Vol 54 No 1, January 2020

ANALOG DEVICES
AHEAD OF WHAT'S POSSIBLE™
AnalogDialogue

RAQ Issue 173: On Building Physically Accurate Analog Switch Macromodels

Barry Harvey, *Analog IC Designer*

Question:

Can I improve the LTspice® model of an analog switch in case my analog design contains switches and muxes?

Answer:

Sure; it is not difficult to generate your own models.

Introduction

I was testing a circuit and found many discrepancies from the paper design I used to create it. The dynamics of the circuit were a bit unexpected, and the noise level was much larger than required. I needed to bring the circuit to a simulator to fully understand it.

The circuit involved analog switches and op amps. There are good macromodels for the op amps employed, but the analog switch macromodel was not designed for generality. In the header of the switch macromodel file is the warning that modeled parameters were only valid for a specific supply and temperature. Well, wouldn't you know it: my circuit has different operating conditions from the modeled one. The thing about analog switches is that they are so general-purpose that one operating point is not enough. The existing industry-standard models provide a good start, but if you enter the analog performance arena, you might need a new macromodeling approach that brings your simulation to a higher level.

As I began to browse through various analog switch macromodels from Analog Devices and other IC companies, I noticed that all their headers tell of no supply nor temperature dependence being modeled. Thus, I would have to make my own macromodel.

My philosophy in this work is that full transistors in the analog switches using the simplest device models provide all the behaviors to be emulated, but the interface from control pin to MOS gates should be the simplest behavioral components.

All work here is done with the LTspice simulator; the code would work on other simulators with a translation of the LTspice behavioral devices to SPICE-like polynomial functions.

We will develop the simulated behaviors in a specific sequence.

Developing LTspice Model Parameters for On Resistance

We will use the simplest model to run real MOS devices. To model on resistance, we will employ:

- W/L, the width (W) divided by the length (L) of an MOS device. W/L is the size or relative strength of the device.
- V_{TO}, the threshold voltage; and gamma, which modifies V_{TO} with device back-bias. The back-bias is the voltage between the on device and its body voltage; the body is frequently connected to the positive supply for the PMOS and to the negative supply for the NMOS in the switch.
- K_P, in the model, also known as K′ or K-prime. This parameter models the strength of the process and is multiplied by W/L to scale MOS currents. For a given process, the NMOS will have ~2.5× the K_P of the PMOS.
- RD, the parasitic resistance of the device's drain.

Different MOS processes have different intrinsic parameters. Table 1 is a collection of common CMOS processes, their characteristics, and estimated intrinsic parameters related to on resistance.

Table 1. Typical Semiconductor Process Parameters

Voltage Node (V)	Device Construction	Gate Oxide Thickness (m)	V_{TO}, n/p, V	Gamma, n/p, $V^{0.5}$	K_P, n/p, $\mu A/V^2$	L, µ	R_D, n/p, Ω
40	Drain drift region	10^{-7}	0.7/–0.9	0.4/–0.57	11/5	2	~80% of $R_{DS,ON}$
15	Soft drain diffusion	4×10^{-8}	0.7/–0.9	0.4/–0.57	22/10	1.5	~20% of $R_{DS,ON}$
5	Simple	1.4×10^{-8}	0.7/–0.9	0.4/–0.57	80/28	0.5	~0

Let us look at the ADG333A R_{ON} curves we wish to reproduce in Figure 1.

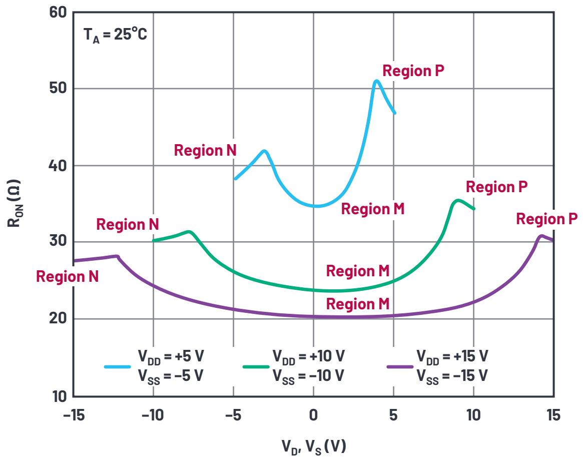


Figure 1. R_{ON} as a function of V_D (V_S), dual supply.

We see a general trend for this and any other analog switch that higher supply voltage reduces on resistance. As more voltage is applied to the switch MOS gates, the on resistance is reduced. We also see a distinct variation of on resistance with the analog level. In the N regions, the NMOS transistor in a switch is fully on, and as the analog voltage rises above the negative rail, the PMOS transistor turns on and helps reduce overall on resistance. The inflection point at region N is roughly a PMOS V_{TO} above the negative supply.

Similarly, in regions P, the PMOS device of the switch is fully on and the NMOS starts assisting the PMOS transistor roughly an NMOS V_{TO} below the positive supply.

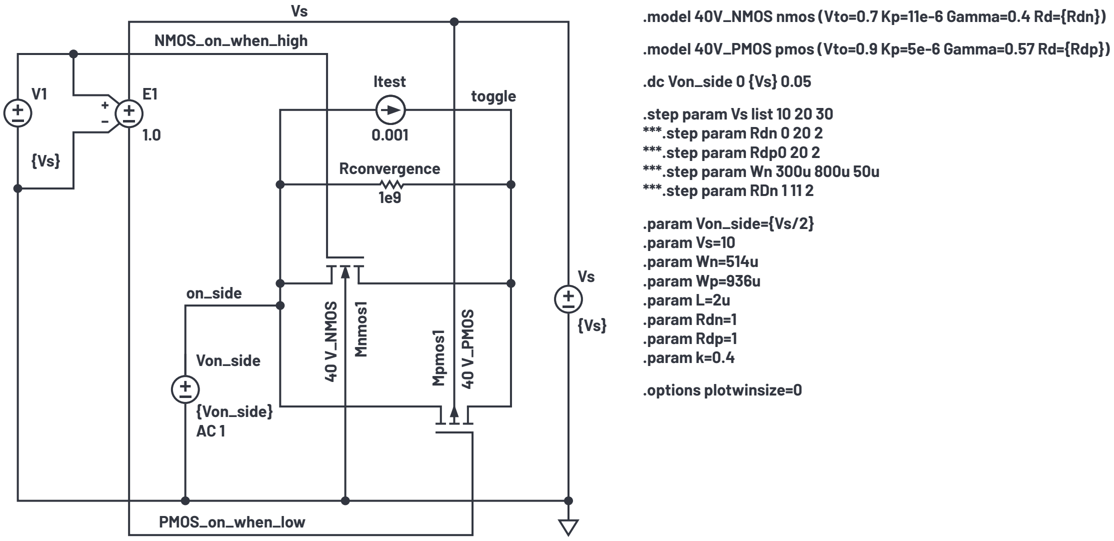
Regions M are in the middle of the N and P regions with the NMOS and PMOS working in parallel, but each varying in on resistance depending on the analog signal level between the supplies.

To start the curve-fitting process, we first estimate the size of each transistor. The low voltage curve gives the best curve-fit for transistor $R_{DS,ON}$. In region N, with the analog signal at the negative supply, the PMOS device is off and R_{ON} of the part is equal to the R_{ON} of the NMOS transistor. With

$$R_{DS,\ ON} = \frac{1}{k_P\left(\frac{W}{L}\right)(vgs - V_{TO})} \tag{1}$$

using the 40 V NMOS typical process values, we set $R_{DS,ON}$ = 38 Ω from the curve in Figure 1 and using the process quantities given find WNMOS = 2 µA/(38 Ω × (11 × 10^{-6} µA/V_2) × (10 V – 0.7 V)) = 514 µm. The PMOS switch would have an on resistance of 47 Ω from the above curve and thus a width of 936 µm.

I used the LTspice test circuit in Figure 2. Note that parameters R_{DN} and R_{DP}, the parasitic drain resistances, are of modest value. I started with a value of 1 µ, which caused simulator convergence slowdown. The R_{DN} value of 1 allows proper simulation speed. Adding $R_{CONVERGENCE}$ improved simulator noise and speed by giving the toggle node a convergeable conductance. I tested a floating current source for measuring on resistance.



```
.model 40V_NMOS nmos (Vto=0.7 Kp=11e-6 Gamma=0.4 Rd={Rdn})
.model 40V_PMOS pmos (Vto=0.9 Kp=5e-6 Gamma=0.57 Rd={Rdp})
.dc Von_side 0 {Vs} 0.05
.step param Vs list 10 20 30
***.step param Rdn 0 20 2
***.step param Rdp0 20 2
***.step param Wn 300u 800u 50u
***.step param RDn 1 11 2
.param Von_side={Vs/2}
.param Vs=10
.param Wn=514u
.param Wp=936u
.param L=2u
.param Rdn=1
.param Rdp=1
.param k=0.4
.options plotwinsize=0
```

Figure 2. On resistance test circuit.

Figure 3 shows the simulated results for various supplies.

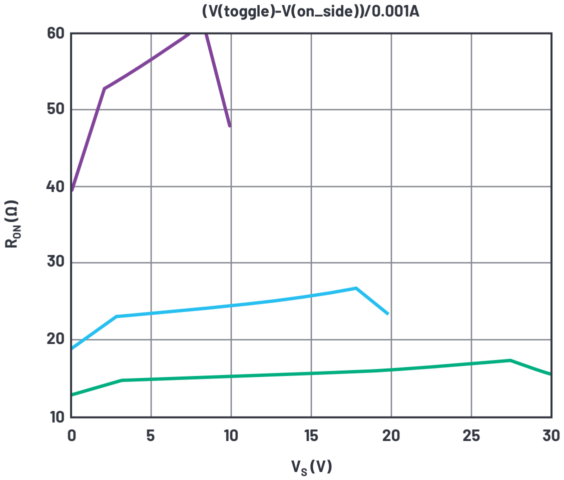


Figure 3. On resistance simulation results with initial model values.

This is a good start. The kink at the low voltage end for V_S = 30 V is at 3.6 V in the simulation and 2.7 V in the data sheet. This suggests we reduce the PMOS V_{TO}, but 0.9 V is already a realistic minimum. Better to adjust the gamma of the PMOS, which was only a guess anyway.

The kink near maximum supply is 2.5 V below the 30 V rail, where in the data sheet it should be ~1 V. Various values of gamma exaggerated the kink voltage from the rail; we will just set the NMOS V_{TO} to 1 V and its gamma to zero. A zero gamma is unexpected, but we're only trying to curve-fit. Figure 4 shows simulation results from these values with the gamma of the PMOS stepped for several supplies. We focus on the 30 V curves, which maximize the gamma effect compared to lower supplies.

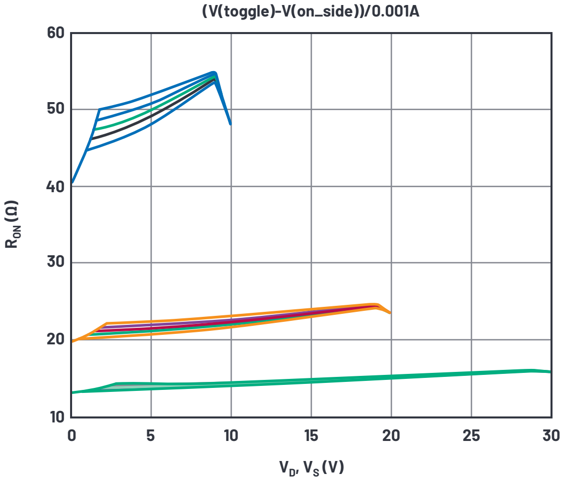


Figure 4. On resistance simulation results with gamma-p varied.

From the stepped curves, we'll choose a PMOS gamma = 0.4.

On to R_{ON}. Observe that the 10 V curves are representative of the data sheet curve at the supply extremes, but the simulation produces too low a R_{ON} for the 20 V and 30 V curves. The R_{ON}s are equal to $R_{DS,ON}$(NMOS) + R_D(NMOS) at the negative supply extreme and $R_{DS,ON}$(PMOS) + R_D(PMOS) at the positive supply extreme. For high supplies, the R_D parameter will be more significant than W/L, and for low supplies, W/L will dominate. We have two variables to juggle here; too laborious. We will posit that R_{ON} varies with supply due to the NMOS being variably enhanced, but the R_D value doesn't change with supply voltage (okay, it probably does in the case of drains with drift regions, but let's keep this simple). If we note the difference in data sheet R_{ON} between 10 V and 30 V supplies (11.4 Ω), we can compare that to the above curves where we step only W_N (width of the NMOS in the switch). After a bit of iterations of W_N in simulations it's clear that we need W_N = 1170 µm to get the required ΔR_{ON}, quite a lot more than the initial guess. Figure 5 shows our current results.

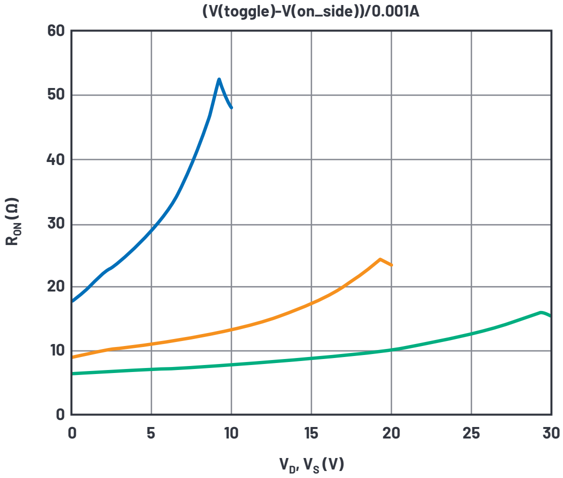


Figure 5. On resistance simulation results with W_N determined.

While the R_{ON} of the NMOS has the right supply sensitivity, the curves are too low a value at zero volts, and we must increase the fixed R_{DN}. After increasing and iterating R_{DN}, we get a best value of R_{DN} = 22 Ω, and the resulting curves are in Figure 6.

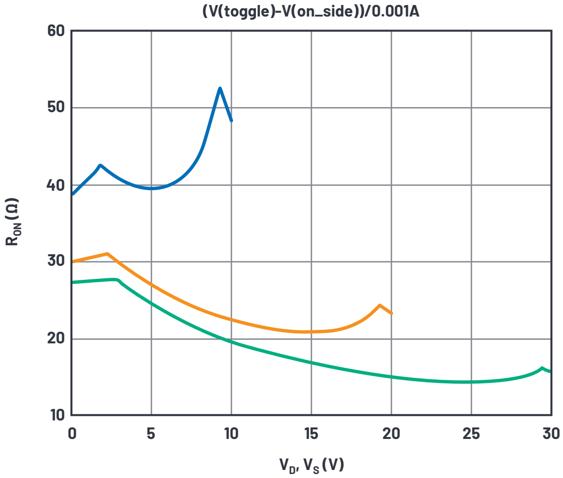


Figure 6. On resistance simulation results with R_{DN} determined.

We next determine W_P (width of the PMOS in the switch) to simulate the R_{ON} at maximum voltage, and get W_P = 1700 μm, again quite a lot more than initially guessed. With R_{DP} also set to 22 Ω, we get the final R_{ON} curve in Figure 7.

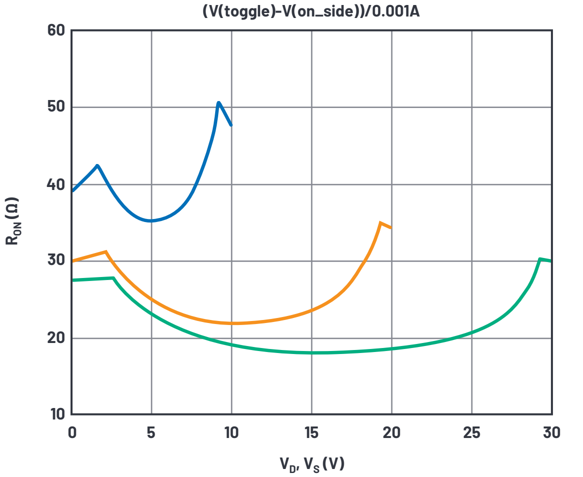


Figure 7. On resistance simulation results with W_P and R_{DP} determined.

Pretty good agreement here; there are only a few features different from the data sheet. One is that the inflection points are smooth in the data sheet curve but truly pointed in simulation. This is probably because the simple MOS model used does not support subthreshold conduction, and the simulated device turns truly off at V_{TO}. Real devices are not off at V_{TO}, but smoothly reduce current below that voltage.

Another error is most obvious in the 30 V curve. R_{ON} is 15% low at midsupply compared to data sheet. Perhaps this is due to JFET effects within the drain drift region, also not modeled.

As for temperature, there is fair but not strong compliance, seen in Figure 8.

The simulation has temperature dependence, but not as much as the data sheet curves. In the simulation model the R_D terms do not have tempco. R_Ds could be modeled by external resistors with correct tempco, but we will leave it as is for simplicity.

Obtaining LTspice Model Parameters for Charge Injection

When MOS transistors turn off, the charge in the channel must go somewhere, so it squirts out of the drain and source terminals. When an analog switch is turned off, charge also goes out and is called charge injection. A common way of measuring it is to place a fixed voltage on one end of an on switch and a large capacitor at the other end. When turned off, the charge is captured by the capacitor and a small voltage step occurs. We will now the add gate oxide thickness $T_{OX} = 1 \times 10^{-7}$ to the MOS models (gate capacitance is the largest source of charge injection). Our simulation setup is shown in Figure 9.

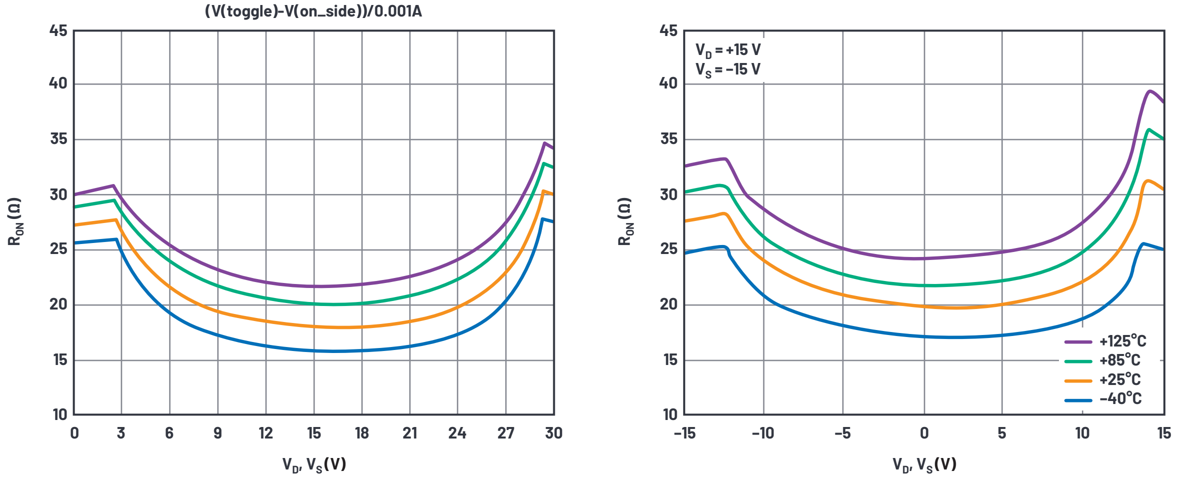
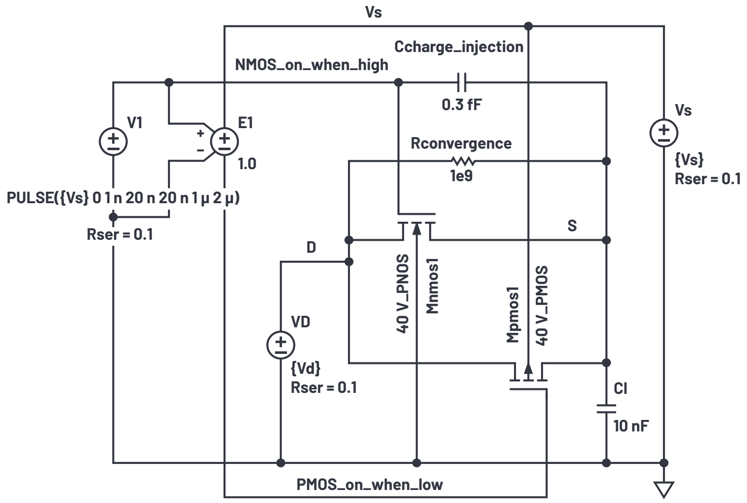


Figure 8. On resistance simulation and data sheet results over temperature.



```
.model 40V_NMOS nmos (Vto=0.7 Kp=11e-6 Gamma=0 Rd=22 tox=0.1u)
.model 40V_PMOS pmos (Vto=0.9 Kp=5e-6 Gamma=0.4 Rd=22 tox=0.1u)
***.dc Vd 0 {Vs} 0.05
.tran 100n
***.step param Vs list 12 33
.step param Vd 0 30 0.2
***.step temp list –40 25 85 125
.param Vs=30
.param Vd={Vs/2}
.param Wn=1170u
.param Wp=1700u
.param L=2u
.options plotwinsize=0
.meas TRAN Charge_Injection find (V(S)–V(D))*10n at=99n
```

Figure 9. Charge injection simulation setup.

The data sheet charge injection test circuit places a voltage source at the D terminal of a switch, and the capacitor Cl at the S terminal of the switch. When the switch transistors are turned off, Cl is isolated and integrates charge pumped into it by the switches. The waveform of such an event with V_D held to 24 V with a 30 V supply is shown in Figure 10.

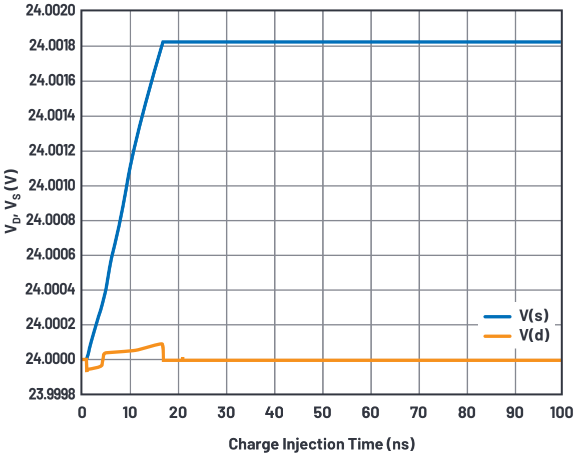


Figure 10. Charge injection simulation waveforms.

The charge injected is the voltage jump between V(S) and V(D) times the 10 nF hold capacitor. We can step the switch voltage V_D across the supply voltage and use the .meas statement to capture the value of charge injection at each voltage. Figure 11 shows the data sheet curve and simulated results.

Our simple MOS model does not mimic the shape of the data sheet curve very well, but the peak-to-peak charge injection is 32 pC in the data sheet curves and 31 pC in simulation. Surprisingly close, but if we had to, we could tweak T_{OX} to perfect the simulation results.

There is an offset between the curves that we can compensate for using $C_{CHARGE\_INJECTION}$. After fiddling with some values, we choose an optimal $C_{CHARGE\_INJECTION}$ = 0.28 pF. If an opposite polarity of shift were needed $C_{CHARGE\_INJECTION}$ would be reconnected to the PMOS_on_when_low node.

The tweak capacitor $C_{CHARGE\_INJECTION}$ was a convenient way to offset the charge injection vs. the analog voltage simulation curve. What if the peak-to-peak injection simulated were too small? Well, most of the charge injection is mostly the switches' gate voltage swings sending charge through the gate-channel capacitance of the switch transistors. If we simulate too little injection, we can simply increase one or both gate areas. To do this we would increase the parameters L and W of a switch device by the same factor, being careful to not modify the W/L ratio that sets on resistance. Rather than use $C_{CHARGE\_INJECTION}$ we could have increased the NMOS W and L.

Alternatively, we could adjust T_{OX} in each device to get better charge injection correlation. This would not be physically possible, but hey—it's just a simulation. With the simple models we are using, T_{OX} does not influence other behaviors.

Obtaining LTspice Model Parameters for Capacitances

Having set up parameters for good R_{ON} and charge injection simulation results, we now simulate S and D terminal capacitances.

One important point is that both the drain and source regions of high voltage MOS switches must have drift regions. As a switch, you can't tell the functional difference between sources and drains, and the body potential to both drains and sources will require the drift regions in each. This is also true of the medium-voltage soft diffusions, but non-existent in low voltage MOS. We have lumped the drift region resistance that would exist in both drain and source into R_D, and that works fine for switches, but not transistors in saturation.

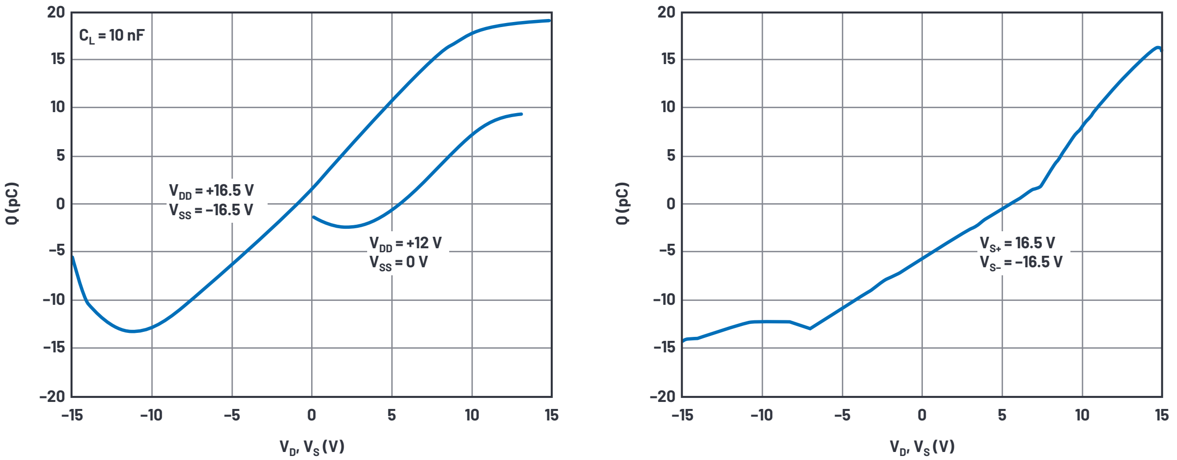


Figure 11. Charge injection data sheet and simulation waveforms.

Figure 12 shows our simulation setup.

In LTspice, you can run an .ac on only one frequency, using the list option in .ac, but offer only one frequency argument (1 MHz here). Then you run a .step V_{SOURCE} dc voltage across the supply range to get a capacitance vs. voltage sweep.

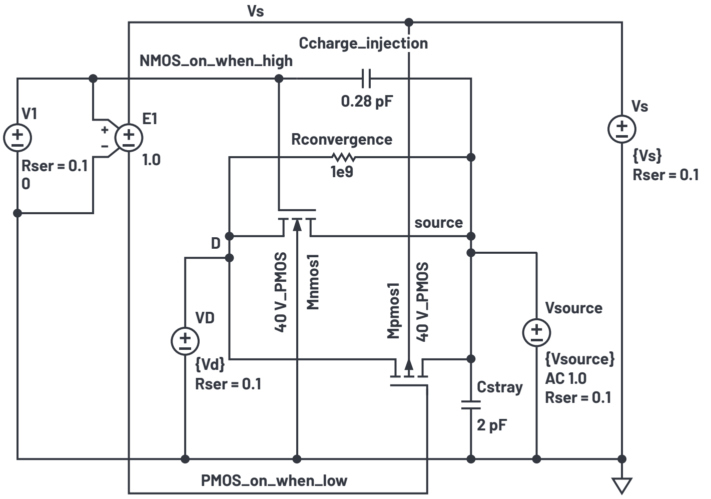
The D terminal of the off-switch device is held to midsupply. The S terminal, renamed source here to prevent confusion with V_S, is driven by a voltage source with dc value sweeping from 0 V to V_S and with an ac drive of 1 V. Capacitance is derived from $I(V_{SOURCE})/(2 \times \pi \times 1\text{ MHz} \times 1\text{ V})$. The logic drive V1 is changed to 0 V to turn the transistors off.

Drain and source capacitances to bulk are C_{BD} and C_{BS} respectively in the model statement. There are built-in default concentrations, built-in voltage, and exponent in the model that make C_{BD} and C_{BS} voltage variable. Because they are symmetrical, drain and source capacitances would be made equal. Further, because the PMOS is a different width from the NMOS, the ratio of $C_{BD,NMOS}/C_{BD,PMOS} = C_{BS,NMOS}/C_{BS,PMOS} = W_N/W_P$, which we established in the on resistance modeling. Figure 13 shows the simulation results.

The displays are $I(V_{SOURCE})/(2 \times \pi \times 1\text{ MHz})$, which is capacitance. LTspice doesn't know this and displays pA instead of pF.

Unfortunately, we have no data sheet curves to compare to. We do know from the specification table in the data sheet that the capacitance—probably at midsupply, but not specified in the data sheet—is typically 7 pF at 30 V supply and 12 pF at 12 V supply. I adjusted the CBs to obtain the 7 pF curve at 30 V, but only simulated 10 pF at a 12 V supply. After fiddling with built-in potential and capacitance formula exponent, the model used allows no flexibility to improve the 12 V/30 V compliance.

Figure 14 shows the on-state capacitance simulation setup.



```
.model 40V_NMOS nmos (Vto=0.7 Kp=11e-6 Gamma=0 Rd=22
+ tox=0.1u cbd=9p cbs=9p)

.model 40V_PMOS pmos (Vto=–0.9 Kp=5e-6 Gamma=0.4 Rd=22
+ tox=0.1u cbd=13.5p cbs=13.5p)

***.dc Vsource 0 {Vs} 0.05
***.tran 100n
.ac list 1e6

***.step param Vs list 12 30
.step param Vsource 0 12 0.2

***.step temp list –40 25 85 125

.param Vs=12
.param Vd={Vs/2}
.param Vsource={Vs/2}

.param Wn=1170u
.param Wp=1700u
.param L=2u

.options plotwinsize=0

***.meas TRAN Charge_Injection find (V(S)–V(D))*10n at=99n
.meas AC Isource FIND mag(i(Vsource)) at=1e6
```

Figure 12. Off-capacitance test simulation setup.

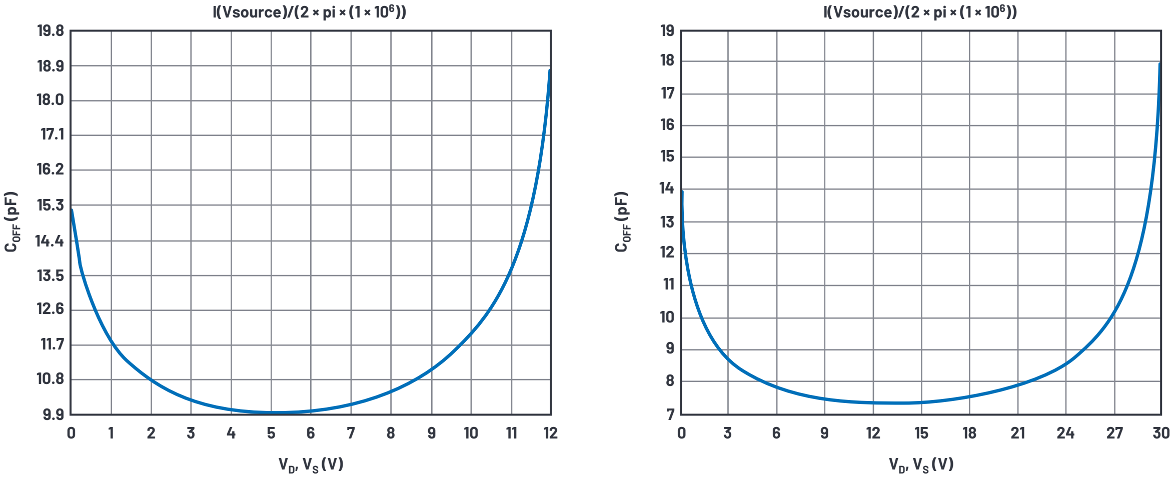


Figure 13. Off-capacitance vs. dc voltage at V_S = 12 V (left) and 30 V (right) results.

Here the right switch of a full spdt switch is on, and the left switch is off and connected to a $V_S/2$ source. The capacitances of the right half of the left switch and full capacitances of the right switch, plus inevitable parasitic capacitances at D and S terminals are all paralleled and driven by our 1 MHz test signal at the V_s source, whose dc level is stepped across ground to V_S. Figure 15 shows the results.

We simulate 29.5 pF and 21.4 pF where the data sheet gives 26 pF and 25 pF. Considering the variability in circuit-board layout capacitance, we'll call this close enough.

Leakage Currents

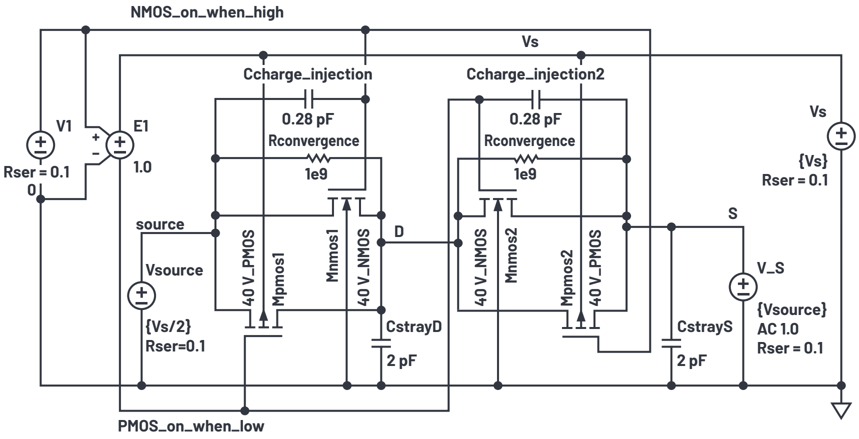
The data sheet curves show voltage-dependent pA-level leakage currents at 25°C, but the data sheet specification only guarantees hundreds of pA. I am swayed more by the curves' results at 25°C. The small leakage currents apparently were not considered important enough in this device to guarantee at test. To be fair, measuring single pA takes a lot of engineering development effort as well as long test times.

At 85°C, the guarantee is a few nA (which can be measured efficiently) with a typical result in the range of a few hundred pA. I'm going to accept these typical results as good.

Leakage current is a product shortcoming; it doesn't have tight statistics and varies wildly with temperature. It is not the kind of specification that we design to—rather, it's a quantity that disrupts the circuits it's connected to. For macromodel use, any leakage of proper magnitude will be simulated as a circuit defect and be a useful warning to the designer. I'll choose a target of 1 nA for an on switch at 85°C.

The model we have shows no leakage beyond $R_{CONVERGENCE}$ and G_{MIN} currents. G_{MIN} is a resistor the simulator places across junctions to assist convergence. It is normally 1×10^{-12} conductance, but in the presence of 30 V supplies we can get multiples of 30 pA currents, way too high for this work. G_{MIN} will be reduced to 1×10^{-15} in the .options line of the simulation and $R_{CONVERGENCE}$ raised to 1×10^{15}.

The physical origin of these leakages is probably mostly from electrostatic discharge (ESD) protection diodes connected to every pin. We will insert them into the simulation setup in Figure 16.



```
.model 40V_NMOS nmos (Vto=0.7 Kp=11e-6 Gamma=0 Rd=22
+ tox=0.1u cbd=9p cbs=9p))

.model 40V_PMOS pmos (Vto=–0.9 Kp=5e-6 Gamma=0.4 Rd=22
+ tox=0.1u cbd=13.5p cbs=13.5p)

***.dc Vsource 0 {Vs} 0.05
***.tran 100n
.ac list 1e6

***.step param Vs list 12 30
.step param Vsource 0 12 0.2

***.step temp list –40 25 85 125

.param Vs=12
.param Vd={Vs/2}
.param Vsource={Vs/2}

.param Wn=1170u
.param Wp=1700u
.param L=2u

.options plotwinsize=0

***.meas TRAN Charge_Injection find (V(S)–V(D))*10n at=99n
.meas AC Isource FIND mag(i(Vsource)) at=1e6
```

Figure 14. On-capacitance test simulation setup.

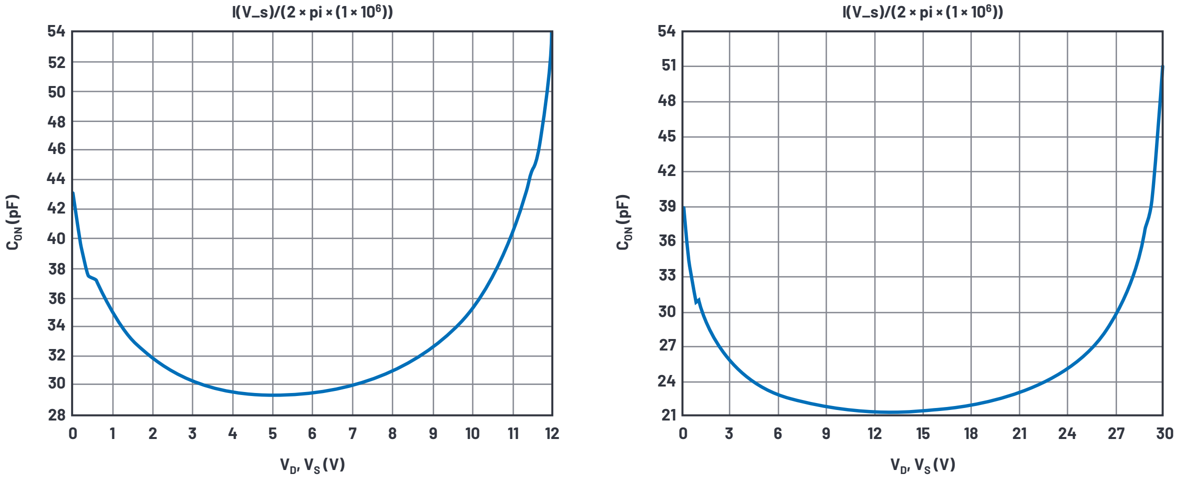
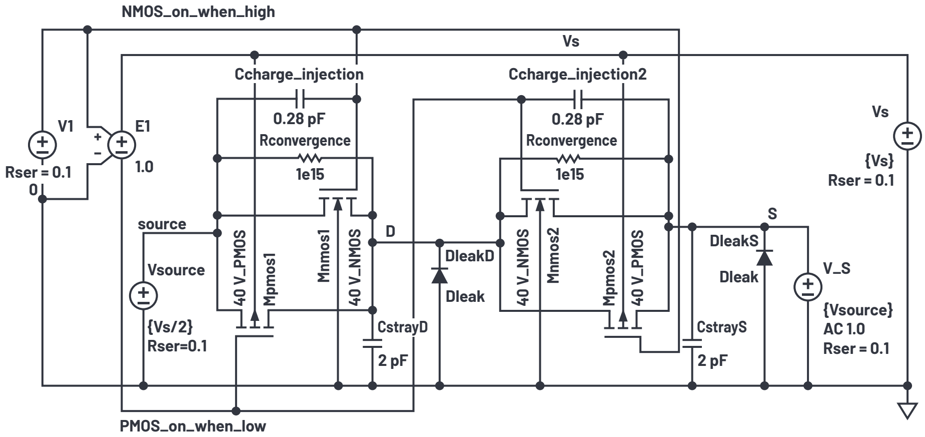


Figure 15. On-capacitance vs. dc voltage at V_S = 12 V (left) and 30 V (right) results.

After fiddling with IS in the diode model, we get leakage over temperature in Figure 17.



```
.model 40V_NMOS nmos (Vto=0.7 Kp=11e-6 Gamma=0 Rd=22
+ tox=0.1u cbd=9p cbs=9p))
.model 40V_PMOS pmos (Vto=-0.9 Kp=5e-6 Gamma=0.4 Rd=22
+ tox=0.1u cbd=13.5p cbs=13.5p)
.model Dleak D (Is=2.8e-13)
***.dc Vsource 0 {Vs} 0.05
***.tran 100n
.ac list 1e6
.op
***.step param Vs list 12 30
***.step param Vsource 0 30 0.2
***.step temp 25 125 5
.param Vs=30
.param Vd={Vs/2}
.param Vsource={Vs/2}
.param Wn=1170u
.param Wp=1700u
.param L=2u
.options plotwinsize=0 gmin=1e-15
***.meas TRAN Charge_Injection find (V(S)-V(D))*10n at=99n
.meas AC Isource FIND mag(i(Vsource)) at=1e6
```

Figure 16. Leakage test simulation setup.

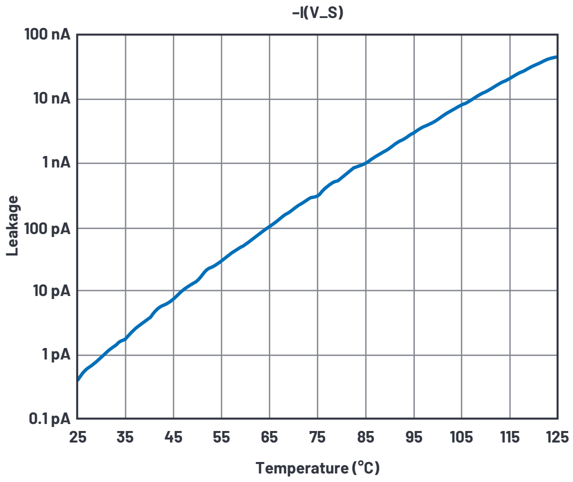


Figure 17. Leakage test over temperature simulation results.

Logic Interface and Gate Drivers

A purely behavioral logic-to-gate drive circuit is shown in Figure 18.

The external logic input is at the In terminal at the left of Figure 18. It is the input of an ideal transconductance Glogic_thresholda, which has a piecewise-linear transfer function. For logic inputs below 1.37 V, the output at logica node is 0 V; for inputs above 1.43 V logica is at 1 V; and between 1.37 V and 1.43 V in logica moves linearly from 0 V to 1 V. Glogic_thresholda thus ignores supply variations to provide a 1.4 V input threshold.

Transiently, Cdelaya slows down the logica node so that we can pick off some time points from it. To make a comparator we again use a transconductance, here Gbreakbeforemakena whose output goes from 0 V to 1 V again but with the threshold skewed a bit above 0.5 V. As seen in Figure 19, the skewed pickoff voltages 0.52 V and 0.57 V rather than 0.5 V allow faster turn-off from exponentials falling from 1 V than the turn-on time for exponentials rising from 0 V.

Full gate drive voltage is produced by the B_non and B_pon behavioral current sources. B_nona sources a current of $V_{DD}/1000$ when node n_breakbeforemakena >0.5 V, driving the voltage at node nona to V_{DD}, as loaded by a 1000 Ω resistor. When node n_breakbeforemakena <0.5 V, the node nona is driven to V_{SS}. Thus, we have a nice rail-to-rail gate drive that complies with supply voltages and has a fixed 1.4 V input threshold.

One more characteristic needs explanation. Note that in Figure 20, higher supply voltages reduce the delay times. This is implemented by B_supplysensitivitya, which feeds back to Cdelaya a fraction of its own dynamic current that varies with V_{DD}. Rsupply_sensitivitya drops very little voltage due to Cdelaya current, leaving Cdelaya's behavior mostly a pure capacitor. Feeding a replica of Cdelaya's current back to Cdelaya essentially creates a controllable variable capacitor, and the math inside Bsupply_sensitivitya creates the delay vs. V_{DD} curve in Figure 20.

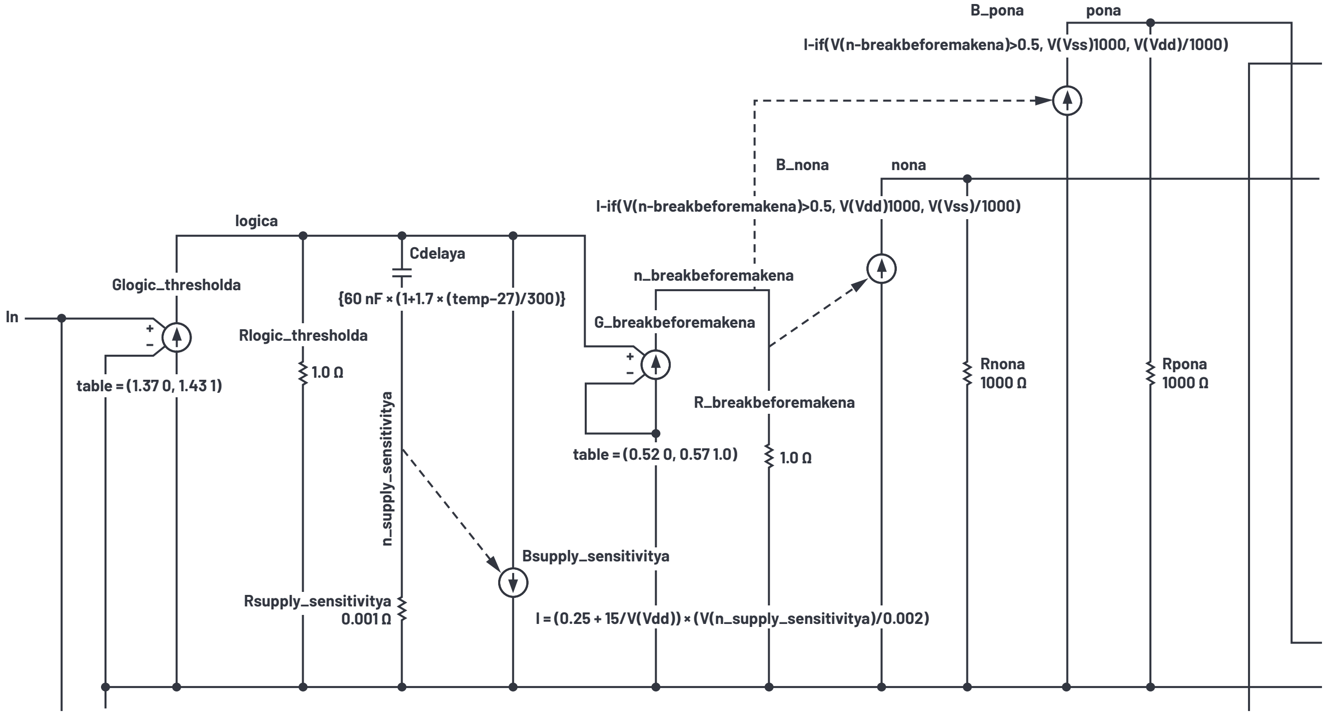


Figure 18. Behavioral logic-to-gate interface.

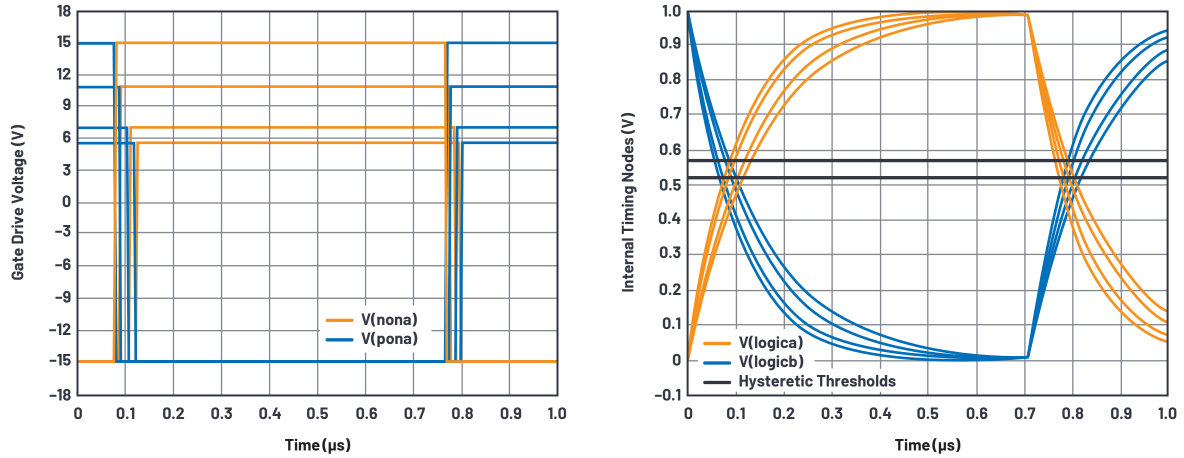


Figure 19. Break-before-make timing.

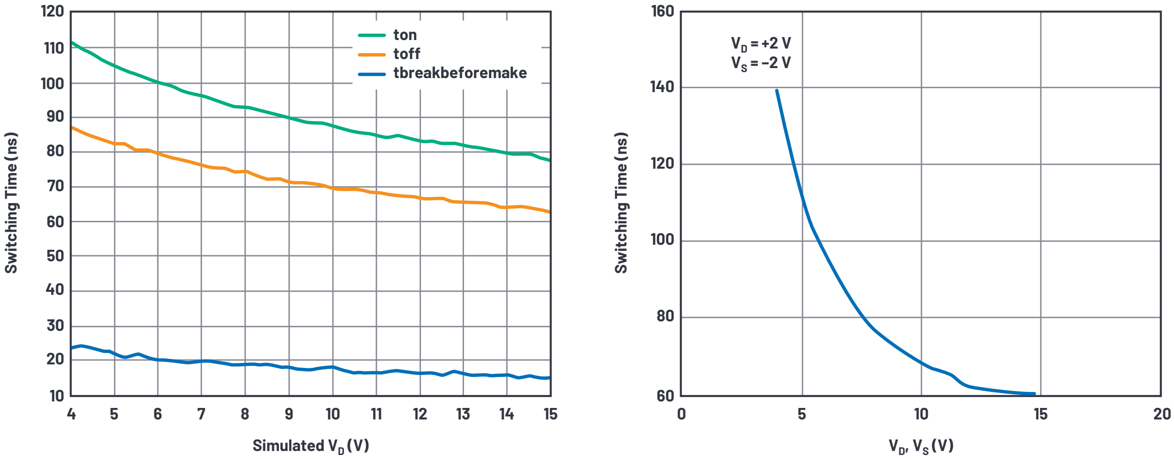


Figure 20. Break-before-make timing results from simulation and data sheet curve.

Well, our circuit emulates the T_{ON} delay as 111 ns for V_{DD} = 4 V while the data sheet curve says 140 ns; and for V_{DD} = 15 V simulated delay is 77 ns vs. data sheet delay of 60 ns. Not great correlation; I'll leave it to the reader to refine the Bsupply_sensitivity function to do better. At least the break-before-make varies nicely between 15 ns and 24 ns.

While we don't have much data sheet data on delay vs. temperature, I added a temperature term in Cdelaya to at least model slowdown when hot, seen in Figure 21.

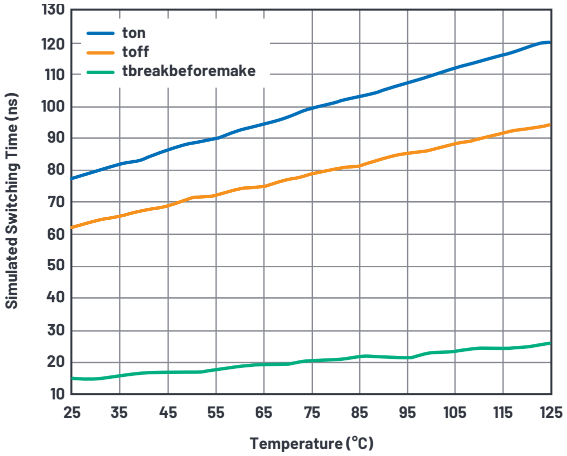


Figure 21. Timing delays vs. temperature.

Assembling the Macromodel

Figure 22 shows the assembled analog switch that will become a subcircuit. Hard L and W numbers were placed into the transistor symbols instead of parameters, and all excitation and I/O are removed in favor of pin connections SA, D, SB, In, V_{DD}, V_{SS}, and Gnd_pin.

A second logic interface is provided for the other switch of the spdt pair. ESD protection diodes are installed between analog terminals and V_{SS} and between the logic In and ground. Note that the "-a" suffix in names of the upper logic interface devices and nodes are replicated as "-b" suffix in the lower interface. Glogic_thresholdb interface has the opposite output from the table in Glogic_thresholda to allow one or the other switch pair to operate rather than be turned on simultaneously.

An alternative ESD protection scheme involves diodes from a protected pin to both V_{DD} and V_{SS}, and a clamp between V_{DD} and V_{SS}. The data sheet generally gives insight as to the protection scheme, and leakage currents are assigned to both supplies.

The spdt subcircuit is given a symbol and used four times in the master schematic ADG333A.asc of Figure 23.

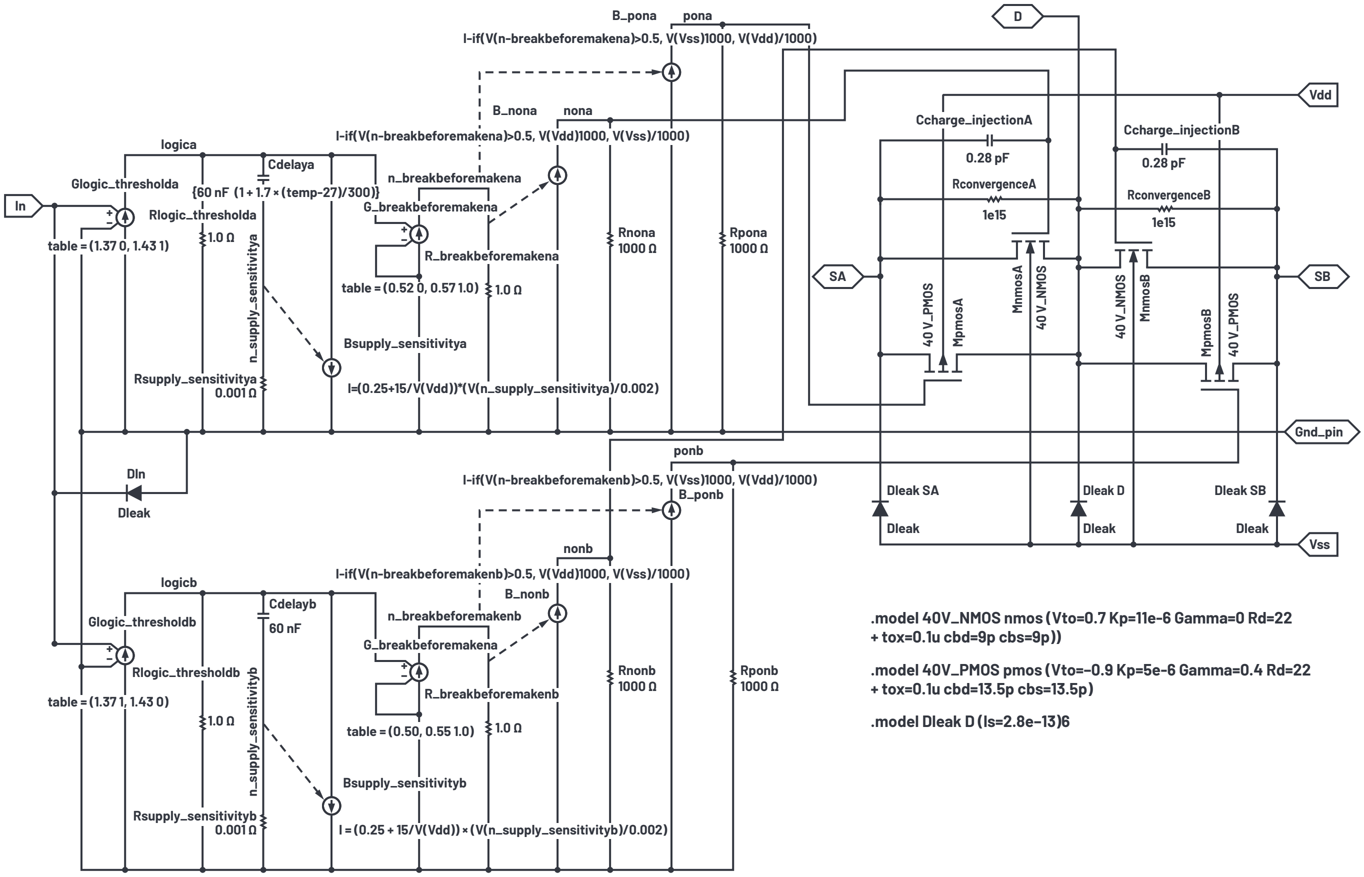


Figure 22. Assembled SPDT subcircuit spdt 40V.asc.

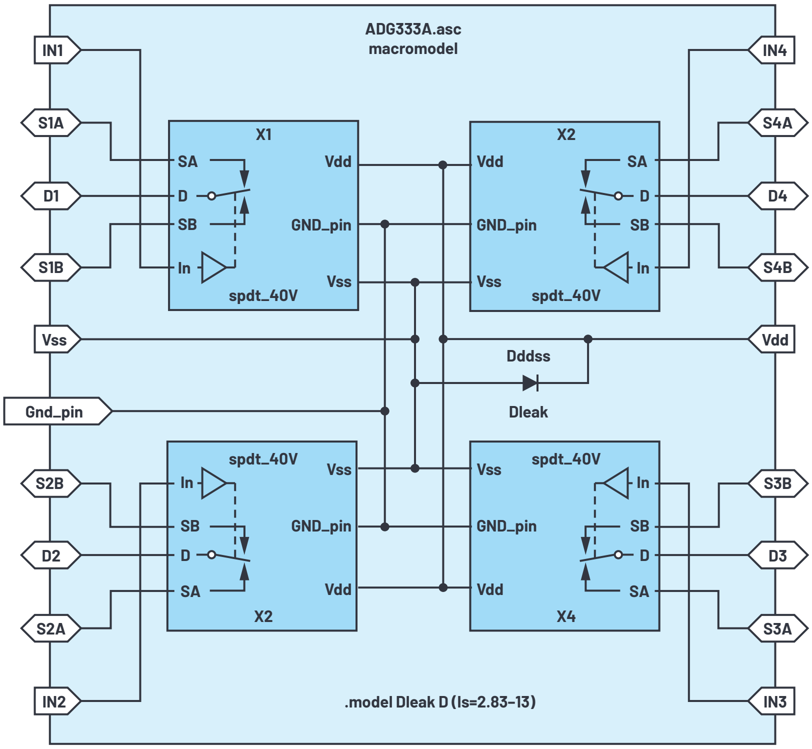
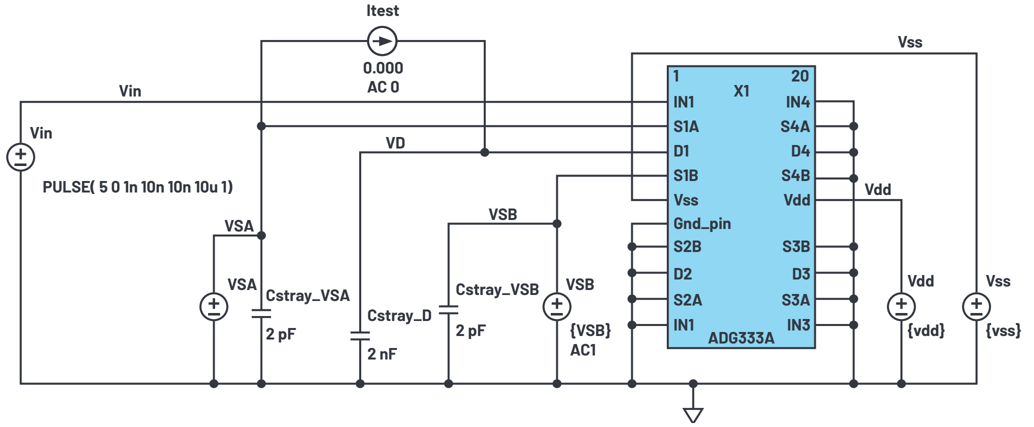


Figure 23. ADG333A macromodel circuit schematic.



```
***.dc VSA {Vss} {Vdd} 0.1
.ac dec 20 1k 1g
***.ac list 1e4
.tran 20u
***.step param VSA -15 15 0.25
***.step param Vdd list 10 20 30

.param Vdd=15
.param Vss=-15
.param VSA=0
.param VSB={VSA}

***.meas TRAN Charge_Injection
+ find (V(VD)-V(VSA))*10n at=1u
```

Figure 24. ADG333A macromodel test bench.

Figure 24 is the test bench schematic for verifying final macromodel results.

Summary

We've seen how to realize a decent macromodel for a specific analog switch and how to obtain parameters that support a few different semiconductor processes used to realize the physical device. The resulting macromodel displays defects such as on resistance and its variations, charge injection as a function of supply and signal level, parasitic capacitances and their variations over voltage, logic interface delays, and leakages. Hopefully, the macromodels will be helpful in simulating the real performance of analog switches.

Addendum

To download LTspice, please visit *analog.com/ltspice*.

Here is the LTspice text file of the macromodel symbol, to be filed under the name ADG333.asy. It contains subcircuit simulation details. Rather than copy the ADG333.asc schematic into every schematic that uses it, we use a symbol that refers to it as the .asy. Within the ADG333 symbol are individual switch symbols. This is the symbol simulation content to be filed as spdt_40V.asc. The actual symbol is to be filed as spdt_40V.asy.

About the Author

Barry Harvey has worked as an analog IC designer, designing high speed op amps, voltage references, mixed-signal circuits, video circuits, DSL line drivers, DACs, sample-and-hold amplifiers, multipliers, and more. He has an M.S.E.E. from Stanford University. He holds more than 20 patents and has published about as many articles and papers. Barry's hobbies include repairing used test equipment, playing guitar, and working on Arduino-related projects. He can be reached at *barry.harvey@analog.com*.