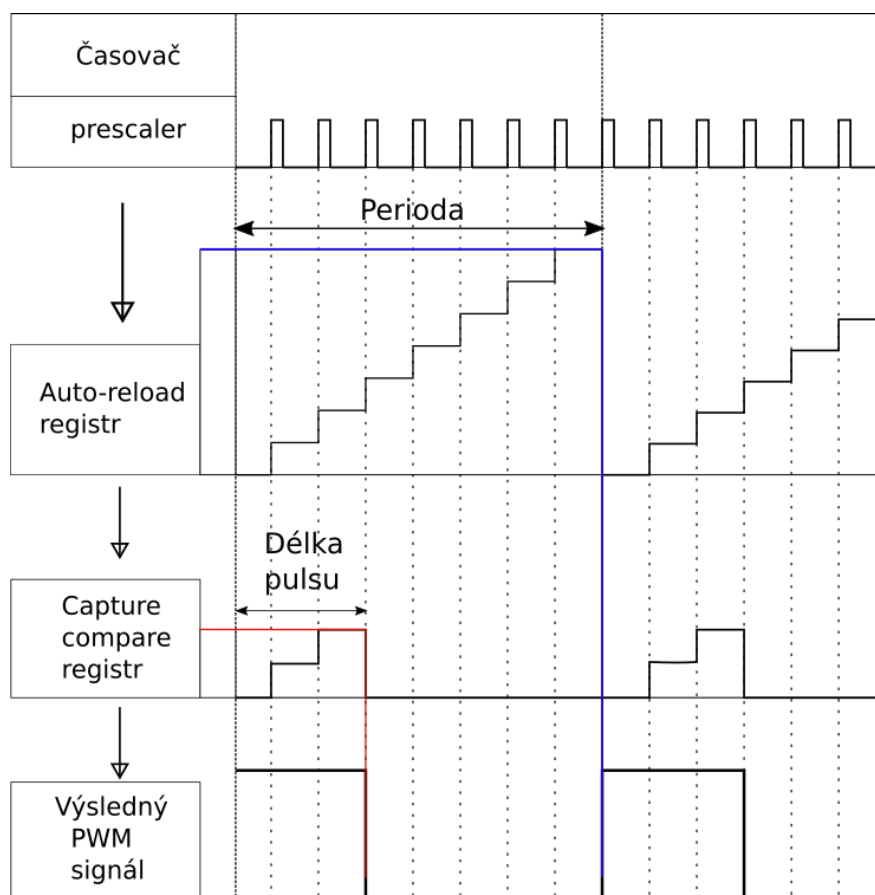
Nahrávání programu pomocí komunikačních periférií je možné spustit tím, že na boot pinu nastavíme logickou 1. Mechanismus nahrávání kódu pomocí komunikačních periférií je následující [3]. Nejprve se nastaví vysokorychlostní interní oscilátor (HSI) s fázovým závěsem na 72 MHz. Následně proběhne inicializace systému a konfigurace daných periférií. Poté kontroluje jednotlivé piny a čeká na příkaz značící počátek některé komunikace. V momentě, kdy tento příkaz přijde, se vypnou všechny zdroje přerušení a hodiny nepoužívaných rozhraní. V ten moment začíná čtení kódu přes vybranou periférii a následný zápis.

Kapitola 4

Interaktivní PWM generátor

Generátory signálů pulsně šířkové modulace se dají pomocí mikrokontroléru vytvořit několika způsoby. Jedním ze způsobů je například použití DA převodníku. Tato metoda umožňuje nastavit amplitudu signálu. Na druhou stranu není kvůli šumu DA převodníku hladina konstantní a pro skokově měnící se signál je omezující také doba převodu, která je většinou v řádu několika μs . Proto se tato metoda pro generaci PWM signálu téměř nepoužívá.

Druhá, a zároveň nejvíce přímočará metoda, která je k dispozici, je použití čítače, který umožňuje práci v PWM módu. Vzhledem k tomu, že je pulsně šířkové modulace velmi často používaná například



obr 4.1: Princip PWM generátoru s proměnnou střídou

k ovládnání motorů, je tento mód u většiny mikrokontrolérů k dispozici. Schéma této metody je na obrázku 4.1. Je z něj zřejmé, že hodinový signál je dělen prescalerem. Kmity vyděleného signálu čítá čítač periody do hodnoty nastavené v ARR. Pokud se hodnota v ARR rovná hodnotě v čítači periody, je pin, který je nastavený jako výstupní pin některého z kanálů daného časovače, nastaven na logickou 1. Pokud se hodnota v čítači periody rovná hodnotě v capture-compare registru (CCR), je naopak výstupní hodnota nastavena do logické 0 [8].

Frekvence signálu (f_{PWM}) se vypočítá následovně:

$$f_{PWM} = \frac{f_{osc}}{ARR \cdot PSC}. \quad (4.1)$$

A pro střidu dle manuálu [8] platí:

$$D = \frac{CCR}{ARR} \cdot 100[\%]. \quad (4.2)$$

Maximální frekvence oscilátoru je 64 MHz. Pokud chceme, aby generátor produkoval 100 různých hodnot střidy, ARR musí mít hodnotu 100 a maximální frekvence je tedy 640 kHz.

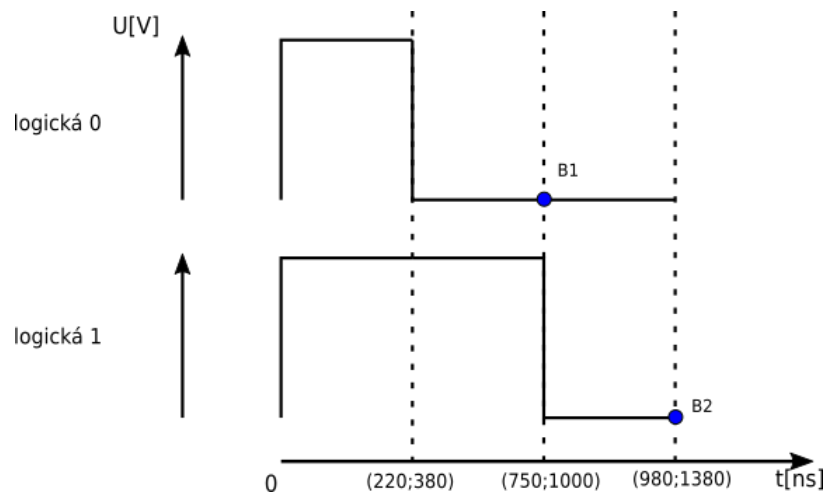
Aby mohl být tento postup implementován, je potřeba mít k dispozici pin, jehož alternativní funkcí je výstup kanálu časovače. Pokud takový pin není k dispozici, je nutné aplikovat jinou metodu. Nabízí se generovat PWM na GPIO pinu pomocí přerušení. Časovač se nastaví do módu PWM bez výstupu a po uplynutí periody se vykoná přerušení a změní se hodnota na GPIO pinu. Tato metoda ovšem významně zatěžuje CPU, a proto nebyla pro implementaci Interaktivního generátoru použita.

4.1 RGB LED

RGB LED slouží k různobarevné indikaci. Na trhu jsou k dispozici analogové LED se společnými elektrodami, kterými mohou být buď katody, nebo anody jednotlivých LED. V takovém případě má LED čtyři piny, přičemž jeden pin je připojen na zem/napájení a dalšími třemi piny se ovládají jednotlivé barvy. Pokud bychom chtěli využít celé spektrum dané LED, museli bychom mít volné tři piny mikrokontroléru. V případě nedostatku pinů je výhodnější použití digitální RGB, kterou lze ovládat pouze jedním pinem. Na něj se posílají bitové sekvence, které určují, v jaké intenzitě mají být barvy ve výsledném světle obsaženy.

U konkrétní použité RGB LED WS 2812 je bit logická 1 reprezentovaný pulsem o velikosti 1,6 – 3 V o délce 750 ns – 1 μ s, a ten musí následovat nulové napětí o délce 220 ns – 380 μ s [12]. Logická nula je přesně naopak – tedy kladné napětí bude trvat 750 ns – 1 μ s a nulové 220 ns – 380 μ s. Celková délka periody jednoho bitu bude tedy přibližně 1 μ s. V praktické implementaci není nezbytné přesně dodržovat dané hodnoty, jsou důležité jen body, které jsou na obrázku 4.2 vyznačeny modře. Bit nula musí být v čase 750 ns – 1 μ s nulový a jedničkový bit nesmí kladným napětím zasahovat do reset části nebo do následujícího bitu. Posloupnost bitů jednoho datového rámce je na obrázku 4.3, má 24 bitů a musí být následováno reset částí – nulovým napětím o délce alespoň 280 μ s.

V Interaktivním PWM generátoru je digitální RGB LED použita pro indikaci frekvence výstupního signálu, která je ovládaná trimrem. Pro praktickou implementaci je použit PWM výstup časovače. Nejprve je nastavena perioda bitů pomocí auto-reload registru. Potom je hodnota z trimru, která je v rozmezí 0–4096, přepočítána na barevné hodnoty RGB 0–255. Cílem je, aby barva byla s rostoucí



obr 4.2: Reprezentace bitů v RGB LED



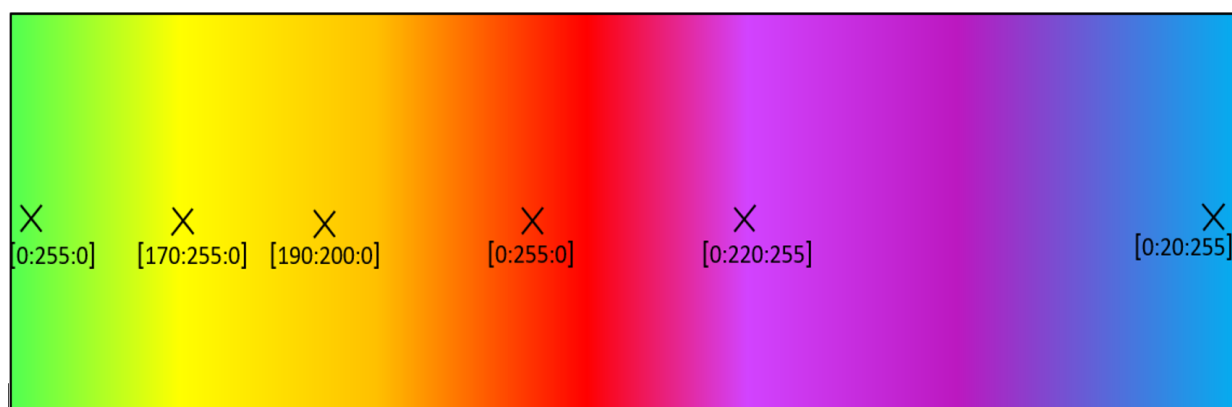
obr 4.3: Datový rámec RGB LED

frekvencí více intenzivní. Toho je dosaženo, jak je vidět na obrázku 4.4, pomocí psychologie barev. Proto nejnižší frekvenci odpovídá světle zelená barva, která obecně symbolizuje klid. Zelená potom přechází přes harmonickou žlutou a výstražnou oranžovou barvu do červené, která značí nebezpečí. Následuje fialová barva, která bývá spojována s tajemstvím, a nejvyšší frekvenci symbolizuje modrá, což je barva evokující v člověku pocit hloubky. Pro vytvoření rovnoměrného intervalu z těchto barev není možné použít hodnoty, které odpovídají virtuálnímu (softwarovému) míchání barev. Důvodem je to, že červené světlo z LED má největší intenzitu, a pokud je smíchané 1 : 1 s jiným světlem, barvu tohoto světla zakrývá. Proto bylo potřeba nalézt jednotlivé barvy experimentálně a pro přechody mezi nimi potom najít lineární funkce.

Nakonec je potřeba odeslat informaci o konkrétní barvě. Poté, co jsou hodnoty R, G, B uloženy do pole v binárním tvaru, jsou přepočítány na informace reprezentující počet kmitů oscilátoru, po který musí být vysílány jednotlivé pulsy, aby reprezentovaly buď logickou 1, nebo 0. Tyto informace jsou následně pomocí DMA přeneseny do časovače.

4.2 Manuální ovládání parametrů signálu trimrem

Manuální ovládání je výhodné v případě, kdy není k dispozici počítač pro nastavení parametrů generátoru nebo kdy jsou potřeba jenom menší změny parametrů. Manuální zásah je jednodušší než ovládání přes UART, pro které je nutné mikrokontrolér připojit k počítači a zapnout a nastavit terminál. Tento způsob ovládání by mohl být zprostředkován pomocí trimrů nebo enkodérů. V tomto případě byly použity dva trimry, protože jsou obecně dostupnější. S jejich pomocí je možné plynule měnit frekvenci a střídu generovaného PWM signálu. Střídu lze nastavit na 0 až 100 % a frekvenci lze měnit v intervalu od 100 do 1000 Hz. Výstup z trimru je nutné zpracovat analogově-digitálním převodníkem.



obr 4.4: Barevný přechod v RGB LED

Mikrokontroléry STM32G030 disponují jedním 12bitovým AD převodníkem, jehož přesnost může být v případě potřeby rozšířena až na 16 bitů [5]. Tento AD převodník má 16 externích a 3 interní kanály, nicméně použití externích kanálů je omezené počtem pinů. Převod může být proveden jednorázově nebo kontinuálně. V případě ovládání parametrů generátoru je potřeba kontinuální mód. Ten je navíc možné doplnit o cyklický DMA přenos, čímž se minimalizuje zatížení CPU.

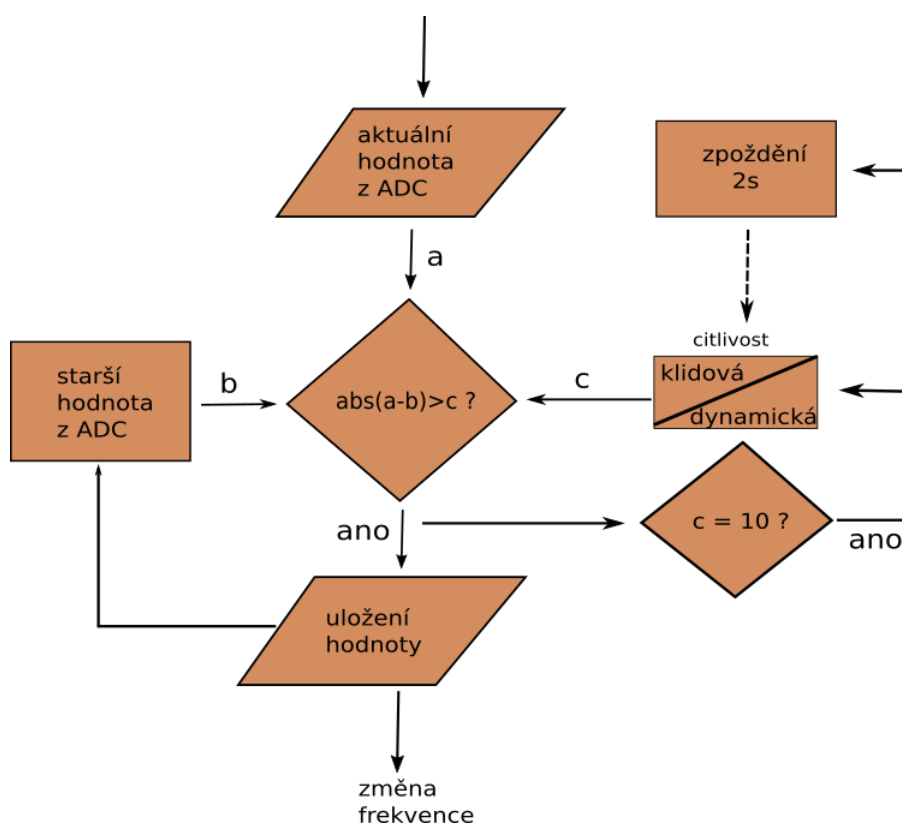
Je potřeba se vyrovnat s chybou, kterou způsobuje šum AD převodníku. Tato chyba se projevuje tak, že při konstantní hodnotě na vstupu ADC pinu mikrokontroléru a kontinuálním konverzním módu se výstupní hodnota na AD převodníku neustále mění o ± 20 (zjištěno experimentálně). Tento interval představuje při napájecím napětí 3,3 V a při použití 12 bit AD převodníku napěťový interval přibližně ± 16 mV. Pokud by se při každé takto malé změně napětí měnila frekvence a střída signálu, jevil by se signál velmi nestabilní. Proto je potřeba snížit citlivost AD převodníku v době, kdy není s trimrem pohybováno. To je znázorněno na algoritmu 4.5.

Pokud s trimrem právě není pohybováno, hodnota citlivosti je "klidová". Každá nová výstupní hodnota z AD převodníku je porovnávána s předchozí uloženou hodnotou. Pokud pohneme s trimrem, změní se výstupní hodnota z AD převodníku o hodnotu větší než je klidová citlivost, což způsobí změnu citlivosti porovnávání na nižší – "dynamickou" – citlivost, a zároveň se spustí časovač, který odpočítává 2 sekundy. Než tyto dvě sekundy uběhnou, je s každou novou výstupní hodnotou z AD převodníku měněna frekvence. Poté, co časovač dopočte, je opět citlivost vrácena na klidovou hodnotu.

4.3 UART komunikace

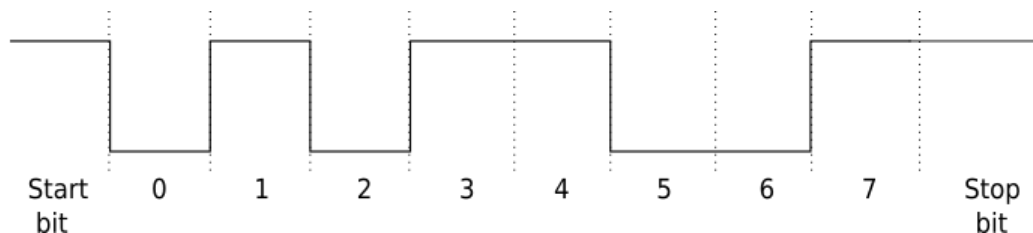
Universální asynchronní obousměrná komunikace neboli UART je plně duplexní komunikace, která používá pro každý směr dat jeden vodič – Rx a Tx. Další používané signály jsou RTS a CTS. Signál RTS představuje dotaz vysílací strany na to, zda má přijímací strana data kam uložit. Odpovědí na RTS signál je CTS signál, který oznámí, že je možné odeslat data, nebo komunikaci pozdrží, dokud nebude příjem připraven. Pokud požadujeme kontinuální přenos bez zastavení, je možné lokálně propojit signál RTS se signálem CTS. V takovém případě bude komunikace probíhat pouze po dvou vodičích – Rx a Tx.

UART přenos probíhá v 7-, 8-, nebo 9bitových rámcích, kde poslední bit je volitelný paritní



obr 4.5: Schéma algoritmu měnícího citlivost ADC pro ochranu před šumem

bit. Pokud právě neprobíhá komunikace, je komunikační signál v úrovni logická 1. V momentě, kdy chce jedno zařízení začít komunikovat, odešle tzv. start bit, což znamená, že stáhne napětí signálu na logickou 0. Od tohoto momentu očekává přijímací zařízení určený počet bitů (typicky 8) v pravidelných intervalech, za nimiž následuje stop bit značící konec komunikace [13]. Schéma popsané komunikace kanálem UART je na obrázku 4.6.



obr 4.6: Datový rámec UART komunikace

Mikrokontrolér využívá pro komunikaci kanálem UART registry Rx/Tx Status, data a posuvné registry. Status registry značí, zda je komunikace aktivní, a v data registrech jsou buď data, která chceme vysílat, nebo přijatá data. Tx posuvný registr generuje vysílané bity včetně start i stop bitu. Do Rx posuvného registru se postupně natahují veškeré přijímané bity.

4.4 Řešení nedostatečného množství pinů

V ideálním případě se mikrokontrolér vybírá podle toho, k jakému účelu bude použit. Většinou se sepíše veškeré funkcionality, které budou potřeba, a parametry, které se od mikrokontroléru očekávají, a následně se najde ten nejlevnější produkt splňující dané požadavky. Výjimečně se ovšem může stát, že máme k dispozici jeden konkrétní mikrokontrolér a potřebujeme na něj implementovat funkce přesahující jeho možnosti. Jeden z těchto případů může být nedostatečný počet pinů, který lze částečně vyřešit tím, že několik pinů bude mít vícenásobnou funkci, která se bude přepínat buď automaticky, nebo zásahem uživatele. Funkce může být buď modifikována pravidelně v průběhu programu, nebo je možné ji nastavit vždy po resetu a v průběhu programu ji nechat nezměněnou.

| | Pin | Periferie |
|----|---------------|------------------|
| 1. | PC14/PB7/8/9 | PWM výstup |
| 2. | PB0/1/PA8/11 | RGB LED ovládání |
| 3. | PA10 | UART TX |
| 4. | PA13 | UART RX |
| 5. | PA14/15/PB5/6 | ADC – frekvence |
| 6. | | ADC – střída |

Tabulka 4.1: Počet dostupných pinů a potřebných periférií pro Interaktivní PWM generátor na mikrokontroléru STM32G030J6

Seznam periférií Interaktivního PWM generátoru, které potřebují vstupní nebo výstupní pin, je v tabulce 4.1 ve třetím sloupci. Seznam dostupných pinů mikrokontroléru STM32G030J6 je ve sloupci druhém. Ze srovnání těchto dvou seznamů je vidět, že přebývá jedna periferie. Pro možnost vytvoření požadovaného PWM generátoru by tedy bylo dostatečné, kdyby dvě periferie sdílely jeden pin. Není možné, aby se o pin dělil výstup generátoru nebo ovládání RGB LED, které jsou z principu potřeba neustále. Obě tyto periferie tedy potřebují samostatný pin. Nabízí se, aby se o pin dělila jedna UART a jedna ADC periferie. Výsledný pinout je v tabulce 4.2.

| Pin | Alt. funkce 1 | Alt. funkce 2 |
|---------------|-----------------|---------------|
| PC14/PB7/8/9 | PWM výstup | - |
| VDD | napájení 3,3 V | - |
| VSS | napájení zem | - |
| NRST | Reset | - |
| PB0/1/PA8/11 | UART TX | - |
| PA10 | UART RX | ADC – střída |
| PA13 | ADC – frekvence | - |
| PA14/15/PB5/6 | RGB LED | - |

Tabulka 4.2: Alternativní funkce pinů Interaktivního PWM generátoru

Aby bylo fungování generátoru pro uživatele přímočaré, je nejvhodnější vytvořit dva režimy ovládání, které se nepřekrývají:

- režim ovládání pomocí UART komunikace
- režim manuálního ovládání pomocí trimérů

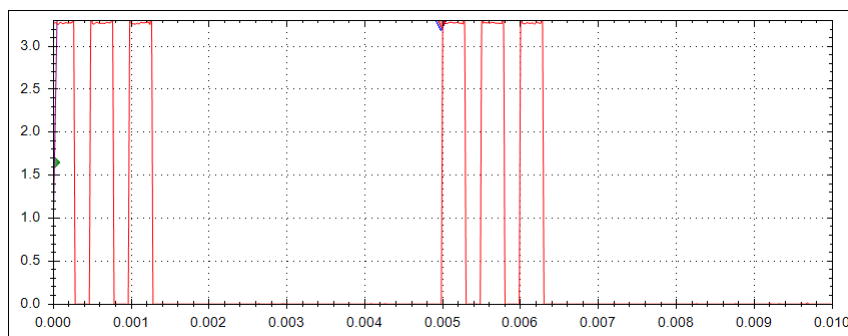
Mezi těmito režimy není možné přecházet v průběhu programu. K oboustrannému přechodu by totiž bylo potřeba tlačítko nebo jiná periférie, která by vyžadovala další vstupní pin. Výběr režimu bude proveden vždy jen po resetu, resp. pro provedení všech reset operací popsaných v kapitole 3.6. V ten moment bude na pin PA9 připojen pull-down rezistor a pomocí IDR registru se zjistí, zdali je na pinu napětí. Pokud je na pinu napětí 3,3 V, značí to, že je k pinu připojen UART vodič jiného zařízení viz 4.3, a mikrokontrolér se zapne do režimu ovládání pomocí UART. V opačném případě, kdy na pinu není detekována logická 1, se spustí do režimu manuálního ovládání.

Pokud je mikrokontrolér v režimu UART a zároveň je k pinu PA10 připojen trimr, mohou nastat problémy v odesílání dat z PC do mikrokontroléru. Konkrétně nastanou, pokud bude trimr natočen tak, že na něm bude napětí větší než 1,8 V nebo menší než 0,8 V (zjištěno experimentálně). Důvodem je, že v prvním případě, kdy je napětí větší než 1,8 V, odesílatel nedokáže přebít napětí z trimru na přijímacím pinu mikrokontroléru a nestáhne ho tedy dostatečně nízko, což povede k neschopnosti mikrokontroléru rozeznat hodnoty jednotlivých bitů. Nerozezná ani start bit a komunikace tedy ani nezačne viz 4.3. V druhém případě, kdy je napětí menší než 0,8 V, zase odesílatel a mikrokontrolér nevytáhnou napětí dostatečně vysoko a není tak rozpoznatelná klidová úroveň. Pro správnou funkci UART komunikace je nutné, aby hodnota napětí z trimru na pinu PA10 byla mezi 0,8 a 1,8 V.

Kapitola 5

Dvoukanálový generátor

Dvoukanálový generátor produkuje dva typy signálů – kontinuální PWM signál a burst signál. Generace probíhá na dvou kanálech, přičemž signály na těchto dvou kanálech jsou na sobě nezávislé. Burst signál je samostatná skupina pulsů trvající krátkou dobu. Příklad signálu tohoto typu je na obrázku 5.1.



obr 5.1: Příklad burst signálu

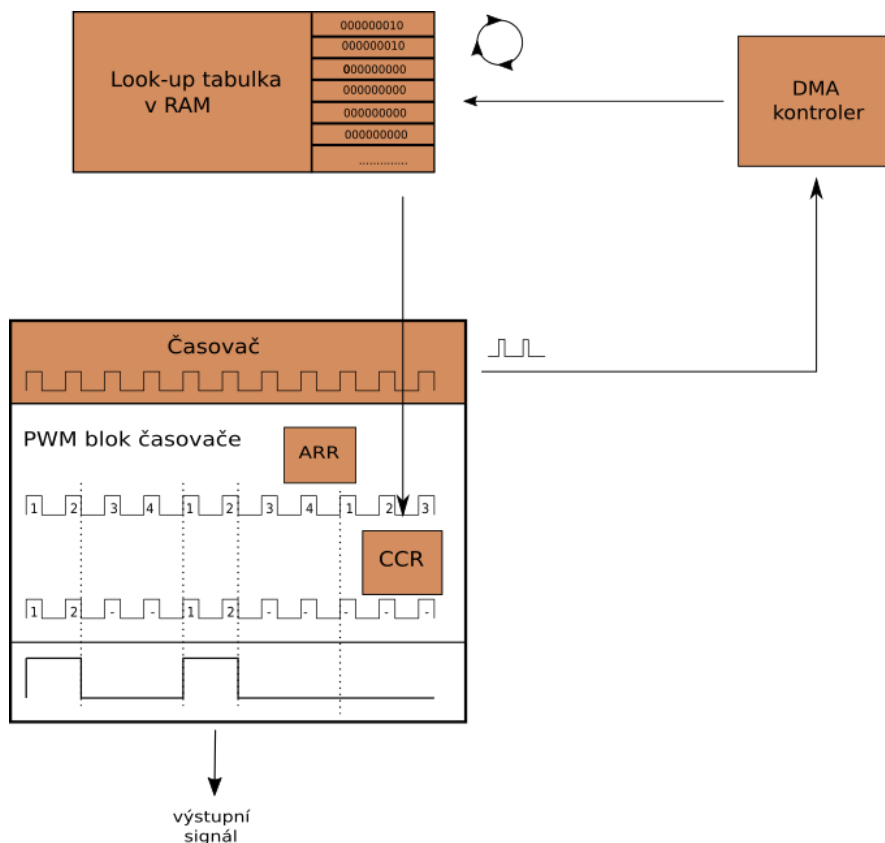
5.1 Metody generace burst signálu

Burst signál je možné generovat dvěma způsoby. Prvním způsobem je použití dvou časovačů, přičemž jeden časovač pouze generuje pulsy s požadovanou frekvencí a střídou. Druhý časovač pravidelně spouští a vypíná první časovač. Počet pulsů je možné nastavit dobou spuštění prvního časovače a frekvenci signálu lze naopak nastavit dobou vypnutí prvního časovače.

Princip druhého způsobu generace burst signálu je zobrazen na obrázku 5.2. Spočívá v použití DMA a časovače v PWM módu. V RAM je uložena look-up tabulka, která obsahuje hodnoty střídy pulsů. Časovač časuje kontinuální DMA přenos, který přenáší hodnoty z look-up tabulky do CCR registru. Hodnota v CCR registru určuje spolu s hodnotou v ARR registru střídu signálu. ARR registr navíc částečně určuje periodu signálu. Hodnota v ARR registru je konstantní. Výstupním signálem je proto skupina pulsů, které mají všechny stejnou periodu, ale mohou mít odlišnou střídu.

Pro generaci burst signálu se do několika prvních řádků look-up tabulky zapíše požadovaná střída a zbytek tabulky se zaplní nulami. Počet nenulových řádků udává počet pulsů. Pro změnu počtu pulsů nebo změnu střídy je tedy dostačující zastavit DMA přenos, modifikovat look-up tabulku a následně DMA přenos opět spustit.

Dalšími parametry, které lze u burst signálu modifikovat, jsou frekvence a tzv. burst rate (viz



obr 5.2: Generace PWM signálu použitím PWM bloku časovače a DMA

5.3). Perioda signálu je vždy násobkem periody burst pulsu. Abychom periodu signálu změnili, je potřeba modifikovat délku look-up tabulky, která přenášena do CCR registru. Pro změnu burst rate je dostatečné změnit frekvenci časovače, resp. prescaler časovače. Pokud ovšem chceme změnit burst rate a zároveň zachovat frekvenci, je potřeba po změně prescaleru ještě změnit délku look-up tabulky.

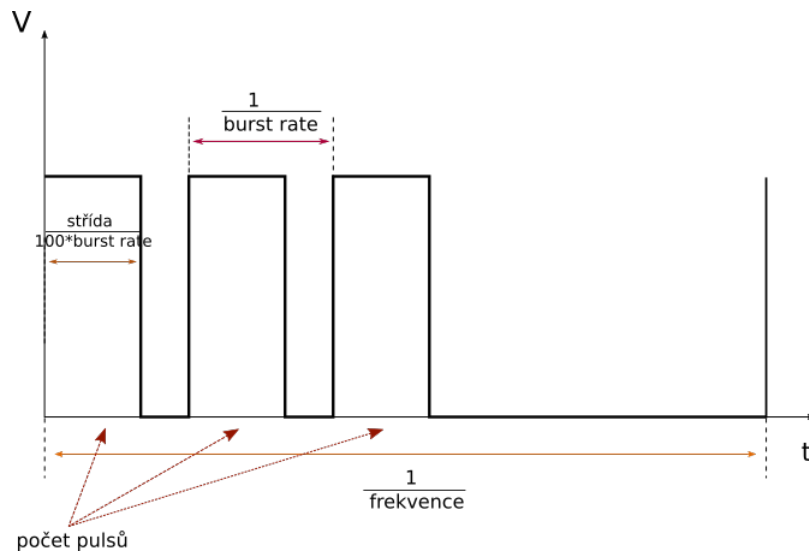
Při změně parametrů burst signálu nesmí dojít k tomu, že pulsy budou přesahovat nastavenou periodu, musí tedy platit:

$$frekvence \leq \frac{burst_rate}{pocet_pulsu}. \quad (5.1)$$

Další omezení je dáno konečnou velikostí datové části RAM mikrokontroléru. Ta omezuje velikost look-up tabulky u mikrokontroléru STM32G030J6 na 1000. Proto musí platit:

$$\frac{burst_rate}{frekvence} \leq 1000. \quad (5.2)$$

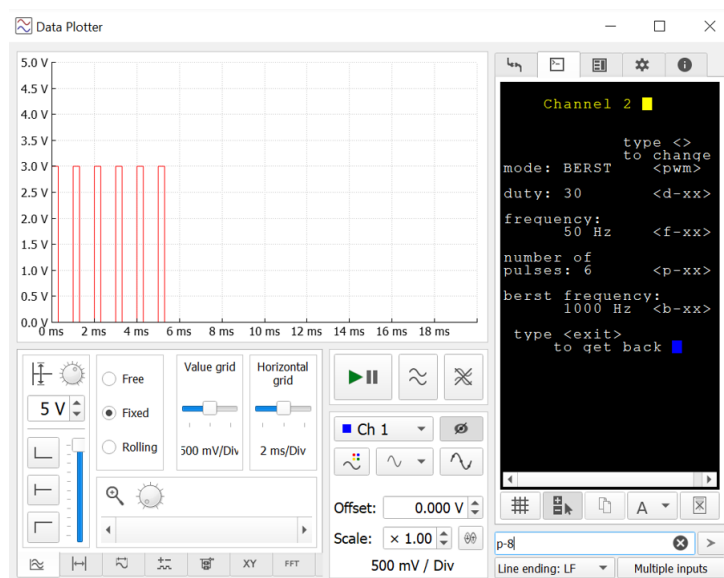
Pro pohodlné používání je vhodné, aby se uživatel nemusel o tyto podmínky starat. Proto je při každé změně burst rate, která nevyhovuje daným podmínkám, frekvence automaticky upravena. Při nevhodné změně frekvence je do terminálu vypsána chybová hláška.



obr 5.3: Parametry burst signálu

5.2 Použití generátoru s Dataplotterem

Dvoukanálový generátor produkuje signál s větším množstvím parametrů, což může být pro uživatele matoucí. Proto je vhodné, aby si uživatel mohl daný signál zobrazit v počítači. K tomu je možné použít program Dataplotter, což je nástroj, který právě zobrazení dat signálu v počítačové aplikaci umožňuje. Navíc má k dispozici vlastní terminál, který podporuje odesílání znaků vypsanych v terminálu zpět do mikrokontroléru na základě kliknutí na daný znak. Tato funkce umožňuje vytvoření tlačítek v terminálu. Ve Dvoukanálovém generátoru slouží tlačítka k přepínání mezi stránkami (viz kap. 5.3) terminálu a k vykreslení jedné periody signálu.



obr 5.4: Příklad použití Dvoukanálového generátoru s Dataplotterem

5.3 Terminálový výstup – přepínání mezi stránkami

Aby bylo ovládání Dvoukanálového generátoru co nejpohodlnější, byl vytvořen několikastránkový koncept ovládání v sériovém terminálu. Jsou k dispozici tři stránky – hlavní stránka a stránka 1. a 2. kanálu. Po spuštění terminálu se vytiskne hlavní stránka, kde jsou pouze základní informace – o jaké zařízení se jedná, jaký typ signálu v danou chvíli produkují jednotlivé kanály a instrukce pro přepnutí na stránky kanálů. Po přepnutí na stránku 1. nebo 2. kanálu jsou vytištěny veškeré parametry právě generovaného signálu na daném kanálu. Také jsou zde vytištěny instrukce pro změnu těchto parametrů a pro přepnutí zpět na hlavní stránku.

The image shows two screenshots of a PuTTY terminal window titled 'COM9 - PuTTY'. The first screenshot, labeled (a), shows the main menu with the following text: 'Department of Measurement', 'K.Pravodva', 'Pulse generator', 'Channel 1' (with a green cursor), 'mode: pwm', 'type <ch1> to set up', 'Channel 2:' (with a yellow cursor), 'mode: berst', and 'type <ch2> to set up'. The second screenshot, labeled (b), shows the settings for Channel 1 with the following text: 'Channel 1' (with a green cursor), 'type <>', 'to change', 'mode: BERST <pwm>', 'duty: 50 <d-xx>', 'frequency: 20 Hz <f-xx>', 'number of pulses: 1 <p-xx>', 'berst frequency: 200 Hz <b-xx>', 'type <exit>', and 'to get back' (with a blue cursor).

(a) : Hlavní stránka

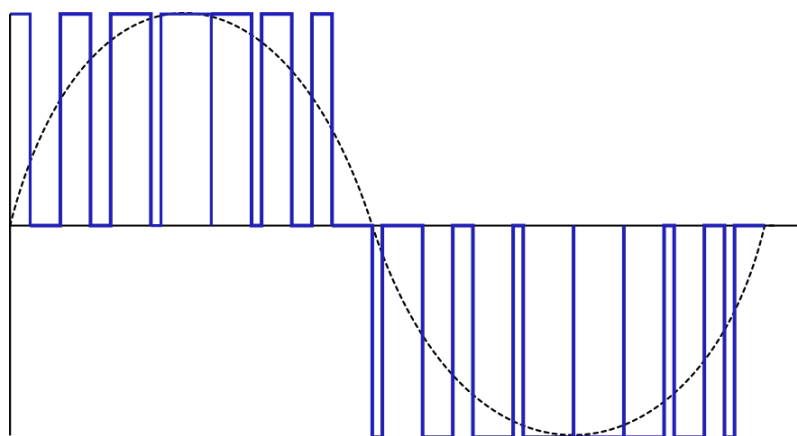
(b) : Stránka 1. kanálu

obr 5.5: Stránky Dvoukanálového generátoru

Kapitola 6

Harmonický generátor

Harmonický generátor by měl být dle zadání implementovaný na mikrokontrolér STM32G030J6. Tento mikrokontrolér ovšem nemá k dispozici DA převodník, proto není možné použít pro implementaci Harmonického generátoru stejnou metodu jako v kapitole 3. Alternativní variantou je použití externího DA převodníku nebo je možné generovat takový PWM signál, ze kterého odfiltrováním vyšších harmonických složek vznikne sinusový signál [16] viz obr. 6.1.



obr 6.1: Generace harmonického signálu využitím PWM s modulací střídy

Tento signál je možné vygenerovat pomocí metody na obrázku 5.2 pouze s využitím jiné look-up tabulky. Hodnoty v nové look-up tabulce pro harmonický signál s maximální amplitudou jsou dle [16] následující:

$$v[i] = \frac{ARR}{2} \cdot \left(\sin\left(\frac{2 \cdot \pi \cdot i}{n}\right) + 1 \right), \quad (6.1)$$

kde $v[i]$ je i -tý vzorek, n délka look-up tabulky a ARR je hodnota auto-reload registru.

Amplitudu signálu lze zmenšit zkrácením všech pulsů o stejný poměr – resp. zmenšením střídy všech pulsů. To lze provést dvěma způsoby. Prvním způsobem je přepočítání look-up tabulky a druhý způsob je změna ARR . Pro implementaci dvoukanalového Harmonického generátoru a Harmonického generátoru na mikrokontrolér STM32F042 byla vybrána druhá varianta, protože méně zatěžuje CPU. Na druhou stranu, z rovnice 6.1 je zřejmé, že změna ARR způsobí změnu výsledné frekvence signálu. Aby zůstala zachována původní frekvence, je potřeba po změně amplitudy signálu upravit

prescaler časovače. Pokud bude ovšem hodnota prescaleru již minimální, bude se frekvence signálu se snižováním amplitudy mírně snižovat. Pro implementaci jednokanálového Harmonického generátoru byla naopak vybrána první metoda.

6.1 Frekvence signálu Harmonického generátoru

Výsledná frekvence sinusového signálu generovaného pomocí PWM je:

$$f_{\text{sig}} = \frac{f_{\text{sysclk}}}{ARR \cdot n \cdot psc}, \quad (6.2)$$

kde f_{sysclk} je frekvence systémových hodin a psc je hodnota prescaleru. Čím větší je hodnota ARR, tím přesněji je možné nastavit střihu pulsů a tím kvalitnější je tedy signál. Na kvalitu signálu má vliv také počet vzorků v look-up tabulce. Se zvětšováním kvality signálu se musí snižovat maximální frekvence generátoru. Hodnota ARR byla nastavena na 64. Počet vzorků v tabulce byl nastaven pro mikrokontrolér F042 na maximální možnou hodnotu, která se spolu s ostatními proměnnými vešla do paměti, 64 vzorků. Při maximální hodinové frekvenci 48 MHz a počtu vzorků v tabulce 64 je maximální frekvence sinusového signálu 11 719 Hz. Nejvyšší možné hodnoty frekvencí jsou v následující tabulce.

| pořadové číslo frekvence (od nejvyšší) | frekvence F042 | frekvence G030 |
|---|-------------------|-------------------|
| 1. | 11 719 | 11 764 |
| 2. | 5 859 | 11 627 |
| 3. | 3 906 | 11 494 |
| 4. | 2 929 | 11 363 |
| 5. | 2 343 | 11 235 |
| 6. | 1 953 | 11 111 |

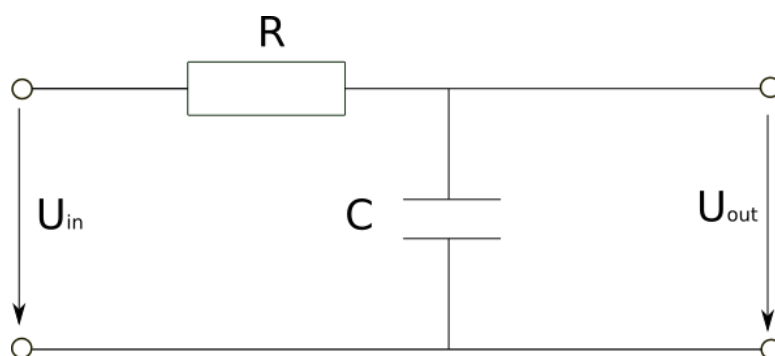
Tabulka 6.1: Možné hodnoty frekvencí signálu Harmonického generátoru na mikrokontrolérech STM32F042 a STM32G030

Z tabulky 6.1 je zřejmé, že nastavení frekvencí Harmonickým generátorem na mikrokontroléru F042 není příliš flexibilní. Lepší parametry vykazují jednokanálový a dvoukanálový Harmonický generátor na STM32G030, u kterých byla použita metoda s proměnnou délkou look-up tabulky, více v kapitole 6.3.

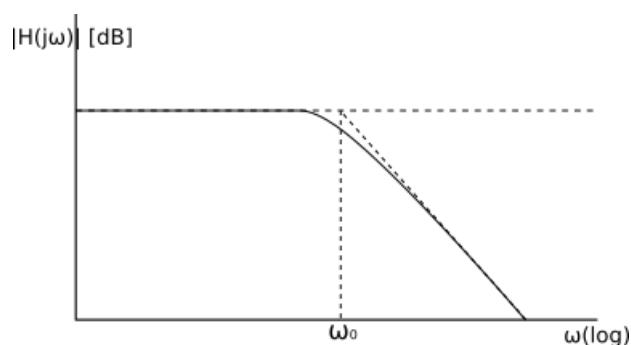
6.2 Návrh filtru typu dolní propust

Pro modifikaci PWM signálu v sinusový signál potřebujeme odfiltrovat vyšší harmonické složky. Je tedy nutné použít filtr typu dolní propust. Nejjednodušší variantou dolní propusti je RC článek 1. řádu (viz obr. 6.2 a 6.3). Bylo by možné použít složitější filtr s lepšími parametry, ale vzhledem k tomu, že jedním z požadavků je možnost rychlého zapojení na nepájivém poli, je RC článek 1. řádu ideální volbou.

V kapitole 6.1 bylo odvozeno, že mezní kmitočet generátoru je cca 11 000 Hz. Pro mezní kmitočet



obr 6.2: Dolní propust 1. řádu



obr 6.3: Amplitudová charakteristika dolní propusti 1. řádu

dolní propusti dle [17] platí:

$$\omega_0 = \frac{1}{R \cdot C}. \quad (6.3)$$

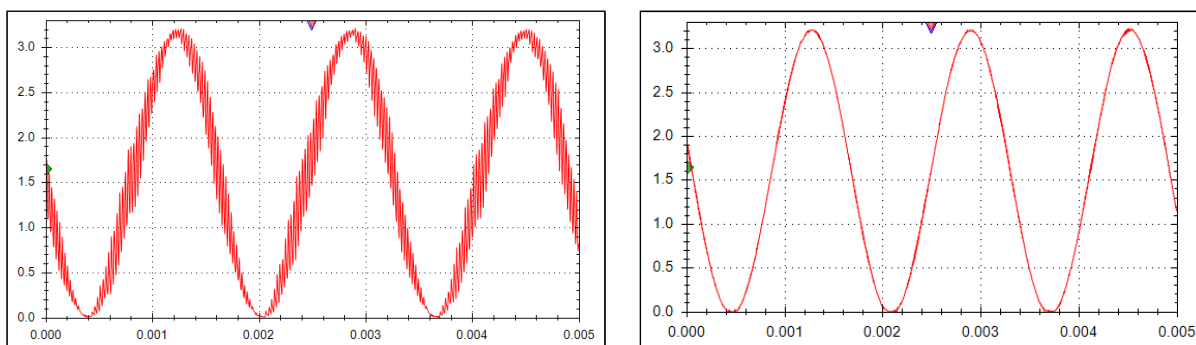
Jednou z možných variant kombinací je kapacita 10 nF a odpor 1000 Ω. Pro frekvence nad 1300 Hz je filtr s těmito hodnotami příliš tlumící, proto byla zvolena kapacita 20 nF a odpor 500 Ω.

6.3 Zamezení šumu a tlumení signálu při použití neměnného RC filtru

Pro jednoduché použití Harmonického generátoru je vhodné, aby měly prvky RC filtru 1. řádu stejný odpor a kapacitu pro všechny frekvence generovaného signálu. To je ovšem problematické. Aby byl signál netlumený a zároveň co nejméně zašumělý, je totiž potřeba se zvyšující frekvencí PWM signálu zvyšovat mezní kmitočet a se snižující frekvencí mezní kmitočet naopak snižovat. Tento problém je možné řešit tím, že frekvence PWM signálu bude neměnná a frekvence sinusového signálu se bude modifikovat pomocí počtu prvků v look-up tabulce.

Změnu počtu prvků v tabulce je možné vykonat přepočítáním celé look-up tabulky. Toto řešení je ovšem, jak již bylo řečeno, výpočetně náročné a pro implementaci Harmonického generátoru na mikrokontroléru STM32G030 se ukázalo být zcela nepoužitelné, kvůli nedostatečnému místu v paměti pro zahrnutí matematické knihovny. Alternativním řešením, které bylo implementováno, je použití jedné předem vypočítané tabulky o 1024 prvcích, ze které je potom vytvořena look-up tabulka o libovolném počtu prvků pravidelným vynecháním několika prvků. Např. pro vznik look-up tabulky o délce 512 bude z předem vypočítané tabulky vynechán každý druhý prvek.

Výše zmíněné řešení vynecháváním prvků vyžaduje, aby bylo v paměti mikrokontroléru k dispozici místo pro dvě tabulky o velikosti 1024 vzorků. Proto toto řešení nebylo možné implementovat na mikrokontroléru F042, který má dvakrát menší velikost paměti než mikrokontrolér G030 [6]. Z toho důvodu je signál generovaný na STM32F042 více zkreslený na nižších frekvencích než signál generovaný na STM32G030 viz obr.6.4.



obr 6.4: Rozdílný šum signálu o frekvenci 616 Hz na mikrokontrolérech F042 (vlevo) a G030 (vpravo)

U dvoukanalového Harmonického generátoru je potřeba jedna tabulka pro každý kanál. Vzhledem k tomu, že paměť nedokázala spolu s ostatními proměnnými obsáhnout tři tabulky o velikosti 1024, byly použity tabulky o velikosti 512. To opět způsobuje zkreslení na nižších frekvencích. Nicméně toto zkreslení není tak podstatné jako u generátoru na mikrokontroléru STM32F042.

6.4 Změna amplitud dvou signálů pomocí trimru

Mikrokontrolér STM32G030J6 má k dispozici osm pinů. Pokud dva piny použijeme pro napájení, jeden pin pro reset a dva piny pro UART komunikaci, zbydou k použití tři piny. V případě jednocanalového Harmonického generátoru je tedy možné využít dva piny pro manuální ovládání frekvence a střídý trimrem. V případě dvoukanalového generátoru je ovšem k dispozici pouze jeden pin pro manuální ovládání.

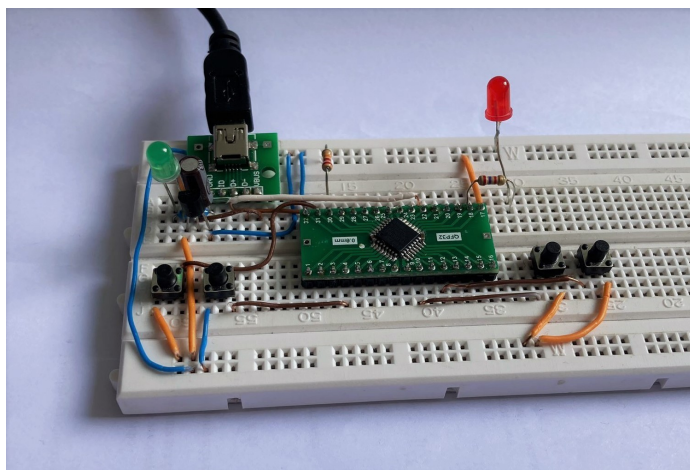
Nedostatek pinů u dvoukanalového Harmonického generátoru byl vyřešen přepínanou funkcí trimru. Po spuštění generátoru je trimrem ovlivňována amplituda prvního kanálu. Pomocí jediného příkazu poslaného přes UART komunikaci je možné trimr přesměrovat tak, aby ovlivňoval amplitudu druhého kanálu. Jeho funkci je také možné zcela vypnout. Pokud by bylo potřeba ovládat trimrem i frekvenci signálu, bylo by v příští verzi generátoru možné přidat další příkazy pro přesměrování trimru právě na ovládání frekvence.

Kapitola 7

Zhodnocení dosažených výsledků práce

V rámci této práce byly navrženy, naprogramovány a vytvořeny čtyři typy generátorů a to Funkční generátor, Interaktivní PWM generátor, Dvoukanálový PWM generátor a Harmonický generátor. Podrobné informace jsou spolu s návodem k použití k dispozici v uživatelských manuálech (viz příloha B), které byly vytvořeny ke každému generátoru.

1. Funkční generátor navržený v rámci této práce dokáže produkovat sinusový, trojúhelníkový, obdélníkový a rampový signál. Byla vytvořena verze pro mikrokontroléry STM32G431KB, STM32G431RB a STM32F303. Generátor ve verzi G431KB používá komunikaci USB o rychlosti 115 200 Bd, pomocí níž je možné měnit typ signálu, jeho frekvenci, amplitudu a offset. Verze na mikrokontrolérech G431RB a F303 umožňují totéž s použitím UART komunikace. Amplituda i offset mají 100 různých hodnot. Rozlišení nastavení frekvence je 2 %. Požadovaná frekvence, aktuálně nastavená frekvence i délka signálu jsou vypisovány do terminálu. Maximální možná frekvence je 300 kHz. Dále je možné použít dvě tlačítka pro změnu frekvence s krokem $o + a - 100$ Hz.



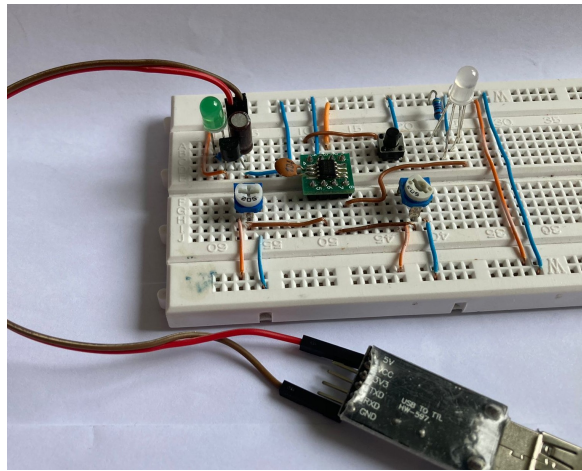
obr 7.1: Funkční generátor

2. Interaktivní PWM generátor produkuje PWM signál s nastavitelnou frekvencí a střídou. Byl navržen pro mikrokontrolér STM32G030J6. Pracuje ve dvou různých módech:
 - UART mód, ve kterém mohou být parametry zobrazeny a modifikovány pomocí kanálu UART při komunikaci o rychlosti Bd.

- Trimr mód, v němž lze parametry měnit pomocí dvou trimrů.

Interaktivní generátor má také výstup pro RGB LED. Pokud je nastaven trimr mód, RGB LED mění barvu v závislosti na frekvenci. V UART módu svítí RGB LED modře.

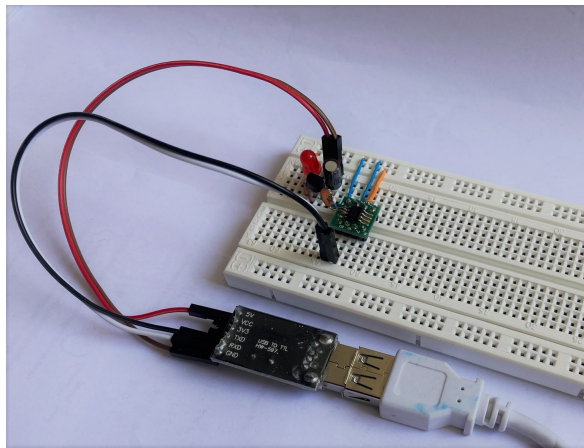
Frekvence v UART módu je od 1 Hz do 640 kHz, střída je mezi 1 a 100 % a má 100 možných hodnot. Frekvence v trimr módu může být od 100 do 1000 Hz, střídu lze také nastavit na hodnotu mezi 1 a 100 %. Základní úrovně napětí jsou 0 a 3,3 V. Kladná hodnota napětí je ovšem dána napájecím napětím mikrokontroléru a může se tedy měnit. Pro zprostředkování komunikace v počítači lze použít sériový terminál Putty, Realterm nebo Xterm. Počáteční parametry PWM signálu jsou střída 50% a frekvence 200 Hz.



obr 7.2: Interaktivní PWM generátor

3. Dvoukanálový PWM generátor byl implementován na mikrokontrolér STM32G030J6. Produkuje kontinuální PWM signál nebo burst signál na dvou kanálech. Frekvence, střída, počet pulsů a tzv. burst rate jsou parametry, které je možné modifikovat pomocí UART komunikace o modulační rychlosti 115 200 Bd. Základní úrovně napětí jsou 0 a 3,3 V. Kladná hodnota napětí je ovšem dána napájecím napětím mikrokontroléru a může se tedy měnit. Maximální frekvence kontinuálního signálu i burst rate je 250 kHz, minimální frekvence je 1 Hz a střída má 30 možných hodnot. Jako sériový terminál pro zprostředkování komunikace v počítači může být použit např. Realterm, Xterm, Putty, ale také sériový terminál v Dataplotteru. Při používání Dataplotteru je možné zobrazit jednu periodu právě generovaných signálů.
4. Harmonický generátor produkuje PWM signál s proměnnou střídou, který může být filtrován dolní propustí v sinusový signál. Maximální generovatelná frekvence je 11,712 kHz. Maximální amplituda signálu je shodná s napájecím napětím mikrokontroléru. Počáteční hodnoty parametrů generovaného signálu jsou: maximální amplituda (nejčastěji 3,3 V), frekvence 1953 Hz. Byly vytvořeny tři verze Harmonického generátoru.

První verze je dvoukanálový Harmonický generátor na mikrokontroléru STM32G030J6. PWM signál je generován na dvou nezávislých kanálech. Je možné měnit amplitudu a frekvenci signálů na obou kanálech pomocí UART komunikačního rozhraní o modulační rychlosti 115 200 Bd. Amplituda signálu má 200 možných hodnot. Amplitudu je možné modifikovat pomocí trimru, který představuje nastavovací prvek, jehož napětí je přepočteno na hodnotu amplitudy. K

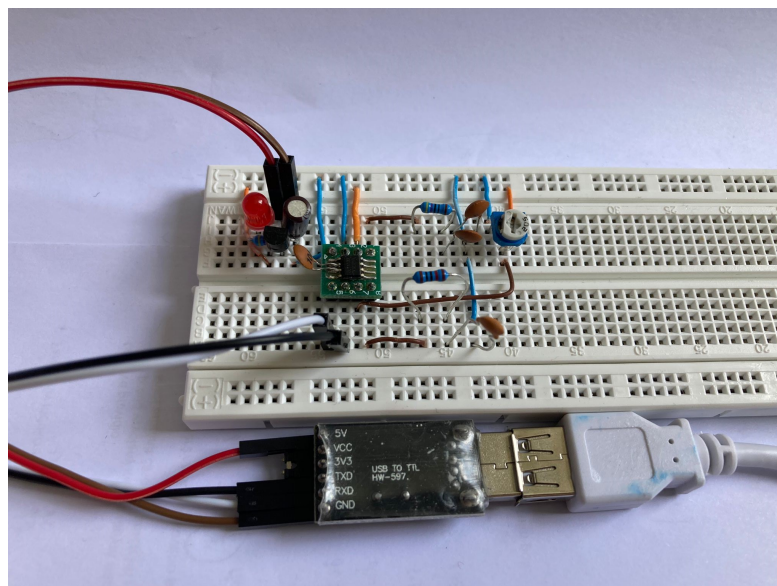


obr 7.3: Dvoukanálový PWM generátor

dispozici je jeden ADC vstup, na který může být trimr připojen. Ve výchozím stavu je trimrem ovládaná amplituda signálu na prvním kanálu. Pomocí UART komunikace je možné přeměřovat trimr na druhý kanál nebo ho vypnout.

Druhá verze je jednocanálový generátor na mikrokontroléru STM32F042. Tento generátor nemá k dispozici trimry k ovládní parametrů signálu. Ty se dají měnit pouze pomocí USB komunikačního rozhraní o modulační rychlosti 115 200 Bd. Amplituda má 200 možných hodnot, možné hodnoty frekvence jsou naopak velmi omezené viz tabulka 6.1.

Poslední verze je jednocanálový generátor na STM32G030J6. K dispozici jsou dva ADC vstupy, jedním je možné modifikovat frekvenci, druhým amplitudu generovaného signálu, která může mít 100 různých hodnot. Pro komunikaci je opět použito rozhraní UART. V této verzi jsou navíc implementovány metody pro zamezení šumu signálu na nižších frekvencích a s nižšími amplitudami.



obr 7.4: Harmonický generátor

Kapitola 8

Závěr

V rámci této práce byly navrženy a implementovány čtyři typy digitálních generátorů signálů realizovaných s mikrokontroléry STM32. Realizace byla provedena v celkem osmi různých verzích. Zařízení mohou sloužit jako výukový prvek, například pro demonstraci vlastností signálů, testování jiných laboratorních přístrojů nebo pro realizaci různých experimentů. Interaktivní PWM generátor je možné použít na akcích pořádaných pro děti skupinou Witches.

Generátory jsou umístěny buď na nepájivém poli, nebo na vývojové desce Nucleo. Mají přímočaré ovládání, které je navíc detailně popsáno v uživatelských manuálech. Detailní popis generátorů je také v kapitole 7.

Každému ze čtyř vytvořených generátorů se věnuje jedna samostatná kapitola. Na začátku každé z těchto kapitol je popsána metoda, která byla pro generaci signálu využita. Následně je popsán způsob, jakým jsou modifikovány parametry signálu. Poté se v závislosti na specifiku daného generátoru zabývá buď způsobem komunikace, použitými periferiemi, nebo speciálními součástkami.

Prvním z navržených generátorů byl funkční generátor čtyř různých průběhů. Pro jeho vytvoření byl použit DA převodník, časovač a přímý přístup do paměti. Navíc na něj byl implementován optimalizační algoritmus pro zlepšení přesnosti frekvence. Dále na něm byla zprovozněna USB komunikace a obsluha tlačítek. Součástí přípravy implementace tohoto generátoru byla také rešerše procesu bootování, který je obecně podstatný pro práci s mikrokontroléry.

Následně byl vytvořen impulsní generátor pomocí časovače v PWM módu. K tomu bylo potřeba zajistit nastavování barev na RGB LED WS2812. Dále bylo nutné se zabývat implementací obsluhy trimrů a implementací UART komunikace. Nakonec se musel vyřešit problém nedostatečného počtu pinů generátoru STM32G030J6 a bylo navrženo možné řešení pomocí implementace dvou oddělených módů ovládání.

Dalším vytvořeným generátorem byl dvoukanálový impulsní generátor. Ten používá, stejně jako předchozí impulsní generátor, časovač v PWM módu, avšak časovač v tomto případě pracuje současně s přímým přenosem vzorků z paměti do CCR registru, což umožňuje generaci burst signálu. S tím bylo ovšem nutné se věnovat mezním podmínkám generace burst signálu. Následně byl generátor uzpůsoben k použití s Datplotterem a byl přidán koncept přepínání stránek v terminálu.

Poslední navržený generátor byl generátor sinusového signálu. Ten byl vytvořen pomocí filtrace speciálního PWM signálu. V rámci návrhu generátoru se bylo tedy nutné zabývat také návrhem RC filtru 1. řádu a metodám, které vedou k hladkému výstupu z filtru při nízkých amplitudách i frekvencích. Nakonec bylo použito jedno z možných řešení nedostatku pinů pomocí přepínané funkce trimru.

Dle mého názoru bylo zadání práce splněno v plném rozsahu. Byly realizovány signálové a impulsní generátory na všech požadovaných typech mikrokontroléru řady STM32. Generátory využívají

nejrůznější periferie, jako jsou komunikační kanály UART a USB či tlačítka, potenciometry a RGB LED. Všechny generátory je plně funkčních a připravených k použití.



Literatura

- [1] HAASZ, Vladimír a Miloš SEDLÁČEK. *Elektrická měření: přístroje a metody*. Dotisk 1. vyd. Praha: Vydavatelství ČVUT, 2000. 327 s. ISBN 80-01-01717-6.
- [2] STMICROELECTRONICS. *Datasheet - STM32G431x6 STM32G431x8 STM32G431xB - Arm® Cortex®-M4 32-bit MCU+FPU, 170 MHz /213 DMIPS, up to 128 KB Flash, 32 KB SRAM, rich analog, math accelerator*. Rev Oct. 2021.
- [3] STMICROELECTRONICS. *RM0440 Reference manual: STM32G4 Series advanced Arm-based 32-bit MCUs - Reference manual*. Rev Feb. 2021.
- [4] HLADÍK, Jiří. Single chip software defined instrumentation for educational purposes. *POSTER* [online]. 2017, 4 [cit. 2021-11-28]. Dostupné z: radio.fel.cvut.cz/conf/poster/poster2017/proceedings/Poster_2017/Section_EI/EI_015_Hladik.pdf.
- [5] STMICROELECTRONICS. *DS12991: Datasheet - STM32G030x6/x8*. Rev Jan. 2022.
- [6] STMICROELECTRONICS. *STM32F042x4 STM32F042x6: ARM®-based 32-bit MCU, up to 32 KB Flash, crystal-less USB FS 2.0, CAN, 9 timers, ADC and comm. interfaces, 2.0 - 3.6 V*. Rev 5 Jan. 2017.
- [7] MALÝ, Martin. *USB 2.0 - Typy a formáty přenosů*. [online] [cit. 2022-02-16]. Dostupné z: <https://vyvoj.hw.cz/navrh-obvodu/rozhrani/rs-485-rs-422/usb-20-typy-a-formaty-prenosu.html>.
- [8] STMICROELECTRONICS. *STM32G0 - GPTIM Advanced-control, general-purpose and basic timers*. [online] Rv 1.0.
- [9] YIU, Joseph. *Definitive Guide to ARM (R) Cortex (R)-M0 and Cortex-M0+ Processors*. Amsterdam: Elsevier Science & Technology, 2015. ISBN 0128032774.
- [10] BENINGO, Jacob. *Embedded Basics – Understanding the Microcontroller Boot Process*. [online] [cit. 2022-02-20]. Dostupné z: <https://www.beningo.com/understanding-the-microcontroller-boot-process/>
- [11] STMICROELECTRONICS. *STM32 microcontroller system memory boot mode*. [online] AN2606. Rev April 2022.
- [12] WORLDSEMI. *WS2812D-F5-1261*. [cit. 2022-02-05]. Dostupné z: <http://www.world-semi.com/solution/list-4-1.html#135>

- [13] GUPTA, Aditya. *The IoT Hacker's Handbook: A Practical Guide to Hacking the Internet of Things*. New York City: Apress L. P., 2019. ISBN 9781484242995.
- [14] STMICROELECTRONICS. *AN4566 Application note: Extending the DAC performance of STM32 microcontrollers* [online]. [cit. 2022-05-06]. Dostupné z: https://www.st.com/resource/en/application_note/dm00129215-extending-the-dac-performance-of-stm32-microcontrollers-stmicroelectronics.pdf
- [15] STMICROELECTRONICS. *AN2606 Application note: STM32 microcontroller system memory boot mode* [online]. [cit. 2022-05-06]. Dostupné z: <https://www.compel.ru/wordpress/wp-content/uploads/2019/10/en.cd00167594.pdf>
- [16] TSAI Charles. *Sine Wave Generation Using PWM With Hercules™ N2HET and HTU*. Texas Instruments., 2015. [online]. [cit. 2022-13-06]. Dostupné z: <https://www.ti.com/lit/an/spna217/spna217.pdf?ts=1652643930895>
- [17] STANESCU, Dan Gabriel, Mircea Emilian ARDELEANU a Andrei Cristian STAN. Designing, simulation and testing of low current passive filters used in the didactic activity. In: *2017 International Conference on Modern Power Systems (MPS)* [online]. IEEE, 2017, 2017, s. 1-4 [cit. 2022-05-15]. ISBN 978-1-5090-6565-3. Dostupné z: doi:10.1109/MPS.2017.7974385



Příloha A

Zdrojové kódy jednotlivých generátorů

- Funkční generátor pro mikrokontrolér STM32G431KB
- Funkční generátor pro mikrokontrolér STM32G431RB
- Funkční generátor pro mikrokontrolér STM32F303RE
- Interaktivní PWM generátor pro STM32G030J6
- Dvoukanálový PWM generátor pro STM32G030J6
- Harmonický generátor pro STM32F042F6
- Harmonický generátor jednokanálový pro STM32G030J6
- Harmonický generátor dvoukanálový pro STM32G030J6



Příloha B

Uživatelské manuály jednotlivých generátorů



B.1 Uživatelský manuál Funkčního generátoru

Function generator

for microcontroller STM32G431KB

Kateřina Pravdov

April 12, 2022

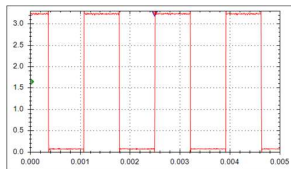
Function

Function generator (FCG) is a generator for microcontroller STM32G431KB. It produces 4 types of waveforms: triangle, square, sine and ramp wave. Frequency, amplitude, and offset can be viewed and modified by using USB communication at a baud rate of 115200 bits/s. FCG has also two buttons that can be used to increase or decrease frequency by 100 Hz.

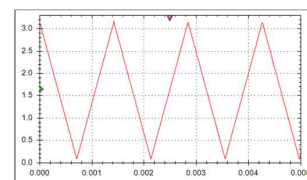
Maximum of generated signal is equal to the supply voltage of the microcontroller. Both amplitude and offset have 100 possible values. The frequency of the signal can be from 1 Hz to 300 kHz. FCG uses DMA with DAC for signal generation, therefore the number of possible frequency values is limited. The accuracy of signal's frequency is 2%. Real frequency and signal length can be also viewed via USB communication. As a serial terminal can be used Putty, Realterm or Xterm.

Types of generated waveforms:

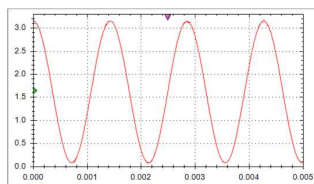
▶ square



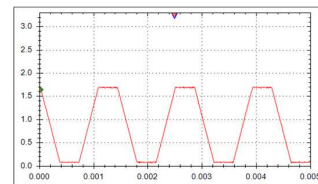
▶ triangle



▶ sine

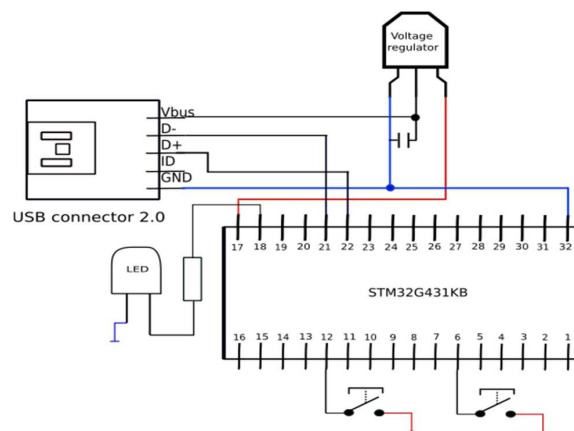


▶ ramp



Default values of generated signal (on both channels): sine wave, 700 Hz frequency

▶ Parameters of the USB communication are: 115200 bit/s, 8 bits of data, no parity, 1 stop bit, line ending: LF or CR. USB connector is needed and together with a voltage regulator can be used as a voltage source for the microcontroller. The wiring example is in the following picture:



Serial Terminal and Commands

► After opening serial terminal (Putty, Realterm, Xterm can be used), the following page is printed. Use commands below to change the parameters of signal.

```

COM7 - PuTTY
Department of Measurement      FEL, CTU
K.Pravodva                    version 1.0

Function Generator

type of signal:      sine
inserted frequency:  700 Hz
real frequency :    703 Hz
signal length :     1024 samples
amplitude level :   100 %
offset :             0 %
insert new command: █

commands: 't'/'q'/'s'/'r' -type of signal
          num 1 to 300000 -frequency
          '+/'/'-'/'++'/'--' -amplitude
          'u'/'d'/'uu'/'dd' -offset
  
```

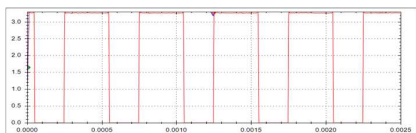
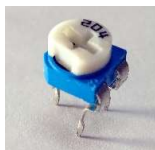
► Commands:

| |
|---|
| Frequency value |
| Type any number between 1 and 300 000 to change the frequency |
| t |
| Generate triangle waveform |
| q |
| Generate square waveform |
| s |
| Generate sine waveform |
| r |
| Generate ramp waveform |
| +/- |
| Increase/decrease the amplitude by the smallest step |
| ++/-- |
| Increase/decrease the amplitude by 20 steps |
| u/d |
| Increase/decrease the offset by the smallest step |
| uu/dd |
| Increase/decrease the offset by 20 steps |

Please report any bugs at: pravdkat@fel.cvut.cz

Simple Laboratory Generators

■ B.2 Uživatelský manuál Interaktivního PWM generátoru



Interactive PWM generator

for microcontroller STM32G030J6

Kateřina Pravdov

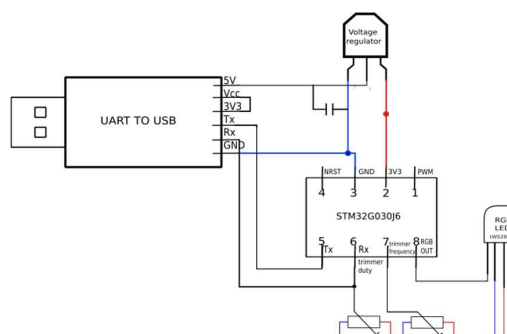
March 24, 2022

User Manual

The Interactive PWM generator is a generator for microcontroller STM32G030J6. It produces a continuous PWM signal, which frequency and duty cycle can be modified. The Interactive PWM generator works in two different modes. The first mode is the UART mode, in which parameters can be viewed and modified by using UART communication at a baud rate of 115200 bits/s. The second mode is the trimmer mode, in which parameters are modified with trimmers. The generator also has an output for RGB LED. LED changes colour depending on the frequency of the generated signal in trimmer mode and it has blue colour in UART mode.

Frequency in UART mode is between 1 and 640 000 Hz, duty cycle is between 1 and 100 % and it has 100 possible values. Frequency in trimmer mode can be from 100 to 1000 Hz, duty cycle can be from 1 to 100 %. ADCs that convert voltage from trimmers have 12 bits resolution. A voltage level of generated PWM signal is equal to the supply voltage of the microcontroller. As a serial terminal can be used Putty, Realterm and Xterm. Note that the microcontroller should be restarted after uploading the code. Default values of generated PWM signal: 50% duty cycle, 200 Hz.

► Parameters of the UART communication are 115200 bit/s, 8 bits of data, no parity, 1 stop bit, line ending: LF or CR. A UART-USB converter is needed and together with a voltage regulator can be used as a voltage source for the microcontroller. The wiring example is in the following picture:



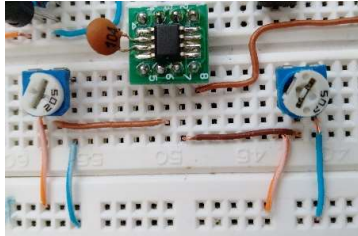
► General STM32G030J8 Interactive PWM generator pinout:



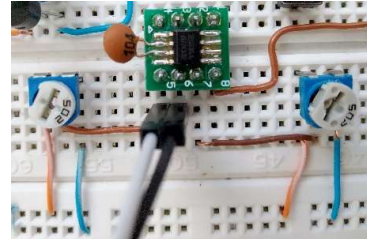
| Pin num | Function mode 1 | Function mode 2 |
|---------|-----------------|---------------------|
| 1 | PWM output | PWM output |
| 2 | VDD | VDD |
| 3 | GND | GND |
| 4 | Reset | Reset |
| 5 | UART TX | - |
| 6 | UART RX | trimmer - duty |
| 7 | - | trimmer - frequency |
| 8 | RGB LED | RGB LED |

- ▶ Mode is chosen immediately after reset.

If UART is not connected, trimmer mode is set up



If UART is connected, UART mode is set up



- ▶ If there is a trimmer connected to 6. Pin (in UART mode), its position must be in the middle (Voltage from trimmer can only be between 0.8 and 1.8 V)! Otherwise, data from PC will not be received by the microcontroller.

- ▶ To change mode, connect the reset pin to the ground.

- ▶ Serial terminal in UART mode

After opening serial terminal, following page is printed, press 'd' or 'f' to change duty or frequency

```
COM9 - PuTTY
Department of measurement          K.Pravodva
FEL, CTU                          pravdkat@fel.cvut.cz
                                     PWM generator
duty:          50                    press <> to change
frequency:    200                    <d>
real frequency: 200                  <f>
version: 1.0
```

Then insert duty or frequency and press enter.

```
COM9 - PuTTY
Department of measurement          K.Pravodva
FEL, CTU                          pravdkat@fel.cvut.cz
                                     PWM generator
duty:          50                    press <> to change
frequency:    200                    <d>
real frequency: 200                  <f>
                                     insert new frequency:
version: 1.0
```

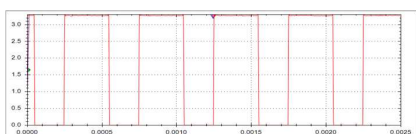
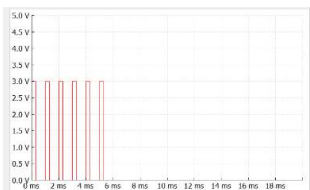
Please report any bugs at: pravdkat@fel.cvut.cz

Simple Laboratory Generators

Interactive PWM Generator



■ B.3 Uživatelský manuál Dvoukanálového PWM generátoru



Dual-Channel PWM generator

for microcontroller STM32G030J6

Kateřina Pravdov

March 24, 2022

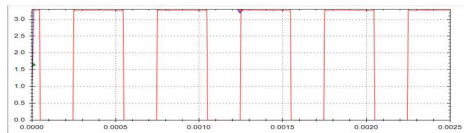
Function

Dual-Channel Generator (DCG) is a generator for microcontroller STM32G030J6. It produces a continuous PWM signal or burst signal on 2 channels. Frequency, duty cycle, burst rate and burst count can be modified. Those parameters can be viewed and modified by using UART communication at a baud rate of 115200 bits/s. DCG has a blinking LED on pin 7 indicating that the programme is running.

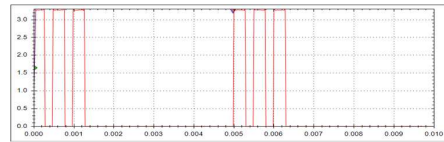
A voltage level of generated signal is equal to the supply voltage of the microcontroller. The frequency of continuous PWM signal and frequency of burst rate can be from 1 Hz to 250 kHz. The duty cycle has 30 possible values. As a serial terminal can be used Putty, Realterm, Xterm but also terminal in Dataplotter, which is a software tool that allows the implementation of software-defined devices with a display of signals. While using DCG with Dataplotter's terminal, one period of the currently generated signal can be plotted. Note that the microcontroller should be restarted after uploading the code.

Types of generated waveforms:

▶ continuous pwm signal

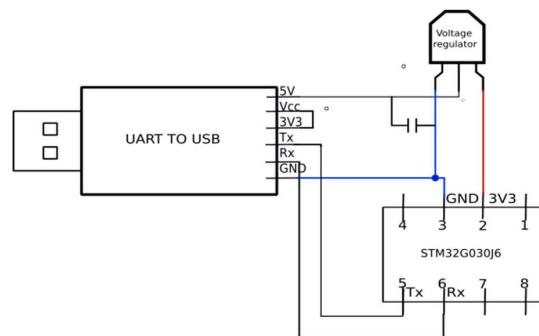


▶ burst signal



Default values of generated signal (on both channels): continuous PWM signal, 50% duty cycle, 20 kHz frequency

▶ Parameters of the UART communication are: 115200 bit/s, 8 bits of data, no parity, 1 stop bit, line ending: LF or CR. UART-USB converter is needed and together with a voltage regulator can be used as a voltage source for the microcontroller. The wiring example is in the following picture:



▶ General STM32G030J8 DCG pinout:

| Pin num | Function |
|---------|-----------|
| 1 | Channel 2 |
| 2 | VDD |
| 3 | GND |
| 4 | Reset |
| 5 | UART TX |
| 6 | UART RX |
| 7 | LED |
| 8 | Channel 1 |



Serial Terminal and Dataplotter

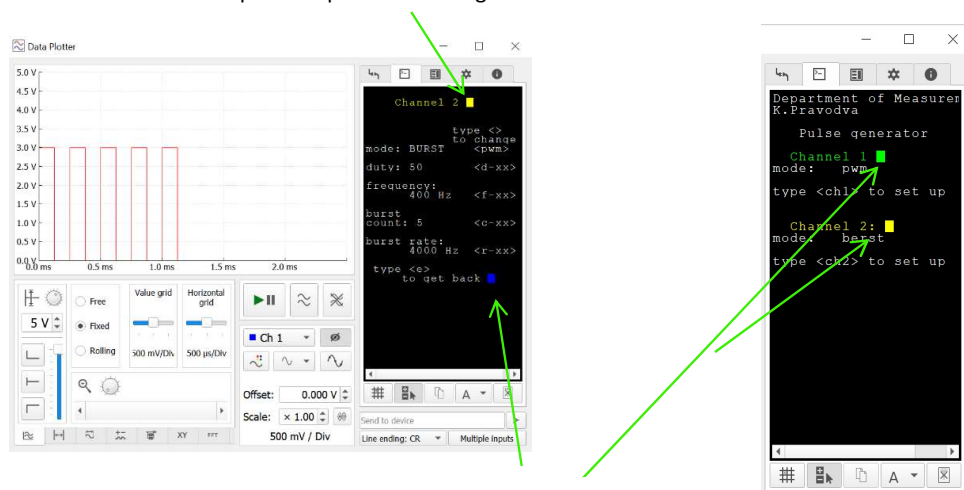
- After opening serial terminal (Putty, Realterm, Xterm can be used), the “main page” is printed. Commands <1> and <2> unlock the “Channel pages”, where the parameters can be viewed and set up.



In the first column on the “Channel page” are current signal parameters. In the second column are hints on how to change those parameters. Type <pwm> or <burst> to change current type of waveform.

- It is possible to use Dataplotter instead of a classical serial terminal. Pages can be switched with buttons in Dataplotter and one period of the signal can be plotted.

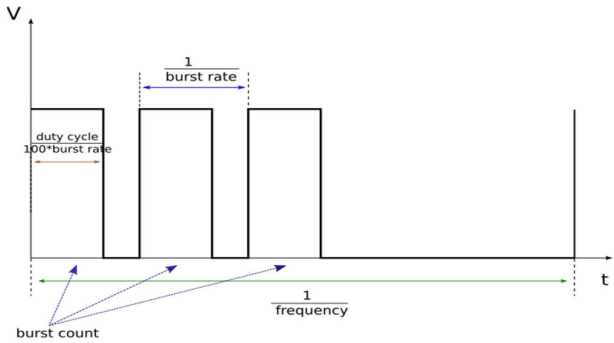
click here to plot one period of the signal



Use “buttons” to switch between pages

Signal Parameters Modification

► Burst signal has 4 parameters that can be modified – frequency, burst rate, duty cycle, burst count:



► Use those commands to change mentioned parameters:

| |
|--------------------|
| Set up burst count |
| c-xx |
| e.g. <c-3> |

| |
|-----------------------------|
| Set up burst rate |
| r-xx |
| e.g. <r-1000> |
| values: up to 250 000 Hz |

| |
|---|
| Set up duty cycle |
| d-xx |
| e.g. <d-20> |
| 30 values between 1 – 100 % : 3,6,10,13,16,20,23,26,30.... |

| |
|--------------------------|
| Set up frequency |
| f-xx |
| e.g. <f-10> |
| values: start at 1 Hz |

► Other commands:

| |
|--|
| pwm |
| Changes the burst signal into continuous pwm signal |
| burst |
| Changes the continuous pwm signal into burst signal |
| exit |
| Switches back to the main page |
| ch1 |
| Switches from the main page into the channel 1 page so the channel 1 parameters can be viewed and modified |
| ch2 |
| Switches from the main page into the channel 2 page so the channel 2 parameters can be viewed and modified |

► Note that for signal generation, following equations must apply:

$$\frac{\textit{burst rate}}{\textit{burst count}} \geq \textit{frequency}$$

$$1000 \geq \frac{\textit{burst rate}}{\textit{frequency}}$$

If a new burst rate, which is not compatible with those equations, is inserted, then new compatible frequency is counted and set up. However, if a new frequency, which is not compatible to those equations, is inserted, then nothing changes, and error message is printed.

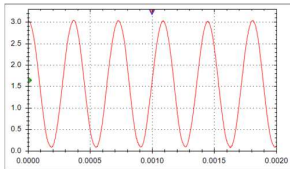
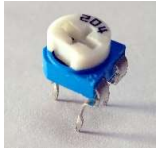
Please report any bugs at: pravdkat@fel.cvut.cz

Simple Laboratory Generators

Dual-Channel Generator

DCG

■ B.4 Uživatelský manuál Harmonického PWM generátoru



Sine Wave Generator

for microcontroller STM32G030J6

Kateřina Pravdov

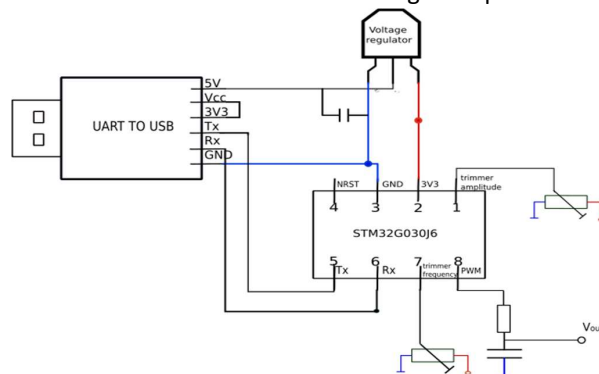
April 1, 2022

User Manual

Sine Wave generator is a generator for microcontroller STM32G030J6. It produces a PWM signal that can be filtered into the sine wave using a low pass filter. Frequency and amplitude can be modified by using UART communication at a baud rate of 115200 bits/s or by using trimmers. Little changes (voltage until circa $V_{cc}/20$) on trimmer are considered as noise. After change which is higher than $V_{cc}/20$ there is a 2 second interval in which the parameters can be set up more precisely – the voltage sensitivity is $V_{cc}/200$.

Frequency of the sine wave can be from 200 Hz to 11.712 kHz. Trimmer sets up frequency between 200 and 4600 Hz. The maximum amplitude of the signal is equal to the supply voltage of the microcontroller. The amplitude has 100 possible values. A simple passive RC Low Pass Filter can be made by connecting in series a Resistor with a Capacitor. Recommended values are: 20 nF, 500 Ω . As a serial terminal can be used Putty, Realterm and Xterm. Note that the microcontroller should be restarted after uploading the code.

► Parameters of the UART communication are 115200 bit/s, 8 bits of data, no parity, 1 stop bit, line ending: LF or CR. A UART-USB converter is needed and together with a voltage regulator can be used as a voltage source for the microcontroller. The wiring example is in the following picture:



► After opening serial terminal, following page is printed:

```

COM9 - PuTTY
Department of Measurement          FEL, CTU
K.Pravodva                         version 1.0

          Sine generator

                                type <> to set up:

Frequency: 2655 Hz                 <f-->
Real Freq: 2659 Hz                 <f-->
Level: 100                         <l-->

insert new command: █

```

- Type <f-number> to change the frequency
- Type <l-number> to change the amplitude

Please report any bugs at: pravdkat@fel.cvut.cz

Simple Laboratory Generators