

Systemy pomiarowe

SYSTEMY POMIAROWE

ANDRZEJ MAJKOWSKI, REMIGIUSZ RAK

SYSTEMY POMIAROWE, ROZPROSZONE SYSTEMY POMIAROWE, WIRTUALNY PRZYZRZĄD POMIAROWY, WIRTUALNE LABORATORIUM.

Systemy informacyjno-pomiarowe stanowią obecnie bardzo ważny obiekt zainteresowań w kręgach metrologów zajmujących się projektowaniem i użytkowaniem nowoczesnych laboratoriów badawczych i dydaktycznych. Wśród pojęć funkcjonujących w tej dziedzinie do tej pory zalicza się systemy pomiarowe, systemy pomiarowo-kontrolne (systemy kontrolno-pomiarowe), a także systemy pomiarowo-sterujące. Pewne zamieszanie wprowadzone zostało wskutek złej interpretacji określenia w języku angielskim: Measurement & Control Systems. Słowo „control” oznacza nie tylko kontrolę, ale także sterowanie. Najnowsze spojrzenie na techniki pomiarowe, skłania jednak do używania określenia systemy informacyjno-pomiarowe, jako że nowoczesne systemy pomiarowe znalazły się w obszarze bardzo silnego oddziaływania ze strony technologii informacyjnej (ICT – Information and Communication Technology). Systemy te mocno osadzone są w sieciach komputerowych zaś dane pomiarowe (informacje pomiarowe) traktowane są w nich tak jak wszystkie inne rodzaje informacji. Najważniejszym elementem nowoczesnego systemu informacyjno-pomiarowego jest tzw. wirtualny przyrząd pomiarowy.

Spis treści

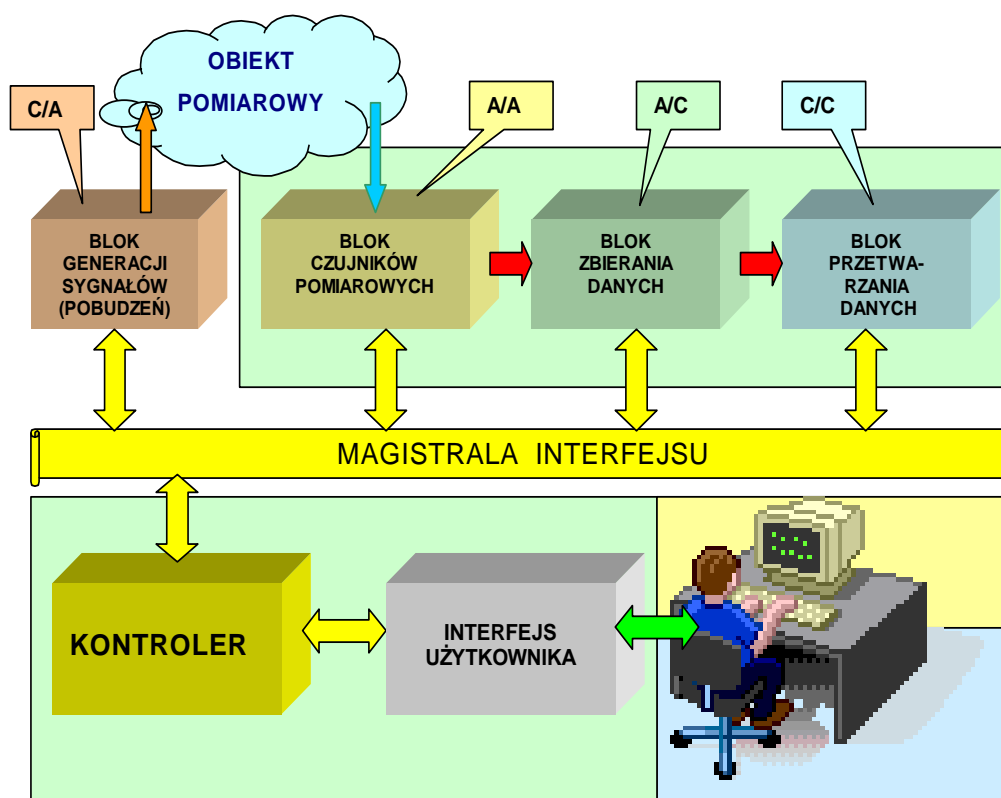
1.	Tradycyjny system pomiarowy	3
1.1.	Czujniki pomiarowe	4
1.2.	Blok zbierania danych	4
1.3.	Blok przetwarzania danych	4
1.4.	Blok generacji sygnałów	4
1.5.	Kontroler	4
1.6.	Blok komunikacji z użytkownikiem	5
1.7.	Interfejs	5
2.	Karty zbierania danych	6
2.1.	Parametry	6
2.2.	Budowa	6
2.3.	Schemat blokowy karty DAQ	7
3.	Wirtualne przyrządy pomiarowe	10
3.1.	Idea wirtualnego przyrządu pomiarowego	12
3.2.	Charakterystyka przyrządów wirtualnych	13
3.2.1	Kategorie przyrządów wirtualnych	13
3.2.2	Cechy przyrządów wirtualnych	13
3.3.	Konfiguracje przyrządów wirtualnych	15
3.3.1	Konfiguracje z użyciem kart zbierania danych	15
3.3.2	Konfiguracja z użyciem magistrali VXI (PXI)	16
3.3.3	Konfiguracja z użyciem interfejsu przyrządowego IEC-625	17
3.3.4	Konfiguracja z użyciem interfejsu szeregowego RS 232C oraz konfiguracje bezprzewodowe	18
3.3.5	Uogólniona struktura wirtualnego przyrządu pomiarowego	18
3.4.	Przykłady realizacji przyrządów wirtualnych	20
3.4.1	Przyrząd do badania parametrów filtrów aktywnych	20
3.4.2	Zestaw przyrządów wirtualnych	21
3.4.3	Cyfrowy analizator widma	22
4	Systemy rozproszone	25
4.1	Idea rozproszonego systemu pomiarowego	26
4.2	Cechy systemów rozproszonych	27
4.2.1	Dzielenie zasobów	27
4.2.2	Otwartość	27
4.2.3	Współbieżność	28
4.2.4	Skalowalność	28
4.2.5	Tolerowanie uszkodzeń	28

4.2.6	Przeźroczystość.....	29
4.3	Rozproszony system pomiarowy w sieci komputerowej	29
4.3.1	Charakterystyka ogólna	29
4.3.2	Przykładowa architektura systemu rozproszonego	31
4.4	Typowe protokoły komunikacyjne i obiekty sieciowe.....	33
4.5	Architektura klient/serwer	35
4.5.1	Przykład aplikacji sieciowej typu klient/serwer - rozproszony system plików.....	37
4.6	Architektura klient/serwer w systemie pomiarowym.....	37
4.7	Architektura peer-to-peer (P2P).....	38
4.8	Java jako programowa platforma dla systemu rozproszonego.....	39
4.8.1	Java a programowanie klient-serwer	39
4.8.2	Wątki w Javie	40
4.8.3	Wyjątki w Javie	40
5.	Wirtualne laboratorium.....	41
5.1.	Wstęp.....	41
5.2.	Idea wirtualnego laboratorium.....	42
5.3.	Założenia ogólne	43
5.4.	Architektura wirtualnego laboratorium	44
5.5.	Organizacja użytkowników i urządzeń w systemie.....	45
5.6.	Przykład zdalnego dostępu do laboratorium.....	46
6.	Ćwiczenia do modułu (rozwiązane problemy praktyczne - zadania, projekty).....	53
6.1.	Przyrząd wirtualny do konwersji sygnałów dźwiękowych.....	53
6.2.	Przyrząd wirtualny realizujący filtrację cyfrową	55
6.3.	Przyrząd wirtualny do czasowo-częstotliwościowej analizy sygnałów	58
7.	Pytania kontrolne	61
8.	Bibliografia.....	64

1. Tradycyjny system pomiarowy

Funkcjonujące przez wiele lat w świadomości metrologów pojęcie systemu pomiarowego w ostatnich latach straciło nieco na aktualności, w odniesieniu do szeregu zastosowań praktycznych. Stało się tak za przyczyną lawinowego rozwoju szeroko rozumianej technologii informacyjnej (informatyka, telekomunikacja). W rezultacie tego rozwoju, pewne znaczące trendy rozwojowe w dziedzinie metrologii narzucone zostały przez informatyków. Wciąż jednak projektowanie czujników pomiarowych, opracowanie metod pomiarowych, opracowanie metod kalibracji przyrządów i układów pomiarowych, legalizacja przyrządów oraz tradycyjnie problematyka analizy błędów pomiarowych i określania niepewności pomiaru pozostają wyłączną domeną metrologów. Powstają tu nowe, śmiałe i oryginalne prace teoretyczne.

System pomiarowy¹ dotychczas określano jako odpowiednio sprzężony zbiór elementów (stąd pojęcie sprzęgu w miejsce interfejsu), stanowiących całość organizacyjną i objętych wspólnym sterowaniem, mający umożliwić jakościową i ilościową ocenę właściwości badanego obiektu pomiarowego. Na rysunku 1.1 zamieszczono uogólniony schemat funkcjonalny systemu pomiarowego, składającego się z bloków, realizujących jedną lub kilka elementarnych funkcji.



Rys. 1.1 Schemat funkcjonalny tradycyjnego systemu pomiarowego

Do podstawowych elementów takiego systemu należą bloki: czujników pomiarowych, zbierania danych, przetwarzania danych, interfejsu, kontrolera i komunikacji z użytkownikiem. Zgodnie z tradycją symbole A/A, A/C, C/A, C/C oznaczają odpowiednio przetwarzanie typu: *analog-analog*, *analog-cyfra*, *cyfra-analog* i *cyfra-cyfra*. Definicje oraz szczegółowe opisy poszczególnych bloków znaleźć można w bardzo wielu klasycznych publikacjach np. [5]. Zostaną one tutaj powtórzone w bardzo skróconej formie.

¹ W przypadku występowania dodatkowych funkcji wykraczających poza tradycyjne zadania pomiarowe często stosowana była również nazwa system pomiarowo-kontrolny.

1.1. Czujniki pomiarowe

Zadaniem czujnika pomiarowego jest przetworzenie wielkości fizycznej, charakteryzującej badany obiekt, do postaci elektrycznej (a także innej np.: optycznej) i przekazanie jej do aparatury pomiarowej. Czujniki pomiarowe można klasyfikować według wielu kryteriów. Przykładowym kryterium podziału może być rodzaj wielkości wejściowych (ciśnienie, temperatura, napięcie)² lub rodzaj sygnału wyjściowego (elektryczny, optyczny). Nowoczesne systemy pomiarowe są wyposażone w tzw. czujniki inteligentne - zintegrowane z układem wstępnego przetwarzania i standaryzacji sygnałów, co pozwala na poprawienie dokładności pomiarów i bardziej skuteczną eliminację zakłóceń. Mogą też mieć wbudowane standardowe interfejsy pomiarowe.

1.2. Blok zbierania danych

Blok zbierania danych służy do odczytywania sygnału, pochodzącego z czujnika pomiarowego. Realizuje on zwykle dwie podstawowe operacje: kondycjonowanie sygnału i przetwarzanie analogowo-cyfrowe.

W zakres kondycjonowania sygnału wchodzi: wzmacnianie, izolacja galwaniczna, filtracja analogowa, a więc funkcje umożliwiające dopasowanie parametrów sygnału do wymagań narzuconych przez wejście przetwornika analogowo-cyfrowego. Normalizacja ta jest realizowana za pomocą specjalizowanych układów analogowych (wzmacniacze, wzmacniacze z izolacją galwaniczną, filtry analogowe, itp.).

Przetwarzanie analogowo-cyfrowe może być zrealizowane w sposób bezpośredni (*napięcie–cyfra*) lub pośredni. W zależności od typu wielkości pośredniej w procesie formowania sygnału cyfrowego wykorzystuje się tu przetwarzanie *czas – cyfra* lub *częstotliwość – cyfra*. Warto w tym miejscu dodać, że obecnie, dzięki znacznemu postępowi technologicznemu, najbardziej powszechne jest użycie bezpośrednich przetworników analogowo-cyfrowych typu kompensacji wagowej.

1.3. Blok przetwarzania danych

Blok przetwarzania danych dokonuje cyfrowego przetwarzania sygnału pomiarowego. W zależności od złożoności obliczeniowej przyjętego algorytmu funkcje bloku przetwarzania danych może pełnić mikrokontroler lub komputer osobisty. W systemach wymagających przetwarzania w czasie rzeczywistym stosuje się zazwyczaj wydzielone bloki funkcjonalne z procesorami sygnałowymi lub wręcz systemy czasu rzeczywistego.

1.4. Blok generacji sygnałów

Blok generacji sygnałów wykorzystuje się do wytwarzania sygnałów pobudzających. Do generacji sygnałów analogowych wymaga on użycia przetwornika cyfrowo-analogowego (C/A). Dyskusyjną jest sprawa zamieszczenia na schemacie bloku generacji sygnałów sterujących elementami wykonawczymi obiektu pomiarowego. Zamieszczenie takiego bloku uzasadnia, coraz bardziej powszechne, używanie nazwy *system pomiarowo-sterujący*.

1.5. Kontroler

Kontroler to urządzenie sterujące pracą całego systemu pomiarowego, zgodnie z zadanym algorytmem działania. Do głównych zadań kontrolera należy ustalanie warunków pomiaru i organizacja przepływu informacji w systemie. W kontekście przyjętych założeń, co do klasy rozpatrywanych systemów, rolę kontrolera systemu

² W przypadku, gdy wielkość wejściowa jest sygnałem elektrycznym raczej użyjemy określenia przetwornik pomiarowy

najczęściej pełni komputer osobisty, pracujący pod nadzorem wielodostępnych i wielozadaniowych systemów operacyjnych. Jest to związane z dużą elastycznością takich systemów i możliwością integracji kontrolera z blokiem przetwarzania sygnałów. W zastosowaniach przemysłowych dominują jednak mikroprocesory jednoukładowe i mikrokontrolery.

1.6. Blok komunikacji z użytkownikiem

Blok komunikacji z użytkownikiem umożliwia interakcję użytkownika z kontrolerem, a przez to z całym systemem pomiarowym. Pozwala wprowadzać dane programujące do systemu, obserwować (podglądać) wyniki pomiarów, a także je odbierać i rejestrować. Funkcję tego bloku, może spełniać komputer osobisty z odpowiednim, przyjaznym dla użytkownika oprogramowaniem realizującym funkcję *graficznego interfejsu użytkownika* (GUI – *Graphical User Interface*).

1.7. Interfejs

Interfejs pomiarowy umożliwia przepływ informacji między różnymi elementami rozpatrywanego systemu. W celu połączenia urządzeń, pochodzących od różnych producentów, opracowano specjalne standardy, które definiują: parametry mechaniczne, parametry elektryczne, protokoły transmisji, kody i formaty danych.

Istnieje wiele różnych standardów interfejsu, które różnią się przeznaczeniem (przrządowe, modułowe) rodzajem transmisji (szeregowa, równoległa i szeregowo-równoległa), szybkością, maksymalną liczbą możliwych do podłączenia urządzeń i dozwoloną odległością pomiędzy elementami systemu. Do najbardziej popularnych należą interfejsy przrządowe szeregowe: RS-232C, RS-485, równoległy IEC-625 (GPIB) oraz modułowy VXI.

2. Karty zbierania danych

Powstanie kart zbierania danych (DAQ – Data Acquisition) było ze wszech miar naturalną i nieuniknioną konsekwencją dostosowania technik pomiarowych do rozwoju technologii komputerowej. Genezę ich powstania określa umieszczenie układów „wzbogacających” komputer na jednej, wspólnej płycie, bezpośrednio współpracującej z magistralą komputera. Umożliwia to łatwą komunikację z procesorem oraz szybkie transfery danych, zwłaszcza po zastosowaniu przerwań i bezpośredniego dostępu do pamięci. Moduł DAQ, zawiera: przetworniki A/C i C/A, dodatkowe porty WE/WY oraz dodatkowe układy zegarowe - TIMER. Takie rozwiązanie umożliwia korzystanie wyłącznie z zasobów sprzętowych modułu, co znakomicie ułatwia realizację oprogramowania z użyciem sterowników firmowych.

2.1. Parametry

Do najważniejszych parametrów kart zbierania danych należą: liczba kanałów (wejść analogowych), zakres amplitud sygnałów wejściowych, rodzaje wejść, wartość częstotliwości próbkowania, rozdzielczość przetwornika A/C, liczba wyjść analogowych. Liczba wejść obejmuje zarówno wejścia niesymetryczne (nieróżnicowe), jak i symetryczne (różnicowe). Wejścia nieróżnicowe mają wspólny punkt uziemiający. Używa się ich w przypadkach, gdy sygnały wejściowe mają stosunkowo duże poziomy (powyżej 1 V), a długości przewodów doprowadzających nie przekraczają 5m. W przypadku, gdy te wymagania nie są spełnione, należy użyć wejść różnicowych, umożliwiających eliminację zakłóceń równoległych (wspólnych). Zwykle buduje się je łącząc w pary wejścia niesymetryczne. Stąd liczba wejść różnicowych stanowi połowę liczby wejść nieróżnicowych. Możliwości konfiguracyjne wejść analogowych karty DAQ są zawsze szczegółowo opisane w instrukcji obsługi karty.

2.2. Budowa

Typowe rozwiązania funkcjonalne wielokanałowych kart zbierania danych zawierają pojedynczy układ przetwornika analogowo-cyfrowego z multiplekserem analogowym i pojedynczym wzmacniaczem pomiarowym. W związku z tym efektywna częstotliwość próbkowania jest odwrotnie proporcjonalna do liczby wykorzystanych kanałów. Typowe liczby kanałów wejściowych to: 8, 16, 32. Celem zwiększenia liczby kanałów można zastosować dodatkowe karty lub moduł multipleksera analogowego, jako moduł kondycjonowania sygnałów.

W standardowych rozwiązaniach stosuje się bezpośrednio przetworniki analogowo-cyfrowe typu kompensacji wagowej o czasie konwersji rzędu mikrosekund. Dzięki temu osiąga się częstotliwości próbkowania od setek kHz do pojedynczych MHz. Rozdzielczości przetworników analogowo-cyfrowych typowo zmieniają się w zakresie 12÷16 bitów. Typowe zakresy napięć wejściowych to: $0 \div +10V$, $0 \div +5V$, $-10V \div +10V$, $-5V \div +5V$. Wzmocnienia zmieniają się w sposób dekadowy: 1, 10, 100, 500 lub oktawowy: 1,2,4,8,16,...,256. Często, karty umożliwiają selektywny dobór niektórych parametrów dla poszczególnych kanałów. Dotyczy to również wspomnianego układu z pojedynczym wzmacniaczem, gdy jego wzmocnienie zmienia się w takt przełączania kanałów. Umożliwia to optymalne wykorzystanie pełnej skali przetwarzania przetwornika A/C, nawet przy bardzo dużej różnorodności w zakresie poziomu sygnałów. Ma to decydujące znaczenie z metrologicznego punktu widzenia, gdy chodzi o minimalizację błędów przetwarzania i utrzymanie możliwie dużej wartości stosunku sygnał/szum (SNR).

W zasadzie wszystkie dostępne karty DAQ wyposażone są ponadto w przetworniki cyfrowo-analogowe (nawet 2 szt.) oraz odpowiadające im wyjścia analogowe. Przykładowe zastosowanie tych układów to np. generacja

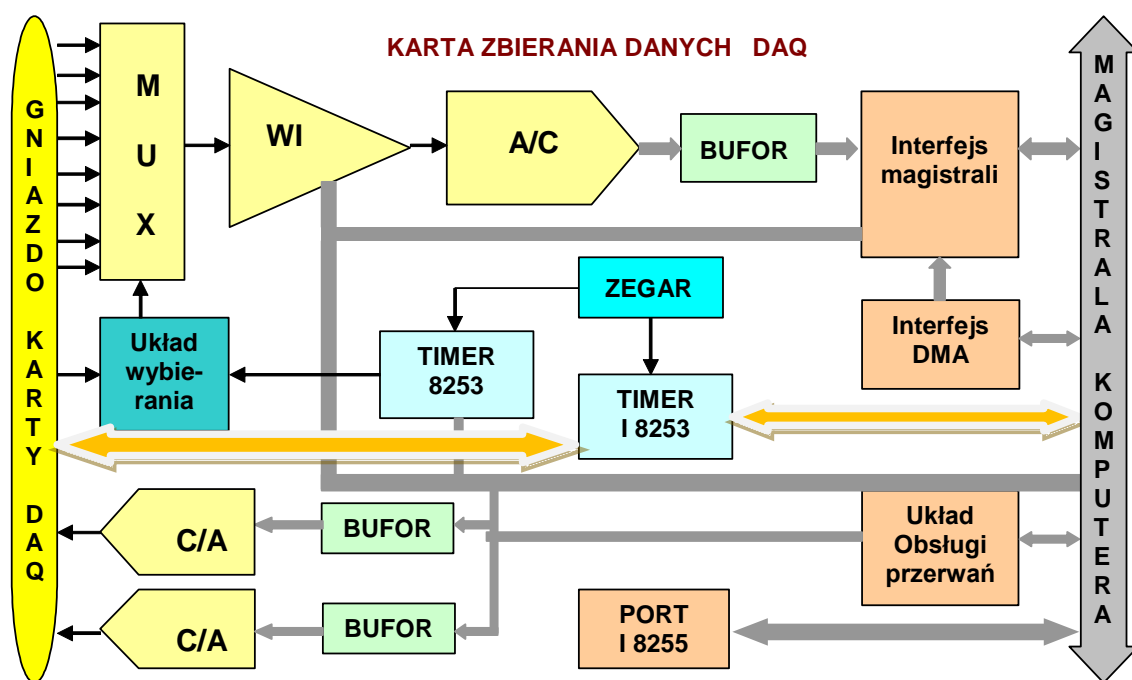
sygnałów pobudzających, o dowolnych parametrach, potrzebnych do wysterowania testowanych układów, czy też generacja sygnałów akustycznych w sytuacjach awaryjnych. Podstawowym parametrem przetwornika C/A jest szybkość ustalania napięcia wyjściowego, która limituje maksymalną wartość częstotliwości pojawiania się próbek sygnału na tym wyjściu. W standardowych rozwiązaniach układów wyjściowych karty DAQ nie ma oczywiście analogowych (odtworzących) filtrów dolnoprzepięsmych. Proces „zgrubnej” filtracji realizowany jest drogą przeciągania wartości amplitudy próbki na wzór próbkowania z pamięcią. Tworzy to obwiednię sygnału, którą można interpretować jako splot ciągu (nieskończenie krótkich impulsów) próbek sygnału z impulsem prostokątnym o jednostkowej amplitudzie i czasie trwania równoważnym okresowi wysyłania próbek sygnału. Istotę działania takiej filtracji najwygodniej jest opisać w dziedzinie widmowej, zastępując splot iloczynem odpowiedników widmowych. Dodatkowe użycie filtra dolnoprzepięsmych może nie być konieczne, jeżeli zadba się o wysłanie odpowiednio dużej liczby próbek w okresie zmienności sygnału. To pociąga za sobą jednak znaczne ograniczenia częstotliwości generowanych sygnałów.

Ponadto, jak już podkreślano wcześniej, karty tego typu są wyposażone w cyfrowe równoległe porty wejścia/wyjścia (Digital I/O), które mogą być (po stosownym oprogramowaniu) wykorzystane do specjalnych zastosowań np.: sterowanie procesem skojarzonym z systemem pomiarowym, komunikacja z urządzeniami peryferyjnymi itp. Typowy przypadek to użycie standardowego, dwukierunkowego portu firmy Intel typu 8255, zawierającego trzy rejestry 8-bitowe i oferującego trzy tryby pracy.

Ostatnią grupę stanowią układy zegarowe (Timing I/O). Układy te bywają bardzo pomocne w wielu zastosowaniach jak np.: pomiar częstotliwości powtarzania zdarzeń, pomiary parametrów przebiegów impulsowych, generacja pojedynczych impulsów czy przebiegów impulsowych. Wszystkie wymienione zastosowania mogą być zrealizowane na bazie trzech podstawowych elementów: *bramka*, *źródło*, *wyjście*, charakterystycznych dla układu 8253 firmy Intel (timer w systemie mikroprocesorowym).

2.3. Schemat blokowy karty DAQ

Schemat blokowy przykładowej (typowej) karty zbierania danych, zamieszczono na rys. 2.1.



Rys. 2.1 Uproszczony schemat blokowy typowej karty zbierania danych (WI-wzmacniacz instrumentalny, MUX-multiplexer, A/C-przetwornik a/c, C/A-przetwornik c/a)

Karta ta ma osiem analogowych wejść nieróżnicowych (cztery różnicowe), dwa 12-bitowe przetworniki C/A, port 8255 z trzema 8-bitowymi rejestrami (dwadzieścia cztery linie cyfrowych wejść/wyjść kompatybilnych z poziomami napięć TTL) i trzy 16-bitowe kanały zegarowe. Kartę łączy z komputerem 8-bitowa szyna danych. Istnieje możliwość wykorzystania przerwań i bezpośredniego dostępu do pamięci. Wczytywanie danych pomiarowych do pamięci komputera może się odbywać w trzech trybach: ciągłego pozyskiwania danych z jednego kanału, wielokanałowego pozyskiwania danych z ciągłym przełączaniem oraz wielokanałowego pozyskiwania danych z przełączaniem w określonych odstępach czasu. W tym ostatnim trybie wszystkie kanały są skanowane przez określony czas, po którym następuje zaprogramowana przerwa i powtórzenie całego cyklu. Skanowana może być dowolna liczba kanałów (od 2 do 8). Istnieją dwa sprzętowe tryby wyzwalania "pretriggering" i "posttriggering" umożliwiające wczytywanie danych pomiarowych przed i po wystąpieniu sygnału wyzwalającego.

Współczesnym komputerom osobistym wystarcza szybkości i mocy obliczeniowej w zakresie zbierania i analizy danych na potrzeby systemów pomiarowych. W niektórych jednak zastosowaniach szybkość tradycyjnego komputera osobistego może nie być wystarczająca do przetwarzania sygnałów w czasie rzeczywistym, pomimo użycia 32-bitowego procesora z towarzyszącym mu koprocesorem. Typowe zastosowania tego typu analiz występują przeważnie w dziedzinie obróbki sygnałów mowy i obrazu. Najprościej sięgnąć wtedy po kartę analizy (cyfrowego przetwarzania sygnałów, w skrócie kartę DSP) wyposażoną w bardzo szybki procesor sygnałowy (np. zmiennoprzecinkowy: TMS320C40, stałoprzecinkowy: TMS320C50, Texas Instruments). Wtedy skomplikowane obliczenia (analizy) mogą być prowadzone jednocześnie z wykonaniem programu procesora głównego. Dostępne są również rozwiązania hybrydowe, łączące typowe karty DAQ z DSP (zwykle DAQ z wbudowanym procesorem sygnałowym).

Oprogramowanie użytkowe karty zbierania danych może być realizowane, co najmniej, na trzech poziomach:

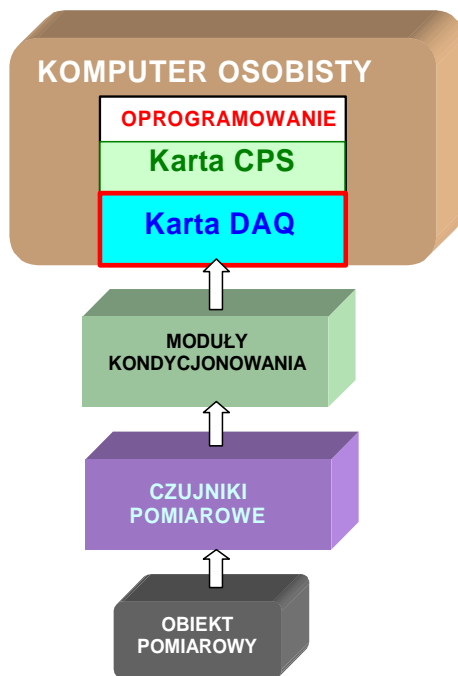
- Oprogramowanie na poziomie rejestrowym: uciążliwe programowanie rejestrów karty DAQ (dowolne języki programowania),
- Oprogramowanie na poziomie sterownika - sterownik (firmowy) oraz programy aplikacyjne (odwołania do funkcji wysokiego poziomu zawartych w sterowniku),
- Oprogramowanie narzędziowe (korzystanie z bibliotecznych funkcji obsługi karty).

Zdarza się, coraz rzadziej, że karty zbierania danych wymagają doprowadzenia do swoich wejść sygnałów elektrycznych o standaryzowanych poziomach. W związku z tym kompletny system powinien zawierać ponadto tzw. moduły standaryzacji, inaczej kondycjonowania sygnałów.

Moduły kondycjonowania sygnału umożliwiają:

- Wzmacnianie sygnałów (w dwu kierunkach),
- linearyzację charakterystyk czujników,
- izolację galwaniczną,
- filtrację sygnałów niepożądanych (w tym antyaliasingową),
- multipleksowanie.

Schemat blokowy konfiguracji sprzętowej komputera osobistego z kartą DAQ zamieszczono na rys. 2.2.



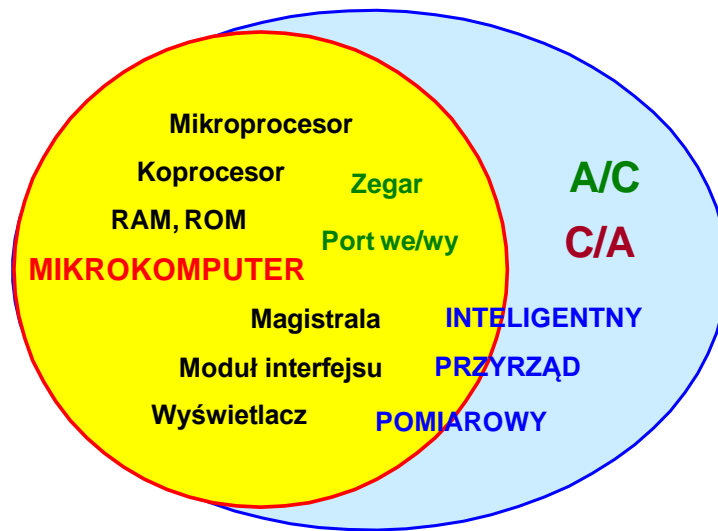
Rys. 2.2 Schemat blokowy konfiguracji sprzętowej komputera osobistego z kartą DAQ

3. Wirtualne przyrządy pomiarowe

Wirtualny przyrząd pomiarowy stanowi ostatnie (dla tradycjonalistów kontrowersyjne) ogniwo w ewolucji przyrządów pomiarowych. Ewolucja ta zawiera kilka etapów dobrze znanych w środowisku metrologicznym. Poniżej przedstawiono krótki przegląd tych etapów, który ma doprowadzić do sformułowania definicji wirtualnego przyrządu pomiarowego [34].

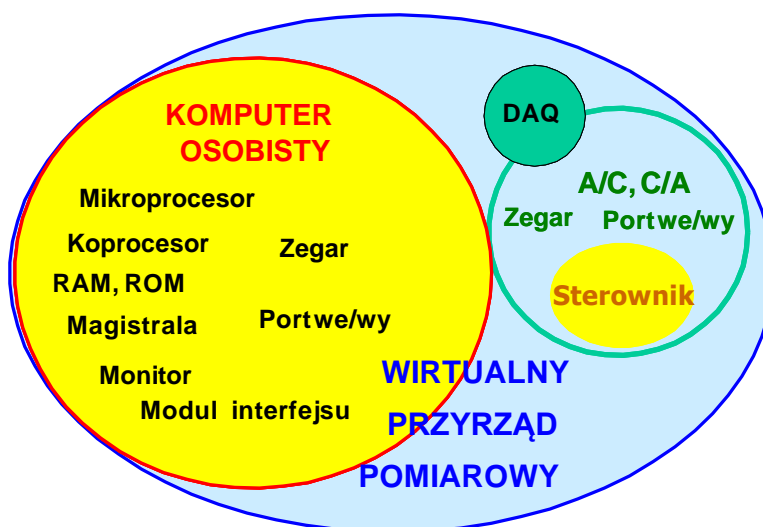
- Przyrządy analogowe: W przypadku przyrządów analogowych charakterystyczna była ręczna obsługa przyrządu, żmudna obserwacja wskaźnika wychyłowego oraz ręczne zapisywanie wyników pomiaru. Wynik pomiaru obarczony był dużą dawką subiektywizmu – elementu wysoce niekorzystnego z metrologicznego punktu widzenia. Automatyzacja pomiarów nie była możliwa.
- Przyrządy cyfrowe: W tym przypadku wciąż obowiązywała ręczna obsługa przyrządu. Wynik pomiaru wyświetlany był na wskaźniku cyfrowym – wystąpiła eliminacja subiektywnego odczytu. Po wyposażeniu przyrządu w specjalne wyjście pojawiła się możliwość wysyłania wyników pomiaru na drukarkę oraz do komputera z wykorzystaniem jego standardowego portu równoległego (układ 8255 Intel). Wszystkie cyfry (znaki alfanumeryczne) wyniku pomiarowego wyprowadzone były w kodzie BCD. Pojawił się impuls „koniec pomiaru” (inicjacja automatycznego odczytu, wyzwalanie drukarki). Dodanie impulsowego wejścia „start pomiaru” zmotywowało, co bardziej zaradnych użytkowników, do oprogramowania portu 8255 w kierunku sterowania wyzwalaniem pomiaru. Był to pierwszy krok w kierunku automatyzacji pomiarów, która zawierała taką sekwencję zdarzeń jak: *wyzwalanie pomiaru, odbiór wyniku, przetwarzanie wyniku, prosta wizualizacja na ekranie monitora*. Sam program obsługi przyrządu był dość żmudny w realizacji (oprogramowanie na niskim poziomie), a efekt mało przyjazny dla użytkownika. Kompendium wiedzy na temat cyfrowych przyrządów pomiarowych zawarte jest w książce [36] – unikalnej pozycji na rynku krajowym.
- Interfejsy specjalizowane. Wprowadzenie specjalizowanych interfejsów pomiarowych (takich jak np. CAMAC), umożliwiło automatyzację pomiarów, w pewnych specyficznych zastosowaniach. Rozwój tych interfejsów był niezbyt ekspansywny. Objawiło się to brakiem powszechnej standaryzacji, a co za tym idzie rozpowszechnienia interfejsów.
- Interfejs IEC-625: Dopiero wprowadzenie interfejsu pomiarowego IEC-625 (w porządku chronologicznym: GPIB, GPIB, IEEE-488) zrewolucjonizowało organizację pełnej obsługi autonomicznych przyrządów pomiarowych. Interfejs IEC-625 umożliwia zdalną obsługę przyrządów (wybór funkcji pomiarowej, zakresu itp.), bezpośrednie przesyłanie wyników do kontrolera, a więc pełną automatyzację pomiarów. Objęty został standaryzacją oraz unifikacją, co doprowadziło do powstania zmodyfikowanej normy 625.2.
- Inteligentne przyrządy pomiarowe: Niemal jednocześnie z wyposażeniem przyrządów w bloki interfejsu rozwijano ich zdolności do przetwarzania sygnałów pomiarowych. Polegało to na zastosowaniu mikroprocesorów (jednoukładowych, uniwersalnych, sygnałowych), co umożliwiało wykorzystanie mniej lub bardziej zaawansowanych algorytmów cyfrowego przetwarzania sygnałów. Niemal wzorcowym przykładem takiego przyrządu był cyfrowy analizator widma.

Idea inteligentnego przyrządu pomiarowego przedstawiona jest na rysunku 3.1.



Rys. 3.1 Idea inteligentnego przyrządu pomiarowego

Taka konfiguracja autonomicznego przyrządu pomiarowego miała racjonalne uzasadnienie dopóki na biurku każdego inżyniera nie pojawił się komputer osobisty. Wtedy to okazało się, że powielanie standardowych bloków komputera, wewnątrz obudowy każdego inteligentnego przyrządu jest wysoce nieracjonalne ze względów finansowych. Równie dobrze można komputer osobisty wyposażyć w kartę (moduł) przetworników (A/C, C/A) ze sterownikiem, napisać odpowiednie oprogramowanie użytkowe i realizować oczekiwane funkcje pomiarowe oraz algorytmy cyfrowego przetwarzania sygnałów. W celu umiejscowienia realizowanych procesów w czasie trzeba wtedy użyć standardowego zegara komputera (np.:8253, Intel). Do komunikacji ze światem zewnętrznym, w sensie sygnałów cyfrowych (sterowanie elementami obiektu pomiarowego), można wykorzystać standardowy równoległy port we/wy (np.: 8255, Intel). Takie podejście jednak nie byłoby wygodne dla użytkownika, bowiem wymagałoby od niego ingerencji w struktury rejestrowe tych układów i oprogramowania ich na niskim poziomie (rejestrowym). Producenci kart (A/C, C/A) szybko znaleźli satysfakcjonujące rozwiązanie, które polegało na dodaniu wspomnianych modułów do karty. Umożliwiło to dołączenie do sterownika karty funkcji wysokiego poziomu do ich obsługi. Taki jest rodowód karty zbierania danych (Data Acquisition: DAQ), a komputer w nią wyposażony ilustruje koncepcję wirtualnego przyrządu pomiarowego (rys. 3.2).

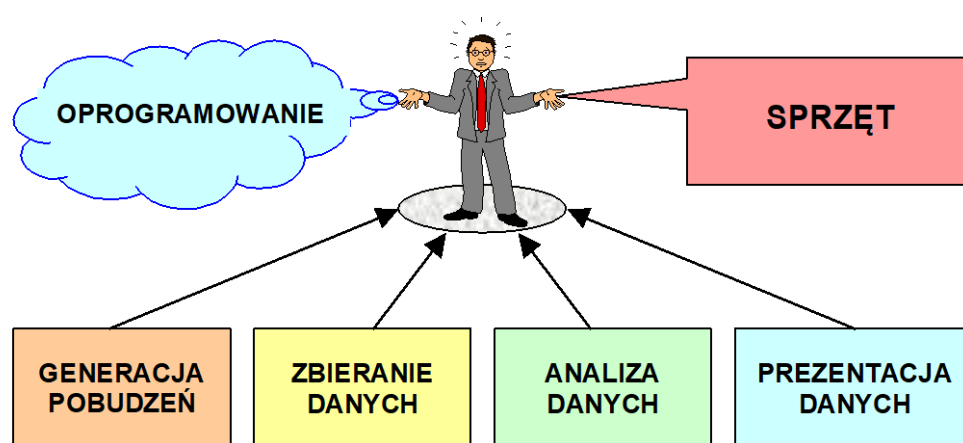


Rys. 3.2 Rodowód karty zbierania danych (DAQ) na tle koncepcji wirtualnego przyrządu pomiarowego

Zaopatrzenie tak wyposażonego komputera osobistego w odpowiednie oprogramowanie, realizujące interfejs użytkownika występujący w postaci wiernego obrazu płyty czołowej konkretnego przyrządu pomiarowego, z możliwością obsługi manipulatorów za pomocą myszy, stwarza użytkownikowi wrażenie obsługi rzeczywistego przyrządu pomiarowego. Taka jest koncepcja wirtualnego przyrządu pomiarowego. Koncepcja ta wykorzystana została do stworzenia teorii opisującej strukturę i konfigurację nowoczesnych przyrządów pomiarowych w szerokim kontekście, nie mającym precedensu w długiej historii rozwoju przyrządów pomiarowych. Początek tej teorii dało rozszerzenie pojęcia wirtualnego przyrządu pomiarowego na nowe struktury, dotychczas uznawane za „systemy pomiarowe”. Kolejne etapy to zdalny dostęp do przyrządu pomiarowego (poprzez sieć komputerową), zdalny dostęp do laboratorium, wreszcie laboratorium wirtualne. Niektóre z tych pojęć wzbudzają kontrowersje czy może łagodniej niechęć tradycjonalistów (w tym niektórych metrologów), ale stały się faktem, od którego nie ma już odwrotu. W tym kontekście warto dodać, że wspomniane zdobycze metrologii mogą być bardzo przydatne w procesie nowoczesnego kształcenia na odległość, czy też wspomagania kształcenia w sensie tradycyjnym, a przede wszystkim kształcenia ustawicznego. Z chwilą pojawienia się niemal powszechnego dostępu do Internetu te znane formy kształcenia nabierają rumieńców, a dalszy ich rozwój jest nieunikniony.

3.1. Idea wirtualnego przyrządu pomiarowego

Idea wirtualnego przyrządu pomiarowego (VI: Virtual Instrument) znajduje wyraźne odniesienie do przyrządu tradycyjnego. Jak wiadomo, nowoczesny model autonomicznego przyrządu pomiarowego zawiera cztery podstawowe bloki funkcjonalne: zbierania danych, przetwarzania danych, generacji pobudzeń³ i prezentacji wyników. Przyrząd wirtualny zawiera te same bloki funkcjonalne, z tą różnicą, że niekoniecznie muszą być umieszczone we wspólnej obudowie. Idea przyrządu wirtualnego polega więc na połączeniu funkcji przyrządu tradycyjnego (ustalonych na sztywno) z elastycznymi, często definiowanymi przez użytkownika, funkcjami komputera osobistego. Poszczególne funkcje mogą być realizowane zarówno z użyciem sprzętu, jak i oprogramowania. Ideę przyrządu wirtualnego zilustrowano na rys. 3.3. Blok decyzyjny oznaczono symbolicznie znakiem postaci projektanta, którym w pewnych przypadkach może stać się sam użytkownik.



Rys. 3.3 Idea wirtualnego przyrządu pomiarowego

Podsumowaniem tego punktu niech będzie autorska definicja wirtualnego przyrządu pomiarowego [34]:

³ Blok generacji pobudzeń jest blokiem specyficznym. Trzeba o tym pamiętać przy formułowaniu oraz interpretacji definicji.

Przyrząd wirtualny to rodzaj inteligentnego przyrządu pomiarowego powstałego w wyniku sprzężenia pewnego sprzętu nowej generacji z komputerem osobistym ogólnego przeznaczenia i przyjaznym dla użytkownika oprogramowaniem, które umożliwia użytkownikowi współpracę z komputerem na zasadach takich jakby obsługiwał tradycyjny przyrząd pomiarowy.

3.2. Charakterystyka przyrządów wirtualnych

Doświadczenia w zakresie implementacji wirtualnych przyrządów pomiarowych są już na tyle duże, że upoważniają do wprowadzenia pewnego rodzaju kategoryzacji i systematyzacji.

3.2.1 Kategorie przyrządów wirtualnych

Już sama definicja, z uwagi na brak precyzji w sformułowaniach, wskazuje na dużą różnorodność w zakresie architektury przyrządów wirtualnych. Z grubsza można je podzielić na trzy kategorie, w których skład wchodzi następujące elementy:

- Kat. A. Fizycznie istniejące przyrządy autonomiczne wyposażone w interfejsy przyrządowe IEC-625 lub RS232 (i pochodne), panel graficzny na ekranie monitora (symulujący płytę czołową) - obsługa przyrządu za pomocą „myszy”,
- Kat. B. Karta DAQ lub moduły VXI, (PXI, PCI - bez płyty czołowej, w miejsce przyrządu autonomicznego), panel graficzny na ekranie monitora (symulujący płytę czołową) - obsługa przyrządu za pomocą „myszy”,
- Kat. C. Brak fizycznego przyrządu (sprzętu - poza PC), dane wejściowe pobierane z plików w pamięci masowej, bazy danych, innych komputerów lub generowane w sposób numeryczny, panel graficzny na ekranie monitora (symulujący płytę czołową), obsługa za pomocą „myszy”.

Ostatnia kategoria dotyczy raczej symulacji przyrządu, systemu lub procesu pomiarowego i jest niezwykle przydatna w dydaktyce. Charakteryzuje się dużą uniwersalnością, elastycznością oraz niskim kosztem opracowania. Wyjątkowo dobrze nadaje się jako uzupełnienie do nowoczesnych podręczników szkolnych i akademickich wydawanych w formie elektronicznej. Najbardziej korzystne aplikacje to tzw. *aplety Javy* dołączane do pliku przygotowanego w formacie HTML.

3.2.2 Cechy przyrządów wirtualnych

Istotne cechy przyrządów wirtualnych można ująć w trzech punktach:

1. powiększona funkcjonalność,
2. otwarta architektura,
3. łatwość w rozpowszechnianiu idei.

ad 1.

Powstają nieograniczone wręcz możliwości wzbogacania funkcji pomiarowych, algorytmów przetwarzania i analizy sygnałów oraz (a może przede wszystkim) prezentacji wyników pomiaru.

ad 2.

Otwarcie architektury przyrządów wygodnie jest zobrazować odwołując się ponownie do rys. 3.1, pokazującego częściową równowagę architektury mikrokomputera i inteligentnego przyrządu pomiarowego. Wynika z niego wyraźnie, że nowoczesny autonomiczny (inteligentny) przyrząd pomiarowy zawiera wiele elementów „komputerowych”. Niektóre specjalizowane moduły sprzętowe zaś, mogą być dostosowane do współpracy z

magistralą komputera. Oznacza to, że znaczna grupa elementów sprzętowych i programowych, powielanych w przyrządach autonomicznych, może być zastąpiona standardowym komputerem. Oprogramowanie aplikacyjne, może być przechowywane w pamięci komputera. Nieograniczona liczba przyrządów wirtualnych może wykorzystywać tę samą platformę (komputer, monitor i mysz) do obsługi funkcji pomiarowych i prezentacji wyników. Kluczowym elementem decydującym o otwarciu architektury przyrządu wirtualnego jest nieograniczony wręcz dostęp do charakterystycznych modułów i funkcji komputera. Zalicza się do nich:

- Porty we/wy (I/O), odwzorowane na przestrzeń adresową komputera - szybkie zapisywanie rejestrów;
- Bezpośredni dostęp do pamięci (DMA), buforowanie - szybkie transfery danych;
- Przerwania sprzętowe i programowe (INT) z natychmiastową reakcją;
- Możliwość wyzwalania i taktowania sprzętowego;
- Możliwość wykrywania i generacji zdarzeń;
- Łatwość synchronizacji;
- Łatwość formatowania danych;
- Szybkie przetwarzanie danych (algorytmy cyfrowego przetwarzania sygnałów);
- Wzbogacona prezentacja wyników: „bajkowy” wygląd płyty czołowej.

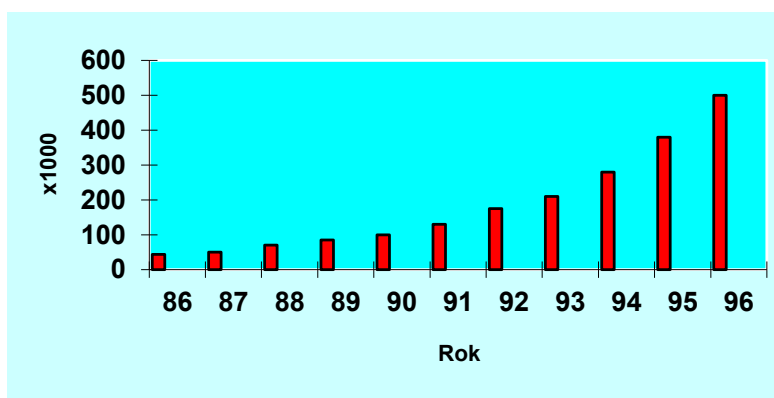
ad 3.

Łatwość w rozprzestrzenianiu idei wirtualnych przyrządów pomiarowych wynika z faktu, że znaczną część przyrządu stanowi oprogramowanie. Taka forma doskonale nadaje się do modyfikowania, unowocześniania, poprawiania i przesyłania za pośrednictwem Internetu. Ta cecha sprawiła, że technologia wirtualnych przyrządów pomiarowych wkroczyła do całego szeregu platform informatycznych. Należą do nich między innymi:

- Systemy operacyjne: DOS, Windows/XP/Win7/Win10, Linux, Unix, HP-UX, Mac OS,
- Procesory: Intel, SPARC i inne,
- Magistrale standardowe: ISA, PCI, PCMCIA,
- Magistrale modułowe: VME, VXI, PXI
- Interfejsy standardowe (porty): równoległy, RS232, RS422, RS423, RS485, USB
- Interfejsy pomiarowe: IEC-625, IEC-625.2,
- Interfejsy przemysłowe: Ethernet, CAN, Device Net, Fieldbus, PROFIBUS
- Języki programowania: Basic, Visual Basic, Basic.NET, HT Basic, C, C++, Visual C++.
- Zintegrowane pakiety oprogramowania narzędziowego, np:
 - PC Instruments, HP VEE (Hewlett-Packard⁴),
 - PCI (Siemens),
 - TestPoint (Keithley Instruments)
 - LabVIEW, LabWindows/CVI, Measurement Studio (National Instruments).

⁴ Firma Hewlett-Packard uległa podziałowi. Sprzęt pomiarowo-kontrolny i odpowiadające mu oprogramowanie to domena firmy Agilent.

Trendy rozwojowe przyrządów wirtualnych najlepiej zilustrować za pomocą wykresu pierwotnego zapotrzebowania na tzw. „digitizery”, tzn. wszelkiego typu urządzenia umożliwiające wprowadzanie sygnałów analogowych do komputera⁵ (rys.3.4).



Rys. 3.4 Zapotrzebowanie na „digitizery” ogólnego przeznaczenia w latach 1986-1996

Tendencja ta występuje z jednoczesnym, wyraźnym spadkiem zapotrzebowania na specjalizowane przyrządy autonomiczne (w tym inteligentne przyrządy pomiarowe). Wy tłumaczenie jest proste: komputer, „digitizer” oraz oprogramowanie narzędziowe dają potencjalną możliwość konstrukcji dowolnego przyrządu pomiarowego, a ich wykorzystanie wielokrotnie znakomicie obniża koszt jednostkowy.

3.3. Konfiguracje przyrządów wirtualnych

Różne konfiguracje przyrządów wirtualnych powstają na bazie szeregu platform sprzętowych. Najważniejsze spośród nich to:

- Karty zbierania danych (DAQ) do współpracy z magistralą komputera (PCI) oraz modułami kondycjonowania sygnału;
- Systemy modułowe VXI, PXI;
- Autonomiczne przyrządy pomiarowe wyposażone w interfejsy pomiarowy IEC-625;
- Rozproszone układy typu Input/Output.

3.3.1 Konfiguracje z użyciem kart zbierania danych

Jak już wiadomo, genezę powstania kart zbierania danych (DAQ – Data Acquisition) określa umieszczenie układów „wzbogacających” komputer na jednej, wspólnej płycie, bezpośrednio współpracującej z magistralą komputera. Umożliwia to łatwą komunikację z procesorem oraz szybkie transfery danych, zwłaszcza po zastosowaniu przerwań i bezpośredniego dostępu do pamięci. Moduł DAQ, zawiera: przetworniki A/C i C/A, dodatkowe porty WE/WY oraz dodatkowe układy zegarowe - TIMER. Takie rozwiązanie umożliwia korzystanie wyłącznie z zasobów sprzętowych modułu, co znakomicie ułatwia realizację oprogramowania z użyciem sterowników firmowych.

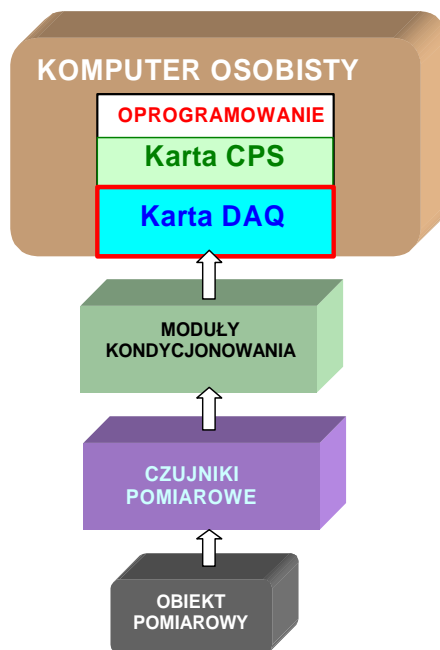
Zdarza się, coraz rzadziej, że karty zbierania danych wymagają doprowadzenia do swoich wejść sygnałów elektrycznych o standaryzowanych poziomach. W związku z tym kompletny system powinien zawierać ponadto tzw. moduły standaryzacji, inaczej kondycjonowania sygnałów (rys. 3.5).

⁵ Dane prezentowane na seminarium firmy National Instruments.

Moduły kondycjonowania sygnału umożliwiają:

- Wzmacnianie sygnałów (w dwu kierunkach),
- linearyzację charakterystyk czujników,
- izolację galwaniczną,
- filtrację sygnałów niepożądanych (w tym antyaliasingową),
- multipleksowanie.

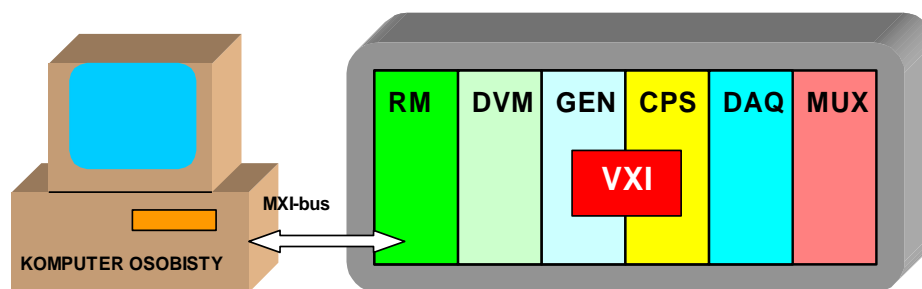
Schemat blokowy konfiguracji sprzętowej wirtualnego przyrządu pomiarowego z DAQ zamieszczono na rys. 3.5. Jest to powtórzenie schematu prezentowanego już przy okazji opisu kart zbierania danych.



Rys. 3.5 Schemat blokowy konfiguracji wirtualnego przyrządu pomiarowego z DAQ

3.3.2 Konfiguracja z użyciem magistrali VXI (PXI)

Magistrala VXI (VXIbus: VME EXTENSIONS FOR INSTRUMENTATION) - powstała w wyniku rozszerzenia możliwości magistrali komputera modułowego (VMEbus) o elementy umożliwiające obsługę modułowych przyrządów pomiarowych (modułów). Przykład konfiguracji przyrządu wirtualnego w standardzie VXI zamieszczono na rysunku 3.6.



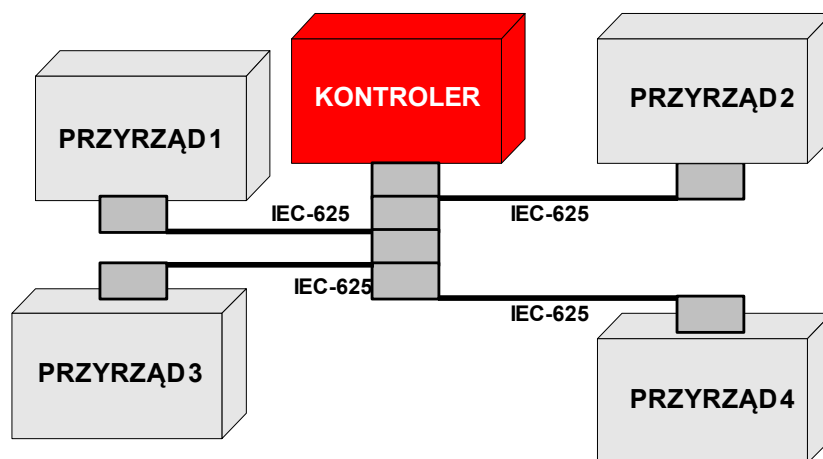
Rys. 3.6 Przykład konfiguracji przyrządu wirtualnego w standardzie VXI.: RM - Resource Manager, DVM - woltomierz cyfrowy, GEN - generator, CPS - moduł/karta procesora sygnałowego, DAQ - moduł/karta zbierania danych, MUX – multiplekser

Przypominamy, że ostatnio, w wyniku rozwoju techniki komputerowej, powstało nowe rozwiązanie, wzorowane na VXI: system modułowy PXI. W systemie z magistralą PXI (**P**CI **E**xtension for **I**nstrumentation) do obsługi modułowych przyrządów pomiarowych wykorzystuje się standardową magistralę PCI (**P**ERIPHERAL **C**OMPONENT **I**NTERCONNECT) z komputera osobistego rozszerzoną o elementy ułatwiające obsługę modułów pomiarowych, podobnie jak w przypadku VXI. W rozwiązaniu tym funkcjonuje tzw. CompactPCI, która łączy specyfikację elektryczną PCI z mechaniką typu *Eurocard*. Szybkość transmisji danych w standardzie PXI wynosi 132MB/s i w tym zasadza się jego siła. Po opcjonalnym rozszerzeniu magistrali danych z 32 do 64 bitów można osiągnąć nawet 264MB/s. Współpracujące z magistralą PXI (umieszczone w kasecie) przyrządy modułowe, widziane są w przestrzeni adresowej urządzeń wejścia/wyjścia systemu komputerowego.

Z czasem, urządzenia stawały się coraz bardziej wymagające co do przepustowości. W rezultacie powstała magistrala PCI Express, aby przezwyciężyć ograniczenia w ramach wspólnej magistrali. W przeciwieństwie do PCI, która dzieli pasmo pomiędzy wszystkie urządzenia, magistrala PCI Express oferuje każdemu urządzeniu własny dedykowany kanał danych (tzw. *pipeline*). Dane są przesyłane szeregowo, w pakietach poprzez pary linii nadawania i odbioru zwane pasami, które umożliwiają uzyskanie szybkości 250 MB/s dla każdego pasa (PCI Express 1.0). Pasy mogą być grupowane (multiplikowane) x1, x2, x4, x8, x12, x16, aby zwiększyć przepustowość dla gniazda i osiągnąć do 4 GB/s całkowitej przepustowości. PCI Express 2.0 podwaja przepustowość per-pas z 250 do 500 MB/s. PCI Express zapewnia skalowalną przepustowość osiągając maksymalnie 30-krotną przepustowość tradycyjnej PCI.

3.3.3 Konfiguracja z użyciem interfejsu przyrządowego IEC-625

Sercem konfiguracji jest kontroler interfejsu - komputer wyposażony w kartę interfejsu IEC-625, z dołączonym oprogramowaniem. Współpracują z nim autonomiczne przyrządy pomiarowe wyposażone w interfejsy. Schemat blokowy konfiguracji IEC obrazuje rys. 3.7.



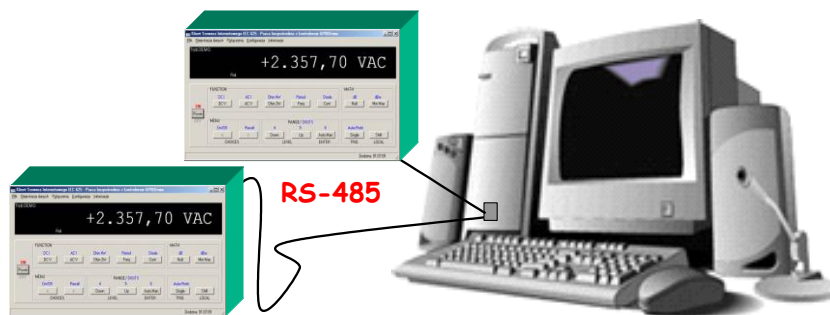
Rys.3.7 Konfiguracja przyrządu wirtualnego z interfejsem IEC-625

Parametry standardu są następujące:

- Szybkość przesyłania danych 1MB/s,
- Liczba przyrządów: 14,
- Długości kabli: 2m.

3.3.4 Konfiguracja z użyciem interfejsu szeregowego RS 232C oraz konfiguracje bezprzewodowe

Konfiguracja z interfejsem szeregowym, jest chyba najbardziej znana. Podstawowa, z punktu widzenia przeciętnego użytkownika, koncepcja konfiguracji opisującej asynchroniczną komunikację między komputerem (DTE), a przyrządem pomiarowym (DTE) nosi nazwę *modemu zerowego bez sterowania transmisją*. Hasło obejmuje pozostałe znane standardy jak: RS423A, RS422A, RS485. Do tej grupy, interfejsów przewodowych, należy również nowoczesny standard typu USB. Przykład konfiguracji wirtualnego przyrządu pomiarowego z interfejsem RS-485 zamieszczono na rys. 3.8.

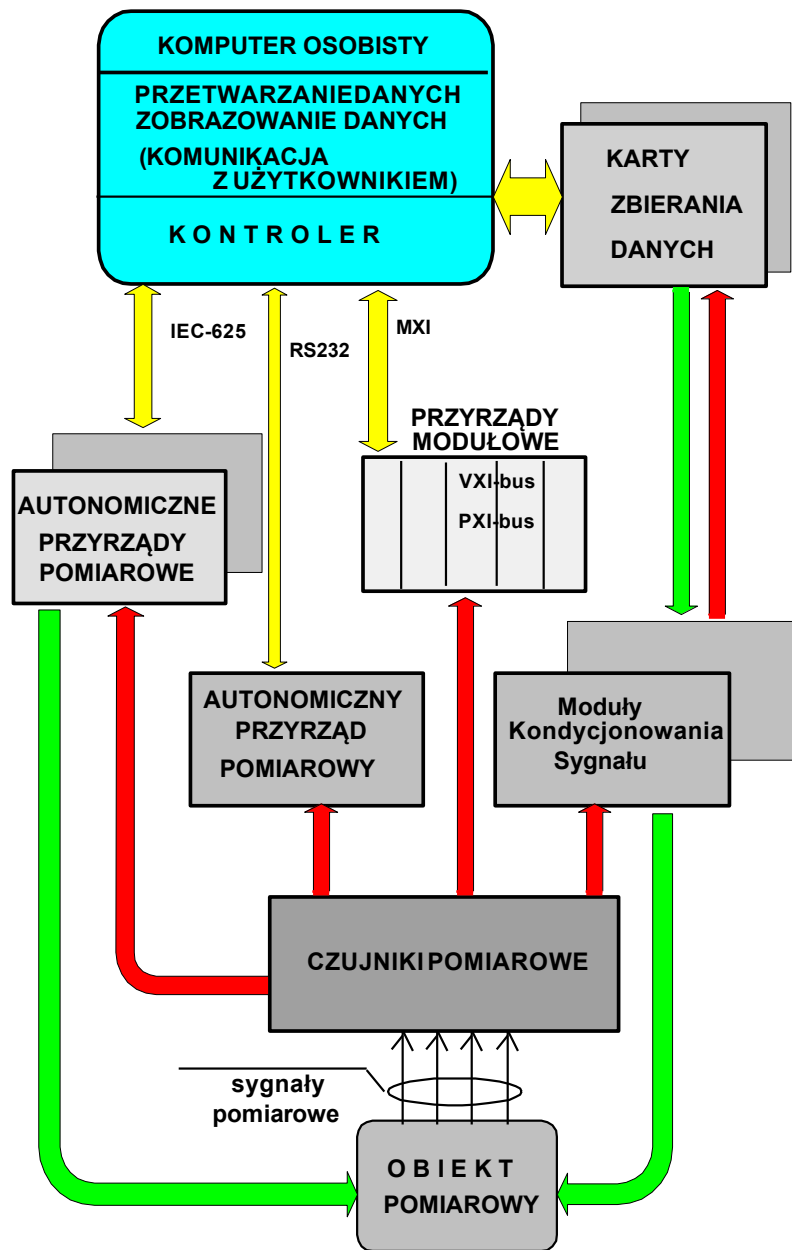


Rys.3.8 Konfiguracja wirtualnego przyrządu pomiarowego z interfejsem RS-485

Na zakończenie tego rozdziału wypada wspomnieć o możliwości bezprzewodowego sterowania przyrządami pomiarowymi. Do tej grupy metod należy technika podczerwieni (standard *IrDA*) oraz coraz bardziej modna technika radiowa typu *Bluetooth*. Nietrudno przewidzieć możliwość ich wykorzystania również w laboratoriach naukowo-badawczych i dydaktycznych. Coraz głośniejszą mowa się też o aplikacjach standardu GSM. Bardziej szeroki opis tych metod znaleźć można w literaturze.

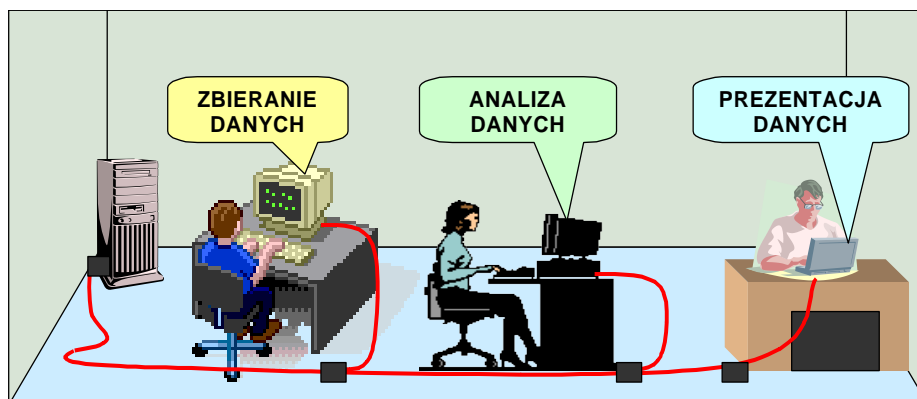
3.3.5 Uogólniona struktura wirtualnego przyrządu pomiarowego

W kontekście ostatnich rozważań warto pokusić się o stworzenie uogólnionej struktury wirtualnego przyrządu pomiarowego, łączącej wszystkie wymienione konfiguracje. Schemat blokowy takiej struktury zamieszczono na rysunku 3.9.



Rys. 3.9 Schemat uogólnionej struktury VI

Najbardziej modernistyczne rozwiązanie konfiguracyjne - to wirtualny przyrząd pomiarowy rozproszony w lokalnej sieci komputerowej. Na rys. 3.10 zamieszczono schemat funkcjonalny takiego przyrządu. Rozproszeniu podlegają tu bloki funkcjonalne przyrządu; zbieranie danych, analiza danych i prezentacja wyników.



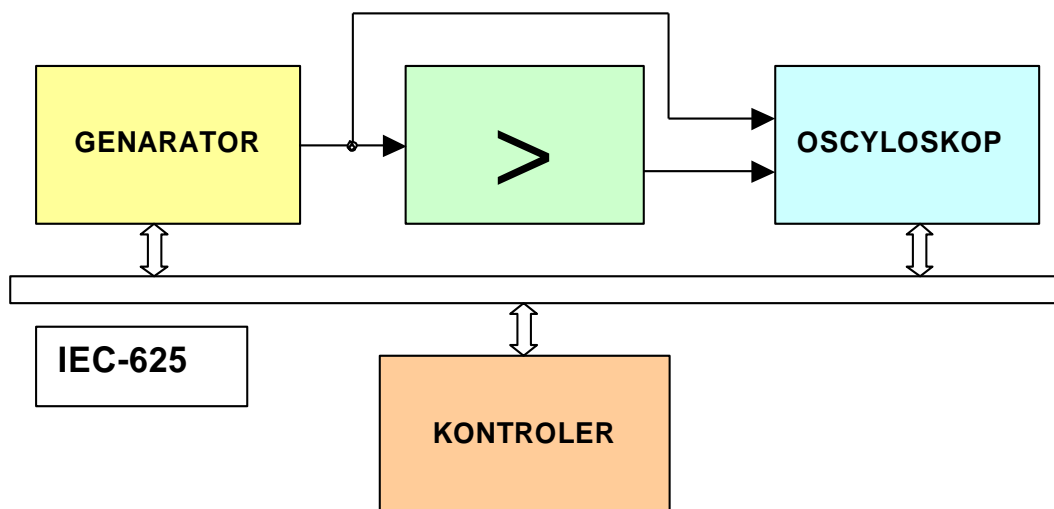
Rys.3.10 Rozproszony w sieci wirtualny przyrząd pomiarowy

3.4. Przykłady realizacji przyrządów wirtualnych

W Zakładzie Systemów Informacyjno-Pomiarowych Politechniki Warszawskiej (SIP) opracowano cały szereg przyrządów wirtualnych zarówno na potrzeby dydaktyki, jak i badań naukowych. Poniżej opisane są wybrane realizacje.

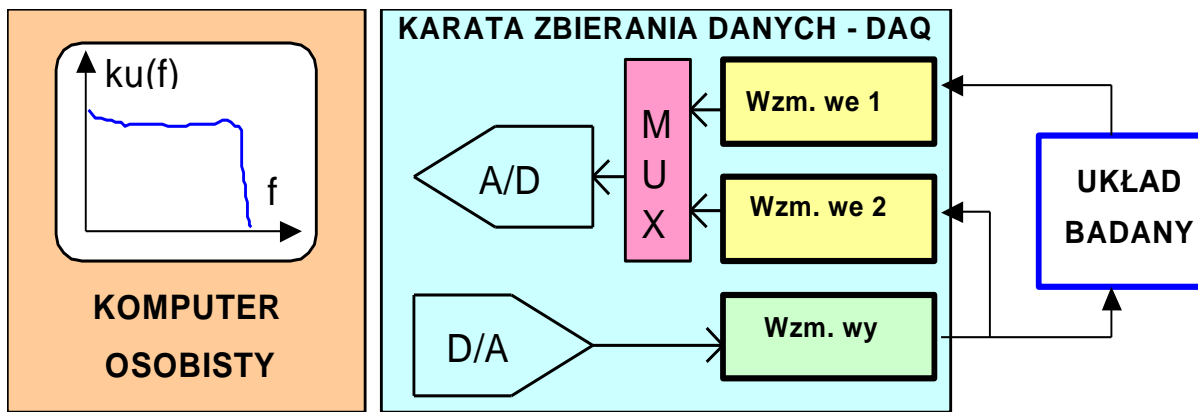
3.4.1 Przyrząd do badania parametrów filtrów aktywnych

Opracowane zostały dwa wariantowe rozwiązania przyrządu wirtualnego do badania charakterystyk wzmacniaczy i filtrów aktywnych. Obydwa umożliwiają automatyczny pomiar i rejestrację charakterystyk: amplitudowo-częstotliwościowych, amplitudowo-fazowych, statycznych i dynamicznych charakterystyk przejściowych, odpowiedzi na skok jednostkowy, odpowiedzi impulsowej, odpowiedzi na napięcie liniowo narastające. Do realizacji pierwszego wariantu (rys.3.11) wykorzystano autonomiczne przyrządy pomiarowe: generator funkcyjny (AFG5101 Tektronix) i oscyloskop cyfrowy (2224 Tektronix). Wymienione przyrządy współpracują z kontrolerem systemu (komputer osobisty wyposażony w kartę GPIB, National Instruments) za pośrednictwem interfejsu IEC-625. Algorytm pracy systemu jest funkcją rodzaju dokonywanego pomiaru.

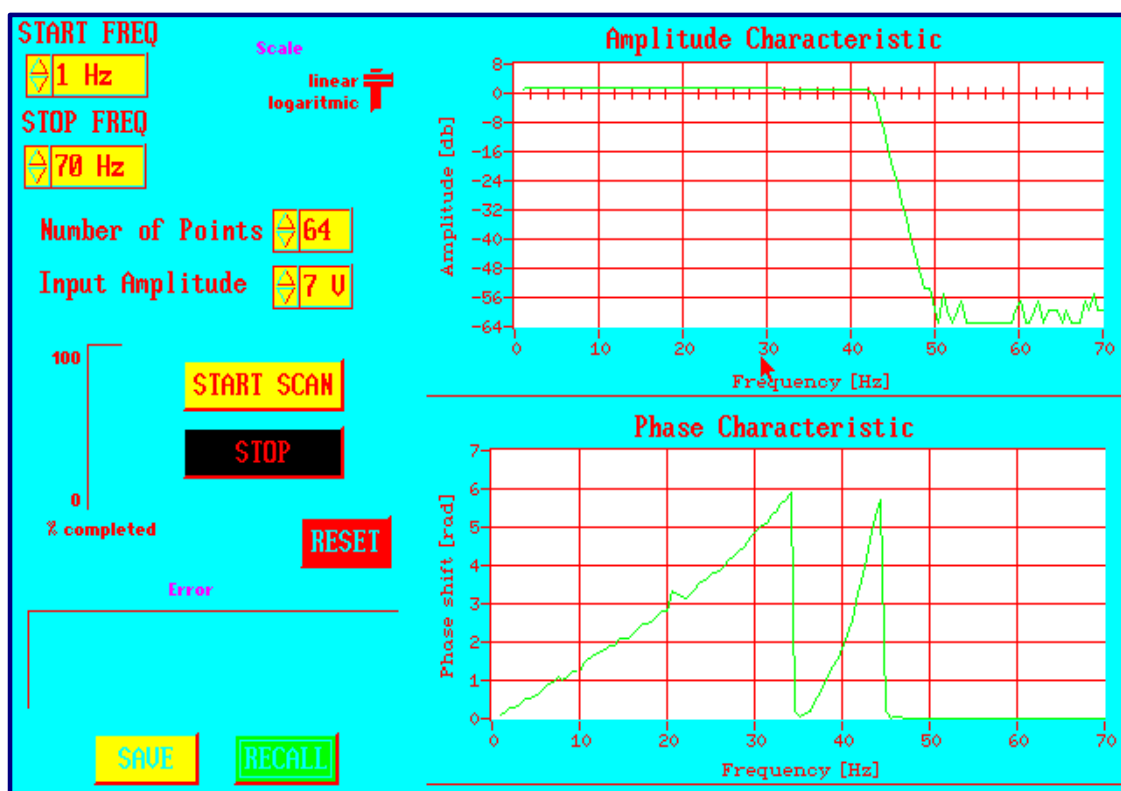


Rys.3.11 Struktura przyrządu wirtualnego do badania charakterystyk filtrów aktywnych

W drugim wariantcie podobne funkcje pomiarowe zrealizowano stosując uniwersalną kartę zbierania danych – LAB PC+ (National Instruments), która współpracuje bezpośrednio z magistralą komputera (PC). Pojęcie „przyrząd wirtualny” nie podlega tutaj dyskusji. W obydwu przypadkach oprogramowanie systemowe (z grafiką), w postaci "przyjaznej dla użytkownika" opracowane zostało w zintegrowanym środowisku LabWindows firmy National Instruments. Zasada działania układu jest bardzo prosta i wynika wprost ze schematu funkcjonalnego zamieszczonego na rysunku 3.12. Płyta czołowa wirtualnego analizatora (dla obydwu wariantów) przedstawiona jest na rys.3.13.



Rys.3.12 Schemat blokowy analizatora charakterystyk filtrów aktywnych: DAQ

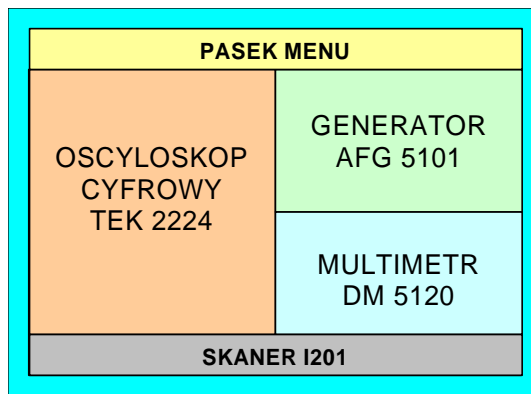


Rys.3.13 Płyta czołowa analizatora charakterystyk filtrów aktywnych: IEC, DAQ Dwukanałowy analizator widma

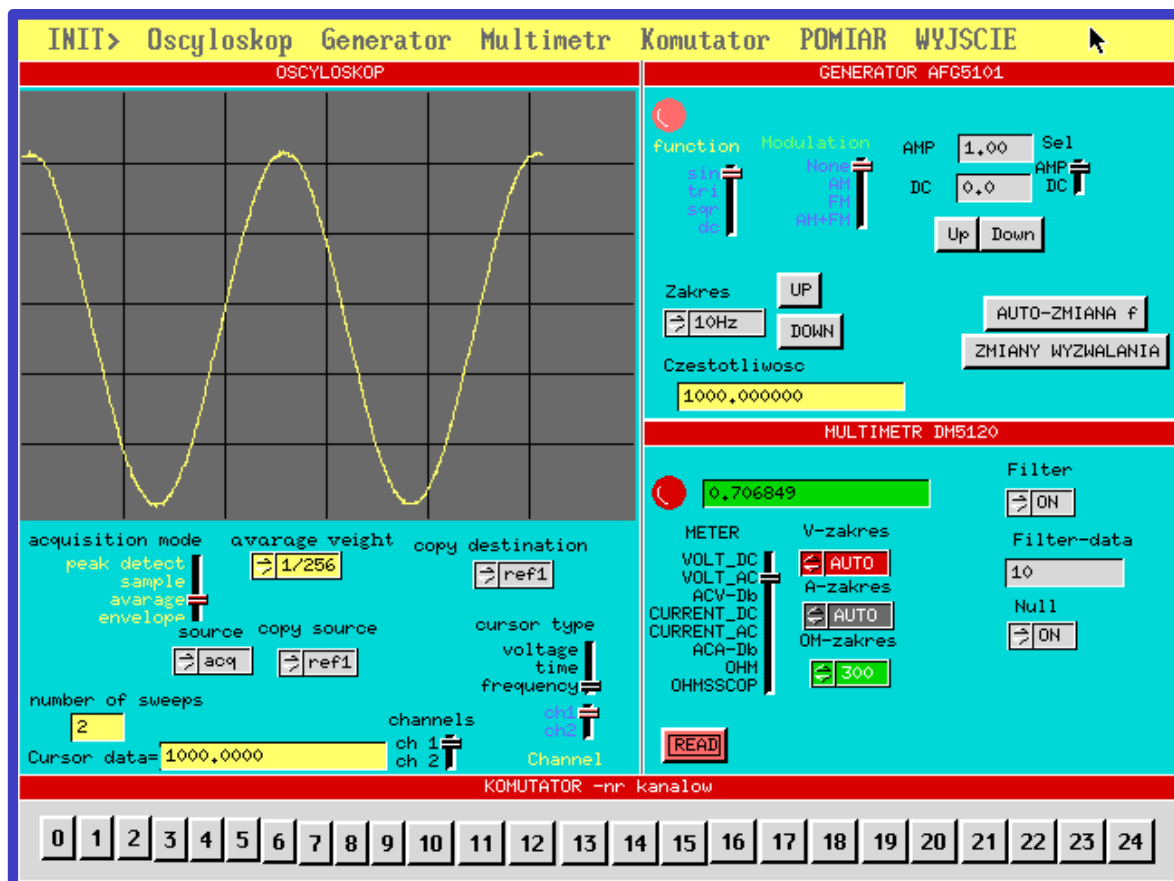
3.4.2 Zestaw przyrządów wirtualnych

Na rysunkach 3.14 (w sposób poglądowy) i 3.15 (w sposób wierny) przedstawiono panel płyty czołowej zestawu wirtualnych przyrządów pomiarowych: oscyloskop, generator, multimetr, komutator. Zestaw ten opracowano stosując przyrządy autonomiczne, odpowiednio:

- Oscyloskop cyfrowy TEK 2224, Tektronix
- Programowany generator funkcyjny AFG 5101, Tektronix
- Multimetr cyfrowy DM5120, Tektronix,
- Skaner I201, Meratronik.



Rys.3.14 Poglądowy obraz płyty czołowej zestawu przyrządów wirtualnych

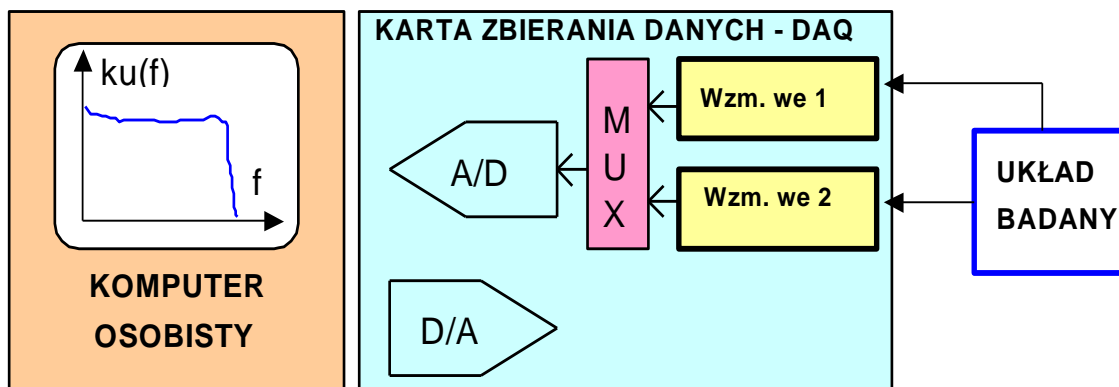


Rys.3.15 Panel płyty czołowej zestawu przyrządów wirtualnych

3.4.3 Cyfrowy analizator widma

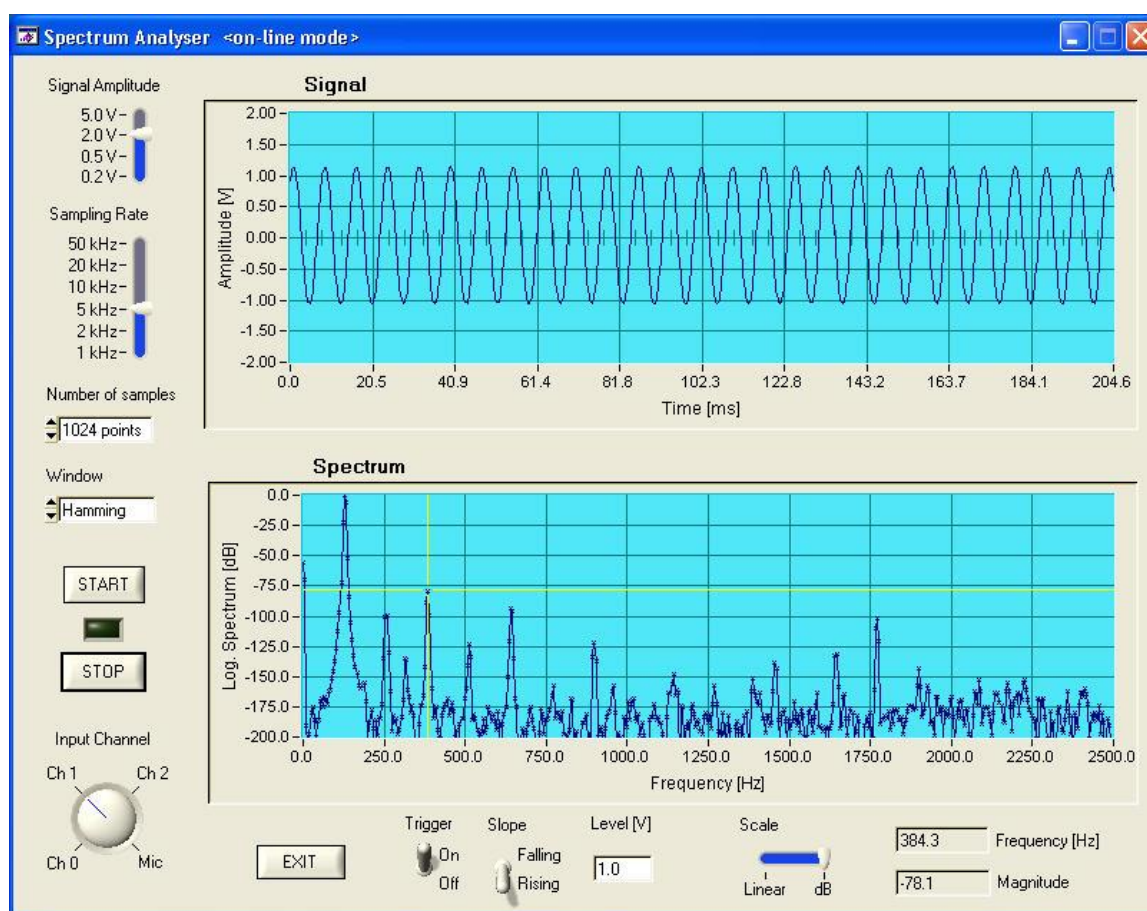
Inny przykład wirtualnego przyrządu pomiarowego, stanowi dwukanałowy analizator widma (rys.3.16)⁶. Wykorzystuje on również kartę LAB 1200 i powstał z użyciem oprogramowania narzędziowego LabWindows/CVI.

⁶ Praca zrealizowana w Zakładzie Systemów Informacyjno-Pomiarowych PW.



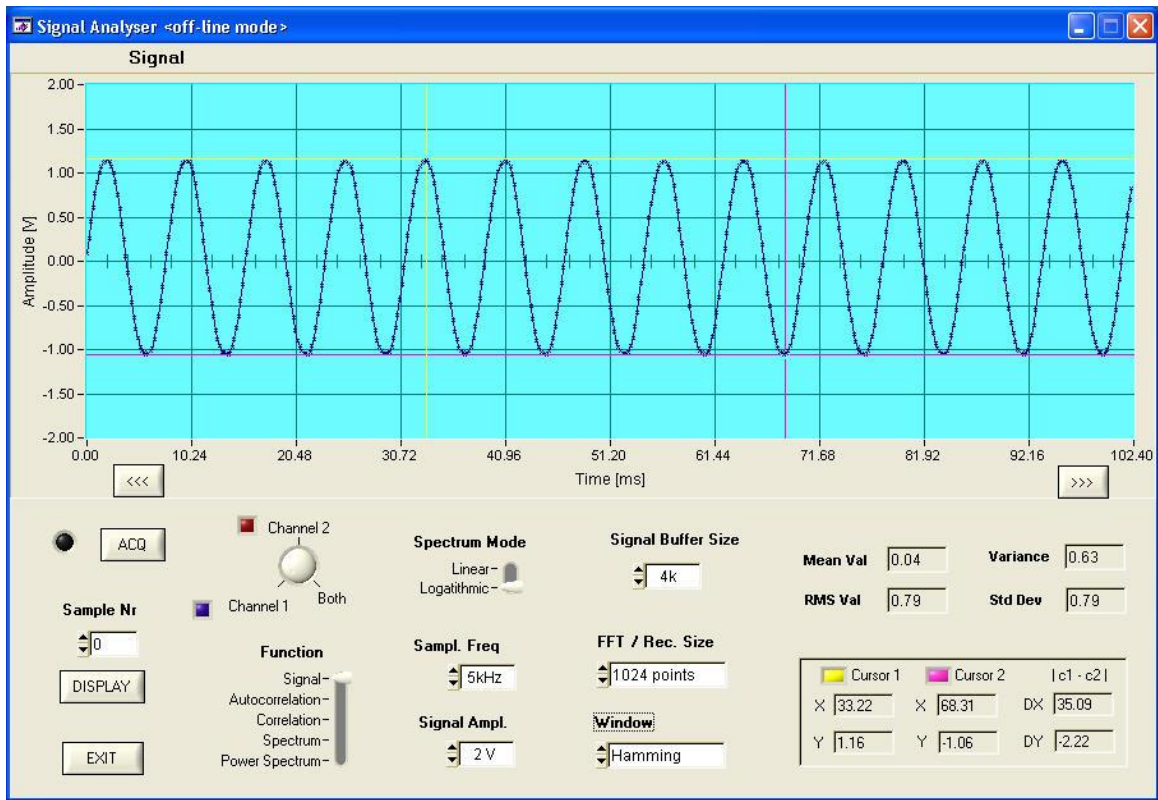
Rys.3.16 Schemat blokowy wirtualnego analizatora widma

Płytę czołową przyrządu do pracy w trybie „on-line”, gdzie jednocześnie obserwuje się przebieg sygnału oraz jego widmo, przedstawia rysunki 3.17.

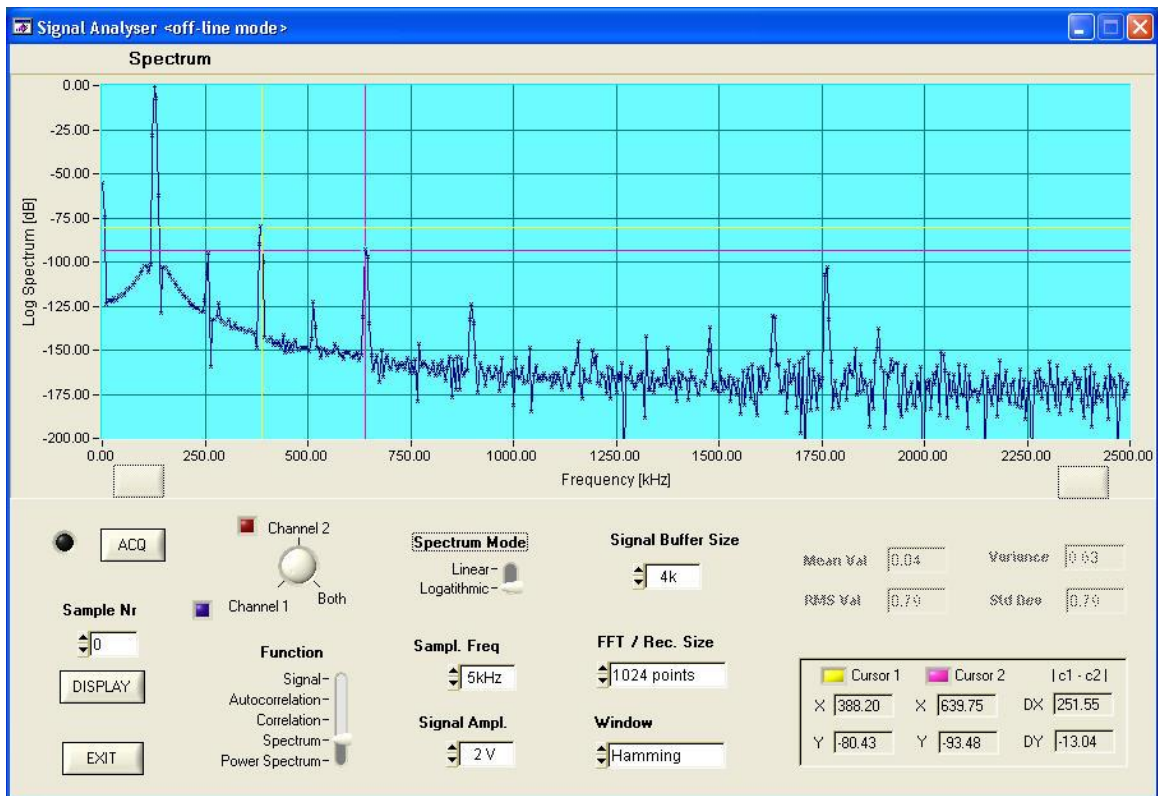


Rys.3.17 Płyta czołowa wirtualnego analizatora widma - tryb „on-line”

Płyty czołowe dla trybu „off-line” przedstawione są na rysunkach 3.18 i 3.19. W pierwszym wariacie na ekranie płyty zobrazowany jest fragment przebiegu czasowego sygnału wybrany do dalszych analiz ze znacznie dłuższej realizacji czasowej. W wariacie drugim na ekranie zobrazowane jest widmo wybranego fragmentu sygnału. Obserwacja płyt czołowych w każdym przypadku umożliwia zapoznanie się z funkcjami przyrządu, jakie są aktualnie dostępne. Występowanie kilku „płyt czołowych” (do wyboru), jest dowodem na szeroki zakresu funkcjonalności przyrządu. Szczęólnego znaczenia ten fakt nabiera w przypadku zastosowań dydaktycznych.



Rys.3.18 Płyta czołowa wirtualnego analizatora widma - tryb „off-line”, zobrazowanie sygnału



Rys. 3.19 Płyta czołowa wirtualnego analizatora widma - tryb „off-line”, zobrazowanie widma

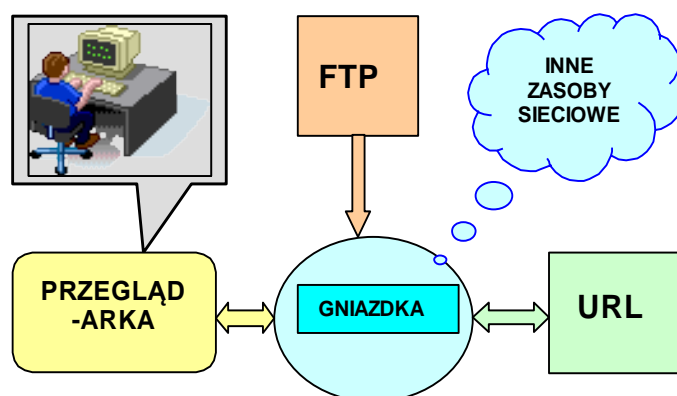
4 Systemy rozproszone

Nie ulega wątpliwości, że nowoczesne osiągnięcia w dziedzinie techniki komputerowej i technologii informacyjnej powinny i są z sukcesem przyjmowane w dziedzinie metrologii [34]. Rozwój komputerowych systemów pomiarowych, a w szczególności zdefiniowanie nowoczesnej architektury wirtualnego przyrządu pomiarowego znacząco przyczyniły się do powstania nowoczesnej struktury systemu pomiarowego rozproszonego terytorialnie. Jest jasnym, że rozproszone systemy pomiarowo-kontrolne zbudowane w sieci komputerowej będą nabierać na znaczeniu. Podstawowe cechy takich systemów to:

- współużywanie pamięci masowych do archiwizacji danych dostarczonych z poszczególnych stanowisk,
- współużywanie zasobów sprzętowych zainstalowanych w sieci, pozwalające na optymalizację ich wykorzystania.
- Możliwość funkcjonalnej integracji rozproszonych zasobów pomiarowych.

Co więcej, wielokomputerowe systemy obliczeniowe mogą być bardzo przydatne w zakresie tworzenia złożonych systemów pomiarowo-kontrolnych (*pomiarowo-sterujących*, czy wręcz *informacyjno-pomiarowych*), gdyż pozwolą przezwyciężyć ograniczenia związane z mocą obliczeniową oraz liczbą zbieranych i przetwarzanych danych.

Systemy zorganizowane w ramach lokalnej sieci komputerowej, można w prosty sposób rozszerzyć na sieć globalną. Rysunek 4.1, w sposób symboliczny, ilustruje ideę dostępu do wielorakich zasobów sieciowych.



Rys.4.1 Tradycyjny sposób dostępu do zasobów sieciowych: **GNIAZDKA** – sposób komunikacji sieciowej **FTP** – File Transfer Protocol, **URL** - Uniform Resource Locator

Istniejące oprogramowanie internetowe, może być w łatwy sposób wykorzystane do organizacji migracji danych pomiędzy różnymi obiektami sieci. Standardowe języki programowania jak C++ i Java mogą być wykorzystane w narzędziowych środowiskach programistycznych do projektowania aplikacji sieciowych i internetowych. Protokoły internetowe, technologia ethernetowa oraz gniazdka (ang: *sockets*) to elementy wykorzystywane do projektowania struktury sieciowej.

Gniazdka, to wywodząca się z systemu UNIX, a używana w środowisku Windows (jako *winsock*) technologia programowania, która umożliwia dzielenie wszystkich zasobów pomiarowych w obrębie sieci komputerowej, w sposób tak prosty jak proste jest zapisywanie danych w pliku. Wykorzystuje ona URL (*Uniform Resource Locator*) do adresowania danych na tej samej zasadzie jak używa się go do wskazywania stron internetowych w przeglądarce internetowej. Gniazdka osadzone w konkretnym środowisku programistycznym są narzędziem

idealnym do realizacji kontroli migracji danych pomiarowych w obrębie sieci, jeżeli chce się uniknąć wnikania w zawite szczegóły protokołu transmisji danych typu TCP/IP (*Transmission Control Protocol / Internet Protocol*).

Rozproszone systemy pomiarowo-kontrolne, osadzone w globalnej sieci komputerowej, stanowią szeroką bazę budowania zdalnego dostępu do laboratorium i wreszcie tzw. *laboratorium wirtualnego*. Są to nowoczesne elementy procesu dydaktycznego, które mogą być wykorzystane nie tylko w modelu kształcenia na odległość, lecz również do wspomagania kształcenia tradycyjnego, stacjonarnego. Elementy te, wyraźnie wpisują się do modelu globalnego społeczeństwa informacyjnego.

Trzeba jednak pamiętać, że niezależnie od stosowanych technik programistycznych, we wszystkich modelach rozproszonych systemów pomiarowych realizowanych w obrębie sieci komputerowych krytyczne znaczenie, na etapie projektowania systemu, ma uwzględnienie problemu realizacji operacji w czasie rzeczywistym.

4.1 Idea rozproszonego systemu pomiarowego

Rozproszony system pomiarowy (ang. *distributed measurement system*) zwykle definiowany jest jako system rozłożony terytorialnie. Wydaje się jednak, że obecnie bardziej adekwatne byłoby opisanie go jako zbioru samodzielnych urządzeń (lub pojedynczych systemów pomiarowych), połączonych za pomocą odpowiedniego interfejsu i wyposażonych w oprogramowanie, umożliwiające koordynowanie ich działań oraz swobodny dostęp do baz danych i badanych obiektów. Termin *rozproszony* odnosi się w takim przypadku nie tylko do przestrzeni, lecz również rozwiązań funkcjonalnych.

Postęp w dziedzinie techniki komputerowej i technologii informacyjnej, a w szczególności rozwój standardów komunikacyjnych pozwolił na ewolucję scentralizowanych systemów badawczych do rozproszonych i otwartych systemów pomiarowo-kontrolnych. Elementami takich systemów są obecnie inteligentne urządzenia, bazujące na układach mikroprocesorowych, w tym komputery osobiste. Wyróżniają się one zdolnością do przetwarzania informacji i dwukierunkowej komunikacji między urządzeniami, wchodzącymi w skład systemu. Dodatkową stymulacją rozwoju takich systemów są rosnące wymagania związane z jakością, niezawodnością, ochroną środowiska, elastycznością produkcji oraz stałym dążeniem do wzrostu wydajności i obniżania kosztów.

Początkowo w systemach pomiarowo-kontrolnych dominowała komunikacja analogowa, pozwalająca budować systemy o architekturze multipleksowej. Ten sposób komunikacji miał jednak ograniczenia wynikające z tego, że przesyłana wartość wielkości mierzonej była zawarta w parametrze sygnału, transmisja była jednokierunkowa i każdy punkt pomiarowy wymagał oddzielnej pary przewodów. Opracowanie i wdrożenie do produkcji na dużą skalę urządzeń mikroprocesorowych dało możliwość przesyłania informacji w postaci cyfrowej. Dane w postaci cyfrowej mają bardzo korzystne właściwości takie, jak: odporność na zakłócenia, łatwość i większa dokładność ich przetwarzania. Transmisja informacji w postaci cyfrowej pozwoliła na: dwukierunkową wymianę danych, podłączenie do jednego przewodu znacznej liczby urządzeń, łatwą rozbudowę, zdalną konfigurację, diagnostykę i serwis.

Znaczące zalety komunikacji cyfrowej spowodowały jej szybką popularyzację. W wyniku tego od początku lat dziewięćdziesiątych stało się możliwe budowanie rozproszonych systemów pomiarowych o architekturze nie tylko multipleksowanej, ale również sieciowej i oczywiście mieszanej.

Architektura sieciowa dała nowe możliwości konfiguracji rozproszonych systemów pomiarowych i spotkała się z dużym zainteresowaniem zarówno producentów, jak i odbiorców aparatury pomiarowej. Zróżnicowanie wymagań, stawianych przez obiekty, procesy technologiczne i aplikacje informatyczne spowodowało powstanie wielu różnorodnych standardów komunikacyjnych stosowanych w systemach pomiarowo-kontrolnych. W chwili obecnej w centrum uwagi użytkowników i producentów tych systemów znajduje się problem wzajemnej

integracji różnych standardów tak, aby możliwe było wykorzystywanie aplikacji i urządzeń produkowanych przez różne firmy.

4.2 Cechy systemów rozproszonych

W rozumieniu ogólnym, uwarunkowanym istnieniem *technologii informatycznej*, system rozproszony, jest zbiorem samodzielnych komputerów połączonych za pomocą sieci i wyposażonych w rozproszone oprogramowanie systemowe. W praktyce, systemy rozproszone są realizowane z wykorzystaniem sprzętu, którego liczność zmienia się od kilku stacji roboczych, poprzez pojedyncze sieci lokalne, aż po Internet – światowy zbiór sieci lokalnych i rozległych, składający się z milionów komputerów.

Oprogramowanie systemu rozproszonego umożliwia komputerom koordynowanie ich działań oraz dzielenie zasobów systemu takich jak: sprzęt komputerowy, oprogramowanie, bazy danych. Dodanie do takiego systemu odpowiednio przystosowanych urządzeń pomiarowo-kontrolnych tworzy system pomiarowy rozproszony terytorialnie. Dobrze zaprojektowany rozproszony system pomiarowy odbierany jest jako jedno, zintegrowane środowisko obliczeniowe i pomiarowo-kontrolne. O użyteczności rozproszonych systemów pomiarowych, osadzonych w sieci komputerowej, decyduje sześć podstawowych cech przeniesionych z dziedziny technologii informatycznej [7]:

- Współużywanie zasobów,
- Otwartość,
- Współbieżność,
- Skalowalność,
- Tolerowanie uszkodzeń,
- Przeźroczystość.

4.2.1 Dzielenie zasobów

Termin *zasób* jest raczej abstrakcyjny, jednak charakteryzuje on najlepiej zbiór obiektów fizycznych i logicznych, które mogą być z pożytkiem dzielone (wspólnie wykorzystywane) w systemie. Zbiór ten obejmuje:

- Sprzęt (drukarki, dyski, przyrządy pomiarowe itp.),
- Dane (pliki, dane pomiarowe, bazy danych, itp.)

Wszystkie zasoby, które mają być efektywnie wykorzystane, muszą być zarządzane przez program obsługujący interfejs komunikacyjny, umożliwiającą dostęp do zasobu oraz działanie na nim. Każdy typ zasobu wymaga odrębnego schematu opisującego ten dostęp. Użytkownicy komunikują się z zarządcami zasobów w celu uzyskania dostępu do nich. Proces ten najlepiej charakteryzuje model *klient/serwer* i jest najczęściej stosowany w praktyce.

4.2.2 Otwartość

Otwartość systemu charakteryzuje jego zdolność do rozszerzania różnych zasobów, na przykład dodawanie:

- urządzeń zewnętrznych (w tym przyrządów pomiarowych),
- bloków pamięci,
- kart rozszerzeń (w tym kart zbierania danych),

- protokołów komunikacyjnych,
- oprogramowania (czyli nowych właściwości dla systemu).

Otwartość rozproszonych systemów pomiarowych, w tym kontekście, jest charakteryzowana przede wszystkim przez możliwość dodawania nowych usług i protokołów, a także urządzeń pomiarowych bez naruszania lub znacznej zmiany istniejących rozwiązań.

Dalszą korzyścią, często cytowaną przy okazji systemów otwartych, jest niezależność od poszczególnych dostawców czy producentów. Osiąga się to między innymi przez stosowanie standardowych i powszechnie używanych interfejsów. Najbardziej zaawansowaną inicjatywą dążącą do stworzenia podwalin systemu otwartego w środowisku sieci przemysłowych jest opracowanie zestawu norm IEEE 1451.x.

4.2.3 Współbieżność

W przypadku, gdy na jednym komputerze działa wiele procesów, to mówimy, że są one wykonywane współbieżnie. W systemach rozproszonych istnieje wiele komputerów i mogą wystąpić następujące działania równoległe:

- wielu użytkowników jednocześnie wydaje polecenia lub współpracuje z programami użytkowymi,
- wiele procesów serwera działa współbieżnie, odpowiadając na zamówienia pochodzące od różnych klientów.

Pierwszy przypadek zachodzi wtedy, gdy na zamówienie każdego z aktywnych użytkowników działa jedna lub więcej aplikacji. Drugi przypadek wynika z istnienia jednego lub więcej procesów usługowych dla każdego typu zasobu. Działają one na ogół w dodatkowych komputerach, pozwalając każdemu procesowi usługowemu pracować równoległe z innymi serwerami oraz procesami wykonywanymi na stacjach roboczych.

4.2.4 Skalowalność

Systemy rozproszone mogą działać skutecznie i wydajnie w wielu różnych skalach. Najmniejszy praktyczny system rozproszony składa się ze stacji roboczej i serwera, natomiast system rozproszony skonstruowany z wykorzystaniem sieci lokalnej, może zawierać kilkadziesiąt stacji roboczych i kilka serwerów. Kilka sieci lokalnych łączy się często ze sobą tworząc intersieci, które mogą zawierać wiele tysięcy komputerów składających się na jeden system rozproszony.

Potrzeba skalowalności nie jest jedynie problemem sprzętowym lub wydajności sieci. Zagadnienie to przenika niemal wszystkie aspekty projektowania systemu rozproszonego. W scentralizowanych systemach komputerowych niektóre wspólne zasoby, jak pamięć, procesory czy kanały wejścia/wyjścia, są udostępnione z ograniczeniami i nie mogą być powielane bez końca. W systemach rozproszonych ograniczenia te są usuwane automatycznie i wydaje się, że liczba komputerów jest praktycznie niemal nieograniczona, lecz wprawdzie trzeba jawnie przewidzieć potrzeby skalowalności. Powiększanie skali systemu i oprogramowania użytkowego nie powinno pociągać za sobą konieczności wykonywania w nich zmian.

4.2.5 Tolerowanie uszkodzeń

Systemy komputerowe mogą ulegać awariom. Wskutek uszkodzenia sprzętu lub oprogramowania mogą generować błędne wyniki lub zatrzymywać się (zawieszać), nie ukończywszy zamierzonych obliczeń. Są dwa tryby, projektowania systemów komputerowych tolerujących uszkodzenia, które mogą być użyte jednocześnie:

- nadmiarowość sprzętowa,

- odtwarzanie programowe.

Aby uniezależnić się od awarii sprzętu, w miejsce pojedynczej sekcji stosuje się dwa połączone ze sobą komputery, z których jeden działa jako zapasowy. W systemach rozproszonych nadmiarowość można planować z większą precyzją. Zwiłokrotnianie może dotyczyć wybranych serwerów o istotnym znaczeniu dla nieprzerwanego działania ważnych procesów.

Odtwarzanie programowe polega na użyciu programów do usuwania skutków uszkodzeń i przywracania stanu trwałych danych po wystąpieniu awarii. Trzeba też pamiętać, że obliczenia wskutek wystąpienia awarii mogą być niekompletne, a przez to aktualizowane pliki mogą stracić spójność. Pomocne może być w tym przypadku użycie systemu śledzenia transakcji.

4.2.6 Przeźroczystość

Przeźroczystość definiuje się jako ukrywanie przed użytkownikiem i programistą różnorodności w realizacji oddzielnych składowych aplikacji w systemie rozproszonym tak, aby system był postrzegany jako całość, a nie zbiór niezależnych składowych. Konsekwencje przeźroczystości wywierają poważny wpływ na model oprogramowania. Wyróżnia się następujące standardy przeźroczystości:

- przeźroczystość dostępu: umożliwia dostęp do lokalnych i odległych obiektów za pomocą takich samych operacji,
- przeźroczystość położenia: umożliwia dostęp do danych bez znajomości miejsca ich przechowywania,
- przeźroczystość współbieżności: umożliwia wielu procesom niezakłócone działanie współbieżne z użyciem wspólnych obiektów,
- przeźroczystość zwiłokrotniania: pozwala na użycie wielu kopii obiektów w celu zwiększenia niezawodności i wydajności bez wiedzy użytkowników i programów użytkowych,
- przeźroczystość awarii umożliwia ukrywanie uszkodzeń, pozwalając użytkownikom i programom użytkowym na kończenie zadań, pomimo awarii sprzętu lub składowych oprogramowania,
- przeźroczystość wędrówki pozwala na przemieszczanie obiektów w obrębie systemu, bez wpływu na działania użytkowników lub programów użytkowych,
- przeźroczystość wydajności umożliwia rekonfigurowanie systemu w celu poprawy jakości działania przy zmianie obciążenia,
- przeźroczystość skalowania umożliwia zmianę skali, bez zmiany struktury poszczególnych elementów systemu i algorytmów użytkowych.

4.3 Rozproszony system pomiarowy w sieci komputerowej

4.3.1 Charakterystyka ogólna

Pierwszym ograniczeniem w projektowaniu złożonych komputerowych systemów pomiarowych, w rozumieniu tradycyjnym, jest liczba i różnorodność mierzonych sygnałów. Gdy procedura pomiarowa wymaga identyfikacji bardzo wielu różnorodnych sygnałów, architektura systemu na pojedynczym komputerze okazuje się niewystarczająca (np. do pomieszczenia wielu kart zbierania danych). Użycie wirtualnych przyrządów pomiarowych i osadzenie ich w sieciowych podsystemach zbierania danych pozwala na znaczne rozszerzenie tych możliwości.

Drugim ograniczeniem jest obszar, jaki może być „obsłużony” przez pojedynczy komputer. I tak, aby zapewnić odpowiednią jakość (w szczególności poziom) sygnałów doprowadzanych do kart zbierania danych, czujniki i układy kondycjonowania nie mogą być zbyt oddalone od źródeł sygnałów. Jeśli sygnały muszą być pobierane z oddalonych czujników, rozproszenie podsystemów zbierania danych także w tym przypadku może być bardzo pomocne.

Kolejne ograniczenie związane jest ze złożonością obliczeniową algorytmów przetwarzania danych pomiarowych. Im proces jest bardziej złożony tym moc obliczeniowa musi być większa, a przez to droższa - aby wykonać zadania stawiane przez aplikacje (w szczególności, gdy chodzi o spełnienie warunku czasu rzeczywistego). Jak już wspomniano przy okazji omawiania konfiguracji wirtualnych przyrządów pomiarowych, w celu wspomżenia głównego procesora, instaluje się procesory DSP. W przypadku zaś, gdy proces przetwarzania staje się jeszcze bardziej wymagający, pojedynczy, szybki komputer może okazać się niewystarczający lub zbyt kosztowny. Rozdział mocy obliczeniowej na kilka niezależnych komputerów połączonych siecią lokalną może okazać się jedynym efektywnym rozwiązaniem. Oczywiście, trzeba zadbać o spełnienie takich warunków współpracy, które gwarantują równoległość i ciągłość procesów. Rozproszenie systemu w sieci lokalnej umożliwi stosunkowo łatwe spełnienie wymogu ciągłości komunikacji. Obciążenie sieci wynikłe z tytułu obsługi przyrządów wirtualnych lub złożonych z nich podsystemów, nawet w tanich sieciach komercyjnych (Ethernet) jest zwykle bardzo małe i nie powoduje opóźnień w komunikacji. W przypadku, gdy czas pomiarów ma znaczenie krytyczne, w całym procesie pomiaru, to dane mogą zostać uzupełnione informacją o czasie ich pobrania. Na szczęście zegary komputerów mogą być zsynchronizowane łatwo i precyzyjnie. Pełna lokalizacja w czasie pobranych danych oraz krótki czas odpowiedzi sprawia, że systemy rozproszone w sieci lokalnej są odpowiednie do monitorowania i nadzoru procesów również w czasie rzeczywistym. W efekcie można odnieść wrażenie, że dane pochodzą z pojedynczego przyrządu wirtualnego.

Pewne problemy mogą pojawić się dopiero w przypadku bardzo dużych firm jak np. zakłady chemiczne lub rafinerie naftowe. Nawet użycie przeznaczonej wyłącznie do tego celu sieci komputerowej może nie być wystarczające do zminimalizowania opóźnień przesyłu danych jak również do zapewnienia pełnej synchronizacji procesów. Wtedy ogólna „spójność” obsługi procesu produkcyjnego może być spełniona jedynie dla procesu o małej dynamice. Alternatywę stanowi podzielenie obszaru produkcji na „komórki” przetwarzania danych i kontrolowanie każdej komórki przez dedykowany podsystem. Współpraca podsystemów i koordynacja całego procesu może doprowadzić do uzyskania wspomnianej spójności. W tym przypadku indywidualne „komórki” przetwarzające dane pracują w czasie rzeczywistym, podczas gdy protokół połączeń między komórkami zapewnia całościową koordynację na wyższym poziomie.

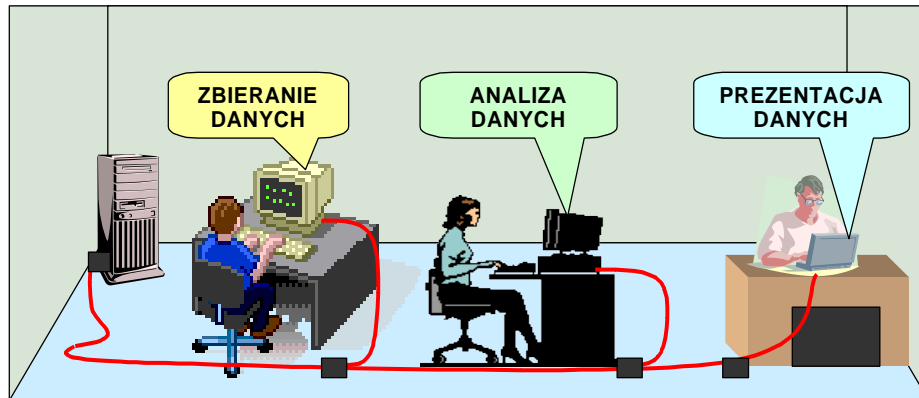
Kolejny krok to znaczące rozproszenie systemu pomiarowo-kontrolnego w sensie terytorialnym. Może to dotyczyć np. systemu stacji meteorologicznych. Do tego celu należy użyć sieciowego systemu pomiarowo-kontrolnego dołączonego do sieci globalnej, Internetu. Proces rozproszony terytorialnie, podzielony jest na komórki obsługiwane przez pojedynczy komputer lub grupę komputerów połączonych w sieć lokalną. Poszczególne jednostki łączone są w sieć rozległą - tworząc rozproszony w sieci globalnej system pomiarowo-kontrolny. W tym przypadku przerwy w komunikacji mogą być niezwykle uciążliwe. W konsekwencji tylko lokalny nadzór staje się gwarantem poprawności operacji w czasie rzeczywistym. Połączenia internetowe istotnie mogą służyć globalnej kontroli i obserwacji procesu – ale raczej nie w czasie rzeczywistym. W globalnym monitorowaniu procesu użytkownik musi być świadom ewentualnego braku synchronizmu danych, przychodzących z rozproszonych jednostek przetwarzania.

Celem zapewnienia poprawnej pracy systemu, podczas projektowania systemów rozproszonych w sieci globalnej, trzeba wziąć pod rozwagę wszystkie wymienione wyżej ograniczenia, a w szczególności problemy wynikające z ewentualnej konieczności obsługi procesu w czasie rzeczywistym. Coraz bardziej powszechne i prawdopodobne staje się stwierdzenie, że operacje w czasie rzeczywistym możliwe są do realizacji nawet w sieci

globalnej. Jest to prawdziwe dopiero po spełnieniu pewnych trudnych, restrykcyjnych warunków. Sygnały próbkowane pochodzące z różnych systemów, które nie mają wspólnego zegara synchronizującego, nie są pobierane w tym samym czasie. W algorytmach przetwarzających te dane trzeba uwzględnić („zanotowany”) dokładny czas pobrania próbki.

Reasumując należy stwierdzić, że rozwój systemów pomiarowo-kontrolnych zawierających wirtualne przyrządów pomiarowe i zaawansowane algorytmy cyfrowego przetwarzania sygnałów umożliwił stosunkowo łatwe ich usytuowanie w lokalnej i globalnej sieci komputerowej. Dalszy rozwój tych systemów, jak zwykle, stymulowany jest ciągłym powiększaniem szybkości i mocy obliczeniowej procesorów, pojemności pamięci, a także uproszczeniami procesu projektowania algorytmów pomiarowych, interfejsu użytkownika i protokołów komunikacyjnych. Decydujące znaczenia ma jednak postęp w zakresie projektowania, osadzonych w sieci systemów czasu rzeczywistego. Wszystko to oznacza, że postęp w dziedzinie rozproszonych systemów pomiarowo-kontrolnych związany jest z rozwojem techniki komputerowej oraz technologii informacyjnej. Coraz bardziej adekwatna staje się nazwa *sieciowe systemy kontrolno-pomiarowe*, czy wręcz *systemy informacyjno-pomiarowe*.

Przyjazny sposób podsumowania powyższych rozważań stanowi rysunek 4.2, łączący ideę systemu rozproszonego z wirtualnym przyrządem pomiarowym w ujęciu modernistycznym. Aplikacje pomiarowo-kontrolne nie są „uwięzione” w pojedynczym komputerze. Upřednio rozdzielone funkcje zbierania, analizy i przetwarzania danych rozproszone są w obrębie lokalnej sieci komputerowej, wewnętrznej sieci *intranet*, a nawet sieci globalnej, Internet. Każda z tych funkcji również nie musi być ograniczona do pojedynczego komputera. Z punktu widzenia metrologa najbardziej istotne jest to, że ma on teraz bezpośredni dostęp do danych kolegi, może wysłać i pobrać dane pomiarowe z bazy danych lub pozyskać z całkiem innej aplikacji.



Rys.4.2 Rozproszony system pomiarowy (wirtualny przyrząd pomiarowy) w sieci LAN

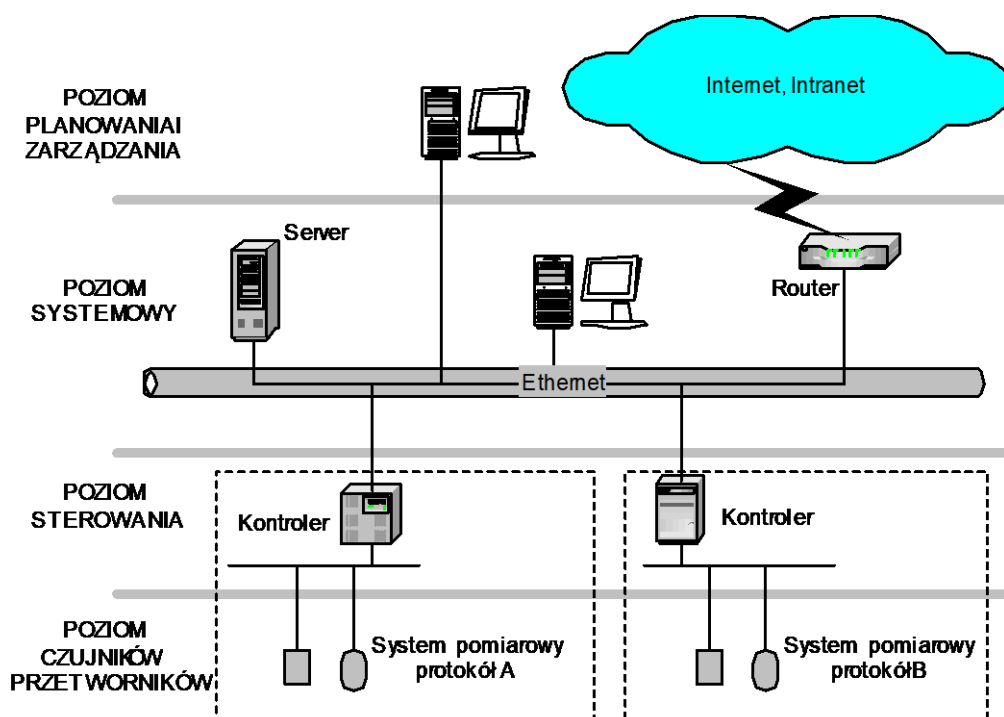
4.3.2 Przykładowa architektura systemu rozproszonego

Integracja sieci komputerowych z systemami pomiarowymi daje możliwość tworzenia wielopoziomowych, i rozbudowanych struktur informacyjno-pomiarowych. Systemy takie, dzięki rozwiniętym standardom protokołów i interfejsów sieciowych, charakteryzują się dużą skalowalnością i otwartością. Najważniejszą ich zaletą jest jednak to, że dzięki technologiom sieci lokalnych, intranetowym i internetowym, można budować systemy, które pozwalają na monitorowanie, nadzór oraz koordynację rozproszonych terytorialnie procesów produkcyjnych i badawczych.

Na rys. 4.3 przedstawiona jest architektura nowoczesnego systemu, który jest rozproszony nie tylko w sensie terytorialnym, ale również w sensie zarządzania, sterowania i kontroli. W systemie takim można wyróżnić cztery warstwy. Najniższe z nich, czyli warstwy *sterowania* i *czujników*, zawierają klasyczne systemy pomiarowe.

Podsystemy te są zarządzane przez autonomiczne, programowalne kontrolery np. sterowniki PLC (ang. *Programmable Logic Controller - PLC*) lub komputery ogólnego przeznaczenia (w tym osobiste) z odpowiednimi kartami interfejsowymi. Każdy z podsystemów może zawierać własny interfejs komunikacyjny (RS-488, IEC-625.2, Profibus, Ethernet itp.), który oferuje najlepsze właściwości do badania danego obiektu. Wybrane informacje z poszczególnych podsystemów przekazywane są do warstwy trzeciej.

Warstwa trzecia, tzw. *systemowa*, zbudowana jest w oparciu o sieć komputerową. Znajdują się w niej serwery aplikacji, serwery baz danych oraz stanowiska do kontroli i serwisu całego systemu. Głównym zadaniem sprzętu i oprogramowania, znajdującego się na tym poziomie jest skoordynowanie działań całej struktury, odpowiednie przetworzenie danych (np. algorytmami CPS), współdzielenie zasobów do przetwarzania rozproszonego, reakcja na typowe zdarzenia awaryjne i umożliwienie pełnej kontroli nad systemem warstwie *zarządzania i planowania*.



Rys.4.3 Architektura rozproszonego w sieci systemu pomiarowo-kontrolnego

Kontrolery warstwy drugiej podłączone są do sieci komputerowej, która pełni rolę ośrodka integrującego cały system, bezpośrednio lub za pomocą specjalnych urządzeń, spełniających funkcje bram (ang. *gateway*). Technologie sprzętowo-programowe połączeń z siecią komputerową są związane z protokołami komunikacyjnymi poszczególnych podsystemów (ang. *application specific*). Oznacza to dużą różnorodność rozwiązań, co ma wpływ na skalowalność i otwartość struktury. Jednak występują już na rynku kompleksowe rozwiązania sieciowych systemów komunikacyjnych, umożliwiające efektywną i prostą integrację całego środowiska. Do najbardziej zaawansowanych technologii należą rozwiązania firm: Hewlett-Packard (Industrial Ethernet), Hirschmann (Industrial Networking), National Instruments (DataSocket) i Sun Microsystems (Java, Jini).

Obecnie, najbardziej rozpowszechnioną technologią lokalnej sieci komputerowej (LAN: Local Area Network) jest Ethernet. Sieci tego typu charakteryzują się dużą szybkością transmisji, prostą konstrukcją, dużą niezawodnością i niską ceną. Ethernet nie został zaprojektowany do komunikacji w systemach pracujących w czasie rzeczywistym, dlatego wykorzystanie jego standardowej wersji jako interfejsu układu pomiarowego budzi czasami kontrowersje. Jest to spowodowane głównie sposobem arbitrażu przy transmisji. Stosowane jest w nim wykrywanie fali nośnej i detekcja kolizji tzw. *CSMA/CD* (ang. *Carrier Sense with Multiple Access / Collision*

Detection). Sposób ten nie gwarantuje determinizmu czasowego. Jednak zmiany w specyfikacji *IEEE-802*, dotyczącej Ethernetu spowodowały, że technologia ta może być z powodzeniem wykorzystywana w strukturach rozproszonych systemów pomiarowych na poziomie systemowym i w niektórych przypadkach na poziomie sterowania. Standard ten gwarantuje, że efektywny czas dostarczenia wiadomości nie przekracza czterech milisekund. Przy oferowanej obecnie przez Ethernet szybkości można powiedzieć, iż jest to rozwiązanie konkurencyjne dla wielu dedykowanych interfejsów przemysłowych.

Wykorzystanie sieci komputerowych w rozproszonych systemach pomiarowych zostało szeroko przebadane zarówno na uczelniach, jak i w przemyśle. Zauważalnym obecnie trendem w takich systemach jest przejście od wyspecjalizowanych, centralnie zarządzanych układów do bardziej elastycznych, otwartych i skalowalnych, mających zdecentralizowane sterowanie i przetwarzanie. Coraz częściej wykorzystywane są w nich zorientowane obiektowo platformy i języki wysokiego poziomu. Wykorzystuje się w nich nie tylko sieci lokalne, ale również technologie intranetowe i internetowe.

4.4 Typowe protokoły komunikacyjne i obiekty sieciowe

Protokoły komunikacyjne to zbiór ścisłych reguł i kroków postępowania, które są automatycznie wykonywane przez urządzenia komunikacyjne w celu nawiązania łączności i wymiany danych. Dzięki temu, że połączenia z użyciem protokołów odbywają się całkowicie automatycznie, typowy użytkownik zwykle nie zdaje sobie sprawy z ich istnienia i nie musi o nich nic wiedzieć.

Klasyczne protokoły, których pierwowzorem był protokół teleksu składają się z trzech części:

- procedury powitalnej (tzw. "handshake"), która polega na przesłaniu wzajemnej podstawowej informacji o łączących się urządzeniach, ich adresu (np. nr telefonu), szybkości i rodzaju transmisji itp.,
- właściwego przekazu danych,
- procedury analizy poprawności przekazu (np. sprawdzania sum kontrolnych) połączonej z procedurą pożegnania, żądaniem powtórzenia transmisji lub powrotem do procedury powitalnej.

Przesyłana informacja może być porcjowana - protokół musi „umieć” odtworzyć informację w postaci pierwotnej. Protokołami tego rodzaju posługują się: teleksy, faksy, modemy, programy komputerowe, wiele innych urządzeń, włącznie z pilotami do telewizorów. Protokoły służące programom komputerowym do porozumiewania się między sobą poprzez Internet są określone przez IETF (Internet Engineering Task Force) w dokumentach zwanych RFC (Request For Comments).

Osobną klasą protokołów komunikacyjnych są protokoły do komunikacji grupowej (multicast), używane m.in. do transmisji telewizyjnych przez Internet, telekonferencji itp. Przykładami takich protokołów są RMTP (Reliable Multicast Transport Protocol), TOTEM, XTP, Muse i inne.

Przesyłanie danych komputerowych to niezwykle trudny proces, dlatego rozdzielono go na kilka "etapów" - warstw. Warstwy oznaczają w istocie poszczególne funkcje spełniane przez sieć.

Poszczególne warstwy, z których składa się sieć komputerowa, zwykło się opisywać zgodnie z koncepcją siedmiowarstwowego modelu połączeń systemów otwartych OSI (*Open Systems Interconnection*) zatwierdzonego przez ISO (*International Organization for Standardization*). W skład modelu OSI wchodzi następujące warstwy: fizyczna, kanałowa, sieciowa, transportowa, sesji, prezentacji, zastosowań. Protokoły Internetu opisywane są za pomocą zredukowanej, czterowarstwowej struktury: fizyczna/kanałowa, sieciowa, transportowa, zastosowań [8]. Poniżej, podano wykaz typowych sieciowych protokołów komunikacyjnych:

IP - (ang. *Internet Protocol*). Protokół IP jest podstawą architektury Internetowej. Na protokół ten składają się operacje: kierowania pakietu do właściwej sieci i obiektu typu *host* (routing), tłumaczenia adresów obiektów *host* na adresy fizyczne związane z daną siecią (protokół arp), segmentacja i składanie pakietów danych.

UDP - (ang. *User Datagram Protocol*) jest to prosty protokół korzystający (w sposób mało wyszukany) z warstwy IP. Jest zawodny – nie daje żadnej pewności, że *datagram* dotrze do adresata i że kolejność odbierania *datagramów* będzie taka jak nadawania. Jego zaleta to szybkość – systemy czasu rzeczywistego.

TCP - (*Transmission Control Protocol*) realizuje transmisje w trybie połączeniowym. Oznacza to, że między komunikującymi się obiektami typu *host* tworzone jest wirtualne połączenie. Protokół TCP utrzymuje je i zapewnia niezawodny transfer danych między komputerami. Realizowane jest to poprzez operacje potwierdzania odbioru pakietów i retransmitowania pakietów zagubionych. W rezultacie powstaje pewność dostarczenia danych do odległego komputera.

NTP - (ang. *Network Time Protocol*), protokół czasu sieciowego – określa architekturę usług czasu oraz protokół rozpowszechniania informacji o czasie w obrębie różnorodnych, połączonych sieci. Przyjęty jako standard synchronizowania zegarów w sieci Internet. Usługi NTP są dostarczane przez sieć serwerów. Serwery podstawowe są bezpośrednio podłączone do źródeł czasu w rodzaju zegarów radiowych odbierających sygnały czasu uniwersalnego UCT (francuski: *Universelle Tempes Coordinate*)⁷.

RTP – (*Real Time Protocol*), protokół transmisji sygnału w czasie rzeczywistym. Protokół ten, różny od TCP, został stworzony do przesyłania w sieci Internet danych czasu rzeczywistego: *strumieni multimedialnych* (dźwięku i obrazu). Chodziło o to, aby możliwe było odtwarzanie dźwięku czy obrazu bez wcześniejszego „ściągnięcia” całego pliku z sieci. Został on szeroko wykorzystany do realizacji wideokonferencji lub nadawania audycji radiowych w sieci Internet.

Model TCP/IP (*Transmission Control Protocol / Internet Protocol*) – teoretyczny model warstwowej struktury protokołów komunikacyjnych. Model TCP/IP został stworzony w latach 70. XX wieku w DARPA (*Defense Advanced Research Projects Agency*), aby pomóc w tworzeniu odpornych na atak sieci komputerowych. Potem stał się on podstawą struktury Internetu.

Podstawowym założeniem modelu TCP/IP jest podział całego zagadnienia komunikacji sieciowej na szereg współpracujących ze sobą warstw (ang. *layers*). Każda z nich może być tworzona przez programistów zupełnie niezależnie, jeżeli narzucimy pewne protokoły według których wymieniają się one informacjami. Założenia modelu TCP/IP są pod względem organizacji warstw zbliżone do modelu OSI. Jednak ilość warstw jest mniejsza i bardziej odzwierciedla prawdziwą strukturę Internetu. Model TCP/IP składa się z czterech warstw.

⁷ Powstała już nieformalna nazwa anglojęzyczna – CUT: *Coordinated Universal Time*

Warstwa aplikacji czy warstwa procesowa czy (ang. *process layer*) to najwyższy poziom, w którym pracują użyteczne dla użytkownika aplikacje takie jak, np. serwer WWW czy przeglądarka internetowa. Obejmuje ona zestaw gotowych protokołów, które aplikacje wykorzystują do przesyłania różnego typu informacji w sieci.

Warstwa transportowa (ang. *host-to-host layer*) zapewnia pewność przesyłania danych oraz kieruje właściwe informacje do odpowiednich aplikacji. Opiera się to na wykorzystaniu portów określonych dla każdego połączenia. W jednym komputerze może istnieć wiele aplikacji wymieniających dane z tym samym komputerem w sieci i nie nastąpi wymieszanie się przesyłanych przez nie danych. To właśnie ta warstwa nawiązuje i zrywa połączenia między komputerami oraz zapewnia pewność transmisji.

Warstwa Internetu lub warstwa protokołu internetowego (ang. *Internet protocol layer*) to sedno działania Internetu. W tej warstwie przetwarzane są „datagramy” posiadające adresy IP. Ustalana jest odpowiednia droga do docelowego komputera w sieci. Niektóre urządzenia sieciowe posiadają tę warstwę jako najwyższą. Są to routery, które zajmują się kierowaniem ruchu w Internecie, bo znają topologię sieci. Proces odnajdywania przez routery właściwej drogi określa się mianem: trasowanie.

Warstwa fizyczna lub warstwa dostępu do sieci lub (ang. *network access layer*) jest najniższą warstwą i to ona zajmuje się przekazywaniem danych przez fizyczne połączenia między urządzeniami sieciowymi. Najczęściej są to karty sieciowe lub modemy. Dodatkowo warstwa ta jest czasami wyposażona w protokoły do dynamicznego określania adresów IP.

W warstwie transportowej może istnieć wiele protokołów, które współzysują w jednej sieci. Ich algorytmy powinny być optymalizowane, aby zapewnić:

- niezawodny transfer danych,
- dużą przepustowość,
- mały czas reakcji.

Niezawodny transfer danych oznacza zapewnienie, że jeżeli tylko istnieje taka możliwość, każdy komunikat dotrze do odbiorcy. Duża przepustowość określa ilość informacji jaką można przesłać w zadanych czasie. Czas reakcji (ping) musi być jak najmniejszy, co jest niezbędne, aby ograniczyć opóźnienie w komunikacji.

W praktyce nie da się pogodzić wszystkich trzech wymogów względem warstwy transportowej. Zwykle, poprawa jednego z nich dokonuje się kosztem pozostałych. Aby rozwiązać ten problem w sieciach komputerowych zaprojektowano szereg protokołów warstw transportowych. Projektant tworzący aplikację musi w takiej sytuacji wybrać, który protokół najlepiej sprawdzi się w jego oprogramowaniu oraz sieci.

4.5 Architektura klient/serwer

Bardzo popularnym sposobem projektowania aplikacji sieciowych jest model *klient-serwer*, znany choćby ze sposobu manipulowania plikami w systemach UNIX, X Windows czy bazach danych. *Klient* oraz *serwer* to



umowne nazwy pewnych typów programów komputerowych. Pierwszy z nich, z powodu braku danych lub funkcji, musi się kontaktować z drugim. Program zbierający dane lub obsługujący usługi, z którym kontaktuje się klient, nazywany jest serwerem. Rozproszony model klient/serwer, to taki, w którym klient oraz serwer rezydują w różnych komputerach połączonych siecią (niezależnie od systemu operacyjnego).

Serwer gromadzi dane, przetwarza je oraz przechowuje. Klient łączy się z serwerem, przedstawiając swoje zapytania lub żądania wykonania usługi. Serwer analizuje polecenia od klienta, przetwarza dane i wysyła odpowiedź. Ten prosty i sprawdzony model jest zastosowany w wielu aplikacjach dostępnych za pośrednictwem Internetu: serwery WWW, wyszukiwarki informacji w sieci, ogólnodostępne bazy danych czy też serwery poczty elektronicznej.

Wszystkie informacje przechowywane są na serwerze, wobec tego możliwe jest lepsze zabezpieczenie danych. Serwer może „decydować” kto ma prawo do odczytywania i zmiany danych. Istnieje wiele rozwiniętych technologii wspomagających działanie, bezpieczeństwo i użyteczność tego typu rozwiązania.

Duża liczba klientów próbujących otrzymać dane z jednego serwera może powodować różnego typu problemy związane z przepustowością łącza oraz technicznymi możliwościami przetworzenia żądań klientów. W czasie gdy serwer nie działa, dostęp do danych jest całkowicie niemożliwy. Do uruchomienia jednostki będącej serwerem z możliwością obsługi dużej liczby klientów potrzebne jest specjalne oprogramowanie oraz hardware, które nie występują w większości komputerów domowych.

Mimo faktu, że główna idea może być zaimplementowana na wiele sposobów najprostszym przykładem są strony internetowe. Przeglądając strony internetowe komputer użytkownika jest klientem, a komputery które obsługują bazy danych i inne aplikacje potrzebne do obsługi połączenia to serwery. W momencie gdy przeglądarka żąda jakiejś strony, serwer wyszukuje odpowiednie informacje w bazie danych, przetwarza je do postaci strony internetowej, a następnie wysyła do klienta.

Serwerem może być zwykły komputer, jednak do pełnego wykorzystania możliwości, jakie daje oprogramowanie serwerowe powinna to być maszyna dedykowana tej roli. Maszyny takie są przystosowane do pracy ciągłej, wyposaża się je w duże i szybkie dyski twarde, głównie SCSI, dużą ilość pamięci RAM, najczęściej z ECC oraz wydajne procesory serwerowe. Często serwerowe płyty główne mogą obsłużyć 2, 4 lub więcej procesorów. Serwer musi być maszyną niezawodną, w tym celu często zawiera 2 lub więcej (wbudowane) zasilacze typu hot-plug i awaryjne zasilanie, a pomieszczenie, w którym stoi powinno być wyposażone w odpowiednią wentylację lub klimatyzację.

Serwer jest zazwyczaj podłączony do Internetu szybkim łączem i potrafi dzielić pomiędzy aktualnie chcących korzystać z zasobów Internetu użytkowników, których nazywa się klientami. Serwer nie podłączony do Internetu, na przykład w sieci lokalnej może zarządzać współdzieleniem zasobów na poszczególnych komputerach (na przykład zainstalowanymi programami, danymi czy też urządzeniami peryferyjnymi).

Serwery najczęściej pracują pod kontrolą systemów operacyjnych takich jak: FreeBSD, GNU/Linux, Solaris, Novell NetWare, Microsoft Windows Server. Oprogramowanie zainstalowane na komputerze, który pełni rolę serwera, zależne jest od jego funkcji. Wśród wielu usług realizowanych przez serwery w Internecie są między innymi: obsługa stron WWW, poczty elektronicznej, przesyłanie plików (np. FTP), komunikacja online czy strumieniowa transmisja audio i wideo oraz wiele innych. Przykładowo jeśli ma to być serwer WWW, wykorzystuje się często wolne oprogramowanie Apache.

Serwlet to mały program wykonywany po stronie serwera WWW (nazwa powstała na wzór nazwy *applet*). Serwlet otrzymuje od serwera komplet informacji zebranych z interakcyjnych elementów strony (zwykle z pól formularza) i po ich przetworzeniu dostarcza gotową stronę WWW - przesyłaną przez serwer do użytkownika. Ponieważ serwlet jest jedną z klas Javy, można w nim korzystać z całego dobrodziejstwa Java API - w tym z

mechanizmów łączących z bazą danych, zdalnych wywołań metod (RMI) oraz CORBA. Parametry pobrane ze strony można przekazywać (forward) do następnego serwletu, tworząc w ten sposób kaskadę, w której każdy serwlet odpowiedzialny jest za fragment witryny.

Do uruchomienia serwletów konieczne jest funkcjonowanie serwera WWW z zaimplementowanym tzw. kontenerem serwletów. Najpopularniejszym z nich jest Tomcat oraz silniki korzystające z niego jak JBoss czy Apache z dodatkowym modułem (Apache Tomcat).

W ogólniejszym znaczeniu serwlet to program działający w modelu żądanie-odpowieź. Poprzez analogię do protokołu HTTP, działającego w takim właśnie modelu, konstruować można inne typy serwletów. Przykładem technologii wykorzystującej koncepcję serwletu jest SIP Servlet API.

4.5.1 Przykład aplikacji sieciowej typu klient/serwer - rozproszony system plików.

Próba otwarcia pliku przez klienta jest równoważna wysłaniu serwerowi wiadomości z typem operacji, nazwą pliku, identyfikatorem użytkownika i szeregiem innych danych. Serwer otrzymujący taką wiadomość przeprowadza niezbędne testy - na przykład sprawdza, czy podany plik istnieje, czy użytkownik ma do niego prawo dostępu itp. W przypadku, gdy wszystkie testy wypadną pomyślnie, serwer wykonuje żadaną operację i wysyła odpowiedź klientowi.

Na etapie projektowania systemu podejmowana jest decyzja, który element modelu (klient, czy serwer) ma zapamiętać wszystkie dane dotyczące stanu czytanego pliku (nazwę pliku, przez kogo został otwarty, aktualne pozycje wskaźników do otwartych plików) oraz dane mogące się zmieniać w wyniku operacji wykonywanych przez klienta. W przypadku, gdy informacje te przechowywane są przez serwer, mamy do czynienia z modelem serwera „ze stanem” (z każdym z klientów mogą być związane struktury danych zawierające informacje na temat poprzednich poleceń), jeżeli przez klienta - jest to model serwera „bez stanu”. Wybór modelu może zależeć od wielu czynników.

Inna charakterystyka modelu klient/serwer obejmuje sposób obsługi poleceń przesyłanych przez klienta: serwery mogą być *wielowątkowe* (wewnętrznie współbieżne) lub też mogą być napisane bez użycia *wątków*. Serwery wielowątkowe są w stanie jednocześnie przetwarzać pewną liczbę żądań i związanych z nimi wątków, czyli tworzyć odpowiednik „lekkiego” procesu. Serwery napisane bez użycia wątków muszą skończyć wykonywanie bieżącego zadania zanim przejdą do obsługi następnego lub też manipulują wieloma wykonywanymi zadaniami za pomocą skomplikowanych struktur danych. Większość popularnych serwerów jest wielowątkowa. Wynika to głównie z postępów poczynionych w ciągu ostatnich lat w projektowaniu oraz implementacji bibliotek wątkowych.

Działanie programów prowadzących interakcje w sieci nie zawsze w pełni odpowiada definicji modelu klient/serwer. Na przykład, wybrany serwer może żądać wykonania usługi od jakiegoś innego serwera stając się jego klientem. Istotnym wymaganiem przy projektowaniu usług świadczonych przez poszczególne elementy modelu jest zasada, aby nie doprowadzać do zamknięcia kręgu zależności pomiędzy elementami modelu (klientami i serwerami).

4.6 Architektura klient/serwer w systemie pomiarowym

Podstawowym kreatorem *rozproszonego systemu pomiarowo-kontrolnego* jest użytkownik (klient), używający swego komputera w roli wirtualnego panelu stworzonego do obsługi wszystkich zasobów pomiarowo-kontrolnych oraz serwery zbierające rozproszone dane pomiarowe. Aplikacja klienta odpowiedzialna jest za wysyłanie poprzez sieć poleceń użytkownika do serwerów, których zadaniem jest sterowanie urządzeniami i uruchamianie określonych procesów pomiarowych. Każdy serwer przetwarza odebrane polecenia (zapytania)

w celu uaktywnienia stosownego przyrządu i uzyskania żądanej przez klienta odpowiedzi (danych pomiarowych). Następnie wysyła odpowiedź do klienta, który odpowiednio przetwarza i wyświetla wynik.

Wymieniona struktura podstawowa ma jednak kilka istotnych wad:

- Użytkownik musi znać wcześniej adresy serwerów, z którymi chce się komunikować.
- Dane (wyniki) przychodzące z różnych serwerów nie są zsynchronizowane w czasie z powodu opóźnień wprowadzanych przez sieć (niemożliwych do przewidzenia).
- W sieci z liczbą klientów większą niż jeden, powstają problemy z podziałem zasobów (plików z danymi, urządzeń pomiarowych itp.).
- Tylko jeden użytkownik ma w danej chwili dostęp do wybranego urządzenia, jednak opóźnienie w sieci może łatwo doprowadzić do sytuacji, gdy kilka poleceń do urządzenia nadejdzie w tym samym czasie od różnych użytkowników, wprowadzając zamęt.

Znane są różne sposoby obejścia wyżej wymienionych problemów. Rozwiązaniem pierwszego problemu może być zastosowanie „listy urządzeń” jako zbioru przyrządów podłączonych do różnych serwerów. Po zdefiniowaniu „listy urządzeń” użytkownik może wysłać żądanie do dowolnego z dostępnych serwerów, który następnie łączy się z innym serwerem przekazując polecenie dalej. Właściwość tę (*auto-routing*) uzyskuje się dzięki koncepcji tzw. „zaufanych serwerów”. Kiedy żądanie dojdzie do odpowiedniego serwera, przetwarza je i wysyła odpowiedź bezpośrednio do klienta.

Problem synchronizacji można rozwiązać przy użyciu protokołu *NTP* (Network Time Protocol). Polega to na ustawieniu takiego samego czasu na wszystkich serwerach należących do tej samej domeny. Natomiast każde polecenie użytkownika powinno zawierać znacznik czasu, czyli informację o chwili wykonania pomiaru. Po wysłaniu poleceń do kilku serwerów, wszystkie wykonują pomiar w określonym czasie i wysyłają odpowiedź do klienta. Dokładność synchronizacji zależy od jakości *NTP* oraz opóźnień systemu operacyjnego i nie przekracza ułamka sekundy.

Problem kolejności dostępu do zasobów można rozwiązać przez zastosowanie *priorytetów*. Polecenia, które nie muszą być wykonane w określonym czasie są kolejgowane, a następnie przetwarzane w kolejności nadeśłania, natomiast polecenia ze znacznikiem czasu (o wyższym priorytecie) są przesyłane do oddzielnej kolejki.

4.7 Architektura peer-to-peer (P2P)

P2P (od ang. *peer-to-peer* - równy z równym) to model komunikacji w sieci komputerowej, który gwarantuje obydwu stronom równorzędne prawa (w przeciwieństwie do modelu klient-serwer).

W sieciach P2P, każdy komputer może jednocześnie pełnić zarówno funkcję klienta, jak i serwera. W najpopularniejszej implementacji modelu P2P, jaką są programy do wymiany plików w Internecie każdy węzeł sieci (czyli komputer użytkownika) odgrywa rolę serwera przyjmując połączenia od innych użytkowników danej sieci, jak i klienta, łącząc się i pobierając dane z innych maszyn działających w tej samej sieci. Wymiana danych jest zawsze prowadzona bez pośrednictwa centralnego serwera. Sieć P2P charakteryzuje się także płynną strukturą, która zmienia się w zależności od tego, jakie komputery są w niej aktualnie zalogowane.

Należy pamiętać, że korzystanie z sieci p2p nie jest "darmowe". Kiedy korzystasz z sieci p2p "koszty", które poniesiesz ponieść to wysłanie (upload) minimum takiej samej ilości danych (choć nie koniecznie tych samych) co pobrałeś (download), w innym przypadku sieć szybko załamie się. Aby pokryć ilość danych "ukradzioną" z sieci przez nieodpowiedzialnych użytkowników, a także komputery należące do organizacji chroniących prawa autorskie, należy wysłać ilość danych równą 110%-120% pobranych danych. To jest właśnie koszt "darmowych" plików z Internetu.

4.8 Java jako programowa platforma dla systemu rozproszonego

Szybki rozwój technik internetowych i zastosowań technologii sieciowych powoduje rosnące zainteresowanie *rozproszonymi systemami pomiarowymi*. Problemy w zakresie projektowania rozproszonych systemów pomiarowych wiążą się głównie ze stosowanymi technologiami sieciowymi i złożonością ich oprogramowania. W aspekcie technologii sieciowych, wymagane jest zastosowanie architektury otwartej współpracującej z wieloma różnymi wirtualnymi przyrządami pomiarowymi czy podsystemami, gwarantującej deterministyczny charakter transmisji. Natomiast w aspekcie oprogramowania, głównym problemem jest to, że konwencjonalne techniki tworzenia programów zawodzą w przypadku rozproszonych i złożonych systemów. Dzieje się tak za przyczyną długiego czasu tworzenia aplikacji sieciowych i wysokich kosztów z tym związanych.

Z pomocą przychodzi *programowanie zorientowane obiektowo*. Technika ta ułatwia tworzenie oprogramowania systemowego poprzez zbiór współpracujących ze sobą, niezależnych elementów - *obiektów*. Obiekty mogą być rzeczywistymi bądź też wirtualnymi przyrządami, sterownikami urządzeń, serwerami, klientami, lub wreszcie abstrakcyjnymi obiektami jak funkcje matematyczne i logiczne.

Językiem umożliwiającym sieciowe programowanie obiektowe jest *Java*, opracowana przez informatyków z *Sun Microsystems*. Język ten ułatwia modułarne programowanie oraz pozwala na wielokrotne wykorzystywanie kodu, w sposób efektywny i nieskomplikowany. Zapewnia on ścisłą kontrolę typów, ma automatyczne zarządzanie pamięcią, wbudowane wątki oraz wygodne w użyciu biblioteki klas programowania sieciowego. Programy kompilowane są do formatu pośredniego, niezależnego od rodzaju komputera i systemu operacyjnego (platformy informatycznej). Pozwala to na wykonywanie skompilowanych programów w dowolnej architekturze sprzętowej, dla której jest dostępny interpreter Javy, a więc również na dowolnych komputerach osobistych i stacjach roboczych.

Związana z Javą technologia *apletów* ożywiła strony Internetowe. *Aplety* to wykonywalny kod Javy, ładowany wraz ze stronami HTML. Architektura Javy zapewnia to, że *aplety* nie mogą wykonywać niepożądanych, ze względu na bezpieczeństwo, operacji na komputerze użytkownika (np. otwierania plików). Aplety Javy doskonale nadają się do wzbogacania podręczników elektronicznych przygotowywanych w formacie HTML.

Szybko rośnie liczba aplikacji napisanych w Javie. Pojawiło się sporo związanych z nią technologii, takich jak rozproszone obiekty Javy, dostęp do baz danych przez Javę, klasy do kodowania danych czy wreszcie JavaOS (system operacyjny napisany w Javie).

4.8.1 Java a programowanie klient-serwer

Podczas tworzenia aplikacji typu klient-serwer w Javie z reguły wykorzystuje się połączenia sieciowe przez tzw. *gniazdka* (ang. *sockets*) za pomocą standardowej biblioteki `java.net`. Posługiwanie się umieszczonymi tam klasami wymaga obsługi wyjątków Javy. Wygodniejszy dostęp do gniazdka zapewniają standardowe klasy strumieni wejścia i wyjścia (*InputStream/OutputStream*). Ponadto, dość często wykorzystywane są wątki - po stronie klienta do zapewnienia asynchronicznego odbioru danych z serwera, a po stronie serwera do jednoczesnej obsługi zadań pochodzących od wielu klientów.

Jak już wspomniano wcześniej, gniazdka to dostępna na wielu systemach unixowych oraz pod Windows metoda komunikacji sieciowej między programami. Gniazdko należy traktować jako kanał komunikacyjny, gdzie jeden program wpisuje do niego dane w postaci strumienia bajtów, a drugi je odbiera. W celu nawiązania połączenia, jeden z programów (serwer) musi otworzyć gniazdko i oczekiwać na połączenie. Z otwartym gniazdkiem musi być związany pewien numer, zwany *portem*. Program łączący się z otwartym gniazdkiem (klient) musi znać sieciowy adres komputera (IP) oraz numer portu. Wiele serwisów, np. telnet, ftp, gopher czy www, ma

przydzielone na sztywno i ogólnie znane porty. Użytkownicy niestandardowych programów muszą sami zadbać o to, żeby numer portu serwera był znany klientom. Program, który otworzył gniazdko może, ale nie musi zaakceptować połączenia.

Gniazdko są łączem dwustronnym - oznacza to, że do otwartego gniazdko dane można zarówno wpisać, jak i odczytać. Tak, więc dla dwustronnej komunikacji między programami wystarczy otwarcie tylko jednego gniazdko. W modelu klient/serwer implementowanym w Javie typowe jest używanie dwóch rodzajów gniazdek, pochodzących ze standardowej biblioteki `java.net:Socket` oraz `ServerSocket`. Klasa `Socket` wykorzystywana jest przez klienta - wystarczy podanie adresu internetowego oraz portu, aby połączyć się z działającym serwerem. Obsługa połączeń jest nieco bardziej skomplikowana od strony serwera, który używa klasy `ServerSocket`. `ServerSocket` czeka na nadchodzące połączenia od klientów. W chwili odebrania żądania połączenia `ServerSocket` tworzy nowy obiekt typu `Socket`, dzięki któremu serwer może porozumiewać się z klientem. W tej sytuacji zarówno klient, jak i serwer mają dostęp do własnych, lokalnych kopii obiektu `Socket`, dzięki którym mogą się porozumiewać ze sobą. Dodatkowo, `ServerSocket` może akceptować dalsze połączenia od klientów, co oznacza, że wielu klientów może jednocześnie mieć otwarte połączenia z tym samym serwerem.

Możliwość jednoczesnej obsługi wielu klientów oznacza, że gdy np. żądanie jednego z nich wymaga czasochłonnego dostępu do dysku, serwer może przydzielić czas procesora innemu klientowi. W przeciwnym wypadku, gdy obsługiwany jest tylko jeden klient, serwer czeka na zakończenie dostępu do dysku i marnuje czas procesora (lub wielu procesorów - gdy serwer działa na wieloprocessorowym komputerze).

4.8.2 Wątki w Javie

Popularnym narzędziem Javy jest *wątek*. Wątek można traktować jako część programu wykonywaną współbieżnie w obrębie danego procesu. Jeden proces może zawierać wiele wątków, mających dostęp do wspólnych danych. Wiele implementacji wątków współdziała z operacjami wejścia/wyjścia, dzięki czemu gdy jeden wątek zablokowany jest na przykład operacją czytania z konsoli lub z dysku, inny jest automatycznie uruchamiany.

Możliwe jest kontrolowanie porządku wykonywania wątków przez związanie z nimi priorytetów. Wykonywanie wątków może być przeplatane w czasie, podobnie jak wykonywanie procesów w systemie operacyjnym, z podziałem czasu. Twórcy Javy bardzo starannie zaprojektowali wątki - są one łatwe i wygodne w użyciu. W serwerze umożliwiają jednoczesną obsługę wielu klientów (odbieranie wielu połączeń i przypisywanie każdemu połączeniu osobnego wątku). Natomiast w przypadku klienta jeden wątek będzie czytał dane pochodzące od serwera, a inny np. wyświetlał nadchodzące dane czasu rzeczywistego.

4.8.3 Wyjątki w Javie

Podczas wykonywania programu może zajść wiele niespodziewanych sytuacji. Niektóre wydarzenia można przewidzieć - na przykład to, że komputer z serwerem przestanie działać albo, że niedostępne będzie połączenie między dwoma komputerami przez sieć. Są też inne sytuacje, uniemożliwiające połączenie się z serwerem klasyfikowane jako nieprzewidywalne - na przykład błąd w kodzie protokołów sieciowych, objawiający się w dość przypadkowy sposób okresową niemożliwością wysyłania pakietów.

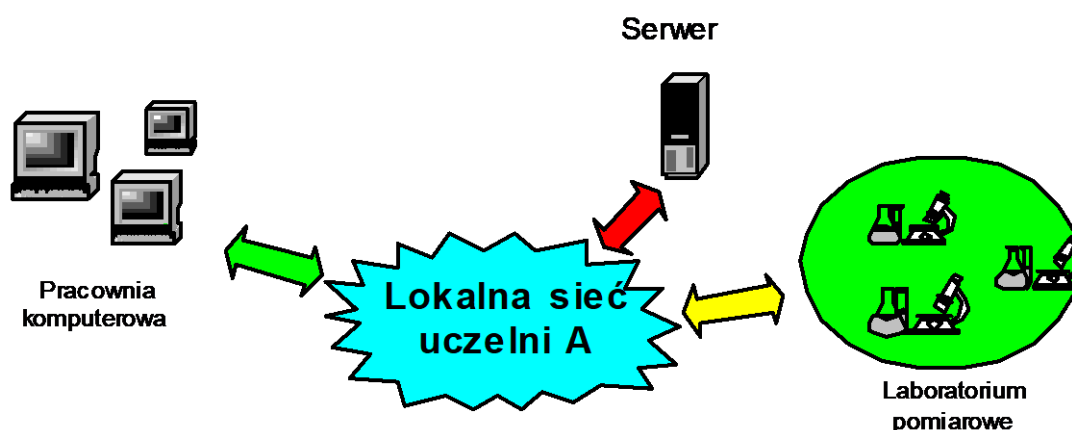
Java zapewnia mechanizm obsługi sytuacji wyjątkowych, który wykorzystuje się do uodpornienia kodu klienta na niepożądane zdarzenia. Mechanizm ten działa na takiej zasadzie, że sytuacje wyjątkowe komunikowane są przez stworzenie obiektu z informacją o wyjątku. Procedury, w których mogą powstać sytuacje wyjątkowe (jak

na przykład próba odczytu nieistniejącego pliku czy próba połączenia przez sieć z nieczynnym przyrządem wirtualnym), muszą deklarować rodzaje wyjątków jakie mogą wystąpić.

5. Wirtualne laboratorium

5.1. Wstęp

Wirtualne przyrządy pomiarowe, w dużym stopniu zastąpiły w laboratoriach dydaktycznych drogie i skomplikowane przyrządy inteligentne, co znacznie uprościło proces projektowania, uruchamiania i modernizacji tych laboratoriów. Graficzny interfejs użytkownika (GUI), który do złudzenia przypomina rzeczywisty przyrząd pomiarowy, powoduje, że użycie i rozumienie przyrządu jest intuicyjne dla tych, którzy korzystali do tej pory z konwencjonalnych przyrządów pomiarowych. Możliwość modyfikowania procedury pomiarowej poprzez zmianę zainstalowanego w komputerze oprogramowania, bez zmiany komponentów sprzętowych sprawia, że badania i eksperymenty stają się coraz bardziej elastyczne, nowoczesne i proste. Sprzęgnięcie przyrządu wirtualnego z lokalną siecią komputerową jest zadaniem niesłychanie prostym. Wymaga zainstalowania karty sieciowej w komputerze i przydzielenia mu numeru IP. Obsługa protokołu sieciowego, natomiast wbudowana jest do większości bibliotek oprogramowania narzędziowego. Przykładowa architektura laboratorium wirtualnego zbudowanego na bazie lokalnej sieci komputerowej przedstawiona jest na rysunku 5.1.



Rys.5.1 Architektura systemu rozproszonego zbudowanego na bazie lokalnej sieci komputerowej

Wśród dostępnych modeli laboratorium wirtualnego, w sensie zdalnego dostępu do laboratorium sprzętowego, można wyróżnić następujące poziomy aktywności użytkownika⁸:

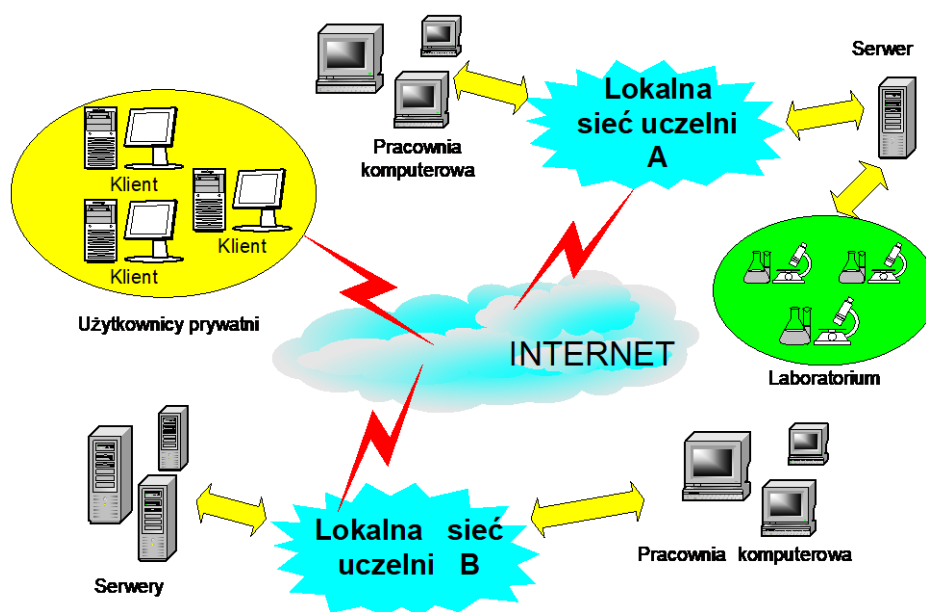
- POZIOM 0. Obserwacja (bierna) pracy nauczyciela (laboranta) wykonującego eksperyment (prezentacja video w czasie nierzeczywistym),
- POZIOM 1. Zdalna wizualizacja przebiegu eksperymentu w czasie rzeczywistym - obserwacja działania automatycznego systemu, odbiór wyników pomiaru,
- POZIOM 2. Zdalna kontrola przebiegu eksperymentu - zdalne sterowanie przebiegiem eksperymentu, obserwacja działania systemu, odbiór wyników pomiaru,
- POZIOM 3. Zdalne projektowanie przebiegu eksperymentu - projektowanie i uruchamianie własnego systemu (ćwiczenia laboratoryjnego) oraz jego pełna implementacja w laboratorium wirtualnym.

⁸ Termin „wirtualne laboratorium” jest bardzo pojemny. Oprócz symulacji eksperymentu obejmuje również „zdalny dostęp do laboratorium rzeczywistego”.

5.2. Idea wirtualnego laboratorium

Osadzenie wirtualnego przyrządu pomiarowego w rozproszonym systemie zlokalizowanym w sieci Internet daje niespotykaną do tej pory możliwość tworzenia zaawansowanych i elastycznych systemów, które mogą służyć prowadzeniu badań naukowych jak i wspomagać proces dydaktyki. Szybki rozwój narzędzi programistycznych ułatwiających komunikację komputerów na duże odległości przesądza o wyjątkowej atrakcyjności wirtualnych laboratoriów [34].

Możliwe staje się prowadzenie eksperymentów, oferowanych przez różne ośrodki naukowe, oraz korzystanie z wyników przez szerokie grono naukowców i studentów, niezależnie od miejsca ich aktualnego pobytu. Sprawia to, że wirtualne laboratoria pomiarowe są w ostatnich latach przedmiotem badań wielu instytucji naukowych. Przykładowy schemat współpracy międzyuczelnianej w zakresie laboratoriów wirtualnych przedstawiono na rys. 5.2.



Rys.5.2 Schemat współpracy międzyuczelnianej w zakresie laboratoriów wirtualnych

Ośrodek badawczy, który udostępnia laboratorium (uczelnia A), w sieci lokalnej stawia do dyspozycji swoich pracowników naukowych i studentów wszystkie bądź też wybrane, zasoby (przyrządy wirtualne). Po sprzężeniu sieci lokalnej z Internetem z zasobów laboratorium korzystać może również uczelnia B. Korzyści z takiego stanu rzeczy czerpią obydwa ośrodki, ponieważ uczelnia B może udostępniać uczelni A wolną moc obliczeniową. W przypadku, gdyby uczelnia B, na zasadzie specjalizacji w innej dziedzinie, udostępniła swoje laboratorium wirtualne, obydwie uczelnie byłyby w stanie uzupełniać się w szerszej skali. Możliwość korzystania z zasobów innych ośrodków badawczych jest szczególnie pożądana, tam gdzie chodzi o kosztowny, wysoko specjalistyczny sprzęt, który często jest poza zasięgiem wielu mniejszych uczelni i instytutów. W zasadzie, za pośrednictwem Internetu, z zasobów wirtualnego laboratorium, mogą korzystać też inni upoważnieni do tego użytkownicy, znajdujący się w swoich domach, szkołach, miejscach pracy itp.. Upowszechnienie tego typu technologii ułatwi dostęp do wiedzy i umożliwi bardziej efektywne wykorzystanie potencjału ośrodków naukowo-badawczych.

Eksperymenty wykonywane w trybie „on-line”, z możliwością bezpośredniego wpływania na badany proces poprzez dobór warunków pomiaru, źródeł i parametrów wymuszeń, sprawiają, że student będzie nie tylko biernym obserwatorem, ale świadomym uczestnikiem prowadzonych doświadczeń.

Dostęp do wirtualnego laboratorium nie musi być ograniczony do kilku godzin w tygodniu, lecz możliwy nawet przez 24 godziny na dobę. Jak już wcześniej wspomniano, jest to nieocenione zwłaszcza w przypadku drogiego,

unikatowego sprzętu, do którego studenci mają ograniczony dostęp w laboratorium tradycyjnym. Studenci z każdego poziomu studiów mogliby korzystać zarówno z kosztownego sprzętu, jak i drogich obiektów badań, rozwijać indywidualne zainteresowania. Organizacja tego dostępu, dokonywanie autoryzacji oraz identyfikacji użytkowników stanowi odrębny problem, który jest już rozwiązany na pewnym poziomie w innych grupach zastosowań informatycznych.

Celem niniejszego rozdziału jest podanie opisu szkieletu systemu wirtualnego laboratorium opracowywanego w Zakładzie Systemów Informacyjno-Pomiarowych, Wydziału Elektrycznego PW. Architektura systemu (niezwykle oszczędna) skonfigurowana jest z użyciem pojedynczego centralnego serwera. Zasadniczymi aplikacjami wchodzącymi w skład projektu są serwer i klient. Zadaniem serwera jest udostępnianie użytkownikom zasobów laboratorium (urządzeń i systemów pomiarowych), nadzorowanie przebiegającej komunikacji, kontrola dostępu do zasobów i organizacja użytkowników. Klient natomiast jest interfejsem użytkownika, zawierającym pulpit z przyrządami pomiarowymi, mającym umożliwić wykonanie badań na wybranych obiektach.

5.3. Założenia ogólne

Podstawowym zadaniem oprogramowania tak skonfigurowanego systemu jest umożliwienie komunikacji między wirtualnym laboratorium, a jego użytkownikami, zapewnienie dostępu do urządzeń pomiarowych i innych zasobów systemu oraz nadzorowanie przebiegającej komunikacji, czyli zarządzanie udostępnianymi zasobami oraz organizacja i kontrola użytkowników.

Zadania organizacyjne i kontrolne polegają na tworzeniu grup użytkowników i określaniu warunków, na jakich może odbywać się korzystanie przez nich z zasobów, na przydzielaniu grupom praw do określonych urządzeń i ich funkcji oraz ograniczaniu bądź rozszerzaniu nadanych uprawnień.

Użytkownicy podlegają autentykacji i autoryzacji. Autentykacja polega na sprawdzeniu (np. na podstawie hasła), czy dany użytkownik lub system jest tym, za kogo się podaje. Po uwierzytelnieniu następuje autoryzacja, czyli proces przydzielania dostępu do zasobów według określonych wcześniej uprawnień i priorytetów.

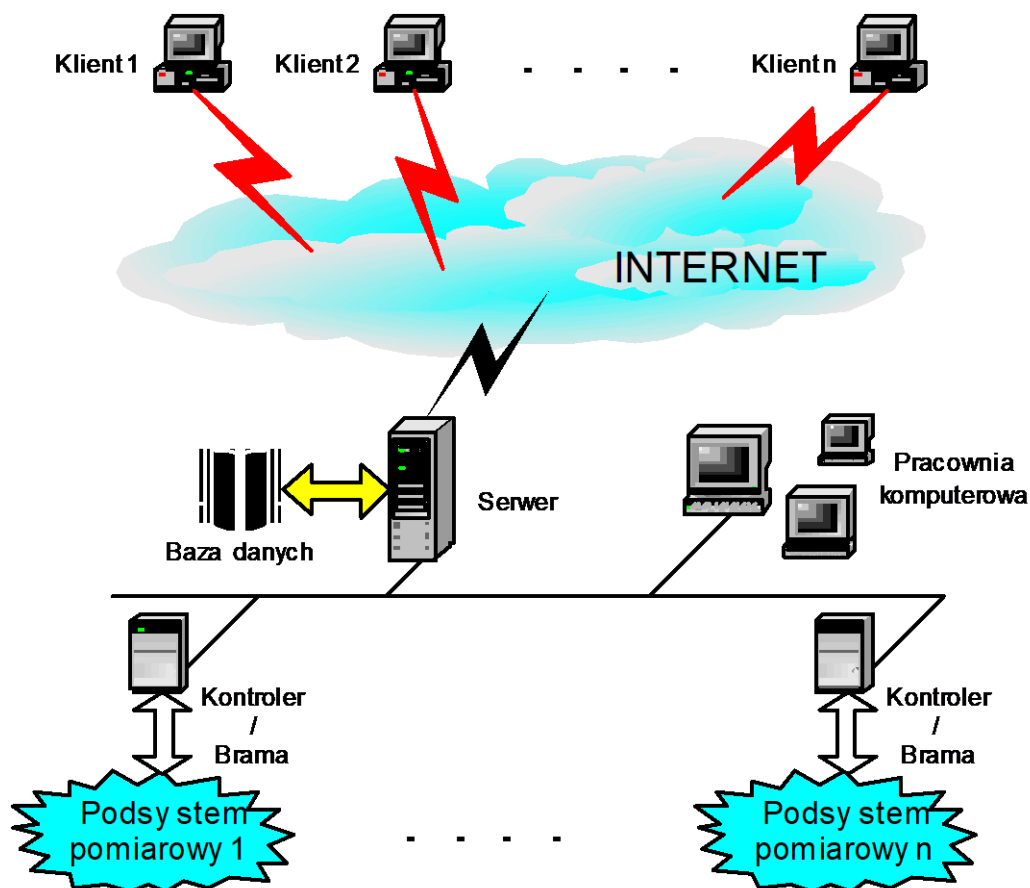
Oprogramowanie serwera umożliwia dokonywanie pomiarów na żądanie i pomiarów, odbywających się w czasie rzeczywistym. Pomiar na żądanie odbywa się na zasadzie: pytanie - odpowiedź. Użytkownik określa parametry pomiaru, a następnie wysyła żądanie jego wykonania i odesłania otrzymanego wyniku. Podczas pomiaru w czasie rzeczywistym użytkownik wchodzi w interakcję z wirtualnym przyrządem, jak z realnym. W czasie pomiaru ma możliwość na bieżąco zmieniać dowolne parametry i obserwować występujące zmiany.

Oprogramowanie musi składać się z dwóch zasadniczych aplikacji: aplikacji serwera i aplikacji klienta oraz ewentualnie, pomocniczej aplikacji, umożliwiającej testowanie programu. Oprogramowanie klienta uzewnętrznia panel z wirtualnymi przyrządami pomiarowymi. Klient może podłączyć się do serwera, stanowiącego bramę, udostępniającą mu część sprzętową zasobów wirtualnego laboratorium. Po zalogowaniu się, klient tworzy sesję, która umożliwia mu sterowanie urządzeniami oraz odbiór danych. Program powinien być wyposażony w „kreatory”, dzięki czemu obsługa jest dla użytkownika bardzo prosta, niemal intuicyjna.

Funkcją serwera jest kontrola uprawnień, zapewnienie bezpieczeństwa i przekazywanie danych od urządzeń do klientów (np. instrukcji sterujących, komunikatów o błędach itp.) oraz w odwrotnym kierunku (np. danych dotyczących parametrów pomiarowych). Serwer umożliwia pełną współbieżną pracę wielu użytkownikom na wielu urządzeniach, czyli wielodostęp.

5.4. Architektura wirtualnego laboratorium

Ogólną zasadę działania (architekturę) takiego laboratorium zaprezentowano na rys 5.3.



Rys.5.3 Architektura wirtualnego laboratorium

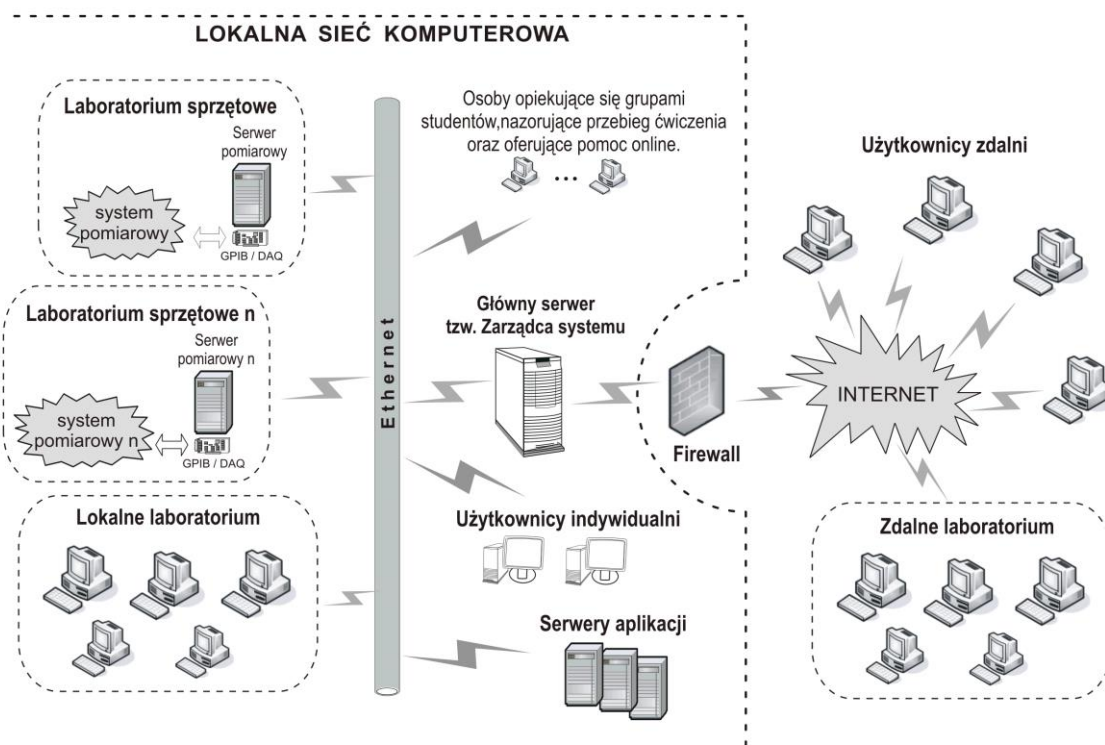
Przedstawia on zbiór podsystemów pomiarowych (przrządów wirtualnych) zarządzanych przez komputery ogólnego przeznaczenia (osobiste). Każdy z nich oprócz funkcji kontrolera systemu pomiarowego spełnia również rolę bramy, przekazującej dane pomiarowe do serwera, który jest centralną częścią laboratorium. Serwer dokonuje dystrybucji danych poprzez Internet do podłączonych do niego klientów, które z kolei za pośrednictwem serwera sterują urządzeniami pomiarowymi. Klienci laboratorium mogą znajdować się nie tylko w Internecie, lecz również, co pokazuje prezentowany rysunek, w sieci lokalnej uczelni.

W systemie, zbudowanym na pojedynczym centralnym serwerze, możliwe jest zarządzanie dostępem (nadawanie i zmiana uprawnień użytkowników, wielodostęp). Klienci mają ułatwione zadanie dotarcia do wybranych urządzeń, znajdujących się w sieci. Wystarczy znajomość adresu IP serwera, aby uzyskać połączenie z przyrządem czy systemem. Istnienie centralnego serwera wymusza unifikację klientów, konieczne jest spełnianie przez nich parametrów zgodnych z określonym standardem.

Serwer stanowiący jednostkę centralną, umożliwia uproszczenie konstrukcji oprogramowania poszczególnych podsystemów pomiarowych. Przeniesienie zadania kontroli dostępu i praw użytkowników na serwer sprawia, że oprogramowanie wirtualnych przyrządów i systemów pomiarowych musi jedynie kontrolować sam proces pomiaru. Głównym jego zadaniem jest zbieranie danych i przesyłanie ich do serwera. Oprogramowanie urządzeń nie realizuje też czasochłonnej i złożonej dystrybucji danych, dzięki czemu nie wymaga bardzo wydajnego komputera. Ograniczenie funkcji spełnianych przez oprogramowanie systemów pomiarowych pozwala zyskać czas na wykonywanie takich dodatkowych zadań, jak np. kodowanie sygnałów, kompresja

sygnałów. Zasadniczą wadą centralnego serwera jest wymaganie dużej wydajności, szczególnie przy obsłudze wielu klientów. Należy jednak pamiętać również o tym, że znaczącą rolę odgrywa tutaj konfiguracja sieci lokalnej, w której znajdują się systemy pomiarowe.

Uogólniony schemat architektury wirtualnego laboratorium dydaktycznego zamieszczono na rys.5.4.



Rys.5.4 Uogólniony schemat architektury wirtualnego laboratorium dydaktycznego

5.5. Organizacja użytkowników i urządzeń w systemie

Zarówno użytkownicy, jak i zasoby systemu zorganizowane są w grupy, tak jak dzieje się to w prawdziwym laboratorium studenckim lub w większości sieciowych systemów operacyjnych. Każde urządzenie wchodzi w skład jednej grupy, która stanowi odpowiednik zbioru przyrządów niezbędnych do przebadania danego obiektu. Użytkownicy natomiast mogą należeć do wielu grup. Każda z tych grup może mieć odmienne prawa dostępu do różnych grup urządzeń. Uproszczony model zarządzania oferuje wyłącznie prawa do odczytu danych pomiarowych i prawa do sterowania urządzeniami wchodzącymi w skład systemu.

Do przechowywania informacji o użytkownikach, ich grupach, grupach urządzeń oraz przydzielonych prawach, może być wykorzystana relacyjna baza danych, która gwarantuje nie tylko wygodny sposób składowania danych, ale również dzięki relacjom zapewnia ich integralność.

Architektura bazy danych zakłada istnienie pięciu tabel. Tabela „Użytkownicy” zawiera informacje o użytkownikach, tzn. unikalną nazwę, będącą kluczem pierwotnym, hasło, imię i nazwisko oraz opis. W tabeli „Grupy_użytkowników” znajdują się grupy użytkowników, z których każda ma jednoznacznie przydzieloną nazwę. Ponieważ każdy użytkownik może należeć do wielu grup, a każda grupa może zawierać wielu użytkowników, między tymi tabelami występuje relacja „wiele do wielu”. Taką relację buduje się za pomocą struktury pośredniej, którą stanowi tu „Członkostwo”. Grupy urządzeń przechowywane są w tabeli „Grupy_przyrządów”. Prawa grup użytkowników do grup przyrządów zapisywane są w tabeli „Pozwolenia”, która wiąże relacjami „Grupy_użytkowników” i „Grupy_przyrządów”. Komunikacja z bazą danych odbywa się poprzez interfejs JDB w języku SQL. Systemem bazy danych może być Oracle, DB2,

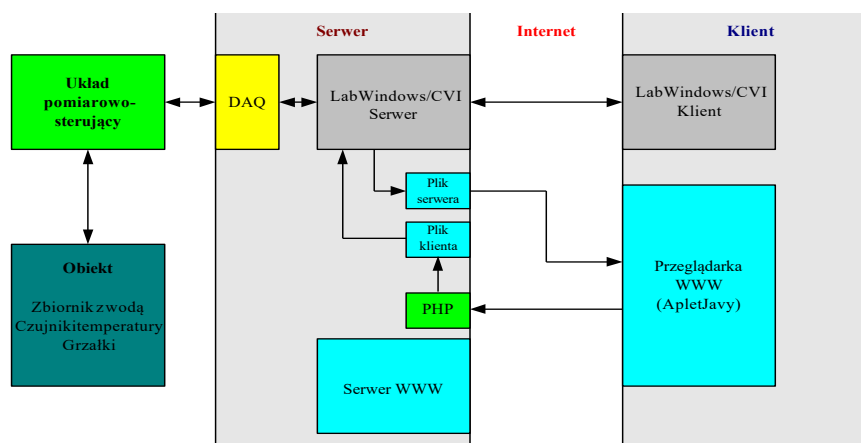
Interbase, Postgres lub inne, zawierające sterownik JDBC oraz gwarantujące odpowiednią zgodność ze specyfikacją SQL.

Wirtualne laboratorium może być rozwiązywane na wiele różnych sposobów, a jego architektura i organizacja są w dużym stopniu zależne od inwencji twórczej i możliwości finansowych projektantów. Nie można oczywiście zapominać o pewnych ramach narzucanych przez dostępne sieciowe protokoły komunikacyjne oraz różne modele oprogramowania komercyjnego. Na szczęście, dzięki uwzględnieniu w procesie projektowania specyficznych właściwości sieciowego systemu rozproszonego, takie laboratorium daje się modyfikować, unowocześniać i rozbudowywać.

5.6. Przykład zdalnego dostępu do laboratorium

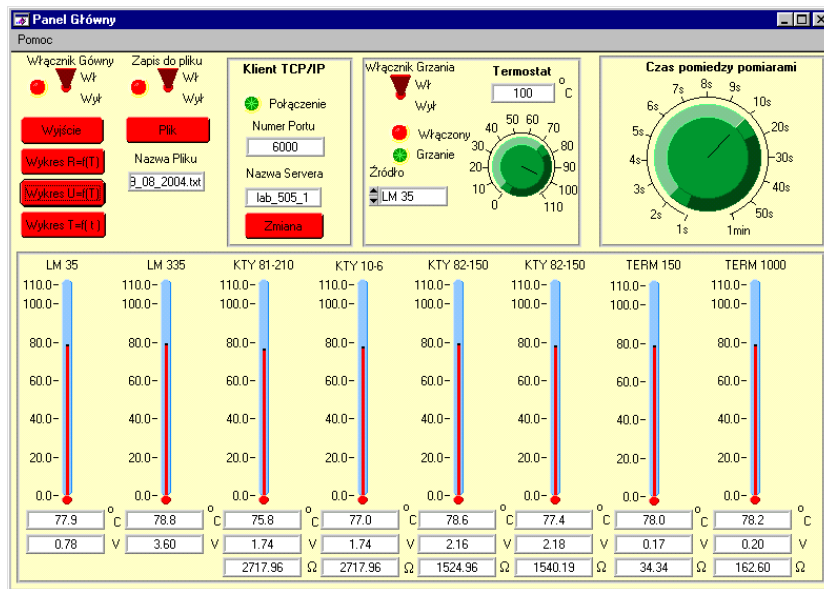
Jako przykład realizacji praktycznej zdalnego dostępu do laboratorium niech posłuży wielopunktowy system do pomiaru i regulacji temperatury wykonany jako praca dyplomowa w Zakładzie Systemów Informacyjno-Pomiarowych. Schemat funkcjonalny tego systemu przedstawiono na rys.5.5.

Część sprzętowa systemu to elektroniczny układ pomiarowo-sterujący współpracujący z wielokanałową kartą zbierania danych (DAQ), umieszczoną bezpośrednio w serwerze pomiarowym. Zadaniem układu pomiarowo-sterującego jest dopasowanie sygnałów wyjściowych z czujników temperatury (osiem czujników różnego typu) do wejść karty zbierania danych oraz umożliwienie sterowania grzałkami.



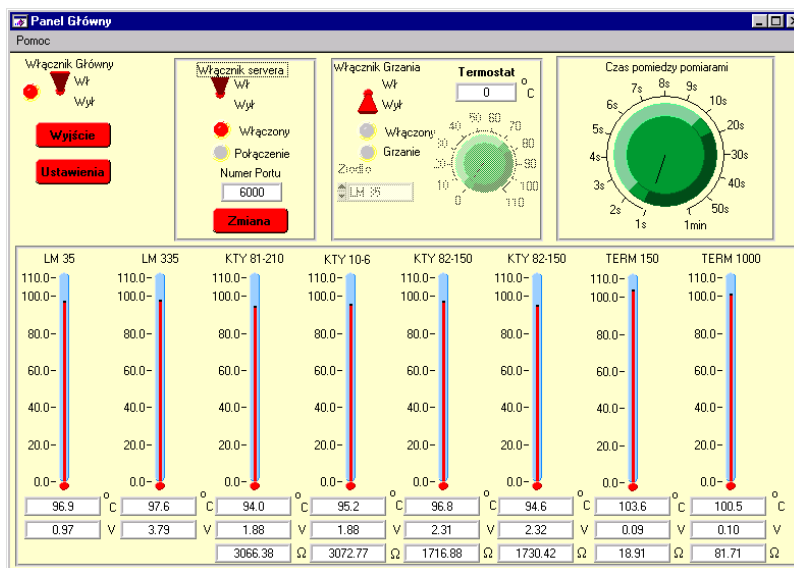
Rys. 5.5 Schemat funkcjonalny systemu do wielopunktowego pomiaru i regulacji temperatury

Pierwszy moduł programu serwera pomiarowego został przygotowany w środowisku LabWindows/CVI (rys. 5.6). Drugi zaś przeznaczony jest do obsługi przesyłania danych z wykorzystaniem przeglądarki WWW.



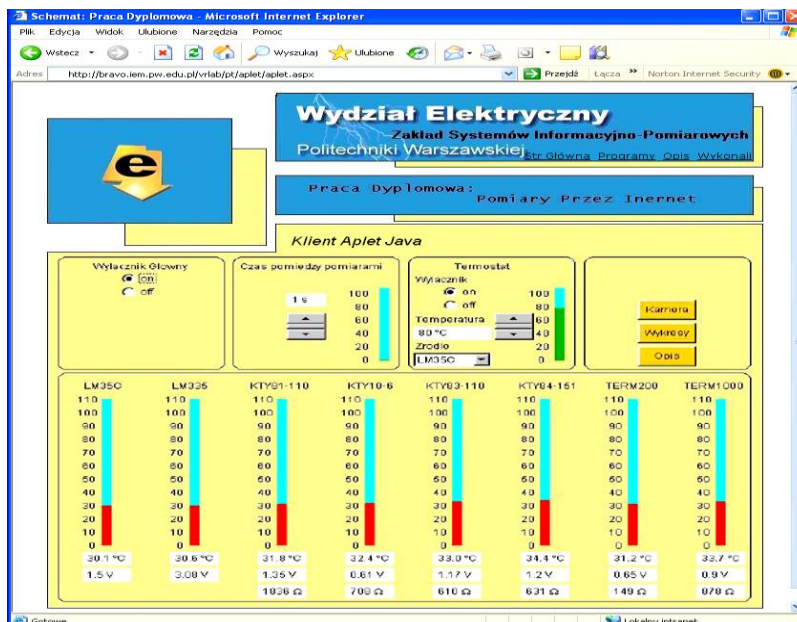
Rys.5.6 Aplikacja serwera (LabWindows/CVI)

Program klienta został również napisany w dwu wersjach. W wersji pierwszej z wykorzystaniem pakietu LabWindows/CVI oraz protokołu DataSocket (rys. 5.7). W wersji drugiej z wykorzystaniem języka Java i języka skryptowego PHP (rys. 5.8).



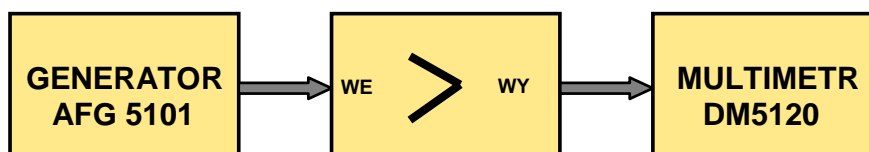
Rys.5.7 Aplikacja klienta (LabWindows/CVI)

W pierwszym przypadku użytkownik instaluje program klienta na swoim komputerze, a w drugim za pomocą przeglądarki łączy się z serwerem WWW i otwiera stronę internetową zawierającą aplet Javy.



Rys.5.8 Aplikacja klienta (aplet Java)

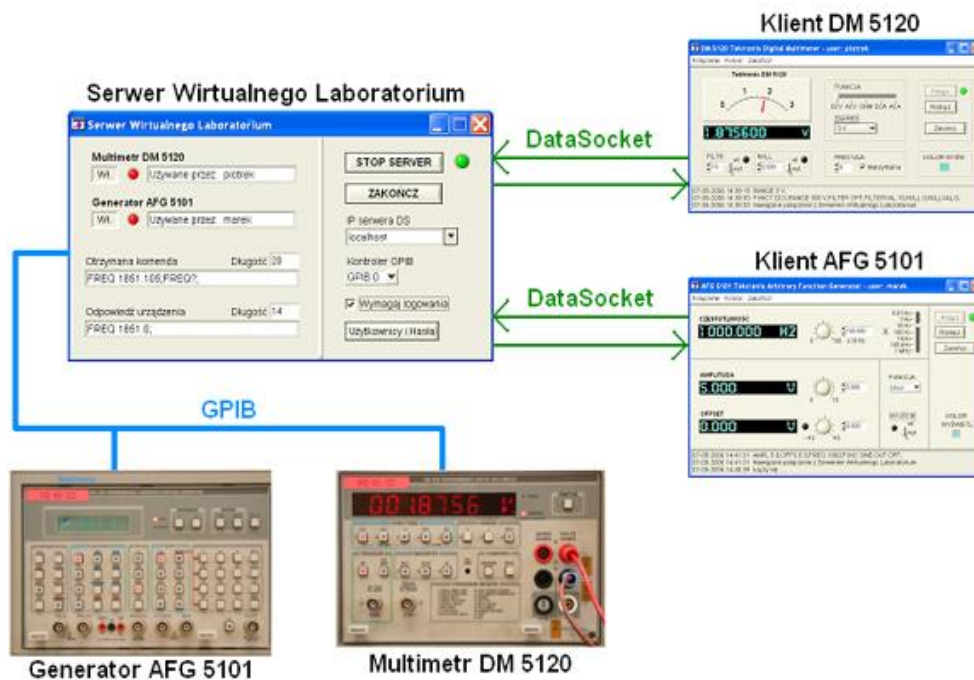
Drugi przykład zdalnego dostępu do laboratorium, również klasyczny, to system do zdejmowania charakterystyk amplitudowo-częstotliwościowych wzmacniaczy, również wykonany jako praca dyplomowa w zakładzie Systemów Informacyjno-Pomiarowych. Schemat blokowy systemu pokazano na rysunku 5.9.



Rys.5.9 Schemat blokowy systemu do zdejmowania charakterystyk amplitudowo- częstotliwościowych wzmacniaczy

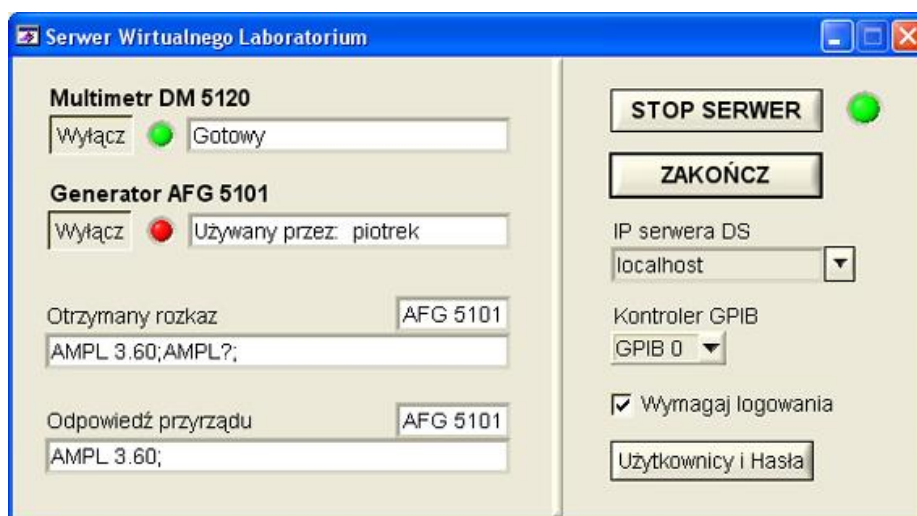
Ze względów dydaktycznych umożliwia on również selektywny dostęp do poszczególnych przyrządów pomiarowych i ich obsługę indywidualną. Do przyrządów tych należą: generator funkcyjny AFG 5101 (Tektronix) oraz multimetr DM 5120 (Tektronix). W związku z tym, przygotowane zostały trzy niezależne programy klienta oraz wspólny program serwera.

Trzy pierwsze aplikacje uwidocznione są na rys. 5.10 zawierającym model komunikacyjny zdalnej obsługi wymienionych wyżej przyrządów pomiarowych.



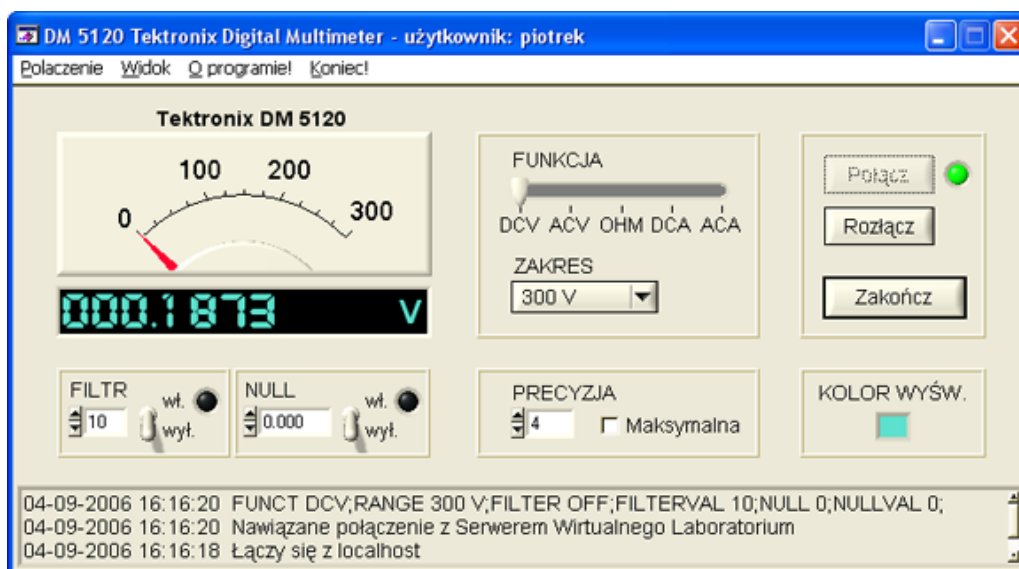
Rys.5.10 Model komunikacyjny zdalnej obsługi przyrządów pomiarowych

Serwer Wirtualnego Laboratorium, przygotowany do obsługi przyrządów (rys. 5.11) – pośredniczy w wymianie danych i rozkazów pomiędzy klientami i przyrządami oraz dokonuje autoryzacji użytkowników łączących się z wirtualnym laboratorium.



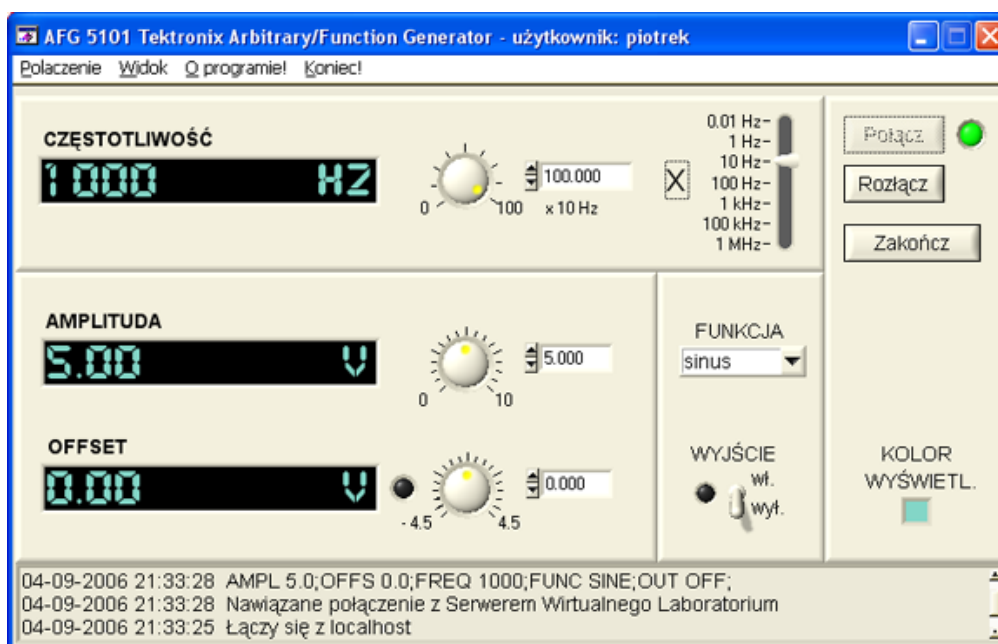
Rys.5.11 Aplikacja serwera AFG 5101/DM5120

Klient Multimetr (rys. 5.12) – pozwala sterować przez Internet multimetrem DM 5120 (wybór funkcji pomiarowej i zakresu, odczyt bieżącego pomiaru).



Rys.5.12 Aplikacja klienta – Multimetr DM 5101

Klient Generator (rys. 5.13) – pozwala sterować przez Internet generatorem AFG 5101 (wybór generowanej funkcji, regulacja częstotliwości i amplitudy generowanego przebiegu).

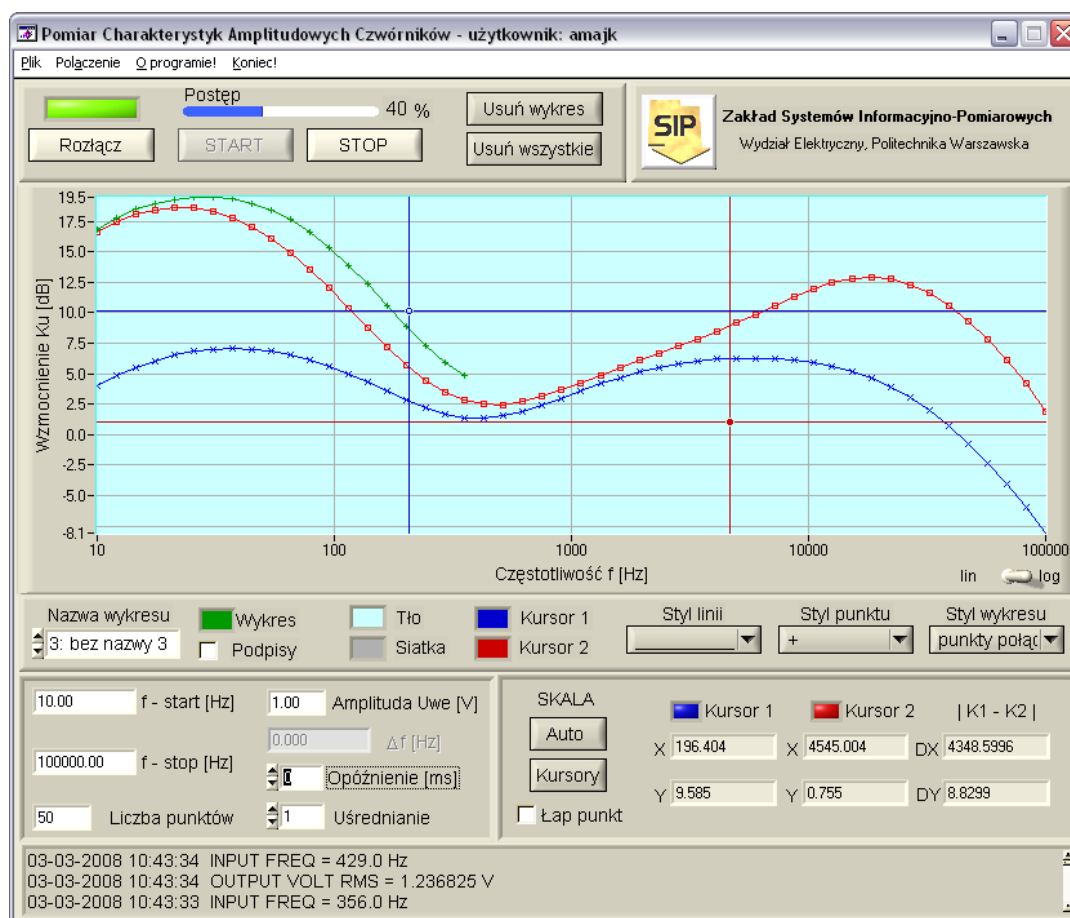


Rys.5.13 Aplikacja klienta – Generator AFG 5101

Czwarta aplikacja (Pomiar Charakterystyk Amplitudowych) przedstawiona jest na rys.5.14, a opisana dokładnie w Przeglądzie Elektrotechnicznym 6/2008. Pozwala ona wykonać automatyczny pomiar charakterystyki amplitudowo–częstotliwościowej wzmacniacza podłączonego do wyjścia generatora i wejścia multimetru, zgodnie ze schematem zawartym na rys. 5.9.

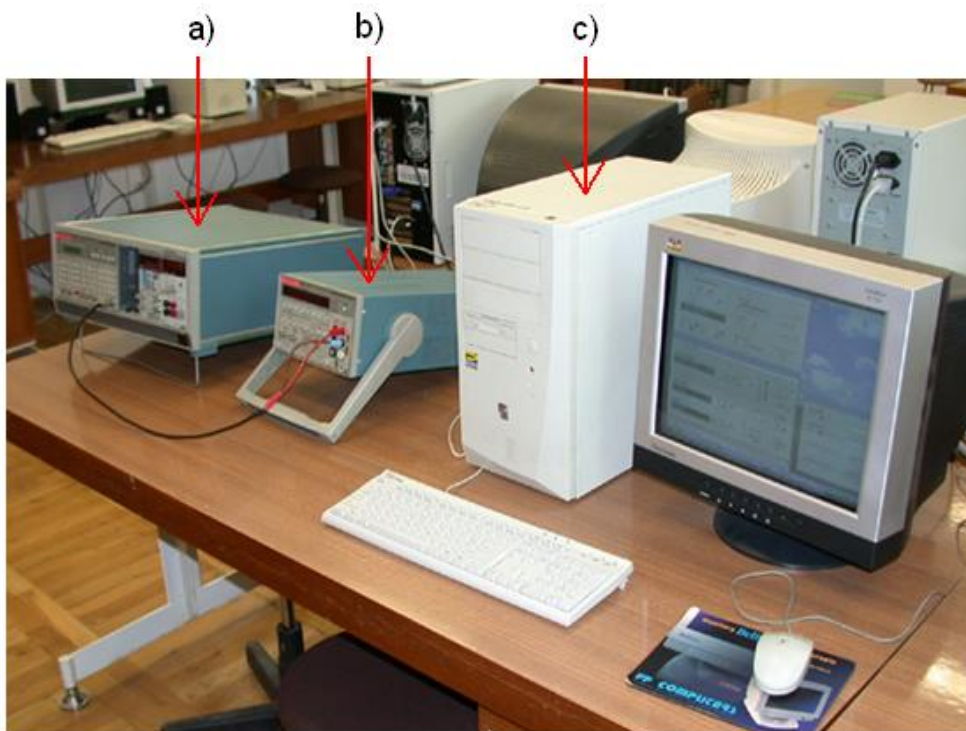
Jak już wspomniano, system pomiarowy jest zbudowany zgodnie z architekturą klient – serwer. Klient łączy się z Serwerem Wirtualnego Laboratorium, aby uzyskać dostęp do jego zasobów, tj. rzeczywistych przyrządów pomiarowych. Aby uzyskać ten dostęp aplikacja klienta musi wysłać do Serwera wszystkie konieczne informacje, tj. nazwę użytkownika, hasło oraz numer identyfikacyjny przyrządu, nad którym chce przejąć kontrolę. W

serwerze dane są automatycznie analizowane, sprawdzana jest autentyczność użytkownika oraz dostępność wybranego przez niego przyrządu, a następnie wysyłana do klienta stosowna odpowiedź.



Rys.5.14 Aplikacja klienta – Pomiar Charakterystyk Amplitudowych

W przypadku gdy autoryzacja użytkownika powiodła się i przyrząd był dostępny, Serwer „tworzy” specjalny kanał DataSocket do komunikacji klienta z przyrządem oraz wysyła do klienta odpowiedź informującą o gotowości przyrządu. Wtedy aplikacja klienta podłącza się do tego kanału DataSocket. Od tej chwili użytkownik może kontrolować przyrząd ze zdalnego komputera, za pośrednictwem panelu (wirtualnej płyty czołowej). Wskazany kanał komunikacyjny, tak jak każdy w tym systemie, jest dwukierunkowy, tzn. umożliwia również przesyłanie komunikatów z przyrządu do klienta. Komunikaty są automatycznie przekazywane do klienta przez serwer. Zdjęcie zestawu laboratoryjnego do pomiaru charakterystyk amplitudowych wzmacniaczy zamieszczono na rys. 5.15.



Rys.5.15 Zdjęcie zestawu laboratoryjnego do pomiaru charakterystyk amplitudowych wzmacniaczy (a – generator AFG5101, b – multimetr DM5120, c – komputer/kontroler)

6. Ćwiczenia do modułu (rozwiązane problemy praktyczne - zadania, projekty)

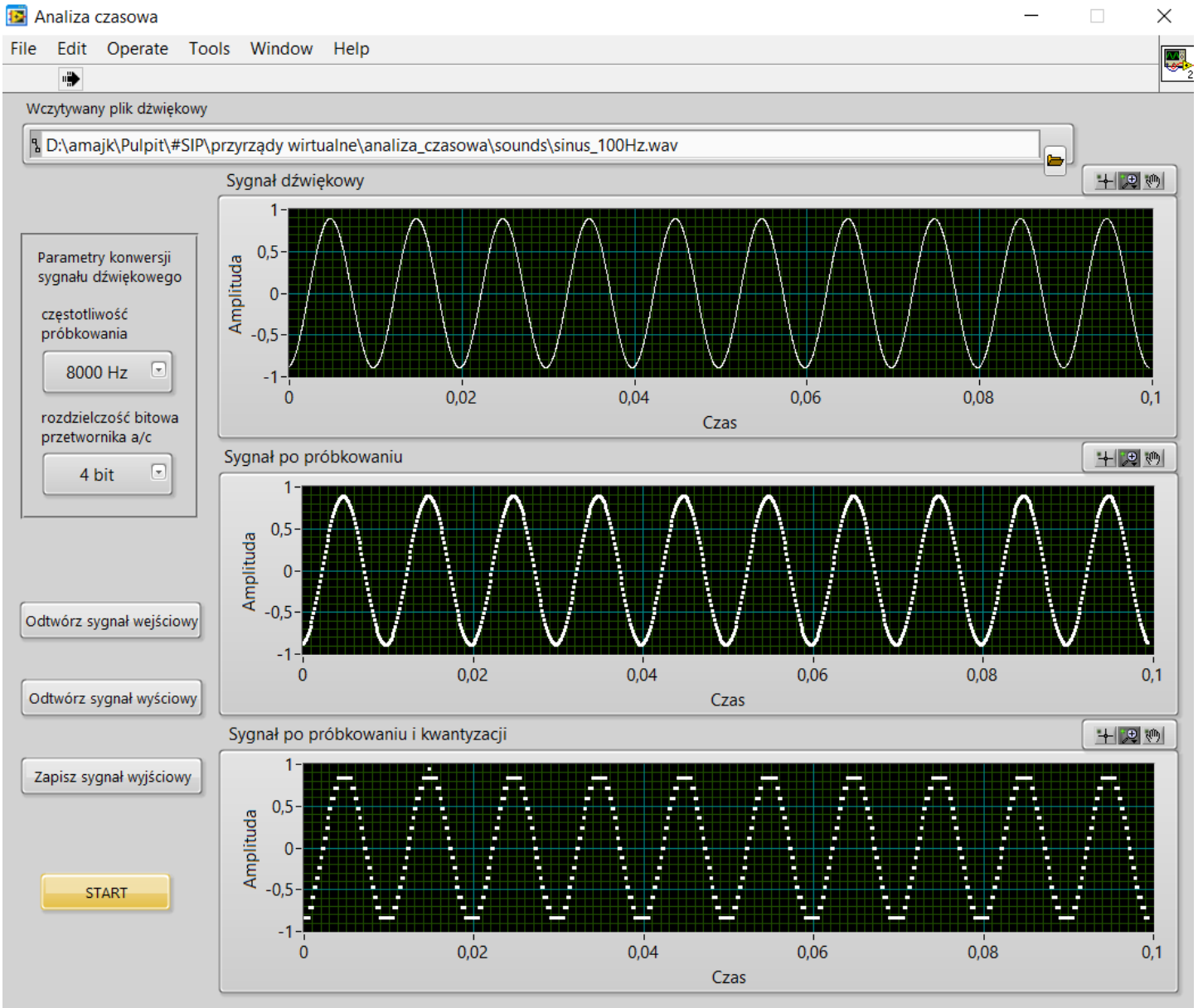
W tej części zostaną zaprezentowane przykłady przyrządów wirtualnych, które mogą posłużyć do rejestracji i analizy różnych sygnałów pomiarowych. Przyrządy takie są coraz częściej używane w miejscu tradycyjnych instrumentów pomiarowych. Mają uniwersalną budowę. Składają się zwykle z komputera i karty zbierania danych (DAQ). Wymieniając oprogramowanie można stworzyć zupełnie inny przyrząd. Jest to wygodne i jednocześnie tanie rozwiązanie. Ponadto przyrządy wirtualne, z racji wykorzystywania komputera, łatwo jest łączyć, a tym samym tworzyć z nich duże i skomplikowane, rozproszone systemy pomiarowe. Programy omawianych przyrządów wirtualnych są dołączone do podręcznika.

6.1. Przyrząd wirtualny do konwersji sygnałów dźwiękowych

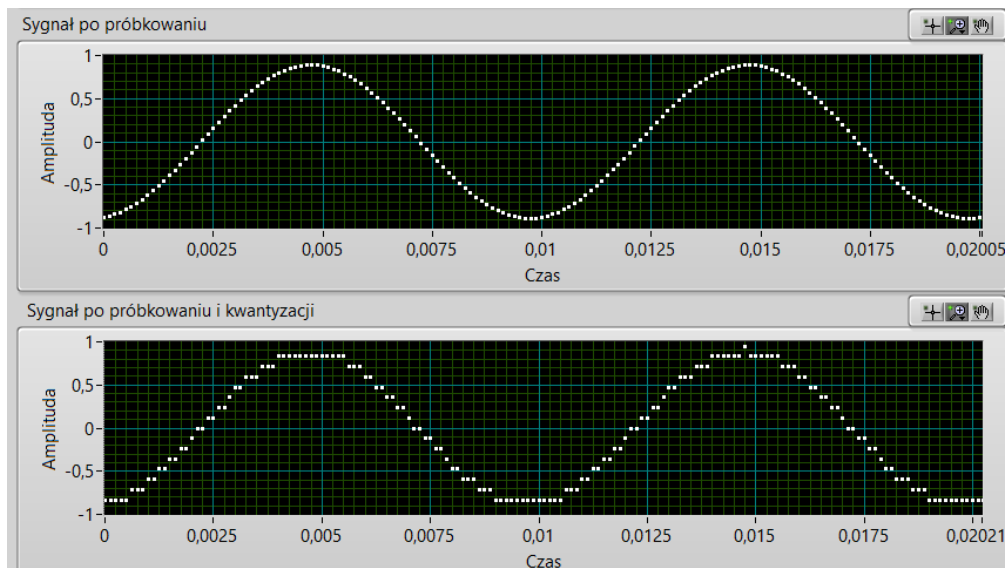
Przyrząd wirtualny do konwersji sygnałów dźwiękowych może posłużyć do zilustrowania wpływu parametrów konwersji sygnału analogowego na cyfrowy, tj. częstotliwości próbkowania i rozdzielczości bitowej, na jakość brzmienia sygnału. Jest to typowy przyrząd o charakterze dydaktycznym, który można wykorzystać do wspomagania np. nauczania cyfrowego przetwarzania sygnałów. Przyrząd został napisany z użyciem oprogramowania LabView i nie wykorzystuje dodatkowego sprzętu do akwizycji danych. Zatem do jego zbudowania potrzebny jest tylko komputer i odpowiednie oprogramowanie (dołączony program konwersja.exe).

Oczywiście komputer przetwarza tylko sygnały cyfrowe, więc w tym przypadku nie będziemy dysponowali sygnałem analogowym. Zamiast tego, do porównań, będzie używany sygnał cyfrowy zarejestrowany z dużą dokładnością. W przypadku dźwięku będą to sygnały zarejestrowane częstotliwością próbkowania 44100 Hz i z rozdzielczością 16 bitów. Dla takich parametrów rejestracji nie słychać w sygnałach dźwiękowych żadnych zniekształceń związanych z konwersją analogowo-cyfrową. Będą to nasze sygnały referencyjne.

Na rys. 6.1 przedstawiony jest panel czołowy przyrządu wirtualnego do konwersji sygnałów dźwiękowych. Pierwszą rzeczą, którą należy wykonać jest wczytanie sygnału dźwiękowego. Należy kliknąć myszką ikonę folderu, która znajduje się na końcu pola z nazwą i ścieżką pliku. Wybieramy plik dźwiękowy i ładujemy go do systemu. W polu nazwy powinna się pojawić nazwa pliku wraz ze ścieżką dostępu. Następnie wybieramy parametry konwersji sygnału dźwiękowego. Mamy do dyspozycji częstotliwości próbkowania: 44100 Hz, 22050 Hz, 11025 Hz, 8000 Hz oraz rozdzielczości: 16, 8, 4, 3 bity. W rzeczywistości, w przypadku dźwięku, nie stosuje się rozdzielczości niższych niż 8 bitów. 4 i 3 bity są tu zamieszczone raczej w celach poznawczych. Po wybraniu odpowiedniej częstotliwości próbkowania i rozdzielczości bitowej należy nacisnąć przycisk START. Zostaną policzone i pokazane odpowiednie sygnały: sygnał po próbkowaniu oraz sygnał po próbkowaniu i kwantyzacji. Można zaobserwować różnice między tymi sygnałami. Na rysunku zaprezentowana jest sinusoida, dla której jest bardzo łatwo zinterpretować wyniki. Sygnały można w różny sposób powiększać i przesuwając, korzystając z narzędzia lupy (rys. 6.2). Po przeprowadzeniu próbkowania mamy informację o sygnale tylko w miejscu wystawienia próbki, po przeprowadzeniu kwantyzacji poziomy amplitud próbek zostają sprowadzone do możliwych wartości (w tym przypadku jednej z 64 wartości – 4 bity rozdzielczości). Sygnał może być oceniony wzrokowo. Możemy też posłuchać sygnału oryginalnego oraz przekonwertowanego w celu oszacowania jego jakości (czy też występujących zakłóceń). Należy zaznaczyć, że sygnał dźwiękowy przed operacją próbkowania jest przepuszczany przez odpowiedni filtr antyaliasingowy w celu uniknięcia zniekształceń. Pasma częstotliwościowe jest zawsze równe połowie częstotliwości próbkowania (czyli w naszym przypadku 40-4000 Hz). Natomiast operacja kwantowania z użyciem czterech bitów powoduje powstanie bardzo wyraźnego szumu kwantyzacji, będącego wynikiem różnic amplitud próbek sygnału oryginalnego i skwantowanego.

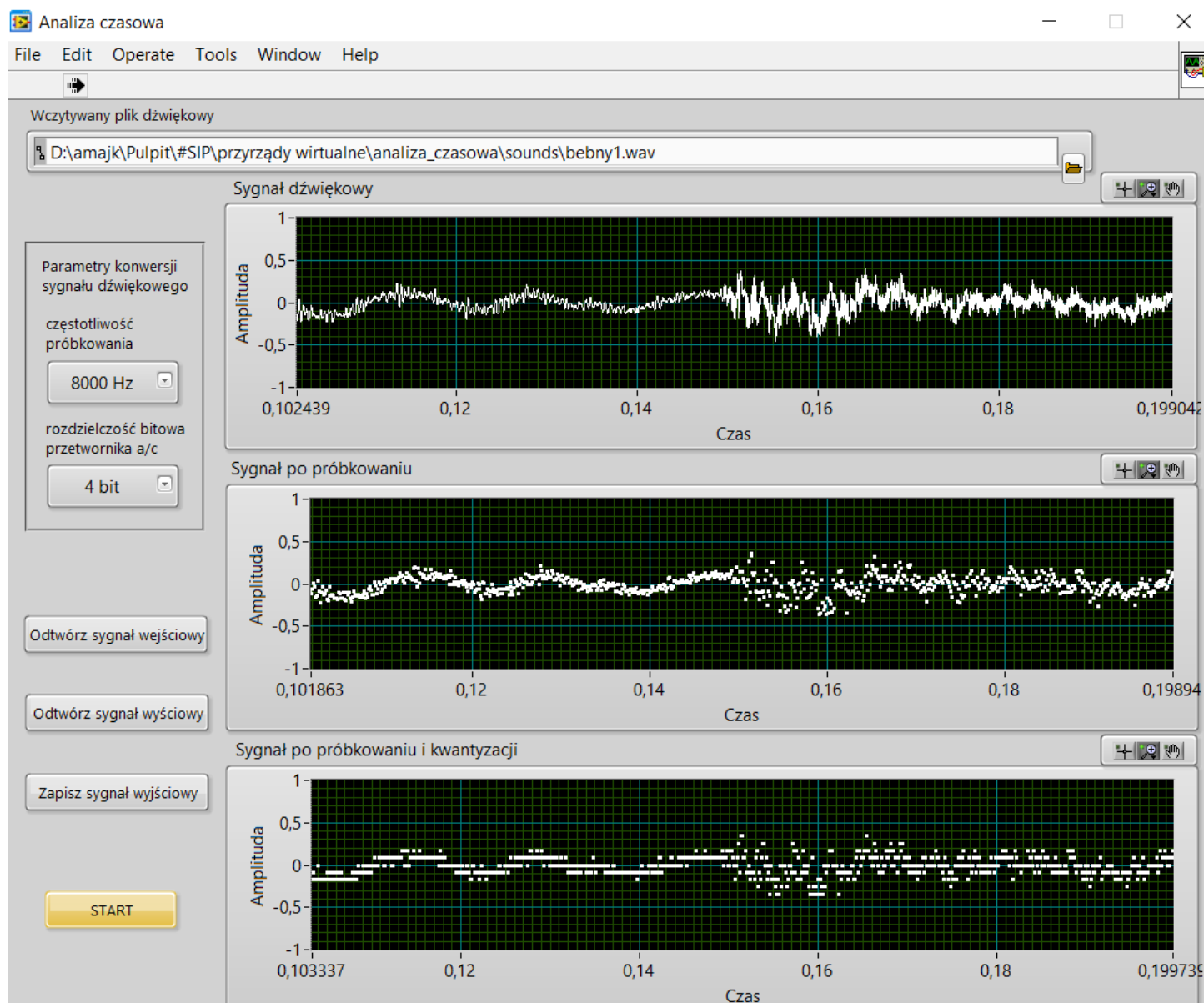


Rys.6.1 Panel czołowy przyrządu wirtualnego do konwersji sygnałów dźwiękowych



Rys.6.2 Przeskalowane sygnały przy użyciu narzędzia lupy

Oczywiście sinusoida nie jest najciekawszym sygnałem dźwiękowym. Weźmy jako następny przykład rzeczywisty sygnał dźwiękowy (muzykę) przekonwertowany z tymi samymi parametrami (8000 Hz, 4 bity). Fragment tego sygnału jest przedstawiony na rys. 6.3. W tym przypadku kształt sygnału jest znacznie bardziej skomplikowany. Można jednak wyraźnie usłyszeć różnice w brzmieniu sygnału oryginalnego i przetworzonego.



Rys.6.3 Analiza sygnału dźwiękowego (muzyka)

Jako dodatkowe ćwiczenie proszę przekonwertować inne sygnały dźwiękowe z różnymi parametrami (częstotliwość próbkowania i rozdzielczość bitowa) i sprawdzić wpływ tych parametrów na brzmienie dźwięku. Jeśli chcemy sprawdzić wpływ tylko jednego parametru, drugi należy ustawić na najwyższą wartość (nie będzie on wtedy zmieniany).

6.2. Przyrząd wirtualny realizujący filtrację cyfrową

Drugi z prezentowanych przyrządów wirtualnych umożliwia cyfrową filtrację sygnałów. Jest on również napisany w środowisku LabView. Jest to przyrząd o charakterze dydaktycznym, do budowy którego jest potrzebny jedynie

komputer i oprogramowanie. Możliwa jest filtracja kilku wybranych sygnałów: sinusoidalnego, trójkątnego, prostokątnego i piłkkształtnego o podanej przez użytkownika częstotliwości. Przyrząd umożliwia zaprojektowanie filtrów: dolnoprzepustowego, górnoprzepustowego, pasmowo-przepustowego lub pasmowo-zaporowego. Możemy skorzystać z rodziny filtrów o skończonej odpowiedzi impulsowej (SOI) lub o nieskończonej odpowiedzi impulsowej (NOI). Parametry filtru, które należy podać, są dla ułatwienia prezentowane w postaci graficznej.

Na rys. 6.4 przedstawiony jest panel czołowy omawianego przyrządu wirtualnego. W tym przypadku sygnałem filtrowanym jest sygnał prostokątny o częstotliwości 100 Hz. Zaprojektowano filtr dolnoprzepustowy SOI o częstotliwości odcięcia ok. 600 Hz. Zatem filtr przepuści trzy pierwsze harmoniczne 100 Hz, 300 Hz i 500 Hz sygnału prostokątnego i zablokuje pozostałe. Przefiltrowany sygnał jest przedstawiony na rysunku w kolorze niebieskim. Poniżej są zamieszczone charakterystyki amplitudowa i fazowa zaprojektowanego filtru. Charakterystyka amplitudowa może być przedstawiona w skali liniowej bądź logarytmicznej. Widać, że pasmo przejściowe mieści się w przedziale 550-650 Hz, co jest zgodne z projektem filtru. Ponieważ filtr SOI ma liniową charakterystykę fazową nie ma zniekształceń fazowych w sygnale przefiltrowanym. Jednak początkowy czas zadziałania filtru jest stosunkowo długi. Zależy on od rzędu filtru (tu 80). Zatem dopiero po przejściu 80 próbek przez filtr zacznie on poprawnie działać.



Rys.6.4 Panel czołowy przyrządu wirtualnego – filtr SOI



Rys.6.5 Panel czołowy przyrządu wirtualnego – filtr NOI

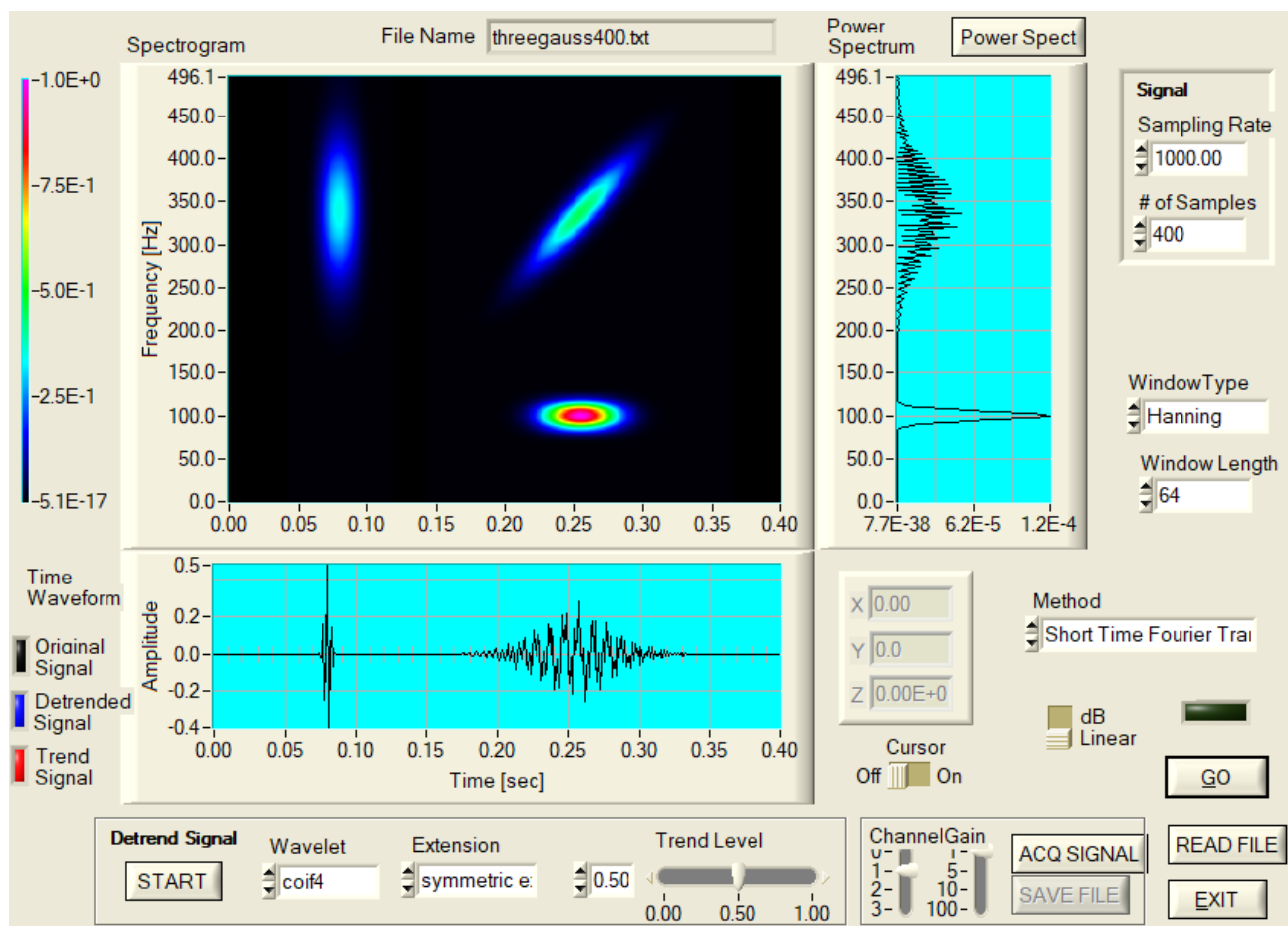
Na rys. 6.5 jest przedstawiona filtracja tego samego sygnału prostokątnego, ale z użyciem filtra dolnoprzepustowego NOI (eliptycznego) o podobnych parametrach (częstotliwość odcięcia 600 Hz). W tym przypadku filtr również przepuści trzy pierwsze harmoniczne 100 Hz, 300 Hz i 500 Hz sygnału prostokątnego i zablokuje pozostałe. Zapodziewalibyśmy się, że wynik filtracji będzie identyczny jak przy zastosowaniu filtra SOI. Jednak tak nie jest. Kształt sygnału po filtracji różni się od uzyskanego wcześniej. Jest to związane z faktem, że filtry NOI mają nieliniową charakterystykę fazową. Stąd też poszczególne harmoniczne są przesuwane w fazie nieliniowo i po ich zsumowaniu sygnał jest zniekształcony. Są to tzw. zniekształcenia fazowe. Jeśli chodzi o charakterystykę amplitudową to jest ona bardzo podobna do otrzymanej dla filtra SOI. Filtry NOI zaczynają działać poprawnie znacznie szybciej niż SOI. Tu rząd filtru wynosi 6, a zatem już po przejściu 6 próbek przez filtr działa on poprawnie.

W ramach dodatkowych ćwiczeń proszę zaprojektować dowolne filtry SOI i NOI górnoprzepustowe, pasmowo-przepustowe i pasmowo-zaporowe i przefiltrować sygnały trójkątne, prostokątne i piłokształtne. Czy otrzymane rezultaty zgadzają się z oczekiwanymi? Filtry cyfrowe są realizowane przez załączony program o nazwie „filtracja_cyfrowa.exe”.

6.3. Przyrząd wirtualny do czasowo-częstotliwościowej analizy sygnałów

Przyrząd wirtualny do czasowo-częstotliwościowej analizy sygnałów został skonstruowany z użyciem komputera i karty zbierania danych (DAQ). Karta zbierania danych umożliwia wczytanie dowolnego sygnału pomiarowego, ale nie jest potrzebna do poprawnego funkcjonowania przyrządu. Dostępne są również wcześniej zarejestrowane sygnały przykładowe. Oprogramowanie przyrządu zostało napisane w środowisku Lab Windows CVI.

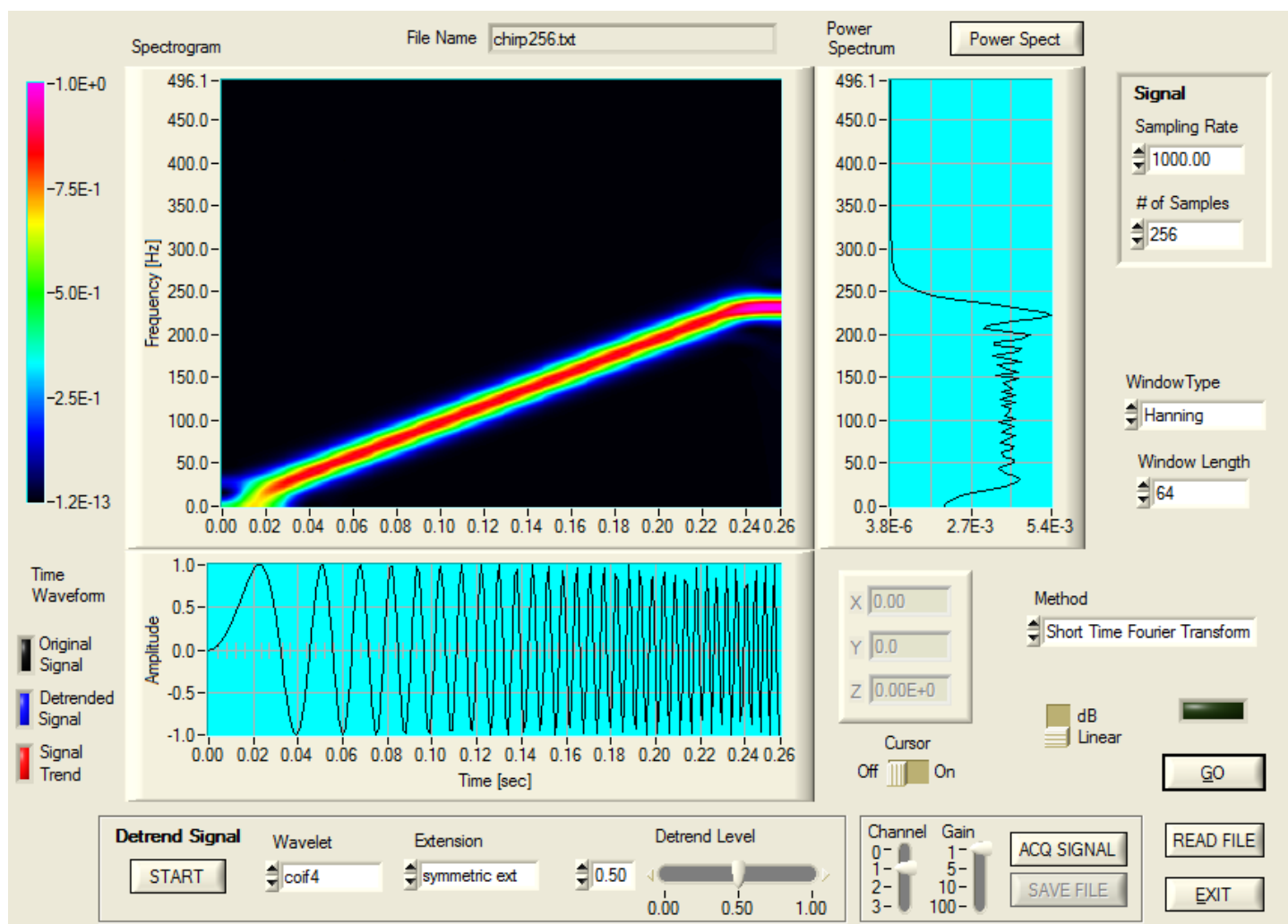
Analizę czasowo-częstotliwościową można przeprowadzić stosując krótkoczasową transformatę Fouriera lub transformatę Gabora (opcja *Method*). Obie transformaty umożliwiają otrzymanie spektrogramu sygnału badanego (czyli wykresu na płaszczyźnie czas-częstotliwość). W przypadku transformaty Gabora stosowane jest okno Gaussowskie. W przypadku krótkoczasowej transformaty Fouriera mamy do dyspozycji okna Haminga, Hanninga, Blackmana Harisa, Blackmana lub prostokątne. Jeżeli mamy podłączoną kartę zbierania danych sygnał badany można wczytać bezpośrednio do komputera (opcja ACQ SIGNAL). Jeśli nie, można wczytać jeden z przykładowych sygnałów (opcja READ FILE). Pliki zawierają dane zapisane w formacie ASCII. W każdym przypadku, przed wczytaniem sygnału, należy podać częstotliwość próbkowania i liczbę wczytywanych próbek (Opcje *Sampling Rate*, *# of Samples*). W przypadku krótkoczasowej transformaty Fouriera należy też podać typ okna (opcja *Window Type*) i szerokość okna (opcja *Window Length*). W przypadku transformaty Gabora należy podać wariancję okna Gaussowskiego (opcja *Var*) i parametr order (opcja *Order*). Im większa wartość tego parametru, tym lepsza rozdzielczość, ale jednocześnie występują silniejsze interferencje. Parametr przyjmuje wartości większe bądź równe zero. Przed analizą sygnału można usunąć z niego trend (opcja *Detrend Signal*).



Rys.6.6 Panel czołowy przyrządu wirtualnego do czasowo-częstotliwościowej analizy sygnałów

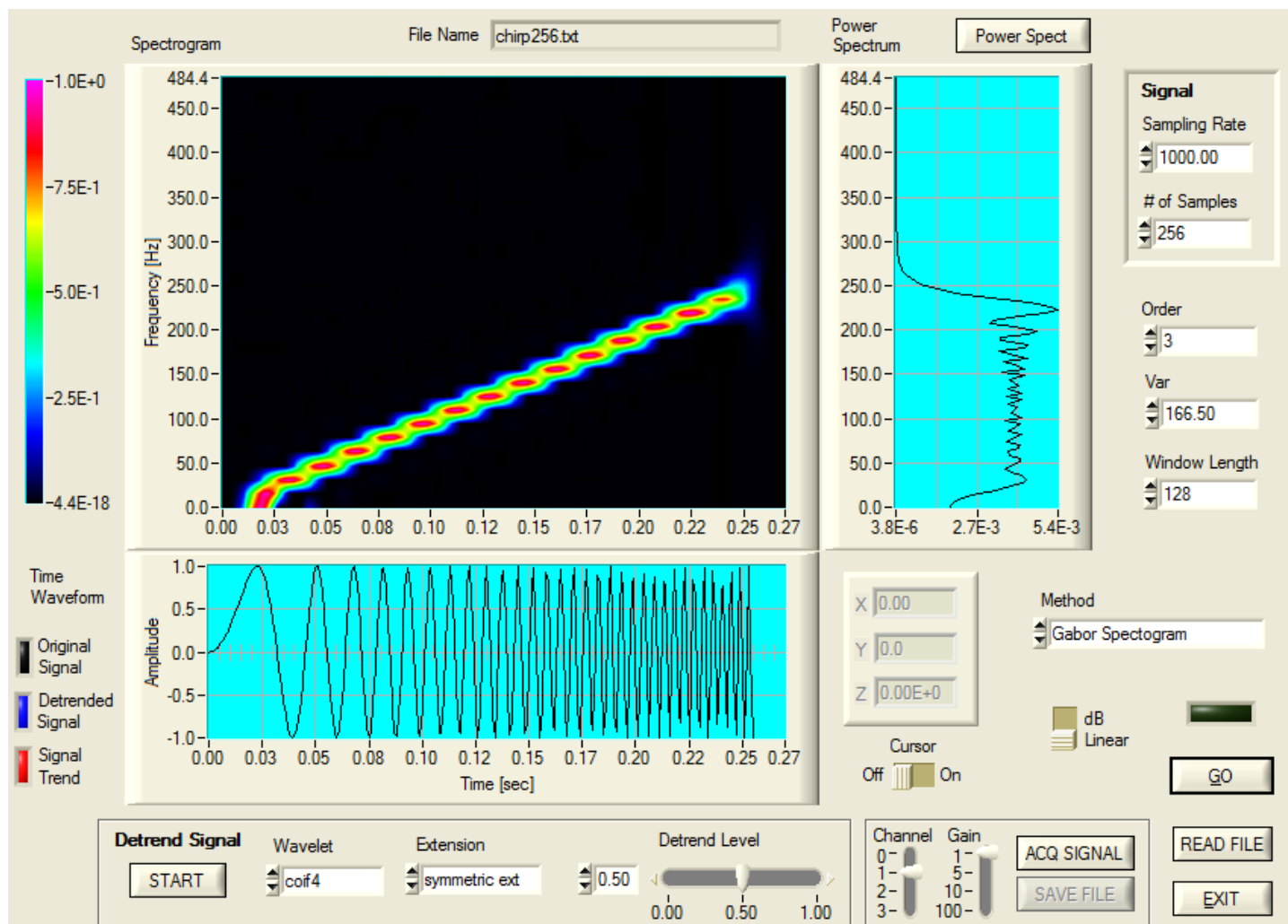
Na rys. 6.6 pokazane są wyniki przykładowej analizy STFT otrzymane przy wykorzystaniu opisywanego wirtualnego przyrządu pomiarowego do przeprowadzania analiz czasowo-częstotliwościowych. Sygnał badany, spróbkowany z częstotliwością 1000 Hz, składa się z dwóch części wyciętych oknami Gaussa. Część pierwszą stanowi sygnał sinusoidalny o częstotliwości 340 Hz, część druga to złożenie dwóch sygnałów: sinusoidalnego o częstotliwości 100 Hz oraz sinusoidalnego o częstotliwości liniowo zmiennej w zakresie 200-500 Hz. Na rys. 6.6, u dołu, pokazany jest przebieg czasowy tego sygnału. Prawa część rys. 6.6 zawiera obraz widma mocy badanego sygnału. Z widma mocy można wywnioskować, jakie składowe częstotliwościowe występują w sygnale badanym, ale nic nie można powiedzieć o chwilach czasu występowania składowych sygnału związanego z tymi częstotliwościowymi. Równie trudno jest określić charakter zmian częstotliwościowych. Natomiast obserwując rozkład na płaszczyźnie t/f (spektrogram), możemy podać zarówno czas występowania składowych sygnału o określonej zawartości częstotliwościowej, jak i charakter zmian częstotliwościowych. Niestety wartości czasu i częstotliwości można określić tylko z pewnym prawdopodobieństwem wyznaczonym przez parametry okna. Zastosowanie wąskiego okna czasowego (i tym samym szerokiego w dziedzinie częstotliwości) objawia się dużym rozmyciem umiejscowienia prążków w dziedzinie częstotliwości i stosunkowo dobrym umiejscowieniem prążków w czasie. Zastosowanie szerokiego okna czasowego daje dobrą lokalizację częstotliwościową i złą czasową.

Na rys. 6.7 przedstawiona jest analiza czasowo-częstotliwościowa przeprowadzona z użyciem krótkoczasowej transformaty Fouriera sygnału o częstotliwości narastającej liniowo w przedziale od 0 Hz do 200 Hz.



Rys. 6.7 Analiza sygnału o częstotliwości narastającej liniowo z użyciem krótkoczasowej transformaty Fouriera

Ze spektrogramu łatwo można odczytać charakter tych zmian. Dla porównania na rys. 6.8 pokazano analizę tego samego sygnału, ale przeprowadzoną z wykorzystaniem transformaty Gabora. Wyniki są prawie identyczne.



Rys. 6.8 Analiza sygnału o częstotliwości narastającej liniowo z użyciem transformaty Gabora

Korzystając z załączonych sygnałów przykładowych proszę przeprowadzić ich analizę czasowo-częstotliwościową i spróbować zinterpretować wyniki (program „jtfa.exe”).

7. Pytania kontrolne

Proszę zaznaczyć poprawne odpowiedzi. Może być jedna bądź dwie poprawne odpowiedzi w każdym pytaniu.

- 1) Do podstawowych elementów tradycyjnego systemu pomiarowego należą:
 - a) *Czujniki pomiarowe
 - b) Karta DAQ
 - c) *Elementy przetwarzania danych
 - d) Komputer PC

- 2) Zadaniem czujnika pomiarowego jest:
 - a) *Przetworzenie wielkości fizycznej, charakteryzującej badany obiekt, do postaci elektrycznej
 - b) Dopasowanie sygnału mierzonego do urządzenia pomiarowego
 - c) Usunięcie średniej z sygnału
 - d) Wstępne przetworzenie sygnału pomiarowego

- 3) Karta zbierania danych:
 - a) Jest specyficznym przyrządem wirtualnym
 - b) *Rozszerza możliwości komputera o akwizycję danych pomiarowych w postaci cyfrowej
 - c) *Zwykle ma przetworniki A/C i C/A, dodatkowe porty WE/WY oraz dodatkowe układy zegarowe
 - d) Jest dostępna standardowo w każdym komputerze PC

- 4) Do najważniejszych parametrów kart zbierania danych należą:
 - a) *Wartość częstotliwości próbkowania przetwornika A/C
 - b) Liczba multiplekserów analogowych
 - c) *Rozdzielczość przetwornika A/C
 - d) Liczba wzmacniaczy pomiarowych

- 5) Wśród elementów karty zbierania danych nie znajdziemy:
 - a) Interfejsu DMA
 - b) Układów zegarowych
 - c) Buforów pamięci
 - d) *Układów odpowiedzialnych za wyświetlanie danych na monitorze komputerowym

6) Idea wirtualnego przyrządu pomiarowego to:

- a) Zapewnienie możliwości połączenia przyrządu pomiarowego z siecią komputerową
- b) *Połączenie komputera z kartą zbierania danych i zaopatrzenia go oprogramowanie, realizujące interfejs użytkownika
- c) Zastosowanie mikroprocesorów (jednokładowych, uniwersalnych, sygnałowych) w przyrządzie autonomicznym
- d) Umożliwienie przetwarzania danych bezpośrednio w autonomicznym przyrządzie pomiarowym

7) Moduły kondycjonowania sygnału umożliwiają:

- a) *Izolację galwaniczną
- b) Zamianę sygnału cyfrowego na analogowy
- c) Przepróbkowanie sygnałów
- d) *Wzmacnianie sygnałów

8) Interfejs IEC-625:

- a) *Jest interfejsem równoległym
- b) Jest interfejsem szeregowym
- c) Jest szybszy niż USB
- d) Jest interfejsem bezprzewodowym

9) System rozproszony to:

- a) System który zajmuje dużą powierzchnię
- b) System, który zawiera więcej niż 5 przyrządów pomiarowych
- c) *System, który może współużywać pamięci masowe do archiwizacji danych dostarczonych z poszczególnych stanowisk,
- d) * System, który może współużywać zasoby sprzętowe zainstalowane w sieci

10) Do typowych sieciowych protokołów komunikacyjnych można zaliczyć:

- a) *UDP - (ang. User Datagram Protocol)
- b) USB
- c) *NTP - (ang. *Network Time Protocol*)
- d) HDMI

11) W modelu klient-serwer

- a) *Serwer gromadzi dane, przetwarza je oraz przechowuje
- b) Może istnieć tylko jeden program klienta
- c) Serwer znajduje się na jednym dedykowanym komputerze
- d) *Klient łączy się z serwerem, przedstawiając swoje zapytania lub żądania wykonania usługi

12) W architekturze peer-to-peer (P2P):

- a) *Każda strona ma równorzędne prawa
- b) Występuje dedykowany kontroler systemu
- c) *każdy komputer może jednocześnie pełnić zarówno funkcję klienta, jak i serwera
- d) Czas obsługi systemu jest bardzo długi

13) Język Java:

- a) Nie nadaje się do oprogramowania systemów pomiarowych
- b) *Jest językiem obiektowym
- c) Jest językiem specjalizowanym
- d) *Istnieją liczne biblioteki wspomagające oprogramowanie rozproszonych systemów pomiarowych w Javie

14) Wirtualne laboratorium:

- a) *Umożliwia dzielenie zasobów sprzętowych pomiędzy różnymi ośrodkami badawczymi
- b) Stanowi tylko symulację różnych przyrządów pomiarowych
- c) *Stanowi złożoną strukturę organizacyjną kontrolowaną zwykle przez wiele serwerów
- d) Nie może być całkowicie bezobsługowe.

15) Architektura wirtualnego laboratorium zwykle umożliwia:

- a) *Dostęp o dowolnej porze dnia i nocy
- b) Manipulowanie rzeczywistymi przyrządami pomiarowymi z możliwością ich uszkodzenia
- c) *Dostęp z dowolnego miejsca na uczelni lub w domu
- d) Zdalną realizację zadań pomiarowych na unikatowych przyrządach

8. Bibliografia

- [1] Arpaia A., Baccigalupi A., Cennamo F., Deponte P.: „*A measurement laboratory on geographic network for remote test experiments*”, IEEE Transactions on Instrumentation and Measurement, vol.49, No.5, October 2000, pp.992-996.
- [2] Bagnasco A., Chirico M., Scapolla A.M.: “*XML Technology to Design Didactical Distributed Measurement Laboratories*”, IEEE Instrumentation and Measurement Technology Conference, Anchorage, USA, 21-23 May 2002, pp. 651-655.
- [3] Benetazzo M., Bertocco M., Ferrero F., Parvis M., Piuri V.: “*A Web-Based Distributed Virtual Educational Laboratory*”, IMTC'99, Wenecja, Włochy, May 1999, pp. 1851-1856.
- [4] Bertocco M., Ferraris F., Offelli C., Parvis M.: “*A Client-Server Architecture for Distributed Measurement Systems*”, IEEE Transactions on Instrumentation and Measurement vol.47, Nr 5 1998.
- [5] Bryzek J.: „*Tworzenie komunikacji cyfrowej dla przetworników pomiarowych*”, COE'96, Szczyrk, pp. 13-30, 1996.
- [6] Cristaldi L., Ferrero A., Piuri V.: “*Programmable Instruments, Virtual Instruments and Distributed Measurement Systems*”. IEEE Instrumentation & Measurement Magazine, September 1999, pp. 20-27.
- [7] Coulouris G., Dollimore J., Kindberg T.: “*Systemy rozproszone podstawy i projektowanie*”, WNT, Warszawa 1999.
- [8] Comer D.E.: „*Sieci komputerowe i intersieci*”, WNT, Warszawa 2001.
- [9] Eckel B.: „*Thinking in Java*”, Helion, Warszawa 2001.
- [10] Ferrero A., Piuri V.: „*A Simulation Tool for Virtual Laboratory Experiments in a WWW Environment*”, IEEE Transactions on Instrumentation and Measurement Magazine, Vol 48, Nr. 3, June 1999.
- [11] Galwas B., Rak R.J.: “*Virtual Laboratory - A Future Part Of The New Web-Based Model of Undergraduate Engineering Studies Developed By Warsaw University Of Technology*”, Joint IMEKO TC-1& XXXIV MKM Conference 2002, 8 - 12 September 2002.
- [12] Godziemba-Maliszewski M.: „*Model systemu pomiarowego rozproszonego terytorialnie*”, Praca Dyplomowa, Politechnika Warszawska, 2001 (opiekun. R.J. Rak).
- [13] Goldberg H.: “*What is Virtual Instrumentation*”, IEEE Instrumentation & Measurement, December 2000.
- [14] Grimaldi D., Libero Nigro L., Pupo F., “*Java-Based Distributed Measurement Systems*”, IEEE Transactions on Instrumentation and Measurement vol.47, Nr 1, 1998.
- [15] Hesselink L., Rizal D., Bjornson E.: „*CyberLab: Remote Access to Laboratories Through the World-Wide-Web*”, International Conference on Advances in Infrastructure for Electronic Bussiness, Science, and Education on the Internet SSGRR'2000, L'Aquila, Italy, 2000.
- [16] “*Java™ Media Framework User's Guide*”.
- [17] Jon B. Postel: „*Transmission Control Protocol*”, RFC 793, September 1981
- [18] Jon B. Postel: „*User Datagram Protocol*”, RFC 793, August 1980
- [19] Projekt badawczy finansowany przez Komitet Badań Naukowych: „*Aplikacje nowoczesnych metod analizy sygnałów w wirtualnych przyrządach pomiarowych*”, Nr rej. 8T10C00921, 2001-2003, Kier.proj.: R.J. Rak.
- [20] Lee K.B., Schneeman R.D.: “*Internet-Based Distributed Measurement System and Control Application*”, IEEE Instrumentation & Measurement Magazine, June 1999, pp. 23-27.
- [21] Lesiak P., Świsulski D.: “*Komputerowa technika pomiarowa – w przykładach*”, Agenda Wydawnicza PAK, Warszawa 2002.
- [22] Lutz A.T.: “*Using TCP/IP as an Instrument Interface*”, Sensors Magazine, July 1998, pp.798-802.
- [23] Martin J.: “*TCP/IP Networking, Architecture, Administration and Programming*”, Prentice Hall 1994.
- [24] Michta E.: „*Rozproszone systemy pomiarowo–kontrolne nowej generacji*”. Pomiary Automatyka Kontrola, Nr 9/98, pp.344-347.
- [25] Michta E.: “*Modele komunikacyjne sieciowego systemu pomiarowo-sterującego*”, Wydawnictwo Politechniki Zielonogórskiej, Zielona Góra, 2000.
- [26] Nawrocki W.: “*Komputerowe systemy pomiarowe*”, WKiŁ, Warszawa, 2002.
- [27] Pyszlak P., Rak R.J., Majkowski A.: “*Selected Problems on the design of a Web Based Virtual Measurement Laboratory*”, IEEE Instrumentation and Measurement Technology Conference, Vail, USA, May 2003.
- [28] Pyszlak P. Rak R.J.: “*The design of a Web Based Virtual Measurement Laboratory*”, XVII IMEKO World Congress, Dubrovnik, June 2003.

- [29] Rak R.J.: *Projektowanie systemów pomiarowych*, Szkoła - Konferencja: Metrologia wspomagana komputerowo, Zegrze k/W-wy, 1993, tom 1 - wykłady str. 187-206.
- [30] Rak R.J.: *Wirtualne przyrządy pomiarowe z kartą Lab-PC+*, Elektronik - magazyn elektroniki profesjonalnej, Nr.5, Maj, 1998.
- [31] Rak R.J.: *"Virtual Instrument – The Main Part of Internet Based Distributed System"*, International Conference on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet - SSGRR'2000, L'Aquila, Italy, 2000.
- [32] Rak R.J., Godziemba-Maliszewski M.: *„Wirtualny przyrząd pomiarowy w systemie rozproszonym terytorialnie jako element nowoczesnych technik w nauczaniu metrologii”*, Krajowy Kongres Metrologii, Politechnika Warszawska, Warszawa, 2001.
- [33] Rak R.J.: *„Modern Tools for Development and Design of Virtual Instruments”*, Third International Conference on Enterprise Information Systems Proceedings vol.2, Setubal, Portugal, July 7-10, 2001 pp.1166-1169.
- [34] Rak R.: *„Wirtualny przyrząd pomiarowy” – realne narzędzie współczesnej metrologii*, OWPW, Warszawa 2003.
- [35] Spoelder H.J.W.: *“Virtual Instrumentation and Virtual Environments”*, IEEE Instrum. Meas. Mag., vol.2, Sept., 1999.
- [36] Stabrowski M.M.: *“Cyfrowe przyrządy pomiarowe”*. PWN, Warszawa 2002