Efficient Metastability Characterization for Schmitt-Triggers

Jürgen Maier and Andreas Steininger
TU Wien, 1040 Vienna, Austria
{jmaier, steininger}@ecs.tuwien.ac.at

Abstract—Despite their attractiveness as metastability filters, Schmitt-Triggers can suffer from metastability themselves. Therefore, in the selection or construction of a suitable Schmitt-Trigger implementation, it is a necessity to accurately determine the metastable behavior. Only then one is able to compare different designs and thus guide proper optimizations, and only then one can assess the potential for residual metastable upsets. However, while the state of the art provides a lot of research and practical characterization approaches for flip-flops, comparatively little is known about Schmitt-Trigger characterization. Unlike the flip-flop with its single metastable point, the Schmitt-Trigger exhibits a whole range of metastable points depending on the input voltage. Thus the task of characterization gets much more challenging.

In this paper we present different approaches to determine the metastable behavior of Schmitt-Triggers using novel methods and mechanisms. We compare their accuracy and runtime by applying them to three common circuit implementations. The achieved results are then used to reason about the metastable behavior of the chosen designs which turns out to be problematic in some cases. Overall the approaches proposed in this paper are generic and can be extended beyond the Schmitt-Trigger, i.e., to efficiently characterize metastable states in other circuits as well.

Index Terms—Schmitt Trigger, Metastability Characterization, SPICE

I. Introduction

To use digital abstraction in electronic circuits we have to "digitize" an essentially analog input, i.e. either assign logic HI or LO depending on whether it is above or below a threshold. In order to prevent oscillation of the output due to noise in case of an input voltage close to the threshold – like in case of a comparator circuit –, the Schmitt-Trigger (S/T) uses a higher threshold for rising transitions than for falling ones, leading to a hysteresis (blue lines in Fig. 1). This, however, directly translates into a dependence of the threshold on the current output state, which, in turn, implies a positive feedback from the output to the input. As a consequence, the S/T must be susceptible to metastability. This intuitive argument has been more formally supported by Marino [1] already, and more recently Steininger et al. [2] have detailed several practically relevant scenarios where metastability may occur and where it may not. While there exist analytic solutions to calculate certain properties such as the threshold voltages [3], none have been presented so far regarding metastability.
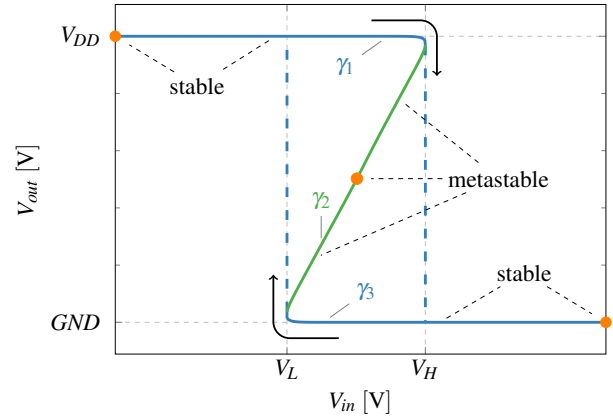
Fig. 1: Stable and metastable states of a latch (dots) and an S/T (lines) forming the characteristic z-shaped curve.

Actually S/T metastability is detrimental to its popular use for "cleaning" noisy input signals, or conditioning metastable outputs produced by other elements. Therefore it is crucial to fully characterize the metastable behavior of an S/T and, in the ideal case, estimate a mean time between (metastable) upsets (MTBU), as it is common with metastability in flip-flops. The latter has been well researched since the seminal work by Kinniment et al. [4], Chaney et al. [5], and Veendrick [6]. However, as it turns out, the S/T case substantially differs by the fact that its input remains connected to the positive feedback loop all the time, which ultimately results in the S/T exhibiting a whole range of metastable voltages $V_M(V_{in})$ rather than just one as in the flip-flop case. This makes the analysis and characterization way more complicated, and, unfortunately, hardly any results are available, apart from the mentioned papers [1] and [2].

Before it is possible to analyze the traces and probabilities to enter and leave these metastable states, and thus achieve a similar expression like the MTBU of a flip-flop, we need an efficient and accurate approach to characterize a given implementation. In [2] the authors used transient analog simulations to search for the metastable values, a very time consuming procedure which includes lots of manual steps yet. In addition they solely showed their analysis on a single circuit, a six-transistor implementation for a 65 nm technology. Therefore it is not clear whether the results are transferable to other implementations and technologies as well.
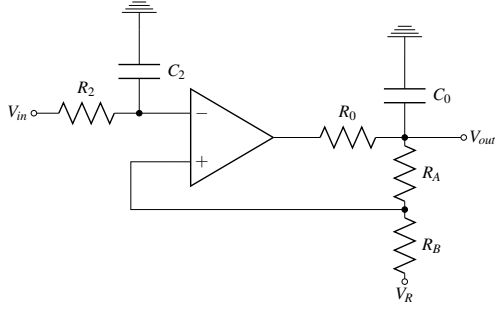
Fig. 2: Schmitt Trigger implementation studied in [1].

Fig. 3: Phase diagram of the S/T inspired by Marino [1]

Contribution: While we cannot solve all the open problems mentioned above in this paper, we extend the work by Steininger et al. in [2] by presenting and critically analyzing different approaches to simulate/evaluate the metastable behavior of an S/T. This not only includes the (meta-)stable states but also the behavior in their surrounding. These data are of interest when investigating more advanced features such as the overall probability to enter metastability or how quickly it is resolved. More specifically, we

- derive a more fine grained map (compared to [2]) of the output derivative V'_{out} over the $V_{in}-V_{out}$ plane, which we use as basis for more accurate estimations and analyses about the general behavior.
- determine all stable states, which partly lie in the undefined voltage range[1] for certain implementations making metastable behavior very easily reachable.
- carry out a preliminary analysis on metastability resolution, which turns out be shaped exponentially (comparable to the flip-flop) but with varying parameter $\tau(V_{in})$.
- introduce a novel method, which is not limited to S/Ts, that makes stable points metastable and vice versa.
- exploit plain *DC* analyses to determine metastability.
- evaluate the single approaches by characterizing three common implementations and comparing the results among each other and with the analytic results from [1].

This paper is organized as follows: In section II we briefly review metastability of S/Ts followed by a description of the proposed characterization methods in section III. Results and discussion for three common implementations are shown in section IV which is followed by a conclusion and an outlook to future research possibilities in section V.

II. BACKGROUND

Metastability has been well researched for latches, formed by cross-coupled inverters, since the seminal work by Kinniment et al. [4] and Veendrick [6]. For these elements there is even an equation for the mean time between metastable upsets (MTBU) available which relies, besides others, on the metastable resolution constant τ_C. The amount of stable (two)

and metastable (one) states is very small which is a direct consequence of the decoupled input. Fig. 1 shows the single states (dots), where V_{in} represents the value inside the loop.

A. Schmitt-Trigger Metastability

Unfortunately the situation is much more complicated for the S/T since the input remains connected continuously and hence has to be considered as well. For this purpose Marino [1] modeled the S/T by a properly wired OpAmp (shown in Fig. 2) and carried out analytic considerations. He used the phase diagram (V'_{out} over the $V_{in}-V_{out}$ plane, as shown in Fig. 3 with A equal to the amplifier gain and M being the output saturation voltage) to divide the behavior in three different regions, where the output in each is governed by the following equations:

$$\text{Region 1:} \qquad \frac{dV_{out}}{dt} = V'_{out} = -\frac{1}{\tau_1}(V_{out} - \gamma_1) \quad (1)$$

$$\text{Region 2:} \qquad \frac{dV_{out}}{dt} = V'_{out} = \frac{1}{\tau_2}(V_{out} - \gamma_2) \quad (2)$$

$$\text{Region 3:} \qquad \frac{dV_{out}}{dt} = V'_{out} = -\frac{1}{\tau_3}(V_{out} - \gamma_3) \quad (3)$$

The functions γ_1 and γ_3 represent the stable states while γ_2, which connects the former, the metastable ones. In contrast to the latch there are now infinitely many (meta-)stable values ranging from the lower (*GND*) continuously to the upper (V_{DD}) supply voltage. As was shown in [2] with the proper input signal (exceeding the threshold and then steering back) any of these values can be reached and held forever.

B. Schmitt-Trigger Metastability Characterization

The phase diagram as proposed by Marino is like a finger print of a Schmitt-Trigger implementation and helps the designer to understand and optimize the circuit. Therefore, in this paper we are searching for ways to determine the phase

---

[1]We call the voltage range above a well-defined LO and below a well-defined HI, according to the logic specification, undefined. In a properly functioning (metastability-free) circuit this range is crossed by steep transitions only.
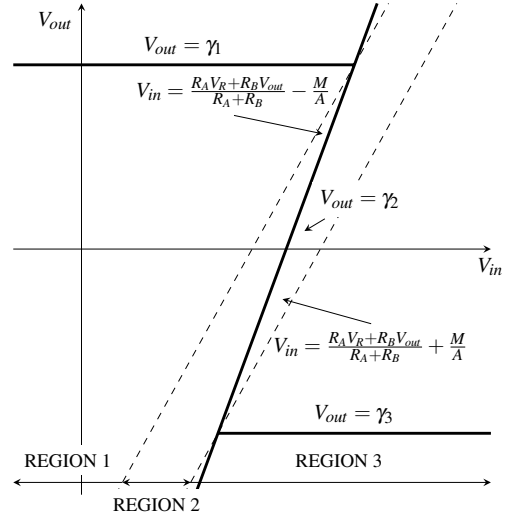
diagram in a fast, simple and yet accurate fashion for state-of-the-art implementations. As analytic considerations are based on certain abstractions, which are a good way to recognize dependencies however lack accuracy especially for modern technologies, we decided to base our analysis on *SPICE* simulations.

1) Stable States (hyst): Let us first focus on γ_1 and γ_3. These are very easy to achieve by starting two *DC* analyses, one sweeping V_{in} from *GND* to V_{DD} and one in the opposite direction. The threshold voltages, V_L and V_H, are easily recognized, as a small change on V_{in} leads to a major jump on V_{out}. Please note that, in contrast to the analysis of Marino, γ_1 and γ_3 are neither constant functions, nor straight lines in real circuits. Instead the stable values start to deviate from (*GND*/V_{DD}) when the threshold voltage is approached (cp. Fig. 1). For certain implementations this change is substantial, as we will show in Section IV, and thus has to be carefully analyzed. This is even more important as these states are actually stable and thus much easier to reach than metastable ones on γ_2, i.e., simply by a rising input stopping at a specific value (cp. [2]).

2) Metastable States: Far more interesting for us is however γ_2. Simply connecting γ_1 and γ_3 by a straight line, as derived by Marino, yields a first approximation, for more accurate result we have to resort however to more advanced methods. Luckily metastable states can be uniquely identified by checking for $V'_{out} = 0$, a property that all points on γ_1, γ_2 and γ_3 share. This immediately follows from the fact that one can stay infinitely long in perfect metastability and of course stable states.

Steiniger et al. [2] used transient analysis for this purpose. In detail they observed for a pair of $\hat{V}_{in}$ and $\hat{V}_{out}$ if $V_{out}(t)$ in- or decreased during a simulation run. Based on the result they implemented a binary search algorithm for the value of $\hat{V}_{out}$ in the next simulation until the desired accuracy was achieved. This procedure was then repeated for numerous values of V_{in} along γ_2. Since this is a very time consuming task we searched for more ingenious solutions and even found several alternatives, which we will describe in the following.

III. METHODS FOR OBTAINING γ_2

Efficient and precise ways to determine the (metastable) characteristics of a Schmitt-Trigger are key for reliability predictions or comparisons between different implementations. In the following we will elaborate several approaches to determine the metastable states (γ_2), as this is currently the biggest challenge. For accurate simulations we resorted to *HSPICE* using a $28\,\text{nm}$ UMC technology library. Comparisons with an older $65\,\text{nm}$ technology showed no qualitative difference so we restrict ourselves to presenting the former in this paper. Our circuit model is pre-layout, but we consider a capacitive output load of $C_L = 2\,\text{fF}$ in our *AC* analyses.

A. Static Analysis of Grid Points (map)

Recall from Section II-B2 that all (meta-)stable states share the property $V'_{out} = 0$. As a first approach we can cover the
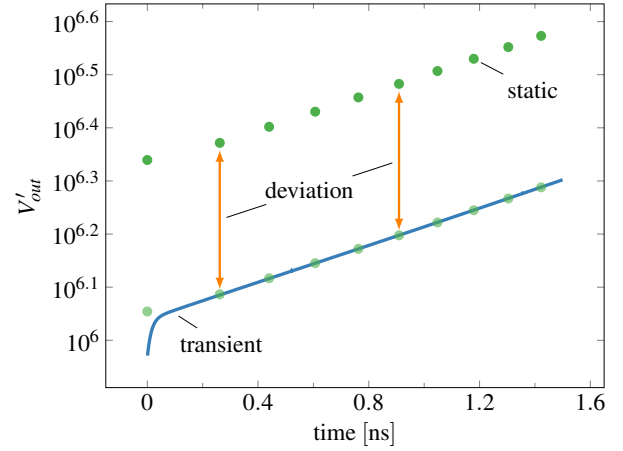

Fig. 4: Output derivative for transient and static simulations. The deviation stays constant as the projection (opaque green dots) shows.

V_{in}-V_{out} plane with a regular grid and determine V'_{out} for each grid point to find where it gets (close to) zero. Albeit this initially appears quite untargeted and laborious, it provides us with a map that will turn out valuable for analyzing the resolution behavior later on.

Listing 1: deriving I_{out} in V_{in}-V_{out} plane in *SPICE*
```
.DC VIN 0 supp width SWEEP VOUT LIN count 0 supp
.PROBE DC I(VOUT)
```

Our approach uses built-in commands from *SPICE* only, as detailed in Listing 1: We sweep V_{in} from 0 to V_{DD} (*supp*) in steps (*width*) corresponding to the grid. In the same way V_{out} is swept (*count* = #steps). For this purpose we replace the load capacitance by a voltage source and actually measure the current through the latter (visible in the second code line). In comparison to the approach from [2], where the authors performed transient analysis and picked V'_{out}, this is considerably faster but serves the same purpose, albeit we get I_{out} as a result instead. To compare the results of static and transient simulation (see Section III-B) in Fig. 4 we can use the transformation $V'_{out} \cdot C_L = I_{out}$, which leads however to a constant deviation. That discrepancy is a result of the internal capacitance of the S/T which we determined to be $1.854\,\text{fF}$ and whose value stays constant even for varying values of C_L. In the following we will therefore use $\hat{C}_L = C_L + 1.854\,\text{fF}$ for transformations between V'_{out} and I_{out}.

Note that our proposed *DC* analysis does not reflect potential dynamic effects: In real circuits V'_{in} most certainly has an effect on V'_{out} through coupling capacitances. However, in our view this only restricts the possible paths a metastable state can be reached, but not the actual value itself, since all (meta-)stable states are per definition static, i.e., $V'_{out} = 0$. Therefore we consider it fundamental to determine the static, general case in the first place – and this is what the *DC* analysis properly does.

Obviously we won't be lucky enough to hit $I_{out} = 0$ (or $V'_{out} = 0$) exactly this way, but those pairs of grid points

between which I_{out} changes its sign already confine γ_2. In a first step contour plots can be used to draw an (interpolated) contour line at $I_{out} = 0$, i.e., at the (meta-)stable line. Furthermore the map may serve as starting point for more precise estimations.

B. Transient Estimation (`expAC`)

When starting transient simulations in the grid points confining γ_2 (changing sign of I_{out}) one observes traces that are nearly perfect exponentials (see Fig. 4), as predicted by theory [1]. This has major implications. Firstly, it suggests that the resolution behavior is comparable to the flip-flop, with the main difference, however, that the resolution time constant τ is now not unique but varies with V_{in}. This will become apparent in Section IV.

Secondly, it gives us the possibility to infer the metastable voltage by recording just a short piece. Assume we start in an arbitrary point $(\hat{V}_{in}, \hat{V}_{out})$ and observe the output, i.e., $V_{out}(t)$ and $V'_{out}(t)$ in the course of the simulation. Since we assume the trace to be exponential we get the following relations:

$$V_{out} = V_M \pm V_x \cdot \exp\left(\frac{t - \hat{t}}{\tau}\right) \tag{4}$$

$$V'_{out} = \pm\frac{1}{\tau} \cdot V_x \cdot \exp\left(\frac{t - \hat{t}}{\tau}\right) \tag{5}$$

where $\hat{t}$ denotes the unknown time shift between our measurement and the actual resolution curve and $V_x > 0$ the unknown scaling factor of the exponential. Let us now apply the natural logarithm on $|V'_{out}|$ leading to

$$\ln(|V'_{out}|) = \ln\left[\frac{1}{\tau} \cdot V_x \cdot \exp\left(-\frac{\hat{t}}{\tau}\right)\right] + \frac{t}{\tau}$$

By applying a linear fit to our simulation results we can easily determine the value of τ, which is inversely proportional to the slope. Going back to Equation (4) and expressing the exponential term by V'_{out} finally yields:

$$V_M = V_{out} \mp \tau \cdot V'_{out}$$

V_M is obtained by plugging in a single pair of measured values for V_{out} and V'_{out}. Note that we actually do not have to know the absolute time (or the parameters $\hat{t}$ and V_x, respectively): We used the time and value difference between some measured values of V'_{out} to determine τ, while a consistent pair of (V_{out}, V'_{out}) sufficed to finally obtain V_M.

For valid results, two aspects have to be considered:

- Initially, V'_{out} observed by the transient analysis changes disproportionately (cp. Fig. 4) leading to a bad fitting. As a consequence the first samples have to be removed.
- At some point the waveform changes from "leaving metastability" (increasing $|V'_{out}|$) to "approaching stable value" (decreasing $|V'_{out}|$). By choosing initial conditions and simulation time it must assured that this point is never reached by the simulation.
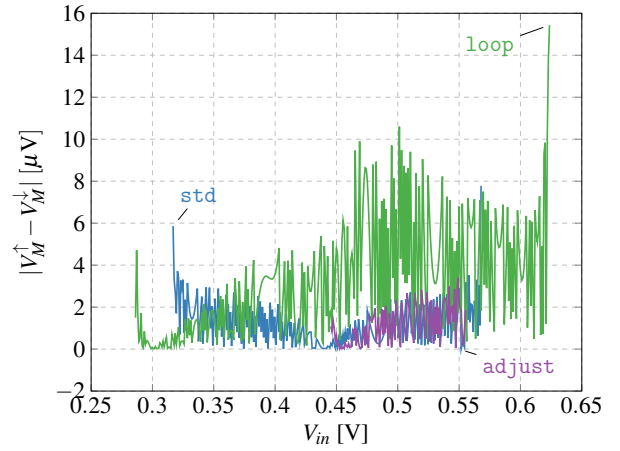

Fig. 5: Absolute deviation between V_M predictions based on the resolution direction for different circuit implementations (cp. Section IV).

C. Static Estimation (`expDC`)

In accordance to the transient measurements of V'_{out} we also see an exponential growth of the static I_{out} as we follow a resolution trajectory, which is reasonable as they only differ by a constant factor. Therefore it is quite natural to apply the same estimations as before also on I_{out}. One can rewrite the expressions in Equation (5) to

$$V'_{out} = \frac{I_{out}}{\hat{C}_L} = \frac{1}{\tau} \cdot (V_{out} - V_M) \tag{6}$$

$\hat{C}_L/\tau$ can be achieved by the slope of I_{out} over V_{out}, i.e., by fitting the data from the I_{out} map. Plugging an arbitrary V_{out}, the corresponding I_{out} and the known $\hat{C}_L/\tau$ into Equation (6) finally yields the metastable voltage V_M.

Compared to the transient analysis the calculations on I_{out} are far easier to execute and thus less prone to errors. Both provide however the possibility to improve the limited accuracy of tools (due to numerical issues).

Please note that for both, `expAC` and `expDC`, τ and V_M can be determined twice: either for traces resolving the metastability to *GND* or for such resolving to V_{DD}. Ideally both would render the same results. In reality, however, we get slightly different values for τ (which may indeed have a physical reason), and slight deviations in V_M (most likely due to numerical issues, which have an exponential effect). For the latter Fig. 5 shows the difference between the predictions for the curve resolving to V_{DD} ($V_M^\uparrow$) and *GND* ($V_M^\downarrow$) for `expDC`, whereat results for `expAC` are slightly worse. For better accuracy we therefore determine V_M (using the corresponding τ) as the point where the two resolution curves meet.

D. Binary Search (`binary`)

A more pragmatic approach is `binary`, where we sweep V_{in} from V_L to V_H, and for each value a binary search is performed to find a fitting value of V_{out}, i.e., where I_{out} is

zero. This is very similar to the approaches in [2], [7] with the difference that we use the static current instead of the transient output derivative.

To our advantage *SPICE* has a mechanism to run a binary search built-in (called "Bisection") which simplifies the task a lot. The corresponding code is shown in Listing 2. The first line states that we want to bisect, and at most 40 steps shall be carried out. Note that this narrows down the initial interval by a factor of 2^{40}, so most of the time the algorithm quits earlier, as the demanded accuracy is reached first.

Listing 2: bisection in *SPICE*

```
.model optMod1 OPT METHOD=BISECTION ITROPT=40
.param outVal=optFunc1(vdd/2, vout_VL, vout_VH)
.DC VIN inVal inVal 1 SWEEP OPTIMIZE=optFunc1
+ RESULTS=optMeasure MODEL=optMod1
```

The second line sets the parameter *outVal* which determines the output voltage and the range within which it shall be swept. We used here the value of V_{out} at $V_{in} = \{V_L, V_H\}$ as we have to be sure to avoid the stable states. The search itself always starts at $V_{DD}/2$. The last line finally launches the *DC* analysis for the input voltage in the range [*inVal,inVal*], meaning that we execute this search for each value of V_{in} separately, since we were not able to convince *SPICE* to do that automatically.

E. Metastability Inversion (`inversion`)

The reason why metastable states are hard to characterize is the fact that it is close to impossible to actually reach them even in simulations, since, by definition, the system consistently works towards leaving them. A good physical analogy is the inverted pendulum. Stable states, in contrast, are naturally assumed by the system. This observation suggests reduced characterization effort if the cases could be reverted, i.e., stable points are made metastable and vice versa.

With this in mind, let us model the system in the metastable state as an (output) current source that is controlled by the output voltage, namely

$$I_{out} = K \cdot (V_{out} - V_M)$$

A positive current charges the load capacitance, i.e., increases V_{out}, which in turn increases the current even more. For the voltage gradient we get

$$V'_{out} = \frac{I_{out}}{\hat{C}_L} = \frac{K}{\hat{C}_L} \cdot (V_{out} - V_M)$$

which yields an exponential function for V_{out}, more specifically, with $K > 0$ an exponentially growing one. We can, however, invert the sign of K by connecting a current source I_L to the output, whose current is controlled by I_{out}, i.e.

$$I_L = p \cdot I_{out}.$$

Mathematically, this changes the current into the capacitance from I_{out} to $(1 - p) \cdot I_{out}$, and we get

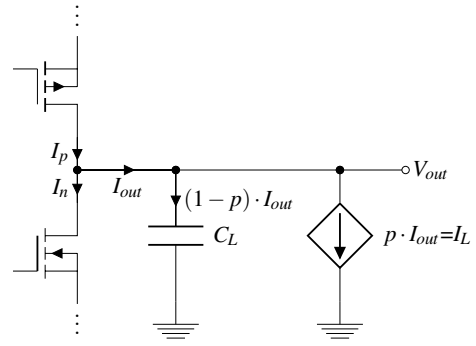$$V'_{out} = \frac{(1-p) \cdot I_{out}}{\hat{C}_L} = \frac{(1-p) \cdot K}{\hat{C}_L} \cdot (V_{out} - V_M)$$

Fig. 6: Circuit setup to invert (meta-)stable points.

which, for $p > 1$, yields a decaying exponential function and hence a stable solution.

Intuitively spoken, any positive I_{out} is overcompensated by I_L, such that C_L, instead of being loaded by I_{out}, now gets discharged through that portion of I_L that exceeds I_{out}. Therefore ultimately V_{out} is reduced, and, as a consequence, I_{out} as well. This naturally drives I_{out} towards zero, which represents exactly the state that is normally metastable. Fig. 6 shows the resulting circuit for $I_{out} > 0$.

Overall the added current source serves as a proportional controller (for the current) that stabilizes the "inverted pendulum". Implementing such an element in *SPICE*, and similarly in other simulation suites, is straightforward. The only challenge left is to choose a proper value for p, which is a very delicate task: In essence, p contributes to the gain in the control loop and is hence, according to control theory, crucial for stability. Choosing p too low (very close to 1) leads to slow stabilization and consequently long simulation times. Choosing p too high, in contrast, causes oscillating behavior.

In order to come up with a guideline for a reasonable choice of p we have used the example of the implementation shown in Fig. 12b that employs the same inverter loop as the latch. Consequently we could build on the model from Veendrick [6]. By extending the latter with our controlled current source we can approximate the behavior of the dynamic system comprising S/T and controller. From that it turns out that the right choice for p is

$$p = 2A^2 \left(1 + \sqrt{1 - \frac{1}{A^2}} \right) \approx 1 + \frac{1}{4A^2} \qquad (7)$$

with A^2 being the product of the (*DC*) gains of each of the two inverters in the model. Note, however, that the inverter gain is not constant, so the "ideal" p will change as we move along γ_2. In our experiments we chose the highest gain, i.e. the one in the middle of the inverter's transfer curve, which yielded very useful values for p. Luckily *SPICE* supports the determination of the loop gain, so A^2 can be obtained by the code piece shown in Listing 3, whereat we connected a constant voltage source to the input and set the initial value
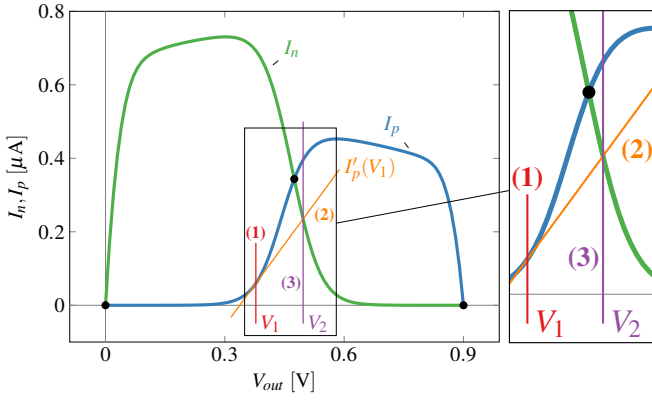
Fig. 7: Application of the *Newton-Raphson* algorithm to find a stable value (marked by black dot) of V_{out} for fixed V_{in}.

Fig. 8: Flip-flop half used for *DC* metastability analysis.

of V_{out} appropriately. For the complete code refer to the tool described in Section IV.

To get as close as possible to the *DC* case we measure the gain at very low frequencies (1×10^{-5} Hz), which is shown in line 3 in the listing. Please note that different S/T implementations will lead to different feedback models, such that Eq. (7) cannot be applied directly in these cases.

Listing 3: measuring loop gain in *SPICE*

```
.ac dec '10' '0' '10'
.lstb mode=single vsource=vlstb
.measure lstb gain loop_gain_at_minifreq
```

The method of metastability inversion is not restricted to S/T but can easily be extended to also determine the metastable point of other circuits. We verified this by introducing a current source into a "latch-style" inverter loop (i.e. with unconnected input) which then quickly approached its metastable value.

F. DC Analysis (`static`)

With metastability inversion transforming metastable states into stable ones we could use *DC* analysis to collect the values on γ_2. While this works well, we discovered in the course of our research, that already *DC* simulations on the unmodified implementation are capable of delivering the metastable values. The reason for that is the *Newton-Raphson* algorithm which *SPICE* uses to determine the *DC* operating point [8]. Let us take a closer look at this procedure: Assume that we fixed V_{in} in the metastable region and want to determine a stable V_{out} (we already know that there are exactly three possibilities). For a stable value the current coming from the p-stack (I_p) and the one flowing into the n-stack (I_n) at the output have to be equal. Determining those currents for various values of V_{out} gives us traces like those shown in Fig. 7. The three stable states marked by black dots are clearly visible. To start the algorithm we have to make a guess, say we pick V_1. The next task would be (1) to determine the derivative of I_p in this point, i.e, $I'_p(V_1)$, (2) find the crossing point of the latter with I_n and finally (3) restart the procedure with $V_{out} = V_2$, i.e., the value in the crossing point. If we start close enough
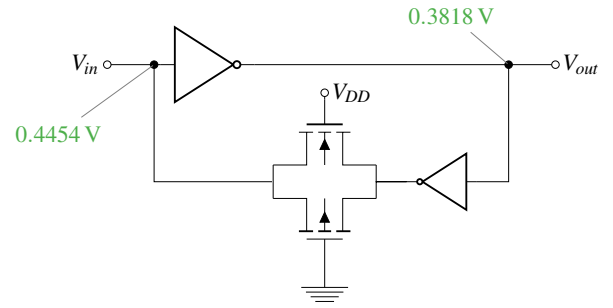
to V_M the algorithm will approach it automatically. It can be seen that a deviation of several tens of millivolts can be tolerated for the initial guess, an accuracy easily achievable by connecting the last stable values of γ_1 and γ_3 by a straight line.

After the first metastable value was found others follow quickly as the current value of V_M serves as starting point for the search on the next one, whose value does not differ much and is thus found very rapidly.

For that reason the most important task is to get a good initial guess for the first value. If we are too far off, we will find the stable state and thus end up with part of the hysteresis that we already know. Assume we start in the point on γ_2 that is closest to γ_3. Due to this close proximity we can infer that for that choice the metastable value will be close to the peak value for V_{out} on γ_3. Therefore, if we choose a slightly higher value for V_{out} we end up with a very good initial guess, since our starting point is closer to γ_2 than to γ_3. The amount of increase is uncritical and can actually be chosen rather big, in our experiments up to $V_{DD}/4$.

We varified our approach also on a flip-flop half, i.e., a loop of asymmetric inverters (width ratio $1/10$) with a transmission gate (see Fig. 8) in a similar fashion. In detail the input values to the first and second inverter were set to $V_{DD}/2 = 0.45$ V and then *SPICE* was asked to calculate the operation point. As a result we got $V_{in} = 0.4454$ V and $V_{out} = 0.3818$ V.

IV. EVALUATION

Provided that an appropriate *SPICE* description of the circuit is available, the complete characterization process can be carried out fully autonomous. Thus we implemented all the approaches presented in the previous section in a small tool which is publicly available[2]. For the simulations presented in the following, we used a 28 nm UMC technology library ($V_{DD} = 0.9$ V) and determined the (meta-)stable states for 1000 equally spaced values of V_{in}.

The aim of these simulations is twofold:

- We want to evaluate and compare the presented methods in a practical application. To this end we apply them for characterizing three different S/Ts: a) the standard 6T implementation (`std`) b) an inverter loop [9] (`loop`) and

[2]*https://github.com/jmaier0/meat*

(a) `map`

(b) V_M predictions vs. `binary`

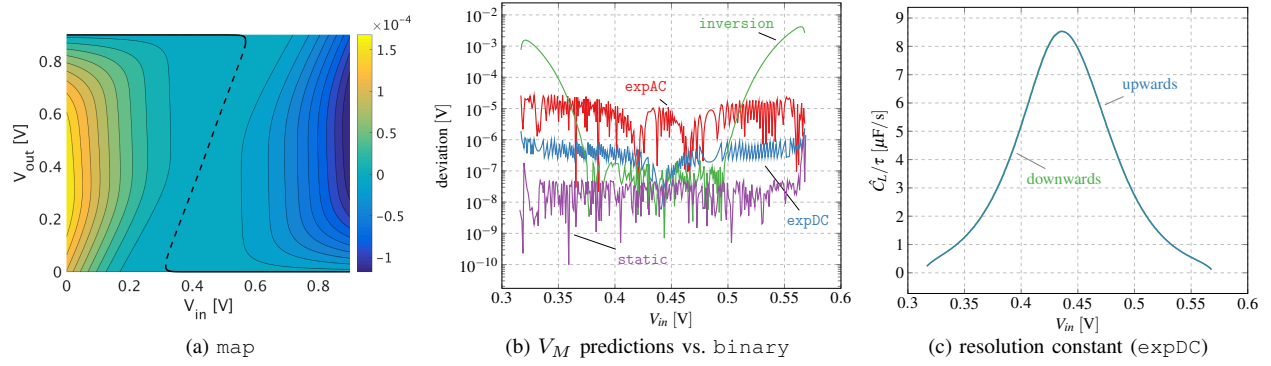(c) resolution constant (`expDC`)

Fig. 9: Simulation results for `std`.

c) an adjustable hysteresis one [10] (`adjust`), as other circuits in literature are heavily based on these.

- The analysis and comparison of the S/T implementations is as such important. In particular it is interesting to see how far the behaviors differ among each other and also from the theoretical results [1].

A. General Remarks

In principle, the *resolution* of all presented methods can be made as high as desired. In practice there exist however limitations such as the finite simulator precision (number format), the required run time and the available output file formats. The latter caused heavy problems as we only managed to export results with 7 positions after the decimal point (solely for `binary` we achieved 10 by using a different method).

The *accuracy* of the results is somewhat limited by the achieved resolution and the assumption of a perfectly exponential resolution trajectory for `expDC` and `expAC`, the simulation time for `inversion`, and in general by the accuracy of the circuit and transistor models underlying the SPICE simulations (which we will neglect from now on, as this issue is immanent to all simulation approaches). To verify that the metastable states obtained through the various methods are indeed accurate, we started transient simulations in each of

Fig. 10: Deviation of V_{out} after 200 ps.

them and measured the output deviation after 200 ps. Fig. 10 shows the results for `std`. As one can see `binary`, which predicts V_M with a resolution of ± 5 nV, has clearly the lowest deviation and thus is the most accurate. Therefore we will use `binary` as golden reference for all further analyses, rather than continuing with the computationally intensive transient simulations for accuracy validation.

Where the single methods differ, however, is the *effort* one needs to invest. To quantify that, we try to approach the metastable value reasonably close (± 10 µV) and then compare the required run times. Table I gives an overview of the achieved results. As the hysteresis differs among the implementations, the number of grid points between V_L and V_H (metastable grid points), and thus the run time, varies. Nevertheless, for the same circuit a comparison among different methods is still valid.

TABLE I: Overview of simulation times

	simulation time [s]		
method	std	loop	adjust
metastable grid points	282	378	125
`hyst`	1.825	2.012	1.842
`binary`	232.279	349.224	104.585
`map`	310.826	318.153	362.692
`expAC`	526.149	806.603	233.369
`expDC`	2.047	2.481	1.310
`inversion`	901.588	1552.542	447.089
`static`	0.850	0.939	0.853

Since the target of ± 10 µV was quite deliberately chosen, let us review the impact of target accuracy on the run time for each method:

binary The amount of binary steps has hardly any impact on the runtime. We experienced a reduction by only 10 % when switching from 40 to 20 iterations with a accuracy loss of four orders of magnitude.

map The run time scales linearly with the amount of points which is quite natural as each one is determined by a separate *DC* analysis.

expAC & expDC The accuracy increases only with a more accurate `map`. Apart from that, their simulation time is constant.

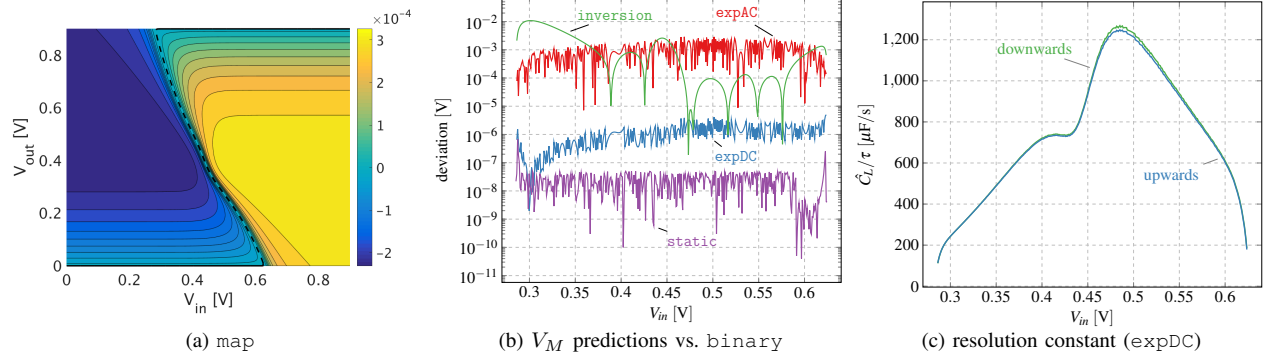(a) map   (b) V_M predictions vs. `binary`   (c) resolution constant (`expDC`)

Fig. 11: Simulation results for `loop`.

inversion The chosen gain p of the current source has a high impact on the simulation time. Higher gain decreases run time but also yields more instabilities, e.g. oscillations. Increasing the simulation period helps to resolve these, however the run time increases almost by the same factor.

static The simulation time is constant, and always the highest possible accuracy is delivered. In our setting, however, the output data format limited the attainable (exportable) accuracy.

Clearly there is still room for optimizations such as

- determining I_{out} solely for grid points close to the metastable line for `map`
- better choice of the gain p in `inversion` or even improving the whole control loop dynamics by an integral or differential part
- non-uniform simulation time for `inversion`

which we left for future research. For this reason the presented run times shall only be used to get an intuition how long the characterization approximately takes.

Since we use both transient and *DC* analyses overall we experienced that the former are much harder to handle, as more parameters have to be defined, primarily the time period of simulation. In addition further complications such as cutting the first part of the simulation in `expAC` or an appropriate gain for the current source in `inversion` have to be overcome. In total *DC* analyses achieve better results in shorter time with simpler methods. During our research we even realized that most methods, in particular `static`, are also applicable to other problems, such as determining the metastable value of a flip-flop. We consider this more a lucky coincidence than a designed feature as it is a side effect of the utilized *Newton-Raphson* algorithm.

B. Standard Implementation (`std`)

To compare our results with those from Steininger et al. we first analyze the implementation used in [2]. The transistor level circuit is shown in Fig. 12a.

The achieved (meta-)stable line (shown in Fig. 9a, γ_1 and γ_3 solid, γ_2 dashed) fits very well to the results published in [2].

Thus we conclude that our tool works as expected. The same figure also shows a heat map of the output current I_{out} in the V_{in}-V_{out} plane. Please note that the I_{out}-spacing of the contour lines is linear. This means that close to the metastable line I_{out} changes only moderately, as expected from the exponential resolution trajectories predicted by theory.

In contrast to the calculations of Marino [1], however, whose V'_{out} only depends on the distance to the final, stable state, our results show additional dependencies of I_{out}. This can be seen very clearly by observing its maximum and minimum, which both are near $V_{DD}/2$.

Fig. 9b shows the (absolute) deviation of the predicted metastable voltages from the ones of `binary`. Please note that with the finite export number format deviations below 5×10^{-8} V were out of reach. From our result we therefore deduce that `static` and `binary` are capable of delivering the same accuracy.

Finally 9c shows the (inverse of the) resolution constant $\hat{C}_L/\tau$ determined by `expDC`. It significantly varies with V_{in}, with the biggest (best) value in the middle and the lowest at points close to the stable states, meaning that the latter are left more slowly. The values achieved for the up- and down-resolving waveform are shown separately, but the graphs nicely overlap.

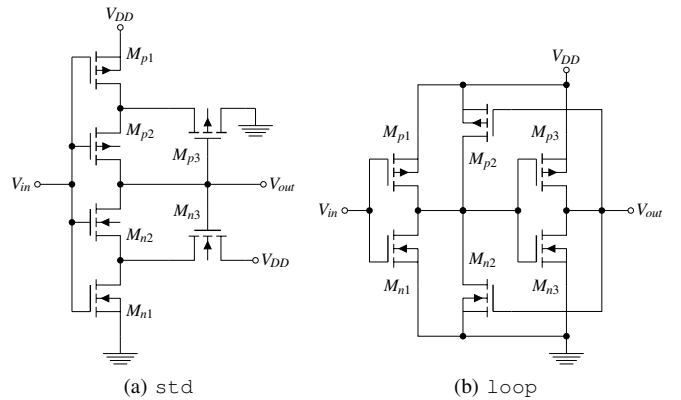The significant change of τ raises the interesting question

(a) `std`   (b) `loop`

Fig. 12: Transistor level circuit implementations.

(a) `map`

(b) V_M predictions vs. `binary`
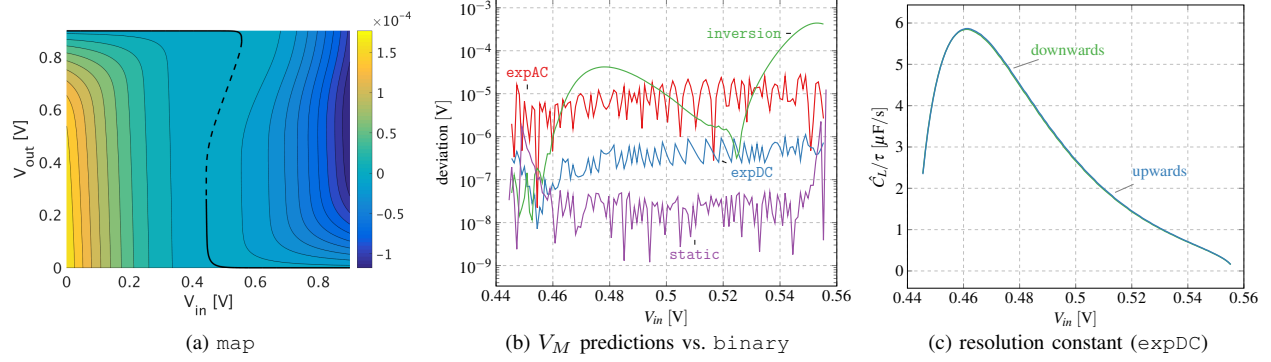
(c) resolution constant (`expDC`)

Fig. 13: Simulation results for `adjust`.

whether the quick resolution from the middle will cross the far distance to the saturation faster than the slow one from the borders that only has a short distance to cross. In consequence one might determine a worst starting point from which resolution takes the longest. However, the answer heavily depends on what is considered the threshold for "resolved", and, most importantly, on how deep the initial metastability was (recall that resolution time is essentially unbounded).
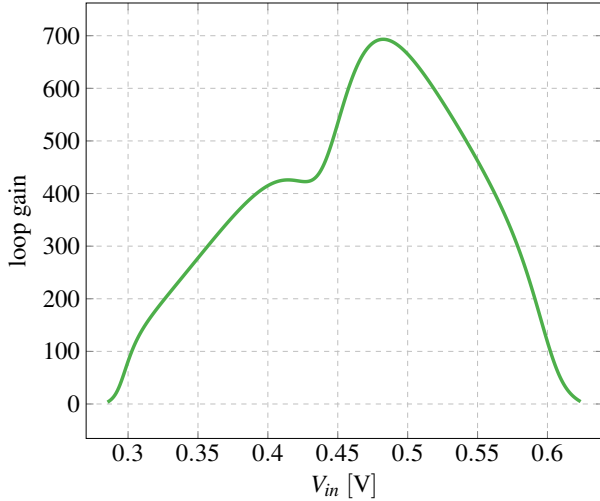

Fig. 14: Memory loop gain of `loop`.

C. Inverter Loop (`loop`)

The second circuit we investigate (transistor level implementation see Fig. 12b) is essentially a latch whose input can not be decoupled any more, i.e., a plain inverter loop (preceded by an additional inverter). The hysteresis is defined by the relation between the driving strength of the first inverter (transistors M_{p1} and M_{n1}) and the weak feed back one (M_{p2} and M_{n2}). For the latter we reduced the width to one tenth.

The I_{out} map (see Fig. 11a) significantly differs from the one of `std`. First of all the contour lines are horizontal, much more like the prediction made by Marino in [1]. Secondly the current changes much more rapidly than before, which also leads to much higher values for $\hat{C}_L/\tau$ (three orders of

magnitude, see Fig. 11c), i.e., metastability is resolved much quicker. According to Eq. (7) p can be calculated using the memory loop gain shown in Fig. 14. Picking the highest value ($A^2 = 700$) results in $p = 1.000357$ which was confirmed to be a suitable value by simulations. Increasing p however quickly leads to oscillations of V_{out}, whereat metastable points with higher loop amplification become instable earlier.

D. Adjustable Hysteresis (`adjust`)

In some applications it is important to adjust the hysteresis of the S/T during operation. One circuit that can be used for this purpose is shown in Fig. 15. The value V_B on an additional input alters the position and width of the hysteresis. In our simulations we used $V_B = V_{DD}$ as in this case the hysteresis is the widest and thus the most stable one. Anyway, similar behavior could be observed for other choices of V_B as well.

The first remarkable thing in Fig. 13a is the relatively large peak value of V_{out} on γ_3. It reaches up to about $0.3\,\text{V}$ which is one third of the supply voltage and almost certainly in the forbidden region. Please recall (see Section II-B1) that those states can be easily reached by a ramp stopping at a defined value, implying that resilience against metastability is weakened a lot. The `map` furthermore reveals nearly vertical
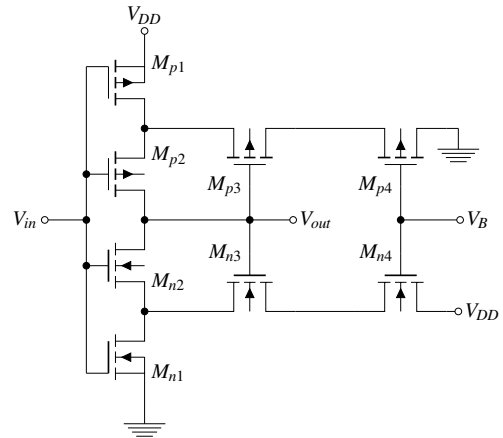

Fig. 15: Circuit implementation of `adjust`.

contour lines in the left half of the plot. This suggests, that in this region the output slope is constant, i.e., we get ramps at the output. Very surprisingly, I_{out} does not seem to depend on the distance to the stable state at all, as it was calculated by Marino [1].

The values of $\hat{C}_L/\tau$ (Fig. 13c) are comparable to `std` . However, since the stable states from below reach far into the mid-voltage region the graph is not symmetric any more.

V. CONCLUSION AND FUTURE WORK

In this paper we have presented several ways to characterize the metastable behavior of a Schmitt-Trigger including estimations that increase accuracy beyond numerical precision, a novel method to convert metastable states into stable ones, and plain *DC* analysis that turned out to be already sufficient to accurately determine metastability. By applying them to three common CMOS implementations we not only verified that they work properly but were also able to compare them, giving an edge to *DC* rather than transient methods regarding accuracy and run time. At the same time simulation results revealed that the metastable behavior only partially follows the theoretical predictions made in the past.

For future work we want to use the results of this paper as starting point to derive an expression for estimating the reliability impact of metastable upsets in S/Ts, comparable to the MTBU formula in flip-flops.

REFERENCES

[1] L. R. Marino, "The Effect of Asynchronous Inputs on Sequential Network Reliability," *IEEE Transactions on Computers*, vol. 26, no. 11, pp. 1082–1090, 1977.

[2] A. Steininger, J. Maier, and R. Najvirt, "The metastable behavior of a schmitt-trigger," in *2016 22nd IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC)*, May 2016, pp. 57–64.

[3] B. L. Dokic, "Cmos and bicmos regenerative logic circuits," in *Cutting Edge Research in New Technologies*, C. Volosencu, Ed. Rijeka: IntechOpen, 2012, ch. 2. [Online]. Available: https://doi.org/10.5772/34934

[4] D. Kinniment and D. Edwards, "Circuit technology in a large computer system," in *Conference on Computers - Systems and Technology*, Oct 1972, pp. 441–450.

[5] T. J. Chaney and C. E. Molnar, "Anomalous behavior of synchronizer and arbiter circuits," *IEEE Transactions on Computers*, vol. C-22, no. 4, pp. 421–422, April 1973.

[6] H. J. Veendrick, "The behaviour of flip-flops used as synchronizers and prediction of their failure rate," *IEEE Journal of Solid-State Circuits*, vol. 15, no. 2, pp. 169–176, Apr 1980.

[7] S. Yang and M. Greenstreet, "Computing synchronizer failure probabilities," in *2007 Design, Automation Test in Europe Conference Exhibition*, April 2007, pp. 1–6.

[8] *HSPICE® User Guide: Basic Simulation and Analysis*, Synopsys®, June 2016, version L-2016.06.

[9] Z. V. Bundalo and B. L. Dokić, "Non-inverting regenerative cmos logic circuits," *Microelectronics Journal*, vol. 16, no. 5, pp. 5 – 17, 1985. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0026269285800021

[10] A. Pfister, "Novel cmos schmitt trigger with controllable hysteresis," *Electronics Letters*, vol. 28, no. 7, pp. 639–641, March 1992.