

STŘEDOŠKOLSKÁ ODBORNÁ ČINNOST

Obor č. 10: elektrotechnika, elektronika a telekomunikace

Měření času s vysokým rozlišením

**Marek Karlíček
Olomoucký kraj**

Šumperk 2022

STŘEDOŠKOLSKÁ ODBORNÁ ČINNOST

Obor č. 10: elektrotechnika, elektronika a telekomunikace

Měření času s vysokým rozlišením

High resolution time measurement

Autoři: Marek Karlíček

**Škola: Vyšší odborná škola a Střední průmyslová škola, Šumperk,
Gen. Krátkého 1, 787 29**

Kraj: Olomoucký kraj

Konzultant: Mgr. Michal Dudka

Šumperk 2022

Prohlášení

Prohlašuji, že jsem svou práci SOČ vypracoval/a samostatně a použil/a jsem pouze prameny a literaturu uvedené v seznamu bibliografických záznamů.

Prohlašuji, že tištěná verze a elektronická verze soutěžní práce SOČ jsou shodné.

Nemám závažný důvod proti zpřístupnění této práce v souladu se zákonem č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších předpisů.

V Novém Malíně dne 2.5. 2022

.....

Marek Karlíček

Poděkování

V první řadě bych rád poděkoval Mgr. Michalu Dudkovi za námět a výborné odborné vedení práce a také Přírodovědecké fakultě UP v Olomouci za zázemí, které jsem k práci mohl využít.

Děkuji Robertu Stárkovi za data a graf pro srovnání dosažených výsledků s komerčně dostupným přístrojem.

ANOTACE

Tato práce se zabývá návrhem a realizací zařízení pro měření krátkých časů s vysokým rozlišením pomocí Time-to-Digital konvertoru (TDC) a mikrokontroleru STM32. Detailně charakterizuje přesnost měření a popisuje konstrukci zařízení se dvěma vstupy. Výsledkem práce je realizované zařízení pro měření času s rozlišením až 6 ps mezi dvěma signály. Dosažené parametry byly ověřeny sérií měření a porovnány s jiným, komerčně vyráběným přístrojem.

ANNOTATION

This thesis deals with the design and assembly of device for measuring of short times with high resolution using a time-to-digital converter (TDC) and STM32 microcontroller. It characterizes in detail the measurement accuracy and describes the construction of the device with two inputs. As a result, is instrument for measuring of time with a resolution of up to 6 ps between two signals. Achieved parameters were verified by a series of measurements and compared with another commercially manufactured device.

KLÍČOVÁ SLOVA/KEYWORDS

TDC; MCU; TOF

Obsah

Úvod	8
1 Motivace	9
2 Koncepce zařízení a výběr komponent.....	10
2.1 Mikrokontroler (MCU)	11
2.2 TDC	11
2.3 Komparátory	11
3 Popis integrovaného obvodu AS6500/AS6501	13
3.1 Časové značky	14
3.2 Registry a konfigurace AS650x	14
3.2.1 Volba dělení periody REFCLK a bitové hloubky	15
3.2.2 Vysoké rozlišení	16
3.2.3 Kombinace kanálů	16
3.3 Komunikační rozhraní SPI.....	17
3.4 Přenos dat.....	17
3.4.1 Čtení dat po SPI.....	18
3.4.2 Oddělené sériové kanály AS6501	18
4 Návrh zapojení.....	20
4.1 Vstupní komparátory	20
4.1.1 Napájení komparátorů	21
4.1.2 Vstupní rozsah a komparační úroveň	21
4.2 Napájení	22
4.3 USB převodník	22
4.4 TDC a převodníky logických úrovní	23
4.5 MCU	24
4.5.1 Programovací rozhraní	25
4.5.2 Zdroj taktovacího signálu	25
4.5.3 Komunikační rozhraní	25
5 Deska plošných spojů a osazení	26
5.1 Rozložení a návrh	26
5.2 Osazení.....	26
5.2.1 Fotodokumentace osazené PCB	27
6 Mechanická konstrukce zařízení	28
7 Program MCU	29

7.1	Konfigurace periférií.....	29
7.1.1	RCC	30
7.1.2	SAI.....	30
7.1.3	DMA.....	31
7.1.4	DAC.....	31
7.1.5	USART	31
7.1.6	SPI	32
8	Zpracování změřených dat.....	33
8.1	Vyobrazení dat a stanovení rozptylu hodnot	33
8.1.1	Skripty pro práci s daty.....	34
9	Charakterizace	35
9.1	Rozptyl měřených hodnot.....	35
9.2	Rozdílný rozptyl měřených hodnot.....	37
9.2.1	Úprava REFCLK	37
9.2.2	Další nedostatky	39
10	Ovládání přístroje	41
10.1	Příkazy pro ovládání	41
11	Možnosti dalšího vývoje.....	42
12	Problémy s realizací.....	43
12.1	Testovací HW	43
12.2	Nekvalitní pájecí maska.....	44
13	Závěr.....	45
13.1	Srovnání výrobku s komerční alternativou	45
13.2	Aplikace zařízení	47
13.2.1	APD detektory	47
13.2.2	Metastable region D klopného obvodu.....	48
14	Seznam obrázků.....	49
15	Seznam tabulek.....	50
16	Seznam zkratk.....	51
17	Použitá literatura.....	52
18	Přílohy	53

ÚVOD

Měření času s vysokým rozlišením a přesností je důležitou součástí řady výzkumných aplikací a laboratorních měření. Cílem mojí práce je detailní popis a charakterizace time-to-digital converter (TDC) čipu a jeho využití v ukázkové aplikaci.

V první části práce jsem vybral TDC – AS650x a optimalizoval propojení s mikrokontrolerem (MCU) STM32H7xx. Otestoval jsem funkčnost na vývojových kitech a následně navrhnul a zrealizoval vlastní zařízení s uvedenými integrovanými obvody, vstupními komparátory a komunikačním rozhraním USB. Zařízení disponuje dvojicí vstupů s impedancí 50 Ω .

Následně jsem vytvořil algoritmus pro zpracování přijatých dat. Měřené časové intervaly mezi kanály jsou zaznamenávány do histogramu; jeden vstup slouží jako začátek měřeného intervalu a druhý jako jeho konec. Čas pro zpracování dat jsem minimalizoval.

Zařízení slouží například k měření vlastností lavinových fotodiodových detektorů (APD) pro výzkumné účely optických laboratoří. Tyto detektory mají určité měřitelné zpoždění a falešné pulzy, které mohou nastat jednak na startovacím signálu, jednak na signálu stop, jenž je mé zařízení schopno ignorovat.

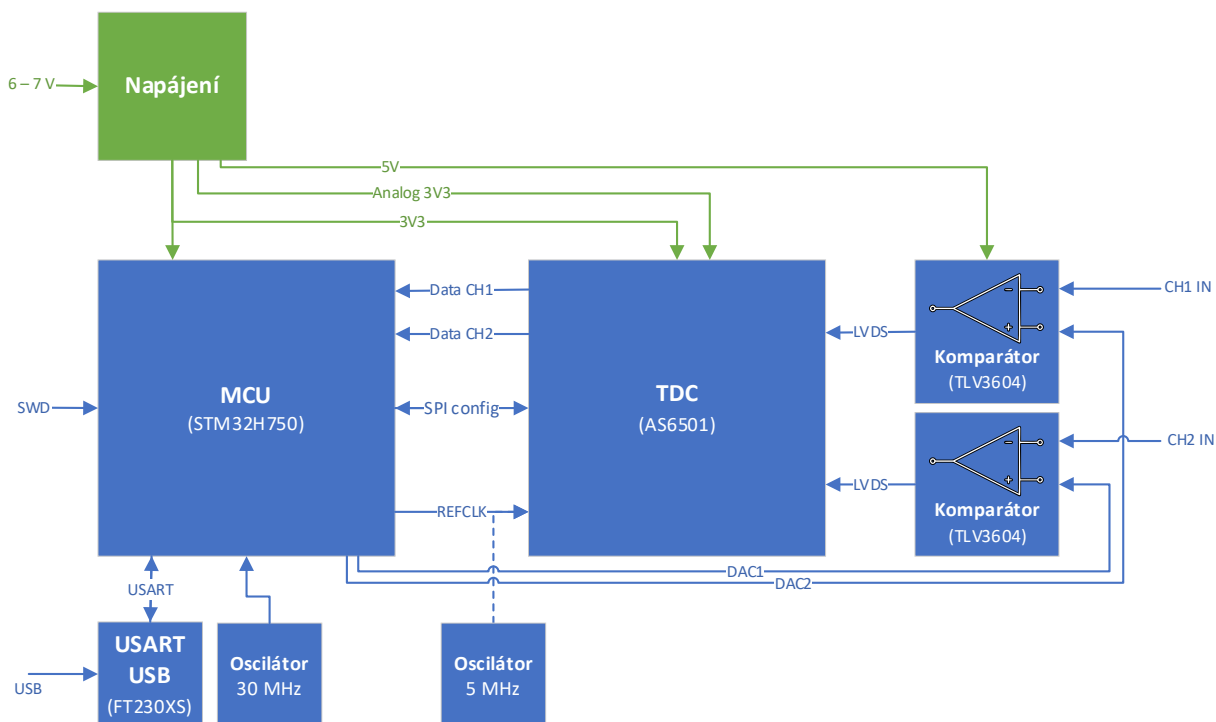
U zařízení jsem detailně charakterizoval rozptyl naměřených dat na široké škále zkoumaných intervalů a následně všechny dosažené výsledky jsem zpracoval do grafů a tabulek.

1 MOTIVACE

Měření času s vysokým rozlišením je důležitou součástí široké řady fyzikálních experimentů. Příkladem může být měření time-of-flight (TOF). To je čas, za který objekt, částice nebo vlna (elektromagnetická, zvuková, ...) urazí určitou dráhu v daném prostředí. Díky tomu můžeme určovat rychlost nebo dráhu letu. Toho je využíváno nejen v digitální fotografii, ale také například v zobrazovací technice jako je magnetické rezonance (MRI) nebo při autonomním řízení vozidel (LIDAR). Dalšími využitími je široká řada například v částicové fyzice, spektroskopii, jaderné fyzice, ale také v medicíně.

Téma mi přišlo zajímavé nejen z pohledu elektroniky, ale také z pohledu možného uplatnění jak v praxi, tak i ve výzkumu. TDC čipy nebyly zatím součástí žádného proprietárního zařízení na katedře, kde jsem přístroj vyvíjel. Mým cílem bylo ukázat potenciál využití těchto čipů a zjednodušit jejich implementaci do nových zařízení.

2 KONCEPCE ZAŘÍZENÍ A VÝBĚR KOMPONENT



Obrázek 1 - Blokové schéma vyvinutého zařízení

Zařízení se skládá ze dvou primárních částí (MCU, TDC) a napájení.

První část se skládá z MCU, který je spojen s čipem pro USB komunikaci virtuálním sériovým portem (VCP), díky tomu je možné dosáhnout snadné obsluhy zařízení a jednoduché přenositelnosti zařízení mezi aplikacemi. K MCU je dále připojen oscilátor.

Druhou část tvoří TDC spolu s komparátory s volitelnou komparační hladinou, kterou lze nastavovat pomocí digitálně analogového převodníku v MCU. Během charakterizace jsem musel přidat druhý oscilátor vzhledem k nedostačujícím parametrům taktu generovaného pomocí časovače v STM32. Přidáním druhého oscilátoru (5 MHz) se stabilita původního oscilátoru (30 MHz) neprojevuje do výsledků měření.

Napájení je rozděleno na 3 části. První větev napájí 3,3V digitální obvody (MCU, převodníky, digitální část TDC). Druhá větev napájí analogovou část TDC; výrobce zde doporučuje oddělené napájení. Třetí větev napájí vstupní komparátory 5 V, aby byl rozšířen rozsah vstupního napětí.

2.1 Mikrokontroler (MCU)

Pro řízení celého zařízení a zpracování dat bylo nutné vybrat dostatečně výkonný mikrokontroler s potřebnými periferiemi a s dostatečnou pamětí RAM. Zařízení využívá výkonný 32bitový MCU STM32H750 s jádrem Cortex-M7, který pracuje s taktovací frekvencí 480 MHz. Pro zařízení je důležitá periferie SAI, která se stará o příjem dat, DMA se stará o jejich transport do paměti. Výpočetní výkon jádra je důležitý pro rychlé zpracování dat.

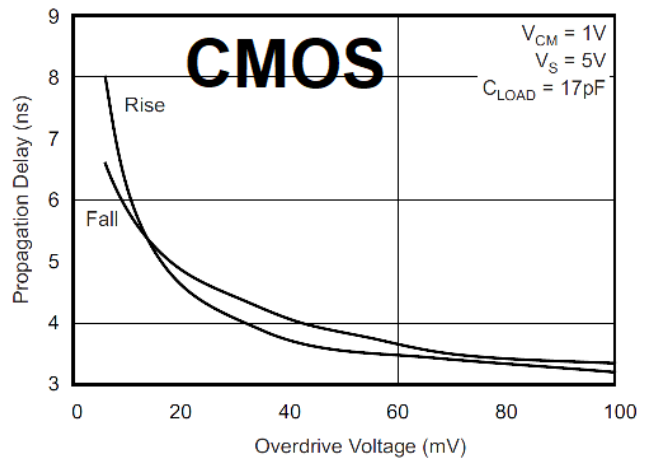
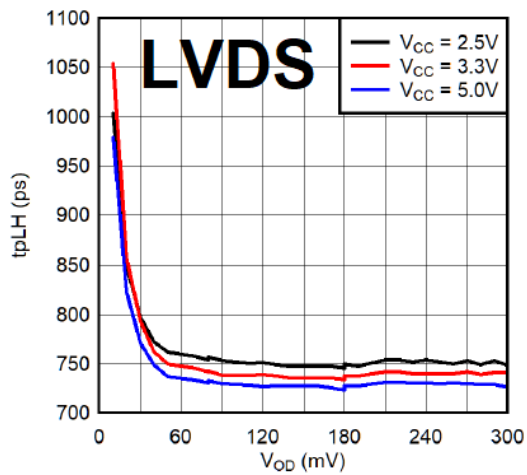
2.2 Časový převodník (TDC)

Time to digital convertor (TDC) – AS6501 jsem vybral s ohledem na požadavek vysokého rozlišení měření, nízkou cenu (přibližně 700 Kč) a dobrou dostupnost i v době součástkové krize. Jeho komunikační rozhraní (s přidáním převodníků) je vyhovující pro práci s mikrokontrolerem. V neposlední řadě pro mě bylo významným faktorem pro výběr pouzdro (LQFP-48), abych se vyhnul problémům spojeným s ručním osazováním komponent.

2.3 Komparátory

TDC čip umožňuje CMOS vstup, čímž by se výrazně zjednodušil návrh PCB. Stálost propagation delay je stěžejní pro výsledné měření. Vzhledem k nárokům na tento parametr jsem vybral komparátor s diferenciálním výstupem LVDS, který je zakončen 100Ω impedancí. Vzhledem k délce spojů mezi komparátorem a TDC čipem nejsou vyšší nároky na návrh PCB (impedanční přizpůsobení spojů). Zkreslení způsobené nevhodným rozvržením je téměř zanedbatelné.

Pro srovnání CMOS a LVDS komparátorů (v podobné cenové hladině) je nejzajímavějším ukazatelem rozdílnost zpoždění (propagation delay) ve vztahu k napětí, o které byla překročena jejich komparační hladina (LVDS – TLV3605 vs CMOS – TLV3502).

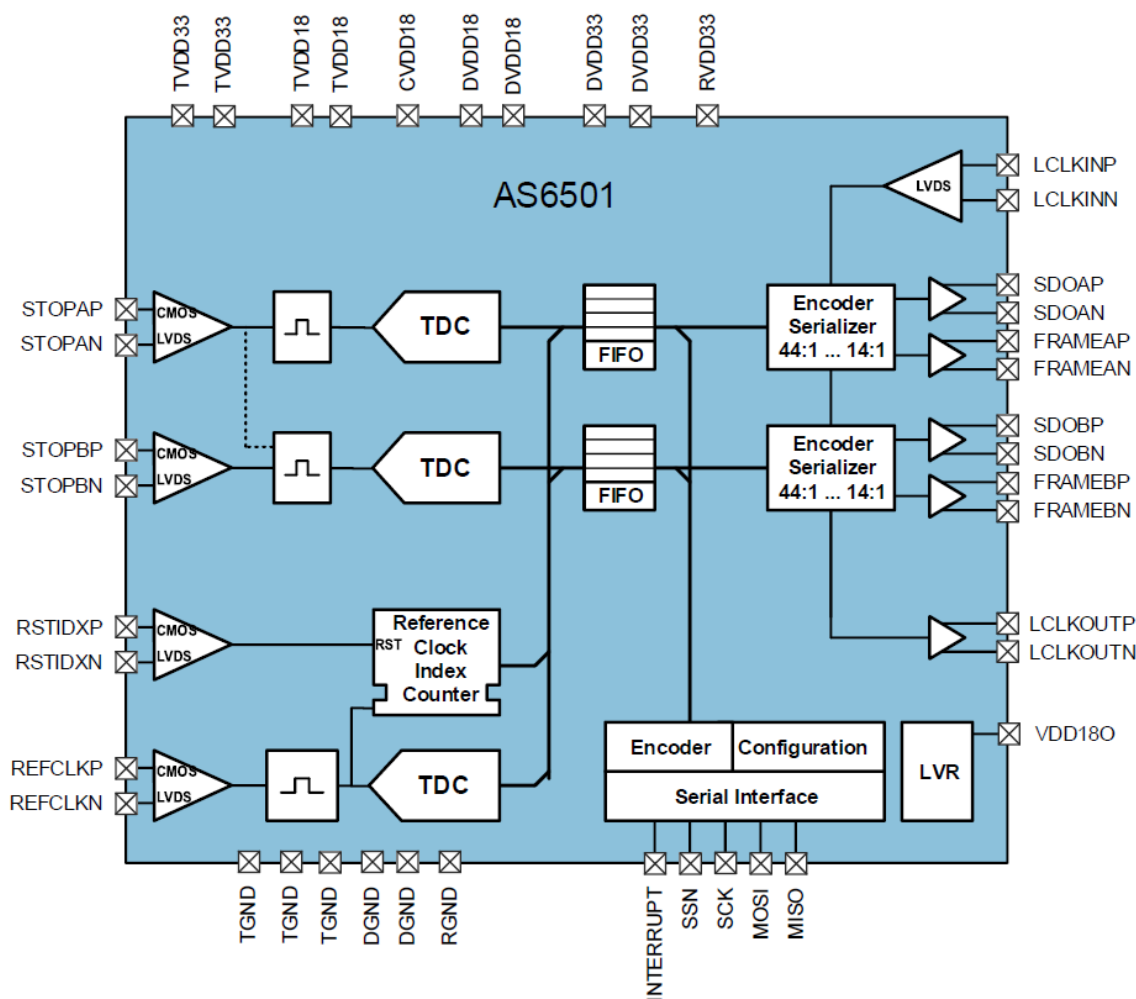


Obrázek 2 - Srovnání komparátorů [12][13]

Ze srovnání (Obrázek 2) jednoznačně vyplývá, že komparátor s LVDS má výrazně nižší a stálejší zpoždění, díky tomu je vhodnější pro mou aplikaci. Vzhledem k ceně, dostupnosti a výborným vlastnostem jsem použil do mé aplikace TLV3604.

3 POPIS INTEGROVANÉHO OBVODU AS6500/AS6501

Funkcí Time to Digital convertoru (TDC) je zaznamenání časové značky s co největší přesností v okamžiku příchodu signálu na jeden z jeho vstupů. Použitý TDC čip se vyrábí v několika verzích. Pro účely své práce jsem dvě z nich vyzkoušel – AS6501 a AS6500.



Obrázek 3 - Blokové schéma TDC [7]

Každá disponuje několika měřicími kanály (AS6501 – dvojicí; AS6500 – čtveřicí) a jedním čítačem společným pro všechny kanály. Jednotlivé interní TDC bloky slouží k zachycení přesného času v rámci jedné periody referenčního čítače. Referenční (společný) čítač je typicky časovaný z externího generátoru taktovacího signálu nebo z krystalu (pouze u AS6500). Jitter tohoto taktu by neměl příliš ovlivňovat přesnost měření v rámci jedné jeho periody, jeho vliv se projevuje v případě měření časů větších než je půl periody referenčního taktu. Konfigurace probíhá po sériové sběrnici SPI. Po sběrnici SPI je rovněž možné získávat změřené časové značky. Výčet změřených hodnot

sběrnicí SPI je pomalejší než rychlost měření. Proto existuje verze čipu AS6501 s odděleným sériovým výstupem pro každý měřicí kanál. Tyto výstupy jsou určené pro komunikaci s FPGA, proto využívají diferenciální logické výstupy (LVDS). Pro tento čip jsem se rozhodl ve finálním zařízení. Přestože AS6501 verze čipu obsahuje pouze dva měřicí kanály, nabízí možnost připojení vstupů LVDS logikou, čímž se zlepší přesnost záznamu časové značky díky rychlejšímu přechodu mezi logickými úrovněmi.

Tabulka 1 - Srovnání AS6501 a AS6500

Parametr	AS6500	AS6501
Počet kanálů	4	2
SPI	ANO	ANO
LVDS vstupy	NE	ANO (volitelné)
Oddělené výstupy dat	NE	ANO
Maximální teoretická rychlost pro 1 kanál	< ~ 890 KSa/s (48 bit)	< ~5 MSa/s (32 bit)
Pouzdro	QFN-40 (dostupný dev. kit)	QFP48

3.1 Časové značky

Zaznamenaná časová značka se skládá ze dvou částí; první část (REFID_x) je okamžitý stav referenčního čítače v okamžiku příchodu nástupné hrany na příslušný kanál, druhá (TSTOP_x) vyjadřuje čitatele zlomku periody REFCLK. Jmenovatele lze konfigurovat.

Přepočet zaznamenaných dat na absolutní čas se provádí podle následujícího vztahu (1).

$$t = (REFID_x * t_{REFCLK-PERIOD}) + \left(\frac{TSTOP_x}{REFCLK_DIVISIONS} * t_{REFCLK-PERIOD} \right) \quad (1)$$

3.2 Registry a konfigurace AS650x

Konfigurace u obou čipů probíhá po sériovém rozhraní SPI zápisem do registrů čipu.

Pro účely mého zařízení (měření start-stop histogramu) jsem TDC konfiguroval následně. Periodu referenčního čítače jsem rozdělil na 2^{16} dílů, aktivoval oba kanály i vstup REFCLK. Vybral jsem vhodný formát výstupních dat (16 a 16 bitů) a výstupy aktivoval. Pro dosažení nejvyšší možné přesnosti jsem využil LVDS vstupní logické úrovně a nejvyšší možné rozlišení.

```
#define REFCLK_DIVISIONS ((uint32_t)65536) // 2^16 - setup according to your needs
#define CFG0 0b00110101 // Enable LVDS_OUT, STOPA, REFCLK
#define CFG1 0b10000101 // High REsolution, STOPA
#define CFG2 0b00001100 // 16bit STOP, 16bit REFINDEX
#define CFG16 0b00000000 // LVDS Inputs

const uint8_t config_register[] = {
    CFG0,
    CFG1,
    CFG2,
    REFCLK_DIVISIONS & 0xFF, // CFG3
    (REFCLK_DIVISIONS>>8) & 0xFF, // CFG4
    REFCLK_DIVISIONS>>16, // CFG5
    0b00000000, // CFG6 - no LVDS test pattern
    0b00000000, // CFG7 - no LVDS adjust
    0xA1, // CFG8 - fixed values
    0x13, // CFG9 - fixed values
    0x00, // CFG10 - fixed values
    0x0A, // CFG11 - fixed values
    0xCC, // CFG12 - fixed values
    0xCC, // CFG13 - fixed values
    0xF1, // CFG14 - fixed values
    0x7D, // CFG15 - fixed values
    CFG16
};
```

3.2.1 Volba dělení periody REFCLK a bitové hloubky

Jako kompromis mezi vysokou rychlostí přenosu, rozlišením a možností měřit delší periody jsem zvolil výstup ve formě 32bit čísla, horních 16 bitů je hodnota REF_INDEX, dolních 16 bitů je hodnota STOP_DATA. Strop tohoto čítače jsem volil tak, aby tato dvě čísla na sebe navazovala (výsledek lze interpretovat jako jedno 32bit číslo), a díky tomu jsem ušetřil čas potřebný ke zpracování výsledků. REFCLK_DIVISIONS jsem tedy volil 2^{16} tj. 65536.

Výsledné rozlišení je možné určit vztahem (2).

$$time_resolution = \frac{f_{REFCLK}^{-1}}{2^{STOP_DATA_BITWIDTH}}$$

(2)

3.2.2 Vysoké rozlišení

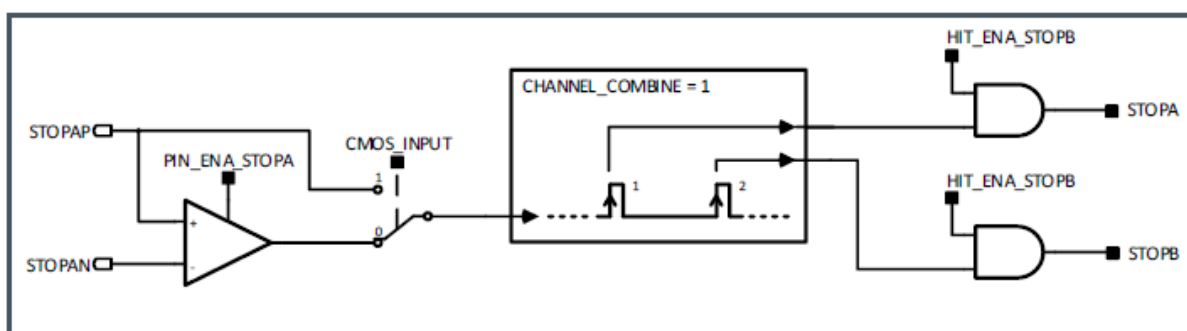
Aktivací HIGH_RESOLUTION, které několikrát (2x nebo 4x) průměruje změřený výsledek, se zvyšuje přesnost měření na úkor horšího minimálního odstupů pulzů (t_{PPS}) na jednom kanálu. Tabulka ukazuje srovnání jednotlivých režimů vysokého rozlišení. Ten lze snížit interní kombinací dvou kanálů (Obrázek 4).

Tabulka 2 - Režimy vysokého rozlišení TDC

Nastavení	Oversampling	Min. t_{PPS}
High_Resolution = 0	Off	20 ns
High_Resolution = 1	2x	50 ns
High_Resolution = 2	4x	100 ns

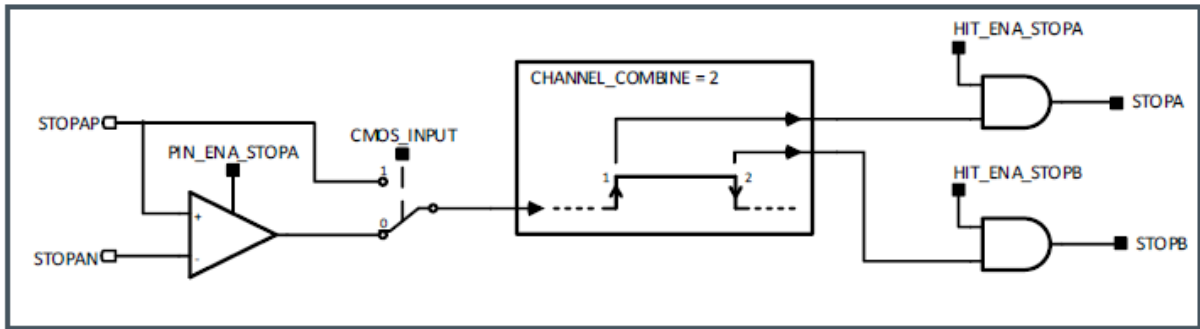
3.2.3 Kombinace kanálů

Čip nabízí možnost interní kombinace kanálů dvěma způsoby. První je kombinace pro snížení minimálního času mezi pulzy (t_{PPS}) na jednom kanálu. Toho je dosaženo spojením dvou kanálů k sobě (Obrázek 4), automatickým střídáním kanálů. Mimo snížení t_{PPS} (pro jednu dvojici pulzů je povolený odstup minimálně 5 ns) získáváme dvojnásobnou hloubku FIFO a dvojnásobný datový průtok z čipu.



Obrázek 4 - Blokové schéma kombinace kanálů 1 [7]

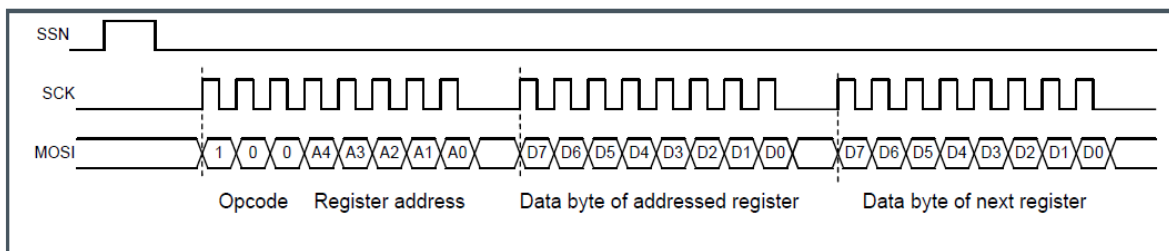
Druhá kombinace slouží k snímání délky pulzu. Jeden kanál zaznamenává nástupné a druhý sestupné hrany (Obrázek 5). Zde je potřeba dbát na přesné pořadí vyčítání dat.



Obrázek 5 - Blokové schéma kombinace kanálů 2 [7]

3.3 Komunikační rozhraní SPI

Konfigurace probíhá po rozhraní SPI, dle standardu Motorola (Clock Phase Bit = 1; Clock Polarity Bit = 0). Hodnota datových linek se mění při nástupné hraně SCK.



Obrázek 6 - Časový diagram komunikace SPI [7]

Každý přenos začíná SSN pulzem, který je následován příkazem („opcode“) a adresou zápisu nebo čtení. Délka příkazu spolu s adresou je 1 byte. Zařízení umožňuje přístup s automatickou inkrementací adresy, po příkazu může následovat zápis/čtení libovolného počtu bytů. Časový diagram celé komunikace je vyobrazen výše (Obrázek 6).

3.4 Přenos dat

Pro přenos dat lze využít sériové rozhraní SPI umožňující nejen přenos naměřených výsledků, ale rovněž konfigurace TDC. V zájmu dosažení co nejvyšší rychlosti jsem vyzkoušel několik různých způsobů komunikace z pohledu použitých periférií. Jako nejlepší se ukázalo řešení za využití periférie QSPI (po SPI sběrnici TDC). Pro finální aplikaci se rozhraní ukázalo jako nevyhovující, a proto jsem využil čip s oddělenými LVDS výstupy. Přes příslušné převodníky jsou přivedeny datové signály na SAI periférii MCU, která data uloží (ve spolupráci s DMA) do paměti.

3.4.1 Čtení dat po SPI

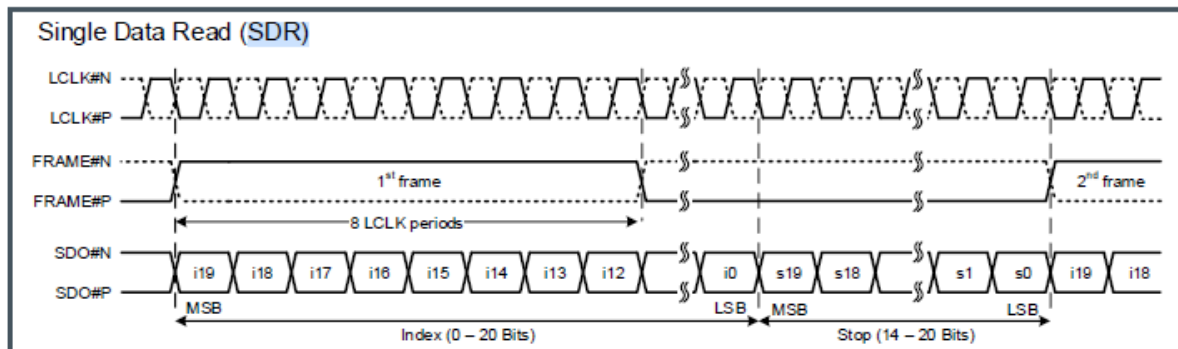
Obě verze čipu podporují vyčítání přes SPI, výhodou je absence potřeby konvertovat logické úrovně a možnost využití AS6500 s více vstupy. Pro vyčtení naměřených dat je potřeba poslat příkaz a adresu registru, ze kterého se bude vyčítat. Všechny registry výsledků je možné číst pouze jedním příkazem (to platí pouze pro AS6500, u AS6501 je mezi prvním a druhým kanálem volná šestice bytů).

Čipy disponují pinem signalizujícím data dostupná k vyčtení z FIFO na jednom nebo všech aktivních kanálech (volba režimu výstupu se dělá za pomoci konfiguračního bitu *COMMON_FIFO_READ*). Při maximalizování datového průtoku z čipu je potřeba si uvědomit dlouhou reakční dobu na přerušení uvnitř MCU. Cyklické vyčítání dat s nejvyšší možnou frekvencí (v případě čtení prázdného bufferu vyčteme hodnotu všech bytů 0xFF, kterou nebudeme dále zpracovávat) může být často výhodnějším řešením, nevýhodou je výrazně vyšší vytížení výpočetního jádra procesoru.

Periferie QSPI slouží primárně pro komunikaci s pamětí. Komunikuje jednou až čtyřmi datovými linkami (AS650x nabízí pouze jednu datovou linku). Její výhodou je, že jedním zápisem do registru zajistí celou komunikaci od poslání příkazu (včetně SSN signálu) ke čtení až po samotné vyčtení dat. QSPI v testovaném mikrokontroleru (STM32H7xx) nabízí tři režimy funkce. Jediný, který lze využít v mojí aplikaci, je „manuální mód“ (Indirect mode). Nevýhodou této periferie je značně náročnější konfigurace DMA, protože přenos lze spustit pouze zápisem do registrů. Periferie neumožňuje změnu polarity časovacího signálu CLK, který je opačný, než požaduje AS650x. Proto je nezbytné přidat hardwarový invertor.

3.4.2 Oddělené sériové kanály AS6501

AS6501 disponuje LVDS sériovými výstupy pro každý kanál a signálem „FRAME“, který upozorňuje na dostupnost dat. Čip vyžaduje stálý takt, na základě kterého generuje výstupní hodinový signál pro komunikaci. TDC tedy vysílá výstupní data na základě vlastního taktu, který není ve fázi s přivedeným. Toto rozhraní je určeno především pro využití v kombinaci s hradlovým polem (FPGA), i přes to jej lze jednoduše využít u MCU za použití periferie SAI a příslušnými převodníky logických úrovní. Toto řešení je na STM32 schopné celý příjem dat obstarat téměř nezávisle na výpočetním jádru a tím zefektivnit příjem i zpracování. Signál „FRAME“ nabývá kladné hodnoty spolu s prvním bitem naměřených dat, jeho trvání je 8 period komunikačního taktu (Obrázek 7).



Obrázek 7 - Časový diagram výstupních dat [7]

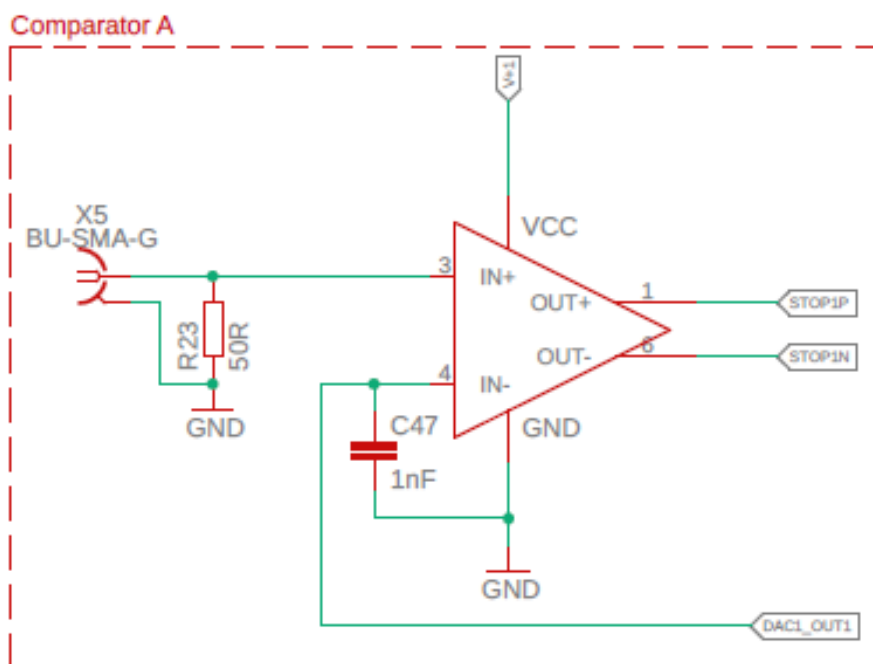
TDC nabízí možnost výstupu dat na obě hrany taktovacího signálu (DDR) a tím zdvojnásobit datový průtok. DDR není vyhovující pro zpracování mikrokontrolerem.

4 NÁVRH ZAPOJENÍ

V rámci mé práce jsem navrhl a vyrobil desku splňující výše uvedené požadavky.

Zařízení jsem testoval jako měřič času mezi nástupnými hranami signálů START a STOP na dvojici vstupů.

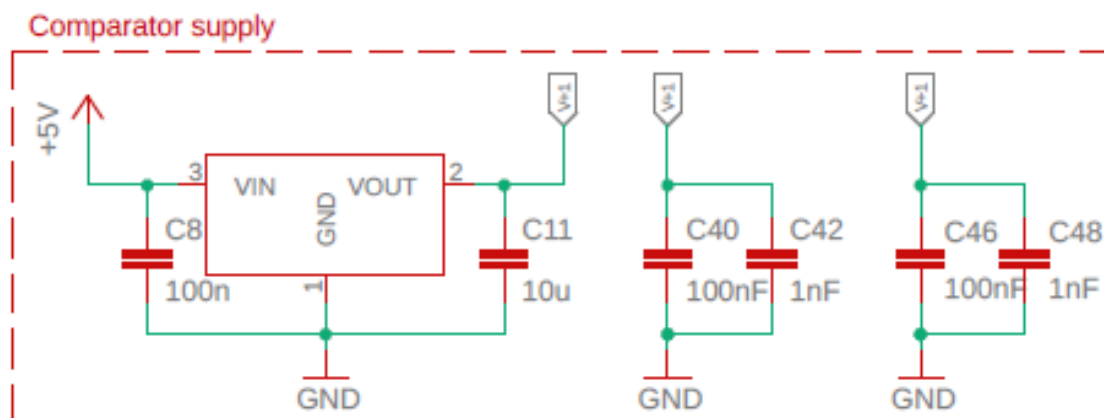
4.1 Vstupní komparátory



Obrázek 8 - Schéma komparátorů

Vstupem zařízení je analogový signál přivedený koaxiálním kabelem na běžně používaný konektor typu SMA. Signál je zakončený 50Ω odporem, což je standardem pro podobná zařízení, aby se zamezilo odrazům ve vedení. Na vstupu se nachází komparátor s LVDS výstupem. Komparační hladina je nastavitelná pomocí integrovaného DAC v mikrokontroleru.

4.1.1 Napájení komparátorů



Obrázek 9 - Schéma napájení komparátorů

Dvojice komparátorů má svůj vlastní lineární stabilizátor napětí (TS9011), pro nízké zvlnění napájecího napětí. Zvolil jsem napájecí napětí 5 V z důvodu co nejvyššího povoleného napětí vstupního signálu.

Napájení komparátoru je odfiltrováno dvojicí kondenzátorů (100 nF a 1 nF). Vzhledem k parazitní indukčnosti 100nF kondenzátoru by mohlo dojít k rezonanci na vysoké frekvenci, blížíci se k frekvenci, na které pracuje komparátor. K odstranění případné rezonance proto přidávám druhý kondenzátor o nižší kapacitě, ale hlavně nižší indukčnosti.

4.1.2 Vstupní rozsah a komparační úroveň

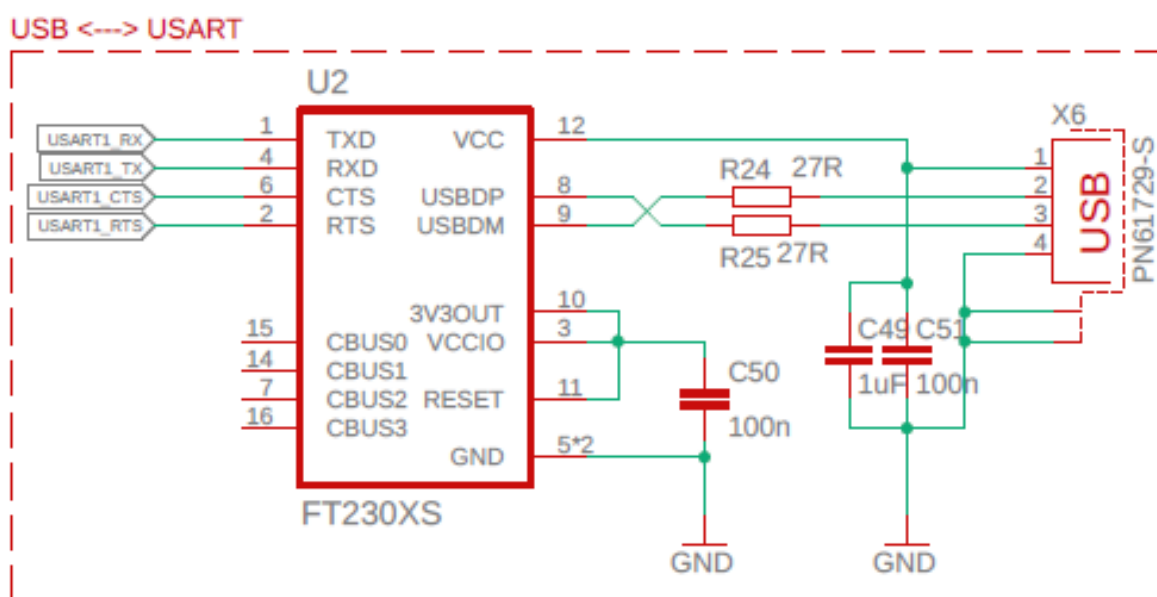
Vstupy použitého komparátoru jsou schopny pracovat v plném rozsahu napájení. Rozdíl mezi invertujícím a neinvertujícím vstupem není dále omezen.

Pro nastavení komparační úrovně jsem využil integrovaný DAC, který nabízí jednak dostatečné rozlišení (12 bitů) a jednak potřebnou stabilitu. Výstupní napětí DAC dosahuje maximálně pouze 2,5 V (při použití interní reference napětí), což se může na první pohled jevit jako omezující (je-li limit vstupního napětí 5 V), ovšem v praktické aplikaci tento parametr není nijak stěžejní. Omezující situace může nastat, pokud bychom snímali signál se základní stejnosměrnou složkou okolo 2,5 V (nebo vyšší) a na něm by byl sledovaný signál nebo v případě snímání pouze pulzů o amplitudě vyšší než 2,5 V. Napětí komparační úrovně je co nejbližší komparátoru odfiltrováno 1nF kapacitou pro zmírnění případných přeslechů z výstupu komparátoru, které by mohly být způsobeny parazitní kapacitou mezi vstupem a výstupem.

4.2 Napájení

Napájení je rozděleno do 3 větví. Pro napájení komparátorů jsem použil 5 V stabilizátor, viz výše. Napájení zbytku obvodu jsem rozdělil do dvou částí na základě doporučení dokumentace k AS650x, kde výrobce doporučuje oddělený stabilizátor pro analogovou část integrovaného obvodu, od digitální části obvodu spolu s MCU.

4.3 USB převodník



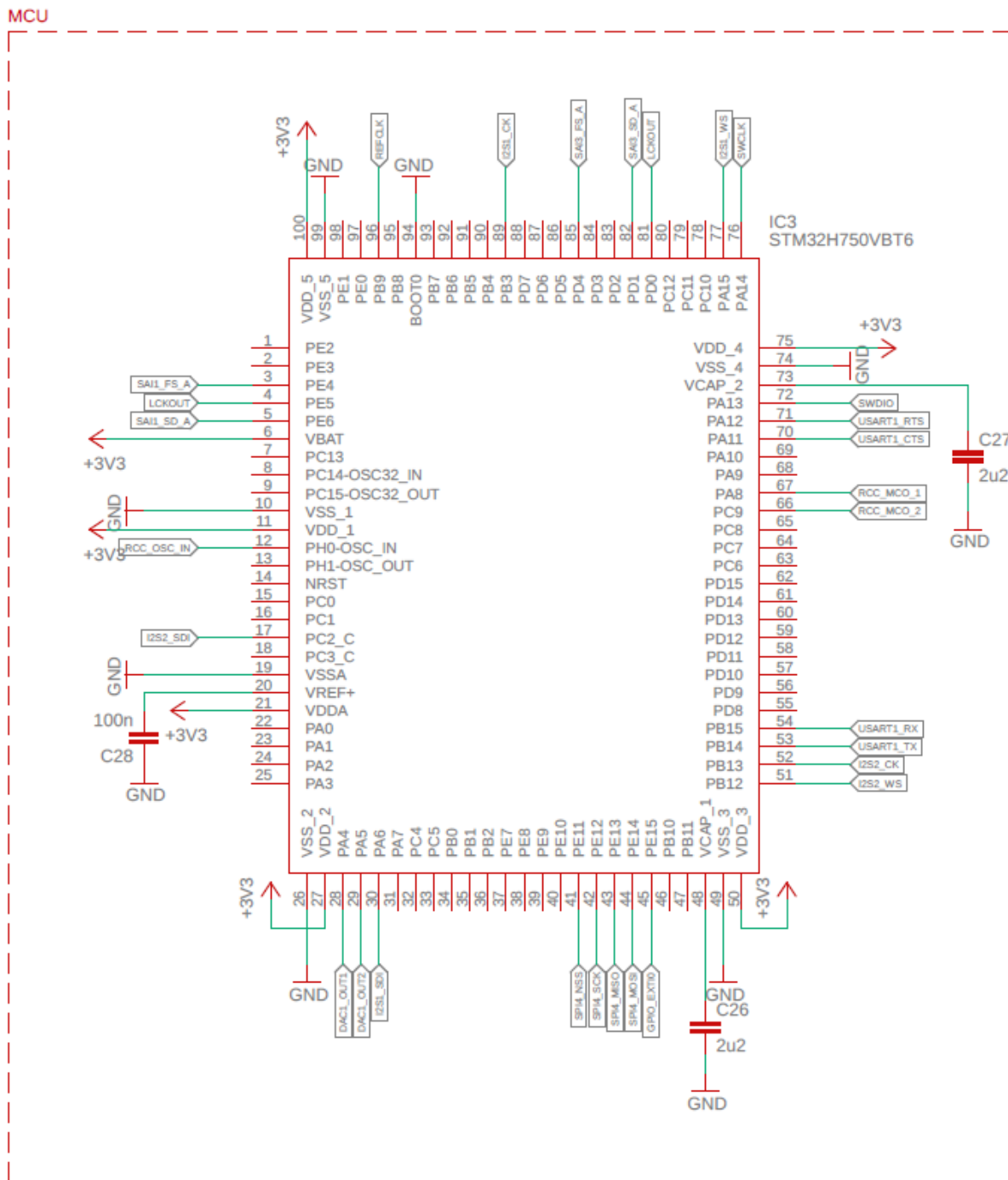
Obrázek 10 - Schéma FTDI

Pro komunikaci mezi počítačem a výrobkem jsem využil běžně užívaný USB – USART převodník FT230XS, který disponuje integrovaným zdrojem hodin a stabilizátorem pro vlastní potřebu. K čipu jsou dostupné originální knihovny pro řadu programovacích jazyků na straně počítače. Může bez problému pracovat s univerzálními knihovnami pro sériovou linku COM. Převodník je napájen z USB.

Kromě základního páru RX a TX jsem využil RTS a CTS signály pro možnost dosažení maximální přenosové rychlosti. RTS (z PC) a CTS (do PC) piny signalizují, zda je dané zařízení připravené přijímat další data.

Jako USB konektor jsem volil USB-B, kvůli jeho vysoké mechanické odolnosti a častému využití v laboratorních/průmyslových podmínkách. Mimo jiné, vzhledem ke svým rozměrům

4.5 MCU



Obrázek 12 - Schéma mikrokontroleru

Využil jsem mikrokontroler STM32H750VB, kromě maximálního možného výkonu nabízí širokou škálu periferií. Pouzdro LQFP-100 je snadno ručně pájitelné a nabízí dostatek vstupních/výstupních pinů.

4.5.1 Programovací rozhraní

Samozřejmostí je vyvedení programovacího rozhraní SWD (Serial Wire Debug) pro ST-LINK. Použil jsem běžný 4pinový konektor s 2,54mm roztečí a zámkem zabraňujícím otočení konektoru. Programovací (debugovací) interface SWD je dvojpínová alternativa JTAG používaná na ARM mikrokontrolerech, se stejným protokolem na horní vrstvě.

4.5.2 Zdroj taktovacího signálu

K MCU byl připojen oscilátor o frekvenci 30 MHz. Frekvence je dále upravována násobením a dělením za pomoci integrované PLL pro chod mikrokontroleru a jeho periférií.

Jeden z výstupů PLL přes pin `RCC_MCO_2` je využit jako zdroj taktovacího signálu komunikace pro TDC.

Z timeru TIM17 je vyveden jeden z výstupů, který lze využít jako konfigurovatelný zdroj REFCLK. To se později ukázalo jako nevhodné řešení.

4.5.3 Komunikační rozhraní

MCU je spojen s TDC piny pro periférii SAI (dva signály pro každý měřicí kanál a společný takt) pro příjem změřených dat. Pro konfiguraci, vzhledem k malému objemu dat a minimálním nárokům na rychlost komunikace, bez kompromisů využívám emulované SPI pomocí „bit-bang“, tj. vysílání pomocí přímého ovládání výstupních pinů přes registry.

Pro komunikaci s PC je připojen USART s využitím flow control.

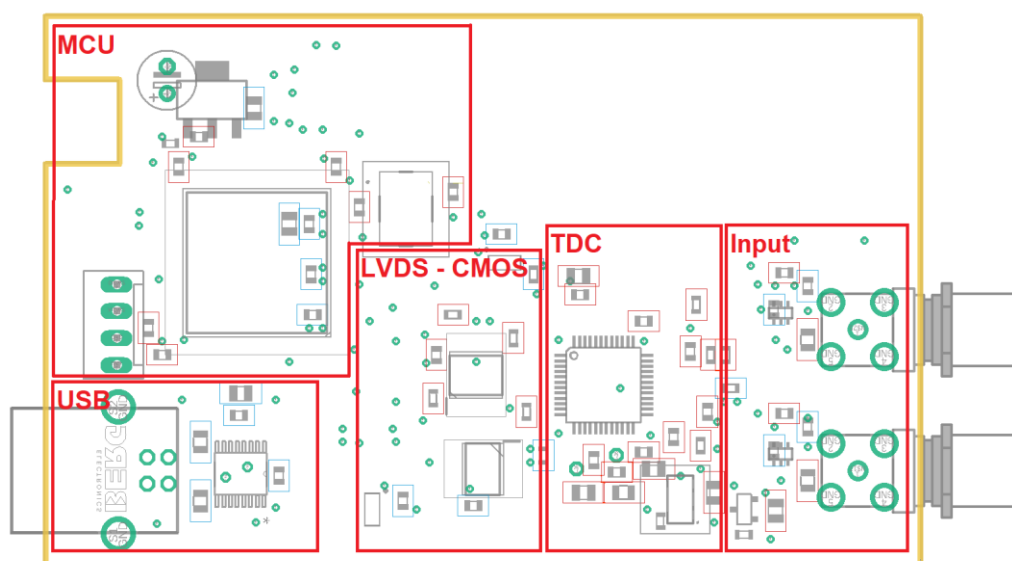
I2S komunikační rozhraní je vyvedeno na „testpointy“ pro případné využití namísto SAI.

5 DESKA PLOŠNÝCH SPOJŮ A OSAZENÍ

5.1 Rozložení a návrh

Desku plošných spojů jsem navrhoval v programu Eagle od společnosti Autodesk. Ideální by byl, vzhledem k diferenciálním signálům, čtyřvrstvý návrh. Vzhledem k výrazně vyšší ceně výroby, vyšší náročnosti návrhu a malým rozměrům jsem toto řešení vyhodnotil jako zbytné.

Rozložení jsem navrhl dle funkčních bloků podobně jako schéma a důrazně jsem dbal na propojení země (GND) pod diferenciálními signály (zvláště pod vstupními).



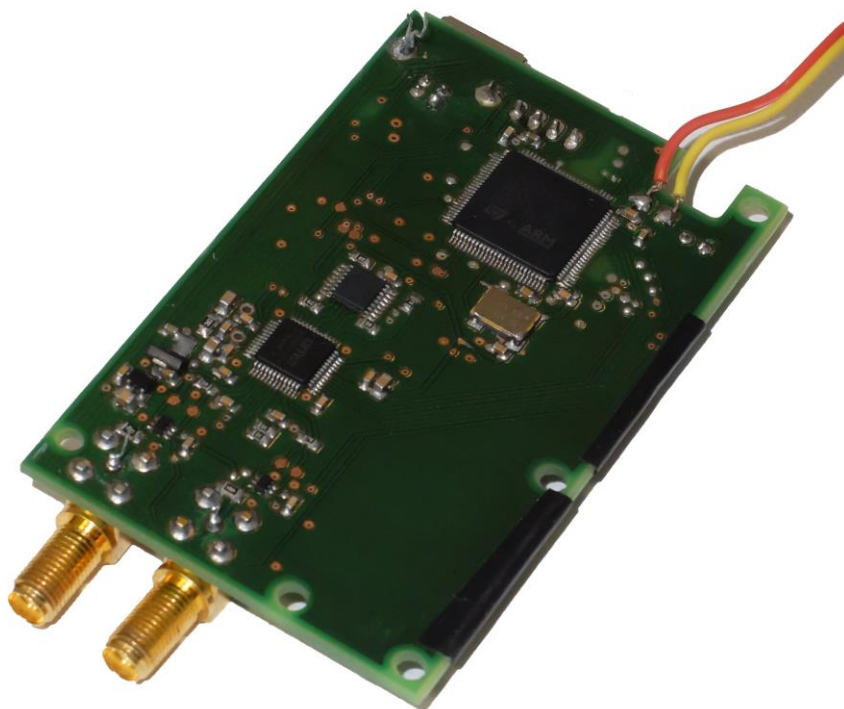
Obrázek 13 - Rozvržení PCB

DPS jsem si nechal vyrobit profesionální českou firmou Printed s.r.o. Výrobu v dostupných podmínkách jsem vyhodnotil jako nereálnou.

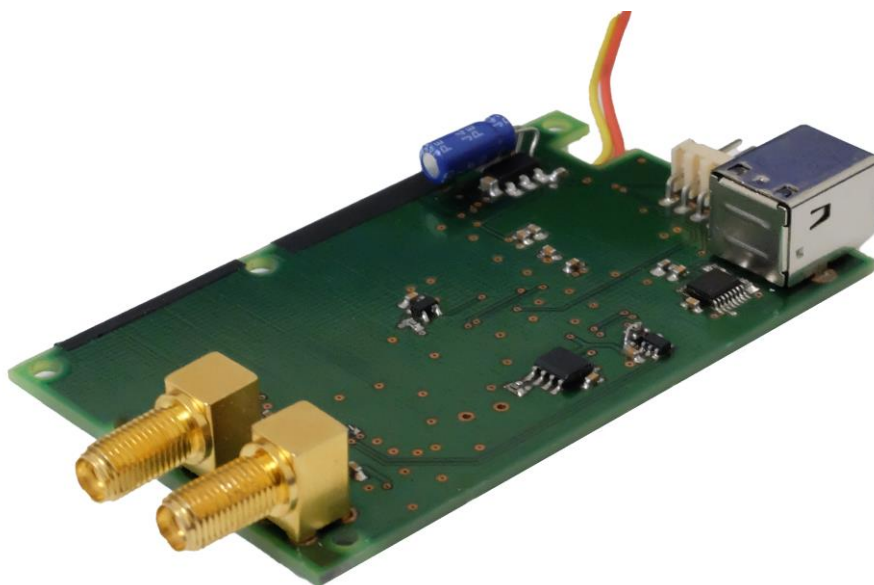
5.2 Osazení

K ručnímu osazení a zapájení jsem využil obyčejnou mikropáječku. Samozřejmostí je použití kvalitního tavidla, které výrazně zjednoduší proces. Vzhledem k výborným předchozím zkušenostem jsem využil tavidlo Weller Lötfett F-SW21, prodávané ve formě pasty. Jeho hlavní nevýhodou je vysoká vodivost, tavidlo je rozpustitelné lihem a jeho dokonalé vyčištění je nutností.

5.2.1 Fotodokumentace osazené PCB



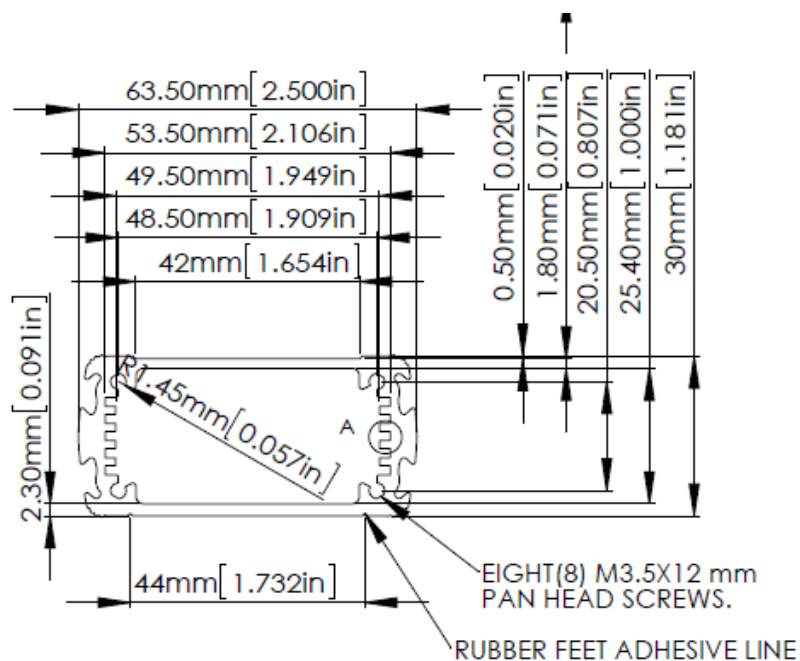
Obrázek 14 - Osazené PCB dolní strana



Obrázek 15 - Osazené PCB horní strana

6 MECHANICKÁ KONSTRUKCE ZAŘÍZENÍ

Pro lepší design, mechanickou odolnost a dokonalé stínění vůči rušivým vlivům byl základ krabičky zakoupen z hliníku. Originální čelní a zadní části byly nahrazeny vlastními kryty (vytištěnými na 3D tiskárně), uzpůsobenými pro mírně větší desku (než je hliníkový profil) a konektory. Výkres (Obrázek 16) zobrazuje pohled zepředu.



Obrázek 16 - Výkres hliníkové součásti konstrukce [6]

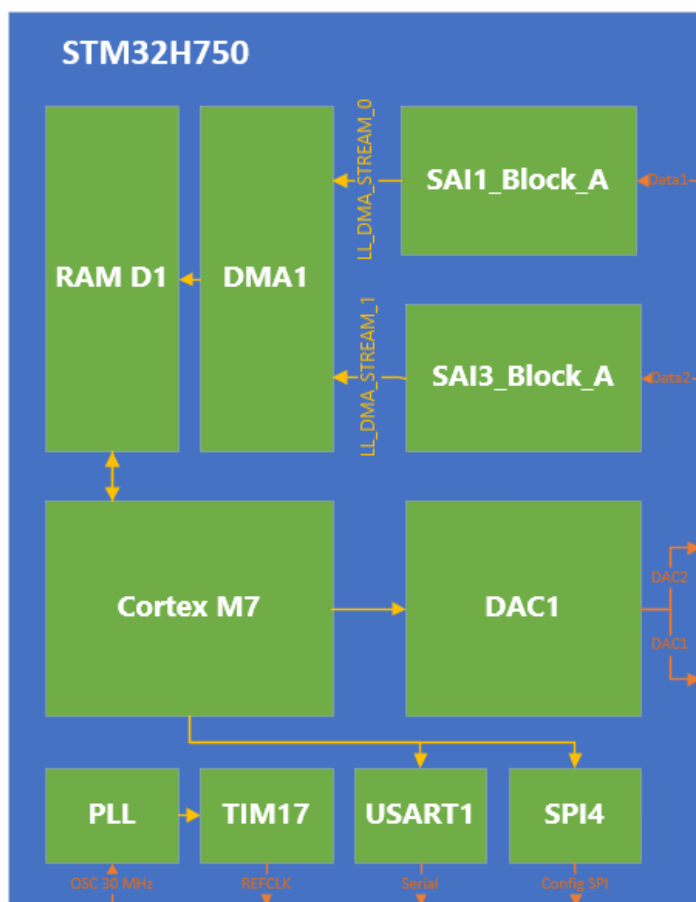
Plastové díly jsem navrhnul v programu Inventor. CAD soubory a STL jsou součástí přílohy. Obrázek 17 ukazuje pohled na sestavené zařízení.



Obrázek 17 - Pohled na vyvinuté zařízení

7 PROGRAM MCU

Program pro MCU jsem vytvořil v jazyku C, s využitím knihoven HAL a LL. Pro dosažení nejvyšší možné rychlosti bylo klíčové optimální využití periférií. Obrázek 18 zobrazuje využití periférie a jejich propojení. Celý kód je uveden v příloze.



Obrázek 18 - Blokové schéma využitých periférií MCU

7.1 Konfigurace periférií

Mikrokontroler STM32H7 disponuje širokou řadou periférií, jejichž využití a konfigurace je klíčová pro optimalizaci rychlosti činnosti programu. Základní konfiguraci periférií a rozvržení pinů jsem provedl za použití STM32Cube generátoru kódu v STM32CubeIDE. Samozřejmostí je konfigurace příslušných pinů (GPIO) pro periférie, tu nebudu pro jednoduchost v následujících ukázkách kódu uvádět.

7.1.1 RCC

Konfiguraci jsem navrhnul v grafickém prostředí a nechal vygenerovat s využitím HAL (Hardware Abstraction Layer) knihoven.

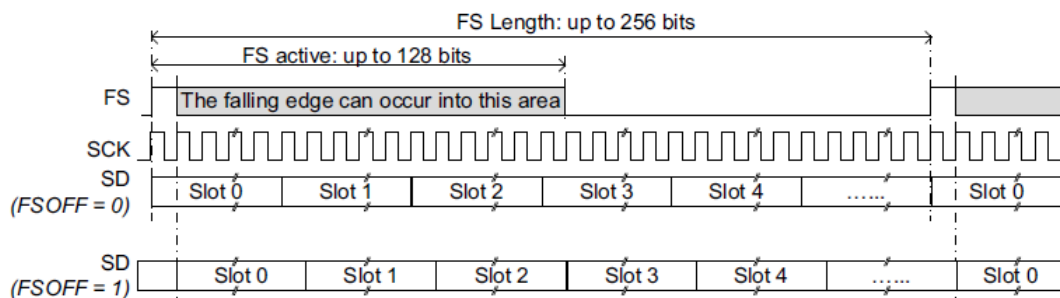
Všem použitým periferiím jsem nakonfiguroval nejvyšší dovolený takt.

Jádro Cortex-M7 umožňuje práci až na 480 MHz, tím dosáhneme nejvyššího možného výkonu.

Komunikační periferie SAI jsou limitovány na 150 MHz, jedná se o nejpomalejší prvek příjmu dat. LCLK takt pro TDC je generován za pomoci MCO2 pinu a je volitelným zlomkem taktu pro SAI periferii.

7.1.2 SAI

Periferie SAI je určena pro přenos zvuku, její adaptace vhodnou konfigurací je ideální volbou pro sbírání dat z TDC. Časový diagram (Obrázek 19) zobrazuje část konfigurovatelných parametrů přenosu.



Obrázek 19 - Časový diagram SAI [8]

```
hsai_BlockA1.Instance = SAI1_Block_A;  
hsai_BlockA1.Init.Protocol = SAI_FREE_PROTOCOL; //No audio protocol  
hsai_BlockA1.Init.AudioMode = SAI_MODESLAVE_RX; //RX only  
hsai_BlockA1.Init.DataSize = SAI_DATASIZE_32; //32bit data  
hsai_BlockA1.Init.FirstBit = SAI_FIRSTBIT_MSB; //MSB first  
hsai_BlockA1.Init.ClockStrobing = SAI_CLOCKSTROBING_RISINGEDGE; //Read on R. edge  
hsai_BlockA1.Init.Synchro = SAI_ASYNCHRONOUS; //Async. to internal clock  
hsai_BlockA1.Init.OutputDrive = SAI_OUTPUTDRIVE_DISABLE; //No output  
hsai_BlockA1.Init.FIFOThreshold = SAI_FIFOTHRESHOLD_EMPTY; //No fifo  
hsai_BlockA1.Init.SynchroExt = SAI_SYNCEXT_DISABLE; //No output  
hsai_BlockA1.Init.MonoStereoMode = SAI_MONOMODE; //1 channel  
hsai_BlockA1.Init.CompandingMode = SAI_NOCOMPANDING;  
hsai_BlockA1.Init.TriState = SAI_OUTPUT_RELEASED; //No output  
hsai_BlockA1.Init.PdmInit.Activation = DISABLE;  
hsai_BlockA1.Init.PdmInit.MicPairsNbr = 1;  
hsai_BlockA1.Init.PdmInit.ClockEnable = SAI_PDM_CLOCK1_ENABLE;  
hsai_BlockA1.FrameInit.FrameLength = 32; //32bit data  
hsai_BlockA1.FrameInit.ActiveFrameLength = 7; //Frame signal length  
hsai_BlockA1.FrameInit.FSDefinition = SAI_FS_STARTFRAME; //FS on first bits  
hsai_BlockA1.FrameInit.FSPolarity = SAI_FS_ACTIVE_HIGH; //FS active on H  
hsai_BlockA1.FrameInit.FSOffset = SAI_FS_FIRSTBIT; //No offset  
hsai_BlockA1.SlotInit.FirstBitOffset = 0;  
hsai_BlockA1.SlotInit.SlotSize = SAI_SLOTSIZE_DATASIZE;
```

```
hsai_BlockA1.SlotInit.SlotNumber = 1; //1 slot
hsai_BlockA1.SlotInit.SlotActive = 0x00000001; //Activate slot
```

7.1.3 DMA

DMA zajišťuje přesun dat mezi SAI periferií a paměti RAM_D1 bez interakce jádra. Periferii využívám za pomoci LL knihoven. Pro spuštění přenosu je potřeba požadovaný stream aktivovat a povolit v SAI DMA přenos.

```
LL_DMA_StructInit(&dma);
dma.Direction = LL_DMA_DIRECTION_PERIPH_TO_MEMORY; //direction of data transfer
dma.Mode = LL_DMA_MODE_CIRCULAR; //go to first after full array
dma.MemoryOrM2MDstAddress = taga; //address of buffer array
dma.MemoryOrM2MDstDataSize = LL_DMA_MDATAALIGN_WORD; //32bit data
dma.MemoryOrM2MDstIncMode = LL_DMA_MEMORY_INCREMENT; //buffer increment
dma.PeriphOrM2MSrcAddress = (uint32_t)&(SAI1_Block_A->DR); //data source address
dma.PeriphOrM2MSrcDataSize = LL_DMA_PDATAALIGN_WORD; //32bit data
dma.PeriphOrM2MSrcIncMode = LL_DMA_PERIPH_NOINCREMENT; //Read from same address
dma.NbData = sizeof(taga)/sizeof(taga[0]); //Buffer length
dma.PeriphRequest = LL_DMAMUX1_REQ_SAI1_A; //Select DMA trigger
dma.Priority = LL_DMA_PRIORITY_MEDIUM; //does not matter
LL_DMA_Init(DMA1, LL_DMA_STREAM_0, &dma);
```

7.1.4 DAC

DAC nevyžaduje složitou konfiguraci, protože výstupní hladina nebude nijak měněna. Nutností je výběr referenčního napětí; generátor kódu CubeMX zde dělá chybu (hodnota vybraná v grafickém rozhraní neodpovídá reálné).

```
//CubeMX chooses wrong value here!
HAL_SYSCFG_VREFBUF_VoltageScalingConfig(SYSCFG_VREFBUF_VOLTAGE_SCALE0); //Sets reference voltage to 2.5V

DAC_InitStruct.TriggerSource = LL_DAC_TRIG_SOFTWARE; //No internal HW trigger
DAC_InitStruct.WaveAutoGeneration = LL_DAC_WAVE_AUTO_GENERATION_NONE; //Manual control of DAC
DAC_InitStruct.OutputBuffer = LL_DAC_OUTPUT_BUFFER_ENABLE; //Reduces output range, increases current
DAC_InitStruct.OutputConnection = LL_DAC_OUTPUT_CONNECT_GPIO; //Output on GPIO
DAC_InitStruct.OutputMode = LL_DAC_OUTPUT_MODE_NORMAL; //No need of external HW
LL_DAC_Init(DAC1, LL_DAC_CHANNEL_1, &DAC_InitStruct);

LL_DAC_Enable(DAC1, LL_DAC_CHANNEL_1); //Enables DAC

LL_DAC_ConvertData12RightAligned(DAC1, LL_DAC_CHANNEL_1, 491); //Set output value
```

7.1.5 USART

Konfigurace USART pro komunikaci s počítačem značně záleží na cílové aplikaci. Během testování jsem neměl vysoké nároky na přenosovou rychlost do počítače, proto využití flow-control nebylo nutností.

```

USART_InitStruct.PrescalerValue = LL_USART_PRESCALER_DIV1; //No clock div.
USART_InitStruct.BaudRate = 115200; //Boudrate selection - standart
USART_InitStruct.DataWidth = LL_USART_DATAWIDTH_8B; //1 byte width
USART_InitStruct.StopBits = LL_USART_STOPBITS_1;
USART_InitStruct.Parity = LL_USART_PARITY_NONE;
USART_InitStruct.TransferDirection = LL_USART_DIRECTION_TX_RX; //Both send and receive
USART_InitStruct.HardwareFlowControl = LL_USART_HWCONTROL_NONE; //Flow control not necessary
USART_InitStruct.OverSampling = LL_USART_OVERSAMPLING_16;
LL_USART_Init(USART1, &USART_InitStruct);

LL_USART_DisableFIFO(USART1); //Disable fifo

LL_USART_ConfigAsyncMode(USART1);
LL_USART_EnableIT_RXNE_RXFNE(USART1); //Enable RX interrupt
LL_USART_Enable(USART1); //enable USART

```

7.1.6 SPI

SPI periferie v cílové aplikaci není ve finále využita a je využíváno SPI za použitím bit-bang. Následující ukázka funkce se stará o odesílání dle požadovaného standardu Motorola SPI.

```

void send_byte_to_SPI(char data)
{
    for(uint8_t i = 0; i<8; i++)
    {
        LL_GPIO_SetOutputPin(GPIOE, LL_GPIO_PIN_12); //CLK
        if(data & (1<<(7-i)))
        {
            LL_GPIO_SetOutputPin(GPIOE, LL_GPIO_PIN_14); //MOSI
        }
        else
        {
            LL_GPIO_ResetOutputPin(GPIOE, LL_GPIO_PIN_14); //MOSI
        }
        LL_GPIO_ResetOutputPin(GPIOE, LL_GPIO_PIN_12); //CLK
    }
}

```


8 ZPRACOVÁNÍ ZMĚŘENÝCH DAT

Pomocí kombinace SAI a DMA se přenáší do paměti (RAM_D1) časové značky ve formě 32bitového čísla čísla. Mým algoritmem v datech nacházím nejbližší páry záznamů (start a stop). A počítám čas mezi nimi.

Ze změřených intervalů interně v MCU vytvářím histogram s maximální šířkou zvolenou okolo 1 us, ~6ps rozlišením (lze jednoduše změnit úpravou FW zařízení) a volitelným začátkem histogramu (lze měřit čas nejen v rozsahu 0–1 us, ale např. 4–5 us).

Histogram je následně po VCP přenesen do počítače, kde může být dále zpracováván.

Pro základní zpracování a zobrazení naměřených dat využívám Python, díky jeho jednoduchosti a široké nabídce knihoven určených pro požadované účely. Pro příjemnější práci se skriptovacím jazykem jsem využil prostředí JupyterLab. Python je nejčastějším skriptovacím jazykem pro podobné aplikace a je obecně známý cílovou skupinou lidí. V příloze lze najít několik skriptů pro obsluhu zařízení.

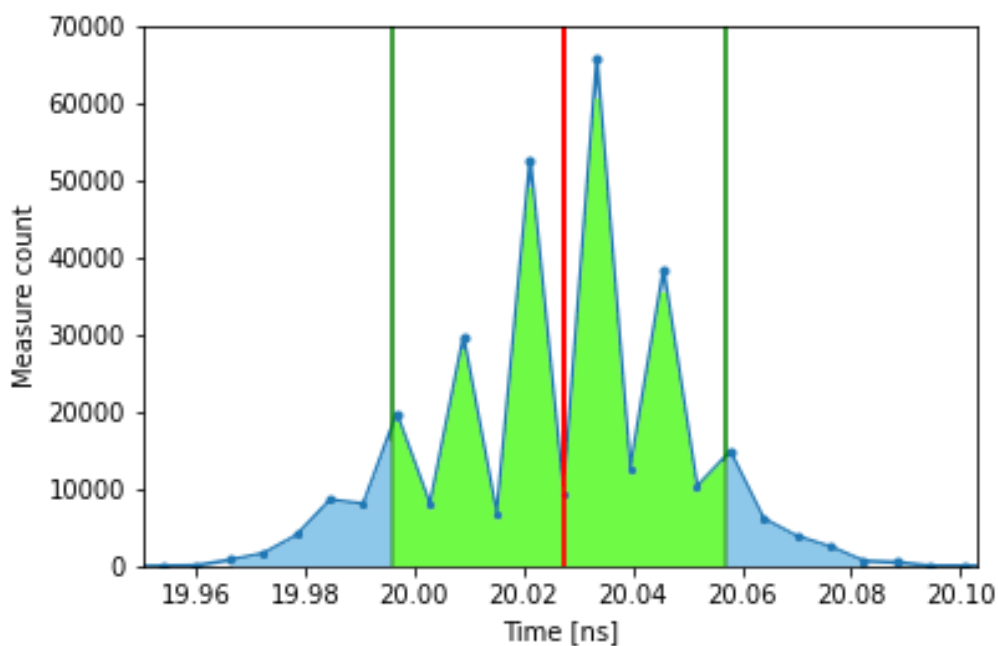
8.1 Vyobrazení dat a stanovení rozptylu hodnot

Po přijetí histogramu do počítače je z dat vykreslen graf pro vizuální kontrolu změřených intervalů a spočítán jejich průměr. Rozptyl měření je určován programem, který porovnává počet vzorků v testovaném intervalu (zelená plocha) s celkovým počtem vzorků měření (plocha pod křivkou), viz Obrázek 20. Je-li poměr v testovaném intervalu (střed intervalu je v průměrné hodnotě) menší než požadovaný poměr, testovaný interval se rozšíří. Za rozptyl hodnot považuji nejužší interval s větším poměrem vzorků, než je zvolený. Pro charakterizaci je využita hodnota poměru 80 % jako referenční.

Pro kreslení grafů jsem využil knihovnu matplotlib.pyplot, která nabízí širokou škálu možností. Téměř samozřejmostí je knihovna NumPy, která je velkým pomocníkem při práci s daty a při matematických i statistických výpočtech.

Příklad výstupu:

```
Time: 20.027 ns (from 19.997 to 20.058 ns)
Deviation for 80.0 % measuments is 61.035 ps
```



Obrázek 20 - Histogram 80% měření

Zelené vertikální čáry označují ohraničení požadované plochy. Červená označuje průměr měření.

8.1.1 Skripty pro práci s daty

Všechna zajímavá měření jsem chtěl mít možnost uložit, a proto jsem napsal skript, který stáhne histogram ze zařízení a uloží jej ve formátu Pythonového pole do textového dokumentu, aby bylo možné je později pouze zkopírovat a dále zpracovávat. K tomu jsem využil knihovnu „simplejson“.

Pro následné zpracování dat jsem vytvořil druhý skript s grafickým výstupem, který lze vidět výše.

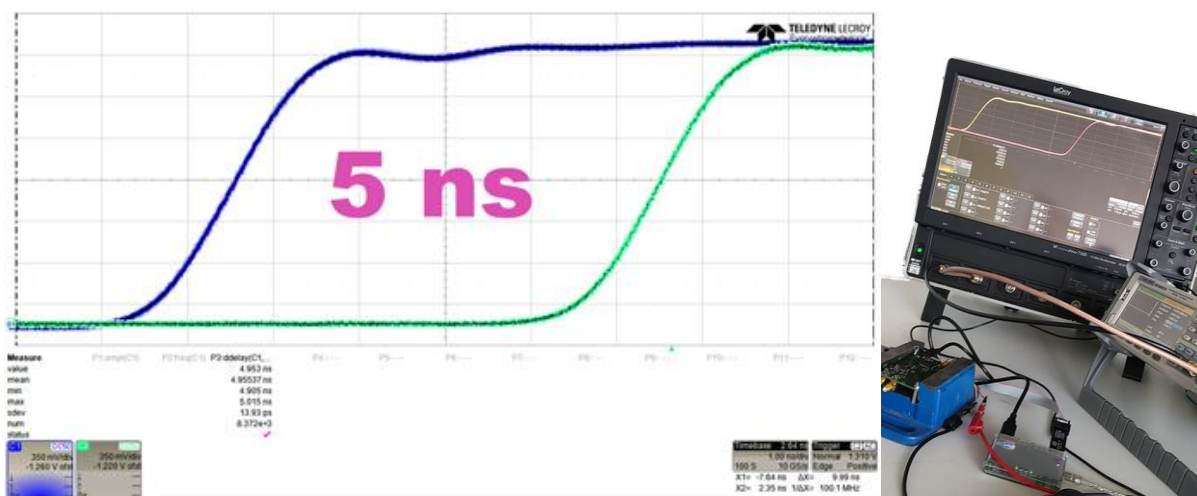
9 CHARAKTERIZACE

V následující kapitole měřeními charakterizují vyvinuté zařízení. Klíčovým parametrem zařízení je úzký rozptyl změřených hodnot, další klíčovou vlastností je minimální odstup pulzů za sebou, a to jednak z pohledu dlouhodobého průměru, jednak v burstu několika málo pulzů.

9.1 Rozptyl měřených hodnot

Jednou z hlavních charakteristik je rozptyl měřených hodnot kolem průměru. Za pomoci DDS generátoru funkcí (SDG6022) jsem vyrobil dvojici obdélníkových signálů s fázovým posunem vůči sobě a měřil jsem absolutní čas mezi náběžnými hranami dvou signálů. Sledovat jednak absolutní čas ve srovnání s osciloskopem, jednak rozptyl pro 80 % měření.

Správnost signálu ověřuji dostatečně rychlým osciloskopem LeCroy (20 GSa/s).



Obrázek 21 - Snímek obrazovky z DSO a testovací souprava

Změřená data byla vynesena do grafu četnosti jednotlivých změřených časů, tzv. histogramu. Z výsledků změřených dat jsem vytvořil přehlednou tabulku.

Tabulka 3 - Výsledky měření

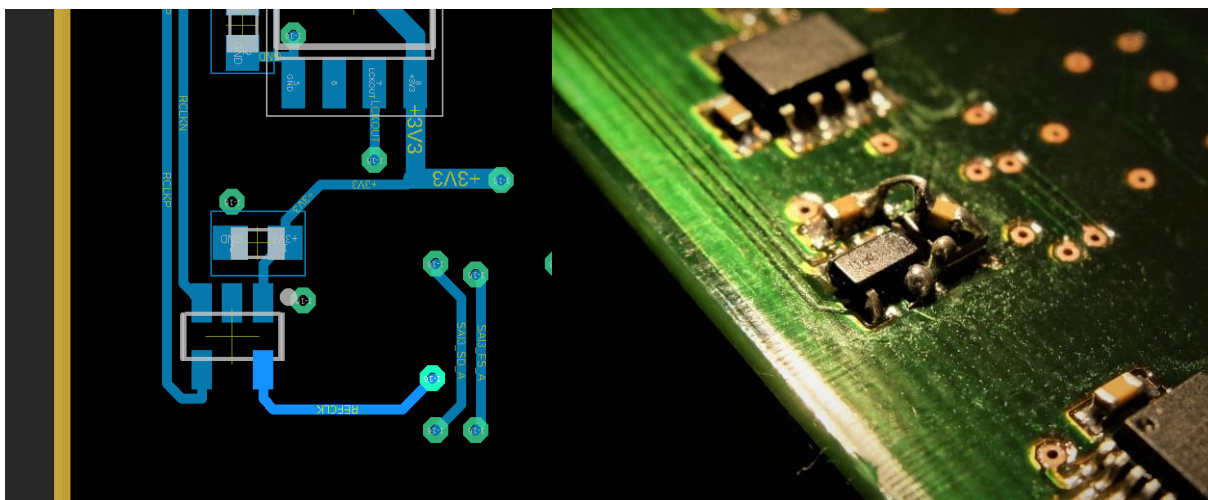
Generátor [ns]	Osciloskop [ns]	Průměr měření (PLL) [ns]	Rozptyl pro 80% měření	
			PLL [ps]	OSC [ps]
1	0,983	0,965	42,7	
2	1,976	1,966	48,8	
3	2,958	2,986	54,9	
4	3,870	3,997	48,8	
5	4,955	4,982	48,8	36,6
7	6,714	6,996	61,0	48,8
6	5,932	5,983	54,9	48,8
8	7,971	7,997	61,0	
9	8,972	8,983	48,8	61,0
10	9,970	9,980	48,8	48,8
20	19,987	20,003	67,1	61,0
50	49,989	49,974	116,0	109,9
100	99,990	99,967	134,3	134,3
200	199,991	199,977	79,3	48,8
500	499,991	499,967	207,5	
1000	999,992	999,965	244,1	48,8
2000	1999,994	1999,961	366,2	
5000	4999,959	4999,949	457,8	

9.2 Rozdílný rozptyl měřených hodnot

S rostoucím měřeným intervalem lze pozorovat značně rostoucí rozptyl změřených hodnot. Hlavním důvodem rozdílnosti rozptylu je měřitelný jitter oscilátoru, jeho nestabilita se zvětšuje po průchodu signálu frekvenční násobičkou (a děličkou) v MCU. Samozřejmostí je korelativní vztah mezi jitterem referenčního taktu (REFCLK) a rozptylem změřených hodnot. To se projevuje pouze u měření intervalů o délce větší než půl periody REFCLK.

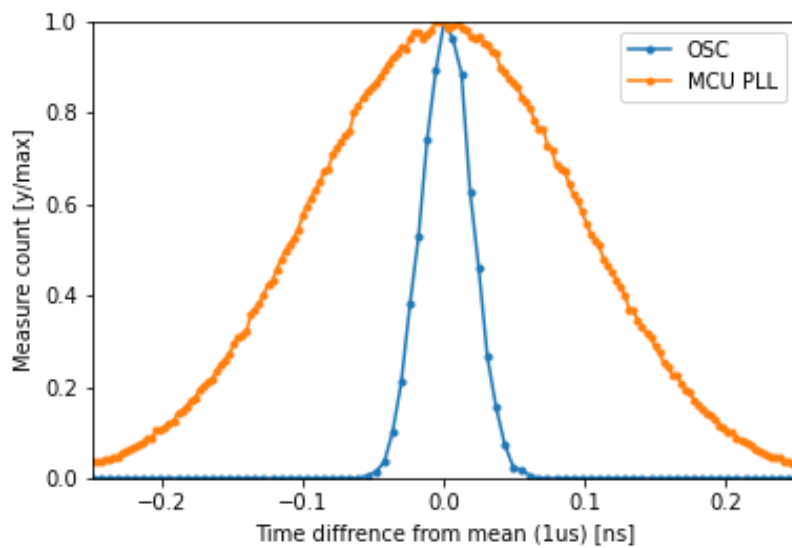
9.2.1 Úprava REFCLK

Z důvodu eliminace uvedeného problému jsem upravil HW přidáním druhého oscilátoru (Obrázek 22) s frekvencí 5 MHz. Díky tomu jsem se zbavil problému s širokým rozptylem hodnot u delších časů na úkor ztráty možnosti úpravy frekvence za pomoci FW, tzn. nelze ovlivnit, jak často přeteče interní čítač v AS6501 jinak, než nastavením bitové hloubky dat. To omezuje maximální měřený čas.



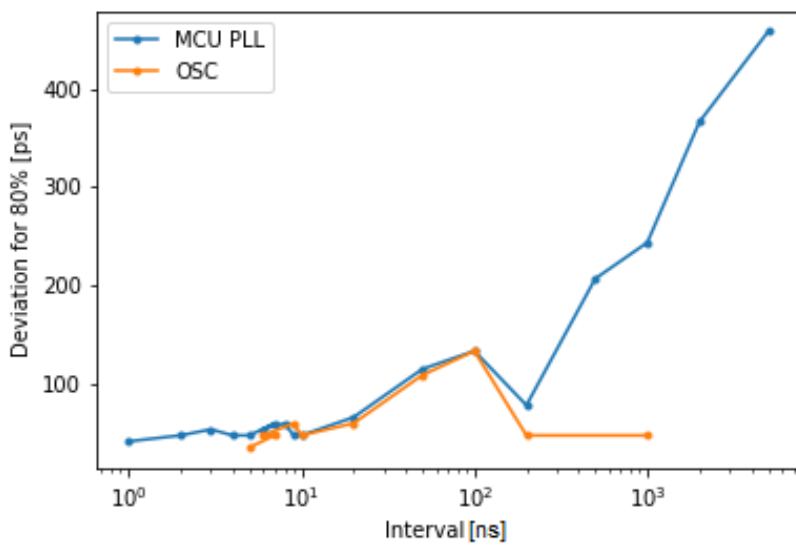
Obrázek 22 - Úprava REFCLK

Přidání uvedeného oscilátoru vhodné frekvence vedlo k značnému zlepšení výsledků měření delších časů. Následující graf (Obrázek 23) ukazuje srovnání měření za pomoci REFCLK generovaného MCU (s využitím PLL a TIM) vůči přidanému oscilátoru. Osa X vyjadřuje odchylku od průměrné hodnoty, osa Y podíl četnosti příslušné odchylky vůči maximu.



Obrázek 23 - Srovnání PLL s přidaným OSC

Následující graf (Obrázek 24) popisuje vztah mezi měřeným intervalem a rozptylem pro 80 % měření. Modrá zobrazuje měření za využití REFCLK přes MCU a oranžová s přidaným oscilátorem.

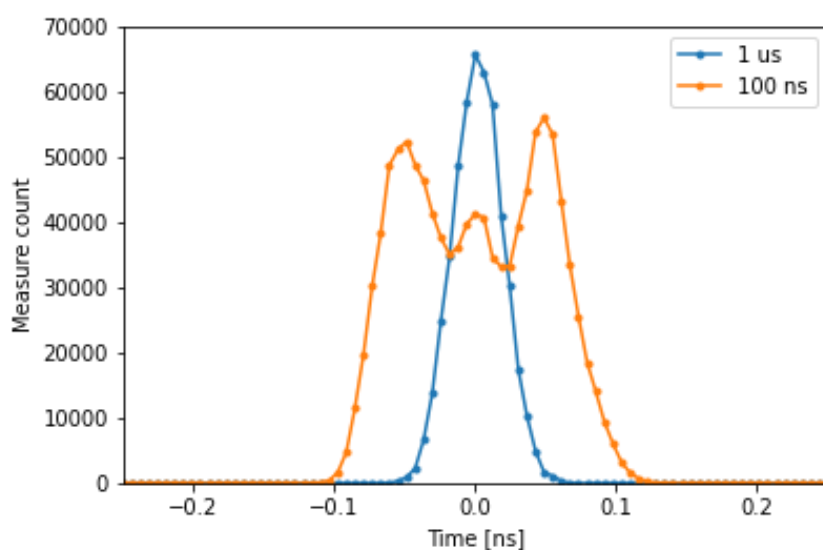


Obrázek 24 - Závislost rozptylu hodnot a délce měřeného intervalu

Z grafu lze stále pozorovat zvláštní odchylky měření v oblasti okolo 100 ns.

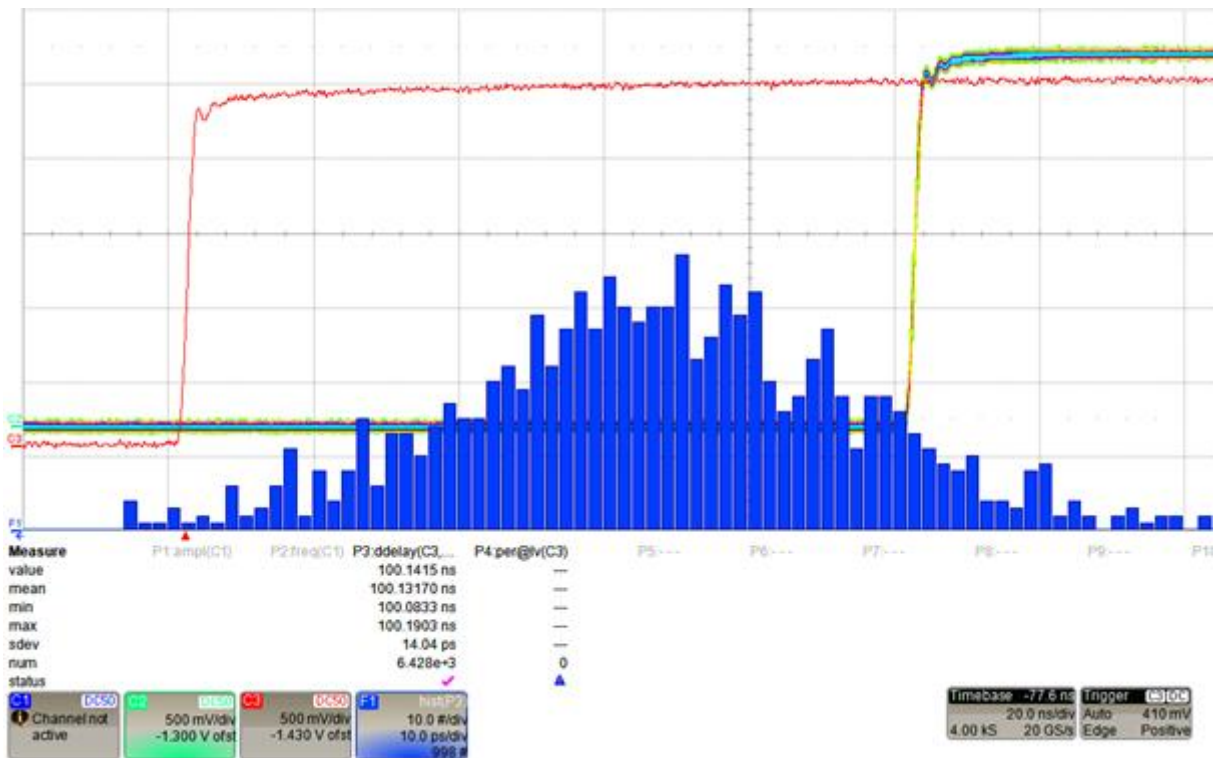
9.2.2 Další nedostatky

Během měření (především s upraveným generátorem časové základny) bylo možné stále pozorovat značné odlišnosti v rozptylu změřených dat (Obrázek 25). Jednoznačný (např. přímo úměrný) vztah mezi měřeným intervalem a rozptylem nebyl nadále zřetelný. Při měření intervalu okolo 100 ns lze pozorovat až více než trojnásobný rozptyl měřených dat (ve srovnání s 1 us) a bylo možné pozorovat dva téměř totožné vrcholy histogramu. Tato charakteristika vykazuje jednoznačnou opakovatelnost. Na ose Y je četnost změřené hodnoty a osa X vyjadřuje odchylku od průměru měřených hodnot.



Obrázek 25 - Srovnání rozptylu hodnot pro 1 us a 100 ns

Za nejpravděpodobnější zdroj „dvojitého vrcholu“ považuji použitý generátor SDG6022 se vzorkovací rychlostí 2,4 GSa/s.



Obrázek 26 - Snímek obrazovky DSO histogram signálu

Bohužel tuto hypotézu se mi nepovedlo jednoznačně potvrdit ani za využití dostatečně rychlého osciloskopu (Obrázek 26 – modře histogram změřený osciloskopem) a nemám k dispozici kvalitnější zdroj signálu požadovaných parametrů, případně jinou možnost jeho ověření.

V příloze jsou uvedeny surová data měření i zpracované histogramy. Výsledky všech měření jsou uvedeny v tabulce výše.

10 OVLÁDÁNÍ PŘÍSTROJE

Po připojení USB k počítači a napájení (6–8 V) z lineárního zdroje, se automaticky instalují ovladače pro integrovaný převodník na sériovou komunikaci a zařízení je připraveno k použití. V případě použití operačního systému bez nativní podpory ovladačů pro použitý čip je možné je stáhnout na webových stránkách výrobce.

Výrobek je možné ovládat za využití sériového terminálu (např. Realterm) nebo vlastní aplikace, s využitím běžných knihoven pro COM port. V příloze lze najít ukázkové Python skripty pro práci se zařízením. Rychlost přenosu je 115200 baudů a standardní konfigurace sériového portu (1 stop bit, bez parity).

10.1 Příkazy pro ovládání

g_hist_raw – Vrací všechny nenulové sloupce histogramu ve formátu:

```
123456-65535 \n
```

První číslo představuje index sloupce (z intervalu 0 až 169000), druhé jeho výšku (16bit).

set_start = 123456 – Nastavuje čas počátku histogramu. Využívá se při měření delších časových intervalů než 1 us. Jednotkou je nejmenší rozlišitelný interval. Zařízení odpovídá potvrzením:

```
OK - start set to 100 ns
```

Číslo představuje přepočtený začátek na nanosekundy.

get_stats – Vrací obecné statistiky měření (počet měření mimo rozsah histogramu, počet měření v histogramu, výška nejvyššího sloupce a jeho index, upozornění na sloupec o maximální výšce, tj. 0xFFFF). Po dosažení maxima v jednom sloupci se měření zastaví.

Formát:

```
Out of range: 123 (12 ppm)\r
Hist peak: 12345\r
Hist peak index: 123456\r
!!! FULL !!!\r
```

g_start – Vrací nastavený začátek histogramu v počtu nejmenších rozlišitelných vzorků.

```
123456\r
```

clear – Vymaže změřená data a znovu aktivuje měření.

```
OK - cleared\r
```

11 MOŽNOSTI DALŠÍHO VÝVOJE

Pro urychlení zpracování dat a jejich transport je výhodné využití FPGA (hradlových polí).

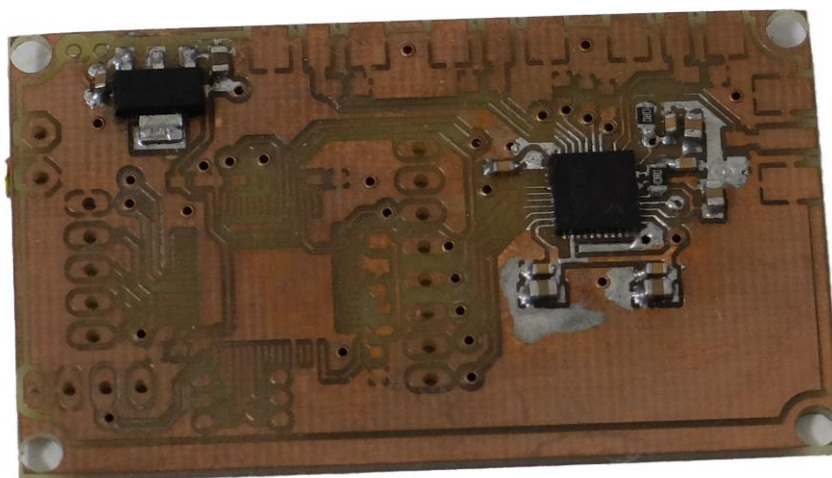
Využitím více kanálového TDC lze kombinací kanálů snížit rozestup mezi pulzy.

Jeden nebo dva výstupní kanály (do 50 Ω) s možností přesného časování umožní širší využití vyvinutého zařízení. Příkladem je měření defektů ve vodičích.

12 PROBLÉMY S REALIZACÍ

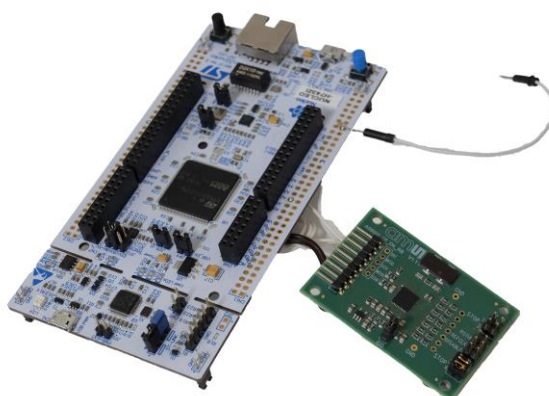
12.1 Testovací HW

Pro prvotní testy jsem navrhnul prototyp PCB s STM32F4xx a AS6500, využitá technologie výroby PCB (použití CNC frézy) se vzhledem k pouzdru TDC (QFN-40) ukázala jako nedostačující. Během osazování se tuto část nepodařilo správně osadit a oživit (Obrázek 27), proto jsem dále tento prototyp dále nevyužíval.



Obrázek 27 - Prototyp HW AS6500

V době návrhu nebyl k dispozici vývojový kit (AS6500-DK), který se později podařilo koupit (Obrázek 28). Další testování probíhalo na něm spolu s Nucleo-F743. Ten nabízí jeden z nejvýkonnějších MCU na trhu.



Obrázek 28 - Vývojové kity

Čip se mi podařilo vyzkoušet bez větších komplikací a rozhodl jsem se jej (respektive jeho verzi s diferenciálním rozhraním) použít ve finální aplikaci. TDC již během testů vykazoval výborné vlastnosti.

12.2 Nekvalitní pájecí maska

I přes výrazně vyšší cenu výroby (ve srovnání s firmou jako je JLCPCB), je zřejmá výrazně nižší „ostrost“ pájecí masky. Je potřeba s tím počítat během návrhu i osazování. Cín se lehce spojuje například mezi menšími piny a polygonem okolo, protože maska nesahá až k jeho kraji, jak je vidět na fotce (Obrázek 29) z mikroskopu níže. Přesah masky je nastavitelný před exportem výrobních souborů gerber a je potřeba „rozmazání“ zohlednit. Oba snímky jsou pod stejným zvětšením.



Obrázek 29 - Srovnání pájecí masky

13 ZÁVĚR

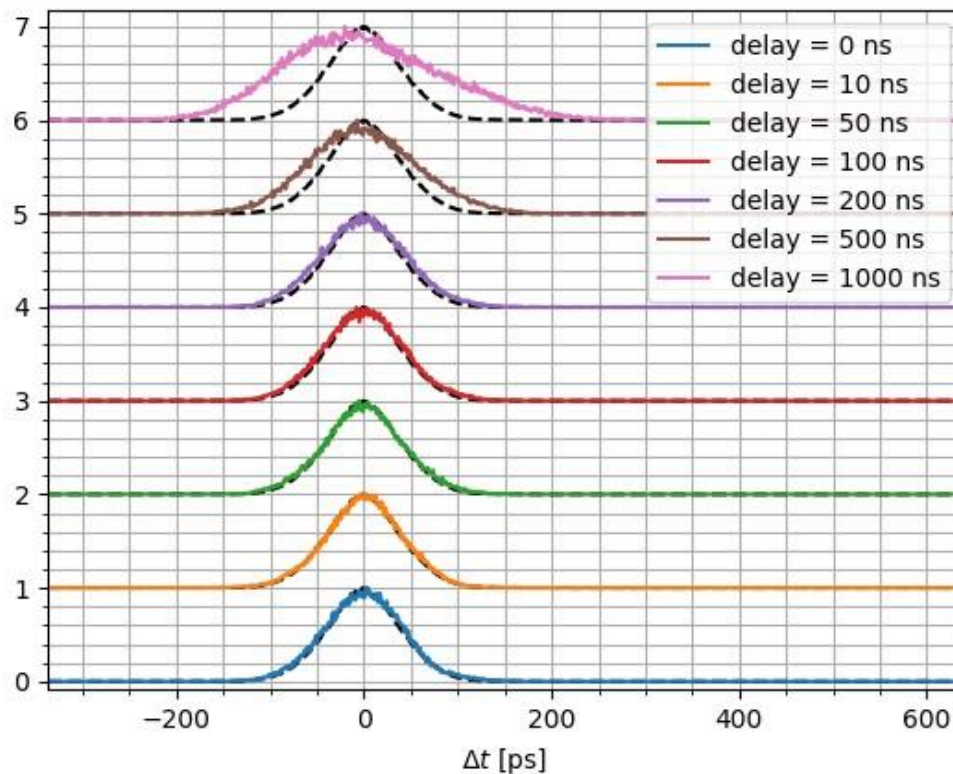
Mojí prací se podařilo navrhnout a realizovat HW pro přesné měření času a optimalizovat přenos mezi měřícím čipem TDC a mikrokontrolerem. Dále jsem vytvořil FW pro zařízení, jehož cílem je měření časů mezi pulzy na dvou kanálech a vytvoření histogramu ze změřených intervalů, ten je následně přenesen do počítače, kde může být dále zpracován. Díky tomu se minimalizuje objem přenášených dat a zjednodušuje práce se zařízením.

13.1 Srovnání výrobku s komerční alternativou

Komerční zařízení do počítače posílají pouze časové značky. Tím je práce se zařízením ve specifických aplikacích výrazně složitější a zdlouhavější. V případě potřeby, úpravou FW je možné dosáhnout stejné funkce u mého zařízení. Níže je mé řešení srovnáno s komerční alternativou Time Tagger 20.

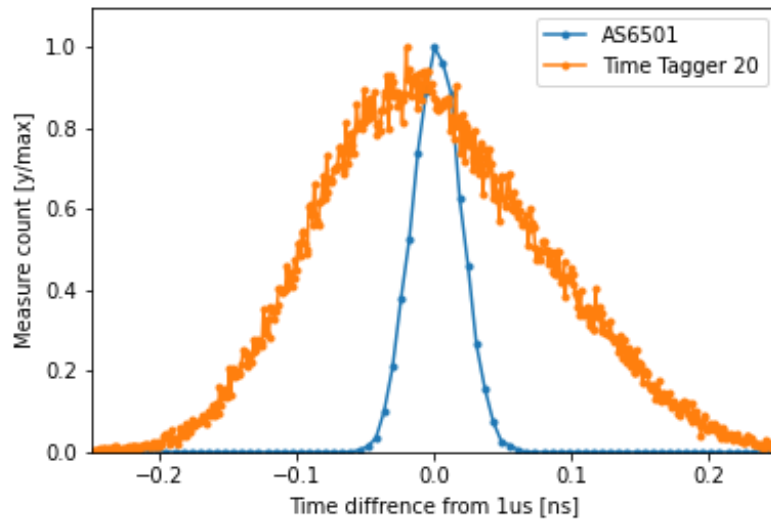
Výhodou mého zařízení je výrazně nižší pořizovací cena, která se pohybuje pod hranicí 3000 Kč, vysoká flexibilita (možná úpravou FW) a jeho jednoduchá obsluha po virtuálním sériovém portu. Jeho nevýhodou jsou pouze dva kanály a nižší datová prostupnost.

Následující graf ukazuje výsledky měření z Time Tagger 20 za využití stejného generátoru signálu.



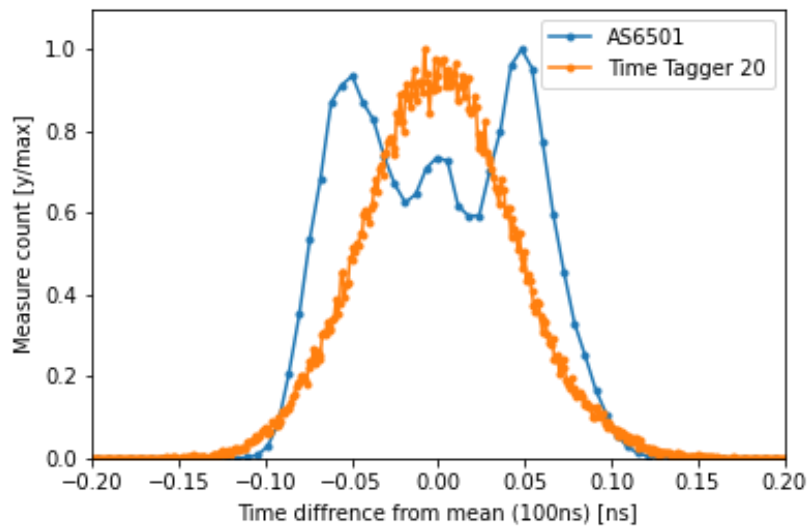
Obrázek 30 - Histogram měření signálu z Time Tagger 20

V grafu (Obrázek 30) lze vidět histogram měření několika časů („delay“) pod sebou. Dolní graf představuje měření při příchodu obou signálů současně. Lze vidět značně rostoucí rozptyl měřených hodnot při snímání delších časových intervalů. Následující graf (Obrázek 31) zobrazuje srovnání uvedeného zařízení s mojí konstrukcí, při snímání intervalu 1 us. (Pro měření byly použité různé kabely, to může způsobit malý posun v absolutním čase.)



Obrázek 31 - Srovnání s komerční alternativou 1 us

Během charakterizace jsem pozoroval nejhorší výsledky mého zařízení v oblasti okolo 100 ns. Následující graf (Obrázek 32) poskytuje srovnání dvou zařízení. Tento signál lze pro mé zařízení považovat za „worst-case scenario“.



Obrázek 32 - Srovnání s komerční alternativou 100 ns

13.2 Aplikace zařízení

13.2.1 APD detektory

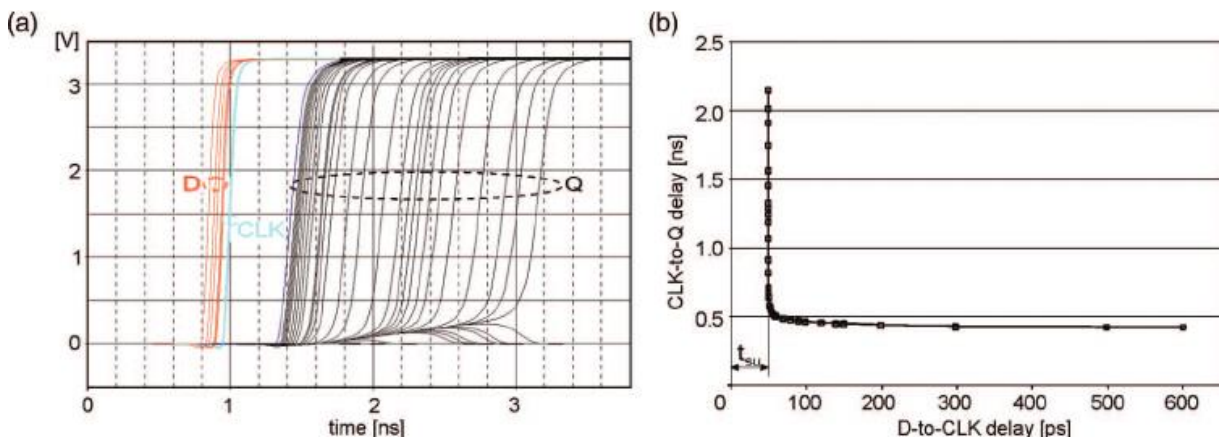
Lavinovou fotodiodou (APD), při dopadu fotonu a závěrném napětí na ní, začne lavinovým efektem růst průtok proudu. Za využití mého zařízení je možné změřit čas mezi vysláním

fotonu a zaznamenáním jeho dopadu na APD detektor. Známost zpoždění reakce detektoru je důležitá pro řadu optických experimentů.

13.2.2 Metastable region D klopného obvodu

Každý D klopný obvod má uvedený minimální čas příchodu hrany dat D před aktivní hranou taktu CLK. Dojde-li k porušení uvedené specifikace, zpoždění mezi výstupem Q a taktům CLK se může značně lišit (Obrázek 33a). S menším odstupem vstupních hran od sebe se prodlužuje reakce klopného obvodu (Obrázek 33b). Výše uvedený jev se nazývá „metastable region“.

Za využití mého zařízení lze uvedené zpoždění měřit s dostatečnou přesností i rozlišením.



Obrázek 33 - Metastable region

14 SEZNAM OBRÁZKŮ

OBRÁZEK 1 - BLOKOVÉ SCHÉMA VYVINUTÉHO ZAŘÍZENÍ	10
OBRÁZEK 2 - SROVNÁNÍ KOMPARÁTORŮ [12][13]	12
OBRÁZEK 3 - BLOKOVÉ SCHÉMA TDC [7]	13
OBRÁZEK 4 - BLOKOVÉ SCHÉMA KOMBINACE KANÁLŮ 1 [7]	16
OBRÁZEK 5 - BLOKOVÉ SCHÉMA KOMBINACE KANÁLŮ 2 [7]	17
OBRÁZEK 6 - ČASOVÝ DIAGRAM KOMUNIKACE SPI [7]	17
OBRÁZEK 7 - ČASOVÝ DIAGRAM VÝSTUPNÍCH DAT [7]	19
OBRÁZEK 8 - SCHÉMA KOMPARÁTORŮ	20
OBRÁZEK 9 - SCHÉMA NAPÁJENÍ KOMPARÁTORŮ	21
OBRÁZEK 10 - SCHÉMA FTDI	22
OBRÁZEK 11 – SCHÉMA TDC	23
OBRÁZEK 12 - SCHÉMA MIKROKONTROLERU	24
OBRÁZEK 13 - ROZVRŽENÍ PCB	26
OBRÁZEK 14 - OSAZENÉ PCB DOLNÍ STRANA	27
OBRÁZEK 15 - OSAZENÉ PCB HORNÍ STRANA	27
OBRÁZEK 16 - VÝKRES HLINÍKOVÉ SOUČÁSTI KONSTRUKCE [6]	28
OBRÁZEK 17 - POHLED NA VYVINUTÉ ZAŘÍZENÍ	28
OBRÁZEK 18 - BLOKOVÉ SCHÉMA VYUŽITÝCH PERIFERIÍ MCU	29
OBRÁZEK 19 - ČASOVÝ DIAGRAM SAI [8]	30
OBRÁZEK 20 - HISTOGRAM 80% MĚŘENÍ	34
OBRÁZEK 21 - SNÍMEK OBRAZOVKY Z DSO A TESTOVACÍ SOUPRAVA	35
OBRÁZEK 22 - ÚPRAVA REFCLK	37
OBRÁZEK 23 - SROVNÁNÍ PLL S PŘIDANÝM OSC	38
OBRÁZEK 24 - ZÁVISLOST ROZPTYLU HODNOT A DÉLCE MĚŘENÉHO INTERVALU	38
OBRÁZEK 25 - SROVNÁNÍ ROZPTYLU HODNOT PRO 1 US A 100 NS	39
OBRÁZEK 26 - SNÍMEK OBRAZOVKY DSO HISTOGRAM SIGNÁLU	40
OBRÁZEK 27 - PROTOTYP HW AS6500	43
OBRÁZEK 28 - VÝVOJOVÉ KITY	44
OBRÁZEK 29 - SROVNÁNÍ PÁJECÍ MASKY	44
OBRÁZEK 30 - HISTOGRAM MĚŘENÍ SIGNÁLU Z TIME TAGGER 20	46
OBRÁZEK 31 - SROVNÁNÍ S KOMERČNÍ ALTERNATIVOU 1 US	47
OBRÁZEK 32 - SROVNÁNÍ S KOMERČNÍ ALTERNATIVOU 100 NS	47
OBRÁZEK 33 - METASTABLE REGION	48

15 SEZNAM TABULEK

TABULKA 1 - SROVNÁNÍ AS6501 A AS6500	14
TABULKA 2 - REŽIMY VYSOKÉHO ROZLIŠENÍ TDC	16
TABULKA 3 - VÝSLEDKY MĚŘENÍ	36

16 SEZNAM ZKRATEK

APD – Lavinová fotodioda

CLK – Hodinový signál

CMOS – Complementary Metal–Oxide–Semiconductor

DAC – Digital to Analog Convertor

DDR – Double Data Rate

DMA – Direct Memory Access

DPS – Deska Plošných Spojů

FIFO – First In, First Out

FPGA – Field Programmable Gate Array

FW – Firmware

GPIO – General-purpose input/output

HAL – Hardware Abstraction Layer

HW – Hardware

LL – Low Level

LVDS – Low-voltage differential signaling

MCU – Mikrokontroler

PCB – Printed Circuit Board

PLL – Phase-Locked Loop

QSPI – Quad Serial Peripheral Interface

RAM – Random Access Memory

RCC – Reset and Clock Control

SAI – Serial Audio Interface

SPI – Serial Peripheral Interface

SSN – Slave Select Not

TDC – Time-to-digital convertor

USART – Universal Synchronous / Asynchronous Receiver and Transmitter

USB – Universal Serial Bus

VCP – Virtual COM Port

17 POUŽITÁ LITERATURA

1. Comparator. *Wikipedia* [online]. [cit. 2022-03-31]. Dostupné z: https://upload.wikimedia.org/wikipedia/commons/thumb/1/1a/Comparator_symbol.svg/904px-Comparator_symbol.svg.png
2. Serial Wire Debug (SWD). *SILICON LABS* [online]. 2021, Jul 9 [cit. 2022-03-31]. Dostupné z: https://community.silabs.com/s/article/serial-wire-debug-swd-x?language=en_US
3. KLEPACKI, K., R. SZPLET a R. PELKA. A 7.5 ps single-shot precision integrated time counter with segmented delay line. In: *Review of Scientific Instruments* [online]. 2014 [cit. 2022-03-31]. ISSN 0034-6748. Dostupné z: doi:10.1063/1.4868500
4. *Time Tagger Series* [online]. [cit. 2022-03-31]. Dostupné z: <https://www.swabianinstruments.com/time-tagger/>
5. DUDKA, Michal. *USART na STM32F0* [online]. 23. 7. 2017 [cit. 2022-01-31]. Dostupné z: http://www.elektromys.eu/clanky/stm_usart1/clanek.html
6. GAINTA INDUSTRIES LTD. *ALUG702XXXXX* [online]. 01. 09. 2015 [cit. 2022-01-31]. Dostupné z: <https://www.tme.eu/Document/202607e857ca6caa6f7cc20fa8ee9c60/ALUG702.pdf>
7. *AS6501* [online]. 11. 5. 2018, 63 [cit. 2022-03-31]. Dostupné z: https://cz.mouser.com/pdfdocs/AS6501_DS000586_1-00.pdf
8. *RM0433* [online]. February 2020, 3319 [cit. 2022-03-31]. Dostupné z: https://www.st.com/resource/en/reference_manual/dm00314099-stm32h742-stm32h743-753-and-stm32h750-value-line-advanced-arm-based-32-bit-mcus-stmicroelectronics.pdf
9. *STM32H750xB* [online]. February 2020, 337 [cit. 2022-03-31]. Dostupné z: <https://www.st.com/resource/en/datasheet/stm32h750xb.pdf>
10. HOROWITZ, Paul a Winfield HILL. *The art of electronics*. Third edition. New York: Cambridge University Press, 2015. ISBN 978-0-521-80926-9.
11. *AS6500* [online]. 28. 2. 2020, 55 [cit. 2022-03-31]. Dostupné z: https://cz.mouser.com/datasheet/2/1081/AS6500_DS000640_4_00-1890452.pdf
12. *TLV360x 800-ps High-Speed RRI Comparator with LVDS Outputs* [online]. JULY, 2021, 29 [cit. 2022-03-31]. Dostupné z: <https://www.ti.com/lit/ds/symlink/tlv3604.pdf>
13. *TLV3502-Q1: 4.5-ns Rail-to-Rail High-Speed Comparator* [online]. December, 2014, 23 [cit. 2022-03-31]. Dostupné z: <https://www.ti.com/lit/ds/sbos507a/sbos507a.pdf>
14. Time of flight. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2022-04-10]. Dostupné z: https://en.wikipedia.org/wiki/Time_of_flight

18 PŘÍLOHY

1. FW
 - 1.1. CubeMX-IDE_project.zip
2. HW
 - 2.1. AS6501.brd
 - 2.2. AS6501.sch
 - 2.3. AS6501.pdf
 - 2.4. AS6500_prototyp.brd
 - 2.5. AS6500_prototyp.sch
3. Mechanics
 - 3.1. Front_panel.ipt
 - 3.2. Front_panel.stl
 - 3.3. Back_panel.ipt
 - 3.4. Back_panel.stl
4. Results
 - 4.1. Raw_data_PLL.txt
 - 4.2. Raw_data_OSC.txt
5. SW
 - 5.1. SerialHist.ipynb
 - 5.2. DataProcess.ipynb