Adaptable reference voltage supply and clock signal phases for high-speed I/O interfaces

Bachelor's Thesis

Arik Küster
December 30, 2024

Supervised by
Joscha Ilmberger
Johannes Schemmel

Heidelberg University

(Contains corrective additions in Fig. 17 and Fig. 24 marked in **RED**)

Abstract

Scaling neuromorphic computing architectures such as BrainScales-2 demands, high-speed, and low-latency interfaces. To meet these requirements, a flexible and robust I/O interface is being developed. It is composed of a wide array of single-ended, source-synchronous I/O cells. Each cell incorporates a sense amplifier to enable data reception across varying voltage swings, necessitating both a reference voltage and a clock signal.
In this work, a digital-to-analog converter (DAC) is implemented to provide the reference voltage and a clock-phase distribution network is designed with the goal of error-free operation under changes in temperature and supply voltage. The DAC employs a resistor chain tapped at multiple points, its functionality and properties are verified in simulation. A 32-output configuration occupies an area of 136.5 µm^2 and consumes between 2.4 µW and 21 µW, dependent on the I/O supply voltage.
The clock-phase distribution network propagates the outputs of a delay-locked loop (DLL) across the I/O bank. It distributes eight phases at six times the minimum metal spacing while maintaining monotonic operation within a 7σ variance, consuming 1.4 mW over a 500 µm-wide I/O bank.
Both circuits were successfully integrated into a test chip for evaluation.

Zusammenfassung

Die Skalierung neuromorpher Rechenarchitekturen wie BrainScales-2 erfordert Hochgeschwindigkeits- und latenzarme Schnittstellen. Um diese Anforderungen zu erfüllen, wird eine flexible und robuste I/O-Schnittstelle entwickelt. Sie besteht aus einer großen Anzahl single-ended, source-synchronen I/O-Zellen. Jede Zelle enthält einen sense-amplifier, der den Dateneingang bei unterschiedlichen Spannungshüben ermöglicht. Dafür sind sowohl eine Referenzspannung als auch ein Taktsignal erforderlich.
In dieser Arbeit wird ein Digital-Analog-Wandler (DAC) implementiert, um die Referenzspannung bereitzustellen, und ein Taktphasenverteilungsnetzwerk entwickelt, das einen fehlerfreien Betrieb bei Änderungen von Temperatur und Versorgungsspannung gewährleisten soll.
Der DAC verwendet eine Widerstandskette, die an mehreren Punkten abgegriffen wird. Seine Funktionalität und Eigenschaften wurden in Simulationen verifiziert. Eine 32-Ausgangs-Konfiguration belegt eine Fläche von 136.5 µm^2 und verbraucht je nach I/O-Versorgungsspannung zwischen 2.4 µW und 21 µW.
Das Taktphasenverteilungsnetzwerk leitet die Ausgänge einer Delay-Locked Loop (DLL) über den gesamten I/O-Bereich weiter. Es verteilt acht Phasen mit dem Sechsfachen des minimalen Metallabstands und erhält dabei einen monotonen Betrieb innerhalb einer 7σ-Varianz aufrecht. Über eine 500 µm breite I/O-Bank verbraucht es 1.4 mW.
Beide Schaltungen wurden erfolgreich in einem Testchip zur Evaluierung integriert.

Contents

1 Introduction

With the current rapid advances in AI and its ever-increasing power budgets, interest into how to increase its efficiency rises accordingly. Analog hardware offers two possible methods for increasing the efficiency. One is the possibility for model inference through analog computation. Another is the efficient execution of alternative model architectures, oftentimes mimicking biological networks, like Spiking Neural Networks (SNNs) [1].
The BrainScales-2 distributed neuromorphic computing architecture[2, 3] offers both of these possibilities. With its HICANN-X application specific integrated circuit (ASIC) and high acceleration factor in neuron emulation of about 1000, it requires large bandwidths for communication.
A flexible, efficient and compact I/O interface, designed for different communication distances, was developed and tested for these purposes [4] [1]. It uses the TSMC 65 nm technology. The same is true for this work. To maximize the interfaces communication speed for a given space, it uses one data pin per signal, i.e. communicates in a single ended way.

When commutation distances are short and high speeds are required, the interface provides a terminated mode that matches the communication line's resistance with a terminating resistor. Conversely, for extended distances with minimal power consumption, the interface supports an unterminated mode, along with a configurable signal swing voltage between the high and low levels. Meanwhile, to ensure optimal performance in terms of space and speed, the core of a chip must utilize full swing logic.

Therefore, the signal needs to be converted back to a full swing signal.
In this I/O bank, this is performed by a pair of sense amplifiers, each converting one half of the received data signal to a full swing signal. For the output of them to match the received data, two additional inputs are necessary.
One is a clock signal, meaning periodically changing between high and low, which is received along with the data signal. It dictates when, in the period of the signal, the sense amplifier is sensitive to the data signal. For optimal operation, a phase shift needs to be applied to the clock input into the sense amplifier. The other input is a reference voltage. Comparison to it enables the amplifier to distinguish between a high and low signal.
For adjusting the clock phase, the current implementation uses a delay line in the form of an inverter chain, enabled by the fact that each inversion adds a small delay. This chain then has outputs or "taps" along multiple points along it. The eight taps span a delay of roughly 700 ps, making for about 90 ps of phase difference between two taps. Which tap of this is ideal one for an error free operation of the sense amplifier is dependent on the supply voltage, process corner[2], and temperature. This is because the delay introduced by any inverter is influenced by these parameters.

[1] Accepted at APPCAS 2024 and in proceedings to be published

[2] Chips are produced on wafers carrying many single chips. The process corner describes the performance difference of chips from different wafers

To improve on this situation, the delay line will be replaced by a delay locked loop (DLL), which self adjusts its delay and *locks* it relative to one or multiple periods of the clock signal, making the delays less dependent on the variations in temperature, corner and supply voltage. As the power draw of the DLL to be used in the next iteration of the interface is substantially higher than that of the current delay line, this work examines distributing the output of one DLL over the whole I/O bank. It is explored in section 3, first by copying the output to metal lines spanning all the I/O bank, and thereafter an expansion on the approach by breaking the metal line at each I/O cell and making the transfer to the next I/O cell optional.
As for the reference voltage, it is broadcast from one modified I/O cell to all others. This work aims to improve upon this by adding a low power digital-to-analog-converter (DAC) in each I/O cell. An additional implication is that, as the ability to broadcast the reference voltage will remain, redundancy is added. Furthermore, as will be shown in section 2, because it is implemented as a resistor chain, tapped at different points, the outputs will retain their position relative to the signal swing, invariant under corner, temperature or supply voltage variation.

The goal of these improvements will be, that the next iteration of the I/O interface, already submitted for production, can operate error free under a larger change in parameters, be that in frequency or the other already mentioned ones.

For completeness, the circuit is built up from the basics. The first fundamental part of integrated circuits are Transistors.

1.1 Transistors

The most common Field-Effect-Transistors in integrated circuits are the ”p-” and ”n-channel metal oxide transistors” (PMOS, NMOS). Their symbols are shown in fig. 1.

The position of source and drain is only determined when a potential is

Figure 1: Symbols for PMOS (left) and NMOS (right) with the terminals source (S), gate (G), bulk (B) and drain (D) with the bulk terminal connected to source.

applied, with the terminal of the higher potential being source (S) for the PMOS and drain (D) for the NMOS. The bulk terminal highlighted in Fig. 1 will, for all cases in this work, be connected to the supply voltage for the PMOS and to ground for the NMOS, they will therefore not be explicitly drawn.

In this configuration, the behavior can be approximated by three modes. Central to all of these is a transistor dependent "threshold voltage" V_T [3]. The voltages needed for explanation are named by the potential from which to which terminal they symbolize, e.g. the voltage V_{GS} being the voltage from G to S and being positive if the potential at G is higher. The current is given in the same way, named after their direction. The modes are:

1. cut-off: $V_{GS} > V_T$ for the PMOS and $V_{GS} < V_T$ for the NMOS. The transistor can then be seen as non-conducting

2. linear region: $V_{GS} \leq V_T$ and $V_{DS} > V_{GS} - V_T$ for the PMOS and $V_{GS} \geq V_T$ and $V_{DS} < V_{GS} - V_T$ for the NMOS. The current running from S to D for PMOS and from D to S for NMOS is then:

$$I_{SD/DS} = \kappa \frac{W}{L} \left((V_{GS} - V_T) V_{DS} - \frac{V_{DS}^2}{2} \right) \tag{1}$$

3. saturation region: $V_{GS} \leq V_T$ and $V_{DS} \leq V_{GS} - V_T$ for the PMOS and $V_{GS} \geq V_T$ and $V_{DS} \geq V_{GS} - V_T$ for the NMOS. $I_{SD/DS}$ is then:

$$I_{SD/DS} = \kappa \frac{W}{2L} (V_{GS} - V_T)^2 (1 + \lambda V_{DS}) \tag{2}$$

In the equations (1) and (2), W stands for the width of the transistor and L for the length, while κ is a constant which is the product of the charge carrier mobility μ of the channel and the Gate-Bulk Capacity C_{ox} per gate area. The last of the parameters is λ, a parameter incorporating the effect of "channel length modulation" into the model. It is always positive and tends to increase with lower V_T and vice versa around.
Lastly, an effect that turns out to be of importance for this work, is the parasitic capacitance between gate and bulk of the transistor, as this results in the possibility of transmission of signals of high frequency through the gate, called kickback. This can be understood using Fig. 2.

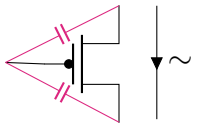


Figure 2: Transistor with source/drain currents having high frequency components

Rapidly changing current through the transistor will also cause the charge density on the bulk side (right side in Fig. 2) of the parasitic gate capacitance to change, inducing an electric field effecting the gate side of the parasitic capacitance. With the field comes a change in potential at the gate.

[3] It is important to note V_T is negative for the PMOS, and positive for the NMOS. In fact, if the V_T for NMOS and PMOS have the same absolute value, PMOS and NMOS are related in the way that flipping all voltages applied to the PMOS will give the same operating mode when applied to the NMOS.

1.2 Required subcircuits

1.2.1 CMOS Inverter and inverting three state inverter

The inverter is the simplest complementary metal oxide field effect transistor (CMOS) circuit, its construction is shown in Fig. 3

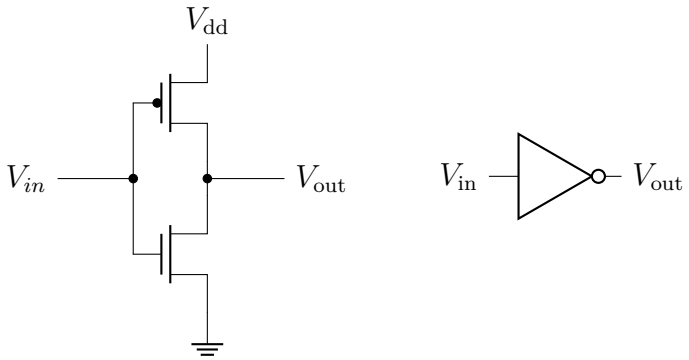


Figure 3: Circuit for a CMOS inverter with the supply voltage V_{dd}. The supply voltage and ground connection are abstracted away in the symbol

The most important feature is its transfer function. This can be seen in Fig. 4.

Using the figure, the logical function of outputting a low signal when receiving a high input and vice versa, is confirmed. Another defining feature is its extremely high direct current (DC) resistance, as only a small leakage DC current can pass through the gate. Comparatively, the output resistance is highly dependent on the used transistors but typically in the hundreds of ohms. This combination makes it perfect for decreasing the output resistance of a logical signal.

A small expansion on this is the three state inverter or tristate inverter. Its circuit is shown in fig. 5a.

Its defining feature is that it enables the output to be switched off. For S high, the NMOS it is connected to is in the linear or saturation region, i.e. conducting. The same is true for the PMOS, as the potential at its gate is low for S high. In this configuration, the three state inverter behaves just like an inverter.

If S is low, both of the inner transistors are in the cut-off region, i.e. not conductive. It is important in this case that the blocking transistors are between the transistors receiving V_{in}, as the other way around, V_{in} would strongly couple to V_{out} capacitances of the transistors. It is also worth mentioning that although this version of a three state inverter contains an inverter to get the inverted signal of S, $\bar{S}$, if S is broadcast to multiple three state inverters, so can $\bar{S}$, making only one inverter necessary for all the three state inverters.

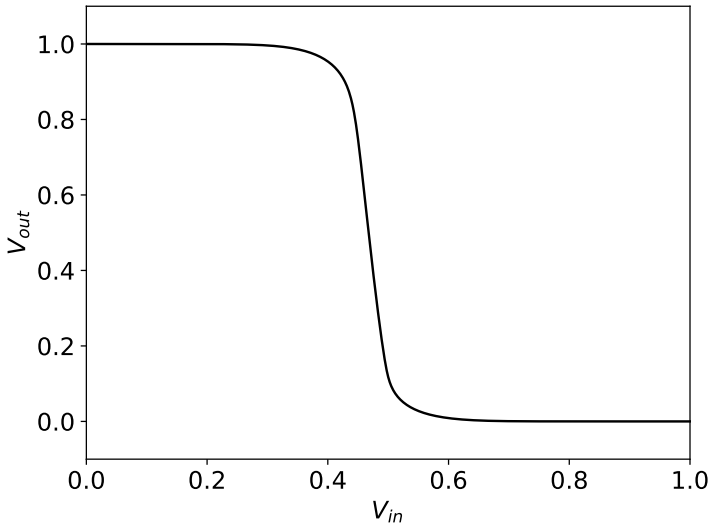


Figure 4: Transfer function for the inverter.

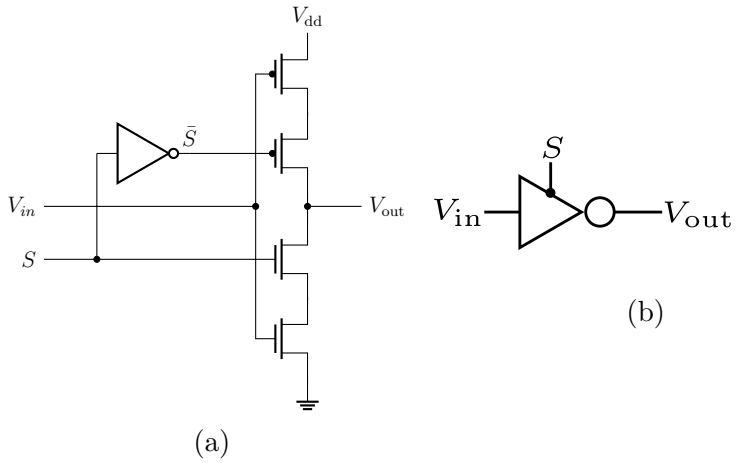


Figure 5: The circuit diagram and symbol used for a simple three state inverter.

1.2.2 Transmission Gate

The circuit and symbol for the transmission gate are shown in fig. 6. It is the true equivalent to a switch, combining that a PMOS transmits voltages higher than V_T well and that the NMOS transmits voltages lower than $V_{\mathrm{dd}} - V_T$ well. This is displayed in fig. 7.

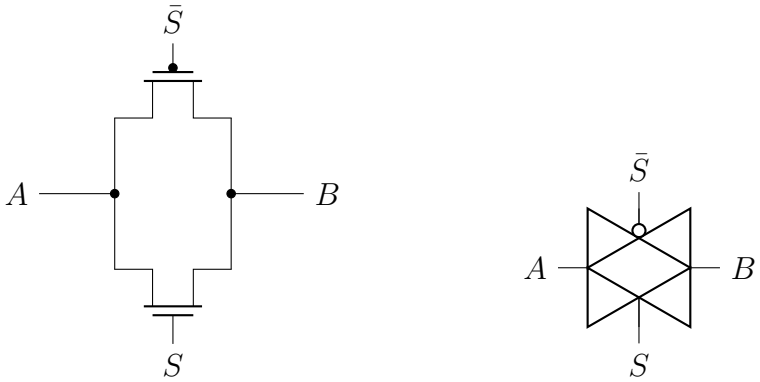


Figure 6: Circuit and symbol of the transmission gate

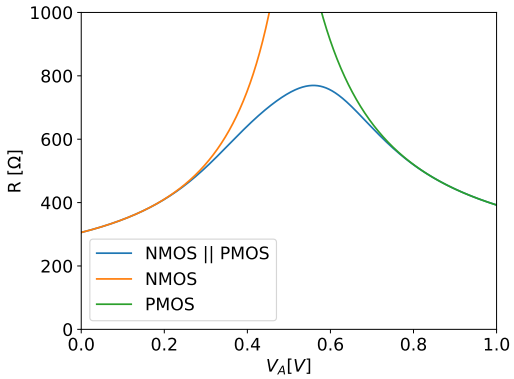


Figure 7: The resistance of the transmission gate for $V_A \approx V_B$

2 Resistive Chain DAC

In previous work [5], a "current DAC" design was explored where binarily weighted transistors were pushing current into a resistor to achieve different output levels. In this work, a different approach will be used.
Here, the reference voltage will be provided by choosing a tap along a long line of resistors. While the output levels for the current DAC are quadratically dependent on the supply voltage, shifting the range of the output levels depending on it, this is not so for the resistor line, the taps of which stay at the same position relative to the supply voltage. For the current DAC this could be resolved by another reference voltage, this time supply voltage invariant. A possible implementation is a bandgap, which outputs a constant reference current. Because of the typically large space requirements of a bandgap, a possible configuration is to have one for the whole chip. To distribute the output current, it could be transported using current mirrors to the I/O cells. There, the reference current can be converted to a voltage using a resistor.
The voltage drop over the resistor will be similar to the one used in the current DAC. Using similar resistances for both will result in a doubling of the space required by resistors and about a doubling of the power draw. Moreover, though this should resolve the dependence on the external supply voltage, the dependence on the process corner remains. Although it is again possible to compensate for this to some degree by adding calibration capability to the DAC, for example by making the changing resistor size configurable, this will again add additional complexity.

It is much simpler to sidestep the dependence on process corner and supply voltage by using a resistor chain. As will be shown in the following chapter, the power draw and output levels for a certain size of the whole circuit is similar to a current DAC with the same specifications. A nice bonus is the guaranteed monotony of the output levels, as there is a resistor between each of them. Having established the resistor chain as worth considering, it will now be explained how to turn it into a DAC. For the circuit to have one output, the different taps have to be merged into one final output line, also known as multiplexing. This can be achieved in multiple different ways.
Deciding whether a tap should be used or not can be achieved using a transmission gate for each of the taps. Since at any point in time only one tap should be active, exactly one of the transmission gates should receive a 'pass' signal, while all the others should block. This could be done using as many signal lines as there are taps. For tap counts of 16 and higher, this far exceeds the necessary signal bits of $\log_2 tapcount$. Some kind of conversion is therefore necessary. A common approach to address decoding, finding application here, would be to take a multi-input AND-gate receiving the signal bits, some of which inverted, and activating on exactly one of the combinations, depending on for which signal bits the inverted signal was connected. Scaling wise, for a signal of width n, 2^n AND-gates are needed. With each using $2(n+1)$ transistors, this makes for a total CMOS count of $(n+1)2^{(n+1)}$. Each AND-gate would then

control one of the transmission gates at the taps. The outputs of all of these can then be connected together, with this being the final output.
Another approach would be to use a binary tree structure of transmission gates, like is displayed in 8, requiring $2^{(n+1)} - 2$ transmission gates in total, each using 2 transistors.
For a 5-bit signal, as will be used in this work, this would make for 384 transistors for the address decoder and 124 for the transmission gate tree. However, this still does not tell the whole story. That is because the transistors of an AND-gate can be physically placed closer to each other and are, for this implementation, less wide than those of the transmission gates. Specifically, for a signal width of 5-bit, a 5−bit AND-gate requires about $3\,\mu m^2$ of space making for $96\,\mu m^2$ for 32 AND-gates. For the case of the tree, these would get replaced by 30 transmission gates, each taking up about $1\,\mu m^2$. The transmission gate tree therefore saves $66\,\mu m^2$ in space. With the final implemented DAC with transmission gate tree reaching about $140\,\mu m^2$, this is substantial.
The disadvantage of a transmission gate tree is that the large internal capacity and resistance, in combination with the switching signal coupling into the tree, makes fast switching impossible. Though this will not be the bottleneck in this case. It will instead be the internal resistance, because of a large stabilizing capacity connected to the output. The AND-gate variant performs better in this regard, as there is only one transmission gate between the output any tap. However, as the reference voltage is supposed to be rarely switched, this is not significant here. Going into more detail on the circuit, a small scale version is shown in fig. 8.

The design outputs the voltage at a certain tap by setting the bit sequence $(S_1, S_2...)$ to the corresponding tap number in binary form. Specifically, for the design to output the voltage at tap 0, (S_1, S_2) must be set to $(0, 0)$ and for tap 3 to $(1, 1)$. The implemented circuit in this project is a larger version of the one shown in fig. 8. Specifically, a chain of 45 resistors with 32 taps along the chain was used. For the $32 = 2^5$ taps, a five-level tree containing $2^5 - 2 = 62$ transmission gates is necessary. The supply voltage at the top of the resistor chain V_S is adaptable and can be set to the signal swing. With 45, more resistors were chosen than necessary, the specific of choice where the resistor chain was tapped is shown in fig. 9a.
The choice was made considering that when outputting a reference for unterminated operation, the expected optimal reference voltage is at $V_S/2$. For terminated operation, the expected optimal reference voltage lies at $V_S/4$. The choice of where to tap the chain was made in consideration with both modes. This means that between $V_S/4$ and $V_S/2$, every resistor is tapped, with lower tap densities towards the bottom and top of the chain. In reality, a binary signal comes closer to ground than to the supply voltage for the case of this I/O interface, since the PMOS transistors are not scaled up enough to compensate for their lower driving[4] strength per width compared to the NMOS transistors. The optimal reference voltage will therefore typically be somewhat lower than

[4]"driving" means to control another part of the circuit

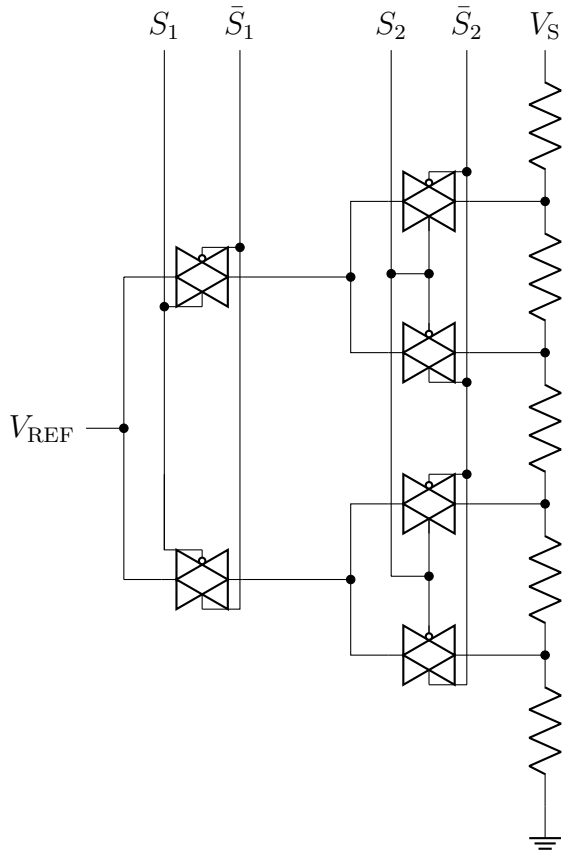


Figure 8: DAC using a resistor chain with a 2 level tree, 5 resistors and 4 taps along the resistor chain. The output level is encoded by S_1, S_2 and their inverses $\bar{S}_1, \bar{S}_2$

$V_S/2$ or $V_S/4$, but by how much is highly design dependent, and was therefore not examined in detail.
Another change in the physical implementation is that a transmission gate was added at the very top of the resistor chain, allowing the circuit to be shut off. A transmission gate instead of a PMOS was necessary in this case, as V_S is supposed to be able to go down to 0.4 V, where a PMOS would not be conducting anymore. An alternative would have been to place an NMOS at the end of the resistor chain, causing the whole circuit to be pulled to the supply voltage upon activation. However, as the ground potential is more stable and closer to where the reference output is expected to be, the transmission gate at the top of the chain was opted for.

2.1 Simulation

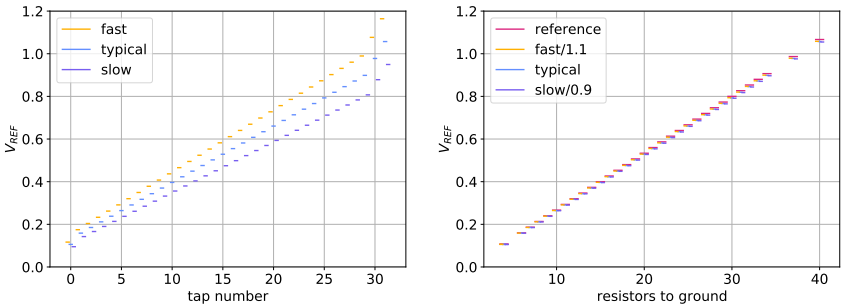
Starting off the simulations will be the behavior of the output levels in the different "far parameter corners" or "far corners". In this case, using T for temperature and V_dd for the supply voltage, this refers to the parameter combinations:

- slow (s): process corner "slow", $V_\mathrm{dd} = 1.08\,\mathrm{V}$, $T = 80°$

- typical (t): process corner "typical", $V_{\mathrm{dd}} = 1.2\,\mathrm{V}$, $T = 55°$
- fast (f): process corner "fast", $V_{\mathrm{dd}} = 1.32\,\mathrm{V}$, $T = 30°$

This comes about from how the parameters affect transistors. In terms of a CMOS inverter, the "slow" process corner, lower supply voltage of 1.08 V and higher temperature of 80° all increase the delay of the CMOS inverter compared to the (t) case. Other circuits behave similarly.
Using these far corners to simulate the behaviour of the DAC, fig. 9a is obtained. More interesting for confirming the claims of the output relative to V_{S} being independent of the supply voltage and the corner is fig. 9b, however. There, the voltage levels relative to the supply voltages are tested in the different far corners and compared to idealized values.
As can be seen from fig. 9b, the ideal values are closely undershot in all far corners, with the error increasing when going up the ladder and culminating at the highest tap with a difference of, rounded to the last digit, 10 mV.
At a supply voltage of 1.2 V, this is about 40% of the minimum voltage step size of 26.7 mV. The shift is because of the resistance of the transmission gate at the top of the resistor chain. Using the coarse approximation of the output level being shifted by 1%, it follows, that the resistance of the transmission gate must also be about 1% of the whole chain. Since the resistance of the chain is about 69 kΩ, the resistance of the transmission gate must be about 690 Ω. This is close to the simulated value of 577 Ω for the small signal resistance for the transmission gate when transmitting 1.2 V. The small signal resistance of the transmission gate is largest at $0.707\,\mathrm{V} \approx 0.7\,\mathrm{V}$ with 1069 Ω. The results for simulating the output levels for the case of $V_{\mathrm{S}} = 0.7\,\mathrm{V}$ are shown in fig. 10. In this case, the shift of the highest stage is 14.1 mV, making for about 90% of the minimum tap distance.
This error could either be reduced by widening the transmission gate or by using a power down NMOS at the bottom of the chain instead of a transmission gate at the top. As the voltage of across the NMOS would always be close to ground, it would always be optimally conducting. Furthermore, as the NMOS would replace the whole transmission gate, it could be made even larger than the combination of NMOS and PMOS of the transmission gate. However, as the error not large and its worth compared to a more stable power down potential and a faster power up time is not clear, it was opted not to make this change in this iteration of the circuit.

Now having verified the behavior of the circuit in the different far corners, the next thing to test is the variation of the resistors due to mismatch between them. Mismatch being the variation of two components on the same chip due to manufacturing inaccuracies. This is shown in fig. 11. The maximum standard deviation of one voltage level relative to the minimum step size (being $\frac{V_{\mathrm{S}}}{45}$) is $(3.5 \pm 0.1)\%$. However, it's important to keep in mind that the deviation of two taps next to each other is highly correlated. In fact, the maximum observed deviation between two voltage levels next to each other was only about 2.5% in relation to the minimum step size. It is also worth mentioning that since the cause of the mismatch is the uneven value of the resistances in



(a) Output levels in different far parameter corners

(b) The outputs for different far corners scaled to the typical supply voltage of 1.2 V and compared to an idealized reference

Figure 9

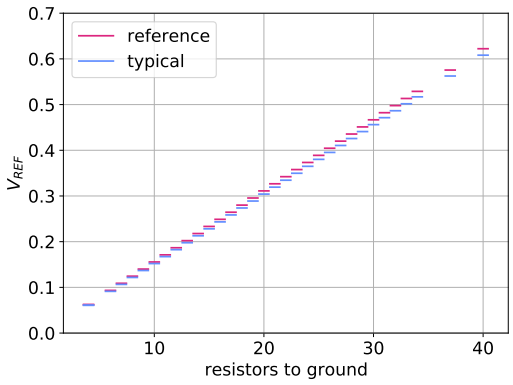


Figure 10: The Simulated output voltages of the DAC with transmission gate compared to without one.

the resistance chain, the mismatch is not affected V_S and the results from fig. 11 are therefore applicable to other values of V_S under the correct scaling of the axes. The mismatch is much smaller than the distance between the levels, and therefore small enough for the intended purpose. A side fact is that the standard deviation gets smaller even when moving the edges of the chain. In fact, it is directly proportional to the output resistance of the chain at a certain tap, which is

$$R_L(R_G - R_L) \tag{3}$$

with R_L being the resistance from the tap to ground and R_G being the total resistance.

Now, having looked at what the DAC would put out under ideal conditions, how the reference voltage behaves under the intended load will be simulated. The load is a sense amplifier, more specifically a PMOS driving into one of the inputs of a CMOS flip-flop, that is periodically being reset for one half of each

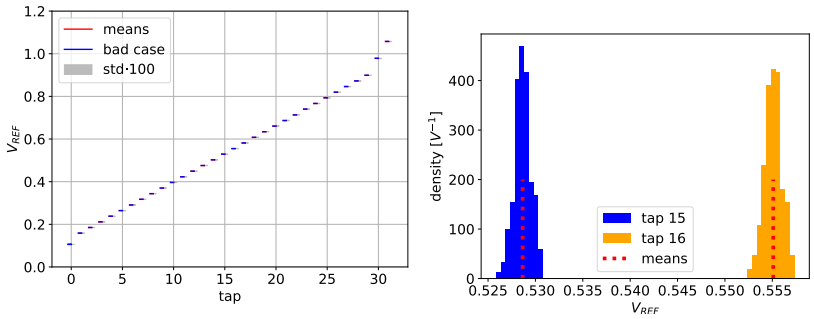


Figure 11: The DAC outputs for a mismatch simulation for 300 samples in the (t) case with $V_S = 1.2$. On the left the means with its standard deviation next to it, to the right, are shown. The "bad case" refers to where a large deviation from one voltage level to the next was observed, it mostly covers the means. As the standard deviations are hard to recognize, a histogram for the taps 15 and 16 is shown to the right

clock period. The effect is that the drain and source of the PMOS periodically switches between high potential and low potential, causing some kickback to V_{REF}. The extent of this is dependent on the output resistance of the DAC and the capacity attached to the V_{REF} line. The circuit was implement using multiple small MOSCAPs, i.e. capacitors using the gate-drain or gate-source capacity of transistors, adding up to a capacity of 90 fF in this case. The output resistance can be determined using the Thévenin equivalent. Using it, the DAC simplifies to what is shown in fig. 12. The model splits the resistance

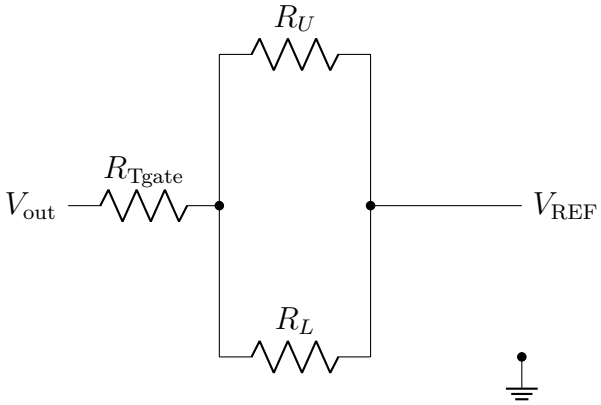


Figure 12: The Thévenin equivalent of the DAC. $R_U = R_G - R_L$ describes the resistance from the tap to the top of the chain. R_L describes, as above, the resistance from the tap to groud. R_{Tgate} is the resistance of the transmission gate tree, which is in effect 5 transmission gates in a row.

into the contribution from the resistive DAC, which are 5 transmission gates in series, as well as the resistances from the tap to V_S and the tap to ground. This division of the large resistance is another advantage compared to the current DAC. While the current DACs output resistance is always approximately equal to the large resistor used in it, this architecture divides up the large resistor,

reducing its contribution to the output resistance by at least a factor of 4. This improvement is somewhat nullified by the additional transmission gate tree not present in the current DAC. Using the model to simulate the output resistance results for V_S of 1.2V and 0.4V, gives the plots in fig. 13.

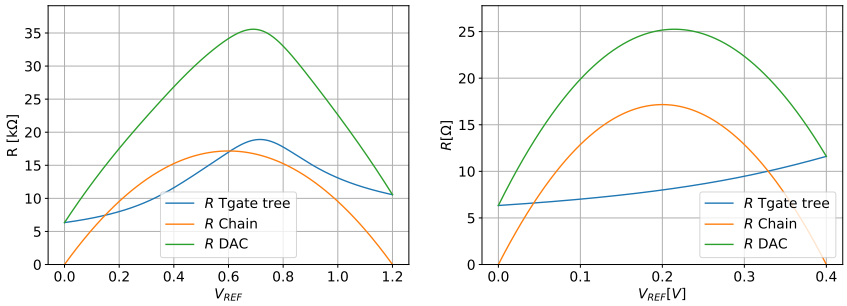
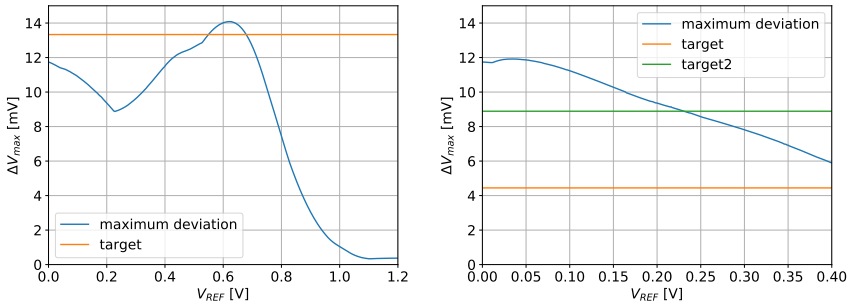


Figure 13: The differential resistance of the DAC design for the voltages $V_S = 1.2, 0.4$, with the contributions from the transmission gate tree and the resistor chain explicitly highlighted.

Fig. 13 shows that for the implemented DAC, in the case of $V_S = 1.2\,\mathrm{V}$, about half of the resistance comes from the resistor chain and the other half is contributed by the transmission gate tree. The resistance peaks at a value of $35.5\,\mathrm{k\Omega}$, at about 50% of the value $68.7\,\mathrm{k\Omega}$ of the chain. For the case of $V_S = 0.4\,\mathrm{V}$, the transmission gate is in the conduction region of the NMOS and therefore contributes less to the overall resistance. The resistance peaks at a value of $25.2\,\mathrm{k\Omega}$, at about 36% of the whole chain. With the output resistance determined, the kickback can now be simulated. The testing circuit for this was adapted from work in [5].
For the instability, the an ambitious target to set would be half of one voltage step at the lowest expected signal swing of $0.4\,\mathrm{V}$, this is $0.4/45/2\,\mathrm{V} = 4.4\,\mathrm{mV}$. The result for the instability for the implemented design is displayed in fig. 14. For fig. 14a, a superposition of two curves can be recognized, one being the kickback decreasing with increasing V_{REF}, the other is the output resistance of the DAC. More importantly, except for the region from $0.55\,\mathrm{V}$ to $0.68\,\mathrm{V}$, the deviation stays underneath half one voltage step.
This can not be said for when $V_S = 0.4\,\mathrm{V}$, as in this case, V_{REF} stays above half of the minimum step for the whole voltage range, being rather close to one voltage step around $V_S/2 = 0.2\,\mathrm{V}$.
In a next iteration of the chip, this can be improved by increasing the capacitance stabilizing V_{REF}, as the capacitor is still small compared to the size of the DAC. If the goal is to have the instability under half of one step for all V_S, the capacitance would need to roughly double, as for a the approximation of a large resistance, the instability and the capacitance are approximately inversely proportional.

Closing the discussion of the DAC, it was mentioned in the beginning that the response time of the transmission gate tree is high, but also not important for this use case. However, for training algorithms, it can still be useful to have



(a) (b)

Figure 14: The simulated instabilities for the designed DAC with an additional 90 fF capacitor for $V_S = 1.2, 0.4$ V and different output voltages.

some rough idea of when the value will have settled down. Just for charging the 90 fF capacitor at the end, the time constance is $t = RC = 36\,\text{k}\Omega \cdot 90\,\text{fF} = 3.24\,\text{ns}$. When the time RC passes, the capacitor will have reduced the distance between the voltage it is being charged with and the voltage it currently holds by a factor of e. The time to reach to be within V_{tol} of V_2 when starting from V_1 is therefore computed as in (4).

$$t = RC \cdot \ln\left(\left|\frac{V_2 - V_1}{V_{tol}}\right|\right) < 36\,\text{k}\Omega \cdot 90\,\text{fF} \ln\left(\left|\frac{1}{1/90}\right|\right) = 14.6\,\text{ps} \tag{4}$$

For the obtained upper bound the maximum output resistance of the DAC and $V_2 - V_1 = V_S$ with a tolerance of $V_S/90$, being half the minimum step size, was used. Testing something similar with the actual DAC, the transition from the level at tap 0 = 00000 to 31 = 11111 is shown in fig. 15. The response time to be within half of one voltage step was 12.67 ns, close to the upper bound.

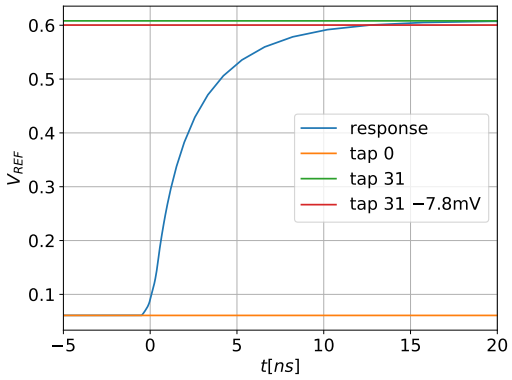


Figure 15: The response of the DAC at the output terminal when switching between tap 15 and 16.

For an overiew, the final specifications are listed in table 1.

	this work	previous work
power draw	$2.4\,\mu W$ to $21\,\mu W$	$< 50\,\mu W$
size LxW	$19.5 \times 7\,\mu m^2$	$11.5 \times 5\,\mu m^2$
size	$136.5\,\mu m^2$	$57.5\,\mu m^2$
output levels	32	16
output resistance	$< 36\,k\Omega$	$< 31\,k\Omega$

Table 1: Overview over the specification of the DAC design. The listed power is for the cases of $V_S = 0.4\,V$ and $V_S = 1.2\,V$, respectively .

Overall, from what is shown in table 1, the designs are quite similar in the shown specifications, trading size for amount of outputs and power draw. The true advantage is in invariance of the voltage levels relative to V_S, which was demonstrated in the beginning of this section.

3 Phase Distribution network

As mentioned in the introduction, this work is concerned with data detection through a clocked sense amplifier locking to one or the other state depending on the inputs. In the previous section 2, it was explored how to provide a stable reference voltage such that the sense amplifier locks into the right state. Another aspect is, since the amplifier gets reset for half of one clock cycle then 'reads' the data for a short time and holds it until a new reset signal, the clock phase must be chosen such that the read time avoids the edges of the data signal. If different phases of the clock are available, the optimum can be found through the minimization of the data transmission error. These different sampling points in time, in the following referred to as clock phases, or even just phases, can be provided by a DLL. The power consumption of the shunt capacitor DLL implemented for the test chip, explicitly designed for the I/O bank this work is concerned with, is roughly comparable to half the power consumption of one I/O cell. To avoid having one DLL per I/O cell, in this work the option of transporting the output signals of the DLL, while maintaining the phase offset between them, will be explored. In this, it is not important to maintain the phase offset relative to the source clock, as the output phases of the DLL span approximately one clock cycle.
An immediate way to achieve the distribution of the clock signals is to have long metal lines, here referred to as "global" lines, spanning all the I/O bank, each carrying one of the clock phases, and being driven by a large driver right next to the DLL. On each I/O cell, there is then a clock multiplexer buffering, which can be thought of as copying, one of clock phase for local reference. The circuit for this is depicted by fig. 16. The goal of this section is to present and simulate a working implementation of the phase distribution network. How to then expand on it, will be explored in the following subsections.

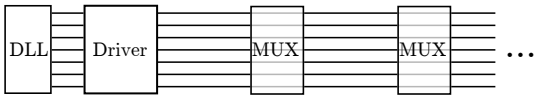


Figure 16: Abstract cicuit for clock phase distribution. MUX is short for multiplexer

For implementation, the driver was realized as two simple banks of CMOS inverters in series, with the first one acting as a predriver. For the multiplexer, a bank of inverting three-state buffers were used. Both choices were made for their simplicity. The specific circuit for slice of the network, corresponding to one of the global metal lines, is shown in fig. 17.

The main challenge with the distribution of the phases is the capacity of the global metal lines. A parasitic extraction (PEX), meaning automatic generation of circuit parts introduced by how the metal like the global lines is routed, can be performed on them. The global lines are minimally wide with 6 times the minimum distance between them. With a total length of 500 µm, they span all the I/O bank. Performing the extraction yielded about 64fF of

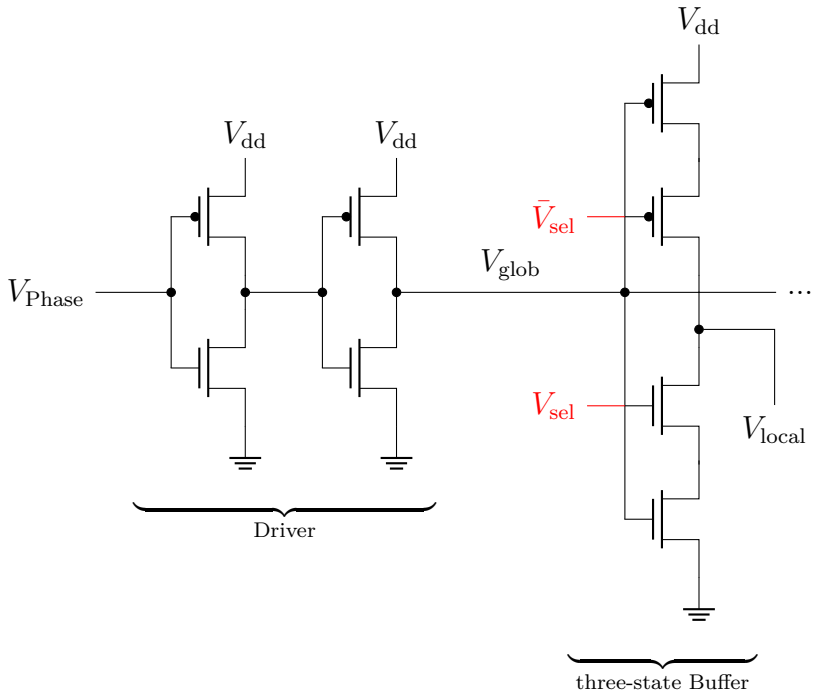


Figure 17: One slice of the phase distribution network, with highlighted drive and buffer. The input from the DLL V_{Phase}, the supply voltage V_{dd}, the potential on the global line V_{glob} and one of the local refeerences V_{local} are displayed. If the inversion of the signal V_{glob} is buffered to V_{local} is determined by V_{sel} and $\bar{V}_{\mathrm{sel}}$.

total parasitic capacitance and a total resistance of 340Ω per metal line. The lower limit on the response time of $t = RC = 22\,\mathrm{s}$ is still small compared to 333 ps, which is half of one period at 1.5 GHz, the limit of what the I/O bank can be expected to be operating at. The power consumption of charging and discharging the parasitic capacitance at a voltage of $V_{dd} = 1.2\mathrm{V}$ for all the eight parallel global lines is,

$$8 \cdot \frac{1}{2} C V^2 \cdot 2f = 1.1\,\mathrm{mW} \tag{5}$$

making it about $\frac{1}{8}$ of the power consumption of the whole I/O bank. Therefore, the drivers have to be correspondingly large. To still keep them compact, LVT transistors, meaning having a low threshold voltage V_T, were used. If this choice is optimal is not entirely clear, however. It will be discussed in section 4. Another design choice not yet mentioned is that the driver was isolated in p-well inside a deep n-well. All this does is separate the bulk of the NMOS transistors of the driver from the rest of the I/O bank. This is necessary because the comparatively high current changes through the NMOS transistors of the driver couple back capacitively into the surrounding bulk. The placement in a deep n-well effectively adds two diodes operated in reverse direction[5], between the bulk potential around the driver and the bulk potential of the rest of the chip. Going back to the motivation through power consumption, the power of the actual PEX Design using these design choices and at the same frequency of 1.5 GHz is then

$$P = 1.4\mathrm{mW} \tag{6}$$

Which is still about a factor of three better than using many DLLs, but 27% higher than above. [6].

3.1 Simulation

Going into more detailed testing, the circuit will in the following be simulated at 1.5 GHz, which is the maximum frequency the circuit can be expected to be operating at. At the same time, the circuit will be tested where the clock phases are the minimum distance apart, the worst case for possible errors introduced by the circuit. This is the case when they span half of one clock cycle, which is one of output modes of the DLL and most demanding on the phase distribution network. Going piece by piece, the different far corners[7] of the signals on the global metal lines, output by the driver with ideal inputs, can be seen in fig. 18.

From the plots in fig. 18, it can be gleaned that both the high and low voltage are reached comfortably. The duty cycle, the time the clock is high over its

[5]for the porpusos of this work, this can be seen as equivalent to non-conducting

[6]The number in (5) is also a low estimate in the sense that for an ideal signal, the capacity between global lines is charged and discharged at every clock transition, doubling its contribution.

[7]See section 2.1 for the explanation

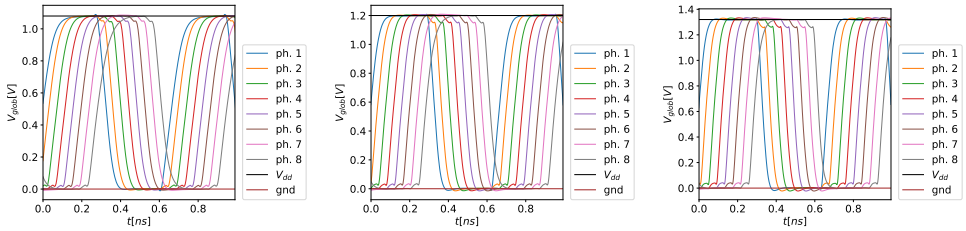


Figure 18: The Driver portion tested in the different far corners. The voltage on global lines V_{glob} is shown for one and a half clock cycles.

period, of about $48.5 \pm 0.2\%$ for the slow corner and tends towards 50% for the fast corner. The error in the duty cycle is still small compared to the phase difference between two phases of 12.5%.
One noticeable aspect is that the first and last clock phases are delayed compared to the others. Being more specific, the differences in when the clock phases cross $V_{\text{dd}}/2$, relative to the expected difference, can be seen in table 3.

Phase difference	2-1	3-2	4-3	5-4	6-5	7-6	8-7
rising edges	0.80	1.00	1.00	1.00	1.00	1.00	1.05
falling edges	0.82	1.00	1.00	1.00	1.00	1.00	1.03

Table 2: Differences between the different clock phases for rising and falling edges relative to the ideal value of 333ps/8 = 41.67ps

From the table, it is clear that the first phase and last phase experience an especially large delay, though this effect is only strong in the slow corner. In the fast corner, this effect is about half as pronounced.
An immediate way to try and address this would be to try and switch around which phase is on which metal line. In the simulation up to this point, the first metal line was used for the first phase, the second for the second and so on. For fig. 19, the phases were assigned to the metal lines differently. Going through the metal lines from one side to the other, the phases $1, 5, 2, 6, 3, 7, 4, 8$ are connected, aiming to maximize the phase difference between adjacent lines. In this case, as can be seen in fig. 19 and table 3, the delays stay similar. The cause for this is therefore something else and will be discussed further below. It is worth mentioning, however, how the plateau of the voltage curves is noisier than in the case where similar phases were adjacent. This is because of the coupling between two adjacent metal lines, causing the first four phases to overshoot, because the last four phases start rising or falling. In the same way, the second four phases undershoot because the first four phases, with which they are interleaved with, start falling or rising.

Another cause for shifts in the relative phases occurs through mismatch between the different transistors. The mismatch simulation can be done directly, but for increasing simulation speed, a schematic, including the most important parasitics, was made. The behavior of this schematic is chiefly governed by the capacity of the metal lines to ground and the capacities between each other,

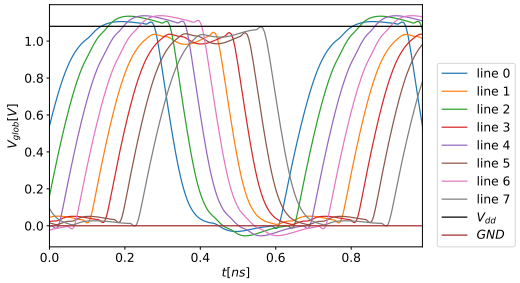


Figure 19: The transient signal for the slow far corner with a different metal-phase assignment. Going through the global lines from one side to the other, the phases $1, 5, 2, 6, 3, 7, 4, 8$ are assigned.

Phase difference	2-1	3-2	4-3	5-4	6-5	7-6	8-7
rising edges	0.80	1.00	1.00	1.00	1.00	1.00	1.05
falling edges	0.82	1.00	1.00	1.00	1.00	1.00	1.03

Table 3: Differences between the rising and falling edges relative to the ideal value of 333ps/8 = 41.67ps, for a phase-metal assingment of $1, 5, 2, 6, 3, 7, 4, 8$. Meaning that going through the global lines from one side to the other, the phases $1, 5, 2, 6, 3, 7, 4, 8$ are assigned.

as they are the largest by an order of magnitude. The model used for part of the metal line in between the different multiplexing parts can be seen in Fig. 20.

Simply emulating the metal lines using multiple of these segments by slotting in the values from the parasitic extraction and increasing the capacitances around 10%, the transients compare as in fig. 21. The plot shows that the slew rate could be well-matched, which is the important part for the differences between the phases. There are also some effects not accounted for, best seen right before the actual clock edges. An explanation for these is the metal lines leading from the pre-stage to the end-stage of the driver coupling to the global metal lines.

Using this to do a mismatch simulation, fig. 22 is obtained. As can be seen in it, the large size of the transistors, in combination with the high slew-rate[8]

[8]slew-rate is the incline of the clock edges

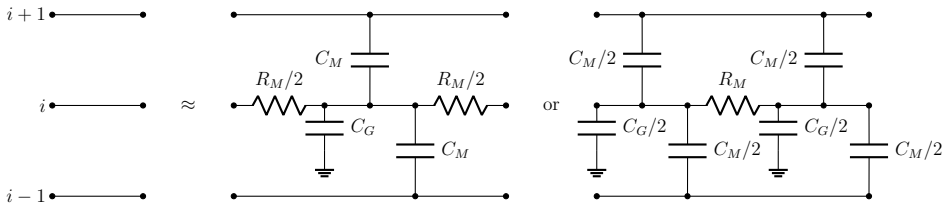


Figure 20: A first order approximation for the global metal lines, including their capacity to ground and to the other metal lines.

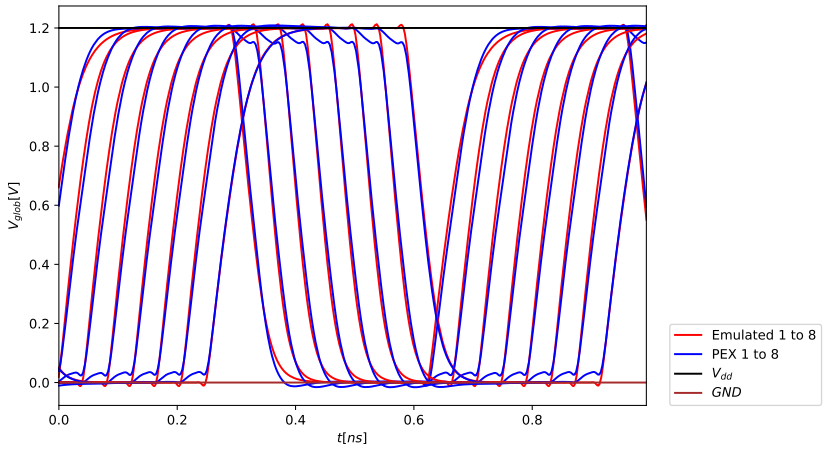


Figure 21: Comparison between the transients of the PEX extracted desgin and a schematic incorporating the most important parasitics.

makes for a low mismatch variance.
Specifically, rounded to the last digit, the standard deviation of the distributions, for the inner metal lines shown in row 2 to 7 of the plot, is 0.43 ps for the falling edge and 0.42 ps for the rising edge. For the metal lines on the edges, the standard deviation is a bit larger with 0.47 ps for the top most plot for both rising and falling edges and 0.48 ps for the bottom most metal line and both kinds of edges. The the difference in the standard deviation can be explained through the different slew-rates for the different metal lines, since the crossing time is not directly affected by the mismatch, but really the gate size affecting the driving strength.

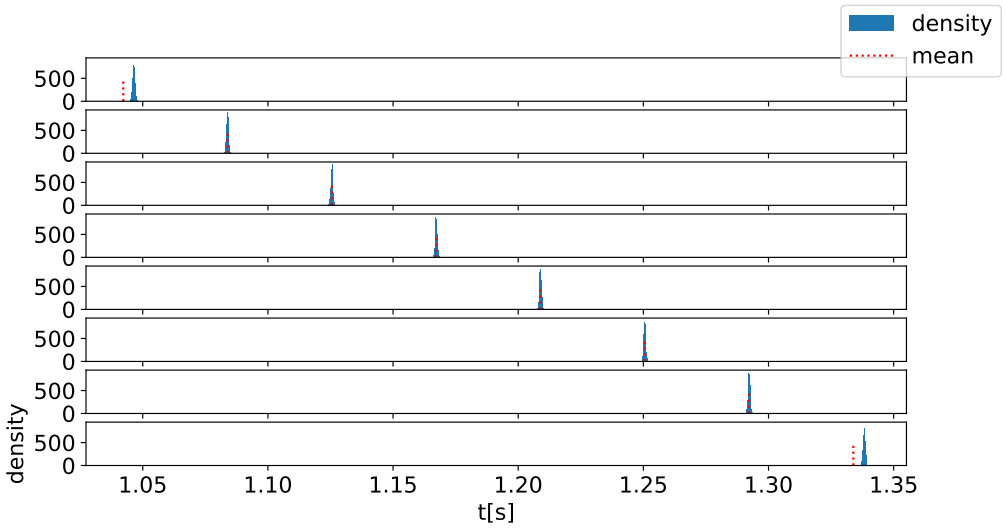


Figure 22: $V_{\mathrm{dd}}/2$ crossing points for a mismatch simualtion with 3000 samples

Moving on to the multiplexing part of the circuit, the signals take the shapes as in fig. 23 for the different corners. By the fact that not all clock phases reach

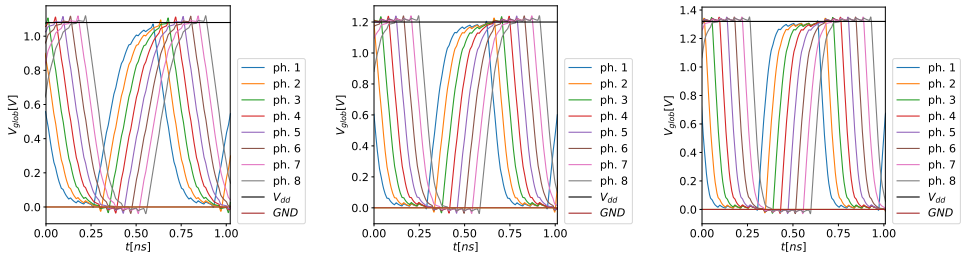


Figure 23: Far corner Simulation of the multiplexing part of the circuit

the supply voltage or ground and the duty cycle goes down to 46% in the slow corner, this part of the circuit can already be assessed to be the bottleneck. Additionally, there is a sawtooth like signal contribution added to the clock. This can be explained through the ideal clock sources coupling capacitively into the output.

To explain the behavior fully, three parasitics are important. The first is the output capacity of the three state buffers, an upper bound of which is estimated to be around 2 fF [9]. The second one is the capacity of the output to ground, extracted to be about 1.2 fF. Lastly, the capacity between the output and each of the respective inputs, being 90 aF each.
To confirm this, another schematic, incorporating these three parasitics, was made. The circuit used, and how it compares to the design with the extracted parasitics can be seen in fig. 24.

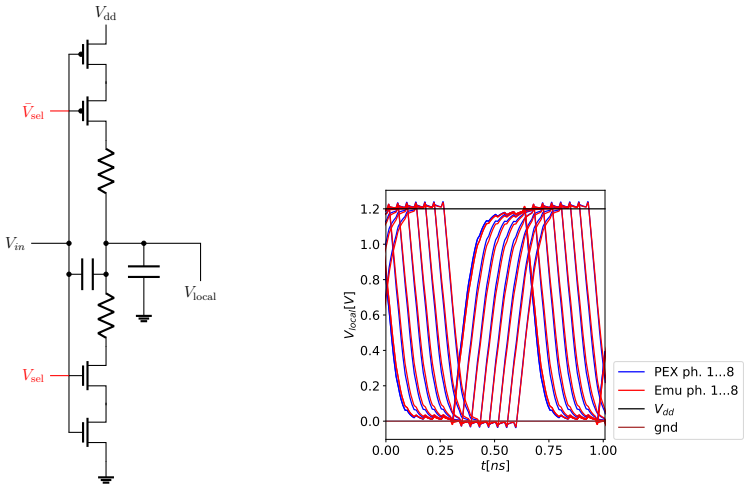


Figure 24: A schematic incorporating the important parasitics and how close it matches the PEX extraction.

Using the circuit from 24, two things were confirmed. First is that the sawtooth like noise can be eliminated by eliminating the capacity between the inputs and output. This capacity could be somewhat reduced by rerouting the signals, placing less emphasis on how small the design will be.

[9]This estimate was made by looking at the capacity of a MOSCAP and scaling the capacity down in accordance with the gate size of the three state buffers.

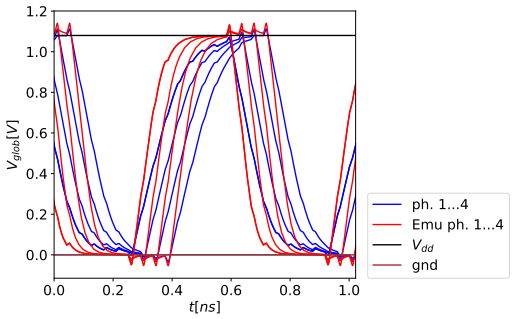


Figure 25: Schematic emulation, including the largest parasitics, of 4 three state inverters (red), compared to 8 three state inverters (blue), each with connected outputs. Four sequential phases are compared.

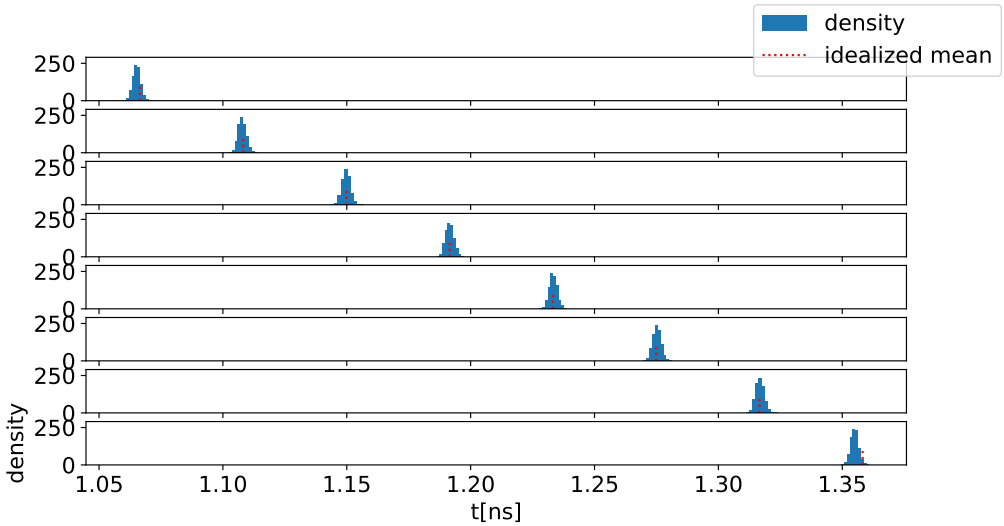


Figure 26: Mismatch for the schematic emulation of multiplexer for times of crossing $V_{dd}/2$ in the typical far corner. The multiplexer is being ideally driven in this case. The idealized mean is computed by adding or substracting 41.67 ps from the mean of phase 4 the corresponding number of times.

Secondly, that to improve upon the slew rate and duty cycle, the architecture of the MUX has to be changed. This is due to both the output capacity of the three-state inverters and the capacity of the metal line to ground being tied to the eight outputs of the three-state inverters all being connected in parallel. An improvement of this could be achieved through a tree structure. How the transients would then hypothetically look when having 4 three state inverter outputs connected, instead of 8, is shown in fig. 25.

The disadvantage of this would be the addition of another stage, introducing one more level of mismatch and delay. The details can be explored in future work.

The mismatch for the current configuration can be found in fig. 26.

As can be seen in 26, the standard deviation is larger, about 1.7 ps. As for fig.

23, there is a systematic offset to the clock phases on the outer metal lines. For the metal line on the top of the plot in 26, the offset is 3.2% of 41.67 ps to the right of the idealized value. For the bottom most line, it is 7.6%.

When combining both, the overall signal remains similar to the one observed with just the MUX. However, the sawtooth component of the signal is noticeably reduced. Also, the duty cycle is slightly improved, as the shifts cancel each other out.

To look at the effect of all this deliberation on the final output that is used locally, an inverter is attached to the output of the MUX, akin to how it is implemented physically. Also, separately, for a more realistic power delivery simulation, a circuit was added between the ideal power supply and the corresponding pin in the circuit, emulating the bondwire delivering power to the chip. The circuit used is shown in Fig. 27 The sizes of the inductor was chosen using

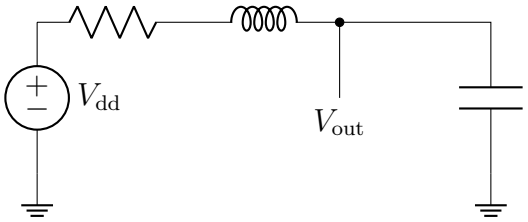


Figure 27: Model used for power delivery through a bondwire

the approximation of $1\,\mathrm{nH\,mm^{-1}}$ to be $< 1\,\mathrm{nH}$, corresponding to the short bondwire the chip is bonded with. For the resistor, a resistance of $1\,\Omega$, representing the combination of outgoing resistance for the DC voltage source, resistance to the start of the bondwire and the bondwire itself, was used. The capacitance implemented on the chip is larger than 30 fF. This changes the output transients and the current supplied by the supply voltage in the way as seen in fig. 28.

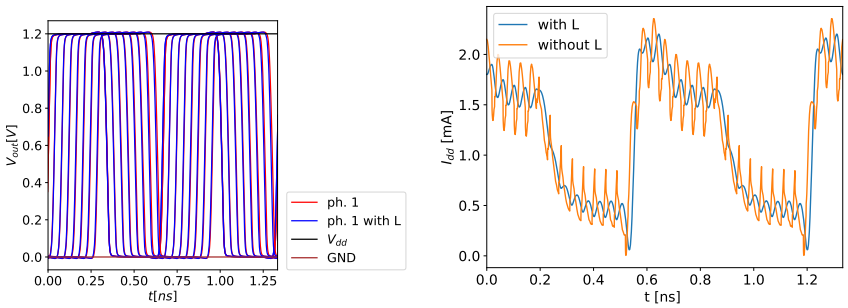


Figure 28: How the transients and current draw from the supply voltage change when accounting for the inductance of the bondwire.

As can be seen in fig. 28, in this simulation the duty cycle of the clock cycle gets slightly improved for the later clock phases, making for a better distribution of the phases across time. Moving on to a mismatch simulation of the whole network, fig. 29 is obtained. The standard deviation has ballooned to up to

3.8 ps, up from 1.7 ps and 0.5 ps for the MUX and driver respectively. The monotony is still maintained within $41.67\,\text{ps}/3.8\,\text{ps}/\sqrt{2} = 7.8 > 7\sigma$, but as monotony is a low bar and other effects like jitter and corner are not accounted for, a better result would be desirable.
The increase in mismatch is not due to the introduction of power through the bondwire, as similar numbers were obtained for a smaller mismatch simulation of 50 samples without this modification. Examining the cause more methodically, looking at the signals after the driver, the MUX and the final inverter revealed that the standard deviation grows from about 0.4 ps after the driver, to about 3 ps after the MUX, to about 3.5 ps after the final inverter. The main cause must therefore be three state inverters of the MUX being impacted by the capacity of the additional inverters that were attached, highlighting the bottleneck of this network once more.

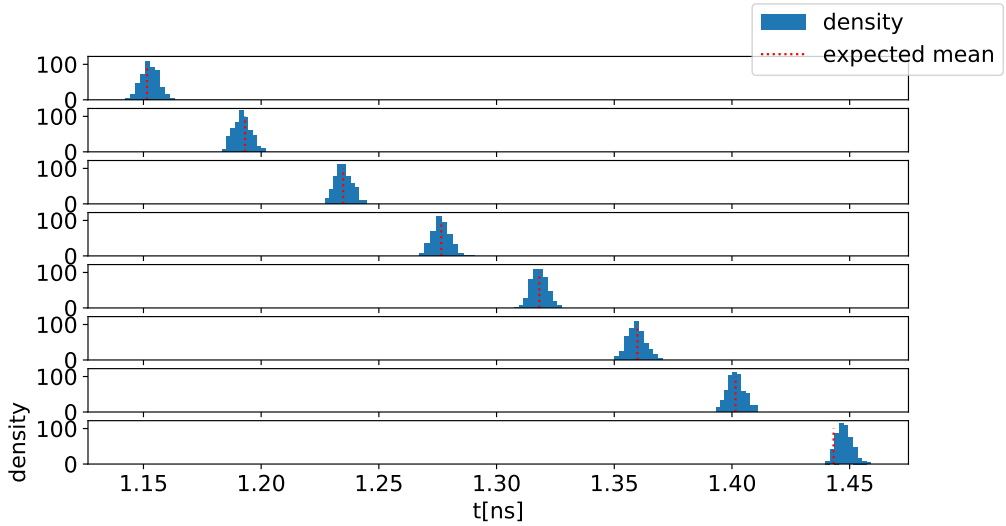


Figure 29: Mismatch simulation for 300 samples for the emulation of a parasitic extraction using a schematic with the parasitics. The expected mean is derived by taking the mean of phase 4 and adding or subtracting 41.67 ps the according number of times.

Lastly, it should be investigated if the phase supplied to the sense amplifier needs to change under change in temperature or supply voltage. This will be tested around the typical corner at 1.2 V and 55°. The delay the phases pick up under change of these paramters can be seen in fig. 30. Going down to 1.08 V delays the phases by 31 ps while increasing it to 1.32 V speeds them up by 21 ps. Increasing the temperature to 80° increases the delay by around 3 ps, while decreasing it to 30° decreases it by the same amount. The temperature variation, derived from the lab condition, is therefore practically irrelevant.

For only requiring one phase setting for multiple conditions, the portion of the clock phase that stays constant under the parameter shifts is important. The specific sense amplifier used requires this window to be at least 40 ps[10]. The

[10]This is the combination of ”setup time” and ”holdtime” for the sense amplifier.

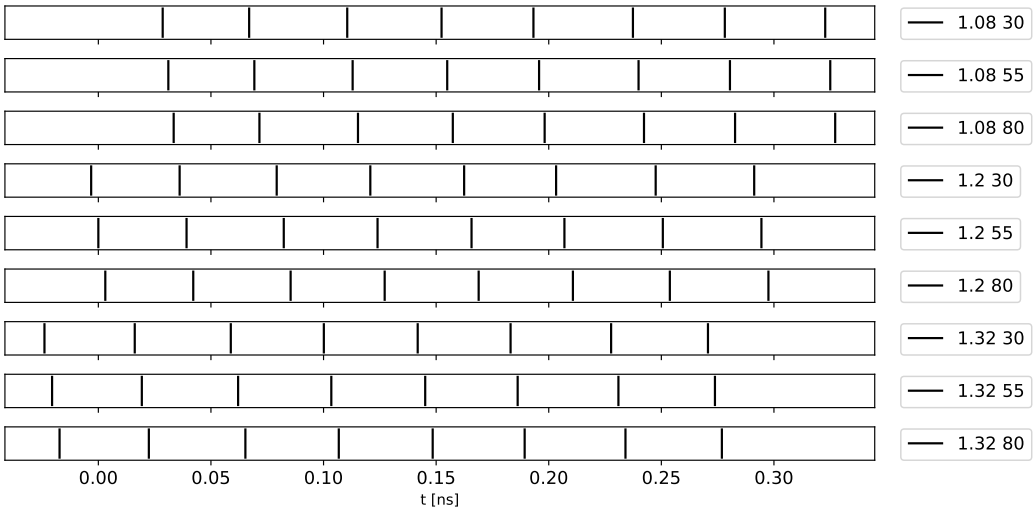


Figure 30: The shift the phases experience under a change of supply voltage and temperature. The numbers in the legends stand for the tested supply voltages ($1.08\,\mathrm{V}, 1.2\,\mathrm{V}, 1.32\,\mathrm{V}$) and temperatures ($30°, 55°, 80°$)

window that remains constant can be directly computed as the time of half of one clock period minus how much the parameter changes shift the clock edges. However, for an accurate computation, more effects like duty cycle, the amount of phases, extent of the clock edges, jitter[11], and the additional buffers before and after the distribution network have to be taken into account.
As most of these effects are beyond the scope of this work, just a reference will be given of how the clock edges shift with changes in the supply voltage in fig. 31.

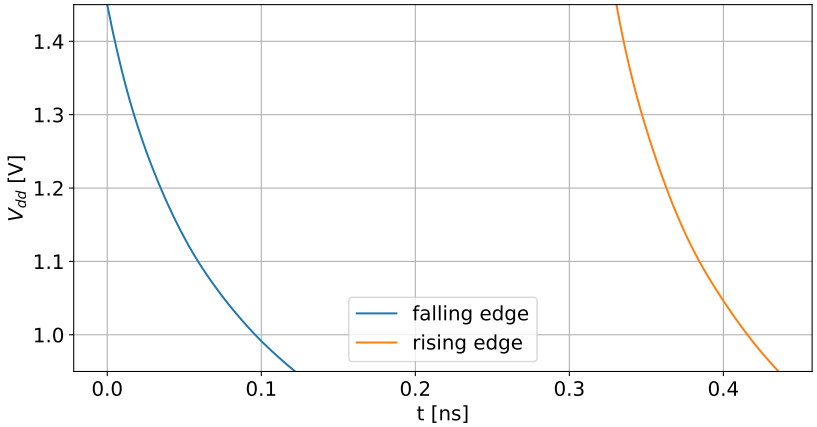


Figure 31: The clock edge timeshift in the phase distribution network due to a change in supply voltage V_{dd} for the rising and the falling edges

A final overview of the network can be found in table 4.

[11] this describes the deviation from a truly periodic signal

power draw (s)	1.09 mW
power draw (t)	1.41 mW
power draw (f)	1.78 mW
size of MUX	13 µm^2
size of Driver	110 µm^2

Table 4: Overview over the clock distribution network. The power draw and in all the far corners as well as the sizes of the core parts of the network are listed.

Summarizing this section, a successful and efficient implementation of a clock distribution network could be achieved.

3.2 Measuring phase, counteracting mismatch

To compensate for the mismatch and systematic errors due to the structure of the circuit, there is the option of adding three-state inverters in parallel to each of the drivers, effectively increasing the possible current through the combination of them. The most simple variant of this is shown in fig. 32. The version shown abstracts away some detail. It shows the V_{in} going into an inverter, functioning as a predriver in this case. The output of the predriver is broadcast to the different components of the driver stage, consisting of the actual driver and some three state inverters for adjusting the slew rate. What the schematic does not show is that S_1, S_2 have to be buffered once. This is because the signal bits would in this case arrive though long metal lines, and if not buffered, would spread the noise from V_{out}, which they are capacitively coupled to, on all the singnal line. Since S_1, S_2 need two inverters to be buffered $\bar{S}_1$ and $\bar{S}_1$ will also be directly available through these making the inverters from the three state buffers unnecessary.

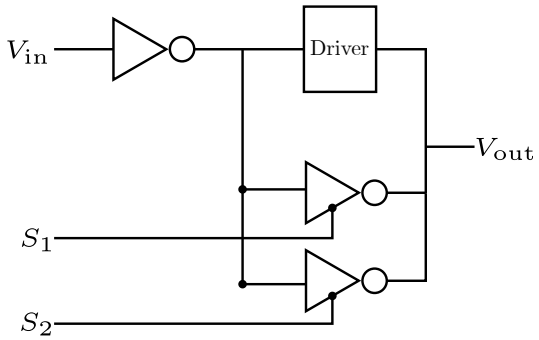


Figure 32: An abstraction of a slice of the phase distribution circuit allowing for compenstion of mismatch and systematic errors.

The current iteration of the driver was designed while keeping in mind this possible extension, by making the driver slices wide enough. Moreover, this would work best with the additional extension for the measurement of the phase timing error. This can be addressed in a future iteration of the chip.

3.3 Outlook on expanding the flexibility

The test-chip the circuits of this work were developed for uses the phase distribution network from section 3. However, there was some work for expanding on its flexibility. It will be presented here.
The approach highlighted in the previous section has some limitation on what originally made the work in [4] stand out. Its strength was in its high flexibility and redundancy, where the bank was not restricted to a certain size and many of the manufacturing errors in the I/O cells could be accounted for by the redundancy added into them.

With the highlighted approach of the previous section, especially the flexibility in terms of the size of the bank and also the options in case of failure are limited. The size of the I/O bank requires some work to alter, as the length of the global metal lines varies with it. This of course adds to the resistive decoupling and, more importantly, to the total parasitic capacitance of them. This also means that the drivers will have to be scaled accordingly. To overcome these restrictions, two simple changes were proposed:

- instead of having one DLL for 10 I/O cells, place them on the bank more frequently, i.e. every one two or more I/O Cells
- break the metal lines such that there is at most one DLL per continuous metal line and add the option to transfer the signal from one metal line to another.

These come with some immediate advantages and disadvantages. The main advantage is of course the flexibility allowing for more widths of the I/O bank and the redundancy in case of a failure of a DLL. On the neutral side, is that the total length in metal should stay approximately the same, as it will just be subdivided, implying that the power due to phase distribution will also remain the same. [12]

The main disadvantage will be that the clock phases will have to travel through more buffers, introducing more jitter and mismatch. This has the implication, that the clock phases will not be able to travel as far in the flexible case. For a whole I/O bank, more DLLs will therefore be activated on average and the power draw will increase. Moreover, the redundant DLL and will take up more space. The same about space is true for the driver, which will be effectively be divided into multiple smaller three state buffer which are, per width, about half as strong. [13]

[12] Technically, if two I/O cells receive the phases from different DLLs, the metal between them does not have to be powered. The possibility of this depends on the implementation of the transfer circuit in the I/O cell, however.

[13] The three state buffer needs two gates in series, compared to one for just and inverter or buffer

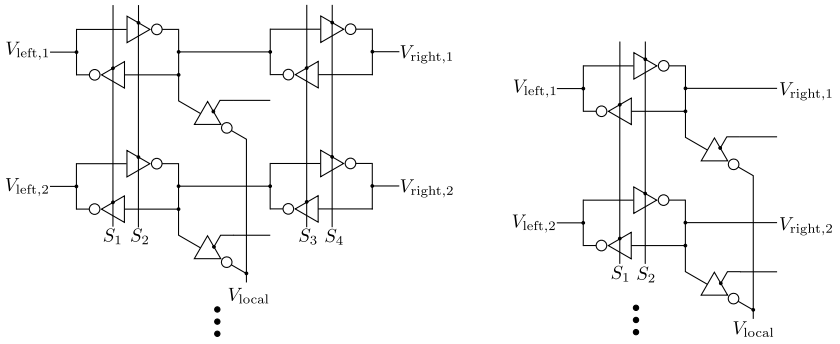


Figure 33

3.3.1 Phase transmitter designs

This work looks at the circuit for transferring the phases from one set of metal lines, to another. Two designs were tested for this case, they can be found in Fig. 33. As they get mentioned quite often, they will be called design "F" for flexibility for fig. 33a and design "S" for simplicity for fig. 33b.

The two designs offer different levels of configurability, they are broken down in Table 5. The first observation that can be made from 5 is that many of the

S_1	S_2	S_3	S_4	meaning in F	meaning S
0	0	0	0	power down	power down / receive right
0	0	0	1	invalid	
0	0	1	0	receive right	
0	0	1	1	invalid	
0	1	0	0	receive left	send right, receive right
0	1	0	1	recv l, send r	
0	1	1	0	invalid	
0	1	1	1	invalid	
1	0	0	0	invalid	send left, receive right
1	0	0	1	invalid	
1	0	1	0	recv r, send l	
1	0	1	1	invalid	
1	1	0	0	invalid	invalid
1	1	0	1	invalid	
1	1	1	0	invalid	
1	1	1	1	invalid	

Table 5: The possible signal combination for design F and S with their interpretation.

configurations are invalid, especially so for design F. By omitting the "power down" state, only four states remain for design F, enabling a simplification to two bits meaning "direction left/right" and "disable/enable". It can also be gleaned, that design F can always receive phases without forwarding them,

this is only the case for design S if the signal comes from the right direction. Since the cases where a signal will not be needed to be forwarded is interleaved with a DLL being powered on, however, the power saving in relation to the consumption of the DLL and the rest of the metal lines cannot be high. Another difference is that the capacitive load for the driver of design S will be higher than in F, as for S, three drivers and one receiver have to be driven, while in case of F it is one driver and two receivers. A back of the envelope computation shows that this could add as much as 20% in capacity when scaling down from a MOSCAP, but as the additional drivers will be non-conducting, this is likely much less.
Lastly, the design S is smaller than F, as the latter contains the driver part effectively twice. This can be somewhat offset by scaling down the receiving part of the circuit. Most importantly, the design S also inverts the clock phases when transferring them, F does not. This means that to stay consistent between the local signals between different I/O cells, an option to invert the locally received signal needs to be present. Specifically, this could be done by giving the option of using a three state inverter of a three state buffer on the signal chosen by the MUX. This circuit is shown in 34. For design F, only one of the two would be necessary. However, with the current DLL having a mode where the phases only span one half of the period, this circuit is implemented nonetheless, as it allows for the local generation of clock phases spanning the other half of the period.

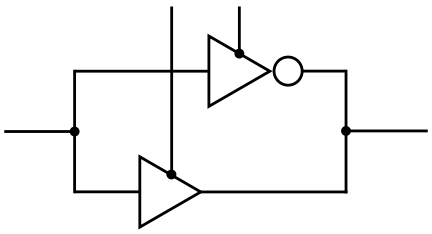
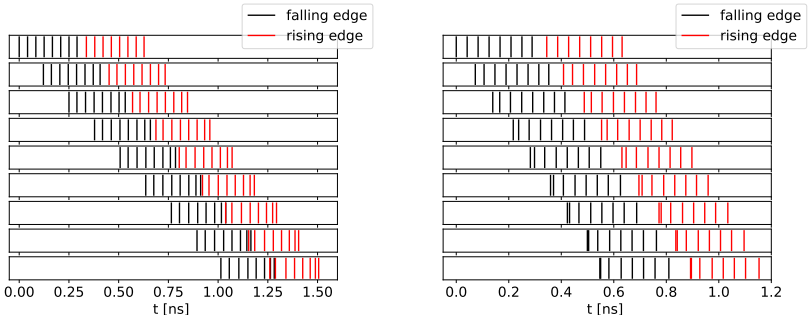


Figure 34: The local choice between inverter and buffer to keep the received I/O cell clocks conssitent.

3.3.2 Phase transmitter Simulation

The interesting aspect for this section is how the drivers behave when chained together. A design was made for both approaches, where the metal lines were broken on each I/O cell and one phase transmitter was used to connect them. To save space with the design F, the receiving three state buffers were linearly scaled down in width by about a factor of two compared to the sending ones. They were then PEX extracted, including the metal lines and dummies on the edges. They were then put in a chain of eight transmitters in a separate schematic. The same was done for design S. The results after each of the transmissions are shown in 35. Effectively for design S, about one phase is lost, making it the phases span 263 ps in the falling case and 260 ps in the rising case, making them about 30 ps smaller than is optimal. The phases for design F look much worse, the rising and falling edges only span a range of

492 ps, 132 ps less than the optimal value of 624 ps.



(a) Crossing times for a chain of transmitters of design F

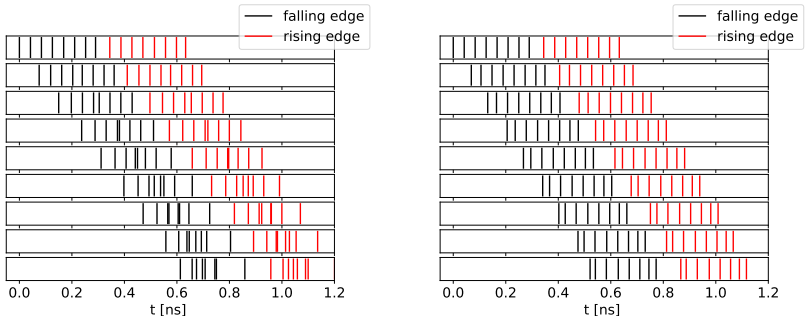
(b) Crossing times for a chain of transmitters of design S

Figure 35: The phases along different parts of the chain. Eight segments were simulated, making for 9 segments when including the input into the first transmitter. From top to bottom the times are shown where the input phases, phases of after one transmitter, phases after two transmitter and so on crossed $V_{\mathrm{dd}}/2$, in this case 0.6 V.

Two effects can be observed.
First, for the approach in design F, the falling and the rising edge are moving closer and closer with each transmission. This is because of the different duty cycle of the sending and receiving part of the circuit. If the duty cycle were the same, the error would cancel out because the error was applied first to the signal and then the inverted signal. For the design F however, the duty cycle is not the same between the sending and receiving parts, both because the scaling down of the receiving part affects the driving strength of the NMOS and PMOS differently, but also because the capacitive load is different. This is a flaw of the design.
The second one is that the phases seem to move somewhat closer to each other for both approaches. This is harder to get behind.
To investigate this a bit further, the approach of design S was tested a bit further, since here, the other effect is not present. First a different assignment of the phases to the metal lines, and removing the dummies, was attempted. The results are shown in 36.

From 36 it can be gleaned that changing the port order only makes the problem worse. For the removing of the dummies, it seems to slightly worsen the situation for the late phases, while a slight improvement for the early phases can be observed. Adding back the dummy line next to the last phase then results in 37.

The phases in 37 are transmitted the best yet among the tested approaches. A possible hypothesis could be the following: As parasitic capacitances between two lines don't have to be charged when they are completely aligned, this state is therefore energetically lower. Repeated interaction through transmission therefore has an attractive effect on two adjacent phases. This fits well with



(a) Assinging phase 1 to line 1, 5 to 2,

(b) After removal of the dummy lines

Figure 36: The phases after (0,1,2, ..., 8 from top to bottom) transmissions using design S and changes to the original implementation.

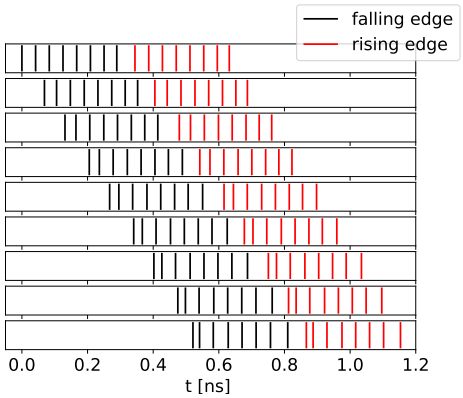


Figure 37: The phases after (0,1,2, ..., 8 from top to bottom) transmissions using design S, with the dummy metal line next to first line/phase removed.

what is happening in 36a, as here, there is less alignment of the clock edges spanning roughly 100 ps, making for "more energy being stored" in this state and therefor more potential for chaotic movement of the clock edges. However, to be certain of this, more quantitative statements to be tested about the shifts need to be made. This was not achieved within the timeframe.

The final check for this work into whether design S is feasible as a design was done through a mismatch simulation, making sure that the phases don't deviate too much from where they are in the typical case. The results of which are shown in 38.
It shows the standard deviation growing from transmission to transmission, but not evenly, as notably the last phase grows by about 1 ps with every transmission, topping out at 7.3 ps. In contrast, the standard deviation of phase 4 grows as shown in the Table 6. It shows that the variance also does not grow linearly, highlighting that the process is not Gaussian because of the mutual interaction.

All in all, the signal quality still drops too quickly for implementation in an actual chip and further research would have to be performed.

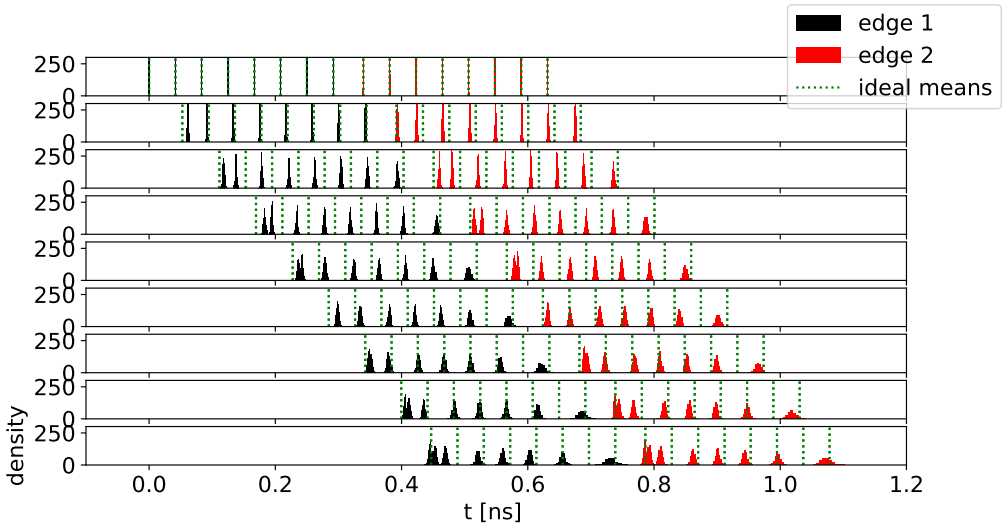


Figure 38: A mismatch simulation for 1000 samples of a schematic for design S including the dummy lines and the parasitics of the metal lines. The "ideal means" are the mean crossing times of the input phases shifted to to the range of the phases after the transmission.

transmission count	1	2	3	4	5	6	7	8
σ ps^2	0.72	1.57	1.80	2.19	2.26	2.70	2.78	3.18
σ^2 ps^2	0.53	2.45	3.24	4.79	5.11	7.25	7.71	10.13

Table 6: The standard deviations for phase 4 after $1, 2, ..., 8$ transmissions.

4 Conclusion and Outlook

This thesis is focused on providing better reference signals to the sense amplifier of an I/O interface.

The first step was providing a reference voltage to distinguish between a high and low signal at configurable signal swing through a DAC based on a resistor chain tapped at different points. The design for this work was introduced in section 2. It operates at a power level of between 2.4 µW and 21 µW, depending on the signal swing and occupies and area of 136.5 µm^2. Thereafter, the DAC design was simulated in subsection 2.1. The most important properties of the DAC were demonstrated at the beginning of this subsection. They are the proportionality of the output levels to the signal swing and the invariance of the output levels under change in corner, temperature and supply voltage. Afterward, the impact of the kickback from the sense amplifier to the reference voltage output by the DAC was investigated. It is highly dependent on the capacitor attached to the reference voltage. Two results were obtained for the capacitor used in the current iteration. Firstly, for a full swing signal, the maximum kickback to the reference voltage could be suppressed to under half of the distance of two output levels for most of the output range. Second is that the maximum kickback at lower signal swings exceeds half of the distance

of two output levels in most cases. Concluding the section, the worst case response time of the DAC, with the attached capacitor was measured to be 12.67 ns.

All in all, the design was shown to fit the application quite well and improves substantially on previous work in its output levels. Looking ahead, the results for the DAC will be able to be physically verified when the test-chip, submitted for fabrication at TSMC, is produced. As for changes for the next iteration, the main improvement would be enlarging the capacitor attached the DAC. Another potential change, is whether to expand the DAC by an additional transmission gate tree, enabling it to supply the two sense amplifiers of each I/O cell with independent reference voltages. The DAC was already implemented this way on the submitted test chip. If this makes a difference will also be tested for after the test chip is obtained.

The second part of this thesis is about transporting phases of a reference clock, produced by a DLL, to every I/O cell by a simple distribution network developed in this work. This is what section 3 focuses on. First, the network, implemented on the test chip, consuming 1.41 mW in the typical corner is tested part by part. The results were that transporting the phases to metal lines spanning all the I/O bank using the designed driver was achieved with very low mismatch. The bottleneck of the circuit was afterward identified to the multiplexer on each I/O cell, copying one of the signals for local reference. The final clock edges had a mismatch of 3.8 ps, maintaining monotony within 7σ.
Next, the phase edges transported by the network were then tested for timeshift under the change of supply voltage and temperature. It was found that any change in temperature, under the lab conditions the chip will be operating at, will not cause a shift exceeding 7 ps. The supply voltage, on the other hand, can cause a change of over 100 ps in the tested region of 0.95 V to 1.45 V. The detailed results provide an important reference for how stable the supply voltage will have to be.

This will presumably find application already when the manufactured test chip arrives back. The possiblities for testing for various kinds of noise will the also be opened up. Also, for future iterations, the implemented design offers some worthwhile avenues for optimization. The first is expanding on the implemented multiplexer by giving it a tree like structure. Promising preliminary results were obtained for this. Another simple one is the change of the LVT transistors of the driver to normal ones, offering potentially lower power draw.
Subsection 3.2 and 3.3 are concerned with other avenues for improvement. In the first, a possibility for calibrating the network is presented. In the second, how to expand on the network's flexibility and redundancy is explored. One promising approach was worked out.

In the end, this thesis contributed to a more stable I/O interface, a small part

in the BrainScales-2 architecture.

5 References

References

[1] Manon Dampfhoffer **andothers**. "Are SNNs Really More Energy-Efficient Than ANNs? an In-Depth Hardware-Aware Study". **in***IEEE Transactions on Emerging Topics in Computational Intelligence*: 7.3 (2023), **pages** 731–741. DOI: 10.1109/TETCI.2022.3214509.

[2] Christian Pehle **andothers**. "The BrainScaleS-2 Accelerated Neuromorphic System with Hybrid Plasticity". **in***Front. Neurosci.*: 16 (2022). ISSN: 1662-453X. DOI: 10.3389/fnins.2022.795876. arXiv: 2201.11063 [cs.NE]. URL: https://www.frontiersin.org/articles/10.3389/fnins.2022.795876.

[3] Johannes Schemmel **andothers**. "Accelerated Analog Neuromorphic Computing". **in***Analog Circuits for Machine Learning, Current/Voltage/Temperature Sensors, and High-speed Communication: Advances in Analog Circuit Design 2021*: Springer International Publishing, 2022, **pages** 83–102. ISBN: 978-3-030-91741-8. DOI: 10.1007/978-3-030-91741-8_6.

[4] Joscha Ilmberger **andothers**. "A flexible multi-standard I/O interface for chip-to-chip links in 65 nm CMOS". **in***2024 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)*: 2024, **pages** 424–427. DOI: 10.1109/APCCAS62602.2024.10808294.

[5] Arik Küster. *Adaptable reference voltage supply for high-speed I/O interfaces.* Unpublished internship report. 2024.

Erklärung

Ich versichere, dass ich diese Arbeit selbstständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Heidelberg, den 30.12.2024