CSCE2301 – Digital Design I

FALL 2020

Project 1

Quine-McCluskey Logic Minimization

Andrew Sinout Shenouda – 900182668  
Mohi Eldin Hashem El Ulabi– 900184260

Objective:

The objective of this project is to make you more familiar with the Quine-McCluskey Logic Minimization algorithm.

Functions used in the program:

1. bool minterms(string filename, int& var, vector<int>& minterm, vector<int>& dontcare)
2. vector<pair<string,string>> dec\_to\_bin(int var, vector<int>& minterm)
3. int count\_ones(int s,string x)
4. bool found (int x, string y)
5. vector<pair<string,string>> comparison (int v, vector<pair<string,string>> a, vector<pair<string,string>> b, vector<bool>& check\_a,vector<bool>& check\_b)
6. int main ()
7. bool minterms(string filename, int& var, vector<int>& minterm, vector<int>& dontcare):

* Inputs:

1. This is a Boolean function that takes a file name in the format of string.
2. An empty variable called var in the format of int by reference.
3. Two empty vectors named minterm and dontcare in the format of vector of int by reference too.

* Implementation:

The goal of this function is to validate the inputs from the text file before the QM process. We first, open the file inside the function and extract the information into the parameters that we passed by reference above. The first line of the text file is the number of variables which is saved in the var parameter. The second line contains minterms and we use built in functions and string streams to push the minterms in the form of integers into the minterm array. We do the exact same things with the third line which contains the don’t care terms and we push them into the dontcare vector. Now that we have extracted our data, we need to validate our data. The way we perform this validation is that we make sure that the minterms and don’t care terms are always one less than two to the power the number of variables. When this condition is satisfied, the function returns true. Otherwise, it returns false and in the main program it outputs a message saying that the data is incorrect.

* Output:

1. It returns true if there are no any errors and if there any errors regarding the number of variables and the minterms it returns false.
2. It also returns the number of variables and the vectors of the minterms and the don’t care terms by reference.
3. vector<pair<string,string>> dec\_to\_bin(int var, vector<int>& minterm):

* Inputs:

1. A variable called var in the format of int that holds the number of variables.
2. A vector of int that is passed by reference.

* Implementation:
* Output:

1. The function returns vector of pairs of minterms or the don’t care terms, where the pair hold the midterms or the don’t care terms in binary and the other string stores them in decimal that corresponds to.
2. int count\_ones(int s,string x):

* Inputs:

1. A variable called s in the format of int that holds the number of variables.
2. A string that contains zeros and ones.

* Implementation:
* Output:

1. This function returns int value which contains the number of ones in that string.
2. bool found (int x, string y):

* Inputs:

1. A variable called s in the format of int that have the minterm that want to search for.
2. A string y that contains the minterms of combined implicant that we want to search for the minterm between them.

* Implementation:

We use this function to confirm that the int x, is in fact a minterm present in the string of minterms corresponding to the desired implicant. We use the same procedure as in the minterm function to turn the string into a vector of integers and then compare the values with x to return true if it is present, and false otherwise.

* Output:

1. This function returns true if the minterm x is found in string y.
2. vector<pair<string,string>> comparison (int v, vector<pair<string,string>> a, vector<pair<string,string>> b, vector<bool>& check\_a,vector<bool>& check\_b):

* Inputs:

1. A variable called s in the format of int that have the minterm that want to search for.
2. A string y that contains the minterms of combined implicant that we want to search for the minterm between them.

* Implementation:
* Output:

1. This function returns true if the minterm x is found in string y.
2. int main ():

Instantiated variables in the beginning of the program

int v,s;

string name;

vector<int> m;

vector<int> d;

vector<pair<string,string>> mm;

vector<pair<string,string>> dd;

vector<vector<pair<string,string>>> groups;

vector<vector<vector<pair<string,string>>>> itterations;

vector<pair<string,string>> PI;

bool b;