# Lecture Notes for ENEE244, Fall 1998

October 16, 1998

Let's continue doing problems from Chapter 4.

## Problem 4-4 Cont'd

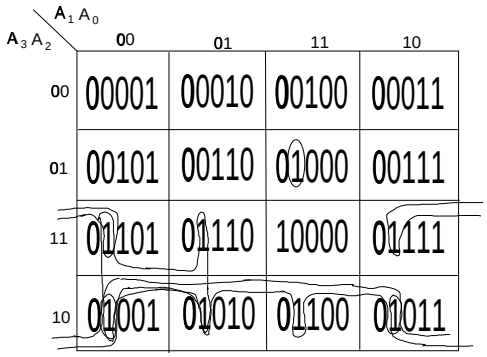
The problem, if you recall, asks us to design a circuit that adds 1 to a four bit binary number. Last time we saw a solution using Half Adders. What if we didn't want to use that particular solution? We can simply draw our Karnaugh Map and proceed from there.

| $A_3A_2$ \ $A_1A_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 00001 | 00010 | 00100 | 00011 |
| 01 | 00101 | 00110 | 01000 | 00111 |
| 11 | 01101 | 01110 | 10000 | 01111 |
| 10 | 01001 | 01010 | 01100 | 01011 |

Let's turn first to $B_4$. From the Karnaugh Map it's easy to see that $B_4$ is equal to one in exactly one place. Therefore:

$$B_4 = A_3A_2A_1A_0$$

How about $B_3$? Turning to the Karnaugh Map:




From there we can write:

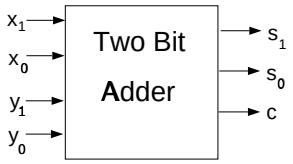
$$B_3 = A_3\bar{A}_1 + A_3\bar{A}_2 + A_3\bar{A}_0 + \bar{A}_3A_2A_1A_0$$

Through the same process that we are all by now familiar with we can get:

$$\begin{aligned} B_2 &= \bar{A}_2A_1A_0 + A_2\bar{A}_1 + A_2\bar{A}_0 \\ B_1 &= \bar{A}_1A_0 + A_1\bar{A}_0 = A_1 \oplus A_0 \\ B_0 &= \bar{A}_0 \end{aligned}$$

# Problem 4-5

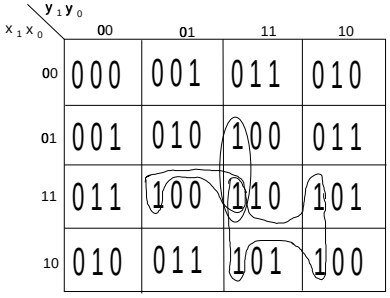
Design a circuit to add two binary numbers $x_1x_0$ and $y_1y_0$ producing the sum $s_1s_0$ and a carry output, c.
From the problem description we write down the block diagram so we have a clear idea of what we're doing:




From the problem description we come up with a Karnaugh Map:

| $x_1x_0$ \ $y_1y_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 000 | 001 | 011 | 010 |
| 01 | 001 | 010 | 100 | 011 |
| 11 | 011 | 100 | 110 | 101 |
| 10 | 010 | 011 | 101 | 100 |

Turning first to the carry bit we circle the appropriate things on the Karnaugh Map to get:




and then we can write:

$$c = x_1y_1 + x_1x_0y_0 + x_0y_1y_0$$

This should make good sense- there will be a carry if both $x_1$ and $y_1$ are 1 (since 1+1 = 0 and a carry out) or if $x_0 + y_0$ produces a carry and either $x_1$ or $y_1$ is a 1.
For $s_1$ and $s_0$ think about what is happening when adding two two bit numbers. The least significant sum bit, $s_0$ depends only on $x_0$ and $y_0$ and will be a one only if either $x_0$ or $y_0$ are one. Therefore we can write:

$$s_0 = x_0 \oplus y_0$$

The most significant bit, $s_1$ should be a one if $x_1$ and $y_1$ are not both one and there wasn't a carry from $x_0 + y_0$, that is $x_0, y_0$ are not both one, or if

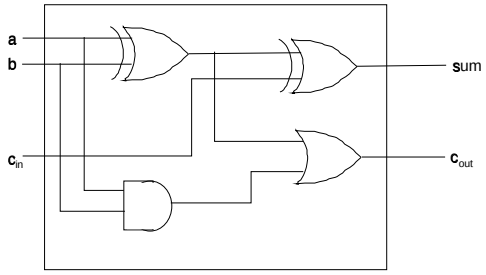
$x_1, y_1$ are both zero and there was a carry from $x_0 + y_0$. Although it might be a little bit complicated to see, all of the above means we can write $s_1$ as:

$$s_1 = x_1 \oplus y_1 \oplus (x_0 y_0)$$

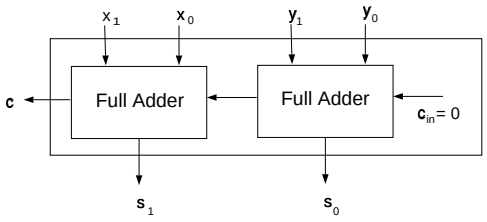
(If you don't see how we got either $s_0$ or $s_1$ don't worry. Turn to the Karnaugh Map and determine them from there).

## Problem 4-6

Problem 4-6 asks us to do the same problem as 4-5 but using two full adders. As you should recall the full adder looks like:

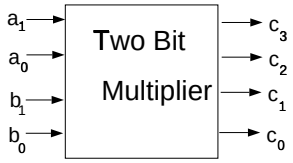



We don't need to worry about the internals but just need to realize that the circuit has three inputs, $a_i, b_i, c_i$ and produces $s_i$ and $c_{i+1}$. To produce a two bit adder we can simply cascade a pair of them together. The only trick is that the first sum doesn't have a signal for the carry input of the first full adder. But this is no problem- since there is no initial carry in we can just set that to 0. The implementation is then:



# Problem 4-7

Multiply two two bit numbers, $a_1a_0$, $b_1b_0$ to produce a four bit number $c_3c_2c_1c_0$ using AND gates and Half Adders.
From the problem description comes first the block diagram:




and then the Karnaugh Map:

| $a_1a_0$ \ $b_1b_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0000 | 0000 | 0000 | 0000 |
| 01 | 0000 | 0001 | 0011 | 0010 |
| 11 | 0000 | 0011 | 1001 | 0110 |
| 10 | 0000 | 0010 | 0110 | 0100 |

We could ignore the suggestion and just proceed as we're used to. Let's not do that though and see if we can't figure out how to do it with the Half Adders and AND gates.
So how does one multiply two numbers?

$$\begin{array}{cccc} & & a_1 & a_0 \\ & & b_1 & b_0 \\ \hline & & a_1b_0 & a_0b_0 \\ + & a_1b_1 & b_1a_0 & 0 \\ \hline c_3 & c_2 & c_1 & c_0 \end{array}$$

So it's easy to see that:

$$c_0 = a_0b_0$$

$$
\begin{aligned}
c_1 &= a_1 b_0 \text{ PLUS } a_0 b_1 \\
c_2 &= a_1 b_1 \text{ PLUS carry out from } c_1 \\
c_3 &= \text{ carry out from } c_2
\end{aligned}
$$

and the circuit looks like: