



WYDZIAŁ ELEKTRONIKI,
TELEKOMUNIKACJI
I INFORMATYKI

Imię i nazwisko studenta: Arkadiusz Borowicki

Nr albumu: 179991

Poziom kształcenia: Studia pierwszego stopnia

Forma studiów: stacjonarne

Kierunek studiów: Elektronika i telekomunikacja

Profil: Komputerowe systemy elektroniczne

PROJEKT DYPLOMOWY INŻYNIERSKI

Tytuł projektu w języku polskim: Inteligentny czujnik pojemnościowy oparty na układzie opóźniającym fazę i module Arduino Micro

Tytuł projektu w języku angielskim: Smart capacitance sensor based on a phase delay circuit and a Arduino Micro module

Opiekun pracy: dr hab. inż. Zbigniew Czaja

STRESZCZENIE PRACY

W niniejszym projekcie dyplomowym opracowano i zrealizowano układ laboratoryjny, którego celem jest dokładne wyznaczenie wilgotności względnej przy zastosowaniu czujnika pojemnościowego. Metoda pomiaru odbywa się w dziedzinie czasu i opiera się na uniwersalnym interfejsie dla mikrokontrolerów z wewnętrznym układem przechwytywania zdarzenia. Układ pomiarowy bazuje na precyzyjnych komparatorach, z zapewnieniem dokładnych napięć referencyjnych.

Opracowano i przetestowano prototyp kompaktowego, pojemnościowego układu laboratoryjnego opartego na 8-bitowym mikrokontrolerze ATMega32U4. Przeanalizowano maksymalną możliwą niedokładność pośrednio mierzonej pojemności, a także przeprowadzono badania eksperymentalne. Wyniki potwierdziły, że maksymalny błąd względny wartości pojemności wyniósł mniej niż 0,1 %, co odpowiada dokładności pomiaru 0,1 pF dla zakresu mierzonych wartości pomiędzy 160 pF a 200 pF, co z kolei odpowiada co najmniej 0,3125 % rozdzielczości wilgotności względnej dla komercyjnych czujników RH tj. HS1101 czy Philips H1.

Słowa kluczowe:

czujnik, układ laboratoryjny, mikrokontroler, układ przechwytywania, uniwersalny interfejs, dziedzina czasu, pojemność, wilgotność względna, ATMega32U4, HS1101

Dziedzina nauki i techniki, zgodnie z wymaganiami OECD:

Nauki inżynierijne i techniczne: elektrotechnika, elektronika i inżynieria informatyczna

ABSTRACT

In this diploma project, a laboratory system was developed and implemented, the purpose of which is to accurately determine the relative humidity with the use of a capacitive sensor. The measurement method takes place in the time domain and is based on a universal interface for microcontrollers with an input event capture module. The measuring system is based on precise comparators, ensuring precise reference voltages.

A prototype of a compact, capacitive laboratory circuit based on the ATMega32U4 8-bit microcontroller was developed and tested. The maximum possible inaccuracy of indirectly measured capacitance was analyzed, and experimental studies were also carried out. The results confirmed that the maximum relative error of the capacitance value was less than 0,1 %, which corresponds to the measurement accuracy of 0,1 pF for the range of measured values between 160 pF and 200 pF, which in turn corresponds to at least 0,3125 % resolution of the relative humidity for commercial RH sensors, i.e. HS1101 or Philips H1.

Keywords:

sensor, laboratory system, microcontroller, input capture, universal interface, time domain measurement, capacity, relative humidity, ATMega32U4, HS1101

The field of science and technology, in line with the OECD requirements:

Engineering and technical sciences: electrical engineering, electronics and information engineering

SPIS TREŚCI

1 WSTĘP I CEL PRACY	7
2 PRZEGŁĄD METOD POMIAROWYCH	8
2.1 <i>Metoda oparta na przesuwniku fazowym.....</i>	8
2.2 <i>Metoda pobudzenia sygnałem prostokątnym</i>	9
2.3 <i>Metoda pomiaru ładowania/rozładowania RC</i>	10
2.4 <i>Wybór metody pomiarowej.....</i>	11
3 BADANIA SYMULACYJNE	13
3.1 <i>Analiza operatorowa obwodu.....</i>	13
3.2 <i>Symulacja SPICE</i>	15
3.2.1 <i>Podbloki układu</i>	15
3.2.2 <i>Wyniki</i>	18
4 BUDOWA UKŁADU LABORATORYJNEGO	22
4.1 <i>Stanowisko laboratoryjne.....</i>	22
4.2 <i>Inteligentny czujnik pojemnościowy</i>	23
4.3 <i>Układ zasilania</i>	24
4.4 <i>Moduł Arduino Micro.....</i>	25
4.5 <i>Moduł SHTC3</i>	26
4.6 <i>Moduł LCD</i>	27
4.7 <i>Układ pomiarowy</i>	28
4.8 <i>Złącza diagnostyczne</i>	31
4.9 <i>Layout PCB</i>	32
4.9.1 <i>Projekt</i>	32
4.9.2 <i>Realizacja</i>	34
5 UŻYTE UKŁADY PERYFERYJNE MIKROKONTROLERA	35
5.1 <i>System przerwań</i>	35
5.2 <i>USB – Universal Serial Bus</i>	36
5.3 <i>ICP – Input Capture</i>	37
5.4 <i>TWI – Two Wire Interface</i>	38
6 OPROGRAMOWANIE MIKROKONTROLERA	40
6.1 <i>Struktura programu</i>	40
6.2 <i>Inicjalizacja urządzenia</i>	40
6.3 <i>Pętla główna programu</i>	41
6.4 <i>Funkcje urządzenia</i>	43
6.4.1 <i>Kalibracja</i>	43
6.4.2 <i>Pomiar temperatury i wilgotności względnej</i>	44
6.4.3 <i>Pomiar pojemności</i>	45

7 INTERFEJS UŻYTKOWNIKA NA KOMPUTERZE PC	48
7.1 Strona startowa	48
7.2 Ustawienia ogólne	49
7.2.1 Eksport wyników pomiarowych do programu Matlab.....	50
7.2.2 Eksport dziennika zdarzeń	51
7.3 Kalibracja	51
7.4 Funkcje pomiarowe.....	52
7.5 Charakterystyki pomiarów	55
7.6 Dziennik zdarzeń	56
8 WERYFIKACJA EKSPERYMENTALNA	58
8.1 Stanowisko pomiarowe	58
8.2 Pomiar pojemności wzorcowych.....	60
8.3 Korekcja pomiarów	65
8.3.1 Algorytm regresji liniowej	65
8.3.2 Algorytm interpolacji Lagrange'a.....	67
9 PODSUMOWANIE	72
WYKAZ LITERATURY	73
WYKAZ RYSUNKÓW.....	74
WYKAZ TABEL	76
WYKAZ LISTINGÓW.....	77
DODATEK A – SYMULACJA LTSPICE	78
DODATEK B – SCHEMAT IDEOWY	80
DODATEK C – SCHEMAT MONTAŻOWY	82

WYKAZ WAŻNIEJSZYCH SKRÓTÓW I OZNACZEŃ

CMOS – (ang. *Complementary Metal-Oxide-Semiconductor*) technologia wykonywania tranzystorów polowych

ICES1 – (ang. *Input Capture Edge Select*) wybór zbocza wyzwalającego moduł przechwytywania zdarzeń zewnętrznych

ICIE1 – (ang. *Input Capture Interrupt Enable*) zezwolenie na obsługę przerwań od modułu przechwytywania zdarzeń zewnętrznych

ICNC1 – (ang. *Input Capture Noise Canceler*) filtr szumów modułu przechwytywania zdarzeń zewnętrznych

ICP – (ang. *Input Capture*) moduł przechwytywania zdarzeń zewnętrznych

ISP – (ang. *In System Programming*) szeregowe programowanie mikrokontrolera

LCD – (ang. *Liquid Crystal Display*) wyświetlacz ciekłokrystaliczny

MCU – (ang. *Microcontroller Unit*) mikrokontroler

PC – (ang. *Personal Computer*) komputer osobisty

PCB – (ang. *Printed Circuit Board*) płytka z połączeniami elektrycznymi i punktami lutowniczymi

PLL – (ang. *Phase-Locked Loop*) układ powielania częstotliwości taktowania mikrokontrolera

PWM – (ang. *Phase-Width Modulation*) modulacja szerokości impulsu

SMD – (ang. *Surface-Mount Device*) element elektroniczny, montowany powierzchniowo

TWEA – (ang. *TWI Enable Acknowledge*) potwierdzenie odebrania danych w interfejsie TWI

TWI – (ang. *Two-Wire Interface*) dwuprzewodowy interfejs, do wymiany informacji z urządzeniami

TWINT – (ang. *TWI Interrupt*) flaga przerwania interfejsu TWI

TWSR – (ang. *TWI Status Register*) rejestr statusowy dwu-przewodowego interfejsu szeregowego

TWSTA – (ang. *TWI Start*) bit generujący warunek startu na magistrali TWI

TWSTO – (ang. *TWI Stop*) bit generujący warunek stopu na magistrali TWI

USB – (ang. *Universal Serial Bus*) uniwersalna magistrala szeregową

XOR – (ang. *Exclusive Or*) bramka realizująca funkcję alternatywy rozłącznej

1 WSTĘP I CEL PRACY

Parametry środowiskowe, takie jak temperatura i wilgotność, są jednymi z głównych czynników wpływających na funkcjonowanie zakładów przemysłowych. Niejednokrotnie, podczas procesów wytwórczych, różnego rodzaju firm, niezbędnym jest spełnienie rygorystycznych wymagań jakościowych dla produkowanych wyrobów.

Podczas produkcji scalonych układów elektronicznych, parametry środowiskowe wpływają na reakcje zachodzące na strukturach krzemu. Monitorowanie i korygowanie czynników jest kluczowe dla zachowania znamionowych parametrów tych układów. Specjalistyczne urządzenia, wymagające niezawodnej i długotrwałej pracy, wymuszają dokładnej stabilizacji wilgotności względnej, temperatury oraz ciśnienia w środowisku wytwórczym. Podobnymi cechami charakteryzuje się sposób przechowania podzespołów, ponieważ zaniedbanie warunków środowiskowych może wpływać na korozję wyprowadzeń, czy zwiększyć prawdopodobieństwo wyładowań ESD i uszkodzenia struktury krzemowej. Systemy monitorujące parametry środowiskowe znalazły także zastosowanie przy produkcji żywności. Zapewnienie optymalnej temperatury i wilgotności pozwala np. na wysuszenie składników do wartości, nadającej się do dalszej obróbki. Natomiast, w gospodarstwach rolnych szczególną uwagę zwraca się na optymalną wilgotność gleby, wymaganą do poprawnego wzrostu roślin.

Środowisko monitorujące parametry składa się z serii czujników, dostarczających informacji o warunkach panujących w otoczeniu, w postaci elektrycznych sygnałów. Pomiar czynników w czasie rzeczywistym umożliwia reakcję podukładów systemu regulującego. Wiele zastosowań wymusza szybką odpowiedź sensora, na zmiany środowiskowe, oraz niezawodną pracę.

Niniejsza praca dyplomowa opierała się o zaprojektowanie i realizację rzeczywistego układu laboratoryjnego, umożliwiającego dokładną kalibrację sensora HS1101 i monitorowanie wilgotności względnej w czasie rzeczywistym. Wspomniany czujnik charakteryzuje się zmianami pojemności struktury wewnętrznej, pod wpływem zmian wilgotności otoczenia. Zarządzanie układem pomiarowym, zostało zrealizowane za pomocą gotowego modułu Arduino Micro, opartego na 8-bitowym mikrokontrolerze z rodziną ATmega. Uniwersalny interfejs łączący sensor z układem zarządzającym, bazuje na module przechwytywania zdarzeń zewnętrznych. W ramach pracy powstała aplikacja na komputer PC, umożliwiająca sterowanie urządzeniem pomiarowym.

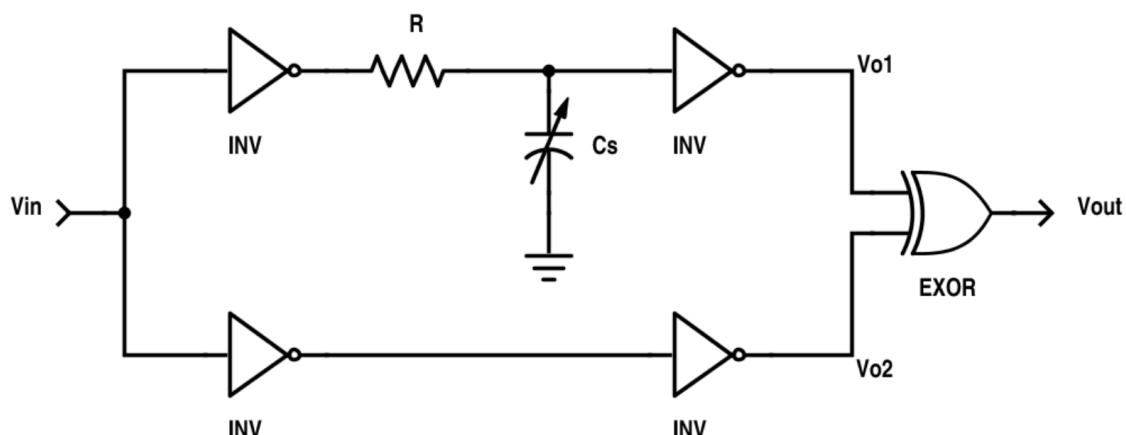
2 PRZEGŁĄD METOD POMIAROWYCH

W niniejszym rozdziale zostaną zaprezentowane metody pomiarowe, umożliwiające wyznaczenie pojemności rzędu pF. Omawiane implementacje bazują na uniwersalnym interfejsie czujnik-mikrokontroler. Szczególną uwagę zwrócono na wykorzystaną, w projekcie dyplomowym, metodę pomiarową. Zostaną omówione szczegółowe parametry każdej z implementacji.

2.1 Metoda oparta na przesuwniku fazowym

Pomiar opóźnienia fazowego członu RC, ze znaną rezystancją umożliwia określenie wartości pojemności [4]. Sygnał taktujący układ pomiarowy, dodatkowo buforowany jest przez szereg bramek inwersyjnych. Ładowanie i rozładowanie pojemności badanej C_s (rys. 2.1) wprowadza dodatkowe opóźnienie, podczas przełączenia stanów na V_{o1} . Stała częstotliwość sygnału sterującego pozwala na uzyskanie sygnału wyjściowego o charakterze PWM. Porównanie przebiegu wejściowego oraz opóźnionego odbywa się za pośrednictwem bramki XOR. Ważnym parametrem wpływającym na jakość sygnału są wprowadzone opóźnienia przez bramki inwersyjne wewnętrzne układu pomiarowego, aby możliwie wykluczyć ich wpływ zastosowano także dodatkowe układy negujące, przez które przechodzi sygnał sterujący, do bloku porównania.

Częstotliwość sygnału taktującego jest kluczowa z punktu widzenia dokładności pomiarowej. Gdyż przekroczenie opóźnienia fazowego powyżej 45° wprowadza znaczący błąd. Wygenerowany sygnał PWM umożliwia konwersję do postaci analogowej, za pośrednictwem filtra dolnoprzepustowego. W takim przypadku wynikową wielkością fizyczną, niosącą informacje o wartości pojemności, byłoby analogowe napięcie, za których pomiar odpowiadałby moduł konwertujący napięci do postaci cyfrowej (ADC – Analog to Digital Converter).

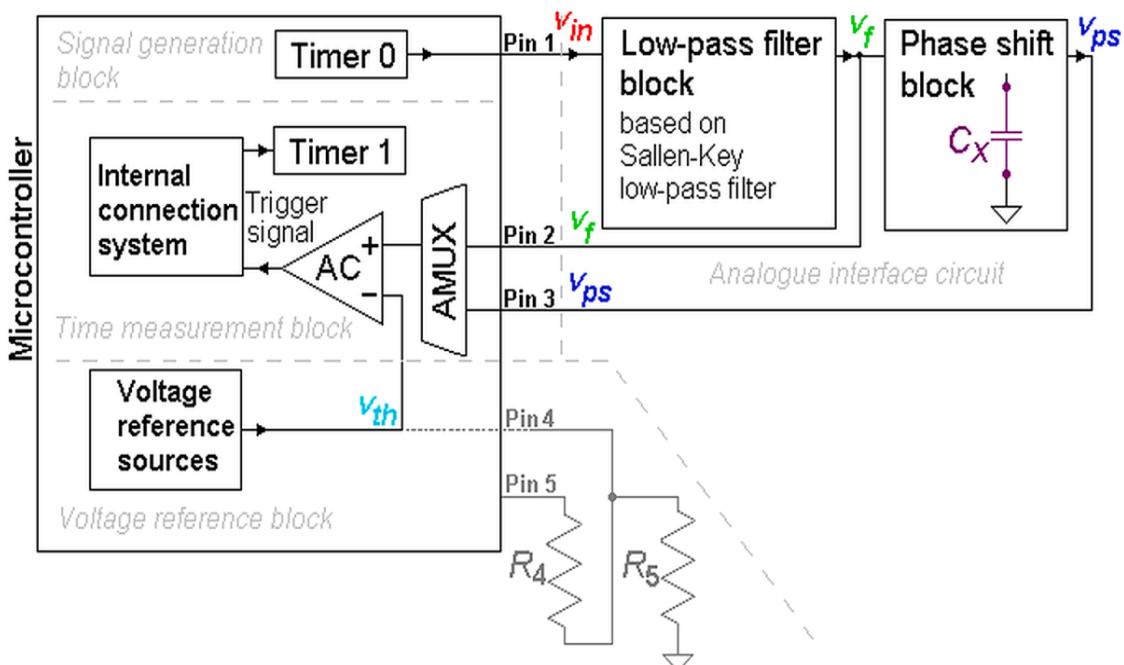


Rys. 2.1. Schemat ideowy układu pomiarowego przesuwnika fazowego [4].

2.2 Metoda pobudzenia sygnałem prostokątnym

Mikrokontroler zarządzający układem pomiarowym, ma dwa główne zadania. Generuje sygnał prostokątny o stałej częstotliwości (rys. 2.2), wyznacza czas opóźnienia wprowadzanego poprzez filtr dolnoprzepustowy oraz układ opóźniający, zawierający badaną pojemność. Głównym zadaniem filtra dolnoprzepustowego jest transformacja sygnału prostokątnego, pochodzącego bezpośrednio z mikrokontrolera, na sygnał w przybliżeniu sinusoidalny. Pobudzanie członu RC sygnałem zmiennym powoduje przesunięcie w fazie i zmniejszenie amplitudy tego sygnału, na wyjściu wspomnianego bloku, zgodnie z analizą wskazową obwodu.

Mikrokontroler wyznaczając okresy przekroczenia wewnętrznego napięcia referencyjnego (najczęściej 2,56 V) [1], względem generowanego sygnału, ma możliwość matematycznego określenia pojemności badanej znajdującej się wewnętrznie członu RC_x . Momenty progowe wykrywane są za pośrednictwem, wbudowanego w mikrokontroler, modułu komparatora. Obecne na rynku mikrokontrolery, w niektórych przypadkach nie posiadają wewnętrznego napięcia odniesienia. W takim przypadku możliwym do symulacji tego potencjału, staje się dzielnik rezystancyjny, zbudowany w oparciu o rezystory R_4 i R_5 . Zapewnienie stabilnego napięcia odniesienia, w przypadku tej metody jest elementem kluczowym, pozwalającym uzyskać akceptowalne dokładności pomiarowe. Podobną cechą charakteryzuje się częstotliwość taktowania mikrokontrolera. Zwiększenie rozdzielczości pozwala na dokładniejsze wyznaczenie różnic czasowych pomiędzy blokami układu pomiarowego.



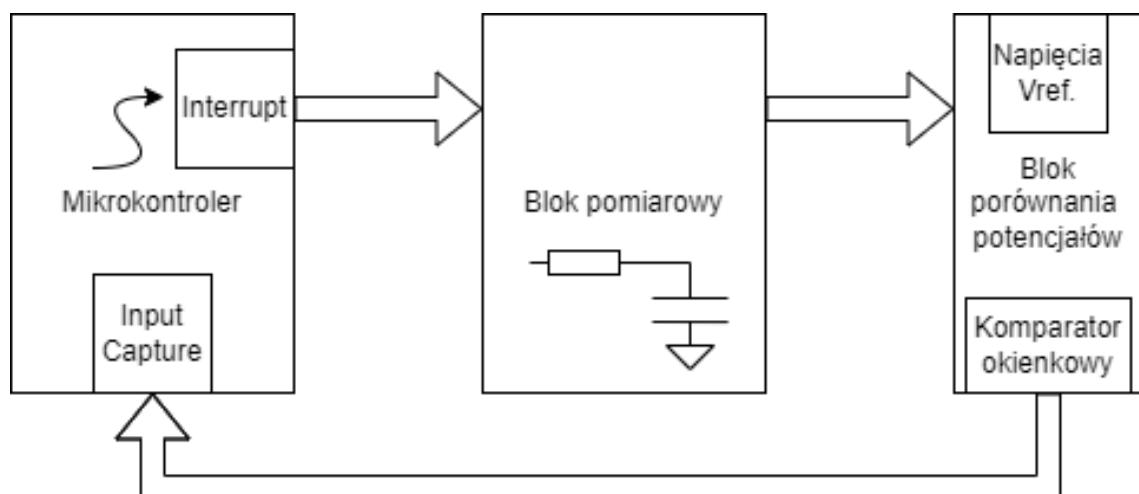
Rys. 2.2. Schemat blokowy metody pomiarowej sygnałem prostokątnym [1].

2.3 Metoda pomiaru ładowania/rozładowania RC

Czas ładowania oraz rozładowania pozwala określić, z wykorzystaniem modelu matematycznego [2], wartości pojemności badanej, przy znanej rezystancji ograniczającej. Zadaniem mikrokontrolera (rys. 2.3), jest generacja sygnału prostokątnego, o częstotliwości zależnej od czasów ładowania/rozładowania pojemności badanej, oraz pomiar okresów generacji [3]. Wymagane jest, aby blok porównywania napięć progowych udostępniał sygnały binarne, informujące mikrokontroler o przekroczeniu dozwolonego napięcia. Wystąpienie takiego zdarzenia wywołuje zmianę stanu sygnału taktującego blok pomiarowy oraz początek zliczania okresu wystąpienia kolejnego zdarzenia.

Każdorazowe przekroczenie potencjałów progowych, przez ładowaną/rozładowaną pojemność, powoduje wymuszenie obsługi przerwania, wewnątrz którego zostaje zapamiętany czas od poprzedniego wystąpienia zdarzenia. Wielokrotność pomiaru czasów pozwala na dodatkowe uśrednianie wyników. Mimo stosunkowo niewielkiej rozdzielczości zegara taktującego, mikrokontroler jest w stanie wyznaczyć bardzo dokładnie badane zależności. Działanie bloku porównania napięć referencyjnych oparte jest o komparator okienkowy, który wykrywa przekroczenie jednego z potencjałów odniesienia, przez ładunek zgromadzony wewnątrz badanej pojemności.

Moduł zliczający czas ładowania oraz rozładowania, osadzony jest wewnątrz mikrokontrolera i bezpośrednio steruje on systemem przerwań. Metoda ta posiada uniwersalny interfejs czujnik-mikrokontroler, dla MCU z wbudowanym modułem przechwytywania zdarzeń zewnętrznych.



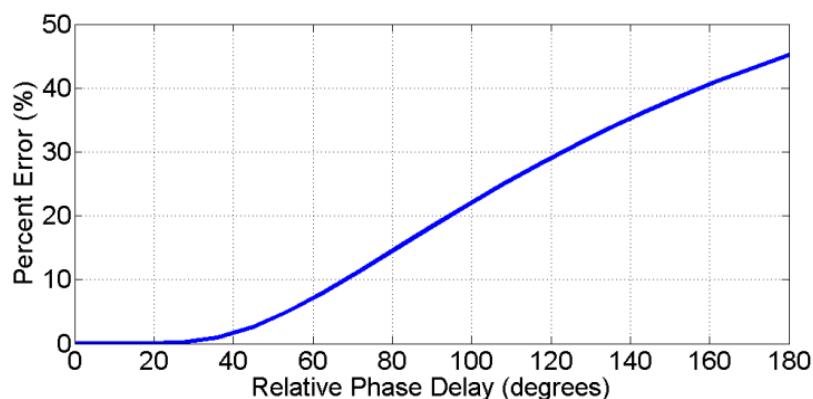
Rys. 2.3. Schemat blokowy metody pomiarowej, bazującej na ładowaniu/rozładowaniu RC.

2.4 Wybór metody pomiarowej

W celu realizacji projektu rozważono zalety i wady, każdej z wymienionych metod pomiarowych. Dokładny pomiar pojemności rzędu pF, wymusza zastosowanie systemu o niewielkich stratach, z ograniczonymi elementami pasożytniczymi i stabilnej pracy ciągłej. Charakter zastosowanej metody wymaga także rozdzielczości pomiarowej, która pozwoli 8-bitowemu mikrokontrolerowi dokładnie wyznaczyć wartość pojemności badanej.

Metoda oparta o przesuwnik fazowy została wykluczona ze względu na brak przewidzianego momentu przełączenia bramki inwersywnej, dołączonej do pojemności badanej. Noty katalogowe układów podają, że typową wartością jest połowa napięcia zasilania. Znajomość tej danej jest kluczowe dla dokładności pomiarowej. Tranzystory CMOS obwodów wejściowych, zamknięte w strukturze wewnętrznej krzemu, posiadają różne parametry ruchliwości nośników ładunku, w zależności od rodzaju kanału. Rozbieżność tych wartości zależna jest od czystości krzemu produkowanych układów, zatem parametry te mogą być odmienne w różnych partiach chipów. Czynnikiem dodatkowo wpływającym na punkt przełączenia stanu bramki jest temperatura otoczenia oraz samej struktury krzemowej, która pod wpływem wydzielanej mocy dodatkowo się ogrzewa, zmieniając tym samym napięcie przełączające.

Wadą przesuwnika fazowego jest także znaczco rosnący błąd względny, wraz ze wzrostem przesunięcia fazowego (rys 2.4). Chęć uzyskania wysokiej precyzji wymusza zastosowanie maksymalnych przesunięć sięgających granicy 20° . Pomiar opóźnienia mierzony za pomocą liczników nie zapewnia wystarczającej rozdzielczości pomiarowej, ze względu na niewielkie różnice czasowe oraz ograniczoną rozdzielczość mikrokontrolera. Czynnikiem zmniejszającym błąd jest zastosowanie tranzystorów MOS [5], których zadaniem jest całkowite rozładowanie lub naładowanie pojemności, w momencie przełączenia stanów logicznych sygnału PWM. Zabieg ten poprawia charakterystykę, jednak wnosi do układu znaczne pojemności pasożytnicze, które wymagają doświadczalnego wyznaczenia i korekcji.

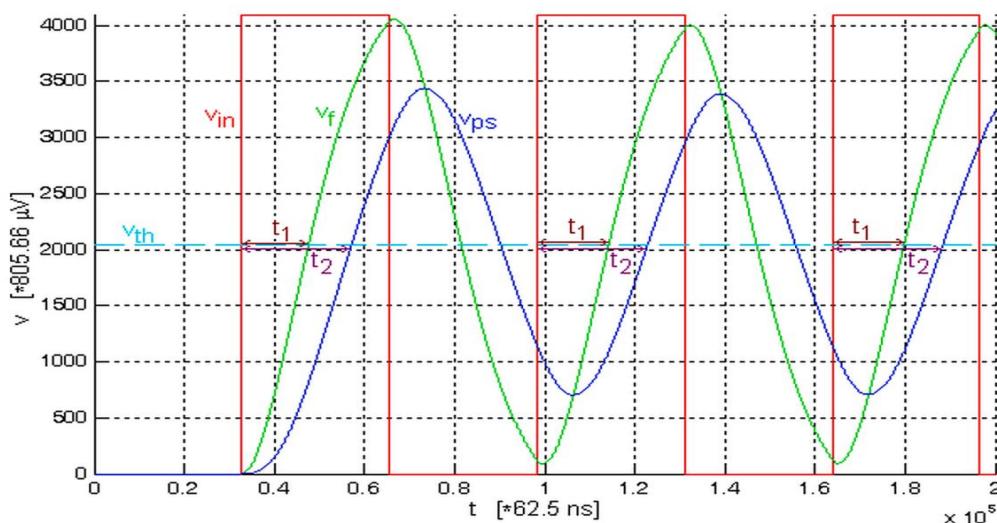


Rys 2.4. Charakterystyka błędu względnego, w zależności przesunięcia fazowego [5].

Metoda pobudzenia sygnałem prostokątnym [1], również posiada wady, wykluczające ją z realizacji fizycznej. Wybrany zakres mierzonych pojemności nie jest łatwo konfigurowalny, bowiem polega na przestrojeniu filtra konwertującego sygnał prostokątny oraz zmianie częstotliwości sygnału taktującego. Zbyt wysoka wartość częstotliwości znaczaco ogranicza możliwości pomiarowe, gdyż dokonywane są w domenie czasowej (rys. 2.5). Czynnikiem oddziałującym staje się rozdzielczość zegara taktującego mikrokontroler. Dobór odpowiedniej częstotliwości staje się kluczowym czynnikiem, wpływającym na dokładność pomiarową.

Zakładany zakres mierzonych pojemności w realizacji projektowej, określono od 100 do 300 pF. Tak szeroki zakres, przy zachowaniu akceptowalnej rozdzielczości mierzonej pojemności (0,1 pF) wymagałby zastosowania częstotliwości taktujące, na poziomie 180 Hz. Następne kolekcjonowanie serii 128 wyników, implikowałoby czas odpowiedzi układu pomiarowego około 1 s. Dodatkowym czynnikiem wpływającym na dokładność są straty wprowadzane przez filtr dolnoprzepustowy, każdorazowa konfiguracja filtra wymusza wyznaczenie współczynników tłumienia dla zadanej częstotliwości.

Metoda pomiarowa oparta o pomiar czasu ładowania i rozładowania członu RC zapewnia największą dokładność pomiarową spośród wybranych [2]. Dodatkową zaletą jest prostota konfiguracji napięć referencyjnych oraz rezystancji pomiarowych. Czasy ładowania i rozładowania zależą od rezystancji członu, zatem zakres pomiarowy zostaje ustalony za pomocą jednego, łatwo konfigurowalnego elementu. Straty wynikające z funkcjonowania układu pomiarowego zależą wyłącznie od prądów polaryzujących obwody wejściowe zastosowanych komparatorów. Metoda wprowadza także elementy pasożytnicze, nisko zależne od warunków otoczenia. Zapewnienie stabilnego napięcia zasilającego oraz referencyjnych gwarantuje wysoką precyzję pomiaru. Metoda ładowania/rozładowania RC spełnia wszystkie założone cele projektowe, co zaważyło na jej realizacji w układzie rzeczywistym.



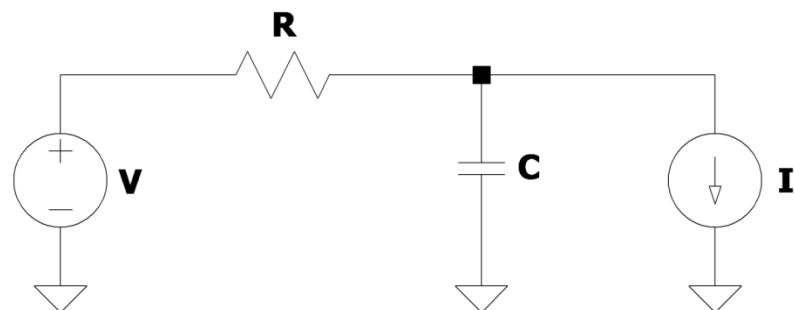
Rys. 2.5. Sygnały panujące w metodzie pobudzenia sygnałem prostokątnym [1].

3 BADANIA SYMULACYJNE

W rozdziale zostanie wyrowadzony model matematyczny obwodu, z wykorzystaniem analizy operatorowej, a także zostaną przedstawione teoretyczne charakterystyki przy użyciu programu Matlab. Przeprowadzono również szczegółową symulację obwodu w programie typu SPICE, obrazując wystąpienie pasożytniczych elementów.

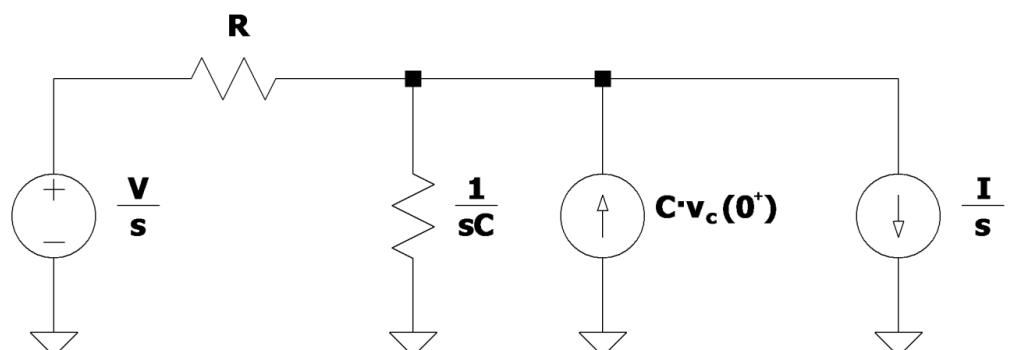
3.1 Analiza operatorowa obwodu

Obwód pomiarowy można przedstawić przy pomocy idealnych źródeł prądowych i napięciowych (rys. 3.1). Źródło prądowe (I) symbolizuje prąd wejściowy komparatorów (prąd polaryzujący tranzystory wejściowe układu scalonego). Zgodnie z dokumentacją układu TLV3502AID, prąd ten może osiągać maksymalne wartości $\pm 20 \text{ pA}$ [16]. Źródło napięciowe (V) odpowiada natomiast napięciu na wyjściu układu buforującego.



Rys. 3.1. Obwód pomiarowy.

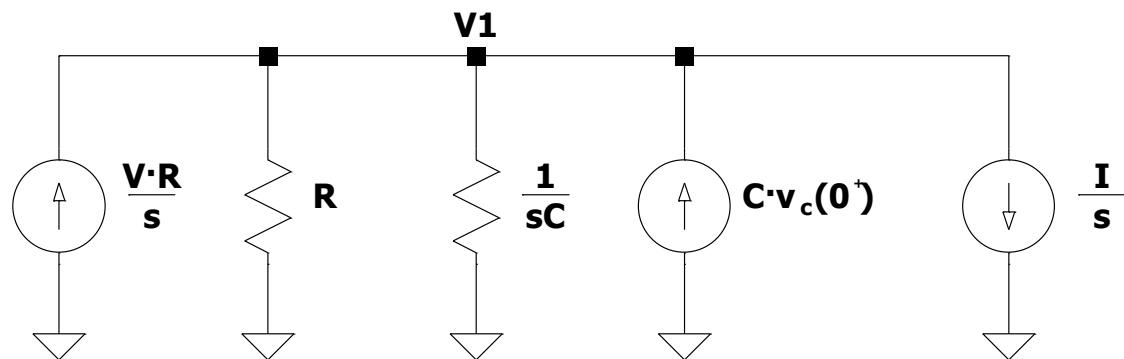
Przyjmując, że stan napięcia na wyjściu mikrokontrolera zmienił się w sposób skokowy, w chwili $t = 0$, w układzie pomiarowym będzie obowiązywać stan nieustalony. Do opisu matematycznego takich stanów służy analiza operatorowa [6] (rys. 3.2).



Rys. 3.2. Analiza operatorowa obwodu pomiarowego.

Pojemność C zastąpiono równoległym połączeniem rezystancji o wartości $\frac{1}{s \cdot C}$ oraz źródła prądowego o wartości $C \cdot v_c(0^+)$, gdzie $v_c(0^+)$ oznacza wartość chwilową napięcia na pojemności w chwili czasu $t \rightarrow 0^+$. Rezystancja R ma charakter liniowy, dlatego nie podlega przekształceniom operatorowym.

Do rozwiązania układu wykorzystano metodę potencjałów węzłowych, w tym celu należało dokonać kolejnego przekształcenia (rys. 3.3), w którym zamieniono rzeczywiste źródło napięciowe utworzone z elementów rezystancji R oraz źródła $\frac{V}{s}$, na rzeczywiste źródło prądowe.



Rys. 3.3. Przekształcony obwód pomiarowy w dziedzinie operatorowej.

Dysponując modelem po przekształceniach możliwe jest wyprowadzenie wzoru na napięcie w węźle $V1$, czyli na badanej pojemności C (1.1).

$$V1 = \frac{V \cdot \frac{R}{s} + C \cdot v_c(0^+) - \frac{I}{s}}{\frac{1}{R} + s \cdot C} \quad (3.1)$$

Po przekształceniach wzoru 3.1 oraz dokonaniu odwrotnej transformaty Laplace'a otrzymano wzór (3.2) opisujący napięcie w dowolnej chwili czasu stanu nieustalonego.

$$V1 = V + \exp\left(-\frac{t}{R \cdot C}\right) \cdot \left(v_c(0^+) - V + \frac{I}{R}\right) - I \cdot R \quad (3.2)$$

Podobnie przekształcając wzór uzyskano zależność czasu ładowania oraz rozładowania pojemności od wartości tej pojemności (3.3). Wybór obliczeń ładowania bądź rozładowania zależy od wartości parametrów użytych we wzorze, ponieważ wyznaczanie czasu ładowania pojemności będzie obejmowało inne warunki początkowe oraz inne wartości napięcia wyjściowego z bufora trójstanowego (V).

$$t = -C \cdot R \cdot \log\left(\frac{V_1 - V + I \cdot R}{v_c(0^+) - V + I \cdot R}\right) \quad (3.3)$$

W kolejnym przekształceniu wzoru otrzymano wartość pojemności, w zależności od znanego czasu ładowania lub rozładowania (3.4). Parametry V, I oraz R przyjmują stałą wartość na czas trwania stanu nieustalonego. Wyznaczanie pojemności może przebiegać zarówno z pomierzonego czasu ładowania jak i rozładowania, dla obu przypadków parametry V_1, V oraz $v_c(0^+)$ będą posiadały różną wartość.

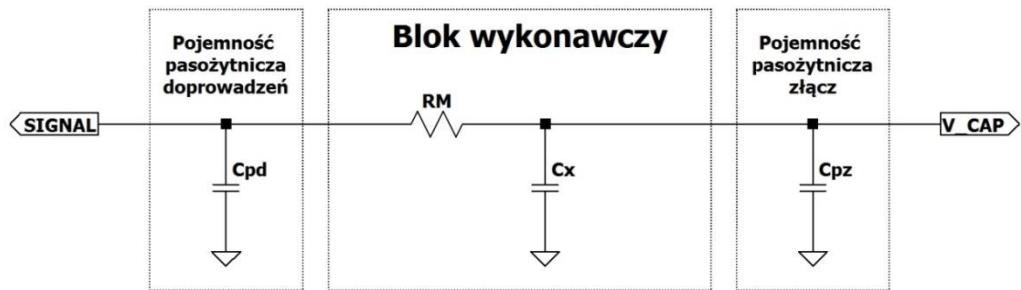
$$C = \frac{-t}{R \cdot \log\left(\frac{V_1 - V + I \cdot R}{v_c(0^+) - V + I \cdot R}\right)} \quad (3.4)$$

3.2 Symulacja SPICE

Opis matematyczny udowodnił, że pomiar wartości pojemności, w idealnych warunkach jest możliwy poprzez zmierzenie czasu ładowania lub rozładowania pojemności. W podrozdziale poddano analizie obwód pomiarowy z uwzględnieniem pojemności pasożytniczych ścieżek PCB, oraz tranzystorów obwodów wejściowych wykorzystanych układów scalonych.

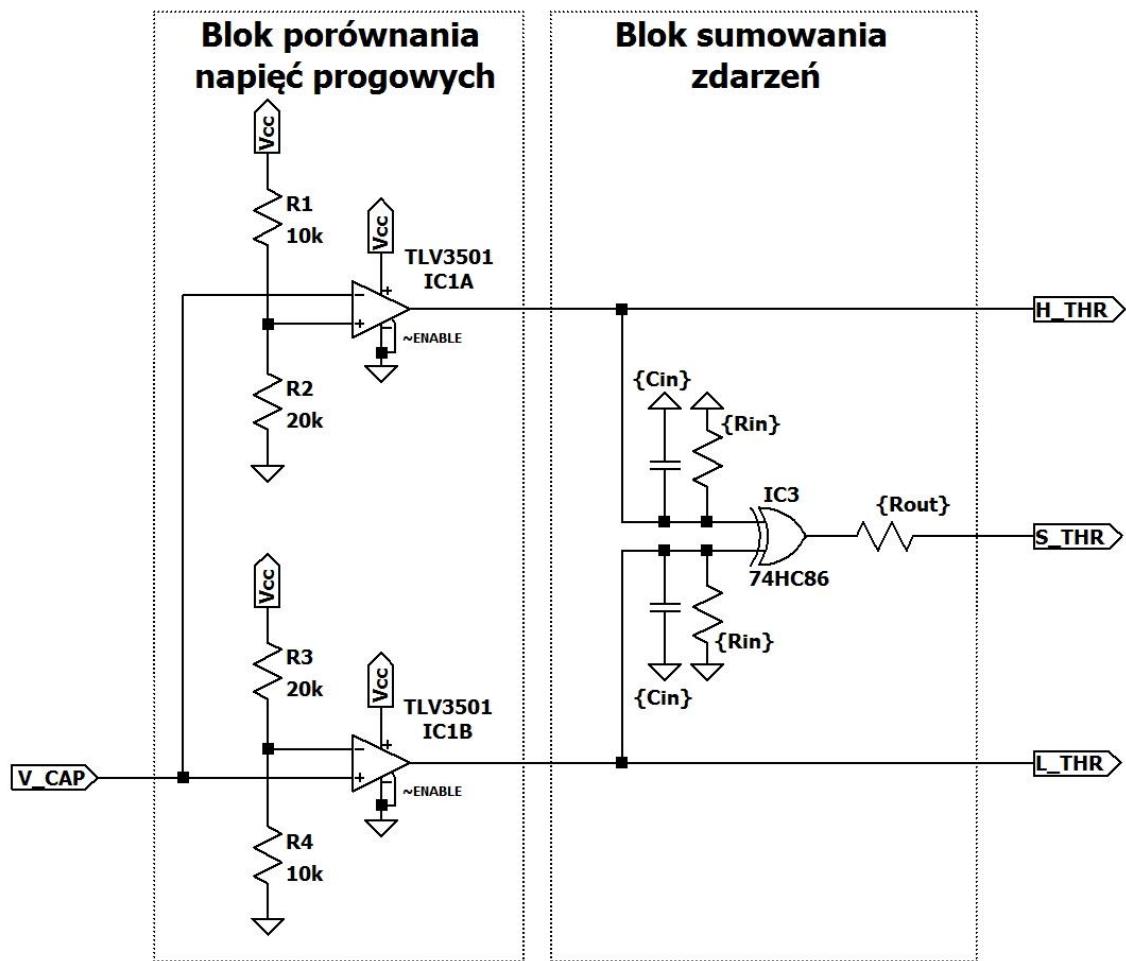
3.2.1 Poblokki układu

Blok pomiarowy (rys. 3.4) symuluje pojemności pasożytnicze związane ze ścieżkami doprowadzającymi oraz pojemnością złącz pomiarowych i diagnostycznych. Do wyprowadzenia SIGNAL dołączono jedno z wyjść mikrokontrolera poprzez trójstanowy bufor. Badana pojemność C_x oraz rezystor pomiarowy R_M tworzą człon RC. Rezystor ogranicza prąd ładowania i rozładowania pojemności przez sygnał taktujący. Ważnym parametrem jest wartość rezystancji, ponieważ zbyt wysoka spowoduje, że prądy polaryzujące komparatorów nasycą układ pomiarowy. W takim wypadku ograniczony prąd popłynie do obwodów wejściowych, nie ładując pojemności badanej.



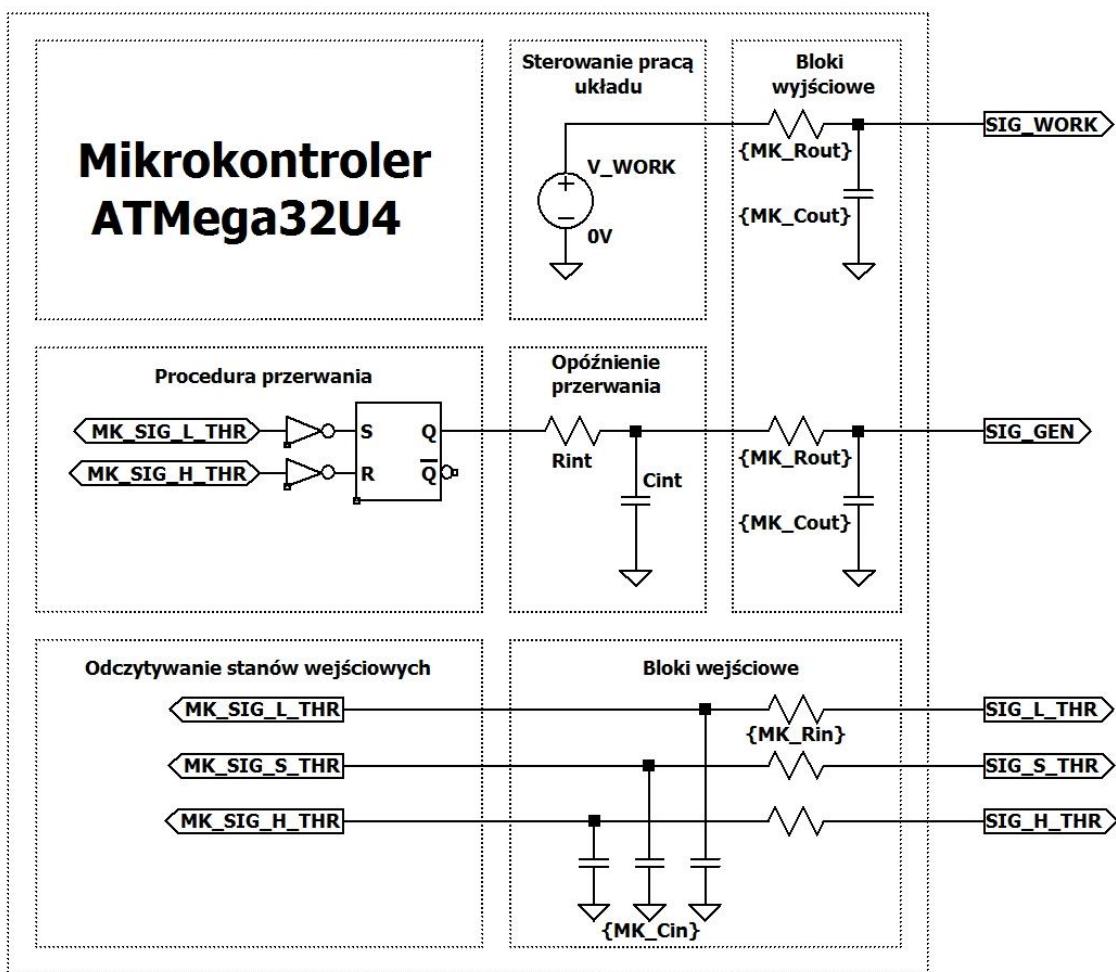
Rys. 3.4. Schemat bloku pomiarowego – symulacja.

Blok przetwarzający napięcie występujące na badanej pojemności (rys 3.5) składa się z komparatora okienkowego, które porównują poziom napięcia do zadanych progów. Jeżeli napięcie uzyska poziom większy od górnego napięcia odniesienia, lub mniejszy od dolnego, będzie to skutkowało ustaleniem stanu wysokiego na wyjściu bramki XOR. Zbocze narastające tego sygnału spowoduje aktywowanie przerwania i zmianę stanu sygnału taktującego przez mikrokontroler.



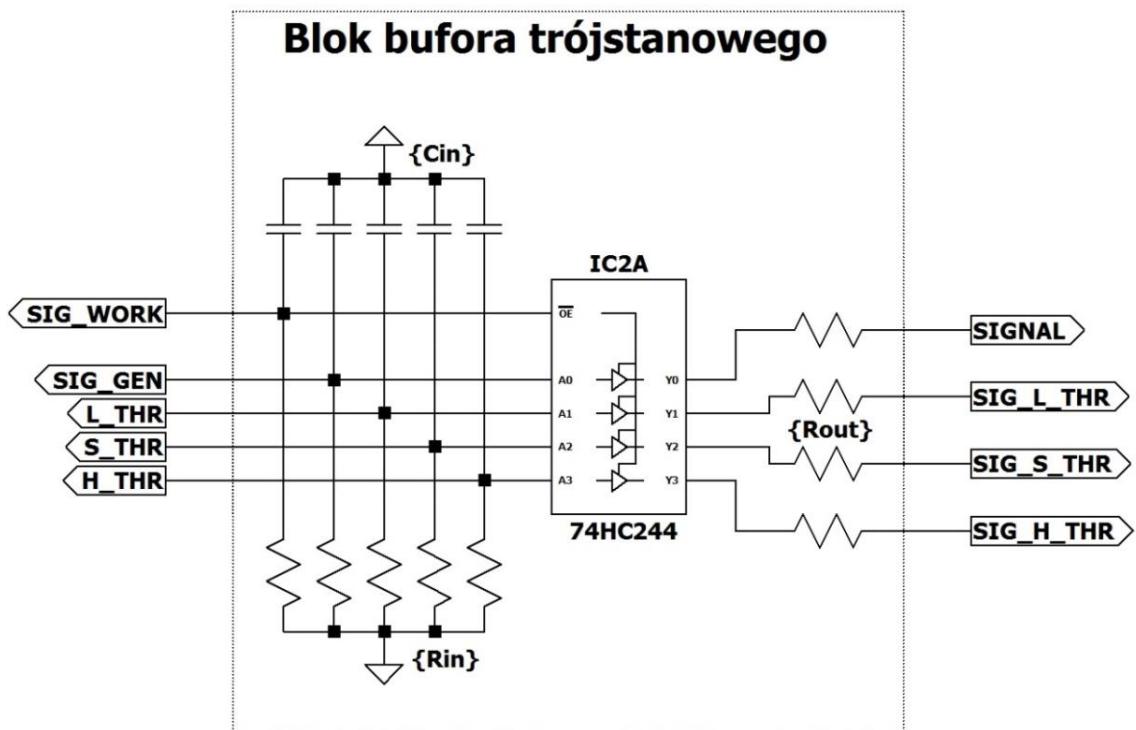
Rys 3.5. Schemat bloku porównywania napięć progowych – symulacja.

Blok symulujący działanie mikrokontrolera (rys 3.6) przewiduje inne wartości pojemności i rezystancji pasożytniczej wejść oraz wyjść układu scalonego, związane to jest z poziomem integracji struktury krzemowej. Na potrzeby symulacji zaimplementowano uproszczoną obsługę przerwania składającą się z przerzułnika RS. Z wywołaniem przerwania ściśle związane jest również opóźnienie jakie wnosi wykonanie kolejnych instrukcji wewnętrz jego obsługi, z tego względu zastosowano człon opóźniający, symulujący czas wykonania przerwania.



Rys. 3.6. Schemat podłączenia mikrokontrolera – symulacja.

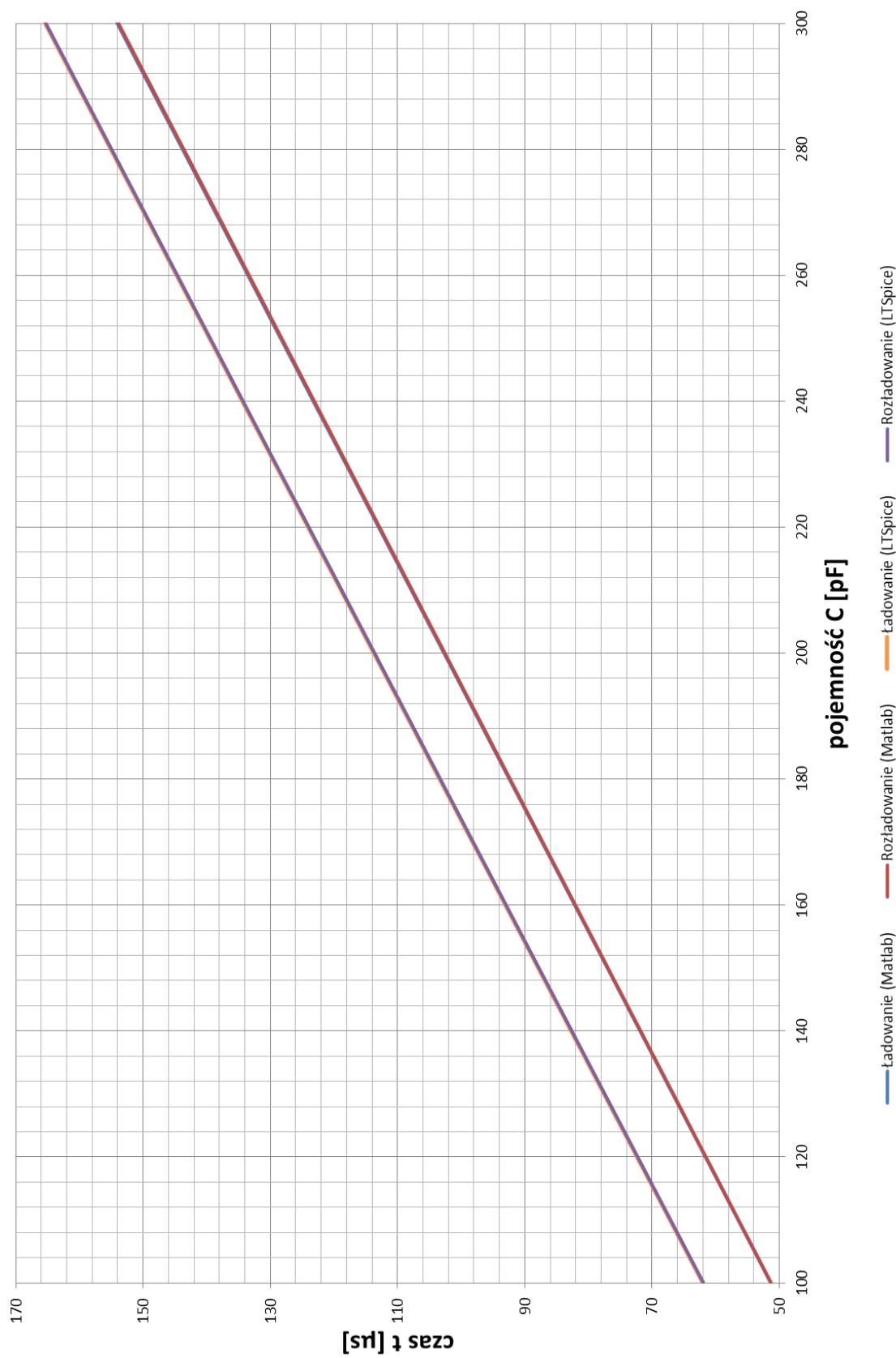
Blok bramki trójstanowej (rys. 3.7) obejmuje pojemności i rezystancje pasożytnicze wejść układu. Wykorzystane do projektu bramki i bufory są wykonane w technologii CMOS, która cechuje się pojemnością pasożytniczą bramki tranzystora oraz niewielkim prądem upływu. Te parametry zostały zaimplementowane na poczet symulacji w postaci pojemności C_{in} oraz rezystancji R_{in} . Podobna zasada występuje na wyjściach bramek, złączony jeden z tranzystorów *pull up* lub *pull down*, posiada niezerową rezystancję dren – źródło, co zostało również wzięte pod uwagę w postaci rezystancji R_{out} na wyjściach. Do projektu wykorzystano bufor, ponieważ parametry jego wyjść są bardziej optymalne w porównaniu do wyjść mikrokontrolera (mniejsze pojemności i rezystancje pasożytnicze). Parametryzowanie obwodu pozwala na szybkie dostosowanie symulacji po realizacji fizycznego układu. Prawdą jest, że wejścia i wyjścia układów scalonych mogą posiadać różne wartości pojemności pasożytniczej, nawet w obrębie pojedynczego układu. Wynika to ze struktury krzemowej oraz nieczystości krzemu wykorzystanego do produkcji. Dlatego też symulacja nie odzwierciedli zachowania rzeczywistego układu, w którym panuje znacznie więcej zjawisk.



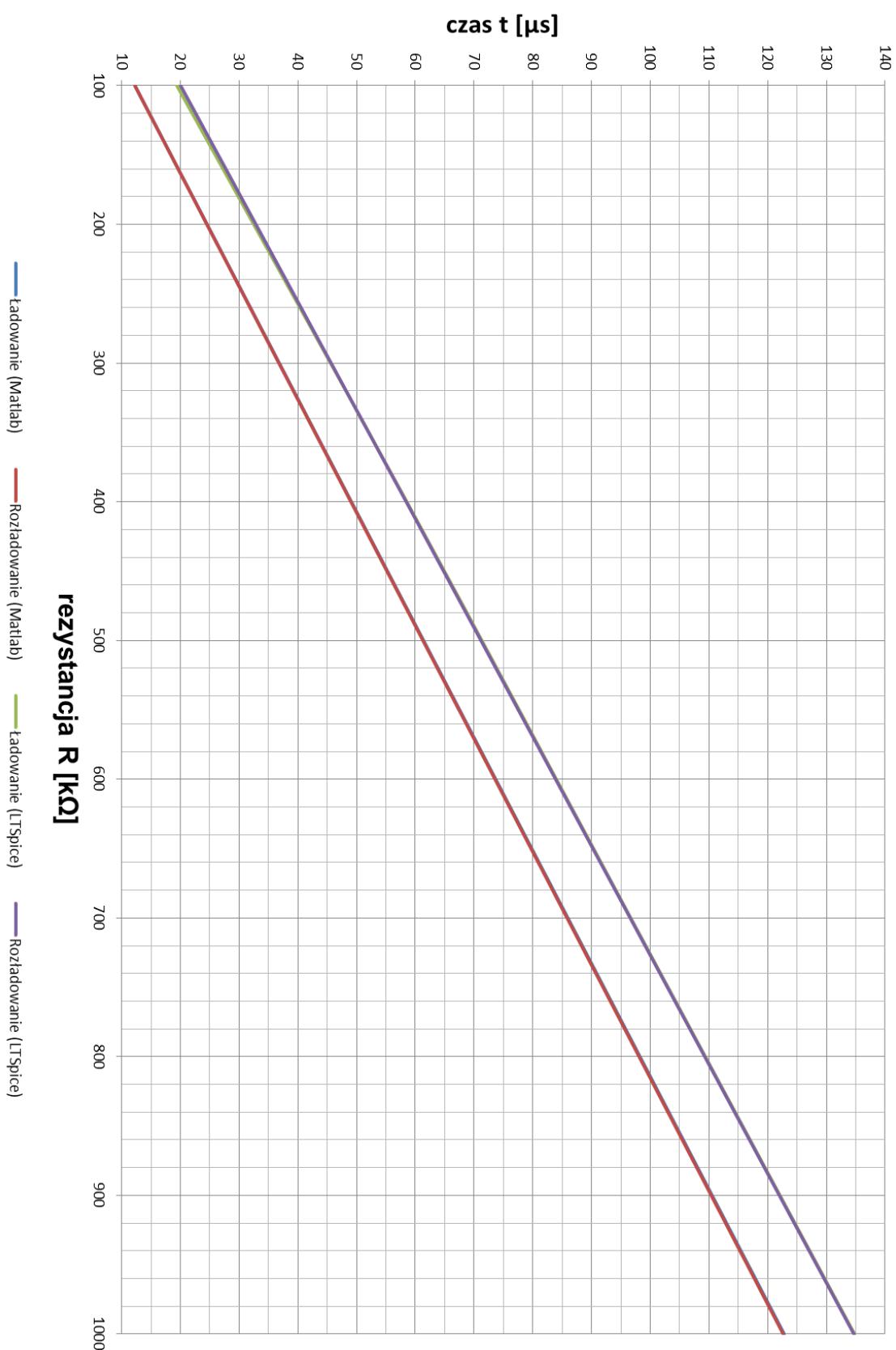
Rys 3.7. Schemat bufora trójstanowego – symulacja.

3.2.2 Wyniki

Symulacje umożliwiają pomiar czasu ładowania i rozładowania. Niewielki prąd polaryzujący komparatorów powoduje, że czasy te są zbliżone, ale nie są sobie równe. Podczas realizacji fizycznej, zachodzi konieczność doświadczalnego wyznaczenia prądu polaryzacji, aby umożliwić dokładny pomiar. Charakterystyka czasu ładowania i rozładowania wyznaczone drogą obliczeń oraz symulacji ma charakter w przybliżeniu liniowy (rys. 3.8). Charakterystyka pozwala na zobrazowanie wpływu pojemności i rezystancji pasożytniczych, które nie zostały wzięte pod uwagę w modelu matematycznym. Występowanie elementów pasożytniczych w układzie pomiarowym wydłuża czas ładowania i rozładowania pojemności, wszystko to spowodowane jest poprzez zwiększoną sumaryczną pojemność elementu badanego oraz elementów pasożytniczych. Dodatkowym czynnikiem wpływającym na czas pomiaru jest obsługa przerwania. Opóźnienie jakie zostaje wprowadzone szacuje się na około 4 μ s. Na ten czas składa się wywołanie przerwania, które wynosi 5 cykli zegara oraz wykonanie instrukcji wewnętrz przerwania. Wszystkie z powyżej wspomnianych elementów powoduje przesunięcie charakterystyki, w stronę wyższych czasów ładowania i rozładowania, o około 10 μ s w całym mierzonym zakresie pojemności. W celu kompensacji elementów pasożytniczych, w obliczeniach została zawarta funkcja korygująca zmierzona pojemność. Współczynniki funkcji zostały wyznaczone doświadczalnie, na podstawie pomiarów znanych, dokładnych wartości pojemności.



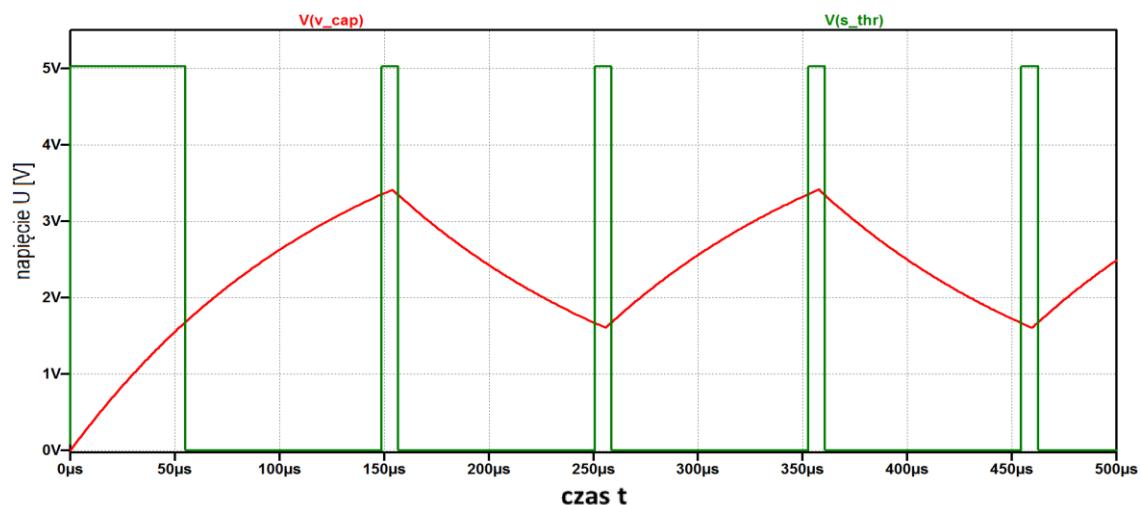
Rys. 3.8. Charakterystyka czasu ładowania i rozładowania dla rezystancji $R = 741,2 \text{ k}\Omega$.



Rys 3.9. Charakterystyka czasu ładowania i rozładowania dla pojemności $C = 177,2 \text{ pF}$.

Wyznaczona charakterystyka czasu ładowania i rozładowania pojemności badanej $C = 177,2 \text{ pF}$ przy zmianie rezystancji rezystora pomiarowego w zakresie od $100 \text{ k}\Omega$ do $1 \text{ M}\Omega$ (rys. 3.9) ukazuje w przybliżeniu liniową zależność [3]. Przesunięcie charakterystyki wywołane wystąpieniem elementów pasożytniczych w układzie pomiarowym wprowadza nierównomierne przesunięcie charakterystyki układu idealnego. Oznacza to, że dla każdej wybranej rezystancji będą wymagane inne współczynniki korekcji charakterystyki. Dodatkową cechą jaką zaobserwowano podczas symulacji jest ugięcie charakterystyki ładowania w zakresie niskich wartości rezystancji tj. mniejszych od $200 \text{ k}\Omega$. Różnice czasowe ściśle są powiązane z opóźnieniem wprowadzanym przez przerwanie. Czym mniejsza rezystancja tym w krótszym czasie badany element zgromadzi większy ładunek, powodując że po przełączeniu sygnału taktującego większy ładunek musi zostać oddany. Dlatego zaleca się wybór rezystancji większych bądź równych od $200 \text{ k}\Omega$. Czym rezystancja jest większa, tym wynik pomiarowy jest dokładniejszy. Jednak zbyt duża wartość rezystancji prowadzi do zjawiska, w którym ograniczony tą rezystancją prąd zdoła jedynie nasycić prądy polaryzujące obwodów wejściowych komparatorów. W takiej sytuacji badana pojemność nie zgromadzi ładunku w czasie pomiaru. Wartość maksymalnej rezystancji, zapewniającej poprawność pomiaru została określona symulacyjnie na około $3 \text{ M}\Omega$, natomiast w układzie rzeczywistym zachodzą również inne zjawiska, nie wzięte pod uwagę podczas symulacji.

Przykładowa charakterystyka obrazująca metodę pomiaru z uwzględnieniem elementów pasożytniczych (rys. 3.10) pomaga użytkownikowi zrozumieć podstawę pomiaru. W układzie rzeczywistym udostępniono możliwość podejrzenia przebiegów przy pomocy oscyloskopu. Pomiar sondą oscyloskopową nie odda jednak rzeczywistych warunków panujących podczas pomiaru, gdyż wnosi ona pewną, dodatkową pojemność pasożytniczą w układ pomiarowy. Przekroczenie jednego z progów napięć referencyjnych objawia się wygenerowaniem dodatniego impulsu sygnału S_{THR} , co analizowane jest przez mikrokontroler.



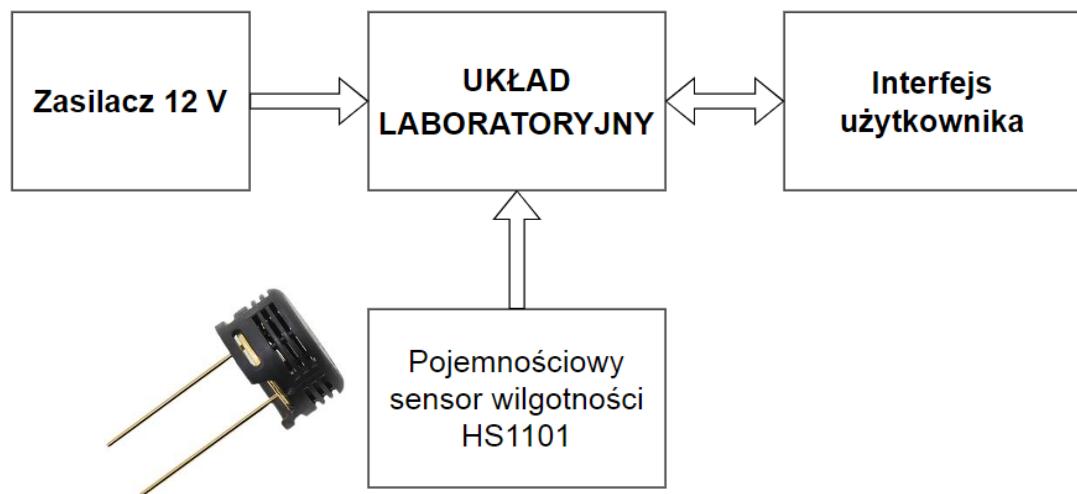
Rys. 3.10. Charakterystyka sygnałów w czasie pomiaru, $C = 177,2 \text{ pF}$, $R = 741,2 \text{ k}\Omega$.

4 BUDOWA UKŁADU LABORATORYJNEGO

W rozdziale zostanie zaprezentowana realizacja fizycznego układu inteligentnego czujnika pojemnościowego. Szczególną uwagę poświęcono omówieniu podkładów zasilania, obwodu pomiarowego, a także modułów Arduino Micro, czujnika temperatury i wilgotności względnej SHTC3 oraz wyświetlacza alfanumerycznego LCD.

4.1 Stanowisko laboratoryjne

Możliwość pomiaru układem laboratoryjnym jest możliwa tylko przy użyciu interfejsu użytkownika (rys. 4.1). Interfejs ten pozwala na pomiar jednokrotny pojemności czujnika HS1101 [14], oraz wielokrotny, z zaprogramowanym odstępem czasu. Ukazuje wyniki pomiarowe w aplikacji okienkowej. Oprócz pomiarów udostępnia funkcję kalibracji układu pomiarowego. Kalibracja umożliwia pomiar poziomu napięcia na wyjściach bufora trójstanowego, który taktuje blok pomiarowy. Interfejs użytkownika zapewnia także obliczenia oraz generuje skrypt programu Matlab, w którym użytkownik może dostosować do swoich wymagań wyniki pomiarowe. Interfejs użytkownika udostępnia możliwość zmian ustawień układu laboratoryjnego, m.in. poziomy napięć progowych i zasilania, czy współczynniki korekcji. Wartości nastaw są zapamiętywane w pamięci trwałej układu laboratoryjnego i aktualizowane wraz z podłączeniem do interfejsu. Dzięki wykorzystaniu protokołu USB interfejs użytkownika dokonuje automatycznej detekcji podłączonego układu i nawiązuje z nim komunikację.

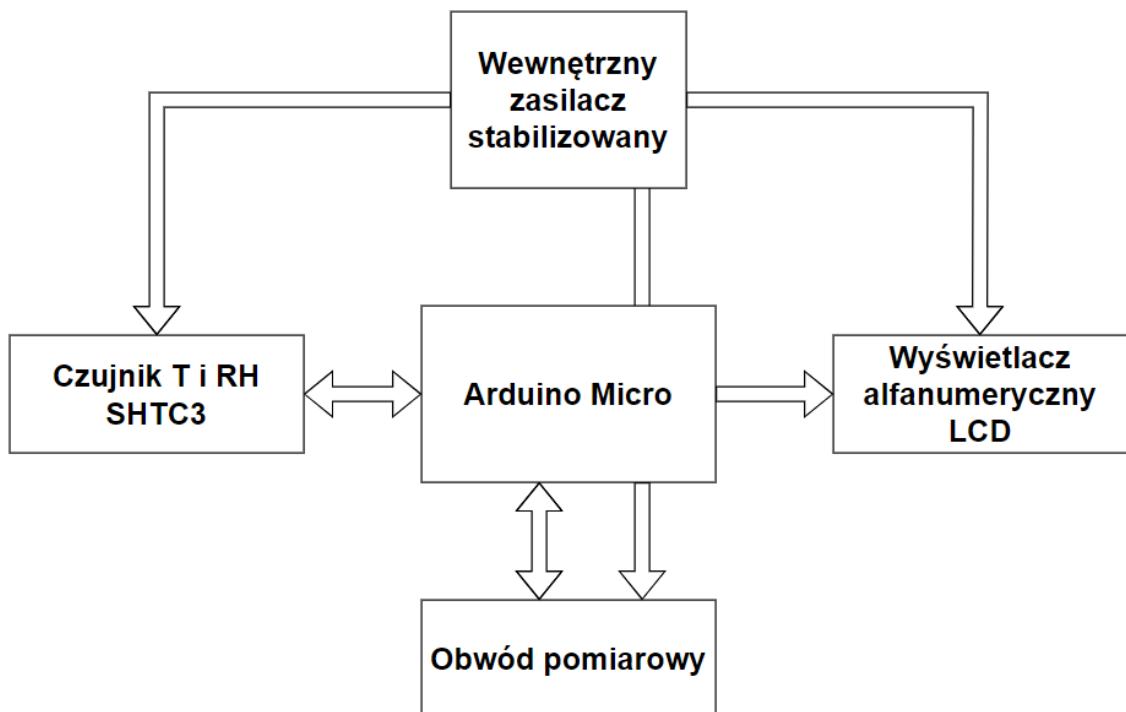


Rys. 4.1. Schemat blokowy stanowiska pomiarowego.

Do poprawnego działania układu laboratoryjnego wymagane jest zewnętrzne napięcie zasilające, o wartości przynajmniej 8 V. Zaś obiektem badanym jest pojemnościowy czujnik wilgotności względnej HS1101, którego pojemność zmienia się w zakresie 161 – 193 pF. Układ laboratoryjny posiada szerszy zakres badanych pojemności.

4.2 Inteligentny czujnik pojemnościowy

Głównym blokiem zarządzającym układem laboratoryjnym jest Arduino Micro, jest on połączony z obwodem pomiarowym (rys. 4.2). Komunikacja z obwodem pomiarowym odbywa się poprzez uniwersalny interfejs czujnik-mikrokontroler dla mikrokontrolerów z modułem przechwytywania zdarzeń zewnętrznych. Stabilny i dokładny pomiar zapewnia wewnętrzny zasilacz stabilizowany. Aby dodatkowo ograniczyć pobierany prąd z wewnętrznego zasilacza, zdecydowano, że moduł Arduino Micro zostanie zasilony z zewnętrznego napięcia, jest to możliwe gdyż wspomniany moduł posiada wbudowany stabilizator monolityczny. Kolejnym blokiem inteligentnego czujnika pojemnościowego jest sensor temperatury i wilgotności względnej SHTC3. Implementacja tego bloku zapewniła dodatkowe funkcje tj. przybliżone sprawdzenie poprawności wyznaczonej wilgotności z czujnika pojemnościowego HS1101. Oba sensory RH zostały umiejscowione możliwie blisko siebie, aby podobnie reagowały na zmiany parametrów otoczenia. Dodatkowa informacja o temperaturze pozwala uzyskać informacje o punkcie rosy, czyli o temperaturze, w której rozpoczęnie się proces skraplania gazu, co znacząco może wpływać na wskazania czujnika pojemnościowego HS1101. Układ laboratoryjny dodatkowo został wyposażony w układ prezentacji danych, czyli wyświetlacz alfanumeryczny LCD. Obecność tego bloku nie jest wymagana do poprawnego działania układu laboratoryjnego.



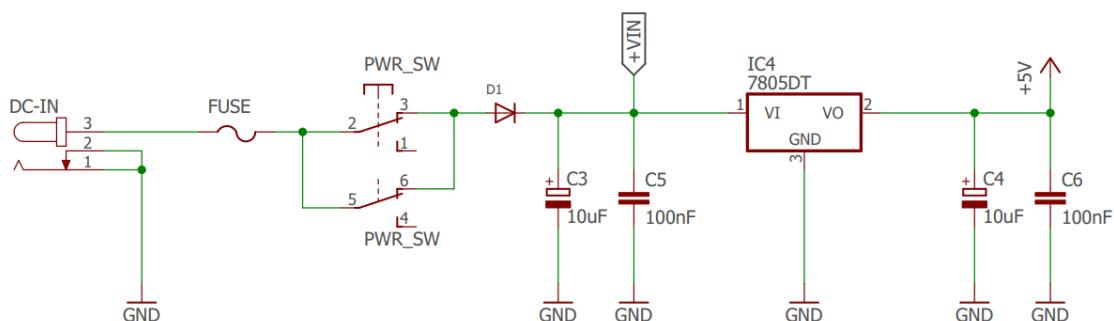
Rys. 4.2. Schemat blokowy układu laboratoryjnego.

4.3 Układ zasilania

Stabilność napięcia zasilającego z punku widzenia układu pomiarowego jest jednym z najważniejszych elementów niniejszej realizacji sprzętowej. Napięcie to wpływa na poziom napięcia wyjściowego z bufora trójstanowego, który stanowi sygnał wejściowy dla bloku pomiarowego. Stabilność napięcia w czasie pomiaru jest nieodzowna dla zapewnienia wysokiej dokładności pomiarowej, gdyż jednym z parametrów w formule 3.4 jest napięcie sygnału taktującego blok pomiarowy. Parametr ten wyklucza użycie zasilania z portu USB, ze względu na niestabilność napięcia, które może wawać się od 4,5 do 5,5 V [11].

Zastosowany w projekcie stabilizator monolityczny 7805 (rys. 4.3) posiada parametry wystarczające do zapewnienia należyjnej stabilności napięcia w układzie pomiarowym. Współczynnik temperaturowy napięcia wyjściowego wynosi zaledwie $-1,1 \text{ mV}^{\circ}\text{C}$ [15]. Czas pomiaru pojemności jest wielokrotnie mniejszy od bezwładności temperaturowej otoczenia, dlatego nie wpływa to na stabilizowane napięcie. Kolejnym parametrem jest regulacja napięcia w stosunku do zmian pobieranego prądu, który został przedstawiony w wielkości typowej 5 mV/A [15]. W trakcie pomiaru jedyna zmienność prądu będzie wynikała z ładowania i rozładowania pojemności badanej, pozostałe układy jak SHTC3 oraz LCD w tym czasie pozostają w spoczynku. Poziom wań prądu jest zależny od rezystancji pomiarowej, która powinna znajdować się w zakresie od $100 \text{ k}\Omega$ do $1 \text{ M}\Omega$, zatem w trakcie pełnego rozładowania pojemności prąd ten będzie znaczco ograniczony, nie wpływając tym samym na stabilizowane napięcie.

Układ zasilania został wyposażony w diodę prostowniczą, zabezpieczającą układ laboratoryjny przed podłączeniem zewnętrznego napięcia zasilającego o odwrotnej polaryzacji. Stabilizator napięcia do poprawnej pracy potrzebuje nadwyżkę 2 V zewnętrznego napięcia zasilania w stosunku do napięcia stabilizowanego, wliczając w to potencjał odkładający się na diodzie prostowniczej, układ laboratoryjny należy zasilać napięciem minimalnym 8 V.



Rys. 4.3. Schemat ideowy układu zasilającego.

4.4 Moduł Arduino Micro

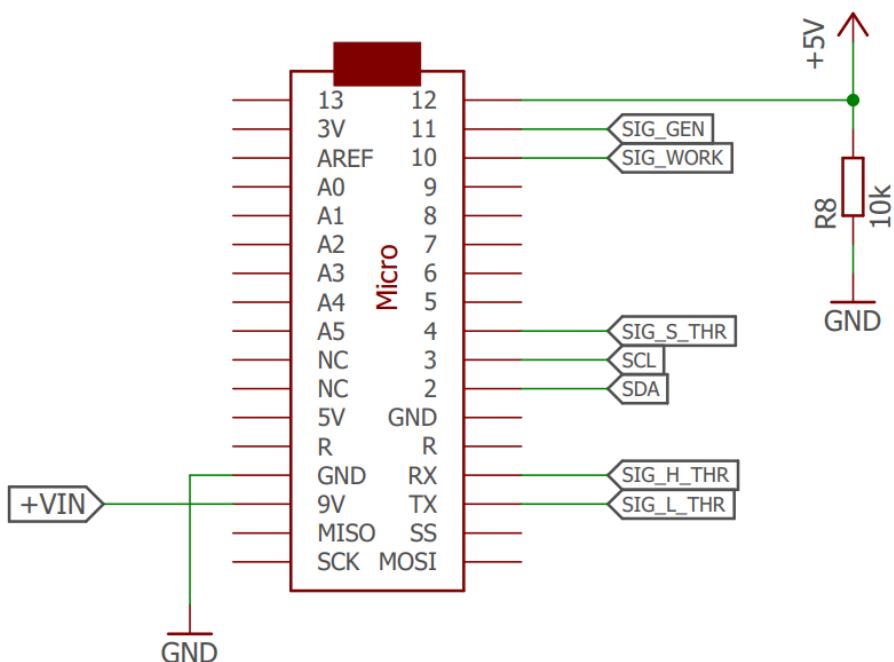
Wykorzystany moduł Arduino Micro (rys. 4.4) oparty jest o 8 bitowy mikrokontroler ATmega32U4 posiada wbudowane złącze Micro USB oraz rezonator kwarcowy o częstotliwości 16 MHz [12]. Na płytce PCB znajduje się także złącze do programowania szeregowego ISP, oraz 3 diody elektroluminescencyjne, które w projekcie sygnalizują wykonanie instrukcji przez układ laboratoryjny, podłączenie do komputera nadziednego, a także skomunikowanie z interfejsem użytkownika. Moduł posiada listwy kołkowe z wyprowadzeniami mikrokontrolera po obu stronach PCB. W projekcie został zrealizowany jako nakładka, możliwa do odłączenia. Moduł wyposażony został w stabilizator napięcia zasilania, o niewielkim poborze mocy. Mikrokontroler, przy zegarze 16 MHz, wymaga zasilania o wartości 5 V, wewnątrz modułu znajduje się także drugi stabilizator monolityczny o napięciu wyjściowym 3,3 V.



Rys. 4.4. Moduł Arduino Micro [10].

Podłączenie bloku Arduino Micro (rys. 4.5) odbywa się z wykorzystaniem interfejsu TWI (SDA, SCL) [11] z modułami wyświetlacza alfanumerycznego LCD oraz z czujnikiem temperatury i wilgotności względnej SHTC3. Magistralowa topologia pozwala na łatwą konfigurację i przyśpiesza rozbudowę układu. Sygnały wyjściowe, sterujące obwodem pomiarowym, to mi. in. sygnał aktywacji buforów trójstanowych (SIG_WORK) oraz sygnał taktujący człon pomiarowy RC (SIG_GEN). Pozostałe sygnały tj. przekroczenie progów napięć przez zgromadzony ładunek pojemności badanej (SIG_L_THR, SIG_H_THR) oraz suma tych zdarzeń (SIG_S_THR), poprzez bufor trójstanowy, są podłączone do wejść mikrokontrolera. Tak jak wspomniano, wykorzystano wewnętrzny stabilizator modułu do zasilania mikrokontrolera, w celu obniżenia poboru prądu ze stabilizatora 7805 zasilającego pozostałe bloki układu laboratoryjnego, w tym celu napięcie zewnętrzne (VIN) zostało podłączone do dedykowanego wejścia.

Dodatkowo, aby programowo sprawdzić obecność zewnętrznego napięcia zasilania, zastosowano rezystor R8. Podczas podłączenia Arduino Micro do portu USB, czerpie on z niego zasilanie, natomiast brak napięcia zewnętrznego powoduje, że wejście 12 pozostaje niepodłączone (nie panuje na nim żadne napięcie), dlatego wejście to musi zostać podciągnięte do masy układu poprzez rezystor R8. Odczyt logicznego „0” z portu 12 informuje mikrokontroler o braku zewnętrznego napięcia zasilającego, generuje on wtedy stosowny komunikat do interfejsu użytkownika.

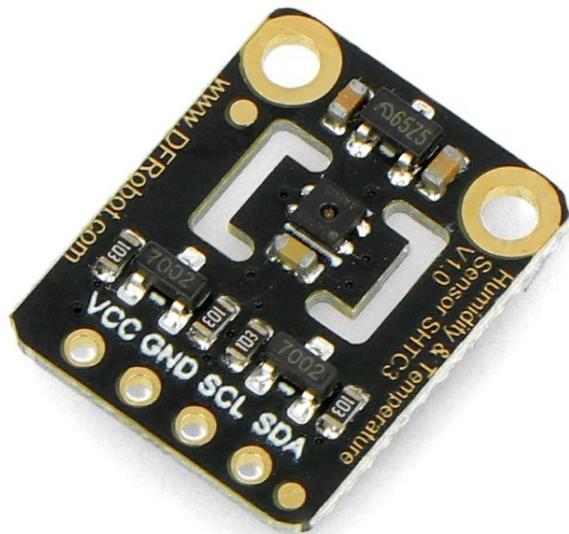


Rys. 4.5. Schemat ideowy podłączenia modułu Arduino Micro.

4.5 Moduł SHTC3

Służący do pomiaru temperatury i wilgotności względnej, moduł SHTC3 został zaimplementowany w celu potwierdzenia poprawności wyznaczonej wartości wilgotności względnej uzyskanej w drodze pomiarów. Nie jest to jednak urządzenie na tyle dokładne, aby kalibrować układ pomiarowy, ponieważ jego typowa dokładność dla wilgotności względnej wynosi $\pm 2\% \text{ RH}$ [13]. Dodatkowa funkcja pomiaru temperatury pozwala na uzyskanie informacji o temperaturze punktu rosy, która określa początek skraplania się gazu. Moment skroplenia się gazu może być istotny z punktu widzenia dokładności pomiarów, gdyż pomiar pojemności wilgotnego czujnika HS1101 może wprowadzać błąd. Układ laboratoryjny umożliwia śledzenie i wyciągnięcie wniosków z otrzymanych pomiarów. Pomiar temperatury i wilgotności zajmuje czujnikowi mniej niż 20 ms, co umożliwia każdorazowy pomiar tych parametrów nawet podczas wielokrotnych serii pomiarowych.

Moduł SHTC3 składa się z czujnika, pracującego na poziomach logicznych 3,3 V, na płytce PCB (rys. 4.6) zaimplementowano zarówno konwertery stanów logicznych z 5 V na poziom 3,3 V oraz monolityczny stabilizator napięcia zasilający czujnik, umożliwia to bezpośrednie podłączenie modułu do układów pracujących w poziomie 5 V [10].



Rys. 4.6. Moduł SHTC3 [10].

4.6 Moduł LCD

Wykorzystanie wyświetlacza alfanumerycznego LCD (rys. 4.7) ma na celu prezentację danych. Obecność wyświetlacza nie jest konieczna do prawidłowego działania układu. Wykorzystano sterowanie poprzez interfejs TWI, dzięki czemu mikrokontroler automatycznie wykrywa jego obecność podczas inicjalizacji systemu i analizuje w pętli głównej programu. Zastosowanie ekspandera wyjść PCF8574t znaczco ogranicza ilość wyprowadzeń, co jest zaletą, ponieważ ze względu na wymiary płytki PCB, zdecydowano żeby wyświetlacz był dołączany do układu laboratoryjnego za pomocą przewodów. Do tego celu potrzeba 4 przewodów, z czego 2 służą jako zasilanie modułu oraz pozostałe 2 zapewniają komunikację z ekspanderem wyjść. Dedykowany ekspander do sterowania wyświetlaczami LCD posiada wbudowany potencjometr, służący do regulacji kontrastu [10]. Interfejs TWI wymaga także rezystorów podciągających do zasilania, ponieważ wyjścia tego interfejsu występują w konfiguracji otwartego kolektora. W tym celu zaimplementowano w układzie laboratoryjnym rezystory R6 oraz R7, o wartości rezystancji 10 kΩ.

Po każdym pomiarze, interfejs użytkownika wysyła do układu laboratoryjnego obliczone wartości pojemności, wilgotności względnej i temperatury. Poza tym wysyłany jest średni czas ładowania i rozładowania badanej pojemności czujnika, ten parametr został udostępniony użytkownikowi tylko poprzez wyświetlacz LCD.

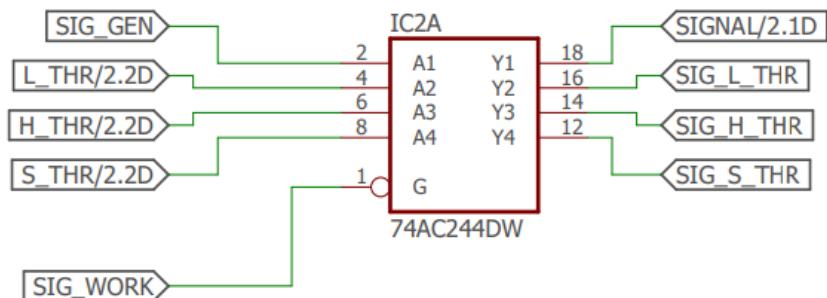


Rys 4.7. Moduł wyświetlacza alfanumerycznego LCD [10].

4.7 Układ pomiarowy

Blok pomiarowy składa się z bufora trójstanowego, członu RC, toru przetwarzania sygnału pomiarowego oraz toru napięć referencyjnych. Jednym z ważniejszych, w tym zagadnieniu, parametrów są stabilne napięcia odniesienia, możliwe najniższe napięcie niezrównoważenia komparatorów oraz niewielki ich prąd polaryzacji obwodów wejściowych.

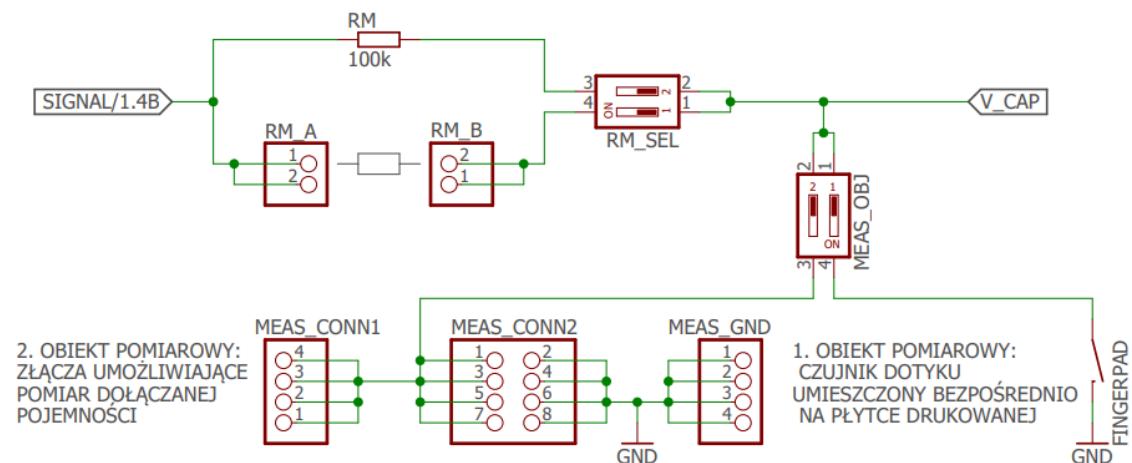
Bufor trójstanowy pełni rolę izolacji obwodu wykonującego pomiar, od mikrokontrolera (rys 4.8). Ze względu na wysoką integrację struktury krzemowej, wyjście mikrokontrolera posiada większe pojemności pasożytnicze. Dodatkowo, zwiększoną stabilność napięcia wyjściowego oraz większy prąd uzyskiwany z wyjścia, są cechami buforów trójstanowych. Kolejną zaletą tych układów jest dodatkowy stan wysokiej impedancji, co znacząco mniejsza pobór prądu w stanie uśpienia układu [17].



Rys 4.8. Schemat ideowy bufora trójstanowego.

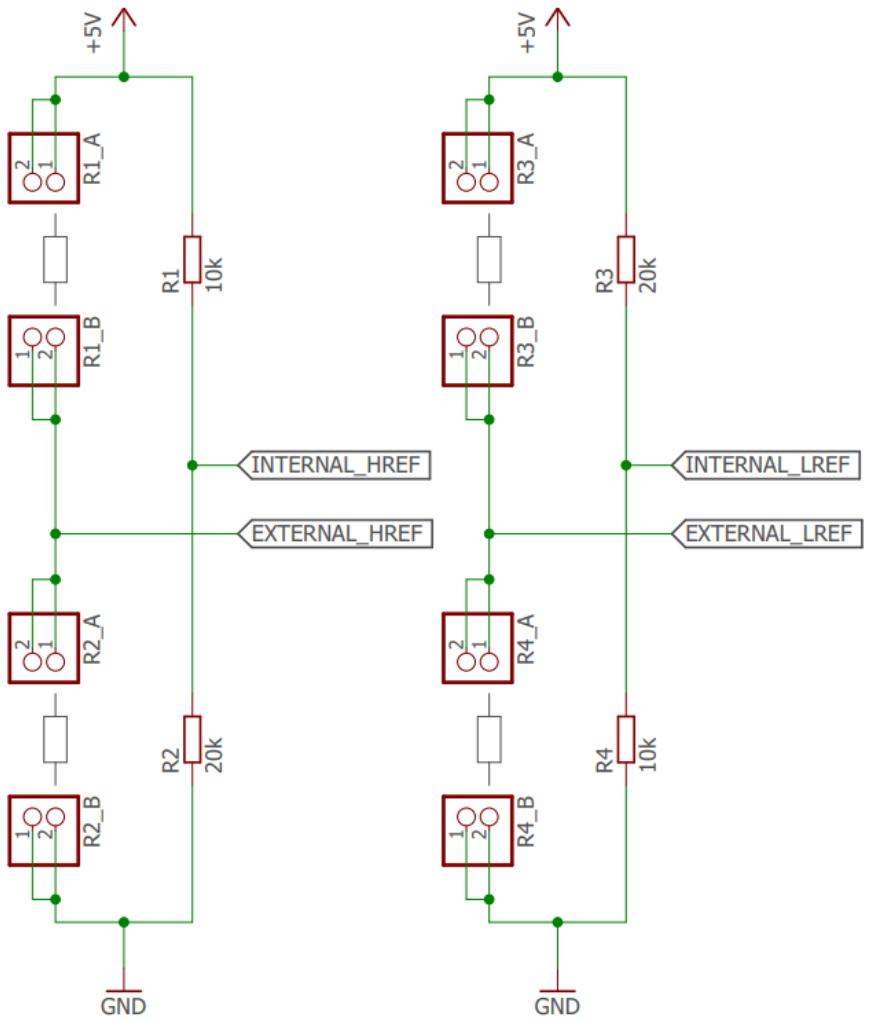
Człon pomiarowy RC (rys. 4.9) składa się z rezystora pomiarowego RM oraz dołączanej pojemności badanej. Dodatkowo, na płytce PCB zaimplementowano czujnik dotyku, którego działanie opiera się na zmianie pojemności pod wpływem przyłożonego palca. Pojemność rośnie wraz ze wzrostem powierzchni stylu czujnika oraz skóry człowieka. Na pojemność mają również wpływ parametry skóry np. jej wilgotność. Podłączenie czujnika natomiast dobiera się poprzez listwy kołkowe. Do wyboru obiektu pomiarowego służą przełączniki DIPSWITCH.

Wartość rezystancji rezystora pomiarowego jest jednym z parametrów obliczeniowych, dlatego należy zmierzyć tą wartość z możliwie największą dokładnością. Układ laboratoryjny umożliwia podłączenie rezystora wybranego przez użytkownika. Wartość rezystancji wbudowanej w układ pomiarowy wynosi $100\text{ k}\Omega$. Wybór wbudowanego lub zewnętrznego rezystora pomiarowego możliwy jest przez konfigurację przełączników DIPSWITCH.



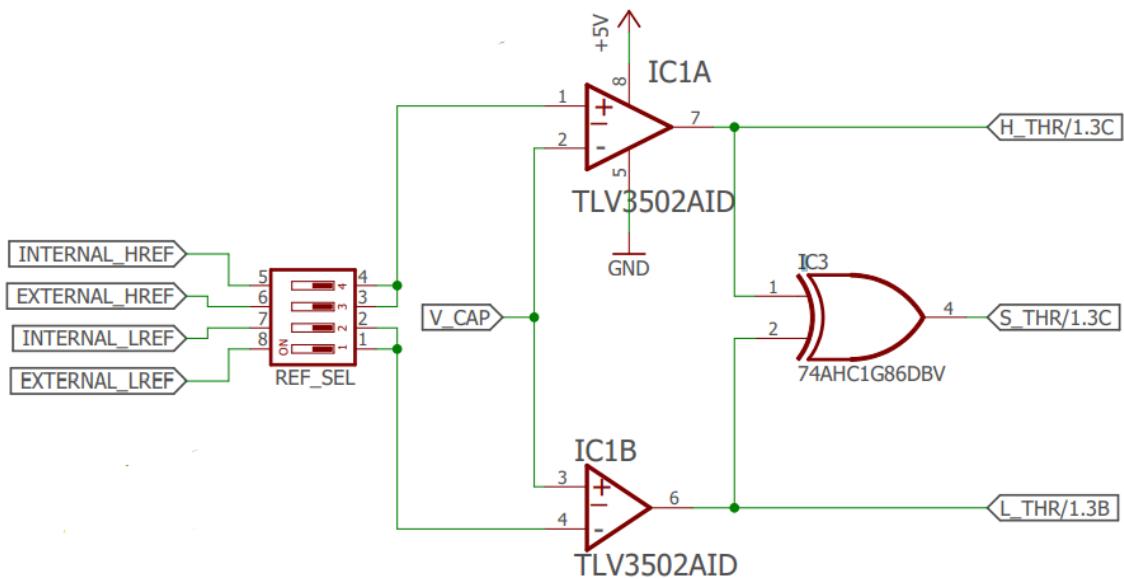
Rys. 4.9. Schemat ideowy członu pomiarowego RC.

Uzyskując stabilne napięcie zasilania możliwa jest konfiguracja napięć odniesienia za pomocą precyzyjnych dzielników rezystancyjnych (rys. 4.10). Zaimplementowano możliwość wyboru dzielników wbudowanych (R_1 , R_2 ; R_3 , R_4) oraz zewnętrznych, konfigurowanych przez użytkownika. Do konfiguracji służą żeńskie listwy kołkowe, w których użytkownik ma możliwość podłączenia dowolnego dzielnika. Zaś wbudowane zapewniają stosunek 2/3 napięcia zasilania (w przypadku górnego progu) oraz 1/3 napięcia zasilania (w przypadku dolnego progu). Zastosowanie rezystorów do realizacji potencjałów odniesienia niesie ze sobą zaletę, ponieważ wahania napięcia zasilającego mają wpływ na wyjście dzielnika, kompensując tym samym napięcie referencyjne. Na wspomnianych rezystorach wydziela się niewielka moc, nie powodując zmian rezystancji ze względu na wydzielane ciepło. Wybór obsługiwanej potencjału odniesienia obsługiwany jest poprzez przełączniki DIPSWITCH.



Rys. 4.10. Schemat ideowy toru napięć referencyjnych.

Tor przetwarzania sygnału pomiarowego (rys. 4.11) oparty jest o komparator okienkowy. Dla dokładności pomiarowej, ważne jest aby dobrać komparator o możliwie najmniejszym prądzie polaryzacji obwodów wejściowych. Wybrany został układ TLV3502AID, którego parametry są wystarczające do zapewnienia należytej dokładności pomiarowej, a zarazem posiada on przystępna cenę. Prąd polaryzacji został określony w dokumentacji na wartość ± 20 pA [16]. Ważnym parametrem jest także napięcie niezrównoważenia, które określa przesunięcie charakterystyki przejściowej, od idealnego komparatora. Wartość ta została zapewniona przez producenta w zakresie 1 – 6,5 mV [16]. Natomiast czas propagacji sygnału wyjściowego tego układu wynosi 12 ns [16]. Wartość ta jest kilku krotnie mniejsza od jednego cyklu zegarowego mikrokontrolera, która dla zegara 16 MHz wynosi 62,5 ns. Zatem czas propagacji nie wpłynie na opóźnienie odczytu sygnałów przez mikrokontroler. Chęć wykorzystania modułu przechwytywania zdarzeń zewnętrznych wymusza sprowadzenie obu sygnałów progowych do jednej linii. Za tę cechę jest odpowiedzialna bramka logiczna XOR. Suma obu sygnałów progowych jest automatycznie rozpoznawana przez mikrokontroler.

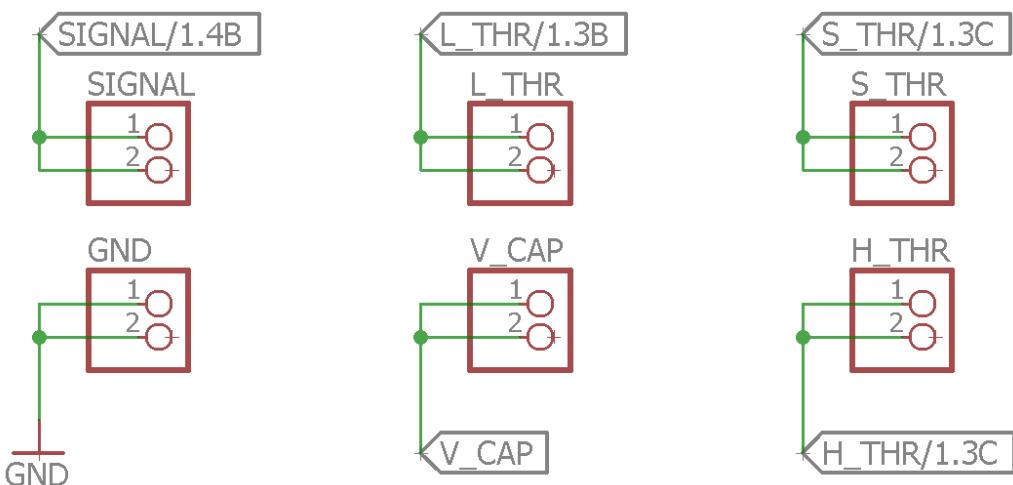


Rys. 4.11. Schemat ideowy toru przetwarzania sygnału pomiarowego.

4.8 Złącza diagnostyczne

Diagnostyka sygnałów w układzie laboratoryjnym jest jedną z ważniejszych elementów dydaktycznych, pozwala użytkownikowi zapoznać się i zrozumieć metodykę pomiarową. W celu realizacji projektu zastosowano złącza diagnostyczne w postaci listew kołkowych, które umożliwiają podłączenie oscyloskopu. Każda sonda oscyloskopowa wnosi pewną, niewielką pojemność do układu, więc samo podłączenie sondy wywoła zmianę badanej pojemności. Aby zminimalizować wpływ sondy należy przestawić ją w tryb pracy „x10”, wykazuje ona w tym trybie mniejszą pojemność.

Złącza diagnostyczne umożliwiają podejrzenie charakterystyki przejściowej sygnału ładowania i rozładowania pojemności. Podgląd przebiegu w tym punkcie najbardziej narażony jest na działanie pojemności pasożytniczej sondy pomiarowej. Wyprowadzony został również sygnał taktowania bloku pomiarowego. Podczas kalibracji urządzenia, mikrokontroler wystawia sygnał prostokątny o częstotliwości 1 MHz, na tym wyprowadzeniu, umożliwia to doświadczalne sprawdzenie opóźnień działania komparatora oraz poziomy napięć wyjściowych bufora trójstanowego. Wyprowadzone zostały również sygnały osiągnięcia progu (L_THR – osiągnięty dolny próg, H_THR – osiągnięty górny próg), a także suma powyższych zdarzeń (S_THR). Dzięki tym sygnałom możliwe jest stwierdzenie, przy jakim poziomie napięcia na pojemności zareaguje komparator.



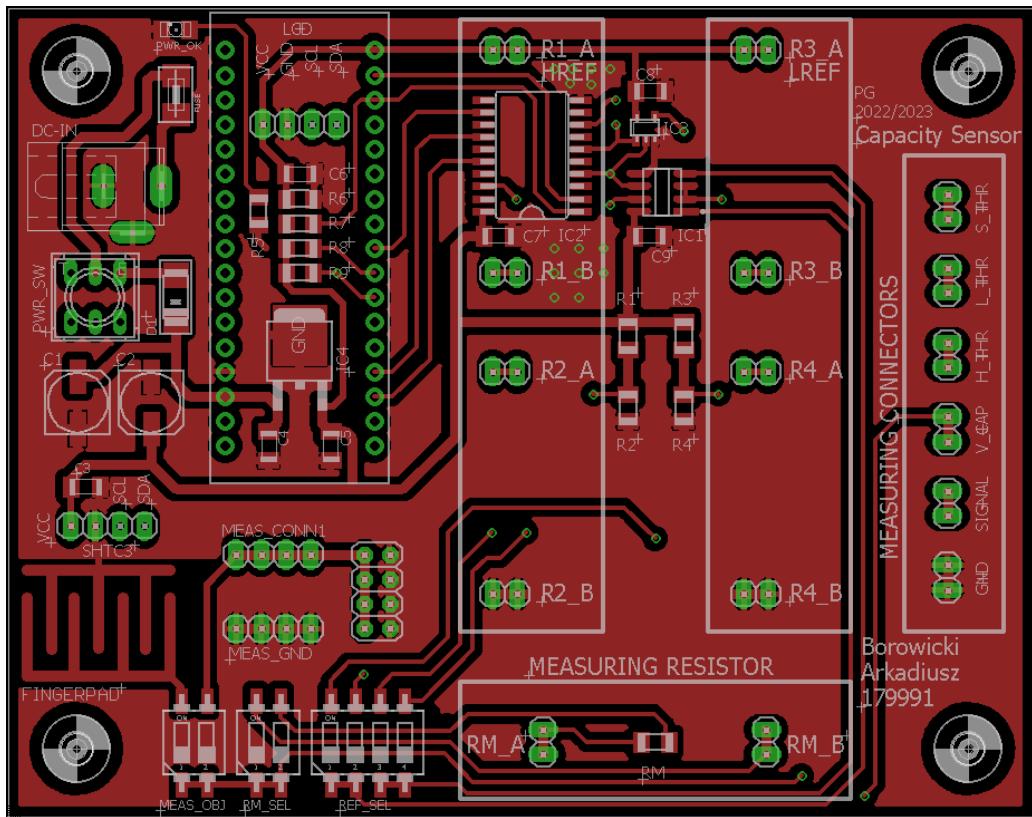
Rys. 4.12. Schemat ideowy złącz diagnostycznych.

4.9 Layout PCB

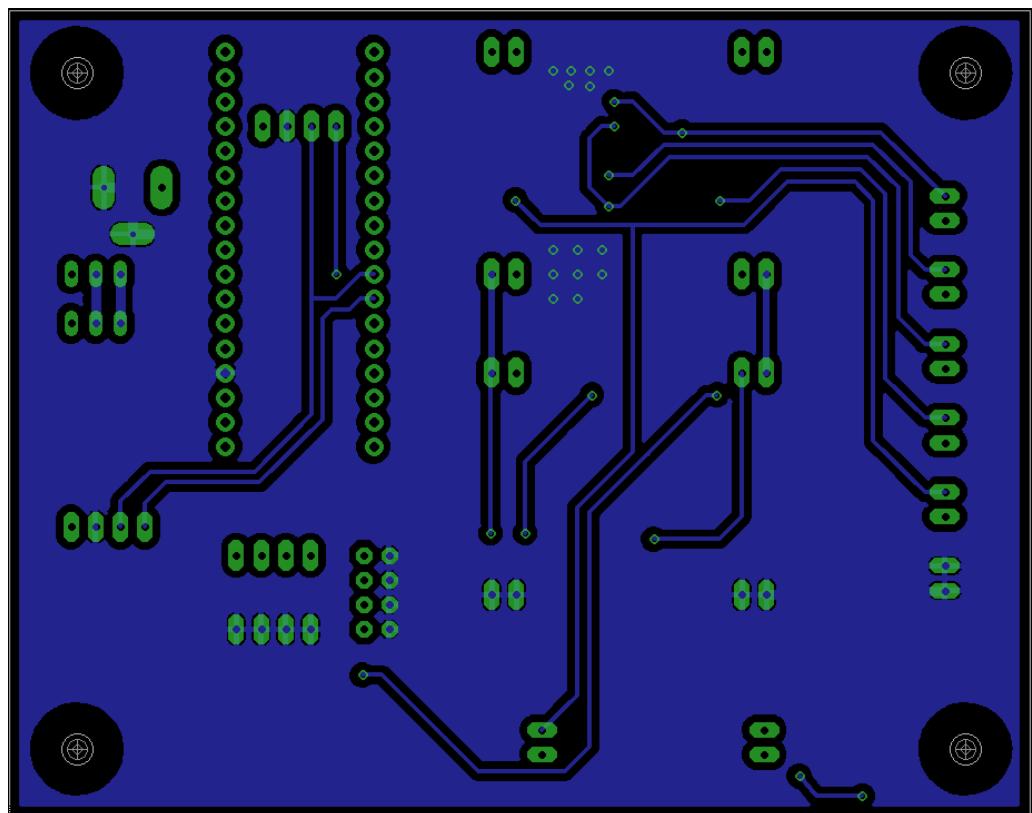
W podrozdziale zostanie zaprezentowany projekt mozaiki PCB oraz realizacja rzeczywistego układu. Szczególną uwagę poświęcono sposobom prowadzenia ścieżek oraz umiejscowienia elementów elektronicznych.

4.9.1 Projekt

Projekt mozaiki PCB został opracowany przy użyciu programu Autodesk Eagle (rys. 4.13; rys. 4.14). Wynikowa płytka posiada wymiary 105 x 83 mm. Projekt wykonano w formie 2 stronnej płytki o szerokości ścieżek sygnałowych na poziomie 16 milsów oraz ścieżek zasilających – 40 milsów. Podane szerokości ścieżek są wystarczające do zapewnienia minimalnej rezystancji oraz wystarczające do przenoszenia prądów przepływających przez układ laboratoryjny. Ponadto zastosowano rozlew masy po obu stronach płytki, w celu ograniczenia szumów mogących wpływać na dokładność pomiaru pojemności. Rozlew masy ma za zadanie ekranowanie ścieżek sygnałowych. Wadą tego rozwiązania jest zwiększoną pojemność ścieżek w stosunku do masy. Warstwa miedzi została pokryta soldemaską, której zadaniem jest zabezpieczenie pól kontaktowych przed zwarciami oraz zanieczyszczeniami. Aby ograniczyć ilość otworów na płytce, zdecydowano o montażu powierzchniowym SMD. Wszystkie układy elektroniczne, dostępne w tym typie montażu zostały zaimplementowane. Z punktu widzenia użytkownika, ważne jest także umiejscowienie elementów, złącza diagnostycznych, aby w przejrzysty sposób korzystać z urządzenia. Wejście zasilania zostało umiejscowione, w górnym lewym rogu płytki zaraz obok znajduje się przełącznik zasilania. Złącza diagnostyczne natomiast jako wyjścia układu zostały umiejscowione z prawej strony płytki PCB. Wszystkie elementy bloków pomiarowych starano się umieścić możliwie blisko siebie. Zainstalowano także podstawki, aby ograniczyć miejsce styku płytki z podłożem, na którym stoi.



Rys. 4.13. Projekt mozaiki PCB – góra strona.

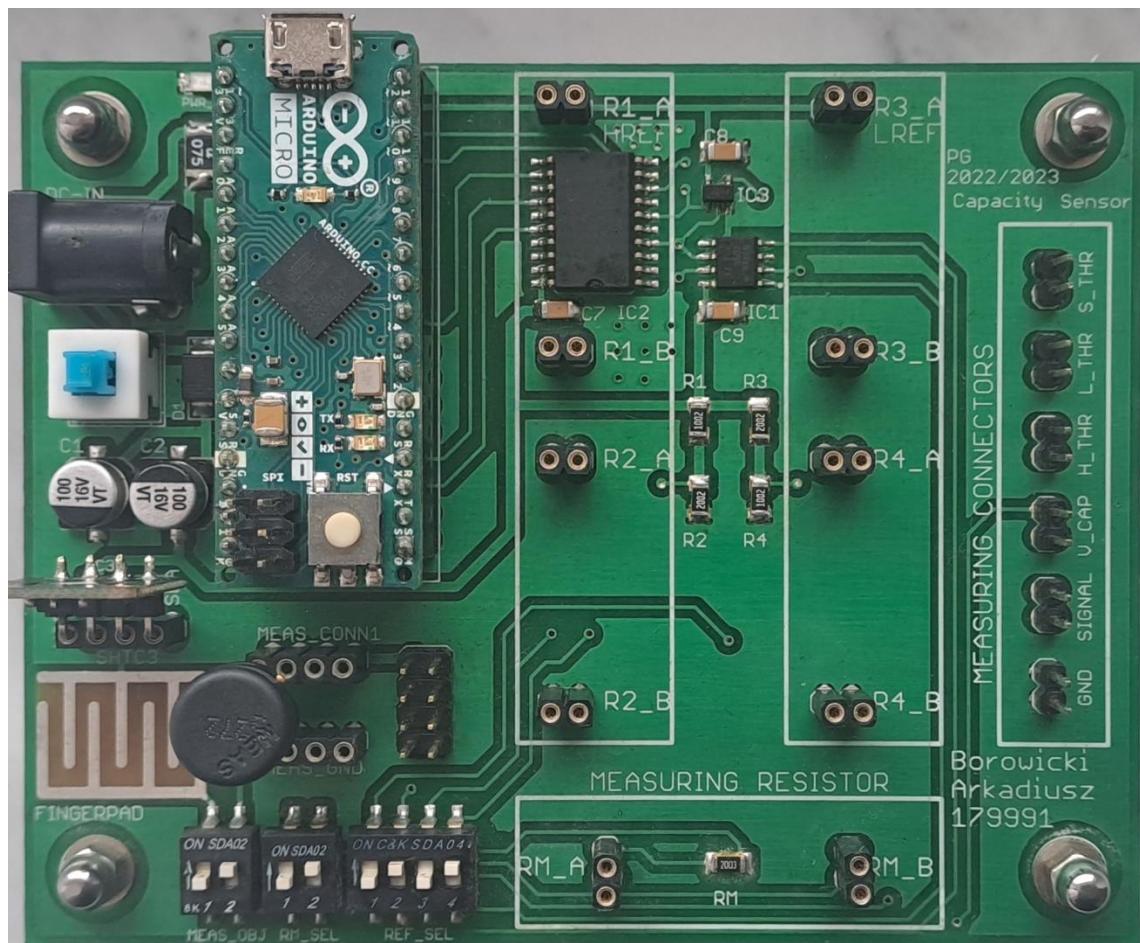


Rys. 4.14. Projekt mozaiki PCB – dolna strona.

4.9.2 Realizacja

Rzeczywisty układ laboratoryjny (rys. 4.15) posiada nakładkę w postaci modułów Arduino Micro, czujnika temperatury i wilgotności względnej SHTC3 oraz LCD. Moduł LCD został wyprowadzony za pomocą przewodów, a kątowe złącze kołkowe umożliwiające jego podłączenie znajduje się pod modułem Arduino Micro. Układ laboratoryjny posiada także miejsce umożliwiające dołączenie czujnika pojemnościowego, co czyni ten układ uniwersalnym, mogącym pracować z różnymi czujnikami. Na rynku są dostępne także inne moduły służące do pomiaru wilgotności, np. czujnik pary wodnej, którego działanie opiera się o badanie pojemności, czy czujnik wilgotności gleby. Miejsce na dołączenie tych czujników zostało wyprowadzone w postaci męskich listew kołkowych, znajdujących się obok czujnika HS1101. Moduł SHTC3 umiejscowiono możliwie blisko czujnika pojemnościowego HS1101, aby jak najbardziej zbliżyć warunki pracy.

Programowanie szeregowego mikrokontrolera odbywa się poprzez dedykowane wyprowadzenie złącza ISP. Moduł posiada także wbudowane złącze Micro USB, służące do komunikacji z interfejsem użytkownika.



Rys. 4.15. Realizacja fizyczna układu laboratoryjnego.

5 UŻYTE UKŁADY PERYFERYJNE MIKROKONTROLERA

W rozdziale zostanie zaprezentowana struktura wewnętrzna mikrokontrolera ATmega32U4. Szczególną uwagę skupiono na objaśnieniu konfiguracji układów peryferyjnych, wraz z omówieniem rejestrów pracy, a także na zaletach wykorzystania interfejsu TWI.

5.1 System przerwań

System przerwań jest jeden z ważniejszych układów peryferyjnych mikrokontrolerów, pozwala przerwać aktualnie wykonujący się ciąg instrukcji programu głównego, na skutek wystąpienia określonego zdarzenia. Pozwala to na możliwie szybką reakcję i obsługę przerwania. Program obsługujący zdarzenie powinien składać się z niewielkiej liczby instrukcji, w ten sposób działanie programu głównego nie zostanie zaburzone.

Każde, uprzednio zaprogramowane, wystąpienie zdarzenia od układu peryferyjnego objawia się ustawieniem bitu żądania obsługi. W pierwszej kolejności zostają obsłużone przerwania o najwyższym priorytecie. System obsługuje również zagnieżdżanie przerwań, wybierając w danej chwili, które zdarzenie potrzebuje natychmiastowej reakcji. Dokumentacja mikrokontrolera określa, że czas pomiędzy wystąpieniem zdarzenia a jego obsługą wynosi minimum 5 cykli zegara taktującego [11]. W tym czasie następuje zapisanie w pamięci stosu aktualnej wartości licznika programu, którego wartość w następnym kroku zostaje ustawiona na przypisaną dla danego przerwania, czyli domyślny wskaźnik na tablicę przerwań. Po zakończeniu przerwania w sposób analogiczny licznik programu zostaje załadowany wartością z pamięci stosu, następnie zerowana jest flaga żądania obsługi przerwania. Cała procedura powrotu do wcześniejszy wykonywanego programu ponownie zajmuje minimum 5 cykli zegarowych.

Przy chęci wykorzystania systemu przerwań konieczne jest zaprogramowanie rejestrów statusowych mikrokontrolera (rys. 5.1). Bit 7 (I – Global Interrupt Enable) jest globalnym włącznikiem [11], przerwania zostają obsługiwane dopiero w momencie ustawienia tego bitu na logiczną wartość „1”. Dodatkowo każdy układ peryferyjny posiada maski przerwań, które uprzednio należy zaprogramować, wybierając dozwolone źródło przerwania.

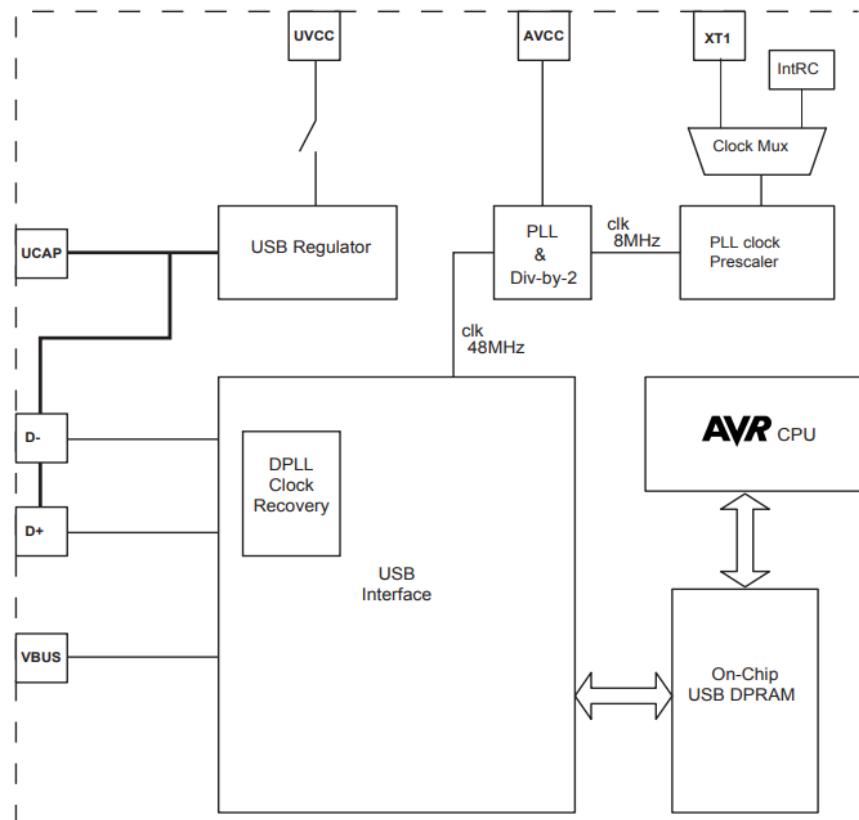
Bit	7	6	5	4	3	2	1	0	SREG
Read/Write	R/W								
Initial Value	0	0	0	0	0	0	0	0	

Rys. 5.1. Rejestr statusowy mikrokontrolera ATmega32U4 [11].

5.2 USB – Universal Serial Bus

Do obsługi interfejsu USB wbudowanego w mikrokontroler, podczas realizacji projektu dyplomowego posłużyono się dedykowaną biblioteką LUFA USB, rekomendowaną przez firmę Microchip. Jednakże, aby zapoznać się z mechanizmami funkcjonowania interfejsu, konieczne jest pogłębienie informacji z dokumentacji mikrokontrolera. Biblioteka pozwala na konfigurację deskryptorów oraz identyfikatora, co umożliwiło zaimplementowanie automatycznej detekcji układu laboratoryjnego podłączonego do komputera, w interfejsie użytkownika.

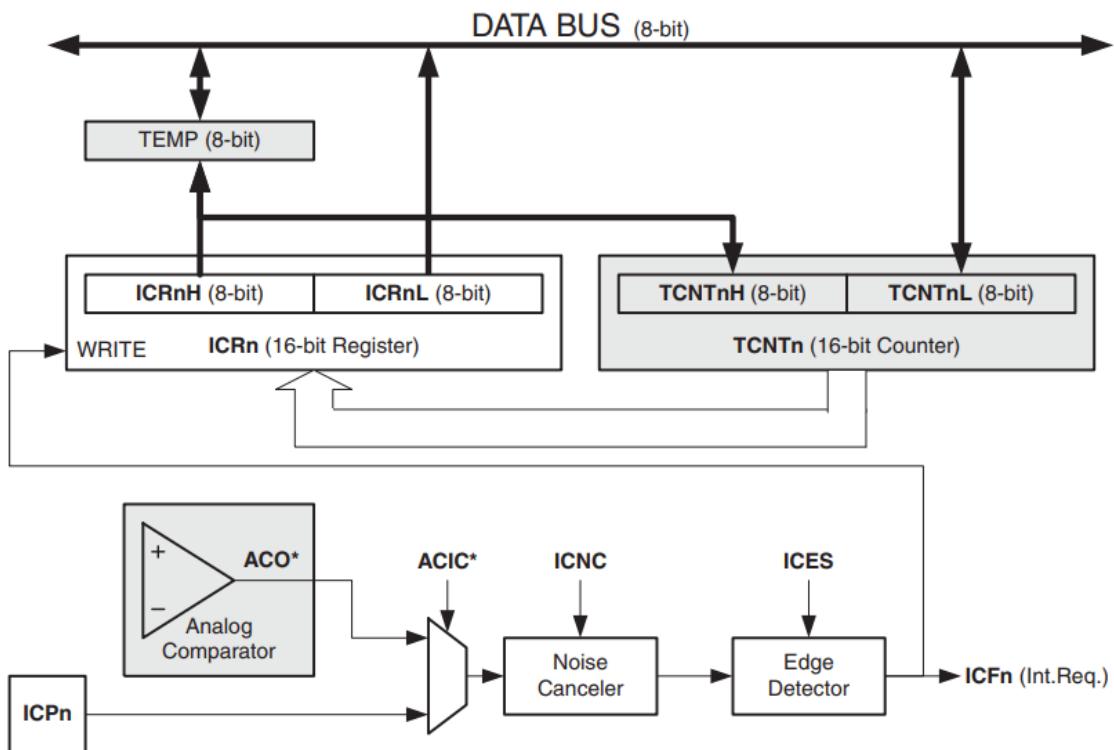
Interfejs USB umożliwia obsługę zdarzenia podłączenia do komputera, a także nawiązania połączenia z aplikacją. Moment, w którym mikrokontroler wykryje pojawienie się napięcia zasilania na dedykowanym wyprowadzeniu VBUS [11], równoważne jest podłączeniu do portu USB komputera, wywoływane jest przerwanie. System przerwań jest konieczny do poprawnego działania kontrolera USB. Natomiast połączenie aplikacji z mikrokontrolerem odbywa się poprzez przesłanie protokołu, po którym również generowane jest przerwanie. Praca mikrokontrolera z maksymalną prędkością, przy częstotliwości zegara 16 MHz, możliwa jest tylko przy zasilaniu napięciem 5V, dlatego w strukturze wewnętrznej zaimplementowano regulatory napięcia (rys. 5.2), dostosowujące wyprowadzenia danych do standardu USB (3 – 3.6 [V]).



Rys. 5.2. Diagram kontrolera USB [11].

5.3 ICP – Input Capture

Do realizacji projektu wykorzystano 16 bitowy licznik Timer/Counter1, z wyłączonym dzielniczką częstotliwości sygnału taktującego, aby zmaksymalizować rozdzielcość pomiaru czasu. Takowy licznik posiada możliwość pracy w trybie przechwytywania zdarzeń zewnętrznych. Realizuje on funkcję zliczania impulsów sygnału taktującego, wysyłając sygnał przerwania po wykryciu zaprogramowanego zbocza na dedykowanym wyprowadzeniu mikrokontrolera i zapamiętując aktualny stan licznika w osobnym rejestrze (rys. 5.3). Układ peryferyjny został wyposażony w filtr szumów, tłumiący krótsze脉波 od 4 cykli zegarowych, w rezultacie filtr ten wprowadza opóźnienie czasowe, 4 cykli zegarowych [11], dla sygnału podawanego na detektor zbocza. Odmierzanie cykli czasowych pomiędzy wystąpieniem kolejnych zdarzeń zostało obsłużone z wykorzystaniem systemu przerwań.



Rys. 5.3. Schemat blokowy układu licznika w trybie przechwytywania zdarzeń [11].

Licznik pracuje w trybie normalnym, zliczając脉波 zegarowe od wartości 0 do 65535, czas w którym licznik zliczy pełen zakres impulsów wynosi 4,096 ms z wyłączonym dzielniczką częstotliwości zegara taktującego licznik [11]. Natomiast maksymalny szacowany czas ładowania i rozładowania pojemności z zadanego zakresu 100 – 300 pF został określony w rozdziale 3 na około 200 μs. Dodatkowym zaimplementowanym zabiegiem zwiększającym dokładność pomiaru jest nad próbkowanie i decymacja, która pozwala zwiększyć rozdzielcość pomiaru czasu.

Konfiguracja licznika (rys. 5.4) opiera się na zaprogramowaniu obecności filtra szumów w układzie, odpowiedzialny za to jest bit 7 w rejestrze TCCR1B (ICNC1 – Input Capture Noise Canceler) [11]. Tryb normalnego zliczania jest domyślnie ustawiony po sygnale zerowania. Uruchomienie licznika odbywa się poprzez wybór sygnału zegarowego (CS12:10), maksymalną prędkość zliczania uzyskuje się dla wyłączonego dzielnika częstotliwości (CS12 = „0”; CS11 = „0”; CS10 = „1”). Za wybór detekcji zbocza decyduje stan bitu 6 w rejestrze TCCR1B (ICES1 – Input Capture Edge Select), wartość logiczna „1” oznacza zbocze narastające, natomiast logiczne „0” – opadające [11].

Bit	7	6	5	4	3	2	1	0	
Read/Write	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10	TCCR1B
Initial Value	0	0	0	0	0	0	0	0	

Rys. 5.4. Rejestr konfiguracyjny licznika 1 [11].

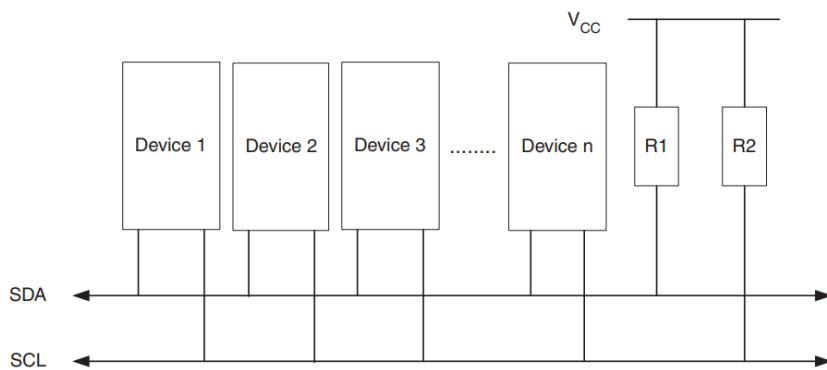
Zezwolenie na przerwanie od układu przechwytywania zdarzeń dla licznika 1 opisuje bit 5 (ICIE1 – Input Capture Interrupt Enable) w rejestrze TIMSK1 (rys. 5.5). Chcąc aktywować przerwanie, należy ustawić powyżej wspomniany bit rejestru na wartość logiczną „1”.

Bit	7	6	5	4	3	2	1	0	
Read/Write	-	-	ICIE1	-	OCIE1C	OCIE1B	OCIE1A	TOIE1	TIMSK1
Initial Value	0	0	0	0	0	0	0	0	

Rys. 5.5. Rejestr masek przerwań licznika 1 [11].

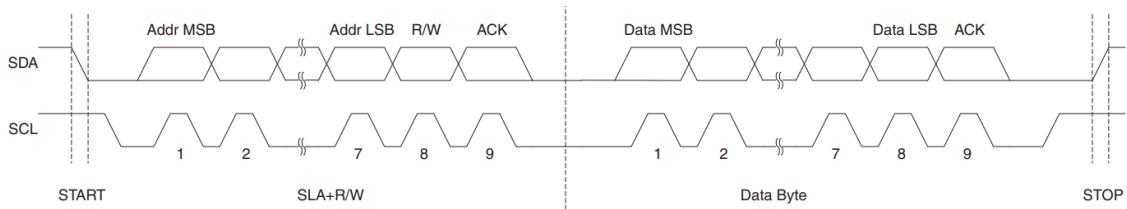
5.4 TWI – Two Wire Interface

Komunikacja z układami zewnętrznymi tj. wyświetlacz LCD z ekspanderem wyprowadzeń PCF8574t oraz czujnik temperatury i wilgotności względnej SHTC3, odbywa się poprzez szeregowy, synchroniczny interfejs TWI. Zaletą tego interfejsu jest jawne potwierdzenie odbioru danych przez układ podrzędny, wykorzystujący operację logiczną AND na drucie. W momencie wystąpienia dziewiątego bitu transmisji danych z układu master do slave, urządzenie podrzędne zwiera linię danych na czas jednego cyklu zegarowego sygnału taktującego transmisję. Mechanizm ten został wykorzystany do wykrycia obecności układów podrzędnych w urządzeniu laboratoryjnym, dzięki czemu brak lub odłączenie jednego z układów nie zaburza działania programu głównego mikrokontrolera. Operacja logiczna AND na drucie jest możliwa poprzez zastosowanie wyprowadzeń linii danych (SDA) i zegara (SCL) w konfiguracji otwartego kolektora. Stan logiczny „1” wymuszany jest poprzez rezystory podciągające R1 i R2 (rys. 5.6), do napięcia zasilającego. Oznacza to, że w stanie bezczynności linie danych i zegara są w stanie logicznym „1”. Zastosowanie wyprowadzeń z otwartym kolektorem umożliwia także detekcję arbitrażu danych na liniach danych i zegara, co pozwala na pracę interfejsu w konfiguracji z wieloma urządzeniami nadzorowanymi.



Rys 5.6. Sposób podłączenia urządzeń w interfejsie TWI [11].

Transmisja danych poprzedzona jest sygnałem startu (rys 5.7), w następnej kolejności wysyłany jest adres urządzenia podrzędnego. Układ nadzorzący odpowiedzialny jest za wysyłanie sygnału zegarowego w całym cyklu transmisyjnym, oraz za wywołanie układu do komunikacji, poprzez transmisję adresu urządzenia, również w przypadku odbioru danych.



Rys 5.7. Typowa transmisja danych interfejsu TWI [11].

Po każdej paczce bajtowych danych występuje bit potwierdzenia odbioru, wystawiany przez obecnie odbierające urządzenie. Zakończenie transmisji sygnalizowane jest przez układ nadzorzący, poprzez wywołanie sygnału stop. Elementy transmisji wywoływane są poprzez ustawienie poszczególnych bitów w rejestrze kontrolnym interfejsu TWI (rys. 5.8). Warunek startu transmisji odbywa się poprzez ustawienie wartości logicznej „1” bitu TWSTA, natomiast warunek stopu analogicznie dla bitu TWSTO. Zakończenie etapów transmisji sygnalizowane jest poprzez automatyczne zerowanie bitu TWINT, oznaczającego flagę przerwania. Transmisja danych odbywa się poprzez uprzednio załadowany rejestr danych TWDR oraz uruchomienie modułu TWI. Natomiast odbiór danych jest dodatkowo sparametryzowany poprzez możliwość zaprogramowania znaku potwierdzenia przez układ nadzorzący. Stan potwierdzenia sygnalizowany jest uprzednio zaprogramowanym bitem TWEA (TWI Enable Acknowledge). Po każdym etapie transmisji aktualizowany jest status interfejsu w rejestrze TWSR [11], informujący o wystąpieniu błędu arbitrażu, czy braku potwierdzenia.

Bit	7	6	5	4	3	2	1	0	TWCR
	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE	
Read/Write	R/W	R/W	R/W	R/W	R	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Rys 5.8. Rejestr kontrolny interfejsu TWI [11].

6 OPROGRAMOWANIE MIKROKONTROLERA

W rozdziale zostanie omówiony proces tworzenia oprogramowania na mikrokontroler ATmega32U4. Szczególną uwagę poświęcono na przedstawienie algorytmów programu, dzięki którym został stworzony kod. Zostaną omówione także najważniejsze funkcje pomiarowe, kalibracyjne i komunikacyjne.

6.1 Struktura programu

Program został napisany w języku C przy użyciu środowiska EclipseIDE z pluginem AVR [7]. Składa się z około 2500 linii kodu i zajmuje niemalże połowę pamięci programu mikrokontrolera. Wykorzystanie interfejsu USB stworzyło wymaganie pracy z wyłączonym układem sprawdzającym poprawność działania programu – WatchDog. Do obsługi interfejsu USB wykorzystano, rekomendowaną przez firmę Microchip, bibliotekę LUFA USB [18]. Biblioteka ta udostępnia szereg możliwości, np. zaprogramowanie deskryptorów, dzięki czemu interfejs użytkownika jest w stanie rozpoznać podłączone urządzenie w rejestrach komputera i nawiązać z nim komunikację. Do obsługi USB wymagany jest włączony system przerwań. Podczas podłączenia i konfiguracji urządzenia z komputerem wywoływane są określone funkcje, udostępnione dla użytkownika biblioteki LUFA. Dzięki temu zabiegowi następuje komunikacja stanu podłączenia modułu do komputera, za pomocą diod elektroluminescencyjnych umieszczonych na module Arduino Micro. Działanie interfejsu USB wymaga wywołania w pętli głównej funkcji opisanych w dokumentacji biblioteki. Natomiast odbiór danych obsługiwany jest poprzez wywołanie funkcji sprawdzającej ich dostępności w punktach końcowych USB.

6.2 Inicjalizacja urządzenia

Po rozpoczęcie programu, wszystkie porty I/O pracują w trybie wejścia. W pierwszej kolejności następuje ich inicjalizacja, wyłączony zostaje bufor trójstanowy, aby odizolować układ pomiarowy. Diody LED zostają wygaszone oraz konfigurowane są wewnętrzne podciągnięcia portów wejściowych do zasilania, przez wbudowane rezystory w strukturę krzemową mikrokontrolera. W ostatnim kroku, po konfiguracji stanów początkowych na portach wyjściowych, ustawiane są one w tryb wyjściowy, eliminując tym samym np. migotań diod elektroluminescencyjnych. Następnie zostaje sprawdzona obecność układów zewnętrznych, komunikujących się poprzez interfejs TWI. Po wykryciu modułu LCD generowany jest komunikat na wyświetlacz, informujący o działaniu programu.

Inicjalizacja USB (listing 6.1) opiera się o wyłączenie modułu WatchDog oraz wywołanie funkcji inicjalizacyjnej biblioteki LUFA [18]. Na potrzeby przesyłania danych zostaje utworzony strumień danych.

```

void Init_USB(void)
{
    /* USB initialization */
    cbi(MCUSR, WDRF);
    wdt_disable();
    clock_prescale_set(clock_div_1);
    USB_Init();
    CDC_Device_CreateStream(&VirtualSerial_CDC_Interface, &USB_Stream);
}

```

Listing 6.1. Funkcja inicjalizacji interfejsu USB.

Inicjalizacja modułu przechwytywania zdarzeń zewnętrznych opiera się na wyborze zbocza (listing 6.2), które ma aktywować przerwanie i zapamiętanie aktualnego stanu licznika 1. Działanie filtru szumów zostało sparametryzowane za pomocą dyrektyw preprocesora [8], co umożliwia szybką konfigurację i komplikację kodu, zarówno z wykorzystaniem lub nie, filtra szumów. Po inicjalizacji, licznik nie zlicza impulsów, dołączenie zegara taktującego licznik jest konfigurowane podczas pomiaru pojemności, w tym samym czasie aktywowane jest zezwolenie na obsługę przerwania.

```

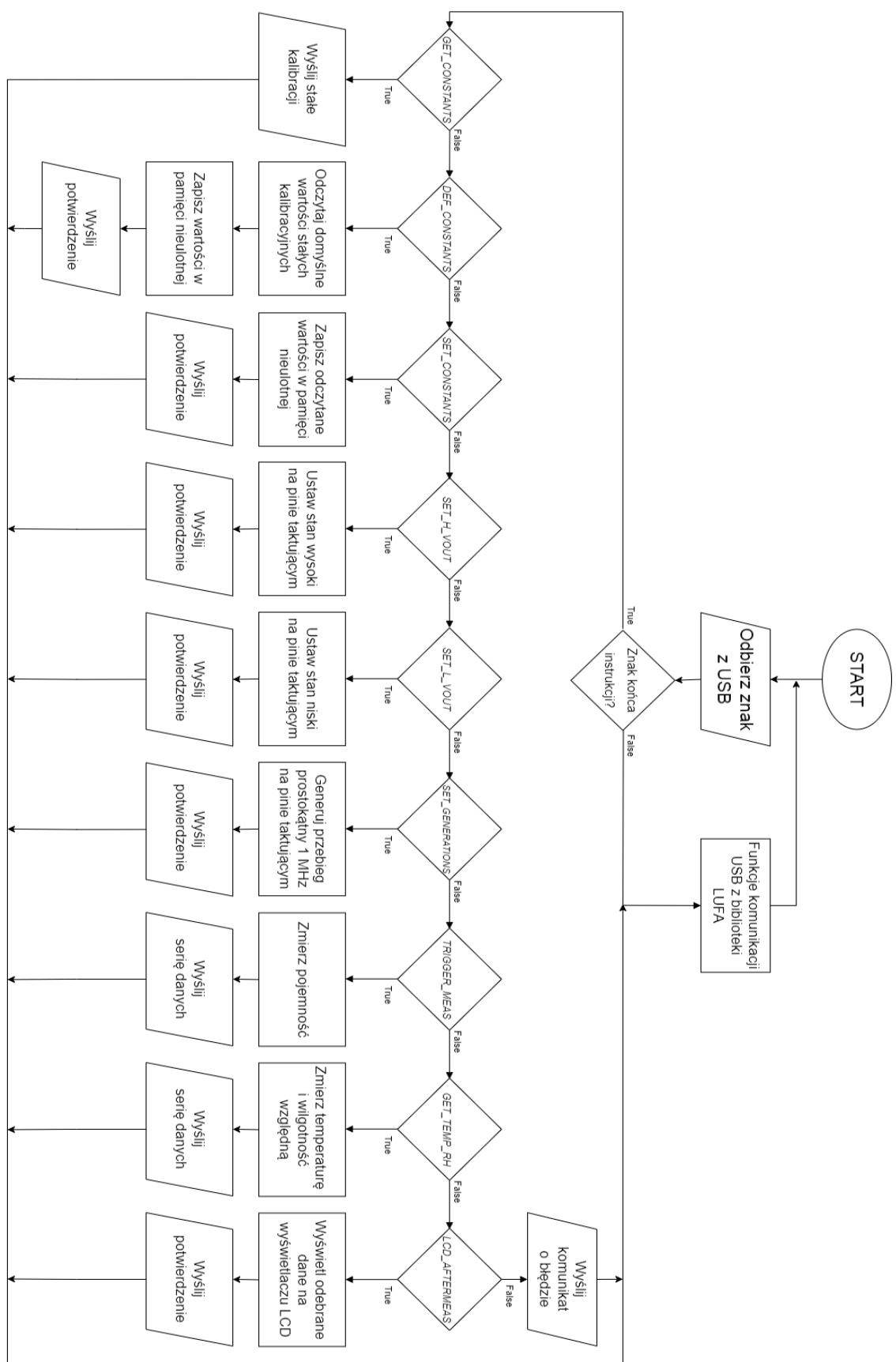
void Init_InputCapture(void)
{
    #if ENABLE_NOISE_CANCELER
        /* Input Capture Noise Canceler For ICP1 */
        sbi(TCCR1B, ICNC1);
    #endif
        /* Input Capture Edge Select - Rising Edge */
        sbi(TCCR1B, ICES1);
}

```

Listing 6.2. Funkcja inicjalizacji modułu przechwytywania .

6.3 Pętla główna programu

Algorytm pętli głównej (rys 6.1) oparty jest o interpretację instrukcji otrzymanych z interfejsu użytkownika, obróbkę danych i odesłanie komunikatu potwierdzenia, serii danych bądź informacji o błędzie. Wszelkie uzyskane błędy są ukazywane użytkownikowi w postaci pojawiającego się okna z informacją. Układ laboratoryjny obsługuje 10 odmiennych instrukcji, w tym osobny pomiar pojemności oraz temperatury i wilgotności względnej. Możliwość zaprogramowania serii pomiarów uzyskano bezpośrednio z interfejsu użytkownika, w którym program komputerowy co wprowadzony czas wywołuje żądanego pomiar od inteligentnego czujnika pojemnościowego. Z punktu widzenia układu laboratoryjnego jest to jednokrotny pomiar, dlatego przerwanie operacji nie wniesie błędów w działanie układu laboratoryjnego. Zastosowanie interfejsu USB wymusza na wywoływanie w pętli głównej programu funkcji z biblioteki LUFA. To ograniczenie powoduje brak możliwości uśpienia mikrokontrolera na czas bezczynności oraz wymusza programowanie w trybie MultiTaskingu (wielozadaniowości) [7].



Rys. 6.1. Algorytm pętli głównej programu mikrokontrolera.

6.4 Funkcje urządzenia

Układ laboratoryjny obsługuje 10 odrębnych instrukcji, które można podzielić na trzy obszary:

- transfer danych kalibracyjnych,
- manipulacja wyjściami, w celu kalibracji poziomów napięć oraz opóźnień,
- pomiar wielkości elektrycznych i fizycznych,

Po każdym wykonanym poleceniu, układ laboratoryjny generuje potwierdzenie wykonania instrukcji, bądź przesyła serię danych pomiarowych. Urządzenie umożliwia pomiar jedynie wartości temperatury i wilgotności względnej, z sensora cyfrowego oraz dodatkowy pomiar pojemności. Serie danych zostają zapamiętane w pamięci operacyjnej mikrokontrolera, po wykonaniu serii pomiarowych następuje formatowanie danych oraz przesyłanie do interfejsu użytkownika. Maksymalny czas pomiaru wynosi 500 ms, w przeciwnym razie urządzenie generuje komunikat błędu o przekroczeniu tej wartości.

6.4.1 Kalibracja

Obszar manipulacji wyjściami, w celu kalibracji poziomów napięć odpowiedzialny jest za obsługę trzech instrukcji (listing 6.3), ustawień obu poziomów logicznych na wyjściu bufora trójstanowego oraz generację przebiegu prostokątnego o częstotliwości 1 MHz, na wyprowadzeniu taktującym układ pomiarowy. Generowanie przebiegu odbywa się z wykorzystaniem wewnętrznego 8-bitowego licznika [8], dzięki czemu uzyskiwana na wyjściu częstotliwość jest dokładna.

```
case SET_H_VOUT:          /* Set Signal Pin as HIGH */
{
    DisableGenerations();
    SET_GEN_STATE;
    MEASURING_SYSTEM_ON;
    SEND_CONFIRMATION;
    break;
}
case SET_L_VOUT:          /* Set Signal Pin as LOW */
{
    DisableGenerations();
    CLR_GEN_STATE;
    MEASURING_SYSTEM_ON;
    SEND_CONFIRMATION;
    break;
}
case SET_GENERATIONS:    /* Set Generation on Signal Pin */
{
    EnableGenerations();
    MEASURING_SYSTEM_ON;
    SEND_CONFIRMATION;
    break;
}
```

Listing 6.3. Obsługa instrukcji kalibrujących.

Stałe wartości kalibracji, będącego podstawą do modelu matematycznego obwodu, zostały zdefiniowane jako struktury danych, które są przechowywane w nieulotnej pamięci EEPROM (listing 6.4). Po każdorazowej inicjalizacji systemu mikrokontroler pobiera je do pamięci operacyjnej [7]. Po interpretacji instrukcji żądania przesłania danych, są one formatowane i kierowane do interfejsu użytkownika. Podczas wyznaczania wartości pojemności interfejs użytkownika korzysta bezpośrednio z danych otrzymanych z układu laboratoryjnego. Użytkownik ma możliwość zmiany wartości każdego z zapisanych elementów. Potwierdzając zapis do urządzenia, interfejs generuje instrukcję żądania nadpisania stałych wartości kalibracji pamięci nieulotnej EEPROM.

```
const CONSTANTS_t EEMEM Constants_EEMEM =
{
    .H_THR    = 3.3500,
    .L_THR    = 1.6770,
    .H_VOUT   = 5.0230,
    .L_VOUT   = 0.0000,
    .R_MEAS   = 191100,
};

const CORRECTIONS_t EEMEM Corrections_EEMEM =
{
    .A0 = 3.2231,
    .A1 = -4.4837,
    .A2 = 2.7930,
    .A3 = -4.6443,
};
```

Listing 6.4. Definicja domyślnych wartości struktur danych kalibracyjnych.

6.4.2 Pomiar temperatury i wilgotności względnej

Temperatura i wilgotność względna jest mierzona za pomocą cyfrowego czujnika SHTC3. Komunikacja z czujnikiem odbywa się poprzez interfejs TWI. Zaletą jaka przeważyła na wyborze tego podzespołu jest krótki czas pomiaru (15 ms) przy zapewnieniu wystarczającej dokładności wskazań $\pm 2\%$ RH oraz $\pm 0.2^\circ\text{C}$ [13]. Dodatkowo element posiada małe wymiary, co czyni możliwym, umiejscowienie go blisko pojemnościowego sensora HS1101, tym samym lepiej odzwierciedli warunki panujące wokół badanego czujnika. Urządzenie charakteryzuje także niski pobór prądu podczas pomiaru, który wynosi 900 μA [13]. Sensor także pracuje na podwyższonej częstotliwości pracy interfejsu TWI (400 kHz).

Sam pomiar sprowadza się do inicjalizacji sensora, wysyłając instrukcje rozpoczętymią pomiar przez interfejs TWI. Po czasie 15 ms mikrokontroler wywołuje odczyt danych z cyfrowego czujnika SHTC3. Transfer wyznaczonych wartości temperatury i wilgotności poparty jest sumą kontrolną, która wewnętrz programu mikrokontrolera zostaje przeanalizowana. Operacja ta znacząco ogranicza możliwości błędnych wskazań mierzonych wartości.

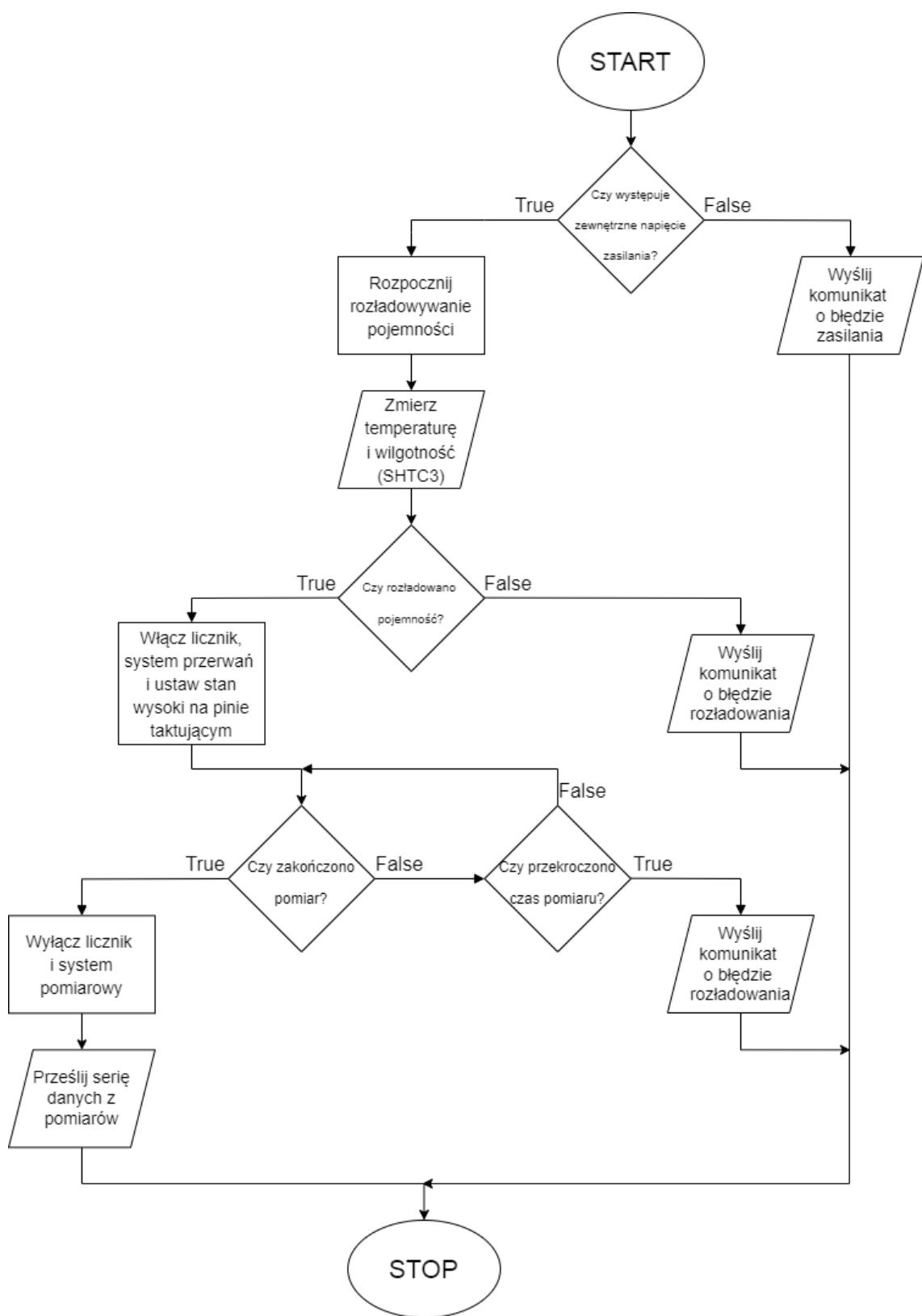
6.4.3 Pomiar pojemności

Najważniejszym celem realizacji projektu jest pomiar pojemności, który opiera się o jak najdokładniejsze zmierzenie czasów ładowania i rozładowania energii gromadzącej się w obiekcie badanym. Odebranie instrukcji aktywującej pomiar, w pierwszej kolejności sprawdza obecność zewnętrznego napięcia zasilania (rys. 6.2), ponieważ podłączenie urządzenia do portu USB spowoduje uruchomienie programu mikrokontrolera, dostarczając z portu zasilanie. W przypadku braku zewnętrznego zasilacza, układ laboratoryjny informuje o tym interfejs użytkownika, przerywając przy tym wykonywanie instrukcji. W następnym kroku, układ rozpoczyna rozładowywanie pojemności badanej, poprzez aktywację obwodu pomiarowego i wystawienie stanu niskiego na wyprowadzeniu taktującym. Rozpoczyna się pomiar temperatury i wilgotności względnej, z wykorzystaniem czujnika cyfrowego SHTC3.

Gdy poziom ładunku zgromadzonego przez badaną pojemność osiągnie wartość napięcia poniżej dolnego progu referencyjnego, dojdzie doinicjalizacji modułu przechwytywania zdarzeń zewnętrznych oraz systemu przerwań. Program główny (listing 6.5) oczekuje na ustawienie flagi zakończenia serii pomiarowej przez przerwanie. W pętli sprawdzającej zakończenie serii pomiarowej, następuje sprawdzenie okresu jego trwania. Przekroczenie progu powoduje anulowanie dotychczasowych wyników oraz wygenerowanie komunikatu do interfejsu użytkownika, informującego o nieprawidłowościach na drodze serii pomiarowej.

Obsługa przerwania (listing 6.6) sprowadza się do zapamiętania w buforze danych, wartości rejestru (ICR1) przechwyconego przez moduł Input Capture [8]. Wartość obecnego stanu licznika 1 zostaje wyzerowana. Powoduje to, że podczas transferu danych do interfejsu użytkownika istnieje możliwość interpretacji liczbowej kolejnych wartości bez konieczności wykonywania obliczeń matematycznych. W kolejnej instrukcji następuje sprawdzenie liczby zarejestrowanych pomiarów. Przekroczenie granicznej wartości powoduje ustawienie flagi zakończenia, którą program główny zinterpretuje, zaraz po wyjściu z przerwania. Przed zakończeniem obsługi przerwania następuje także zmiana stanu na wyjściu bufora trójstanowego, taktującego obwód pomiarowy, aby odwrócić stan ładowania / rozładowania badanego obiektu.

Zakończenie pomiarów, zinterpretowane przez program główny, w pierwszej kolejności deaktywuje system przerwań oraz moduł przechwytywania zdarzeń zewnętrznych. Obwód pomiarowy zostaje odłączony od zasilania, poprzez przejście bufora trójstanowego w stan wysokiej impedancji. Ostatecznie następuje formatowanie danych i ich transfer do interfejsu użytkownika.



Rys. 6.2. Algorytm prezentujący pomiar pojemności.

```

/* Capacity Measuring Function */
STATUS_t CapacityMeasurement(void)
{
    /* Check Power Status */
    if (!PWR_STATUS)
    {
        fprintf_P(&USB_Stream, PSTR("Error: Check the power supply of the module.\r\n"));
        return Status_PowerError;
    }
    /* Discharge Capacity */
    CLR_GEN_STATE;
    MEASURING_SYSTEM_ON;
    /* Temperature and Humidity Measurement */
    STATUS_t Status = TemperatureMeasurement();
    /* Wait for LOW State on V_Cap Pin */
    while (!V_CAP_IS_UNDER_L THR)
    {
        [...]
    }
    /* Measurement conditions ready - capacity discharged, power supply checked */
    /* Set sample index to 0 */
    System.SampleIdx = 0;
    System.IsMeasurementEnd = false;
    EnableInputCapture();
    SET_GEN_STATE;
    /* Waiting for end measurement */
    cnt = 0;
    while (!System.IsMeasurementEnd)
    {
        [...]
    }
    /* Measurement End - Disable measuring system */
    DisableInputCapture();
    CLR_GEN_STATE;
    MEASURING_SYSTEM_OFF;
    /* Data presentation */
    [...]
}

```

Listing 6.5. Funkcja pomiaru pojemności.

Pomimo wysokiej optymalizacji kompilatora, wygenerowany kod języka assemblera, posiada szereg zbędnych instrukcji wewnętrz przerwania. Przewagą oprogramowania niskopoziomowego, w tym przypadku, okazałoby się znaczne skrócenie kodu, a co za tym idzie, czasu opóźnienia jakie wprowadza. Będzie to celem rozwojowym prac nad projektem. Obecne zajmowana pamięć przerwania to 100 bajtów, czas opóźnienia jakie generuje jego wykonanie wynosi około 6 µs.

```

/* Input Capture Interrupt */
ISR(TIMER1_CAPT_vect, ISR_BLOCK)
{
    TCNT1 = 0;
    System.Samples[System.SampleIdx++] = ICR1;
    if (System.SampleIdx == MAX_SAMPLES_BUF_SIZE)
        System.IsMeasurementEnd = true;
    TOG_GEN_STATE;
}

```

Listing 6.6. Obsługa przerwania modułu przechwytywania zdarzeń zewnętrznych.

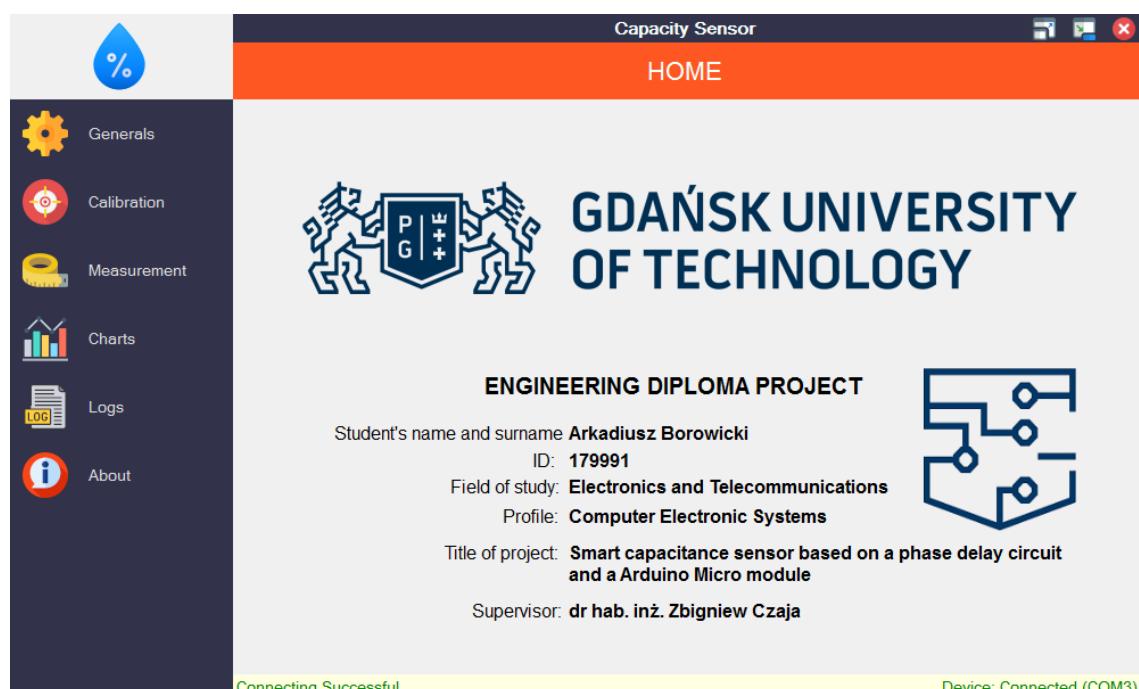
7 INTERFEJS UŻYTKOWNIKA NA KOMPUTERZE PC

Interfejs użytkownika jest jednym z najważniejszych elementów projektu, zapewnia komunikację z układem laboratoryjnym, wykonuje obliczenia i prezentuje dane. W rozdziale zostanie zaprezentowana realizacja interfejsu użytkownika. Szczegółową uwagę poświęcono na omówienie funkcji matematycznych, korekcji charakterystyki oraz generacji skryptów Matlab.

Interfejs użytkownika został zrealizowany w języku C# z biblioteką WindowsForms [9]. Kod składa się z około 3500 linii. Program zapewnia komunikację z urządzeniem laboratoryjnym, wykresla charakterystyki przejściowe z zaprogramowanych serii pomiarowych, generując skrypt Matlab pozwalający na obróbkę danych przez użytkownika systemu. Prowadzony jest także szczegółowy dziennik zdarzeń, umożliwiający zapoznanie się użytkownikowi z działaniem systemu.

7.1 Strona startowa

Po uruchomieniu interfejsu użytkownika (rys. 7.1) ukazują się najważniejsze informacje dotyczące osób odpowiedzialnych za projekt. W dolnej części aplikacji okienkowej widnieje informacja o statusie połączenia z układem laboratoryjnym, pasek informacyjny jest widoczny z poziomu każdego okna. Prócz statusu widnieje także informacja o ostatnim zdarzeniu w systemie. Z poziomu okna istnieje możliwość przejścia do ustawień ogólnych, kalibracji czy pomiarów, które dzięki przyjaznemu interfejsowi są intuicyjne podczas korzystania z systemu.



Rys. 7.1. Strona startowa interfejsu użytkownika.

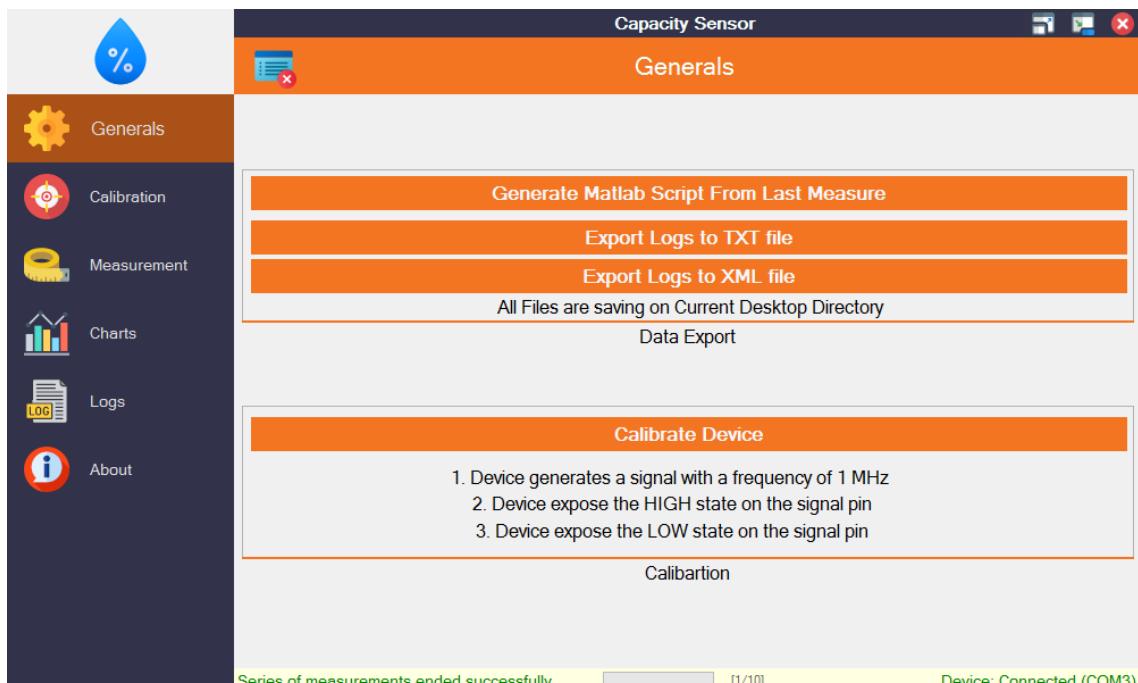
7.2 Ustawienia ogólne

Z poziomu okna ustawień ogólnych (rys. 7.2) istnieje możliwość kalibracji urządzenia, która realizowana jest w trzech krokach,

- na wyprowadzeniu taktującym obwód RC, generowany jest przebieg prostokątny o częstotliwości 1 MHz,
- wyprowadzenie taktujące obwód pomiarowy przyjmuje stan wysoki,
- wyprowadzenie taktujące obwód pomiarowy przyjmuje stan niski.

Zastosowanie trzech kroków pozwala na zmierzenie poziomów napięć wyjściowych z bufora trójstanowego w obu stanach, które może się zmieniać np. pod wpływem zmian temperatury otoczenia. Zmierzone wartości następnie należy wprowadzić do interfejsu użytkownika w oknie kalibracji. Natomiast przy pomocy generowanego sygnału 1 MHz, możliwe jest podejrzenie przebiegu sondą oscyloskopową i określenie opóźnień wprowadzanych przez układ. Pozwala to także na oszacowanie pojemności pasożytniczych obecnych w układzie, które będą wpływały na rozmywanie się zbocz sygnału generowanego.

Oprócz kalibracji, interfejs umożliwia wyeksportowanie danych pomiarowych w postaci skryptu Matlab oraz eksportowanie dziennika zdarzeń w postaci pliku tekstowego lub XML [9]. Utworzonye pliki zostają zapisane na pulpicie komputera i są otwierane przez domyślnie wybrany program, zaraz po ich zapisaniu.



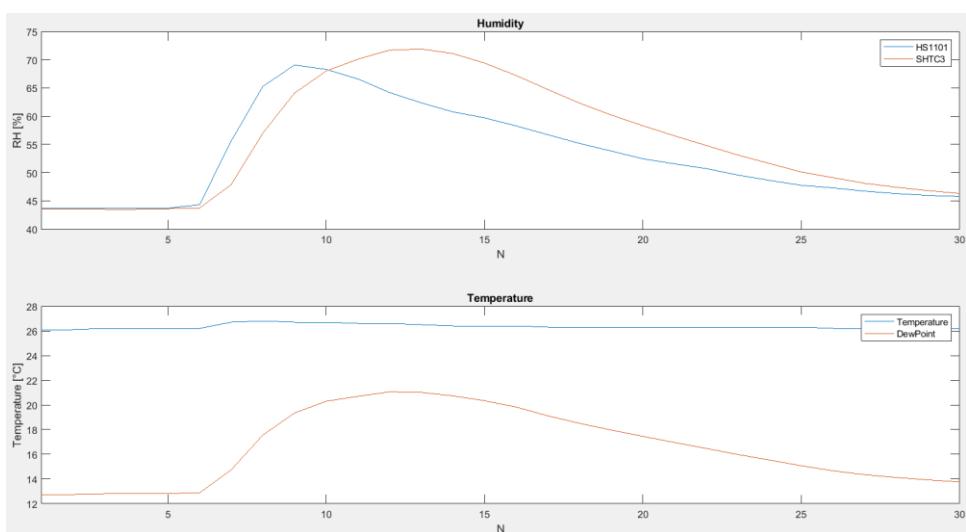
Rys. 7.2. Ustawienia ogólne interfejsu użytkownika.

7.2.1 Eksport wyników pomiarowych do programu Matlab

Skrypt Matlab generuje charakterystyki czasu ładowania oraz rozładowania, w zależności od pojemności z zakresu od 100 do 300 pF, ze stałą wartością rezystancji pomiarowej. Zobrazowana została także charakterystyka czasu ładowania i rozładowania pojemności badanej dla wartości rezystancji z zakresu od 100 kΩ do 1 MΩ. Wygenerowane charakterystyki pozwalają zobrazować sensowność pomiaru przy podanych wartościach rezystancji oraz mierzonej pojemności, ponieważ zbyt mała rezystancja będzie prowadziła do krótkich czasów pomiarowych, co będzie skutkowało nadmiarowym ładowaniem pojemności ponad skonfigurowane napięcie progowe, to zjawisko wywołane jest opóźnieniem obsługi przerwania.

Skrypt wyznacza histogram pomiarów podczas wykonywanych serii pomiarowych. Odpowiednia duża liczba wyników powinna tworzyć rozkład Gaussa, co zaprezentowano w dalszej części pracy. Histogram pozwala statystycznie ocenić użytkownikowi czy pomiar został wykonany z należytą precyzją, ocenić to można poprzez szerokość rozkładu normalnego. Zostają wyznaczone także charakterystyki błędów losowych dla czasów ładowania i rozładowania badanej pojemności, które informują o wystąpieniu nieprzewidzianego zakłócenia spowodowanego np. źródłem zewnętrznym. Powtarzany błąd losowy przy każdym pomiarze informuje użytkownika systemu o wystąpieniu błędu systematycznego, który występuje z powodu niedoskonałości układu laboratoryjnego.

Wyznaczony także zostaje charakterystyka wilgotności względnej, zmierzonej za pomocą czujnika pojemnościowego HS1101 oraz cyfrowego SHTC3 (rys. 7.3). Użytkownik ma możliwość określenia poziomu kalibracji układu laboratoryjnego oraz doświadczalnego sprawdzenia bezwładności i pojemności cieplnej obu podzespołów.



Rys. 7.3. Przykład charakterystyki wilgotności i temperatury w czasie.

7.2.2 Eksport dziennika zdarzeń

W przypadku dużej liczby pomiarów, wygenerowany plik XML umożliwia prezentację danych ze wszystkich wykonanych pomiarów w ciągu całej sesji. Na przykład generując dokument XSLT przetwarzający plik XML (listing 7.1), można ukazać sformatowane dane w przeglądarce internetowej. Obecne przeglądarki interpretują instrukcję języka JavaScript, umożliwiając nie ograniczone prezentowanie danych, w przejrzysty dla użytkownika sposób. Funkcja nie została zaimplementowana na potrzeby tego projektu lecz będzie jednym z celów jego rozwoju.

Format tekstowy XML nie jest przejrzysty dla użytkownika, dlatego zdecydowano również na eksport danych w formacie tekstowym, aby umożliwić ewentualną kopię danych pomiarowych z całej sesji.

```
<CapacitySensorLogs DateGenerated="2022-11-06 20:13:31">
  <Log Date="2022-11-06 19:13:09.401" From="User" Type="Info">The application was opened</Log>
  <Log Date="2022-11-06 19:13:10.525" From="Device" Type="Info">Device Connection Success - COM3</Log>
  <Log Date="2022-11-06 19:13:10.612" From="PC" Type="Info">Sent Command: GET_CONSTANTS (A)</Log>
  <Log Date="2022-11-06 19:13:10.615" From="Device" Type="Info">Data Received: 3.3500 1.6770 5.0230 0.0000 741200</Log>
  <Log Date="2022-11-06 19:13:10.703" From="PC" Type="Info">Sent Command: GET_CORRECTIONS (I)</Log>
  <Log Date="2022-11-06 19:13:10.706" From="Device" Type="Info">Data Received: 3.2231 -4.4837 2.7930 -4.6443</Log>
  <Log Date="2022-11-06 19:13:20.048" From="PC" Type="Info">Series start of full measurements</Log>
  <Log Date="2022-11-06 19:13:20.172" From="PC" Type="Info">Sent Command: TRIGGER_MEAS (M)</Log>
  <Log Date="2022-11-06 19:13:20.221" From="Device" Type="Info">Data Received: T 24.7 H 46.1 C 2605 1638 1644 1640...</Log>
  <Log Date="2022-11-06 19:13:21.164" From="PC" Type="Info">Sent Command: TRIGGER_MEAS (M)</Log>
  <Log Date="2022-11-06 19:13:21.213" From="Device" Type="Info">Data Received: T 24.7 H 46.0 C 2669 1637 1643 1638...</Log>
  <Log Date="2022-11-06 19:13:22.164" From="PC" Type="Info">Sent Command: TRIGGER_MEAS (M)</Log>
  <Log Date="2022-11-06 19:13:22.213" From="Device" Type="Info">Data Received: T 24.7 H 45.9 C 2695 1637 1643 1638...</Log>
  <Log Date="2022-11-06 19:13:23.176" From="PC" Type="Info">Sent Command: TRIGGER_MEAS (M)</Log>
  <Log Date="2022-11-06 19:13:23.224" From="Device" Type="Info">Data Received: T 24.7 H 45.8 C 2704 1638 1643 1638...</Log>
  <Log Date="2022-11-06 19:13:24.188" From="PC" Type="Info">Sent Command: TRIGGER_MEAS (M)</Log>
  <Log Date="2022-11-06 19:13:24.237" From="Device" Type="Info">Data Received: T 24.7 H 45.8 C 2758 1638 1644 1639...</Log>
  <Log Date="2022-11-06 19:13:25.069" From="PC" Type="Info">Series of measurements ended successfully</Log>
  <Log Date="2022-11-06 19:13:31.253" From="User" Type="Info">Log XML Document Generated - Desktop\Logs-201331.xml</Log>
</CapacitySensorLogs>
```

Listing 7.1. Przykład eksportu dziennika zdarzeń w formacie XML.

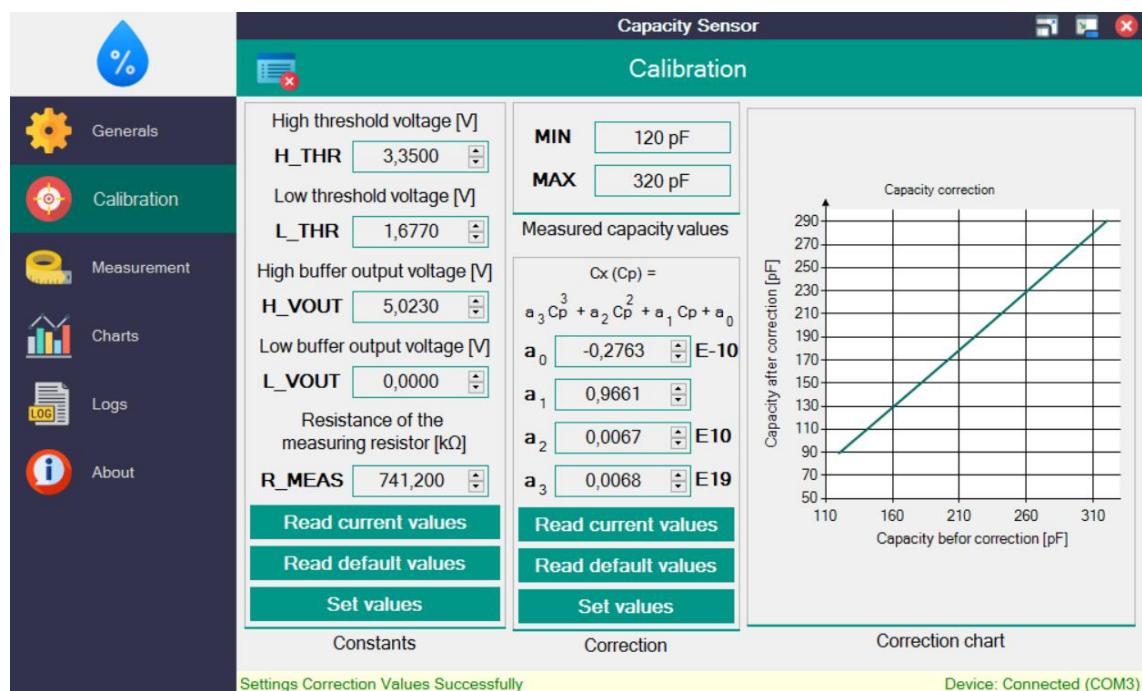
7.3 Kalibracja

Kalibracja urządzenia w przypadku pomiaru wielkości rzędu pF jest jednym z kluczowych elementów wpływających na dokładność pomiarową. Zmiana temperatury czy wilgotności bezpośrednio wpływa na stabilizowane napięcie układu 7805, czy napięcie wyjściowe z bufora trójstanowego. Również zmiana otoczenia, w którym występują zakłócenia elektromagnetyczne będzie wpływała na dokładność pomiarów, np. zaindukowane napięcie na linii połączonej z badaną pojemnością może zostać zinterpretowane przez komparator jako wystąpienie zdarzenia, co w efekcie będzie powodowało błędne wywołanie przerwania. Zrealizowany model układu laboratoryjnego nie został izolowany od takich zjawisk. Wszelkie wartości możliwe do zmiany ustawień są zapisywane w pamięci nieulotnej EEPROM mikrokontrolera znajdującego się w układzie laboratoryjnym. Pamiętanie ustawień jest pomocne z punktu widzenia użytkownika, raz stworzona konfiguracja, dla konkretnego urządzenia pomiarowego, zostanie zapamiętana po wyłączeniu zasilania i odtworzona po inicjalizacji systemu, co nie zwalnia z każdorazowej kalibracji urządzenia.

Wartości kalibracji są źródłem danych dla obliczeń matematycznych. Dane kalibracyjne, określone za pomocą wzoru 3.4 stanowią:

- napięcia referencyjne (H_THR, L_THR),
- napięcia wyjściowe bufora w stanie niskim i wysokim (H_VOUT, L_VOUT),
- rezystancję rezystora pomiarowego (R_MEAS),
- współczynniki korekcji charakterystyki pojemnościowej ($a_0 - a_3$).

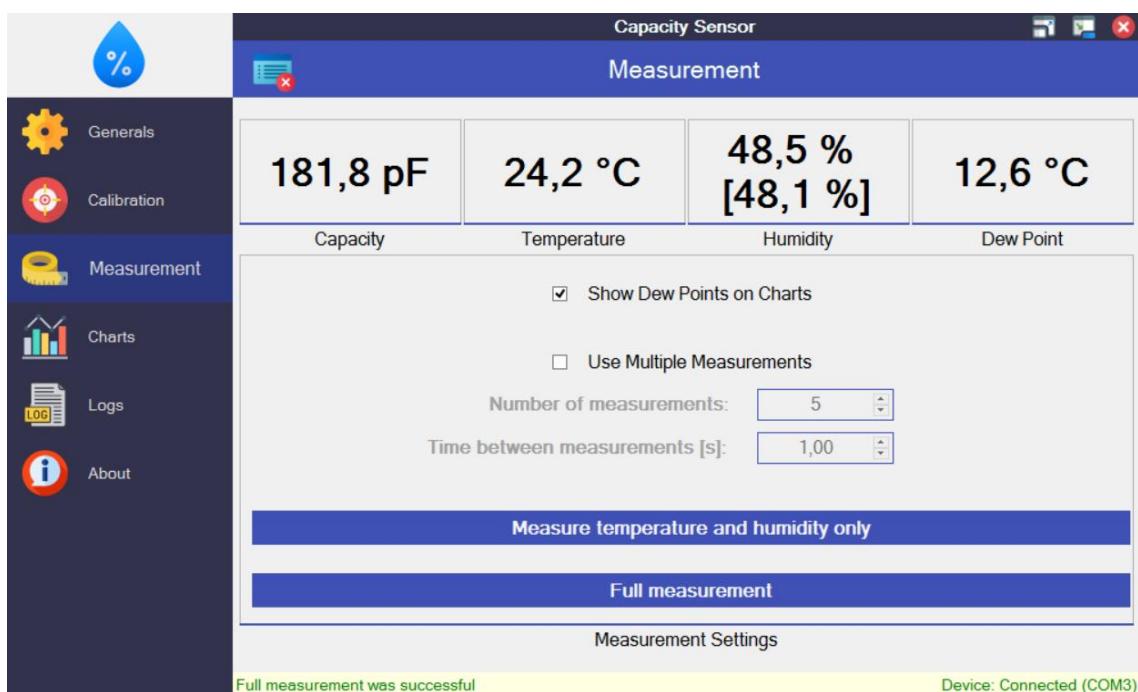
Dodatkowo z ustawień współczynników korekcji generowany jest wykres, którego celem jest zobrazowanie jakie wartości przyjmą pojemności zmierzone z zakresu 120 – 320 pF.



Rys 7.4. Kalibracja urządzenia z poziomu interfejsu użytkownika.

7.4 Funkcje pomiarowe

Pomiar pojemności może odbywać się w jednym z dwóch trybów (rys 7.5) – pojedynczego pomiaru oraz serii pomiarowej, dla której generowany jest wykres. Wielkość wykresu zależy od ilość punktów pomiarowych, które użytkownik systemu definiuje. Sparametryzowany został także odstęp czasowy pomiędzy pomiarami. Minimalną wartość odstępu do zarejestrowania pomiaru wynosi 500 ms. Dodatkowo w oknie widnieją wyniki dotyczące ostatniego pomiaru – wyświetlana jest zmierzona pojemność czujnika, temperatura otoczenia, temperatura punktu rosy oraz wskazania wilgotności względnej przez czujnik SHTC3 oraz wyznaczone drogą obliczeń na podstawie znanej pojemności czujnika. Ta druga znajduje się w nawiasie kwadratowym.



Rys 7.5. Wywołanie pomiarów z poziomu interfejsu użytkownika.

Wyznaczenie wartości pojemności badanej z danych pomiarowych (listing 7.2) sprowadza się do podzielenia kolejnych próbek czasu na pochodzące od ładowania oraz od rozładowania. W następnej kolejności na obu zbiorach wprowadza się nad próbkiowanie i decymacje, które przebiegają na danych reprezentujących liczbę taktów zegara. Przejście na dziedzinę czasu realizowane jest mnożeniem przez okres jednego cyklu zegarowego. Otrzymane dane pozwalają w następnym kroku na obliczenie wartości pojemności. Pojemność wynikowa otrzymywana jest poprzez korekcję charakterystyki, średniej pojemności, otrzymanych za przy użyciu czasów ładowania i rozładowania.

```
public static void ParseC(string Received)
{
    [...]
    var DP_Capacity = Tick * Oversampling(DischargingProbes);
    var CP_Capacity = Tick * Oversampling(ChargingProbes);

    DP_Capacity = Capacity(DP_Capacity, Calibration.R_MEAS, Calibration.J,
                           Calibration.L_THR, Calibration.H_THR, Calibration.L_VOUT) * 1E3;
    CP_Capacity = Capacity(CP_Capacity, Calibration.R_MEAS, Calibration.J,
                           Calibration.H_THR, Calibration.L_VOUT, Calibration.H_VOUT) * 1E3;

    var CapacityCorr = Correction((DP_Capacity + CP_Capacity) / 2.0);
    var CapacityRound = double.Parse(string.Format("{0:0.0}", CapacityCorr));

    MainForm.Instance.C = CapacityRound;
    MainForm.Instance.RH = CalcHumidity(CapacityRound);
    [...]
}
```

Listing 7.2. Funkcja wyznaczająca pojemność z danych pomiarowych.

W celu poprawy dokładności pomiarowej zostaje zwiększa reprezentacja bitowa wyników pomiarowych. Aby uzyskać o jeden bit większą reprezentację liczbową należy próbkować sygnał czterokrotnie. Zwiększenie o N bitów wymaga dostarczenia algorytmowi 4^N próbek sygnału (listing 7.3). Wewnątrz programu mikrokontrolera zdefiniowano liczbę pomiarów na 128, tzn. 64 próbki ładowania oraz 64 rozładowania. Wykonanie takiej serii pozwala na zwiększenie reprezentacji liczbowej o 3 bity. Zatem zamiast reprezentacji próbki w zakresie 16 bitów (0 – 65535), uzyskano 19 bitów (0 – 524287). Kolejną z zalet operacji nadpróbkowania jest uśrednianie serii wyników pomiarowych. Znacząco zmniejsza to wartość błędów losowych. Sam algorytm nadpróbkowania i decymacji oparty jest o wyznaczenie sumy wartości próbek pomiarowych i następnym przesunięciu bitowym wyniku, o liczbę zwiększych bitów.

```
public static double Oversampling(List<int> Probes)
{
    int bits = 3;
    int N = (int)Math.Pow(4, bits);
    long sum = 0;
    for (int i = Probes.Count - N; i < Probes.Count; i++)
        sum += Probes[i];
    double result = sum >> bits;
    return result * Math.Pow(2, -bits);
}
```

Listing 7.3. Funkcja realizująca nadpróbkowanie i decymację.

Jedna z głównych funkcji pomiarowych dotyczy wyznaczenia wartości pojemności badanej, na podstawie otrzymanych czasów ładowania i rozładowania (listing 7.4). Funkcja realizuje wzór 3.4 – uwzględniając jedynie idealny obwód pomiarowy, dlatego konieczna jest późniejsza korekcja charakterystyki wielomianem 3. stopnia. Metoda ta zawiera listę parametrów, będących podstawą kalibracji urządzenia. Wyznaczanie wartości pojemności w zależności od czasu ładowania lub rozładowania, różni się podstawieniem parametrów napięć: początkowego, końcowego oraz wyjściowego z bufora trójstanowego. Wartość pojemności zostaje obliczona na podstawie nadpróbkowanych serii pomiarowych czasów ładowania i rozładowania badanego obiektu. Współczynniki wielomianu korygującego, zostały wyznaczone doświadczalnie, na podstawie pomiaru znanych, wzorcowych wartości pojemności.

```
public static double Capacity(double T, double RM, double JC,
    double VCapStop, double VCapStart, double VOut)
{
    return -T / RM / Math.Log((VCapStop - VOut + JC * RM) /
        (VCapStart - VOut + JC * RM));
}
```

Listing 7.4. Funkcja obliczająca pojemność.

Producent HS1101 określa, że charakterystyka pojemności w zależności od wilgotności względnej, nie przebiega w sposób liniowy [14]. Wzór konwersji został zasięgnięty wprost z dokumentacji i stanowi odwrotny wielomian odpowiedzi czujnika pojemnościowego na zmiany wilgotności względnej otoczenia. Wzór 7.1 zostaje zrealizowany za pomocą funkcji, wewnątrz interfejsu użytkownika (listing 7.5).

$$RH = -3465,5 \cdot X^3 + 10732 \cdot X^2 - 10457 \cdot X + 3245,9 \quad (7.1)$$

$$X = \frac{C_{zmierzona}}{C_{typowe}} \quad (7.2)$$

gdzie:

- RH – wilgotność względna [%],
- $C_{zmierzona}$ – zmierzona wartość pojemności [pF],
- C_{typowe} – typowa wartość pojemności czujnika dla RH = 55% – 180 pF [14].

```
public static double CalcHumidity(double HS1101_Capacity)
{
    double X = HS1101_Capacity / 180.0;
    double RH = -3465.5 * Math.Pow(X, 3) +
                10732.0 * Math.Pow(X, 2) - 10457.0 * X + 3245.9;
    return RH;
}
```

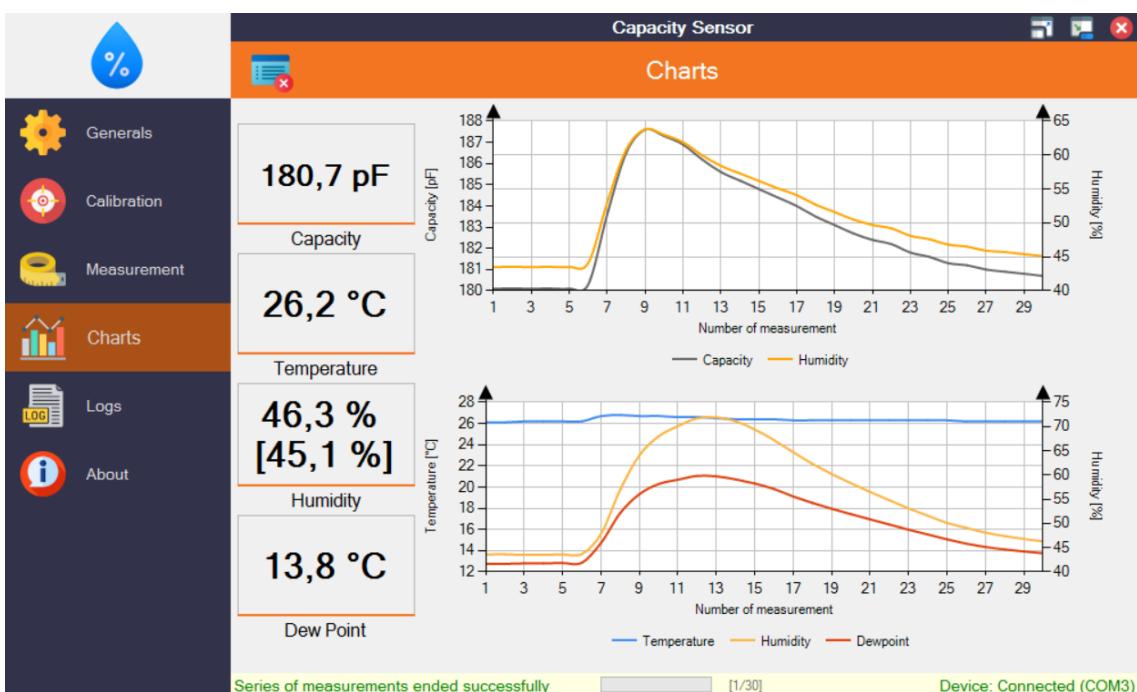
Listing 7.5. Funkcja obliczająca wilgotność względную na podstawie pojemności czujnika.

7.5 Charakterystyki pomiarów

Prezentacja danych w czasie rzeczywistym jest ogromną zaletą systemu. Możliwość zaprogramowania serii pomiarów, co zadany, ścisłe określony czas pozwala na rejestrację temperatury i wilgotności względnej otoczenia, w długim okresie, bez potrzeby ciągłego wyzwalania pomiaru przez użytkownika. Interfejs użytkownika rejestruje (rys 7.6):

- zmierzona pojemność,
- wilgotność względną, na podstawie pojemności czujnika HS1101,
- temperaturę i wilgotność względną otoczenia, z wykorzystaniem sensora SHTC3,
- temperaturę punktu rosy, wyznaczoną na podstawie temperatury i wilgotności.

Ponadto, wygenerowany skrypt Matlab, umożliwia dodatkową prezentację histogramu wskazań pojemności, w czasie pomiaru oraz charakterystykę błędów losowych. Użytkownik systemu posiada również możliwość ukrycia charakterystyki temperatury punktu rosy, dzięki czemu wykres temperatury otoczenia staje się czytelniejszy.



Rys. 7.6. Podgląd charakterystyk w interfejsie użytkownika.

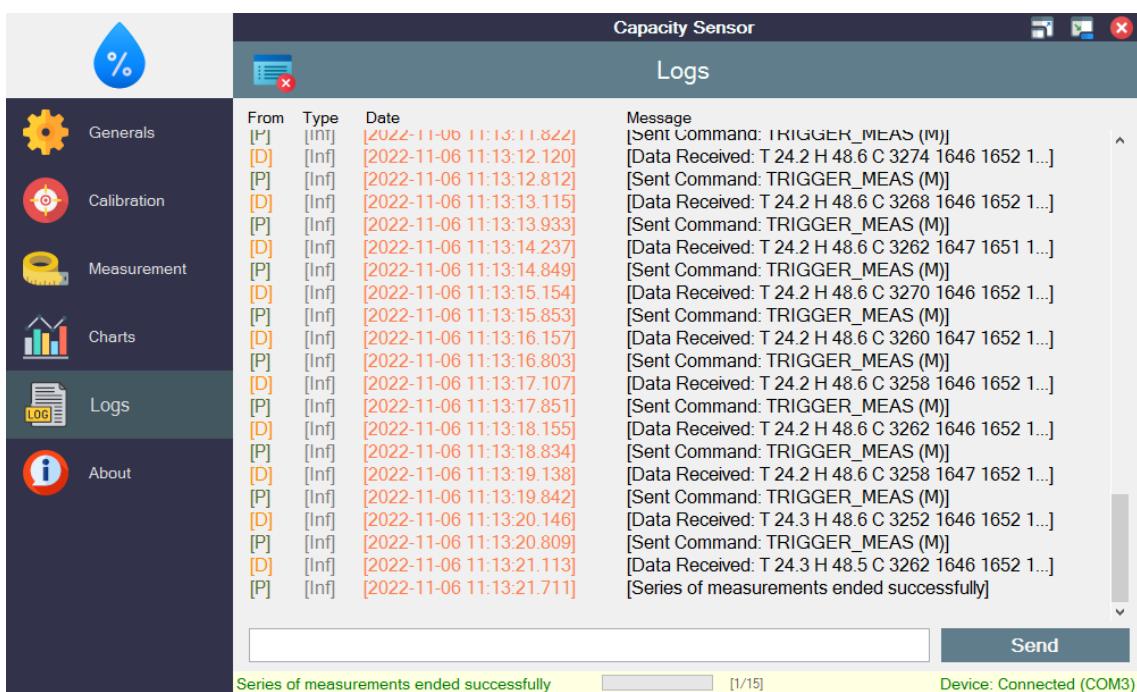
Wygenerowane charakterystyki przejściowe pozwalają porównać pojemność właściwą oraz czas odpowiedzi wykorzystanych sensorów. Zachowanie obu czujników przy powolnych zmianach wilgotności prowadzi do zbliżonych rezultatów wartości wilgotności. Nagłe zmiany, spowodowane np. podmuchem wilgotnego powietrza, wyłonią różnicę polegającą na różnych właściwościach cieplnych, pojemnościach własnych, stałych czasowych odpowiedzi, wielkości obudowy itp.

7.6 Dziennik zdarzeń

Ważnym elementem diagnostycznym działania systemu jest rejestracja zdarzeń, zarówno z punktu widzenia programistycznego jak i użytkowego. Natychmiastowe informacje o błędach oraz wewnętrznych zdarzeniach systemu, pojawiające się na pasku statusowym, w dolnej części okna pozwalają na ocenę działania systemu, bez konieczności przełączania widoków interfejsu.

Dziennik zdarzeń (rys. 7.7) informuje użytkownika systemu o sytuacjach, jakie zaszły w czasie sesji pomiarowej. Interfejs zachowuje czas wystąpienia wydarzeń, z dokładnością do pojedynczych ms. Zapisywany jest również jego rodzaj i pochodzenie. W interfejsie odróżnienie błędów sygnalizowane jest kolorystycznie.

Z poziomu ustawień ogólnych istnieje możliwość wyeksportowania danych w formacie tekstowym jak i XML. Z poziomu dziennika zdarzeń użytkownik może wynieść informacje na temat obsługiwanych instrukcji.



Rys 7.7. Dziennik zdarzeń interfejsu użytkownika.

Dodatkową, zaimplementowaną cechą jest terminal do urządzenia. Użytkownik z poziomu interfejsu ma możliwość wysłania dowolnej instrukcji (listing 7.6). Jest to dydaktyczny element, pozwalający nowym użytkownikom systemu na zapoznanie się z działaniem układu laboratoryjnego. Terminal pozwala jedynie na przesyłanie danych pomiędzy urządzeniami, zatem nie ma możliwości np. wywołania funkcji pomiaru pojemności, czy generacji skryptu Matlab.

```

typedef enum
{
    GET_CONSTANTS      = 'A',           // Get Constants Values
    SET_CONSTANTS      = 'S',           // Set Constants Values
    DEF_CONSTANTS      = 'D',           // Set Default Constants Values
    GET_CORRECTIONS    = 'I',           // Get Corrections Values
    SET_CORRECTIONS    = 'O',           // Set Corrections Values
    DEF_CORRECTIONS    = 'P',           // Set Default Corrections Values
    SET_L_VOUT          = 'L',           // Set Signal Pin as LOW
    SET_H_VOUT          = 'H',           // Set Signal Pin as HIGH
    SET_GENERATIONS    = 'G',           // Set Generations on Signal Pin
    SET_NOMINAL         = 'N',           // Set Signal Pin as NOMINAL (HIGH-Z)
    TRIGGER_MEAS        = 'M',           // Trigger Measurement
    GET_TEMP_RH          = 'R',           // Read Temperature and Humidity
    LCD_AFTERMEAS       = 'W',           // Send Calculated Data To Display on LCD
} COMMAND_t;

```

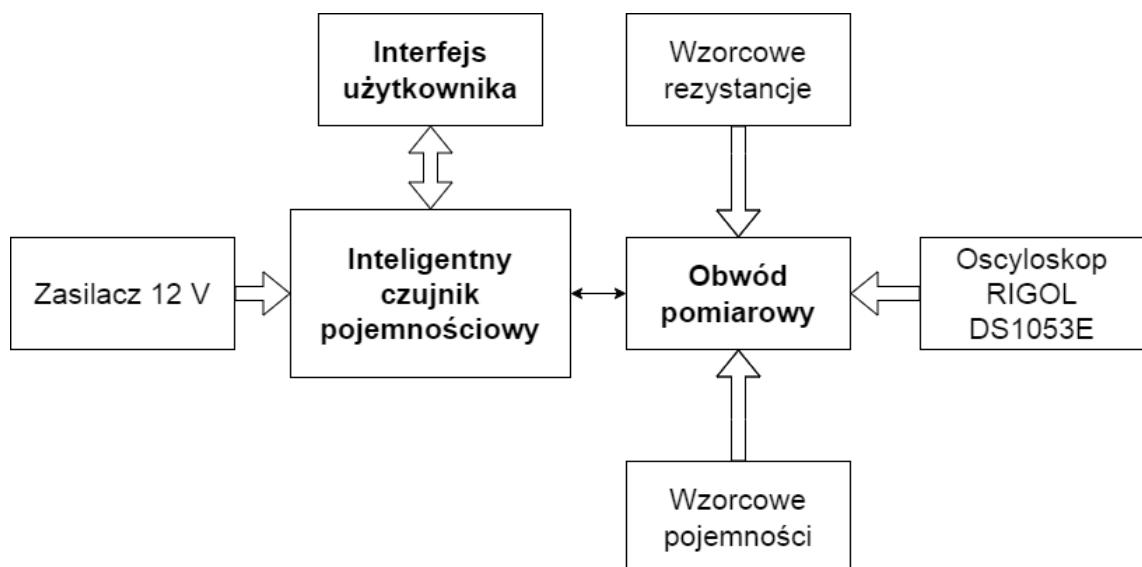
Listing 7.6. Zestaw instrukcji interpretowanych przez układ laboratoryjny.

8 WERYFIKACJA EKSPERYMENTALNA

W rozdziale zostanie zaprezentowany proces kalibracji urządzenia pomiarowego. Zostaną wyznaczone charakterystyki rzeczywiste układu pomiarowego. Szczególną uwagę poświęcono na omówieniu błędów, wynikających z nieidealności obwodu pomiarowego oraz wyznaczeniu charakterystyki korygującej pomiar pojemności.

8.1 Stanowisko pomiarowe

Stanowisko pomiarowe (rys. 8.1) opiera się na pomiarze charakterystyk rzeczywistego układu laboratoryjnego. W tym celu przeprowadzono szereg pomiarów z wykorzystaniem pojemności wzorcowych oraz rezystancji. Uzyskane wyniki pozwoliły na korekcję charakterystyki, z dokładnie wyznaczonymi współczynnikami. Porównano współczynniki otrzymane drogą regresji liniowej oraz interpolacji Lagrange'a 4 stopnia.



Rys. 8.1. Diagram stanowiska pomiarowego.

Zastosowanie laboratoryjnego oscyloskopu RIGO DS1053E pozwala zaobserwować charakterystykę przejściową podczas przeprowadzania badania. Pomaga doświadczalnie określić opóźnienie, jakie jest wprowadzane przez system przerwań mikrokontrolera. Podłączenie sondy oscyloskopowej zachodzi z wykorzystaniem złącz diagnostycznych. Podczas tego procesu warto pamiętać o zastosowaniu trybu „x10” sondy oscyloskopowej, co znacząco zmniejszy pojemności wprowadzane do obwodu pomiarowego.

Do poprawnego działania systemu pomiarowego, konieczne jest także wykorzystanie interfejsu użytkownika, komunikującego się z układem laboratoryjnym, oraz zewnętrznego napięcia zasilania, które zapewnia stabilność i precyzję wskazań urządzenia.

Weryfikacja układu pomiarowego (rys. 8.2) odbywała się poprzez podgląd przebiegów wewnętrz urządzeń. Kanał pierwszy przedstawia sygnał taktujący układ pomiarowy. Można zauważyć reakcję urządzenia na przekroczenie, przez charakterystykę ładowania/rozładowania (kanał 2), odpowiednich progów napięcia referencyjnego.

Mimo zastosowania trybu „x10” sondy oscyloskopowej, wniosła ona pewną pojemność pasożytniczą, dlatego pomiar z dołączonym urządzeniem wymaga dodatkowej kalibracji. Obecne ustawienia progów referencyjnych podczas pomiaru wynosiły 3,33 V (H_THR) oraz 1,67 V (L_THR). Przekroczenie jednego z tych progów aktywuje przerwanie mikrokontrolera, który przełącza stan na wyjściu taktującym.

Pomiar czasu ładowania i rozładowania odbywa się pomiędzy przełączeniami stanów sygnału taktującego. W ramach jednego pomiaru pojemności, mierzonych jest 128 próbek czasowych, łącznie ładowania i rozładowania obiektu badanego.

Charakterystyki umożliwiają także ocenienie stromości zbocz bufora trójstanowego, a także stabilność napięć wyjściowych tego elementu. Oba te czynniki zważą na dokładności pomiarowej.



Rys. 8.2. Przebieg ładowania i rozładowania rzeczywistego układu pomiarowego.

8.2 Pomiar pojemności wzorcowych

Wyznaczenie wzorcowych wartości pojemności uzyskano na podstawie serii pomiarów urządzeniami laboratoryjnymi: Agilent E4980A Precision LCR Meter oraz Agilent 16047A Text Fixture.

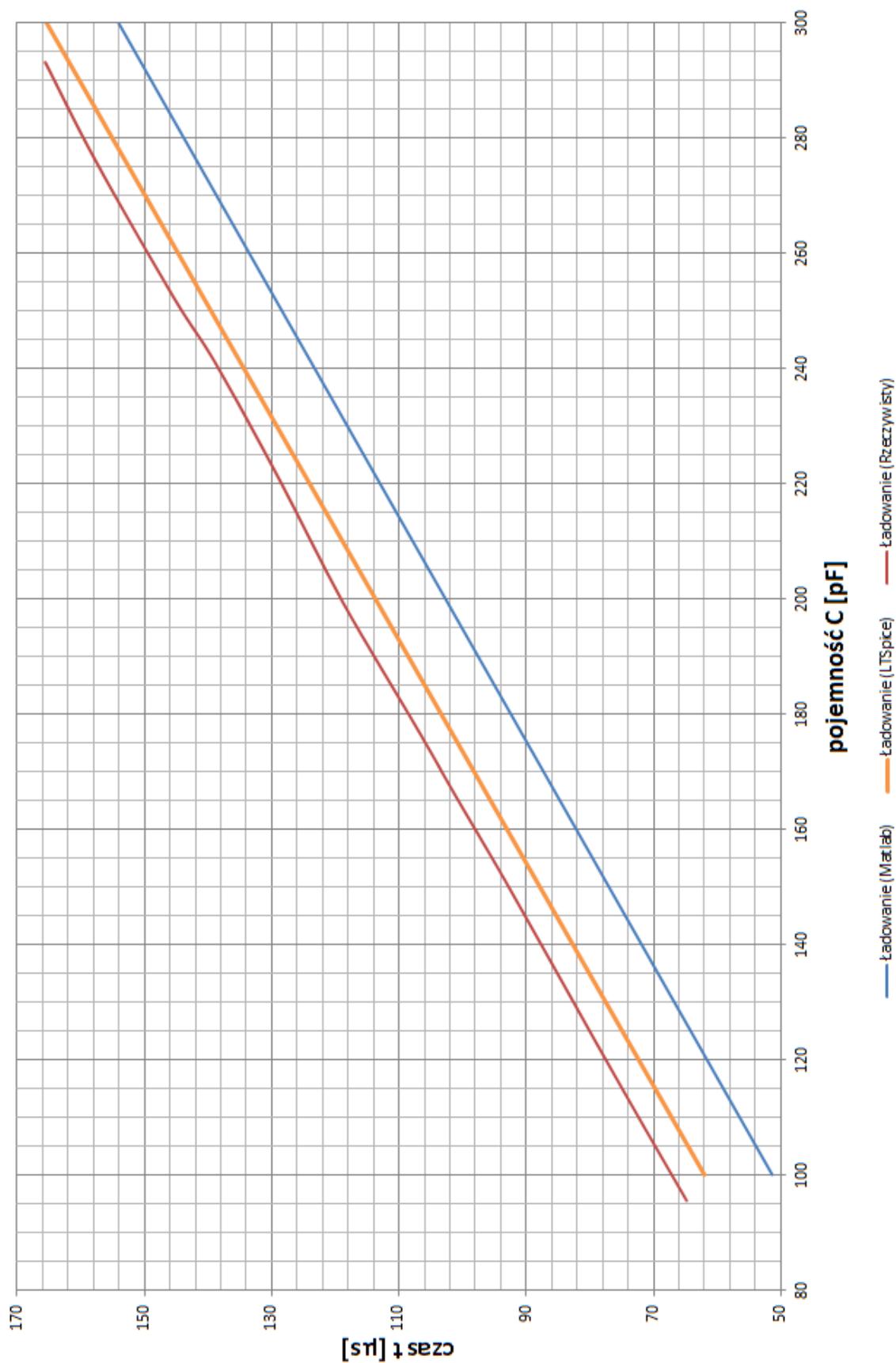
Pomiar wartości pojemności, odbywał się w tym samym środowisku pomiarowym, wykluczając wpływ np. zmian temperatury otoczenia na wskazania pojemności. Przeprowadzona seria 100 pomiarów została uśredniona, uzyskując tym samym dokładny, pozbawiony błędów losowych, wynik pojemności kondensatora. Wykorzystane mikowe, elementy pojemnościowe zapewniają stałe i dokładne parametry w długim okresie. Każdy pomiar wielkości, został poprzedzony kilku minutową pracą kondensatora w obwodzie prądu zmiennego, co pozwoliło na uzyskanie optymalnej temperatury pracy.

Zakres wyznaczonej charakterystyki obejmuje wartości pojemności od 100 do 300 pF. Do jej określenia wyznaczono 11 elementów wzorcowych (tabela 8.1).

Tabela 8.1. Wyniki pomiaru pojemności elementów wzorcowych [$R = 741,2 \text{ k}\Omega$].

Pojemność wzorcowa [pF]	Pojemność zmierzona [pF]	Błąd bezwzględny [pF]	Błąd względny [%]
95,487	126,183	30,696	32,147
116,656	147,601	30,945	26,526
149,561	180,222	30,662	20,501
165,738	196,822	31,084	18,755
177,278	208,429	31,149	17,570
199,371	231,355	31,984	16,042
220,511	250,606	30,096	13,648
240,784	270,195	29,411	12,215
252,610	283,432	30,822	12,202
276,201	307,149	30,948	11,205
293,120	322,389	29,269	9,985

Błędy powstałe na skutek nieidealności układu laboratoryjnego sięgają około 30 pF na przestrzeni wszystkich wykonanych pomiarów. Wielkość ta oznacza pojemności pasożytnicze, powstające w rzeczywistym układzie laboratoryjnym. Celem kalibracji jest wyeliminowanie wpływu elementów pasożytniczych.



Rys. 8.3. Porównanie charakterystyk czasów ładowania różnych modeli [$R = 741,2 \text{ k}\Omega$].

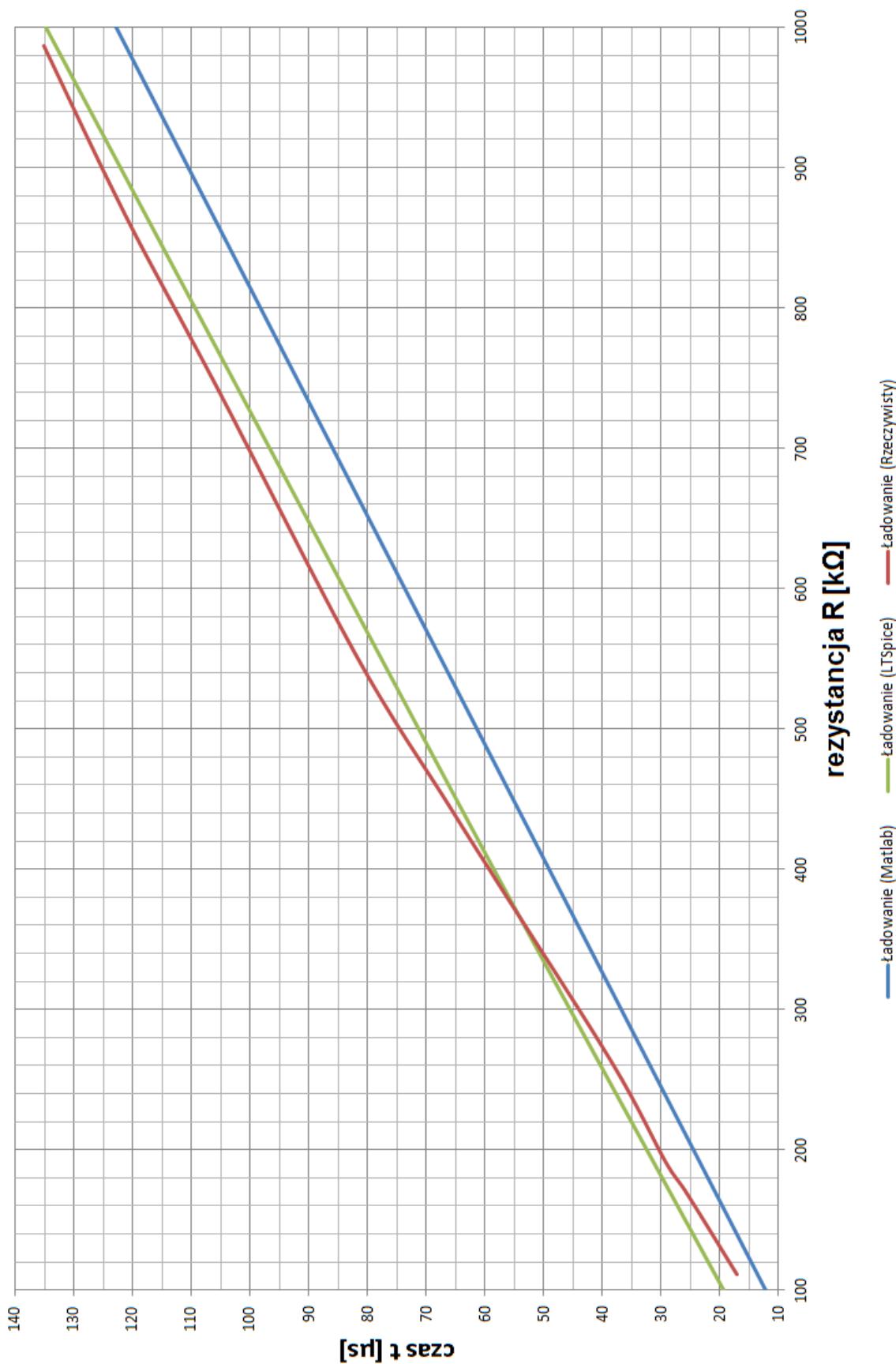
Charakterystyka czasów ładowania elementu pojemnościowego modelu rzeczywistego jest w przybliżeniu liniowa (rys. 8.3). Okazuje się, że oszacowane wartości elementów pasożytniczych, uwzględnionych w modelu symulacyjnym są zbyt optymistyczne. Model rzeczywisty bowiem, charakteryzuje się jeszcze większymi wartościami pojemnością pasożytniczą. Ukaże to sensowność wykonywania symulacji, przed wykonaniem układu rzeczywistego. W niektórych przypadkach mogłoby okazać się, że pomiar zadanej wielkości fizycznej jest niemożliwy w modelu rzeczywistym, poprzez niepoprawnie zaprojektowany układ.

Za pomocą rezystancji wzorcowych (tabela 8.2) została wyznaczona charakterystyka czasów ładowania pojemności wzorcowej 177,2 pF. Wspomniana charakterystyka (rys. 8.4) ukazuje nieliniową zależność pomiędzy rezystancją pomiarową oraz badaną pojemnością. Model rzeczywisty jest znacznie bardziej rozbudowany, niż zakładana symulacja. Oprócz pojemności pasożytniczych występują także pewne indukcyjności ścieżek na płytce PCB, taktowanie obwodu sygnałem zmiennym w czasie powoduje powstanie nieliniowych zjawisk, np. rezonansu.

Każdorazowa zmiana rezystora pomiarowego, wymusza na użytkowniku systemu wykonanie kalibracji urządzenia. W tym celu został opracowany skrypt Matlab wyznaczający współczynniki korekcji regresji liniowej oraz interpolacji Lagrange'a. Dodatkowo wyznacza charakterystykę korygującą punkty pomiarowe w zakresie od 100 do 300 pF, wykres błędów względnych, bezwzględnych oraz losowych, na podstawie punktów pomiarowych.

Tabela 8.2. Wpływ rezystancji na czasy ładowania i rozładowania pojemności 177,2 pF.

Wartość rezystancji wzorcowej [kΩ]	Zmierzony czas ładowania [μs]	Zmierzony czas rozładowania [μs]
110,9	17,13	16,70
168,5	25,64	25,65
191,2	29,30	29,02
260,9	38,27	38,20
385,1	57,10	57,63
446	66,30	66,15
552,6	82,04	81,96
741,2	105,43	105,17
866,4	121,34	121,12
986,7	135,15	135,07

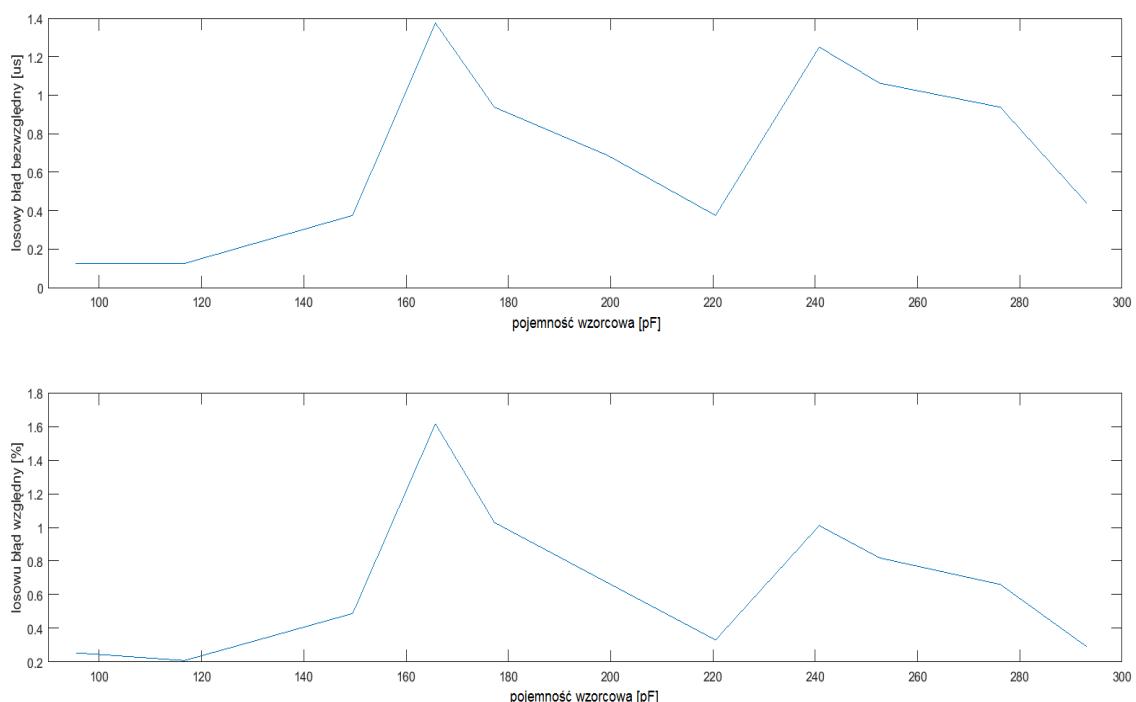


Rys. 8.4. Porównanie charakterystyk czasów ładowania pojemności 177,2 pF.

Błędy losowe powstałe na skutek oddziaływania czynników zewnętrznych prezentują niepowtarzalny zbiór (rys. 8.5), co świadczy o braku błędów systematycznych i poprawnym zaprojektowaniu układu pomiarowego. Dodatkowe ograniczenie przez zjawiskami zewnętrznymi zapewniłaby ekranowana obudowa, chroniąca np. doprowadzenia przed zewnętrznymi zakłóceniami elektromagnetycznymi. Na błąd losowy może wpływać również program wykonywany przez mikrokontroler ATmega32U4, ponieważ określony w dokumentacji czas wejścia do obsługi przerwania określony jest minimalną wartością. Program wykonujący instrukcję składającą się z kilku bajtów musi dokończyć jej wykonanie, co zajmuje więcej okresów zegara taktującego. Maksymalny błąd losowy sięga około 1,4 μ s, co równoważne jest około 22 cykli zegara taktującego mikrokontroler.

Błędy losowe zostały wyznaczone na podstawie serii 100 pomiarów, w tym samym środowisku i krótkim odstępie czasowym, dla każdej ze wzorcowych wartości pojemności. Nie zauważono znaczącego wpływu błędów losowych na wyniki pomiarowe.

Podczas każdego z pomiarów uzyskano 128 próbek czasu ładowania oraz rozładowania, metoda wyznaczająca błędy losowe oparta jest o różnicę najmniejszej wartości próbki od jej największej wartości w zbiorze. Otrzymane 100 wartości z serii, zostały uśrednione. Automatyzacja obróbki danych za pomocą programu Matlab ułatwiała, zarówno składowanie danych oraz analizę.



Rys. 8.5. Charakterystyka błędów losowych.

8.3 Korekcja pomiarów

W podrozdziale zostaną zaprezentowane metody korekcji punktów pomiarowych, wykorzystane podczas realizacji projektu. Szczególną uwagę poświęcono prezentacji skryptu Matlab, wyznaczającego współczynniki korekcji, oraz omówieniu generowanych charakterystyk błędów.

8.3.1 Algorytm regresji liniowej

Regresja liniowa aproksymuje serię danych pomiarowych do funkcji pierwszego stopnia $y = a \cdot x + b$, współczynniki wzoru określone są wzorami 8.1 oraz 8.2. Charakterystyka pomiarowa jest w przybliżeniu liniowa, dlatego jedna z metod korekcji wykorzystuje tą cechę. Algorytm realizujący regresję liniową (listing 8.1) został zaimplementowany w skrypcie Matlab.

$$a = \frac{\sum_{i=1}^n (y_i \cdot (x_i - \bar{x}))}{\sum_{i=1}^n (x_i - \bar{x})^2} \quad (8.1)$$

$$b = \bar{y} - a \cdot \bar{x} \quad (8.2)$$

gdzie:

- a, b – współczynniki regresji liniowej,
- \bar{x} – średnia wartość zbioru danych objaśniających,
- \bar{y} – średnia wartość zbioru danych rzeczywistych,
- x – zbiór danych objaśniających,
- y – zbiór danych rzeczywistych.

```
function [a, b, Y] = LinearRegression(x, y)
    x_avg = mean(x);
    y_avg = mean(y);
    a = sum(y .* (x - x_avg)) / sum((x - x_avg). ^ 2);
    b = y_avg - x_avg * a;
    Y = a * x + b;
end
```

Listing 8.1. Implementacja algorytmu regresji liniowej.

Maksymalna uzyskana niedokładność pomiaru po korekcji otrzymaną metodą wyniosła 0,678 % w zakresie od 100 do 300 pF (tabela 8.3). Oznacza to, że rozdzielcość pomiaru wilgotności względnej wyniesie aż 3 %, aby dodatkowo zwiększyć dokładność pomiarową użyto algorytmu interpolacji Lagrange'a.

Tabela 8.3. Korekcja z wykorzystaniem regresji liniowej.

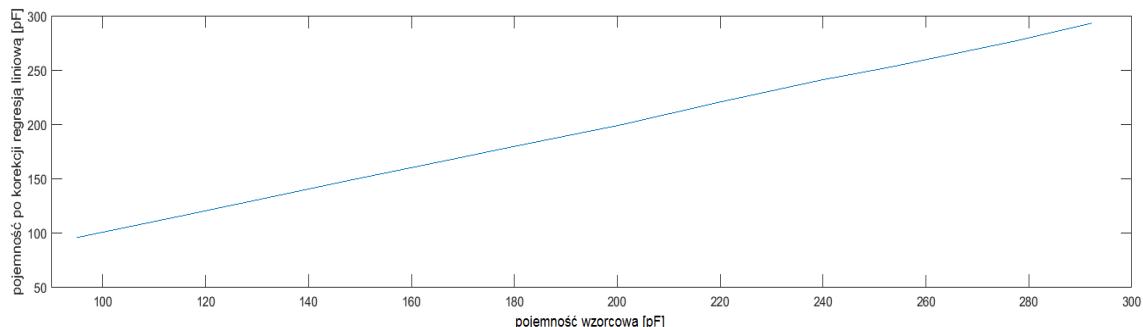
Pojemność wzorcowa [pF]	Pojemność po korekcji [pF]	Błąd bezwzględny [pF]	Błąd względny [%]
95,487	95,013	-0,474	-0,497
116,656	116,540	-0,117	-0,100
149,561	149,327	-0,233	-0,156
165,738	166,013	0,275	0,166
177,278	177,679	0,399	0,225
199,371	200,722	1,351	0,678
220,511	220,072	-0,438	-0,199
240,784	239,761	-1,024	-0,425
252,610	253,066	0,456	0,181
276,201	276,905	0,704	0,255
293,120	292,222	-0,899	-0,307

Wzór korygujący pojemność zmierzona został zaprezentowany w formule 8.3, algorytm zastosowano dla wszystkich zmierzonych punktów pomiarowych.

$$C_{korekcja} = 1,0051 \text{ [pF/pF]} \cdot C_{zmierzone} - 31,816 \text{ [pF]} \quad (8.3)$$

gdzie:

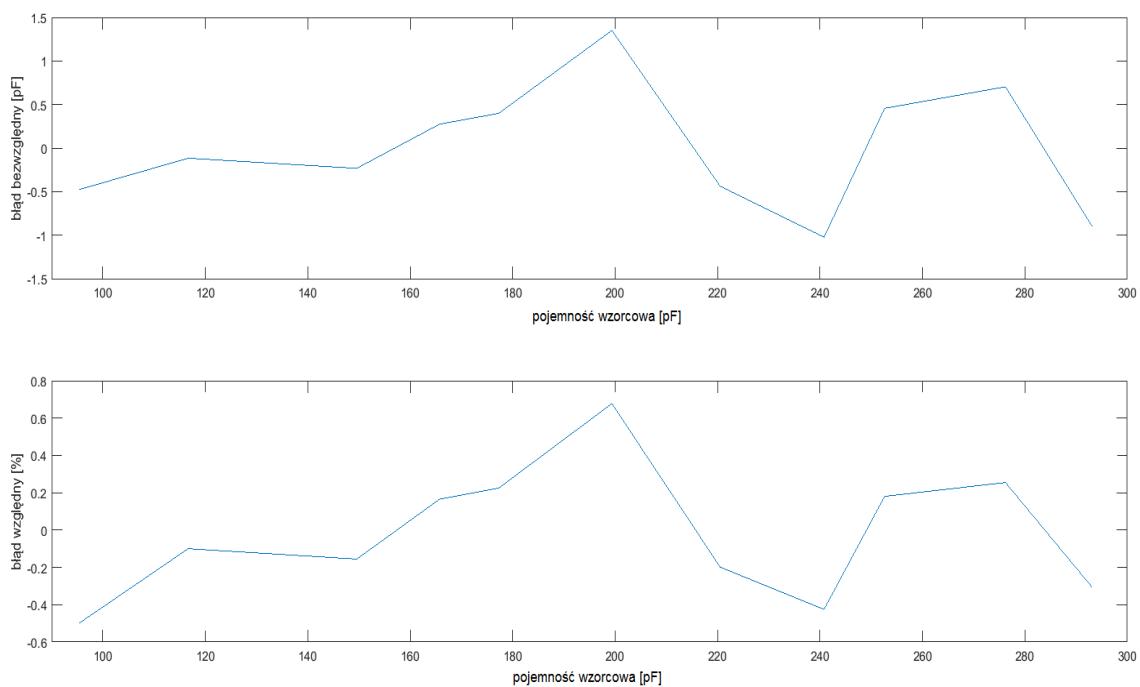
- $C_{korekcja}$ – wartość pojemności po korekcji [pF],
- $C_{zmierzone}$ – wartość pojemności zmierzanej [pF].



Rys. 8.6. Charakterystyka korekcji pojemności wielomianem pierwszego stopnia.

Charakterystyka aproksymacji wielomianem pierwszego stopnia (rys. 8.6) nie jest wymagająca obliczeniowo. Może zostać zaimplementowana wewnątrz programu mikrokontrolera. Nie zachowuje jednak wystarczającej dokładności, w przypadku realizacji układu rzeczywistego, powodem tego może być znacznie bardziej złożony model, zawierający zjawiska nieliniowe np. rezonanse czy linie długie, oraz elementy pasożytnicze.

Błędy korekcji wielomianem pierwszego stopnia (rys. 8.7), w zakresie małych wartości pojemności, posiadają akceptowalne błędy sięgające 0,2 %. Największe błędy zaobserwowano w okolicach 200 pF wartości pojemności wzorcowej, co może częściowo nachodzić na charakterystykę przyjmowanych pojemności przez sensor HS1101.



Rys. 8.7. Charakterystyki błędów po korekcji pojemności wielomianem 1. stopnia.

8.3.2 Algorytm interpolacji Lagrange'a

Interpolacja Lagrange'a jest numeryczną metodą przybliżania funkcji wielomianem n stopnia w wybranych $n + 1$ punktach. Oznacza to, że otrzymany wielomian przechodzi dokładnie przez wyznaczone punkty pomiarowe, aproksymując charakterystykę pomiędzy nimi. Algorytm służący do wyznaczenia wielomianu interpolacyjnego (8.4) wymaga serii danych pozwalających na jego wyznaczenie. Wewnątrz programu zaimplementowano możliwość interpolacji wielomianem 3 stopnia. Oznacza to, że algorytm skorzysta z 4 punktów pomiarowych pojemności wzorcowych, które dobrano doświadczalnie, aby zachować jak najwierniejsze odwzorowanie charakterystyki w przedziale przybieranych wartości przez sensor HS1101 (161 – 193 pF) [14].

$$W_n(x) = y_0 \frac{(x - x_1)(x - x_2) \dots (x - x_n)}{(x_0 - x_1)(x_0 - x_2) \dots (x_0 - x_n)} + y_1 \frac{(x - x_0)(x - x_2) \dots (x - x_n)}{(x_1 - x_0)(x_1 - x_2) \dots (x_1 - x_n)} + \dots + \\ (8.4) \\ y_n \frac{(x - x_0)(x - x_1) \dots (x - x_{n-1})}{(x_n - x_0)(x_n - x_1) \dots (x_n - x_{n-1})}$$

gdzie:

- $y_0 - y_n$ – zbiór wartości pojemności wzorcowych,
- $x_0 - x_n$ – zbiór wartości pojemności zmierzonych.

Algorytm interpolacji Lagrange'a został zaimplementowany w skrypcie Matlab (listing 8.2), który umożliwia zdefiniowanie symboli zmiennych, pozwalając w przejrzysty sposób wprowadzać skomplikowane formuły. Język ten zapewnia także funkcje umożliwiające przejście ze struktury formuły do tablicy współczynników wielomianu (ang. polynomial). W późniejszym etapie charakterystyka korekcyjna zostaje wyznaczona bezpośrednio z otrzymanych współczynników wielomianu.

```
function Poly = LagrangeInterpolation(IntX, IntY)
    syms x;
    Int(x) = ...
        IntY(1) * (x - IntX(2)) * (x - IntX(3)) * (x - IntX(4)) / ...
        ((IntX(1) - IntX(2)) * (IntX(1) - IntX(3)) * (IntX(1) - IntX(4))) + ...
        IntY(2) * (x - IntX(1)) * (x - IntX(3)) * (x - IntX(4)) / ...
        ((IntX(2) - IntX(1)) * (IntX(2) - IntX(3)) * (IntX(2) - IntX(4))) + ...
        IntY(3) * (x - IntX(1)) * (x - IntX(2)) * (x - IntX(4)) / ...
        ((IntX(3) - IntX(1)) * (IntX(3) - IntX(2)) * (IntX(3) - IntX(4))) + ...
        IntY(4) * (x - IntX(1)) * (x - IntX(2)) * (x - IntX(3)) / ...
        ((IntX(4) - IntX(1)) * (IntX(4) - IntX(2)) * (IntX(4) - IntX(3)));
    Poly = sym2poly(Int);
end
```

Listing 8.2. Implementacja algorytmu interpolacji Lagrange'a.

Ponieważ głównym celem realizacji projektu jest integracja urządzenia laboratoryjnego z czujnikiem wilgotności względnej HS1101, algorytm interpolacji Lagrange'a został wykalibrowany z myślą o pracy układu pomiarowego z pojemnościami z przedziału: 161 – 193 pF [14]. Zapewniając również w szerszym zakresie błąd względny, mniejszy od 0,1 %. Taka dokładność pomiaru zapewnia rozdzielcość odczytu wilgotności względnej na poziomie 0,3125 %, co jest w zupełności wystarczającą wartością (tabela 8.4). Chęć dokładniejszej aproksymacji większego przedziału charakterystyki doprowadziła do uzyskania maksymalnej wartości błędu względnego na poziomie 0,5 %. Niezwykłą zaletą tej metody jest możliwa konfiguracja charakterystyki korygującej, poprzez dobór serii danych pomiarowych, zasilających algorytm interpolacji.

Tabela 8.4. Korekcja z wykorzystaniem interpolacji Lagrange'a.

Pojemność wzorcowa [pF]	Pojemność po korekcji [pF]	Błąd bezwzględny [pF]	Błąd względny [%]
95,487	94,016	-1,471	-1,541
116,656	116,656	0	0
149,561	149,560	0	0
165,738	165,838	0,100	0,060
177,278	177,124	-0,156	-0,088
199,371	199,371	0	0
220,511	218,208	-2,302	-1,044
240,784	237,737	-3,047	-1,266
252,610	251,244	-1,366	-0,541
276,201	276,300	0,099	0,036
293,120	293,120	0	0

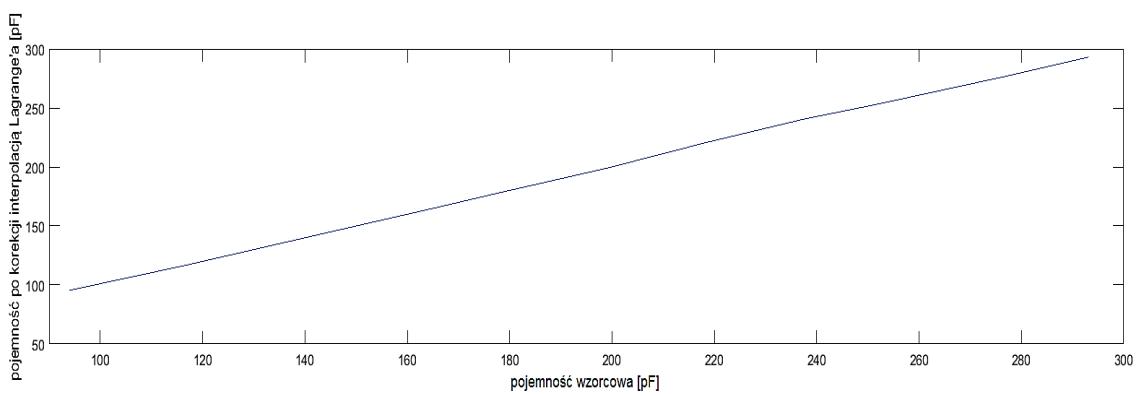
Wyznaczony wielomian korekcyjny 3 stopnia, został zaprezentowany w formule 8.5, seria danych zapewniająca akceptowane dokładności pomiarowe, składa się z wartości wzorcowych: 116,656 pF, 149,561 pF, 199,371 pF oraz 293,120 pF. Potwierdzają to zerowe błędy interpolacji (tabela 8.4).

$$C_{korekcja} = 4,6 \cdot 10^{18} \cdot C_{zmierzone}^3 - 2,9 \cdot 10^9 \cdot C_{zmierzone}^2 + 1,6 \cdot C_{zmierzone} - 7,2 \cdot 10^{11} \quad (8.5)$$

gdzie:

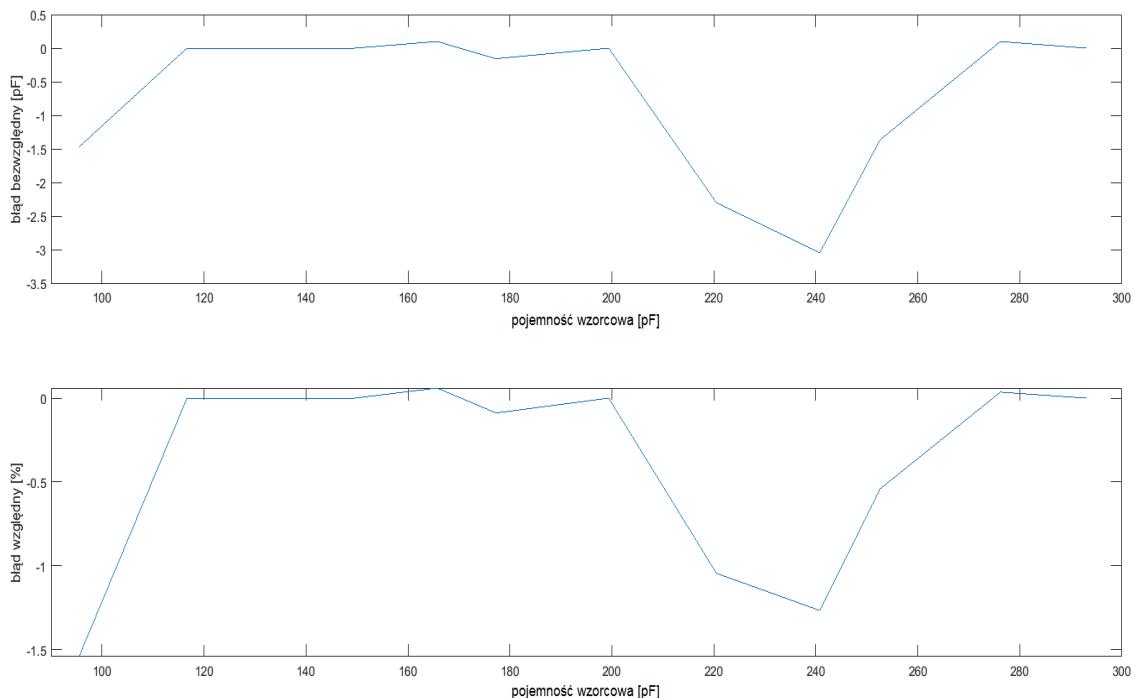
- $C_{korekcja}$ – wartość pojemności po korekcji [F],
- $C_{zmierzone}$ – wartość pojemności zmierzonej [F].

Charakterystyka korekcyjna trzeciego stopnia (rys. 8.8) jest wymagająca obliczeniowo, ze względów czasowych, wykorzystanie go wewnątrz 8-bitowego mikrokontrolera jest niemożliwe. Aproxymacja zapewnia najmniejsze błędy wewnątrz przedziału obejmującego wskaźania czujnika HS1101. Każdorazowa zmiana rezystora pomiarowego wymaga kalibracji układu pomiarowego oraz charakterystyki korekcyjnej.



Rys 8.8. Charakterystyka korekcji pojemności wielomianem trzeciego stopnia.

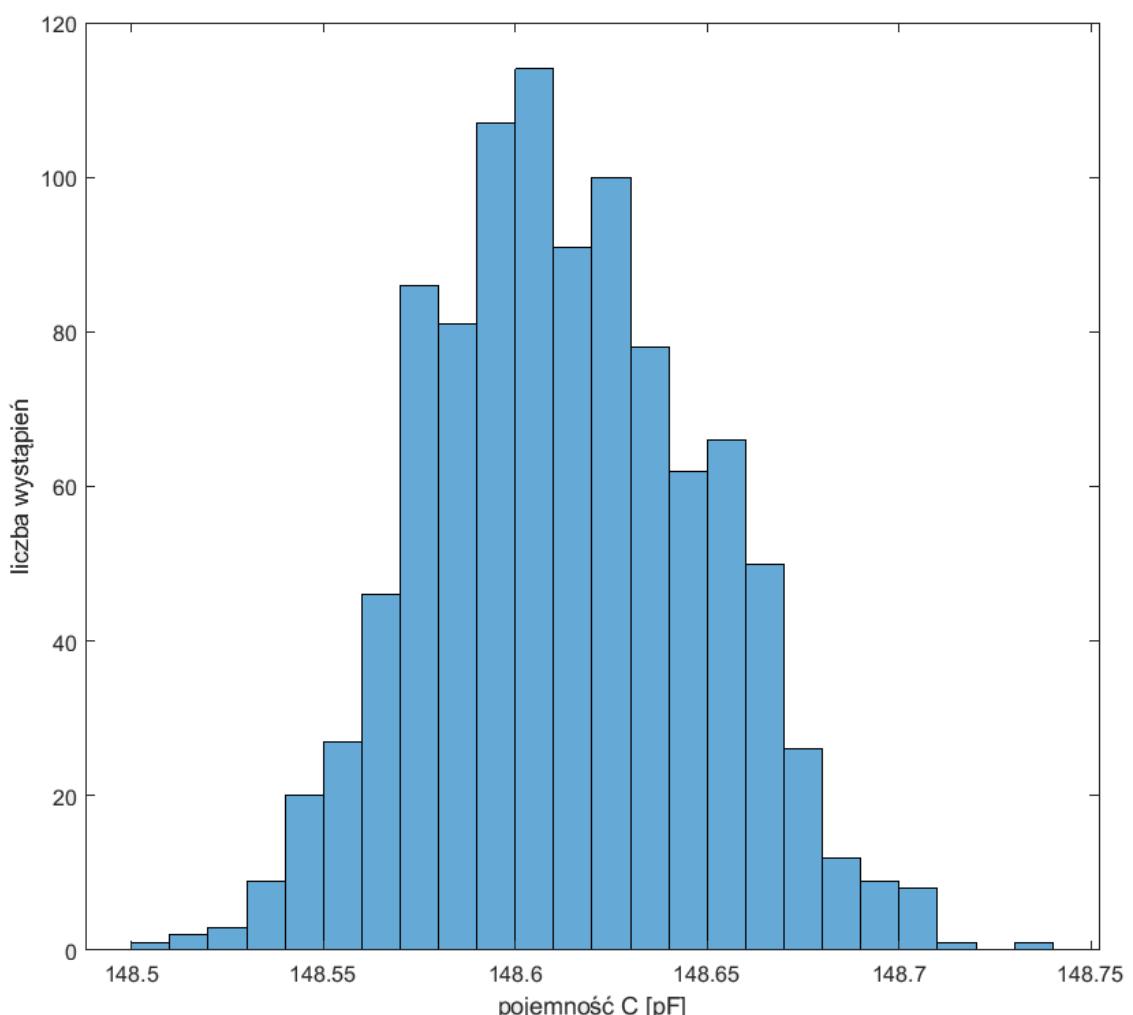
W przedziale od 115 do 200 pF wskazania pojemności zmierzanej praktycznie pokrywają się z wartościami wzorcowymi (rys 8.9). Podczas pomiaru wartości większej niż 200 pF zaobserwowano wyraźne odchylenie charakterystyki błędu do -1,2 %. Cechą powtarzającą się dla 3 bliskich siebie wartości pojemności, świadczy to o zajściu zjawisk nieliniowych podczas pomiaru. Występujące błędy mają charakter systematyczny, gdyż wartości zostały uśrednione, eliminując tym samym błąd losowy. Podobne zjawisko zaobserwować można dla pojemności mniejszej od 100 pF. Pomiar tak niskich wartości oparty jest błędem rozdzielczości czasowej mikrokontrolera oraz nadmiernym ładowaniem i rozładowaniem pojemności badanej, w czasie obsługi przerwania przez procesor. Dlatego pomiar ten nie został wzięty pod uwagę podczas wyznaczania korekcji danych.



Rys 8.9. Charakterystyki błędów po korekcji pojemności wielomianem 3. stopnia.

Za pomocą histogramu możliwe jest graficzne przedstawienie rozkładu badanej pojemności. W celu jego wyznaczenia wykonano 1000 pomiarów, pominięto zaokrąglanie wyniku, dla pojemności badanej $148,6 \text{ pF}$ (rys. 8.10). Kolejne wyzwalanie pomiaru odbywało się co $0,5 \text{ s}$. Wynik doświadczenia ukazuje liczbę wystąpień zmierzonych wartości pojemności. Zgodnie z założeniami posiada on charakter rozkładu Gaussa.

Jednym z najważniejszych wniosków płynących z doświadczenia jest określenie precyzji pomiarowej, która w tym przypadku wynosi 0,2 pF, co jest bardzo dobrym wynikiem. Przełożenie tej wartości na wilgotność względową odpowiada 0,625 % RH w przeciągu ponad 8 minutowego pomiaru. Natomiast znaczna większość wyników mieści się w granicach 148,58 – 148,64 pF. Oznacza to, że zaokrąglanie wyniku pomiarowego pozytywnie wpłynie na stabilność i dokładność wskazań. Dodatkową poprawę precyzji pomiarowej udałoby się uzyskać, dzięki zastosowaniu ekranowanej obudowy, wpływ otoczenia podczas kilkuminutowego pomiaru z pewnością oddziaływały na układ pomiarowy.



Rys. 8.10. Histogram serii 1000 pomiarów pojemności 148,6 pF.

9 PODSUMOWANIE

W niniejszej pracy dyplomowej zaprojektowano i zrealizowano układ inteligentnego czujnika pojemnościowego, bazującego na 8-bitowym mikrokontrolerze ATmega32U4. Dzięki takiej konfiguracji wykorzystano wbudowany interfejs USB, służący do komunikacji z interfejsem użytkownika. Aplikacja komputerowa umożliwiła kalibrację, analizę danych oraz monitorowanie wilgotności względnej w czasie rzeczywistym. Pomiar oparto na czujniku pojemnościowym HS1101, którego parametry zależą bezpośrednio od wilgotności otoczenia. Dokonano także weryfikacji pomiarowej, dzięki której osiągnięto błąd względny na poziomie 0,1 % dla zakresu pokrywającego się ze wskazaniami sensora HS1101, realizując tym samym wszystkie założone cele.

Część teoretyczna została zapoczątkowana przeglądem możliwych metod pomiarowych. Słusznie zauważono, że metoda ładowania/rozładowania członu RC pozwoli na dokładny pomiar pojemności rzędu pF. Szczególną uwagę poświęcono omówieniu modelu matematycznego, idealnego układu pomiarowego, a także symulacji modelu z uwzględnieniem elementów pasożytniczych, w programie typu Spice. Okazało się, że model symulacyjny, realnie odzwierciedla zjawiska wyłaniające się z rzeczywistym układzie pomiarowym.

Wyniki pomiarów, jakie uzyskano drogą doświadczalną, są zadowalające. Powstałe błędy są w pełni akceptowalne i pozwalające na wyznaczenie wilgotności względnej, z dokładnością 0,3125 % RH. Natomiast, precyzyja wewnętrz 10 minutowego, ciągłego pomiaru została doświadczalnie określona wartością 0,625 % RH. Takie parametry charakteryzują dokładne urządzenia laboratoryjne.

Realizacja prototypowego układu pozwala na łatwą konfigurację progów napięć referencyjnych oraz rezystancji pomiarowych, żądanych przez użytkownika systemu. Jednym z ograniczeń rozwiązania jest brak obudowy ekranującej. Szereg złącz diagnostycznych oraz układ pomiarowy, są narażone na działanie zewnętrznych zakłóceń. Czynnikiem wpływającym na charakter pomiaru jest także, rozdzielcość zegara taktującego mikrokontroler. Zastosowanie 32-bitowych jednostek, opartych na pętli PLL, pozwoliłoby na znaczące zmniejszenie opóźnienia wnoszonego przez przerwanie, a tym samym dokładniejsze wyznaczenie czasów ładowania i rozładowania pojemności mniejszych od 100 pF.

WYKAZ LITERATURY

- [1] Czaja Z. *A measurement method for capacitive sensors based on a versatile direct sensor-to-microcontroller interface circuit* (2020).
- [2] Czaja Z. *A measurement method for lossy capacitive relative humidity sensors based on a direct sensor-to-microcontroller interface circuit* (2021).
- [3] Czaja Z. *Measurement method for capacitive sensors for microcontrollers based on a phase shifter* (2022).
- [4] Meng Y., Dean R. N. *A Technique for Improving the Linear Operating Range for a Relative Phase Delay Capacitive Sensor Interface Circuit* (2016).
- [5] Meng Y., Dean R. N., Adams M. L. *Improving the phase delay capacitive interface circuit technique using MOSFET switches* (2019).
- [6] Skiba A., Tiliouine H. *Stany nieustalone w obwodach elektrycznych. Przykłady i zadania.* Wyd. Politechniki Gdańskiej, Gdańsk 2022.
- [7] Kardaś M. *Mikrokontrolery AVR. Język C. Podstawy programowania.* Wyd. ATNEL, Szczecin 2011.
- [8] Baranowski R. *Mikrokontrolery AVR ATmega w praktyce.* Wyd. BTC, Warszawa 2005.
- [9] Sells C. *Windows Forms Programming in C#.* Wyd. Addison-Wesley Professional, 2003.
- [10] Sklep internetowy z modułami elektronicznymi (data dostępu: 1.10.2022 r.)
<https://botland.com.pl/>
- [11] Nota katalogowa mikrokontrolera ATmega32U4 (data dostępu: 1.10.2022 r.)
https://ww1.microchip.com/downloads/en/devicedoc/atmel-7766-8-bit-avr-atmega16u4-32u4_datasheet.pdf
- [12] Nota katalogowa modułu Arduino Micro (data dostępu: 1.10.2022 r.)
<https://docs.arduino.cc/hardware/micro>
- [13] Nota katalogowa modułu SHTC3 (data dostępu: 1.10.2022 r.)
https://sensirion.com/media/documents/643F9C8E/6164081E/Sensirion_Humidity_Sensors_SHTC3_Datasheet.pdf
- [14] Nota katalogowa czujnika wilgotności HS1101 (data dostępu: 1.10.2022 r.)
<https://www.jameco.com/Jameco/Products/ProdDS/2082901.pdf>
- [15] Nota katalogowa stabilizatora monolitycznego 7805 (data dostępu: 1.10.2022 r.)
<https://www.ti.com/lit/ds/symlink/lm340.pdf>
- [16] Nota katalogowa układu komparatora TLV3502AID (data dostępu: 1.10.2022 r.)
<https://www.ti.com/lit/ds/sbos507a/sbos507a.pdf>
- [17] Nota katalogowa bufora trójstanowego, 74AHC244 (data dostępu: 1.10.2022 r.)
<https://www.ti.com/lit/ds/symlink/sn54ahc244-sp.pdf>
- [18] Nota katalogowa biblioteki LUFA USB (wersja oprogramowania: 210130)
<http://www.fourwalledcubicle.com/LUFA.php>

WYKAZ RYSUNKÓW

Rys 2.1. Schemat ideowy układu pomiarowego przesuwnika fazowego [4]	8
Rys 2.2. Schemat blokowy metody pomiarowej sygnałem prostokątnym [1].	9
Rys 2.3. Schemat blokowy metody pomiarowej, bazującej na ładowaniu/rozładowaniu RC.	10
Rys 2.4. Charakterystyka błędu względnego, w zależności przesunięcia fazowego [5].....	11
Rys 2.5. Sygnały panujące w metodzie pobudzenia sygnałem prostokątnym [1].....	12
Rys 3.1. Obwód pomiarowy.....	13
Rys 3.2. Analiza operatorowa obwodu pomiarowego.	13
Rys 3.3. Przekształcony obwód pomiarowy w dziedzinie operatorowej.	14
Rys 3.4. Schemat bloku pomiarowego – symulacja.....	15
Rys 3.5. Schemat bloku porównywania napięć progowych – symulacja.	16
Rys 3.6. Schemat podłączenia mikrokontrolera – symulacja.....	17
Rys 3.7. Schemat bufora trójstanowego – symulacja.	18
Rys 3.8. Charakterystyka czasu ładowania i rozładowania dla rezystancji $R = 741,2 \text{ k}\Omega$	19
Rys 3.9. Charakterystyka czasu ładowania i rozładowania dla pojemności $C = 177,2 \text{ pF}$	20
Rys 3.10. Charakterystyka sygnałów w czasie pomiaru, $C = 177,2 \text{ pF}$, $R = 741,2 \text{ k}\Omega$	21
Rys 4.1. Schemat blokowy stanowiska pomiarowego.	22
Rys 4.2. Schemat blokowy układu laboratoryjnego.	23
Rys 4.3. Schemat ideowy układu zasilającego.	24
Rys 4.4. Moduł Arduino Micro [10].	25
Rys 4.5. Schemat ideowy podłączenia modułu Arduino Micro.	26
Rys 4.6. Moduł SHTC3 [10].....	27
Rys 4.7. Moduł wyświetlacza alfanumerycznego LCD [10].....	28
Rys 4.8. Schemat ideowy bufora trójstanowego.	28
Rys 4.9. Schemat ideowy członu pomiarowego RC.	29
Rys 4.10. Schemat ideowy toru napięć referencyjnych.	30
Rys 4.11. Schemat ideowy toru przetwarzania sygnału pomiarowego.	31
Rys 4.12. Schemat ideowy złącz diagnostycznych.	32
Rys 4.13. Projekt mozaiki PCB – górna strona.	33
Rys 4.14. Projekt mozaiki PCB – dolna strona.	33
Rys 4.15. Realizacja fizyczna układu laboratoryjnego.	34
Rys 5.1. Rejestr statusowy mikrokontrolera ATmega32U4 [11].....	35
Rys 5.2. Diagram kontrolera USB [11].	36
Rys 5.3. Schemat blokowy układu licznika w trybie przechwytywania zdarzeń [11].....	37
Rys 5.4. Rejestr konfiguracyjny licznika 1 [11].	38
Rys 5.5. Rejestr masek przerwań licznika 1 [11].	38
Rys 5.6. Sposób podłączenia urządzeń w interfejsie TWI [11].	39
Rys 5.7. Typowa transmisja danych interfejsu TWI [11].	39

Rys 5.8. Rejestr kontrolny interfejsu TWI [11].....	39
Rys 6.1. Algorytm pętli głównej programu mikrokontrolera.....	42
Rys 6.2. Algorytm prezentujący pomiar pojemności.....	46
Rys 7.1. Strona startowa interfejsu użytkownika.....	48
Rys 7.2. Ustawienia ogólne interfejsu użytkownika.....	49
Rys 7.3. Przykład charakterystyki wilgotności i temperatury w czasie.....	50
Rys 7.4. Kalibracja urządzenia z poziomu interfejsu użytkownika.....	52
Rys 7.5. Wywołanie pomiarów z poziomu interfejsu użytkownika.....	53
Rys 7.6. Podgląd charakterystyk w interfejsie użytkownika.....	56
Rys 7.7. Dziennik zdarzeń interfejsu użytkownika.....	57
Rys 8.1. Diagram stanowiska pomiarowego.....	58
Rys 8.2. Przebieg ładowania i rozładowania rzeczywistego układu pomiarowego.....	59
Rys 8.3. Porównanie charakterystyk czasów ładowania różnych modeli [$R = 741,2 \text{ k}\Omega$].....	61
Rys 8.4. Porównanie charakterystyk czasów ładowania pojemności $177,2 \text{ pF}$	63
Rys 8.5. Charakterystyka błędów losowych.....	64
Rys 8.6. Charakterystyka korekcji pojemności wielomianem pierwszego stopnia.....	66
Rys 8.7. Charakterystyki błędów po korekci pojemności wielomianem 1. stopnia.....	67
Rys 8.8. Charakterystyka korekcji pojemności wielomianem trzeciego stopnia.....	70
Rys 8.9. Charakterystyki błędów po korekci pojemności wielomianem 3. stopnia.....	70
Rys 8.10. Histogram serii 1000 pomiarów pojemności $148,6 \text{ pF}$	71

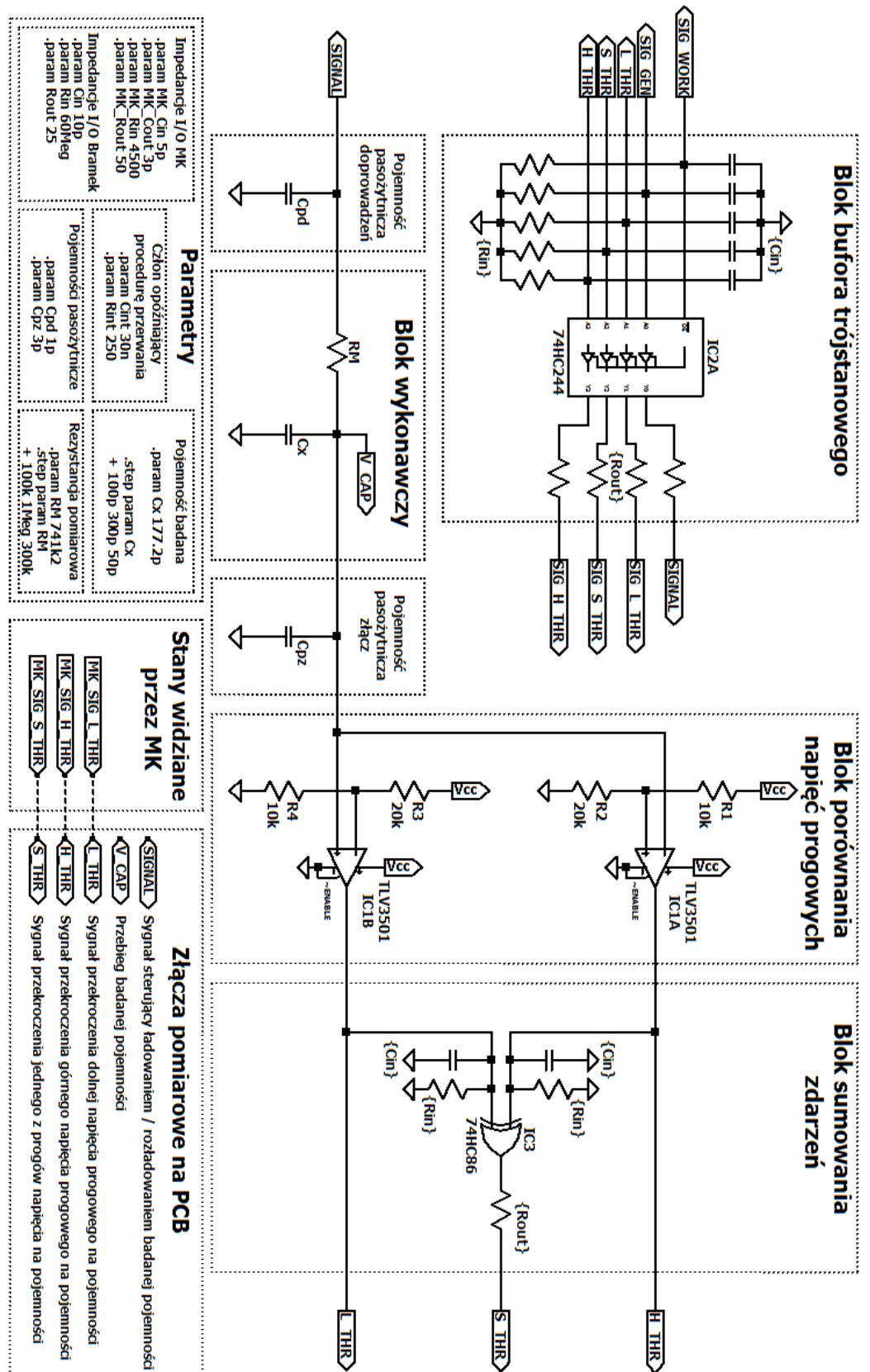
WYKAZ TABEL

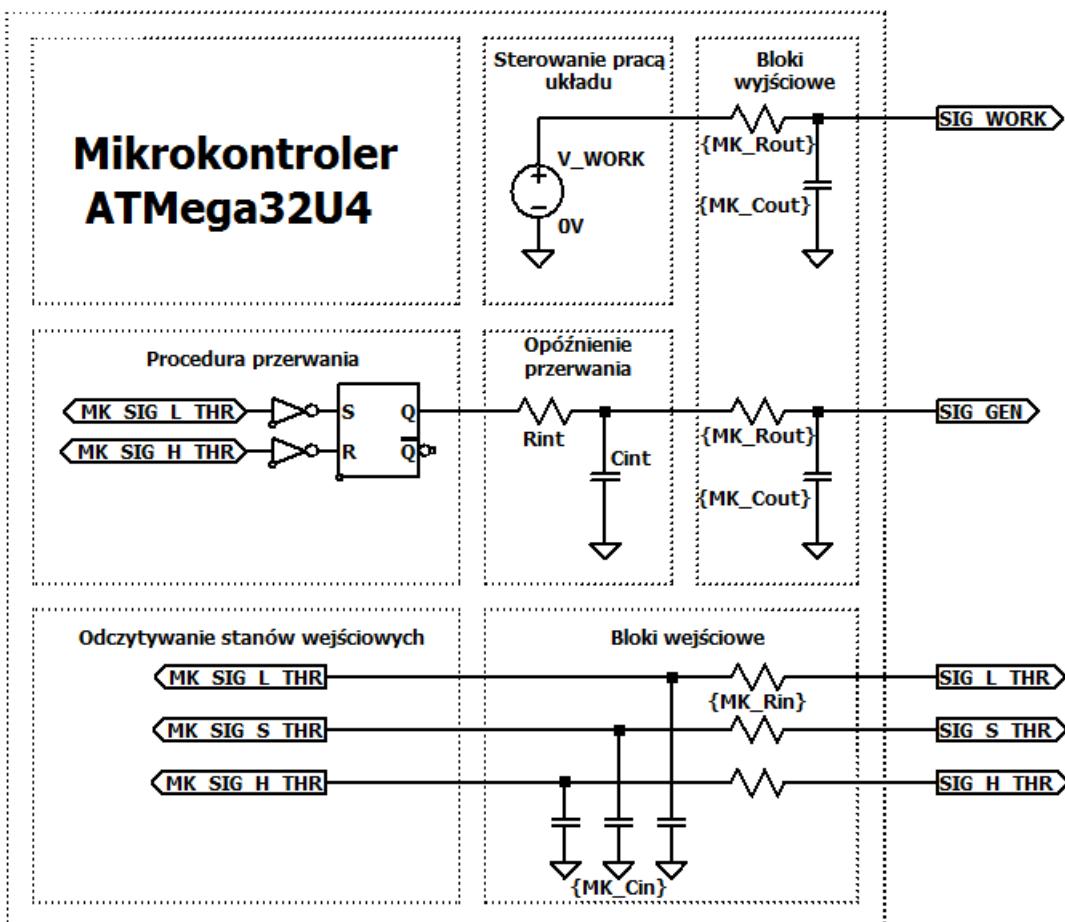
Tabela 8.1. Wyniki pomiaru pojemności elementów wzorcowych [$R = 741,2 \text{ k}\Omega$].	60
Tabela 8.2. Wpływ rezystancji na czasy ładowania i rozładowania pojemności 177,2 pF.	62
Tabela 8.3. Korekcja z wykorzystaniem regresji liniowej.	66
Tabela 8.4. Korekcja z wykorzystaniem interpolacji Lagrange'a.	69

WYKAZ LISTINGÓW

Listing 6.1. Funkcja inicjalizacji interfejsu USB	41
Listing 6.2. Funkcja inicjalizacji modułu przechwytywania	41
Listing 6.3. Obsługa instrukcji kalibrujących	43
Listing 6.4. Definicja domyślnych wartości struktur danych kalibracyjnych	44
Listing 6.5. Funkcja pomiaru pojemności	47
Listing 6.6. Obsługa przerwania modułu przechwytywania zdarzeń zewnętrznych	47
Listing 7.1. Przykład eksportu dziennika zdarzeń w formacie XML	51
Listing 7.2. Funkcja wyznaczająca pojemność z danych pomiarowych	53
Listing 7.3. Funkcja realizująca nadpróbkowanie i decymację	54
Listing 7.4. Funkcja obliczająca pojemność	54
Listing 7.5. Funkcja obliczająca wilgotność względną na podstawie pojemności czujnika	55
Listing 7.6. Zestaw instrukcji interpretowanych przez układ laboratoryjny	57
Listing 8.1. Implementacja algorytmu regresji liniowej	65
Listing 8.2. Implementacja algorytmu interpolacji Lagrange'a	68

DODATEK A – SYMULACJA LTSPICE





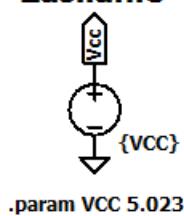
Biblioteki LTSpice

.inc 74HC.lib
.inc TLV3501.lib

Symulacje

```
.tran 0.5ms
.meas tran T1 find time when
+ V(MK_SIG_S_THR)=2.5 rise 1
.meas tran T2 find time when
+ V(MK_SIG_S_THR)=2.5 rise 2
.meas tran T3 find time when
+ V(MK_SIG_S_THR)=2.5 rise 3
.meas tran T4 find time when
+ V(MK_SIG_S_THR)=2.5 fall 2
```

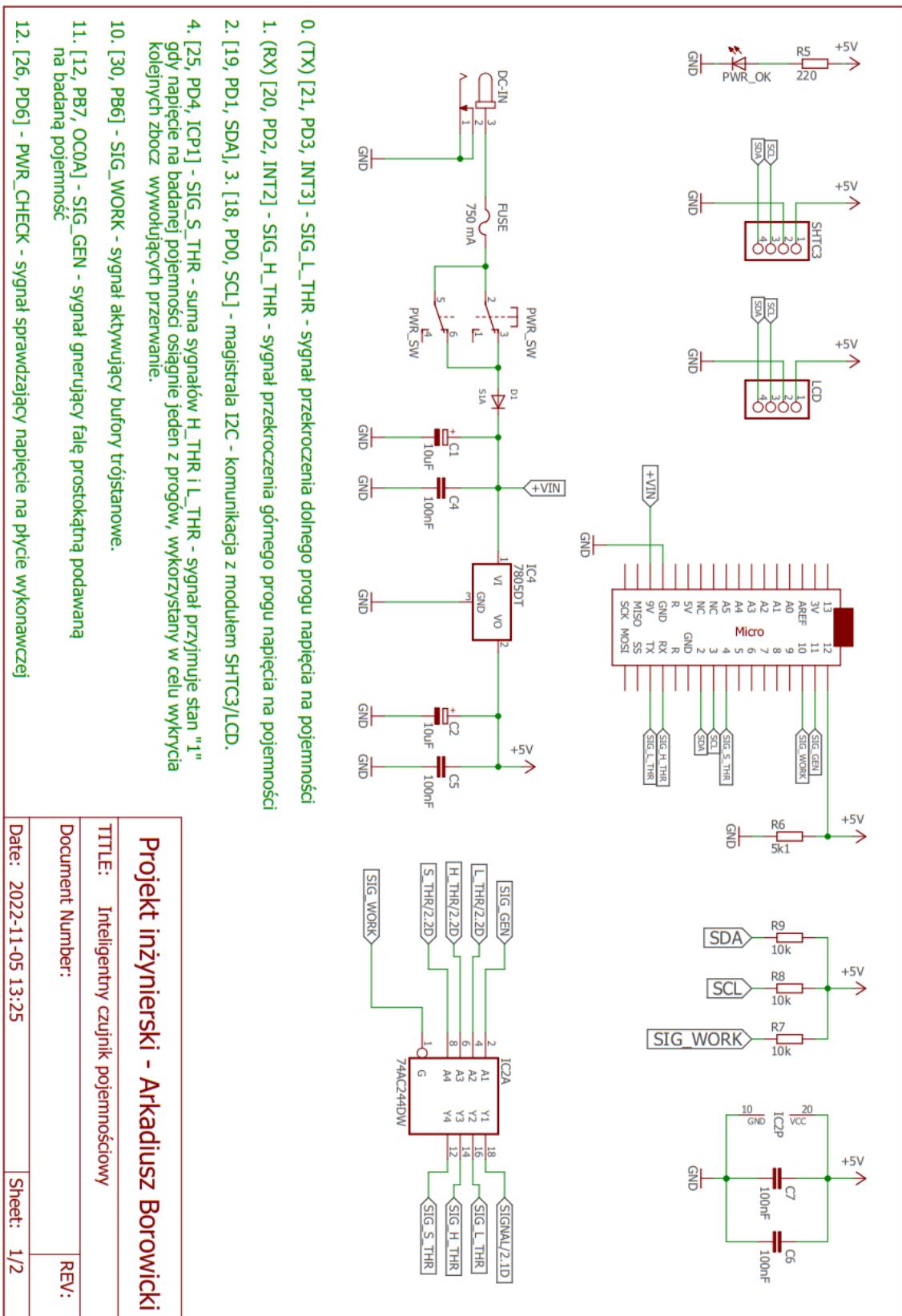
Zasilanie

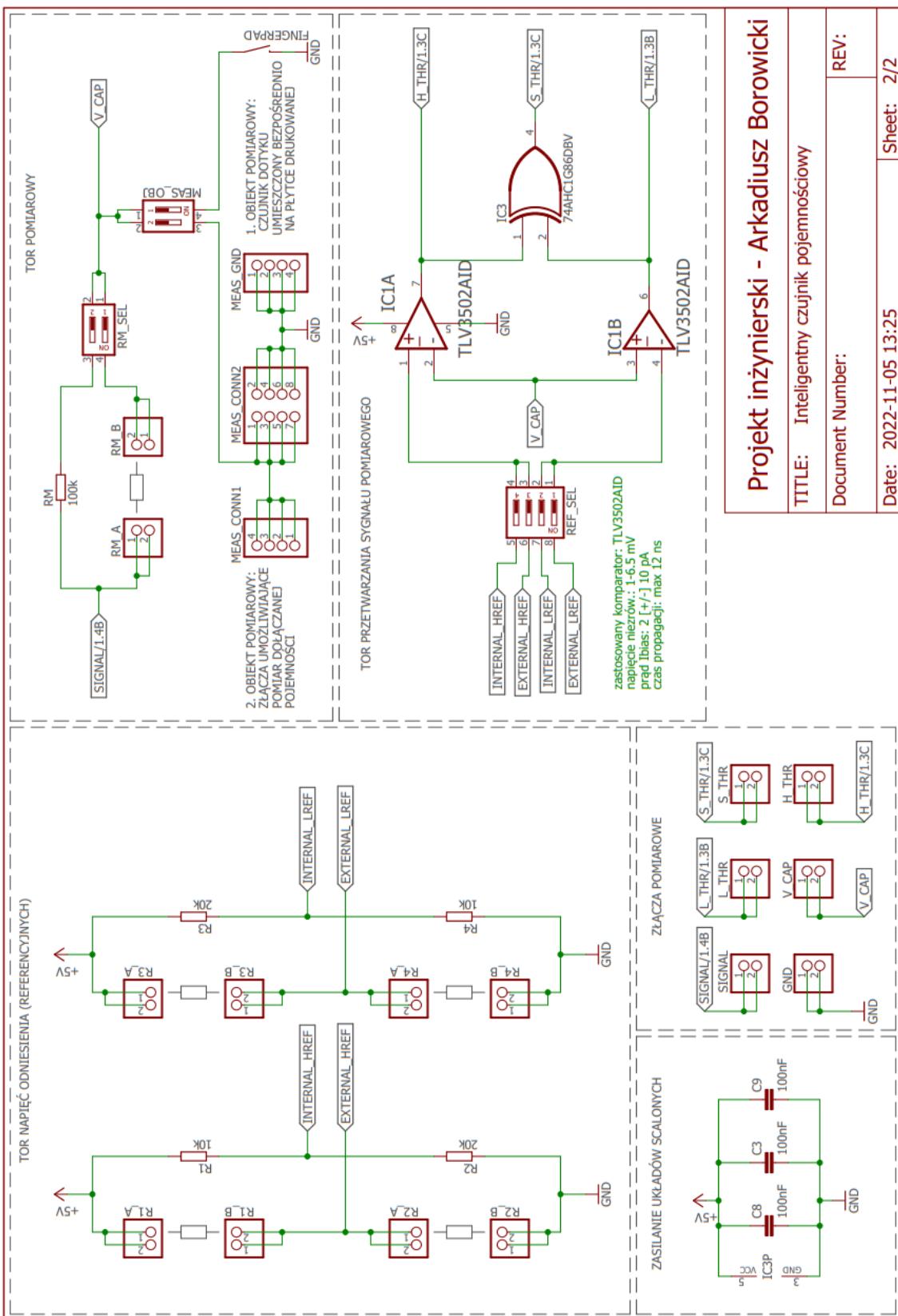


Pomiar zależności czasowych

.meas tran IntTime param T4-T1	Czas przerwania
.meas tran ChargingTime param (T3-T2)*1E6	Czas ładowania pojemności [us]
.meas tran ChargingTicks param (T3-T2)*16E6	Czas ładowania wyrażony w cyklach zegara procesora
.meas tran DischargingTime param (T2-T1)*1E6	Czas rozładowania pojemności [us]
.meas tran DischargingTicks param (T2-T1)*16E6	Czas rozładowania wyrażony w cyklach zegara procesora

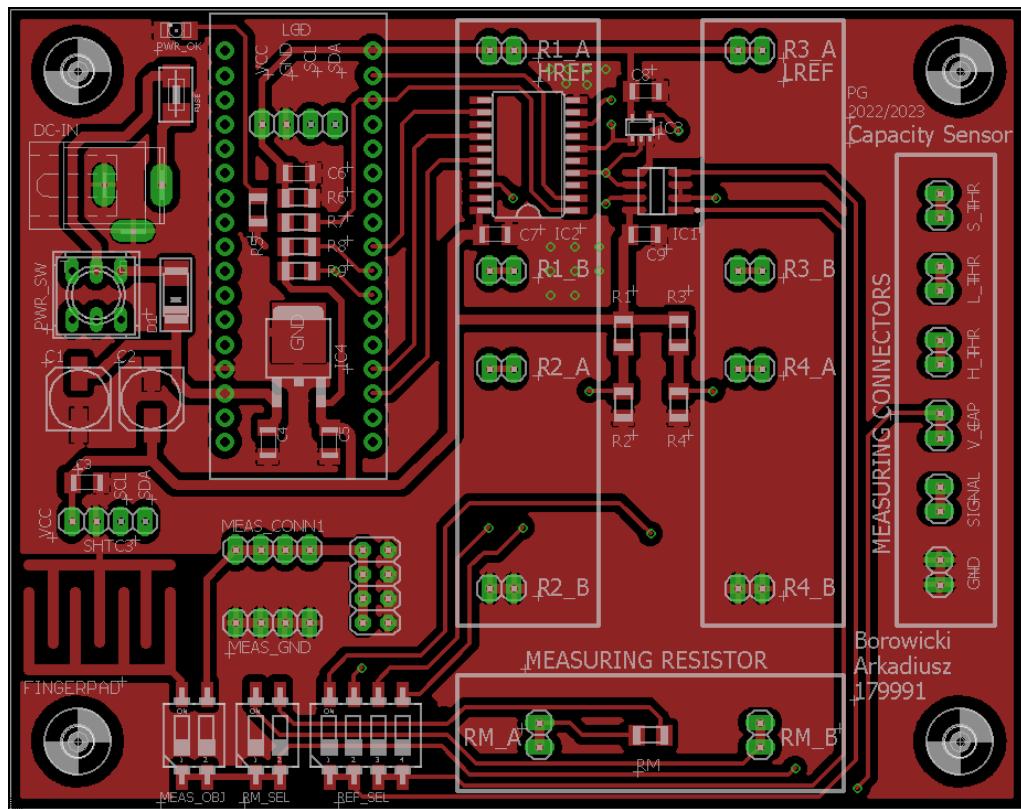
DODATEK B – SCHEMAT IDEOWY





DODATEK C – SCHEMAT MONTAŻOWY

Warstwa góra:



Warstwa dolna:

