Modeling Electrical Circuits

CS 6115 Small Project Proposal

Brittany Nkounkou and Yao Wang

For our project, we propose a Coq-certified software system that models electrical circuits. The design and implementation of the modeling system includes the definition of a circuit and functions on circuits. Various properties about circuits and the defined functions on them are then proved.

Circuit Definition

- Circuits are defined dependently, where the number of (ordered) inputs and outputs are included in a circuit's type, i.e. circuit: nat -> nat -> Type.
- Circuits are defined inductively as follows:
    - Various nullary constructors model power, ground and basic indivisible logic gate components:
        - HIGH: circuit 0 1
        - LOW: circuit 0 1
        - WIRE: circuit 1 1
        - NOT: circuit 1 1
        - AND: circuit 2 1
        - OR: circuit 2 1
        - XOR: circuit 2 1
        - NAND: circuit 2 1
        - 3AND: circuit 3 1
        - ...
    - Two binary constructors model compositional circuits (circuits in series) and parallel circuits:
        - comp: circuit m n -> circuit n o -> circuit m o
        - par: circuit m n -> circuit o p -> circuit (m+o) (n+p)

Circuit Functions

We define three functions on the circuit type.

- Area function: circuit m n -> nat. The area function takes a circuit and outputs a natural number which denotes the area of the circuit. We define area for the compositional and parallel constructors as follows:
    - Area(comp A B) = Area(A) + Area(B)
    - Area(par A B) = Area(A) + Area(B)
- Delay function: circuit m n -> nat. The delay function takes a circuit and outputs a natural number which denotes the overall delay of the circuit. We define delay for the compositional and parallel constructors as follows:
    - Delay(comp A B) = Delay(A) + Delay(B)

    - Delay(par A B) = max(Delay(A), Delay(B))
- Behavior function: circuit m n -> ((bool*) -> (bool*)). The behavior function takes a circuit and outputs a function from boolean vectors to boolean vectors, where the lengths are respective to the number of inputs and outputs of the circuit. This returned function defines the behavior of the circuit. For example:
    - Behavior(HIGH) = [true]
    - Behavior(LOW) = [false]

Properties to Prove

With circuits and circuit functions defined, various properties about them are proved, including the following. (For brevity, the capital letters below represent circuit objects)

- Assuming A's # of inputs = B's # of outputs and A's # of outputs = B's # of inputs
    - Area(comp A B) = Area(comp B A)
    - Delay(comp A B) = Delay(comp B A)
- Assuming A_i, i = {0, 1, …, n} are n circuits. set X = {x_0, x_1, …, x_n} = {0, 1, …, n} (but can be in any order). The following property holds:
    - Area (par A_0 (par A_1…)) = Area(par A_{X0} (par A_{X1}…))
    - Delay (par A_0 (par A_1…)) = Delay(par A_{X0} (par A_{X1}…))
- par(comp A B) (comp C D) ≡ comp (par A C) (par B D) where ≡ denotes the equality of area, delay and behavior.
- Given a specific behavior specification, prove some circuit is an implementation with the minimal delay. (e.g. NAND has smaller delay than (comp AND NOT))
- Given a specific behavior specification, demonstrate a tradeoff between area and delay in different circuit designs. For example, a 2-bit adder with a ripple carry design has lower area higher delay, while one with a carry look-ahead design has higher area lower delay.