KMap-Logic Minimization Contd..
Outline

• Karnough map simplification
  – 4 variable karnaugh map
  – Don’t care condition
  – Algorithm for better grouping

• Karnough map with >= 5 variable
### 4-Variable Karnaugh Map

#### Table

<table>
<thead>
<tr>
<th>CD</th>
<th>AB</th>
<th>00</th>
<th>01</th>
<th>11</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>m0</td>
<td>m1</td>
<td>m3</td>
<td>m2</td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>m4</td>
<td>m5</td>
<td>m7</td>
<td>m6</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>m12</td>
<td>m13</td>
<td>m15</td>
<td>m14</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>m8</td>
<td>m9</td>
<td>m11</td>
<td>m10</td>
<td></td>
</tr>
</tbody>
</table>

#### Diagram

- **A’CD’**
- **AC’D**

**Function**

\[ F = AC'D + A'CD' + B \]

*Note the row and column orderings. Required for adjacency.*
Find a POS Solution

Find solutions to groups of 0’s to find $F'$
Invert to get $F$ then use DeMorgan’s

$$F' = A'D + A'B + AB'CD'$$
$$F = (A+D)(A+B')(A'+B+C'+D)$$
Don’t Care

• A **don't-care term** is an input to a function that the designer does not care about
• Because that input would never happen
• Example:
  – BCD number (0-9, A-F) are 4 bits, don’t care about input A-F
  – Suppose a system have 5 type of input
    • Unfortunately we can’t have 2 input line
    • Make 3 input line and last 3 sequence as don’t care
    • S0, S1, S2,S3,S4, X,X,X == > 000, 001....,111
Dealing With Don’t Cares

\[ F = \sum m(1, 3, 7) + \sum d(0, 5) \]

\[ A' \overline{B}'C + \overline{A}B'C + A'BC + ABC = C \]

Circle the x’s that help get bigger groups of 1’s
Don’t circle the x’s that don’t
7 Segment Display

[Diagram of a 7-segment display with labels a, b, c, d, e, f, and g]
Activation of LEDs

- 0: a,b,c,d,e,f
- 1: b,c
- 2: a,b,g,e,d
- 3: a,b,g,c,d
- 4: f,g,b,c
- 5: a,f,g,c,d
- 6: a,f,g,c,d,e
- 7: a,b,c
- 8: a,b,c,d,e,f,g
- 9: a,b,c,d,f,g
BCD to 7 Segment Display

• BCD are 4 bit
• Design a decoder to drive 7 segment LED
Prime Implicants

• A group of one or more 1’s which are adjacent and can be combined on a Karnaugh Map is called an implicant.

• The biggest group of 1’s which can be circled to cover a given 1 is called a prime implicant.
  — They are the only implicants we care about.
Are there any additional prime implicants in the map that are not shown above?
All The Prime Implicants

When looking for a minimal solution – only circle prime implicants… A minimal solution will never contain non-prime implicants
Essential Prime Implicants

Not all prime implicants are required...

A prime implicant which is the only cover of some 1 is *essential* – a minimal solution requires it.
A Minimal Solution Example

F = AB' + BC + AD

<table>
<thead>
<tr>
<th></th>
<th>00</th>
<th>01</th>
<th>11</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>01</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Minimum

Not required…
Another Example

<table>
<thead>
<tr>
<th>CD</th>
<th>AB</th>
<th>00</th>
<th>01</th>
<th>11</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td></td>
<td>1 0</td>
<td>0 0</td>
<td>1 1</td>
<td>1 0</td>
</tr>
<tr>
<td>01</td>
<td></td>
<td>1 1</td>
<td>0 0</td>
<td>0 0</td>
<td>0 1</td>
</tr>
<tr>
<td>11</td>
<td></td>
<td>1 1</td>
<td>1 1</td>
<td>1 1</td>
<td>0 0</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td>1 0</td>
<td>0 0</td>
<td>1 1</td>
<td>1 0</td>
</tr>
</tbody>
</table>
Another Example

A’B’ is not required…

Every one one of its locations is covered by multiple implicants

After choosing essentials, everything is covered…
Finding the Minimum Sum of Products

1. Find each essential prime implicant and include it in the solution.
2. Determine if any minterms are not yet covered.
3. Find the minimal # of remaining prime implicants which finish the cover.
Yet Another Example
(Use of non-essential primes)
Yet Another Example
(Use of non-essential primes)

Essentials: $\boxed{A'D'}$ and $\boxed{AD}$
Non-essentials: $\boxed{A'C}$ and $\boxed{CD}$
Solution: $\boxed{A'D'} + \boxed{AD} + \boxed{A'C}$
or $\boxed{A'D'} + \boxed{AD} + \boxed{CD}$
The planes are adjacent to one another (one is above the other in 3D)
Some Implicants in a 5-Variable KMap

Some of these are not prime…
5-Variable KMap Example

Find the minimum sum-of-products for:
\[ F = \Sigma m (0, 1, 4, 5, 11, 14, 15, 16, 17, 20, 21, 30, 31) \]
Find the minimum sum-of-products for:
\[ F = \sum m (0, 1, 4, 5, 11, 14, 15, 16, 17, 20, 21, 30, 31) \]

\[ F = B'D' + BCD + A'BDE \]
6-Variable Karnaugh Map

AB=00

AB=01

AB=10

AB=11
Solution = AC’D’ + CDEF
KMap Summary

• A Kmap is simply a folded truth table
  – where physical adjacency implies logical adjacency

• KMaps are most commonly used hand method for logic minimization

• KMaps have other uses for visualizing Boolean equations
  – you may see some later.
Thanks