# ECE 2504: Introduction to Computer Engineering

Homework Assignment 2 (40 points)

**Make a reasonable effort to show your work. Clearly indicate your answers.**

Problem 1 (12 points)

Even though we aren’t trying to prove any of our axioms, a truth table is a useful means for justifying the truth of many axioms. If we can demonstrate that two seemingly distinct expressions give the same value for all combinations of a set of inputs, then the two expressions are actually the same.

For example, demonstrating that the AND function is associative amounts to showing that (AB)C = A(BC):

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | AB | C | (AB)C | A | BC | A(BC) |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

(AB)C = A(BC) for all combinations of A, B, and C. Thus, the AND function exhibits associativity.

Using a truth table in the fashion shown above, demonstrate each of the following:

1. The distributive property A(B + C) = AB + AC

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | B+C | **A(B+C)** | AB | AC | **AB+AC** |
| 0 | 0 | 0 | 0 | **0** | 0 | 0 | **0** |
| 0 | 0 | 1 | 1 | **0** | 0 | 0 | **0** |
| 0 | 1 | 0 | 1 | **0** | 0 | 0 | **0** |
| 0 | 1 | 1 | 1 | **0** | 0 | 0 | **0** |
| 1 | 0 | 0 | 0 | **0** | 0 | 0 | **0** |
| 1 | 0 | 1 | 1 | **1** | 0 | 1 | **1** |
| 1 | 1 | 0 | 1 | **1** | 1 | 0 | **1** |
| 1 | 1 | 1 | 1 | **1** | 1 | 1 | **1** |

Thus A(B+C) = AB+AC

1. The distributive property A + BC = (A + B)(A + C)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | BC | **A+BC** | A+B | A+C | **(A+B)(A+C)** |
| 0 | 0 | 0 | 0 | **0** | 0 | 0 | **0** |
| 0 | 0 | 1 | 0 | **0** | 0 | 1 | **0** |
| 0 | 1 | 0 | 0 | **0** | 1 | 0 | **0** |
| 0 | 1 | 1 | 1 | **1** | 1 | 1 | **1** |
| 1 | 0 | 0 | 0 | **1** | 1 | 1 | **1** |
| 1 | 0 | 1 | 0 | **1** | 1 | 1 | **1** |
| 1 | 1 | 0 | 0 | **1** | 1 | 1 | **1** |
| 1 | 1 | 1 | 1 | **1** | 1 | 1 | **1** |

Thus A+BC = (A+B)(A+C)

Problem 1 (continued)

1. The NAND form of DeMorgan’s Theorem for three variables – that is, show that (ABC)′ = A′ + B′ + C′.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | (ABC) | **(ABC)’** | A’ | B’ | C’ | **(A’+B’+C’)** |
| 0 | 0 | 0 | 0 | **1** | 1 | 1 | 1 | **1** |
| 0 | 0 | 1 | 0 | **1** | 1 | 1 | 0 | **1** |
| 0 | 1 | 0 | 0 | **1** | 1 | 0 | 1 | **1** |
| 0 | 1 | 1 | 0 | **1** | 1 | 0 | 0 | **1** |
| 1 | 0 | 0 | 0 | **1** | 0 | 1 | 1 | **1** |
| 1 | 0 | 1 | 0 | **1** | 0 | 1 | 0 | **1** |
| 1 | 1 | 0 | 0 | **1** | 0 | 0 | 1 | **1** |
| 1 | 1 | 1 | 1 | **0** | 0 | 0 | 0 | **0** |

Thus (ABC)′ = A′ + B′ + C′.

1. The NOR form of DeMorgan’s Theorem for three variables – that is, show that (A + B + C) ′ = A′B′C′.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | A’ | B’ | C’ | (A+B+C) | **(A+B+C)’** | **(A’B’C’)** |
| 0 | 0 | 0 | 1 | 1 | 1 | 0 | **1** | **1** |
| 0 | 0 | 1 | 1 | 1 | 0 | 1 | **0** | **0** |
| 0 | 1 | 0 | 1 | 0 | 1 | 1 | **0** | **0** |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | **0** | **0** |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | **0** | **0** |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | **0** | **0** |
| 1 | 1 | 0 | 0 | 0 | 1 | 1 | **0** | **0** |
| 1 | 1 | 1 | 0 | 0 | 0 | 1 | **0** | **0** |

1. The three-input XOR function gives true output if and only if an odd number of its inputs are true. (Try “building” the three-input XOR from two-input XORs.)
2. The XOR function is associative – that is, show that (A ⊕ B) ⊕ C = A ⊕ (B ⊕ C)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | A+B | (A⊕B) | **(A⊕B) ⊕C** | (B ⊕ C) | **A ⊕ (B ⊕ C)** |
| 0 | 0 | 0 | 0 | 0 | **0** | 0 | **0** |
| 0 | 0 | 1 | 0 | 0 | **1** | 1 | **1** |
| 0 | 1 | 0 | 1 | 1 | **1** | 1 | **1** |
| 0 | 1 | 1 | 1 | 1 | **0** | 0 | **0** |
| 1 | 0 | 0 | 1 | 1 | **1** | 0 | **1** |
| 1 | 0 | 1 | 1 | 1 | **0** | 1 | **0** |
| 1 | 1 | 0 | 1 | 0 | **0** | 1 | **0** |
| 1 | 1 | 1 | 1 | 0 | **1** | 0 | **1** |

## Problem 2 (12 points)

Using Boolean algebra manipulation, simplify the following expressions to their simplest terms:

1. A + AB (This is a Boolean algebra theorem called an absorption theorem.)

A+AB

= A(1+B) == A+AB

=A(1) == x+1 = 1

=A == X(1)= X

1. CD + CD′ (This is an important Boolean algebra theorem; it forms the basis of most logic minimization.)

CD+CD’

= C(D+D’) ==CD+CD’

=C(1) == X+X’ = 1

=C == X(1) = X

1. BC + ABC′ + A′BC′

BC+ABC’+A’BC’

=B(C+AC’+A’C’) == BC+ABC’+A’BC’

=B( (C+A)(C+C’)+(A+C)’) == x+yz = (x+y)(x+z)

(x+y)’=x’y’

=B((C+A)(1)+(C+A)’))

=B((C+A)+(C+A)’)) == X\*1=X

=B(1) == x+x’ = 1

=B == x\*1=x

## Problem 2 (continued)

1. BD + B′C′D

BD+B’C’D

=D(B+B’C’) == x(y+z) xy+xz

=D(B+(B+C)’) == (x+y)’ = x’y’

=D(B(1+(1+C)’)

=D(B(1+(1)’) == x+1 =1

=D(B+B’) == x(y+z) = xy+xz

=D(1) == x+x’ = 1

=D == x\*1 = x

1. AB′D + AC′D + BD

=D(B’A+B+AC’) == x(y+z) xy+xz

=D((B+B’)(B+A)+AC’) == (x+yz) = (x+y)((x+z)

=D(1)(B+A)+(AC’) == (x+x’)=1

=D(B+A +(AC’))

=D(B+A(1+C’)) == x(y+z) xy+xz

=D(B+A(1)) == 1+x’ = 1

=D(B+A)

1. (BC + A′D)(AB′ + C′D′) (Hint: You can apply an approach similar to the multiplication of polynomials to expand this expression before you begin simplifying it.)

(BC + A′D)(AB′ + C′D′)

= BCAB’+BCC’D’+A’DAB’+A’DC’D’ ==x(y+z)=xy+xz

= 0 + 0+0+0 == x\*x’=0

=0 == x+0=x

## Problem 3 (8 points)

Consider the three-variable function F(A,B,C) = AC′ + A′BC:

1. Using DeMorgan’s Theorem, express F′. (I recommend that you express F′ in SOP form.)
2. Using Boolean algebra and the function that you derived in part (a), show that F • F′ = 0 and that F + F′ = 1.

(Hint: Remember the hint from Problem 2(f).)

a)

(A+C)\*(A+B’+C’)

= (A’A + A’B’ + A’C’ + CA +CB’ + CC’)

b)

F\*F’=0

(AC’+A’BC)\* (A’A + A’B’ + A’C’ + CA +CB’ + CC’)

A’A= 0 == x\*x’=0

CC’ == x\*x’=0

=(AC’+A’BC) \* (A’B’ + A’C’ + CA +CB’ + 0 + 0)

=0 == x\*0 = 0

Outputs 0 if any are 0 thus F\*F’=0

b)

F+F’=1

(AC’+A’BC)+ (A’A + A’B’ + A’C’ + CA +CB’ + CC’)

=AC’+A’BC+A’(B’+C’)+C(A+B’)

=AC’+A’(B’+C’+BC)+C(A+B’)

=AC’+A’( (B’+B) (B’+C) + C’) + C(A+B’)

=AC’+A’((1) (B’+C) + C’)) + C(A+B’)

=AC’+A’ ( (C’+C) +B’) + C(A+B’)

=AC’+ A’((1)+B’) + C(A+B’)

=AC’ + A’+B’+C(A+B’)

=(A’+A)(A’+C’) + B’ + C(A+B’)

=(1)(A’+C’)+B’+C(A+B’)

=A’+C’+B’+CA+CB’

=( C’+C)(C’+C’) +A’+B’+CA

=(1) (A’+B’+CA+C’+B’)

=A’+B’+(C’+A)(C’+C)+B’

=A’+B’+(C’+A)(1)+B’

=A’+C’+A+B’

=1+C’+B’

1+x = 1 thus F+F’ =1 as there needs to result in just at least one 1 for an output of 1 to happen.

Problem 4 (8 points)

Consider the three-variable function F(A,B,C) = A′BC + ABC + ABC′.

1. Show the truth table for the function.
2. Using Boolean algebra, simplify the function to its simplest terms.
3. Show a truth table for the simplified function, and use it to demonstrate that the simplified function is equivalent to the original function.

(Hint: Sometimes, simplifying a function using Boolean algebra can involve making an expression less simple before making it simpler.)

a)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | A’ | B’ | C’ | A’BC | ABC | ABC’ | A′BC + ABC + ABC′ |
| 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

b)

A’BC+ABC+ABC’

=B(A’B+AC+AC’) == x(y+z) xy+xz

= B( A’C+ AC + AC’) == x(y+z)=xy+xz

=B(C (A’+A) +AC’) == x(y+z)=xy+xz

=B( C (1) +AC’) == x+x’=1

=B (C +AC’)

= B( (C+A) (C+C’)) == x+yz = (x+y)(x+z)

=B(C+A)

C)

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | A’ | B’ | C’ | A’BC | ABC | ABC’ | **A′BC + ABC + ABC′** | C+A | **B(C+A)** |
| 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | **0** | 0 | **0** |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | **0** | 1 | **0** |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | **0** | 0 | **0** |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | **1** | 1 | **1** |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | **0** | 1 | **0** |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | **0** | 1 | **0** |
| 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | **1** | 1 | **1** |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | **1** | 1 | **1** |