PIOTR GAJ

ZASTOSOWANIE PROTOKOŁU TCP/IP DO TRANSMISJI INFORMACJI DLA POTRZEB PRZEMYSŁOWYCH SYSTEMÓW KONTROLNO-NADZORCZYCH

ROZPRAWA DOKTORSKA

(rew. 3.9.1033)

PROMOTOR

PROF. DR HAB. INŻ. JÓZEF OBER

Spis Treści

1. Wstęp

We współczesnych rozwiązaniach informatycznych kontrolujących procesy przemysłowe stosuje się głównie komputerowe układy automatyki [93, 69]. Elementy tych systemów łączy się za pomocą sieci komputerowych w celu zapewnienia wymiany informacji pomiędzy nimi. Stanowią one tym samym informatyczne systemy rozproszone czasu rzeczywistego.

W celu zapewnienia transmisji informacji w czasie rzeczywistym na poziomie procesu stosowane sieci i protokoły muszą zapewnić zdeterminowaną obsługę wymian. Istnieje szereg komercyjnych protokołów specjalizowanych realizujących tę funkcję. Przykłady mogą stanowić protokoły:

- Modbus [146], SNP [128],
- Profibus [148],
- N10 [126], N80 [6], Interbus [141],
- FIP/WorldFIP [37, 152], CAN/CANOpen [139, 85], oraz wiele innych [2, 138].

Niezaprzeczalną zaletę rozwiązań specjalizowanych stanowi wysoki stopień dopasowania do potrzeb obsługiwanych obiektów, objawiający się spełnieniem wymagań stawianych przez obsługiwane obiekty i proces. Jednak rozwiązania te posiadają również szereg wad. Do podstawowych wad należy wysoki koszt projektowania, wdrożenia i eksploatacji oraz wysoka specjalizacja ograniczająca możliwość tworzenia systemów otwartych.

Jako alternatywę proponuje się najpopularniejszy obecnie standard sieci oraz najpopularniejszy protokół, a mianowicie sieć Ethernet [79] i protokół TCP/IP [77, 14, 15, 16].

Obecnie na całym świecie istnieje silna tendencja do wykorzystywania łącza Ethernet w lokalnych połączeniach rozproszonych systemów przemysłowych. Zastosowanie tego łącza nie ogranicza się do poziomów sieci lokalnych zakładu stanowiących warstwę nadzorczą [96, 97], lecz stanowi podstawowe łącze wymiany informacji w systemie na poziomie procesu [22, 19, 63, 95, 138]. Istnieją trzy powody stosowania takiego rozwiązania:

- cena,
- popularność,
- dostępność.

Koszt całej struktury komunikacyjnej opartej o standard Ethernet jest znikomy w porównaniu ze specjalizowanymi systemami dedykowanymi dla przemysłu. Pomijając koszt oprogramowania narzędziowego i aplikacyjnego koszt przykładowego systemu

komunikacyjnego[1] opartego o Ethernet jest niższy od ok. 10% do 35% od innych, specjalizowanych rozwiązań[2]. Tendencja ta jest jeszcze słabo zauważalna dla cen modułów sterowników swobodnie programowalnych, lecz rosnąca popularność rozwiązań Ethernetowych powinna spowodować w najbliższym czasie zwiększanie dysproporcji pomiędzy kosztami systemów opartych na Ethernecie a systemów specjalizowanych. Dla większości wiodących producentów sprzętu PLC (ang. *Programmable Logic Controller*) tendencja ta jest podobna.

Protokół TCP/IP staje się obecnie standardem komunikacyjnym dla wielu dziedzin gospodarki wykorzystujących sieci komputerowe. Usługi zdalnego dostępu do systemów informatycznych poprzez intersieci wdzierają się do wielu dziedzin życia. Dzieje się tak również w warstwie komunikacyjnej systemów przemysłowych.

Aktualnie istnieje szereg firmowych rozwiązań protokołów opartych o TCP/IP do zastosowań w systemach przemysłowych. Poniżej przedstawiono kilka przykładowych nazw protokołów komercyjnych:

- SRTP firmy Alstom/GE Fanuc [138],
- MELSEC firmy Mitsubishi [138],
- EtherNet/IP firmy Allen-Bradley [138],
- FINS firmy Omron [138],
- SuiteLink firmy Wonderware [138, 135],
- Modbus over TCP firmy Schneider [138, 123, 122] i wiele innych.

Ich specyfikacja przeważnie nie jest publikowana a uzyskanie szczegółowych informacji lub analiz na temat działania tych rozwiązań jest praktycznie niemożliwe.

Celem pracy jest znalezienie ograniczeń i uwarunkowań stosowania protokołu TCP/IP do realizacji komunikacji w systemach informatycznych wykorzystywanych w przemyśle. Stosowanie będzie rozpatrywanie względem pracy protokołu w sieci systemowej opartej na standardzie Ethernet i obsługującej poziom procesu oraz na poziomie intersieci i realizacji zdalnego dostępu do systemów automatyki.

[1] Interfejsy dla jednego komputera PC oraz trzech sterowników PLC
[2] Porównano Ethernet, WorldFIP, Profibus, Modbus; cennik firm GeFanuc i Applicom; stan na wrzesień 2003

2. Tezy pracy

Teza 1:

Zastosowanie odpowiednich modułów programowych w siódmej warstwie aplikacyjnej modelu ISO/OSI interfejsu komunikacyjnego wszystkich abonentów sieci Ethernet stwarza możliwość budowy przemysłowej sieci komputerowej z wymianami zdeterminowanymi czasowo.

Teza 2:

Wykorzystanie modułu deterministycznej kontroli wymian w sieci Ethernet umożliwia zdeterminowaną w czasie realizację wymian sieciowych z wykorzystaniem protokołu TCP/IP.

Teza 3:

Uzupełnienie funkcji klasycznego modułu ściany ogniowej o specjalistyczne dodatkowe funkcje umożliwi stworzenie modułu o roboczej nazwie *firewall++* zapewniającego:

a) wymianę danych pomiędzy lokalną siecią systemową a intersiecią,
b) przekazywanie informacji użytecznej pomiędzy lokalną siecią systemową a intersiecią bez naruszania deterministycznego cyklu wymian sieci systemowej,
c) przekazywanie informacji użytecznej poza sieć systemową wraz z informacją statusową określającą spójność czasową tej informacji.

Teza 4:

Istnieje możliwość zastosowania niezdeterminowanych sieci komputerowych wykorzystujących zestaw protokołów TCP/IP dla potrzeb prezentacji i rejestracji informacji obsługiwanej przez informatyczny system kontrolno-nadzorczy.

3. Streszczenie pracy

Treść pracy została podzielona na piętnaście rozdziałów. Rozdział pierwszy stanowi wprowadzenie do tematyki pracy natomiast wnioski oraz nakreślenie perspektyw rozwoju poruszanych zagadnień stanowią tematykę rozdziału ostatniego.

Kompozycję pracy zbudowano tak, aby wnioski stanowiące rezultaty rozważań w danym rozdziale stanowiły aspekty uwzględniane w rozważaniach prowadzonych w dalszej części pracy. Dyskutowane wątki doprowadzają do wykazania tez pracy, które zamieszczone są w rozdziale drugim.

W rozdziale czwartym przedstawiono podstawowe pojęcia związane z tematem pracy. Naświetlono pojęcie obiektu przemysłowego, jakie są ich rodzaje oraz jakie zjawiska opisują działanie obiektów z punktu widzenia tematyki pracy, w szczególności determinizmu działania komunikacji. Dodatkowo opisano problemy determinizmu w sieciach przemysłowych, które są kluczowe w rozważaniach na temat warstw komunikacyjnych systemów czasu rzeczywistego.

Rozdział piąty dotyczy istotnych zagadnień związanych z wykorzystywaniem sieci komputerowych w zastosowaniach przemysłowych. Szczególnie naświetlono te aspekty, które mają duży wpływ na specyfikę pracy systemów informatycznych obsługujących procesy przemysłowe. Zaproponowano podział wymian sieciowych na płaszczyzny oraz przeprowadzono klasyfikację abonentów. Dla potrzeb dalszych rozważań przedstawiono model warstwowy interfejsów sieciowych, na którym bazuje reszta pracy. Zaprezentowano również wykorzystanie protokołu TCP/IP w sieciach komputerowych oraz wykorzystanie sieci Ethernet w konfrontacji z problemem czasu rzeczywistego. Na koniec rozdziału zamieszczono informacje dotyczące obecnego stanu rozwoju technologicznego sieci komputerowych i jego wpływu na rozpatrywane zagadnienia.

W rozdziale szóstym, rozpatrzono wykorzystanie protokołu TCP/IP dla potrzeb dostarczania informacji dla systemów informatycznych pracujących na poziomie procesów przemysłowych. Rozważania oparto na sieci Ethernet jako optymalnej sieci dla wykorzystywania protokołu TCP/IP w kontekście aplikacji przemysłowych. Zaproponowano w nim wykorzystanie warstw nadrzędnych do kontrolowania wymian, co jest związane z tezą pierwszą pracy.

W rozdziale siódmym zaproponowano podział systemów na otwarte, zamknięte i separowane. Zanalizowano możliwość pracy systemu przemysłowego z protokołem TCP/IP w intersieciach. Wyprowadzono wniosek, iż najlepsze rozwiązanie stanowi system separowany ze specjalnym abonentem pośredniczącym.

Rozdział ósmy dotyczy budowy warstwy aplikacyjnej abonenta pośredniczącego. Zaproponowano konstrukcję modułu programowego nazwaną *firewall++*. Budowa i działanie tego modułu jest powiązane tezą pierwszymi trzema tezami pracy.

W rozdziale dziewiątym przedstawiono uwarunkowania stosowania protokołu intersieciowego w informatycznych systemach przemysłowych. Zaproponowano mechanizmy określania jakości danych użytecznych krążących w intersieci na bazie wyselekcjonowania odrębnych obiegów informacji w stosowanych strukturach. Przedstawiono również problemy wykorzystywania protokołu IP w obecnie stosowanej wersji oraz ich potencjalne rozwiązanie dla wersji następnej.

W rozdziale dziesiątym zawarto rozważania na temat potrzeb, celowości i możliwości wykorzystania ujednoliconych usług intersieciowych. Podano przykłady dla usług, które wydają się być najbardziej przydatne dla informatycznych systemów przemysłowych. Opisano różnego typu drogi realizacji takich systemów od rozwiązań typowych po systemy wbudowane (ang. *embedded*). W rozdziale rozważono aspekty bezpieczeństwa stosowania protokołu TCP/IP, w szczególności ze strony bezpieczeństwa kontrolowanego procesu oraz jego właściciela i użytkownika. Poruszane w rozdziałach 9 i 10 zagadnienia dotyczą tezy czwartej pracy.

W kolejnych trzech rozdziałach wykonano analizy czasowe różnych zestawów protokołów dla rozważanych przypadków wykorzystania sieci Ethernet i stosu TCP/IP. W części zawartej w rozdziale jedenastym, wykonano analizę czasową deterministycznych protokołów sieci specjalizowanych oraz protokołów transportowych TCP i UDP dla wykorzystania w sieci Ethernet. Skoncentrowano się w nim na porównaniach sprawności i przepustowości użytecznej. Większość obliczeń wykonano przyjmując szereg uproszczeń zarówno w kwestii analizy ruchu jak i obszaru działania protokołu. Z punktu widzenia systemu przemysłowego wyniki obliczeń stanowią wyznacznik do porównań, na podstawie których wyciągnięto szereg interesujących wniosków przydatnych w procesie doboru protokołów dla aplikacji. W rozdziale dwunastym wykonano analizę czasową protokołów UDP i TCP z uwzględnieniem warstw nadrzędnych kontrolujących wymiany według modelu wymian Master-Slave i PDC. Podobną analizę wykonano w rozdziale trzynastym zakładając tunelowanie transakcji cyklicznych sieci Modbus i WordFIP w sieci Ethernet z wykorzystaniem warstwy transportowej UDP.

W rozdziale czternastym zawarto określenie zakresu stosowalności protokołu TCP/IP dla potrzeb systemów informatycznych pracujących w warstwach kontroli procesów przemysłowych. Wyznaczono grupę zadań funkcjonalnych realizowanych przez informatyczny system kontroli i zaproponowano trzy kryteria stosowalności protokołu, jakimi są proces, użytkownik i bezpieczeństwo informacji. Zaproponowano grupy rozwiązań spełniających powyższe kryteria względem możliwości stosowania protokołu TCP/IP. Przedstawiono również zestawienie możliwości stosowania stosu TCP/IP względem zadań funkcjonalnych systemów kontrolno-nadzorczych.

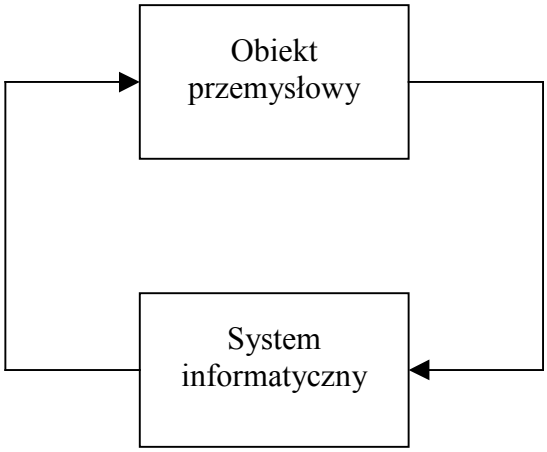
Na końcu pracy znajdują się dodatki w postaci indeksów i spisów oraz załączniki. W załączniku VI.A, zamieszczono przykłady praktycznych aplikacji naświetlających poruszane problemy, a zrealizowanych przez autora lub przy jego współudziale. W pracy znajdują się odwołania do szeregu przeprowadzanych obserwacji i testów, których opis i wyniki zamieszczono w załącznikach VI.B i VI.C.

4. Prezentacja problematyki

Ponieważ cała praca dotyczy systemów informatycznych wykorzystujących obiekty obsługujące procesy przemysłowe, na wstępie naświetlono pojęcie obiektu przemysłowego, jakie są ich rodzaje oraz jakie zjawiska opisują działanie obiektów z punktu widzenia tematyki pracy.

4.1. Obiekt

Proces przemysłowy, jako przebieg następujących po sobie zjawisk fizycznych, mających związek przyczynowo skutkowy, odnoszący się do obróbki i przeróbki materiału [82], z punktu widzenia automatyki oraz informatyki jest obsługiwany przez obiekty przemysłowe.



Rys. 1 Obiekt przemysłowy i jego interakcja z systemem informatycznym

Obiektami takimi mogą być różnego typu urządzenia jak aparatura kontrolno pomiarowa, urządzenia wykonawcze, urządzenia przetwarzające typu sterownik swobodnie programowalny czy komputer. Z punktu widzenia procesu przemysłowego obiekty podzielono na trzy kategorie:

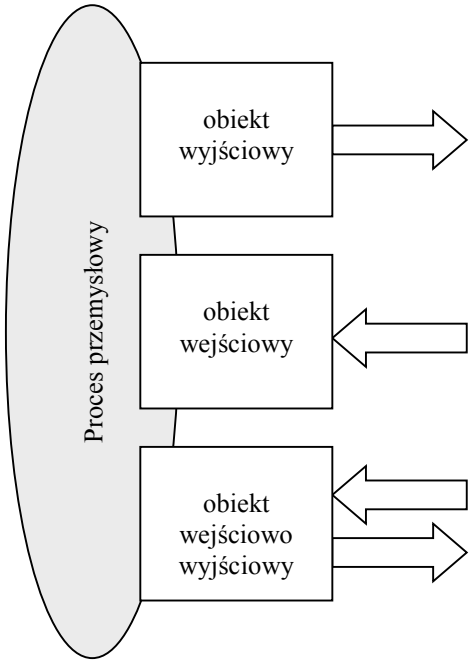
- inicjatory (obiekty pobierające informację o stanie procesu; obiekty wejściowe),
- układy wykonawcze (obiekty modyfikujące stan procesu; obiekty wyjściowe),
- układy mieszane (obiekty pobierające i wyprowadzające informację; obiekty wejściowe i wyjściowe).

W celu wykazania, iż z punktu widzenia informatycznego obiekty przemysłowe mają budowę warstwową, gdzie z jednej strony istnieje warstwa sterująca obiektem a z drugiej warstwa współpracująca z procesem fizycznym, przedstawiono rozgraniczenie typów danych obsługiwanych przez te obiekty. Obiekty komunikują się z procesem fizycznym przekazując do lub z procesu informację procesową. Mowa tu o informacji reprezentującej fizyczny stan

procesu lub na niego wpływającą. Jako przykład takich informacji można podać informację reprezentującą wartości temperatur, ciśnień, stany urządzeń wykonawczych i tym podobne dane odzwierciedlające zjawiska fizyczne.

Dla poprawnej pracy obiektu informacja procesowa może być niewystarczająca i należy ją wzbogacić o dane, które opisują pracę tego obiektu oraz stanowią informację współdzieloną z innymi obiektami. Stanowi to informację użyteczną danego obiektu, określającą jak dany obiekt powinien współdziałać z procesem fizycznym oraz resztą obiektów. Do informacji użytecznej, oprócz informacji procesowej należy zaliczyć informacje typu parametry pracy, komunikaty diagnostyczne, rozkazy wykonawcze, dane synchronizujące pracę systemu itp. Z punktu widzenia tematu pracy, interesujące jest przetwarzanie informacji użytecznej.

Proces przemysłowy widziany jest przez system informatyczny jako zbiór przetwarzających informację użyteczną obiektów, które mogą się komunikować ze światem zewnętrznym. Schematycznie przedstawiono to na rysunku 2. Strzałki na rysunku 2 obrazują informację pobieraną z procesu bądź zwracaną do procesu z poziomu systemu informatycznego. W skład takiego systemu mogą wchodzić obiekty związane z procesem oraz obiekty bezpośrednio z procesem niezwiązane jak np. stacje SCADA [80, 94].



Rys. 2 Rodzaje obiektów przemysłowych

Jeżeli proces przemysłowy posiada więcej niż jeden obiekt, wówczas obiekty takie, aby umożliwić sterowanie i nadzór nad zjawiskami procesu muszą mieć możliwość przekazywania danych użytecznych między sobą. Wprowadzono rozróżnienie obiektów przemysłowych z punktu widzenia ich możliwości komunikacyjnych. Rozróżnienie to bazuje na przekazywaniu informacji użytecznej, w skład której nie wchodzi informacja wymagana do zrealizowania komunikacji taką czy inną techniką. Proponuje się następujący podział:

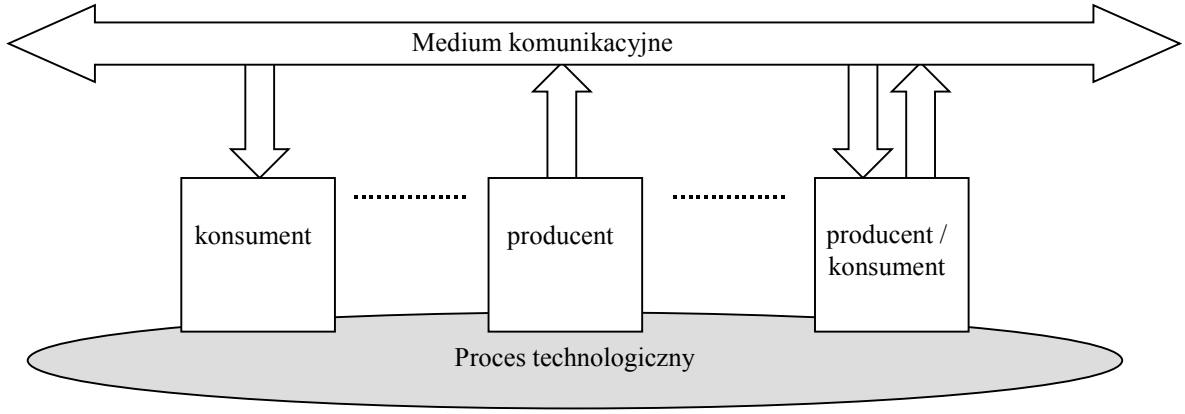
- producenci informacji (obiekty zapisujące informację do innych obiektów),
- konsumenci informacji (obiekty odczytujące informację z innych obiektów),

– obiekty mieszane, producencko – konsumenckie.

Podział ten oparto na podziale abonentów występującym w sieciach komputerowych opartych na modelu kontroli wymian typu PDC [114]. Obiekty przemysłowe komunikujące się z innymi obiektami, ze względu na analizę sieciowego aspektu ich wykorzystywania i chcąc podkreślić ten właśnie aspekt, nazwano abonentami obiektowymi.

4.2. Komunikacja

Otrzymano zestaw trzech typów obiektów – abonentów obiektowych, który zamieszczono na rysunku 3.



Rys. 3 Podział obiektów przemysłowych względem komunikacji

Ponieważ temat pracy dotyczy zagadnień komunikacji, dla dalszych rozważań przyjęto, iż każdy obiekt obsługuje daną grupę informacji użytecznej wymienianą pomiędzy innymi obiektami systemu informatycznego obsługującymi proces przemysłowy. Grupy informacji rozumiane są jako grupy pewnych abstrakcyjnych zmiennych reprezentujących niepodzielne jednostki informacyjne stanowiące elementy informacji użytecznej obiektu. Zmienne obsługiwane są przez system jako całość a ich fizyczną reprezentację dla systemów przemysłowych mogą stanowić dla przykładu wartości pomiarów, stany dyskretne, rozkazy, nastawy, flagi stanu, zdarzenia itp.

Wymagania procesu specyfikowane są względem obiektów go obsługujących i określają mniej lub bardziej deterministyczne zachowanie tychże obiektów. Determinizm rozumiany jako twierdzenie, że wszystkie zjawiska podlegają nieuchronnym prawidłowością, i że każde zdarzenie jest jednoznaczne i w sposób konieczny wyznaczone przez ogół warunków, w jakich zachodzi [82], ma swoje przełożenie na opisywane zagadnienia. Proces przemysłowy, aby przebiegał prawidłowo, czyli aby zostały zachowane wszystkie wymagane związki przyczynowo-skutkowe musi przebiegać w sposób, gdzie każde zdarzenie i zjawisko zachodzi w sposób jednoznaczny w stworzonych dla niego warunkach. Warunki stwarzane są przez obiekty, zatem aby było to możliwe, obiekty obsługujące proces muszą również działać w sposób deterministyczny. Proponuje się zdefiniować pojęcie determinizmu w odniesieniu do działania obiektów przemysłowych.

Na podstawie słownikowych definicji [82, 86] stwierdzono, iż determinizm działania obiektu polega na takiej jego pracy, aby każda reakcja na zdarzenia pojawiające się spoza obiektu była jednoznaczna w danych warunkach. Deterministyczny charakter procesu przemysłowego wymusza charakter przepływu informacji pomiędzy procesem a obiektem. Najistotniejszym parametrem określającym ten charakter jest czas. Występuje zatem czas w kontekście determinizmu, czyli pojęcie determinizmu czasowego. Determinizm czasowy w funkcjonowaniu obiektu definiuje się jako zgodne z działaniem deterministycznym reakcje obiektu na zdarzenia, w określonym i skończonym czasie. Dodatkowo proponuje się rozróżnienie determinizmu czasowego na określony ściśle (ostry) i określony granicznie (nieostry). Przy determinizmie czasowym ściśle określonym reakcja następuje po upływie określonego czasu od zdarzenia inicjującego, natomiast w determinizmie określonym granicznie reakcja musi nastąpić nie później niż do określonego czasu od zdarzenia inicjującego:

determinizm określony ściśle:

$$T_{OS} = T_{G,} \tag{1}$$

determinizm określony granicznie:

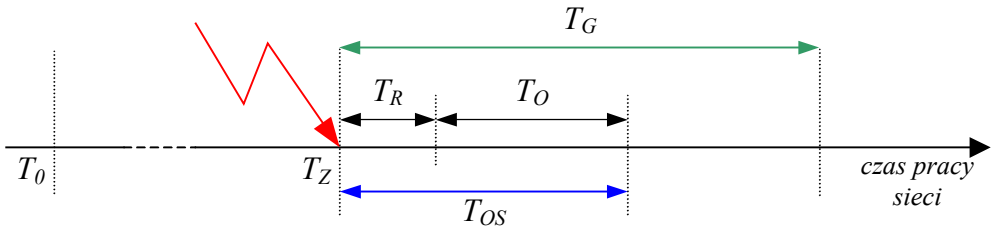
$$T_{OS} \leq T_{G,} \tag{2}$$

gdzie:

T_{OS} – czas obsługi sieciowej zdarzenia Z,

T_Z – czas wystąpienia inicjującego zdarzenia Z,

T_G – czas graniczny obsługi zdarzenia Z.



Rys. 4 Determinizm określony w dopuszczalnych granicach

Przy czym:

$$T_{OS} = T_R + T_{O,} \tag{3}$$

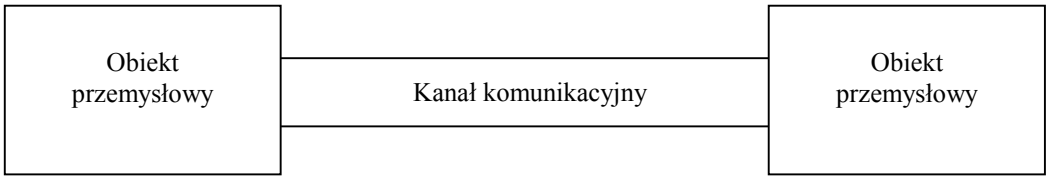
gdzie:

T_R – czas reakcji na zdarzenie Z (np. czas uzyskania dostępu do łącza),

T_O – czas obsługi zdarzenia Z (np. czas realizacji transmisji).

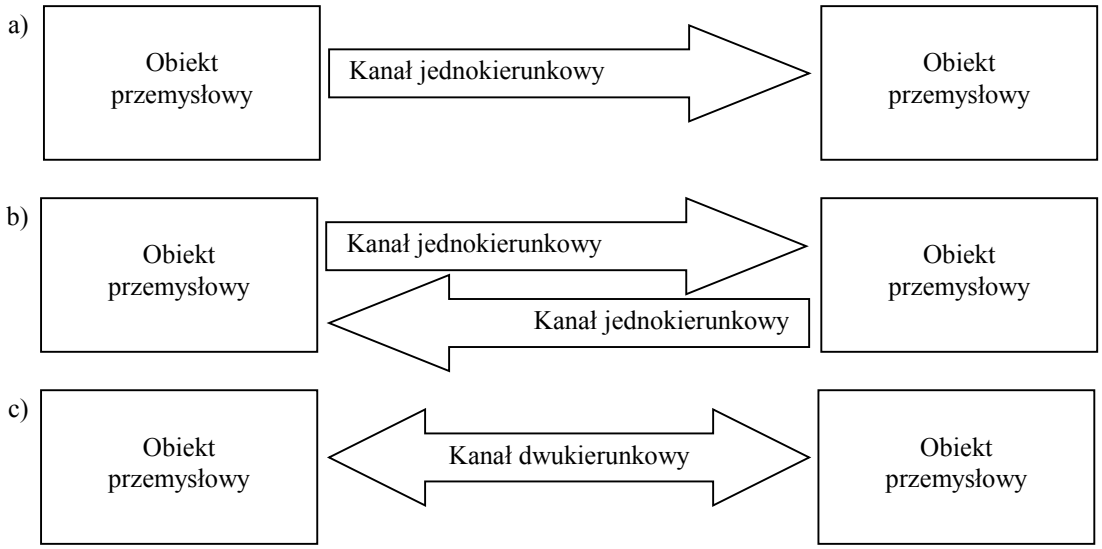
Przy określaniu powyższych czasów należy mieć zawsze na uwadze pewną niedokładność ΔT wynikającą ze zjawisk inercji.

Aby naświetlić problem determinizmu obiektów i przesyłania danych między takimi obiektami należy się przyjrzeć połączeniu typu punkt – punkt (ang. *point-to-point*). Schemat połączenia pokazano na rysunku 5.



Rys. 5 Połączenie obiektów typu punkt–punkt

Rozważono trzy przypadki budowy kanałów komunikacyjnych. Przedstawiono je na rysunku 6.



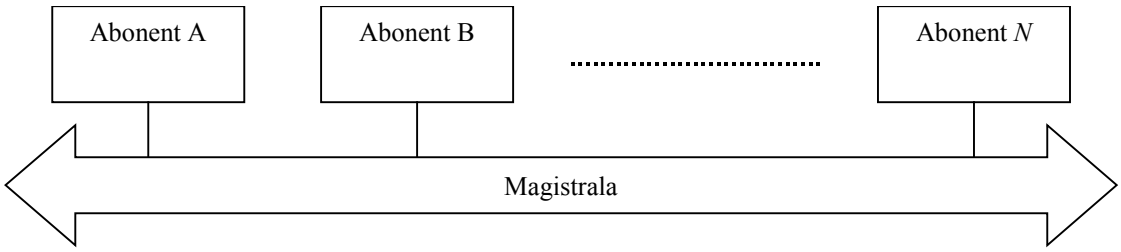
Rys. 6 Rodzaje kanałów komunikacyjnych w połączeniu punkt-punkt

Dwa pierwsze przypadki (rysunek 6a i 6b) z racji samej budowy gwarantują zdeterminowane czasowo działanie systemu. Warunkiem jest determinizm czasowy obiektów. W tych przypadkach nie istnieje, poza awarią, możliwość opóźniania transmisji lub gubienia pakietów. Jednak w rozwiązaniach praktycznych tego typu konstrukcje stosuje się rzadko i są dość kosztowne. Działanie dwukierunkowego kanału z przypadku trzeciego (rysunku 6c) polega na przesyłaniu informacji w obu kierunkach tym samym łączem. Może zaistnieć przypadek, gdy oba obiekty w tym samym czasie rozpoczną transmisję danych, doprowadzając tym samym do powstania kolizji i tracąc właściwość determinizmu czasowego wymian informacji. Aby rozwiązać ten problem i utrzymać determinizm wymian, należy stosować transmisję synchroniczną lub dla asynchronicznej wprowadzić mechanizm synchronizacji wymian (np. typu ang. *hand shake* ze standardu RS lub protokół nadzorujący dostęp). Mechanizm synchronizacji musi być stosowany również dla kanałów jednokierunkowych, gdy obiekty wymagają komunikacji zsynchronizowanej.

Budowanie deterministycznej warstwy komunikacyjnej na bazie połączenia typu punkt-punkt, byłoby zatem wskazane. Niezależnie od budowy kanału połączenia takie nie wymagają stosowania skomplikowanych reguł dostępowych do medium oraz adresacji, czyli nie stwarzają problemów opóźnień i utraty danych. Jednak połączenia każdy z każdym w systemie informatycznym, w skład którego wchodzi już kilka abonentów stają się technicznie skomplikowane i kosztowne. Rozrasta się również warstwa aplikacyjna, która

w sposób niejawny przejmuje zadania adresowania wymian. Dlatego dla dalszych rozważań zrezygnowano z połączeń typu punkt-punkt na rzecz popularnych i tanich połączeń magistralowych. Jedną z sieci opartych na tej topologii jest Ethernet [74, 79], czyli sieć, na której bazuje się w dalszej części pracy.

Połączenie magistralowe (rys. 7) bazuje na kanale dwukierunkowym ze wspólnym medium, przy czym w przeciwieństwie do połączenia punkt-punkt, do magistrali przyłączone jest wiele abonentów [32]. Stosowanie magistrali wymusza kontrolę dostępu do medium jako do obiektu współdzielonego. Sieć Ethernet posiada w swoim standardzie zdefiniowane mechanizmy spełniające tę funkcję. Przydatność tych mechanizmów dla systemów przemysłowych została opisana w rozdziale 5.



Rys. 7 Kanał komunikacyjny w formie magistrali

Istnieje jeszcze jeden sposób umożliwiający łączenie obiektów ze sobą. Jest to tzw. (ang.) *switching technology*. Polskim odpowiednikiem nazwy zaczerpniętym z zagadnień telefonii może być określenie technologii komutacyjnej. Do budowy kanału transmisyjnego pomiędzy wieloma obiektami, jest to rozwiązanie bardzo dobre, gdyż umożliwia dynamiczną realizację połączeń typu punkt-punkt a co za tym idzie rezerwację deterministycznych kanałów. Jednak rozwiązania takie są obecnie zbyt kosztowne, aby było ekonomicznie uzasadnione stosowanie ich w warstwach komunikacyjnych systemów lokalnych.

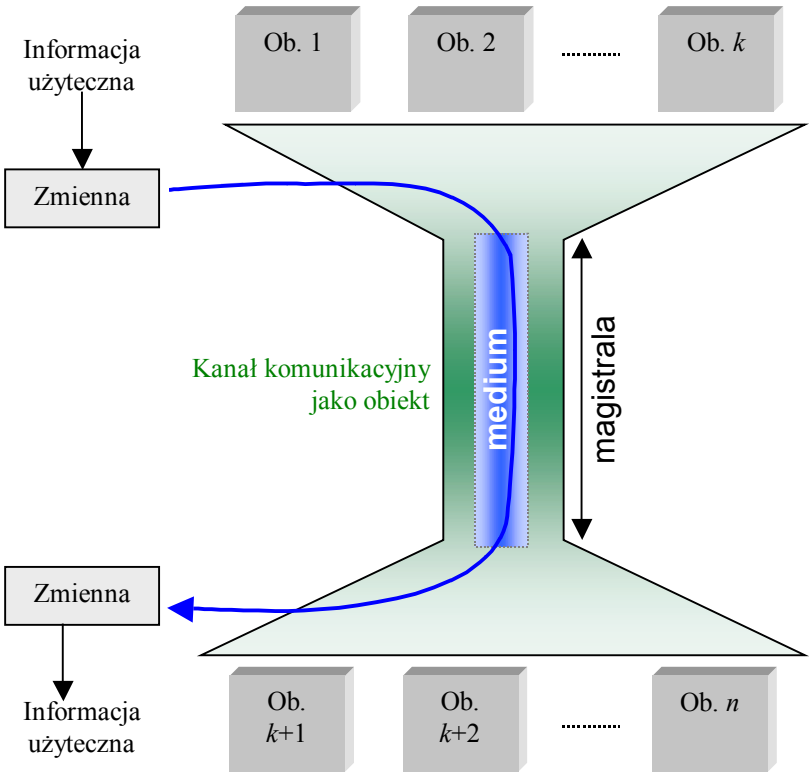
Każdy z elementów przedstawionych na rysunku 7 może posiadać cechę determinizmu działania oraz determinizmu czasowego. Traktując kanał komunikacyjny jako dwukierunkowy otrzymano trzy rodzaje połączeń stosowanych w informatycznych systemach przemysłowych:

A) obiekt deterministyczny ↔ kanał deterministyczny ↔ obiekt deterministyczny,
B) obiekt deterministyczny ↔ kanał niedeterministyczny ↔ obiekt niedeterministyczny,
C) obiekt niedeterministyczny ↔ kanał niedeterministyczny ↔ obiekt niedeterministyczny,

Proponowany podział bazuje na założeniu, że o ile chociaż jeden z obiektów biorących udział w dwukierunkowym przesyłaniu informacji użytecznej jest niedeterministyczny wówczas stosowanie kanału deterministycznego nie jest potrzebne. Trzeci rodzaj konfiguracji elementów (punkt C) nie może obsługiwać procesów wymagających deterministycznej reakcji na zdarzenia, gdyż żaden z obiektów nie reaguje w sposób deterministyczny.

Zdeterminowany obiekt przemysłowy reaguje na zdarzenia w sposób jednoznaczny, czyli jeżeli zaistnieje określona kombinacja zdarzeń zewnętrznych, to obiekt wypracuje jedną i zawsze taką samą reakcję na te zdarzenia. Podobnie może działać kanał komunikacyjny. W obsłudze procesów przemysłowych wymagających zdeterminowanych reakcji,

deterministyczny charakter działania tych elementów jest koniecznością. Problem pojawia się podczas rozważania działania zdeterminowanego czasowo. Aby obiekt był zdeterminowany czasowo wystarczy, aby był w stanie przetworzyć informację wejściową i wypracować reakcję w skończonym czasie wynikającym z założeń. W przypadku kanału komunikacyjnego realizowalność kanału zdeterminowanego czasowo zależy od jego budowy oraz przyjętych reguł wymiany informacji. Aby cały układ elementów był zdeterminowany czasowo względem obsługi zdarzeń akcji i reakcji na poziomie procesu, każdy z jego elementów obsługujących informację użyteczną związaną z obsługą tych zdarzeń, musi być zdeterminowany czasowo. Specjalizowane systemy komunikacyjne do zastosowań w przemyśle tworzone są tak, aby spełnić ten wymóg. Ponieważ temat pracy dotyczy komunikacji między abonentami obiektowymi pracującymi w przemyśle, zatem w dalszych rozważaniach zajęto się problemami związanymi tylko z kanałem komunikacyjnym, przyjmując, że wykorzystuje się abonentów, z których przynajmniej jeden działa w sposób zdeterminowany czasowo, a w skład systemu wchodzą przynajmniej dwa. Wymiana informacji zawsze zachodzi pomiędzy dwoma abonentami lub pomiędzy jednym a grupą innych.



Rys. 8 Kanał komunikacyjny jako wspólny element przepływu informacji

Na rysunku 8 przedstawiono system informatyczny składający się z *n* obiektów połączonych wspólnym kanałem komunikacyjnym. Obiekty podzielono na trzy grupy, które można wiązać na przykład z grupą abonentów mających kontakt z procesem (1..k), grupę nie mającą kontaktu z procesem (k+1..n) i kanał jako obiekt komunikacyjny. Kanał komunikacyjny można traktować jako obiekt wejściowo-wyjściowy. Oczywiście zaproponowany podział jest czysto przykładowy, a rysunek ma zobrazować wspólny element,

jakim jest obiekt komunikacyjny, przez który musi przejść każda informacja, która ma zostać przesłana pomiędzy abonentami. Głównymi zadaniami warstwy komunikacyjnej systemu informatycznego obsługującego proces przemysłowy jest przesyłanie zmiennych pomiędzy warstwami aplikacji abonentów.

Wymogi, jakie postawiono przed sposobem realizacji tego zadania zależą od wymogów procesu względem danej zmiennej, i tak:

dla zmiennych niewymagających obsługi zdeterminowanej realizacja wymian powinna uwzględniać:

- zachowanie kolejności przesyłu kolejnych wartości zmiennych,

dla zmiennych wymagających obsługi zdeterminowanej:

- zachowanie gwarancji doręczenia zmiennej,
- zachowanie kolejności przesyłu kolejnych wartości zmiennych,

dla zmiennych wymagających obsługi zdeterminowanej czasowo:

- zachowanie gwarancji doręczenia zmiennej,
- zachowanie kolejności przesyłu kolejnych wartości zmiennych,
- zachowanie ścisłego determinizmu czasowego działania wymian cyklicznych,
- zachowanie granicznego determinizmu czasowego działania wymian aperiodycznych.

Wszystkie przypadki odnoszą się do przekazywania zmiennych od producenta do konsumenta lub grupy konsumentów.

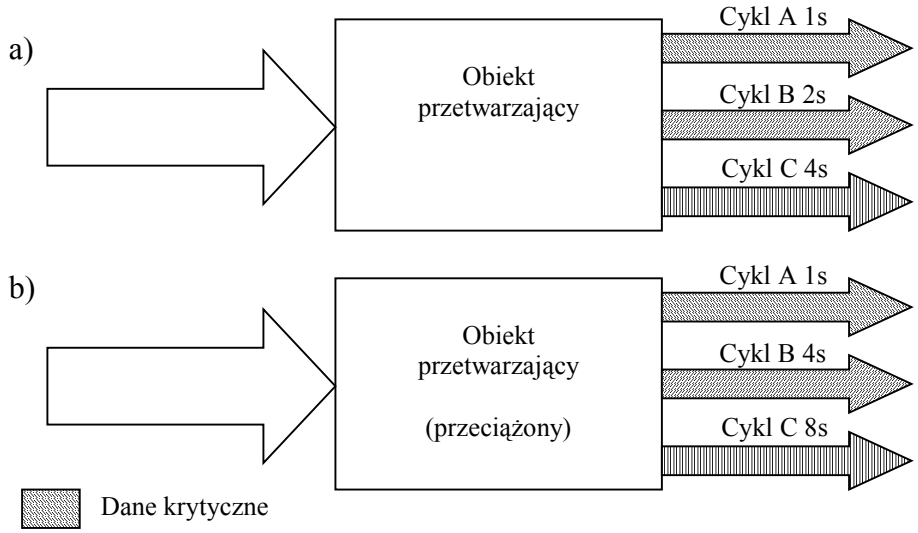
4.3. Czas rzeczywisty

Istnieje jeszcze jedno istotne pojęcie, które należy na wstępie przedstawić – pojęcie czasu rzeczywistego. Czas stanowi istotny parametr, który determinuje sposób przetwarzania danych w systemach przemysłowych. Czas określa maksymalny cykl pracy programu jednostek przetwarzających, sposób i możliwości reakcji systemu na zdarzenia, rozdzielczość rejestracji danych, sposoby prezentacji danych czy w końcu sam dostęp do danych procesowych. Czas jest parametrem, który w instalacjach przemysłowych określa przydatność danego rozwiązania względem potrzeb procesu i użytkownika.

W celu wyjaśnienia tego pojęcia, dla dalszych rozważań zdefiniowano, iż systemy czasu rzeczywistego są to systemy, które na bieżąco pracują w interakcji z elementami nie stanowiącymi składowych tych systemów. Rozumiejąc pojęcie czasu rzeczywistego jako brak zwłoki czasowej pomiędzy akcją a reakcją, nie istnieją systemy pracujące w tak rozumianym idealnym czasie rzeczywistym. Fizyka zawsze wprowadza opóźnienie pomiędzy wystąpieniem danego zdarzenia a jego obsługą, czyli reakcją systemu na nie. Zatem czas jako parametr występujący w przepływie informacji w systemach przemysłowych również nie będzie zerowy. W dalszej części pracy określono jak będzie się on kształtował i oddziaływał na pracę systemu.

Procesy przemysłowe wymagają systemów pracujących w czasie rzeczywistym. Wynika to z omówionego wcześniej determinizmu. Zatem, jeżeli dany system ma pracować jako system czasu rzeczywistego, to należy określić, na jakie zwłoki czasowe można dla danych

grup zmiennych pozwolić i jakie algorytmy reakcji przyjąć w sytuacjach, gdy system nie może dotrzymać zadanych parametrów. Powinny istnieć granice optymistyczne i pesymistyczne na zrealizowanie przepływu typu akcja – reakcja. W systemach przemysłowych granice te powinny być stałe dla danego zdarzenia w systemie lub danego systemu. Pojęcie czasu rzeczywistego jest nadrzędne względem zdefiniowanych wcześniej pojęć determinizmu. Na system czasu rzeczywistego mogą składać się obiekty obsługujące zdarzenia w sposób ściśle zdeterminowany czasowo jak również zdeterminowany granicznie. System czasu rzeczywistego może mieć również zdeterminowaną czasowo obsługę części zdarzeń. Zdarzenia te w dalszej części pracy nazwano zdarzeniami krytycznymi. Obsługa pozostałych zdarzeń w takich systemach może nie mieć charakteru zdeterminowanego czasowo. Na rysunku 9 przedstawiono przykład obiektu, który przetwarza strumień danych wejściowych w czasie rzeczywistym. Obiekt przetwarzający (przypadek a) realizuje cykliczne transmisje zmiennych z określonym okresem. Dla niniejszego przykładu założono, iż cykl A obsługuje zmienne krytyczne, natomiast cykle B i C pozostałe zmienne. Gdy zaistnieje sytuacja gdzie obiekt nie jest w stanie dotrzymać ograniczeń czasowych na wyjściu (przypadek b), następuje prioretyzacja danych i zmiana okresów wytwarzania danych niekrytycznych B i C.



Rys. 9 Prioretyzacja przetwarzania danych

W odniesieniu do zastosowań sieciowych przedstawiony obiekt przetwarzający można odnieść do obiektu komunikacyjnego na bazie protokołu TCP/IP oraz Ethernetu. Funkcjonujący w tym protokole mechanizm priorytetów jest w stanie regulować strumień danych wyjściowych o ile strumień wejściowy zmienia się w ustalonym zakresie.

W literaturze [61, 9, 26, 72, 76, 118] często spotyka się podział systemów czasu rzeczywistego na tzw.:

- *Soft Real – Time Systems*,
- *Hard Real – Time Systems.*

Systemy typu *soft* pracują realizując swoje funkcje w czasie rzeczywistym, lecz nie gwarantują dotrzymania narzuconych ograniczeń czasowych. Jako przykłady przytacza się systemy bankowe czy rezerwacji biletów [60]. Określanie takich systemów mianem systemów czasu rzeczywistego w myśl przytoczonej wcześniej definicji jest zabiegiem czysto marketingowym. Wynika to z braku zarówno ostrej jak i nieostrej formy determinizmu, a co za tym idzie braku możliwości pracy w czasie rzeczywistym. Bardziej uzasadnionym byłoby traktowanie jako systemy *soft* systemów priorytetyzujących dane przedstawionych na rysunku 9. W systemach tych mamy stale do czynienia z pracą w czasie rzeczywistym, jedynie zmieniają się wartości graniczne czasu definiujące deterministyczne zachowanie obiektu w czasie. Jednak i wówczas, dane nie będące krytycznymi zostaną obsłużone w czasie skończonym, a dla danych warunków w czasie z góry określonym. Działanie takich systemów będzie jednoznaczne w skończonym czasie, czyli również jest to przypadek systemu czasu rzeczywistego. Wspomniany podział nie będzie dalej stosowany w określaniu proponowanych rozwiązań.

4.4. Wymagania i ograniczenia

Główny problem, jaki pojawia się w pracy, to czy i w jaki sposób można zastosować zestaw protokołów intersieci TCP/IP w systemach przemysłowych, oraz czy jest to bezpieczne z punktu widzenia prowadzenia kontroli procesu. Wykorzystanie tego zestawu, choćby dla ograniczonego zakresu możliwości funkcjonalnych systemów kontrolno-nadzorczych, lub wykorzystanie tylko niektórych jego warstw, dałoby nowe, interesujące możliwości dla kontroli procesów przemysłowych. Dlatego właśnie wydaje się istotne rozważenie tematu niniejszej pracy, a wnioski z niej wypływające staną się przydatne dla projektantów przemysłowych systemów kontrolno-nadzorczych.

Z sieciami przemysłowymi zawsze kojarzył się tzw. determinizm czasowy dostępu do informacji. W sferze aplikacyjnej można było to obserwować dość intensywnie właśnie w ostatnim pięcioleciu. Rozwiązania, których warstwa transportu informacji nie była czasowo zdeterminowana uważano za nieprofesjonalne i niepewne. Ostatnio jednak, ze względu na duży rozwój technologii konstrukcji sprzętu, rozwinęły się sieci lokalne i protokoły nie mające charakteru determinizmu czasowego transmisji, a ze względu na inne cechy, mogące konkurować z sieciami przemysłowymi. Obecnie wśród konstruktorów, projektantów oraz aplikantów systemów przemysłowych pojawiła się pewna konsternacja. Czy wykorzystanie protokołów niezdeterminowanych czasowo jest bezpieczne? A może istnieje możliwość wykorzystywania ich w sposób gwarantujący determinizm lub ograniczenia wynikające z jego braku są pomijalnie małe? Wszystko zależy od tego, na jakim poziomie transmisji danych występuje wykorzystanie oraz dla jakiej klasy zastosowań. Dlatego w pracy pokazano, jakie są wymagania obiektów przemysłowych względem warstwy komunikacyjnej oraz jakie ograniczenia wprowadza warstwa oparta na TCP/IP i Ethernecie względem obsługiwanych obiektów.

4.5. Determinizm czasowy w sieciach przemysłowych

Podstawową cechą, jaką powinna się charakteryzować komputerowa sieć przemysłowa jest determinizm czasowy przekazywania danych. Cechę taką posiadają tylko te sieci, w których czas obsługi pakietu w węźle sieciowym jest skończony i określony w sposób ścisły lub w przedziale, a co za tym idzie czas dostępu do danych jest również skończony i określony ściśle lub w przedziale [32]. Cecha determinizmu nie wymaga, aby przekazywanie między węzłami było niezawodne. Istotny jest tylko fakt, aby przy poprawnej pracy sieci czas dostępu do informacji był obarczony krytycznym czasem granicznym. Przez poprawną pracę sieci rozumie się pracę, podczas której nie występują stany uznawane przez definicję sieci jako awaryjne, zarówno ze strony fizycznej jak i logicznej. Istnienie mechanizmu gwarantującego realizację zdeterminowanych w czasie wymian jest podstawą do określenia pracy protokołu w czasie rzeczywistym.

Istnieją trzy modele opisujące sposób wymiany danych [60, 61] pomiędzy węzłami na bazie których można zbudować sieci umożliwiające obsługę transmisji informacji z wykorzystaniem tej cechy. Są to:

- Master-Slave,
- Token,
- PDC.

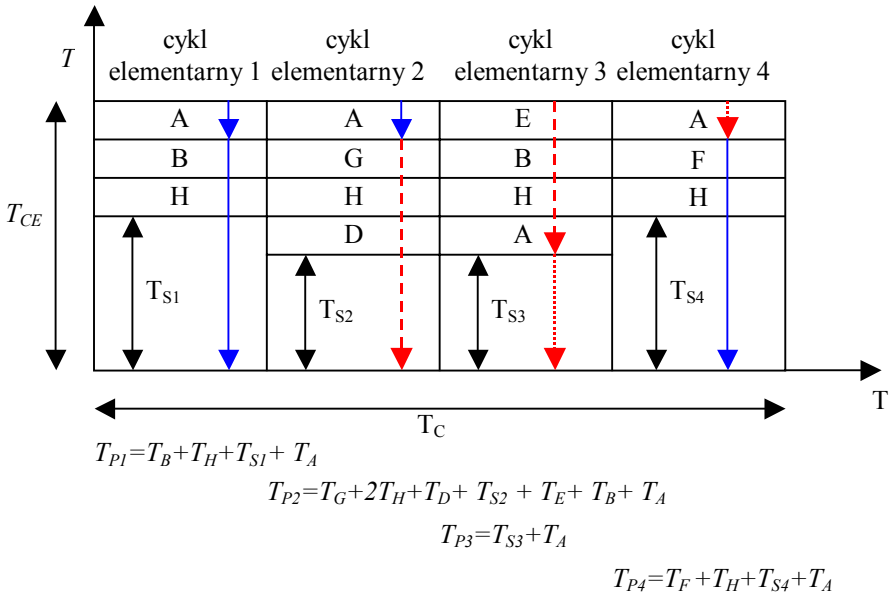
Wszystkie sieci spełniające wymóg determinizmu czasowego bazują na jednym z powyższych modeli, stanowią ich hybrydę lub uproszczenie. Poniżej skrótowo scharakteryzowano te modele.

Model Master – Slave bazuje na przesyłaniu informacji pomiędzy dwoma rodzajami stacji. Stacją Master i stacjami Slave. Stacja Master jest abonentem zarządzającym ruchem w sieci. Przechowuje on scenariusz wymian i według niego realizuje transmisję danych do abonentów Slave. Wszystkie transakcje są inicjowane przez abonenta Master i wszystkie dane użyteczne przesyłane od abonenta do abonenta przez nią przechodzą, na zasadzie redystrybucji.

Model Token opiera się na przesyłaniu informacji pomiędzy równorzędnymi stacjami, z których okresowo każda staje się uprzywilejowana, przez pozyskanie żetonu. Żeton jest specjalnym rodzajem informacji, która krąży w sieci o jednego abonenta do drugiego. Stacja, która odczyta żeton ma prawo przez określony czas realizować zapisy. Jednak po upływie tego czasu musi bezwzględnie przekazać żeton do następnej stacji.

Model PDC bazuje na przesyłaniu informacji pomiędzy trzema rodzajami abonentów. Abonenci dzielą się na producentów, konsumentów i dystrybutorów informacji. Dystrybutor przechowuje scenariusz wymian i według niego określa, kiedy jaka informacji ma być obsłużona przez sieć. Dystrybutor w przeciwieństwie do stacji Master nie inicjuje transmisji danych użytecznych ani ich nie retransmituje. Określa jedynie, kiedy producent danej informacji ma zrealizować zapis rozgłoszeniowy w sieci.

Pojawia się pytanie czy rzeczywiście w sieciach przemysłowych spełniony jest w pełni warunek transmisji danych w sposób zdeterminowany czasowo. Biorąc pod uwagę rozwiązania modelowe, czyli oparte na wspomnianych wyżej modelach kontroli wymian, zachwiania determinizmu są pomijalnie małe i wynikają przeważnie z niejednoznaczności czasu realizacji cyklu scenariusza wymian. Oczywiście pomijane są tu przypadki awarii, które wystąpić mogą zawsze i nie zależą od konstrukcji interfejsów czy protokołów. Niejednoznaczność cyklu wymian ma znaczenie jedynie tam, gdzie ma być on określony w sposób ostry. Przykładem mogą być scenariusze wymian cyklicznych w modelu PDC lub cykl zapytań i odpowiedzi w modelu Master-Slave. Przedstawiono to na rysunku 10 dla cyklu scenariusza wymian składającego się z czterech cykli elementarnych,



Rys. 10 Niejednoznaczność cyklu wymian w modelu PDC

gdzie:

A, B, D, E, F, H – nazwy symboliczne wymienianych zmiennych.

T_{CE} – oznacza czas cyklu elementarnego,

T_C – oznacza czas cyklu scenariusza wymian ($4T_{CE}$),

$T_{A..H}$ – oznacza czas trwania transakcji danej zmiennej,

T_{Pn} – oznacza czas okresu pomiędzy kolejnymi wymianami zmiennej A,

T_S – oznacza czas cyklu pozostały na wymiany aperiodyczne.

Dla tak skonstruowanego scenariusza wymian, i przy założeniu, iż czas trwania transakcji jest stały:

$$T_A = T_B = T_D = T_E = T_F = T_H = const, \quad (4)$$

czasy pomiędzy kolejnymi wymianami zmiennej A nie są jednakowe. Ostry determinizm zachodzi z dokładnością wynikającą z różnicy pomiędzy największą a najmniejszą wartością okresu.

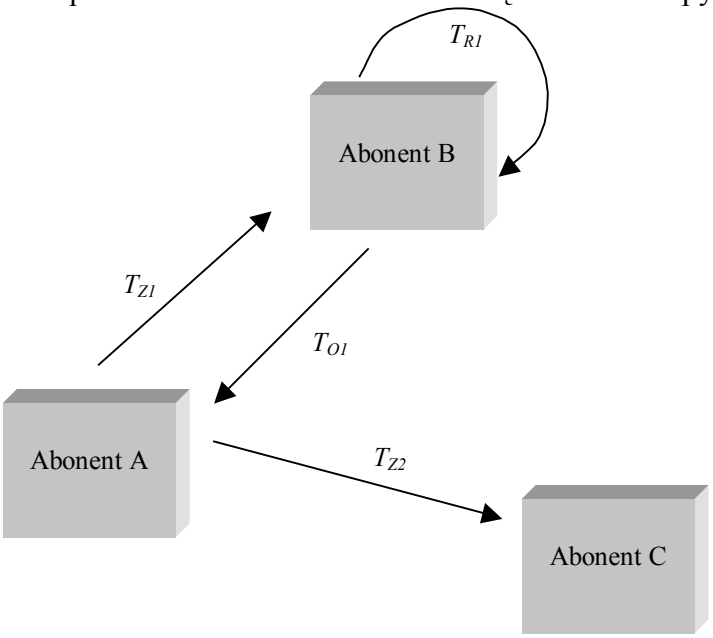
$$T_{PMIN} \le T_P \le T_{PMAX} \quad (5)$$

Czyli dla przedstawionego przykładu

$$T_{S2} + T_A \leq T_P \leq T_{S2} + 7T_A \quad (6)$$

Zatem dla przykładu, niejednoznaczność cyklu może sięgać do sześciokrotnej wartości czasu transmisji pojedynczej zmiennej.

Podobny problem można zaobserwować przy cyklicznym odpytywaniu abonentów. Przedstawiono to na rysunku 11 na przykładzie zapytań kierowanych do dwóch abonentów B, C) przez abonenta A. Odpowiedzi abonentów kierowane są do abonenta pytającego (A).



Rys. 11 Niejednoznaczność cyklu przy odpytywaniu abonentów

T_{Z1}, T_{Z2} – czasy realizacji transakcji zapytań do abonenta B i C,
T_{R1} – czasy realizacji przygotowania odpowiedzi abonenta B,
T_{O1} – czasy realizacji transakcji odpowiedzi abonenta B,

Zapytania realizowane są cyklicznie z okresem T_P, który powinien być stały w przypadku ostrego determinizmu. Niestety w praktyce tak nie jest. Wynika to z faktu, iż:

$$T_{T1} = T_{Z1} + T_{R1} + T_{O1}, \quad (7)$$

gdzie T_{T1} stanowi wartość czasu realizacji kompletnej transakcji zapytania i odpowiedzi pomiędzy abonentem A i B. Wartości T_{Z1}, T_{R1}, T_{O1} nie są stałe dla poszczególnych wymian [5, 61]. Ich wartość zależy od długości ramki, czasu obliczeń CRC, czasu synchronizacji aplikacji abonenta B i wielu innych mniej znaczących czynników. Zatem $T_{T1} \neq const$, a co za tym idzie T_P zależne od T_T może zmieniać swą wartość w każdym cyklu odpytywania. Opisany problem występuje w każdym rodzaju sieci.

Z powyższych rozważań wynika, iż sieci przemysłowe, w których stosowane protokoły z założenia gwarantować mają determinizm czasowy transmisji, gwarantują parametry czasowe z pewną niejednoznacznością. Analizowanie ostrego determinizmu w sieciach komputerowych jest bezcelowe, gdyż żadna sieć i żaden protokół takiego determinizmu nie zapewni. W dalszych rozważaniach skoncentrowano się tylko i wyłącznie na przypadkach sieci, w których protokół zapewnia co najwyżej determinizm czasowy graniczny. Brano zatem pod uwagę transakcje z punktu widzenia najgorszego przypadku, gdzie:

$$T_{PMIN} \leq T_P \leq T_{PMAX} \quad (8)$$

gdzie T_{PMIN}, T_{PMAX} są czasami minimalnym i maksymalnym okresu.

Jeżeli dla q wymian sieciowych istnieje takie i gdzie:

$$\underset{0<i\leq q}{T_{Pi}} = \infty, \quad (9)$$

wówczas zaistnieje przypadek gdzie:

$$T_{PMAX} = \infty, \quad (10)$$

czyli przypadek skrajny, w którym może zaistnieć utrata pakietu. Ogólnie dla sieci niedeterministycznej, na q wymian sieciowych, zaistnienie przypadku utraty pakietu jest możliwe z danym prawdopodobieństwem.

W praktyce nie każdy informatyczny system przemysłowy wymaga obsługi przez deterministyczną warstwę komunikacji. Bardzo często wystarcza określenie czasu T_P bazując właśnie na zapewnieniu przedziału niejednoznaczności na poziomie danego prawdopodobieństwa. W sieciach takich utraci się stabilność wartości okresu dla wymian cyklicznych, lecz można uzyskać zwiększenie przepustowości. Zjawisko takie można zaobserwować zarówno dla sieci niedeterministycznych jak i dla specjalizowanych sieci przemysłowych, chociażby analizując obciążenie sieci Profibus z występowaniem ograniczeń czasowych oraz bez ograniczeń czasowych [66].

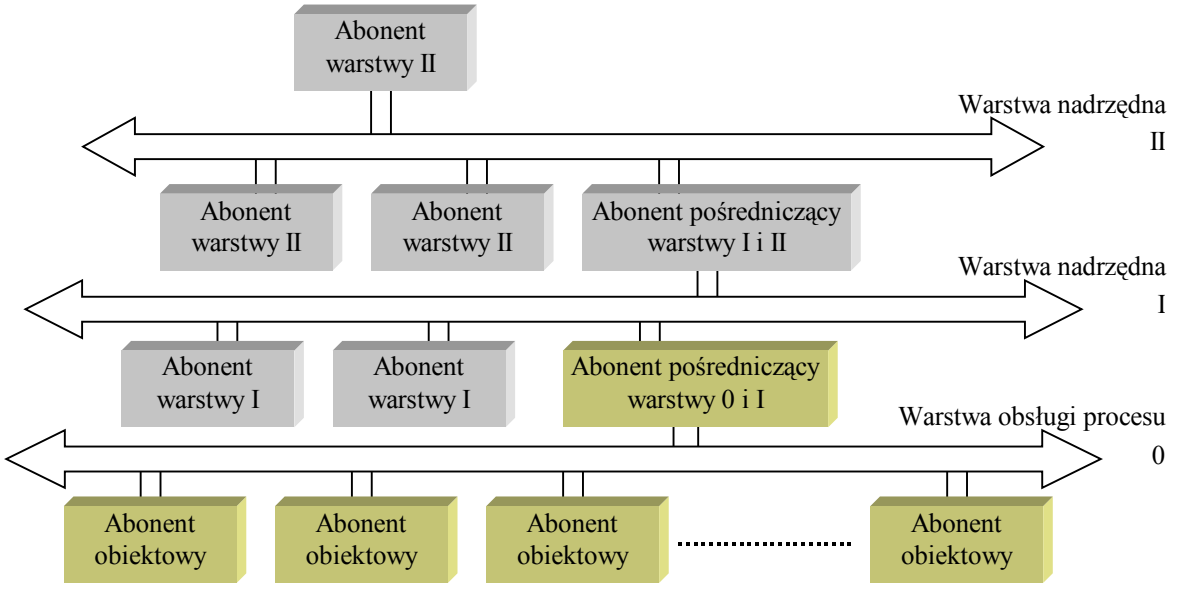
Sieć oparta o Ethernet i TCP/IP nie musi wprowadzać $T_{PMAX} = \infty$. Zależy to od konstrukcji warstw aplikacji, co w dalszej części pracy będzie wykazane.

5. Sieci komputerowe w zastosowaniach przemysłowych

Problem, którego dotyczy temat pracy, pojawia się przed projektantem mechanizmu mającego za zadanie przedstawić człowiekowi to, co dzieje się w rzeczywistym procesie przemysłowym, oraz umożliwić mu interakcję z tym procesem za pomocą infrastruktury informatycznej. Kontrola taka, aby spełniała swoje podstawowe zadanie musi być odpowiednio skonstruowana. System kontrolno-nadzorczy nie stanowi tylko i wyłącznie konstrukcji zbudowanej ze sprzętu i oprogramowania, będącego dla użytkownika interfejsem do postrzegania i ewentualnej interakcji. Komputer czy tablica synoptyczna stanowią jedynie wykonawcze urządzenia interfejsowe dla całości mechanizmu umożliwiającego kontrolę procesu.

System kontroli procesu musi zapewnić trzy funkcje. Po pierwsze pobrać informację z procesu, po drugie przetworzyć i dostarczyć ją do innych obiektów systemu, oraz po trzecie dostarczyć informację do procesu i wykonać prezentację informacji na sprzęcie stanowiącym interfejs pomiędzy systemem a użytkownikiem.

Konstrukcja współczesnych wielowarstwowych systemów kontrolnych bazuje na schemacie, jaki przedstawiono na rysunku 12 [21].

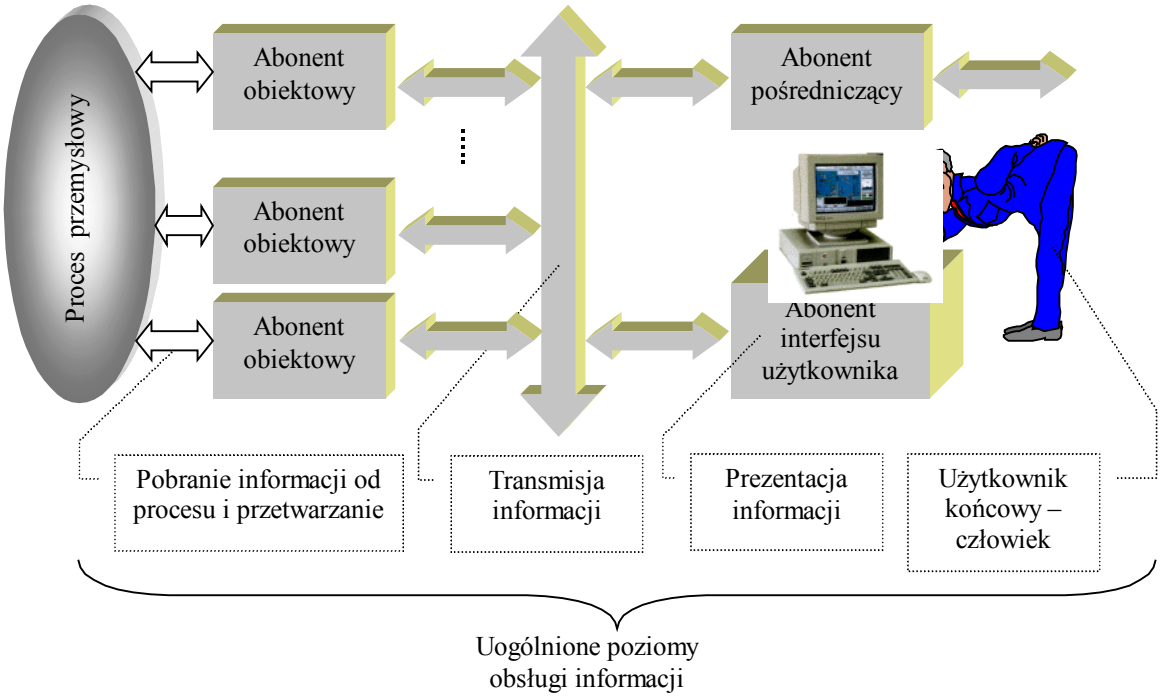


Rys. 12 Uogólniony schemat hierarchicznej struktury systemu kontrolnego

W przedstawionej strukturze zachodzą dwojakiego rodzaju wymiany informacji. Pierwszym rodzajem są wymiany pomiędzy abonentami na danej warstwie. W literaturze [61, 21] wymiany takie nazywa się wymianami poziomymi. Drugi rodzaj stanowią wymiany pomiędzy abonentami danej warstwy a warstwami nadrzędnymi. Są to wymiany pionowe.

Z punktu widzenia tematu pracy istotna będzie warstwa obsługi procesu wraz z podłączeniem jej do warstw nadrzędnych.

Konstrukcję warstwy obsługi procesu przedstawiono w sposób bardziej szczegółowy na rysunku 13.



Rys. 13 Uogólniona konstrukcja systemu kontrolnego

Stacje nadrzędne w postaci abonentów pośredniczących lub abonentów interfejsoiwych często są implementowane jako ten sam abonent spełniający funkcje pośrednictwa międzywarstwowego (rys. 12) oraz interfejsu użytkownika. Proces wymiany informacji w takim systemie odbywa się na płaszczyznach:

1. abonent obiektowy → abonent obiektowy – sterowanie procesem, wymiana danych, synchronizacja procesu itp.,
2. abonent obiektowy → stacja nadrzędna – prezentacja, rejestracja danych, synchronizacja procesu, przekazywanie danych do warstw nadrzędnych.
3. stacja nadrzędna → abonent obiektowy – polecenia i rozkazy, synchronizacja procesu.

Pierwsza płaszczyzna jest związana tylko z wymianami poziomymi, natomiast w przypadku pozostałych dwóch, gdzie pojawia się stacja nadzorcza, z wymianami pionowymi [60, 61, 23].

Protokół TCP/IP proponuje się związać z dowolną płaszczyzną wymian. Kanał komunikacyjny dla działania całości systemu stanowi element newralgiczny i od tego jak jest zbudowany, a w szczególności jakie protokoły, czyli jakie języki wykorzystywane są do porozumiewania, zależy praca wszystkich pozostałych elementów systemu. W dalszej części pracy, możliwość stosowania stosu protokołów TCP/IP przedstawiono jako część wspólną

teoretycznych możliwości wynikających z funkcjonalności stosu i listy wymagań pochodzących od pozostałych elementów systemu, w odniesieniu do różnych elementów zestawu protokołów TCP/IP.

Biorąc pod uwagę poziom transmisji informacji, rozważano tylko taki przypadek, w którym poziom ten bazuje na wykorzystaniu sieci komputerowych. Przypadek łącza nie mającego charakteru sieci jest dla tematu pracy nieprzydatny, gdyż wykorzystanie protokołu TCP/IP jest wówczas bezcelowe.

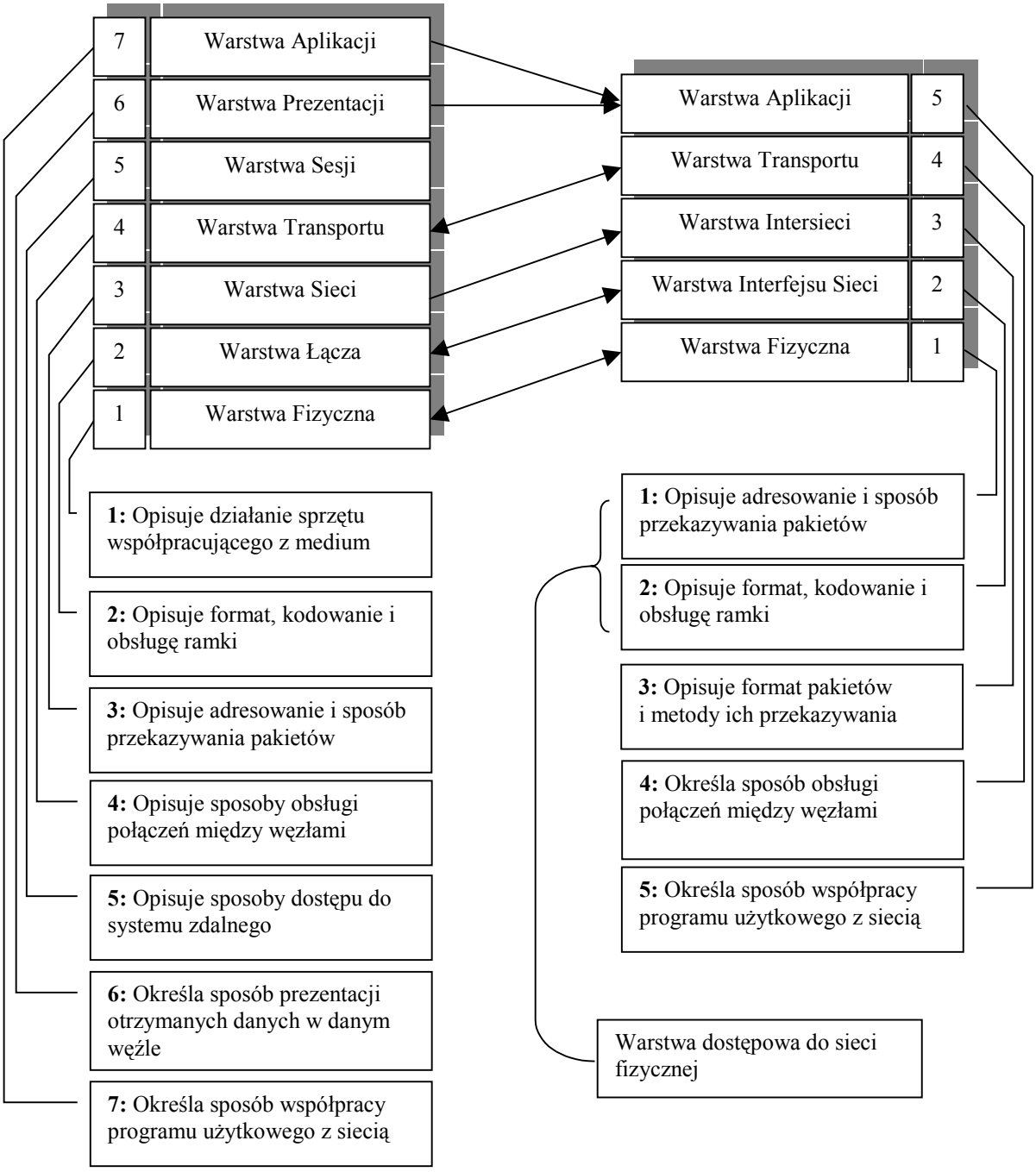
Ogólnie, sieć komputerowa zbudowana jest z warstwy fizycznej, która realizuje transmisję rzeczywistych sygnałów fizycznych po rzeczywistym medium oraz z warstw przetwarzania, czyli sposobów reprezentacji i przetwarzania tych sygnałów przy użyciu tzw. protokołu. Sygnały, które pojawiają się na medium stanowią kwintesencję pracy wszystkich warstw protokołu. Reprezentują one nie tylko informację użyteczną, ale również informację związaną z funkcjonowaniem protokołu na poszczególnych jego warstwach. W literaturze [13, 32, 45, 96] istnieją różne metody opisania i projektowania mechanizmów realizujących zadanie komunikacji. Jedna z najważniejszych to model podziału na warstwy. Istnieją przynajmniej dwa istotne modele opisujące sieci komputerowe. Są to:

- siedmiowarstwowy model wzorcowy ISO – jest to już historyczny, ponad dwudziestoletni model podziału warstwy przetwarzającej interfejsu sieciowego na siedem warstw protokołów. Mimo swej „długowieczności" oraz faktu, iż wiele nowszych protokołów do niego nie pasuje (choćby TCP/IP), często stanowi podstawę rozważań o interfejsach komunikacyjnych. W niniejszej pracy również często pojawiają się odwołania do warstw definiowanych przez ten model.
- warstwowy model intersieci – jest to nowszy model składający się z pięciu warstw częściowo pokrywających się z modelem ISO. Model ten pojawił się wraz ze stworzeniem pierwszych intersieci opartych na bazie protokołów TCP/IP.

Model ISO mimo swej uniwersalności i przejrzystości nie definiuje mechanizmów działania protokołu w przypadku pracy w środowisku intersieciowym. Dlatego też model intersieciowy posiada zdefiniowaną dodatkowo warstwę intersieci określającą format i metody przesyłania pakietów pomiędzy sieciami. Natomiast pomija warstwę sesji jako nieprzydatną w związku z brakiem zagadnień wielodostępu w intersieciach. Wzajemne powiązania pomiędzy modelami przedstawiono na rysunku 14.

Nie istnieje bezpośrednie odzwierciedlenie warstwowych modeli interfejsów sieciowych i wspomnianych wcześniej modeli wymiany danych w sieci (rozdział 4.5), gdyż opisują zupełnie co innego.

W dalszej części pracy, przed dokonaniem analizy problemów, jakie pojawiają się w intersieciach, przeanalizowano problemy występujące na wydzielonych segmentach sieci lokalnych danego typu. Dla tego typu przypadku mają zastosowanie tylko niektóre z warstw przedstawionych modeli.



Rys. 14 Modele warstwowe ISO i intersieci i ich wzajemna odpowiedniość

5.1. Protokół TCP/IP w sieciach komputerowych

Tworzeniu protokołu TCP/IP przyświecały raczej cele wojskowe niż przemysł. Dlatego bezpośrednie wykorzystanie tego protokołu w zagadnieniach informatycznych bazujących na pojęciu czasu rzeczywistego nie jest oczywiste i wymaga zastanowienia.

Zestaw protokołów TCP/IP został opracowany w celu umożliwienia komunikacji pomiędzy systemami komputerowymi różnego typu i podłączonych do różnego rodzaju sieci fizycznych. Celem dodatkowym było zaoferowanie jednolitych usług sieciowych w niejednolitych środowiskach sprzętowo programowych. Dzięki protokołom TCP/IP zaistniała możliwość tworzenia sieci wirtualnych, w których występują odwołania do

abonenta przez jego unikalny adres abstrahując od rodzaju sprzętu, oprogramowania i sieci fizycznej. Dzięki temu połączenie abonentów ogólnoświatową siecią komputerową stało się kwestią czasu.

Rodzina protokołów TCP/IP składa się między innymi z następujących protokołów [13, 14, 132]:

protokoły warstwy dostępowej:

- PPP – Point-to-Point Protocol – Protokół przeznaczony do transmisji szeregowej synchronicznej i asynchronicznej po łączach dzierżawionych i komutowanych z możliwością adresacji IP.
- SLIP – Serial Line IP – Protokół przeznaczony do transmisji szeregowej synchronicznej i asynchronicznej po łączach dzierżawionych i komutowanych bez możliwości adresacji.
- ARP/RARP – Address Resolution Protocol/Reverse Address – konwersja adresów – powiązanie adresu IP z adresami fizycznymi sieci.
- L2TP – Layer 2 Tunneling Protocol – Protokół tunelowania dla warstwy drugiej używany do integracji multiprotokołowych usług dial-up z POP i ISP

protokoły warstwy intersieci:

- IP, IPv6 – Internet Protocol – podstawowy protokół Internetu
- ICMP, ICMPv6 – Internet Control Message Protocol – protokół kontroli poprawności działania sieci IP
- DHCP – Dynamic Host Configuration Protocol – protokół umożliwiający dynamiczną konfigurację hostów.
- IGMP – Internet Group Management Protocol – protokół zarządzania grupowego.
- MARS – Multicast Address Resolution Server – definicja grup i mechanizm dystrybucji pakietów point – multipoint
- PIM – Protocol Independent Multicast-Sparse Mode (PIM-SM) – protokół efektywnego rutowania przy obsłudze grup
- RIP2 – Routing Information Protocol – protokół rutowania tras.
- RIPng – Protokół rutujący dla IPv6
- RSVP – Resource ReSerVation setup Protocol – używany przez usługę QoS.
- IPSec – Protokół bezpieczeństwa transmisji warstwy intersieci.

protokoły warstwy transportowej:

- TCP – Transmission Control Protocol – Protokół połączeniowy umożliwiający ustanowienie i utrzymanie połączenia wirtualnego pomiędzy dwoma użytkownikami sieci. Służy do przesyłania danych, sterowania przepływem, przesyłania potwierdzeń oraz kontroli i korekcji błędów.
- UDP – User Datagram Protocol – Protokół bezpołączeniowy nie posiadający mechanizmów kontroli przesyłu. Służy do bezpośredniego korzystania z usług protokołu IP.

– Van Jacobson – skompresowana wersja TCP.
– XOT – X.25 over TCP – protokół tunelowania X.25 przez TCP.
– SSL – Secure Socket Layer – protokół bezpieczeństwa transmisji warstwy transportowej.
– Mobile IP – Protokół umożliwia węzłom sieci przesyłanie pakietów z jednej podsieci IP do innej.

protokoły warstwy aplikacyjnej bazujące na przesyłach TCP:

– TELNET – protokół zdalnej sesji dla usług terminalowych. Pozwala na rozpoczęcie zdalnej sesji poprzez sieć.
– TFTP – Trivial File Transfer Protocol – Protokół prostych transferów plików: zapis i odczyt bez usług katalogowych i autoryzacji.
– FTP – File Transfer Protocol – Protokół interakcyjnego transferu plików.
– SMTP – Simple Mail Transfer Protocol – Protokół poczty elektronicznej.

protokoły warstwy aplikacyjnej bazujące na przesyłach UDP:

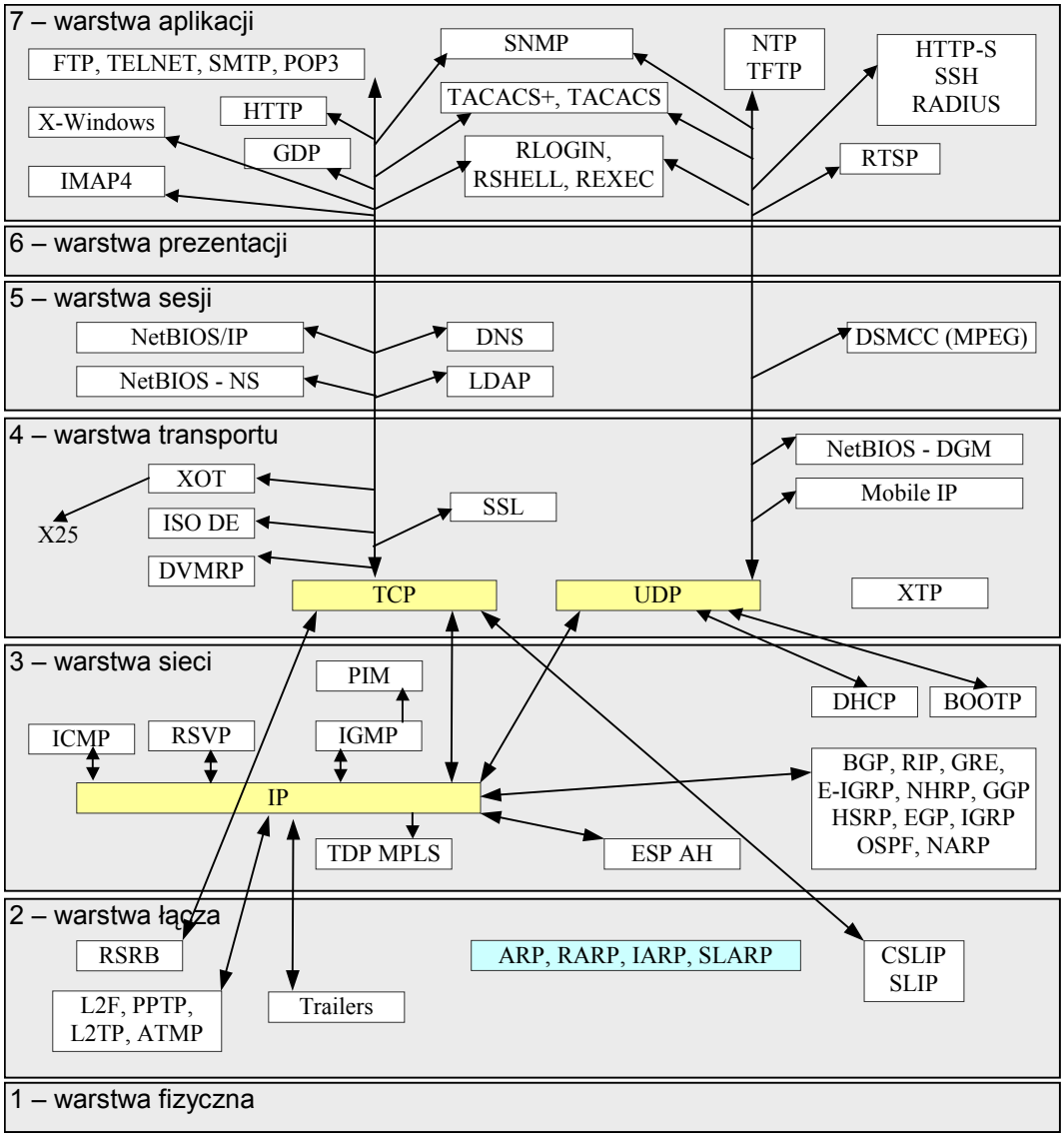
– DNS – Domain Name Service – Protokół zmiany adresów IP na nazwy symboliczne.
– RIP – Routing Information Protocol – Protokół reguł doboru tras.
– OSPF – Open Shortest Path First – Protokół reguł doboru tras.
– EGP – Exterior Gateway Protocol – Protokół reguł doboru tras.
– BGP – Border Gateway Protocol – Protokół reguł doboru tras.
– NFS – Network File System – Protokół umożliwiający współdzielenie plików przez różne komputery podłączone do sieci.
– HTTP – Hyper Text Transfer Protocol – Protokół dedykowany do szybkiej dystrybucji dokumentów hypertekstowych.
– S-HTTP – Secure Hypertext Transfer Protocol. Protokół dedykowany do szybkiej dystrybucji dokumentów hypertekstowych z użyciem kryptograficznych mechanizmów bezpieczeństwa.
– SNMP – Simple Network Management Protocol – Protokół zarządzania siecią. Umożliwia różnym obiektom sieciowym na uczestniczenie w globalnej architekturze zarządzania siecią.

inne protokoły warstwy aplikacyjnej:

– COPS – Common Open Policy Service – Protokół opisuje proste zapytania i odpowiedzi które mogą być użyte do wymiany informacji zabezpieczającej pomiędzy serwerem zabezpieczeń (Policy Decision Point or PDP) i jego klientami.
– Finger – Protokół aplikacyjnej usługi pozyskiwania informacji.
– IMAP4 – Internet Message Access Protocol rev 4 – umożliwia klientom manipulację komunikatami poczty elektronicznej na serwerze.

- ISAKMP – Internet Message Access Protocol version 4rev1 – definiuje procedury I postać pakietów w celu nawiązania, modyfikacji, negocjacji i usuwania powiązań bezpieczeństwa (SA)
- NTP – Network Time Protocol – protokół synchronizacji zegarów komputerów przez Internet
- POP3 – Post Office Protocol version 3 – przeznaczony do umożliwiania stacjom roboczym uzyskania dynamicznego dostępu do skrzynek pocztowych na serwerze.
- Radius – Protokół obsługi rozproszonych grup połączeń szeregowych i modemowych dla dużej liczby użytkowników.
- RLOGIN – Remote Login – protokół zdalnego logowania (dla Unix).
- RTSP – Real-time Streaming Protocol – przeznaczony do obsługi strumieni audio i video.
- SLP – Service Location Protocol – dostarcza skalowalne narzędzie do znajdywania i używania usług sieciowych.
- SOCKS – Protokół dostarcza podstawy dla aplikacji typu klient-serwer do obsługi transferu TCP i UDP w celu dogodnego i bezpiecznego używania usług sieciowej ściany ogniowej (ang. *firewall*).
- TACACS+ – Protokół dostarcza mechanizmów kontroli dostępu do ruterów, sieciowych serwerów dostępu i innych mechanizmów sieciowo obsługiwanych urządzeń przez jedno lub więcej serwerów scentralizowanych.
- WCCP – Web Cache Coordination Protocol – Umożliwia ruterowi realizację transparentnych przekierowań ruchu w celu obsługi tzw. usługi web-cache.
- X-Window – Protokół dostarcza zdalnego okienkowego interfejsu dla rozproszonych aplikacji sieciowych.
- NetBIOS/IP – Protokół wspomaga usługi NetBIOS w środowisku protokołu TCP/IP.
- LDAP – Lightweight Directory Access Protocol – umożliwia dostęp do katalogów X.500 bez używania DAP (Directory Access Protocol).
- HSRP – Cisco Hot Standby Router Protocol – Protokół rutujący.
- IGRP – Interior Gateway Routing – Protokół rutujący.
- NARP – NBMA Address Resolution Protocol – Protokół rutujący.
- NHRP – Next Hop Resolution Protocol – Protokół rutujący.
- AH – IP Authentication Header – Protokół bezpieczeństwa dodający informacje umożliwiającą autentyfikację do datagramu IP
- ESP – IP Encapsulating Security Payload – Protokół bezpieczeństwa umożliwiający szyfrowanie danych.

Powyższe wyszczególnienie protokołów jest poglądowe i niekompletne. Istnieje szereg protokołów na każdej z warstw stosu TCP/IP, których zastosowanie przejawia się w specjalizowanych rozwiązaniach lub po prostu są one mniej popularne od wymienionych. Jak widać, zbiór jest ogromny i na jego temat można by napisać pokaźną książkę, która zresztą szybko by się zdezaktualizowała. W niniejszej pracy nie opisano działania ani nie odwołano się do wszystkich z wymienionych protokołów. Nie jest to celem pracy. Dla dalszych rozważań dobrze jednak mieć świadomość z jak złożonym i rozbudowanym stosem mamy do czynienia.

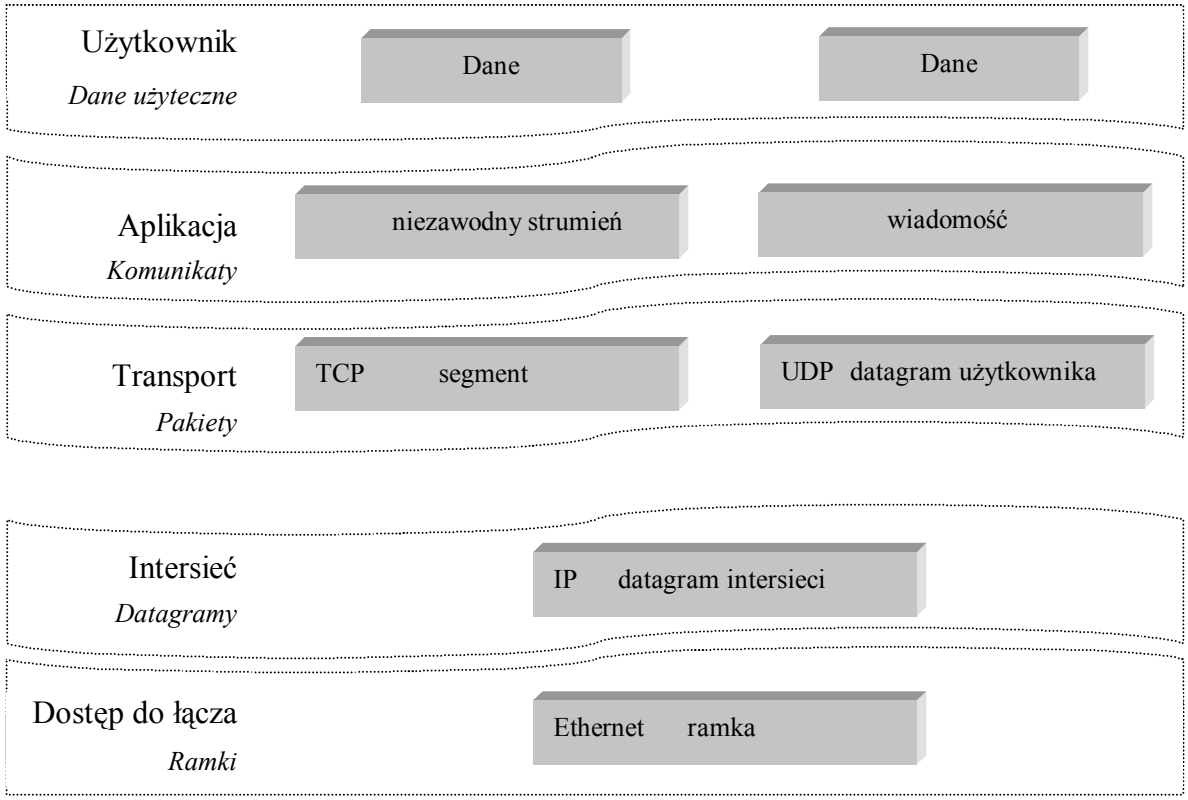


Rys. 15 Rozmieszczenie i współzależność przykładowych protokołów składowych rodziny TCP/IP

Aby naświetlić zadania i współpracę protokołów składowych TCP/IP, na rysunku 15 przedstawiono diagram współzależności dodatkowo wpisany w standardowy siedmiowarstwowy model OSI. Rozmieszczenie w poszczególnych warstwach jest tutaj nieco inne niż można by się spodziewać z rysunku 14. Przykładowe protokoły z warstw 7, 6 i 5 powinny być traktowane w modelu pięciowarstwowym jako protokoły aplikacyjne o specyficznych specjalizowanych funkcjach. Ta drobna niespójność wynikająca z trudności

w zaszeregowaniu funkcjonalnym protokołów pokazuje, iż model OSI nie nadaje się do opisu protokołów intersieciowych i w dalszej części częściej będą odwołania do przedstawionego wcześniej modelu czterowarstwowego[3].

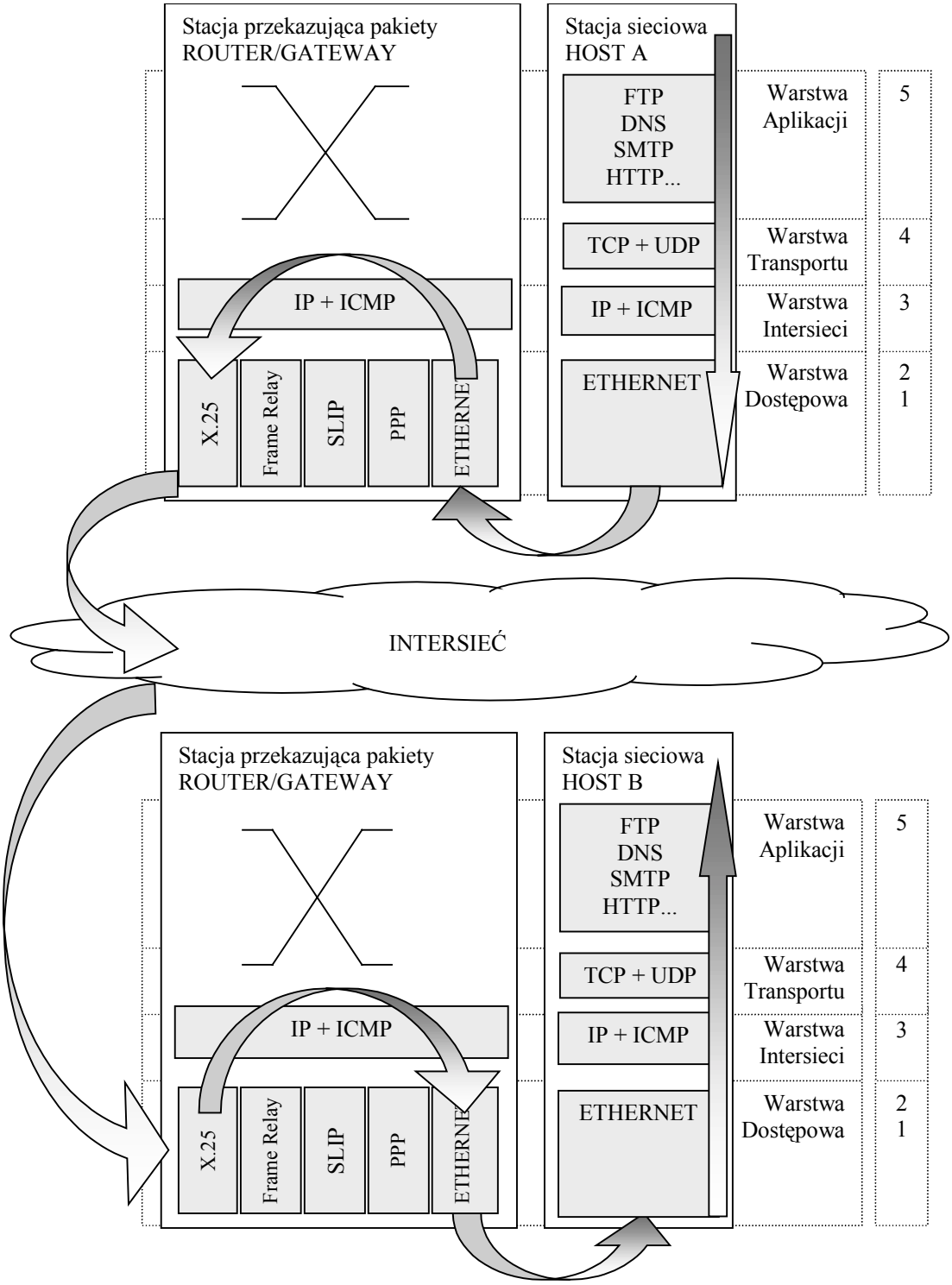
Na rysunku 16 przedstawiono schematyczną budowę i nazewnictwo abstrakcyjnych jednostek transmisji informacji wykorzystywanych przez najważniejsze z protokołów stosu TCP/IP. Jednostki te przedstawione są w powiązaniu z warstwami interfejsu komunikacyjnego co pokazuje w jaki sposób jednostki te współpracują ze sobą.



Rys. 16 Abstrakcyjne jednostki transmisji danych w TCP/IP

Na rysunku 17 schematycznie przedstawiono przekazywanie informacji pomiędzy dwoma stacjami w sieci opartej o protokół TCP/IP. W przykładzie przyjęto, iż stacje robocze uzyskują dostęp do routerów przy wykorzystaniu sieci lokalnych zbudowanych na bazie standardu Ethernet. Oczywiście w dostępie do sieci rozległych nie jest to regułą. W niniejszej pracy skoncentrowano się na wykorzystaniu sieci Ethernet, zarówno w pośrednictwie dostępu do sieci wirtualnej jak i w konstrukcji lokalnych segmentów sieci systemowych.

[3] Pięcio lub czterowarstwowego, gdyż dwie najniższe warstwy można połączyć w tzw. warstwę dostępową.



Rys. 17 Uproszczony przykładowy przepływ informacji w intersieci TCP/IP

5.2. Determinizm czasowy w sieciach TCP/IP

Protokoły TCP/IP nie definiują ani charakteru obsługiwanych zdarzeń, ani czasu obsługi w węźle, ani żadnego ze wspomnianych w rozdziale 4.5 modeli wymian. Same warstwy TCP/IP to za mało, aby spełnić wymogi determinizmu czasowego. Istnieje jeszcze jedno pojęcie, bez którego realizacja zdeterminowanych czasowo wymian, niezależnie od przyjętego modelu, byłaby niemożliwa. Jest to tzw. scenariusz wymian [61]. W zestawie

protokołów TCP/IP brak jest standardowych protokołów, które definiowałyby i obsługiwały scenariusz wymian. Jednak otwartość stosu nie wyklucza ich istnienia. Zatem bez wnikania w szczegóły można zauważyć, iż głównym problemem wykorzystania protokołu TCP/IP w systemach czasu rzeczywistego będzie realizowalność deterministycznego modelu kontroli wymian. Osobne zagadnienie stanowi oczywiście problem, czy i kiedy determinizm jest niezbędny, a kiedy system komunikacyjny może z niego zrezygnować. Problem ten będzie rozwinięty w dalszej części pracy.

Biorąc pod uwagę szerokość pasma warstwy fizycznej sieci, otrzyma się wartość maksymalnej prędkości przesyłania danych. Duża prędkość przesyłania danych będzie miała miejsce tylko przy jednokierunkowym połączeniu typu punkt-punkt. W pozostałych przypadkach zależeć ona będzie od opóźnień sieci i przyjętego protokołu wymiany informacji, a konkretnie od przyjętego w nim modelu wymiany danych. W skrajnym przypadku, mimo dużej wartości prędkości przesyłania danych, wymiana informacji może stać się niemożliwa. Każda sieć rzeczywista wprowadza opóźnienia. Są to opóźnienia wynikające z propagacji sygnału po medium, opóźnienia urządzeń pośredniczących czy w końcu opóźnienia wynikające z oczekiwania na dostęp do współdzielonego medium. Biorąc pod uwagę wzór na opóźnienie bieżące [13]:

$$D = \frac{D_0}{(1-U)} \tag{11}$$

gdzie:

- D_0 – opóźnienie nieobciążonej sieci,
- U – współczynnik bieżącego wykorzystania sieci (liczba z zakresu 0-1) względem szerokości pasma,
- D – opóźnienie bieżące,

można zauważyć, iż wraz ze zmniejszaniem się wartości mianownika opóźnienie bieżące rośnie do nieskończoności. Zatem jeśli sieć jest obciążana, to musi rosnąć czas transmisji informacji przez tą sieć i w skrajnym przypadku musi to prowadzić do przeciążenia i niewydolności komunikacyjnej. W rozważaniach o wydajności sieci musi pojawić się zatem pojęcie przepustowości efektywnej, która jest zawsze mniejsza od wartości szerokości pasma. Dla zastosowań, gdzie niedopuszczalne jest pojawienie się przeciążeń, ratunkiem staje się protokół deterministyczny.

Należy przyjąć, iż w systemach przemysłowych istnieje grupa danych krytycznych, których zaburzenie dystrybucji jest niedopuszczalne. Systemy, w których brak jest takich danych nie są systemami przemysłowymi czasu rzeczywistego. Jednak najczęstszy przypadek, to systemy zawierające grupę danych krytycznych oraz grupę danych niekrytycznych. Przykład może stanowić prezentacja informacji dla użytkownika. Przy prezentacji zachodzi potrzeba stabilnej w czasie akwizycji danych wraz ze stemplami czasowymi określającymi ich jakość, natomiast przesył do urządzenia prezentującego nie musi być obarczony czasem krytycznym.

W systemie czasu rzeczywistego pojawienie się przeciążenia powoduje, iż system przestaje reagować w określonym czasie, a co za tym idzie przestaje nim być. Wynika z tego, że użycie niedeterministycznych mechanizmów komunikacyjnych na poziomie wymiany informacji pomiędzy uczestnikami kontroli procesu wymieniającymi dane krytyczne jest niedopuszczenie. W dalszej części pracy pokazano, iż istnieje możliwość projektowania systemów komunikacyjnych TCP/IP w taki sposób, aby wymiany poziome realizowały transport danych krytycznych a wymiany pionowe nie musiały mieć charakteru zdeterminowanego w czasie. Daje to szansę wykorzystania mechanizmów komunikacyjnych TCP/IP w przemysłowych systemach kontrolnych pracujących w czasie rzeczywistym.

5.3. Zastosowanie sieci ETHERNET

Stos TCP/IP może pracować na różnych fizycznych standardach sieciowych. Jeden z najpopularniejszych standardów w sieciach LAN stanowi sieć Ethernet. Standard Ethernet definiuje dwie najniższe warstwy interfejsu sieci. W dalszych rozważaniach skoncentrowano się w tych warstwach, na tym właśnie rozwiązaniu. Współdziałanie protokołu TCP/IP z siecią Ethernet odbywa się właśnie na poziomie warstwy fizycznej i interfejsu sieci.

Obecnie sieć Ethernet stała się standardem w sieciach intranetowych, czyli również standardem w podłączeniu węzła końcowego do wirtualnej intersieci pracującej ze stosem TCP/IP. Podłączenia *point-to-point* protokołem PPP czy SLIP lub inne mechanizmy sieci lokalnych są stosowane znacznie rzadziej niż Ethernet. Jedynie dla specyficznych rozwiązań, gdzie mamy do czynienia z intersieciowymi abonentami dołączanymi tymczasowo do systemu lokalnego opartego o wydzielony system komunikacyjny, można wykorzystywać podłączenie telekomunikacyjne komutowane lub inne tego typu. Najczęściej takie rozwiązania stosuje się dla potrzeb zdalnego serwisowania lub podglądu stanu procesu o charakterze sporadycznym. Generalnie, tego typu podłączenia systemów lokalnych do intersieci nie są stosowane dla celów realizacji pracy samego systemu, a jedynie jako funkcje dodatkowe, bez których system jest w stanie funkcjonować samodzielnie. W przypadkach, gdy abonenci intersieciowi stanowić mają integralną część systemu, wskazane jest podłączenie systemu na bazie sieci lokalnej. Zarówno dla wydzielonych zamkniętych systemów komunikacyjnych jak i systemów otwartych. Więcej na ten temat znajduje się w rozdziale 6 oraz 7.

Niniejsza rozprawa ma na celu pokazać zastosowanie warstw TCP/IP i nie ma sensu udowadniać, czy taka lub inna platforma fizyczna dla pracy tych warstw jest lepsza czy też gorsza. Ethernet stanowi platformę, na której może być wykorzystany protokół TCP/IP w sieciach lokalnych. Sieć ta, pomimo, iż jej założenia koncepcyjne zostały stworzone wiele lat temu, nadal jest rozwijana, unowocześniana i nic nie wskazuje na jej rychły koniec.

W sieci Ethernet, rozwiązaniem, które jest wykorzystywane do koordynacji wysyłania ramek jest mechanizm CSMA/CD, czyli wykrywanie fali nośnej z detekcją kolizji. Jeżeli na sieci działa przynajmniej dwóch abonentów, wówczas istnieje prawdopodobieństwo wystąpienia kolizji pakietów. W zależności od tego, jaka jest stosowana technika budowy

magistrali, mogą wystąpić fizyczne kolizje na medium lub logiczne w przełączniku. Prawdopodobieństwo kolizji zwiększa się wraz ze wzrostem ruchu na sieci.

Protokół Ethernet ma za zadanie przesłać pakiet od nadawcy do abonenta docelowego w obrębie sieci lokalnej. Jeżeli przy takiej transmisji wystąpi kolizja, stosowany jest mechanizm dwójkowego wykładniczego odczekiwania. Stacja wówczas opóźnia kolejną próbę transmisji o losowy czas z zakresu od 0 do *d*. Kolejne kolizje dla danego pakietu powodują podwojenie zakresu czasu, z którego wybierany jest czas opóźnienia. Zatem prawdopodobieństwo wystąpienia kolizji w kolejnych próbach maleje, gdyż maleje prawdopodobieństwo wylosowania takich samych czasów opóźnień przez stacje zainteresowane transmisją. Prawdopodobieństwo wystąpienia kolizji jednak nigdy nie jest zerowe.

5.4. Problemy zastosowania sieci względem wymogów czasu rzeczywistego

Generalnie w pracy przyjęto, iż istnieją trzy zadania funkcjonalne przemysłowych systemów kontrolno nadzorczych: sterowanie, prezentacja oraz rejestracja informacji (więcej na ten temat znajduje się w rozdziale 14). Do celów takich jak przekazywanie informacji w celu sterowania procesem dostęp do danych powinien być deterministyczny. Proces prezentacji informacji nie jest obarczony tak silnymi wymaganiami czasowymi. Dla procesu rejestracji danych istotne są stemple czasowe określające jednoznacznie czas wystąpienia zdarzenia, natomiast czas dostarczenia danych do stacji raportującej jest drugorzędny.

Jeżeli od systemu wymaga się sterowania procesem w czasie rzeczywistym, to niezbędne jest zagwarantowanie kanału komunikacyjnego na bazie sieci umożliwiającej deterministyczny dostęp do danych. W pozostałych przypadkach można nie przejmować się tym zagadnieniem a jedynie dopasować możliwości sieci do wymagań funkcjonalnych systemu [34].

Zgodnie z informacjami przedstawionymi w podrozdziale 5.2, prawdopodobieństwo kolizji w sieci Ethernet może maleć, lecz nigdy nie osiągnie wartości 0. Wynika stąd, iż z prawdopodobieństwem zmierzającym do pewnego minimum, zależnego od maksymalnej liczby powtórzeń prób retransmisji pakietu (standard określa 10 prób), istnieje możliwość nieuzyskania dostępu do medium w danej próbie transmisji.

Pierwsze pytanie, jakie się pojawia, to czy jest jakieś graniczne prawdopodobieństwo, które można uznać za pomijalnie małe, natomiast drugie pytanie, to czy istnieje sposób na eliminację kolizji.

Nie można powiedzieć, iż jeżeli prawdopodobieństwo utraty pakietu lub przekroczenia czasu krytycznego dostępu do łącza jest małe i wynosi np. 10^{-12}, to zdarzenia takie nie zachodzą. Można natomiast spróbować ustalić dla danej sieci gwarantowany poziom prawdopodobieństwa wystąpienia danego zdarzenia, i zastanowić się czy taki poziom jest wystarczający dla konkretnego zastosowania.

Przy eliminacji kolizji, protokół CSMA/CD ma za zadanie przetransmitować otrzymane od warstw wyższych dane od nadawcy do adresata. Protokół ten nie decyduje kiedy powinien

nadejść pakiet od warstwy wyższej. Każdy pakiet, który warstwy wyższe przekazują do warstw Ethernetu zostaje natychmiast przez nie obsłużony. Wstrzymanie transmisji na medium może zaistnieć tylko na wskutek kolizji i działania protokołu CSMA/CD. Jeżeli zatem warstwa wyższa stosu protokołów danej stacji zadba o to, aby w momencie przekazania pakietu do warstwy Ethernetowej, na sieci nie nadawała żadna inna stacja, to otrzymamy mechanizm nadrzędnego strażnika eliminującego możliwość powstania kolizji. Mechanizm detekcji kolizji Ethernetu stanie się wówczas niewykorzystany. Aby dokonać takiej modyfikacji należy zaimplementować w warstwach aplikacyjnych jeden z deterministycznych modeli wymiany informacji. Spowoduje to, iż warstwy aplikacji poszczególnych abonentów będą wzajemnie koordynowały transmisję pakietów. Wymiany staną się wówczas zdeterminowane czasowo, czego wykazanie stanowi cel pracy względem tezy pierwszej. Mechanizm taki zastosowano w praktyce dla sieci Ethernet i modelu PDC (załącznik VI.A) i opisano w rozdziale 6.

Istotne dla określenia współpracy sieci komputerowej z systemem przemysłowym jest określenie charakteru wymian. Sieci deterministyczne jako jedyne są w stanie zapewnić realizację wymian cyklicznych. Wymiany te umożliwiają stały dostęp do danej informacji z gwarantowanym czasem, określonym przez wartość okresu cyklu. Wymiany aperiodyczne w sieciach przemysłowych są realizowane na żądanie, jednak czas ich realizacji jest określony w przedziale. Należy zauważyć jedną bardzo istotną cechę wymian cyklicznych, która będzie miała znaczenie przy wyborze metod komunikacji ze stosu TCP/IP. Przy realizacji wymian cyklicznych nie można pozwolić na dokonywanie retransmisji. Powód jest prosty. Każda retransmisja zmiennej powoduje zajęcie czasu sieci, co w konsekwencji może doprowadzić do zaburzenia cyklu. Gdy czas zużyty na retransmisję przekroczy okres cyklu danej zmiennej, sama retransmisja nie ma już sensu, gdyż retransmisję zapewni sam cykl sieci. Czas ten natomiast, będzie musiał zostać zabrany innym zmiennym lub trzeba będzie przewidzieć zapas czasu w cyklu, kosztem samego cyklu. Niektóre rozwiązania stosują mechanizm retransmisji, jednak jest ona zawsze ograniczona do konkretnej liczby prób.

W sieciach niedeterministycznych, zapewnienie stabilnego cyklu nie jest możliwe. Zatem mamy do czynienia zawsze z wymianami acyklicznymi. Jeżeli istnieje potrzeba budowania transakcji cyklicznej, wówczas można pokusić się o stworzenie pseudo cyklu, który wymuszany będzie przez odpowiednie warstwy aplikacji, lecz jego realizacja zależeć będzie od samej sieci i jej mechanizmów kontrolowania dostępu do medium i przekazywania pakietów. Wprowadzanie możliwości retransmitowania zmiennych w tego typu sieciach wydaje się najlepszą drogą do zapychania łącza. Inaczej sytuacja wygląda w przypadku wymian z założenia aperiodycznych. Z wymianami tymi często wiązane są polecenia i rozkazy. Dla tego typu zmiennych otrzymywanie potwierdzeń od odbiorcy i ewentualna retransmisja jest wymogiem bezpieczeństwa pracy systemu.

5.5. Rozwój technologiczny sieci i jego wpływ na parametry transmisji

Sieć Ethernet zaistniała oferując szybkość transmisji do 10Mbps. Obecnie dominującą wersją jest Ethermet 100Mbps. Pojawienie się nowej wersji tzw. gigabitowej (1Gbps, 10Gbps) spowodowało dalsze poszerzenie pasma transmisyjnego tej sieci [49, 74].

Ethernet gigabitowy (1000BASE-X, IEEE 802.3z) [137] udostępnia teoretyczną szybkość transmisji wynoszącą 1000 Mbps. Używa on tego samego formatu ramki Ethernetowej i mechanizmu kontroli dostępu do medium co wszystkie poprzednie technologie Ethernetowe (IEEE 802.3). Również tak samo jak jego wolniejsi prekursorzy Ethernet gigabitowy określa tylko warstwę fizyczną i łącza modelu OSI. Zatem w dalszym ciągu stanowi on uzupełnienie warstw wyższych protokołu TCP/IP, nie wnosząc nic nowego oprócz szybkości.

Istotne staje się rozważenie, co daje poszerzenie pasma przy zachowaniu protokołu. Przede wszystkim, poszerzenie pasma w sieciach opartych o CSMA/CD zmniejsza maksymalny czas transmisji pakietu w sieci, a zatem przy takim samym natężeniu ruchu spada prawdopodobieństwo wystąpienia kolizji. Konsekwencją tego jest obniżenie gwarantowanego poziomu prawdopodobieństwa utraty pakietu lub nieuzyskania dostępu do łącza. Pomimo, że prawdopodobieństwo to nigdy nie osiągnie wartości zero, to zwiększenie szybkości transmisji może spowodować obniżenie prawdopodobieństwa przeciążenia poniżej gwarantowanego parametru MTBF dla sprzętu. W wielu przypadkach jest to wystarczający powód, aby uznać taki mechanizm transmisji za zadowalający. Rozwiązanie takie traktuje protokół jako kolejny element mogący ulec uszkodzeniu. Natomiast sam protokół w sensie pewnej abstrakcji, jak np. algorytm, może być idealny względem możliwości utraty danych a co za tym idzie w pełni niezawodny. Wykorzystując protokół zakładający prawdopodobieństwo dostarczenia informacji na poziomie mniejszym od jedności, decydujemy się jednocześnie na dodatkowy awaryjny element systemu.

Całkowite rozwiązanie problemu determinizmu jest możliwe tylko po przez zmianę modelu wymiany danych. Poszerzanie pasma sieci tylko redukuje problem. Zawsze istnieje możliwość niedotrzymania wymaganych parametrów transmisji. Poza tym, jeżeli określono prawdopodobieństwo utraty danych dla danego protokołu na danym poziomie, to należy to zrobić dla skończonego zbioru zmiennych systemowych i stałych parametrów sieci. Prawdopodobieństwo utraty danych w protokołach niedeterministycznych jest funkcją wielu czynników. Do najważniejszych należą:

- prędkość transmisji,
- natężenie ruchu,
- liczba obsługiwanych zmiennych,
- liczba abonentów.

Dla przykładu, jeśli przyjęto możliwość utraty pakietu na poziomie prawdopodobieństwa równego 10^{-6}, i będzie to oznaczało, że co milionowa wymiana będzie niepomyślna. Przy cyklu wymiany informacji dla dziesięciu zmiennych wynoszącym 5 [ms], w ciągu 1 sekundy wystąpi 2000 wymian. Spowoduje to, że co ok. 8 minut będzie występować utrata danych.

W niektórych systemach przemysłowych lub dla określonych grup informacji taka awaryjność jest niedopuszczalna.

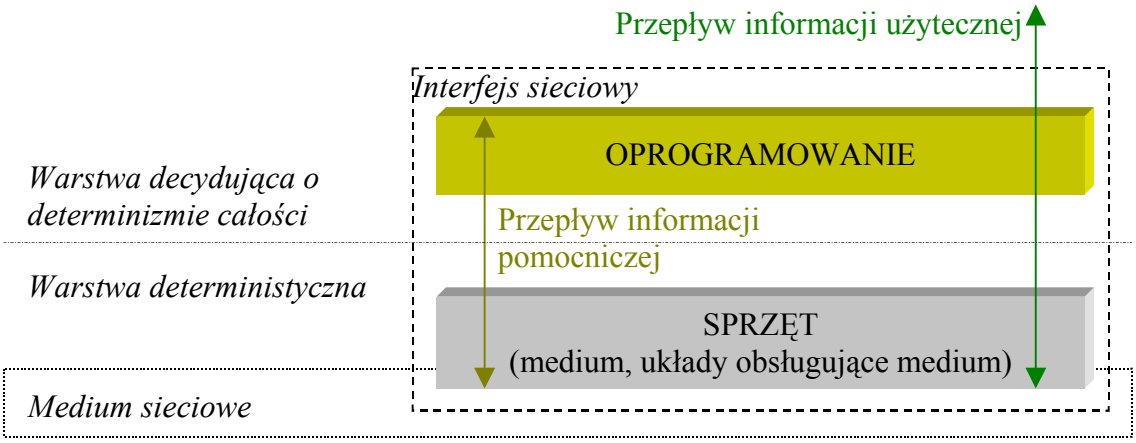
Dla stałej grupy informacji, wzrost prędkości transmisji zmniejsza możliwość kolizji, gdyż każda zmienna przebywa w medium przez krótszy okres czasu. Jednak dla każdej prędkości transmisji można spowodować taki ruch w sieci, który spowoduje zwiększoną awaryjność.

Analizując awaryjność niedeterministycznej warstwy komunikacyjnej nie wolno bazować tylko na prędkości transmisji. Informacja, iż sieć działa z daną prędkością nie określa zawodności połączenia. Dopiero informacja o prawdopodobieństwie wystąpienia utraty danych dla danej prędkości oraz dla danego cyklu wymian, może stanowić wiarygodny wyznacznik niezawodności połączenia. Precyzyjne określenie cyklu wymian dla sieci bez kontroli tych wymian jest niemożliwe, lecz ponieważ rozważane są zjawiska oparte o statystykę, można zbudować cykl przybliżony. Cykl taki proponuje się oprzeć o średnią liczbę wymian na jednostkę czasu. Zatem wskazane jest określanie, jakie jest prawdopodobieństwo utraty danych dla sieci realizującej transmisję z natężeniem danej liczby bajtów na sekundę.

6. Wykorzystanie TCP/IP do kontroli i nadzoru procesów przemysłowych

Składowe stosu protokołów TCP/IP przedstawione w rozdziale 5.1 oraz szczegółowy opis budowy tego stosu zamieszczony w [13, 14, 15, 16, 19, 77] dowodzą, iż protokół TCP/IP nie definiuje całości struktury interfejsu komunikacyjnego[4]. Dlatego nie sposób określić czy interfejs sieciowy wykorzystujący protokół TCP/IP umożliwia zdeterminowaną w czasie realizację wymian czy też nie, bazując tylko na podstawie standardowych jego warstw.

Model warstwowy wspomniany w rozdziale 5, można uprościć do dwóch warstw: warstwy fizycznej i warstwy wyższej obsługującej warstwę fizyczną. Można inaczej powiedzieć, że każdy interfejs komunikacyjny składa się z dwóch elementów: sprzętu i oprogramowania. Przedstawiono to na rysunku 18.



Rys. 18 Uproszczona budowa interfejsu sieciowego

Warstwa sprzętu, rozpatrywana w oderwaniu od reszty systemu, jest zawsze deterministyczna. Stosując dowolne medium do transmisji danych w topologii niezależnych jednokierunkowych kanałów typu punkt – punkt (4.2) zawsze otrzymamy łącze działające deterministycznie a wymiany będą realizowane w sposób zdeterminowany czasowo w przedziale czasu ograniczonym wielkościami fizycznymi charakteryzującymi dany sprzęt np. prędkością rozchodzenia się sygnału w danym medium. Niezdeterminowanie działania systemu pojawia się wówczas, gdy system zawiera obiekt współdzielony. System taki można wówczas porównać do typowego systemu rozproszonego z problemem synchronizacji międzyprocesowego dostępu do współdzielonego zasobu [12]. Zatem aby system pracował

[4] wewnętrznej budowy, elementów składających się na jego konstrukcję

w sposób zdeterminowany należy do takiego obiektu zapewnić dostęp synchronizowany z zadanym scenariuszem wymian.

W rozważanym systemie zasób współdzielony stanowi obiekt komunikacyjny. Proponowaną aplikacją obiektu komunikacyjnego jest sieć Ethernet. Stanowi ona kanał magistralowy (rys. 8). W celu zapewnienia synchronizacji dostępu do takiego obiektu należy stosować mechanizmy zapewniające przejmowanie kontroli nad obiektem na zasadzie wyłączności przez jednego i tylko jednego abonenta z wywłaszczaniem. Za determinizm czasowy działania takiego mechanizmu odpowiedzialny jest algorytm zaimplementowany w warstwach oprogramowania (rys. 18), czyli protokół. Protokół zawsze jest zaszyty w warstwach oprogramowania, nawet jeżeli pozornie jest realizowany przez komponenty sprzętowe (np. karta sieciowa) [59, 38]. Jeżeli protokół umożliwia stworzenie i wykorzystanie procedury obsługującej kontrolę dostępu do medium i wszyscy abonenci pracujący na tym medium będą posiadali tak samo działający interfejs komunikacyjny, wówczas determinizm czasowy prowadzenia wymian będzie realizowalny.

Sieci komputerowe można ogólnie podzielić na mono i heterogeniczne. Przez sieci monogeniczne rozumiane będą takie rozwiązania komunikacyjne, w których wszyscy abonenci wykorzystują to samo medium oraz taki sam protokół. Sieci heterogeniczne natomiast są sieciami, w których pracuje więcej niż jeden protokół i/lub składają się z segmentów różnego typu mediów o różnej topologii.

Praktyczne zastosowanie jednolitych interfejsów zarządzających dostępem do medium można rozważać tylko w skali sieci monogenicznych, jakimi mogą być na przykład sieci lokalne [13, 78]. Stanowi to jeden z aspektów zastosowania protokołu TCP/IP. Analizę możliwości wykorzystania takich interfejsów sieciowych należy rozpocząć od warstwy najniższej, która realizuje fizyczny transfer po medium. Warstwa ta nie jest definiowana przez standard TCP/IP. Definicje stosu TCP/IP nie mówią, w jaki sposób należy transmitować sygnały i z wykorzystaniem jakiego medium. Sytuacja taka ma miejsce, gdyż głównym obszarem działania protokołu TCP/IP jako protokołu intersieciowego jest heterogeniczna sieć wirtualna. Analizowanie pracy warstw najniższych w skali takiej wirtualnej intersieci jest niemożliwe. Należy zatem wykonać analizę pracy protokołu zależną od obszaru, na którym on funkcjonuje. Ponieważ funkcjonowanie protokołu odbywa się na rzecz systemów kontrolno-nadzorczych niezbędne staje się zdefiniowanie podziału tych systemów, względem którego analiza pracy może być prowadzona.

Podział ten został wykonany mając na uwadze, iż celem pracy jest wykorzystanie sieci Ethernet w warstwie sprzętowej a protokołu TCP/IP w warstwie programowej interfejsu sieciowego abonenta przemysłowego systemu kontrolno nadzorczego.

6.1. Podział systemów kontrolno-nadzorczych

Określenie obszarów funkcjonowania sieci komputerowych, w których zastosowanie znajduje protokół TCP/IP lub z którymi on współpracuje stanowi kluczowe zagadnienie na bazie którego zastosowanie to można rozważać.

Proponuje się, aby jako podstawę do dalszych rozważań przyjąć funkcjonalny podział rozważanego systemu informatycznego na:

– system lokalny,

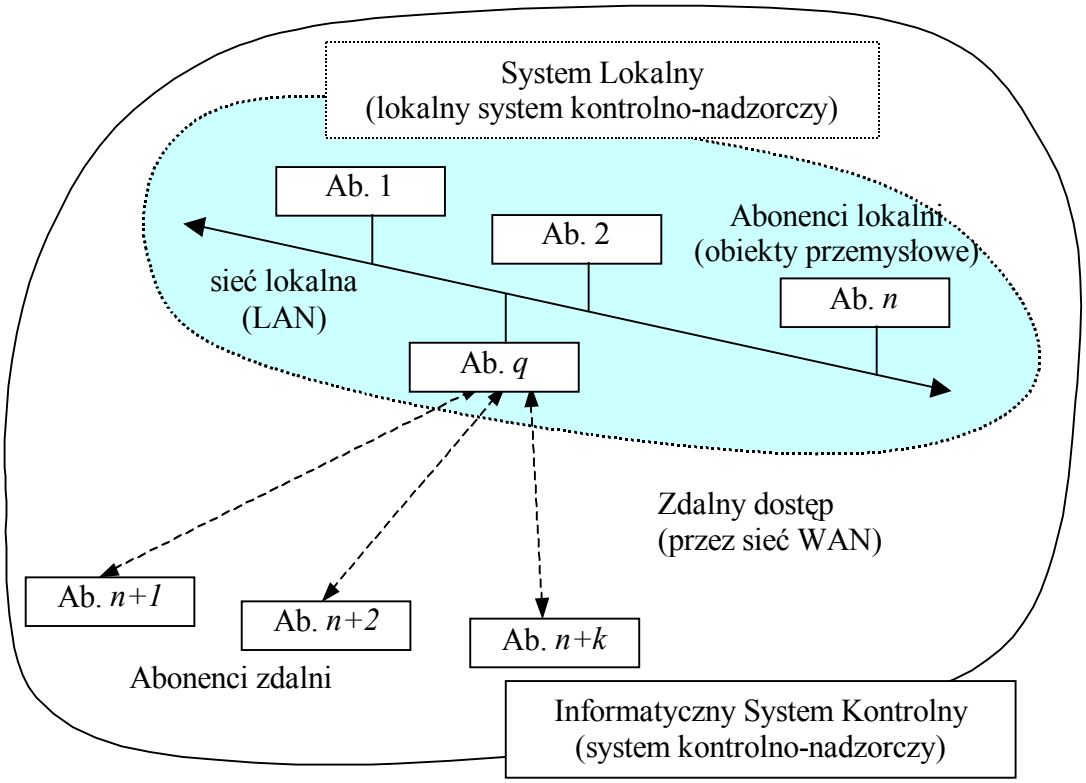
– system zdalny.

System lokalny składa się z określonej i skończonej liczby abonentów obsługujących określoną i skończoną liczbę zmiennych. Abonenci oraz zmienne nie wchodzące w skład systemu lokalnego stanowią system zdalny. Określenie granic systemu lokalnego stanowi istotne zagadnienie, które jest rozważane w rozdziale 6.2.

Przyjęto, iż w systemie lokalnym abonenci komunikują się ze sobą przy wykorzystaniu wspólnego kanału komunikacyjnego zrealizowanego w formie magistrali i stanowiącego sieć lokalną[5] systemu (LAN) (zob. rys. 8). Dla odróżnienia do klasycznych sieci lokalnych [13, 96, 97, 28], sieć tą przyjęto nazywać siecią systemową. Wszyscy abonenci sieci systemowej ze swoimi stosami protokołów i aplikacjami stanowią system lokalny.

6.1.1. Zdalny dostęp

Na rysunku 19 przedstawiono schemat systemu kontrolnego z podziałem na system lokalny (*n* abonentów o indeksach 1..*q*..*n*) oraz abonentów realizujących do niego dostęp zdalny (*k* abonentów o indeksach *n+1*..*n+k*). Abonent *q* został przedstawiony jako „brama" pomiędzy systemem lokalnym a abonentami zdalnymi. Zadanie pośrednictwa może spełniać również więcej niż jeden abonent systemu lokalnego.



Rys. 19 System kontrolny, system lokalny i zdalny dostęp

[5] stanowi ona sieć obiektową (terenową, polową czy też sterującą)

Zdalny dostęp do danej informacji z grupy informacji obsługiwanej przez system lokalny można zdefiniować jako możliwość obsługi tej informacji poza systemem lokalnym. Wymiana pakietów zachodzi wówczas pomiędzy abonentem lub abonentami systemu lokalnego a abonentem lub abonentami zdalnymi, nie stanowiącymi elementów systemu lokalnego. Na przykład odwołujących się z poziomu sieci rozległej (WAN, WLAN) lub innego systemu lokalnego.

Całość heterogenicznych struktur sieciowych łączących abonentów zdalnych z systemem lokalnym wraz z lokalnymi systemami komunikacyjnymi tych abonentów traktowana będzie jako wirtualna sieć zewnętrzna. Sieć zewnętrzna będzie rozumiana jako sieć nie należąca do struktury sieci systemowej, a będąca wykorzystywana przez warstwę komunikacyjną danego informatycznego systemu przemysłowego. Cała wykorzystywana zewnętrzna sieć wirtualna wraz ze zdalnymi abonentami nazwana została systemem zdalnym lub systemem zewnętrznym. Abonentów sieci zewnętrznej nazwano abonentami zdalnymi.

6.1.2. Zadania komunikacyjne protokołu

W kontekście całego przemysłowego systemu kontrolno – nadzorczego, który nie musi być ograniczony do systemu lokalnego, wykorzystanie protokołu TCP/IP powinno być rozważane w dwóch aspektach, jako realizacja zadań komunikacyjnych na poziomie:

1. sieci systemu lokalnego,
2. zdalnego dostępu do systemu lokalnego.

W dziedzinie sieciowej obsługi systemu kontrolnego na poziomie warstwy obsługi procesu (rys. 12) skoncentrowano się na zapewnieniu realizacji deterministycznych wymian w sieci Ethernet. Natomiast dla przypadków dopuszczających wykorzystanie zdalnego dostępu skupiono się na możliwości określania jakości danych użytecznych, bezpieczeństwie oraz nowych możliwościach funkcjonalnych systemów kontrolnych.

6.1.3. Przepływ danych w systemie

W systemie kontrolnym można wyróżnić dwa rodzaje połączeń:

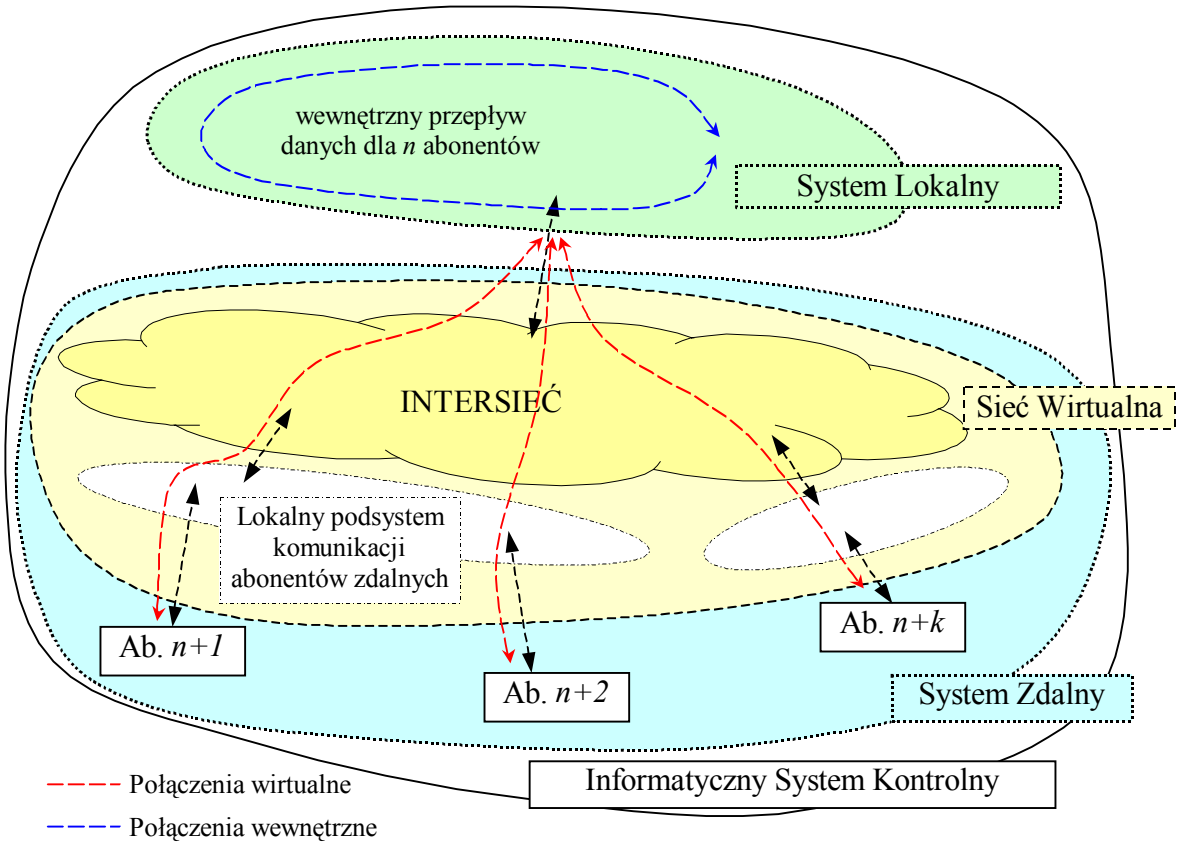
- wewnętrzne – połączenia podsystemu lokalnego,
- zewnętrzne – połączenia wirtualne realizujące zdalny dostęp.

Na rysunku 20 przedstawiono uogólniony schemat przepływu danych pomiędzy abonentami zdalnymi a systemem lokalnym.

Połączenia wewnętrzne są to połączenia, przy użyciu których dokonuje się przepływ informacji pomiędzy abonentami systemu lokalnego. Realizowane one są przy użyciu sieci systemowej i abonentów lokalnych.

Połączenia wirtualne są to połączenia pomiędzy abonentami zdalnymi a systemem lokalnym. Realizowane są poprzez specjalnego abonenta systemu lokalnego (rys. 19 „Ab. *q*”). Przepływ danych odbywa się pomiędzy abonentami systemu lokalnego a abonentami systemu zdalnego. Wymiany sieciowe są jednak realizowane zawsze za pośrednictwem abonenta

„bramy” (rys. 19) umożliwiającego podłączenie warstw komunikacyjnych systemu lokalnego do środowiska intersieciowego.



Rys. 20 Uogólniona struktura przepływu danych przy dostępie zdalnym

6.2. Definicja systemu lokalnego

Do określenia granicy systemu lokalnego można podejść w sposób analityczny. Zaproponowano metodę bilansowania informacji wzorowaną na metodzie wyznaczania granic bilansowania energii w termodynamice [83]. System kontrolny składa się z grupy abonentów oraz grupy obsługiwanych przez nie zmiennych. Można przyjąć, iż każda wytworzona zmienna ma przynajmniej jednego odbiorcę, a dla każdego odbiorcy zmiennej istnieje w systemie producent tejże zmiennej. Można wyznaczyć wirtualnego producenta i konsumenta informacji w systemie, zakładając że producent produkuje wszystkie zmienne wytwarzane przez rzeczywistych producentów systemu a konsument konsumuje wszystkie zmienne odczytywane przez rzeczywistych konsumentów systemu. System jest skonstruowany poprawnie, gdy:

$$L_{VP} = L_{VK} \tag{12}$$

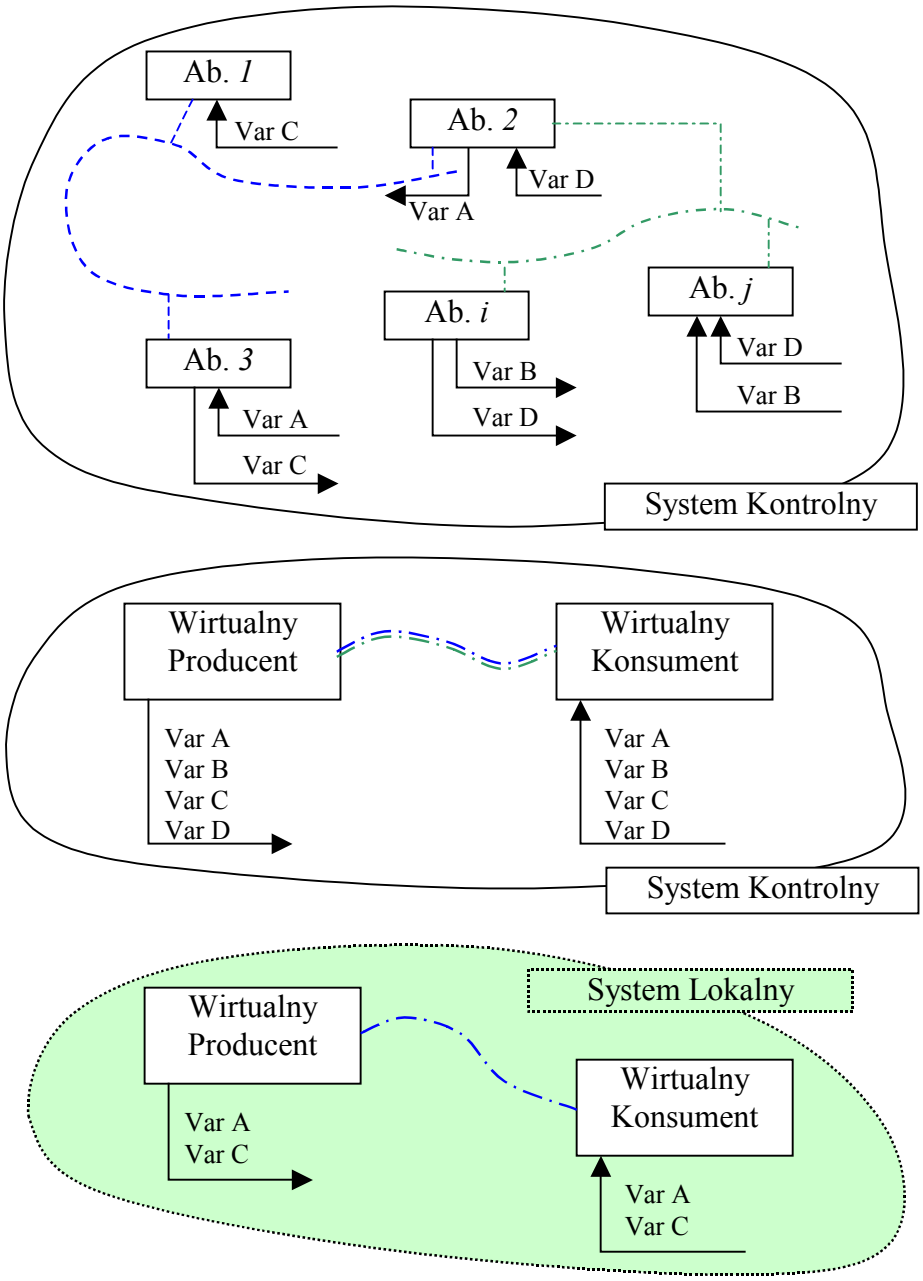
gdzie:

L_{VP} – liczba zmiennych produkowanych,

L_{VK} – liczba zmiennych konsumowanych.

Zasada bilansowania informacji odnosi się do całego systemu kontrolnego. Jest to podstawowa zasada jego poprawnej konstrukcji informatycznej. Możliwe jest jednak wydzielenie podsystemu, w którym również będzie zachodziło bilansowanie.

Jeżeli wyznaczanie podsystemu bilansowania zostanie odniesione do zmiennych obsługiwanych w ramach sieciowego podsystemu monogenicznego, to otrzyma się system lokalny. Zilustrowano to na rysunku 21.



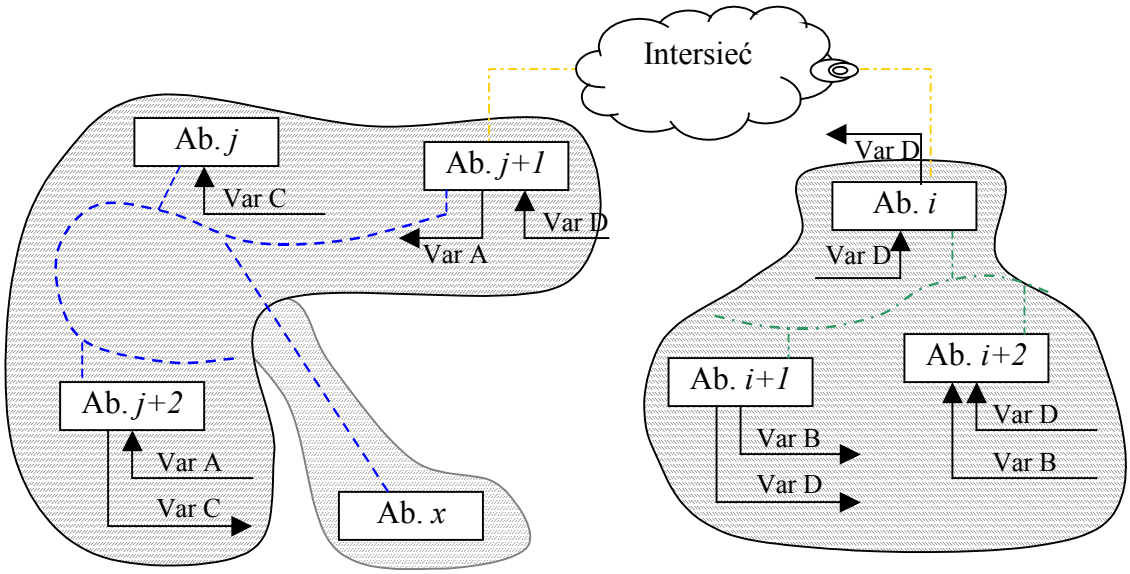
Rys. 21 Określanie granic systemu lokalnego przez bilansowanie informacji

System taki posiada określoną i skończoną, choć niekonieczne stałą, liczbę abonentów oraz określoną i skończoną liczbę obsługiwanych zmiennych.

Ogólnie można przyjąć, iż podział systemu na podsystem lokalny i zdalny jest sprawą względną i umowną. Dany podsystem może być lokalny a inny zdalny lub odwrotnie w zależności od punktu widzenia użytkownika. Jednak z punktu widzenia systemu

kontrolnego podsystem lokalny będzie związany z monogenicznym systemem komunikacyjnym. Szczególnym przypadkiem może być sytuacja, gdy mamy do czynienia z wieloma systemami lokalnymi połączonymi heterogenicznym systemem komunikacyjnym.

Logiczne podłączenie abonenta zdalnego do systemu lokalnego niezależnie od tego czy jest to producent czy konsument informacji, nie może wprowadzać zakłóceń w bilansowaniu systemu lokalnego i musi być zgodne z bilansowaniem informacji w całym systemie.



Rys. 22 Granice bilansowania informacji

Na rysunku 22 przedstawiono system kontrolny z wyodrębnionymi dwoma systemami lokalnymi. Połączenie między podsystemami zrealizowano w oparciu o intersieć. Z punktu widzenia danego podsystemu wszyscy abonenci do niego nie należący stanowią abonentów zdalnych. Przekazywanie informacji odbywa się po przez abonentów uczestniczących w wymianie informacji w więcej niż jednej sieci. Są to abonenci posiadający dwa lub więcej interfejsy sieciowe i realizujące odrębne scenariusze wymian. Abonent *x* podłączony bezpośrednio do systemu lokalnego nie stanowi abonenta zdalnego, lecz abonenta lokalnego danego podsystemu lokalnego, niezależnie od jego fizycznej lokalizacji i momentu podłączenia, o ile informacja przez niego obsługiwana bilansuje się z informacją danego podsystemu.

Przedstawiony sposób wyznaczania granic systemu lokalnego umożliwia jednoznaczne i dokładne określenie, którzy abonenci wchodzą w skład systemu i jakie zmienne system obsługuje. Będzie to niezbędne dla dalszej klasyfikacji systemów pod kątem współdziałania systemu lokalnego z systemem zdalnym oraz z abonentami nie stanowiącymi składowych systemu kontrolnego. Reasumując system lokalny charakteryzuje się następującymi cechami:

- praca w sieci monogenicznej (wspólny sprzęt i protokół – strona 42),
- określona i skończona liczba abonentów,
- określona i skończona liczba zmiennych,
- bilansowanie produkcji i konsumpcji zmiennych,
- określona granica systemu.

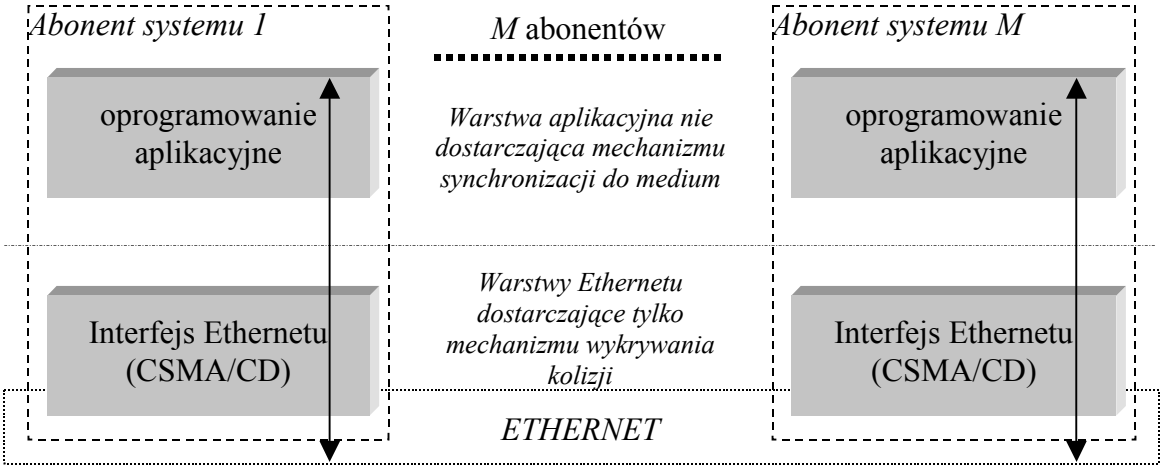
6.3. Problemy z wykorzystaniem sieci ETHERNET w aplikacjach przemysłowych

Jeden z celów pracy stanowi wykorzystanie standardu Ethernet do realizacji sprzętowej warstwy komunikacyjnej przemysłowego systemu kontrolno-nadzorczego. W świetle wcześniejszych rozważań należy przeanalizować czy sieć Ethernet nadaje się do konstrukcji zdefiniowanego powyżej systemu lokalnego.

Obecnie wielu producentów wyposaża swoje urządzenia w interfejsy sieci Ethernet. Często interfejs taki stanowi podstawowe łącze komunikacyjne dla systemów przemysłowych budowanych w oparciu te urządzenia. Istnieje również szereg komercyjnych rozwiązań wykorzystujących standard Ethernet do realizacji specjalizowanych sieci komunikacyjnych dedykowanych dla rozwiązań przemysłowych [63]. Są to dla przykładu:

- Industrial Ethernet firmy Siemens,
- Ethway firmy Schneider.

Specyfikacje tych rozwiązań są przeważnie niedostępne, a zatem użytkownik niewiele wie na temat obsługi informacji w jego systemie. Stosowanie klasycznego Ethernetu odbywa się często na zasadzie domyślnego założenia, iż „szybka sieć musi działać".



Rys. 23 System kontrolny na bazie Ethernetu bez deterministycznej kontroli dostępu do medium

Niezbędna staje się zatem analiza tych aspektów pracy sieci Ethernet, które są istotne dla pracy w środowisku systemów przemysłowych. Chodzi przede wszystkim o czas obsługi informacji w obiekcie komunikacyjnym, abstrahując od rodzaju protokołu pracującego w warstwach wyższych oraz zakładając, iż warstwy te nie wnoszą mechanizmów kontrolujących dostęp do medium (rys. 23).

6.3.1. Uwarunkowania pracy sieci ETHERNET

Ethernet dopuszcza możliwość utraty pakietu z pewnym prawdopodobieństwem. Utrata pakietu ma miejsce, gdy podczas transmisji z wykorzystaniem CSMA/CD nie uzyskano dostępu do łącza.

Od szybkości transmisji zależy tylko prawdopodobieństwo wystąpienia pierwszej kolizji dla danego pakietu, rozstrzygnięcie tej kolizji już nie. Mechanizm rozstrzygający działa dla

wszystkich stacji uczestniczących w kolizji. Następuje losowanie czasów opóźnień transmisji w interfejsach uczestniczących w kolizji według mechanizmu opisanego w 5.3.

Niestety nie wszystkie karty sieciowe działają na równych prawach. Praktyka i pomiary pokazują [28], że niektóre adresy sieciowe są preferowane przy rozstrzyganiu kolizji. Podobnie nowsze i szybsze modele kart sieciowych działają efektywniej niż modele starsze z wolniejszymi procesorami, częściej wygrywając pojedynki o dostęp do medium. Powszechnie występuje zjawisko monopolizacji łącza. Dzieje się tak wówczas, gdy dany interfejs sieciowy przegra rywalizację o dostęp z innym interfejsem. Generowana wartość czasu oczekiwania dla danej stacji jest wprost proporcjonalna do licznika kolizji tej stacji. Zatem im więcej kolizji tym dłużej interfejs sieciowy będzie czekał na ponowienie próby transmisji pakietu.

Wynika stąd, iż dokładne zamodelowanie pracy sieci jest trudne, a z punktu widzenia oceny przydatności dla systemów deterministycznych zbędne. Na potrzeby niniejszej pracy można założyć, iż mechanizm dostępu konkurencyjnego w sieci Ethernet działa sprawiedliwie, czyli rozkład czasu oczekiwania na dostęp do łącza w danym segmencie sieci jest taki sam dla wszystkich abonentów. Jest to założenie optymistyczne i w celu wykazania nieprzydatności standardowego mechanizmu kontrolowania dostępu wystarczające.

6.3.2. Ograniczoność stosowania standardu ETHERNET

Podtrzymując wstępne założenia, przydatność sieci Ethernet z punktu widzenia systemów czasu rzeczywistego jest ograniczona. Pomimo dość skutecznego mechanizmu wykrywania i rozwiązywania kolizji oraz znaczącej prędkości transmisji, jest to sieć, w której wymiany obsługiwane są w sposób niedeterministyczny. Ethernet wraz z protokołami pracującymi w warstwach wyższych, które nie umożliwiają uzyskania zdeterminowanego dostępu do łącza, można stosować tylko dla systemów nie wymagających ścisłych zależności czasowych lub z założeniem określonego poziomu ufności w odniesieniu do całego protokołu.

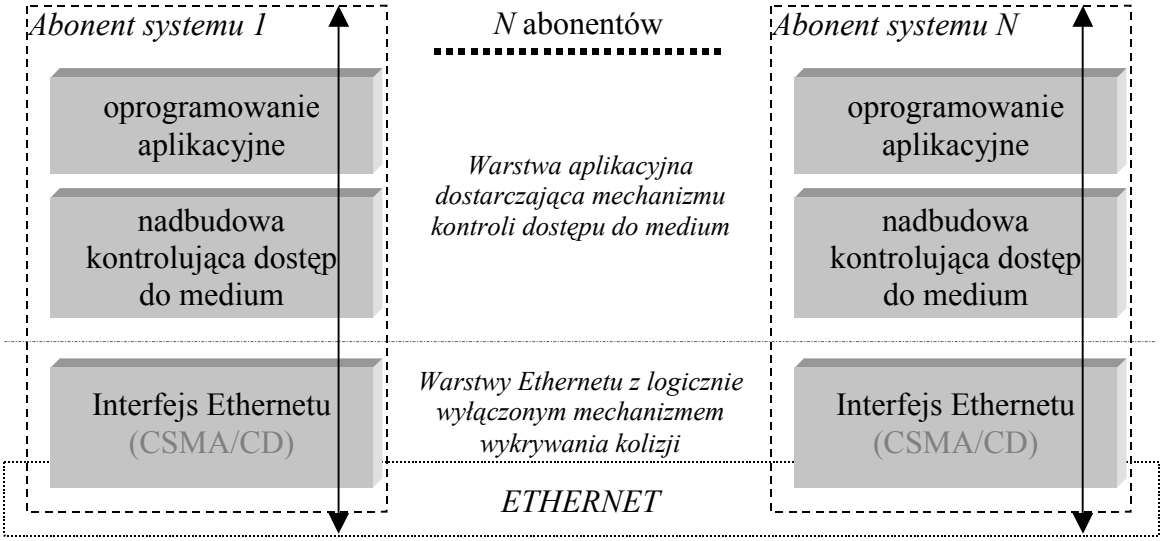
Polepszanie parametrów transmisji przez stosowanie szybszych standardów Ethernetu (5.5) nie zmienia uwarunkowań pracy i nie wnosi mechanizmów determinizmu czasowego. Zatem z punktu widzenia zastosowań w systemach przemysłowych szybki Ethernet nadal nie gwarantuje obsługi informacji w czasie rzeczywistym (5.4).

6.4. Determinizm czasowy wymian w sieci ETHERNET

Z punktu widzenia determinizmu czasowego wymian, Ethernet nie wnosi warstw umożliwiających kontrolowanie dostępu do medium. Rozwiązania takie są jednak możliwe przez tworzenie warstw dodatkowych (rys. 24) w postaci nadbudowy aplikacyjnej sterującej wymianami w taki sposób, aby dostęp do medium był synchronizowany względem scenariusza wymian [60, 61] związanego z pracą danego systemu lokalnego.

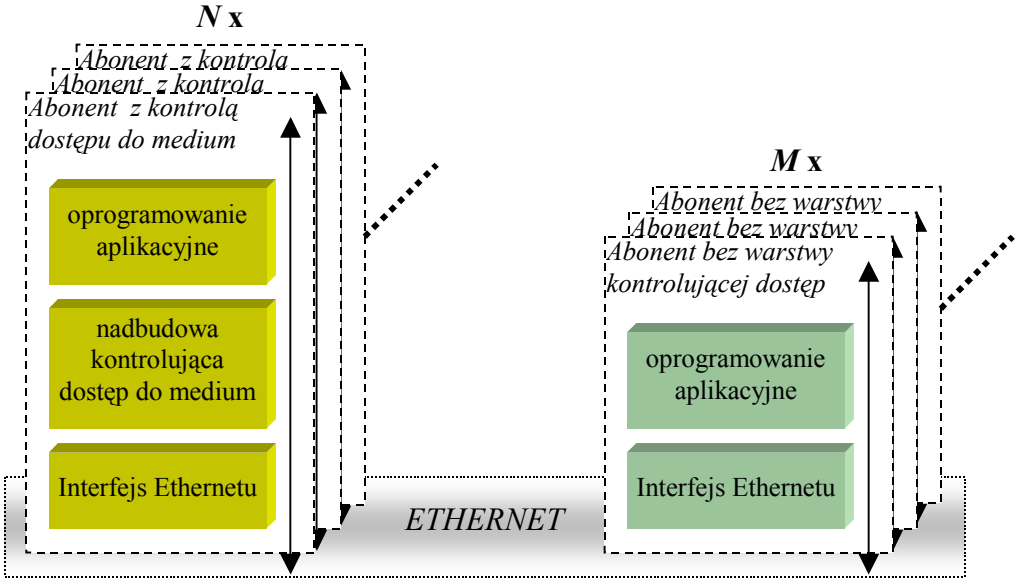
Wszystkie żądania zapisu na sieć generowane przez aplikację użytkownika powinny przechodzić przez warstwę nadrzędną. Warstwa ta nie przekazuje żądań do warstw niższych natychmiastowo, lecz dopasowuje momenty przekazania żądania do momentów

przewidzianych przez scenariusz wymian. Warstwa aplikacji użytkownika wykonuje tylko zlecenie zapisu do warstwy kontrolnej. Zapis sieciowy jest realizowany niezależnie od warstwy użytkownika przez nadrzędną warstwę kontrolną. Mechanizm CSMA/CD zostaje wówczas logicznie wyłączony i praktycznie nie jest wykorzystywany.



Rys. 24 System kontrolny na bazie Ethernetu z warstwą kontrolującą dostęp do medium

Otrzymuje się magistralowy kanał komunikacyjny, w którym tylko jeden abonent w danym czasie ma prawo wysyłania danych. Działanie tych warstw musi bazować na jednym z deterministycznych modeli wymian (zob. strona 21), a realizacja tych wymian musi być określona przez scenariusz [60, 61]. Warstwy takie mogą zagwarantować pracę łącza zdeterminowaną w czasie, o ile wszyscy abonenci systemu lokalnego będą posiadać tak samo działającą nadbudowę. Uzasadnia to pierwszą tezę pracy. Więcej na temat konstrukcji warstwy nadrzędnej znajduje się w rozdziale 8.



Rys. 25 System kontrolny z mieszanym dostępem do medium

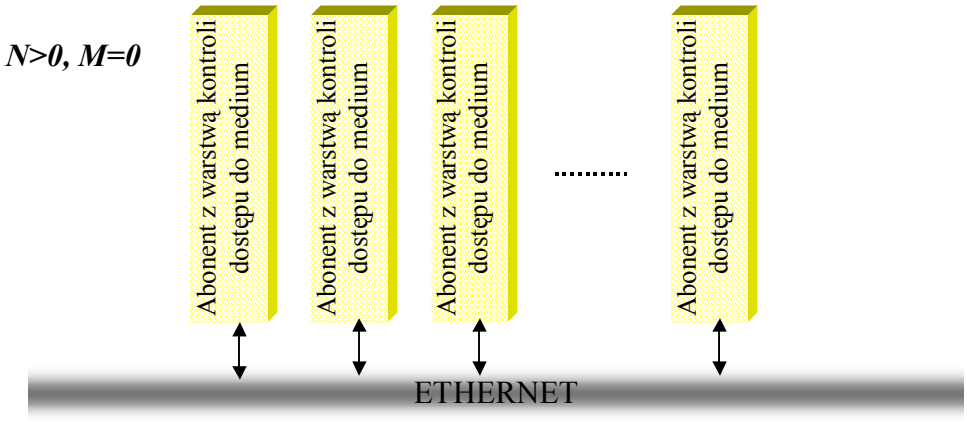
Przypadek, że wszyscy abonenci mają jednakowo działające warstwy nadbudowy stanowi przypadek komfortowy. W praktyce nie każdego abonenta można przystosować do pracy

zgodnej z przyjętymi zasadami zarządzania dostępem. Na rysunku 25 przedstawiono system z *N* abonentami posiadającymi warstwy kontrolujące dostęp do medium oraz *M* abonentów nie posiadających takich warstw.

Pojawienie się w systemie choćby jednego abonenta niedostosowanego do przyjętych zasad (*M*>0) powoduje, że traci się cechę zdeterminowanej realizacji wymian. Wówczas należy oceniać pracę kanału komunikacyjnego bazując na szacowaniu intensywności ruchu [57] i szacowaniu prawdopodobieństwa utraty pakietu względem przyjętego kryterium np. awaryjności sprzętu. Skrajnym przypadkiem takiego systemu jest przypadek, gdy *N*=0, czyli wszyscy abonenci pracują w sposób klasyczny. Otrzymano trzy przypadki funkcjonowania sieci Ethernet:

1. Sieć realizuje wymiany w sposób zdeterminowany czasowo.

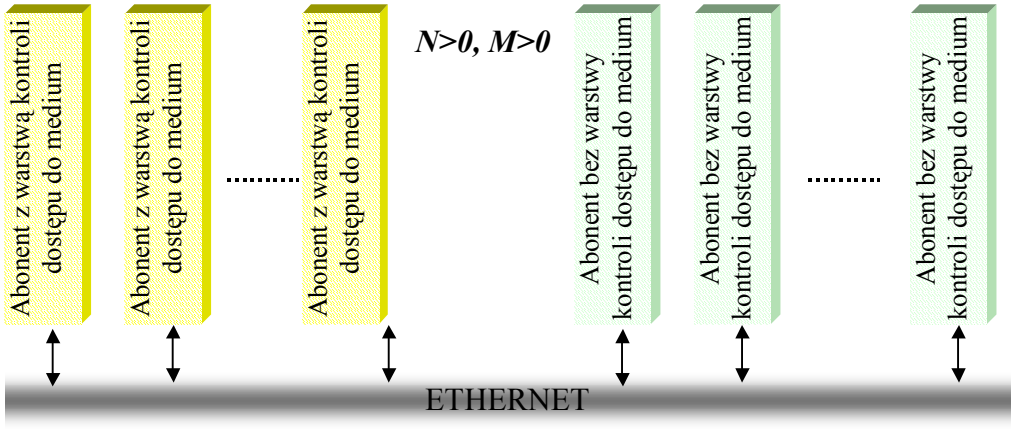
W przypadku tym, wszyscy abonenci posiadają specjalne warstwy aplikacyjne umożliwiające zarządzanie dostępem do medium komunikacyjnego. Oznacza to, że w warstwach interfejsów komunikacyjnych wszystkich abonentów został zaimplementowany mechanizm powodujący, iż w danym momencie czasu jeden i tylko jeden abonent ma dostęp do wysyłania informacji na medium. Przypadek taki został praktycznie przetestowany (załączniki VI.A.1., VI.C).



Rys. 26 Przypadek sieci Ethernet z abonentami posiadającymi warstwę kontrolującą dostęp do medium

2. Sieć realizuje wymiany w sposób niezdeterminowany czasowo.

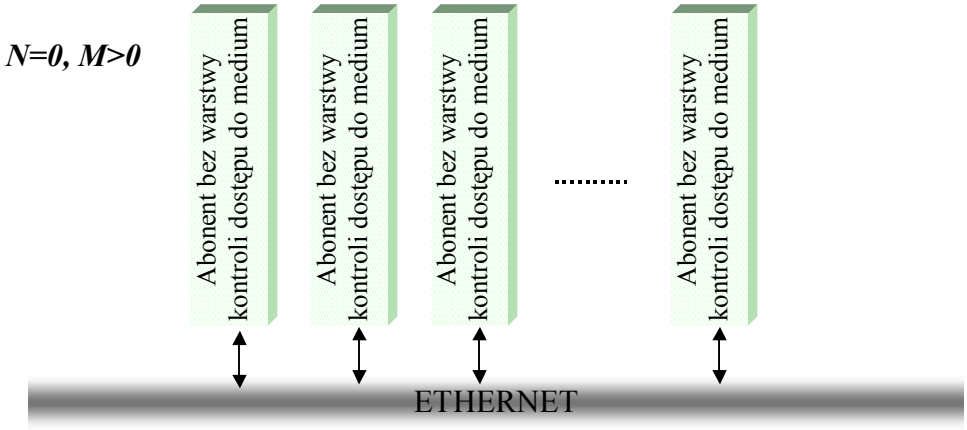
Dla tego przypadku istnieje grupa abonentów działająca tak jak w przypadku pierwszym lecz oprócz nich znajduje się grupa abonentów nie podlegająca działaniu mechanizmu zarządzania dostępem. Powoduje to, iż wymiany przestają mieć charakter zdeterminowany w czasie. Pakiety pochodzące od abonentów bez warstw nadzorczych, mogą pojawiać się w dowolnych i nieprzewidywalnych momentach czasu. Charakter takiego ruchu na sieci był testowany a opis i wyniki zamieszczono w załącznikach VI.A.1., VI.C.



Rys. 27 Przypadek sieci Ethernet z różnymi typami abonentów

3. Sieć realizuje wymiany w sposób klasyczny.

Przypadek, gdy *N*=0 stanowi typowe rozwiązanie komunikacyjne z wykorzystaniem standardowych mechanizmów sieci Ethernet. Ze względu na występowanie kolizji i działanie mechanizmu CSMA/CD przypadek taki należy traktować jako nie gwarantujący dostarczenia danych w sposób zdeterminowany czasowo.



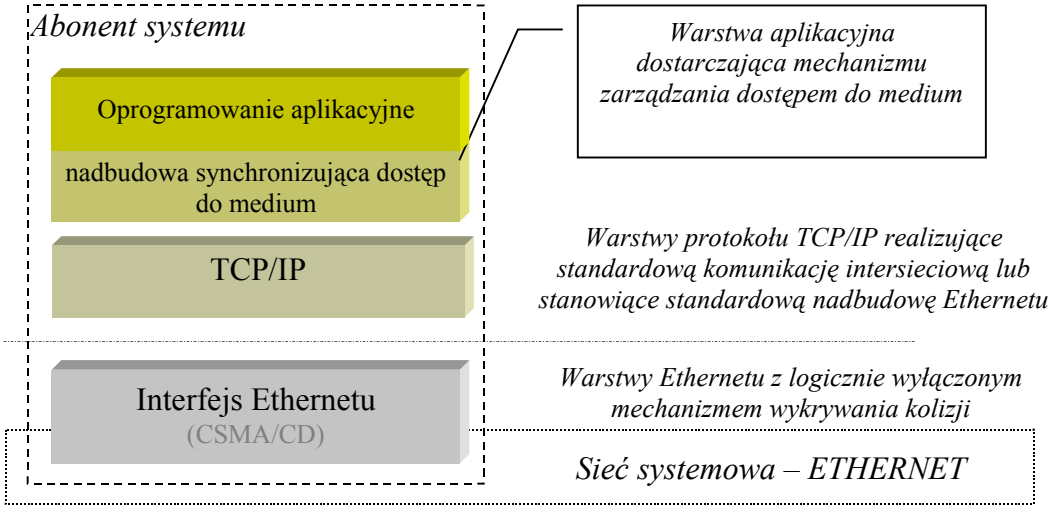
Rys. 28 Klasyczny przypadek pracy sieci Ethernet

Zagadnienia tworzenia nadbudowy aplikacyjnej będą jeszcze poruszane w dalszej części pracy (rozdział 8).

6.5. Wykorzystanie TCP/IP w sieci ETHERNET

Wykorzystywanie protokołu TCP/IP do przesyłania informacji przez intersieć jest oczywiste ze względu na fakt, iż stanowi on obecnie jedyne uniwersalne i standardowe rozwiązanie komunikacyjne w sieciach heterogenicznych. W środowisku sieci lokalnej, gdy mamy do czynienia z siecią Ethernet, rodzaj wykorzystywanego protokołu transportowego nie jest już taki oczywisty, gdyż istnieje wiele protokołów pracujących z wykorzystaniem Ethernetu w warstwach niższych. Przykładem mogą być: NetBEUI, IPX/SPX/NetBIOS, AppleTalk, DLC i inne. Mimo tego zastosowanie protokołu TCP/IP w sieci Ethernet jest możliwe i powszechnie stosowane. Decydują o tym głównie aspekty powszechności, popularności i dobrej standaryzacji protokołów TCP/IP.

Protokół TCP/IP nie wnosi warstw umożliwiających zarządzaniem dostępu do medium zgodnie z którymkolwiek z deterministycznych modeli wymian (zob. strona 21). Zatem z poziomu wcześniejszych rozważań dotyczących możliwości utraty pakietu i czasu opóźnień wprowadzenie protokołu TCP/IP nie zmieni sposobu obsługi informacji przez łącze. Aby zmienić ten sposób należy tak jak dla przypadku sieci Ethernet zastosować dodatkową warstwę porządkującą dostęp do medium. Warstwa ta musi się znajdować ponad warstwami Ethernetu (rys. 29).



Rys. 29 System kontrolny na bazie Ethernetu i TCP/IP z kontrolą dostępu do medium

Najlepszą lokalizacją dla implementacji tej warstwy jest warstwa aplikacji według modelu ISO/OSI (strona 27). Na sposób działania warstwy zarządzającej nie ma wpływu fakt obecności protokołu TCP/IP w interfejsie abonenta. Chyba, że w stosie TCP/IP używane są protokoły generujące samoczynnie wymiany sieciowe z pominięciem tejże warstwy zarządzającej. Aktywacja protokołów samodzielnie generujących ruch sieciowy może zaburzać cykl deterministyczny narzucany przez warstwę aplikacji. Warunkiem koniecznym i wystarczającym deterministycznej pracy protokołu jest istnienie nadrzędnej warstwy kontrolującej wymiany oraz brak wymian generowanych poza kontrolą tej warstwy. Zakłócenia generowane przez protokoły niezależne (np. ARP) mogą być ograniczone, jeżeli protokoły te działają tylko przez określony czas w celu osiągnięcia przez interfejs jakiegoś stanu, np. zbudowania tablic adresowych. W lokalnych sieciach przemysłowych (systemowych) liczba abonentów jest stała i po ustaleniu jej struktury, w normalnej pracy, nie zachodzi potrzeba ciągłego odpytywania o powiązania adresów Ethernetowych z adresami IP. Wystarczy zatem, aby przykładowy mechanizm ARP działał tylko po załączeniu abonentów. Gdy każdy z abonentów sieci ustali swoje lokalne tablice powiązań, protokół ARP staje się niepotrzebny aż do wystąpienia awarii lub modyfikacji struktury sieci.

Konieczność implementacji mechanizmu kontroli wymian zgodnego z jednym z trzech deterministycznych modeli wymian lub ich mutacją stanowi wymóg dotyczący sposobu funkcjonowania warstwy nadrzędnej (zob. strona 21). Wybór konkretnego modelu jest sprawą wtórną i nie ma znaczenia z punktu widzenia zapewnienia determinizmu. Jednak, jak zostanie

pokazane w rozdziale 11, ma wpływ na uzyskiwane parametry sprawności i przepustowości sieci.

Podsumowując, zastosowanie protokołów transportowych TCP i UDP oraz protokołu IP wraz z nadbudową aplikacyjną opisaną w rozdziale 6.4 w interfejsach sieciowych wszystkich abonentów danej sieci Ethernet gwarantuje uzyskanie zdeterminowanej czasowo pracy tych protokołów. Dowodzi to słuszności tezy drugiej. Wykorzystując stos TCP/IP należy przeanalizować jego składowe w celu eliminacji lub modyfikacji tych protokołów, które mogą samoczynnie generować wymiany sieciowe.

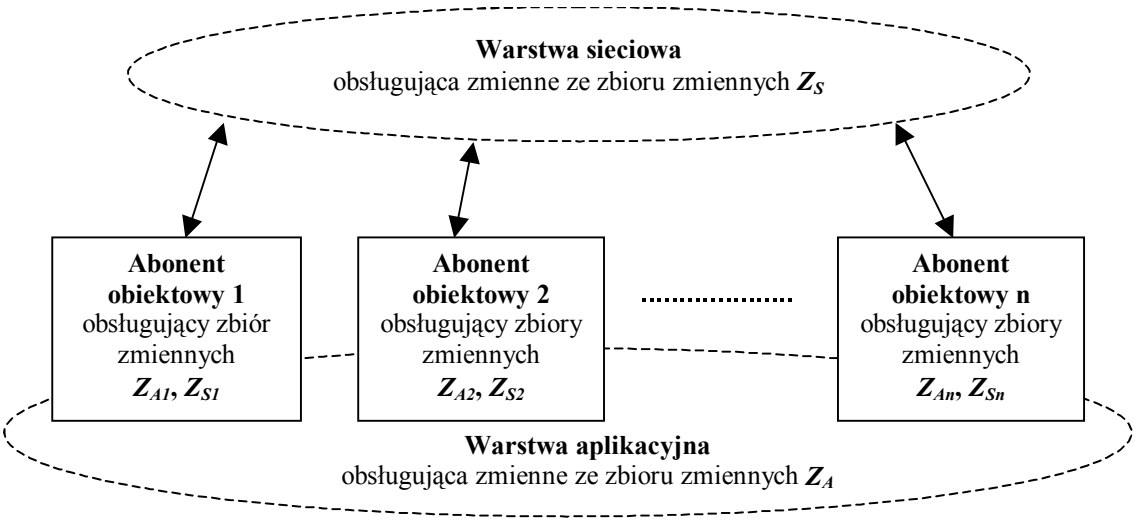
Na potrzeby testów związanych z niniejszą pracą wykonano nadbudowę bazującą na modelu wymian PDC [60, 61, 114] (zob. dodatki VI.A.1., VI.C), gwarantującym uzyskanie najlepszych wartości sprawności i przepustowości.

7. Współpraca podsystemów lokalnego i zdalnego

Protokół TCP/IP został zaprojektowany z myślą o uniwersalnym otwartym mechanizmie przenoszenia danych w heterogenicznym środowisku intersieciowym. Otwartość daje duże nadzieje co do jego możliwości adaptacyjnych, jednak w związku ze swoim uniwersalizmem, zawiera szereg protokołów zupełnie zbędnych w typowych systemach przemysłowych. Nadmiarowość funkcjonalna protokołu generalnie nie szkodzi. Z części protokołów można nie korzystać lub wykorzystywać je nie w pełni. Negatywny wpływ może się jedynie objawić po przez zwiększenie narzutu czasowego związanego z transmisją i obsługą danych pochodzących od tych protokołów. Zostało to pokazane w rozdziale 11.2. Dodatkowe problemy pojawiają się, gdy w rozproszonym systemie przemysłowym należy obsłużyć informację w czasie rzeczywistym [73]. Analizując charakterystykę ruchu [57], dostępność informacji w czasie [53, 50], sprawność oraz przepustowość [60, 61], można stwierdzić, iż uzyskanie determinizmu czasowego zależy od obszaru pracy protokołu TCP/IP, zatem należy rozgraniczyć jego zastosowanie w zależności od rodzaju systemu, w którym ma on pracować oraz sposobu współpracy pomiędzy podsystemami.

7.1. Globalizacja abonentów i podział zmiennych

Na rysunku 30 przedstawiono schemat informatycznego systemu przemysłowego wykorzystującego sieć komputerową do komunikacji pomiędzy abonentami (rozdz. 4.2).



Rys. 30 Lokalny przemysłowy system kontrolny z warstwowym podziałem zmiennych

Przyjęto, iż jest to system lokalny (rozdz. 6.1, 6.2) wykorzystujący zmienne v_{Ai} ze zbioru zmiennych Z_A, gdzie $i = 1 .. p$, $p < \infty$ oraz liczebność zbioru Z_A wynoszącą:

$$l_A = p. \qquad (13)$$

Każdy z n abonentów obsługuje k_j zmiennych v_{Aj}, gdzie $j = 1 .. n$. Pod względem zakresu obsługi zmiennych przez obiekty można wyodrębnić dwie grupy informacji krążących w systemie.

Pierwsza z nich to zbiory zmiennych pracujące w warstwie aplikacji. Zbiory Z_{Aj} stanowią zbiory zmiennych obsługujących informację użyteczną poszczególnych abonentów.

$$Z_{Aj} = \{l_{Aj}; v_{Aj\,k_j} \in Z_A : 0 < k_j \le p\}, \tag{14}$$

gdzie

l_{Aj} – liczebność zbioru Z_{Aj}.

Suma zbiorów Z_{Aj} daje zbiór zmiennych obsługujących informację użyteczną warstwy aplikacyjnej systemu lokalnego (Z_A):

$$Z_A = \{l_A; V_{Ap} : 0 < p \le \infty\}, \tag{15}$$

gdzie

l_A – liczebność zbioru Z_A.

Zmienne obsługujące tę grupę nazwano zmiennymi aplikacyjnymi.

Pomiędzy abonentami systemu lokalnego a obiektem komunikacyjnym w tym systemie zachodzi wymiana danych. Wymiany te nazwano wymianami lokalnymi. Obsługują one zmienne sieciowe v_S stanowiące drugą grupę informacji krążących w systemie (Z_S)

$$Z_S = \{l_S : V_{Sg}; 0 < g \le l_A\}, \tag{16}$$

gdzie

l_S – liczebność zbioru Z_S.

Każdy z abonentów produkuje sieciowo zmienne ze zbioru:

$$Z_{Sj} = \{l_{Sj}; V_{Sj\,h_j} \in Z_S : 0 < h_j \le g\}, \tag{17}$$

gdzie

l_{Sj} – liczebność zbioru Z_{Sj}.

Suma zbiorów Z_{Sj} daje zbiór zmiennych obsługujących informację użyteczną warstwy sieciowej systemu lokalnego (Z_S).

Pomiędzy zmiennymi aplikacyjnymi a zmiennymi sieciowymi zachodzi ścisła współzależność. Zmienne sieciowe są konstruowane ze zmiennych aplikacyjnych, gdyż celem pracy warstwy komunikacyjnej jest przekazywanie zmiennych aplikacyjnych między rozproszonymi aplikacjami systemu. Jednak nie zawsze zbiory tych zmiennych są tożsame. W skład zmiennej sieciowej może wchodzić jedna lub wiele zmiennych aplikacyjnych, ale jednocześnie nie wszystkie zmienne aplikacyjne muszą wchodzić w skład zbioru informacji przenoszonej siecią.

Dla systemu przemysłowego można stwierdzić, iż:

$$l_A = \sum_{j=1}^{n} l_{Aj} = const, \tag{18}$$

gdzie:

n – liczba abonentów lokalnych systemu,

Stała liczebność zbioru Z_A stanowi warunek zachowania determinizmu działania systemu. Rozmiar danych wchodzących w skład zbioru Z_A jest również stały, natomiast rozmiar danych w zbiorze zmiennych sieciowych Z_S nie musi być stały, ponieważ grupa informacji przesyłanej siecią składa się z podzbioru informacji użytecznych Z_A oraz informacji pochodzących od narzutu poszczególnych warstw wykorzystywanych protokołów.

$$I_S = I_U + I_P \neq const \tag{19}$$

gdzie:

I_S – liczebność zbioru informacji przesyłanego siecią,

I_U – liczebność zbioru informacji użytecznych,

I_P – liczebność zbioru informacji pochodzących od poszczególnych warstw stosowanych protokołów.

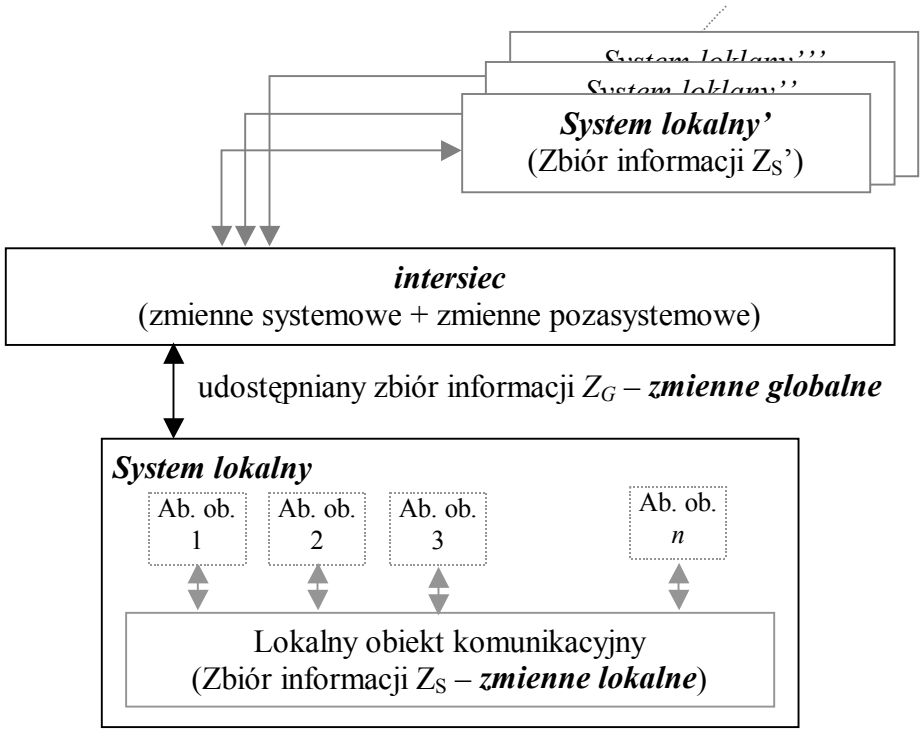
Rozmiar przesyłanych siecią danych określa się w przedziale. Wynika to z faktu, iż w warstwie komunikacyjnej funkcjonuje skończona liczba abonentów a zatem skończona liczba stosów protokołów pracujących na tej warstwie. Dla danego zbioru dostępnych protokołów liczba danych w zbiorze I_P zawiera się w przedziale określonym przez minimum i maksimum.

Dynamiczny charakter rozmiaru danych nie wpływa na liczebność zbioru Z_S, gdyż informacje I_P nie stanowią nowych zmiennych systemowych a jedynie informacje serwisową umożliwiającą transfer danych pomiędzy aplikacjami. Ponadto, ponieważ liczba zmiennych komunikacyjnych może być co najwyżej równa liczbie zmiennych aplikacyjnych oraz liczba ta nie zmienia się w trakcie pracy systemu, można stwierdzić, iż liczebność zbioru Z_S jest również stała:

$$l_S = \sum_{j=1}^{n} l_{Sj} = const\,. \tag{20}$$

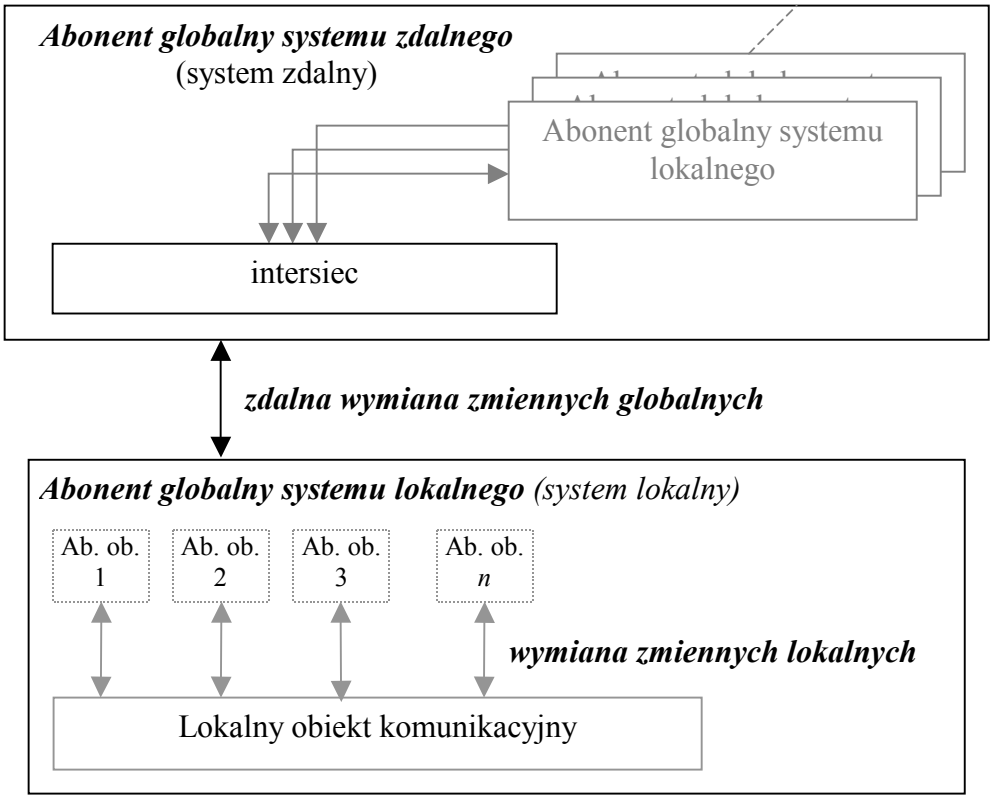
Zmienne ze zbioru Z_A stanowią zmienne wewnętrzne warstw aplikacji systemu kontrolno-nadzorczego i z punktu widzenia warstwy komunikacyjnej nie są istotne, dlatego dalej nie będą brane pod uwagę. Zmienne ze zbioru Z_S stanowią zmienne lokalne lokalnego systemu komunikacyjnego i za ich pomocą realizowane są wymiany wewnętrzne w systemie lokalnym. Zmienne, które są udostępnianie poza system lokalny (określone przez zbiór Z_G oraz liczebność tego zbioru I_G) stanowią zmienne globalne. Zilustrowano to na rysunku 31.

Przekazywanie informacji użytecznej z poziomu zmiennych lokalnych na poziom zmiennych globalnych i odwrotnie, może się odbywać bądź to za pomocą intersieciowych mechanizmów przekazywania danych bądź za pomocą specjalnej warstwy aplikacyjnej, o której będzie mowa w dalszej części pracy.



Rys. 31 Przedstawienie zmiennych lokalnych i globalnych

Dla uproszczenia reprezentacji podsystemów lokalnego i zdalnego abonenci tych podsystemów wraz ze swoimi systemami komunikacyjnymi reprezentowani będą dalej przez abonentów globalnych. Sprowadzenie systemu kontrolnego do obiektów globalnych z zaznaczeniem wymian informacji między obiektami przedstawiono na rysunku 32.



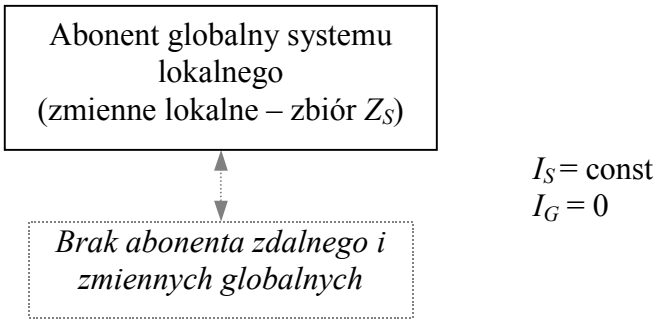
Rys. 32 Obiektowe przedstawienie systemu kontrolnego

Pojawia się zatem dwojaki aspekt wykorzystywania protokołu TCP/IP wspomniany na wstępie rozdziału: w wymianie lokalnej oraz w wymianie zdalnej.

W celu zastosowania protokołu TCP/IP w systemach przemysłowych należy usystematyzować budowę systemów przemysłowych, z punktu widzenia współdziałania lokalnego oraz zdalnego abonenta globalnego. Istnieje co najmniej kilka przypadków takiej współpracy.

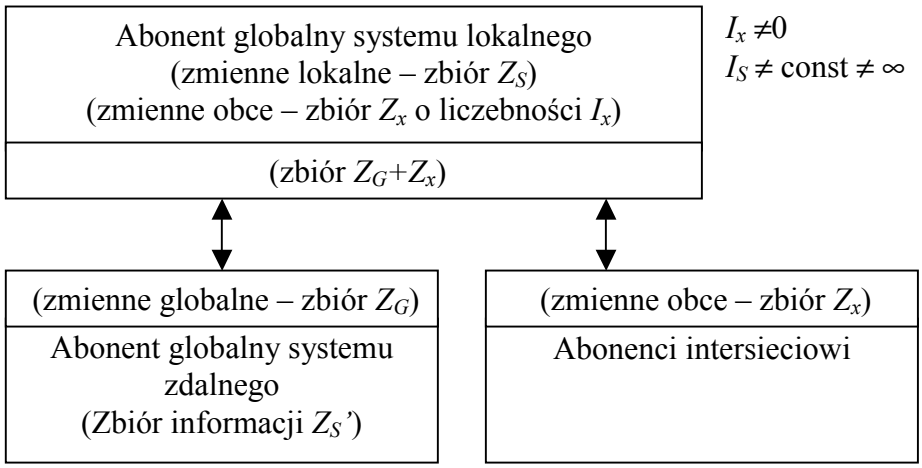
7.2. Rodzaje współpracy

Jeżeli liczba zmiennych lokalnych jest ściśle określona na etapie konfiguracji systemu i nie ulega zmianie w czasie jego pracy, oraz brak jest zmiennych globalnych, to sieć systemowa jest siecią zamkniętą a system ma obieg informacji zamknięty w granicach systemu lokalnego. Czyli gdy I_S = const, I_G = 0, to wówczas otrzymuje się lokalny system przemysłowy z zamkniętym obiegiem informacji (rys. 33).



Rys. 33 Lokalny system przemysłowy z zamkniętym obiegiem informacji

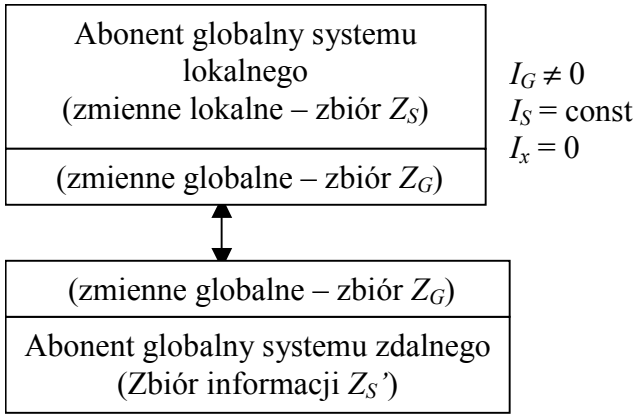
W przypadku przeciwnym, gdy w sieci systemowej krąży informacja nie związana z systemem (Z_x), wówczas sieć i system będzie miał charakter otwarty (rys. 34). Stanowi to przypadek pracy warstwy komunikacyjnej bezpośrednio w intersieci. Określanie liczebności zbioru Z_G jest bez znaczenia, gdyż obsługa sieciowa jest taka sama niezależnie od liczby zmiennych globalnych.



Rys. 34 Lokalny system przemysłowy z otwartym obiegiem informacji

Istnieje również trzeci przypadek, w którym sieć systemowa nie obsługuje zmiennych spoza systemu I_S = const, natomiast liczba zmiennych globalnych jest niezerowa. Wówczas

można mówić o systemie otwartym z kontrolowanym przepływem informacji pomiędzy systemem lokalnym o systemem zdalnym, czyli o lokalnym systemie separowanym (rys. 35).



Rys. 35 Lokalny system przemysłowy z separowanym obiegiem informacji

W następnych podrozdziałach rozważono w sposób bardziej szczegółowy powyższe możliwości.

7.2.1. Praca systemów z zamkniętym obiegiem informacji

Schemat systemu z warstwą komunikacji typu zamkniętego przedstawiono na rysunku 30. W systemach zamkniętych pracę sieci określają parametry, które można określić jako stałe czy też predefiniowane na etapie jej konfiguracji. Biorąc pod uwagę zależność 18, liczebność zmiennych sieciowych w systemie można wyrazić zależnością:

$$L_S = l_S = const\,, \tag{21}$$

gdzie:

L_S – liczebność zbioru zmiennych sieciowych w całym systemie kontrolno - nadzorczym,

Jeśli nie ma zmiennych globalnych, istnieje brak obcego ruchu oraz dynamicznej liczby zmiennych lokalnych, wówczas system stanowi system zamknięty i można zastosować w nim dowolny protokół. Wykorzystanie TCP/IP staje się zatem możliwe.

Jeżeli lokalny abonent globalny nie umożliwia połączenia ze strukturami intersieciowymi, wówczas mechanizmy intersieciowe, a w szczególności protokół IP, są nadmiarowe i stanowią one niepotrzebny balast. Objawia się to zaniżeniem sprawności użytecznej sieci (rozdział 11) [61], gdyż rośnie liczba informacji I_P przy stałej I_U (wzór *31* strona 108).

Rozwiązanie problemu dostępu do medium można przeprowadzić w dwojaki sposób. Można wykorzystać potrzebne warstwy protokołu TCP/IP, a rozwiązywanie konfliktów pozostawić warstwom niższym związanym z konkretną siecią, jak np. mechanizm CSMA/CD dla sieci Ethernet. W podejściu tym protokół pracuje zgodnie z modelem niedeterministycznym. Jeżeli jednak system wymaga determinizmu dostępu, można wówczas nadbudować w warstwie aplikacji mechanizm kontroli oparty o dowolny model kontroli wymian (rozdz. 6.4, 6.5). Oczywiście rozwiązanie takie wprowadza dodatkowe narzuty związane z dodatkową podwarstwą aplikacji.

Istnieje jeszcze jeden przypadek sieci zamkniętych. Są to zamknięte intersieci prywatne czy też korporacyjne. Mamy wówczas do czynienia z środowiskiem heterogenicznym wymagającym protokołu IP oraz wiele domen kolizyjnych praktycznie wykluczających użycie mechanizmu nadzorowania wymian. Pomimo architektury zamkniętej bez swobodnego dostępu publicznego, to ruch w takiej sieci staje się nieprzewidywalny. Należy taką sieć traktować jako sieć otwartą. Wynika to z faktu możliwości pracy wielu systemów informatycznych w jednej strukturze komunikacyjnej oraz możliwości wystąpienia zagrożeń od grupy abonentów i użytkowników tej struktury nie zaangażowanej w działanie systemu przemysłowego. Wydzielenie domeny dla systemu przemysłowego i izolowanie ruchu reszty intersieci stanowi jedyne rozwiązanie umożliwiające uruchamianie transmisji deterministycznej.

Podsumowując, istnieje możliwość wykorzystania protokołu TCP/IP w systemach z zamkniętym obiegiem informacji. Mogą one służyć dla realizacji aplikacji przemysłowych wymagających ograniczeń czasowych i niekorzystających ze współpracy z intersiecią. Wykorzystanie TCP/IP może mieć dwojaki charakter, zarówno zachowując determinizm czasowy dostępu jak i nie. Niezależnie jednak od sposobu, rozwiązanie takie ma niewielką sprawność użyteczną łącza. Zatem, jeżeli nie ma szczególnych przesłanek do stosowania tego protokołu, na przykład ekonomicznych, to należy tego unikać. Lepiej w takich przypadkach dążyć do budowy systemów opartych o specjalistyczne sieci przemysłowe lub bazować tylko na warstwach Ethernetu. Więcej na ten temat znajduje się w rozdziale 11.

7.2.2. Praca systemów z otwartym obiegiem informacji

Schemat systemu opartego o otwarty system komunikacyjny przedstawiono na rysunku 36. System zawiera grupę abonentów lokalnych, abonentów zdalnych generujących dynamiczne zbiory informacji oraz abonentów obcych, nie związanych z systemem.

Zakładając prawdziwość zależności 18 oraz heterogeniczną konstrukcję sieci systemowej, można stwierdzić, że:

$$l_d = \sum_{j=1}^{k} l_{dj} = const\,, \tag{22}$$

gdzie:

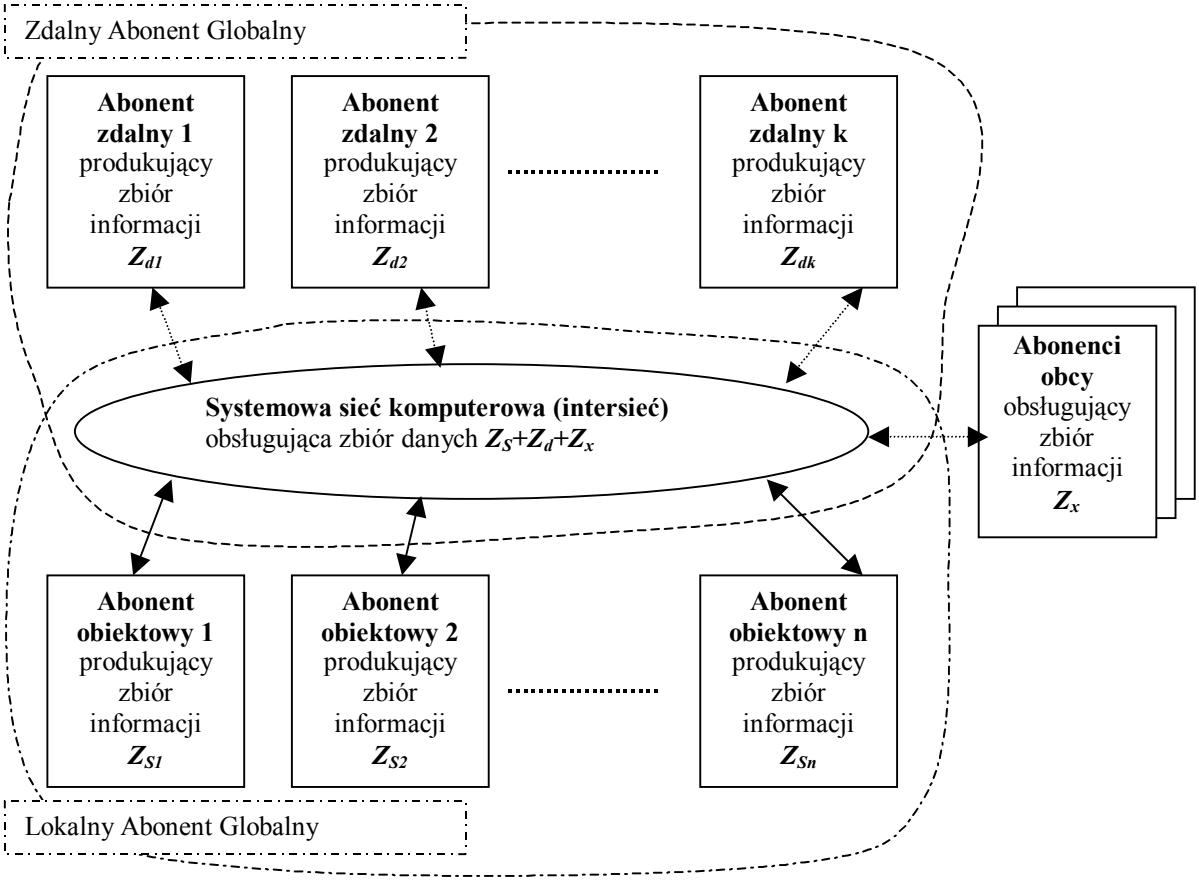
- l_d – liczebność zbioru zmiennych produkowanych sieciowo przez wszystkich abonentów zdalnych,
- l_{dj} – liczebność zbioru zmiennych produkowanych przez konkretnego abonenta zdalnego,
- k – liczba abonentów dołączanych zdalnie do systemu,

oraz

$$L_S = l_S + l_d + l_x \neq const\,, \tag{23}$$

gdzie:

- l_x – liczebność zbioru zmiennych obcych,



Rys. 36 Otwarty system kontrolno-nadzorczy

Z punktu widzenia systemów przemysłowych przypadek taki stanowi rozwiązanie nieprawidłowe, ponieważ w sieci systemowej pojawia się ruch nie związany z danym systemem kontrolnym (zbiór Z_x). Powoduje to, iż w systemie zostaje zakłócone bilansowanie informacji poprzez konieczność obsługi przez lokalny system komunikacyjny zmiennych nie należących do tego systemu. Ruch na sieci wywołany na rzecz zbioru Z_S jest przewidywalny i dla protokołów deterministycznych precyzowany w przedziałach czasu. Natomiast liczebność zbioru wszystkich zmiennych sieciowych stanowi wartość nieprzewidywalną, ze względu na dynamiczną liczebność zbioru Z_x.

Ponieważ następuje współdzielenie sieci systemowej pomiędzy abonentami globalnymi, zachodzi swobodne przekazywanie zmiennych z poziomu globalnego na lokalny i odwrotnie. Nie występuje rozdzielenie zmiennych globalnych od lokalnych, czyli:

$Z_S \cup Z_d = Z_G$.

Zatem można stwierdzić:

$$L_S = l_G + l_x \neq const\,, \tag{24}$$

W tego typu systemach nie ma możliwości zastosowania protokołów wykorzystujących jakikolwiek deterministyczny model nadzoru przepływu informacji. Zarówno model PDC, Master-Slave jak i Token wymagają ścisłego współdziałania wszystkich abonentów. Jest to niemożliwe w systemach otwartych, gdyż dynamiczny charakter Z_x uniemożliwia

współdziałanie abonentów na płaszczyźnie transakcyjnej oraz może powodować zapychanie sieci pakietami obcymi.

System otwarty można zamodelować przy użyciu systemu zamkniętego oraz stałej składowej obciążającej sieć w postaci strumienia danych ze zbioru Z_x. Ruch wywołany na rzecz obsługi sieciowej zmiennych zbioru Z_x stanowi swoisty narzut obciążenia wywołanego na rzecz zmiennych systemowych.

Najodpowiedniejszym protokołem dla systemów typu otwartego jest protokół intersieciowy, czyli TCP/IP. Narzucenie protokołu deterministycznego, gdzie kontrola dostępu realizowana jest na poziomie warstw aplikacji, może być trudna lub niewykonalna. Narzucenie protokołu rozwiązującego dostęp do medium na poziomie warstw niższych może być skuteczne tylko dla segmentów sieci o charakterze zamkniętym.

Budowanie komunikacyjnych systemów otwartych dla potrzeb przemysłowych systemów kontrolno-nadzorczych na bazie protokołów ze zdeterminowanym czasowo dostępem do mediów stanie się możliwe tylko wtedy, gdy protokół intersieci umożliwi gwarantowany dostęp do informacji, a sprzęt integrujący zrealizuje odseparowanie obcych pakietów. Obecnie systemy otwarte mogą służyć tylko dla realizacji aplikacji nie wymagających ograniczeń czasowych i nie mających wymagań odnośnie bezpieczeństwa dostępu. Mimo, iż z punktu widzenia poszerzenia zakresu stosowalności takich systemów konieczny jest rozwój protokołu IP, transmisja zmiennych systemowych przez obszary sieci, nad którymi użytkownik systemu nie sprawuje nadzoru zawsze stanowić będzie potencjalne zagrożenie. Dlatego systemy otwarte nie powinny być aplikowane dla systemów przemysłowych.

7.2.3. Praca systemów z separowanym obiegiem informacji

Schemat systemu opartego o dwa rozdzielone podsystemy komunikacyjne przedstawiono na rysunku 37. System zawiera grupę abonentów lokalnych, grupę abonentów pośredniczących obsługujących stałe zbiory informacji na sieci wewnętrznej oraz abonentów zdalnych pracujących w sieci zewnętrznej.

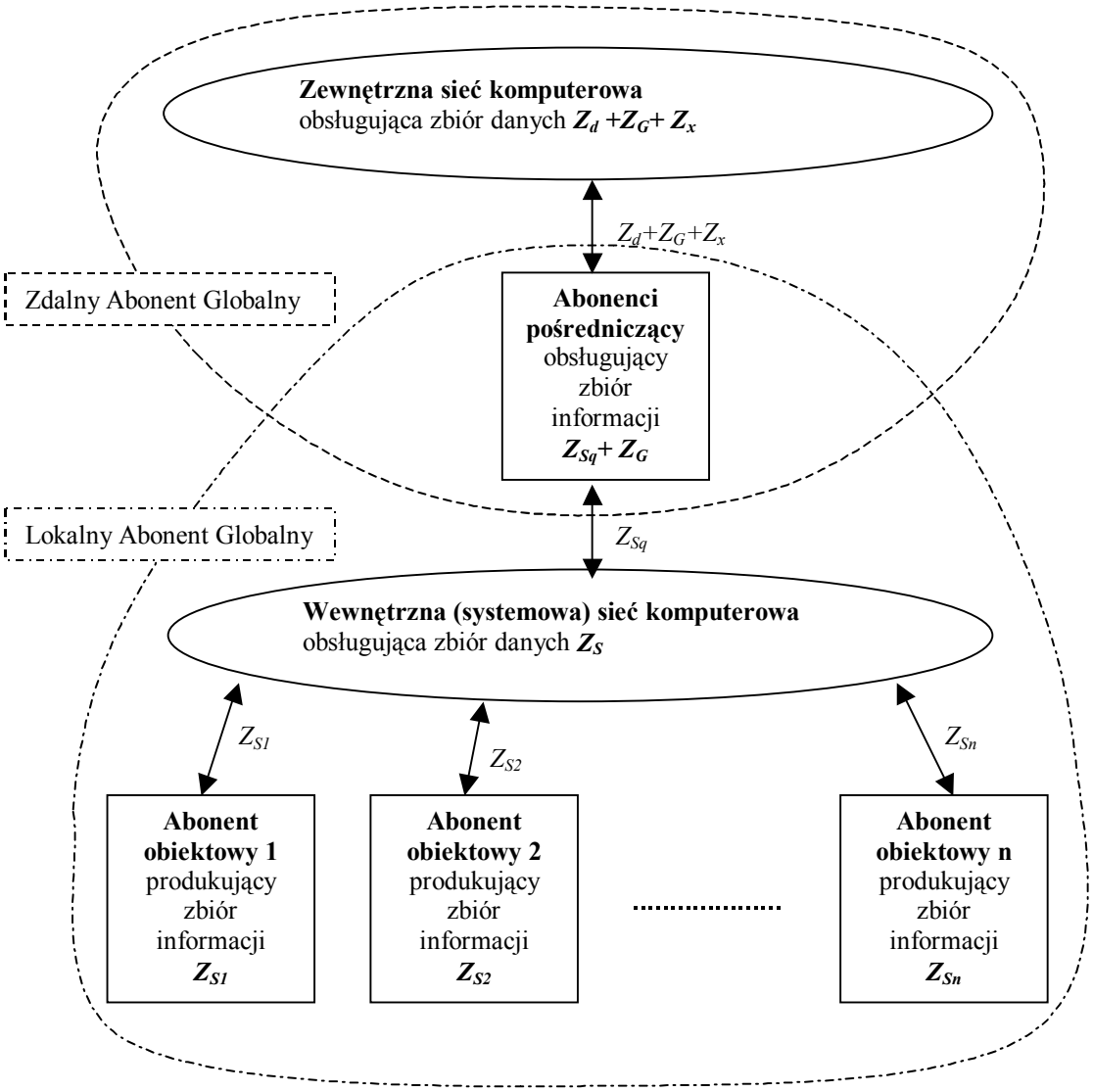
Tak jak dla wcześniejszych przypadków (wzory 18, 20, 22):

$$l_S = const\,, \tag{25}$$

$$l_d = const\,. \tag{26}$$

Zbiór zmiennych globalnych Z_G stanowi część wspólną zbioru Z_d oraz Z_S ($Z_G = Z_d \cap Z_S$) przekazywaną pomiędzy lokalnym abonentem globalnym a zdalnym abonentem globalnym.

W systemie wykorzystującym odizolowanie wymiany zmiennych lokalnych i globalnych proponuje się zastosowanie specjalnego abonenta lub grupy abonentów realizujących fizyczną separację sieci oraz bramę aplikacyjną dla zmiennych obcych. Abonent ten pracuje pomiędzy siecią systemu lokalnego a sieciami systemu zdalnego, w szczególności intersiecią. Następuje wówczas współdzielenie abonenta pośredniczącego przez abonentów globalnych a nie współdzielenie sieci jak to miało miejsce dla systemów otwartych.



Rys. 37 System kontrolno - nadzorczy z rozseparowanymi obiegami informacji

W sieci systemowej może pracować dowolny protokół, tak jak dla przypadku sieci zamkniętych. Wówczas informacje przychodzące z zewnątrz muszą być po pierwsze filtrowane a po drugie przeadresowane. Filtracja jest konieczna ze względu na odrzucanie pakietów obcych i nie przekazywanie danych nie związanych z systemem do obiegu wymian lokalnych. Przeadresowanie jest niezbędne ze względu na fakt, iż liczba zmiennych lokalnych w przeciwieństwie do globalnych jest skończona. Abonent separujący, z punktu widzenia podsystemu komunikacyjnego abonenta globalnego, logicznie stanowi abonenta globalnego współpracującego z tym podsystemem. W systemach przemysłowych, gdy protokoły sieci lokalnych abonentów globalnych są różne od protokołów zdalnych abonentów globalnych mamy do czynienia ze specyficznym przypadkiem integracji sieci heterogenicznych, gdzie celem nadrzędnym musi stać się sposób przekazywania informacji względem czasu rzeczywistego a nie sama translacja protokołów. Więcej na temat warstw separujących znajduje się w rozdziale 8.

7.3. Wybór rozwiązania optymalnego

Analizując omówione powyżej trzy przypadki konstrukcji systemu kontrolnego, jako najlepszy do dalszych rozważań, wybrano system otwarty z separowanym obiegiem informacji. Główne kryteria takiego wyboru to:

- zapewnienie determinizmu czasowego dostępu do danych w systemie lokalnym,
- zapewnienie możliwości zdalnego dostępu,
- maksymalizacja bezpieczeństwa dostępu do danych systemu lokalnego.

System z separowanym obiegiem informacji spełnia wszystkie powyższe kryteria. Determinizm czasowy można uzyskać poprzez wykorzystanie warstw aplikacyjnych kontrolujących wymiany dla Ethernetu lub stosu TCP/IP oraz ewentualnie przez stosowanie protokołów specjalizowanych na poziomie systemu lokalnego. Realizacja zdalnego dostępu staje się możliwa przez abonenta pośredniczącego wykonującego przekazywanie danych użytecznych ze zmiennych globalnych do zmiennych lokalnych i odwrotnie, niezależnie od tego, jaki protokół pracuje w systemie lokalnym. Dodatkowo abonent pośredniczący dba o bezpieczeństwo dostępu do zmiennych lokalnych chroniąc system lokalny przed niepowołanym odczytem lub zapisem.

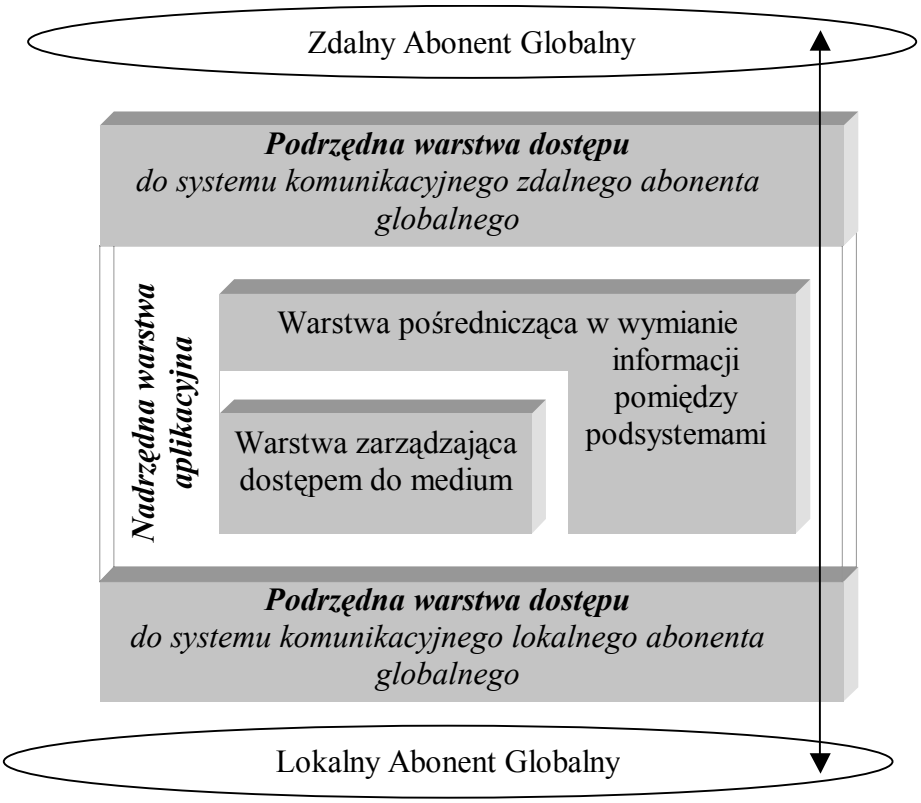
Zalety tak skonstruowanego systemu to przede wszystkim:

- zachowanie cech zdeterminowanego w czasie łącza komunikacyjnego systemu lokalnego (6.4, 8),
- duża przepustowość łącza komunikacyjnego systemu lokalnego (11.2),
- atrakcyjność ekonomiczna,
- możliwość uzyskania zdalnego dostępu (10),
- standaryzacja zdalnego dostępu (10),
- możliwość standaryzacji łącza komunikacyjnego systemu lokalnego (6.5),
- możliwość integracji z określaniem jakości usług (9.1.2, 9.1.3, 9.1.4),
- wysoki poziom bezpieczeństwa zdalnego dostępu (10.4),
- brak wpływu pracy abonentów globalnych na siebie (8, 8.2).

8. Budowa warstwy aplikacyjnej

Warstwa aplikacyjna stosu protokołów, niezależnie od przyjętego modelu opisu (rozdz. 5), jest umiejscowiona na szczycie tego stosu. Zostało to pokazane na rysunku 14. Klasyczna warstwa aplikacji odpowiada za współpracę programu aplikacyjnego użytkownika z interfejsem komunikacyjnym. W celu wprowadzenia mechanizmu nadrzędnej kontroli wymian opisanego w rozdziale 6.4, niezbędna jest modyfikacja funkcji warstwy aplikacyjnej przez dodanie mechanizmów uzależniających transmisję od scenariusza wymian a nie od pracy warstwy aplikacji użytkownika.

Wprowadzenie takiej funkcji do warstwy najwyższej jest podyktowane wygodą jej implementacji. Im niższa warstwa stosu protokołów interfejsu sieciowego tym trudniej ingerować w jej budowę i powiązania. W praktyce możliwe jest nawet implementowanie funkcji nadzorczych w warstwie aplikacji użytkownika, przy założeniu, że tylko ona korzysta z interfejsu.



Rys. 38 Budowa warstw deterministycznego interfejsu komunikacyjnego abonenta sieci Ethernet

Budowa sprzętu (zastosowane koprocesory, konstrukcje pamięci, magistral itp.) abonenta pośredniczącego w wymianie zmiennych pomiędzy abonentami globalnymi nie wpływa na logiczną konstrukcję stosu protokołu i nie rozważa się jej w niniejszym rozdziale. Kluczowy

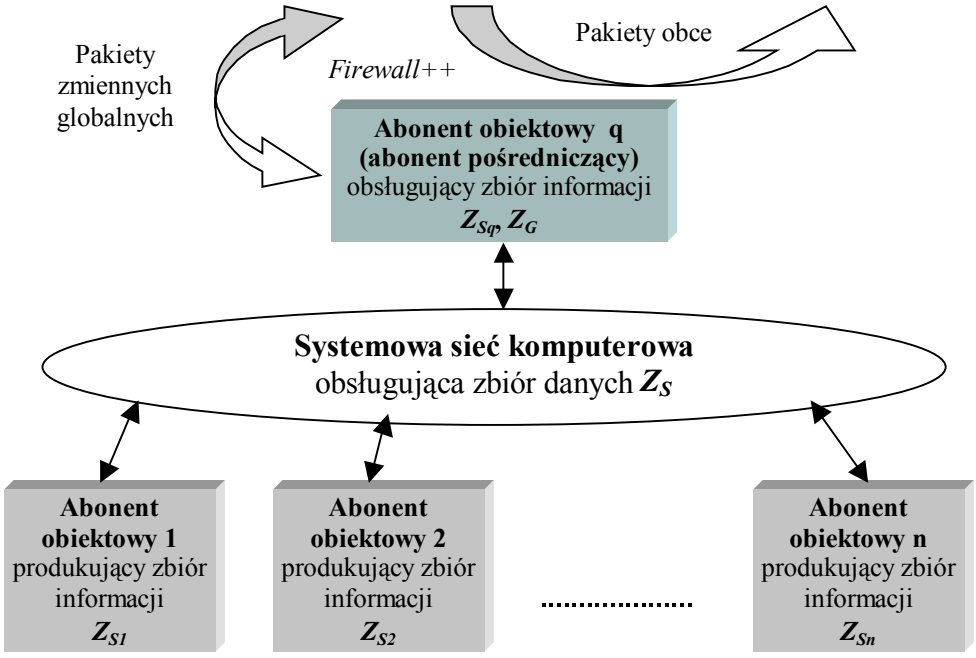
problem stanowi budowa warstwy aplikacyjnej stanowiącej nadrzędną warstwę spełniającą funkcje opisane w rozdziałach 6.4, 6.5, 7.2.3 i do działania której odwołuje się pierwsza teza pracy. Działanie warstwy nadrzędnej musi opierać się na funkcji:

1. kontrolującej dostęp do medium lokalnej sieci systemowej,
2. realizującej zdalny dostęp do systemu lokalnego.

W celu zapewnienia determinizmu czasowego w systemie lokalnym warstwa kontrolująca dostęp musi być zaimplementowana w interfejsie każdego abonenta tego systemu. Warstwa druga jest niezbędna tylko dla abonentów pośredniczących w wymianie informacji użytecznej pomiędzy systemem lokalnym a systemem zdalnym. Zostało to przedstawione na rysunku 38.

Warstwa zarządzająca musi pracować realizując scenariusz wymian zgodnie z jednym z deterministycznych modeli wymian (strona 21). Jej działanie jest pochodną działania tych modeli. Na potrzeby niniejszej pracy i związanej z nią testów wykonano nadbudowę w oparciu o model PDC. Szczegóły dotyczące tej implementacji zostały przedstawione w załączniku VI.A.1., VI.C.

Warstwa separująca pod względem realizowanych funkcji będzie przypominała działanie tzw. ściany ogniowej (ang. *firewall*). Ideę tą przedstawiono na rysunku 39. Istnieją komercyjne rozwiązania pakietów typu „firewall" dostarczane dla systemów przemysłowych. Przykładem może być urządzenie Firewall FWA-230 firmy Advantech, gdzie na specjalizowanym komputerze pracuje system operacyjny Linux RedHat oraz oprogramowanie Check Point™ FireWall-1® lub Check Point™ VPN-1®. Są to jednak urządzenia, w których pakiety kontrolujące dostęp działają jak pakiety standardowe bez dodatkowych funkcji uwzględniających specyfikę ruchu w sieciach przemysłowych.



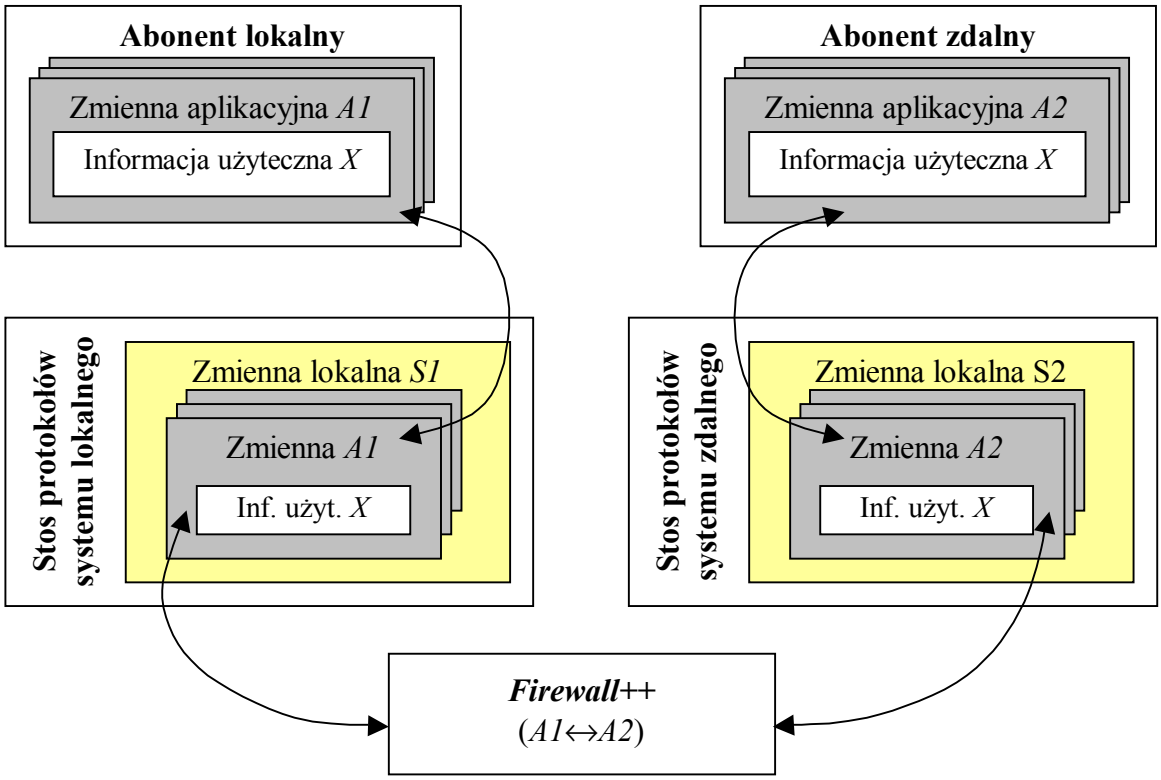
Rys. 39 System lokalny z abonentem *Firewall++*

Poza klasyczną funkcją filtrowania pakietów warstwa ta musi wykonywać dodatkowe funkcję mające na celu wprowadzenie zmiennych systemu komunikacyjnego zewnętrznego

do systemu wewnętrznego bez zakłócania tego drugiego (zob. 7.2.3). Proponuje się nazywać abonenta posiadającego taką warstwę separującą terminem *Firewall++* [33].

Na rysunku 39 zbiór zmiennych Z_{Sq} jest obsługiwany przez system komunikacyjny zgodnie z zasadami pracy sieci wewnętrznej i z punktu widzenia systemu lokalnego są to informacje produkowane i konsumowane przez abonenta *q*. W rzeczywistości w skład tej grupy, oprócz informacji związanej tylko z abonentem *q*, wchodzi informacja pochodząca od abonentów sieci zewnętrznej, czyli ze zmiennych globalnych.

Najprostszy przypadek przekazywania zmiennych istnieje wówczas, gdy każda zmienna ze zbioru Z_G ma swój odpowiednik w zbiorze Z_{Sq} i odwrotnie, ($Z_G = Z_{Sq}$). Ponieważ jednak liczba fizycznych abonentów systemu zdalnego nie jest określona, zatem liczba zmiennych globalnych może się dynamicznie zmieniać. Musi zatem istnieć grupa zmiennych lokalnych, która służyć będzie do przesyłania informacji użytecznej spoza systemu oraz muszą istnieć zmienne aplikacyjne funkcjonujące zarówno w lokalnym abonencie globalnym jak i w zdalnym abonencie globalnym (zob. rozdz. 7.1). Są to globalne zmienne aplikacyjne. Zmienne takie muszą zawierać w sobie adres aplikacyjny, a abonenci obsługujący takie zmienne mechanizm obsługi tego adresu. Adresacja na poziomie aplikacyjnym musi istnieć, aby możliwe było przekazanie między abonentami lokalnymi i zdalnymi, dowolnej informacji użytecznej, identyfikowalnej na poziomie warstw aplikacji. Proponuje się zatem przekazywanie systemowej informacji użytecznej za pomocą protokołu sieci systemowej, której wymiana zachodzi pomiędzy abonentami obu sieci.



Rys. 40 Przekazywanie informacji użytecznej pomiędzy zmiennymi

Na rysunku 40 przedstawiono schemat przekazywania danych użytecznych przy wykorzystywaniu zmiennych aplikacyjnych oraz sieciowych. Zmienne sieciowe *S1* i *S2* są

różne pod każdym względem. Zawierają inny zestaw zmiennych aplikacyjnych, inne adresy sieciowe, posiadają różny rozmiar. Zmienne aplikacyjne *A1* i *A2* z punktu widzenia implementacji są różne, gdyż różna może być platforma implementacyjna. Jednak z punktu widzenia logicznego są to te same zmienne stanowiące aplikacyjne zmienne globalne. Obsługują one jedną i tą samą informację użyteczną *X*. W skali całego systemu kontrolnego globalne zmienne aplikacyjne posiadają taki sam identyfikator stanowiący unikalny adres informacji użytecznej. Zadanie przeadresowania spoczywające na *Firewallu++* polega na skojarzeniu takiej pary zmiennych sieciowych, aby obie zawierały tę samą zmienną aplikacyjną *A* identyfikowaną przez unikalny identyfikator i przepisaniu informacji użytecznej z jednej do drugiej zmiennej sieciowej *S*. Przy użyciu lokalnych i zdalnych systemów komunikacyjnych oraz *Firewalla++* następuje stworzenie tunelu do transmisji zmiennych aplikacyjnych pomiędzy abonentami lokalnymi i zdalnymi, w którym zachodzi przekazywanie informacji użytecznej. Cechę charakterystyczną takiego tunelu stanowi zapewnienie zdeterminowanego w czasie dostępu do zmiennych sieciowych w tej jego części, która przebiega przez system lokalny, pomimo jednoczesnej realizacji niezdeterminowanego w czasie przesyłu globalnych zmiennych aplikacyjnych.

Opisane powyżej działanie mechanizmów przekazywania danych modułu *Firewalla++* dowodzi poprawności tezy trzeciej w punkcie a i b.

8.1. Przypadki zestawiania warstw podrzędnych

Przy separacji systemu lokalnego i zdalnego można stosować szereg kombinacji protokołów z wykorzystaniem stosu TCP/IP.

Pierwszy charakterystyczny przykład stanowi kombinacja systemu zdalnego na bazie sieci Internet z protokołem TCP/IP oraz systemu lokalnego opartego o standard Ethernet również ze stosem TCP/IP. Przypadek ten jest przypadkiem najprostszym. Odfiltrowanie pakietów IP polega na odrzuceniu pakietów, których adresy nie odnoszą się do interfejsów sieci systemowej. Wystarczy zastosować klasyczne mechanizmy filtrowania stosowane w urządzeniach typu ściana ogniowa. Przeadresowanie zmiennych dla przypadku, gdy $Z_G = Z_{Sq}$ polega na zamianie adresacji IP widzianej na zewnątrz na adresację systemu lokalnego i odwrotnie. W sytuacji, gdy równość między zbiorami nie zachodzi należy dodatkow7o przeadresowywać zmienne aplikacyjne. System może pracować zarówno z wykorzystaniem mechanizmów determinizmu czasowego jak i bez niego. W przypadku stosowania kontroli wymian niezbędne staje się zaimplementowanie na stacji pośredniczącej mechanizmu buforowania pakietów. Dostęp do zmiennych lokalnych jest wówczas deterministyczny natomiast dostęp do zmiennych aplikacyjnych przenoszonych przez te zmienne lokalne nie ma charakteru zdeterminowanego w czasie. Pojawia się niespójność czasowa pomiędzy cyklem pracy sieci a cyklem pracy aplikacji. Wymusza to stosowanie mechanizmów określania takich niespójności, opisanych w rozdziale 9.1.

Drugi przypadek to system zdalny z intersiecią oraz system lokalny oparty o specjalizowany protokół przemysłowy. W tym przypadku abonenci pośredniczący muszą

spełniać funkcję bram (ang. *gateway*). Poza filtrowaniem ruchu pakietów, muszą wykonać konwersję protokołów łącznie z koniecznością adresowania danych, tak jak w przypadku wcześniejszym. Komplikuje to budowę stacji pośredniczącej, lecz zyskujemy wówczas w pełni wydajny przemysłowy system komunikacyjny z możliwością współpracy z intersiecią. Mechanizm pośredniczący *Firewalla++* może być zrealizowany na bazie warstw komunikacyjnych stacji typu SCADA [18, 23, 38], lub jako specjalizowana aplikacja pracująca na wybranym abonencie. Aplikacja modułu *Firewalla++* na stacji SCADA stanowi wygodne rozwiązanie z punktu widzenia implementacji, gdyż większość systemów przemysłowych taką stację posiada, i z reguły łatwo ją dostosować do pełnienia takiej funkcji. Na potrzeby niniejszej pracy testowano połączenia wykorzystując stację Kronos [23, 48, 120] oraz moduł *Firewalla++* dla sieci WorldFIP oraz testowego protokołu UDP/PDC (załącznik VI.A).

Kolejne rozwiązanie stanowi wykorzystanie protokołu TCP/IP do połączenia dwóch systemów pracujących na bazie protokołów przemysłowych po przez intersieć. Najprostszym rozwiązaniem tego problemu będzie tunelowanie ramek protokołów przemysłowych w kanale transmisji IP. Oczywiście wszelkie problemy determinizmu w dostępie do wartości tunelowanych zmiennych nie zanikają, i należy je rozwiązywać na poziomie warstw aplikacji.

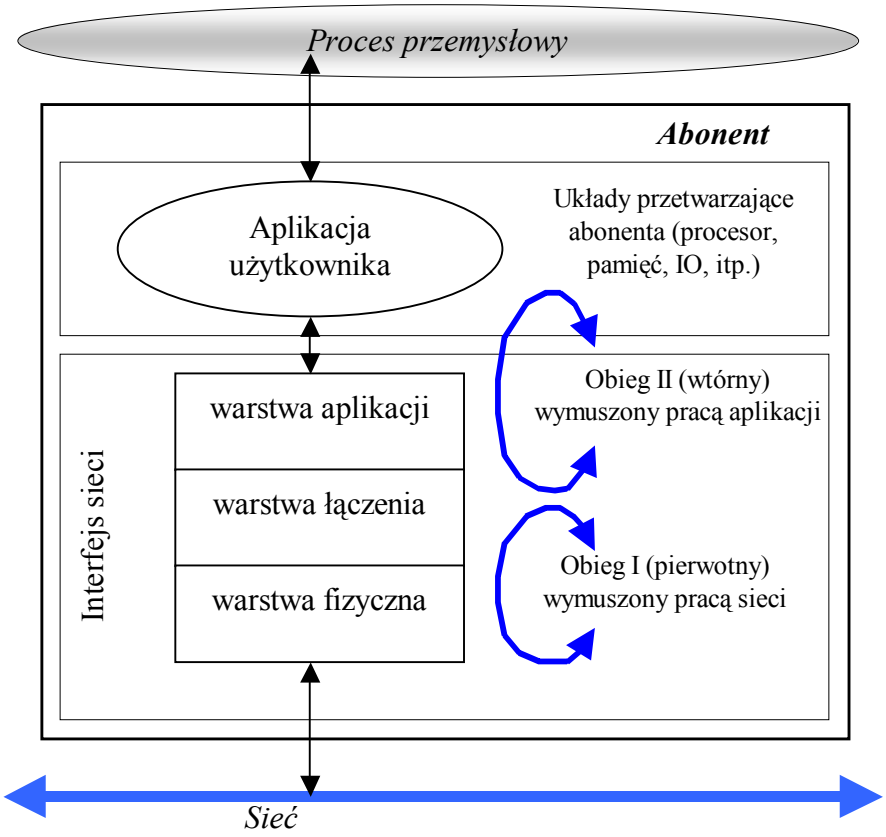
8.2. Wewnętrzne obiegi informacji w interfejsach komunikacyjnych

Konstrukcja warstw nadrzędnych ma związek z przepływem informacji w interfejsach komunikacyjnych. W każdej sieci pomiędzy medium a warstwą aplikacji funkcjonują dwa niezależne obiegi informacji. Zapis wartości informacji wykonywany z poziomu aplikacji użytkownika jest zapisem wykonywanym do warstw niższych i nie zależy od pracy sieci, lecz od pracy samej aplikacji użytkownika. Natomiast zapisy warstwy fizycznej na sieć wykonywane są niezależnie od stanu warstwy aplikacji generującej wartość i przebiegają zgodnie z regułami mechanizmu warunkującego dostęp do medium. Zilustrowano to na rysunku 41.

W interfejsach sieciowych zawsze istnieje jakiś mechanizm nadzorujący dostęp do medium. Mechanizm ten wymusza pierwotny obieg informacji związany bezpośrednio z pracą sieci. Dla sieci Master-Slave cykl pracy tego obiegu wymusza stacja Master, dla sieci PDC arbiter, dla sieci z żetonem sam żeton, dla sieci Ethernet mechanizm CSMA/CD itd. Obieg II, który można nazwać wtórnym, jest wymuszany przez warstwę aplikacji użytkownika. Zależy on od cyklu pracy aplikacji, wektora stanu procesu użytkownika oraz innych czynników wpływających na pracę warstw wyższych.

Podstawowy wymóg deterministycznego charakteru pracy interfejsu sieciowego stanowi zsynchronizowanie zdarzeń w obiegach I i II. Istnieją aplikacje wykonujące taką synchronizację, umożliwiając pracę aplikacji w czasie rzeczywistym pracy sieci. Jednak synchronizacja nie zawsze jest możliwa. Zarówno zmuszenie aplikacji do pobrania wartości zmiennej jak i zmuszenie aplikacji do wygenerowania wartości zmiennej z cyklem sieci może

być trudne. Typowym tego przykładem może być abonent pośredniczący, czyli omawiany *Firewall++*.



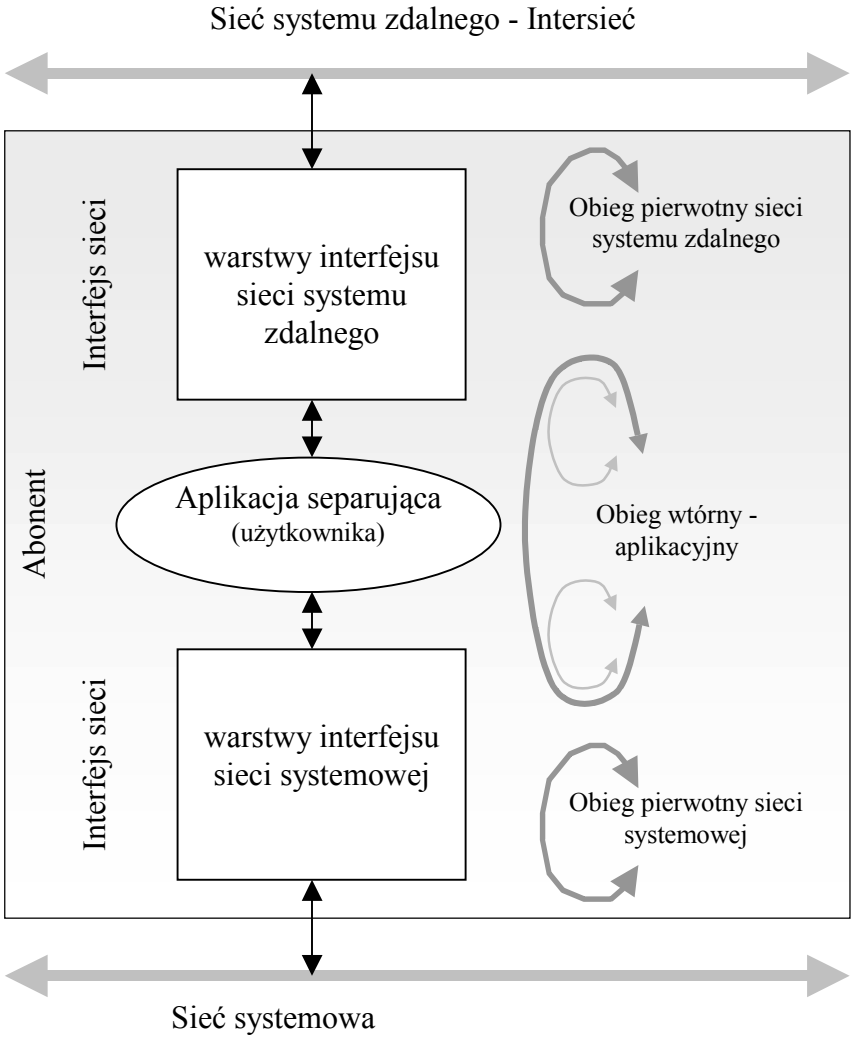
Rys. 41 Rozdzielne obiegi informacji w interfejsie sieciowym

Przy założeniu korzystania z abonenta pośredniczącego pomiędzy siecią systemową a siecią systemu zdalnego, nadrzędną warstwą aplikacyjną interfejsu będzie oprogramowanie kontrolujące dostęp do medium w sposób zdeterminowany czasowo oraz pozostałe warstwy *Firewalla++* (rys. 38).

Strukturę tak skonstruowanego abonenta przedstawiono na rysunku 42. Wtórny obieg aplikacyjny w rzeczywistości składa się z przynajmniej dwóch niezależnych strumieni informacji. Jeden związany z warstwą aplikacji interfejsu intersieci, drugi związany z warstwą aplikacji interfejsu sieci systemowej. Jednak dla uproszczenia, ze względu na niespójność czasową współdziałania tych obiegów z punktu widzenia obsługi informacji w sieci systemowej, można traktować je jako pojedynczy strumień. Z punktu widzenia pierwotnego obiegu sieci systemowej obieg pierwotny intersieci nie jest widziany lub jest widziany jako funkcjonalny fragment warstwy aplikacji, z którą on pracuje. Analogicznie dla pierwotnego obiegu intersieci. Dość dobrze można to wyjaśnić na przykładzie implementacji opisanej w załączniku VI.A.1.

Implementacja mechanizmu określania niespójności czasowej przekazywanej informacji w aplikacji separującej abonenta z rysunku 42, stworzy rozwiązanie nadające się do zastosowań w przemysłowych systemach informatycznych wykorzystujących intersieć wraz ze zdalnym dostępem do systemu lokalnego. Opisana możliwość implementacji takiego

mechanizmu dowodzi poprawności tezy trzeciej w punkcie c. Zagadnienie to będzie szerzej opisane w rozdziale 9.1.3, 9.1.4.

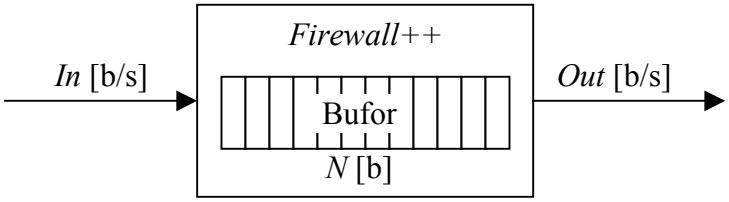


Rys. 42 Schemat obiegu informacji w abonencie separującym sieci

8.3. Określenie jakości usług przekazywania danych

Ponieważ podstawowe zadanie *Firewalla++* to przekazywanie danych z jednego do drugiego podsystemu komunikacyjnego, należy charakteryzując *Firewalla++* określić jakość takiej usługi. Cykle sieci systemowej oraz intersieci nie są zsynchronizowane, tak samo jak obiegi informacji w interfejsie komunikacyjnym *Firewalla++*. Może zatem zaistnieć sytuacja, iż strumień danych przychodzących do abonenta pośredniczącego nie będzie równy strumieniowi wyjściowemu. Wynika to z faktu, że cykl zdeterminowany sieci systemowej generuje strumień stały lub określony w przedziale oraz umożliwia wprowadzenie do obiegu liczbę danych określoną przez wartość maksymalną. Natomiast cykl intersieci jest nieokreślony, więc nieprzewidywalny. Jednocześnie nie wolno dopuścić do utraty danych. *Firewall++* nie gwarantuje spójności czasowej przekazywanych danych, lecz jednocześnie nie powinien podczas poprawnej pracy powodować utraty przekazywanych zmiennych a także zmieniać kolejności przekazywania zmiennych.

Aby zagwarantować jakość usług (ang. *quality of service*) przekazywania zmiennych takiego oprogramowania, należy zapewnić wspominane wcześniej buforowanie nadchodzących pakietów. W sytuacji, gdy strumień wejściowy będzie większy od wyjściowego *Firewall++* powinien być w stanie zapewnić przekazywanie danych przy określonym strumieniu przez określony czas.



Rys. 43 Buforowanie danych

Stosując mechanizm buforowania zmiennych i zakładając, że czas obsługi danych jest mniejszy od minimalnego okresu cyklu sieci docelowej, *Firewall++* jest w stanie zapewnić jakość usług przekazywania danych na poziomie:

$$Q = \frac{N}{In - Out}[s]. \tag{27}$$

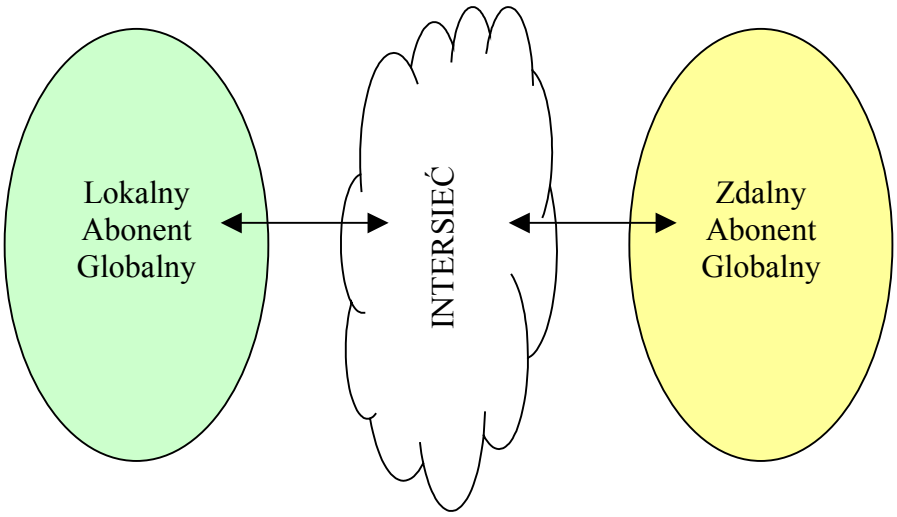
Oznacza to, że warstwy oprogramowania pośredniczącego umożliwiają przekazywanie danych bez ich utraty przez Q sekund przy strumieniu wejściowym In [b/s] i wyjściowym Out [b/s].

8.4. Podsumowanie cech *Firewalla++*

Systemy oparte o *Firewalla++* mogą służyć dla realizacji aplikacji przemysłowych wymagających ograniczeń czasowych jednak dostęp nie do wszystkich danych użytecznych obsługiwanych przez ten system ma charakter deterministyczny. *Firewall++* nie gwarantuje spójności czasowej aplikacyjnych zmiennych globalnych, lecz może zapewnić obsługę mechanizmów określania tej niespójności oraz gwarancję przekazania każdej kolejnej zmiennej przez określony czas. Dla wybranej grupy informacji możliwe jest tworzenie aplikacji czasu rzeczywistego. Dzięki idei *Firewalla++* można skonstruować jednoczesną obsługę niezdeterminowanych w czasie zmiennych globalnych w zdeterminowanym cyklu obsługi zmiennych lokalnych.

9. Uwarunkowania pracy systemów przemysłowych w intersieci

Zgodnie z przedstawionym wcześniej podziałem systemów przemysłowych można określić dwie dziedziny zastosowania protokołu TCP/IP w tych systemach. Protokół ten może być wykorzystywany w systemach komunikacyjnych systemów: lokalnych i zdalnych, czyli obsługiwać wewnętrzne mechanizmy komunikacyjne globalnych abonentów: lokalnego i zdalnego. W niniejszym rozdziale przedstawiona została analiza wykorzystania protokołu TCP/IP w sieciach umożliwiających przekazywanie danych pomiędzy abonentami globalnymi. Stosując strukturę systemu przemysłowego przedstawioną na rysunku 37 rozdziału 7.2.3 można wyodrębnić warstwę komunikacyjną służącą do przekazywania danych pomiędzy abonentami. Warstwę tę stanowi intersieć (rys. 44). Sieć Internet jest obecnie jedyną ogólnoświatową intersiecią opartą o TCP/IP. Dokonano zatem analizy zastosowania protokołu TCP/IP w systemach przemysłowych pod kątem wykorzystania intersieci, zakładając, że główny cel tej analizy stanowi rozwiązanie atrakcyjne ekonomicznie przy zachowaniu bezpieczeństwa zdalnego dostępu.



Rys. 44 Intersieciowa współpraca abonentów globalnych

Wymiana informacji w środowisku Internetu oparta jest o protokół IP. Poniżej dokonano analizy jak warstwa IP wpływa na pracę informatycznego systemu przemysłowego, wykorzystującego zdalny dostęp. Istnieje kilka zagadnień, które wpłyną na taką pracę, są to:

- zakres adresacji,
- opóźnienia transmisji,
- niezawodność transmisji,
- dostęp do danych użytecznych.

Dla powyższych zagadnień rozważania przeprowadzono dla dwóch przypadków. Pierwszy to wykorzystanie obecnie stosowanej wersji IP, a drugi to jego rozwojowa wersja IP6.

9.1. Protokół IP

W swobodnym ruchu w intersieci opartej o protokół transmisji datagramów IPv.4.0 (ang. *Internet Protocol version 4.0*) nie pracują żadne mechanizmy determinujące parametry czasowe przekazywania danych. Niezależnie od użytych protokołów warstw wyższych. Niezawodne dostarczanie pakietów w oparciu o TCP gwarantuje przesłanie danych do odbiorcy, lecz nie gwarantuje czasu doręczenia. Podobnie z transportem UDP, tylko bez gwarancji doręczenia.

9.1.1. Charakterystyka ogólna

Zakres adresacji w protokole IP dla systemów zamkniętych oraz separowanych nie stanowi żadnego problemu. Abonenci nie muszą posiadać unikalnego w skali intersieci adresu IP, gdyż odizolowana sieć nie jest widoczna w tej intersieci. Zakres ten, wynoszący ponad cztery miliardy unikalnych adresów, a wynikający z cztero bajtowego adresu IP, znacząco przekracza potrzeby tego typu systemów. Problem może pojawić się dla systemów otwartych pracujących w Internecie. W systemach tych każdy abonent musi mieć unikalny adres IP. Dzieje się tak, gdyż wszyscy abonenci dostępni są bezpośrednio w intersieci, a w skali całej intersieci nie mogą istnieć abonenci o tych samych adresach. Praktyczny przydział unikalnego adresu może nie być łatwy. Powszechnie wiadomo, iż w związku z dynamicznym rozwojem Internetu, przestrzeń adresowa IP się kurczy. Przyznawanie adresów dla urządzeń przemysłowych może nie być w interesie opiekuna klasy adresów IP. Nawet, jeżeli jest nim sam użytkownik systemu. Rozwiązania typu DHCP (ang. *Dynamic Host Configuration Protocol*) niczego nie rozwiązują, gdyż dynamiczny przydział i tak odbywa się z jakiejś puli i nie powoduje rozrostu przestrzeni adresowej. Poza tym specyfika urządzeń przemysłowych oraz celów ich stosowania wymusza ciągłość pracy interfejsu sieciowego. Mechanizm DHCP staje się zatem nieprzydatny.

Opóźnienia, jakie wnosi intersieć są nieszacowalne. Można określić jedynie minimalny czas dotarcia pakietu. Jednak obliczenie takiego parametru dla intersieci jest trudne i ma charakter chwilowy[6]. Określenie czasu maksymalnego jest niemożliwe i należy przyjąć, że czas maksymalny dąży do nieskończoności. Można również bazować na parametrze czasu życia datagramu oraz liczbie retransmisji. Na tej podstawie można ustalić granicę *time-outu*, po której pakiet uznaje się za zaginiony. Z teorii kolejek wynika, że wariancja czasu podróży pakietu zmienia się proporcjonalnie do $1/(1-L)$, gdzie L oznacza współczynnik aktualnego obciążenia sieci ($0 \le L \le 1$) [14, 25]. Wariancja czasu przesyłania danych przez intersieć będzie

[6] Nie można założyć stabilności struktury intersieci. Podlega ona ciągłym modyfikacjom.

rosła wraz ze wzrostem obciążenia tej sieci. Jest to problem, którego nie sposób wyeliminować.

Transmisja z użyciem IP stanowi bezpołączeniową zawodną metodę przenoszenia danych. Nawet w przypadku ustanowienia niezawodnego strumienia na poziomach wyższych (TCP), nie otrzymuje się żadnej gwarancji, że transmisja zostanie zakończona poprawnie. Dzieje się tak, dlatego, że pakiety wędrują przez sieci fizyczne nie będące w zakresie administracyjnym użytkownika systemu. Błędy i wyczerpanie zasobów tych sieci może prowadzić do utraty dostępu z żądanym abonentem. Problem ten jest generalnie problemem zaufania. Żaden protokół nie zagwarantuje poprawnego przejścia przez sieci obce. Przyjmując założenie, że wszystkie niezbędne podsieci wykorzystywanej przez nas sieci wirtualnej są i będą sprawne, można stwierdzić, iż przesył będzie działał w sposób niezawodny. Protokoły TCP/IP nie niszczą pakietów celowo. Zagubienie, utrata lub inne problemy związane np. z przeciążeniem występują zawsze z powodu błędów lub utraty zasobów [14, 15, 16]. Jednak dla większości instalacji przemysłowych problem zaufania powinien być priorytetowy. Nie wolno zakładać, że intersieć udostępni niezawodny przesył danych. Nawet specjalistyczne sieci pracujące w systemach zamkniętych z redundantnymi interfejsami i medium nie są w stanie tego zapewnić. Zawsze może wystąpić awaria a sieć wirtualna jest narażona na o wiele więcej czynników destrukcyjnych, niezależnych od użytkownika systemu, niż sieć lokalna czy polowa.

Swobodny sposób połączeń stosowany w Internecie, powodujący istnienie wielu tras alternatywnych, stwarza wysoki stopień niezawodności połączeń. W Internecie pakiety mogą podróżować różnymi trasami, co w przełożeniu na terminologię sieci systemowych można w przybliżeniu porównać do redundancji medium. W przypadku awarii trasy wykorzystywanej (optymalnej) wyznaczana jest automatycznie trasa alternatywna, jeśli ta zawiedzie, to wówczas kolejna itd. Stwarza to dość skuteczny mechanizm zabezpieczenia przed awarią, w skali globalnej skuteczniejszy niż w skali lokalnej mechanizmy redundancji. Warunkiem podstawowym musi być jednak rozbudowanie intersieci w całej domenie terytorialnej działania systemu. Musi istnieć n alternatywnych niezależnych tras, gdzie niezawodność przesyłu jest wprost proporcjonalna do n. Skuteczność awaryjnego trasowania pakietów jest tak dobra jak maksymalna liczba alternatywnych i niezależnych tras pomiędzy dowolnymi dwoma węzłami sieci mogącymi uczestniczyć w wymianie informacji.

9.1.2. Określanie jakości usług

Mechanizmy sterowania priorytetami i rezerwacji zasobów takie jak QoS (ang. *Quality of Service*) [108, 27], są przydatne dla usług przesyłania danych multimedialnych, poprawiając ciągłość odbioru strumienia danych, lecz bez wspomagania na poziomie samego IP nie są w stanie zagwarantować czasu dostarczenia informacji określonego w przedziale przez minimalną i maksymalną jego wartość. Mechanizm QoS działa na zasadzie sterowania priorytetami poszczególnych strumieni danych, powodując podwyższanie lub obniżanie ich priorytetów kolejek transmisji lub zarządzanie trasami pakietów [30]. Widać tu pewną

analogię do systemu przedstawionego w rozdziale 4.3 rysunek 9. W wyniku przyjęcia mechanizmu QoS jako zapewniającego dynamiczną prioretyzację zadań transmisji otrzyma się system klasy *Soft Real Time*, czyli nie spełniający wymogów determinizmu czasowego. Wspomaganie na poziomie protokołu IP przewidywane w IP wersji 6. Jednak obecnie znajduje się ono na etapie propozycji do standardu, więc nie można mówić o jakimkolwiek praktycznym wykorzystywaniu tych mechanizmów w praktycznych aplikacjach (rozdział 9.2) [13, 108, 142].

9.1.3. Określanie jakości danych użytecznych

Dla szerokiego spektrum aplikacji przemysłowych istnieje potrzeba pozyskiwania danych z gwarancją czasu doręczenia [61]. Istnieje jednak szereg informacji, dla których wymagania procesu sprowadzają się do określenia jakości danych użytecznych. Przez jakość danych rozumie się klasyfikację logiczną względem czasu dostarczenia danych i ich czasu ważności. Jest to klasyfikacja odzwierciedlająca tzw. świeżość informacji lub jej czas życia [36].

Istnieje możliwość określania jakości otrzymywanych danych użytecznych w systemach z separowanym obiegiem informacji. Określenie takie nie rozwiąże problemu zdeterminowanego dostępu do informacji, lecz da aplikacjom systemu możliwość dokładniejszej identyfikacji zdarzeń związanych a informacjami dostarczanymi bez gwarancji czasowych. Poszczególne warstwy aplikacyjne interfejsów sieciowych mogą posiadać specjalne procedury reakcji dla przypadków czasowych niespójności dostarczania danych z sieci zewnętrznej. Jako przykłady takich procedur można wymienić wymuszanie ustawień bezpiecznych, przyjmowanie parametryzacji domyślnych, generacje alarmów, logów ltp. Celowość wykorzystania warstwy generującej informacje statusowe zależy od wymagań kontrolowanego procesu i powinna być określana na etapie doboru warstw protokołów [35, 57].

9.1.4. Mechanizmy statusowe

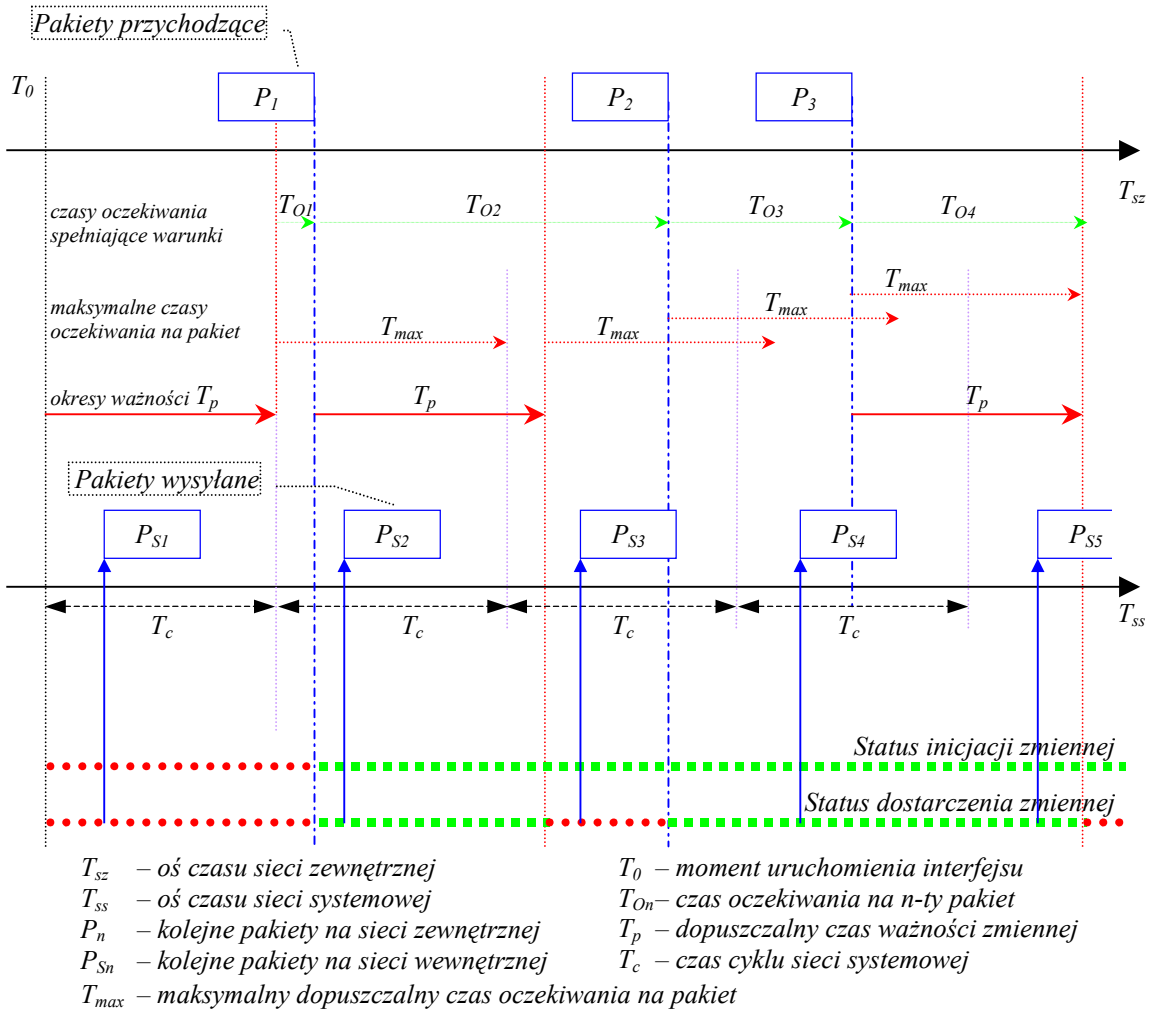
Mechanizmy statusowego określania spójności czasowej informacji przesyłanej w systemie komunikacyjnym są stosowane w zaawansowanych protokołach sieci przemysłowych. Gdy intersieć ma współpracować z siecią o takim właśnie protokole, wówczas współpraca warstw aplikacji względem obsługi informacji statusowej jest bardziej naturalna i prostsza w realizacji niż inne metody określania jakości informacji użytecznej.

Istnieją komercyjne rozwiązania dla stosu TCP/IP obsługujące stemple czasowe oraz określające jakość transmisji jak np. w protokole SuiteLink mechanizm VQT [95]. Mechanizm ten obsługuje stemple czasowe, które jednak bez precyzyjnej synchronizacji zegarów nadajnika i odbiornika są mało przydatne. Informacja o jakości przesłanych danych zawiera informacje o ogólnej poprawności danej, powodu niedostarczenia poprawnej danej oraz informacje o przekroczeniu limitów przez wartości pomiarowe. Informacje te nie określają jednak spójności czasowej dostarczanej informacji.

Mechanizm określania jakości danych użytecznych można zaimplementować bazując na obsłudze specjalnych informacji statusowych obsługiwanych przez warstwy aplikacji. Do informacji tych należą:

- status inicjacji,
- status dostarczenia,
- status pobrania.

Przy przyjęciu założenia, że intersieć pracuje poprawnie, czyli bez występowania przeciążeń, można oszacować przedział czasu, którym sieć dostarczy nowy pakiet z danym prawdopodobieństwem. Jeśli z drugiej strony określi się, biorąc pod uwagę wymagania procesu czas, po którym dana zmienna powinna zostać odczytana od nadawcy, otrzymamy wymagany czas ważności zmiennej. Porównując te dwa parametry, będzie się w stanie określić czy stabilna intersieć jest w stanie dostarczyć informację w czasie mniejszym lub równym od czasu jej ważności. Jeżeli tak, to wówczas można dla danej zmiennej skonstruować mechanizm generacji i testowania statusów jakości informacji przenoszonej przez tą zmienną.



Rys. 45 Przykład określania jakości danych użytecznych

Na rysunku 45 przedstawiono mechanizm określania jakości danych użytecznych na przykładzie dostarczania pakietów z sieci zewnętrznej do sieci systemowej. Pakiety *P* są

dostarczane z intersieci z cyklem nieokreślonym, czyli aperiodycznie. Na osi czasu T_{sz} przedstawiono zdarzenia na sieci zewnętrznej, natomiast na osi czasu T_{ss} przedstawiono zdarzenia na sieci systemowej. Czas T_0 określa moment rozpoczęcia pracy przez interfejs sieci wewnętrznej. Mechanizm określania jakości rozpoczyna odmierzanie czasu. Do warstw niższych interfejsu sieci systemowej zapisuje domyślną (lub pozostawia nieokreśloną) wartość zmiennej wraz z informacją o niespełnieniu statusu inicjacji przez tą zmienną. Zmienna taka będzie rozesłana z cyklem pracy sieci do odpowiednich abonentów. Warstwy aplikacji tych abonentów otrzymają niezainicjowaną wartość zmiennej wraz z informacją o braku inicjacji wartością od abonenta pracującego w intersieci. Na podstawie tej informacji mogą przedsięwziąć odpowiednie kroki przygotowane na taką okoliczność.

W sytuacji, gdy informacja z pakietu zostanie dostarczona, status inicjacji zostanie ustawiony na spełniony i abonenci odczytujący zmienną otrzymają informację o zainicjowaniu danych przenoszonych przez tą zmienną. Na rysunku 45 przedstawiono czas niespełnienia statusu inicjacji w kolorze czerwonym, natomiast w kolorze zielonym czas, w którym jest on spełniony.

Czas T_p określa okres ważności zmiennej dostarczanej pakietem P. Jeżeli informacja użyteczna z pakietu P nie będzie dostępna przed upłynięciem czasu T_p, czyli jeżeli czas oczekiwania

$$T_O > T_p\,, \tag{28}$$

wówczas mechanizm określania jakości powinien pozostawić bez zmian ostatnio zapisaną wartość zmiennej wraz z informacją o niespełnieniu statusu dostarczenia przez tą zmienną. Po dostarczeniu informacji użytecznych z pakietu P dla niższych warstw sieciowych, czyli dla obiegu pierwotnego sieci systemowej, następuje ustawienie stanu statusu dostarczania na spełniony. Oznacza to, że od tego momentu wartość zmiennej będzie ważna przez okres T_p. Na rysunku 45 przedstawiono okresy ważności statusu doręczenia w kolorze zielonym, natomiast w kolorze czerwonym okresy niespełnienia statusu dostarczenia. Dla tego przykładu pakiet P_{S3} zostanie przesłany do abonentów sieci systemowej z wartością dostarczoną przez pakiet P_1 wraz z informacją, że jest to wartość dostarczona od producenta w czasie późniejszym niż przyjęta wartość czasu ważności T_P. Zanik statusu dostarczenia dla tego przypadku wynika z opóźnienia pakietu P_2 wynoszącego:

$$\Delta T_{P2} = T_{o2} - T_p. \tag{29}$$

Dla ogólnego przypadku, aby status dostarczenia był spełniony nie może zachodzić nierówność (28) lub dla n-tego pakietu musi być spełniona nierówność:

$$(T_{Pn-1}+T_{on}) \le (T_{Pn-1}+T_P) \qquad \text{dla } n>0, \tag{30}$$

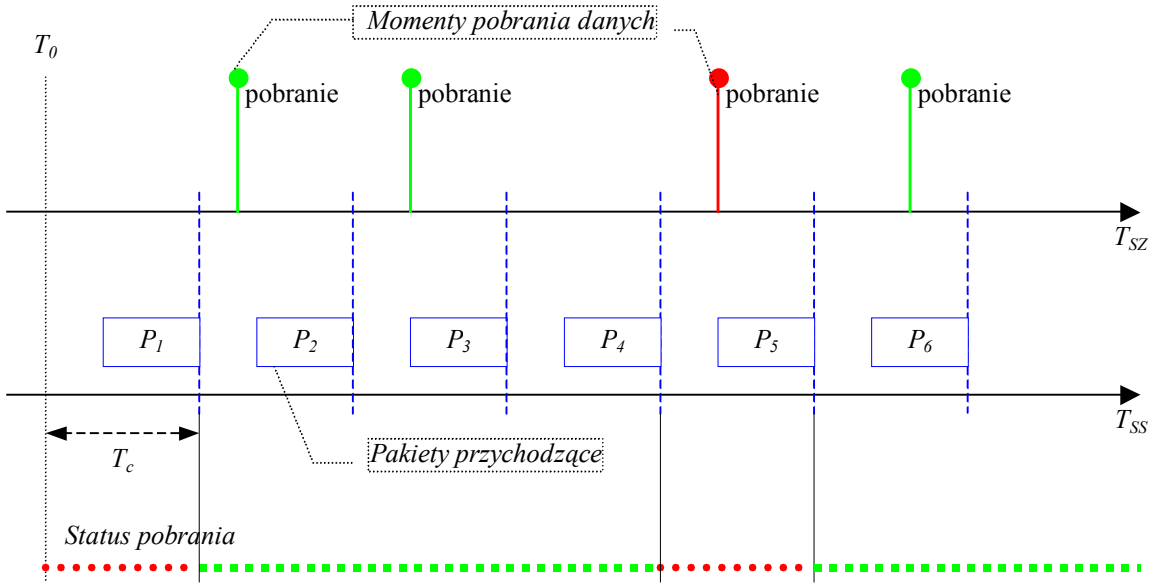
gdzie $T_{P0} = T_P$.

Proponuje się przesyłanie powyższych informacji statusowych wraz z wartością zmiennej, czyli wraz z samą informacją. Oznacza to określanie jakości danych przez abonenta zapisującego, czyli generującego informację w sieci, a nie przez stację odbiorcze. Mechanizm taki zwiększa wydajność określania jakości oraz zapewnia rozsyłanie informacji ostatnio

wygenerowanej zgodnie z regułami pracy sieci izolowanej. Mechanizmy te są podobne do stosowanych mechanizmów określania świeżości danych w sieciach z protokołem FIP/WorldFip [114].

Dla trzeciej z proponowanych informacji statusowych należy rozważyć odwrotny kierunek przesyłania danych. A mianowicie dostarczanie informacji z sieci systemowej dla intersieci. Jeżeli obieg wtórny interfejsu pobiera informację z buforów komunikacyjnych warstw niższych interfejsu sieci systemowej z cyklem mniejszym lub równym cyklowi tej sieci, to wówczas status pobrania będzie spełniony, a każda informacja pochodząca od abonenta z tej sieci zostanie przekazana do intersieci.

W przypadku przeciwnym, jeśli okres cyklu wtórnego jest większy od okresu cyklu pierwotnego, wówczas może nastąpić zjawisko gubienia danych. Zostało to zobrazowane na rysunku 46.



Rys. 46 Przykład określania statusu pobrania

T_c określa nam czas cyklu sieci dla pakietu P. Na osi czasu T_{SZ} przedstawiono zdarzenia zachodzące w obiegu wtórnym. Jak widać z rysunku, informacja z pakietu P_4 nie została przekazana do retransmisji w intersieci. Wówczas, przy następnym pobraniu informacji, status pobrania przenoszony razem z tą informacją powinien zostać ustawiony na niespełniony. Oznaczać to będzie, iż otrzymana informacja jest poprawna, lecz jakaś informacja pochodząca z sieci systemowej, która była przez nią transmitowana przed tą otrzymaną, została utracona.

Tak jak poprzednio status jest wypracowywany przez abonenta separującego sieci i transmitowany wraz z informacją użyteczną. Jednak nie informuje o zależnościach czasowych związanych z dostarczeniem danych, lecz o niebezpieczeństwie możliwości pominięcia niektórych informacji. Przypadki takie nie wynikają z braku synchronizacji obiegu pierwotnego sieci zewnętrznej z obiegiem wtórnym, lecz z możliwości przepełnienia buforów komunikacyjnych. Sieć zewnętrzna jako sieć niedeterministyczna powoduje, że

wytransmitowanie informacji zgromadzonej w buforze może być opóźniane o czas nieokreślony. Przy krótkim okresie aktualizacji danych w tych buforach, wynikającym z obiegu wtórnego, strumień zasilający bufor może być większy niż strumień opróżniający bufor. Doprowadzi to do jego przepełnienia i utraty możliwości retransmisji każdej wartości danej zmiennej z sieci wewnętrznej. Jakość pracy oprogramowania pośredniczącego zależy zatem od możliwości buforowania danych co było opisane w rozdziale 8.3.

Analogiczne do wcześniejszych rozważań, dla przypadku zapisu do sieci zewnętrznej może być z powodzeniem wykorzystywany status inicjacji. Teoretycznie również wtedy może wystąpić pobranie w pierwszym cyklu pracy sieci systemowej, gdy informacja nie jest jeszcze zainicjowana. Jego działanie będzie identyczne jak dla przypadku zapisu do sieci systemowej.

Mechanizmy określania jakości danych użytecznych w sieciach z odizolowanym obiegiem informacji nie gwarantują żadnych parametrów czasowych dostarczania danych. Stanowią tylko mechanizm informowania warstw aplikacyjnych abonentów systemu o stanie otrzymywanych danych. Na podstawie takiego stanu warstwy te mogą w szybki i pewny sposób określać przydatność otrzymywanych informacji i wypracowywać odpowiednią reakcję.

Status inicjacji eliminuję możliwość pracy na wartościach niezainicjowanych. W systemach przemysłowych praca na takich wartościach jest niedopuszczalna. Status doręczenia jest miernikiem czasowej spójności integrowanej niedeterministycznej intersieci i deterministycznej sieci wewnętrznej. Na jego bazie można budować mechanizmy typu *watch-dog*, od których można uzależniać pracę aplikacji. Status pobrania może mieć szczególne znaczenie dla zdalnych stacji realizujących funkcje raportowania i alarmowania. Funkcje takie opierają się na gromadzeniu lub śledzeniu bieżących wartości wybranych zmiennych. Takie jest właśnie główne zadanie tego typu abonenta, niezależnie czy dołączonego lokalnie czy zdalnie. Przy procesach alarmowania przeważnie niezbędny jest czas krytyczny reakcji systemu. Przy podłączeniu intersieciowym nie jest to realizowalne. Dlatego z praktycznego punktu widzenia istotniejszą funkcją jest raportowanie. Dla raportowania czas krytyczny dostarczenia informacji nie jest tak istotny jak sam fakt jej doręczenia oraz stemple czasowe zdarzeń. Jeżeli proces wymaga monitorowania danej wartości zmiennej co określony okres czasu, to wartość takiej zmiennej powinna być dostarczana do stacji raportującej nie rzadziej niż ten okres. Nawet, gdy pakiet ze zmienną zostanie przez intersieć doręczony lecz zmienna będzie zawierała starą wartość, co oznacza że jakaś wartość została utracona, to analiza pracy systemu na podstawie tak wygenerowanego raportu będzie nieścisła. Przykład aplikacji wykorzystującej opisywane statusy na stykach wielu warstw aplikacyjnych przedstawiony jest w rozdziale VI.A.

Bezpośrednie wykorzystanie obecnie stosowanego protokołu IP nie zapewni zdeterminowanej w czasie pracy interfejsów sieciowych systemów przemysłowych. Jednakowoż jego elastyczność i otwartość pozwala na implementację mechanizmów, które dla pewnej grupy funkcji umożliwiają współpracę sieci deterministycznych z intersiecią, lub

dla pewnej grupy zastosowań, jak systemy nie wymagające zdeterminowania wymian, taką pracę umożliwiają. Implementacja przedstawionych mechanizmów w module *Firewall++* dowodzi słuszności tezy trzeciej w punkcie c.

9.2. Rozwój protokołu IP – IPng

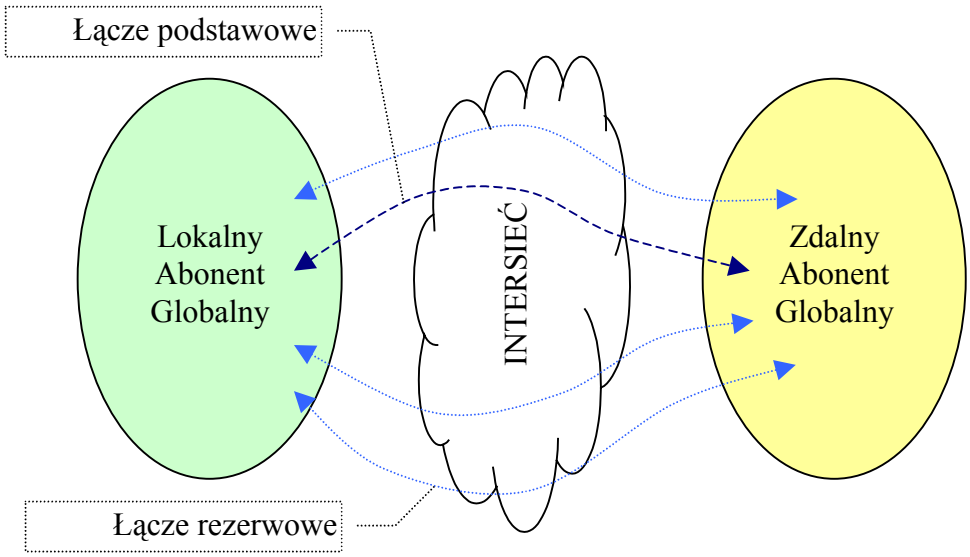
W niniejszym podrozdziale przedstawiono analizę mechanizmów protokołu IP w wersji IP6 pod kątem uwzględnienia opisanych wcześniej problemów. Obecnie trwają pracę nad ustaleniem standardu dla tego protokołu [151]. Całość tych prac nazywa się często IPng (ang. *IP next generation*). Obecnie istnieje niewiele produktów mających funkcjonalność IPv6 rozwiniętą w stopniu umożliwiających prowadzenie badań. Jednym z pierwszych produktów posiadających stos TCP/IP jest Windows XP wraz z pakietem SP1 oraz QNX 6.2 natomiast w pełni sprawny stos posiada dla przykładu system Windows Server 2003 [145, 144, 150, 147, 149, 136]. Pełną listę implementacji można znaleźć w [142].

W kwestii dostępu do medium nic się nie zmieni. Zasada działania i idea pozostaje bez zmian, więc dostęp do medium pozostaje poza zakresem definiowanym przez IP.

Trzydziestodwubitowy zakres adresacji, który powoduje problemy z przestrzenią adresową Internetu i nie jest bez znaczenia dla otwartych systemów przemysłowych, ulegnie w nowej wersji IP drastycznemu powiększeniu. Adres IPv6 będzie miał 128 bitów, zatem wyczerpania przestrzeni adresowej nie da się oszacować w przewidywalnej przyszłości. Można zatem stwierdzić, iż problem adresacji dla systemów otwartych zostanie rozwiązany.

Nowy protokół IP przewiduje możliwość stosowania wsparcia dla rezerwowania zasobów sieci. Mechanizm ten jest tworzony z myślą o transmisji multimediów, dla której niezbędna jest gwarantowana przepustowość oraz stałe opóźnienia w kanale transmisyjnym. Mechanizm taki, znajdzie zastosowanie dla tworzenia zdeterminowanych kanałów transmisji w intersieci.

Problemy związane z niezawodnością transmisji w sieciach obcych nie mogą być rozwiązane przez sam protokół. Rozwiązaniem tego problemu może być wprowadzenie redundancji deterministycznych kanałów wirtualnych (rys. 47).



Rys. 47 Redundancja kanałów wirtualnych

Redundancja powinna być zrealizowana na bazie kanałów niezależnych, czyli takich, które nie posiadają części wspólnych w postaci routerów, łączy i innych urządzeń infrastruktury intersieciowej. Łącza wirtualne można traktować podobnie jak zasoby sprzętowe zakładając ich awaryjność na pewnym ustalonym poziomie. Wprowadzając redundancję można doprowadzić do ustalenia ich awaryjności na poziomie dopuszczalnym dla użytkownika, czyli na przykład na takim, jakie ma łącze sieci systemowej. Rozwiązanie takie stanowi jedyną drogę do budowy rozległych systemów czasu rzeczywistego.

Nowa wersja IP przewiduje wprowadzenie mechanizmu fragmentacji pakietu na końcach, czyli zakłada się stałe MTU ustalonej ścieżki, jako MTU minimalne dla wszystkich składników tej ścieżki. Pakiet zostaje fragmentowany u nadawcy i defragmentowany u odbiorcy. Powoduje to zwiększenie wydajności przy poprawnym przesyle, lecz kłopoty przy przekierowywaniu pakietów w wypadku awarii składowego rutera lub sieci. Jeżeli ścieżka alternatywna ma większe lub równe MTU wówczas nie ma problemu, jednak, gdy jest ono mniejsze, wówczas pakiety muszą zostać tunelowane wewnątrz nowych specjalnie tworzonych datagramów dla trasy z mniejszym MTU. Cecha ta nie wpływa jednak negatywnie na wykorzystywanie redundantnych kanałów przedstawionych na rysunku 47, gdyż MTU dla kanału jest ustalane na wstępie, a w przypadku awarii zmieniamy jest cały kanał a nie jego fragment.

W kwestii dostępu do danych protokół IP6 nie będzie się różnił od IP w wersji obecnie używanej. Jednakowoż największe problemy związane z czasem dostępu a wynikające z opisanych powyżej opóźnień oraz zawodności przesyłu, rozwiązywać może mechanizm rezerwowania zasobów. Wykorzystanie nadrzędnych warstw aplikacyjnych umożliwiających kontrolę wymian będzie możliwe dla segmentów sieci systemowych oraz dla całego zarezerwowanego kanału lub grupy kanałów.

Przewidywane jest również wprowadzenie cechy rozszerzalności protokołu. Będzie to mechanizm, który spowoduje realizację dynamicznego dodawania funkcji do protokołu,

umożliwiając tym samym rozbudowę protokołu o nowe mechanizmy pozwalające dostosowywać protokół do nowych urządzeń i oprogramowania użytkowego.

Generalnie protokół IP w wersji 6 daje duże perspektywy dla rozwoju mechanizmów komunikacyjnych informatycznych systemów przemysłowych. Włączając w to zagadnienia determinizmu czasowego.

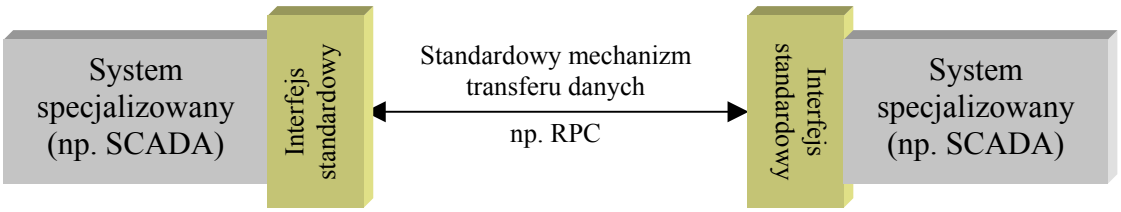
10. Wykorzystanie usług intersieci

Główną ideą tworzenia i korzystania z intersieci jest łączenie heterogenicznych systemów komunikacyjnych (strona 42). Aby tego dokonać musi być określony zbiór standardowych usług niezależnych od rozwiązań sprzętowo-programowych poszczególnych elementów składowych intersieci.

10.1. Podejścia tradycyjne

Przemysłowy system informatyczny wykorzystujący intersieć do transmisji danych użytecznych może wykorzystywać tą sieć przynajmniej w dwojaki sposób.

Po pierwsze może wykorzystywać protokół oraz medium do transferu danych użytecznych z jednego specjalizowanego systemu do drugiego specjalizowanego systemu (rys. 48). Podejście to jest znane i obecnie często stosowane. Przykład może stanowić dowolny system nadzorczy umożliwiający podłączenie drugiego systemu nadzorczego przez Internet z wykorzystaniem mechanizmu RPC (ang. *Remote Procedure Call*) [99, 101, 23]. Otrzymuje się zmultiplikowane stacje typu SCADA, które wykorzystując standardową usługę RPC definiowaną niezależnie od protokołu intersieci, mogą komunikować się ze sobą w środowisku heterogenicznym [67].

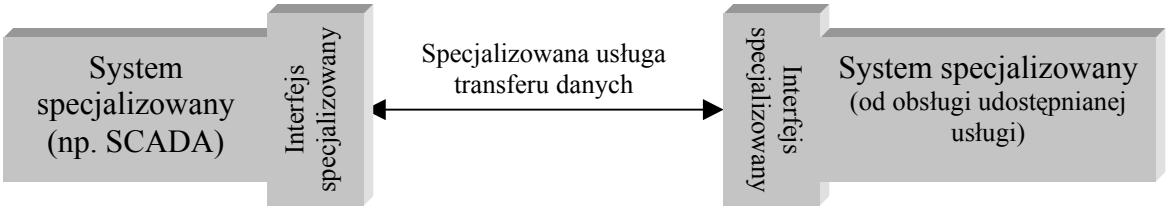


Rys. 48 Klasyczny przypadek wykorzystania standardowych usług intersieci

Można znaleźć szereg analogicznych przykładów wzajemnej komunikacji pomiędzy systemami. Może one być oparta o dowolnego rodzaju transfer danych z zakresu udostępnianego przez zestaw TCP/IP jak również mogą korzystać z dowolnej aplikacyjnej nadbudowy w postaci tunelowania innych protokołów. Rozwiązania takie mają jednak podstawową wadę. Użytkownik musi po obu stronach połączenia stosować specjalistyczne oprogramowanie. Jest to niewygodne z punktu widzenia mobilności użytkownika, a także dość kosztowne. Specjalistyczne pakiety oprogramowania jak np. SCADA, są w znacznej części niekompatybilne wzajemnie z poziomu warstw aplikacyjnych. Aby połączyć warstwy aplikacji różnych systemów pracujących w różnych środowiskach należy korzystać ze standardów niższych poziomów, przez co zachodzi utrata wydajności i strukturalnej spójności

całości systemu. Ponadto pakiety specjalizowane są kosztowne, wymagają dedykowanej konfiguracji i parametryzacji oraz nie są powszechnie i swobodnie dostępne.

Drugie podejście do wykorzystania usług intersieci, to tworzenie specjalizowanych usług na bazie standardowych protokołów TCP/IP. Problem polega na opracowaniu lub dopasowaniu takiej usługi, która będzie spełniała wymagania specjalizowanego użytkownika, jakim jest użytkownik przemysłowych systemów kontrolno-nadzorczych (rys. 49).

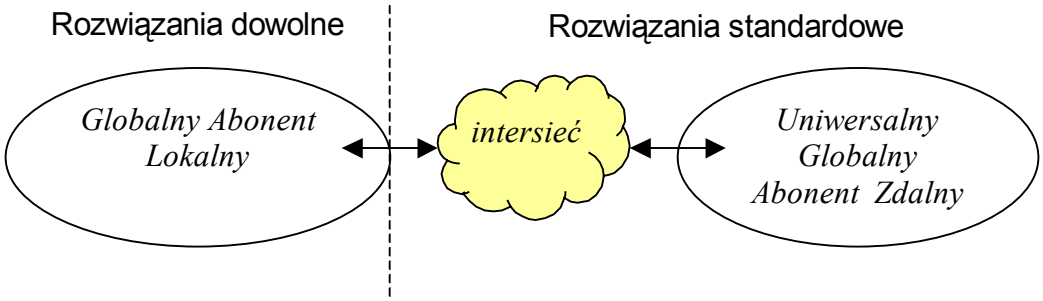


Rys. 49 Przypadek wykorzystania specjalizowanych usług intersieci

W przypadku protokołu TCP/IP i sieci Internet trudno rozważać opracowywanie standardowej usługi. Próby wprowadzenia usługi nowego typu są możliwe, gdyż standard TCP/IP ciągle się rozwija, jednak ze względu na wąskie i specjalistyczne grono użytkowników takiej usługi, standaryzacja byłaby trudna a osiągnięte zyski niewystarczające. Poza tym nadal w systemie muszą funkcjonować specjalizowane pakiety obsługujące transmisję i prezentację informacji.

10.2. Koncepcja uniwersalnego abonenta globalnego

Koncepcja uniwersalnego abonenta globalnego to skorzystanie z abstrakcyjnego, intersieciowego medium, zestawu standardowych protokołów intersieci i usług definiowanych w ramach opisu standardu tych protokołów (rys. 50). Jeżeli w danej intersieci istnieje zdefiniowana usługa, to wraz z nią, a właściwie dla niej, zdefiniowane są grupy wykorzystywanych protokołów oraz standardowy interfejs użytkownika. Rozpatrywane są tutaj usługi, dostępne bezpośrednio z poziomu użytkownika, czyli takie, z którymi użytkownik ma kontakt przez standardowy interfejs tej usługi.

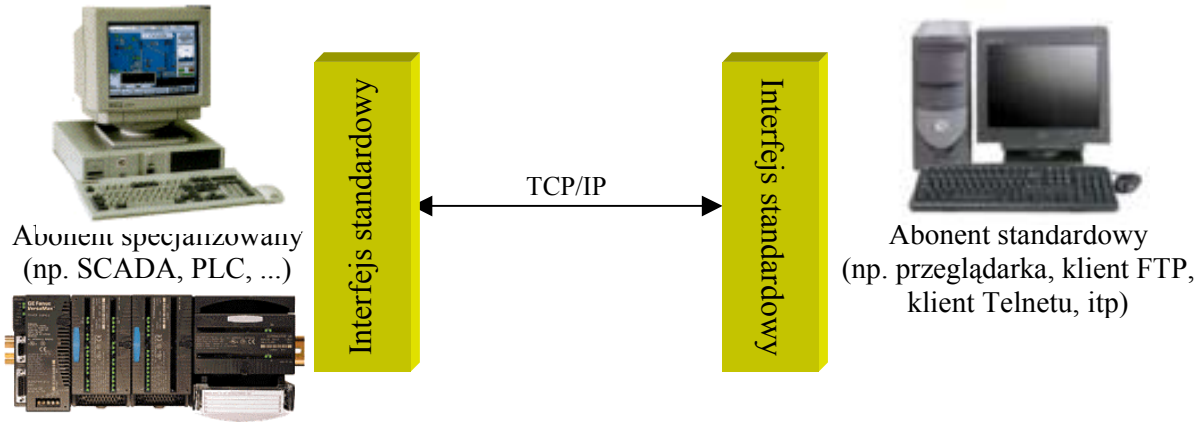


Rys. 50 Uniwersalny abonent globalny

W sieci Internet istnieje cały szereg usług umożliwiających użytkownikom korzystanie z intersieci w sposób czysto użytkowy. Jednym z istotniejszych celów stojących przed mechanizmami standaryzacji intersieci jest taki podział zadań funkcjonalnych sieci, aby stworzone grupy najbardziej pasowały do potrzeb jej użytkowników. Jeżeli zostanie zdefiniowana usługa obsługująca daną grupę funkcjonalną w sposób standardowy, wówczas

użytkownik jest uniezależniony od rodzaju sprzętu i systemu operacyjnego. Główne wytyczne do obsługi interfejsu użytkownika stają się standardowe, zatem użytkownik nie musi posiadać zaawansowanej wiedzy z zakresu intersieci i lokalnego systemu stacji roboczej. Przekładając to na wcześniejszy przykład z rozproszonymi stacjami SCADA, można stwierdzić, że bardziej korzystnym rozwiązaniem jest udostępnienie informacji przez powszechnie znaną standardową usługę sieciową i powszechnie znany standardowy interfejs użytkownika, który jest do tego powszechnie dostępny (rys. 51), niż tworzenie specjalizowanych dedykowanych i kosztownych stacji roboczych (rys. 48, 49).

Na bazie protokołów aplikacyjnych TCP/IP, czyli takich, które pracują na najwyższym poziomie modelu warstwowego interfejsu komunikacyjnego, oraz numerów portów identyfikujących usługę na zasadzie ang. *well-known port number*, można wykorzystywać standardowe usługi intersieci przeznaczone dla użytkowników tejże sieci, tworząc tym samym uniwersalnego abonenta zdalnego.



Rys. 51 Wykorzystanie standardowych usług intersieci i standardowych aplikacji klienta

Zakres obecnie wykorzystywanych usług Internetu nie odpowiada bezpośrednio wymaganiom stawianym przez informatyczne systemy przemysłowe. Tworzenie nowych usług przy obecnych możliwościach protokołów, na których one bazują, nie wprowadzi zmian jakościowych w obsłudze danych, gdyż nie da się wyeliminować ograniczeń wynikających z niedoskonałości istniejącej wersji protokołu. Jedyne, co można osiągnąć, to rozbudowa wykorzystywanych przez usługę protokołów, pod kątem dostosowania usługi do potrzeb wynikających z wymagań informatycznych systemów przemysłowych. Aspekt ten będzie rozważany w dalszej części pracy.

Należy wyraźnie rozgraniczyć wymagania systemu względem branej pod uwagę płaszczyzny wymian. Dla wymian poziomych nie istnieje potrzeba rozważania usług przeznaczonych dla użytkownika w postaci człowieka. Urządzenia uczestniczące w tego typu wymianach rzadko posiadają interfejs użytkownika. Natomiast dla wymian pionowych należy rozważyć takie usługi, które umożliwiają bezpośrednią interakcję z człowiekiem.

W dalszych rozważaniach skoncentrowano się na rozwiązaniach wykorzystujących standardowe usługi, interfejsy komunikacyjne i protokoły. Należą do nich usługi WWW (ang. *World Wide Web*), poczta elektroniczna (ang. *e-mail*), FTP (ang. *File Transfer Protocol*) oraz

Telnet. Wykorzystanie tych usług będzie omówione w aspekcie podstawowych zadań funkcjonalnych systemów kontrolno-nadzorczych.

Do najważniejszych dziedzin stosowania informatycznych systemów przemysłowych należą:

- wizualizacja bieżącego przebiegu procesu przemysłowego,
- monitorowanie i rejestracja wybranych parametrów oraz alarmowanie o przekroczeniu limitów wybranych parametrów procesu przemysłowego,
- generacja raportów, zestawień i podsumowań o stanie procesu przemysłowego i eksport danych do procesów nadrzędnych.
- sterowanie procesem przemysłowym z poziomu systemu oraz z poziomu użytkownika,

Zagadnienia sterowania procesem nie będą rozważane w kontekście zdalnego dostępu z poziomu intersieci jako z założenia niebezpieczne i nie zalecane do aplikowania.

10.2.1. Wizualizacja

W rozwiązaniach tradycyjnych zadanie wizualizacji stanu procesu spoczywa na stacjach typu SCADA (ang. *Supervisory Control And Data Acquisition*) [80, 89, 94]. Stacje wizualizacyjne stanowią interfejs pomiędzy systemem obsługującym proces a człowiekiem. Interfejs taki to najczęściej ekran graficzny współdziałający z klawiaturą i urządzeniem wskazującym. Najistotniejsze zastosowanie w tej dziedzinie będzie miała usługa WWW.

WWW jest usługą udostępniającą użytkownikowi dokument interaktywny. Takie samo zadanie wykonuje stacja wizualizacyjna. Zarówno jedne jak i drugie przekazują użytkownikowi informacje graficzno-tekstowe umożliwiając jednocześnie interakcję przy użyciu standardowych urządzeń wejścia jak klawiatura czy mysz. Różnice, jakie tutaj zaistnieją grupują się wokół następujących zagadnień:

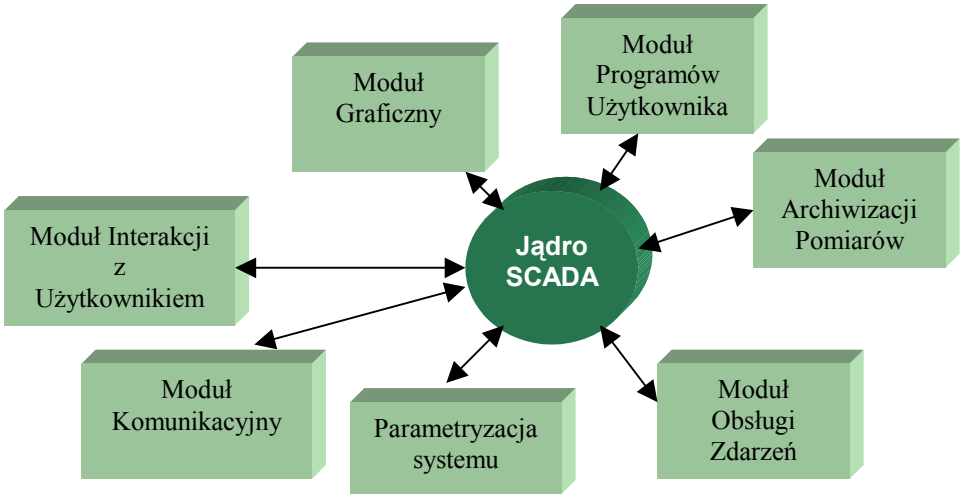
- sposoby prezentacji danych,
- sposoby dostarczania danych,
- mechanizmy kontroli dostępu do danych.

Usługa WWW działa w oparciu o model Klient-Serwer. Model ten doskonale pasuje do realizacji zdalnego dostępu do lokalnego systemu kontrolnego. Serwer, serwery lub hybrydy serwerów WWW z lokalnymi stacjami typu SCADA, mogą być aplikowane w systemie lokalnym. Aplikacje klienckie – przeglądarki (ang. *browsers*), mogą pracować w systemie zdalnym.

Abstrahując od różnic w udostępnianiu, przy korzystaniu ze standardowego interfejsu usługi WWW istnieje możliwość eliminacji specjalistycznego oprogramowania we wszystkich warstwach oprogramowania abonenta globalnego. Eliminacja taka spowoduje, że stacje zdalne staną się uniwersalne i nie będą wymagały żadnego oprogramowania specjalistycznego poza danymi pobieranymi przez sieć. Stanowi to rozwiązanie atrakcyjne ze względów ekonomicznych oraz wygody użytkowania.

10.2.1.1. Porównanie budowy systemów SCADA i przeglądarek WWW

Na rysunku 52 przedstawiono schemat nowoczesnej konstrukcji systemu typu SCADA [23]. Konstrukcja ta jest konstrukcją modułową. Każdy z modułów posiada własny zestaw zadań i funkcji do realizacji oraz komunikuje się z jądrem systemu. Jądro zapewnia funkcje nadzorcze i umożliwia przepływ danych.

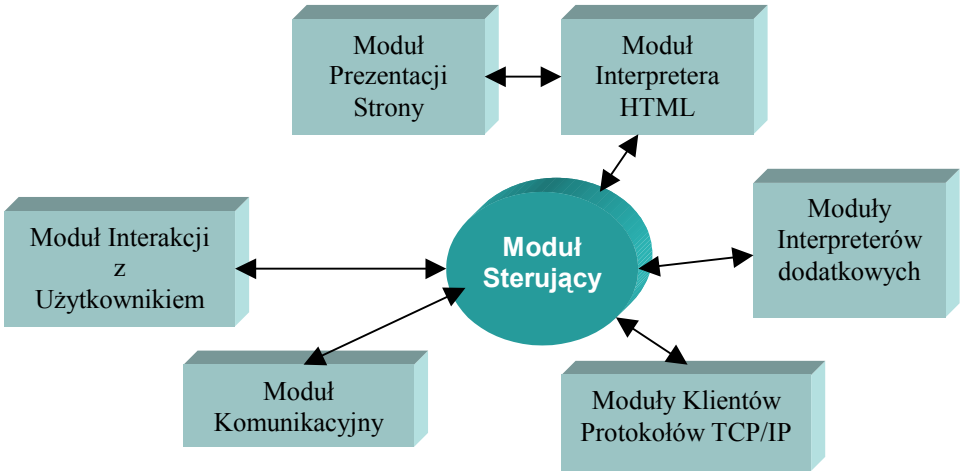


Rys. 52 Schemat modułowego systemu typu SCADA

W takim systemie można wyodrębnić kilka zadań funkcjonalnych:

- nadzorowanie pracy modułów,
- parametryzacja pracy modułów,
- obsługa komunikacji z systemem,
- obsługa interfejsu użytkownika,
- obsługa zdarzeń,
- obsługa specyficznego przetwarzania.

Podobne grupy funkcji realizuje przeglądarka WWW. Na rysunku 53 przedstawiono schemat budowy przeglądarki WWW. Podobnie jak dla systemu SCADA moduł sterujący integruje pracę pozostałych komponentów. Obsługa stosu TCP/IP zapewnia komunikację z resztą systemu oraz jednocześnie stanowi kanał dostępu do parametryzacji pracy.



Rys. 53 Schemat modułowej konstrukcji przeglądarki WWW

W standardowej stacji SCADA parametryzację pracy aplikacji wizualizacyjno-nadzorczej stanowi całość plików konfigurujących. W przypadku przeglądarki odpowiada jej zestaw plików HTML [43, 8] opisujących wygląd i działanie stron WWW. Każda przeglądarka posiada moduł interpretera HTML. Wynikiem działania tego modułu jest sterowanie działaniem modułu graficznego, czyli w efekcie to, co użytkownik widzi na ekranie.

Oprócz interpretera standardowego HTML przeglądarka może posiadać opcjonalne interpretery innych języków jak np. JavaScript, DHTML, Flash czy też języka Java. Wszystkie znaczące aplikacje przeglądarek istniejących na rynku posiadają standardowo wbudowany szereg opcji interpreterów lub umożliwiają dynamiczne rozszerzenie swoich możliwości przez dołączenie odpowiednich bibliotek (tzw. ang. *plug-in*).

Można wyprowadzić bezpośrednie zależności pomiędzy modułami.

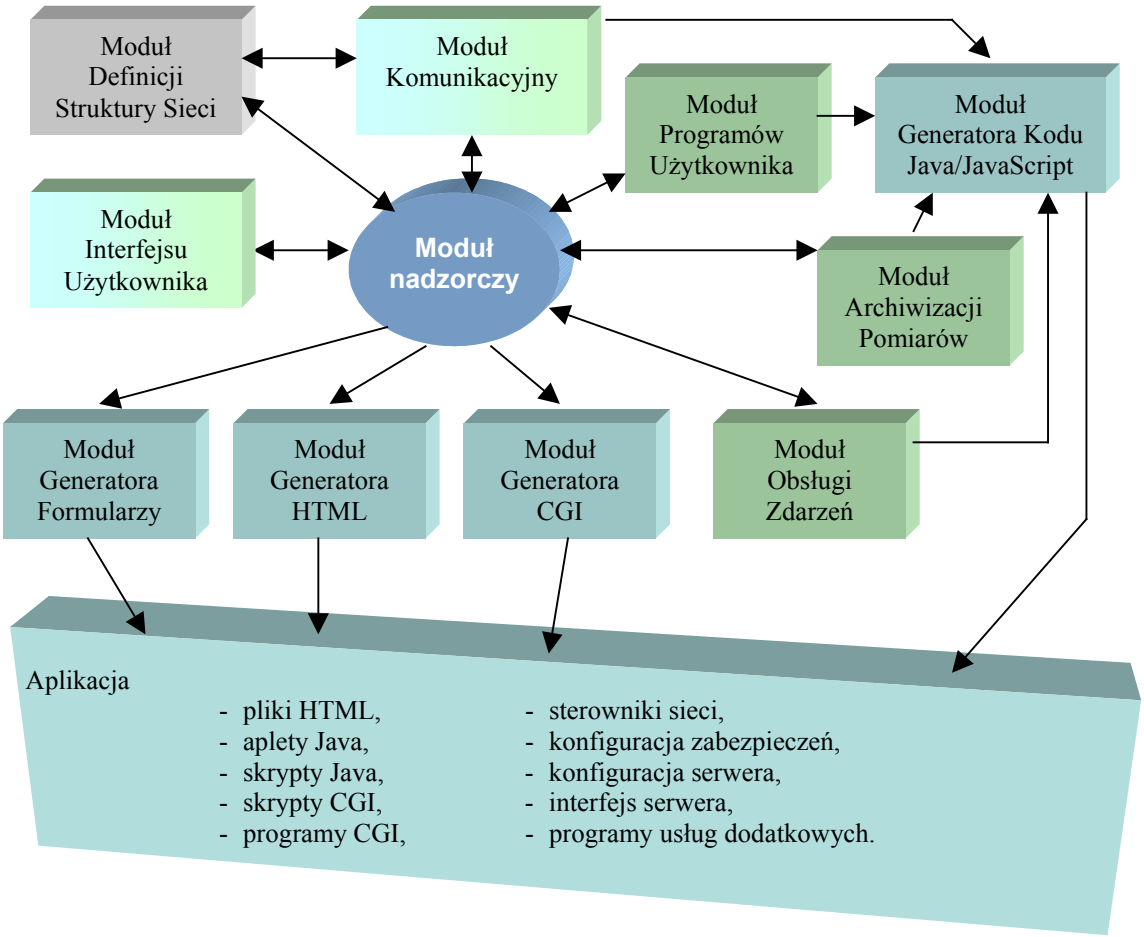
System SCADA	***Przeglądarka WWW***
Nadzorowanie pracy modułów	
Jądro systemu	Moduł sterujący
Parametryzacja pracy modułów	
Pliki konfiguracyjne	Pliki HTML
Obsługa komunikacji z systemem	
Moduły komunikacyjne sieci komputerowych	Moduły komunikacyjne TCP/IP
Obsługa interfejsu użytkownika	
Moduł interakcji z użytkownikiem (obsługa myszy, klawiatury itp. urządzeń) Moduł graficzny	Moduł interakcji z użytkownikiem (obsługa myszy, klawiatury itp. urządzeń) Interpreter HTML Moduł obsługi wyświetlania dokumentów (prezentacji dokumentów)
Obsługa zdarzeń	
Moduł raportowania Moduł obsługi reakcji na zdarzenia	Obsługa skryptów i programów Java
Obsługa specyficznego przetwarzania	
Moduł programów użytkownika	Obsługa skryptów i programów Java

Część funkcji, które są specjalnie wydzielone w systemach SCADA, jak np. obsługa zdarzeń czy przetwarzanie specyficzne, wymagają dedykowanych aplikacji stworzonych przy użyciu specjalnych mechanizmów. Część funkcji współdzieli dany mechanizm wykorzystując go na różne sposoby. Wynika to z uniwersalności tych mechanizmów i samych przeglądarek.

Poza tym moduły w przeglądarce, a w szczególności mechanizmy Javy, należy potraktować jako interfejs programowy środowiska webowego (ang. *web*). Wówczas warstwy komunikacyjne, serwer, przeglądarka i jej mechanizmy stanowią abstrakcyjny system operacyjny maszyny rozproszonej. Powyższe porównanie odbywa się na różnych poziomach abstrakcji. Porównuje się możliwości specjalizowanego narzędzia działającego pod kontrolą danego systemu operacyjnego z możliwościami innego systemu operacyjnego. Czyli zrealizowane dedykowane funkcje z mechanizmami możliwości ich realizacji. Bardziej

poprawnym byłoby porównanie funkcji systemu SCADA z narzędziem służącym do generacji rozproszonych aplikacji wizualizacji przemysłowej. Mimo tego zbieżność funkcjonalna usługi WWW i systemów typu SCADA jest na tyle duża, iż wskazane jest przynajmniej określenie jej dziedziny zastosowań w kontekście przemysłowych systemów wizualizacyjno-nadzorczych.

Na rysunku 54 przedstawiono przykładową strukturę systemu do tworzenia rozproszonej aplikacji wizualizacyjnej.



Rys. 54 Proponowana struktura narzędzia do tworzenia rozproszonej aplikacji wizualizacji przemysłowej

Wykorzystanie takiego narzędzia opiera się o następujące etapy:

– Przygotowywanie wizualizacji przy użyciu klasycznego interfejsu użytkownika.
  - Definicja zmiennych, grafiki ekranów i ich elementów składowych,
  - definicja komunikacji,
  - definicja programów użytkownika itp.
– Generacja dokumentów dla serwera WWW.
  - Generacja dokumentów statycznych dla serwera WWW.
    - Pliki HTML,
    - dołączanie grafiki.
  - Generacja dokumentów dynamicznych dla serwera WWW.
    - Tworzenie lub dołączanie z bibliotek skryptów i programów CGI,

        - generacja formularzy i dołączanie grafiki.
    - Generacja dokumentów aktywnych dla serwera WWW.
        - Tworzenie lub dołączanie z bibliotek skryptów i apletów Java,
        - dołączanie aplikacji usług dodatkowych.
- Przesłanie plików do serwerów WWW.
- Dostęp do stworzonych stron WWW z poziomu przeglądarek.

Program narzędziowy może być uruchamiany na dowolnym komputerze, a pliki będące efektem jego pracy mogą być przesyłane na serwer przy wykorzystaniu usługi FTP. Zasada działania całego interfejsu użytkownika mogłaby być oparta na interfejsach współczesnych pakietów developerskich systemów typu SCADA. Dobrze skonstruowany interfejs powinien działać tak, aby ukrywać mechanizmy WWW, nie pozwalając jednocześnie na tworzenie funkcji i elementów nie dających się zaimplementować w technologiach WWW. Dobrym rozwiązaniem byłoby stworzenie modułu generatora aplikacji dla danego istniejącego systemu typu SCADA. Projektanci oraz twórcy aplikacji mogliby dzięki temu pracować w znanym sobie środowisku mając jednocześnie możliwość wygenerowania aplikacji standardowej lub opartej o technologię WWW.

10.2.1.2. Sposoby prezentacji danych w usłudze WWW

Istnieją trzy podstawowe typy dokumentów WWW umożliwiających prezentację informacji i interakcję z użytkownikiem. Są to dokumenty:

- statyczne (obsługa danych przy użyciu języka HTML [43]),
- dynamiczne (obsługa danych przy użyciu skryptów np. CGI [11]),
- aktywne (obsługa danych przy użyciu technologii Java [65, 42, 41, 75, 100], ActiveX [62, 81] itp.).

Przy wizualizacji przemysłowej program CGI może pobierać dane od systemu i modyfikować zawartość przekazywanej klientowi informacji przy każdorazowym odświeżeniu strony. Połączenie elementów statycznych z elementami generowanymi przez CGI daje już skuteczny sposób prezentacji stanu procesu. Uzyskujemy możliwość obsługi zarówno elementów statycznych jak i animacyjnych i kontrolnych [23]. Elementy kontrolne mogą być realizowane przez wykorzystanie mechanizmu formularzy w połączeniu z mechanizmem krótkoterminowego przekazywania informacji o stanie [43, 107, 111, 112]. Daje to możliwość budowania stron WWW jako dialogów parametryzacyjnych dla procesu.

W przypadku stosowania technik aktywnych, program klienta WWW może stać się klientem nowych usług definiowanych na potrzeby aplikacji. Zawartość dokumentu aktywnego może się zmieniać w sposób ciągły bez konieczności retransmisji jego opisu. Dla prostych stacji monitorujących wystarczającym jest mechanizm cyklicznego wypychania [13]. Jednak, gdy strona zawiera dużo informacji w formie graficznej, metoda ta powoduje nieprzyjemne miganie ekranu, a w przypadkach zwiększonych opóźnień transmisji staje się uciążliwa. Wówczas idealna okazuje się technologia Java lub ActiveX. Dzięki lokalnemu sterowaniu wyglądem strony, strona może niczym nie różnić się od dedykowanej aplikacji

uruchomionej lokalnie. Różnica polega na konieczności wypracowywania stanów elementów animacyjnych na podstawie danych uzyskiwanych z połączenia TCP/IP.

Podsumowując można zauważyć duże podobieństwo pomiędzy typami dokumentów WWW i rodzajami elementów stosowanych przy wizualizacji przemysłowej [23]. Technika dokumentów aktywnych w połączeniu z dokumentami statycznymi stanowi skuteczną metodę prezentacji danych w systemach wizualizacji przemysłowej opartej o usługę WWW. Jednak technika ta dając szerokie pole do aplikowania specyficznych usług stwarza również zagrożenia z punktu widzenia bezpieczeństwa systemu.

10.2.1.3. Sposoby dostarczania danych w usłudze WWW

Usługa WWW działa w sieci Internet, a więc korzysta z protokołów TCP/IP, a konkretnie z protokołu HTTP [107]. W warstwie niższej transfer danych zachodzi z wykorzystaniem protokołu TCP [103]. Protokół HTTP, tak jak wiele innych usług Internetowych, działa w oparciu o model klient-serwer. Serwerem jest zawsze specjalizowany abonent udostępniający i uaktualniający dokumenty, natomiast klientem jest oprogramowanie przeglądarki (ang. *browser*).

Praca serwera polega na oczekiwaniu w pętli na zgłoszenie żądania transferu dokumentu od klienta. Gdy takie żądanie nadejdzie, serwer przesyła dany dokument, zamyka połączenie i czeka na kolejne żądania. Znacznie bardziej złożona jest konstrukcja aplikacji klienta. Ponieważ na transfer dokumentu hypertekstowego może składać się wiele żądań transferu jego elementów składowych, przeglądarka musi składać się z całego zestawu klientów, interpreterów, a także modułów zarządzania i interakcji z użytkownikiem. Dzieje się tak dlatego, gdyż dokument hyperteksktowy jest dokumentem multimedialnym. Może się on składać z szeregu elementów typu tekst, obraz, dźwięk oraz elementów dynamicznych typu skrypty, odnośniki czy programy wykonywalne. Protokół HTTP definiuje transfer takiego dokumentu jako szereg transmisji jego elementów składowych. Po odebraniu elementu połączenie z serwerem zostaje zamknięte, i o ile istnieje potrzeba pobrania następnego elementu znowu otwarte jako kolejna transakcja.

Upraszczając protokół HTTP działa podobnie do protokołu FTP, lecz oprócz standardowego przesłania pliku na żądanie klienta, protokół umożliwia przesyłanie danych generowanych dynamicznie przez aplikacje pracujące na serwerze oraz przetwarzanie w tych aplikacjach danych pobranych z aplikacji klienta.

10.2.1.4. Przykład webowej usługi wizualizacyjnej

Na rysunku 55 przedstawiono przykład mechanizmu wizualizacji opartego na strukturze dwuwarstwowej. Wizualizacyjna stacja kliencka posiada przeglądarkę WWW z zestawem klientów realizujących różne usługi dostarczania i transmisji danych a także wyświetlania dokumentów WWW i interakcji z nimi. W procesie przemysłowym pracuje serwer lub grupa serwerów stanowiących źródła informacji dla wizualizacji oraz miejsce docelowe do zapisu parametryzacji i rozkazów dla procesu. Dodatkowo aplikacje wykonywalne utworzone przez

lokalny system wykonawczy języka Java, mogą komunikować się z innymi aplikacjami w intersieci. Cała warstwa komunikacyjna zrealizowana jest w oparciu o intersieć. W podanym przykładzie podniesienie wydajności systemu osiąga się poprzez odpowiednie scalenie różnych dokumentów WWW. Elementy statyczne ekranów pobierane są przez odczyt dokumentów statycznych, elementy kontrolne obsługiwane są przez skrypty CGI lub skrypty Java. Natomiast dla elementów wymagających stałego odświeżania jak wykresy, animacje itp. Stosuje się bezpośrednie połączenia komunikacyjne ustalane przez aplikacje wytworzone z dostarczonych siecią apletów języka Java. Uzyskuje się prezentację, która przy inicjalizacji transmituje całą postać ekranu, natomiast w czasie pracy dostarcza tylko te dane, na podstawie których sterowane są elementy animacyjne. Oczywiście istnieje obecnie cały szereg innych usług (Flash, ASP, XML, DHTML, .NET) [65, 46, 10, 42, 41, 75, 98, 91, 7, 71, 1, 84, 8], które można wykorzystać w tego typu rozwiązaniach. W niniejszej pracy starano się bazować jednak na rozwiązaniach najbardziej popularnych i co do standaryzacji których nie ma wątpliwości.

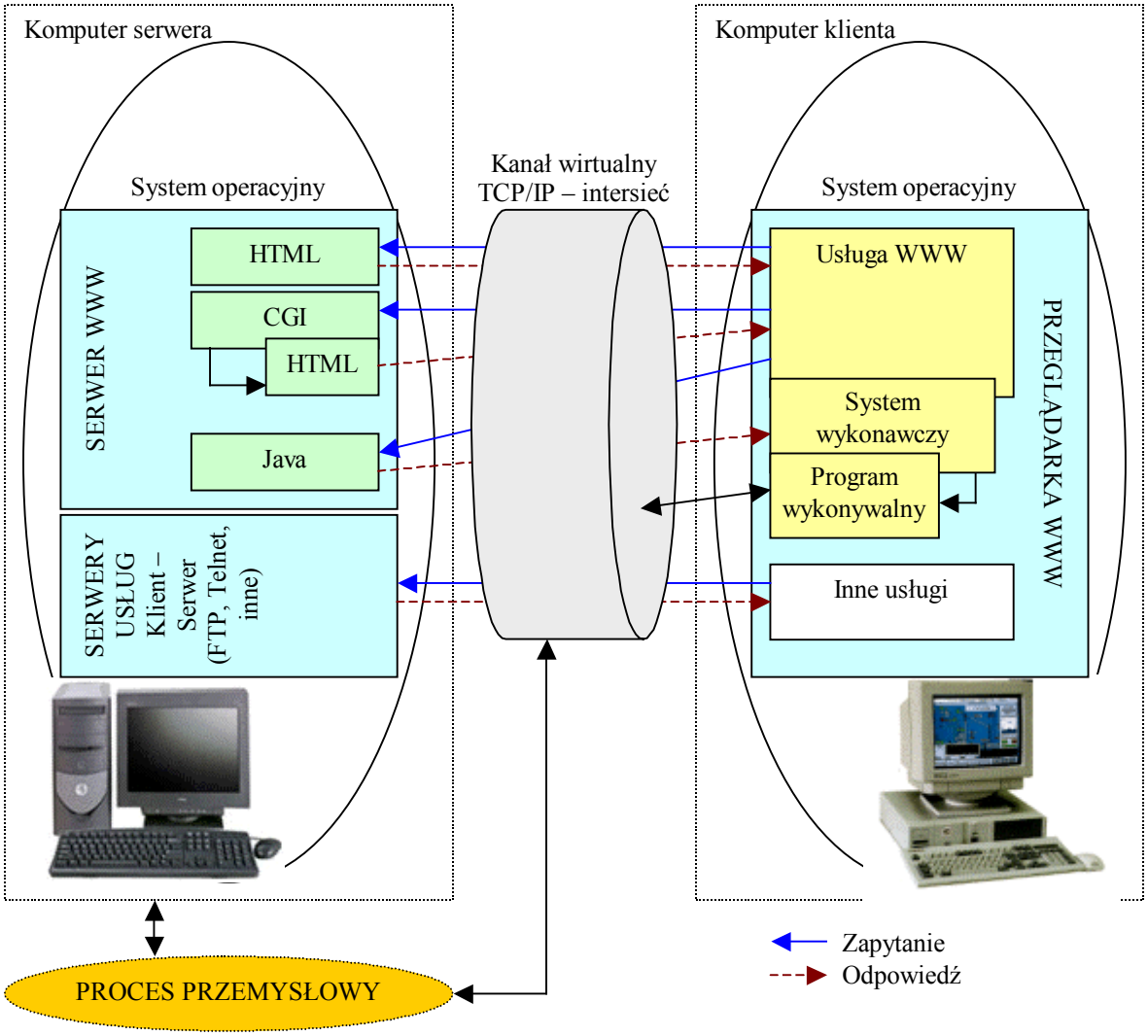
Struktury trójwarstwowe aplikacji przeniesione na płaszczyznę usługi WWW umożliwiają dalsze funkcjonalne rozdzielenie elementów systemu. Warstwa obsługi danych może zostać powiązana z abonentem fizycznie związanym z systemem lokalnym. Jego zadanie zostanie ograniczone do zapewnienia współpracy pomiędzy bazą danych zmiennych lokalnych a rozproszoną bazą danych zmiennych systemu zdalnego. Warstwa pośrednia tzw. reguł decyzyjnych lub biznesowych może znajdować się na odrębnym serwerze nie związanym bezpośrednio z procesem. Warstwa trzecia będzie natomiast tak jak poprzednio warstwą prezentacyjną opartą o przeglądarkę WWW. Pojawienie się warstwy pośredniczącej realizowanej na odrębnym fizycznym urządzeniu spowoduje:

- ułatwienie zarządzania całym procesem dostępu zdalnego,
- zwiększenie bezpieczeństwa dostępu do abonentów procesowych,
- uniezależnienie warstw aplikacji abonenta procesowego od aplikacji obsługującej zdalny dostęp,
- zwiększenie skalowalności rozwiązania po przez możliwość dopasowywania modułów składowych,
- możliwość rozpraszania warstwy danych.

Architektura trójwarstwowa jest bez wątpienia bardziej złożona i wymaga więcej czynności serwisowych niż aplikacja dwuwarstwowa. Jej stworzenie i przeprowadzenie konfiguracji jest także bardziej kosztowne. Jednak dla systemów, gdzie wiele zmiennych procesowych powinno być udostępnianych w intersieci a kontrola bezpieczeństwa powinna być maksymalnie szczelna, takie rozwiązanie jest korzystniejsze.

Kontrola dostępu do danych przesyłanych z jak i do serwera WWW jest rozwiązana dość dobrze. Ponieważ przed dziedziną przemysłu, intersieciowym dostępem zainteresowała się dziedzina handlu i bankowości, standardowe mechanizmy uwierzytelniania i autoryzacji użytkownika oraz szyfrowania danych [4, 3], są w protokołach TCP/IP mocno

zaawansowane. Transmisja danych w intersieci przebiega przez sieć wirtualną, a więc właściwie nie wiadomo gdzie i którędy, i tym samym jest narażona na ataki osób postronnych z każdego miejsca na świecie.



Rys. 55 Struktura obiegu informacji w dwuwarstwowej usłudze wizualizacji przez WWW

10.2.1.5. Podsumowanie wizualizacji w intersieci

Usługa WWW udostępnia szerokie możliwości prezentacji danych, nie ustępujące typowym aplikacjom kontrolno-nadzorczym. Za najbardziej przydatną dla wizualizacji systemów przemysłowych proponuje się technologię Java. Technologia ta jest stale rozwijana, również pod kątem systemów czasu rzeczywistego. Grupa pod nazwą RTJEG (ang. *The Real-Time for Java Expert Group)*, stworzona w ramach programu Java Community Process stale zajmuje się rozwojem technologii Java w kontekście tworzenia aplikacji czasu rzeczywistego. Grupa ta, działająca pod auspicjami The National Institute for Standards and Technology (NIST) ze Stanów Zjednoczonych, przyjęła na siebie obowiązek stworzenia specyfikacji dla rozszerzenia języka Java oraz specyfikacji maszyny wirtualnej Javy. W wyniku swoich prac udostępniono interfejs programowy, który umożliwia tworzenie, weryfikację, analizę,

wykonywanie, oraz zarządzanie specjalnymi wątkami Javy, które uwzględniają ograniczenia czasowe wynikające z pracy w czasie rzeczywistym (tzw. wątki czasu rzeczywistego) [100].

Ograniczenia prezentacji wynikają z faktu, że elementy statyczne ekranu lub program je obsługujący muszą zostać przetransmitowane siecią a nie stanowią integralnej części oprogramowania klienckiego. Przeglądarki są przystosowane z założenia do obsługi odwołań do dokumentów zdalnych. Założeniem WWW jest rozproszenie informacji. W systemach wizualizacji przemysłowej większość elementów ekranu, czyli wyświetlanej strony, może być przechowywana lokalnie, gdyż mają charakter statyczny. Jedynie dane służące do parametryzacji wyglądu elementów animacyjnych lub dane związane z elementami kontrolnymi muszą być transmitowane przez sieć. Problem braku lokalności odwołań może być rozwiązany w najprostszy sposób przez użycie mechanizmów buforowania i składowania lokalnego danych. Mechanizm pamięci podręcznej (ang. *cache*) przeglądarek mający standardowo za zadanie tymczasowo zwiększyć lokalność odwołań może znacząco poprawiać tę niedogodność.

W porównaniu możliwości względem standardowych stacji typu SCADA, mechanizmy HTML, CGI, Java i inne gwarantują stworzenie interfejsu użytkownika na poziomie porównywalnym lub nawet wyższym niż rozwiązania tradycyjne. Niestety parametry czasowe rozwiązania opartego o standardowy interfejs przeglądarki WWW będą znacznie gorsze. Aktualizacja elementów kontrolnych oraz animowanych zależy w głównej mierze od warstwy komunikacji. Stabilność cyklu oraz opóźnienia generacji zależą również od warstwy oprogramowania systemowego oraz konkretnej implementacji aplikacji klienta.

Główne problemy, jakie mogą się pojawić w realizacji wizualizacji to:

- brak mechanizmu priorytetowej obsługi wybranej grupy informacji stanowiącej użyteczną informację procesową,
- konieczność transferu kompletnego środowiska graficznego nie stanowiącego użytecznej informacji procesowej,
- pełna zależność od pracy systemu operacyjnego,
- zależność od pracy innych procesów,
- brak możliwości priorytetyzacji zadań,
- brak możliwości osadzania mechanizmów użytkownika w jądrach systemów.

Wszystkie te braki wynikają ze specyfiki protokołu lub z faktu, iż twórcy oprogramowania przeglądarek nie optymalizują ich pod kątem zadań funkcjonalnych wizualizacji przemysłowej. Mimo wad i ograniczeń, usługa WWW dostarcza szereg możliwości, których brak w systemach standardowych:

- eliminacja specjalistycznego oprogramowania stacji nadzorczej,
- obniżenie kosztów,
- rezygnacja ze specjalistycznego oprogramowania stacji SCADA,
- rezygnacja z konfiguracji i parametryzacji stacji klienta,
- funkcja częściowej mobilności,

– niezależność parametrów czasowych transmisji od odległości od systemu lokalnego.

10.2.2. Raportowanie

Raportowanie ma na celu przekazywanie użytkownikowi informacji podsumowujących stan kontroli procesu. Jest to akcja przeprowadzana jednokierunkowo od systemu do użytkownika. Usługę intersieci, która może być stosowana do realizacji tej funkcji stanowi poczta elektroniczna.

W systemie lokalnym usługa przesyłania poczty elektronicznej, nie nadaje się dla wymian poziomych. Przesyłanie poczty elektronicznej ma znaczenie w transmisjach pionowych, tylko dla płaszczyzn abonent→serwer_poczty.

W systemie zdalnym może występować automatyczne przesłanie komunikatu z serwera do abonenta zdalnego. Efektem będzie transfer raportu od abonenta systemu lokalnego do nadzorcy, administratora lub innego użytkownika systemu znajdującego się w intersieci. Obiektem docelowym transmisji staje się skrzynka pocztowa danego użytkownika lub grupy użytkowników. Niezależnie od swojej lokalizacji użytkownik taki po przez dostęp do Internetu może uzyskiwać dostęp do tych informacji.

Najprostszą realizacją wymiany danych opartej o pocztę elektroniczną, będzie instalacja modułu klienta poczty w stacji nadzorczej typu SCADA. Wówczas warstwa aplikacji może przygotować i wysłać komunikat bezpośrednio do intersieci adresując datagramy na porty usługi pocztowej do serwera obsługującego daną skrzynkę. W przypadku, gdy komunikat miałby być wysłany przez abonenta separowanej sieci systemowej, dane użyteczne komunikatu należy tunelować przez sieć systemową i przeadresować w warstwach *Firewalla++* abonenta separującego. Działanie takie jest skuteczne niezależnie od rodzaju protokołu pracującego na sieci wewnętrznej. Zastosowanie usługi nie ma nic wspólnego z transmisją danych obarczonych czasem krytycznym, a zatem jej wykorzystanie może mieć miejsce w sferach funkcjonalnych systemu nie wymagających determinizmu wymian. Natomiast usługa z samej swojej definicji zawiera mechanizmy kontroli dostępu i identyfikacji użytkownika, co niezwykle ułatwia zachowanie poufności transmitowanych danych wskazanej dla funkcji raportowania.

10.2.3. Monitorowanie

Funkcja monitorowania polega na zdalnym śledzeniu wybranych parametrów procesowych i ich analizie. Monitorowanie może odbywać się w trybie *on-line* (w czasie rzeczywistym) lub *off-line* (po fakcie). Do realizacji tego typu funkcji proponuje się wykorzystanie usług intersieci FTP oraz Telnet.

Usługa FTP jest właściwie protokołem warstwy aplikacyjnej. Dokumentacja RFC959 nie definiuje interfejsu użytkownika, lecz wiele implementacji tej usługi korzysta ze standardowego powszechnie znanego zbioru poleceń. Usługa ta jest użyteczna praktycznie dla każdego typu płaszczyzn wymian. Dotyczy ona transferu plików, czyli zbiorów danych. Zakładając, że każdy abonent posiadający wystarczająco rozbudowaną warstwę aplikacji,

potrafi tworzyć abstrakcyjne zbiory danych, należy przyjąć, iż o ile istnieje potrzeba transmisji tych danych do stacji odległych, można tego dokonać wykorzystując transfer plików. Jest to wygodne z tego względu, że do samego procesu pobierania danych użytkownik nie potrzebuje specjalistycznego oprogramowania, a jedynie standardowego oprogramowania klienta FTP. Idealne zastosowanie tej usługi ma miejsce w monitorowaniu *off-line*, gdzie abonent lokalny gromadzi dane w pliku np. zapisy trendów wybranych parametrów, a abonent zdalny korzystający z intersieci podłącza się usługą FTP do systemu lokalnego i pobiera dane do wglądu lub dalszej obróbki.

Praktyczna implementacja będzie najprostsza, tak jak poprzednio, przy wykorzystaniu abonenta pośredniczącego jako serwera FTP. Dla bezpośredniego wykorzystania abonenta systemu lokalnego niezbędne byłoby przeadresowywanie usługi w *Firewallu++* lub tunelowanie danych. Inne podejście, bezpieczniejsze z punktu obcej widzenia ingerencji abonentów obiektowych, polega na uruchamianiu klientów FTP na płaszczyznach tychże abonentów, a serwer traktować jako stację zdalną.

Podobnie jak dla poczty elektronicznej, zastosowanie usługi nie ma nic wspólnego ze zdeterminowaną w czasie transmisją danych. Wykorzystanie usługi może mieć miejsce w transmisjach nie wymagających determinizmu. Również w kwestii bezpieczeństwa dostępu i identyfikacji użytkownika usługa posiada mechanizmy zachowania poufności udostępnianych danych. Jednak usługa nie jest odporna na monitorowanie danych, gdyż nazwa użytkownika i hasło przesyłane są w postaci jawnej. Dodatkowym zabezpieczeniem może być uruchamianie usługi na innym porcie niż standardowy 21 lub wykorzystywanie niestandardowych aplikacji klientów i serwerów.

Alternatywnym rozwiązaniem monitoringu jest wykorzystanie usługi Telnet [104]. Telnet stanowi nazwę protokołu aplikacyjnego umożliwiającego wszystkim zdalną interakcyjną pracę na odległym komputerze przy wykorzystaniu wirtualnego terminala znakowego. Zastosowanie tego protokołu ma sens dla wymian pionowych, ze względu na możliwość definiowania usługowego interfejsu użytkownika. Protokół definiuje mechanizm komunikacji interakcyjnej nie definiując samej usługi. Możliwe i wskazane jest zatem wykorzystywanie tego protokołu do implementacji komunikacji w różnych niestandardowych usługach. Inicjując połączenie telnetowe, w większości systemów można podać opcjonalnie numer portu[7] inicjując automatycznie tym samym połączenie z inną usługą niż standardowa usługa zdalnego terminala.

Dla przykładu polecenie:

```
telnet top.iinf.polsl.gliwice.pl
```

spowoduje podłączenie do usługi zdalnego terminala na domyślnym porcie 23, natomiast polecenie:

```
telnet top.iinf.polsl.gliwice.pl 110
```

[7] standardowo Telnet łączy się z portem 23 dla usługi zdalnego terminala

spowoduje podłączenie do usługi serwera pocztowego standardowo pracującego na porcie 110. W tym przypadku następuje wykorzystanie Telnetu do zdalnego dostępu do usługi poczty elektronicznej.

Mając zatem dostęp do standardowego klienta protokołu Telnet, można korzystać z różnych usług zainstalowanych na serwerze, które współdziałają z tym protokołem. Tworząc klienta i serwera specjalizowanego i można utworzyć specjalizowaną usługę realizującą dowolny zbiór zadań w ramach dziedziny wynikającej z ograniczeń niezdeterminowanej czasowo komunikacji.

Najprostszym, najwygodniejszym i najtańszym rozwiązaniem jest stworzenie aplikacji pracującej na serwerze, która wykorzystując Telnet w powiązaniu z danym portem (najlepiej niestandardowym) będzie realizować usługę dostępu do wybranej grupy danych procesowych, oraz będzie zgodnie z akcją użytkownika oddziaływać na tą lub inną grupę. Aplikacja taka może stanowić proces uruchamiany zdalnie na żądanie ze zdalnego terminala i z tymże terminalem związać swoje strumienie wejścia-wyjścia użytkownika. Możliwe jest również uruchomienie takiej aplikacji jako serwera dodatkowej usługi.

Dodatkowo możliwa jest implementacja mechanizmu określania jakości pozyskiwanych danych. Istnieją się dwie możliwości realizacji takiej usługi. Pierwsza określa (nie zapewnia) czasową spójność danych poprzez transmisję stempla czasowego wraz z danymi użytecznymi, druga określa stabilność cyklu odczytu danych na stacji klienta. Niestety obie wiążą się niedogodnościami. Pierwsza wymaga synchronizacji zegarów nadawcy i odbiorcy, natomiast druga wiąże się z koniecznością posiadania specjalizowanej aplikacji klienta. W przypadku stempla czasowego, czas wytworzenia wartości zmiennej powinien być dołączony do pakietu przenoszącego samą wartość, jednak aby poprawnie interpretować wartość stempla na stacji odbiorczej niezbędne są urządzenia umożliwiające synchronizację zegarów z dokładnością wynikającą z cyklu transmisji zmiennej. Jako takie urządzenia mogą posłużyć zegary DCF synchronizowane radiowo z wzorca zegara atomowego. W przypadku testowania stabilności cyklu, jesteśmy w stanie określić „świeżość" zmiennej cyklicznej. Mechanizm powinien funkcjonować mierząc różnicę czasu pomiędzy nadejściem kolejnych pakietów z daną zmienną. Na podstawie wyliczonej różnicy oraz zadeklarowanego czasu życia wartości zmiennej można dokonać binarnego określenia flagi aktualności czasowej dla danej wartości.

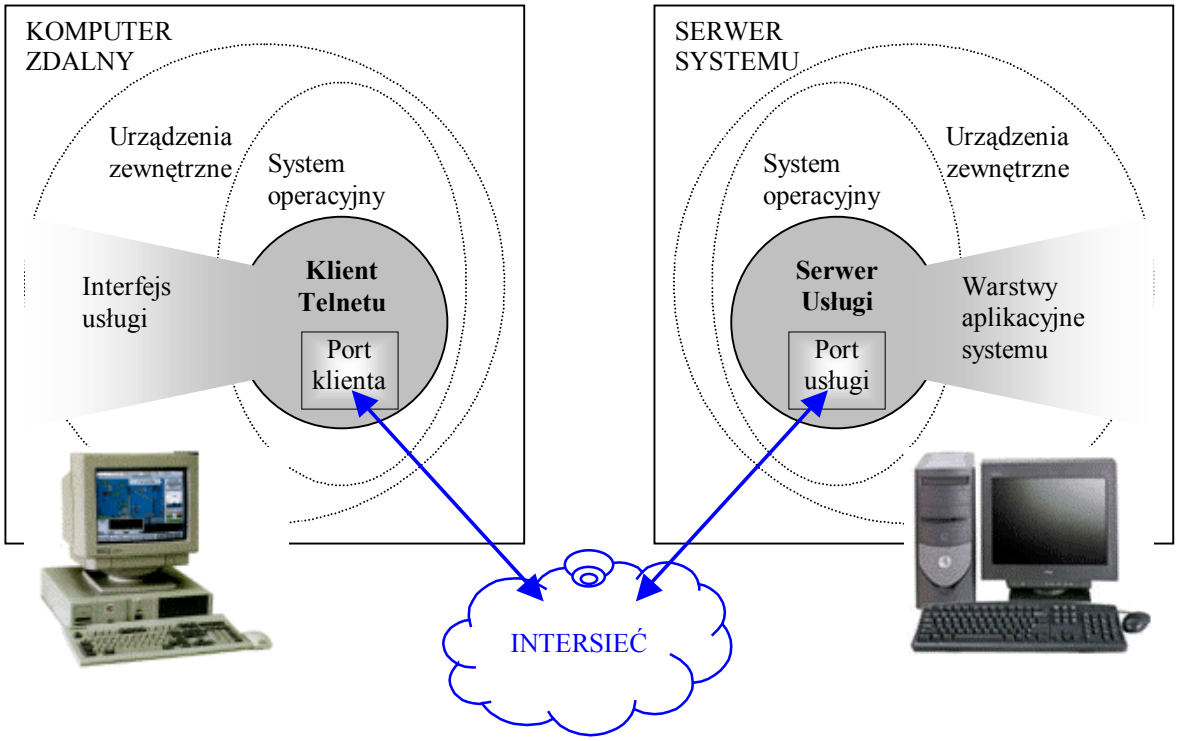
Określanie stabilności cyklu jest możliwe tylko dla zmiennych cyklicznych. W przypadku zmiennych aperiodycznych potrzebna jest informacja od producenta zmiennej, aby móc określić na ile zmienna została opóźniona przez warstwy komunikacyjne. Również w przypadku zachwiania cyklu, bez stempla czasowego można określić jedynie, że pakiety nie pojawiają się z zamierzoną częstotliwością. Aby określić w czasie, kiedy dana wartość została wytworzona niezbędny jest stempel czasowy od nadawcy. Opisywane mechanizmy dotyczą innego charakteru określania jakości danych niż w przypadku opisywanego wcześniej określania na poziomie integracji systemu izolowanego z intersiecią. Nie ma tutaj do czynienia z przekazywaniem danych z jednego otwartego systemu komunikacyjnego do

drugiego separowanego, a jedynie z przesyłem pomiędzy abonentami w obrębie intersieci otwartej pracującej na bazie TCP/IP. Podobne mechanizmy określania świeżości danych funkcjonują w firmowych rozwiązaniach na przykład w protokole SuiteLink [95].

W usługach wykorzystujących protokół Telnet, mechanizmy identyfikacji, kontroli dostępu czy też szyfrowania transmisji, ze względu na ich brak w samym protokole, mogą lub też muszą być wbudowane bezpośrednio w usługę.

Wykorzystanie Telnetu do zdalnego monitorowania parametrów procesu typu *on-line* może odbyć się niewielkim kosztem. Rozwiązaniem takie może bazować na usłudze zdalnego terminala znakowego i specjalizowanego programu uruchamianego zdalnie na serwerze. Dla tworzenia nowych, specjalizowanych usług, niezbędna jest modyfikacja interfejsów komunikacyjnych abonentów, szczególnie w warstwach aplikacji użytkownika.

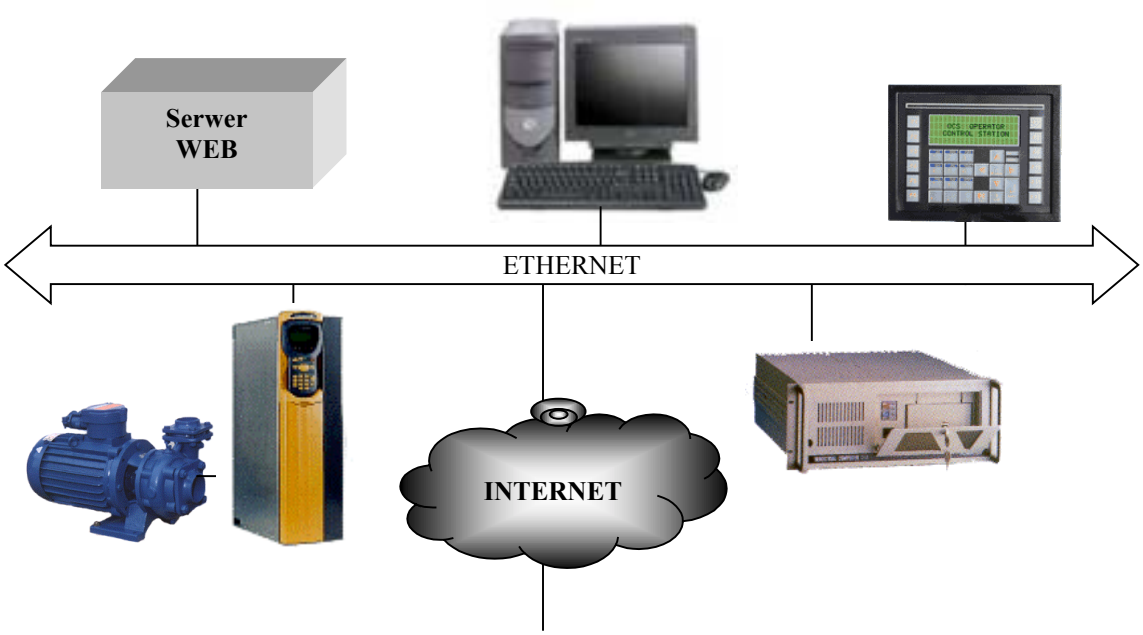
Schemat systemu komunikacyjnego opartego o aplikacje klienta i serwera protokołu Telnet pokazany jest na rysunku 56. Praktyczna implementacja mechanizmu opisana jest w załączniku VI.A.2. Przedstawiono tam również dodatkowe aspekty adaptacji protokołu Telnet do zastosowań w systemach informatycznych pracujących w przemyśle oraz wyniki testów dla zaimplementowanego mechanizmu określania stabilności cyklu pobierania danych.



Rys. 56 Wykorzystanie protokołu Telnet

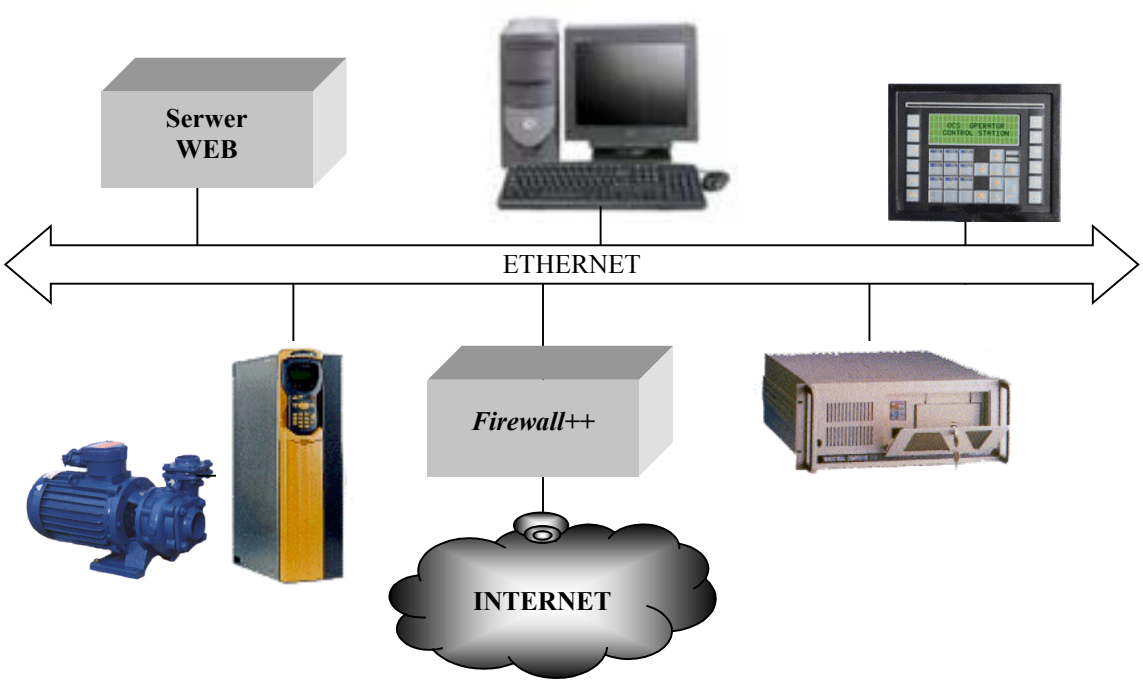
10.3. Sposoby konstruowania systemu lokalnego

Praktyczna realizacja systemu wykorzystującego usługi WWW wymaga umieszczenia w systemie serwera WWW zwanego często serwerem webowym (z ang. *web*). Przykład takiej struktury dla systemu typu otwartego pokazany jest na rysunku 57.



Rys. 57 Przykładowa struktura systemu otwartego z wykorzystaniem serwera WEB

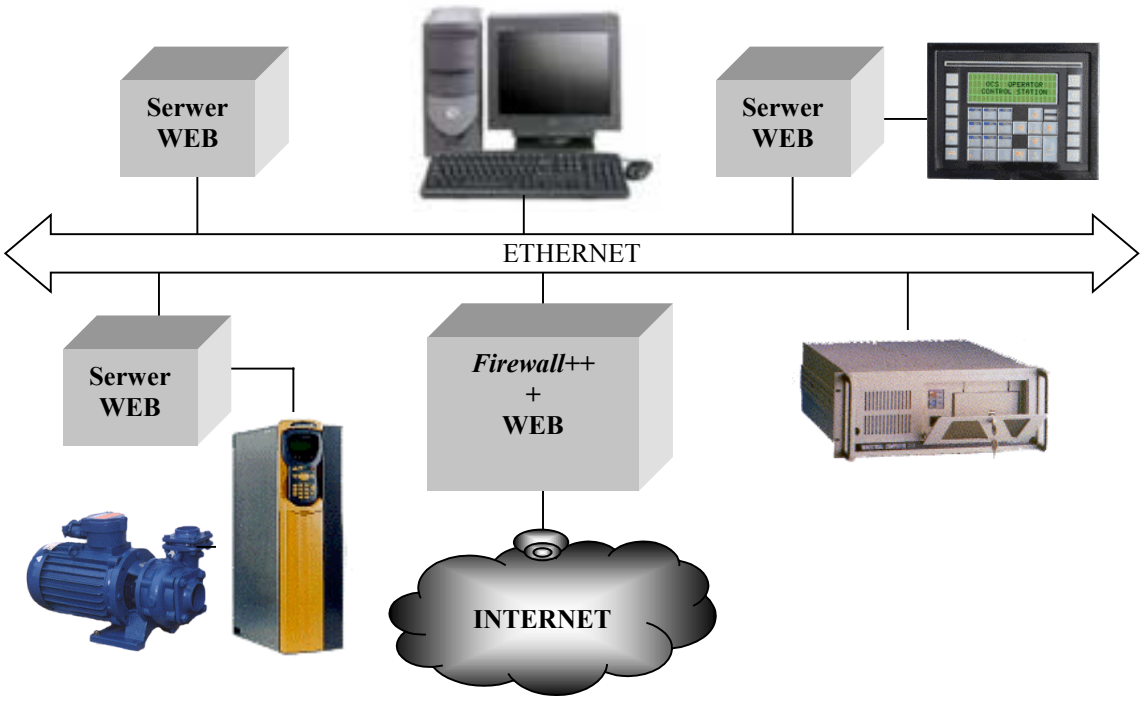
W przedstawionym przykładzie urządzenia przemysłowe pracują na segmencie sieci Ethernet stanowiącej sieć lokalną. W sieci funkcjonuje urządzenie spełniające funkcję serwera WWW. Może to być komputer, moduł sterownika, urządzenie specjalizowane lub tzw. serwer wbudowany (ang. *embedded*) [47]. Sieć Ethernet posiada połączenie z Internetem. Sposób podłączenia nie jest rozpatrywany. Serwer jako urządzenie przystosowane do zbierania informacji z systemu i publikowania ich w postaci dokumentów hypertekstowych, umożliwia, iż dowolna stacja komputerowa systemu lokalnego lub zdalnego wyposażona w przeglądarkę WWW może te dokumenty przeglądać oraz przesyłać zwrotnie informacje do serwera. Stanowi to rozwiązanie niebezpieczne ze względu na cechy otwartości (rozdz. 7.2.2).



Rys. 58 Przykładowa struktura systemu z *Firewallem++*

Bezpieczniejszą strukturę stanowi struktura przedstawiona na rysunku 58. Dzięki *Firewallowi++* uzyskuje się ruch separowany, umożliwiając dostęp np. tylko do portu 80 serwera WEB. Możliwe staje się stworzenie nadbudowy aplikacyjnej do kontroli procesu realizacji wymian. Mając kontrolę nad warstwami aplikacji we wszystkich urządzeniach podłączonych do sieci można stworzyć komunikację zdeterminowaną czasowo.

Kolejną propozycję rozwiązania struktury sytemu lokalnego stanowi eliminacja serwera WEB jako osobnego urządzenia i rozproszenie jego funkcji w sieci dodając funkcję serwera do *Firewalla++*. Przykład struktury systemu separowanego przedstawiony jest na rysunku 59. Rozproszone serwery webowe obsługują daną grupę urządzeń lub pomiarów udostępniając dane serwerowi pośredniczącemu. Uzyskuje się koncepcyjnie ciekawsze rozwiązanie, a także dające większe możliwości dostępu do poszczególnych elementów systemu. Podobnie jak w poprzednim przykładzie istnieje możliwość stworzenia sieci systemowej o zdeterminowanym w czasie dostępie do medium.



Rys. 59 Przykładowa struktura systemu izolowanego z rozproszonymi serwerami WEB

Istnieje jeszcze jedna możliwość budowy systemu lokalnego w oparciu o usługi WWW. Polega on na połączeniu funkcji stacji systemu SCADA, serwera WEB oraz *Firewalla++* w jednym abonencie pośredniczącym. Rozwiązanie takie jest najbardziej uniwersalne.

10.4. Zagadnienia bezpieczeństwa

Bezpieczeństwo transmisji w przemysłowych systemach informatycznych dotyczy przede wszystkim nie tyle kontrolowanego dostępu do informacji, co gwarancji, iż dane dotrą do adresata w żądanym czasie i nie będą przekłamane. Dodatkowy element tak pojętego bezpieczeństwa stanowi informacja, iż dane zostały wytworzone i zinterpretowane w czasie określonym pojęciem „czasu życia danych". Współczesne komputerowe sieci przemysłowe,

muszą mieć wbudowane określone mechanizmy sprzętowe lub programowe zapewniające żądany poziom bezpieczeństwa transmisji. Owe mechanizmy to nie tylko poprawnie zaprojektowane urządzenia wyposażone w sprawdzone oprogramowanie, ale również protokoły transmisji [31, 44].

Rozważanie aspektów bezpieczeństwa wykorzystania TCP/IP w informatycznych systemach przemysłowych w kontekście determinizmu jest bezcelowe. Zagadnienia bezpieczeństwa analizowane w gospodarce elektronicznej (ang. *e-commerce*), która bazując na Internecie korzysta z protokołów TCP/IP, nie znajdują zastosowania w systemach przemysłowych. Niektóre zagadnienia zabezpieczania obiektów teleinformatycznych z tej dziedziny jak uwierzytelnianie, autoryzacja i szyfrowanie na pewno znajdą zastosowanie w systemach przemysłowych. Jednak sposób wykorzystywania sieci w rozwiązaniach *e-commerce* jest odmienny od wymagań systemów przemysłowych [31, 39]. Gospodarka elektroniczna oznacza zaproszenie swoich klientów i partnerów do zakładowego Intranetu. Natomiast w struktury komunikacyjne systemów kontroli procesów przemysłowych powinni mieć wgląd tylko abonenci związani z prowadzeniem i nadzorem tego procesu. Musi pojawić się zatem bardziej precyzyjna kontrola, i to nie tylko na poziomie zasobów lecz także na poziomie medium.

Poniżej przeprowadzono rozważania zagrożeń bezpieczeństwa systemu abstrahując jednocześnie od zagrożeń wynikających z braku determinizmu, biorąc pod uwagę dwie grupy elementów wynikających z dwóch podstawowych kryteriów, jakimi są:

- charakter upublicznienia intersieci,
- określenie poziomu zaufania, ryzyka i kompetencji dla użytkowników.

Problemy wewnętrzne sieci	*Problemy zewnętrzne sieci*
Zagrożenia związane z medium.	Utajnianie danych procesowych.
Zagrożenia związane z arbitrażem wymian.	Określenie praw dostępu do funkcji.
Zagrożenia na poziomie poszczególnych warstw oprogramowania interfejsów komunikacyjnych.	Określenie praw dostępu do danych.
Zagrożenia związane z poprawnością transmisji informacji użytecznych.	Określenie zabezpieczeń dostępu do abonentów.
Zagrożenia związane z jakością informacji użytecznych.	Określenie zabezpieczeń dostępu dla użytkowników.

Struktura sieci określa nam podstawowe grupy zagrożeń. Określenie, w jakich systemach (7.2) pracuje analizowana sieć jest niezbędne dla właściwej oceny zagrożeń. Istnieją zatem dwie grupy problemów. Grupa pierwsza – problemów wewnętrznych, określa problemy wynikające z działania samej sieci, jej topologii, struktury, idei realizacji transakcji i wymian danych. Grupa druga – problemów zewnętrznych, dotyczy tylko takich sieci, do których dostęp ma charakter publiczny lub ocena poziomu ryzyka w systemach zamkniętych lub Intranecie wskazuje na istnienie tego typu zagrożeń.

10.4.1. Zagrożenia wewnętrzne

Zagrożenia wewnętrzne wiążą się z działaniem samej sieci i odnoszą się do poszczególnych warstw, z których zbudowany jest interfejs komunikacyjny abonentów. W warstwie fizycznej należy założyć, iż uszkodzenie jest możliwe niezależnie od przyjętego rozwiązania. Zatem aby zwiększyć bezpieczeństwo funkcjonowania systemu należy rozważyć możliwość redundancji medium. Jest to rozwiązanie proste i skuteczne. Redundancja jest możliwa w obrębie sieci systemowych systemu (sieci sterujących, polowych itp.). W skali intersieci rolę redundancji przejmuje swobodna topologia umożliwiająca kierowanie pakietów różnymi drogami (rozdział 9).

W warstwach wyższych, zajmujących się arbitrażem wymian problem jest podobny. Jeżeli mechanizm dostępowy oparty jest o dostęp swobodny[8], wówczas każdy abonent działa niezależnie. Jeżeli jednak istnieje zaimplementowany mechanizm deterministycznego nadzoru, wówczas należy uwzględnić możliwość redundancji takiego mechanizmu. Dla przykładu w modelu Master – Slave będzie to abonent potrafiący przejąć funkcję stacji Master a w modelu PDC dodatkowy dystrybutor działający w trybie nasłuchu. W skali sieci wirtualnej trudno rozważać zarządzanie wymianami. Porównywalną funkcję pełnią wówczas rutery, bramy i inne urządzenia kierujące dystrybucją pakietów. Swobodna topologia także w tej mierze zapewnia redundancję.

Zapewnienie poprawności transmisji z punktu widzenia wykrywania przekłamań wynikających z zakłóceń realizują odpowiednie warstwy wszystkich zaawansowanych protokołów komunikacyjnych. Wszelkie operacje tunelowania i kapsułkowania z jednej strony zwiększając ramkę powodują wzrost prawdopodobieństwa przekłamania poszczególnych bitów, jednak z drugiej strony każdy protokół wprowadzając swoje sumy kontrolne dodatkowo zabezpiecza dane użyteczne przed wystąpieniem błędów i utratą integralności.

Protokół TCP/IP w przeciwieństwie do specjalizowanych protokołów sieci przemysłowych składa się z wielu protokołów należących do różnych warstw modelu warstwowego. Abstrahując od zastosowań tych protokołów i traktując je jako współdziałające ze sobą moduły, można zauważyć, iż każdy z nich stanowi potencjalne zagrożenie z dwóch powodów. Po pierwsze wraz ze wzrostem złożoności programowej interfejsu rośnie również jego awaryjność. Po drugie każdy moduł ze zdefiniowanym zbiorem funkcji oraz kanałem wejścia stanowi potencjalny cel ataku użytkowników chcących przejąć kontrolę nad abonentem. Należy zatem stosować gruntownie przetestowane moduły i w minimalnej liczbie niezbędnej do realizacji zadań interfejsu. Jeżeli dla przykładu wykorzystuje się protokół Telnet, a nie ma potrzeby usługi WWW czy też FTP, to nie należy udostępniać portu z tymi usługami lub wręcz nie implementować nie wykorzystywanych protokołów. Udostępniając

[8] np. CSMA/CD w sieci Ethernet

jakąś usługę warto powiązać ją z nietypowym portem. Zabiegi te utrudnią potencjalnemu hackerowi lokalizację gniazd i rozeznanie w strukturze komunikacyjnej systemu.

Określanie jakości danych użytecznych w kontekście protokołu TCP/IP było omawiane w rozdziale 9.1. Mechanizm informacji statusowych lub stempli czasowych w systemach komunikacyjnych, gdzie występuje brak mechanizmów kontrolujących czas, a pracujących w systemach wymagających kontroli tegoż czasu jest niezbędny.

10.4.2. Zagrożenia zewnętrzne

Całość zagrożeń określanych jako zewnętrzne, wynika z istnienia możliwości zdalnego dostępu do systemu komunikacyjnego. Względem tych możliwości nasuwają się następujące zagrożenia:

- Wystąpienie przechwytywania danych tajnych i poufnych, wymuszające stosowanie mechanizmów ich utajniania, czyli szyfrowania.
- Wystąpienie nieautoryzowanego dostępu do urządzenia lub systemu komunikacyjnego, wymuszające stosowanie mechanizmów uwierzytelniania, autoryzacji, monitorowania i filtracji pakietów, ścian ogniowych lub segmentów buforujących.
- Wystąpienie przekazania kodu wirusa na poziom abonenta lub grupy abonentów systemu, wymuszające kontrolę pakietów mogących przenosić tego typu kod.

Z powyższych zagrożeń wynikają następujące potencjalne starty:

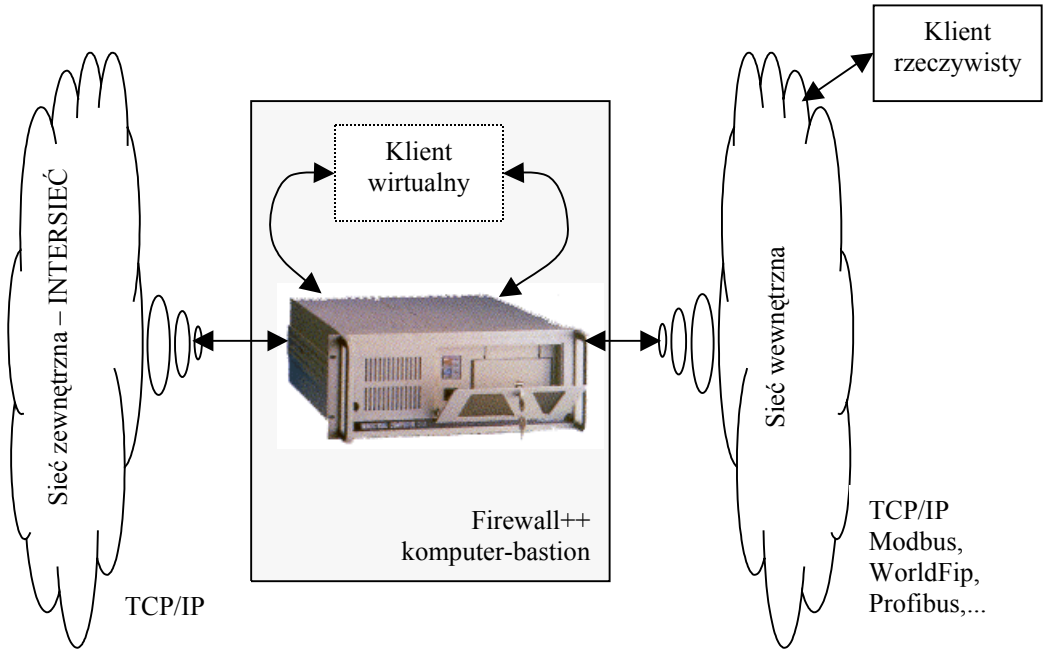
- kradzież danych objętych tajemnicą (np. chronione patentem procesy produkcyjne, parametryzacje procesu, programy technologiczne itp.),
- uszkadzanie urządzeń, produktów lub całych procesów itp. (niewłaściwa parametryzacja procesu, blokada łącz komunikacyjnych)
- wpływ na parametry produkcji (np. zmiana parametrów w celu zaniżenia jakości),

Większość zagrożeń można zminimalizować stosując mechanizmy uwierzytelniania, autoryzacji i szyfrowania. Jeżeli segment sieci jest podłączony do publicznej intersieci, to realne zagrożenie stanowi nieznany użytkownik posiadający umiejętności obejścia zabezpieczeń i wejścia do systemu. Dlatego w pełni zabezpieczony system przed zagrożeniami zewnętrznymi to system odłączony od intersieci.

Dodatkowo należy rozważyć jeszcze jedną opcję zagrożeń w zdalnym dostępie. Powszechnie panuje przekonanie, że zagrożenie może stanowić tylko klient dołączający się do serwera usługi systemu lokalnego. Nic bardziej mylnego. W celu na przykład kradzieży danych parametryzacyjnych procesu lub przechwycenia kodów dostępu, czy też błędnego informowania o stanie rzeczywistym, pod właściwy serwer związany z systemem lokalnym może podszyć się serwer fałszywy prowadząc tzw. ang. s*poofing*. Staje się to możliwe, jeśli oszust uzyska dostęp do podsieci użytkownika zdalnego. Możliwość taka wymusza obustronne uwierzytelnianie połączenia.

Podstawowym zabezpieczeniem lokalnego systemu komunikacyjnego pracującego w aplikacjach przemysłowych jest jego separacja względem intersieci (zob. 7.2.3). Separację

taką najlepiej wykonać na bazie *Firewalla++*. W przypadku, gdy abonenci sieci systemowej wymagają pracy jako klienci zewnętrznych serwerów usług, wówczas dobre rozwiązanie stanowi uruchamianie procesów klienckich na specjalnie wydzielonym komputerze tzw. komputerze-bastionie [14, 15, 16] (rys. 60). Również i taką funkcje może przejąć opisywany wcześniej *Firewall++*.



Rys. 60 Separacja sieci z pośrednictwem usług

Uwierzytelnianie w systemach przemysłowych powinno polegać na określeniu czy użytkownik lub abonent starający się o dostęp do zasobów innego abonenta jest w istocie tym za kogo się podaje. Użytkownika można identyfikować bazując na specyficznej informacji, którą zna tylko on. Czyli na znanym powszechnie działaniu opartym o nazwę użytkownika i hasło. Możliwe jest również stosowanie uwierzytelniania przez przesyłanie specjalnych danych zwanych certyfikatami (np. X.509) [110, 140] przyznawanymi przez osoby trzecie[9]. Abonent nie będący człowiekiem może uwierzytelniać swój dostęp w taki sam sposób, upraszczając przypadek z hasłem do przesyłania tylko swojego kodu dostępowego. Uwierzytelnianie serwera musi się odbywać przez przesłanie certyfikatu z wykorzystaniem organizacji pośredniczących.

Na podstawie określenia, z kim lub z czym ma się do czynienia możliwe jest przydzielenie praw dostępu na wykonanie określonych funkcji w systemie, zmiany parametrów lub pobrania danych. Stanowi to proces autoryzacji użytkownika. Proces taki staje się niezbędny tylko wówczas, gdy użytkownicy dołączani z poziomu intersieci są podzieleni na różne grupy z różnymi uprawnieniami. Procesy uwierzytelniania i autoryzacji mają miejsce również w lokalnych systemach przemysłowych. Najczęściej na stacjach typu SCADA.

[9] specjalne organizacje

Szyfrowanie przesyłanej informacji jest potrzebne, gdy wymaga się pewności, iż monitorowane dane procesowe lub przekazywane parametry nie będą narażone na podgląd osób niepowołanych. TCP/IP zawiera specjalne protokoły realizujące transmisję szyfrowaną. Są to protokół SSL oraz IPSec. Różnią się one warstwami interfejsu, na których operują. Niezależnie od tego, oba stanowią silną ochronę dla danych i doskonały standardowy mechanizm dla realizacji bezpiecznych transmisji.

Tak samo jak dla sieci wykorzystywanych do prowadzenia działalności gospodarki elektronicznej, tak samo w przypadku sieci wykorzystywanych w informatycznych systemach przemysłowych z dostępem do intersieci, niezbędne jest prowadzenie starannego i przemyślanego zarządzania bezpieczeństwem. Propozycją może być czterostopniowy model opisany w [3], dotyczący ogólnej strategii postępowania. Model ten na każdym ze swoich stopni może zostać adaptowany pod wymogi separacji lokalnych systemów przemysłowych.

Dziedzina zastosowań przedstawionych usług intersieciowych odnośnie systemów przemysłowych sprowadza się do zagadnień prezentacji i rejestracji informacji (wizualizacja, monitorowanie, raportowanie). Zaprezentowana w niniejszym rozdziale możliwość wykorzystania tych usług w środowisku intersieciowym potwierdza słuszność tezy czwartej.

11. Analiza czasowa przepływu informacji na poziomie sieci komputerowych

Aby obiektywnie określić dziedzinę zastosowań protokołu TCP/IP w informatycznych systemach przemysłowych należy poddać analizie parametry czasowe proponowanych we wcześniejszych rozdziałach rozwiązań wykorzystujących ten protokół. Analiza została przeprowadzona względem dwóch parametrów: sprawności zestawionego stosu oraz oferowanej przez niego przepustowości. W celu odniesienia proponowanych rozwiązań do rozwiązań specjalizowanych wykonano porównanie wyników analizy zarówno dla sprawności jak i przepustowości użytecznej. Biorąc pod uwagę przedstawione w rozdziałach 6, 7 oraz 8 aspekty, w niniejszym rozdziale przedstawiono, jaki wpływ ma wykorzystanie TCP/IP oraz sieci Ethernet na parametry czasowe transmisji informacji.

11.1. Analiza wpływu warstw interfejsu na przepustowość i sprawność transmisji

Aby uniezależnić pojęcia sprawności i przepustowości [61] od konkretnych rozwiązań i modeli proponuje się stosowanie następujących pojęć związanych ze sprawnością i przepustowością użyteczną danej wymiany:

$$\eta = \frac{T_U}{T_T}, \tag{31}$$

$$P = \frac{L_U}{T_T}\ [\text{b/s}], \tag{32}$$

gdzie:

- η współczynnik sprawności użytecznej,
- P przepustowość użyteczną wyrażoną w bitach na sekundę,
- T_U czas transmisji danych użytkowych w pojedynczej transakcji danego typu,
- L_U liczba bitów danych użytkowych w pojedynczej transakcji danego typu,
- T_T całkowity czas pojedynczej transakcji danego typu uwzględniający bity serwisowe dołożone przez warstwy protokołu, a nie stanowiące informacji użytecznych.

Dla transmisji idealnej, gdy $T_U = T_T$, współczynnik sprawności wynosi 1. Zależności *31* i *32* odnoszą się do danego typu transakcji, a wartości wyliczone z ich pomocą będą różne dla różnych rodzajów transakcji. Przez transakcję rozumie się całość wymian wchodzących w skład danej sekwencji transmisji ramek pomiędzy nadawcą i odbiorcą, mającą na celu przeniesienie zmiennej sieciowej, a określonej przez definicję protokołu. Transakcja wykorzystuje wybrane warstwy interfejsów i reguły protokołów tych warstw. Wynika z tego, że o ile liczba informacji serwisowej nie zależy liniowo od liczby informacji użytecznej, to

zarówno sprawność jak i przepustowość zależą do liczby przenoszonej informacji użytecznej w ramach danej transakcji. Chcąc uzyskać informację o sprawności i przepustowości całej sieci należy określić cykl pracy sieci uwzględniając wszystkie wykorzystywane rodzaje transakcji oraz wszystkie zmienne sieciowe (zbiór Z_S). Tak utworzony scenariusz wymian można potraktować jako jedną transakcję i obliczyć parametry na bazie przedstawionych wcześniej zależności (*31*, *32*). Wyliczenia takie są jednak mało precyzyjne, gdyż trudno określić cykl sieci dla wszystkich rodzajów transakcji, szczególnie aperiodycznych. Znacznie lepiej posługiwać się pojęciami sprawności i przepustowości w odniesieniu do danej transakcji z poziomu konkretnej warstwy protokołu. Zakładając, że każdy rodzaj transakcji realizowany jest w pewnej warstwie interfejsu komunikacyjnego przez osobny protokół należy analizować protokoły osobno, a nie jako cały stos. Dotyczy to protokołów każdego typu, zarówno stosu TCP/IP jak i protokołów deterministycznych. Dla przykładu w protokole WorldFip [114], aplikacyjny protokół usługi MPS da inne wyniki niż protokół MMS, tak jak w protokole TCP/IP, protokół UDP będzie miał inne parametry wydajnościowe niż FTP.

Dla użytkownika sieci najważniejszy parametr stanowi czas realizacji cyklu wymiany wszystkich informacji w sieci. Parametr ten stanowi o efektywności protokołu. Czas pojedynczej wymiany w oderwaniu od cyklu sieci określa pracę sieci w sposób bardziej elementarny, lecz z punktu widzenia skuteczności wymiany informacji, stanowi sprawę wtórną. W ramach danej transakcji, każda dodatkowa warstwa lub każdy dodatkowy protokół pracujący w stosie interfejsu dokłada informacje serwisowe pogarszające sprawność i przepustowość użyteczną transmisji w obrębie sieci systemowej, gdyż przy stałym T_U rośnie T_T. Zasada ta nie koniecznie musi mieć zastosowanie w środowisku intersieciowym. Informacje serwisowe przekazywane pomiędzy niektórymi warstwami protokołu mogą te parametry polepszać. Dla przykładu protokoły wyboru tras (zob. str. 29) potrafią optymalizować drogę pakietu skracając tym samym czas jego transmisji T_T, a protokoły serwisowe takie jak ICMP potrafią reagować na przeciążenia segmentu unikając tym samym zapaści ($T_T \rightarrow \infty$).

Określenie w kontekście uniwersalnym wpływu warstw interfejsu na parametry czasowe transmisji jest trudne i należy je rozpatrywać indywidualnie dla każdego przypadku zestawionego kanału transakcyjnego. Bardziej interesujące jest porównanie wybranych kanałów dla danego stosu lub dla różnych stosów. Porównanie takie jest przedstawione w następnych podrozdziałach.

11.2. Analiza porównawcza

W kontekście tematu pracy interesujące staje się porównanie sprawności i przepustowości użytecznej transakcji tego samego typu opartych o protokoły deterministyczne oraz wybrane protokoły w stosie TCP/IP. Rozważanie sprawności i przepustowości użytecznej bez analizy porównawczej względem rozwiązań dedykowanych nie dałoby pełnego obrazu korzyści i strat, jakie dają rozwiązania oparte na TCP/IP i Ethernecie.

Określono współczynnik sprawności i przepustowość dla wybranych protokołów deterministycznych oraz protokołów transportowych TCP i UDP. Wykonano analizę dla protokołów Modbus, WorldFip oraz N10. Każdy z tych protokołów jest reprezentatywny dla podstawowych modeli deterministycznego zarządzania wymianami, a mianowicie odpowiednio *Master-Slave*, *PDC* oraz *Token passing*.

Dla protokołów deterministycznych, wykonane obliczenia są dokładne, gdyż nie istnieje potrzeba szacowania opóźnień wynikających z obciążenia lub utraty pakietów. W przypadku protokołów TCP/IP uwzględniono pracę na odizolowanym segmencie sieci Ethernet i szacowano jego obciążenie. Dla wszystkich przypadków przyjęto standardowe lub typowe prędkości transmisji dla danej sieci oraz wspólną minimalną i maksymalną liczbę danych w wielokrotnościach pojedynczego bajtu. Do analizy przyjęto:

- obliczenia dla prędkości transmisji standardowej względem danej sieci,
- obliczenia dla hipotetycznej prędkości transmisji 100 Mb/s,
- minimalny rozmiar danych użytecznych równy 2 B,
- maksymalny rozmiar danych użytecznych równy 126 B,

Zakres rozmiaru danych użytecznych ustalono na podstawie wyznaczenia części wspólnej z zakresów rozmiarów oferowanych przez analizowane protokoły. Rozmiar 126 bajtów danych nie jest dużym pakietem z punktu widzenia protokołu IP. W systemach komunikacyjnych pracujących na najniższym poziomie wymiany informacji (rozdz. 5) nie występuje konieczność przesyłu dużych pakietów danych. Są to albo pojedyncze zmienne aplikacyjne albo, co najwyżej grupy kilkunastu zmiennych aplikacyjnych, z czego każda ma rozmiar od jednego bita do kilku bajtów. Dla wszystkich protokołów przyjęto analizę pojedynczej transakcji przekazania danych pomiędzy abonentami nie realizującymi strategicznych funkcji związanych z działaniem protokołu. Wszystkie pozostałe parametry charakterystyczne dla danego stosu protokołów przyjęto jako typowe tak, aby sieć pracowała w warunkach najbardziej zbliżonych do standardowych. W obliczeniach nie uwzględniono parametrów związanych z czasami detekcji ramki, przygotowania odpowiedzi i innych związanych bezpośrednio ze sprzętem. Nie uwzględniano żadnych mechanizmów optymalizacji wymian [64, 55].

Przyjęto założenie do analizy, iż:

$$T_O = T_{DR} + T_{AR} + T_P + T_{ONO} = 0 \qquad (33)$$

gdzie:

T_O – czas opóźnień w warstwach nierozpatrywanych,

T_{DR} – czas detekcji ramki przez odbiorców,

T_{AR} – czas analizy ramki przez odbiorców,

T_P – czas przygotowania odpowiedzi przez nadawców,

T_{ONO} – czas oczekiwania na odpowiedź,

Wynika to z faktu, iż tematem analizy jest protokół a nie cała sieć łącznie z konstrukcją warstw sprzętowych. Szybki rozwój technologiczny urządzeń powoduje, iż parametry

opóźnień związanych z detekcją ramek, ich analizą itp. stale podlegają polepszeniu. Czasy reakcji abonentów nie stanowią części składowej definicji protokołu, a ich uwzględnianie musiałoby pociągnąć za sobą zawężenie rozważań do konkretnych rozwiązań sprzętowych. Przyjęcie stałego niezerowego czasu opóźnień (wzór 34) dla wszystkich protokołów stanowiłoby również rozwiązanie wystarczające.

$$T_O = const \tag{34}$$

Otrzymane wyniki będą się wówczas różnić od rzeczywistości, lecz dla potrzeb porównania byłoby to skuteczne. Parametr T_O wywiera bardzo istotny wpływ na sprawność i przepustowość użyteczną. Każda operacja wymiany danych pomiędzy abonentami składająca się na całość transakcji, powoduje interakcję ze sprzętem, a co za tym idzie opóźnienia. Im więcej wymian wchodzi w skład całej transakcji tym większy wpływ na efektywność wywierają niezerowe opóźnienia wnoszone przez interfejsy abonentów. Poprawną wartością T_O jest maksymalny czas opóźnień, tzw. *time-out*. Stanowi on maksymalny czas reakcji sieci na wysłaną ramkę. Można wówczas uzyskać analizę przypadku pesymistycznego, która jest niezbędna dla określania możliwości protokołów przemysłowych. Dla porównań, w których biorą udział protokoły z dostępem rywalizacyjnym, określenie *time-out*-u jest umowne, gdyż z samego protokołu wynika brak jego granicznej wartości. *Time-out* jaki pojawia się w protokołach deterministycznych wynika z przetwarzania danych w warstwach interfejsów komunikacyjnych abonentów. Przekroczenie tego czasu oznacza awarię abonenta lub medium. W protokołach intersieciowych dodatkowym składnikiem *time-out*-u jest opóźnienie wynikające z rywalizacji o dostęp.

Sprawność użyteczna jest miarą jakości samego protokołu. Natomiast przepustowość użyteczna określa jego ilościowe możliwości transmisji. Zarówno jedna jak i druga wielkość będzie liczona dla abonentów idealnych, gdzie $T_O = 0$, czyli z wyłączeniem parametrów czasowych wnoszonych przez interfejsy komunikacyjne abonentów korzystających z danego protokołu. Otrzymano charakterystyki działania samych protokołów na poziomie algorytmu działającego na medium.

Dla porównania transmisyjnej skuteczności protokołów należy brać pod uwagę ich sprawność użyteczną natomiast przepustowość ma większe zastosowanie przy doborze rozwiązania dla konkretnego systemu. Wykonując rzeczywisty dobór rozwiązań komunikacyjnych [35, 57] należy brać pod uwagę rzeczywiste wartości czasu T_O dla wykorzystywanego sprzętu.

11.2.1. Protokół sieci Modbus

Bazując na zależnościach *31* i *32* oraz na analizie transakcji cyklicznej protokołu Modbus RTU [61], przyjęto współczynnik sprawności sieci jako:

$$\eta = \frac{T_U}{T_T} = \frac{\frac{8n}{V}}{T_{TŻ} + T_{TO}} = \frac{8n}{V\left(T_{TŻ} + T_{TO}\right)}, \tag{35}$$

oraz przepustowość jako:

$$P = \frac{L_U}{T_T} = \frac{8n}{T_{TŻ} + T_{TO}} \text{ [b/s]}, \tag{36}$$

gdzie:

n – określa liczbę bajtów danych użytkowych,

V – prędkość transmisji w [b/s],

$T_{TŻ}$ – czas transmisji żądania,

T_{TO} – czas transmisji odpowiedzi,

Wzory te dotyczą wymiany danych pomiędzy stacjami Master i Slave typu zapytanie – odpowiedź, zatem aby zrealizować transakcję wymiany danych pomiędzy dwoma dowolnymi abonentami, musi zajść transakcja typu odczyt ze stacji Slave – zapis do stacji Slave. Wymusza to branie pod uwagę następującej sekwencji transakcji:

- żądanie odczytu ze stacji Master do stacji Slave,
- odczyt ze stacji Slave do stacji Master,
- zapis ze stacji Master do stacji Slave,
- potwierdzenie ze stacji Slave do stacji Master.

Czyli

$$\eta = \frac{8n}{V\left(T_{TŻ} + T_{TO} + T_{TZ} + T_{TP}\right)}, \tag{37}$$

oraz przepustowość jako:

$$P = \frac{8n}{T_{TŻ} + T_{TO} + T_{TZ} + T_{TP}} \text{ [b/s]}, \tag{38}$$

gdzie:

T_{TO} – czas transmisji żądania,

T_{TO} – czas transmisji odpowiedzi odczytu,

T_{TZ} – czas transmisji odpowiedzi zapisu,

T_{TP} – czas transmisji potwierdzenia zapisu,

W protokole Modbus, czas transmisji pakietu T_P jest równy:

$$T_P = \frac{8L_{ZT} + 11}{V} \text{ [s]}, \tag{39}$$

gdzie L_{ZT} jest liczbą przesyłanych znaków w ramce danego typu wymiany.

Przeprowadzono analizę dla transakcji wymiany zmiennej pomiędzy stacją Master a stacją Slave dla transmisji ośmiobitowego znaku bez bitu parzystości i bez bitu stopu.

Zakładając następujące wartości parametrów pracy sieci:

- liczba znaków ramki dla wymiany typu:
  - żądanie odczytu ze stacji Master do stacji Slave
    $L_{ZT} = 8$,
  - odczyt n słów ze stacji Slave do stacji Master,
    $L_{ZT} = 5+2n$,

– zapis *n* słów ze stacji Master do stacji Slave,
  $L_{ZT} = 9+2n$,
– potwierdzenie ze stacji Slave do stacji Master.
  $L_{ZT} = 8$,

- prędkość transmisji równa jest 19200 b/s.

Na podstawie wielkości ramek używanych w poszczególnych wymianach, otrzymano czasy transakcji dla przyjętych rozmiarów danych użytecznych.

$$T_T = \frac{75}{V} + \frac{16n+51}{V} + \frac{16n+83}{V} + \frac{75}{V} = \frac{32n+284}{V} \text{ [s]}, \tag{40}$$

gdzie T_T jest czasem trwania całej transakcji.

■

Sprawność i przepustowość użyteczna dla tak poczynionych założeń będzie wyrażała się następującymi zależnościami:

$$\eta = \frac{8n}{284+32n} = \frac{4n}{71+8n}, \tag{41}$$

$$P = \frac{8nV}{284+32n} = \frac{4nV}{71+8n} \text{ [b/s]}. \tag{42}$$

Ze wzorów 41 i 42 widać, iż wartość przepustowości użytecznej transakcji zależy wprost proporcjonalnie od zastosowanej prędkości transmisji. Wzrost liczby bitów serwisowych w mianowniku ma wpływ odwrotnie proporcjonalny na sprawność transakcji. Im bardziej skomplikowana transakcja tym mniejsza będzie jej sprawność użyteczna.

■

11.2.2. Protokół sieci N10

Bazując na zależnościach *31* i *32* oraz na analizie transakcji cyklicznej protokołu sieci N10 [61], przyjęto współczynnik sprawności sieci jako:

$$\eta = \frac{T_U}{T_T} = \frac{\frac{8n}{V}}{3T_{TŻ} + T_{TD} + T_{TP}} = \frac{8n}{V\left(3T_{TŻ} + T_{TD} + T_{TP}\right)}, \tag{43}$$

oraz przepustowość:

$$P = \frac{L_U}{T_T} = \frac{8n}{3T_{TŻ} + T_{TD} + T_{TP}} \text{ [b/s]}, \tag{44}$$

gdzie:

n – określa liczbę bajtów danych użytkowych,
V – prędkość transmisji w [b/s],
$T_{TŻ}$ – czas transmisji żetonu,
T_{TD} – czas transmisji danych,
T_{TP} – czas transmisji potwierdzenia.

Zakładając, że czas transmisji jest wyrażony zależnością:

$$T_T = \frac{8L_{ZT} + 11}{V} \text{ [s]}, \tag{45}$$

oraz

- liczba znaków ramki dla wymiany typu:
  - transmisji żetonu
    $L_{ZT} = 4$,
  - transmisji danych,
    $L_{ZT} = n$,
  - transmisji potwierdzenia,
    $L_{ZT} = 4$,
- prędkość transmisji równa jest $V = 19200$ [b/s].

Na podstawie wielkości ramek używanych w transakcji cyklicznej, otrzymano zależność czasu transakcji od przyjętych rozmiarów danych użytecznych.

$$T_T = \frac{43}{V} + \frac{8n + 11}{V} + \frac{43}{V} = \frac{8n + 97}{V} \text{ [s]}, \tag{46}$$

gdzie

T_T jest czasem trwania całej transakcji,

n oznacza, liczbę bajtów informacji użytecznej.

■

Sprawność i przepustowość użyteczna dla rozważanego przypadku wynosi:

$$\eta = \frac{8n}{183 + 8n}, \tag{47}$$

$$P = \frac{8nV}{183 + 8n} \text{ [b/s]}. \tag{48}$$

Podobnie jak dla wzorów 41 i 42 zależności pomiędzy prędkością transmisji a przepustowością użyteczną oraz liczbą bitów serwisowych a sprawnością są dla wzorów 47 i 48 takie same. Jednocześnie widać, porównując wzory 41 i 47, iż sprawność transakcji cyklicznej protokołu N10 jest lepsza od omawianej wcześniej transakcji cyklicznej protokołu Modbus. Obliczenia i pomiary wykonane dla dużego systemu [131] są zgodne z otrzymanymi zależnościami.

■

11.2.3. Protokół sieci WorldFip

Bazując na zależnościach *31* i *32* oraz na analizie transakcji cyklicznej sieci FIP/WorldFip [61, 58], zdefiniowano współczynnik sprawności sieci jako stosunek czasu transmisji danych użytkownika do całkowitego czasu transakcji.

Na podstawie wielkości ramek używanych w transakcji cyklicznej [116, 114], założono, że:

- czas transmisji danych użytecznych (T_U) wynosi $\frac{8n}{V}$ [s],

- czas transmisji ramki żądania (T_Z) wynosi $\frac{61}{V}$ [s] i jest stały,
- czas transmisji ramki odpowiedzi (T_O) wynosi $\frac{61+8n}{V}$ [s].

Otrzymano określenie czasu transakcji:

$$T_T = T_Z + T_O \text{ [s]}, \tag{49}$$

$$T_T = \frac{61}{V} + \frac{61+8n}{V} = \frac{2(4n+61)}{V} \text{ [s]}, \tag{50}$$

gdzie

T_T jest czasem trwania całej transakcji,

n oznacza, liczbę bitów informacji użytecznej.

■

Po podstawieniach otrzymano:

$$\eta = \frac{T_U}{T_T} = \frac{\frac{8n}{V}}{\frac{2(4n+61)}{V}} = \frac{4n}{4n+61}, \tag{51}$$

Przyjęto również zależność na przepustowość użyteczną, jako stosunek liczby bitów użytecznych transmitowanych w danej transakcji do całkowitego czasu transakcji, wyrażoną w bitach na sekundę. Po przekształceniach otrzymano:

$$P = \frac{L_U}{T_T} = \frac{8n}{\frac{2(4n+61)}{V}} = \frac{4nV}{4n+61} \text{ [b/s]}. \tag{52}$$

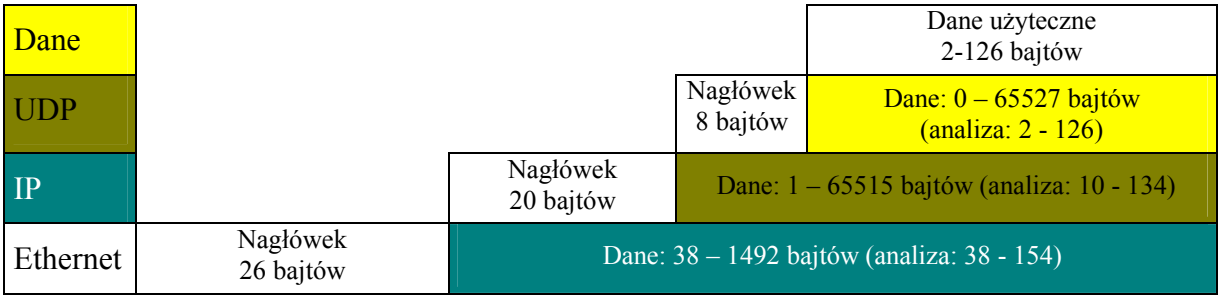
Podobnie jak dla wzorów 41 i 42 oraz 47 i 48 zależności pomiędzy prędkością transmisji a przepustowością użyteczną oraz liczbą bitów serwisowych a sprawnością są dla wzorów 51 i 52 takie same i będą zawsze takie same dla każdego rodzaju transakcji. Wynika to z samej definicji sprawności i przepustowości użytecznej przedstawionej na stronie 108 (wzory *31* i *32*). Wzrost czasu T_T może nastąpić na skutek wzrostu liczby danych użytecznych lub wzrostu liczby danych serwisowych. W przypadku sprawności, czas transmisji danych użytecznych wpływa jednakowo na licznik i mianownik zatem spadek sprawności będzie wynikał tylko ze złożoności transakcji. Z omawianych trzech transakcji cyklicznych protokołów deterministycznych najlepszą sprawność wykazuje transakcja cykliczna protokołu WorldFIP. Wyniki obliczeń dla zadanego przedziału danych użytecznych są zebrane na wykresie w rozdziale podsumowującym 11.3. Pomiary wykonane dla praktycznej aplikacji systemu kontrolnego [130] są zgodne z otrzymanymi zależnościami.

■

11.2.4. Protokół UDP w sieci Ethernet

W celu wyznaczenia sprawności i przepustowości użytecznej dla stosu protokołów UDP [102], IP, Etehrnet należy określić rozmiar transmitowanego pakietu, prędkość transmisji, rozmiar danych użytecznych i wynikające z nich czasy transmisji.

Datagram użytkownika protokołu UDP składa się z ośmiu bajtów nagłówka oraz pola danych mogącego przenosić od zera do 65527 bajtów danych użytecznych, czyli pochodzących z warstwy aplikacji użytkownika. Kapsułkując datagram UDP w datagramie IP, protokół IP dokłada swój nagłówek w rozmiarze 20 bajtów. Przyjęto, że na polu danych datagramu IP znajduje się tylko datagram UDP. Ponieważ zakres testowanych rozmiarów pakietów wynosi od 2 do 126 bajtów, to datagram UDP zmieści się wraz z nagłówkiem na polu danych datagramu IP, które to pole może przenosić do 64k bajtów danych. Nie zaistnieje zatem konieczność dzielenia datagramu UDP. Założono również, że siecią fizyczną jest Ethernet. Zatem całość zostanie kapsułkowana w ramce Ethernetu. Spowoduje to dołożenie 26 bajtów nagłówkowych od warstw Ethernetu oraz wymusi minimalny rozmiar danych na 38 bajtów i maksymalny na 1492 bajty. Ograniczenie to wynika z minimalnego oraz maksymalnego rozmiaru ramki Ethernetowej dla standardu 10 Mb i 100 Mb, który wynosi odpowiednio 64 oktety i 1518 oktetów [13, 14]. Na poniższym rysunku przedstawiono kapsułkowanie danych.



Rys. 61 Kapsułkowanie datagramu UDP

Przedstawiony nagłówek Ethernetu jest pojęciem abstrakcyjnym, gdyż odnosi się do całości narzutu danych sterujących dokładanych przez warstwy sieci. Zawiera zatem preambułę warstwy fizycznej, pola adresowe, typ, jak również sumę kontrolną CRC, która w rzeczywistości znajduje się na końcu ramki.

Czas transmisji datagramu wynosi:

$$T_T = T_{TD}$$

gdzie:

T_T jest czasem transakcji,

T_{TD} jest czasem transakcji pakietu z danymi,

Czas transmisji pakietu zależy od jego rozmiaru, który jest stały dla $2 \le n \le 10$ oraz zmienny dla $n > 10$. Zatem:

$$T_T = \frac{512}{V}\ [\mathrm{s}] \qquad \text{dla rozmiaru datagramu IP} \le 38 \text{ bajtów, czyli } 2 \le n \le 10, \qquad (53)$$

oraz

$$T_T = \frac{8(26+20+8)+8n}{V}$$

$$T_T = \frac{8(54+n)}{V}\text{[s]} \qquad \text{dla rozmiaru datagramu IP > 38 bajtów, czyli } n>10, \tag{54}$$

gdzie n jest liczbą bajtów danych użytecznych.

Aby rozmiar danych datagramu IP był większy od 38 bajtów warstwa aplikacyjna musi dołożyć przynajmniej 11 bajtów danych. Dane te zostaną umieszczone w polach przeznaczonych dla danych użytecznych (pole danych użytecznych na rysunku 61). Z punktu widzenia warstwy aplikacji systemu kontrolnego (rysunek 30) nie stanowią one danych użytecznych (wg definicji ze strony 12) lecz jedynie wypełnienie niezbędne do osiągnięcia minimalnego rozmiaru pakietu sieci Ethernet.

■

Przy założeniu standardowych prędkości transmisji czasy transmisji są niewielkie. Jednak czas transmisji ramki nie jest wyznacznikiem wydajności protokołu. Przyjmując kontrolę wymian na poziomie sieci Ethernet, czyli wykorzystując mechanizm CSMA/CD do rozwiązywania kolizji, uzyskujemy uproszczenie działania protokołu względem protokołów analizowanych wcześniej. Stacja wykonująca zapis zmiennej do innej stacji, po prostu generuje wymianę. Przy założeniu, że medium nie jest zajęte następuje niezwłoczne bezpołączeniowe przekazanie datagramu do zaadresowanego abonenta. Niestety jest to sytuacja teoretycznie możliwa, lecz praktycznie rzadko obserwowalna. Aby wyniki analizy miały związek z rzeczywistością należy uwzględnić występowanie kolizji.

Bazując na zależnościach *31* i *32* oraz transakcji datagramu UDP można wyznaczyć sprawność i przepustowość dla przypadku idealnego, czyli bez opóźnień.

$$\eta_O = \frac{T_U}{T_T} = \frac{\frac{8n}{V}}{\frac{512}{V}} = \frac{n}{64} \qquad \text{dla } n \geq 2 \text{ i } n \leq 10, \tag{55}$$

$$\eta_O = \frac{T_U}{T_T} = \frac{\frac{8n}{V}}{\frac{8\,(54+n)}{V}} = \frac{n}{n+54} \qquad \text{dla } n > 10, \tag{56}$$

$$P_O = \frac{L_U}{T_T} = \frac{8n}{\frac{512}{V}} = \frac{nV}{64} \qquad \text{dla } n \geq 2 \text{ i } n \leq 10, \tag{57}$$

$$P_O = \frac{L_U}{T_T} = \frac{8n}{\frac{8(54+n))}{V}} = \frac{nV}{n+54} \qquad \text{dla } n > 10. \tag{58}$$

Dla przypadku rzeczywistego należy uwzględnić wartość, która będzie reprezentować opóźnienia wynikające z pracy protokołu. Ponieważ opóźnienia wynikają z wielu czynników

i są silnie zmienne, uwzględnienie precyzyjnych wartości jest niemożliwe. Można natomiast oprzeć się na prawdopodobieństwie ich wystąpienia wynikającego z obciążenia sieci. Stąd przy założeniu liniowej zmiany sprawności i przepustowości w funkcji opóźnień otrzymuje się zależność na wielkości średnie sprawności i przepustowości użytecznej:

$$\eta = \eta_O \omega, \tag{59}$$

$$P = P_O \omega, \tag{60}$$

gdzie ω jest uśrednionym współczynnikiem występujących opóźnień reprezentującym ubytek efektywności przypadku idealnego. Korzystając z teorii kolejek można stwierdzić, iż o ile czas podróży pakietów łącznie z opóźnieniami określony jest daną wariancją, to wariancja ta jest proporcjonalna do wartości ilorazu $\frac{1}{(1-U)}$ [14]. Bazując na tym, a także na wzorach 11, 59, 60 można stwierdzić, iż:

$$T_{TR} = \frac{T_T}{(1-U)}, \tag{61}$$

gdzie T_{TR} określa rzeczywisty czas transmisji.

Stąd:

$$\omega = 1\text{-}U \tag{62}$$

zatem sprawność użyteczna może być szacowana ze wzoru:

$$\eta = \eta_O (1-U), \tag{63}$$

a przepustowość ze wzoru:

$$P = P_O (1-U) \text{[b/s]}. \tag{64}$$

Powyższe zależności umożliwiają szacowanie optymistyczne, gdyż w rzeczywistości efektywność protokołu opartego na CSMA/CD nie maleje liniowo ze wzrostem obciążenia. Zależność 11 jest jednak wystarczająca dla szacowania oczekiwanych opóźnień wielu uogólnionych przypadków rozwiązań sieciowych.

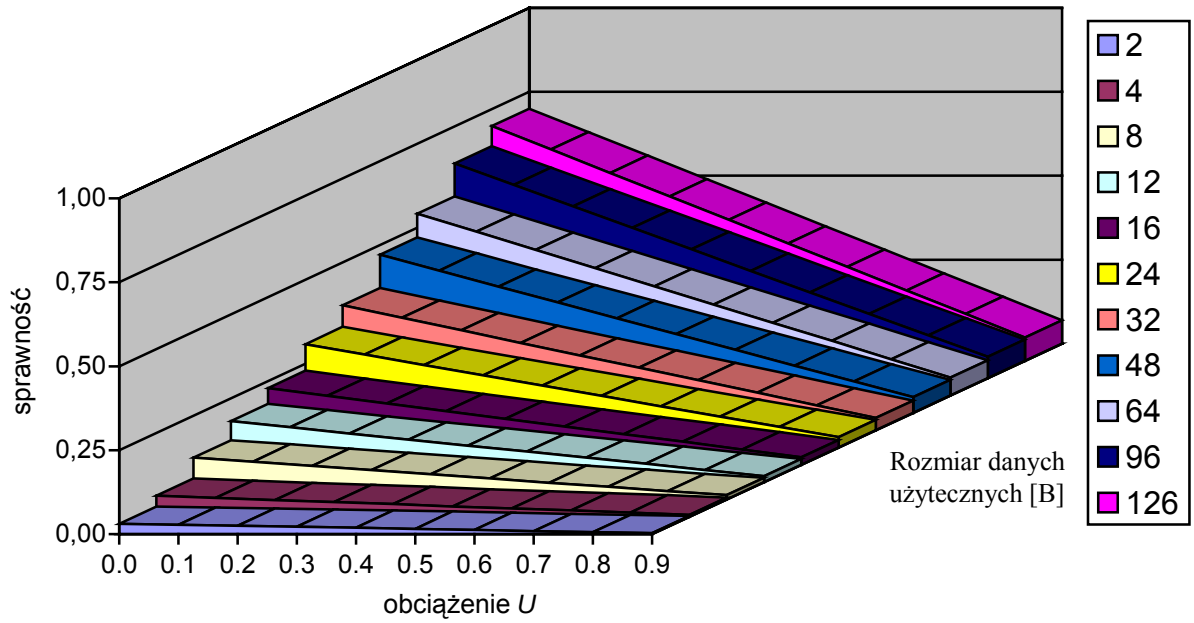
Współczynnik U reprezentuje bieżące wykorzystanie sieci względem danej przepływności sieci [13]. Do obliczeń przyjęto wartości z całego przedziału 0-1 ze skokiem 10%.

■

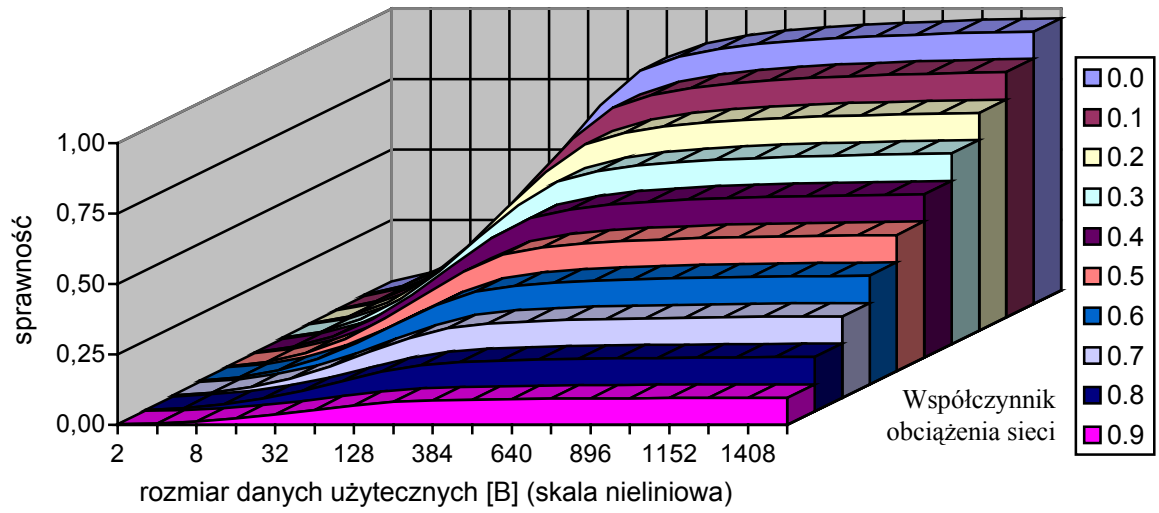
Dodatkowo obliczono wartości sprawności dla większych pakietów danych użytecznych. Wyliczenie takie pozwala na określenie czy wzrost rozmiaru paczki danych użytecznych może polepszyć parametry efektywności protokołu. Określenie optymalnego rozmiaru stanowi jedno z kryteriów dla wyznaczenia dziedziny zastosowań protokołu.

Jako maksymalny rozmiar pakietu przyjęto maksymalny rozmiar danych użytecznych przenoszonych przez ramkę Ethernetu, czyli 1492 bajty [14]. Powyżej tego rozmiaru wcześniejsze zależności tracą znaczenie, ze względu na konieczność transmisji danych przy użyciu więcej niż jednej transakcji podstawowej protokołu UDP.

Poniższe wykresy zawierają prezentację wyników obliczeń sprawności użytecznej w zależności od obciążenia sieci oraz rozmiaru transmitowanej paczki danych użytecznych.



Rys. 62 Wykres zmiany sprawności użytecznej protokołu UDP w funkcji wzrostu obciążenia i rozmiaru danych użytecznych



Rys. 63 Wykres zmiany sprawności protokołu UDP w funkcji wzrostu rozmiaru pakietu i obciążenia sieci

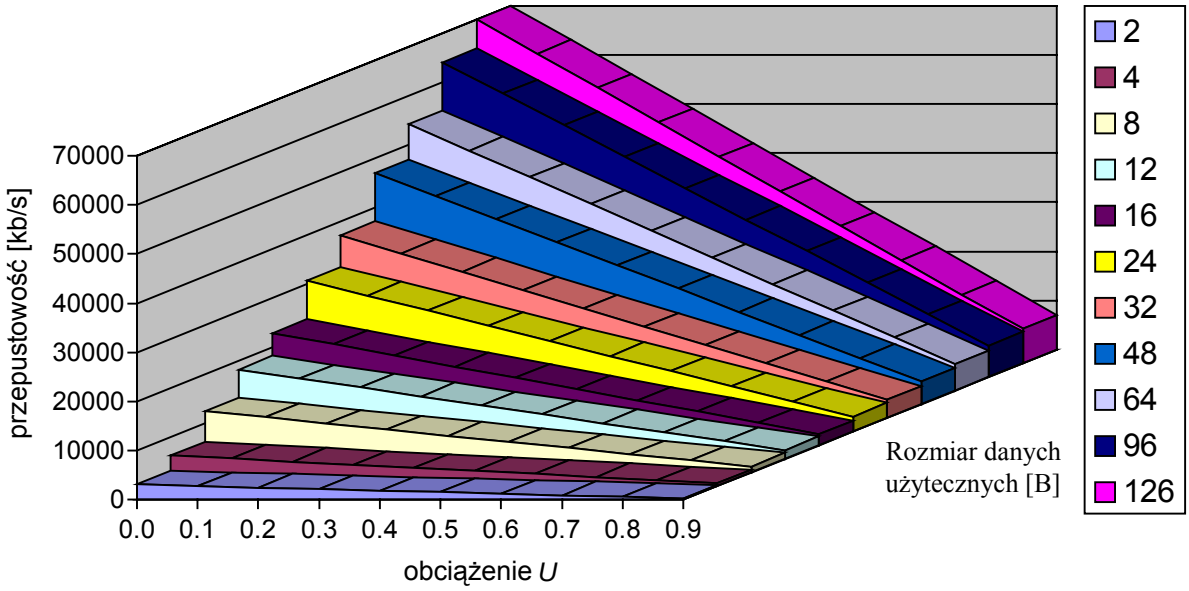
Z powyższego rysunku widać, iż efektywność protokołu UDP wzrasta ze wzrostem rozmiaru przesyłanej paczki danych użytecznych. Wzrost ten nie jest liniowy. Im mniej obciążona sieć tym występuje większy przyrost sprawności w funkcji rozmiaru paczki. Intensywny wzrost sprawności ma miejsce jednak tylko do rozmiaru pakietu około pół kilobajta. Dalsze zwiększanie rozmiaru nie poprawia znacząco sprawności. Wynika stąd, iż bardziej efektywne wykorzystanie protokołu będzie miało miejsce w przypadku stosowania UDP do transmisji większych paczek danych niż przyjętych do testów a wynikających z protokołów przemysłowych.

Protokół jest wydajny tylko dla niewielkich obciążeń sieci. Spadek efektywności pracy protokołu, pomimo optymistycznego szacowania zależnością liniową, jest i tak dość drastyczny. Jako wartości charakterystyczne dla dalszych porównań przyjęto:

$U = 0$ – praca idealna bez opóźnień,

$U = 0{,}3$ – praca typowa,

$U = 0{,}9$ – praca silnie dociążona ruchem pakietów – duże opóźnienia.



Rys. 64 Wykres zmiany przepustowości użytecznej protokołu UDP w funkcji wzrostu obciążenia i rozmiaru danych użytecznych

Na powyższym rysunku przestawiono przepustowość protokołu dla testowego przedziału rozmiaru danych użytecznych oraz dla pełnego zakresu obciążeń przy prędkości 100 [Mb/s]. Ponieważ przepustowość jest proporcjonalnie zależna od sprawności wykres obrazuje te same zjawiska co obserwowane wcześniej wykresy sprawności użytecznej.

11.2.5. Protokół TCP w sieci Ethernet

Podobnie jak poprzednio, aby wyznaczyć sprawność i przepustowość użyteczną należy określić rozmiary segmentu TCP i wynikające z nich czasy transmisji. Rozmiar segmentu w protokole TCP może być negocjowany pomiędzy punktami końcowymi utworzonego połączenia wirtualnego. Wynika to z konieczności ustalenia minimalnego rozmiaru segmentu obsługiwanego przez abonentów biorących udział w połączeniu. Do analizy przyjęto, iż maksymalny rozmiar segmentu będzie wynosił 126 bajtów, co stanowi maksymalny rozmiar danych użytecznych przyjęty do analizy. Rozmiar ten jest znacznie mniejszy niż umożliwia negocjacja parametru MTU na sieci Ethernet, więc takie założenie nie spowoduje kłopotów z koniecznością dzielenia segmentów przez IP.

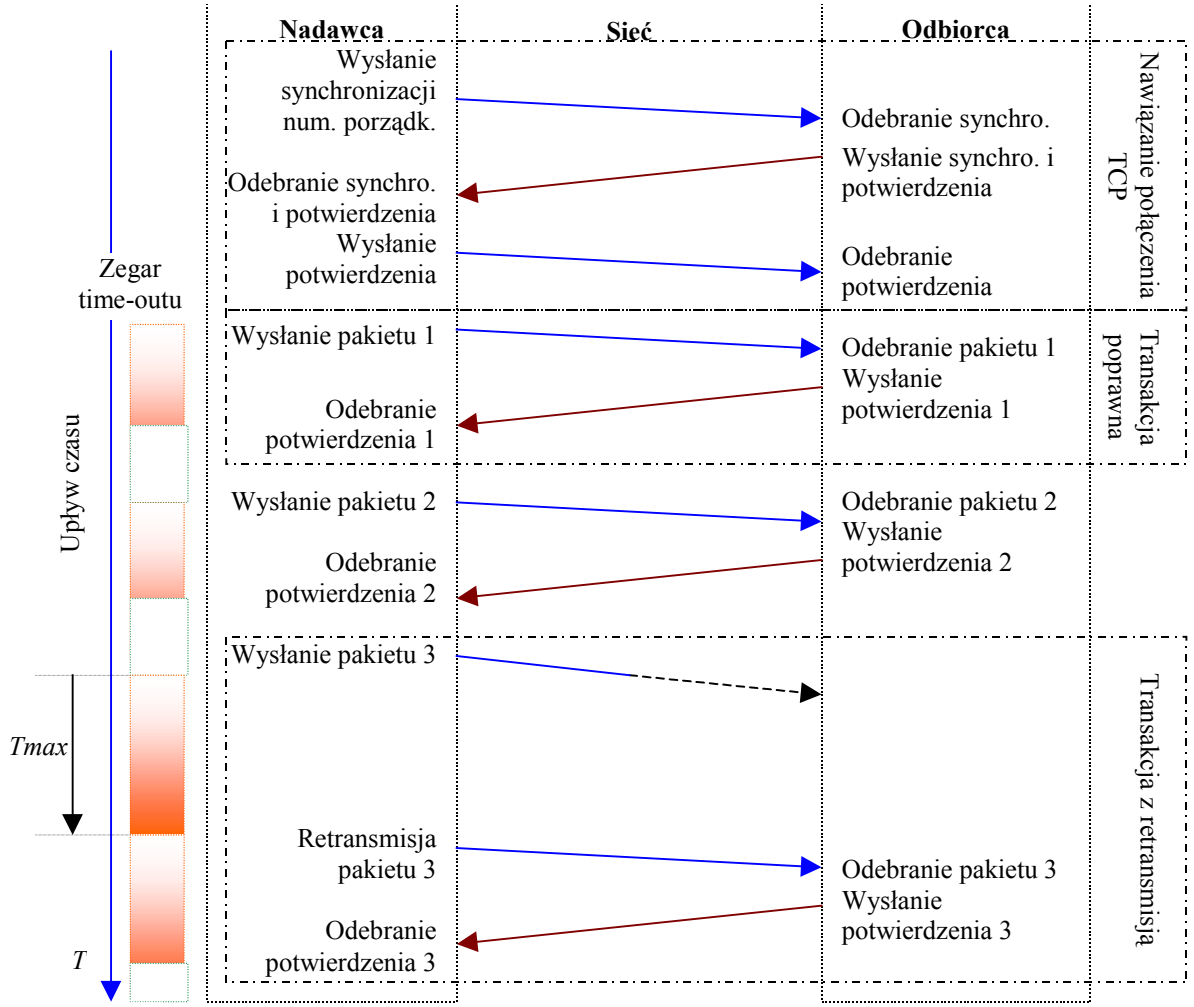
Wszystkie obliczenia poczyniono przy założeniu minimalnych rozmiarów nagłówków TCP oraz IP czyli braku pola opcji i uzupełnień [14]. Zakłada się zatem, że oprogramowanie protokołu nie wykonuje żadnych negocjacji w czasie wykorzystywania sieci systemowej.

<table>
<tr><td>Dane</td><td colspan="3"></td><td>Dane użyteczne
2-126 bajtów</td></tr>
<tr><td>TCP</td><td colspan="2"></td><td>Nagłówek
20 bajtów</td><td>Dane: 1 – 536 bajtów
(analiza: 2 - 126)</td></tr>
<tr><td>IP</td><td></td><td>Nagłówek
20 bajtów</td><td colspan="2">Dane: 1 – 65515 bajtów (analiza: 22 - 146)</td></tr>
<tr><td>Ethernet</td><td>Nagłówek
26 bajtów</td><td colspan="3">Dane: 38 – 1492 bajtów (analiza: 42 - 166)</td></tr>
</table>

Rys. 65 Kapsułkowanie pakietu TCP

Z powyższego rysunku wynika, że tak jak dla datagramu UDP, nie ma potrzeby dzielenia strumienia TCP w celu jego transmisji przez IP na sieci Ethernet. Zatem rozmiar ramki Ethernetu będzie wynosił od 68 do 192 bajtów.

Protokół TCP jest niezawodnym protokołem połączeniowym, w którym wykorzystuje się w nim mechanizm nawiązywania połączenia, potwierdzeń i powtórzeń.



Rys. 66 Transmisja strumieni TCP

Na rysunku 66 przedstawiono schemat nawiązywania połączenia i przesyłania pakietów w standardowej komunikacji TCP.

Przedstawiona transmisja z potwierdzeniem jest wysoce nieefektywna. W praktyce często stosuje się transmisję z tzw. przesuwającym się oknem. Polega ona na transmitowaniu pakietów z zakresu od danego numeru p do $p+q$ bez oczekiwania na potwierdzenie. Przesunięcie okna na pozycję $p+q+1$ jest możliwe tylko wówczas, gdy przyjdzie potwierdzenie dla pakietu p. W przypadku poprawnego działania sieci metoda ta daje większą przepustowość transmisji niż algorytm standardowy. Niezależnie od tego mechanizmu, pojedyncza transakcja wymiany zmiennej polega na przesłaniu komunikatu i potwierdzenia. W przypadku, gdy po każdorazowej transmisji następuje rozłączenie połączenia, do każdej następnej transakcji należy dołożyć czas jego nawiązywania.

W celu analizy sprowadzono całość transakcji TCP do problemu przesłania paczki danych użytecznych, a pakiety serwisowe potraktowano jako dane pochodzące od narzutu protokołu. Pomijając czasy opóźnień w interfejsach oraz nie uwzględniając opóźnień wynikających z kolizji otrzymano:

$$T_T = T_N + T_P$$

gdzie:

T_T jest czasem trwania transakcji z nawiązaniem połączenia,

T_N jest czasem trwania transakcji nawiązywania połączenia,

T_P jest czasem transakcji pakietu.

Na te czasy składają się:

$$T_N = T_{SYN} + T_{SYNACK} + T_{ACK}$$

$$T_P = T_{TD} + T_{ACK}$$

gdzie:

T_{SYN} jest czasem trwania wymiany segmentu synchronizacyjnego,

T_{SYNACK} jest czasem trwania wymiany segmentu potwierdzenia segmentu synchronizacyjnego,

T_{ACK} jest czasem trwania wymiany segmentu potwierdzenia,

T_{TD} jest czasem trwania wymiany segmentu z danymi.

Ponieważ różnica pomiędzy T_{SYN}, T_{ACK} oraz T_{SYNACK} polega jedynie na ustawieniu innych bitów w polu bitów kodu ramki można przyjąć do analizy, że czas ich transmisji jest taki sam. Ramki te nie transmitują danych użytecznych.

Można zatem określić czas trwania całej transakcji.

$$T_T = T_{TD} + 4T_{ACK}$$

Powyższa zależność jest mocno uproszczona względem rzeczywistości, gdyż nie uwzględnia czasu retransmisji, renegocjacji rozmiaru strumienia, czasów opóźnień oraz czasów transmisji danych przesyłanych poza głównym strumieniem. Jednak dla analizy pojedynczej transakcji w warunkach optymistycznych w zupełności wystarczy.

Uwzględniając rozmiary ramek i oraz fakt, że nawet dla minimalnego rozmiaru danych użytecznych zachodzi transmisja ramki Ethernetu o rozmiarze większym od minimalnego, otrzymano:

$$T_T = \frac{8(26+20+20+n)}{V} + \frac{8(4(26+38))}{V} = \frac{8(66+n+256)}{V} = \frac{8(322+n)}{V} \text{ [s]}, \quad (65)$$

gdzie n jest liczbą bajtów danych użytecznych.

■

Tak jak dla przypadku UDP, czasy transmisji są również niewielkie. Jednak na przetransmitowanie zmiennej, protokół TCP potrzebuje ponad trzy razy więcej czasu niż UDP. Wynika to z zapewnienia niezawodności.

Podobnie jak poprzednio, aby wyniki analizy miały związek z rzeczywistością należy uwzględnić występowanie kolizji. Kolizja może wystąpić przy każdej wymianie, dlatego im więcej wymian zawiera transakcja tym więcej kolizji może wystąpić. Zatem czas opóźnień z nich wynikający jest zależny wprost proporcjonalnie od liczby wymian w danej transakcji.

Podobnie jak poprzednio, na podstawie zależności 31, 32 oraz 65 można wyznaczyć użyteczną sprawność i przepustowość dla przypadku skrajnie optymistycznego.

$$\eta_O = \frac{T_U}{T_T} = \frac{\frac{8n}{V}}{\frac{8(322+n)}{V}} = \frac{n}{n+322}, \quad (66)$$

$$P_O = \frac{L_U}{T_T} = \frac{8n}{\frac{8(322+n)}{V}} = \frac{nV}{n+322} \text{ [b/s]}. \quad (67)$$

Podobnie jak dla protokołu UDP, dla przypadku rzeczywistego należy uwzględnić wielkość, która będzie reprezentować opóźnienia wynikające z pracy protokołu. Zatem analogicznie, bazując na wzorze 11 sprawność użyteczna może być szacowana ze wzoru:

$$\eta = \eta_O(1-U), \quad (68)$$

a przepustowość ze wzoru:

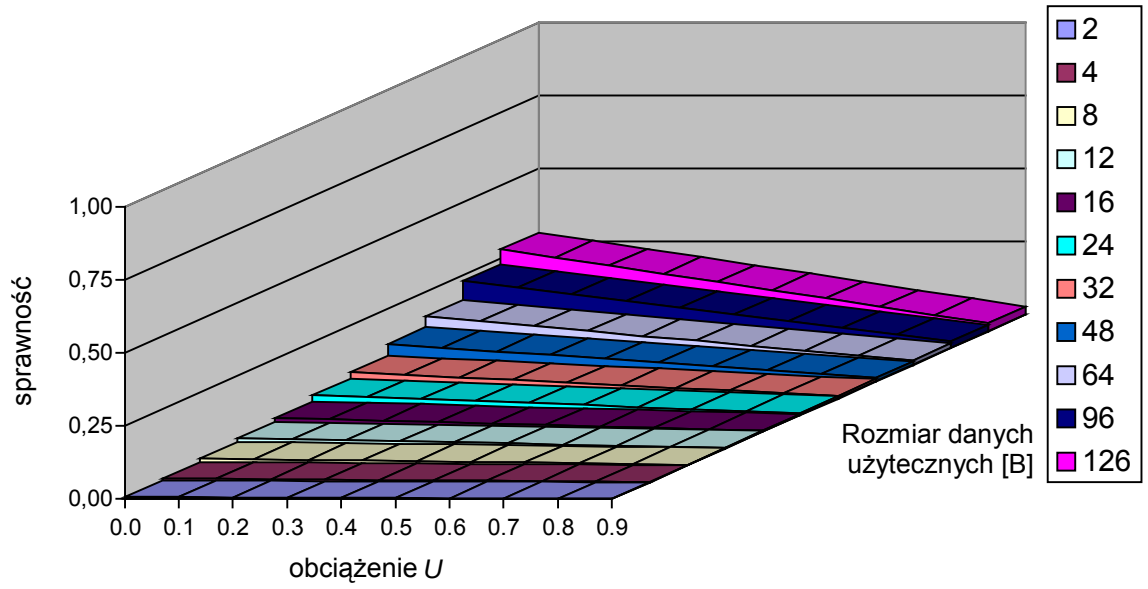
$$P = P_O(1-U) \text{ [b/s]}. \quad (69)$$

Powyższe zależności, tak jak dla UDP, umożliwiają szacowanie optymistyczne i tak jak poprzednio do obliczeń przyjęto wartości U z całego przedziału 0-1 ze skokiem 10%.

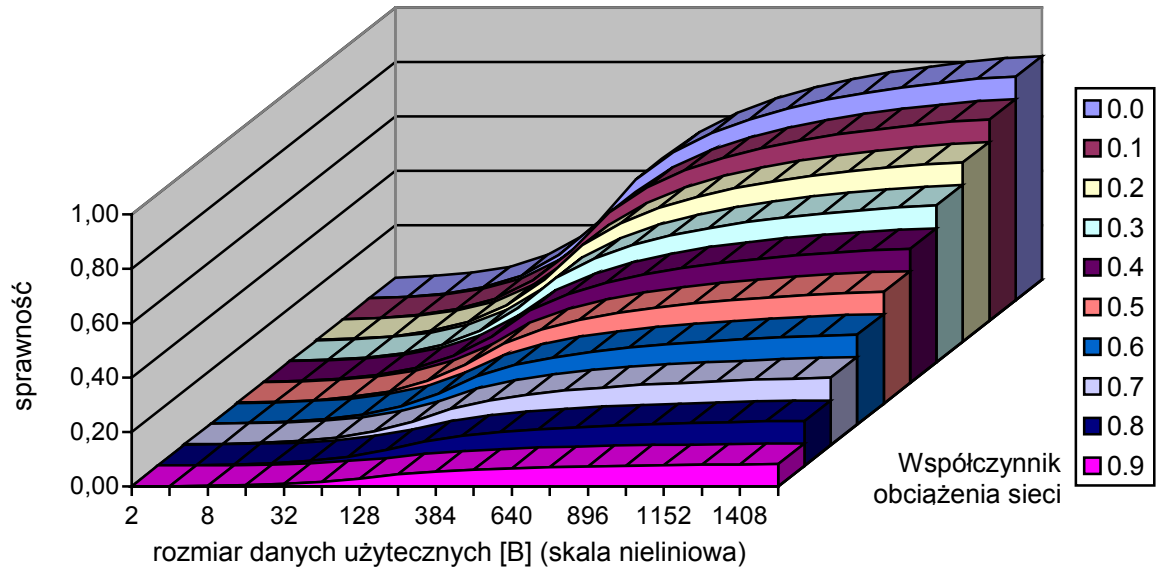
■

Jako maksymalny rozmiar pakietu przyjęto maksymalny rozmiar danych użytecznych przenoszonych przez ramkę Ethernetu, czyli 1492 bajty. Powyżej tego rozmiaru wyprowadzone wzory tracą znaczenie, ze względu na konieczność transmisji danych przy użyciu więcej niż jednej wymiany TCP.

Poniższe wykresy zawierają prezentację wyników obliczeń sprawności użytecznej w zależności od obciążenia sieci oraz rozmiaru transmitowanej paczki danych użytecznych.



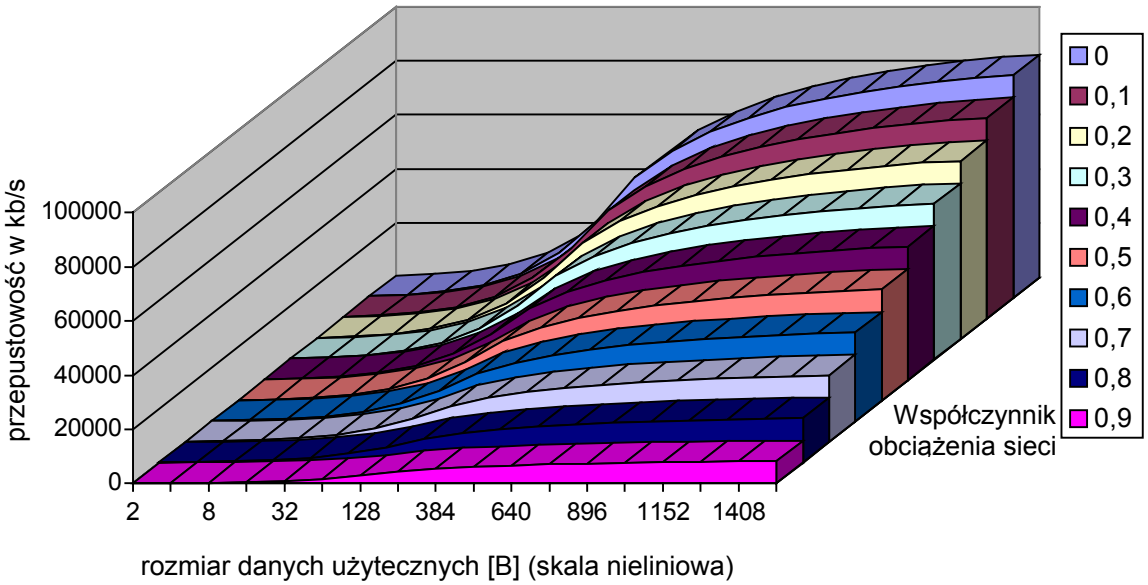
Rys. 67 Wykres zmiany sprawności protokołu TCP w funkcji wzrostu obciążenia i rozmiaru paczki danych użytecznych



Rys. 68 Wykres zmiany sprawności protokołu TCP w funkcji wzrostu rozmiaru danych użytecznych i obciążenia

Obserwowane zjawiska są analogiczne do protokołu UDP. Zmniejszenie efektywności protokołu wynika z konieczności stosowania mechanizmów niezawodnościowych i transmisji połączeniowej. W rzeczywistości, co nie zostało uwzględnione w analizie, protokół TCP względem protokołu UDP wprowadza dodatkowe opóźnienia wynikające z możliwości retransmisji pakietów, oraz posiada mechanizmy zapobiegania przeciążeniu przez dopasowywanie czasów oczekiwania na potwierdzenie. Testy laboratoryjne [14] pokazują, że protokół TCP jest w stanie utrzymać przepustowość użyteczną na poziomie ok. 8 [Mb/s] pomiędzy dwoma stacjami w sieci Ethernet przy prędkości 10 [Mb/s]. Wynik ten, pomimo

przyjętych do obliczeń uproszczeń, jest analogiczny do wyników otrzymanych dla pakietu o rozmiarze maksymalnym (1492 bajty), obciążenia sieci rzędu 0 lub 0,1 i prędkości 100 [Mb/s] (zob. wykres z rys. 69).



Rys. 69 Przepustowość protokołu TCP w funkcji rozmiaru paczki danych i obciążenia

11.3. Porównanie analizowanych parametrów

Porównując sprawność użyteczną analizowanych protokołów stwierdzono, że teoretyczna efektywność protokołów TCP i UDP jest bardzo silnie pomniejszana przez zjawiska opóźnień wynikające z braku mechanizmu nadzoru nad dostępem do medium. Przy obciążeniu rzędu 90% sprawność użyteczna realizowanych transakcji spada poniżej 10%.

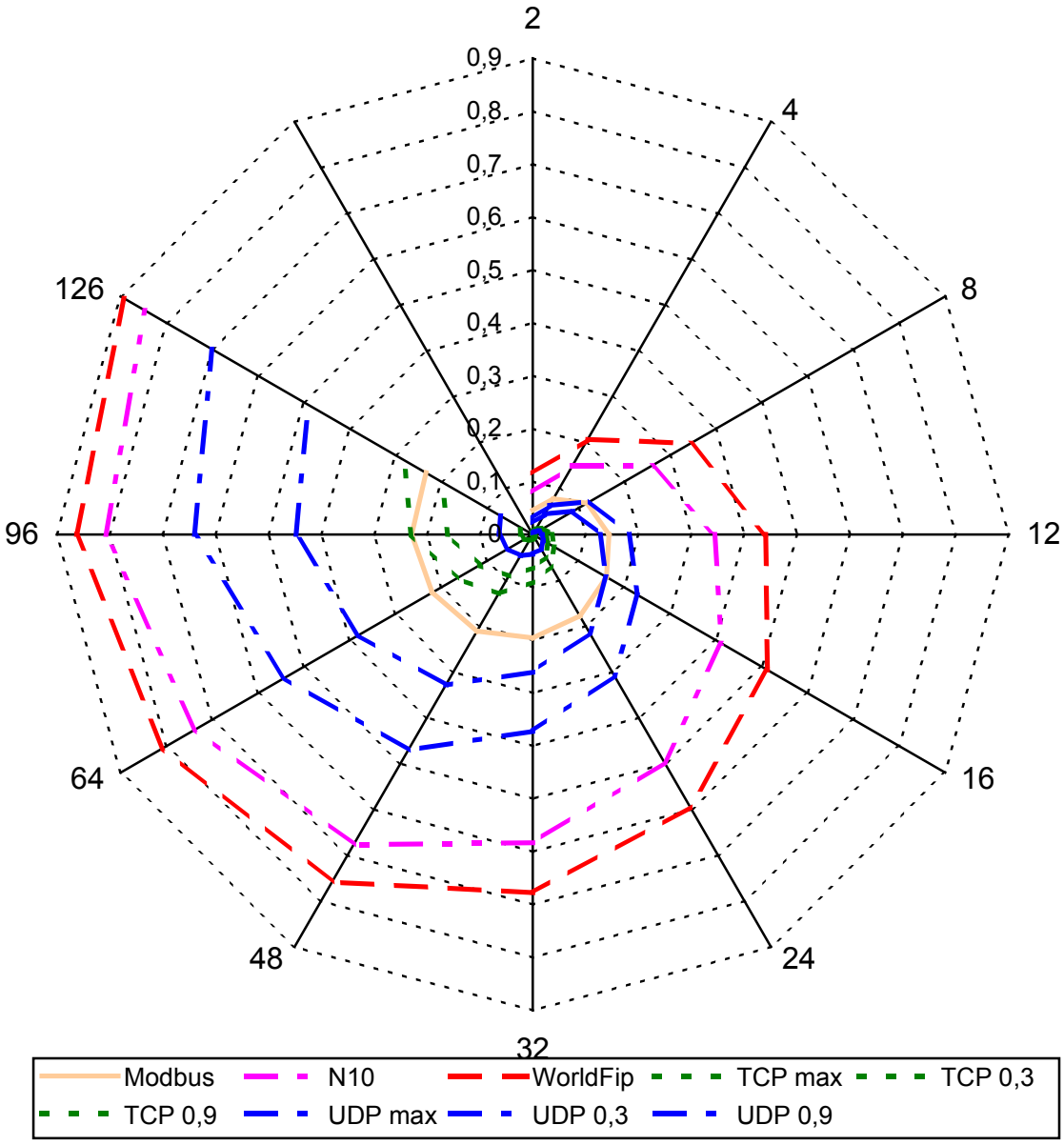
Porównując sprawność użyteczną powyższych protokołów stwierdzono, że teoretyczna efektywność protokołów TCP i UDP jest bardzo silnie pomniejszana przez zjawiska opóźnień wynikające z braku mechanizmu nadzoru nad dostępem do medium. Protokoły te są również znacząco mniej sprawne w porównaniu ze specjalizowanymi protokołami deterministycznymi. Na rysunku 70 przedstawiono wykres sprawności użytecznej transakcji cyklicznych analizowanych protokołów specjalizowanych oraz transakcji cyklicznych opartych na TCP/IP z przykładowym obciążeniem.

Oznaczenia przy nazwach protokołów na rysunku 70 i 71 należy interpretować jako:

TCP max – protokół TCP pracujący na łączu nieobciążonym,
TCP 0,3 – protokół TCP pracujący na łączu obciążonym ze współczynnikiem U=0,3,
TCP 0,9 – protokół TCP pracujący na łączu obciążonym ze współczynnikiem U=0,9,
UDP max – protokół UDP pracujący na łączu nieobciążonym,
UDP 0,3 – protokół UDP pracujący na łączu obciążonym ze współczynnikiem U=0,3,
UDP 0,9 – protokół UDP pracujący na łączu obciążonym ze współczynnikiem U=0,9.

Wykonana analiza wykazała, że sprawność użyteczna transakcji cyklicznych realizowanych na bazie Ethernetu i TCP/IP jest niska a wraz ze wzrostem obciążenia zdąża do zera. Wynika to z:

– konstrukcji pakietów,
– braku mechanizmów kontroli wymian.



Rys. 70 Wykresy sprawności porównywanych protokołów w funkcji obciążenia sieci i rozmiaru pakietu

Protokół TCP/IP został zaprojektowany z myślą o transmisji większych od przyjętych do analizy pakietów danych. Zakładając celowość transmisji dużych pakietów danych użytecznych, uzyskana sprawność byłaby wyższa.

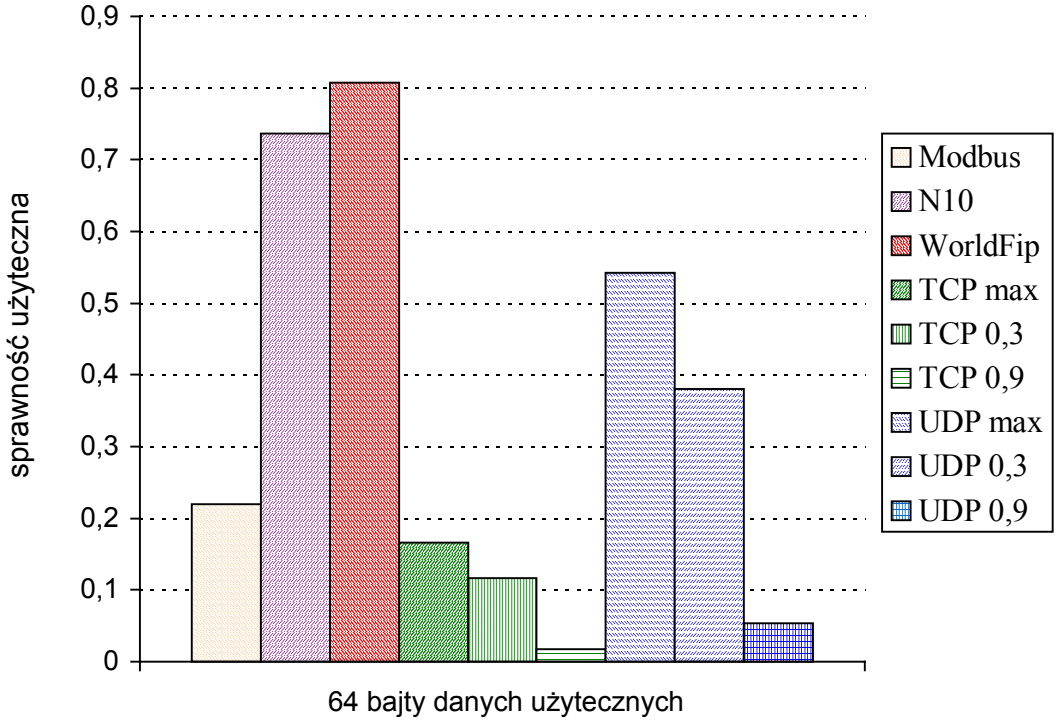
Niska sprawność nie dyskwalifikuje zastosowania protokołów TCP/IP w systemach przemysłowych. Z wykresu z rysunku 70 można odczytać, iż przemysłowy protokół Modbus ma również bardzo niską sprawność. Istotny jest fakt, że jeżeli zapewni się kontrolę wymian na całym segmencie sieci, to otrzyma się sieć zdeterminowaną czasowo o niskiej sprawności,

lecz o przepustowości gwarantującej realizację zadanych zadań komunikacyjnych. Zostało to pokazane na rysunku 75 w następnym rozdziale dotyczącym analizy przypadku zarządzania warstwą transportową.

W przypadku braku warstwy nadzorczej silna zależność sprawności od obciążenia sieci jest bardzo widoczna i nie sposób pominąć tej kwestii przy adaptacji UDP i TCP dla niedeterministycznych warstw komunikacyjnych systemów przemysłowych.

Powyższy wykres obrazuje również fakt opisywany wcześniej, a mianowicie, że protokół UDP charakteryzuje się znacząco wyższą sprawnością względem protokołu TCP.

Dla lepszego zobrazowania powyższych spostrzeżeń na rysunku 71 przedstawiono wycinek z wykresów z rysunku 70. Przedstawione dane dotyczą przypadku przesyłania pakietu obsługującego 64 bajtów danych użytecznych.



Rys. 71 Porównanie sprawności dla przykładowej paczki danych o rozmiarze 64 bajtów

11.4. Koncepcja wykorzystania protokołów TCP i UDP

Stos TCP/IP oferuje dwa protokoły transportowe TCP [103] oraz UDP [102]. Ich zastosowanie w systemach przemysłowych będzie zależne od sposobu wymiany informacji jakiego wymaga obsługiwany system. W celu takiego określenia oparto się na wcześniej opisanej strukturze systemu i charakterystyce ruchu (rozdział 5) oraz na analizie transakcji z poprzednich podrozdziałów.

Przy realizacji cyklicznych transakcji poziomych mechanizmy niezawodnego transferu protokołu TCP nie są istotne. Zadaniem wymian poziomych jest zapewnienie stabilnego cyklu wymiany danych pomiędzy abonentami na poziomie procesu. Implementacja potwierdzeń i retransmisji na tym poziomie może tylko zaszkodzić sprawności użytecznej łącza nie

wnosząc polepszenia mechanizmu przekazywania danych. Specjalizowane protokoły deterministyczne dla transakcji cyklicznych nie posiadają mechanizmów retransmisji ani potwierdzeń. Biorąc pod uwagę otrzymane wielkości sprawności użytecznej (rys. 70 i 71) protokół UDP daje lepsze od TCP wyniki, oferując szybki i sprawny transfer łącznie z możliwością rozgłaszania danych. Dotyczy to również wykorzystywania UDP wraz z wersją IPv6 [109].

Specyficzne wymiany poziome lub wymiany pionowe pomiędzy systemem przemysłowym a stacjami nadzorczymi mogą zawierać wymiany aperiodyczne, służące do jednorazowego przekazywania poleceń i rozkazów. Eliminacja mechanizmów niezawodności transmisji byłaby w tym przypadku wysoce nierozważna. Mogłoby zaistnieć zjawisko utraty transmitowanego polecenia i w efekcie jego niewykonanie przez obiekt. W sytuacji, gdy system nie wymaga potwierdzenia wykonania rozkazu w innej formie, np. przez stan innej zmiennej, wówczas użytkownik może mieć mylne przekonanie o jego wykonaniu. Implementacja mechanizmów potwierdzeń i retransmisji pakietów dla wymian pionowych oraz poziomych wymian aperiodycznych stanowi konieczność, niezależnie od sposobu jej realizacji.

Wykorzystanie protokołów transportowych TCP i UDP zależy od dziedziny zastosowań obsługiwanych transakcji. Proponowana koncepcja polega na tym, iż dla wymian cyklicznych wykorzystywanych głównie do monitorowania powinno się stosować transmisję danych w oparciu o protokół UDP, a dla wymian aperiodycznych w oparciu o protokół TCP. Oczywiście podział ten wynika tylko z przeprowadzonego wcześniej porównania czasów trwania transakcji i sposobu realizacji połączenia. Funkcjonalnie nie ma przeszkód, aby całość transmisji pomiędzy stacjami przeprowadzać za pomocą połączenia TCP lub UDP. Połączenie UDP może się nadawać do wykorzystania przy założeniu istnienia informacyjnego sprzężenia zwrotnego od adresata do nadawcy datagramu. Jednak dla dalszych rozważań przyjęto bazować na wykorzystaniu omawianych protokołów transportowych względem realizowanych przez nie funkcji.

Analiza czasowa dla przypadku zarządzania warstwami TCP lub UDP przeprowadzona w następnym rozdziale pokaże, że dobór protokołu transportowego zależy dodatkowo od idei działania samych transakcji i dla niektórych z nich nie jest możliwe stosowanie dowolnego protokołu transportowego.

12. Analiza czasowa przepływu informacji w przypadku zarządzania warstwą transportową

Oferowana wysoka przepustowość łącza zbudowanego na bazie sieci Ethernet wykazana w rozdziale 11 nie kwalifikuje w sposób automatyczny takiego łącza jako łącza nadającego się do zastosowań w systemach przemysłowych (rozdz. 4.5, 5, 6) [34]. Niezależnie od zastosowanego standardu [78, 45, 79] oraz topologii połączeń łącze nie zapewnia determinizmu wymiany zmiennych sieciowych. Najprostszym sposobem wprowadzenia determinizmu w dostępie do medium jest implementacja warstwy aplikacyjnej kontrolującej wymiany realizowane przez warstwę transportową według konkretnego modelu kontroli wymian (rozdział 6, 8). W niniejszym rozdziale proponuje się wprowadzenie prostych protokołów aplikacyjnych umożliwiających kontrolę wymian i analizę sprawności i przepustowości użytecznej stosu zbudowanego z wykorzystaniem takich protokołów. Rozważanie oparto na modelu o niskiej sprawności: Master – Slave [61] oraz na wysokosprawnym modelu PDC [61, 58]. Do realizacji obliczeń przyjęto założenie, iż wszystkie wymiany wykonują się w systemie separowanym (rozdział 7.2).

12.1. Protokół oparty na modelu Master – Slave

Wszystkie transakcje przekazywania danych pomiędzy abonentami w modelu Master – Slave odbywają się przez pośrednictwo stacji Master i adresowanie urządzeń. W rozważaniach skoncentrowano się na standardowej cyklicznej transakcji sieci Modbus działającej wg schematu ze strony 112. Do analizy uwzględniono oktety dokładane przez warstwę aplikacyjną stanowiące niezbędny element mechanizmu działania protokołu, w szczególności adresacji danych.

Korzystając ze wzorów 39 i 40 oraz analizy protokołu UDP z rozdziału 11.2.4, a także odrzucając warstwy protokołu Modbus (preambuły oraz CRC) zachowując przy tym funkcjonalność transakcji otrzymano czas trwania transakcji:

$$T_T = T_{UDP\_\dot{Z}} + T_{UDP\_O} + T_{UDP\_Z} + T_{UDP\_P} \text{ [s]}, \qquad (70)$$

gdzie T_{UDP_x} są czasami transmisji poszczególnych wymian w transakcji (zgodnie z oznaczeniami na stronie 112 i wzorami 53 i 54) przez protokół UDP. Czasy transmisji żądania $T_{UDP\_\dot{Z}}$ i potwierdzenia T_{UDP_P} są jednakowe. Czasy odpowiedzi zależą do liczby transmitowanych danych. Dla składowych $T_{UDP\_\dot{Z}}$ oraz T_{UDP_P} a także dla rozmiaru danych użytecznych 2 i 3 bajtów i składowej T_{UDP_O}, istnieje konieczność transmisji ramki Ethernetu minimalnego rozmiaru, zatem:

$$T_T = \frac{512}{V} + \frac{512}{V} + \frac{8(54+n)}{V} + \frac{512}{V} \text{ [s]} \qquad \text{dla } n \geq 2 \text{ i } n < 3, \qquad (71)$$

$$T_T = \frac{512}{V} + \frac{8(54+n)}{V} + \frac{8(54+n)}{V} + \frac{512}{V} \text{ [s]} \qquad \text{dla } n \geq 3, \qquad (72)$$

upraszczając otrzymujemy:

$$T_T = \frac{8(246+n)}{V} \text{ [s]} \qquad \text{dla } n \geq 2 \text{ i } n < 3, \qquad (73)$$

$$T_T = \frac{16(118+n)}{V} \text{ [s]} \qquad \text{dla } n \geq 3. \qquad (74)$$

Stąd sprawność użyteczna:

$$\eta = \frac{T_U}{T_T} = \frac{\frac{8n}{V}}{\frac{8(246+n)}{V}} = \frac{n}{246+n} \qquad \text{dla } n \geq 2 \text{ i } n < 3, \qquad (75)$$

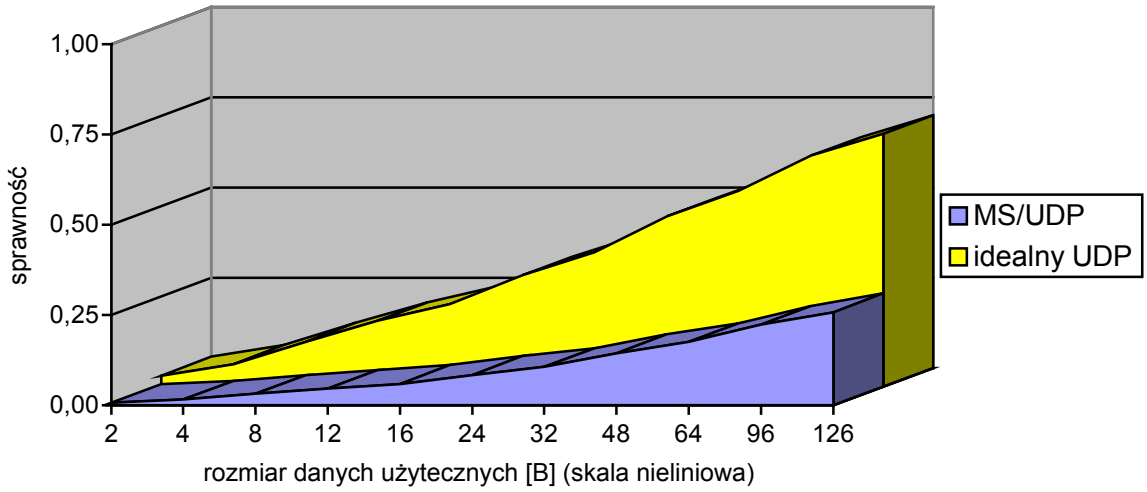
$$\eta = \frac{T_U}{T_T} = \frac{\frac{8n}{V}}{\frac{16(118+n)}{V}} = \frac{n}{2(118+n)} \qquad \text{dla } n \geq 3, \qquad (76)$$

oraz przepustowość użyteczną:

$$P = \eta V \text{ [b/s]}. \qquad (77)$$

■

Otrzymane wyniki przedstawiono na poniższym wykresie.



Rys. 72 Porównanie sprawności protokołu UDP ze sprawnością wymian modelu Master – Slave opartych na UDP w funkcji rozmiaru danych użytecznych

Na wykresie widać ubytek ponad 60% sprawności idealnych wymian UDP.

12.2. Protokół oparty na modelu PDC

Do analizy przyjęto protokół oparty na mechanizmie kontroli wymian wykorzystywany w wymianach cyklicznych zmiennych oraz acyklicznych wymianach zmiennych na żądanie protokołu Fip/WorldFip [116, 115, 114, 119]. Model PDC nie jest modelem popularnym, jednak warto poświęcić mu uwagę ze względu na jego wysokie parametry efektywności [61, 58].

Ponieważ wykorzystano ideę działania, a nie sam protokół WorldFip, można odrzucić wszystkie dane serwisowe dokładane przez warstwy poniżej aplikacji. Przy rozsyłaniu pakietów protokół będzie wykorzystywać możliwość rozgłaszania (ang. *broadcast*) w adresacji TCP/IP. Odfiltrowaniem danych użytecznych będzie zajmować się warstwa aplikacji każdego abonenta.

12.2.1. Transakcje periodyczne

Dla transakcji cyklicznych opartych o wymiany UDP wystąpi niewielka utrata efektywności działania względem standardowego UDP. Wynika to z faktu, iż transakcje cykliczne modelu PDC wymagają arbitralnego zapytania o informację z poziomu abonenta – dystrybutora, czego nie wymaga żadna z warstw TCP/IP. Transakcja w proponowanym protokole PDC/UDP będzie przebiegała wg schematu:

1. Arbiter: zapytanie o nazwę zmiennej cyklicznej,
2. Producent: odpowiedź z informacją użyteczną związaną z daną zmienną.

To dodatkowe zapytanie stanowi niewielką paczkę danych wymagającą transmisji pojedynczej ramki Ethernetu o minimalnym rozmiarze. Ponieważ zawsze będzie miało stały rozmiar i nie przenosi danych użytecznych, to jego wpływ objawi się jako stały ubytek sprawności niezależny od liczby transmitowanych danych użytecznych.

Względem sprawności protokołu WorldFip, należy uwzględnić tylko liniowy jej spadek wynikający ze wzrostu liczby bitów systemowych dołączanych do każdej wymiany przez warstwy TCP/IP. Warstwa aplikacyjna analizowanego protokołu PDC/UDP nie dokłada dodatkowych oktetów nie stanowiących danych użytecznych, gdyż mechanizmy adresacji abonentów w protokołach PDC są zbędne a adresacja informacji jest realizowana przez pakiety od abonenta – dystrybutora. Korzystając ze wzoru 49 oraz wzorów 53 i 54 obliczono czas transakcji:

$$T_T = T_{UDP\_\dot{Z}} + T_{UDP\_O}\,[\mathrm{s}], \tag{78}$$

$$T_T = \frac{512}{V} + \frac{512}{V} = \frac{1024}{V}\,[\mathrm{s}] \qquad \text{dla } n \geq 2 \text{ i } n \leq 10, \tag{79}$$

$$T_T = \frac{512}{V} + \frac{8(54+n)}{V} = \frac{8(118+n)}{V}\,[\mathrm{s}] \qquad \text{dla } n > 10, \tag{80}$$

Na tej podstawie obliczono sprawność użyteczną dla wymian cyklicznych:

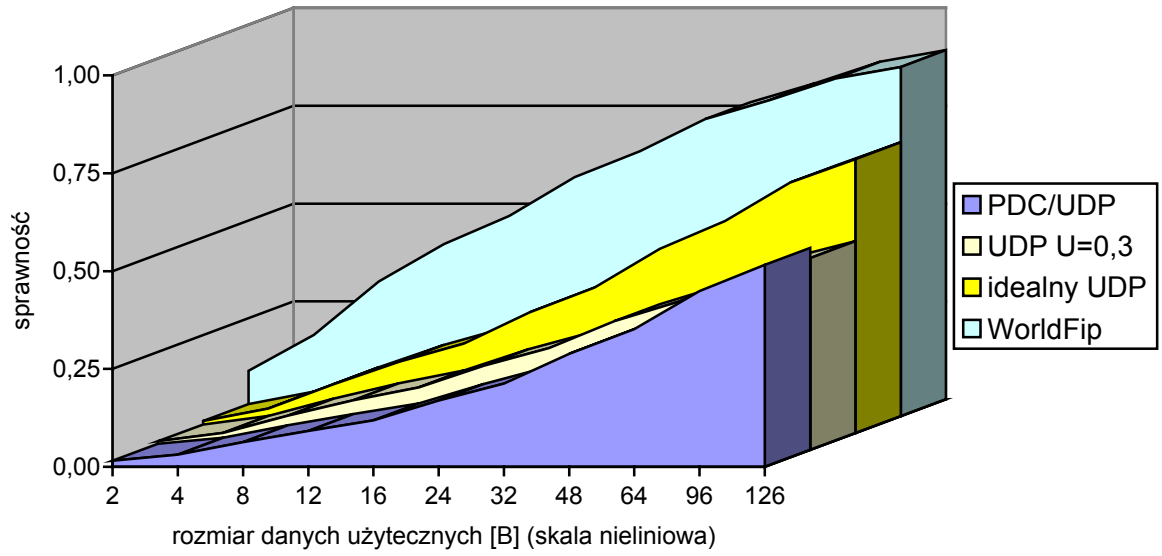
$$\eta = \frac{T_U}{T_T} = \frac{\frac{8n}{V}}{\frac{1024}{V}} = \frac{n}{128} \qquad \text{dla } n \geq 2 \text{ i } n < 10, \qquad (81)$$

$$\eta = \frac{T_U}{T_T} = \frac{\frac{8n}{V}}{\frac{8(118+n)}{V}} = \frac{n}{n+118} \qquad \text{dla } n > 10, \qquad (82)$$

oraz przepustowość użyteczną:

$$P = \eta V \text{ [b/s]}. \qquad (83)$$

Dla obliczeń uwzględniono typową prędkość transmisji dla sieci Ethernet równą 100Mb/s. Wartości sprawności są przedstawione na poniższym wykresie w porównaniu ze sprawnościami protokołów UDP bez kontroli wymian dla przypadku idealnego oraz wymian cyklicznych protokołu deterministycznego WorldFip.



Rys. 73 Porównanie sprawności protokołów UDP z kontrolą wymian, UDP bez kontroli oraz WorldFip w funkcji rozmiaru transmitowanych danych użytecznych

Z wykresu widać, iż występuje spadek sprawności rzędu 30% dla całego przedziału analizowanych rozmiarów danych względem sprawności idealnej protokołu UDP. Uzyskana sprawność jest porównywalna dla typowego obciążenia sieci (U=0,3). Kosztem utraty 30 procent sprawności użytecznej uzyskuje się prostą komunikację ze zdeterminowanym w czasie dostępu do medium.

12.2.2. Transakcje aperiodyczne

Dla wymian aperiodycznych, zgodnie z wcześniejszymi wnioskami przyjęto wykorzystanie protokołu TCP. Pojedyncza transakcja dla proponowanego protokołu TCP/UDP przebiegnie wg schematu:

1. Arbiter: zapytanie o nazwę żądanej zmiennej ($T_{UDP\_\dot{Z}}$),
   – 1 identyfikator: 2 oktety,
2. Abonent żądający: odpowiedź abonenta żądającego (T_{UDP_O})
   – 1 identyfikator: 2 oktety,
3. Arbiter: zapytanie o żądaną zmienną (T_{UDP_Z}),
   – 1 identyfikator: 2 oktety,
4. Producent: odpowiedź producenta żądanej zmiennej (T_{UDP_D}),
   – dane użyteczne: n oktetów.

Korzystając ze wzoru 49 oraz wzorów 53 i 54 obliczono czas transakcji:
$T_T = T_{UDP\_\dot{Z}} + T_{UDP\_O} + T_{UDP\_Z} + T_{UDP\_D}$ [s],

$$T_T = \frac{512}{V} + \frac{512}{V} + \frac{512}{V} + \frac{512}{V} = \frac{2048}{V} \text{ [s]} \quad \text{dla } n \geq 2 \text{ i } n \leq 10, \tag{84}$$

$$T_T = \frac{512}{V} + \frac{512}{V} + \frac{512}{V} + \frac{8(54+n)}{V} = \frac{8(246+n)}{V} \text{ [s]} \quad \text{dla } n > 10. \tag{85}$$

Dla każdej z powyższych wymian musi nastąpić interakcja pomiędzy dwoma abonentami. Realizacja takiej transakcji nie jest tak prosta jak dla przypadku UDP. Protokół TCP jest protokołem połączeniowym. Wykonanie rozgłoszenia nie jest możliwe. Niezbędne zatem jest istnienie procesu w warstwach komunikacyjnych abonentów, który będzie przechowywał tablicę powiązań nazw zmiennych systemowych z adresami IP abonentów produkującymi daną zmienną i wykonywał przeadresowanie na żądanie połączenia z warstw nadrzędnych. Umożliwi to nawiązywanie połączeń pomiędzy abonentami na podstawie nazw żądanych zmiennych. Kolejny problem pojawi się w momencie odpowiedzi producenta. Aby zachować spójność danych oferowaną przez model PDC należałoby nawiązać połączenia ze wszystkimi abonentami zainteresowanymi daną zmienną. Nie jest to możliwe ze względu na brak informacji dotyczącej innych abonentów poza abonentem żądającym. Zatem niezbędne staje się ograniczenie działania transakcji do odpowiedzi do tylko tego abonenta, który zażądał danej zmiennej. Przy czym w zapytaniu o tą zmienną należy przekazać producentowi adres IP abonenta lub nazwę produkowanej przez niego zmiennej, w celu umożliwienia nawiązania z nim połączenia. Poniżej przedstawiono zmodyfikowany schemat transakcji aperiodycznej modelu PDC dostosowany do protokołu TCP.

1. Arbiter: nazwa zmiennej → IP abonenta żądającego → nawiązanie połączenia → pakiet zapytania o nazwę żądanej zmiennej
2. Abonent żądający: pakiet z nazwą żądanej zmiennej
3. Arbiter: nazwa żądanej zmiennej → IP producenta → nawiązanie połączenia → pakiet zapytania o informację + IP abonenta żądającego
4. Producent: IP abonenta żądającego → nawiązanie połączenia → pakiet odpowiedzi z informacją związaną z żądaną zmienną

Adres IP wraz z nazwą zmiennej może być traktowany jako para adresująca informację w sieci. Wystąpi uniezależnienie to od adresów sprzętowych umożliwiając przynajmniej

częściowe adresowanie informacji, a nie urządzenia. Przetworzenie IP na sprzętowe adresy Ethernetowe spoczywa na protokołach warstw niższych (np. ARP). Oczywiście działanie tych protokołów w sytuacjach wymagających odświeżania tablic odwzorowujących IP w adresy fizyczne oparte jest na swobodnym rozgłaszaniu, co może zaburzyć determinizm czasowy wynikający z działania warstw wyższych. Zjawisko to może mieć również miejsce w przypadku korzystania z protokołu UDP.

Przedstawiony sposób wykorzystania protokołu TCP dla wymian aperiodycznych działających według modelu PDC ma szereg wad:

– konieczność istnienia dodatkowej tablicy przeadresowań,
– konieczność istnienia dodatkowego typu zapytania o zmienną z przekazaniem adresu IP,
– utrata spójności czasowej aktualizacji zmiennych w skali całej sieci,
– strata czasu na tworzenie transmisji połączeniowej.

Budowanie mechanizmu wymian aperiodycznych w systemach separowanych na bazie protokołu TCP nie jest konieczne. Jeżeli mechanizm kontroli wymian zapewnia determinizm dostępu do medium, to transmisje połączeniowe nie zwiększą w sposób istotny niezawodności sieci, natomiast skomplikują w sposób istotny warstwę aplikacji i wyeliminują cechy protokołu istotne dla procesu. Wniosek ten jest ogólny i dotyczy każdego przypadku wykorzystania modeli kontrolowania dostępu do medium.

Poniżej wyliczono jak będzie kształtować się sprawność użyteczna w przypadku wymian aperiodycznych wg schematu ze strony 133 dla modelu PDC i protokołu UDP.

Obliczono sprawność użyteczną:

$$\eta = \frac{T_U}{T_T} = \frac{\frac{8n}{V}}{\frac{2048}{V}} = \frac{n}{256} \qquad \text{dla } n \geq 2 \text{ i } n \leq 10, \qquad (86)$$

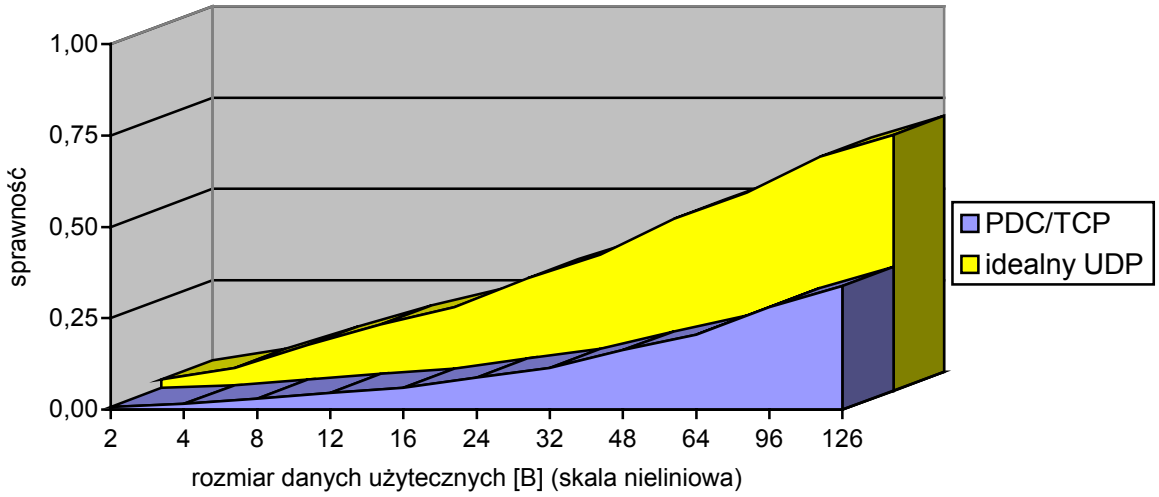
$$\eta = \frac{T_U}{T_T} = \frac{\frac{8n}{V}}{\frac{8(246+n)}{V}} = \frac{n}{n+246} \qquad \text{dla } n > 10, \qquad (87)$$

oraz przepustowość użyteczną:

$$P = \eta V \text{ [b/s]}. \qquad (88)$$

■

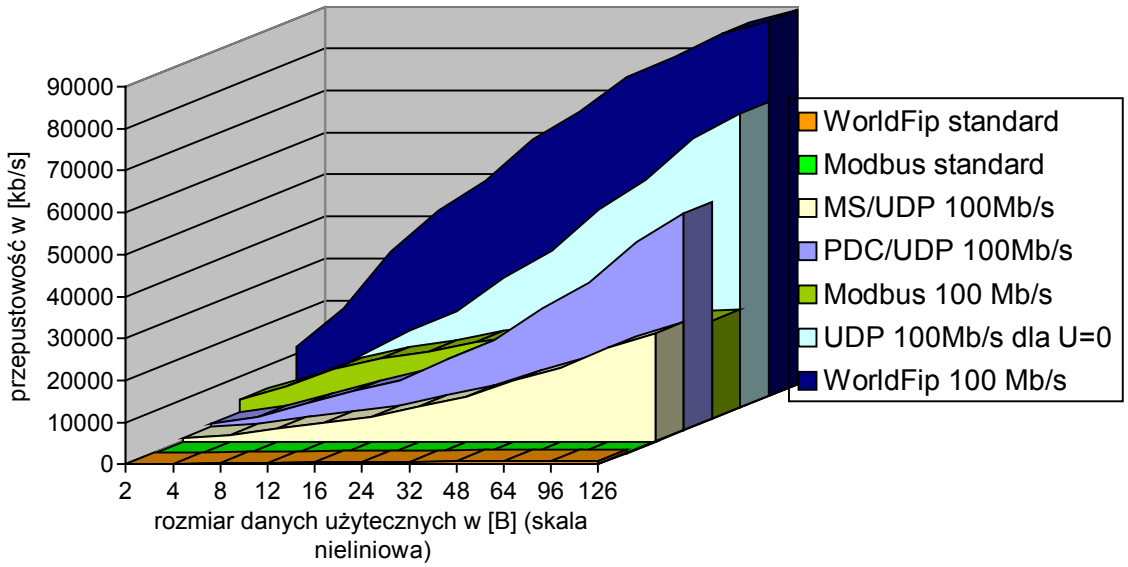
Wartości sprawności są przedstawione na poniższym wykresie w porównaniu ze sprawnościami protokołów UDP bez kontroli wymian dla przypadku idealnego. Ubytek sprawności jest rzędu ponad 50%. Wynika to z konieczności prowadzenia dialogu pomiędzy abonentami. Wraz ze wzrostem liczby abonentów biorących udział w transakcji wydłuża się również sama transakcja.



Rys. 74 Porównanie sprawności protokołu UDP ze sprawnością wymian aperiodycznych modelu PDC/UDP w funkcji rozmiaru danych użytecznych

12.3. Wnioski z analizy zarządzania warstwą transportową

Wykonano porównanie przepustowości analizowanych rozwiązań dla prędkości 100 Mb/s. Wyniki przedstawiono na rysunku 75.



Rys. 75 Porównanie przepustowości protokołu UDP oraz przepustowości rozważanych protokołów specjalizowanych w funkcji rozmiaru danych użytecznych

Wykorzystanie zarządzania wymianami dla protokołu UDP nie polepsza sprawności sieci, a wręcz przeciwnie, ze względu na swój algorytmiczny narzut czasowy parametry te pogarsza. Jednak z punktu widzenia przepustowości użytecznej, żaden deterministyczny protokół pracujący na swojej standardowej prędkości nie zapewnia takich wartości przepustowości, jakie można osiągnąć przy wykorzystaniu TCP/IP i Ethernetu. Zobrazowano to na wykresie z rysunku 75. Przedstawiono na nim przepustowości protokołów z kontrolą

wymian typu PDC i Master – Slave wraz z wartościami przepustowości dla typowych protokołów deterministycznych wykorzystujących te modele. Przepustowość protokołów deterministycznych podano dla prędkości standardowych oraz prędkości teoretycznych na poziomie 100 Mb/s.

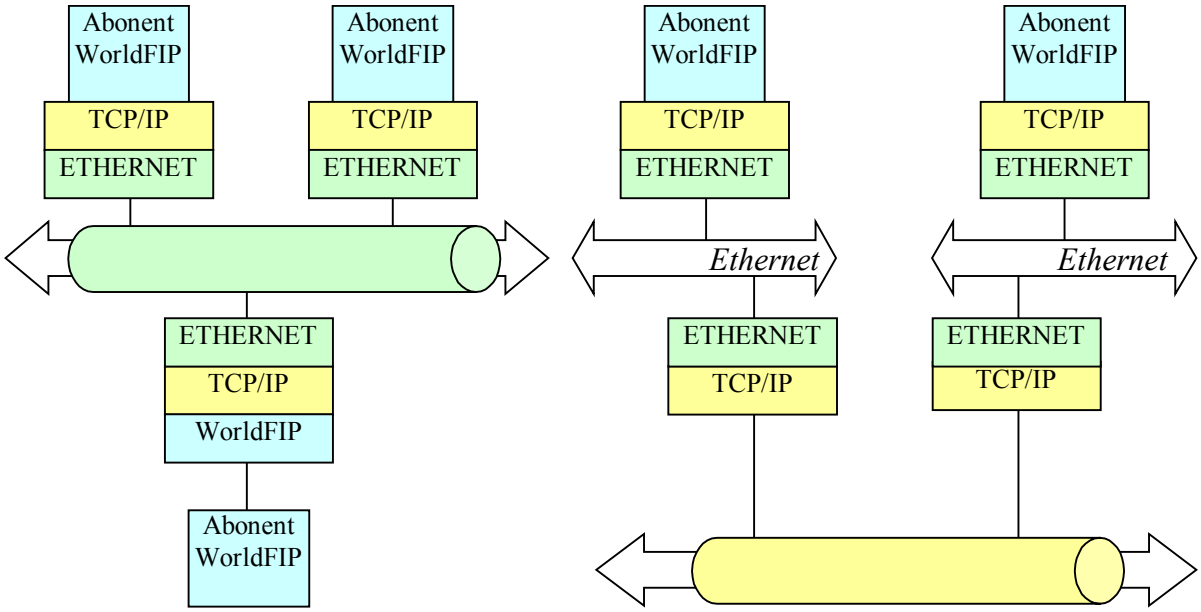
Reasumując:

- W sieci, w której występuje brak obcego ruchu, istnieje możliwość stworzenia nadbudowy warstwy aplikacji kontrolującej kolejność wymian i tworzącej zdeterminowany w czasie dostęp do medium.
- Najlepszym protokołem transportowym do realizacji takiego stosu jest protokół UDP. Zastosowanie protokołu TCP jest możliwe, lecz nie zalecane, gdyż może przysporzyć kłopotów implementacyjnych dla niektórych modeli sieci.
- Warstwy aplikacji wszystkich stacji zapisujących informację na sieć muszą zostać zmodyfikowane w przeciwieństwie do stacji tylko odczytujących, których stos TCP/IP nie musi być rozszerzany.
- Otrzymana sprawność protokołu może być niższa od sprawności protokołu UDP, jednak przyjmie ona konkretną wartość i nie będzie zależna od obciążenia sieci.
- Zakłócenia cyklu pracy sieci mogą występować w sytuacji rekonfigurowania adresacji, wprowadzenia obcego ruchu lub wykorzystywanie w stosie TCP/IP protokołów generujących wymiany bez kontroli warstwy aplikacji.
- Należy dokonać wyboru rozwiązania z aplikacyjnym zarządzaniem wymianami, gdy istnieją przesłanki do stosowania protokołu TCP/IP, np. wymagana przepustowość lub względy ekonomiczne i wymagany jest czas krytyczny realizacji wymian.

Przedstawione rozwiązanie oparte na protokołach aplikacyjnych kontrolujących wymiany staje się realizowalne w praktyce, gdy projektant ma wpływ na konstrukcję warstw aplikacyjnych stosowanych abonentów. Znacznie częściej zachodzi sytuacja, gdy nie wszyscy abonenci systemu umożliwiają modyfikację warstwy 7 swojego interfejsu komunikacyjnego. Dla takich przypadków przedstawione rozwiązanie nie jest realizowalne. Kolejny rozdział traktuje o alternatywnym podejściu do problemu kontrolowania wymian, w którym abonenci dysponują interfejsami specjalizowanymi a ich pakiety są tunelowane w sieci systemowej opartej na protokole TCP/IP i sieci Ethernet.

13. Analiza czasowa przepływu informacji w przypadku tunelowania protokołów deterministycznych

Problem implementacyjny zarządzania wymianami pojawia się, gdy interfejsy abonentów sieci systemowej opartej na standardzie Ethernet i protokole TCP/IP nie umożliwiają swobodnej rekonfiguracji. Rozwiązanie, które umożliwia wykorzystywanie takich abonentów, stanowi zastosowanie modułów tunelujących protokół tych abonentów w sieci systemowej. Tunelowanie [14] stanowi proces polegający na transmisji danych przez umieszczanie pakietów danego rodzaju protokołów w polu danych protokołu wykorzystywanego do transportu w danym systemie komunikacyjnym. Tunelowanie wykorzystuje proces kapsułkowania, czyli obsługi pakietu pochodzącego z wyższej warstwy stosu protokołów przez warstwę niższą taj jakby to były dane użyteczne. Tunelowanie stanowi rozwinięcie omawianego wcześniej zagadnienia sterowania warstwą transportową. Celem analizowanego tunelowania jest uzyskanie możliwości wymiany warstwy fizycznej tunelowanego protokołu na warstwy sieci Ethernet i stos TCP/IP. Uzyska się tym samym pełną funkcjonalność protokołu tunelowanego wraz z zachowaniem jego standardu od warstwy drugiej modelu ISO/OSI przy jednoczesnym wykorzystaniu prędkości transmisji sieci Ethernet i uniwersalności łącza bazującego na TCP/IP.



a) Tunel w systemie lokalnym *b) Tunel pomiędzy abonentami globalnymi*

Rys. 76 Lokalizacja tuneli względem rodzajów stosowanych systemów

Istnieją dwa sposoby zestawiania tunelu pomiędzy abonentami. Sposoby te wynikają z rodzaju stosowanych systemów (rys. 76). W niniejszym rozdziale wykonano porównanie sprawności specjalizowanego protokołu deterministycznego ze sprawnością tego protokołu w wersji tunelowanej przez stos TCP/IP. Rozważania przeprowadzono dla analizowanego wcześniej (11.2.3, 12.2) modelu deterministycznego PDC i propozycji mechanizmu tunelowania protokołu WorldFIP z wykorzystaniem protokołów UDP oraz TCP. Analizę przeprowadzono dla przypadku systemu lokalnego z rysunku 76a oraz współpracy intersieciowej z rysunku 76b.

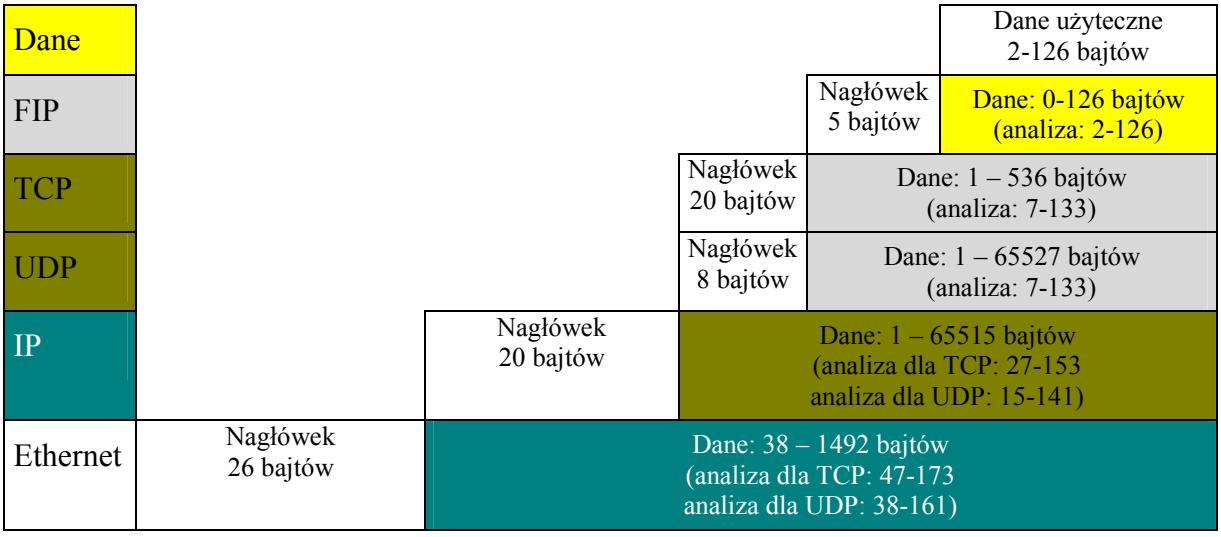
13.1. Tunelowanie protokołu WorldFIP w systemie lokalnym

Analiza tunelowania transakcji protokołu WorldFIP będzie zbliżona do analizy zarządzania wymianami dla modelu wymian PDC przedstawionej w rozdziale 12.2. W rozważanym przypadku tunelowania uwzględniono konkretną implementacje transakcji cyklicznych i acyklicznych [114, 116] wraz z danymi dokładanymi przez warstwy tego protokołu. Czasy transmisji poszczególnych ramek w protokole WorldFIP wyrażają następujące zależności (por. rozdz. 11.2.3):

Czas transmisji ramki zapytania: $\frac{61}{V}$ [s]

Czas transmisji ramki danych: $\frac{61+8n}{V}$ [s] dla n = 1..126

Dla potrzeb analizy odrzucono pola dokładane przez warstwy fizyczne protokołu WorldFIP. Założono, iż w zamian standardowej warstwy fizycznej wykorzystywana jest warstwa sieci Ethernet. Pola te mają odpowiednio rozmiar 14 i 7 bitów. Powoduje to zmniejszenie umownego nagłówka pakietów WorldFIP do 40 bitów, czyli pięciu bajtów.



Rys. 77 Kapsułkowanie zmiennej protokołu WorldFIP

Kapsułkowanie ramki danych można uogólnić zgodnie z rysunkiem 77.

Dla ramki zapytania, po usunięciu warstwy fizycznej, rozmiar nagłówka wynosi 3 bajty a pola danych wynosi 2 bajty. Rozmiary pakietów wynoszą:

$$S_{ID} = 5 = const, \quad (89)$$

$$7 \le S_D \le 133, \quad (90)$$

gdzie:

S_{ID} jest rozmiarem pakietu zapytania,

S_D jest rozmiarem pakietu przenoszącego dane użyteczne.

13.1.1. Transakcja periodyczna

Zgodnie z wnioskami z rozdziału 11.4 dla transakcji periodycznej, w warstwie transportowej brano pod uwagę protokół UDP. Korzystając ze wzoru 49 czas trwania transakcji cyklicznej zmiennej dla protokołu WorldFIP będzie wynosił:

$$T_T = T_{ID} + T_{TD}. \quad (91)$$

Cały narzut w rozmiarze pakietu dostarczonego do warstwy Ethernetu względem oryginalnego pakietu WorldFIP wynosi $S_{ADD} = 28$ bajtów. Ze wzoru 89 wynika, iż dla pakietu zapytania ramka Ethernetu musi mieć rozmiar minimalny (38 bajtów danych), gdyż:

$$S_{ID} + S_{ADD} < 38,$$

gdzie:

S_{ADD} jest rozmiarem danych serwisowych w bajtach pochodzących od narzutu protokołów tunelujących,

Natomiast dla pakietu danych, z przedziału opisanego nierównością 90 wynika, iż minimalny rozmiar ramki Ethernetu musi wystąpić dla

$$n < 38 - (S_{ADD} + 5),$$

czyli dla $n < 5$.

Dla n≥5 rozmiar ramki Ethernetu będzie większy od minimalnego. Zatem dla tunelowania transakcji cyklicznej przez UDP i Ethernet otrzymano:

$$T_T = \frac{512}{V} + \frac{512}{V} \text{ [s]}$$

$$T_T = \frac{1024}{V} \text{ [s]} \qquad \text{dla } n<5. \quad (92)$$

oraz

$$T_T = \frac{512}{V} + \frac{8(59+n)}{V} \text{ [s]}$$

$$T_T = \frac{8(123+n)}{V} \text{ [s]} \qquad \text{dla } n\ge5. \quad (93)$$

Aby rozmiar danych datagramu IP był większy lub równy 38 bajtów, przy wykorzystywaniu protokołu UDP warstwa aplikacyjna WorldFIP musi dołożyć przynajmniej 5 bajtów danych użytecznych. Gdy to nie zachodzi, brakujące bajty muszą wypełnić warstwy tunelujące.

■

Bazując na zależnościach *31* i *32* dla systemu z zamkniętym lub separowanym obiegiem informacji sprawność użyteczna będzie wynosiła:

$$\eta_O = \frac{T_U}{T_T} = \frac{\frac{8n}{V}}{\frac{1024}{V}} = \frac{n}{128} \qquad \text{dla } n \geq 2 \text{ i } n < 5 \text{ B},$$

$$\eta_O = \frac{T_U}{T_T} = \frac{\frac{8n}{V}}{\frac{8(123+n)}{V}} = \frac{n}{n+123} \qquad \text{dla } n \geq 5 \text{ B}, \qquad (94)$$

a przepustowość użyteczna:

$$P_O = \frac{L_U}{T_T} = \frac{8n}{\frac{1024}{V}} = \frac{nV}{128} \text{ [b/s]} \qquad \text{dla } n \geq 2 \text{ i } n < 5 \text{ B},$$

$$P_O = \frac{8n}{\frac{8(123+n)}{V}} = \frac{nV}{n+123} \text{ [b/s]} \qquad \text{dla } n \geq 5 \text{ B}. \qquad (95)$$

Dla przypadku sieci z otwartym obiegiem informacji uwzględniono opóźnienia. Zatem sprawność użyteczna na podstawie zależności 59 i 63 została szacowana ze wzoru:

$$\eta = \eta_O(1-U), \qquad (96)$$

a przepustowość na podstawie zależności 60, 64 ze wzoru:

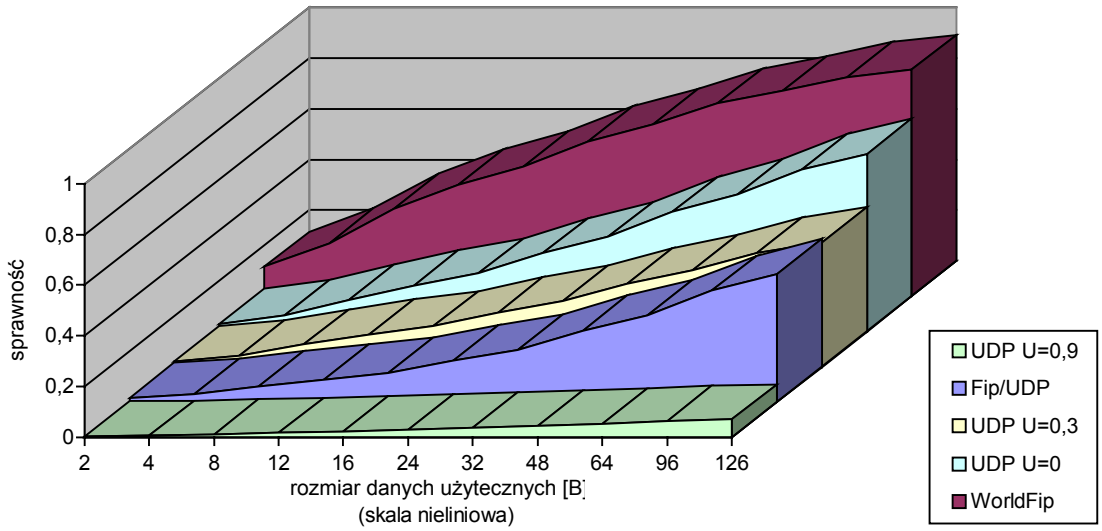
$$P = P_O(1-U) \text{[b/s]}. \qquad (97)$$

■

Wyniki dla obciążenia zerowego można traktować jako praca na segmencie separowanym. Na poniższym rysunku pokazano sprawności protokołu WorldFIP, UDP oraz przypadek tunelowania. Nastąpiło zmniejszenie sprawności protokołu względem idealnego UDP, a także względem specjalizowanej sieci z protokołem WorldFIP. Wynika to ze wzrostu bitów serwisowych oraz wprowadzenia nadzoru wymian. Tunelowany protokół WorldFIP ma sprawność porównywalną z UDP pracującym na średnio obciążonej sieci (ok. $^1/_3$ przepływności). Istotne jest to, że dla odseparowanego segmentu sieci tak skonstruowany protokół ma właściwości deterministyczne, a wartość jego sprawności jest stała.

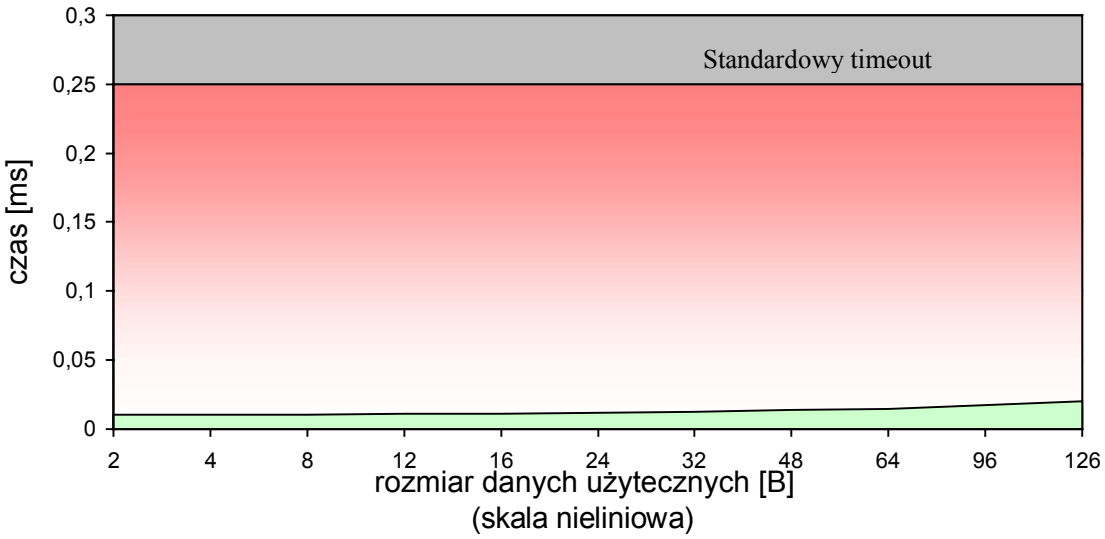
Dla takiego przypadku stosowanie mechanizmów tunelowania staje się celowe. Utrata efektywności jest niewielka natomiast uzyskuje się zdeterminowany w czasie cykl i deterministyczny dostęp do danych. Natomiast w przypadku systemów z obiegiem otwartym stosowanie tunelowania mija się z celem. Wszelkie parametry czasowe będą gorsze od samego UDP, a przy tym nie uzyska się ani niezawodności połączeń ani determinizmu dostępu.

Rzeczywista prędkość transmisji a co za tym idzie i przepustowość będzie zależała od zastosowanych interfejsów protokołu tunelowanego. Standardowe interfejsy sieci WorldFIP nie są w stanie przetwarzać pakietów szybciej niż wynika to ze standardowej prędkości transmisji.



Rys. 78 Porównanie sprawności protokołów WorldFIP, UDP oraz sprawności tunelowania transakcji cyklicznej WorldFIP/UDP w funkcji rozmiaru danych użytecznych

Poza omawianymi parametrami sprawności i przepustowości użytecznej, istotny jest również sam czas transmisji pakietu w tunelu. Dla abonentów ze standardowym interfejsem protokołu tunelowanego podłączanym przez urządzenie typu most czy brama, tunel powinien pozostać transparentny. Zatem w czasie przeznaczonym na reakcję sieci (timeout) powinien zmieścić się czas transmisji pakietu wraz z odpowiedzią i czasem kapsułkowania pakietów.



Rys. 79 Czas transakcji cyklicznej w funkcji rozmiaru pakietu w porównaniu z czasem timeoutu protokołu tunelowanego

Na rysunku 79 pokazano czas transmisji pakietów transakcji cyklicznej w porównaniu do standardowego czasu oczekiwania (timeoutu) dla protokołu WorldFIP. Czas ten stanowi średnio ok. 6% czasu oczekiwania, zatem umożliwia przeprowadzenie tunelowania pomniejszając jedynie w niewielkim stopniu założony margines czasu przeznaczonego na przetwarzanie

13.1.2. Transakcja aperiodyczna

Zgodnie z wnioskami z rozdziału 11.4 dla transakcji aperiodycznej, w warstwie transportowej brano pod uwagę protokół TCP. Czas transakcji aperiodycznej dla protokołu WorldFIP wynosi [61, 58, 114]:

$T_T = T_{TŻ} + T_{TŻID} + T_{ID} + T_{TD}$,

gdzie $T_{TŻID}$ jest czasem transmisji pakietu odpowiedzi z żądanymi identyfikatorami.
Czyli czas trwania transakcji aperiodycznej protokołu WorldFIP wynosi:

$$T_T = \frac{8(3+2)}{V} + \left(\frac{24+16k}{V}\right) + \frac{8(3+2)}{V} + \left(\frac{40+8n}{V}\right) \text{[s]}, \qquad (98)$$

gdzie k jest liczbą żądanych do transmisji aperiodycznej zmiennych. Przyjęto wartość k równą 1, co będzie oznaczało, że przesyła się jedną zmienną n bajtową. Otrzymano zatem:

$$T_T = \frac{40}{V} + \frac{40}{V} + \frac{40}{V} + \left(\frac{40+8n}{V}\right) \text{[s]}. \qquad (99)$$

Rozmiar danych datagramu IP dla zbioru rozmiarów testowych, jest zawsze większy od 38 bajtów, zarówno dla ramki zapytań jak i dla ramki danych. Dla każdej wymiany zachodzącej pomiędzy abonentami w transakcji aperiodycznej musi odbywać się nawiązanie połączenia w warstwie TCP, za wyjątkiem ustalenia żądanej zmiennej pomiędzy dystrybutorem a abonentem żądającym. Wynika to z faktu, iż nie ma gwarancji, że kolejne wymiany będą zachodziły pomiędzy tymi samymi abonentami. Stąd do czasu trwania każdej wymiany protokołu WorldFIP należy dołożyć czas nawiązywania połączenia wynikający z protokołu TCP.

$$T_N = \frac{4T_{ACK}}{V} = \frac{2048}{V} \text{[s]}. \qquad (100)$$

Zatem dla tunelowania przez TCP czas transakcji będzie wynosił:

$$T_T = \left(\frac{2048+512}{V}\right) + \frac{512}{V} + \left(\frac{2048+512}{V}\right) + \left(\frac{2048+568+8n}{V}\right) \text{[s]} \qquad \text{dla } n \geq 2 \text{ i } n < 126.$$

Zatem:

$$T_T = \frac{8(n+1031)}{V} \text{[s]},$$

Dla systemu zamkniętego sprawność użyteczna będzie wynosiła:

$$\eta_O = \frac{\frac{8n}{V}}{\frac{8(n+1031)}{V}} = \frac{n}{n+1031} \qquad \text{dla } n \geq 2 \text{ B}, \qquad (101)$$

a przepustowość użyteczna:

$$P_O = \frac{8n}{\frac{8(n+1031)}{V}} = \frac{nV}{n+1031} \text{ [b/s]} \qquad \text{dla } n \geq 2 \text{ B}. \qquad (102)$$

Uwzględniając opóźnienia, na podstawie zależności 59 i 63 sprawność użyteczna może być szacowana ze wzoru:

$$\eta = \eta_O(1-U), \tag{103}$$

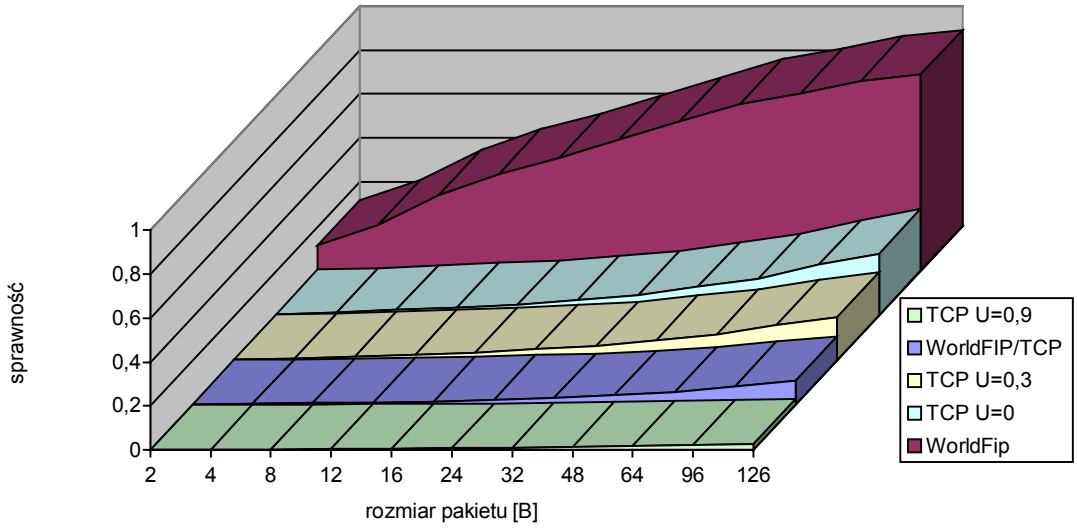
a przepustowość na podstawie zależności 60, 64 ze wzoru:

$$P = P_O(1-U)\,[\text{b/s}]. \tag{104}$$

Podobnie jak dla transakcji cyklicznej przepustowość rzeczywista zależy od prędkości pracy interfejsów protokołu WorldFIP.

■

Zgodnie z przewidywaniami występuje obniżenie sprawności. Ubytek sprawności jest znaczący i większy niż dla tunelowania wymiany cyklicznej w protokole UDP (rys. 80). Sprawność zbudowanego tunelu nie jest już porównywalna ze sprawnością protokołu nośnego na poziomie obciążenia 0,3. Ubytek ten nie wynika tylko ze złożoności protokołu TCP, lecz przede wszystkim z nałożenia wymian warstwy TCP oraz warstwy WorldFip. Każda wymiana z transakcji protokołu WorldFip pociąga za sobą pełną transakcję TCP.



Rys. 80 Porównanie sprawności protokołów transakcji aperiodycznej TCP, WorldFip oraz WorldFip/TCP

Istotną cechą takiego połączenia jest jego niezawodność. Dla informacji sterujących typu parametryzacja procesu czy polecenia obsługi należy stosować połączenia TCP mimo niższej wydajności tego protokołu.

Dla protokołu TCP implementacja rozgłaszania sprzętowego nie jest możliwa. Problemy z wykorzystaniem protokołu TCP opisane w rozdziale 12.2.2 mają również miejsce dla opisywanego przypadku. Implementacja tunelu wymian aperiodycznych przez protokół TCP jest kłopotliwa i spowoduje utratę istotnych cech tunelowanego protokołu. Następuje utrata przezroczystości tunelu powodując, iż tunelowanie pełnego protokołu WorldFIP, przy użyciu TCP, staje się nierealizowalne. Podobny problem występuje przy tunelowaniu aperiodycznym komunikatów. Dlatego dla systemów z zamkniętym lub separowanym obiegiem informacji

lepsze rozwiązanie stanowi stosowanie protokołu bezpołączeniowego UDP, również dla wymian aperiodycznych.

13.2. Współpraca systemu lokalnego i zdalnego

Aplikacyjne zarządzanie wymianami z punktu widzenia systemów z zamkniętą wymianą informacji daje uporządkowany ruch realizowany według danego scenariusza wymian. Trudniejszym przypadkiem do analizy są systemy z wymianami otwartymi. Pojawia się w nich obcy ruch oraz możliwość dowolnego kierowania ruchem pakietów (rozdział 7.2.1). Wówczas scenariusz wymian jest nadal realizowany, lecz bez zachowania determinizmu czasowego wymian.

Z punktu widzenia analizy ruchu pakietów, precyzyjne opisanie takiej sieci nie jest możliwe. Wynika to z dynamiki parametrów czasowych intersieci i braku mechanizmów rezerwacji zasobów. Opisanie efektywności połączenia jest możliwe tylko na zasadzie określania obciążenia sieci lub pomiarów empirycznych. Zależności, które dla sieci zamkniętych określały sprawność i przepustowość użyteczną, z uwzględnieniem obciążenia sieci mogą stanowić mechanizm szacowania optymistycznego dla systemów z obiegiem otwartym.

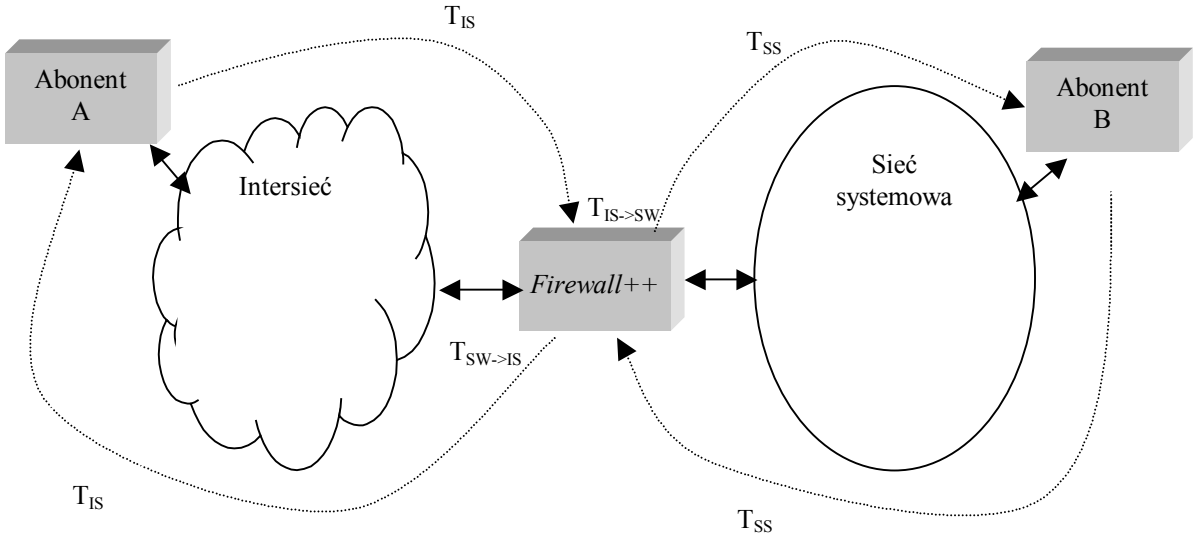
Ze względu na obcy niekontrolowany ruch w sieciach otwartych, bezcelowym jest stosowanie aplikacyjnej nadbudowy kontrolującej wymiany w celu uzyskania jakichkolwiek gwarancji czasowych. Zarówno w formie tunelowania protokołów deterministycznych jak i zarządzania warstwą transportową. Jakakolwiek metoda kontroli dostępu do medium jest nieprzydatna w chwili pojawienia się pakietów nie kontrolowanych przez tą metodę. Zatem w sieciach ze swobodnym ruchem pakietów jedynym rozsądnym zastosowaniem protokołów TCP/IP jest wykorzystywanie ich w postaci podstawowej bez tworzenia dodatkowych usług zarządzających ruchem.

Możliwe jest także stosowanie tunelowania w celu połączenia odległych sieci wykorzystujących protokół tunelowany w jedną, czyli wirtualne zespolenie dwóch lub więcej systemów lokalnych. Ponieważ intersieciowe połączenie wirtualne nie zapewnia detrminizmu wymian zespolenie takie będzie bez gwarancji czasowych. Z punktu widzenia protokołu tunelowanego tunel nie będzie widoczny, jednak jego obecność będzie wprowadzała zakłócenia powodujące przekraczanie czasów oczekiwania.

Jeżeli proces przemysłowy wymaga obsługi w czasie rzeczywistym, przy jednoczesnej możliwości komunikacji z intersiecią, wówczas konieczne jest stosowanie systemów z *Firewallem++* (rozdział 7.2.3). Analizując ruch informacji pomiędzy siecią wewnętrzną a intersiecią można doszukać się zależności czasowych. Należą do nich:

- opóźnienie interfejsów bramy,
- opóźnienie konwersji protokołów,
- obsługa informacji statusowych.

Występują wówczas zjawiska na sieci deterministycznej z jednej strony, intersieci z drugiej i urządzenia pośredniczącego (*Firewall++*) wnoszącego do układu opóźnienia dotyczące danych wymienianych pomiędzy sieciami.



Rys. 81 Przekazywanie informacji w systemie z *Firewallem++*

Na powyższym rysunku przedstawiono wymianę informacji pomiędzy abonentami A i B. Czas transmisji informacji od abonenta A do abonenta B będzie wynosił:

$$T_{A\to B} = T_{IS} + T_{IS\to SS} + T_{SS}, \tag{105}$$

natomiast w drugą stronę:

$$T_{B\to A} = T_{SS} + T_{SS\to IS} + T_{IS}, \tag{106}$$

gdzie:

$T_{A\to B}$ czas transmisji pakietu od abonenta A do abonenta B,

$T_{B\to A}$ czas transmisji pakietu od abonenta B do abonenta A,

T_{IS} czas transmisji pakietu w intersieci,

T_{SS} czas transmisji pakietu w sieci systemowej,

$T_{IS\to SS}$ czas pobytu pakietu zewnętrznego w bramie izolującej,

$T_{SS\to IS}$ czas pobytu pakietu wewnętrznego w bramie izolującej.

Czas T_{IS} oraz $T_{SS\to IS}$ są czasami zmiennymi z zakresu (T_{min}, ∞). Dlatego nie można posługiwać się konkretnymi wartościami a jedynie nieskończonym przedziałem wartości lub wartościami czasów średnich. Stąd:

$$\overline{T_{A\to B}} = \overline{T_{IS}} + T_{IS\to SS} + T_{SS}, \tag{107}$$

$$\overline{T_{B\to A}} = T_{SS} + \overline{T_{SS\to IS}} + \overline{T_{IS}}. \tag{108}$$

Czas transmisji w sieci systemowej T_{SS} wynika z cyklu pracy sieci i jest określony z skończonym przedziale. Czas przetwarzania pakietu przychodzącego z intersieci do bramy $T_{IS\to SS}$ jest stały i wynika z czasu działania warstw interfejsów komunikacyjnych oraz stabilnego cyklu sieci wewnętrznej. Zmienność czasu $T_{SS\to IS}$ wynika z faktu, iż dostarczenie pakietu do bramy oraz jego przetworzenie nie gwarantuje wysłania, gdyż czas dostępu do

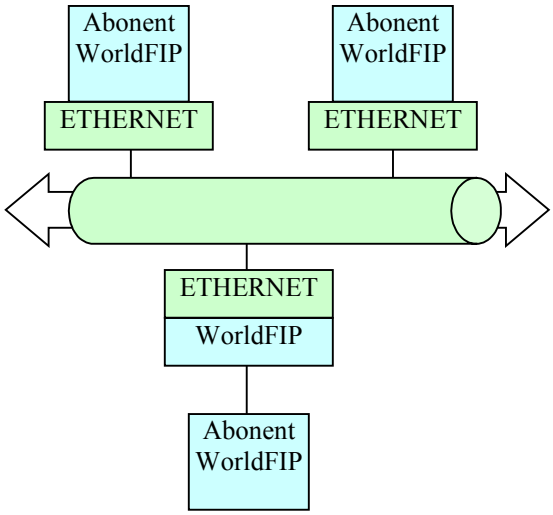
medium intersieciowego nie jest określony. Jeżeli średnie wielkości $T_{A\to B}$ oraz $T_{B\to A}$ nie mieszczą się w granicach dopuszczalnych jako czas oczekiwania na odpowiedź protokołu tunelowanego, wówczas zestawienie tunelu nie jest możliwe.

Generacja stempli czasowych w celu określania czasowej jakości przesyłanych danych powinna odbywać się w dowolnym momencie w czasie pobytu pakietów zewnętrznych w bramie oraz tuż przed transmisją dla pakietów pochodzących z sieci wewnętrznej.

13.3. Wnioski z analizy

Opisane w powyższym podrozdziale zależności mają miejsce dla tunelowania dowolnych protokołów deterministycznych. Warstwa transportowa UDP musi dołożyć swoje dane systemowe powodując w przypadku idealnym liniowe obniżenie sprawności protokołu tunelowanego. W przypadku warstwy transportowej TCP i warstwy nadrzędnej następuje współpraca przynajmniej dwóch algorytmów transakcyjnych. Dla każdej transakcji TCP pomiędzy danymi abonentami musi nastąpić nawiązanie połączenia i wymiana danych użytecznych. Ze względu na możliwość istnienia specyficznych wymagań czasowych warstwy nadrzędnej, realizacja takiego tunelu może być trudna lub wręcz niewykonalna. Do realizacji tunelowania w oparciu na protokole TCP należy bardzo starannie dobrać typ transakcji protokołu tunelowanego, zarówno od strony funkcjonalnej jak i dopasowania wzajemnego warstw. Gdy transakcje protokołu bazują na rozgłaszaniu komunikatów wykorzystanie TCP może okazać się niemożliwe.

Dobre rozwiązanie spełniające cel zwiększenia szybkości warstwy fizycznej stanowi zastosowanie tylko warstw sieci Ethernet bez stosu TCP/IP (rys. 82). Rozwiązanie takie zmniejsza znacząco narzuty czasowe od protokołów pośredniczących a jednocześnie umożliwi realizację tunelu dla dowolnego protokołu w systemie lokalnym.



Rys. 82 Tunel oparty na sieci Ethernet

Tunelowanie protokołów deterministycznych do systemów zdalnych nie gwarantuje parametrów sprawności i przepustowości, jak również sama intersieć nie obsługuje rozsyłania grupowego. Pewnym rozwiązaniem problemu rozsyłania może być stosowanie protokołu

IGMP [14], który obsługuje tunel sprzętowego rozsyłania grupowego pomiędzy węzłami intersieci.

Mechanizm tunelowania nie może być stosowany dla abonentów z interfejsami przystosowanymi tylko dla protokołu tunelowanego. W takim przypadku musi być stosowana brama konwertująca protokoły na zasadzie odcinania danych z warstw protokołów TCP/IP i przekazywania pakietów źródłowych do odpowiednich warstw interfejsu abonenta. Ponieważ nie zawsze jest to możliwe, pojawia się konieczność tworzenia „krótkich sieci" pomiędzy abonentem a bramą, w celu symulacji warstwy fizycznej. Poza tym narzut czasowy TCP/IP uniemożliwia dla wielu przypadków konwersję protokołów *on-line* z zachowaniem parametrów czasowych wymaganych przez standard sieci. Zmienność czasowa cyklu pracy w intersieciach powoduje utratę właściwości deterministycznych danego protokołu. Dlatego najlepszym rozwiązaniem dla przypadku tunelowania jest stosowanie urządzeń posiadających interfejsy komunikacyjne przystosowane do obsługi danego tunelu.

Tunelowanie może być podyktowane dwoma powodami. Po pierwsze, gdy istnieje potrzeba zapewnienia gwarantowanego czasu dostępu do informacji. Aby tunelowany protokół działał sprawnie z tego punktu widzenia, musi on pracować w systemie separowanym. Jakikolwiek obcy ruch znajdujący się poza kontrolą warstw aplikacji systemu spowoduje uaktywnienie dostępu rywalizacyjnego i utratę parametrów czasowych sieci. Drugim powodem tunelowania może być konieczność przetransmitowania danych przez intersieć bez konwertowania protokołu. Jeżeli intersieć zapewnia stabilną transmisję z parametrami odpowiednimi dla tunelowanego protokołu, wówczas intersieć dla systemu przemysłowego może być przeźroczysta. Obecny IP nie zapewnia jednak stabilnego transferu i rozwiązanie takie, pomimo, że w większości przypadków będzie działało poprawnie, należy traktować jako nie gwarantujące dostępu do danych w czasie rzeczywistym. Mechanizmem polepszającym transfer w intersieci przez minimalizację opóźnień jest QoS (rozdział 9.1.2), dlatego jego użycie w takich przypadkach jest wysoce wskazane.

14. Określenie zakresu stosowalności protokołu TCP/IP

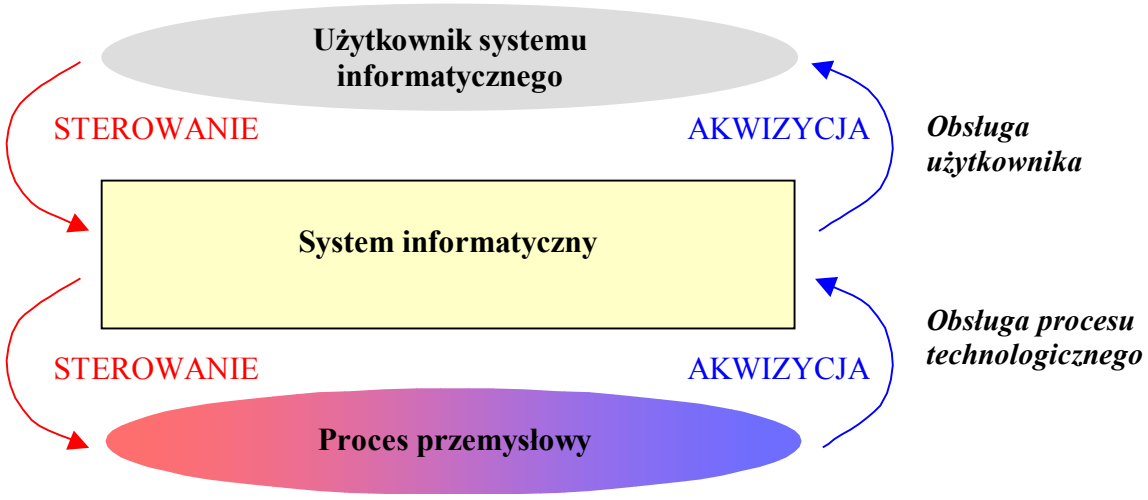
Głównym zadaniem informatycznego systemu kontrolno nadzorczego jest obsługa informacji związanej z procesem przemysłowym. Stosowalność protokołu komunikacyjnego wykorzystywanego w takim systemie należy rozważyć z punktu widzenia obsługi informacji wykonywanej przez system. Istnieją dwa typy zewnętrznych zadań wykonywanych przez system w ramach tej obsługi:

- akwizycja,
- sterowanie.

Akwizycja danych w kontekście sieci komputerowej wiąże się z pobraniem przez obiekt danych z procesu technologicznego lub od użytkownika i produkcją (rozdz. 4.1) zmiennej sieciowej zawierającej te dane. Sterowanie stanowi proces odwrotny, a mianowicie dane po przetworzeniu przez system informatyczny zostają przesłane przez sieć do konsumenta (rozdz. 4.1), który następnie oddziałuje na proces technologiczny lub użytkownika.

Obsługa informacji jest wykonywana przez system informatyczny na rzecz procesu technologicznego lub użytkownika systemu. Z punktu widzenia interakcji systemu informatycznego z otoczeniem system spełnia dwie role (rysunek 84):

1. obsługa użytkownika.
2. obsługa procesu technologicznego,



Rys. 83 Podstawowe role przemysłowego systemu informatycznego

Zadania akwizycji i sterowania mają miejsce zarówno dla obsługi procesu przemysłowego jak i dla obsługi użytkownika, jednak charakter tych zadań jest różny. W celu sprecyzowania zadań należy rozpatrzyć przepływ informacji pomiędzy użytkownikiem, systemem informatycznym i procesem technologicznym.

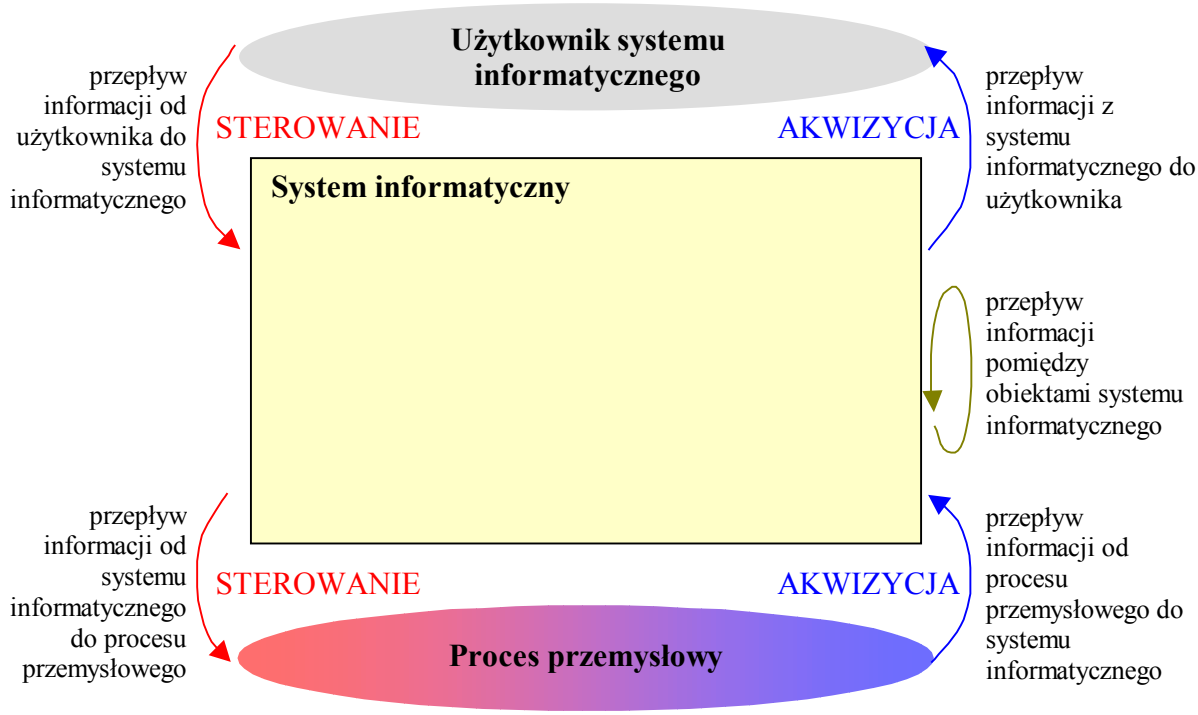
14.1. Przepływ informacji

Producenci i konsumenci przy użyciu sieci komputerowej przekazują między sobą dane użyteczne. W skład danych użytecznych oprócz zmiennych aplikacyjnych reprezentujących informację opisującą proces technologiczny mogą wchodzić inne informacje stanowiące zmienne aplikacyjne systemu informatycznego, niezbędne do poprawnego funkcjonowania warstwy aplikacji. Wymiany zmiennych mają na celu zapewnienie zarówno obsługi procesu jak i użytkownika.

Z punktu widzenia przepływu danych można wyodrębnić trzy obiegi informacji:

1. przepływ proces przemysłowy ↔ system informatyczny,
2. przepływ obiekt systemu ↔ obiekt systemu,
3. przepływ system informatyczny ↔ użytkownik.

Na rysunku 84 przedstawiono interakcję systemu informatycznego z procesem i z użytkownikiem oraz zaznaczono występujące obiegi informacji.



Rys. 84 Przepływ danych w systemie informatycznym

Zgrupowanie abonentów obsługujących proces i abonentów obsługujących użytkownika ma charakter symboliczny, gdyż nie istnieje potrzeba tworzenia dedykowanych obiektów do roli, jaką obiekt ma odegrać względem otoczenia. Podział taki jednak nastąpi względem rodzajów realizowanych wymian.

Zgodnie z opisem zamieszczonym w rozdziale 5 wymiany poziome realizują wymianę informacji pomiędzy obiektami systemu informatycznego a pionowe służą do transmisji danych na wyższe poziomy w hierarchii, w szczególności do transmisji danych do użytkownika Obsługa wewnętrzna informacji zaznaczona na rysunku 84 nie dotyczy bezpośrednio wymiany danych z otoczeniem. Reprezentuje ona sieciowe wymiany poziome i pionowe realizowane w warstwach komunikacyjnych systemu informatycznego.

Ponownie można wyodrębnić dwa typy obsługi informacji w informatycznym systemie kontrolno nadzorczym, uwzględniające zagadnienia przepływu danych:

1. obsługa systemowa
   - obsługa wymiany informacji z procesem realizowana lokalnie przez obiekty
     - akwizycja
     - sterowanie
   - wewnętrzna obsługa informacji wymienianej pomiędzy obiektami realizowana przez warstwy komunikacyjne systemu
     - wymiany poziome
2. obsługa funkcjonalna
   - obsługa informacji wymienianej z użytkownikiem realizowana lokalnie przez obiekty
     - akwizycja
     - sterowanie
   - wewnętrzna obsługa informacji wymienianej ze stacjami nadrzędnymi realizowana przez warstwy komunikacyjne systemu
     - wymiany pionowe

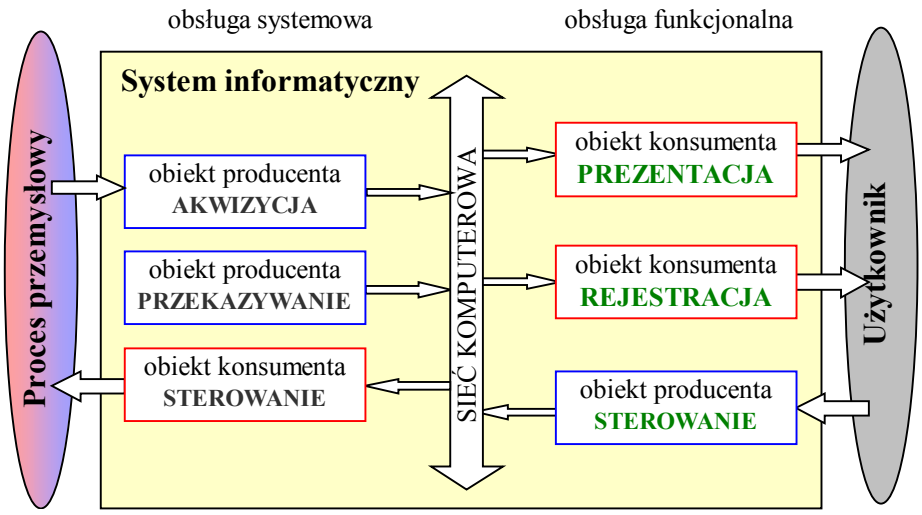
Na określenie wymagań dotyczących protokołu komunikacyjnego ma wpływ zarówno systemowa jak i funkcjonalna obsługa informacji mająca miejsce w danym systemie kontrolnym. Rozważanie dotyczące określenia zakresu stosowalności protokołu TCP/IP w sieci komputerowej wykorzystywanej do wymiany danych między obiektami sprowadza się do określenia czy protokół TCP/IP może być stosowany do realizacji systemowej obsługi informacji w kontekście funkcjonalnej obsługi informacji.

14.2. Obsługa funkcjonalna

Proces lokalnego przetwarzania informacji użytecznej przez obiekty stanowi funkcjonalną obsługę informacji. Obsługa funkcjonalna informacji przekazywanej siecią wiąże się z funkcjami wykonywanymi przez poszczególnych abonentów na rzecz dostarczanych danych. Obsługa funkcjonalna polega na przekazywaniu informacji użytkownikowi systemu i/lub wprowadzaniu informacji do systemu informatycznego przez użytkownika oraz przekazywanie danych użytecznych z/do obiektów nadrzędnych stanowiących interfejs użytkownika. Można wyróżnić trzy jej rodzaje:

1. prezentacja informacji – przedstawianie wartości zmiennych dla użytkownika,
2. rejestracja informacji – składowanie wartości zmiennych wraz ze znacznikami identyfikującymi kolejność i czas ich powstania,
3. sterowanie systemem informatycznym – przekazywanie informacji od użytkownika – zlecenie zapisu lub odczytu zmiennych z wytworzonymi lub przetworzonymi wartościami.

Na rysunku 85 przedstawiono zadania realizowane przez abonentów w ramach systemowej i funkcjonalnej obsługi informacji.



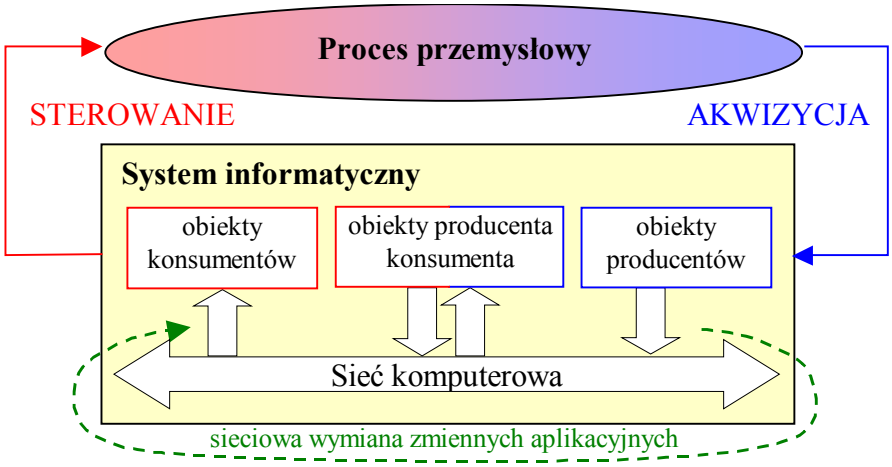
Rys. 85 Zadania realizowane w ramach systemowej i funkcjonalnej obsługi informacji

14.3. Obsługa systemowa

Systemową obsługę informacji stanowi pozyskiwanie danych i wypracowywanie informacji oddziałujących na proces przemysłowy oraz przekazywanie danych użytecznych pomiędzy obiektami systemu. Do obsługi systemowej należy zaliczyć:

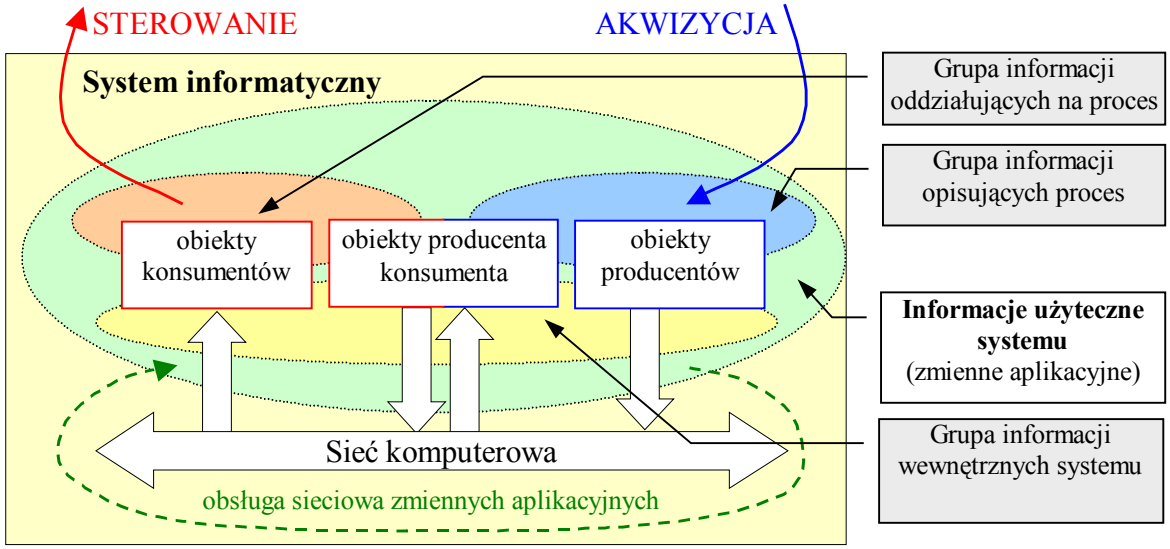
1. akwizycję danych procesowych – pozyskiwanie przez obiekty producenckie informacji opisujących proces,
2. sterowanie procesem – zwracanie przez obiekty konsumenckie informacji oddziałujących ma proces,
3. przekazywanie informacji – sieciowa wymiana zmiennych – wymiana informacji pomiędzy obiektami systemu informatycznego.

Na rysunku 86 przedstawiono obsługę informacji wykonywaną przez obiekty systemu klasyfikowane według podziału przedstawionego w rozdziale 4.1.



Rys. 86 Współpraca systemu informatycznego z procesem przemysłowym

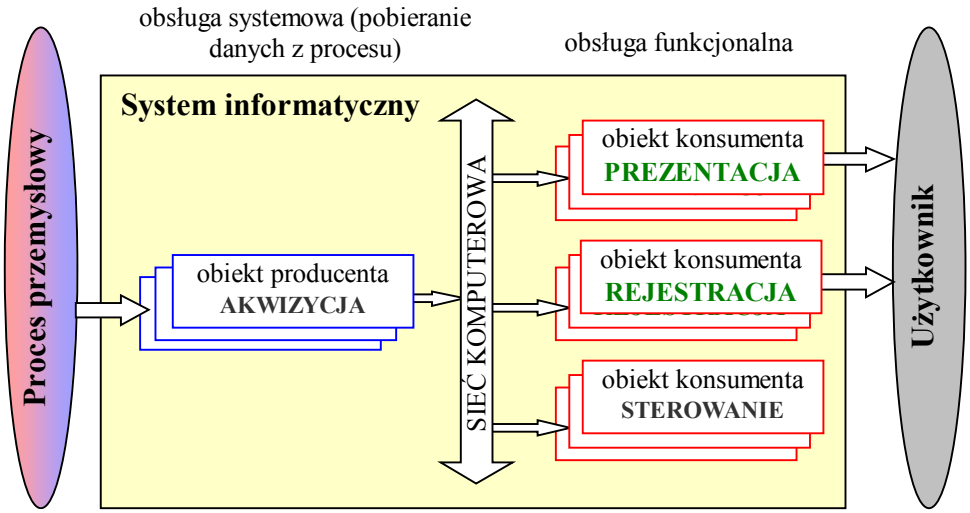
Na rysunku 87 przedstawiono grupy informacji uczestniczące w obsłudze systemowej. Informacje te stanowią podzbiór zbioru informacji użytecznych obsługiwanych przez system informatyczny.



Rys. 87 Grupy informacji obsługiwane systemowo przez przemysłowy system informatyczny

Ad. 1. Akwizycja danych procesowych

Akwizycja danych w obiekcie musi odbywać się w czasie rzeczywistym. Sieciowa obsługa tych danych zależy od celu prowadzenia tej akwizycji, czyli działań, jakie mają zostać wykonane na tych danych.

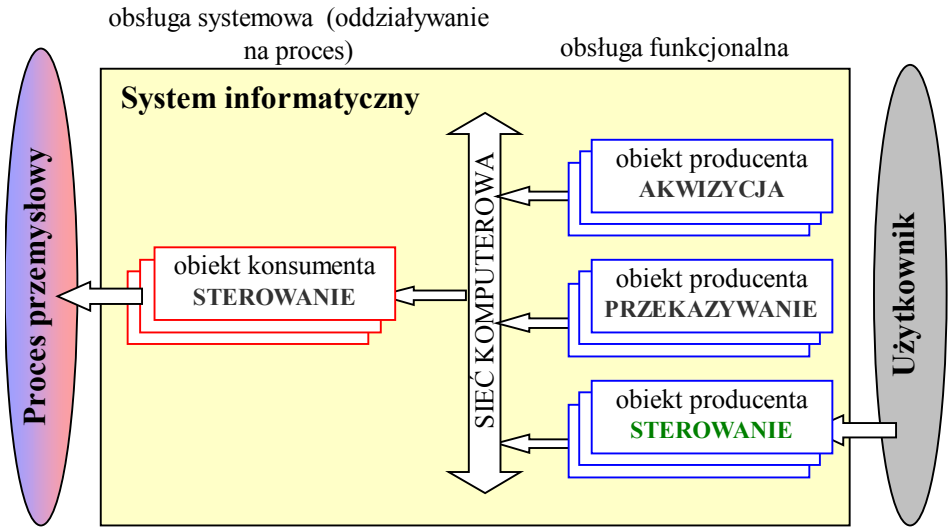


Rys. 88 Sieciowe przekazywanie danych poddawanych akwizycji

Ocena przydatności protokołu TCP/IP z punktu widzenia akwizycji danych procesowych musi być rozpatrywana względem aspektów obsługi funkcjonalnej oraz sterowania procesem. Zostało to przedstawione na rysunku 88.

Ad. 2. Sterowanie procesem

Dla każdego procesu technologicznego wypracowane oddziaływanie systemu informatycznego na ten proces musi być zrealizowane w czasie rzeczywistym. W przeciwnym wypadku istnienie systemu informatycznego byłoby bezcelowe. Dotyczy to zarówno samego obiektu jak i sieci, o ile sterowanie wykonywane przez obiekt zależy od wymian sieciowych.

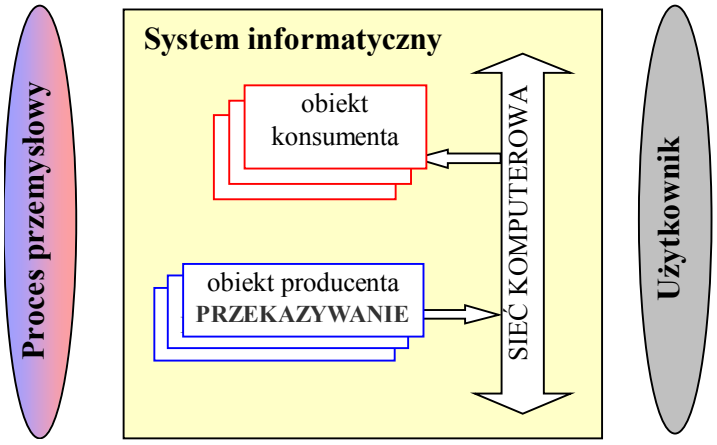


Rys. 89 Sieciowe przekazywanie danych służących sterowaniu

Ocenę przydatności protokołu do procesu sterowania należy rozważać dla danych pozyskiwanych z akwizycji i danych przekazywanych przez abonentów. Przedstawiono to na rysunku 89.

Ad. 3. Wymiana zmiennych aplikacyjnych systemu informatycznego

Zakładając, że przemysłowy system informatyczny składa się z więcej niż jednego obiektu i nie są to obiekty niezależne, można stwierdzić, iż stanowi on system rozproszony czasu rzeczywistego, w którym istnieje wspólna warstwa aplikacyjna podzielona pomiędzy te obiekty (rys. 30). Przekazywanie informacji pomiędzy obiektami pokazano na rysunku 90.

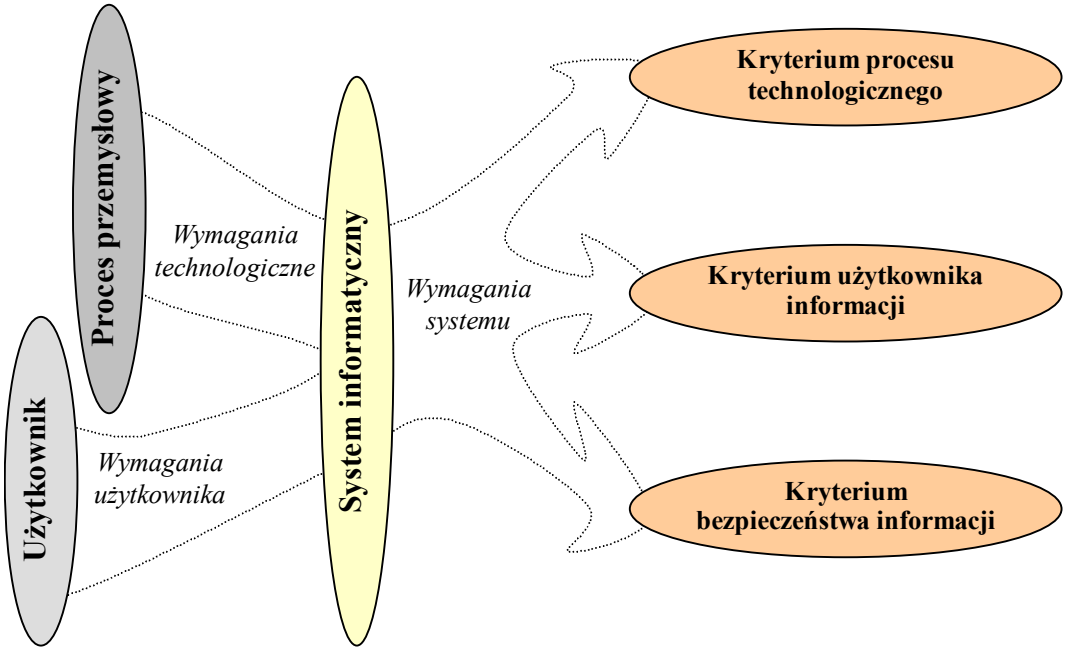


Rys. 90 Sieciowe wewnętrzne przekazywanie danych

Ocenę przydatności protokołu komunikacyjnego do procesu przekazywania zmiennych aplikacyjnych w takim systemie należy rozważać osobno względem każdego rodzaju wykorzystywanych wymian. Podstawowy podział wymian wynika z budowy hierarchicznej przedstawionej na rysunku 12 i 13 i dzieli sieciowe wymiany danych na wymiany poziome i pionowe. Dodatkowo należy rozpatrzyć aspekt cykliczności i acykliczności tych wymian.

14.4. Kryteria określania stosowalności protokołu

W celu zdefiniowania zakresu stosowalności protokołu TCP/IP, niezbędne jest określenie wymagań od strony technologicznej procesu przemysłowego obsługiwanego przez system kontrolno nadzorczy oraz od strony użytkownika tego systemu. Wymagania technologiczne wynikają z systemowej obsługi informacji natomiast wymagania użytkownika wynikają z funkcjonalnej obsługi informacji. Wymagania systemu informatycznego względem warstw komunikacyjnych zapewniających wymianę danych pomiędzy obiektami zależą zatem od obsługi systemowej i obsługi funkcjonalnej wykorzystywanej w danym systemie. Wymagania te mogą być charakteryzowane według trzech kryteriów przedstawionych na rysunku 91.



Rys. 91 Kryteria określania stosowalności protokołu

Pierwszym kryterium stosowalności protokołu jest kryterium wynikające z procesu technologicznego. Wewnętrzny obieg informacji w systemie lokalnym związany z cyklem sieci systemowej (rys. 20, 41) wprowadza podstawowe rozgraniczenie wymagań względem warstwy komunikacyjnej. Jest to obieg niezależny od użytkownika i niezbędny dla prawidłowego funkcjonowania kontroli procesu. Kluczowe zagadnienie stanowi określenie czy jest wymagany determinizm czasowy cyklu i względem jakich wymian. Z przekazywaniem zmiennych pomiędzy aplikacjami poszczególnych abonentów wiążą się płaszczyzny wymian informacji (rozdział 5). Obsługiwany przez wymiany poziome zbiór informacji jest zbiorem zmiennych aplikacyjnych systemu kontrolno – nadzorczego i jest zależny od stanu kontrolowanego procesu, a nie od zdarzeń generowanych przez użytkownika. Informacja wprowadzana do systemu przez użytkownika (wymiany pionowe: np. rozkazy, nastawy itp.) powoduje zmianę stanu jednego z obiektów (interfejsu użytkownika), a to może, ale nie musi pociągać za sobą zmiany wartości zmiennych aplikacyjnych ze zbioru zmiennych wymian poziomych. W systemie rozproszonym czasu rzeczywistego wymagania dotyczące wymian poziomych wiążą się z koniecznością

wymuszenia determinizmu czasowego dostępu do danej informacji. Wymiany pionowe w takich systemach mogą, lecz nie zawsze muszą być zdeterminowane czasowo. Zależy to od funkcji realizowanych przez stacje nadrzędne.

Drugie kryterium określania stosowalności protokołu stanowi kryterium obsługi informacji przez użytkownika. Określa ono sposób, w jaki następuje interakcja z użytkownikiem. Istnieją dwa główne aspekty wymiany danych z użytkownikiem. Pierwszy dotyczy zagadnień wyprowadzania danych systemowych dla użytkownika w celu przekazywania mu informacji o stanie procesu. Drugi dotyczy zagadnień wprowadzania danych do systemu i określania sposobu reakcji na takie dane.

Kontrola poprawności danych, wymogi czasowe technologii procesu jak również potencjalna konieczność ochrony informacji i kontroli dostępu do niej, wprowadza ostatnie kryterium stosowalności protokołu TCP/IP – kryterium bezpieczeństwa.

Poniższe podrozdziały zawierają rozważania na temat ograniczeń wynikających z poszczególnych kryteriów względem obsługi funkcjonalnej informacji, jaką ma zapewnić system informatyczny.

14.4.1. Ograniczenia wynikające z kryterium procesu

Informacje poddawaną akwizycji można wykorzystać na dwa sposoby:

1. przetworzyć i zapomnieć (wartości chwilowe),
2. zapamiętać i przetworzyć (wartości historyczne).

Poniżej rozważane zadania funkcjonalne odnoszą się do obu powyższych przypadków.

14.4.1.1. Prezentacja informacji

Prezentacja informacji użytkownikowi wiąże się głównie z funkcjonalnym zadaniem wizualizacji. Wizualizacja przemysłowa rozumiana jest jako działanie polegające na odzwierciedlaniu człowiekowi stanu procesu przemysłowego przy użyciu urządzeń i programów informatycznych [24, 23]. Z punktu widzenia wykorzystania danych można wyodrębnić dwa typy działań wizualizacyjnych:

- wizualizację w czasie rzeczywistym (ang. *on-line*),
- wizualizację po fakcie (ang. *off-line*).

Wizualizacja w czasie rzeczywistym jest działaniem wymagającym stałego napływu strumienia informacji opisującego ten proces. Natomiast wizualizacja po fakcie, która obrazuje historyczny przebieg zdarzeń w procesie, nie wymaga napływu informacji w czasie rzeczywistym, lecz nie powinna dopuszczać utraty danych. Wizualizacja po fakcie składa się z procesu rejestracji i procesu wizualizacji zarejestrowanych informacji, przy czym proces wizualizacji stanowi moduł analizy zarejestrowanych danych i powinien mieć możliwość sterowania czasem prezentacji.

Jeżeli wizualizacja ma przekazywać użytkownikowi bieżący stan procesu, to czas transmisji danych nie może być opóźniony powyżej pewnej granicy, gdyż operator wówczas nie będzie obserwował tego co się dzieje w procesie, lecz to co się już stało w procesie. O ile

wizualizacja nie jest związana z rejestracją, to odbiorcą prezentowanej informacji jest człowiek. Odczyt informacji przez człowieka nie jest zdeterminowany w czasie a percepcja użytkownika pozwala jedynie na rejestrację i analizę zdarzeń wolnozmiennych [57]. Wynika stąd, iż dostarczanie informacji powinno mieć charakter ciągły, lecz nie koniecznie zdeterminowany w czasie. Utrata pojedynczych danych nie jest niebezpieczna, gdyż sposób obsługi tej informacji z założenia zakłada możliwość jej utraty.

Można wykorzystywać protokół TCP/IP dla potrzeb wizualizacji pod warunkiem określenia czasu opóźnienia wnoszonego przez warstwę komunikacyjną i gwarancji utrzymywania jego wartości z danym prawdopodobieństwem. Dotyczy to każdego rodzaju systemu kontrolno nadzorczego (rozdziały 6.1, 7.2).

Bardzo użytecznym mechanizmem wspomagającym wizualizację, dla której dane dostarczane są w sposób niedeterministyczny jest mechanizm statusowy opisany w rozdziale 9.1.4.

14.4.1.2. Rejestracja informacji

Rejestracja danych wiąże się z szeregiem zadań funkcjonalnych wymagających dostępu do danych historycznych wraz lokalizacją zdarzeń w czasie oraz analizie i generacji informacji pochodnych. Można wyodrębnić dwa istotne przypadki związane z wykorzystaniem akwizycjonowanych danych:

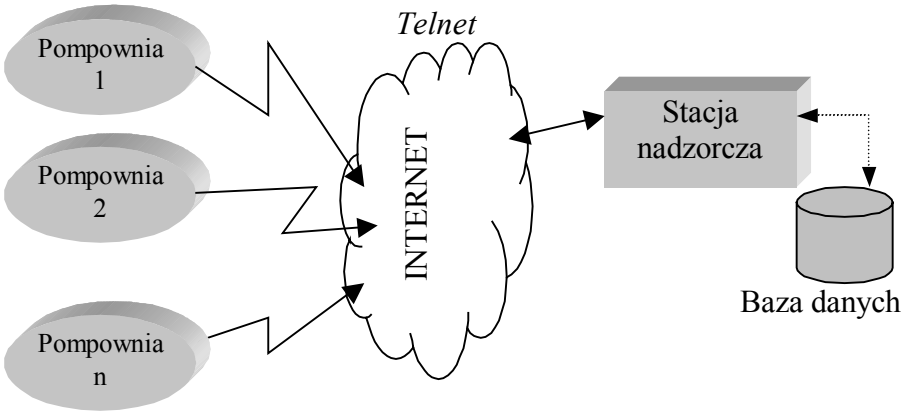
1. rejestracja danych celem sporadycznego wglądu w stan obiektu i dokładnej analizy po fakcie (ang. *off-line*),
2. rejestracja danych celem ciągłego wglądu w stan obiektu, wypracowywania informacji o nieprawidłowościach i dokładnej analizy w ruchu (ang. *on-line*).

Przypadek pierwszy wymaga mechanizmów dostarczających dane w sposób niezawodny, czyli w taki, aby żadna wartość wytransmitowana nie uległa zagubieniu. Dodatkowo każda wartość danej monitorowanej zmiennej powinna mieć swój stempel czasowy, określający czas jej wytworzenia lub przynajmniej określająca kolejność wytworzenia. Dostarczanie deterministyczne w czasie nie jest wymagane. Dane mogą przychodzić w sposób aperiodyczny a nawet w porządku nie chronologicznym, gdyż na ich podstawie nie generuje się informacji oddziałujących zwrotnie na proces.

Zakładając poprawną pracę sieci wykorzystującej protokół TCP/IP, nie ma przeszkód, aby transmisję danych na potrzeby rejestracji zrealizować o sieć systemu zamkniętego, separowanego lub otwartego.

Prostym przykładem systemu informatycznego, w którym zachodzi monitorowanie nie wymagające determinizmu wymian jest system zbierania informacji o stanie zasuw w przepompowniach ściekowych. Przepompownie są rozproszone terytorialnie, a każda z nich posiada lokalny system informatyczny, kontrolujący jej działanie (rys. 92). Stan wszystkich pompowni może być obserwowany na stanowisku nadzorczym wykorzystującym usługę Telnet (rozdział 10.2.3, 2) do pseudo cyklicznego i aperiodycznego przekazywania

parametrów. Pozyskiwane pomiary mogą być gromadzone w bazie danych i obrabiane w trybie *off-line*.



Rys. 92 Przykład monitorowania przepompowni w trybie *off-line*

Innym przykładem systemu z rejestracją *off-line*, gdzie znaczenie mają znaczniki czasowe, może być ciąg technologiczny produkcji papieru. Maszyna produkująca składa się z lokalnych sekcji i stanowisk nadzoru. W przypadku awarii niezmiernie istotne okazuje się precyzyjne prześledzenie sekwencji zdarzeń poprzedzających awarię w celu szybkiej diagnostyki tejże awarii. System informatyczny powinien zapewnić w takiej sytuacji niezakłóconą obserwowalność procesu technologicznego. Akwizycja danych z procesu powinna przebiegać na poziomie lokalnym obiektów w czasie rzeczywistym, natomiast transmisja sieciowa danych musi zapewnić niezawodność transmisji, choć niekoniecznie w czasie rzeczywistym.

Przypadek drugi – rejestracji *on-line*, ma wymaga determinizmu czasowego transmisji. Abonenci, których warstwy aplikacji wypracowują informację mogącą oddziaływać na proces muszą otrzymywać dane w sposób deterministyczny. Raportowanie *on-line*, jest możliwe tylko dla zdarzeń wolnozmiennych, ze względu na ograniczoną percepcję człowieka. Jednak dla systemów wymagających rejestracji danych w czasie rzeczywistym powinna ona być realizowalna niezależnie od charakteru występowania zdarzeń. Szczególnie duże ma to znaczenie w sytuacjach, gdy aplikacja na podstawie określonej paczki zarejestrowanych danych ma wygenerować informację dla użytkownika o nieprawidłowości przebiegu procesu lub wypracować informację sterującą. Wówczas system po pierwsze nie może dopuszczać utraty informacji, a po drugie nie mogą pojawiać się nieokreślone opóźnienia w transmisji. Dla takich zastosowań, protokół TCP/IP może być wykorzystany tylko w systemach zamkniętych i separowanych z zastosowaniem mechanizmów kontroli wymian.

Tak jak dla zagadnienia wizualizacji, dobrym narzędziem do określania jakości rejestracji jest opisany wcześniej mechanizm generacji informacji statusowych (por. rozdz. 9.1.4). Przykład znajduje się w załączniku A.2 na stronie 186 dodatków.

14.4.1.3. Przekazywanie informacji

W aspekcie przekazywania informacji zadaniem abonenta – producenta jest przetworzenie posiadanej informacji i zlecenie transmisji do konsumenta, który na jej podstawie wykona

funkcje z nim związane (rys. 88). Aspekt prezentacji i rejestracji został omówiony powyżej. Poniżej został omówiony aspekt funkcji sterowania.

Przy sterowaniu najistotniejszym staje się determinizm czasowy dostępu do informacji. Warstwa komunikacyjna systemu kontrolnego, nawet jeśli steruje zjawiskami wolnozmiennymi [57], musi dostarczać deterministycznie rozkaz od nadawcy do odbiorcy. Wiąże się to tylko z płaszczyznami wymian, na których odbywają się transmisje rozkazów sterujących, a nie z całym systemem komunikacyjnym. Konieczność realizacji zadań sterowania wymusza implementację mechanizmów zapewniających możliwość użycia wymian deterministycznych, lecz nie warunkuje budowania całej warstwy komunikacyjnej w oparciu o takie mechanizmy. Gdy system wymaga przesłania zmiennej sterującej, należy do tego celu użyć wymiany deterministycznej natomiast dla pozostałych zadań można wykorzystać inne możliwości systemu komunikacyjnego.

Człowiek nie jest w stanie sterować procesem w czasie rzeczywistym, zatem grupa rozkazów inicjowanych przez człowieka w poprawnie zaprojektowanym systemie kontrolnym nie może być traktowana jako deterministyczna. Natomiast istotne w tym przypadku staje się potwierdzanie wykonania rozkazu, aby użytkownik wiedział, że wydany przez niego rozkaz został przez system przetworzony. Poprawnie zaprojektowany system informatyczny w sytuacji wymagającej interwencji użytkownika ma prawo oczekiwać na jego reakcję, jednak zakładając brak determinizmu użytkownika, system powinien posiadać zabezpieczenia uwzględniające brak reakcji ze strony obsługi lub reakcje zbyt powolną.

Może zaistnieć przypadek, w którym z poziomu stacji typu SCADA będą wydawane rozkazy warunkujące poprawną pracę urządzeń procesowych lub rozkazy serwisowe. Jako przykład można posłużyć się systemem, w którym operator z poziomu konsoli zatrzymuje transport blachy w maszynie do cięcia. Gdyby system dopuszczał dowolnie długi czas na przesył rozkazu z konsoli do napędu, wówczas zatrzymanie transportu blachy w danym miejscu byłoby w większości przypadków nieprecyzyjne lub nierealizowalne. Alternatywnym rozwiązaniem jest zmiana koncepcji sterowania, dla przypadku, gdy wymagane jest zatrzymanie precyzyjne. Operator powinien wówczas zlecać do systemu rozkaz zatrzymania napędu, a system na podstawie zadanego parametru (np. liczby obrotów, pomiaru długości taśmy, wartości kątowej wału napędu) powinien zatrzymać napęd, gdy wartość ta zostanie osiągnięta. Zlecenie rozkazu stopu mogłoby być realizowane aperiodyczną wymianą pionową natomiast fizyczny rozkaz aperiodyczną deterministyczną wyminą poziomą (rozdział 5).

14.4.2. Ograniczenia wynikające z kryterium użytkownika

Informacja krążąca w systemie informatycznym może być przetwarzana automatycznie przez aplikacje abonentów, lub przetwarzana za pośrednictwem użytkownika. Ograniczenia komunikacyjne związane z obsługą systemowa są ściśle związane z charakterystyką wymian poziomych i zostały omówione przy okazji przepływu w podrozdziale poprzednim. Ograniczenia wynikające z obsługi informacji na poziomie prezentacji (por. rys. 16) związane

są tylko z tymi dziedzinami stosowania systemów kontrolnych, które dotyczą przekazywania informacji użytkownikowi i od użytkownika, czyli:

- prezentacją przebiegu procesu,
- parametryzacją stanu procesu.

Podczas przekazywania informacji pomiędzy maszyną a człowiekiem najistotniejszym staje się charakter zdarzeń związanych z tą informacją. Charakter samej informacji nie jest istotny z punktu widzenia jej obsługi, gdyż zawsze znajdzie się sposób na prezentację wielkości fizycznej w sposób zrozumiały dla człowieka, a poza tym jest to odrębne zagadnienie nie poruszane w niniejszej pracy [24, 23].

Zdarzenia zachodzące w systemach kontroli procesów przemysłowych mogą być wolnozmienne lub szybkozmienne [57]. Problem obsługi informacji po stronie użytkownika polega na tym, aby niezależnie od charakteru zdarzeń informacja była czytelna i aby obsługa nie zmieniała tego charakteru.

Pierwszy i kluczowy problem obsługi leży w konstrukcji samego abonenta. Warstwa aplikacji protokołu komunikacyjnego jest spleciona z oprogramowaniem spełniającym dla danego abonenta funkcję systemu operacyjnego. W powyższych rozważaniach zakładano, że systemy operacyjne abonentów są systemami czasu rzeczywistego. Dla abonentów pośredniczących w wymianie informacji pomiędzy użytkownikiem a systemem informatycznym nie zawsze tak musi być. Bardzo często system operacyjny współpracujący z użytkownikiem nie jest systemem czasu rzeczywistego lub potencjalne mechanizmy jego pracy w czasie rzeczywistym nie są wykorzystywane przez aplikację użytkownika. Zatem jeżeli system komunikacyjny dostarcza zdeterminowany w czasie strumień danych, a system operacyjny pod kontrolą którego pracuje interfejs użytkownika nie jest w stanie przetworzyć tego strumienia w czasie rzeczywistym, to następuje utrata determinizmu w warstwach niezależnych od warstw komunikacyjnych. Jeżeli warstwy obsługi informacji abonenta nie pracują w czasie rzeczywistym umożliwiającym synchronizację z cyklem pracy sieci, to nie ma sensu wykorzystywać wymian deterministycznych do dostarczania danych dla takiego systemu. Zatem wymiany pionowe do takich abonentów można realizować przy użyciu transakcji niedeterministycznych. Dostarczanie danych przy użyciu protokołu TCP/IP pracującego na płaszczyźnie wymian pionowych dla abonentów z systemami operacyjnymi nie spełniającymi wymogów czasu rzeczywistego jest poprawne i nie ograniczy w sposób istotny funkcji interfejsu użytkownika. Przypadek taki ma miejsce dla większości popularnych systemów operacyjnych, stacji SCADA, wizualizacji i monitorowania przez usługi Internetu. Przetwarzanie informacji w warstwach oprogramowania takich abonentów można próbować zoptymalizować [5, 38, 54, 20], jednak nie zawsze otrzymuje się system czasu rzeczywistego.

Drugim problemem jest sam człowiek. Jeżeli percepcja użytkownika nie jest doskonała, to użycie sposobów dostarczania informacji, których niezawodność określana jest danym prawdopodobieństwem jest dopuszczalna. Rozważania nad tym problemem nabierają

szczególnego znaczenia wtedy, gdy warstwy interfejsowe pomiędzy systemem kontroli a użytkownikiem mogą pracować w czasie rzeczywistym. Mamy wówczas do czynienia z urządzeniami, których konstrukcja sprzętowa i programowa pozwala na przetwarzanie informacji od warstwy fizycznej interfejsu komunikacyjnego do aplikacyjnej warstwy prezentacyjnej w zdefiniowanym przedziale czasu. Mogą to być komputery z systemami operacyjnymi czasu rzeczywistego jak np. QNX, RTLinux lub VxWorks [90, 121, 87, 149] czy też urządzenia specjalizowane jak tablice synoptyczne lub wszelkiego rodzaju pulpity i konsole. Mimo, iż urządzenia te przekazują często bardzo istotne informacje, to charakter obsługi powoduje, że wystarczającym jest zagwarantowanie dostawy danych z systemu komunikacyjnego dostarczającego z zadanym poziomem ufności.

Kolejnym problemem związanym z jednej strony z prezentacją a z drugiej ze sterowaniem jest implementacja interakcji użytkownika z systemem. Systemy typu SCADA służą między innymi do zadawania parametrów i rozkazów od użytkownika. Zagadnienia sterowania były poruszane w skali całego systemu w poprzednim rozdziale. Jednak charakterystyka obsługi szerokiego spektrum różnego typu informacji przekazywanych przez użytkownika do systemu kontrolnego różni się od obsługi informacji nie podlegającej bezpośredniej interakcji z użytkownikiem. Przede wszystkim obsługa zmiennych wewnętrznych systemu nie zależy od użytkownika, lecz od systemu i wymogów technologicznych. Obsługa zmiennych wchodzących w interakcję z użytkownikiem wymusza natomiast takie działania, aby użytkownik przekazując dane do systemu miał możliwość otrzymania potwierdzenia przyjęcia przez system przekazywanych danych. Opóźnienia dostarczania informacji sięgające nawet kilkukrotności czasu percepcji niczego w procesie nie zaburzą, gdyż z założenia operacje te muszą być zaprojektowane jako aperiodyczne. Dotyczy to zarówno zmiennych odpowiedzialnych za zmianę wszelkiego typu nastaw i parametrów jak i rozkazów oraz poleceń. Sterowaniem urządzeń zajmują się rozproszone warstwy aplikacyjne abonentów lub programy urządzeń przetwarzających typu PLC a nie użytkownik. Użytkownik z poziomu swojego interfejsu może decydować o kluczowych sprawach pracy systemu, jednak wszelkie rozkazy wymagające synchronizacji zdarzeń w czasie muszą być generowane automatycznie. Użytkownik może jedynie zainicjować proces.

Z punktu widzenia obsługi informacji przez użytkownika nie ma przeciwwskazań do stosowania protokołu TCP/IP. Dyskutowalny w skali danego systemu informatycznego jest jedynie obszar jego działania. Intersieć wprowadza ograniczenia i zagrożenia omawiane wcześniej, dlatego nie każdy system można oprzeć na otwartym charakterze tejże sieci. Struktura systemów zamkniętych oraz separowanych wraz z protokołem TCP/IP nie ogranicza możliwości obsługi i może być bezpiecznie stosowana.

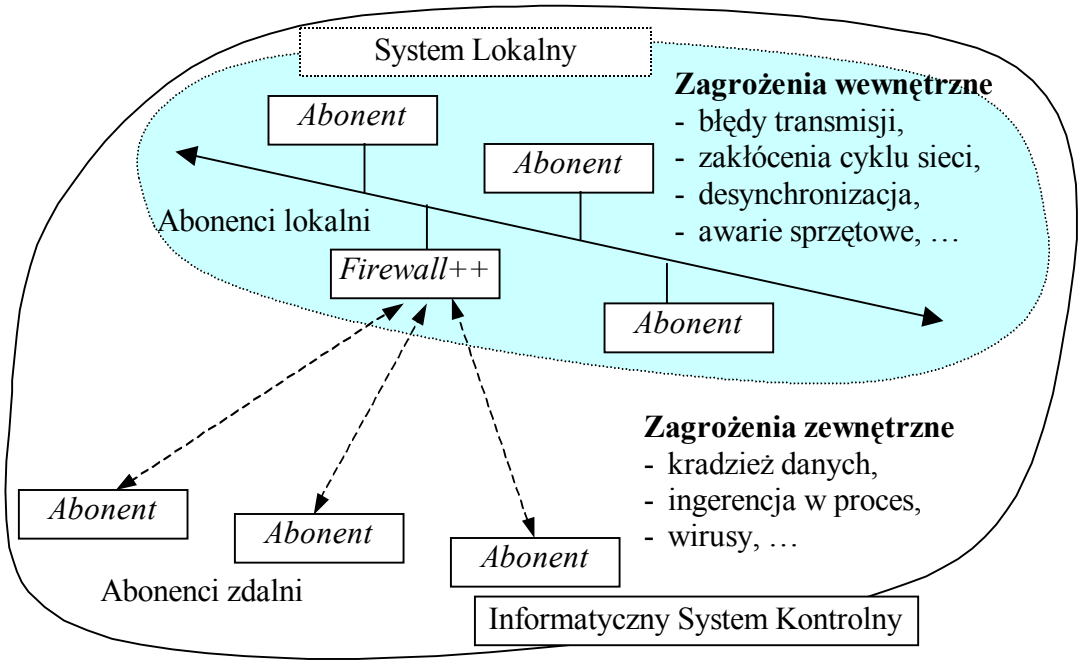
14.4.3. Ograniczenia wynikające z kryterium bezpieczeństwa

Ograniczenia stosowalności protokołu TCP/IP wynikające z bezpieczeństwa przetwarzanej informacji, a co za tym idzie i całego procesu technologicznego, można podzielić na dwie kategorie ograniczeń wynikających z:

- ochrony wewnętrznej systemu informatycznego,
- ochrony zewnętrznej systemu informatycznego.

Do zagrożeń wewnętrznych systemu należy zaliczyć wszystkie niebezpieczeństwa wynikające z działania samego systemu, czyli związane z przekłamaniami danych, stabilnością cyklu dostarczania danych, niezawodnością sieci, awariami sprzętu, medium czy też synchronizowania aplikacji i zapewnienia spójności danych [70, 56].

Zagrożenia zewnętrzne sprowadzają się do zagadnień zdalnego dostępu i określania stopnia upublicznienia informacji procesowych. W kontekście podziału systemów kontrolno nadzorczych opisanego w rozdziale 6.1 zagrożenia te przedstawiono na rysunku 93.



Rys. 93 Zagrożenia bezpieczeństwa informacji

W systemach zamkniętych mechanizmy ochrony informacji wynikające z zagrożeń wewnętrznych nie muszą być większe niż w przypadku specjalizowanych sieci przemysłowych. Wszelkie zabezpieczenia typu preambuła czy suma kontrolna ramki są zapewniane przez odpowiednie warstwy TCP/IP oraz Ethernetu. Przepływ danych może być deterministyczny, jeśli wymaga tego proces. Sieć jest administrowana przez użytkownika a awaryjność sprzętu jest porównywalna do sprzętu wykorzystywanego przez moduły sieci specjalizowanych. Wszelkie usługi aplikacyjne możliwe są do zapewnienia przez odpowiednie warstwy aplikacji abonentów na takiej samej zasadzie jak dla sieci specjalizowanych. Nie ma zatem potrzeby stosowania dodatkowych zabezpieczeń.

W sieciach systemów separowanych niebezpieczeństwa wewnętrzne są takie same jak w systemach zamkniętych. Istotne stają się natomiast zagrożenia zewnętrzne a co za tym idzie konstrukcja bramy pośredniczącej (*firewall*++ rozdz. 8). Brama stanowi ogniwo narażone na ataki z zewnątrz. Powinna ona umożliwiać tunelowanie informacji użytecznej i nie umożliwiać dostępu użytkownikom nieupoważnionym. Historia i analiza rozwiązań tego typu dostępnych na rynku [17, 68] pokazuje, że w każdym abonencie pośredniczącym można

znaleźć luki w zabezpieczeniach. Jest to tylko kwestią czasu. Bardzo trudne lub wręcz niemożliwe będzie to dla nietypowej usługi i działającej w bardzo ograniczonym zakresie, takiej jak proponowany *firewall++*.

W systemach otwartych występują duże zagrożenia zarówno wewnętrzne jak i zewnętrzne. Stosowanie sieci otwartych jest dopuszczalne tylko dla zastosowań, gdzie dostęp sieciowy nie jest w stanie zagrozić poprawnemu przebiegowi kontrolowanego procesu.

Z punktu widzenia bezpieczeństwa informacji można stwierdzić, iż w sieciach zamkniętych stosowanie protokołów TCP/IP nie ma żadnych ograniczeń. Na poziomie systemu lokalnego mechanizmy bezpieczeństwa protokołu TCP/IP dają większy poziom zabezpieczenia wewnętrznego niż sieci specjalizowane. Jednak każde podłączenie sieci systemowej do intersieci, niezależnie od sposobu, stanowi zagrożenie. Głównym niebezpieczeństwem stosowania TCP/IP jest zdalny dostęp do abonentów lokalnych lub do samej sieci, zagrażający uszkodzeniem danych procesowych lub samego cyklu wymiany danych w systemie lokalnym. Zdalny dostęp do treści informacji poufnych jest bezpieczny.

14.5. Wnioski dotyczące zakresu stosowalności protokołu TCP/IP

Dziedziny stosowania stosu protokołów TCP/IP w systemach przemysłowych przedstawiono w sposób uproszczony jako tabelaryczne zestawienie zadań funkcjonalnych i proponowanych kryteriów. Zestawienie takie zawiera Tabela 1.

W tabeli stosowane są następujące symbole:

- ✓ rozwiązanie stosowalne bez ograniczeń,
- ✓? rozwiązanie stosowalne pod warunkiem zastosowania dodatkowych mechanizmów bezpieczeństwa,
- ✘ rozwiązanie niestosowalne,
- – nie dotyczy (jest nierealizowalne z przyczyn niezależnych od protokołu).

Kolumny zawierają kryteria stosowalności protokołu TCP/IP w systemach przemysłowych natomiast w wierszach zamieszczono podstawowe zadania funkcjonalne przemysłowych systemów kontroli z podziałem na proponowane podstawowe typy architektur sieci wykorzystujących ten protokół.

Interpretację należy rozpoczynać od określenia typu wymaganego systemu. Dla danego systemu należy określić wykorzystywaną funkcję wraz ze sposobem dostępu do danych. Tabela określa czy protokół TCP/IP nadaje się do zrealizowania danej funkcji w określonym typie systemu i z określonym rodzajem dostępu do danych zakładając spełnienie założonych kryteriów. Dane rozwiązanie jest realizowalne w danej dziedzinie funkcjonalnej wówczas, gdy spełnione jest każde z trzech przyjętych kryteriów. Warstwa komunikacyjna musi zagwarantować wymagany przepływ, umożliwić realizowalność współpracy z użytkownikiem oraz musi być bezpieczna.

Tabela 1

		Kryterium procesu											
		Przepływ deterministyczny						Przepływ niedeterministyczny					
		Wymiany poziome		Wymiany pionowe				Wymiany poziome		Wymiany pionowe			
		Kryterium użytkownika											
				Interfejs deterministyczny		Interfejs niedeterministyczny				Interfejs deterministyczny		Interfejs niedeterministyczny	
		Kryterium bezpieczeństwa											
	funkcja	wew.	zew.	wew.	zew.	wew.	zew.	wew.	zew.	wew.	zew.	wew.	zew.
System zamknięty	Przekazywanie	✓	—	✓	—	✓	—	✓	—	✓	—	✓	—
	Prezentacja	✓	—	✓	—	✓	—	✓	—	✓	—	✓	—
	Rejestracja	✓	—	✓	—	✓	—	✓	—	✓	—	✓	—
System separowany	Przekazywanie	✓	✗	✓	✗	✗	✗	✓	✓?	✓	✓?	✓	✓?
	Prezentacja	✓	—	✓	✗	✓	✗	✓	—	✓	✓?	✓	✓?
	Rejestracja	✓	—	✓	✗	✓	✗	✓	—	✓	✓?	✓	✓?
System otwarty	Przekazywanie	✗	✗	✗	✗	✗	✗	✓?	✓?	✓?	✓?	✓?	✓?
	Prezentacja	✗	✗	✗	✗	✗	✗	✓?	✓?	✓?	✓?	✓?	✓?
	Rejestracja	✗	✗	✗	✗	✗	✗	✓?	✓?	✓?	✓?	✓?	✓?

Z tabeli wynika, iż optymalnym rozwiązaniem jest stosowanie systemów separowanych. Architektura taka daje maksimum możliwości realizacji zadań funkcjonalnych systemów kontrolnych przy dość dobrym wykorzystaniu możliwości, jakie oferują protokoły TCP/IP. System zamknięty jest bezpieczny, lecz nie umożliwia realizacji szeregu funkcji związanych z dostępem zdalnym. System otwarty daje natomiast pełną swobodę w implementacji funkcji, lecz niestety tylko z zakresu działań niedeterministycznych.

15. Wnioski końcowe

Niniejsza praca potwierdza, iż trend stosowania protokołu TCP/IP dla obsługi systemów informatycznych wykorzystywanych w przemyśle jest słuszny. Aspekt ekonomiczny takich zastosowań jest niepodważalny i niezależny od zagadnień merytorycznych. Pokazano również, iż techniczna realizacja staje się możliwa przy odpowiedniej konstrukcji stosu protokołów interfejsu komunikacyjnego abonentów.

Analizy przedstawione w pracy dają odpowiedź jak należy postępować przy konstruowaniu systemu lokalnego oraz systemu zdalnego, aby rozwiązanie komunikacyjne można było oprzeć na sieci Ethernet i protokole TCP/IP.

Praca daje również wytyczne odnośnie zakresu stosowalności protokołu TCP/IP jego przydatności i ograniczeń w konstrukcji informatycznych systemów przemysłowych.

Pierwsza teza postawiona w pracy została w potwierdzona w rozdziale 6 i 8. Zastosowanie modułów kontrolujących wymiany w sposób nadrzędny z poziomu warstwy aplikacji interfejsu komunikacyjnego modelu warstwowego ISO/OSI wszystkich abonentów sieci Ethernet, stwarza możliwość pracy sieci komputerowej jako sieci przemysłowej z wymianami zdeterminowanymi czasowo. Moduł będzie działał poprawnie zarówno z wykorzystaniem jak i bez wykorzystania protokołu TCP/IP, przy założeniu, iż nie istnieją w stosie protokołów takie jego elementy, które generują wymiany w sposób autonomiczny z pominięciem warstwy nadrzędnej. Potwierdza to tezę drugą pracy.

Możliwość niezakłóconej współpracy cyklu deterministycznego z cyklem niedeterministycznym przez specjalny moduł pośredniczący zwany w pracy *firewall++* została udowodniona w rozdziale 7 i 8. Zastosowanie tego modułu w warstwie aplikacji abonenta pośredniczącego, umożliwia przekazywanie informacji użytecznej pomiędzy siecią systemową a intersiecią bez naruszania deterministycznego cyklu wymian sieci systemowej. Tym samym została potwierdzona teza trzecia pracy punkt a i b.

Moduł *firewall++* umożliwia również obsługę informacji statusowej określającą czas przekazania wartości zmiennej w stacji pośredniczącej oraz informację o zainicjowaniu tej wartości przez producenta. Zostało to pokazane w rozdziale 9 i stanowi dowód punktu c tezy trzeciej.

W pracy pokazano również, iż istnieje dziedzina zastosowań protokołu TCP/IP, gdzie transmisja nie musi mieć charakteru deterministycznego a wprowadza nowe wartości do zagadnień kontroli i nadzoru procesów przemysłowych. Zostało to pokazane w rozdziale 10. Potwierdza to tezę czwartą o istnieniu możliwości zastosowania niezdeterminowanych usług zdalnej prezentacji i rejestracji informacji obsługiwanej przez informatyczny system

kontrolno-nadzorczy. Protokół TCP/IP stwarza dodatkowe możliwości związane z tworzeniem nowej jakości interfejsów użytkownika. Dotyczą one głównie standardowych usług intersieciowych, których realizacja bez użycia protokołu TCP/IP jest niemożliwa lub ekonomicznie nieopłacalna.

Wykonane czasowe analizy porównawcze wykazały, iż wykorzystanie sieci Ethernet i protokołu TCP/IP wraz z nadbudową aplikacyjną kontrolującą wymiany spowoduje obniżenie sprawności użytecznej łącza oraz zwiększenie przepustowości użytecznej łącza względem analizowanych protokołów specjalizowanych. Analizy te zostały przeprowadzone w rozdziałach 11 i 12.

Wykorzystanie sieci komputerowych z protokołem TCP/IP dla potrzeb informatycznych systemów przemysłowych może stanowić rozwiązanie bezpieczne tylko dla określonej dziedziny zastosowań. Istnieją zastosowania, jak choćby zdalna parametryzacja procesu, dla których należy stosować dodatkowe mechanizmy bezpieczeństwa takie jak identyfikacja i autoryzacja użytkownika oraz szyfrowanie danych. Sterowanie procesem z poziomu sieci niedeterministycznej lub przez zdalny dostęp nie jest rozwiązaniem bezpiecznym. Sterowanie jest dopuszczalne, gdy informacja użyteczna obsługująca to sterowanie, jest obsługiwana przez system w sposób zdeterminowany czasowo. Oznacza to, że warstwa komunikacyjna systemu informatycznego musi w takich przypadkach dostarczać dane pomiędzy abonentami również w sposób zdeterminowany czasowo. Sterowanie systemem przez użytkownika nie musi mieć charakteru deterministycznego, jednak istotne jest, aby transakcje danych przekazywanych do systemu odbywały się przy użyciu transakcji niezawodnych, wymagających potwierdzeń i retransmisji.

Biorąc pod uwagę intensywny rozwój protokołu TCP/IP można przypuszczać, że znajdzie on coraz szersze zastosowanie również w informatycznych systemach przemysłowych. Obecnie szczególnie dotkliwie odczuwalny jest brak standardowej warstwy w stosie TCP/IP, która obsługiwałaby wymiany deterministyczne. Funkcje takie trzeba dobudowywać w warstwie aplikacji, co znacząco utrudnia implementację protokołu. Aktualnie grupy badawcze zajmujące się rozwojem i standaryzacją TCP/IP nie zajmują się aspektem wykorzystania protokołu w sieciach lokalnych. Jedynie firmy komercyjne produkujące sprzęt dla systemów przemysłowych starają się wprowadzać pewne modyfikacje do stosu w celu wykorzystania protokołu w sieciach systemowych. Rozwiązania te nie są jednak standaryzowane, a zatem działają w obrębie niewielkiej grupy urządzeń. Pomimo, iż prace rozwojowe stosu TCP/IP dotyczą głównie komunikacji intersieciowej, ze szczególnym uwzględnieniem zestawiania kanałów o gwarantowanej przepustowości, rynek oraz postępująca uniwersalizacja zastosowań powinna wymusić w najbliższym czasie opracowanie standardów lokalnej transmisji deterministycznej.

Dalsze prace związane z poruszanym tematem będą kontynuowane w zakresie tworzenia uniwersalnego narzędzia do tworzenia usług prezentacyjnych w środowisku intersieci

opisanego w rozdziale 10.2.1 oraz udoskonalania funkcji modułu *firewall++* włączając w to zagadnienia tunelowania opisane w rozdziale 13.

I. Spis ilustracji

II. Indeks istotnych nazw i pojęć

Spis zawiera odwołania do pierwszego oraz następnych znaczących wystąpień danego hasła w treści.

III. Definicje wykorzystywanych pojęć i terminów

Abonent	Urządzenie stanowiące element systemu informatycznego posiadające interfejs sieciowy, podłączone do sieci komputerowej i realizujące wymianę danych z innymi abonentami – obiektami systemu.
Akwizycja	Pobranie przez obiekt danych z procesu technologicznego lub od użytkownika i produkcja zmiennej sieciowej zawierającej te dane.
Dane użyteczne	Wszelkie dane transmitowane siecią komputerową w systemie informatycznym za wyjątkiem informacji dokładanej przez warstwy stosu protokołów komunikacyjnych.
Determinizm	Twierdzenie, iż wszystkie zjawiska podlegają nieuchronnym prawidłowością, i że każde zdarzenie jest jednoznaczne i w sposób konieczny wyznaczone przez ogół warunków, w jakich zachodzi.
Determinizm czasowy	Determinizm czasowy dostępu do informacji lub medium. Zgodne z działaniem deterministycznym reakcje obiektu na zdarzenia, w określonym i skończonym czasie. Występowanie determinizmu czasowego w danym protokole gwarantuje dostęp do informacji w skończonym i ściśle określonym czasie.
Determinizm działania obiektu	Determinizm działania obiektu polega na takiej jego pracy, aby każda reakcja na zdarzenia pojawiające się spoza obiektu była jednoznaczna w danych warunkach.
FIP	(ang. *Factory Instrumentation Protocol*) WorldFip. Nazwa protokołu sieci przemysłowej opartej o model PDC. Jest to protokół sieci polowej o niezwykle dużej sprawności użytecznej.
Informatyczny System Przemysłowy	Zbiór urządzeń, programów i protokołów komunikacyjnych umożliwiających prowadzenie procesu kontroli.
InTouch	System wizualizacyjny wykorzystywany w niniejszej pracy do testów.
Kontrola procesu technologicznego	Ogół metod akwizycji, przetwarzania i sterowania wykorzystywany do prowadzenia obsługi informatycznej danego procesu technologicznego.
Kronos	System wizualizacyjny wykorzystywany w niniejszej pracy do testów.
InTouch	System wizualizacyjny wykorzystywany w niniejszej pracy do testów.
Master-Slave	Model sieciowej wymiany danych. Jeden z modeli gwarantujących czas dostępu do informacji.
Modbus	Protokół typu Master – Slave wykorzystywany powszechnie w informatycznych systemach przemysłowych.
MTBF	ang. *Medium Time Between Fault* – Średni czas pracy bezawaryjnej danego urządzenia.
MTU	ang. *Minimum Transfer Unit* – Minimalna jednostka transmisji informacji w danej sieci.
N10	Protokół typu Token wykorzystywany w informatycznych systemach przemysłowych.
Obiekt	Element systemu informatycznego – abonent sieciowy.

Pakiet	Logicznie spójna i skończona paczka danych transmitowana siecią.
PDC	ang. *Producer Distributor Consumert*; PDK – Producent Dystrybutor Konsument. Model sieciowej wymiany danych. Jeden z modeli gwarantujących czas dostępu do informacji.
PDK	zob. PDC
Proces	Proces technologiczny, przemysłowy – zespół fizycznych zjawisk poddawanych kontroli. Proces informatyczny – przetwarzanie informacji stanowiące umowną całość.
Protokół	Całość zasad i reguł wymiany informacji w sieci komputerowej.
Przeglądarka	Oprogramowanie umożliwiające interpretacje dokumentów hypertekstowych.
Przekazywanie informacji	Transmisja informacji użytecznej pomiędzy abonentami systemu informatycznego.
Przepływność	Miara szybkości, z jaką można przesyłać dane w sieci w odniesieniu ramki. [b/s]
Przepustowość użyteczna	Miara szybkości, z jaką można przesyłać dane w sieci w odniesieniu do danych użytecznych. [b/s]
RFC	ang. *Request For Comment* – dokumenty, w których zawarto propozycje lub definicje standardów protokołów TCP/IP.
Serwer	Urządzenie udostępniające informacje innym urządzeniom.
Sieć komputerowa	Wszystkie warstwy interfejsów komunikacyjnych abonentów systemu wraz z medium.
Sprawność użyteczna	Sprawność sieci dotycząca transmisji danych użytecznych.
Sterowanie procesem	Wypracowywanie informacji i oddziaływanie za jej pomocą na proces przemysłowy.
Sterowanie systemem	Wprowadzanie do systemu informatycznego informacji dotyczącej sposobu jego działania.
Stos protokołów	Zestaw protokołów składający się na konstrukcję interfejsu komunikacyjnego abonentów.
Token	Żeton. Abstrakcyjna jednostka informacji przekazywana pomiędzy abonentami w celu przekazania im szczególnych uprawnień do aktywności w sieci.
Transakcja	Całość wymian pomiędzy abonentami składające się na sekwencję realizacji przesłania informacji użytecznej pomiędzy nadawcą a odbiorcą lub odbiorcami.
Wizualizacja	Wizualizacja przemysłowa. Proces polegający na odzwierciedlaniu człowiekowi stanu procesu przemysłowego przy użyciu urządzeń i programów informatycznych.
WorldFIP	zob. FIP
Wymiana	Pojedyncza transmisja pomiędzy nadawcą a odbiorcą lub odbiorcami informacji.
Żeton	zob. Token.

IV. Zestawienie wykorzystywanych oznaczeń

oznaczenie	*opis*
ω	uśredniony współczynnik występujących opóźnień reprezentujący ubytek efektywności idealnego przypadku transmisji w sieci
η	współczynnik sprawności użytecznej
η_O	współczynnik sprawności idealnej – bez opóźnień
D	opóźnienie bieżące sieci [s]
D_0	opóźnienie nieobciążonej sieci [s]
l_A	liczebność zbioru zmiennych aplikacyjnych systemu lokalnego
l_d	liczebność zbioru zmiennych produkowanych sieciowo przez wszystkich abonentów zdalnych
l_S	liczebność zbioru zmiennych sieciowych systemu lokalnego
L_S	liczebność zbioru zmiennych sieciowych w całym systemie kontrolno – nadzorczym
L_U	liczba bitów danych użytkowych w pojedynczej transakcji danego typu
L_{ZT}	liczba przesyłanych znaków w ramce danego typu wymiany
n	W zależności od kontekstu: – liczba bajtów danych użytecznych, – liczba abonentów lokalnych systemu.
P	W zależności od kontekstu: – przepustowość użyteczną wyrażoną w bitach na sekundę, [b/s] – pakiet
P_O	przepustowość idealna – bez opóźnień [b/s]
Q	miara jakości [s]
T_{ACK}	czas trwania wymiany segmentu potwierdzenia [s]
T_{AR}	czas analizy ramki przez odbiorców [s]
T_C	czas cyklu scenariusza wymian [s]
T_{CE}	czas cyklu elementarnego [s]
T_{DR}	czas detekcji ramki przez odbiorców [s]
T_G	czas graniczny reakcji na zdarzenie [s]

T_{ID}	czas transmisji pakietu identyfikatora [s]
T_{IS}	czas transmisji pakietu w intersieci [s]
T_N	czas trwania transakcji nawiązywania połączenia [s]
T_O	czas opóźnień [s]
T_{OS}	czas obsługi sieciowej [s]
T_P	W zależności od kontekstu: [s] – czas okresu obsługi zmiennej w sieci, [s] – czas transakcji pakietu, [s] – okres ważności zmiennej [s] – czas przygotowania odpowiedzi przez nadawców [s]
T_{PMAX}	maksymalny czas okresu obsługi zmiennej w sieci [s]
T_{PMIN}	minimalny czas okresu obsługi zmiennej w sieci [s]
T_R	czas reakcji na zdarzenie [s]
T_{SS}	czas transmisji pakietu w sieci systemowej [s]
T_{SYN}	czas trwania wymiany segmentu synchronizacyjnego [s]
T_{SYNACK}	czas trwania wymiany segmentu potwierdzenia segmentu synchronizacyjnego [s]
T_{SZ}	czas transmisji pakietu w uogólnionej sieci zewnętrznej [s]
T_T	całkowity czas pojedynczej transakcji danego typu uwzględniający bity serwisowe dołożone przez warstwy protokołu, a nie stanowiące informacji użytecznych [s]
T_{TD}	czas transmisji pakietu danych użytecznych [s]
T_{TO}	czas transmisji odpowiedzi odczytu [s]
T_{TP}	czas transmisji potwierdzenia zapisu [s]
T_{TR}	rzeczywisty czas transmisji [s]
T_{TZ}	czas transmisji odpowiedzi zapisu [s]
$T_{TŻ}$	czas transmisji żądania [s]
$T_{TŻID}$	czas transmisji pakietu odpowiedzi z żądanymi identyfikatorami [s]
T_U	czas transmisji danych użytkowych w pojedynczej transakcji danego typu [s]
$T_{UDP\ x}$	czasy transmisji poszczególnych wymian w transakcji UDP [s]
T_Z	czas wystąpienia inicjującego zdarzenia [s]
U	współczynnik bieżącego wykorzystania sieci względem danej przepływności
V	prędkość transmisji [b/s]
Z_A	zbiór zmiennych aplikacyjnych
Z_B	zbiór zmiennych sieciowych
Z_d	zbiór zmiennych produkowanych sieciowo przez wszystkich abonentów zdalnych
Z_G	zbiór zmiennych globalnych

V. Bibliografia

1. Ahmed E.: JScript .NET - programowanie. Biblia; Helion 2002
2. Berge J.: Fieldbuses for Process Control: Engineering, Operation and Maintenance; ISA, September 2001
3. Białas A.: Polityka bezpieczeństwa i problemy zarządzania bezpieczeństwem. Materiały konferencyjne BISK'2002: Tom 1. WPKJS, Gliwice 2002.
4. Białas A.: Techniki szyfrowania. WPKJS, Gliwice 2001
5. Bigewski Z.: Optymalizacja pracy sieci przemysłowych. ZN Pol. Śl. s. Informatyka z. 28, Gliwice 1995.
6. Bigewski Z., Gaj P.: Sieć typu krążący żeton ALSPA - N80. Materiały dydaktyczne do laboratorium „Sieci przemysłowe", Instytut Informatyki Pol. Śl. Gliwice 1997-2000.
7. Buczek G.: ASP. Kompendium programisty; Helion 2002
8. Campbell B., Darnell R.: Dynamic HTML. Helion, Gliwice 1998
9. Carcagno L., Dours D., Facca R., Sautet B., Distributed Hard-Real-Time Systems: from specification to Realisation, Repr. 13th IFAC Workshop on Distributed Computer Control Systems, Toulouse, pp. 49-54, 1995
10. Castro E.: XML for the World Wide Web; Peachpit Press; 1st edition October 2000
11. Colburn R.: CGI. Helion, Gliwice 1998
12. Colourois G. i inni: Systemy rozproszone. WNT Warszawa 2000
13. Comer D. E.: Sieci komputerowe i intersieci. WNT Warszawa 2001
14. Comer D.: Sieci komputerowe TCP/IP (tom 1). WNT, Warszawa 2001.
15. Comer D.: Sieci komputerowe TCP/IP (tom 2). WNT, Warszawa 2001.
16. Comer D.: Sieci komputerowe TCP/IP (tom 3). WNT, Warszawa 2001.
17. ComputerWorld; 1994-2003
18. Cupek R., Fojcik M.: Budowa modułów komunikacyjnych stacji nadzorczej z sieciami przemysłowymi. Zeszyty Naukowe Pol. Śląskiej, S.Informatyka z.32 Gliwice 1997
19. Cupek R., Kwiecień A.: Ocena przydatności protokołu TCP/IP dla sieci przemysłowych najniższego poziomu. Materiały konferencyjne SCR'01, AGH Kraków 2001.
20. Cupek R.: Koncepcja elastycznego szeregowania zadań w sterownikach PLC; Materiały Konferencyjne SCR'03 Gliwice 2003
21. Cupek R.: Metody hierarchizacji rozproszonych procesów przemysłowych. ZN Pol. Śl. s. Informatyka z. 28, Gliwice 1995.

22. Cupek R.: Protokół TCP/IP w systemach wizualizacji procesów przemysłowych. ZN Pol. Śl. s. Studia Informatica Vol. 22 Number 3, Gliwice 2001
23. Cupek R.: Wizualizacja rozproszonych procesów przemysłowych. Rozprawa doktorska, Instytut Informatyki Politechniki Śląskiej, Gliwice, 1998.
24. Cupek R.: Wizualizacja systemów automatycznego sterowania. Zeszyty Naukowe Pol. Ś l., z.23, Gliwice 1993.
25. Czachórski T.: Modele kolejkowe w ocenie efektywności sieci i systemów komputerowych. WPKJS, Gliwice 1999
26. Davoli R., Giachini L.–A.: Schedulability Checking of Data Flow Task in Hard–Real–Time Distributed Systems, Technical Report UBLCS–94–4, University of Bologna, 1994.
27. Domański A, Gaj P.: Wykorzystanie usługi QoS do transmisji danych w informatycznych systemach przemysłowych. Materiały Konferencyjne SCR'03, WPŚ Gliwice 2003
28. Domański A.: „Wybrane zagadnienia analizy protokołów komunikacyjnych", Rozprawa doktorska. Instytut Informatyki Politechniki Śląskiej, Gliwice 2003
29. Doroszewski W. (red.). Słownik Języka Polskiego. PAN, PWN, Warszawa 1967.
30. Ferguson, Paul, Huston.: Geoff. Quality of Service: Delivering QoS on the Internet and in Corporate Networks. New York: John Wiley & Sons, 1998.
31. Gaj J. Kwiecień A.: Bezpieczeństwo transmisji w sieciach przemysłowych. Materiały konferencyjne BISK'2002: Tom 2. WPKJS, Gliwice 2002.
32. Gaj P. i inni: Laboratorium sieci komputerowych. Praca zbiorowa pod redakcją A. Grzywaka, Wyd. Naukowe Pol. Śl. Gliwice 1999
33. Gaj P. Ober J.: Firewall++ do zastosowań w systemach przemysłowych. Studia Informatica Vol. 24 Nr. 3 (55), Gliwice 2003.
34. Gaj P. Ober J.: Problemy z wykorzystaniem sieci ETHERNET w aplikacjach przemysłowych. Studia Informatica Vol. 24 Nr. 3 (55), Gliwice 2003.
35. Gaj P.: Dobór protokołów dla interfejsów komunikacyjnych urządzeń współpracujących z przemysłowymi systemami kontrolno-nadzorczymi. ZN Pol. Śl. s. Studia Informatica Vol. 23 Number 3 (50), Gliwice 2002
36. Gaj P.: Określanie jakości informacji użytecznej przy stosowaniu protokołu TCP/IP w informatycznych systemach przemysłowych. Matriały Konferencyjne SCR'02, WPŚ Gliwice 2002.
37. Gaj P.: Polowa sieć przemysłowa typu FIP. Materiały dydaktyczne do laboratorium „Sieci przemysłowe", Instytut Informatyki Pol. Śl. Gliwice 1994-2002.
38. Gaj P.: Szybka sieć przemysłowa a system wizualizacji – problem interfejsu. ZN Pol. Śl. s. Informatyka z. 36, Gliwice 1999.
39. Garfinkel S. Spafford G: Bezpieczeństwo w Unixie i internecie. RM, Warszawa 1997

40. Ghosh i in.: A survey of Real-Time Operating Systems, Georgia Institute of Technology, Atlanta 1994.
41. Goodman D.: JavaScript. Księga eksperta; Helion 2000
42. Goodwill J.: Java Server Pages; Helion 2001
43. Graham I.: HTML Sourcebook. John Willey & Sons, Inc 1995
44. Grzywak: Bezpieczeństwo w sieciach rozproszonych, ZN Pol. Śl. s. Informatyka z. 24, Gliwice 1993.
45. Habraken J.: ABC sieci komputerowych; Helion 2002
46. Holzner S.: XML. Vademecum profesjonalisty, Helion 2001
47. Insam E.: TCP/IP Embedded Internet Applications; Newnes, September 2003
48. Kądziela Z.: Analiza metod optymalizacji makrocykli pracy sieci FIP i implementacja wybranych algorytmów w module kreatora opisu sieci dla systemu wizualizacyjnego KRONOS. Praca Dyplomowa, Politechnika Śląska, Wydział Automatyki, Elektroniki i Informatyki, Kierunek Informatyka, Gliwice 1998.
49. Kennedy R.: Burning up the wires. Info World, San Mateo January 21, 2002
50. Kwiecień A. Przemysłowe sieciowe systemy rozproszone czasu rzeczywistego. Cechy i wymagania. Studia Informatica Vol.21 No.1. Gliwice 2000
51. Kwiecień A., Bigewski Z., Cupek R., Fojcik M., Gaj P.: Dokumentacja kontrolerów sieci FIP po rewizji oraz sprawozdanie z uruchomienia i badań. Praca PC-2/ RAu-2/95. Gliwice 1996.
52. Kwiecień A., Bigewski Z., Cupek R., Fojcik M., Gaj P.: Projekt adaptacji oprogramowania narzędziowego z jego ewentualną rozbudową dla potrzeb sieci FIP. Projekt adaptacji oprogramowania komunikacyjnego dla kontrolera sieci FIP. Praca PC-2/ RAu-2/95. Gliwice 1996.
53. Kwiecień A., Bigewski Z., Mrówka Z.: Analiza czasu najgorszego przypadku w sieciach przemysłowych.. ZN Pol.Śl. s.Informatyka z.36, Gliwice 1999.
54. Kwiecień A., Gaj P., Mrówka Z.: O pewnej implementacji interfejsu sieci typu FIP. ZN Pol. Śl. s. Informatyka z. 34, Gliwice 1998.
55. Kwiecień A., Gaj P., Mrówka Z.: Optymalizacja wymian w sieci FIP. ZN Pol. Śl. s. Informatyka z. 32, Gliwice 1997.
56. Kwiecień A., Gaj P.: Bezpieczeństwo systemów komputerowych i telekomunikacyjnych. Bezpieczeństwo transmisji w sieciach przemysłowych. Sotel Chorzów 1999
57. Kwiecień A., Gaj P.: Dobór protokołów w sieciach przemysłowych. ZN Pol. Śl. s. Studia Informaitca z. 22, Gliwice 2001.
58. Kwiecień A., Gaj P.: Sieć FIP, wstęp do analizy czasowej. ZN Pol. Śl. s. Informatyka z. 28, Gliwice 1995.

59. Kwiecień A., Grzywak A., Gaj P.: Dokumentacja funkcjonalna, techniczna i uruchomieniowa karty sieciowej FIP. Praca PC-2/ RAu-2/95. Instytut Informatyki Pol. Śl. Gliwice 1996.
60. Kwiecień A.: Analiza przepływu informacji w komputerowych sieciach przemysłowych. WPKJS 2000.
61. Kwiecień A.: Analiza przepływu informacji w komputerowych sieciach przemysłowych. ZN Pol. Śl. s. Studia Informaitca z. 22, Gliwice 2002.
62. Lalani S., Chandak R.: Active X – Biblioteka programisty. Helion Gliwice 1998
63. Marshall P.: Industrial Ethernet: A Pocket Guide; ISA, May 2002
64. Materiały konferencyjne SCR 2000: Kwiecień A.: Poprawa parametrów pracy sieci przemysłowych z cyklicznymi transakcjami wymiany informacji. AGH Kraków, Kraków 2000
65. McLaughlin B.: Java i XML; Helion 2001
66. Michta E.: Planowanie obciążenia sieci PROFIBUS z występowaniem ograniczeń czasowych. Studia Informatica z. 22, Gliwice 2001
67. Miozga A.: Moduł interfejsu programowego pozwalającego na wymianę plików, danych i poleceń pomiędzy stacją kontrolno-nadzorczą sieci przemysłowej a komputerem klasy IBM-PC, z wykorzystaniem procedur RPC dla systemu Windows 95, praca magisterska, Instytut Informatyki Politechniki Śląskiej, Gliwice 1997.
68. NetWorld. Warszawa 2000-2002
69. Niederliński: Systemy komputerowe automatyki przemysłowej t.1 Sprzęt i oprogramowanie, Wydawnictwo Naukowo–Techniczne, Warszawa 1984.
70. Ober J.: Bezpieczeństwo informacji w przemysłowych systemach pomiarowych. Materiały konferencyjne BISK'2002: Tom 2. WPKJS, Gliwice 2002.
71. Payne Ch.: ASP.NET dla każdego; Helion 2002
72. Puchol C., Mok A.K.: The Integration of Control and Dataflow Structures in Distributed Hard Real–Time Systems. Proc. Of the Int. Workshop on parallel and Distributed Real–Time Systems, WPDRTS, 1994.
73. Rahkonen: Distributed industrial control systems – a critical review regarding openness. Control Eng. Practice, Vol. 3, No. 8, 1995.
74. Riley S., Breyer R.: Switched, Fast, and Gigabit Ethernet; Que, 3rd edition, January 1999
75. Romowicz W.: HTML i JavaScript; Helion 1998
76. Ruitz L., Raja P., Fischer N., Decotigne J.D.: Self Configuration Protocol for a Hard Real–Time Network. Repr. 13th IFAC Workshop on Distributed Computer Control Systems, Toulouse, 1995.
77. Siyan K., Parker T.: TCP/IP. Księga eksperta; wydanie II Helion 2002
78. Sportack M.: Networking Essentials Unleashed; SAMS, March 1998
79. Spurgeon Ch.: Ethernet: The Definitive Guide; O'Reilly & Associates; February 2000

80. Stuart A.: SCADA: Supervisory Control and Data Acquisition; ISA, 2nd Edition, January 1999
81. Świderski M.: Koncepcja obiektów samodzielnie komunikujących się z bazami danych, osadzonych w stronach WWW. ZN Pol. Śl. s. Studia Informatica Vol. 23 Number 2B (49), Gliwice 2002.
82. Szymczak M.(red.): Słownik Języka Polskiego. PWN Warszawa 1983
83. Szargut J.: Termodynamika. Wyd. 6 rozsz. Warszawa; PWN, 1998
84. Teague J. C.: Po prostu DHTML i CSS; Helion 2002
85. Tindell: Calculating controller area network (CAN) message response times, Control Eng. Practice 1995.
86. Tokarski J.(red.): Słownik Wyrazów Obcych. PWN Warszawa 1980
87. Tomana M.: System Unix. WPKJS Bielsko-Biała 2002
88. Torngren: Fundamentals of Implementing Real –Time Control Applications in Distributed Computer Systems, Journal of Real Time Systems 1996.
89. Tsai J.: Distributed Real-Time Systems: Monitoring, Visualization, Debugging, and Analysis; Wiley-Interscience, July 1996
90. Uresh V.: Jądro systemu UNIX. Nowe horyzonty; Wydawnictwa Naukowo-Techniczne, Listopad 2001
91. Webster S.: Flash i PHP. Podstawy; Helion 2002
92. Wiebe M.: A Guide to Utility Automation: Amr, Scada, and It Systems for Electric Power; Pennwell Pub; January 2000
93. Węgrzyn S.: Podstawy automatyki. Państwowe Wydawnictwo Naukowe, Warszawa 1980.
94. Williams R. Handbook of SCADA systems; Elsevier Advanced Technology; 1st edition 2001
95. Wojtulewicz M.: SuiteLink – protokół z przyszłością. Astor – Biuletyn Automatyki, Numer 16 (2/1998), Kraków 1998
96. Wolisz A.: Podstawy Sieci Komputerowych tom 1. WNT Warszawa 1990, 1992
97. Wolisz A.: Podstawy Sieci Komputerowych tom 2. WNT Warszawa 1990, 1992
98. Woods P.: Flash 5. Kompendium programisty; Helion 2002

DOKUMENTY RFC ORAZ DOKUMENTY I OPISY STANDARDÓW

99. Barkley J. Comparing Remote Procedure Calls, NIST, October 1993
100. Bollella Greg i inni, The Real-Time Specification for Java, ADDISON-WESLEY, 2000 USA
101. ISO Remote Procedure Call Specification. ISO/IEC CD 11578 N6561, ISO/IEC, November 1991.
102. RFC0768 – User Datagram Protocol

103. RFC0793 – Transmission Control Protocol
104. RFC0854 – Telnet Protocol Specification
105. RFC0879 – TCP maximum segment size and related topics
106. RFC1180 – TCP/IP tutorial
107. RFC2068 – Hypertext Transfer Protocol -- HTTP/1.1
108. RFC2212 – Specification of Guaranteed Quality of Service
109. RFC2454 – IP Version 6 Management Information Base for the User Datagram Protocol
110. RFC2459 – Internet X.509 Public Key Infrastructure Certificate and CRL Profile
111. RFC2616 – Hypertext Transfer Protocol -- HTTP/1.1
112. RFC2660 – The Secure HyperText Transfer Protocol

DOKUMENTACJA FIRMOWA

113. Borland C++ Builder 5, Inprise Corporation, Scotts Valley, CA USA, 2000
114. FIP NETWORK General Introduction; DPS 50249 aA, Clamart 1990.
115. FIP STARTER KIT materiały klubowe WorldFip; Clamart 1995.
116. FIPCODE Software v. 5.0 User Reference Manual; DPS 50263 b–en, Clamart 1994.
117. FipView v.3.5 Dokumentacja SDD, Proloc, Katowice 2000
118. Furr S.: What Is Real Time And Why Do I Need It?; QNX Software Systems Ltd. Canada 2002
119. Klubowe materiały szkoleniowe; WorldFip, Clamart 1995, 1996, 1997, 1998.
120. Kronos – Stacja Kontrolno Nadzorcza; Dokumentacja SDD; Proloc, Katowice 2001
121. Leroux P.: Real Time or Real Linux? A Realistic Alternative; QNX Software Systems Ltd. Canada 2002
122. Ludovic HERISSON. Modbus on Ethernet. applicomIO 2.0, Applicom 2002
123. Materiały firmowe Open MODBUS/TCP Specification. Schneider Electric, 1999
124. Microsoft Developer Network MSDN. Periodic CD-ROM. Microsoft Corp. 1999/2000.
125. Microsoft® Visual Basic®. Zintegrowane środowisko programowania. Microsoft Corporation. Version 5.0, 1997.
126. Network Coprocesor. Multiprotocol. Dokumentacja techniczna firmy Cegelec, Clamart Cedex France 1993
127. SCO-UNIX Driver. FIP Device Manager. Dokumentacja firmy Cegelec. Clamart Cedex France 1996.
128. Serial Communication modules for Alspa 8000 PLCs – User Manual; ALS52506 a-en, Cegelec, Clamart Cedex France 1993
129. Software Development Kit. Microsoft Corp. 1999.
130. Stacja pomp obiegowych. Elektrociepłownia „Poznań Garbary”. Dokumentacja wykonawcza i powykonawcza firmy „PROLOC” Katowice 1998.

131. Stacja uzdatniania wody. Elektrownia „TRZEBOWICE”. Dokumentacja wykonawcza i powykonawcza firmy „PROLOC” Katowice 1993.
132. TCP/IP Suite. Internetworking Technologies Handbook. Cisco Co. USA 1999
133. Visual Basic 5.0 Documentation. Component Tools Guide. Microsoft Corp. 1997.
134. Visual C++ 5.0 Documentation. Visual C++ Tutorials: Autoclick: Automation. Microsoft Corp. 1997.
135. Wonderware Reference Disk. USA California 2003

INTERNET

136. Sunbelt W2Knews Electronic Newsletter 2000-2003
137. www.10gigabit-ethernet.com: Technical Essence Webs
138. www.applicom-int.com: Open industrial communication concept
139. www.canopen.org: CAN in Automation
140. www.ietf.org: The Internet Engineering Task Force
141. www.interbusclub.com: Interbus Club Website
142. www.ipv6.org: IPv6 Information Page
143. www.isoc.org: Internet Society Website
144. www.linux-ipv6.org: Linux IPv6 Development Project
145. www.microsoft.com: Technology Centers: Internet Protocol Version 6; Microsoft 2003
146. www.modbus.org: Modbus Organization
147. www.openqnx.com: OpenQNX Pages
148. www.profibus.com: Profibus Website
149. www.qnx.com: QNX Software System Ltd Website
150. www.redhat.com: RedHat Linux Pages
151. www.rfc-editor.org: RFC Editor Website
152. www.worldfip.org: International WorldFIP Organization

VI. Załączniki

A. Praktyczne implementacje

1 PDC/UDP/IP

Dla celów testowych stworzono specjalną warstwę aplikacyjną dla interfejsu komunikacyjnego komputera klasy PC z systemem operacyjnym Windows 2000 Professional. Warstwa ta kontroluje wymiany cykliczne realizowane przez warstwę transportową interfejsu w oparciu o protokół UDP. Praktyczna implementacja warstwy została stworzona w postaci programów „Arbiter" oraz „Abonent" umożliwiających przesyłanie informacji w postaci definiowanych zmiennych.

Ze względu, iż model sieci PDC wymaga komunikacji dokonywanej przez rozgłaszanie, wykorzystano w warstwie transportowej protokół UDP, umożliwiający taki właśnie rodzaj transferu danych. Warstwa aplikacji abonentów wprowadza natomiast własny protokół zgodny z modelem kontroli wymian PDC. Aplikacje w wersji podstawowej, wykorzystywanej do testów, obsługują jedynie wymiany cykliczne zmiennych.

Wykorzystywane protokoły TCP/IP mają za zadanie zapewnić transfer zmiennych. Zmienna jest abstrakcyjnym tworem stanowiącym podstawowy nośnik informacji użytecznej. Każda zmienna jest identyfikowana przez swój unikalny numer. Na poziomie UDP każdy datagram wygląda tak samo. To co rozróżnia typ wymiany i zmiennej, to informacje zawarte w polu danych datagramu UDP. Zatem można powiedzieć, iż zmienne są w pewnym sensie tunelowane przez tunel utworzony z wykorzystaniem datagramów UDP. Ramka protokołu z punktu widzenia warstwy aplikacji przedstawiona jest na poniższym rysunku.

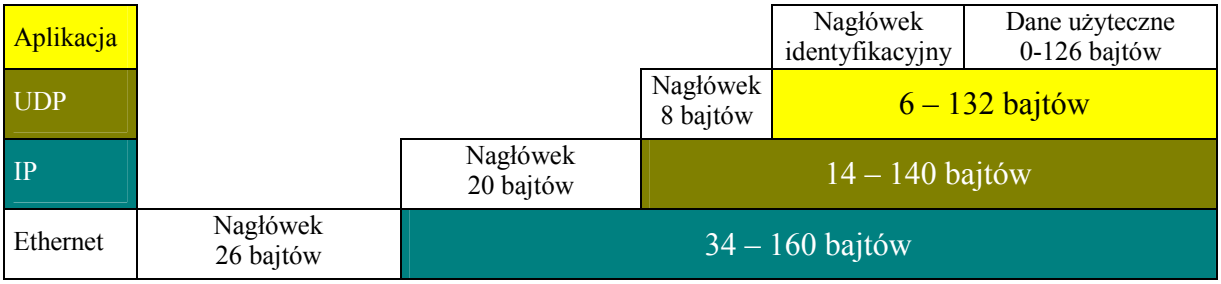
Nagłówek Identyfikacyjny				*Dane użyteczne*
Kod wymiany	Rozmiar danych użytecznych	Identyfikator zmiennej	Status zmiennej	Dane
1 bajt	2 bajty	2 bajty	1 bajt	0-126 bajtów

Rys. 94 Ramka aplikacyjna i jej nagłówek identyfikacyjny

Nagłówek zawiera pola informacyjne określające kolejno:

- typ ramki aplikacyjnej (1 – ramka zapytania od arbitra, 2 – ramka odpowiedzi od abonenta),
- rozmiar danych użytecznych przesyłanych w ramce podawany w bajtach,
- 16-bitowy identyfikator zmiennej określający przesyłane dane,
- bajt statusowy przenoszący informacje statusowe wytwarzane przez nadawcę (bit 0 – status inicjacji, bit 1 – status wytworzenia).

Z punktu widzenia całego stosu protokołów TCP/IP schemat przekazywania danych w stosie przedstawiono na rysunku 95.



Rys. 95 Przekazywanie danych w ramkach stosu TCP/IP

Aplikacja arbitra umożliwia:

- Stworzenie scenariusza wymian umożliwiającego realizację n transakcji z parametryzowanym czasem cyklu.
- Definiowanie zmiennych i powiązanie ich ze scenariuszem.
- Rozsyłanie zapytań o informację.
- Oczekiwanie na odpowiedź w skończonym czasie.

Aplikacja abonenta umożliwia:

- Definiowanie zmiennych i wiązanie ich z informacją użyteczną.
- Pracę zapisu i odczytu zmiennych według cyklu arbitra.
- Swobodną pracę zapisu i odczytu zmiennych bez arbitra.
- Obsługę statusów inicjacji, wytworzenia, dostarczenia i pobrania.
- Obserwację cyklu sieci oraz aktywności pakietów UDP.
- Współpracę z innymi aplikacjami przez interfejs DDE.

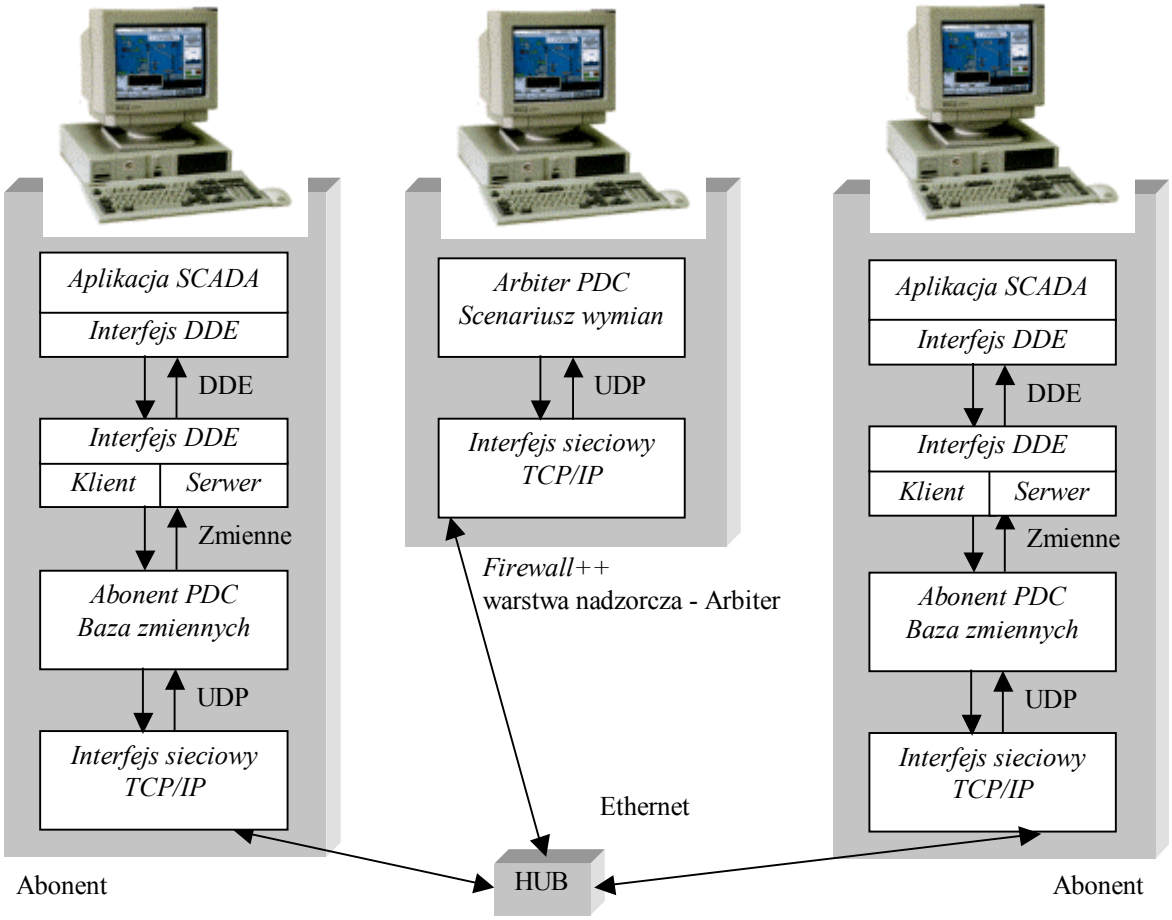
Transakcje odbywają się według klasycznego schematu transakcji cyklicznych modelu PDC [60, 61, 58, 114, 119, 115]. Pojawia się zapytanie o zmienną (informację) ze strony arbitra i pojawia się odpowiedź producenta. Wszyscy abonenci korzystający ze zmiennej mogą wówczas w tym samym czasie odczytać wartość informacji. Dla sieci zamkniętej lub izolowanej, gdzie w sieci wewnętrznej pracuje protokół TCP/IP i brak jest ruchu obcego, stworzone aplikacje gwarantują uzyskanie zdeterminowanego w czasie dostępu do przesyłanych informacji.

Przepływ informacji w systemie komunikacyjnym opartym o opisywane programy przedstawiony jest na rysunku 96. Analizując rysunek można zauważyć, iż system komunikacyjny zawiera szereg rozdzielnych niezsynchronizowanych obiegów informacji. Część obiegów związana jest z pracą samej sieci oraz ewentualnym cyklem arbitra, a część z przepływem w warstwach wyższych oraz warstwach interfejsowych pomiędzy procesami. Dla takich obiegów idealnym rozwiązaniem określania jakości dostarczanej informacji są opisywane wcześniej mechanizmy statusowe. Informacje statusowe można związać z każdym stykiem pomiędzy obiegami i zawsze może to być informacja trojakiego rodzaju:

- status inicjacji,

- status wytworzenia/dostarczenia,
- status pobrania.

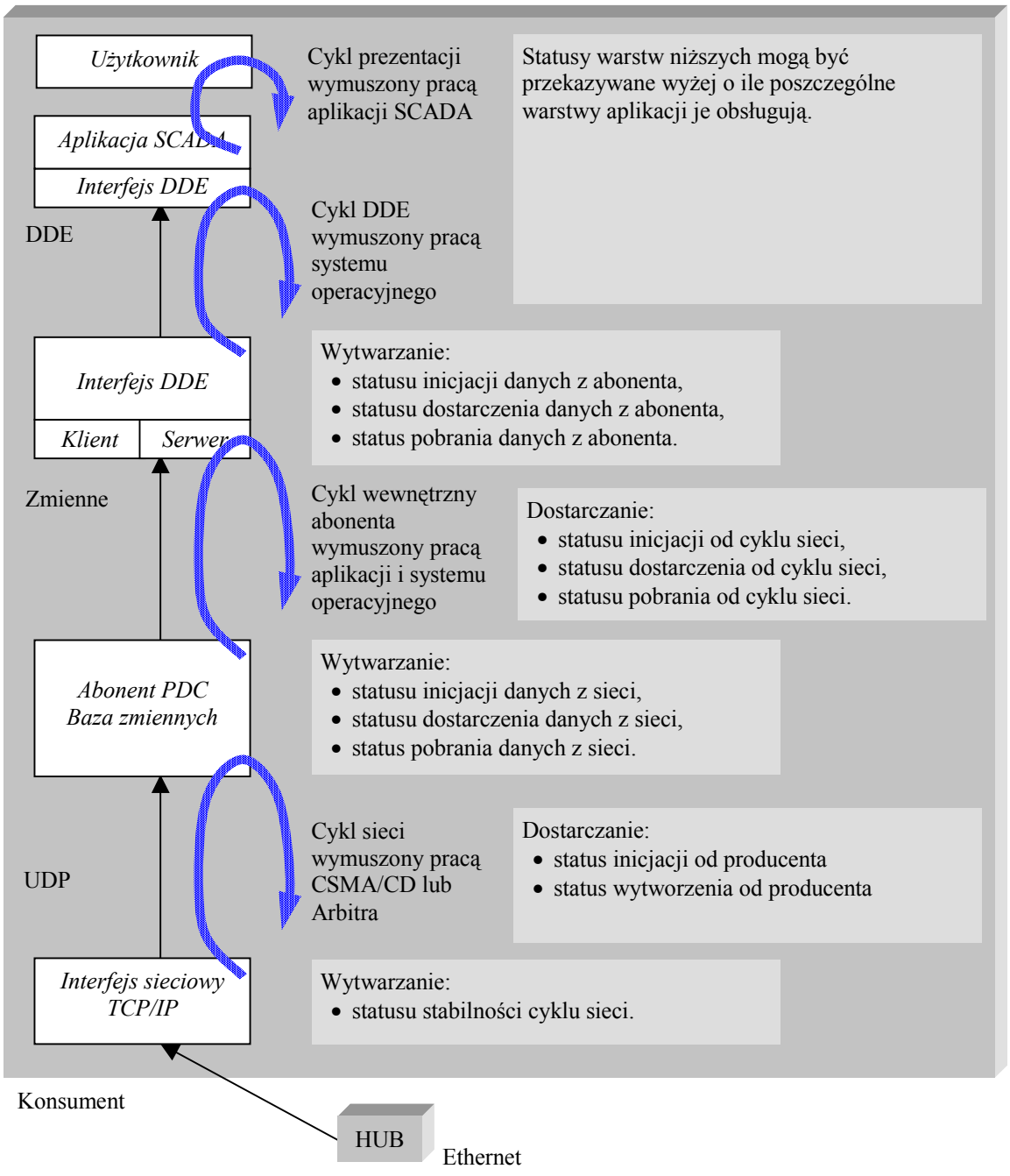
Generalnie mówiąc, status inicjacji odpowiada czy obieg wtórny został zainicjowany przez obieg pierwotny, status wytworzenia / dostarczenia określa czy informacja zawarta w pakiecie przekazywanym z obiegu pierwotnego została wytworzona przez proces uaktualniający zgodnie z zadanym cyklem a status pobrania określa czy obieg wtórny odczytuje z cyklem uniemożliwiającym gubienie danych. Mówiąc o statusie wytworzenia i dostarczenia mamy na myśli ten sam mechanizm. Rozróżnienie następuje w miejscu generacji tej informacji. Jeżeli abonent wytwarzający dane użyteczne pracuje według obiegu cyklicznego uaktualniającego te dane w sieci na podstawie danych dostarczanych z cyklicznych obiegów warstw wyższych to informację o cyklu aktualizacyjnym może zawrzeć w transmitowanym pakiecie. Wówczas mamy do czynienia z informacją statusową określającą okres wytworzenia danych. Natomiast gdy abonent odbierający dane użyteczne posiada obieg pracujący według cyklu uaktualniającego obieg wewnętrzny na podstawie obiegu związanego z siecią, wówczas informacja o pracy cyklu związanego z siecią może być przedstawiona jako status dostarczenia i może być przekazana do obiegów warstw wyższych.



Rys. 96 Przepływ informacji pomiędzy stacjami

Schemat możliwej obsługi statusowej przedstawiono na rysunku 97. Wszystkie informacje statusowe są wypracowywane i interpretowane na stacjach abonenckich. Stacja arbitra nie jest

związana z procesem określania jakości, gdyż nie jest również związana z procesem dystrybucji bądź redystrybucji informacji użytecznej. Analogicznie do przedstawionego schematu dla konsumenta będzie wyglądała obsługa statusów dla producenta, tyle że informacja statusowa będzie „schodziła" od warstwy aplikacji użytkowej do warstw sieciowych. Pojawiający się najniżej status stabilności cyklu jest sposobem na określanie z poziomu odbiorcy stabilności czasowej nadchodzenia pakietów związanych logicznie z daną informacją np. ze zmienną cykliczną.



Rys. 97 Możliwości obsługi informacji statusowej

Łatwo również zauważyć, że większość obiegów informacji zależy od pracy systemu operacyjnego. Aplikacja została zaimplementowana [113, 134, 133, 129, 125] pod systemem

Windows 2000 Professional w jego warstwie użytkowej, a zatem brak jest gwarancji na pracę tych obiegów w czasie rzeczywistym, gdyż w tej warstwie system nie jest czasowo zdeterminowany [124]. Jedyne na co może pozwolić sobie aplikacja to podwyższenie priorytetów wątków obsługujących te obiegi. Umieszczenie jednak ich w warstwie jądra systemu lub zaimplementowanie dla innego systemu operacyjnego spełniającego wymogi pracy w czasie rzeczywistym (np. QNX, VxWorks), może rozwiązać ten problem [127, 90, 118, 121, 87]. Wybór implementacji pod systemem Windows 2000 był podyktowany względami ekonomicznymi. Stworzone programy obsługują większość z przedstawionych na rysunku 97 informacji statusowych, za wyjątkiem przekazywania statusów do warstw najwyższych służących do integracji z innymi procesami.

Aplikacje są skonstruowane w taki sposób, aby mogły również pracować w sieci otwartej. Jest to możliwe, lecz bez wykorzystania arbitra i mechanizmów rozgłaszania. W intersieci komunikacja pomiędzy aplikacjami odbywa się na zasadzie połączeń przez konkretne adresy IP i wymiany swobodne. Nie ma wówczas możliwości uzyskania determinizmu, lecz dzięki możliwości śledzenia cyklu sieci i statusów można sparametryzować pracę komunikacji w sposób optymalny, czyli taki, aby spełniała wymagania procesu.

Wykonano szereg testów dla pracy aplikacji z arbitrem oraz pracy swobodnej. Wyniki testów zamieszczono w załączniku VI.C.

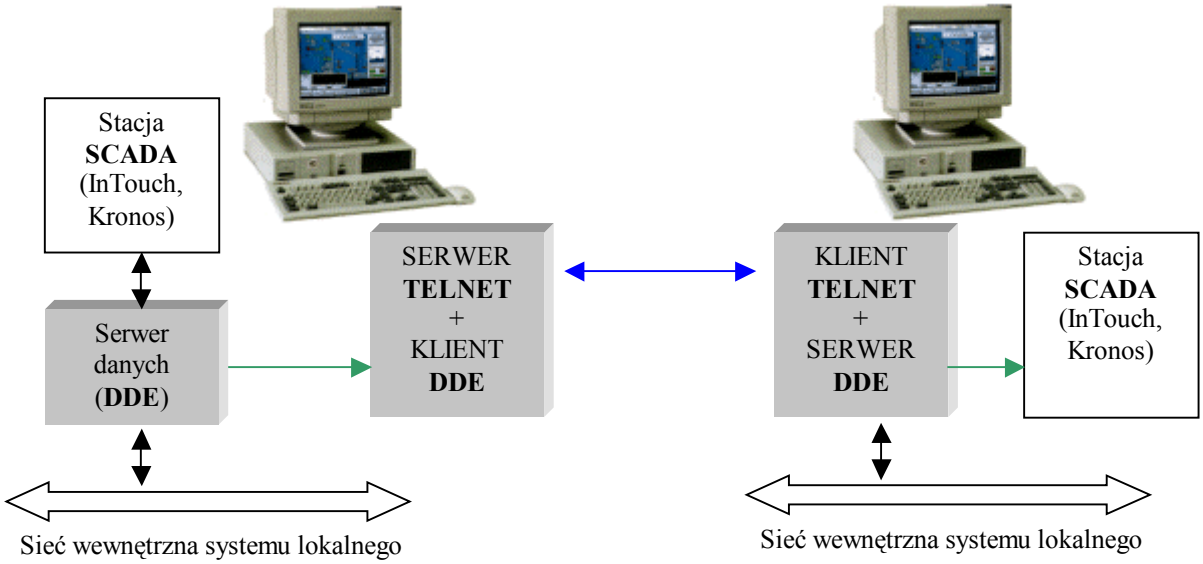
2 Zdalny monitoring – Telnet

Zdalny monitoring jest najczęściej spotykaną usługą intersieciową wykorzystywaną w pracy systemów przemysłowych. Często zdarza się, iż potrzebujemy uzyskać zdalny dostęp do kilku wybranych pomiarów, w prosty i niedrogi sposób. Podczas prac nad niniejszą rozprawą powstały dwa programy (rys. 98) do realizacji usługi zdalnego monitoringu wybranych parametrów procesu przy użyciu protokołu Telnet [104]. Programy te umożliwiają połączenie dwóch dowolnych programów posiadających interfejs DDE przez sieć intersieć na bazie TCP/IP. Telnet został wybrany ze względów opisanych w rozdziale 10.2, a szczególnie ze względu na możliwość eliminacji specjalizowanego oprogramowania klienta po stronie użytkownika zdalnego.

Idealnym przykładem takiego połączenia są stacje wizualizacyjne InTouch. W systemie InTouch cały lokalny ruch zmiennych odbywa się przez mechanizm DDE. Program serwera protokołu Telnet umożliwia połączenie ze stacją InTouch. Stacja ta produkuje zmienne mające być monitorowane na stacji zdalnej. Program klienta protokołu Telnet umożliwia połączenie z serwerem i używając specjalnie zdefiniowanego polecenia może uaktywnić połączenie DDE serwera ze stacją InTouch. Po aktywacji wartości wybranych zmiennych są transmitowane cyklicznie w postaci tekstowej do klienta. Program klienta posiada również możliwość pracy jako serwer DDE umożliwiając podłączenie zdalnej stacji InTouch lub innego programu z interfejsem DDE. Program serwera umożliwia obsługę trzech adresów

DDE co daje możliwość śledzenia zdalnego trzech zmiennych pochodzących z systemu wizualizacyjnego.

Istotną cechą użytkową takiego rozwiązania jest duża uniwersalność zastosowań. Najprostszą wersją aplikacji jest połączenie stacji SCADA z serwerem usługi monitoringu. Użytkownik otrzymuje możliwość podłączenia się do systemu z dowolnego miejsca intersieci przy użyciu dowolnego programu klienta protokołu Telnet. Aby zrealizować bardziej wyszukaną aplikację można użyć specjalizowanego programu klienta. Program ten potrafi lepiej obsłużyć sam proces monitorowania przez intersieć jak również umożliwia wyliczanie statusu stabilności cyklu sieci. Specjalizowany program klienta wyposażony jest dodatkowo w wizualizację czasu transmisji pakietu związanego z monitorowaniem umożliwiając tym samym wizualną ocenę jakości połączenia.



Rys. 98 Zdalne monitorowanie przy użyciu usługi Telnet

W zestawie poleceń serwera zdefiniowano również polecenie umożliwiające statystyczną ocenę danego łącza pod kątem czasu transmisji pakietu od klienta do serwera i z powrotem. Po zrealizowanym teście dla n próbek (n<10000) serwer przesyła wyniki pomiarów do programu klienta wraz z wyliczonymi wartościami średnimi, minimalnymi i maksymalnymi czasu.

Istotną zaletą takiego rozwiązania jest jego niski koszt, sprowadzający się do kosztów instalacji serwera usługi. Do wad należy zaliczyć ograniczoną funkcjonalność oraz niewielki stopień zabezpieczenia transmisji.

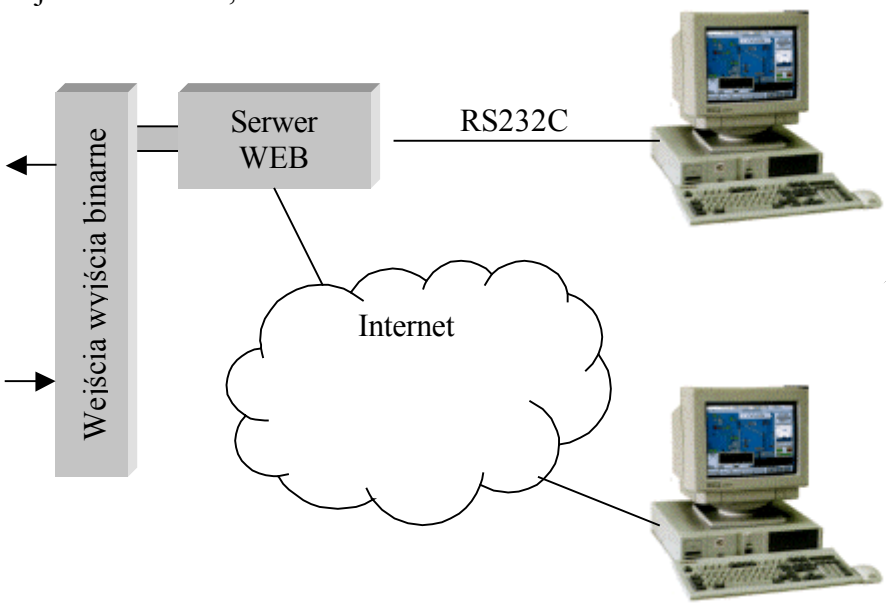
Przy użyciu opisywanych narzędzi wykonano testy połączenia pomiędzy stacjami InTouch pracującymi w różnych sieciach. Do realizacji połączenia wykorzystano Internet. Wyniki testów przedstawiono w załączniku VI.B.

3 Specjalizowany serwer wbudowany WEB

Na potrzeby zajęć dydaktycznych prowadzonych w laboratorium sieci przemysłowych oraz przemysłowych systemów komputerowych w Instytucie Informatyki Politechniki Śląskiej stworzono stanowisko laboratoryjne demonstrujące dwuwarstwową architekturę systemu zdalnej wizualizacji i sterowania. Rozwiązanie oparto o wbudowany serwer webowy Rabbit 2000, oraz klientów w postaci przeglądarek internetowych IE 5.0.

Stanowiska składają się z następujących elementów sprzętowo-programowych:

- komputera PC,
- serwera webowego na bazie procesora Rabbit 2000,
- układu wejść i wyjść binarnych obsługiwanych przez serwer,
- środowiska do tworzenia programów w języku Dynamic C,
- komputera testowego z dostępem do Internetu i przeglądarką internetową,
- lokalnej sieci Ethernet,



Rys. 99 Stanowisko laboratoryjne dla zdalnej wizualizacji

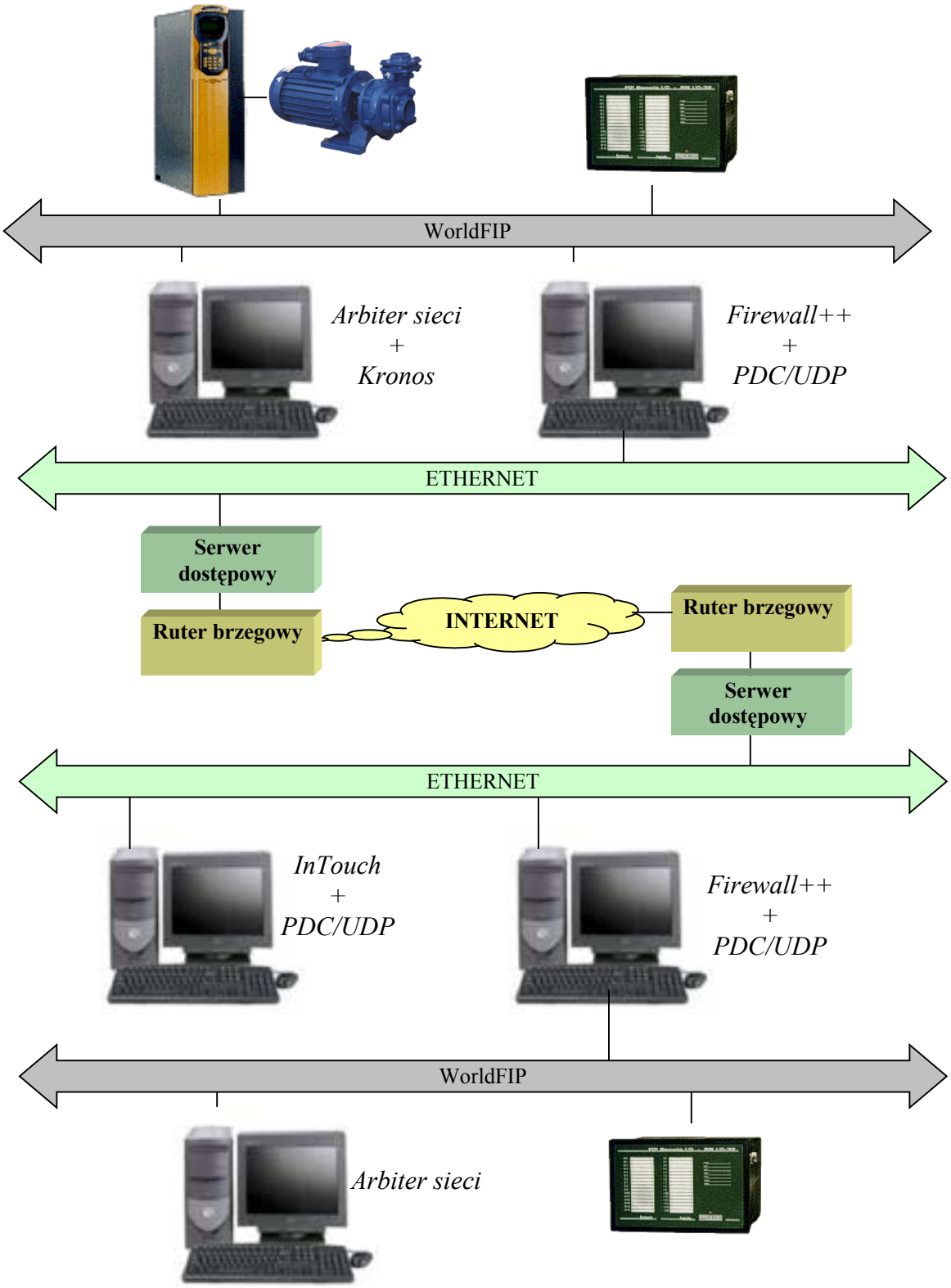
Celem ćwiczeń wykorzystujących opisywane stanowisko jest zapoznanie się z możliwościami prowadzenia wizualizacji procesów technologicznych przez wykorzystanie zdalnego dostępu do systemu lokalnego z poziomu Internetu. Zadaniem studentów jest zaprogramowanie serwera w taki sposób, aby na komputerze testowym można było odczytać stan wyjść i wejść binarnych urządzenia, a także sterować wyjściami. Na zajęciach rozważany jest aspekt bezpieczeństwa zdalnego sterowania.

4 System Kronos

System Kronos [23, 48, 50, 120] stanowi klasyczną aplikację systemu typu SCADA. Zajęcia dydaktyczne prowadzone z wykorzystaniem tego systemu zostały wzbogacone o możliwość generacji stron WWW oraz zdalnego podłączenia przez mechanizm DDE i protokół Telnet.

5 Integracje systemów lokalnych

W ramach prac laboratoryjnych wykonano próby połączenia dwóch systemów lokalnych wykorzystujących deterministyczne sieci specjalizowane. Schemat całości połączeń przedstawiono na rysunku 100.



Rys. 100 Połączenie dwóch systemów lokalnych przy użyciu Internetu, *Firewalla++*, oraz protokołu UDP

Wykonano połączenie sieci systemowej WorldFIP przez firmową sieć lokalną i Internet z systemem lokalnym opartym na sieci Ethernet. W sieci systemowej pracował moduł zdalnych wejść wyjść oraz przekształtnik MV3000. Połączenie z systemem zdalnym zostało wykonane przez moduł *firewalla++* pracującego na komputerze klasy PC [52, 117]. Połączenie intersieciowe zrealizowane zostało pomiędzy miastem Katowice a siecią uczelnianą w Gliwicach. W sieci lokalnej Instytutu podłączone zdalnie do systemu lokalnego były komputery PC z systemem SCADA InTouch oraz *firewall++*, który realizował przekazywanie danych do i z modułu zdalnych wejść wyjść w sieci systemowej WorldFIP. Komputery pracowały pod kontrolą systemu operacyjnego Windows 2000 Professional [113, 114, 115, 116, 124, 125, 129, 133, 134].

Ćwiczenie udowodniło możliwość praktycznego połączenia dwóch systemów lokalnych przy użyciu intersieci. Wykonano również próby zestawienia intersieciowego tunelu dla protokołu WordFIP, lecz efekty do momentu publikowania pracy nie były zadowalające.

B. Wyniki testów dla monitoringu przy użyciu połączenia protokołem Telnet

Do testów wykorzystano następujące aplikacje:

- IICSClient – klient protokołu Telnet z dodatkową obsług informacji przesyłanych przez program serwera.
- IICSSerwer – serwer protokołu Telnet z obsług dodatkowych informacji przesyłanych do klienta.
- Telnet – standardowy klient protokołu Telnet systemu Windows 2000.

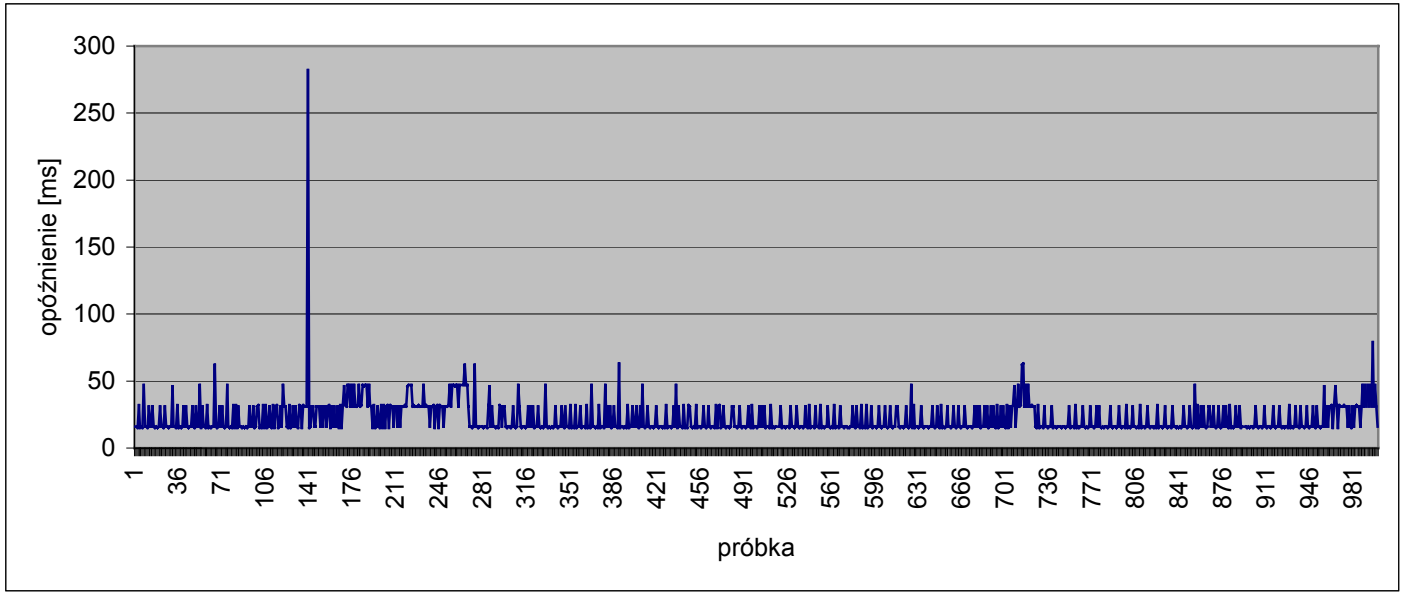
Testy były prowadzone pomiędzy dwoma komputerami pracującymi w sieciach lokalnych i połączonych ze sobą Internetem. Typowa droga pomiędzy nimi przedstawiona jest poniżej w postaci nazw serwerów pośredniczących w transmisji:

```
1. gateway.proloc.com.pl [195.117.205.65]
2. prt.proloc.com.pl [195.117.205.2]
3. 194.204.144.125
4. z.kat-ar2.do.kat-r1.tpnet.pl[195.205.0.153]
5. do.kat-ar1.z.kat-r1.tpnet.pl[213.25.5.194]
6. do-sask.katowice.tpnet.pl[194.204.145.130]
7. G-CK-r6--K-PSE-r2.SILWEB.PL[157.158.254.190]
8. yogi.iinf.polsl.gliwice.pl[157.158.11.1]
9. top.iinf.polsl.gliwice.pl[157.158.57.1]
```

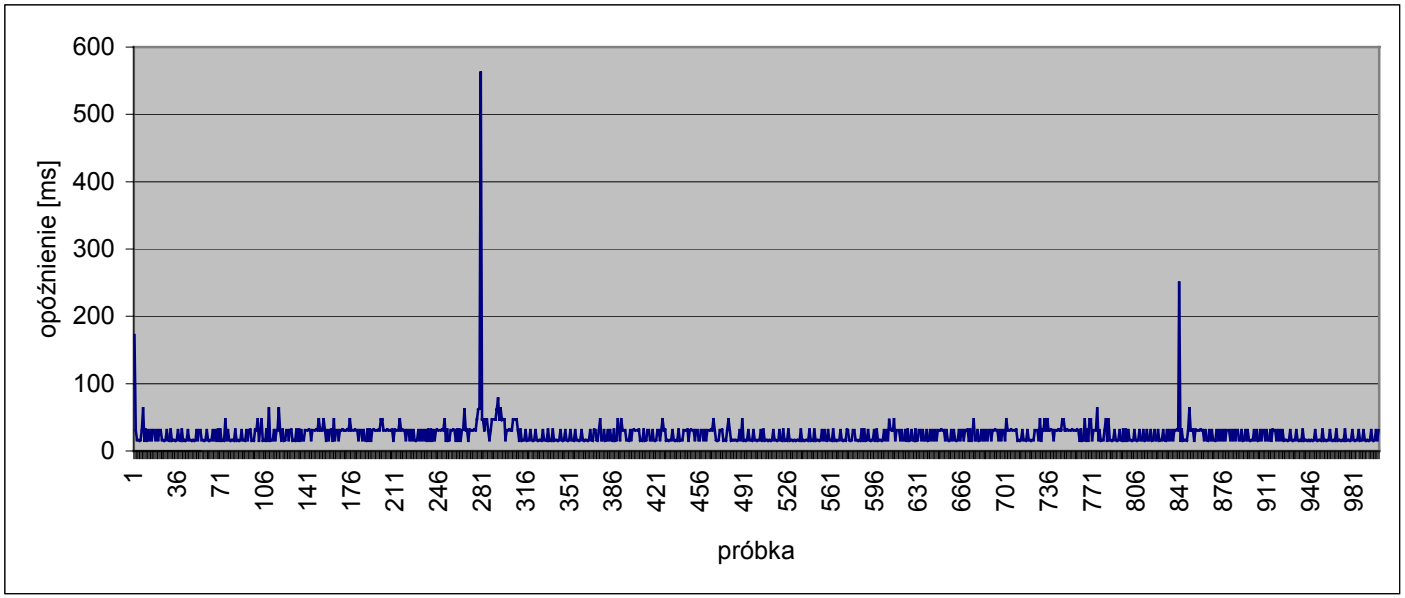
Wykonywano pomiary opóźnień pomiędzy kolejnymi odczytami pakietów dla transmisji 10, 100, oraz 1000 pakietów według transakcji serwer – klient – serwer. Wykonywano także pomiary stabilności cyklu sieci dla pakietów przesyłanych od serwera dla klienta i przenoszących wartość przykładowej monitorowanej zmiennej. Stabilność cyklu była obliczana po stronie klienta na podstawie pomiaru opóźnień pomiędzy kolejnymi nadejściami pakietów ze zmienną. Testy powtarzano wielokrotnie a prezentowane wykresy zostały wybrane jako przykłady.

Prezentowane wyniki doskonale obrazuj charakter pracy intersieci. Na wykresach opóźnień można zaobserwować chwilowe przeciążenia sieci objawiające się nagłym wzrostem wartości opóźnień transmisji. Przy dokładniejszej analizie wycinka cyklu można zaobserwować drobne niestabilności jego pracy. Lepiej jest to zobrazowane na wykresach przedstawiających status cyklu sieci. Wartość „1” oznacza dla danego pakietu wydłużenie cyklu ponad zdefiniowany okres. Jak widać z wykresu sieć przez większość czasu trwania pomiaru zachowywała parametry cyklu. Dość częste zaburzenia są chwilowe, a podczas innych testów zaburzenia te nie zdradzały charakteru ciągłego.

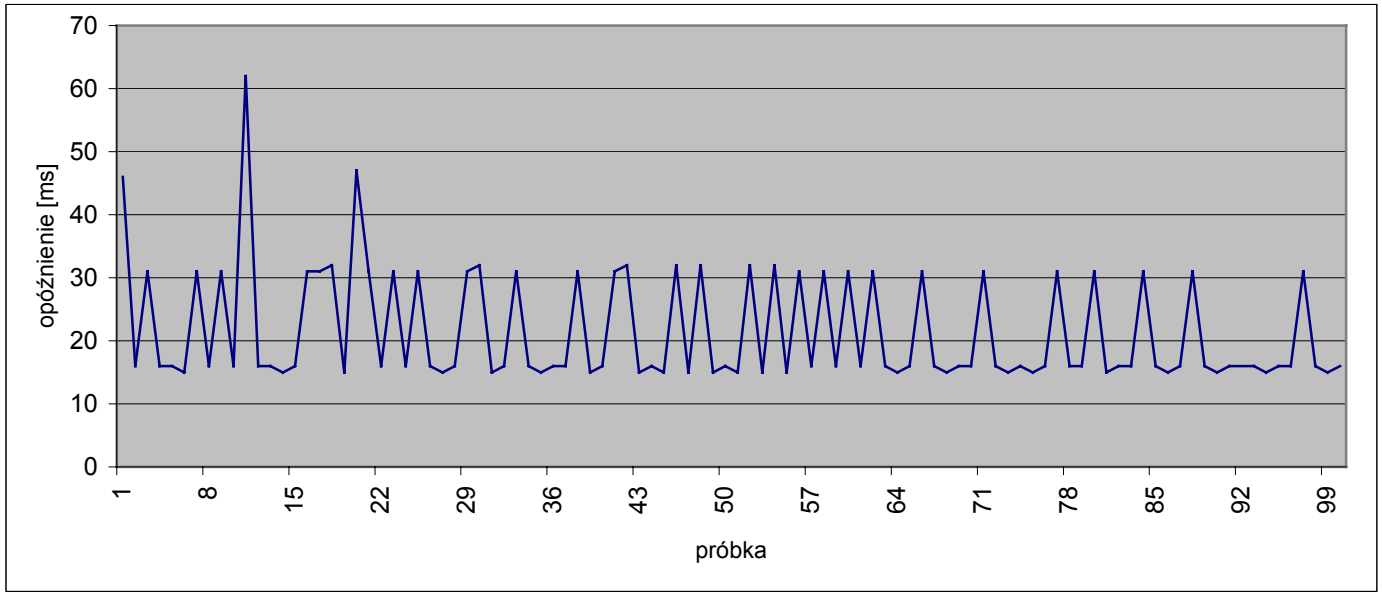
Wyniki te potwierdzają, iż dla zastosowań opisanych w 10 oraz 14 pracująca bezawaryjnie sieć Internet nie stanowi przeszkody i może być stosowana. Jednocześnie, obserwowane zakłócenia cyklu oraz wielkości opóźnień z nimi związane dyskwalifikują Internet do prowadzenia komunikacji w procesach wymagających zastosowania czasu krytycznego.



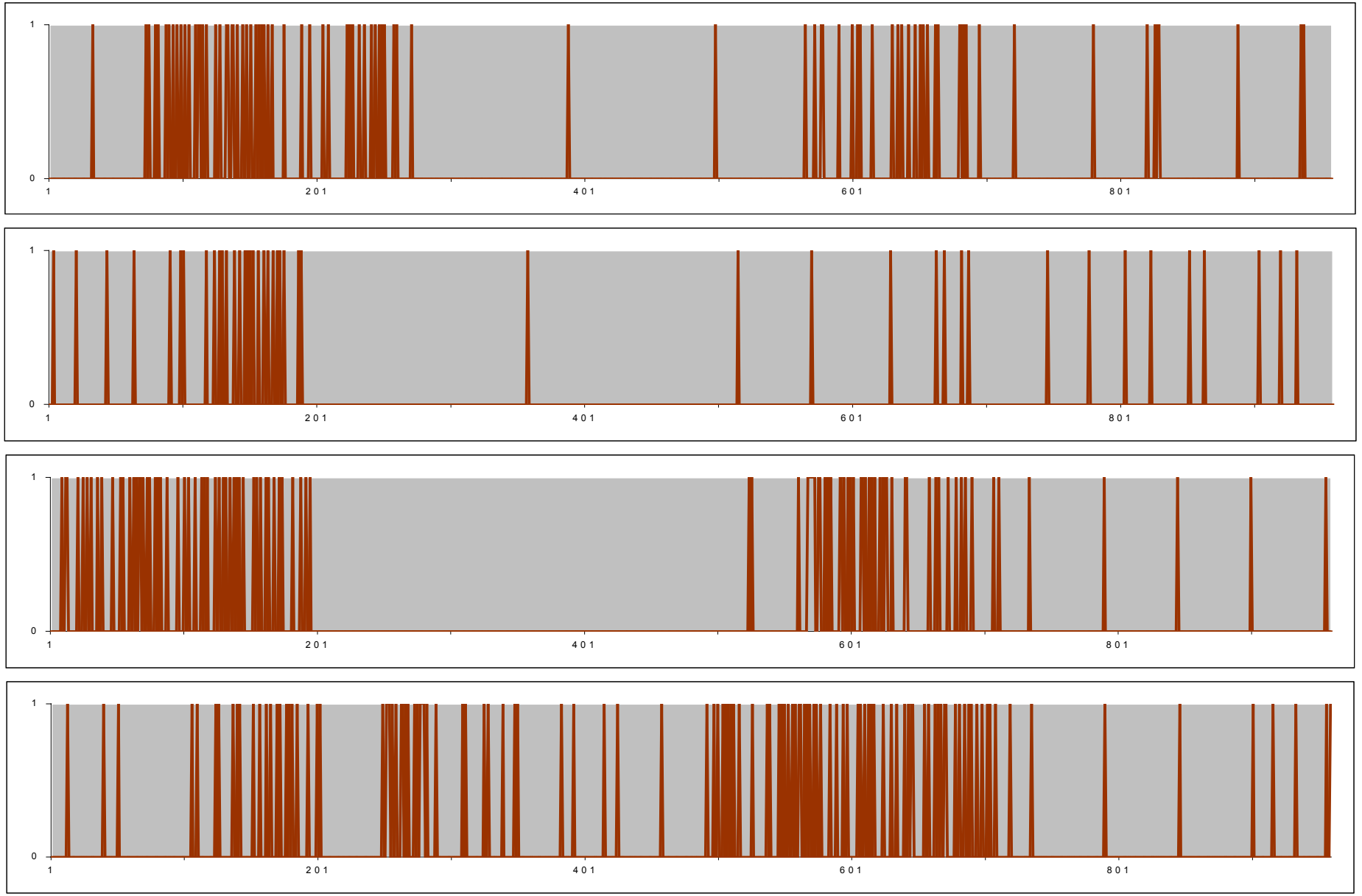
Wykres 1 Wartości opóźnień transmisji 1000 pakietów (próbka I) dla sieci Internet



Wykres 2 Wartości opóźnień transmisji 1000 pakietów (próbka II) dla sieci Internet



Wykres 3 Wartości opóźnień transmisji 100 pakietów dla sieci Internet



Wykres 4 Odwrócony status stabilności cyklu sieci dla przykładowej transmisji protokołem Telnet dla sieci Internet

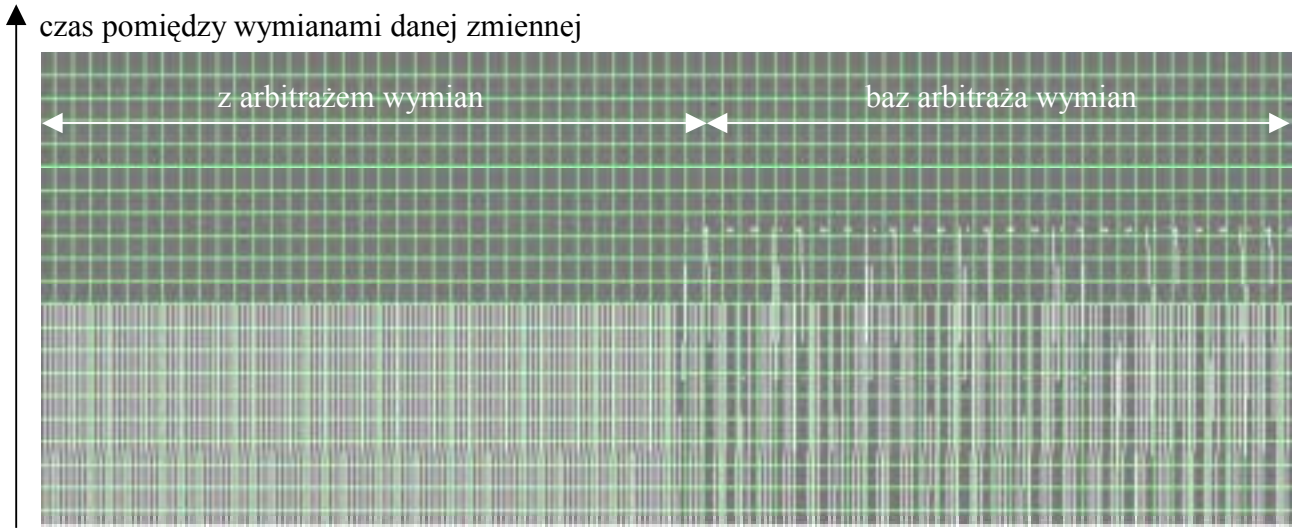
C. Wyniki testów dla sieci Ethernet i protokołu na bazie modelu PDC

Poniżej przedstawiono wyniki testów dla pracy aplikacji wizualizacyjnych przy wykorzystaniu do połączenia sieciowego izolowanej sieci Ethernet i protokołu aplikacyjnego na bazie modelu PDC. Do testów wykorzystano następujące aplikacje:

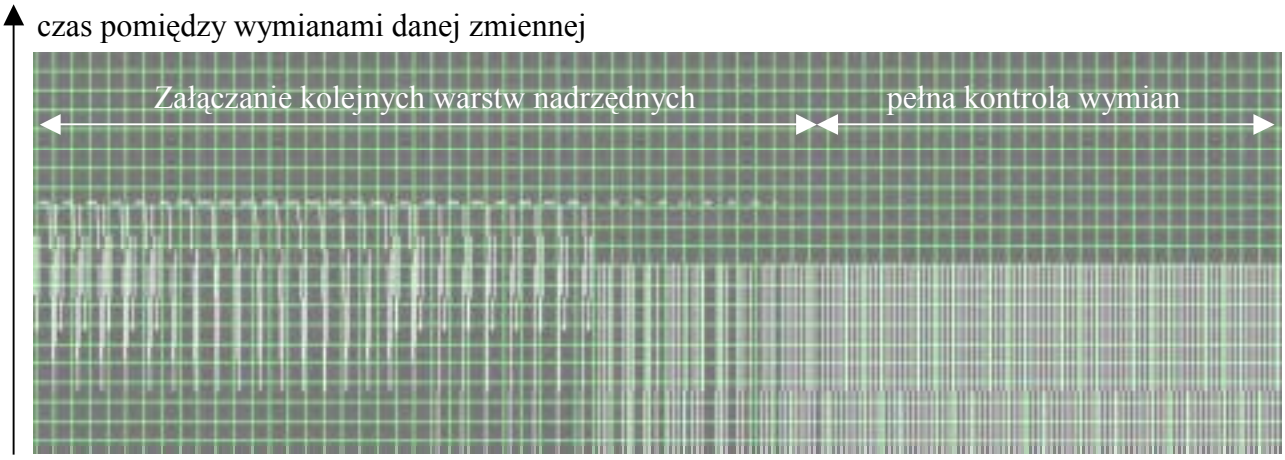
- Arbiter program zarządzający transakcjami,
- Abonent program obsługujący zmienne oraz łącza z programami SCADA.

Testy prowadzona na sieci lokalnej zbudowanej na bazie trzech komputerów klasy PC, Ethernetu 100Mb w trybie halfduplex oraz 100Mb przełącznika (ang. *switch*). Jeden z komputerów posiadał uruchomione programy „Arbiter" oraz „Abonent", pozostałe posiadały tylko uruchomionego abonenta. Zmienne obsługiwane przez programy abonentów były połączone ze zmiennymi z aplikacji systemu InTouch 7.1. Warstwa transportu bazowała na protokole UDP.

Poniższe wykresy przedstawiają wyniki pomiarów cyklu sieci.



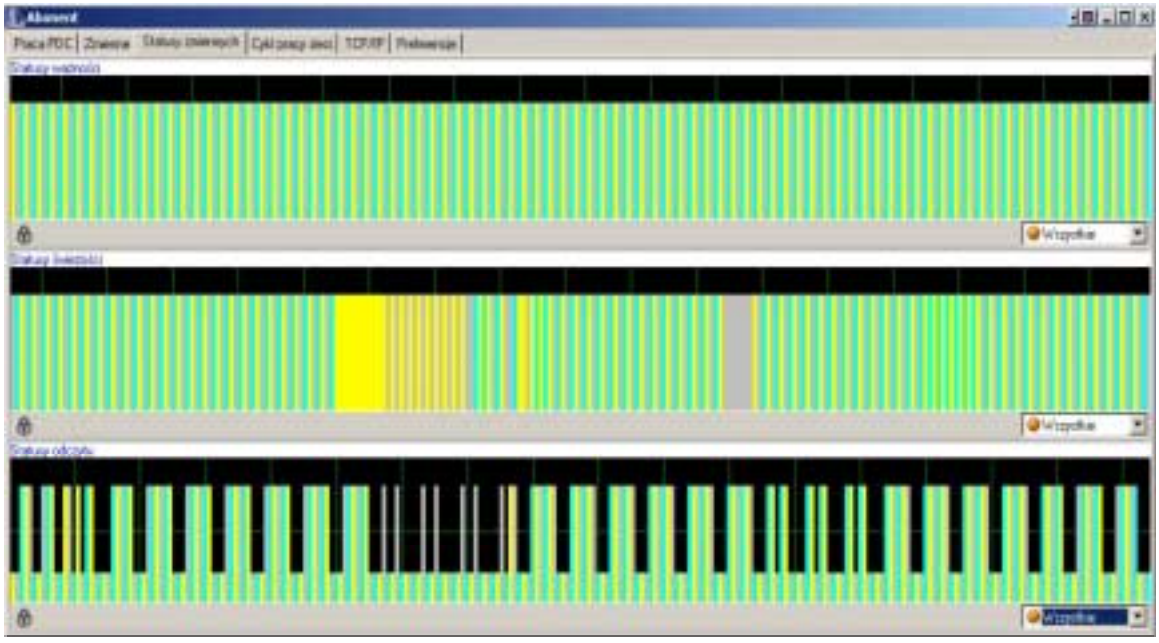
Rys. 101 Cykl wymian UDP



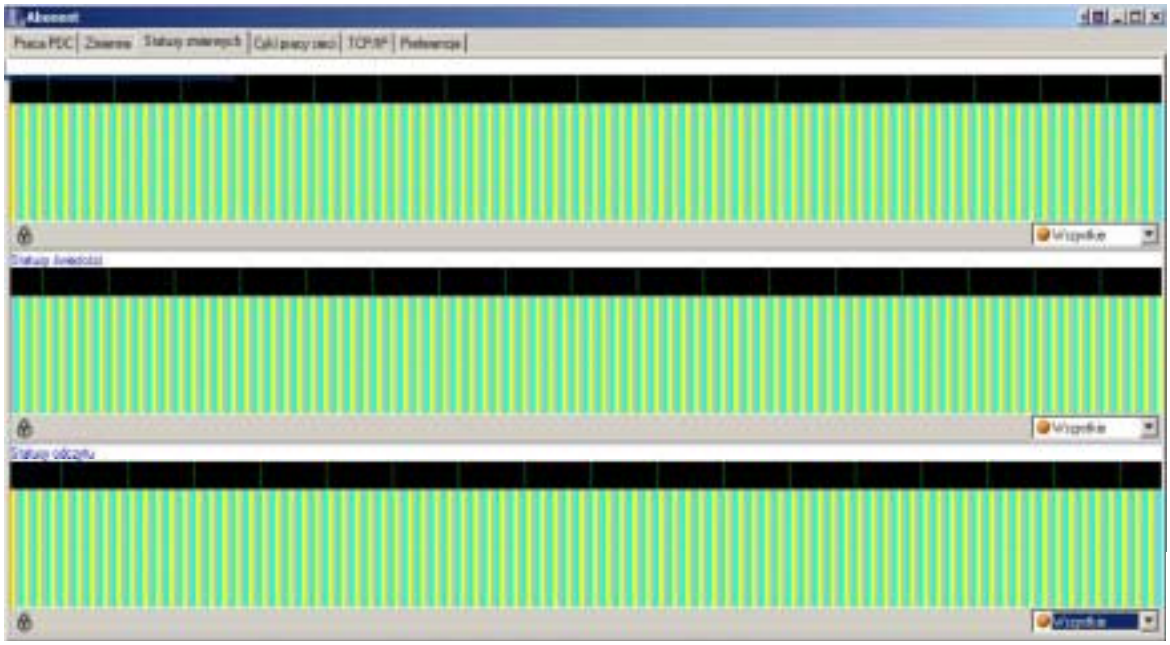
Rys. 102 Załączanie nadrzędnych warstw aplikacyjnych w sieci Ethernet

Rys. 103 Makrocykl wymian z protokołu WorldFIP na sieci Ethernet

Poniżej zamieszczono wykresy statusów ważności, świeżości i odczytu zmiennych wygenerowane przez program konsumenta. Wysoki słupek oznacza spełnienie warunków danego statusu w danym cyklu, niski brak spełnienia.



Rys. 104 Statusy bez uruchomionego arbitrażu



Rys. 105 Statusy z uruchomionym arbitrażem

Autor: Piotr Gaj – Politechnika Śląska, Instytut Informatyki
Temat: Rozprawa doktorska/Zastosowanie protokołu TCP/IP do transmisji informacji dla potrzeb przemysłowych systemów kontrolno-nadzorczych
Plik: Praca doktorska.doc/2001-07-21 19:27
Druk: 2004-05-05 10:17 Poprzedni: 2003-09-08 15:38
Wersja: 3.9.1033 (*modyfikacja główna.przebudowa wewnętrzna.numer zapisu*)
Czas: 24983 min