Quiz #1

Name (Last, First):

Student Id #:

Do not start working until instructed to do so.

1. *You must answer in the **space provided** for answers after every question. We will ignore answers written anywhere else in the booklet. **All pages in this booklet must be accounted** for otherwise it will not be graded.*
2. *This quiz is closed book/notes.*
3. *You may not use any electronic device.*

Following table to be filled by course staff only

	Maximum Score	Your Score
TOTAL	20	20

<intentionally blank – you may use this for scratch work>

Recall from Homework #1 where we have the 4 DNA nucleotides of G, C, T, and A. Let us represent them using 4-bit codes, *T[3:0]*, where G=4'b0101, C=4'b1101, T=4'b1111, and A=4'b1011. A logic is used to indicate whether the input t[3:0] corresponds to a nucleotide. The output of this logic is *N*.

(a) Complete the truth table for N.

G = 0101
C = 1101
T = 1111
A = 1011

T[3]	T[2]	T[1]	T[0]	N
1	1	1	1	1
1	1	1	0	0
1	1	0	1	1
1	1	0	0	0
1	0	1	1	1
1	0	1	0	0
1	0	0	1	0
1	0	0	0	0
0	1	1	1	0
0	1	1	0	0
0	1	0	1	1
0	1	0	0	0
0	0	1	1	0
0	0	1	0	0
0	0	0	1	0
0	0	0	0	0

(b) Write the expression for *N* in Fully-Disjunctive Normal Form.

$$N = (T[3] \wedge T[2] \wedge T[1] \wedge T[0]) \vee (T[3] \wedge T[2] \wedge \neg T[1] \wedge T[0]) \vee (T[3] \wedge \neg T[2] \wedge T[1] \wedge T[0]) \vee (\neg T[3] \wedge T[2] \wedge \neg T[1] \wedge T[0])$$

(c) Write the expression for $\neg N$ in Fully-Conjunctive Normal Form

$$\neg N = (\neg T[3] \vee \neg T[2] \vee \neg T[1] \vee \neg T[0]) \wedge (\neg T[3] \vee \neg T[2] \vee T[1] \vee \neg T[0]) \wedge (\neg T[3] \vee T[2] \vee \neg T[1] \vee \neg T[0]) \wedge (T[3] \vee \neg T[2] \vee T[1] \vee \neg T[0])$$

4

(d) Draw the Karnaugh Map of the Truth Table. Circle the prime implicants, indicate which one(s) are essential (if any).

T[1:0] \ T[3:2]	00	01	11	10
00	0	0	0	0
01	0	1	1	0
11	0	0	1	1
10	0	0	0	0

essenti-l

essenti-l

2

(e) Using the K-map in (d) as the minimized 2-level logic, write the sum-of-product expression using the fewest number of terms and literals.

$N = (T[2] \wedge \neg T[1] \wedge T[0]) \vee (T[3] \wedge T[1] \wedge T[0])$

2

(f) Can you do better (reduce the number of literals), by using Boolean properties? If so, write the expression and indicate which property you used.

$N = T[0] \wedge ((T[2] \wedge \neg T[1]) \vee (T[3] \wedge T[1]))$

Property: Distributive

2

(g) Use De-Morgan's theorem and write the complement of the function in (f).

$\neg N = \neg T[0] \vee ((\neg T[2] \vee T[1]) \wedge (\neg T[3] \vee \neg T[1]))$

2

(h) Show a combinational circuit that implements N from (f) (with *T[3:0]* as inputs) using only 2-input NAND gates.