Project GF2: Software
First Interim Report
Software Design Team 1

George Ayris	James Glanville	Andrew Holt
gdwa2	jg597	ah635
Emmanuel	Emmanuel	Emmanuel

21 May 2013

1 Introduction

This project requires the development of a logic simulation program, implemented in C++. When completed, the application will read a definition file containing a list of devices and the connections between them. It will then graphically display the values of specified monitor points in the circuit as the simulation is run. Some legacy code has been provided, but lacks a scanner, parser and GUI which will be developed in this project.

1.1 Teamwork Planning

The GUI development requires learning and reference to the wxWidgets and OpenGL packages, so makes sense for one team member to focus efforts here. The scanner and parser will require robust coding and testing, so it was decided to combine the two as a peer programming project to allow implementation and testing to be carried out by different team members.

The work is to be distributed as follows: Andrew will code the GUI, and James and George will write and test the scanner and parser. The time frame for the development is that the software should be designed by the end of Tuesday 21st May; implemented and unit tested by Tuesday 28th; allowing time for system integration and testing by 11.00am on Friday 31st May.

2 Syntax Specification

The syntax for the circuit definition files was defined and described using the following EBNF grammar:

```
DEFINITION = '{' DEVICES CONNECTIONS MONITORS INIT '}'

DEVICES = 'DEVICES' '{' device {device} '}'
device = devicename '=' devicetype [ '(' digit {digit} ')' ] ';'
devicename = letter {letter | digit}
devicetype = 'AND' | 'NAND' | 'OR' | 'NOR' | 'XOR' | 'DTYPE' | 'CLK' | 'SW'

CONNECTIONS = 'CONNECTIONS' '{' connection {connection} '}'
connection = input '<=' output ';'
input = letter {letter|digit} ['.'letter|digit{letter|digit}]
output = letter {letter|digit} ['.'letter|digit{letter|digit}]
```

```
MONITORS = 'MONITORS' '{' monitor {monitor} '}'
monitor = monitorname '<=' output ';'
monitorname = letter{letter|digit}

INIT = 'INIT' '{' {init} '}'
init = devicename '=' digit{digit} ';'
```

3 Semantics Specification

4 Error Handling

5 Example Circuits and Definition Files

5.1 XOR Circuit Composed of NAND Gates

5.2 3-bit Gray Code Counter