

Rafał CUPEK, Marcin FOJCIK  
Politechnika Śląska, Instytut Informatyki

## BUDOWA MODUŁÓW KOMUNIKACYJNYCH STACJI NADZORCZEJ Z SIECIAMI PRZEMYSŁOWYMI

**Streszczenie.** Publikacja przedstawia próbę rozwiązania problemu konstruowania modułów komunikacyjnych dla stacji kontrolno-nadzorczych w sieciach przemysłowych. W pracy przedstawiano wymagania stawiane przed sterownikami komunikacyjnymi wykorzystywanymi dla celów wizualizacji procesów przemysłowych. Podjęto próbę klasyfikacji powszechnie stosowanych protokołów sieci przemysłowych pod kątem ich zastosowania do transmisji danych wykorzystywanych przez stację kontrolno-nadzorczą. Autorzy opracowania starali się także przedstawić własną propozycję realizacji sterowników komunikacyjnych obejmującą modułową budowę stacji kontrolno-nadzorczej oraz obiektową strukturę poszczególnych modułów oprogramowania.

## THE CONSTRUCTING COMMUNICATION MODULES FOR INDUSTRIAL CONTROL STATION

**Summary.** In this paper a few aspects of constructing communication modules for industrial control station are presents. This article describes also the requirements of timing and data transmitting. Short look of classification industrial network including visualization requirements have been presented. Authors of this paper try to describe their propositions of realization communication modules for industrial control station.

## 1. Wstęp

Rozwój technologii urządzeń sterownikowych budowanych w oparciu o mikroprocesory sprawia, iż systemy automatyki przemysłowej znajdują coraz to nowe zastosowania w dziedzinie sterowania i kontroli procesem przemysłowym. Duża ilość dostępnych informacji pozwala operatorowi na pełniejszą kontrolę nad realizacją procesu, konieczne staje się jednak stosowanie komputerowych stacji wizualizacji, zarządzania i nadzoru. Stacje te zbierają w jednym miejscu informacje pochodzące z różnych punktów obsługiwane procesu dając pełną informację technologiczną na temat jego przebiegu.

Najbardziej istotną funkcją tych stacji jest cykliczne pobieranie danych w celu ich przetwarzania. Dokonuje się to za pomocą specjalizowanych modułów komunikacyjnych. Można powiedzieć, że sprawność i jakość tych modułów decyduje o sprawności całego systemu.

## 2. Funkcje modułów komunikacyjnych

Moduły komunikacyjne służą do transmisji danych do i ze stacji nadzorczej. Należą do nich sterowniki sieci przemysłowych, sieci lokalnych i moduł dialogu z użytkownikiem. Podstawowym ich zadaniem jest pobieranie danych z zewnątrz do systemu. Moduły powinny posiadać kilka podstawowych funkcji:

*Odpowiednio szybkie pobieranie informacji* - dane pobierane przez moduł powinny być aktualne. Istotne jest to w przypadku danych zmieniających się bardzo szybko. Jeżeli moduł komunikacyjny nie jest w stanie na bieżąco pobierać potrzebnych informacji, ale pobiera dane nieaktualne, cały system nie może poprawnie pracować.

*Praca cykliczna* - moduły powinny mieć odpowiednio ustawione parametry pracy (tj. kolejność pobierania danych, cykl pobierania), a w przypadku braku możliwości działania zgodnie z opisem moduł powinien sygnalizować przekroczenia parametrów. Jest to funkcja potrzebna m.in. dla lepszej kontroli działania modułu - moduł jest autonomiczny, sam kontroluje swoją pracę i przy każdej akcji sprawdza, czy przewidziane czasy są zachowane. Ciągłe wykonywanie tych działań z poziomu stacji nadzorczej byłoby trudne.

*Ujednoczenie danych* - każda sieć przemysłowa czy lokalna ma swój własny protokół, sposób przekazywania informacji, postać danych, czas oczekiwania na dane. W stacji nadzorczej dane te powinny być dostępne w sposób ujednoczony. Najprostszym sposobem realizacji tego jest przygotowanie odpowiedniej struktury (bufora) przechowującej wszystkie dane w postaci ujednoczonej - tzw. obraz systemu. Na przykład, można stworzyć strukturę - kopię całego systemu rozproszonego, tj. sterowników, w pamięci stacji nadzorczej. Dane ze sterow-

ników zbierane byłyby w różnych formach: bitów, słów, grup słów o różnych rozmiarach, a następnie przetwarzane za pomocą modułów komunikacyjnych i wpisywane do struktury bufora (obrazu systemu) jako słowa (z uwzględnieniem czasu pobrania). W takim przypadku można dobierać czasy aktualizacji indywidualnie dla poszczególnych pomiarów, a inne moduły stacji nadzorczej mogą używać w tym samym czasie wszystkich danych pomiarowych (lecz z różnymi czasami aktualizacji). Ujednoczenie danych obejmuje także dopasowanie formatu przesyłanych danych do formatu wielkości pomiarowych. Często zdarza się, że w celu zwiększenia dokładności pomiaru lub uproszczenia komunikacji wartość przesyłana nie jest wprost wartością pomiarową. Na przykład, w sieci MODBUS przesyłane są jako dane wartości z zakresu 0-65535 (na 16 bitach). Jeżeli chcemy przesłać wartość 100000 (np. przepływ w t/h dla stacji rozdzielczej), należy wartość mierzoną podzielić w sterowniku, aby otrzymać wielkość, która mieści się w zakresie danych przesyłanych przez sieć, a następnie pomnożyć ją w stacji nadzorczej, aby uzyskać poprawną wartość pomiarową do zapisania w buforze-obrazie systemu rozproszonego. Inną sytuacją jest używanie w sterownikach wejść analogowych. Wejścia te mogą mieć różne zakresy 0..20mA, 4..20mA, -10..10V. Ten sam parametr po zmierzeniu go za pomocą różnych przetworników daje inne wartości. W przypadku zmiany przetwornika podczas pracy w systemie pojawiają się nagle nieprawidłowe wartości. W tym przypadku wskazane jest, aby w module komunikacyjnym była możliwość wyboru przez użytkownika typu przetwornika, a co za tym idzie, prawidłowe dopasowanie się do aktualnych danych.

*Próg nieczułości* - jest to minimalna wartość, o którą musi zmienić się wielkość mierzona, aby uznać to za zmianę wartości. W systemach używanych w przemyśle jest ściśle określona dokładność pomiaru. Zdarzają się niewielkie zmiany, których nie da się uniknąć, natomiast nie powinny być one odbierane jako zmiana wartości, np. zmiana mierzonej temperatury o 0.05C jest możliwa do zmierzenia, ale nie zawsze potrzebna. W przypadku każdorazowej rejestracji takiego zdarzenia i uaktualnienia wszystkich ekranów graficznych, na których występuje ta wielkość, mogą być wykonywane niepotrzebne czynności, które zmniejszają efektywność stacji nadzorczej.

### 3. Wizualizacja przebiegu procesu

#### 3.1. Graficzna prezentacja procesu

Większość pomiarów zbieranych poprzez moduły komunikacyjne powinna być zaprezentowana użytkownikowi. Na podstawie ich wyglądu użytkownik może poznać aktualny stan obiektu oraz historię pomiarów. Ważną rolę odgrywa sposób pokazywania; powinien on być jednoznaczny, czytelny i przyjazny dla odbiorcy. Istotny jest kolor prezentacji elementu,

ewentualne zmiany koloru (miganie), wielkość, położenie na ekranie komputera, a nawet tło (wartość mierzona pokazana jest na rysunku instalacji z dokładnym pokazaniem miejsca pomiaru). Wizualizacja w trybie graficznym jest dużo bardziej skuteczna, przekazuje więcej informacji niż sam opis tekstowy.

Najczęściej używanym elementem graficznej prezentacji procesu są pomiary. Służą do bezpośredniego przedstawienia stanu obiektu. Za ich pomocą można pokazać aktualną wartość - jako liczbę, ewentualne przekroczenia wcześniej zadanych wielkości - jako kolor prezentacji liczby, przekroczenia stanów alarmowych jako zmiany kolorów (miganie) liczby. Pomiar w połączeniu z odpowiednim tłem daje informację również o miejscu pobierania wartości. Dodatkowo do pomiaru można dołączyć opis słowny, tzn. opis pomiaru według funkcji lub według przyjętych przez użytkownika zasad opisu. Istotne są też jednostka pomiarowa i dokładność pomiaru.

Inną formą przedstawienia pomiarów są trendy. Jest to przedstawienie kolejnych wartości danego pomiaru w systemie. Mogą mieć postać tekstową - pokazanie ciągu wartości pomiarowych z czasami ich otrzymania, lub formę graficzną - wykres przebiegu wartości. Forma wykresu daje więcej informacji pozwalając na łatwą obserwację zmian (zwiększenie, zmniejszenie, szybkość zmiany).

Trendy przedstawiają dane przygotowane przez moduł archiwum, które jest niezależnym elementem pozwalającym na zapamiętywanie pewnych danych. Mogą to być sytuacje alarmowe - mówimy wtedy o archiwum zdarzeń. Innym rodzajem jest archiwum pomiarów. Przechowywane są w nim wartości chwilowe, średnie, przyrosty, wartości minimalne i maksymalne za określony czas.

### 3.2. Wizualizacja zdarzeń i alarmów

Alarmy są to sytuacje nieprawidłowej pracy (zazwyczaj wcześniej określone). Istotne jest powiadomienie użytkownika o tej sytuacji jak najszybciej. Ze względu na inny sposób obsługi można wyróżnić trzy rodzaje alarmów: ostrzeżenie, alarm, alarm o najwyższym priorytecie.

Każde wystąpienie alarmu winno być kwitowane przez operatora. Wystąpienie, skwitowanie i ustąpienie alarmu powinno być automatycznie archiwizowane przez system. Bardzo istotnym wymogiem jest takie zaprojektowanie wyglądu alarmów, aby użytkownik natychmiast mógł zauważyć pojawienie się nowego alarmu i mógł rozpoznać jego rodzaj. Alarmy powinny być widoczne jako tekst lub jako zmiana stanu obiektu, w którym pojawił się stan awaryjny, poza tym powinny być dodatkowo sygnalizowane dźwiękowo.

W przypadku tekstu przydatne jest dodanie do opisu alarmu czasu powstania. Treść napisu powinna odzwierciedlać, który z powyższych trzech typów alarmów miał miejsce. Można tego dokonać za pomocą wyboru odpowiedniego koloru i migotania pola opisu. Celowe jest

prezentowanie tych alarmów na osobnym monitorze (komputerze); poprawia to czytelność i jest prostsze w percepcji dla użytkownika.

Rzeczą bardzo ważną dla zdarzeń i alarmów jest zachowanie kolejności ich pojawiania się. W dużych systemach rozproszonych nie zawsze istnieje możliwość natychmiastowego otrzymania wiadomości o zaistnieniu pewnych zdarzeń. Konieczne jest natomiast zapamiętanie stanów związanych z tymi zdarzeniami i przesłanie ich jak najszybciej z zachowaniem kolejności ich powstawania.

#### 4. Sterowniki sieci przemysłowych

Oprogramowanie sterownika sieci przemysłowej dla stacji kontrolno-nadzorczej powinno uwzględniać cechy charakterystyczne obsługiwanej sieci przemysłowej, takie jak czas obiegu informacji w sieci, parametry czasowe transmisji, czy wpływ transmisji danych dla wizualizacji na przepustowość sieci.

Podstawową cechą charakteryzującą wszystkie rozwiązania sieci przemysłowych jest *zde-terminowany czas dostępu do danych*. Właściwość ta umożliwia określenie maksymalnego czasu, po jakim wytworzona informacja dotrze do adresata. W zależności od zastosowanego protokołu transmisyjnego i przyjętej struktury wymian czas ten może być określany wspólnie dla wszystkich transmitowanych danych, z podziałem na dane o określonych priorytetach, jak też wyznaczany indywidualnie dla poszczególnych zmiennych lub ich grup.

Parametry czasowe sieci są wypadkową dwóch czynników: parametrów elektrycznych łącza i zastosowanego protokołu komunikacyjnego. Cechy fizyczne medium transmisyjnego, jego rozległość, parametry elektryczne oraz podatność na zakłócenia decydują o fizycznej prędkości transmisji informacji możliwej do uzyskania za pomocą określonego medium komunikacyjnego. Drugim czynnikiem wpływającym na parametry czasowe warstwy fizycznej są zastosowane urządzenia elektroniczne odpowiedzialne za sposób kodowania stanów logicznych, realizacji algorytmów dostępu do pamięci i innych zasobów sprzętowych abonenta sieci.

Realizowany poprzez warstwę łącza danych *protokół komunikacyjny* decyduje o kształcie i wielkości ramki danych, sposobie generacji sum kontrolnych reakcji na przekłamanie danych lub na nieobecność abonenta. Znając parametry czasowe warstwy fizycznej i realizowany protokół można pokusić się o *pełną analizę czasową transmisji danych* realizowaną dla konkretnej sieci.

Stosowane podczas wizualizacji procesu pionowe transmisje informacji cechują zazwyczaj mniej rygorystyczne wymogi czasowe niż w przypadku transmisji poziomych realizowanych pomiędzy poszczególnymi abonentami obsługującymi dany proces przemysłowy. Wpływ wy-

mian pionowych na parametry czasowe pozostałych transmisji zależy od zastosowanego protokołu sieciowego. Fakt ten należałoby uwzględnić podczas konstruowania stacji kontrolno-nadzorczej. Znając założenia dotyczące liczby wymian i wymaganych parametrów czasowych tak dla transmisji poziomych, jak i pionowych można na etapie konfiguracji systemu dobrać właściwą architekturę systemu rozproszonego stosowną do potrzeb konkretnego procesu.

Klasyfikując sieci przemysłowe ze względu na stosowane protokoły komunikacyjne można wyodrębnić trzy główne grupy: sieci z protokołem „Master-Slave”, sieci z protokołem „krążący żeton” oraz sieci polowe.

#### 4.1. Sieci typu „Master-Slave”

Architektura systemów „Master-Slave” charakteryzuje się wyróżnieniem jednego z abonentów i przypisaniem mu właściwości urządzenia nadrzędnego („mastery” sieci). Pozostali abonenci sieci pełnią rolę urządzeń podrzędnych („slave”-ów). O czasie i rodzaju przeprowadzanych wymian decyduje wyłącznie abonent nadrzędny, zaś rola abonentów podrzędnych ogranicza się jedynie do udzielania odpowiedzi na transmisje inicjowane przez mastery. Sieć oparta na architekturze „Master - Slave” nadaje się więc szczególnie dla systemów, w których dane, produkowane przez urządzenia peryferyjne, przesyłane są do centrum. Na podstawie zebranych informacji centrum podejmuje decyzje i rozsyła polecenia sterujące do poszczególnych elementów wykonawczych. Poszczególne implementacje tego protokołu dostarczają szeregu dodatkowych usług, takich jak: diagnostyka modułów komunikacyjnych, pełna diagnostyka abonenta sieci, czy też zdalny odczyt i zapis programów dla urządzeń swobodnie programowalnych wraz z możliwościami zatrzymywania i uruchamiania przesyłanego oprogramowania. Jako najbardziej rozpowszechnione przykłady implementacji protokołu „Master-Slave” można wymienić takie standardy komunikacyjne, jak MODBUS, SNP, SNPX czy też AKO[1],[7]. Rozwiązania oparte na protokole „Master-Slave” konstruowane są w znacznej mierze jako standardy uniwersalne, wychodzące poza ramy konkretnej grupy producentów sprzętu. Pozwala to na wykorzystywanie sieci tego typu do łączenia urządzeń pochodzących od różnych producentów. Dobrym przykładem jest tu protokół komunikacyjny MODBUS, który z większymi lub mniejszymi zmianami implementacyjnymi doczekał się adaptacji przez większość producentów sterowników swobodnie programowalnych, regulatorów prędkości obrotowej czy też zdalnych modułów pomiarowych.

Sieć typu „Master-Slave” nadaje się dobrze do wizualizacji systemów pomiarowych i telemetrycznych, o niewygórowanych wymaganiach czasowych. Stacja kontrolno-nadzorcza pełni zazwyczaj rolę „mastery” sieci, decydując o tym, jakie informacje są w danym momencie pobierane lub wysyłane do poszczególnych abonentów. Sposób wymiany informacji pomiędzy „Masterem” sieci a abonentami „Slave”-ami definiują dwie listy wymian. Lista wymian cy-

klicznych zawiera opis transmisji realizowanych automatycznie z zadanym uprzednio okresem. Lista wymian aperiodycznych zawiera opis transmisji generowanych w sposób aperiodyczny bądź przez operatora systemu, bądź też automatycznie poprzez wystąpienie jakiegoś zdarzenia. Ze względu na słabe parametry czasowe opisywanego protokołu oraz konieczność każdorazowego udziału stacji nadrzędnej w transmisji wykorzystanie sieci typu „Master-Slave”, w zastosowaniach wymagających prowadzenia wizualizacji wraz z jednoczesnymi poziomymi wymianami danych, wydaje się problematyczne.

#### 4.2. Sterowniki sieci o protokole „krążący żeton”

W zastosowaniach przemysłowych sieci lokalne typu „krążący żeton” są dedykowane komunikacji pomiędzy wieloma abonentami dając gwarancję dużej niezawodności transmisji. Ponadto sieci te cechują się znaczną, jak na zastosowania przemysłowe, prędkością przesyłu informacji. Generalna charakterystyka tego typu sieci jest następująca:

- architektura zdecentralizowana,
- łatwa rozbudowa w systemy rozległe,
- możliwość tworzenia struktury drzewiastej, co oznacza, że wszyscy abonenci danego segmentu sieci są na tym samym poziomie hierarchii i mogą inicjować dialog,
- bezpośredni dialog pomiędzy segmentami,
- gwarantowany czas wymiany informacji pomiędzy abonentami tego samego segmentu,
- automatyczna rekonfiguracja,
- ograniczona długość ramki,
- możliwość pracy sieci nawet w przypadku awarii któregoś z abonentów,
- "przezroczystość" wymian informacji,
- możliwość tworzenia systemów redundancyjnych.

W sieciach przemysłowych wykorzystujących protokół „krążący żeton” nie wyróżnia się w sposób szczególny żadnego z abonentów. Funkcje związane z zarządzaniem pracą sieci realizowane są kolejno poprzez wszystkich jej abonentów. Transmitowany cyklicznie żeton jest uprawnieniem dla abonenta, który z chwilą jego przejęcia zaczyna pełnić funkcję „Mastery” sieci. Po zrealizowaniu własnych żądań, dotyczących transmisji, abonent przekazuje żeton następcy, który według ustalonego protokołu zaczyna pełnić funkcję urządzenia nadrzędnego. Realizację postulatu zdeterminowanego czasu dostępu do łącza gwarantuje ograniczenie maksymalnego czasu przetrzymywania żetonu przez poszczególnych abonentów sieci. Indywidualnie ustalany maksymalny czas dostępu do łącza oraz stosowanie różnych priorytetów transmitowanych danych stwarza możliwości konfiguracji sieci w sposób stosowny do wymagań czasowych procesu. Protokół „krążący żeton” doczekał się szeregu implementacji firmowych,

jak np. sieci przemysłowe N10, N850 Cegeleca, czy też sieć N80 stosowana przez firmę General Electric[1],[7]. Poszczególne implementacje tego protokołu różnią się pomiędzy sobą dość znacznie, co utrudnia integrację urządzeń pochodzących od różnych producentów za pomocą tego typu sieci.

Protokół „krążący zeton” dobrze nadaje się do realizacji wymian poziomych. Zastosowanie sieci, opartych na tym modelu wymiany informacji, do wizualizacji procesu wymaga uwzględnienia wpływu dodatkowych transmisji wykonywanych przez stację kontrolno-nadzorcą na czas obiegu informacji wykorzystywanej do sterowania procesem. Dla procesów wymagających szybkich transmisji danych pomiędzy poszczególnymi abonentami sieci możliwości wykorzystania tego samego segmentu sieci jednocześnie do prowadzenia wymian poziomych i pionowych mogą być znacznie ograniczone lub wręcz niemożliwe do wykorzystania.

#### 4.3. Sterowniki sieci połowych

We współczesnych systemach sterowania obiektami przemysłowymi występuje stały wzrost liczby sygnałów obiektowych, co pociąga za sobą konieczność instalacji wielu jednostek centralnych. Stanowi to dużą nadmiarowość w sensie mocy obliczeniowych. Fakt ten był jednym z głównych powodów rozpoczęcia prac nad sieciami, które umożliwiłyby bezpośrednie podłączenie do nich modułów wejścia/wyjścia, urządzeń pomiarowych, analogowych modułów wyjściowych itp. Sieci te nazwano sieciami terenowymi lub połowymi. Przykładami takich sieci mogą być sieć Profibus firmy Simens, lub też opracowana przez francuską firmę CELELEC sieć FIP będąca implementacją protokołu komunikacyjnego o tej samej nazwie<sup>[1],[8],[9],[11]</sup>. Głównym założeniem dla sieci połowych jest rozproszone przetwarzanie informacji znajdującej się w rozproszonej bazie danych. Sieć Profibus zrealizowano na bazie protokołu „krążącego zetonu”, natomiast standard FIP-a oparto na nowym jakościowo modelu wymiany informacji typu Producent Dystrybutor Konsument.

Sieć połowa FIP została zaprojektowana w taki sposób, aby zaspokajać wymagania stawiane przed transmisją danych w rozproszonym systemie przemysłowym zarówno w zakresie transmisji, pomiędzy poszczególnymi abonentami procesu (komunikacje poziome), jak i w zakresie zbierania danych z obiektu i przekazywania ich do centrum kontrolnego w celu wizualizacji i archiwizacji, oraz rozsyłania poleceń do poszczególnych abonentów (komunikacje pionowe). Takie właściwości sieci FIP predestynują ją do zastosowania na poziomie stacji kontrolno-nadzorczej segmentu, lub kilku segmentów lokalnej sieci przemysłowej. Podczas projektowania architektury stacji wizualizacyjnej należy uwzględnić oba obszary zastosowania tej sieci.



#### 4.4. Podział funkcji realizowanych przez sterownik sieci przemysłowej

Sterownik sieci przemysłowej powinien, z jednej strony, zaspokajać wymagania stawiane przez standard zastosowanej sieci przemysłowej, z drugiej zaś strony powinien umożliwiać wykorzystywanie danych i usług dostarczanych przez sieć innym modułom systemu. Propozycja struktury interfejsu dla sieci FIP stanowi dobrą ilustrację problemów występujących podczas projektowania sterowników sieci przemysłowych.

Karta sterownika sieci FIP, wyposażona w układ procesora sieci, może spełniać funkcję zarówno arbitra magistrali, jak również producenta i konsumenta danych transmitowanych siecią. Oprogramowanie sterownika sieci powinno stwarzać możliwość wykorzystania obu tych własności. Sterownik powinien umożliwiać pracę wyłącznie w trybie producenta/konsumenta danych. W takim przypadku arbitrem magistrali jest inna stacja operatorska lub sterownik przemysłowy. Drugim rodzajem pracy jest tryb gorącej rezerwy, w którym sterownik ogranicza się jedynie do wymiany informacji, przejmując jednocześnie funkcje arbitra magistrali w przypadku wypadnięcia stacji z pracującym arbitrem. Trzeci tryb pracy umożliwia jednoczesną pracę w charakterze arbitra, jak i pełnienie funkcji producenta/konsumenta danych. Tryb ten pozwala na elastyczne dopasowywanie scenariusza wymian sieciowych do bieżących potrzeb stacji wizualizacyjnej. Tryb ten jednak nie zawsze jest możliwy do zastosowania, np. w przypadku wykorzystania kilku stacji operatorskich umieszczonych na jednym segmencie sieci, lub też w przypadku konieczności umiejscowienia funkcji arbitra sieci w sterowniku przemysłowym.

Karta sterownika sieci FIP wymaga własnego oprogramowania komunikacyjnego umieszczonego w pamięci karty. Oprogramowanie to powinno współpracować ze sterownikiem sieci, który ze względu na konieczność wykorzystywania bezpośrednich odwołań do sprzętu oraz obsługi przerwań powinien zostać napisany jako oprogramowanie niskiego poziomu. Trzecim elementem sterownika sieci FIP powinna być część interfejsu pozwalającego na współpracę z jądrem systemu. Część ta może zostać zrealizowana w sposób obiektowy (w sensie OOP), sam zaś moduł sterownika sieci FIP powinien posiadać ogólne właściwości sterowników sieci przemysłowych.

## 5. Propozycja struktury systemu kontrolno-nadzorczego

Do rozwiązania problematyki wizualizacji szerokiej gamy obiektów automatyki przemysłowej wydaje się celowe zastosowanie takich metod, które cechuje maksymalna elastyczność budowy stacji kontrolno-nadzorczej, oraz łatwość adaptacji ze względu na różnorodność ob-

sługiwanych urządzeń i systemów. Konstrukcja takiego oprogramowania powinna uwzględniać następujące cechy obiektów automatyki:

- *różnorodność rozwiązań technicznych poszczególnych układów* automatyki przemysłowej dostarczanych przez różnych producentów sprzętu, jednak z uwzględnieniem tych cech funkcjonalnych, które są wspólne dla danego typu urządzeń,

- *brak jednolitego standardu w zakresie rozwiązań komunikacyjnych* dla urządzeń automatyki wpływa na konieczność obsługi wielu popularnych standardów komunikacyjnych, jak również możliwość łatwej adaptacji dostarczanego oprogramowania dla realizacji wymogów nietypowych protokołów teletransmisyjnych,

- *różnorodność form oraz stopnia złożoności realizowanych obiektów* stacji kontrolno-nadzorczych. Oprogramowanie powinno z jednej strony dawać projektantowi aplikacji dużą swobodę w doborze środków, jak również pozwalać na łatwe kopiowanie i przenoszenie raz utworzonych elementów w celu zastosowania ich w wizualizacji pracy innych urządzeń tego samego typu.

- *konieczność dopasowania poziomu bezpieczeństwa, klasy zabezpieczeń i stopnia redundancji systemu* stosownie do wymogów obsługiwanej instalacji. Oprogramowanie powinno uwzględniać zarówno możliwości tworzenia prostych układów bez redundancji, realizowanych na pojedynczym stanowisku operatorskim, jak i układów złożonych, gdzie część funkcji zostanie zdublowana, a część zadań rozdzielona pomiędzy wiele stanowisk obsługujących dany proces.

W celu realizacji zdefiniowanych powyżej postulatów zaproponowano rozwiązanie procesu wizualizacji, obejmujące modułową budowę stacji kontrolno-nadzorczej, oraz obiektową (w sensie OOP) strukturę poszczególnych modułów oprogramowania.

*Modułowa struktura* stacji ułatwi dobór konfiguracji stacji kontrolno-nadzorczej stosownie do potrzeb i wielkości obsługiwanego procesu. Struktura ta pozwoli również na rozdzielanie funkcji realizowanych przez poszczególne moduły, które po określeniu interfejsu dla przekazywanych danych będą mogły być stosowane wymiennie. Podział zadań realizowanych przez system pomiędzy poszczególne moduły funkcjonalne wymaga określania „zakresu kompetencji” każdego z modułów oraz interfejsu wymiany informacji pomiędzy modułami. Dzięki takiemu podejściu moduły np. wizualizacji graficznej będą mogły pracować niezależnie od modułów obsługujących komunikację sieciową, co pozwoli na łatwą adaptowalność nietypowych standardów komunikacyjnych oraz uprości obsługę nowych typów urządzeń. Wyodrębnienie funkcji systemu i zamknięcie ich w modułach ułatwi rozwiązywanie problemów związanych z koniecznością rozdziału, dublowania czy też kopiowania funkcji systemu pomiędzy różne stanowiska obsługujące proces.

Wykorzystanie metod *programowania obiektowego* ułatwi opisywanie systemów automatyki, które ze swej natury charakteryzują się również strukturą obiektową. Podejście obiektowe

we upraszcza pracę nie tylko projektanta oprogramowania służącego do wizualizacji, ale również skraca drogę do realizacji konkretnej aplikacji przemysłowej, umożliwiając łatwą adaptację dostarczanych przez system obiektów programowych (w sensie OOP) do potrzeb stawianych przez konkretne fizyczne obiekty automatyki przemysłowej. Cechy programowania obiektowego, takie jak dziedziczenie i polimorfizm, odciążają osobę pragnącą modyfikować właściwości istniejącego systemu od konieczności żmudnego śledzenia źródłowego kodu programu. Znajomość hierarchii zastosowanych klas i metod wystarczy do tworzenia własnych elementów wizualizacji procesu przemysłowego. Nowe typy obiektów mogą powstawać jako złożenie właściwości funkcjonalnych wielu typów podstawowych lub też być efektem dopasowania zachowań dostarczanych klas programu do wymogów stawianych przez realia konkretnej aplikacji. Łatwość zastosowania stworzonych w ten sposób klas potomnych uprości adaptację oprogramowania dla realizacji konkretnej klasy zadań, jak również ułatwi dalszą modyfikację systemu.

Wadą stosowania programowania obiektowego jest konieczność przekładu kodu obiektowego języka wyższego poziomu programowania na kod assemblera procesora realizującego program, który jest ze swej natury kodem nieobektowym. Przekład ten może spowodować spadek optymalności realizowanego programu i pogorszenie jego parametrów czasowych w stosunku do tych samych funkcji realizowanych przez analogiczny program napisany w oparciu o klasyczne podejście proceduralne.

Wymienione wady ograniczają zastosowanie do realizacji oprogramowania stacji kontrolno-nadzorczej czysto obiektowych języków programowania (jak np. Smalltalk). Dla realizacji procesu wizualizacji lepszym rozwiązaniem byłoby użycie języka z jednej strony dobrze oddającego obiektową strukturę obsługiwanego instalacji, z drugiej zaś strony pozwalającego na bezpośredni dostęp do specjalizowanych urządzeń takich jak np. karty sieci przemysłowej, oraz zapewniającego jak najszybszą obsługę programową takich elementów systemu jak np. alarmy i zdarzenia, których obróbka powinna dokonywać się na bieżąco.

Prezentowane postulaty wydaje się spełniać język C++ klasyfikowany jako hybrydowy język obiektowy. Język ten pozwala zarówno na wykorzystanie udogodnień języka obiektowego, jak również na dostęp niskiego poziomu do zasobów sprzętowych. Dodatkową zaletą języka C++ jest stosunkowo wysoka efektywność przekładanego kodu oraz możliwość stosowania fragmentów napisanych w kodzie maszynowym.

Istotne z punktu widzenia konstruowanego oprogramowania staje się wydzielenie fragmentów kodu realizowanych w sposób obiektowy, fragmentów realizowanych proceduralnie oraz fragmentów odwołujących się bezpośrednio do sprzętu realizowanych jako sterowniki urządzeń zewnętrznych.

### 5.1. Jądro programu

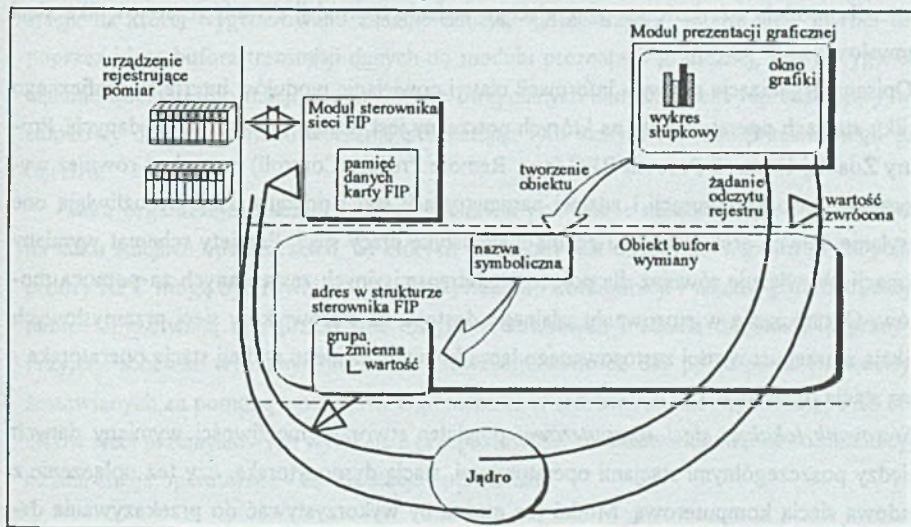
Jądro programu pełni funkcję kręgosłupa całej aplikacji. Umożliwia ono osadzanie kolejnych modułów funkcjonalnych oraz pozwala na dostęp do oferowanych przez nie danych i usług. Powinno ono zapewniać tak komunikację pomiędzy poszczególnymi modułami, w tym również komunikację pomiędzy poszczególnymi stacjami abonenckimi, jak również oferować grupę funkcji systemowych pozwalających na rejestrację działań obsługi, diagnostykę pracy poszczególnych modułów, wykrywanie sytuacji nietypowych i wadliwego działania systemu. Jądro programu powinno zawierać wskazania do wszystkich istniejących w systemie modułów, dostarczać funkcji diagnostycznych i systemowych oraz umożliwiać dostęp do danych dla obiektowych buforów wymiany informacji.

### 5.2. Obiektowe bufor wymiany informacji

Podstawową funkcją jądra systemu wizualizacji jest przekazywanie informacji pomiędzy poszczególnymi modułami systemu. Ze względu na możliwości rekonfiguracji poszczególnych modułów wskazane jest, aby wymiana ta nie odbywała się wprost, lecz za pomocą buforów pozwalających ukryć szczegóły implementacyjne, dotyczące budowy poszczególnych modułów. Moduły powinny nie tylko być niezależne od ich wzajemnego umieszczenia w systemie, ale również powinny być wolne od konieczności znajomości wewnętrznej struktury danych modułu, do którego są przesyłane lub pobierane informacje. Taka organizacja jądra systemu możliwa jest do zrealizowania dzięki wykorzystaniu „inteligentnych” buforów konstruowanych w oparciu o klasy, których hierarchia pozwala na przejście od przekazywania informacji opisywanej w sposób ogólny do specjalistycznych funkcji komunikacyjnych właściwych dla odpowiednich modułów systemu. Zasadę przekazywania informacji przez obiekt bufora można zilustrować na przykładzie pobierania informacji przez element graficzny wykres słupkowy - bargraf bazujący na danych pomiarowych transmitowanych siecią FIP. Algorytm wymiany danych prezentuje rysunek 1. Dane pomiarowe z urządzenia odczytywane są przez sterownik sieci FIP i umieszczane w pamięci karty sieciowej. Odczyt danych może występować cyklicznie, być wykonywany na żądanie lub zachodzić w reakcji na wystąpienie określonej sytuacji zgodnie ze specyfikacją transmisji określającą funkcjonowanie sterownika sieci FIP.

Moduł graficznej prezentacji danych zawiera, w jednym ze swoich okien, wykres słupkowy - bargraf prezentujący stan rejestru pomiarowego. W momencie utworzenia obiektu graficznego tworzona jest również instancja klasy bufora wymiany informacji wraz z nazwą symboliczną identyfikującą prezentowany rejestr. Poprzez odwołanie do jądra systemu klasa bufora odwołuje się do modułu sterownika sieci FIP i tam stara się znaleźć daną o nazwie symbolicznej zgodnej z nazwą podaną przez obiekt prezentacji graficznej. W przypadku poprawnej

identyfikacji rejestru obiekt bufora zapamiętuje dane dotyczące jego lokalizacji wewnątrz sterownika sieci FIP. Danymi tymi są Grupa, Zmienna i Wartość oraz położenie sterownika sieci FIP. Późniejsze odwołania klasy buforowej do sterownika sieci będą wykonywane z użyciem tak zdefiniowanej struktury adresowej. Opisane czynności dotyczyły inicjalizacji obiektu bufora danych. Podczas normalnej pracy programu okno grafiki będzie wymagało cyklicznej aktualizacji danych pomiarowych i co za tym idzie, odświeżania rysowanego wykresu słupkowego. Żądanie odczytu danych, kierowane do klasy buforowej, zostanie z użyciem jądra i bezpośredniej struktury adresowej skierowane do modułu sterownika sieci FIP. W przypadku gdy sterownik posiada jakieś nowe dane dotyczące rejestru (pamiętany jest indeks poprzedniego odczytu przez instancję klasy bufora), dane te umieszczane są w klasie bufora, skąd pobiera je obiekt graficzny, który dokonuje następnie samoodtworzenia w oparciu o nową wartość pomiaru. Jeżeli sterownik sieci FIP nie posiada nowych informacji dotyczących danego rejestru, zwraca tę informację za pomocą klasy bufora informując tym samym obiekt graficzny, iż nie zachodzi konieczność odtworzenia nowej wartości.



Rys. 1. Pobieranie informacji transmitowanej siecią FIP przez moduł prezentacji graficznej  
Fig. 1. Presentation graphics data getted from FIP network

Poszczególne klasy buforów danych różnią się między sobą tak pod względem sposobu adresacji danych wynikającej z rodzaju modułu, który odpowiada za dostarczanie informacji, jak i pod względem specyfiki przekazywanych informacji. Ich budowa wewnętrzna powinna być dostosowana do obsługiwanego sterownika sieci przemysłowej.

### 5.3. Koncepcja zdalnej kontroli procesu z wykorzystaniem lokalnych sieci komputerowych

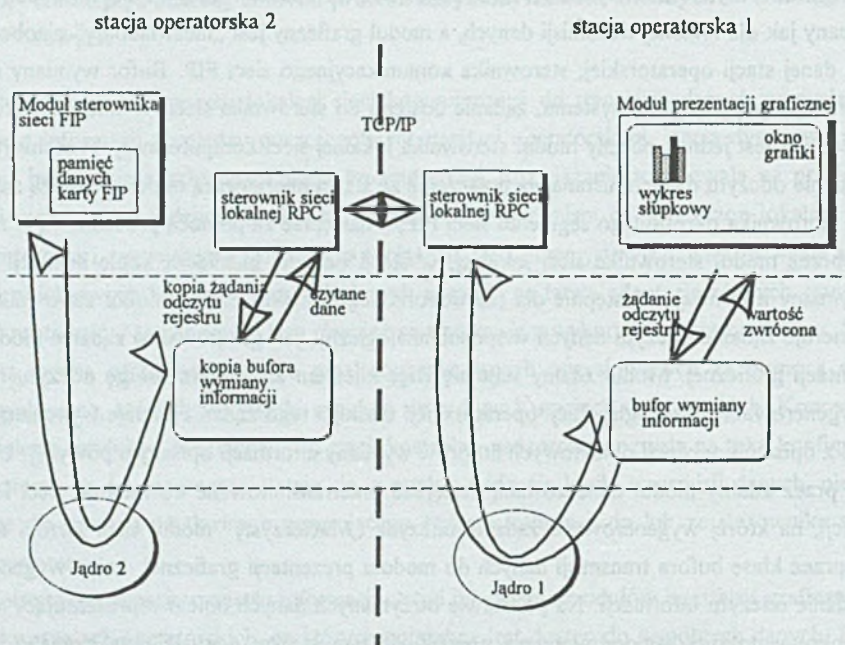
Wykorzystanie segmentu lokalnej sieci komputerowej do transmisji danych pozwala na wymianę informacji pomiędzy poszczególnymi stacjami operatorskimi, stacją dyspozytorską, czy też połączenie z zakładową siecią komputerową. Rozwiązanie to pozwala na przejście części mniej odpowiedzialnych funkcji związanych z transmisją danych przez lokalną sieć komputerową i zmniejszenie tym samym obciążenia sieci przemysłowych. Zastosowanie koncepcji obiektowych buforów transmisji danych pozwala na łatwą adaptację różnych standardów i protokołów transmisji, w tym również na transmisję z wykorzystaniem modemów. Sieć lokalną można wykorzystywać do przekazywania danych pozyskiwanych za pomocą sieci przemysłowych, których sterowniki znajdują się w innych stacjach operatorskich. Koncepcja wydzielenia modułów funkcjonalnych stacji kontrolno-nadzorczej pozwala na taką konfigurację systemu, że jej rozproszenie staje się, z punktu widzenia logiki transmisji danych, niewidoczne dla modułu użytkowego generującego żądanie transmisji do lub ze sterownika sieci przemysłowej

Opisana organizacja przesyłu informacji ułatwi powielanie modułów interfejsu graficznego na kilku stacjach operatorskich, na których potrzebny jest dostęp do wspólnych danych. Procedury Zdalnej Kontroli Procesu RPC (ang. Remote Process Control) mogą być również wykorzystywane do konfiguracji i zdalnej parametryzacji stacji operatorskiej. Umożliwiają one przesyłanie danych archiwalnych i zdalną diagnostykę pracy sieci. Przyjęty schemat wymiany informacji obowiązuje również dla połączeń teletransmisyjnych zestawianych za pomocą modemów. Ograniczenia w stosowaniu zdalnego dostępu do sterowników sieci przemysłowych wynikają z przepustowości zastosowanego łącza komunikacyjnego na linii stacja operatorska - zdalna stacja operatorska.

*Sterownik lokalnej sieci komputerowej* powinien stworzyć możliwości wymiany danych pomiędzy poszczególnymi stacjami operatorskimi, stacją dyspozytorską, czy też połączenie z zakładową siecią komputerową. Moduł ten można by wykorzystywać do przekazywania danych, pozyskiwanych za pomocą sieci przemysłowych, których sterowniki znajdują się w innych stacjach operatorskich. Konstrukcja sterownika pozwala na taką jego konfigurację, że jego działanie staje się niewidoczne dla modułu użytkowego generującego żądanie transmisji danych do lub ze sterownika sieci przemysłowej. Efekt taki można osiągnąć dzięki wykorzystaniu klas buforów danych, ukrywających szczegóły związane z przekazywaniem informacji. Przykładem takiego wykorzystania może być mechanizm zdalnego odczytu rejestru pomiarowego przez element prezentacji graficznej - wykres słupkowy. Schemat wymiany informacji zaprezentowany został na rys. 2. Moduł prezentacji graficznej tworzy instancję klasy bufora wymiany danych, a następnie generuje żądanie odczytu rejestru, którego wartość ma mu po-

służyć do odrysowania wykresu słupkowego. Cała ta operacja odbywa się w sposób analogiczny jak dla lokalnej transmisji danych, a moduł graficzny jest „nieświadomy” nieobecności, na danej stacji operatorskiej, sterownika komunikacyjnego sieci FIP. Bufor wymiany danych generuje, poprzez jądro systemu, żądanie dostępu do sterownika sieci. W miejsce sterownika sieci FIP jest jednak obecny moduł sterownika lokalnej sieci komputerowej. Przechwytuje on żądanie odczytu danych, ustanawia połączenie ze stacją operatorską odpowiedzialną za obsługę sterownika określonego segmentu sieci FIP, a następnie za pomocą procedur RPC tworzy, poprzez moduł sterownika sieci lokalnej, w stacji odległej bliźniaczą kopię instancji bufora wymiany informacji. Następnie dla tak utworzonego obiektu zdalny moduł sterownika sieci generuje żądanie odczytu danych w sposób analogiczny, jak generował to żądanie moduł prezentacji graficznej. Moduł zdalny staje się więc klientem zlecającym usługę odczytu rejestru wygenerowaną na odległej stacji operatorskiej. Obsługa tego żądania zostaje wykonana zgodnie z opisem koncepcji obiektowych buforów wymiany informacji opisanym powyżej. Uzyskane przez zdalny moduł dane zostaną następnie przetransmitowane do modułu sieci lokalnej stacji, na której wygenerowano żądanie odczytu. „Macierzysty” moduł sieci zwróci te dane poprzez klasę bufora transmisji danych do modułu prezentacji graficznej, który wygenerował żądanie odczytu informacji. Na podstawie otrzymanych danych obiekt reprezentujący wykres słupkowy dokona samoodtworzenia prezentując tym samym wartość przekazanego zdalnie rejestru.

Taka organizacja przesyłu informacji ułatwia powielanie modułów interfejsu graficznego na kilku stacjach operatorskich, na których potrzebny jest dostęp do wspólnych danych. Procedury RPC mogą być również wykorzystywane do konfiguracji i zdalnej parametryzacji systemu. Umożliwiają one przesyłanie danych archiwalnych i zdalną diagnostykę pracy sieci. Przyjęty schemat wymiany informacji obowiązuje również dla połączeń teletransmisyjnych zestawianych za pomocą modemów. Ograniczenia w stosowaniu zdalnego dostępu do sterowników sieci przemysłowych wynikają z przepustowości zastosowanego łącza komunikacyjnego na linii stacja operatorska - zdalna stacja operatorska.



Rys.2. Zdalna transmisja danych z wykorzystaniem sterowników lokalnej sieci komputerowej  
Fig.2. Remote data transmission with using Local Area Network

## 6. Podsumowanie

Treść opracowania jest efektem praktycznych doświadczeń i badań Autorów nad firmowymi rozwiązaniami oprogramowania stacji kontrolno-nadzorczych. Ponadto podczas trwającej od kilku lat praktyki związanej z aplikacją stacji wizualizacyjnych dla konkretnych systemów automatyki przemysłowej zauważono stopniową ewolucję od systemów zamkniętych, tworzonych w oparciu o zwartą konstrukcję procedur programu, poprzez systemy z wyróżnioną strukturą modułów, aż do systemów udostępniających użytkownikowi możliwości zastosowania własnych fragmentów oprogramowania. W niniejszym opracowaniu Autorzy starali się zwrócić uwagę na te właściwości procesów, które są wspólne szerszej gamie zastosowań stacji kontrolno-nadzorczych. Próba klasyfikacji typów protokołów stosowanych dla sieci przemysłowych ma za zadanie zwrócić uwagę Czytelnika na problematykę związaną z właściwym wyborem sieciowej struktury systemu rozproszonego stosownie do potrzeb aplikacji. Przedstawiona w pracy propozycja struktury stacji kontrolno-nadzorczej stanowi rozwinięcie metod modularyzacji oprogramowania wizualizacyjnego o elementy obiektowej budowy mo-



dułów. Programowanie obiektowe zdaje się z jednej strony upraszczać i przybliżać opis instalacji automatyki przemysłowej, jak też, z drugiej strony, ułatwia modyfikację i adaptację cech funkcjonalnych oprogramowania stacji kontrolno-nadzorczej z uwzględnieniem wymogów konkretnej aplikacji.

## LITERATURA

- [1] Praca zbiorowa: Rozproszone systemy komputerowe. Pronet, Gliwice 1994
- [2] Grzywak A., Kwiecień A.: Perspektywy rozwoju zastosowań sieci komputerowych w górnictwie. *Mechanizacja i Automatykacja Górnictwa* nr 5, 6/94
- [3] Grzywak A., Kwiecień A.: Rozproszone systemy sterowania i zarządzania procesami technologicznymi. *Mechanizacja i Automatykacja Górnictwa* nr 8/94.
- [4] Grzywak A., Kwiecień A.: Sieci komputerowe w systemach sterowania i zarządzania w górnictwie. Konferencja Międzynarodowa ICAMC'95 World Mining Congress.
- [5] Cupek R.: Wizualizacja systemów automatycznego sterowania. *Zeszyty Naukowe Politechniki Śląskiej*, s. Informatyka z. 23, Gliwice 1993
- [6] Cupek R.: Metody hierarchizacji rozproszonych procesów przemysłowych. *Zeszyty Naukowe Politechniki Śląskiej*, s. Informatyka z. 28, Gliwice 1995
- [7] Bigewski Z.: Optymalizacja pracy sieci przemysłowych. *Zeszyty Naukowe Politechniki Śląskiej*, s. Informatyka z. 28, Gliwice 1995
- [8] Kwiecień A.: Sieć rozległa FIP. *Zeszyty Naukowe Politechniki Śląskiej*, s. Informatyka z. 24, Gliwice 1993
- [9] Kwiecień A., Gaj P.: Sieć FIP, wstęp do analizy czasowej. *Zeszyty Naukowe Politechniki Śląskiej*, s. Informatyka z. 28, Gliwice 1995
- [10] Kwiecień A., Gaj P., Grzywak A., Mrówka Z.: Rozwiązania sprzętowe i programowe sieci przemysłowej FIP. *Zeszyty Naukowe Politechniki Śląskiej*, s. Informatyka z. 30, Gliwice 1996
- [11] Chapuis H.: Sieć terenowa FIP. *Zeszyty Naukowe Politechniki Śląskiej*, s. Informatyka Zeszyt specjalny, Gliwice 1992

Recenzent: Dr hab.inż. Adam Mrózek  
Prof.nadzw. Politechniki Śląskiej

Wpłynęło do redakcji 6 grudnia 1996 r.

## Abstract

This paper try to show some practical problems connected with constructing and using communication modules for control stations. There were described functions of this modules, requirements of timing and data transmit, and changes made in constructing this modules from some years.

First are described elements of visualization process, and their possibilities. Next there are described some industrial networks and protocols i.e. MASTER-SLAVE, token-bus, PDC, and try to compare their requirements and ability of using this kinds of networks in automatic control systems.

Authors try to describe their classification of network and visualizations modules, and show proposition of realization communication modules using object-oriented functions and making connections between them in decentralized industrial control station.