

Západočeská univerzita v Plzni  
Fakulta aplikovaných věd  
Katedra informatiky a výpočetní techniky

## **Bakalářská práce**

# **Měřicí jednotka průtoku vody pro bytové stanice**

# Prohlášení

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 22. června 2016

Milan Ševčík

---

## Abstract

This bachelor thesis describes the principles of ultrasound flow measuring of fluids for use in residential flow meters.

The first task was to research the principles of fluid flow measuring, emphasizing on the ultrasound methods.

The second task was to build a flow meter working on this principle using a sample of low power, yet precise, Time-to-digital converter. A 32-bit microcontroller based on the ARM architecture is interfacing the controller with the ability to confirm the functionality and precision of the measuring unit and stability of measurement.

## Key words

Flow metering, ultrasound metering, ARM microcontroller, MAX35101.

## Abstrakt

Bakalářská práce popisuje principy ultrazvukového měření průtoku tekutin pro použití v průtokoměrech pro obytné stanice.

Prvním úkolem byl průzkum principů měření průtoku kapalin s důrazem na ultrazvukové metody.

Druhým úkolem bylo sestavení prototypu průtokoměru, pracujícího za pomoci této metody za použití nízkoodběrového, specializovaného, integrovaného obvodu. Pro komunikaci s tímto obvodem je využit 32 bitový mikrokontrolér založený na architektuře ARM, pro který bude naprogramováno programové vybavení sloužící k potvrzení funkcionality a přesnosti měřicí jednotky a stability jejího měření

## Klíčová slova

Měření průtoku, ultrazvuková měření, ARM mikrokontrolér, MAX35101.

# Obsah

<b>1</b>	<b>Úvod</b>	<b>1</b>
<b>2</b>	<b>Teoretická část</b>	<b>2</b>
2.1	Metody měření průtoku . . . . .	2
2.1.1	Mechanické metody . . . . .	2
2.1.2	Tlakové metody . . . . .	2
2.1.3	Vírové metody . . . . .	3
2.1.4	Ultrazvukové metody . . . . .	3
2.2	Měřicí modul . . . . .	5
2.2.1	Mikrokontroléry Energy Micro EFM32 Gecko . . . . .	5
2.2.2	Obvod Maxim MAX35101 . . . . .	8
<b>3</b>	<b>Praktická část</b>	<b>10</b>
3.1	Vývojové prostředí . . . . .	10
3.2	Programové vybavení . . . . .	10
3.2.1	Ovladač obvodu MAX35101 . . . . .	11
3.2.2	Inicializace mikrokontrolérů a obvodu MAX35101 . . . . .	14
3.3	Testování programového vybavení . . . . .	15
3.3.1	Test komunikace s obvodem . . . . .	15
3.3.2	Test měření teploty . . . . .	16
3.3.3	Test měření doby průletu ultrazvukové vlny . . . . .	19
<b>4</b>	<b>Závěr</b>	<b>22</b>

A Záznamy z osciloskopu

24

# 1 Úvod

Bakalářská práce se zaměřuje především na problematiku ultrazvukového měření průtoku tekutin, které poskytuje vysokou citlivost a malý dopad na vlastní tok v potrubí. V práci je vysvětlen princip tohoto typu měření a popsány předpokládané vlastnosti toku.

Cílem této práce je sestavení měřicí jednotky průtoku s ohledem na odběr vody v potrubí bytových stanic. Pro sestavený průtokoměr jsou vybrány obvody s nízkým odběrem, neboť je předkládán provoz na baterie.

V závěru je měřicí jednotka otestována na stabilitu a přesnost měření. Ke kontrole měření je využita testovací soustava simulující potrubí používané v bytových domech.

## 2 Teoretická část

### 2.1 Metody měření průtoku

Měření průtoku je možné uskutečnit pomocí mnoha rozdílných metod. Vhodnost jednotlivých metod závisí na dalších parametrech jako je laminarita proudu, ztráta tlaku v tekutině kvůli měřicímu přístroji, ...

V praktické části byla zkoušena ultrazvukové metoda s uložením snímačů koaxiálně s proudem tekutiny. Důvodem je možnost využití integrovaného obvodu MAX35101, který je připraven pro tuto metodu měření a poskytuje vysokou přesnost měření doby průletu ultrazvukové vlny.

#### 2.1.1 Mechanické metody

Mechanické metody fungují na principu kvantování kapaliny na známý objem. Měřena je rychlost otáčení části měřidla ponořené do tekutiny. Rotační část bývá nejčastěji osazena lopatkovým kolem nebo turbínkou.

Výhodou těchto metod je jednoduchost konstrukce a jejich přesnost. Nevýhodou je narušení laminarity toku a mechanické namáhání měřidla.

#### 2.1.2 Tlakové metody

Metody využívající změny tlaku za zúženou částí potrubí.

Tyto metody nepotřebují žádné pohyblivé součásti, což je jejich výhodou, neboť nedochází k mechanickému namáhání. Nevýhodou jsou však nezanebatelné ztráty tlaku.

### 2.1.3 Vírové metody

Tyto metody vytvářejí v tekutině pomocí různých stacionárních a pohyblivých objektů umělé víry. Počet vírů a frekvence jejich tvorby pak závisí na rychlosti průtoku, kterou je tím možné určit.

Výhodou těchto metod je vysoká citlivost na změny, díky které je možné zaznamenat i drobné odchylky. Nevýhodou jsou ztráty tlaku v tekutině a velká nepřesnost měření pulzujícího průtoku. [4, str. 119]

### 2.1.4 Ultrazvukové metody

Tyto metody využívají k měření průtoku tekutin ultrazvukové vlny. Využívané jsou především 2 principy měření:

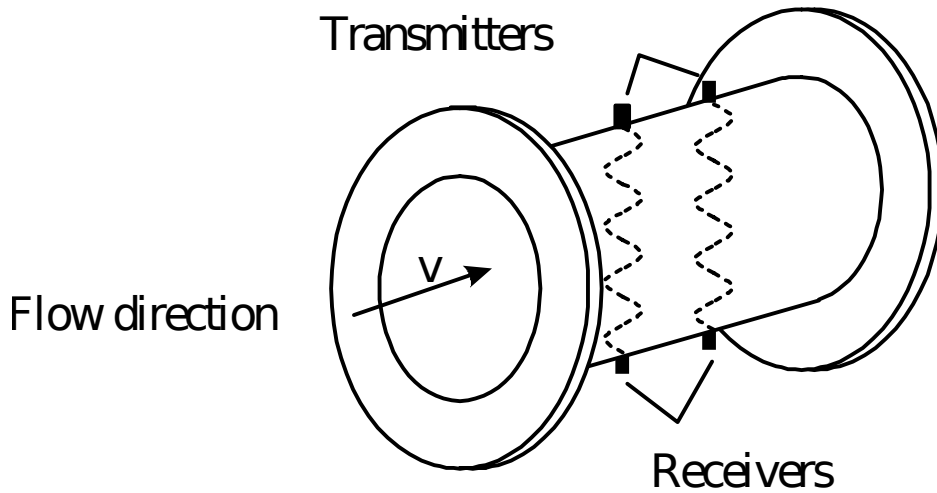
- Ultrazvuková vlna je vyslána kolmo k proudu tekutiny. Prostup pohybující se kapalinou se otiskne do tvaru vlny jako fázový posuv a různé úrovně zeslabení (snížení amplitudy). Na sejmutou vlnu jsou poté aplikované postupy podobné demodulaci fázově a amplitudově modulovaného signálu. Fázový posuv je způsoben složkou rychlosti tekutiny ve směru vyslané vlny, čímž jí uděluje zrychlení. Změny amplitudy jsou způsobeny difrakcí, odrazy a tlumením vln, kvůli změnám v hustotě.

Tyto posuvy jsou způsobené turbulentním prouděním [4, str. 116], které je vhodné tvarovat tak, aby nedocházelo k druhotným vírům, které ztěžují zpracování signálu. Tvarování se provádí vkládáním těles různých tvarů a velikostí do proudící tekutiny. Oproti klasickým vírovým metodám je však vhodné využít těleso velmi malé, což snižuje ztráty tlaku. [4, str. 99]

Na obrázku 2.1 je nákres tohoto typu průtokoměru.

- Vysílání ultrazvukových vln v ose proudění tekutiny, ideálně v obou

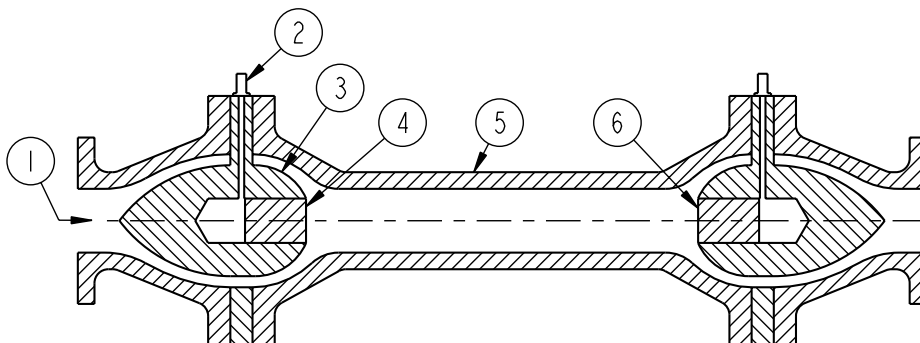




Obrázek 2.1: Uložení snímačů s vysíláním vlny kolmo k proudu tekutiny [4, str. 80]

směrech (po i proti proudu). Ultrazvuková vlna je unášena proudící tekutinou. Z rozdílné doby propagace vlny v opačných směrech je možné získat rychlost proudění kapaliny. Uložení vysílačů a přijímačů do osy proudění tekutiny je obětován pokles tlaku za zvýšenou přesnost měření průtoku. [7, str. 18]

Na obrázku 2.2 je možné vidět schéma tohoto typu průtokoměru. Číslem 1 je označen vstup tekutiny, 2 označuje přívod kabelu ke snímači, číslo 3 označuje pouzdro snímače, 4 a 6 snímače a číslem 5 je označena měřená jednotka tekutiny. Další možností je umístění snímačů



Obrázek 2.2: Uložení vysílačů koaxiálně s proudem tekutiny [7, str. 5]

vně toku kapaliny. Předpokladem je krátká rovná část potrubí, zakončená koleny, která je možné použít jako body, ve kterých jsou zafixované ultrazvukové snímače. Nevýhodou tohoto řešení může být velmi velká vzdálenost mezi snímači, která může být působit potíže při měření za použití specializovaných měřicích obvodů. Toto řešení uchycení ultrazvukových transducerů bylo použito na testovacím přípravku, který byl použit v praktické části této práce.

Obě metody vyžadují časovací obvody s vysokým rozlišením (maximálně v řádu desítek ps). Ultrazvukové vlny jsou nejčastěji používány v řádu stovek kHz až jednotek MHz [7, 4].

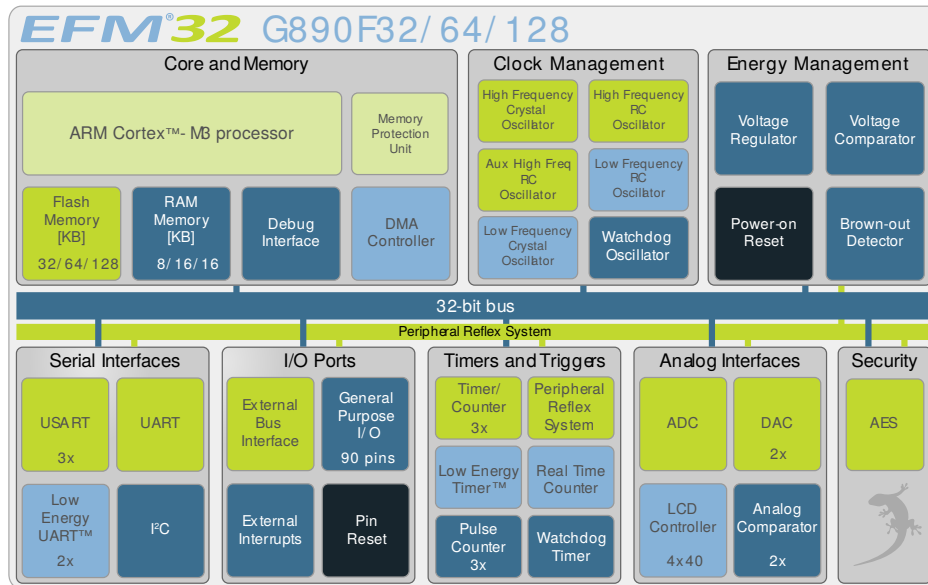
## 2.2 Měřicí modul

Měřicí modul sestává ze 2 částí:

- Vývojová deska s mikrokontrolérem a dalšími perifériemi.
- Deska zajišťující vlastní měření průtoku. Osazená časově číslicovým převodníkem a analogovými vstupy/výstupy snímačů ultrazvukových vln.

### 2.2.1 Mikrokontroléry Energy Micro EFM32 Gecko

Jde o 32 bitové mikrokontroléry založené na procesorové architektuře ARM Cortex-M3. Společnost Energy Micro tyto čipy dále vybavila schopností běhu se sníženými energetickými nároky, kdy je procesor vypnut a může být probuzen periferními zařízeními (řadiče, ...). [6] Tyto úsporné režimy byly důvodem pro výběr tohoto mikrokontroléru.



Obrázek 2.3: Blokové schéma vlastností mikrokontroléru EFM32G890 [6]

## Úsporné režimy

Mikrokontroléry EFM32 podporují 5 různých energetických módů očíslovaných od 0 do 4, kde s rostoucím číslem módu klesá spotřeba elektrické energie.

- **EM0 (Run mode):** V tomto módu provádí procesor kód uložený ve flash paměti a mohou být aktivní všechny periférie obsažené v pouzdře čipu spolu s procesorem. Potřebný elektrický proud pro funkci mikrokontroléru v tomto módu je od 180  $\mu\text{A}$  / MHz.
- **EM1 (Sleep mode):** V tomto módu je procesor uspán, všechny ostatní periférie mohou být aktivní. Potřebný elektrický proud pro funkci mikrokontroléru je 45  $\mu\text{A}$  / MHz.
- **EM2 (Deep sleep mode):** Tento mód oproti EM1 vypíná vysokofrekvenční oscilátor (32,768 kHz je stále v provozu). V tomto energetickém módu je dostupná jen menší množina periférií (např. hodiny reálného času, LCD kontrolér, řadič rozhraní I<sup>2</sup>C, ...). S běžícími ho-

dinami reálného času je potřebný proud pro funkci mikrokontroléru  $0,9 \mu\text{A}$ .

- **EM3 (Stop mode):** Oproti EM2 je vypnutý i 32,768 kHz oscilátor. Dostupné periférie jsou pouze čítač pulzů, analogový komparátor, řadič I<sup>2</sup>C a časovač Watchdogu. I tento mód je zajištěný proti ztrátě dat z paměti a registrů procesoru. Z tohoto módu je možné procesor probudit pouze jednou z dostupných periférií nebo externím přerušením. Elektrický proud potřebný pro funkci mikrokontroléru v tomto režimu je  $0,6 \mu\text{A}$ .
- **EM4 (Shutoff mode):** Potřebný proud je již pouze 20 nA. Veškerá funkcionality čipu je vypnuta s výjimkou resetu (který je také jedinou možností k přepnutí mikrokontroléru do nižších energetických módů). Všechny piny jsou nastavené do jejich resetového stavu.

Jelikož je obvod použitý k ultrazvukovému měření průtoku kapalin schopný periodického měření, které při každém průběhu vyvolá přerušení, je velmi výhodný energetický mód EM3, ze kterého je stále možné mikrokontrolér probudit externím přerušením.

## Vývojové desky

Pro vývoj byly použity vývojové desky EFM32G-DK3550 od firmy Energy Micro s mikrokontrolérem EFM32G890F128 a EM-32G880F128-H od firmy Olimex s mikrokontrolérem EFM32G880F128. Oba kity využívají téměř totožný mikrokontrolér (EFM32G880F128 má dostupný menší počet obecných vstupů/výstupů a je dostupný v jiném pouzdře oproti EFM32G890F128).

Deska EFM32G-DK3550 má osazenou širokou škálu periférií včetně microSD řadiče, segmentového LCD displeje a barevného dotykového LCD displeje. Tato deska umožňuje naprogramování mikrokontroléru i bez připojení počítače.

tače, díky Board Controlleru, který je schopný dynamicky připojovat periférie osazené na desce.

Díky bohaté výbavě vývojového kitu a obsáhlé knihovně k ní dodávané, je možné velmi jednoduše využít segmentový LCD displej pro zobrazení aktuální průtoky. Obecné vstupy/výstupy jsou vyvedeny na lišty na prototypovací desku EXP32, které se dají snadno vsadit do konektorů jiné desky. Během práce na praktické části však bylo zjištěno velké obsazení obecných vstupů/výstupů a složité přesměrování některých periférií mikokontrolérů.

Deska Olimex EM-32G880F128-H obsahuje pouze obvody pro napájení, krystaly a další pasivní elektronické součástky, které jsou potřeba pro chod mikokontroléru. Obecné vstupy/výstupy jsou vyvedené pomocí lišt na spodní straně plošného spoje.

## 2.2.2 Obvod Maxim MAX35101

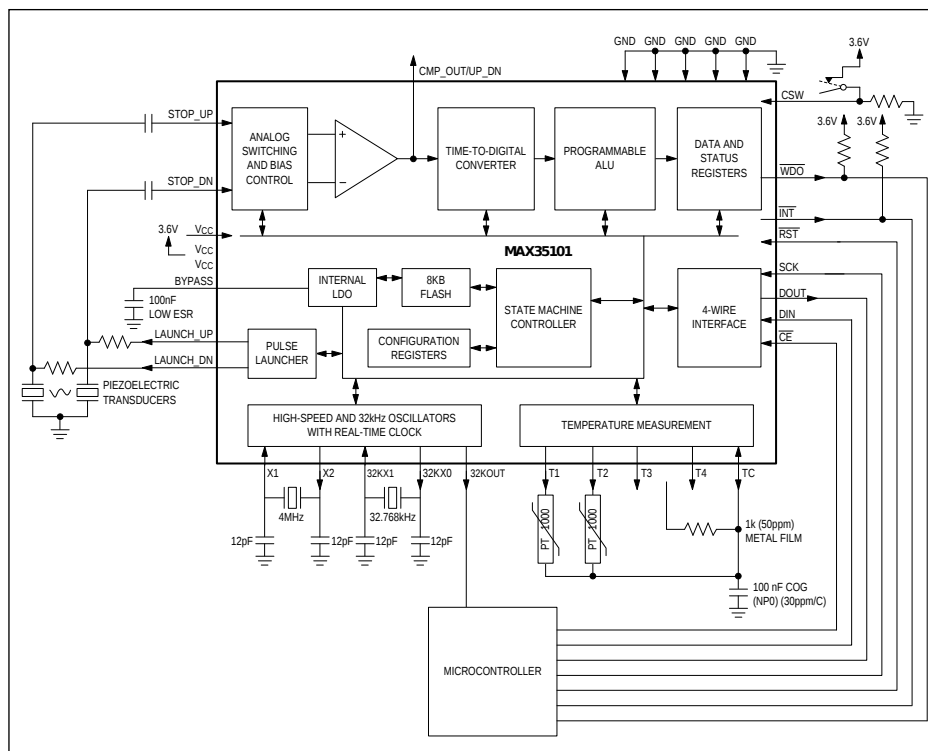
Jedná se o kompletní řešení pro ultrazvukové průtokoměry. Obvod umožňuje měření doby průletu ultrazvukové vlny ve 2 směrech a měření teploty až na 4 teplotních čidlech. Jde o časově číslicový převodník s přesností 20 ps. V pouzdře je již obsažen i zesilovač a analogový komparátor, čip je tedy možné použít jako kompletní analogovou část pro ultrazvukové průtokoměry. [2]

V obrázku 2.4 je možné zhlédnout všechny použité bloky. Také jsou vidět všechny vstupy a výstupy tohoto obvodu.

Naměřená doba průletu je vztažena ke 4 MHz oscilátoru, pro dosažení přesnosti 20 ps, která je deklarovaná v technické dokumentaci tohoto obvodu, je obvod schopný určit zlomek 1 periody tohoto oscilátoru.

Měření teploty je zajištěno použitím platinových teplotních čidel PT1000,

kdy je měření odpor na těchto čidlech vybíjením kondenzátoru v definovaném RC obvodu. Měření je poté vztaženo k referenčnímu 1 k $\Omega$  odporu.



Obrázek 2.4: Blokové schéma TDC převodníku MAX35101 a jeho typické zapojení v obvodu [2]

## 3 Praktická část

Úkolem praktické části bylo vytvořit programové vybavení, které umožní měření průtoku pomocí obvodu MAX35101. Toto vybavení pak poslouží ke zjištění přesnosti ultrazvukové metody měření průtoku.

### 3.1 Vývojové prostředí

Programové vybavení mikrokontroléru bylo vytvářeno ve vývojovém prostředí Simplicity Studio od firmy Silicon Labs. Jde o volně dostupné prostředí k mikrokontrolérům této firmy (v roce 2013 odkoupila firmu Energy Micro).

Díky podpoře vývojového kitu EFM32G-DK3550 je v tomto prostředí možné využít ladící sondy J-Link, protože deska má osazené řešení od stejné firmy. Tím je možné využít jakoukoliv sondu J-Link (v tomto případě J-Link EDU) k ladění programů na mikrokontrolérech firmy Silicon Labs i na dalších deskách.

### 3.2 Programové vybavení

Většina napsaného kódu je obsažena v souborech `max35101.c` a `max35101.h`, které představují ovladač tohoto obvodu. Rozdělení funkcí je podobné dalším ovladačům z balíčku knihoven dostupných pro mikrokontroléry Energy Micro EFM32.

### 3.2.1 Ovladač obvodu MAX35101

Ovladač obsahuje funkce pro usnadnění běžných operací s tímto obvodem.

- **MAX35101\_Init**: Inicializace obvodu včetně nastavení registrů pro měření a periférií pro komunikaci.
- **MAX35101\_Config**: Nastavení registrů pro měření.
- **MAX35101\_Reset**: V případě, že pin  $\overline{\text{RST}}$  obvodu není připojen na RESET mikrokontroléru, je možné jej touto funkcí vyvolat.
- **MAX35101\_RegisterInt**: Zaregistrování přerušení na dané množině obecných vstupů/výstupů.
- **MAX35101\_SendCMD**: Vyslání příkazu po sběrnici SPI<sup>1</sup>.
- **MAX35101\_ReadData**: Přečtení registru daného opkódem.
- **MAX35101\_WriteData**: Zapsání registru daného opkódem.
- **MAX35101\_WriteDataForced**: Zapsání registru daného opkódem, včetně kontroly zapsané hodnoty.
- **MAX35101\_TOFMeasurementResult**: Funkce sloužící k vyčtení dob průletu ultrazvukových vln v obou směrech z registrů obvodu. Vyčtené hodnoty jsou převedené do formátu čísla s plovoucí desetinnou čárkou. Dále je vyčítán také obvodem spočtená diference v časech.
- **MAX35101\_DNMeasurementResult**: Funkce sloužící k vyčtení dob po jakých byly zachycené definované vlny ve směru po proudu kapaliny. Vyčtené hodnoty jsou převedené do formátu čísla s plovoucí desetinnou čárkou. Tato funkce byla použita k otestování měření doby průletu ultrazvukové vlny.

---

<sup>1</sup>Serial Peripheral Interface



- **MAX35101\_TempMeasurementResult:** Funkce sloužící k vyčtení naměřených odporů teplotních čidel PT1000 z registrů obvodu, která vyčtené hodnoty slučuje (v případě, že je desetinná část uložena v jiném registru než část celočíselná) a převádí do formátu čísla s plovoucí desetinnou čárkou.
- **MAX35101\_GetCalibration:** Vyčtení kalibrační hodnoty 4 MHz krystalu vůči 32,768 kHz krystalu.

Pin  $\overline{CS}$  není řízen řadičem USART<sup>2</sup>, jelikož jsem s ním nedosáhl spolehlivého nastavování tohoto pinu. Řízen je tedy jako obecný výstup. Funkce obstarávající komunikaci po SPI však musí být blokující do doby, než jsou všechna data odeslána, jinak by docházelo k předčasné změně na pinu  $\overline{CS}$ .

## Komunikace s obvodem MAX35101

Obvod je k mikrokontroléru připojený běžnou sériovou sběrnicí SPI. Opkódy všech příkazů jsou 1 bytové. V případě přenosu dat z registrů nebo integrované flash paměti je potřeba zajistit chod hodin po celou dobu přenosu ze strany mikrokontroléru. Datové signály jsou obvodem vzorkovány při doběžné hraně hodin, což je potřeba neopomenout při nastavení řadiče integrovaného v mikrokontroléru.

Obvod obsahuje velké množství registrů rozdělených do 3 kategorií.

- Konfigurační registry: Umožňují nastavení keramický kondenzátor datasheetu parametry měření doby průletu ultrazvukové vlny a měření teploty. Dále je možné nastavit hodiny reálného času a případný alarm v nastavenou dobu. Všechny tyto registry je možné jak zapisovat, tak

---

<sup>2</sup>Universal Synchronous/Asynchronous Receiver/Transmitter - zařízení zajišťující sériovou komunikaci, které je možné použít pro implementaci různých sériových rozhraní, např. SPI, RS232, ...

číst. Čtecí opkódy mají vždy číslo o 0x80 vyšší. Všechny registry (s výjimkou registru TOF1), jsou ve výchozím stavu vynulované.

- Datové registry: Slouží především k získání naměřených hodnot. Dále jsou dostupné i průběžné průměry při dlouhodobých měřeních a případná odchylka 4 MHz krystalu během kalibrace. Tyto registry jsou dostupné pouze pro čtení.
- Stavové registry: Obvod obsahuje 2 stavové registry: Interrupt Status Register a Control Register. Interrupt Status Register podává informaci, z jakého důvodu je nastavený vývod přerušeni. Čtením tohoto registru je přerušeni zrušeno. Control Register obsahuje pouze 2 příznaky: příznak alarmu hodin reálného času a příznak otevřeného krytí obvodu. Tyto registry jsou dostupné pouze pro čtení.

### Přerušeni na mikrokontroléru s jádrem ARM Cortex-M3

K vytvoření obsluhy přerušeni na mikrokontrolérech s jádrem ARM Cortex-M3 je pouze potřeba vytvořit funkci se jménem daného přerušeni a postfixem Handler. Tato funkce je díky knihovně CMSIS<sup>3</sup> automaticky zaregistrována jako obsluha daného přerušeni.

Ukázka 3.1: Obsluha přerušeni GPIO\_ODD\_IRQ

```
11 void GPIO_ODD_IRQHandler(void)
12 {
13     /* clear flag for PE3 interrupt */
14     GPIO_IntClear(0x0008);
15 }
```

Jelikož je přerušeni využité pouze k probuzení z úsporného módu EM3, je v jeho obsluze pouze zrušen příznak přerušeni. Aby nastavení obecného

<sup>3</sup>Cortex Microcontroller Software Interface Standard

vstupu/výstupu na logickou 0 způsobilo přerušení, je potřeba pin nastavit jako vstup a poté jej nastavit jako přerušení funkcí `GPIO_IntConfig` dostupné v souboru knihoven dodávaných k mikrokontrolérům řady EFM32 emlib. V ukázce 3.2 je vidět nastavení pinu daného makry `MAX35101_INTPORT` a `MAX35101_INTPIN` a nastavení přerušení na tomto pinu při sestupné hraně signálu.

Ukázka 3.2: Nastavení přerušení na pinu připojeném k  $\overline{\text{INT}}$  pinu obvodu MAX35101

```
119 void MAX35101_RegisterInt() {
120     //Enable GPIO interrupt on port and pin defined in the header, on
    ↪ falling edge
121     GPIO_PinModeSet(MAX35101_INTPORT, MAX35101_INTPIN, gpioModeInput, 0);
122     GPIO_IntConfig(MAX35101_INTPORT, MAX35101_INTPIN, false, true, true);
123
124     NVIC_ClearPendingIRQ(MAX35101_GPIOIRQ);
125     NVIC_EnableIRQ(MAX35101_GPIOIRQ);
126 }
```

Obvod je možné buď po každém resetu znovu nastavovat konfiguračními registry, nebo je možné uložit konfiguraci do flash paměti. Všechna měření je možné vyvolávat libovolně řídicím mikrokontrolérem nebo je možné měření provádět periodicky a konec měření signalizovat logickou 0 na vývodu přerušení. Touto metodou je možné snížit energetickou náročnost celého zařízení, pokud je řídicí mikrokontrolér schopný využít přerušení k probuzení z některého ze svých úsporných režimů.

### 3.2.2 Inicializace mikrokontrolérů a obvodu MAX35101

Prvním krokem po zapnutí mikrokontroléru je funkce `CHIP_Init()`, která zajišťuje jednotný stav po resetu na celé rodině mikrokontrolérů a aplikuje úpravy pro některá errata pro různé revize čipů [5]. Dále je inicializován čítač

reálného času, který je použit k probuzení ze stavu EM2 po vypršení času čítače.

Běh dále pokračuje inicializací napsaného ovladače. Ke komunikaci s obvodem je potřeba rozhraní SPI, která je podmnožinou funkcí řadiče USART. Tento je tedy inicializován a jsou nastaveny módy pinů, které bude řadič obsluhovat. Dále je zaregistrováno přerušení na množině pinů, do které spadá připojený pin  $\overline{\text{INT}}$ . Ovladač také upravuje danou obsluhu přerušení. Nakonec je obvodu poslán příkaz k inicializaci a jsou nastavené parametry měření.

### 3.3 Testování programového vybavení

Pro otestování programového vybavení byl vytvořen testovací vzorek na univerzální desce plošných spojů. Pasivní elektronické součástky byly využity ze zásob laboratoře Katedry informatiky a výpočetní techniky.

#### 3.3.1 Test komunikace s obvodem

Cílem tohoto testu byla konfigurace a inicializace obvodu. Pro možnost ověření bylo vybráno zapnutí 32 kHz výstupu, který je generován z 32 kHz oscilátoru připojeného na dané piny. Na výstupu 32KOUT (pin 4) je po nastavení v registru vysílána obdélníková vlna s úrovněmi upravenými pro zpracování CMOS logickými obvody.[2, str. 51]

Během tohoto testu bylo odhaleno několik problémů. Nejznatelnějším problémem bylo opomenutí nastavení řadiče USART pro vzorkování dat při běžné hraně hodin. Kvůli tomuto opomenutí obvod neposílal žádná data zpět.

Dále bylo řešeno nastavování pinu  $\overline{\text{CS}}$ . Automatické nastavování řadičem

se projevilo nefunkční, z manuálů k mikrokontrolérům rodiny EFM32 není úplně jasné, zda je tato vlastnost dostupná na použité skupině mikrokontrolérů EFM32 Gecko, nebo zda je dostupná až na vyšších modelech ve skupině EFM32 Wonder Gecko.

### 3.3.2 Test měření teploty

Jelikož obvod provádí měření teploty pomocí teplotních čidel Pt1000, je možné test přesnosti provést pomocí odporové dekády. Měření teploty je prováděno vybíjením kondenzátoru v definovaném RC obvodu, předávána je poté doba vybíjení na určitou úroveň. Měření je možné provádět na až 4 portech, z toho jeden je potřeba využít pro referenční odpor 1 k $\Omega$ . Určení teploty změřené daným teplotním čidlem je přenecháno na programovém vybavení řídicího mikrokontroléru (z datasheetů teplotních čidel je možné získat lookup tabulku[1]).

Během tohoto testu byly opraveny chyby v operacích s registry obvodu. Po každém přenosu dat je potřeba vyprázdnit přijímací buffer, ve kterém jinak zůstávají stará data (a to i nesmyslná - např. 8 jedničkových bitů, které jsou přečteny při vysílání dat mikrokontrolérem, kdy je přijímací pin trvale nastaven). Dále byla zajištěna obvodem vyžadovaná endianita dat na rozhraní SPI. V ukázce 3.3 je vidět kód zajišťující tyto vlastnosti.

Ukázka 3.3: Funkce MAX35101\_WriteData

```
155 void MAX35101_WriteData(uint8_t cmd, uint16_t data) {
156     //Resolve endianness
157     uint16_t dataTX = ((data & 0xff) << 8) | ((data & 0xff00) >> 8);
158
159     GPIO_PinOutClear(MAX35101_GPIOPORT, MAX35101_CSPIN);
160
161     USART_Tx(MAX35101_USART, cmd);
162     USART_TxDouble(MAX35101_USART, dataTX);
163     //Wait for the end of the transmission
```

```

164     while (!(MAX35101_USART->STATUS & USART_STATUS_TXC));
165
166     MAX35101_USART->CMD |= USART_CMD_CLEARRX;
167     GPIO_PinOutSet(MAX35101_GPIOPORT, MAX35101_CSPIN);
168 }

```

Poté co byly tyto chyby opraveny, byla funkce zápisu do řídicích registrů rozšířena o kontrolu zapsaných dat. Díky tomuto ověření je již pravděpodobnost chybného nastavení velmi malá, neboť by muselo dojít k chybnému přenosu dat oběma směry. V ukázce 3.4 je vidět funkce `MAX35101_WriteDataForced`, která zápis registru opakuje, dokud není navrácená hodnota stejná jako nastavovaná.

Ukázka 3.4: Funkce `MAX35101_WriteDataForce`

```

170 void MAX35101_WriteDataForced(uint8_t cmd, uint16_t data) {
171     bool finished = false;
172
173     while (!finished) {
174         MAX35101_WriteData(cmd, data);
175         if (MAX35101_ReadData(cmd + 0x80) == data) {
176             finished = true;
177         }
178     }
179 }

```

Tabulka 3.1: Naměřené hodnoty při testu měření teploty

$R_0$ [ $\Omega$ ]	$T_R$ [ $\mu\text{s}$ ]	$T_{\text{ref}}$ [ $\mu\text{s}$ ]	$R_1$ [ $\Omega$ ]	$\Delta$ [ $\Omega$ ]
100	37,75937	376,25235	100,3565	0,3565
250	93,97473	376,21744	249,78835	-0,21165
500	188,03387	376,23314	499,7802	-0,2198
650	244,53755	376,13985	650,12403	0,12403
666	250,5672	376,21811	666,01579	0,01579
800	301,08607	376,22092	800,29063	0,29063

$R_0$ [ $\Omega$ ]	$T_R$ [ $\mu\text{s}$ ]	$T_{\text{ref}}$ [ $\mu\text{s}$ ]	$R_1$ [ $\Omega$ ]	$\Delta$ [ $\Omega$ ]
900	338,69818	376,21991	900,2665	0,2665
950	357,5912	376,15286	950,65394	0,65394
1000	376,50232	376,13664	1000,97219	0,97219
1030	387,93097	376,13953	1031,34859	1,34859
1070	402,97015	376,22313	1071,09351	1,09351
1099	413,94579	376,12445	1100,55537	1,55537
1099,9	414,13487	376,14986	1100,98373	1,08373
1100	414,14105	376,13083	1101,05586	1,05586
1100	414,14009	376,40988	1100,23703	0,23703
1100,1	414,30632	376,13176	1101,49252	1,39252
1101	414,59694	376,13797	1102,24697	1,24697
1101,1	414,60472	376,13097	1102,28819	1,18819
1105	416,12631	376,13104	1106,33334	1,33334
1110	418,11337	376,22072	1111,35127	1,35127
1150	433,12708	376,13632	1151,51622	1,51622
1190	448,23451	376,13664	1191,68	1,68
1200	452,10861	376,13168	1201,99556	1,99556
1320	497,33014	376,13022	1322,22862	2,22862
1500	565,23065	376,57019	1500,9968	0,9968
1700	640,77095	376,11987	1703,63492	3,63492
1800	678,58379	376,43471	1802,65999	2,65999
2000	754,57962	376,10175	2006,31778	6,31778
2500	943,60089	376,00743	2509,52724	9,52724
3000	1133,1273	375,60387	3016,8148	16,8148
3500	1322,11987	375,47386	3521,20349	21,20349
3850	1454,59932	375,11702	3877,72147	27,72147
4000	1511,60049	375,02547	4030,66092	30,66092
4300	1625,69955	374,60197	4339,80509	39,80509
4500	1701,584	374,27823	4546,30772	46,30772
4800	1815,23557	373,88588	4855,05248	55,05248

$R_0$ [ $\Omega$ ]	$T_R$ [ $\mu\text{s}$ ]	$T_{ref}$ [ $\mu\text{s}$ ]	$R_1$ [ $\Omega$ ]	$\Delta$ [ $\Omega$ ]
5000	1899,60687	373,12343	5091,09514	91,09514
5250	1994,59996	369,81671	5393,4825	143,4825

V tabulce 3.1 je možné vidět naměřené hodnoty během testovacího měření. Pro výpočet odporu je potřeba použít vzorec 3.1[2, str. 13], po úpravě 3.2. V této tabulce hodnoty s označením  $R_0$  představují odpor nastavený na odporové dekádě,  $T_R$  dobu vybíjení kondenzátoru za použití měřeného odporu,  $T_{ref}$  dobu vybíjení kondenzátoru za použití referenčního odporu,  $R_1$  odpor vypočtený z časů vybíjení kondenzátoru a  $\Delta$  odchylku naměřeného odporu od reálné hodnoty.

$$\frac{T_{ref}}{T_R} = \frac{R_0}{R_1} \quad (3.1)$$

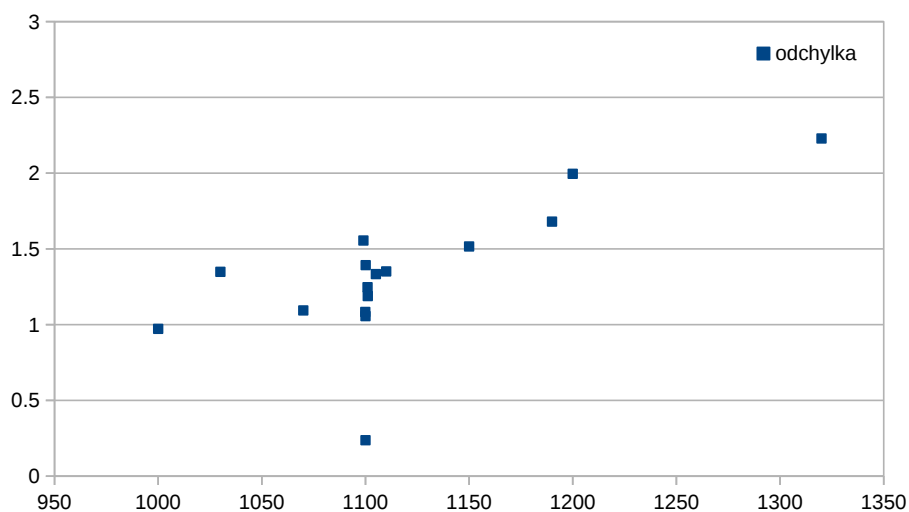
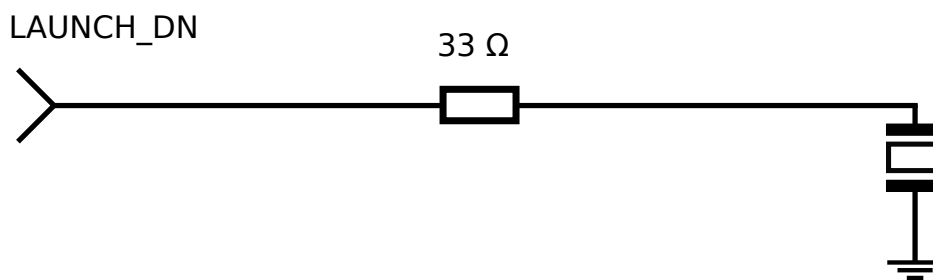
$$R_1 = \frac{R_0 \cdot T_R}{T_{ref}} \quad (3.2)$$

Velké odchylky při měření vysokých odporů nepůsobí velké problémy, neboť při použití teplotních čidel Pt1000 a předpokládaném rozsahu měřených teplot 0 - 80 °C se měřený odpor pohybuje v rozsahu 1000 - 1320  $\Omega$ [1], ve kterém je odchylka měření velmi malá, jak je možné vidět v grafu na obrázku 3.1. Odchylka může být částečně způsobená přesností použité odporové dekády.

### 3.3.3 Test měření doby průletu ultrazvukové vlny

Pro potřeby testování měření doby průletu byly vytvořeny redukce z koaxiálních kabelů na piny konektorových lišt. Přímo na kabel redukce byl připájen potřebný odpor a kondenzátor udržující bias. Schémata vytvořených kabelů jsou na obrázcích 3.2 a 3.3.

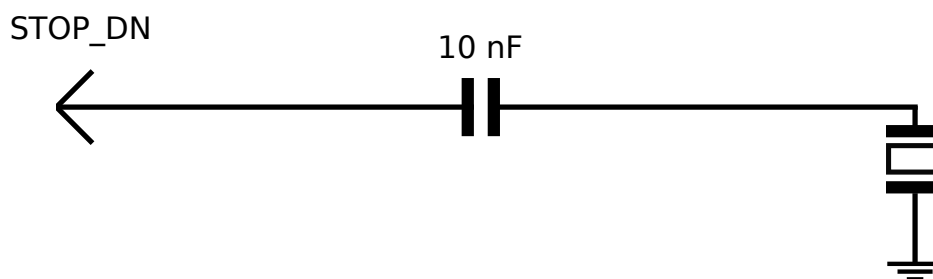


Obrázek 3.1: Odchyly naměřených odporů v rozsahu 1000 - 1320  $\Omega$ 

Obrázek 3.2: Schéma kabelu sloužícího k vyslání ultrazvukové vlny

První měření byla provedena s velikostí odporu 330  $\Omega$  a kapacitou kondenzátoru 1 nF, které byly uvedeny v aplikačních poznámkách jako vyhovující pro většinu piezoelektrických transducerů [3]. Tyto hodnoty se ukázaly jako nevhodné. Kondenzátor nebyl schopný udržet nastavený bias po dostatečně dlouhou dobu a amplituda vlny byla příliš nízká na její spolehlivé zachycení (měření bylo snadno ovlivněné šumem).

Během dalších pokusů jsem poté dospěl k použití 33  $\Omega$  odporu a kondenzátoru s kapacitou 15 nF. Odpor přispěl ke zvýšení amplitudy na 100 mV. Kapacita kondenzátoru snížila rozdíl v biasu mezi vysláním a příjmem vlny



Obrázek 3.3: Schéma kabelu sloužícího k přijmutí ultrazvukové vlny

na 18 mV. Tyto 2 změny zajistily dlouhodobě spolehlivé zachytávání ultrazvukových vln.

Ideálním stavem je stabilní hodnota biasu. Řešením může být zkrácení doby měření, tzn. zkrácení vzdálenosti mezi transducery, čímž je možné snížit úbytek biasu. Další možností je použití kvalitního kondenzátoru s vysokým koeficientem jakosti.

V tabulce 3.2 jsou naměřené hodnoty ze 6 měření. Na těchto hodnotách je vidět stabilita měření po správném nastavení měřicího obvodu.

Tabulka 3.2: Naměřené hodnoty při testu měření doby průletu ultrazvukové vlny.

Číslo měření	Naměřený čas [1/4 $\mu$ s]	Naměřený čas [ $\mu$ s]
1	1629,2841949	407,32105
2	1629,2842559	407,32106
3	1629,2845154	407,32113
4	1629,2834778	407,32087
5	1629,2839661	407,32099
6	1629,2840576	407,32101

## 4 Závěr

V rámci práce jsou shrnuty metody měření průtoku kapalin. Do detailu je popsána ultrazvuková metoda s transducery umístěnými ve směru průtoku, pro kterou byl využitý testovací přípravek sestavený na Katedře informačních technologií a výpočetní techniky na Západočeské univerzitě v Plzni. Cílem práce pak bylo ověření schopností této metody za použití specializovaného obvodu.

Řízení měřicí jednotky bylo zajištěno mikrokontrolérem Energy Micro EFM32 Gecko, který využívá licencované jádro ARM Cortex-M3. mikrokontroléry této řady umožňují velmi nízkou spotřebu elektrické energie díky implementaci několika úrovní velmi úsporných režimů.

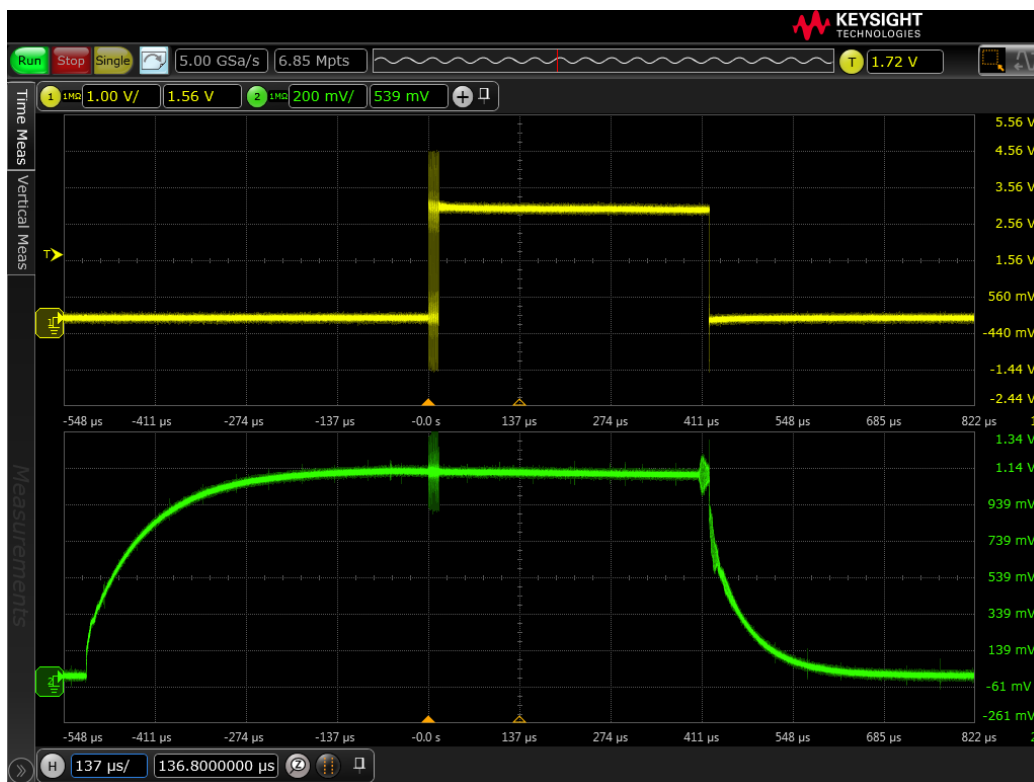
V rámci praktické části, bylo naprogramováno programové vybavení řídicího mikrokontroléru, které posloužilo pro ověření funkčnosti měřicího modulu. Ověření přesnosti probíhalo vyčítáním hodnot za pomoci ladící sondy J-Link a debuggeru zabudovaného do vývojového prostředí Simplicity Studio.

Ultrazvukové měření průtoku kapalin za použití integrovaného obvodu MAX35101 od společnosti Maxim Integrated se během testů projevilo jako stabilní. Jediným zádrhelem může být nutnost přesného nastavení měřících parametrů za pomoci osciloskopu, neboť je nutné zajistit správnou úroveň napětí, na které je zachycena první přijmutá vlna. V případě špatného nastavení je možné, že měření budou nestabilní (hlavně v případě, že je vzdálenost mezi transducery v řádu desítek centimetrů a výše). Nepřesnost může být způsobena šumem, který může překročit nastavenou zachytávací úroveň.

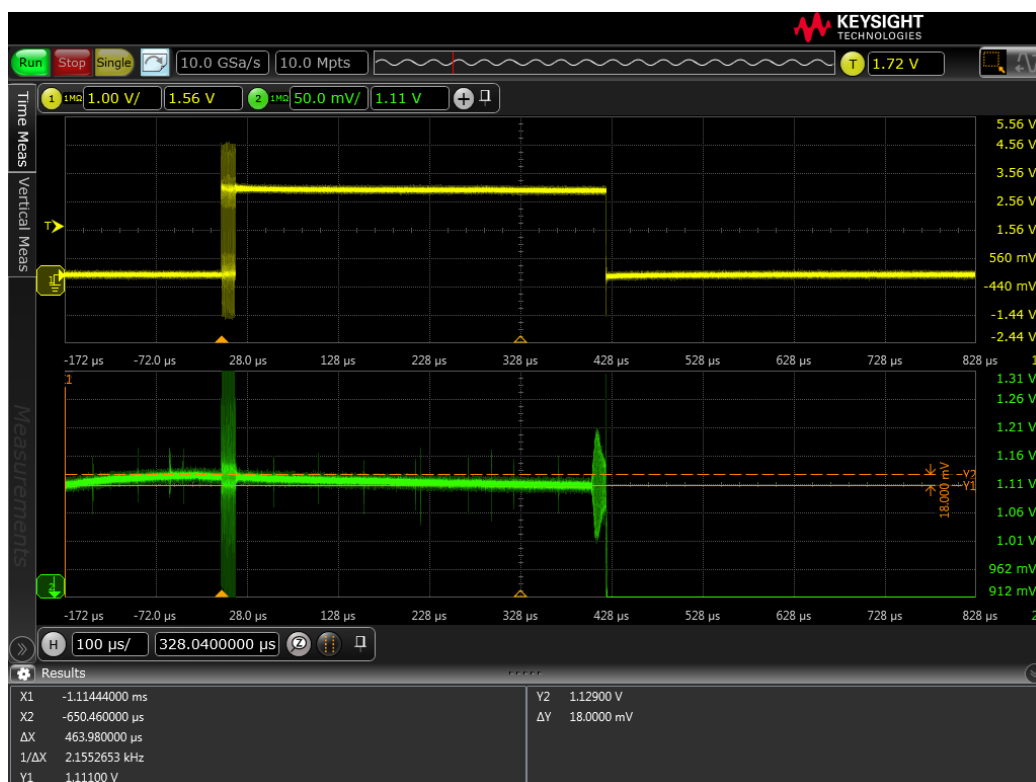
# Literatura

- [1] *Platinum resistance temperature sensors Pt100 (Pt1000)*. Kongsberg, 2004. Dostupné z: [https://www.km.kongsberg.com/ks/web/nokbg0397.nsf/AllWeb/A707D00EE0F558D6C12574E1002C2D1C/\\$file/tsiec751\\_ce.pdf](https://www.km.kongsberg.com/ks/web/nokbg0397.nsf/AllWeb/A707D00EE0F558D6C12574E1002C2D1C/$file/tsiec751_ce.pdf).
- [2] *MAX35101 Datasheet*. Maxim Integrated, 2015. Dostupné z: <https://datasheets.maximintegrated.com/en/ds/MAX35101.pdf>.
- [3] *APPLICATION NOTE 6106: MAX35101, MAX35102, MAX35103 Frequently Asked Questions*. Maxim Integrated, 2015. Dostupné z: <https://www.maximintegrated.com/en/app-notes/index.mvp/id/6106>.
- [4] MERZKIRCH, W. (Ed.). *Fluid Mechanics of Flow Metering*. Německo : Springer-Verlag Berlin Heidelberg, 2005. ISBN 3-540-22242-1.
- [5] *What does the CHIP\_Init function do?* Silicon Labs, 2012. Dostupné z: <http://blog.siliconlabs.com/t5/32-bit-MCU/What-does-the-CHIP-Init-function-do/td-p/97957>.
- [6] *EFM32G890 Datasheet*. Silicon Labs, 2015. Dostupné z: <https://www.silabs.com/Support%20Documents/TechnicalDocs/EFM32G890.pdf>.
- [7] TEMPERLEY, N. C. *Optimisation of an Ultrasonic Flow Meter Based on Experimental and Numerical Investigation of Flow and Ultrasound propagation*. PhD thesis, University of New South Wales, 2002.

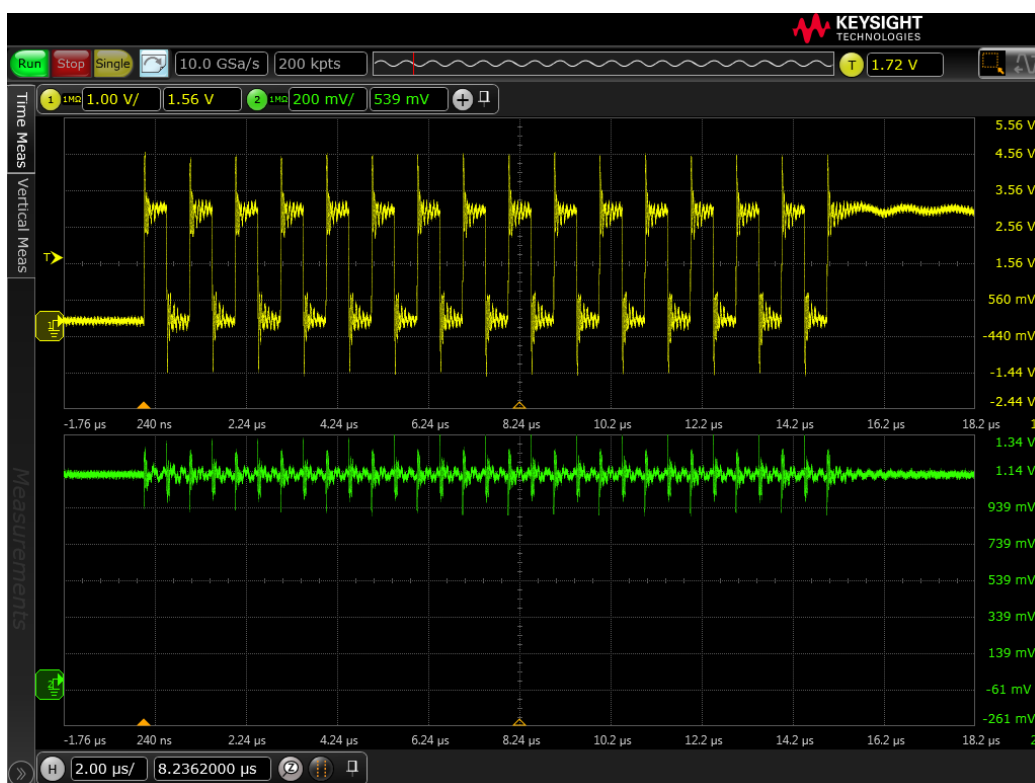
# A Záznamy z osciloskopu



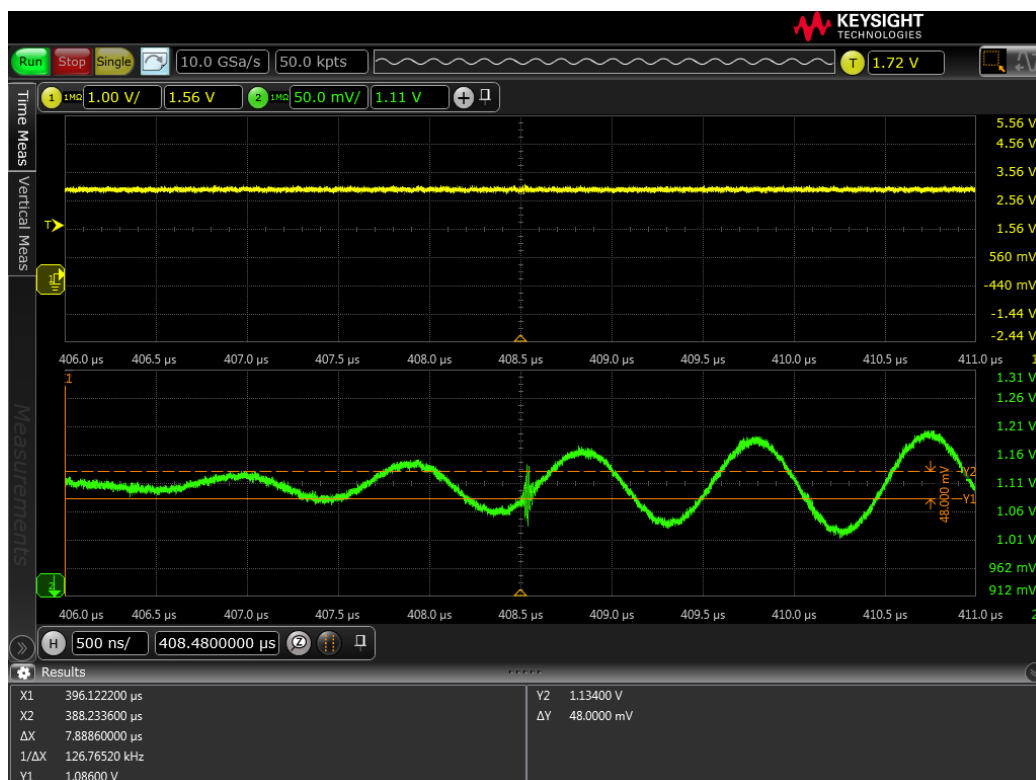
Obrázek A.1: Pohled na celý průběh měření.



Obrázek A.2: Měření poklesu napětí biasu. Kurzor Y2 je nastaven na velikost biasu počátek měření, Y1 je nastaven na bias na konci měření.



Obrázek A.3: Pohled na vysílanou vlnu.



Obrázek A.4: Pohled na přijímanou vlnu. Kurzor Y2 je nastavený na počáteční hodnotu biasu, Y1 je nastaven na hodnotu, při které obvod MAX35101 detekuje první vlnu. Tato hodnota je nastavitelná, během testů hodnota 48 mV umožňovala dosažení stabilních měření. Další vlny jsou poté již měřeny jako průniky vlny s nastaveným biasem.