UiO : Department of Informatics
University of Oslo

High-Speed Digital Ultra-Low Voltage Floating Gate Design

Master's thesis

Halfdan Solberg Bechmann

May 2014

Abstract

This thesis covers the design, production and measurement of digital ultra-low voltage floating gate logic.

The increasing demand for low-power electronics, fueled by the expanding market for portable devices and the growth of the Internet of things, both with a desire for longer battery life, enhances the importance of low-power logic styles in modern integrated circuit design. This leads to a rising demand for low-voltage design topologies because the simplest way to reduce both the static and dynamic power consumption is to reduce the supply voltage. The cost of this however is a severe penalty to the circuit speed.

The ultra-low voltage (ULV) logic styles used in this thesis are designed to reduce gate delay and increase the circuit speed by utilizing capacitive coupling on the inputs to super-charge the gate terminal of critical transistors and thereby increase transistor current. The presented designs, simulations and measurements of the ULV logic prove the topology to be significantly faster than conventional electronics in ultra-low voltage operation.

To demonstrate the high-speed qualities of ULV logic in hardware and compare its analog properties to conventional logic, a test circuit with an inverter from each topology is designed and manufactured. The ULV inverter used is from the 7^{th} iteration ULV topology (ULV7) and is scaled to drive the capacitive load of a test setup. The conventional inverter is scaled equivalently and placed on the same chip. After production in Taiwan by TSMC using their 90nm Nexsys® process the finished chip produces measurements that show that the theorized and simulated qualities of ULV logic are highly applicable in silicon hardware implementations.

The low propagation delay of the ULV logic makes it ideal for use in adder carry circuits. To utilize the ULV properties, new carry circuits are presented and simulated in this thesis, yielding results that prove them more energy efficient and significantly faster than conventional carry propagation circuits.

Preface

This thesis is a part of the degree Master of Science in Nano- and microelectronics carried out at the Department of Informatics, Faculty of Mathematics and Natural Sciences at the University of Oslo. It contributes 60 credits to the degree.

Acronyms

V_{GS}	Transistor **G**ate-**S**ource **V**oltage
V_{TH}	Transistor **Th**reshold **V**oltage
‘0’	Logical zero, voltage close to GND
‘1’	Logical one, voltage close to VDD
‘Z’	Logical high-impedance
ALU	**A**rithmetic **L**ogic **U**nit
ASIC	**A**pplication-**S**pecific **I**ntegrated **C**ircuit
CC	**C**hip **C**arrier
Cin	**C**arry **in**put
CLK	**Cl**oc**k**
CMOS	**C**omplementary **M**etal-**O**xide-**S**emiconductor
Cout	**C**arry **out**put
CPA	**C**arry **P**ropagate **A**dder
CSA	**C**arry **S**ave **A**dder
DIBL	**D**rain **I**nduced **B**arrier **L**owering
DRC	**D**esign **R**ule **C**hecker
EAGLE	PCB design tool **(E**asily **A**pplicable **G**raphical **L**ayout **E**ditor)
EDP	**E**nergy-**D**elay **P**roduct
FBB	**F**orward **B**ulk **B**ias
FG	**F**loating **G**ate
GND	**G**rou**nd**
GPIB	**G**eneral **P**urpose **I**nterface **B**us
Idsn	**D**rain-**S**ource current of **N**MOS transistor
IMEC	**I**nteruniversity **M**icroel**e**ctronics **C**entre
Isdp	**S**ource-**D**rain current of **P**MOS transistor
JLCC	**J**-**L**eaded **C**hip **C**arrier (ceramic)
L_n	**L**ength of **N**MOS transistor
L_p	**L**ength of **P**MOS transistor
LVS	**L**ayout **V**ersus **S**chematic
MIMCAP	**M**etal-**I**nsulator-**M**etal **Cap**acitor
MOMCAP	**M**etal-**O**xide-**M**etal **Cap**acitor
MOSCAP	**MOS**FET without source and drain connections, used as a **cap**acitor.
MOSFET	**M**etal-**O**xide-**S**emiconductor **F**ield-**E**ffect **T**ransistor
MUX	**Mu**ltiple**x**er
nch_lvt	tsmcN90rf **n**-**ch**annel transistor with a **l**ow **t**hreshold **v**oltage
nch_mac	tsmcN90rf **n**-**ch**annel transistor, Monte Carlo enabled
NMOS	**N**-type **MOS**FET, MOSFET with negative charge carriers (electrons),
PCB	**P**rinted **C**ircuit **B**oard

PDP	**P**ower-**D**elay **P**roduct
PLCC	**P**lastic **L**eaded **C**hip **C**arrier
PMOS	**P**-type **MOS**FET, MOSFET with positive charge carriers (holes)
PTL	**P**ass **T**ransistor **L**ogic
RBB	**R**everse **B**ulk **B**ias
RSCE	**R**everse **S**hort-**C**hannel **E**ffect
RTMOM	**R**otative **M**etal-**O**xide-**M**etal (capacitor type)
RTMOMCAP	**RTMOM Cap**acitor
SMD	**S**urface-**M**ount **D**evice
SMT	**S**urface-**M**ount **T**echnology
TG	**T**ransmission **G**ate
TSMC	**T**aiwan **S**emiconductor **M**anufacturing **C**ompany
tsmcN90rf	TSMC 90 nanometer high-frequency production process, also called Nexsys®
ULV	**U**ltra-**L**ow-**V**oltage
ULVPTL	**U**ltra-**L**ow-**V**oltage **P**ass-**T**ransistor **L**ogic
UV	**U**ltra**v**iolet
Vdd	Supply Voltage for MOSFET circuits (**V**oltage **D**rain **D**rain)
W_n	**W**idth of NMOS transistor
W_p	**W**idth of PMOS transistor

m	$\times 10^{-3}$
µ	$\times 10^{-6}$
n	$\times 10^{-9}$
p	$\times 10^{-12}$
f	$\times 10^{-15}$
a	$\times 10^{-18}$
z	$\times 10^{-21}$
y	$\times 10^{-24}$

Contents

Figures

Tables

1 Introduction

The expanding market for portable devices and the rapid growth of the internet of things creates an increasing demand for electronics powered by lightweight batteries and alternative energy sources. This leads to higher low-power data processing requirements increasing the importance of low-power logic styles in modern integrated circuit (IC) design. The simplest way to reduce the power consumption of a system is to reduce the supply voltage which drastically reduces both the static and dynamic power dissipation at the cost of a higher gate delay and a slower circuit. The increasing use of energy harvesting systems to power lightweight devices also introduces a demand for circuits that are not only low-power but also need to run on an low supply voltage in order to avoid the power overhead of dc-dc converters. The continuous reduction of transistor sizes also causes increased leakage through subthreshold currents and gate-oxide tunneling [1] that can only be reduced by lowering the supply voltage.

To increase the speed while maintaining a low supply voltage, a floating gate logic style can be utilized to increase the transistor current and device speed by super charging the gate of the evaluation transistor but still maintain the low supply voltage needed to tackle sub-threshold leakage and keep the power consumption low.

1.1 Thesis outline

The focus of this thesis will be on the design of the type of ultra-low voltage (ULV) floating gate logic mentioned above.

In the next chapter, the thesis will start off by explaining the theoretical background for the work conducted. This includes Floating Gate Logic, Adders, Sub-Threshold Effects and the Figures of merit used to analyze the circuits.

The thesis will then continue to the design of a test circuit to compare the ULV logic style to conventional logic in implemented hardware and see the correlation with simulation results. A layout of this design is made in section 3.1.3 and produced by Taiwan Semiconductor Manufacturing Company (TSMC) using their 90nm Nexsys® process. A Printed Circuit Board (PCB) is also made, this in section 3.1.6 to allow for chip measurements in section 3.1.8.

The thesis also contains design and simulations of circuits implemented in ULV logic. This includes simulations of a ULV static carry generate circuit from [2], design and simulations of a new carry topology based on ULV5 logic and the design and simulation of a multiplexer and a new carry circuit based on ULV PTL logic [3].

2 Background

In this chapter, an introduction to the concepts needed in the design of ultra-low-voltage electronics will be given. The first concept explained is floating gate logic. The idea of a floating, or a semi-floating gate is the cornerstone of the ULV logic styles, and what makes this logic especially fast at low voltages. Because of these high-speed properties, the ULV logic styles are ideal for serial circuits with a high logic depth enabling more complex operations in the same clock period. One example of this circuit type is the adder, which is the next concept explained and it is a circuit where a moderately complex operation needs to be done on one bit after another in a serial manner making the total processing time directly proportional to the gate delay and the number of bits and the circuit.

The third section will focus on the effects of lowering the supply voltage below the transistor threshold voltage. These effects will be simulated to get an understanding of their effect on the devices from the production process used in this thesis.

To compare the new circuits to each other and to conventional electronics, some figures of merit are necessary. The figures of merit used will be explained in the last section of this chapter.

2.1 Floating Gate Logic

The idea of a floating gate is the cornerstone of the ULV logic styles and is the main reason for its high speed in low-voltage operation.

Floating gate logic is a logic style that uses a capacitor on the gate of an evaluation transistor. An illustration of this is shown in the figure below.

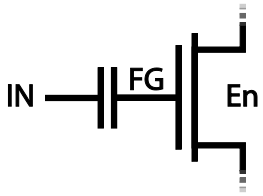

Figure 2.1.1: Schematic: Floating gate (FG) on an NMOS transistor.

The floating gate capacitor makes the input isolated from the DC voltage level on the gate of the transistor and allows the designer to set the operating point for the transistor while keeping most of the input voltage swing. The amplitude of this swing depends on the input capacitor size relative to parasitic capacitances like gate-source and gate-drain capacitance. Experiments with setting the floating gate operation point using UV light have been successfully executed in [4] (FGUVMOS) using a 0.8µm production process, but in newer processes with smaller transistors like the 90nm process used in this thesis, the gate leakage is too large to keep the charge on the floating gate for the entire circuit lifetime. To overcome this leakage problem and to avoid the postproduction UV light procedure, a transistor that charges the floating gate to a predefined value can be added as shown in Figure 2.1.2. This transistor is called a recharge transistor (***Rp***).

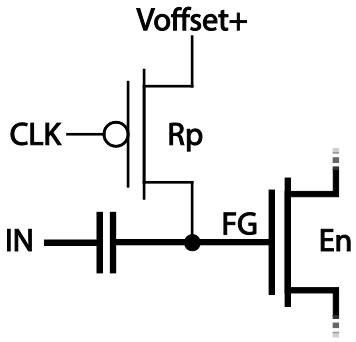

Figure 2.1.2: Schematic: Floating gate with recharge transistor (*Rp*).

This new ***Rp*** transistor introduces a clocked element to the circuit, making it synchronous and introducing a precharge phase in the circuit operation. This circuit will have a precharge phase when the ***CLK*** signal is '0' turning the precharge transistor ***Rp*** on and setting the floating gate to ***Voffset+*** as shown in Figure 2.1.3. When ***CLK*** switches to '1' the circuit enters the evaluation phase and the ***Rp*** transistor is off as shown in Figure 2.1.4 resulting in a floating gate on the ***En*** transistor with an operation point set by ***Voffset+***.

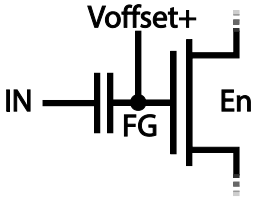

Figure 2.1.3: Floating gate in precharge mode.

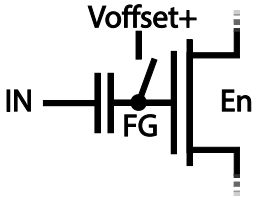

Figure 2.1.4: Floating gate in evaluation mode.

One of the simplest ways to utilize the floating gate concept is demonstrated in the Ultra-Low-voltage logic style (ULV) as shown in Figure 2.1.5 and presented in [5].

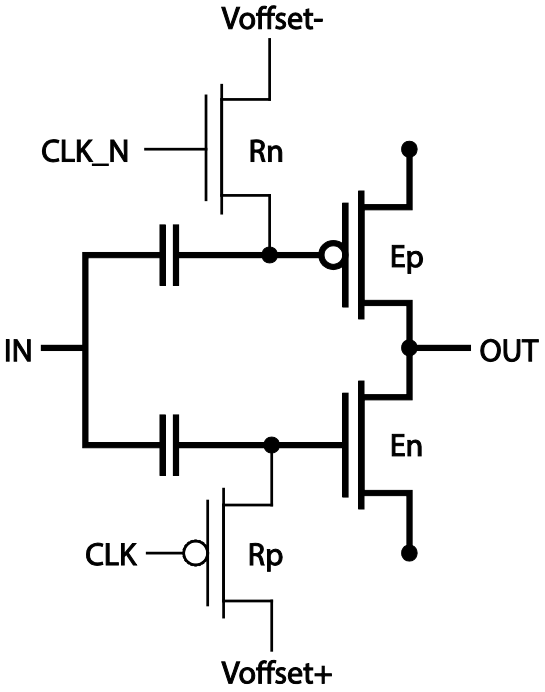

Figure 2.1.5: Schematic: Basic ULV floating gate inverter.

The ULV inverter exploits the capacitive coupling between the input and the floating gate to achieve a gate voltage that is higher than the circuit supply voltage resulting in a significantly higher current, especially at low voltages due to the drain-source currents exponential dependence on the gate-source voltage in ultra-low-voltage operation. A higher output current will charge the load and parasitic capacitances faster and result in a smaller propagation delay and a faster circuit.

2.2 Adders

An adder, in this context, is a circuit that adds two binary numbers. The basic building block of an adder is the 1-bit full-adder which is a circuit that adds two bits and a carry input to produce a sum and an output carry according to the truth table in Table 2.2.1.

A	B	C	Sum	C out
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Table 2.2.1: Full-adder truth table.

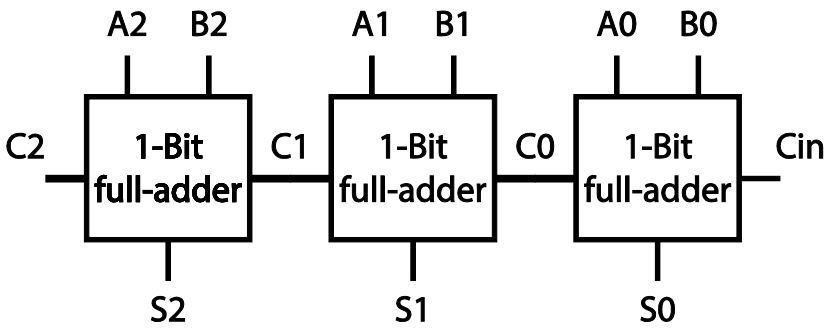

Figure 2.2.1: 3-Bit CPA.

From this building block we can create two types of adders, Carry Save adders (CSA) and Carry Propagate adders (CPA), which can be implemented in a number of ways using different optimization algorithms like in [6] and [7].

A CPA can be created by placing a number of full-adders in a series configuration with the carry input connected to carry output of the previous circuit as shown in Figure 2.2.1. In a CPA circuit the most important parameter is the carry propagation delay (t_P) because the carry signal needs to propagate through all the full-adders in the chain resulting in a total time of $t_P \cdot n$ for an n-bit adder to produce the correct result. Using an optimization algorithm like the Kogge-Stone [7] reduces the total time to $t_P \cdot log_2 n$ but it is still proportional to the propagation delay.

A CSA is essentially an array of full-adders in parallel; it receives three n-bit numbers and produces two n-bit arrays, one is the output carry and one is the sum. The CSA uses a total time of t_P independant of the number of bits due to its high level of parallelism.

When adding many operands a CSA is often used in a tree structure to reduce the number of operands. In a Wallace-tree configuration [8] for example, an n-operand addition is reduced to a two operand addition in $t_P \cdot log_2 n$ reduction layers with only a single gate delay per layer, reducing the total time to $t_P \cdot log_2 n$ from $t_P \cdot \log^2 n$ using only CPAs. In the final step of any reduction tree, a CPA will always be needed to add the final array of carry bits to the sum. The final CPA will use one gate delay per bit and because the number of bits is usually larger than the number of operands, this is the most important part to optimize, making the CPA speed critical.

An arithmetic logic unit (ALU) can consist of multipliers, dividers, square root extractors, adders and other arithmetic logic. In most of these, a CPA is the bottleneck of the algorithm.

Multipliers, dividers and square root extractors all employ algorithms that depend on reduction trees to sum their partial products. Even though the number partial products can be reduced by using high-radix algorithms, the bottleneck will still be the CPA of the reduction tree which cannot be trimmed away with algorithmic changes. This shows the importance of high-speed CPA carry propagation.

2.3 Sub-Threshold Effects

The ULV logic style operates at ultra-low voltages, which means that near-threshold and subthreshold transistor conditions apply. This makes the sub-threshold transistor effects necessary to consider in order to properly adjust the transistor strengths and to understand why the circuits are not behaving the same way as in higher voltage design.

In conventional CMOS design the transistors are considered to be "off" when the gate voltage (V_{GS}) is below the threshold voltage (V_{TH}). This is because the sub-threshold, or "off"-current current, is insignificant[1] in comparison to the "on"-current. When the supply voltage (Vdd) is lower than (V_{TH}), the subthreshold current dominates, resulting in equation 2.3.1 from [9].

$$I_{DS} = I_0 \frac{W}{L} e^{(V_{GS}-V_{TH})/n \cdot v_t} \left(1 - e^{-V_{DS}/v_t}\right) \quad (2.3.1)$$

Where I_0 is the technology-dependent sub-threshold current, v_t is the temperature dependent thermal voltage and n is the process dependent sub-threshold factor. The V_{TH} parameter is the transistor threshold voltage which is affected by both the drain-source voltage (V_{DS}) through drain-induced barrier lowering (DIBL) [9] and the bulk-source voltage (V_{BS}) through the body-effect as shown in equation 2.3.2 [10].

$$V_{TH} = V_{TH0} - \lambda_{DS} V_{DS} - \lambda_{BS} V_{BS} \quad (2.3.2)$$

Where λ_{DS} is the DIBL coefficient and λ_{BS} is the body-effect coefficient.

Drain induced barrier lowering (DIBL) is used to describe the effects of the drain voltage on the threshold voltage and transistor current [11]. The body effect is used to describe effect of the bulk biasing voltage on the transistor threshold voltage.

From equation 2.3.1 we can see that the transistor current is exponentially proportional to both the gate-source voltage and the threshold voltage, making the threshold voltage an effective parameter for tuning the transistor strength when V_{GS} is limited.

The effects of these threshold voltage tuning techniques in the 90nm process used in this thesis are explored in this section, first the reverse narrow- and short-channel effects, then the body biasing effects and a summary comparing and combining these techniques.

[1] A difference is in the order of $I_{on}/I_{off} > 10^4$

2.3.1 Reverse narrow- and short-channel effects

To explore the reverse narrow channel effect and the reverse short channel effect in the TSMC 90nm Nexsys® process, the current through an NMOS and a PMOS transistor is simulated for a range of transistor sizes and plotted as a function of minimum length and minimum width in Figure 2.3.1 for ULV operation, and in Figure 2.3.2 for traditional voltage level operation.

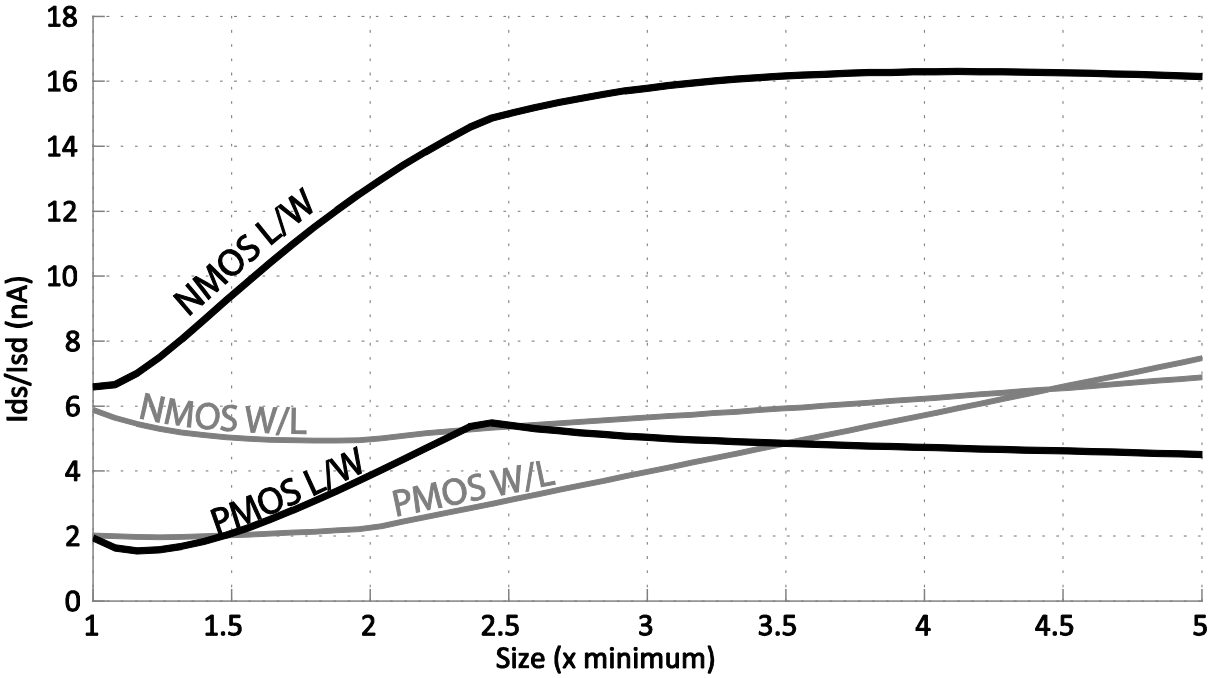

Figure 2.3.1: Simulation: Drive strength of near minimum size transistors at 200mV.

When transistors operate in the sub-threshold region the usual design rules based on above threshold operation shown in Figure 2.3.2 get less relevant. As shown in Figure 2.3.1 the transistor strength for small transistors will increase with the device length until it is about 4 times minimum for the NMOS transistor and 2.5 times for the PMOS transistor due to the reverse short channel effect [9].This indicates that increasing the transistor length is a good way to increase the transistor drive strength. Increasing the transistor length can therefore be used to strengthen the pull-up- or pull-down-network of a circuit. The PMOS strength at 2.5x minimum length is close to matching the strength of a minimum sized NMOS transistor and shows that increasing the length of the PMOS can be used for e.g. balancing the rise- and fall-time of an inverter.

The transistor current is not necessarily translatable to circuit speed if the load capacitance of the circuit is also affected by the applied changes. Because the gate capacitance of the transistor is directly proportional to the length, strengthening the transistor by increasing the length leads to a parallel rise in gate capacitance. Increasing the speed of a system by increasing all transistor lengths is therefore only a viable solution if the gradient of the current as a function of length is larger than 1, meaning that the transistor strength increases more than the load capacitance. Figure 2.3.3 shows that the gradient of the simulated transistors is approximately 1. This means that increasing the transistor length is only beneficial for circuits where not all transistors need to be strengthened to reduce the worst case delay and for circuits where the affected transistor gate parasitics are not the dominating the input and/or output capacitances.

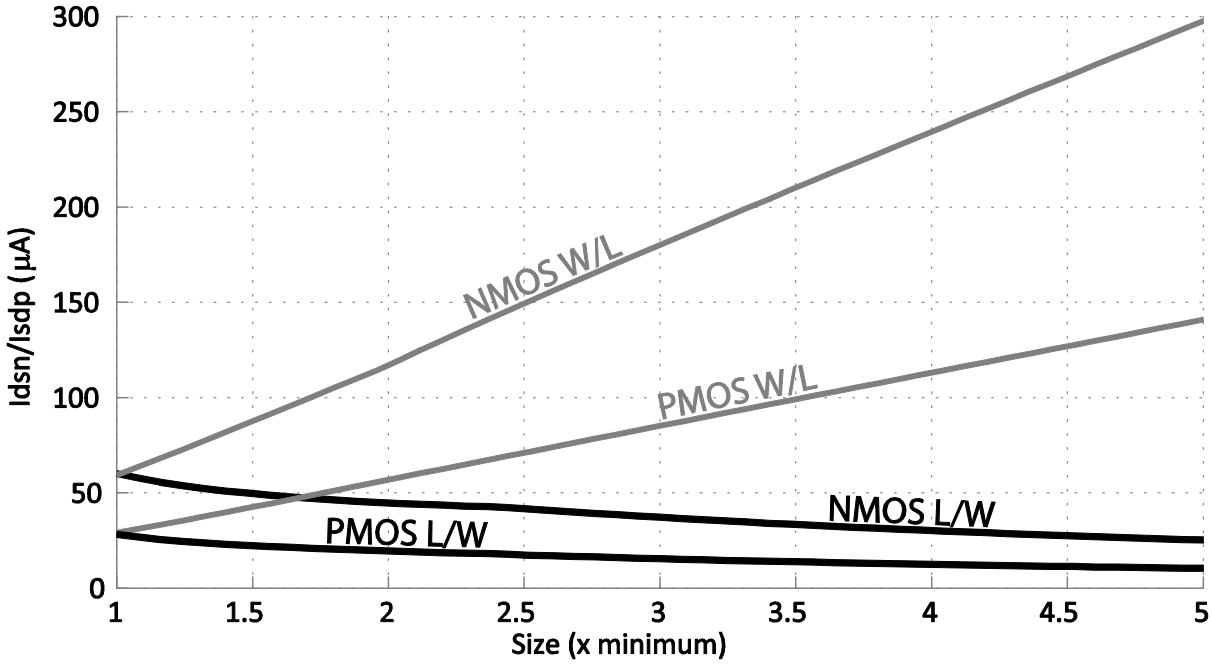

Figure 2.3.2:Simulation: Drive strength of near minimum size transistors at 1.2V.

Changing the transistor width proves less useful than in above-threshold design and even reduces the transistor strength for small $\frac{W}{L}$ –values. This is caused by the reverse narrow channel effect [9] and shows that increasing the transistor width can be counter productive and is an ineffective way to increase drive strength compared to increasing the length for small transistors.

2.3.2 Body effect

The body effect is a reduction of the effective threshold voltage that occurs when the body of the transistor is not biased with the same voltage level as the source terminal. In above-threshold design a reverse bulk bias (RBB) is always applied, mainly to avoid latchup generating a bipolar junction transistor and frying the circuit, but also because of leakage and area penalties.

When the supply voltage is lower than the diode drop[1] a forward body bias can be applied without causing latchup. The effect of the body biasing voltage (V_{BS}) on the threshold voltage (V_{TH}) can be seen in equation 2.3.2 on page 6 and the effect on the resulting transistor strength is shown below.

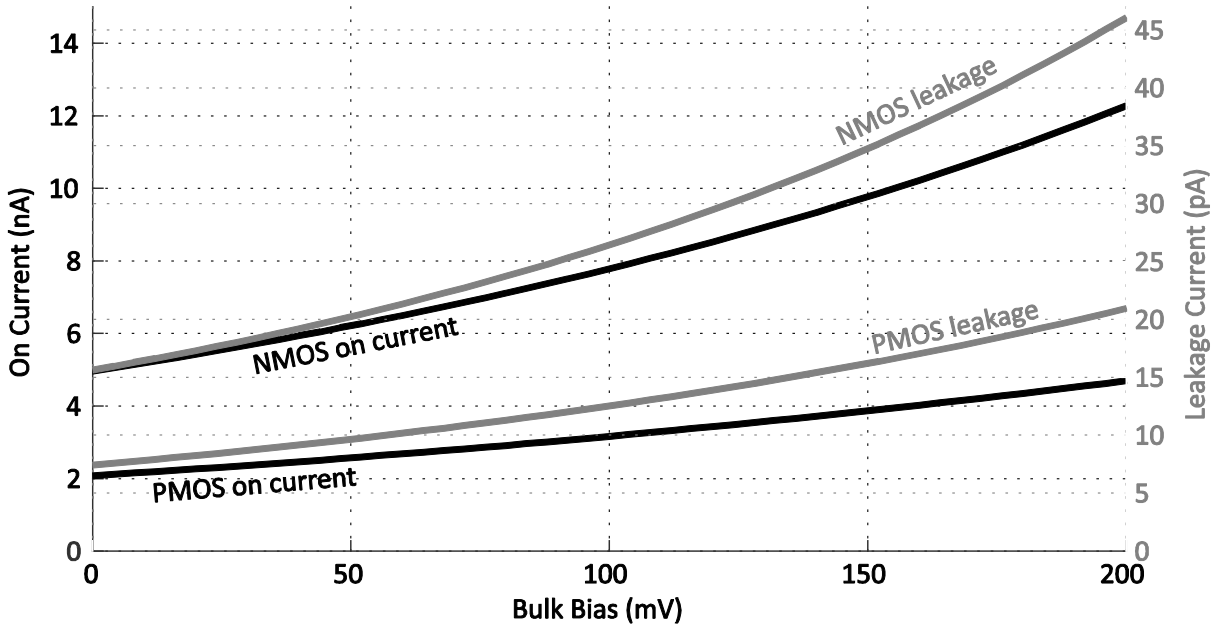

Figure 2.3.3: Simulation: Transistor strength for varying body bias (Vdd = 200mV).

Figure 2.3.3 shows the increased current caused by a higher body bias voltage for the nch_mac and pch_mac transistors in the TSMCN90RF process library at 200mV. Both the NMOS and the PMOS is 2.4 times stronger with a forward body bias (FBB) than with a reverse body bias applied.

The simulation results also show that the relative increase of leakage current is larger than the relative increase of the on-current when forward biasing the circuit. The leakage current through the NMOS transistor is 3 times higher with FBB applied than with RBB applied, meaning that the relative increase of the leakage current is 23% higher than the relative increase of on-current. For the PMOS transistor the relative leakage increase is in the same range with 20%. This results in a 20% lower I_{on}/I_{off} ratio when forward biasing is applied compared to reverse biasing.

A floating transistor bulk is also an option can be used by not connecting the bulk to any set voltage potential. According to simulations this is equivalent to a bias of Vdd/2 under DC

[1] Between 0.6V and 0.7V for a silicon semiconductor p-n junction

conditions but can increase the on-current by up to 5% for transient signals with a full transition on the input due to parasitic capacitive coupling between the gate and bulk.

To apply a bias voltage to the body of an NMOS transistor which is effectively the substrate of the chip, it needs to be placed in a deep n-well. This is done to isolate the body of the transistor from the rest of the ground connected substrate. An illustration of this isolation is given in Figure 2.3.4.

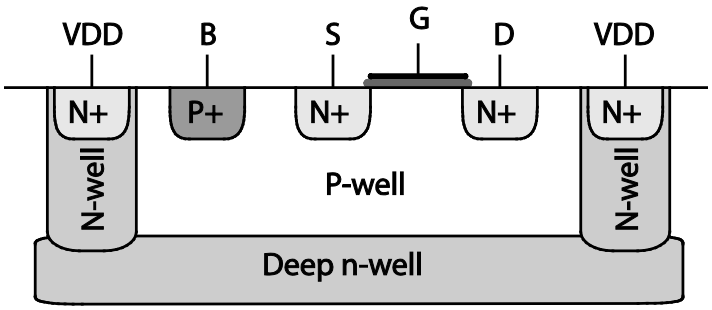

Figure 2.3.4: Deep n-well illustration.

As we can see from this figure, the deep n-well adds an area overhead to the transistor. This is further increased by strict spacing rules needed for the manufacturing of the deep n-well, resulting in a high area penalty.

The PMOS transistor already has an isolated n-well and therefore does not need any extra modification for bulk biasing. This makes FBB an effective way to strengthen PMOS devices without the same increase in area.

Advantages	Disadvantages
• Increases transistor strength.	• Deep N-well area penalty for NMOS. • Increased leakage / reduced I_{on}/I_{off} ratio

2.3.3 Subthreshold scaling summary

In the previous sections several ways to increase the transistor strength in subthreshold operation are presented and simulated. These show that even though many of the conventional scaling techniques are ineffective and some even counterproductive at low voltages there are still ways to increase the transistor strength, even at subthreshold and near-threshold supply voltage levels.

In Table 2.3.1 a list of the presented subthreshold scaling techniques and their effects on the transistor "on"-current and the I_{on}/I_{off} ratio is presented. This table shows that the transistor strength can be significantly increased by combining the three scaling techniques described. It also shows that they have a larger effect on low threshold devices. The I_{on}/I_{off} ratio benefits most from increasing the length and using low-threshold transistors. By doubling the length alone this ratio can be increased by 36% while doubling the on-current, but together they can achieve an I_{on}/I_{off} ratio increase of 80% with 3 times the current of the minimum sized standard V_{TH} device. All three techniques can also be used at the same time to make the transistor as strong as possible. This is shown in Table 2.3.1 where a low-threshold transistor with a length of twice the minimum and full forward biasing yields an on-current 6 times higher than the standard-threshold, minimum sized and reverse biased device.

To sum up these findings the best way to increase the transistor strength is to use low-threshold transistors and further tune the strength by adjusting the channel length and keep the transistors reverse biased, at least for NMOS transistors. The advantages of using low-threshold transistors and increasing the length are both obvious in the table below and in the previous discussion, but when it comes to bulk biasing there are two reasons for this conclusion. One is the reduced ratio between the "on"-current (I_{on}) and leakage current (I_{off}) because static power consumption increases more than the dynamic, making it a less energy efficient circuit. The other is the area overhead this introduces for NMOS devices by requiring a deep n-well for bulk isolation.

Table 2.3.1 shows the effects of subthreshold scaling techniques on the transistor current that can be used to reduce the effects of process variations and increase circuit speed. As explained in section 2.3.1 the gate capacitance of the transistors with a 2×min length is twice that of the minimum length transistors which may have an effect on the delay reduction achieved by the increased current, depending on the application. For circuits where the gates are connected in series with an output capacitance dominated by the parasitic capacitance of gates with the same scaling, forward biased low threshold transistors will achieve the highest speed, but with the penalties to area and energy efficiency previously mentioned

Type	**Bias**	**Length**	I_{off}	I_{on}/I_{off}		I_{on}		
nch_mac	Rev	1 x min.	26.22pA	251.3x	-	6.59nA	-	-
nch_mac	Rev	2 x min.	37.43pA	340.6x	+36 %	12.75nA	1.9x	+93 %
nch_mac	Fwd	1 x min.	65.62pA	217.0x	-14 %	14.24nA	2.2x	+116 %
nch_mac	Fwd	2 x min.	99.77pA	270.3x	+8 %	26.97nA	4.1x	+309 %
nch_lvt	Rev	1 x min.	32.50pA	340.6x	+36 %	11.07nA	1.7x	+68 %
nch_lvt	Rev	2 x min.	41.49pA	**451.9x**	**+80 %**	18.75nA	2.8x	+185 %
nch_lvt	Fwd	1 x min.	84.83pA	271.5x	+8 %	23.03nA	3.5x	+249 %
nch_lvt	Fwd	2 x min.	115.20pA	340.1x	+35 %	39.18nA	**5.9x**	**+495 %**

Table 2.3.1: Simulation: Effect of sub-threshold scaling on transistor drive strength at 200mV.

2.4 Figures of merit

To analyze the performance of the circuits created in this thesis it is beneficial to have some figures of merit to be able to compare them to other logic styles and circuit topologies. The circuits created will be compared to standard CMOS circuits in order to give an idea about the value and quality of the circuits presented.

The conventional CMOS carry circuit shown in Figure 2.4.1 will serve as a benchmark for the carry circuits that will be proposed later in the thesis.

The figures of merit presented are; propagation delay, power, PDP and EDP.

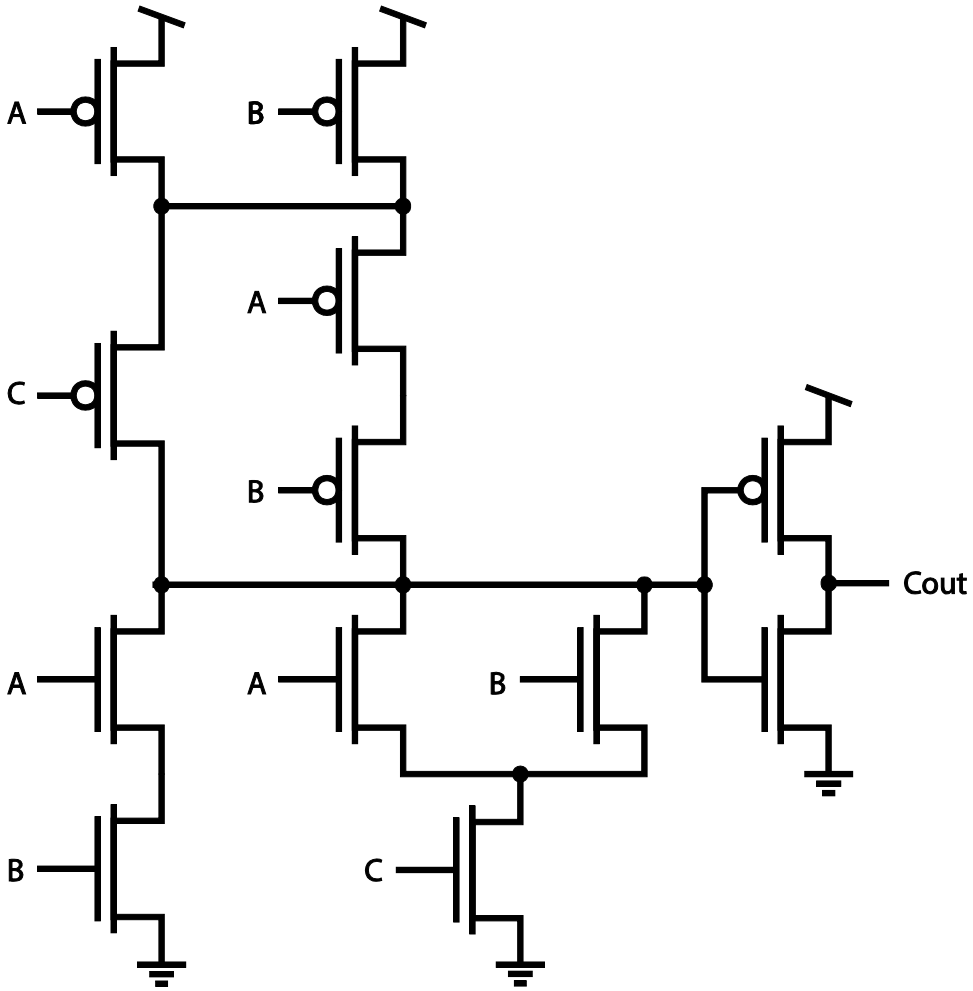

Figure 2.4.1: Standard CMOS carry circuit.

2.4.1 Propagation Delay

The propagation delay of a circuit is an important factor that determines the speed and maximum operating frequency of an electronic system.

The carry propagation and delay of the conventional CMOS carry circuit can be seen in Figure 2.4.2 and shows that the carry propagation of the circuit is uniform and has a good noise margin. The delay of the 32-bit carry chain with a 1.2V supply voltage is 1.57ns, on average a 48ps propagation delay for each carry bit.

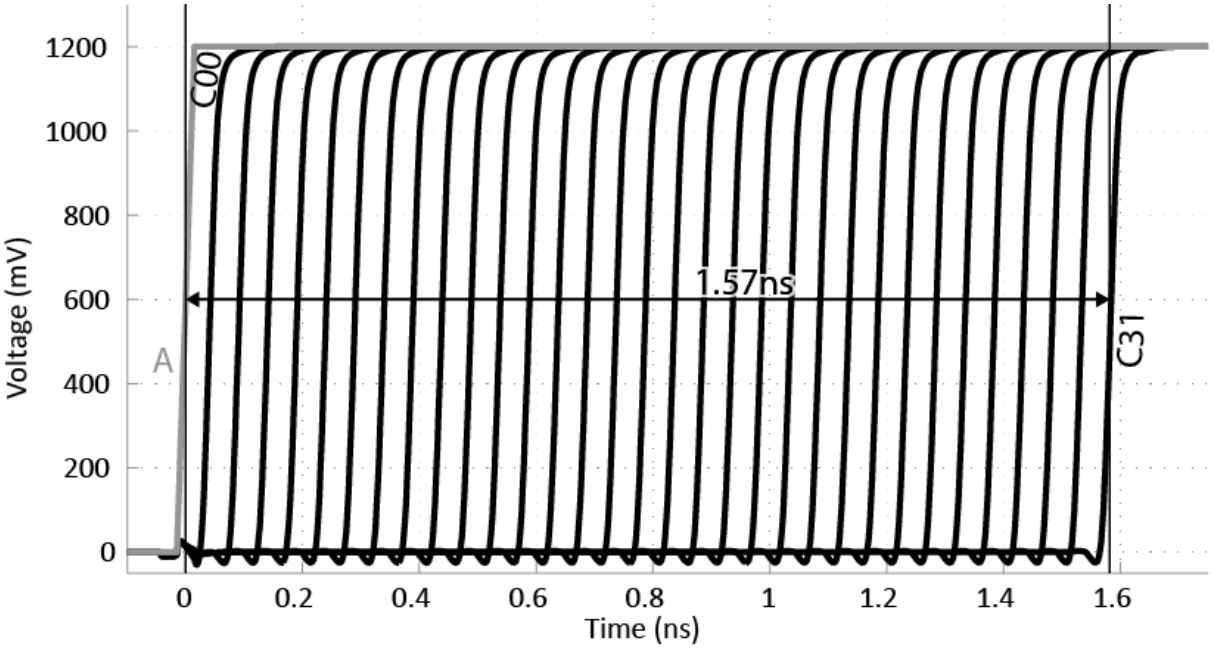

Figure 2.4.2: Simulation: Carry propagation through a standard CMOS carry chain at 1.2V.

Because the propagation delay is exponentially dependent on the supply voltage (Vdd), the increase in delay is significant when Vdd is lowered, especially near the threshold voltage. The relationship between the supply voltage and the propagation delay is shown in Figure 2.4.3 where the delay per bit can be seen decreasing by orders of magnitude when the supply voltage is lowered to near-threshold values.

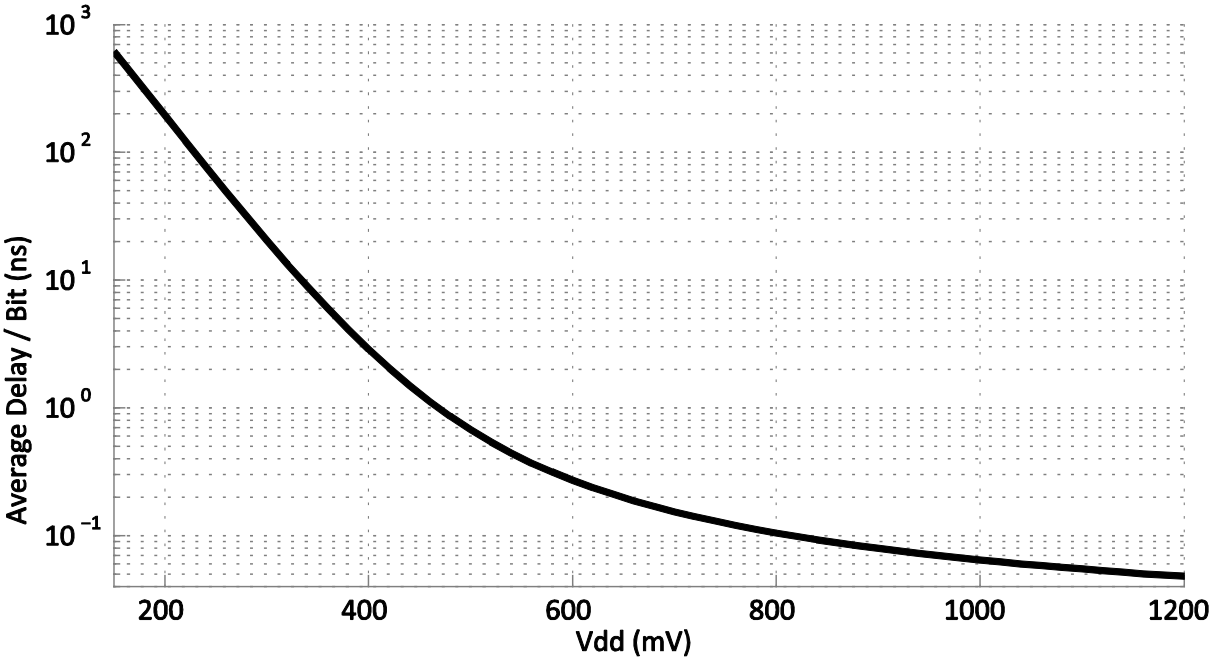

Figure 2.4.3: Delay per bit of a standard CMOS inverter.

	Best	@ 200mV	@ 1.2V
Delay	max.	194.9ns	48.01ps

Table 2.4.1: Conventional CMOS Carry Propagation Delay.

2.4.2 Power

The power consumption is an important factor to consider when designing a circuit, especially in applications with a limited supply of energy like in battery operated devices or a limited maximum power consumption like in devices relying on energy harvesting but can also be a factor in terms of heat development.

The power consumption of a circuit is strongly dependent on the supply voltage. Reduced power consumption is therefore one of the main reasons for lowering Vdd. The relationship between the supply voltage and the power consumption is shown for the conventional CMOS carry circuit in Figure 2.4.4 where the power consumed at 1.2V is almost five orders of magnitude[1] higher than that consumed at 0.2V. The power consumption at these voltages is shown in Table 2.4.1.

	Best	@ 200mV	@ 1.2V
Power	min.	862.8pW	71.27µW

Table 2.4.2: Conventional CMOS Carry Power Consumption.

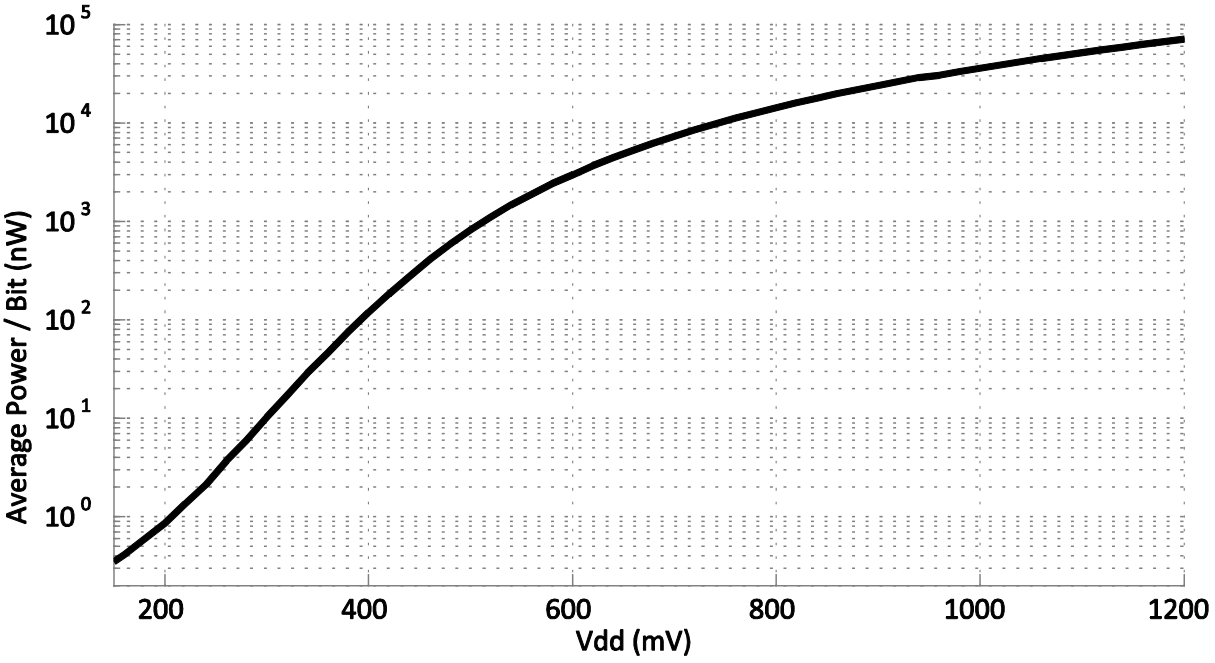

Figure 2.4.4: Simulation: Power consumption of a standard CMOS carry circuit.

[1] 0.826×10^5

2.4.3 Power-Delay Product

The power-delay product (PDP) is simply the average power consumption times the delay of a circuit and represents the average energy consumed per switching event, it is therefore also known as the switching energy. The PDP says something about how much energy is consumed to perform an operation and is therefore a good figure of merit to determine circuit performance in terms of energy efficiency. In Figure 2.4.5 it is shown that circuit 1 and circuit 2 will spend the same amount of energy to perform the same calculation because circuit 2 uses half as much power as circuit 1 but spends twice the time on it, resulting in the same PDP and showing that the two circuits are equally energy efficient.

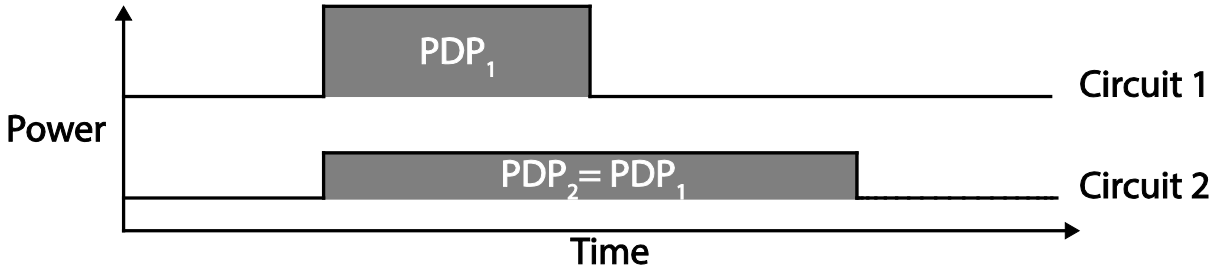

Figure 2.4.5: Illustration of PDP.

For this reason PDP can be used to compare energy efficiency for different circuit topologies with different power and speed characteristics. The PDP varies with the voltage supply and this relationship is shown in Figure 2.4.6. The optimal supply-voltage for the conventional CMOS carry circuit according to the PDP analysis is at 240mV, just below the transistor threshold voltage[1], with a PDP of 0.166fJ as presented in Table 2.4.2.

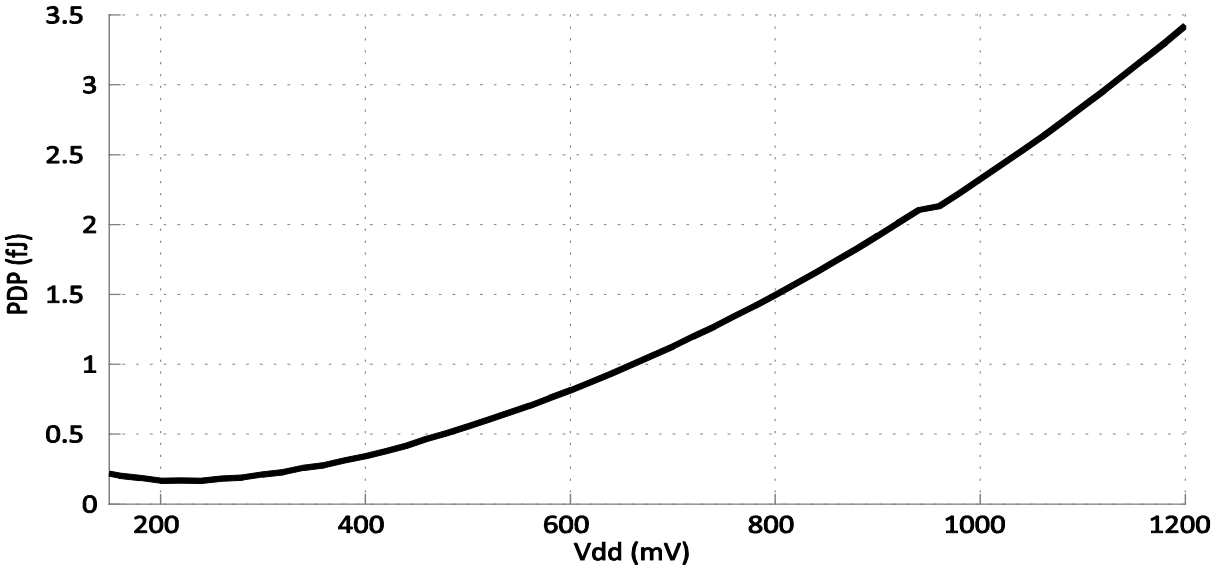

Figure 2.4.6: Simulation: PDP of a standard CMOS carry circuit.

	Best	@ 200mV	@ 1.2V
PDP	0.1659fJ	0.1681fJ	3.411fJ

Table 2.4.3: Conventional CMOS Carry PDP.

[1] V_{TH} = 267mV according to the TSMCN90RF spice models.

2.4.4 Energy-Delay Product

The energy-delay product is a metric representing the achieved speed of the circuit relative to the energy consumed and is calculated by multiplying the switching energy (PDP) the propagation delay.

In terms of PDP, the two circuits in Figure 2.4.5 perform equally well, but in reality circuit 1 would be the obvious choice for most applications because a faster circuit leaves more flexibility in terms of data throughput, system response time and algorithm complexity.

In contrast to the PDP of the Figure 2.4.5 circuits, the EDP of Circuit 1 is twice that of circuit 2 because it finishes the operation twice as fast using the same amount of energy. The more speed-oriented but still energy aware nature of the EDP is the reason it is a widely accepted metric for comparing performance of digital circuits. For the conventional CMOS carry circuit, the EDP is highly dependent of the supply voltage, and as shown in Figure 2.4.7 the PDP increases by four orders of magnitude by only reducing the Vdd by 80%.

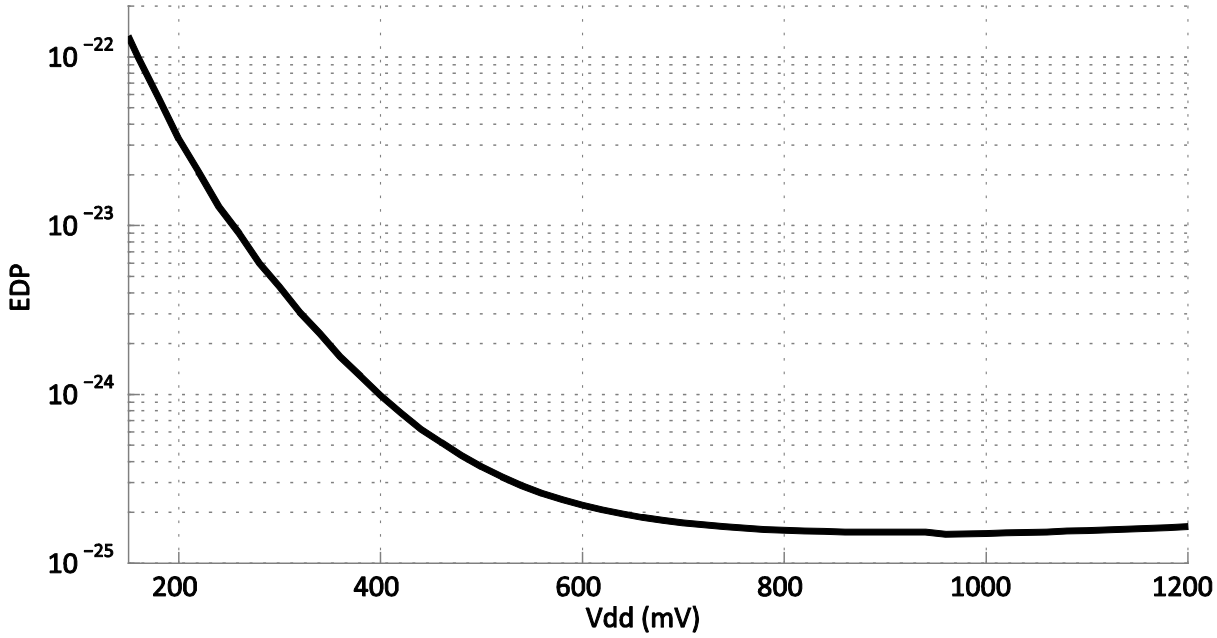

Figure 2.4.7: Simulation: EDP of a standard CMOS carry circuit.

The optimal supply voltage for the standard CMOS carry circuit according to the simulation results in Figure 2.4.7 is 0.96V where it achieves an EDP of just 0.1486yJs.

	Best	@ 200mV	@ 1.2V
EDP	0.1486yJs	32.77yJs	0.643yJs

Table 2.4.4: Conventional CMOS Carry EDP.

2.4.5 Standard deviation (σ)

Standard deviation, commonly represented by σ, is a figure representing variation, which in IC design is caused by process variations.

The standard deviation represents how much the result will deviate from the mean value with a certain probability, and can in turn be used in IC design to show the probability of the circuit performing within the required boundaries. This probability is referred to as yield and shows the number of circuits that need to be thrown away in an average production run. As shown in Figure 2.4.8, one σ represents the deviation from the mean (μ) where 68.2% of the results will be, and for two σ, 2×13.6% can be added giving a yield of 95.4% and so forth. Put in another way, if one σ can be tolerated, 68.2% of the circuits will fulfill the requirements.

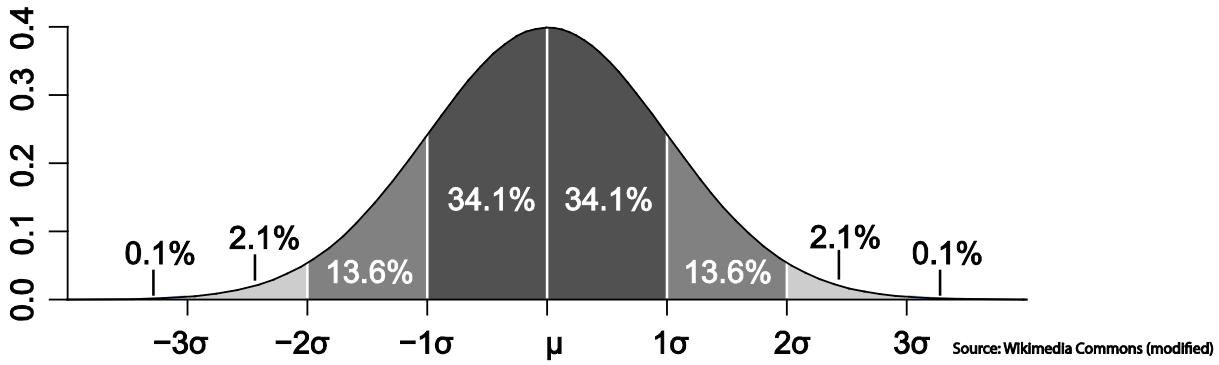

Figure 2.4.8: Standard deviation diagram.

This is why a low standard deviation is desirable and will result in a higher production yield by being able to accept a higher number of standard deviations and/or increase the precision and performance of the circuit.

For discrete random variables from x_i to x_N, which is most relevant for the results in this theses the formula for the standard deviation is:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}\,, where\ \mu = \frac{1}{N}\sum_{i=1}^{N} x_i = mean(x)$$

In this thesis standard deviation will not be a significant focus, but will be used to analyze chip measurements.

3 Design

This chapter will cover the design, production and measurement of ULV circuits. The tools used for design and simulation of both the schematic and the layout of the circuits are from the Cadence® Custom IC Design kit where Virtuoso® and Spectre® are used together with the TSMCN90RF device model library for the 90nm TSMC® Nexsys® process.

3.1 Implementation of an Ultra-Low-Voltage Inverter

3.1.1 Intro

A lot of simulations have been performed on ULV logic in papers, master theses and PhD dissertations, but few implementations and little work has been done in the manufacturing these circuits. So to show that the features of ULV logic also apply in hardware, an implementation of ULV logic in a 90 nanometer process will be manufactured in this thesis.

The logic style chosen for implementation is the 7th iteration of the ULV logic (ULV7) because it is the latest, most robust and one of the fastest ULV topologies published this far. The chosen circuit is the ULV7 N-type inverter, scaled to handle the load of a measurement setup. This is because a minimum scaled circuit would need buffers to drive the outputs that would mask the analog properties of the circuit with those of the buffer used. A traditional CMOS inverter with equivalent scaling will also be implemented on the same chip as a reference and a benchmark for the circuit performance.

A paper on the chip production is also written [12] and accepted for publishing at the IEEE International Nanoelectronics Conference (IEEE INEC2014). See appendix, section 5.1.

3.1.2 7th Generation Ultra-Low Voltage Logic

The ULV7 logic style chosen for this chip has been called robust low-power CMOS precharge logic in previous papers like [13] and is based on the ULV5 [14] logic style. Like other domino logic it has a precharge phase and an evaluation phase. In the precharge phase the output is charged to ‘1’ for the N-type circuit and to ‘0’ for the P-type circuit, but when the evaluation phase arrives, the circuits will either switch or not switch depending on the input(s) and logic. For an N-type circuit with a precharge value of ‘1’ the output logic is decided by a pull-down network (PDN). The complimentary P-type circuit logic is decided by a pull-up network (PUN).

The two phases of operation for these circuits are defined as the precharge phase when ***CLK*** = ‘0’ and ***CLK_N*** = ‘1’, and as the evaluation phase when ***CLK*** = ‘1’ and ***CLK_N*** = ‘0’. The following is an elaboration on the roles of the transistors and signals for the N-type inverter in both of these phases:

Precharge phase: When the ULV7 N-type inverter shown in Figure 3.1.1 enters the precharge phase, the floating gate of the evaluation transistor (***En***) is charged to ***Voffset+*** through ***REp,*** and the gate of the precharge transistor (***Pp***) is charged to ***Voffset-*** through ***RPn***. ***Voffset+*** and ***Voffset-*** are usually set to Vdd and ground respectively but can also have other values to reduce or increase the speed and power consumption. This means that both ***En*** and ***Pp*** are semi-conducting and slowly charging the output to '1' during the precharge phase.

Evaluation phase: During the evaluation phase the ***RPn*** and ***REp*** transistors are turned off, allowing the ***Pp*** and ***En*** gates to float so that an arriving rising edge will cause the ***En*** floating gate to be super-charged and drive the output to '0' from ***CLK_N***. The low output will then turn ***KPp*** on and pull the ***Pp*** gate to '1' effectively securing the output to '0'. On the other hand if the input stays at '0' the weak ***KEn*** transistor will be on and slowly lower the ***En*** floating gate voltage. This means that the ***KEn*** strength needs to be designed to make sure that the precharge value is held and that the circuit accepts inputs through the entire evaluation phase.

The P-type circuit works based on the same principle but in a complimentary manner to the N-type, with the floating gate on the ***Ep*** PMOS transistor as shown in Figure 3.1.2, making the NMOS the precharge transistor (***Pn***).

ULV7 Inverter signal phases			N-type		P-type	
Signal	*CLK*	*CLK_N*	*IN*	*OUT*	*IN*	*OUT*
Precharge phase	0	1	0	1	1	0
Evaluation phase	1	0	0/↑	1/↓	1/↓	0/↑

Table 3.1.1: Signal phases of the ULV7 inverter.

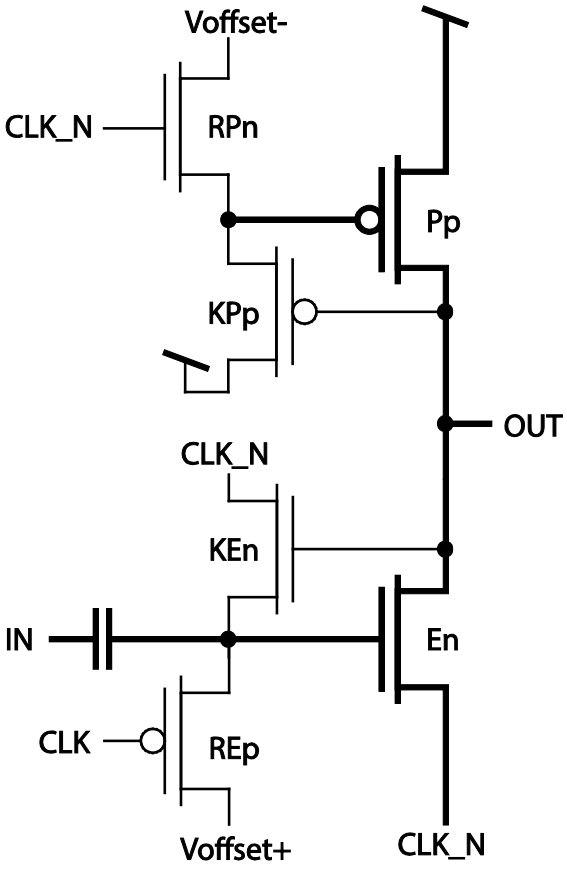

Figure 3.1.1: Schematic: ULV7 N-type inverter.

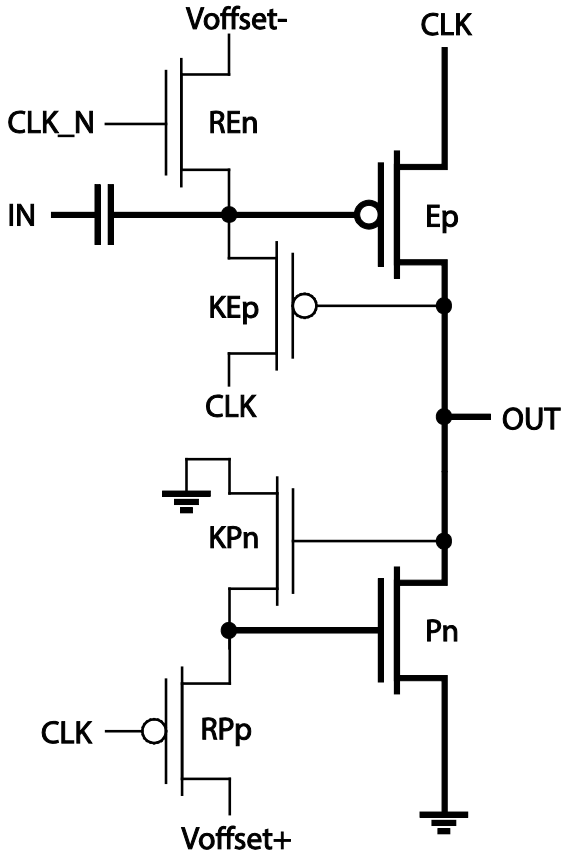

Figure 3.1.2: Schematic: ULV7 P-type inverter.

3.1.3 Design Considerations

When designing an electronic circuit in a nanoscale process it is necessary to be aware that a single minimum size transistor is not strong enough to drive the capacitive load of an output pad. For this reason, a strategy to allow measurements of the circuit needs to be chosen. The two viable solutions available are to either scale the circuit to be able to handle the capacitive output load or add a buffer for each output. For larger digital systems where only the digital properties are tested, buffers are the best choice because they allow for a smaller circuit area for the implemented logic and can still deliver the correct output. The downside to using buffers is that they can only verify or refute that the logical value of the output is correct. Any analog properties like speed and noise margin measured on these outputs will be those of the buffer and not the implemented logic, so for this test circuit the scaling solution is chosen because, even though it is a digital circuit, it is the analog properties that are interesting at this research stage.

To scale the circuit properly, the output capacitance needs to be estimated before a test bench is made and simulations are run. The output capacitances that are considered will consist of probe pads, bonding wire, chip carrier leads, socket, PCB and oscilloscope measuring probes. The total estimated output capacitance will therefore be based on the following assumptions:

Pad frame + bonding + CC leads	~1pF
Socket + PCB	3-4pF
Measuring probes	15pF
Sum	**20pF**

These assumptions are used to create a test bench for each of the inverters with an output load capacitance of 20pF.

In addition to being scaled to drive the output load, the transistors will be scaled to give a reasonably fair comparison between the two inverters that resembles minimum size conditions.

Nominal pre-layout schematic simulations indicate that the ULV7 inverter is approximately 65 times[1] faster than the conventional CMOS inverter. Further elaborations on simulation results are presented in section 3.1.4.

[1] 300mV: 1.01us/15.5ns = 66x
200mV: 9.1us/154.7ns= 63x

3.1.3.1 High-Speed Layout

The goal of this version of the layout is to achieve high speed per area for both inverters and is therefore designed using the ultra-low voltage strength enhancement techniques explained in section 2.3.

To increase the transition speed, the current through critical transistors needs to be increased in order to charge the load and parasitic capacitances faster. Achieving a higher current through the device can be done by lowering the effective threshold voltage of the transistor [10]. For this purpose the low-threshold transistors, nch_lvt from the TSMC® Nexsys® process were used, this alone reduces the threshold voltage by 7%[1] and increases the drive strength by 68% compared to the standard nch_mac transistors from the same process. The effective threshold voltage was further reduced by applying a forward bias to the bulk of the transistor. Together with the lvt device this resulted in a 249% drive strength increase. Forward biasing the PMOS devices is done by connecting the n-well to ground; the NMOS devices on the other hand require the transistors to be enclosed by a deep n-well for insulation to allow increasing the substrate voltage around the transistor while keeping the rest of the substrate grounded. The deep N-well can be seen around the ***En***, ***RPn*** and the ***En (cmos)*** transistors in Figure 3.1.3

The last low-voltage trick applied in this version to increase the drive strength for critical transistors was increasing the transistor length to exploit the reverse short channel effect, lowering the effective threshold voltage even further and increasing the drain-source current to achieve the correct strength ratio. The increased finger length is visible on the ***Ep*** and ***Ep (cmos)*** transistors in Figure 3.1.3.

To save area, a metal-oxide-semiconductor capacitor (MOSCAP) was chosen as the input capacitor for the ULV inverter because of its high capacitance density despite its poor accuracy. This is done because the capacitor is used as a blocking capacitor and its value is not a critical parameter for this circuit.

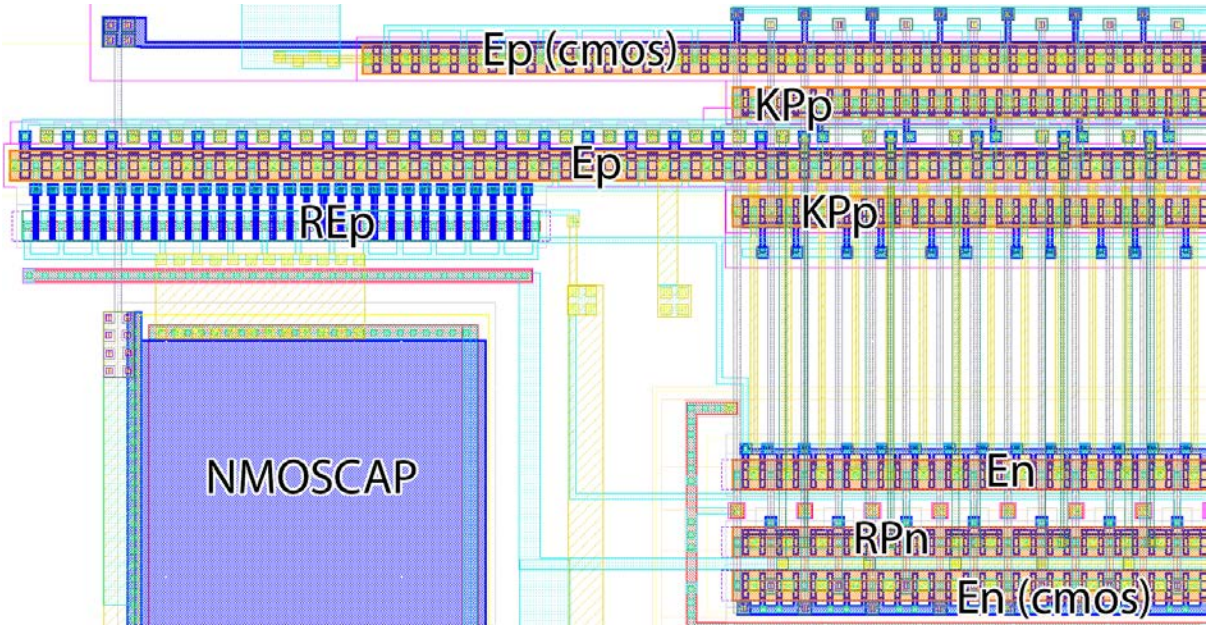

Figure 3.1.3: High-Speed Layout.

[1] Spice model Vth of nch_lvt:0.2489V, standard nch_mac:0.2668V. 1-(0.2489/0.2668)=6.7%.

3.1.3.2 High-Yield Layout

This version of the layout targets a high production yield to increase the probability of achieving measurable results and still keep a fair and accurate comparison between the two inverters.

The high-speed layout version was created using subthreshold strength enhancement design techniques to make both the ULV7 inverter and the traditional inverter as fast as possible. In this version, the transistor strengths are only adjusted with the number of transistor fingers. This causes a larger layout area but also supports a higher production yield and allows for a wider range of supply voltages. The increased area is not an issue for this design as long as it does not violate area the constraints for the project which provides plenty of space. All transistors have also been reverse-biased to avoid the need for a deep n-well and to reduce the static leakage of the circuits.

For the coupling capacitor on the evaluation transistor floating gate in this version of the layout, a rotative metal-oxide-metal (RTMOM) capacitor was chosen instead of a MOSCAP because it can be placed in the metal layers above the transistors and thereby reduce the effective area consumed by the capacitor, they also provide a higher accuracy for the capacitive value. As mentioned before the accuracy of the capacitor is not critical for the correct operation of the circuit but large variations can affect the speed. The capacitor is placed next to the transistors to make the layout more lucid. The important parts of the layout are shown in Figure 3.1.4.

Although a smaller effective substrate area is consumed by the RTMOMCAP due to its placement in the metal layers, its physical size is larger than that of a MOSCAP of the same capacitance, and in addition to a larger capacitance needed due to the larger number of transistor fingers, the overall capacitor area in this layout version is several times larger than in the high-speed layout version.

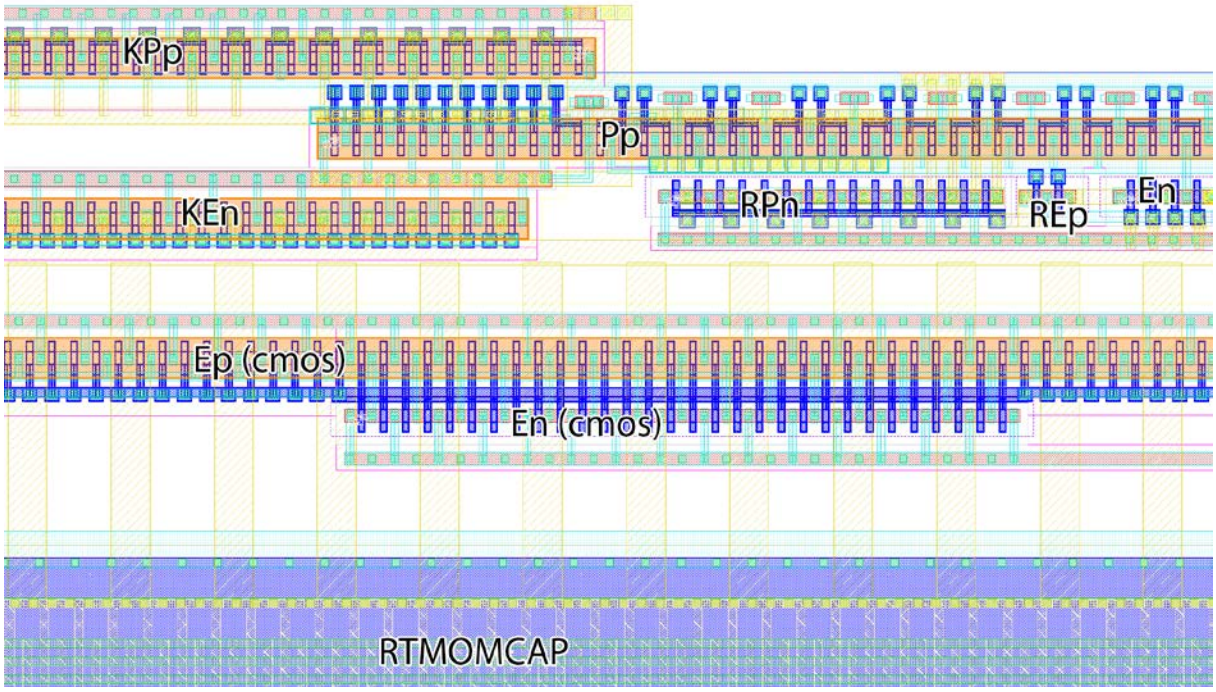

Figure 3.1.4: High-Yield Layout.

3.1.3.3 Final Layout

Because the goal of creating the chip is to compare the ULV7 logic style to the conventional CMOS logic style and not to achieve the highest possible speed reachable in the given process, the final layout is based on the high-yield layout. The biggest changes to this version is that the inverters are moved apart and separated with substrate ground connections seen as the blue, red and purple grid in Figure 3.1.10 or the pink area in Figure 3.1.9. This has been done to isolate the circuits from each other and to reduce crosstalk and leakage between the two.

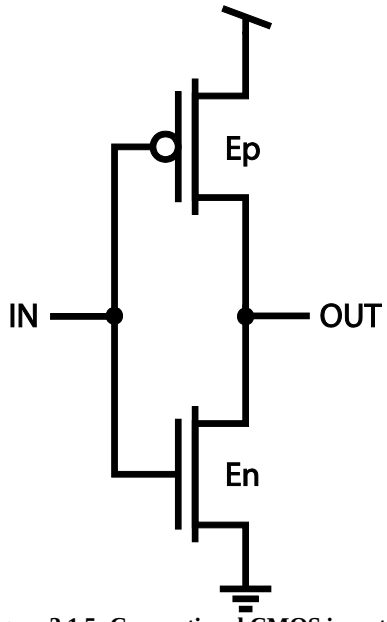

Figure 3.1.5: Conventional CMOS inverter.

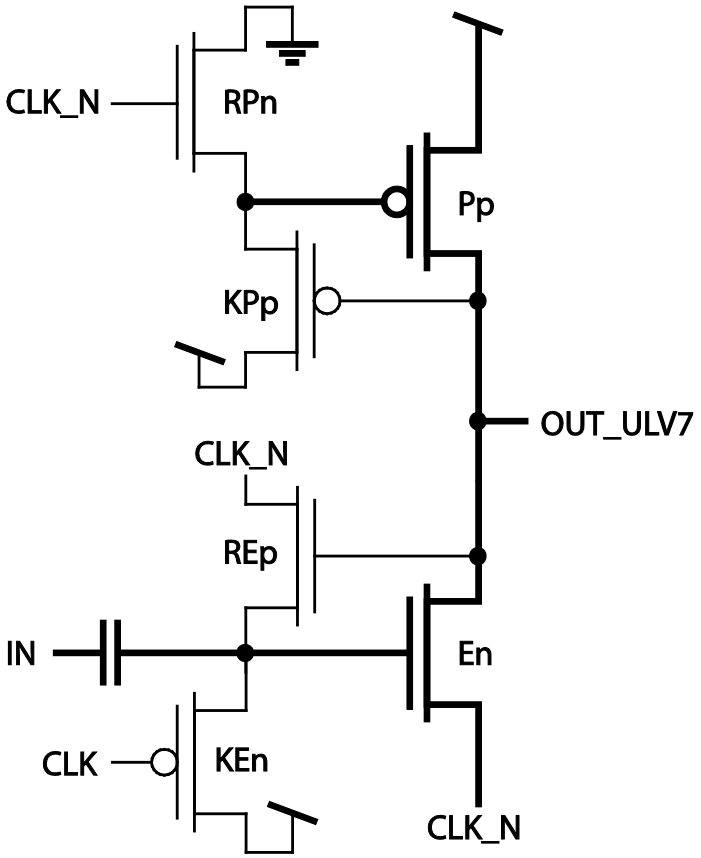

Figure 3.1.6: Implemented ULV7 inverter.

Transistors that are connected to die pads need to be shielded using guard rings to protect against static electricity and voltage spikes, because all the transistors in this design are connected to an input or an output, they are all protected with guard rings.

The final sizing of the transistors is shown in Table 3.1.2 and is a result of optimizing the circuit operation for layout simulations. There are two things that stand out in this table, one is the ***KEn*** transistor with a size of only 2 fingers and the other is the large ULV7 ***En*** transistor. The reason for the small size of the ***KEn*** transistor is that this transistor drains the floating node during the evaluation phase as long as the output is '1' making it degrade the circuit performance after a certain amount of time, so a smaller transistor means a longer operational evaluation phase and a larger transistor means a more static '1'. The sizing of the ULV7 ***En*** transistor is optimized for speed based on layout simulations that allow the ***En*** transistor to be larger than the ***Pp*** and still produce a strong '1' and a low-delay output transition.

	ULV7						CMOS	
Transistor:	En	Pp	KPp	RPn	REp	KEn	NMOS	PMOS
Fingers:	100	90	30	15	30	2	30	100

Table 3.1.2: Chip transistor sizing.

The optimal transistor sizing obtained when running schematic simulations were closer to the sizes intuitively expected, but the layout simulation optimizations were chosen as the more reliable source because more variables, parasitics and layout specific choices are considered.

The Standard inverter has a PMOS-to-NMOS size ratio of 3 to 1 because the NMOS is approximately 3 times stronger than the PMOS in the ultra-low voltage region as shown in section 2.3.1. A segment of the conventional CMOS layout displaying the NMOS (***En***) and PMOS (***Ep***) transistors with guard rings and the substrate ground connection mesh for isolation is shown in Figure 3.1.7.

Because of the small ***KEn*** transistor, a discharge diode is added to the ***CLK*** input to satisfy the TSMC antenna rules requiring a certain area of oxide diffusion per area of metal connected. This can be seen next to the ***CLK*** pad connection in Figure 3.1.10 on page 26. The size of this diode is purely based on approximations and intuition because the connected pad frame is a TSMC trade secret, making the amount of connected metal hidden. Final design rule checks (DRC) and layout versus schematic (LVS) checks were run with the pad frame by TSMC before production to allow correcting errors arising from the added metal.

To reduce the area impact of the capacitor, a rotative metal-oxide-metal (RTMOM) capacitor like in the high-yield layout is chosen because it can be placed in the metal layers above the substrate and logic. The RTMOM capacitor has a simulated capacitance of 890fF and is placed next to the transistors in the layout for improved lucidity as shown in Figure 3.1.10. The full capacitor specifications are presented in the table below.

Model		Capacitance	Fingers				Metal layers	
Name	IMEC #	Simulated	Width	Spacing	Horiz	Vert	Bottom	Top
CRTMOM	2668561832	890.403fF	140nm	140nm	200	20	1	7

Table 3.1.3: RTMOM Capacitor details

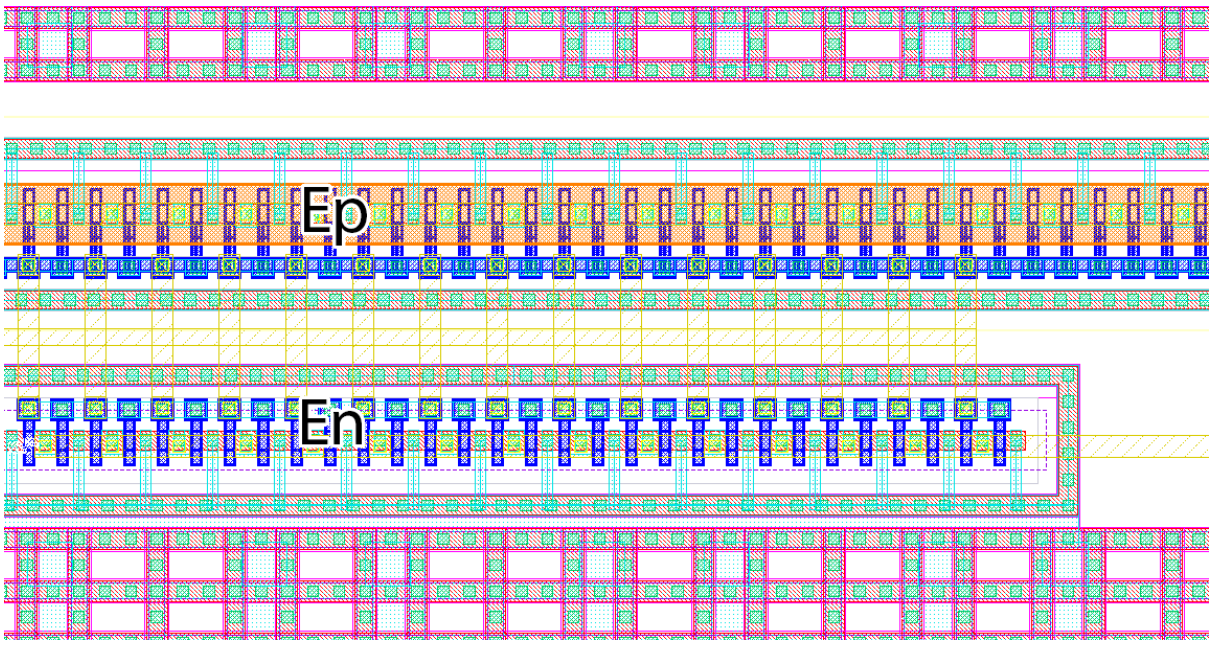

Figure 3.1.7: Traditional CMOS inverter layout.

The segment of the ULV7 layout in Figure 3.1.8 shows the placement of the transistors and the guard rings combined for the NMOS and PMOS transistors respectively. A layout overview and size comparison of the inverters and their placement is shown in Figure 3.1.10 on page 26.

Some capacitance is also added on the Vdd connections to reduce supply voltage noise, seen as the green grid on each side of the circuit in Figure 3.1.9.

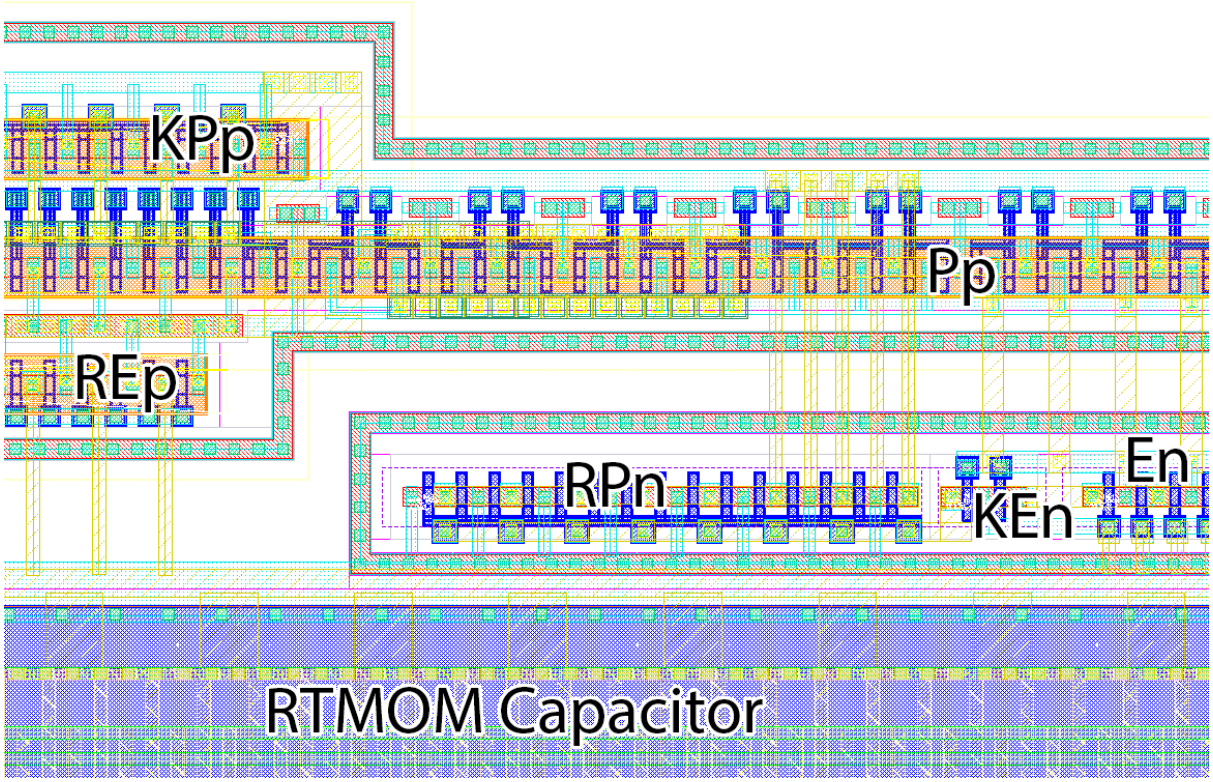

Figure 3.1.8: ULV7 inverter layout.

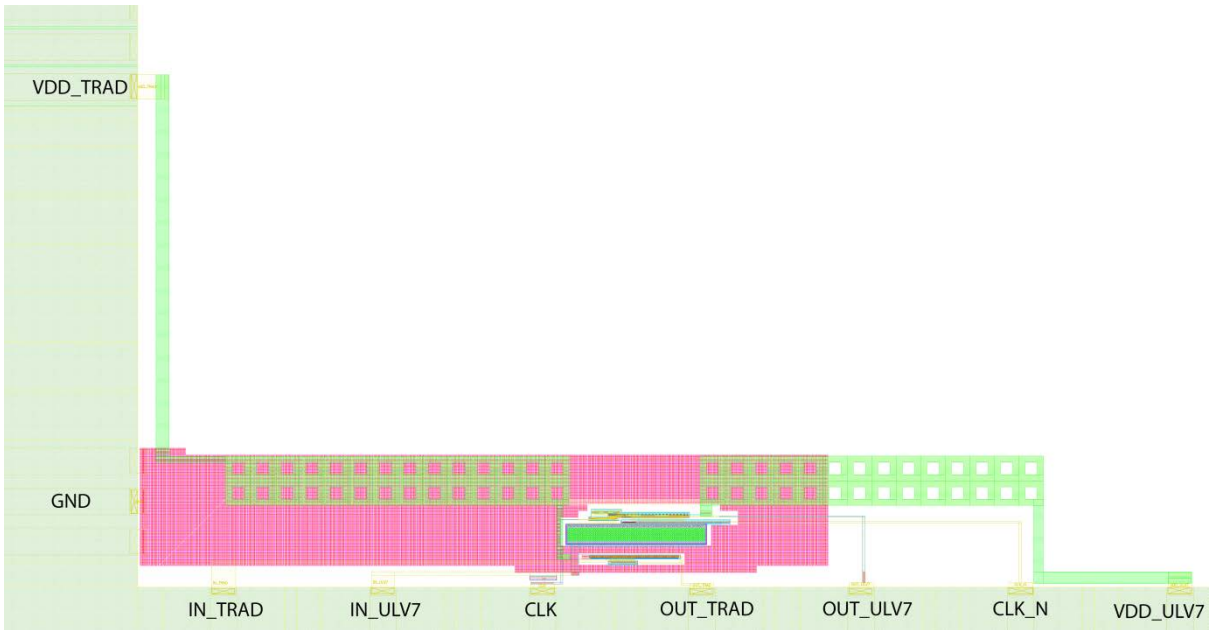

Figure 3.1.9: Full Layout.

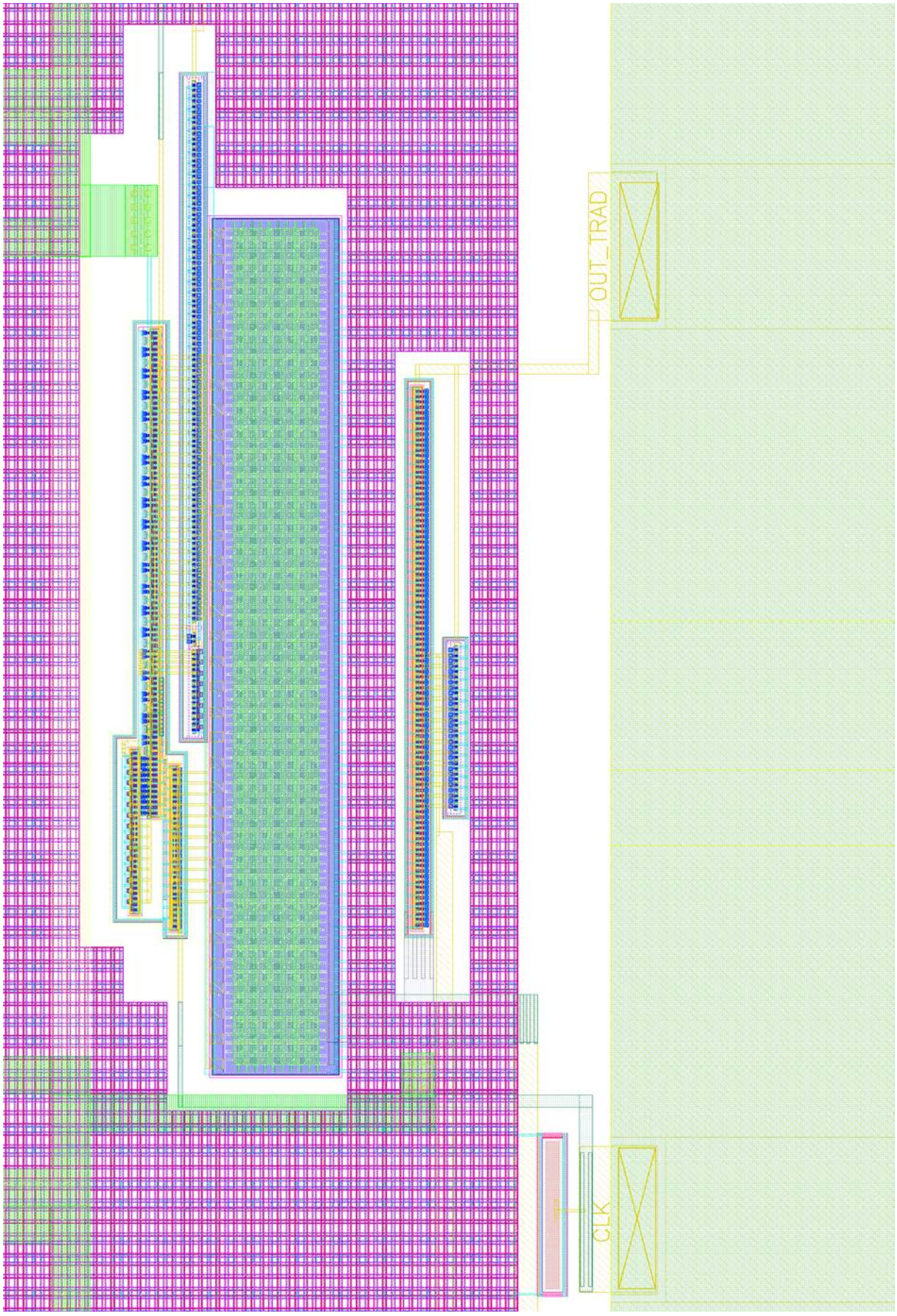

Figure 3.1.10: Layout overview

3.1.4 Simulations

The circuits were simulated on both schematic and layout level. The same test bench is used for both inverters with the only difference being the model of the design; where one is extracted from the circuit schematic and the other from the layout, yet the nominal simulations show significant differences between the two. The expected result would be a slight reduction in performance due to parasitic capacitance, resistance and inductance but this is not the case here because the performance is significantly increased in the layout with a larger difference than expected. The most reasonable explanation is that the transistor models used are not properly installed and/or not properly tested for Ultra-low voltage operation with a large number of fingers.

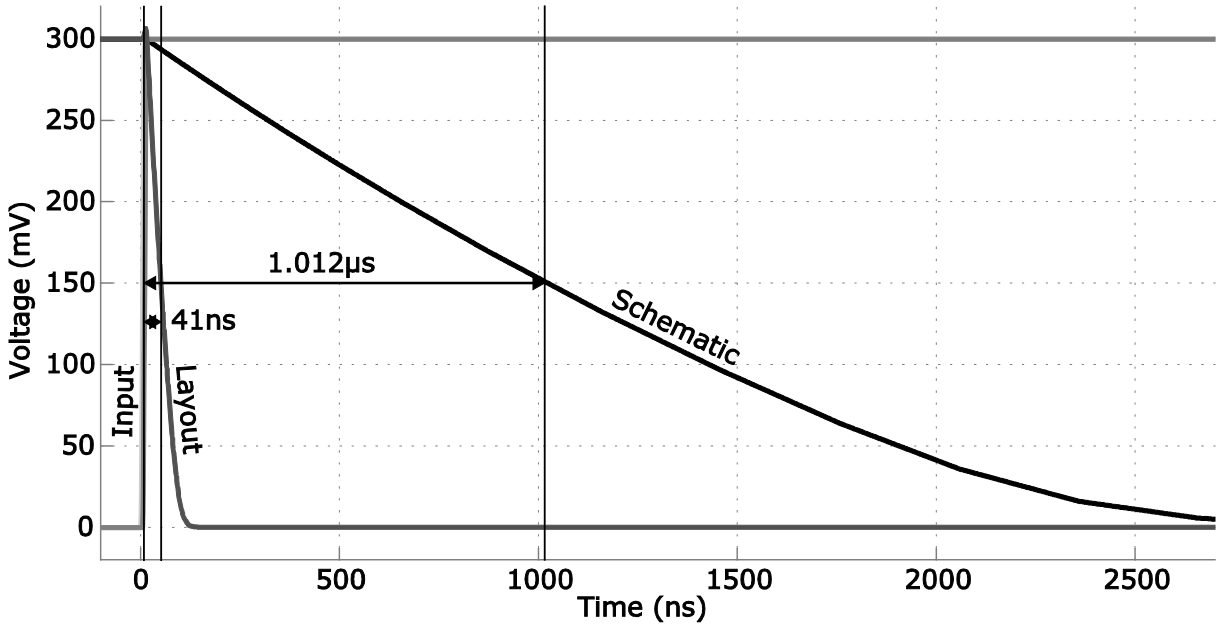

Figure 3.1.11: Simulation of schematics and layout of the conventional CMOS inverter.

The differences between layout and schematics simulation for the ULV7 inverter are a lot smaller than for the traditional CMOS inverter layout. This could be an indication of a lower threshold voltage for the layout version of the transistors because the ULV7 circuit is less susceptible to these changes. The differences can be seen in Figure 3.1.11 and Figure 3.1.12; they show that the delay of the traditional CMOS inverter is 96% lower than the schematic of the same circuit. The layout simulation of the ULV7 inverter is affected in a different way and introduces the distortion of the output signal seen in Figure 3.1.12, resulting in a higher fall time.

Throughout the design process, the layout simulations are chosen as the more trustworthy verification source because it considers more parameters, so the circuits are designed to work optimally in layout conditions.

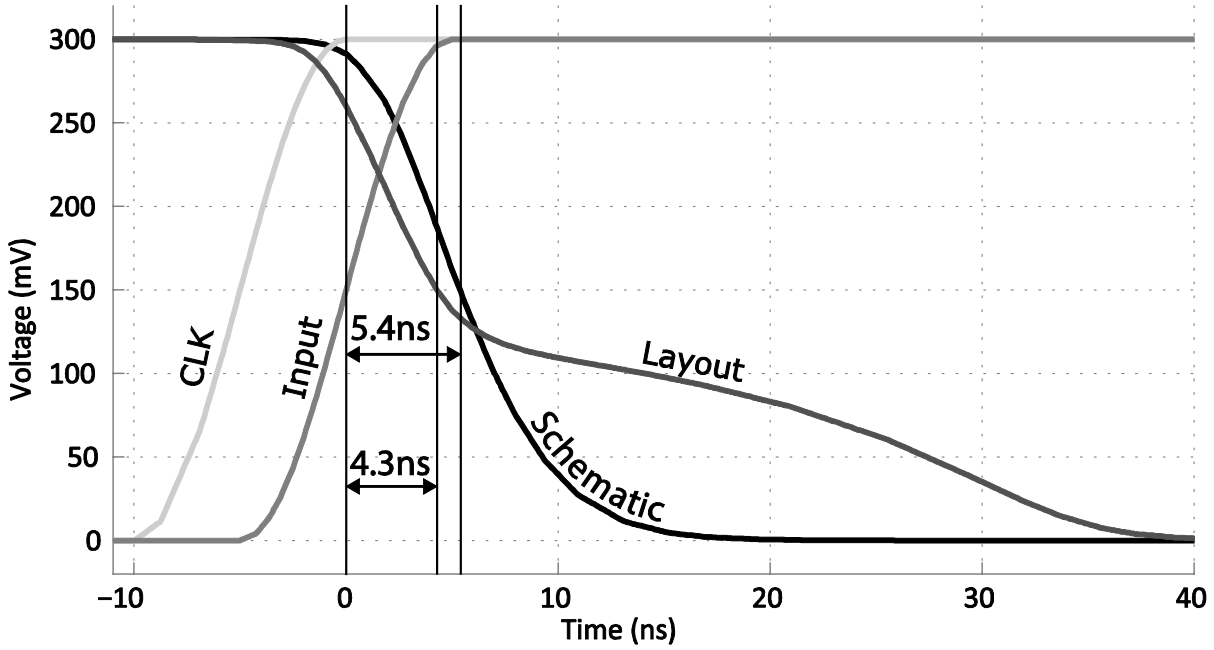

Figure 3.1.12: Simulation of schematics and layout of the ULV7 inverter.

It might seem as though the threshold voltage of the layout transistor model for many-fingered transistors is lower than the schematic model because of the increased speed of the traditional inverter and what seems like a large leakage from the floating gate introducing distortion of the layout ULV7 inverter. For the traditional inverter the transient response of the layout seems to be closer to the schematic when half the supply voltage is used, as shown in Figure 3.1.13.

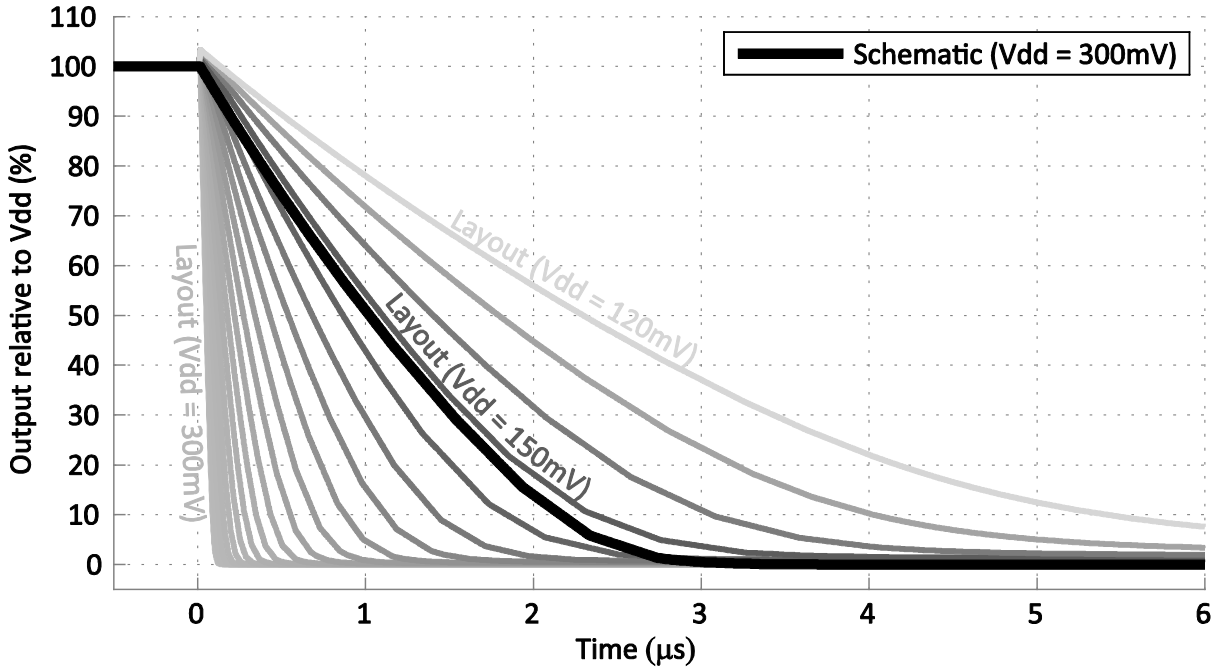

Figure 3.1.13: Simulation: Sweep of Vdd on the CMOS inverter layout, compared to the schematic.

As these differences are most likely due to Cadence installation errors and not design, they will not be discussed further.

3.1.5 Final chip

After production the chip was bonded and packaged at the TSMC foundry in Taiwan. The package used is a J-lead ceramic chip carrier with 84 leads (JLCC84). It has 21 leads on each of its four sides, and is bonded using gold wires. The finished and bonded package is shown in Figure 3.1.14.

The RTMOM capacitor and the M9 Vdd paths can be seen in the red square in the lower left corner of the die close-up in Figure 3.1.15 and the rest of the chip area belongs to a different project.

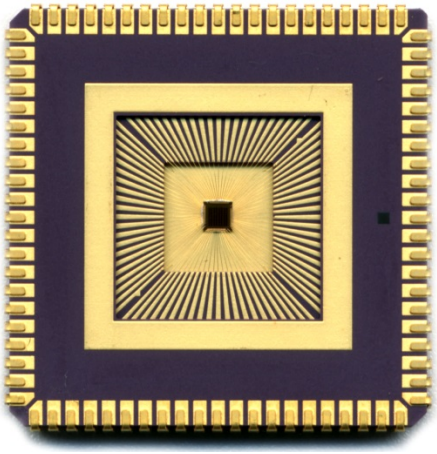

Figure 3.1.14: Final chip bonded in JLCC84 package.

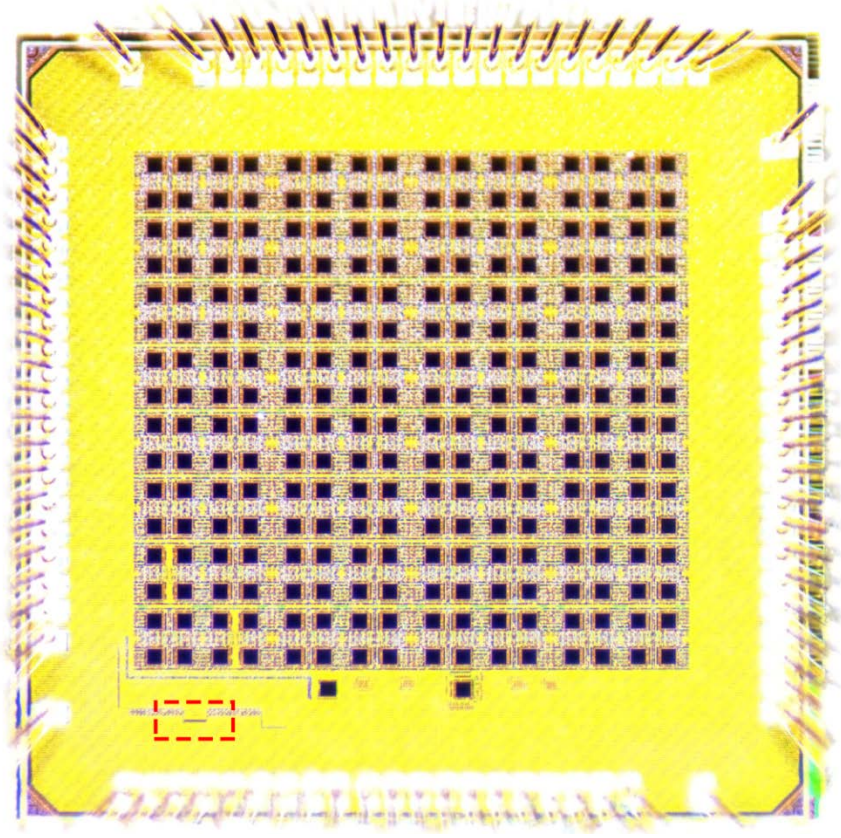

Figure 3.1.15: Die close-up, the capacitor and M9 Vdd paths are visible in the red square.

3.1.6 Circuit Board

To do measurements on the chip, a printed circuit board (PCB) is needed to connect the leads of the chip carrier to the right probes and wires. For this purpose a PCB was designed using EAGLE and manufactured in China by Elprint. It is a small and simple double-sided design with one capacitor of 10uF and one of 10nF added to each supply voltage path to reduce noise. These can be seen in Figure 3.1.18 as ***C2***, ***C3***, ***C4*** and ***C5***. The top side copper layer is shown in Figure 3.1.16 and the bottom side copper layer which is all connected to ground can be seen in Figure 3.1.17. A series of pins need to be connected to ground to disable the other project on the chip. All the ground connected pins have vias placed next to them; the vias and component placement is shown in Figure 3.1.18.

Because the JLCC84 package is the ceramic equivalent of the PLCC84 package, a PLCC84 socket is used to make changing the chip easier when measuring several chips to analyze variations in the production process.

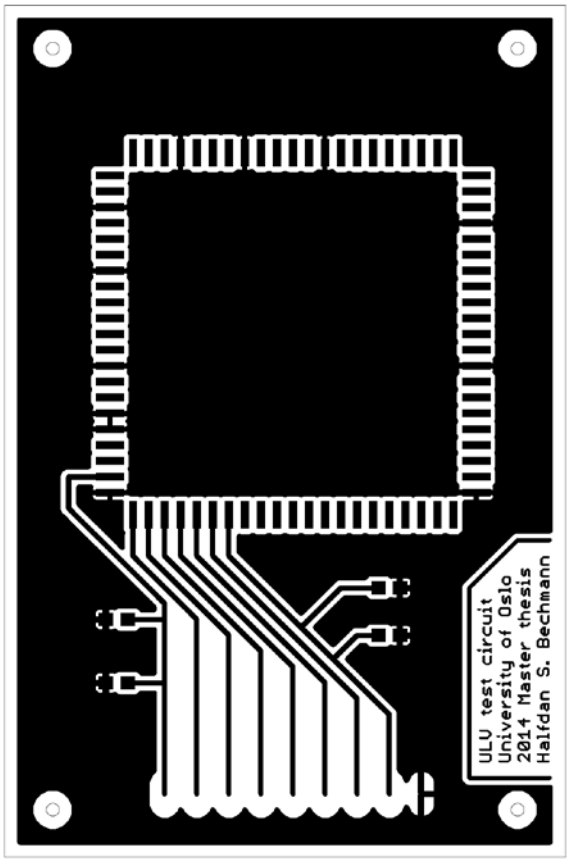

Figure 3.1.16: PCB: Top electric layer.

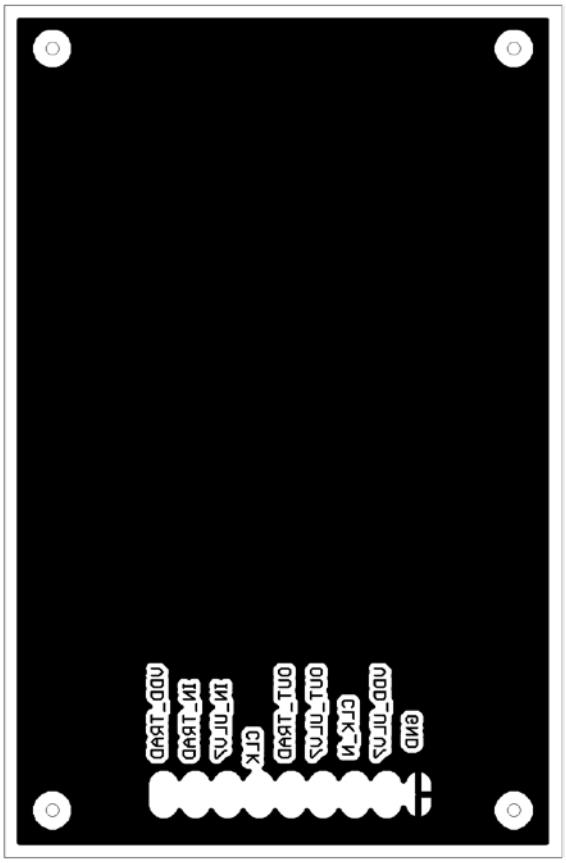

Figure 3.1.17: PCB: Bottom electric layer.

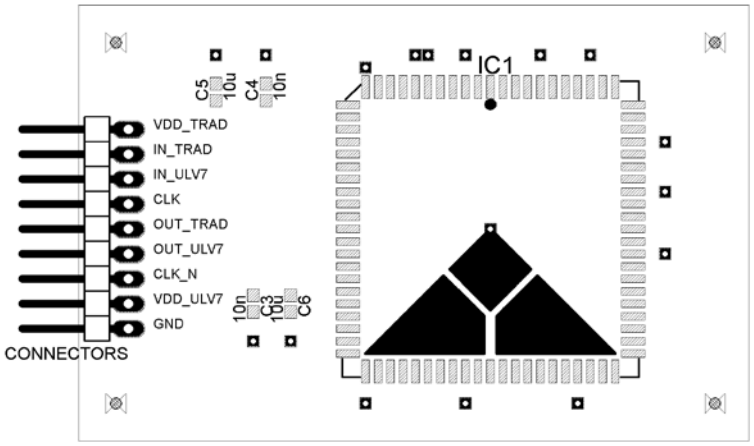

Figure 3.1.18: PCB: Top paste, silk and via layers.

Both surface-mount technology (SMT) and thru-hole technology (THT) is used on this PCB because SMT is quick and easy for larger quantities of components and pads when the proper equipment is in place and THT is necessary for the chosen header connectors. The surface mounted devices (SMDs) were soldered on by applying solder paste manually to the required pads and using a reflow oven for the curing process, but the thru-hole pin header was soldered on manually. The top and bottom of the finished PCB are shown below and the soldered PCB with the chip carrier mounted is shown in Figure 3.1.21.

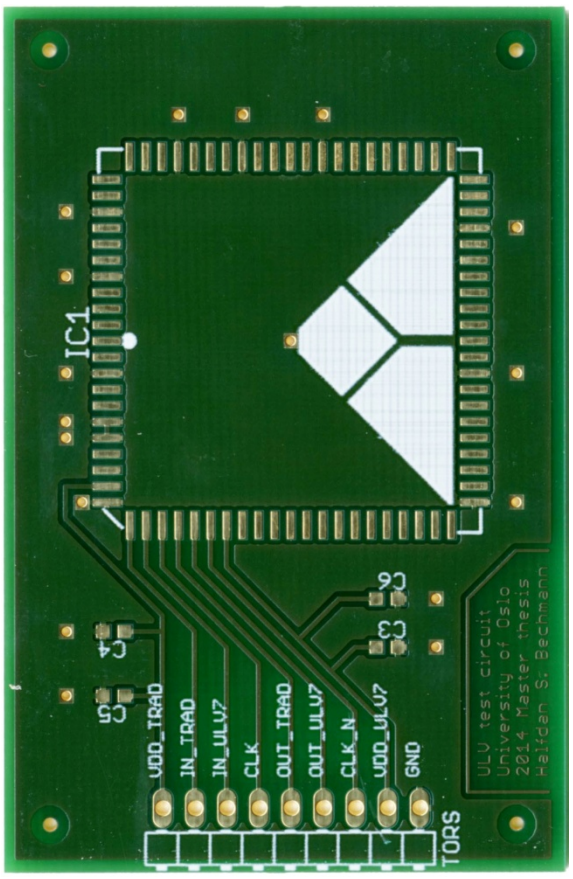

Figure 3.1.19: Top side of the finished PCB.

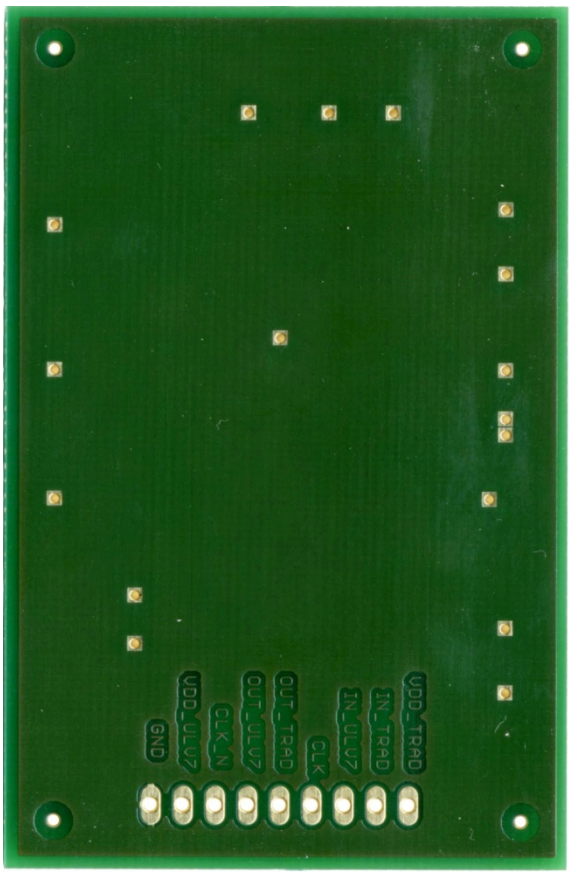

Figure 3.1.20: Bottom side of the finished PCB.

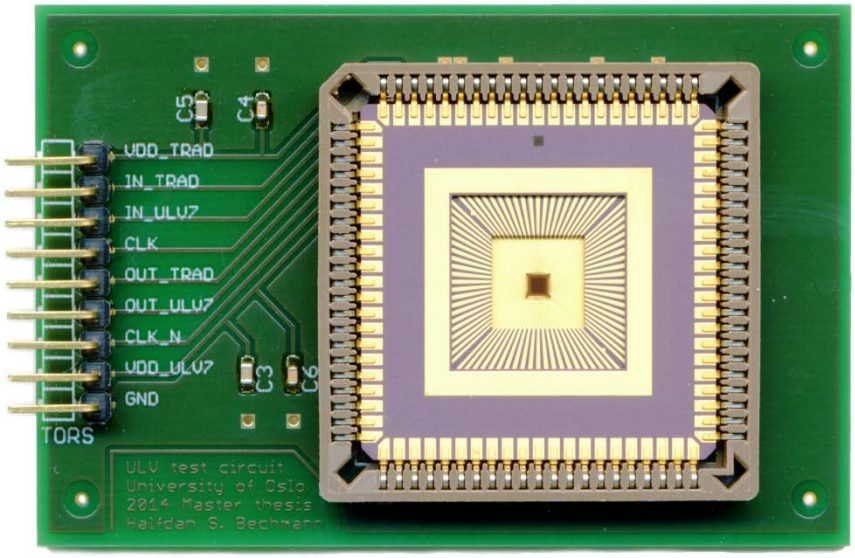

Figure 3.1.21: Soldered PCB with chip carrier mounted.

3.1.7 Test setup

To measure the analog properties of the circuits, a proper test setup is needed. To be able to do scripted measurements and run parameter sweeps it is important that all the instruments can be controlled remotely. One of the simplest protocols for this is the general purpose interface bus (GPIB), which is found on most high-end instruments, is the protocol that will be used for remote instrument control in this setup. When it comes to the circuit inputs, the ULV7 inverter needs 3 input signals in addition to the ground and supply voltage, and the inverter needs one. This creates a demand for four different channels which is available on the TGA1244 from Thurlby Thandar Instruments (TTi). The TGA1244 will therefore be used to generate the needed inputs. A remote controlled voltage source is also needed and the Agilent HPE3631 is chosen for this purpose. Lastly, an oscilloscope with at least 4 channels is needed to measure both inputs and outputs, or all three inputs and the output of the ULV7 inverter. The Agilent HP54622 is chosen as the oscilloscope for the setup.

An illustration of the full test setup is presented in Figure 3.1.22 and examples of the measurement scripts run are provided in the appendix (section 5.2.1).

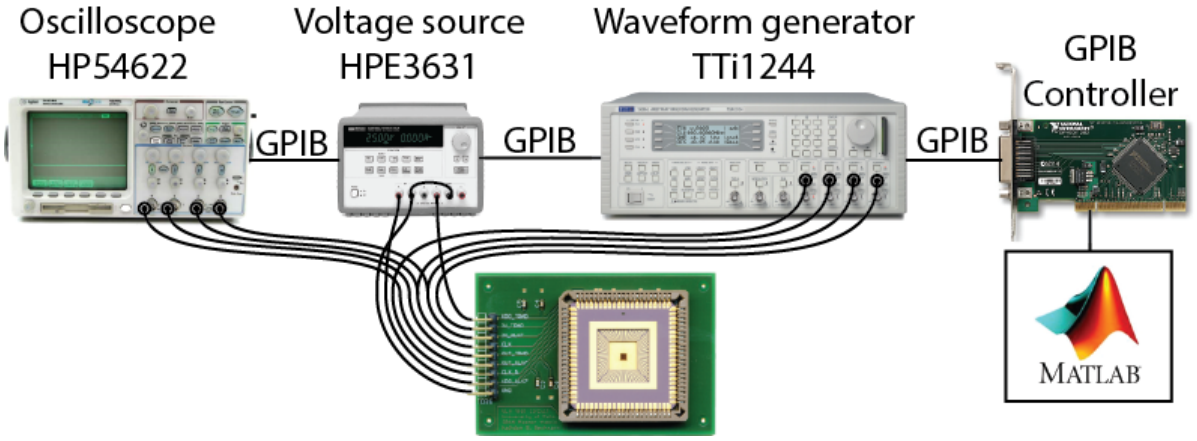

Figure 3.1.22: Test setup

3.1.8 Measurements

3.1.8.1 Simulation comparison

In this section the measured results will be compared to the previous simulations to get an idea of the simulation accuracy of the tools and models used.

The final measurements seem to be closer to the schematic simulations than the layout simulations. This is shown in Figure 3.1.23 and Figure 3.1.24 and might indicate that the layout models are not properly adjusted or calibrated for the large transistor sizes and low voltages used in this design. Because the transistors were scaled under the assumption that the layout simulations were the most accurate; the strength of the evaluation transistor in the ULV7 inverter has been set too high. This introduces the need for a post-production strength reduction of the ***En*** transistor, which is performed in the next section.

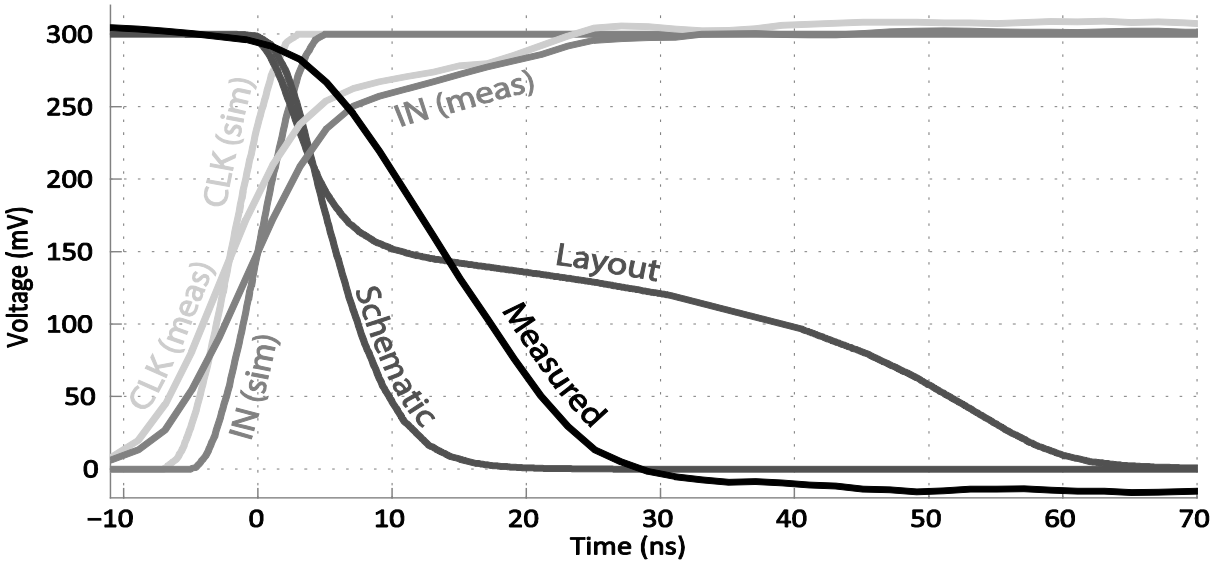

Figure 3.1.23: ULV7 inverter, schematic simulation, layout simulation and measured results.

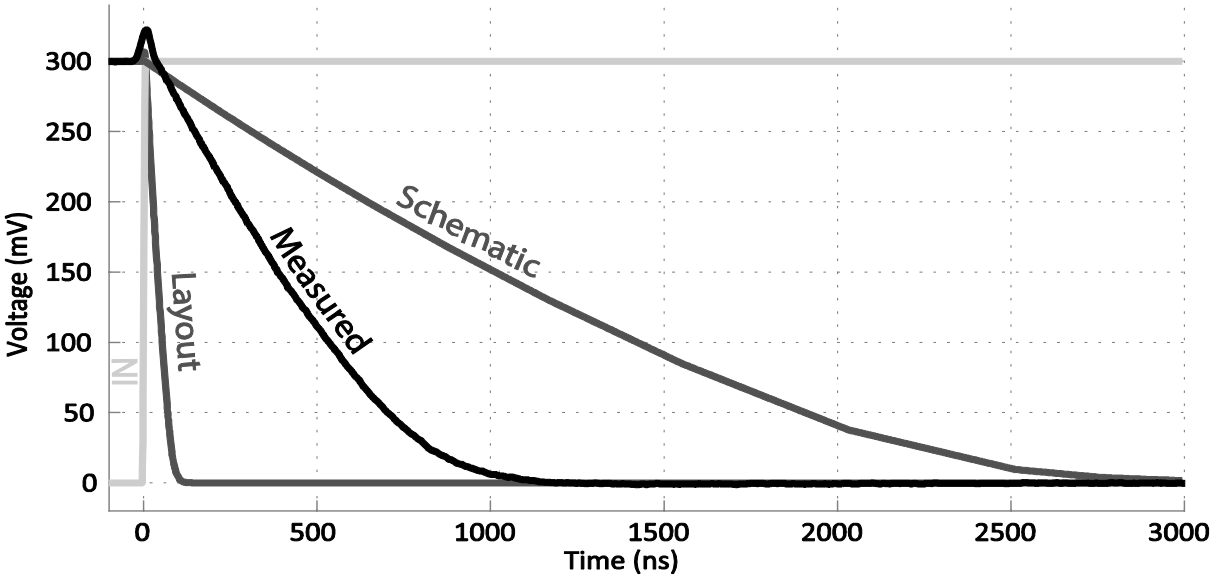

Figure 3.1.24: Schematic and layout simulation, and measured results of the conventional CMOS inverter.

The measurements have been compared to the simulations and the nominal schematic simulations both have a delay that deviates from the average measurements by approximately a factor two. This deviation is within $2 \times \sigma$ for the schematic Monte Carlo simulations, making the schematic results acceptable. The layout simulations from this setup on the other hand should be disregarded because the average of the measured results deviate from the nominal simulations by more than $5 \times \sigma$ in addition to introducing distortions.

3.1.8.2 Postproduction adjustments

When the chip came back from production and measurements were performed the ***En*** transistor in Figure 3.1.26 was found to be too strong resulting in an erroneous output value as shown in Figure 3.1.27 and Figure 3.1.28. To perform a post-production strength reduction, the precharge level of the ***En*** floating gate needs to be lowered.

Because ***V_OFFS+*** in Figure 3.1.25 is hard-wired to Vdd in the implemented circuit as shown in Figure 3.1.26 due to a limited number of pads weakening the ***En*** transistor requires some creativity. One way to lower the precharge level is to apply a positive offset to the ***CLK*** signal, reducing the ***REp*** current and thereby the time it takes to charge the floating gate. The offset needed is therefore influenced by both the clock frequency and the supply voltage, so by adjusting both correctly the precharge of the floating gate can be stopped before the gate is fully recharged resulting in a lower operation point and a weaker transistor.

Valid results using this method was not achieved for supply voltages lower than 190mV because the measurements at these voltages require a long precharge period, where the current through ***REp*** could not be reduced enough using reasonable offset voltage levels. For voltages of 190mV and above the postproduction adjustments were successful.

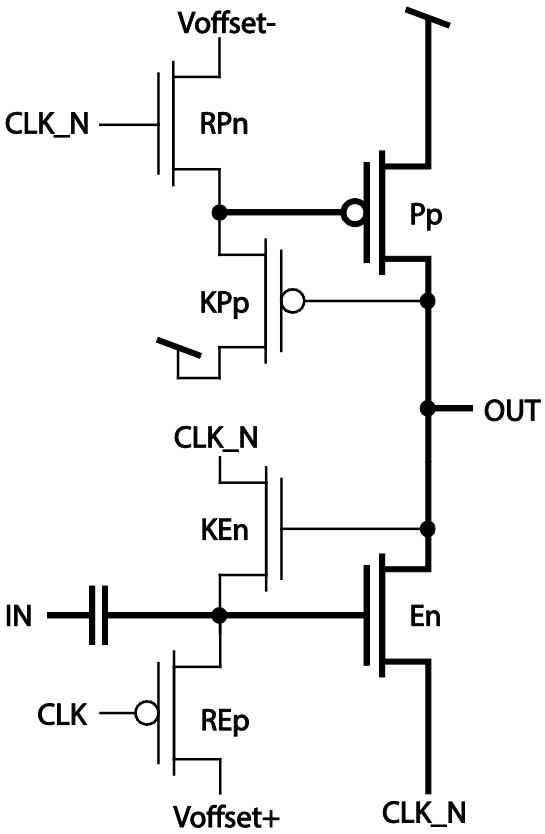

Figure 3.1.25: Schematic: N-type ULV7 inverter.

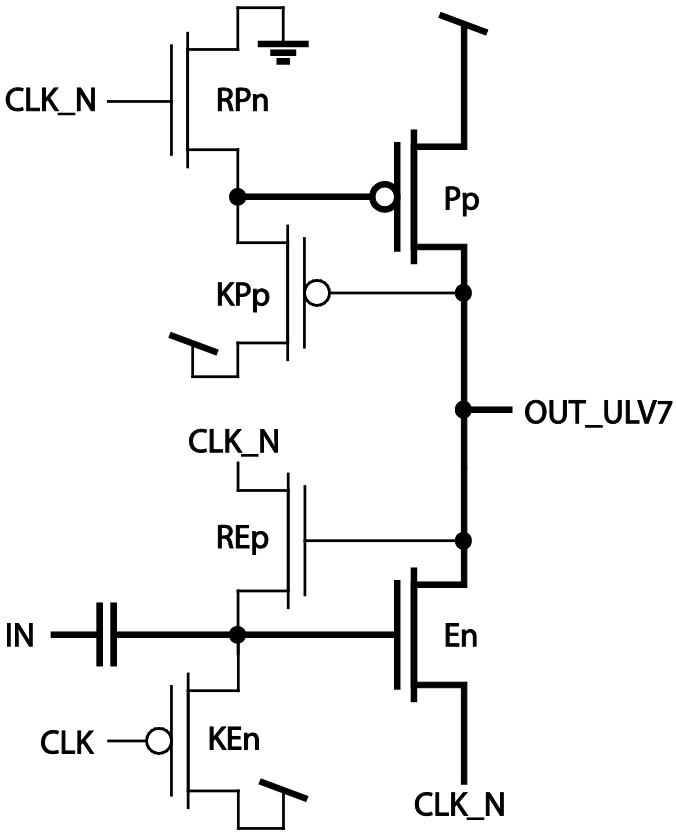

Figure 3.1.26: Schematic: Implemented N-type ULV7 inverter.

At 300mV an offset of 170 mV results in a stable logic ‘1’ output and the inverter delay is actually reduced by 6%[1] most likely due to less charge leaking trough REp at the start of the evaluation phase. To achieve a stable logic ‘1’ with a 200mV supply voltage, a 120mV offset is needed. The delay using this offset is 19%[2] higher than without the offset due to the lower floating gate voltage. The difference between the ‘1’ with and without the offset applied is shown in Figure 3.1.27 for the 300mV measurements and in Figure 3.1.28 for the measurements run at 200mV. The input signal is not shown but keeps its precharge value of ‘0’ for the Logic ‘1’ output and switches to ‘1’ for the Logic ‘0’ output.

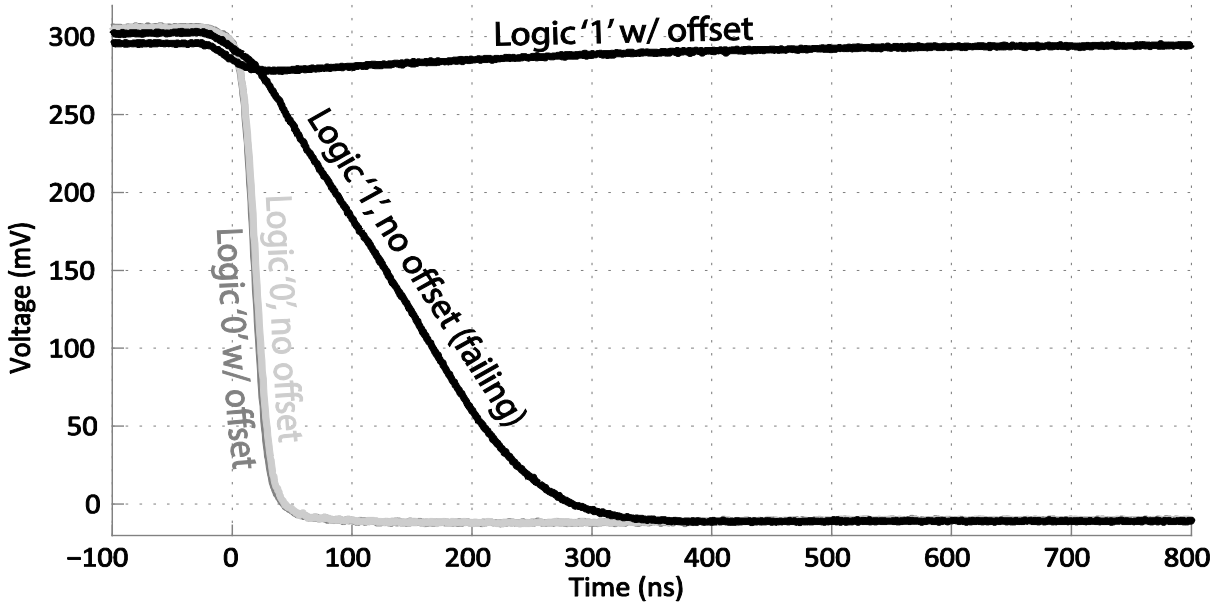

Figure 3.1.27: Measured: ULV7 output (170mV offset on CLK at 300mV and 400kHz).

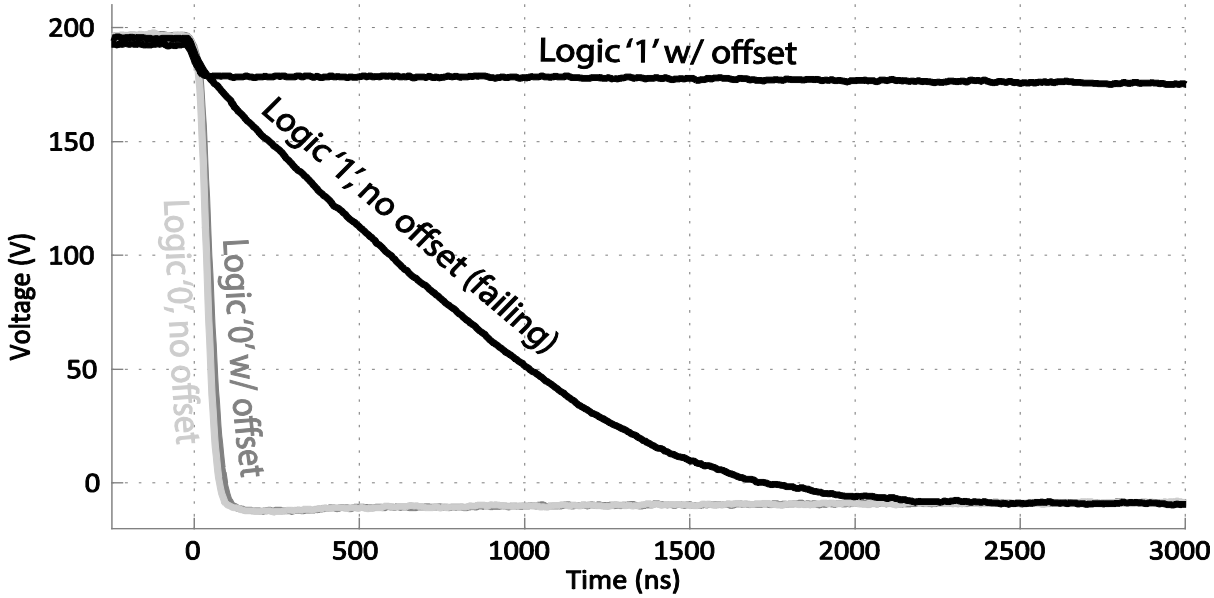

Figure 3.1.28: Measured: ULV7 output (150mV offset on CLK at 200mV and 40kHz).

[1] Delay measured at 200mV: w/o offset: 19.8ns, w/ offset: 18.6, 1-18.6/19.8=0.0606
[2] Delay measured at 300mV: w/o offset: 36.ns, w/ offset: 43ns, 43/36-1=0.1944

3.1.8.3 Results

The transient measurements of the inverters shown in Figure 3.1.29 clearly show the difference in delay between the two. Here shown with a supply voltage of 200mV. The capacitive coupling between the input and output is also less apparent. The input of the traditional inverter is significantly steeper than what is realistic for this circuit. This is done because of limitations in the frequency generator not allowing phase-lock synchronization of custom waveforms. This is achievable for square-waves, which is why these are used for clocks and inputs for both circuits with a 10MHz Bessel filter applied to increase the rise- and fall-time. Because of these limitations, the measurements portray the traditional CMOS circuit as slightly faster than it should, which causes the differences between the two measured inverters to be slightly understated.

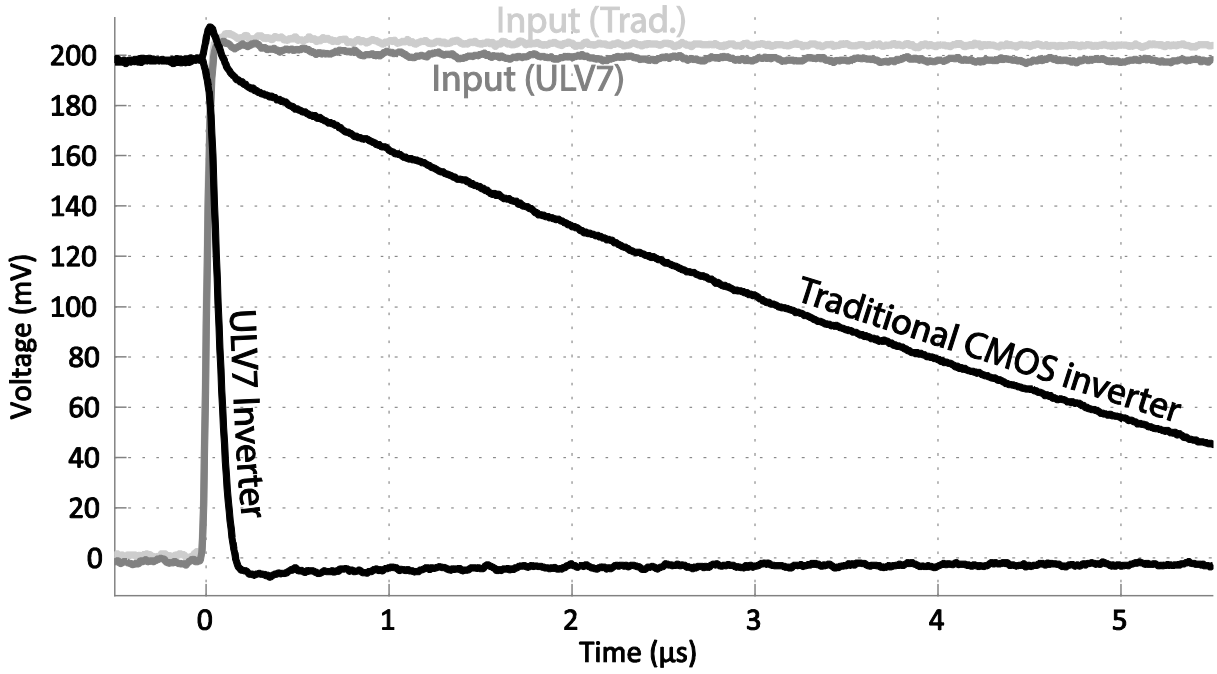

Figure 3.1.29: Measured: Transient response of the inverters at 200mV.

The delay for both inverters is highly dependent on the supply voltage, but the ULV7 circuit is less affected by voltage changes than the standard inverter is.

The delay dependence on the supply voltage for the two inverters is shown in Figure 3.1.30 where the delay of the conventional CMOS inverter increases by two orders of magnitude between 500mV and 190mV while the delay of the ULV7 inverter increases by less than one. This results in an increasing difference in delay between them when the voltage is lowered, highlighting one of the benefits of the ULV7 logic style. The delay of the ULV7 inverter relative to the traditional inverter is plotted for a range of supply voltages in Figure 3.1.31. The average relative delay for the ULV7 inverter varies from 24.6% at 490mV to 1.5% at 190mV. At 190mV the delay of the ULV7 inverter is only 51ns while the conventional inverter has a delay of over 3.3µs. This means that the ULV7 inverter is more than 65 times faster than the conventional CMOS inverter at 190mV.

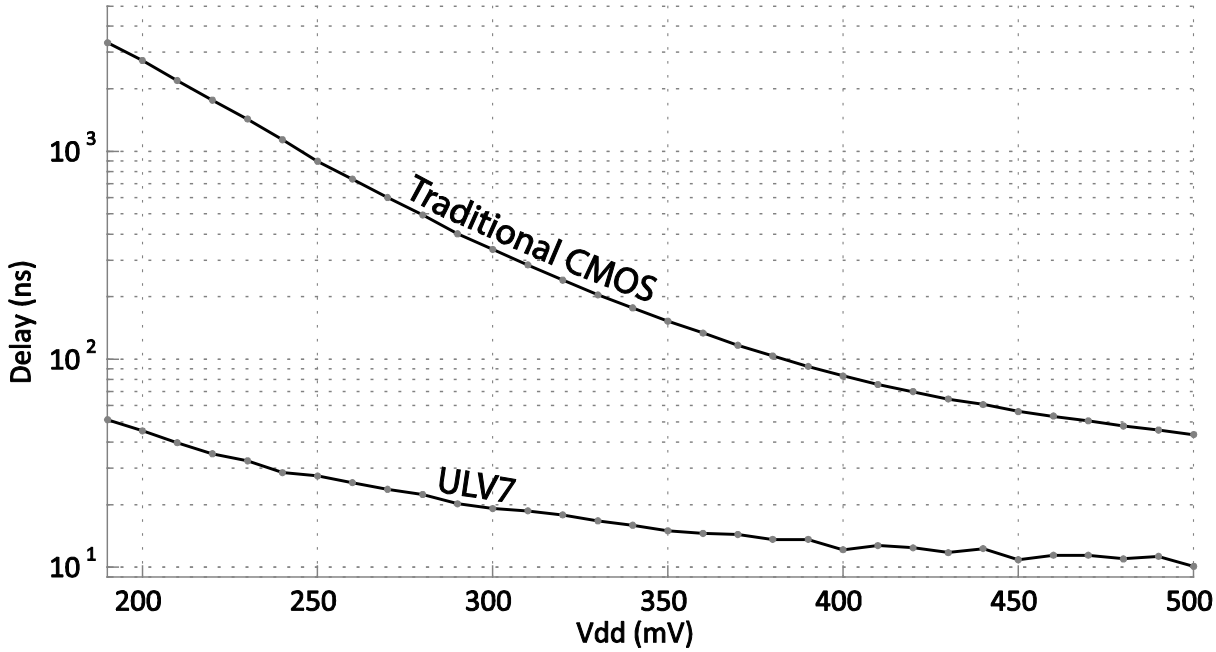

Figure 3.1.30: Delay of traditional logic and ULV7 with CLK offset.

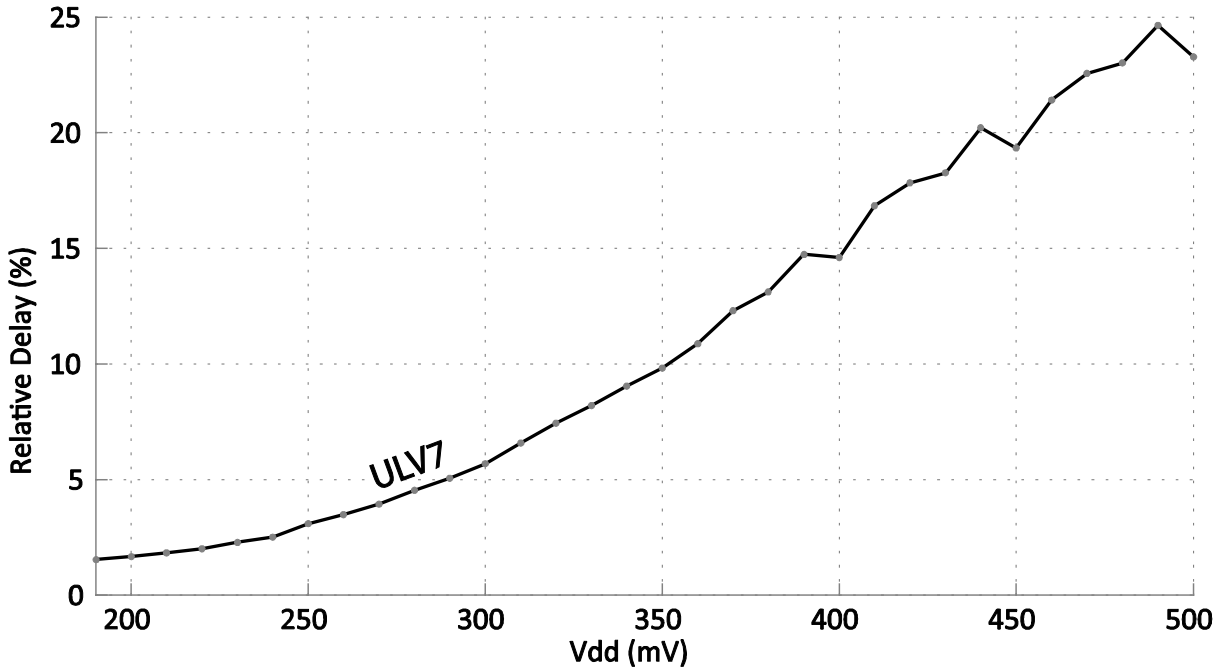

Figure 3.1.31: Delay of ULV7 inverter relative to a traditional inverter.

To further investigate the optimal supply voltage for the relative delay, a sweep is performed at even lower voltages. Even though the implemented circuit can not produce a robust ‘1’ below 190mV, the gate delay of the switching event is measured down to 100mV.

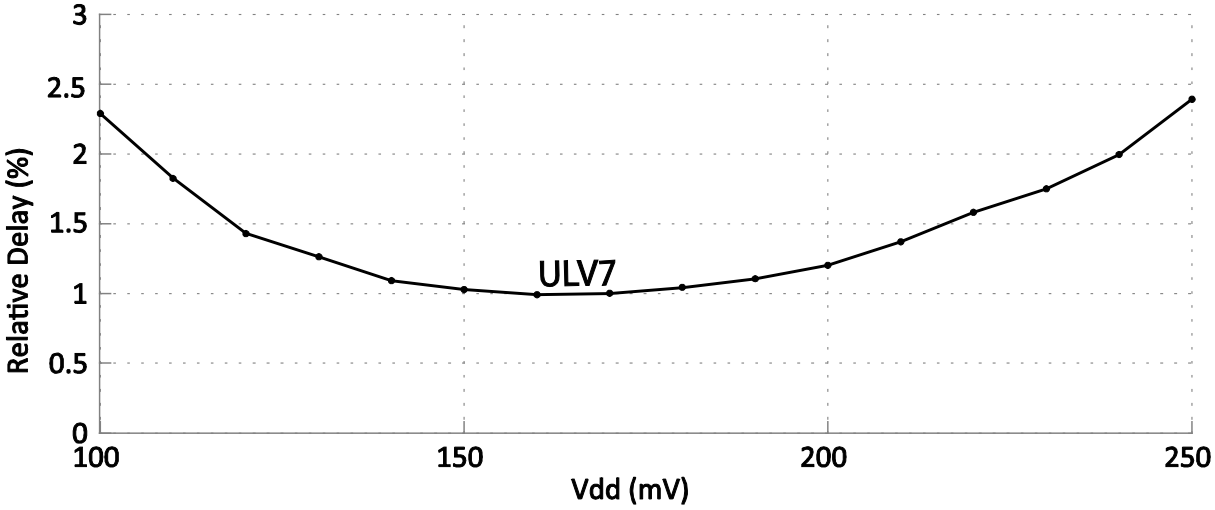

Figure 3.1.32: Measured: Delay of ULV7 inverter relative to conventional CMOS (no offset applied).

From the measurements in Figure 3.1.31 the relative delay seems to decrease even further below 190mV but the measurements plotted in Figure 3.1.32 shows that it does not continue to decrease much more below this level. The lowest relative delay is here achieved with a supply voltage of between 140mV and 190mV showing that most of the ULV7 inverter potential can be measured with this circuit.

Below 100mV the noise margin is degraded, but the ULV7 is less affected than the conventional CMOS inverter. In Figure 3.1.33 this is shown for a supply voltage of 50mV where the precharge value of the ULV7 inverter is 35mV, 70% of Vdd, and the '1' value for the conventional inverter is below Vdd/2.

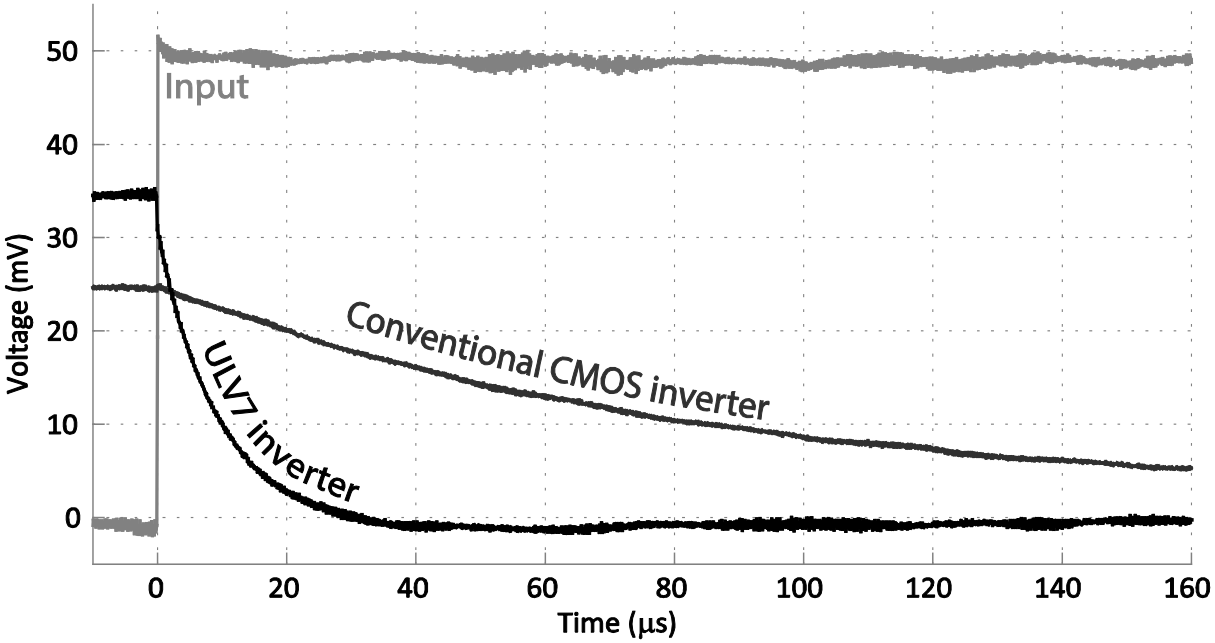

Figure 3.1.33: Transient measurements at 50mV.

Due to the larger current in the ULV7 inverter, the standard deviation (σ) of the delay is significantly better than for the traditional inverter, especially at low supply voltages. Although the measurement selection is limited to only 5 chips it can still give an idea of the difference between them. In Figure 3.1.34 the large difference in standard deviation between the two inverters and their dependence on the supply voltage is shown. An interesting observation from these measurements is that the standard deviation of the ULV7 inverter running on a 190mV supply voltage is lower than half the σ of the conventional inverter running on a 500mV supply.

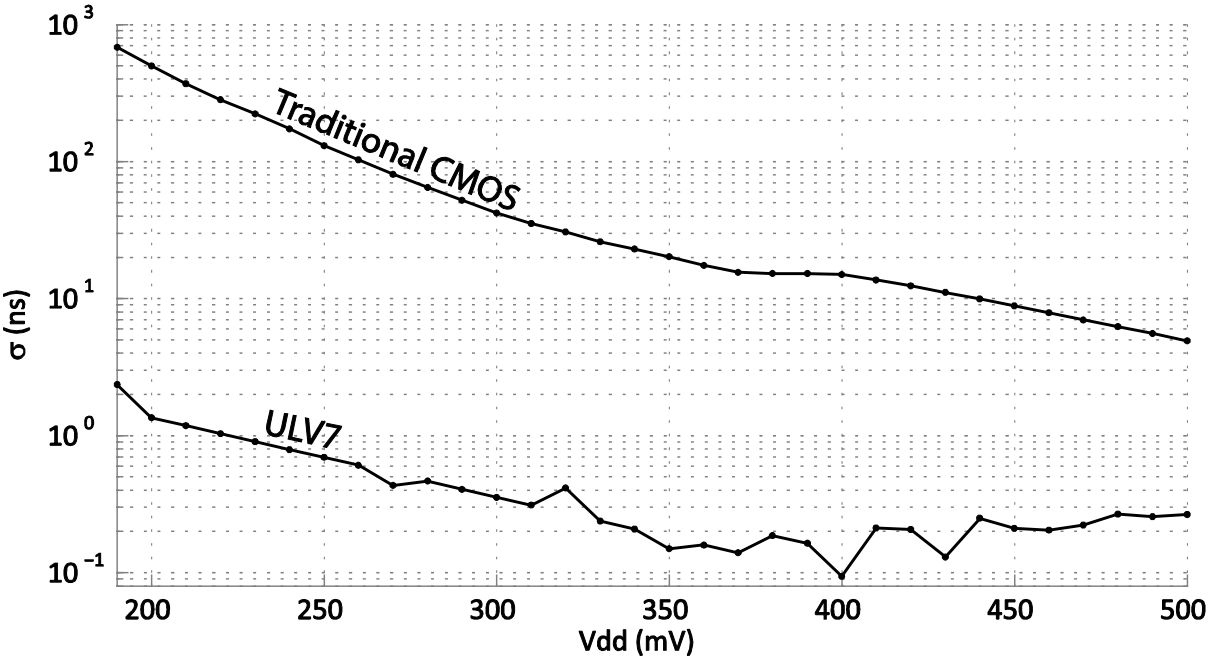

Figure 3.1.34: Standard deviation of the two inverters.

At a supply voltage of 200mV the standard deviation of the ULV7 inverter is only 0.27% relative to the standard inverter. This means that the σ of the ULV7 inverter is 370 times smaller than the conventional CMOS inverter σ at 200mV. The σ relative to the delay is also an interesting figure to consider, and for the ULV7 inverter the σ is only 2.9% of its 2.4ns propagation delay at 200mV. In comparison, the relative σ for the conventional CMOS inverter running on the same supply voltage is 18.2%, more than 6 times higher than the ULV7 inverter.

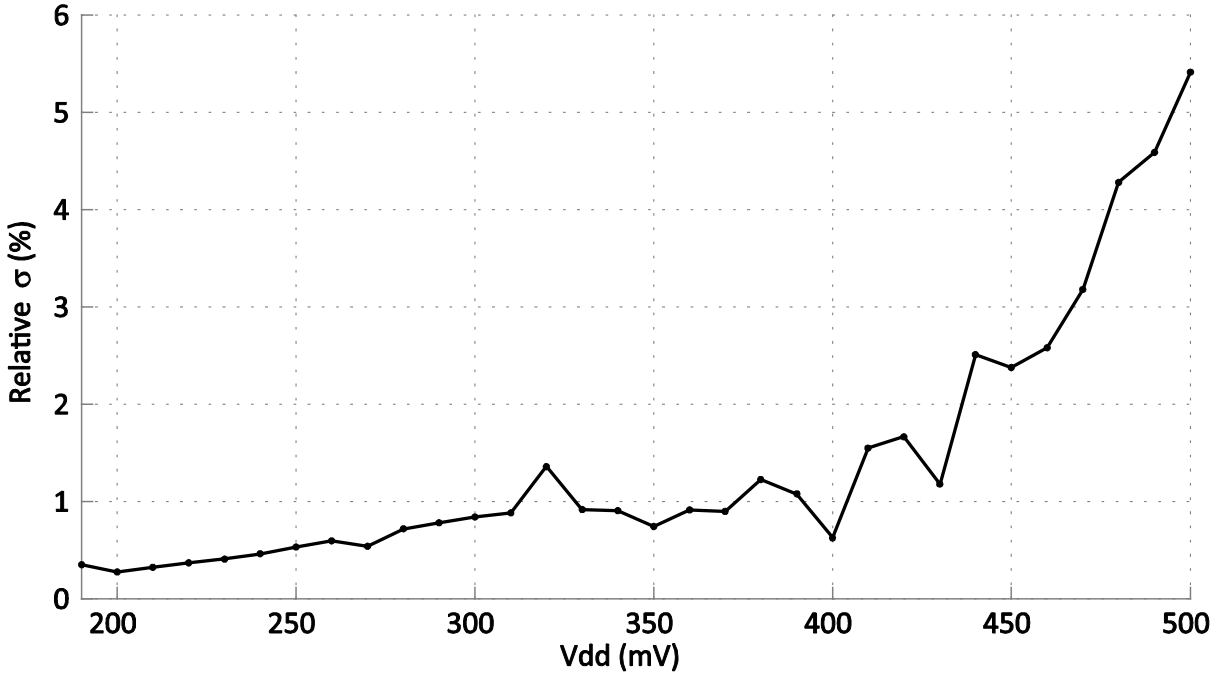

Figure 3.1.35: Measured: σ of ULV7 relative σ of traditional CMOS.

3.1.9 Conclusion

In the presented results the ULV7 inverter has proven to be more than 65 times faster than the conventional CMOS inverter with a delay of only 51ns when operating at 190mV. An outstanding result in terms of standard deviation was also achieved with a standard deviation of just 1.36ns (2.9%) at supply voltage of only 200mV which is 370 times that achieved by the traditional inverter. So according to these measurements the ULV7 logic style can be a good choice for achieving high speed and robustness at ultra-low voltages.

Vdd		ULV7	Conv.	Diff
190mV	Average Delay	51.19ns	3.33 µs	65x
	Standard Deviation	2.36ns	682.9ns	289x
200mV	Average Delay	45.35ns	2.736 µs	60x
	Standard Deviation	1.36ns	498.7ns	367x

Table 3.1.4: Comparison of the ULV7 and conventional inverter.

3.2 ULV Static Carry Generate Circuit

A high-speed and low-voltage logic style like the ULV can be utilized for many purposes, where one of the more obvious is a serial carry propagate adder because the propagation delay is directly proportional to the number of bits these circuits are capable of processing in a clock cycle. One circuit based on an early ULV generation, with a precharge to Vdd/2, is the ULV Static Carry generate circuit (ULVSC) proposed in [15] and shown in Figure 3.2.1.

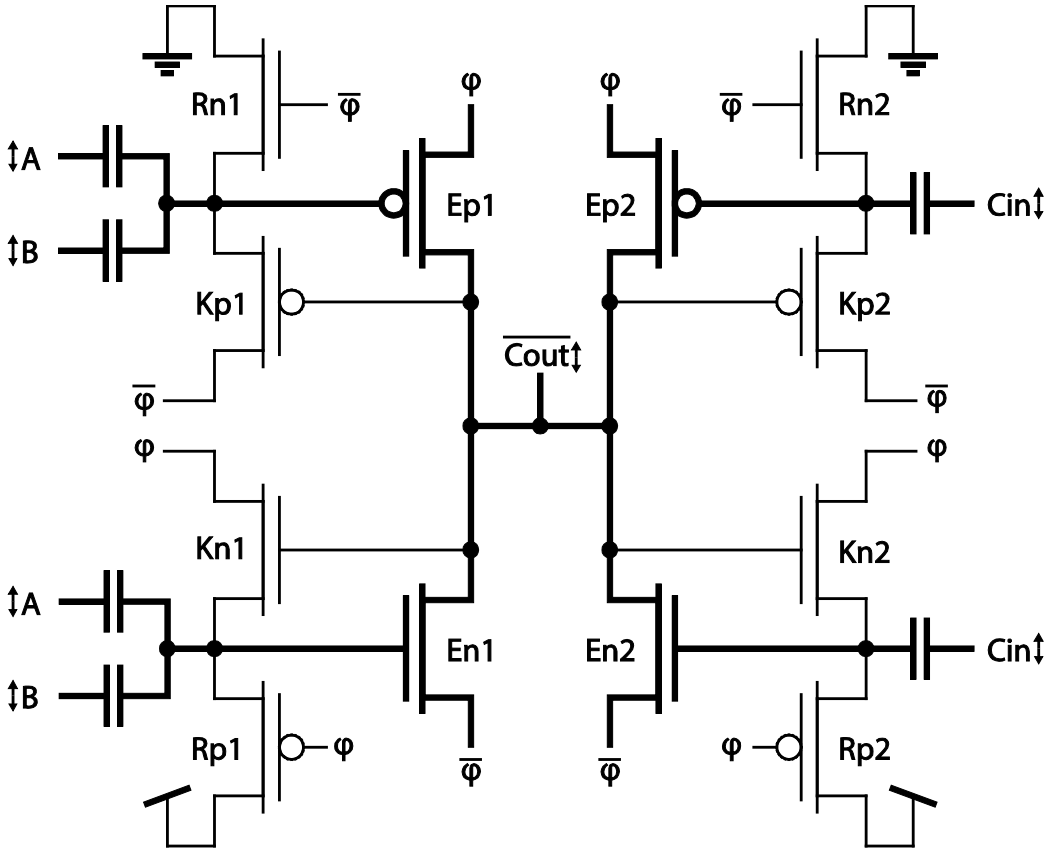

Figure 3.2.1: Schematic: ULV Static carry generate circuit [15]

This circuit is constructed by connecting the outputs of two ULV1 inverters and use two parallel capacitors on the floating gates of one. The inverter with two inputs will only generate a defined output if both ***A*** and ***B*** are equal because they will otherwise cancel each other out. The idea is to allow this part of the circuit to process the A and B inputs in parallel when the clocks enter the evaluation phase and the input signals become available. In the beginning of the evaluation phase the output either switches or keeps its precharge value of Vdd/2 until the serial part of the circuit receives a carry input. This has been illustrated in Table 3.2.1.

A	B	Cout
0	0	1 (inverted)
0	1	Wait for Cin
1	0	Wait for Cin
1	1	0 (inverted)

Table 3.2.1: Truth table for the parallel part of the ULVSC.

If the output is pulled either high or low, the ***Kn2*** and ***Kp2*** keeper transistors will drain the ***En2*** or ***Ep2*** gates to keep the achieved value. If the ***A*** and ***B*** inputs cancel each other out the output will keep its Vdd/2 precharge level until an edge is received on ***Cin*** that will decide the final value.

The ULVSC circuit, being based on an inverter, has an inverted output. On the schematics in Figure 3.2.1, the carry output is inverted and the A, B and Cin inputs are not. This is not problematic because the carry circuit truth table is symmetrical so that inverting all the input signals will invert the output and still keep the logic functionality. However this requires the inputs of every other circuit to be inverted as shown in Figure 3.2.2.

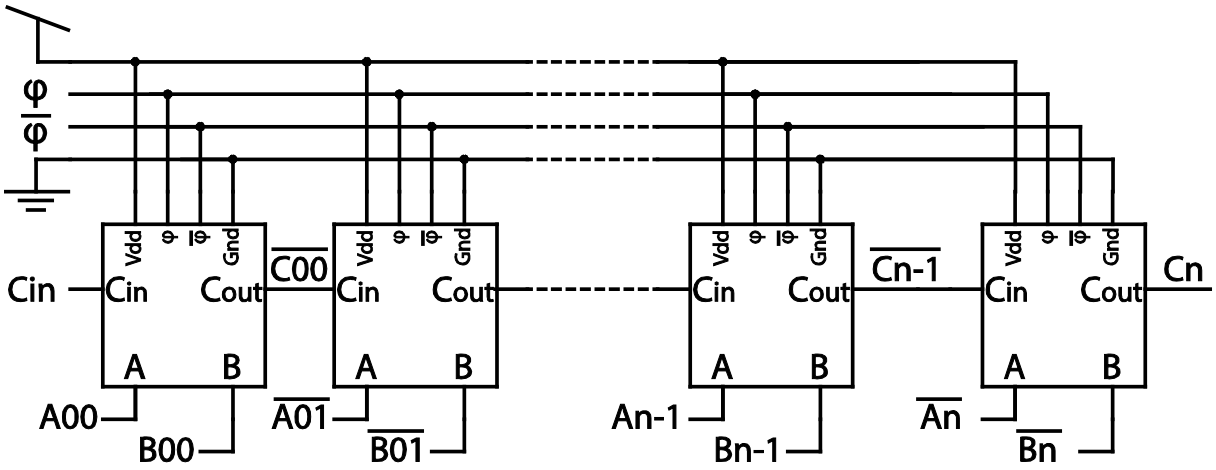

Figure 3.2.2: Schematic: Series connection of SULVC circuits.

The speed of the ULVSC can be determined by simulating a chain of these bits and measure the total time of the chain and thereby find the average delay. For this purpose, a 32-bit chain is simulated for the two worst case inputs where all bits need to wait for a propagating carry. The two cases are when all bits are waiting for a propagating ‘1’ as shown in Equation 3.2.1 and when all bits are waiting for a propagating ‘0’ as shown in Equation 3.2.2. For all cases, the carry input of the first circuit is a falling edge signal effectively making it a half-adder.

```
   0xFFFFFFFF  (A)
+  0x00000001  (B)
  0x1FFFFFFFE  (Carry)
=  0x00000000  (Sum)
```

Equation 3.2.1: Propagating '1'.

```
   0xFFFFFFFF  (A)
+  0x00000000  (B)
  0x000000000  (Carry)
=  0xFFFFFFFF  (Sum)
```

Equation 3.2.2: Propagating '0'.

In the case of the propagating ‘1’ all the circuits are kept in the “wait for Cin” state shown in Equation 3.2.1 by setting all ***A***-inputs to ‘1’ and all ***B***-inputs to ‘1’ except in the case of ***C00***, the least significant bit (LSB), where both ***A*** and ***B*** are ‘1’, causing a carry output to be generated and starting the carry propagation through the circuit as shown in Figure 3.2.3. The “wait for Cin”-state is also observable here, e.g. the ***C31*** output keeps its precharge value until it switches after about 18 ns.

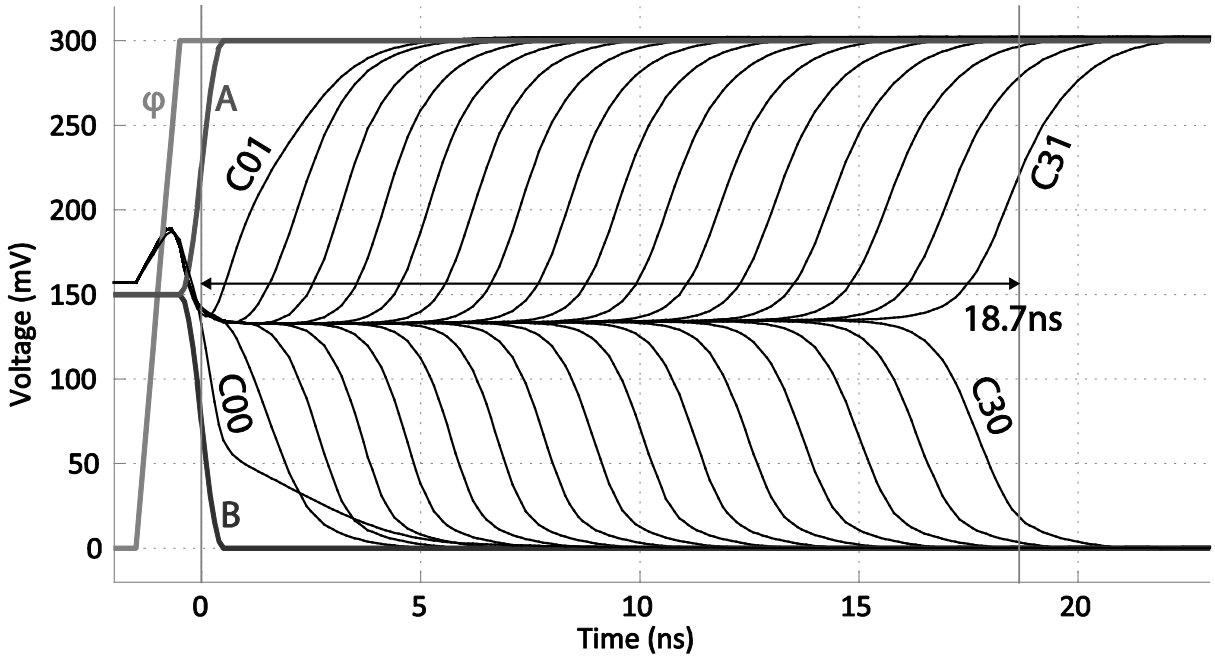

Figure 3.2.3: Simulation: SULVC propagating '1'.

In the propagating ‘0’ case, all A-inputs are ‘1’ and all B-inputs are ‘0’. This results in a carry propagation similar to the propagating ‘1’ but with opposite carry values, started by the LSB half-adder. The resulting carry outputs can be seen in Figure 3.2.4.

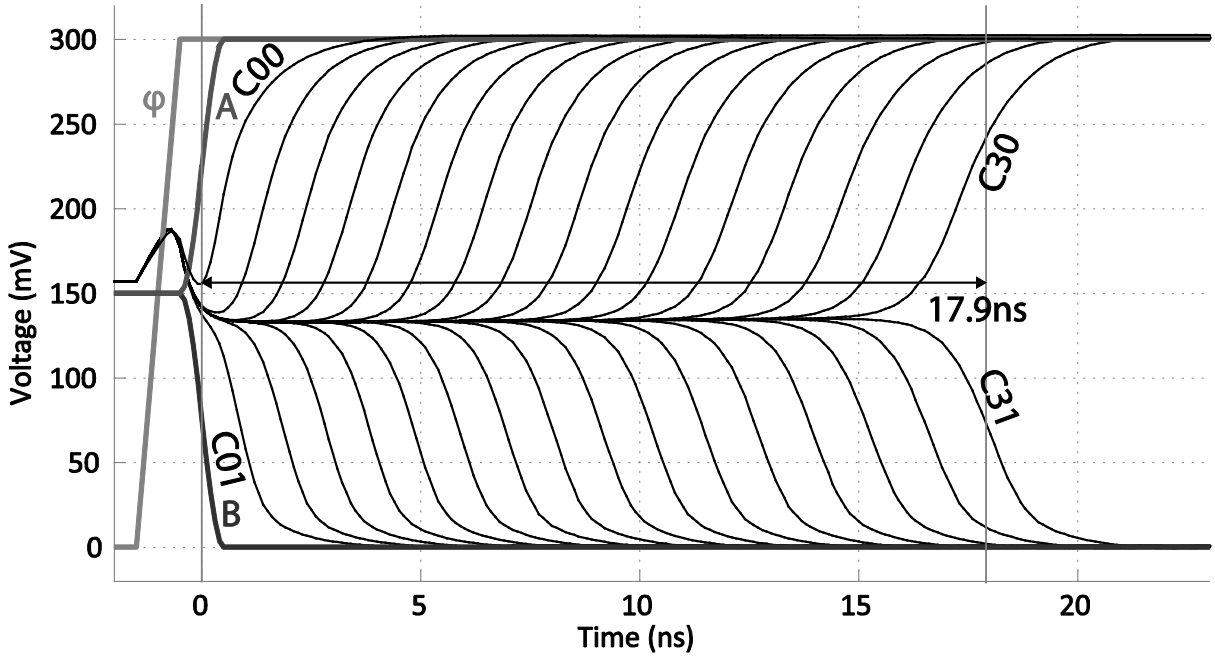

Figure 3.2.4: Simulation: SULVC propagating '0'.

The precharge to Vdd/2 nature of the SULVC circuit introduces some drawbacks to the topology. One is the high power consumption during the precharge phase caused by providing a low-resistance path between the clocks, φ and $\bar{\varphi}$ when both ***Ep2*** and ***En2*** are on (Figure 3.2.1) to achieve the correct precharge value, but this is partially made up for by the high speed of the circuit. The other drawback is its sensitivity to supply voltage changes and transistor sizing. A supply voltage change of more than ~40mV requires transistor resizing or reduction of carry

chain length to avoid output drifting during the evaluation phase, and also makes it less robust by being susceptible to process variations.

3.2.1 Performance

For the performance analysis of this circuit, the delay and power of the circuit is simulated to calculate the PDP and EDP. The results from these simulations are presented in Table 3.2.2. As the table shows, the PDP of the SULVC is 4.6 times higher than that of the conventional CMOS carry circuit at optimal supply voltage indicating that the circuit will use 4.6 times the energy to perform the same operation. The EDP of the SULVC is also higher, 4.9 times that of the conventional CMOS carry circuit running at optimal supply voltage.

	Std. CMOS	SULVC
Delay @ 280mV	31.65ns	0.947ns
Power @ 280mV	5.99nW	805nW
PDP @ 280mV	189.6aJ	763.3aJ
Optimal PDP	165.9aJ	
EDP @ 280mV	6.001yJs	0.7219yJs
Optimal EDP	0.1486yJs	

Table 3.2.2: Comparison chart for SULVC.

These results show that the SULVC is not a good choice when energy efficiency and speed are the only design concerns which will reduce the number of viable applications. However in systems where a low supply voltage is required it is still a worthy choice because of its superior speed in ultra-low voltage operation. With a supply voltage of 280mV the SULVC is 33 times[1] faster than the conventional CMOS circuit, resulting in an EDP eight times lower[2] than the conventional CMOS at this voltage despite the 805nW of power consumed.

Advantages	Disadvantages
• High speed at ultra-low voltages.	• Poor EDP and PDP performance. • Robustness issues.

[1] See Table 3.2.2, 31.65ns/0.947ns = 33.42
[2] See Table 3.2.2, 6.001yJs/0.7219yJs = 8.33

3.3 Capacitive Precharge NP domino carry

The capacitive precharge NP domino carry circuit, which will be named CPULVC, is a carry circuit based on the ULV5 logic style presented in [14].

The CPULVC is created in an effort to make a simpler and more reliable circuit by utilizing capacitive division to implement the carry logic. The idea is to let the capacitive division between three equal size capacitors handle the logic of the circuit as shown in Table 3.3.1(page 46).

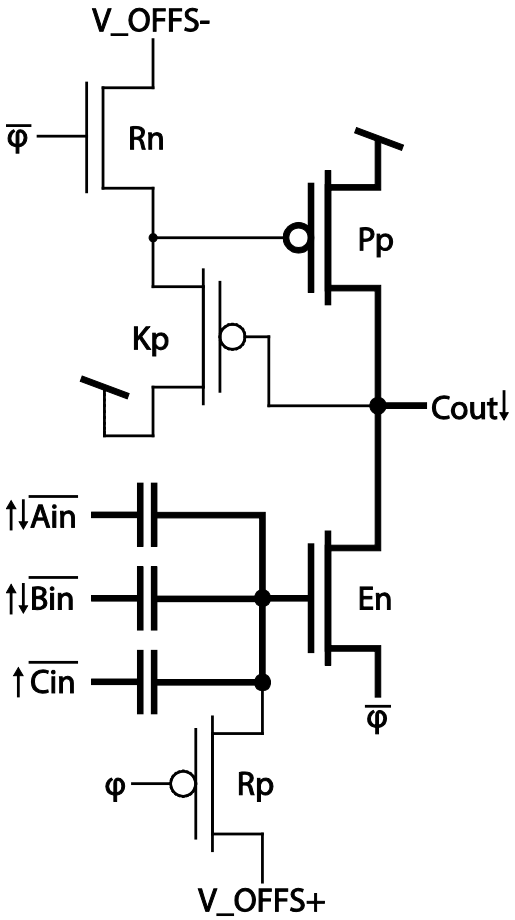

Figure 3.3.1: Schematic: ULV5 carry N-type circuit.

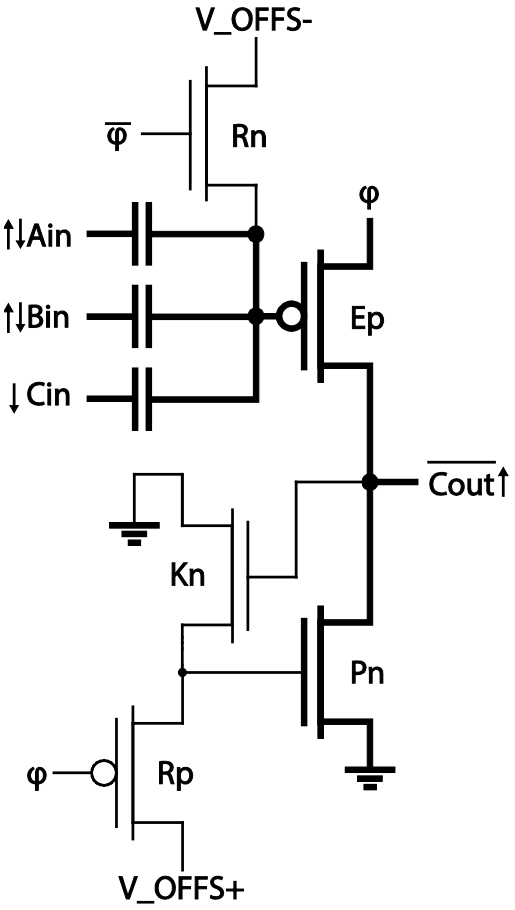

Figure 3.3.2: Schematic: ULV5 carry P-type circuit.

The ***A*** and ***B*** input needs to be either a rising or a falling edge signal generated by a level-to-edge converter to get a full transition both ways, or an output from a logic style like the ULV2[5] providing a transition of Vdd/2 both ways. A level-to-edge converter suitable for this purpose is proposed in section 3.4.2.1.

If both $\overline{\boldsymbol{A}}$ and $\overline{\boldsymbol{B}}$ in the N-type circuit are rising edge signals, a carry output transition is triggered instantly due to a voltage increase of ~2/3Vdd on the ***En*** floating gate (Figure 3.3.1) making it stronger than the ***Pp*** precharge transistor and independent of the carry signal. The output is also independent of the carry input when both $\overline{\boldsymbol{A}}$ and $\overline{\boldsymbol{B}}$ are falling and thereby reducing the voltage on the floating gate by ~2/3Vdd. This turns off the ***En*** transistor and makes the ~1/3Vdd voltage rise ***Cin*** is able to produce insignificant. For the case where $\overline{\boldsymbol{A}}$ and $\overline{\boldsymbol{B}}$ are contrasting, the two signals will cancel each other and leave the circuit working as a

ULV5 inverter with the $\overline{Cin}$ signal as input, although the gate delay is slightly higher because the capacitive division will reduce the floating gate voltage swing.

Carry truth table				N-type signals				P-type signals			
A	B	Cin	Cout	$\bar{A}$	$\bar{B}$	$\overline{Cin}$	Cout	A	B	Cin	Cout'
0	0	0	0	↑	↑	↑	↓	↓	↓	↓	↑
0	0	1	0	↑	↑	0	↓	↓	↓	1	↑
0	1	0	0	↑	↓	↑	↓	↓	↑	↓	↑
0	1	1	1	↑	↓	0	1	↓	↑	1	0
1	0	0	0	↓	↑	↑	↓	↑	↓	↓	↑
1	0	1	1	↓	↑	0	1	↑	↓	1	0
1	1	0	1	↓	↓	↑	1	↑	↑	↓	0
1	1	1	1	↓	↓	0	1	↑	↑	1	0

Table 3.3.1: Precharge NP domino capacitive divider carry logic.

The P-type circuit is designed in a manner complementary to the N-type, still using capacitive division to achieve the desired logic result but using a precharge level of '0' and keeping the output unchanged in the evaluation phase in the case of a carry and producing a rising edge in the case of no carry. Hence the signal produced by the P-type circuit is in the form needed for the N-type circuit as the N-type output is in the form needed for the P-type circuit. This is shown in Table 3.3.1.

The circuit has been scaled for a 300mV supply voltage and the strength of the PMOS transistor has only been increased using fingers to reduce the impact of changes to the Vdd. With a supply voltage of 300mV the circuit is able to process 64 bits in one clock cycle as shown in Figure 3.3.3 despite problems with output drifting that can be seen in both Figure 3.3.3 and Figure 3.3.4, which is an attribute commonly associated with dynamic logic.

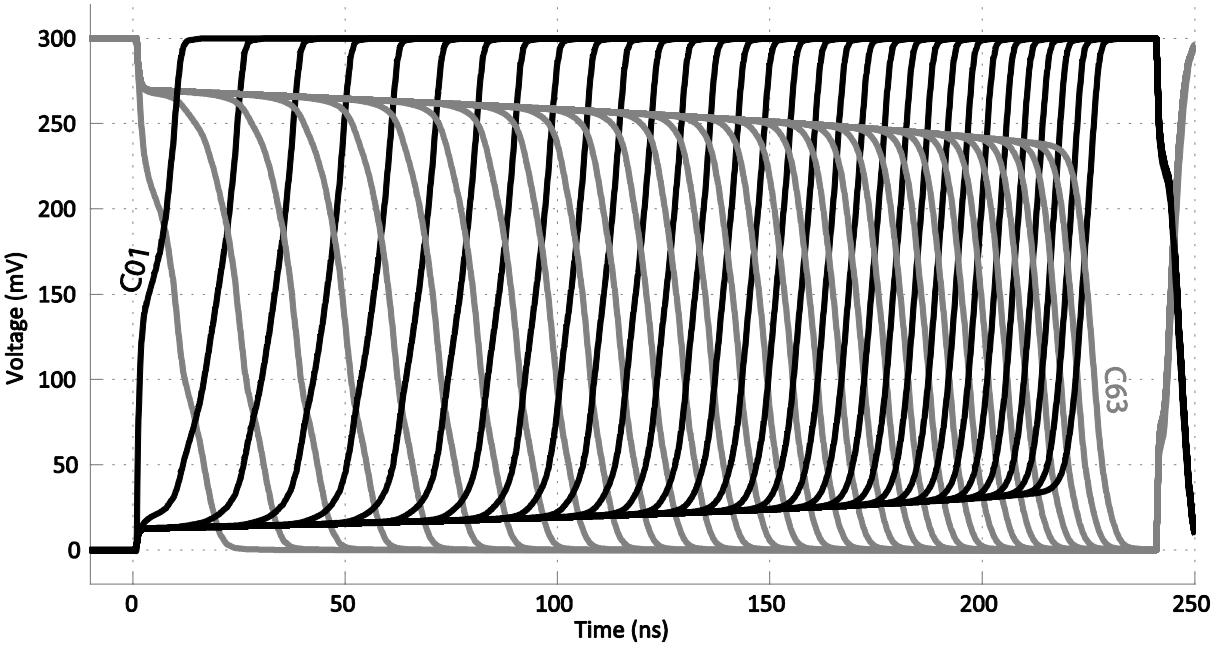

Figure 3.3.3: Simulation: 64 bit ULV5 n-p domino carry chain, propagating '0'.

The propagating ‘0’ simulation in Figure 3.3.3 is based on the conditions in Equation 3.2.2 on page 42. Accordingly the simulation conditions in Figure 3.3.4 are the same as in Equation 3.2.1.

To increase stability the number of bits per clock cycle can be reduced together with a higher clock frequency. The cost of this is a slight reduction of average speed per bit, caused by flip-flop setup time overhead and slower carry propagation for the LSBs.

The CPULVC has been analyzed and compared to the traditional CMOS carry circuit in terms of delay, power, PDP and EDP. The results are presented in Table 3.3.2 and shows that the CPULVC is 6 times faster than the standard CMOS at 300mV achieved without increasing the power consumption more than two times. The PDP of the SULVC is therefore 3 times better at 300mV but the SULVC also has a 2.6 times better PDP than the conventional CMOS carry circuit running at PDP-optimal supply voltage.

Because of the good delay characteristics of the SULVC at low voltage the EDP is 18 times higher than traditional CMOS running at 300mV. However the traditional carry circuit is able to achieve a 40% lower EDP at its EDP-optimal supply voltage.

	Std. CMOS	CPULVC
Delay @ 300mV	20.5ns	3.5ns
Power @ 300mV	10.4nW	19.26nW
PDP @ 300mV	212.3aJ	64.4aJ
Optimal PDP	165.9aJ	
EDP @ 300mV	4.35yJs	0.236yJs
Optimal EDP	0.1486yJs	

Table 3.3.2: CPULVC performance analysis.

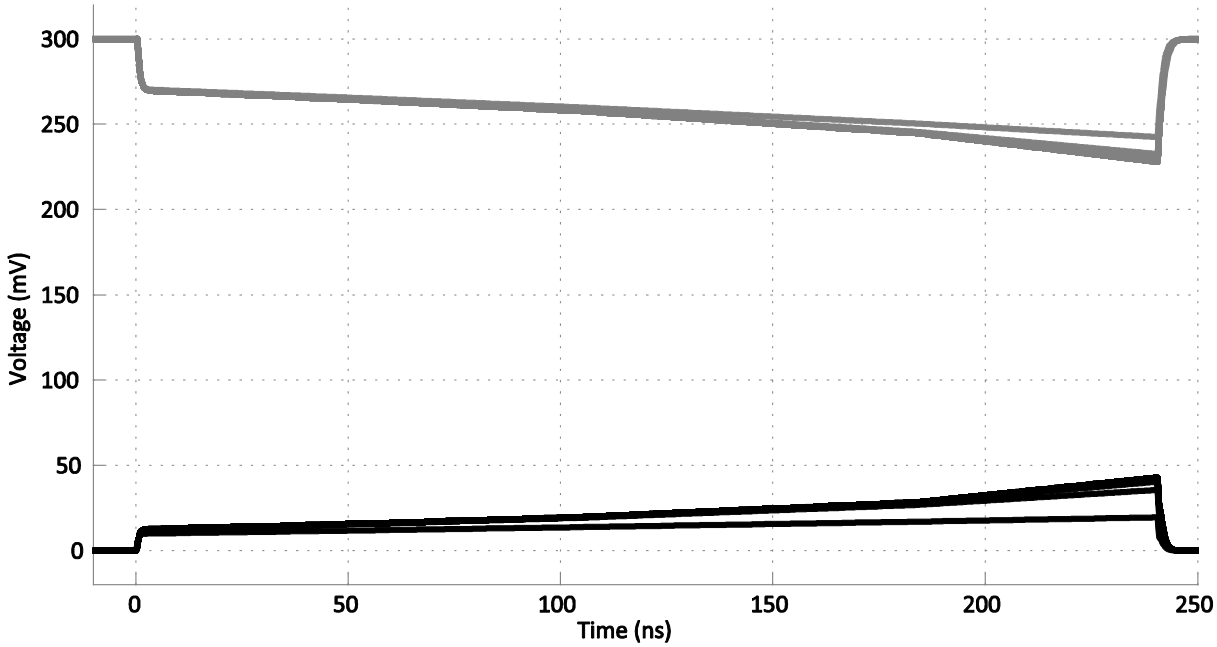

Figure 3.3.4: Simulation: 64 bit ULV5 n-p domino carry chain, “propagating” ‘1’.

The CPULVC can therefore be a good choice for systems where it is important to get as much data as possible processed for a certain amount of energy because of its low PDP. It is also a better choice than the conventional CMOS carry circuit, both EDP and PDP wise in low-voltage systems.

Advantages	Disadvantages
• Good PDP. • EDP ok for low-voltage domain.	• Limited logic depth due to drifting.

Experiments using 4 input capacitors with two chains in parallel to create a partially differential topology were also conducted yielding marginally better stability at the cost of a substantial decrease in delay due to the extra capacitor in the capacitive division on the floating gate. A fully differential version has also been developed in collaboration with another master student.

3.4 Ultra-Low Voltage Pass Transistor Logic

Pass transistor logic (PTL) is a logic style where transistors are used as switches that open or close a path between the input and output. So instead of opening and closing a path between the output and Vdd or ground like in conventional CMOS logic, the output is either driven by an input signal or not driven by the gate at all.

In the following section this principle will be applied with ULV floating gate logic to increase the PTL speed at ultra-low voltages.

3.4.1 Single pass transistors

A single PTL gate is either based on an NMOS or a PMOS transistor as shown in Figure 3.4.1 and Figure 3.4.2 respectively.

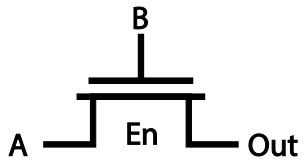

Figure 3.4.1: NMOS as PTL gate.

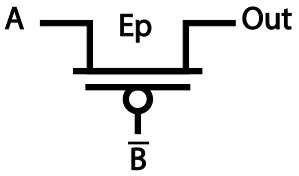

Figure 3.4.2: PMOS as PTL gate.

A characteristic attribute of PTL logic is that it has a tri-state output. This is because the gate can "disconnect" the output and thereby introduce a third, high-impedance, state 'Z' in addition to the '1' and '0' state. The truth table of the single pass transistor gate, showing that the output equals ***A*** when B is '1' and that does it not drive the output when B is '0', is presented below.

A	B	Out
0	0	Z
0	1	0
1	0	Z
1	1	1

Table 3.4.1: Truth table for a single PTL gate.

The ULV pass transistor logic style presented in [16] uses floating gates to increase the drive strength and speed of the PTL in low-voltage design. This is done by introducing a floating gate with a precharge transistor in the same way as this is done in the original ULV logic styles. The schematics of the ULVPTL gates are shown below, the NMOS equivalent in Figure 3.4.3 and the PMOS equivalent in Figure 3.4.4.

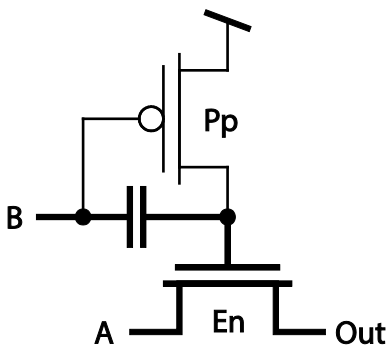

Figure 3.4.3: Schematic: ULV N-type pass transistor

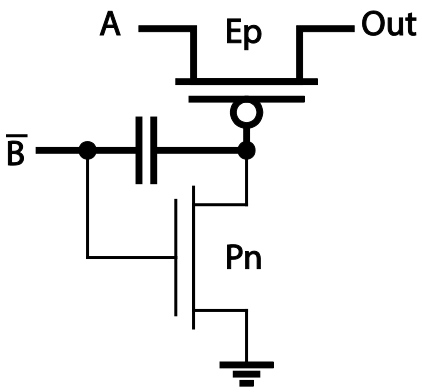

Figure 3.4.4: Schematic: ULV P-type pass transistor

The ULV PTL logic style utilizes the concept of super-charging a floating transistor gate through capacitive coupling as explained in section 2.1. The operation of the NMOS version in Figure 3.4.3 will be explained first; when the ***B*** input is '0', the floating node of ***En*** is charged to Vdd making the ***En*** transistor in relative terms semi-conducting. Instead of Vdd, an offset voltage can also be applied to adjust the transistor strength and current in this semi-conducting state. When input ***B*** rises, the ***Pp*** transistor is turned off while the floating node of the ***En*** transistor is charged to a voltage level higher than Vdd through the capacitor. This results in a highly conductive ***En*** transistor and low gate delay. The PMOS version works in the same way, only with the NMOS and PMOS transistor switched as shown in Figure 3.4.4 making it better at conducting high ***A*** signals.

The difference in propagation delay between the N-type and the P-type circuit is significant for both input. This is shown in Figure 3.4.5 for a falling ***A***-input and shows the poor pull-down properties of the P-type circuit, the corresponding poor pull-up properties of the N-type can be seen in Figure 3.4.6. The slower conventional PTL is also shown in these figures. For high ***A*** and ***Out*** values this is because a PMOS transistor will have a large difference between its low gate voltage and the high source voltage while the NMOS will have a smaller difference because of its higher gate voltage resulting in less current through the NMOS than the PMOS transistor. For low ***A*** and ***Out*** values the same effect will make the NMOS stronger than the PMOS.

For this reason PTL gates are often placed in parallel like in Figure 3.4.7 and are in this configuration called a transmission gate (TG). These are used when both high and low input values are expected. However in some cases like in domino logic where a precharge level is set before evaluation, only one of the gates is needed to pull the output in a predefined direction or leave it at the precharged level.

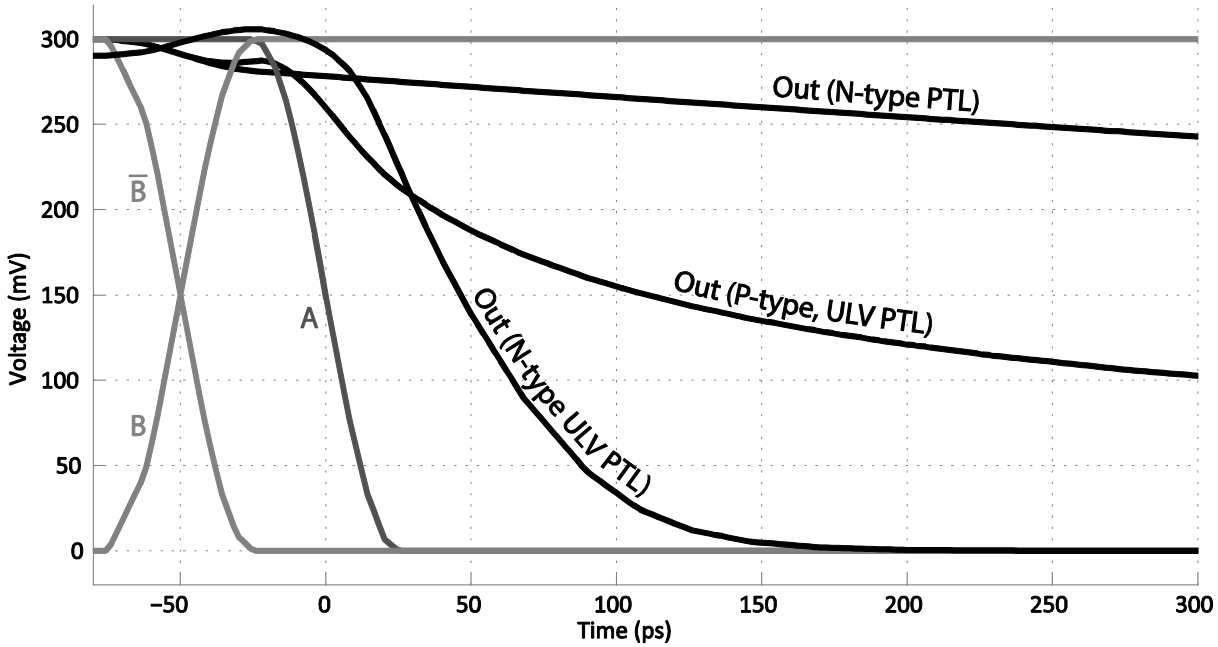

Figure 3.4.5: Simple n- and p-type ULV pass transistors, falling *A* (Ideal inputs).

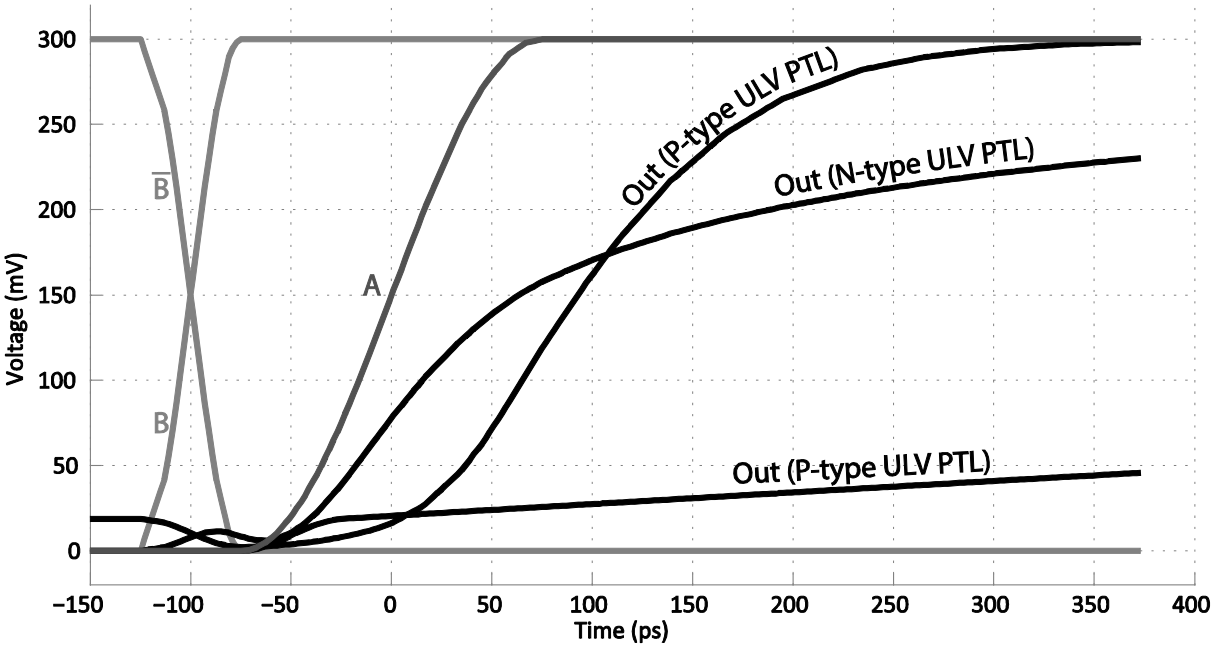

Figure 3.4.6: Simple n- and p-type ULV pass transistors, rising *A* (Ideal inputs).

The simulations of the ULV PTL and the conventional pass transistor logic show that the super-charged floating gate increases the speed of the ULVPTL significantly due its higher drive-strength. Both the N-type and the P-type ULV pass transistors prove to be about 25 times faster than the conventional PTL at 300mV. The exact numbers are presented in Table 3.4.2.

Circuit	PTL	ULVPTL	Diff.
N-Type	1.276ns	46.30ps	27.6x
P-Type	2.259ns	92.95ps	24.3x

Table 3.4.2: Propagation delay for minimum sized PTL and ULVPTL (Vdd=300mV, Load=1fF).

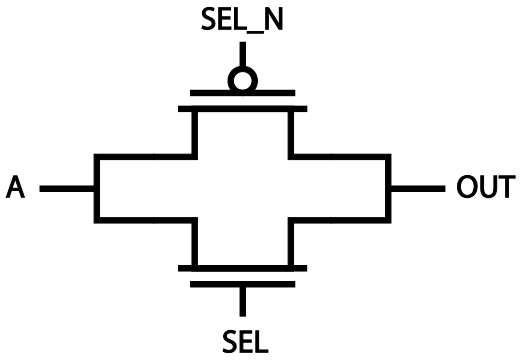

Figure 3.4.7: Schematic: PTL Transmission Gate.

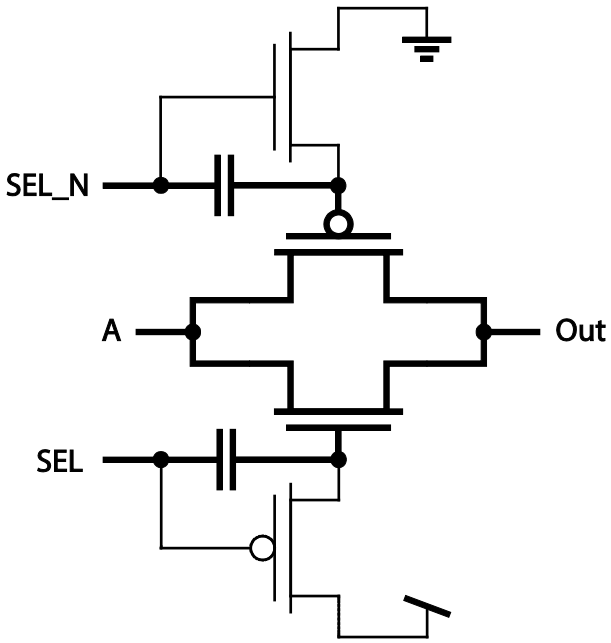

Figure 3.4.8: Schematic: ULV PTL Transmission Gate

3.4.2 Ultra-Low Voltage Pass Transistor Multiplexer

A multiplexer (MUX) is a circuit that selects one of its inputs to pass through to the output based on an address or select signal. The schematic symbol for a 2-to-1 multiplexer is shown in Figure 3.4.9 and the conventional PTL transistor level schematic is shown in Figure 3.1.11 where ***A*** is let through if the one-bit ***SEL*** address is '0' and ***B*** is let through if ***SEL*** is '1'. The "let through" nature of these circuits makes the multiplexer one of the most common uses of pass transistor logic (PTL) and the MUX is therefore a good circuit to start with when exploring ULV PTL logic.

Because the N-type pass transistor is good at passing '0' signals but not as good at passing '1' signals and the P-type is complimentary good at '1' and not '0', the two are often put in parallel to pass both '0' and '1' well. When the two are placed in parallel they make up a transmission gate (TG) as previously mentioned. A conventional PTL TG is shown in Figure 3.4.7 and the new ULV PTL equivalent is shown in Figure 3.4.8. These can be used as a tri-state buffer or as a building block in larger systems like the multiplexer. An implementation of a 2-to-1 multiplexer made with two conventional transmission gates is shown in Figure 3.4.11, and an implementation using ULV transmission gates is presented in Figure 3.4.10. Both are scaled for ultra low voltage operation by using 3-fingered PMOS transistors.

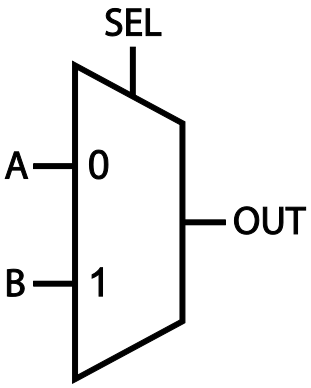

Figure 3.4.9: 2-to-1 MUX

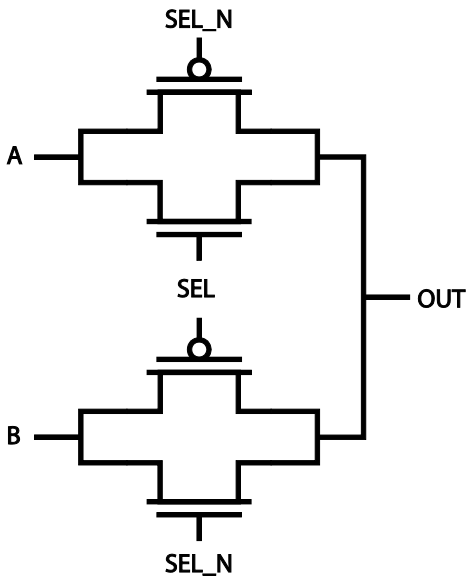

Figure 3.4.11: Schematic: Standard TG mux.

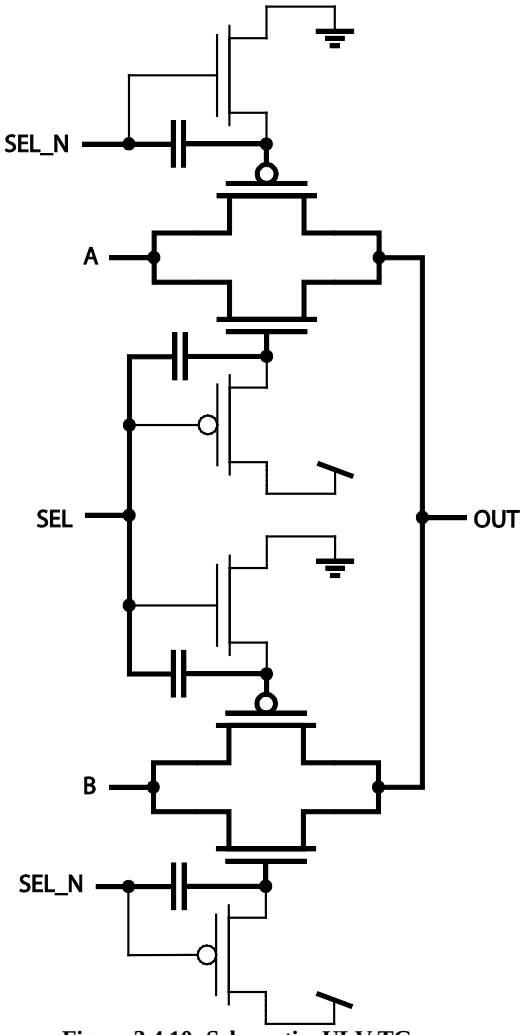

Figure 3.4.10: Schematic: ULV TG mux

According to the simulations run in Figure 3.4.12 and Figure 3.4.13 the ULV TG MUX is significantly faster than the traditional TG MUX. The simulated worst-case propagation delay for the ULV TG MUX is only 77.6ps for a load of 1fF at 300mV. This makes the ULV PTL multiplexer 40 times faster than the standard TG MUX when operating at 300mV and makes it a good choice for low voltage applications.

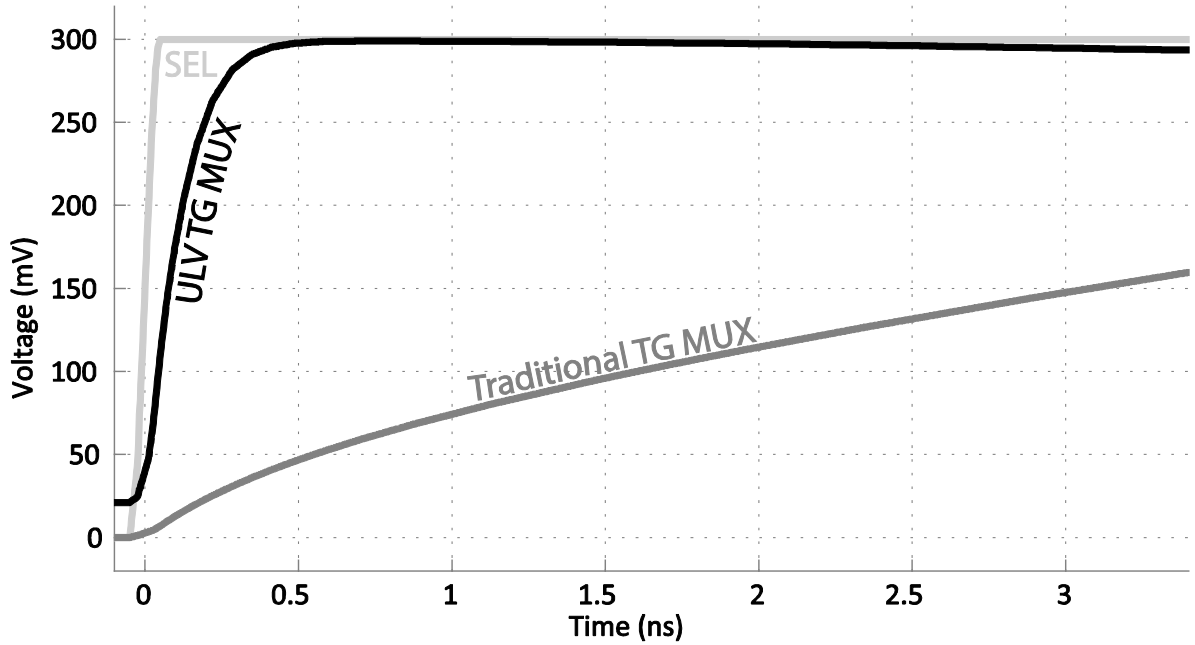

Figure 3.4.12: Simulation: Standard and ULV TG MUX; A=0, B=1, SEL=↑

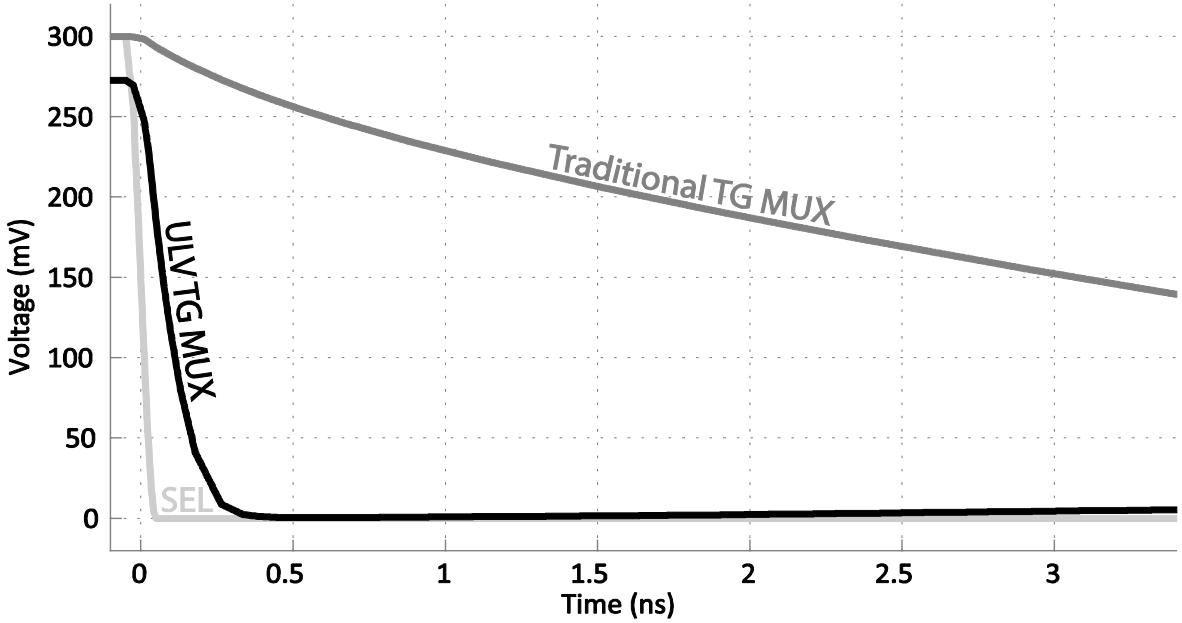

Figure 3.4.13: Simulation: Standard and ULV TG MUX; A=0, B=1, SEL=↓

(A=0, B=1)@300mV	SEL	ULV TG MUX	TG MUX	Diff
Rising transition	↑	77.56ps	3.056ns	39x
Falling transition	↓	72.36ps	3.069ns	42x
Worst case		77.56ps	3.069ns	40x

Table 3.4.3: MUX propagation delay at 300mV, (A=0, B=1).

3.4.2.1 Level-to-Edge Converter

The mux is also useful for the CPULV circuit designed in section 3.3 because it can be used to make a level-to-edge converter. This can be done by setting the clock as the select signal, ***B*** as the input level and ***A*** as the boolean inversion of the input level. An illustration of this concept is presented in Figure 3.4.14. The circuit will make the output switch from the inverted input level when the clock signal is '0' to the actual input level when the clock signal rises to '1' resulting in a rising edge output for an input level of '1' and a falling edge output for an input level of '0'. The ULV PTL implementation will have the analog properties of the ULV PTL MUX simulated in Figure 3.4.12 and Figure 3.4.13 but requires both the ***A*** and $\overline{\boldsymbol{A}}$ input level to be ready before the arrival of the ***CLK*** rising edge.

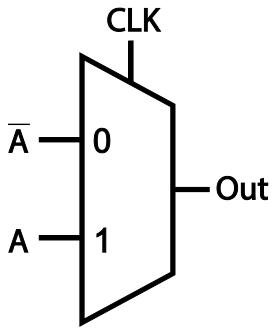

Figure 3.4.14: MUX based level-to-edge converter.

3.4.3 Ultra-Low Voltage Pass Transistor Carry Gate

The following carry circuit design is based on the ULV pass transistor logic presented in [3] and is a modification of the carry circuit presented in [17]. It is a domino logic style which means that both an N-type and a P-type circuit needs to be designed. The first design presented is the N-type circuit.

3.4.3.1 N-type carry circuit

The N-type carry circuit is presented in Figure 3.4.15, the idea is to precharge the output to '1' through the ***Pp*** transistor during the precharge phase and pull it down if needed through a ULV pass transistor during the evaluation phase. The circuit is designed so that the carry input signal does not drive the output. This is done to increase the speed and robustness so that one signal is not driving several bits, and so that all the inputs used to drive the output are parallel signals generated independently for each carry bit, i.e. the ***A*** and ***B*** inputs. The truth table for the circuit and required inputs is shown in Table 3.4.4.

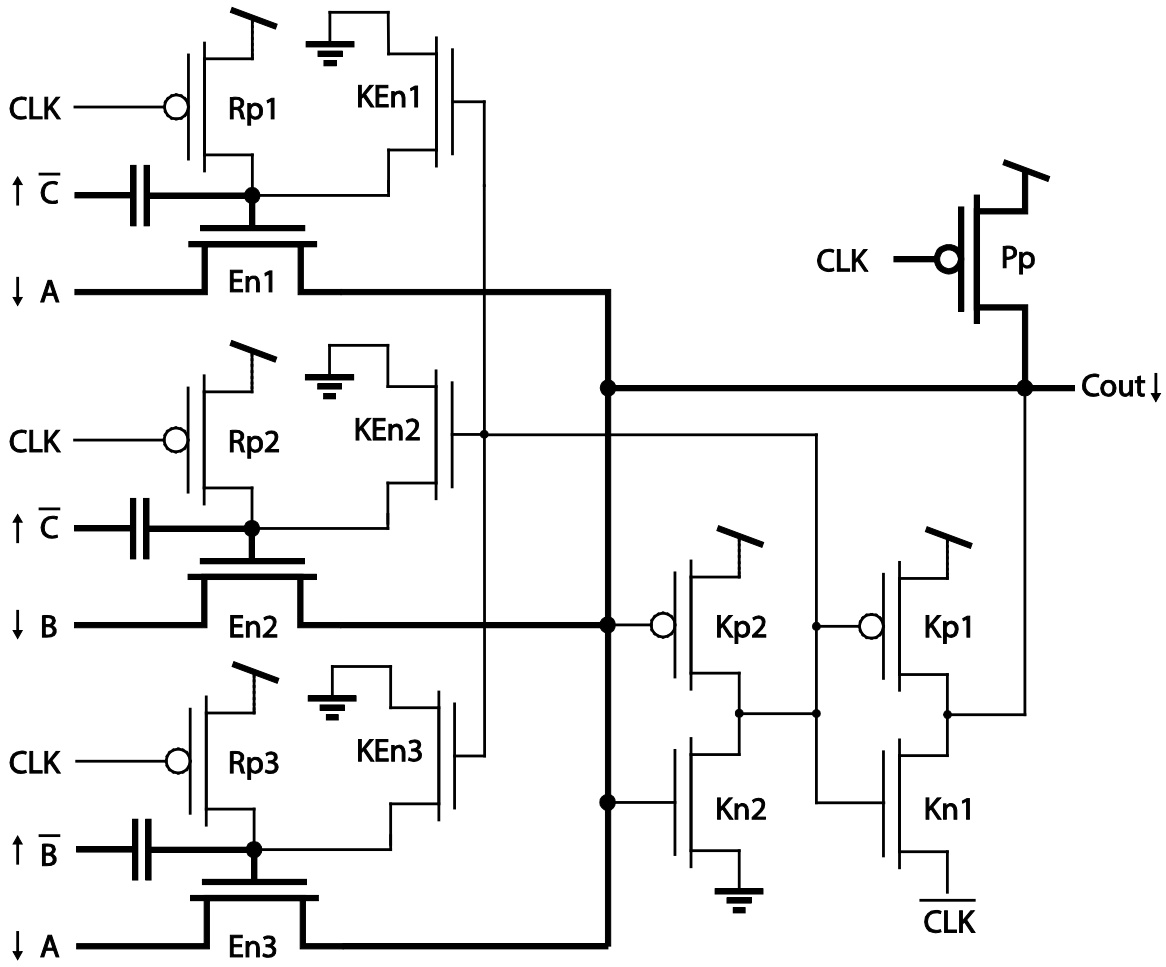

Figure 3.4.15: Schematic: N-type ULV PTL Carry circuit.

The transistors used in this circuit and their purposes are the following:

The ***Pp*** transistor is used to precharge the output to '1' during the precharge phase.

The ***En*** transistors are used to pull the output low if the ***A*** and/or ***B*** signal is low while a rising edge is generated on $\overline{\boldsymbol{B}}$ and/or $\overline{\boldsymbol{C}}$, increasing the ***En*** floating gate voltage.

The ***Rp*** transistors are used to charge the gates of the ***En*** transistors during precharge and are turned off during the evaluation phase to achieve the desired floating gate effects.

The ***KEn*** transistors are used to reduce the power consumption and increase noise margin by turning off the evaluation transistors when the circuit switches. This in turn will stop high inputs from pulling on the output during switching and denying a full output swing. This is important because the circuit is designed based on the condition that the ***En*** transistors pass '0' values better than '1' values, but when ***Cout*** is sinking, the difference between the floating gate and the output increases, resulting in a higher gate-source voltage and a stronger ***En*** transistor for high inputs.

The ***Kn*** and ***Kp*** transistors are used to enhance the output value. The ***Kn2*** and ***Kp2*** transistors turn on the ***KEn*** transistors when needed, the ***Kp1*** transistor keeps the circuit static by keeping the output high when the circuit has not switched and ***Kn1*** pulls the output all the way down after the ***KEn*** transistors have discharged the ***En*** gates and keeps the low output static. The source terminal of the ***Kn1*** transistor is connected to $\overline{\boldsymbol{CLK}}$ to reduce the current needed during precharge and thereby reduce the power consumption and size of the ***Pp*** precharge transistor.

A	B	C	Cout	$\overline{B}$	$\overline{C}$	Cout
0	0	0	0	↑	↑	↓
0	0	1	0	↑	0	↓
0	1	0	0	0	↑	↓
0	1	1	1	0	0	1
1	0	0	0	↑	↑	↓
1	0	1	1	↑	0	1
1	1	0	1	0	↑	1
1	1	1	1	0	0	1

Table 3.4.4: Truth table and generated signals for the N-type ULV PTL Carry.

The transient simulations in Figure 3.4.16 show that the circuit struggles to pull the output low when ***A*** is '1' and ***B*** = ***C*** = '0'. This is because $\overline{\boldsymbol{C}}$ is super-charging the ***En1*** and ***En2*** floating gates and turning both transistors on. The reason it is pulled down at all is that the high voltage on the ***A***-input reduces the gate-source voltage and weakens the ***En1*** transistor. A problem arises when the output voltage drops because it makes the difference in strength between the pull-up and the pull-down transistor smaller and would equalize when reaching half the supply voltage if it were not for the ***KEn*** and ***Kn*** transistors. Developing a fully dynamic high-speed ULV circuit might reduce this effect by having weak pull-up capabilities and thereby give the '0' a higher priority than the ULV7 inverter does. Altering the circuit to tackle the encountered switching issues seems to be challenging without increasing the delay and introducing an area penalty through extra transistors or sacrifice noise margin and robustness for non-switching outputs.

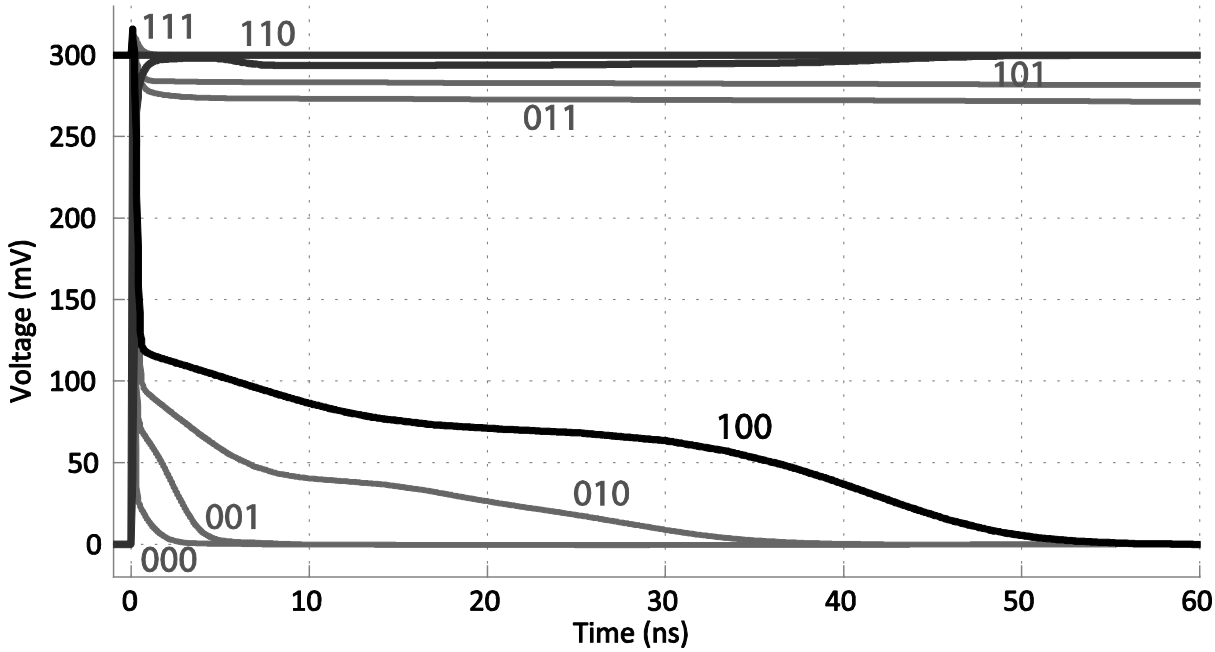

Figure 3.4.16: Transient evaluation phase simulation of the N-type ULV PTL carry circuit.

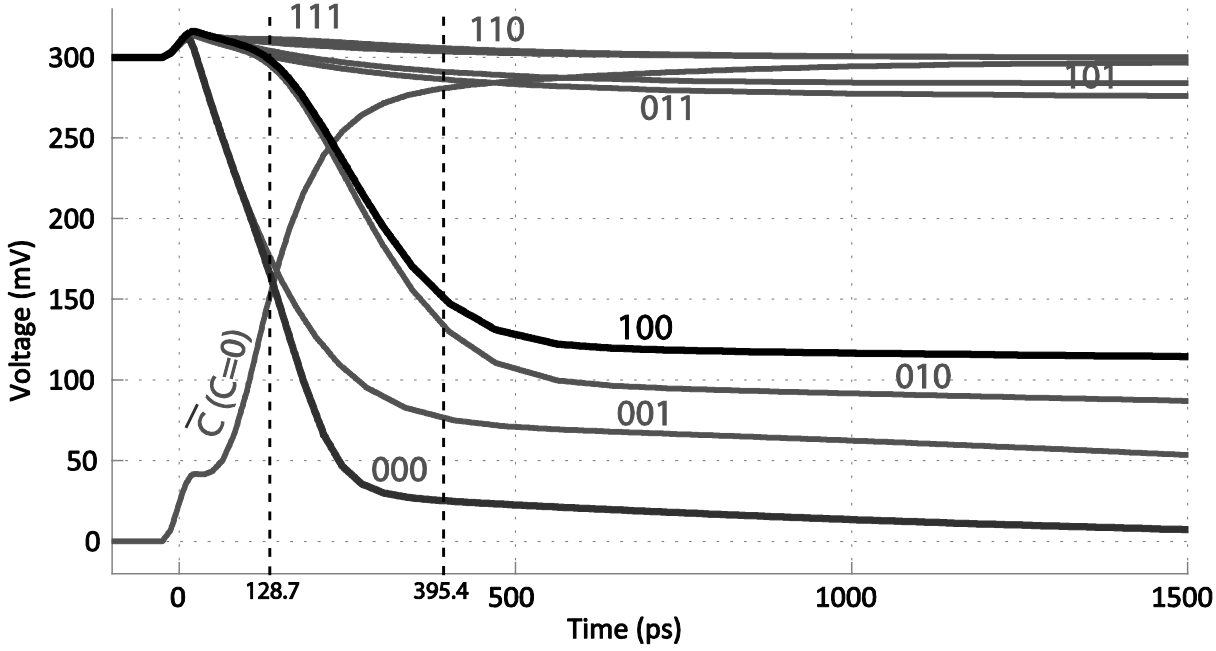

Figure 3.4.17: Close-up of the beginning of the evaluation phase from Figure 3.4.16.

In Figure 3.4.17 a transient simulation of the N-type ULV PTL carry gate is performed, showing the different outputs generated from all 8 possible input combinations named after the input values in the order: A, B, C. The worst case propagation delay from these simulations is in the case where ***A***='1' and ***B***=***C***='0' and yields a worst case propagation delay for the N-type ULV PTL carry circuit of 266.7ps at 300mV. This is significantly lower than the 20.5ns delay achieved by the traditional CMOS carry circuit at the same voltage in section 2.4.1, so if the pull-down issues are ignored this means that the ULV PTL N-type carry gate is 77 times faster than the conventional carry gate at 300mV.

In the ULV PTL logic style, most of the power is consumed through the inputs. The consumed power also varies largely for the different input combinations ranging from 28nW for “110” to 740nW for the “100” case. The power is therefore calculated as the average of all 8 scenarios assuming that the probability of each input combination is the same. The performed calculations show that the average power consumption of the ULV PTL carry gate is 354nW, which is 34 times that of the conventional CMOS carry gate resulting in a PDP less than half of that achieved by the conventional CMOS carry circuit at 300mV and 40% lower than its optimal PDP. The low gate delay however results in an EDP of just 0.025yJs, which is 174 times lower than the EDP performance of conventional CMOS at 300mV. The EDP of the ULV PTL carry gate is even 6 times lower than the conventional carry gate running at EDP optimal supply voltage, making it a better carry circuit choice, regardless of energy and supply voltage constraints and limitations.

	Std. CMOS	N-type ULV PTL
Delay @ 300mV	20.5ns	266.7ps
Power @ 300mV	10.4nW	354nW
PDP @ 300mV	212.3aJ	94aJ
Optimal PDP	165.9aJ	
EDP @ 300mV	4.35yJs	0.025yJs
Optimal EDP	0.1486yJs	

Table 3.4.5: ULV PTL N-type carry performance analysis.

3.4.3.2 P-type carry circuit

The P-type ULV PTL carry circuit is the complementary circuit to the N-type ULV PTL carry and works on the same principles. The circuit is shown in Figure 3.4.18 and the roles of the transistors are the same as for the N-type circuit previously explained although the signal- and transistor types are changed.

Transient simulations of the ULV PTL carry P-type circuit are shown in Figure 3.4.19 and Figure 3.4.20. The issues with output switching experienced for the N-type circuit also applies to the P-type carry circuit for the same input combinations in a similar way and for the same reasons as for the N-type circuit. The effect on the P-type circuit is shown in the simulations in Figure 3.4.19.

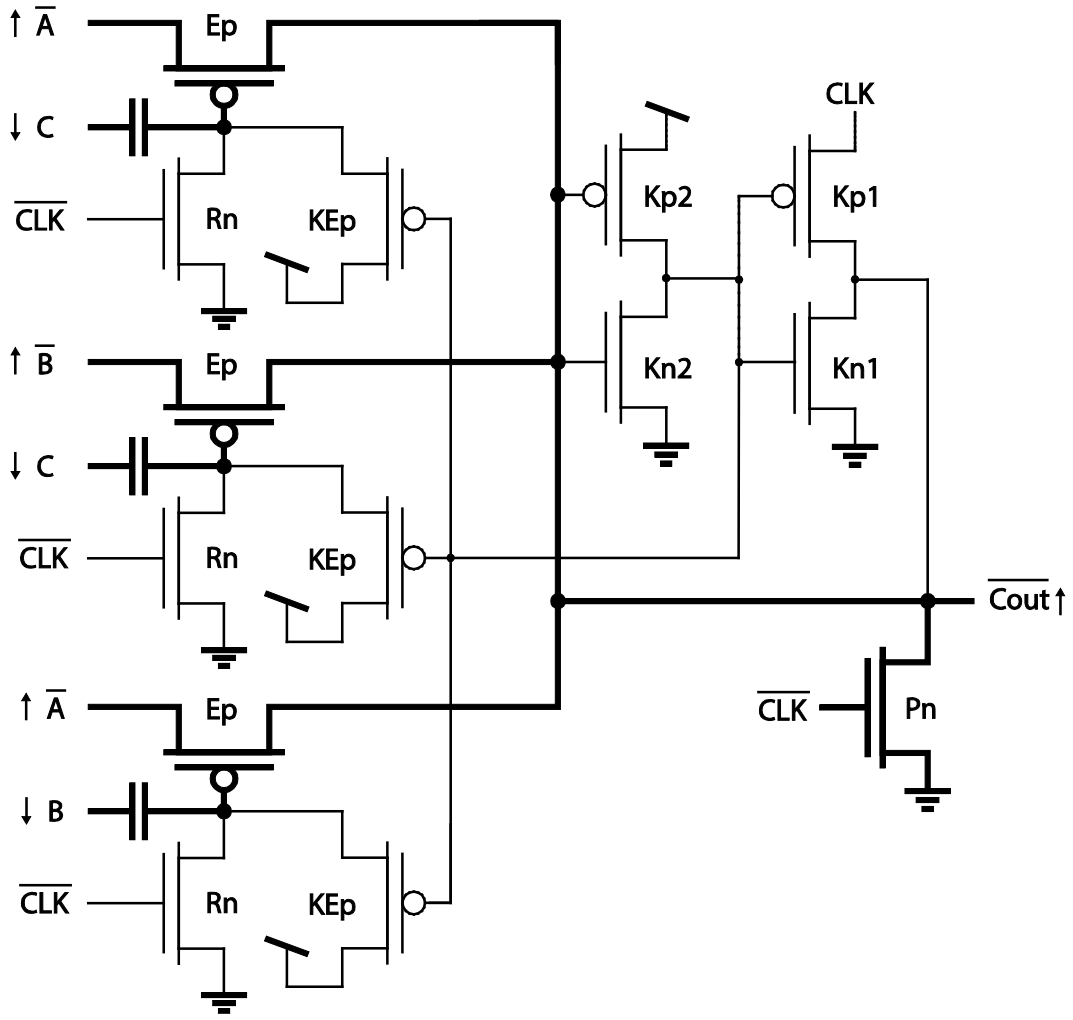

Figure 3.4.18: Schematic: P-type ULV PTL carry.

The propagation delay achieved by the ULV PTL P-type carry circuit is slightly higher than for the N-type. This is most likely due to more parasitic capacitance caused by having more fingers on the PMOS transistors reducing the floating gate voltage swing and thereby reducing the transistor strength slightly. The power consumption of the P-type ULV PTL carry circuit is 20% lower than that of the N-type circuit, resulting in even lower PDP and EDP figures. The numbers are presented in Table 3.4.6 and show that the ULV PTL P-type carry circuit is a fast and energy efficient circuit with a performance in the same range as the N-type circuit.

	Std. CMOS	P-type ULV PTL
Delay @ 300mV	20.5ns	280.3ps
Power @ 300mV	10.4nW	275nW
PDP @ 300mV	212.3aJ	77aJ
Optimal PDP	165.9aJ	
EDP @ 300mV	4.35yJs	0.022yJs
Optimal EDP	0.1486yJs	

Table 3.4.6: P-type ULV PTL performance analysis.

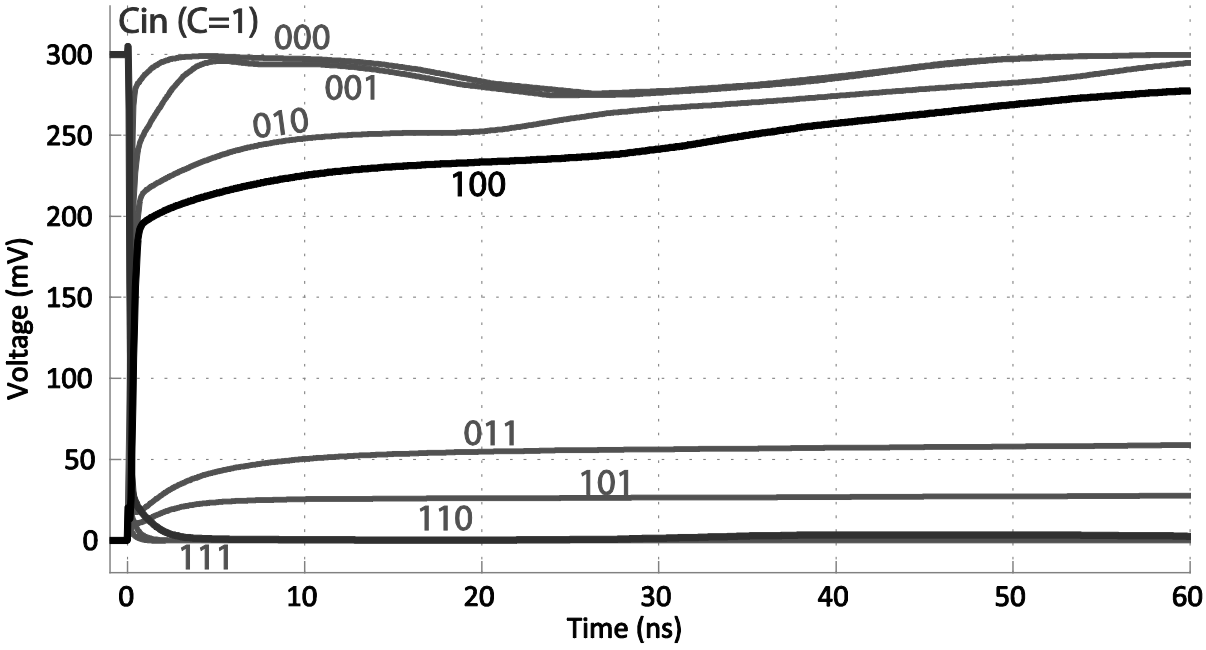

Figure 3.4.19: Transient evaluation phase simulation of the P-type ULV PTL carry circuit.

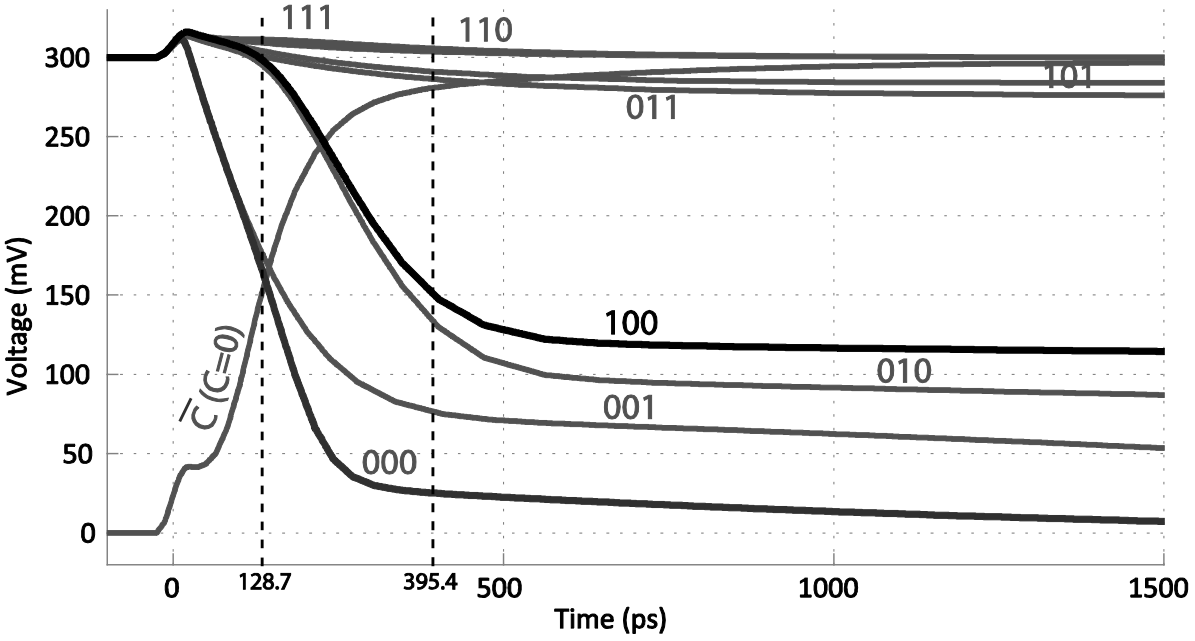

Figure 3.4.20: Close-up of the beginning of the evaluation phase from Figure 3.4.19

3.4.3.3 Carry chain

In this section the N- and P-type carry gates are connected in a series configuration to analyze the performance of the topology in a carry chain. Connecting long carry chains seem to quickly impact the circuit worst case performance due to the pull-down and pull-up issues previously described. In Table 3.4.7, a 2-bit carry cell is simulated and shows that replacing the ULV7 inverter with the ULV PTL N-type carry circuit on the input of the P-type ULV PTL carry circuit has a drastic impact on the performance of the circuit. The circuit propagation delay per bit more than doubles making the total delay for the 2-bit cell 1.22ns. Even though the power consumption of the 2-bit cell is lower than the sum of the two carry bits with ULV7 inverters on the carry input, the PDP and especially the EDP increases significantly. Even though the EDP is still lower for the 2-bit "chain" than for the conventional CMOS, this circuit does not seem to function optimally in a carry chain in its current form. The performance is even further reduced when more links are introduced in the chain.

Although the distortions seen in these simulations are suspiciously similar to the model errors in the layout simulations of the ULV7 chip, this is not expected to be the reason in this case because it is simulated on a schematic level that previously proved to be within $2 \times \sigma$ of the measured results and the effect is likely to have the reasonable explanation given in section 3.4.3.1.

To conclude the findings for the ULV PTL carry circuit topology, the circuit is still not suited for daisy chaining in adders in its current form, but could contribute with significant performance increases if the switching issues are solved.

WC: A0=A1='1', B0=B1=C0='0'	**ULV PTL**			**CMOS**	
	N-type	P-type	Chain avg.	@optimal	@300mV
Worst case delay	266.7ps	280.3ps	610ps	-	20.5ns
Average power consumption	354nW	275nW	288nW	-	10.4nW
PDP	94aJ	77aJ	175aJ	166aJ	212.3aJ
EDP	0.025yJs	0.022yJs	0.11yJs	0.15yJs	4.35yJs

Table 3.4.7: ULV PTL P-type carry performance analysis.

Advantages	**Disadvantages**
• Good PDP. • Very good EDP.	• Significant performance reduction in serial connections.

4 Conclusion

The goal of this thesis was to explore and further develop the field of floating gate ultra-low-voltage electronics with a focus on speed and energy efficiency.

Through this work a chip demonstrating the high-speed properties of the ULV logic in hardware was designed, produced and measured. The design introduced the concept of designing circuits scaled for direct measurements of a single gate's analog properties to establish a good foundation for further development of physical ULV circuits. The measured results presented show that the theorized and simulated high-speed properties of the ULV logic are highly applicable to physical hardware implementations of these circuits, and thereby adding credibility to previously proposed ULV logic.

Two new carry circuits and a multiplexer are also proposed, simulated and discussed. These show good high-speed and energy efficient properties for ultra-low voltage operation and they introduce new challenges for future research.

4.1 Further Work

To bring the research from this thesis further, a new iteration of the presented chip can be made. The next chip iteration should include:

- Modify N-type ULV:
  - All signals should be available separately on pad (***Voffset+***, ***Voffset-***).
  - Size of ***En*** transistor should be reduced.
  - Reduce ***KPp*** size.
  - Rely less on layout simulations and more on intuition.
  - Set ***Ep*** of both inverters to the same size.
- Implement P-type ULV7 inverter:
  - Test average properties in a chain configuration.
- Make new PCB:
  - Use coax cables for all pad connections to reduce noise.
  - Add extra measuring connection for all inputs.
- Perform additional measurements:
  - Use instruments that can measure smaller currents[1] to be able to measure current and calculate power, PDP and EDP.

On the carry gate circuits more work could be done on the ULV PTL carry circuits to solve their switching issues.

[1] Nano Ampere Resolution, preferably with transient current plots.

5 Bibliography

[1] D. Bol, R. Ambroise, D. Flandre, and J. Legat, "Interests and Limitations of Technology Scaling for Subthreshold Logic," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 17, pp. 1508-1519, 2009.

[2] Y. Berg, "Ultra low voltage static carry generate circuit," in *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, 2010, pp. 1476-1479.

[3] Y. Berg, "Differential high speed ultra low-voltage pass transistor boolean logic," in *Faible Tension Faible Consommation (FTFC), 2013 IEEE*, 2013, pp. 1-4.

[4] Y. Berg, D. T. Wisland, T. S. Lande, and S. Mikkelsen, "Ultra low-voltage digital floating-gate UVMOS (FGUVMOS) circuits," in *Circuits and Systems, 1998. ISCAS '98. Proceedings of the 1998 IEEE International Symposium on*, 1998, pp. 37-40 vol.2.

[5] Y. Berg, O. Mirmotahari, J. G. Lomsdalen, and S. Aunet, "High Speed Ultra Low Voltage CMOS inverter," in *Symposium on VLSI, 2008. ISVLSI '08. IEEE Computer Society Annual*, 2008, pp. 122-127.

[6] R. P. Brent and H. T. Kung, "A Regular Layout for Parallel Adders," *Computers, IEEE Transactions on,* vol. C-31, pp. 260-264, 1982.

[7] P. M. Kogge and H. S. Stone, "A Parallel Algorithm for the Efficient Solution of a General Class of Recurrence Equations," *Computers, IEEE Transactions on,* vol. C-22, pp. 786-793, 1973.

[8] C. S. Wallace, "A Suggestion for a Fast Multiplier," *Electronic Computers, IEEE Transactions on,* vol. EC-13, pp. 14-17, 1964.

[9] Y. Tsividis, *Operation and Modeling of the MOS Transistor*: Oxford University Press, 1999.

[10] M. Alioto, "Ultra-Low Power VLSI Circuit Design Demystified and Explained: A Tutorial," *Circuits and Systems I: Regular Papers, IEEE Transactions on,* vol. 59, pp. 3-29, 2012.

[11] R. R. Troutman, "VLSI limitations from drain-induced barrier lowering," *Solid-State Circuits, IEEE Journal of,* vol. 14, pp. 383-391, 1979.

[12] H. S. Bechmann, "Implementation of an Ultra-Low Voltage Robust Low-Power Static Domino Inverter," ed. Nanoelectronics Conference (INEC), 2014 IEEE 5th International, 2014.

[13] O. Mirmotahari and Y. Berg, "Robust low-power CMOS precharge logic," in *Faible Tension Faible Consommation (FTFC), 2013 IEEE*, 2013, pp. 1-4.

[14] Y. Berg and O. Mirmotahari, "Ultra low-voltage and high speed dynamic and static CMOS precharge logic," in *Faible Tension Faible Consommation (FTFC), 2012 IEEE*, 2012, pp. 1-4.

[15] Y. Berg, *Ultra low voltage static carry generate circuit Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, 2010.

[16] Y. Berg and M. Azadmehr, "Novel ultra low-voltage and high-speed CMOS pass transistor logic," in *Faible Tension Faible Consommation (FTFC), 2012 IEEE*, 2012, pp. 1-4.

[17] Y. Berg, "Title," unpublished|.

6 Appendix

6.1 Implementation of an Ultra-Low Voltage Robust Low-Power Static CMOS Domino Inverter

Implementation of an Ultra-Low Voltage Robust Low-Power Static Domino Inverter

Halfdan Solberg Bechmann
halfdasb@ifi.uio.no

Yngvar Berg
yngvarb@ifi.uio.no

Department of Informatics, University of Oslo, Norway

***Abstract*— The Ultra-Low Voltage (ULV) logic style utilizes floating-gates to increase the current through evaluation transistors when lowering the supply voltage. In this manner the ULV logic can achieve a high speed for low supply voltages. Research and simulations on ULV logic in the past years have shown a significant potential for high-speed properties at ultra-low supply voltages. Therefore, the main focus of this paper is on the high-speed operation of this logic style in the ultra-low voltage domain. This paper is the first to verify these ULV logic style properties trough measurements. These first measurements are made by implementing the 7th iteration of the ultra-low voltage logic style (ULV7). The manufactured circuit is an N-type ULV7 inverter and the layout specific details of the implementation in the 90nm Nexsys® process from TSMC are presented. Further elaborations on the design choices are also made. In addition to the ULV7 inverter, a conventional CMOS inverter with equivalent scaling is manufactured on the same chip to serve as a benchmark for the ULV7 results. In order to make direct measurements and analyze analog properties of the circuits, both designs are scaled to drive a 20pF measurement setup. This first chip implementation is made in a batch of 5 chips, which are all used in the measurements to include process variations. The measured results presented show that the average propagation delay of the ULV7 inverter relative to the conventional CMOS inverter is more than 60 times faster with a lower relative standard deviation for a supply-voltage of 200mV.**

***Index Terms*— ULV, Ultra-Low Voltage, Floating Gate, Digital, High-Speed, Domino Logic, 90nm, CMOS**

I. Introduction

The expanding market for portable devices and the rapid growth of the internet of things creates an increasing demand for electronics powered by lightweight batteries and alternative energy sources. This leads to higher low-power data processing requirements increasing the importance of low-power logic styles in modern IC design. The simplest way to reduce the power consumption of a system is to reduce the supply voltage which drastically reduces both the static and dynamic power dissipation at the cost of a higher gate delay and a slower circuit. The increasing use of energy harvesting systems to power lightweight devices in the internet of things also introduces a demand for circuits that are not only low power but also need to run on a low supply voltage to avoid the power overhead of dc-dc converters. The continuous reduction of transistor sizes also causes increased leakage through sub-threshold currents and gate-oxide tunneling [1] that can only be reduced by lowering the supply voltage.

One way to increase the speed while maintaining a low supply voltage is to use a floating gate logic style to increase the transistor current and device speed by super charging the gate of the evaluation transistor but still maintain the low supply voltage needed to tackle sub-threshold leakage and keep the power consumption low. This type of ultra low voltage logic (ULV) has been in development for years and has been presented in several papers, such as [2] and [3]. A lot of work in has also been done on simulation analysis of ULV circuits but little research has been done on manufacturing hardware implementations of these circuits. This paper aims to bring the focus of the ULV research to hardware implementations and physical measurements of the logic style by implementing an inverter from the 7th iteration of the ULV logic (ULV7) and provide measurements showing the high-speed properties of the ULV7 logic style at ultra-low voltages.

II. ULV7 Logic

The idea behind the ULV logic is to increase the current through the evaluation transistors to in turn increase the speed while keeping the supply voltage low to save energy and thereby achieve more processing power per energy consumed and higher speed for applications where a low supply voltage is desired.

The ULV7 logic style has been referred to as robust low-power CMOS precharge logic in previous papers like [4] and is based on the ULV5 [3] logic style. Like other domino logic it has a precharge phase and an evaluation phase. In the precharge phase the output is charged to ‘1’ for the N-type circuit and to ‘0’ for the P-type circuit, but when the evaluation phase arrives, the circuits will either switch or not switch depending on the input(s) and logic. For an N-type circuit with a precharge value of ‘1’ the output logic is decided by a pull-down network (PDN) and the complimentary P-type circuit logic is decided by a pull-up network (PUN). An N-type and a P-type ULV7 inverter is shown below followed by an explanation of the two phases of operation and the role of each transistor.

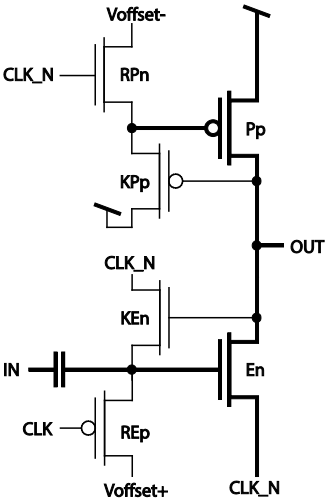

Fig. 1. N-type ULV7 inverter.

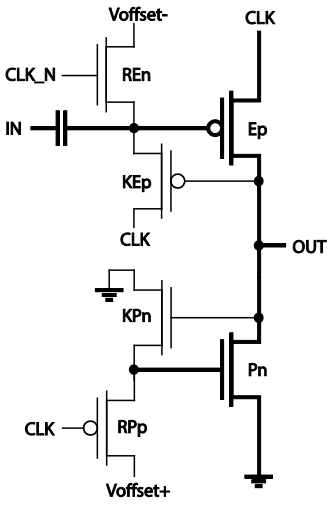

Fig. 2. P-type ULV7 inverter.

Precharge phase: When the ULV7 N-type inverter shown in Fig. 1 enters the precharge phase, the floating gate of the evaluation transistor (***En***) is charged to ***Voffset+*** through ***REp***, and the gate of the precharge transistor (***Pp***) is charged to ***Voffset-*** through ***RPn***. ***Voffset+*** and ***Voffset-*** are usually set to Vdd and ground respectively but can also have other values to reduce or increase the speed and power consumption. A signal can also be applied to implement logic functions or power gating. This means that both ***En*** and ***Pp*** are semi-conducting and slowly charging the output to '1'.

Evaluation phase: During the evaluation phase the transistors recharging the ***Pp*** and ***Ep*** gates are turned off so that an arriving rising edge will cause the En floating gate to be super-charged and drive the output to '0' through ***CLK_N***. The low output will then turn ***KPp*** on and pull the ***Pp*** gate to '1' effectively securing the output to '0'. On the other hand if the input stays at '0' the weak ***KEn*** transistor will be on and slowly lower the ***En*** floating gate voltage. This means that the ***KEn*** strength needs to be designed to make sure that the precharge value is held and that the circuit accepts inputs through the whole evaluation phase.

The P-type circuit works based on the same principle but in a complimentary manner to the N-type, with the floating gate on the ***Ep*** PMOS transistor as shown in Fig. 2.

III. Design Considerations

Simulations of the ULV7 logic style have proven the circuit to be fast, energy efficient and robust at ultra-low voltages in [5] and [4]. To show that these properties are also evident in silicon implementations of the logic, a layout of the ULV7 N-type inverter is designed and produced using the 90nm Nexsys® process from TSMC. To analyze the performance of the ULV7 inverter, a conventional CMOS inverter is also manufactured on the same chip as a benchmark for the ULV7 inverter.

The devices used in the ULV7 design are; the evaluation NMOS transistor (***En***) with the ***RTMOMCAP*** as the coupling capacitor on the gate terminal, the precharge PMOS transistor (***Pp***), the precharge and evaluation transistor gate keepers (***KPp*** and ***KEn***) and gate recharge transistors (***REp*** and ***RPn***). These are all shown in the layout in Fig. 3 where the NMOS transistors are placed within one guard ring and the PMOS transistors in another. The guard rings are necessary because all the transistors have at least one terminal connected to the pad-frame. To satisfy the antenna rules for the small ***KEn*** transistor, an N+/P-well diode is added to the ***CLK*** input to reduce the ratio between the metal and oxide diffusion area. The diode is visible in the lower right corner of the full layout in Fig. 6.

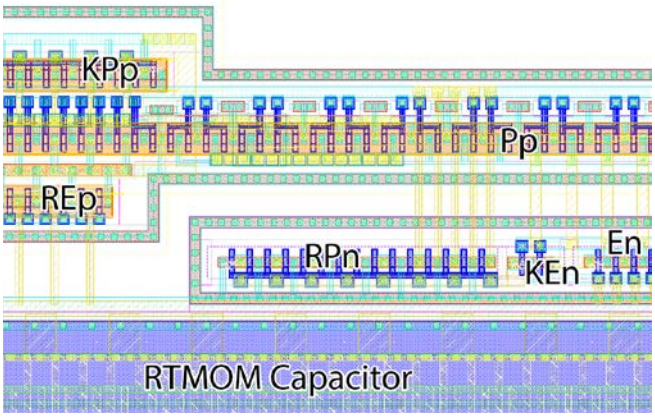

Fig. 3. ULV7 layout.

The devices used in the conventional inverter are an NMOS and a PMOS transistor, both with a guard ring because of their pad-frame connections. These are shown in Fig. 4. A grid of substrate ground connection is placed around the two inverters for electrical isolation to avoid crosstalk between the two, and can be seen both in the top and bottom of Fig. 4 and around the two inverters in Fig. 6.

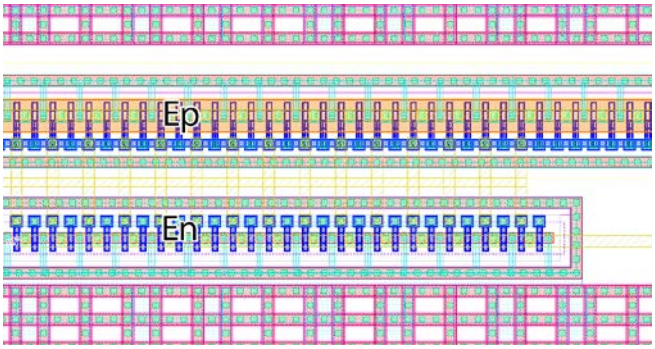

Fig. 4. Traditional inverter layout.

Ideally the transistors would all be minimum sized to save area, but to be able to do measurements without using buffers on the output that would mask their analog properties, the inverters are both scaled to drive a 20pF load. The test setup capacitance of 20pF comes mostly from the oscilloscope probes with a 15pF intrinsic capacitance, the other 5pF are based on the assumption that the PCB capacitance is 3-4pF including the socket, and that the pad frame, bonding wires and chip carrier leads have a joint capacitance of about 1pF.

The sizing of the transistors is done exclusively using transistor fingers. This is done to reduce the impact of process variations and allow the circuit to operate in a wider range of supply voltages because the effects of other low-voltage scaling techniques [6], although more area efficient, are highly dependent on the supply voltage and more susceptible to process variations.

Creating a "fair" conventional CMOS benchmark circuit can be a slightly unobjective task, so for simplicity the maximum number of fingers is set to 100 for both circuits and optimized based on this constraint. This makes the circuits equally wide but gives the traditional CMOS a small advantage in terms of area. The transistor sizes are shown in TABLE I and the full layout is presented in Fig. 6.

The transistors used for both inverters are standard threshold devices from the 90nm Nexsys process with a simulated threshold voltage of 266.8mV with no subthreshold strength enhancement applied.

TABLE I. TRANSISTOR SIZING

	ULV7						CMOS	
Transistor:	***En***	***Pp***	***KPp***	***RPn***	***REp***	***KEn***	***NMOS***	***PMOS***
Fingers:	100	90	30	15	30	2	30	100

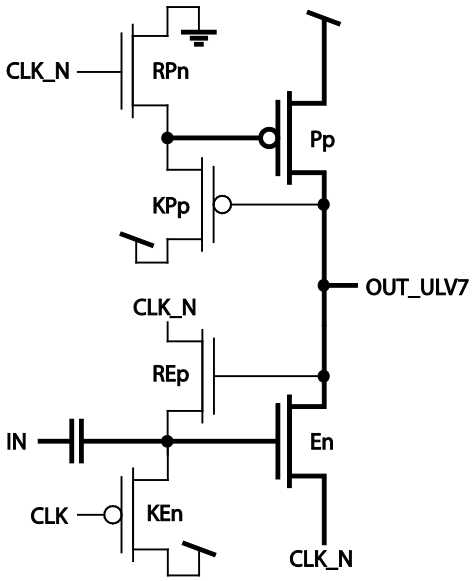

Fig. 5. Schematic of the implemented ULV7 N-type inverter.

To reduce the area impact of the capacitor, a rotative metal-oxide-metal (RTMOM) capacitor was chosen because it can be placed in the metal layers above the substrate and logic. The RTMOM capacitor has a simulated capacitance of 890fF and is placed next to the transistors in the layout for improved lucidity as shown in Fig. 6.

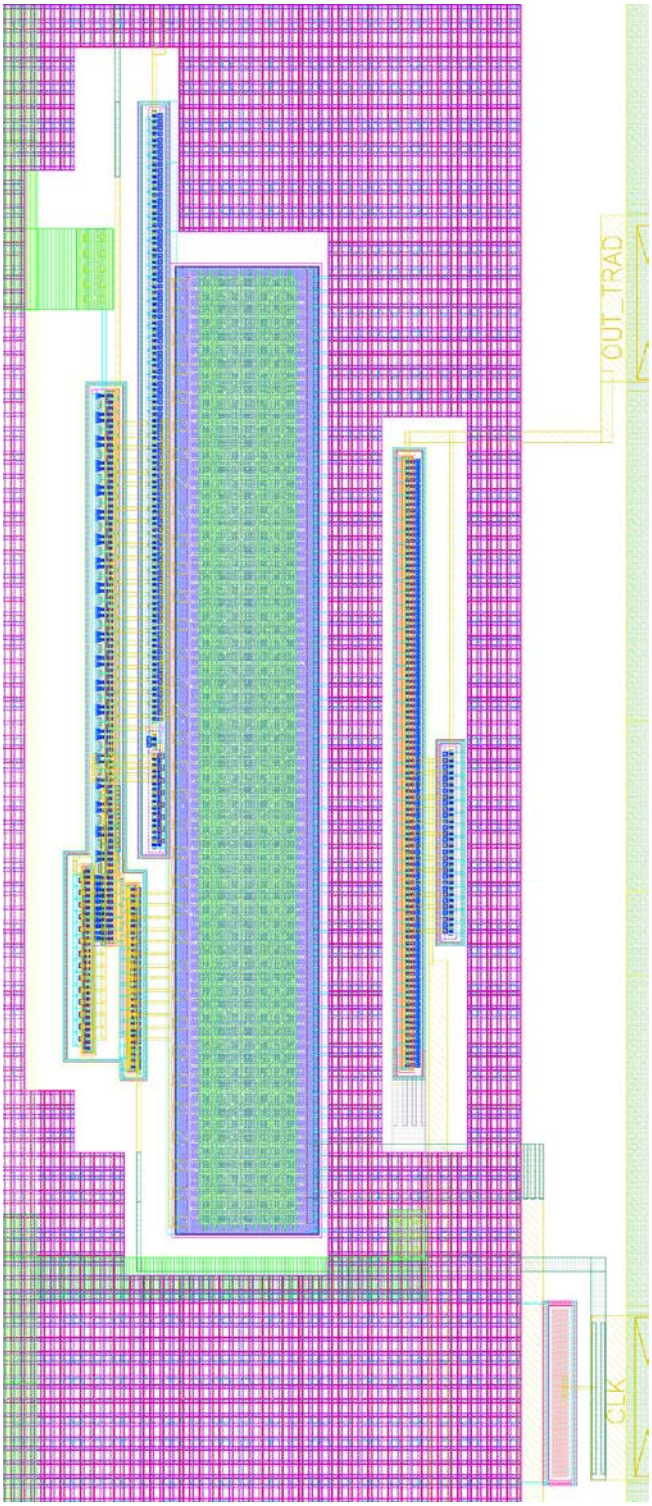

Fig. 6. Full layout.

IV. TEST SETUP

The test setup consists of a 4-channel signal generator to generate the clocks and inputs, a voltage supply and an oscilloscope, all controlled from matlab through GPIB.

An illustration of the instruments and he test setup is provided in Fig. 7. The input generated by the TTi1244

waveform generator are square wave signals filtered with a 10MHz Bessel filter applied for ULV amplitudes to reduce the input rise and fall time. The square wave inputs give a slight advantage to the conventional CMOS inverter because of the short rise/fall time.

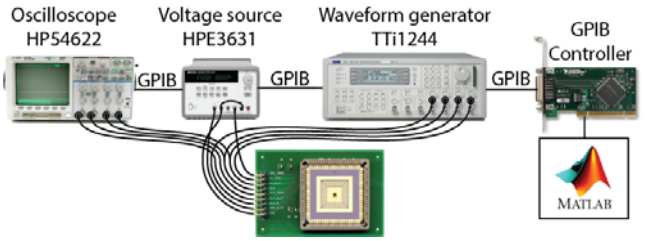

Fig. 7. Measurement setup.

To achieve a stable and static high output from the ULV7 inverter throughout the evaluation phase, the strength of the evaluation transistor needed to be reduced; this was done by applying an offset to and reducing the amplitude of the CLK signal effectively denying a full recharge of the ***En*** floating gate. A better way to allow post-production strength adjustments would be to route the ***Voffset+*** and ***Voffset-*** signals (Fig. 1) out to a pad instead of Vdd and ground.

V. Measurement Results

Measurements for a 1.2V supply are presented in Fig. 9 and show that the speed of the two inverters at this voltage is approximately the same.

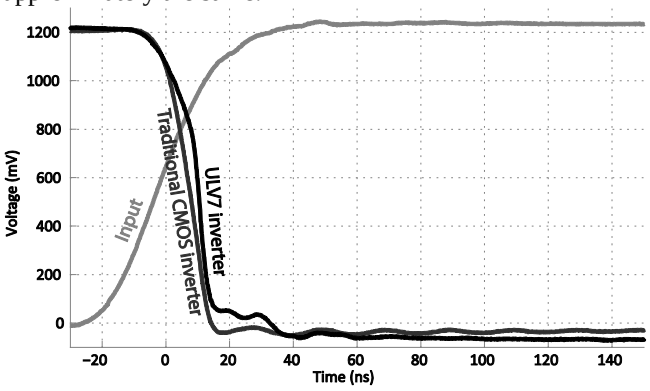

Fig. 8. ULV7 and traditional CMOS inverter, 1.2V supply.

The transient plot in Fig. 10 shows the significantly higher speed of the ULV7 inverter compared to the traditional CMOS inverter; a 2-point moving average filter has been applied in the oscilloscope to reduce noise.

A batch of 5 chips was measured and the results are presented in TABLE II showing that the average ULV7 inverter delay is only 45.4ns at 200mV compared to the average of 2.74µs for the conventional inverter. This shows that the ULV7 inverter on average is more than 60 times faster than the conventional CMOS inverter. This difference is shown in Fig. 9. The measurements also show that the slowest of the 5 traditional inverters has a gate delay that is 72 times higher than the slowest ULV7 inverter. This is because the relative standard deviation of delay calculated based on the 5 measured chips is smaller for the ULV7 circuit. These results are also presented in TABLE II.

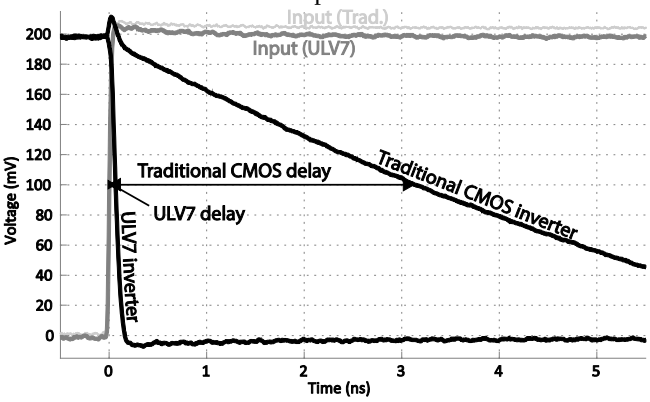

Fig. 9. ULV7 and a traditional CMOS inverter, 200mV supply voltage.

TABLE II. Gate delay comparison (5 chips)

Delay @ 200mV	ULV7	Std. CMOS	Diff.
Average	45.35ns	2.736µs	60.3x
Standard deviation (σ)	1.36ns	498.7ns	367x
Relative σ	2.9%	18.2%	6.3x
Worst Case	47.00ns	3.371µs	71.7x

VI. Conclusion

In this paper a ULV7 inverter has been implemented in a 90nm process showing that the ULV7 logic style offers significant improvement of delay at low voltages.

The ULV7 inverter was proven to be at least 60 times faster than a traditional inverter operating at 200mV.

VII. References

[1] D. Bol, R. Ambroise, D. Flandre, and J. Legat, "Interests and Limitations of Technology Scaling for Subthreshold Logic," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 17, pp. 1508-1519, 2009.

[2] Y. Berg, O. Mirmotahari, J. G. Lomsdalen, and S. Aunet, "High Speed Ultra Low Voltage CMOS inverter," in *Symposium on VLSI, 2008. ISVLSI '08. IEEE Computer Society Annual*, 2008, pp. 122-127.

[3] Y. Berg and O. Mirmotahari, "Ultra low-voltage and high speed dynamic and static CMOS precharge logic," in *Faible Tension Faible Consommation (FTFC), 2012 IEEE*, 2012, pp. 1-4.

[4] O. Mirmotahari and Y. Berg, "Robust low-power CMOS precharge logic," in *Faible Tension Faible Consommation (FTFC), 2013 IEEE*, 2013, pp. 1-4.

[5] O. Mirmotahari and Y. Berg, "Robustness of the Ultra Low-Voltage Domino Gates CMOS," in *CENICS 2013, The Sixth International Conference on Advances in Circuits, Electronics and Micro-electronics*, 2013, pp. 7-12.

[6] M. Alioto, "Ultra-Low Power VLSI Circuit Design Demystified and Explained: A Tutorial," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 59, pp. 3-29, 2012.

6.2 Scripts

Simple examples of the different types of scripts that are written to do measurements, plotting and calculations are provided in this section.

6.2.1 Measurement setup

6.2.1.1 Get data from oscilloscope

To read data from all four oscilloscope channels, an existing script from the University of Oslo servers was altered.

```
function [time, data] = HP54622_GetData3(HP54622Adr)
% function [time, data] = HP54622_GetData(HP54622Adr):
% Skrevet av henningg@ifi.uio.no 10/5-2004
% Drastically changed by Hï¿½kon A. Hjortland 2004-Nov-09
% Re-altered by Halfdan S. Bechmann 2014

% Documentation:
% http://cp.literature.agilent.com/litweb/pdf/54622-97038.pdf
% OUTPUT 707;":ACQUIRE:TYPE AVERAGE"<terminator>
% OUTPUT 707;":ACQUIRE:COMPLETE 100"<terminator>
% OUTPUT 707;":WAVEFORM:SOURCE CHANNEL1"<terminator>
% OUTPUT 707;":WAVEFORM:FORMAT BYTE"<terminator>
% OUTPUT 707;":ACQUIRE:COUNT 8"<terminator>
% OUTPUT 707;":WAVEFORM:POINTS 500"<terminator>
% OUTPUT 707;":DIGITIZE CHANNEL1"<terminator>
% OUTPUT 707;":WAVEFORM:DATA?"<terminator>

if nargin==0,
        HP54622Adr=HP54622_DefaultAdr
end

% Read data
GPIB_Write('WAVEFORM:FORMAT ASCII', HP54622Adr);
GPIB_Write('WAVEFORM:POINTS 2000', HP54622Adr);
GPIB_Write('DIGITIZE CHANNEL1, CHANNEL2, CHANNEL3, CHANNEL4', HP54622Adr);
data(:,1) = readChannel(HP54622Adr, '1');
data(:,2) = readChannel(HP54622Adr, '2');
data(:,3) = readChannel(HP54622Adr, '3');
data(:,4) = readChannel(HP54622Adr, '4');

% Get x-axis
GPIB_Write('WAVEFORM:XORIGIN?', HP54622Adr);
xo= GPIB_Read(HP54622Adr);
GPIB_Write('WAVEFORM:XINCREMENT?', HP54622Adr);
```

```
xi = GPIB_Read(HP54622Adr);
xinc = str2num(xi);
xorig = str2num(xo);
time = xorig + [0:1999]*xinc;

% Run
GPIB_Write('RUN', HP54622Adr);

% Function to read a channel's data
function data = readChannel(HP54622Adr, chan)
GPIB_Write(['WAVEFORM:SOURCE CHANNEL' chan], HP54622Adr);
GPIB_Write('WAVEFORM:DATA?', HP54622Adr);
s = GPIB_Read(HP54622Adr);
header_length = 2 + sscanf(s(2), '%d');
data = sscanf(s((header_length+1):length(s)), '%f,');
```

6.2.1.2 Run transient measurement

To control the instruments of the test setup and run transient simulations the following script is used.

```
function [time, data] = meas_script(vdd, f)
% Written by Halfdan S. Bechmann
if(nargin == 1)
    f = 1e4;
end
TTi1244Name = 6;

% Set up voltage source
HPE3631_Operate();
HPE3631_Init();
% Limit max current to avoid burning circuit on short
HPE3631_SetILimit(1,0.003);
HPE3631_SetILimit(2,0.003);
% Set Vdd
HPE3631_SetVolt(1, vdd);
HPE3631_SetVolt(2, vdd);

% Set up frequency generator
try
TTi1244_Init()
catch err
    error(sprintf('Set TTi1244 in GPIB mode!\nPress UTILITY->remote-
>\ninterface:GPIB\naddress:0%d', TTi1244Name))
end

% Set channel 1, CLK
TTi1244_SetChannel(1);
TTi1244_LockMode('MASTER'); % Set this channel to master
TTi1244_SetWaveform('SQUARE'); % Choose square waveform for digital clock

TTi1244_SetAmplitude(vdd);
%TTi1244_DCoffset(vdd/2); % no offset
TTi1244_DCoffset(vdd*0.7 + 0.11) % formula for setting a working dc offset

TTi1244_SetFrequency(f);
TTi1244_LockStatus('ON'); % Synchronize channels
GPIB_Write('FILTER BESS', TTi1244Name);

% Set channel 2, CLK_N
TTi1244_SetChannel(2);
TTi1244_LockMode('SLAVE'); % Sychronize the channel with master
TTi1244_SetWaveform('SQUARE');
TTi1244_SetAmplitude(vdd);
```

```
TTi1244_DCoffset(vdd/2);
GPIB_Write('PHASE 180', TTi1244Name);
TTi1244_SetFrequency(f);
TTi1244_LockStatus('ON');
GPIB_Write('FILTER BESS', TTi1244Name);

% Set channel 3, IN_TRAD
TTi1244_SetChannel(3);
TTi1244_LockMode('SLAVE');
TTi1244_SetWaveform('SQUARE');
TTi1244_SetAmplitude(vdd);
TTi1244_DCoffset(vdd/2);
TTi1244_SetFrequency(f);
TTi1244_LockStatus('ON');
GPIB_Write('FILTER BESS', TTi1244Name);

% Set channel 4, IN_ULV7
TTi1244_SetChannel(4);
TTi1244_LockMode('SLAVE');
TTi1244_SetWaveform('SQUARE');

TTi1244_SetAmplitude(vdd);
TTi1244_DCoffset(vdd/2);
GPIB_Write('SYNCOUT WFMSYNC', TTi1244Name);
GPIB_Write('SYNCOUT ON', TTi1244Name);

TTi1244_SetFrequency(f);
TTi1244_LockStatus('ON');
GPIB_Write('FILTER BESS', TTi1244Name);

TTi1244_ChannelEnable('ON', 1); % Enable CLK
TTi1244_ChannelEnable('ON', 2); % Enable CLK_N
TTi1244_ChannelEnable('ON', 3); % Enable IN_TRAD
TTi1244_ChannelEnable('ON', 4); % Enable IN_ULV7

% Set up Oscilloscope
pause(7);
HP54622_AutoScale(1);
pause(1);
HP54622_SetTimeScale(0.2/f);
HP54622_SetVerticalRange(1, vdd*1.5, vdd/1.2+0.11);
HP54622_SetVerticalRange(2, vdd*1.5, vdd/2.2);
HP54622_SetVerticalRange(3, vdd*1.5, vdd/2.2);
HP54622_SetVerticalRange(4, vdd*1.5, vdd/2.2);

trig = sprintf('TRIG:LEV %d', vdd*0.7);%+0.11);
GPIB_Write(trig, 24);
```

```
GPIB_Write('ACQ:TYPE AVER', 24); % Averaging to reduce noise
GPIB_Write('TIM:REF LEFT', 24);

pause(3);

% Get data
[time, data] = HP54622_GetData3;
 str = sprintf('SYSTEM:DSP "Measurement done at %dmV ;)"', vdd*1e3);
GPIB_Write(str, 24);
```

6.2.1.3 Supply voltage sweep

To do measurements with many different supply voltages.

```
% Written by Halfdan S. Bechmann
vdd = 0.19:0.01:0.5;
freq = logspace(4.5, 6.3, length(vdd));

for i = 1:1:length(vdd)
[time, data] = meas_script(vdd(i), freq(i));
%max_ulv = max(data(:, 4));
%max_trad = max(data(:, 3));
dly_ulv7(i) = time(1,find(data(:, 4)<(vdd(i)/2),1));
dly_trad(i) = time(1,find(data(:, 3)<(vdd(i)/2),1));
end
close all;
semilogy(vdd, [dly_ulv7; dly_trad], '*');
grid on;

fid = fopen( 'finished.txt', 'wt' );
  fprintf( fid, 'Finished!');
fclose(fid);
```

6.2.2 Reading and plotting data

6.2.2.1 Measurements

The measured data is saved in a matlab matrix (mat) file and then read, plotted and saved as an encapsulated post script (eps) file.

```
% Written by Halfdan S. Bechmann
name = 'measurements_200mV_wdcoffs';
close all;
% load data from measurement
load([name '.mat'])
xval = time*1e6-0.075;
yval = data*1e3;

% Plot
FigHandle = figure('Position', [100, 400, 720, 340]);
plot1 = plot(xval, yval, 'LineWidth', 2,  'color',[0,0,0]);

ylim(gca,[-15 215]);
xlim(gca,[-0.5 5.5]);

% Adjust figure cosmetics
set(gca, 'XGrid', 'on', 'XColor', [0.5 0.5 0.5], 'YGrid', 'on', 'YColor', [0.5 0.5
0.5], 'Fontname', 'calibri');
box off;
set(plot1(2),'color',[0.5 0.5 0.5], 'LineWidth', 2);
set(plot1(1),'color',[0.8 0.8 0.8], 'LineWidth', 2);
xlabel 'Time (ns)';
ylabel 'Voltage (mV)';
set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 9]);

% Save in encapsulated postscript vector graphic
print('-depsc','-painters','-loose','tmp');

%Changing font of eps file
fin = fopen('tmp.eps');
fout = fopen(['..\Figures\' name '.eps'], 'w+');
while ~feof(fin)
    s = fgets(fin);
    s = strrep(s, 'Courier', 'Calibri');
    fprintf(fout,'%s',s);
end
fclose(fout);
fclose(fin);
```

6.2.2.2 *Simulations*

Simulations are run in cadence and saved as a comma separated value (csv) file, then used to plot and save it as an encapsulated post script (eps) file.

```
% Written by Halfdan S. Bechmann
close all;
name = 'ULV_static_carry_32bit_prop0'; % prop0 / prop1
filename = [name '.csv'];
delimiter = ',';
startRow = 2;
formatSpec
= '%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f
%f%f%f%[^\n\r]';
fileID = fopen(filename,'r');
dataArray = textscan(fileID, formatSpec, 'Delimiter',
delimiter, 'MultipleDelimsAsOne', true, 'HeaderLines' ,startRow-1, 'ReturnOnError',
false);
fclose(fileID);
data = [dataArray{1:end-1}];
clearvars filename delimiter startRow formatSpec fileID dataArray ans;

xval = data(:, 1)*1e9-300.5;
yval = data(:, 2:end)*1e3;

FigHandle = figure('Position', [100, 400, 720, 340]);
plot1 = plot(xval, yval, 'color',[0,0,0 ], 'LineWidth', 2);

ylim(gca,[-20 320]);
xlim(gca,[-2 23]);

set(gca, 'XGrid', 'on', 'XColor', [0.5 0.5 0.5], 'YGrid', 'on', 'YColor', [0.5 0.5
0.5], 'Fontname', 'calibri');
box off;
set(plot1(1),'color',[0 0 0], 'LineWidth', 2);
set(plot1(33),'color',[0.5 0.5 0.5], 'LineWidth', 2);
set(plot1(34),'color',[0.2 0.2 0.2], 'LineWidth', 2);
set(plot1(35),'color',[0.32 0.32 0.32], 'LineWidth', 2);

uistack(plot1(1), 'top')

xlabel 'Time (ns)';
ylabel 'Voltage (mV)';

set(gcf, 'PaperUnits', 'centimeters');
set(gcf, 'PaperPosition', [0 0 16 8]);

print('-depsc','-painters','-loose','tmp');%['..\Figures\' name]);
```

```
%Changing font
fin = fopen('tmp.eps');
fout = fopen(['..\Figures\' name '.eps'], 'w+');
while ~feof(fin)
    s = fgets(fin);
    s = strrep(s, 'Courier', 'Calibri');
    fprintf(fout,'%s',s);
end
fclose(fout);
fclose(fin);
```