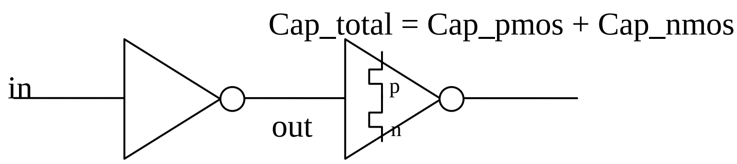
Characterizing CCS Timing Model of Memory IP by Using CharFlo-Memory!

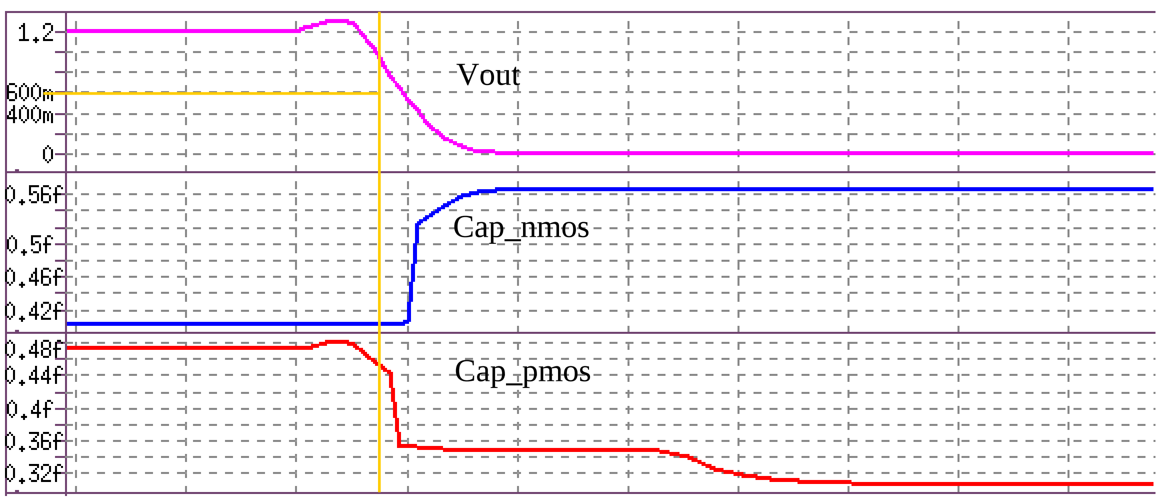
Legend Design Technology, Inc.
2905 Stender Way, Suite 50, Santa Clara, CA 95054
Phone: 408-748-8888   Web: www.LegendDesign.com

Circuit under characterization:

A test circuit for the discussion is introduced first for the comparison between CCST receiver model and NLDM capacitance. Here is an inverter with inverter as loading:

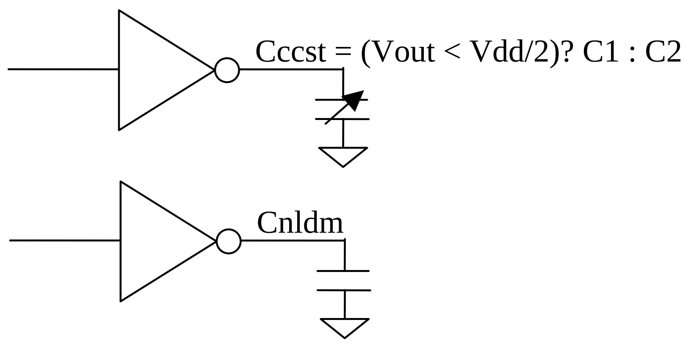


Vout, Cap_nmos and Cap_pmos waveforms:

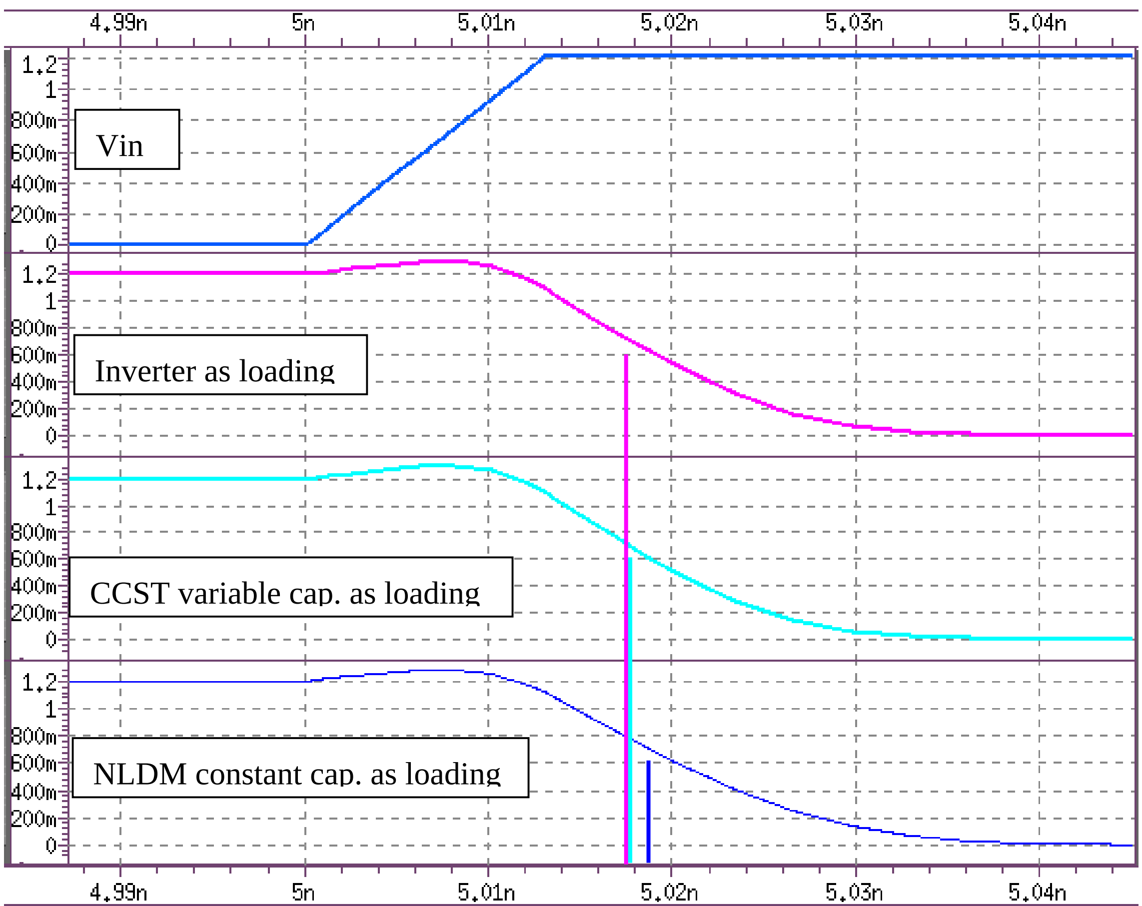


Capacitance of the loading inverter at output node (Cap_total) is a variable because the gate capacitance of the NMOS (Cap_nmos) and PMOS (Cap_pmos) is changing. A critical point for the capacitance change is usually close to Vdd/2 (600m volt in this case).

CCST variable load and NLDM constant load:



CCST receiver model consists of 2 values, C1 for voltage before half Vdd, or time before "reference_time" which is half the input slew, and C2 for the one after. The comparison of output waveform with inverter as loading, CCST variable capacitance as loading and NLDM constant capacitance as loading is shown in the following figure.

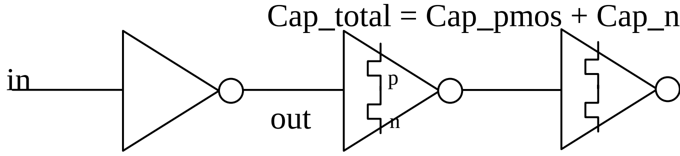


Legend Design Technology, Inc.
2905 Stender Way, Suite 50, Santa Clara, CA 95054
Phone: 408-748-8888 Web: www.LegendDesign.com

NLDM value use only one value for both rising and falling transition, while CCST has different value for rising and falling input transition, and can result in a much accurate delay calculation result. In general, CCST falls within 2% of the Spice's result and NLDM is about 7% away on the other hand.

CharFlo-Memory! CCST receiver model verification :

1. Include second order effect : CCST C1/C2 are not just the summation of Cap_nmos and Cap_pmos. They include first and second order effective capacitances that can be seen on the input nodes. Therefore, the CCB (channel connected block) need to be included for accurate CCST C1/C2 characterization.



2. CCST verification with NLDM : When there's already a "trustable" NLDM capacitance value in the liberty file. CharFlo-Memory!'s CCST checking report can be used to highly the ones that are out of the coverage of CCST's C1/C2 rising/falling. The checks that Charflo-Memory! runs are:
   - C1/C2 sign check : Error if either C1 or C2 is negative
   - NLDM vs. min(C1,C2) check : Error if NLDM value is smaller than min(C1_rise,C2_rise) AND smaller than min(C1_fall, C2_fall). It is a warning if it's smaller than one of them.
   - NLDM vs. max(C1,C2) check : Error if NLDM value is larger than max(C1_rise,C2_rise) AND larger than max(C1_fall, C2_fall). It is a warning if it's larger than one of them.

Example of CCST verification result:

For a case with an input pin CENB whose NLDM value equals 0.003, the old liberty file will be like this:

```
pin (CENB) {
```

```
    direction : input;
    capacitance : 0.003;
    timing() {
      .......
    }
  }
```

With the CCST receiver model written back by CharFlo-Memory!, the new liberty file is:

```
  pin (CENB) {
    direction : input;
    capacitance : 0.0005598; $ new NLDM value
    timing() {
    .......
    receiver_capacitance () {
      receiver_capacitance1_rise(ccst_receiver) {
        index_1 ("5.000e-03, 1.000e+00");
        values ( \
        "4.968000e-04, 6.834000e-04");
      }
      receiver_capacitance2_rise(ccst_receiver) {
        index_1 ("5.000e-03, 1.000e+00");
        values ( \
        "4.706000e-04, 6.087000e-04");
      }
      receiver_capacitance1_fall(ccst_receiver) {
        index_1 ("5.000e-03, 1.000e+00");
        values ( \
        "3.629000e-04, 4.162000e-04");
      }
      receiver_capacitance2_fall(ccst_receiver) {
        index_1 ("5.000e-03, 1.000e+00");
        values ( \
        "4.769000e-04, 8.139000e-04");
      }
      /* CCSTREC vs. NLDM check :
       * Info: CCST equivalent NLDM
       *      cap(min,avg,max) = (4.5180e-04, 5.4117e-04, 6.3055e-04)
       * cap_rise(min,avg,max) = (4.8370e-04, 5.6488e-04, 6.4605e-04)
       * cap_fall(min,avg,max) = (4.1990e-04, 5.1747e-04, 6.1505e-04)
       * Sign check ................ [pass]
       * Error: old_NLDM 0.003 > max(0.00063055, 0.00064605, 0.00061505)
       * Warning: old_NLDM is not accurate for rising edge
       * Warning: old_NLDM is not accurate for falling edge
       * old NLDM Range check ...... [fail]
       * new NLDM Range check ...... [pass]
       */
    }
  }
```

The CCST receiver model sections :

```
receiver_capacitance1_rise,
receiver_capacitance2_rise,
receiver_capacitance1_fall,
receiver_capacitance2_fall
```

state the C1 and C2 for rising edge and C1 and C2 for falling edge respectively. In addition a new NLDM value characterized by CharFlo-Memory! is also written back to the new liberty file. The CCST versus NLDM checking report is placed behind the CCST receiver model. In this case, the old NLDM capacitance value is larger than the maximum value of C1/C2 rising/falling from CCST and thus is reported as error. All the checks are executed on the new NLDM value from CharFlo-Memory! too. Most of the time, the checks on new NLDM value will pass, but there might still some warning or even error for some extreme cases like the ones abnormal P/N ratio.

Summary:

CharFlo-Memory! can automatically cut the necessary circuits out of the original might-be-large netlist to get a pretty small circuit for CCST receiver model characterization while keeping the secondary effects intact to get the most accurate result for CCST model. Nonetheless, CharFlo-Memory! will run a series of calculation and verification to justify the result for both CCST and NLDM capacitance to ensure the accuracy of the final liberty file generated.

Legend Design Technology, Inc.

2905 Stender Way, Suite 50, Santa Clara, CA 95054

Phone: 408-748-8888   Web: www.LegendDesign.com