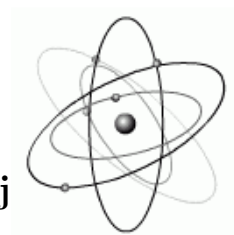




Akademia Górniczo-Hutnicza
im. Stanisława Staszica w Krakowie



Wydział Fizyki i Informatyki Stosowanej

AGH

Praca inżynierska

Rafał Zuzak

kierunek studiów: fizyka techniczna

specjalność: fizyka jądrowa

Oprogramowanie urządzeń pracujących w standardzie General Purpose-Interface Bus (GP-IB)

Opiekun: dr hab. Marek Idzik

Ocena: Data: Podpis:

Kraków, styczeń 2008

Oświadczam, świadomy(-a) odpowiedzialności karnej za poświadczenie nieprawdy, że niniejszą pracę dyplomową wykonałem(-am) osobiście i samodzielnie i nie korzystałem(-am) ze źródeł innych niż wymienione w pracy.

SPIS TREŚCI

1	Wstęp	4
2	Standard <i>General Purpose-Interface Bus</i>(GP-IB)	4
2.1	Historia	4
2.2	Struktura fizyczna	4
2.3	Komendy i rozkazy	7
2.4	Obsługa zdarzeń i błędów w GP-IB	8
3	Oprogramowanie	9
3.1	Program dla analizatora HP4195A	9
3.1.1	Dostępne funkcje	9
3.1.2	Procedury testowe	12
3.2	Program dla zasilacza HP6624A	13
3.2.1	Dostępne funkcje	13
3.2.2	Procedury testowe	13
4	Wyniki pomiarów	14
4.1	Wyniki pomiaru szumów analizatorem HP4195A	14
4.1.1	Wyniki dla układu Front-End board 2	15
4.2	Wyniki pomiaru widma częstotliwościowego analizatorem HP4195A i zasilaczem HP6624A	20
4.2.1	Wyniki dla filtra aktywnego pasmowo przepustowego	21
4.2.2	Wyniki dla filtrów pasywnych	22
5	Podsumowanie	23
6	Dodatek A	24
6.1	Testy sondy dołączonej do analizatora	24
7	Dodatek B	25
7.1	Spis dostępnych funkcji dla napisanych programów	25
8	Literatura	26
9	Spis rysunków	26
10	Spis tabel	27

1 Wstęp

Celem pracy było napisanie programów, w języku C++, do obsługi dwóch urządzeń pracujących w laboratorium elektroniki jądrowej Wydziału Fizyki i Informatyki Stosowanej AGH, urządzeniami tymi były analizator HP4195a i zasilacza HP6624a. W pracy przedstawiono szczegóły standardu, na którym pracują oprogramowywane urządzenia, następnie opisano urządzenia od strony sprzętowej i programowej zwracając uwagę na istotne aspekty techniczne wykorzystanego sprzętu, które miały znaczenie podczas tworzenia oprogramowania. Opisane zostały wszystkie procedury testowe, które wykorzystane zostały podczas sprawdzania poprawności programów. Na końcu przedstawiono przykładowe pomiary wraz z opisem i wynikami, wykonane za pomocą urządzeń i napisanego oprogramowania.

2 Standard *General Purpose-Interface Bus*(GP-IB)

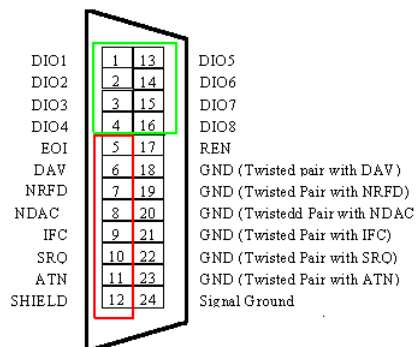
W tej części pracy przedstawiono standard GP-IB, z którego korzystają urządzenia. Najpierw opisano historię standardu(jego rozwój, i uaktualnienia), następnie strukturę fizyczną(czyli medium przesyłowe, maksymalną prędkość łącza, stany w jakich mogą pracować urządzenia etc.), kolejno przedstawiono część programową standardu, czyli opis komend i rozkazów, ich format i sposób przekazywania do urządzenia. W ostatniej części pokrótce wspomniano o systemie obsługi zdarzeń i błędów.

2.1 Historia

Standard GP-IB, powstał w roku 1976. Stworzony został przez firmę Hewlett Packard, głównym celem który przyświecał twórcom, było ułatwienie komunikacji pomiędzy komputerem, a urządzeniem pomiarowym. Pierwsza wersja standardu (zwana HP-IB, od nazwy firmy Hewlett Packard) ustandaryzowała jedynie medium przesyłowe, a więc opisano przewód, łączący urządzenia. Wyznaczono maksymalną liczbę urządzeń pracujących na jednym kontrolerze, ustalono stany w jakich dane urządzenia mogą pracować. Dopiero w roku 1986, międzynarodowa organizacja IEEE wprowadziła do standardu pewien obowiązujący zbiór komend i rozkazów, nadal jednak brakowało opisu i usystematyzowania sposobu w jaki komendy są tworzone. W tamtym okresie praktycznie każda firma pisała komendy we własnym języku, co powodowało duże trudności w kompatybilności. W roku 1989, Hewlett Packard zaproponował, by komendy pisać w języku TML (z ang. Test Measurement Language), niestety chciał go udostępnić jedynie na zasadzie licencji, co nie zostało zaakceptowane przez inne firmy (takie na przykład jak Tektronix). Do konsensusu doszło w roku 1990, kiedy to powstał język SCPI (z ang. Standard Commands for Programmable Instruments), który wyewoluował z języka TML, ale w przeciwieństwie do niego, był to język otwarty. Język ten do dziś uznawany jest jako oficjalny, którym należy tworzyć nowe komendy dla standardu GP-IB.

2.2 Struktura fizyczna

Medium przesyłowe stanowi przewód 24 żyłowy, podzielony na trzy bloki, w którym osiem żył odpowiada za przysyłanie rozkazów, kolejnych osiem to linie przesyłania danych (oznaczone jako DIO x , gdzie $x \in [0,7]$), a ostatnie osiem żył to linie masy, których celem jest eliminacja powstałych zakłóceń. Poniżej przedstawiono widok na wyjście przewodu z opisanymi odpowiednimi pinami.



Rysunek 1: Widok na przekrój przewodu GP-IB, na czerwono zaznaczone linie odpowiedzialne za kontrolę rozkazów, na zielono linie danych, pozostałe to linie masy.

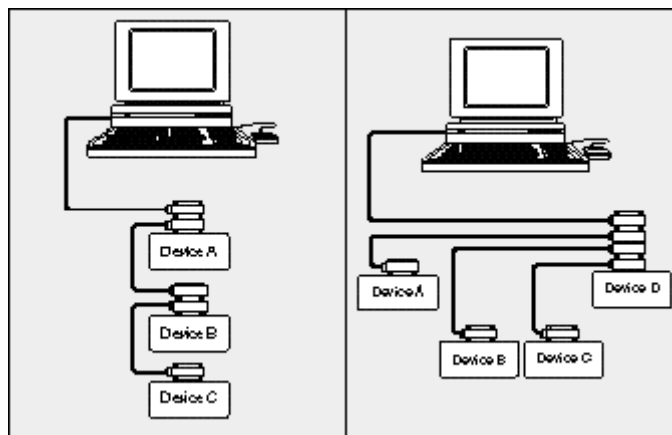
Linie kontrolujące rozkazy wydawane urządzeniom są odpowiedzialne za kontrolę całej transmisji, od jej nawiązania i rozpoczęcia, do zakończenia połączenia. Poniżej w tabeli przedstawiono krótki opis wszystkich ośmiu linii kontroli transmisji.

Nazwa linii	Stan urządzenia	Opis
NRFD	Listner	żądanie dodatkowych informacji, np. zakresu etc.
NDAC	Listner	potwierdzenie odebrania danych i ewentualne zgłoszenie gotowości przyjęcia kolejnych
DAV	Talker	urządzenie nadające zawiadania, że są dostępne do odebrania dane na liniach DIOx
ATN	C.-In-Ch.(*)	aktywny wtedy gdy wysyłane są komunikaty przez kontroler
EIO	Talker	używany jako zakończenie polecenia
IFC	Sys. Cont. (**)	kontroler zeruje informacje o adresach i stanach
REN	Sys. Cont.	aktywowanie, powoduje że kontrolę nad daną transmisją przejmuje adresat
SRQ	Talker	informacja urządzenia o tym iż wymaga obsłużenia (odebrania danych etc.)

Tablica 1: Linie do obsługi rozkazów. (*) Controller-In-Charge, czyli kontroler podrzędny. (**) System Controller, czyli kontroler główny(najczęściej karta PC w komputerze).

Linie oznaczone jako DIOx jak było to już wspomniane służą przesyłowi informacji. Maksymalna prędkość przesyłania informacji w standardzie GP-IB została ustalona na 1 Mb/s, jest to jednak możliwe jedynie w specyficznych warunkach, gdyż na szybkość przesyłu ma wpływ wiele czynników takich jak na przykład długość stosowanych przewodów. W standardzie przyjęto, iż maksymalna długość przewodów w sieci nie powinna być dłuższa niż 20 metrów, natomiast długość łącza pomiędzy dwoma wybranymi urządzeniami nie powinna przekraczać 2 metrów. Każde urządzenie w sieci posiada swój adres (od 1 do 30), ustalany albo fizycznie (za pomocą na przykład zworek), albo zapisywany w wewnętrznej pamięci EEPROM urządzenia. Specjalnym adresem jest adres 31, odpowiednik broadcastu IP, czyli jest to adres rozsyłowy.

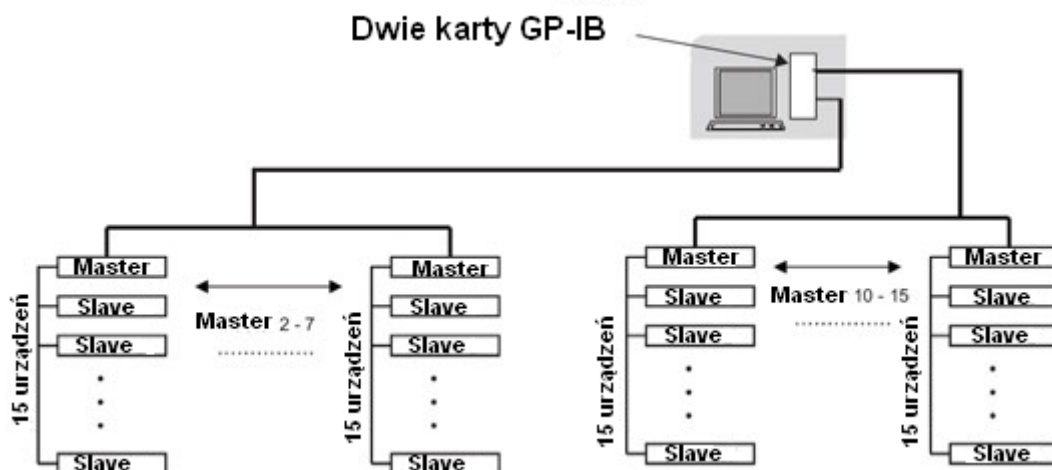
Sieć urządzeń można rozbudować w strukturę liniową lub gwiazdową, co przedstawiono na rysunkach poniżej.



Rysunek 2: Przykład struktury połączenia, po lewej połączenie liniowe, po prawej gwiazdowe.

Dostępne są trzy stany w których mogą znajdować się urządzenia. Stanami tymi są: *Listner*, *Talker* i *Controller*. Dodatkowo specjalny status posiada karta sterująca, która jest kontrolerem głównym i to do niej należy rozdysponowywanie rozkazów i nadawanie statusów. Gdy urządzenie przechodzi w stan nadawania pracuje jako *Talker*, a gdy kontroler główny nadaje do urządzenia pracuje ono jako *Listner*. Kontroler główny może nadać także jednemu z urządzeń w sieci tak zwany status kontrolera lokalnego, który także może rozdysponowywać odpowiednie statusy pozostałym urządzeniom w sieci.

Maksymalna liczba urządzeń w sieci to 15 (łącznie z kontrolerem), ale strukturę taką można znacznie rozszerzyć poprzez na przykład stosowanie kilku kart GP-IB w jednym komputerze. Przykładowy schemat takiej szerszej struktury przedstawiono poniżej.



Rysunek 3: Schemat szerszej struktury, z siecią zawierającą więcej niż jeden kontroler.

Jak widać poprzez takie podłączanie urządzeń możemy rozbudować sieci do większych rozmiarów, w których pracować może kilkadziesiąt urządzeń.

2.3 Komendy i rozkazy

W tym podpunkcie opisano strefę programową standardu, przedstawiono dostępne komendy i rozkazy. Jak było już wspomniane we wstępie, istnieje pewna określona grupa poleceń, którą muszą rozpoznawać wszystkie urządzenia. Ich krótki opis przedstawiono w tabeli poniżej.

*IDN?	urządzenie poda nam swoje dane, takie jak nazwa producenta, nr. seryjny etc.
*CLS	żądanie wyczyszczenia statusu
*RST	reset urządzenia
*WAI	żądanie wstrzymania danej operacji
*STB	bajt stanu urządzenia, czyli jak urządzenie jest ustawione
*SRE	ustawienie maski SRE
*SRE?	pobranie maski SRE
*OPC	komunikat zakończenia danej operacji
*OPC?	pobranie stanu operacji
*ESR?	pobranie bajtu stanu ESR urządzenia
*ESE	ustawienie maski stanu urządzenia
*ESE?	pobranie maski stanu urządzenia
*TST?	wyzerowanie i sprawdzenie stanu urządzenia

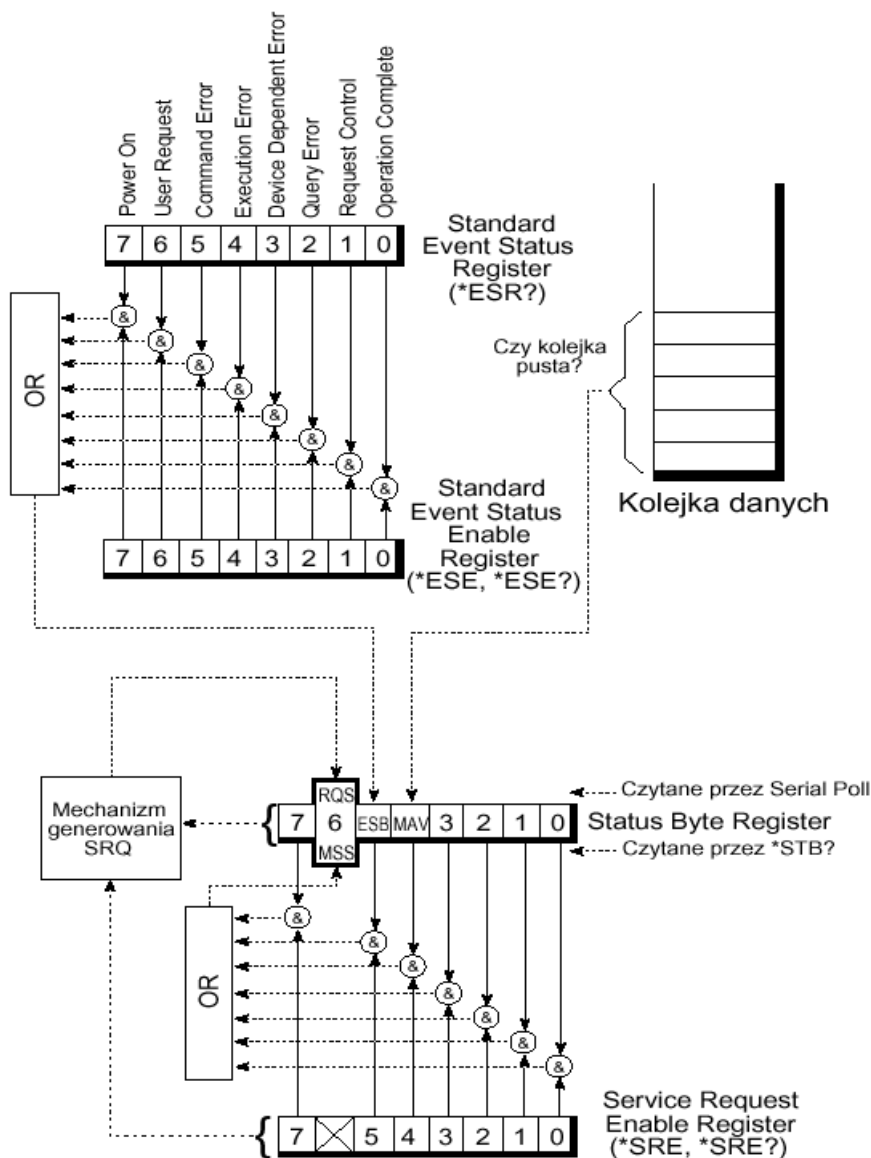
Tablica 2: Podstawowe komendy standardu GP-IB, wprowadzone do normy IEEE 488.2

Komendy przesyłane są do urządzenia jako 7 bitowy kod ASCII, dzięki temu w łatwy sposób można kontrolować urządzenie poprzez na przykład komputer PC. Koniec komendy oznacza się poprzez znak LF i/lub aktywny sygnał EOI. Liczby przesyłane są zawsze w formacie stało-przecinkowym z kropką dziesiętną (czyli na przykład 4.5). Jeżeli istnieje konieczność wysłania kilku komend jednocześnie to oddziela się je średnikiem (na przykład *komenda1;komenda2*). Istnieją dwa rodzaje komend, czyli komendy poprzez które żądamy aby urządzenie ustawiło daną wielkość (napięcie, zakres etc.), oraz komendy, w których żądamy zwrócenia wybranej wielkości (przykładowo żądanie podania ustawionego napięcia), komendy te najczęściej podaje się w formie *komenda?* (na końcu komendy podajemy znak zapytania).

Samych komend jest bardzo dużo, gdyż każde urządzenie poza komendami standardowymi, posiada swój własny zestaw komend charakterystycznych, tworzony przez producenta (obecnie we wspomnianym wcześniej języku SCPI). Komendy standardowe precyzuje i opisuje norma 488.2 ustanowiona przez IEEE (z ang. Institute of Electrical and Electronics Engineers).

2.4 Obsługa zdarzeń i błędów w GP-IB

Obsługa zdarzeń i błędów w standardzie GP-IB jest dość skomplikowana i rozbudowana. Istnieje szereg specjalnie przeznaczonych do tego celu rejestrów, których zadaniem jest gromadzenie i przetwarzanie zdarzeń z transmisji, czy informowanie o wystąpieniu błędu. Opisano jedynie najważniejsze rejestry, które znacznie wpływają na przebieg transmisji. Poniżej zaprezentowano schemat blokowy, reprezentujący obsługę zdarzeń i błędów.



Rysunek 4: Schemat blokowy, reprezentujący obsługę zdarzeń i błędów w GP-IB.

Rejestr ESR to rejestr w którym przechowywane są informacje na temat stanu urządzenia, oraz o wykonywanych komendach. Rejestr ten zawiera 8 bitów z tym, że bit 6 nie jest używany przez sam rejestr i jest bitem dostępnym dla użytkownika. Rejestr ESE służy do sprawdzania stanów jakie zapisane zostały w rejestrze ESR, bity z rejestru ESE i ESR poddawane są sumie logicznej, a następnie wynik dostępny jest w piątym bicie rejestru STB. Kolejnym ważnym bitem w rejestrze STB jest bit 4, zwany MAV, który sygnalizuje, czy są dostępne w buforze danych informacje do odczytania.

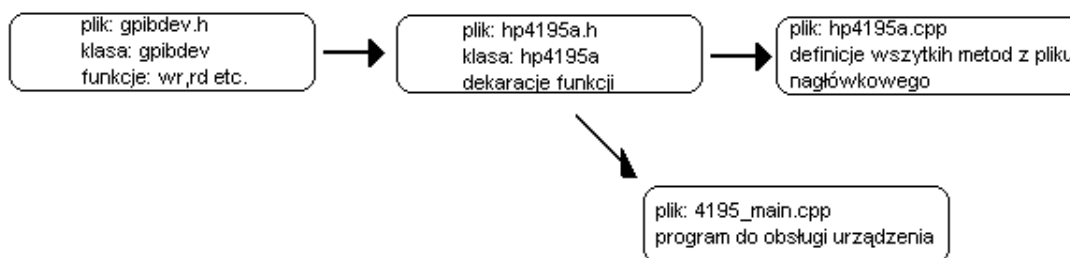
3 Oprogramowanie

Główną klasą po której dziedziczą oba napisane programy, była klasa **gpibdev** dostępna poprzez plik nagłówkowy **gpibdev.h**. W pliku tym zdefiniowana była większość metod wykorzystanych przy pisaniu oprogramowania. Poniżej opisano podstawowe metody, które użyte zostały w pisanych programach.

wr(cmd)	żądanie zapisania jakiegoś ustawienia (napięcia etc.)
query(cmd)	żądanie zwrócenia danej wielkości (ustawionego prądu etc.), zwraca wartość jako string
rd(cmd)	żądanie podania danej wartości, zwraca wartość jako liczbę całkowitą

Tablica 3: Podstawowe metody wykorzystywane podczas pisania programów, pochodzące z biblioteki *gpibdev*.

Poza tymi podstawowymi komendami, istnieją rozkazy takie jak na przykład ***IDN?**, ale z powodu tego, iż na przykład analizator pracuje na wersji standardu z roku 1978 wymagały pewnych przeróbek. Poniżej przedstawiono schemat (na podstawie analizatora HP4195a) dziedziczenia i struktury plików dla napisanych programów.



Rysunek 5: Struktura plików i dziedziczenia dla napisanych programów.

3.1 Program dla analizatora HP4195A

Głównym celem pisanego oprogramowania, było odczytywanie widma częstotliwościowego i na jego podstawie wyznaczenia wielkości szumów w badanym układzie. Dodatkowo napisano funkcje odpowiedzialne za uzyskiwanie z urządzenia widma częstotliwościowego.

3.1.1 Dostępne funkcje

Struktura programu obejmuje trzy główne bloki. Na początku program wczytuje dane z pliku konfiguracyjnego o nazwie **4195a.cnf**. W pliku tym znajdują się wszystkie konfigurowalne opcje, począwszy od pasma częstotliwości, poprzez inne ustawienia takie jak jednostki, aktywne rejestry, czy tryb analizy (szumowa, czy widmo częstotliwościowe). Plik konfiguracyjny, to typowy plik **.ini**, którego wycinek zaprezentowano dalej.

```

#rodzaj analizy
ANALYZE = 1

[SPECTRUM]
PORTS = 1

TRACE_A = 1
TRACE_B = 0
. . .

```

Rysunek 6: Wycinek pliku konfiguracyjnego dla HP4195A.

Po wczytaniu wszystkich ustawień, program jest gotowy do wykonania pomiaru. Do wyboru jest pomiar jednokrotny, lub seria pomiarowa składająca się z kilku/kilkunastu pomiarów. Po tym etapie, program pyta użytkownika do jakiego pliku zapisane mają zostać uzyskane dane. Format pliku wyjściowego zaprezentowano poniżej.

```

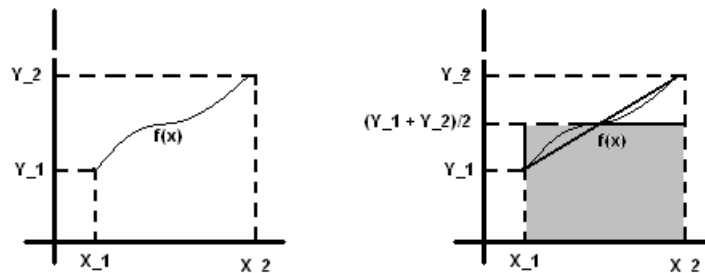
#Całka: 2.17097 mV
#Ilość powtórzeń: 5
#Odchylenie standardowe: 0.0673228 mV

#Ucięto pikę sieci: tak
10 12.9628
10.453 14.5925
10.927 34.3172
11.422 18.1097
11.94 6.27066 . . .

```

Rysunek 7: Wycinek pliku wyjściowego dla HP4195A.

Przechodząc do funkcji zawartych w programie należy wspomnieć, iż podstawową funkcją, którą obsługuje napisany program jest wyliczanie całki z gęstości widmowej mocy szumów, czyli wartości średniokwadratowej napięcia szumów. Całka wyliczana jest metodą trapezów, a więc następuje aproksymacja widma do funkcji liniowej. Analizator wszystkie dane pomiarowe gromadzi w dwóch rejestrach o budowie macierzy, które posiadają po 401 komórek. Program pobiera informacje z dwóch sąsiednich komórek rejestru X (w którym zapisane są wartości częstotliwości), następnie analogicznie pobiera dwie wartości z sąsiednich komórek rejestru A (w którym zapisane są wartości szumów w $[\frac{\mu V}{\sqrt{Hz}}]$). Kolejno liczona jest średnia arytmetyczna z pobranych komórek rejestru A, a także liczona jest różnica pomiędzy pobranymi komórkami z X. Na końcu następuje obliczenie iloczynu uzyskanych wartości i dopisanie ich do zmiennej, która reprezentuje sumę. Poniżej przedstawiono sposób aproksymacji funkcji reprezentującej szumy, funkcją liniową.



Rysunek 8: Po lewej funkcja pierwotna, po prawej aproksymacja funkcji pierwotnej funkcją liniową. Na szaro: pole, które wlicza się do szukanej całki.

Drugą równie ważną funkcją zaimplementowaną w programie jest metoda odpowiedzialną za wycinanie pików pochodzących od zakłóceń sieciowych 50 Hz. Funkcja ta wycina owy pik w przedziale $\pm 5\text{Hz}$, wycinane także są wszystkie harmoniczne aż do częstotliwości 600 Hz. Opcja ta jest konfigurowalna i istnieje możliwość jej wyłączenia. Przed każdym pomiarem/serią pomiarową program pyta użytkownika czy chce z niej skorzystać. Ostatnią ważną funkcją jest funkcja odpowiedzialna za wyliczanie odchylenia standardowego według wzoru:

$$S(\sigma) = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}}$$

gdzie x_i to całka uzyskana w i-tym pomiarze, a \bar{x} to średnia wartość całki z wszystkich pomiarów. n to oczywiście ilość pomiarów w serii pomiarowej.

Dzięki temu istnieje możliwość kontroli każdej serii, i w razie konieczności można odrzucać serie ze zbyt dużym odchyleniem.

Program realizuje szereg opcji dodatkowych, które ustawiane są automatycznie po załadowaniu pliku konfiguracyjnego. Budowa tych funkcji jest dość prosta, a odpowiedzialne są one, jak było to wspomniane na przykład za ustawienie pasma w jakim prowadzony będzie pomiar. W ogólności schemat funkcji wygląda tak jak przedstawiono poniżej.

```
bool hp4195a::setAnalyze( int type )
{
    {
        ostreamstream cmd;
        cmd.setf(ios::fixed);
        cmd.precision(5);
        cmd << "FNC " << type;
        cout << "Tryb pracy: ";
        if( type == 1) cout << color(YELLOW) << "NETWORK \n" <<color();
        if( type == 2) cout << color(YELLOW) << "SPECTRUM \n" <<color();

        return wr(cmd.str());
    }

return true;
}
```

Zaprezentowany kod dotyczy ustawianego trybu analizy, ale schemat dla większości funkcji jest podobny. Rozkazy przesyłane są jako odpowiednio modyfikowane stringi (ustawiana jest flaga precyzji etc. jest to konieczne gdyż ze zwykłym stringiem urządzenie czasami nie pracowało poprawnie) za pomocą metody z dziedziczonej klasy gpibdev, czyli **wr(cmd)**. Podobnie tworzone są funkcje odczytujące żądane wielkości z tym, iż korzystamy z funkcji **query(cmd)**, która jak było to już wspomniane żąda od urządzenia zwrotu wybranej wielkości.

Na koniec tej części należy wspomnieć o programie sterującym. Został on wyposażony w dwa parametry z którymi może zostać wywołany, tymi parametrami są:

-**l** nazwa pliku konfiguracyjnego

Powoduje załadowanie z pliku konfiguracyjnego zapisanych w nim ustawień.

-**p** ilość powtórzeń

Rozpoczyna pomiar z zadaną ilością powtórzeń (dla analizy widmowej domyślnie wartość powtórzeń ustawiona jest na jeden).

Program sterujący nie jest skomplikowany, ale to było założeniem, aby służył on tylko do wywoływania żądanych metod. Wszystkie funkcje dostępne w programie opisane są w dodatku B.

3.1.2 Procedury testowe

Gdy program został stworzony należało określić procedury testowe, które dały by odpowiedź, czy wszystko dobrze zostało zaimplementowane. Pierwszą procedurę sprawdzającą poprawność programu, było badanie przebiegu sinusoidalnego generowanego na generatorze, o zadanej częstotliwości i amplitudzie. Zgodnie z teorią, RMS (wartość skuteczna) takiego przebiegu może być wyznaczony według zależności:

$$RMS = \frac{\text{Amplituda}}{\sqrt{2}}$$

Badano przebiegi sinusoidalne o dwóch amplitudach 25mV(50mV pik-pik) i 50 mV(100mV pik-pik) przy częstotliwości 300kHz, dla których wartości skuteczne to odpowiednio $RMS_{25mV} = 17.7mV$ i $RMS_{50mV} = 35.5mV$. Poniżej zaprezentowano uzyskane wyniki z analizatora w porównaniu do wyników teoretycznych.

Amplituda [mV]	Analizator [mV]	RMS teoretyczne [mV]
25	18.8	17.7
50	37.2	35.5

Tablica 4: Porównanie wyników z analizatora z wynikami teoretycznymi dla dwóch przebiegów sinusoidalnych o różnych amplitudach.

Z zaprezentowanych wyników widać, iż wynik uzyskany na analizatorze jest większy niż teoretyczny, ale tego należało się spodziewać ponieważ sam analizator posiada swoje szумы, które dają pewien przyczynek do wyniku. Po za tym zastosowany generator także nie jest urządzeniem idealnym i generowany przebieg sinusoidalny, posiada pewne niedoskonałości. Tak więc uznać należy, iż napisany program przeszedł test pomyślnie i wyniki pokrywają się w spodziewanych granicach z wynikami oczekiwanymi.

Drugą formą testowania programu, było porównanie wyników z analizatora, z wynikami jakie uzyskiwano na analogowym mierniku szumów, HP3400. Porównywane były wyniki, jakie uzyskiwano podczas właściwych pomiarów układu Front-End, o którym szerzej będzie w części pracy **Wyniki**. Miernik analogowy, pracuje w dedykowanym paśmie 10 Hz-10 Mhz, natomiast analizatorem szумы mierzone były najczęściej w paśmie 1 Hz-500 MHz, spodziewano się więc,

iż wynik z analizatora będzie większy niż ten z miernika analogowego. Zazwyczaj tak było, ale zwrócić należy uwagę na fakt, iż analizator obcinał piki od sieci, co czasami obniżało znacząco wartości mierzonych szumów. Dokładne porównanie wartości, jak było to wspomniane, zostaną przedstawione w części pracy, w której przedstawiono wyniki.

3.2 Program dla zasilacza HP6624A

Program dla zasilacza, był prostszy niż ten stworzony dla analizatora. Obejmuje on tylko część odpowiedzialną za ustawianie i odczytywanie nastawionych wartości napięcia/prądu. Z dodatkowych funkcji zaimplementowane zostały funkcje OVP (z ang. Over Voltage Protection) i OCP (z ang. Over Current Protection), które zapewniając ochronę podczas ustawiania wartości odpowiednio napięcia i prądu. Samo urządzenie pracuje na nowszej (rok 1992) wersji standardu GP-IB, co też miało znaczenie podczas pisania programu, gdyż nie było konieczności (tak jak w przypadku analizatora) modyfikacji niektórych metod z biblioteki gpibdev.h.

3.2.1 Dostępne funkcje

Jak wspomniane było we wstępie, zasilacz posiada standardowe funkcje ustawiania napięcia/prądu na cztery wyjścia, które posiada. Napięcie nastawiać można w zakresie 0 - 50V, a prąd 0-5A. Dodatkowo napisano funkcje do ustawiania OVP i OCP na wybranym kanale. Tak samo stworzone zostały funkcje do odczytu wartości napięcia/prądu jakie zostało ustawione na wyjściu, co jest przydatne podczas badania układu, który mógłby w jakiś sposób obciążyć wyjście zasilacza. Dzięki tym funkcjom cały czas możemy kontrolować faktyczną ustawioną wartość na wyjściu.

Schemat pisanych funkcji zaprezentowano poniżej.

```
bool hp6624::setCurrent( int chn, float current )
{
string cmd = "ISET "+StringUtils::IntToStr(chn)+", "+
              StringUtils::FloatToStr(current);

wr(cmd);
string c = query("ISET? "+StringUtils::IntToStr(chn));
cout << "Ustawiono: " << c << "A na kanale " << chn << ".\n";
return true;
}
```

Funkcja karze ustawić prąd na wybranym kanale, a następnie żąda zwrotu ustawionej wartości. Funkcje IntToStr(int), i FloatToStr(float), pochodzą z biblioteki pomocniczej String Utils, natomiast sam kanał wybierany jest w pliku odpowiedzialnym za sterowanie klasą, program pyta użytkownika po wybraniu konkretnej opcji o kanał.

3.2.2 Procedury testowe

Urządzenie nie wymagało skomplikowanych procedur testowych. W ramach testu, najpierw sprawdzono czy wskazane napięcia/prądy są ustawiane na wybranych kanałach (poprzez odczyt z wbudowanego w urządzenie wyświetlacza LCD). Następnie zasilano urządzeniem testowy układ i sprawdzano, czy wszystko działa poprawnie.

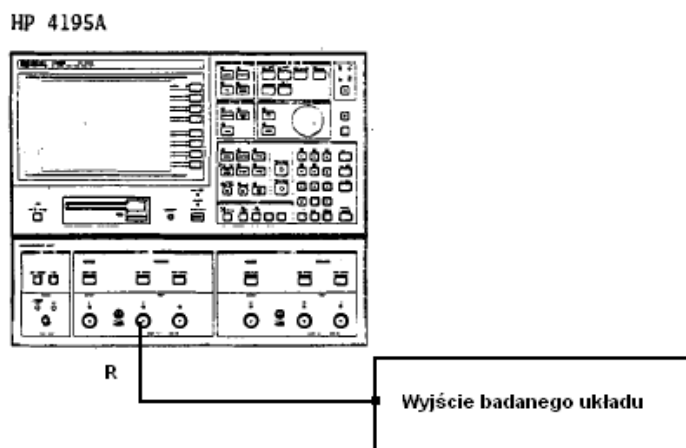
4 Wyniki pomiarów

W części tej zaprezentowane zostaną przykładowe pomiary, jakie udało się wykonać przy pomocy zaprogramowanych urządzeń. Są to wyniki przykładowe, gdyż same pomiary nie były głównym celem podczas tworzenia pracy.

Za pomocą samego analizatora zbadano szумы układu Front-End dostępnego w laboratorium, zbadano wartości dla dwóch wzmocnień (wysokie i niskie), a także dla szeregu różnych pojemności wejściowych C_{det} . Natomiast przy użyciu zasilacza i analizatora, zbadano odpowiedź częstotliwościową filtru aktywnego pasmowo przepustowego o zadanych wartościach R i C . Dodatkowo zbadano także dwa filtry pasywne, dolnoprzepustowy i górnoprzepustowy.

4.1 Wyniki pomiaru szumów analizatorem HP4195A

Szумы badane były w paśmie 1 Hz-500 MHz, przy automatycznie ustawianym RBW i atenuacji na ustawionym porcie 0dB. Serie testowe obejmowały 8 powtórzeń. Wszystkie pomiary wykonane zostały na porcie R1. Poniżej schemat podłączenia podczas pomiarów.



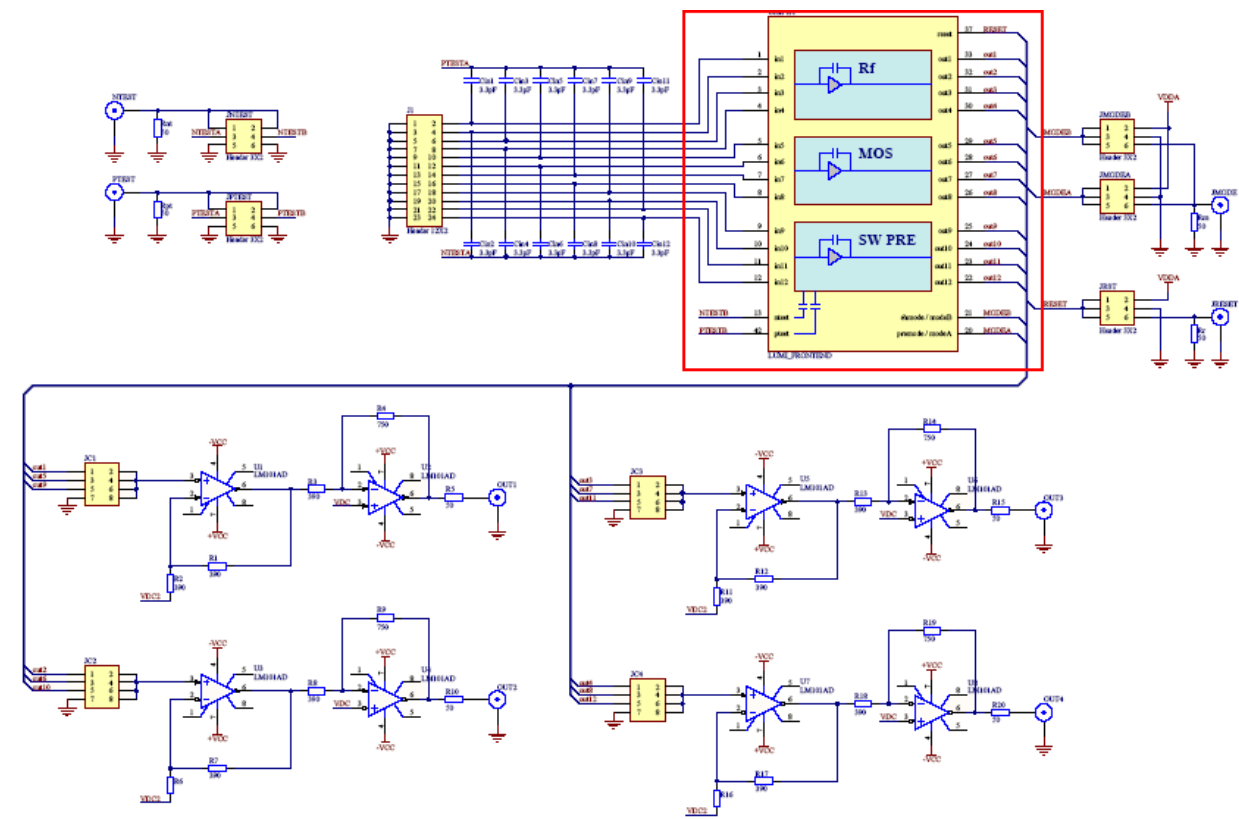
Rysunek 9: Sposób podłączenia analizatora podczas wykonywania pomiarów.

Dodatkowo sprawdzono poprawność wyników z kanału drugiego (port R2), aby sprawdzić, czy nie istnieją jakieś niepożądane rozbieżności wynikłe na przykład z wieku urządzenia. Okazało się, iż wyniki z portu drugiego są zbieżne z wynikami z portu pierwszego, i port drugi może z powodzeniem być wykorzystywany w pomiarach. Wspomnieć można także, o dostępnej funkcji, w której urządzenie pracuje przy niskich szumach własnych (funkcja High sensitivity). Po jej aktywacji szумы własne analizatora utrzymują się w granicach 0.2-0.25 mV, czyli są dwukrotnie mniejsze, niż w trybie pracy normalnej (w trybie pomiaru normalnego szумы własne to 0.4-0.45 mV). Sprawdzenie tej funkcji istotne było zwłaszcza podczas testowania układu Front-End z niskim wzmocnieniem, o czym będzie w dalszej części pracy.

W laboratorium dostępne były dwie płytki z układem Front-End oznaczone jako *board 1* i *board 2*. Zbadana została jedynie płytka druga, ponieważ na płytce numer jeden dolutowane zostały dodatkowe pojemności wejściowe, które miały wpływ na pomiary szumów. Przeprowadzone testowe pomiary wykazały, iż wpływ ten był znaczny, dlatego też zaniechano badania tej płytki.

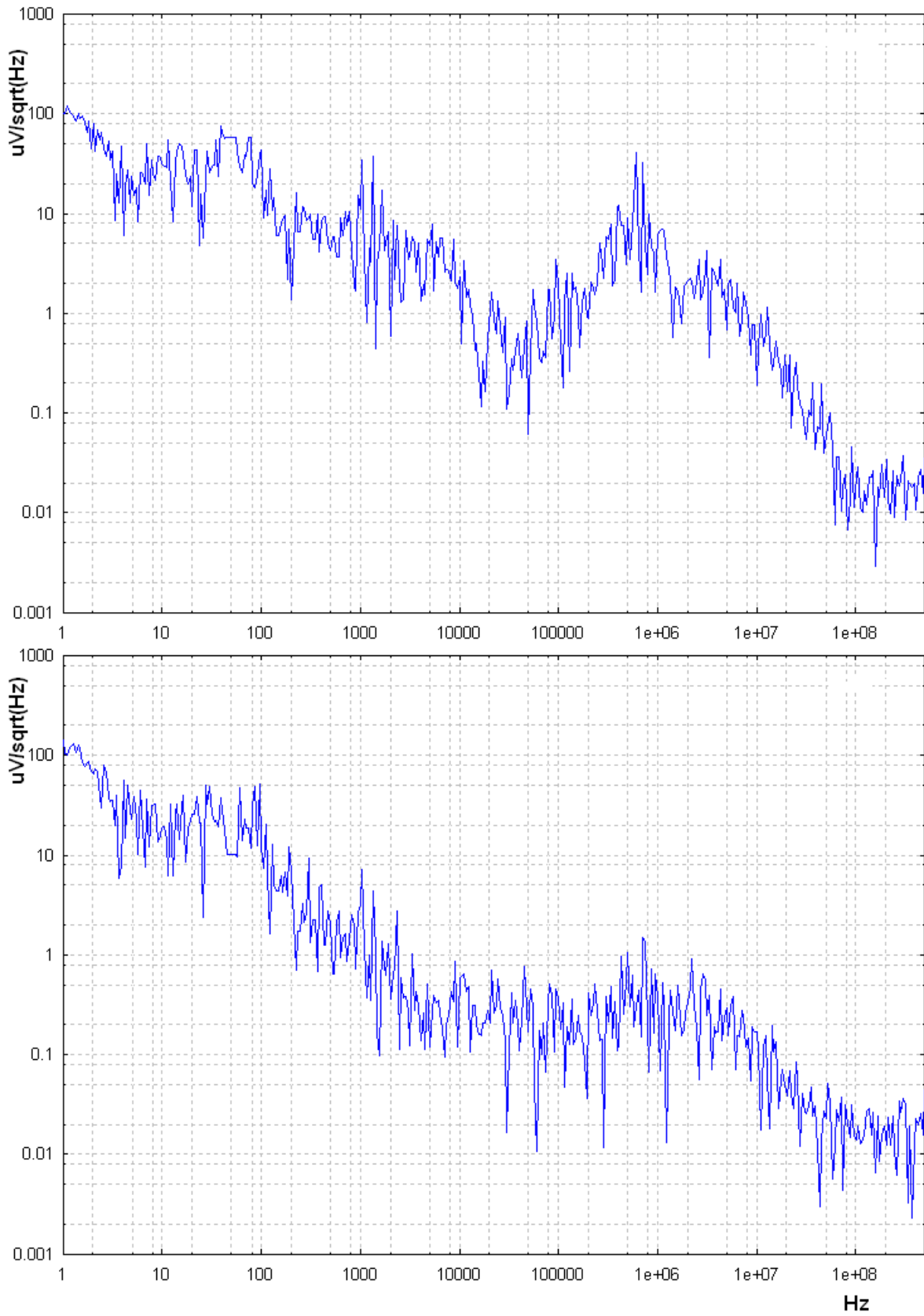
4.1.1 Wyniki dla układu Front-End board 2

Schemat układu.



Rysunek 10: Schemat układu Front-End. W ramce sam Front-End, dalej wtórnik i za nim wzmacniacz o wzmacnieniu 2, podpięte do każdego z wyjść Front-End'u

Zbadano szumy dla wysokiego i niskiego wzmacnienia elektroniki Front-Endu, dla pasywnego (R_f) sprzężenia zwrotnego i aktywnego (MOS) sprzężenia zwrotnego w funkcji zmiennej wartości pojemności wejściowych C_{det} . Zbadane zostało pięć pojemności wejściowych o wartościach odpowiednio 0pF, 22pF, 44pF, 66pF i 88pF. Natomiast ze względu na bardzo małe szumy układu przy małym wzmacnieniu, które wahały się w granicach 0.15-0.2 mV, niemożliwe było zbadanie ich analizatorem, którego szumy własne wynoszą około 0.4-0.45mV w trybie normalnym, i jak było to wspomniane we wstępie 0.2-0.25mV w trybie o wysokiej czułości. Nawet jednak w tym drugim trybie, jak widać z danych przytoczonych powyżej, szumy własne urządzenia miały wartość porównywalną z wartością szumów układu Front-End, co w oczywisty sposób wyklucza zastosowanie analizatora do poprawnych pomiarów tych wartości. Testowo sprawdzono jedynie wartości szumów dla trzech bardzo dużych, jak na badany układ, pojemności wejściowych C_{det} , czyli 220pF, 330pF i 470pF. Dalej przedstawiono wyniki widma szumowego uzyskanego za pomocą analizatora dla dwóch skrajnych pojemności wejściowych C_{det} , czyli 0pF i 88pF i dużym wzmacnieniu, przy pasywnym sprzężeniu zwrotnym.



Rysunek 11: *Widmo szumowe. U góry dla $C_{det}=88pF$, u dołu dla $C_{det}=0pF$.*

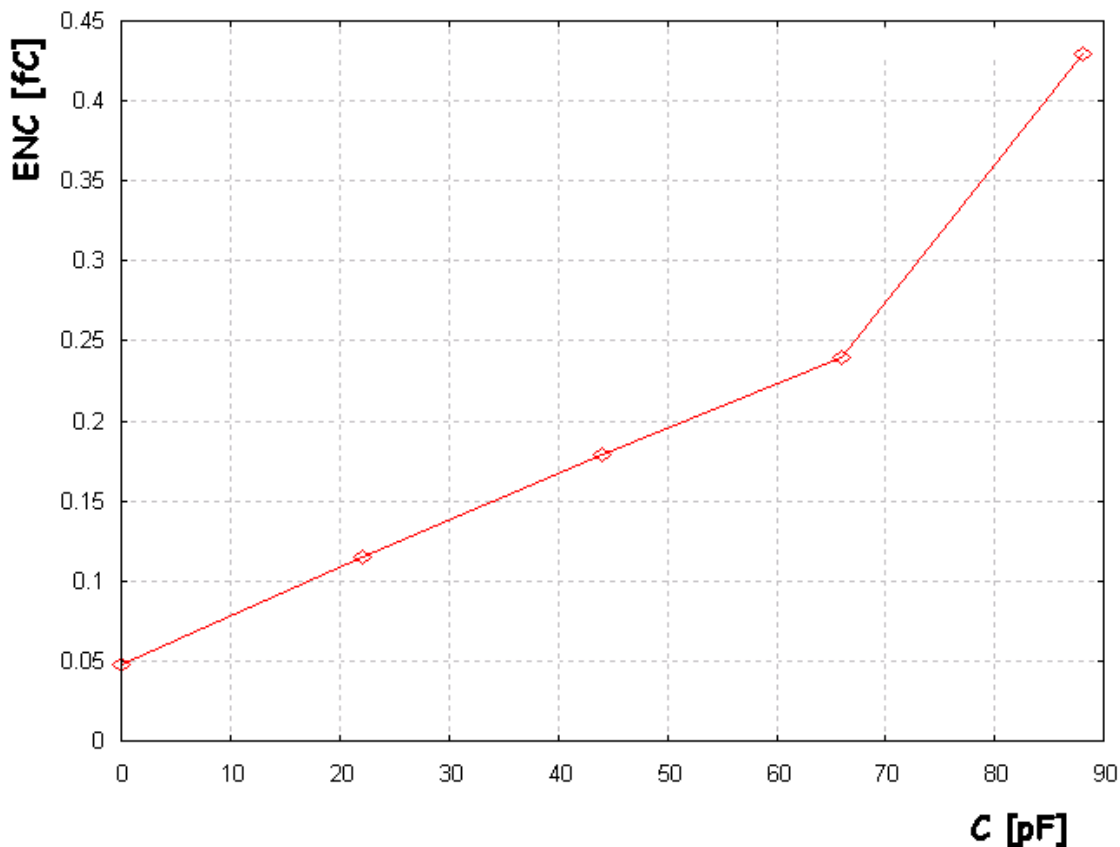
Po zebraniu wszystkich widm, wykreślono zależność ENC od pojemności wejściowych.

ENC(z ang. Equivalent Noise Charge, ekwiwalentny ładunek szumowy) - zgodnie z definicją jest to taki ładunek wprowadzony na wejściu, który daje odpowiedź napięciową równą średniej kwadratowej wartości napięcia szumów.

Przykładowo, dla układu badanego na wejściu pojemność zewnętrzna to 0.5pF , z generatora pochodził sygnał prostokątny o amplitudzie $V_{in} = 50\text{mV}$, następnie za pomocą oscyloskopu badana była amplituda wyjściowa V_{out} . Ładunek Q_{in} w takim przypadku, to $Q_{in} = V_{in} * 0.5\text{pF} = 25\text{pC}$. Korzystając z uzyskanej na analizatorze wartości szumów, i poprzez proste przekształcenie matematyczne, dostajemy w ogólności:

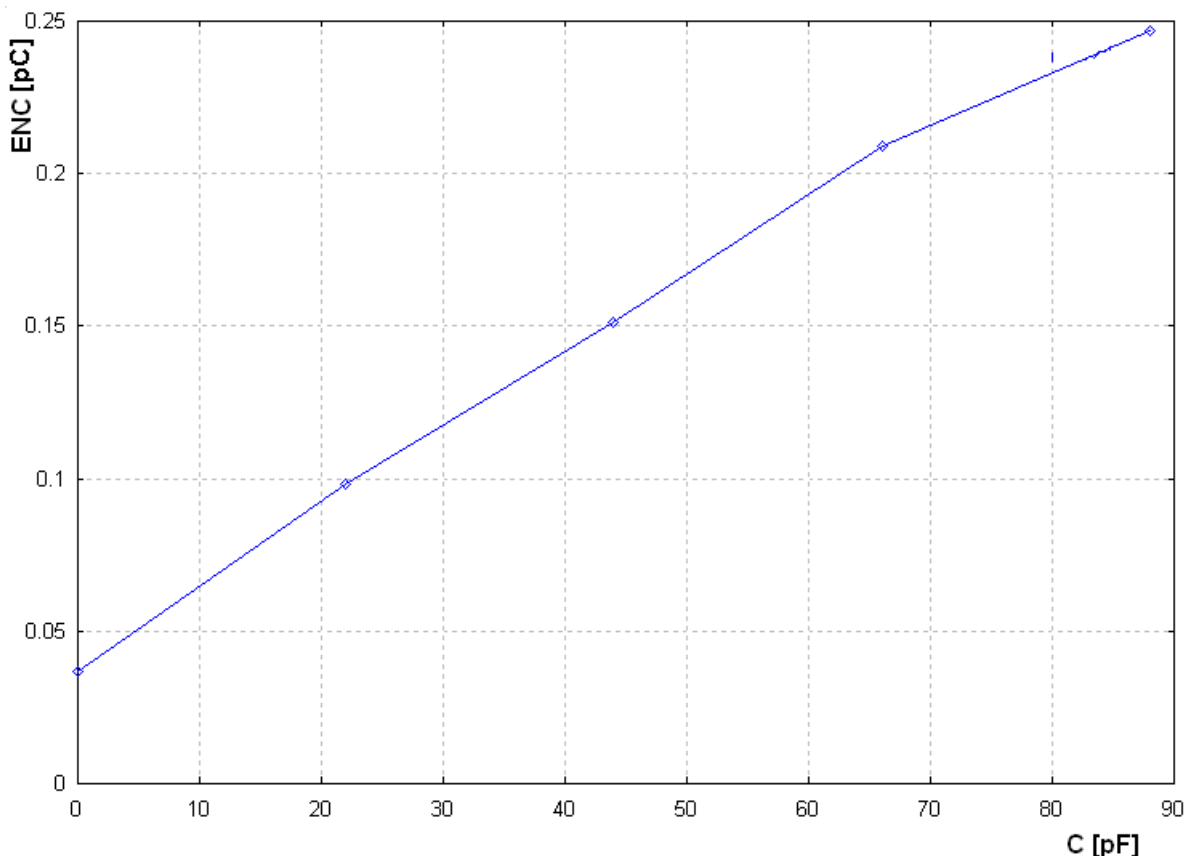
$$ENC = \frac{Q_{in} * \text{wart.szumow}}{\text{Amp.wyjciowa}}$$

W taki sposób obliczone zostały wszystkie wartości ENC. Następnie wykreślane były w funkcji zbadanych pojemności wejściowych C_{det} . Poniżej ENC, dla sprzężenia zwrotnego pasywnego R_f .



Rysunek 12: ENC dla R_f , wysokie wzmocnienie.

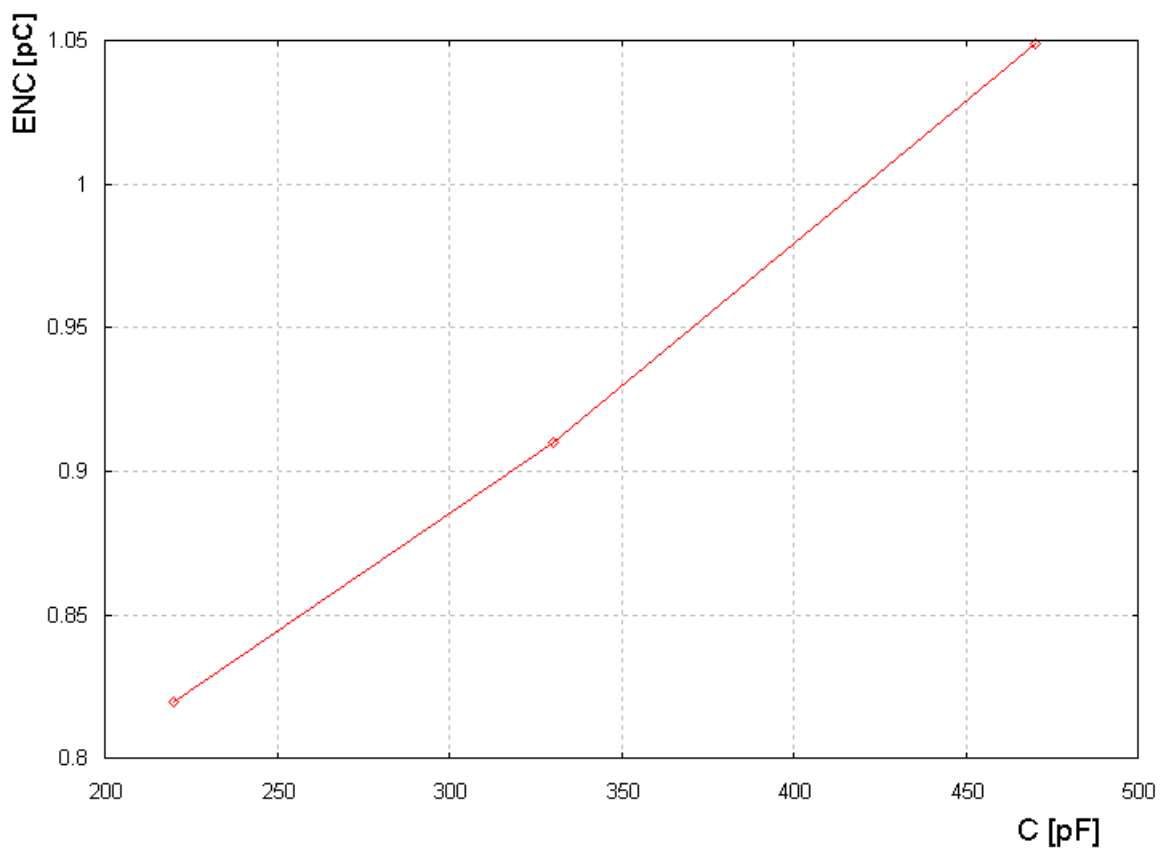
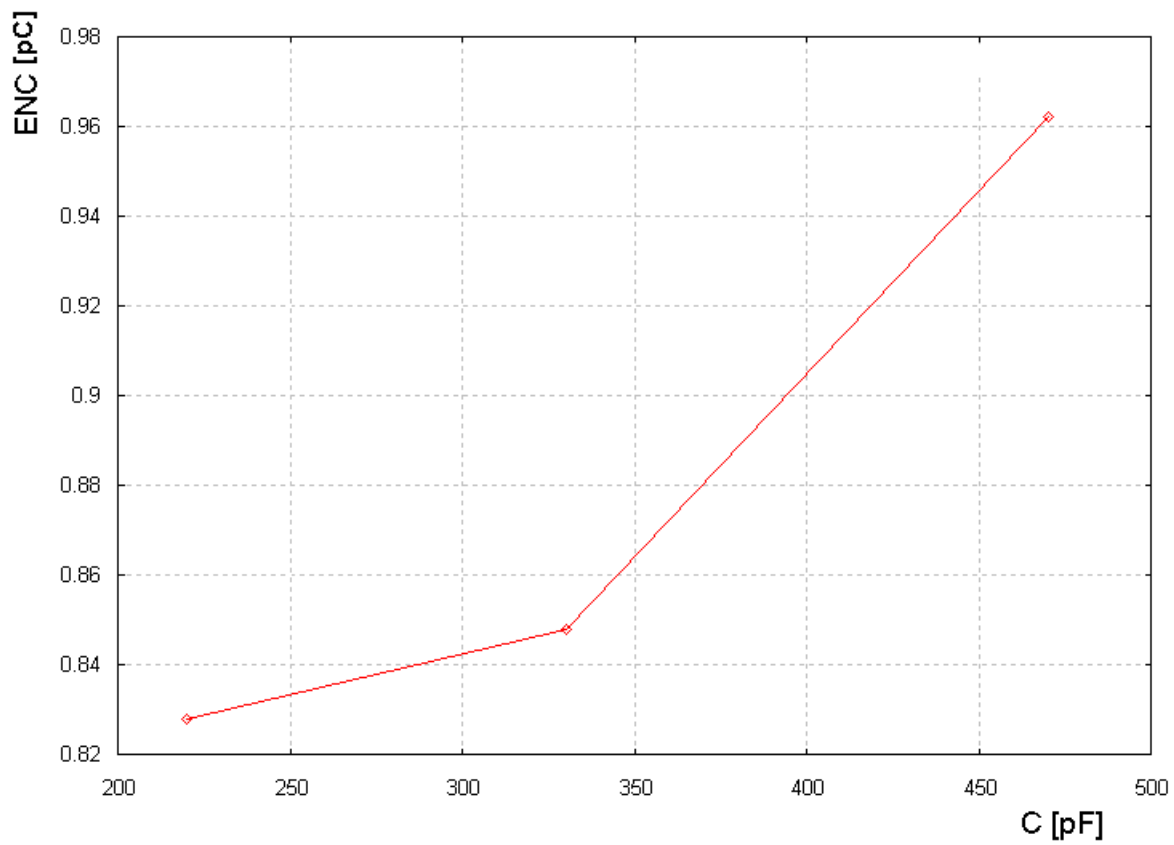
Dalej ENC, dla sprzężenia zwrotnego aktywnego MOS.



Rysunek 13: *ENC dla MOS, wysokie wzmocnienie.*

W dalszej części przedstawiono wyniki dla małego wzmocnienia układu Front-End. Jak było wspomniane wcześniej, są to jednak wyniki przykładowe, gdyż analizator posiada zbyt duże tło szumów własnych (nawet w trybie wysokiej czułości). Na poziomie takim samym, jak szумы pochodzące z badanego układu Front-End. Aby urządzenie było w stanie w pełni poprawnie zbadać widmo szumów dla małego wzmocnienia, jego szумы powinny być co najmniej o połowę mniejsze, niż te które posiada układ Front-End.

Dopiero przy bardzo dużych jak na badany układ pojemnościach, czyli powyżej 200pF, szумы generowane przez układ, są na tyle duże, iż da się je w miarę sensownie zbadać analizatorem. Tu jednak pojawia się problem poprawnej pracy układu, gdyż badana elektronika, zaprojektowana została do pracy z pojemnościami wejściowymi w granicach 100pF. Dlatego też badanie tak dużych pojemności nie gwarantuje nam poprawnej pracy układu. Przedstawione wyniki wykonane zostały dla pojemności wejściowych C_{det} o trzech wartościach kolejno 220pF, 330pF, 470pF.



Rysunek 14: *ENC*. U góry dla R_f , u dołu dla MOS, małe wzmocnienie.

Cały czas wyniki były porównywane z wynikami z analogowego miernika HP3400. Poniżej przedstawiono krótkie porównanie wyników pomiędzy analizatorem, a miernikiem HP3400, dla kilku przykładowych wielkości pojemności wejściowych Cdet.

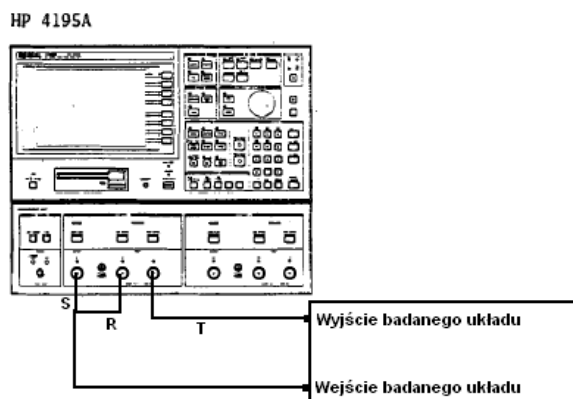
Urządzenie	0 [pF]	22 [pF]	44 [pF]
HP3400	1.65mV	4.5mV	6.5mV
HP4195A	1.9mV	4.2mV	6.8mV

Tablica 5: Porównanie wyników z analizatora z wynikami miernika analogowego Hp3400, dla pasywnego sprzężenia zwrotnego i kanału drugiego.

Jak można zauważyć, raz wyniki na mierniku były mniejsze, innym razem większe. Fakt ten po części tłumaczyć może wycinanie przez analizator pików sieci, które miernik uwzględniał w swoich pomiarach. Pewną rolę na pewno odegrały także dość duże szумы analizatora, które szczególnie przy zerowej wartości pojemności wejściowej Cdet, mogły także mieć znaczenie. Czynnikiem, który cały czas próbowano wyeliminować był wpływ wszystkich urządzeń, tych w laboratorium i pomieszczeniach obok, które jak zauważono poprzez wspólną sieć, wpływały na pomiary. Niestety mimo usilnych prób nie da się ze 100% pewnością stwierdzić, iż na wykonane pomiary wpływu nie miały te zakłócenia. Aby być w pełni pewnym niezależności uzyskanych wyników potrzeba by podczas pomiarów zastosować niezależne źródło zasilania, czego z oczywistych względów nie udało się zapewnić. Mimo tych obaw pomiary głównie przeprowadzane były w godzinach porannych (kiedy to w laboratorium nie było innych pracowników, wszystkie inne urządzenia były wyłączone), a także cały czas istniała kontrola wyników pomiędzy analizatorem, a miernikiem, co pozwala wnioskować, iż wspomniany wpływ nie był znacząco duży (jakkolwiek, nie da się go całkowicie wykluczyć).

4.2 Wyniki pomiaru widma częstotliwościowego analizatorem HP4195A i zasilaczem HP6624A

Analizator pozwala mierzyć widmo częstotliwościowe w zadanym wcześniej w pliku konfiguracyjnym paśmie, sam sposób podłączania urządzenia podczas takich pomiarów zaprezentowano poniżej.



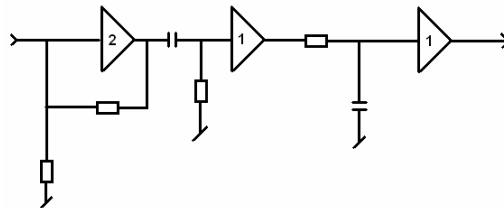
Rysunek 15: Sposób podłączania analizatora podczas wykonywania pomiarów widma częstotliwościowego.

Widmo zbierane jest w żądanym paśmie z określoną dokładnością. Na wyjściu S analizatora możliwe jest ustawienie zadanego wymuszenia (Opcja OSC w pliku konfiguracyjnym).

Program zapamiętuje uzyskane widmo w pliku o podanej przez użytkownika nazwie, co umożliwia późniejszą analizę uzyskanych danych.

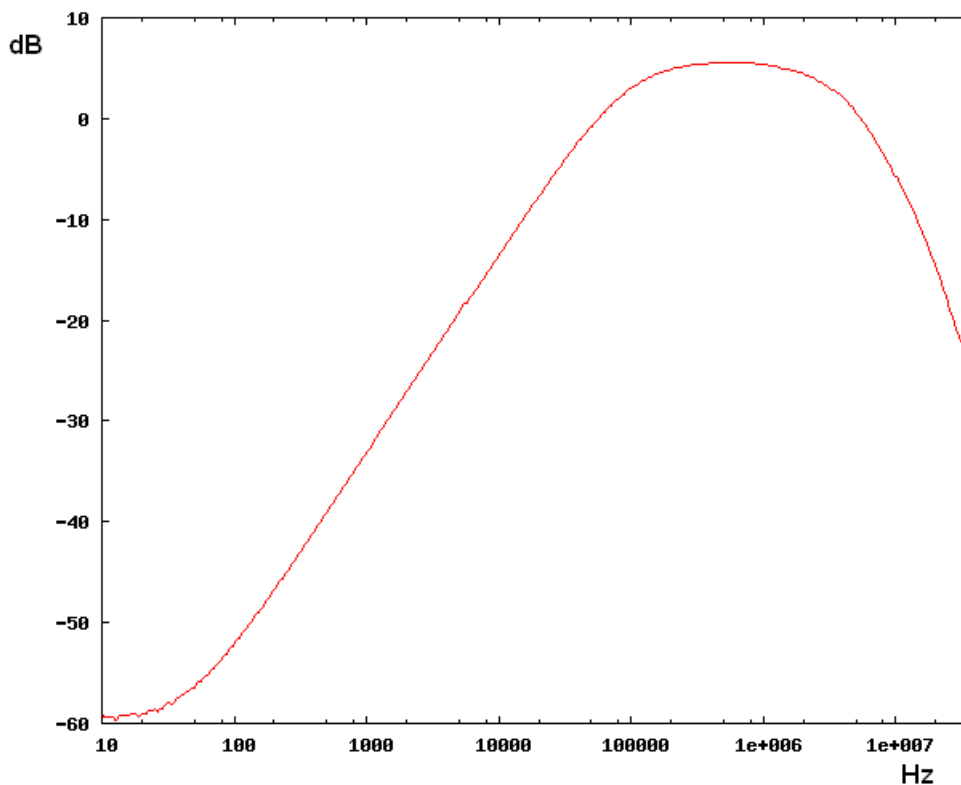
4.2.1 Wyniki dla filtra aktywnego pasmowo przepustowego

Filtrem aktywnym był filtr oparty o wzmacniacz operacyjny i znanym iloczynie RC. Dalej uproszczony schemat badanego układu i wyniki z programu gnuplot.



Rysunek 16: *Uproszczony schemat filtra aktywnego.*

Na schemacie nie ma podanych wartości elementów, gdyż na badanym układzie (pochodzącym z laboratorium dydaktycznego elektroniki Wydziału FilS) dostępna była jedynie informacja o iloczynie $C * R$ i $R * C$.

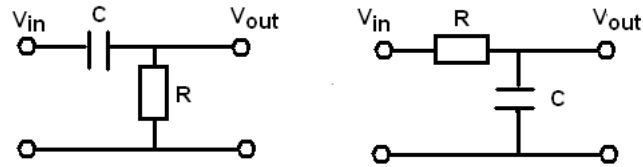


Rysunek 17: *Widmo częstotliwościowe dla filtra aktywnego.*

Łatwo można wyliczyć, korzystając ze wzoru $f = \frac{1}{2 * \pi * RC}$, iż częstotliwość graniczna od dołu to $f = 625 \text{ kHz}$, od góry to $f = 19.250 \text{ MHz}$, co dość dobrze zgadza się z otrzymanym widmem częstotliwościowym z analizatora.

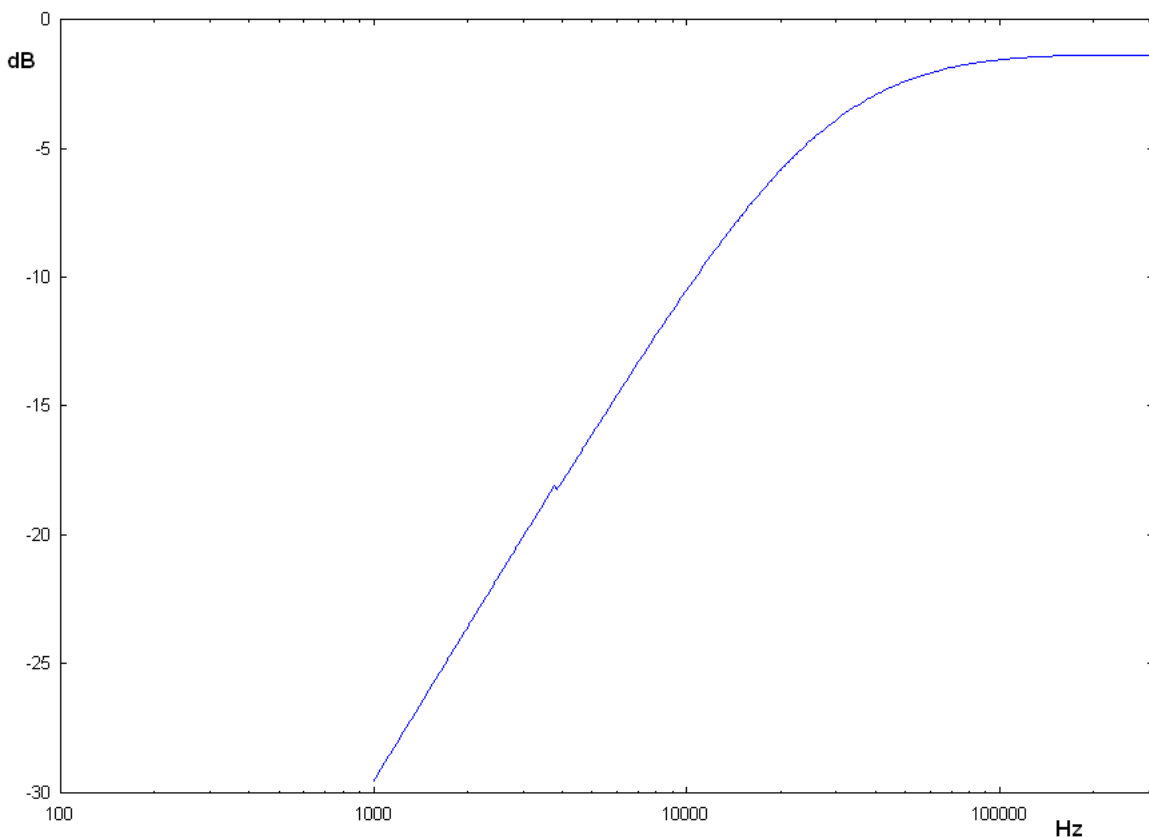
4.2.2 Wyniki dla filtrów pasywnych

Zbadano dwa filtry pasywne, dolno i górnoprzepustowy. Filtr górnoprzepustowy składał się z kondensatora o wartości $0.1 \mu F$ i opornika 50Ω (rezystancja wejściowa portu R1 analizatora). Natomiast filtr dolnoprzepustowy składał się z kondensatora o wartości $0.1 \mu F$ i opornika 100Ω . Schemat przedstawiono poniżej.



Rysunek 18: Po lewej filtr górnoprzepustowy, po prawej dolnoprzepustowy.

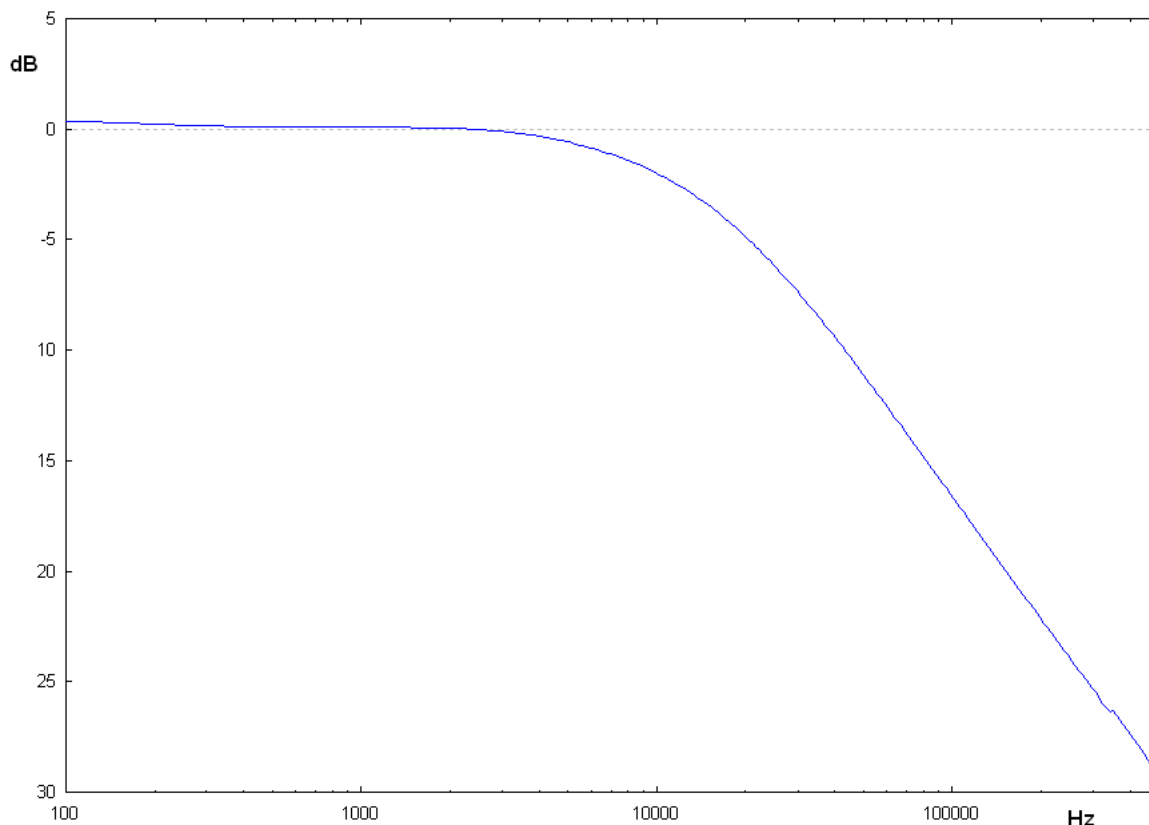
Odpowiedź dla filtru górnoprzepustowego.



Rysunek 19: Widmo częstotliwościowe. Filtr górnoprzepustowy.

Podobnie jak dla filtru aktywnego, łatwo wyliczyć można, iż częstotliwość graniczna liczona ze wzoru $f = \frac{1}{2*\pi*RC}$, to $f = 32\text{kHz}$. Dodatkowo sprawdzić możemy (z danych pomiarowych dostępnych w pliku wyjściowym), iż początkowe nachylenie na dekadę to spodziewane 20dB.

Odpowiedź dla filtru dolnoprzepustowego.



Rysunek 20: *Widmo częstotliwościowe. Filtr dolnoprzepustowy.*

Tu także wyliczono teoretycznie wartość częstotliwości granicznej $f = 16\text{kHz}$. Podobnie jak dla filtru górno przepustowego, tak i tu na dekadę przypada 20dB w górnym obszarze pasma.

Podsumowując, analizator i napisany program w łatwy sposób umożliwiają analizę i ściąganie widm częstotliwościowych. Co dalej w prosty sposób umożliwia ich analizę, odnośnie chociażby nachylenia na dekadę, czy wykreślenia uzyskanych widm w programach do analizy danych (na przykład gnuplot).

5 Podsumowanie

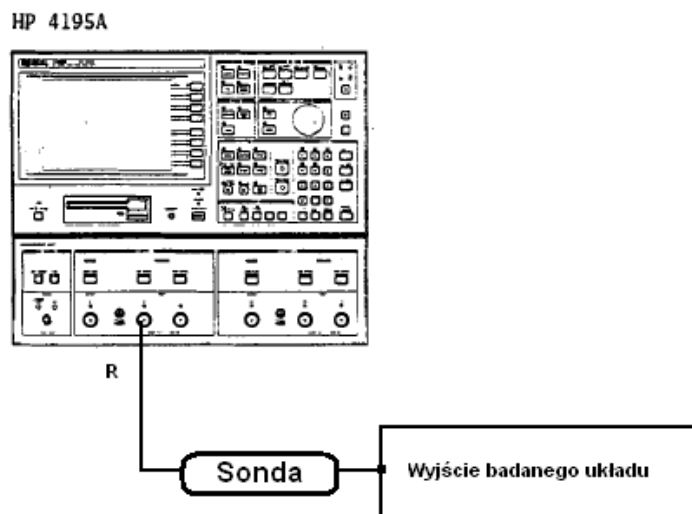
W pracy przedstawione zostały programy, które stworzone zostały do wskazanych urządzeń. Omówiono standard GP-IB, z którego korzystają oba wspomniane urządzenia, jego historię, strukturę fizyczną, a także aspekty programowe. Opisano budowę programów, ich funkcje i sposób realizacji ważniejszych zagadnień. Przedstawiono procedury testowe, które stosowane były podczas sprawdzania poprawności oprogramowania. Pokazano próbne wyniki, uzyskane za pomocą napisanego oprogramowania.

Do pracy dołączony zostaje kod w języku C++, całego napisanego oprogramowania, dostępny na dysku CD-ROM.

6 Dodatek A

6.1 Testy sondy dołączonej do analizatora

Do analizatora HP4195a dołączona została sonda o wysokiej impedancji wejściowej, wynoszącej $100\text{k}\Omega$ (sam analizator posiada rezystancję wejściową 50Ω). Ważne było sprawdzenie, czy sonda ta nadaje się do pomiarów szumów. W tym celu podpięto sondę do analizatora, tak jak to pokazano na rysunku poniżej.



Rysunek 21: *Sposób podpinania sondy podczas pomiarów szumowych.*

I przeprowadzono przykładowe pomiary testowe, o procedurze takiej jak dla analizatora bez sondy, czyli zmierzono generowany przebieg sinusoidalny o zadanej amplitudzie. Przykładowo, dla przebiegu sinusoidalnego o amplitudzie 50 mV (pik-pik 100 mV), analizator z podpiętą sondą podawał zmierzoną wartość jako 38.5 mV , co było wyższe o około 3 mV od wartości teoretycznej (35.5 mV). Także pomiary układu Front-End, wskazywały, iż sonda jest urządzeniem aktywnym podczas pomiarów szumowych, i sama powoduje generację dodatkowych zakłóceń, których źródła jednakże nie udało się dokładnie ustalić. Wniosek jednak był taki, iż do badania wskazanych układów sonda nie nadaje się i jeje zastosowanie zniekształca wyniki.

7 Dodatek B

7.1 Spis dostępnych funkcji dla napisanych programów

Spis dostępnych funkcji w programie dla analizatora HP4195A:

Funkcja	Dopuszczalne wartości	Opis
ANALYZE	0-1	typ analizy
ATT1	0-40dB	atenuacja na porcie T1
ATT2	0-40dB	atenuacja na porcie T2
ATR1	0-40dB	atenuacja na porcie R1
ATR2	0-40dB	atenuacja na porcie R1
START	0.1Hz-500MHz	częstotliwość początkowa
STOP	0.1Hz-500Mhz	częstotliwość końcowa
PORT	0-4	port pomiarowy
VIDEO	0-1	ustawienie filtracji video
OSC	-20-15dBm	ustawienie wymuszenia

Spis dostępnych funkcji w programie dla zasilacza HP6624A:

Funkcja	Dopuszczalne wartości	Opis
VOLT	0 - 50 V	ustawienie napięcia na wybranym kanale
CURRENT	0 - 5A	ustawienie prądu na wybranym kanale
OVP	0 - 50V	ustawienie maksymalnej wartości napięcia na wybranym kanale
OCP	0,1	ustawienie zabezpieczenia prądowego na wybranym kanale

8 Literatura

Z powodu specyficznego charakteru pracy głównymi źródłami z których korzystano podczas pisania, były podręczniki sprzętowe dostarczone wraz z urządzeniami, a także oficjalna dokumentacja standardu GP-IB. Resztę źródeł stanowiły podręczniki języka C++ i elektroniki, a także źródła internetowe (głównie strony firmy National Instruments, która stworzyła wykorzystywany w laboratorium kontroler urządzeń pracujących w standardzie GP-IB).

Książki:

1. Jerzy Grębosz - *Symfonia C++*
2. Jerzy Grębosz - *Pasja C++*
3. Kazimierz Korbel - *Układy elektroniki Front-End*

9 Spis rysunków

Spis rysunków

1	<i>Widok na przekrój przewodu GP-IB, na czerwono zaznaczone linie odpowiedzialne za kontrolę rozkazów, na zielono linie danych, pozostałe to linie masy.</i>	5
2	<i>Przykład struktury połączenia, po lewej połączenie liniowe, po prawej gwiazdźście.</i>	6
3	<i>Schemat szerszej struktury, z siecią zawierająca więcej niż jeden kontroler.</i>	6
4	<i>Schemat blokowy, reprezentujący obsługę zdarzeń i błędów w GP-IB.</i>	8
5	<i>Struktura plików i dziedziczenia dla napisanych programów.</i>	9
6	<i>Wycinek pliku konfiguracyjnego dla HP4195A.</i>	10
7	<i>Wycinek pliku wyjściowego dla HP4195A.</i>	10
8	<i>Po lewej funkcja pierwotna, po prawej aproksymacja funkcji pierwotnej funkcją liniową. Na szaro: pole, które wlicza się do szukanej całki.</i>	10
9	<i>Sposób podłączania analizatora podczas wykonywania pomiarów.</i>	14
10	<i>Schemat układu Front-End. W ramce sam Front-End, dalej wtórnik i za nim wzmacniacz o wzmacnieniu 2, podpięte do każdego z wyjść Front-End'u</i>	15
11	<i>Widmo szumowe. U góry dla $C_{det}=88pF$, u dołu dla $C_{det}=0pF$.</i>	16
12	<i>ENC dla R_f, wysokie wzmacnienie.</i>	17
13	<i>ENC dla MOS, wysokie wzmacnienie.</i>	18
14	<i>ENC. U góry dla R_f, u dołu dla MOS, małe wzmacnienie.</i>	19
15	<i>Sposób podłączania analizatora podczas wykonywania pomiarów widma częstotliwościowego.</i>	20
16	<i>Uproszczony schemat filtra aktywnego.</i>	21
17	<i>Widmo częstotliwościowe dla filtra aktywnego.</i>	21
18	<i>Po lewej filtr górno przepustowy, po prawej dolnoprzepustowy.</i>	22
19	<i>Widmo częstotliwościowe. Filtr górno przepustowy.</i>	22
20	<i>Widmo częstotliwościowe. Filtr dolnoprzepustowy.</i>	23
21	<i>Sposób podpinania sondy podczas pomiarów szumowych.</i>	24

10 Spis tabel

Spis tablic

1	<i>Linie do obsługi rozkazów. (*) Controller-In-Charge, czyli kontroler podrzędny. (**) System Controller, czyli kontroler główny(najczęściej karta PC w komputerze).</i>	5
2	<i>Podstawowe komendy standardu GP-IB, wprowadzone do normy IEEE 488.2</i>	7
3	<i>Podstawowe metody wykorzystywane podczas pisania programów, pochodzące z biblioteki gpibdev.</i>	9
4	<i>Porównanie wyników z analizatora z wynikami teoretycznymi dla dwóch przebiegów sinusoidalnych o różnych amplitudach.</i>	12
5	<i>Porównanie wyników z analizatora z wynikami miernika analogowego Hp3400, dla pasywnego sprzężenia zwrotnego i kanału drugiego.</i>	20