EE 465 Fall 2014

Final Design Project

Project Team – Anh Q. Ho, Yu Zheng.

Due Date – Dec. 18- 2014

Introduction

The importance of this project is for students to utilize the acquired skills from the labs throughout the semester. The project for this course is passed on a question in an integrated circuit (IC) design contest held by Chip Implementation Center (CIC) in Taiwan. The task of this project is to design a Gravity Center Calculator (GCC).

Requirements

The basic requirement is to have a functionally design. Below are constraints to complete the project.

Project Constraint

* Project should be coded in Verilog using ModelSim.
* Synthesized by RTL compiler.
* Encounter layout.
* Design optimization – depend on group.

Design Content

Input/Outputs

GCC is to speed up the computation of gravity center in an automatic control system or computer animation. The inputs of this calculator are the 2-Dimensional coordinate and the weight at each point, and the system will return the center of mass or of the total points.

Function Description

The formula of the center of gravity is describe is follow:

If i is the number of ith point

if (i <7) then:

if (i > 6) then:

For the 7th point and beyond the program should be able to compare with the 6 current points and to replace the furthest point in the system. The comparison will start with the distance between the points and the center of mass. Below is the comparison test of the system and shows the importance of the comparison by distance, x, y-coordinate and weight. The point further away or larger mass will be replace by the new point. 1 – most important, and 4 – least important.

1. Distance from the location of the weight to the center of mass.
2. X-coordinate.
3. Y-coordinate.
4. Weight.

Design Method

In order to design a functional gravity center calculator. The group's design will include 4 different modules (GCC (main), Distance, Comparison, Center).The figure below shows the relationship between each modules.

The Inputs and Outputs of the systems are as follows:

X: x-coordinate, 8-bit input unsigned data of the ith point.

Y: y-coordiante, 8-bit input unsigned data of the ith point.

W: non-zero 4-bit unsigned weight of the ith point.

Xc, Yc: 8-bit unsigned rounded to the closest integer output of the x, y coordinate of the center of mass.

The basic concept is to use the input of module GCC (X, Y, and W) and output the coordinate of the center of mass (Xc, Yc) using module Center. Module Center includes the divider, and to operate such that by inputting the SUMXW or SUMYW and SUMW to determine the center of mass.

SUMXW, SUMYW and SUMW are calculated in module GCC.

GCC – the main code, gather passing out inputs and gather information from different module.

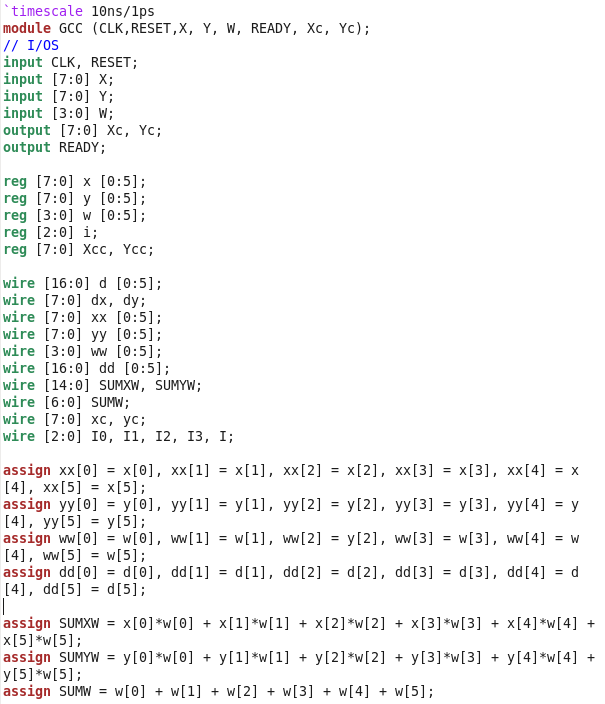
Distance – Calculate the distance from one point to the center of mass. Using the inputs of X, Y, and the previous center of mass Xc and Yc. Distance is calculate by:

Comparison -

Divider

Result

Appendix A.1 – Verilog Code [GCC.v]



![](data:None;base64,)

![](data:None;base64,)

Appendix A.2 Verilog Code [Comparator.v]

![](data:None;base64,)

Appendix A.3 Verilog Code [Center.v]

![](data:None;base64,)

Appendix A.4 Verilog Code [Distance.v]

![](data:None;base64,)

Appendix A.5 [TestFixture.v] TestBench

Appendix B.2 – Acknowledgement

Unknown. “2002 University/College IC Contest Cell-Based IC for Graduate Level”. College Integrated Circuit. <http://home.eng.iastate.edu/~cnchu/465/lab/proj/proj-GCC.pdf>