Project I

K-Maps

The American University in Cairo

School of Science and Engineering

Department of Computer Science and Computer Engineering

Digital Design 1

CSCE2301/230

Project Report

Name: Rana El Gahawy

ID: 900202822

March 25, 2022

Introduction:

This Project gets a Boolean function of an input of 2 or 3 variables using its minterms as decimal numbers, then it uses these minterms to create the corresponding Karnaugh-Map and it print it to the screen and it formulate the minimal simplified Boolean expression that can be used to represent the function.

Program Design:

The program makes sure that the user uses only 2 or 3 variables as an input to the function he/she want to generate the K-Map and the simplified Boolean expression for, then it checks for the special cases:

- If the number of minterms = 0

  Then without completing program it is stated that the simplified Boolean expression is 0 and it presents the corresponding K-Map full of zeros and the program quits.
- If the number of minterms = 2^num of inputs

  Then without completing program it is stated that the simplified Boolean expression is 1 and it presents the corresponding K-Map full of ones and the program quits.

If it is not a special case the program proceeds to put the minterms in their right positions in the K-Map then it displays the K-Map and it moves on to calculated the simplified Boolean expression.

This program uses The Quine-McCluskey Minimization Technique also known as The Tabular Method to get the minimal simplified Boolean expression. First, it gets the corresponding binary numbers of the entered decimal minterms then it distributes them into different groups according to the number of ones in their binary number. Second it compares each group with the next one

and check if there is two Boolean expressions that are different in only one bit, if there was any it will get a new Boolean expression where it will replace this different bit with the letter ‘t’. Then it will keep comparing the new groups with the groups after it until it finishes. Now, we will have all the prime implicants ready, we will just need to identify the essential implicants for the function. It gets these essential implicants by considering how many times each minterm was included as a part of the Prime Implicants, if there was an implicant that was found in only one prime implicant then this implicant is for sure an essential prime implicant and it it’s a product term in the final Boolean expression..∴

Instructions:

The user will first be asked to enter the number of the input variables and it will keep asking for them as long as they are not 2 or 3.

Then after that that it will ask for the number of minterms:

- If the number of minterms = 0

  Then it is a special case a mentioned above
- If the number of minterms = 2^num of inputs

  Then it is a special case a mentioned above
- If the number of minterms > 2^num of inputs

  The program will print that it is an invalid number, and it will keep asking for the correct number of minterms
- If the number of minterms < 2^num of inputs:

  In this case the program will proceed to the next step which is getting the minterms from the user.

The user will them be asked to enter the specified number of minterms ad for each minterm it will be checked if it is less than or equal < 2^num of inputs +1 or not.

After that the program will process al the input data and it will print the corresponding K-Map and the simplified Boolean Expression.