Design Procedure

1. Specification
   - Write a specification for the circuit if one is not already available
2. Formulation
   - Derive a truth table or initial Boolean equations that define the required relationships between the inputs and outputs, if not in the specification
3. Optimization
   - Apply 2-level and multiple-level optimization
   - Draw a logic diagram or provide a netlist for the resulting circuit using ANDs, ORs, and inverters
4. Technology Mapping
   - Map the logic diagram or netlist to the implementation technology selected
5. Verification
   - Verify the correctness of the final design

Design Example

1. Specification
   - BCD to Excess-3 code converter
   - Transforms BCD code for the decimal digits to Excess-3 code for the decimal digits
   - BCD code words for digits 0 through 9: 4-bit patterns 0000 to 1001, respectively
   - Excess-3 code words for digits 0 through 9: 4-bit patterns consisting of 3 (binary 0011) added to each BCD code word
   - Implementation:
     - multiple-level circuit
     - NAND gates (including inverters)

Design Example **(continued)**

1. Formulation
   - Conversion of 4-bit codes can be most easily formulated by a truth table
   - Variables
     - BCD: A,B,C,D
   - Variables
     - Excess-3 W,X,Y,Z
   - Don't Cares
     - BCD 1010 to 1111

Input BCD A B C D	**Output Excess-3 WXYZ**
0 0 0 0	**0 0 1 1**
0 0 0 1	**0 1 0 0**
0 0 1 0	**0 1 0 1**
0 0 1 1	**0 1 1 0**
0 1 0 0	**0 1 1 1**
0 1 0 1	**1 0 0 0**
0 1 1 0	**1 0 0 1**
0 1 1 1	**1 0 1 0**
1 0 0 0	**1 0 1 1**
1 0 0 1	**1 0 1 1**

Design Example **(continued)**

1. Optimization
   a. 2-level using K-maps

$W = A + BC + BD$

$X = B'C + B'D + BC'D'$

$Y = CD + C'D'$

$Z = D'$

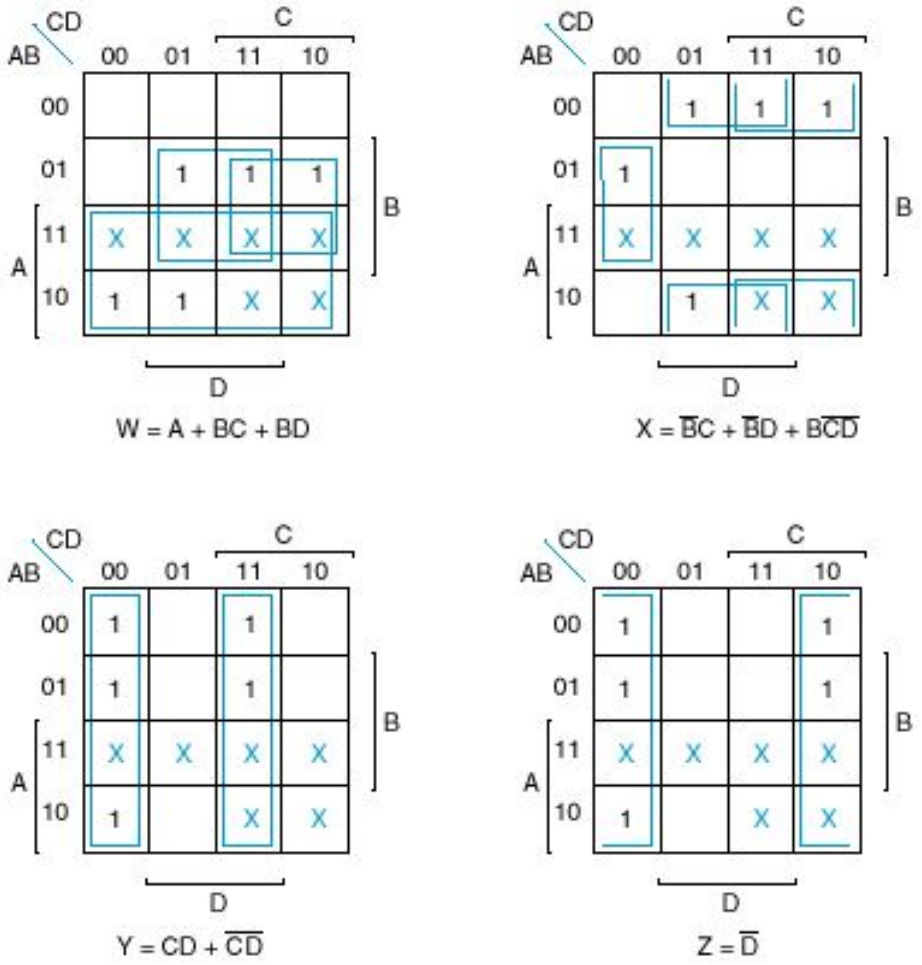

Fig. 3-10 Maps for BCD-to-Excess-3 Code Converter

Design Example **(continued)**

1. Optimization (continued)
   a. Multiple-level using transformations

$W = A + BC + BD$
$X = B'C + B'D + BC'D'$
$Y = CD + C'D'$
$Z = D'$

$G = 7 + 10 + 6 + 0 = 23$

   – Perform extraction, finding factor:

$T_1 = C + D$
$W = A + BT_1$
$X = B'T_1 + BC'D'$
$Y = CD + C'D'$
$Z = D'$

$G = 2 + 4 + 7 + 6 + 0 = 19$

Design Example **(continued)**

1. Optimization (continued)

   a. Multiple-level using transformations

   $T_1 = C + D$

   $W = A + BT_1$

   $X = B'T_1 + BC'D'$

   $Y = CD + C'D'$

   $Z = D'$

   $G = 19$

   – An additional extraction not shown in the text since it uses a Boolean transformation: ($C'D' = (C + D)' = T_1'$):

$W = A + BT_1$

$X = B'T_1 + BT_1'$

$Y = CD + T_1'$

$Z = D'$

$G = 2 + 4 + 6 + 4 + 0 = 16$

Design Example **(continued)**

1. Technology Mapping

- Mapping with a library containing inverters and 2-input NAND, 2-input NOR, and 2-2 AOI gates