A 2-Variable Expression K-Map Example

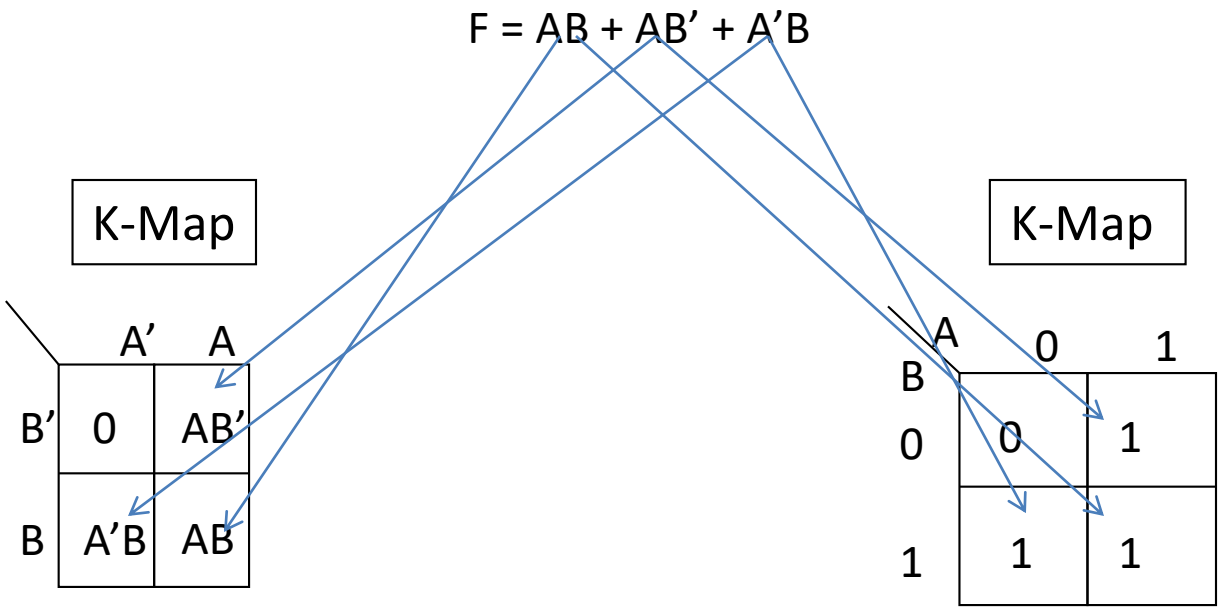
A Worked Example

A 2-Variable Expression K-Map Example

Minimize the Following Expression: F = AB + AB’ + A’B
Technique: Use Karnaugh Map

Since the expression has only two unique variables, the K-Map must have enough cells to cover all of the possible terms. Count the number of variables in the expression (not including their respective complementary representations). This value m becomes the logarithm (base 2) that determines the number of required K-map cells (i.e. $2^m = x$, where x = the number of K-map cells). The expression F = AB + AB’ + A’B has 2 unique variables {A,B}, therefore, $m = 2$ and $2^2 = 4$.

Look at it this Way



Take a look at a couple of ways to view the K-Map organization. The K-map on the left maps the actual terms of the Boolean expression into the K-Map. The K-map on the right shows a '1' in each cell where a term maps into the K-Map

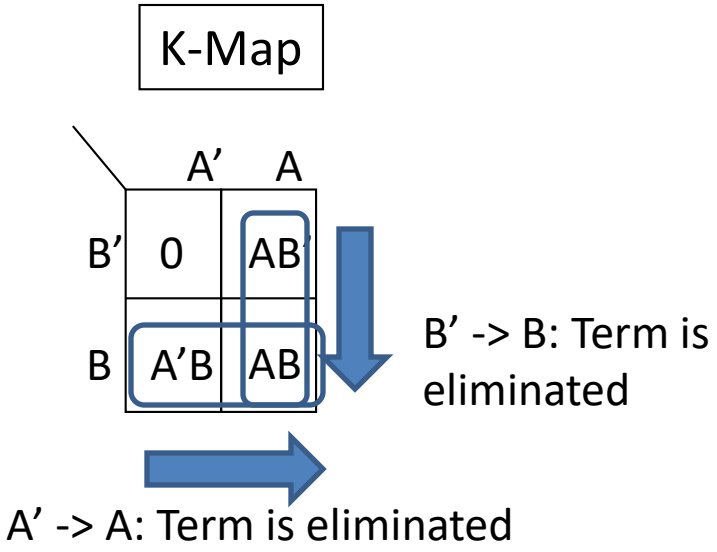
Let's Go A Little Further

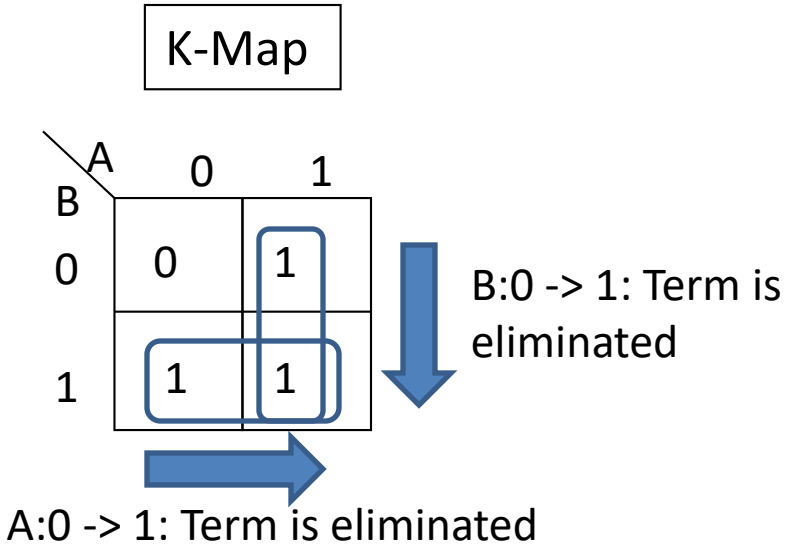
F = AB + AB' + A'B

Selecting Prime Implicants is the next step in the minimization process. We do this using the same power of two rule used to select the number of cells for the K-map itself. Pick the largest power of two groupings possible. Stop grouping when all of the terms belong to a prime implicant grouping.

The Final Step: Writing the Minimized Expression

F = AB + AB’ + A’B





For each of the prime implicants the term that changes across adjacent cells, is eliminated.

Minimized expression: F = A + B