



Středoškolská technika 2023

Setkání a prezentace prací středoškolských studentů na ČVUT

DIGITÁLNÍ GENERÁTOR

Jan Nevyhoštěný

SPŠE V Úžlabině
V Úžlabině 320, Praha 10

Čestné prohlášení

Odevzdáním této maturitní práce na téma **Digitální generátor** potvrzuji, že jsem ji vypracoval/a pod vedením vedoucího samostatně za použití v práci uvedených pramenů a literatury. Dále potvrzuji, že odevzdaná vytištěná verze písemné zprávy (protokolu) a plakátu se plně shoduje s odevzdanou elektronickou verzí.

V Praze dne.....

.....
Podpis autora/autorky práce

Anotace

Tato dlouhodobá maturitní práce se zabývá digitálním generátorem, který má sloužit jako nahrazení různých senzorů a vstupních signálů v testeru elektrických zařízení ve firmě Lubíno s.r.o. Důraz je kladen na kvalitu signálu a jeho nemožnost rušení okolím. Digitální generátor se musí vejít do 3U racku. Ovládán je z PC přes sběrnici RS-485 pomocí terminálu. Generátor musí umět sinusový průběh výstupního napětí a stejnosměrnou hodnotu.

Klíčová slova

FPGA, převodník, vzorkovací frekvence, signál

Annotation

This long term graduation thesis deals with a digital generator to replace various sensors and input signals in the electrical equipment tester at Lubíno s.r.o. The emphasis is on the quality of the signal and its impossibility of interference by the environment. The digital generator must fit into a 3U rack. It is controlled from a PC via RS-485 bus using a terminal. The generator must be able to provide a sinusoidal output voltage waveform and a direct current value.

Keywords

FPGA, converter, sampling frequency, signal

Obsah

1	ÚVOD A CÍL PRÁCE	5
2	PRINCIP A VÝBĚR SOUČÁSTEK	6
2.1	DIGITÁLNÍ ČÁST	6
2.1.1	RS-485.....	6
2.1.2	D/A převodník	6
2.1.3	FPGA	7
2.1.4	Ostatní digitální obvody.....	8
2.2	ANALOGOVÁ ČÁST	8
2.2.1	Zdroje.....	8
2.2.2	Filtr.....	10
2.2.3	Zesilovač.....	11
3	SOFTWARE.....	12
3.1	SPI PRO DAC.....	12
3.2	VÝPOČET HODNOT PRO DAC.....	13
3.3	PŘÍJEM RS-485.....	16
4	PRAKTICKÁ ČÁST	17
4.1	PROTOTYP.....	17
4.2	NÁVRH PLOŠNÉHO SPOJE	17
4.3	ČELNÍ PANEL.....	18
4.4	CHLADIČ.....	18
4.5	PÁJENÍ	18
4.6	OŽIVENÍ.....	18
4.7	FUNKCE V PRAXI.....	19
4.7.1	Spektrální analýza.....	19
4.7.2	Měření zatěžovací charakteristiky.....	20
4.7.3	Ověření přesnosti frekvencí	22
5	ZÁVĚR	23
6	SEZNAM POUŽITÝCH ZDROJŮ.....	24
7	SEZNAM PŘÍLOH.....	26

1 Úvod a cíl práce

Rád bych vás seznámil s mou maturitní prací Digitální generátor. Generátory rozdělujeme na analogové a digitální. Obvyklým pravidlem bývá, že analogový generátor je vždy přesnější a dokáže generovat vyšší frekvence. Nelze ho ale ovládat z počítače. Proto existují digitální generátory. Můj digitální generátor funguje na koncepci řídicího prvku, převodu digitálního signálu na analogový a následně jeho vyhlazení a zesílení. Generátor má být schopný generovat 50 kHz sinusový signál při deseti voltové amplitudě do zátěže 50 mA. Generátor navrhnu do rackové skříně s výškou 3U. Ideálně by měl být výstupem zcela čistý sinusový signál s jednou harmonickou složkou. Musím tedy vymyslet, jak bude generátor řízený, jaké součástky budou použity, navrhnout plošný spoj, naprogramovat řídicí prvek a celé zařízení zprovoznit. Následně pomocí harmonického analyzátoru ověřím, zda se blížím k analogovému sinusovému signálu.

2 Princip a výběr součástek

Generátor má přijmou digitální data z počítače za pomoci sériové komunikace RS-485, obsahem těchto digitálních dat jsou požadované parametry výstupního signálu. V generátoru tedy musí být řídicí prvek, já jsem zvolil hradlové pole FPGA, který tyto data zpracuje a pošle je do D/A převodníku, který podle těchto dat zrekonstruuje diskrétní sinusový signál, který se následně musí vyfiltrovat filtrem a uhladit tak, aby se co nejvíce podobal spojitému sinusovému signálu. Následně pro větší výstupní proud a větší amplitudy signálu, než dokáže vytvořit převodník, potřebuji zesilovač s rozkmitovým a koncovým stupněm. Generátor je napájen z 24 V AC transformátoru, který je součástí 3U racku, ve kterém bude provozován. 24 V AC se následně transformuje na potřebné napětí a výkony pro jednotlivé stabilizované zdroje, které jsou třeba pro napájení jednotlivých součástí generátoru, pomocí mnou navrženým transformátorem.

2.1 Digitální část

2.1.1 RS-485

Generátor je digitální z důvodu možnosti ovládní z PC, které je realizováno prostřednictvím sériové linky RS-485. Software pro posílání sériových dat je vyřešen aplikací v PC. Jako hardwarové rozhraní pro RS-485 jsem využil integrovaný obvod ADM483EARZ [6], který se ve firmě, se kterou spolupracuji, obvykle používá. Sériovou linku jsem také ochránil před přepětím pomocí integrovaného obvodu USB6B1 [7], který jsem vybral jako nejlepší možný a dostupný pro half-duplex komunikaci. Následně je sériová linka galvanicky oddělena pomocí optočlenů HCPL0531 [8].

2.1.2 D/A převodník

Pro převedení digitálních dat na schodový sinusový signál je potřeba využít digitálně/analogový převodník. Ideálně by měl převodník umět výstupní napětí ± 10 V. Musí být schopný přenastavit výstup během maximálně $1\mu\text{s}$, aby zvládl vzorkovací frekvenci $1,048576$ MHz. Také musí mít rozlišení alespoň 20 bitů pro dostatečnou jemnost výstupních hodnot. Z převodníků s těmito hodnotami jsem vybral DAC11001B [1], protože měl ze všech nejmenší odchylku při generování vzorku. Převodník také potřebuje výstupní buffer (sledovač). Ten jsem podle nabídky doporučených operačních zesilovačů v datasheetu vybral OPA189ID pro jeho malý napěťový offset. Převodník také potřebuje velmi přesné

referenční napětí, podle kterého nastavuje výstup. V datasheetu se také doporučuje filtr napájecí napětí.[1]

2.1.3 FPGA

Pro příjem dat z RS-485, výpočet okamžitých hodnot generovaného signálu a posílání vypočtených hodnot do D/A převodníku potřebuji řídicí prvek. Ten má za úkol posílat do D/A převodníku periodicky, data taková, aby vznikl sinusový signál s parametry, které přijme z RS-485.

Pro periodické posílání dat potřebujeme tzv. vzorkovací frekvenci, se kterou budeme obnovovat vzorek neboli velikost napětí v čase. Čím více vzorků budu mít na jednu periodu výstupního signálu, tím menší bude jeho zkreslení. Stanovil jsem si tedy 20 dílků na jednu periodu výstupního signálu. To znamená, že vzorkovací frekvence musí být alespoň 20krát větší než frekvence výstupního signálu, pro 50 kHz výstupní frekvenci budu potřebovat vzorkovací frekvenci minimálně 1 MHz. Frekvenci jsem si tedy stanovil na mocninu dvou neboli 1 048 576 Hz, aby se s touto frekvencí dalo dobře pracovat v hradlovém poli.

Dále musí řídicí prvek posílat data do D/A převodníku specifickým způsobem. Převodník, který používám komunikuje na sériové komunikaci s protokolem SPI. Abychom dosáhly požadované vzorkovací frekvence 1,048576 MHz, musí se podle datasheetu data posílat do převodníku rychlostí 38,5 MHz [1]. Data, která se posílají do D/A převodníku jsou velikosti napětí v čase takové, aby přetvořila schodový obrys sinusového signálu určité frekvence a amplitudy. Tyto hodnoty se musí počítat v řídicím prvku. Tento výpočet se musí vhodně naprogramovat.

Obvykle máme na výběr z řídicích prvků mezi mikroprocesory a hradlovými poli FPGA. FPGA má tu výhodu, že dokáže dělat téměř neomezený počet operací zároveň. Toto je velká výhoda pro výpočet a posílání hodnot do D/A převodníku. Mikroprocesor dělá operaci po operaci a nestíhal by zároveň posílat data do převodníku, přijímat data z RS-485 a už v tuto chvíli počítat další vzorek pro D/A převodník. Hradlové pole jsem vybral Spartan Xilinx 6 XC6SLX9, výkonově je značně předimenzované pro mé použití, ale potřebuji takto velké pole kvůli velikosti jeho blokové RAM paměti, ve které mám uschovanou tabulku hodnot pro výpočet. [3]

2.1.4 Ostatní digitální obvody

Paměť EEPROM

V této paměti se uchovává software pro hradlové pole FPGA. [5]

Signalizační LED

Určeny k signalizaci funkčnosti přístroje. Červená a zelená.

Přepínač adres

Určuje nastavení adresy přístroje, podle které se volají přístroje v testeru.

Watchdog

Určen k resetování hradlového pole, pokud by došlo k chybě v programu. Watchdog funguje tak, že se do něj minimálně několikrát za sekundu musí poslat signál, pokud se tak nestane, resetuje FPGA. Watchdog také pomocí děliče hlídá napájení z PWR.L na trafu [4].

2.2 Analogová část

2.2.1 Zdroje

Generátor má spoustu obvodů, které potřebují různé napájecí napětí.

Omezovače napětí

Omezovače napětí omezují napětí za usměrňovačem, které může být až 85 V, když není transformátor zatížen. Začal jsem tedy zvažovat řešení tohoto problému. Jako první jsem se zamyslel a došel k třem řešením. První a nejjednodušší řešení by byl integrovaný obvod lineárního stabilizátoru. Bohužel toto řešení nejde použít, protože na trhu v době realizace tohoto projektu nebyl žádný lineární stabilizátor, který by vydržel vstupní napětí 85 V a měl výstup 30 V. Druhým řešením bylo za pomoci tranzistorů MOSFET. Toto řešení nemělo žádné úskalí až na moji nezkušenost s tímto druhem tranzistorů. Proto jsem zvolil řešení třetí, které spočívá v Darlingtonově zapojení bipolárních tranzistorů se zenerovou diodou v bázi. Z tohoto omezeného napětí se napájí zesilovač a veškeré lineární stabilizátory.

Schéma viz příloha č.1

Lineární stabilizátory

Lineární stabilizátory omezují a stabilizují napájecí napětí pro napájení všech digitálních a analogových obvodů Digitálního generátoru. Lineární stabilizátory jsem vybíral podle ceny, výkonu a toho, jestli mají stejný stabilizátor pro záporné napětí stejné velikosti.

Potřebuji napětí ± 15 V pro napájení D/A převodníku a operačních zesilovačů pro filtry, výstupní buffer D/A převodníku, reference D/A převodníku a rozkmitový stupeň zesilovače. Pro toto napětí jsem zvolil pro tuto aplikaci často používané LM78L15ABD pro +15 V a LM79L15ABD pro -15 V. Ty jsou napájeny z výstupu omezovačů (2).

Dále potřebuji napětí +5 V pro integrovaný obvod sériové komunikace a pro určení hladiny SPI komunikace D/A převodníku. Pro +5 V jsem zvolil LM78L05ABD, které firma, pro kterou generátor vyrábím, měla skladem. Ten je napájen nízkonapět'ovou odbočkou z trafa (3).

Napětí +3,3 V pro rozhraní FPGA, napájení optočlenů, watchdogu flash paměti, adresového přepínače a LED diody. Pro +3,3 V jsem zvolil U37NCV4274ADT33, které také byly skladem ve firmě. Ten je také napájen nízkonapět'ovou odbočkou z trafa (3).

Napájecí napětí +1,2 V potřebuji už jen na napájení jádra FPGA hradlového pole. Stabilizátor pro +1,2 V jsem zvolil NCP585-12, který také byl skladem. Ten je napájen ze stabilizátoru +3,3 V (3).

Všechny stabilizátory byly vybírané s ohledem na výkon, který však u těchto obvodů bude velmi malý, takže jsem nebyl při výběru téměř omezován, a součástky, které byly skladem, vyhovovaly. Za každým stabilizátorem je elektrolytický filtrační kondenzátor, který funguje jako zdroj energie pro větší výkyvy v odběru za stabilizátorem a malé keramické blokovací kondenzátory, které filtrují vysokofrekvenční výkyvy odběru součástky, u které jsou umístěny. U každého stabilizátoru je také transil, ten funguje jako přepět'ová ochrana.

Zdroje referenčních napětí

D/A převodník potřebuje velmi přesné zdroje referenčních napětí. Od přesnosti a stability jejich velikostí se odvíjí i přesnost výstupního signálu z D/A převodníku. Pro moji aplikaci vyžadujeme referenční napětí +10 V a -10 V. Pro referenci +10 V jsem zvolil MAX6143AASA10+, toto je stabilizátor, který měl ve chvíli výběru nejlepší teplotní stabilitu a má možnost seřízení trimerem. Za stabilizátorem jsem použil doporučené zapojení

tří operačních zesilovačů OPA189ID, které využívají symetrických, teplotně závislých rezistorů uvnitř D/A převodníku. Operační zesilovač U14 je obraceč, který obrací +10 V na -10 V. Operační zesilovače U13 a U16 dorovnávají rozdíly mezi přesnými ± 10 V ze stabilizátoru nebo obraceče a hodnotou napětí v D/A převodníku [1].

Schéma viz. příloha č.4

Filtry napájecích napětí

Pro obvody, které jsou náchylné na šum jsem se rozhodl, že navrhnu filtry pro jejich napájecí napětí. Těmito obvody jsou D/A převodník, výstupní filtr a napájení obvodů sériové komunikace. Každý filtr je složen ze dvou dolních propustí, aby se šum nedostával ani do filtrovaného obvodu tak ani z něj. Použil jsem zapojení z datasheetu D/A převodníku [1].

Transformátor a usměrňovač

Transformujeme 24 V AC na vstupu zařízení na různá napětí, která potřebujeme pro další zpracování v obvodu. Transformátor má dva sloupky, na jednom je navinuté primární vinutí a na druhém jsou sekundární vinutí. Tím se nám přístroj opět galvanicky oddělí od zbytku světa. Potřebuji 2 sekundární vinutí, jedno na malé napětí a jedno na vyšší. Malé napětí jsem určil 6,8 V podle už jednoho hotového trafo, které mělo připojené podobné periferie. Vyšší napětí už nebylo tak jednoduché. Nejprve jsem potřeboval sekundární vinutí se středem, abych mohl usměrňovat dvoucestně a dostal kladné a záporné napětí. Bylo zapotřebí, aby při maximální zátěži celého generátoru bylo napětí za usměrňovačem ± 30 V. A to ještě při minimálním napětí v síti. Za každým dvoucestným usměrňovačem je filtrační kondenzátor, blokovací kondenzátory a přepěťová ochrana ve formě transilu. Za zařízení trafo musím poděkovat firmě. Já si jen změřil jejich trafo a řekl jsem si co změnit. Pracovník firmy následně podle mých požadavků objednal trafo ve firmě Tronic. viz. příloha č.5

2.2.2 Filtr

Výstupní filtr má za úkol odfiltrovat co nejvíce vyšších harmonických a vysokofrekvenčního šumu. Filtr je ta součást, kde se z diskretního výstupního signálu z D/A převodníku stane téměř spojitý signál. Proto je potřeba ho navrhnout co nejvyššího řádu, aby byl co nejstrmější a zároveň do takové mezní frekvence, aby výstupní signál byl co možná nejméně omezen. Hlavní frekvence, kterou je třeba brát jako potenciální šum, je vzorkovací frekvence D/A převodníku neboli 1,048 576 MHz.

Filtr jsem navrhoval podle programu Filter Pro od firmy Texas Instruments [17]. Chtěl jsem navrhnout co nejstrmější filtr, ale zároveň nesměl měnit své zesílení v propustném pásmu. Právě filtr koncepce Chebysev je nejstrmější ale má malý zákmit ještě v pásmu propusti. Nakonec nejlépe vyšel filtr koncepce Butterworth, je dost strmý a zároveň nezakmitává. Filtr jsem konstruoval na dvakrát. Poprvé na 50 kHz jako je v zadání a potom jsem experimentem zjistil, že dokážu generovat i 100 kHz bez větších problémů díky tomu, že filtr mohl být 6. řádu. Větší řád už nešel kvůli limitaci parametru operačních zesilovačů GBW. Na trhu už nebyly lepší operační zesilovače než mnou vybrané OPA211ID. Návrh filtru v příloze č.6. Při sestavení filtru jako prototypu došlo k jedinému problému, a to nedostatku úplně přesných součástek. Záleželo na každém nF u kondenzátorů. Po objednání přesných součástek bylo vše v pořádku a filtr fungoval naprosto bezchybně.

2.2.3 Zesilovač

Zvolení koncepce zesilovače byla velmi zdlouhavá věc, existují desítky různých zapojení pro mou aplikaci. Byl zapotřebí rozkmitový stupeň s velmi malým offsetem, aby byl výstupní signál naprosto přesný, a výkonový stupeň ve třídě AB, aby nevznikalo zkreslení. Také byla podmínka možnosti měnit zesílení, abych si mohl doladit výstup z převodníku.

Zvolil jsem koncepci zesilovače, kde se rozkmitový stupeň skládá z operačního zesilovače s malým offsetem, OPA189ID, a dvou komplementárních rozkmitových bipolárních tranzistorů DZT5401 pro kladnou půlvlnu a DZT5551 pro zápornou půlvlnu. Jako koncový stupeň jsem zvolil komplementární bipolární zesilovače 2SC4793 pro kladnou a 2SA1837 pro zápornou půlvlnu. Do jejich báze je připojen bipolární tranzistor BD139, kterým se nastavuje příčný proud koncového stupně. Poměrem odporů v bázi a emitoru u rozkmitových stupňů se nastavovalo, jak moc k napájení se budou tranzistory otevírat. Zesílení se nastavuje potenciometrem P3 ve zpětné vazbě. Při sestavování zesilovače mi zesilovač neustále kmital, pomohlo uzemnit chladič, ale pořád kmital. Až jako řešení byl kondenzátor C79, který fungoval jako dolní propust. Vyladil jsem jeho hodnotu tak aby celý zesilovač měl co největší šířku pásma a zároveň nekmital. Zesilovač spolehlivě zesiluje do 200 kHz a při rozkmitu ± 20 V a umí být zatížen až 230 mA. Inspiroval jsem se zapojením z amatérského rádia [2]. Měnil jsem hodnoty v amatérském rádiu dle mého informovaného odhadu a tím jsem došel k funkčnímu produktu.

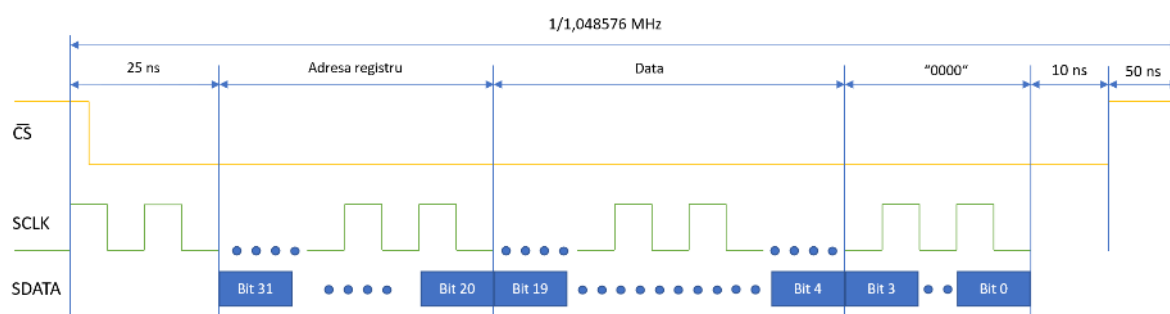
Schéma viz přílohy č.7

3 Software

V této kapitole vás seznámím s teoretickým provedením softwaru pro Digitální generátor.

3.1 SPI pro DAC

D/A převodník komunikuje pomocí sběrnice SPI. Máme 3 hlavní signály, CS – chip select, SCLK – Serial clock a SDATA – Serial data. Pokud dáme chip select do aktivní úrovně, u mého převodníku je CS s negací, to znamená do logické 0 začne komunikace. S každou náběžnou hranou se pošle jeden bit. Potom, co se odkomunikuje daný počet bitů, u nás 32 bitů, CS se nastaví do logické 1 a komunikace se uzavře. Prvních 8 bitů je adresa registru pro data, podle datasheetu hodnota "00000001". V dalších 20 bitech se posílají data, která představují jednu velikost výstupního napětí. Poslední 4 bity jsou "0000"[1]. Takto se to opakuje frekvencí 1,048 576 MHz.



Obrázek 1: Časy SPI, vlastní tvorba

Podle datasheetu jsem si našel časy, které musím dodržet pro spolehlivou funkčnost SPI komunikace. Jedna perioda trvá $1/1,048576 = 953,67431$ ns. Odečtu celkově 85 ns a vyjde mi, že na odeslání 32 bitů dat mi zbývá 868,64731 ns.

$$SCLK = \frac{1}{\frac{868,67431}{32}} = 36,8377 \text{ MHz}$$

SCLK tedy musí být alespoň 36,8377 MHz, aby se přenos mohl uskutečnit na takto vysoké vzorkovací frekvenci.

V datasheetu je maximální povolená hodnota SCLK 38,5 MHz. Tu jsem tedy využil. Tím mi zbývá hodně času navíc, který jsem využil pro ochranu přenosu mezi entitou výpočtu, která řídí chip select a entitou posílání dat po SPI. To dělám proto, že tyto dvě entity fungují na rozdílných hodinových signálech. Entita pro výpočet pracuje na MCLK (Master clock)

neboli 67,108864 MHz a entita pro posílání dat do převodníku funguje na 77 MHz, ty pak dělíme dvěma a z toho nám vznikne požadovaných 38,5 MHz. Jelikož tyto dva hodinové signály nejsou navzájem dělitelné násobkem dvěma, musí být fázově posunuté. Proto necháváme alespoň dva hodinové cykly na začátku přenosu jako spolehlivou synchronizaci.

```

process (CLK, CLR)
begin
    if CLR = '1' then Pocitadlo <= "000000";           -- nastav výchozí hodnoty registrů do 0
        SCLK <= '0';
        OutShiftReg <= x"00000000";
    elsif rising_edge (CLK) then
        if nSYNC = '1' then                             -- pokud je CS neaktivní tak nuluj počítadlo
            Pocitadlo <= "000000";
            SCLK <= '0';
        else
            Pocitadlo <= Pocitadlo + 1;                 -- hodiny nech v nule aby si převodník nemyslel,
            -- že chceš komunikovat
            -- pokud CS je aktivní přičítej 1
        end if;
        if Pocitadlo = 1 then                             -- při prvním přičtení stusť komunikaci
            SCLK <= '1';
            OutShiftReg <= DataToDAC;
        end if;
        if Pocitadlo > 1 and Pocitadlo < 66 then         -- odpočítej 32 clocku
            SCLK <= not Pocitadlo(0);                 -- dělím frekvenci 77 MHz na 38,5 MHz
            if Pocitadlo(0) = '0' and Pocitadlo > 2 then -- počkáním až na třetí clock splním časovou podmínku
                OutShiftReg <= OutShiftReg(30 downto 0) & '1'; -- Shift registr, posílá data do převodníku
            end if;
        end if;
    end if;
end process;
end;

```

Obrázek 2: Program pro SPI, vlastní tvorba

3.2 Výpočet hodnot pro DAC

Výpočet hodnot byla asi nejnáročnější část na vymyšlení. V hradlovém poli se hodně obtížně pracuje s desetinnými čísly a už vůbec jr skoro nemožně dělit jinak než mocninou dvěma. Při programování ve VHDL pro FPGA si člověk musí představit vše ve dvojkové soustavě na rozdíl například od programování v C++ nebo arduinu, kde funguje desítková soustava. Výpočet musel vzít zadané parametry, podle nich vypočítat hodnoty pro D/A převodník. Opět jsou tu 2 koncepce. Jelikož počítám sinusový průběh, potřebuje počítat hodnotu sinu v čase nebo mít tabulku, kterou budu násobit hodnotou z PC. Zvolil jsem možnost s tabulkou, protože výpočet sinu v reálném čase by zabíral spoustu klopných obvodů v hradlovém poli a pro mě jako začátečníka s VHDL by bylo velmi obtížné tento způsob naprogramovat. Následně jsem musel vyřešit, jak budu vybírat frekvenci. Nejjednodušší řešení bylo udělat paměť vzorků, do které se bude vstupovat s každým vzorkem, když bude frekvence 1 Hz, bude brát v paměti každý vzorek, než přeteče po 1 sekundě, když 1000 Hz, bude brát každý tisící vzorek atd...

Za chodu mě napadlo, že bych zkusil vylepšit parametry generátoru, proto jsem si řekl, že zkusím počítat s rozkmitem ± 20 V. Nastavení výstupu jsem chtěl mít po 1 mV, to znamenalo,

že rozsah amplitudových hodnot z PC je 0-20000 mV. Rozsah D/A převodníku je 0-1048576 dílků, to znamená, že abych kmital sinusovým signálem okolo 0, musel jsem přičíst půlku rozsahu tzv. 524288 dílků ke každému výpočtu. Také jsem narazil na problém, že s čísly, která měla na pozicích jednotek, desítek, stovek atd... různá čísla, špatně počítalo. Proto jsem se rozhodl, že budu k půlce rozsahu přičítat +500 000 nebo -500 000 dílků. Zároveň se mi ale změnil rozsah převodníku na 24288–1024288 To znamená, že vzorec v tuto chvíli vypadá takto:

$$d = \frac{p \cdot \text{tabulka}}{x} + 524288$$

$p = 0-20\ 000$, x a tabulku hledáme. Už víme, že x musí být mocnina dvojky a zlomek na pravé straně se musí rovnat 500 000 při $p = 20\ 000$, tím nám vychází rozumné hodnoty. Tabulka ± 25600 a $x = 1024$. Dělení 1024 = odříznutí posledních 10 bitů výsledku násobení. Do konečného vzorce ještě přidáme offset:

$$d = \frac{p(0 : 20\ 000) \cdot \text{tabulka} (\pm 25\ 600)}{1024} + 524288 + \text{offset}(\pm 20\ 000)$$

Pak přišel čas na výběr vzorku z tabulky. Z datasheetu jsem se dozvěděl, že paměť vzorků jde zkonfigurovat na 32 bloků po 18 bitech dlouhém vzorku. To je 32768 hodnot. Na tolik hodnot stačí adresování 15 bity. Prvních 5 bitů adresuje jeden z 32 bloků a zbylých 10 bitů na adresování vzorku uvnitř bloku. Vzal jsem tedy horních 15 bitů z 20bitového čítače, který čítá na vzorkovací frekvenci, a tím byl výběr vzorků z tabulky vyřešen. [3]

Jak jsem psal, je potřeba z tabulky brát hodnoty ve stejný čas a 1048576krát za sekundu neboli obnovovací frekvence. Toho jsem docílil tak, že celý hlavní stavový automat, který má za úkol čekat na hodnoty přijaté z RS-485, vypočítat hodnoty a operovat s CS v periodickém čase podle vzorkovací frekvence. Ten jsem naprogramoval jako tzv. zvonkohru, 6bitový čítač, který mi dělí frekvenci 67,108864 MHz na požadovaných 1,048576 MHz. Podle stavu čítače pak provádím jednotlivé stavy ve stavovém automatu.
Příloha č.9

```

when 0 => if MCLKCounter = 0 then
            DataToDAC(31 downto 0)      <= CONFIG1;
            State <= 1;
            CS <= '0';
            nSYNC <= '0';
        end if;
when 1 => if MCLKCounter = 59 then
            CS <= '1';
            nSYNC <= '1';
            State <= 2;
        end if;
when 2 => if MCLKCounter = 0 then
            DataToDAC(31 downto 0)      <= CONFIG2;
            State <= 3;
            CS <= '0';
            nSYNC <= '0';
        end if;
when 3 => if MCLKCounter = 59 then
            CS <= '1';
            nSYNC <= '1';
            Amplituda <= x"0000";
            OffsetPC <= x"0000";
            State <= 4;
        end if;

```

Obrázek 3: Stavový automat pro konfiguraci D/A převodníku, vlastní tvorba

Na obrázku vidíme, že automat nejdříve pošle konfigurační data do D/A převodníku. Nastavil zapisování do registru, vypnul teplotní stabilizaci, zapnul režim rychlého přechodu a nastavil rozsah referencí 20 V, CONFIG1 <= X"02000200". Podle strany 35 v datasheetu jsem nastavil zapisování do registru, vypnutí funkce TH a nastavení vzorkovací frekvence na 1,05 MHz, což je mých 1,048576 MHz, při tomto nastavení nám již datasheet říká, ať použijeme 38,5 MHz SCLK což je více než jsem v teorii spočítal, že je potřeba, takže tam bude rezerva, CONFIG2 <= X"060000D0".

3.3 Příjem RS-485

Příjem dat z RS-485 byla jedna z jednodušších věcí, jen se muselo nastavit, kolik bajtů se má posílat. Taky se musí posílat synchronizační bit o tom, že entita příjmu RS-485 přijala nová data a přijímat bit o potvrzení příjmu další entitou. Aby byl přenos spolehlivý, přidal jsem ještě kontrolní součet. Sečtou se postupně všechny bajty a vezme se posledních 8 bitů z tohoto součtu, a to se pošle z PC jako poslední bajt. Tento součet se provede i v hradlovém poli a pokud souhlasí s přijatými daty, pracuje dál. Pokud někde nastala chyba a neposlala se všechna data nebo se poslala špatně, s daty se dále nepracuje a čeká se na další.

```
process (CLK, CLR, RX_STROBE) begin
  if CLR = '1' then Stav <= "0000";
  elsif rising_edge (CLK) then
    if Spocital = '1' then -- bitem Spocital si nuluji bit Pocitej abych ho mohl nastavit vzdy kdyz prijdu nova data
      Pocitej <= '0';
    end if;
    if RX_STROBE = '1' then
      case Stav is
        when "0000" => if RX_DATA = x"55" then -- synch znak
          Stav <= "0001";
        end if;
        when "0001" => RXData(55 downto 48) <= unsigned(RX_DATA); -- AMP
          Stav <= "0010";
        when "0010" => RXData(47 downto 40) <= unsigned(RX_DATA); -- AMP
          Stav <= "0011";
        when "0011" => RXData(39 downto 32) <= unsigned(RX_DATA); -- FREQ
          Stav <= "0100";
        when "0100" => RXData(31 downto 24) <= unsigned(RX_DATA); -- FREQ
          Stav <= "0101";
        when "0101" => RXData(23 downto 16) <= unsigned(RX_DATA); -- FREQ
          Stav <= "0110";
        when "0110" => RXData(15 downto 8) <= unsigned(RX_DATA); -- OFFSET
          Stav <= "0111";
        when "0111" => RXData(7 downto 0) <= unsigned(RX_DATA); -- OFFSET
          Stav <= "1000";
        -- kontrolni soucet
        when "1000" => if unsigned(RX_DATA) = RXData(7 downto 0) + RXData(15 downto 8) + RXData(23 downto 31)
          DataRecieved <= std_logic_vector(RXData); -- posli data do hlavni entity FPGA
          Pocitej <= '1'; -- nastav bit ze jsou nove data pro prijem v FPGA
        end if;
        Stav <= "0000";
        when others => NULL;
      end case;
    end if;
  end if;
end process;
```

Obrázek 4: příjem bajtů z RS-485

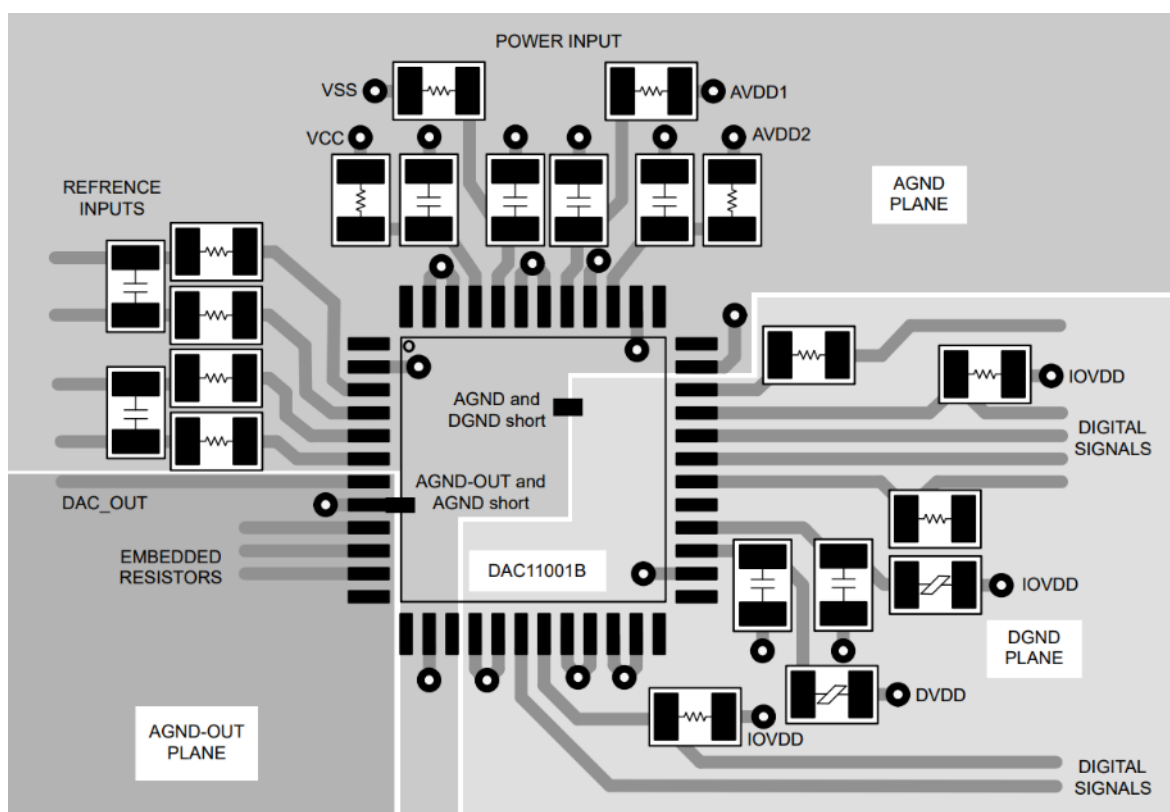
4 Praktická část

4.1 Prototyp

Začínal jsem s deskou, na které bylo hradlové pole, a k té jsem si postupně připojoval co jsem potřeboval. Jako první jsem začal s D/A převodníkem a jeho referenčními a napájecími napětími. Nejlepší mi přišlo začít programováním, protože dokud nebude fungovat převodník tak se okolo něj špatně staví hardware. Začal jsem tedy programováním. Po základním naprogramování jsem začal řešit obvody omezovačů, lineárních stabilizátorů a výrobu traťů, po zadání výroby traťů jsem sestavil filtr a zesilovač. Potom co jsem si ověřil funkci všech obvodů, pustil jsem se do kreslení desky.

4.2 Návrh plošného spoje

Návrh plošného spoje jsem konzultoval s firmou, pro kterou je generátor vyráběn. Doporučili mi rozlité země pro co nejmenší indukčnost spojů k zemi. Našel jsem si tedy podle datasheetu D/A převodníku, jak nejlépe propojit země, a z toho jsem vycházel.



Obrázek 5: doporučené zapojení země [1]

Následně jsem si udělal i rozlité napájení 3,3 V a 1,2 V. K tomuto všemu mi stačily 4 vrstvy plošného spoje. Celkový rozměr desky plošného spoje měl být 100 mm na výšku a 160 mm

na délku. Celkově návrh plošného spoje zabral okolo 60 h. Plošný spoj jsem si nechal vyrobit u firmy PCB Benešov. Výroba trvala týden a povedlo se na poprvé. Návrh viz. příloha č.10

4.3 Čelní panel

Čelní panel slouží k usazení do rackové vany a možnosti přišroubování přístroje. Jsou v něm uloženy LED diody pro signalizaci. Technický výkres viz příloha č.11

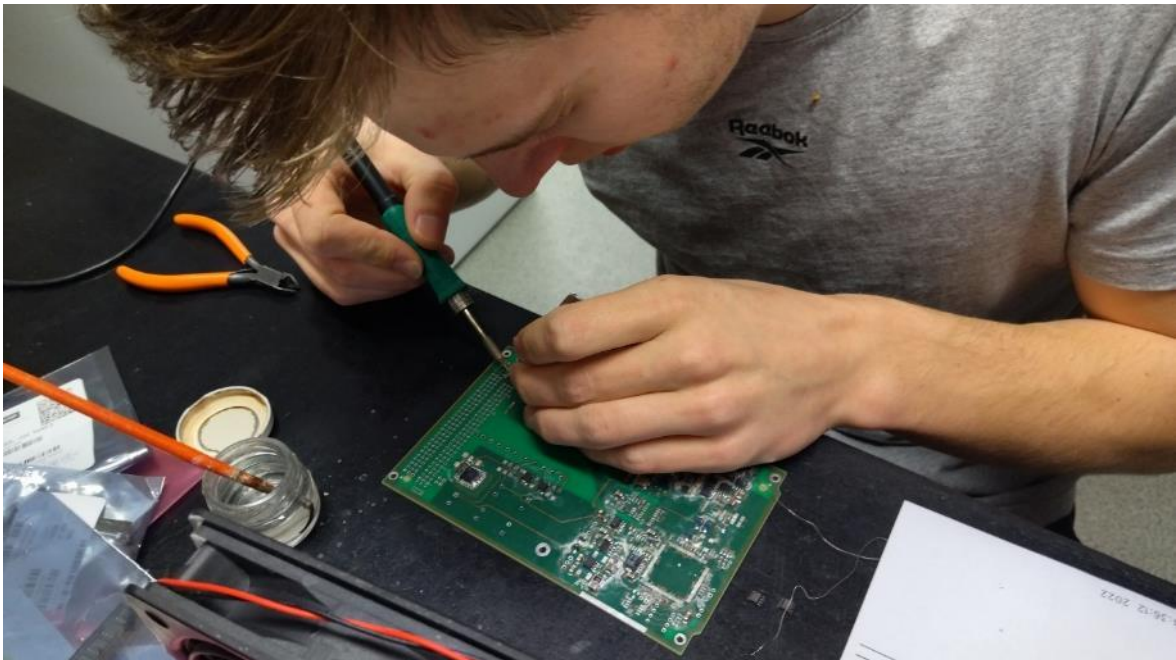
4.4 Chladič

Použil jsem chladič a seříznul jsem ho o 4 mm. Následně jsem si do něj vyvrtal závity pro omezovače a zesilovač. [21]

4.5 Pájení

Po doručení všech součástek jsem začal generátor sestavovat.

Začal jsem pájením. Většina součástek je SMD, tedy velmi malé. Nejmenší pouzdra, co se dělají jsou označována 0603. Pájel jsem i tato malá pouzdra pomocí tavidla a tenčí pájkou o průměru 0,2 mm. Postupoval jsem tedy od nejmenšího po největší. Po zapájení jsem začal oživovat.



Obrázek 6: pájení, vlastní tvorba

4.6 Oživení

Při oživení nenastaly žádné větší problémy a všechna napětí souhlasila s teorií.

4.7 Funkce v praxi

Generátor jsem tedy vložil do 3U racku a zkusil s ním komunikovat, generátor poslouchal a fungoval tak, jak bylo určeno. Po intenzivnějším měření jsem dosáhl maximálních parametrů:

Tabulka 1: porovnání naměřených a zadaných hodnot, vlastní tvorba

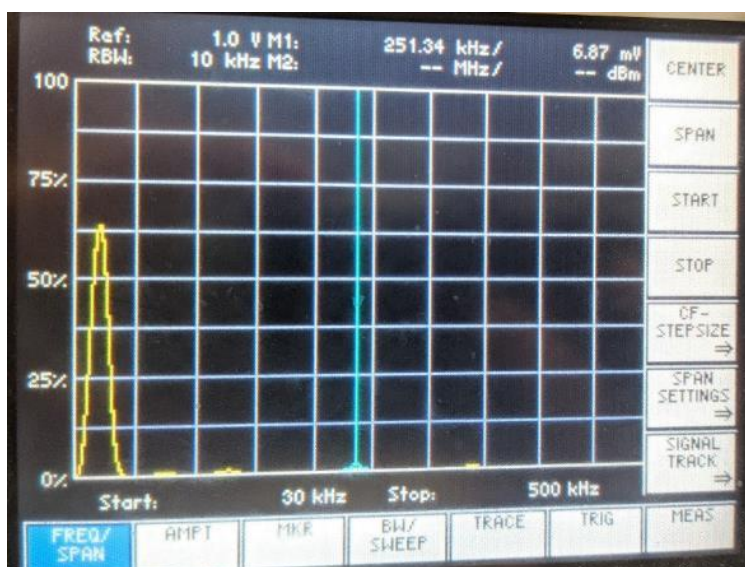
	zadání	dosažené
f	50 kHz	100 kHz
U	10 V	22,3 V
I	50 mA	230 mA

4.7.1 Spektrální analýza

Jako ověření kvality sinusového signálu jsem provedl spektrální analýzu pomocí spektrálního analyzátoru Rohde & Schwarz FS300, kterým jsem měřil u nás ve škole v laboratoři měření. Měření proběhlo od 30 kHz do 500 kHz a jeho výsledky:

Tabulka 2: Výsledky harmonické analýzy, vlastní tvorba

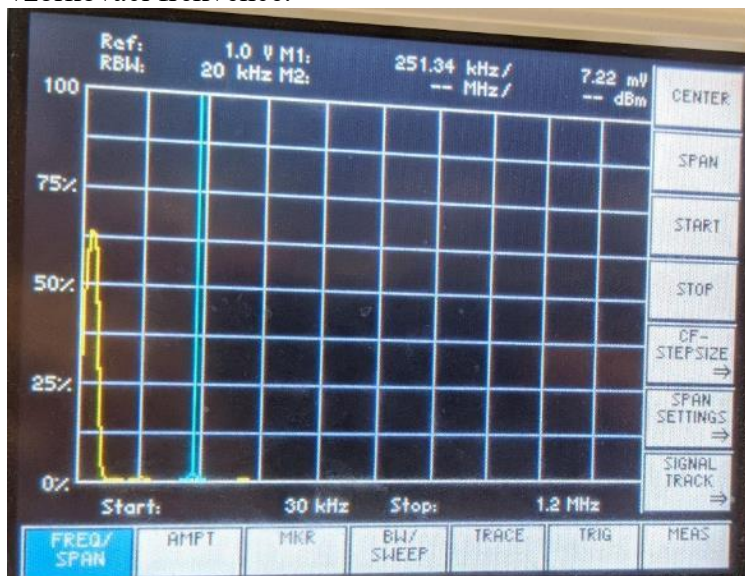
Harmonická	Velikost (%)
První	100
Druhá	0,89
Třetí	1,41
Čtvrtá	0
Pátá	1,1



Obrázek 7: Měření spektra, vlastní tvorba

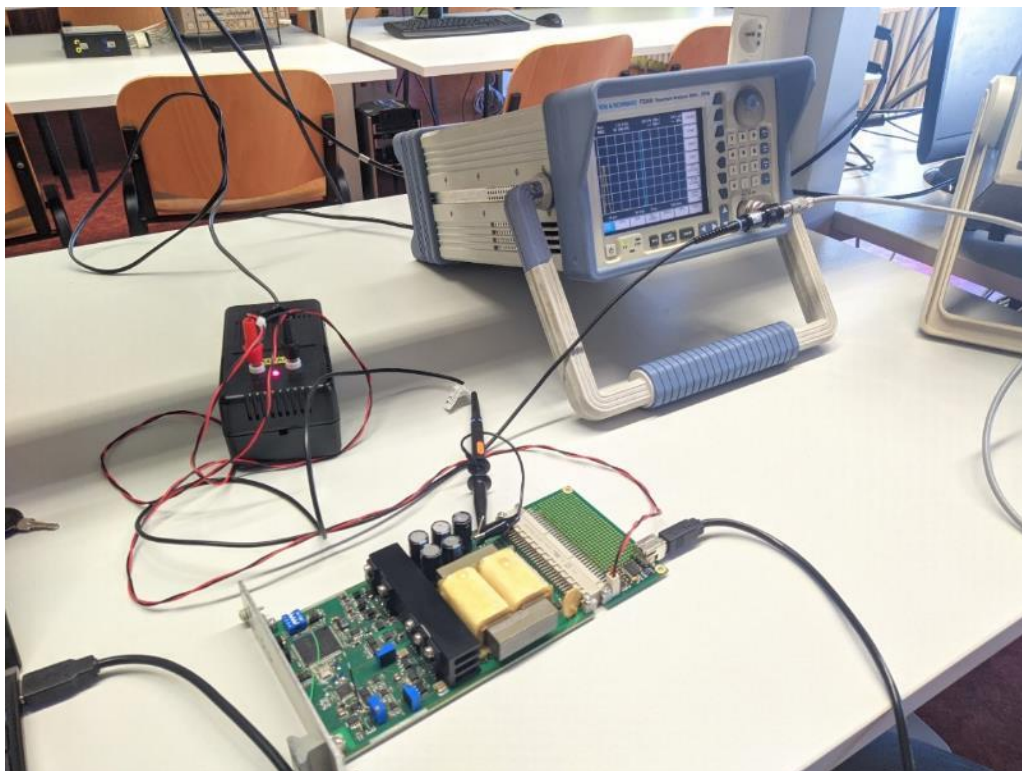
Výsledky měření byly nadmíru uspokojivé. Signál nemá žádné další harmonické, které by byly dost velké na to, aby se podílely výstupním signálu.

Následně jsem měřil do 1,2 GHz, abych si ověřil, že v signálu není už zmiňovaná vzorkovací frekvence.



Obrázek 8: měření velikosti spektrální čáry vzorkovací frekvence

Vzorkovací frekvence není vůbec vidět a ani nebyla změřitelná, což byl cíl.



Obrázek 9: pracoviště pro měření spektrálním analyzátořem, vlastní tvorba

Měření probíhalo při výstupním signálu o frekvenci 50 kHz a 10 V amplitudou.

4.7.2 Měření zatěžovací charakteristiky

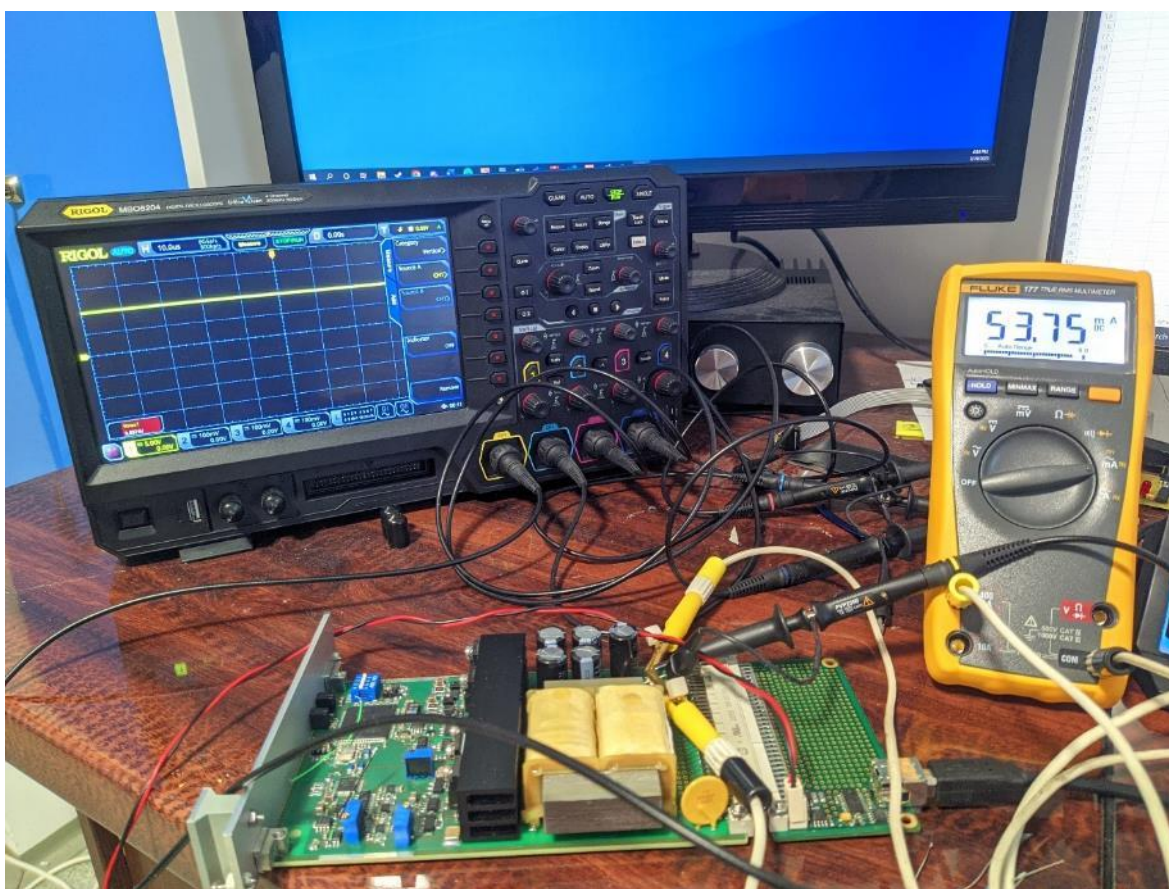
Při měření zatěžovací charakteristiky bylo výstupní napětí +10 V stejnosměrných.

Měřil jsem od 10 mA až do 50 mA.

Měřil jsem velmi přesným multimetrem Fluke 177. Při měření jsem používal odpory z řady E12, proto nejsou proudy přesně po 10 mA jak bych chtěl.

Tabulka 3: Měření zatěžovací charakteristiky, vlastní tvorba

I _z	9,77	20,74	29,34	35,97	44,1	53,75
U _{out}	10	10	10	10	10	10



Obrázek 10: Měření zatěžovací charakteristiky, vlastní výroba

Kontrolní výpočet:

$$I_z = \frac{U_{out}}{R_z} = \frac{10\text{ V}}{220\ \Omega} = 45,45\text{ mA}$$

Rozdíl mezi vypočteným a naměřeným proudem může způsobit tolerance rezistoru nebo úbytky na kabelech.

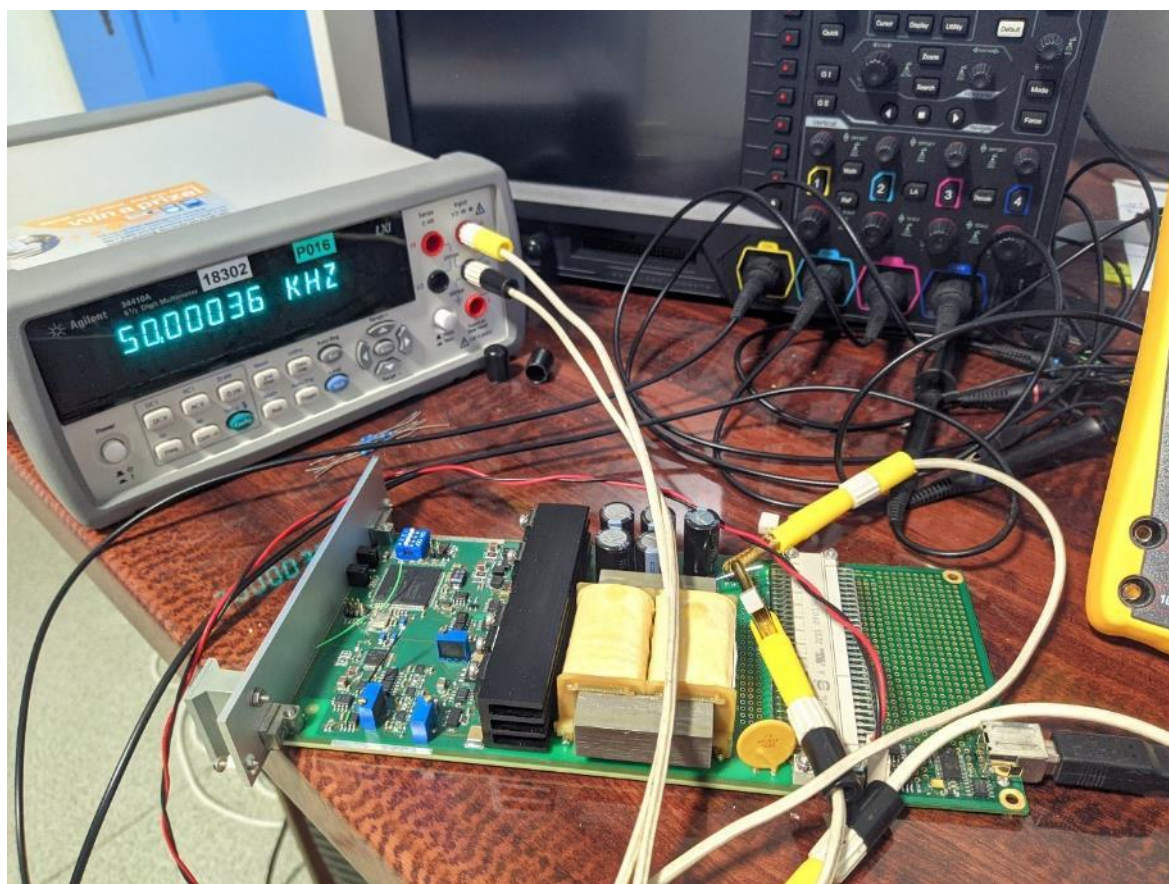
4.7.3 Ověření přesnosti frekvencí

Při měření frekvencí jsem měl výstupní amplitudu 5 V.

Frekvence jsem měřil pomocí velmi přesného stolního multimetru Agilent 34410A.

Tabulka 4: Ověření výstupních frekvencí, vlastní tvorba

frekvence z PC (Hz)	50 000	5000	500	50	5
Naměřená frekvence (Hz)	50 000,36	5000,035	500,003	50,00021	5,0102



Obrázek 11: Ověření přesnosti frekvencí

5 Závěr

Cílem práce bylo vynalezení a zkonstruování digitálního generátoru, který bude ovládán pomocí PC. Tento generátor jsem konstruoval pro firmu Lubíno s.r.o., která požadovala, aby se digitální generátor vešel do rackové vany 3U, měl maximální výstupní frekvenci alespoň 50 kHz a výstupní sinusové napětí o amplitudě 10 V. Musel jsem navrhnout řešení generátoru, napsat software pro generátor a nakreslit desku plošného spoje.

Toto byly stanovené cíle na začátku práce, postupně jak šel čas, firmu napadlo, že by se jim hodilo, aby generátor uměl výstupní frekvenci 100 kHz a 20 V amplitudu. To mi naštěstí nezkomplikovalo práci, protože jsem byl ve fázi programování, ještě před návrhem filtru a zesilovače, které by byly ovlivněny touto změnou.

Řešení generátoru jsem vymyslel velmi dobře, správně jsem začal hradlovým polem FPGA a co nejlepším D/A převodníkem. Koncepti filtru a zesilovače jsem také zvolil správně, a tudíž jsem přišel ke zdárnému konci a měl jsem dobrý základ pro splnění cílů.

Software byla asi nejnáročnější část celého projektu. Musel jsem se naučit v úplně novém jazyce VHDL a napsat složitý stavový automat, který musí fungovat za každých podmínek. To se mi povedlo asi po třech měsících práce, ve kterých jsem programování věnoval zhruba 210 h.

Návrh plošného spoje byl vcelku jednotvárný, trval dlouho, zhruba 80 h a hodně informací mi poskytla firma. Bylo to můj první obsáhlý plošný spoj, u kterého jsem musel dbát na přesnost zařízení. Plošný spoj se povedl a mohl jsem tedy používat generátor v praxi.

Přinesl jsem tedy generátor do firmy, kde se zapojil do 3U racku a začala zkouška. Na ovládání použili jejich vlastní program. Generátor reagoval okamžitě a fungoval tak, jak firma chtěla. Aktuálně se generátor nevyužívá, protože bude odevzdaný ve škole a bude tam až do mé obhajoby před odbornou komisí. Poté bude generátor používán v testeru elektrických zařízení a bude přinášet velké pohodlí při testování.

Vývoj generátoru firma financovala a celková částka vyšla na cca 35 000 Kč. Celková práce mi trvala zhruba 400 h. Práce byla zdařilá a bylo splněno zadání.

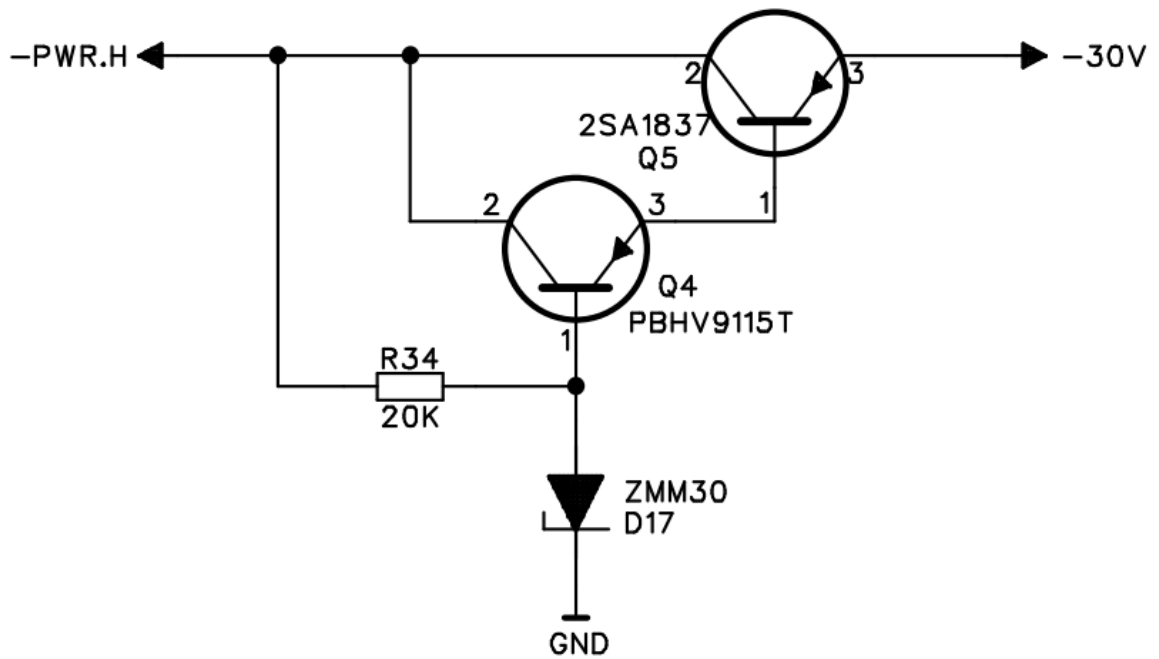
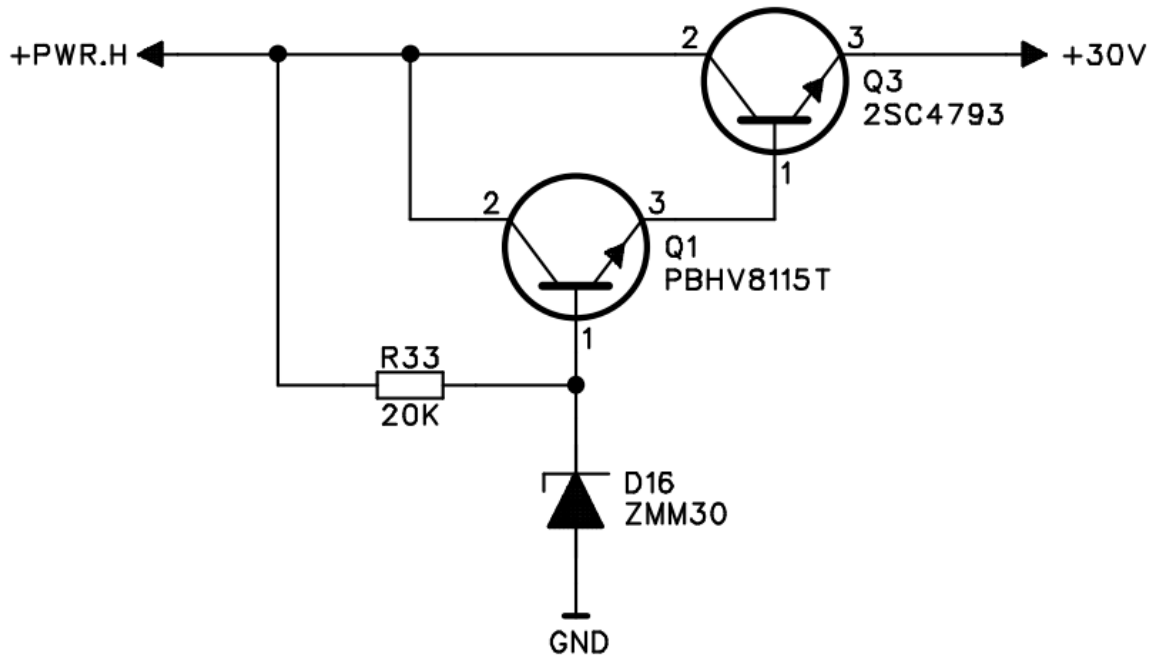
6 Seznam použitých zdrojů

- [1] TI.com: Datasheet DAC11001B [online]. Dallas: Texas Instruments, 2020 [cit. 2023-03-19]. Dostupné z: https://www.ti.com/lit/ds/symlink/dac11001a.pdf?ts=1679259572337&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FDAC11001A
- [2] Praktická elektronika - Amatérské radio. Amatérské rádio: Řada B - Pro konstruktéry. Praha: Amaro, 2002, 2011-, 51(5), 10-11. ISSN 1211-3557.
- [3] Docs.xilinx.com: Datasheet FPGA [online]. San Jose: Xilinx, 2015 [cit. 2023-03-19]. Dostupné z: <https://docs.xilinx.com/v/u/en-US/ds162>
- [4] Analog.com: Datasheet Watchdog [online]. Wilmington: Analog devices, 2016 [cit. 2023-03-19]. Dostupné z: https://www.analog.com/media/en/technical-documentation/datasheets/adm706p_706r_706s_706t_708r_708s_708t.pdf
- [5] Atmel: Datasheet EEPROM [online]. San Jose: Atmel, 2010 [cit. 2023-03-19]. Dostupné z: <https://datasheet.octopart.com/AT45DB041D-SU-2.5-Atmel-datasheet-9816548.pdf>
- [6] Analog.com: Datasheet RS-485 [online]. Wilmington: Analog Devices, 2007 [cit. 2023-03-19]. Dostupné z: <https://www.analog.com/media/en/technical-documentation/datasheets/adm483e.pdf>
- [7] St.com: Datasheet USB6B1 [online]. Geneva: STMicroelectronics, 2007 [cit. 2023-03-19]. Dostupné z: <https://www.st.com/resource/en/datasheet/usb6b1.pdf>
- [8] Datasheet.octopart.com: Datasheet optočleny [online]. Belmont: Jameco Electronics, 2004 [cit. 2023-03-19]. Dostupné z: <https://datasheet.octopart.com/HCPL-0531-Avago-datasheet-7276901.pdf>
- [9] Laro.com.pl: Datasheet tranzistor A1837 [online]. Minato city: Toshiba, 2004 [cit. 2023-03-19]. Dostupné z: <https://www.laro.com.pl/pdf/2sa1837.pdf>
- [10] Mouser.com: Datasheet tranzistor SC4793 [online]. Minato city: Toshiba, 2006 [cit. 2023-03-19]. Dostupné z: https://www.mouser.com/catalog/specsheets/toshiba_2SC4793.pdf

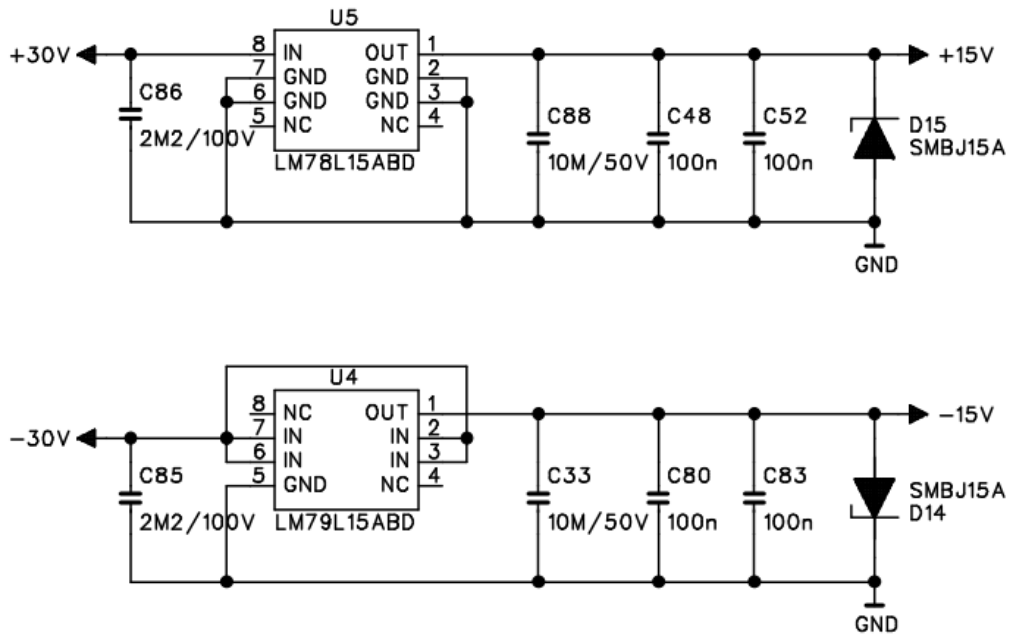
- [11] Ti.com: Datasheet pro lin. stabilizátory 5, -15, +15 V [online]. Dallas: Texas Instruments, 2020 [cit. 2023-03-19]. Dostupné z: https://www.ti.com/lit/ds/symlink/lm78l.pdf?ts=1679239422585&ref_url=https%253A%252F%252Fwww.google.com%252F
- [12] Onsemi.com: Datasheet Lin. stabilizátor +3,3 V [online]. Phoenix: Onsemi, 2013 [cit. 2023-03-19]. Dostupné z: <https://www.onsemi.com/pdf/datasheet/ncv4274-d.pdf>
- [13] Onsemi.com: Datasheet Lin. stabilizátor +1,2 V [online]. Phoenix: Onsemi, 2009 [cit. 2023-03-19]. Dostupné z: <https://www.onsemi.com/pdf/datasheet/ncp585-d.pdf>
- [14] Analog.com: Datasheet reference +10 V [online]. Wilmington: Maxim Integrated, 2004 [cit. 2023-03-19]. Dostupné z: <https://www.analog.com/media/en/technical-documentation/data-sheets/MAX6143.pdf>
- [15] Ti.com: OZ OPA189ID [online]. Dallas: Texas Instruments, 2021 [cit. 2023-03-19]. Dostupné z: https://www.ti.com/lit/ds/symlink/opa189.pdf?ts=1679240812649&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FOPA189
- [16] Webench.ti.com: Návrh Filtru [online]. Dallas: Texas Instruments, 2022 [cit. 2023-03-19]. Dostupné z: <https://webench.ti.com/filter-design-tool/filter-type>
- [17] Diodes.com: Tranzistor DZT5401 [online]. Plano: Diodes Incorporated, 2017 [cit. 2023-03-19]. Dostupné z: <https://www.diodes.com/assets/Datasheets/DZT5401.pdf>
- [18] Diodes.com: Tranzistor DZT5551 [online]. Plano: Diodes Incorporated, 2012 [cit. 2023-03-19]. Dostupné z: <https://www.diodes.com/assets/Datasheets/ds31219.pdf>
- [19] St.com: Tranzistor BD139 [online]. Geneva: ST Microelectronics, 2008 [cit. 2023-03-19]. Dostupné z: <https://www.st.com/resource/en/datasheet/cd00001225.pdf>
- [20] Farnell.com: Chladič [online]. Tábor: Fischer elektronik, 2010 [cit. 2023-03-19]. Dostupné z: <https://cz.farnell.com/fischer-elektronik/sk-454-100-sa/led-heatsink-standard-extruded/dp/1850028#anchorTechnicalDOCS>

7 Seznam příloh

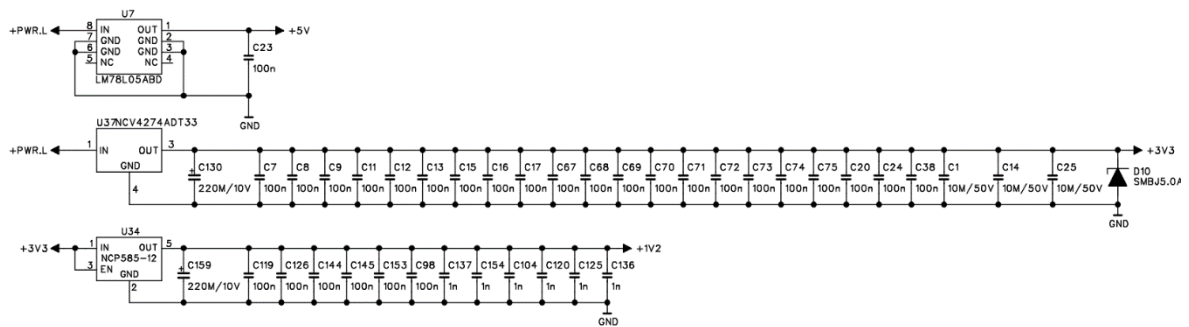
Příloha č. 1: Schéma omezovačů



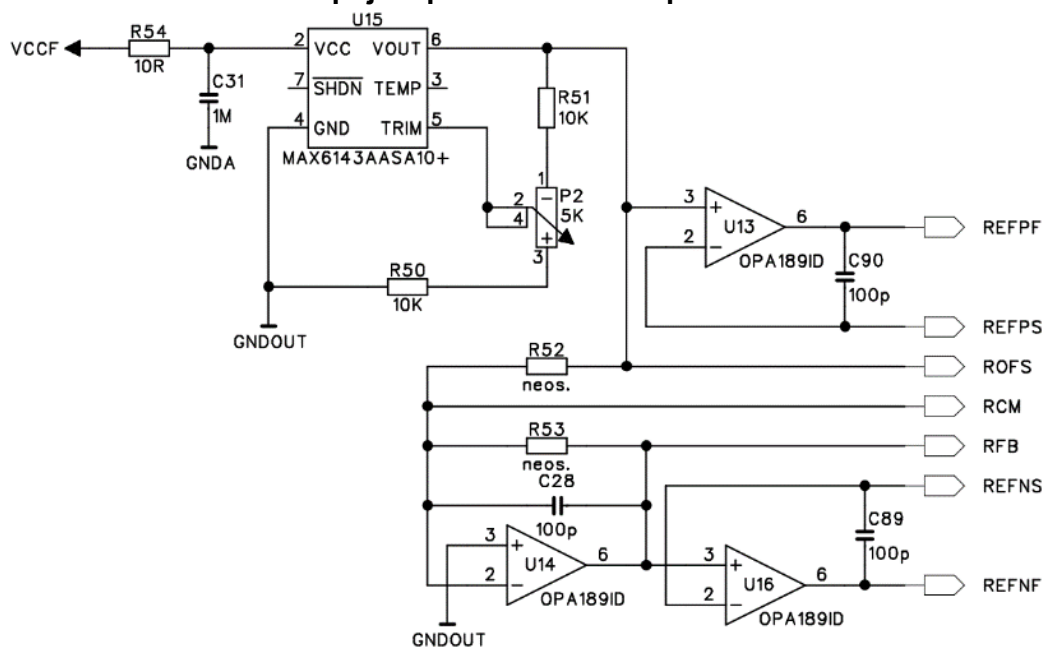
Příloha č. 2: Schéma lineárních omezovačů ±15 V



Příloha č. 3: Schéma lineárních stabilizátorů +5 V, +3,3 V a +1,2 V



Příloha č. 4: schéma zapojení pro referenční napětí



Příloha č. 5: objednání transformátoru

Dobrý den,

objednávám expresní výrobu jednoho vzorku transformátoru.

Bude odvozen z typu 2001626, na kterém provedete pouze změnu sekundárního vinutí takto:

- hlavní vinutí 2x40V na pinech 16-18-20, odběr veškerého výkonu,
- odbočky hlavního vinutí 2x8V na pinech 17-18-19, odběr zanedbatelný,
- vinutí tedy může být celé stejným drátem dimenzovaným na odběr veškerého výkonu při 80 V mezi piny 16 a 20.

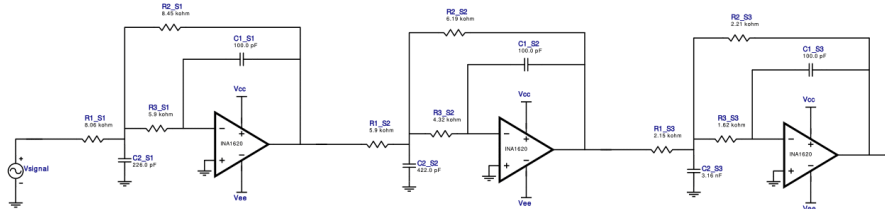
Vše ostatní zůstane beze změny (primár 28V piny 7-10, slabší plechy 0,35mm, rozměr kostry UI39/13, atd.).

Příloha č. 6: Návrh filtru

Type : Lowpass
 Response : Butterworth
 Order : 6
 Number of Stages : 3

Filter Design Report

Design : Lowpass Filter - 6th order Butterworth
 Design ID: 2



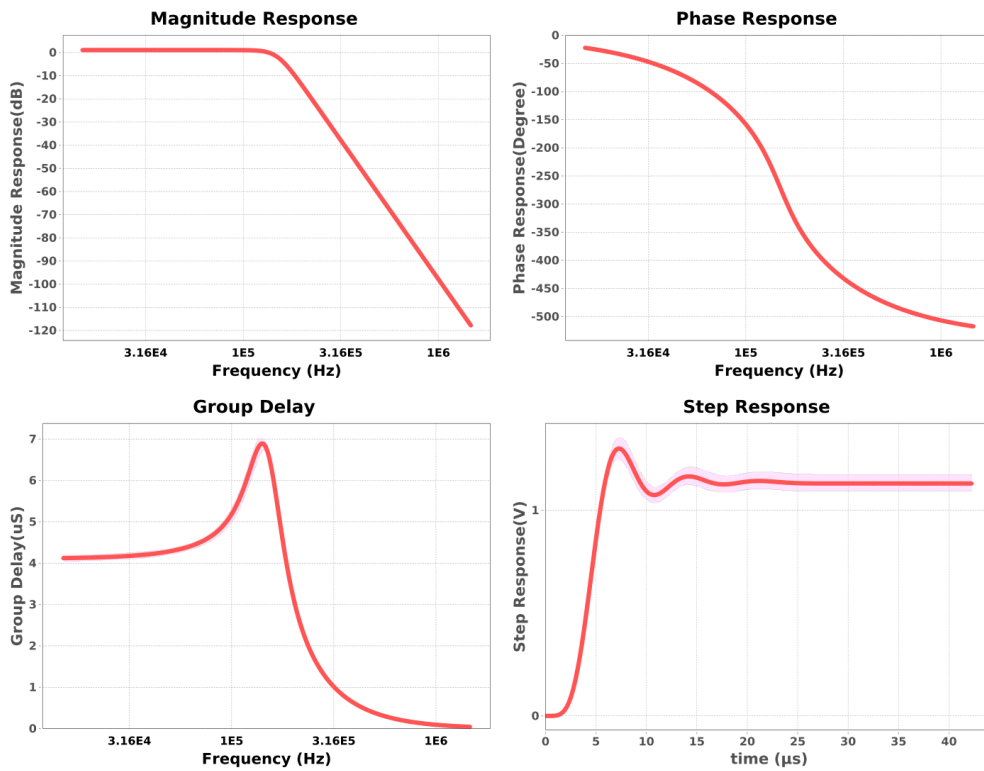
Electrical BOM

#	Name	Manufacturer	Part Number	Properties	Qty
1.	A1_S1	Texas Instruments Inc.	INA1620	GbwTyp= 32MHz VccMax= 36V VccMin= 4V	1
2.	A1_S2	Texas Instruments Inc.	INA1620	GbwTyp= 32MHz VccMax= 36V VccMin= 4V	1
3.	A1_S3	Texas Instruments Inc.	INA1620	GbwTyp= 32MHz VccMax= 36V VccMin= 4V	1
4.	C1_S1	Generic	Ideal	Cap= 100.0 pF Tolerance= 2.0 %	1
5.	C1_S2	Generic	Ideal	Cap= 100.0 pF Tolerance= 2.0 %	1
6.	C1_S3	Generic	Ideal	Cap= 100.0 pF Tolerance= 2.0 %	1
7.	C2_S1	Generic	Ideal	Cap= 226.0 pF Tolerance= 2.0 %	1
8.	C2_S2	Generic	Ideal	Cap= 422.0 pF Tolerance= 2.0 %	1
9.	C2_S3	Generic	Ideal	Cap= 3.16 nF Tolerance= 2.0 %	1
10.	R1_S1	Generic	Ideal	Res= 8060.0ohm Tolerance= 1%	1
11.	R1_S2	Generic	Ideal	Res= 5900.0ohm Tolerance= 1%	1
12.	R1_S3	Generic	Ideal	Res= 2150.0ohm Tolerance= 1%	1
13.	R2_S1	Generic	Ideal	Res= 8450.0ohm Tolerance= 1%	1
14.	R2_S2	Generic	Ideal	Res= 6190.0ohm Tolerance= 1%	1
15.	R2_S3	Generic	Ideal	Res= 2210.0ohm Tolerance= 1%	1
16.	R3_S1	Generic	Ideal	Res= 5900.0ohm Tolerance= 1%	1

#	Name	Manufacturer	Part Number	Properties	Qty
17.	R3_S2	Generic	Ideal	Res= 4320.0ohm Tolerance= 1%	1
18.	R3_S3	Generic	Ideal	Res= 1620.0ohm Tolerance= 1%	1

Sensitivity Analysis

#	Name	Series	Tolerance
1.	Cap	E48	2%
2.	Res	E96	1%



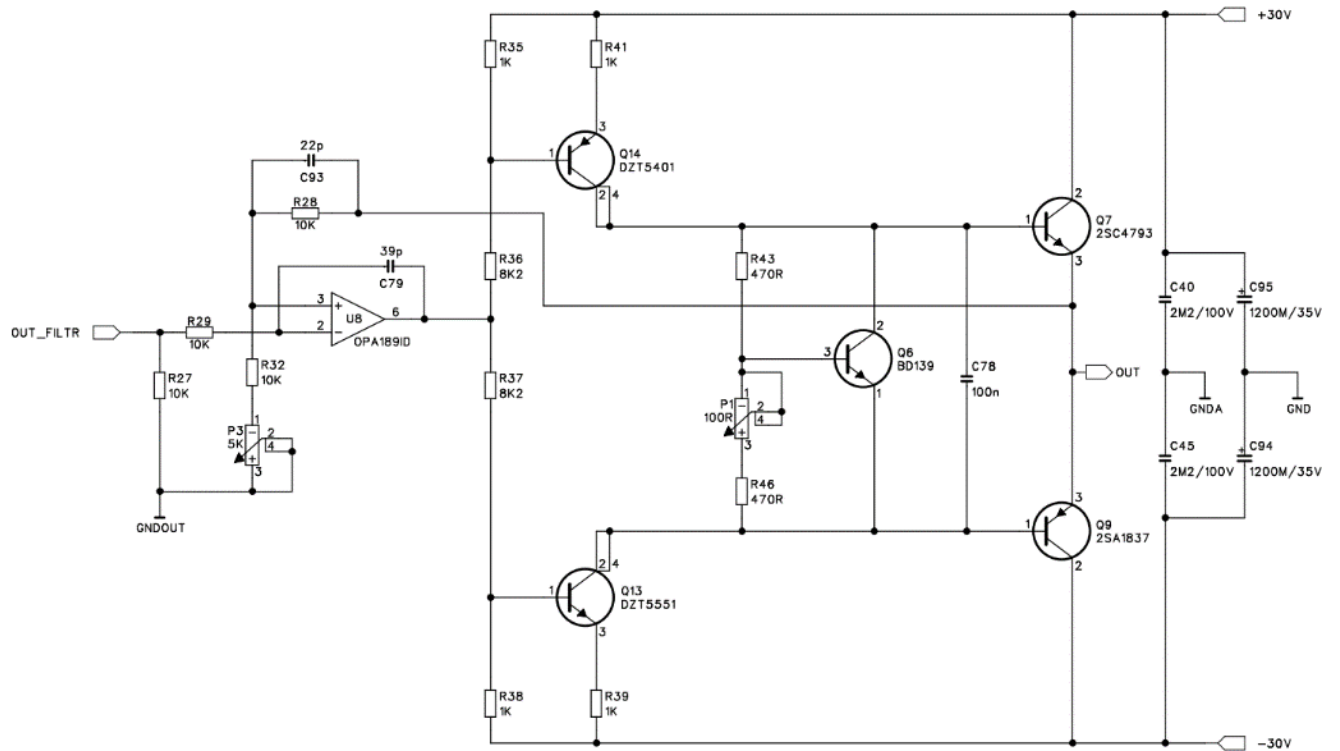
Design Inputs

#	Name	Value	Description
1.	FilterType	lowpass	
2.	FilterResponse	Butterworth	
3.	FilterOrder	6.0	
4.	FilterTopology	Multiple Feedback	
5.	NumberOfStages	3.0	
6.	PassbandFrequency	150.0 k	
7.	StopbandAttenuation	-120.001	
8.	StopbandFrequency	1.5 M	
9.	Gain	1.122	
10.	DualSupply	+/-5.00 V	Power supply(s) to active chips
11.	ResistorTolerance	E96	Resistor series - 1% Passive resistor tolerance
12.	CapacitorTolerance	E48	Capacitor series - 2% Passive capacitor tolerance

Design Assistance

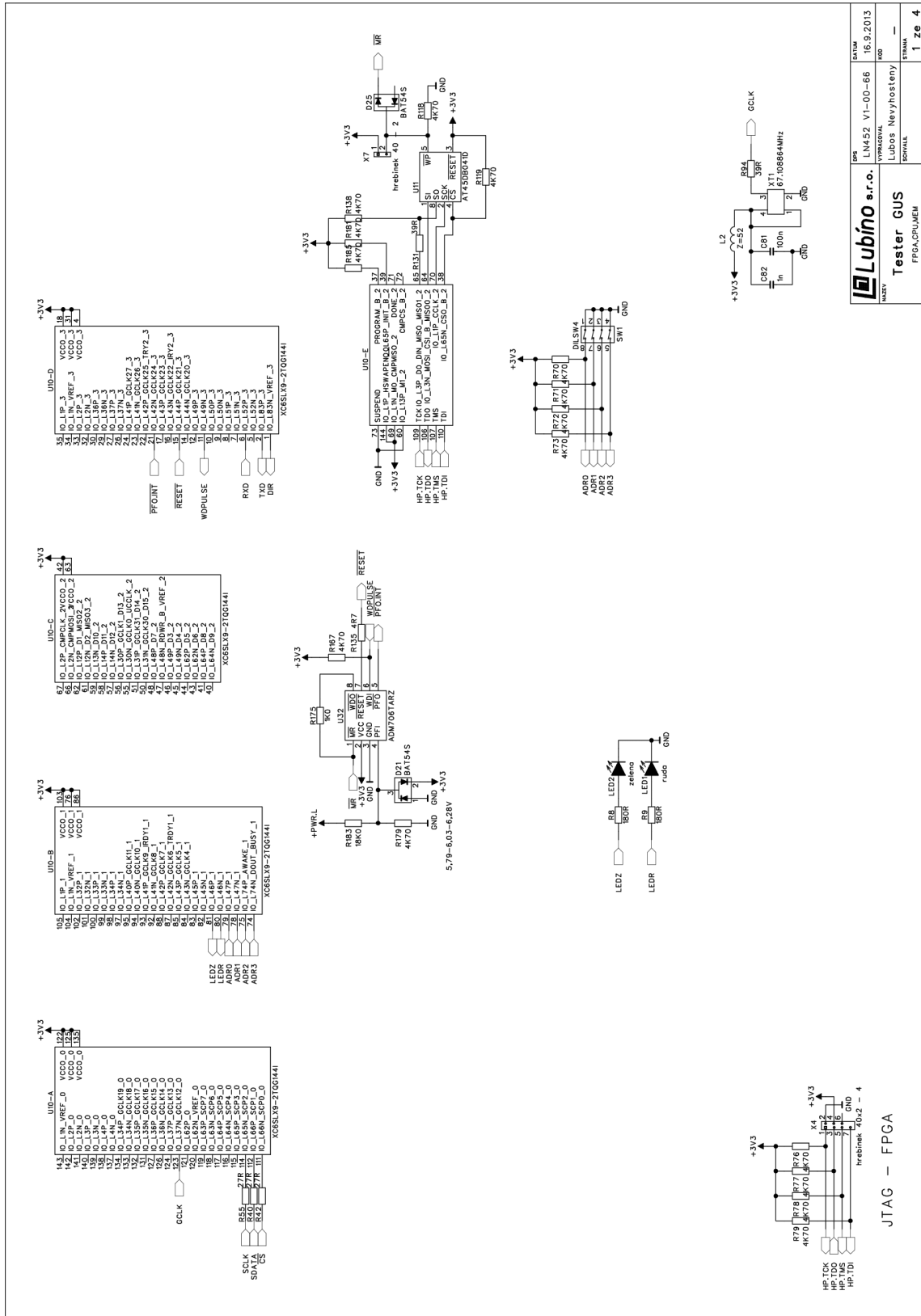
1. **INA1620** Product Folder : <http://www.ti.com/product/INA1620> : contains the data sheet and other resources.

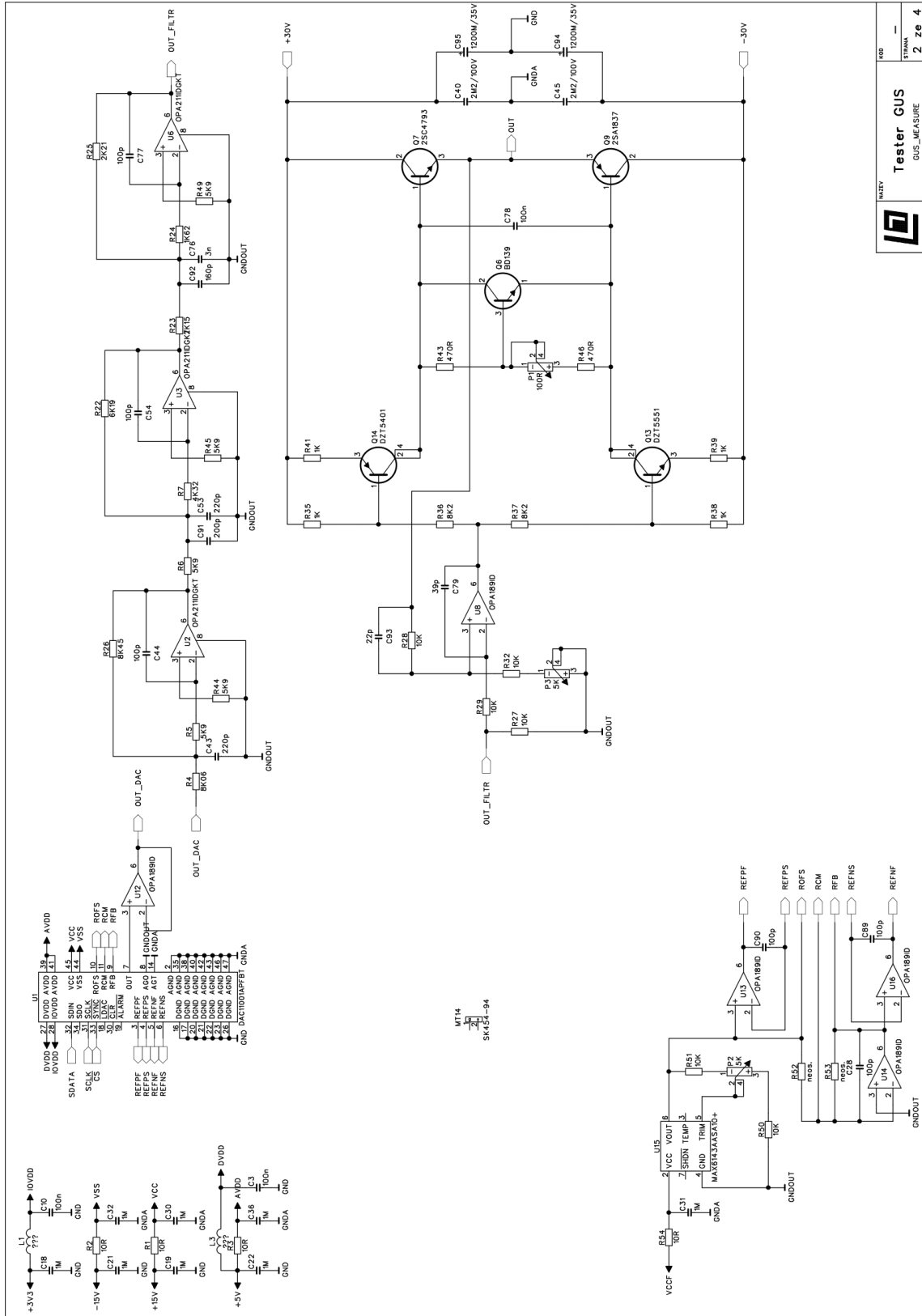
Příloha č. 7: Schéma zesilovače



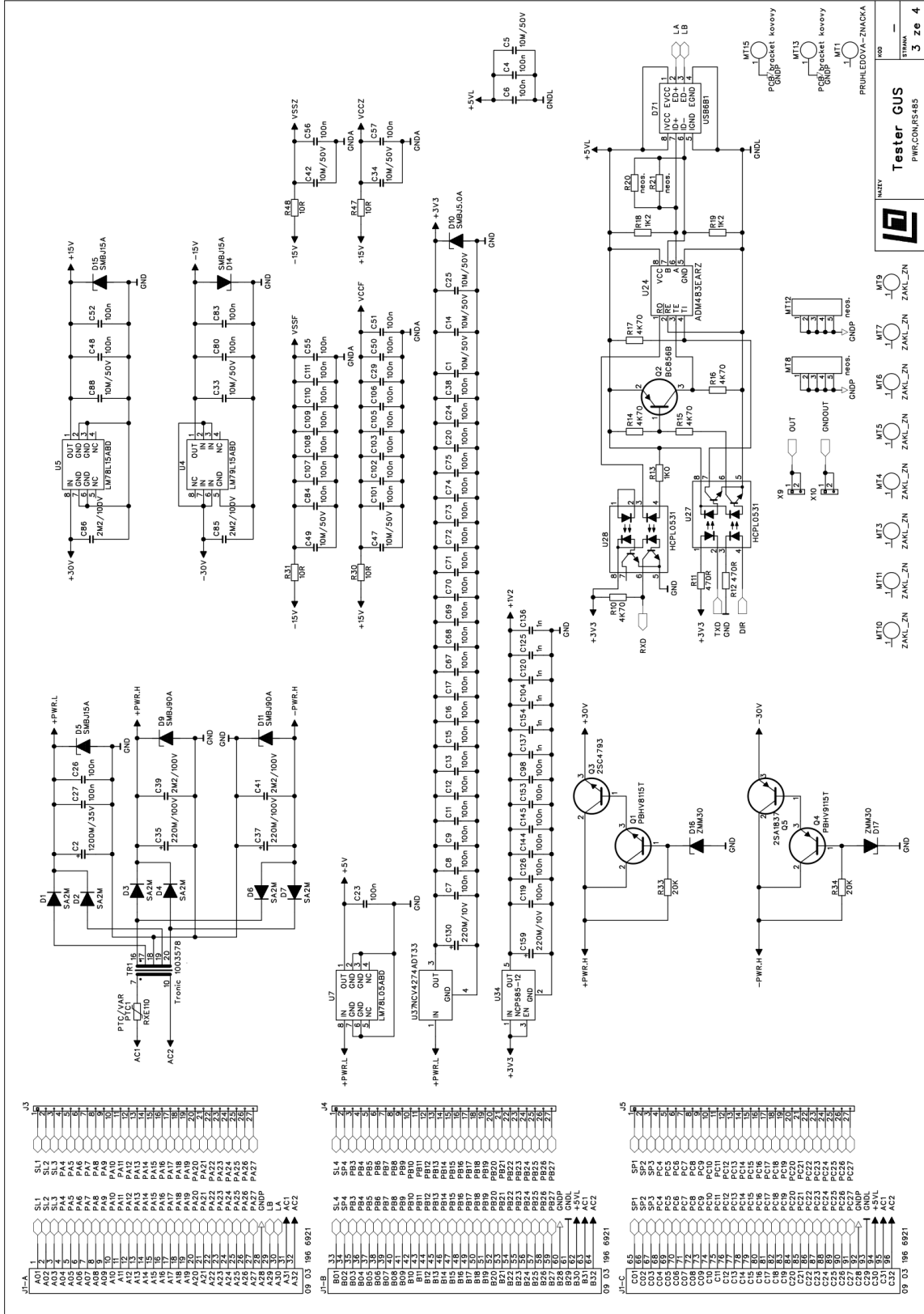
Příloha č. 8: Schéma celého zařízení

JN-645 Generátor V1-00-51.sch-1 - Sun Mar 19 12:15:27 2023






Tester GUS
 GUS_MEASURE
 KOD: —
 STRANA: 2 ze 4



Tester GUS
PWRCOINRS4B5

MT10 ZAKL_ZN
MT11 ZAKL_ZN
MT12 ZAKL_ZN
MT13 ZAKL_ZN
MT14 ZAKL_ZN
MT15 ZAKL_ZN
MT16 ZAKL_ZN
MT17 ZAKL_ZN
MT18 ZAKL_ZN
MT19 ZAKL_ZN

X8
X9
X10
X11
X12
X13
X14
X15
X16
X17
X18
X19
X20
X21
X22
X23
X24
X25
X26
X27
X28
X29
X30
X31
X32
X33
X34
X35
X36
X37
X38
X39
X40
X41
X42
X43
X44
X45
X46
X47
X48
X49
X50
X51
X52
X53
X54
X55
X56
X57
X58
X59
X60
X61
X62
X63
X64
X65
X66
X67
X68
X69
X70
X71
X72
X73
X74
X75
X76
X77
X78
X79
X80
X81
X82
X83
X84
X85
X86
X87
X88
X89
X90
X91
X92
X93
X94
X95
X96
X97
X98
X99
X100

MT10 ZAKL_ZN
MT11 ZAKL_ZN
MT12 ZAKL_ZN
MT13 ZAKL_ZN
MT14 ZAKL_ZN
MT15 ZAKL_ZN
MT16 ZAKL_ZN
MT17 ZAKL_ZN
MT18 ZAKL_ZN
MT19 ZAKL_ZN

09 03 196 6921

Příloha č. 9: Hlavní stavový automat

```

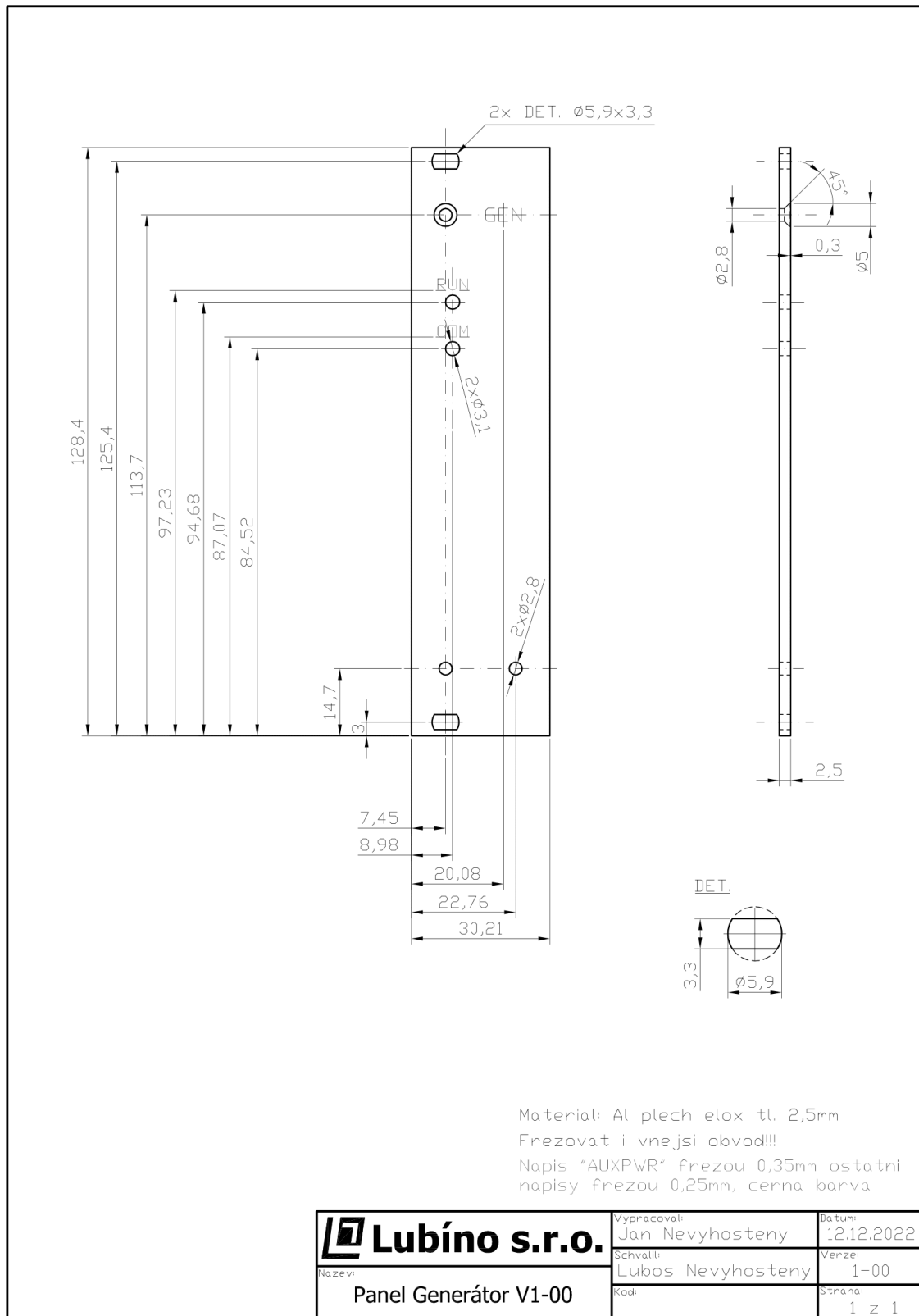
case State is
when 0 => if MCLKCounter = 0 then
            DataToDAC(31 downto 0)      <= CONFIG1;
            State <= 1;
            CS <= '0';
            nSYNC <= '0';
        end if;
when 1 => if MCLKCounter = 59 then
            CS <= '1';
            nSYNC <= '1';
            State <= 2;
        end if;
when 2 => if MCLKCounter = 0 then
            DataToDAC(31 downto 0)      <= CONFIG2;
            State <= 3;
            CS <= '0';
            nSYNC <= '0';
        end if;
when 3 => if MCLKCounter = 59 then
            CS <= '1';
            nSYNC <= '1';
            Amplituda <= x"0000";
            OffsetPC <= x"0000";
            State <= 4;
        end if;
;
when 4 => if Pocitej = '1' then
            Amplituda <= signed(Parametr(55 downto 40));
            OffsetPC <= signed(Parametr(15 downto 0));
            State <= 5;
            Spocital <= '1';
        else
            State <= 5;
        end if;
when 5 => if MCLKCounter = 0 then
            CS <= '0';
            nSYNC <= '0';
            Fcounter <= Fcounter + Parametr(35 downto 16); -- posun se v tabulce
            State <= 6;
        end if;
when 6 => AdresaRAM <= Fcounter(19 downto 5);
            State <= 7;
            Spocital <= '0';
when 7 => if MCLKCounter = 40 then
            Tabulka <= signed(RAMData);
            Offset <= OffsetPC * 50;
            State <= 8;
        end if;
when 8 => Vysledek <= Tabulka * Amplituda;
            State <= 9;
when 9 => Konecna <= Vysledek(29 downto 10) + 524288 + Offset(19 downto 0); -- dělení 10 bity (1024)
            State <= 10;
when 10 => DataToDAC(31 downto 24) <= x"01";
            DataToDAC(23 downto 4) <= std_logic_vector(Konecna(19 downto 0)); -- data
            DataToDAC(3 downto 0) <= x"0"; -- 4 bity v 0
            State <= 11;
when 11 => if MCLKCounter = 59 then
            LEDR <= '0';
            CS <= '1';
            nSYNC <= '1';
            State <= 4;
        end if;
when others => NULL;

```

Příloha č. 10: Plošný spoj

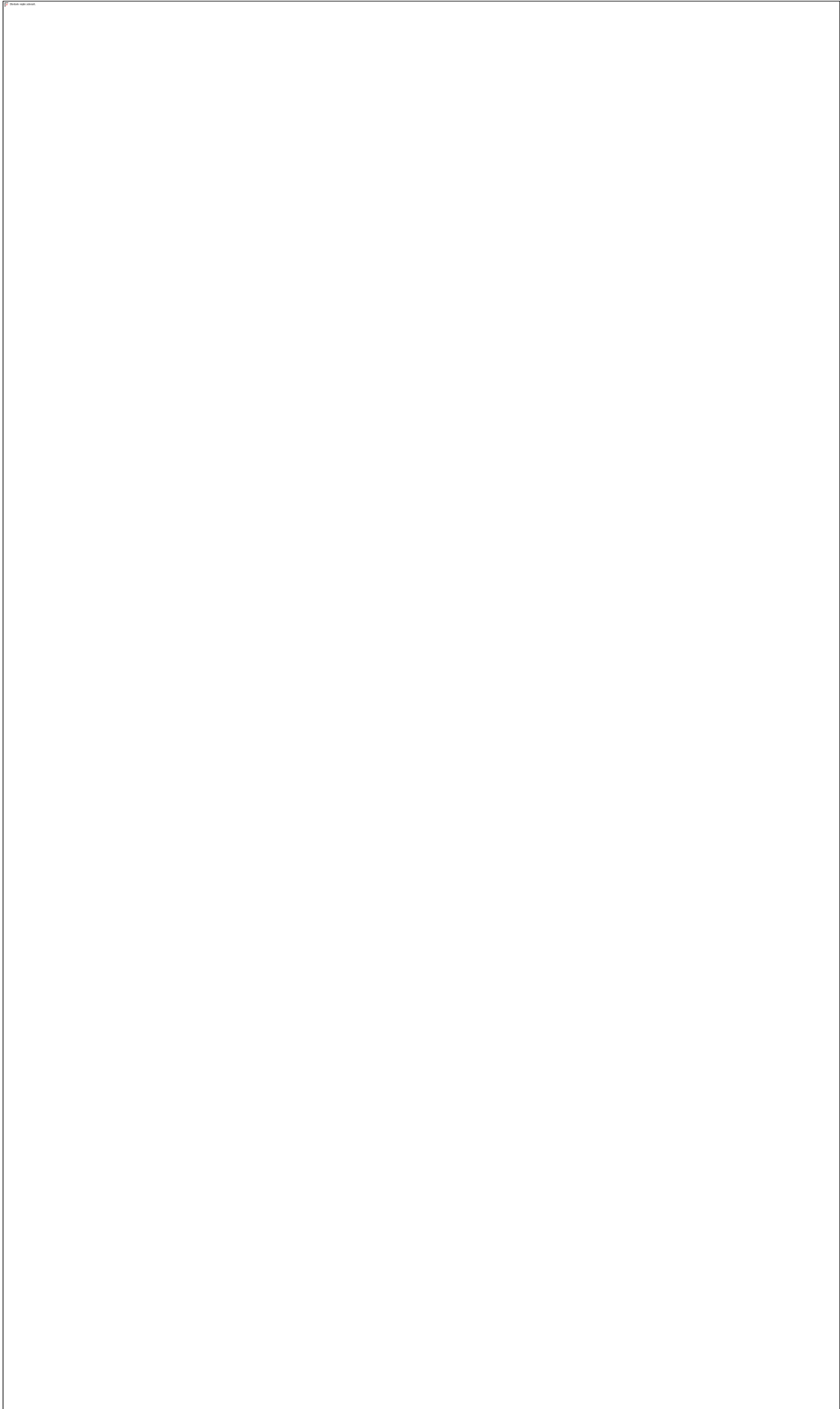


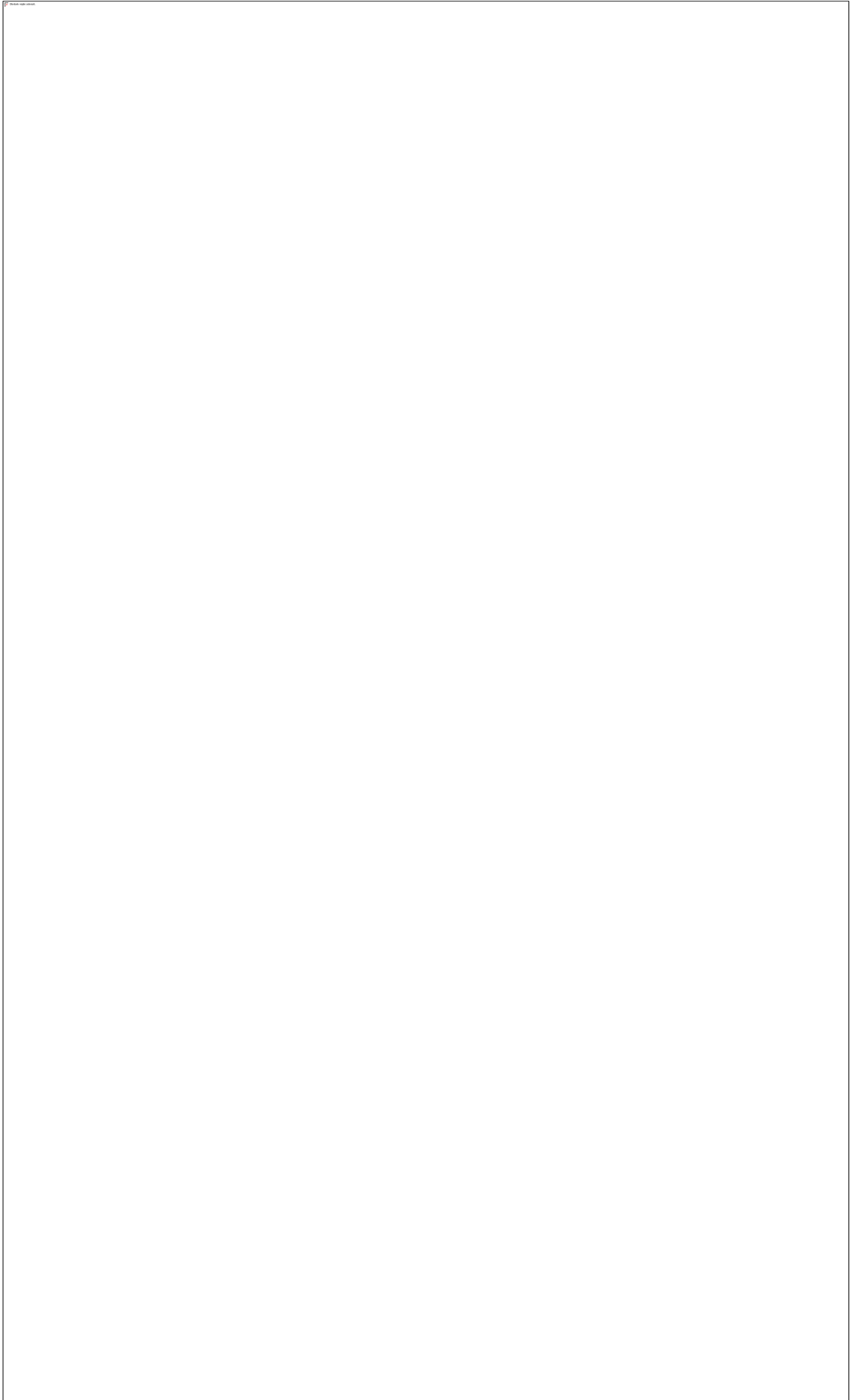
Příloha č. 11: Technický výkres čelního panelu



Příloha č. 12: Rozpiska součástek

Bill Of Materials for JN-645 Generátor V1-00-51.sch on Sun Mar 19 17:58:14 2023			
Item	Qty	Reference	VALUE
1	1	U24	ADM483EARZ
2	1	U7	LM78L05ABD
3	1	U5	LM78L15ABD
4	1	U37	NCV4274ADT33
5	1	U4	LM79L15ABD
6	1	U32	ADM706TARZ
7	1	U34	NCP585-12
8	1	U11	AT45DB041D
9	57	C3-4	100n
		C6-13	
		C15-17	
		C20	
		C23-24	
		C26-27	
		C29 C38	
		C48	
		C50-52	
		C55-57	
		C67-75	
		C78	
		C80-81	
		C83-84	
		C98	
		C101-103	
		C105-111	
		C119 C126	
		C144-145	
		C153	
10	6	C28 C44	100p
		C54 C77	
		C89-90	
11	10	C1 C5 C14	10M/50V
		C25	
		C33-34	
		C42 C47	
		C49 C88	
12	1	C92	160p
13	8	C18-19	1M
		C21-22	
		C30-32	
		C36	
14	7	C82 C104	1n
		C120 C125	
		C136-137	
		C154	
15	1	C91	200p
16	2	C43 C53	220p
17	1	C93	22p
18	6	C39-41	2M2/100V
		C45	
		C85-86	





Příloha č. 12: Manuál použití

1. Zasuňte generátor do racku, zatlačte, dokud neuslyšíte cvaknutí, přišroubujte.
2. Na PC otevřete aplikaci terminál, tam si zvolte parametry výstupního signálu a klikněte odeslat.

Seznam obrázků

Obrázek 12: Časy SPI, vlastní tvorba

Obrázek 13: Program pro SPI, vlastní tvorba

Obrázek 14: Stavový automat pro konfiguraci D/A převodníku, vlastní tvorba

Obrázek 15: příjem bajtů z RS-485

Obrázek 16: doporučené zapojení zemí [1]

Obrázek 17: pájení, vlastní tvorba

Obrázek 18: Měření spektra, vlastní tvorba

Obrázek 19: měření velikosti spektrální čáry vzorkovací frekvence

Obrázek 20: pracoviště pro měření spektrálním analyzátozem, vlastní tvorba

Obrázek 21: Měření zatěžovací charakteristiky, vlastní výroba

Obrázek 22: Ověření přesnosti frekvencí

Seznam tabulek

Tabulka 5: porovnání naměřených a zadaných hodnot, vlastní tvorba

Tabulka 6: Výsledky harmonické analýzy, vlastní tvorba

Tabulka 7: Měření zatěžovací charakteristiky, vlastní tvorba

Tabulka 8: Ověření výstupních frekvencí, vlastní tvorba