

UKŁADY SEKWENCYJNO-CZASOWE

Zbiornik z dwoma zaworami. Podnośnik góra-dół z nawrotem. Reaktor chemiczny.

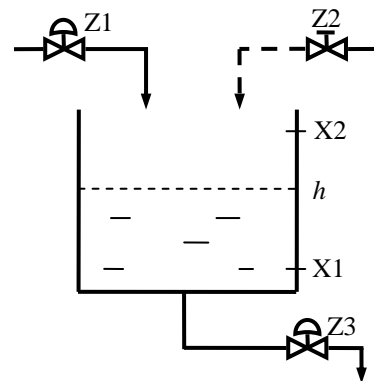
ZBIORNIK Z DWOMA ZAWORAMI

1. Zadanie

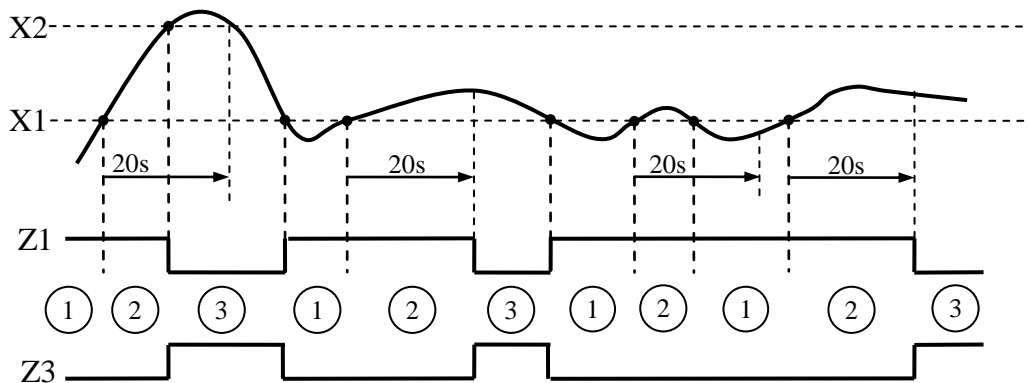
Zbiornik jest napełniany zaworem Z1, a opróżniany zaworem Z3. Przebiega to następująco:

- Dla $h < X1$ otwarty jest zawór Z1.
- Jeśli h przekroczy $X1$, to Z1 jest otwarty przez 20 sekund, chyba że wcześniej h osiągnie $X2$ (wtedy Z1 zamyka się natychmiast).
- Po zamknięciu Z1 otwierany jest Z3, aż h spadnie do $X1$.

Dodatkowy zawór Z2 jest otwierany ręcznie.



2. Przebiegi czasowe

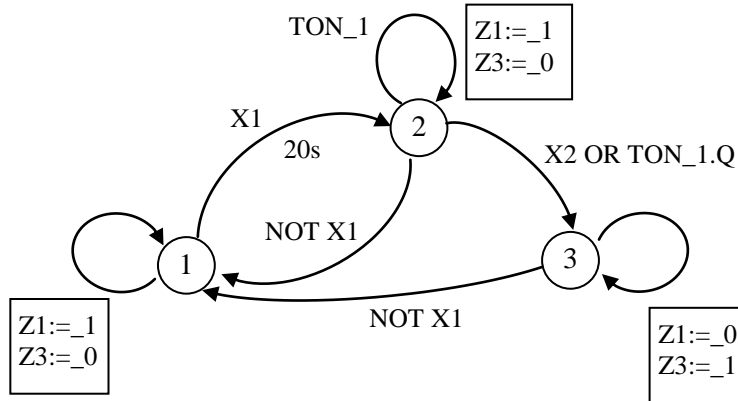


3. Automat

- Stany

- ① nalewanie, gdy $h < X1 \rightarrow Z1 = 1$
- ② nalewanie, gdy $h > X1$, ale nie dłużej niż przez 20 sekund lub do momentu, gdy h osiągnie $X2$ ($h = X2$) $\rightarrow Z1 = 1$
- ③ oczekiwanie na opróżnienie, tzn. gdy $h = X1 \rightarrow Z1 = 0$

- Graf



4. Kod

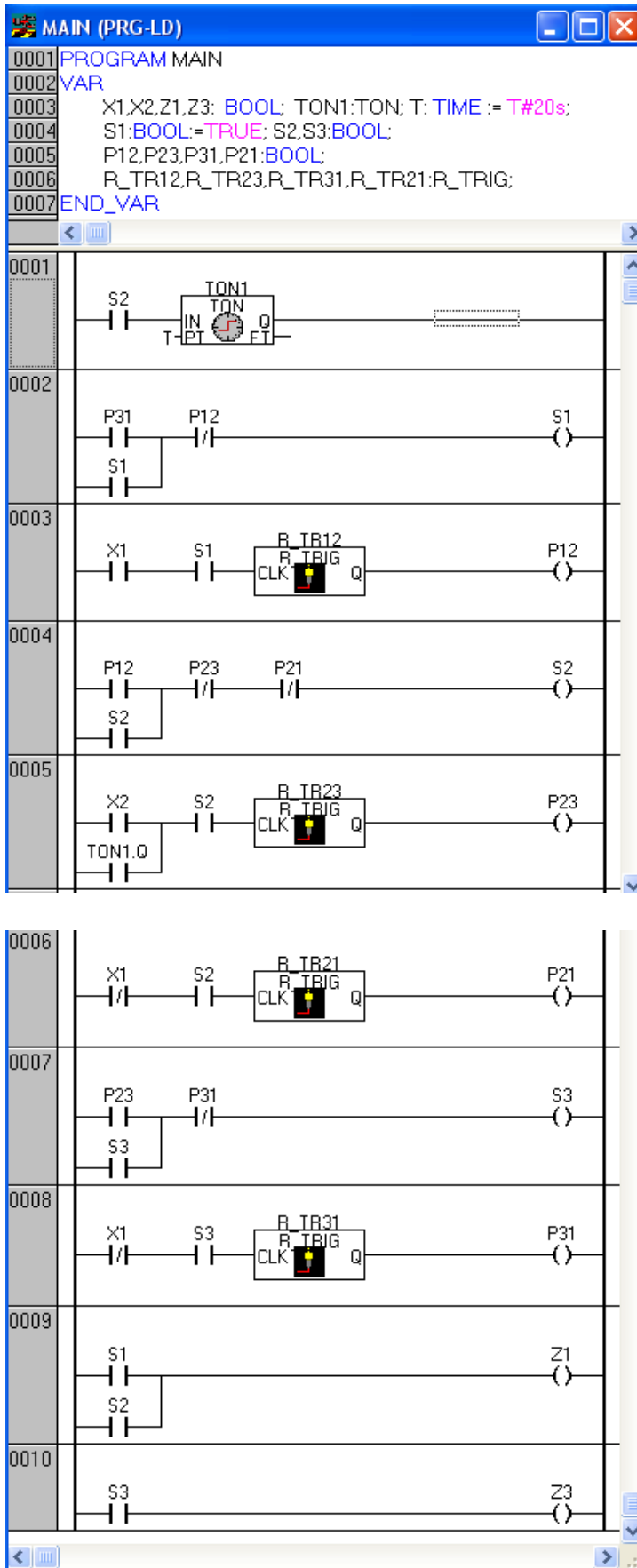
ST

```

MAIN (PRG-ST)
0001 PROGRAM MAIN
0002 VAR
0003   X1,X2,Z1,Z3: BOOL; stan: INT:=1;
0004   TON_1:TON;
0005   T: TIME := T#20s;
0006   _0:BOOL:=FALSE; _1:BOOL:=TRUE;
0007 END_VAR

0001 TON_1(IN:=stan=2, PT:=T);
0002
0003 CASE stan OF
0004 1: Z1:=_1; Z3:=_0;
0005   IF X1 THEN stan:=2; END_IF
0006 2: Z1:=_1; Z3:=_0;
0007   IF NOT X1 THEN stan:=1;
0008   ELSIF X2 OR TON_1.Q THEN stan:=3; END_IF
0009 3: Z1:=_0; Z3:=_1;
0010   IF NOT X1 THEN stan:=1; END_IF
0011 END_CASE
  
```

LD



PODNOŚNIK GÓRA-DÓŁ Z NAWROTEM

1. Zadanie

Sygnaly obsługujące podnośnik są następujące:

- Wejścia – przyciski w górę, w dół: *pg*, *pd*
– wyłączniki krańcowe górny, dolny: *kg*, *kd*
- Wyjścia – sterowania jazda w górę, w dół: *jpg*, *jpg*.

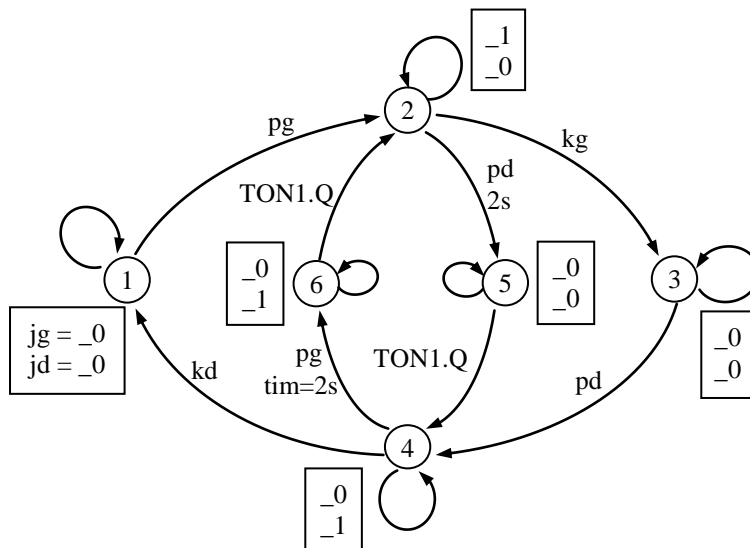
Naciśnięcie przycisku jazdy w przeciwną stronę podczas ruchu podnośnika powoduje zatrzymanie go na 2 sekundy, a potem zmianę kierunku.

2. Automat

- Stany

- 1 postój na dole
- 2 jazda w górę
- 3 postój na górze
- 4 jazda na dół
- 5 postój 2 s po przyciśnięciu „w dół” podczas jazdy w górę
- 6 postój 2 s po przyciśnięciu „w górę” podczas jazdy na dół

- Graf

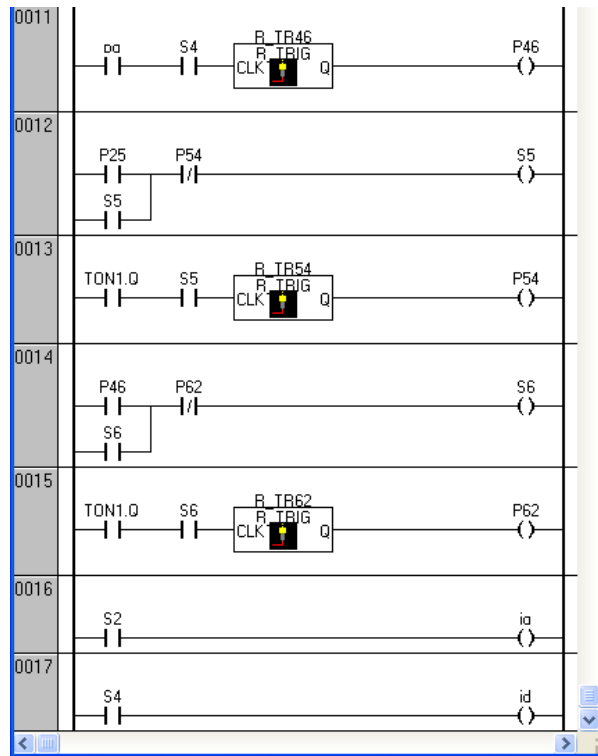
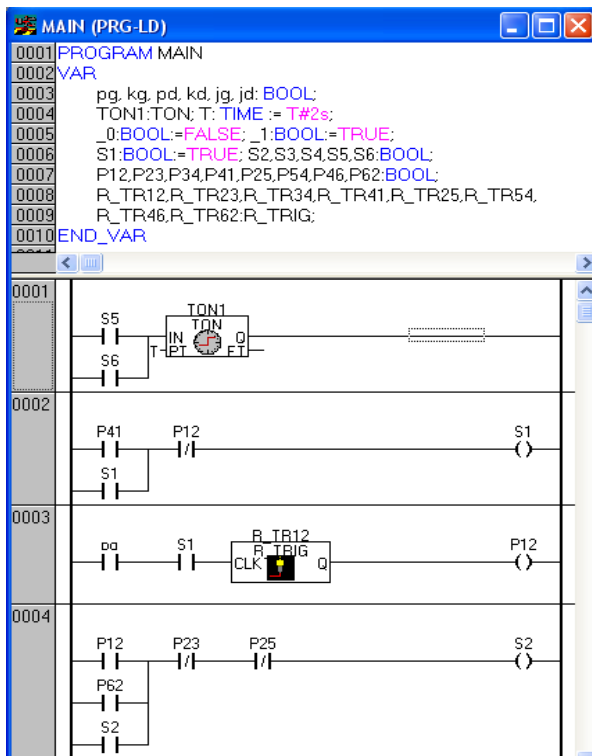


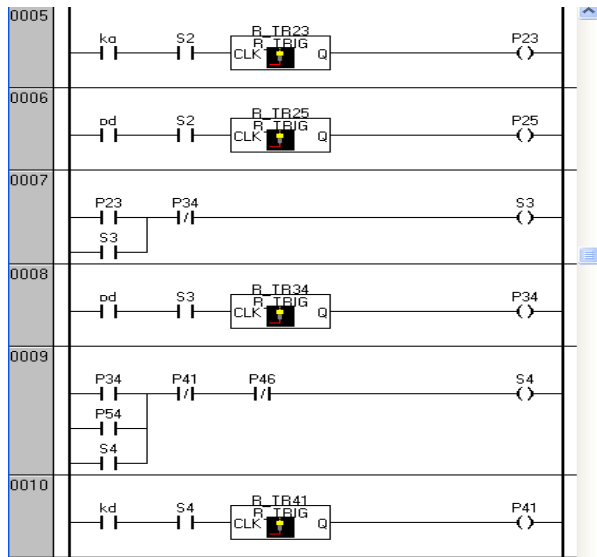
3. Kod

ST

```
MAIN (PRG-ST)
0001 PROGRAM MAIN
0002 VAR
0003   pg, kg, pd, kd, jg, jd: BOOL;
0004   stan: INT:=1;
0005   TON1:TON; T:TIME := T#2s;
0006   _0:BOOL:=FALSE; _1:BOOL:=TRUE;
0007 END_VAR
0008
0001 TON1(IN:=stan=5 OR stan=6, PT:=T);
0002
0003 CASE stan OF
0004 1: jg:=jd=_0;
0005   IF pg THEN stan:=2; END_IF
0006 2: jg=_1; jd=_0;
0007   IF kg THEN stan:=3;
0008   ELSIF pd THEN stan:=5; END_IF
0009 3: jg:=jd=_0;
0010   IF pd THEN stan:=4; END_IF
0011 4: jg=_0; jd=_1;
0012   IF kd THEN stan:=1;
0013   ELSIF pg THEN stan:=6; END_IF
0014 5: jg:=jd=_0;
0015   IF TON1.Q THEN stan:=4; END_IF
0016 6: jg:=jd=_0;
0017   IF TON1.Q THEN stan:=2; END_IF
0018 END_CASE
```

LD



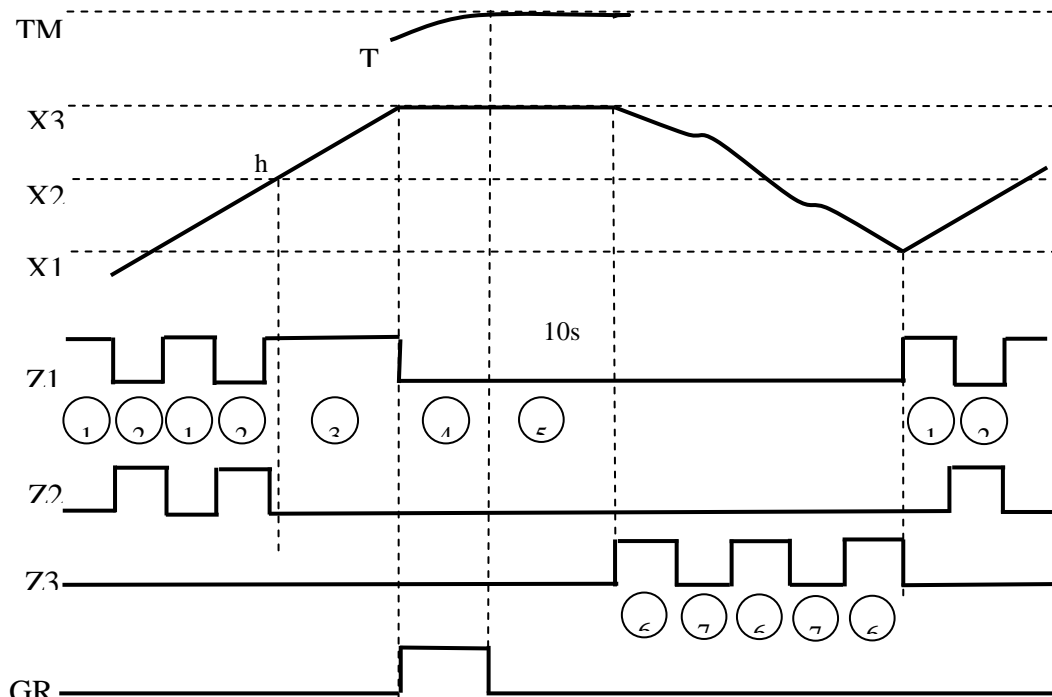
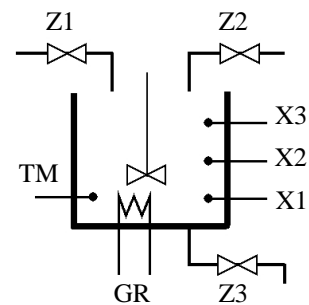


REAKTOR CHEMICZNY

1. Zadanie

Należy zaprojektować układ sterowania reaktorem funkcjonujący według poniższego algorytmu:

- Nalewanie na zmianę z zaworów Z1/Z2 w cyklu 3+3 s do poziomu X2
- Nalewanie dalej z Z1 aż do X3 (Z2 zamknięty)
- Zamknięcie Z1, włączenie grzejnika GR, aż temperatura osiągnie TM
- Wyłączenie GR i oczekiwanie przez 10 s.
- Opróżnianie z Z3 w cyklu 4+2 s (otw./zamkn.) do X1
- Zamknięcie Z3 i powrót na początek (nalewanie)

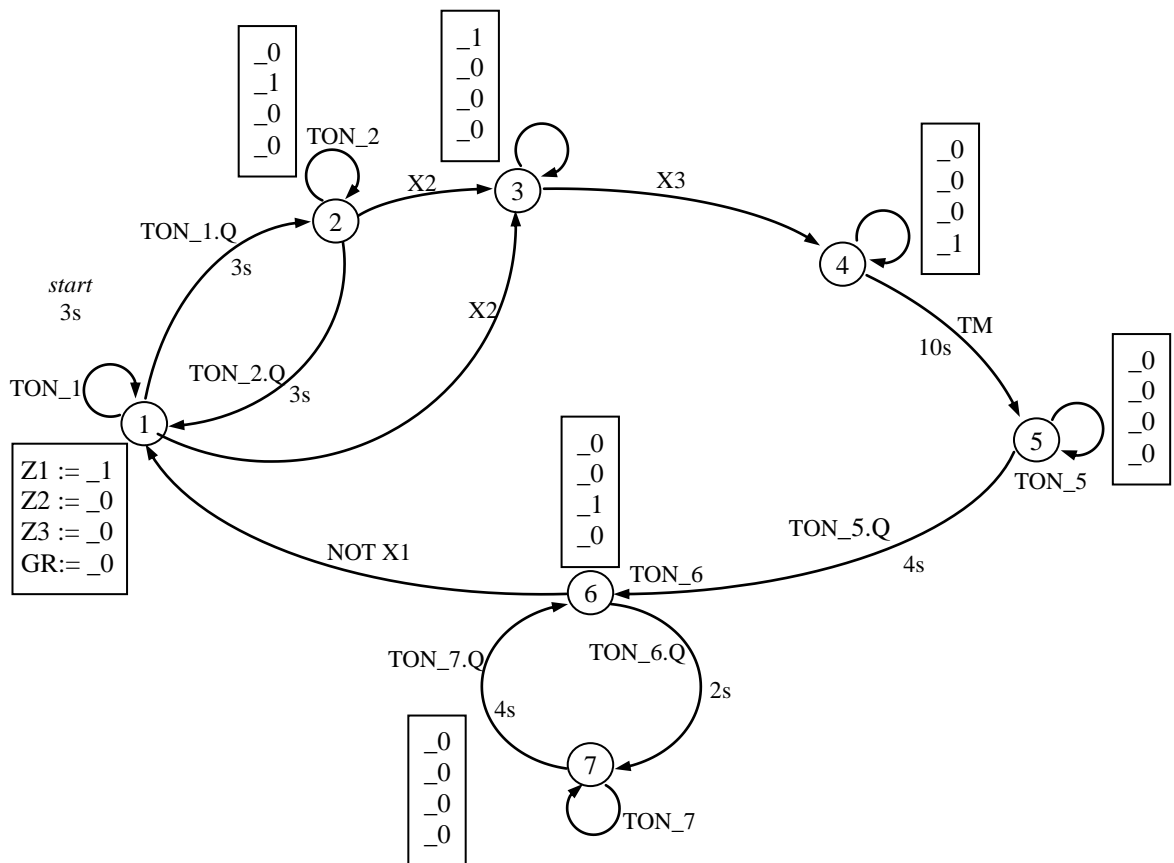


2. Automat

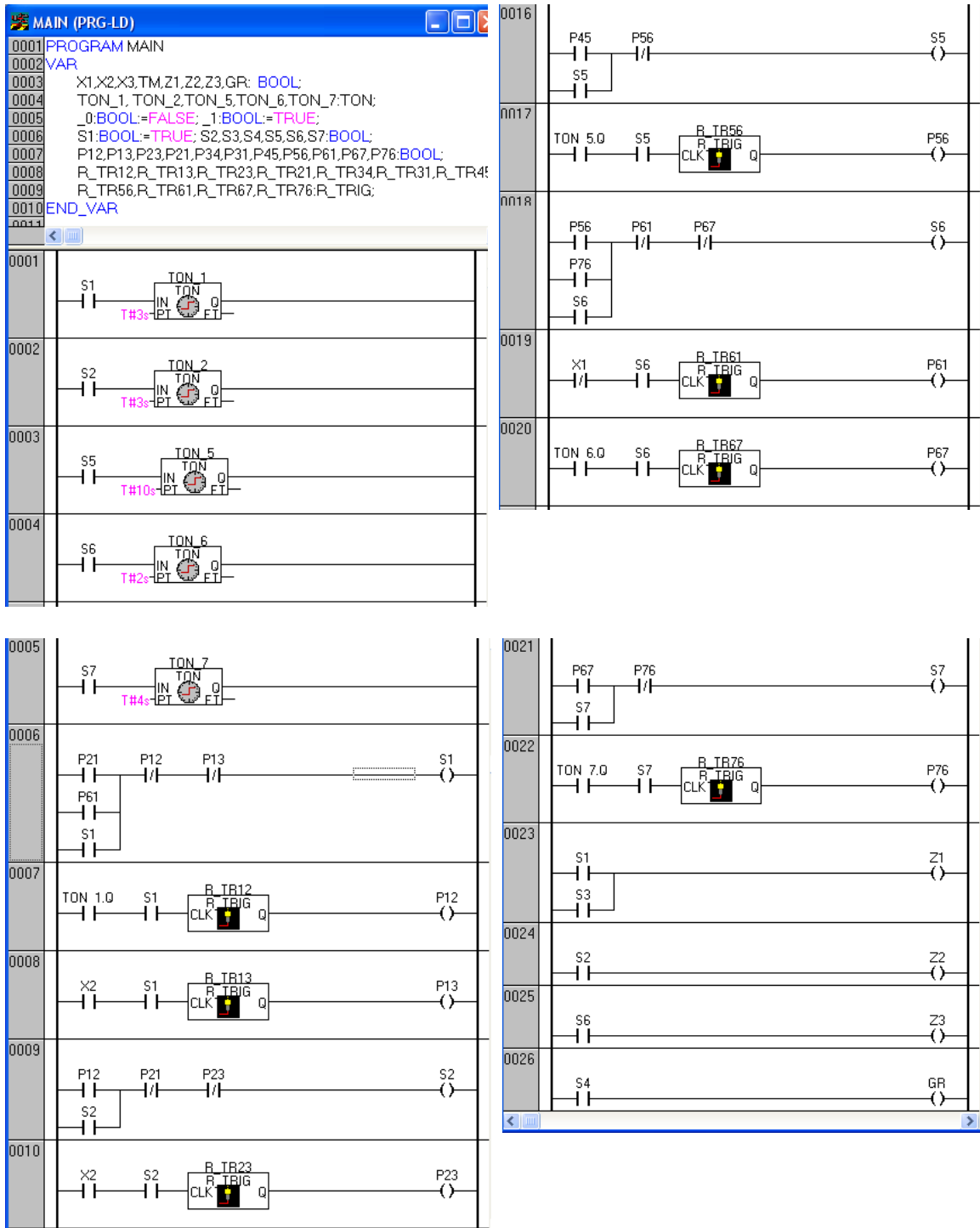
- Stany

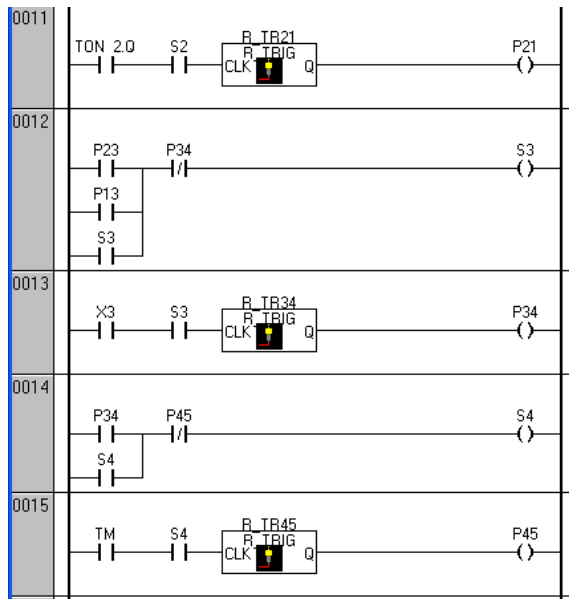
- ① nalewanie z Z1 przez 3 s, $h < X2$
- ② nalewanie z Z2 przez 3 s, $h < X2$
- ③ nalewanie z Z1 do X3
- ④ grzanie do TM
- ⑤ oczekiwanie przez 10 s
- ⑥ opróżnianie, Z3 otwarty przez 4 s.
- ⑦ Z3 zamknięty przez 2 s.

- Graf



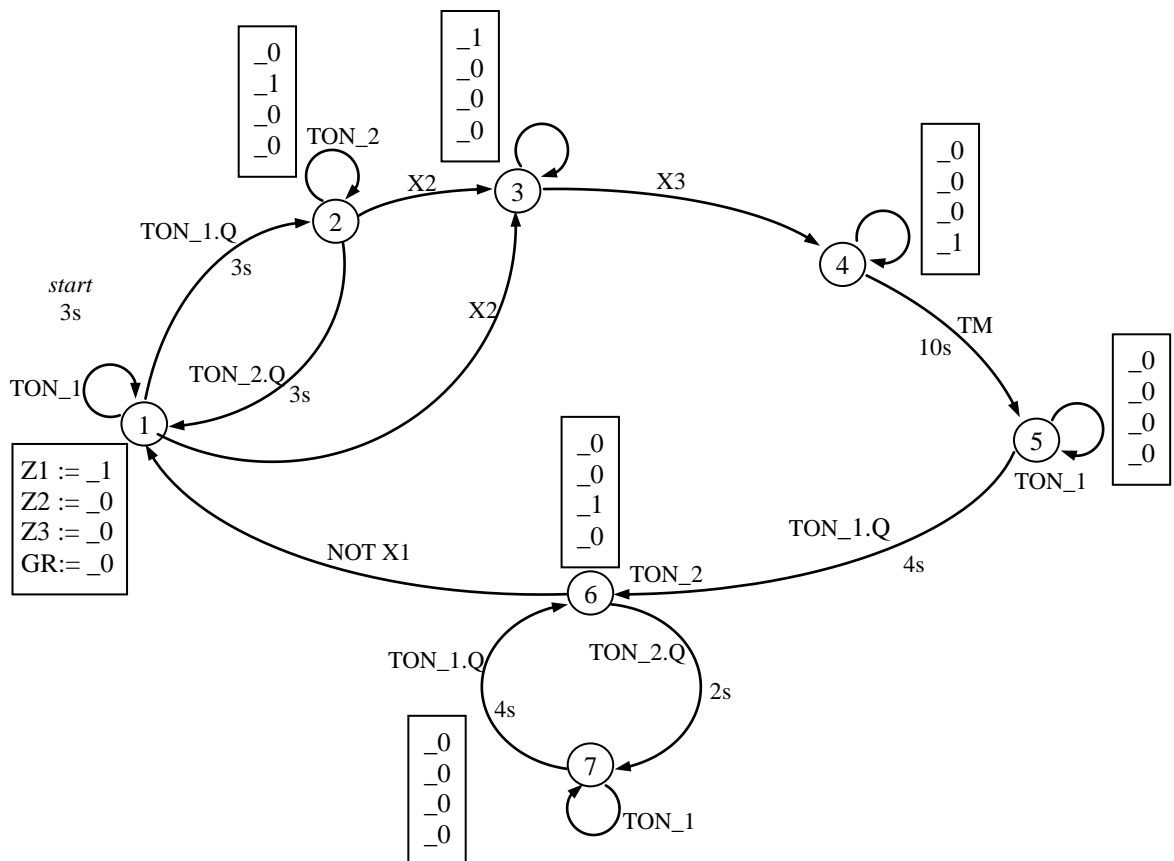
3. Kod LD





4. Kod ST

- W typowej realizacji wykorzystywane byłyby pięć czasomierzy, po jednym dla każdego przedziału czasu. Można jednak ograniczyć się do dwóch czasomierzy ustawiając odmienne wartości na wejściach PT odpowiednio do odmierzanych przedziałów w konkretnych stanach.
- Zmodyfikowany graf



```
MAIN (PRG-ST)
0001 PROGRAM MAIN
0002 VAR
0003   X1,X2,X3,TM,Z1,Z2,Z3,GR: BOOL; stan: INT:=1;
0004   TON_1:TON; TON_2:TON;
0005   T1: TIME := T#3s; T2: TIME := T#3s;
0006   _0:BOOL:=FALSE; _1:BOOL:=TRUE;
0007 END_VAR
0001 TON_1(IN:=stan=1 OR stan=5 OR stan=7, PT:=T1);
0002 TON_2(IN:=stan=2 OR stan=6, PT:=T2);
0003
0004 CASE stan OF
0005 1: Z1:=_1; Z2:=_0; Z3:=_0; GR:=_0;
0006   IF TON_1.Q THEN T2:=T#3s; stan:=2;
0007   ELSIF X2 THEN stan:=3; END_IF
0008 2: Z1:=_0; Z2:=_1; Z3:=_0; GR:=_0;
0009   IF TON_2.Q THEN T1:=T#3s; stan:=1;
0010   ELSIF X2 THEN stan:=3; END_IF
0011 3: Z1:=_1; Z2:=_0; Z3:=_0; GR:=_0;
0012   IF X3 THEN stan:=4; END_IF
0013 4: Z1:=_0; Z2:=_0; Z3:=_0; GR:=_1;
0014   IF TM THEN T1:=T#10s; stan:=5; END_IF
0015 5: Z1:=_0; Z2:=_0; Z3:=_0; GR:=_0;
0016   IF TON_1.Q THEN T2:=T#4s; stan:=6; END_IF
0017 6: Z1:=_0; Z2:=_0; Z3:=_1; GR:=_0;
0018   IF TON_2.Q THEN T1:=T#2s; stan:=7;
0019   ELSIF NOT X1 THEN T1:=T#3s; T2:=T#3s; stan:=1; END_IF
0020 7: Z1:=_0; Z2:=_0; Z3:=_0; GR:=_0;
0021   IF TON_1.Q THEN T2:=T#4s; stan:=6; END_IF
0022 END_CASE
```

Przed uruchamianiem złożonych programów w pakiecie TwinCAT zaleca się stosować *Reset*.