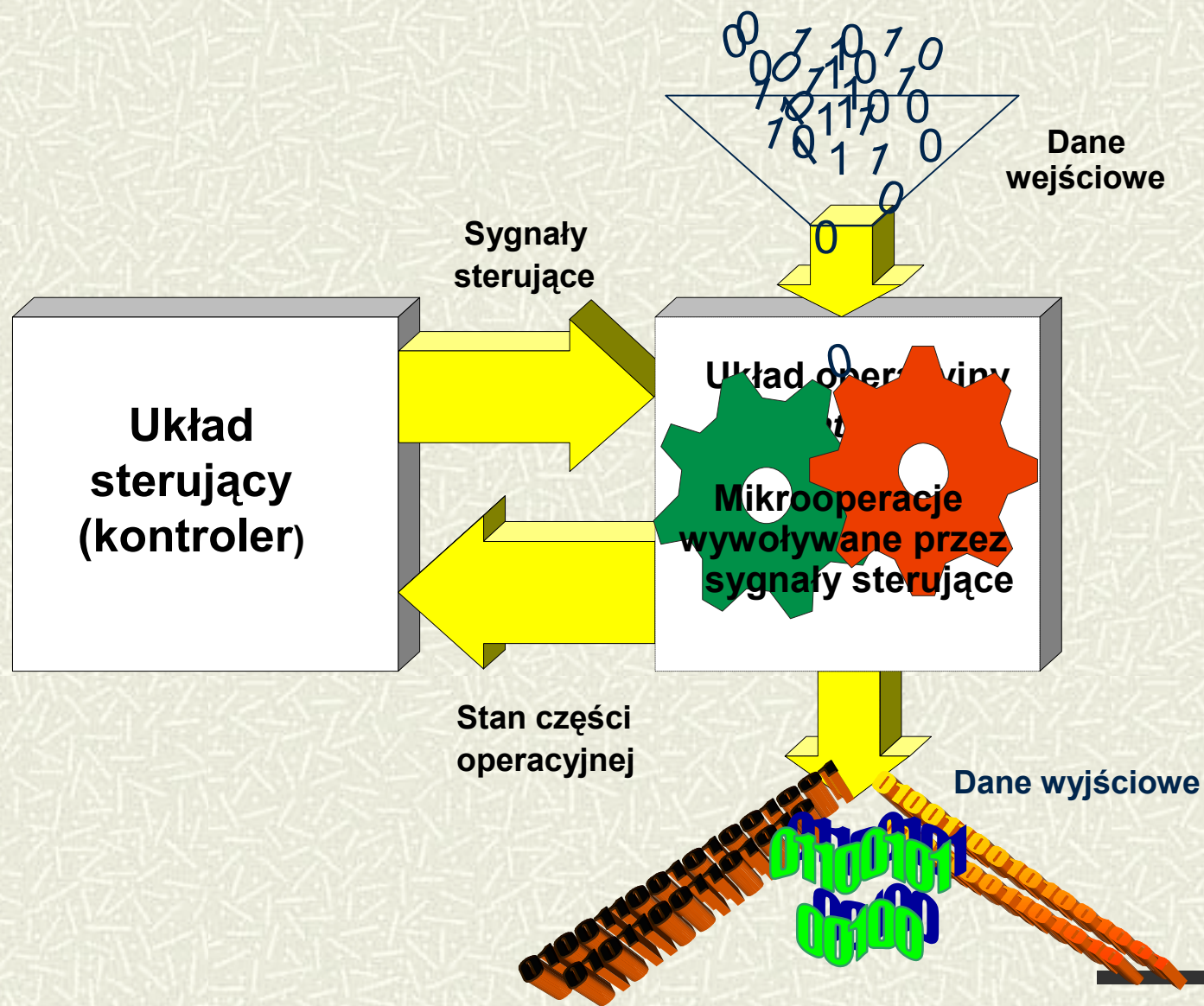
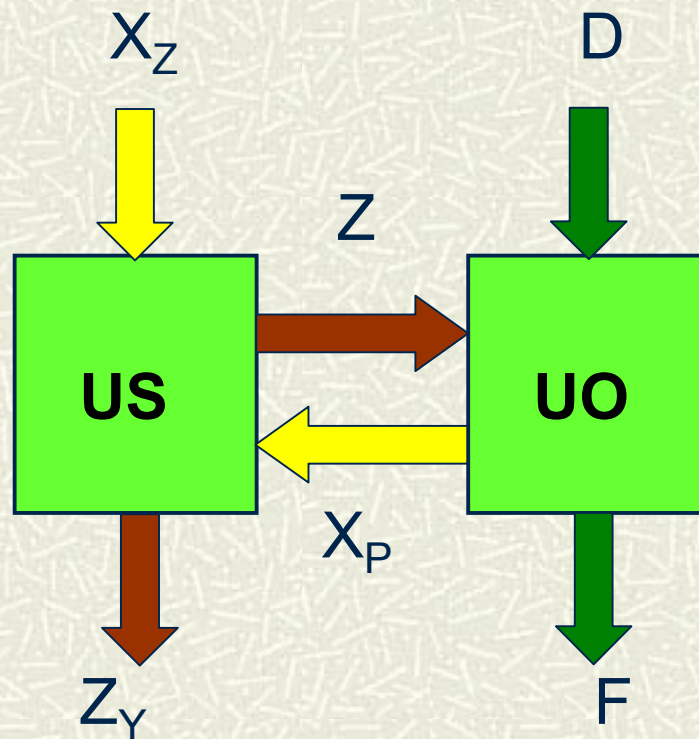


System cyfrowy



System cyfrowy - realizacja



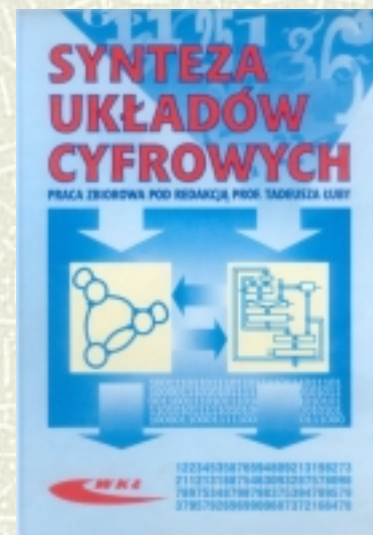
UO – z bloków funkcjonalnych

**US – automat lub (coraz rzadziej stosowany)
układ mikroprogramowany**

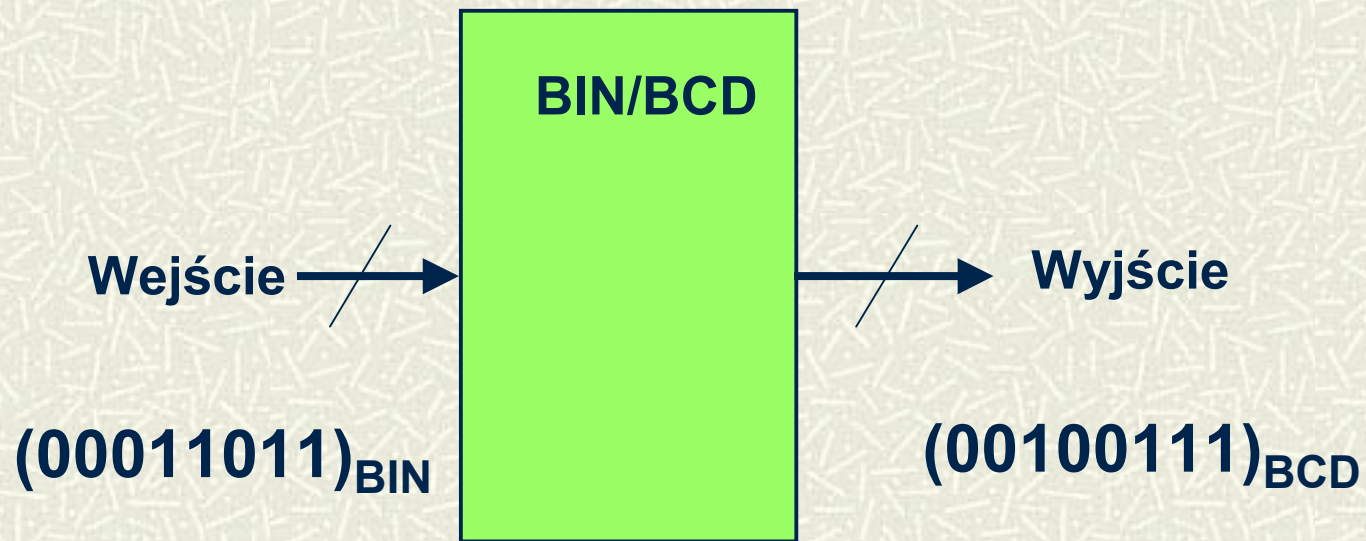
Konwerter kodu binarnego na kod BCD¹⁾:

- | | | |
|------|------|------|
| 4 | 8 | 9 |
| 0100 | 1000 | 1001 |

1) Przykład ten dokładnie omówiony jest w książce:



Konwerter Bin2BCD

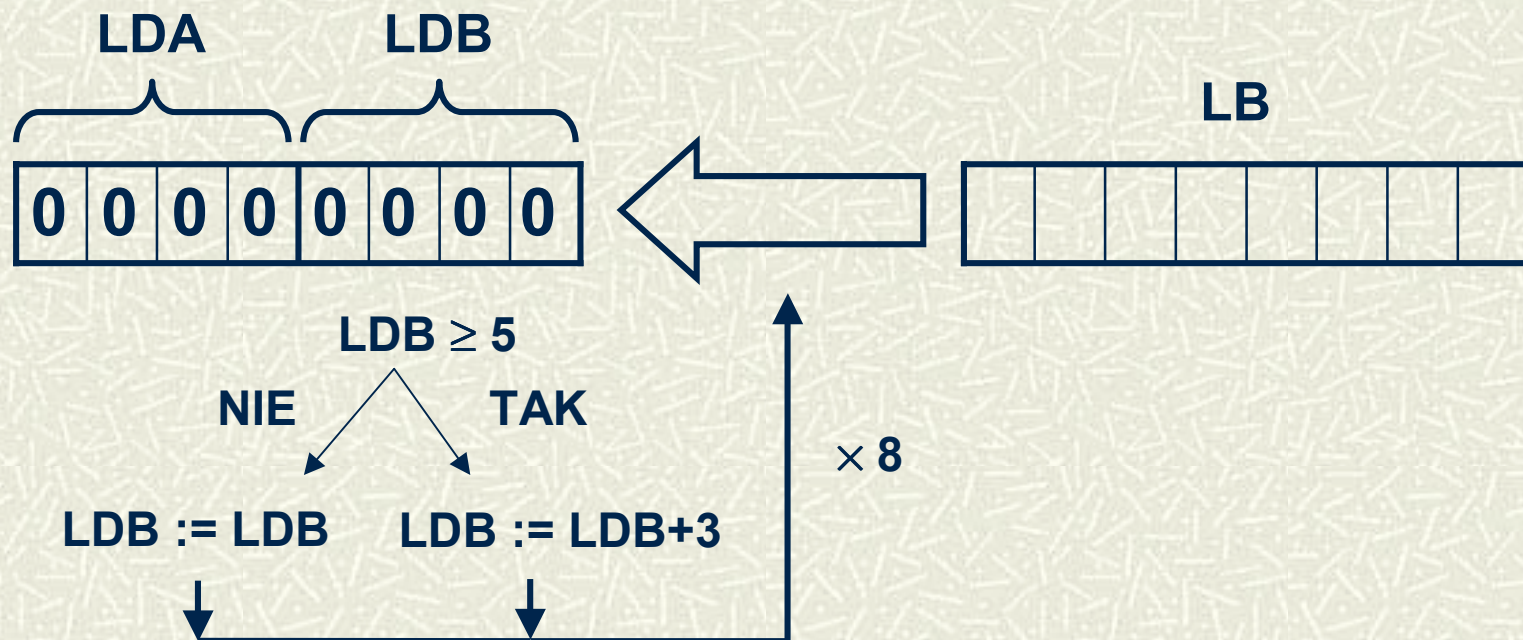


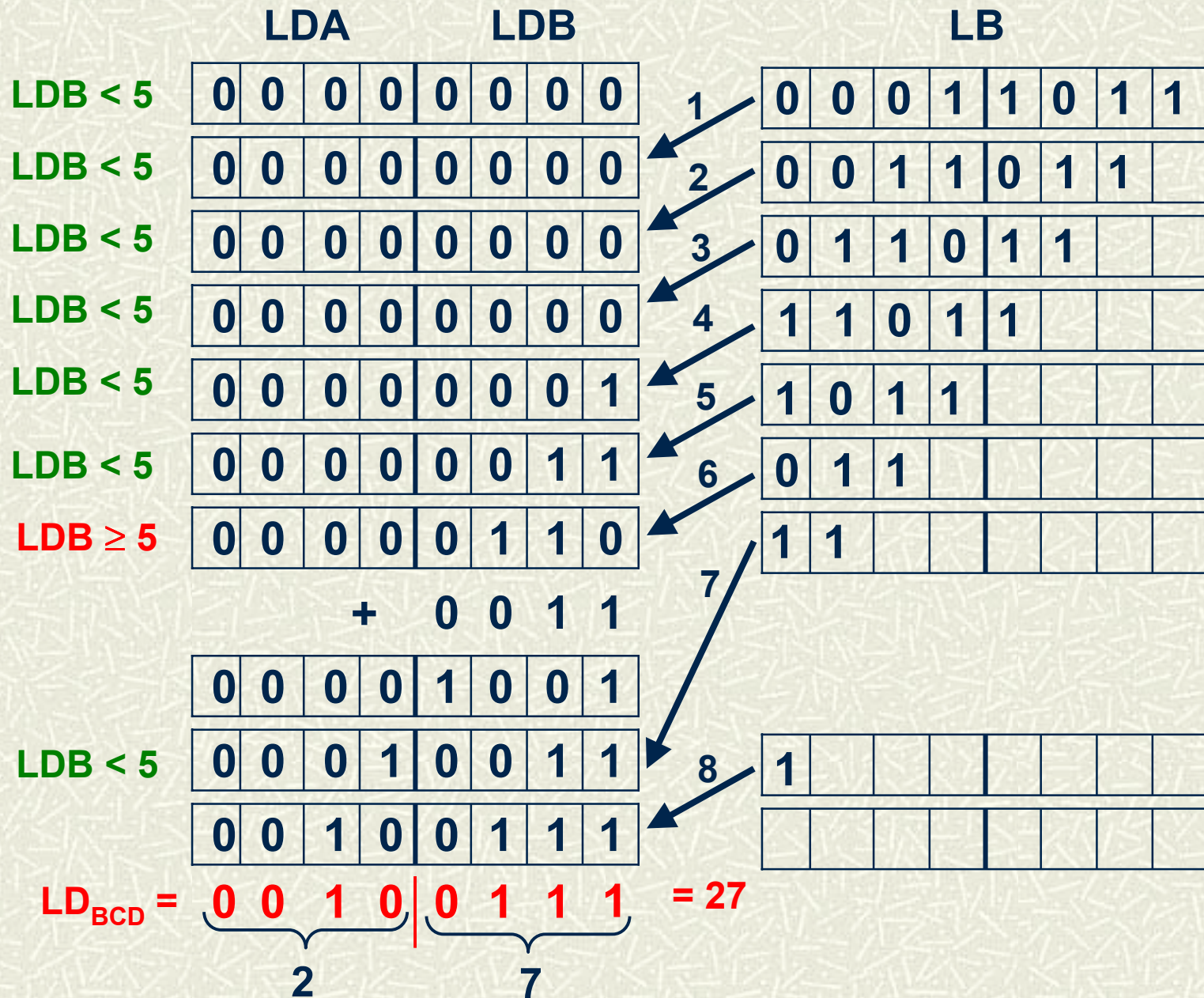
$0 \leq \text{liczby} \leq 99$

Metoda +3

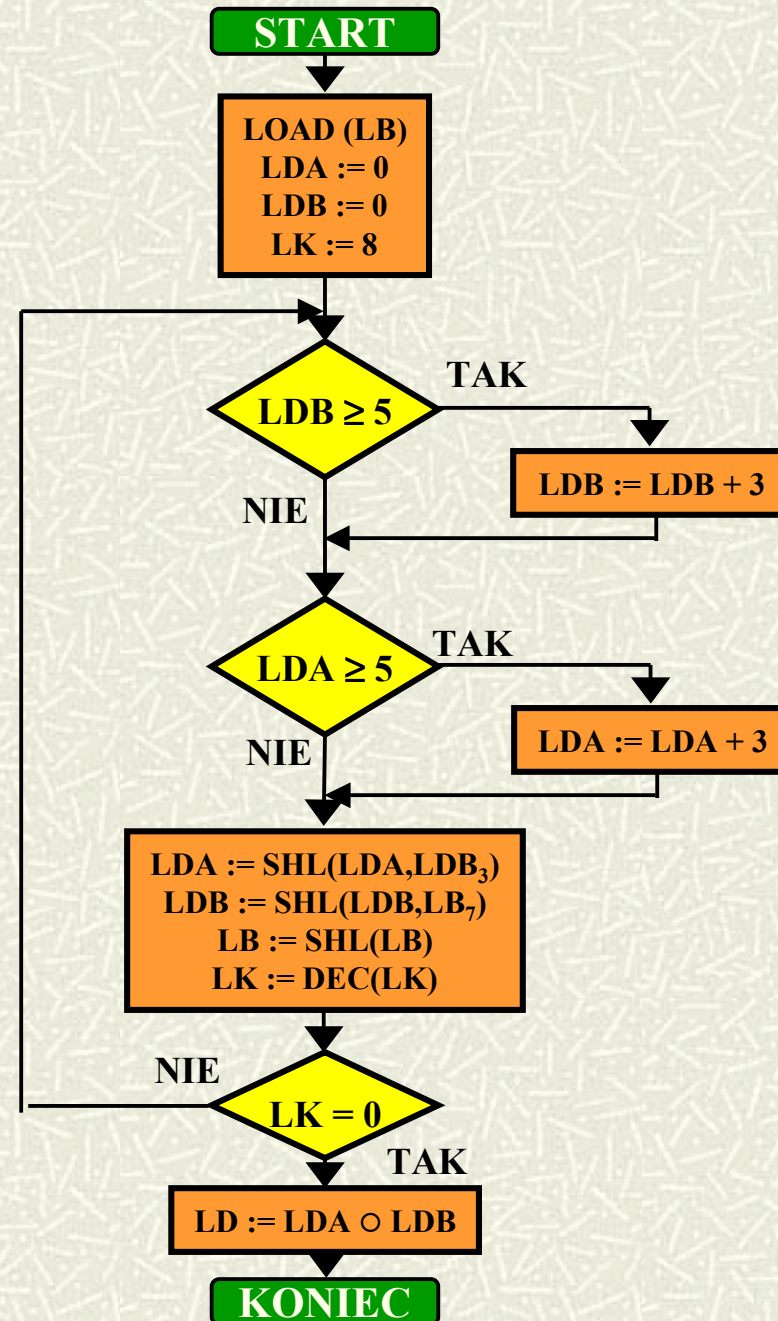
0 0 0 0 0 0 0 0

27 = 0 0 0 1 1 0 1 1

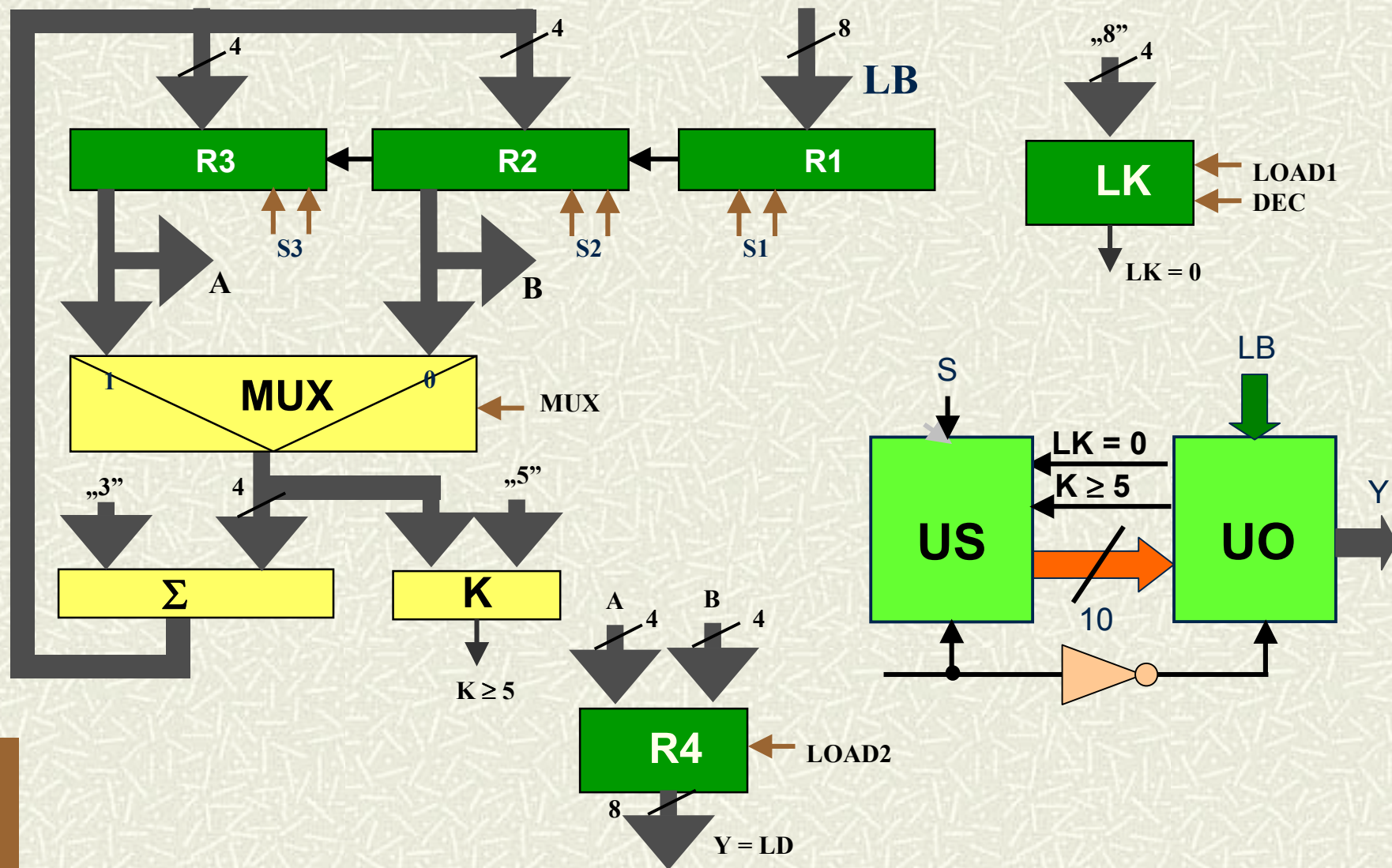




Algorytm konwersji z kodu BIN na BCD



Opis strukturalny



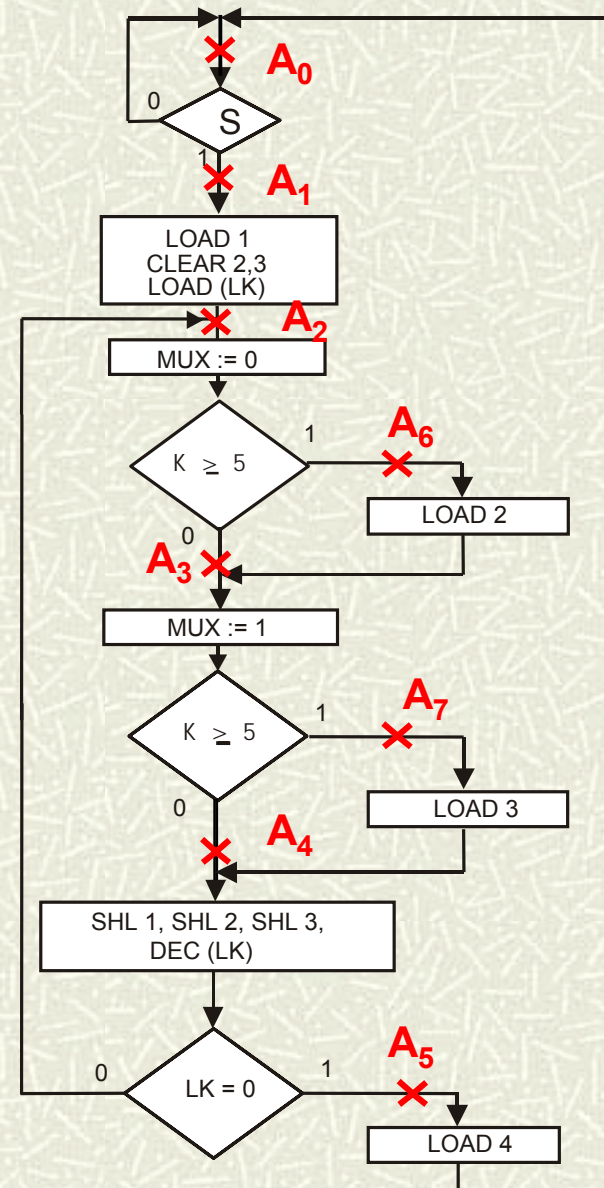
I
T
P
W

ZPT

Sieć działań z uwzględnieniem UO

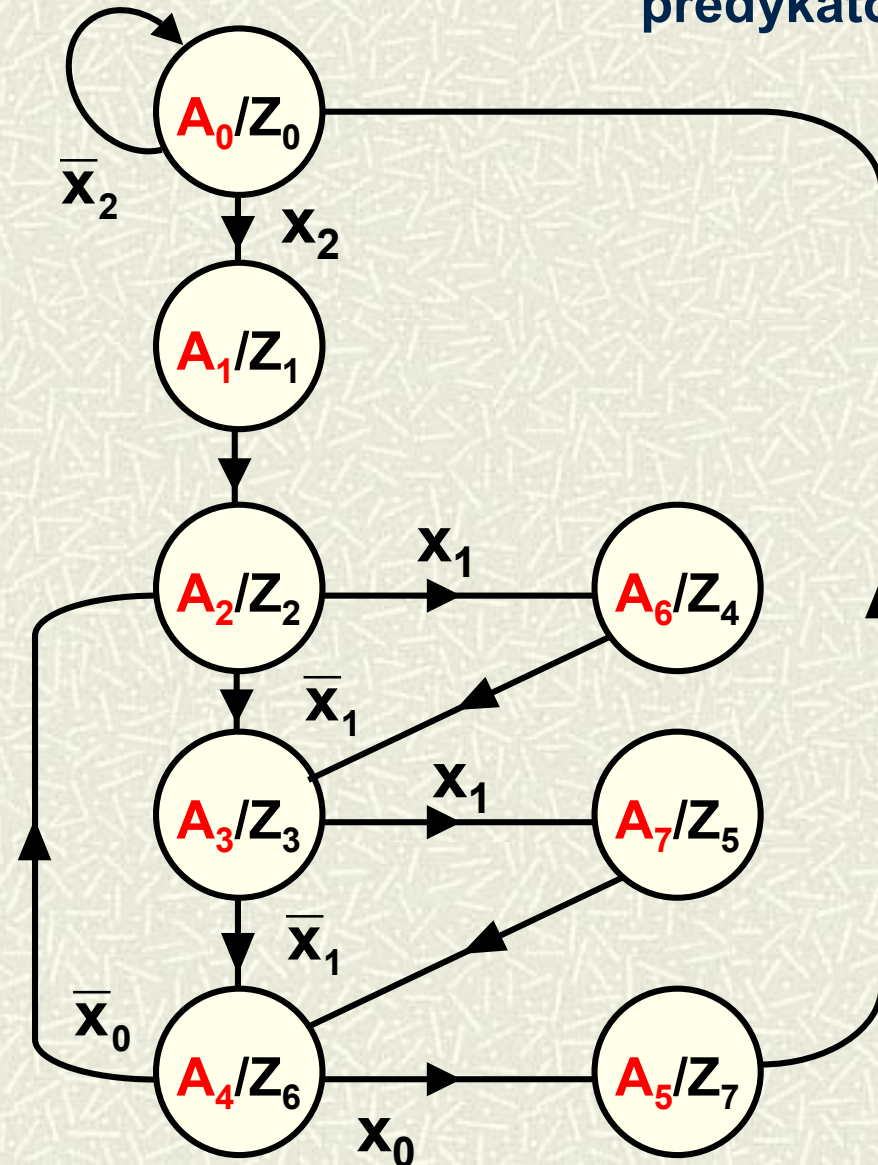
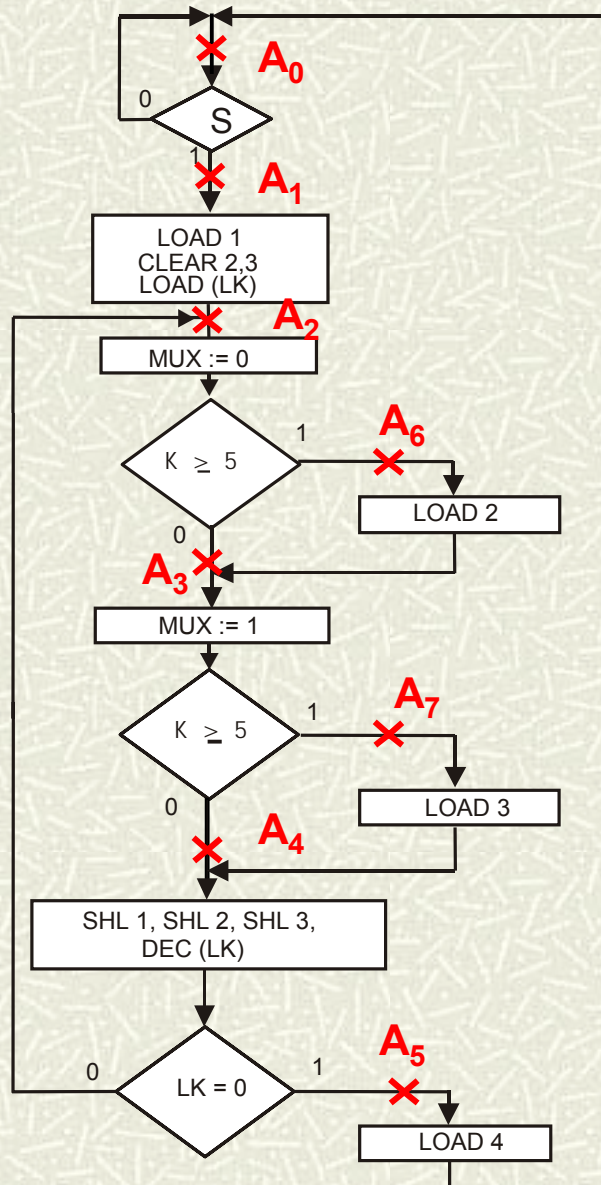
Synteza układu
sterującego!

Numeracja stanów
wewnętrznych

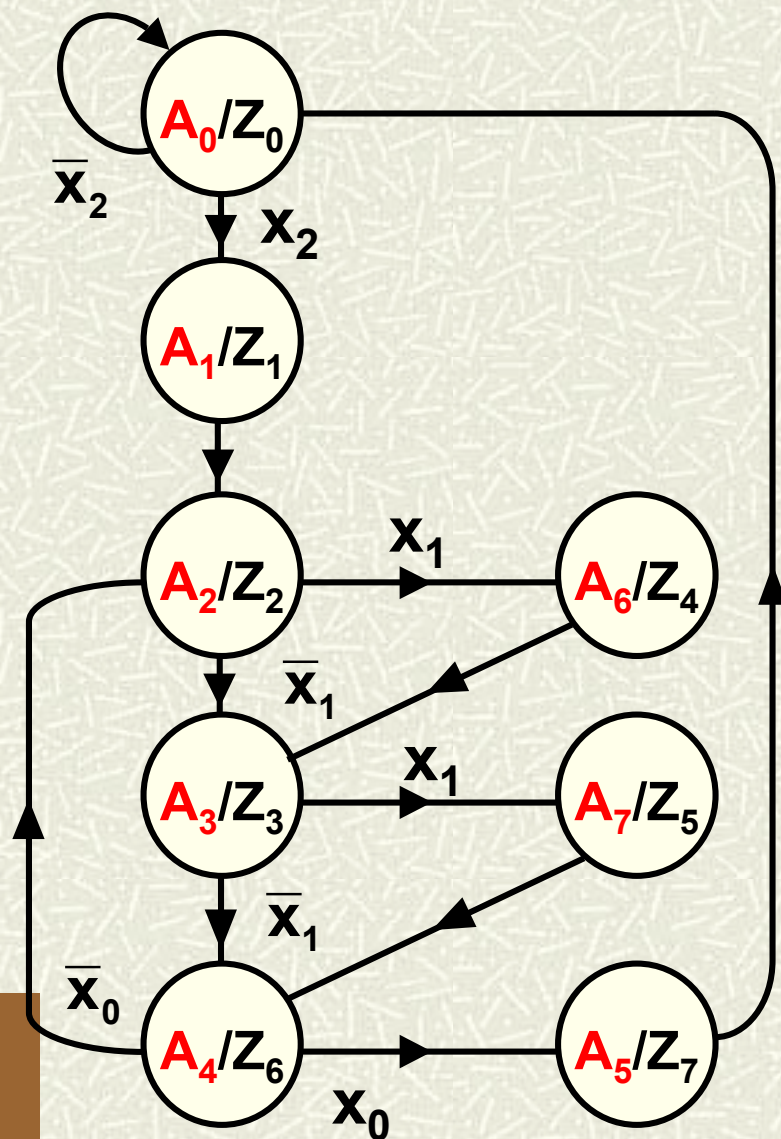


Zamiana SD na automat sterujący

Nowe oznaczenia sygnałów
predykatowych



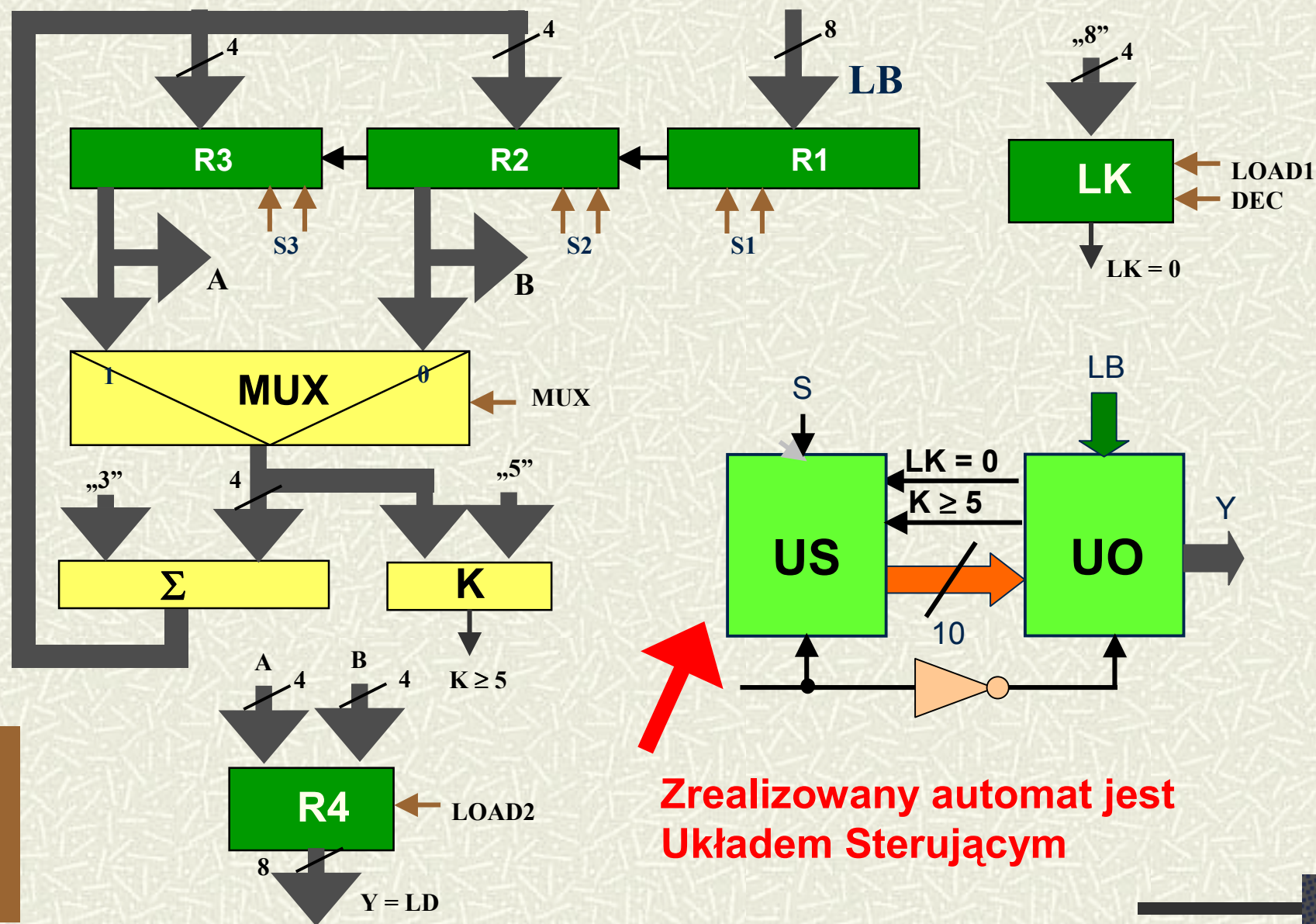
Tablica p-w automatu sterującego



$\begin{matrix} x_2 x_1 x_0 \\ S \end{matrix}$	000	001	011	010	110	111	101	100	Z
A ₀	A ₀	A ₀	A ₀	A ₀	A ₁	A ₁	A ₁	A ₁	Z ₀
A ₁	A ₂	A ₂	A ₂	A ₂	A ₂	A ₂	A ₂	A ₂	Z ₁
A ₂	A ₃	A ₃	A ₆	A ₆	A ₆	A ₆	A ₃	A ₃	Z ₂
A ₃	A ₄	A ₄	A ₇	A ₇	A ₇	A ₇	A ₄	A ₄	Z ₃
A ₄	A ₂	A ₅	A ₅	A ₂	A ₂	A ₅	A ₅	A ₂	Z ₆
A ₅	A ₀	A ₀	A ₀	A ₀	A ₀	A ₀	A ₀	A ₀	Z ₇
A ₆	A ₃	A ₃	A ₃	A ₃	A ₃	A ₃	A ₃	A ₃	Z ₄
A ₇	A ₄	A ₄	A ₄	A ₄	A ₄	A ₄	A ₄	A ₄	Z ₅

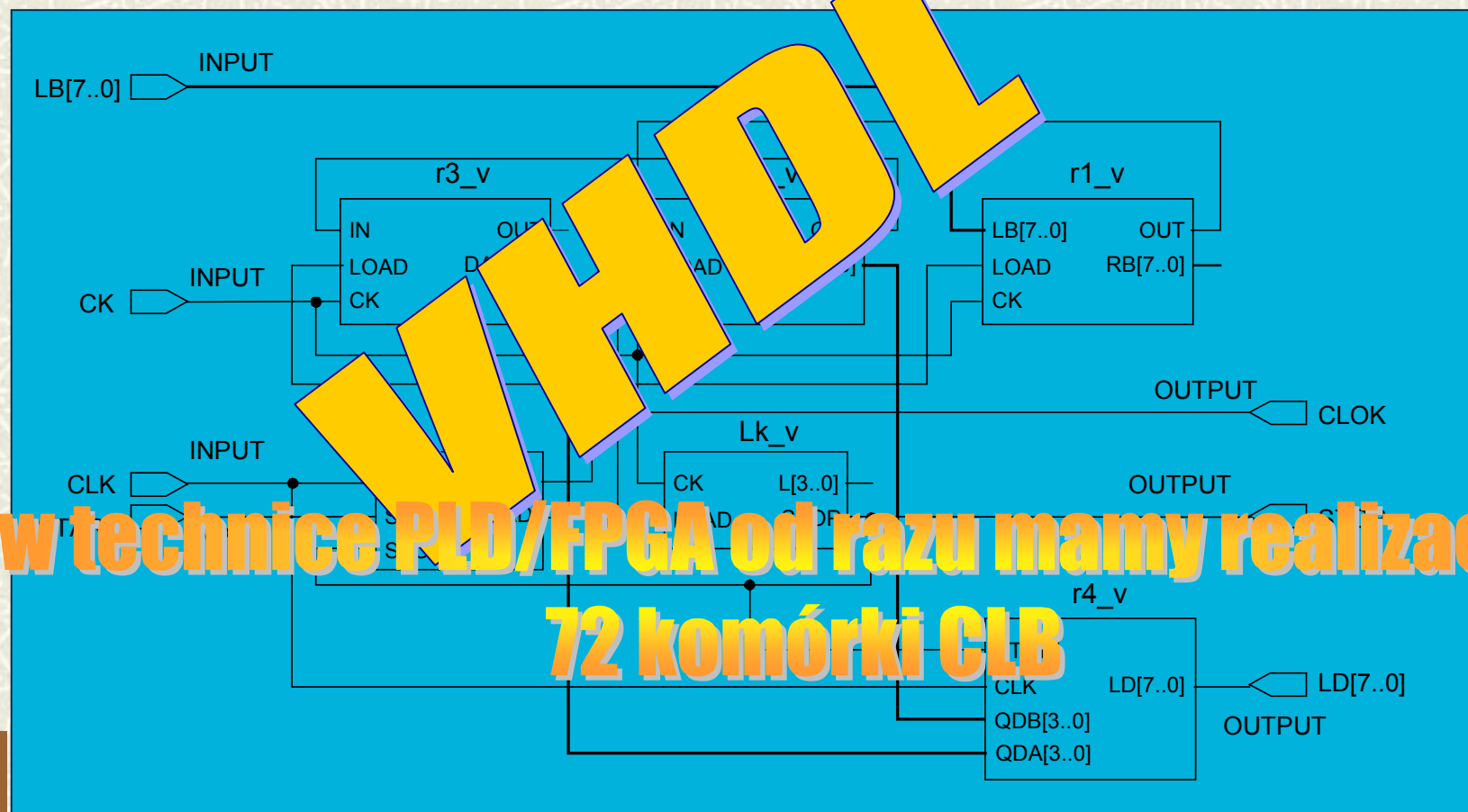
Tablica p-w wystarczy do realizacji automatu!

Opis strukturalny



a jak jest w rzeczywistości...

uzyskaną strukturę zapisujemy w języku opisu sprzętu i kompilujemy



w technice PLD/FPGA od razu mamy realizację:
72 komórki CLB

Realizacja wg języka VHDL (raport systemu MAX+PLUSII)

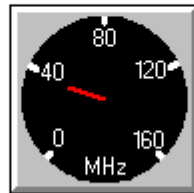
** DEVICE SUMMARY **

Chip/ POF	Device	Input Pins	Output Pins	Bidir Pins	Memory Bits	Memory % Utilized	LCs	% Utilized
bin2bcd	EPF10K20RC240-3	10	9	0	0	0 %	72	6 %
User Pins:		10	9	0				

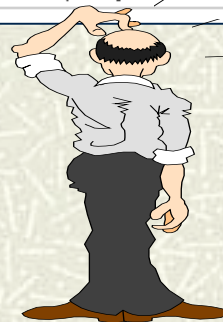
Registered Performance

Clock: zegar (10 paths)

Source: |us:k2|aut1.Q
Destination: |us:k2|aut0.Q



Clock period: 24.1ns
Frequency: 41.49MHz



Można inaczej: Bin2BCD – specyfikacja behawioralna w języku AHDL

```
SUBDESIGN bin2bcd
(
    lb[7..0], start, zegar      : INPUT;
    ld[7..0], koniec           : OUTPUT;
)
VARIABLE
    lda[3..0], ldb[3..0]      : DIF;
    lb_r[7..0], lk[3..0]      : DIF;
    ld[7..0], koniec          : DIF;
BEGIN
    (lda[], ldb[], lb_r[], lk[]) clk = 0;
    (ld[], koniec).clk = 0;
    IF start THEN
        lb_r[] = lb[];
        lk[] = 0;
    ELSE
        IF lk[] > 0 THEN
            IF ldb[] >= 5 THEN
                lda[] = (lda[2..0], B"1"); -- bit lb[3] = 1
                ldb[] = (ldb[2..0] + 3, lb_r[7]);
            ELSE
                lda[] = (lda[2..0], B"0"); -- bit lb[3] = 0
                ldb[] = (ldb[2..0], lb_r[7]);
            END IF;
            lb_r[] = (lb_r[6..0], B"0"); -- przesun w lewo
            lk[] = lk[] - 1; -- zmniejsz lk
        ELSE
            lda[] = lda[];
            ldb[] = ldb[];
            ld[] = (lda[], ldb[]);
            koniec = B"1";
        END IF;
    END IF;
END;
```

Opis behawioralny (raport systemu)

** DEVICE SUMMARY **

Chip/ POF	Device	Input Pins	Output Pins	Bidir Pins	Memory Bits	Memory % Utilized	Memory LCs	LCs % Utilized
bin2bcd	EPF10K20RC240-3	10	9	0	0	0 %	41	3 %
User Pins:		10	9	0				

Registered Performance

Clock: zegar (10 paths)

Source: lk2.Q

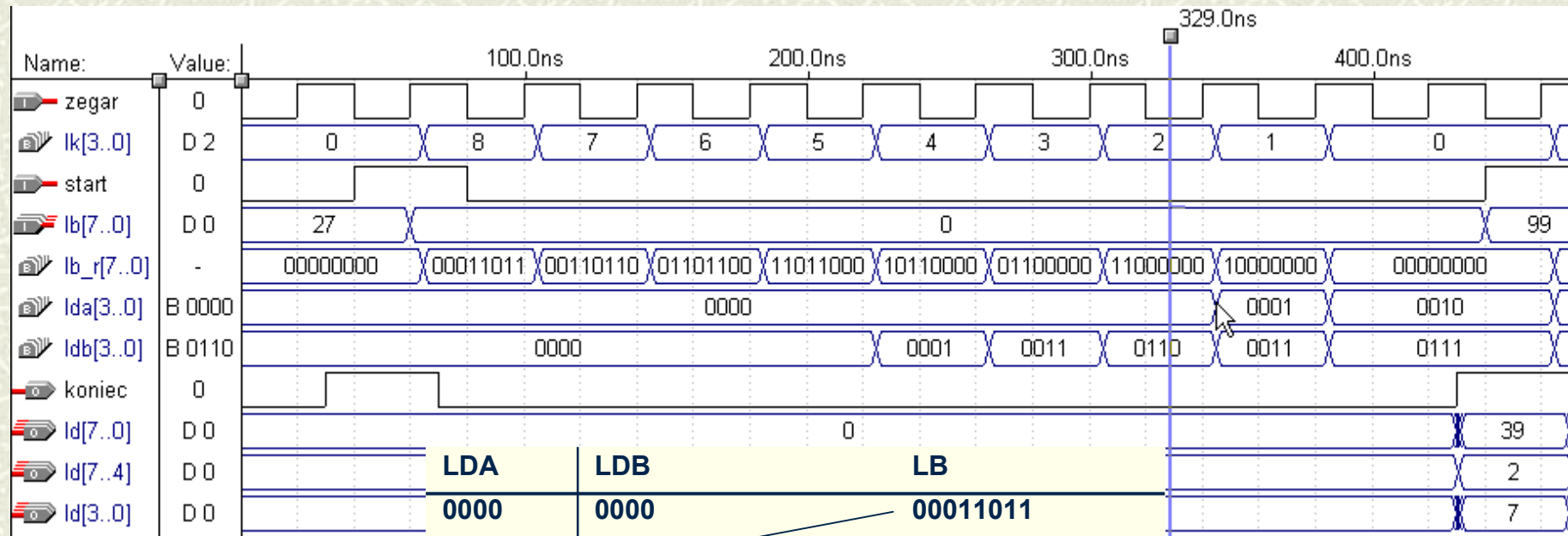
Destination: ldb2.Q



Clock period: 15.9ns
Frequency: 62.89MHz

Wynik otrzymujemy po 8 taktach zegara więc jedna liczba konwertowana jest w około **127 ns**, czyli szybkość konwersji wynosi **7,8 mln** liczb na sekundę

Opis behawioralny (symulacja)



LDA	LDB	LB
0000	0000	00011011
0000	0000	00110110
0000	0000	01101100
0000	0000	11011000
0000	0001	10110000
0000	0011	01100000
0000	0110	11000000
+	0011	
0000	1001	
0001	0011	10000000
0010	0111	00000000

Opis behawioralny...



Spisuje się doskonale, ale czy rzeczywiście jest to zawsze najlepsza metoda syntezy. Są podobno doskonałe procedury syntezy logicznej, np. Espresso

Konwerter Bin2BCD na poziomie logicznym

```
.type fr
```

```
.i 9
```

```
.o 8
```

```
.p 100
```

```
000000000 0000 0000
```

```
000000001 0000 0001
```

```
000000010 0000 0010
```

```
000000011 0000 0011
```

```
000000100 0000 0100
```

```
000000101 0000 0101
```

```
000000110 0000 0110
```

```
000000111 0000 0111
```

```
000001000 0001 0000
```

```
000001001 0001 0001
```

```
000001010 0001 0010
```

```
000001011 0001 0011
```

```
000001100 0001 0100
```

```
000001101 0001 0101
```

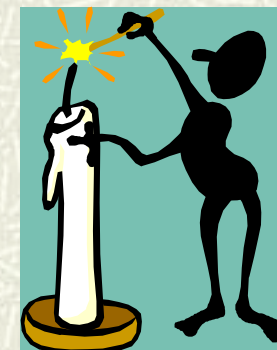
```
* * *
```

```
001100001 1001 0111
```

```
001100010 1001 1000
```

```
001100011 1001 1001
```

```
.e
```



MAX+PLUSII → ??? komórek

Tablica prawdy – bin2bcd

```

SUBDESIGN binbcd99
(
  i[8..0] : INPUT;
  o[7..0] : OUTPUT;

)

BEGIN

TABLE
  (i[8..0]) => (o[7..0]);
  B"00000000" => B"00000000";
  B"00000001" => B"00000001";
  B"00000010" => B"00000010";
  B"00000011" => B"00000011";
  B"00000100" => B"00000100";
  B"00000101" => B"00000101";
  B"00000110" => B"00000110";
  B"00000111" => B"00000111";
  B"00001000" => B"00001000";
  B"00001001" => B"00001001";
  B"00001010" => B"00010000";
  B"00001011" => B"00010001";
  B"00001100" => B"00010010";
  B"00001101" => B"00010011";

```



Rozwiązanie bardzo szybkie...
gdyby nie ta ogromna ilość zasobów
potrzebnych do realizacji

I
T
P
W

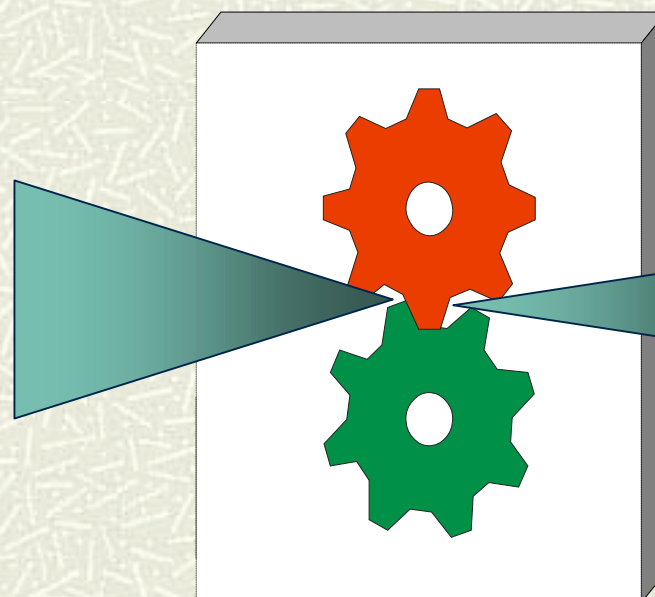
Chip/ POF	Device	Input Pins	Output Pins	Bidir Pins	Memory Bits	Memory % Utilized	Memory %	LCs	LCs % Utilized
binbcd99	EP1K10TC100-1	7	8	0	0	0	%	131	28 %
User Pins:		7	8	0					

**A jak sobie poradzą z tym konwerterem
najnowsze algorytmy dekompozycji**



Tablica prawdy – bin2bcd

```
# Konwerter
bin2bcd
.type fr
.i 7
.o 8
.p 100
0000000 00000000
0000001 00000001
0000010 00000010
0000011 00000011
0000100 00000100
0000101 00000101
. . . . .
1011111 10010101
1100000 10010110
1100001 10010111
1100010 10011000
1100011 10011001
.e
```



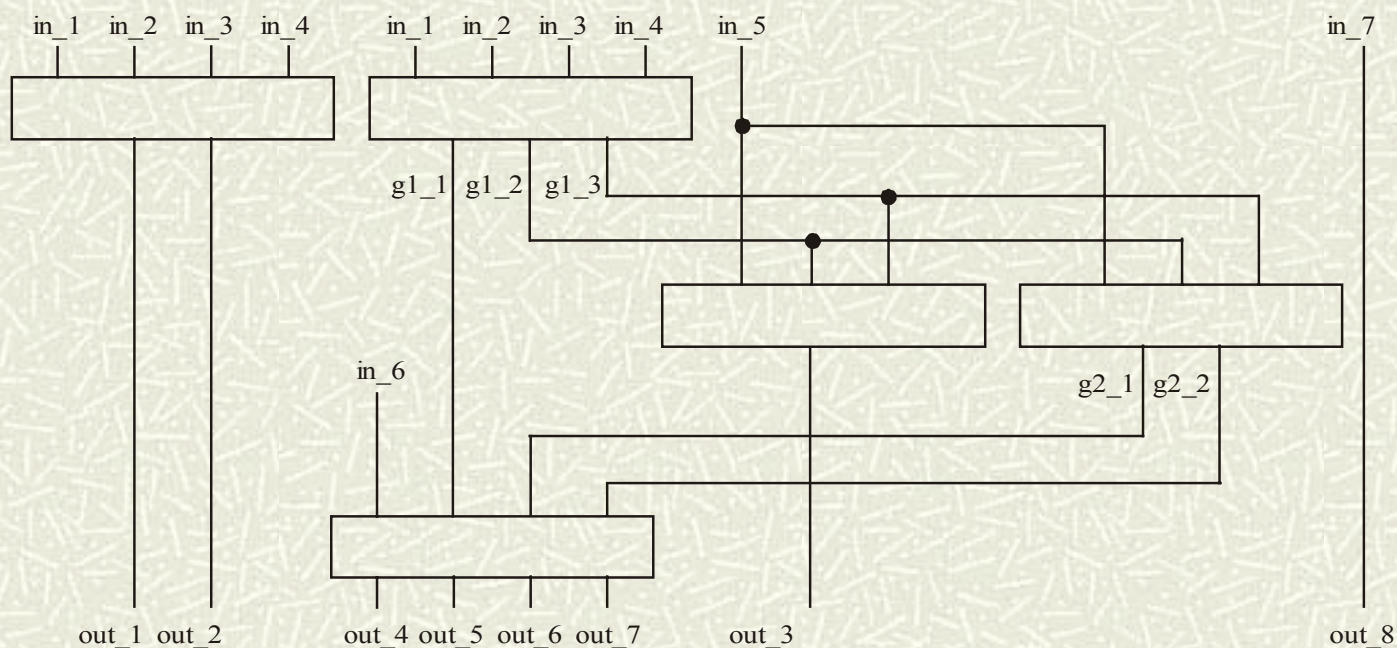
```
TITLE " Decomposed project: bin2bcd ";
% Translated from DEMAIn format %
% Warsaw University of Technology %
% Institute of Telecommunications %
SUBDESIGN A
(
    in_1, in_2, in_3, in_4
        :INPUT;
    in_5, in_6, in_7
        :INPUT;
    out_1, out_2, out_3, out_4 :OUTPUT;
    out_5, out_6, out_7, out_8 :OUTPUT;
)
VARIABLE
    g1_1, g1_2, g1_3, g3_1      :LCELL;
    g2_1, g2_2                  :LCELL;
BEGIN
---- Level 1 ----
TABLE
    (in_1, in_2, in_3, in_4) => (g1_1);
    B"1000" => B"0";
    B"0011" => B"0";
    .....
    B"0001" => B"1";
    B"1011" => B"1";
END TABLE;
TABLE
    (in_6, g1_1, g2_1, g2_2) => (out_7);
    B"0X10" => B"0";
    .....
    B"1011" => B"1";
END TABLE;
END;
```

Demain

+

ans2hdl

Realizacja Bin2bcd wg Demaina



Bin2bcd wg Demaina i MAX+PLUSII

SUBDESIGN BINBCD99

```
(
  in_1, in_2, in_3, in_4      :INPUT;
  in_5, in_6, in_7           :INPUT;
  out_1, out_2, out_3, out_4  :OUTPUT;
  out_5, out_6, out_7, out_8  :OUTPUT;
)
```

VARIABLE

```
g1_1, g1_2, g1_3, g2_1      :LCELL;
g2_2                        :LCELL;
```

BEGIN

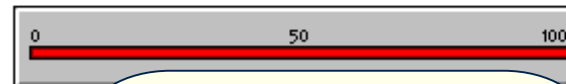
----- Level 1 -----

TABLE

```
(in_1, in_2, in_3, in_4) => (g1_1);
B"1000" => B"0";
B"0011" => B"0";
B"0100" => B"1";
B"1001" => B"1";
B"0010" => B"0";
B"1100" => B"0";
B"0111" => B"0";
```

Delay Matrix

	Destination				
	out_3	out_4	out_5	out_6	out_7
6ns	8.4ns/8.6ns	9.0ns/10.3ns	9.8ns/10.2ns	8.8ns/10.3ns	8.8ns/10.3ns
7ns	8.4ns	9.2ns/10.3ns	9.6ns/10.2ns	9.0ns/10.3ns	9.0ns/10.3ns
4ns	8.2ns/8.7ns	9.2ns/10.4ns	9.8ns/10.3ns	9.0ns/10.4ns	9.0ns/10.4ns
6ns	8.5ns/8.6ns	9.3ns/10.4ns	9.8ns/10.3ns	9.1ns/10.4ns	9.1ns/10.4ns
	9.5ns	11.1ns/11.3ns	11.0ns/11.2ns	11.3ns	
		7.7ns	7.5ns	7.7ns	



Rozwiązanie jeszcze
lepsze i mało komórek

I
T
P
W

Chip/		Input Pins	Output Pins	Bidir Pins	Memory Bits	Memory % Utilized	LCs	LCs % Utilized
POF	Device							
binbcd99	EP1K10TC100-1	7	8	0	0	0 %	13	2 %
User Pins:		7	8	0				

Konwerter Bin2BCD na poziomie logicznym

.type fr

.i 9

.o 8

.p 100

000000000 0000 0000

000000001 0000 0001

000000010 0000 0010

000000011 0000 0011

000000100 0000 0100

000000101 0000 0101

000000110 0000 0110

000000111 0000 0111

000001000 0001 0000

000001001 0001 0001

000001010 0001 0010

000001011 0001 0011

000001100 0001 0100

000001101 0001 0101

001100001 1001 0111

001100010 1001 1000

001100011 1001 1001

.e



MAX+PLUSII → 131 komórek



DEMAIN → 13 komórek (!!!)

