Appendix only

A Test definition files

A.1 Test 101: Binary adder

2-bit adder

```
//This one is a 2-bit binary adder with combinational logic
DEVICES AND A1 2,
AND A2 2,
XOR X1,
XOR X2,
XOR X3,
SWITCH S0 1,
SWITCH S1 1,
SWITCH S2 1,
SWITCH S3 1;
CONNECT S0 => A1.I1,
S2 => A1.I2,
S1 => A2.I1,
S3 => A2.I2,
S0 => X1.I1,
S2 => X1.I2,
S1 => X2.I1,
S3 => X2.I2,
A1 => X3.I1,
X2 => X3.I2;
MONITOR X1,
X3,
A2;
```

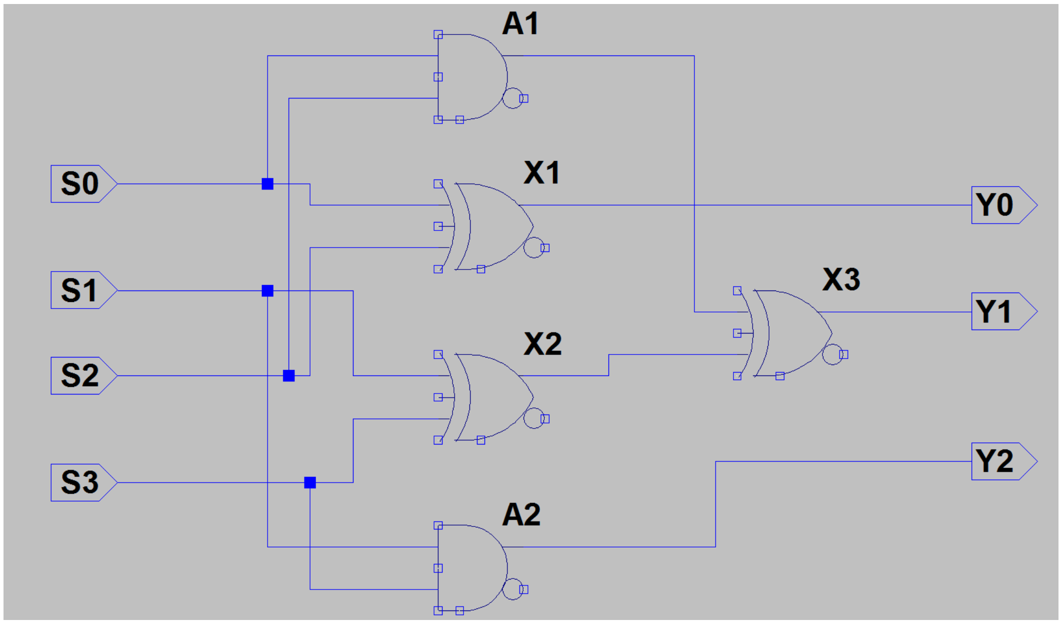
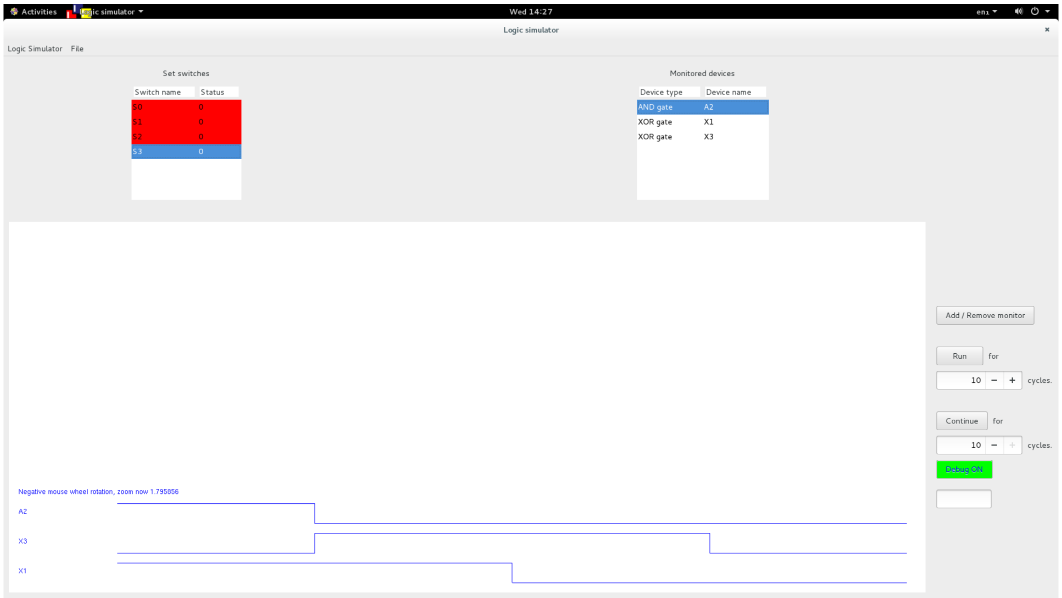


Figure 1: File 1, Binary adder



Set switches

Switch name	Status
S0	0
S1	0
S2	0
S3	0

Monitored devices

Device type	Device name
AND gate	A2
XOR gate	X1
XOR gate	X3

Negative mouse wheel rotation, zoom now 1.795856

Figure 2: Binary Adder Output

A.2 Test 102: Vending machine

The control circuit of a vending selling machines at 15p. S1 and S2 corresponds to insertion of 10p and 5p coins. Inserting 2 coins at the same time is not allowed. S3 and S4 sets and resets the flip flops.

```
//a vending machine where S1 and S2 indicate whether a 10p or 5p are inserted
DEVICES SWITCH S1 1,
SWITCH S2 1,
SWITCH S3 0,
SWITCH S4 1,
AND A1 2,
AND A2 2,
AND A3 2,
AND A4 2,
AND A5 2,
AND A6 2,
AND A7 2,
AND A8 2,
OR O1 4,
OR O2 3,
D_TYPE D1,
D_TYPE D2,
NAND N2 1,
CLOCK CLK1 5,
AND A9 2;
CONNECT S2 => N2.I1,
S1 => A1.I1,
D1.QBAR => A1.I2,
A1 => O1.I1,
S1 => A2.I1,
D2.QBAR => A2.I2,
A2 => O1.I2,
D1.Q => A3.I1,
D2.QBAR => A3.I2,
A3 => O1.I3,
A3 => A8.I1,
D1.QBAR => A4.I1,
D2.Q => A4.I2,
A4 => A6.I2,
A4 => A7.I1,
S2 => A5.I1,
D2.QBAR => A5.I2,
A5 => O2.I3,
S2 => A6.I1,
A6 => O1.I4,
N2 => A7.I2,
A7 => O2.I1,
S1 => A8.I2,
A8 => O2.I2,
O1 => D1.DATA,
O2 => D2.DATA,
CLK1 => D1.CLK,
CLK1 => D2.CLK,
D1.Q => A9.I1,
D2.Q => A9.I2,
S3 => D1.SET,
S3 => D2.SET,
S4 => D1.CLEAR,
S4 => D2.CLEAR;
MONITOR A9,
D1.Q,
D2.Q;
```

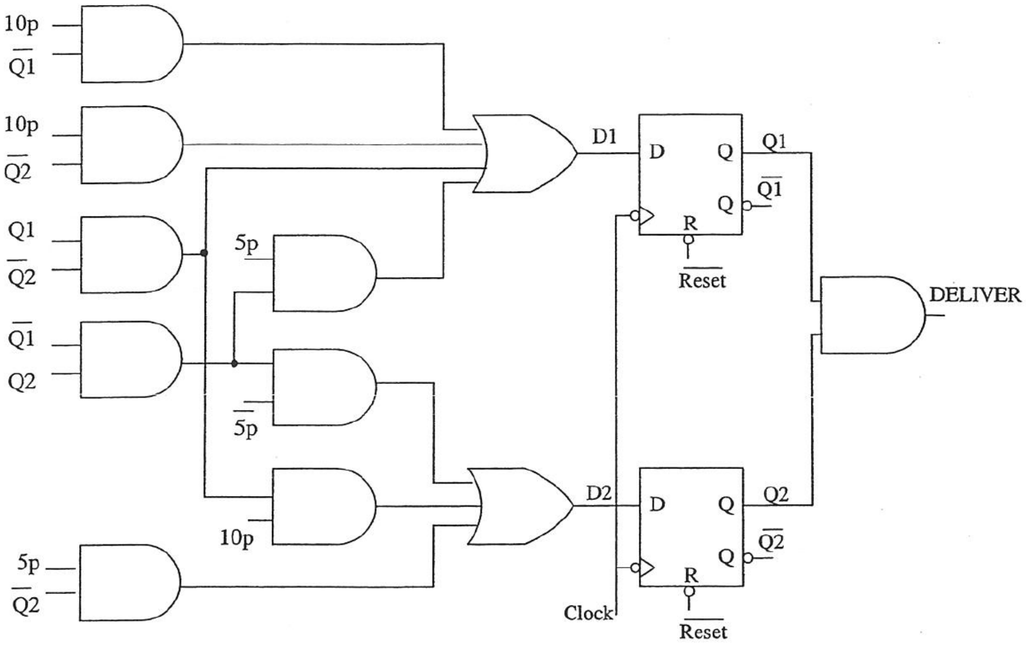
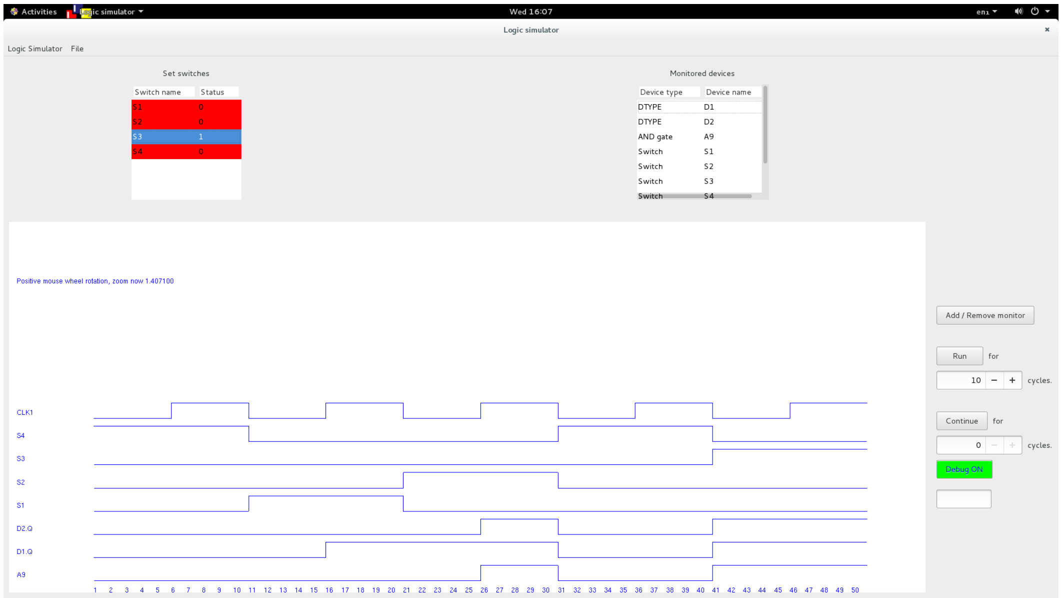


Figure 3: File 2, Vending Machine



Switch name	Status
S1	0
S2	0
S3	1
S4	0

Device type	Device name
DTYPE	D1
DTYPE	D2
AND gate	A9
Switch	S1
Switch	S2
Switch	S3
Switch	S4

Positive mouse wheel rotation, zoom now 1.407100

Figure 4: Vending Machine Output

A.3 Test103: Incorrect test definition file

```
//a faulty vending machine where S1 and S2 indicate whether a 10p or 5p are inserted
DEVICES SWITCH S1 1,
SWITCH S2 1,
SWITCH SWITCH 1, //Unexpected Token: SWITCH cannot be device name
SWITCH S4 3, //Semantic Error: Invalid switch
AND A1, //Unexpected Token: Number of inputs not defined
AND A2 2,
AND A3 2,
AND A4 22, //Semantic Error: Invalid gate
AND A5 2,
AND A6 2,
AND A7 2,
AND A8 2,
OR jiangxueaihaozhe 4, //Warning: Name too long
OR O2 3,
D_TYPE O2, //Warning: Name conflict
D_TYPE D2,
NAND N2 1,
CLOCK CLK1 5,
AND u2he*2w 2; //Unexpected Token: Device name cannot contain *
CONNECT S2 => N2.I1,
S1 => A1.I1,
D1.QBAR => A1.I2,
A1 => O1.I1,
S1 => A2.I1,
D2.QBAR => A2.I2,
A2 => O1.I2,
D1.Q => A3.I1,
D2.QBAR => A3.I2,
A3.I2 => D1.Q, //Syntax Error: Not Output & Not Input
A3 => O1.I3,
A3 => A8.I1,
A3.I1 => A4.I2, //Syntax Error: Not Output
D1.QBAR => A4.I1,
D2.Q => A4.I2,
A4 => A6.I2,
A4 => A7.I1,
S2 => A5.I1,
D2.QBAR => A5.I2,
A4 => A5.I3, //Semantic Error: Undefined pin
A5 => O2.I3,
S2 => A6.I1,
A6 => O1.I4,
N2 => A7.I2,
A7 => O2.I1,
S1 => A8.I2,
A8 => O2.I2,
O1 => D1.DATA,
O2 => D2.DATA,
CLK1 => D1.CLK,
CLK1 => D2.CLK //Unexpected Token: Missing a stop symbol
D1.Q => A9.I1,
D2.Q => A9.I2,
S3 => D1.SET,
S3 => D2.SET,
S4 => D1.CLEAR,
D1.QBAR => D2.Q, //Syntax Error: Not Input
S4 => D2.CLEAR;
// Semantic Error: Floating Input
MONITOR A9,
D1.Q,
D3.Q, //Semantic Error: Undefined device
D2.Q;

Expect Name Symbol
***Unxpected Token
SWITCH SWITCH 1,
       ^
***Error: Invalid Switch
SWITCH S4 3,
          ^
Expect a Number
***Unxpected Token
AND A1,
      ^
***Error: Invalid Gate
AND A4 22,
       ^
***Warning: Name Too Long
OR jiangxueaihaozhe 4,
   ^
***Warning: Name Conflict
D_TYPE O2,
       ^
Expect Name Symbol
***Unxpected Token
AND u2he*2w 2;
    ^
***Error: Undefined Device
S1 => A1.I1,
      ^
***Error: Undefined Device
```

```
D1.QBAR => A1.I2,
^
***Error: Undefined Device
D1.QBAR => A1.I2,
           ^
***Error: Undefined Device
A1 => O1.I1,
^
***Error: Undefined Device
A1 => O1.I1,
      ^
***Error: Undefined Device
A2 => O1.I2,
      ^
***Error: Undefined Device
D1.Q => A3.I1,
^
***Error: Not an Output Pin
A3.I2 => D1.Q,
^
***Error: Undefined Device
A3.I2 => D1.Q,
         ^
***Error: Undefined Device
A3 => O1.I3,
      ^
***Error: Not an Output Pin
A3.I1 => A4.I2,
^
***Error: Undefined Device
A3.I1 => A4.I2,
         ^
***Error: Undefined Device
D1.QBAR => A4.I1,
^
***Error: Undefined Device
D1.QBAR => A4.I1,
           ^
***Error: Undefined Device
D2.Q => A4.I2,
        ^
***Error: Undefined Device
A4 => A6.I2,
^
***Error: Undefined Device
A4 => A7.I1,
^
```

```
***Error: Undefined Device
A4 => A5.I3,
^
***Error: Undefined Pin
A5 => O2.I3,
      ^
***Error: Undefined Device
A6 => O1.I4,
      ^
***Error: Undefined Pin
A7 => O2.I1,
      ^
***Error: Undefined Pin
A8 => O2.I2,
      ^
***Error: Undefined Device
O1 => D1.DATA,
^
***Error: Undefined Device
O1 => D1.DATA,
      ^
***Error: Undefined Pin
O2 => D2.DATA,
^
***Error: Undefined Device
CLK1 => D1.CLK,
        ^
Expect Stop Symbol
***Unxpected Token
D1.Q => A9.I1,
^
***Error: Undefined Device
D2.Q => A9.I2,
        ^
***Error: Undefined Device
S3 => D1.SET,
^
***Error: Undefined Device
S3 => D1.SET,
      ^
***Error: Undefined Device
S3 => D2.SET,
^
***Error: Undefined Device
S4 => D1.CLEAR,
^
***Error: Undefined Device
```

```
S4 => D1.CLEAR,
      ^
***Error: Undefined Device
D1.QBAR => D2.Q,
^
***Error: Not an Input Pin
D1.QBAR => D2.Q,
           ^
***Error: Undefined Device
S4 => D2.CLEAR;
^
Unconnected Input : D2.CLEAR
Unconnected Input : D2.SET
Unconnected Input : D2.DATA
Unconnected Input : O2.CLEAR
Unconnected Input : O2.SET
Unconnected Input : O2.CLK
Unconnected Input : O2.DATA
Unconnected Input : O2.I3
Unconnected Input : O2.I2
Unconnected Input : O2.I1
Unconnected Input : jiangxue.I4
Unconnected Input : jiangxue.I3
Unconnected Input : jiangxue.I2
Unconnected Input : jiangxue.I1
Unconnected Input : A7.I1
Unconnected Input : A6.I2
Unconnected Input : A3.I1
***Error: Floating Input
MONITOR A9,
^
***Error: Undefined Device
MONITOR A9,
        ^
***Error: Undefined Device
D1.Q,
^
***Error: Undefined Device
D3.Q,
^
Total Syntax Error Count: 7
Total Semantics Error Count: 38
Total Warning Count: 2
```

A.4 Test 104: Logic Gates

Combination of Logic gates, which can be reduced to the logic expression, $Z = \bar{B}\bar{F} + \bar{D} + \bar{E}(\bar{A} + \bar{B})$

```
DEVICES SWITCH S1 1,
SWITCH S2 1,
SWITCH S3 1,
SWITCH S4 1,
SWITCH S5 1,
SWITCH S6 1,
NAND A1 2,
NAND A2 2,
NAND A3 2,
NAND A4 2,
NAND A5 2,
NAND A6 3,
NAND A7 3,
NAND A8 3,
NOR N1 1,
NOR N2 1,
NOR N3 1;
CONNECT S1 => A1.I1,
S2 => A1.I2,
A1 => A7.I2,
A1 => A5.I1,
S1 => N1.I1,
N1 => A3.I1,
S3 => A3.I2,
A3 => A5.I2,
A3 => A6.I2,
S5 => A2.I1,
S6 => A2.I2,
A2 => A4.I1,
S2 => N2.I1,
N2 => A4.I2,
A4 => A8.I3,
S5 => N3.I1,
N3 => A7.I1,
A5 => A6.I1,
S4 => A6.I3,
S4 => A8.I2,
A6 => A7.I3,
A7 => A8.I1;
MONITOR A8;
```

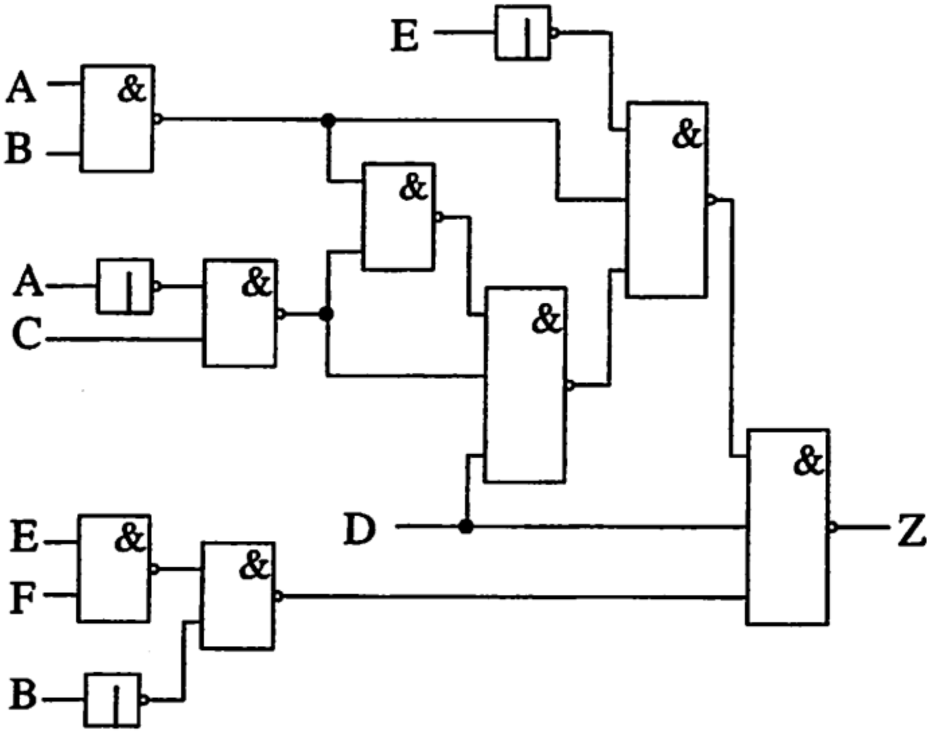


Figure 5: File 3

A.5 Test 107: 3-Bit Counter

3-Bit Counter

```
DEVICES AND A1 2,
XOR X1,
XOR X2,
XOR X3,
D_TYPE D1,
D_TYPE D2,
D_TYPE D3,
CLOCK CLK1 5,
SWITCH I 1,
SWITCH Reset 0,
SWITCH Set 0;
CONNECT I => X1.I2,
D1.Q => X1.I1,
CLK1 => D1.CLK,
X1 => D1.DATA,
D1.Q => A1.I1,
D1.Q => X2.I2,
D2.Q => X2.I1,
X2 => D2.DATA,
CLK1 => D2.CLK,
D2.Q => A1.I2,
A1 => X3.I1,
D3.Q => X3.I2,
X3 => D3.DATA,
CLK1 => D3.CLK,
Reset => D1.CLEAR,
Reset => D2.CLEAR,
Reset => D3.CLEAR,
Set => D1.SET,
Set => D2.SET,
Set => D3.SET;
MONITOR D3.Q;
```

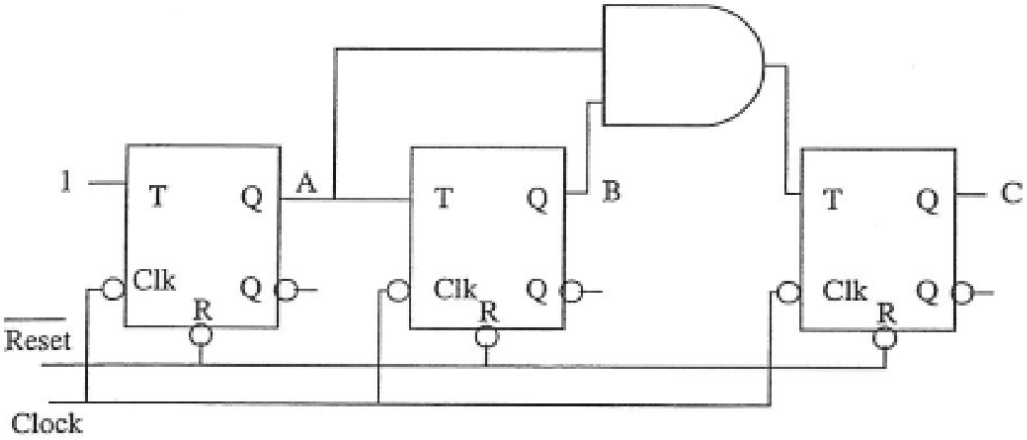
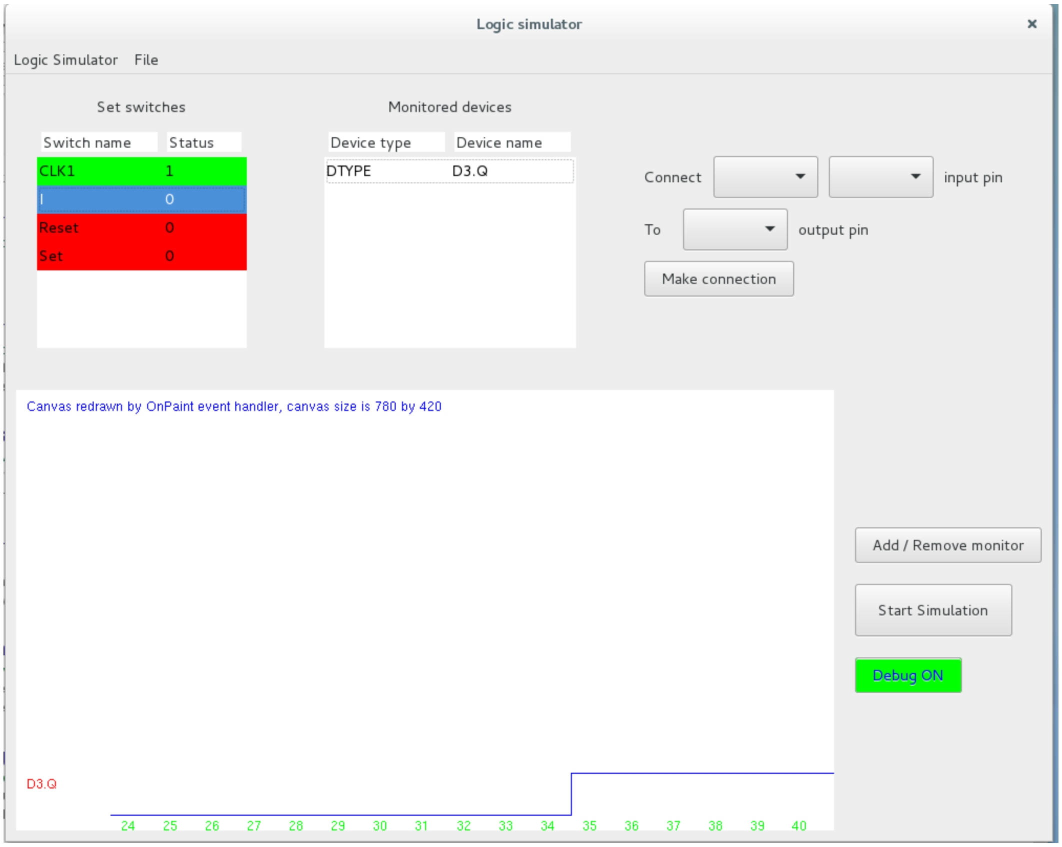


Figure 6: File 4



Switch name	Status
CLK1	1
I	0
Reset	0
Set	0

Device type	Device name
DTYPE	D3.Q

Canvas redrawn by OnPaint event handler, canvas size is 780 by 420

Figure 7: 3-Bit Counter Output

B Logic description language specification

```
file =  ‘DEVICES’, DEV, {‘,’, DEV}, ‘;’, ‘CONNECT’, CON, {‘,’, CON}, ‘;’,
        ‘MONITOR’, MON, {‘,’, MON}, ‘;’;
DEV  =  ‘CLOCK’, DEV_NAME, digit, {digit} |
        ‘SWITCH’, DEV_NAME, ( 1 | 0 ) |
        ‘SIGGEN’, DEV_NAME, ( 1 | 0 ), { 1 | 0 } |
        ‘AND’ | ‘NAND’ | ‘OR’ | ‘NOR’, DEV_NAME, [1], digit |
        ‘D_TYPE’, DEV_NAME |
        ‘XOR’, DEV_NAME;
DEV_NAME  =  (digit | letter | ‘_’), {digit | letter | ‘_’};
CON       =  O_PIN, ‘=>’, I_PIN;
O_PIN     =  DEV_NAME |
             DEV_NAME, ‘.Q’, [‘BAR’];
I_PIN     =  DEV_NAME, ‘.I’, [1], digit |
             DEV_NAME, ‘.’, (‘DATA’|‘CLK’|‘SET’|‘CLEAR’);
MON       =  O_PIN;

letter = "A" | "B" | "C" | "D" | "E" | "F" | "G"
       | "H" | "I" | "J" | "K" | "L" | "M" | "N"
       | "O" | "P" | "Q" | "R" | "S" | "T" | "U"
       | "V" | "W" | "X" | "Y" | "Z" | "a" | "b"
       | "c" | "d" | "e" | "f" | "g" | "h" | "i"
       | "j" | "k" | "l" | "m" | "n" | "o" | "p"
       | "q" | "r" | "s" | "t" | "u" | "v" | "w"
       | "x" | "y" | "z" ;
digit = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9" ;
```

Note: DEV_NAME can be any combination of letter and number and '_', **other than** ”DEVICES”, ”CONNECT”, ”MONITOR”, ”CLOCK”, ”SWITCH”, ”AND”, ”NAND’, ”OR”, ”NOR”, ”D_TYPE”, ”XOR”, ”SIGGEN”, ”I1”, ”I2” etc.

C User guide

D Description of file system

Under the Test directory Test 1 - Test 16 are designed to test the scanner-parser subsystem, which includes reading in different types of characters to test whether the scanner can read the words and lines correctly without any bugs. Also they test how the subsystem will be handling errors in the definition files. These tests are used in the early testing stage when the scanner and parser are initially merged.

Test 101 - Test 107 are test definition files designed to demonstrate functionality of the whole programme. They can be opened in the graphical user interface for testing and illustrate how the logic simulator works. The test definition files are written in such a standard that most of the functionalities designed are shown throughout these definition files. Also it also contains a file with errors to demonstrate how the error reporting works and how to fix a definition file in case there are errors.

In the main directory scanner.cc It is the major cpp file for the scanner part. It performs the reading and processing of the definition files and into words. It interacts with the parser.cc to receive and send symbols of data types. It also reports error. scanner.h it is the header file for the scanner. It contains all the public and private function definitiions and is included in other parts of the programme. parser.cc It is the major cpp file for the parser part. It takes in the symbol and word from scanner and performs a logical process to make up the whole structure of the digital circuit. gui.cc It is the majoy cpp file for the GUI part. It takes the signal and information fed from parser and scanner and turns it into a graphical output. It enables making up and cancelling connections and also changing of the monitored signals. devices.cc It is the file for establishing a device in the memory when constructing the digital circuit. It interacts with the parser and carry out operations when a new device is read in.