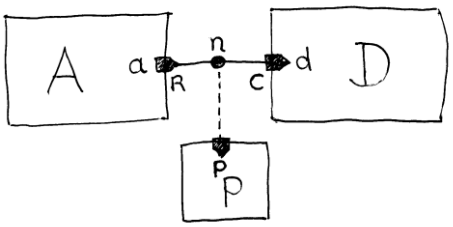
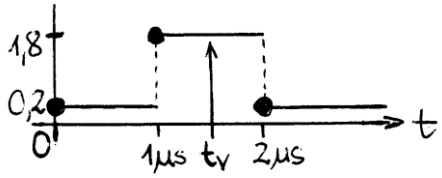
Example:

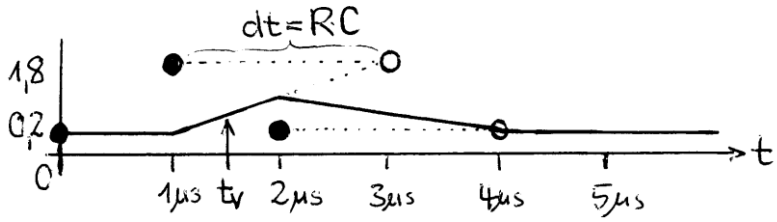
You are having an “analog” real-value driver “A” with a driving-strength “R”,
that drives a discrete sequence of real values through its port “a” to a net “n”.
And you are attaching a digital logic receiver “D” with a loading-strength “C” to this net “n”
via its port “d”.
A verification engineer later binds an assertion “P” to this net “n”, in order to read the value at arbitrary time points via the assertion´s port “p”.
The schematic for all of that:



Now lets assume an “analog” real-value sequence (black dots), that shall be driven by “A”, as pictured below and lets not care about any delay due to driving- or loading-strengths, so the resolved value (black line) of the net “n” shall be:



If the assertion “P” asks for a value at time t_v=1,5us, it will receive the value 1,8 or evtl. 0,2 because of a delay or some interpolated value like (1,8+0,2)/2=1, which can be **deadly wrong**. What actually happens in the device under verification is this (assume RC=2us):



So a reasonable value for the assertion would have been 0,6! The signal would barely have reached the 0/1-threshold because the “analog” 1,8V-pulse was too short or the driver too weak or the load too strong!

In practice it is very important to verify critical effects due to weak/slow/highZ/standby drivers or due to strong/too-heavy/variable loads. We can easily accomplish this, if we allow for a continuous-time function that provides the actual state of the particular net and that is derived from the resolved driving- and loading strengths.
Again for clarification: that continuous-time function does **not** require additional samples or denser time steps! It is just a term like “a*t + b”, that is evaluated, whenever a probe/assertion/receiver asks for the actual value.

Achim Bauer, exl-modeling, Tel.: 0049 8161 807 484