**Wen-Ying Choy**
**Barry John Chuan**

**EE 200 PRINCIPLES OF DIGITAL DESIGN**
**- Section 104**

# LAB 1: COMBINATIONAL CIRCUITS

*Introduction to the Xilinx Foundation System: Schematic entry and functional simulation*

September 17, 1999

## PRE-LAB

Question 1:

| A | B | C | Sum | Co | Maxterm |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | $M_0 = A + B + C$ |
| 0 | 0 | 1 | 1 | 0 | $M_1 = A + B + C'$ |
| 0 | 1 | 0 | 1 | 0 | $M_2 = A + B' + C$ |
| 0 | 1 | 1 | 0 | 1 | $M_3 = A + B' + C'$ |
| 1 | 0 | 0 | 1 | 0 | $M_4 = A' + B + C$ |
| 1 | 0 | 1 | 0 | 1 | $M_5 = A' + B + C'$ |
| 1 | 1 | 0 | 0 | 1 | $M_6 = A' + B' + C$ |
| 1 | 1 | 1 | 1 | 1 | $M_7 = A' + B' + C'$ |

* where 'X represents the complement of X

Table 1: Truth-table of Full Adder

$S = \Pi M(0, 3, 5, 6)$
$Co = \Pi M(0, 1, 2, 4)$

Question 2:

$S = M_0 . M_3 . M_5 . M_6$

$= (A' + B' + C')( A + B' + C')( A' + B + C')(A' + B' + C)$ ……equation 1

$S' = M_1 . M_2 . M_4 . M_7$

$S = \overline{M_1 . M_2 . M_4 . M_7}$

$= (A + B + C')( A + B' + C)( A' + B + C)(A' + B' + C')$

$= A'B'C + A'BC' + AB'C' + ABC$…….equation 2

<De Morgan's Theorem>

$= A' (B'C + BC') + A(B'C' + BC)$ <Distributive property>

$= A'(B \oplus C) + A(B \oplus C)'$

$= A \oplus B \oplus C$

Co $= M_0 . M_1 . M_2 . M_4$

$= (A + B + C)\ (A + B + C')(\ A + B' + C)(\ A' + B + C)$…….. equation 3

Co' $= \overline{M_3 . M_5 . M_6 . M_7}$

Co $= \overline{M_3 . M_5 . M_6 . M_7}$

$= \overline{(A + B' + C')(A' + B + C')(A' + B' + C)(A' + B' + C')}$

$= A'BC + AB'C + ABC' + ABC$…… equation 4

<De Morgan's Theorem>

$= (A'BC + ABC) + (AB'C + ABC) + (ABC' + ABC)$ <Law: X + X = X>

$= BC(A' + A) + AC(B' + B) + AB\ (C' + C)$ <Distributive, Associative Laws>

$= BC + AC + AB$ <Law: X + X' = 1>

$= A\ (B+C) + CB$ <Distributive Law>

Question 3:

The design process consists of

√ Design Verification
√ Design Implementation
√ Design Entry

Question 4:

Designs can be entered through

√ Schematic Editor
√ Behavioral Description
√ Schematic Editor and Behavior Description

Question 5:

Hierarchical Design Entry - Xilinx supports hierarchical design entry

√ True

Question 6:

Timing simulation does not provide timing information such as delays, race condition, set-up and hold-time violations
√ False

**Exercise**. Draw the logic diagram in your lab notebook for the S and Co function using XOR, AND and OR gates. An XOR gate can have only two inputs

## LAB REPORT

### 1. Goals of the experiment

The main aim of the experiment is to use the Xilinx M1 Foundation CAE system - a development tool that has the ability to create, simulate and implement digital designs in a FPGA or CPLD - to design and simulate the operations of the Full Adder, a simple combinational circuit.

### 2. Theory of operation

The full adder is a logic circuit that adds three bits (the inputs "A" and "B" to be added; and the Carry bit from a previous addition, "C"). The outputs produced are a Sum bit and a Carry-out Bit.

Even though the full adder circuit can be implemented with equation 1 (or equation 2) and equation 3 (or equation 4) using a series of AND, OR and NOT gates (see the pre-lab section for the equations referred to), there are advantages in simplifying the design of the circuit to facilitate implementation. Fewer literals, fewer gates and faster implementation, are but some of the advantages.

Hence, the Sum and Carry out functions (as a product of maxterms) were further simplified in the pre-lab, using De Morgan's theorem and other Boolean simplification techniques. The resulting circuit was entered into the Xilinx system using the Schematic Editor. The schematics of the circuit is attached on the following page. After this stage, a functional simulation was carried out to verify that the circuit was functioning properly. The simulation waveform is also attached.

### 3. Experimental Results

The truth table obtained by analyzing the waveform produced by the functional simulator is given below:

| STAGE | A | B | C | SUM | CARRY | Correspond with outputs of full adder? |
|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 1 | 0 | √ |
| 2 | 0 | 1 | 0 | 1 | 0 | √ |
| 3 | 1 | 1 | 0 | 0 | 1 | √ |
| 4 | 0 | 0 | 1 | 1 | 0 | √ |
| 5 | 1 | 0 | 1 | 0 | 1 | √ |
| 6 | 0 | 1 | 1 | 0 | 1 | √ |
| 7 | 1 | 1 | 1 | 1 | 1 | √ |
| 8 | 0 | 0 | 0 | 0 | 0 | √ |

Table 2: Truth-table of simulated waveform

The logic states listed in the table follow the order in which they were generated by the function simulator. These are marked out in the simulated waveform.

## 4. Discussion of Results and Conclusion

Comparing the generated truth table (table 2) with the truth table of a Full Adder (table 1), we confirmed that the Sum and Carry outputs generated by their respective inputs corresponded with that of a Full Adder.

It can thus be concluded that the Full Adder circuit design, which was based on the derivations and simplifications that made use of Boolean operations, was also implementable. The implementability of the design was tested with the aid of Xilinx, which enabled a quick simulation of the circuit. Since the functional simulation of the design generated a waveform that matched the desired outcome, the design can be deemed as being representative of a real Full Adder circuit.

Report submitted and signed by:

| | |
|---|---|
| Wen-Ying Choy | Barry John Chuan |
| 18 September 1999 | 18 September 1999 |