Exercise 1

For each of the following formulas use the DPLL procedure to determine whether ϕ is satisfiable or unsatisfiable. Give a complete trace of the algorithm, showing the simplified formula for each recursive call of the DPLL function. Assume that DPLL selects variables in alphabetical order (i.e. $A, B, C, D, E, \ldots$), and that the splitting rule first attempts the value False (F) and then the value True (T).

(a) $\phi_1 = (A \vee B) \wedge (B \vee C) \wedge (\neg B \vee D) \wedge (\neg A \vee \neg D) \wedge (\neg C \vee E) \wedge (A \vee \neg B \vee \neg D)$

(b) $\phi_2 = (\neg A \vee \neg B \vee \neg C) \wedge (A \vee \neg B) \wedge (A \vee \neg D) \wedge (B \vee \neg E) \wedge (\neg C \vee D) \wedge (C \vee E) \wedge (C \vee \neg E) \wedge (\neg D \vee E)$

Exercise 2

For the two formulas below, use resolution to prove that the formulas are unsatisfiable. To do so, first give a set of clauses Δ that is equivalent to the formula and second, use resolution to prove that it is unsatisfiable. Write the resolution process in the form of a tree for easier readability.

(a) $\phi_1 = (A \vee B \vee C) \wedge (\neg A \vee B \vee C) \wedge (A \vee \neg B \vee C) \wedge (A \vee B \vee \neg C) \wedge (\neg A \vee \neg B \vee C) \wedge (\neg A \vee B \vee \neg C) \wedge (A \vee \neg B \vee \neg C) \wedge (\neg A \vee \neg B \vee \neg C)$

(b) $\phi_2 = (G \vee \neg H) \wedge (\neg G \vee H) \wedge (A \rightarrow B) \wedge (\neg B \vee C) \wedge (A \vee D) \wedge (\neg C \vee \neg A \vee \neg E) \wedge (\neg D \vee A) \wedge F \wedge (F \leftrightarrow E)$

Exercise 3

Transform the following formulas to CNF. To do so, follow the steps from the lecture (Chapter 9 slide 21) and give the intermediate results (you may skip those steps where there is nothing to do for the formula at hand). Simplify the resulting formulas where possible.

(a) $(P \rightarrow Q) \leftrightarrow (P \rightarrow R)$

(b) $\neg(P \leftrightarrow Q) \vee (Q \rightarrow R)$

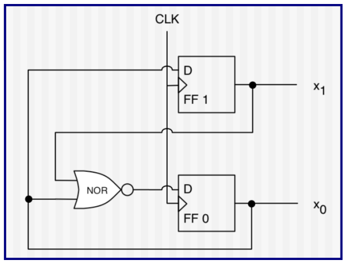
(c) $(P \rightarrow R) \wedge (Q \rightarrow R) \wedge \neg(\neg Q \wedge (\neg R \vee P))$

Exercise 4

Consider the formulas in Exercise 3. Are they satisfiable? Draw the truth table and give the number of interpretations that satisfy the formula.

Exercise 5

Encode the example in the slides of Chapter 7 as a logical formula to verify that if the counter is initialized in the range 0-2, it cannot transition to the value 3.



- Counter, repeatedly from $c = 0$ to $c = 2$.
- 2 bits x_1 and x_0; $c = 2 * x_1 + x_0$.
- (“FF” Flip-Flop, “D” Data IN, “CLK” Clock)

$\rightarrow$ The circuit simply repeats the operations: $x_0 \leftarrow NOR(x_0, x_1) = \neg(x_0 \vee x_1)$; and $x_1 \leftarrow x_0$. The clock is there so that both operations happen simultaneusly, so x_0 and x_1 are updated at the same time.

Hint: you need to refer to two different states of the counter (before and after an operation), so you’ll need different propositions for those.

Hint: If you want to prove that given your knowledge encoded as a formula ϕ_K, some statement ϕ_S is true, then you need to encode the formula $\phi = \phi_K \wedge \neg\phi_S$. If ϕ is unsatisfiable, then it means that if we assume ϕ_K, ϕ_S must be true (as $\neg\phi_S$ leads to contradiction).