Assignment 2 Solution

Ngo Van Canh

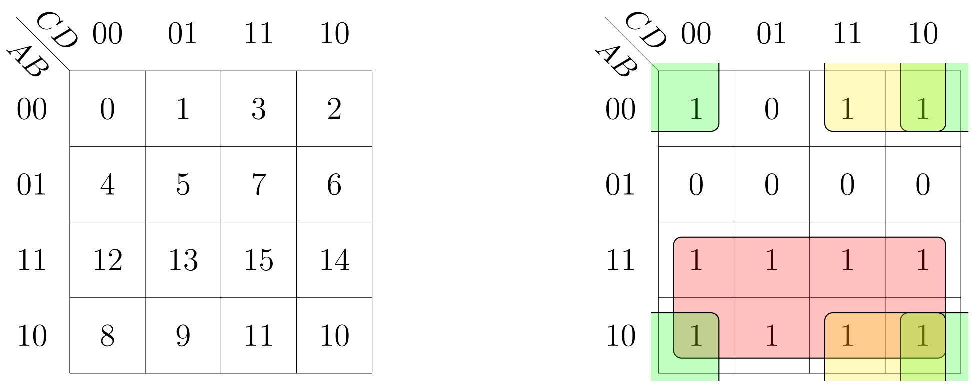
1 Oct 2023

Problem 1 :

$F(A, B, C, D) = \sum m(0, 2, 3, 8, 9, 10, 11, 12, 13, 14, 15)$

	A	B	C	D	F
0	0	0	0	0	1
1	0	0	0	1	0
2	0	0	1	0	1
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	0
8	1	0	0	0	1
9	1	0	0	1	1
10	1	0	1	0	1
11	1	0	1	1	1
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	1

Table 1: Truth Table



(a) Layout (b) Prime implicant

Figure 1: Karnaugh-map

$\Rightarrow F = A + B'D' + B'C$

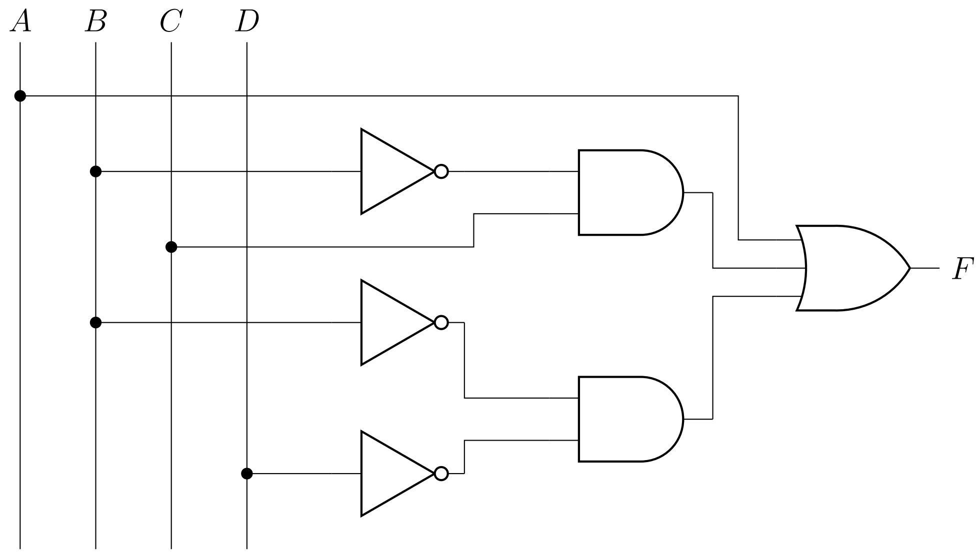


Figure 2: Circuit

```
module Problem1(F,A,B,C,D);
input A,B,C,D;
output F;
wire S1,S2,S3,S4;
not N1(S1,B);
not N2(S2,D);
and A1(S3,S1,C);
and A2(S4,S1,S2);
or O1(F,A,S3,S4);
endmodule
```

Listing 1: Verilog code for Problem 1

Problem 2 : Design a 4-to-1 Multiplexer

$s0$	$s1$	F
0	0	$i0$
0	1	$i1$
1	0	$i2$
1	1	$i3$

Table 2: Truth Table for a 4-to-1 Multiplexer

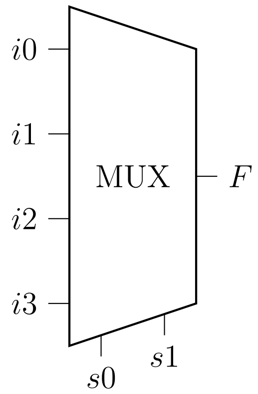


Figure 3: 4-to-1 Multiplexer

$\Rightarrow F = s0's1'i0 + s0s1'i1 + s0's1i2 + s0s1i3$

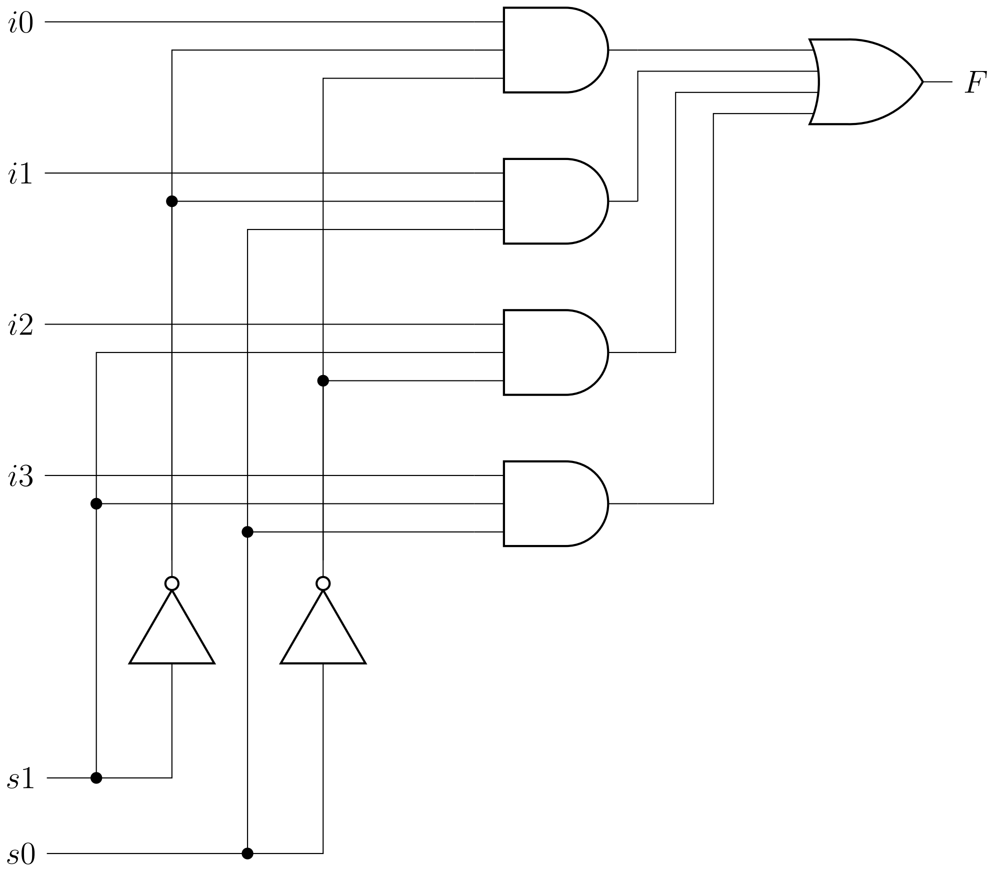


Figure 4: 4-to-1 Multiplexer Circuit

```
module mux4_to_1 (F, i0, i1, i2, i3, s1, s0);
output F;
input i0, i1, i2, i3, s1, s0;
wire s1n, s0n;
wire y0, y1, y2, y3;
not (s1n, s1);
not (s0n, s0);
and (y0, i0, s1n, s0n);
and (y1, i1, s1n, s0);
and (y2, i2, s1, s0n);
and (y3, i3, s1, s0);
or (F, y0, y1, y2, y3);
endmodule
```

Listing 2: Verilog code for 4-to-1 Multiplexer

Problem 3 : Adder Circuit

1.Half Adder

X	Y	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Table 3: Truth Table for Half Adder

$$\Rightarrow \begin{cases} S = X'Y + XY' = X \oplus Y \\ C = XY \end{cases}$$

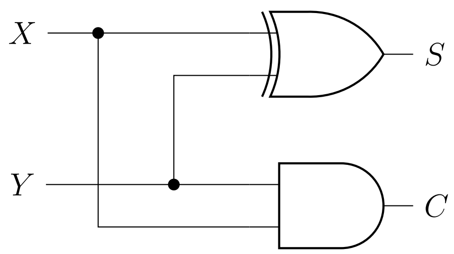


Figure 5: Half Adder Circuit

```
module half_adder (X, Y, S, C);
input X, Y;
output S, C;
xor Xor (S, X, Y);
and And (C, X, Y);
endmodule
```

Listing 3: Verilog code for Half Adder

2.Full Adder

X	Y	Ci	S	Co
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Table 4: Truth Table for Full Adder

Ci \ YX	00	01	11	10
0	0	1	0	1
1	1	0	1	0

(a) $S = Ci \oplus X \oplus Y$

Ci \ YX	00	01	11	10
0	0	0	1	0
1	0	1	1	1

(b) $Co = XY + YCi + XCi$

Figure 6: Karnaugh-map Full Adder

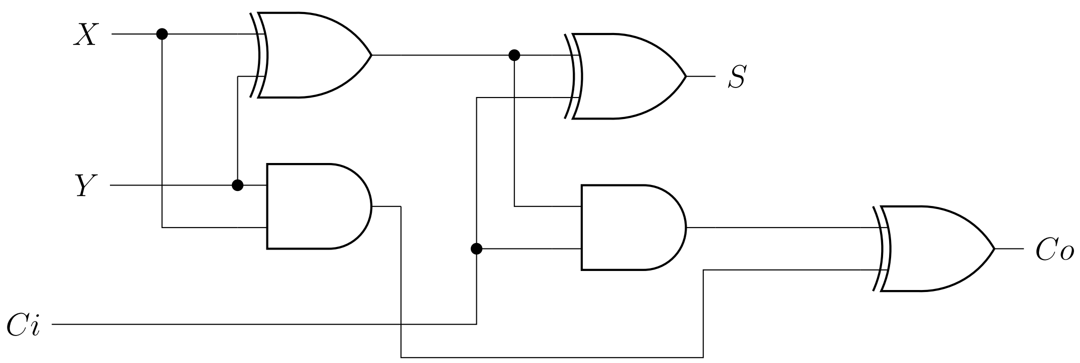


Figure 7: Full Adder Circuit

```
module full_adder (X, Y, Ci, S, Co);
input X, Y, Ci;
output S, Co;
wire w1, w2, w3
xor (w1, X, Y);
and (w1, X, Y);
xor (S,  w1, Ci);
and (w3, w1, Ci);
xor (Co, w3, w2);
endmodule
```

Listing 4: Verilog code for Full Adder

3.Ripple Carry 4-bit Adder

```
module Ripple_Carry_Adder(X, Y, Ci, S, Co);
output [3:0] S;
output Co;
input[3:0] X, Y;
input Ci;
wire w1, w2, w3;
 full_adder fa0(S[0], w1, X[0], Y[0], Ci);
 full_adder fa1(S[1], w2, X[1], Y[1], w1);
 full_adder fa2(S[2], w3, X[2], Y[2], w2);
 full_adder fa3(S[3], Co, X[3], Y[3], w3);
endmodule
```

Listing 5: Verilog code for Ripple Carry Adder

4.Ripple Carry 6-bit Adder

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.