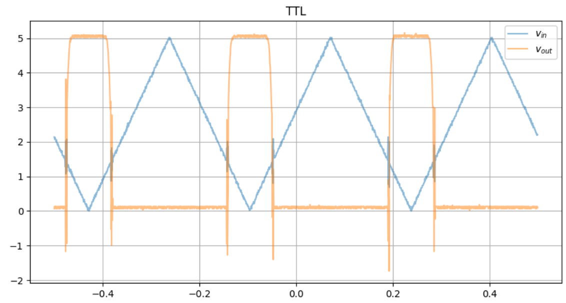
Digital Systems Lab 1 Report

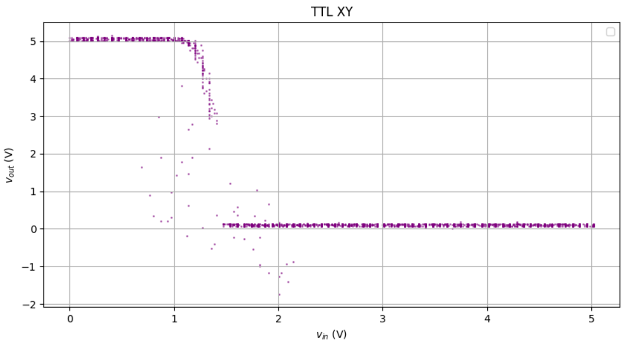
Ming Gong (mg4264), Xuanyi Wu (xw3036)

Q1 TTL Transfer Characteristic

DC Transfer Function

Below is the raw XY plot we gathered for the DC transfer function:

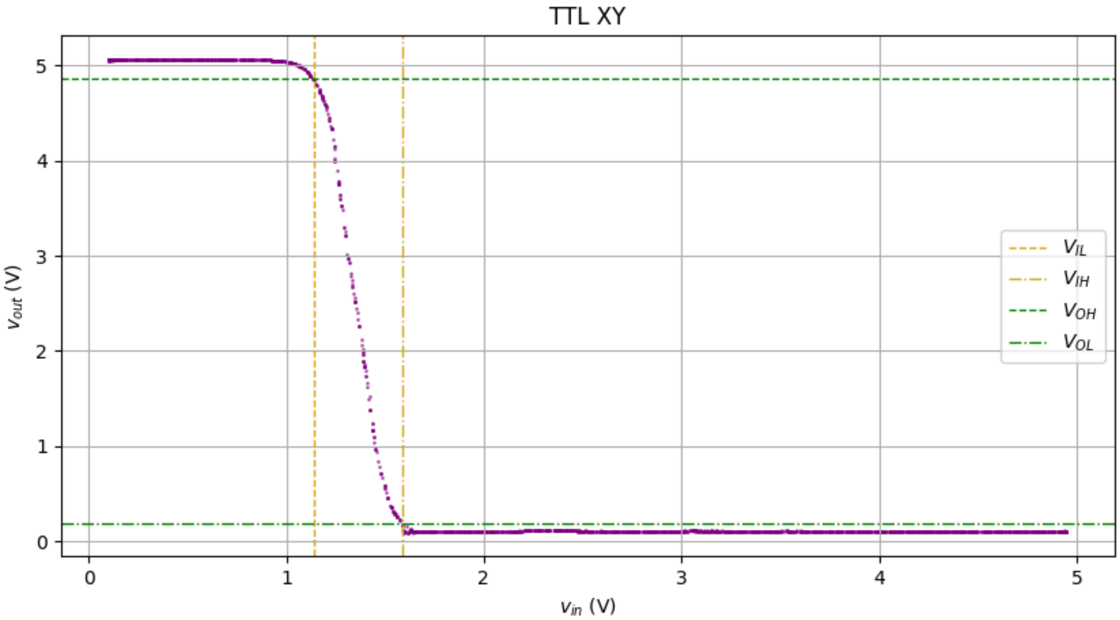




We tuned the frequency to try to obtain the cleanest result, but the output is still extremely noisy, especially near transitions.

To measure the margins, we smoothened the data via KNN – averaging a data point's K nearest neighbors. We found that K = 20 yields a smooth result without distorting the graph too much.

Below is the smoothened XY plot, with the cursors measuring the margins:



Noise Margins

The XY plot is noisy due to the noisy output voltage near transitions. We used cursors to determine where the derivative of the curve is approximately -1, and the points are shown in the oscilloscope shot above at the intersection of the xy cursors. We yield

$V_{IH} = 1.59\,V$

$V_{IL} = 1.14\,V$

$V_{OH} = 4.86\,V$

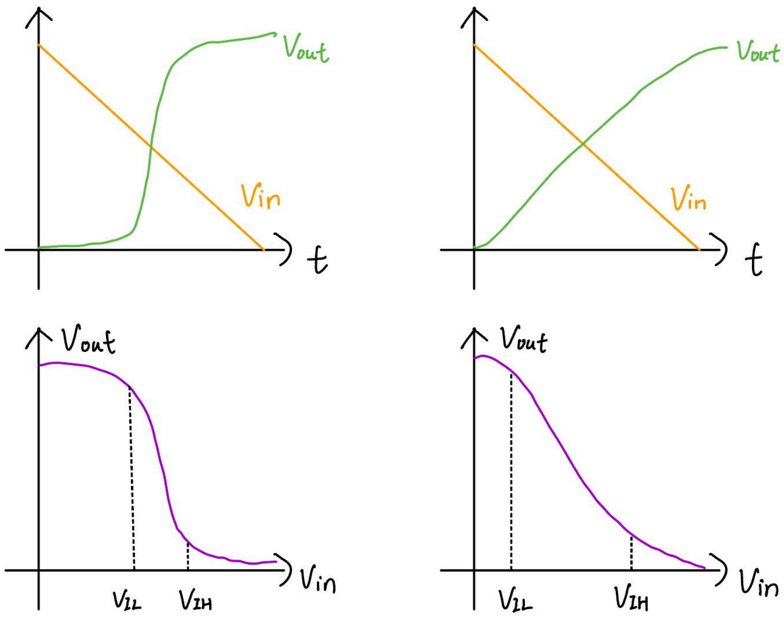
$V_{OL} = 0.18\,V$

$NM_H = V_{OH} - V_{IH} = 3.27\,V$

$NM_L = V_{IL} - V_{OL} = 0.96\,V$

Capacitance Load

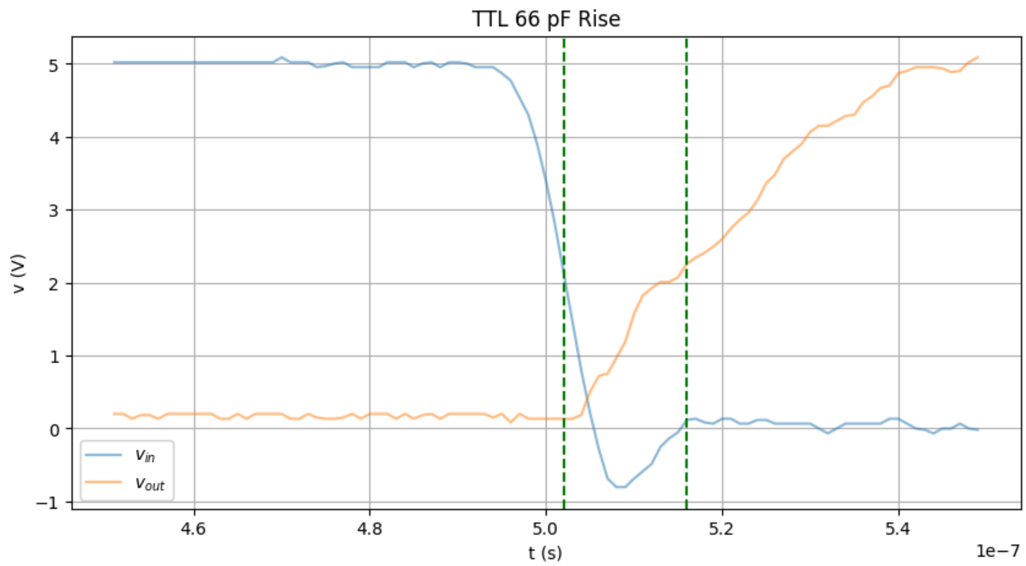
Output capacitance load will slow down the gate response. The XY transfer plot will be less steep. We'll see a higher V_{IH} and a lower V_{IL}. V_{OH} and V_{OL} will also change, but not as much because they are already close to the supply voltages. The combined effect will decrease both the Low Noise Margin and the High Noise Margin



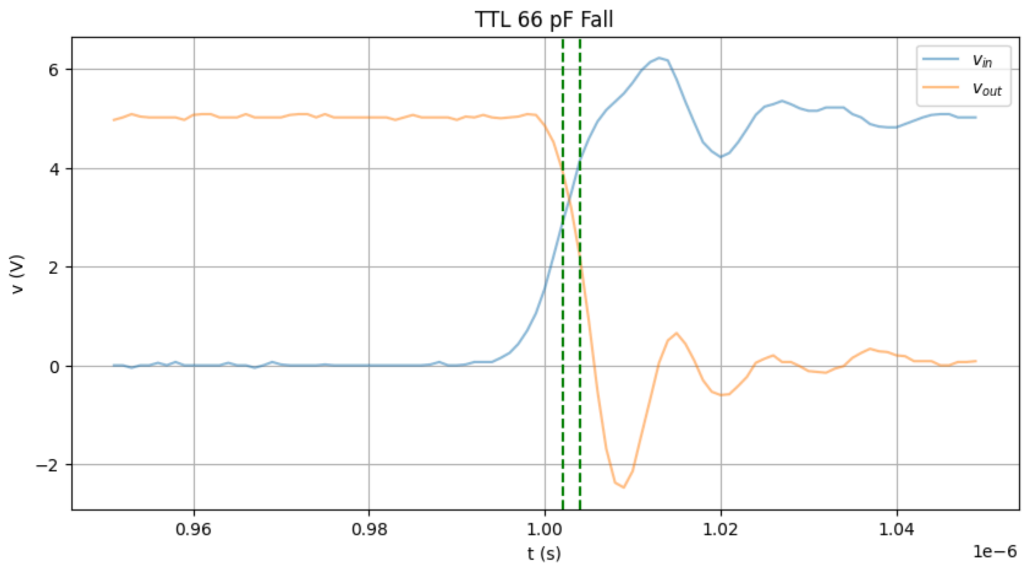
Q2 TTL Propagation Delay

We measured the time it takes between the input and output signal to reach the midpoint of VDD and ground.

66 pF Output Load



$t_{pLH} = 516\,ns - 502\,ns = 14\,ns.$

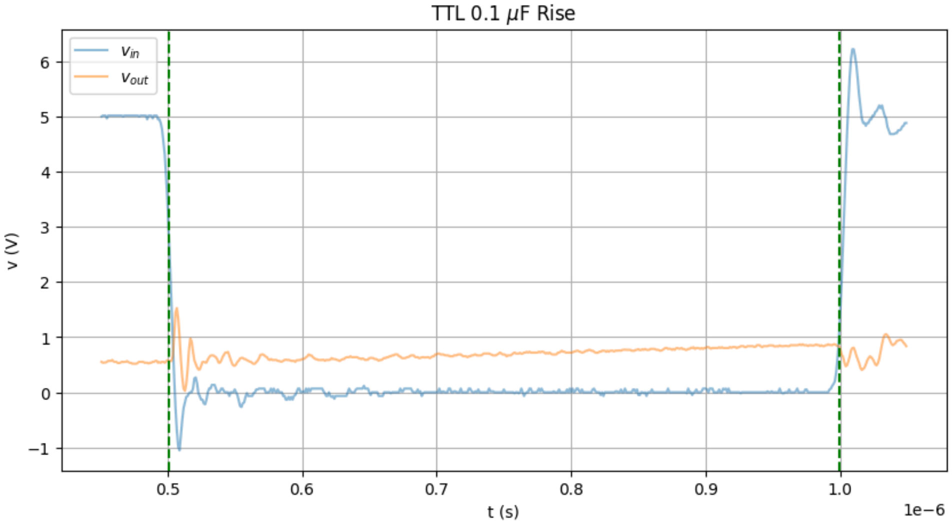


$t_{pHL} = 1004\,ns - 1002\,ns = 2\,ns.$

At 66 pF, $t_{pLH} = 14\,ns$, $t_{pHL} = 2\,ns$

0.1 uF Output Load

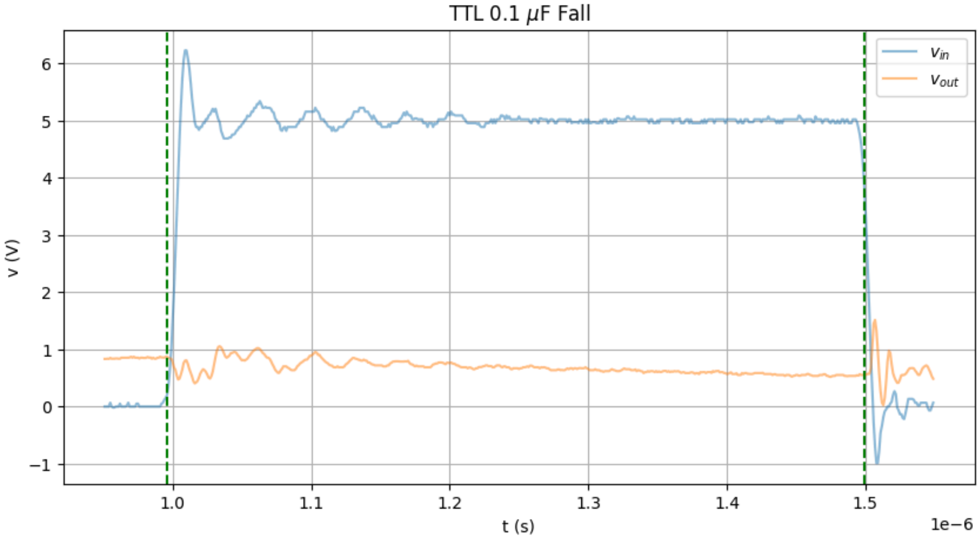
For 0.1 uF, the propagation delays are so large that the output can't follow the 1 MHz input signal. The output does not have enough time to change. The output curve is approximately linear, so we can calculate the gate's slew rate in V/s. Then, we can interpolate the rise and fall times from the slew rate and voltage gap



$slew\ rate\ = \frac{0.837\ V - 0.569V}{999\ ns - 501\ ns} = 538\ kV/s$

We need the output to change from 0 V to 2.5 V (VCC/2)

$t_{pLH} \approx \frac{2.5\ V - 0\ V}{538\ kV/s} = 4.22\ \mu s$



Same idea: $slew\ rate\ = \frac{0.870\ V - 0.552V}{996\ ns - 1499\ ns} =-\ 632\ kV/s$

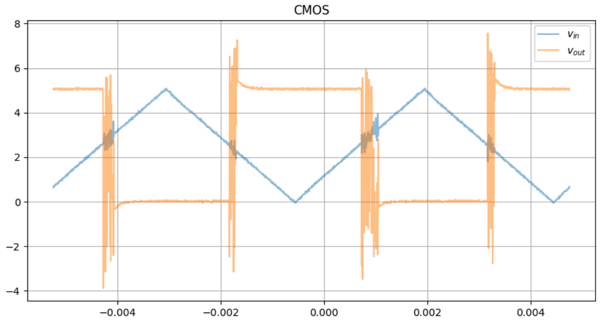
We need the output to change from 5 V to 2.5 V.

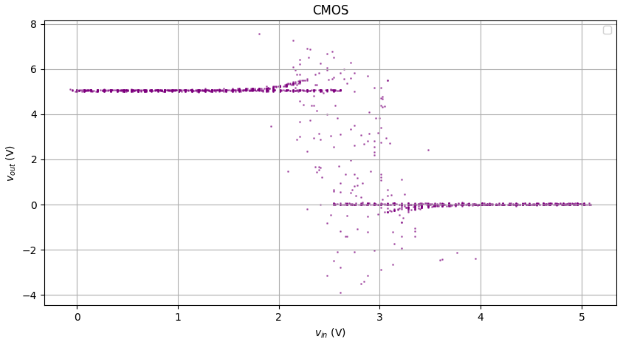
$t_{pHL} \approx \frac{2.5\ V - 5\ V}{-632\ kV/s} = 3.95\ \mu s$

At 0.1 uF, $t_{pLH} = 4.22\ \mu s$, $t_{pHL} = 3.95\ \mu s$

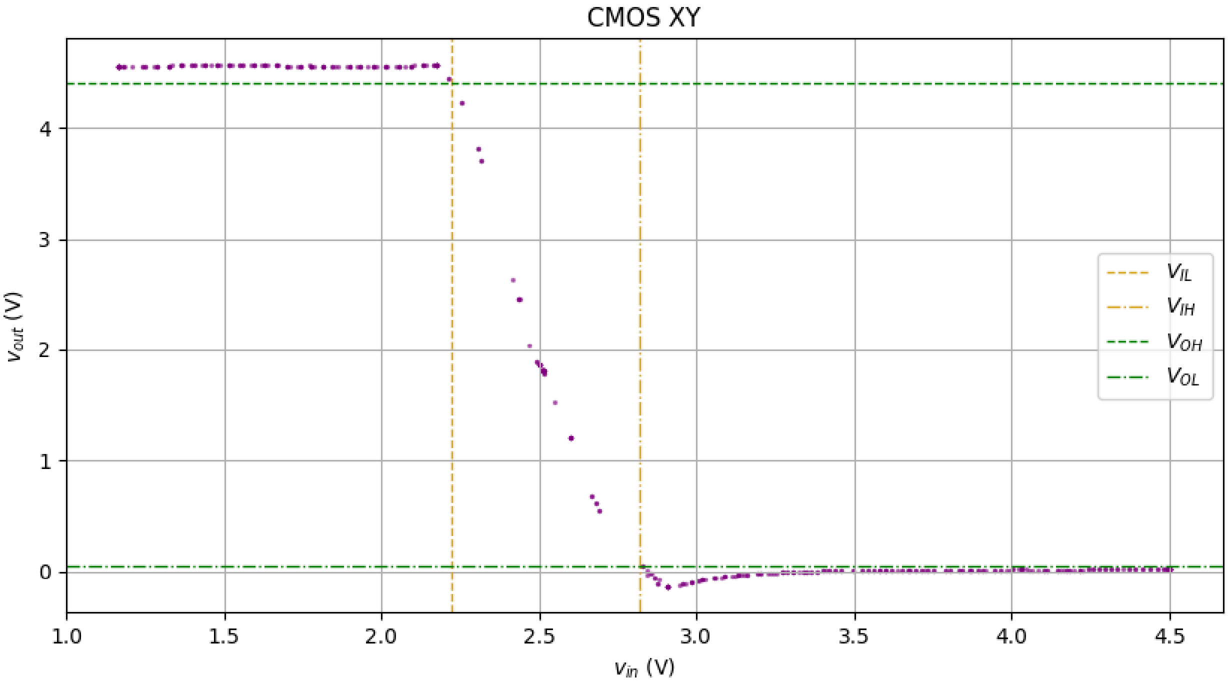
Q3 CMOS DC Transfer Function

Again, the DC transfer curve is extremely messy. Also, the rise and fall curves are asymmetrical. We took one set of the rising portions and smoothed them via KNN to obtain a clean XY plot.





Below is the smoothened XY plot, with cursors for measurement:



Using a similar method as in Q1, we used cursors to approximate where the derivative is about -1. We yield

$V_{IH} = 2.82\,V$

$V_{IL} = 2.22\,V$

$V_{OH} = 4.40\,V$

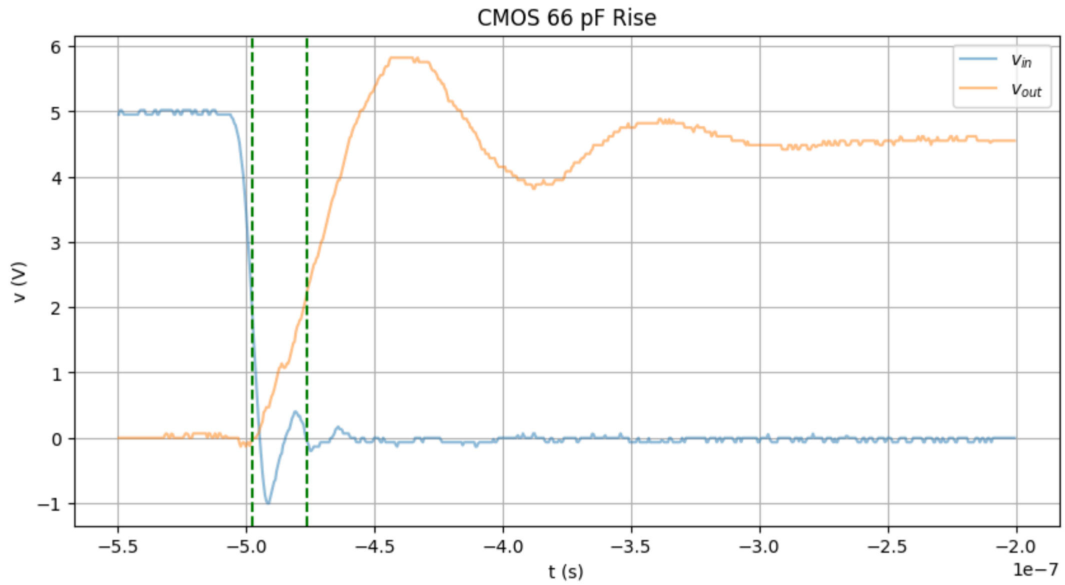
$V_{OL} = 0.05\,V$

$NM_H = V_{OH} - V_{IH} = 1.58\,V$

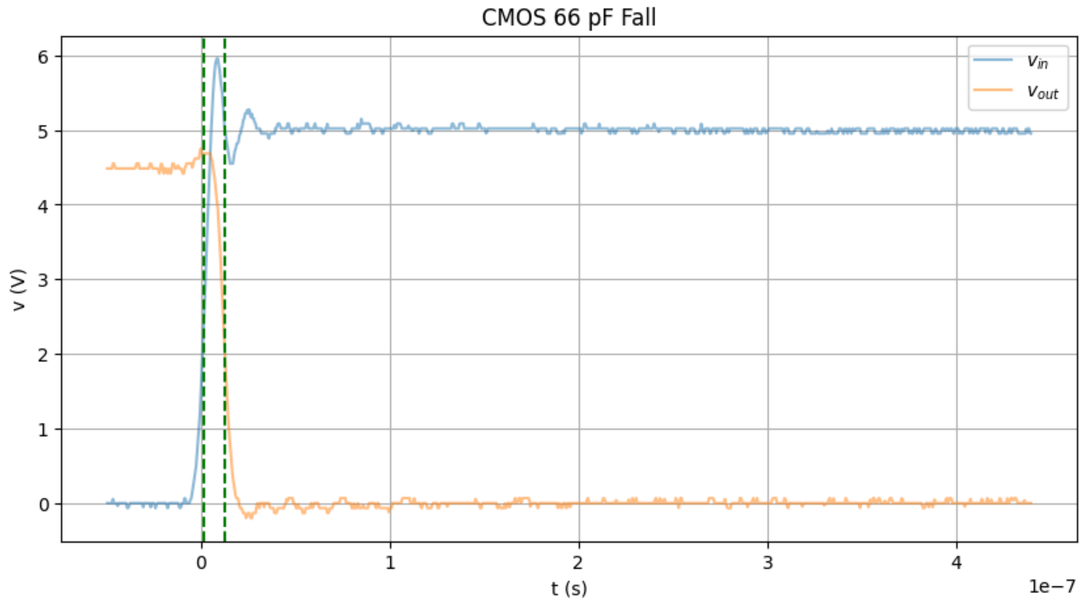
$NM_L = V_{OL} - V_{IL} = 2.17\,V$

Q4 CMOS Propagation Delay

66 pF Output Load



$t_{pLH} =- 476\, ns\ - (- 497.5\, ns)\ =\ 21.5\, ns$

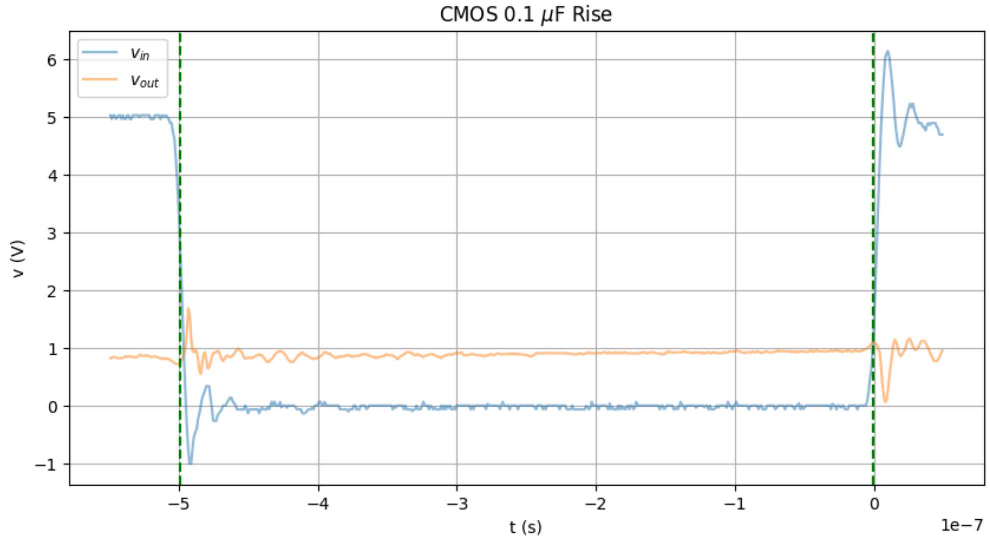


$t_{pHL}\ = 12.5\, ns\ - 1.5\, ns\ =\ 11.0\, ns$

For 66 pF output load, $t_{pLH}\ = 21.5\, ns$, $t_{pHL}\ = 11.0\, ns$

0.1uF Output Load

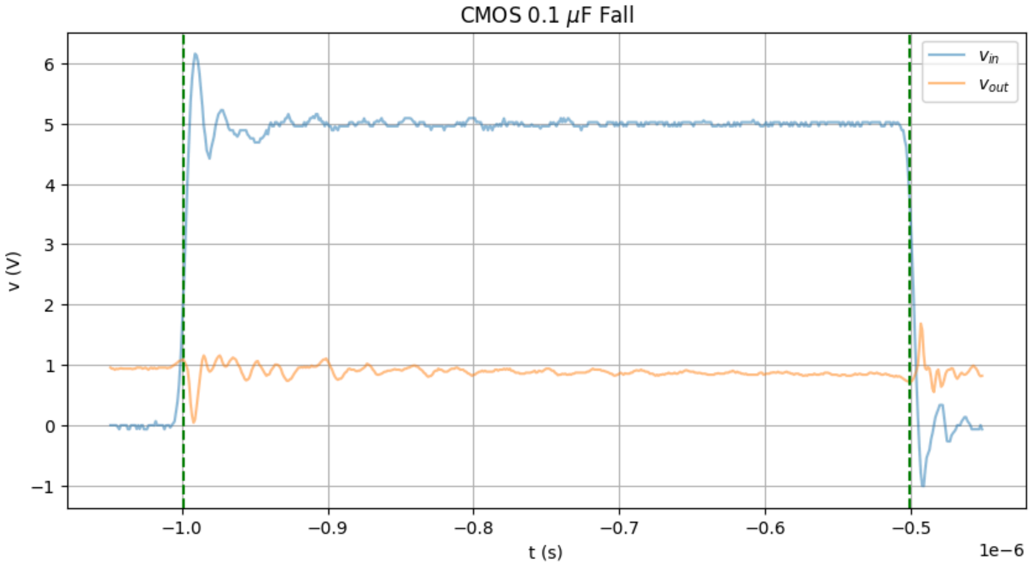
We used the same method to interpolate. Note that VDD is 4.5 V for CMOS.



$slew\ rate\ = \frac{1.087\ V-0.736V}{-1\ ns+499\ ns} = 701\ kV/s$

We need the output to change from 0 V to 2.25 V (VCC/2)

$t_{pLH} \approx \frac{2.25\ V-0\ V}{701\ kV/s} = 3.21\ \mu s$



$slew\ rate\ = \frac{0.720\ V-1.088V}{-501\ ns+999\ ns} =- 736\ kV/s$

We need the output to change from 5 V to 2.25 V (VCC/2)

$t_{pHL} \approx \frac{2.25\ V-5\ V}{736\ kV/s} = 3.05\ \mu s$

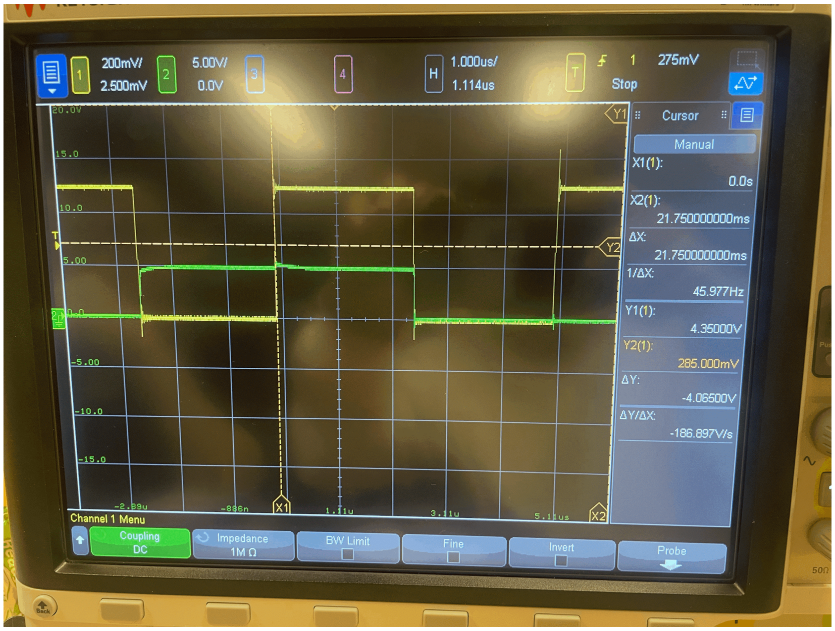
At 0.1 uF, $t_{pLH} = 3.21\ \mu s$, $t_{pHL} = 3.05\ \mu s$

Q5 Data Flip Flop

Frequency Relationship

The output frequency is half the input frequency, since the output toggles at every falling edge of the clock input.

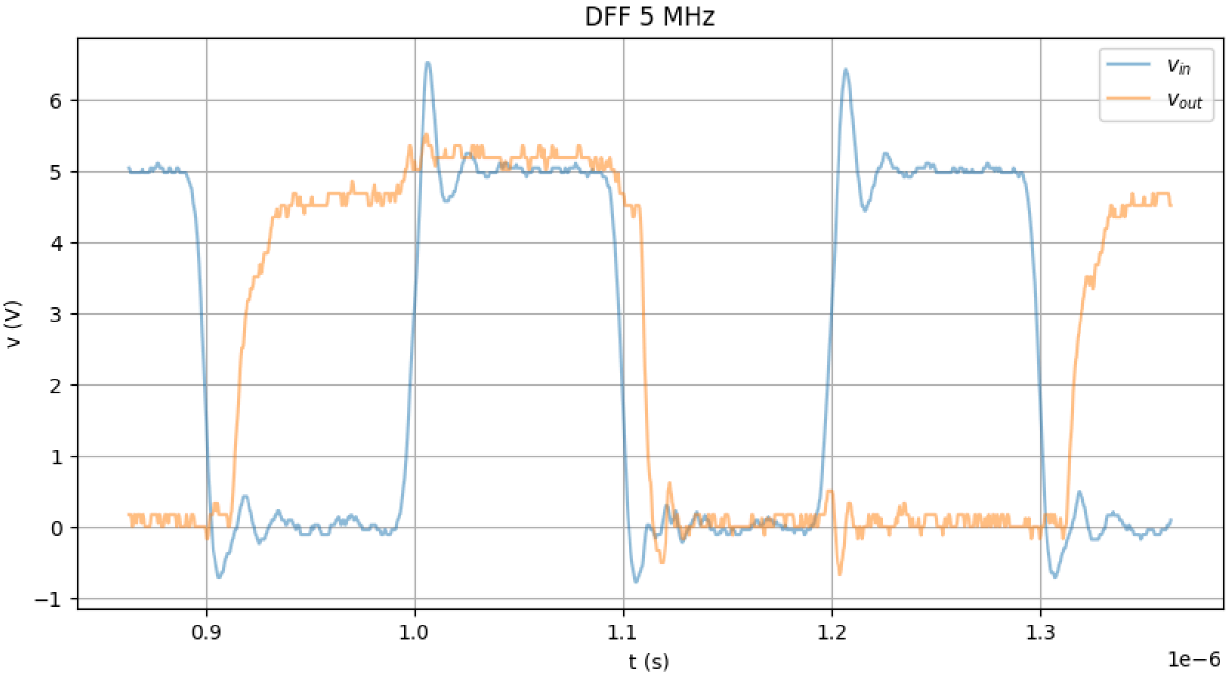
Waveform



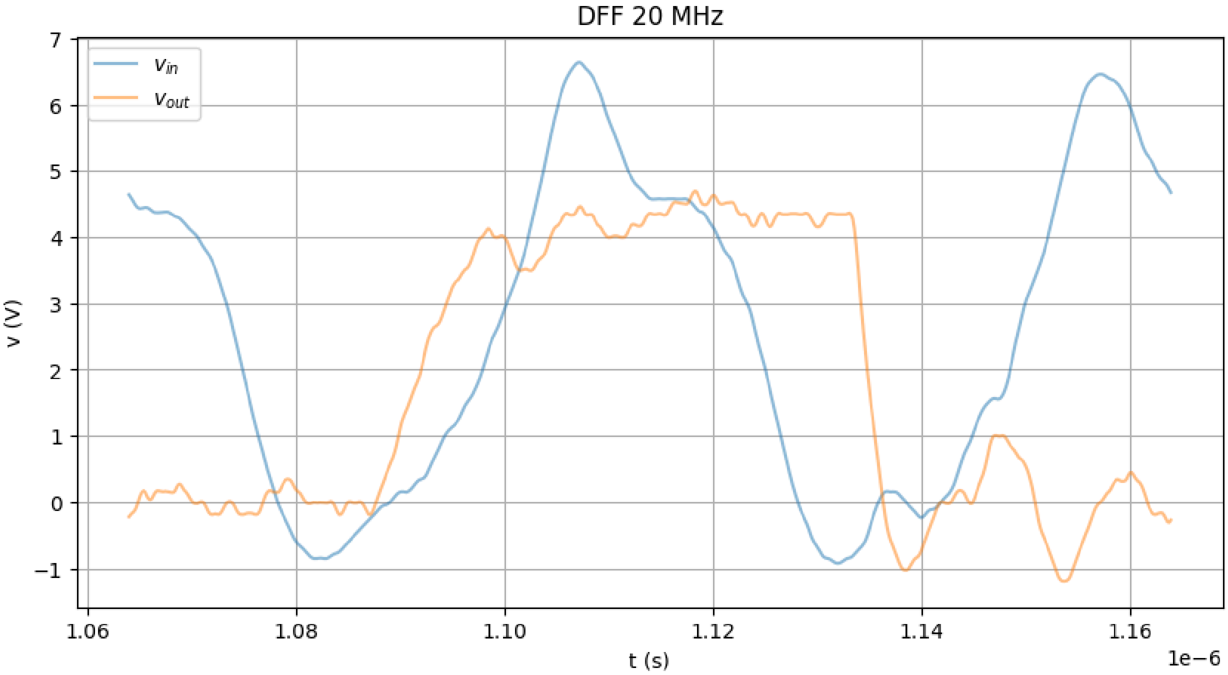
Waveforms at 1 kHz. Channel 1 (yellow) is the input. Channel 2 (green) is the output.

Frequency Limit

Waveforms at 5 MHz



Waveforms at 20 MHz (function generator limit)



At 20 MHz, the input from the function generator is very distorted, yet the output waveform frequency is still approximately 10 MHz.
The circuit continued to function at the high input frequency limit, because 74F74 has a typical maximum clock frequency of 125 MHz.

Q6 RS Switch Debouncer

Operation

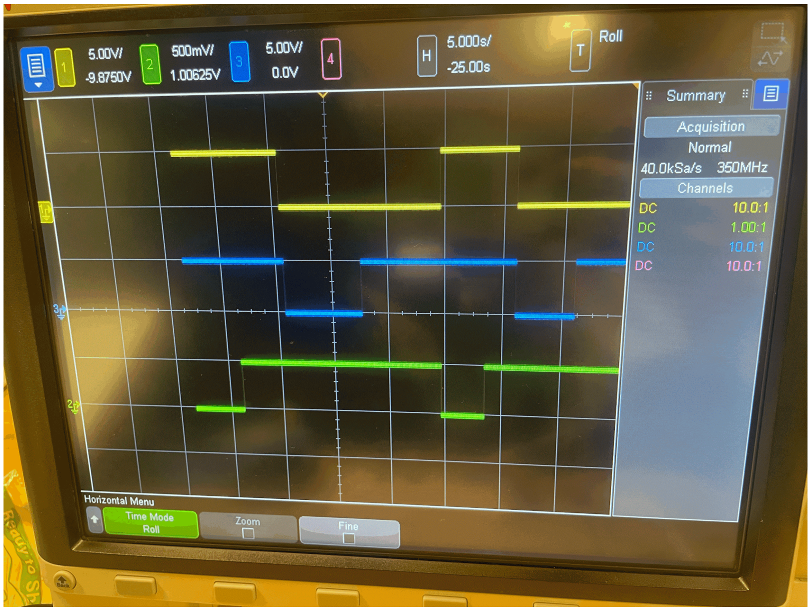
When a node is switched to ground, its voltage is 0 V (0). When it is not connected to ground, its voltage is pulled up to 5 V (1).

- When the switch grounds R, R = 0, S = 1; so Q = 1, Q’ = 0.
- When the switch grounds S, S = 0, R = 1; so Q = 0, Q’ = 1.
- When the switch is in between, R = S = 1, the two NAND gates function as NOT gates. Q = (Q’)’ = Q, Q’ = (Q)’ = Q’. Q and Q’ hold their previous values.

The circuit behaves as a regular switch, but when the switch is between R and S (connected to neither), the output holds its values, preventing noise from toggling the output.

R, S, Q Relationship

We plotted R, S, and Q in one graph, with DC bias for clarity.



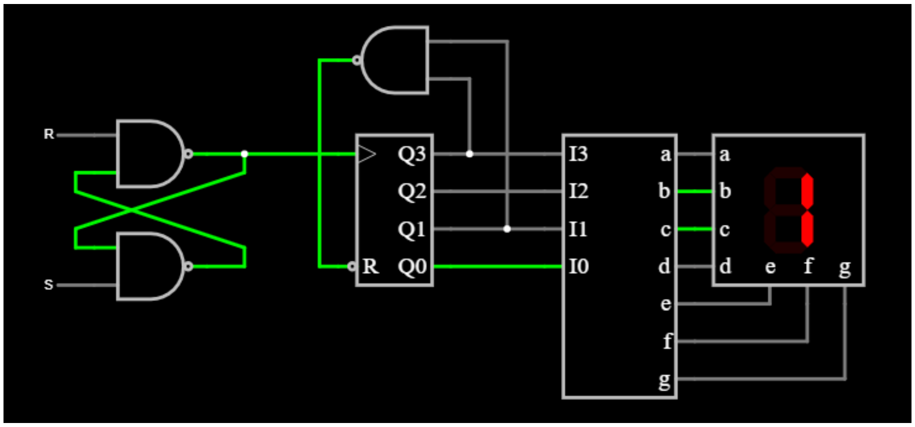
Channel 1 (yellow) is Q as the output. Channel 2 (green) is R. Channel 3 (blue) is S.

The results are consistent with the behavior discussed above.

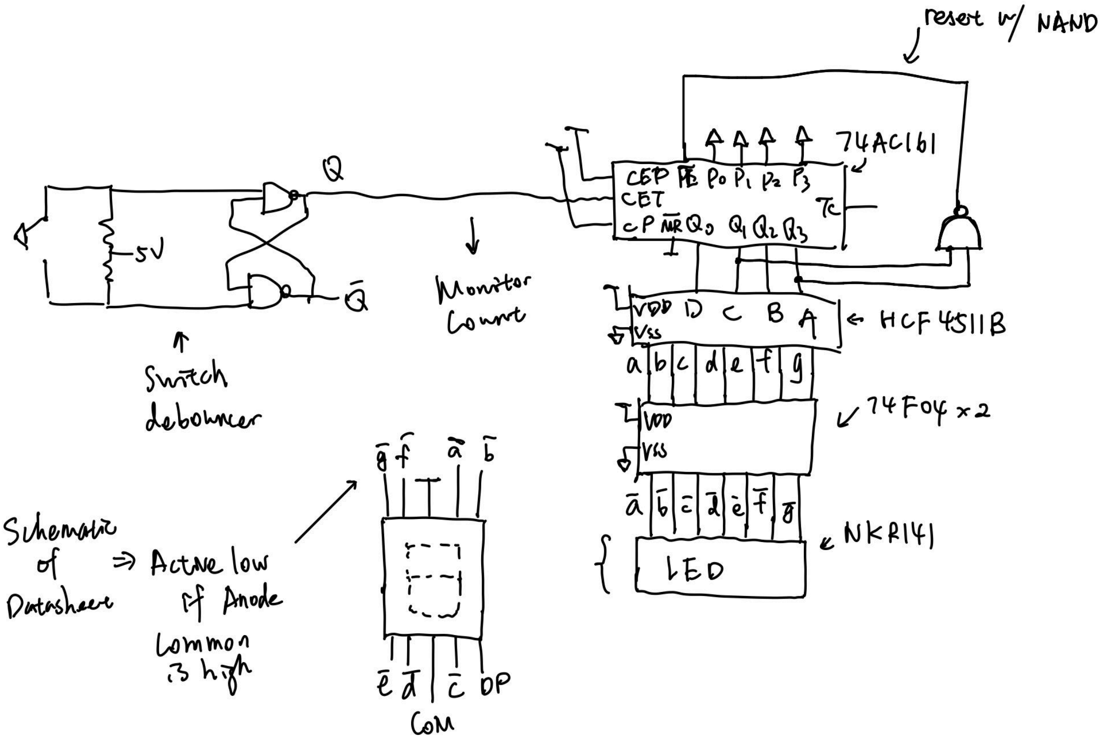
D Count-to-10

Schematic Design

The basic framework is shown below: connect the output of the RS switch debouncer to the clock input of the 16-counter. Once the counter reaches 10 (1010), trigger the reset. This can be done by setting R = Q3 NAND Q1. Q3 NAND Q1 is true if and only if the number reaches (or goes past) 10. The outputs of the of the counter go through a decoder before being fed into the 7-segment display.



Below is the detailed implementation:



Schematic Explanation

We use the switch debouncer to ensure a clean clock signal. At the rising edge of Q (clock), the counter will count up (based on the logic table provided in the 74AC161 datasheet).

The 74AC161 counter will increment the count at the rising edge of the input clock signal. However, when the first time both Q1 and Q3 are high, 74AC161 has counted up to 10, which means we need to reset our counter to start from 0 again. We thus use a NAND gate to reset 74AC161's output to all 0s for recount.

The output of the 74AC161 hex counter is fed to HCF4511B decoder to convert to 7 7-segment representation for LED.

We choose to drive the common Anode of the NKR141 7-segment LED high such that we will need active low input signals to turn on each segment of LED (based upon the schematic provided in the NKR141 datasheet). Thus we used two 74F04 inverters to convert the active high signals from HCF4511B to active low signals to actualize our design.

All chips are driven by VCC = 5V. Logic outputs are pulled up to VCC by 10k resistors. We tested the intermediate outputs with LEDs. All LEDs, including the 7-segment display, are connected in series with 220 ohm resistors.

Result

The circuit worked as expected. Below are some shots from 3 to 6 for illustration.