**LAB 3 Report**

**Name: Donald Maze-England**

**UT EID: dsm2588**

**Section: 15465**

**Checklist:**

**Part 2 -**

1. Constraint File (Just the uncommented portion)

**Part 3 -**

1. Truth Table of the function
2. K-maps showing minimization of the logic functions (outputs)
3. Algebraic expression of the minimized logic functions (outputs)
4. Verilog codes of module and testbench for structural modelling
5. Simulation waveform for structural modelling
6. Constraint File (Just the uncommented portion)

***Note*** *🡪 The Verilog codes and the uncommented portions of the constraint files should be copied in your lab report and the actual Verilog (.v) and Constraint (.xdc) files need to be zipped and submitted as well on Canvas. You are not allowed to change your Verilog codes after final submission as the TAs may download the submitted codes from Canvas during checkouts. For the truth Table, K-maps minimizations and algebraic expressions, you are free to draw them on paper and then put the pictures in your lab report, but please make sure it is legible for the TAs to grade it properly.*

**Part 2:**

Constraints File:

set\_property PACKAGE\_PIN V17 [get\_ports {cd}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {cd}]

set\_property PACKAGE\_PIN V16 [get\_ports {bd}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {bd}]

set\_property PACKAGE\_PIN W16 [get\_ports {ad}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {ad}]

set\_property PACKAGE\_PIN W13 [get\_ports {ed}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {ed}]

set\_property PACKAGE\_PIN E19 [get\_ports {d0}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {d0}]

set\_property PACKAGE\_PIN U19 [get\_ports {d1}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {d1}]

set\_property PACKAGE\_PIN V19 [get\_ports {d2}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {d2}]

set\_property PACKAGE\_PIN W18 [get\_ports {d3}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {d3}]

set\_property PACKAGE\_PIN U15 [get\_ports {d4}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {d4}]

set\_property PACKAGE\_PIN U14 [get\_ports {d5}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {d5}]

set\_property PACKAGE\_PIN V14 [get\_ports {d6}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {d6}]

set\_property PACKAGE\_PIN V13 [get\_ports {d7}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {d7}]

**Part 3:**

Truth Table:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Decimal | A | B | C | D | a | b | c | d | e | f | g | dp |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 2 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 |
| 3 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| 4 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| 5 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| 6 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| 7 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 8 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 9 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| greater than 9 | | | | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| ... | | | | ... | ... | ... | ... | ... | ... | ... | ... | ... |

K Maps:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a: | AB |  |  |  |
| CD | 00 | 01 | 11 | 10 |
| 00 | 0 | 1 | 1 | 0 |
| 01 | 1 | 0 | 1 | 0 |
| 11 | 0 | 0 | 1 | 1 |
| 10 | 0 | 1 | 1 | 1 |
|  |  |  |  |  |
| a = A'B'C'D+A'BD'+AB+AC | | |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| b: | AB |  |  |  |
| CD | 00 | 01 | 11 | 10 |
| 00 | 0 | 0 | 1 | 0 |
| 01 | 0 | 1 | 1 | 0 |
| 11 | 0 | 0 | 1 | 1 |
| 10 | 0 | 1 | 1 | 1 |
|  |  |  |  |  |
| b = AC+AB+BC'D+BCD' | | |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| c: | AB |  |  |  |
| CD | 00 | 01 | 11 | 10 |
| 00 | 0 | 0 | 1 | 0 |
| 01 | 0 | 0 | 1 | 0 |
| 11 | 0 | 0 | 1 | 1 |
| 10 | 1 | 0 | 1 | **1** |
|  |  |  |  |  |
| c = AB+ AC+B'CD' | |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| d: | AB |  |  |  |
| CD | 00 | 01 | 11 | 10 |
| 00 | 0 | 1 | 1 | 0 |
| 01 | 1 | 0 | 1 | 1 |
| 11 | 0 | 1 | 1 | 1 |
| 10 | 0 | 0 | 1 | 1 |
|  |  |  |  |  |
| d = AB + AC+AD+BCD+BC'D'+B'C'D | | | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| e: | AB |  |  |  |
| CD | 00 | 01 | 11 | 10 |
| 00 | 0 | 1 | 1 | 0 |
| 01 | 1 | 1 | 1 | 1 |
| 11 | 1 | 1 | 1 | 1 |
| 10 | 0 | 0 | 1 | 1 |
|  |  |  |  |  |
| e = D+BC'+AC | |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| f: | AB |  |  |  |
| CD | 00 | 01 | 11 | 10 |
| 00 | 0 | 0 | 1 | 0 |
| 01 | 1 | 0 | 1 | 0 |
| 11 | 1 | 1 | 1 | 1 |
| 10 | 1 | 0 | 1 | 1 |
|  |  |  |  |  |
| f = CD+AB+AC+A'B'D+B'CD' | | |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| g: | AB |  |  |  |
| CD | 00 | 01 | 11 | 10 |
| 00 | 1 | 0 | 1 | 0 |
| 01 | 1 | 0 | 1 | 0 |
| 11 | 0 | 1 | 1 | 1 |
| 10 | 0 | 0 | 1 | 1 |
|  |  |  |  |  |
| g = AB+AC+BCD+A'B'C' | | |  |  |

Minimized Equations:

a = A'B'C'D+A'BD'+AB+AC

b = AC+AB+BC'D+BCD'

c = AB+ AC+B'CD'

d = AB + AC+AD+BCD+BC'D'+B'C'D

e = D+BC'+AC

f = CD+AB+AC+A'B'D+B'CD'

g = AB+AC+BCD+A'B'C'

Verilog Code For 7 Segment Display:

`timescale 1ns / 1ps

//////////////////////////////////////////////////////////////////////////////////

// Company:

// Engineer:

//

// Create Date: 02/21/2018 09:19:49 PM

// Design Name:

// Module Name: 7SegmentDisplay

// Project Name:

// Target Devices:

// Tool Versions:

// Description:

//

// Dependencies:

//

// Revision:

// Revision 0.01 - File Created

// Additional Comments:

//

//////////////////////////////////////////////////////////////////////////////////

module SegmentDisplay(

input A,

input B,

input C,

input D,

output a,

output b,

output c,

output d,

output e,

output f,

output g,

output dp,

output AN0,

output AN1,

output AN2,

output AN3

);

//wires for not gates

wire A\_not, B\_not, C\_not, D\_not;

//wire for and gates

wire AnBnCnD, AnBDn, AB, AC, BCnD, BCDn, BnCDn, AD, BDn, BC, BnCnD, BCn, CD, AnBnD, BCD, AnBnCn, BCnDn;

//register to transmit a 1 constantly

reg sendOne = 1'b1;

//register to transmit a 0 constantly

reg sendZero = 1'b0;

//not gates

not notA (A\_not, A);

not notB (B\_not, B);

not notC (C\_not, C);

not notD (D\_not, D);

//instantiating and gates as needed by the truth table

and and\_AnBnCnD (AnBnCnD, A\_not, B\_not, C\_not, D);

and and\_AnBDn (AnBDn, A\_not, B, D\_not);

and and\_AB (AB, A, B);

and and\_AC (AC, A, C);

and and\_BCnD (BCnD, B, C\_not, D);

and and\_BCDn (BCDn, B, C, D\_not);

and and\_BnCDn (BnCDn, B\_not, C, D\_not);

and and\_AD (AD, A, D);

and and\_BDn (BDn, B, D\_not);

and and\_BC (BC, B, C);

and and\_BnCnD (BnCnD, B\_not, C\_not, D);

and and\_BCn (BCn, B, C\_not);

and and\_CD (CD, C, D);

and and\_AnBnD (AnBnD, A\_not, B\_not, D);

and and\_BCD (BCD, B, C, D);

and and\_AnBnCn (AnBnCn, A\_not, B\_not, C\_not);

and and\_BCnDn (BCnDn, B, C\_not, D\_not);

and andSendZero (AN0, sendZero, sendOne);

//instantiating or gates as needed by the truth table

or or\_a (a, AnBnCnD, AnBDn, AB, AC);

or or\_b (b, AC, AB, BCnD, BCDn);

or or\_c (c, AB, AC, BnCDn);

or or\_d (d, AB, AC, AD, BCD, BCnDn, BnCnD);

or or\_e (e, D, BCn, AC);

or or\_f (f, CD, AB, AC, AnBnD, BnCDn);

or or\_g (g, AB, AC, BCD, AnBnCn);

or orSendOne1(dp, sendOne, sendZero);

or orSendOne2(AN3, sendOne, sendZero);

or orSendOne3(AN2, sendOne, sendZero);

or orSendOne4(AN1, sendOne, sendZero);

endmodule

Test Bench Code for 7 Segment Display:

`timescale 1ns / 1ps

//////////////////////////////////////////////////////////////////////////////////

// Company:

// Engineer:

//

// Create Date: 02/22/2018 08:52:36 AM

// Design Name:

// Module Name: tb\_SegmentDisplay

// Project Name:

// Target Devices:

// Tool Versions:

// Description:

//

// Dependencies:

//

// Revision:

// Revision 0.01 - File Created

// Additional Comments:

//

//////////////////////////////////////////////////////////////////////////////////

`timescale 1ns / 1ps

module tb\_Decoder\_Structural;

//inputs to be defined as registers

reg A;

reg B;

reg C;

reg D;

//outputs to be defined as wires

wire a;

wire b;

wire c;

wire d;

wire e;

wire f;

wire g;

wire dp;

wire AN0;

wire AN1;

wire AN2;

wire AN3;

//Initiat the unit under test (UUT)

SegmentDisplay uut (

.A(A),

.B(B),

.C(C),

.D(D),

.a(a),

.b(b),

.c(c),

.d(d),

.e(e),

.f(f),

.g(g),

.dp(dp),

.AN0(AN0),

.AN1(AN1),

.AN2(AN2),

.AN3(AN3)

);

initial begin

//initialze inputs

A = 0;

B = 0;

C = 0;

D = 0;

#50; //wait 50 seconds for global reset to finish

//stimulus - all input combinations followed by some wait time to observe the o/p

$display ("TC01");

if ({a,b,c,d,e,f,g} != 7'b0000001) $display ("Result is wrong");

A = 0;

B = 0;

C = 0;

D = 1;

#50

$display ("TC02");

if ({a,b,c,d,e,f,g} != 7'b1001111) $display ("Result is wrong");

A = 0;

B = 0;

C = 1;

D = 0;

#50

$display ("TC03");

if ({a,b,c,d,e,f,g} != 7'b0010010) $display ("Result is wrong");

A = 0;

B = 0;

C = 1;

D = 1;

#50

$display ("TC04");

if ({a,b,c,d,e,f,g} != 7'b0000110) $display ("Result is wrong");

A = 0;

B = 1;

C = 0;

D = 0;

#50

$display ("TC05");

if ({a,b,c,d,e,f,g} != 7'b1001100) $display ("Result is wrong");

A = 0;

B = 1;

C = 0;

D = 1;

#50

$display ("TC06");

if ({a,b,c,d,e,f,g} != 7'b0100100) $display ("Result is wrong");

A = 0;

B = 1;

C = 1;

D = 0;

#50

$display ("TC07");

if ({a,b,c,d,e,f,g} != 7'b1100000) $display ("Result is wrong");

A = 0;

B = 1;

C = 1;

D = 1;

#50

$display ("TC08");

if ({a,b,c,d,e,f,g} != 7'b0001111) $display ("Result is wrong");

A = 1;

B = 0;

C = 0;

D = 0;

#50

$display ("TC11");

if ({a,b,c,d,e,f,g} != 7'b0000000) $display ("Result is wrong");

A = 1;

B = 0;

C = 0;

D = 1;

#50

$display ("TC12");

if ({a,b,c,d,e,f,g} != 7'b0001100) $display ("Result is wrong");

A = 1;

B = 0;

C = 1;

D = 0;

#50

$display ("TC13");

if ({a,b,c,d,e,f,g} != 7'b1111111) $display ("Result is wrong");

A = 1;

B = 0;

C = 1;

D = 1;

#50

$display ("TC14");

if ({a,b,c,d,e,f,g} != 7'b1111111) $display ("Result is wrong");

A = 1;

B = 1;

C = 0;

D = 0;

#50

$display ("TC15");

if ({a,b,c,d,e,f,g} != 7'b1111111) $display ("Result is wrong");

A = 1;

B = 1;

C = 0;

D = 1;

#50

$display ("TC16");

if ({a,b,c,d,e,f,g} != 7'b1111111) $display ("Result is wrong");

A = 1;

B = 1;

C = 1;

D = 0;

#50

$display ("TC17");

if ({a,b,c,d,e,f,g} != 7'b1111111) $display ("Result is wrong");

A = 1;

B = 1;

C = 1;

D = 1;

#50

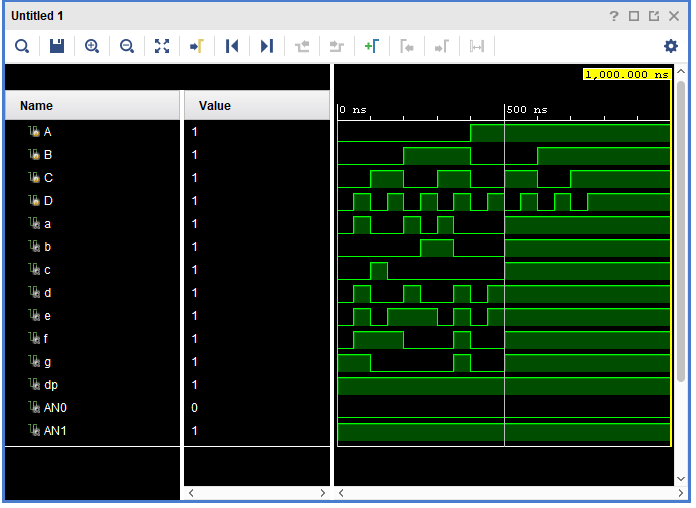
$display ("TC18");

if ({a,b,c,d,e,f,g} != 7'b1111111) $display ("Result is wrong");

end

endmodule

Simulation Wave Form:



7 Segment Display Constraint File:

set\_property PACKAGE\_PIN V17 [get\_ports {D}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {D}]

set\_property PACKAGE\_PIN V16 [get\_ports {C}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {C}]

set\_property PACKAGE\_PIN W16 [get\_ports {B}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {B}]

set\_property PACKAGE\_PIN W17 [get\_ports {A}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {A}]

set\_property PACKAGE\_PIN W7 [get\_ports {a}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {a}]

set\_property PACKAGE\_PIN W6 [get\_ports {b}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {b}]

set\_property PACKAGE\_PIN U8 [get\_ports {c}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {c}]

set\_property PACKAGE\_PIN V8 [get\_ports {d}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {d}]

set\_property PACKAGE\_PIN U5 [get\_ports {e}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {e}]

set\_property PACKAGE\_PIN V5 [get\_ports {f}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {f}]

set\_property PACKAGE\_PIN U7 [get\_ports {g}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {g}]

set\_property PACKAGE\_PIN V7 [get\_ports {dp}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {dp}]

set\_property PACKAGE\_PIN U2 [get\_ports {AN0}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {AN0}]

set\_property PACKAGE\_PIN U4 [get\_ports {AN1}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {AN1}]

set\_property PACKAGE\_PIN V4 [get\_ports {AN2}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {AN2}]

set\_property PACKAGE\_PIN W4 [get\_ports {AN3}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {AN3}]