# Dynamic Effects in Asynchronous Circuits

Florian Deeg, Sebastian M. Sattler
Chair of Reliable Circuits and Systems
Friedrich-Alexander-University Erlangen-Nuremberg
Paul-Gordan-Str. 5, 91052 Erlangen, Germany
Email: {florian.deeg,sebastian.sattler}@fau.de

***Abstract*—The aim of this paper is to summarize the dynamic effects of asynchronous circuits again and to verify them using logical models and real basic examples. The effects of hazards and races are presented theoretically and formally, and a new formula for recognizing races is derived from this. Examples of how these formulas can be applied are then shown. The effects are made visible with the help of a specially developed circuit board and an oscilloscope.**

## I. Introduction

The automotive industry is an industry with very high requirements on safety and security, and, especially in the area of e-cars, with high demands on power-saving systems due to the increasing use of electrical systems for propulsion and electronic systems for control. These systems could be characterized as modular, distributed and discrete event systems. From 2019-2024, the automotive market is forecast to grow at 9.7%, the strongest CAGR (compound annual growth rate) of any end-use segment. [1]
Synchronous circuits are said to have a single point of failure and high power consumption due to their clock line. [4] For safety-critical circuits, autonomously operating subsystems that do not require a clock line could be developed, enabling the design of functionally safe and further energy-saving circuits and systems.
Asynchronous sensor signals in the vehicle could then be processed by such autonomous subsystems on an event-driven basis only when a signal occurs without a global clock that is always clocked. Asynchronous circuits are therefore of key importance. To understand the challenge of these designs, it is thus important to know the dynamic effects of asynchronous circuits and to avoid them by appropriate implementations and models.
If a function is not designed according to its structure, it is possible that the circuit will exhibit unpredictable behavior. Thus, errors such as hazards or races can occur in asynchronous circuits. For a better understanding, these errors are first theoretically reviewed, analyzed with example circuits and then implemented with suitable basic circuits for illustration by structure and analysis.

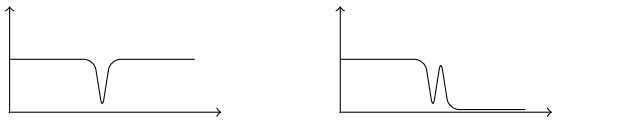
## II. Hazards

Hazards describe the possibility that a momentary fault may affect an electronic circuit. If a hazard actually appears at the output of a circuit, it is referred to as a hazardfehler. In principle, hazards are differentiated in their origin in function (time) and structure (space) and in their effect in static and dynamic. The descriptions of the derivation operations are taken from [5] and [6].

### A. Function Hazards

Function hazards are potential faults at the output of a circuit when at least two input quantities do not change simultaneously and an intermediate fault signal is generated by this malfunction.

*1) Static Function Hazards:* Static function hazards maintain their value $f(x)$ at the end while they briefly generate intermediate false signals, see Fig. 1a.



(a) Static 1-Hazard (b) Dynamic 1-0 Hazard

Fig. 1: Function Hazards

To derive a static function hazard, it must be ensured that both input assignments produce the same output, i.e. that its vectorial derivative according to Eq. (1) equals 0.

$$\left.\frac{\mathrm{d}f(x)}{\mathrm{d}x_\mathrm{p}}\right|_{X_\mathrm{p}} = f(X_\mathrm{p},x_\mathrm{q}) \nsim f(X_\mathrm{p} \nsim 1,x_\mathrm{q}) \text{ with } x=(x_\mathrm{p},x_\mathrm{q}) \tag{1}$$

The condition now is that there are input variations due to non-simultaneous switching that result in a different output signal, i.e., its varational derivative equals 1 according to Eq. (2).

$$\begin{gathered}\left.\frac{\Delta f(x)}{\Delta x_\mathrm{p}}\right|_{X_\mathrm{p}} = \bigvee_c f(X_\mathrm{p},x_\mathrm{q}) \nsim f(X_\mathrm{p} \nsim c,x_\mathrm{q}) \\ \text{with } x_\mathrm{p}=(x_1,x_0) \text{ and } c \in \{01,10,11\}\end{gathered} \tag{2}$$

*2) Dynamic Function Hazards:* Dynamic function hazards are potential false signals that can occur during a transition of an output signal if at least three input signals change their values, see Fig. 1b. To derive a dynamic function hazard, it must be ensured that both input assignments produce a different output, i.e. that its vectorial derivative according to Eq. (3) equals 1.

$$\left.\frac{\mathrm{d}f(x)}{\mathrm{d}x_\mathrm{p}}\right|_{X_\mathrm{p}} = 1 \text{ with } x=(x_\mathrm{p},x_\mathrm{q}) \tag{3}$$

The requirement now is that there are input variations due to non-simultaneous switching from $x_1$ to $x_4$ that lead to different output signals, so that $f(x_1) = \overline{f}(x_2) = f(x_3) = \overline{f}(x_4)$. Since a dynamic functional hazard combines two static functional hazards, it can also be calculated by a formula that is not presented in this paper.

### B. Structure Hazards

Structure hazards, as the name implies, describe the possibility of a short-time false signal due to differences in the time delay in a circuit, i.e. due to different delays between real paths of a circuit. To find a static structure hazard, it must be ensured that there is no function hazard whose varational derivative equals 0 according to Eq. (4).

$$\frac{\Delta f(x)}{\Delta x_\mathrm{p}} = 0 \tag{4}$$

Next, each individual path must be encoded as a variable in $a$ to set up the overall time model $f(a)$, which encodes each input path as a dimension (totzeitmodell). The condition now is that there are input variations when switching the vector, i.e. that its varation derivative according to Eq. (5) equals 1.

$$\frac{\Delta f(a)}{\Delta a_p} = 1 \text{ with } a = (a_\mathrm{p}, a_\mathrm{q}) \tag{5}$$

## III. Races

When at least two feedbacks compete with the output value of their states, so-called races are created. [6] These underlying structures, inverted with respect to each other, produce different solutions with respect to the physical conditions applied to the circuit. There are two types of races. In one race, the final state of the scenario is certain, but the respective transitions are different. This is called a non-critical race. The second race is called a fatal race situation. In this case, there is additional unpredictable behavior in the final state, i.e. different defined solutions are undoubtedly possible.

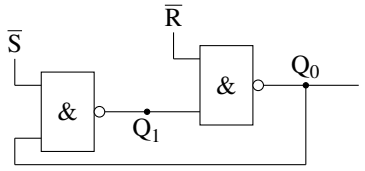



Fig. 2: Sequential Circuit

In the following, we will look at a sequential circuit in Fig. 2 with two state variables $q_1$ and $q_0$ inverted with respect to each other. A potential race can be found if it meets two conditions. First, one must determine whether at least two signals can change simultaneously. To do this, the individual transitions of the state variables of a sequential circuit must be superimposed. Eq. (6), e.g., shows all transitions (reflections) of $\delta_{Q_1}$ from 0 to 1 and $\overline{\delta}_{\overline{Q}_1}$ from 1 to 0 (see also the expressions in Eq.(17)).

$$(\tilde{\delta}_{Q_1}, \tilde{\overline{\delta}}_{\overline{Q}_1}) = (\overline{Q}_1 \wedge \delta_{Q_1}, Q_1 \wedge \overline{\delta}_{\overline{Q}_1}) \tag{6}$$

The superposition of the reflections of two inverted coupled state variables then leads to the double transitions in (7).

$$\begin{aligned} \tilde{\delta}_{Q_1Q_0} &= \overline{Q}_1\overline{Q}_0 \wedge (\delta_{Q_1}, \delta_{Q_0}) \\ \tilde{\overline{\delta}}_{\overline{Q}_1\overline{Q}_0} &= Q_1Q_0 \wedge (\overline{\delta}_{\overline{Q}_1}, \overline{\delta}_{\overline{Q}_0}) \end{aligned} \tag{7}$$

Second, the transitions of one state variable affect the output of the other state variable. Therefore, the partial derivative of the transition of one state variable over the other state variable must be determined in (8).

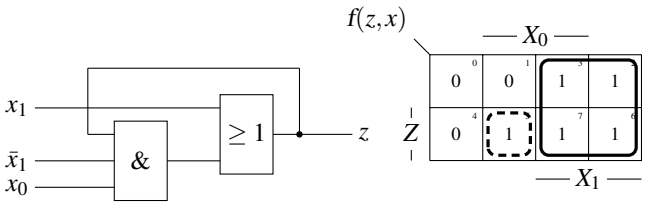
$$\begin{aligned} \frac{\partial \delta_{Q_1}}{\partial q_0} &= \delta_{Q_1}(Q_0) \rightsquigarrow \delta_{Q_1}(\overline{Q}_0) \\ \frac{\partial \delta_{Q_0}}{\partial q_1} &= \delta_{Q_0}(Q_1) \rightsquigarrow \delta_{Q_0}(\overline{Q}_1) \end{aligned} \tag{8}$$

The boundary conditions of possible races are then given by (9). These are two infima of the double transitions (reflections) of the two state variables and their mutually non-exclusive transitions.

$$\begin{aligned} \delta_{Q_1}(Race) &= (\tilde{\delta}_{Q_1Q_0}, \tilde{\overline{\delta}}_{\overline{Q}_1\overline{Q}_0}) \wedge \frac{\partial \delta_{Q_1}}{\partial q_0} \\ \delta_{Q_0}(Race) &= (\tilde{\delta}_{Q_1Q_0}, \tilde{\overline{\delta}}_{\overline{Q}_1\overline{Q}_0}) \wedge \frac{\partial \delta_{Q_0}}{\partial q_1} \end{aligned} \tag{9}$$

## IV. Examples

Given is the following circuit, see Fig. 3a, with its logical function $f(z,x) = z\bar{x}_1x_0 \vee x_1$, and its KV diagram (multi-set) in Fig. 3b. These representations are used to demonstrate the calculation of function and structue hazards.




(a) Logical Model (b) KV Diagram

Fig. 3: Example Logical Circuit

### A. Function Hazard

The structure has a static function hazard when switching from input [101] to [110] in (10) (and vice versa).

$$\left.\frac{\mathrm{d}f(z,x)}{\mathrm{d}x_\mathrm{p}}\right|_{X_\mathrm{p}} = f(z, X_\mathrm{p}) \rightsquigarrow f(z, \overline{X}_\mathrm{p}) \text{ with } x_\mathrm{p} = (x_1, x_0) \tag{10}$$

Therefore, we should calculate in Eq. (11) that the two output values are not different from each other.

$$\left.\frac{\mathrm{d}f(z,x)}{\mathrm{d}x_\mathrm{p}}\right|_{101} = 1 \rightsquigarrow 1 = 0 \tag{11}$$

Finally, by the varational derivation in Eq. (12), we must prove that there are assignments in Eq. (13) that lead to output variations instead.

$$\left.\frac{\Delta f(z,x)}{\Delta x_\mathrm{p}}\right|_{X_\mathrm{p}} = \bigvee_c f(z,X_\mathrm{p}) \nsim f(z,X_\mathrm{p} \nsim c) \quad \text{with } c \in \{01,10,11\} \tag{12}$$

$$\left.\frac{\Delta f(z,x)}{\Delta x_\mathrm{p}}\right|_{101} = 1 \nsim 0 \vee 1 \nsim 1 \vee 1 \nsim 1 = 1 \tag{13}$$

Since the results in Eq. (13) are true, a static function hazard can be proved.

### B. Structure Hazard

The next goal is to find all the structure hazards of the circuit in Fig. 3a. We recall that there may not be any timing difference when applying digital signals to the input pins. Since the blocks of the circuit do not overlap, a structure hazard should still be detected. By switching from input [111] to [101] for a short time, so to speak, $(x_1,\bar{x}_1)$ = [00] holds. This happens because of the faster switching of $x_1$ over $\bar{x}_1$ in the circuit.

$$\left.\frac{\mathrm{d}f(z,x)}{\mathrm{d}x_\mathrm{p}}\right|_{111} = f(z,[11]) \nsim f(z,[01]) = 1 \nsim 1 = 0 \tag{14}$$

Since the vectorial derivative in Eq. (14) is equal to 0, no function hazard is proved.

$$\frac{\Delta f(a)}{\Delta a_\mathrm{p}} = \bigvee_c f(A_\mathrm{p},a_\mathrm{q}) \nsim f(A_\mathrm{p} \nsim c,a_\mathrm{q}) \quad \text{with } a_\mathrm{p} = (a_2,a_0) \text{ and } c \in \{01,10,11\} \tag{15}$$

But due to the varational derivative after the extension of the function $f(a)$ in Eq. (15), a static structure hazard is proved with (16).

$$\left.\frac{\Delta f(a)}{\Delta a_\mathrm{p}}\right|_{1011} = 1 \nsim 0 \vee 1 \nsim 1 \vee 1 \nsim 1 = 1 \tag{16}$$

The totzeitmodell $f(a)$ maps the input vector $(z,x_1,\bar{x}_1,x_0)$ to $(a_3,a_2,a_1,a_0)$ where now $f(a) = a_3\bar{a}_1a_0 \vee a_2$ with $a_2 \sim a_1$. This proves that there is a static structure hazard due to $x_1$ or not. However, only the real physical boundary conditions will show whether this is a hazardfehler.

### C. Race

Using the low active RS-latch in Fig. 2, it is shown how race conditions can be determined. It is explained in more detail how a race is detected and how the table of values of the logical and real circuits can be determined to obtain the structure-preserving model. Therefore, the equations of the positive and negative rails (function, literal) of each state $q_1$ and $q_0$ are formulated in Eq. (17-18),

$$(\delta_{Q_1},\bar{\delta}_{\overline{Q}_1}) = (S \vee \overline{Q}_0, \bar{S} \wedge Q_0) \tag{17}$$

$$(\delta_{Q_0},\bar{\delta}_{\overline{Q}_0}) = (R \vee \overline{Q}_1, \bar{R} \wedge Q_1) \tag{18}$$

and the to-1-transitions and to-0-transitions in Eq. (19-20) are determined.

$$(\tilde{\delta}_{Q_1},\tilde{\delta}_{Q_0}) = (\overline{Q}_1 \wedge \delta_{Q_1}, \overline{Q}_0 \wedge \delta_{Q_0}) \tag{19}$$

$$(\tilde{\bar{\delta}}_{\overline{Q}_1},\tilde{\bar{\delta}}_{\overline{Q}_0}) = (Q_1 \wedge \bar{\delta}_{\overline{Q}_1}, Q_0 \wedge \bar{\delta}_{\overline{Q}_0}) \tag{20}$$

From these reflections (transition), the expressions for the double transitions in Eq. (21-22) can be derived. They describe input assignments, where both state variables $q_1$ and $q_0$ switch simultaneously.

$$\tilde{\delta}_{Q_1Q_0} = \overline{Q}_1\overline{Q}_0 \wedge (\delta_{Q_1},\delta_{Q_0}) = \overline{Q}_1\overline{Q}_0 \tag{21}$$

$$\tilde{\bar{\delta}}_{\overline{Q}_1\overline{Q}_0} = Q_1Q_0 \wedge (\bar{\delta}_{\overline{Q}_1},\bar{\delta}_{\overline{Q}_0}) = Q_1Q_0\bar{S}\bar{R} \tag{22}$$

In the last step, the dependencies from output variables (actual states) on input variables (preceeding states) is checked in Eq. (23-24),

$$\frac{\mathrm{d}\delta_{Q_1}}{\mathrm{d}q_0} = \delta_{Q_1}(Q_0) \nsim \delta_{Q_1}(\overline{Q}_0) = \bar{S} \wedge 1 = \bar{S} \tag{23}$$

$$\frac{\mathrm{d}\delta_{Q_0}}{\mathrm{d}q_1} = \delta_{Q_0}(Q_1) \nsim \delta_{Q_0}(\overline{Q}_1) = \bar{R} \wedge 1 = \bar{R} \tag{24}$$

and finally combined to the race conditions in Eq. (25-26).

$$\delta_{Q_1}(Race) = (\tilde{\delta}_{Q_1Q_0},\tilde{\bar{\delta}}_{\overline{Q}_1\overline{Q}_0}) \wedge \frac{\partial\delta_{Q_1}}{\partial q_0} = (\overline{Q}_1\overline{Q}_0\bar{S}, Q_1Q_0\bar{S}\bar{R}) \tag{25}$$

$$\delta_{Q_0}(Race) = (\tilde{\delta}_{Q_1Q_0},\tilde{\bar{\delta}}_{\overline{Q}_1\overline{Q}_0}) \wedge \frac{\partial\delta_{Q_0}}{\partial q_1} = (\overline{Q}_1\overline{Q}_0\bar{R}, Q_1Q_0\bar{S}\bar{R}) \tag{26}$$

In addition, Fig. 4 shows the NAND2 gate of the output

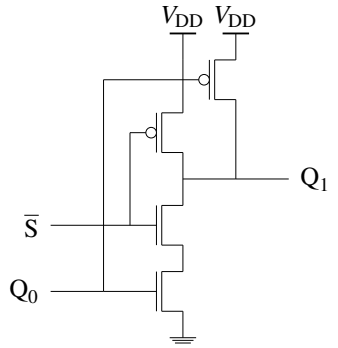



Fig. 4: NAND2 of $q_1$

variable (state variable) $q_1$. It is easy to see that for $\bar{S}$ = 1 the upper NMOS in the pull-down is conducting (on) and therefore the output (pin) $Q_1$ depends only on the input at $Q_0$. These boundary conditions are formulated and computable by the partial derivative in Eq. (23). The race condition is the reduction to just this boundary condition, of course only for double reflections (transitions). Using the results of Eqs. (25-26), the partial value table for the real low-active RS latch can be constructed in Table I. I. All race conditions for which the trailing initial value is not predictable are marked with $*$. In reality, however, the output values depend deterministically on physical boundary conditions that can be preset. However, they are not represented in the logical model.

TABLE I: Truth Table of RS Latch (logic and real)

| | | | | logic | | digital | |
|---|---|---|---|---|---|---|---|
| $Q_1$ | $Q_0$ | S | R | $Q_1$ | $Q_0$ | $Q_1$ | $Q_0$ |
| 0 | 0 | 0 | 0 | 1 | 1 | * | * |
| 0 | 0 | 0 | 1 | 1 | 1 | * | 1 |
| 0 | 0 | 1 | 0 | 1 | 1 | 1 | * |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 | 0 | * | * |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

## V. Implementation

In this section, the fundamentals presented so far are demonstrated (verified) using a real PCB (Printed Circuit Board). A microcontroller is used for the simultaneous switching of the input signals. For this purpose, it reads the signal values (assignments) from its registers by means of a clock signal. We notice that this simultaneous reading of input signals can nevertheless lead to errors.

The circuit board, specially designed for dynamic effects, is shown in Fig. 5. The circuit board consists of input switches, NAND gates, RC gates, inverters, and RS buffers. The RC gates can be used to simulate parasitic effects, the NAND gates are used to build DNF functions in NAND structure and the RS buffer can be used for asynchronous design in Dual-Rail, this was already described in [2].

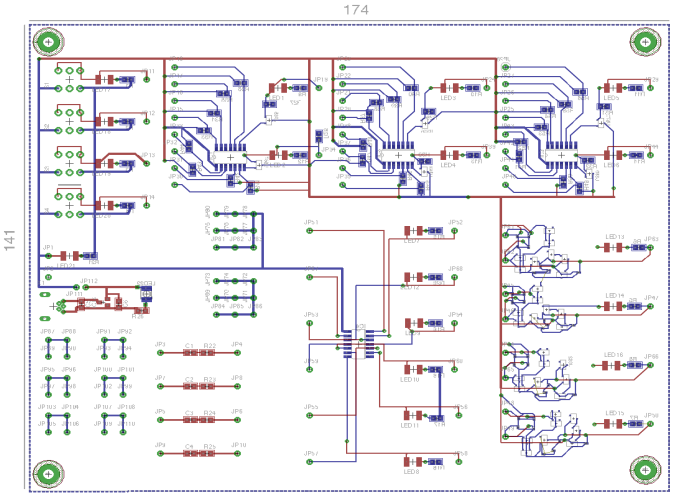


Fig. 5: PCB of the Trainingboard

### A. Static Function Hazard

To realize a static function hazard, the structure of Fig. 3a was implemented with NANDs. The expression for the circuit-under-measurement (CUM) is given in equation (27).

$$y = f(x) = x_1 \vee z\overline{x_1}x_0 = \bar{x}_1 \wedge \overline{z\overline{x_1}x_0} \tag{27}$$

According to Section IV-A, the function hazard from [101] to [110] is considered. When $x_0$ switches first at the transition of input assignments, a function hazardfehler is now detected by the measurement in Fig. 6. The y-axis is in units of 2V and the x-axis is in units of [ns] per square. The trigger point for the transition is also given. It is easy to see that the output value holds its value constant before and after the transition under consideration, while the output value deviates briefly but significantly from this value.

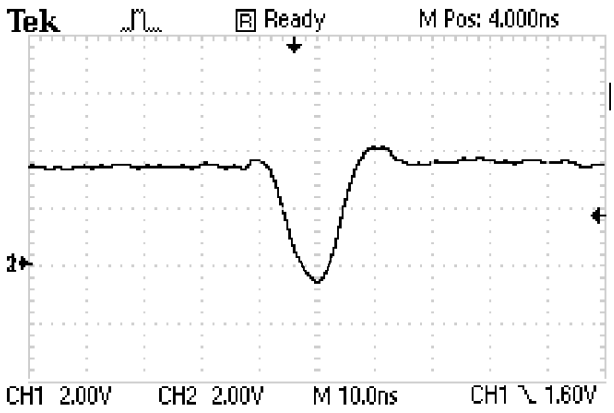



Fig. 6: Oscilloscope Image of a Static Function Hazard

It can be seen that the output value changes for a short time, but at the end the correct output signal of the assignment [110] is present. This is because the transistion into to the end assignment does not depend on the state. To get a better understanding of this, the dependencies of a state transfer function $\delta$ on its own state variable $z$ are considered, see Fig. 7 from [3]. Comparing $Z$ to $\bar{Z}$ in Fig. 7, it can be seen whether the state transfer function $\delta$ depends on the state variable $z$. For example, considering the single solid region, the state transfer function $\delta$ has the same value as the variable $z$ decomposed into $(Z,\bar{Z})$. This is the state stable part. If the loosely dotted area is considered, it can be seen that the function does not depend on $z$, since independent on the value of $z$ a logical 1 is always seen at the output.

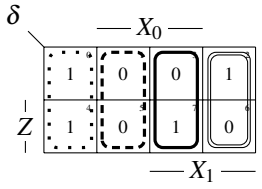



Fig. 7: Parts of a State Transfer Function

### B. Static Structure Hazard

To verify the static structure hazard from section IV-B, only the input variable $x_1$ is switched from 1 to 0. The delay of

the signal is caused by the real delay of the inverter (low pass behavior 1st order). This generates the negative literal $\bar{x}_1$, which is then present minimally later. This can be seen well in the screenshot of the oscilloscope in Fig. 8. However,

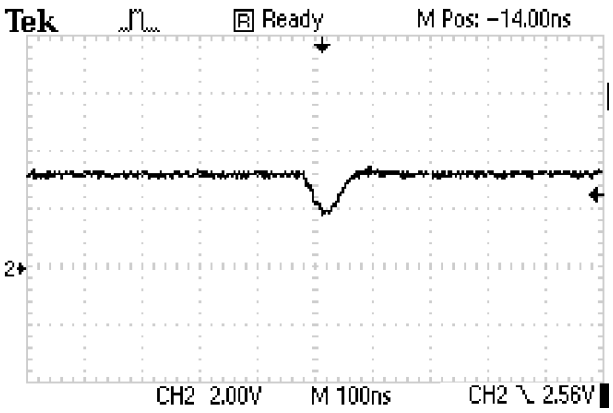



Fig. 8: Oscilloscope Image of a Structure Hazard

the delay of the inverter is smaller than the static functional disturbance of the previous example. Thus the disturbance has less influence on the output and the hazardfehler does not fall below 50 percent.

### C. Races

The last part of this section is about the fatal race. Therefore, the low active RS latch from Fig. 2 is wired on the PCB, which is shown in Fig. 9. Unused inputs of the NAND4 gate are set to VDD in this case, since logic 1 is the neutral element of the NAND.

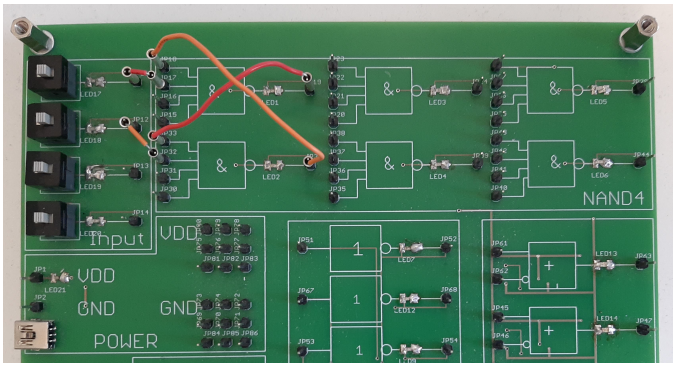



Fig. 9: Low Active RS Latch (cut out area)

In the state transition shown in the oscilloscope in Fig. 10, with input assignment $(s,r) = [00]$, both output states $Q_1$ and $Q_0$ are first clamped to logic 1. Then $(s,r) = [11]$ is switched. Both NAND gates try to pull their respective output state to the value 0. But since the two gates cannot switch at the same time (they are inverted to each other) the stronger and thus the faster gate pulls its output state to 0 and thus influences the input of the other gate essentially. The change of the other gate cannot be completed, but is overruled, and switches back (or does not switch).

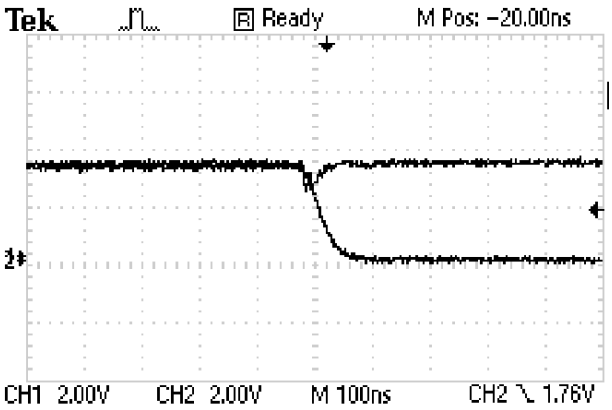



Fig. 10: Oscilloscope Image of a Race

## VI. Conclusion and Future Work

At the beginning, the theoretical basics of dynamic effects of asynchronous circuits from the literature were elaborated again in a summarized way. Special attention was paid to the race condition (boundary conditions for races of edges) and its formal description. These findings could then be reproduced using real basic circuits. This is very important for learning such effects, and can be experienced in an exemplary way. With the formulas provided, future more complex designs can be formally analyzed and designed to be hazard error free and race error free, which has already been shown in [2]. These designs could then be verified on the presented board, as the implemented functions only need to replace the NAND4 gates. This would allow safety critical asynchronous designs to be compared. Certainly, announced new CAD tools for realizing more extensive designs can be validated. However, one of our stated goals would be to use FPGAs for low-cost prototyping and realization of safe designs.

## References


[1] *Automotive IC Market Forecast With Strongest CAGR Through 2024*. IC INSIGHTS, June 17, 2020.

[2] Florian Deeg, Jie Zhu, and Sebastian M. Sattler. Asynchronous design. In *AmE 2020 - Automotive meets Electronics; 11th GMM-Symposium*, pages 1–5, 2020.

[3] Mustafa Oezguel, Florian Deeg, and Sebastian M Sattler. Mealy-to-Moore Transformation - A state stable design of automata. *Advances in Science, Technology and Engineering Systems Journal*, 2(6):162–174, 2017.

[4] Bishwajeet Pandey, Jyotsana Yadav, Yogesh Kumar Singh, and Pradip Swarnkar. Energy efficency of asynchronous and synchronous vlsi circuit on 40nm and 90nm fpga. In *2013 International Conference on Energy Efficient Technologies for Sustainability*, pages 57–60, 2013.

[5] Bern Steinbach and Christian Posthoff. *Boolean Differential Calculus*. Morgan & Claypool Publishers, 2017.

[6] Hans Joachim Zander. *Logischer Entwurf binaerer Systeme*. VEB Verlag Technik, 1989.