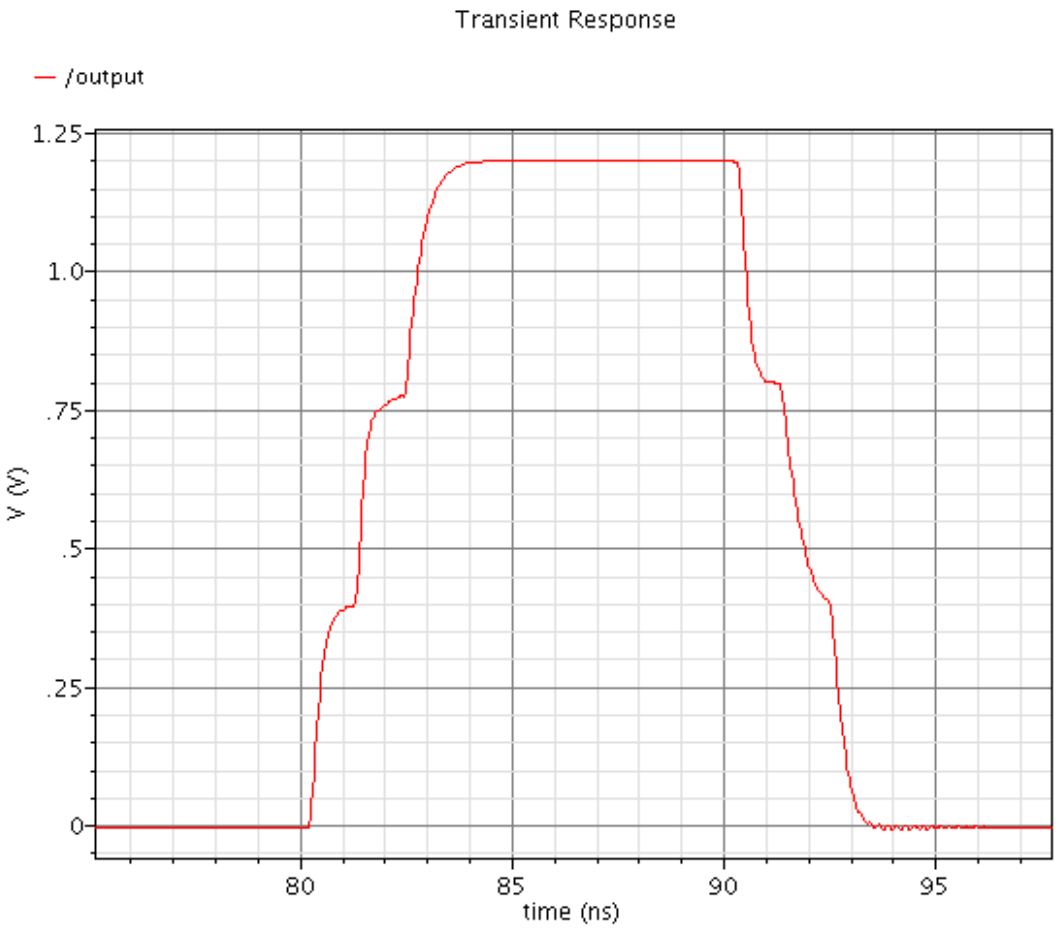
CHALMERS



Stepwise Pad Driver in Deep-Submicron Technology

Master of Science Thesis

SAMUEL KARLSSON

Chalmers University of Technology
University of Gothenburg
Department of Computer Science and Engineering
Göteborg, Sweden, October 2010

Stepwise Pad Driver in Deep-Submicron Technology

SAMUEL P. KARLSSON

Examiner: LARS SVENSSON

Chalmers University of Technology
University of Gothenburg
Department of Computer Science and Engineering
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

Cover: Simulated output signal of the stepwise pad driver, an energy efficient way to send signals to external devices. See page 6 and onward for further details.

Department of Computer Science and Engineering
Göteborg, Sweden October 2010

Abstract

In this report, we investigate the changes needed to implement a previous design of an energy efficient pad driver in a more modern 130 nm CMOS process, driving a load of 5 pF. A modified design procedure is presented and applied to a test case, resulting in a working driver. Simulations indicated energy savings of up to 50 percent compared to the traditional implementation of a pad driver, at switching times as low as 1.2 ns.

Contents

1 Introduction

Today, many electronic devices contain one or more integrated circuits. It is possible to fit a very large number of logical gates and connections inside one. An integrated circuit consists of a thin wafer of silicon with layers of other material diffused into it by advanced machinery. These material can be used to construct transistor switches, metal wires and other electronic devices onto the silicon to create a network, a circuit, that can manipulate information represented by electrical voltages.

Much effort is put into making this manufacturing equipment more and more sophisticated, allowing them to construct finer and finer details, so that more and faster logic can be fitted onto the limited amount of silicon wafer available. With these improvements comes another advantage: it takes less electrical energy to control the transistors inside. Each transistor is controlled by a gate, which is a rectangle of conducting silicon on top of the silicon wafer, with a layer of insulating oxide in between. This structure functions as a capacitor, with a capacitance determined by the area of the gate and the thickness of the oxide. Capacitance is a ratio between the voltage difference across the capacitor plates and the amount of electrical charge present at each plate. As the transistors are made smaller, the capacitance of the gate is reduced.

Unfortunately, sending signals around the circuit, by charging and discharging these capacitances, cannot be done for free. Electrical energy is supplied to the circuit from the outside, for example from a battery, all of which is eventually dissipated as heat. This is problematic because if integrated circuits gets too hot, they will no longer function properly. Much work has gone into advancing the technology used to manufacture integrated circuits in order to deal with these problems. In the last few decades, there have been enormous improvements to the speed and energy efficiency of integrated circuits.

However, an integrated chip isn't useful on its own. It needs to be connected to other devices in order to, for example, take in readings from keypads, control a speaker, or display information on a screen. These devices are very large compared to the microscopic transistors inside an integrated circuit and present larger capacitances to charge and discharge. These capacitances don't benefit from the technological advances of the machinery used to construct integrated circuits, and any improvements to them happens slowly, if at all. In many cases, these input capacitances cannot be reduced without interfering with their function. The designer of the logic inside the integrated circuits rarely has any control over the design of the external units. It is, however, possible to reduce the impact this has on the energy consumption

of the circuit, by spending some extra effort on the interface between the internal and external units.

To send signals from inside the chip to devices on the outside, a driver is used. Usually a driver is simply an inverter with transistors wide enough to send large currents to the load. This driver is controlled by a buffer that consists of a chain of inverters. The inverters start small in order to easily interface with the rest of the logic on the chip, and are gradually increased in size until they have the drive strength to properly send signals to the much larger driver transistors. This is illustrated in figure 1.

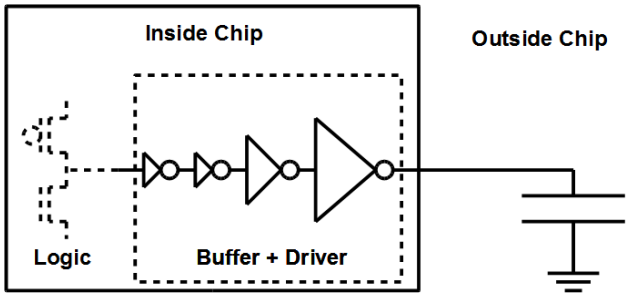


Figure 1: Sending signals from small internal logic to large external capacitance by using an inverter chain

The driver connects the output load to either the highest voltage source V_{dd} to send a logical 1 or the grounded source Gnd to send a logical 0. The voltage across the output capacitance will then change to match the voltage of the connected source. The transistors needs to be wide to allow enough current flowing through them to complete this process in a reasonable time. A schematic of this type of driver can be seen in figure 2.

This structure is robust and easy to design, but not optimal with regard to energy usage. Regardless of how much effort is spent fine-tuning it, there is a theoretical minimum energy usage for operating the driver. The formula for this is:

$$E = 1/2 \times C \times V_{dd}^2$$

C is the capacitive size of the load and V_{dd} is the supply voltage used. Reduction in energy consumption therefore relies on reducing either of those two factors. Usually, it is not possible to reduce the capacitance C of the load because it is determined by the external device. That leaves reducing the voltage that is used to charge the load. Some designs includes a secondary source with a lower voltage that is used only for the drivers, but whether

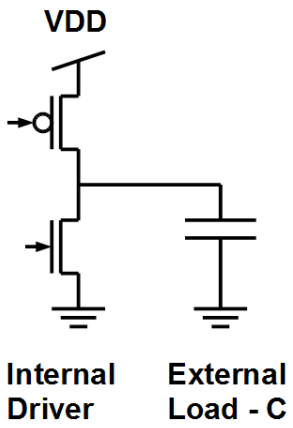


Figure 2: The last and largest inverter in the chain, known as the driver

this is possible depends on the requirements of the external units. It is rarely possible to use this method without reducing the performance of the system. Instead, a novel approach is required.

In this report, I describe the step-wise driver, a modification of the traditional driver. In a step-wise driver, the load is charged in several steps, each step trading extra charging time for lower energy used. The theoretical energy usage limit of this design is significantly lower than that of a traditional driver. However, extra logic is needed to control the individual charging steps. The main goal of this project is to verify that the overall design can be implemented correctly using the integrated circuit technologies available today, and to estimate how close the practically achievable energy savings are to the theoretical values.

2 Background

The overall design of the step-wise driver was proposed in 1994 by Svensson et al. [2]. Extensive theoretical modeling was done to determine the optimal number of steps as well as to aid in the selection of transistor types and sizes. Both simulations and fabrication of test chips using a 2μm process were performed. Practical results were close to the theoretical calculations. A reiteration of this work was performed in 1996 [1], putting higher demands on the speed of the design as well as utilizing a newer 800nm IC-technology.

These earlier designs used a simple design method that was sufficient to test the basic theory, but some minor differences between simulation results

and measurements on fabricated test chips were detected. Some thoughts were presented on the causes for this, in particular the effects of the chip packaging, but no further work was done at the time. Since then, there have been several technological advances. As transistors continue to shrink and the speed requirements of both the internal logic and communication with external devices increase, some of these unexplored effects may turn out to now be important, and other unexpected issues may arise.

3 Step-Wise Design

3.1 Driver

The basic structure of the step-wise driver can be seen in figure 3. The charging is split into a number of steps, corresponding to the voltages of the power supplies on the left. The best number of steps can be determined by the transition times required as well as the size of the load. In this work, I will use a design with 3 steps. Increasing the number of steps may provide some further energy usage reduction [2], but the gain is typically small and requires the transition time to be extended. The design will also increase in complexity and size with each additional step.

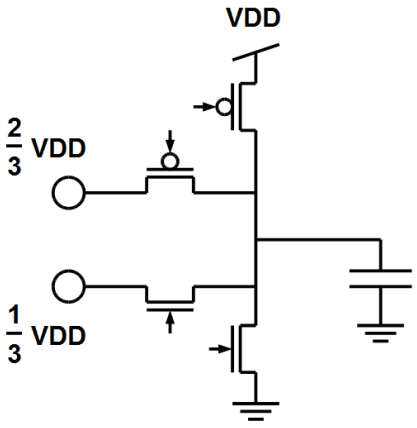


Figure 3: The step-wise driver design for 3 charging steps

The step-wise driver requires not only the usual voltage sources V_{dd} and Gnd, but also intermediate voltage sources distributed evenly in between. These voltage sources can for example be provided in the same way as normal sources for the rest of the chip, or with large tank capacitors outside of the chip [1], but the exact details are left to the designer. When the output

signal is stable at a low signal, the transistor connected to Gnd is conducting and all others are not conducting. In order to change the output to a high signal, the output is disconnected from Gnd and the intermediate transistors are turned on in sequence, as seen in figure 4. At any given time, only one transistor is turned on to prevent short-circuiting the voltage sources. The voltage drop experienced by the transistor switch is thus V_{dd}/N (where N is the number of steps selected; in our example design I use 3 steps) when each step starts, giving an average energy consumption of $1/2 \times C \times (V_{dd}/N)^2$ per step. After a certain amount of time, the transistor for that step is turned off and that of the next step is turned on. Multiplying with the number of steps, we get a theoretical lower limit of the energy usage that is only $1/N$ times that of the traditional driver. In our test case with 3 steps, this indicates an ideal energy usage of one third of what the traditional driver (described above) uses.

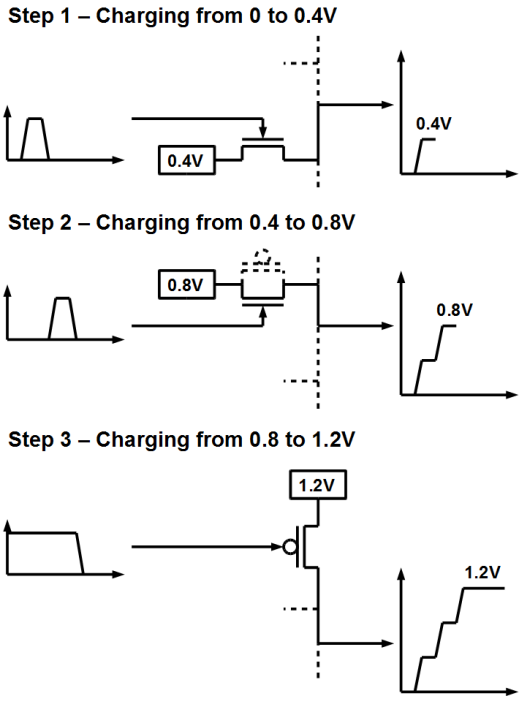


Figure 4: Changing the output signal from low to high, using three charging steps

For a fully comprehensive model of a transistor, fairly complicated formulas are needed. These complete models are generally too complex to use in simple pen and paper calculations and are therefore more suited for use in

computer programs. However, we still wish to make some early calculations to find rough values of transistor sizes and gate capacitances to base our design schematic on. More accurate computer simulations can then be used later to ensure that no unexpected complications has occurred.

In order to simplify the mathematical models, we may investigate what operating region the transistors will be in while the driver is active. Since this driver will be used in a digital design, we can safely assume that the voltages at the transistor gates will be either Gnd (logical 0) or V_{dd} (logical 1) when stable. The maximum voltage drop across each transistor is however only the size of one step, which is one third of V_{dd}. In the cases where the voltage across the conducting channel is much smaller than the drop between the gate and the silicon bulk, the transistor operates in the triode mode. In this mode, the current through the transistor is linearly controlled by the voltage across the channel. This is fortunate, because a transistor operating only in this region can be modeled simply as a resistor connected to a switch, as shown in figure 5. We call this model the RC-model since it only takes into consideration the component values of resistances and capacitances. The value of the resistor is determined by the size of the transistor and manufacturing parameters such as carrier mobility in the selected material. These parameters are usually built into simulation models provided by the manufacturer or provided as a combined value in their documentation.

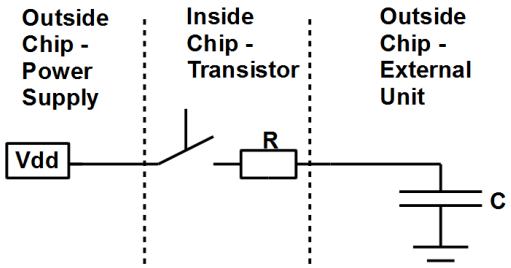


Figure 5: Simple RC-model of the charging path when modeling transistors as switch and resistance

Some extra consideration is needed when selecting the transistors used for each step, since other voltages than the usual V_{dd} and Gnd are used. In order to determine size and type of the transistor used at each step, it is useful to set up a computer simulation of the transistor on-resistance for a nMOS and pMOS transistor of a reference width (for ease of calculations 1μm was used). By sweeping the value of the voltage source connected to the transistors through the whole voltage span I determined the resistance

per unit of width at different voltages. The on-resistance curve will look like figure 6.

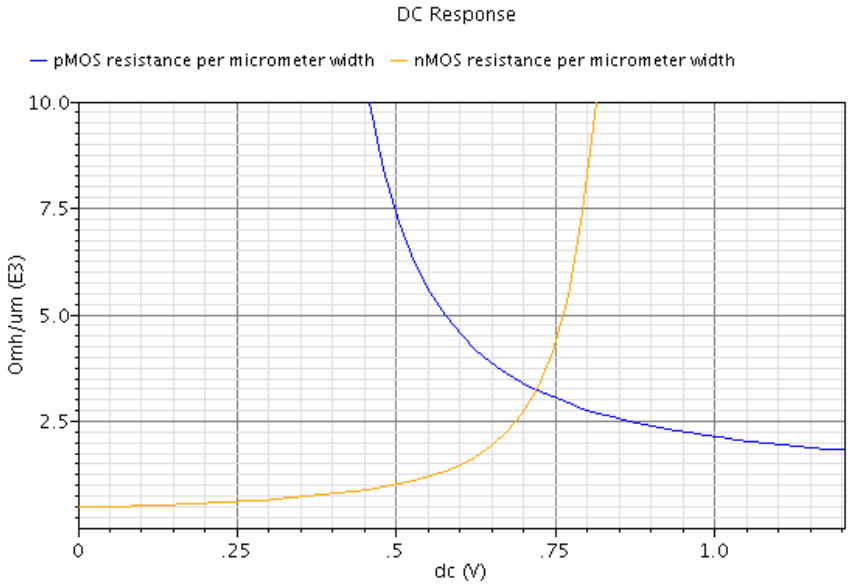


Figure 6: Resistance of 1μm wide nMOS transistor (yellow) and 1μm wide pMOS transistor (blue) at source voltage levels between 0V and 1.2V

With the aid of this curve, I could determine which type of transistor has the lowest resistance for each of the charging steps. In the cases where a voltage supply is only used to charge (up to 1.2V) or discharge (down to 0V), we check the only span reaching those levels and use the type of transistor best suited for it: in this case, a pMOS device for charging to 1.2V and an nMOS device for discharging to 0V. For 0.4V and 0.8V we have a span above as well as below that voltage. If both those spans are best covered with the same type of transistor, only one transistor of that type is needed. This is the case with 0.4V which is best covered with an nMOS transistor. However, for 0.8V we have lowest resistance with an nMOS device in the 0.4 to 0.8V span, but a pMOS device in the 1.2V to 0.8V span. In this case, one transistor of each type may be connected in parallel, as seen in figure 7. The transistors will be turned on and off individually.

If the transistor sizes have been chosen properly, the output voltage will have had just enough time to reach a value close to the supply voltage for that step before the next step is about to start. If the design is not matched to the timing requirements so that each step is too short, the voltage drop

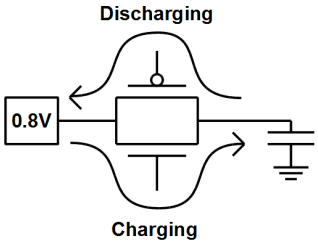


Figure 7: Using both nMOS and pMOS transistors for steps where resistance for one type is uneven

across the switch used in each step after the first will be larger. This will increase the energy consumption of the driver but still allow for correct overall function.

So far we have not yet determined the actual transistor widths to use, because we have not decided what resistance value we need. In our model of the charging path in figure 5, there is no theoretical limit to the size of the transistors. Making them wider will simply allow stronger electrical current to flow through the resistance, so that the charging process completes faster. However, in practice, there is a limited amount of chip area available to the designer, and this in turn limits the fastest possible speed of the driver. A transistor also contributes to the total amount of capacitance that needs to be charged. If the transistors are wide enough, the additional capacitances will eventually approach or even equal the size of the load. At this point, any further increase will increase the energy consumption significantly without providing much of a speed increase. Basing the transistor sizes on the RC-model was the method used in the early designs of the step-wise driver during the 90's and worked for the process technologies available at that time.

At this point, it is important to consider if there are any practical limitations to the charging time that we have not considered so far. The major issue of interest that was brought up in the earlier works but not actually explored is that of how the transistors are connected to the power supplies. Power supplies cannot be constructed inside the chip itself and must therefore be connected to the chip physically, for example through some sort of metal wire. This is done with a pad: a metal square on the chip where a bonding wire can be attached. The wires inevitably come with some inductance. These effects and how to model them have been explored in quite some depth [3] and accurate models exist. A practically effective way is to consider the pad and bond wire as one unit and model them as a small resistance due to the resistivity of the wire, a small capacitance due to the

plate-like shape of the pad, and an inductance due to the magnetic fields around the wire. However, the resistances and capacitances of the wire and pad are much smaller than those already present in our RC model, and may therefore be ignored. All these considerations taken together yields a still fairly simple model for the path between power supplies chip internals and the external unit, as seen in figure 8. In a computer simulation, all of the components may easily be included.

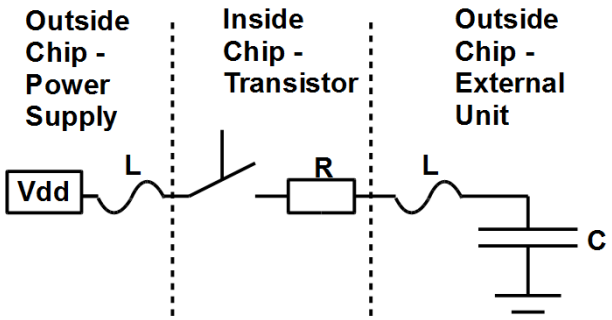


Figure 8: Charging path when modeling the chip packaging as inductances

When the inductance of the wires is connected to the capacitance of the load and energy is injected, it resonates at a certain frequency as the energy flows back and forth among the components [5]. This resonance frequency is determined only by the capacitances and inductances present, and these are fixed by the physical construction of the chip packaging and the load. The formula for this is:

$$f = 1/2 \times \pi \times 1/\sqrt{LC}$$

Substituting the values used in my implementation, the resonance has a period of approximately one nanosecond. This is within the range a driver can be expected to operate in and must therefore be taken into consideration.

The resonance frequency sets the absolute limit for how much current may flow through the system. As long as we don not try to charge the load too quickly, we will not reach this limit and the shape of the output signal will be correct. However, if we try to move charge into the load quicker than the rate determined by the resonance, we will not actually achieve the desired speed increase. Any attempts to reduce the charging time instead leads to the output signal not being able to stop rising when it reaches the desired output value, as seen in figure 9. This is called overshoot and is undesirable because, if extreme enough, it can damage the circuit. We therefore wish to

keep the overshoot to a minimum, while still maintaining a charging time that is as short as possible.

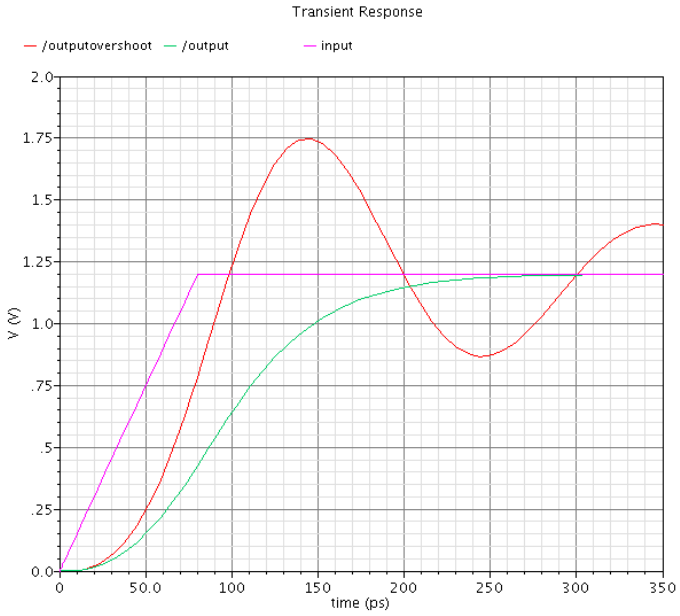


Figure 9: Ideal input signal shown in purple. Green waveform shows desired output. Red waveform shows output with overshoot

Fortunately, there is a straightforward theoretical method for determining when overshoot occurs. If the component values are known, we can calculate the quality factor, Q, of the circuit. The formula for Q when the inductance and capacitance are connected in series, as in our design, is:

$$Q = 1/R \times \sqrt{L/C}$$

Overshoot occurs when the resistance R is too low and the value of Q goes above 1/2. At exactly 1/2, the circuit will be as fast as possible without any overshoot. If we enter this value into the formula, we can solve for R:

$$R = 2 \times \sqrt{L/C}$$

For our test case, this results in a value of $R = 56.6$ Ohm.

The R given above will result in the fastest driver. In the case where we have a fixed timing requirement for the external devices, we might not need to operate at maximum speed and instead may pick larger R-values. In these cases, the transistor widths are determined according to those timing

requirements and the inductances in the model can be largely ignored. In my implementation, I had no specific requirement in mind, but instead wanted to investigate how quickly the step-wise driver can operate compared to traditional drivers. The next step is to determine if the transistor widths needed to reach this speed limit are small enough to be implemented on a chip. Referring back to figure 6, we can translate the required value of R given above to the required transistor widths. In this case, transistors are between 10 and 50 μm wide. Transistors of these sizes are easily implemented.

Since we are operating near the resonance frequency, it is difficult to match the high speed of the traditional driver with this new design. To first order, each charging step normally takes as long in either design, but since the step-wise driver needs several steps instead of just one, it is always slower. If this is a problem or not depends entirely on the timing requirements of the external device that is connected to the driver.

Now that we know the desired widths of the transistors, we can complete the design of the driver itself as well as what input signals we need to send to it. This is seen in figure 10. Note that input signals to pMOS transistors are inverted compared to those for nMOS; otherwise the shape of them are the same.

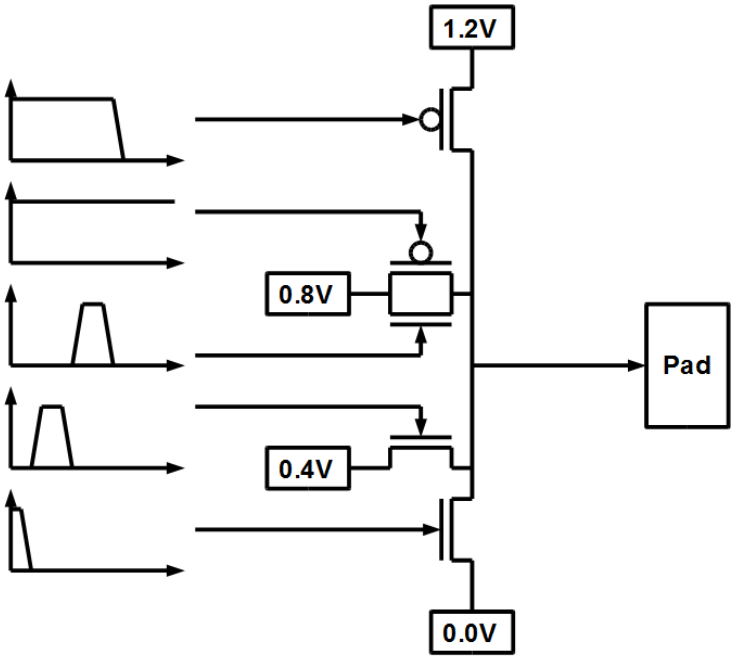


Figure 10: Step-wise driver design for 3 steps, with input signals included

3.2 Buffers

Since the the transistors in the drivers are considerably larger than those in the internal logic of the chip (0.15 μm), buffers are required to interface between them. The construction of the buffers are the same as for a traditional pad driver. However, since the size, and therefore the gate capacitance, of our transistors are not all identical (between 10 and 50 μm in width), we might want to construct differently sized buffers for the various transistor sizes to ensure that all signals goes through the buffers at roughly the same speed. In this work, I built two different buffers, one corresponding to the transistors connected to the two highest voltage supplies and one corresponding to those connected to the two lowest voltages. Depending on the amount of design time available for this step, the designer might want to construct one individual buffer for each transistor and attempt to match their delays to each other [4]. In our design, simulations showed that this was not required; it is unlikely to become important unless we pick a significantly higher number of steps.

Since only one transistor should be conducting at any given time, we must provide individual buffers for each transistor. In this design we have 5 transistors and so must have 5 buffers, as shown in figure 11. This is one of the downsides with this design. Not only does each buffer need to be larger since we have wider driving transistors, we must also include a significant number of them. As long as the buffers can be kept small, the total undesirable capacitance they contribute will be low enough compared to the load capacitance. If the goal is a driver that is very fast and the buffers controlling it will be large, it is important to perform some theoretical calculations early on to see which type of design is most suited for our chip. Fortunately, in our 3-step driver the buffers are still only a very small part of the whole design. Quick calculations shows that the buffer array will contribute less than 10 percent of the total energy dissipation.

3.3 Control Unit

A control unit is needed to interface with the rest of the chip. Normally, the input to a driver is a single input signal that should be replicated on the output load. However, since we have several buffers that needs to be controlled individually, a logic block must be designed that turns on or off each transistor switch in sequence. The exact implementation of this block is not integral to the function of the step-wise driver, but here I will explain one possible design.

Our unit is split into two parts. The first part (shown in figure 12)

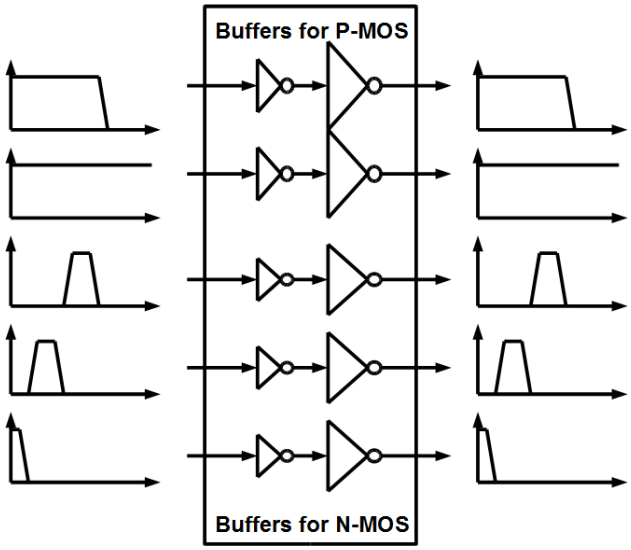


Figure 11: Buffer array, with one buffer for each driver transistor. Signal sequences sent through the buffer are not altered in shape

takes the input signal and creates copies of it. The original input signal is passed on unchanged but the first copy is delayed once and the second copy is delayed twice. The delay time corresponds to the time available for each step of the charging procedure. The signal delay is achieved by using a chain of throttled inverters. Every other inverter has additional throttle transistors that are controlled by an external control voltage, as seen in figure 13. This control voltage is intentionally set to a value other than the usual values Gnd and V_{dd}. This means that the gate-to-source voltage drop for the throttle transistor is lower than the full V_{dd} used in the rest of the design, and the throttle transistor's conducting ability is reduced. This leads to a longer charging time of the output, and signals sent through this inverter will be delayed more than with a normal inverter operating at full speed. One benefit of having an external control voltage is that the design can be adapted for multiple time constraints without requiring any changes to the internal transistor widths. A current mirror is used to ensure that the same current flows through both the P and N-transistors and achieve the same delay for both rising and falling signals.

The second part is a decoder block, consisting of a number of smaller logic gates, that translates the various input flanks into a sequence of short pulses to determine which transistor should be conducting at any given time. Due to the small number of input signals, the optimal design for this block

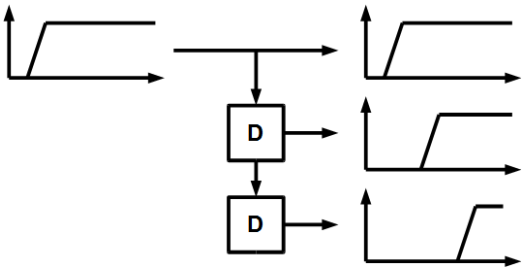


Figure 12: Abstract view of the block creating two delayed versions of it's input

can be found with hand calculations, for example with the use of a Karnaugh diagram. The overall structure of the decoder is shown in figure 14. To keep the figure simple, only the connections for the 0V and 0.4V decoders are shown explicitly. The other blocks are connected in the same manner but uses different combinations of the input signals. Since the 0.4V block is used both for charging and discharging, it has more input signals compared to the other blocks.

As an example of how each gate is constructed, we'll look closer into the gate controlling the 0V buffer. This buffer should send out a logical 1 only when the first and third flanks are at logical 0. This can be achieved with an AND-gate that uses the first and third flanks as inputs. The other units are constructed in a similar manner but uses other input flanks.

3.4 Full design

Now that all the individual blocks have been explained and all the input and output signals defined, we can combine them into a fully functioning step-wise driver. All the blocks are simply put together in a line, with each block feeding all its output signals straight into the next block only. This is illustrated in figure 15. The system takes a single input signal and attempts to replicate it on the output using the step-wise charging. Since the input signal is the same as for a standard driver, no big changes needs to be made to the other logic present on the chip if we wish to use a step-wise driver, or replace an earlier driver with one. This also lets us use the same design at multiple different parts of the chip without any changes to it. This design approach, to create self-contained blocks with one purpose each, also allows easier verification of functionality and easy estimates of energy consumption and timing issues later.

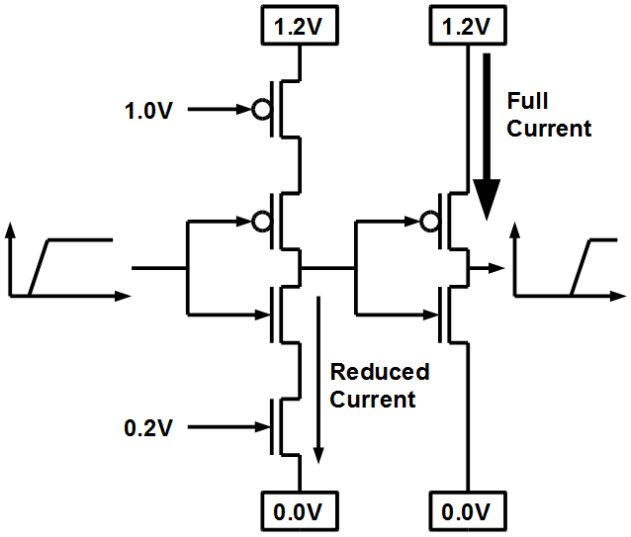


Figure 13: Schematic view of one delay element. Throttled inverter on the left side charges it's output slowly and delays the input signal considerably. Normal inverter on the right returns it to non-inverted form

4 Simulation

To ensure that the various parts functioned as expected, extensive simulations were performed at each step in the design process.

4.1 Simulation of Output waveform

In order to verify the functionality of our implemented design, we can look at the output and ensure that it looks like we expected. The simulated output waveform is shown in figure 16. There is no noticeable overshoot and each step is allowed roughly the right amount of time. This indicates that the Q-factor of the circuit is very close to the optimal value of 1/2 and that we calculated the transistor widths correctly.

We can also see that each step charges almost completely to the expected value before the next one begins. For the simple traditional driver, this is not something we need to take into consideration, because it has only one step. As long as it has time to reach a value that is interpreted as the correct logic signal, the external unit will receive the information we wish to send it. However, for the step-wise driver, if a step is not allowed enough time to charge, the output might not reach the desired voltage level before the step is turned off. Then, later steps will start at a lower voltage level than

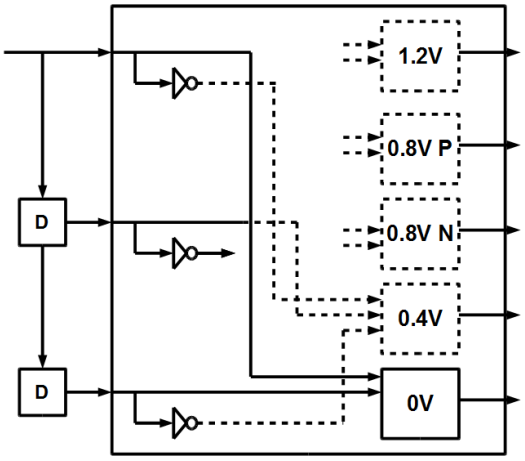


Figure 14: Connections inside the decoder unit

expected and have to charge more than the optimal V_{dd}/N per step. The larger voltage drop increases the total energy dissipation somewhat. It is not possible for each step to reach the exact desired voltage level since that would require waiting forever, but if the transistor widths are correctly selected, the mismatch will be minimized. If we find that the time for each step is not fully matched to the driver design, we may use the control voltage input to fine-tune the time per step, even after chip fabrication.

4.2 Simulation of Energy Consumption

To investigate the sensitivity of this control voltage tuning, simulations of the energy dissipation of the circuit were performed for a wide range of control voltage values (which controls the time each charging step is given). The results are plotted in figure 17, showing energy consumption as a function of the time needed to fully transition the output signal. For comparison, a series of traditional drivers were constructed for a range of required transition times. The energy usage for each traditional driver was estimated, and a curve interpolated from those values is also shown in the figure.

As long as the charging time is kept long enough, the energy consumption is fairly constant and shows a significant reduction compared to the traditional driver. Since small variations in the charging time do not alter the function of the circuit very much, we can simply use the simulation results to determine the voltage level to use for the fabricated chip later. In the case where we need to operate very close to the fastest possible speed, however, the exact value of the control voltage is very important. Setting it too high

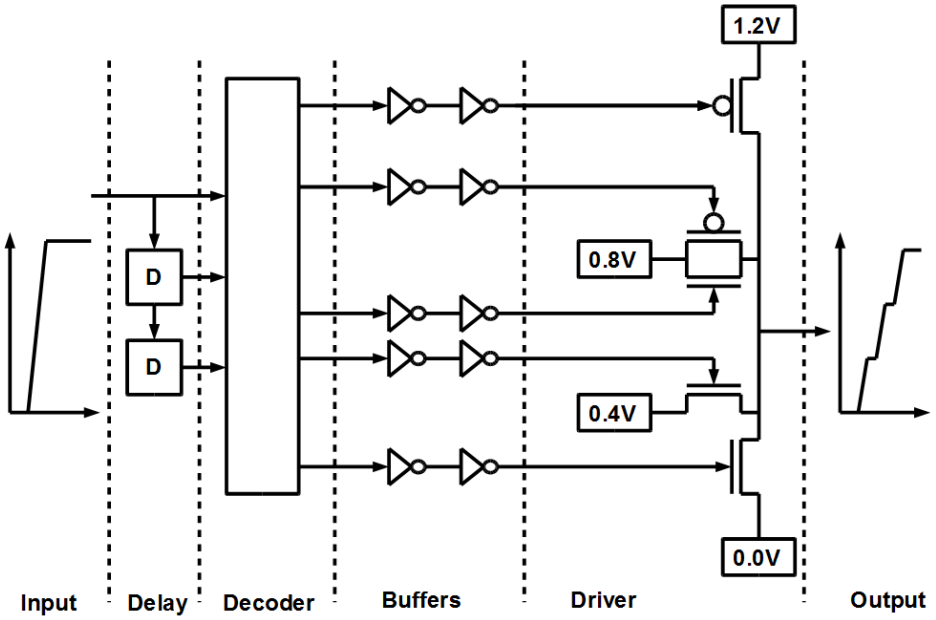


Figure 15: All units connected into the full step-wise driver design

will result in some of the steps hardly even getting halfway before the next one starts, and each following step then has a significantly higher voltage drop. Taken to the extreme, only the last step will have any significant time to charge at all, and the driver will then function like a traditional driver. However, all the extra logic then performs no useful function but still contributes to the energy dissipation of the circuit. At this point it is better to use a traditional driver design.

For my design, simulations results show a minimum energy dissipation of 3.94 pJ per transition for the step-wise driver. Compared to the simulated traditional driver, which has an energy dissipation of 8 pJ per transition, I achieved an improvement of about 50 percent.

Since we have implemented a current mirror as part of the control unit, it is of interest to simulate how much energy it consumes over time. Simulations show a rate of 1.08 pJ/μs. Thus, the driver needs to be operated at least once per 4 μs, otherwise the energy saved with the step-wise approach is lost to the extra cost of running the current mirror.

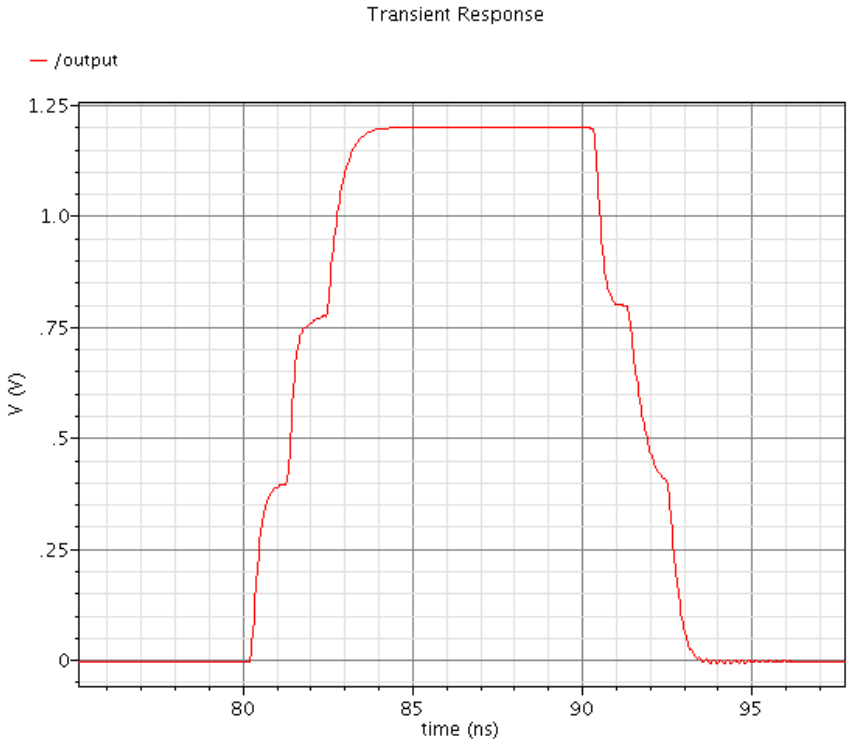


Figure 16: Simulated output signal from the complete step-wise driver design, switching from low to high and then back to low

5 Layout

The layout of the step-wise driver can be seen in figure 18. The V_{dd} and Gnd can be supplied in the same way as for other logic by using wide metal rails. The cells for the control unit were implemented in a long row between these rails, with each cell feeding the signal it generates to the next cell in the row. Some care should be taken early on to ensure that there is enough room between the rails to allow all the signals to be passed between and inside cells. In the layout created for this design, a space of 10μm between the rails was sufficient. The only cell that requires an unusual layout is the driver itself, where two extra rails were added, for supplying the 0.4V and 0.8V voltages.

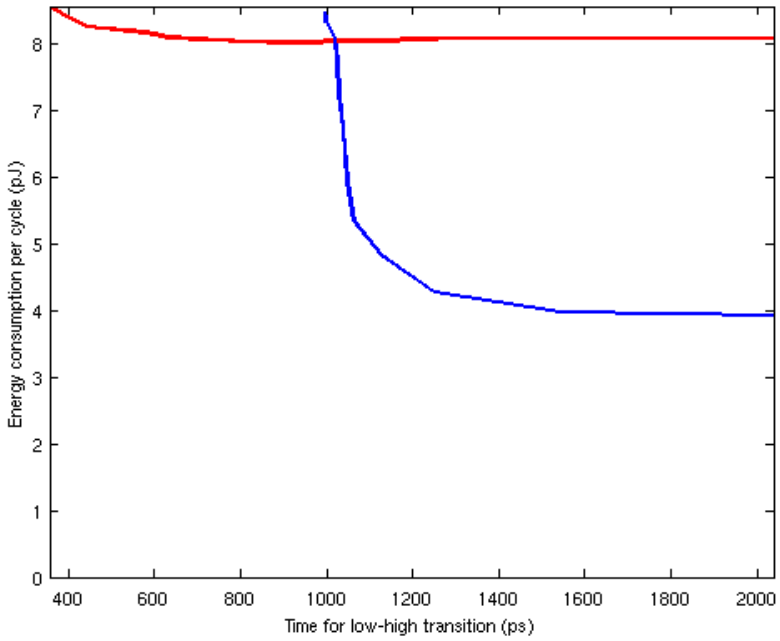


Figure 17: Simulated energy consumption as a function of time allowed for full transition. Top line shows standard driver consumption as comparison

5.1 Area Usage

The size of this particular implementation of the step-wise driver is $2300\mu m^2$, including the voltage supply rails. The area can most likely be reduced further by a more experienced designer. A traditional driver will only take up about $200\mu m^2$.

5.2 Pad Usage

Both the step driver and the traditional driver require pads for supplying the V_{dd} and Gnd voltages as well as pads for the output signals. The pad cell for the supply voltages along with cells to create a distribution network for them were included by the foundry. However, including a step driver on a chip requires extra pads for the intermediate voltages. In this case we only implemented one driver on the chip, and this allowed us to simply use normal input-output pads connected directly to the driver with wide metal wires. The control voltage for setting the delay between steps can be treated like a normal input signal. In designs with several step drivers, these signals

and voltages will need to be distributed somehow. Since it is likely that all the drivers will be located near the outer parts of the chip, near their respective output pad, one possible implementation is to add three additional metal rings there.

Input signals are usually generated internally on the chip, but since our design prototype is not connect to anything else on the chip, the input is taken from an additional input pad.

6 Conclusion

The RC-based method for designing a step-wise driver, proposed by Lars Svensson in 1994 [2], was found to be outdated when designing for modern CMOS circuits. Calculations and simulations showed that the charging speed of modern drivers can reach that of the resonant frequency between the capacitance of the load and the inductances of the bond wires in the chip packaging. A generalized methodology that takes this into account has been presented and shown to be practically sound.

A test structure has been designed, implemented and evaluated by simulation. It offers energy savings of up to 50 percent compared to a traditional pad drivers. This 3-step driver can reach transition speeds of down to 1.2 ns without compromising its energy efficiency.

The technological improvements in IC manufacturing during the last decade has made it easier and more motivated to implement step-wise drivers instead of traditional drivers. The main downside of the step-wise design is now the additional pads and additional design time needed. In some extreme cases, the slightly slower minimum transition times may also become an issue.

References

[1] L. Svensson, W.C. Athas, and R.S-C. Wen, "A sub-CV2 pad driver with 10ns transition time", *Proceedings of the 1996 International Symposium on Low Power Electronics and Design*, 1996, pages 105-108

[2] L. Svensson, and J.G. Koller, "Driving a capacitive load without dissipating fCV2", *Proceedings of the 1994 International Symposium on Low Power Electronics and Design*, 1994, pages 100-101

[3] H. Patterson, "Analysis of Ground Bond Wire Array for RFICs", *Proceedings of the 1997 Radio Frequency Integrated Circuits Symposium*, 1997, pages 237-240

[4] I. Sutherland, B Sproull, D. Harris, *Logical Effort: Designing Fast CMOS Circuits*. San Diego: Academic Press, 1999.

[5] R.C. Dorf, and J.A. Svoboda, *Introduction to Electric Circuits: 6th edition*. New York: John Wiley and Sons, 2004.

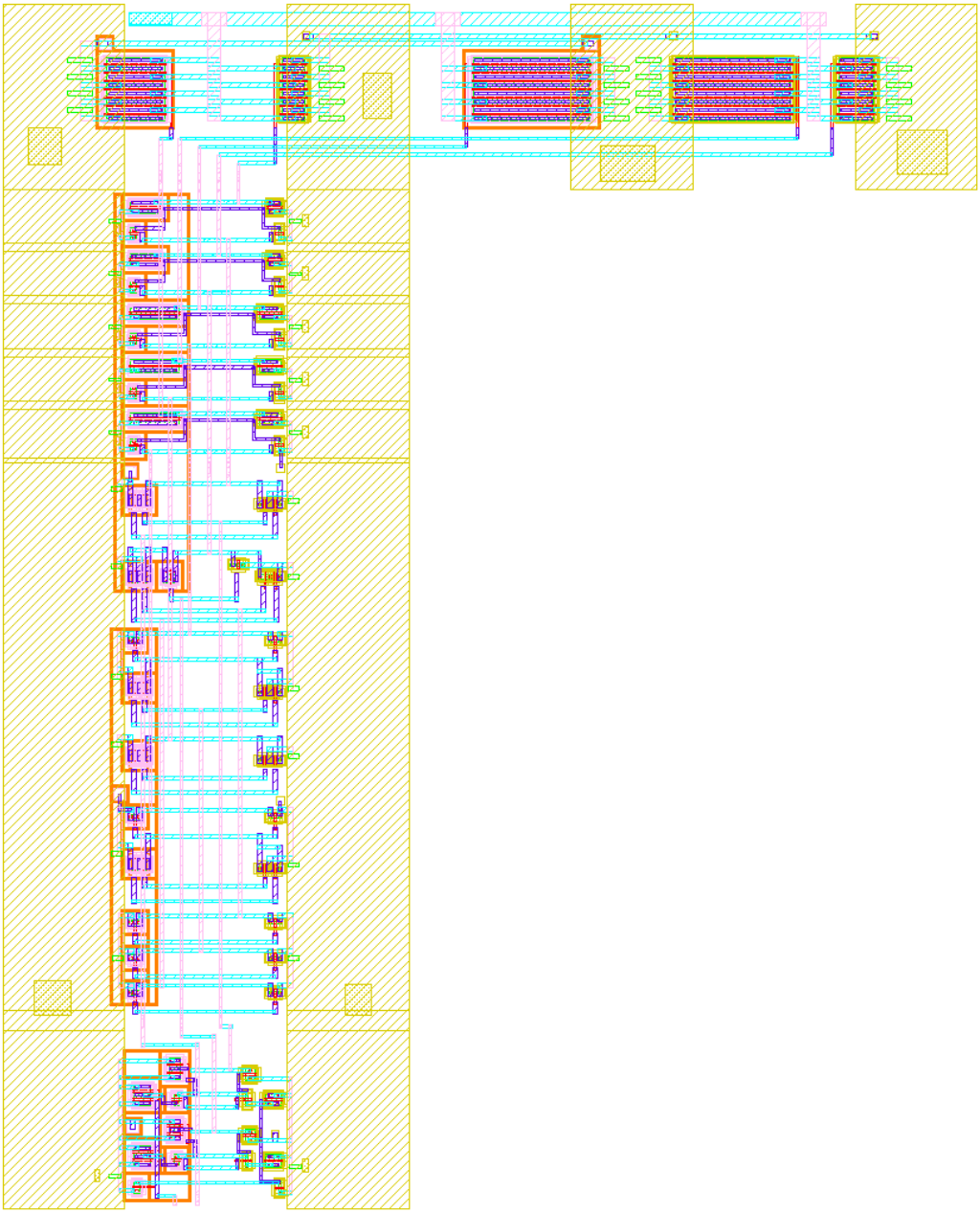


Figure 18: Layout of the step-wise driver with 3 steps