

# PLC

## Programmable Logic Controller

- ✓ sterowanie logiczne
- ✓ sterowanie sekwencyjne

### Producenci m.in.:

Simens

Simatic: S5,S7

General Electric-Fanuc

GE Fanuc 90: Micro, 30, 70; VersaMax

Rockwell Automation

MicroLogix 1000, SLC-500, PLC-5

(Allen-Bradley)

Omron

OMRON: CPM1, CQM1, C500, C2000

Schneider Electric

TSX: Nano, Micro, Premium; Quantum

Mitsubishi

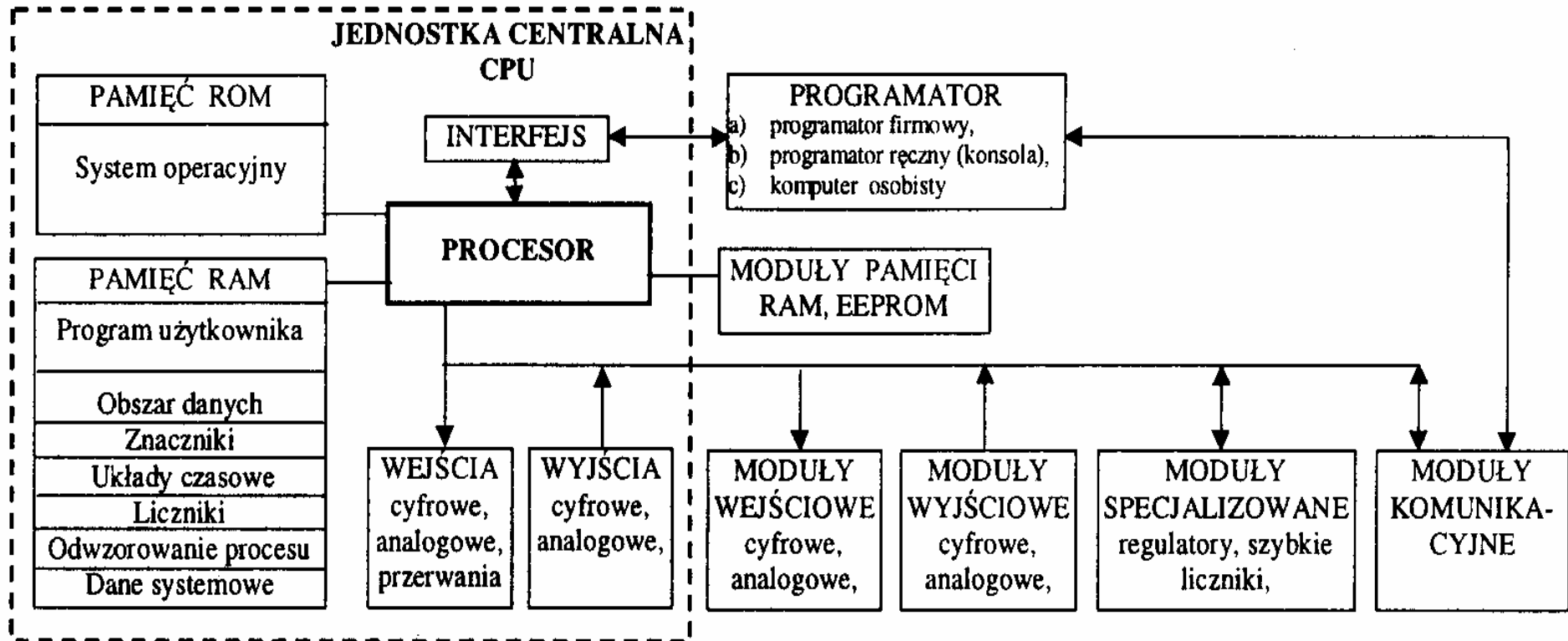
SAIA-Burgess Electronics

SAIA: PCD 1, PCD 2, PCD 4, PCD 6

Advantech

ADAM 5000

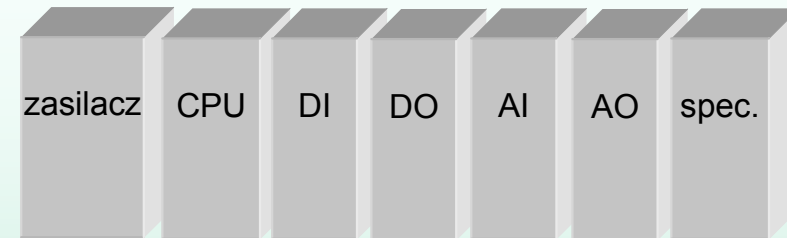
# Struktura sprzętowa



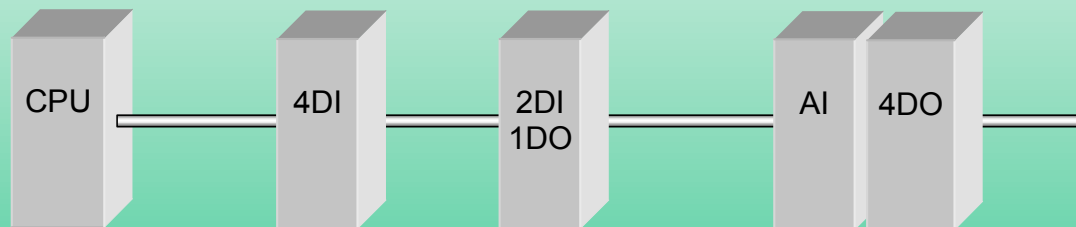
# Budowa



sterownik kompaktowy  
(compact)



sterownik modułowy  
(modular)



sterownik rozproszony  
(distributed)

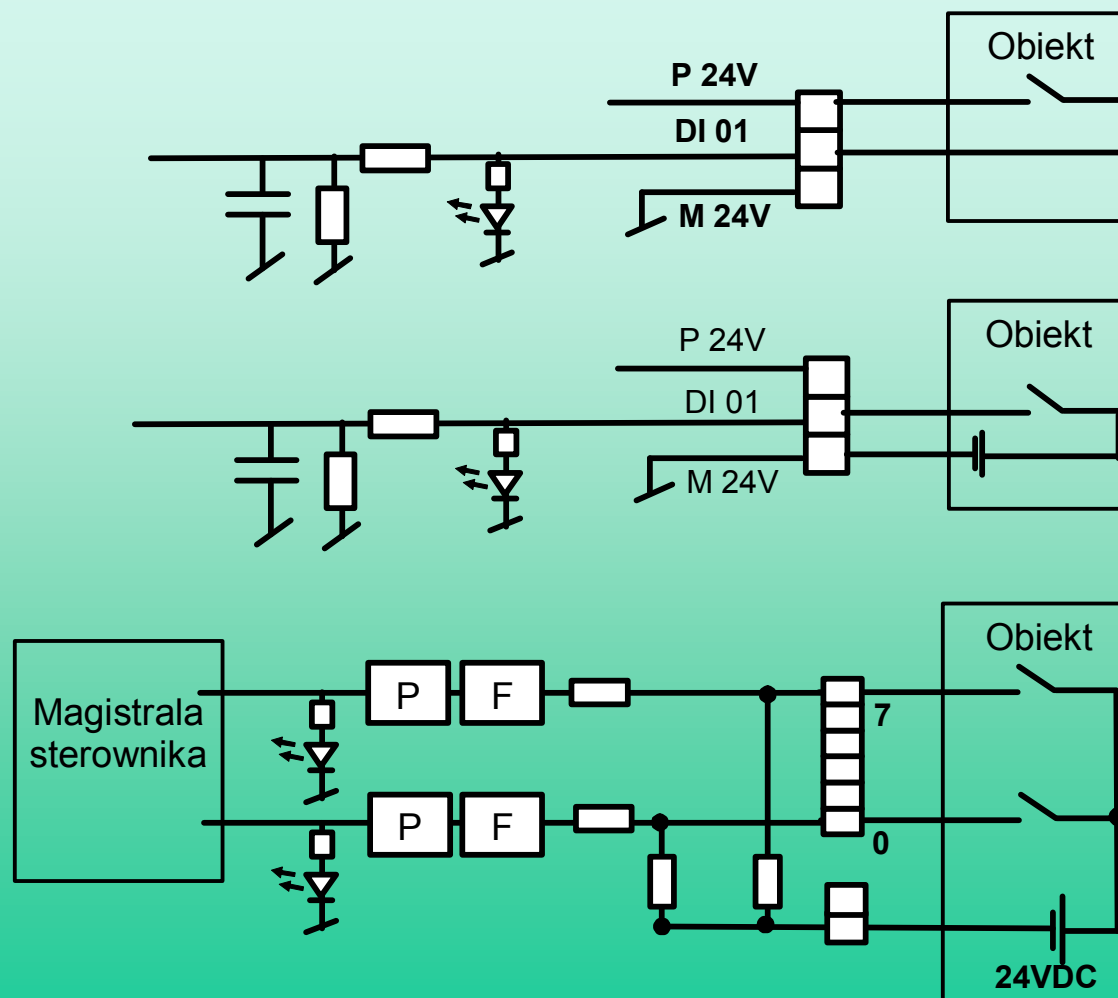
# Moduły DI

5/12/24 VDC / 110-240 VAC

logika + / -

separacja galwaniczna

filtry wejściowe



# Moduły DO

tranzystorowe

10..32 VDC

z/bez separacji

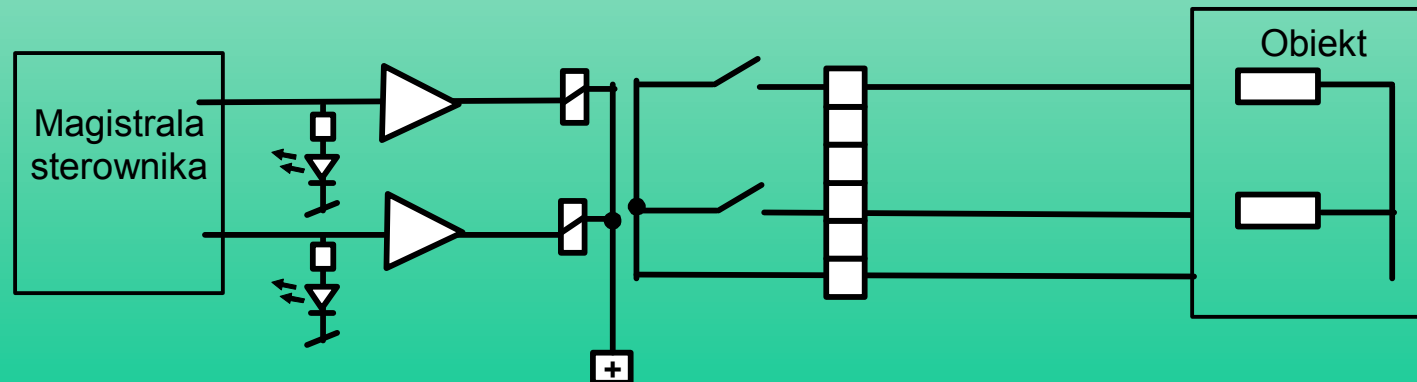
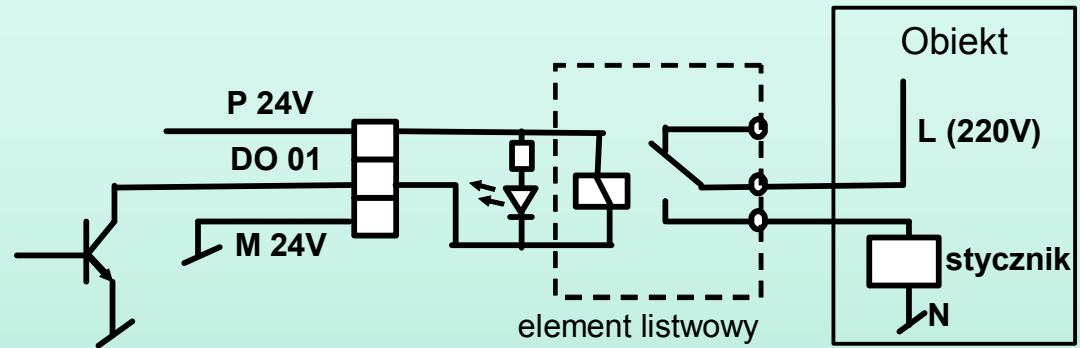
przełącznikowe

separacja galwaniczna

VAC / VDC

z/bez wspólnego zacisku

z/bez zabezpieczenia styków



## Moduły AI

napięciowe / prądowe  
dokładność (ilość bitów)  
multiplexer przetwornika  
szybkość

## Moduły AO

obciążalność  
zabezpieczenie wy

## Moduły specjalne

szybkie liczniki  
PID  
procesor komunikacyjny  
...

TYP	Moduły wejść/wyjść dyskretnych	...	/IO
PCD2.E110	<b>8 DI</b> (24 VDC, pos/neg, 8 ms, bez separacji)	<b>70</b>	8.8
PCD2.E160	<b>16 DI</b> (24 VDC, pos/neg, 8 ms, bez separacji)	<b>150</b>	9.4
PCD2.E610	<b>8 DI</b> (24 VDC, pos/neg, 8 ms, z separacją)	<b>90</b>	11.3
PCD2.E500	<b>6 DI</b> (110..240 VAC, z separacją)	<b>125</b>	20.8
PCD2.A200	<b>4 DO</b> (250 VAC, 2 A, przekaźnikowe)	<b>115</b>	28.8
PCD2.A250	<b>8 DO</b> (250 VAC, 2 A, przekaźnikowe)	<b>155</b>	19.4
PCD2.A400	<b>8 DO</b> (24 VDC, 0.5 A, tranzystorowe)	<b>90</b>	11.3
PCD2.B100	<b>2 DI</b> (24 VDC), <b>2 DO</b> (24 VDC, 0.5 A), <b>4 DI/DO</b> dowolne	<b>110</b>	13.8
	Moduły wejść/wyjść analogowych		
PCD2.W100	<b>4 AI</b> , napięciowe (0÷10VDC / ±10VDC) 12 bit	<b>445</b>	111
PCD2.W105	<b>4 AI</b> , prądowe (0÷20 mA / 4÷20 mA) 12 bit	<b>445</b>	111
PCD2.W110	<b>4 AI</b> , Pt-100, 12 bit	<b>530</b>	<b>132</b>
PCD2.W200	<b>8 AI</b> , napięciowe (0÷10VDC) 10 bit	<b>270</b>	34
PCD2.W210	<b>8 AI</b> , prądowe (0÷20 mA / 4÷20 mA) 10 bit	<b>270</b>	34
PCD2.W400	<b>4 AO</b> , napięciowe (0÷10 VDC) 8 bit	<b>250</b>	63
PCD2.W410	<b>4 AO</b> , uniwersalne (0÷10 VDC / 4÷20 mA) 8 bit	<b>310</b>	78
PCD2.W500	<b>2 AI</b> (0÷10VDC / ±10VDC), <b>2 AO</b> (0÷10VDC / ±10 VDC)	<b>685</b>	171

*PCD1/PCD2 SAIA*

TYP	Moduły wejść/wyjść dyskretnych	...	/IO
IC300DIM210	8 DI (12/24 VDC, pos/neg)	<b>336</b>	42
IC300DQM201	8 DO (5/12/24 VDC, 0.5 A, neg)	<b>359</b>	42
IC300DQM202	8 DO (250 VAC, 4 A)	<b>485</b>	60
IC300DIQ611	8 DI (12/24 VDC, pos/neg), 8 DO (5/12/24 VDC, 0.5 A, neg)	<b>431</b>	27
IC300DIQ616	8 DI (12/24 VDC, pos/neg), 8 DO (5/12/24 VDC, 0.5 A, pos)	<b>431</b>	27
IC300DIQ624	8 DI (120 VAC), 8 DO (80-250 VAC, 0.3 A)	<b>493</b>	31
IC300DIQ612	8 DI (12/24 VDC, pos/neg), 6 DO (250 VAC, 4 A)	<b>515</b>	37
	Moduły wejść/wyjść analogowych		
IC300ADC110	4 AI, napięciowe (0÷5VDC / ±5VDC / 0÷10VDC / ±10VDC)	<b>508</b>	127
IC300ADC120	4 AI, prądowe (0÷20 mA / 4÷20 mA)	<b>508</b>	127
IC300RTD100	4 AI, RTD (Pt-100, Pt-200, Pt-500, Pt-1000)	<b>752</b>	188
IC300ADC010	2 AI, napięciowe (0÷5 VDC / ±5VDC / 0÷10VDC / ±10VDC)	<b>440</b>	220
IC300ADC020	2 AI, prądowe (0÷20 mA, 4÷20 mA)	<b>440</b>	220
IC300RTD000	2 AI, RTD (Pt-100, Pt-200, Pt-500, Pt-1000)	<b>647</b>	324
IC300DAC101	4 AO, napięciowe (±10 VDC)	<b>752</b>	188
IC300DAC102	4 AO, prądowe (4÷20 mA)	<b>752</b>	188
IC300DAC001	2 AO, napięciowe (±10 VDC)	<b>636</b>	318
IC300DAC002	2 AO, prądowe (4÷20 mA)	<b>636</b>	318
IC300MIX111	2 AI, napięciowe (±10VDC), 2 AO, napięciowe (±10 VDC)	<b>799</b>	199
IC300MIX122	2 AI, prądowe (4÷20 mA), 2 AO, prądowe (4÷20 mA)	<b>799</b>	199

*OCS GeFanuc*



# Norma IEC 1131 "Programmable Controllers"

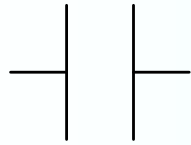
1. Definicje pojęć
2. Wymogi elektryczne, mechaniczne i funkcyjne
3. Ujednolicenie języków
4. Przewodnik dla projektantów
5. Opis komunikacji

Zmienne: **I** (wejście), **Q** (wyjście), **M** (pamięć)

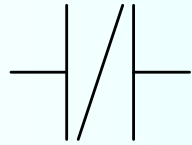
**X** (bit), **B** (bajt), **W** (słowo), **D** (2x słowo), **L** (4x słowo)

np.: %I1.1    %IW1.2

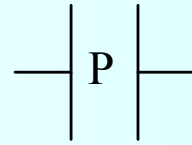
# Operacje podstawowe



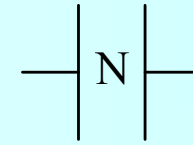
Styki zwarte  
(normalnie otwarte)



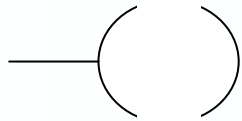
Styki rozwierne  
(normalnie zamknięte)



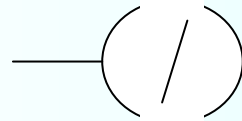
Styki reagujące na  
narastające zbcze



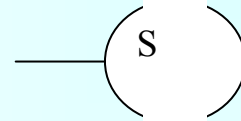
Styki reagujące na  
opadające zbcze



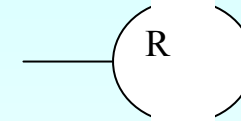
Cewka  
standardowa



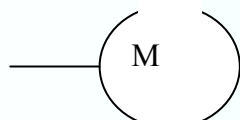
Cewka  
negująca



Cewka  
ustawiająca



Cewka  
resetująca



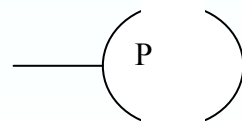
Cewka  
z zapamiętaniem stanu



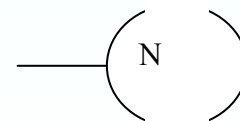
Cewka ustawiająca  
z zapamiętaniem stanu



Cewka resetująca  
z zapamiętaniem stanu



Cewka reagująca  
na narastające zbcze



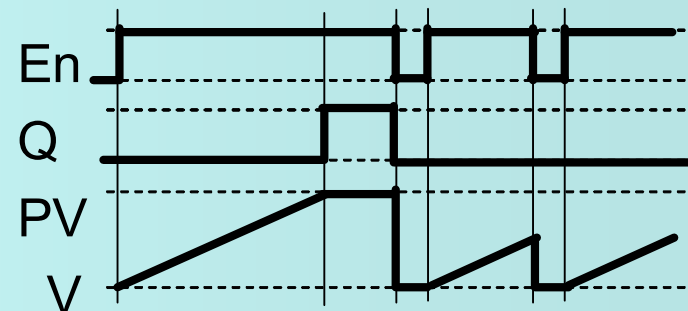
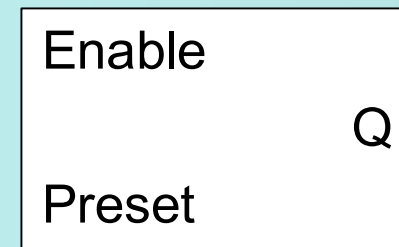
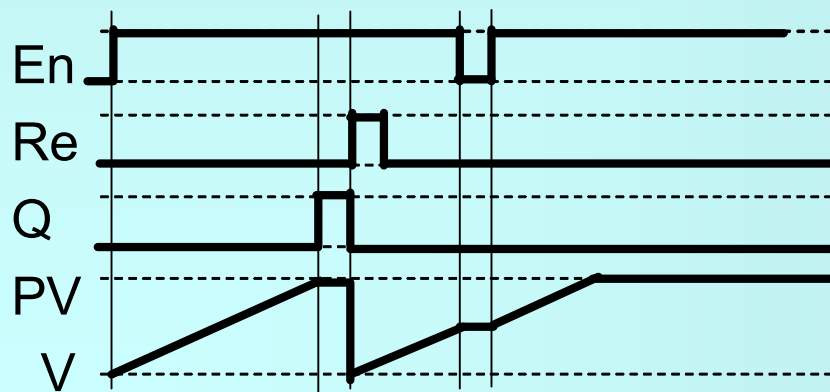
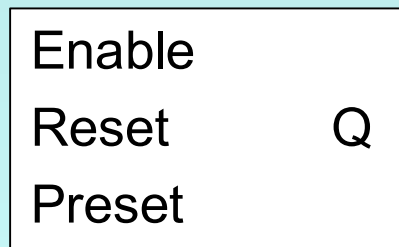
Cewka reagująca  
na opadające zbcze

# Układy czasowe

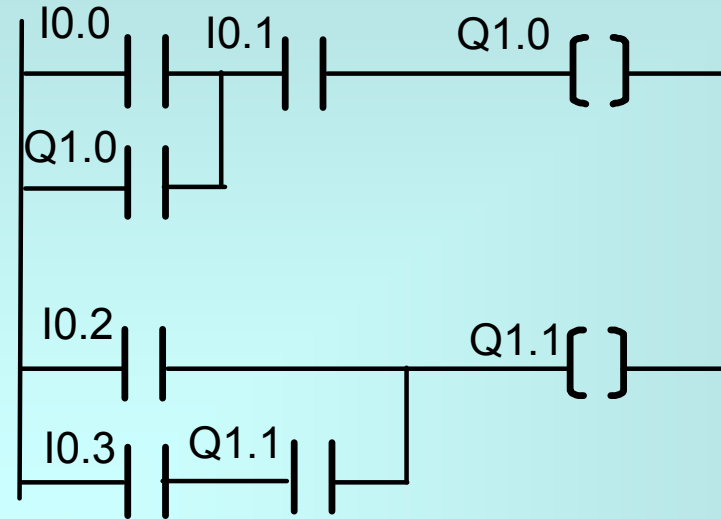
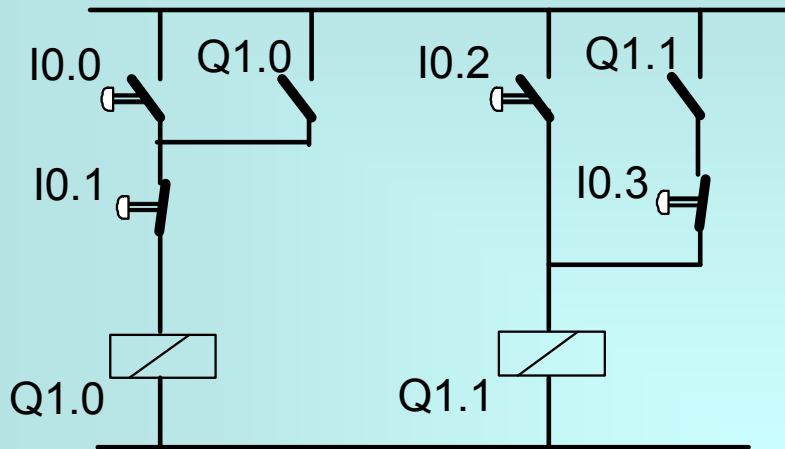
opóźnienie czasowe, generowanie zadanego przebiegu, odmierzanie czasu

# Układy licznikowe

licznik zdarzeń, dzielnik częstotliwości



# PLC a układy przekaźników



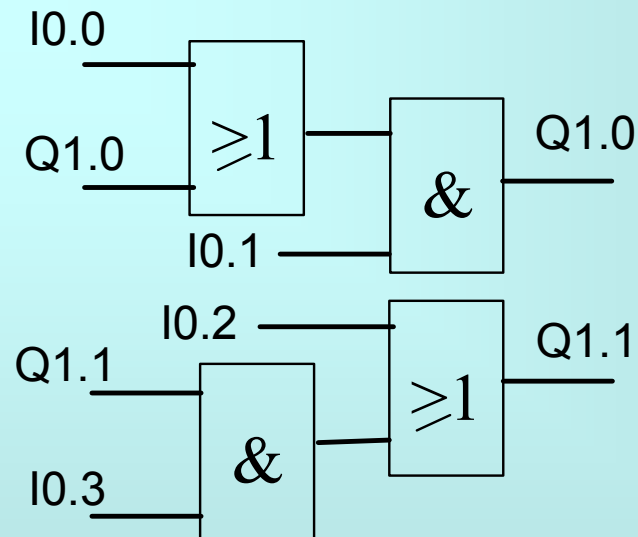
*OMRON:*

**LD 0.0**  
**OR 1.0**  
**AND 0.1**  
**OUT 1.0**

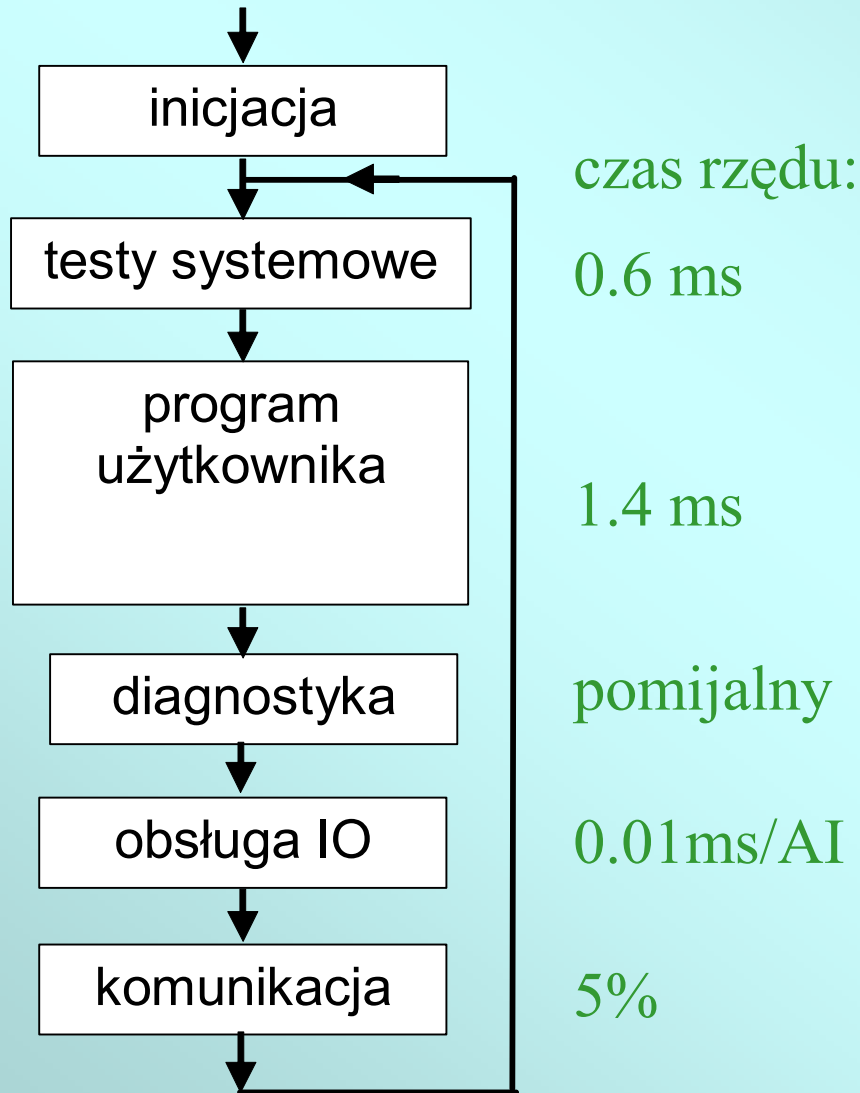
**LD 0.2**  
**LD 1.1**  
**AND 0.3**  
**OR LD**  
**OUT 1.1**

*SIEMENS:*

**:A(**  
**:O I0.0**  
**:O Q1.0**  
**:)**  
**:A( I0.1**  
**:= Q1.0**  
**:A I0.2**  
**:O**  
**:A Q1.1**  
**:A I0.3**  
**:= Q1.1**



# Pętla programowa – cykl pracy



Przykład:

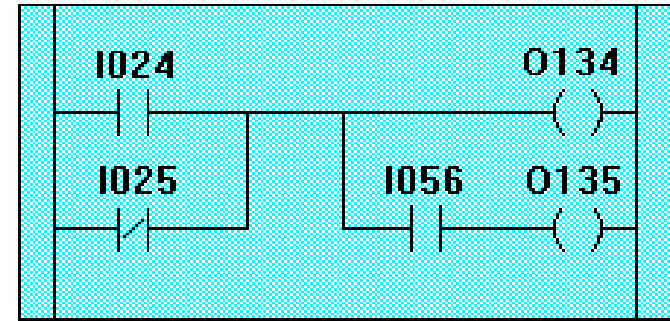
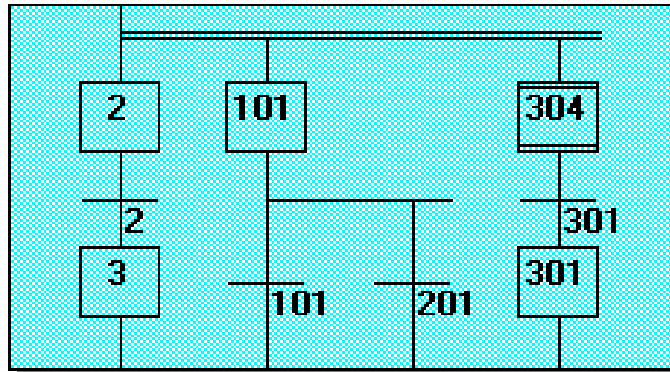
czas reakcji na zdarzenie

- 8 ms opóźnienia sygnału we
- 1 ms ustawienia systemowe
- 14 ms program użytkownika
- 10 ms opóźnienie sygnału wy

-----  
Σ 33 ms

lub 48 ms (+1 cykl)

# Języki programowania sterowników



**SFC** Sequential Function Chart

**LD** Ladder Diagram

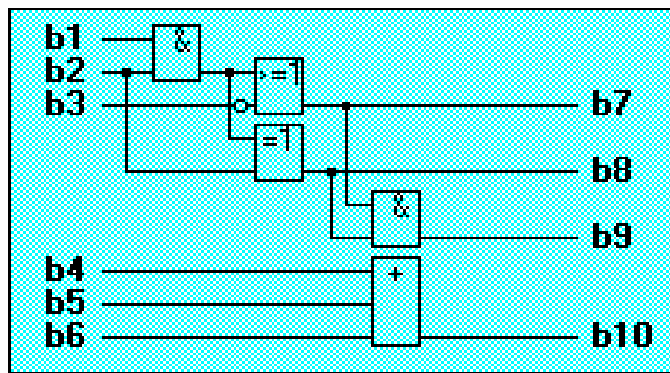
**ST** Structured Text

**IL** Instruction List

**FBD** Function Block Diagram

```

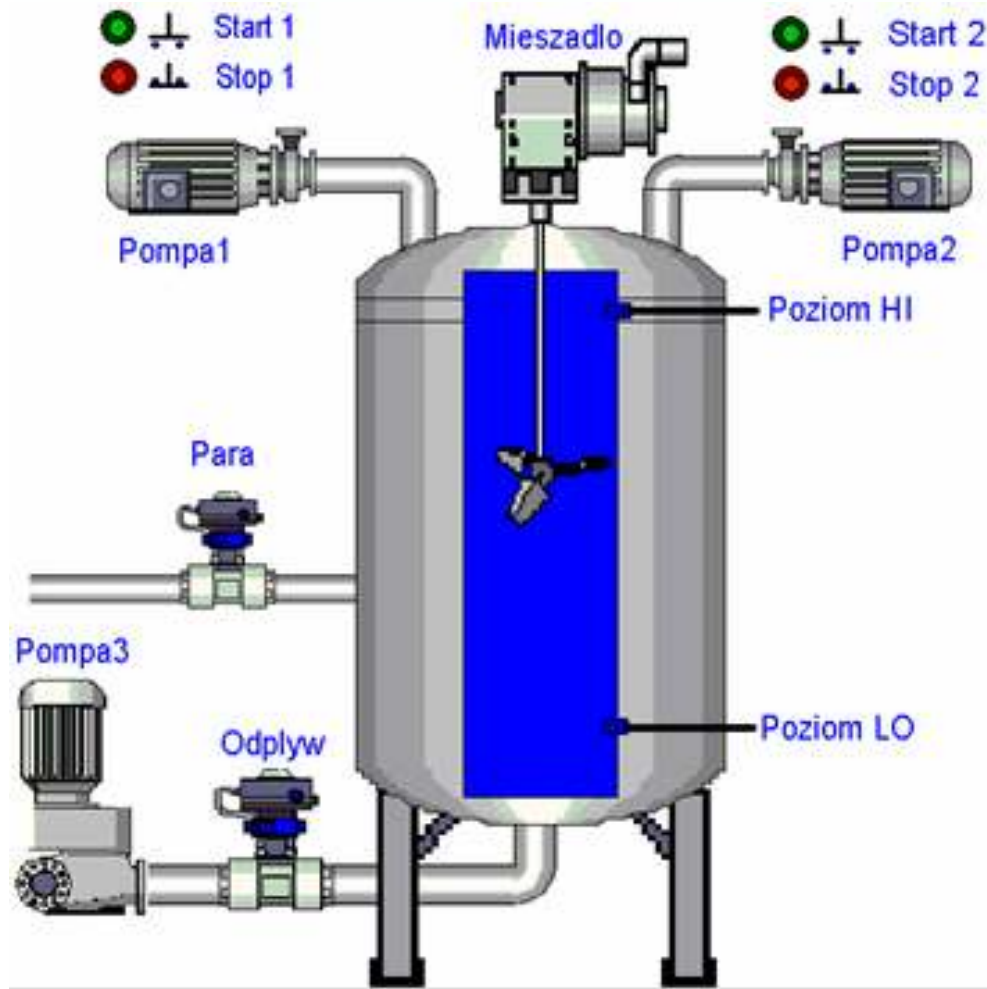
if (level <= level_max)
then
  out_valve := false;
  m_vlv := (vlv23 + dbh18) / 2;
else
  alarm_level := true;
end_if;
    
```



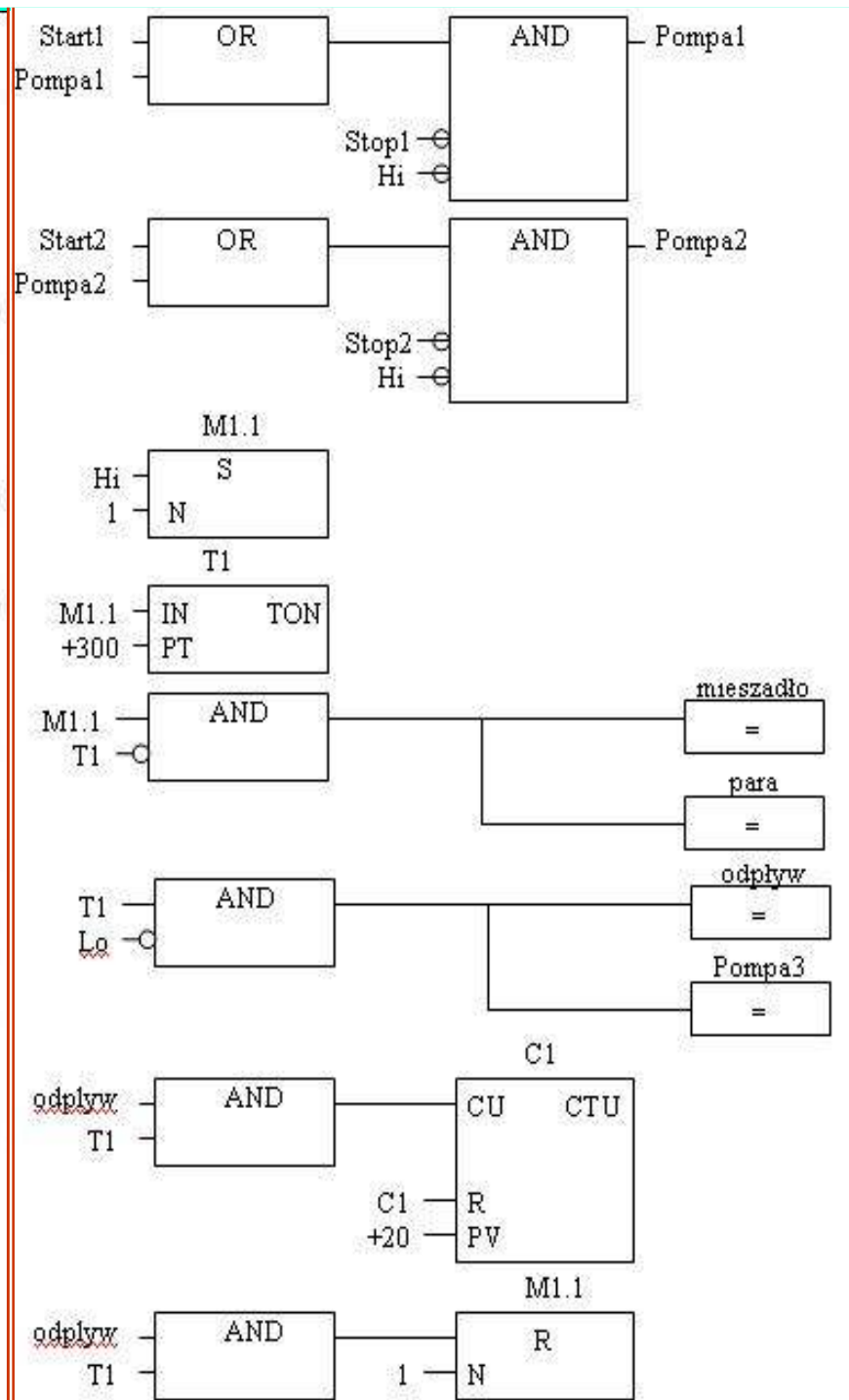
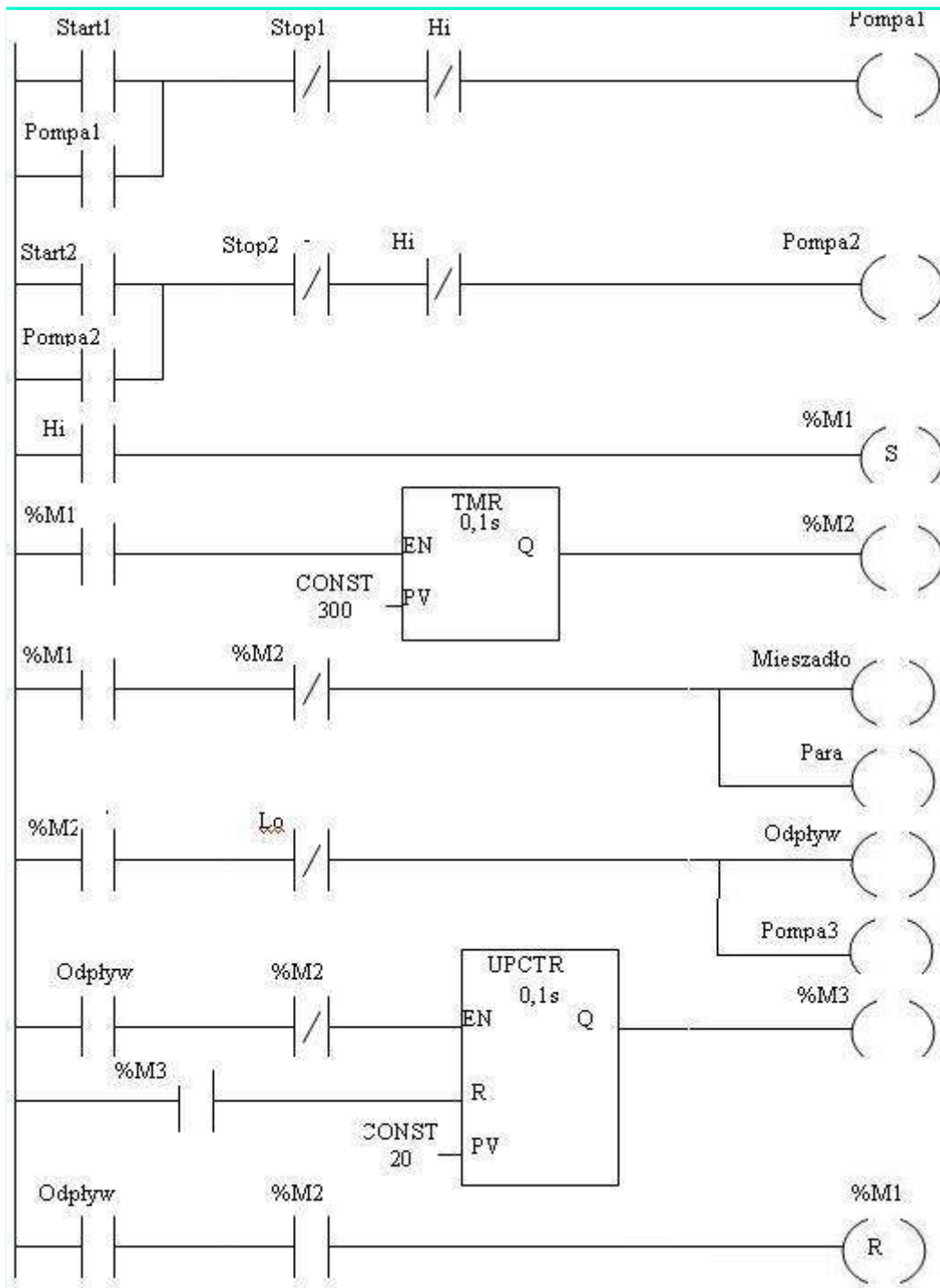
```

start_cmd: LD    bi101
            ADD   10
mul_op:    MUL(  i_bcmd
            SUB   bo100
            )
            ST    bcmd
            JMPNC mul_op
    
```

# Przykład



- napęłnić (*HI, pompa1, pompa2*)
- mieszać 30 s (*mieszadło*)
- podgrzewać (z. *para*)
- opróżnić (*LO, z.odpływ, pompa 3*)





**NETWORK 1**

```
LD "start1"  
O "pompa1"  
AN "stop1"  
AN "hi"  
= "pompa1"
```

**NETWORK 2**

```
LD "start2"  
O "pompa2"  
AN "stop2"  
AN "hi"  
= "pompa2"
```

**NETWORK 3**

```
LD "hi"  
S M1.1, 1
```

**NETWORK 4**

```
LD M1.1  
TON T1, +300
```

**NETWORK 5**

```
LD M1.1  
AN T1  
= "mieszadlo"  
= "para"
```

**NETWORK 6**

```
LD T1  
AN "lo"  
= "odplyw"  
= "pompa3"
```

**NETWORK 7**

```
LD "odplyw"  
A T1  
LD C1  
CTU C1, +20
```

**NETWORK 8**

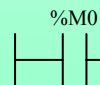
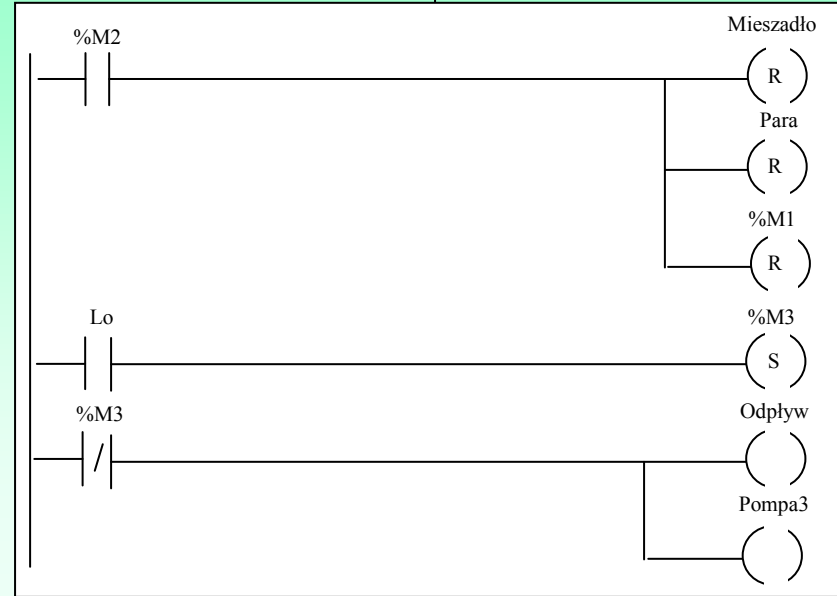
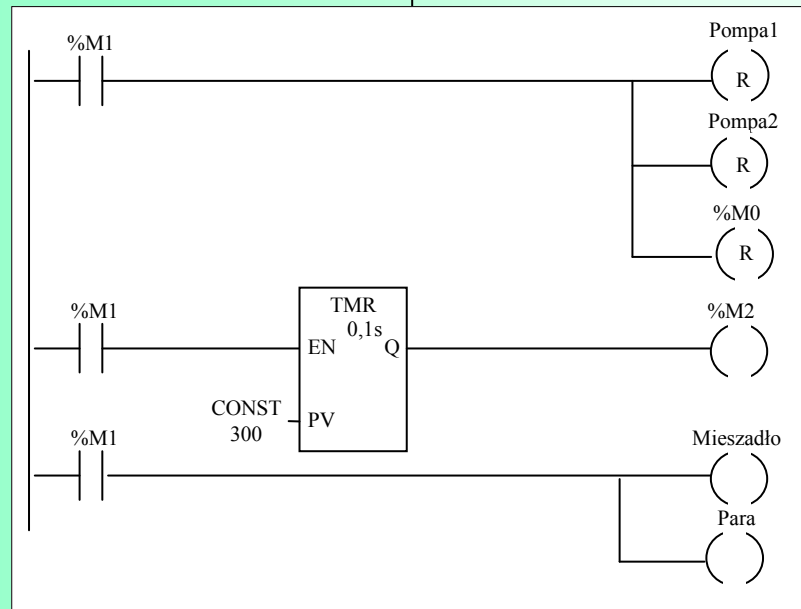
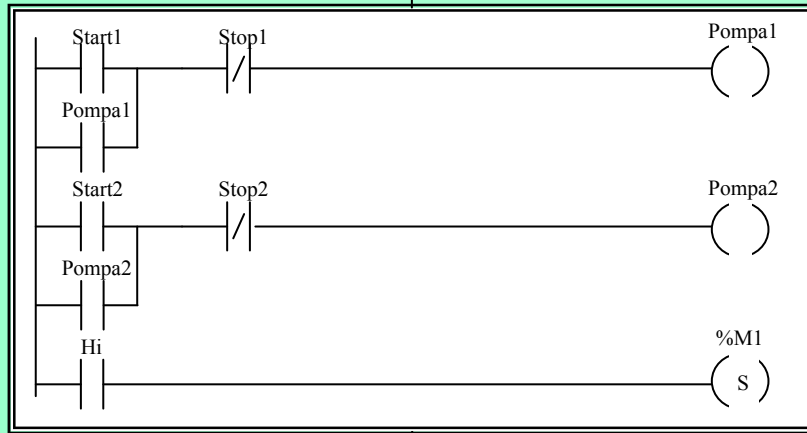
```
LD "odplyw"  
A T1  
R M1.1, 1
```

**VAR**

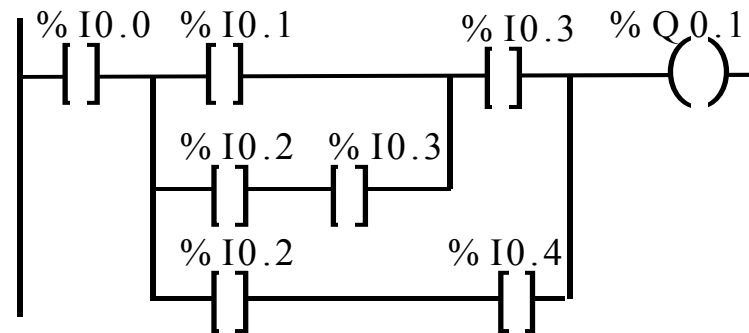
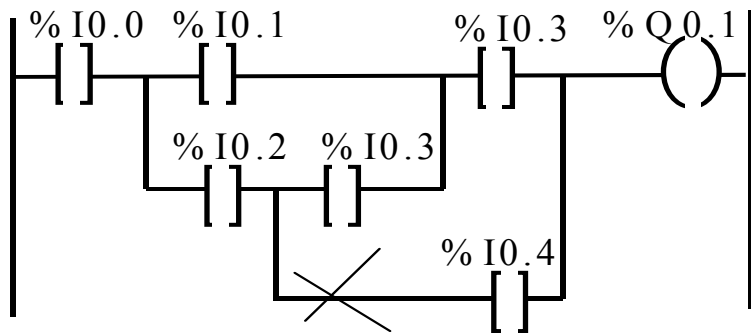
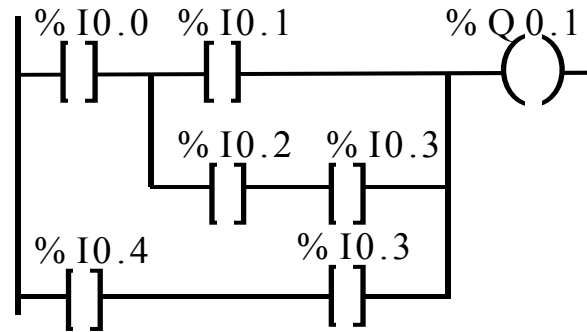
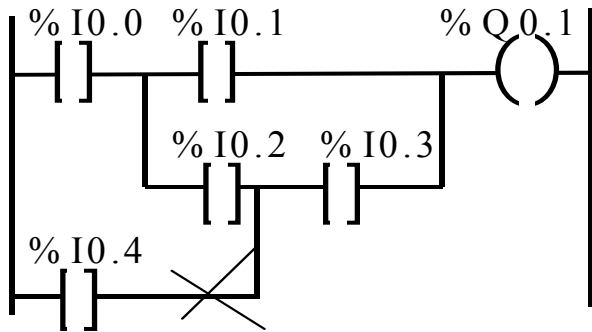
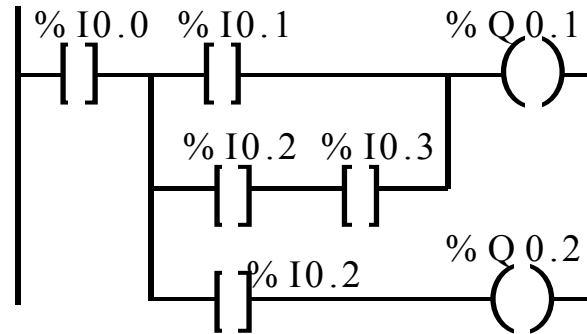
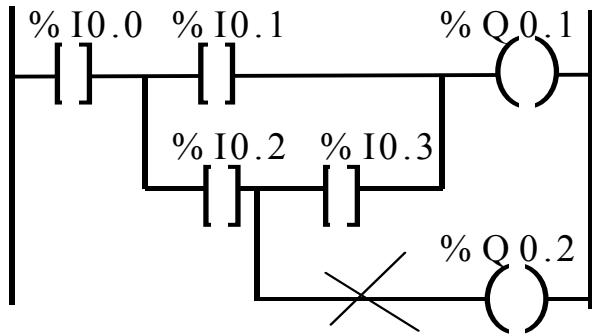
```
T1 : TON;  
C1 : CTU;  
Czas : TIME := t%300ms  
Ustaw1 : SR;  
Ustaw2 : RS;  
Start1 : BOOL At %I1  
Stop1 : BOOL At %I3  
Start2 : BOOL At %I2  
Stop2 : BOOL At %I4  
Pompa1 : BOOL At %Q1  
Pompa2 : BOOL At %Q2  
Mieszadlo : BOOL At %Q3  
Hi : BOOL At %I5  
LO : BOOL At %I6  
Para : BOOL At %Q4  
Odplyw : BOOL At %Q5  
Pompa3 : BOOL At %Q6
```

**END\_VAR****REPEAT**

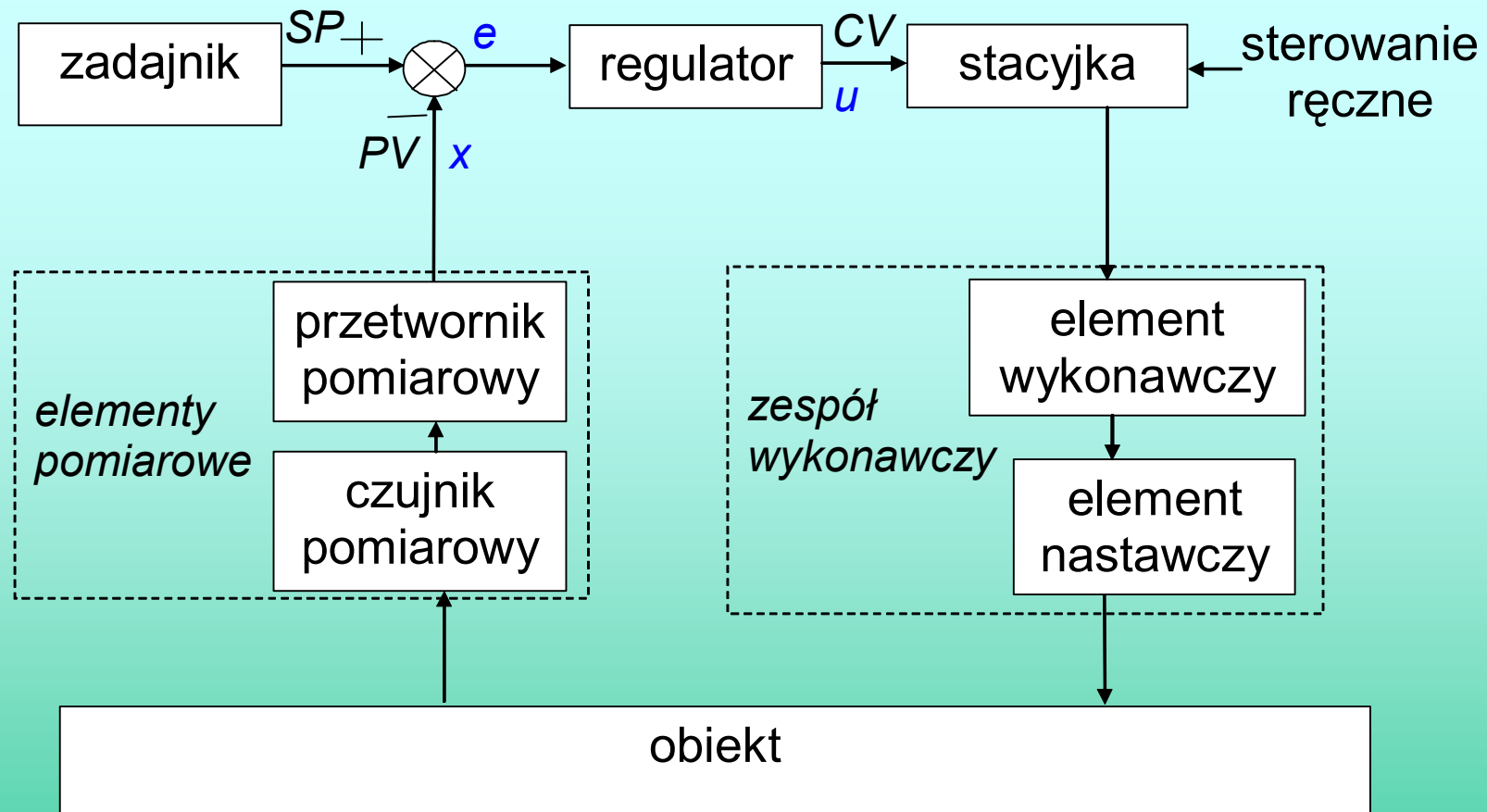
```
Pompa1:=(Start1 OR Pompa1) ANDN Stop1 ANDN Hi;  
Pompa2:=(Start2 OR Pompa2) ANDN Stop2 ANDN Hi;  
Ustaw1(S1:=Hi,R:=False);  
%M1:=Ustaw.Q;  
T1(IN:=%M1,PT:=czas);  
%M2:=T1.Q;  
Para:=%M1 ANDN %M2;  
Mieszadlo:= Para;  
Odplyw:= %M2 ANDN Lo;  
Pompa3:= Odplyw;  
%M4:=Odplyw ANDN %M2;  
C1(CU:=%M4,R:=%M3,PV:=20);  
%M3:=C1.Q;  
%M4:=Odplyw AND %M2;  
Ustaw2(S:=False,R1:=%M4);  
UNTIL (NOT C1.Q);
```



# Ograniczenia PLC



# Schemat funkcjonalny układu automatycznej regulacji



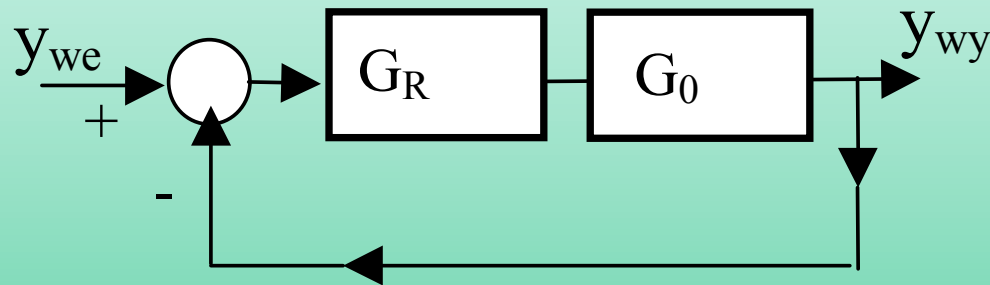
$$u(t) = K_p \left[ e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right] + U_0$$

# Struktura układu sterowania

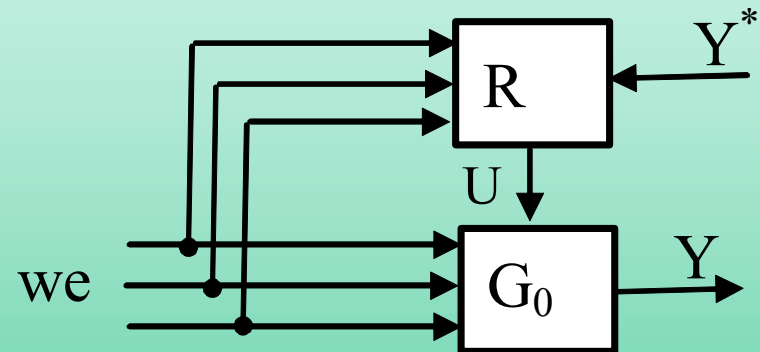
sterowanie

układ zamknięty  
(regulacja)

układ otwarty

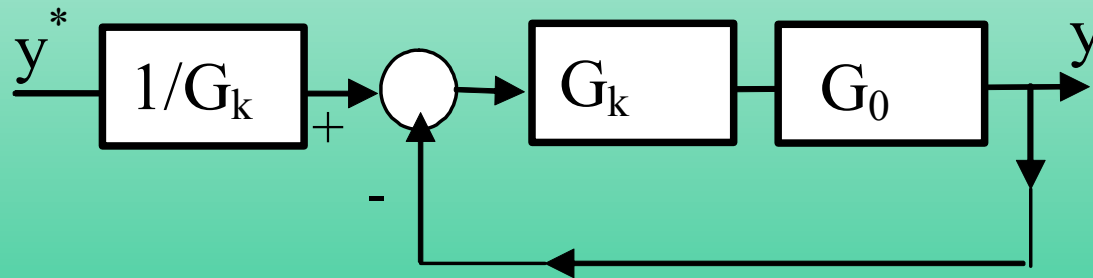
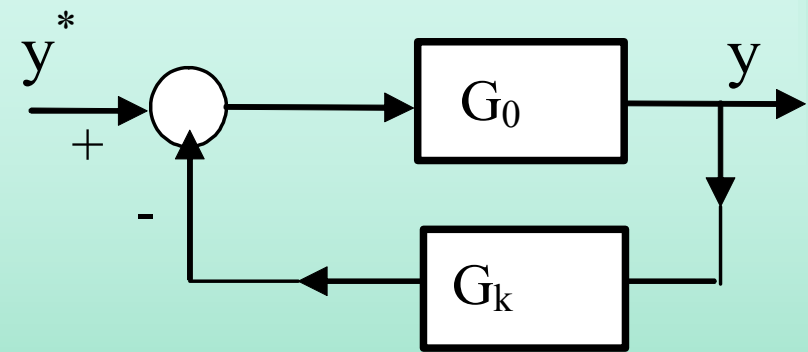
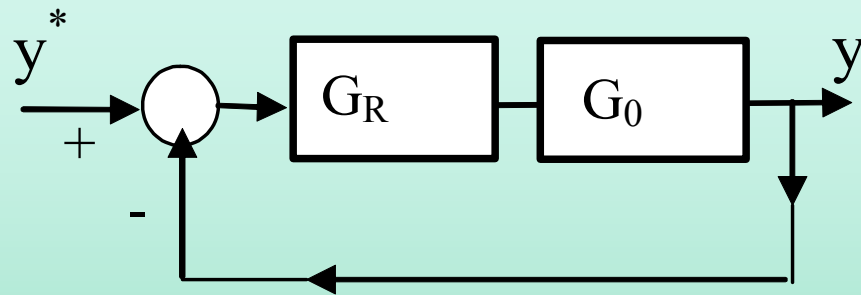


*Feedback  
Control*



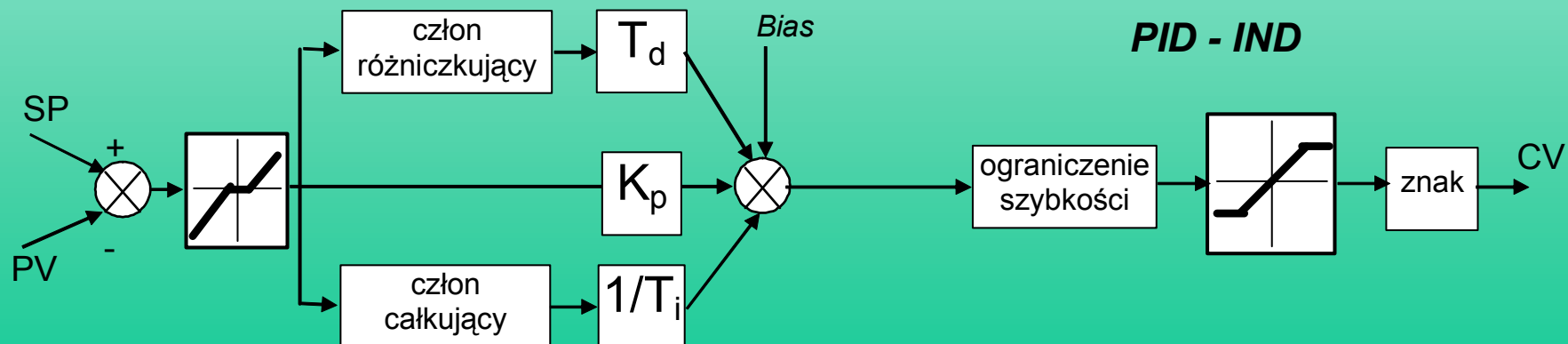
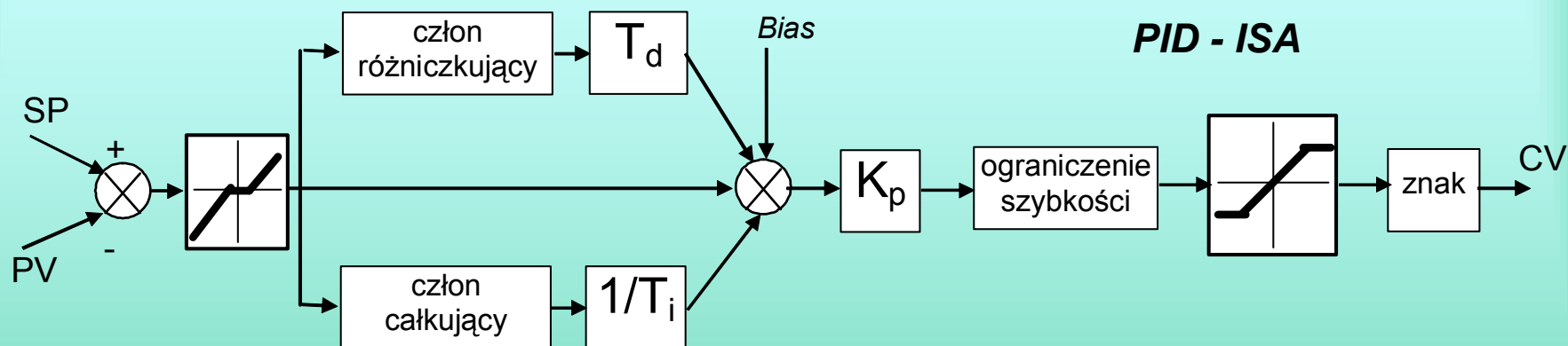
*Feedforward  
Control*

# Regulacja - korekcja

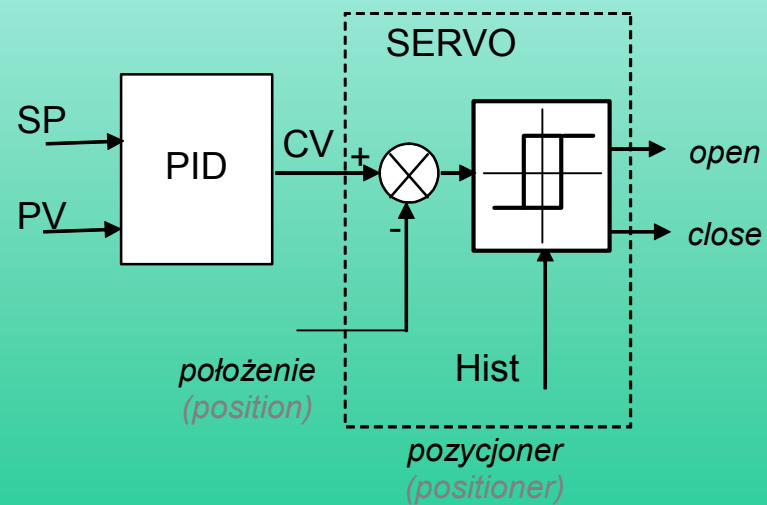
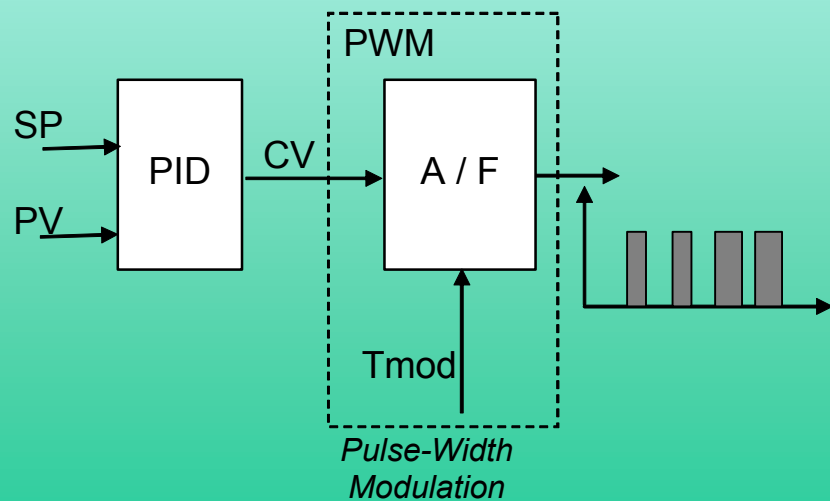
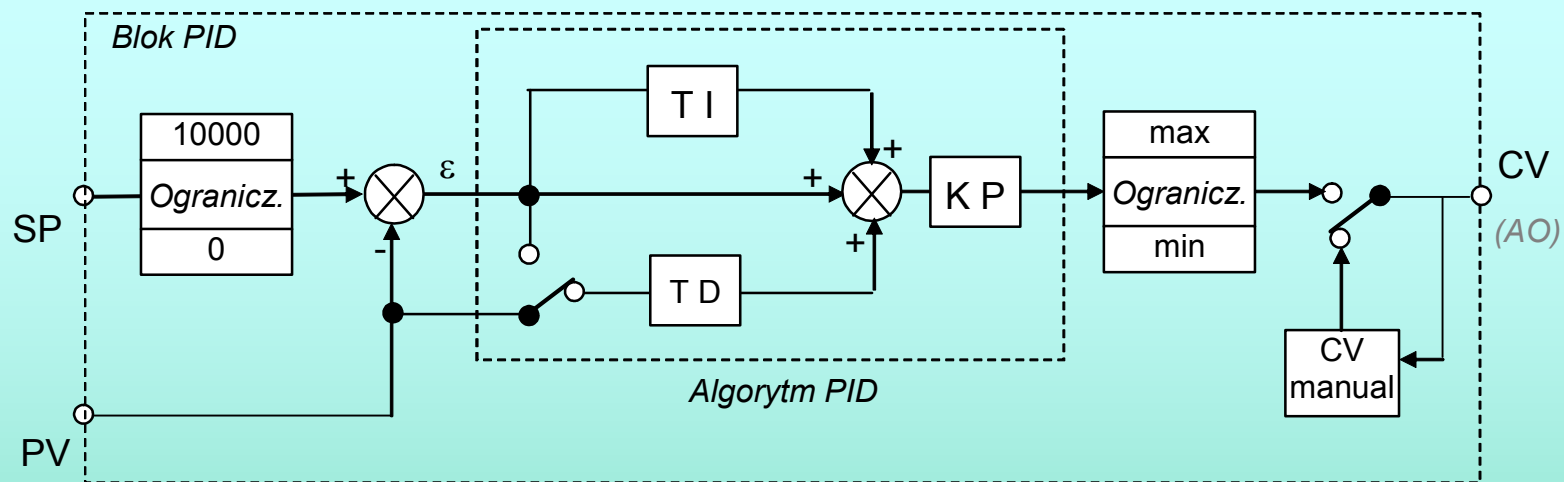


# Algorytm PID

$$u(t) = K_p \left[ e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right] + U_0$$



# PID / PL7 Micro Modicon



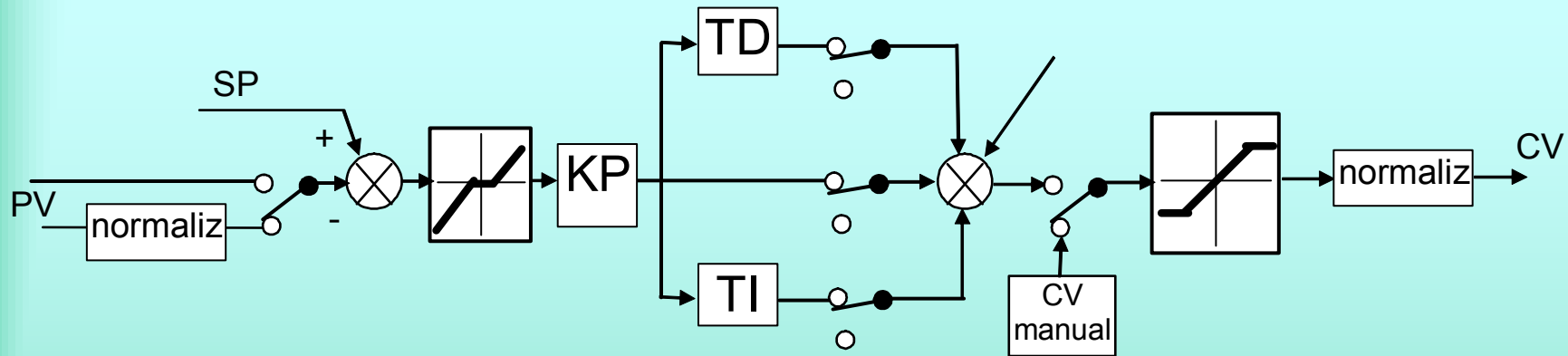




# Bloki PID w S7-300, S7-400

(Function blocks PID Control)

**FB 41 = CONT\_C** (continuous control)



**FB 42 = CONT\_S** (step control)

**FB 43 = PULSEGEN** (pulse generation)

**FB 58 = TCONT\_CP** (out: continue / pulse)

**FB 59 = TCONT\_S** (out: open, close)

# Moduły regulatorów

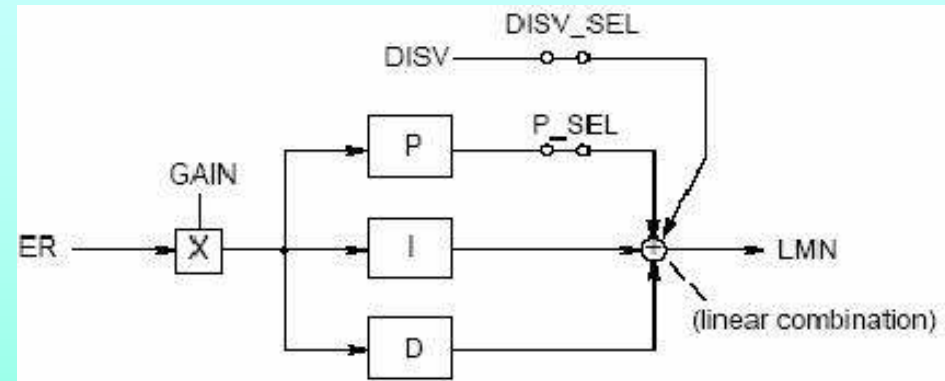
(Controller modules)

## S7-300

FM355C: 4 AI, 8 DI, 4 AO

FM355S: 4 AI, 8 DI, 8 DO

4 kanały

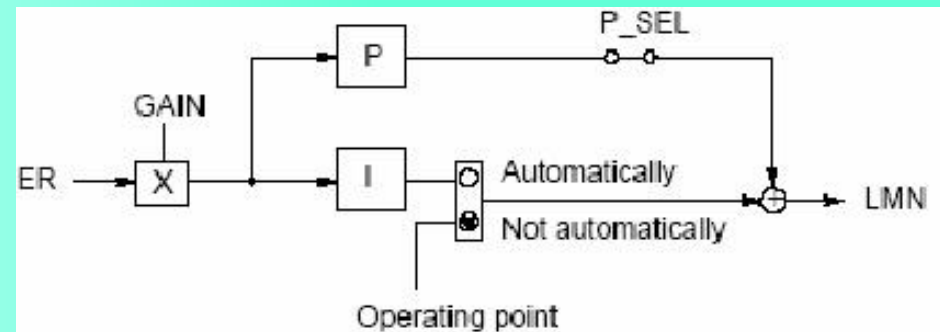
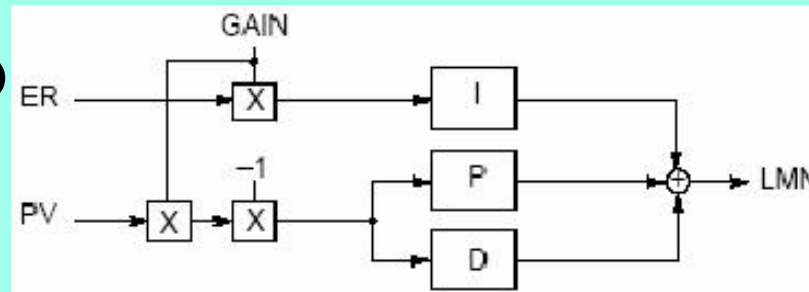


## S7-400

FM455C: 16 AI, 16 DI, 16 AO

FM455S: 16 AI, 16 DI, 32 DO

16 kanałów



# Konfiguracja regulatorów

- Struktura algorytmu PID
  - Różniczkowanie wartości wyjścia obiektu
- Strefa nieczułości
- Ograniczenie wartości na wyjściu
- Zabezpieczenie przed nasyceniem członu całkującego
- Ograniczenie szybkości wzrostu wartości wyjścia regulatora
- Stacyjka sterowania
- Typ sygnału wyjściowego (analogowy, impulsowy, binarny)
- **Struktury układów regulacji**

$$C = T_o/T:$$

$C < 0.05$  – łatwa ( $T_o < T/20$ )

$0.05 < C < 0.25$  – dość trudna

$C > 0.25$  – trudna

# Regulatory cyfrowe

(digital controllers)

## DDC – Direct Digital Control

- ❑ Honeywell - BC TDC 2000 - 1975
  - 8x PID, 10-modułowa kasetta
- ❑ regulatory kasetowe
  - Foxboro, Siemens, Fisher Controls, Moore, Bailey
  - Siemens TELEPERM M
- ❑ regulatory aparatowe (tablicowe)
  - P-200 Micon - 1978
- ❑ samonastrajanie i adaptacja (autotuning & adaptation)
  - Leeds-Northrup – 1981 (ciągła/okresowa korekcja nastaw)
  - Siemens TELEPERM M
  - Siemens SIPART - 1989

# Regulatory wielofunkcyjne

W Polsce:

- SIEMENS

**SIPART** - DR 20, DR 22, DR 24

- Zakład Produkcji Doświadczalnej Automatyki

**PSW** - RF-537, PSW-81-02, PSW-84-11,  
PSW-166-02, PSW-166-11

- CONTROLMATICA ZAP-PNEFAL

**EFTRONIC** - X (U486), V (U487), XP (U488),  
SO (U497), XF (U496)

- LABOR-ASTER

**As** - 520, 610

- LAB-EL Elektronika Laboratoryjna

**LB** - 600

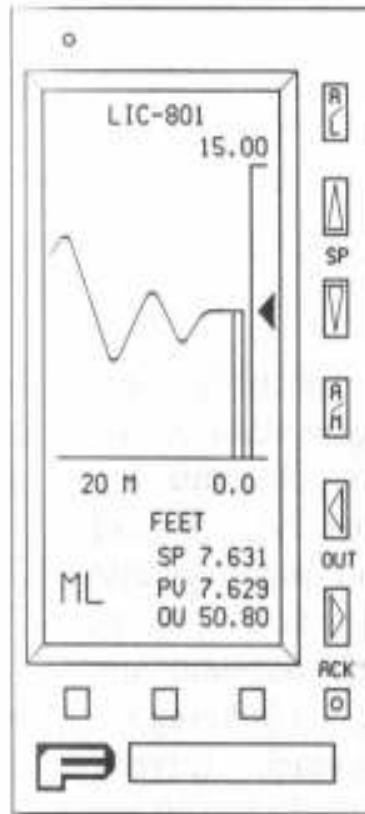
Na świecie:

- Bailey
- Foxboro.
- Micen
- Moore
- Kent-Taylor
- Toshiba
- Turnbull
- Simens
- Smar
- Yamatake-Honywell
- Yokogawa

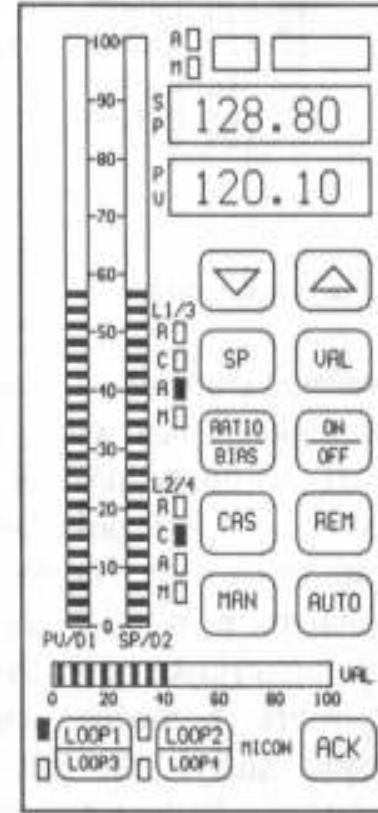
# Panel regulatora wielofunkcyjnego



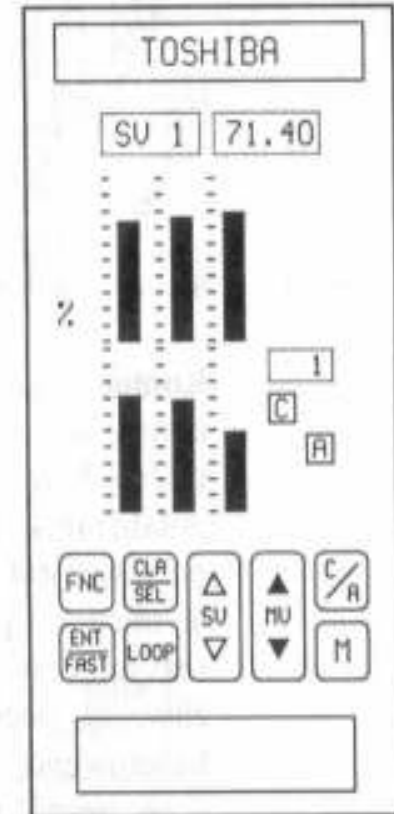
DR24 Sipart  
Siemens



MC5000  
Fisher Porter

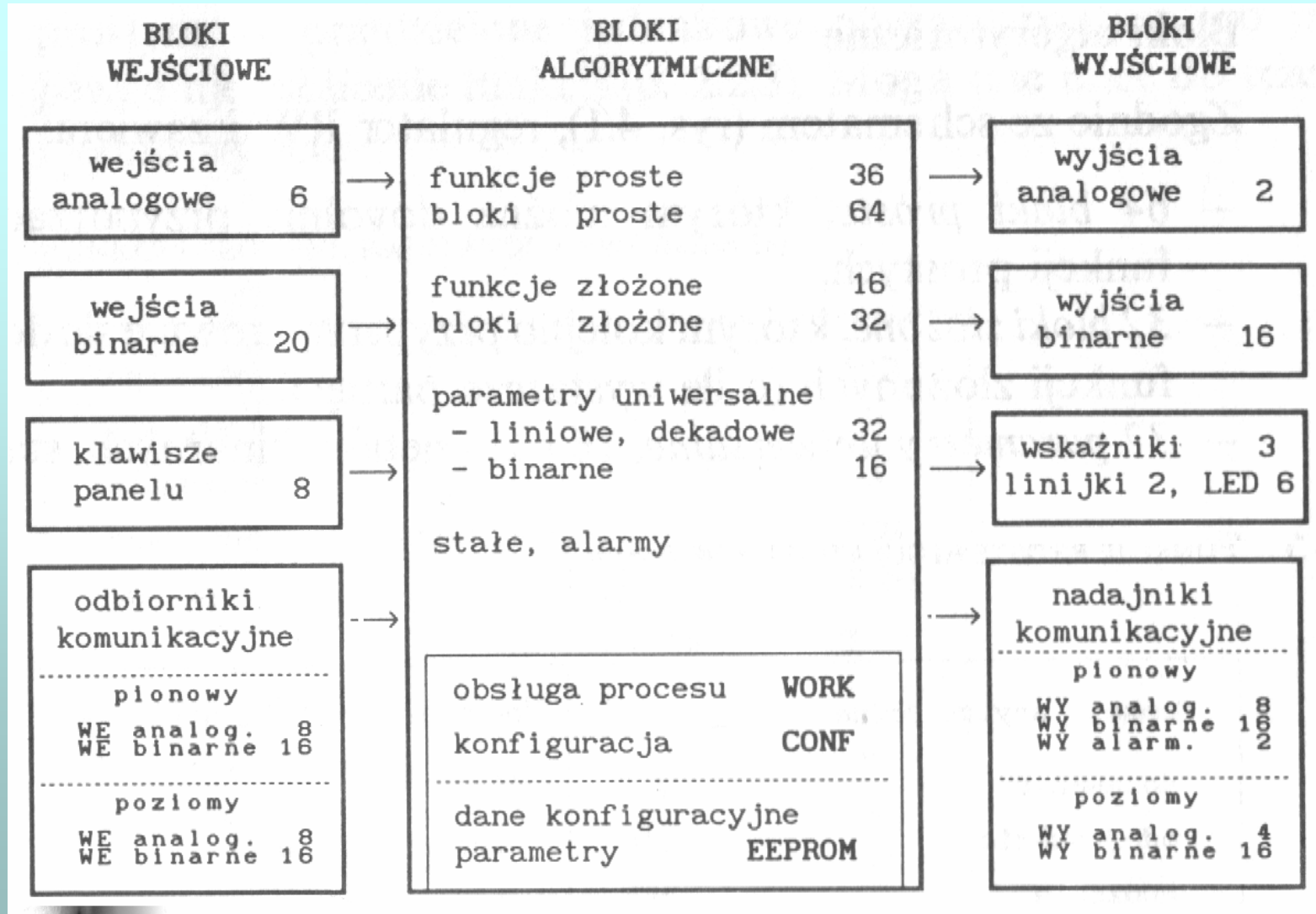


S-32  
Micon



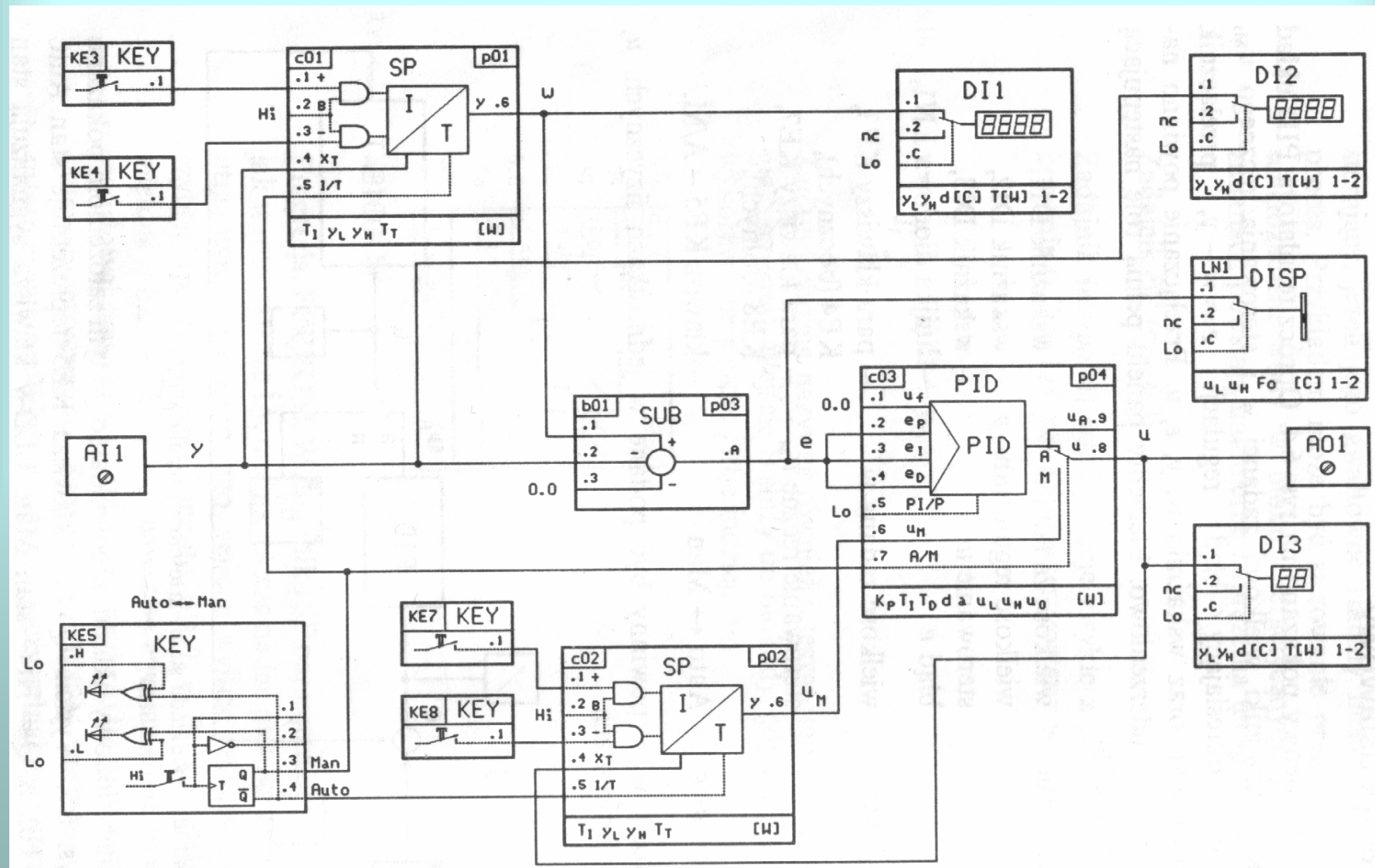
EC300  
Toshiba

# Schemat funkcjonalny regulatora wielofunkcyjnego



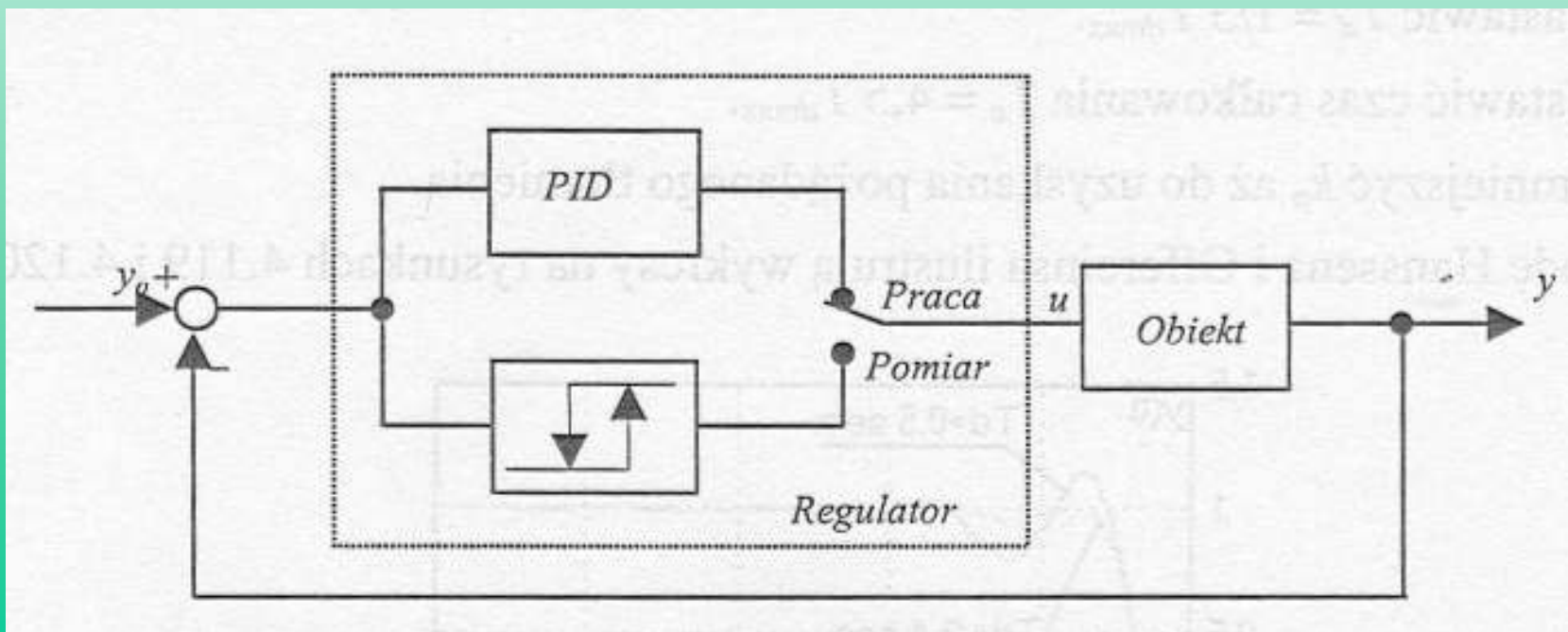


# Przykład konfiguracji regulatora wielofunkcyjnego

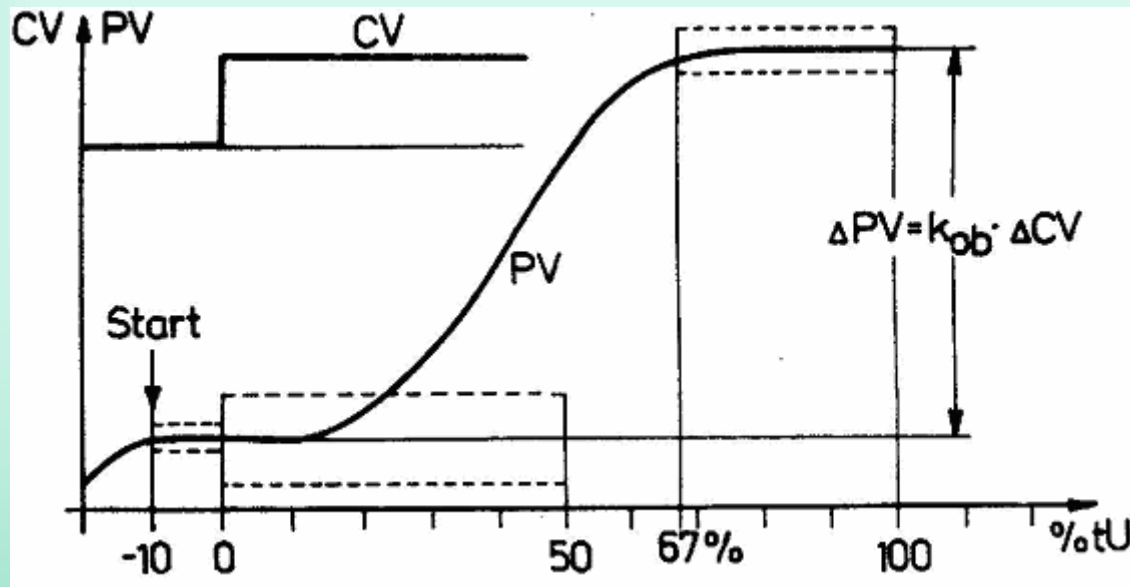


# Samonastrajanie (autotuning)

- 1 metoda Zieglera – Nicholosa (odpowiedzi skokowej)
  - identyfikacja modelu (przy otwartej pętli)
  - nastawy wg tabel dla różnych klas obiektów
- 2 metoda Zieglera – Nicholosa (metoda cyklu granicznego)
  - nastaw działanie proporcjonalne
  - zwiększaj wzmacnienie i odczytaj okres oscylacji
  - nastawy wg wzorów dla różnych regulatorów (modyfikacje Pessena, Opelta, Hanssena)



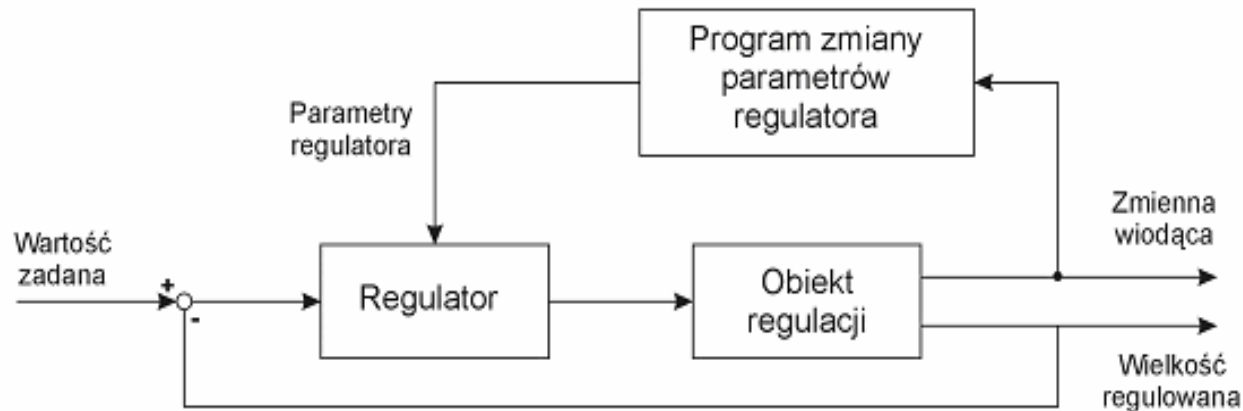
# SIPART - autotuning



- I. Sprawdzenie stałości PV przez okres  $0,1 t_u$
- II. Wmuszenie skokowe CV i zapis odpowiedzi skokowej
- III. Zbadanie warunków wiarygodności eksperymentu
- IV. Obliczenie parametrów modelu obiektu (model Strejca):  
$$G(s) = k / (Ts + 1)^n \quad \text{dla } 1 \leq n \leq 8$$
- V. Wylczenie nastaw i sugestia algorytmu (PI lub PID)
- VI. Wybór algorytmu i akceptacja nastaw przez operatora

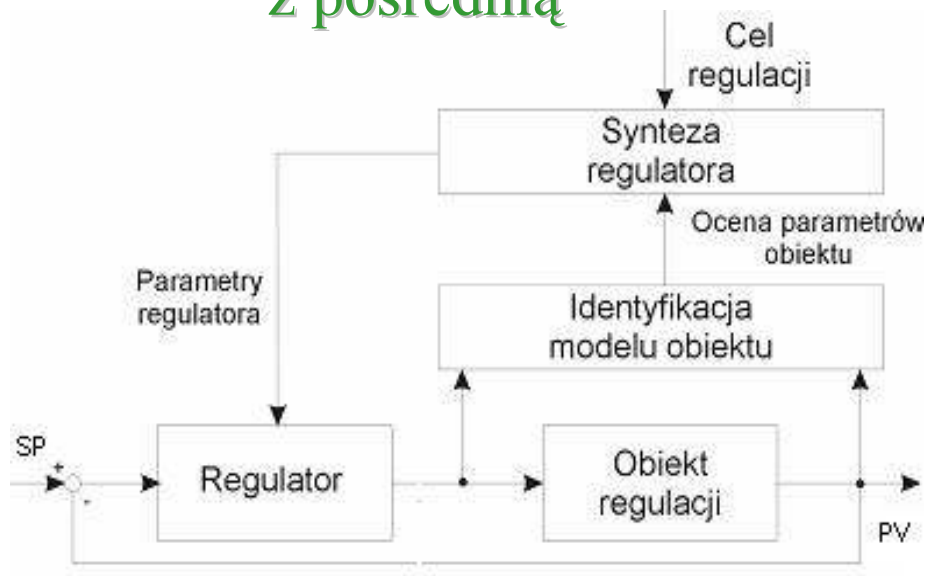
# Adaptacja

## Układ z programowanymi zmianami parametrów regulatora

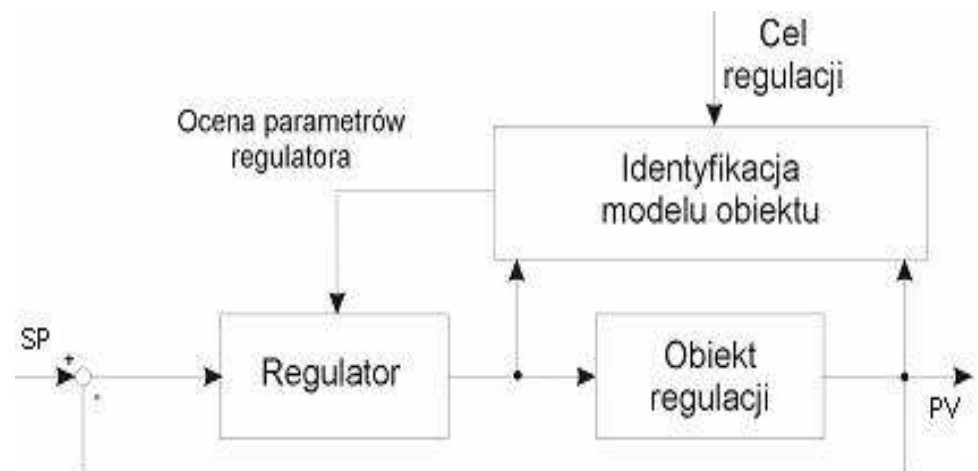


## Układy z identyfikacją modelu

### z pośrednią

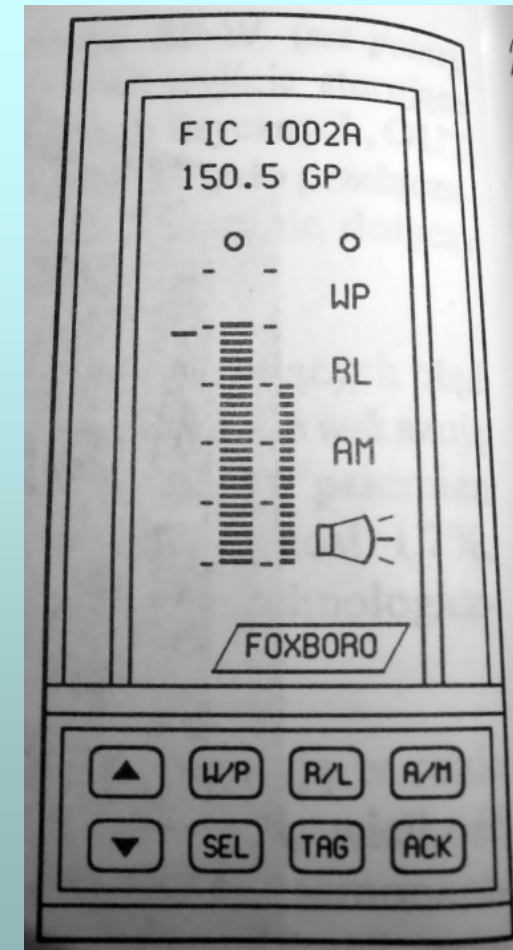


### z bezpośrednią



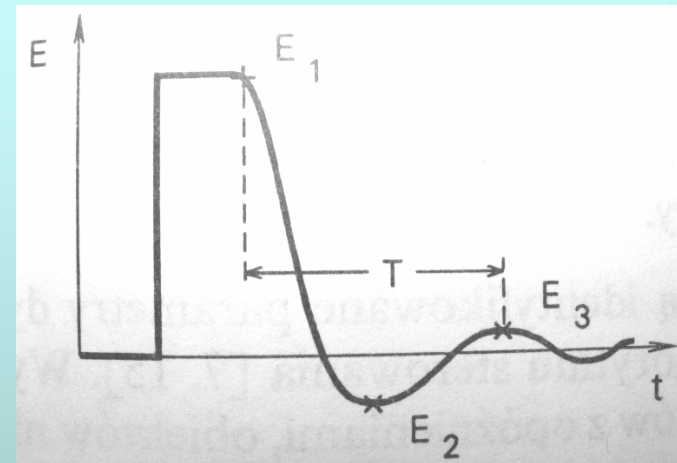
# *Algorytm adaptacyjny EXACT* *(zastosowany w regulatorze 761 Micro FOXBORO)*

1. Oczekiwanie na istotne zakłócenie tzn. o amplitudzie większej niż trzykrotny poziom szumów – start algorytmu EXACT;
2. Rejestracja odpowiedzi procesu na zakłócenie;
3. Obliczenie wartości nastaw regulatora PID;
4. Wprowadzenie obliczonych nastaw do regulatora – stop algorytmu EXACT;
5. Regulacja obiektu z nowymi nastawami, powrót do pkt. 1.



# Opis algorytmu EXACT

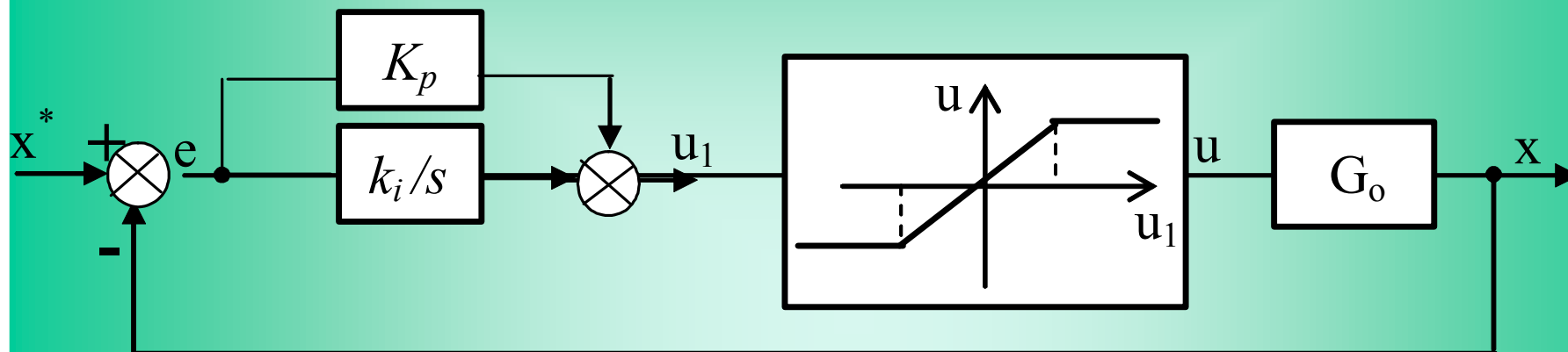
- odchyłka regulacji trzykrotnie większa od szumów - algorytm „budzi się”
- Rejestruje trzy kolejne amplitudy  $E_1$ ,  $E_2$ ,  $E_3$  oraz okres  $T$ .
- Wyznacza przeregulowanie  $o = E_1/E_2$  oraz tłumienie  $d = (E_2 + E_3)/(E_1 + E_2)$
- Porównuje  $o$ ,  $d$  z zadanymi wartościami  $o^*$ ,  $d^*$
- W razie konieczności koryguje nastawy



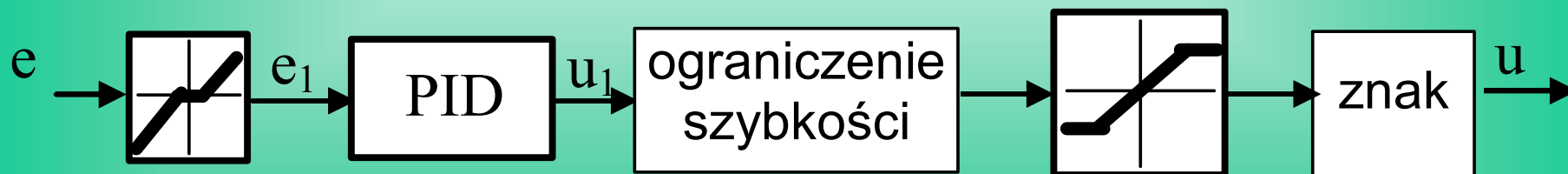
Zakłócenie

# Regulator ciągły + elementy nieliniowe

## Ograniczenie elementu wykonawczego

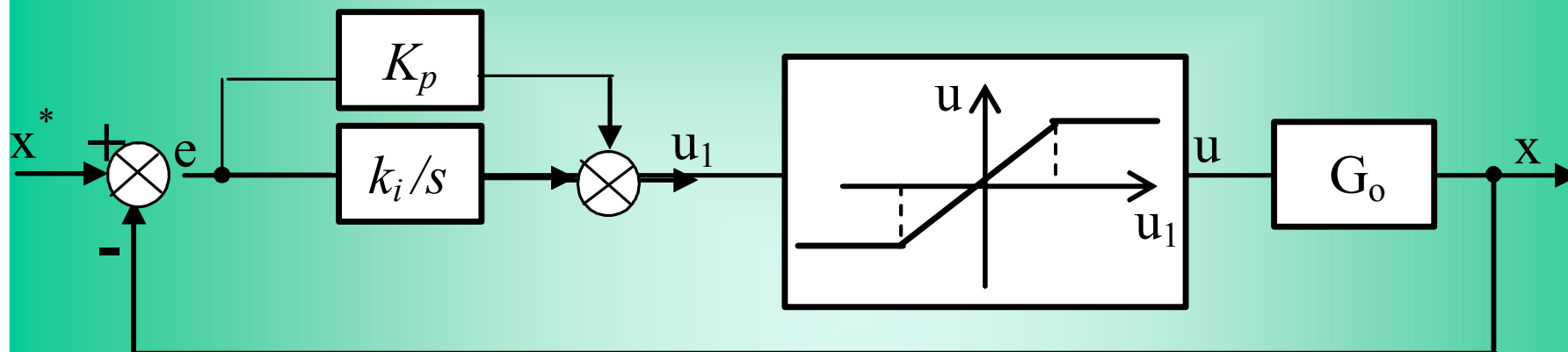


## Strefa martwa

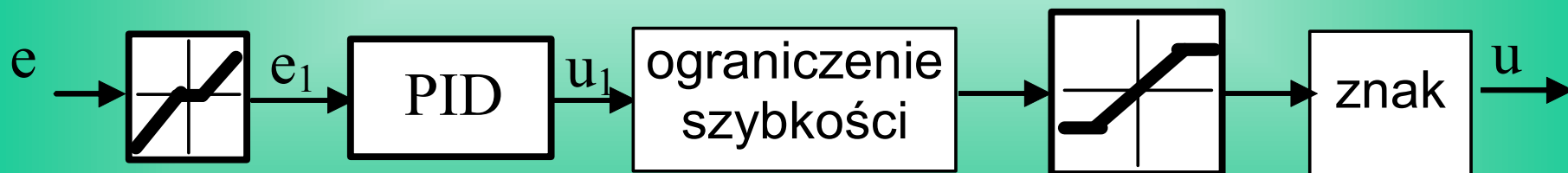


# Regulator ciągły + elementy nieliniowe

## Ograniczenie elementu wykonawczego



## Strefa martwa

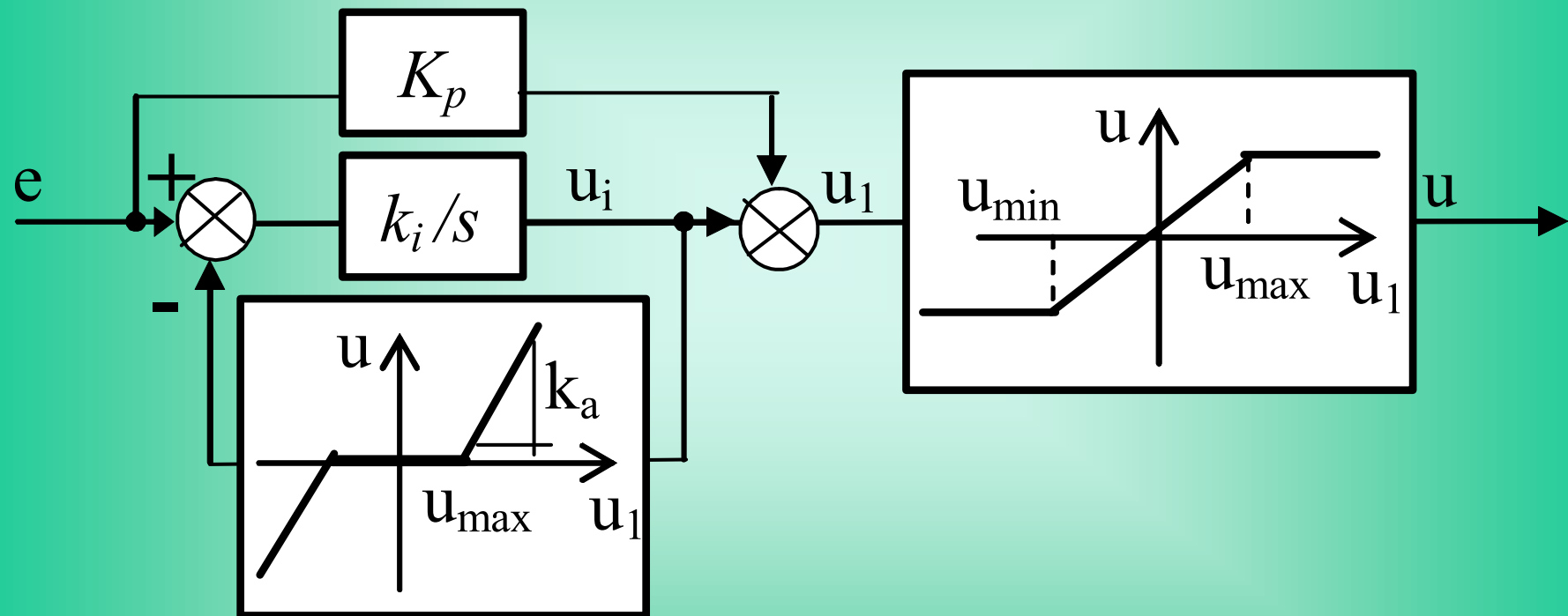




# Regulator ciągły + elementy nieliniowe

## Ograniczenie całkowania

integrator antiwindup circuit

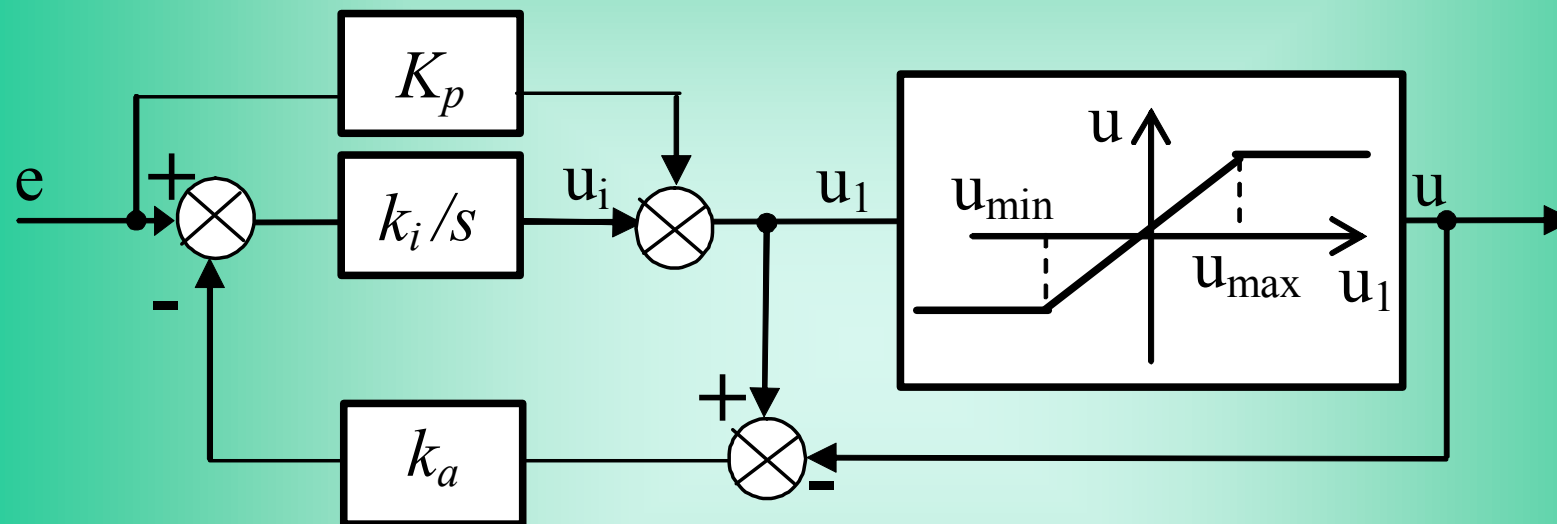


układ z dodatkową nieliniowością

# Regulator ciągły + elementy nieliniowe

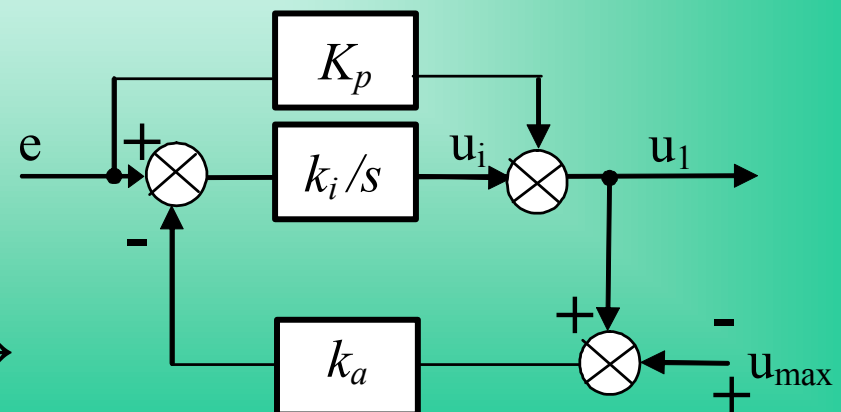
## Ograniczenie całkowania

integrator antiwindup circuit



układ z wykorzystaniem istniejącego nasycenia

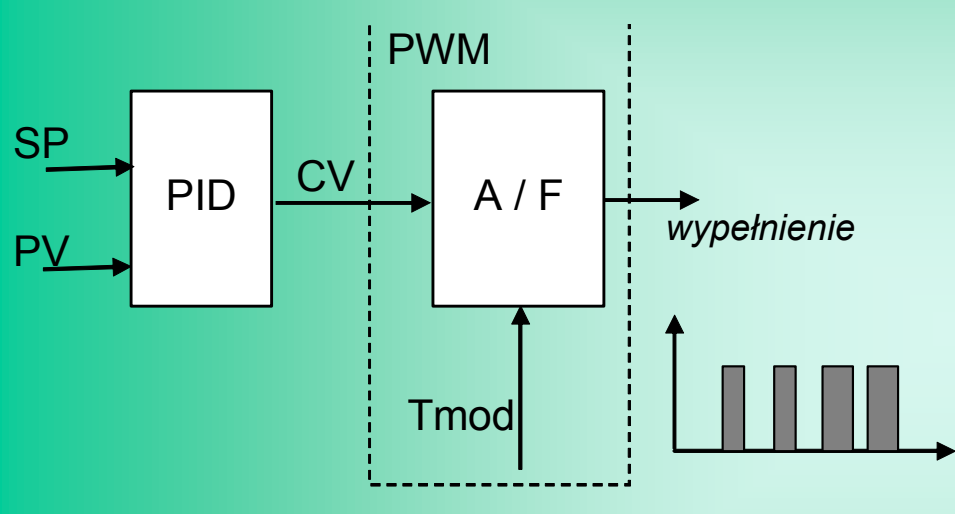
w czasie nasycenia →



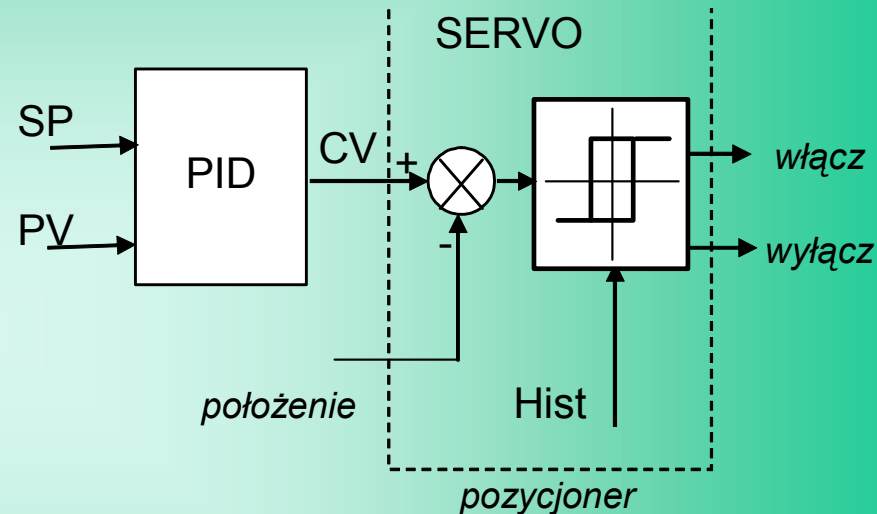
# Regulator ciągły + elementy nieliniowe

## Regulator ciągły z wyjściem dyskretnym

wyjście dwustanowe



wypełnienie

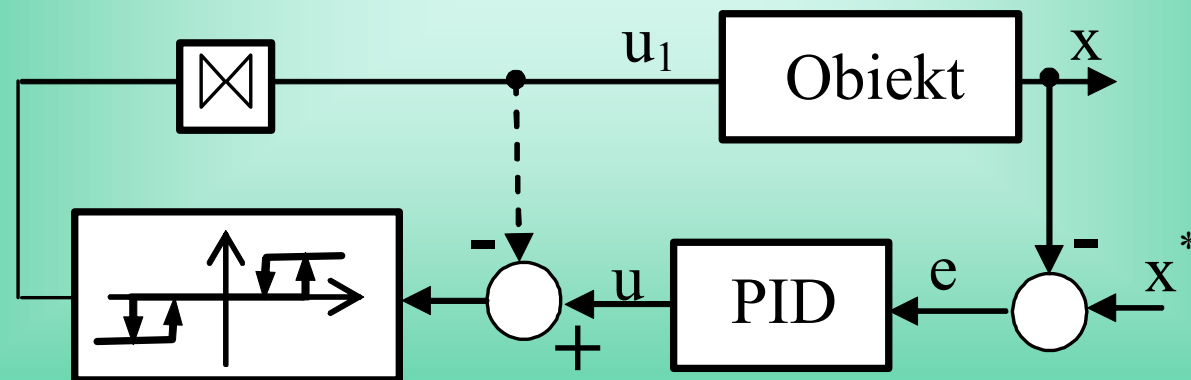
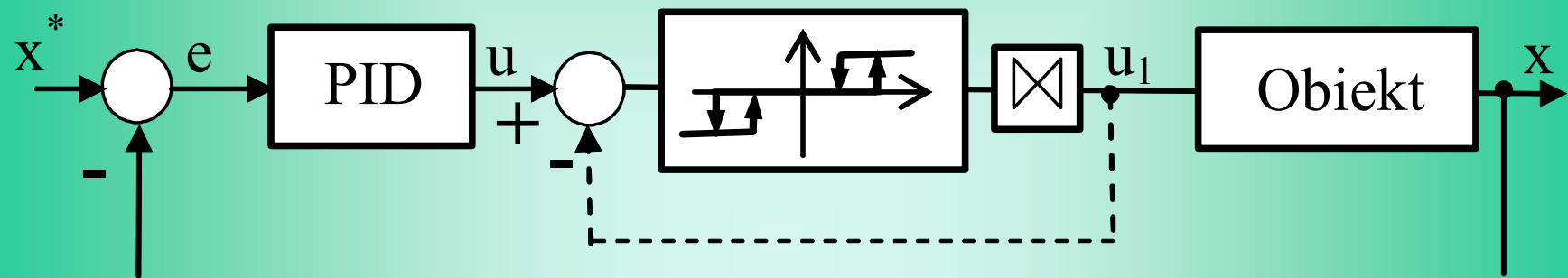


włącz/wyłącz

# Regulator ciągły + elementy nieliniowe

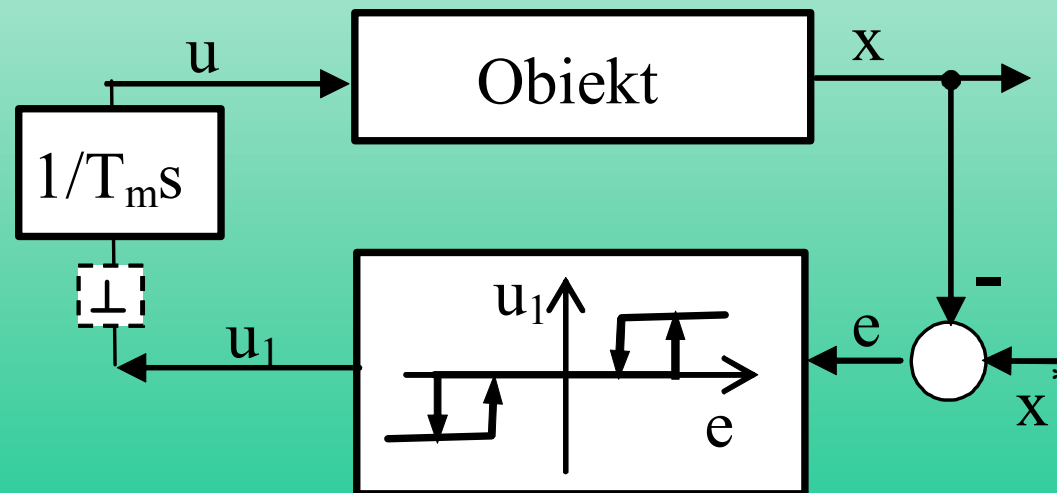
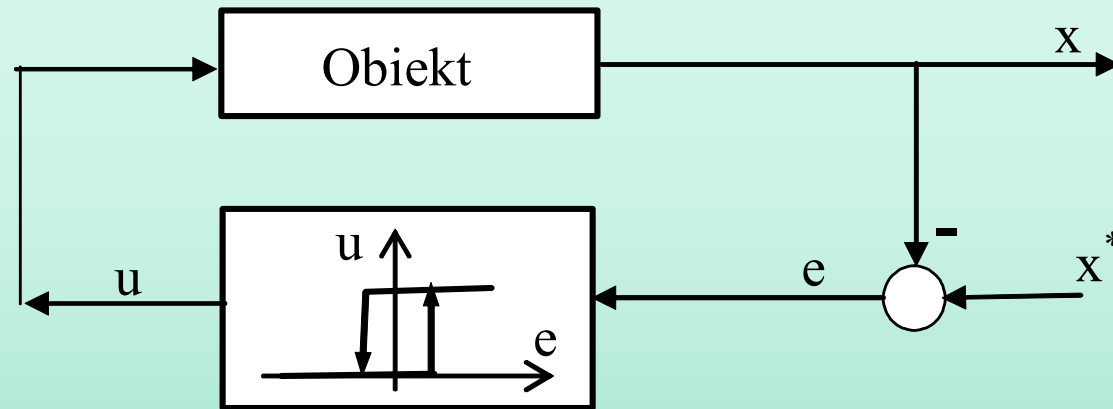
## Regulator ciągły z wyjściem dyskretnym

wyjście trójstanowe

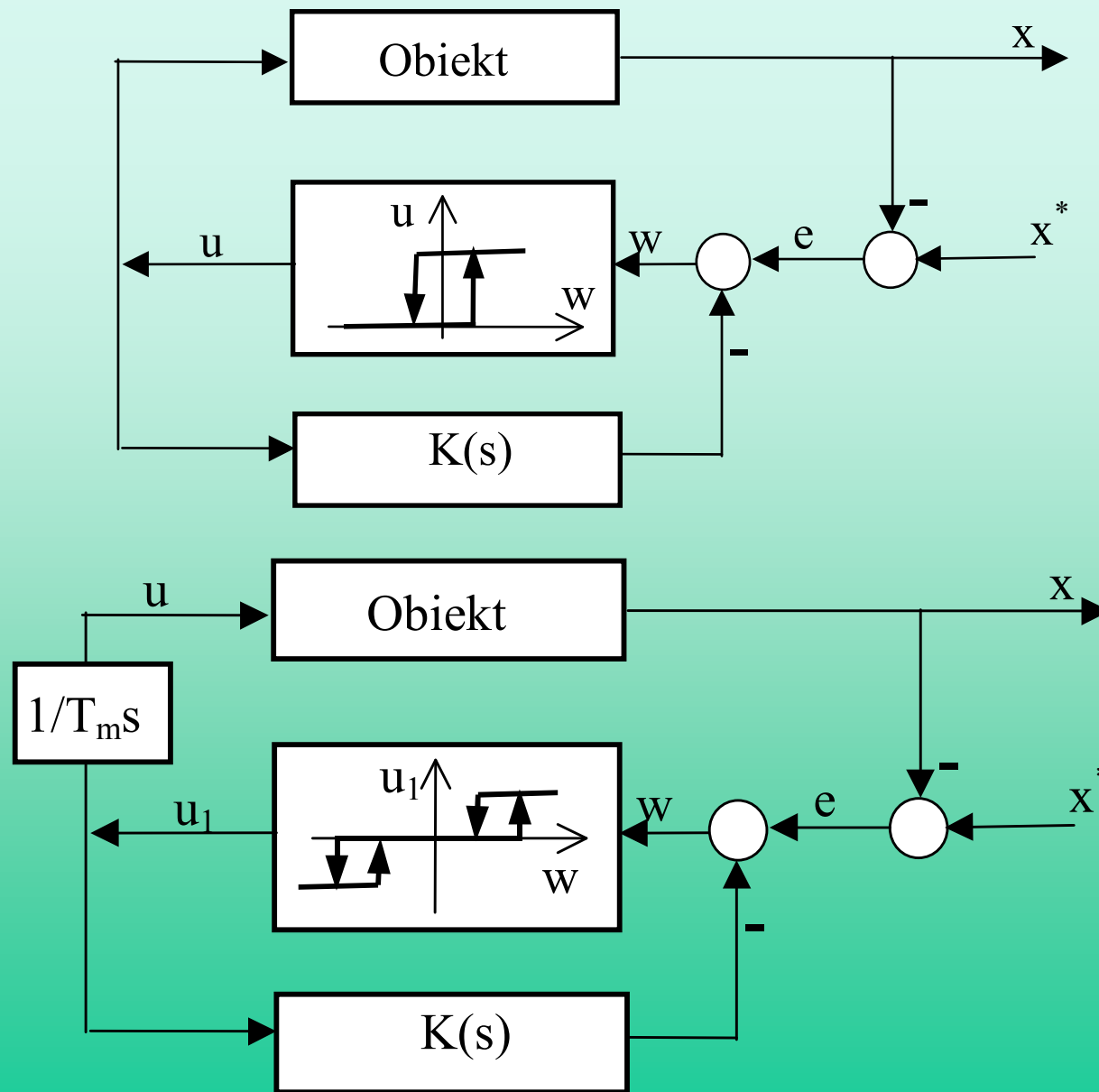


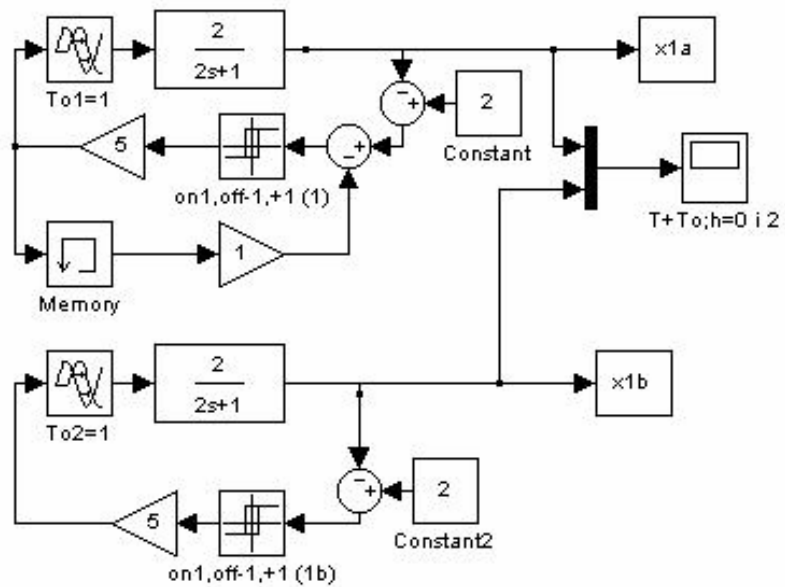
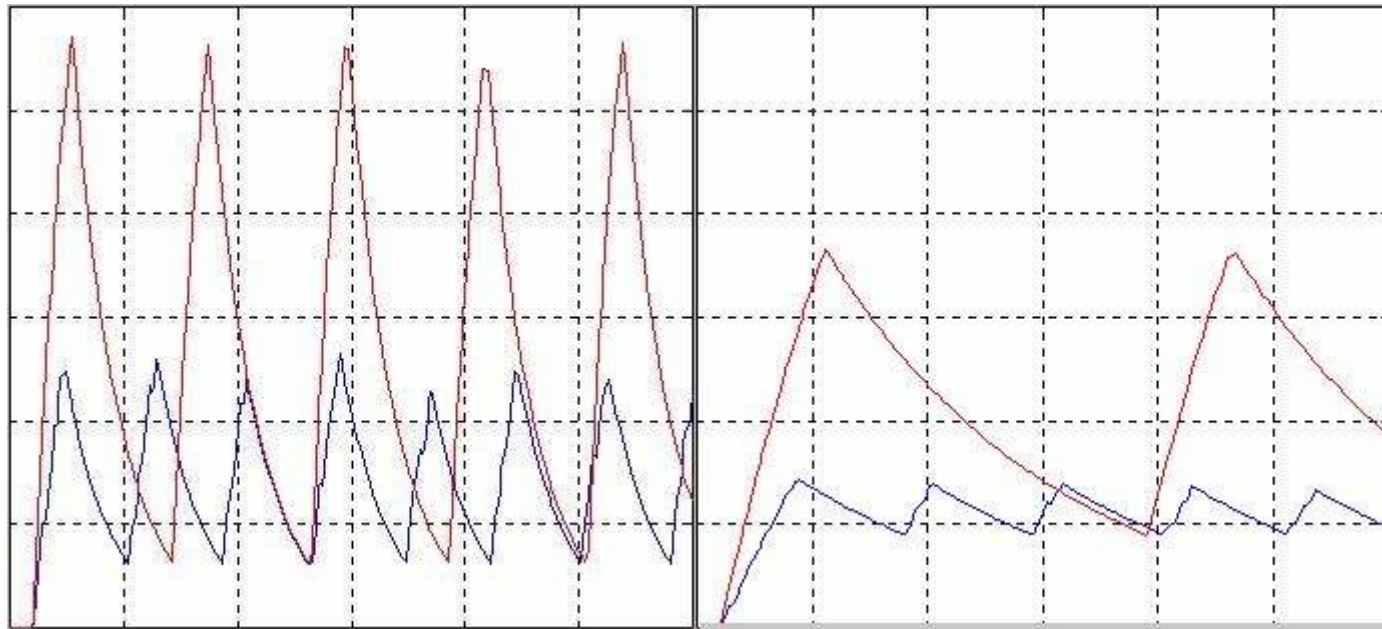
pozycjoner trójstanowy (zamykanie-stop-otwieranie)

# Regulatory przekąźnikowe

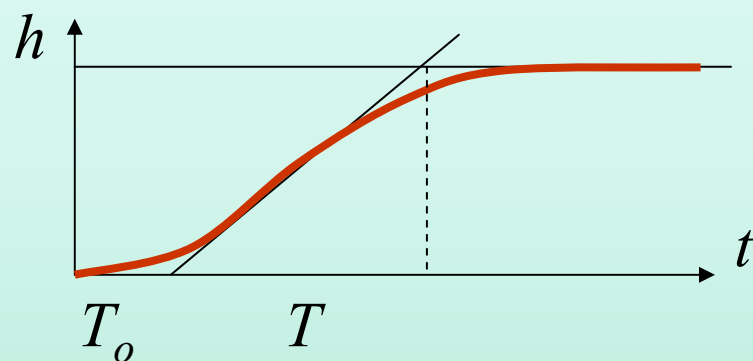


# Regulatory przekąźnikowe + korekcja





## Wybór typu regulatora - zalecenia



$$k_r = T_o / T$$

$k_r$		Regulacja	Typ regulatora
$k_r < \frac{1}{20}$	$T > 20T_o$	łatwa	regulator dwupołożeniowy jeśli tylko obiekt „mocno” uśrednia
$\frac{1}{20} < k_r < \frac{1}{4}$		dość trudna	regulator PI lub PID
$k_r > \frac{1}{4}$	$T < 4T_o$	trudna	



# Stacje

## rozproszonego systemu automatyki

### ❖ stacje procesowe

- pomiary, sterowanie, zabezpieczenia

### ❖ stacje operatorskie

- wizualizacja

### ❖ stacja inżynierska

- konfiguracja/programowanie

### ❖ stacje robocze

- baza danych, archiwacja, raporty

# Stacje procesowe w systemie

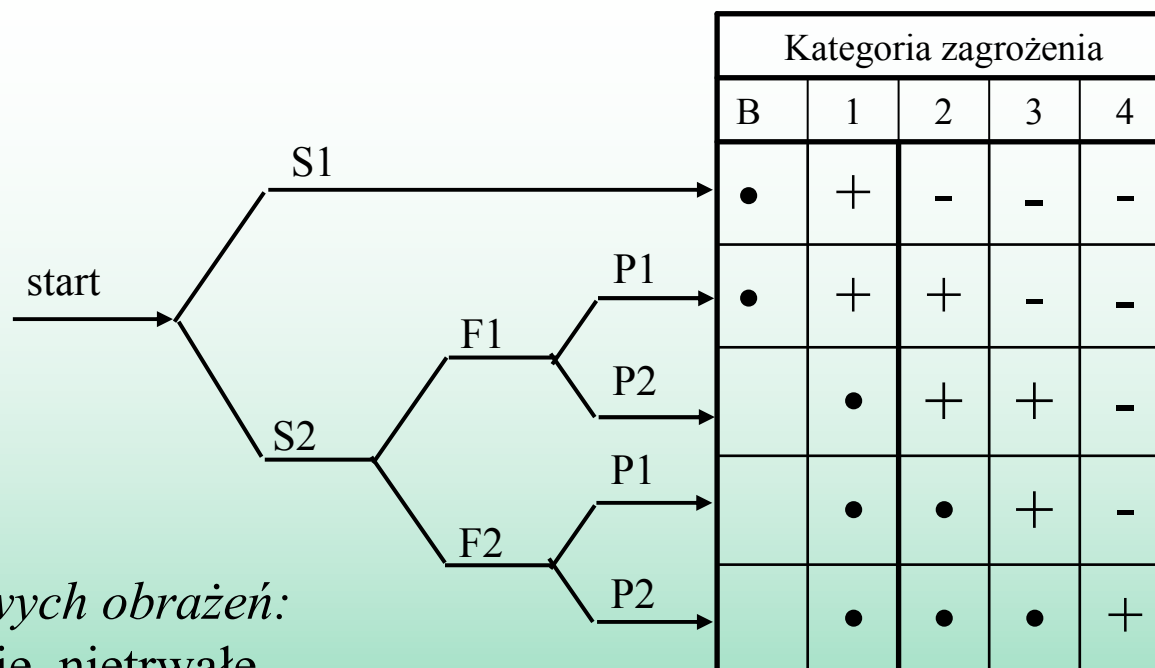
## ➤ Komunikacja:

- bezpośrednio
- wspólne obszary pamięci
- sieci
- oddalone I/O

## ➤ Bezpieczeństwo:

- Redundancja
  - kanałów, modułów, łączy komunikacyjnych, sterowników
- Analiza ryzyka

# Graf zaszeregowania zagrożeń i analizy ryzyka



*stopień możliwych obrazów:*

S1 – lekkie, nietrwałe

S2 – ciężkie

*częstość / czas trwania ekspozycji:*

F1 – rzadko / krótki czas

F2 – często / długi czas

*możliwość uniknięcia zagrożenia:*

P1 – możliwe pod pewnymi warunkami

P2 – prawie niemożliwe

+ kategoria zalecana

• kategoria dopuszczalna  
(z dodatkowymi zabezpieczeniami)

- przewymiarowane zabezpieczenia