

CSE 140 – Handout #4:

Row/Column Dominance Relationships in Prime Implicant Tables

Row and column dominance relationships can be used to simplify the prime implicant table in the Quine McCluskey algorithm, as explained by the following definitions and theorems.

Definition 1

Two identical rows (columns) **a** and **b** of a reduced prime table are said to be *interchangeable*.

Definition 2

Given two rows **a** and **b** in a reduced prime implicant table, **a** is said to *dominate* **b**, if **a** has checks in all the columns in which **b** has checks and **a** and **b** are not interchangeable.

Definition 3

Given two columns **a** and **b** in a reduced prime implicant table, **a** is said to *dominate* **b**, if **a** has checks in all the rows in which **b** has checks and **a** and **b** are not interchangeable.

Theorem 1

Let **a** and **b** be rows of a reduced prime implicant table. Then, if **a** dominates **b** or **a** and **b** are interchangeable, there exists a minimal sum of products that does not include **b**; *dominated rows can be eliminated*.

Theorem 2

Let **a** and **b** be columns of a reduced prime implicant table. Then, if **b** dominates **a** or **a** and **b** are interchangeable, there exists a minimal sum of products that does not include **a**; *dominating columns can be eliminated*.

Quine McCluskey Control Structure

The Quine McCluskey algorithm is based on the infinite execution of a loop on the prime implicant table until no change occurs in a loop instance. The components of this loop are:

- The identification and the storage of Essential Prime Implicants (EPI)
- The elimination of dominating columns
- The elimination of dominated rows

The algorithm may terminate successfully with a set of EPIs and an empty table. In case it terminates with a nonempty table, you need to employ alternative methods, such the utilization of logic conjunctions that is explained on page 142 in your text book.

Example:

A prime implicant table based on K-map consisting of a set of unspecified minterms was derived. Once the prime implicant table given below was derived, we observed that the minterm you see in the last column of the table was really not a minterm after all, but a *don't care*. How do we apply the Quine McCluskey algorithm on this table?

[illegible]

Solution: We first eliminate the column m that corresponds to a *don't care*; covering m is not necessary as it corresponds to a *don't care*.

[illegible]

We go on by eliminating the dominating columns: the column h is dominated by the columns e, f, and g. These three columns can therefore be eliminated:

[illegible]

Now we can eliminate the dominated rows; the last two rows, P8 and P9, are being dominated by P7 and P6, respectively:

[illegible]

In the current reduced table, the column a is dominated by the columns d and i, and the column c is dominated by the columns b and k; the columns d, i, b and k can therefore be eliminated:

[illegible]

In the current reduced table, the rows P3 and P4 and the rows P1 and P5 are interchangeable. We can eliminate either P3 or P4 and either P1 or P5. Let us eliminate P1 and P3:

[illegible]

There is no other dominance relationship among the rows or the columns in the reduced prime implicant table. As each of the remaining rows correspond to an EPI, the algorithm successfully terminates with the following solution: $P_2 + P_4 + P_5 + P_6 + P_7$.