

Komputerowe układy sterowania

Dr inż. Janusz Baran

LITERATURA

- Ackermann J.: *Regulacja impulsowa*, WNT, 1976.
- Orłowski H.: *Komputerowe układy automatyki*, WNT, 1987.
- Niederliński A.: *Systemy komputerowe automatyki przemysłowej*, Tom 1. *Sprzęt i oprogramowanie*, 1984, Tom 2. *Zastosowania*, WNT, 1985.
- Trybus L.: *Regulatory wielofunkcyjne*, WNT, 1992.
- Brzózka J.: *Regulatory cyfrowe w automatyce*, Wyd. MIKOM, 2002.
- Brzózka J.: *Regulatory i układy automatyki*, Wyd. MIKOM, 2004.
- Åström K.J., Wittenmark B.: *Системы управления с ЭВМ (Computer Controlled Systems)*, Mir, 1987.
- Tatjewski P.: *Sterowanie zaawansowane obiektów przemysłowych. Struktury i algorytmy*, Akadem. Oficyna Wyd. EXIT, 2002.
- Driankov D., Hellendoorn H., Reinfrank M.: *Wprowadzenie do sterowania rozmytego*, WNT, 1996.
- Piegat A.: *Modelowanie i sterowanie rozmyte*, Akademicka Oficyna Wydawnicza EXIT, 1999.
- Niederliński A., Mościński J., Ogonowski Z.: *Regulacja adaptacyjna*. PWN, Warszawa 1995.
- Söderström T., Stoica P.: *Identyfikacja systemów*, PWN, 1997.

Zastosowania szczególne:

- Wójciak A.: *Mikroprocesory w układach przekształtnikowych*, WNT, 1992
- Hejmo W., Koziół R.: *Systemy mikroprocesorowe w automatyce napędu przekształtnikowego*, WNT, 1994

Skrypty:

- Beliczyński B., Koziński W.: *Wprowadzenie do regulacji cyfrowej*, Wyd. Pol. Warszawskiej, 1987.
- Englot A.: *Sterowanie dyskretne*, Wyd. Pol. Krakowskiej, 1999.

Sterowniki PLC

- Legierski T., Kasprzyk J., Wyrwał J., Hajda J.: *Programowanie sterowników PLC*, Wyd. Pracowni Komp. J.Skalmierskiego, 1998.
- Kasprzyk J.: *Programowanie sterowników przemysłowych*, WNT, 2006.
- Mikulczyński T., Samsonowicz Z.: *Automatyzacja dyskretnych procesów produkcyjnych*, WNT, 1997.
- Seta Z.: *Wprowadzenie do zagadnień sterowania. Wykorzystanie programowalnych sterowników logicznych PLC*, Wyd. MIKOM, 2002.

Czasopisma

Pomiary-Automatyka-Kontrola
Pomiary-Automatyka-Robotyka
Biuletyn automatyki ASTOR

Zarys rozwoju komputerowych układów sterowania

Źródło: Åström K.J., Wittenmark B.: *Computer Controlled Systems*,
3rd ed., Prentice Hall

Okres pionierski 1955-1962	Komputery pracują w trybie doradczym <i>off-line</i> - drukowanie instrukcji dla operatora procesu, przemysł chemiczny, hutniczy, elektrownie, do 1962 ok. 150 systemów, wprowadzenie przerwań, dodanie funkcji cyfrowych do analogowych regulatorów PID 1959 rafineria Texaco w Texasie - zastosowanie do sterowania nadrzędnego reaktora polimeryzacyjnego w trybie <i>on-line</i> MTBF do 100h
Wprowadzanie cyfrowego sterowania bezpośredniego 1962-1967	1962 zakłady chemiczne ICI Anglia, komputer Ferranti Argus 200 do sterowania bezpośredniego 129 zaworami, pomiar 224 zmiennych, wprowadzenie monitorów zamiast drukarek MTBF 1000h, +100ms, *1ms
Era minikomputerów 1967-1972	Rozwój technologii układów scalonych, komputery mniejsze, tańsze (koszt systemu \$10000-100000), 1965 pierwszy minikomputer przemysłowy DEC PDP-8, do 1975 ok. 50 tys. systemów (wzrost x10 w ciągu 5 lat), pulpity operatorskie, 1968 pierwszy sterownik PLC (Modicon) – General Motors, Bedford MTBF 20000h, +1ms, *10ms

Komputerowe układy sterowania

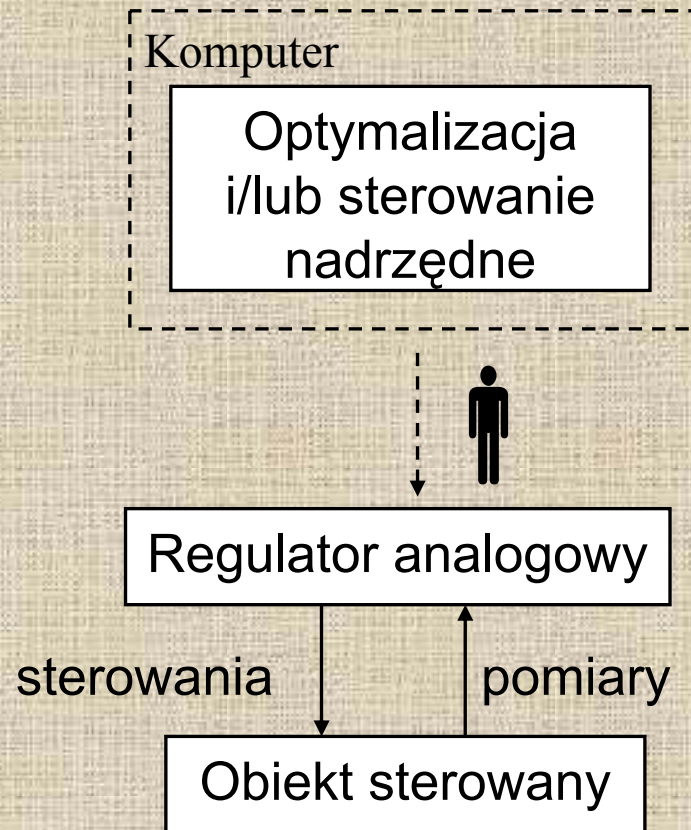
Era mikrokomputerów 1972-1980	<p>Technologia VLSI, 1972 Intel 4004 (Hoff), cena mikrochipa \$10-50, mikrokomputery zastępują pojedyncze pętle regulacji analogowej, lawinowy wzrost liczby aplikacji (mikrosystemy DDC), systemy interakcyjne i graficzne, układy wieloprocesorowe</p> <p>Rozwój zdecentralizowanych systemów hierarchicznych, np. dwupoziomowy Honeywell TDC 2000 od 1975 (<i>Total Distributed Control</i>, ponad 80 tys. układów w 1980), pierwsze regulatory cyfrowe BC (Basic Controllers) – 16-bit., modułowe, 8 kanałów, 28 algorytmów (10 wariantów PID), szeregową magistrala Data Highway</p>
Powszechne stosowanie sterowania cyfrowego od 1980	<p>Rozpowszechnienie sterowania cyfrowego w sprzęcie powszechnego użytku. Procesory specjalizowane DSP (architektura optymalizowana pod kątem przetwarzania sygnałów w czasie rzeczywistym), przetwarzanie równoległe. Oprogramowanie do wspomaganie projektowania regulacji.</p> <p>Rozproszone wielopoziomowe systemy przemysłowe (sterowniki obiektowe-sterowniki nadrzędne-komputery do zadań optymalizacji i zarządzania), elastyczne układy modułowe, sieci CAN, <i>fieldbus</i> LAN, 1984 algorytm samostrojenia EXACT Foxboro, algorytmy optymalne i adaptacyjne, predykcyjne, <i>robust</i>, systemy sztucznej inteligencji i eksperckie</p> <p>MTBF dla CPU = czas życia systemu, *0.1-0.001ms</p>



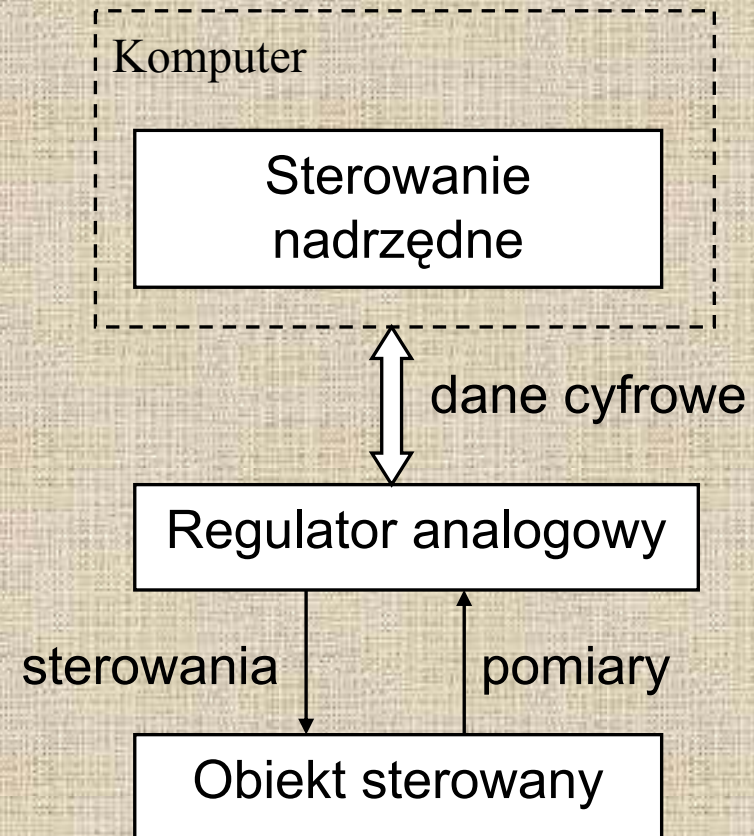
Rys. Pierwszy mikroprocesor Intel 4004 (1972)



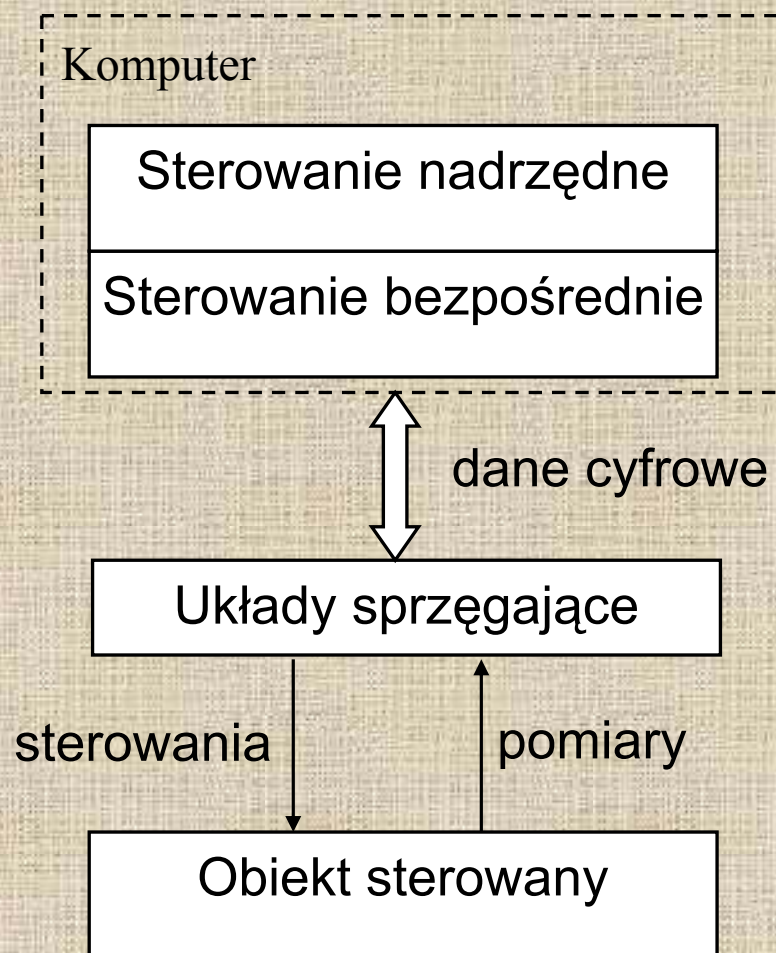
Rys. Regulatory BC (Basic Controllers)
rozproszonego systemu sterowania Honeywell
TDC 2000 (1975)



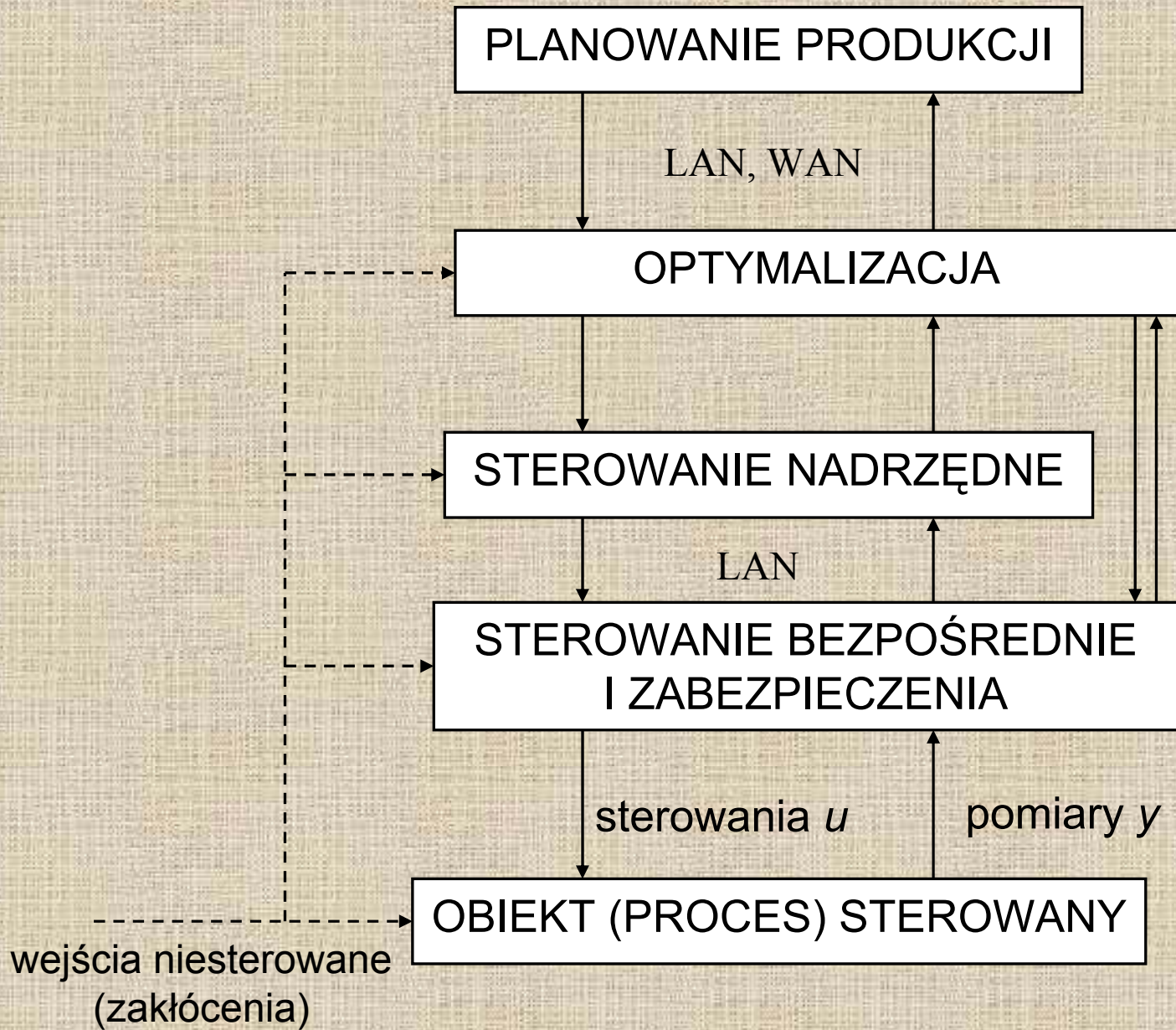
Sterowanie cyfrowe w trybie doradczym (off-line)



Sterowanie cyfrowe w trybie nadrzędnym (on-line)
(DSC – *Digital Supervisory Control*)

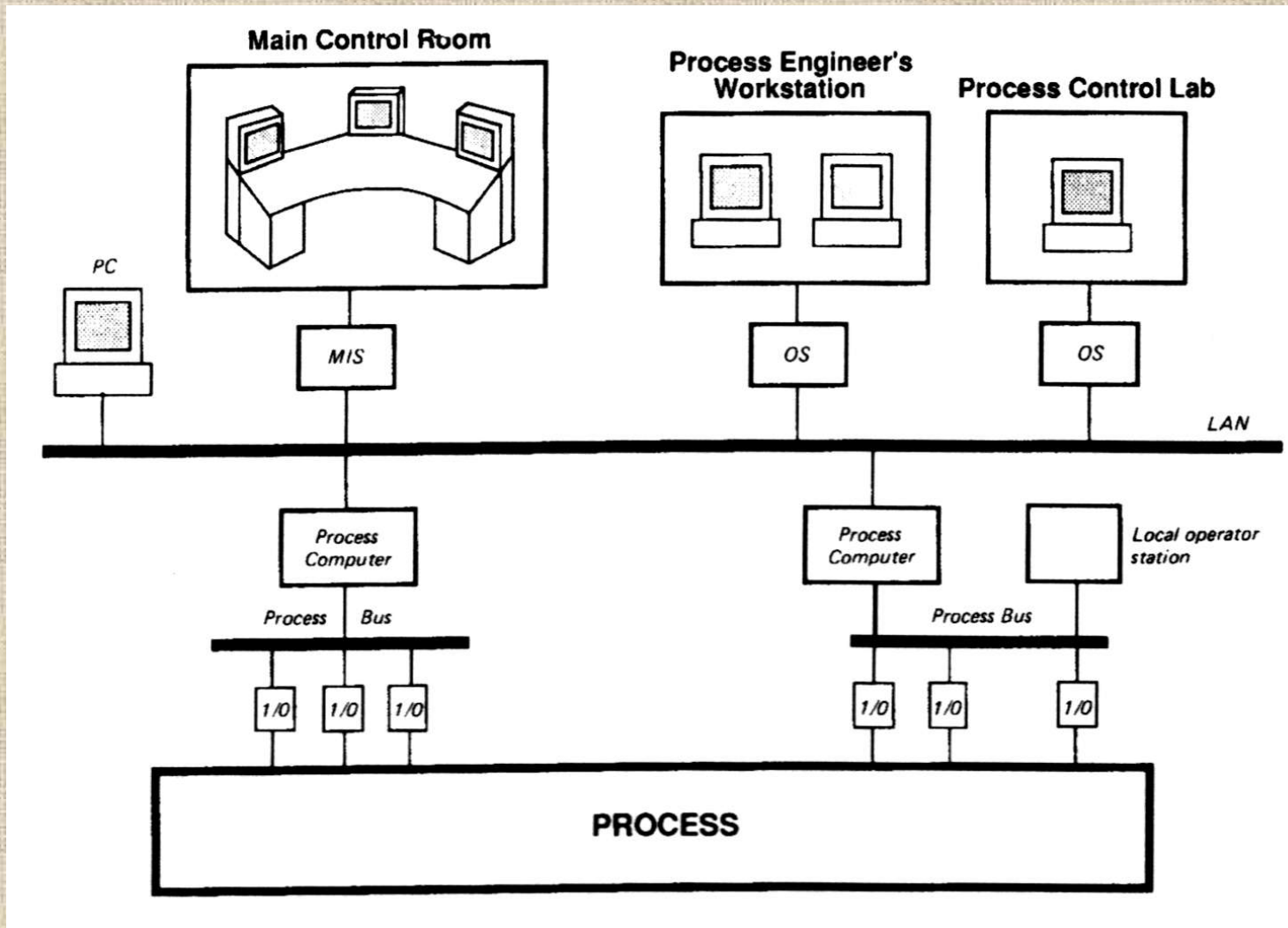


Sterowanie cyfrowe w trybie bezpośrednim
(DDC – *Digital Direct Control*)



Rys. Warstwowa (hierarchiczna) struktura układu sterowania

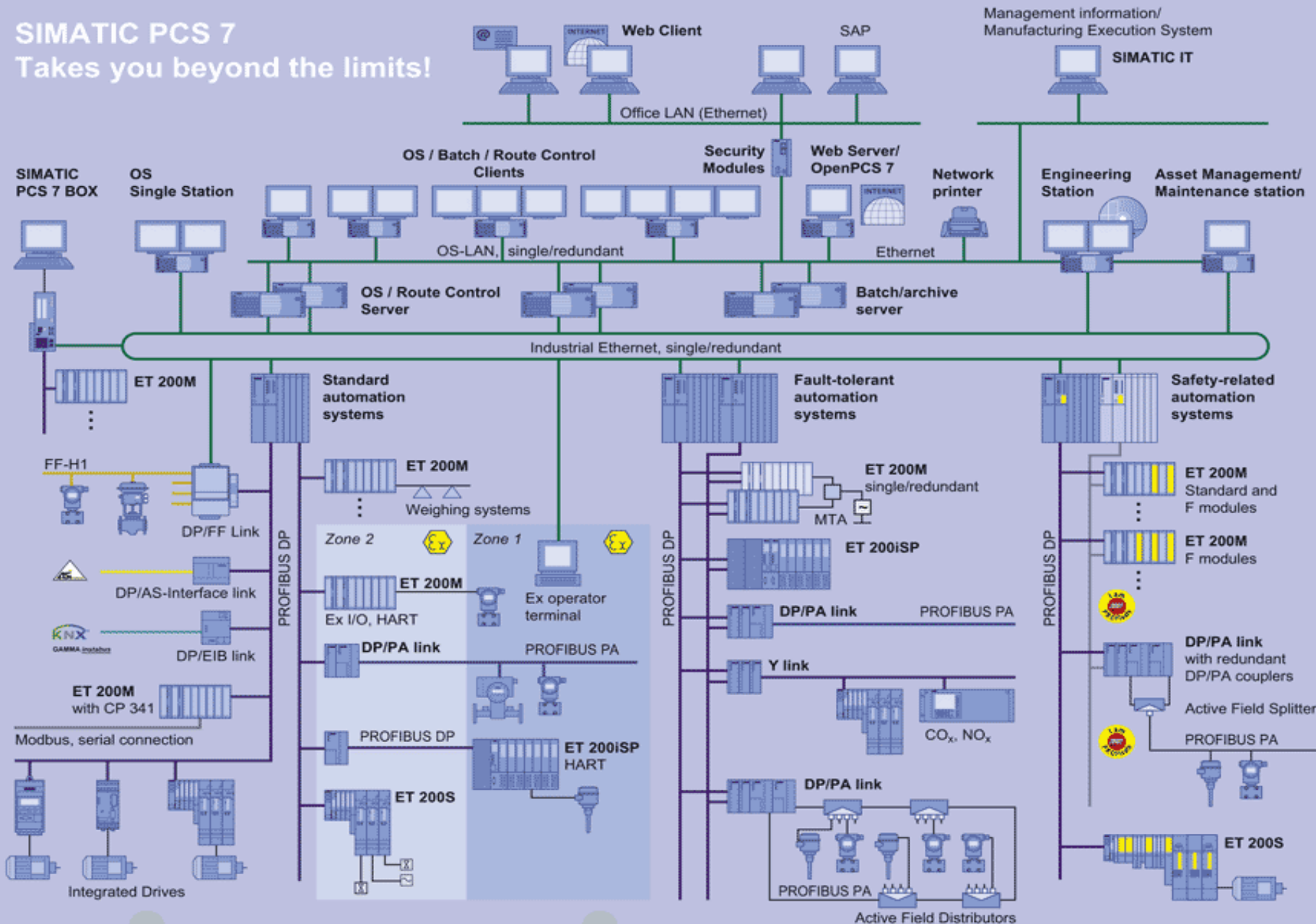
Warstwa systemu sterowania	Podstawowe zadanie	Typowy okres interwencji
regulacja bezpośrednia i zabezpieczenia	stabilizacja, sterowanie nadążne lub programowe, nadzorowanie procesu	ułamki sekundy, sekunda
regulacja nadrzędna	nadzorowanie jakości procesu – zaawansowana regulacja kluczowych zmiennych (z reguły z ograniczeniami)	minuta
optymalizacja	maksymalizacja bieżących efektów ekonomicznych	godzina
zarządzanie produkcją	maksymalizacja efektów ekonomicznych w dłuższych okresach	doba, kilka dni



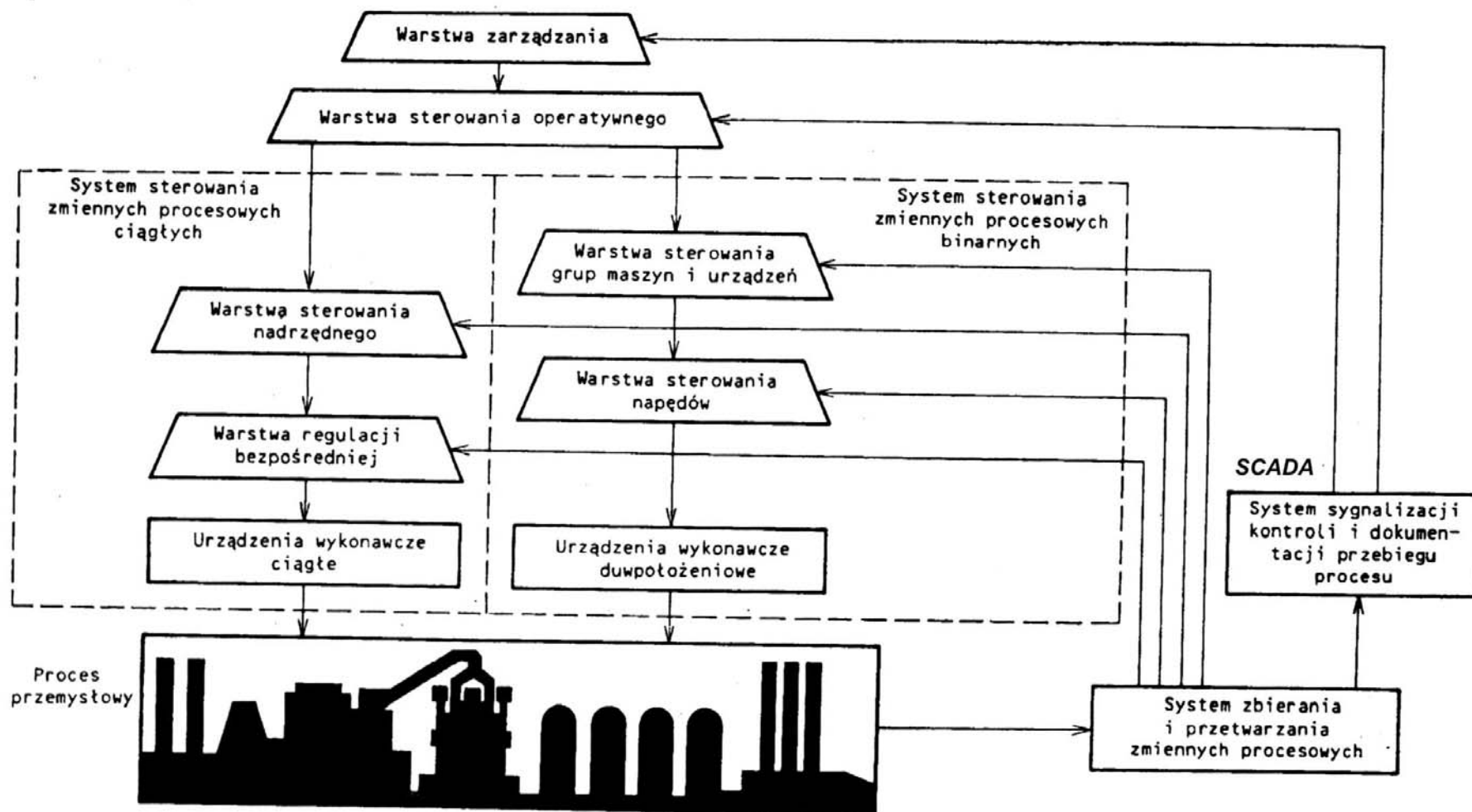
Rys. Architektura rozproszonego systemu sterowania:
I/O – układy we/wy z dopasowaniem sygnału (izolacja galwaniczna) i interfejsem (lub proste sterowniki), *Process Bus* – szyna procesowa, obecnie często oparta na CAN – *Controller Area Network*

Komputerowe układy sterowania

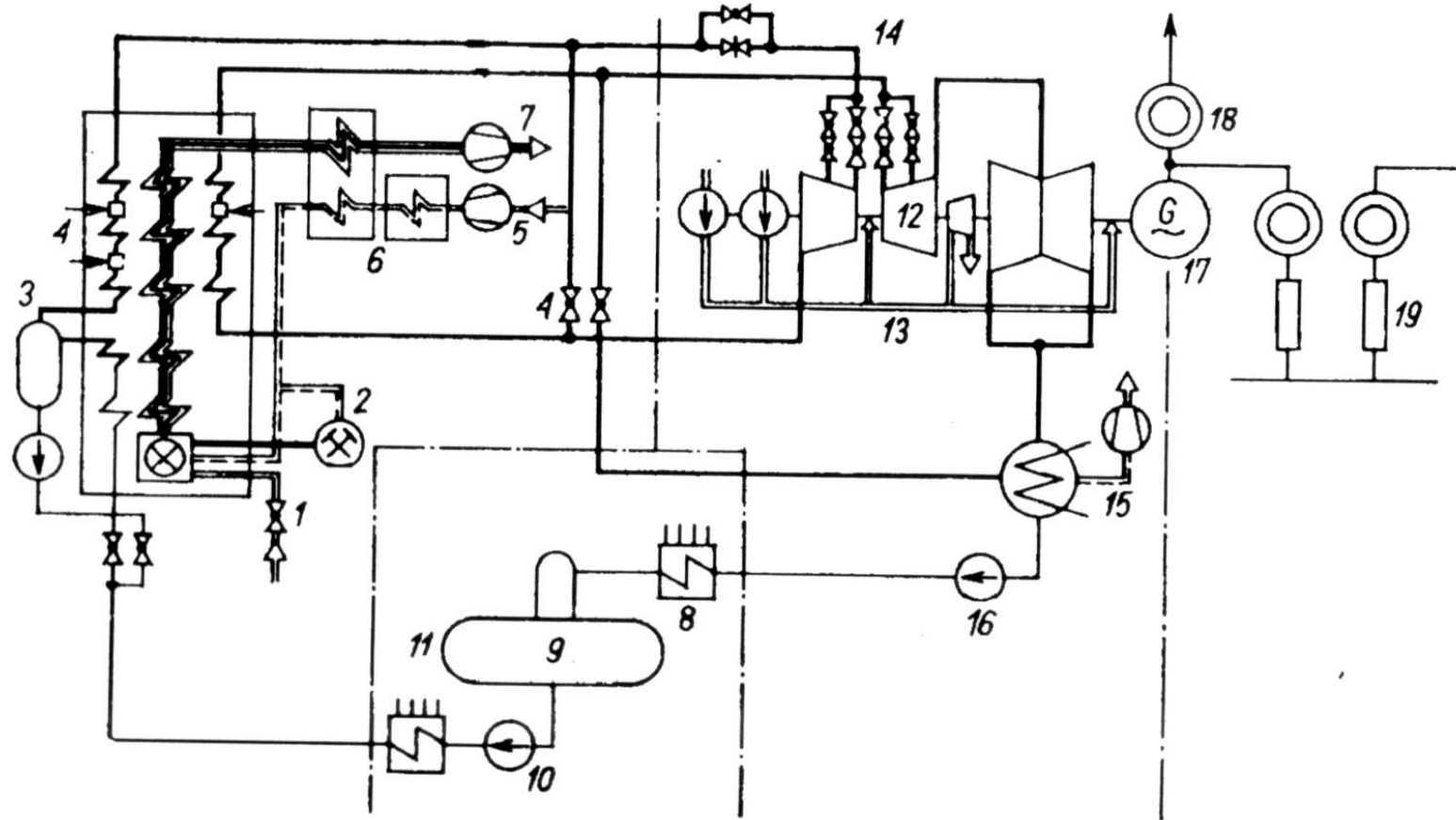
SIMATIC PCS 7
Takes you beyond the limits!



Komputerowe układy sterowania



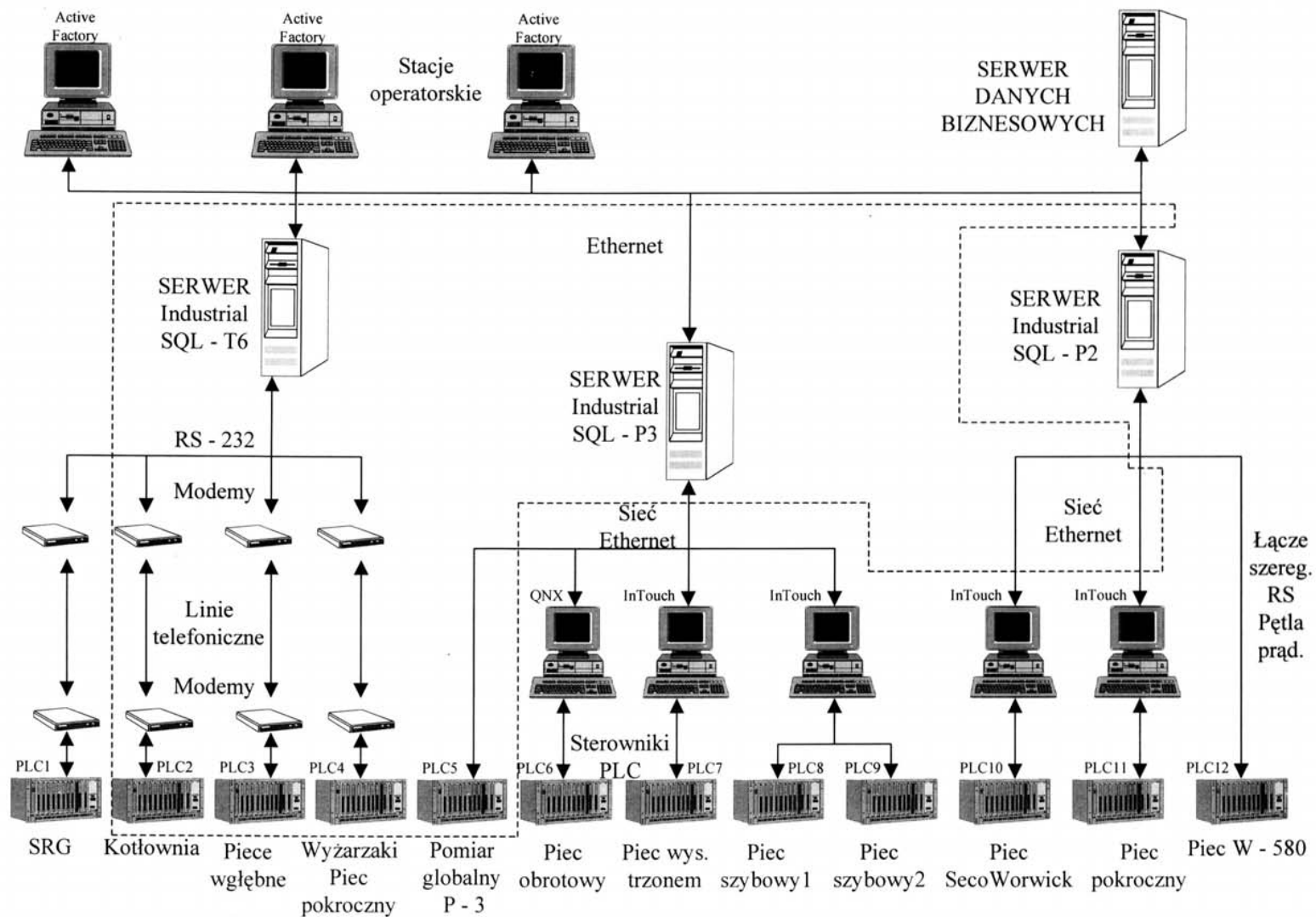
Rys. Struktura funkcjonalna typowego systemu automatyki



Rys. 2.3.1. Schemat ideowy bloku energetycznego i jego podstawowe bloki funkcyjne. Bloki funkcyjne kotła:

1 — palniki rozruchowe; 2 — palniki robocze; 3 — odparowacz; 4 — przegrzewacz; 5 — dmuchawy powietrza; 6 — podgrzewacz powietrza; 7 — ssawy gazów spalinowych. Bloki funkcyjne systemu zasilania wodą: 8 — podgrzewacz niskociśnieniowy; 9 — odgazowywacz wody; 10 — pompy wody; 11 — podgrzewacz wysokociśnieniowy. Bloki funkcyjne turbiny: 12 — układ sterowania turbiny; 13 — układ zasilania olejem; 14 — rurociągi ogrzewające; 15 — układ ewakuacji; 16 — pompy kondensatu. Bloki funkcyjne generatora i urządzeń elektrycznych; 17 — układ zasilania olejem i wodorem; 18 — zasilanie sieci; 19 — zasilanie potrzeb własnych

Projekt Sieci przemysłowej Huty Bankowej umożliwiającej monitoring poboru gazu

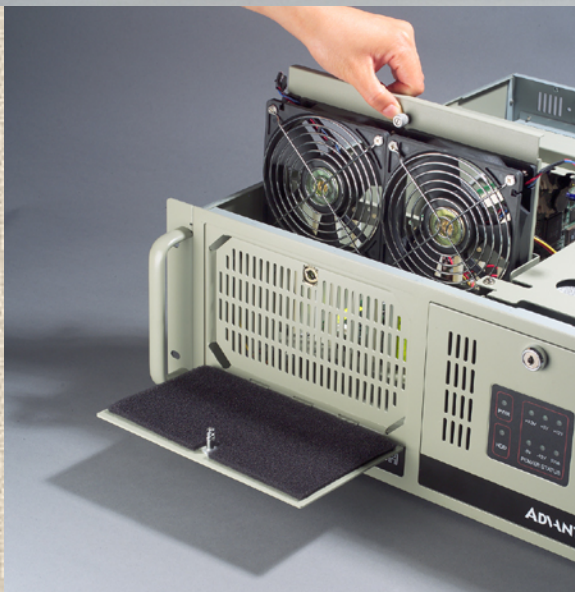


Opracował: inż. Robert Haraziński

Konsultacja: inż. Tadeusz Spisak

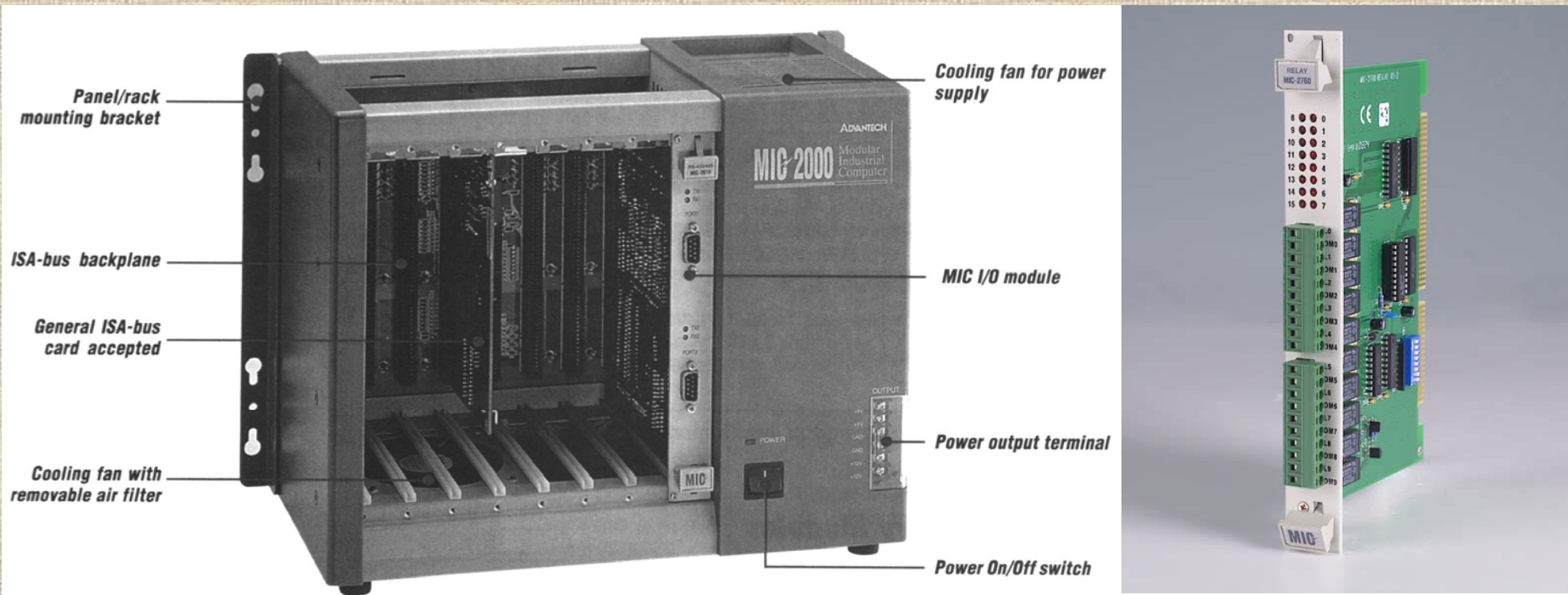
Rozwiązania sprzętowe i oprogramowanie w komputerowych układach sterowania

Komputerowe układy sterowania

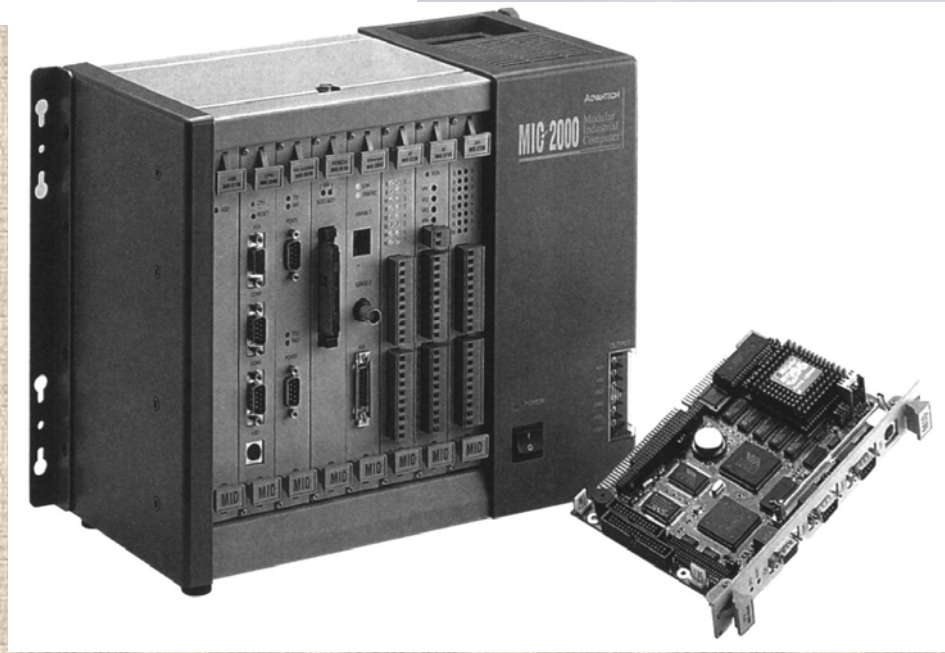


Rys. Komputer przemysłowy SYS-4U610-2S03 firmy Advantech (Intel® 915GV Platform , Pentium® 4 LGA775 procesor do 3.8 GHz, DDR2 SDRAM do 4 GB, do 10 kart PCI i 2 ISA, podwójny wentylator, filtry przeciwpyłowe, zasilacz ATX wysokiej klasy, ekranowanie, 4U rackmount.

Komputerowe układy sterowania



Rys. Modułowy komputer przemysłowy systemu MIC-2000 firmy Advantech w obudowie z pasywną magistralą ISA (8 lub 11 slotów, karta CPU Pentium, możliwość montowania kart PC-ISA).

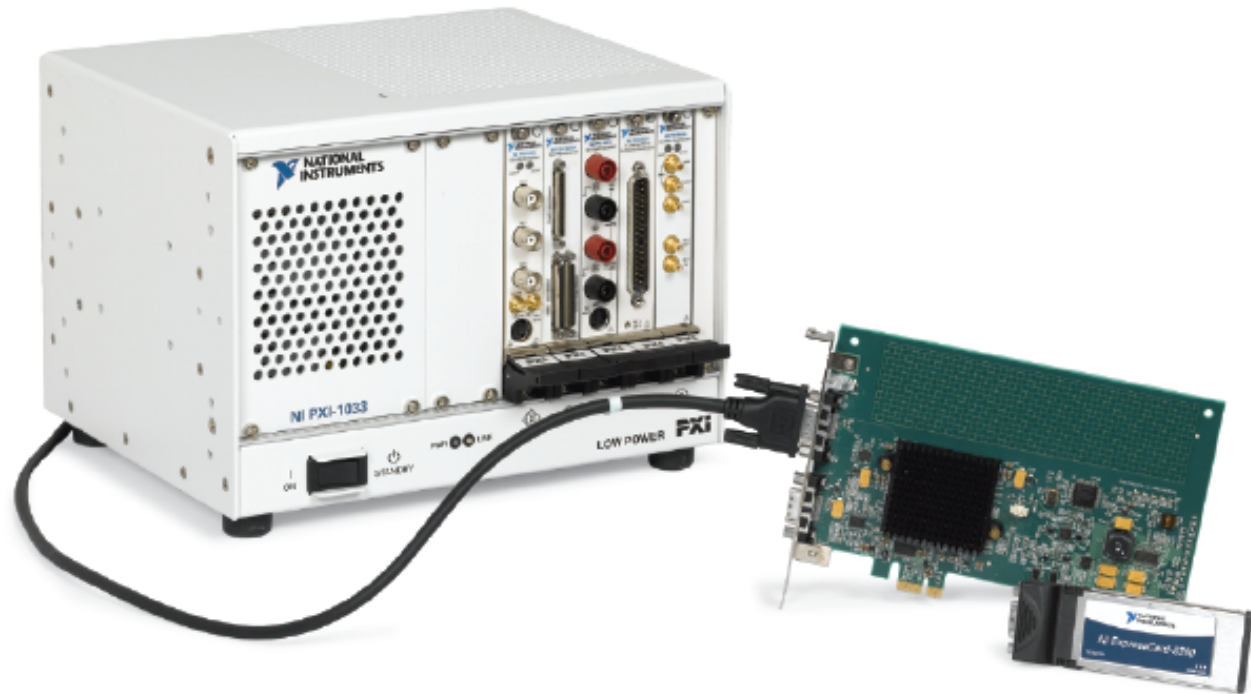


NI PXI-1033 Series

- Low-cost chassis for remote control applications
- Controlled from either a PCI Express desktop host or an ExpressCard laptop host
- MXI-Express remote controller achieves up to 110 MB/s sustained throughput
- Rugged, compact package accepts up to 5 peripheral modules
- Acoustic noise as low as 38 dBA
- Accepts both 3U PXI and CompactPCI modules

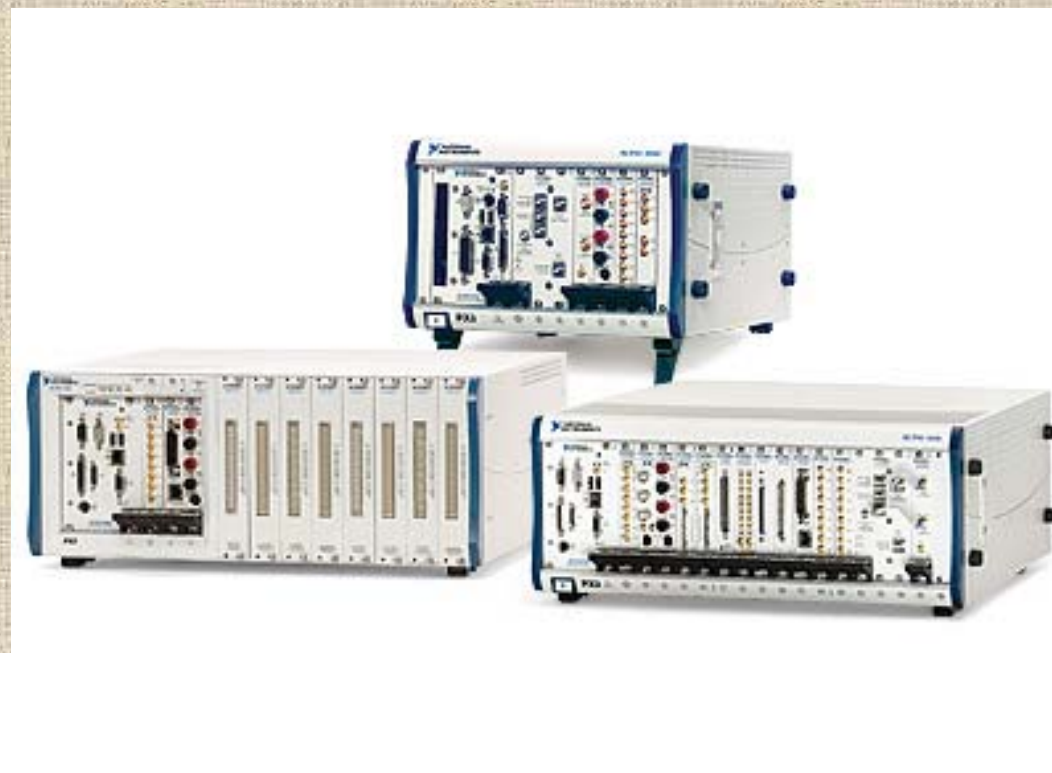
Options

- Compatible with all PXI-103x family accessories
 - Rack-mount kit
 - Handle and feet kit



Rys. Obudowa z modułami pomiarowo-sterującymi sterowanymi z hosta PC poprzez kartę kontrolera PCI (PCMCIA) z szybkim interfejsem MXI (do 110 MB/s). Wewnętrzna magistrala PXI (*PCI eXtension for Instrumentation*) na *backplane*. Może mieć własny moduł procesora i stanowi wtedy odrębny sterownik.

Rys. Obudowy sterowników PXI z modułami CPU (sterownika, z lewej strony) i kartami instrumentów.



Rys. Czterordzeniowy sterownik National Instruments PXI 8110



Rys. Compact Embedded Computer Advantech ARK-4180: Win XP Embedded lub Win CE, Celeron M do 1 GHz, 1 GB pamięci CF w wykonaniu przemysłowym, 512 MB DDR SDRAM, slot rozszerzeń PCI, bez wentylatora, temp. pracy. -40 ~ +75° C (możliwość pracy na zewnątrz, w zapyleniu i wilgoci)

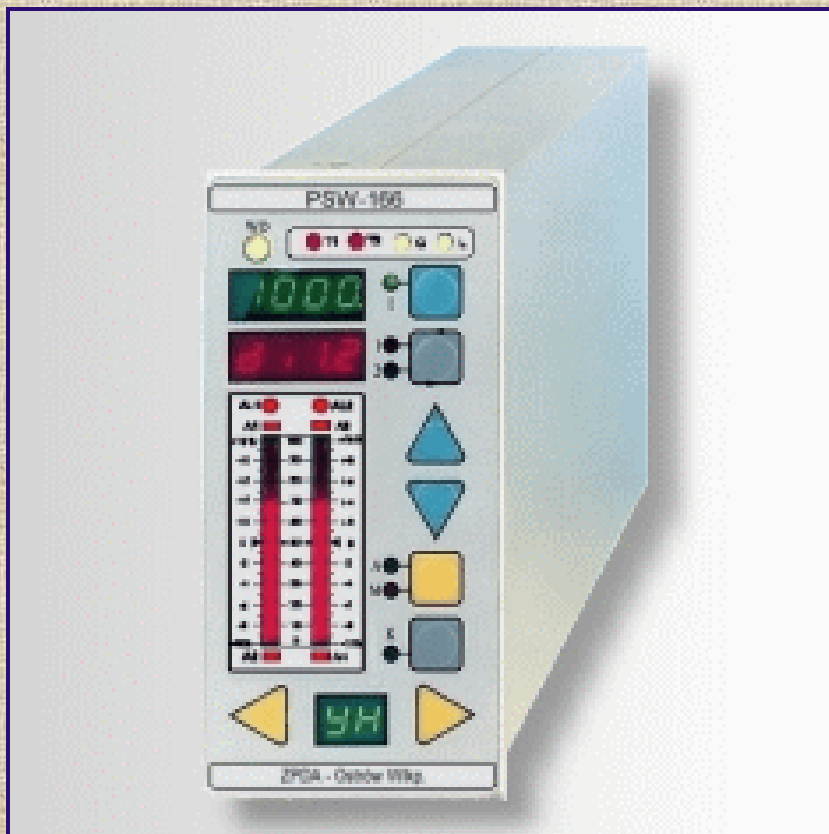
Rys. Compact Embedded Computer ARK-5280: Pentium M lub Celeron M do 1.8 GHz, VGA/DVI, slot rozszerzeń PCI i PCI/ISA, antywstrząsowy montaż HDD i kart rozszerzeń, bez wentylatora





Rys. Panel PC PPC-155T: 15" TFT XGA LCD, Pentium/Celeron M up to 1.8 GHz, System memory up to 2 GB DDR RAM, Built-in CF card interface (IDE interface for OS booting), One PCI expansion slot, Two 10/100Base-T Ethernet ports, Two IEEE 1394 ports, Resistive, capacitive, SAW touch screen are available

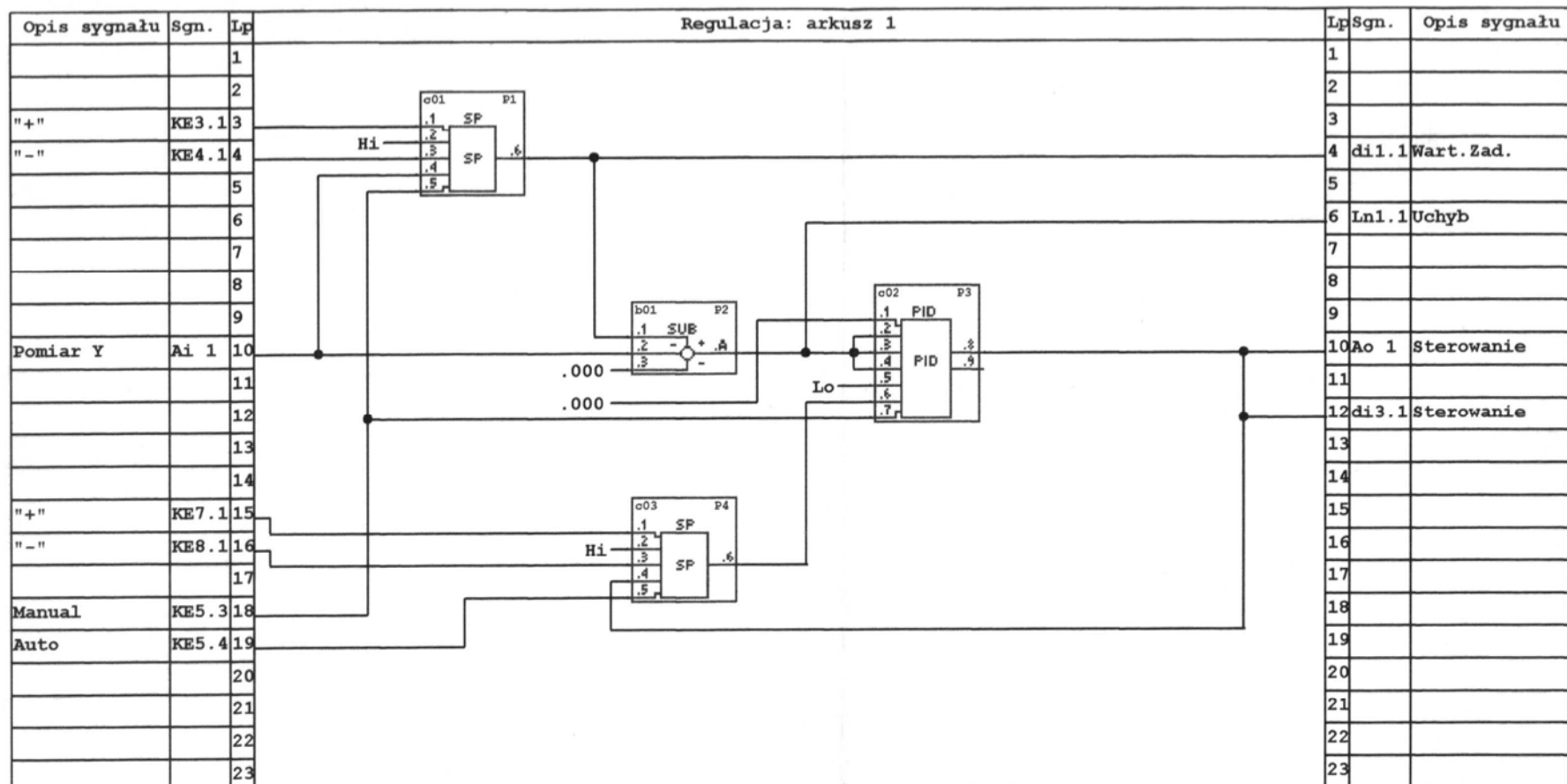
Regulatory wielofunkcyjne



Rys. Regulator - sterownik wielofunkcyjny PSW-166-11 (ZPDA). Zasilanie 24VDC, procesor 80C166, zegar RTC, we/wy separowane. Transmisja wg protokołu MODBUS-RTU lub TRANS. Programowanie FBD (99 bloków), współpracuje z systemami FIX, InTouch, WIZCON.

8/12 wejść analog.	<i>0/4-20mA/50Ω</i>
	<i>opcja 0-1V, 0-10V</i>
8 wyjść analog.	<i>0/4-20mA/450Ω</i>
	<i>opcja 0-10V</i>
20 wejść binarnych	<i>24VDC/10mA</i>
16 wyjść binarnych	<i>24V/0,4A</i>
zab. przed zwarcie	<i>>0,5A</i>
2 wyjścia przek.	<i>30V/1A</i>
2 wyjście watch-doga	<i>30V/1A</i>
2 wejścia częst.	<i>0-50kHz</i>
Wej. i wyj. analog. oraz bin.-separowane	
Dokładność przetw.	<i><0,2%</i>
Cykl oblicz. regul.	<i>200ms</i>
Cykl oblicz. ster.	<i>20ms</i>
Komunikacja pionowa	<i>RS232C sep. lub 485 sep.</i>
Komunikacja pozioma	<i>CAN sep. (1MB/s)</i>

Komputerowe układy sterowania



OPIS:

Regulacja Stałowartościowa

Data	02.03.2004	Obiekt: Schemat blokowy Układ regulacji stałowarto	ZAKŁAD PRODUKCJI DOŚWIADCZALNEJ AUTOMATYKI Sp. z o.o.	Aparat :	Plik :	
Opracował	M.Sz.			PSW - 166	rys 6.5.p16	
Kreślił	M.Sz.			List :	Projekt :	Reg.
Sprawdził				p16_list.001	rys 6.5	1/ 2

Program KGS166 v. 1.13 PSW-166 v.0.46

Rys. Wydruk programu drabinkowego stałowartościowej regulacji PID



Rys. Regulator mikroprocesorowy EFTRONIK XS U498 (Mera-Pnefal)

Realizowane funkcje:

- algorytmy regulatorów ciągłych (PID), dwu- i trójstawnych oraz krokowych,
- możliwość samostrojenia (alg. przekąźnikowy Astroma-Haglunda z aproksymacją modelem jednoinercyjnym z opóźnieniem $0.1 \leq T_0/T_{zast} \leq 0.6$)
- programowa zmiana wartości zadanej,
- funkcje logiczne i arytmetyczne, funkcje czasowe,
- wybór rodzaju wejścia: elektryczne standardowe napięciowe lub prądowe, czujnik temperatury, np. Pt100, pneumatyczne (20...100kPa),
- nieulotność danych po zaniku napięcia, transmisja RS-485,
- algorytmy integratora przepływu,
- regulacja stałowartościowa, kaskadowa, kaskadowa regulacja stosunku.

DANE TECHNICZNE

1. Wejścia analogowe:

- 2 lub 4 wejścia
- prądowe 0/4 ... 20 mA, 0,1 ... 5 mA,
- napięciowe 0/1 ... 5 V, 0 ... 10 V
- termorezystancyjne, Pt100, Pt500 (rezyst. linii $\leq 10 \Omega$)
- potencjometryczne, 100 Ω , 200 Ω , 0,5k Ω , 1k Ω , 1,5k Ω
- pneumatyczne, 1 lub 2 wejścia 20 ... 100kPA
- dokładność przetwarzania < 0,25%

Możliwość programowego skalowania wejść pozwala na uzyskanie innych niż w/w zakresy sygnałów wejść.

2. Wejścia dyskretne:

- 4 wejścia binarne
- sygnał napięciowy "1" log: 16 ... 24V dc lub 3 ... 5 V dc
- sygnał napięciowy "0" log: 0 ... 7V dc lub 0 ... 1,5V dc
- sygnał zwarciový "1" log: zwarcie wejścia
- sygnał zwarciový "0" log: rozwarćie
- rezystancja obwodu sygn. zwarciového: $\leq 100 \Omega$

3. Wyjścia analogowe:

- 2 wyjścia prądowe 4 ... 20mA
- (R_{obc} = 0 ... 750 Ω)
- lub napięciowe 1 ... 5 Vdc,

4. Wyjścia dyskretne:

- 4 wyjścia
- zestyki przełączne 24 V dc, 200 mA/wyj.

- zestyki zwierne 220 V ac, 1 A/wyj.
- triakowe 220 V ac, 1 A/wyj.

5. Zasilanie urządzeń dodatkowych:

- możliwość zasilania maks. 5 szt. przetworników o łącznym poborze prądu maks. 125 mA.

6. Zasilanie:

220 V, 50Hz, (opcjonalnie 24 V dc),

7. Pobór mocy:

maks. 20 VA,

8. Interfejs szeregowy

do współpracy z komputerem IBM:

RS 485,

9. Stopień ochrony obudowy wg PN-92/E-08106

- płyta czołowa IP54
- obudowa za tablicą IP20

10. Dopuszczalna temperatura:

- otoczenia +5 ... +40°C
- transportu i składowania -20 ... +50°C

11. Dopuszczalne wibracje:

- do 55 Hz, / 0,075 mm

12. Wilgotność względna otoczenia:

30 ... 80%

13. Wymiary gabarytowe:

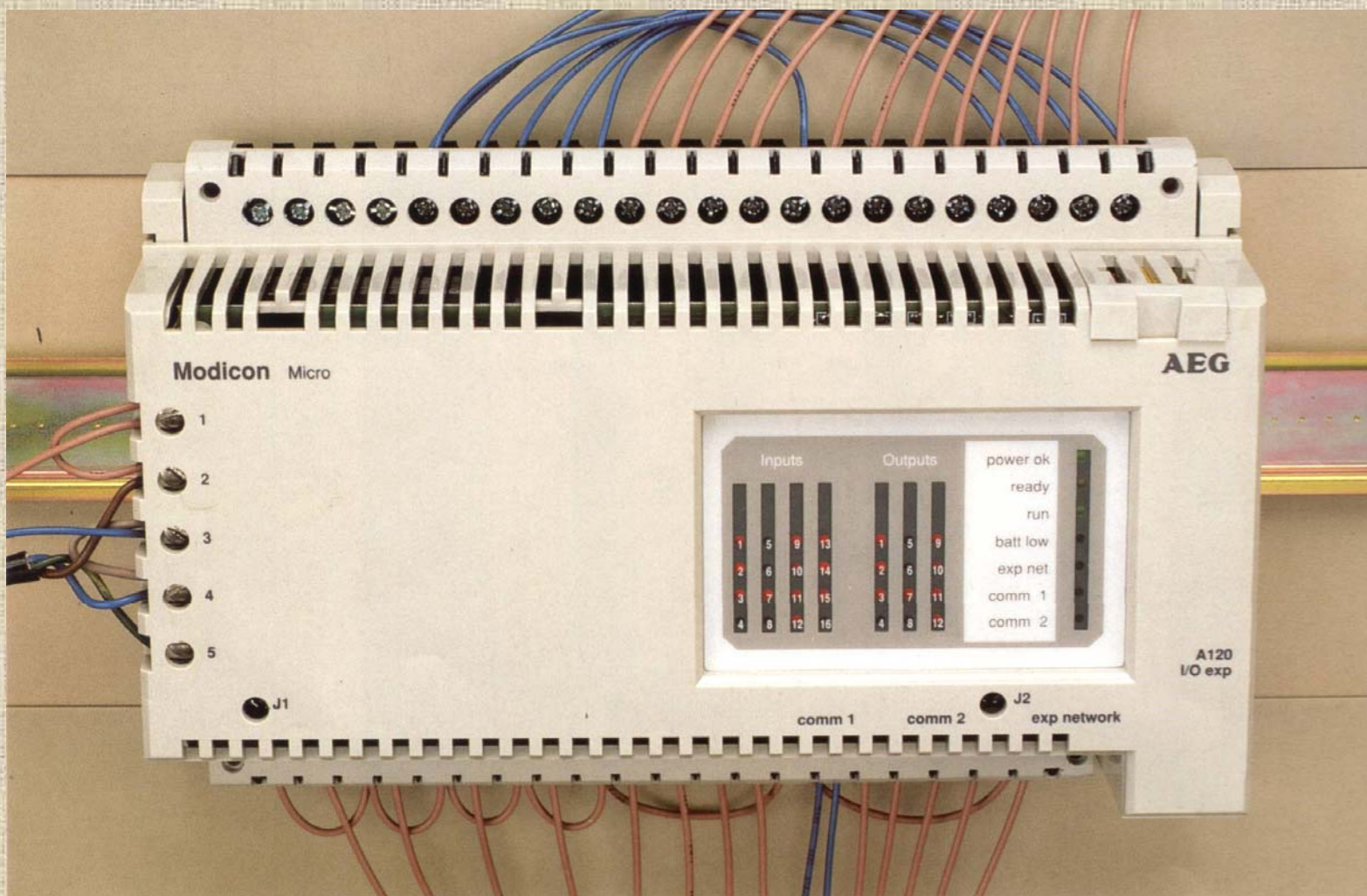
72 x 144 x 235 mm

Istnieje możliwość "montażu płaskiego", tzn. z oddzielnym pulpitem operatorskim,

14. Masa:

2 kg

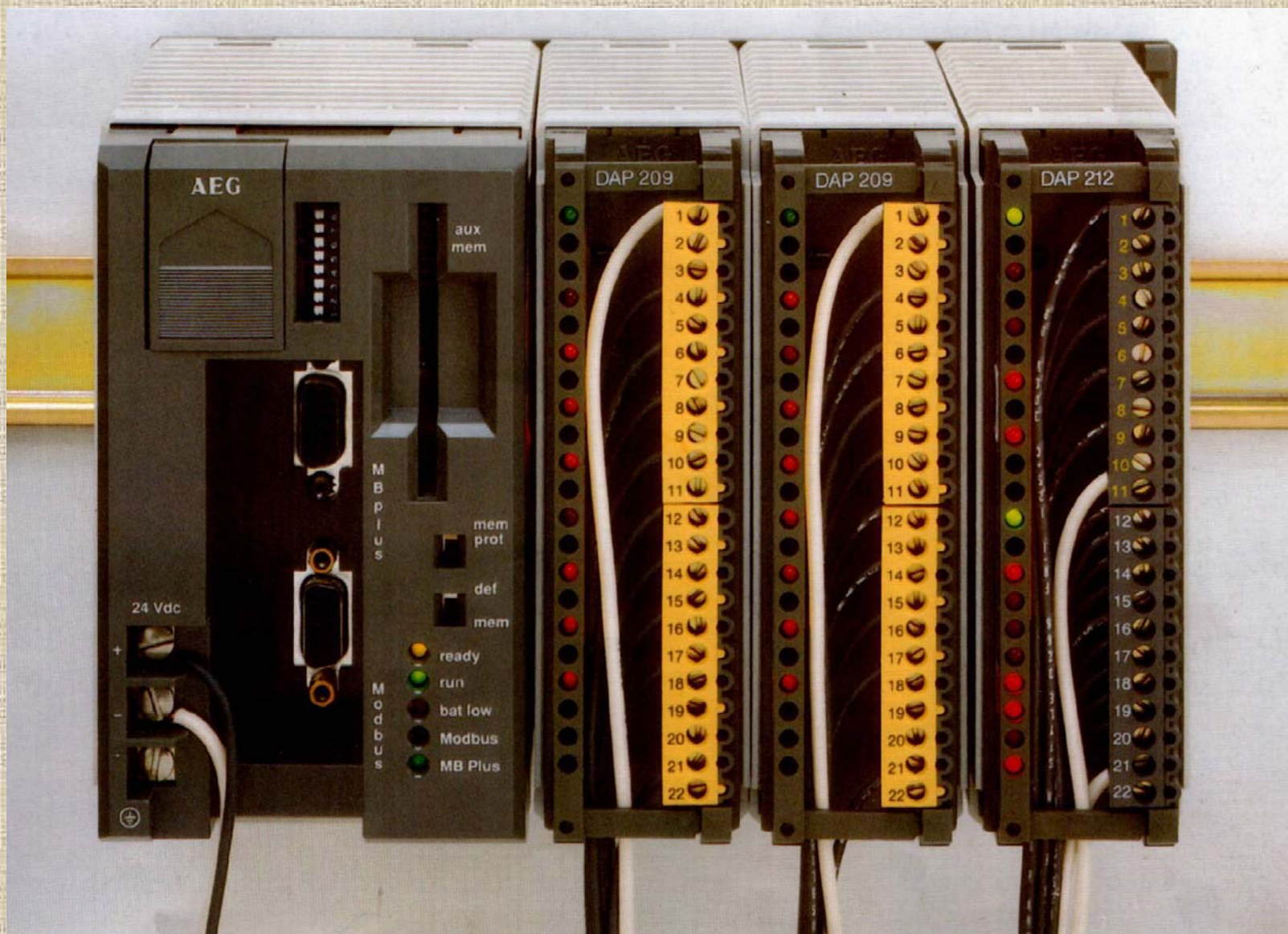
Programowalne sterowniki logiczne PLC



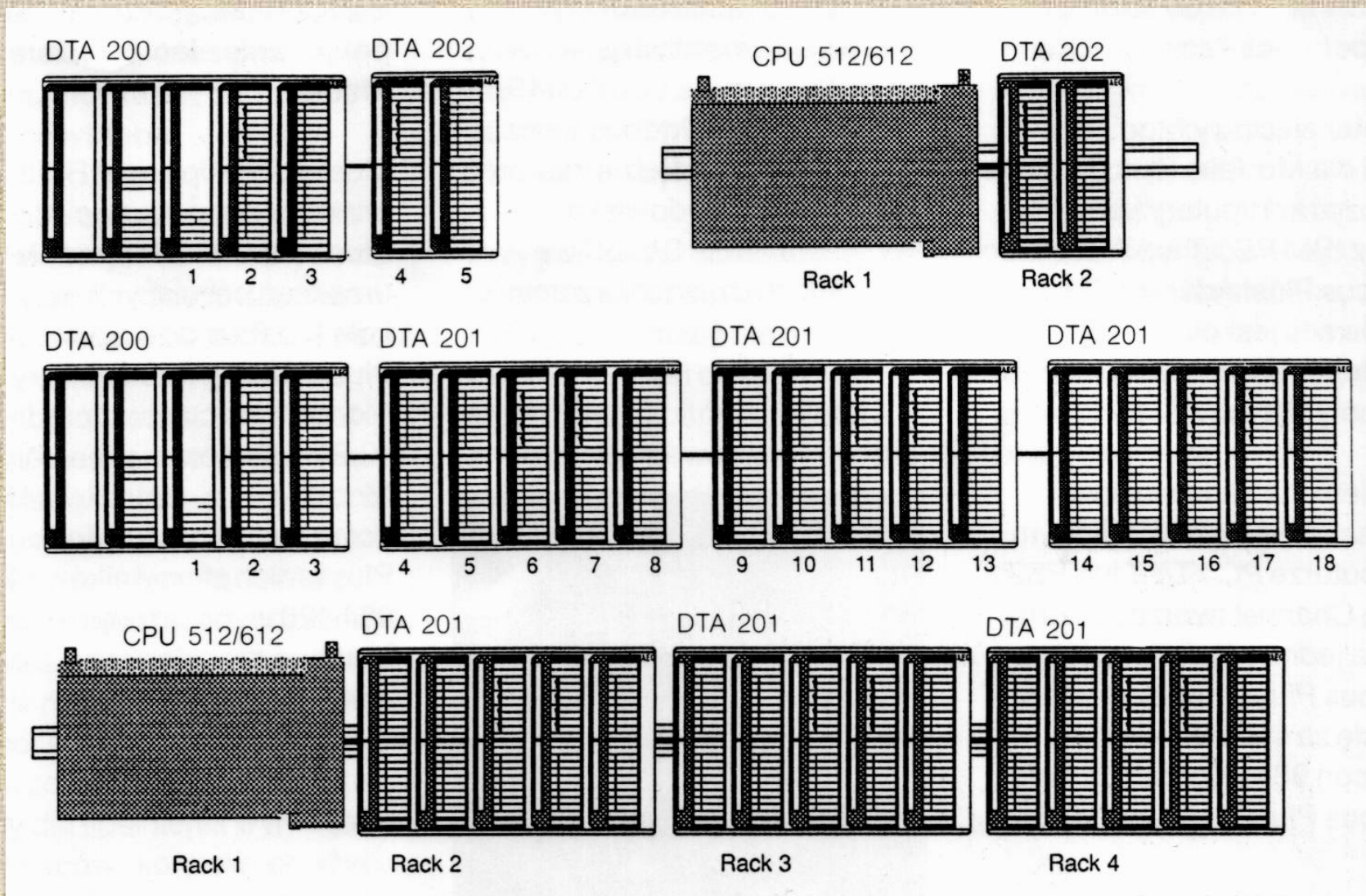
Rys. Kompaktowy sterownik przemysłowy PLC (*Programmable Logic Controller*) Modicon Micro: 4 AI, 2 AO, 16 DI, 12 DO, 2 porty szeregowo, zasilanie 24 VDC



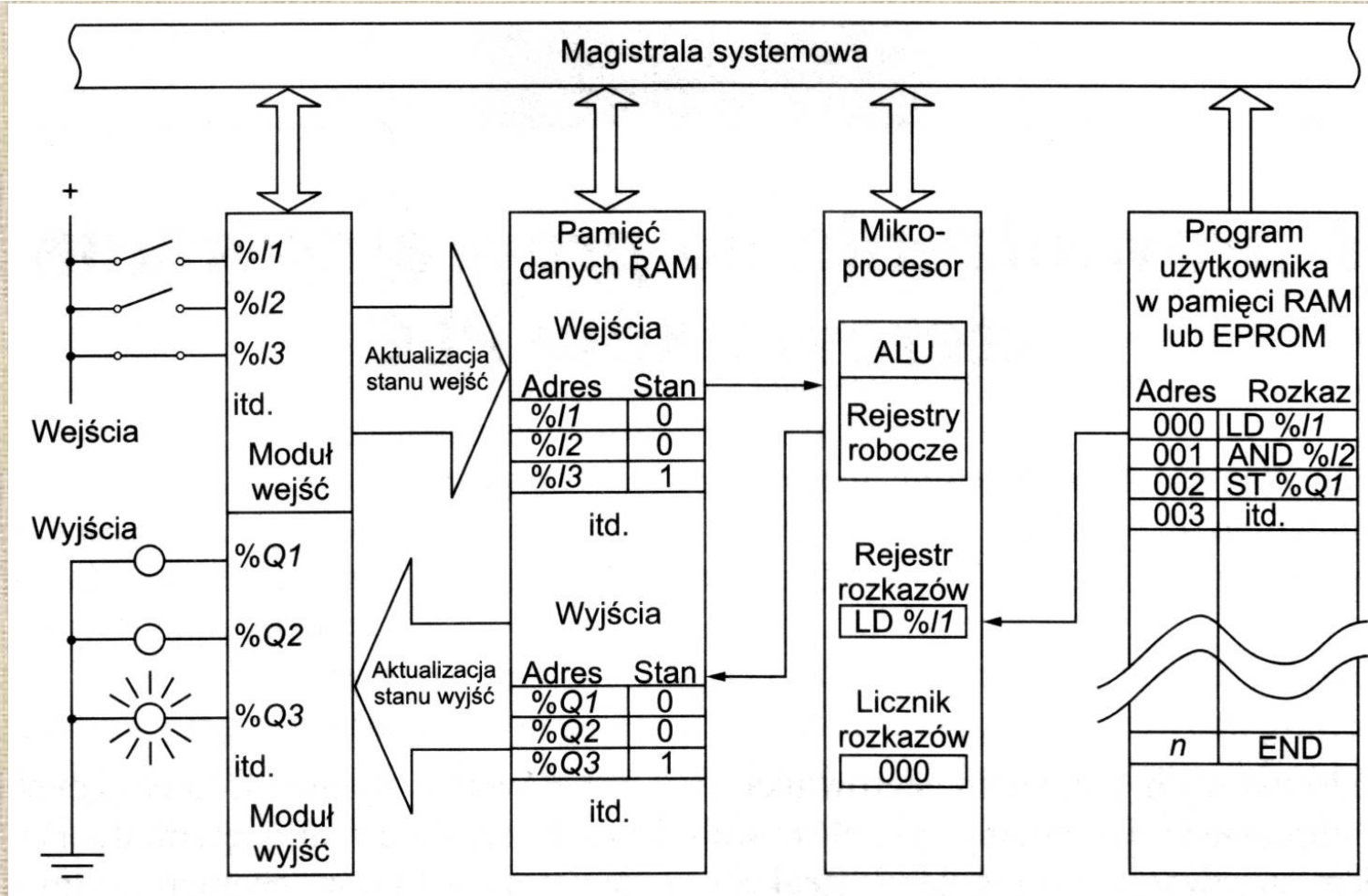
Rys. Mały modułowy sterownik PLC Modicon 984-120 Compact kompatybilny programowo z Modicon Micro



Rys. Sterownik PLC Modicon 984-120 Compact bez kłapek modułów



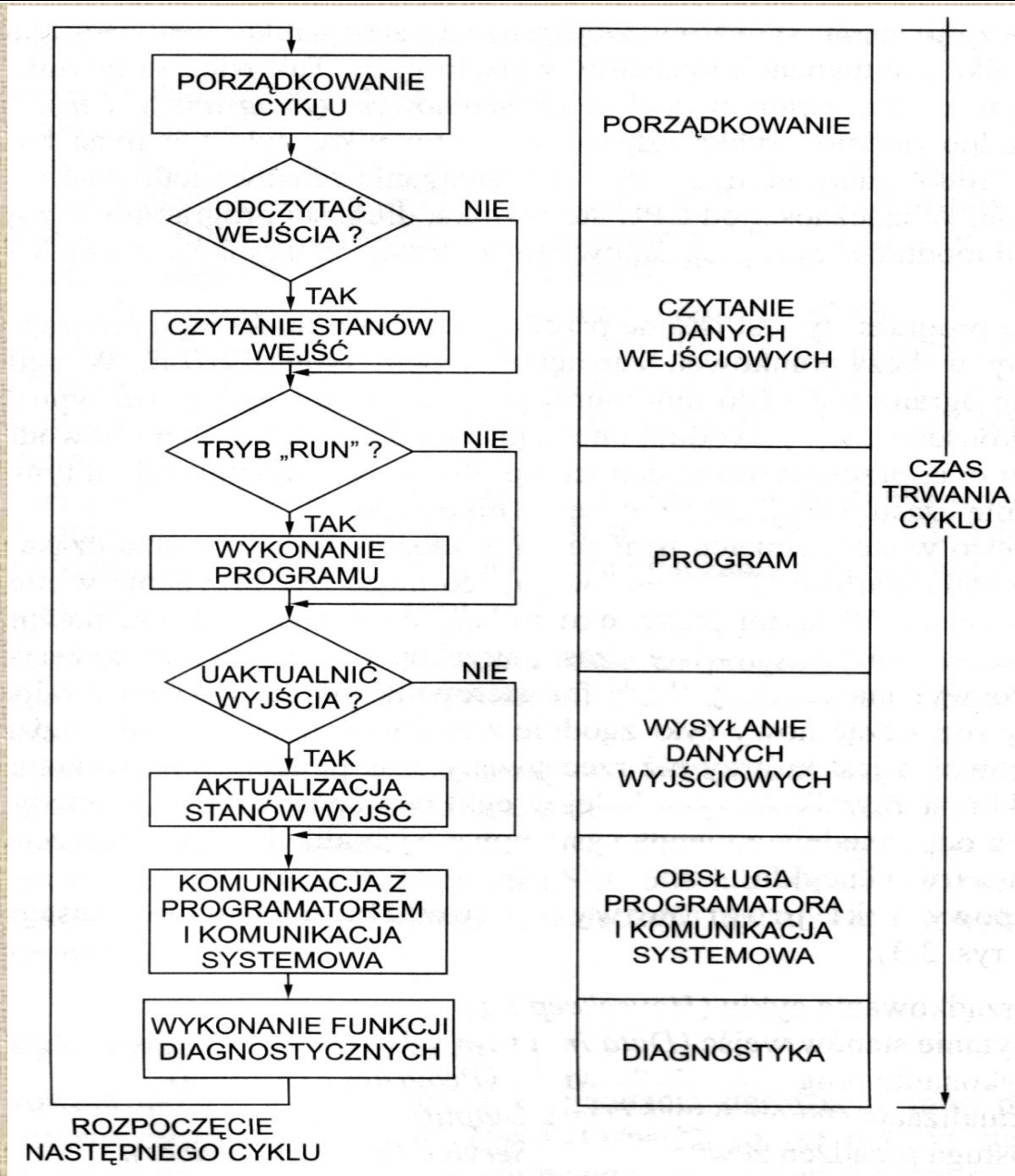
Rys. Konfiguracja liniowa rozszerzeń sterowników Modicon Micro (CPU 512/612) i 984-120 Compact: maksymalnie 3 moduły rozszerzeń DTA 201 (15 dodatkowych modułów we/we, razem 256 we/we binarnych)



Rys. Ideowy schemat funkcjonalny sterownika.

Program sterujący jest realizowany w bezwarunkowej pętli WHILE. W niektórych sterownikach istnieje możliwość ustawienia stałego czasu cyklu (*Constant Sweep*, musi on być nie mniejszy niż rzeczywisty czas potrzebny sterownikowi).

Komputerowe układy sterowania



Rys. Typowe fazy cyklu programowego sterownika PLC.

Fazy cyklu *Program Sweep*:

Faza porządkowania obejmuje obliczanie czasu trwania cyklu, planowanie startu i określenie trybu następnego, aktualizację tablicy błędów i kasowanie *Watchdoga*. Długość tej fazy zależy od CPU i jest stała.

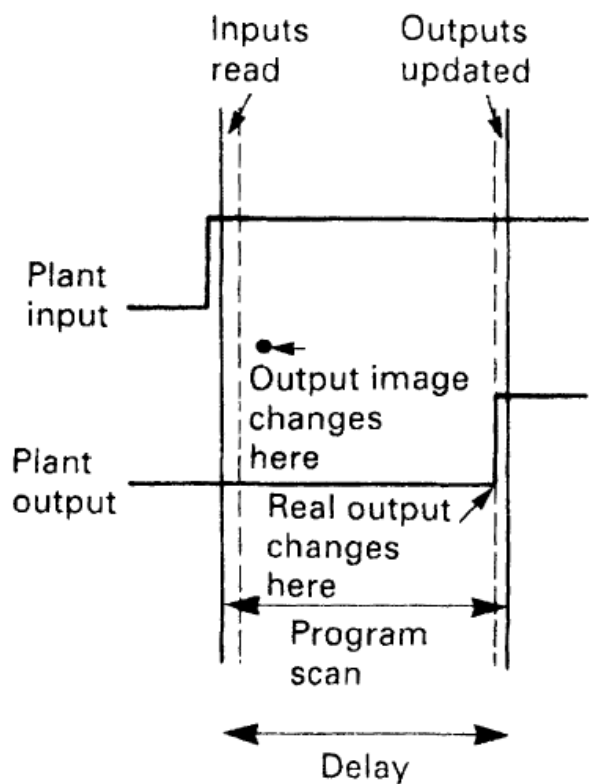
Faza czytania stanów wejść to przepisanie danych z buforów w modułach wejściowych (portów binarnych, 16-bitowych rejestrów A/C) do odpowiadających im obszarów pamięci CPU – *State RAM*. Czas trwania fazy zależy od liczby i typu modułów oraz ich lokalizacji (np. płyta główna lub kaseeta rozszerzeń).

Faza wykonania programu to jeden przebieg programu użytkownika (od pierwszej instrukcji do napotkania instrukcji END). Jego czas zależy od CPU, długości programu i użytych instrukcji. W trybie STOP nie jest wykonywana.

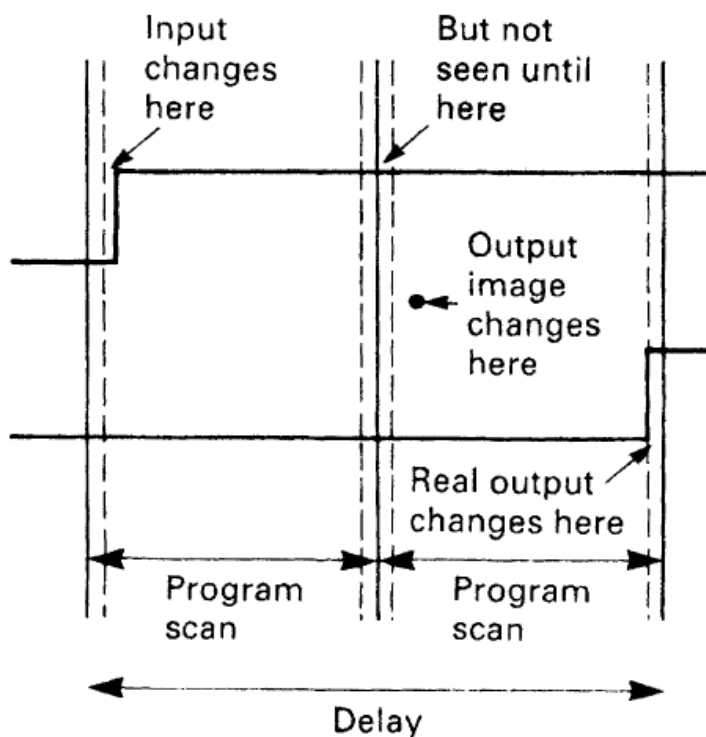
Faza aktualizacji stanów wyjść polega na przepisaniu danych z pamięci *State RAM* do odpowiednich buforów w modułach wyjściowych (porty binarne, 16-bitowe słowa C/A). Czas trwania fazy zależy od liczby i typu modułów oraz ich lokalizacji.

Faza obsługi urządzeń zewnętrznych obejmuje obsługę żądań wymiany danych od programatora (w trybie RUN czas obsługi jest ograniczony), modułów inteligentnych (które same wykonują pewne obliczenia) i modułów komunikacyjnych. Czas trwania zależy od liczby urządzeń i ilości wymienianych danych.

Faza diagnostyki obejmuje sprawdzenie konfiguracji oraz poprawności kolejnego fragmentu programu użytkownika (zadanej liczby słów). Czas jej trwania zależy od wydajności CPU.



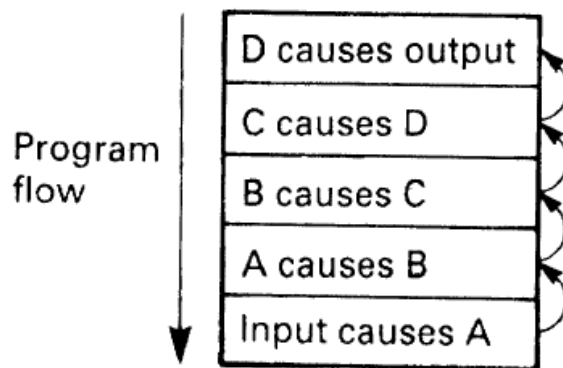
(a)



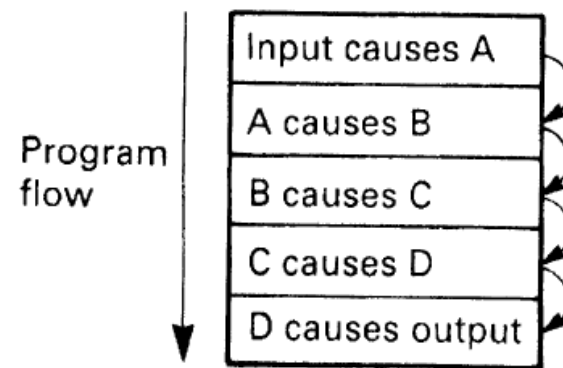
(b)

Rys. Wpływ skanowania programu na czas reakcji:
 a) najlepszy przypadek,
 b) najgorszy przypadek

Rys. a) Logika programu niezgodna z kierunkiem skanowania (wymagane 5 okresów),
 b) Logika programu zgodna z kierunkiem skanowania

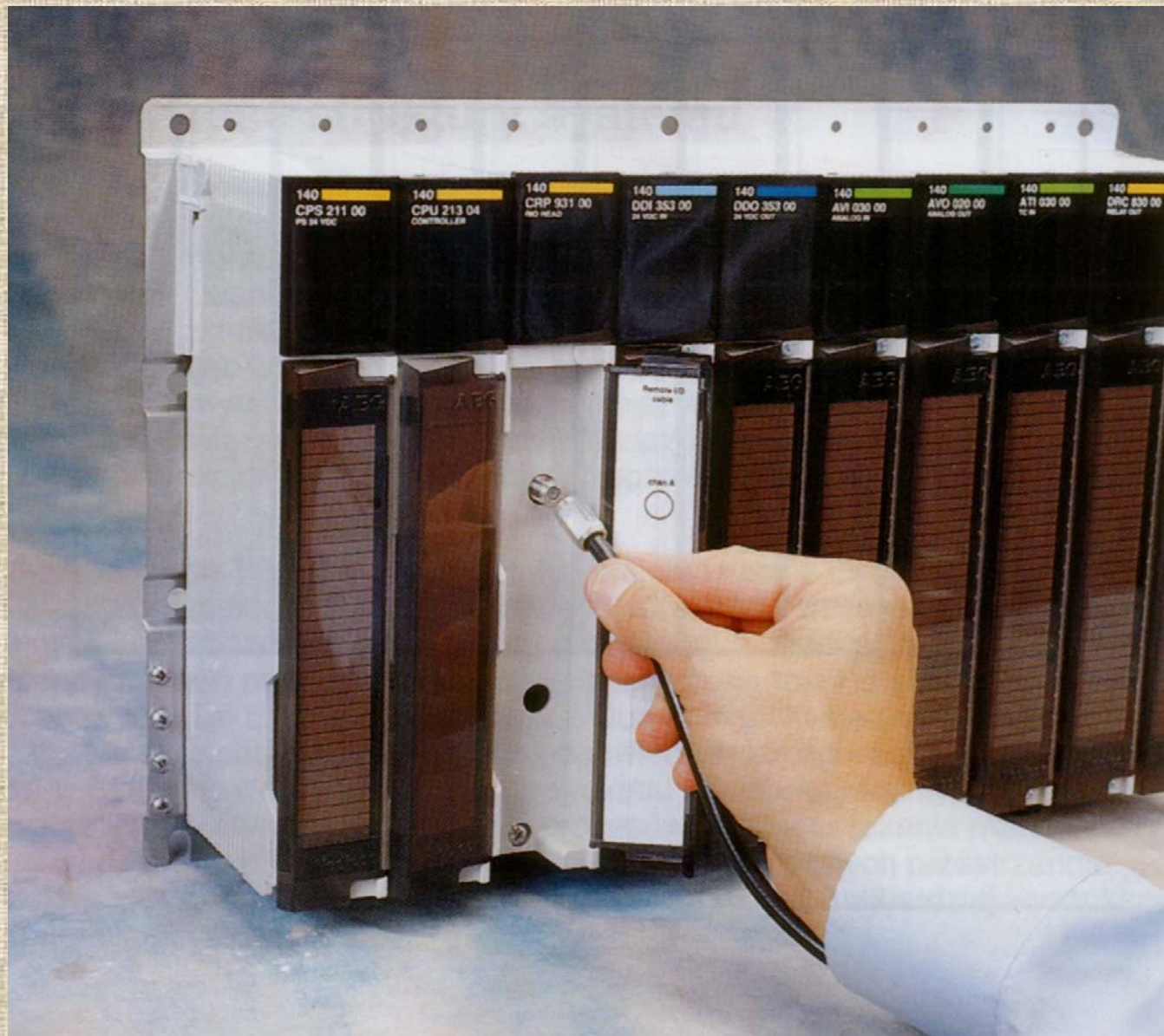


(a)



(b)

Komputerowe układy sterowania

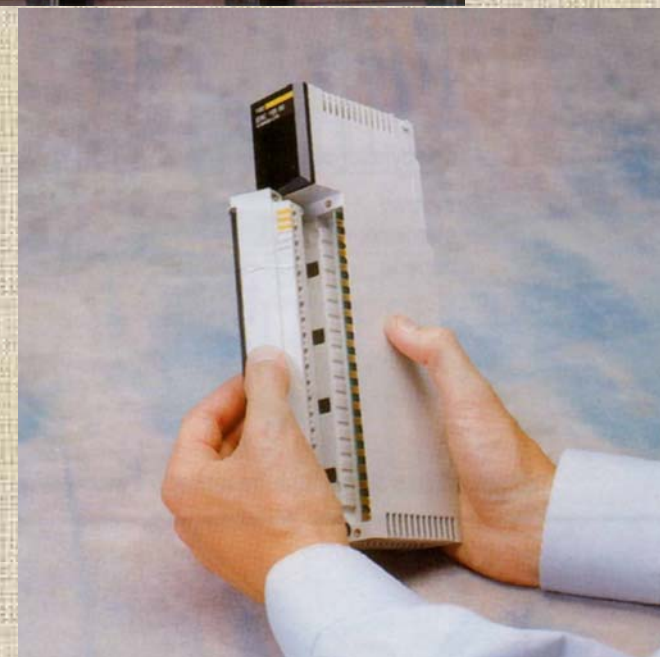


Rys. Modułowy sterownik Modicon TSX Quantum: - duża moc obliczeniowa (procesory 486 DX i szybsze), możliwość redundancji sterowników, zasilaczy i okablowania we/wy, - odporność na zakłócenia polem EM i ESD, - możliwość definiowania stanu wyjść dyskretnych w przypadku błędów komunikacji, - szybka obsługa przerw

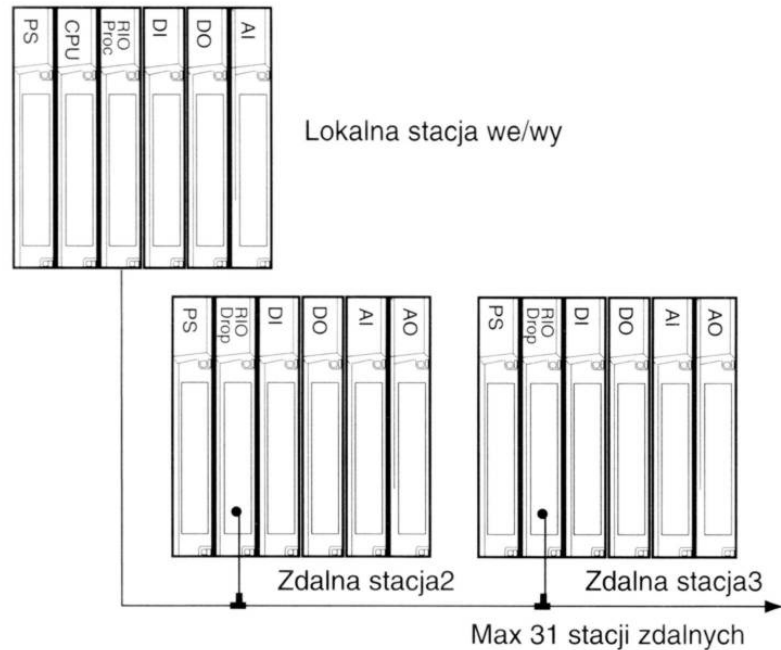
Komputerowe układy sterowania



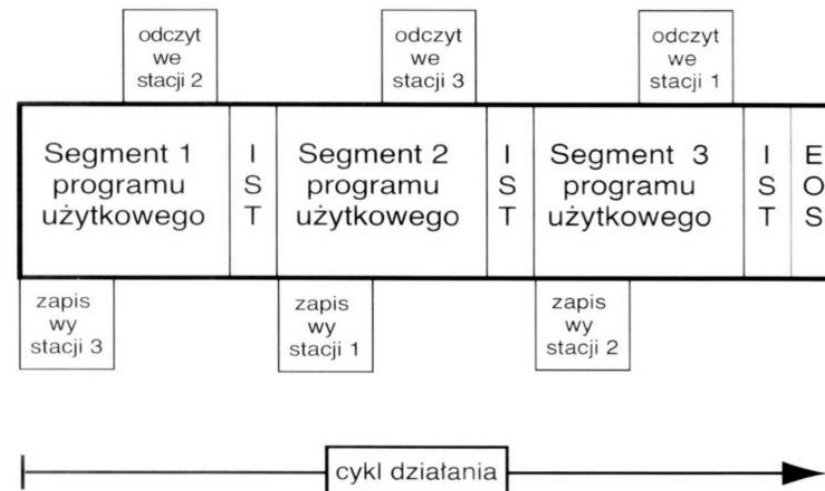
Rys. Modicon TSX Quantum: jedna stacja do 448 we/wy, do 64000 we/wy w przypadku sieci wielu stacji (do 31 zdalnych, 63 z repeaterem), pamięć do 4 MB na płycie, 64k rejestrów programu krótki czas próbkowania i duża dokładność we/wy analogowych 12-/16-bit, typowy czas realizacji cyklu programu 0.09-0.45 ms/kilo słów instrukcji



Typowy system zdalnych stacji we/wy procesowych



Segment Scheduler



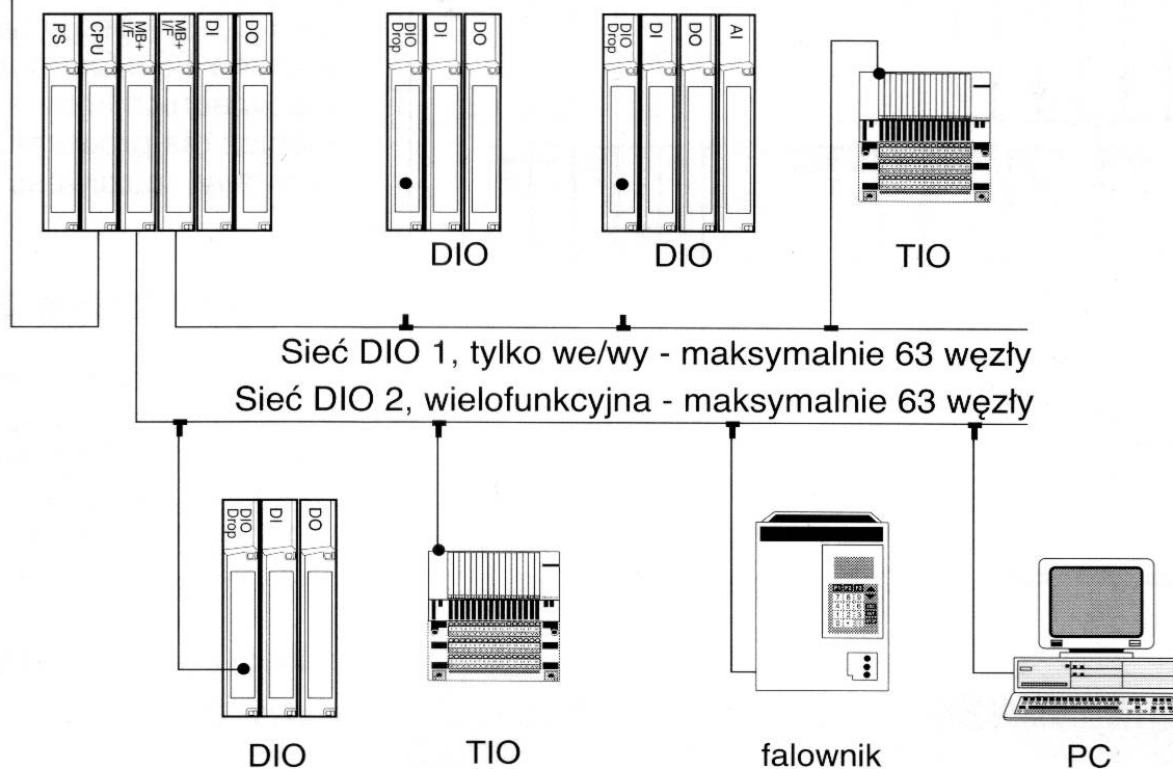
Rys. Segment Scheduler – program użytkowy podzielony na segmenty logiczne odpowiadające stacjom zdalnym; CPU stacji lokalnej przed rozpoczęciem danego segmentu obsługuje wejścia, a po zakończeniu segmentu – wyjścia procesowe danej stacji zdalnej.

IST - Intersegment Transfer - wymiana danych pomiędzy CPU stacji lokalnej i kontrolerem odpowiedniej stacji zdalnej

EOS - End of Scan - diagnozowanie i obsługa komunikacji kończące cykl działania

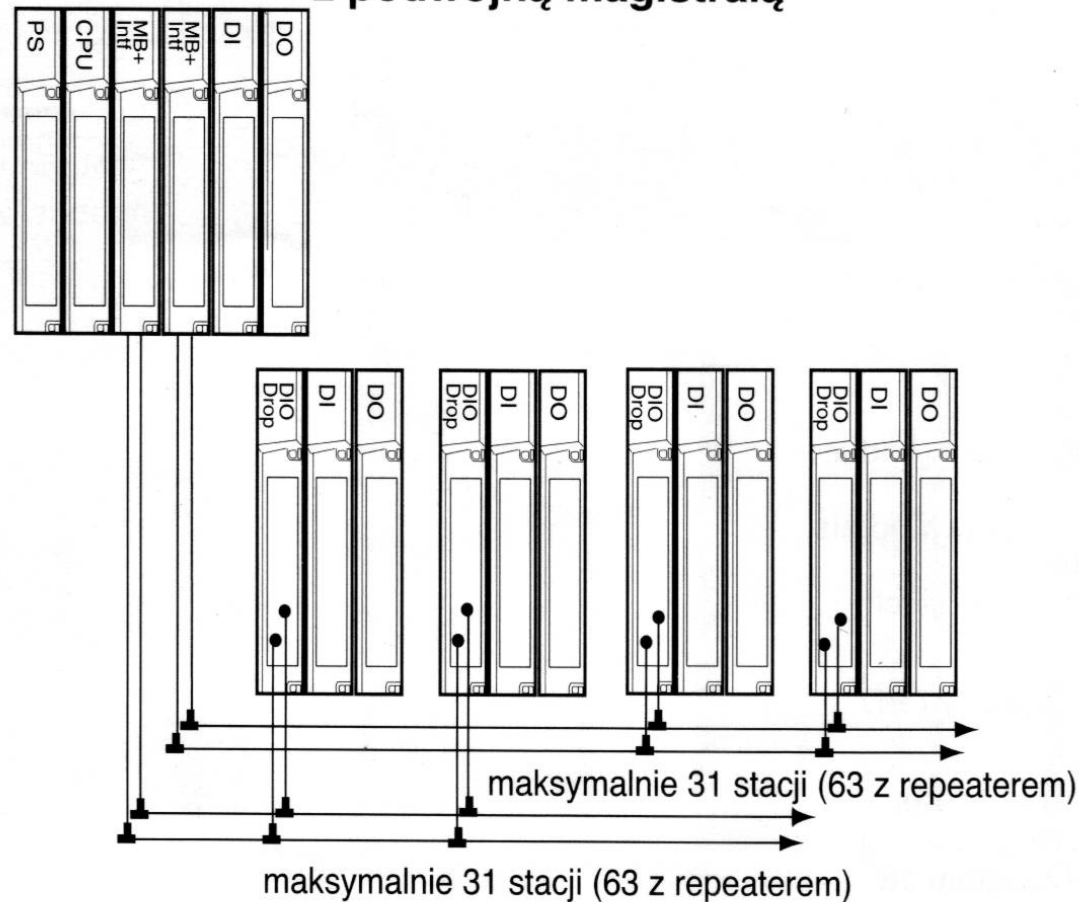
Typowy wielosieciowy rozproszony system we/wy procesowych

Modbus Plus do komputera nadrzędnego

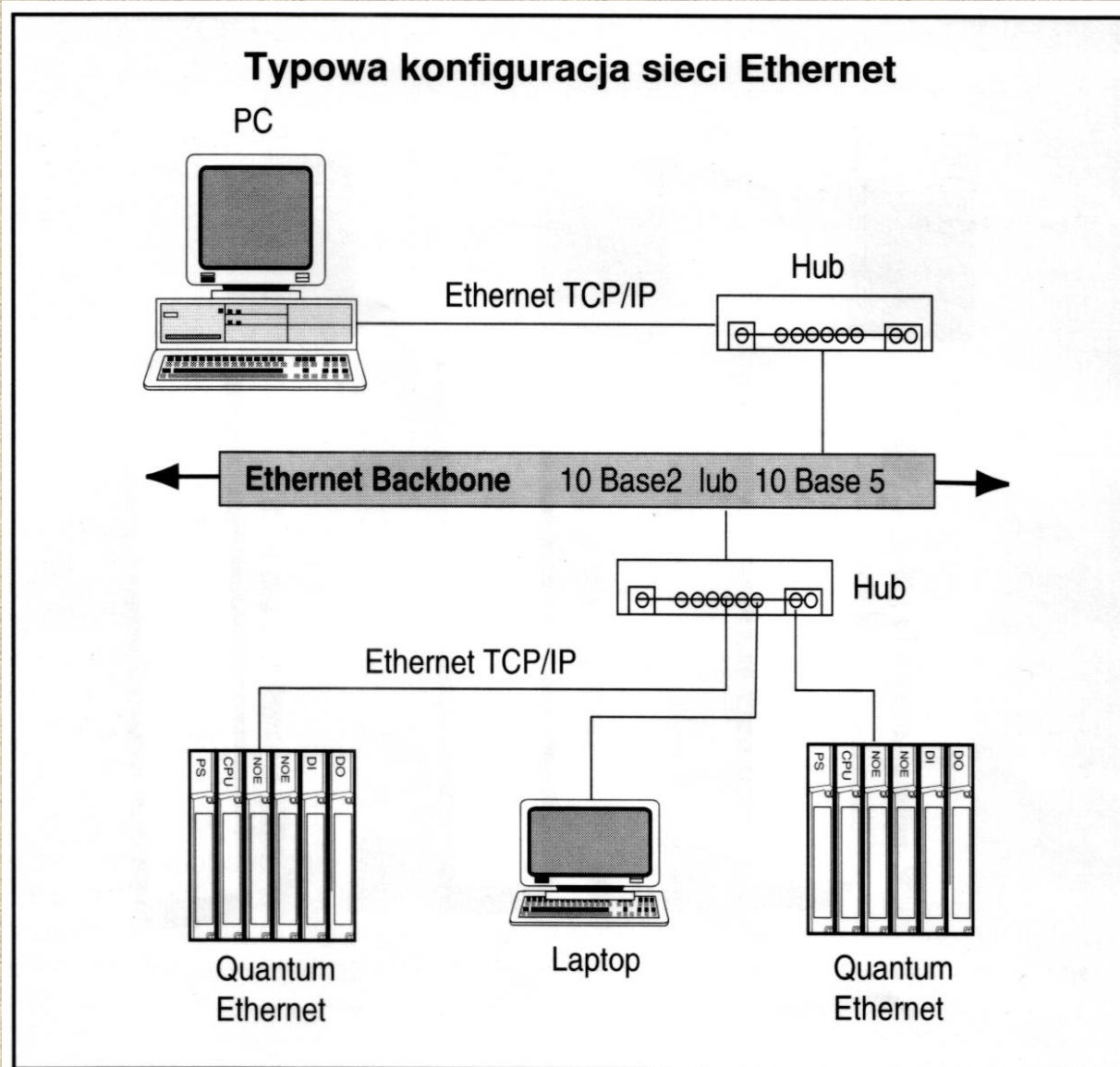


Rys. Konfiguracja wielosieciowa rozproszonych we/wy procesowych (*Distributed IO*) połączonych przez stację lokalną z kontrolerami sieci MB+IF (*Modbus Plus Interface*)

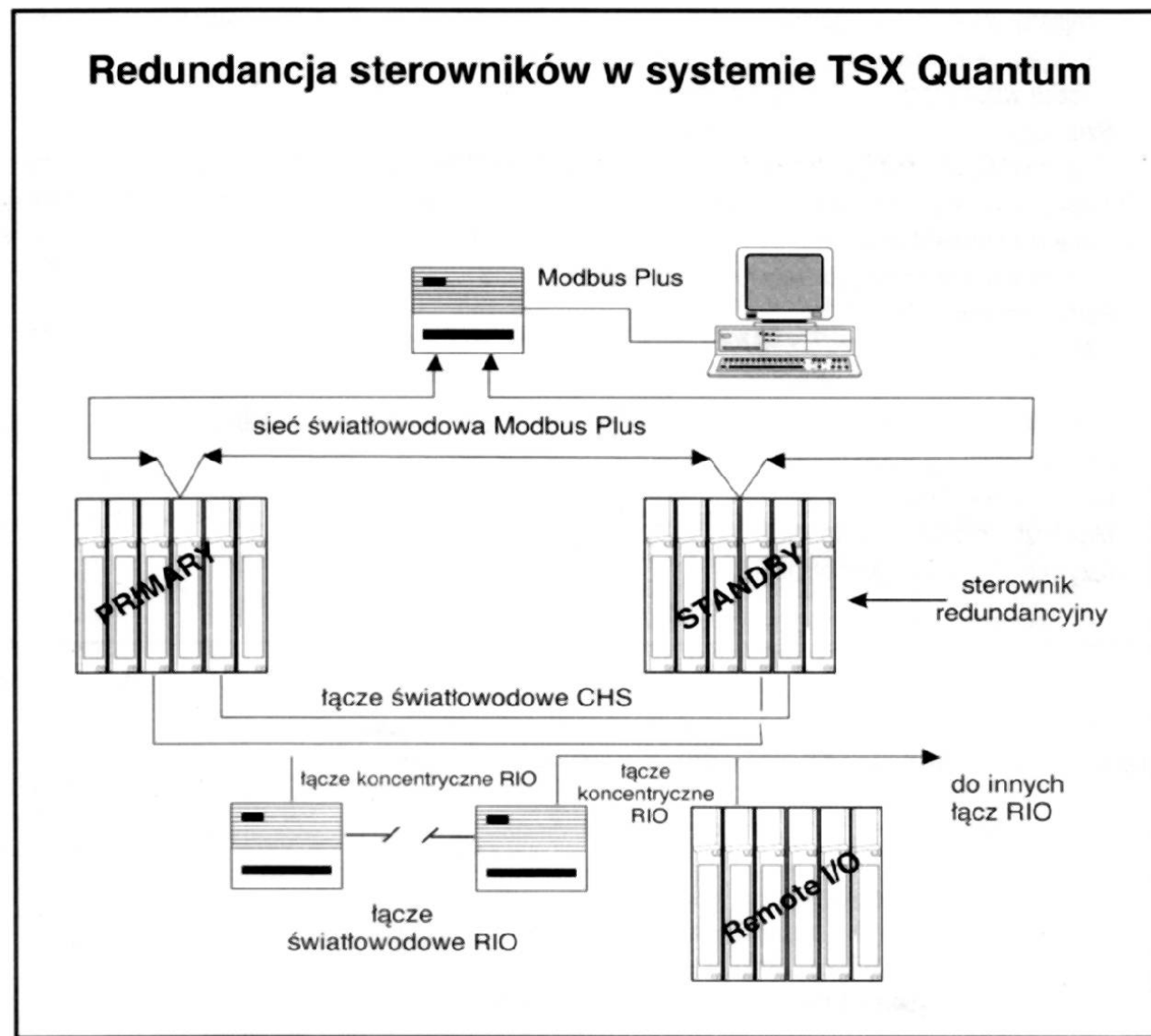
Typowy system rozproszonych we/wy procesowych z podwójną magistralą



Rys. Sieć stacji rozproszonych z podwójnym okablowaniem – zapewnia pracę w przypadku uszkodzenia pojedynczych kabli lub łączy, stan połączeń jest automatycznie monitorowany i sygnalizowany przez stacje.



Rys. Kontrolery Ethernet umożliwiają efektywną współpracę Quantum z innymi stacjami procesowymi; obsługiwane są protokoły TCP/IP, MMS i Sy/Max, medium: skrętka dwużyłowa 10BaseT lub światłowód 10BaseFL

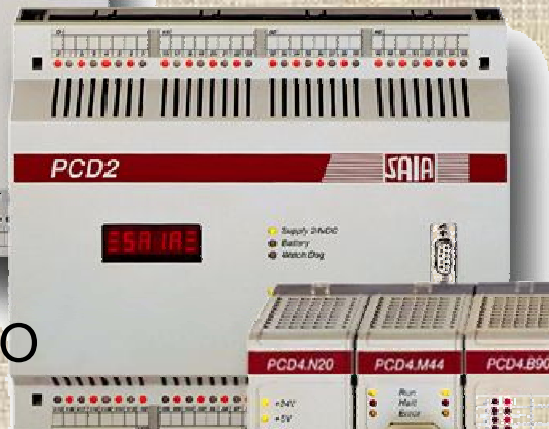


Rys. Redundancja sterowników Quantum: dwa identycznie skonfigurowane sterowniki z modułami CHS (*Controller Hot Standby*) umożliwiającymi płynne przełączenie na sterownik rezerwowy (oba sterowniki mają identyczny stan procesu)

PCD1



PCD2



max. 64 I/O

max. 256 I/O

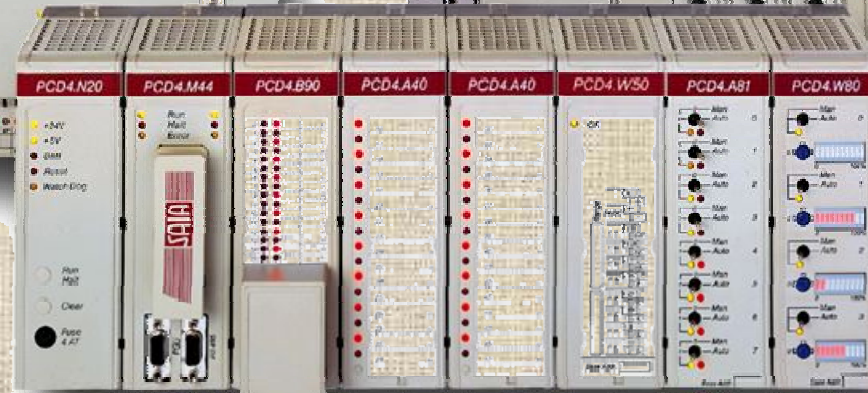
PCD6



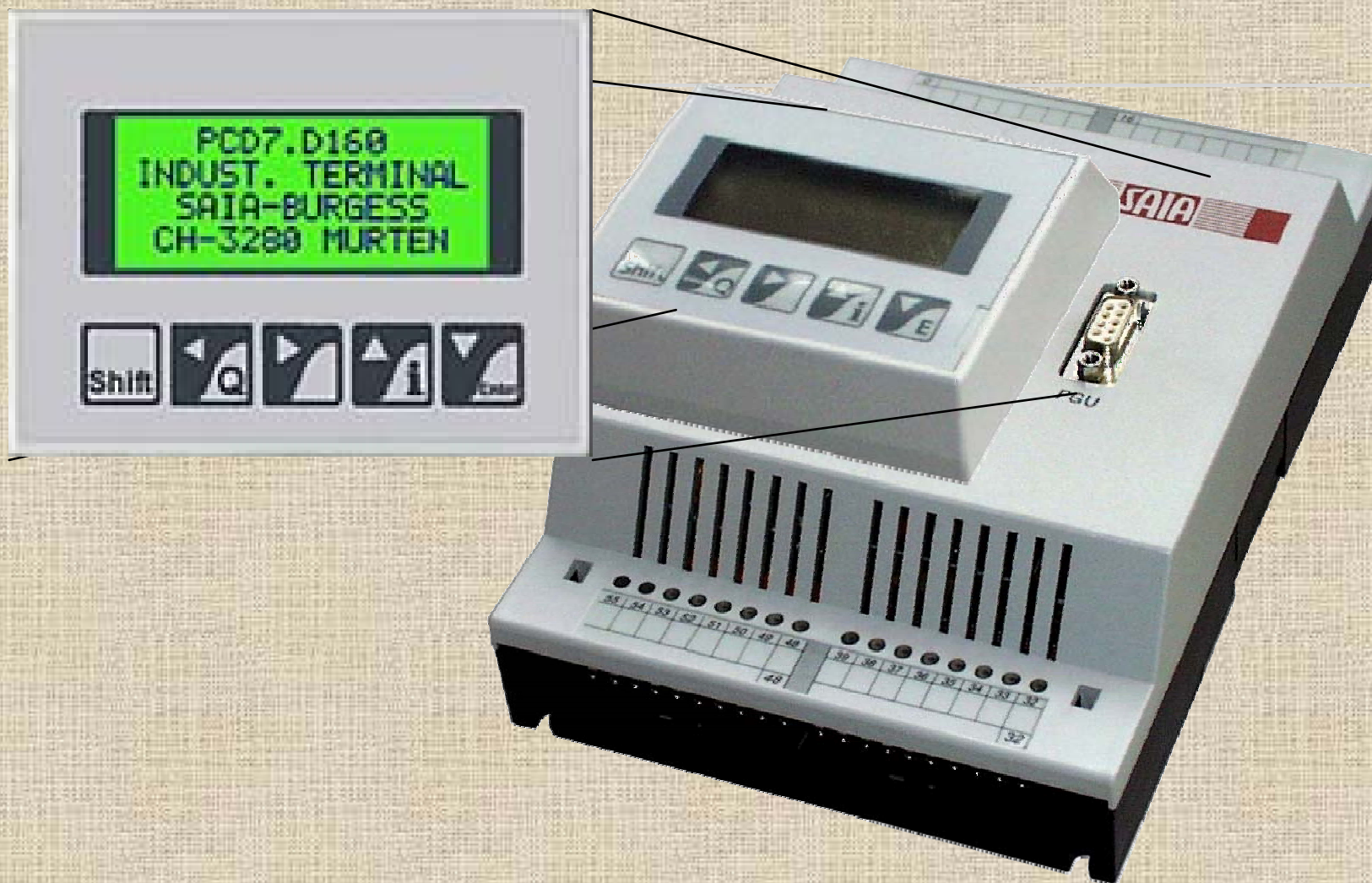
max. 5120 I/O

max. 512 I/O

PCD4



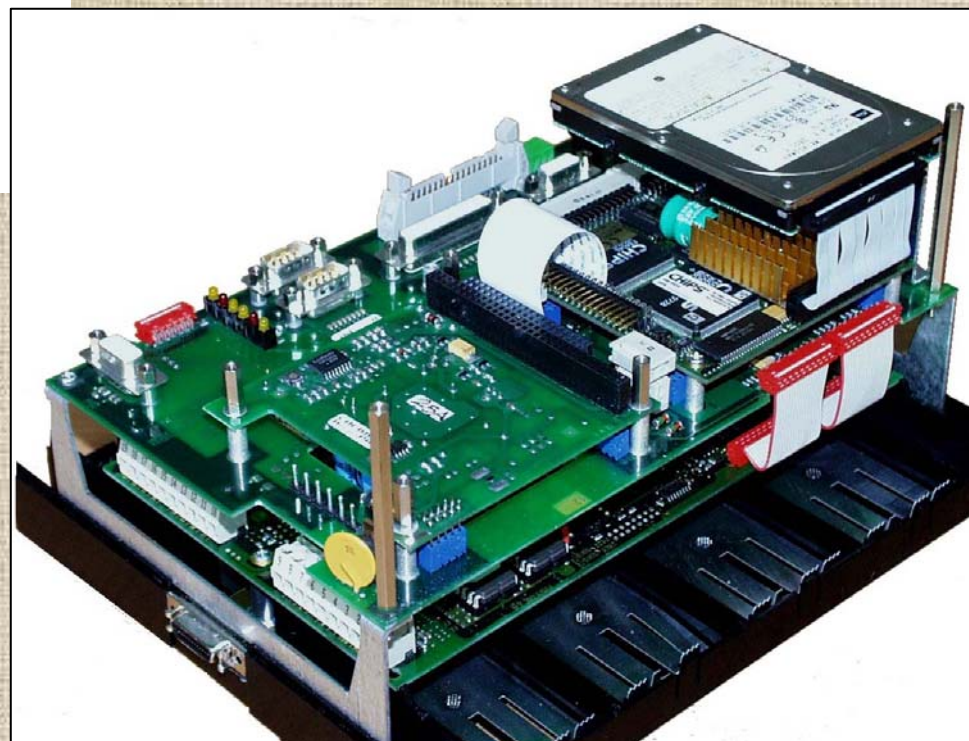
Rys. Rodzina sterowników PLC SAIA PCD1-PCD6



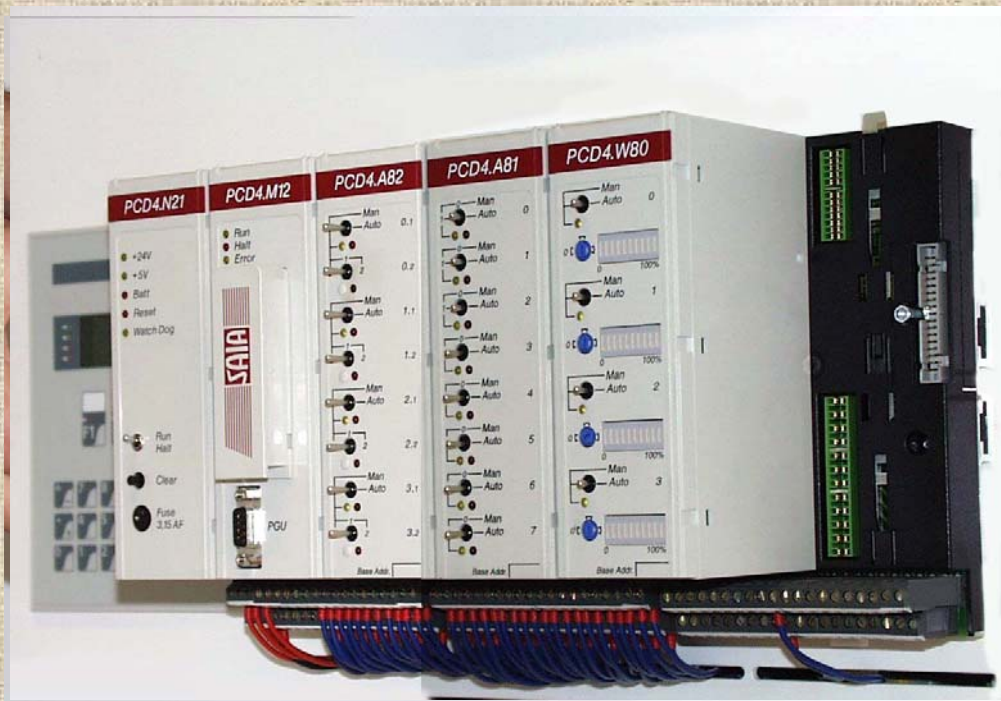
Rys. Najmniejszy sterownik PCD-1 (moduły IO jak do PCD-2)



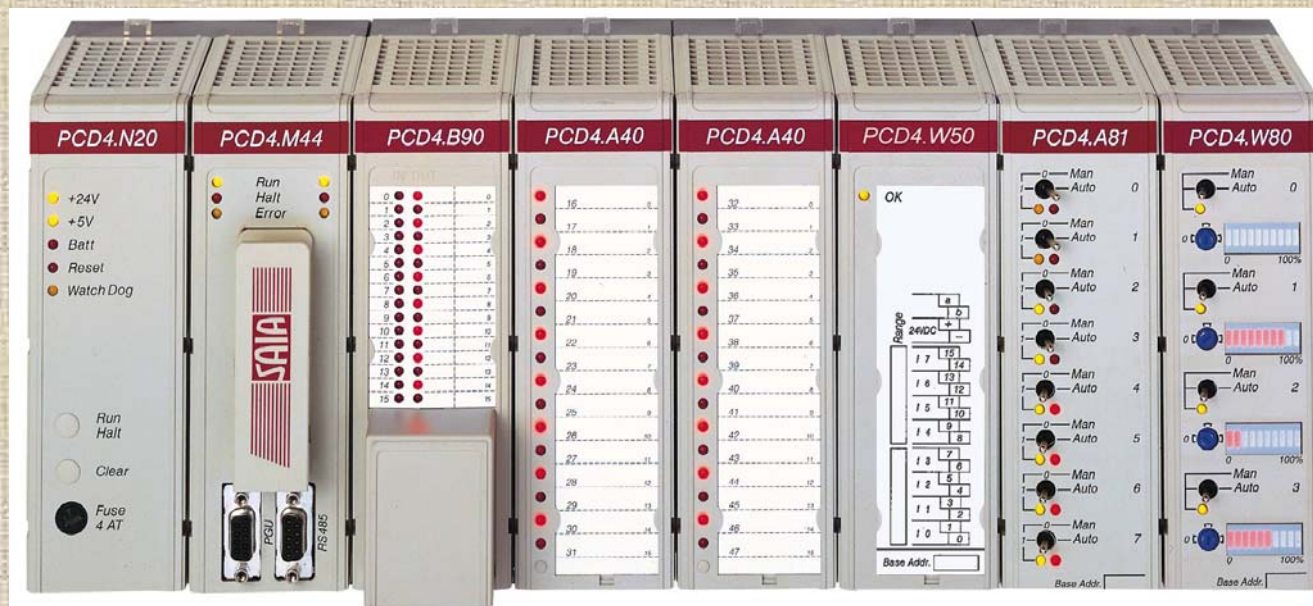
Rys. Sterownik PCD-2 (CPU M220) z modułem komputera PC-104

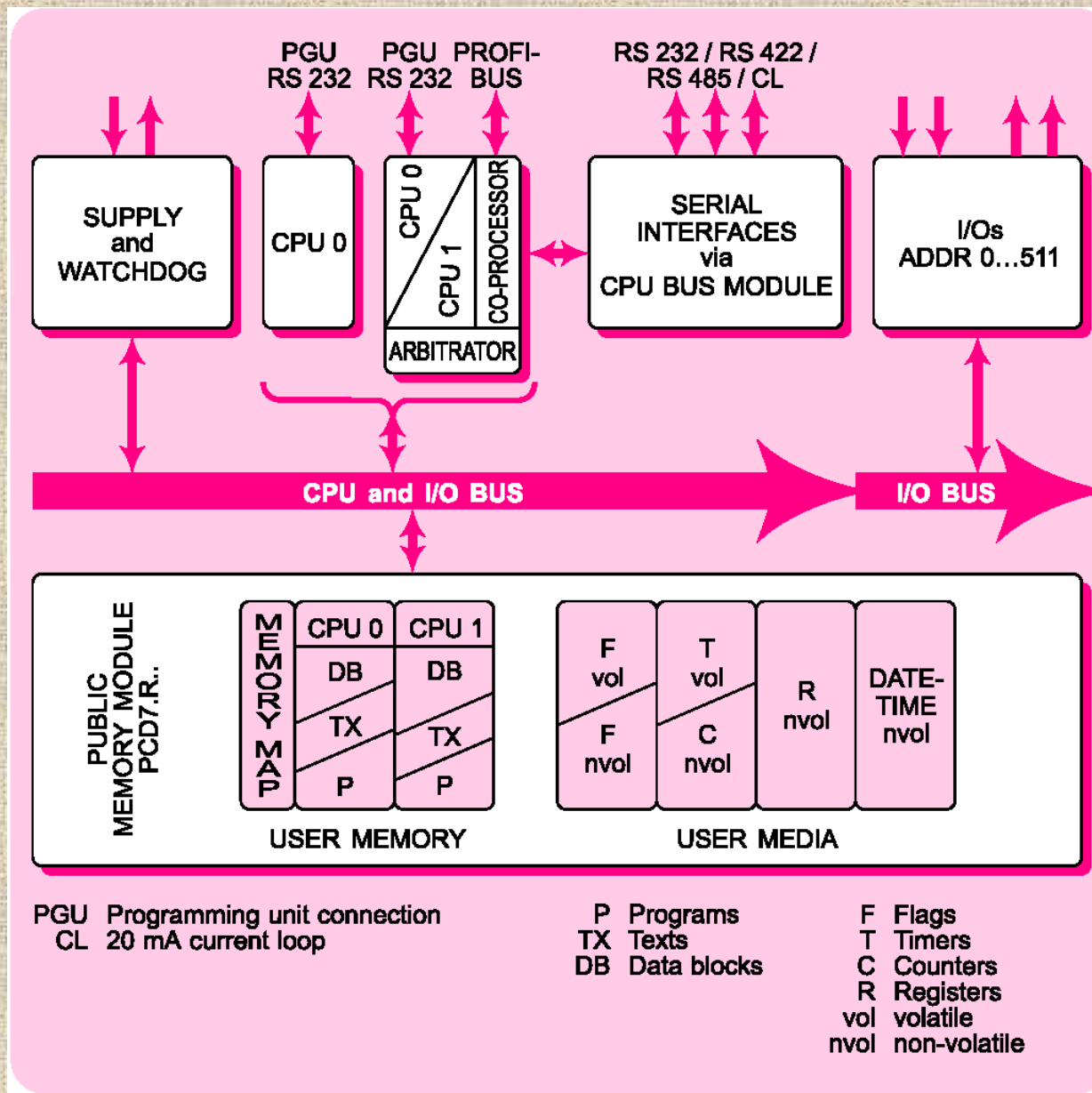


Komputerowe układy sterowania



Rys. Montaż modułów sterownika PCD-4





Rys. Schemat blokowy sterownika PCD-4

Performance range of the SAIA®PCD4

Processor module	PCD4.M110	PCD4.M..5
Number of I/Os	up to 512 I/Os or 32 I/O modules ¹⁾	
User memory	up to 428 KBytes as RAM and/or EPROM for programs, texts and data blocks	
Processor	32 bit μ P 68000 or μ C 68340	
Processing time ²⁾	6...10 μ s bit processing 35...60 μ s word processing (32 bit)	3.6...6 μ s bit processing 20...30 μ s word processing (32 bit)
Communications interfaces	up to 4 interfaces: RS 232, RS 422/RS 485, RS 422 and 20 mA current loop	
Network coupling	SAIA®S-Bus as master or slave (all PCD4.M..) PROFIBUS FMS as master or slave (..M445)	
Flags	8192 \times 1 bit, volatile or non-volatile, division programmable	
Registers	4096 \times 32 bit, non-volatile	

Rys. Podstawowe parametry sterownika PCD-4

Index registers	17 \times 13 bit (1 each per COB, 1 for all XOB)
Timers/counters	1600 timers volatile or counters non-volatile, T/C division programmable
Counting range	31 bit without sign bit (0...2 147 483 647)
Timing range	31 bit without sign bit (0...2 147 483 647 timebase pulses, programmable from 10 ms to 10 s)
Calculating ranges	integers: $-2\ 147\ 483\ 648 \dots +2\ 147\ 483\ 647$ ($-2^{31} \dots +2^{31}-1$); floating point numbers: $\pm 9.22337 \times 10^{18} \dots \pm 5.42101 \times 10^{-20}$; units: decimal, binary, BCD, hexadecimal or floating point
Date-time	time values: s/min/h, week/day of week, month/day of month, year; precision: better than 15 s/month; battery power reserve: >2 month

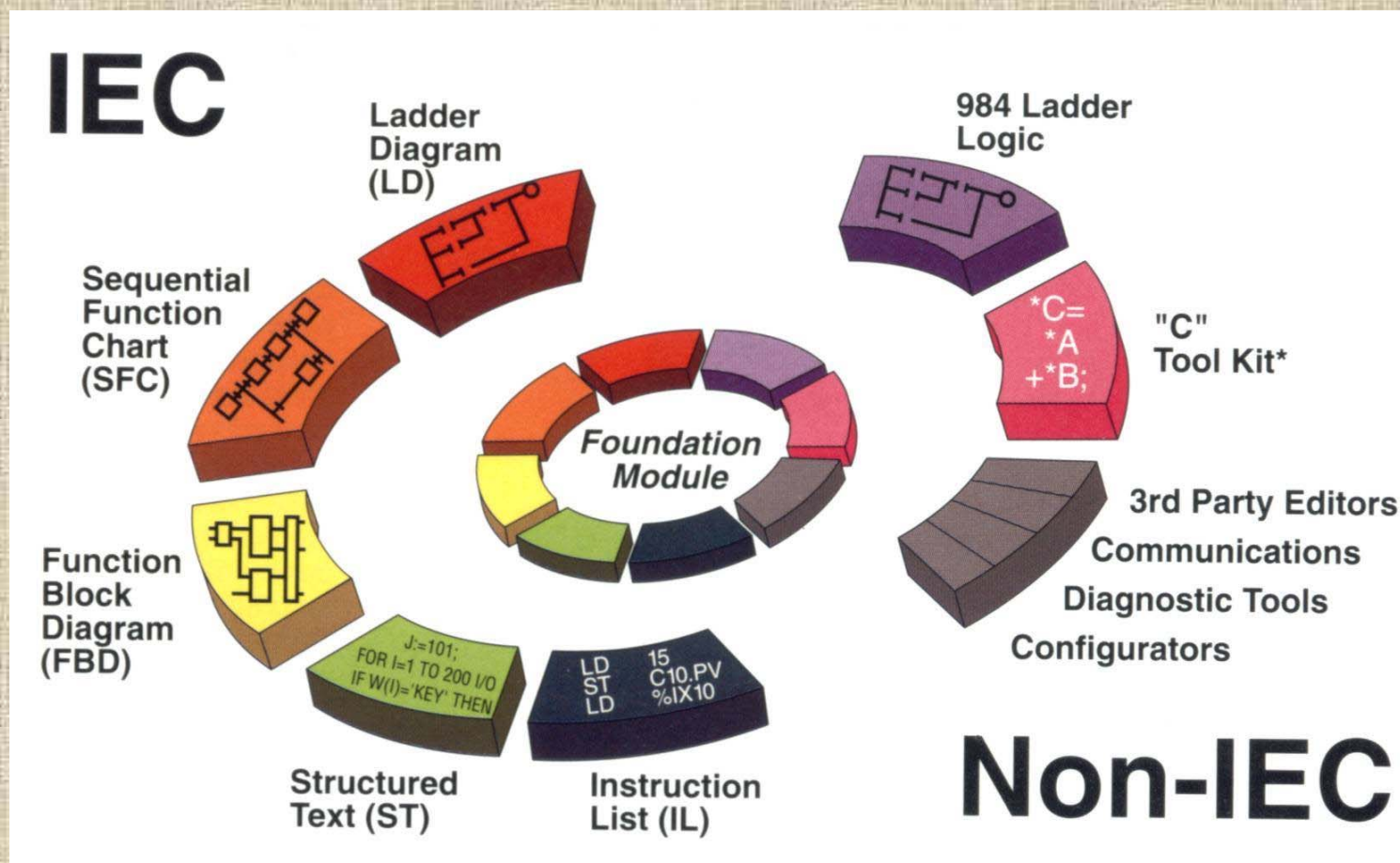
¹⁾ With 32 \times PCD4.B900 modules, an I/O capacity of 512 inputs and 512 outputs is achieved.

²⁾ Processing time is dependent on load at the communications interfaces.

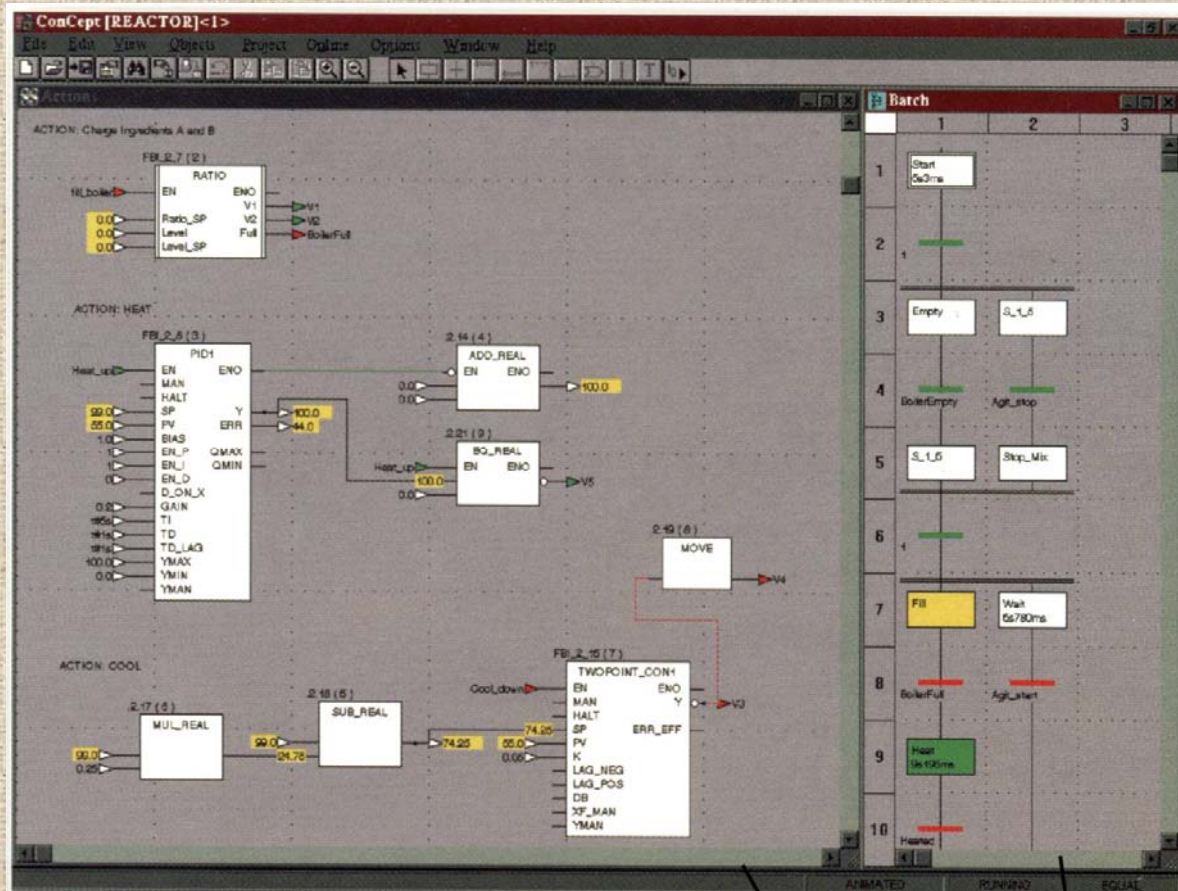


Rys. Wieloprocessorowy (do 6 CPU) sterownik PCD6. Wszystkie CPU mają sterowany przez „arbitratora” dostęp do „pamięci publicznej”. Każda CPU realizuje indywidualne zadania (np. sterowanie maszyną, komunikacja), ma prywatną pamięć i tablicę historii procesu oraz dostęp do zasobów wspólnych. Możliwa redundancja CPU ze zwielokrotnieniem czujników i elementów wykonawczych.

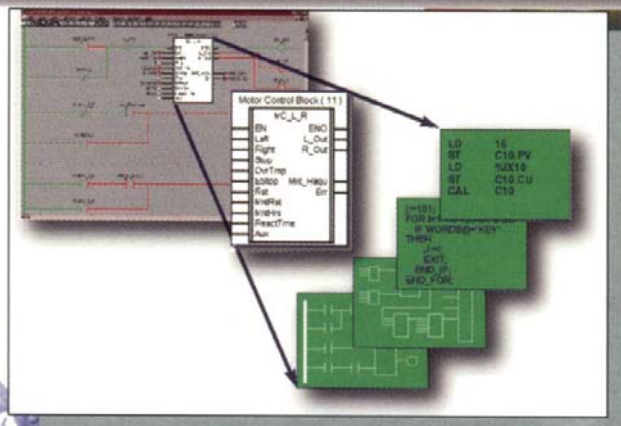
Języki programowania sterowników PLC



Rys. Języki programowania sterowników PLC wg normy IEC-1131-3: schemat drabinkowy LD, schemat bloków funkcyjnych FBD, graf stanów (akcje) i tranzycji (warunki) SFC, tekst strukturyzowany ST, lista instrukcji IL



Rys. Okna środowiska programowania sterowników Modicon - CONCEPT



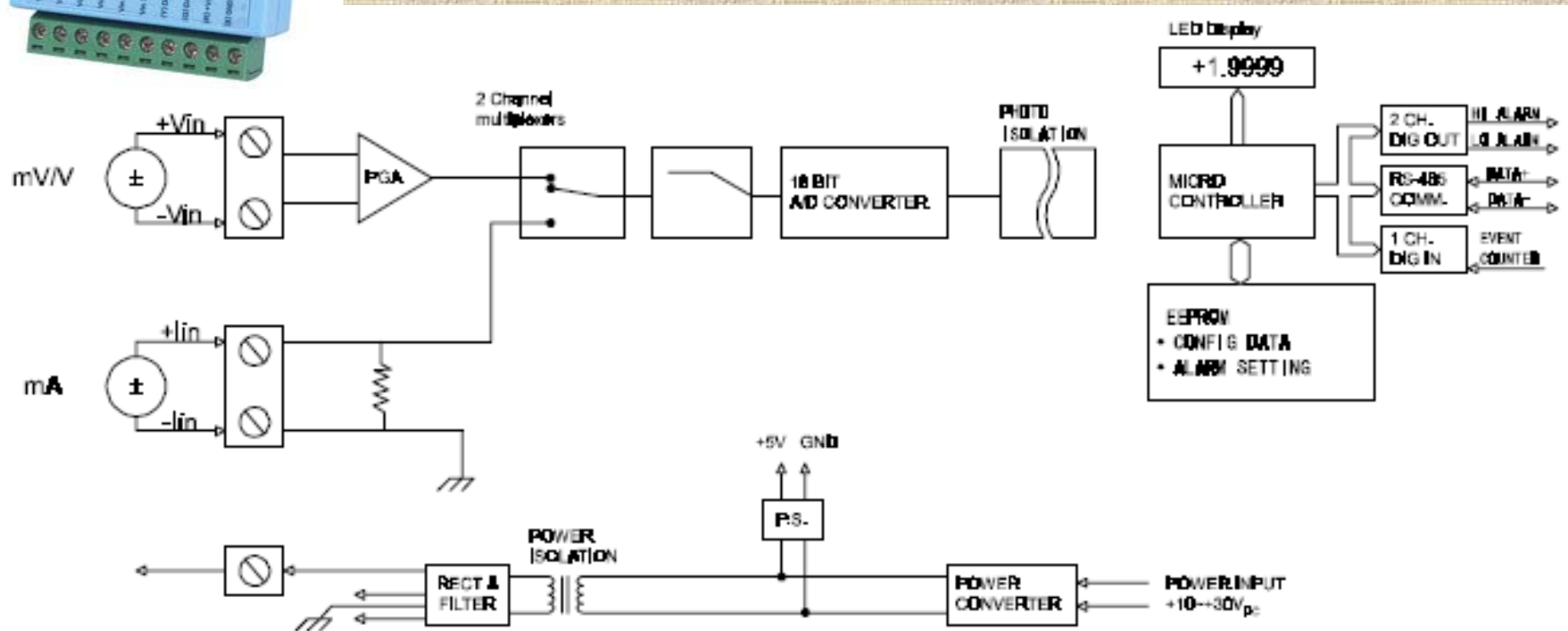
FBD

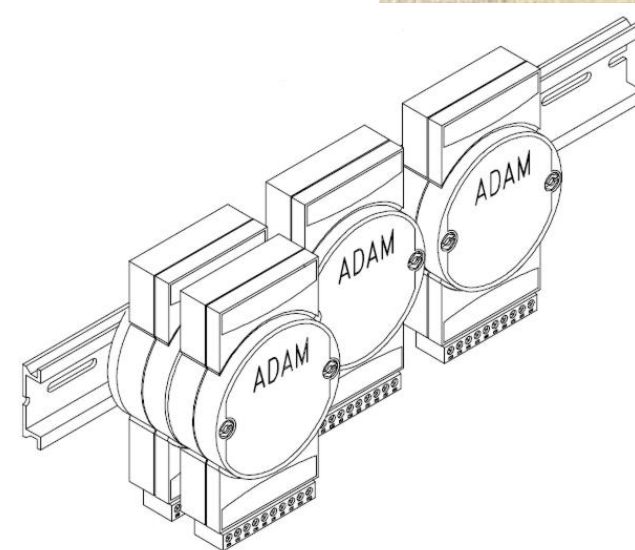
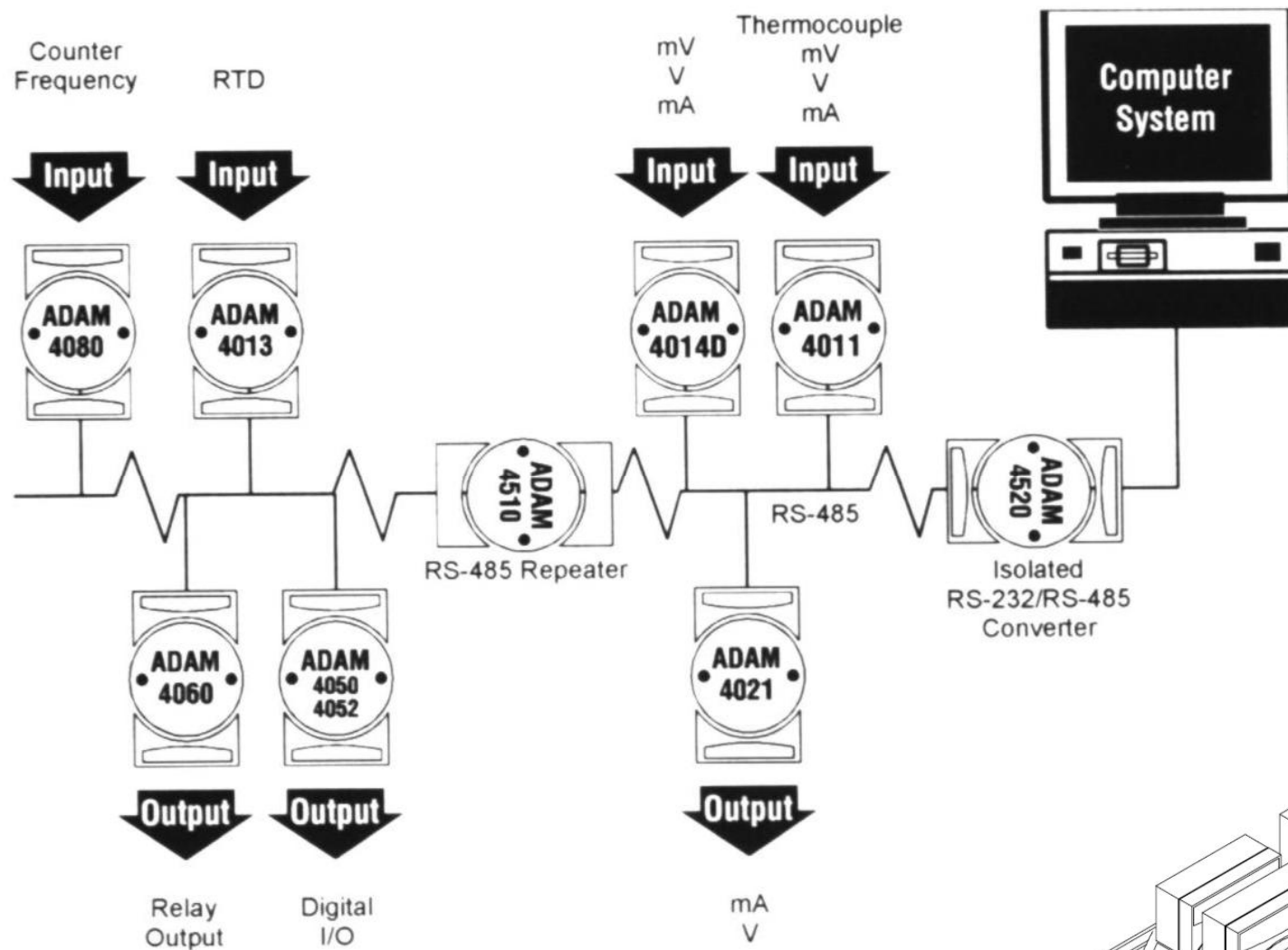
SFC

Inteligentne moduły RIO (Remote Input-Output)

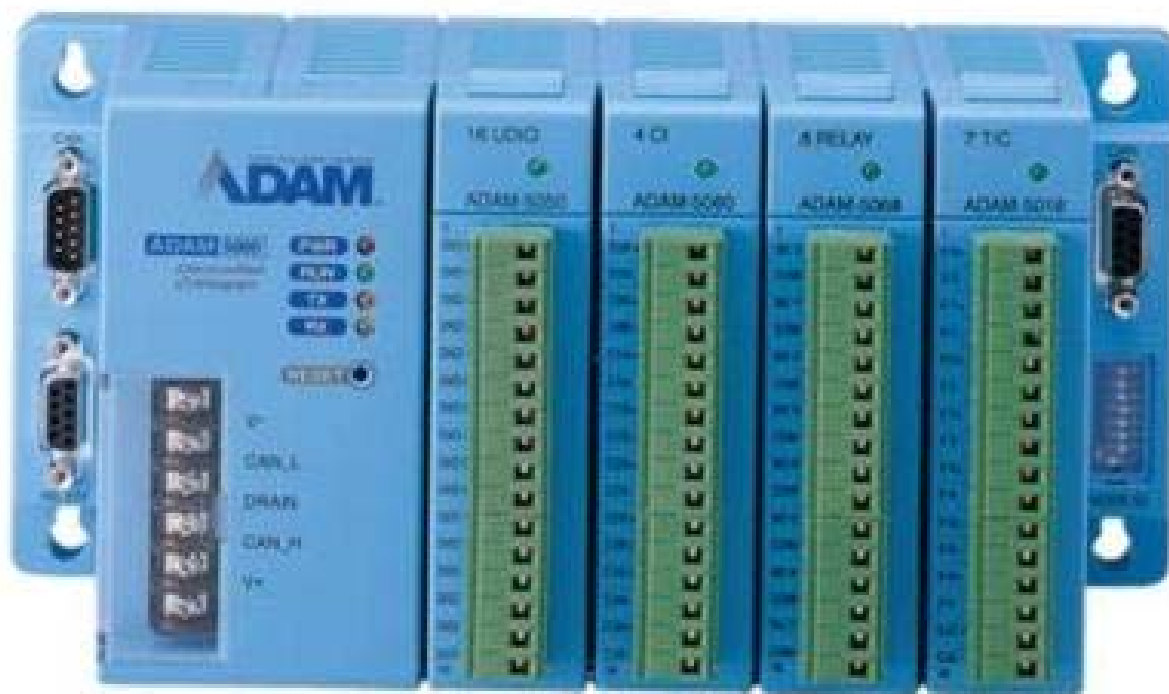


Rys. Moduł RIO serii ADAM 4000 z interfejsem RS-485 (moduł wejść analogowych 4117 (16-bit) i schemat blokowy). Podobne moduły serii ADAM-6000 są wyposażone w interfejs 10/100 Base-T Ethernet (obsługują protokoły Modbus/TCP, TCP/IP, UDP i HTTP).



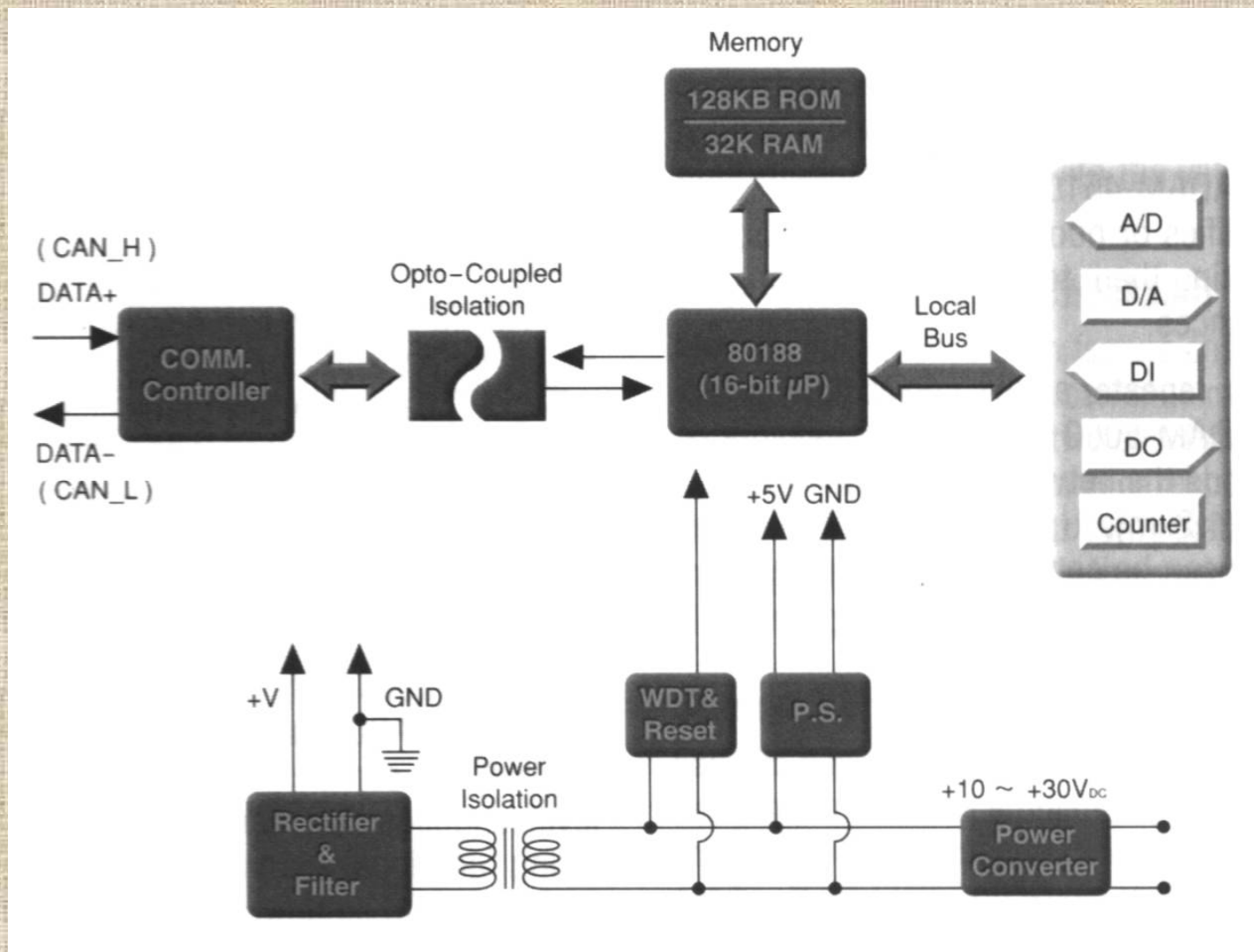


Rys. Sieć modułów RIO ADAM 4000 z interfejsem RS-485 (do 256 węzłów z repeaterami) i montowanie modułów na szynie DIN-35



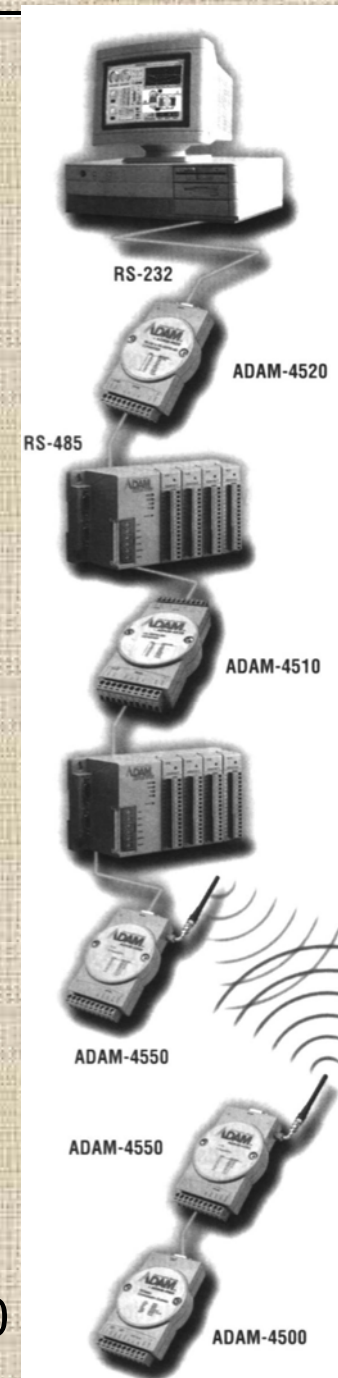
Rys. System rozproszonej akwizycji danych i sterowania ADAM-5000 (interfejs RS-485 lub CAN)

Rys. Moduły serii ADAM-5000: ADAM-5017 - 8 AI z optoizolacją, ADAM-5069 – 8 DO przekaźnikowych



Rys. Schemat blokowy modułu serii ADAM-5000 z interfejsem CAN (zamiast RS-485)

Rys. Sieć RIO RS-485 złożona z modułów ADAM-4000 i 5000



Specifications

Control System

- **CPU** 16-bit 80188 microprocessor
- **I/O Slots** ADAM-5000/485: 4
ADAM-5000E: 8
- **LED Indicators** Power, CPU, communications
- **Watchdog Timer** 1.6 sec. (System)

Communications

- **Command Format** ASCII command/response protocol, Modbus/RTU
- **Communication** RS-485: 1.2 km (4000 feet)
- **Distance**
- **Data Format** Asynchronous. 1 start bit, 8 data bits, 1 stop bit, no parity
- **Network Protocols** Programming link: RS-232 (3-wire: TX, RX, GND)
Communication: RS-485 (2-wire)
- **Reliability Check** Communication error checking with checksum
- **Max. Nodes** 128 (in RS-485 daisy-chain network)
- **Speeds (kbps)** 1.2, 2.4, 4.8, 9.6, 19.2, 38.4, 57.6, and 115.2

Power

- **Power Consumption** 3 W @ 24 Vdc (ADAM-5000/485)
(not including I/O modules)
4.0 W @ 24 Vdc (ADAM-5000E)
(not including I/O modules)
- **Power Input** Unregulated 10 ~ 30 V_{DC}

Protection

- **Communication Line Isolation** 2500 V_{DC} (ADAM-5000/485)
3000 V_{DC} (ADAM-5000E)
- **Communication Power Isolation** 3000 V_{DC}
- **I/O Module Isolation** 3000 V_{DC}
- **Transient Protection** RS-485 communication lines, power input
- **Power Reversal Protection** Yes

General

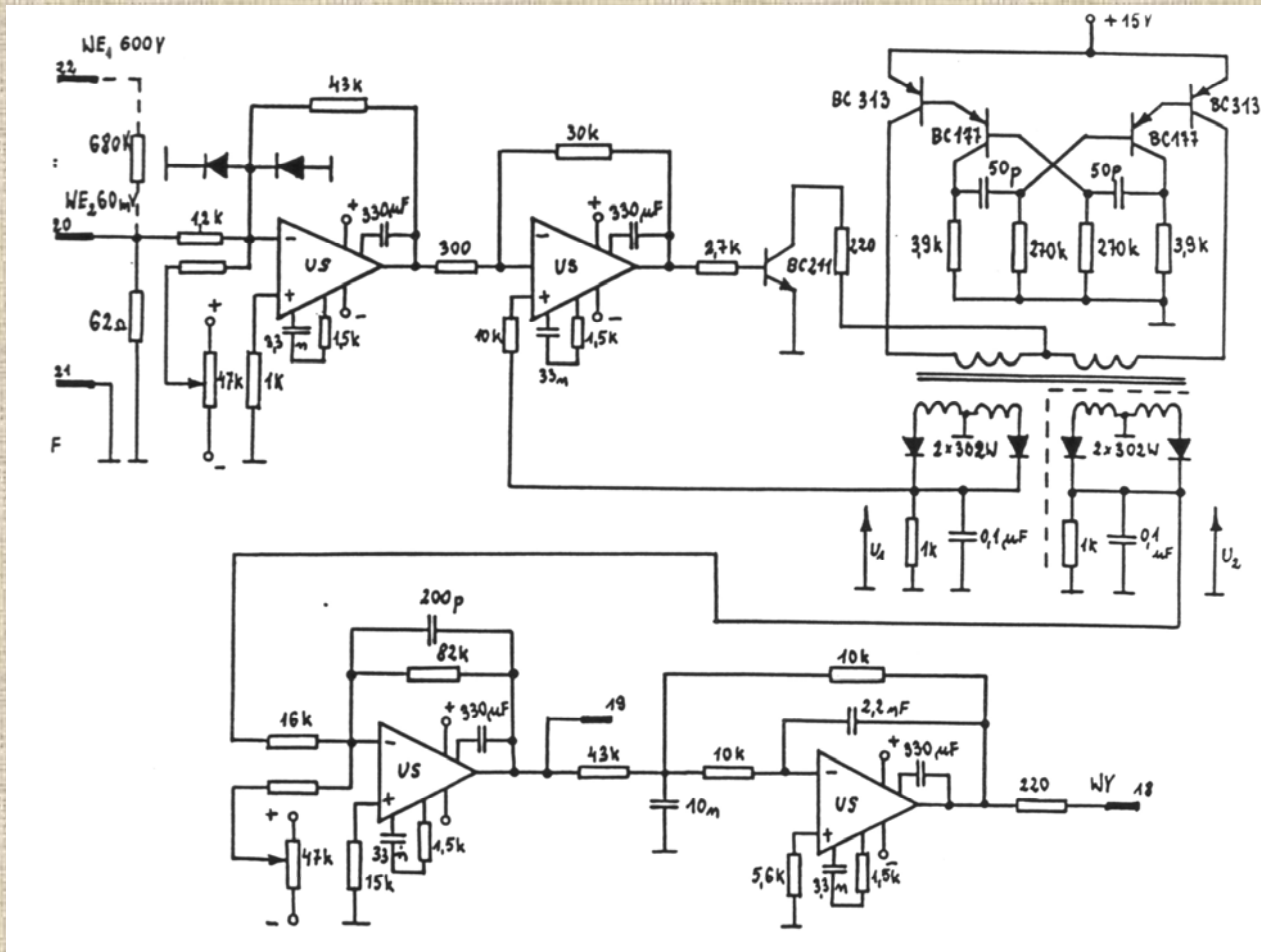
- **Certifications** CE, FM
- **Connectors** 1 x DB9-M/DB9-F/screw terminal for RS-485 (communication)
1 x DB9-F for RS-232 (configuration)
1 x Screw-terminal for power input
- **Dimensions (WxHxD)** 4-slot: 231 x 110 x 75 mm
8-slot: 355 x 110 x 75 mm
- **Enclosure** ABS+PC
- **Mounting** DIN 35 rail, wall, rack (with mounting kit)

Environment

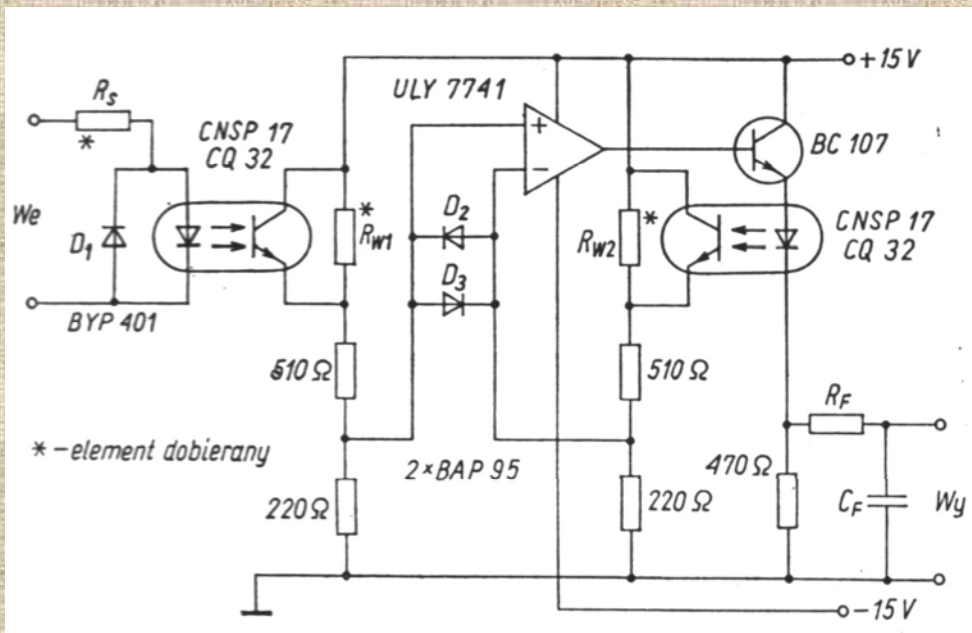
- **Humidity** 5 ~ 95%, non-condensing
- **Operating Temperature** -10 ~ 70° C (14 ~ 158° F)
- **Storing Temperature** -25 ~ 85° C (-13 ~ 185° F)

Rys. Najważniejsze specyfikacje systemu ADAM-5000
(interfejs RS-485 lub CAN)

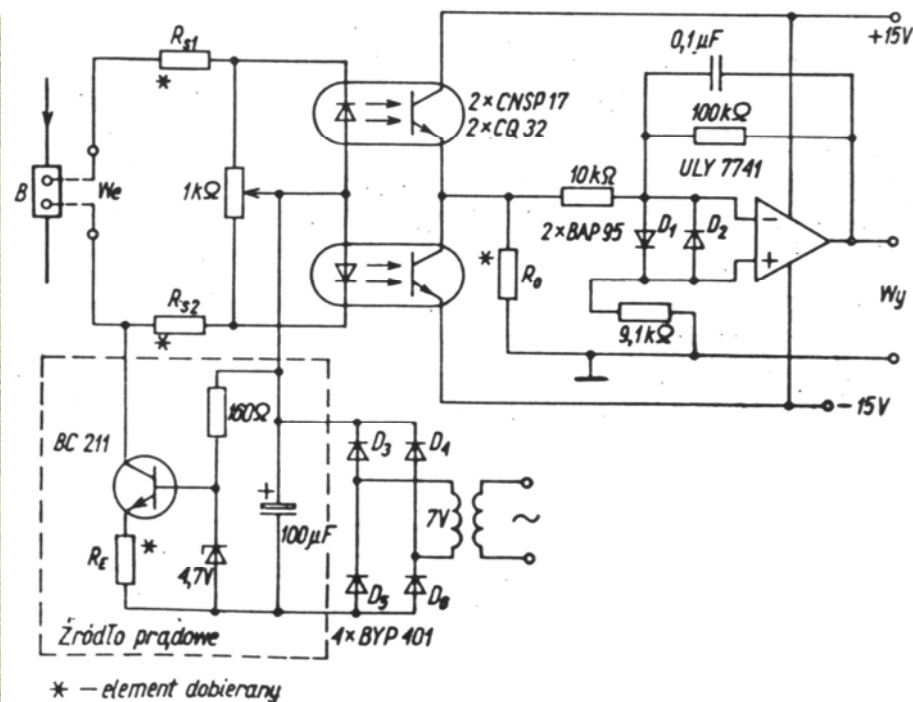
Typowe obwody układów wejściowych sterowników



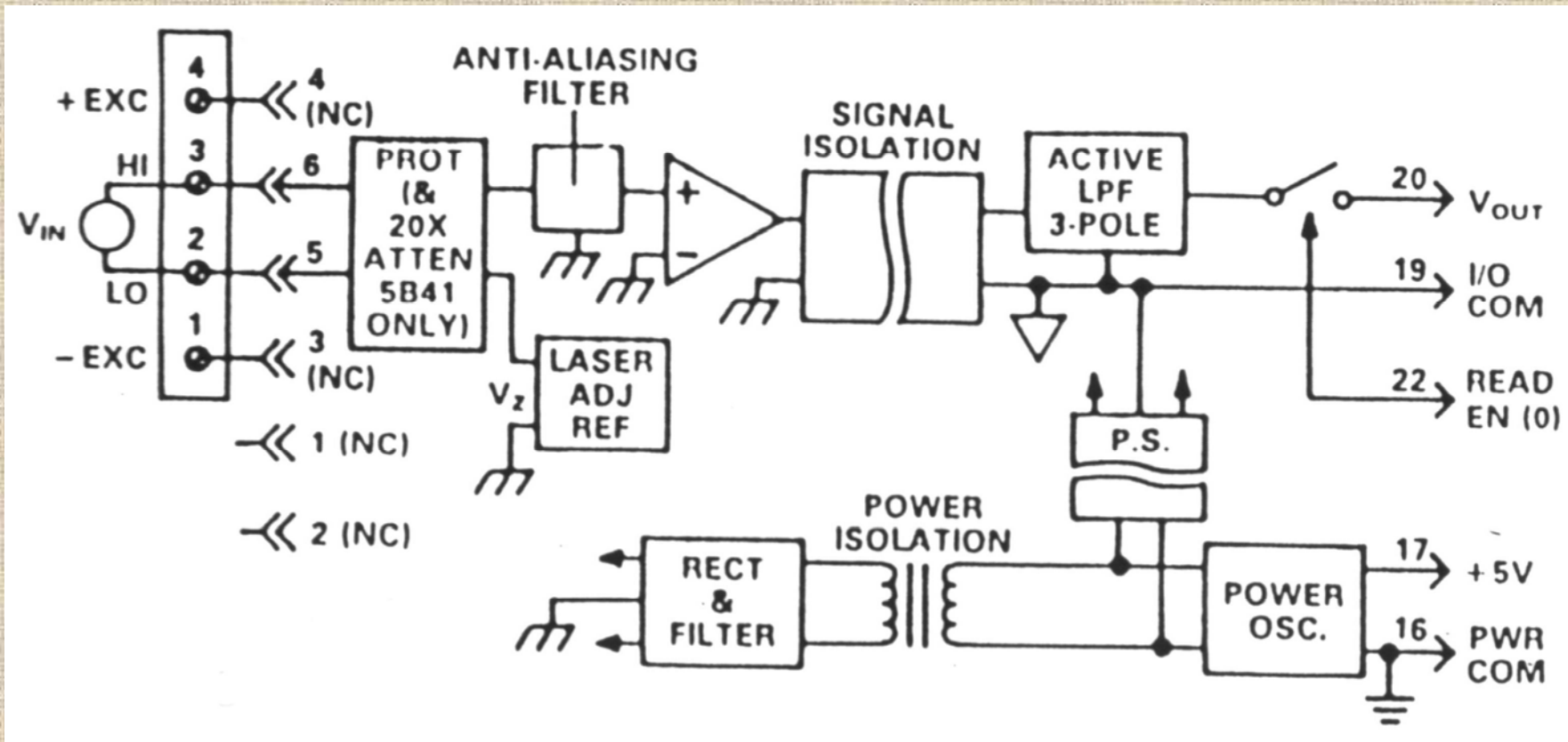
Rys. Przetwornik pomiarowy prądu z multiwibratorem astabilnym i separacją magnetyczną



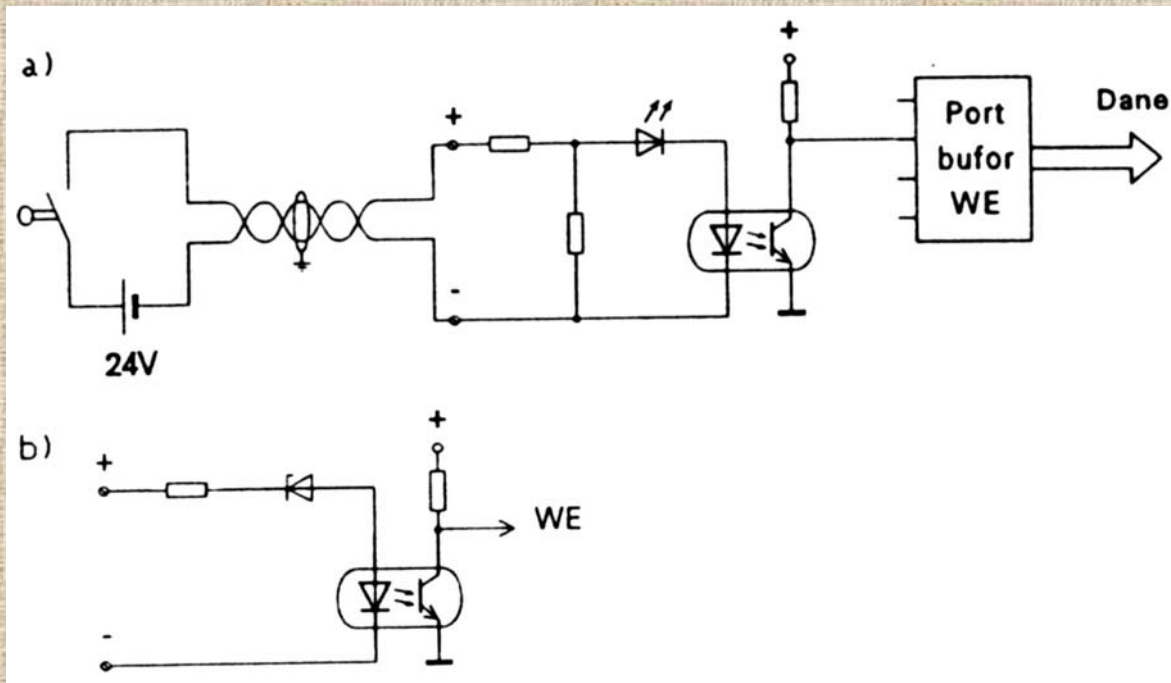
Rys. Przetwornik pomiarowy napięcia z separacją optoelektroniczną i linearyzującym sprzężeniem zwrotnym



Rys. Przetwornik pomiarowy prądu z obwodem różnicowym

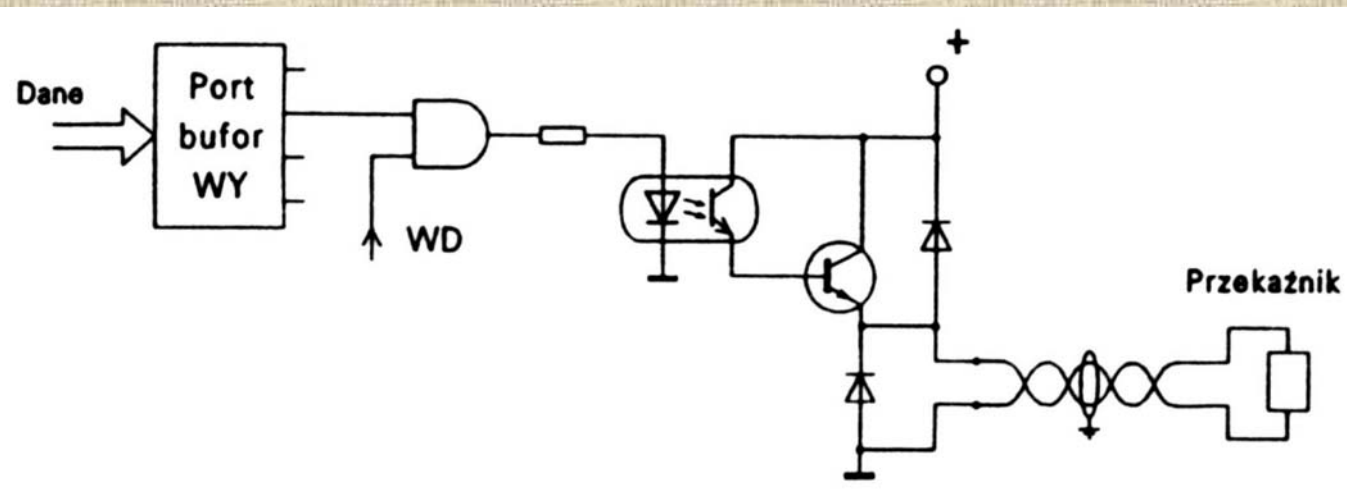


Rys. Schemat modułu dopasowania i separacji wejść analogowych 5B41 (zakresy $\pm 1 \dots \pm 20$ V, pasmo przenoszenia 10 kHz, dokładność 0.05% FSR ± 10 mV)



Rys. Obwody wejść binarnych: a) z dzielnikiem rezystancyjnym, b) z diodą Zenera

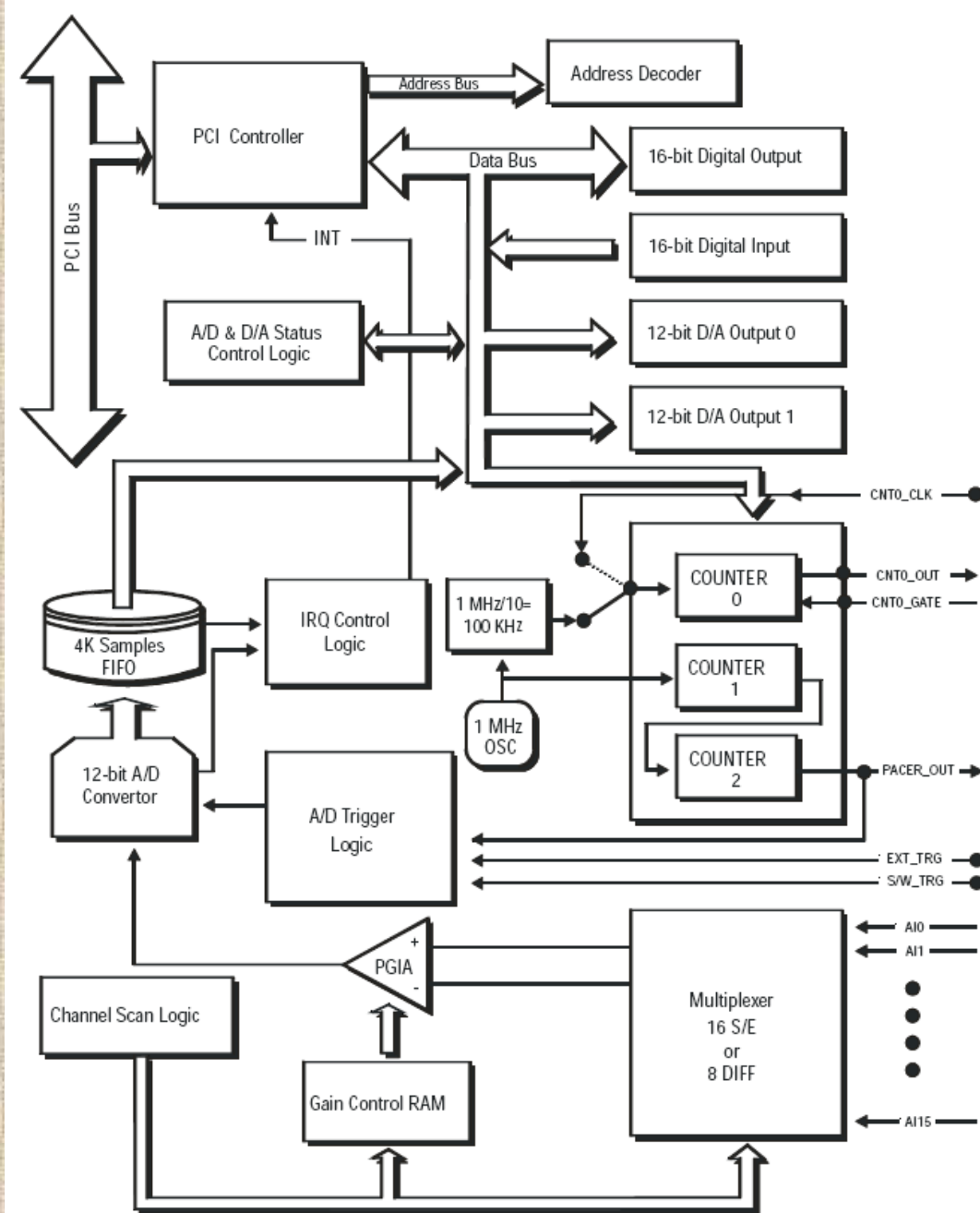
Rys. Typowy obwód wyjścia binarnego przekaźnikowego



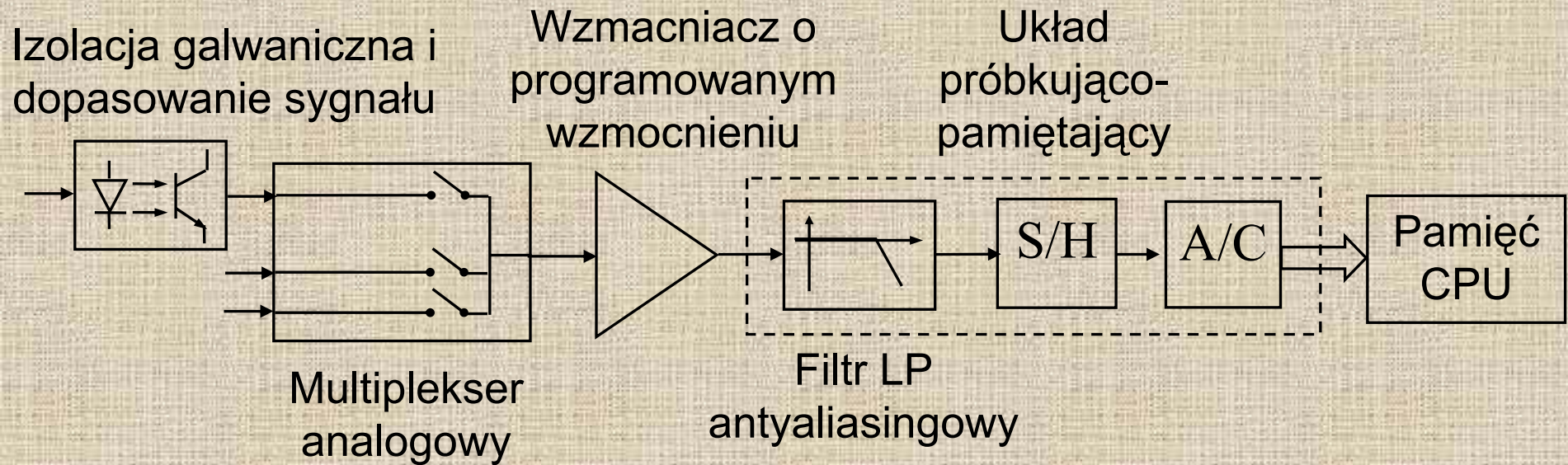
Uniwersalna karta akwizycji danych (DAQ) i sterowania do PC



Rys. Uniwersalna karta DAQ (akwizycji danych i sterowania) PCI-1710 HG:
16 single-ended (0-10...0.01V) or 8 differential ($\pm 10 \dots \pm 0.005V$) AI, 12-bit
A/D converter, with up to 100 kHz sampling rate, Programmable gain,
Automatic channel/gain scanning, Onboard FIFO memory (4096 samples),
Two 12-bit AO ($\pm 5, \pm 10V$), 16 DI and 16 DO, Onboard programmable counter

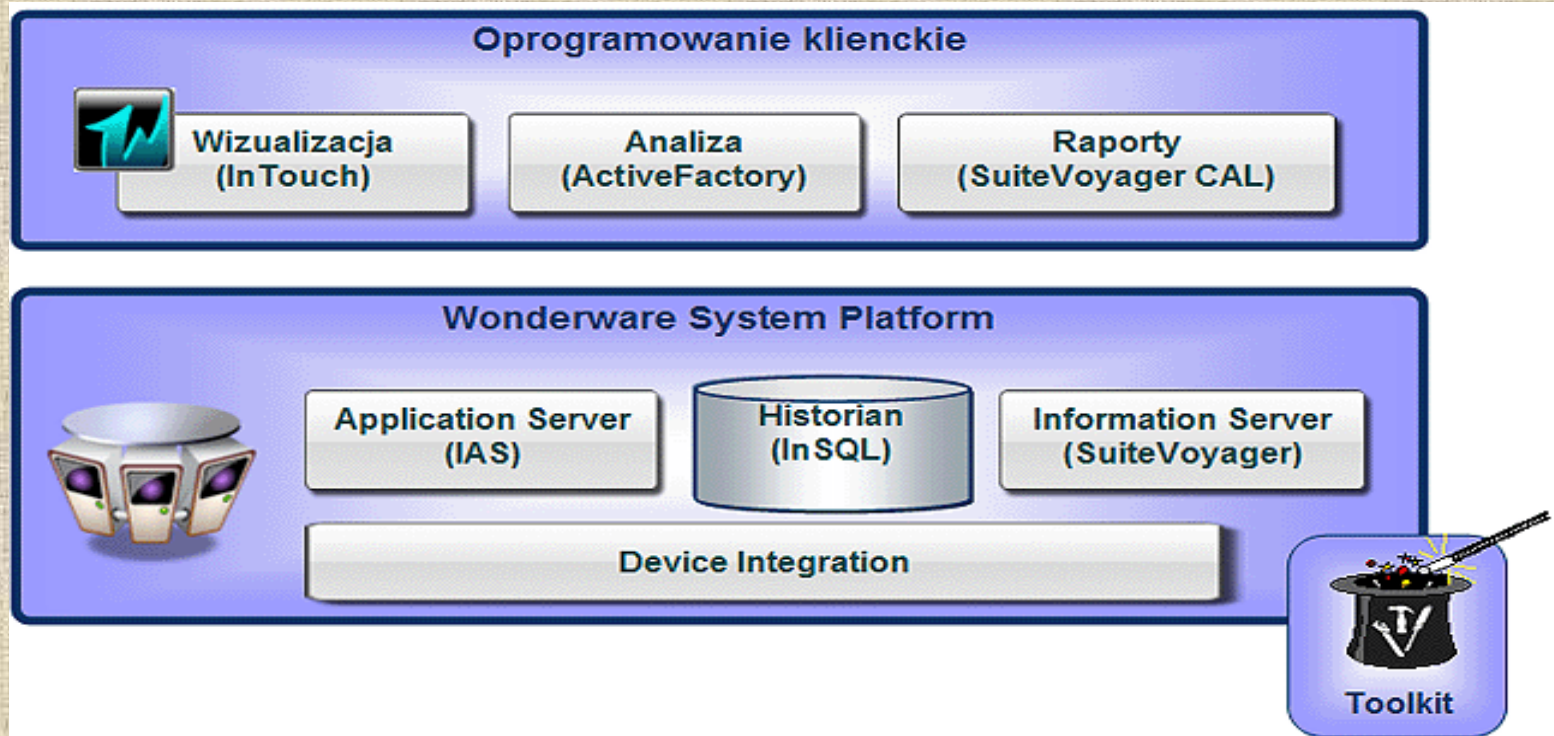


Rys. Schemat blokowy karty DAQCI-1710 HG



Rys. Elementy elektroniczne wielokanałowego toru przetwarzania analogowo-cyfrowego sygnału

Platforma Systemowa Wonderware - przykład oprogramowania SCADA

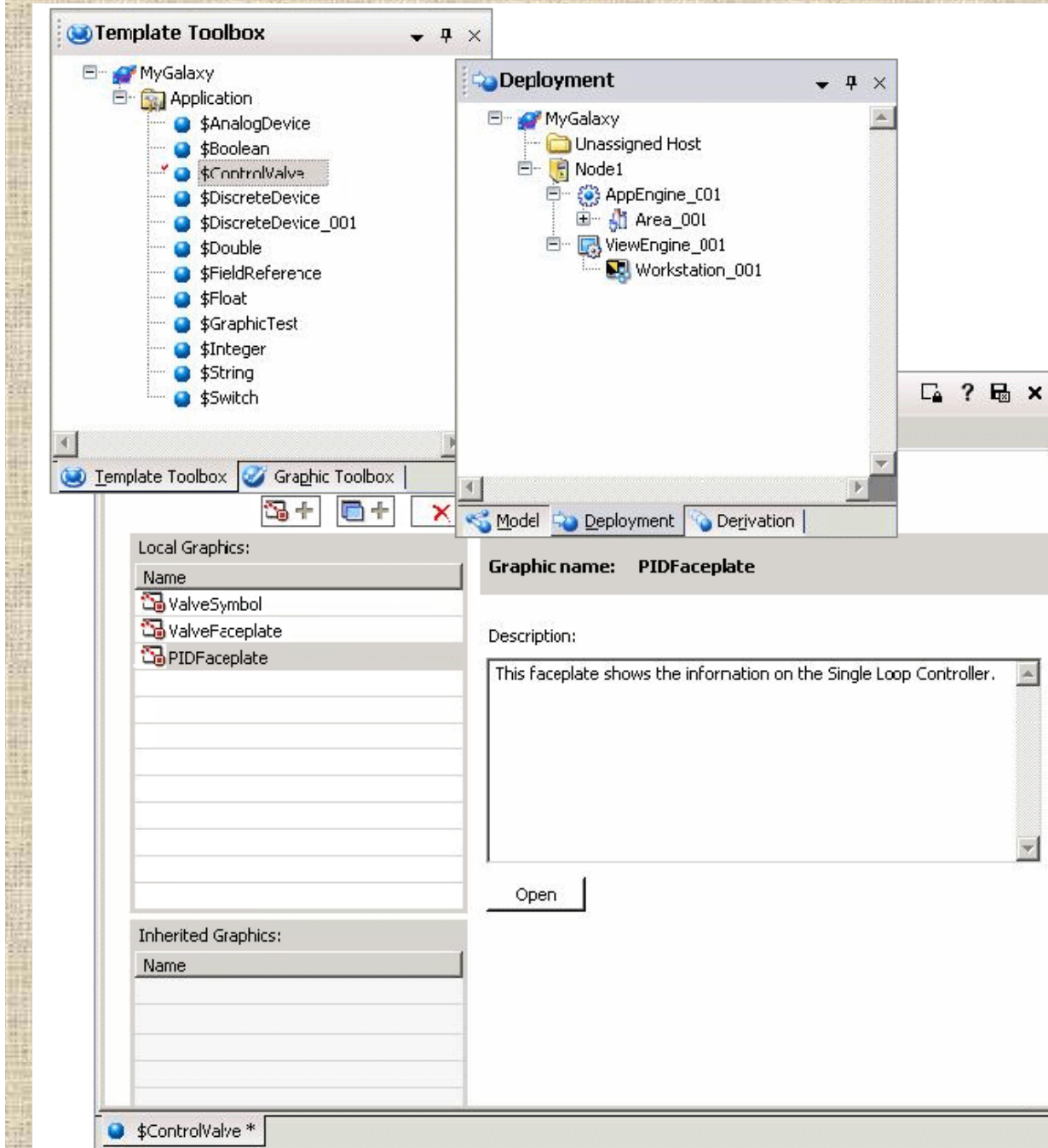


ArchestrA jest zaawansowaną architekturą z zakresu automatyki przemysłowej i informatyki - sposobem na *integrację* aplikacji przemysłowych działających na platformach Microsoftu. ArchestrA to infrastruktura (zestaw usług) instalowana w systemie operacyjnym w oparciu o technologię .NET, która daje możliwość tworzenia własnych aplikacji przemysłowych. Aplikacje buduje się i konfiguruje „z klocków”. Na technologii ArchestrA bazuje **Platforma Systemowa Wonderware** stanowiąca podstawę do budowy nowoczesnych aplikacji przemysłowych. Platforma współpracuje z aplikacjami klienckimi takimi jak [InTouch](#) do wizualizacji operatorskiej czy aplikacje analityczno-raportowe np. ActiveFactory.

Application Server - produkt, który ogranicza konieczność programowania - umożliwia łatwe tworzenie, instalowanie i uruchamianie systemów automatyki. Aplikacje powstają przez dobór i konfigurację obiektów-szablonów z katalogu gotowych elementów. Za pomocą jednego edytora konfiguruje się całe obiekty, określa progi alarmowe parametrów obiektu, ustala, które z tych parametrów mają być logowane historycznie, a także projektuje dla każdego obiektu automatyki jedną lub więcej grafik.

Application Server zawiera: wspólną bazę danych zmiennych całej (wielostanowiskowej) aplikacji, konfigurację aplikacji, skrypty, alarmy i wszelkie mechanizmy pozwalające na zdalną dystrybucję i uruchamianie elementów aplikacji. W zakresie struktury, serwer można dowolnie konfigurować, projektant ma pełną swobodę decyzji co do architektury aplikacji: peer-to-peer czy klient-serwer.

Komputerowe układy sterowania



Rys. Aplikacje powstają przez dobór i konfigurację gotowych obiektów-szablonów (np. zawór, regulator) i mogą być rozmieszczone (*deployed*) w określonych węzłach (komputerach) systemu.

Historian zaprojektowany na bazie architektury [ArchestrA](#) i *Industrial SQL* wraz z programami raportowymi Wonderware ActiveFactory jest narzędziem do zbierania i analizy danych z produkcji. Produkt ten nie tylko daje całościowy wgląd w proces produkcyjny, ale umożliwia także przygotowywanie szczegółowych analiz dotyczących interesującego fragmentu produkcji oraz umożliwia wykorzystanie zarejestrowanych danych w innych systemach funkcjonujących w firmie, np. w systemach biznesowych.

Information Server jest rozwiązaniem do zarządzania dystrybucją informacji z systemów produkcyjnych HMI, SCADA, MES i innych. System agreguje w jednym miejscu dostęp do informacji poprzez *przeglądarki internetowe*. W celu poszerzenia i usprawnienia dostępu użytkowników do informacji znajdujących się w przedsiębiorstwie, korzysta z najnowszych technologii internetowych - WebParts i usług internetowych (Web service).

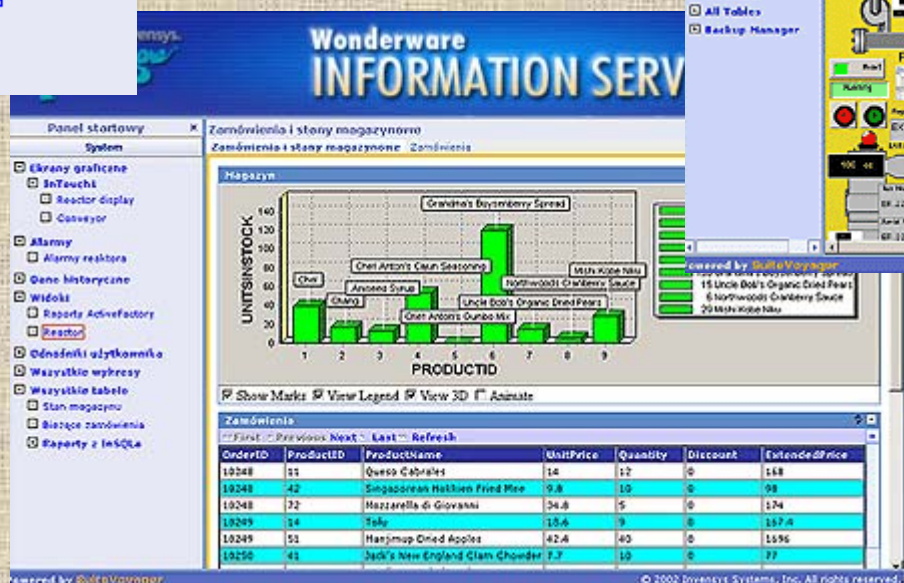
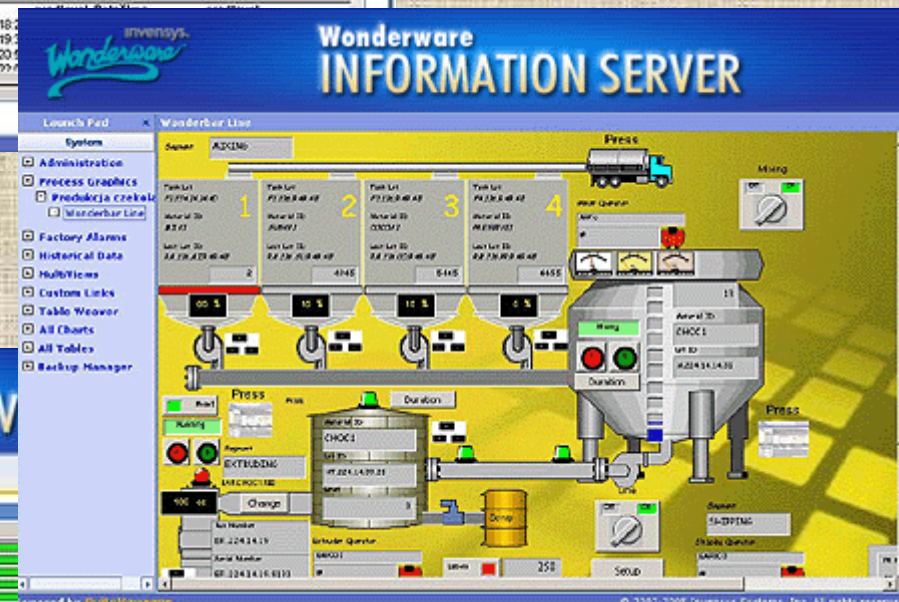
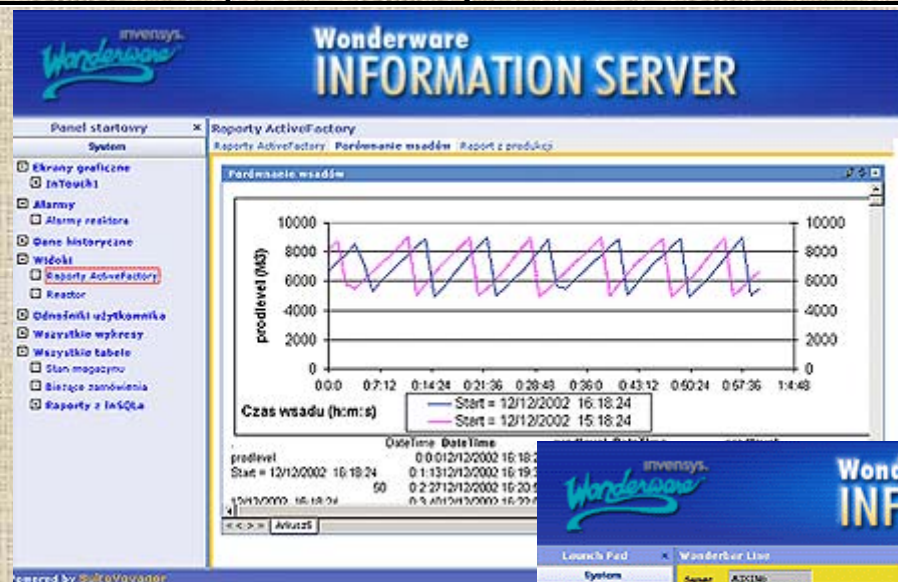
Komputerowe układy sterowania

invenysys.
Wonderware

Panel startowy x

System

- Ekran graficzny
 - InTouch1
- Alarmy
 - Alarmy reaktora
- Dane historyczne
- Widoki
 - Raporty ActiveFactory
 - Reaktor
- Odnośniki użytkownika
- Wszystkie wykresy
- Wszystkie tabele
 - Stan magazynu
 - Bieżące zamówienia
 - Raporty z InSQLa



InTouch - przemysłowe oprogramowanie zaprojektowane do wizualizacji oraz kontroli procesów produkcyjnych, w pełni zgodny z wytycznymi dla systemów klasy **SCADA** (*Supervisory Control And Data Acquisition*) oraz **HMI** (*Human-Machine-Interface*). Oferuje łatwe w użyciu i intuicyjne środowisko do projektowania aplikacji oraz rozległą funkcjonalność umożliwiającą szybkie projektowanie, testowanie oraz wdrażanie systemów udostępniających użytkownikom dane bezpośrednio z systemów sterownia i produkcji. Jest komponentem pakietu Wonderware Development Studio wykorzystującym nowatorską technologię [ArchestrA](#). Stosowany od prostych aplikacji jednostanowiskowych, poprzez rozproszone systemy sieciowe o architekturze klient/serwer. Może być zintegrowany z [Platformą Systemową Wonderware](#) i korzystać z usług Terminal Services.

W celu przyspieszenia procesu projektowania graficznej strony aplikacji można korzystać z kontrolek ActiveX, kontrolek .NET, gotowej biblioteki symboli ArchestrA Graphics zawierającej ponad 500 gotowych obiektów graficznych z możliwością ich edycji i tworzenia szablonów obiektów graficznych.

InTouch udostępnia użytkownikom dostęp do najnowszych protokołów komunikacyjnych jak SuiteLink, OPC oraz znanych już fast/net DDE.

Valve_Hand *

Graphic Edit View Arrange Format Special Help

Save and Close [Icons]

Arial 12 [B I U A A] [Color] [Fill] [Stroke] [Align] [Text] [Image]

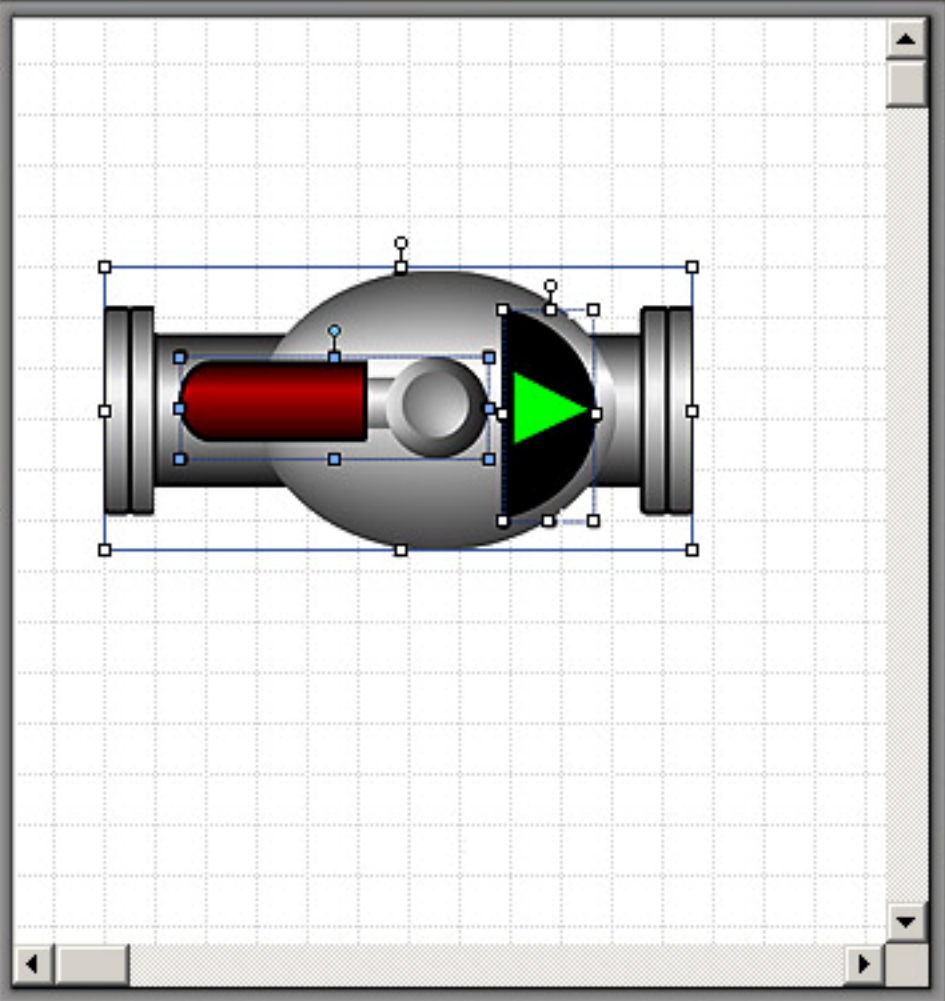
[Icons]

Tools

[Icons]

Elements

- + [Icon] Handle
- + [Icon] Indicator
- + [Icon] Housing



Properties

[Icons]

Appearance

X	
Y	
Width	
Height	
Angle	0
AbsoluteOrigin	
RelativeOrigin	
Transparency	0
Locked	False

Runtime Behavior

Enabled	True
TabOrder	0
TabStop	True
TreatAsIcon	False
Visible	True

Komputerowe układy sterowania

InTouch - WindowViewer - \\GAINTELF\DEMOAPP2-SAAREACTORDEMO

File Log: Special Development

BATCH NUMBER: 00000014

REACTOR LEVEL (L): 0

CONCENTRATE (%): 00000045

REACTOR TEMP (°C): 00000058

FILLING

C. INJECTION

MIXING

TRANSFER

Time	Type	Name	Provider	Alarm Comment
03:58:37 PM	Comm	Demo_Area B...	Gateway	Lost alarm co...
10:28:07 AM	Lo	Reactor_MTC...	Gateway	This is the Re...

10:38:07

Logged on Operator

Administrator

Security

BATCH REACTOR

WATER

CONCENTRATE

STEAM

MODE AUTO MAN

REACTOR 0.0 Litre

TRANSFER

PRODUCT STORAGE

Tank Level 0.924.0 Litre

OUTPUT

Temperature

Level

Batch in process

Auto

REACTOR

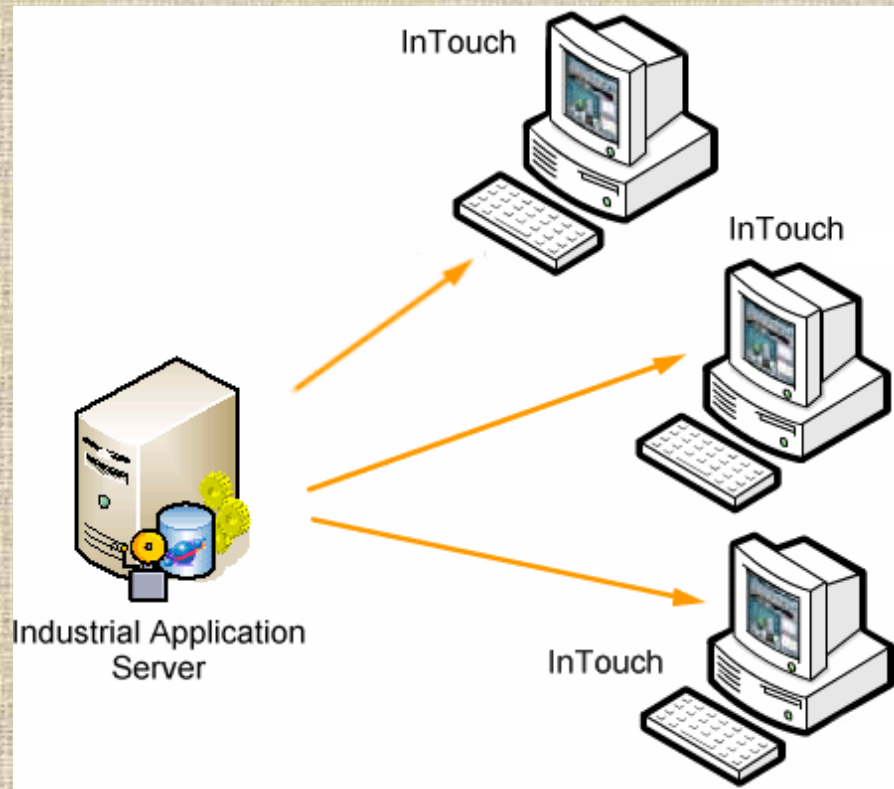
CONC. %

SYSTEM

MAINTENANCE

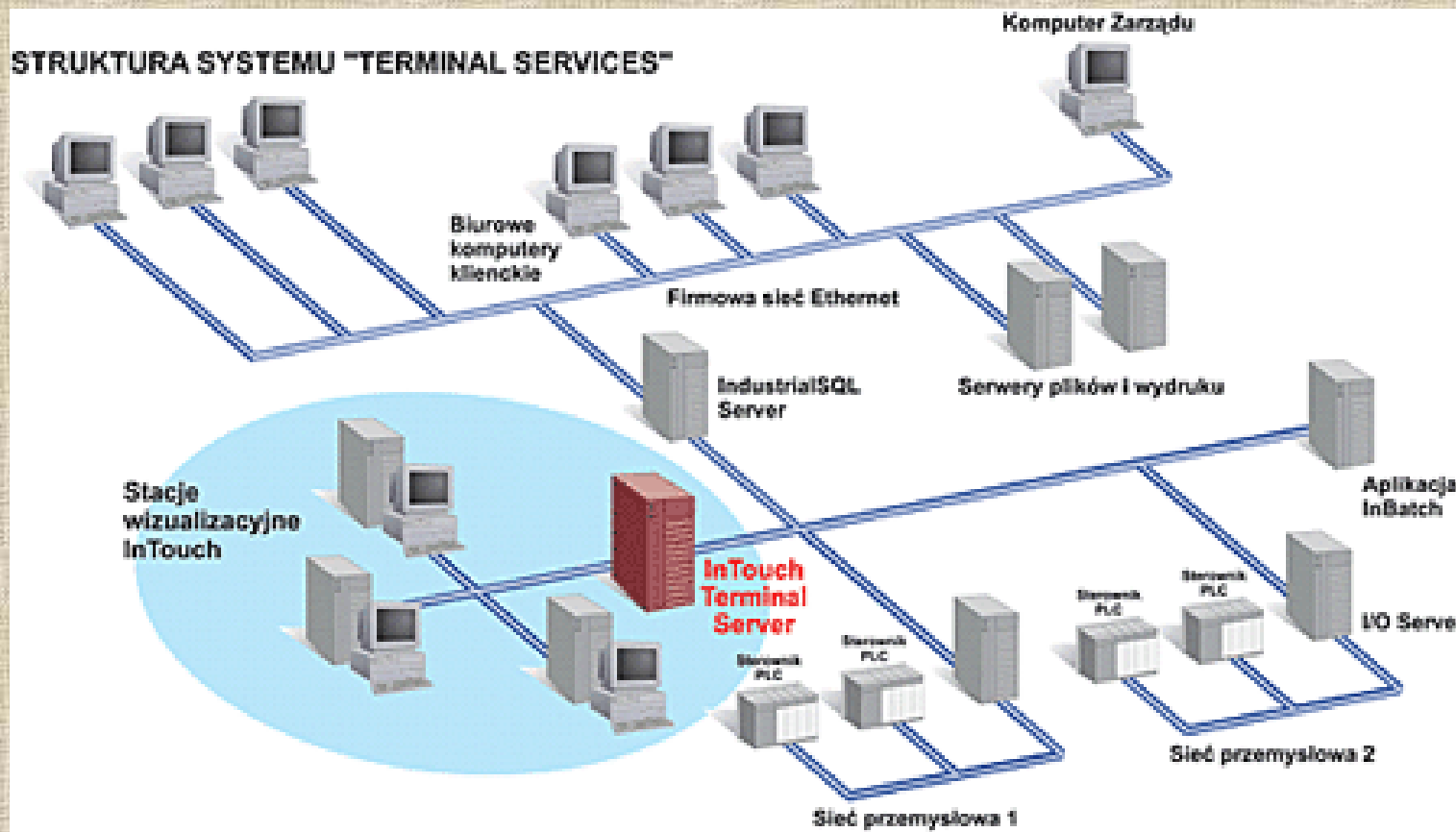
CONVEYOR

Integracja InToucha z platformą ArchestrA, opartą na technologii .NET firmy Microsoft i z **Wonderware Application Server**, umożliwia tworzenie bardzo rozbudowanych aplikacji wizualizacyjnych pracujących w systemie rozproszonym bez konieczności duplikowania logiki działania systemu na wszystkich stacjach operatorskich.



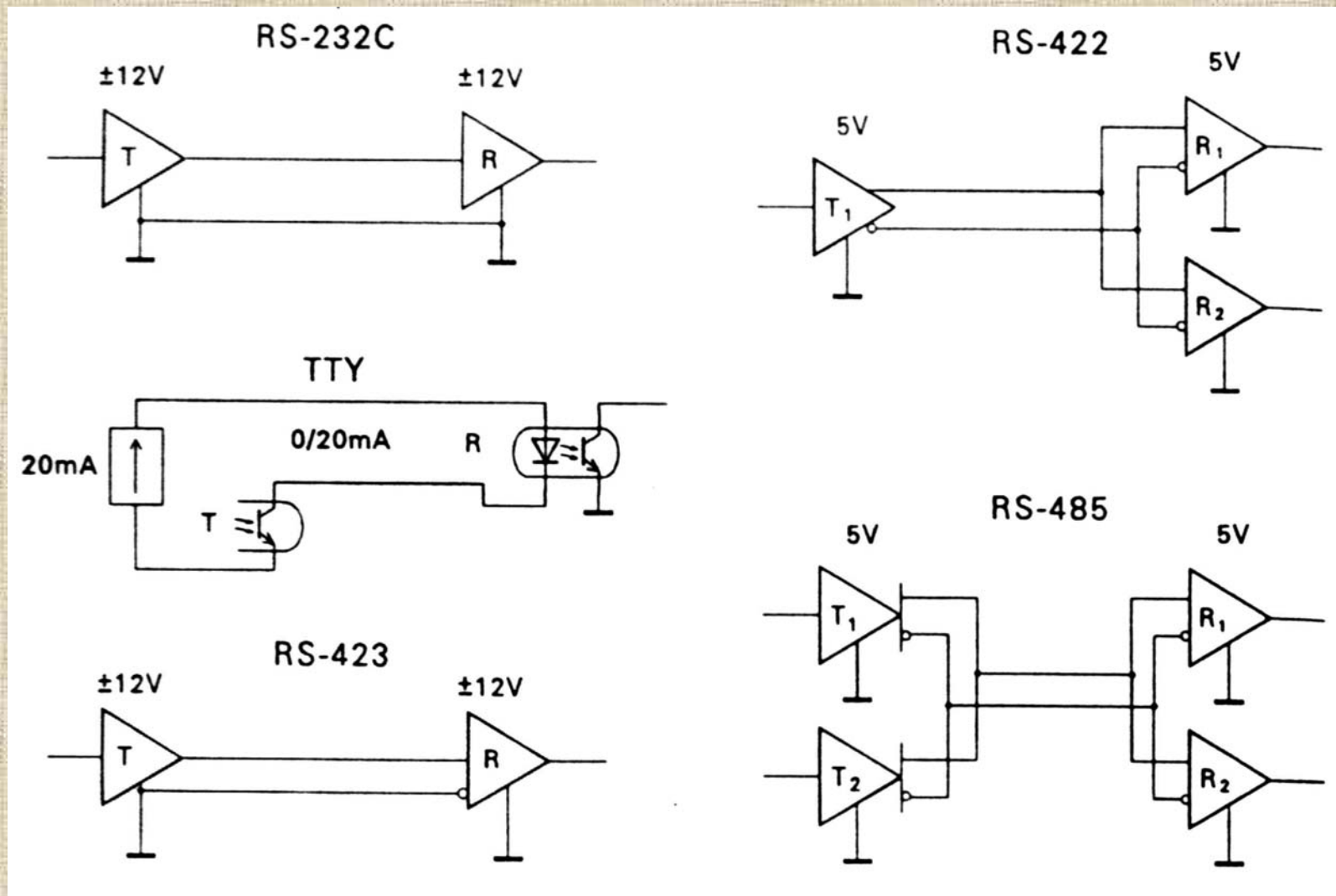
- Stanowisko pojedyncze - aplikacje InTouch'a mogą działać jako stacje pojedyncze. Każda stacja jest w pełni niezależna choć możliwe jest połączenie tych stacji.
- Klient/Serwer - aplikacje InTouch'a mogą także działać w środowisku klient/serwer oszczędzając czas i koszty obsługi oraz administracji systemami.
- Tag Server - jeden lub kilka komputerów działa jako „serwer zmiennych” posiadający listę wszystkich zmiennych używanych w aplikacjach InTouch'a, zarządzający ich logowaniem historycznym, uruchamianiem skryptów, alarmami oraz połączeniami z urządzeniami zewnętrznymi. Aplikacje uruchomione na stacjach klienckich łączą się z Tag Serverem i udostępniają dane operatorom.
- NAD (Network Application Development) - dynamiczna konfiguracja sieciowa NAD ułatwia centralne zarządzanie aplikacją InTouch'a przez jeden serwer sieciowy. Każda stacja kliencka tworzy lokalną kopię aplikacji serwera. Daje to możliwość tworzenia systemów redundantnych pracujących także bez włączonego serwera. Inną ważną cechą NAD jest uaktualnianie aplikacji klienckich "w locie" bez ich restartowania.

Usługi Terminalowe (*Terminal Services*) to architektura pozwalająca na centralne wdrażanie, obsługę i zarządzanie oprogramowaniem, ponowne użycie starszego sprzętu PC, wysoki stopień bezpieczeństwa oraz użycie wielu systemów operacyjnych. Dodatkowo istnieje możliwość użycia technologii Thin-client dającej wgląd w system także za pomocą urządzeń przenośnych udostępniając lokalnie wgląd do danych z czujników temperatury, rejestratorów i innych urządzeń.



Standardy komunikacyjne sterowników przemysłowych

Standardy interfejsów komunikacji szeregowej



Rys. Połączenie nadajnika T (transmitter) i odbiornika R (receiver) w różnych standardach komunikacji szeregowej (RS - Recommended Standard)

RS-232C (1962, wersja C - 1969)

Interfejs komunikacji *point-to-point*: jedno urządzenie to terminal DTE (*Data Terminal Equipment*), drugie to „modem” DCE (*Data Communication Equipment*). Długość kabla do 15m, szybkość transmisji do 19.2 kb/s.

Obejmuje: linie danych (kanału podstawowego TxD, RxD i powrotnego STxD, SRxD), linie sterujące (gotowości do pracy DSR, DTR i gotowości do transmisji RTS, CTS), linie synchronizacji (DA, DB, DD – sygnały zegarowe transmisji synchronicznej), linie masy (ochronnej PG, sygnałowej SG).

Parametry elektryczne: linie danych – logika ujemna („1” = $-15 \leq U_L \leq -3V$, „0” = $+3 \leq U_L \leq +15V$), linie sterujące – logika dodatnia; różnica poziomów zapewnia odporność na zakłócenia; wariant europejski V24.

Transmisja: • simpleks (jednokierunkowa), • półdupleks (dwukierunkowa niejednoczesna po jednej linii), • duplex (dwukierunkowa jednoczesna po dwóch liniach).

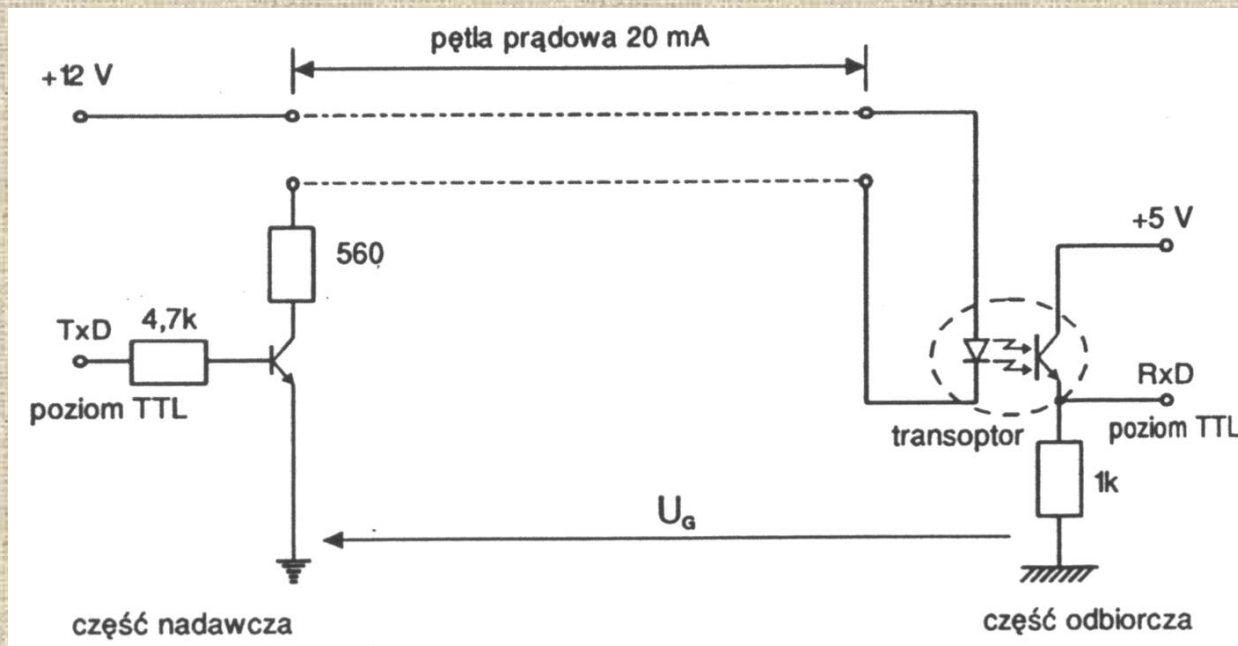
Transmisja synchroniczna (w określonych przedziałach czasu, synchronicznie z taktami zegarowymi) jest stosowana rzadko. Stosuje się zwykle transmisję asynchroniczną (format znaku: bit startu-bity danych-bit kontrolny (parzystości =1 dla nieparzystej liczby jedynek, opcjonalnie)-bit stopu). W przypadku silnych zakłóceń kontrolę błędów realizuje się za pomocą sumy kontrolnej (CRC lub LRC).

Pętla prądowa TTY

Ekspander interfejsu RS-232C z linią sterowaną prądem 0-20mA (zasilanie źródła prądowego 12V), zasilaną z linii DTR lub RTS. Zasięg 1500m przy szybkościach transmisji jak w RS-232C. Struktura sieci nie musi być ograniczona do gwiazdy.

Brak linii sterujących, są tylko dwie pary skręcanych przewodów do transmisji (TxD) i odbioru (RxD) danych. Format jak przy asynchronicznej transmisji znakowej RS-232C.

Parametry elektryczne: „1” = 13...33mA, „0” = -2...4.5mA, optoizolacja na napięciu 400V.



Rys. Obwód pętli prądowej z separacją galwaniczną.

RS-423A (1965)

Zmodyfikowany RS-232 z odbiornikiem z symetrycznym wejściem różnicowym (co eliminuje wpływ napięć pasożytniczych i przesłuchów między liniami) o czułości $\pm 200\text{mV}$ w zakresie napięcia wspólnego $\pm 6\text{V}$.

Zasięg 1200m przy szybkości transmisji 100 kb/s.

Do nadajnika można dołączyć 10 odbiorników o standardowej impedancji wejściowej.

RS-422A (1965)

Standard występujący w większości regulatorów i sterowników PLC.

Nadajnik z wyjściem różnicowym na poziomie -2 do +5V, minimalna rezystancja obciążenia 60 Ω , odbiornik z wejściem różnicowym o czułości $\pm 200\text{mV}$, zakres napięcia wejściowego odbiornika -7 do 12 V. Zasilanie nadajnika i odbiornika 5V, niepotrzebne układy dopasowujące.

Zasięg 1200m przy szybkości transmisji 100 kb/s, przy długości kabla równej 30m szybkość 10 Mb/s.

Do nadajnika można dołączyć 10 odbiorników o standardowej impedancji wejściowej (4 k Ω). Ze zmodyfikowanymi sterownikami można tworzyć sieć w formie 4-przewodowej magistrali wielodostępnej (*multi-drop bus*).

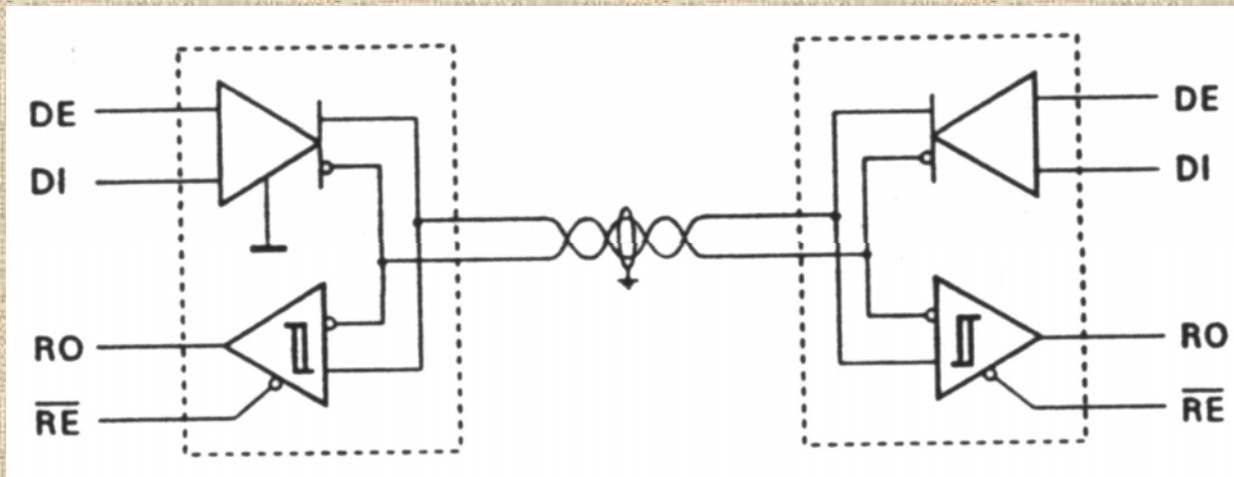
RS-485 (1983)

Powszechny obecnie standard w regulatorach i sterownikach przemysłowych. Urządzenia standardu RS-485 można łączyć z urządzeniami RS-422.

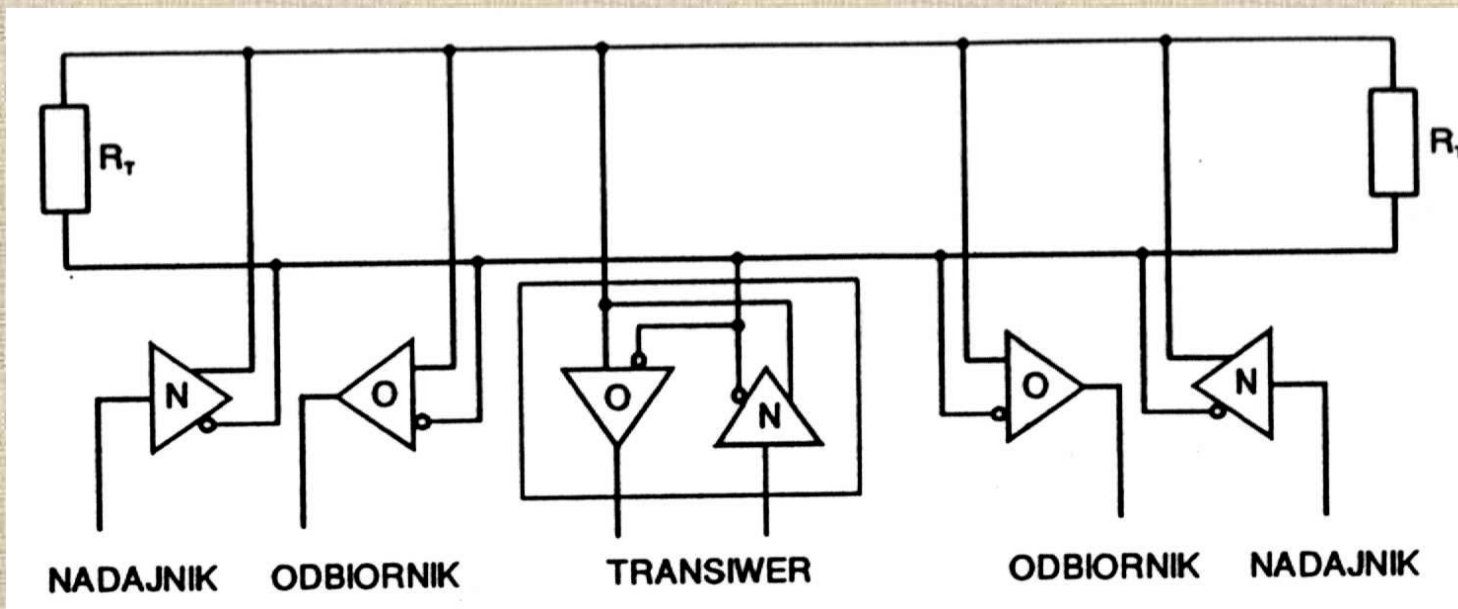
Poziomy sygnałów, napięcia zasilania, zasięg i szybkości transmisji jak dla RS-422.

W porównaniu z RS-422 zastrzone parametry elektryczne, m.in. zwiększona impedancja wejściowa odbiornika ($12\text{ k}\Omega$), co pozwala na połączenie 32 nadajników i odbiorników (więcej ze wzmacniaczmi-*repeaterami*). Nieaktywne nadajniki przechodzą w stan odcięcia (wysokiej impedancji), kolizje w przypadku jednoczesnego włączenia nadajników nie powodują uszkodzeń.

Linia jest zwykle 2-przewodową magistralą wielodostępną (półdupleks).



Rys. Połączenie sterowników linii RS-485



Rys. Magistrala interfejsu RS-485 z terminatorami $R_T (\geq 60 \Omega)$

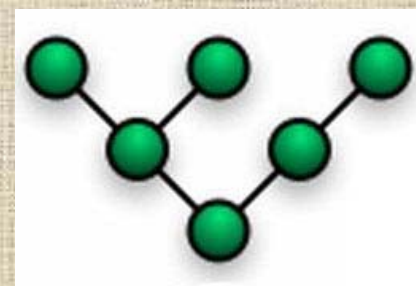
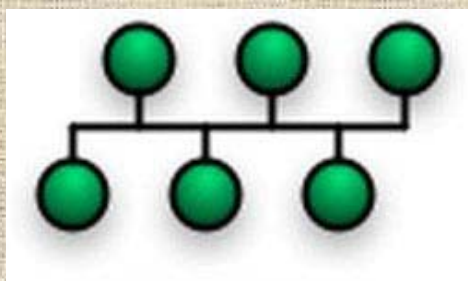
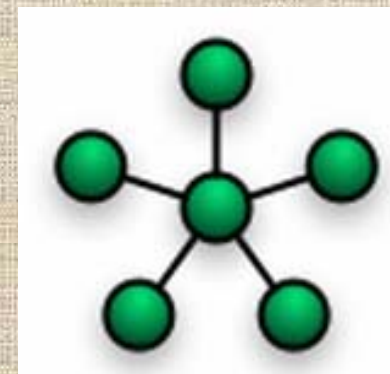
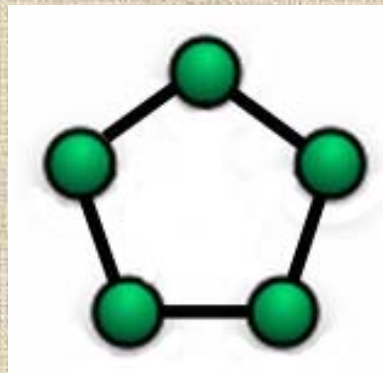
Komputerowe układy sterowania

Parametr		RS-232C	RS-423A	RS-422A	RS-485
Rodzaj transmisji		Niesymetryczna	Niesymetryczna	Różnicowa	Różnicowa
Dozwolona ilość nadajników i odbiorników		1 Nadajnik 1 Odbiornik	1 Nadajnik 10 Odbiorników	1 Nadajnik 10 Odbiorników	32 Nadajniki 32 Odbiorniki
Maksymalna długość kabla [m]		15	1200	1200	1200
Maksymalna szybkość transmisji (bity/s)		20 k	100 k	10 M	10 M
Maksymalne napięcie wspólne		± 25 V	± 6 V	+ 6 V - 0.25 V	+ 12 V - 7 V
Wyjście nadajnika		± 5 V min ± 15 V max	± 3,6 V ± 6.0 V	± 2 V min	± 1.5 V min
Obciążenie nadajnika		3 kΩ do 7 kΩ	450 Ω min	100 Ω min	60 Ω min
Szybkość zmian napięcia na wyjściu nadajnika		30 V/μs max	sterowana zewnątrz		
Ograniczenie prądu zwarcioowego nadajnika		500 mA przy zwarciu do V _{cc} lub GND	150 mA przy zwarciu do GND	150 mA przy zwarciu do GND	150 mA przy zwarciu do GND 250 mA przy zwarciu do - 8V lub 12V
Rezystancja wyjściowa nadajnika – wysoka impedancja	Zasilanie załączone				120 kΩ
	Zasilanie wyłączone	300 Ω	60 kΩ	60 kΩ	120 kΩ
Rezystancja wejściowa odbiornika		3 kΩ do 7 kΩ	4 kΩ	4 kΩ	12 kΩ
Czułość odbiornika		± 3 V	± 200 mV	± 200 mV	± 200 mV

Rys. Porównanie wymagań stawianych obwodom transmisyjnym standardowych interfejsów szeregowych

Typ	Ilość układów w module	Opis	Zastosowanie
SN55/75116	2	Wyjście 3-stanowe, 40 mA, przeciwne	Ogólne przeznaczenie
SN55/75117	2	Wyjście 3-stanowe, 40 mA, przeciwne	Ogólne przeznaczenie
SN55/75118	2	'116 z trójstanowym wyjściem odbiornika	Ogólne przeznaczenie
SN55/75119	2	'117 z trójstanowym wyjściem odbiornika	Ogólne przeznaczenie
SN75176B	2	Wyjście trójstanowe, 60 mA, odbiornik z histerezą	RS-485
SN75177B	2	'176 skonfigurowany jako wzmacniacz magistrali (repeater)	RS-485
SN75178B	2	'177 z odwróconym wejściem wyzwajającym	RS-485
SN75179B	2	dupleksowa wersja '176	RS-485

Rys. Przykłady scalonych różnicowych układów nadawczo-odbiorczych



Rys. Topologie połączeń w sieciach komunikacyjnych:
linia, pierścień, gwiazda, magistrala, drzewo

Sieciowe protokoły komunikacyjne

Protokoły komunikacyjne definiują sposób, w jaki dwie lub więcej stacji wymieniają dane wykorzystując odpowiednie ramki komunikacyjne. Cechą sieci polowych (*fieldbus*) jest optymalna transmisja małej ilości danych, krytycznych czasowo oraz maksymalne uproszczenie procesu transmisji. Sterowanie procesami wymaga reakcji (odpowiedzi) w określonym *nieprzekraczalnym* czasie.

Kontrola dostępu do sieci (*Medium Access Control*) jest specjalną procedurą, która określa w jakim momencie stacja może wysyłać dane. Stacje *aktywne* mogą same rozpoczynać wymianę danych, a stacje *pasywne* mogą rozpocząć komunikację tylko jeżeli dostaną zapytanie od stacji aktywnej.

Rozróżnia się procedury dostępu *deterministyczne* z pracą w czasie rzeczywistym (np. *master-slave* w sieci PROFIBUS) oraz losowe, *niedeterministyczne* (np. CSMA/CD – *Carrier Sense Multiple Access with Collision Detection* - w sieci Ethernet).

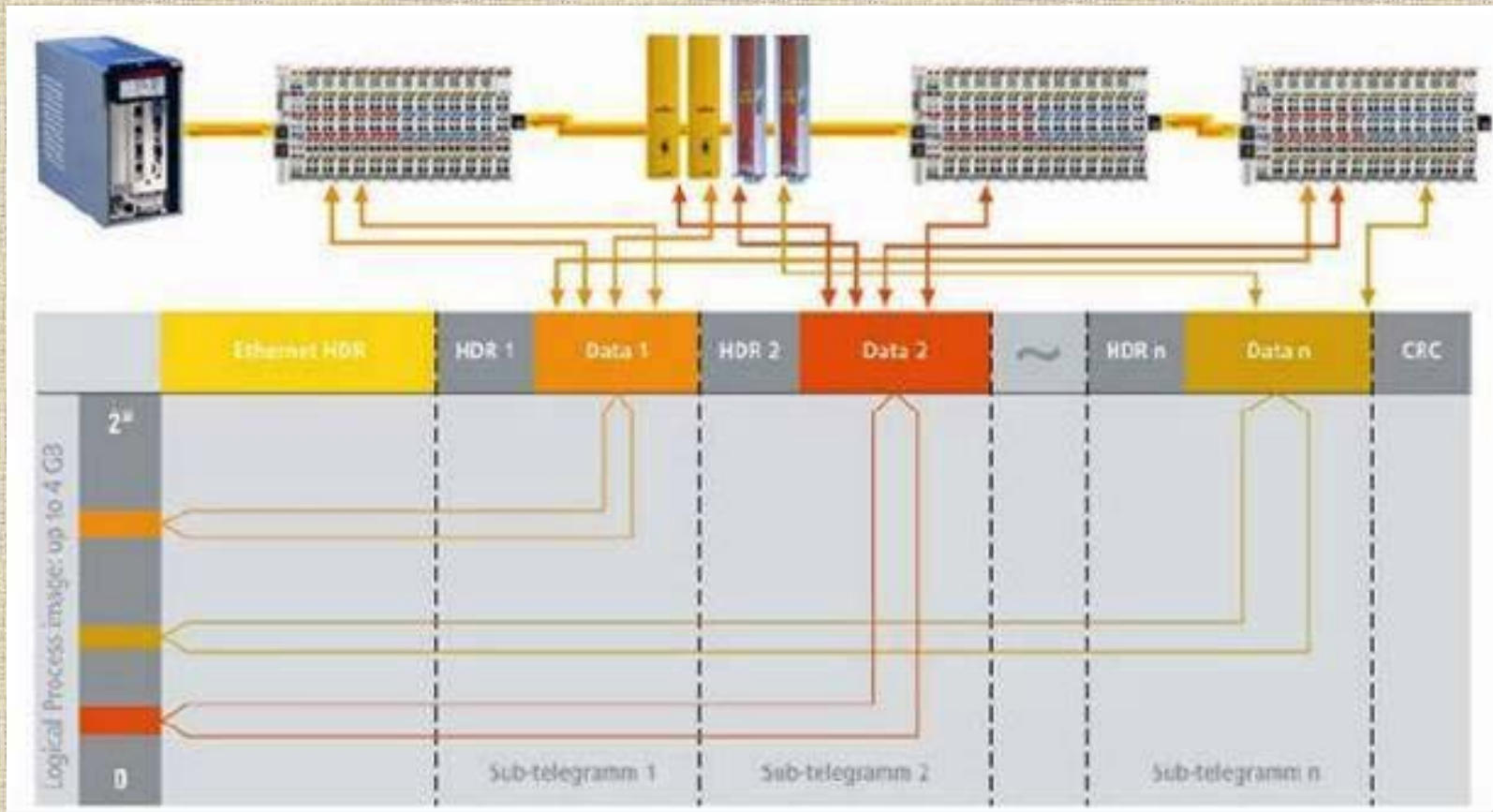
W przypadku CSMA/CD stacja nasłuchuje łącze przed transmisją i ewentualnie czeka na zwolnienie linii. W przypadku kolizji stacje odczekują krótki, losowo określony dla każdej, odcinek czasu i ponawiają próbę. Czas dostępu do sieci zależy od jej obciążenia. Jest to dobry protokół dla sieci lokalnych np. do zastosowań biurowych.

Przemysłowe protokoły sieciowe

Dla transmisji szeregowej:

- **MODBUS**
- **PROFIBUS**
- **CAN**
- **LonWorks** – opracowana przez firmę Echelon, 1990; sieć *peer-to-peer*, niehierarchiczna z inteligentnymi węzłami z zaimplementowanym protokołem i fizyczną warstwą sprzęgającą z medium: skrętka, instalacja elektryczna 50-60Hz/110-230V, fale radiowe; stosowany w instalacjach budynków: systemy przeciwpożarowe, alarmowe, oświetleniowe, grzewcze, windowe, bramowe, sterowania i integracji itp.), max. 32 385 węzłów w domenie, transmisja do 1.25Mb/s, długość linii do 2km)
- **EIB** - *European Installation Bus*, europejski standard sieci instalacji budynków)
- **WorldFIP** - otwarta (3 warstwy OSI jak Profibus), magistralowa (*multi-point*) sieć *fieldbus* do automatyki przemysłowej w niższych warstwach sterowania, przebieg transmisji informacji w postaci tzw. zmiennych pomiędzy rozproszonymi abonentami nadzoruje jednostka sterująca - *bus arbitrator*),

- **EtherCAT** – *Ethernet Control Automation Technology*, w przeciwieństwie do Ethernet nie ma konieczności przesyłu pakietów do każdego urządzenia osobno (min. ramka to 84 bajty), technologia oferuje tworzenie komunikatów dla wielu węzłów (*multicast*): odbiorniki czytają adresowane do nich dane w momencie gdy ramka przelatuje przez węzeł w drodze do następnego. Wtedy dodawana jest też odpowiedź (opóźnienie odbioru wynosi kilka nanosekund).



Rys. Ułożenie danych dla wielu urządzeń w jednym komunikacie EtherCAT

Ponieważ ramka niesie dane wielu urządzeń naraz w kierunku nadawania jak i odpowiedzi (full-duplex), rozmiar użytecznych danych wzrasta do 90%; na łączu 100BaseTX możliwe jest osiągnięcie efektywnego przesyłu danych powyżej 100 Mb/s, co umożliwia zlikwidowanie „wąskiego gardła” sieci *fieldbus*. EtherCAT jest zoptymalizowany do transmisji bezpośrednio w ramce Ethernetu dzięki specjalnemu typowi telegramu *Ethertype*. Może on zawierać kilka podtelegramów, każdy obsługujący daną część procesu. Całość przekazu może nieść do 4 GB danych. Możliwa jest komunikacja *broadcast* i *multicast* oraz komunikacja między urządzeniami *slave*.

Nagłówek Ethernetu			ECAT	Telegram EtherCAT			Enet
Adres Nadajnika	Adres Odbiornika	Typ	Nagłówek Ramki	Nagłówek EtherCAT	Dane	Suma kontrolna Telegramu	Suma kontrolna Ramki
(6)	(6)	(2)	(2)	(10)	32....1486	(2)	(4)

Nagłówek Enet			ECAT	Telegram EtherCAT			Telegram EtherCAT			Enet
AN	AO	TYP	Nagłówek Ramki	Nagłówek EtherCAT	Dane	Suma kontrolna Telegramu	Nagłówek EtherCAT	Dane	Suma kontrolna Telegramu	Suma kontrolna Ramki
(6)	(6)	(2)	(2)	(10)	32....1486	(2)	(10)	32....1486	(2)	(4)

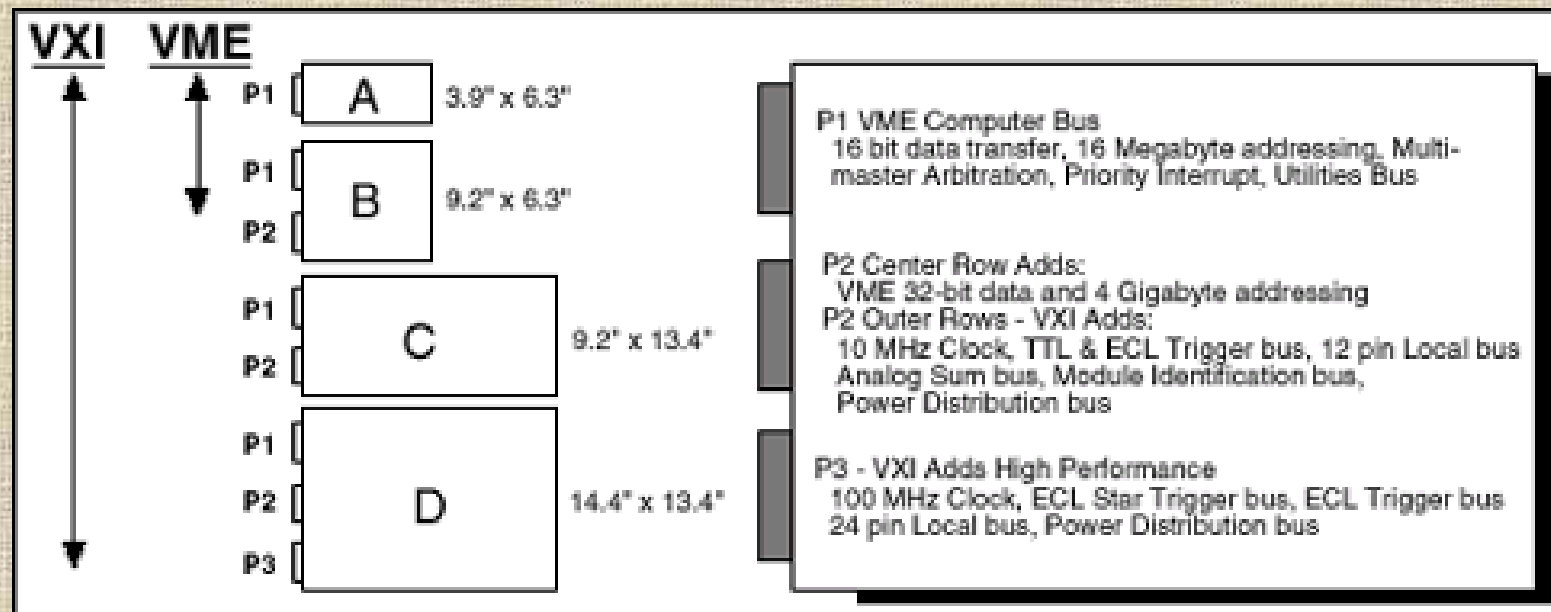
Rys. Ramki EtherCAT dla pojedynczego procesu i wielu procesów. Kolejność niesionych informacji i adresacja jest niezależna od fizycznego rozmieszczenia urządzeń.

Dla transmisji równoległej:

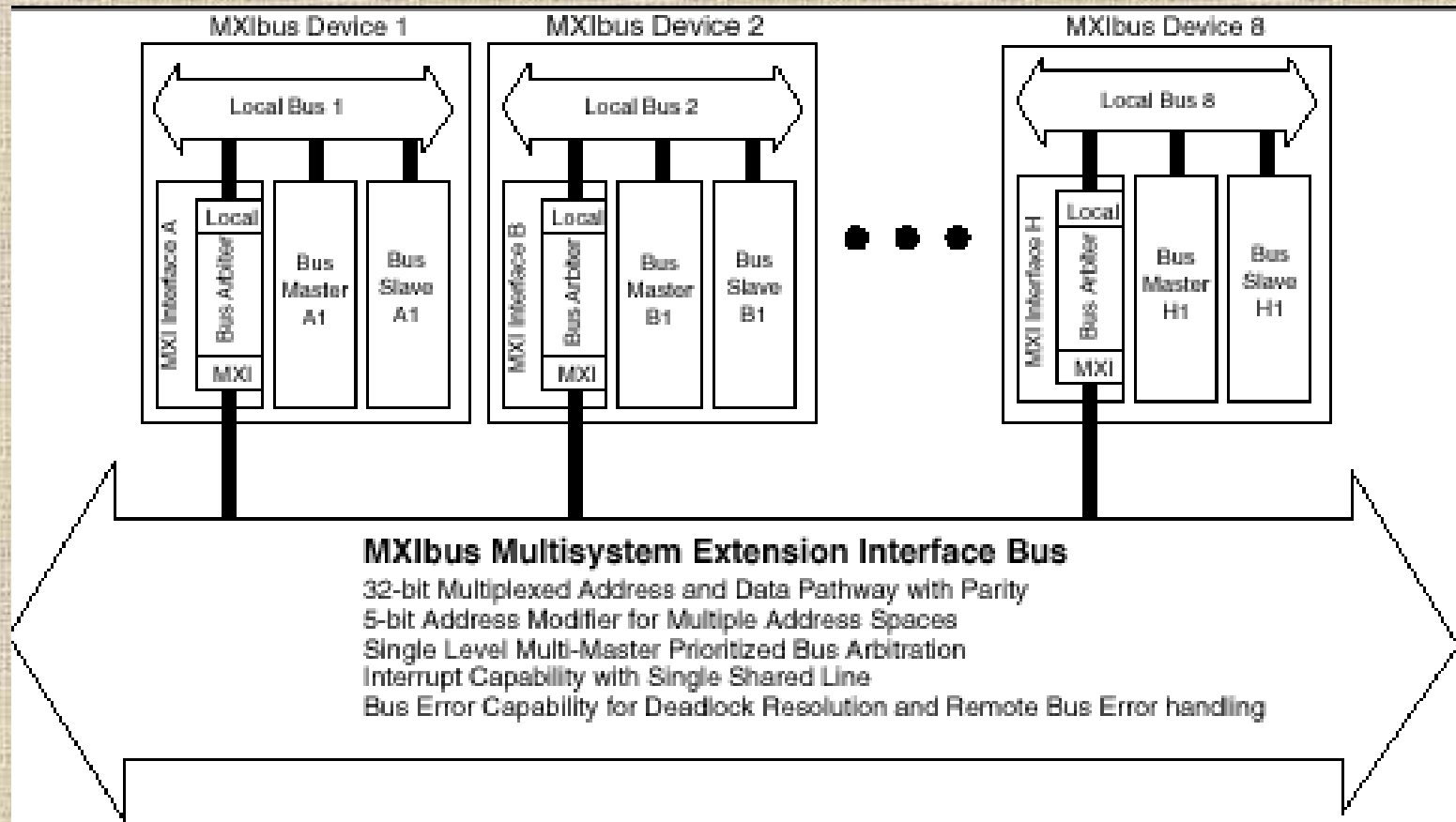
- **GPIB** – *General Purpose Interface Bus*, odmiana europejskiego **IEC 625** i amerykańskiego **IEEE-488** opracowanych w latach 70-tych. Do sprzęgania elementów aparatury kontrolno-pomiarowej i sterującej. Elastyczny i łatwy w programowaniu. Komunikacja za pomocą komunikatów (adresów, poleceń, danych) lokalnych, wewnątrz urządzenia, oraz zdalnych (magistralowych) przesyłanych pomiędzy urządzeniami. Szybkość transmisji 1 MB/s. liczba urządzeń do 15 (28 z ekspanderem), zasięg do 20m. Urządzenia są dołączone równoległe do magistrali składającej się z 16 linii sygnałowych (8 linii danych, 3 linie synchronizacji, 5 linii sterowania) i 9 linii masy.

▪ **VXI (MXI) - VMEbus eXtension for Instrumentation**, 1987 (norma IEEE 1155, 1993). Łączy zalety IEC-625 z szybkością pełnej 32-bitowej architektury VME opracowanej w 1982 przez Motorolę, zapewnia lepszą synchronizację czasową. Konstrukcja oparta na płycie *backplane*, z kartami-instrumentami VXI, które mogą ze sobą współpracować. *Backplane* jest montowana w kasecie z zasilaczem i chłodzeniem. Można zainstalować do 256 urządzeń, przy sterowaniu z komputera zewnętrznego szybkość transmisji do 1MB/s, z kartą procesora w kasecie – do 40MB/s.

W skład VXI wchodzi 3 magistrale: - ogólnego przeznaczenia dostępna dla każdego urządzenia (4 szyny), - specjalna (3 szyny), - wydzielona, do komunikacji pomiędzy sąsiednimi modułami. Protokół komunikacyjny typu *Commander/Servant* (możliwość logicznej hierarchicznej struktury drzewa).

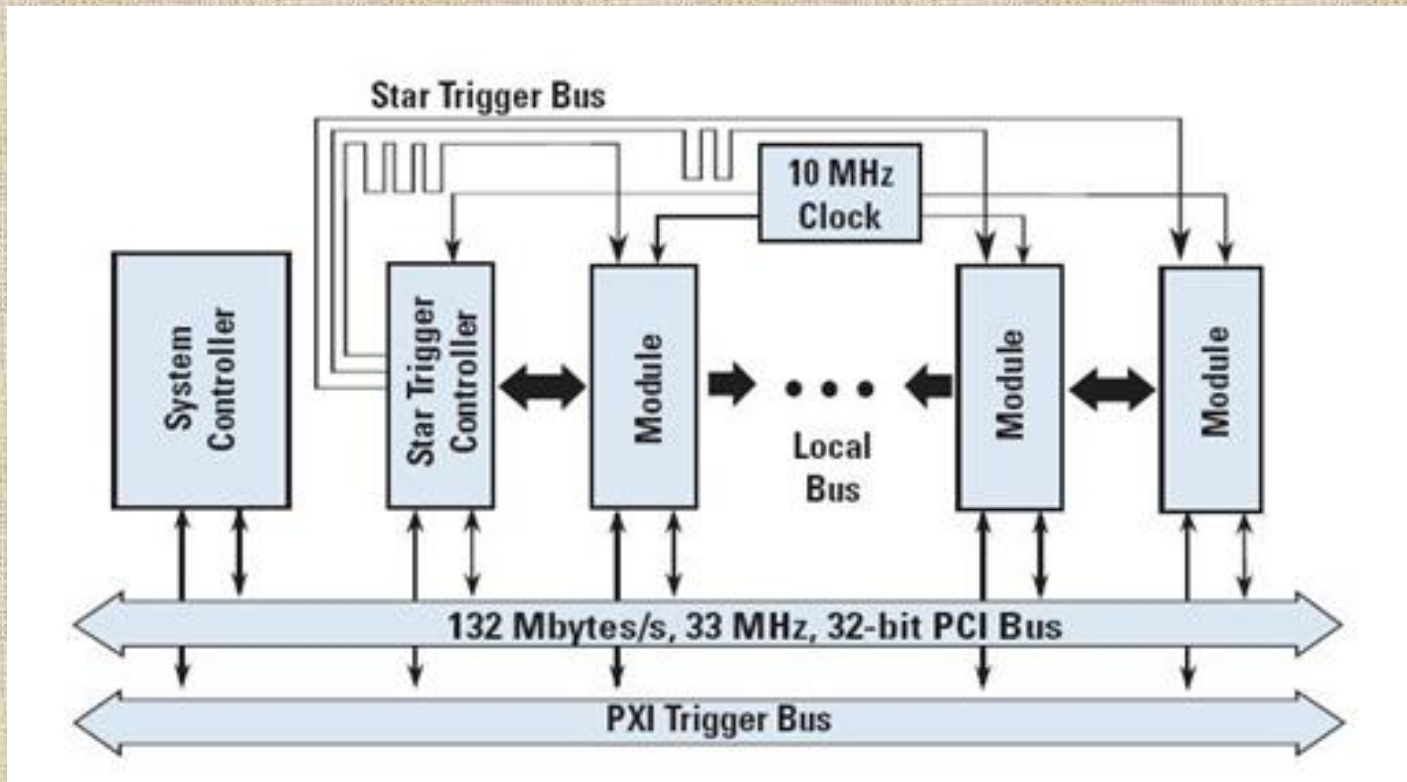


Kasety z instrumentami VXI mogą być kontrolowane ze stacji roboczych PC za pomocą interfejsów MXI (*Multisystem eXtension Interface*) lub GPIB-VXI (do ok. 20m, elastyczne okablowanie, ciągły przepływ do 20 Mb/s, dane 8,16,32 bity).



Rys. Interfejs MXI – komunikacja z mapowaniem pamięci – odwzorowaniem ma poziomie sprzętowym oddzielnych systemów fizycznych do wspólnego obszaru pamięci, gdzie następują zapisy i odczyty.

- **PXI** – *PCI eXtension for Instrumentation*, National Instruments 1998 – otwarty standard przemysłowy. System składa się z *backplane* PCI z modułami w formie Eurocard. Zawiera układ precyzyjnej synchronizacji magistrali i odporną instalację mechaniczną. 2005 - PCI Express. (PXI-CAN i LabVIEW są wykorzystywane m.in. do monitorowania paneli awioniki Boeinga 787).



Rys. Szyny wyzwalania i synchronizacji systemu PXI (*Timing and Triggering Buses*) na backplane. Nowszy system *PXI Express* zawiera 100MHz *Differential CLK* i jest bardziej odporny na zakłócenia.

Normy dotyczące przemysłowych protokołów sieciowych

IEC 61158 (1999): „Cyfrowa komunikacja danych w sieciach polowych, pomiarowych i sterujących, do zastosowania w przemysłowych systemach sterujących” (*Digital Data Communication for Measurement and Control – Fieldbus for Use in Industrial Control Systems*). Definiuje 10 typów protokołów sieciowych (np. PROFIBUS – Typ 3, PROFINet – Typ 10).

IEC 61784: „Zbiór profili dla systemów ciągłych i dyskretnych bazujących na sieciach polowych wykorzystywanych w przemysłowych systemach sterowania” (*Profile Sets for Continuous and Discrete Manufacturing Relative to Fieldbus Use in Industrial Control Systems*). Określa zbiór profili danego protokołu komunikacyjnego opartego na normie IEC 61158.

MODBUS

Protokół software'owy (półdupleks, hardware nie zdefiniowany) *master-slave* opracowany przez AEG Schneider Automation dla sterowników Modicon. Urządzenie *master* może zarządzać 247 urządzeniami *slave*.

Typowym urządzeniem *master* jest komputer PC z panelem operatorskim lub nadrzędny sterownik (*host*). Urządzenia *slave* są wykorzystywane do sterowania (sterowniki PLC, regulatory) oraz zbierania danych (urządzenia peryferyjne: mierniki, czujniki, falowniki, przekaźniki). Głównym obszarem zastosowania protokołu są małe i średnie sieci z jedną jednostką główną i oddalonymi urządzeniami peryferyjnymi, np. w sterowaniu obiektami gdzie nie jest możliwe zainstalowanie połączenia przewodowego - wykorzystuje się wtedy transmisję bezprzewodową z radiomodemami. W odmianie Modbus Multi-Master może być więcej jednostek *master*, które przekazują sobie prawo do nadawania w sieci.

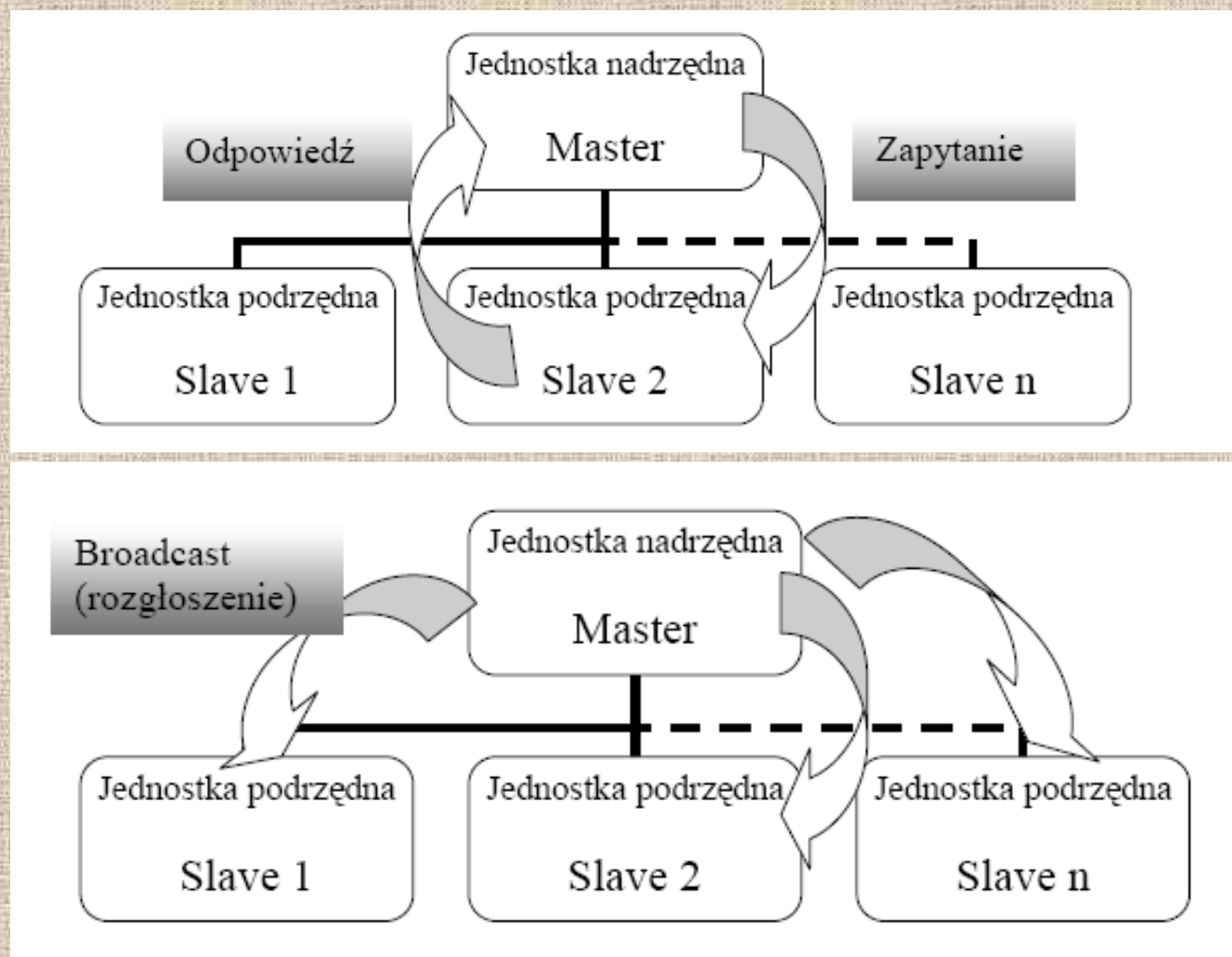
MODBUS RTU (*Remote Terminal Unit*) – działa na zasadzie wymiany pakietów (adres odbiornika, polecenia do wykonania, dane) + kontrola CRC (*Cyclical Redundancy Check*).

MODBUS ASCII – transmisja znakowa, heksadecymalne znaki ASCII 0-9, A-F (7 bitów) + kontrola LRC (*Longitudinal Redundancy Check*); możliwe odstępy między znakami do 1s bez powodowania błędów.

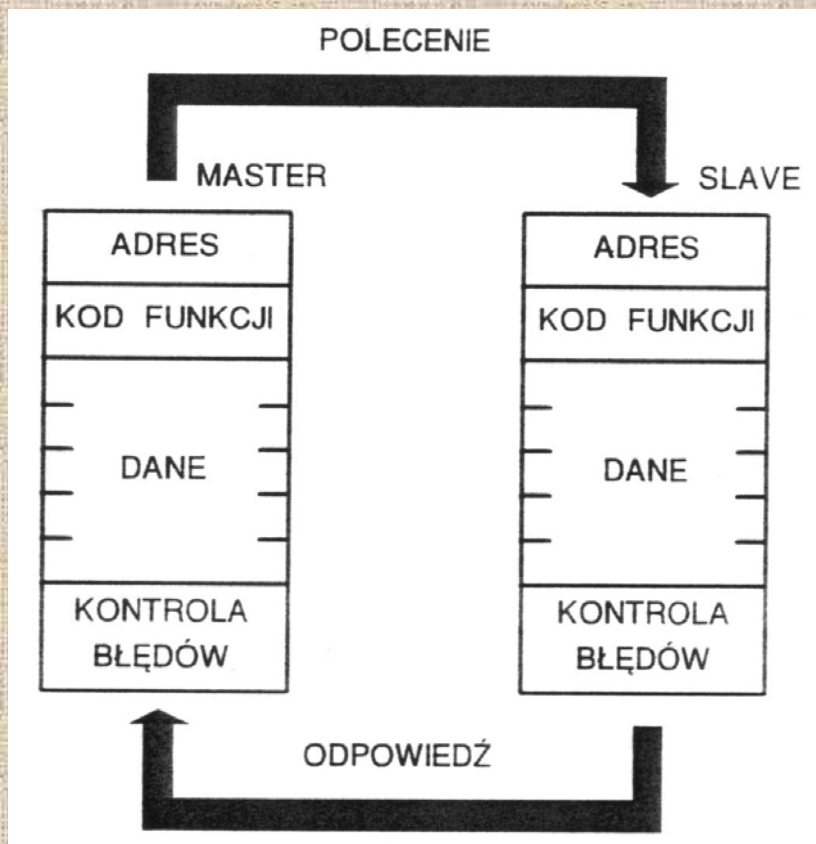
Transakcja składa się z polecenia (*query*) wysyłanego przez stację *master* do indywidualnej stacji *slave* (z adresem odbiorcy) lub wiadomości rozgłoszeniowych (*broadcast*) przeznaczonych do wszystkich stacji (jednostki *slave* nie odsyłają wtedy odpowiedzi).

Na zapytanie stacja *slave* odsyła odpowiedź (*response*) z żądanymi danymi lub potwierdzeniem realizacji polecenia (w polu funkcji zostaje umieszczony kod wykonanego rozkazu - potwierdzenie) albo komunikat o błędzie (*exception response*, niemożność wykonania polecenia lub błąd przy odbiorze, dane dotyczące występującego błędu dołączane są do pola danych ramki odpowiedzi).

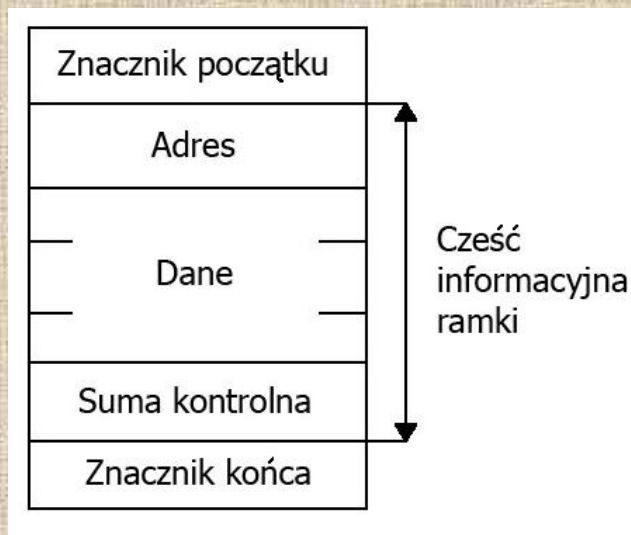
Użytkownik programuje w stacji *master* ograniczenie czasu oczekiwania na odpowiedź *slave'a*. Jeżeli w tym czasie odpowiedź nie nadejdzie, *master* uznaje że komunikacja ze *slave'm* została zerwana i generuje komunikat o błędzie.



Rys. Komunikacja *master-slave* typu *query-response* i typu *broadcast*



Rys. Transakcja w sieci MODBUS i ogólny format ramki



Rys. Schemat rozkazu i odpowiedzi przy wystąpieniu błędu.

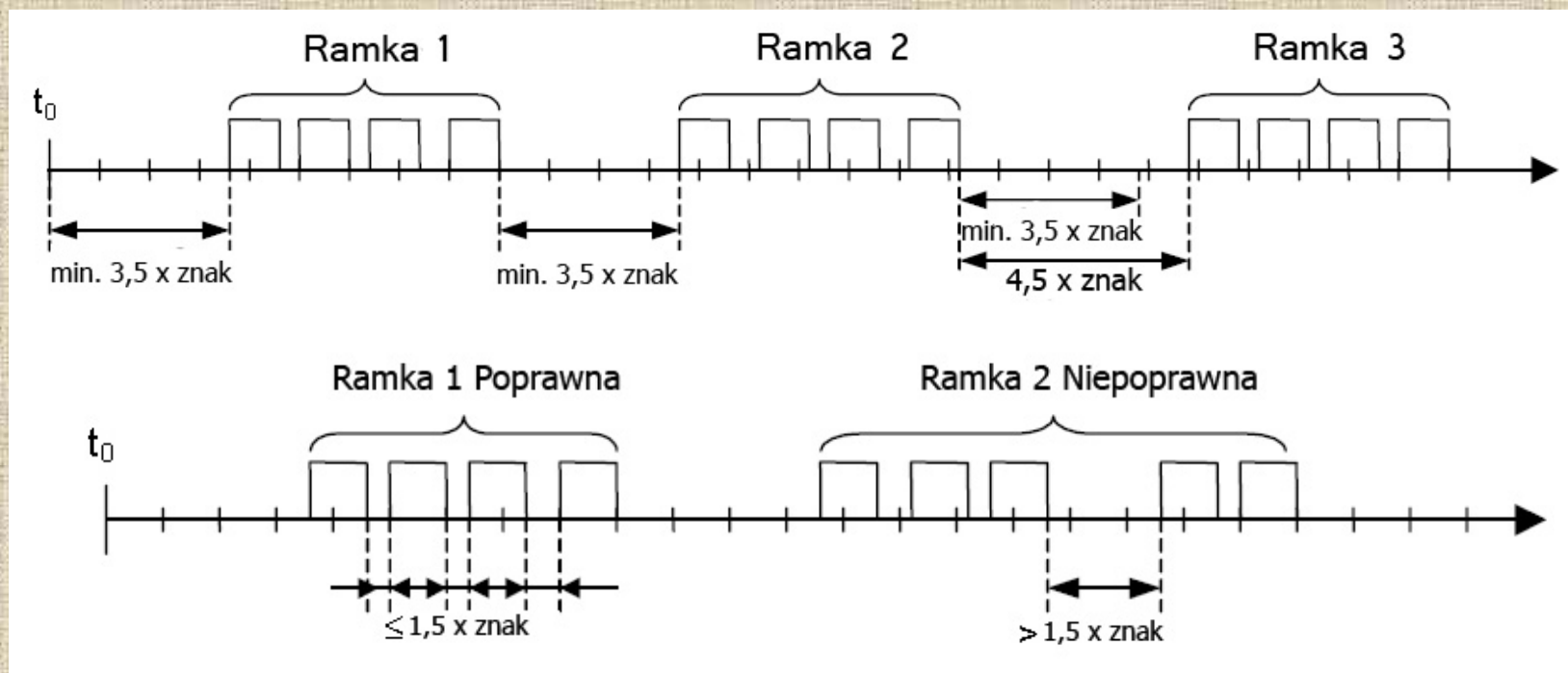


Znacznik początku	Adres	Funkcja	Dane	Kontrola CRC	Znacznik końca
T1-T2-T3-T4	1 bajt	1 bajt	0÷256 bajtów	2 bajty	T1-T2-T3-T4

START	1	2	3	4	5	6	7	8	PARITY	STOP
START	1	2	3	4	5	6	7	8	STOP	STOP

Rys. Format ramki i pojedynczego bajtu informacji w trybie Modbus RTU (T1-T2-T3-T4 – cztery takty o długości znaku każdy)

Wiadomości zaczynają się i kończą interwałem (min. 3.5 x czas trwania 1 znaku), w którym na łączu panuje cisza. Interwał końcowy poprzedniej ramki może być interwałem początkowym następnej. Cała ramka musi być przesłana w sposób ciągły, z odstępami pomiędzy znakami nie większymi niż (1.5 x czas trwania znaku). Zaletą trybu RTU jest większa przepustowość łącza w porównaniu z trybem ASCII.



Rys. Zasady przesyłania komunikatu w trybie RTU

Znacznik początku	Adres	Funkcja	Dane	Kontrola LRC	Znacznik końca
1 znak ':'	2 znaki	2 znaki	0÷256 x 2 znaki	2 znaki	2 znaki CR LF

START	1	2	3	4	5	6	7	PARITY	STOP
START	1	2	3	4	5	6	7	STOP	STOP

Rys. Format ramki i jednostki informacyjnej znaku w trybie Modbus ASCII

Każda jednostka informacji (bajt) jest przesyłana w formie 2 znaków ASCII Hex (dopuszczalne znaki to '0'-'9', 'A'-'F', znacznik początku ':'=3Ah). Zaletą trybu ASCII jest to, że pozwala na długie (do 1 s) odstępy między znakami bez powodowania błędów.

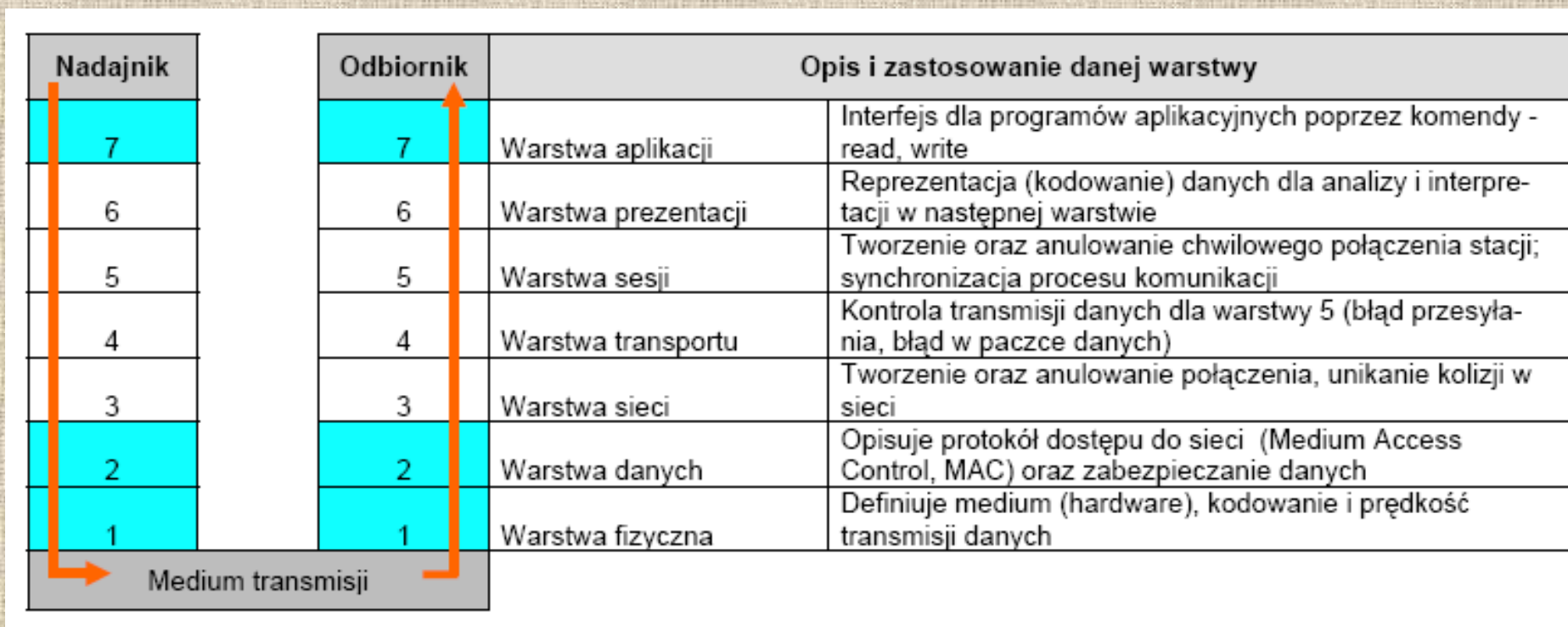
PROFIBUS (*PRO*cess *Fie*ld *BUS*)

Opracowany w 1991 r. przez Siemens, ustanowiony normą EN 50170. Elastyczny i uniwersalny.

- Topologia sieci: pierścień logiczny (*token ring*)
- Max. liczba węzłów w sieci z repeaterami: 126 (32 w segmencie)
- Max. długość linii – 1200 m (4800 m ze wzmacniaczami)
- Szybkość transmisji od 9.6 kb/s do 1.5 Mb/s (zależnie od długości linii)
- Transmisja asynchroniczna (półdupleks) zgodna z RS-485
- Medium transmisyjne: skrętka lub światłowód (również w obrębie jednej sieci).

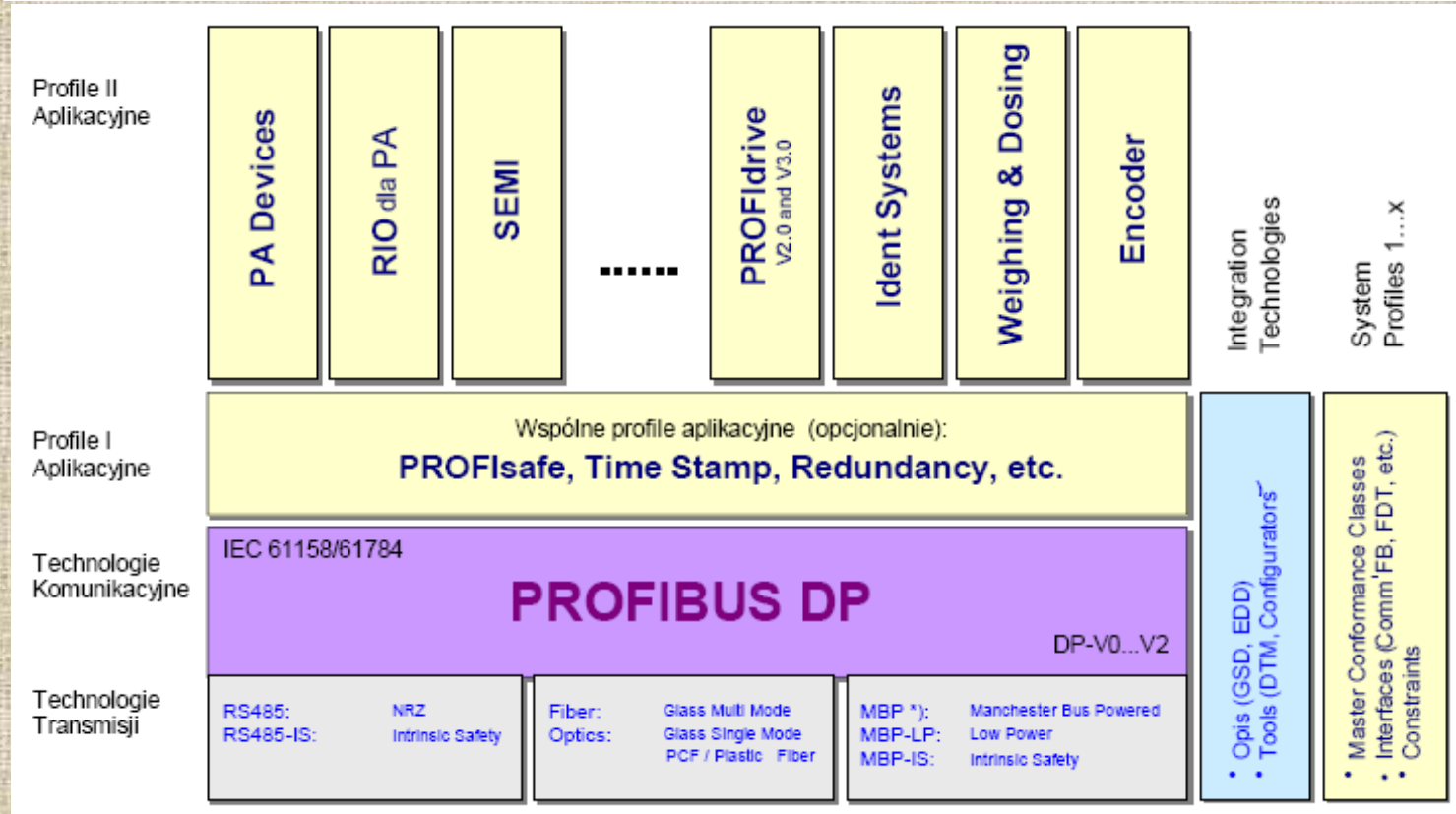
PROFIBUS jest liderem na rynku europejskim. W 2007r. liczba węzłów sieciowych stosowanych w sieciach PROFIBUS przekroczyła 20 milionów. Architektura PROFIBUS oparta jest na międzynarodowym modelu ISO/OSI. W warstwie aplikacji określa się strukturę sieci za pomocą uniwersalnego opisu urządzeń (typ wbudowanego interfejsu, adresy portów, relacje pomiędzy urządzeniami) zapisanego w pliku GDS.

Zestaw wybranych funkcji protokołu tworzy profil. W zależności od tego wyróżnia się kilka typów magistrali PROFIBUS.



Rys. Model referencyjny ISO/OSI (*Open System Interconnection*) z warstwami stosowanymi w standardzie PROFIBUS

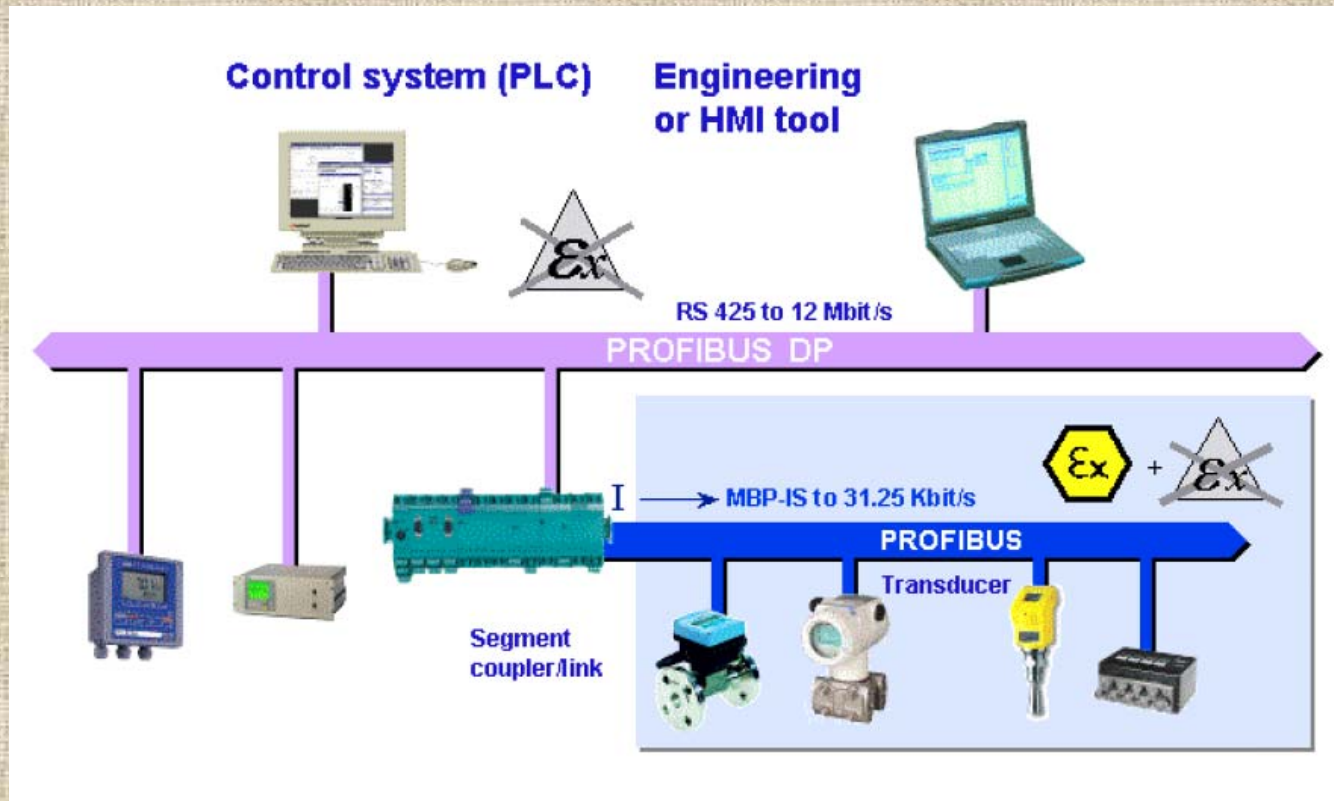
Komputerowe układy sterowania



Rys. Modułowa struktura protokołu PROFIBUS z podziałem na warstwy OSI.

PROFIBUS FMS (*Field Message Specification*) – najwyższy poziom do współpracy z PLC, stacjami PC i in. urządzeniami wyższego poziomu i przesyłu znacznych ilości danych; medium: przewód zgodny z RS-485 lub światłowód.

PROFIBUS PA (*Process Automation*, 1995) – do inteligentnych urządzeń automatyki procesowej przesyłających znaczne ilości danych; określa poziomy bezpiecznych napięć i prądów; medium jest zwykle magistrala Manchester MBP (stosowana też dla stacji iskrobezpiecznych; *Manchester coding*, *Bus Powered*, synchroniczna transmisja 31 Kb/s).



Rys. Struktura sieci przy zastosowaniu magistrali MBP (do 32 stacji w segmencie).

PROFIBUS DP (*Decentralized Periphery*, 1993) – zarządza 3 urządzeniami *master*, które inicjują komunikację z urządzeniami *slave* będącymi węzłami rozproszonej sieci. Prostszy i szybszy od FSM. Obecnie wersje DP-V0÷V2, medium jest przewód zgodny z RS-485 lub światłowód.

Motion Control with PROFIBUS – do sterowania układów napędowych za pomocą aplikacji PFOFIdrive i interfejsu RS-485.

PROFIBUS FMS

Protokół oparty na strukturze Klient-Serwer (*Token Passing* – z przekazywaniem *tokena*, tj. uprawnień do nadawania), dzięki czemu w jednej sieci może się komunikować ze sobą wiele stacji (*multi-master*). FMS umożliwia dostęp do obiektów programowalnych istniejących w innych węzłach sieci (zmiennych, tablic czy rekordów). Istnieje możliwość przekazania programów i sterowania wykonaniem programów między węzłami sieci.

Protokół FMS definiuje każdą usługę za pomocą czterech operacji:

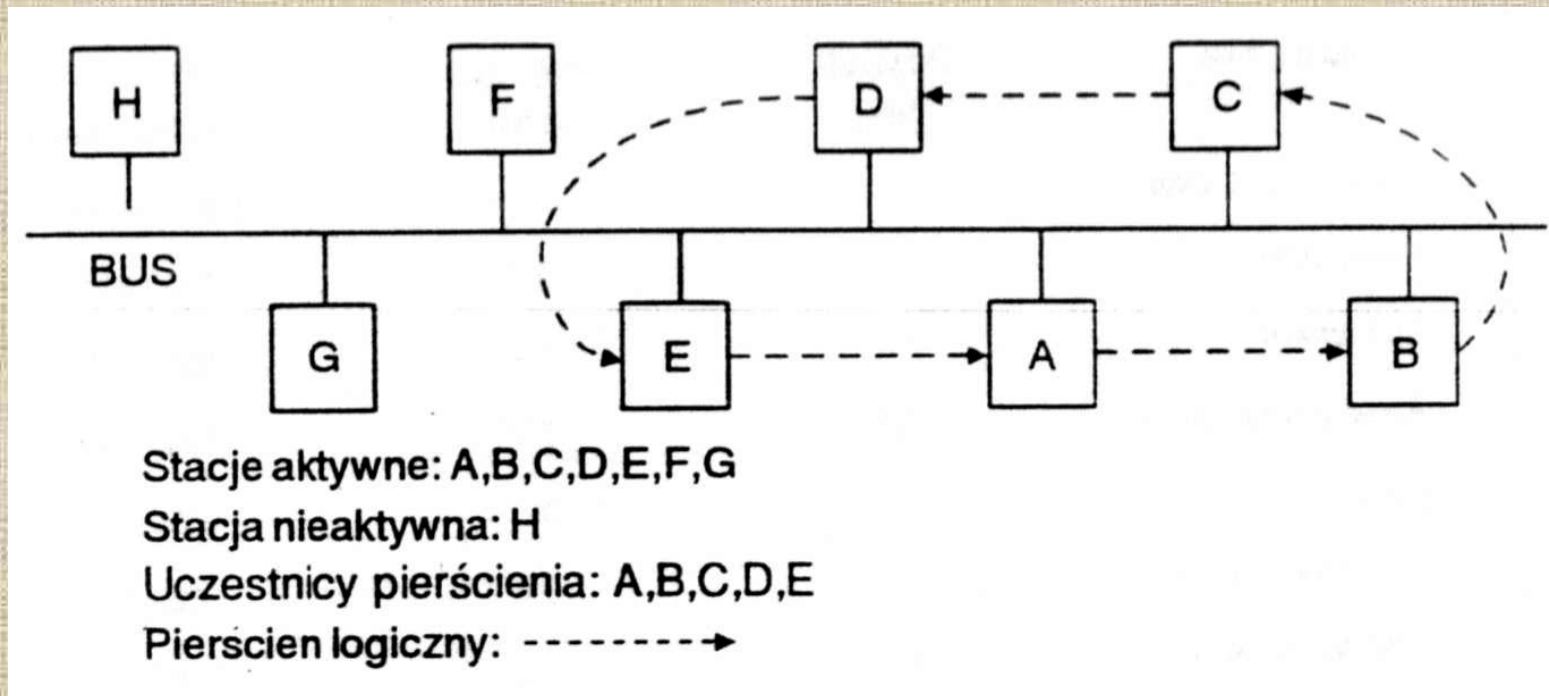
- Wywołanie usługi (*Request*)
- Sygnalizacja odebrania wywołania (*Indication*)
- Przekazanie potwierdzenia (*Response*)
- Sygnalizacja odebrania potwierdzenia (*Confirmation*)

Każda z tych operacji ma dokładnie określoną funkcję, argumenty oraz format.

Mechanizm *Token Passing* ma na celu zagwarantowanie nieprzekraczalnego czasu odpowiedzi stacji, co jest istotne przy sterowaniu. Pozwala efektywniej wykorzystać kanał, ponieważ nie dochodzi do kolizji w kablu. Wadą jest złożony algorytm protokołu (istnieją sprzętowe kontrolery TOKENBUS).

Prawo dostępu na określony czas określa przekazywana ramka *token frame* (żeton). Jeżeli nie ma ramek do transmisji lub wyczerpał się limit czasu, następuje przekazanie tokena do następnej stacji w pierścieniu, która przejmuje prawo do transmisji.

Stacje dołączone do magistrali, które nie wchodzą w skład pierścienia, mogą tylko odpowiadać na zdalne zapytania lub żądać potwierdzenia.



Rys. Topologia sieci typu *token ring* (PROFIBUS FMS). Topologia fizyczna – magistralna, topologia logiczna – pierścieniowa.

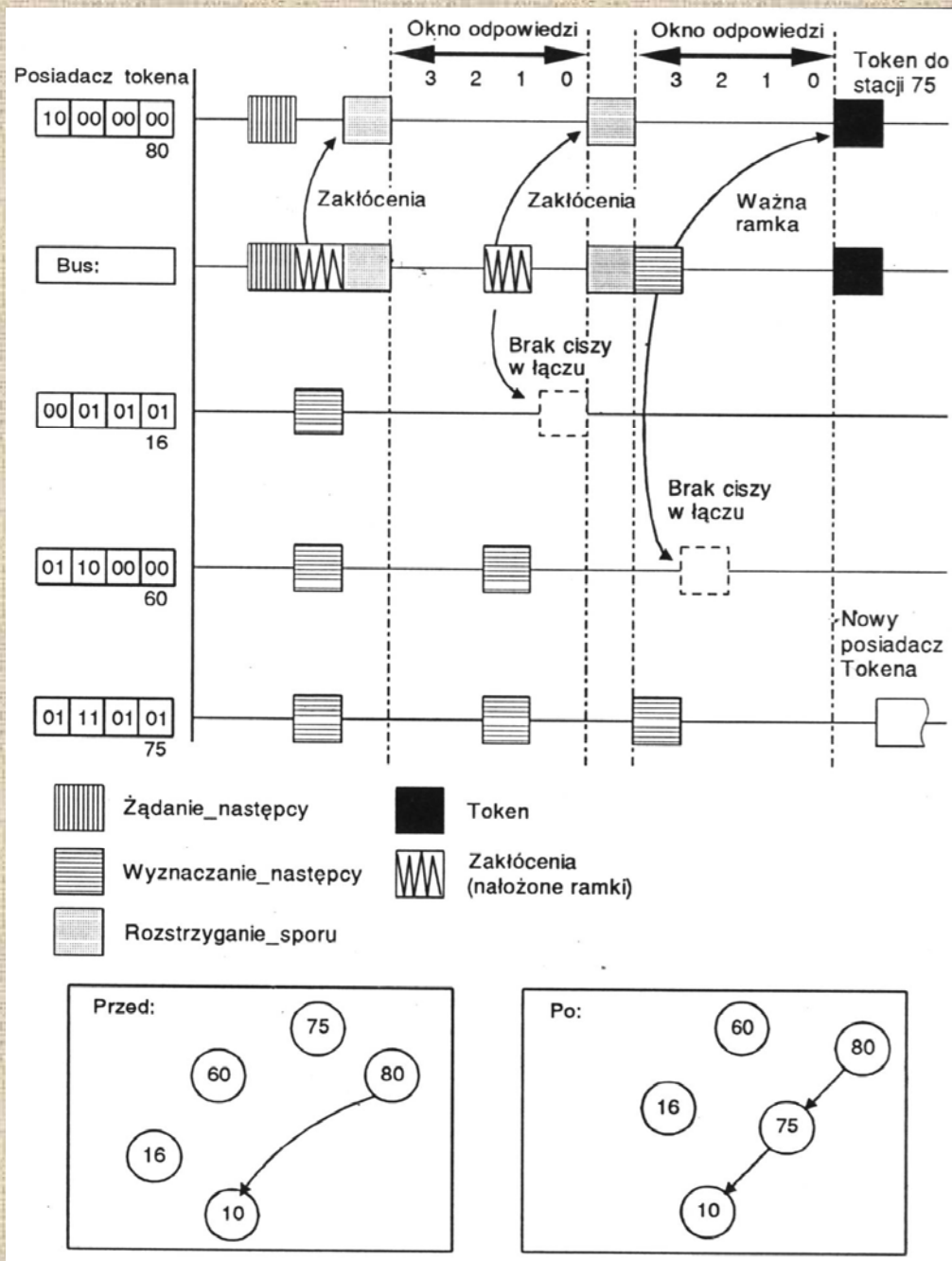
Funkcje pierścienia logicznego:

- (re)inicjalizacja pierścienia (przy włączeniu zasilania lub po rozpadzie),
- włączanie nowej stacji do istniejącego pierścienia (*resolve responder*). Każda stacja w pierścieniu zna adres poprzednika i następnika. Adresy te są określane dynamicznie, co umożliwia dołączanie i odłączanie stacji. Posiadacz tokena wysyła żądanie następcy do stacji o adresach pomiędzy jego adresem a adresem następnej stacji w pierścieniu i czeka na ew. zgłoszenie chęci dołączenia.
- opuszczanie pierścienia (zmiana adresów u poprzednika i następnika),
- ogólny nadzór nad pierścieniem, zarządzanie w przypadku wykrycia błędów, odtwarzanie zagubionego tokena.

<i>Preamble</i>	<i>Start Delimiter</i> SD	<i>Frame Control</i> FC	<i>Destination Address</i> DA	<i>Source Address</i> SA	<i>Data Units</i> DU	<i>Frame Check Sequence</i> FCS	<i>End Delimiter</i> ED
do synchronizacji odbiornika	Znacznik początku	Pole sterujące	Adres przeznaczenia	Adres źródła	Dane	Suma kontrolna ramki	Znacznik końca
min. 3 bajty	1 bajt	1 bajt	2 lub 6 bajtów	2 lub 6 bajtów	0 lub więcej bajtów	4 bajty	1 bajt

Rys. Format ramki w protokole TOKENBUS. Pole FC informuje o rodzaju ramki: dane LLC (*Logical link Control*) lub sterująca, np. *token*. Dane DU – przygotowane przez LLC dla komunikacji LLC-LLC lub przez MAC (*Medium Access Control*) do sterowania i zarządzania.

Komputerowe układy sterowania



Rys. Dołączanie stacji do pierścienia logicznego w systemie *token ring*.
 W przypadku zgłoszenia chęci dołączenia przez kilka stacji (nr 16, 60 i 75) żądanie następcy powtarza się po czasie okna odpowiedzi, przy czym zgłaszają się powtórnie tylko te stacje, których para najstarszych bitów numeru jest największa (nr 60 i 75, bity 01). Po ponownej kolizji decyduje wartość kolejnej pary bitów numeru (11 stacji nr 75 jest większe od 10 stacji nr 60). Do pierścienia włączona zostaje stacja nr 75 (w następnym obiegu tokena stacja nr 75 może dołączyć stację nr 60).

PROFIBUS DP

Protokół DP jest prostym i szybkim protokołem do *cyklicznej* wymiany danych pomiędzy stacją (stacjami) *master*, a stacjami *slave*. Stacja centralna Master (sterownik PLC, stacja PC) komunikuje się z rozproszonymi urządzeniami peryferyjnymi (we/wy cyfrowe lub analogowe, zawory, przetworniki, napędy).

Rozróżnia się dwa typy urządzeń *master*: DP-Master Class I – cyklicznie kontrolują układ w czasie normalnej pracy oraz DP-Master Class II - programatory sieci lub stacje kontrolno-diagnostyczne pracujące acyklicznie.

W przypadku struktury Multimaster prawo nadawania komunikatu przekazywane jest między stacjami *master* na zasadzie znacznika (*token passing*).



Rys. Struktura sieci PROFIBUS DP ze stacjami *master* i urządzeniami peryferyjnymi *slave*.

W PROFIBUS DP istnieją cztery różne komunikaty akcji:

Komunikat wysłania danych z potwierdzeniem – SDA (Send Data with Acknowledge) - wysłanie komunikatu akcji zawierającego dane i odebranie potwierdzenia dostarczenia komunikatu (jednoznakowego komunikatu SD5). Potwierdzenie wysyłane jest po otrzymaniu przez urządzenie podrzędne kompletnego komunikatu. W przypadku gdy komunikat nie dotarł lub wystąpił błąd podczas transmisji, odpowiedź nie jest wysyłana i następuje retransmisja komunikatu.

Komunikat wysłania danych bez potwierdzenia – SDN (Send Data with No acknowledge) - dane wysyłane są do urządzenia docelowego, grupy urządzeń, lub wszystkich urządzeń w sieci. Urządzenie nadrzędnie nie zostaje powiadomione o poprawności przesłanych danych, bo nie jest generowana odpowiedź.

Komunikat wysłania danych i odbioru odpowiedzi – SRD (Send and Request Data with reply) - komunikat akcji stacji nadrzędnej może zawierać żądanie i dane, lub jedynie żądanie. Komunikat odpowiedzi może zawierać żądane dane lub odpowiedź negatywną w przypadku braku żądanych danych. Odpowiedź musi być nadana zaraz po otrzymaniu żądania. Brak odpowiedzi lub błąd w odbiorze komunikatu odpowiedzi powoduje wysłanie komunikatu akcji z urządzenia nadrzędnego. Stacja nadrzędna wysyła komunikat o odebraniu z urządzenia podrzędnego odpowiedzi.

Komunikat Cyklicznego wysyłania danych i odbieranie odpowiedzi – CSRD (Cyclic Send and Request with reply) - umożliwia urządzeniu inicjującemu cykliczne odpytywanie jednego lub wielu urządzeń docelowych, z tak zwanej listy odpytywania (*poll list*). Proces polega na wysłaniu komunikatu do kolejnych zaadresowanych odbiorników umieszczonych na liście odpytywania i odebranie od nich odpowiedzi. Odpytywane są kolejne urządzenia aż do wyczerpania listy. Po wyczerpaniu listy odpytywanie zaczyna się od początku. To samo urządzenie może znajdować się na liście wielokrotnie. W budowie zapytania oraz odpowiedzi komunikat CSRD przypomina komunikat SRD z tą różnicą, że komunikaty CSRD mają niższy priorytet.

Mechanizmy zabezpieczeń

PROFIBUS DP oferuje zabezpieczenia zarówno urządzeń *master* jak i *slave*. Każda stacja podlega ochronie za pomocą mechanizmu kontroli czasowej (*time monitor*) definiowanego podczas konfiguracji sieci.

Stacje *master* wykorzystują tzw. *Data Control Timer* ustawiony odpowiednio dla każdej stacji *slave*. Jeżeli nastąpi błąd w transmisji, włączany jest system kontroli czasowej. Po upływie przewidzianego dla danego urządzenia *slave* czasu, *master* przyjmuje że została utracona komunikacja. Jeżeli parametr *Auto Clear=True* dla danego *slave'a*, rozpoczęty zostaje proces automatycznej obsługi błędów i zawieszona zostaje wymiana danych, a wyjścia *slave'a* zostają przełączone w stan bezpieczny (o ile to możliwe).

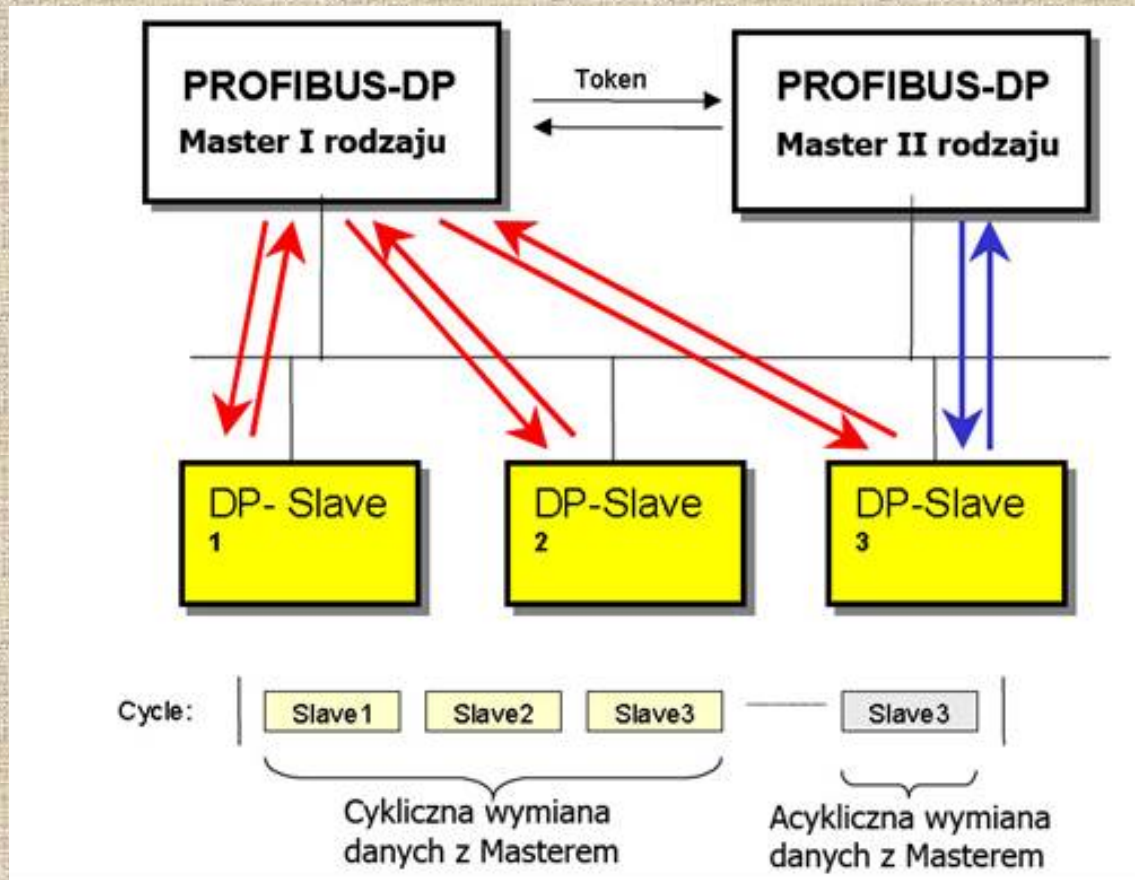
Stacje *slave* w celu zabezpieczenia transmisji danych wykorzystują mechanizm *watchdog*. W przypadku utraty komunikacji ze stacją *master*, po upływie czasu określanego jak *Watchdog Time*, stacja *Slave* przestawia stan wyjść w stan bezpieczny. Czas reakcji modułu *watchdog* powinien być krótszy od czasu zadziałania zabezpieczenia stacji *master Data Control Timer*.

Stacje *slave* posiadają również zabezpieczenie przed dostępem, pozwalające jedynie uprawnionej stacji *master* na zmianę stanu wyjść. Wszystkie inne stacje *master* mogą mieć dostęp tylko do obrazu wejść oraz do odczytu stanu.

DP-V0 to podstawowe opcje i funkcje pozwalające na cykliczną pracę i diagnostykę modułów i kanałów. Służy do cyklicznego odczytu stanów wejść oraz cyklicznego zapisu wyjść stacji *slave*. *Master* odpytuje kolejne przyporządkowane mu stacje *slave*. Transmisja do 12 Mb/s pozwala np. na przesłanie 512 bitów wejściowych i 512 bitów wyjściowych dla 32 stacji w czasie ok. 1 ms.

DP-V1 jest rozszerzoną wersją protokołu DP-V0. Oprócz cyklicznej wymiany danych, możliwa jest także *acykliczna* komunikacja urządzenia master z wybranymi urządzeniami *slave*.

Acykliczna wymiana danych pozwala m.in. na zastosowanie parametryzacji oraz kalibrację urządzeń slave w czasie ich normalnej pracy, oraz na wprowadzenie różnego rodzaju alarmów. Acykliczna transmisja danych wykonywana jest równoległe do cyklicznej.

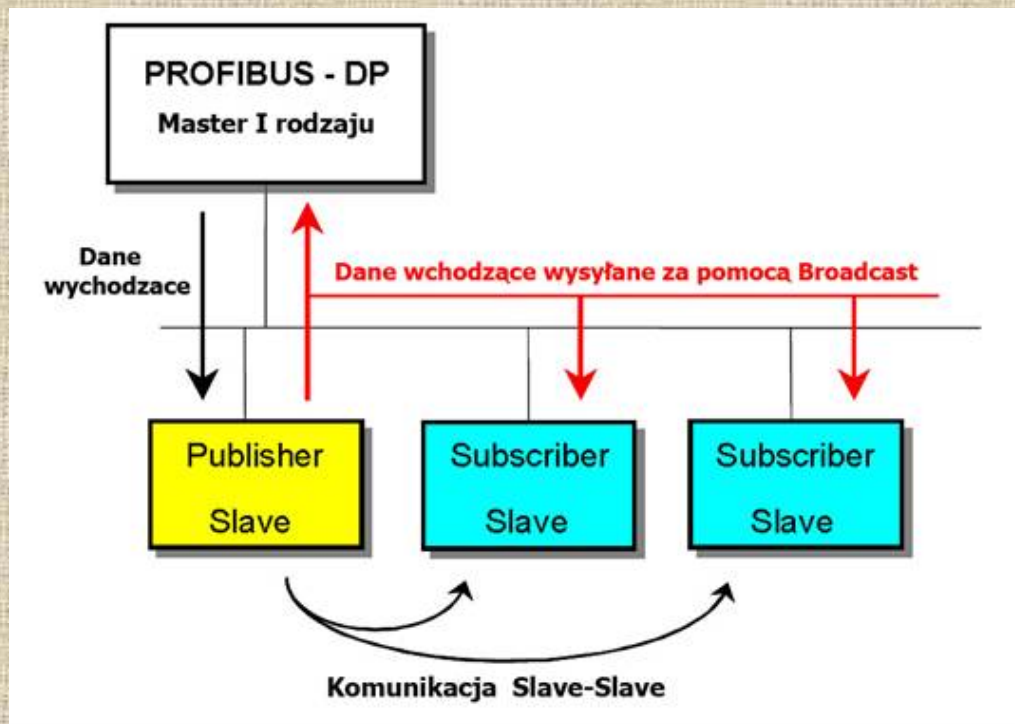


Rys. Cykliczna i acykliczna wymiana danych w DP-V1

DP-V2 jest najszerszą wersją protokołu Profibus DP. Umożliwia komunikację slave-slave, wprowadza także tryb izochroniczny oraz Clock Control.

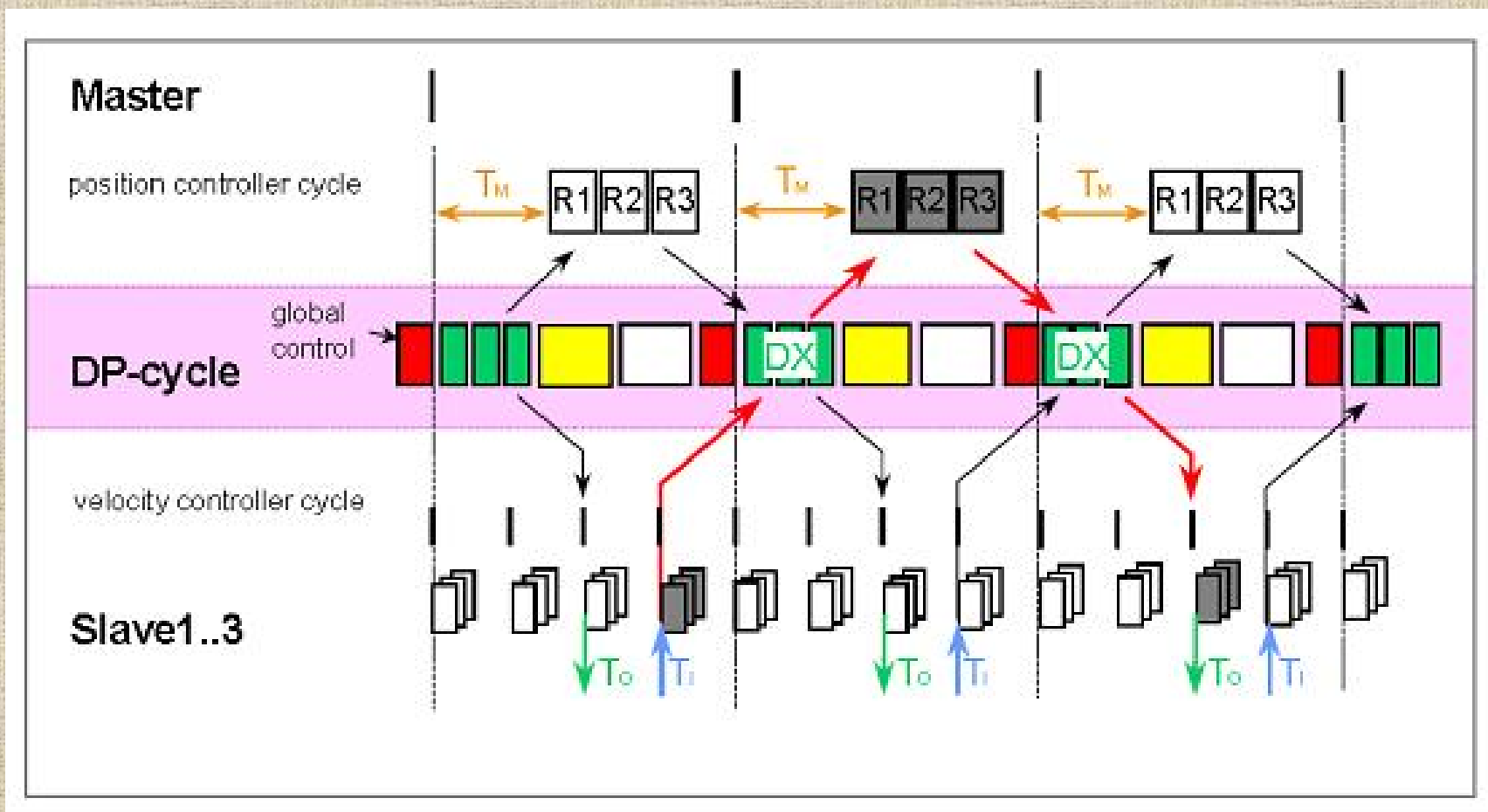
Komunikacja slave-slave (DXB) wykorzystuje tryb *broadcast* do komunikacji między stacjami *slave* (*Publisher* do *Subscribers*) z pominięciem stacji *master*. Pozwala to na bezpośredni odczyt danych z innych stacji, oszczędza do 90% czasu odpowiedzi i rozszerza możliwości aplikacyjne.

Tryb izochroniczny umożliwia wprowadzenie sterowania synchronicznego pomiędzy stacjami *master* i *slave* niezależnie od obciążenia sieci. Możliwe jest pozycjonowanie z odchyłką zegara rzędu 1 ms. Stacje synchronizują się cykl do cyklu z wykorzystaniem trybu *broadcast*.



Funkcja Clock control synchronizuje wszystkie stacje Slave z czasem systemowym z rozdzielczością na poziomie milisekund. Master wysyła tzw. *time stamp* do stacji slave poprzez specjalny serwis.

Rys. Komunikacja *slave-slave* między stacjami w DP-V2.



Rys. Komunikacja izochroniczna w DP-V2.

CAN (Controller Area Network)

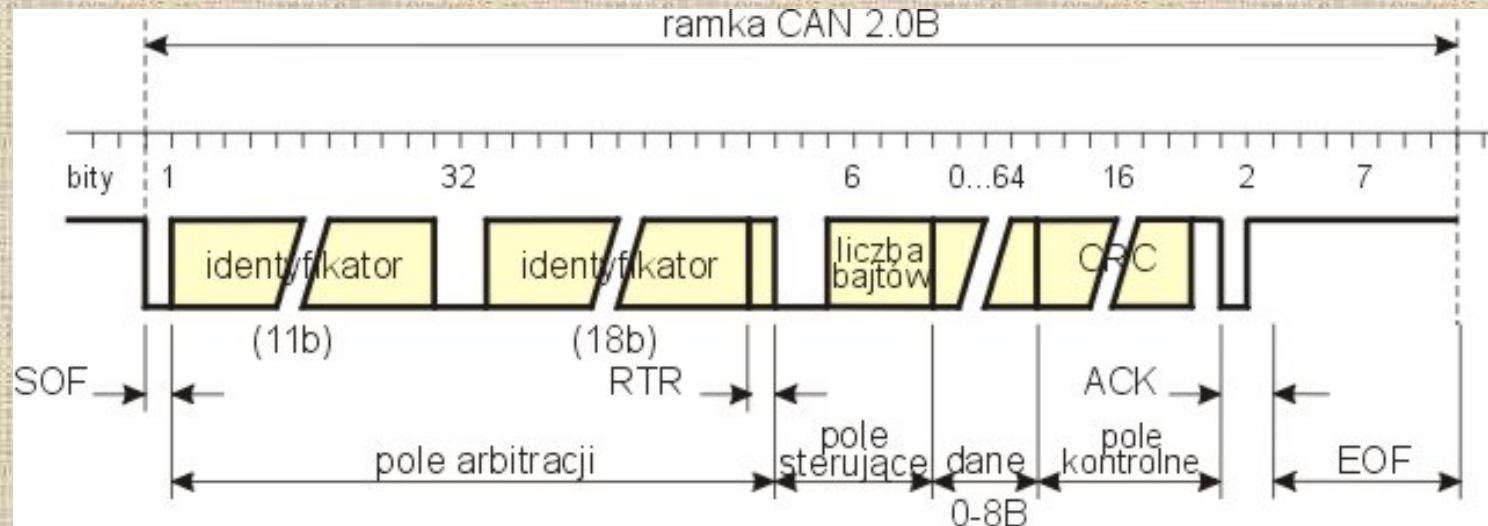
CAN jest protokołem komunikacji szeregowej opracowanym w firmie Bosch (w latach 80-tych) z myślą o zastosowaniach w przemyśle samochodowym (ABS, sterowanie silnika). Obecnie znajduje zastosowanie również w przemysłowych magistralach polowych *fieldbus* (CANopen, DeviceNet i in.)

W CAN nie ma wyodrębnionej jednostki nadrzędnej, dlatego należy do protokołów *Multi-Master*. Komunikacja ma charakter rozgłoszeniowy (*broadcast*), komunikaty nadawane na magistralę obierane są przez wszystkie urządzenia.

Główne cechy standardu CAN:

- Pojemność sieci – ponad 5 mln komunikatów,
- Maksymalna szybkość transmisji – 2.5 Mb/s,
- Maksymalna długość linii – 10 km,
- Media transmisyjne - różne

Magistrala CAN wykorzystuje zwykle skrętkę dwuprzewodową i pracuje z prędkością do 1 Mb/s przy dystansie do 40m.



Rys. Ramka komunikatu CAN w standardzie 2.0B. Ramka składa się z 7 pól: początku, arbitracji (32 bity w tym 29-bitowy identyfikator, w 2.0A odpowiednio 12 i 11 bitów), sterującego, danych (do 8 bajtów), sumy kontrolnej, potwierdzenia i końca.

Najważniejsze cechy komunikacji CAN:

- komunikaty rozpoznawane przez identyfikatory, - automatyczna obsługa dostępu do magistrali, - sprzętowa obsługa błędów.

Identyfikator jest przypisany do komunikatu, a nie do urządzenia, i określa priorytet dostępu do magistrali. Dostęp jest przyznawany metodą dominacji bitowej: wszystkie stacje badają stan magistrali czekając na możliwość wysłania własnego komunikatu. Im mniejsza wartość liczbową identyfikatora, tym większy priorytet.

Konflikty równoczesnego nadawania są rozwiązywane w trakcie wysyłania pola arbitracji zawierającego identyfikator. Jeżeli medium transmisyjne posiada własność dominacji zera (jak w przypadku tzw. iloczynu na drucie), wysłanie przez jedną stację wartości logicznej 0, a przez drugą 1 powoduje, że na magistrali ustala się 0. Dostęp do łącza otrzyma stacja o niższym numerze identyfikacyjnym.

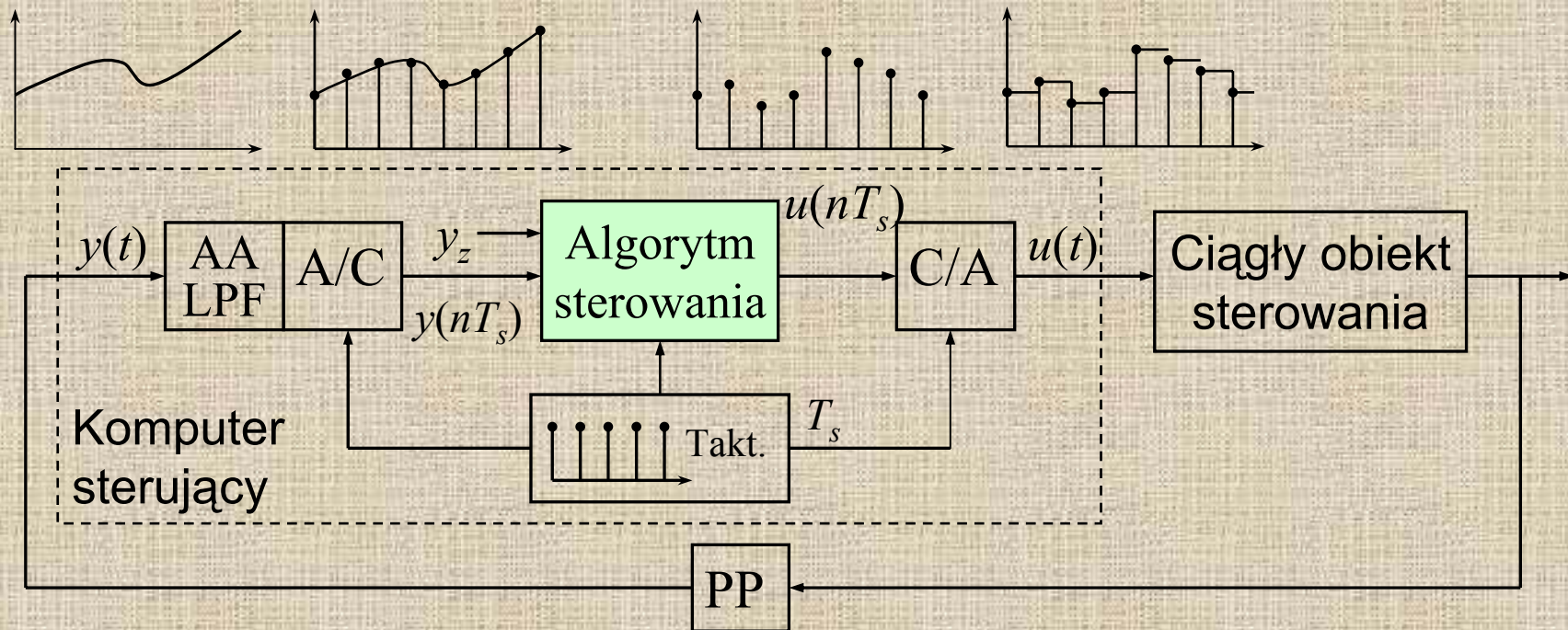
Warunek poprawnej arbitracji wymaga, aby dwa urządzenia nie mogły nadawać komunikatów o tym samym identyfikatorze.

CAN charakteryzuje się odpornością na zakłócenia i niezawodnością. Uzyskano to poprzez nadawanie danych w postaci napięciowego sygnału różnicowego. Specjalizowane kontrolery formują komunikaty, sterują bezkolizyjnym dostępem do magistrali i filtrują komunikaty.

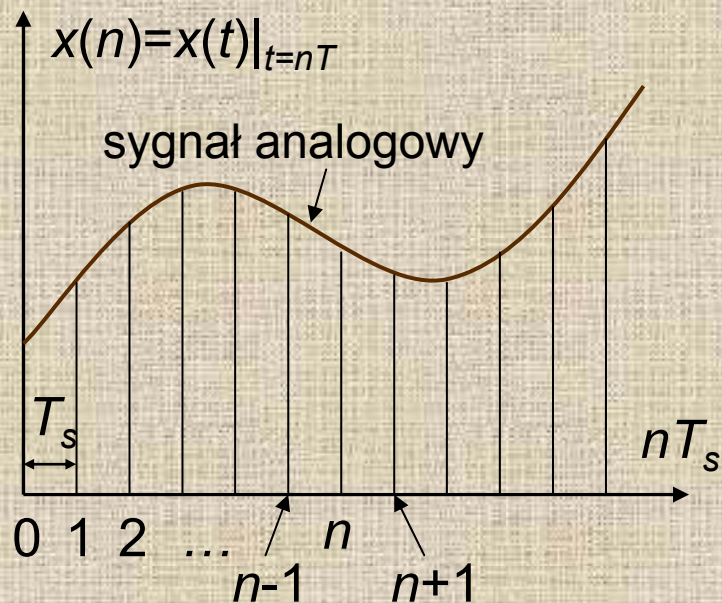
Większość czołowych firm elektronicznych produkuje kontrolery CAN jako układy peryferyjne lub wbudowane np. w mikrokontrolery.

Opis teoretyczny sygnałów i układów z czasem dyskretnym

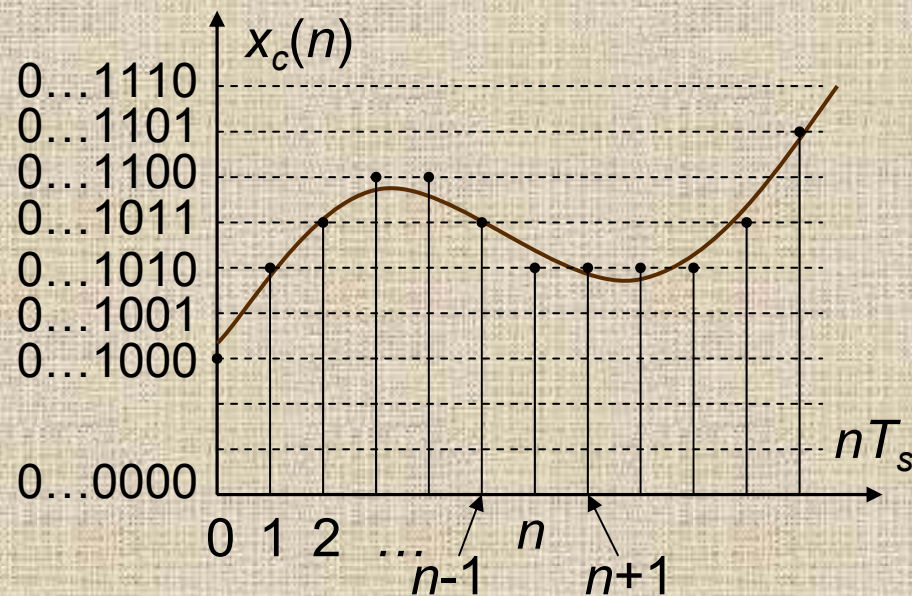
Regulacja dyskretna wielkości analogowej w układzie ze sprzężeniem zwrotnym



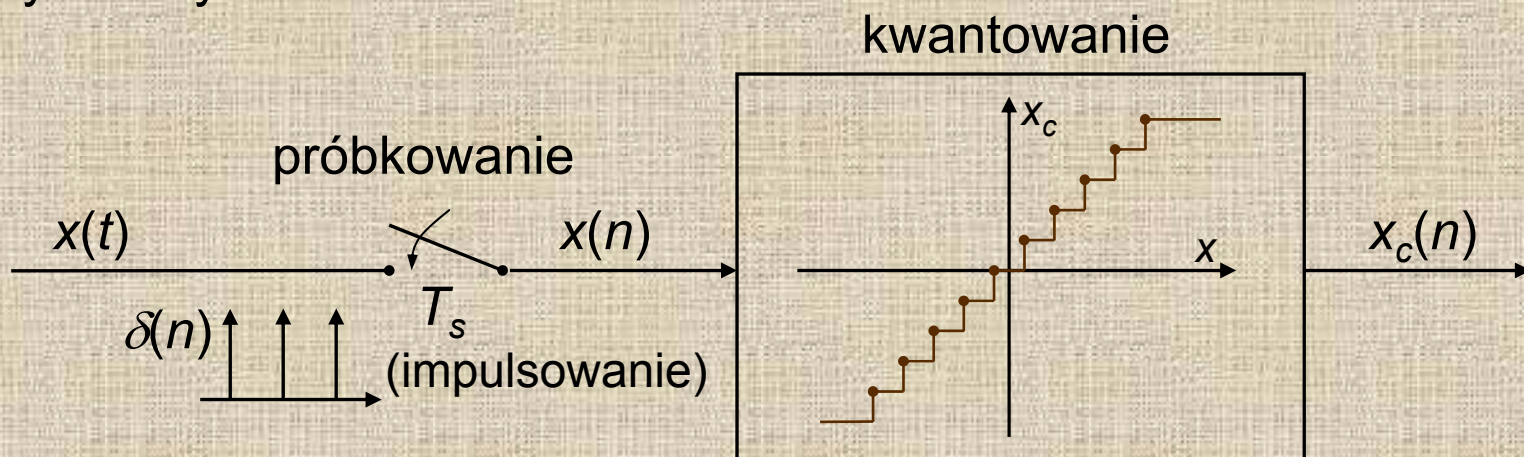
Rys. Elementy i sygnały kanału dyskretnego (cyfrowego) układu sterowania:
 AA LPF – dolnoprzepustowy filtr antyaliasingowy (analogowy),
 PP – czujnik i przetwornik pomiarowy,
 T_s – okres próbkowania



próbkowanie → sygnał dyskretny w czasie



kwantowanie → sygnał cyfrowy



Rys. Przetwarzanie A/C – próbkowanie i kwantowanie sygnału analogowego

Próbkowanie sygnału analogowego $x_a(t)$ jest opisywane z użyciem funkcji impulsowej:

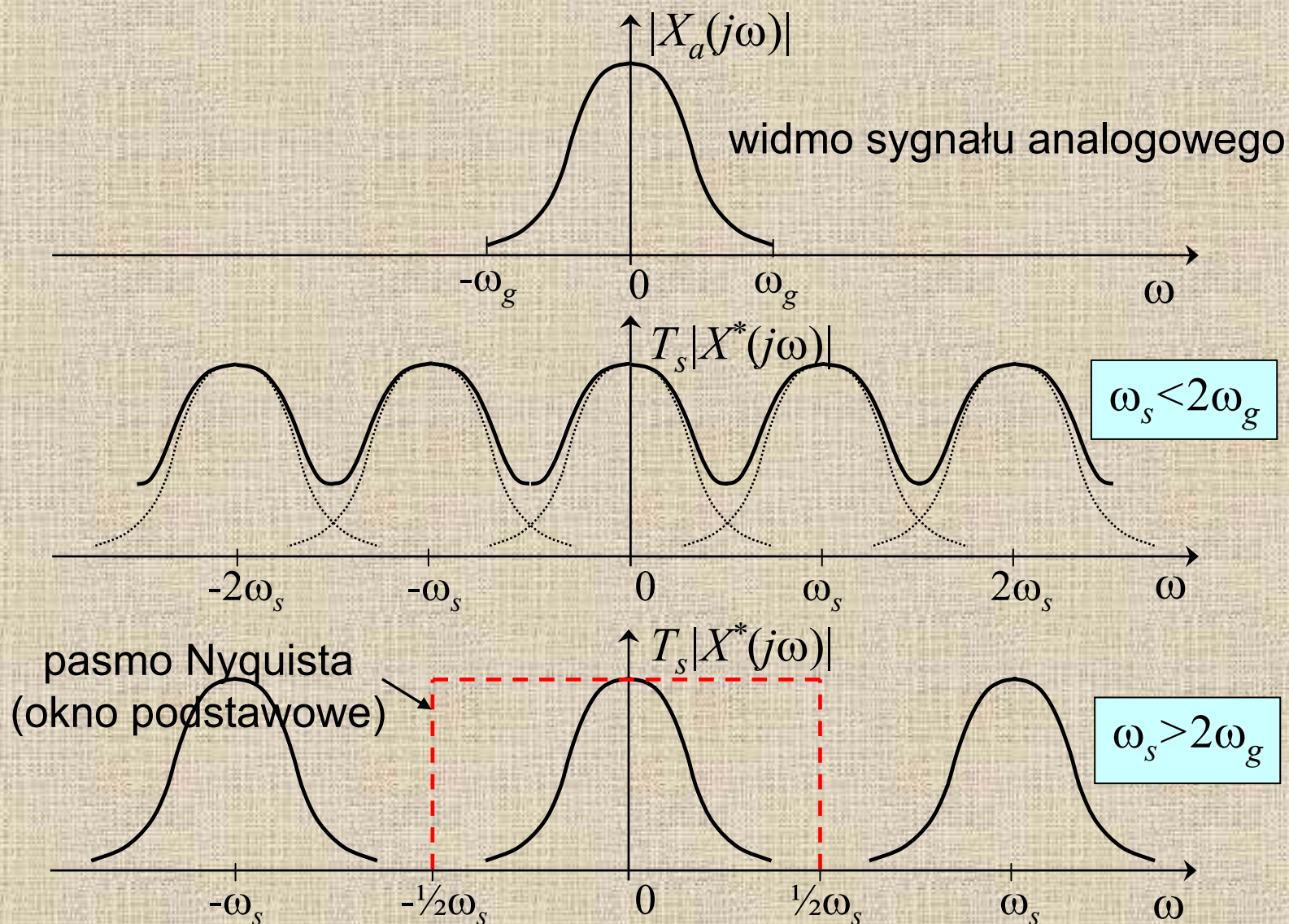
$$x^*(t) = \sum_{n=0}^{\infty} x_a(t) \delta(t - nT_s)$$

Widmo Fouriera funkcji impulsowej:

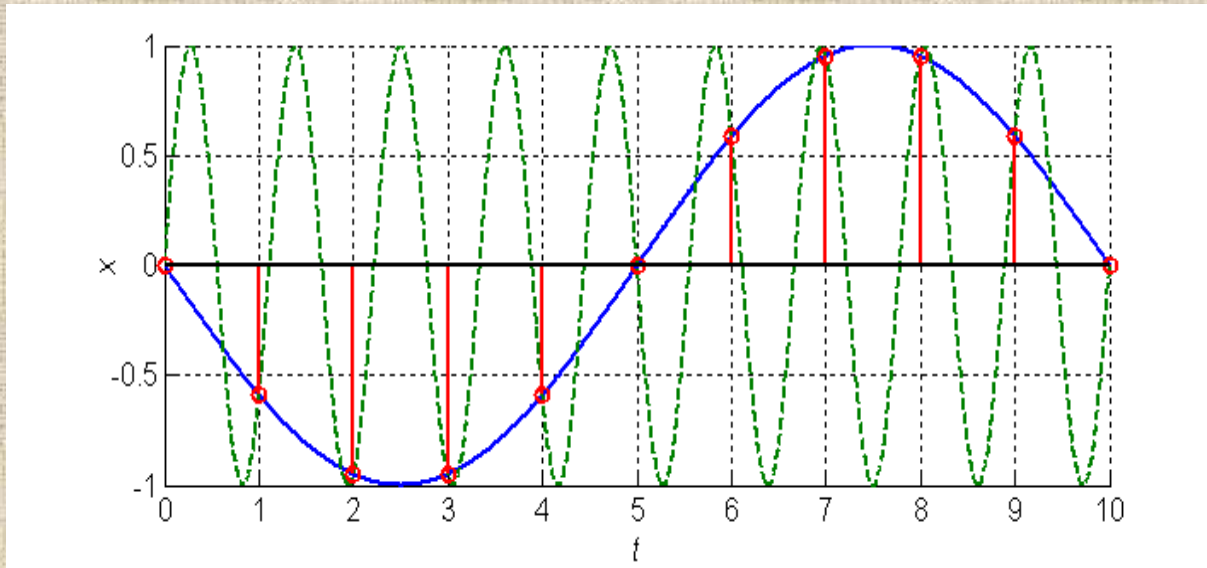
$$X^*(j\omega) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X_a(j\omega - jk\omega_s)$$

$$\omega_s = 2\pi f_s, \quad f_s = \frac{1}{T_s}$$

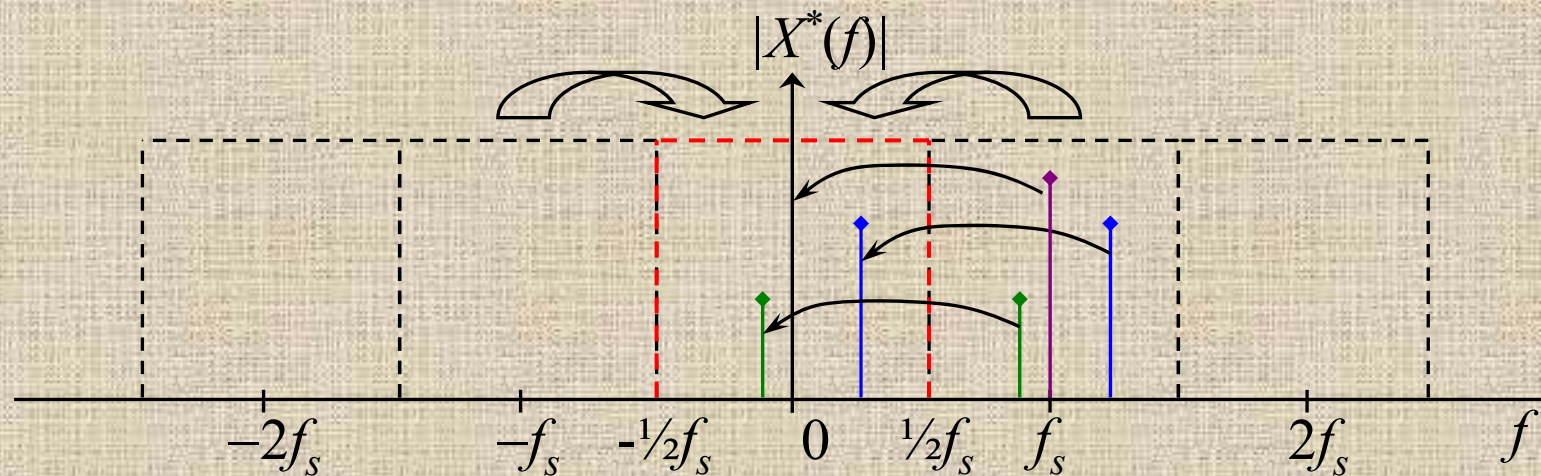
Wniosek: widmo sygnału dyskretnego w czasie (sygnału impulsowego) jest sumą poprzesuwanych o wielokrotność pulsacji próbkowania widm oryginalnego sygnału analogowego (z którego pochodzą próbki).



Rys. Moduł widma Fouriera sygnału ciągłego $x_a(t)$ oraz powielone widma sygnału impulsowego $x^*(nT_s)$. W praktyce $\omega_s = 5 \div 10 \omega_g$



Rys. *Aliasing* pokazany w dziedzinie czasu: dwa sygnały o częstotliwościach 0.9Hz i 0.1Hz mogą mieć takie same wartości w chwilach próbkowania, $f_s=1\text{Hz}$

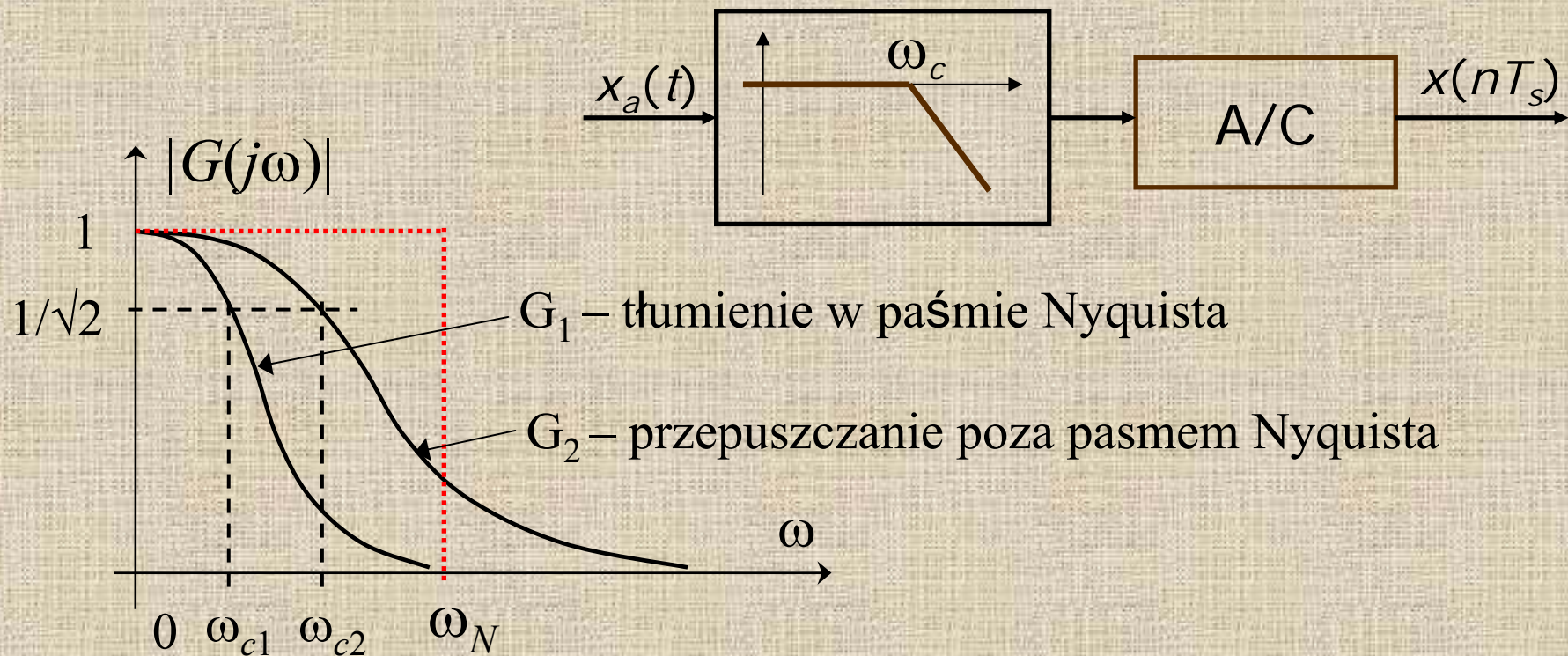


Rys. *Aliasing* w dziedzinie częstotliwości: nakładanie się okien wokół wielokrotności f_s na pasmo podstawowe

$\frac{1}{2}f_s = f_N$ – częstotliwość Nyquista

Filtracja antyaliasingowa

Analogowy filtr LP

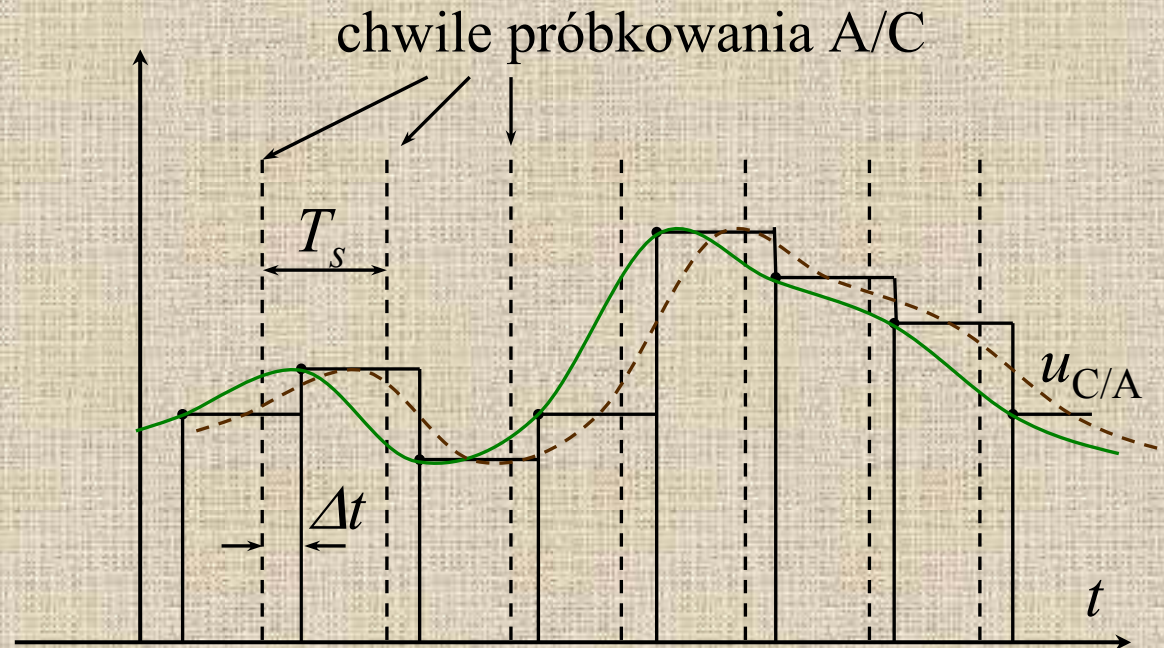


Rys. Problem wyboru szerokości pasma filtra antyaliasingowego. Zwykle filtr projektuje się tak, żeby $0.001 \leq |G(j\omega_N)| \leq 0.5$

Możliwości:

- 1) zwiększenie stromości opadania ch-ki filtra poza pasmem przepuszczania (rzędu filtra) → bardziej skomplikowana część analogowa układu
- 2) zwiększenie częstotliwości próbkowania (szerokości okna Nyquista) → potrzebny szybszy procesor (większa częstość obliczeń) i przetwornik A/C

Przetwarzanie C/A

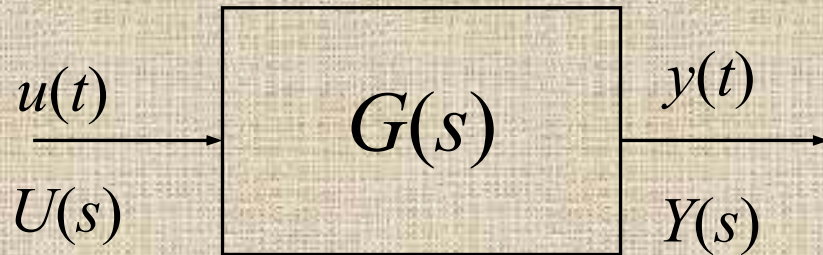


Rys. Opóźnienie ułamkowe chwil generowania sygnału sterującego na wyjściu C/A względem chwil próbkowania i opóźnienie wnoszone przez ekstrapolację zerowego rzędu ZOH (*zero order hold*) względem hipotetycznego sygnału ciągłego, równe $\frac{1}{2}T_s$.

Czas przetwarzania A/C, czas obliczania sterowania przez algorytm i czas przetwarzania C/A powodują opóźnienie generowania sterowania $u(t)$ w stosunku do momentów próbkowania:

$$\Delta t = t_{A/C} + t_{obl} + t_{C/A}$$

Liniowe równanie różniczkowe i transmitancja ciągła $G_a(s)$



Rys. Model jednowymiarowego układu dynamicznego liniowego czasu ciągłego

$$\begin{aligned} a_n y^{(n)} + a_{n-1} y^{(n-1)} + \dots + a_1 \dot{y} + a_0 y(t) &= \\ &= b_m u^{(m)} + b_{m-1} u^{(m-1)} + \dots + b_1 \dot{u} + b_0 u(t) \end{aligned}$$

Transmitancja ciągła:

$$G(s) = \left. \frac{Y(s)}{U(s)} \right|_{WP=0} = \frac{b_m s^m + \dots + b_1 s + b_0}{a_n s^n + \dots + a_1 s + a_0} \quad n \geq m$$

Właściwości dynamiczne (przebiegi swobodne) układu opisanego transmitancją $G(s)$ zależą od miejsc zerowych równania charakterystycznego (biegunów układu $s_i = a_i \pm j\omega_i$, $i=1, 2, \dots, n$):

$$a_n s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0 = 0 \Leftrightarrow a_n (s - s_1)(s - s_2) \dots (s - s_n) = 0$$

Liniowy układ dynamiczny $G(s)$ jest stabilny wtedy i tylko wtedy, kiedy *wszystkie* bieguny $G(s)$ układu mają ujemne części rzeczywiste (leżą w lewej półpłaszczyźnie zmiennej zespolonej s). Składowa przejściowa (swobodna) takiego układu zanika w czasie do zera.

Transmitancja widmowa (charakterystyka częstotliwościowa):

$$G(j\omega) = |G(j\omega)| \cdot e^{j \cdot \arg G(j\omega)}$$

Charakterystyka amplitudowa:

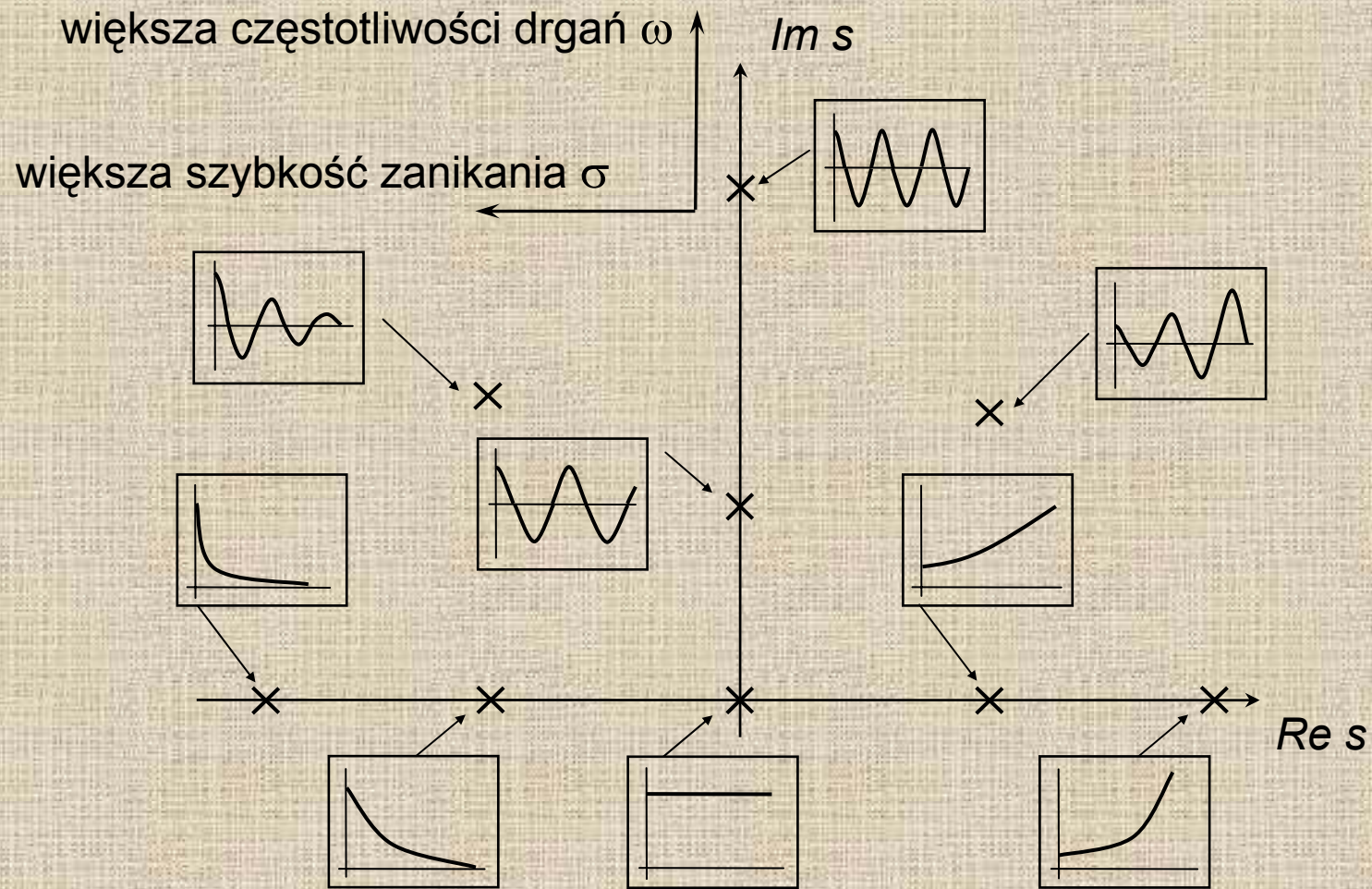
$$M(\omega) = |G(j\omega)|, \quad Lm(\omega) = 20 \log_{10} |G(j\omega)| \quad [\text{dB}]$$

Charakterystyka fazowa: $\varphi(\omega) = \arg G(j\omega)$

Model układu dynamicznego (wielowymiarowego) w formie równań stanu:

$$\begin{aligned} \frac{d\mathbf{x}}{dt} &= \mathbf{Ax} + \mathbf{Bu} \\ \mathbf{y} &= \mathbf{Cx} + \mathbf{Du} \end{aligned} \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

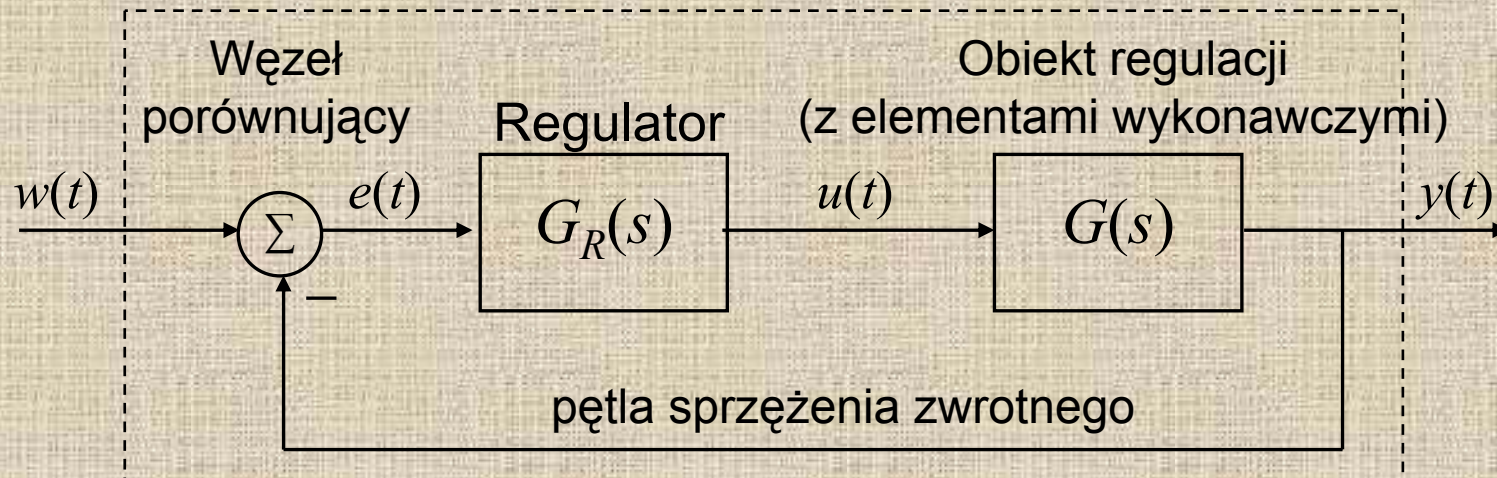
$\mathbf{x}(t)$ – wektor zmiennych stanu, $\mathbf{u}(t)$ – wektor wejść, $\mathbf{y}(t)$ – wektor wyjść



Rys. Mapa przebiegów przejściowych związanych z położeniem biegunów układu na płaszczyźnie zmiennej zespolonej s .

Dla ujemnych części rzeczywistych stosuje się oznaczenie $|\alpha| = \sigma$

Układ regulacji ze sprzężeniem zwrotnym



Sygnały (wielkości) podstawowego schematu zamkniętego układu regulacji ze sprzężeniem zwrotnym (ujemnym):

$y(t)$ - wielkość sterowana

$u(t)$ – wielkość sterująca

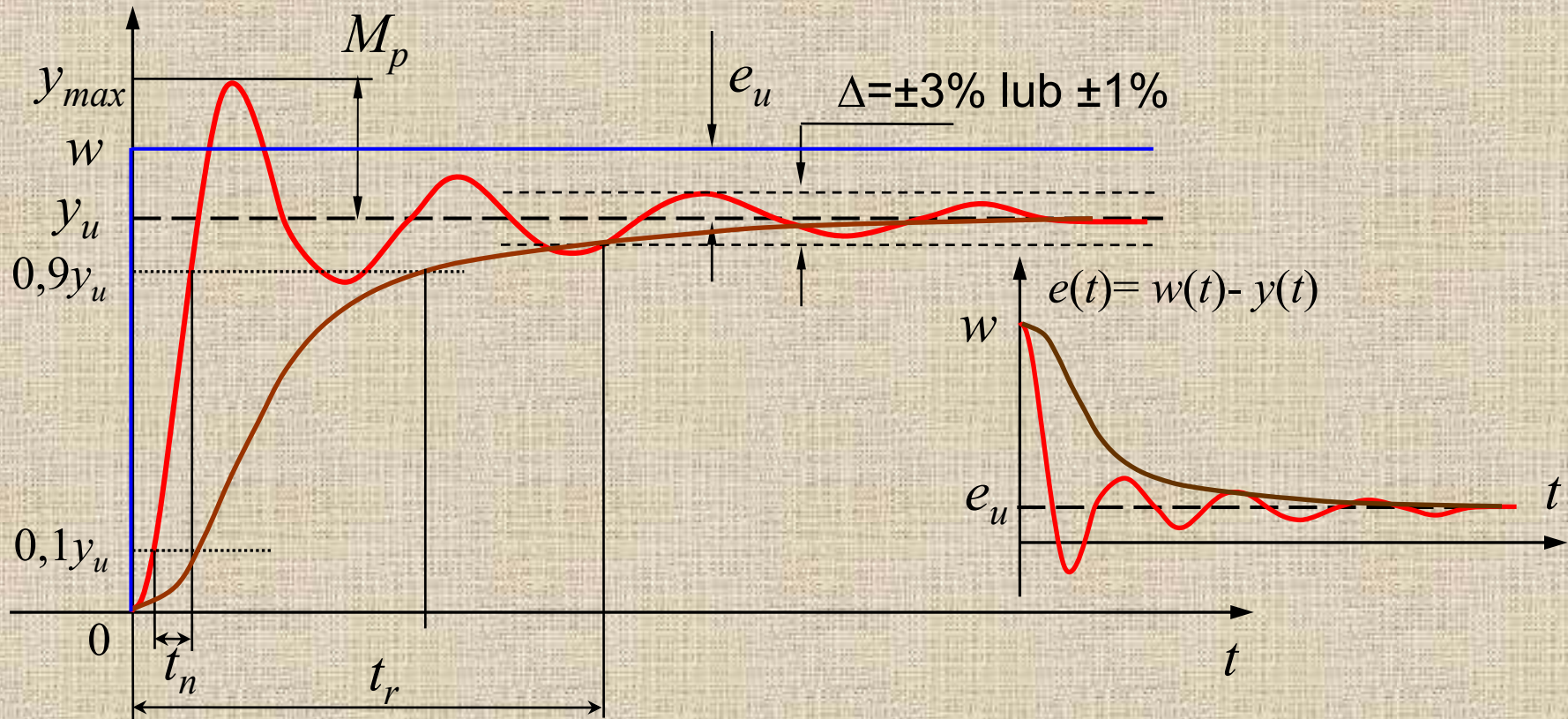
$w(t)$ – wartość zadana wielkości sterowanej

$e(t) = w(t) - y(t)$ – błąd (uchyb) regulacji

Zadanie regulacji: wielkość sterowana $y(t)$ ma odwzorowywać przebieg wielkości zadanej $w(t)$.

Transmitancja zastępcza zamkniętego układu regulacji ciągłej:

$$G_z(s) = \frac{Y(s)}{W(s)} = \frac{G_R(s) \cdot G(s)}{1 + G_R(s) \cdot G(s)}$$



Rys. Przebieg oscylacyjnej i aperiodycznej odpowiedzi układu regulacji przy skokowej zmianie wartości zadanej $w(t)$ (wymuszenia)

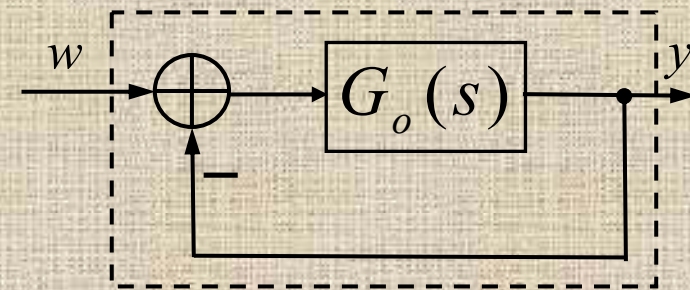
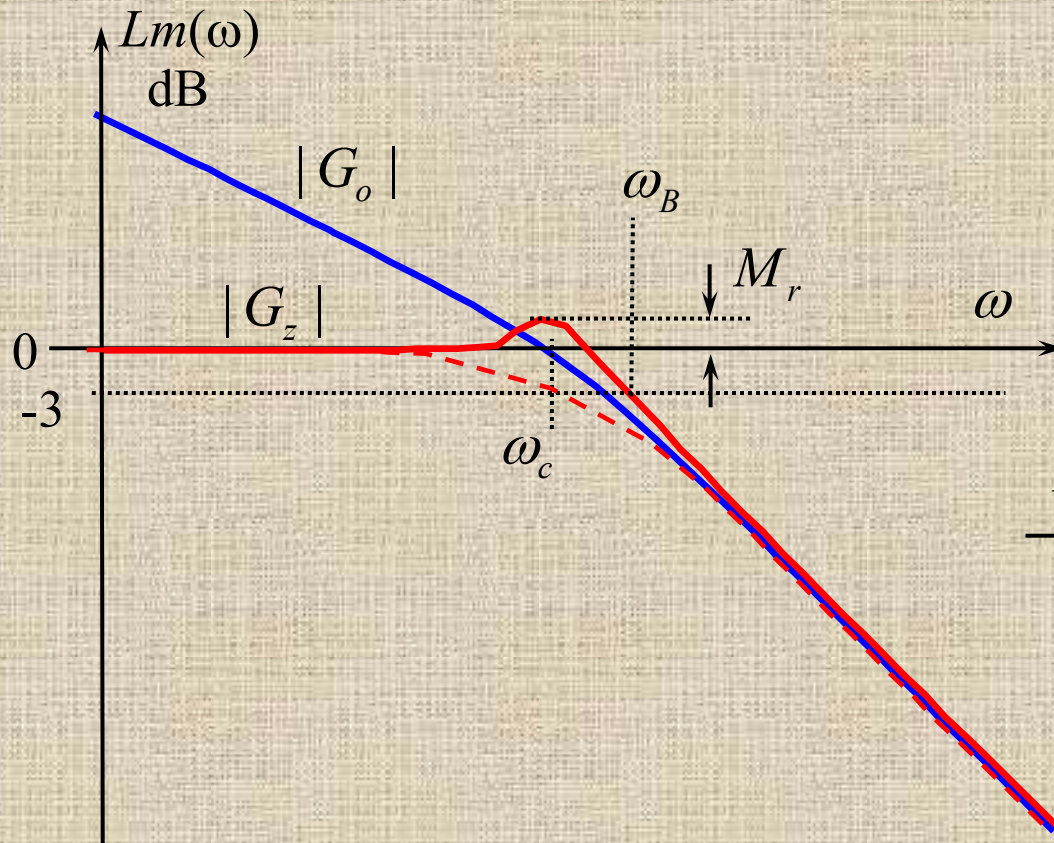
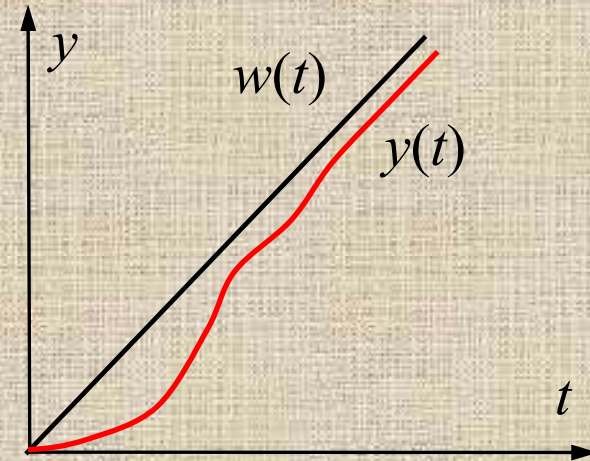
t_r - czas ustalania (regulacji); określa czas trwania przebiegów przejściowych w układzie i jest miarą stabilności i szybkości regulacji

t_n - czas narastania; określa szybkość działania układu regulacji

M_p - max. przeregulowanie (%); jest miarą stabilności układu regulacji

e_u - błąd (uchyb) ustalony regulacji; określa dokładność statyczną regulacji

Rys. Przykładowy przebieg odpowiedzi układu zamkniętego na wymuszenie liniowe $w(t)=a \cdot t$ (istotna w przypadku układów regulacji nadążnej)



Rys. Typowy przebieg charakterystyki amplitudowej układu otwartego G_o i układu zamkniętego $G_z = G_o / (1 + G_o)$;

ω_c – pulsacja odcięcia (*cut-off*), ω_B – pasmo przenoszenia (*bandwidth*)

Jeżeli układ zamknięty (nawet jeśli jest to układ wyższego rzędu) ma 2 bieguny dominujące, to można go *aproksymować* transmitancją członu II rzędu:

$$G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

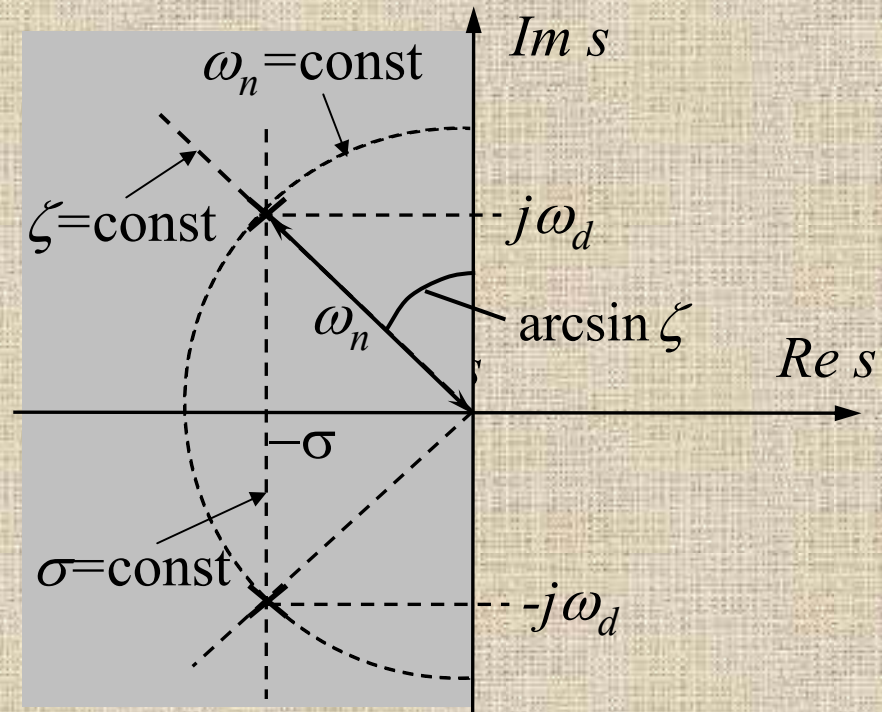
Analizując wzór na charakterystykę skokową tego członu można podać zależności między parametrami transmitancji a wskaźnikami jakości:

$$t_r = \frac{-\ln \Delta}{\sigma} = \frac{\beta}{\sigma}, \text{ gdzie } \beta = \begin{cases} 4.6 \text{ dla } \Delta e = 1\% \\ 3.5 \text{ dla } \Delta e = 3\% \end{cases}, \quad t_n \approx \frac{\alpha}{\omega_n}, \text{ gdzie } \alpha = \begin{cases} 1.4 \text{ dla } \zeta = 0.3 \\ 1.8 \text{ dla } \zeta = 0.5, \\ 2.1 \text{ dla } \zeta = 0.7 \end{cases}$$

$$M_p = \exp\left(\frac{-\pi\sigma}{\omega_d}\right) \cdot 100\%$$

gdzie: $\sigma = \zeta\omega_n$, $\omega_d = \omega_n\sqrt{1-\zeta^2}$ Peak time: $t_{max} = \pi/\omega_d$

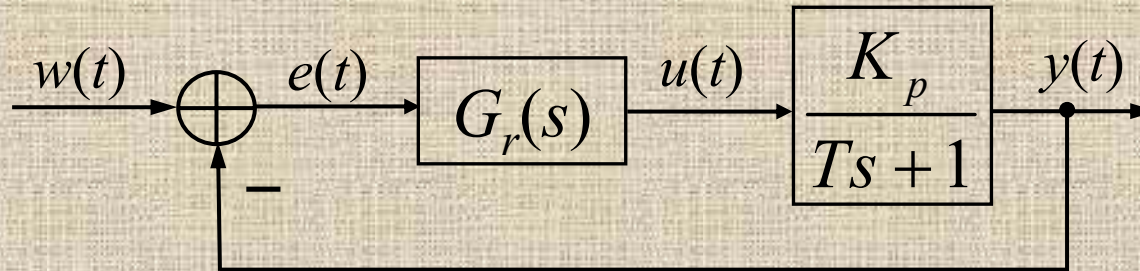
Często spotykane wartości przybliżone: $M_p \cong \begin{cases} 5\% \text{ dla } \zeta = 0.7 \\ 15\% \text{ dla } \zeta = 0.5 \\ 35\% \text{ dla } \zeta = 0.3 \end{cases}$



Rys. Linie stałych wartości parametrów dynamicznych (ζ , σ , ω_n) oscylacyjnej i aperiodycznej odpowiedzi układu regulacji przy skokowej zmianie wartości zadanej $w(t)$ (*wymuszenia*)

Znaczenie działania całkującego

Przykład: Wyznaczyć uchyb ustalony odpowiedzi skokowej ($w(t)=A \cdot 1(t)$) układu zamkniętego dla: 1) $G_r(s)=K_r$, 2) $G_r(s)=K_r/s$



$$1) \quad G_z(s) = \frac{K_r K_p}{Ts + 1 + K_r K_p}$$

układ zamknięty jest układem inercyjnym o stałej czasowej $T/(1+K_r K_p)$ i współczynniku wzmocnienia $K_r K_p / (1+K_r K_p)$

$$E(s) = W(s) - Y(s) = [1 - G_z(s)]W(s)$$

$$E(s) = \left(1 - \frac{K_r K_p}{Ts + 1 + K_r K_p} \right) \frac{A}{s} = \frac{A(Ts + 1)}{s(Ts + 1 + K_r K_p)}$$

$$e_u = \lim_{s \rightarrow \infty} sE(s) = \frac{A}{1 + K_r K_p}$$

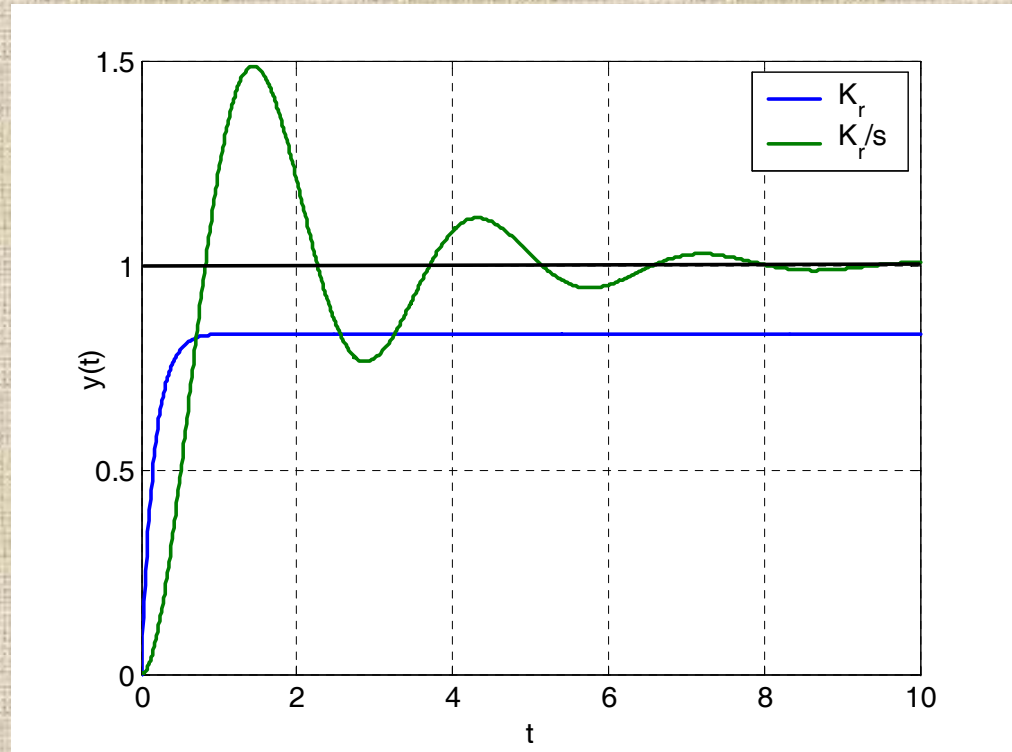
$$2) G_z(s) = \frac{K_r K_p}{s(Ts + 1) + K_r K_p} = \frac{K_r K_p / T}{s^2 + s/T + K_r K_p / T}, \quad \zeta = \frac{1}{2\sqrt{K_r K_p T}}, \quad \omega_n = \sqrt{\frac{K_r K_p}{T}}$$

Układ zamknięty jest układem II rzędu (inercyjnym dla $\zeta > 1$ lub oscylacyjnym dla $0 < \zeta < 1$) o współczynniku wzmocnienia równym 1.

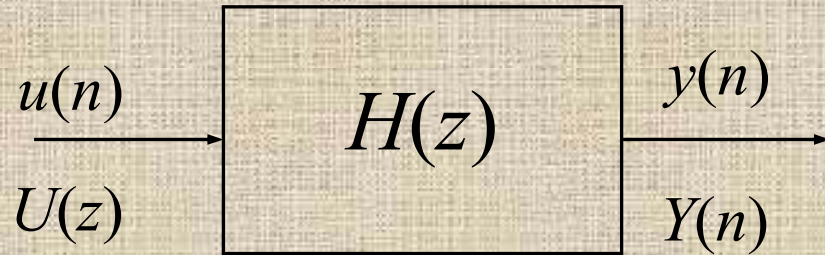
$$E(s) = \left(1 - \frac{K_r K_p}{s(Ts + 1) + K_r K_p} \right) \frac{A}{s} = \frac{A(Ts + 1)}{s(Ts + 1) + K_r K_p}$$

$$e_u = \lim_{s \rightarrow 0} sE(s) = 0$$

Rys. Charakterystyki skokowe, $w(t)=\mathbf{1}(t)$, układu zamkniętego. Parametry $K_r=1$, $K_p=5$, $T=1$. (Zmniejszenie wsp. wzmocnienia regulatora całkującego spowoduje zmniejszenie przeregulowania).



Liniowe równanie różnicowe, przekształcenie Z i transmitancja dyskretna $H(z)$



Rys. Model jednowymiarowego układu dynamicznego liniowego czasu dyskretnego

$$a_0 y(n) + a_1 y(n-1) + \dots + a_{N_a} y(n-N_a) = b_0 u(n) + b_1 u(n-1) + \dots + b_{N_b} u(n-N_b)$$

Zwykle przyjmuje się $N_a = N_b = N$, gdzie $N = \max(N_a, N_b)$ - rząd układu, oraz $a_0 = 1$.

Transformata Z (jednostronna):

$$Z[x(n)] = X(z) \triangleq \sum_{n=0}^{\infty} x(n) z^{-n} = x(0) + x(1)z^{-1} + x(2)z^{-2} + \dots$$

Transformata odwrotna:

$$Z^{-1}[X(z)] = x(n) = \frac{1}{2\pi j} \oint_C X(z) z^{n-1} dz$$

Właściwości przekształcenia Z

	Funkcja dyskretna	Transformata Z
Liniowość	$a_1 x_1(n) + a_2 x_2(n)$	$a_1 X_1(z) + a_2 X_2(z)$
Zmiana skali częstotliwości zespolonej	$e^{-an} x(n)$	$X(ze^a)$
Splot	$x_1(n) * x_2(n) = \sum_{k=0}^n x_1(k) \cdot x_2(n-k)$	$X_1(z) \cdot X_2(z)$
Przesunięcie w czasie	$x(n-k)$ $x(n+1)$	$z^{-k} \left[X(z) + \sum_{j=1}^k x(-j)z^j \right]$ $zX(z) - x(0)$
Różniczkowanie transformaty	$n \cdot x(n)$	$-z \frac{dX(z)}{dz}$
Wartość początkowa		$x(0) = \lim_{z \rightarrow \infty} X(z)$
Wartość końcowa		$x(\infty) = \lim_{z \rightarrow 1} (z-1)X(z)$

Transformaty Z niektórych funkcji dyskretnych

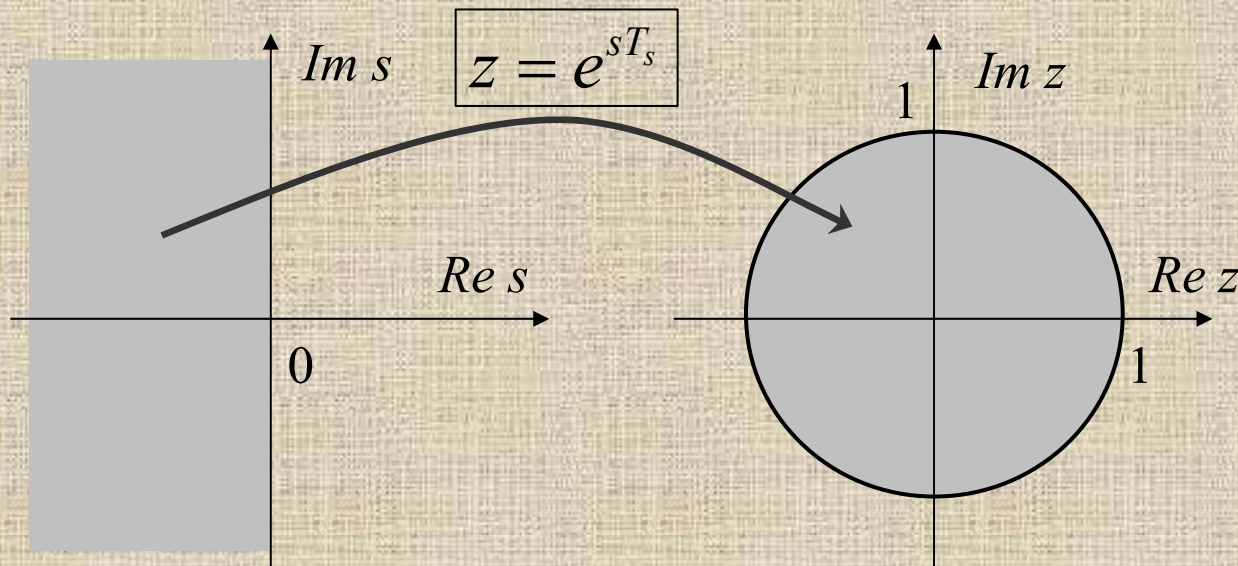
Funkcja dyskretna $x(n)$, $n \geq 0$	Transformata $X(z)$
$\delta(n)$	1
$1(n)$	$\frac{z}{z-1}$
$a \cdot n$	$\frac{az}{(z-1)^2}$
a^n	$\frac{z}{z-a}$
$n \cdot a^n$	$\frac{az}{(z-a)^2}$
$\sin(\Omega n)$	$\frac{z \sin \Omega}{z^2 - 2z \cos \Omega + 1}$
$\cos(\Omega n)$	$\frac{z(z - \cos \Omega)}{z^2 - 2z \cos \Omega + 1}$
$a^n \cos(\Omega n)$	$\frac{z(z - a \cos \Omega)}{z^2 - 2az \cos \Omega + a^2}$

Związek transformaty Z funkcji dyskretnej $x(n)=x(nT_s)$ z transformacją Laplace'a funkcji impulsowej $x^*(t)$ otrzymanej w wyniku próbkowania impulsowej funkcji ciągłej (analogowej) $x_a(t)$.

$$x^*(t) = \sum_{n=0}^{\infty} x_a(t) \delta(t - nT_s)$$

$$X(z) \Big|_{z=e^{sT_s}} = X^*(s)$$

gdzie: $X(z) = Z[x(nT_s)]$, $X^*(s) = L[x^*(t)]$ (L – transformata Laplace'a)



Rys. Przekształcenie lewej półpłaszczyzny i osi urojonej zmiennej zespolonej 's' na koło jednostkowe na płaszczyźnie zmiennej zespolonej 'z'

Sygnal ciągły: → próbkowanie → Sygnal dyskretny:

$$x_a(t) = e^{-\alpha t} \cos \omega t, \quad \alpha > 0$$

$$x(nT_s) = e^{-\alpha n T_s} \cos(n\omega T_s) = r^n \cos(n\Omega)$$

$$X_a(s) = \frac{s + \alpha}{(s + \alpha)^2 + \omega^2}$$

$$X(z) = \frac{z(z - r \cos \Omega)}{z^2 - 2r \cos \Omega \cdot z + r^2} \quad r = e^{-\alpha T_s}$$

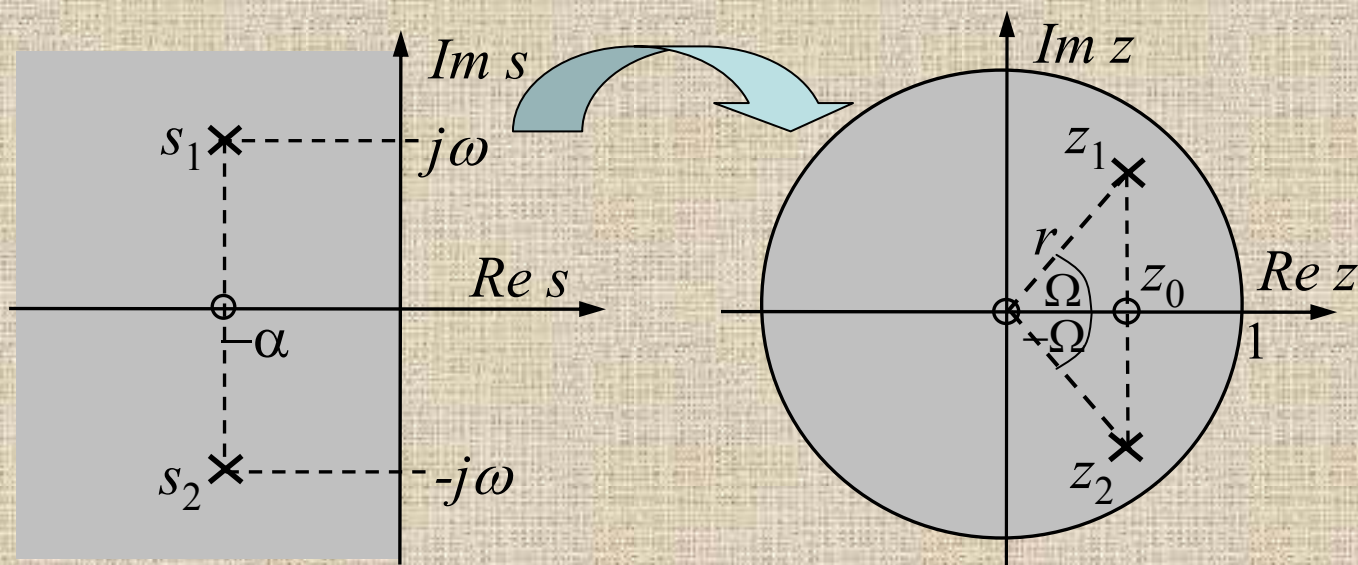
$$\Omega = \omega T_s$$

Bieguny: $s_{1,2} = -\alpha \pm j\omega$

$$z_{1,2} = r(\cos \Omega \pm j \sin \Omega) = r e^{\pm j\Omega}$$

$$z_{1,2} = e^{-\alpha T_s} e^{\pm j\omega T_s} = e^{(-\alpha \pm j\omega) T_s} = e^{s_{1,2} T_s}$$

Zera $X(z)$: $z_0 = r \cdot \cos \Omega$ i $z=0$ (nie ma ciągłego odpowiednika).



Rys. Bieguny transformaty ciągłej i ich odpowiedniki dla transformaty dyskretnej

Transformując równanie różnicowe wyraz po wyrazie dla WP=0 dostajemy:

$$a_0 y(n) + a_1 y(n-1) + \dots + a_N y(n-N) = b_0 u(n) + b_1 u(n-1) + \dots + b_N u(n-N)$$

$$a_0 Y(z) + a_1 z^{-1} Y(z) + \dots + a_N z^{-N} Y(z) = b_0 U(z) + b_1 z^{-1} U(z) + \dots + b_N z^{-N} U(z)$$

Transmitancja dyskretna – model układu liniowego jednowymiarowego (SISO):

$$H(z) = \left. \frac{Y(z)}{U(z)} \right|_{WP=0} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_N z^{-N}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_N z^{-N}} = \frac{B(z)}{A(z)}$$

Warunki początkowe:

$$y(-1) = y(-2) = \dots = y(-N) = 0, \quad u(-1) = u(-2) = \dots = u(-N) = 0$$

Równanie różnicowe i transmitancja z przesunięciem indeksów do przodu:

$$a_0 y(n+N) + a_1 y(n+N-1) + \dots + a_N y(n) = b_0 u(n+N) + b_1 u(n+N-1) + \dots + b_N u(n)$$

$$H(z) = \frac{b_0 z^N + b_1 z^{N-1} + b_2 z^{N-2} + \dots + b_N}{a_0 z^N + a_1 z^{N-1} + a_2 z^{N-2} + \dots + a_N}$$

Warunki początkowe: $y(0), y(1), \dots, y(N-1)$

Zera z_i i bieguny p_k (miejsca zerowe) zespolonej funkcji transmitancji:

$$H(z) = \frac{b_0(z - z_1)(z - z_2) \cdot \dots \cdot (z - z_N)}{a_0(z - p_1)(z - p_2) \cdot \dots \cdot (z - p_N)}$$

Układ liniowy opisany transmitancją $H(z)$ jest *stabilny* wtedy i tylko wtedy, kiedy *wszystkie bieguny jego transmitancji leżą wewnątrz okręgu jednostkowego*, tj.:

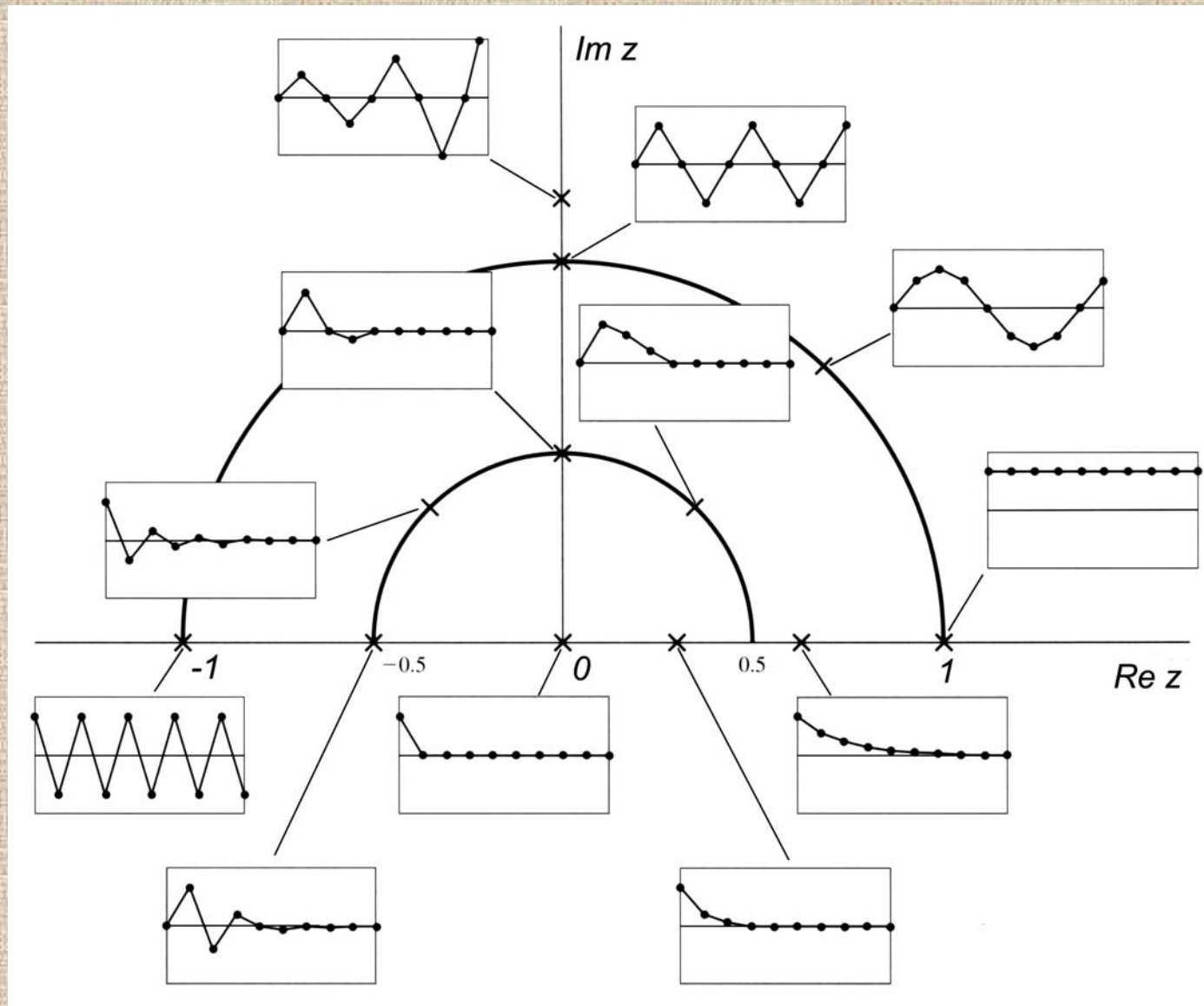
$$|p_i| < 1 \quad \text{dla } i = 1, 2, \dots, N$$

Z jednokrotnym biegunem rzeczywistym $p_i = a$ związana jest składowa odpowiedzi impulsowej postaci

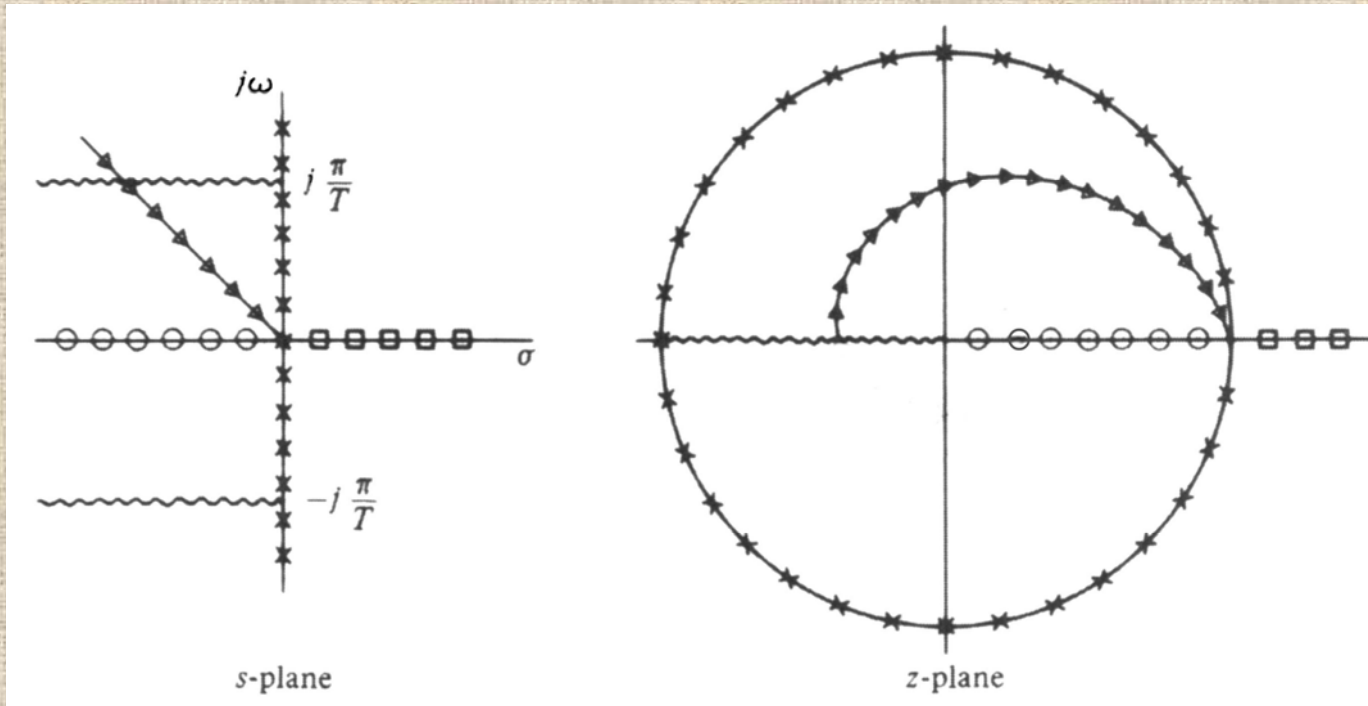
$$C_i \cdot a^n,$$

natomiast z parą biegunów zespolonych sprzężonych $p_{i,i+1} = |p_i| \cdot \exp(\pm j\Omega_i)$, składowa odpowiedzi impulsowej postaci

$$C_i \cdot |p_i|^n \cdot \cos(n\Omega_i + \varphi_0).$$



Rys. Mapa przebiegów czasowych (odpowiedzi impulsowej) związanych z położeniem bieguna transformaty Z na płaszczyźnie zespolonej.



Rys. Linie położeń biegunów odpowiadające tym samym właściwościom dynamicznym sygnałów (układów) ciągłych i dyskretnych.

Charakterystyka częstotliwościowa układu dyskretnego liniowego opisanego transmitancją $H(z)$:

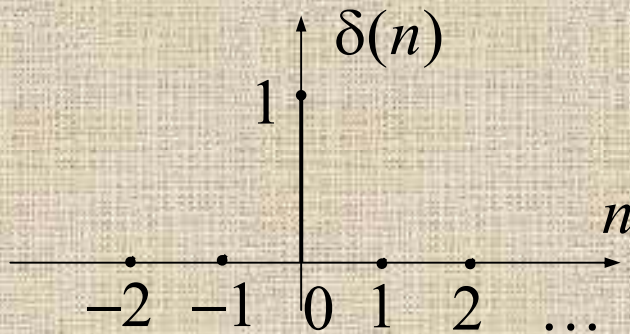
$$H(e^{j\Omega}) = H(z)|_{z=e^{j\Omega}} = |H(e^{j\Omega})| \cdot \exp[j \arg H(e^{j\Omega})]$$

Charakterystyka amplitudowa: $M(\Omega) = |H(e^{j\Omega})|$

Charakterystyka fazowa: $\varphi(\Omega) = \arg[H(e^{j\Omega})] = \angle H(e^{j\Omega})$

Układ liniowy stacjonarny w pełni charakteryzuje jego odpowiedź impulsowa $h(n)$. Jest to odpowiedź $y(n)$ na impuls $u(n)=\delta(n)$ ($\delta(n)=1$ dla $n=0$ i $\delta(n)=0$ dla $n\neq 0$). Dla dowolnego wymuszenia $u(n)$ wyjście jest *splotem*:

$$y(n) = h(n) * u(n) = \sum_{k=0}^{\infty} h(k)u(n-k) = \sum_{k=0}^{\infty} h(n-k)u(k)$$



Rys. Wymuszenie impulsowe,
odpowiedź $y(n) = h(n) \cdot \delta(n)$

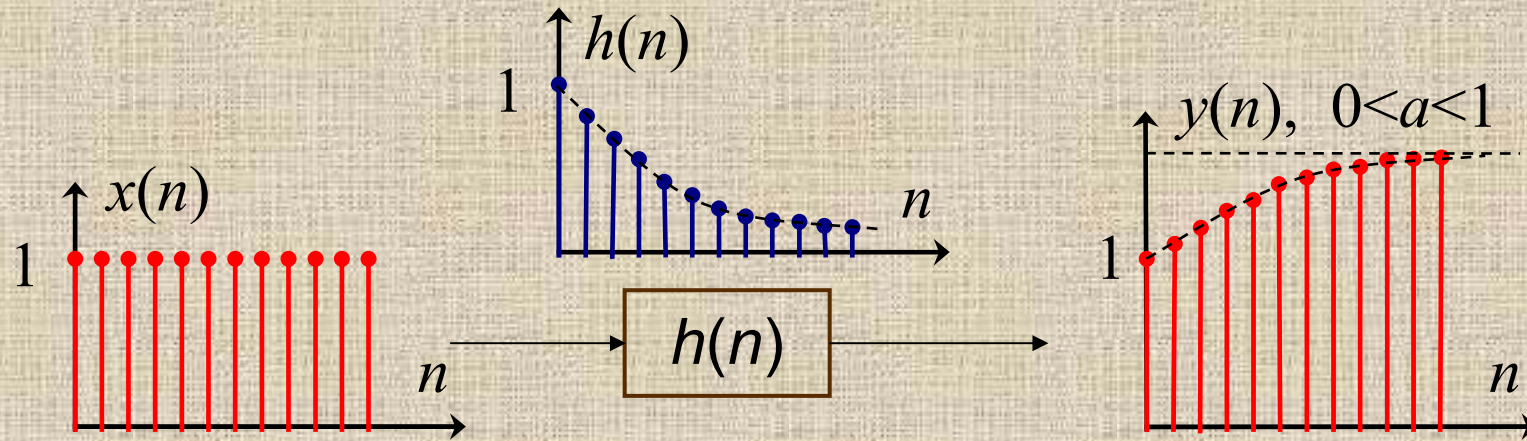
Transmitancja układu to transformata Z odpowiedzi impulsowej:

$$H(z) = Z[h(n)]$$

Układ liniowy jest stabilny w sensie BIBO (*bounded input – bounded output*), jeżeli jego odpowiedź impulsowa spełnia warunek:

$$\sum_{k=-\infty}^{\infty} |h(k)| < \infty$$

Przykład: $h(n)=a^n$, $|a|<1$, wymuszenie $x(n)=\mathbf{1}(n)$



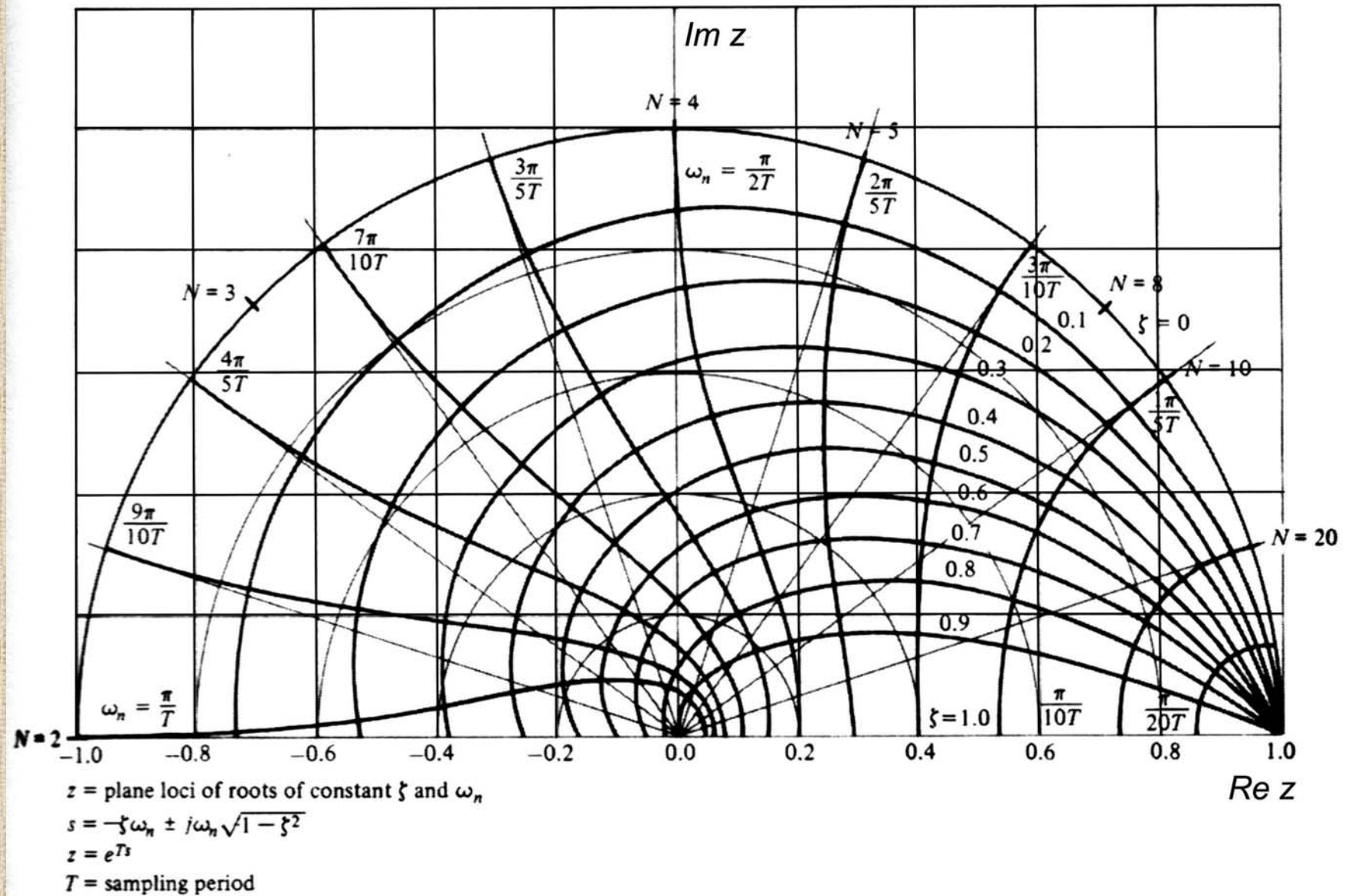
$$y(n) = h(n) * x(n) = \sum_{k=0}^{\infty} h(k) \cdot \mathbf{1}(n-k) = \sum_{k=0}^n h(k) = \sum_{k=0}^n a^k,$$

$$y(0) = 1, \quad y(1) = 1 + a, \quad y(2) = 1 + a + a^2, \dots, \lim_{n \rightarrow \infty} y(n) = \frac{1}{1-a}$$

W dziedzinie transformat Z:

$$H(z) = Z[a^n] = \frac{z}{z-a}, \quad X(z) = Z[\mathbf{1}(n)] = \frac{z}{z-1}$$

$$Y(z) = H(z) \cdot X(z) = \frac{z^2}{(z-1)(z-a)} = \frac{1}{1-a} \left(\frac{z}{z-1} - \frac{az}{z-a} \right) \rightarrow y(n) = \frac{1-a^{n+1}}{1-a}, \quad n \geq 0$$



Rys. Linie stałego współczynnika tłumienia ζ i pulsacji drgań nietłumionych ω_n okręgu jednostkowym na płaszczyźnie zmiennej zespolonej z

Przykład: Wyznaczyć charakterystykę częstotliwościową układu dyskretnego o transmitancji $H(z)=\frac{1}{2}+\frac{1}{2}z^{-1}$.

Odp.: $M(\Omega)=|\cos(\Omega/2)|$, $\varphi(\Omega)=-\Omega/2$

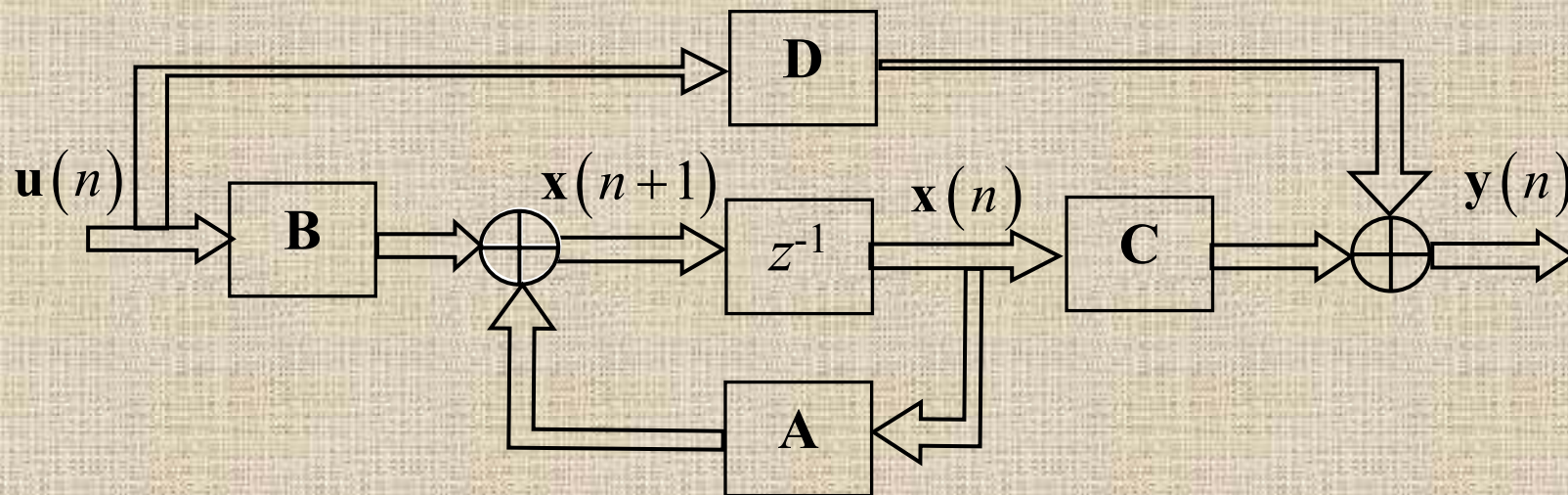
Model układu dyskretnego (wielowymiarowego - MIMO) w formie równań stanu:

$$\mathbf{x}(n+1) = \mathbf{A}\mathbf{x}(n) + \mathbf{B}\mathbf{u}(n)$$

$$\mathbf{y}(n) = \mathbf{C}\mathbf{x}(n) + \mathbf{D}\mathbf{u}(n)$$

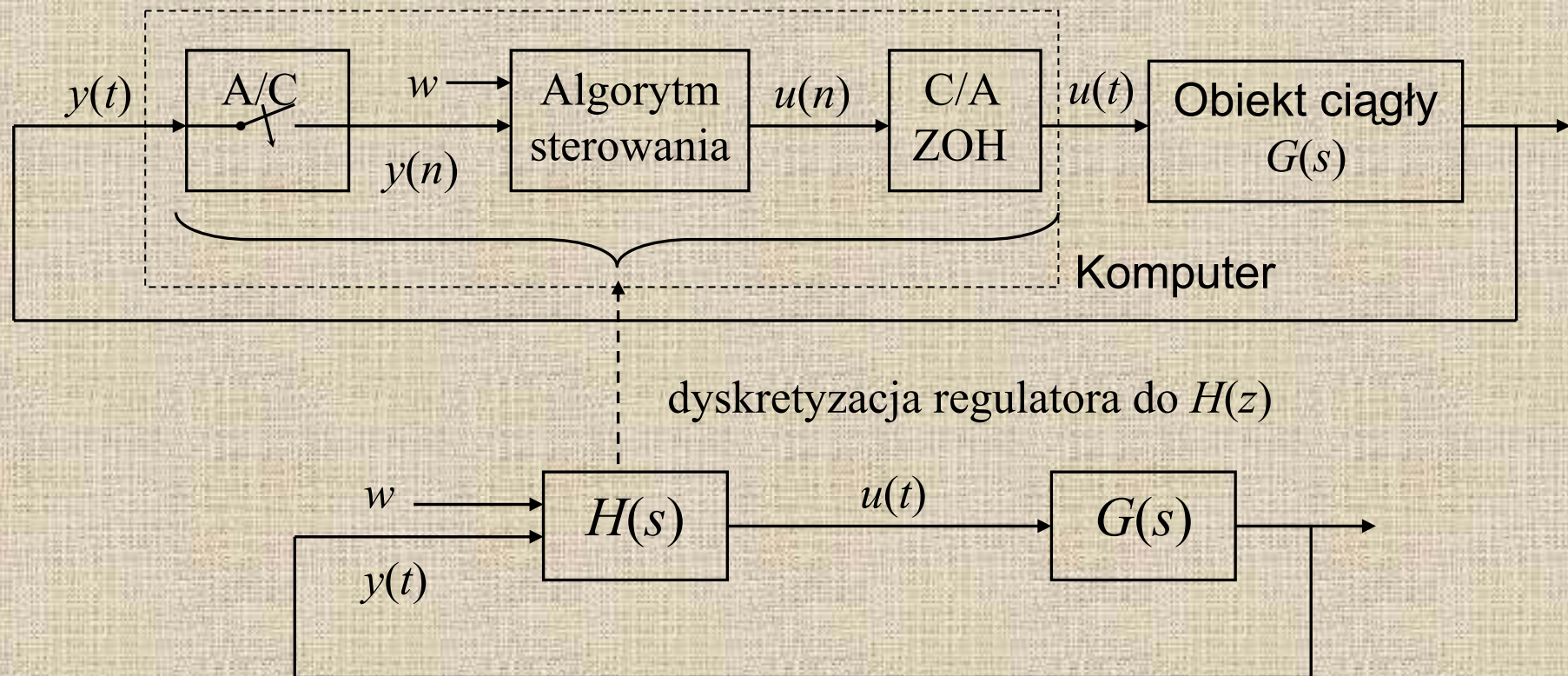
$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

$\mathbf{x}(n)$ – wektor zmiennych stanu, $\mathbf{u}(n)$ – wektor wejść, $\mathbf{y}(n)$ – wektor wyjść



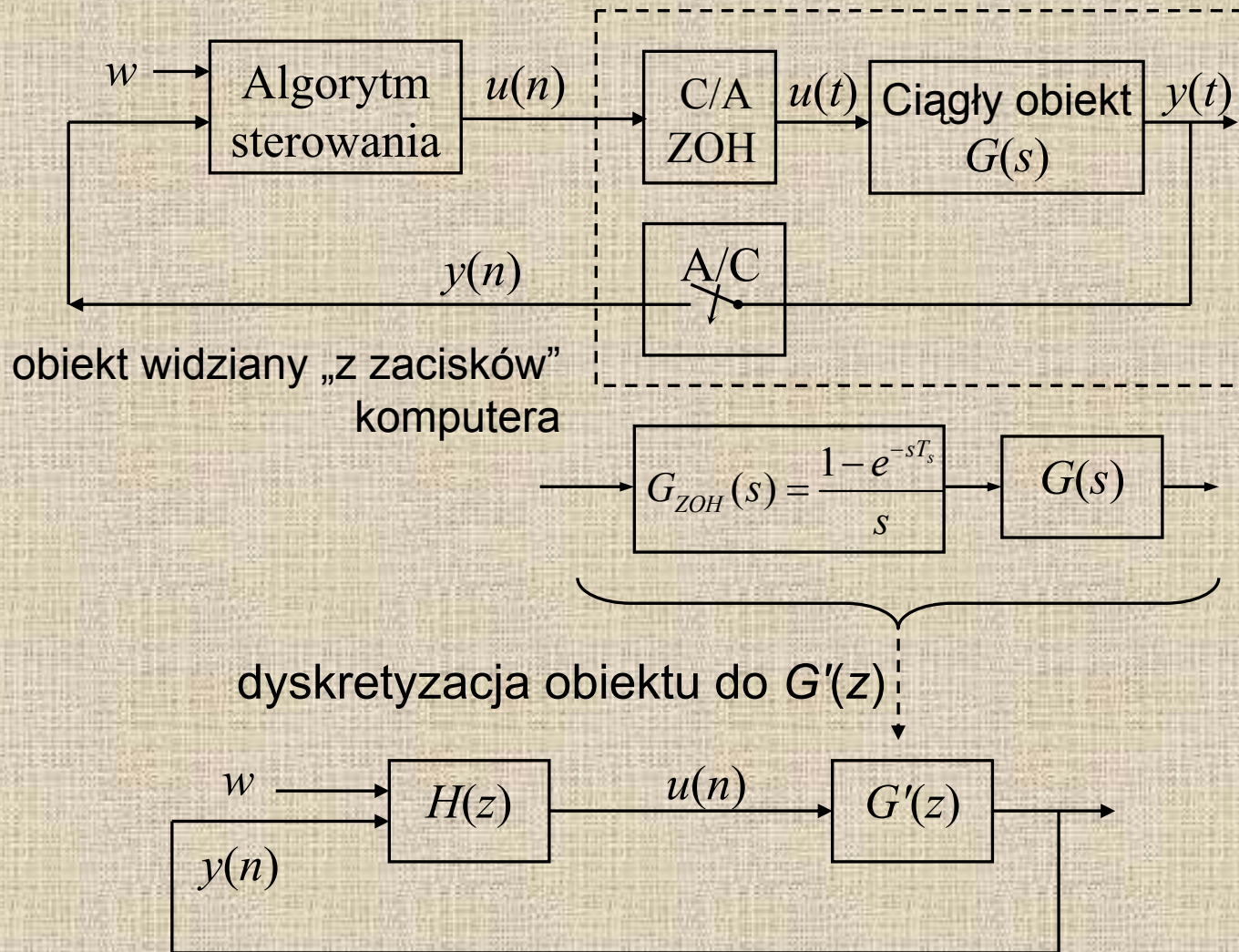
Metodologia projektowania regulacji dyskretnej

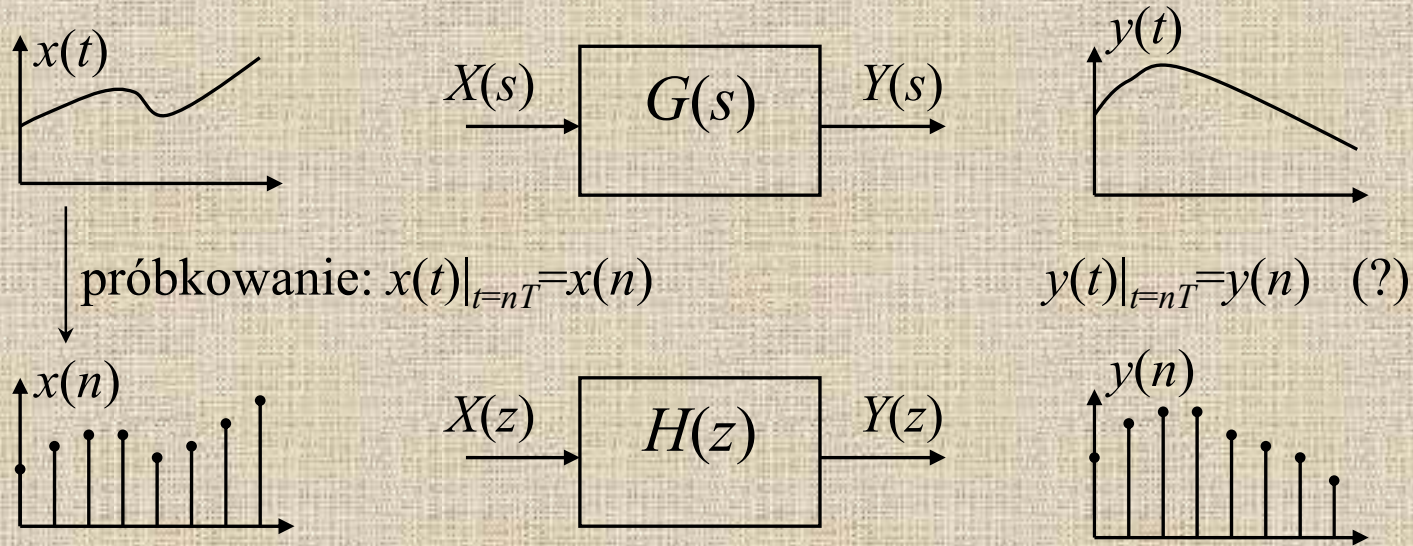
Podjęcie 1



Rys. Projektowanie regulacji ciągłej - emulacja dyskretna regulacji analogowej

Podejście 2

Rys. Bezpośrednie projektowanie regulacji dyskretnej $H(z)$



Rys. Równoważność modelu ciągłego i dyskretnego – zależy od $x(t)$!

Istnieją różne możliwości przybliżenia równania różniczkowego za pomocą równania różnicowego. Jedną z możliwości jest wprowadzenie różnic skończonych.

$$\dot{x}(t) = \frac{dx}{dt} \leftrightarrow \Delta x(n) = \frac{x(n+1) - x(n)}{T_s}$$

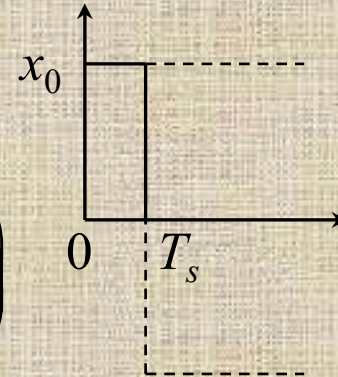
$$\ddot{x}(t) = \frac{d^2x}{dt^2} \leftrightarrow \Delta^2 x(n) = \frac{\Delta x(n+1) - \Delta x(n)}{T_s} = \frac{x(n+2) - 2x(n+1) + x(n)}{T_s^2}$$

Metody dyskretyzacji modeli ciągłych układów liniowych

Metoda niezmienniczej odpowiedzi na wymuszenie schodkowe (Zero Order Hold – ZOH)

Odpowiedź na pojedynczy schodek wymuszenia
(próbka x_0):

$$Y_0(s) = G(s) \cdot x_0 \cdot L[\mathbf{1}(n) - \mathbf{1}(n-1)] = G(s) \cdot x_0 \left(\frac{1}{s} - \frac{1}{s} e^{-sT_s} \right)$$



Odpowiedź na superpozycję wymuszeń w postaci sumy schodków przesuniętych w czasie o amplitudach równych wartościom wymuszenia ciągłego w momentach próbkowania:

$$\underbrace{Y_0(s) + Y_1(s) + Y_3(s) + \dots}_{Y(s)} = G(s) \cdot \frac{1 - e^{-sT_s}}{s} \cdot \underbrace{(x_0 + x_1 e^{-sT_s} + x_2 e^{-2sT_s} + \dots)}_{\text{suma wszystkich schodków}}$$

Po wstawieniu $e^{-ksT_s} = z^{-k}$ otrzymujemy:

$$Y(z) = (1 - z^{-1}) \cdot Z \left[\frac{G(s)}{s} \right] \cdot X(z)$$

$$H_{ZOH}(z) = \frac{z-1}{z} \cdot Z \left[\frac{G(s)}{s} \right]$$

Metodę ZOH stosuje się do dyskretyzacji modeli obiektów sterowanych sygnałem schodkowym z przetwornika C/A. W chwilach próbkowania sygnały wejściowe i wyjściowe $G(s)$ i $H_{ZOH}(z)$ są takie same.

Przykład: Stosując metodę ZOH wyznaczyć dyskretny odpowiednik transmitancji ciągłej $G(s)=a/(s+a)$ dla okresu próbkowania T_s .

Odp.:

$$H_{ZOH}(z) = \frac{1-p}{z-p}$$

biegun $p = e^{-aT_s}$

Równanie różnicowe: $y(n) - py(n-1) = (1-p)x(n-1)$

$$y(n) = py(n-1) + (1-p)x(n-1)$$

Laplace Transforms and z-transforms of Simple Discrete Time Functions

$F(s)$ is the Laplace transform of $f(t)$, and $F(z)$ is the z-transform of $f(kT)$. Note: $f(t) = 0$ for $t = 0$.

Number	$\mathcal{F}(s)$	$f(kT)$	$F(z)$
1		$1, k = 0; 0, k \neq 0$	1
2		$1, k = k_0; 0, k \neq k_0$	z^{-k_0}
3	$\frac{1}{s}$	$1(kT)$	$\frac{z}{z-1}$
4	$\frac{1}{s^2}$	kT	$\frac{Tz}{(z-1)^2}$
5	$\frac{1}{s^3}$	$\frac{1}{2!}(kT)^2$	$\frac{T^2}{2} \left[\frac{z(z+1)}{(z-1)^3} \right]$
6	$\frac{1}{s^4}$	$\frac{1}{3!}(kT)^3$	$\frac{T^3}{6} \left[\frac{z(z^2+4z+1)}{(z-1)^4} \right]$
7	$\frac{1}{s^m}$	$\lim_{a \rightarrow 0} \frac{(-1)^{m-1}}{(m-1)!} \left(\frac{\partial^{m-1}}{\partial a^{m-1}} e^{-akt} \right)$	$\lim_{a \rightarrow 0} \frac{(-1)^{m-1}}{(m-1)!} \left(\frac{\partial^{m-1}}{\partial a^{m-1}} \frac{z}{z - e^{-at}} \right)$
8	$\frac{1}{s+a}$	e^{-akt}	$\frac{z}{z - e^{-aT}}$
9	$\frac{1}{(s+a)^2}$	$kT e^{-akt}$	$\frac{Tz e^{-aT}}{(z - e^{-aT})^2}$
10	$\frac{1}{(s+a)^3}$	$\frac{1}{2} (kT)^2 e^{-akt}$	$\frac{T^2}{2} e^{-aT} \frac{z(z+e^{-aT})}{(z - e^{-aT})^3}$
11	$\frac{1}{(s+a)^m}$	$\frac{(-1)^{m-1}}{(m-1)!} \left(\frac{\partial^{m-1}}{\partial a^{m-1}} e^{-akt} \right)$	$\frac{(-1)^{m-1}}{(m-1)!} \left(\frac{\partial^{m-1}}{\partial a^{m-1}} \frac{z}{z - e^{-aT}} \right)$
12	$\frac{a}{s(s+a)}$	$1 - e^{-akt}$	$\frac{z(1 - e^{-aT})}{(z-1)(z - e^{-aT})}$
13	$\frac{a}{s^2(s+a)}$	$\frac{1}{a} (akT - 1 + e^{-akt})$	$\frac{z[(aT - 1 + e^{-aT})z + (1 - e^{-aT} - aTe^{-aT})]}{a(z-1)^2(z - e^{-aT})}$
14	$\frac{b-a}{(s+a)(s+b)}$	$e^{-akt} - e^{-bkt}$	$\frac{(e^{-aT} - e^{-bT})z}{(z - e^{-aT})(z - e^{-bT})}$
15	$\frac{s}{(s+a)^2}$	$(1 - akT)e^{-akt}$	$\frac{z[z - e^{-aT}(1 + aT)]}{(z - e^{-aT})^2}$
16	$\frac{a^2}{s(s+a)^2}$	$1 - e^{-akt}(1 + akT)$	$\frac{z[z(1 - e^{-aT} - aTe^{-aT}) + e^{-2aT} - e^{-aT} + aTe^{-aT}]}{(z-1)(z - e^{-aT})^2}$
17	$\frac{(b-a)s}{(s+a)(s+b)}$	$be^{-bkt} - ae^{-akt}$	$\frac{z[z(b-a) - (be^{-aT} - ae^{-bT})]}{(z - e^{-aT})(z - e^{-bT})}$
18	$\frac{a}{s^2 + a^2}$	$\sin akT$	$\frac{z \sin aT}{z^2 - (2 \cos aT)z + 1}$
19	$\frac{s}{s^2 + a^2}$	$\cos akT$	$\frac{z(z - \cos aT)}{z^2 - (2 \cos aT)z + 1}$
20	$\frac{s+a}{(s+a)^2 + b^2}$	$e^{-akt} \cos bkt$	$\frac{z(z - e^{-aT} \cos bT)}{z^2 - 2e^{-aT}(\cos bT)z + e^{-2aT}}$
21	$\frac{b}{(s+a)^2 + b^2}$	$e^{-akt} \sin bkt$	$\frac{ze^{-aT} \sin bT}{z^2 - 2e^{-aT}(\cos bT)z + e^{-2aT}}$
22	$\frac{a^2 + b^2}{s[(s+a)^2 + b^2]}$	$1 - e^{-akt} \left(\cos bkt + \frac{a}{b} \sin bkt \right)$	$\frac{z(Az + B)}{(z-1)[z^2 - 2e^{-aT}(\cos bT)z + e^{-2aT}]}$ $A = 1 - e^{-aT} \cos bT - \frac{a}{b} e^{-aT} \sin bT$ $B = e^{-2aT} + \frac{a}{b} e^{-aT} \sin bT - e^{-aT} \cos bT$

Dyskretne aproksymacje wynikające z metod całkowania numerycznego

Są to metody przybliżone, które wynikają z różnych aproksymacji pochodnej w równaniu różniczkowym:

$$a_n \frac{d^n y}{dt^n} + a_{n-1} \frac{d^{n-1} y}{dt^{n-1}} + \dots + a_1 \frac{dy}{dt} + a_0 y(t) = b_m \frac{d^m u}{dt^m} + b_{m-1} \frac{d^{m-1} u}{dt^{m-1}} + \dots + b_1 \frac{du}{dt} + b_0 u(t)$$

- metoda różnicowa Eulera (różnic do przodu, **forward Euler**):

$$\left. \frac{dx}{dt} \right|_{t_n} \cong \frac{x(n+1) - x(n)}{T_s} \rightarrow x(n+1) = x(n) + T_s \cdot \left. \frac{dx}{dt} \right|_{t_n}, \quad sX = \frac{zX - X}{T_s}$$

$$G_{FE}(z) = G(s) \Big|_{s = \frac{z-1}{T_s}}$$

- metoda różnic wstecznych (**backward Euler**):

$$\left. \frac{dx}{dt} \right|_{t_n} \cong \frac{x(n) - x(n-1)}{T_s} \rightarrow x(n+1) = x(n) + T_s \cdot \left. \frac{dx}{dt} \right|_{t_{n+1}}, \quad sX = \frac{X - z^{-1}X}{T_s} = \frac{zX - X}{zT_s}$$

$$G_{BE}(z) = G(s) \Big|_{s = \frac{z-1}{zT_s}}$$

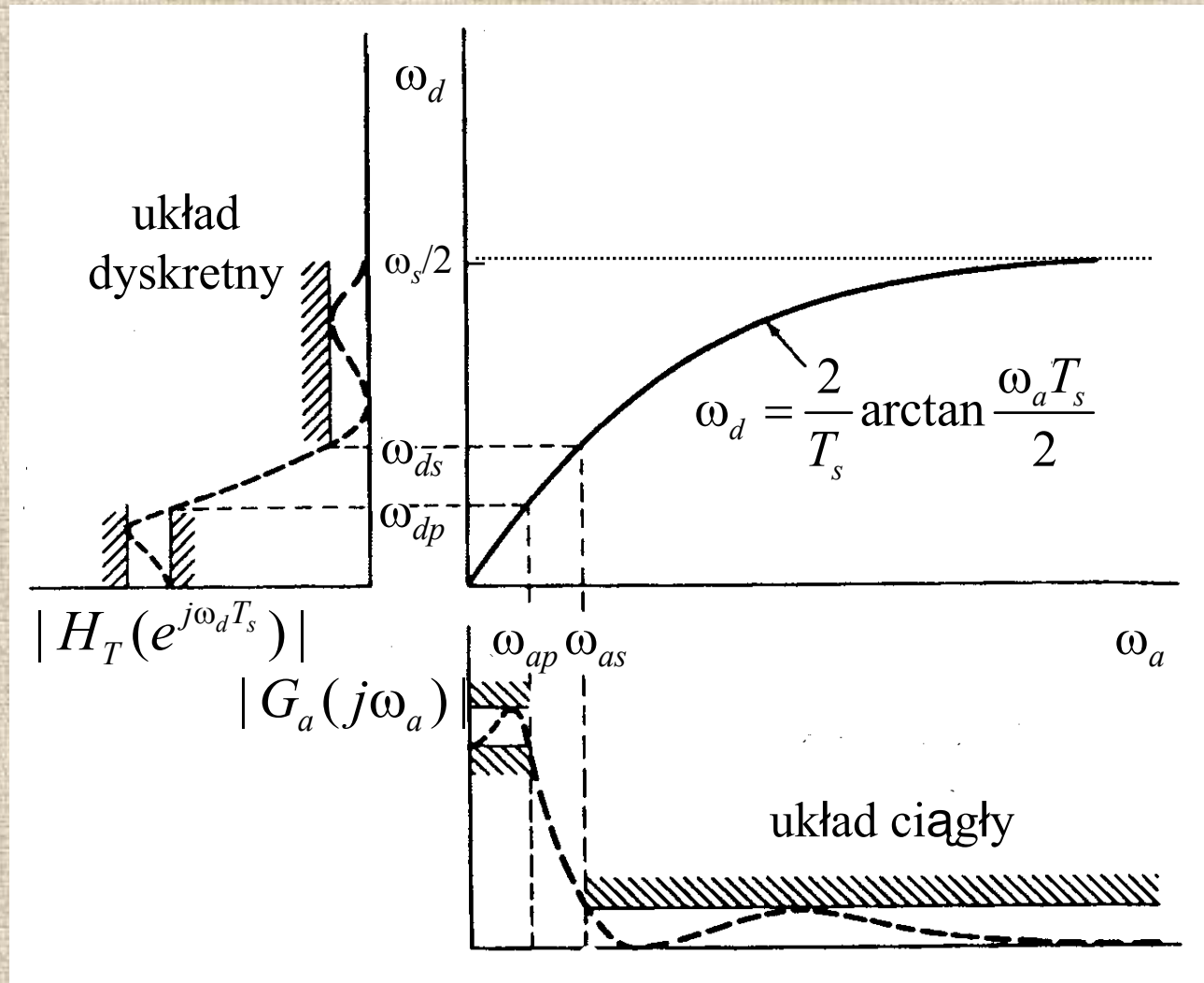
- metoda trapezów (Tustina, odwzorowania biliniowego):

$$x(n+1) = x(n) + \frac{T_s}{2} \cdot \left(\left. \frac{dx}{dt} \right|_{t_{n+1}} + \left. \frac{dx}{dt} \right|_{t_n} \right)$$

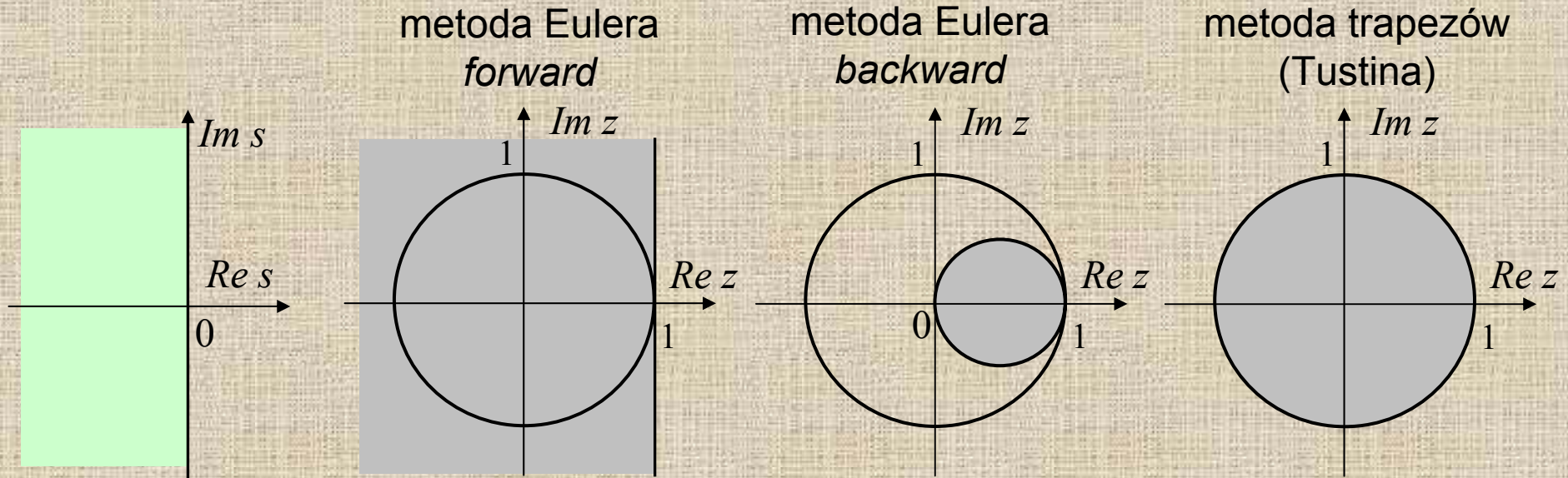
$$G_T(z) = G(s) \Big|_{s = \frac{2}{T_s} \cdot \frac{z-1}{z+1}}$$

Metoda odwzorowania biliniowego charakteryzuje się nieliniowym przekształceniem skali częstotliwości analogowej na skalę częstotliwości dyskretnej:

$$\omega_d = \frac{2}{T_s} \cdot \operatorname{arctg} \frac{\omega_a T_s}{2}$$



Rys. Wpływ nieliniowego przekształcenia skali częstotliwości metody trapezów na różnicę przebiegu charakterystyk częstotliwościowych układu ciągłego i jego dyskretnego odpowiednika



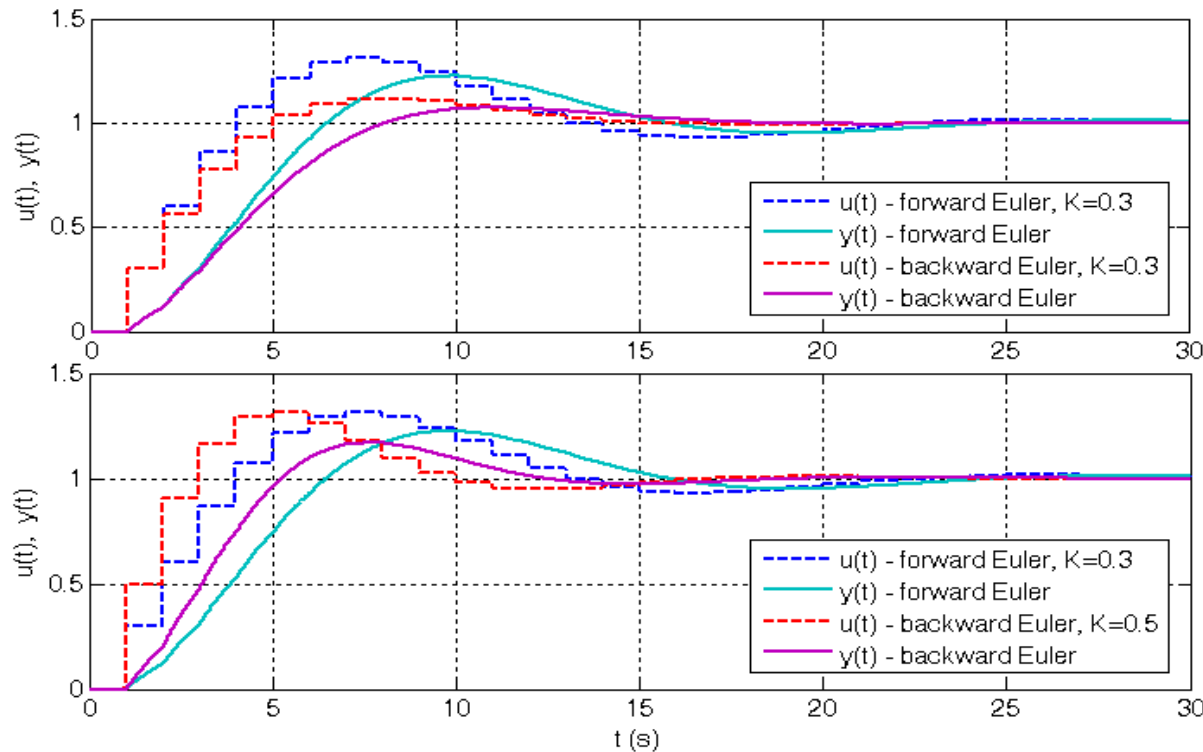
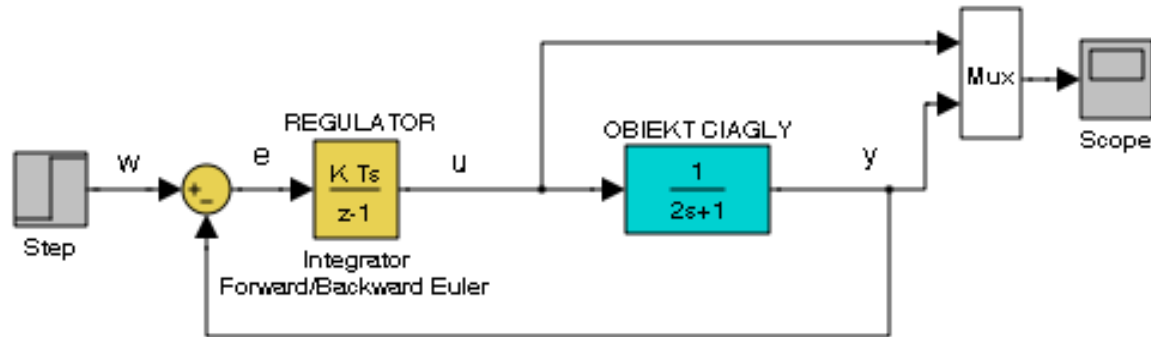
Rys. Odwzorowanie lewej półpłaszczyzny 's' na płaszczyznę 'z' dla dyskretyzacji wynikających z metod całkowania numerycznego (problem zmiany właściwości dynamicznych po dyskretyzacji)

Przykład: Stosując przedstawione metody wyznaczyć dyskretne odpowiedniki transmitancji ciągłej $G(s)=a/(s+a)$ dla okresu próbkowania T_s .

Odp.:

$$H_{FE}(z) = \frac{1-p}{z-p} \quad H_{BE}(z) = \frac{z(1-p)}{z-p} \quad H_T(z) = \frac{(1-p)(z+1)}{2(z-p)}, \quad p = \frac{2-aT_s}{2+aT_s}$$

$$p = 1-aT_s \quad p = \frac{1}{1+aT_s}$$



Rys. Porównanie działania regulatora I z całkowaniem metodą Eulera *forward* i *backward*; we wszystkich przypadkach $T_s=1s$

Równanie integratora *forward Euler*: $u(n) = u(n-1) + KT_s e(n-1)$

Równanie integratora *backward Euler*: $u(n) = u(n-1) + KT_s e(n)$

Metoda odwzorowania zer i biegunów (zero-pole matching)

Metoda ta jest rozszerzeniem odwzorowania $z = e^{sT_s}$ biegunów transformat sygnału dyskretyzowanego i ciągłego pierwowzoru na transmitancje. W taki sam sposób odwzorowuje się też skończone zera transmitancji.

1. Wszystkie bieguny $G(s)$ są odwzorowywane na bieguny $z = e^{sT_s}$ transmitancji $H(z)$.
2. Wszystkie skończone zera $G(s)$ są odwzorowywane na zera $z = e^{sT_s}$ transmitancji $H(z)$ w taki sam sposób.
3. Zera $G(s)$ leżące w nieskończoności $s=\infty$ (ich liczba odpowiada nadmiarowi biegunów $r=n-m$ w $G(s)$) są odwzorowywane na zera $z=-1$ transmitancji $H(z)$ – w liczniku dopisuje się czynnik $(z+1)^r$ zrównujący stopnie licznika i mianownika $H(z)$.
- 3a. Metoda zmodyfikowana: w liczniku $H(z)$ dopisuje się czynnik $(z+1)^{r-1}$ pozostawiając stopień licznika o 1 mniejszy od stopnia mianownika.
4. Współczynnik wzmocnienia $H(z)$ wybiera się w taki sposób, aby wzmocnienie $H(z)$ odpowiadało wzmocnieniu $G(s)$ dla określonej częstotliwości, zwykle wzmocnieniu DC:

$$G(s) \Big|_{s=0} = H_{ZP}(z) \Big|_{z=1}$$

Przykład: Wyznaczyć dyskretne odpowiedniki transmitancji ciągłej $G(s)=a/(s+a)$ metodą odwzorowania zer i biegunów.

Odp.:

$$H_{ZP}(z) = \frac{(1-p)(z+1)}{2(z-p)}, \quad p = e^{-aT_s}$$

$$y(n) = py(n-1) + (1-p) \frac{x(n) + x(n-1)}{2}$$

$$H_{ZPM}(z) = \frac{1-p}{z-p}$$

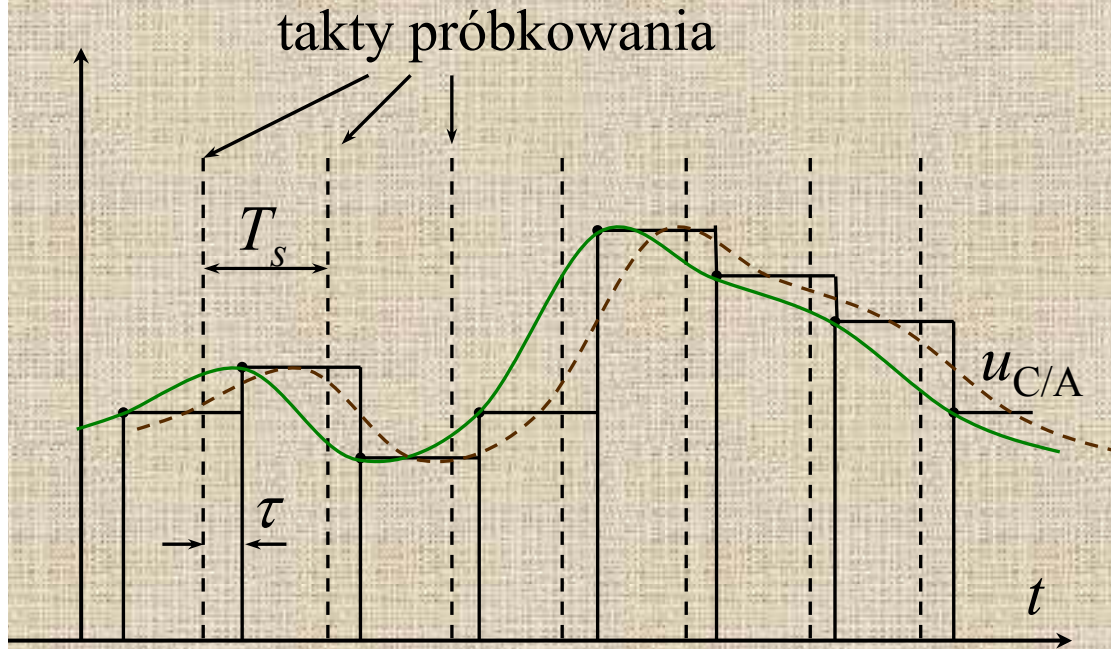
$$y(n) = py(n-1) + (1-p)x(n-1)$$

Opóźnienie ZOH i opóźnienie ułamkowe

Przedstawione metody aproksymacji dyskretnej są poprzedzone projektowaniem regulacji w formie ciągłej $G_r(s)$, bez uwzględnienia opóźnienia (średnio $\frac{1}{2}T_s$) wnoszonego przez uciąglenie ZOH sygnału sterującego $u(t)$. Opóźnienie to można w przybliżeniu uwzględnić dodając przed transmitancją obiektu $G_p(s)$ szeregową transmitancję

$$G(s) = \frac{1}{\frac{T_s}{2}s + 1} \quad \text{lub} \quad G_{Pade}(s) = \frac{1 - s\frac{T_s}{4}}{1 + s\frac{T_s}{4}} \approx e^{-s\frac{T_s}{2}}$$

Pominięcie wpływu opóźnienia ZOH nie ma znaczenia tylko przy szybkim próbkowaniu, tj. $\omega_s > 10\omega_B$ (pasmo przenoszenia układu zamkniętego). Nieuwzględnienie opóźnienia ZOH może być przyczyną pogorszenia tłumienia a nawet niestabilności układu zamkniętego (dla $\omega_s \leq 5\omega_B$).



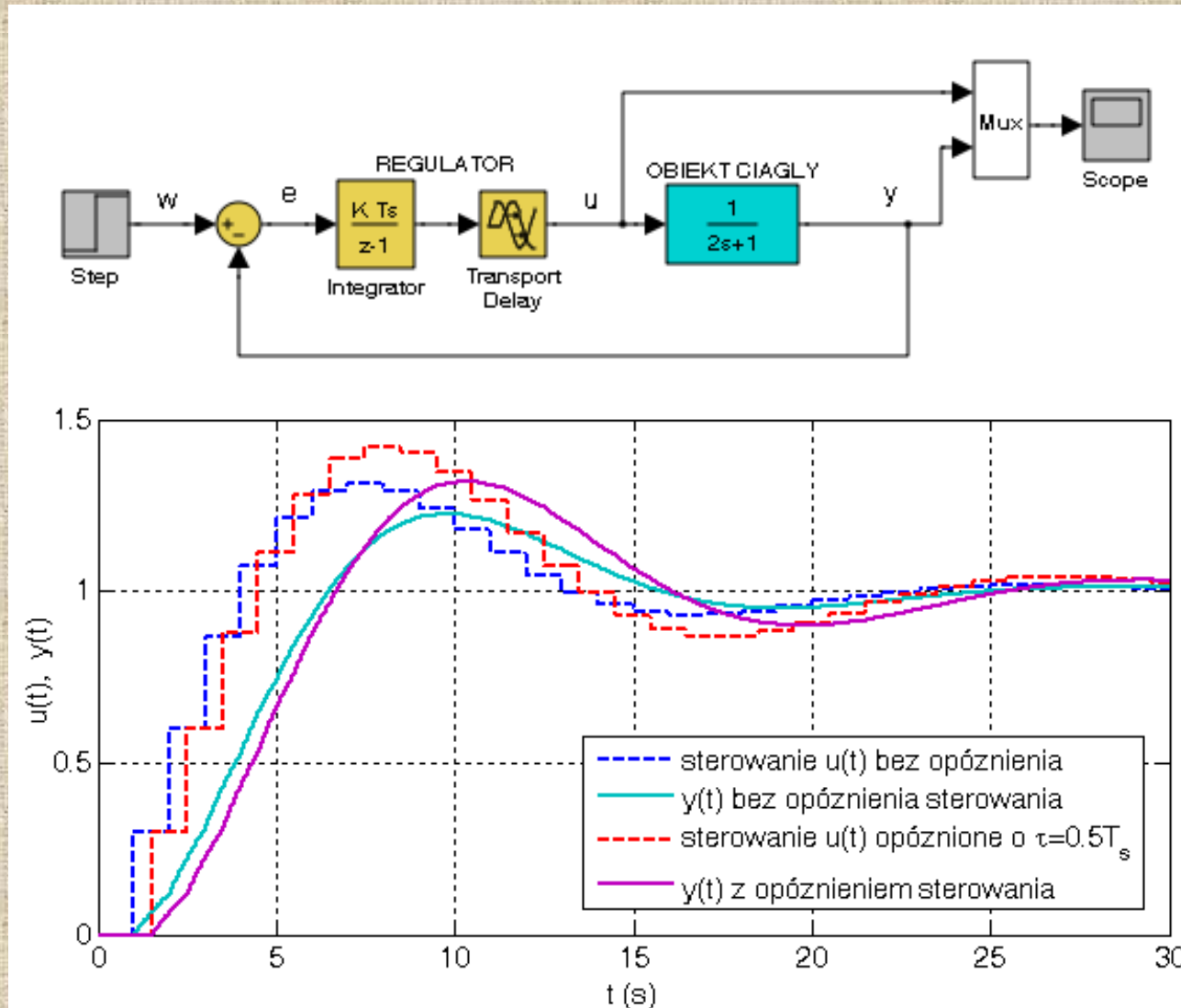
Rys. Opóźnienie ułamkowe τ chwil generowania sygnału sterującego (na wyjściu C/A) względem chwil próbkowania i opóźnienie wnoszone przez ekstrapolację zerowego rzędu ZOH (zero order hold) względem hipotetycznego sygnału ciągłego, równe $\frac{1}{2}T_s$.

Opóźnienie ułamkowe generowania sygnału sterującego względem taktów wyznaczających momenty próbkowania wynika z czasu przejścia sygnału przez tor przetwarzania:

$$\tau = t_{A/C} + t_{obl} + t_{C/A}$$

Jeżeli nieuwzględniane w algorytmie regulacji opóźnienie związane z t_{obl} jest zbyt duże, to można zaprojektować algorytm na sterowanie $u(n)$ nie wykorzystujący ostatniej próbki sygnału wejściowego $e(n)$. Sterowanie zależy wtedy tylko od próbek wcześniejszych $e(n-1)$, $e(n-2)$ itd., co oznacza, że $u(n)$ można wyliczyć przed taktem n -tym, a po jego nadejściu wysłać wynik na C/A.

Opóźnienie ułamkowe ma zwykle niewielkie znaczenie, jeśli $\tau < 0.3T_s$. Przy $\tau \geq 0.5T_s$ opóźnienie może spowodować znaczne pogorszenie lub nawet utratę stabilności układu regulacji.



Rys. Schemat układu symulacji i przebiegi czasowe sygnału sterującego $u(t)$ i wielkości sterowanej $y(t)$ bez opóźnienia (przypadek idealny) i z ułamkowym opóźnieniem τ generowania sterowania;
 $T_s=1s, K=0.3$

Zalecenia co do wyboru okresu próbkowania

Stosuje się zwykle następujące wytyczne co do wyboru częstotliwości próbkowania projektowanego układu regulacji:

– $\omega_s = 10 \div 20 \omega_B$ (ω_B - pasmo przenoszenia układu zamkniętego, $\omega_B = 1 \div 2 \omega_c$ (zależnie od ζ) – pulsacja odcięcia układu otwartego)

lub

- $T_s = 0.1 \div 0.25 t_n$ (tzn. 4 do 10 próbek na czas narastania odpowiedzi skokowej projektowanego układu zamkniętego)

Cyfrowa regulacja PID

Równanie analogowego regulatora PID:

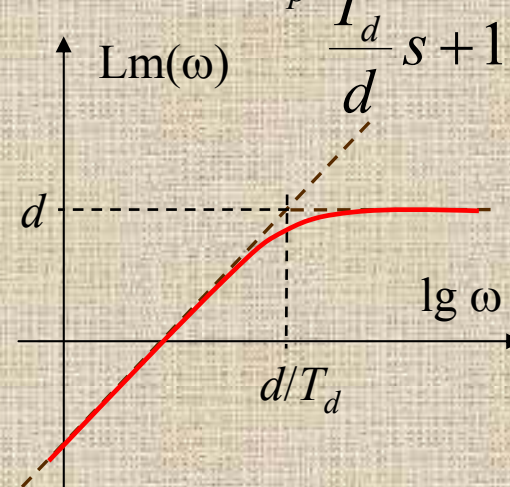
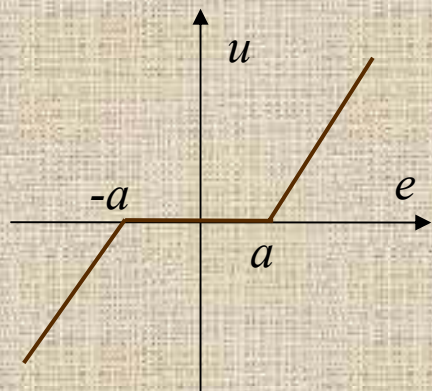
$$u(t) = K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de}{dt} \right)$$

gdzie: $e(t) = w(t) - y(t)$ - błąd regulacji

Transmitancja ciągła: $G_{PID}(s) = \frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i s} + T_d s \right)$

Żeby uniknąć różniczkowania skokowych zmian $e(t)$ przy zmianie wartości zadanej, pochodną oblicza się często tylko z $-y(t)$.

Część różniczkująca z inercją (rzeczywista): $D(s) = K_p \frac{T_d s}{\frac{T_d}{d} s + 1} \cdot E(s)$



Rys. Strefa martwa regulatora (lub części P)

Dyskretyzacja algorytmu PID:

- Część proporcjonalna: $P(n) = K_p e(n)$

- Część całkująca: $I(n) = I(n-1) + \Delta I(n)$ (równanie rekurencyjne)

- w metodzie Eulera: $\Delta I(n) = \frac{K_p T_s}{T_i} e(n-1)$

- w metodzie trapezów: $\Delta I(n) = \frac{K_p T_s}{T_i} \cdot \frac{e(n-1) + e(n)}{2}$

- Część różniczkująca (z inercją):

- w metodzie różnic wstecznych (aproksymacja: $\left. \frac{de}{dt} \right|_{t=nT_s} \cong \frac{e(n) - e(n-1)}{T_s}$)

$$D(n) = \frac{T_d}{T_d + T_s d} D(n-1) + \frac{K_p T_d d}{T_d + T_s d} \underbrace{[e(n) - e(n-1)]}_{\Delta D(n)}$$

- z tzw. czteropunktową różnicą centralną:

$$\Delta D(n) = \frac{1}{6} [e(n) + 3e(n-1) - 3e(n-2) - e(n-3)]$$

Całkowity sygnał sterujący – algorytm pozycyjny (absolutny):

$$u(n) = P(n) + I(n) + D(n)$$

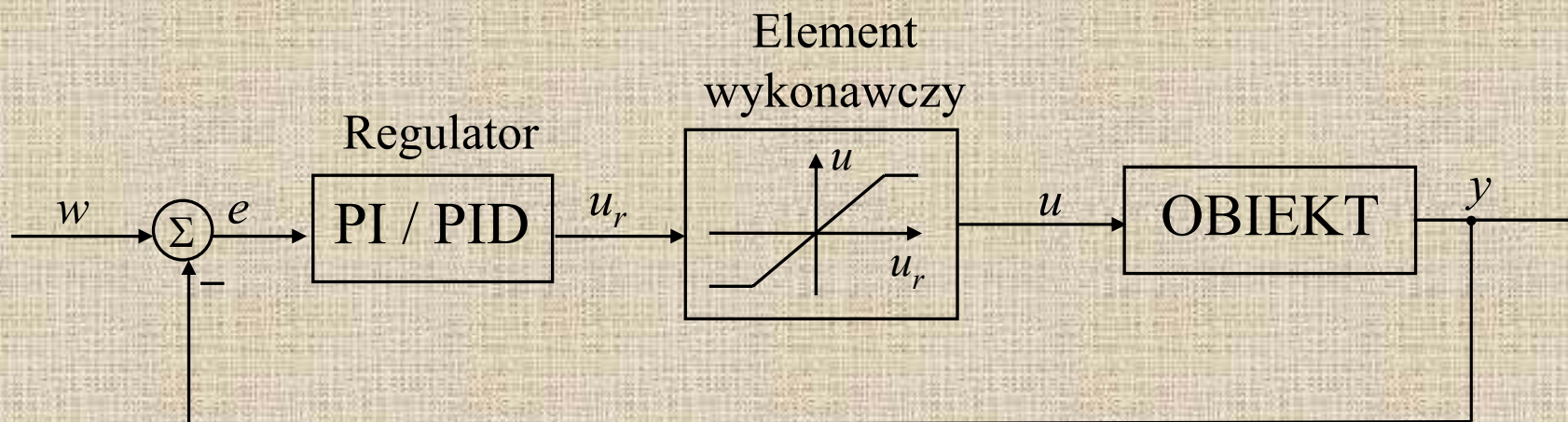
W algorytmie prędkościowym (przyrostowym), stosowanym dla regulacji PI z obiektem zawierającym element całkujący (np. zawór), na wyjście regulatora wyprowadza się *przyrost* sygnału sterującego:

$$\Delta u(n) = u(n) - u(n-1)$$

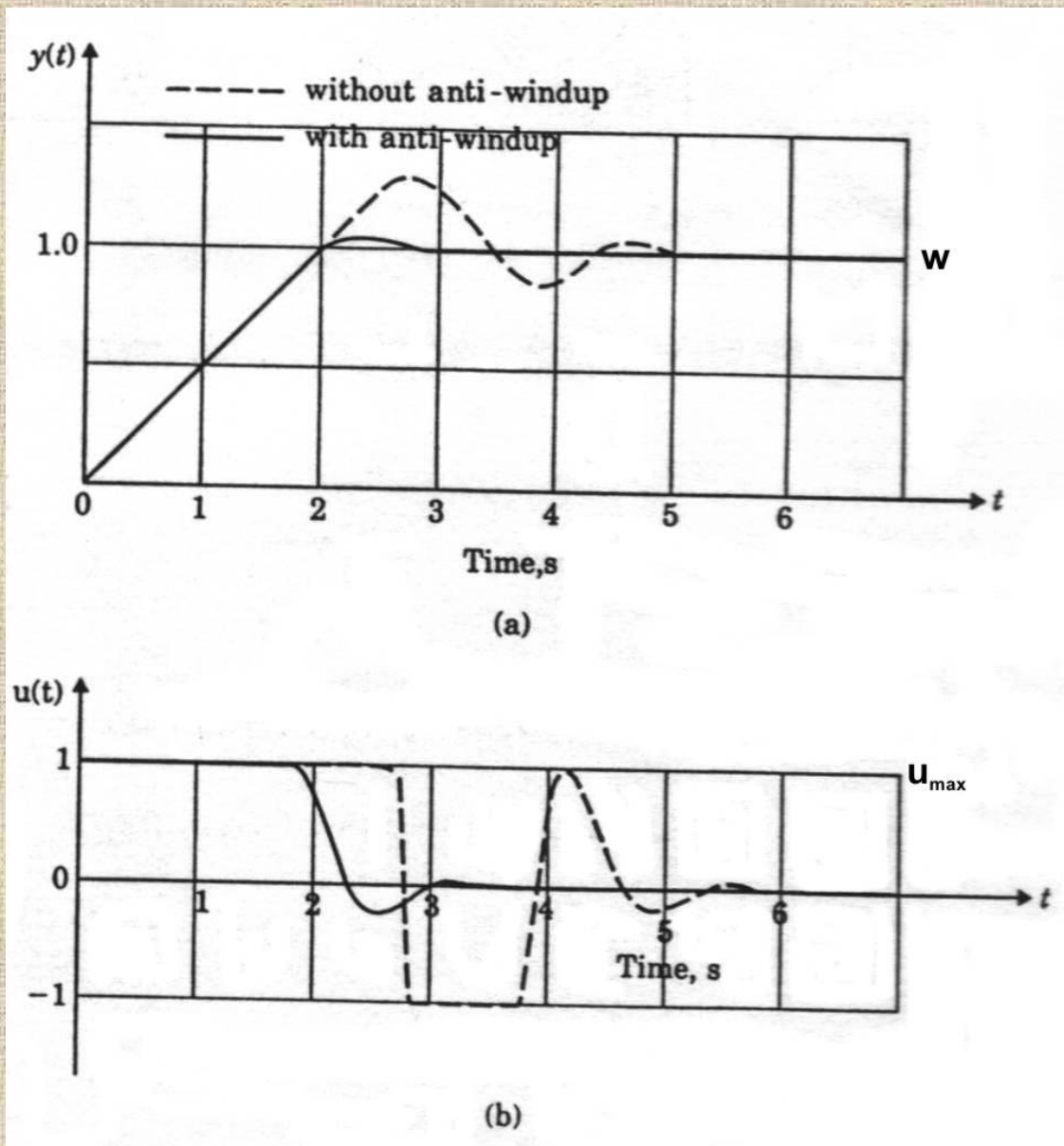
Dla PI z całkowaniem metodą prostokątów:

$$\Delta u(n) = K_p [e(n) - e(n-1)] + \frac{K_p T_s}{T_i} e(n-1)$$

Efekt przeregulowania całkowego (integrator windup)



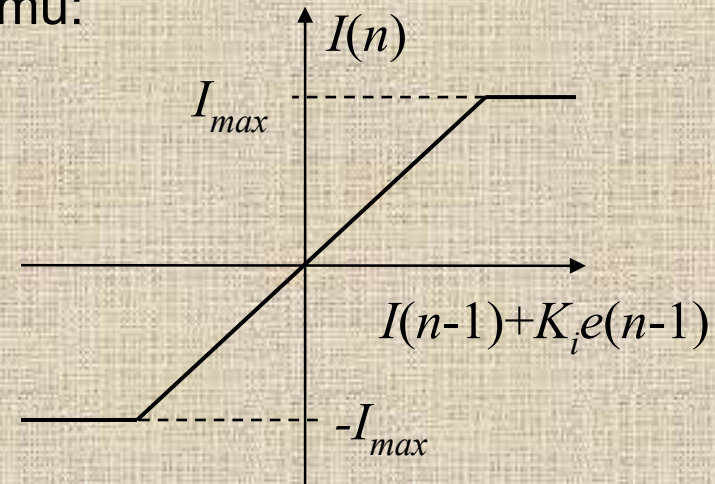
Rys. Schemat układu regulacji PI/PID z elementem z nasyceniem



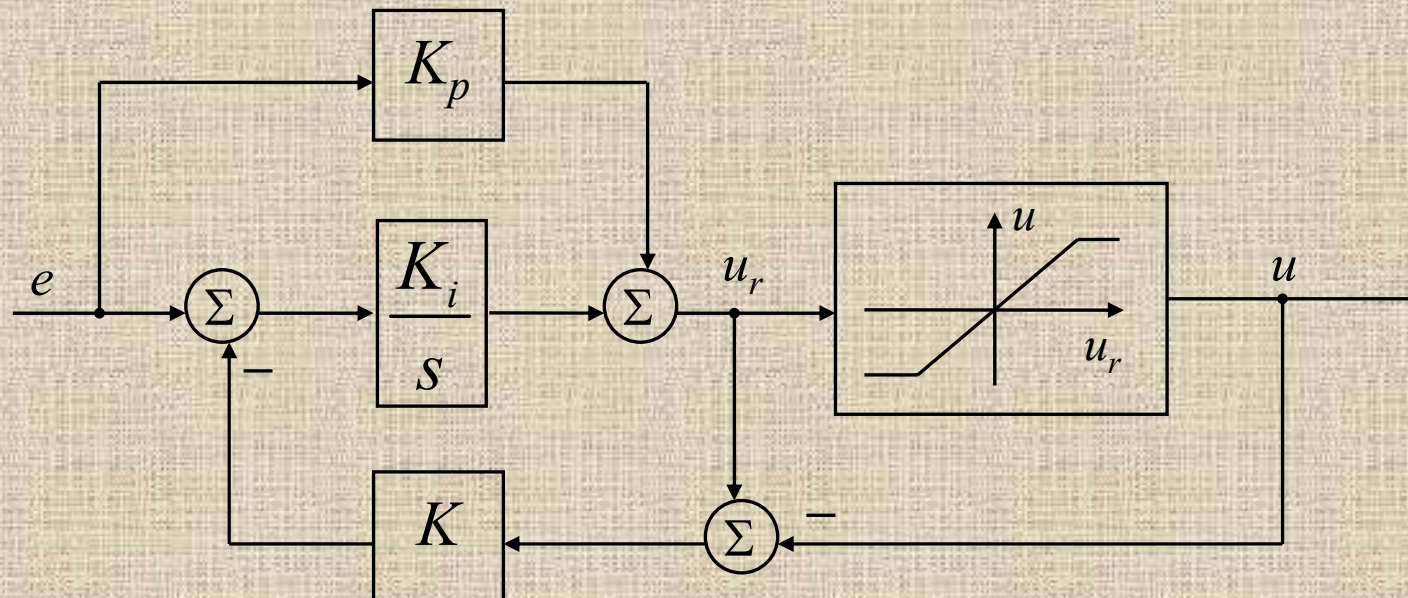
Rys. Typowe przebiegi wielkości regulowanej y i sterowania u

Zapobieganie przeregulowaniu całkowemu:

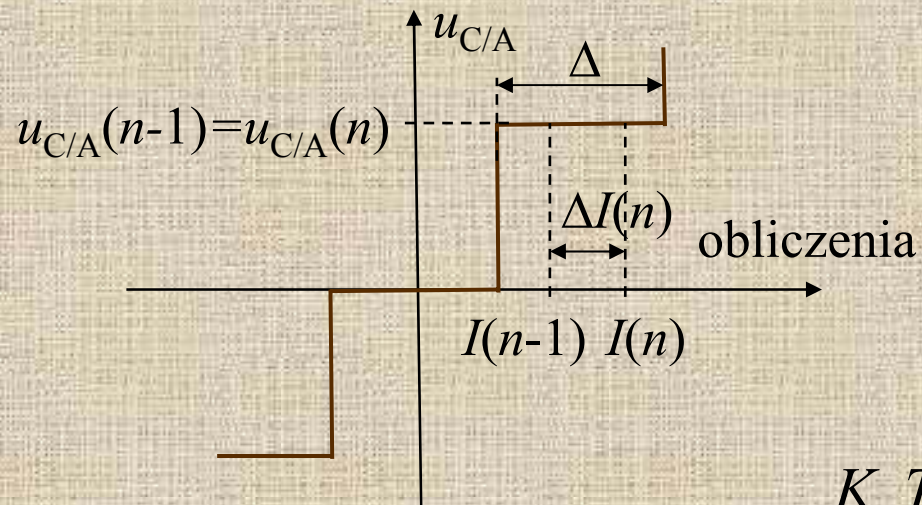
1) całkowanie z nasyceniem



2) pomiar sygnału z wyjścia elementu wykonawczego i „wyłączanie” całkowania wejściu elementu wykonawczego w nasycenia



Uchyb całkowania (integration offset)



$$\Delta I(n) = K_I e(n-1), \quad K_I = \frac{K_p T_s}{T_i}$$

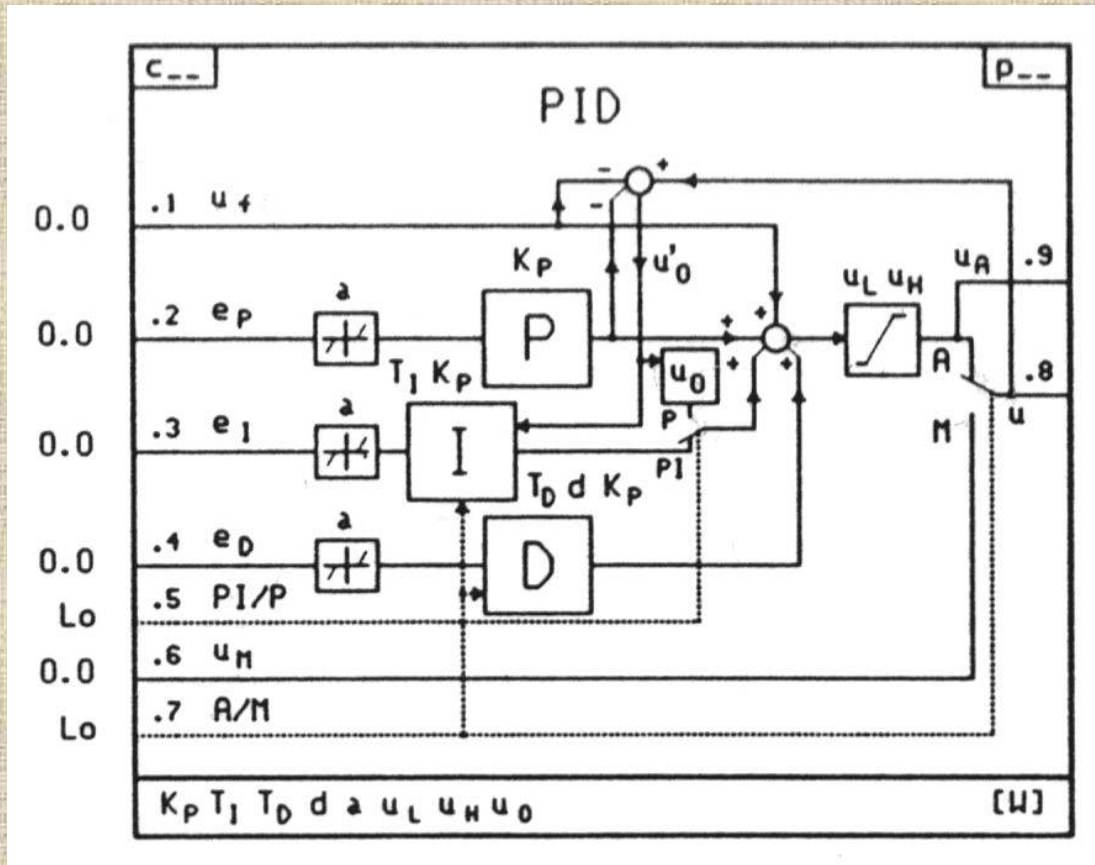
Rys. Jeżeli wartości obliczonego sygnału sterującego w kolejnych chwilach mieszczą się w jednym poziomie kwantowania C/A, to wyjściowy sygnał sterujący nie zmieni się (powstaje martwa strefa reakcji regulatora).

Efekt offsetu całkowania powstaje w przypadku małego K_p (sterowanie obiektów z opóźnieniem) i/lub przy zbyt częstym próbkowaniu $T_s \ll T_i$. Przy częstym próbkowaniu część całkująca szybciej kumuluje też błędy zaokrągleń. Błąd offsetu dla C/A $m+1$ -bitowego przy reprezentacji ułamkowej $[-1, +1]$:

$$e_{\text{off_max}} = \frac{2^{-m}}{K_I} \cdot 100\%$$

Przykład: Dla przetwornika $m=12$ -bitowego o zakresie $0 \dots 10$ V i $K_p=1$, $T_i=40$ s, $T_s=0.1$ s błąd regulacji $e \approx 1$ (10% zakresu C/A) może nie zmienić sygnału sterującego.

Typowe bloki funkcyjne cyfrowego regulatora PID



Typowe zakresy parametrów
aparatowego regulatora PID:

$$K_p = 0.1 \dots 100$$

$$T_i = 0.1 \dots 10\,000 \text{ s}$$

$$T_d = \text{OFF}, 1 \dots 3000 \text{ s}$$

$$d = 0.1 \dots 10.0$$

$$a = 0 \dots 10\%$$

$$u_L = -10 \dots 110\%$$

$$u_H = \text{j.w.}, u_H \geq u_L$$

$$u_0 = \text{AUTO}, -10 \dots 110\%$$

Rys. Schemat algorytmu PID

Bezuderzeniowe przełączanie *Manual/Auto* ($u_o=AUTO$)

Dla części P w stanie *Auto*: $u_A = K_p e_p + u_f + u_0$

W stanie *Manual*: $u_0' = u_M - K_p e_p - u_f$ (*)

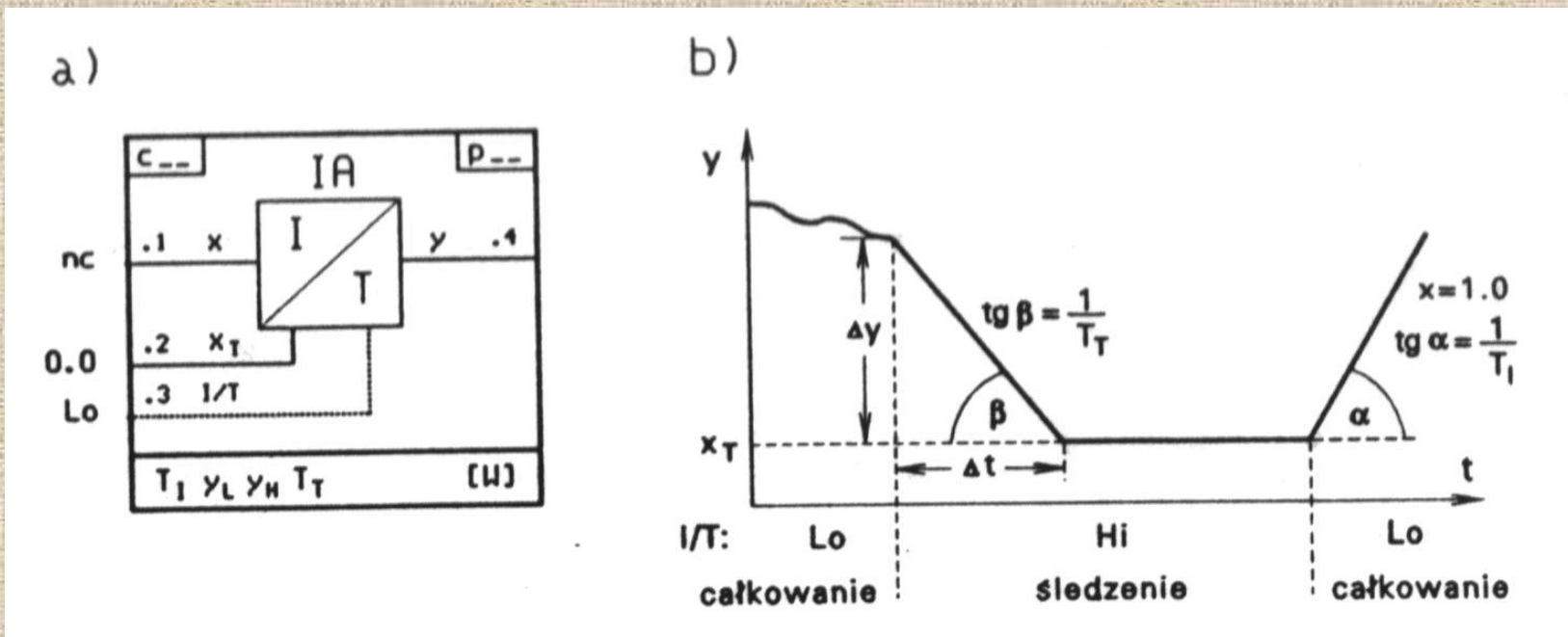
co w chwili przełączenia zapewnia zapewnia $u_A = u_M$.

Część I pracuje wg wzoru: $u_A = K_p e_p + \frac{K_p}{T_i} \int_0^t e dt + u_f + u_{oI}$

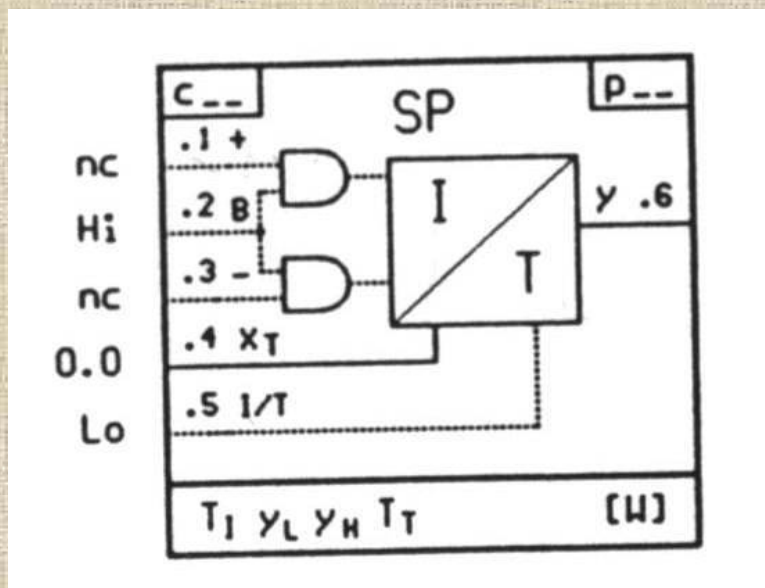
gdzie u_{oI} to początkowa wartość całki zastępowana przez u_0' wg wzoru (*).

Przy przełączaniu (w stanie ustalonym) *Auto*→*Manual* początkowa wartość sygnału sterującego u_M musi być równa końcowej wartości sygnału sterującego u_A .

Przy przełączaniu *Manual*→*Auto*: 1) początkowa wartość u_A musi być równa końcowej wartości u_M , 2) początkowa wartość błędu sterowania e_0 musi być równa zero.

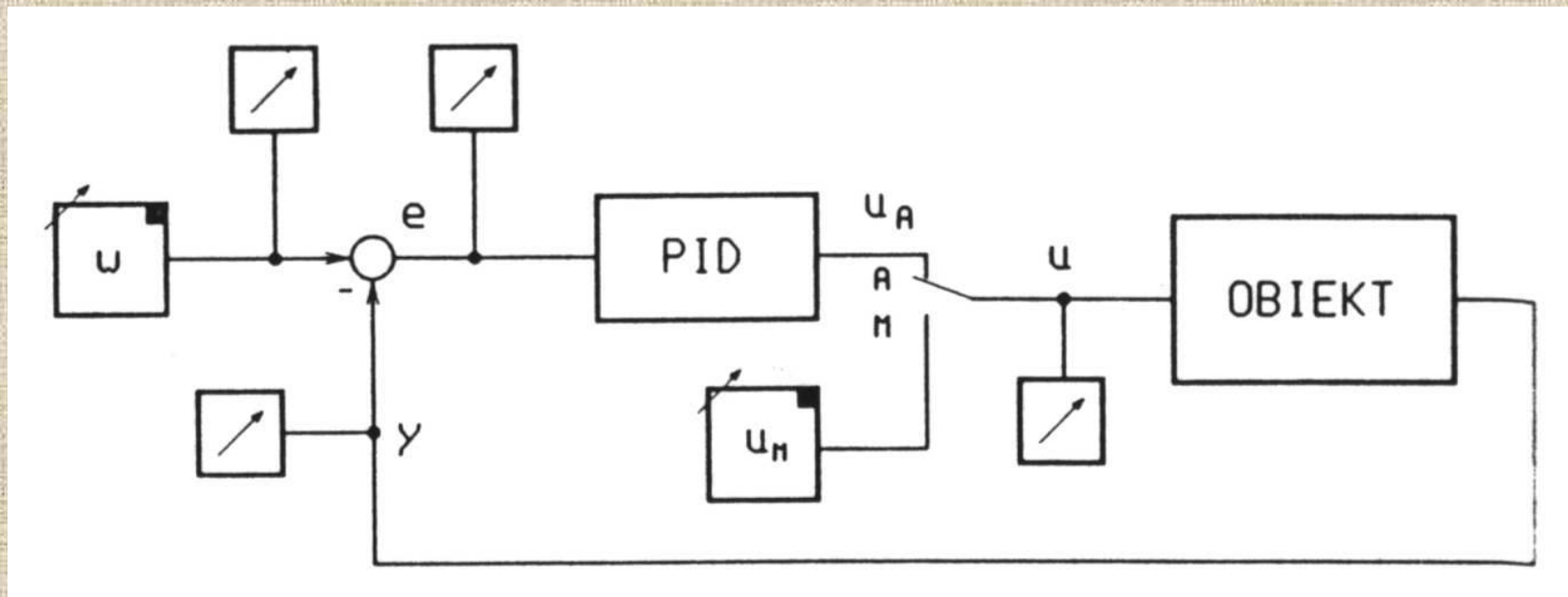


Rys. Integrator: a) schemat, b) przebiegi czasowe

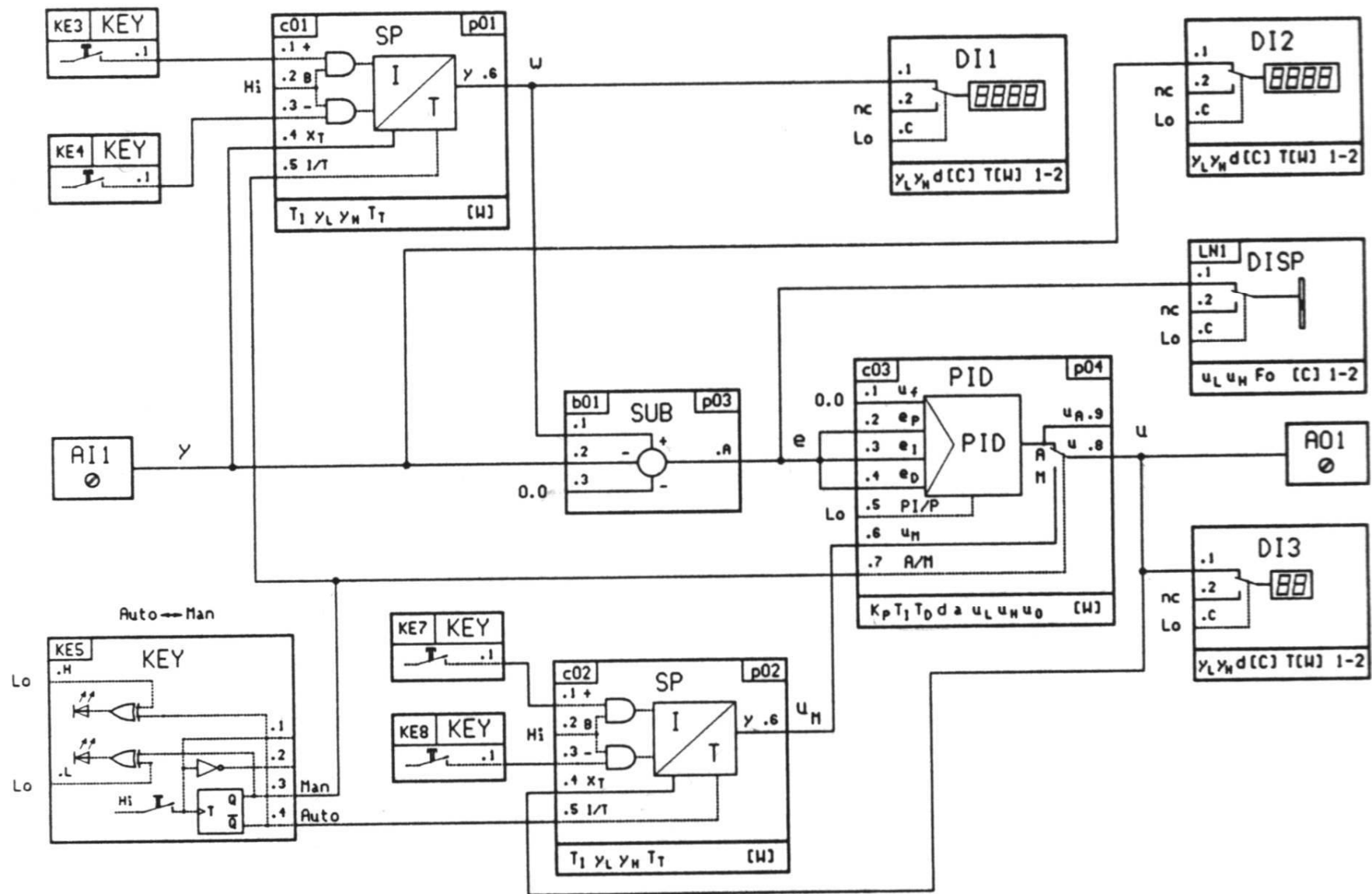


$$y = \frac{1}{T_i} \int_0^t x dt + y_0$$

Rys. Zadajnik SP (setpoint)



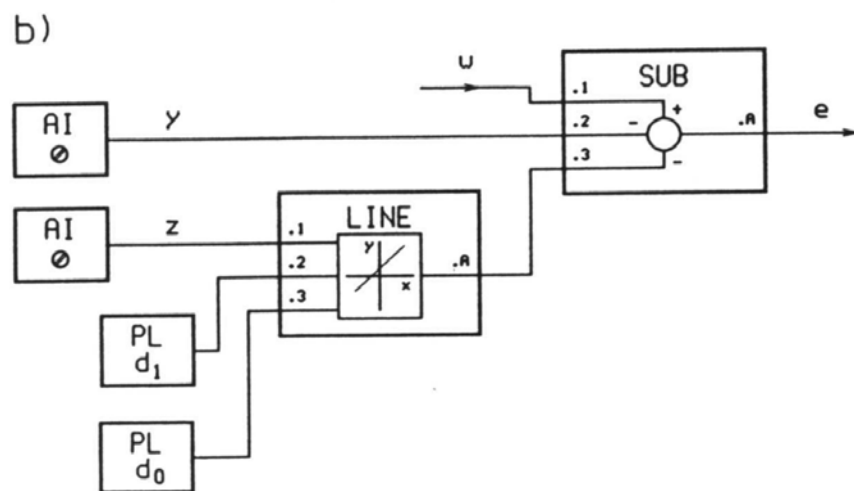
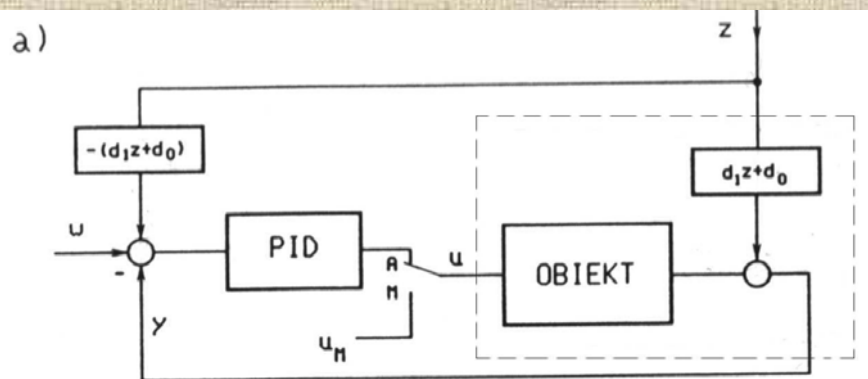
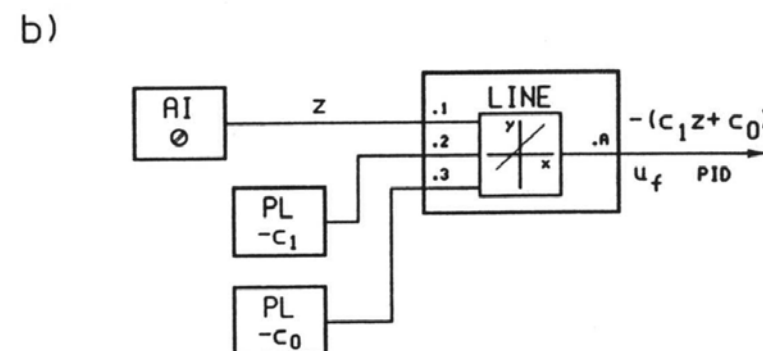
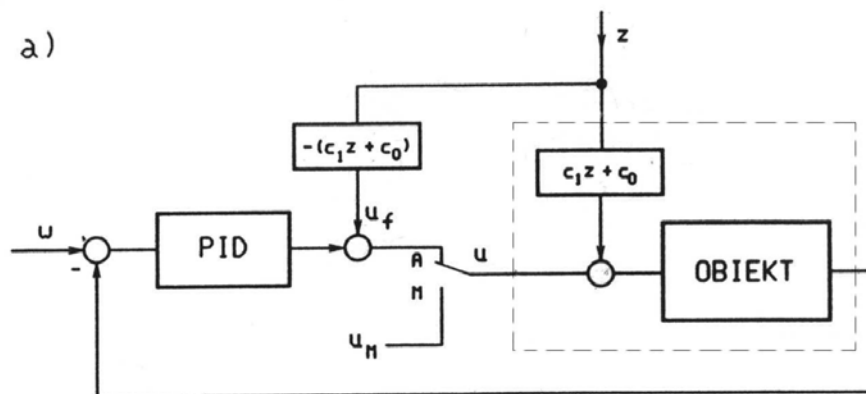
Rys. Schemat ogólny układu regulacji stałowartościowej



Rys. Schemat szczegółowy układu regulacji stałowartościowej

Kompensacja zakłóceń mierzalnych

Rys. Kompensacja zakłócenia działającego na wejście obiektu regulacji



Rys. Kompensacja zakłócenia działającego na wyjściu obiektu regulacji

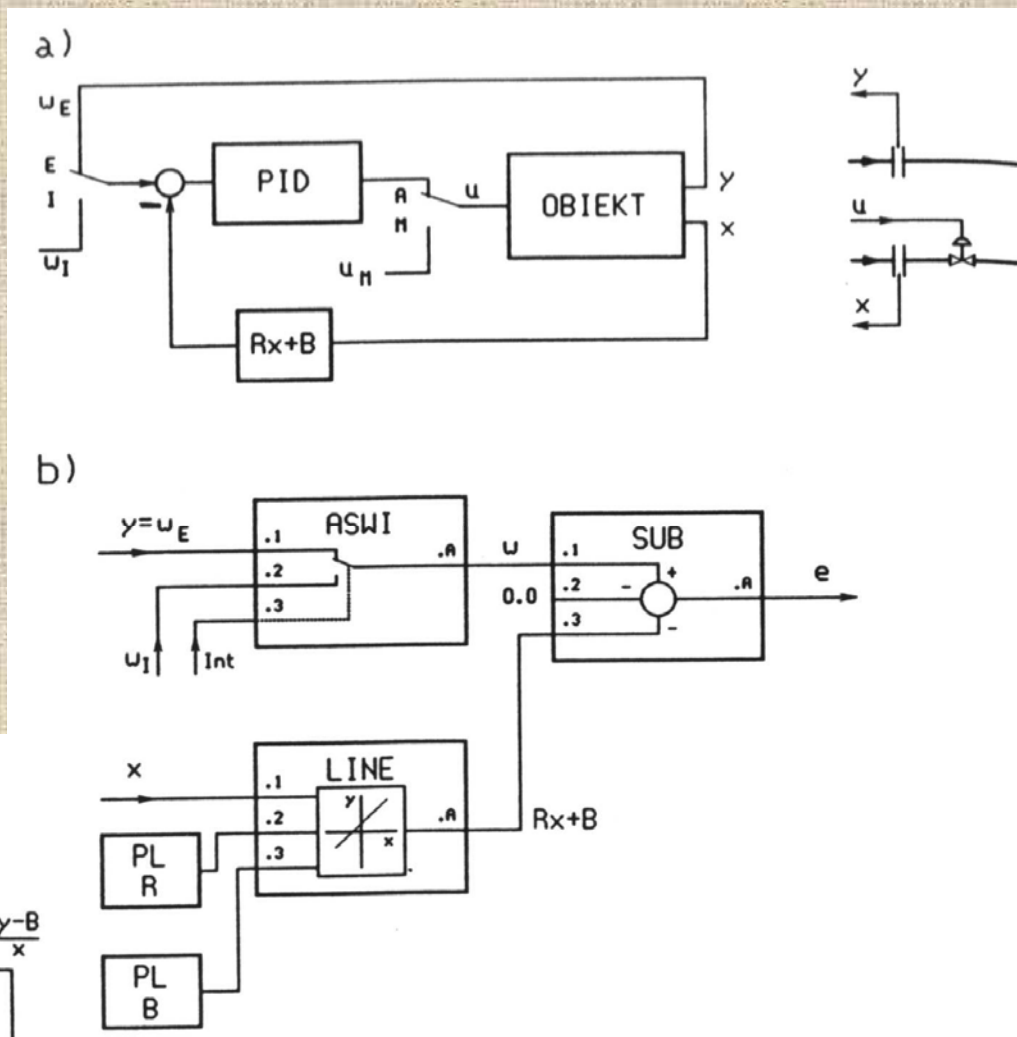
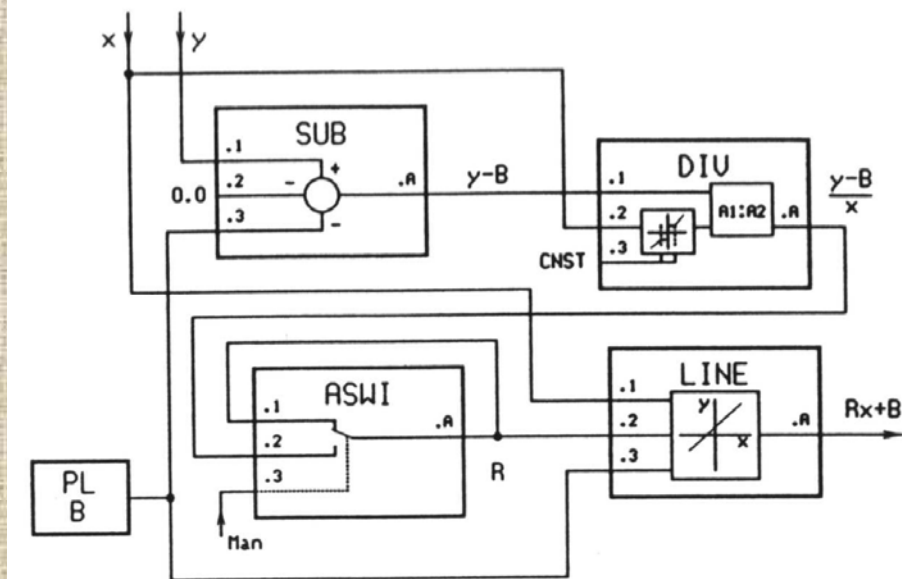
Regulacja stosunku

$$\bar{y} = R\bar{x} + B$$

Mierzone są zmienne x i y ,
ale tylko x podlega sterowaniu.
Dla $B=0$:

$$\frac{\bar{y}}{\bar{x}} = R$$

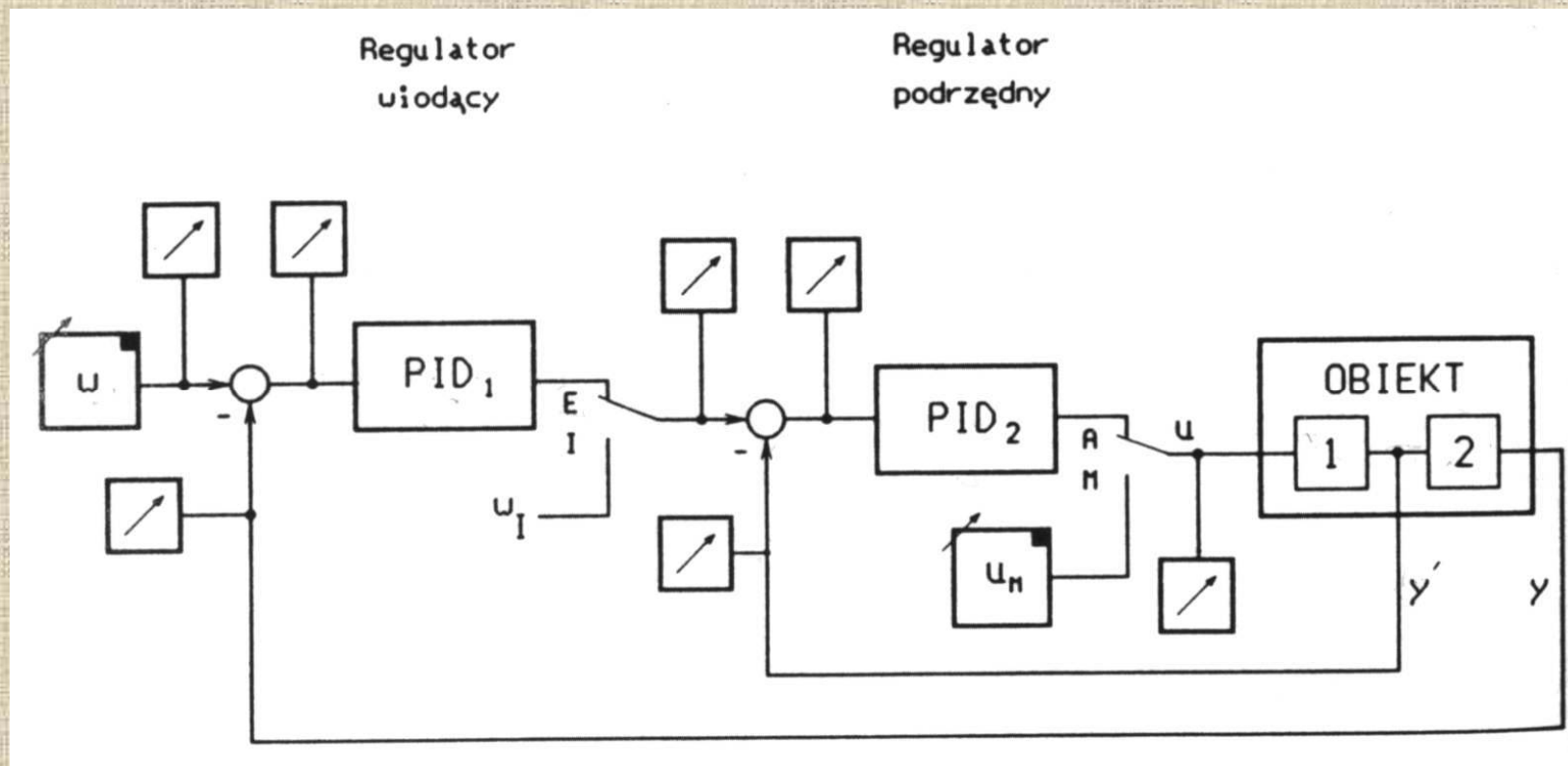
Auto-Ratio



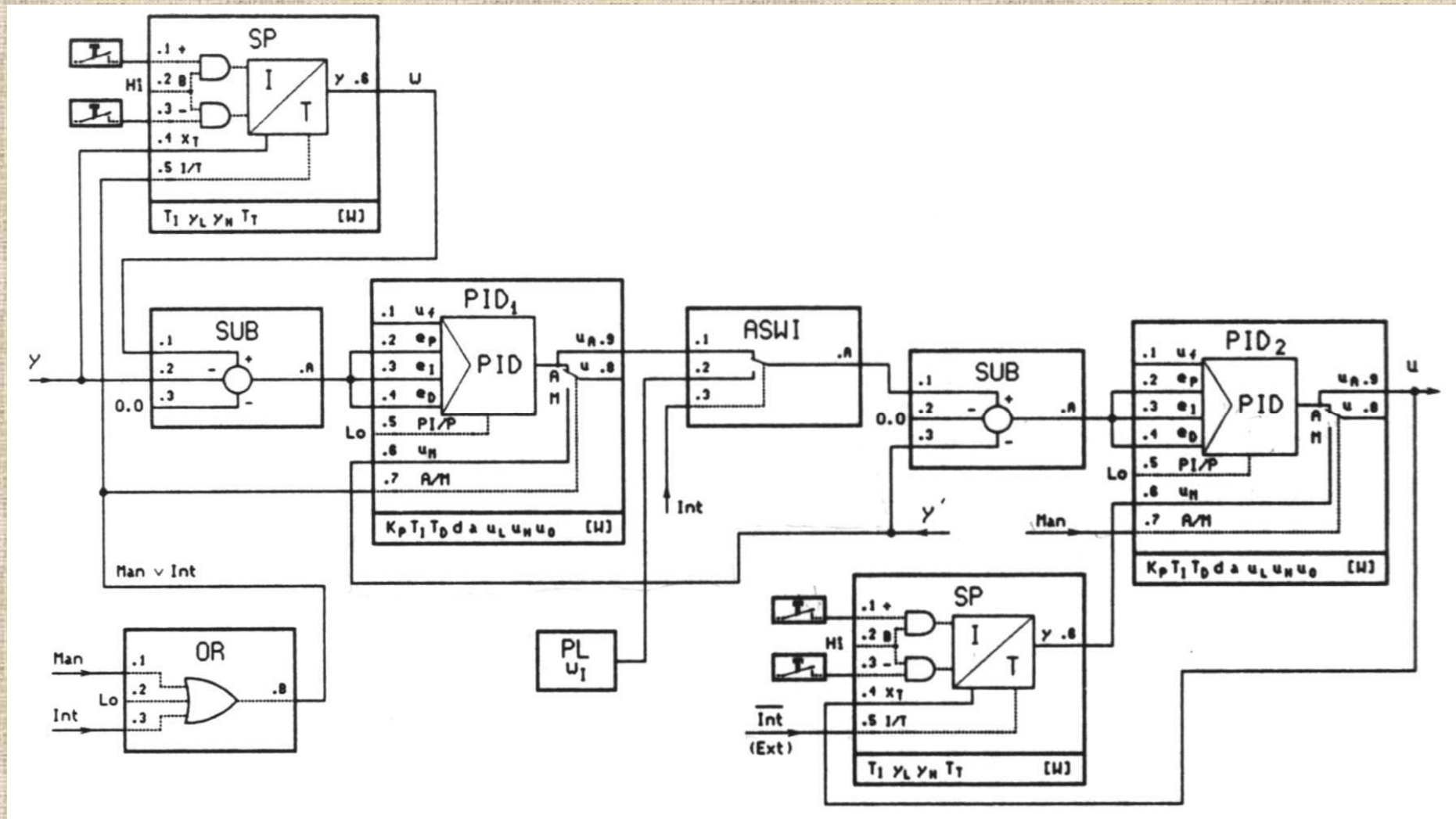
W trybie *Manual* stosunek R jest
określany automatycznie wg wzoru:

$$R = \frac{y - B}{x}$$

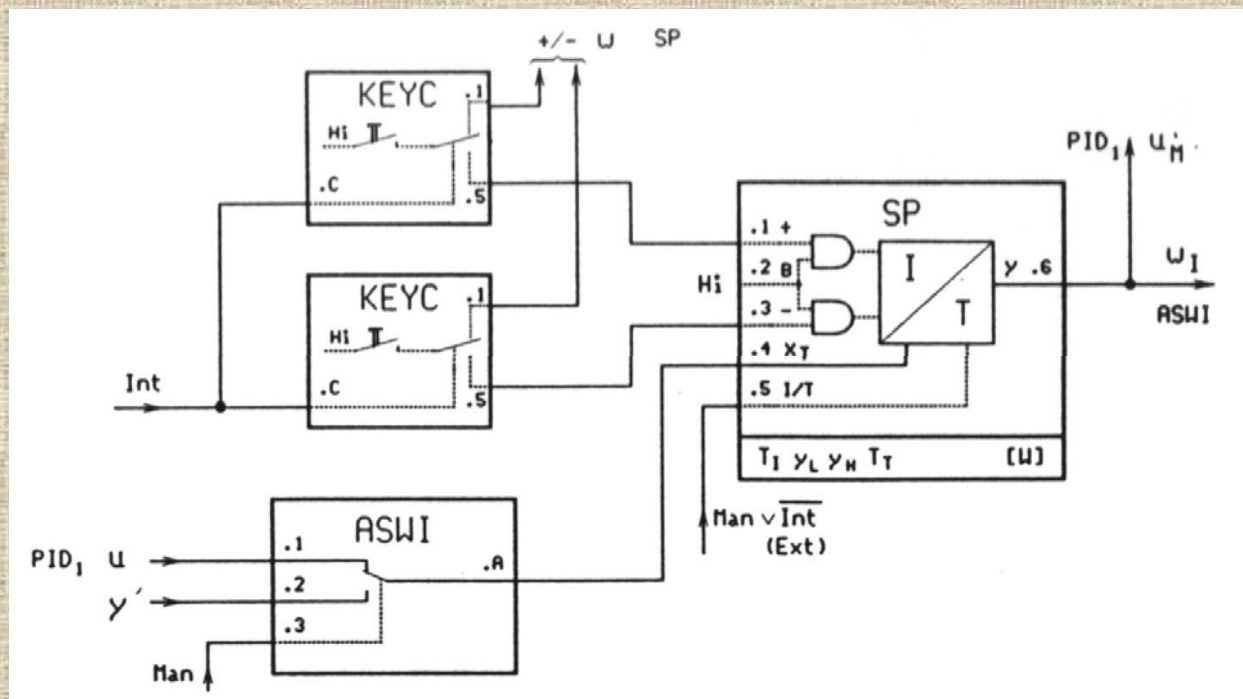
Regulacja kaskadowa



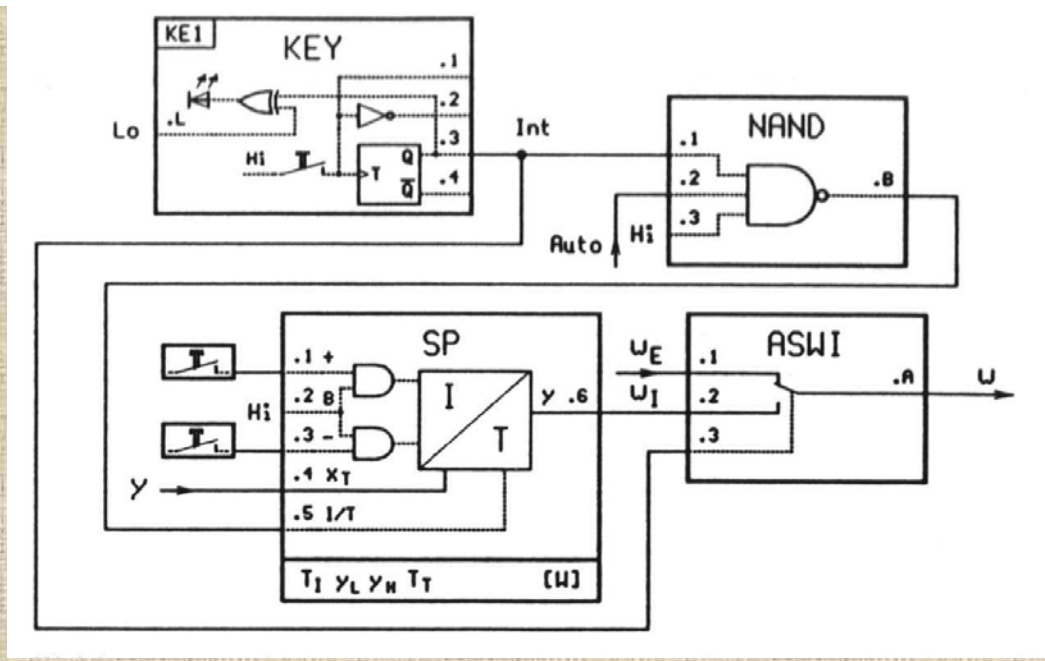
Rys. Schemat ogólny układu regulacji kaskadowej



Rys. Schemat szczegółowy układu regulacji kaskadowej



Rys. Regulacja kaskadowa – zadawanie zmiennej pośredniej w_i



Rys. Wybór wartości zadanej.

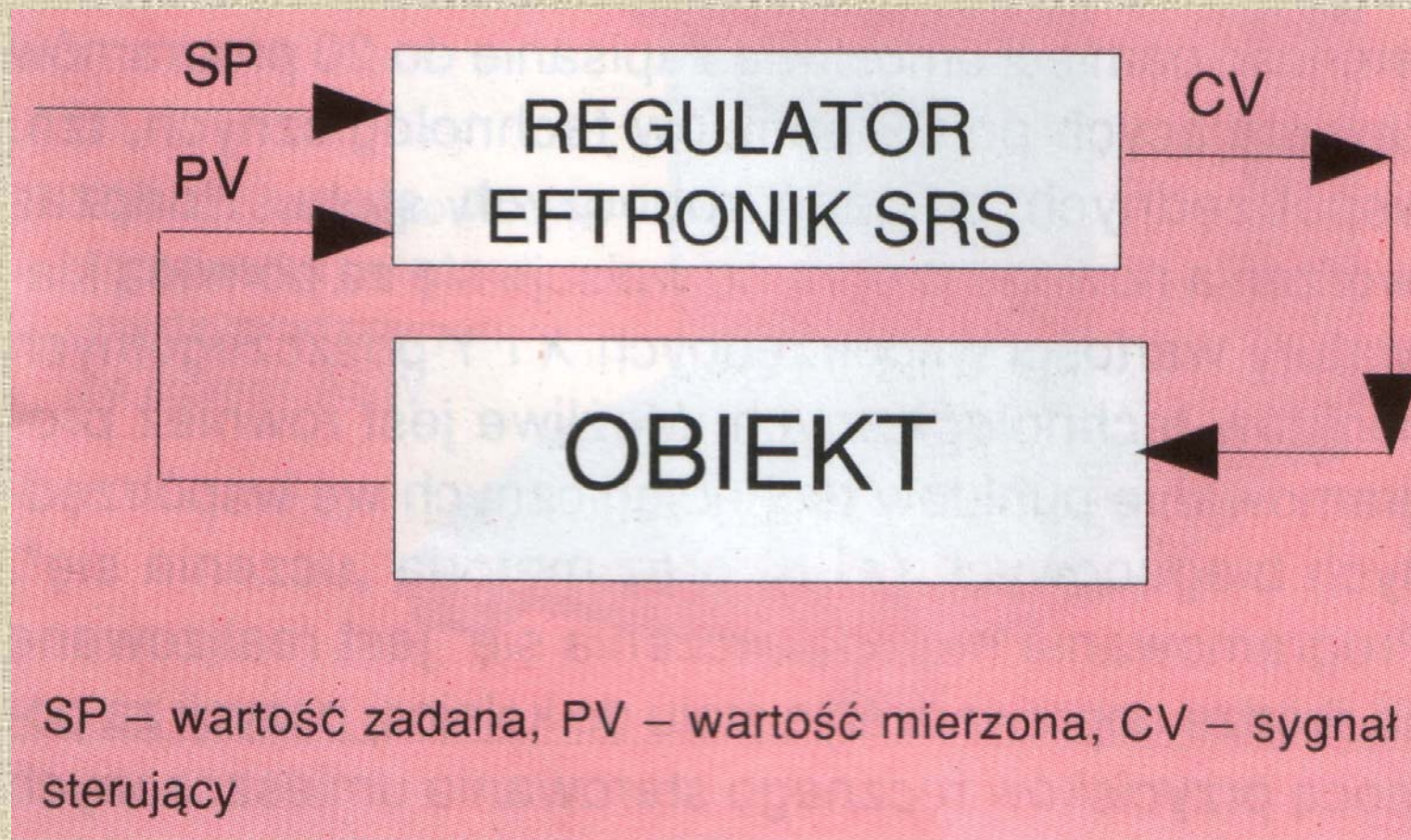


Regulator
EFTRONIK XF
typ U496



Regulator
EFTRONIK SRS
typ U494

Rys 1. Pulpity operatorskie regulatorów EFTRONIK XF i SRS



Regulatory PID wyższych rzędów

Implementacja cyfrowa pozwala na łatwe rozszerzanie algorytmu PID, co umożliwia uzyskanie regulacji lepszej w porównaniu z klasycznym PID.

Ogólna postać transmitancji regulatora rzędu l :

$$G_R(z) = \frac{U(z)}{E(z)} = \frac{p_0 + p_1 z^{-1} + \dots + p_l z^{-l}}{1 - z^{-1}}$$

- rząd $l=1$ daje algorytm PI: $u(n) - u(n-1) = p_0 e(n) + p_1 e(n-1)$

Przy całkowaniu w części I metodą prostokątów:

$$p_0 = k_p, \quad p_1 = k_p \left(\frac{T_s}{T_i} - 1 \right)$$

- rząd $l=2$ daje algorytm PID (z idealnym różniczkowaniem 2-punktowym):

$$u(n) - u(n-1) = p_0 e(n) + p_1 e(n-1) + p_2 e(n-2)$$

$$\text{Stąd: } p_0 = k_p \left(1 + \frac{T_d}{T_s} \right), \quad p_1 = k_p \left(\frac{T_s}{T_i} - 2 \frac{T_d}{T_s} - 1 \right), \quad p_2 = k_p \frac{T_d}{T_s}$$

- rząd $l=3$ daje algorytm PID² itd.

Struktura regulatora o różnej dynamice dla sygnałów: zadanego i sprzężenia zwrotnego

Skrótowny zapis równania różnicowego regulatora:

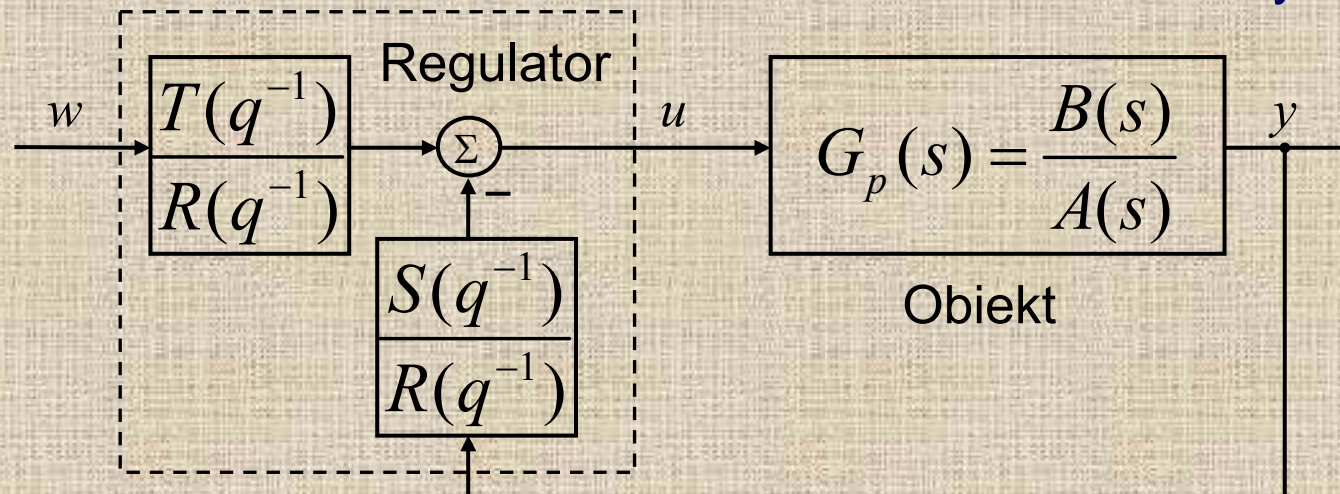
$$R(q^{-1})u(n) = T(q^{-1})w(n) - S(q^{-1})y(n)$$

gdzie z^{-1} jest operatorem przesunięcia, tj. $q^{-k}u(n) = u(n-k)$.

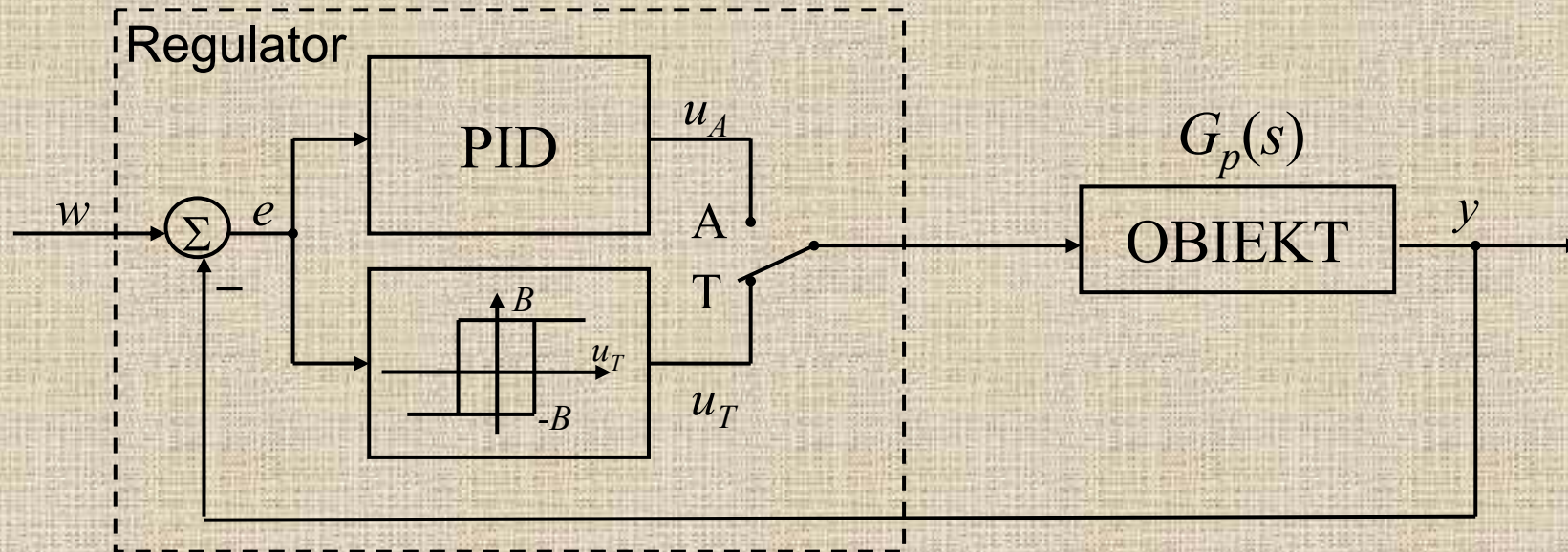
Dla wielomianu postaci: $S(q^{-1}) = s_0 + s_1q^{-1} + \dots + s_Nq^{-N}$
skrótowny zapis równania różnicowego oznacza:

$$S(q^{-1})y(n) = s_0y(n) + s_1y(n-1) + \dots + s_Ny(n-N)$$

Całkowanie: $R(q^{-1}) = (1 - q^{-1})(1 - aq^{-1})$ (dodatkowa inercja – rzeczywiste D)



Regulator PID z samonastrajaniem (autotuning)



Kryterium stabilności Nyquista dla układu z elementem nieliniowym:

$$1 + J(A)G(j\omega) = 0$$

$$J(A) = \frac{4B}{\pi A}$$

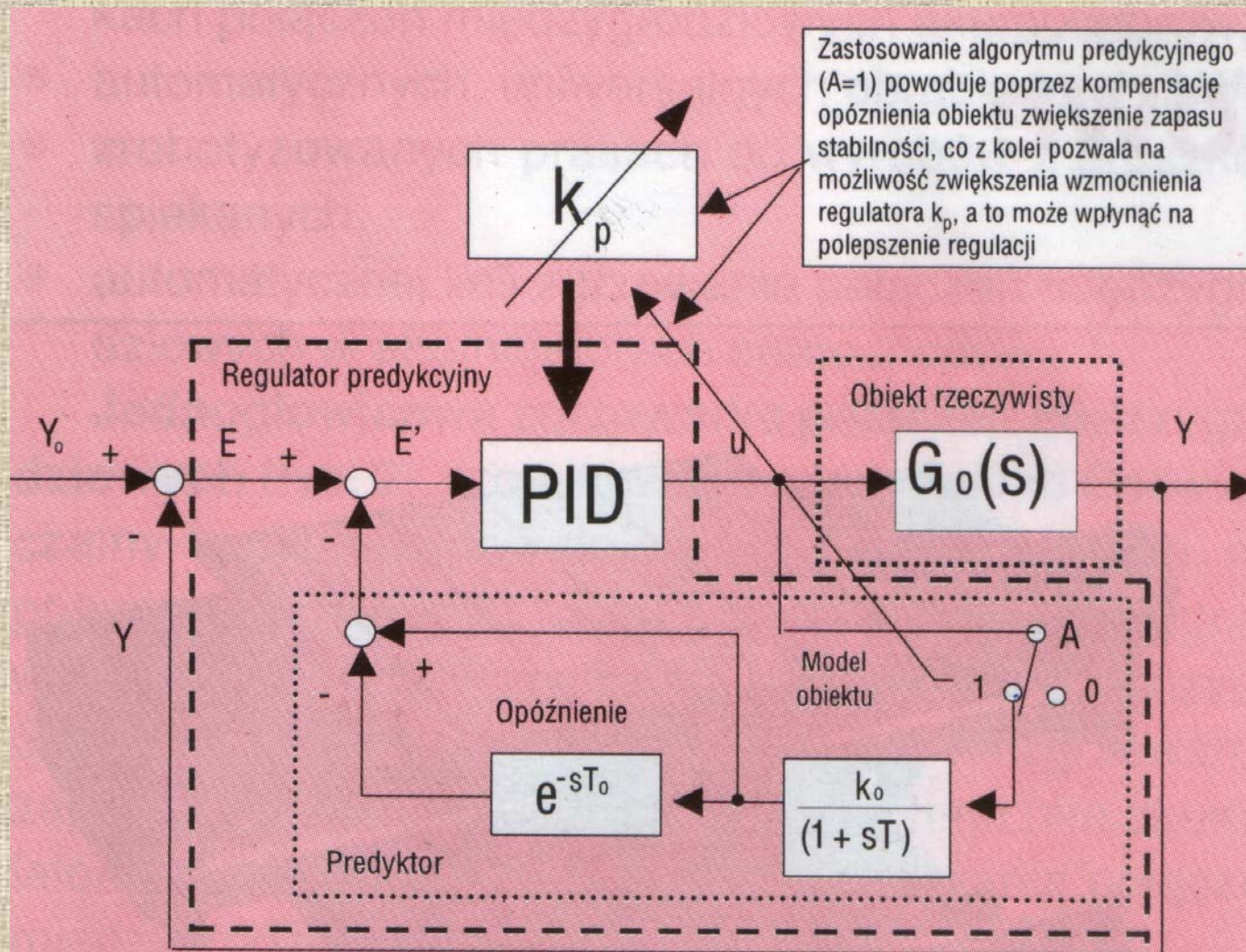
Funkcja opisująca przekaźnika idealnego (bez histerezy):

W trybie T regulator wzbudza w układzie zamkniętym drgania wokół punktu pracy (o niewielkiej, nieszkodliwej amplitudzie) i wyznacza ich okres, amplitudę i ew. opóźnienie, co umożliwia oszacowanie parametrów modelu obiektu, np. postaci:

$$G_{pm}(s) = \frac{k_0}{Ts + 1} e^{-sT_0} \quad \text{i dobrane do nich nastawy regulatora.}$$

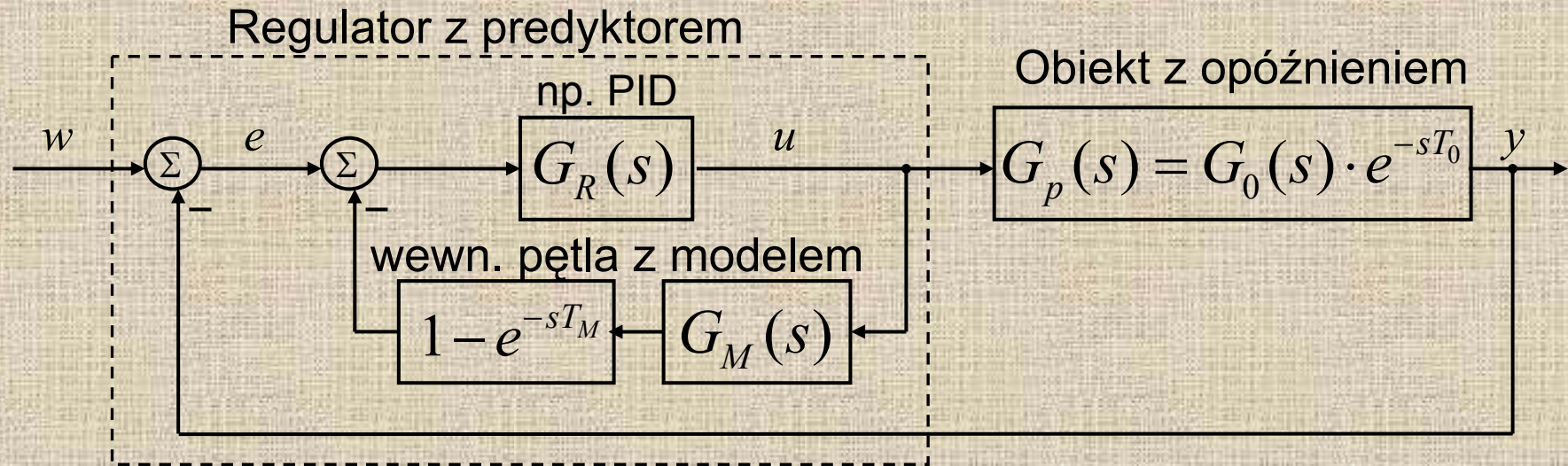
Autotuning może być powtarzany po zmianie punktu pracy lub okresowo.

Predyktor Smitha



Rys. 4. Schemat regulatora predycyjnego Smitha

Predyktor jest wewnętrzną pętlą sprzężenia w regulatorze (np. PID), w której modelowana jest dynamika obiektu z opóźnieniem i kompensowany sygnał sprzężenia zwrotnego z obiektu, a wytwarzany sygnał sprzężenia z *modelu obiektu bez opóźnienia*.



Transmitancja ciągła układu zamkniętego:

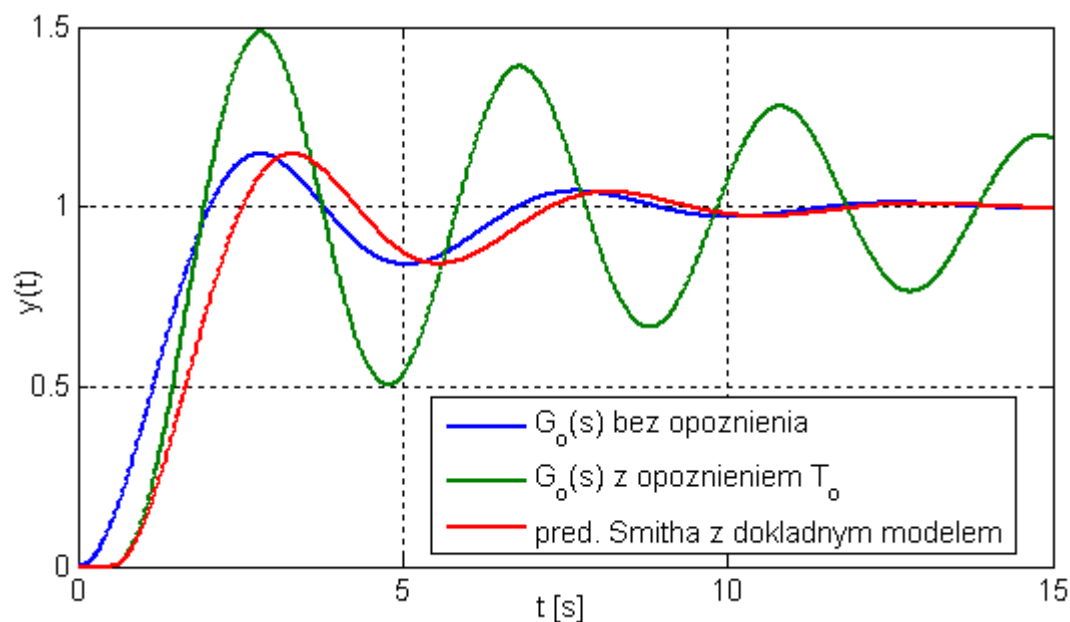
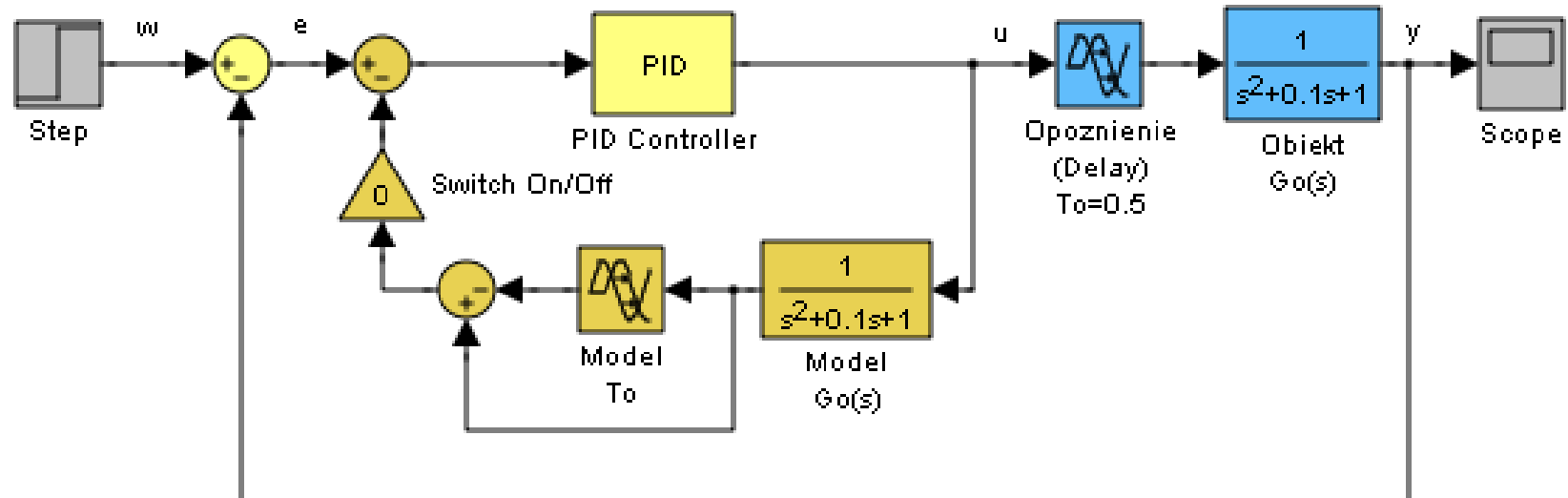
$$G_z(s) = \frac{G_R G_0 e^{-sT_0}}{1 + G_R G_M - G_R G_M e^{-sT_M} + G_R G_0 e^{-sT_0}}$$

Jeżeli model obiektu w regulatorze jest dokładny, tj. $G_M(s) = G_0(s)$, $T_M = T_0$, to:

$$G_z(s) = \frac{G_R G_0}{1 + G_R G_0} e^{-sT_0}$$

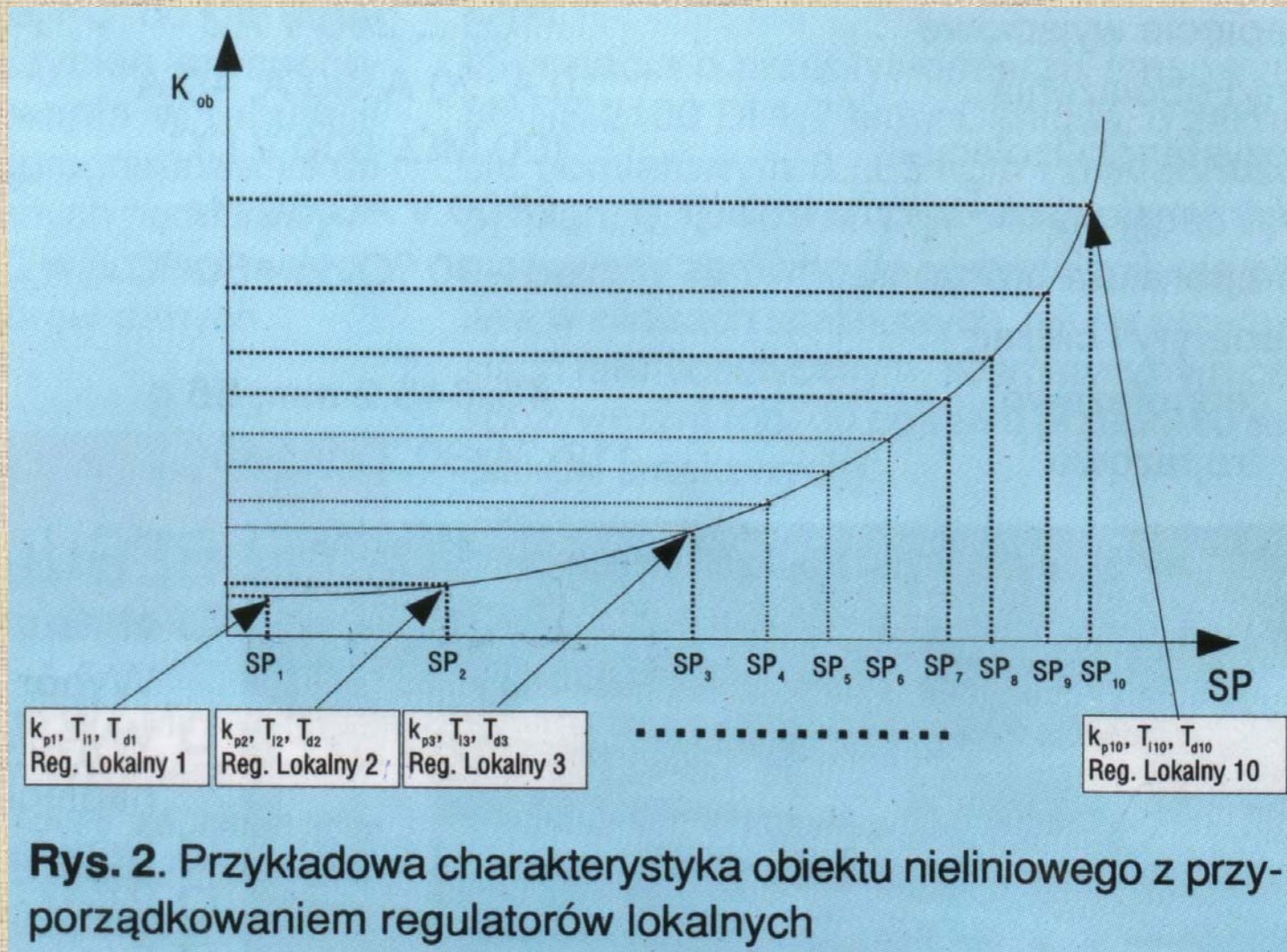
Wniosek: Przy idealnej kompensacji dynamika układu zamkniętego z predyktorem jest taka sama, jak dla obiektu bez opóźnienia, tylko $y(t)$ jest opóźnione o T_0 . Parametry regulatora nastawia się jak dla obiektu bez opóźnienia.

REGULATOR PID Z PREDYKTOREM SMITHA

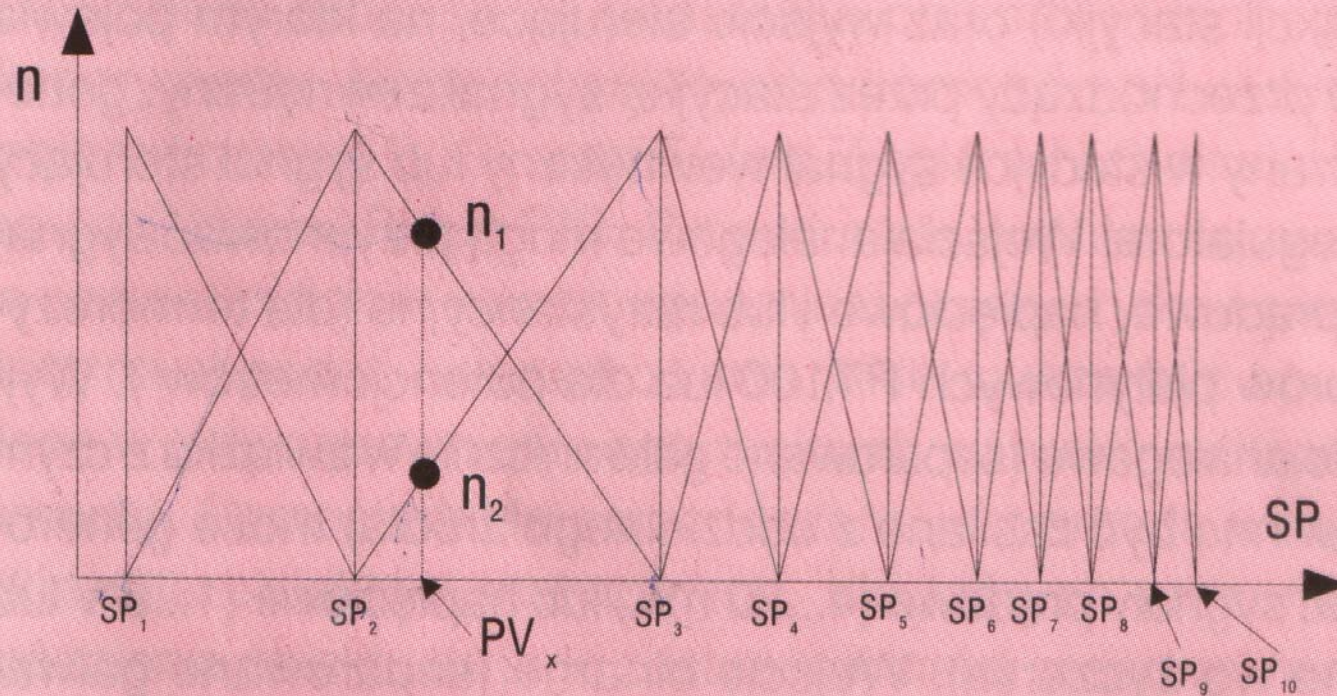


Rys. Symulacja działania regulatora z predyktorem Smitha dla obiektu z opóźnieniem

Regulacja obiektów nieliniowych – lokalne regulatory PID, płynna zmiana nastaw



Rys. 2. Przykładowa charakterystyka obiektu nieliniowego z przy-
porządkowaniem regulatorów lokalnych



Rys. 3. Trójkątne funkcje przynależności wg zasad logiki rozmytej PV_x – jest to sygnał wyjściowy obiektu regulacji (wielkość regulowana), n_1, n_2 – wartości współczynników przynależności odpowiadające chwilowej wartości sygnału PV_x

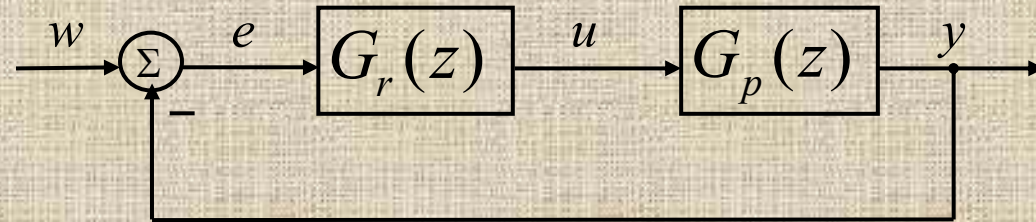
$$n_1 = (SP_3 - PV_x) / (SP_3 - SP_2)$$

$$n_2 = 1 - n_1$$

$$k_{px(\text{fuzzy})} = n_1 * k_{p2} + n_2 * k_{p3}$$

$$T_{ix(\text{fuzzy})} = 1 / k_{px(\text{fuzzy})} * (k_{p2} * n_1 * T_{i2} + k_{p3} * n_2 * T_{i3})$$

$$T_{dx(\text{fuzzy})} = 0,25 * T_{ix(\text{fuzzy})}$$

Regulacja dead-beat

Model obiektu ciągłego po dyskretyzacji (ZOH):

$$G_p(z) = \frac{B(z)}{A(z)} z^{-d} = \frac{b_0 + b_1 z^{-1} + \dots + b_m z^{-m}}{1 + a_1 z^{-1} + \dots + a_m z^{-m}} \cdot z^{-d}$$

gdzie: d – opóźnienie transportowe obiektu ciągłego w okresach próbkowania,
 $N=m+d$ - rząd obiektu dyskretnego.

Dla wymuszenia skokowego: $w(n) = \mathbf{1}(n) \rightarrow W(z) = \frac{1}{1-z^{-1}}$

Poszukujemy regulacji spełniającej warunki:

$$\begin{cases} y(n) = w(n) = 1 & \text{dla } n \geq N \\ u(n) = u(m) & \text{dla } n \geq m \end{cases}$$

$$Y(z) = \sum_{n=0}^{\infty} y(n) z^{-n} = y(1)z^{-1} + y(2)z^{-2} + \dots + \underset{y(N)}{1} \left(z^{-N} + z^{-(N+1)} + \dots \right)$$

$$U(z) = u(0) + u(1)z^{-1} + u(2)z^{-2} + \dots + u(m) \left(z^{-m} + z^{-(m+1)} + \dots \right)$$

Transmitancja układu zamkniętego:

$$G_z(z) = \frac{Y(z)}{W(z)} = (1 - z^{-1}) \left[y(1)z^{-1} + y(2)z^{-2} + \dots + 1 \left(z^{-N} + z^{-(N+1)} + \dots \right) \right] =$$

$$= p_1 z^{-1} + p_2 z^{-2} + \dots + p_N z^{-N} = P(z)$$

(układ zamknięty o skończonej odpowiedzi impulsowej)

Analogicznie: $\frac{U(z)}{W(z)} = q_0 + q_1 z^{-1} + \dots + q_m z^{-m} = Q(z)$

gdzie: $q_0 = u(0)$

$p_1 = y(1), \quad q_1 = u(1) - u(0)$

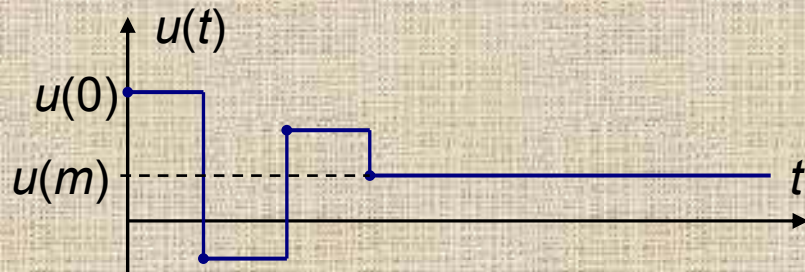
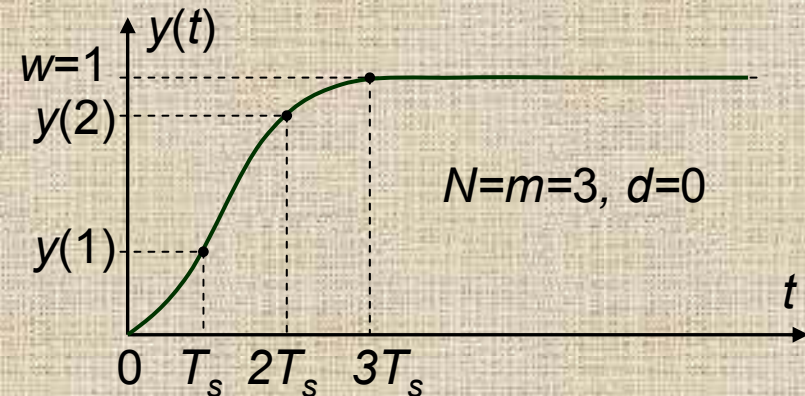
$p_2 = y(2) - y(1), \quad q_2 = u(2) - u(1)$

... ..

$p_N = 1 - y(N - 1), \quad q_m = u(m) - u(m - 1)$

Dla opóźnienia transportowego $d > 0$:

$y(1) = y(2) = \dots = y(d) = 0$ i $p_1 = p_2 = \dots = p_d = 0$.



$$G_z(z) = \frac{G_r(z)G_p(z)}{1 + G_r(z)G_p(z)}, \quad G_p(z) = \frac{Y(z)}{U(z)} = \frac{Y(z)}{W(z)} \cdot \frac{W(z)}{U(z)} = \frac{P(z)}{Q(z)}$$

Transmitancja poszukiwanego równania regulatora dead-beat:

$$G_r(z) = \frac{1}{G_p(z)} \cdot \frac{G_z(z)}{1 - G_z(z)} = \frac{Q(z)}{1 - P(z)} = \frac{q_0 + q_1 z^{-1} + \dots + q_m z^{-m}}{1 - p_1 z^{-1} - \dots - q_N z^{-N}}$$

Współczynniki $P(z)$ i $Q(z)$ wyznacza się przez porównanie wyrazów o jednakowych potęgach z , ponieważ:

$$G_p(z) = \frac{B(z)}{A(z)} z^{-d} = \frac{P(z)}{Q(z)}$$

$$p_{d+1} + p_{d+2} + \dots + p_N = 1 = q_0 \sum_i b_i \quad \text{Stąd:} \quad q_0 = \frac{1}{\sum_i b_i}$$

$$p_{d+1} = b_1 q_0, \quad q_1 = a_1 q_0$$

$$p_{d+2} = b_2 q_0, \quad q_2 = a_2 q_0$$

...

$$p_N = b_m q_0, \quad q_m = a_m q_0$$

Transmitancja układu zamkniętego z regulatorem dead-beat:

$$G_z(z) = \left(p_{d+1}z^{-1} + p_{d+2}z^{-2} + \dots + p_{d+m}z^{-m} \right) z^{-d} = \frac{p_{d+1}z^{N-1} + p_{d+2}z^{N-2} + \dots + p_N}{z^N}$$

$$G_z(z) = \frac{q_0(b_0z^{N-d} + b_1z^{N-d-1} + \dots + b_m)}{z^N}, \quad N = m + d$$

Transmitancję regulatora można zapisać jako:

$$G_r(z) = \frac{q_0 A(z)}{1 - q_0 B(z) z^{-d}}$$

W liczniku $G_r(z)$ występuje wielomian $A(z)$ mianownika transmitancji obiektu, który kasuje bieguny obiektu. Regulacja *dead-beat* przesuwa bieguny układu zamkniętego do $z=0$.

Jedynym parametrem regulacji *dead-beat* jest T_s , od którego zależą współczynniki $G_p(z)$. Początkowa amplituda sterowania $u(0)$ rośnie jak $1/T_s^2$. Regulację *dead-beat* stosuje głównie dla stanów przejściowych, kiedy pożądanym końcowym punktem pracy jest odległy od stanu początkowego.

Inna wersja algorytmu dead-beat zakłada, że czas regulacji zajmuje $N+1$ kroków, co wprowadza dodatkowy wolny parametr *umożliwiający ograniczenie sterowania*. W tej wersji:

$$P(z) = p_1 z^{-1} + p_2 z^{-2} + \dots + p_N z^{-N} + p_{N+1} z^{-(N+1)}$$

$$Q(z) = q_0 + q_1 z^{-1} + \dots + q_m z^{-m} + q_{m+1} z^{-(m+1)}$$

$$q_0 = u(0) \quad - \quad \text{zadane}$$

$$p_{d+1} = q_0 b_1, \quad q_1 = q_0 (a_1 - 1) + \frac{1}{\sum_i b_i}$$

$$p_{d+2} = q_0 (b_2 - b_1) + \frac{b_1}{\sum_i b_i}, \quad q_2 = q_0 (a_2 - a_1) + \frac{1}{\sum_i b_i}$$

...

$$p_N = q_0 (b_m - b_{m-1}) + \frac{b_{m-1}}{\sum_i b_i}, \quad q_m = q_0 (a_m - a_{m-1}) + \frac{1}{\sum_i b_i}$$

$$p_{N+1} = b_m \left(q_0 - \frac{1}{\sum_i b_i} \right), \quad q_{m+1} = a_m \left(-q_0 + \frac{1}{\sum_i b_i} \right)$$

Przykład: Projektowanie regulacji dead-beat dla obiektu ciągłego $G(s)=1/s^2$.

Po dyskretyzacji ZOH:

$$G_p(z) = \frac{T_s^2}{2} \cdot \frac{z+1}{(z-1)^2} = \frac{T_s^2}{2} \cdot \frac{z^{-1} + z^{-2}}{1-2z^{-1} + z^{-2}}, \quad N = m = 2, \quad d = 0$$

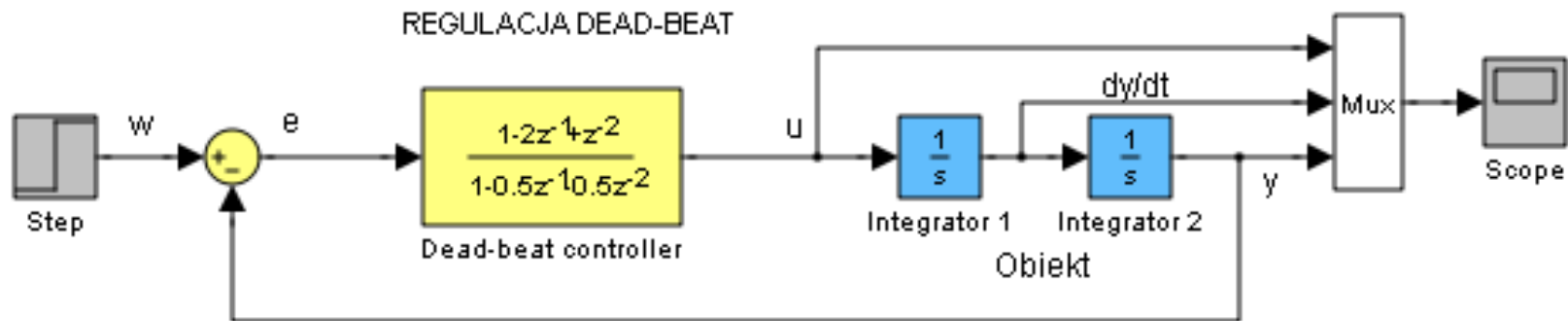
(st.licznika = st.mianownika)

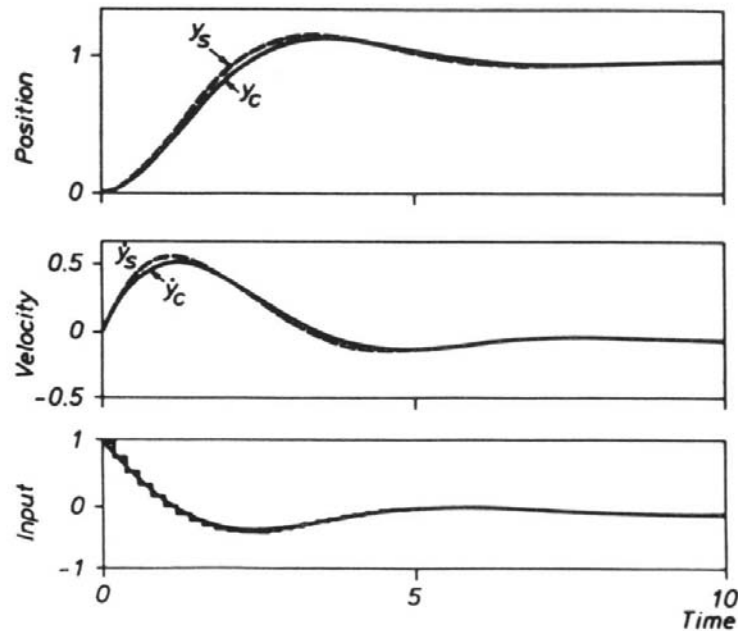
Dla $T_s=1$: $b_0 = 0, \quad b_1 = b_2 = \frac{T_s^2}{2} = \frac{1}{2}, \quad a_1 = -2, \quad a_2 = 1$

$$q_0 = \frac{1}{b_1 + b_2} = 1, \quad q_1 = a_1 q_0 = -2, \quad q_2 = a_2 q_0 = 1, \quad p_1 = b_1 q_0 = \frac{1}{2}, \quad p_2 = b_2 q_0 = \frac{1}{2}$$

Transmitancja regulatora: $G_r(z) = \frac{1}{T_s^2} \cdot \frac{1-2z^{-1} + z^{-2}}{1 - \frac{1}{2}z^{-1} - \frac{1}{2}z^{-2}} = \frac{1}{T_s^2} \cdot \frac{(z-1)^2}{z^2 - \frac{1}{2}z - \frac{1}{2}}$

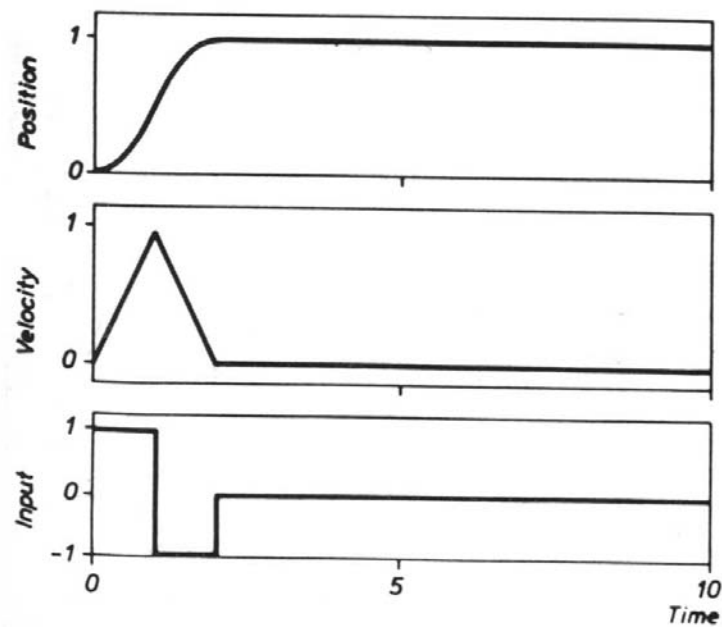
Schemat układu w Simulinku:





Rys. Emulacja dyskretna (*sampled*) regulacji ciągłej

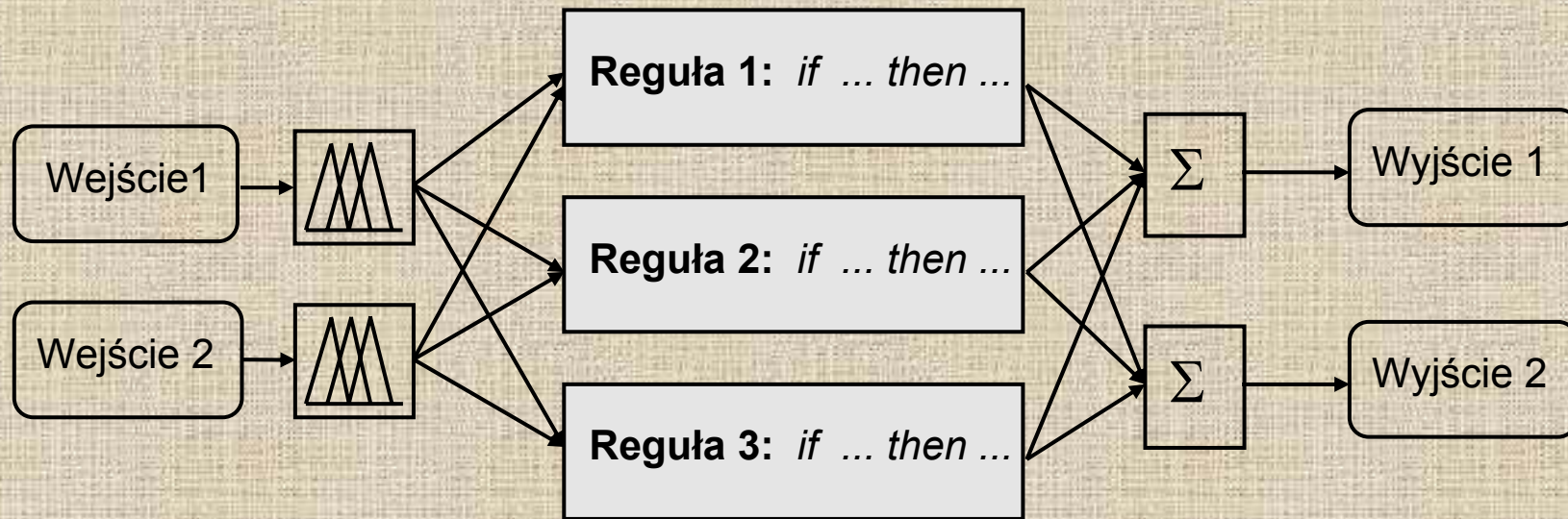
Figure 1.7 Step response of a double-integrator plant with state feedback, continuous-time solution y_c , and sampled approximation y_s , when the sampling period is 0.2 s.



Rys. Regulacja dead-beat (projektowanie regulacji dla dyskretnego modelu obiektu)

Figure 1.8 Computer control of the double-integrator plant with a deadbeat strategy. The sampling period is 1 s.

Podstawy regulacji rozmytej



Wejścia - konkretne liczby, podlegają fuzyfikacji

Reguły obliczane w sposób równoległy z zastosowaniem zasad wnioskowania rozmytego

Wyniki implikacji są łączone (agregacja) i poddawane defuzyfikacji

Wyjścia - konkretne liczby

Rys. Ogólna struktura układu wnioskowania rozmytego Mamadani'ego (*Fuzzy Inference System – FIS*)

Reguła wnioskowania rozmytego: *angle*, *velocity*, *force* – zmienne wej i wyj, Big, Pos_Small, Neg_Big – zbiory rozmyte (zmienne lingwistyczne)

if (*angle is Big*) and (*velcity is Pos_Small*) then (*force is Neg_Big*)

Rozmyte operatory logiczne:

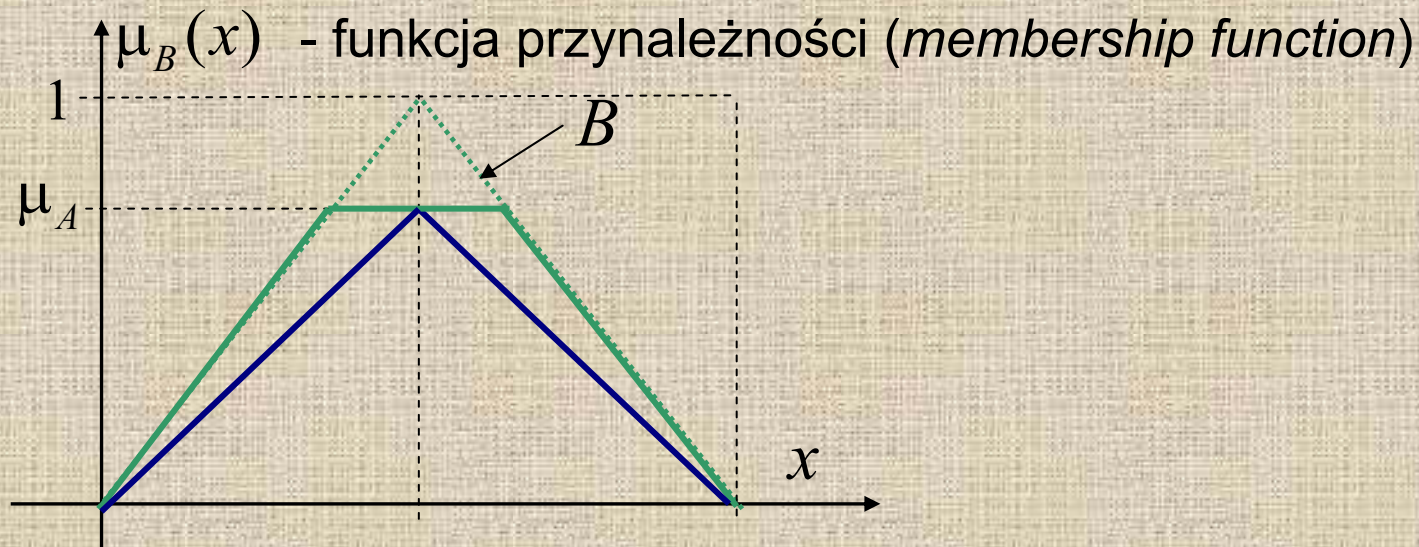
A AND B: $\min[\mu(A), \mu(B)]$ lub $\text{prod}[\mu(A), \mu(B)] = \mu(A) * \mu(B)$

A OR B: $\max[\mu(A), \mu(B)]$ lub $\text{probOR}[\mu(A), \mu(B)] = \mu(A) + \mu(B) - \mu(A) * \mu(B)$

NOT A: $1 - \mu(A)$

Metody **implikacji** $A \rightarrow B$ (przesłanka \rightarrow konkluzja B – zbiór rozmyty):

$\mu_{Bimp} = \min(\mu_A, \mu_B)$ – obcięcie, $\mu_{Bimp} = \mu_A * \mu_B$ – skalowanie



Regulacja rozmyta prowadzi do regulatorów o *zmiennej strukturze* (kombinacji ważonej reguł), przełączenia następują w momentach odpalenia (wygaszenia) poszczególnych reguł.

Rozmyte regulatory PID (fuzzy controllers)

$$u = K_p e + K_i \underbrace{\int_0^t e d\tau}_{\rho} + K_d \dot{e}$$

Wektor zmiennych:

$$\mathbf{x} = [x_1 \ x_2 \ x_3]^T = [e \ \rho \ \dot{e}]^T$$

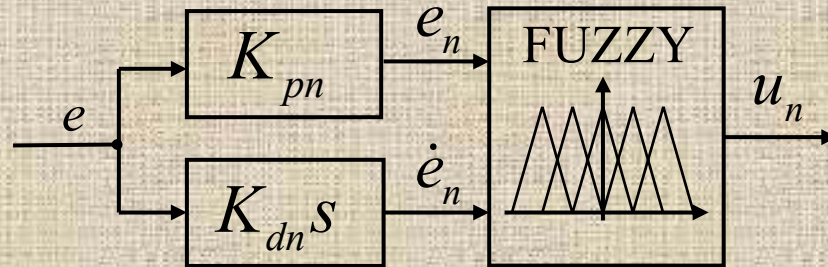
Ogólna forma i -tej reguły regulatora PID (typu Mamdani'ego):

$R^{(i)}$: if (e is $WP^{(i)}$) and (ρ is $WI^{(i)}$) and (\dot{e} is $WD^{(i)}$) then (u is $WU^{(i)}$)

gdzie: $WP^{(i)}$, $WI^{(i)}$, $WD^{(i)}$ – wartości lingwistyczne i -tej reguły (zbiory rozmyte), $i=1,2,\dots,m$.

Wartości sygnałów normalizuje się tak, aby po normalizacji współczynniki poszczególnych składowych były jednakowe: $K_{pn} = K_{in} = K_{dn} = 1$ (wagi składowych są wtedy takie same).

$$K_p = N_e / N_u, \quad K_i = N_\rho / N_u, \quad K_d = N_{\dot{e}} / N_u \rightarrow e_n = N_e e, \quad \rho_n = N_\rho \rho, \quad \dot{e}_n = N_{\dot{e}} \dot{e}, \quad u_n = N_u u,$$

Przykład: Rozmyty regulator PD (Mamdani'ego):

Rys. Schemat rozmytego regulatora PD: liniowa część dynamiczna + rozmyta część statyczna



Znormalizowane przedziały zmienności:

$$-50 \leq e_n, \dot{e}_n \leq 50, -10 \leq u_n \leq 10$$

Po 6 symetrycznych trójkątnych funkcji przynależności dla zmiennych wejściowych.

Wartości zmiennych w k -tym kroku: $e_n(k)=27, \dot{e}_n(k)=14$

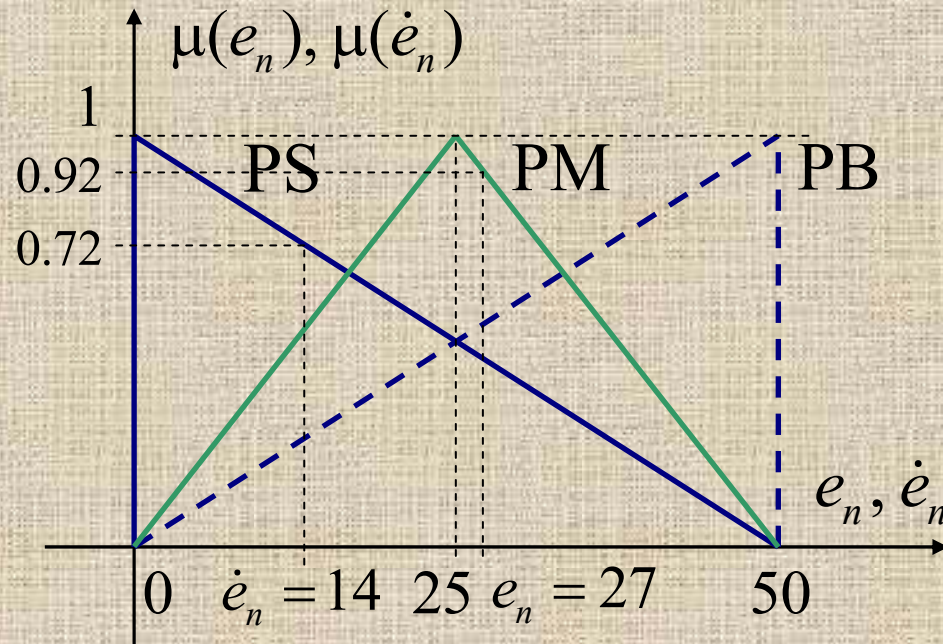
Rys. Baza reguł rozmytego regulatora PD

				\dot{e}_n		
PB	<small>PZ</small> NZ	PS	PM	PM	PB	PB
PM	NS	<small>PZ</small> NZ	PS	PM	PM	PB
PS	NM	NS	<small>PZ</small> NZ	PS	PM	PM
NS	NM	NM	NS	<small>PZ</small> NZ	PS	PM
NM	NB	NM	NM	NS	<small>PZ</small> NZ	PS
NB	NB	NB	NM	NM	NS	<small>PZ</small> NZ
	NB	NM	NS	PS	PM	PB
						e_n

Zaznaczone reguły:

$R^{(A)}$: if (e_n is **PM**) and (\dot{e}_n is **PS**) then (u_n is PM)

$R^{(B)}$: if (e_n is **PS**) and (\dot{e}_n is **PM**) then (u_n is PM)



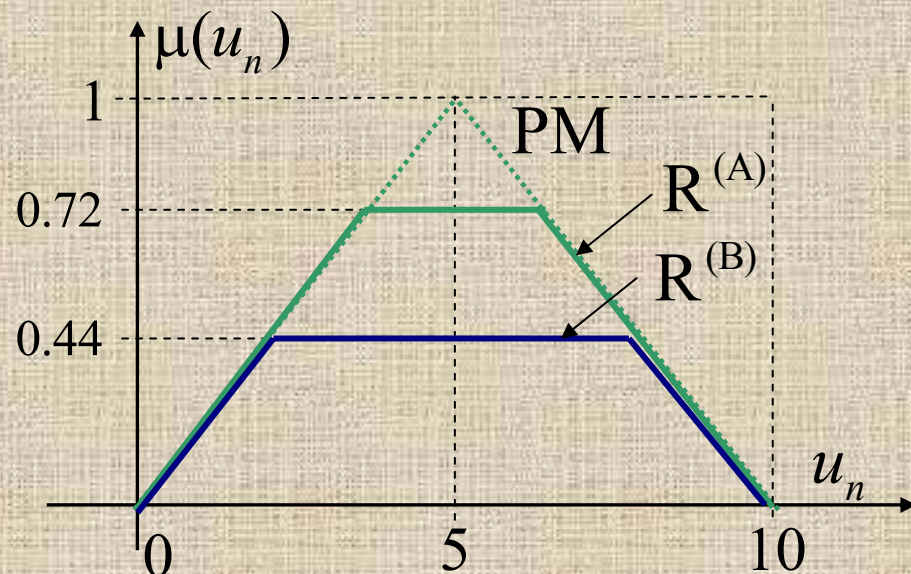
$R^{(A)}$:

$$\mu_{PM}(e_n = 27) = \frac{50 - 27}{50 - 25} = 0.92$$

$$\mu_{PS}(\dot{e}_n = 14) = \frac{50 - 14}{50 - 0} = 0.72$$

Rozmyte **AND**:

$$\min(\mu_{PM}(e_n), \mu_{PS}(\dot{e}_n)) = 0.72$$

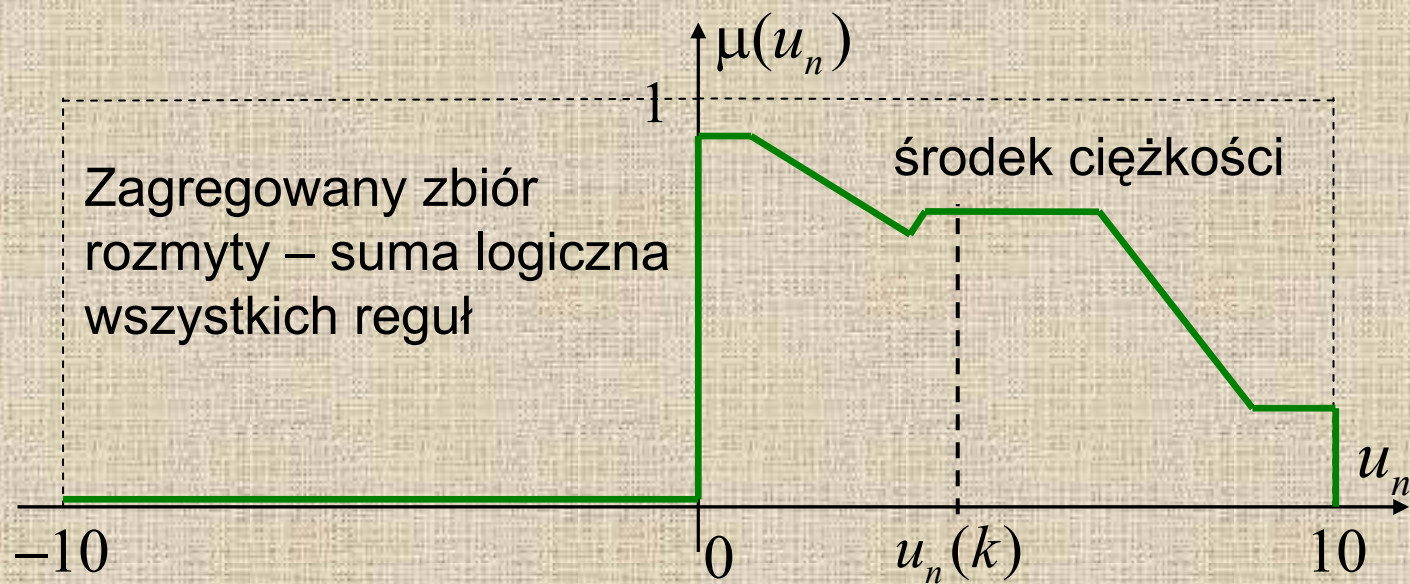


$R^{(B)}$:

$$\mu_{PS}(e_n = 27) = \frac{50 - 27}{50 - 0} = 0.46$$

$$\mu_{PM}(\dot{e}_n = 14) = \frac{25 - 14}{25 - 0} = 0.44$$

$$\min(\mu_{PM}(e_n), \mu_{PS}(\dot{e}_n)) = 0.44$$



Rys. Defuzyfikacja – metoda centroidów

Tabela. Stabelaryzowane zdyskretyzowane funkcje przynależności

Przedział	e_n is PS	e_n is PM	\dot{e}_n is PS	\dot{e}_n is PM	Przedział	u_n is PM
(-50, -45]	0	0	0	0	(-10, -9]	0
(-45, -40]	0	0	0	0	(-9, -8]	0
(-40, -35]	0	0	0	0	(-8, -7]	0
(-35, -30]	0	0	0	0	(-7, -6]	0
(-30, -25]	0	0	0	0	(-6, -5]	0
(-25, -20]	0	0	0	0	(-5, -4]	0
(-20, -15]	0	0	0	0	(-4, -3]	0
(-15, -10]	0	0	0	0	(-3, -2]	0
(-10, -5]	0	0	0	0	(-2, -1]	0
(-5, 0]	0	0	0	0	(-1, 0]	0
(0, 5]	1	0	1	0	(0, 1]	0
(5, 10]	0.88	0	0.88	0	(1, 2]	0
(10, 15]	0.78	0.2	0.78	0.2	(2, 3]	0.2
(15, 20]	0.65	0.6	0.65	0.6	(3, 4]	0.6
(20, 25]	0.55	0.8	0.55	0.8	(4, 5]	0.8
(25, 30]	0.44	1	0.44	1	(5, 6]	1
(30, 35]	0.32	0.8	0.32	0.8	(6, 7]	0.8
(35, 40]	0.21	0.6	0.21	0.6	(7, 8]	0.6
(40, 45]	0.11	0.2	0.11	0.2	(8, 9]	0.2
(45, 50]	0	0	0	0	(9, 10]	0

Wnioskowanie wg reguły $R^{(A)}$ (obciążenie na poziomie 0.78):

$$\mu_{PM}^{(A)}(u_n) = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.2, 0.6, 0.78, 0.78, 0.78, 0.6, 0.2, 0)$$

Wnioskowanie wg reguły $R^{(B)}$ (obciążenie na poziomie 0.2):

$$\mu_{PM}^{(B)}(u_n) = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0)$$

Agregacja (tylko dla reguł A i B):

$$\max[\mu_{PM}^{(A)}(u_n), \mu_{PM}^{(B)}(u_n)] = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.2, 0.6, 0.78, 0.78, 0.78, 0.6, 0.2, 0)$$

Defuzyfikacja: wyznaczenie wartości sygnału wyjściowego regulatora:

$$u_n = \frac{(2.5) \cdot 0.2 + (3.5) \cdot 0.6 + (4.5) \cdot 0.78 + (5.5) \cdot 0.78 + (6.5) \cdot 0.78 + (7.5) \cdot 0.6 + (8.5) \cdot 0.2}{0.2 + 0.6 + 0.78 + 0.78 + 0.78 + 0.6 + 0.2} = 5.5$$

gdzie liczby w nawiasach to środki odpowiednich przedziałów z tabeli, tzn.

2.5 jest środkiem przedziału (2,3)

3.5 jest środkiem przedziału (3,4) itd.

Regulator typu Takagi-Sugeno

W następnikach (konkluzjach) reguł występują zwykłe funkcje zmiennych:

$R^{(i)}$: if $(x_1 \text{ is } WX_1^{(i)})$ and ... $(x_n \text{ is } WX_n^{(i)})$ then $u_i = f_i(x_1, x_2, \dots, x_n)$

przynależności: $\mu_{WX_1}^{(i)}(x_1) \quad \dots \quad \mu_{WX_n}^{(i)}(x_n)$

$\mathbf{x} = [x_1 \ x_2 \ \dots \ x_n]^T$ - znormalizowane sygnały wejściowe

Waga składowej u_i sygnału wyjściowego i -tej reguły (AND-min):

$$w^{(i)} = \min[\mu_{WX_1}^{(i)}(x_1), \dots, \mu_{WX_n}^{(i)}(x_n)]$$

Całkowity znormalizowany wyjściowy
sygnał sterujący:

$$u = \frac{\sum_{i=1}^m w^{(i)} u_i}{\sum_{i=1}^m w^{(i)}}$$

Funkcje $f_i = \text{const}$ nazywa się *singletonami*. Jeżeli f_i są liniowymi funkcjami \mathbf{x} ,

$$u_i = f_i(x_1, x_2, \dots, x_n) = a_1 x_1 + a_2 x_2 + \dots + a_n x_n$$

to otrzymujemy *lokalne regulatory liniowe*.

Model Sugeno nie wymaga defuzyfikacji i jest prosty obliczeniowo.

Regulacja ślizgowa (sliding mode control)

Rozważamy nieliniowy układ dynamiczny (obiekt sterowania):

$$x^{(n)}(t) = f(x, \dot{x}, \dots, x^{(n-1)}; t) + u(t)$$

dla którego pochodne tworzą wektor stanu $\mathbf{x} = [x, \dot{x}, \dots, x^{(n-1)}]^T$, a u jest sterowaniem. Można zapisać:

$$x^{(n)}(t) = f(\mathbf{x}, t) + u(t)$$

Przyjmujemy, że funkcja \hat{f} modelu układu jest obarczona niepewnością Δf :

$$f(\mathbf{x}, t) = \hat{f}(\mathbf{x}, t) + \Delta f(\mathbf{x}, t)$$

Zakładamy, że niepewność Δf modelu jest ograniczona od góry przez znaną funkcję:

$$|\Delta f(\mathbf{x}, t)| \leq F(\mathbf{x}, t)$$

Można przyjąć, że Δf obejmuje też ew. ograniczone zakłócenia $z(\mathbf{x}, t)$ działające na obiekt. Celem sterowania jest śledzenie przez wektor stanu \mathbf{x} wartości pożądanej (zadanej): $\mathbf{w} = [w, \dot{w}, \dots, w^{(n-1)}]^T$

Uchyb sterowania definiujemy jako wektor: $\mathbf{e}(t) = \mathbf{x}(t) - \mathbf{w}(t)$

Zdefiniujemy tzw. uchyb uogólniony s jako funkcję liniową współrzędnych wektora \mathbf{e} (wyraz w nawiasie jest operatorem różniczkowania):

$$s(\mathbf{x}, t) = \left(\frac{d}{dt} + \lambda \right)^{n-1} e, \quad \lambda \geq 0$$

Powierzchnia $s(\mathbf{x}, t) = 0$ jest *powierzchnią ślizgową* w przestrzeni stanu.

Zadaniem sterowania jest sprowadzenie wektora stanu na powierzchnię ślizgową i utrzymanie go na niej, ponieważ na powierzchni uchyb uogólniony

$$s = \left(\frac{d}{dt} + \lambda \right)^{n-1} e = 0$$

Ze względu na uchyb $e = \mathbf{x} - \mathbf{w}$ jest to stabilne równanie różniczkowe rzędu $(n-1)$. Wszystkie pierwiastki jego równania charakterystycznego są równe $-\lambda$. *Na powierzchni ślizgowej $s=0$ uchyb sterowania e maleje asymptotycznie do zera z szybkością ślizgania proporcjonalną do parametru λ .*

Aby osiągnąć powierzchnię ślizgową dla warunku początkowego $e(0) \neq 0$ powinien być spełniony warunek (dostateczny):

$$\frac{1}{2} \frac{d}{dt} s^2(\mathbf{x}, t) \leq -\eta |s|, \quad \eta > 0$$

Po przekształceniu: $\frac{1}{2} \frac{d}{dt} s^2(\mathbf{x}, t) = s \cdot \dot{s} = \dot{s} |s| \operatorname{sgn}(s)$

Stąd *warunek ślizgania* (utrzymywania się na powierzchni ślizgowej):

$$\dot{s} \cdot \operatorname{sgn}(s) \leq -\eta$$

Warunek ten jest równoważny warunkowi Lapunowa stabilności układu

$$\dot{s}(\mathbf{x}, t) = 0$$

dla którego funkcja Lapunowa $V(\mathbf{x}) = s^2(\mathbf{x}, t)$.

Przy spełnieniu warunku ślizgania powierzchnia ślizgowa jest osiągana w skończonym czasie $t_{reach} \leq |s(t=0)| / \eta$.

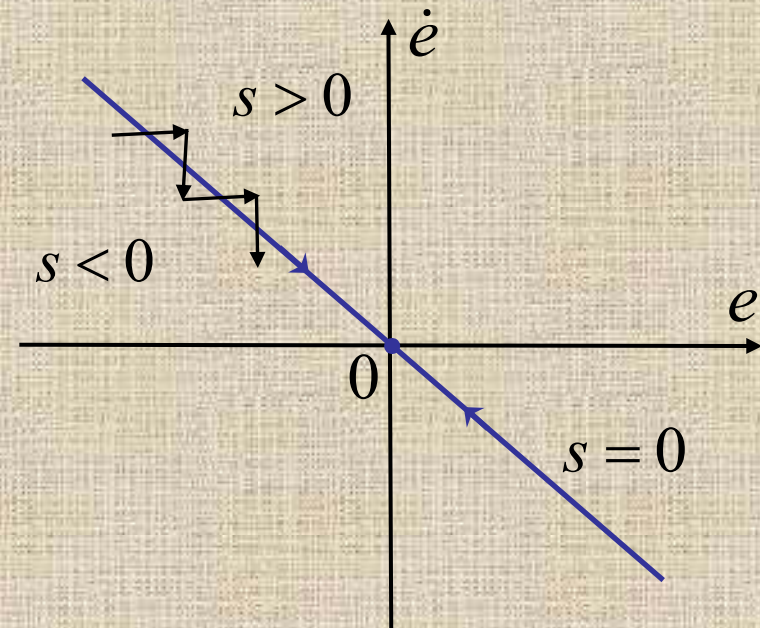
Sterowanie ślizgowe sprowadza zadanie sterowania do stabilizacji pierwszego rzędu względem uchybu uogólnionego s .

Osiągnięcie pożądanego wektora stanu \mathbf{w} można podzielić na dwie fazy:

- 1) fazę dochodzenia do powierzchni ślizgowej,
- 2) fazę ślizgania (utrzymywania się na powierzchni ślizgowej).



Rys. Przebieg regulacji ślizgowej w przypadku *idealnego* ślizgania



Rys. Rzeczywisty przebieg fazy ślizgania

Metoda funkcji Lapunowa - przypomnienie

Def. Funkcję $V(\mathbf{x})$ wektora stanu x nazywa się *funkcją Lapunowa* w obszarze D , jeżeli spełnia one następujące warunki:

1. $V(\mathbf{x})$ jest *dodatnio* określona w obszarze D :

$$V(\mathbf{x}) > 0 \text{ dla } \mathbf{x} \neq \mathbf{0} \text{ oraz } V(\mathbf{0}) = 0.$$

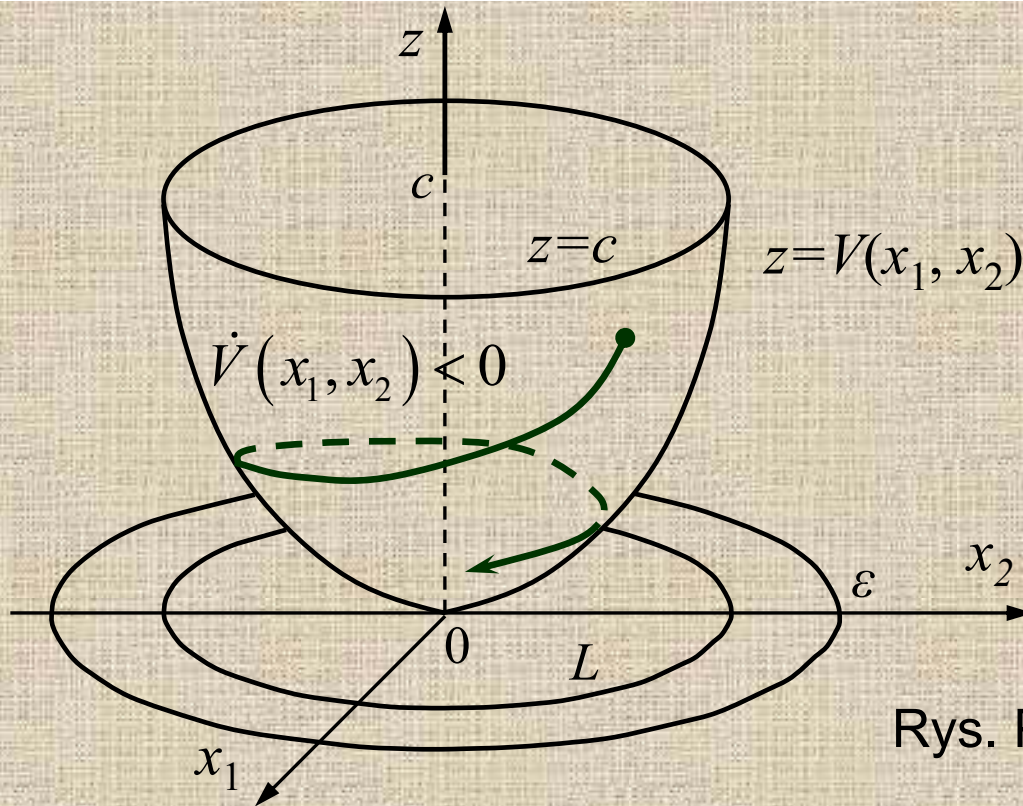
2. Pochodna względem czasu funkcji $V(\mathbf{x})$ jest *ujemnie* określona w D :

$$\dot{V}(\mathbf{x}) < 0 \text{ dla } \mathbf{x} \neq \mathbf{0}, \quad \dot{V}(\mathbf{x}) = 0 \text{ dla } \mathbf{x} = \mathbf{0}.$$

3. $V(\mathbf{x}) \rightarrow \infty$ dla $|\mathbf{x}|^2 = (x_1^2 + x_2^2 + \dots + x_n^2) \rightarrow \infty$.

4. $V(\mathbf{x})$ jest ciągła i ma ciągłe pierwsze pochodne $\partial V(\mathbf{x})/\partial x_i$.

Twierdzenie: Układ nieliniowy $\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x})$, który ma punkt równowagi $\mathbf{x} = \mathbf{0}$, jest **stabilny asymptotycznie** w obszarze D zawierającym $\mathbf{x} = \mathbf{0}$, jeżeli można w tym obszarze dobrać dla niego funkcję Lapunowa $V(\mathbf{x})$, której pochodna względem czasu wzdłuż trajektorii fazowej jest ujemnie określona.



Rys. Przykładowa funkcja Lapunowa

Pochodną względem czasu funkcji Lapunowa $V(\mathbf{x})$ można przedstawić w postaci:

$$\dot{V}(\mathbf{x}) = \frac{\partial V(\mathbf{x})}{\partial x_1} \frac{dx_1}{dt} + \dots + \frac{\partial V(\mathbf{x})}{\partial x_n} \frac{dx_n}{dt} =$$

$$= \left(\frac{d\mathbf{x}}{dt} \right)^T \cdot \text{grad } V(\mathbf{x}) = \mathbf{F}^T(\mathbf{x}) \cdot \text{grad } V(\mathbf{x})$$

$$\text{grad } V(\mathbf{x}) = \begin{bmatrix} \frac{\partial V(\mathbf{x})}{\partial x_1} \\ \vdots \\ \frac{\partial V(\mathbf{x})}{\partial x_n} \end{bmatrix}$$

Wyprowadzenie prawa sterowania spełniającego warunek ślizgania

Bez utraty ogólności można rozważyć przypadek układu *stacjonarnego* rzędu $n=2$. Zakładamy, że zmienne stanu x, \dot{x} są *mieralne*.

$$\ddot{x}(t) = f(\mathbf{x}) + u(t), \quad \mathbf{x} = [x, \dot{x}]^T$$

Powierzchnia ślizgowa jest w tym przypadku *linią przełączającą*:

$$s = \left(\frac{d}{dt} + \lambda \right) e = \dot{e} + \lambda e = 0, \quad \lambda \geq 0$$
$$\dot{e} + \lambda e = 0 \Rightarrow e \rightarrow 0$$

Równanie dynamiki ślizgania (uchyb $e=x-w$):

$$\dot{s} = \ddot{e} + \lambda \dot{e} = \ddot{x} - \ddot{w} + \lambda \dot{e} = f + u - \ddot{w} + \lambda \dot{e} = 0$$

Jeżeli prawdziwa funkcja układu f jest znana, to stabilizację $\dot{s} = 0$ gwarantuje tzw. sterowanie równoważne:

$$u_{eq} = -f + \ddot{w} - \lambda \dot{e}$$

W przypadku nieznanego układu za funkcję f podstawia się jej model (estymator) \hat{f} , co prowadzi do sterowania

$$u_c = -\hat{f}(\mathbf{x}) + \ddot{w} - \lambda \dot{e}$$

Aby skompensować niepewność spowodowaną zastąpieniem prawdziwej funkcji układu przez jej model, do sterowania dodaje się składową nieciągłą (przełączającą - *switching*):

$$u_s = -K(\mathbf{x}) \operatorname{sgn}(s)$$

której amplituda spełnia warunek:

$$K(\mathbf{x}) = F(\mathbf{x}) + \eta, \quad \eta > 0$$

Pełne prawo sterowania ślizgowego z tzw. *twardym przełączaniem*:

$$u = u_c + u_s = -\hat{f}(\mathbf{x}) + \ddot{w} - \lambda \dot{e} - K(\mathbf{x}) \operatorname{sgn}(s)$$

$u_c = -\hat{f}(\mathbf{x}) + \ddot{w} - \lambda \dot{e}$ – składowa kompensacyjna sterowania (ciągła)

$u_s = -K(\mathbf{x}) \operatorname{sgn}(s)$ – składowa przełączająca sterowania, zależna od niepewności modelu obiektu

Wykazanie, że wyprowadzone prawo spełnia warunek ślizgania:

$$\begin{aligned} \frac{1}{2} \frac{d}{dt} s^2(\mathbf{x}) &= s \cdot \dot{s} = (f + u - \ddot{w} + \lambda \dot{e}) \cdot s = \\ &= [f + [-\hat{f} + \ddot{w} - \lambda \dot{e} - K(\mathbf{x}) \operatorname{sgn}(s)] - \ddot{w} + \lambda \dot{e}] \cdot s = \\ &= (f - \hat{f})s - K(\mathbf{x}) \operatorname{sgn}(s)s = \Delta f(\mathbf{x})s - K(\mathbf{x}) |s| \end{aligned}$$

Korzystając z początkowego założenia o ograniczonej niepewności modelu:

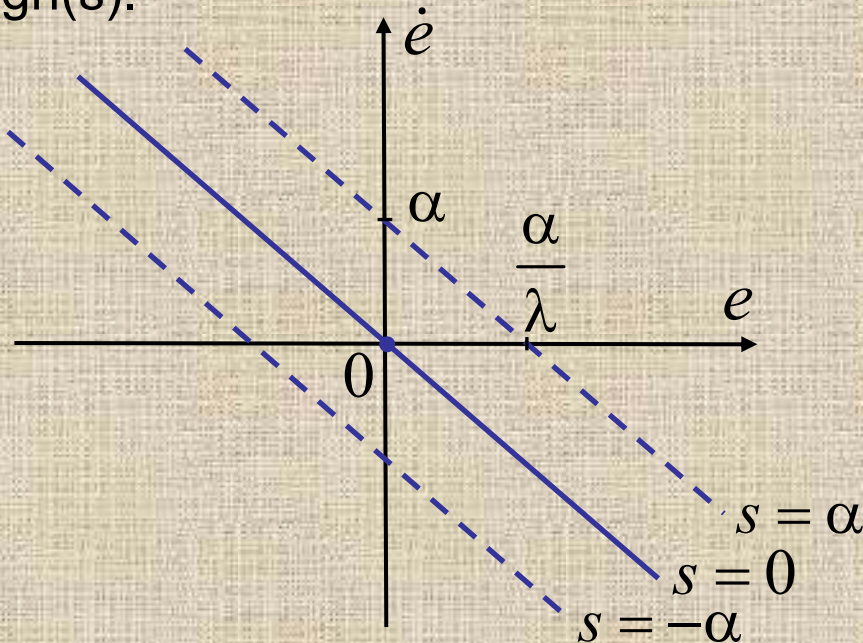
$$|\Delta f(\mathbf{x})| \leq F(\mathbf{x})$$

stwierdzamy spełnienie warunku ślizgania *pomimo niedokładnego modelu*:

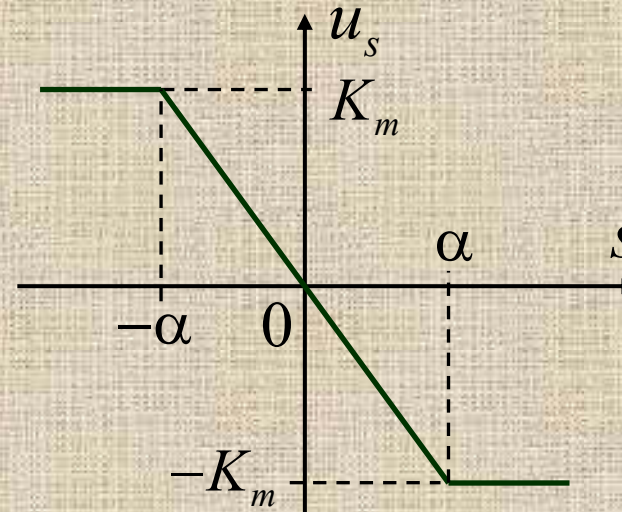
$$\frac{1}{2} \frac{d}{dt} s^2(\mathbf{x}) \leq Fs - (F + \eta) |s| \leq -\eta |s|$$

Sterowanie ślizgowe jest sterowaniem o zmiennej strukturze: przy przejściu przez linię ślizgania następuje skokowa zmiana składowej nieciągłej. Wysoki poziom aktywności sterowania (tzw. *chattering*) jest niepożądany, bo powoduje dodatkowe obciążenie elementów wykonawczych i straty mocy.

W celu uniknięcia podczas ślizgania twardych przełączeń, sterowanie jest wygładzone przez wprowadzenie wokół powierzchni ślizgowej $s=0$ warstwy granicznej o szerokości 2α , w której funkcja $\text{sat}(s/\alpha)$ zastępuje funkcję $\text{sgn}(s)$.



Rys. Warstwa graniczna wokół powierzchni ślizgowej



Rys. Wykres wygładzonej składowej przełączającej sterowania

$$u_s = -K_m \text{sat}\left(\frac{s}{\alpha}\right)$$

Wartość α/λ charakteryzuje *precyzję śledzenia* (uchyb statyczny dla $\dot{e}=0$) związany z aproksymacją skokowej składowej przełączającej sterowania ślizgowego przez ciągłą funkcję nasycenia.

Prawo miękkiego sterowania ślizgowego z warstwą graniczną ma postać:

$$u_{SMC} = -\hat{f}(\mathbf{x}) + \ddot{w} - \lambda \dot{e} - K_m \operatorname{sat}\left(\frac{s}{\alpha}\right)$$

gdzie stały parametr K_m jest górnym ograniczeniem funkcji niepewności przesuniętej o η :

$$K_m \geq K(\mathbf{x}) = F(\mathbf{x}) + \eta$$

Nachylenie linii ślizgowej λ jest ważnym parametrem sterowania. Jest to pulsacja graniczna reprezentująca ograniczenie dolne częstotliwości nie ujętych w modelu dynamiki układu $\hat{f}(\mathbf{x})$.

Wartość λ wyznacza zakres częstotliwości, w którym model układu jest adekwatny. Poza tym zakresem dynamika może być determinowana przez niemodelowane elementy układu.

Zwiększenie λ powoduje zwiększenie precyzji śledzenia, ale może doprowadzić do „wzbudzenia” układu sterowania (ciągłe przełączanie sterowania i przeskakiwanie przez linię ślizgową).

Reguły wyboru wartości λ

1) Należy nie przekraczać najmniejszej pulsacji λ_0 modów nie ujętych w modelu (np. przy celowym upraszczaniu modelu analitycznego)

$$\lambda \leq \lambda_0$$

2) Należy nie przekraczać pulsacji związanej z największym opóźnieniem (lub stałą czasową) T_m nie ujętym przez model

$$\lambda \leq \lambda_T = \frac{1}{3T_m}$$

3) Powinno być kilka razy mniejsze od częstości próbkowania f_s

$$\lambda \leq \lambda_s = \frac{1}{5T_s} \quad \text{Stosowana zależność: } \lambda \leq \frac{1}{2(T_m + T_s)}$$

Kroki projektowania regulacji ślizgowej z warstwą graniczną

Step 1: Oszacowanie $\lambda \leq \min(\lambda_0, \lambda_T, \lambda_s)$

Step 2: Oszacowanie min. K_m spełniającego warunek $K_m \geq F(\mathbf{x}, t) + \eta$

Step 3: Wyznaczenie szerokości warstwy granicznej $\alpha = K_m / \lambda$

Przypadek ogólniejszy

Sterowanie ślizgowe można zaprojektować dla bardziej ogólnej funkcji opisującej obiekt (wzory dla układu rzędu $n=2$):

$$\ddot{\mathbf{x}}(t) = f(\mathbf{x}) + \hat{g}(\mathbf{x})[u(t) + \Delta g(\mathbf{x}, u)], \quad \mathbf{x} = [x, \dot{x}]^T$$

gdzie $\hat{g}(\mathbf{x}, t) > 0$ jest znaną funkcją. Zakłada się, że znane są górne ograniczenia niepewności:

$$|\Delta f(\mathbf{x}) = f(\mathbf{x}) - \hat{f}(\mathbf{x})| \leq F(\mathbf{x}), \quad |\Delta g(\mathbf{x}, u)| \leq G(\mathbf{x}, u)$$

W tym przypadku składowa kompensacyjna sterowania ślizgowego ma formę:

$$u_c = \frac{-\hat{f}(\mathbf{x}) + \ddot{w} - \lambda \dot{e}}{\hat{g}(\mathbf{x})}$$

a kompensująca niepewności składowa przełączająca:

$$u_s = -K(\mathbf{x}) \operatorname{sgn}(s) \quad \text{gdzie} \quad K(\mathbf{x}) = \frac{F(\mathbf{x}) + \eta}{\hat{g}(\mathbf{x})} + G(\mathbf{x}, u), \quad \eta > 0$$

Całkowite sterowanie:

$$u = u_c + u_s = \frac{-\hat{f}(\mathbf{x}) + \ddot{w} - \lambda \dot{e}}{\hat{g}(\mathbf{x})} - K(\mathbf{x}) \operatorname{sgn}(x)$$

Sterowanie to spełnia warunek ślizgania, bo:

$$\begin{aligned}
 \frac{1}{2} \frac{d}{dt} s^2(\mathbf{x}) &= s \cdot \dot{s} = [f + \hat{g}(u + \Delta g) - \ddot{w} + \lambda \dot{e}] \cdot s = \\
 &= \left[f + \hat{g} \left(\frac{-\hat{f} + \ddot{w} - \lambda \dot{e}}{\hat{g}} - K \operatorname{sgn}(s) \right) + \hat{g} \Delta g - \ddot{w} + \lambda \dot{e} \right] \cdot s = \\
 &= (f - \hat{f})s - \hat{g} \left(\frac{F + \eta}{\hat{g}} + G \right) \operatorname{sgn}(s)s + \hat{g} \Delta g \cdot s = \\
 &= \Delta f \cdot s - F |s| - \eta |s| - \hat{g}(G |s| - \Delta g \cdot s) \leq -\eta |s|
 \end{aligned}$$

Układy wyższych rzędów

Układy sterowania z obiektami wyższych rzędów różnią się jedynie ciągłą składową kompensacyjną sterowania. Składowa przełączająca jest taka sama jak wyprowadzona dla obiektu drugiego rzędu.

W przypadku $n=3$:

$$\ddot{x}(t) = f(\mathbf{x}) + \hat{g}(\mathbf{x})[u(t) + \Delta g(\mathbf{x}, u)], \quad \mathbf{x} = [x, \dot{x}, \ddot{x}]^T$$

Powierzchnia ślizgowa: $s = \left(\frac{d}{dt} + \lambda \right)^2 \cdot e = \ddot{e} + 2\lambda\dot{e} + \lambda^2 e = 0, \quad \lambda \geq 0$

$$\dot{s} = \ddot{x} - \ddot{w} + 2\lambda\ddot{e} + \lambda^2\dot{e} = f + \hat{g}u + \hat{g}\Delta g - \ddot{w} + 2\lambda\ddot{e} + \lambda^2\dot{e} = 0$$

Sterowanie równoważne (bez niepewności): $u_{eq} = \frac{-f + \ddot{w} - 2\lambda\ddot{e} - \lambda^2\dot{e}}{\hat{g}}$

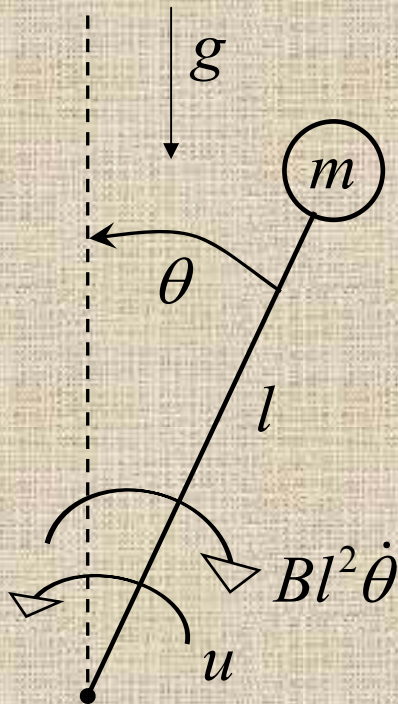
a sterowanie pełne:

$$u = \frac{-\hat{f} + \ddot{w} - 2\lambda\ddot{e} - \lambda^2\dot{e}}{\hat{g}} - K \operatorname{sgn}(s)$$

gdzie: $K = \frac{F + \eta}{\hat{g}} + G$

Przykład: Sterowanie ślizgowe odwróconego wahadła z tarciem

$$ml^2\ddot{\theta} = -mgl \sin \theta - Bl^2\dot{\theta} + u$$



$$\ddot{\theta} = -a \sin \theta - b\dot{\theta} + cu, \quad a = \frac{g}{l}, \quad b = \frac{B}{m}, \quad c = \frac{1}{ml^2}$$

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -a \sin x_1 - bx_2 + cu \end{cases} \quad \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix}$$

$$f(\mathbf{x}) = -a \sin x_1 - bx_2, \quad \hat{f}(x) = -\hat{a} \sin x_1 - \hat{b}x_2$$

$$\Delta f(\mathbf{x}) = f(\mathbf{x}) - \hat{f}(\mathbf{x}) = -(a - \hat{a}) \sin x_1 - (b - \hat{b})x_2$$

$$\hat{g}(\mathbf{x}) = c$$

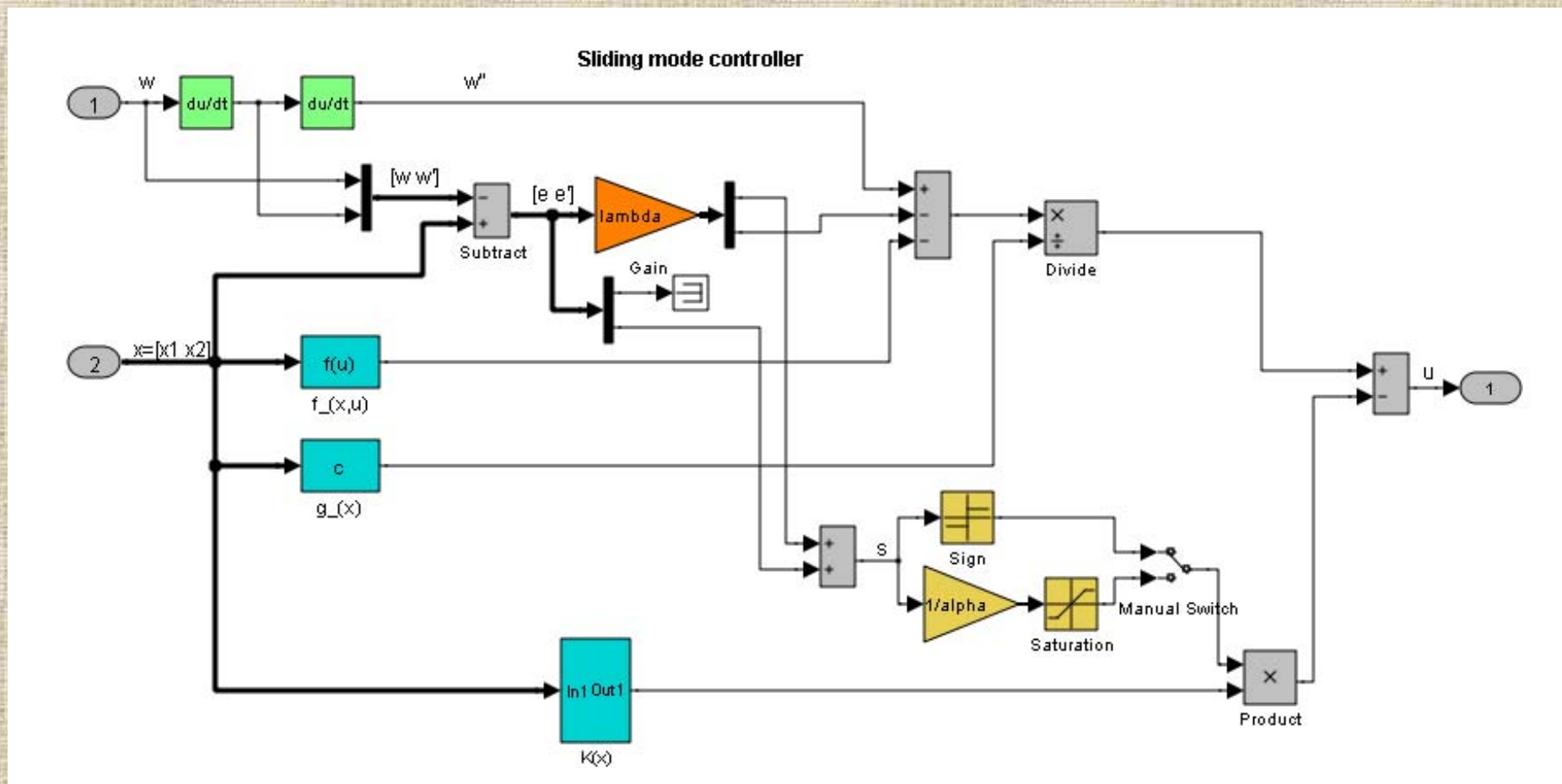
Niech niepewność: $\Delta g(\mathbf{x}) = \delta \sin x_1$

Górne ograniczenia $F(\mathbf{x}) = k_1 + k_2 |x_2|$, $k_1 \geq |a - \hat{a}|$, $k_2 \geq |b - \hat{b}|$

niepewności:

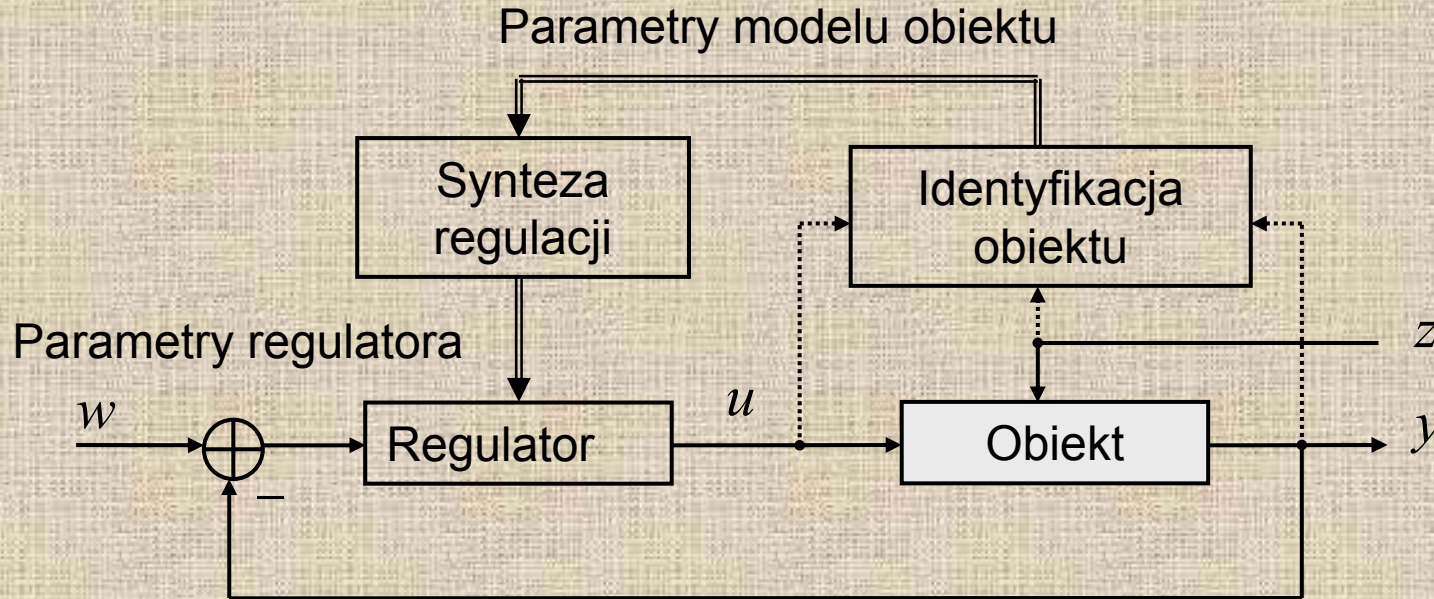
$$G(\mathbf{x}) = k_3, \quad k_3 \geq |\delta|$$

Sterowanie:
$$u = \frac{-\hat{f} + \ddot{w} - \lambda \dot{e}}{\hat{g}} - K \operatorname{sgn} s, \quad K = \frac{F + \eta}{\hat{g}} + G$$



Rys. Schemat Simulinka regulatora ślizgowego

Regulator adaptacyjny samonastrajający (self-tuning controller)



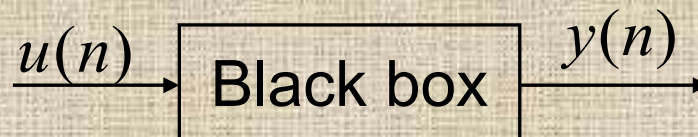
Rys. Schemat układu regulacji z zamkniętym obwodem adaptacji - regulator samonastrajający się (*Self-Tuning Controller*)

Identyfikacja układu dynamicznego

Identyfikacja jest podejściem do budowania modelu dynamiki układu (podlegającego sterowaniu) polegającym na analizie zależności pomiędzy jego sygnałami wejściowymi i wyjściowymi. Stosuje się ją, kiedy prawa rządzące procesem są nieznane albo zbyt skomplikowane, aby wykorzystać je do konstrukcji modelu sterowania (metoda „czarnej skrzynki”). Dane eksperymentalne wykorzystuje się do określenia (estymacji) nieznanymi parametrów modelu dynamiki rzeczywistego procesu.

Ze względu na charakter danych pomiarowych matematyczne modele identyfikacji są modelami dyskretnymi. Rozwiązanie problemu estymacji parametrów wymaga:

- znajomości danych wejściowych $u(n)$ i wyjściowych $y(n)$ z procesu,
- określenia klasy modeli identyfikacji,
- kryterium optymalnego doboru parametrów; metoda estymacji zależy od przyjętego modelu i sposobu prowadzenia obliczeń: *on-* lub *off-line*



Regresja liniowa

Zadanie polega na dopasowaniu do danych pomiarowych $y(n)$, $n=1,2,\dots,N$, (n - dyskretny czas) modelu postaci:

$$\hat{y}(n) = \theta_1 \phi_1(n) + \theta_2 \phi_2(n) + \dots + \theta_k \phi_k(n) = \phi^T(n)\theta$$

gdzie:

$$\phi(n) = \begin{bmatrix} \phi_1(n) \\ \phi_2(n) \\ \vdots \\ \phi_k(n) \end{bmatrix} \quad \text{- znane funkcje (regresory),}$$

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_k \end{bmatrix} \quad \text{- nieznane parametry,}$$

Estymacji dokonuje się stosując kryterium najmniejszych kwadratów (*LS – Least Squares*):

$$V(\bar{\theta}) = \frac{1}{2} \sum_{n=1}^N \varepsilon^2(n)$$

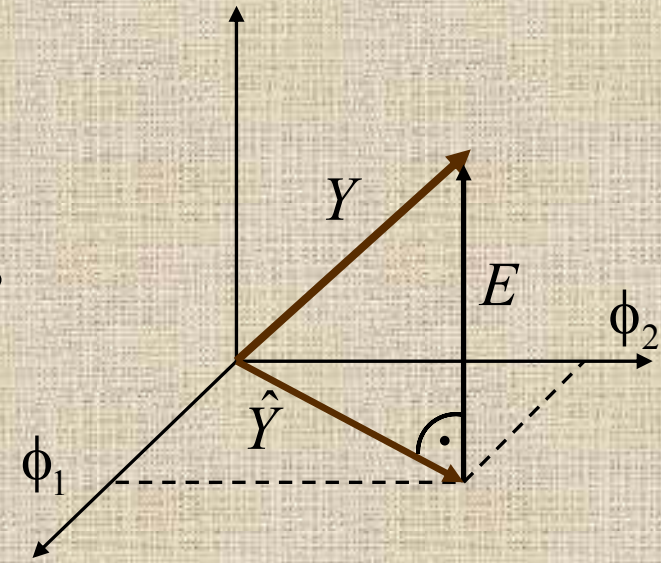
gdzie: $\varepsilon(n) = y(n) - \hat{y}(n)$ - błędy dopasowania (błędy predykcji)

$\bar{\theta}$ - estymata parametrów, N - liczba danych pomiarowych.

W notacji wektorowej: $E = Y - \hat{Y} = Y - \Phi \bar{\theta}$

$$Y = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(N) \end{bmatrix}, \quad \Phi = \begin{bmatrix} \phi^T(1) \\ \phi^T(2) \\ \vdots \\ \phi^T(N) \end{bmatrix}, \quad E = \begin{bmatrix} \varepsilon(1) \\ \varepsilon(2) \\ \vdots \\ \varepsilon(N) \end{bmatrix},$$

$$V(\bar{\theta}) = \frac{1}{2} E^T E = \frac{1}{2} \|E\|^2$$



Minimum funkcji strat uzyskuje się obliczając jej pochodną i przyrównując ją do zera. Otrzymuje się w ten sposób równanie *liniowe względem parametrów*, które można rozwiązać analitycznie.

$$\frac{\partial V(\bar{\theta})}{\partial \bar{\theta}} = 0 \quad \rightarrow \quad \bar{\theta}_{opt} = \hat{\theta} = (\Phi^T \Phi)^{-1} \Phi^T Y = P \sum_{n=1}^N \phi(n) y(n) \quad (*)$$

gdzie $P = (\Phi^T \Phi)^{-1}$.

Warunkiem istnienia optymalnego rozwiązania jest odwracalność macierzy $\Phi^T \Phi$ (tzw. warunek wystarczającego pobudzenia).

Interpretacja statystyczna metody najmniejszych kwadratów

Założmy, że dane pomiarowe $y(n)$ są generowane przez układ

$$y(n) = \phi^T(n)\theta_0 + e(n)$$

gdzie: θ_0 - wektor "prawdziwych" parametrów,
 $\{e(n), n=1,2,\dots\}$ - ciąg *niezależnych* zmiennych losowych o zerowej średniej i wariancji σ^2 (szum biały).

Wtedy estymator LS (*) ma następujące właściwości:

- 1) $E[\hat{\theta}] = \theta_0$ (nieobciążona estymata parametrów)
- 2) $\text{var } \hat{\theta} = \sigma^2 (\Phi^T \Phi)^{-1}$ (w typowych warunkach $\text{var } \hat{\theta}$ maleje jak $1/N$)
- 3) $\hat{\sigma}^2 = 2V(\hat{\theta}, N) / (N - n)$ (nieobciążona estymata wariancji szumu)

Z powyższej interpretacji wynika, że metoda LS daje poprawne wyniki jedynie wtedy, kiedy prawdziwym modelem (z którego uzyskuje się pomiary) jest model ARX lub ARMAX z $C(q)=q^m$ (zakłócenia nie są ze sobą skorelowane).

Uwaga ogólna: metoda LS jest wrażliwa na obecność w danych trendów i szumy kolorowe – wywołują obciążenie estymacji. Trendy, w tym wartość średnią należy usunąć (*detrending*).

Rekurencyjne obliczanie estymaty LS

W wielu przypadkach, np. w regulatorach adaptacyjnych, pomiary $y(n)$ i identyfikacja są wykonywane *on-line*. Obliczanie kolejnych estymat powinno wtedy zorganizowane w taki sposób, aby wynik uzyskany dla N pomiarów mógł być wykorzystany do wyznaczenia estymaty dla $N+1$ pomiarów.

Po uzyskaniu dodatkowego pomiaru:

$$\Phi(N+1) = \begin{bmatrix} \Phi(N) \\ \phi^T(N+1) \end{bmatrix}, \quad Y(N+1) = \begin{bmatrix} Y(N) \\ y(N+1) \end{bmatrix}$$

Wykorzystując wzór (*) oraz zależność:

$$P^{-1}(N+1) = P^{-1}(N) + \phi(N+1)\phi^T(N+1)$$

można wyprowadzić następujące zależności na rekurencyjną estymatę najmniejszych kwadratów (RLS - *Recursive Least Squares*):

$$\hat{\theta}(N+1) = \hat{\theta}(N) + K(N) \left[y(N+1) - \phi^T(N+1)\hat{\theta}(N) \right],$$

$$K(N) = P(N+1)\phi(N+1) = P(N)\phi(N+1) \left[1 + \phi^T(N+1)P(N)\phi(N+1) \right]^{-1},$$

$$P(N+1) = \left[I - K(N)\phi^T(N+1) \right] P(N).$$

Dla układów z jednym wyjściem odwracanie macierzy o wymiarze równym liczbie dodawanych pomiarów w drugim wzorze w ramce sprowadza się do dzielenia przez skalar $(1 + \phi^T P \phi)$.

Nowa estymata RLS jest obliczana przez dodanie do estymaty poprzedniej poprawki proporcjonalnej do *błędu predykcji*

$$\varepsilon(N+1) = y(N+1) - \hat{y}(N+1)$$

$$\hat{y}(N+1) = \phi^T(N+1)\hat{\theta}(N)$$

(wartość pomiaru w chwili $N+1$ przewidzianą przez model regresji liniowej na podstawie poprzedniej estymaty parametrów)

Składowe wektora $K(N)$ są czynnikami wagowymi poprawki.

Jeżeli N jest wystarczająco małe, to macierz $(\Phi^T \Phi)$ jest osobliwa. Oznacza to, że liczba pomiarów jest zbyt mała do estymacji k parametrów. Żeby uzyskać warunki początkowe do obliczeń wybiera się takie N_0 , że macierz $P(N_0)$ i pierwsza estymata $\hat{\theta}(N_0) = P(N_0)\Phi^T(N_0)y(N_0)$ są określone i prowadzi obliczenia dla $N \geq N_0$.

Często startuje się początkowym $P(0)=P_0$, gdzie P_0 jest macierzą dodatnio określoną, np. $P_0 = \alpha I$ (I - macierz jednostkowa), gdzie α jest dużą liczbą.

Estymacja parametrów układów zmiennych w czasie

W oryginalnej funkcji $V(\theta)$ metody LS wszystkie dane mają tę samą wagę.

Jeśli parametry układu zmieniają się w czasie zachodzi potrzeba eliminowania wpływu starych danych. Można to zrobić stosując funkcję strat z wagami malejącymi wstecz:

$$V(\bar{\theta}) = \frac{1}{2} \sum_{n=1}^N \lambda^{N-n} \left[y(n) - \phi^T(n) \bar{\theta} \right]^2$$

gdzie $\lambda < 1$ jest tzw. "współczynnikiem zapominania" (*forgetting factor*, typowo 0.96- 0.995). Rekursywna estymata LS przyjmuje postać:

$$\begin{aligned} \hat{\theta}(n+1) &= \hat{\theta}(n) + K(n) \left[y(n+1) - \phi^T(n+1) \hat{\theta}(n) \right], \\ K(n) &= \frac{P(n) \phi(n+1)}{\lambda + \phi^T(n+1) P(n) \phi(n+1)}, \\ P(n+1) &= \left[I - K(n) \phi^T(n+1) \right] \frac{P(n)}{\lambda}. \end{aligned}$$

W praktyce wykorzystuje się pomiary z okna czasowego o długości N . Wtedy:

$$V(\bar{\theta}) = \frac{1}{2} \sum_{n=L+1}^{L+N} \lambda^{N-n} \left[y(n) - \phi^T(n) \bar{\theta} \right]^2$$

Matematyczne modele identyfikacji

Ze względu na charakter danych pomiarowych matematyczne modele identyfikacji są modelami dyskretnymi, zwykle liniowymi względem parametrów. Uwzględniają również, że dane pomiarowe są obarczone błędami poprzez wprowadzenie do modeli przypadkowych zakłóceń.

Prostym modelem, którego parametry estymuje się bezpośrednio metodą LS z modelu regresji liniowej jest model ARX (*AutoRegressive with eXogenous input*, autoregresja to zależność $y(n)$ od poprzednich próbek na wyjściu):

$$\begin{aligned} y(n) + a_1 y(n-1) + \dots + a_{N_a} y(n-N_a) &= \\ &= b_0 u(n-d) + b_1 u(n-1-d) + \dots + b_{N_b} u(n-N_b-d) + e(n) \end{aligned}$$

gdzie: $y(n)$ - sygnał wyjściowy,

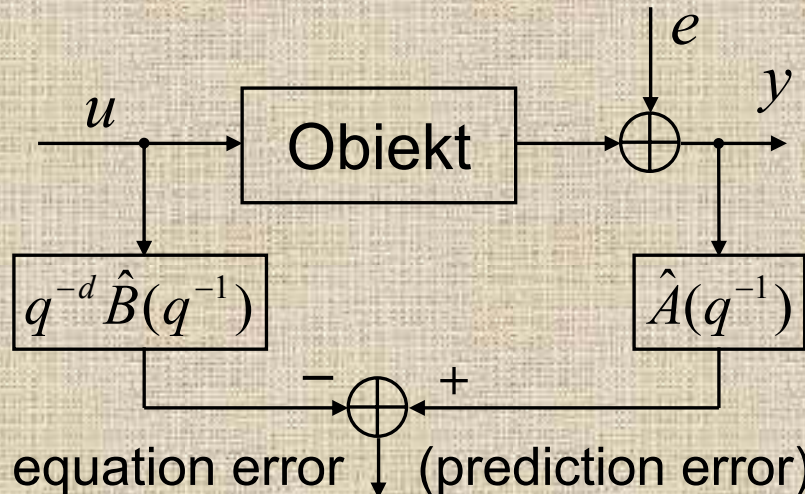
$u(n)$ - sygnał wejściowy, d – opóźnienie propagacji (*dead time*)

$e(n)$ - szum biały (ciąg nieskorelowany) modelujący zakłócenia.

W celu skrócenia zapisu równań różnicowych wprowadza się operator q^{-k} przesunięcia o k próbek w tył: $q^{-k}x(n)=x(n-k)$ (analogicznie do zespolonego operatora z^{-k} w dziedzinie transformaty Z):

$$A(q^{-1}) = 1 + a_1 q^{-1} + \dots + a_{N_a} q^{-N_a}, \quad B(q^{-1}) = b_0 + b_1 q^{-1} + \dots + b_{N_b} q^{-N_b}$$

$$A(q^{-1})y(n) = q^{-d} B(q^{-1})u(n) + e(n)$$



Rys. Model błędu predykcji (błędu równania) typu ARX

Reformulacji modelu ARX do modelu regresji liniowej dokonuje się tworząc następujące wektory regresorów (znane dane pomiarowe) i poszukiwanych parametrów:

$$\phi(n) = [-y(n-1), -y(n-2), \dots, -y(n-N_a), u(n-d), u(n-1-d), \dots, u(n-N_b-d)]^T$$

$$\theta = [a_1, a_2, \dots, a_{N_a}, b_0, b_1, \dots, b_{N_b}]^T$$

Takie podejście nazywa się *metodą błędu predykcji* (dla $d \geq 1$), ponieważ opiera się na minimalizacji wartości oczekiwanej sumy kwadratów błędów predykcji:

$$\min_{\hat{\theta}} E \left[\sum_{n=1}^{N-1} \varepsilon^2(n) \right], \quad \varepsilon(n) = y(n) - \hat{y}(n | n-1)$$

gdzie estymata: $\hat{y}(n | n-1) = \phi^T(n) \hat{\theta}(n-1)$

jest 1-kroowym predyktorem wyjścia w chwili n na podstawie danych do $n-1$.

Bardziej ogólnym modelem, którego identyfikacja wymaga rozszerzonej metody LS (ELS - *Extended Least Squares*), jest model ARMAX (*AutoRegressive, Moving Average with eXogenous input*):

$$y(n) + a_1 y(n-1) + \dots + a_{N_a} y(n - N_a) = b_0 u(n-d) + b_1 u(n-1-d) + \dots + b_{N_b} u(n - N_b - d) + v(n) + c_1 v(n-1) + \dots + c_{N_c} v(n - N_c)$$

gdzie zakłócenie $s(n)$ jest ciągiem *skorelowanych* próbek (ciągiem średniej ruchomej MA):

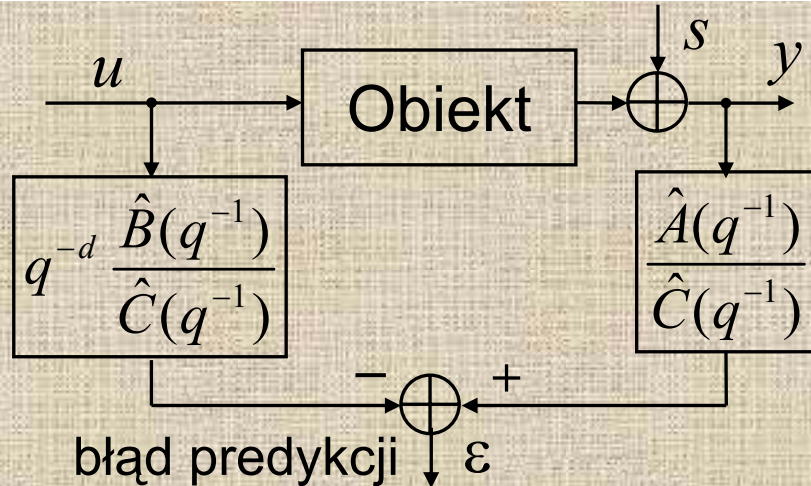
$$s(n) = v(n) + c_1 v(n-1) + \dots + c_{N_c} v(n - N_c)$$

Zakłada się, że macierz kowariancji szumu $\text{cov } v = E[v \cdot v^T]$ jest nieznana.

W zapisie skróconym: $A(q^{-1})y(n) = q^{-d} B(q^{-1})u(n) + C(q^{-1})v(n)$

gdzie: $C(q^{-1}) = 1 + c_1 q^{-1} + \dots + c_{N_c} q^{-N_c}$

Zastosowanie metody LS do modelu ARMAX daje estymaty obciążone. Jego identyfikacja wymaga rozszerzonej metody LS (ELS - *Extended Least Squares*) z estymacją parametrów korelacji $C(q^{-1})$.



Rys. Schemat blokowy modelu ARMAX

Model regresji w metodzie RLS tworzą rozszerzone wektory regresorów i poszukiwanych parametrów:

$$\phi(n) = [-y(n-1), \dots, -y(n-N_a), u(n-d), u(n-1-d), \dots, u(n-N_b), \varepsilon(n), \dots, \varepsilon(n-N_c)]^T$$

$$\theta = [a_1, \dots, a_{N_a}, b_0, b_1, \dots, b_{N_b}, c_1, \dots, c_{N_c}]^T$$

gdzie niemierzalne zakłócenia $\{v(n)\}$ są uwzględniane przez aproksymowanie ich obliczanymi błędami predykcji:

$$\varepsilon(n) = y(n) - \phi^T(n) \hat{\theta}(n-1)$$

Estymacja ELS jest dla szumu gaussowskiego estymacją *największej wiarygodności* (Maximum Likelihood - ML). **Problem estymacji jest nieliniowy względem parametrów** (parametry występują w wektorze regresorów ϕ).

Modele identyfikacji transmitancji dyskretnej $G(z)$

W kontekście identyfikacji funkcję transmitancji faktoryzuje się w postaci wielomianów licznika i mianownika. Popularne są dwa modele identyfikacji transmitancji:

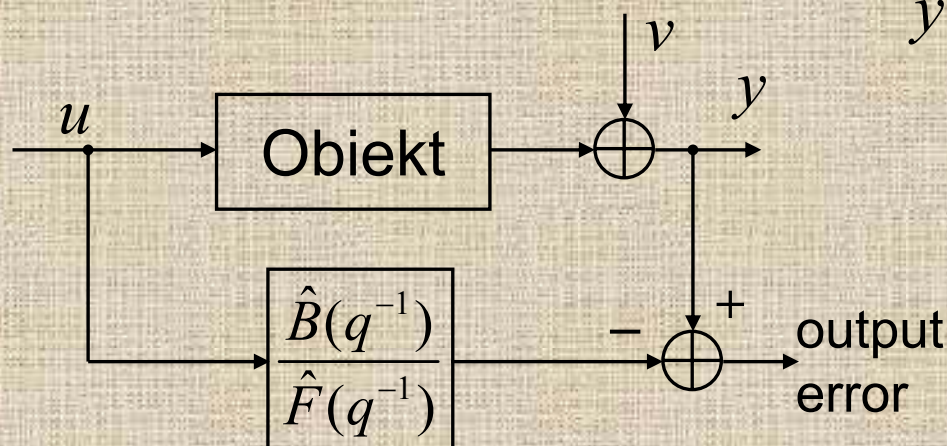
$$y(n) = q^{-d} \frac{B(q^{-1})}{F(q^{-1})} u(n) + v(n) \quad \text{- model błędu wyjściowego OE (output error model)} \quad F(q^{-1}) = 1 + f_1 q^{-1} + \dots + f_{N_f} q^{-N_f}$$

$$y(n) = q^{-d} \frac{B(q^{-1})}{F(q^{-1})} u(n) + \frac{C(q^{-1})}{D(q^{-1})} e(n) \quad \text{- Box-Jenkins model (BJ, ogólniejszy)}$$

W modelu OE nie czyni się założeń co do skorelowania ciągu zakłóceń $\{v(n)\}$, w modelu BJ ciąg szumu białego $\{e(n)\}$ jest filtrowany przez transmitancję C/D .

Ogólnie:

$$y(n) = G(q^{-1})u(n) + H(q^{-1})e(n)$$



Rys. Model błędu wyjściowego OE.
Część deterministyczna:

$$\hat{G}(z^{-1}) = q^{-d} \frac{\hat{B}(z^{-1})}{\hat{F}(z^{-1})}$$

Kryterium optymalizacji ma postać:

$$\min_{\hat{B}, \hat{F}} E \left[\sum_{n=1}^N \varepsilon^2(n) \right], \quad \varepsilon(n) = y(n) - q^{-d} \frac{\hat{B}(q^{-1})}{\hat{F}(q^{-1})} u(n)$$

z estymatą próbek wyjściowych:

$$\hat{y}(n) = q^{-d} \frac{\hat{B}(q^{-1})}{\hat{F}(q^{-1})} u(n)$$

Przykład: Różnica pomiędzy modelem błędu predykcji ARX

$$y(n) + ay(n-1) = bu(n-1) + e(n), \quad A(q^{-1}) = 1 + aq^{-1}, \quad d = 1, \quad B(q^{-1}) = b$$

a modelem błędu wyjścia OE postaci:

$$y(n) = \frac{bq^{-1}}{1 + aq^{-1}} u(n) + v(n), \quad F(q^{-1}) = 1 + aq^{-1}, \quad d = 1, \quad B(q^{-1}) = b$$

$$\text{Estymata metody błędu predykcji:} \quad \hat{y}(n) = -\hat{a}y(n-1) + \hat{b}u(n-1)$$

$$\text{Estymata metody błędu wyjściowego:} \quad \hat{y}(n) = -\hat{a}\hat{y}(n-1) + \hat{b}u(n-1)$$

Identyfikacja z modelem błędu wyjściowego jest nieliniowym problemem estymacji.

Estymacja parametrów modelu *output-error* z wybielaniem danych

1) Model początkowy („ARX” z v): $F(q^{-1})y(n) = q^{-d}B(q^{-1})u(n) + v(n)$

Służy do wyznaczenia początkowych estymat \tilde{F}, \tilde{B} standardową metodą LS.

2) Filtrowane (wybielanie – *prewhitening*) danych pomiarowych:

$$y^F(n) = \frac{1}{\tilde{F}(q^{-1})} y(n), \quad u^F(n) = \frac{1}{\tilde{F}(q^{-1})} u(n)$$

3) Estymacja \hat{F}, \hat{B} metodą LS ze zmodyfikowanymi danymi z modelu:

$$F(q^{-1})y^F(n) = q^{-d}B(q^{-1})u^F(n) + v(n)$$

Wybielanie redukuje obciążenie (*bias*) estymat spowodowany przez korelacje w ciągu zakłóceń.

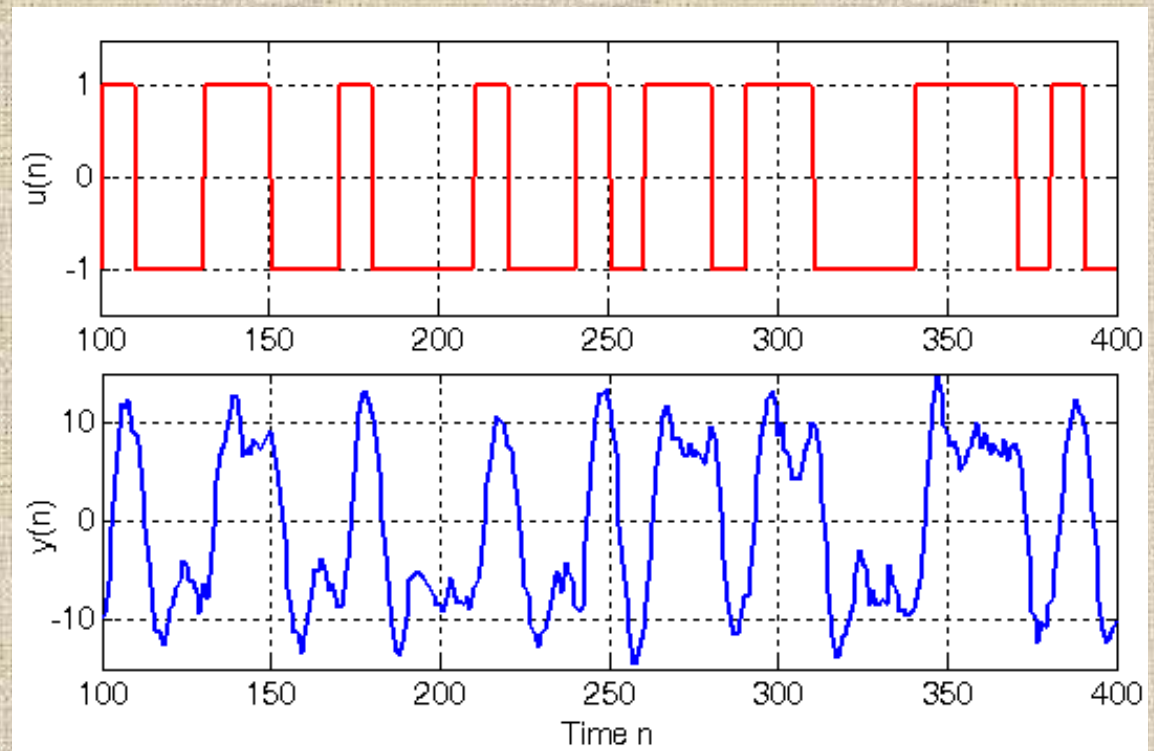
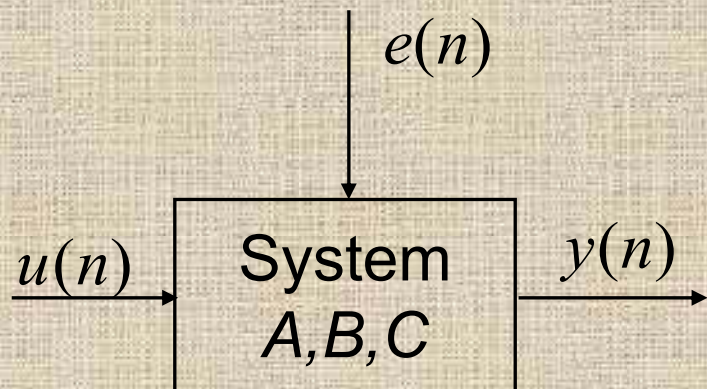
Rozszerzeniem takiego podejścia jest metoda zmiennych instrumentalnych (*Instrumental Variable Method*), w której wprowadza się nową macierz regresorów Z skorelowanych z pomiarami y , ale *nieskorelowanych* z zakłóceniami, tj. $E[Z^T v] = 0$. Estymata przyjmuje postać:

$$\hat{\theta}^{IV} = (Z^T \Phi)^{-1} Z^T Y$$

Przykład: Niech prawdziwy system (z którego pochodzą dane) będzie opisany modelem (ARMAX 2-go rzędu, jeden z benchmarków):

$$y(n) - 1.5y(n-1) + 0.7y(n-2) = \\ u(n-1) + 0.5u(n-2) + e(n) - e(n-1) + 0.2e(n-2)$$

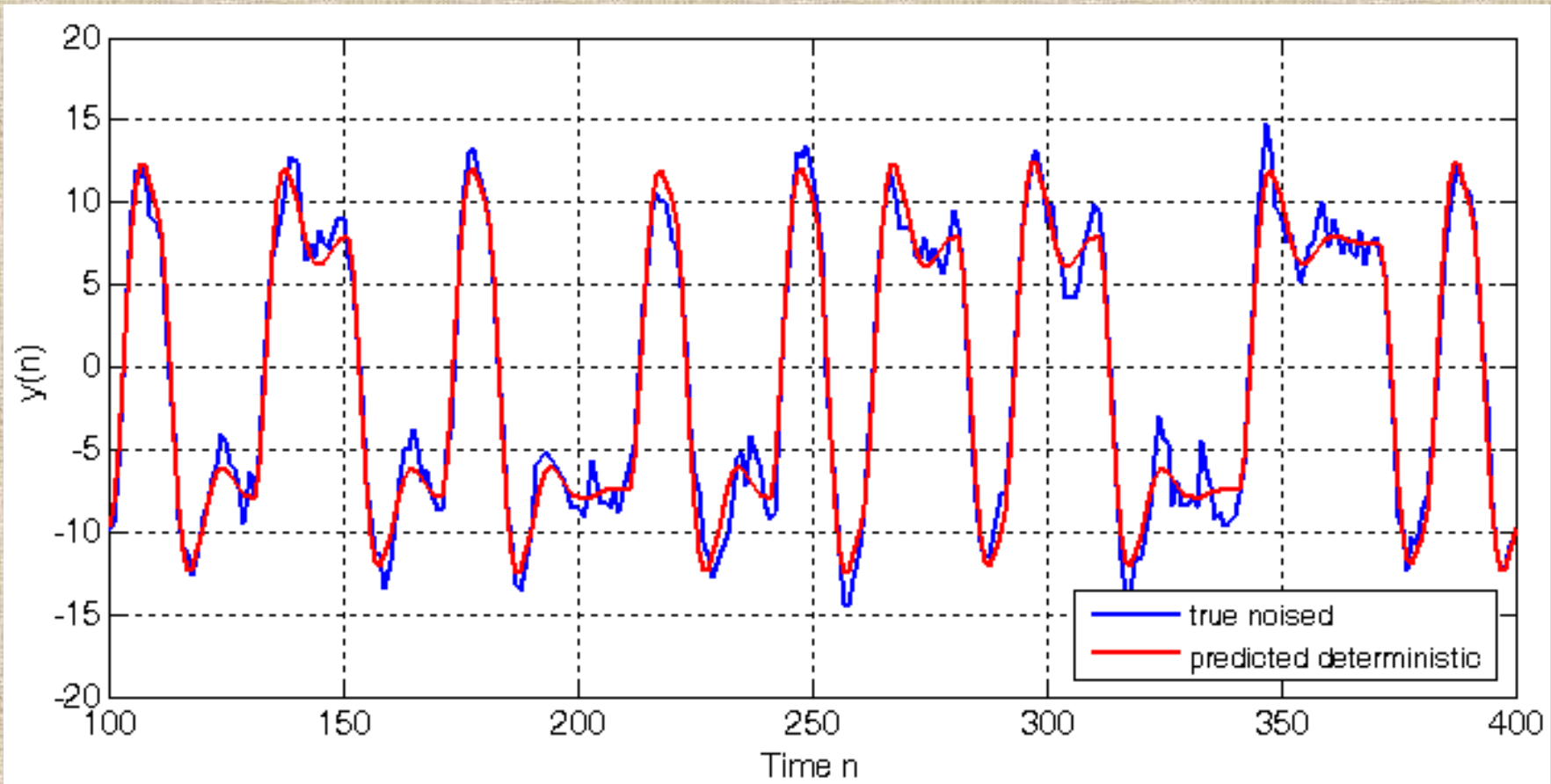
gdzie e jest szumem białym o średniej $m_e=0$ i odchyleniu standardowym $\sigma_e=1$.



Rys. Schemat i symulacja sygnału wejściowego PRBS (*Pseudo Random Binary Sequence*) i wyjściowego (z zakłóceniem)

Tabela. Wyniki estymacji parametrów z odchyleniem standardowym i wartościami funkcji kosztu V dla różnych modeli (drugiego rzędu) i metod identyfikacji. Model ARMAX jest dokładnym modelem struktury systemu.

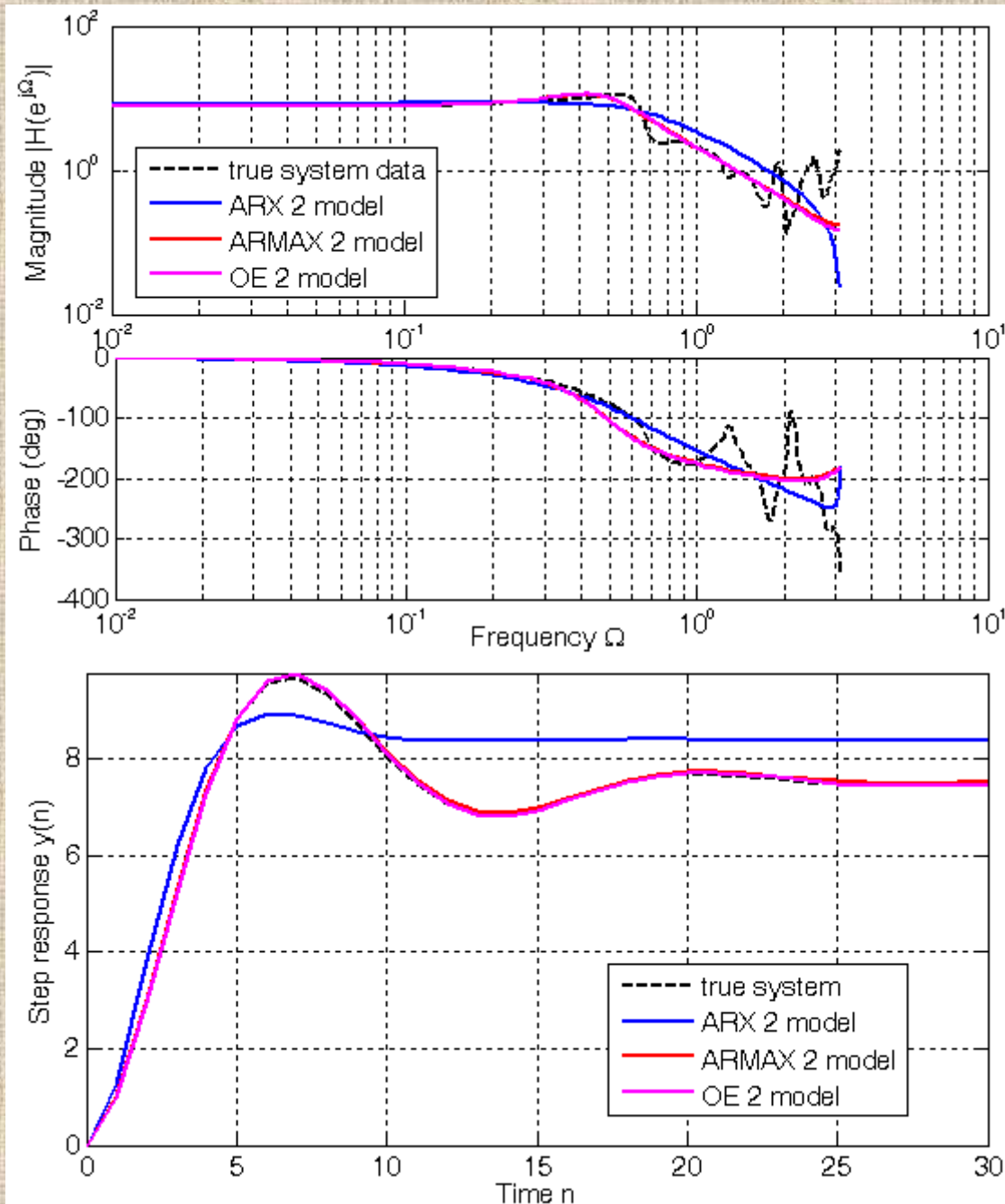
True value (ARMAX 2)	ARX $N=2$, LS	ARMAX $N=2$, ELS	OE $N=2$,	OE $N=2$ with prewhitening
$a_1=-1.5$	$a_1=-1.130\pm 0.039$	$a_1=-1.506\pm 0.013$	$f_1=-1.512\pm 0.013$	$f_1=-1.507\pm 0.013$
$a_2=0.7$	$a_2=0.417\pm 0.031$	$a_2=0.702\pm 0.009$	$f_2=0.709\pm 0.009$	$f_2=0.708\pm 0.009$
$b_1=1.0$	$b_1=1.237\pm 0.146$	$b_1=1.035\pm 0.091$	$b_1=0.985\pm 0.102$	$b_1=0.947\pm 0.083$
$b_2=0.5$	$b_2=1.177\pm 0.187$	$b_2=0.445\pm 0.128$	$b_2=0.488\pm 0.146$	$b_2=0.557\pm 0.113$
$c_1=-1.0$		$c_1=-1.051\pm 0.052$		
$c_2=0.2$		$c_2=0.217\pm 0.051$		
	$V=1.677$	$V=0.972$	$V=1.432$	$V=1.042$



Rys. Rzeczywiste dane wyjściowe (z zakłóceniami) $y(n)$ oraz ich estymaty

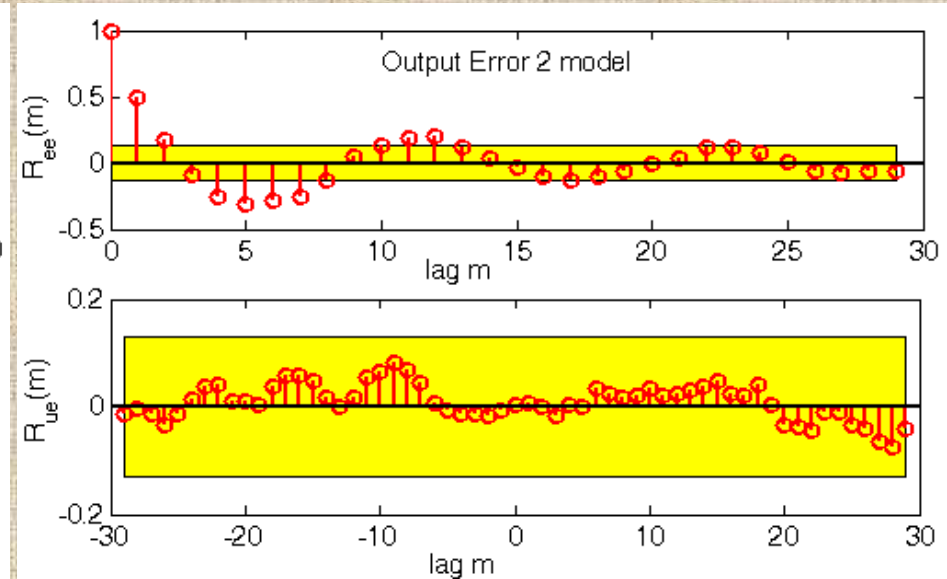
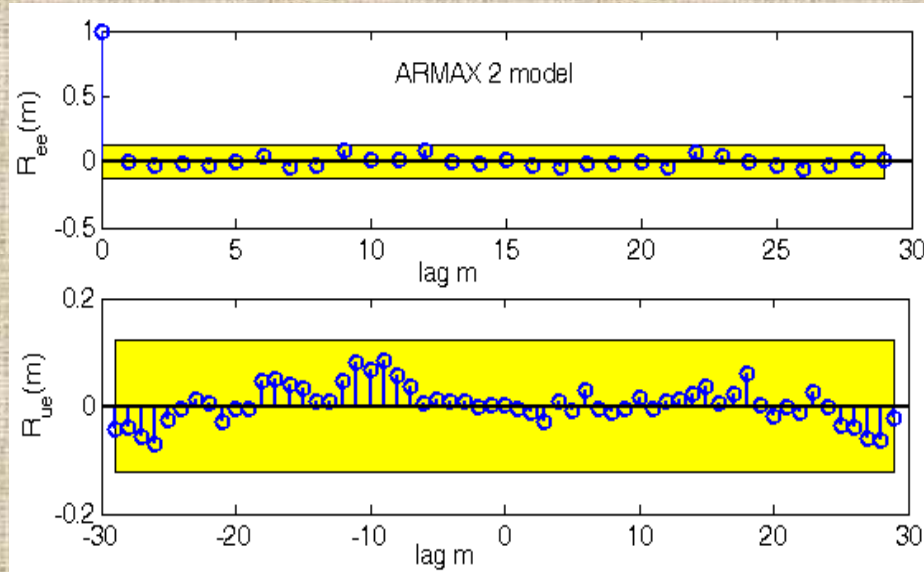
$$\hat{y}_{\text{det}}(n) = [\hat{B}(q^{-1}) / \hat{A}(q^{-1})]u(n)$$

wyznaczone dla części deterministycznej modelu ARMAX 2-go rzędu.



Rys. Charakterystyki widmowe części deterministycznej (od u do y) rozpatrywanych modeli oraz widmo danych z prawdziwego układu.

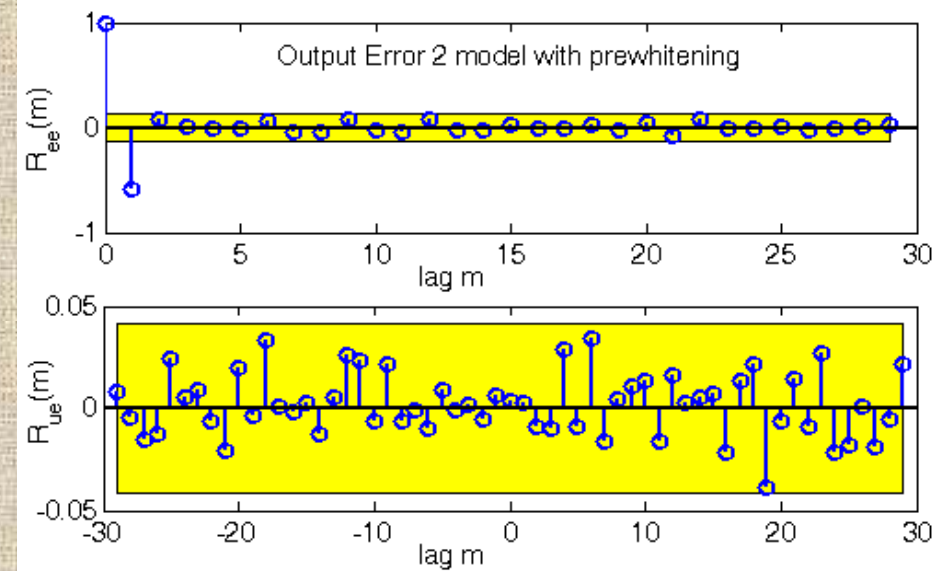
Rys. Odpowiedzi skokowe części deterministycznych prawdziwego układu i rozpatrywanych modeli.

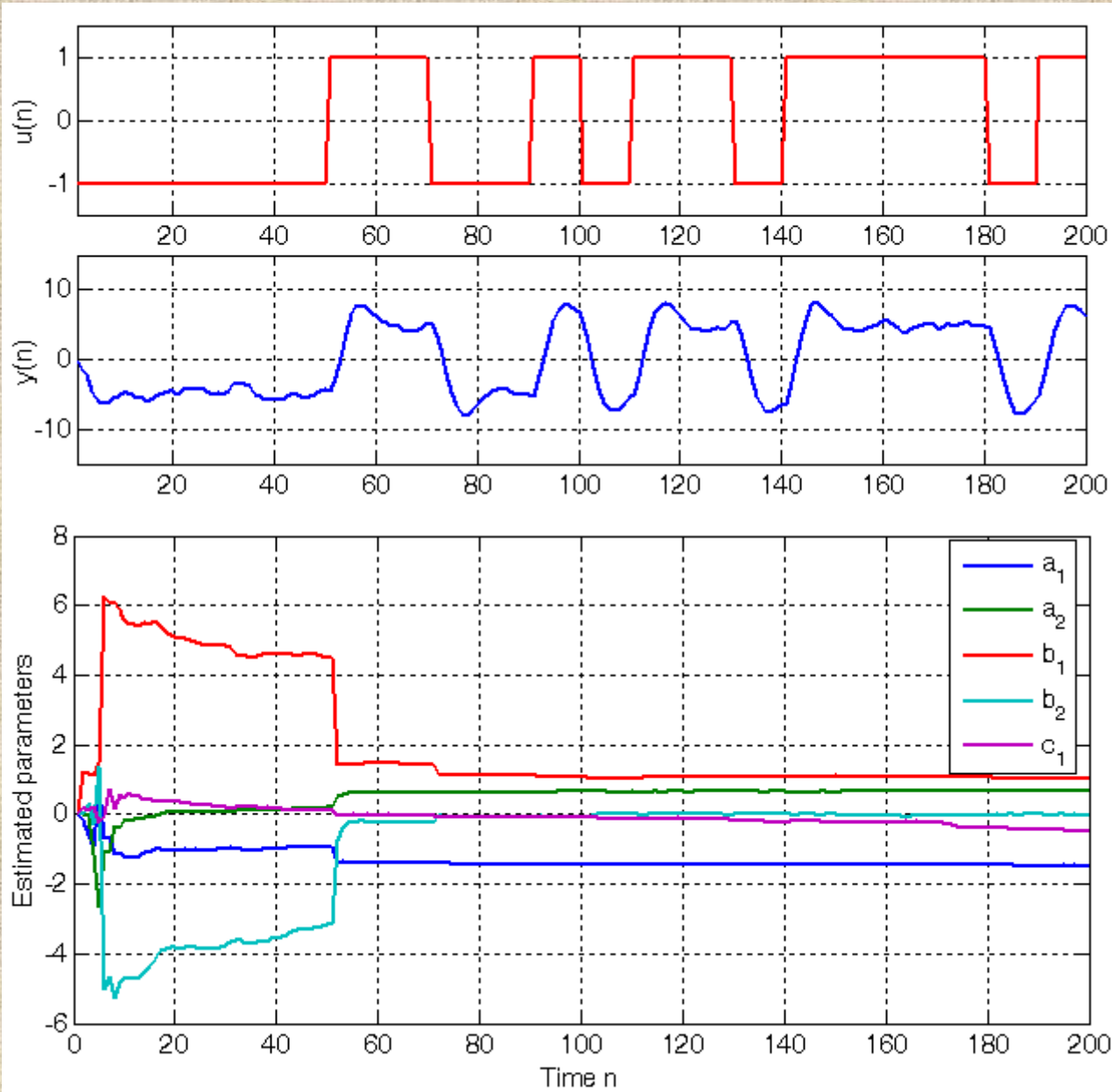


Rys. Estymaty autokorelacji R_{ee} błędów dopasowania (*residuals*) $\varepsilon(n)$ wyjść oraz korelacji wzajemnej R_{ue} błędów dopasowania z próbkami wejścia $u(n)$ dla modeli ARMAX, OE oraz OE z wybielaniem.

Żółte obszary pokazują 99% przedziały ufności korelacji przy założeniu, że ε jest szumem białym niezależnym od u . Jeżeli model jest dobrym opisem danych, to estymaty korelacji powinny mieścić się w przedziałach ufności.

Wniosek: model OE nie odzwierciedla wystarczająco korelacji danych.

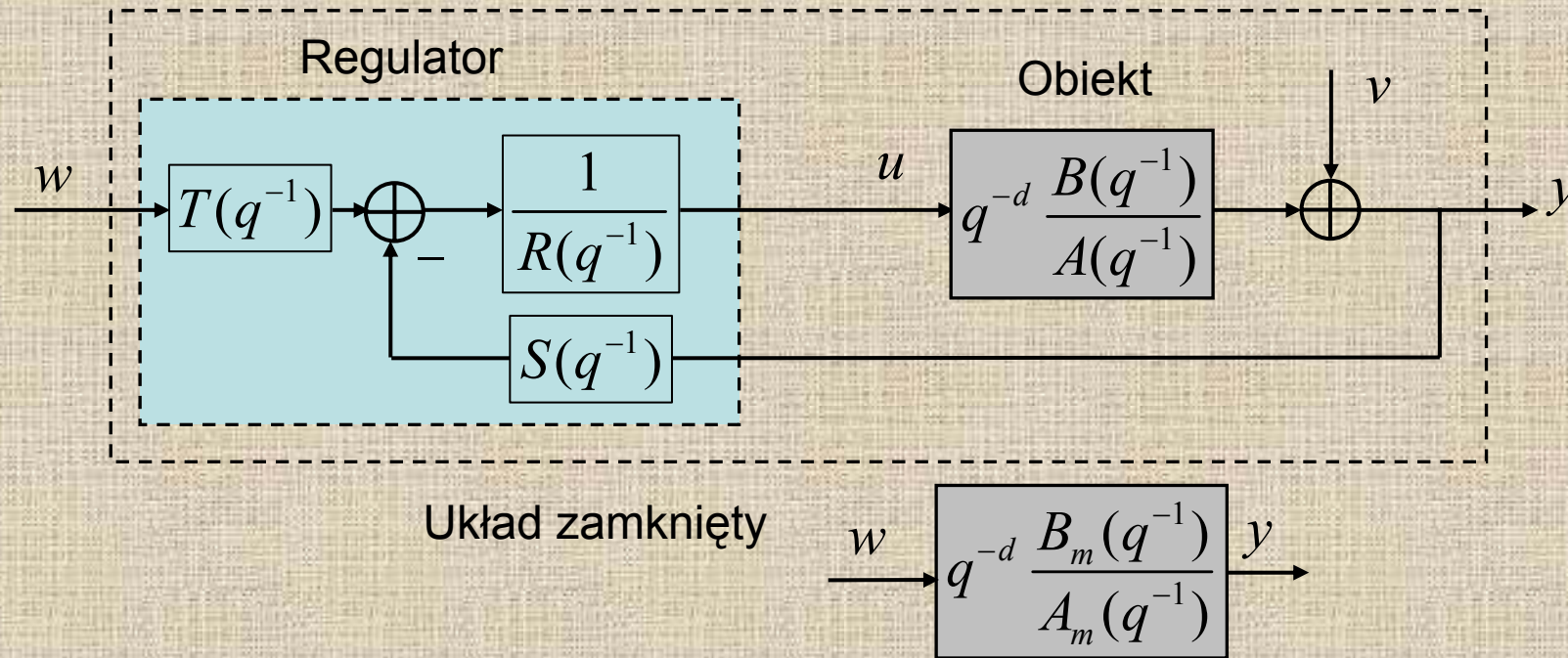




Przykład: Estymacja rekurencyjna (*on-line*) modelem ARMAX 2-go rzędu danych pochodzących z układu opisanego równaniem poniżej, gdzie e jest gaussowskim szumem białym, $\sigma_e=0.3$, wsp. zapomnienia $\lambda=0.98$. Estymacja wymaga *wystarczającego pobudzenia* (skoki u). Zerowe param. początk. Wart. końcowe:
 $a_1=-1.480$, $a_2=0.684$,
 $b_1=1.031$, $b_2=-0.037$
 (param. nadmiarowy),
 $c_1=-0.492$

$$y(n) - 1.5y(n-1) + 0.7y(n-2) = u(n-1) + e(n) - 0.5e(n-1)$$

Projektowanie regulacji metodą rozmieszczania zer i biegunów (zero-pole placement)



Rys. Schemat zamkniętego układu regulacji: $B(q^{-1})$, $A(q^{-1})$, d – znane z etapu identyfikacji, $B_m(q^{-1})/A_m(q^{-1})$ – zakładana transmitancja układu zamkniętego.

Równanie regulatora: $R(q^{-1})u(n) = T(q^{-1})w(n) - S(q^{-1})y(n)$

Równanie układu zamkniętego (UZ):

$$\frac{TB}{AR + q^{-d}BS} = \frac{B_m}{A_m}$$

$$\frac{TB}{AR + q^{-d}BS} = \frac{B_m}{A_m} \rightarrow \frac{(A_0 B_{m1})(B^+ B^-)}{AR_1 B^+ + q^{-d} B^+ B^- S} = \frac{B_{m1} B^-}{A_m}$$

Czynniki B , które nie są czynnikami B_m , muszą być czynnikami R . Są to zera obiektu do skasowania. Dokonajmy rozbicia: $B=B^+B^-$ (B^+ - dobrze tłumione zera, B^- - słabo tłumione lub niestabilne zera, których nie można skasować, bo znalazłyby się w mianowniku UZ). Zatem $B_m=B_{m1}B^-$.

Regulator traktuje się jako kombinację sprzężenia stanu i obserwatora A_0 . Obserwator nie powinien podlegać błędom wskutek zmian w , więc nie występuje w równaniu charakterystycznym UZ. Ponieważ

$$\deg A_m \leq \deg(AR + q^{-d}BS)$$

więc czynnik kasujący interpretuje się jako A_0 , które się zadaje i

$$\deg A_0 = \deg(AR + q^{-d}BS) - \deg A_m$$

(A_0 uprości się po lewej stronie równania UZ, więc musi tam występować w liczniku w T). Dynamika A_0 jest szybsza niż dynamika A_m .

1) Rozwiązuje się równanie

$$AR_1 + q^{-d} B^- S = A_m A_0$$

ze względu na R_1 i S (równanie diofantyczne wielomianowe)

2) Wyznacza się wielomiany regulatora: $R = R_1 B^+$, $T = A_0 B_{m1}$

$$\text{zawiera } A_0 \text{ do uproszczenia } \rightarrow \frac{A_0 B_{m1} B^-}{AR_1 + q^{-d} B^- S} = \frac{B_{m1} B^-}{A_m}$$

Równanie diofantyczne ma nieskończenie wiele rozwiązań (istnieje rozwiązanie jednoznaczne jeśli: $\deg R_1 < \deg (q^{-d} B^-)$ i $\deg S < \deg A$).

Czynniki wyboru rozwiązania:

- tłumienie zakłóceń (*noise rejection*), np. transmitancja zakłóceniewa

$$\frac{Y(z)}{V(z)} = \frac{AR}{AR + q^{-d} BS} = \frac{AR}{A_m A_0 B^+}$$

- funkcje S/R i T/R nie wprowadzają dodatkowego opóźnienia:

$$\deg R = \deg S = \deg T = \deg A - 1$$

co odpowiada sprzężeniu stanu; jeżeli może wystąpić istotne opóźnienie ułamkowe przyjmuje się $\deg R - 1 = \deg S = \deg T$ (opóźnienie o 1 okres)

Naturalne wybory:

- kompensacja „całkowa”: $\begin{cases} \deg S = \deg A - 1 \\ \deg R_1 = \deg A_m + \deg A_0 - \deg A \end{cases}$
- kompensacja „różniczkowa”: $\begin{cases} \deg S = \deg A_m + \deg A_0 = \deg B^- - d \\ \deg R_1 = \deg B^- + d - 1 \end{cases}$

Etapy projektowania:

- 1) rozkład B na B^+ i B^- (*spectral factorization*),
- 2) rozwiązanie równania diofantycznego i wyznaczenie R, S, T .

Dwa przypadki szczególne eliminują etap 1:

- a) kasowanie wszystkich zer obiektu: $B=B^+$ i $B_m=B_{m1}=K$ (stała).

Wtedy: $R=R_1B$ i $T=A_0$

- b) Nie kasowane są żadne zera: $B_m=KB$ i $T=KA_0$

Regulacja z całkowaniem

Duże wzmocnienie DC i tłumienie zakłóceń LF uzyskuje się przez wstawienie do R całkowania $1/(1-q^{-1})$. Wtedy: $R=(1-q^{-1})^l R'$ i $\deg R = \deg R' - l$.

Oznacza to zwiększenie stopnia obserwatora:

$$\deg A_0 \geq 2 \deg A - \deg A_m - \deg B^+ + l - 1$$