p7 of 2
MJCG

Specification and Verification II 2001

SV2.1: Solution Notes

Describe the difference between the simple and threshold switching models of transistors. [4 marks]

> The simple switch model identifies boolean truthvalues with 'Hi' and 'Lo' signal values. Switches are defined by.
>
> ```
> Ntran(g,a,b) = (g=T) ==> (a=b)
> Ptran(g,a,b) = (g=F) ==> (a=b)
> ```
>
> The threshold (difference) switch model uses a non-boolean values 'Hi' and 'Lo' . Switches are defined by.
>
> ```
> Ntran(g,a,b) = (g=Hi) ==> ((a=Lo) = (b=Lo))
> Ptran(g,a,b) = (g=Lo) ==> ((a=Hi) = (b=Hi))
> ```

Give and explain an example of a circuit that illustrates when the threshold switching model is superior to the simple switching model. [4 marks]

> A simple example is an Ntran with source and gate both connected to power. In the simple switch model this outputs a 'full strength' `T` on the drain. The threshold model only entails the weaker requirement that the drain not be `Lo`. Since 'not `Lo`' is not necessarily `Hi`, this better reflects the fact that there is a threshold drop. There are more convincing examples, where the threshold model can motivate additional pass transistors that have no behavioural effect visible in the simple switch model, but are necessary for the correct behaviour in the threshold model. An XOR-gate example was given in the lectures.

Describe how combinational devices can be modelled as zero-delay sequential devices. When is this appropriate? [4 marks]

> A combinational device
>
> ```
> D(i,o) = (o = f i)
> ```
>
> can be modelled a a zero-delay sequential device by defining
>
> ```
> D(i,o) = ∀t. (o t = f(i t))
> ```

This is necessary when sequential components, e.g. registers, are connected to combinational components, e.g. adders, and so, for well-typing, all signal variable have to be functions from time to signal values.

Write down a CTL formula expressing the property: Ack *is true on all paths sometime between 2 units and 5 units of time later.* [4 marks]

$$\mathbf{AX}(\mathbf{AX}(\mathtt{Ack} \vee \mathbf{AX}(\mathtt{Ack} \vee \mathbf{AX}(\mathtt{Ack} \vee \mathbf{AX}\,\mathtt{Ack}))))$$

Describe how an edge-triggered D-type register can be abstracted to a unit delay. Give the formula in higher-order logic that expresses the abstraction. [4 marks]

And edge-triggered D-type abstracts on rising edges (assuming a posedge-triggered device) to a unit delay. This means that if the sequence of values at sucessive rising edges is observed, then the result is a unit delay. The formula expressing the abstraction is shown below:

$$\vdash \mathsf{Inf}(\mathsf{Rise}\ ck) \Rightarrow (\mathsf{Dtype}(ck, d, q) \Rightarrow \mathsf{Del}(d\ \mathsf{when}\ (\mathsf{Rise}\ ck), q\ \mathsf{when}\ (\mathsf{Rise}\ ck)))$$

Here Dtype and Del are the predicates representing a D-type and a unit-delay, respectively. Rise ck is the sequence of truth-values, that is true on rising edges of ck, when selects from its left agument when the right argument is true, and Inf f is true if f is true infinitely often.