Four Pixel Digital Camera

TFE4152 – Design of Integrated Circuits

Alexander Rambøl Snarli & Andrea Catharina Worsøe
Group 26
November 18, 2019

Faculty of Information Technology and Electrical Engineering
Norwegian University of Science and Technology

Abstract

This design report describes the design of the integrated circuits needed for a four pixel digital camera. Both the digital and analog part of the circuit is described and simulated. The digital circuit is created using the hardware descriptive language Verilog, and simulated using the program Active-HDL. The analog part of the circuit it simulated using the program AIM-Spice. This report also describes the process of choosing the right dimensions for both NMOS and PMOS transistors using 180nm transistor technology, and choosing the right capacitors for the circuit. Both the digital and analog circuit has been tested in the simulations, to make sure the system behaves as intended. The simulations and tests shows that both the digital and analog circuits behave as inteded, and that the system operates as a four pixel digital camera successfully.

Contents

1 Introduction

While film photography might be very trendy, digital cameras are far more common today. A regular cellphone camera has maybe 12 megapixels, and can be made very small. These are quite complex digital systems. They consist of millions of sensors, which are analog, that sample the light and store it digitally. To illustrate how this works one can design a smaller camera, in this case of four pixels.

Figure 1 shows a block diagram of a four pixel digital camera. Inside **`Readout and Control`**, there are three blocks. The block **`Pulse_shaper`** will not be discussed in this report. The other two blocks, **`RE_Control`** and **`Pixel_electronics`**, represent the digital and analog circuits respectively. This report describes both of these systems. The digital circuit receives input from the user, and send signals to the analog circuit based on the input. The analog part acts accordingly to the signals received from the digital circuit. The analog circuit output signals to an analog to digital converter, called an ADC. The ADC will not be discussed in this report. **`RE_Control`** will be simulated in Active-HDL, using the hardware descriptive language Verilog to build the circuit. **`Pixel_electronics`** represent the analog part of the circuit, and will be simulated using AIM-Spice. The analog part of the circuit is based on 180 nm CMOS technology, where transistors are used as switches and to create a buffer circuit. This report will also describe how to choose the right dimensions for the transistors and the capacitor in the analog circuit.

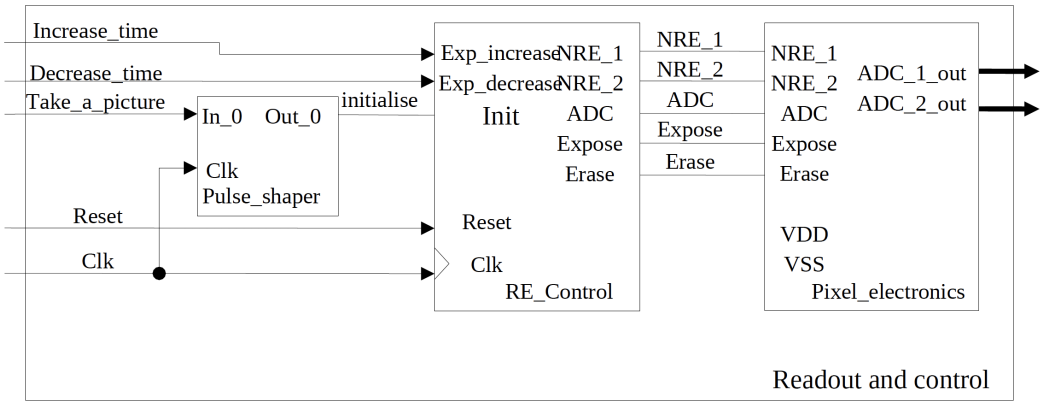

Figure 1: Block diagram of a four pixel digital camera. [1]

2 Theoretical Background

2.1 Digital Circuit Theory

The digital part of the circuit, **`RE_Control`**, receives input from the user, and control the analog circuit using these inputs. Figure 2 show the module for the digital circuit, with input signals on the left side and output signals on the right side of the module. **`init`** is the trigger signal, and is logically high when the user wants to take a photo. **`exp_inc`** and **`exp_dec`** will

be logically high, to increase and decrease the exposure time respectively, if the user wants to adjust the exposure time. `Clk` and `Reset` is controlled by control logic which will not be discussed in this paper. `Reset` will be logically high if the camera should be reset, and `Clk` is the clock signal for the circuit. The outputs `NRE_1` and `NRE_2` control which pixels should be read, while the `ADC` controls when the analog to digital converter should be initialized. `Expose` controls when the analog circuit should be exposed to light, and `Erase` erases the last exposure and resets the analog circuit.

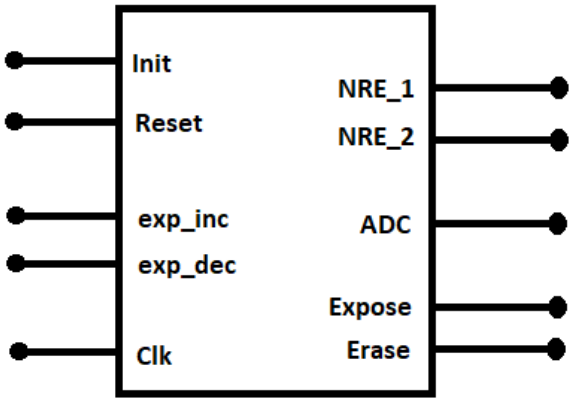

Figure 2: The digital circuit with inputs and outputs.

`RE_Control`, as seen in figure 3, consists of three modules called `CTRL_ex_time`, `Timer_counter` and `FSM_ex_control`. These are the exposure time controller, exposure time counter and finite state machine respectively.

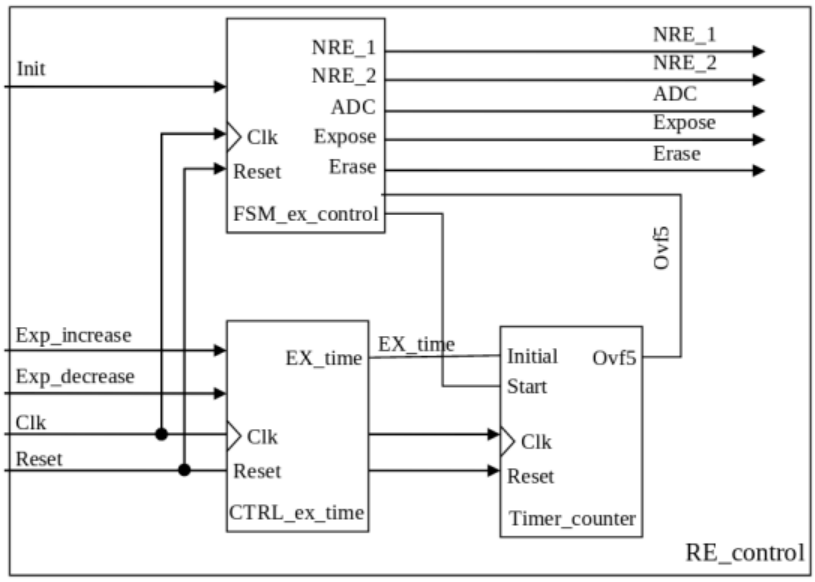

Figure 3: Block diagram of the digital circuit. [1]

2.1.1 Exposure Time Controller

The exposure time controller calculates the exposure time for the camera. The module has two user inputs, `exp_inc` and `exp_dec`. These inputs will increase or decrease the exposure time respectively. The user might want to adjust the exposure time based on the lighting conditions, which can affect the photographs quality. The exposure time is set to a base value, and increments or decrements a set value each time the user adjusts the exposure time. Exposure time also has an upper and a lower time boundary, and this module has to make sure exposure time stays within these boundaries. The exposure time controller sends the exposure time, `EX_time`, to the exposure time counter when the exposure time is calculated.

2.1.2 Exposure Time Counter

The exposure time counter controls the exposure time. The module receives the desired exposure time from the exposure time controller, and starts the counter when it receives the start signal, `Start`, from the finite state machine. When the exposure time is over, the overflow signal, `Ovf5`, will be sent back to the finite state machine.

2.1.3 Finite State Machine

The finite state machine is the most complex part of the digital circuit. A finite state machine, FSM, is a machine that has a set amount of states, which all have different characteristic values and functions. The FSM can only be in one of the states at any given time. The amount of states needed depends on the complexity of the system. The FSM will receive a trigger signal `init` which will start off the process of taking a picture. The start signal will be sent to the exposure timer counter, and it will receive the overflow signal when the exposure time is over. All the output signals from `RE_Control` to the analog circuit, are sent out through the finite state machine.

2.2 Analog Circuit Theory

The analog circuit is based on four identical circuits, which will output one pixel each. The circuit for each pixel is shown in figure 4.

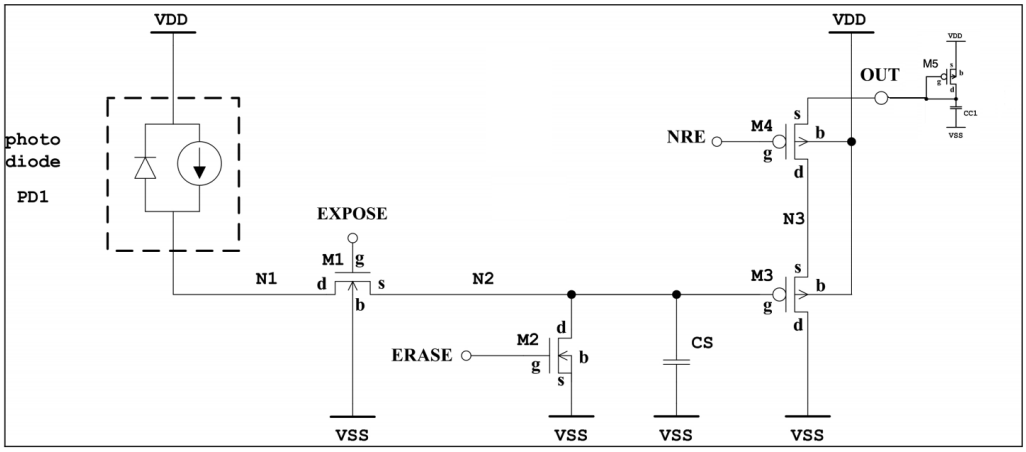

Figure 4: The analog circuit with the inputs `Erase`, `Expose` and `NRE` from the digital circuit. [1]

The circuit is based on a photo-diode, which will generate current. The generated current will depend on the lighting conditions, so that brighter conditions create a higher current and darker conditions create a lower current. This current is used to charge a capacitor, C_S. Since we are charging a capacitor, we can achieve the same charge by exposing small current over a longer time interval, as exposing a higher current for a shorter time interval. Transistors $M1$ and $M2$ are NMOS transistors, and are controlled by logic signals from the digital circuit. While `Expose` is logic high, the capacitor C_S is being charged. While `Erase` is logically high, we discharge the capacitor, and erase our last exposure so that we are ready to charge the capacitor again. In figure 4 you can also see the active load transistors $MC1$ and $MC2$ and its corresponding capacitors.

The circuit in figure 4 has nodes $N1$, $N2$ and $N3$ labeled. When the diode, $PD1$, is exposed to light and the `Expose` signal is set high the current I_{PD1} from the diode will continue to node $N2$. As long at the `Erase` signal is low, it will continue on to $N3$. The ADC can only read two pixels at a time, so the `NRE` signal decides whether the ADC is ready to read the signal or not. This is seen in figure 5.

The most important design decisions in this circuit are the dimensions of the transistors and the sizes of the capacitors. The design specifications are given in 3.2.

2.2.1 Choosing the Capacitors

The effective voltage of a transistor is given by equation (1).

$$V_{eff} = V_{GS} - Vt \tag{1}$$

V_{eff} should be above 100mV, and it is a good rule to keep it above 200 mV [3]. Since $V_{GS} = V_G - V_S$ this can be used to find the voltage over the capacitor, V_S, as shown in section 3.2.

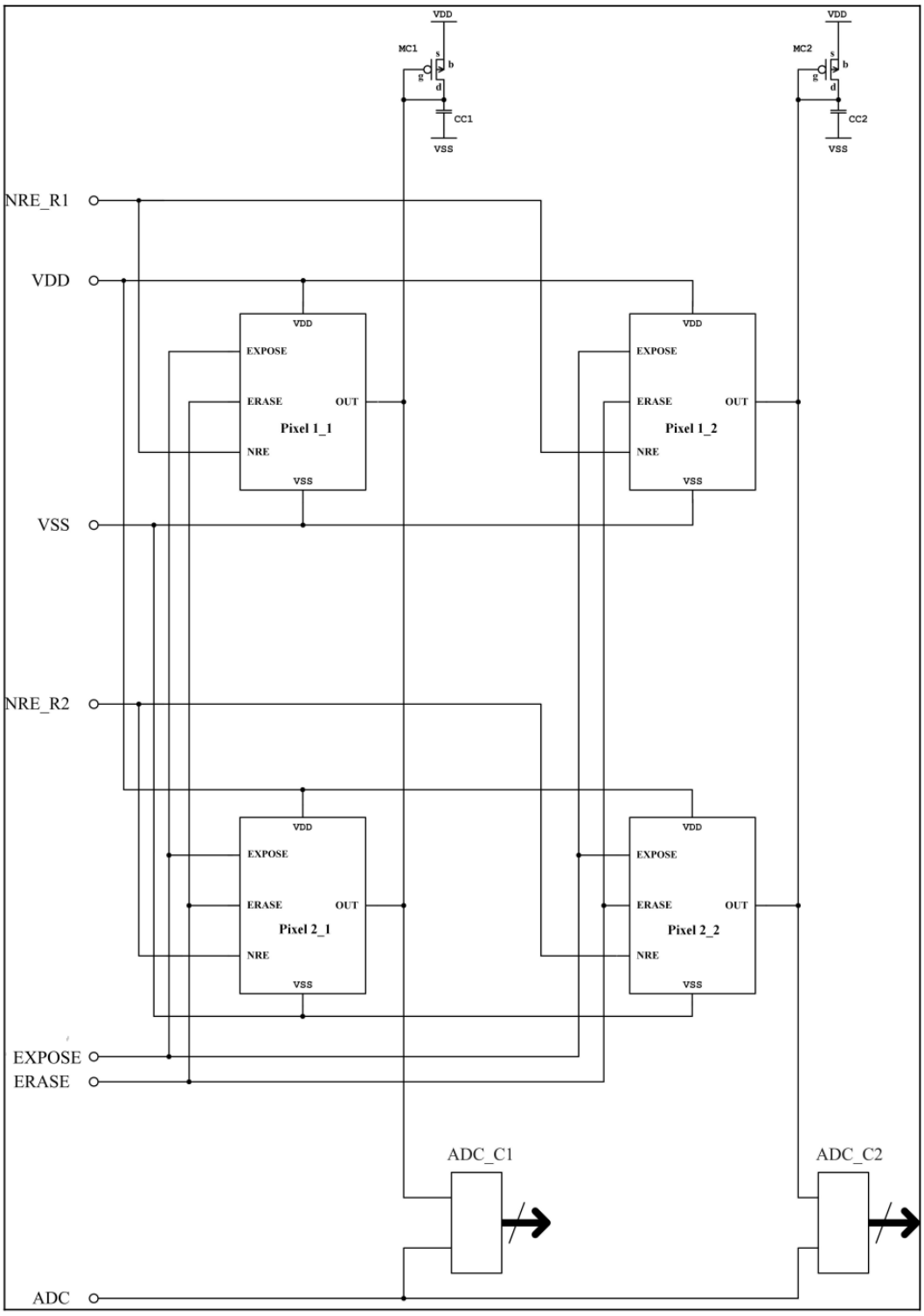

Figure 5: Four pixel array consisting of one pixel circuits, ADC_C1, ADC_C2 and active load transistor [1].

Due to the body-effect[3], which is caused by different voltages between source and bulk, we have to consider some more factors when doing calculations. For instance the voltage V_t is given by equation (2).

$$V_t = V_{t0} + \gamma(\sqrt{V_{SB} + |2\Phi_F|} - \sqrt{|2\Phi_F|}) \tag{2}$$

Equation (2) illustrates that the threshold voltage will be slightly higher with the body effect. The various components of this equations are given in equation (3) and equation (4).

$$\Phi_F = \frac{kT}{q} ln \frac{N_A}{n_i} \tag{3}$$

$$\gamma = \frac{\sqrt{2qN_A K_{Si}\epsilon_0}}{C_{ox}} \tag{4}$$

The other are constants found in table 4 and 5. V_{SB} is the voltage between source and bulk.

$$C = \frac{It}{V} \tag{5}$$

The version of Ohm's law, seen in equation (5), can be used to find the appropriate size for the capacitor C_s, using V_s as the voltage of the capacitor.

The other capacitors are given in the specifications, which can be found in section 3.2.

2.2.2 Choosing Dimensions for the Transistors

The equation for active region drain current is seen in equation (6) for NMOS and equation (7) for PMOS [3].

$$I_D = \frac{1}{2}\mu_n C_{ox} \frac{W}{L}(V_{tn} - V_{GS})^2[1 + \lambda(V_{eff} - V_{DS})] \tag{6}$$

$$I_D = \frac{1}{2}\mu_p C_{ox} \frac{W}{L}(V_{GS} - V_{tn})^2[1 + \lambda(V_{DS} - V_{eff})] \tag{7}$$

The resistance between drain and source, r_{ds} can be related to to the drain current, I_D, by (8) [3].

$$r_{ds} \approx \frac{1}{\lambda I_D} \tag{8}$$

Here we use λ given by (9) for NMOS and (10) for PMOS [3]. Φ_0 is the built in voltage of the junction.

$$\lambda = \frac{1}{L\sqrt{V_{DS} - V_{eff} + \Phi_0}} \tag{9}$$

$$\lambda = \frac{1}{L\sqrt{V_{eff} - V_{DS} + \Phi_0}} \tag{10}$$

Which again gives relation (11).

$$I_D \propto \frac{W}{L} \tag{11}$$

The transistors $M1$, $M2$ and $M4$ work as switches activated by specific signals from the digital circuit.

The buffer should be designed so that the gain is as close to 1 as possible. The important transistors here are $M3$ and MC. Here, the transistor MC is used as an active load. The transistor $M4$ is no concern, since it will work as a switch: either it has very low resistance, like a wire, or it has a very very high resistance, like an open circuit.

To find the gain, A, small signal analysis can be applied. The small signal model with body effect is shown in figure 6 [2].

To combine the current sources use that $g_{m3}V_{sg} + g_{m3}V_{bs} = g_{m3}(V_s - V_g + V_b - V_s) = g_{m3}(V_b - V_g) = g_{m3}V_{bg}$. See also from the figure that by Kirchoff's voltage law: $V_{bg} = V_{sg}$.

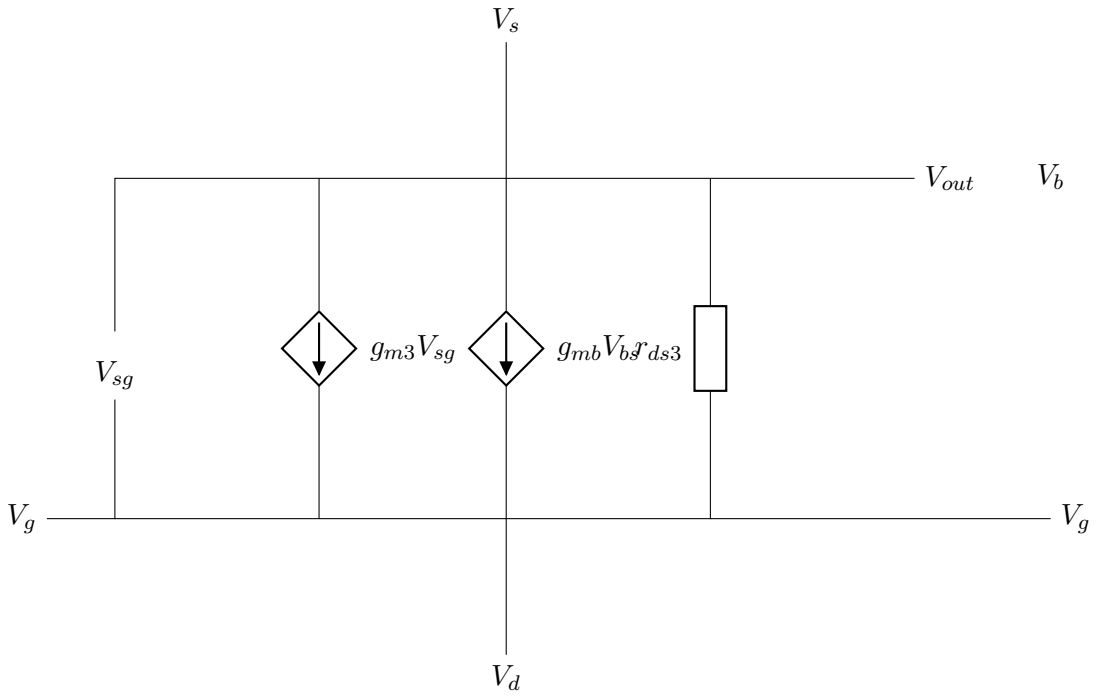

Figure 6: Small signal model for low frequencies. [2]

If one chooses to ignore the body effect, in the case that it is sufficiently small, this can be simplified. Figure 7 shows a simplified version of the model, along with the model for M5 simplified to a resistance. Because this is an active load common gate amplifier, we can simplify the impedance seen at V_{out} at low frequencies to $\approx \frac{1}{g_{mC}}$ [3, p. 126].

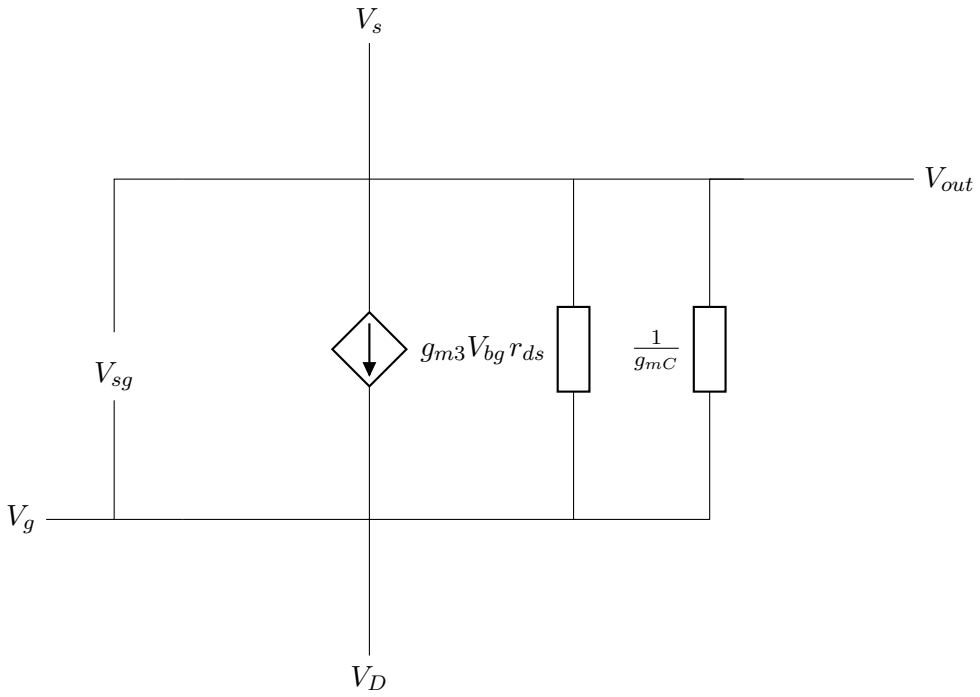

Figure 7: Small signal model for $M3$ and MC with a combined current source.

Knowing that $V_{sg} = V_{bg}$, this is used in further calculations. The output is V_s and given by equation (12).

$$V_{out} = -V_{sg}(r_{ds}||\frac{1}{g_{mC}}) \tag{12}$$

Looking at the model, it also appears that V_g is the same as V_{in}. V_{sg} is given by equation (13).

$$V_{sg} = V_s - V_g = V_{out} - V_{in} \tag{13}$$

The resistance can be assumed to be $r_{sd} >> \frac{1}{g_{mC}}$ [2], which allows for the equation (14).

$$V_{out} = -(V_{out} - V_{in})g_{m3}\frac{1}{g_{mc}} \tag{14}$$

The gain with simplifications is then given by equation (15).

$$
\begin{aligned}
A &= \frac{V_{out}}{V_{in}} \\
&= \frac{g_{m3}\frac{1}{g_{mC}}}{1 + g_{m3}\frac{1}{g_{mC}}} \\
&= \frac{g_{m3}}{g_{m3} + g_{mC}}
\end{aligned}
\tag{15}
$$

The value g_m is given by equation (16) [3].

$$
g_m = \sqrt{\mu C_{ox} I_D \frac{W}{L}} \tag{16}
$$

This formula is the same regardless of which g_m one is looking at, and since all parameters are the same for both transistors the gain can be simplified, as shown in equation (17).

$$
A^2 = \frac{\frac{W}{L}_3}{\frac{W}{L}_3 \frac{W}{L}_C} \tag{17}
$$

To achieve a gain as close as possible to 1, the transistor dimensions should be chosen accordingly.

3 Design

3.1 Digital Circuit Design

The digital circuit has been designed using Verilog and simulated in Active-HDL. The modules were created in separate Verilog modules, and put together in a separate module. All Verilog files can be found in Appendix A.

The specifications for this design is shown in table 1.

Specifications	Value
Clock speed (`Clk`)	$1k$Hz
Exposure time minimum (`EX_time`)	$2ms$
Exposure time maximum (`EX_time`)	$30ms$

Table 1: Specifications for the digital circuit.

3.1.1 Exposure Time Controller

The exposure time controller, `CTRL_ex_time`, controls the exposure time. The lower and upper boundaries for the exposure time is $2ms$ and $30ms$ respectively. The initial value of the exposure time is set in the middle of the boundaries, to $15ms$. Every clock period the signal `exp_inc` or `exp_dec` is logically high, the exposure time is either incremented or decremented respectively, by $1ms$. The exposure time controller also makes sure that the user cannot increase or decrease the exposure time outside of these boundaries. The exposure time, `EX_time`, is sent to the exposure time counter with a 5 bit signal. 5 bits is the least amount of bits to represent all the values within the boundaries. If the `Reset` signal is set logically high, the exposure time is set back it the initial value, $15ms$.

3.1.2 Exposure Time Counter

The exposure time counter receives the 5 bit signal, `EX_time`, with the desired exposure time. When the `Start` signal is set logically high from the finite state machine, the counter begins. The time counter is designed to count down from the exposure time to zero. When the timer hits zero, it will set the overflow signal, `Ovf5`, high for one clock period, to the finite state machine.

3.1.3 Finite State Machine

The finite state machine is the most complex part of the digital circuit. This design has three states for the FSM. The three states are `idle`, `Exposure` and `Readout`. The `idle` state is the state the system is in when it is not in the process of taking a picture. In this state, the analog circuit is being reset, so that the system is ready to take a picture. The `Exposure` state is the state where the system is exposing the analog circuit, so that the pixels can be created. Table 2 shows the characteristic values for the states `idle` and `Exposure`.

Output	Idle	Exposure
NRE_1	1	1
NRE_2	1	1
ADC	0	0
Expose	0	1
Erase	1	0

Table 2: Characteristic output values for the states `idle` and `Exposure`.

`Readout` is a much more complex state. This is the state where the readout from the analog circuit to the ADC happen. In this design, the `Readout` state is nine clock periods long, where one output changes each period. To accomplish this, the FSM has a timer which begins when the FSM enters the `Readout` state. Figure 8 shows a time chart of the desired signal outputs in the different states.

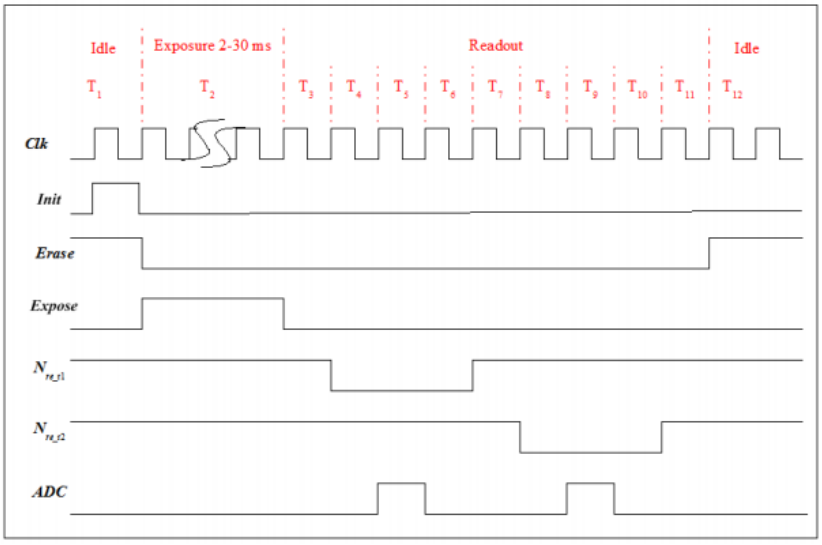

Figure 8: Desired time chart for the digital circuit. [1]

Figure 9 is a state diagram of the states in the finite state machine. The diagram shows what triggers the machine to change between states. When the system is in the `idle` state, the state can be changed to `Exposure` if `init` is set logically high while reset is not. While the system is in the `Exposure` state, the system will stay in this state until `Ovf5` is set logically high. When `Ovf5` is set logically high, the system changes state to the `Readout` state. The system will stay in the `Readout` until readout cycle is done. The diagram also shows that no matter what state the system is in, if `Reset` is set logically high, the system will change state back to `idle`.

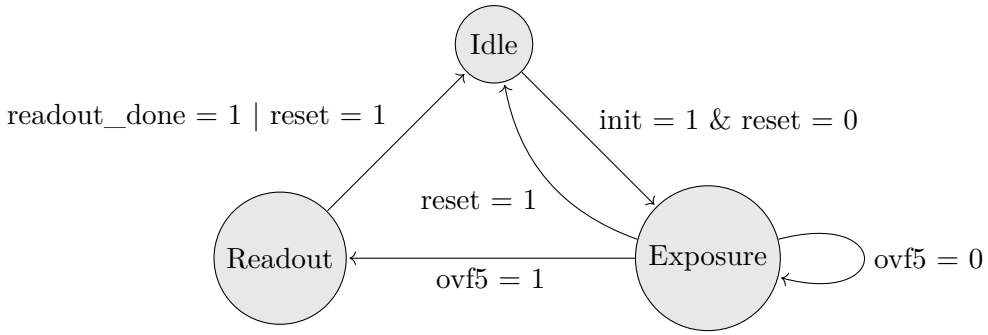

Figure 9: State diagram for the FSM in the digital circuit.

3.2 Analog circuit

The analog circuit can be seen in figure 4. It is created using a photo diode, and depends on the signals `Erase`, `Expose` and `NRE` from the digital circuit. This will give an `Out` signal containing the current from the diode. The current from the diode is based on the amount of light and the exposure time from the `Erase` signal. The `Out` signal is then stored and used in the digital circuit shown in figure 1.

The specification and boundaries for the analog circuit components is shown in table 3. Table 4 shows the specifications for tranistors using 180nm technology.

Specifications	Value
Transistor length	$0.36\mu m < L < 1.080\mu m$
Transistor width	$0.1080\mu m < W < 5.040\mu m$
C_s	$\leq 3pF$
$C_{C1} = C_{C2}$	$3pF$
V_{DD}	$1.8V$
I_{PD1}	$50pA \leq 750pA$

Table 3: Specifications for the circuit [1].

Parameter	Value NMOS	Value PMOS
μC_{ox}	$270\frac{\mu A}{V^2}$	$70\frac{\mu A}{V^2}$
V_{t0}	$0.45V$	$-0.45V$
λL	$0.08\frac{\mu m}{V}$	$0.08\frac{\mu m}{V}$
C_{ox}	$8.5\frac{fF}{\mu V^2}$	$8.5\frac{fF}{\mu V^2}$
t_{ox}	$5nm$	$5nm$
n	1.6	1.7
θ	$1.7\frac{1}{V}$	$1.0\frac{1}{V}$
m	1.6	2.4

Table 4: Parameters for 180 nm transistors [3].

Variable	Symbol	Value
Elementary charge	q	$1.602 \cdot 10^{-19}$
Boltzmann constant	k	$1.38 \cdot 10^{-23} JK^{-1}$
Intrinsic carrier concentration	n_i	$1.1 \cdot 10^{16} carriers m^{-3}$
Concentration of acceptor atoms	N_A	Not given
Temperature (room temp.)	T	$300K$
Electric constant	ϵ_0	$8.854 \cdot 10^{-12} Fm^{-1}$
Dielectric constant	K_{si}	11.8

Table 5: A collection of various constants needed for calculations.

3.2.1 The capacitor Cs

To complete the calculations, constants given in table 5 are useful. Without knowing the acceptor concentration, its hard to calculate the the body-effect constant exactly. However if N_A is assumed to be in the order of $10^{16} m^{-3}$, which is a typical value, (4) can be used to calculate the body-effect constant to be $\gamma \approx 4.73 \cdot 10^{-8}$. This is very small, and the body effect can be assumed to be negligible.

As mentioned in section 2.2.1, V_{eff} was chosen to be $200mV$, and assumed $V_{t0} \approx V_t \approx 0.45V$. The upper case for for V_G, when `Expose` is high, equals $V_{DD} = 1.8V$. Using (1), this gives $V_S = 1.15V$.

The minimal current, $50pA$, is used for the maximum exposure time, which is $30ms$ and the maximum current, $750mA$, is used for minimum exposure time which is $2ms$.

This gives equation 18.

$$It = 50 \cdot 10^{-12}A \cdot 30 \cdot 10^{-3}s = 750 \cdot 10^{-12}A \cdot 2 \cdot 10^{-3}s = 1.5 \cdot 10^{-12}C \tag{18}$$

Equation (5) from section 2.2.1 is used to get $C_s \approx 1.3045 \cdot 10^{-12}$. $C_s = 1.3pF$ is chosen for this design.

3.2.2 Transistor dimensions

Since the transistors $M1$ and $M4$ are used as switches, minimizing the voltage drop over these is desired. This can be done by choosing wider and narrow as possible. From (8), r_{ds} is smallest for a high current, I_D, which again allows us to use the relationship in (11). Using table 3, $W_{1,4} = 5.040\mu m$ and $L_{1,4} = 0.36\mu m$ can be chosen.

It is desired to minimize the leakage current over transistor $M2$. Equation (11), shows that it is desired to have a long and narrow transistor. Therefore $W_2 = L_2 = 1.080\mu m$ are great values..

When choosing $M3$ and MC, equation (17) can be used. Because the body-effect is so small, figure 6 can be used, and the calculations are simplified. To make the gain as close to 1 as possible, dimensions $\frac{W}{L}_5$ should be as small as possible and $\frac{W}{L}_3$ should be as large as possible. Thus $W_3 = 5.04\mu m, L_3 = 0.36\mu m, W_C = 1.08\mu m$ and $L_C = 1.08\mu m$ are chosen. A collection of the transistor dimensions can be found in table 6.

—	Width	Length
M1	$5.04\mu m$	$0.36\mu m$
M2	$1.08\mu m$	$1.08\mu m$
M3	$5.04\mu m$	$0.36\mu m$
M4	$5.04\mu m$	$0.36\mu m$
MC	$1.08\mu m$	$1.08\mu m$

Table 6: Chosen transistor values

4 Simulations

Both the digital and analog circuit has been tested through simulations to check if they behave as planned. The digital circuit has been simulated in Active-HDL, while the analog circuit

has been simulated in AIM-Spice.

4.1 Simulations from Active-HDL

The circuit has been simulated using the hardware description language Verilog in the program Active-HDL. The code used to simulate the circuit can be found in appendix A. The system has been tested with different inputs to check the behaviour of the system. Figure 10 shows simple test of the whole system at the initial values. The test shows that the system starts off in idle, and goes through the whole cycle, from exposure to readout, then back to idle, when init is pressed.

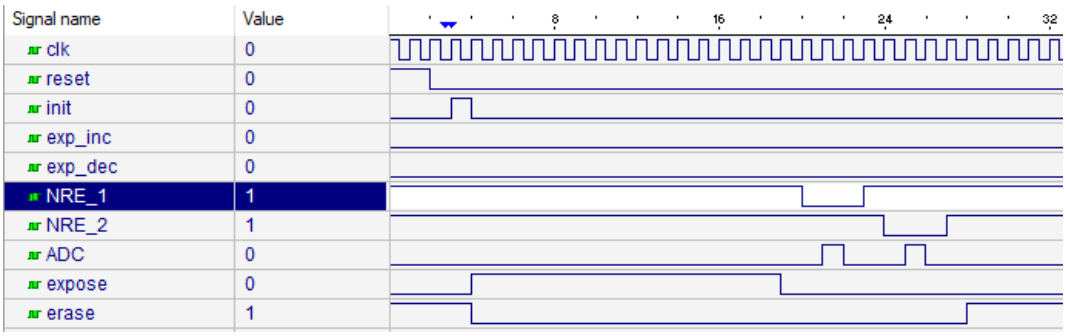

Figure 10: Test of one cycle. Going from idle to expose, then to readout, and back to idle.

Figure 11 shows a test of the lower boundary for the exposure time. Lower boundary is 2ms, so the system was tested for lowering the exposure time below this. The test shows that the exposure time is set to the lower boundary.

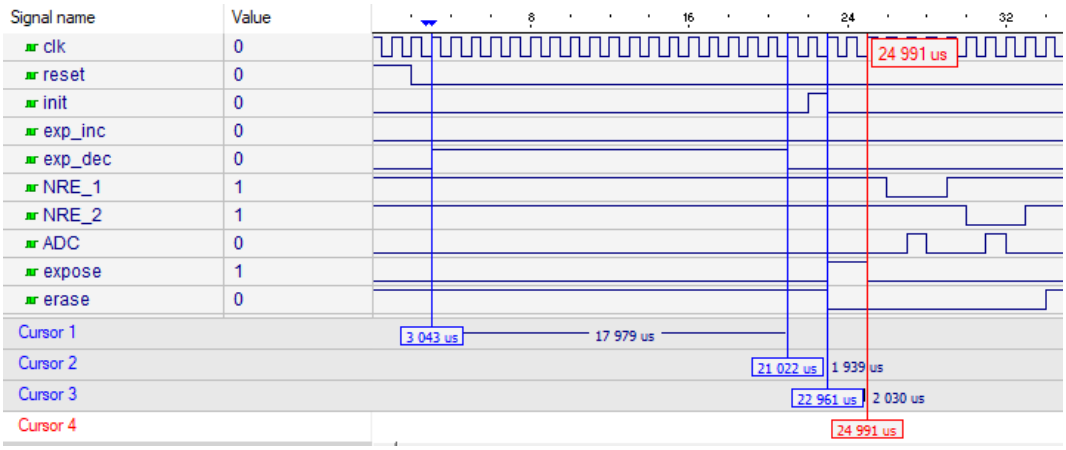

Figure 11: Test of the lower limit of exposure time. Here we try to decrease it to -2ms, but the exposure time is set to 2 as planned.

Figure 12 shows a test of the upper boundary for the exposure time. Upper boundary is 30ms, so the system was tested for increasing the exposure time over this limit. The test

shows that the exposure time is set to the upper boundary.

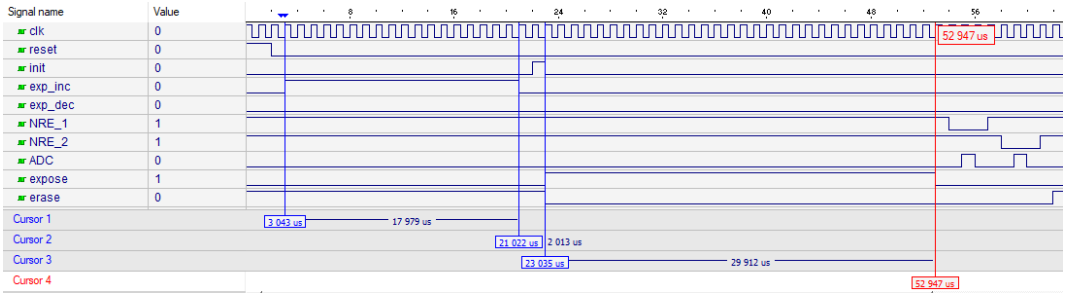

Figure 12: Test of the upper limit of exposure time. Here we try to increase it to 33ms, but the exposure time is set to 30 as planned.

Figure 13 shows a test of both exposure increase and decrease pressed at the same time. The test shows that the exposure time will not change when both are pressed at the same time.

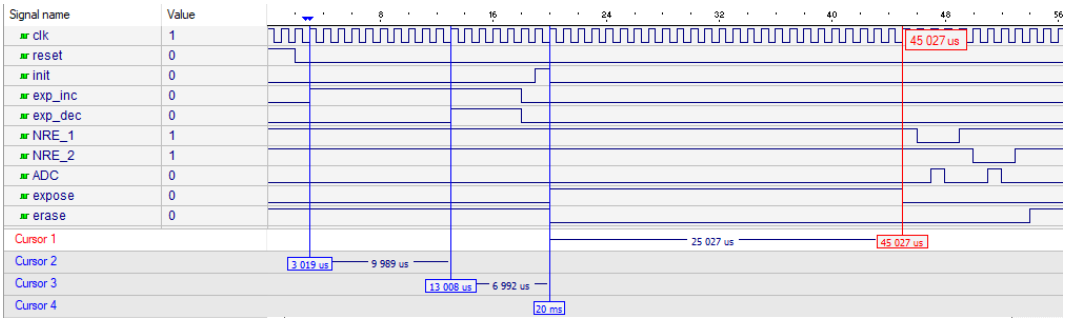

Figure 13: Test of both exposure increase and decrease pressed at the same time. The circuit behaves as planned, by not changing the exposure time when both are pressed at the same time.

Figure 14 shows a test of the reset button in the middle of the readout state. The cycle is stopped immediately, and all values are set back to idle mode.

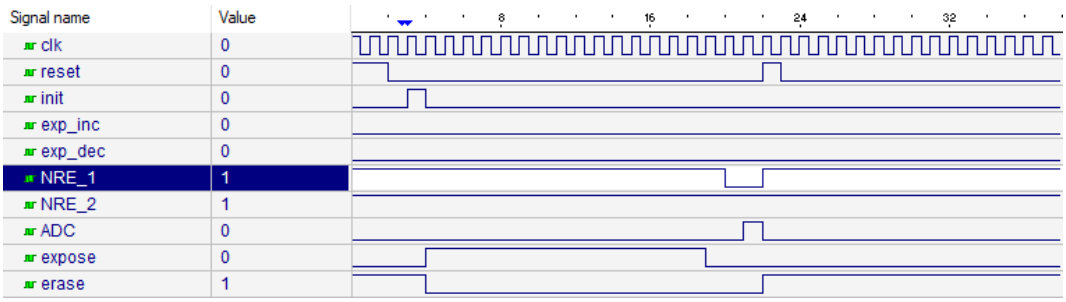

Figure 14: Test of the reset function, here shown in the middle of the readout state.

4.2 Simulations from AIM-Spice

The transistor and capacitor sizes were simulated in AIM Spice for both the highest possible diode current, 750 mA, and for the shortest, 50 mA. In figures 15, 21, 17, 18, and 19 all the different simulations are shown for the lowest current, 50 mA, combined with the longest possible exposure time.

The two out voltage's, `V(out1)` and `V(out2)` are signals for `ADC_C1` and `ADC_C2`. The two signals are pretty much identical, but we do two reading of two signals which again equals four pixels in total.

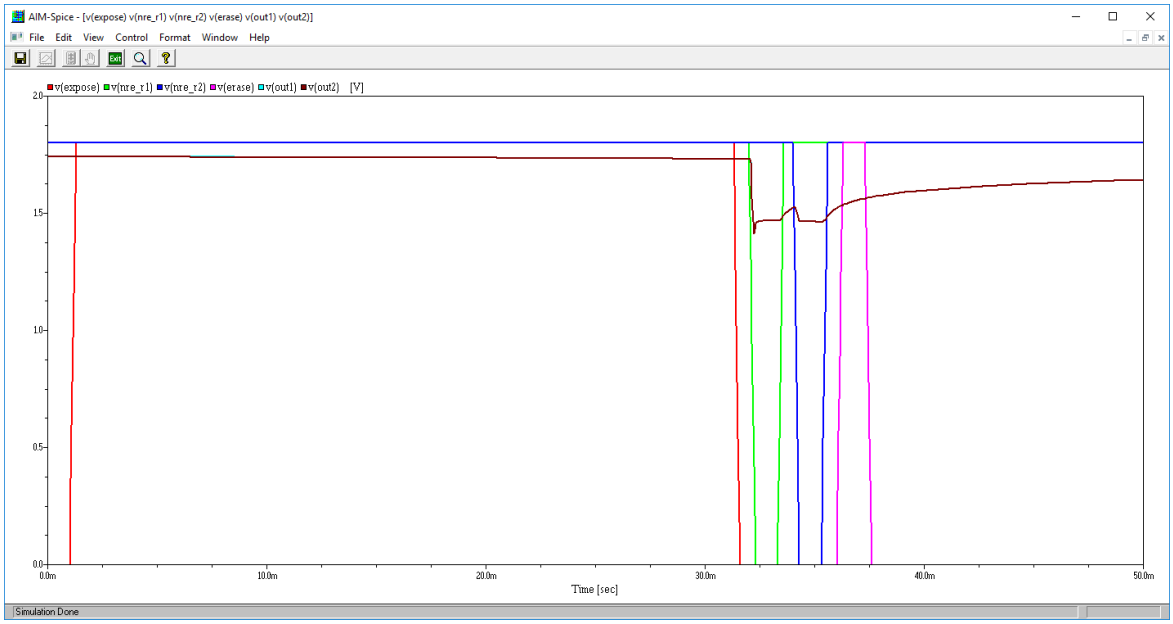

Figure 15: All logic values, `V(out1)` and `V(out2)` shown for 0.05s with 50mA from PD1.

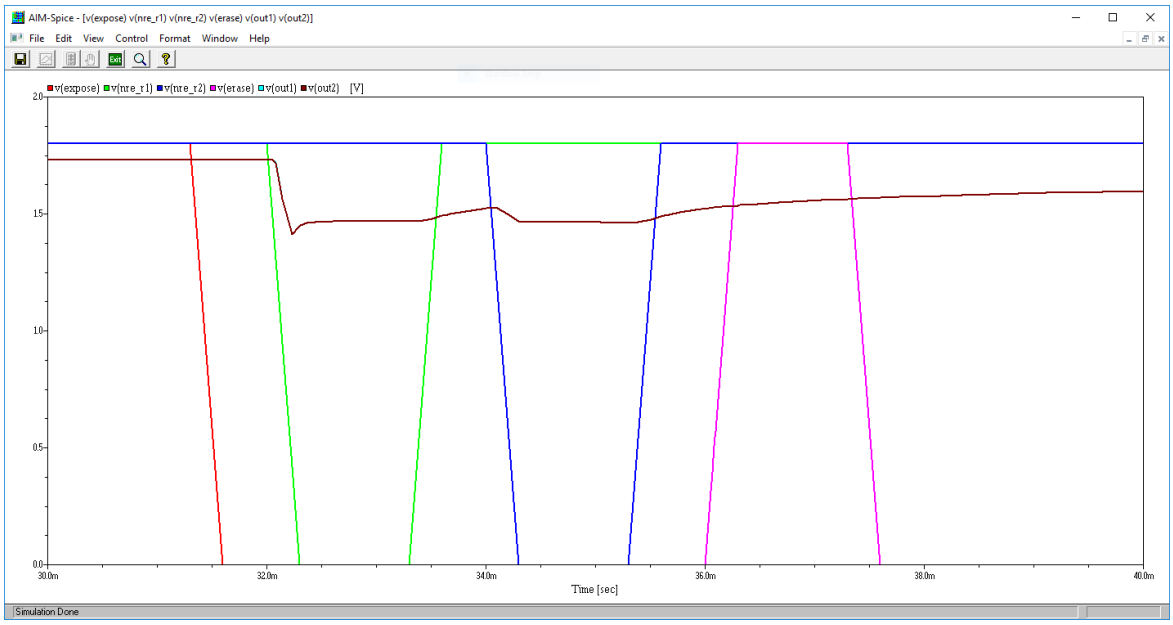

Figure 16: All logic values, `V(out1)` and `V(out2)` shown for 0.01s with 50mA from PD1.

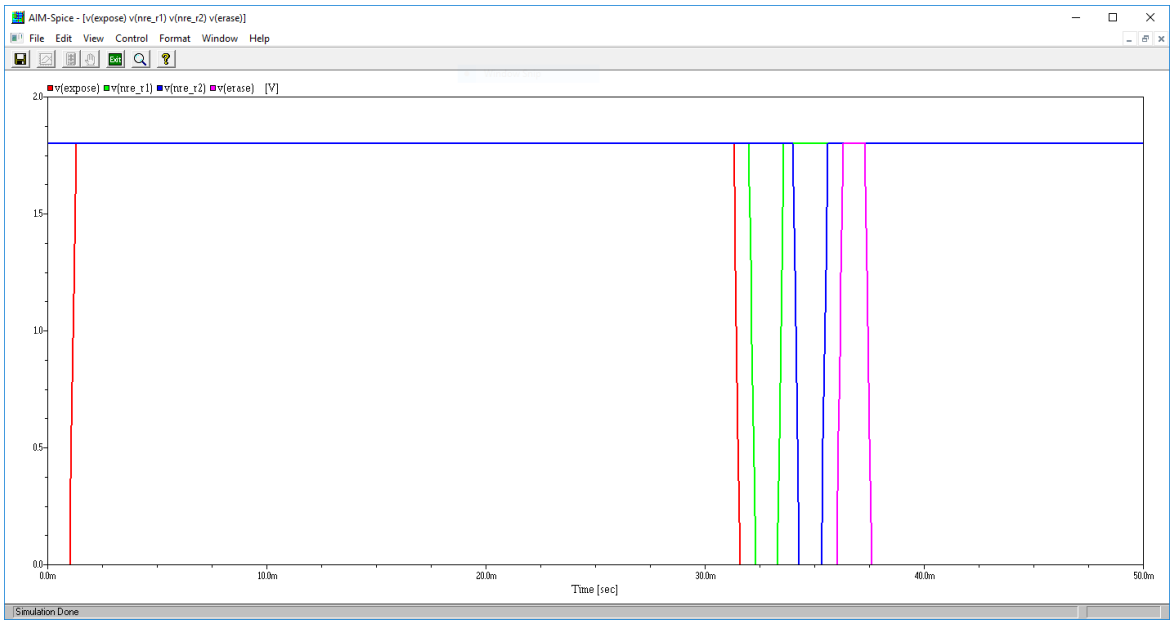

Figure 17: All logic values shown for 0.03s with 50mA from PD1.

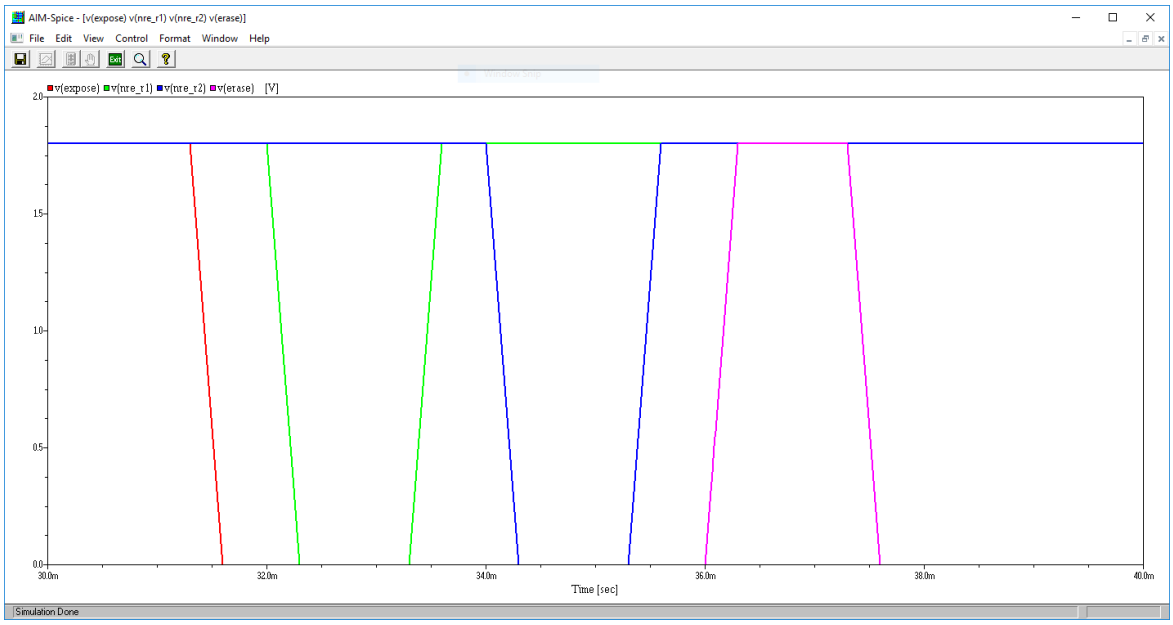

Figure 18: All logic values shown for 0.01s with 50mA from PD1.

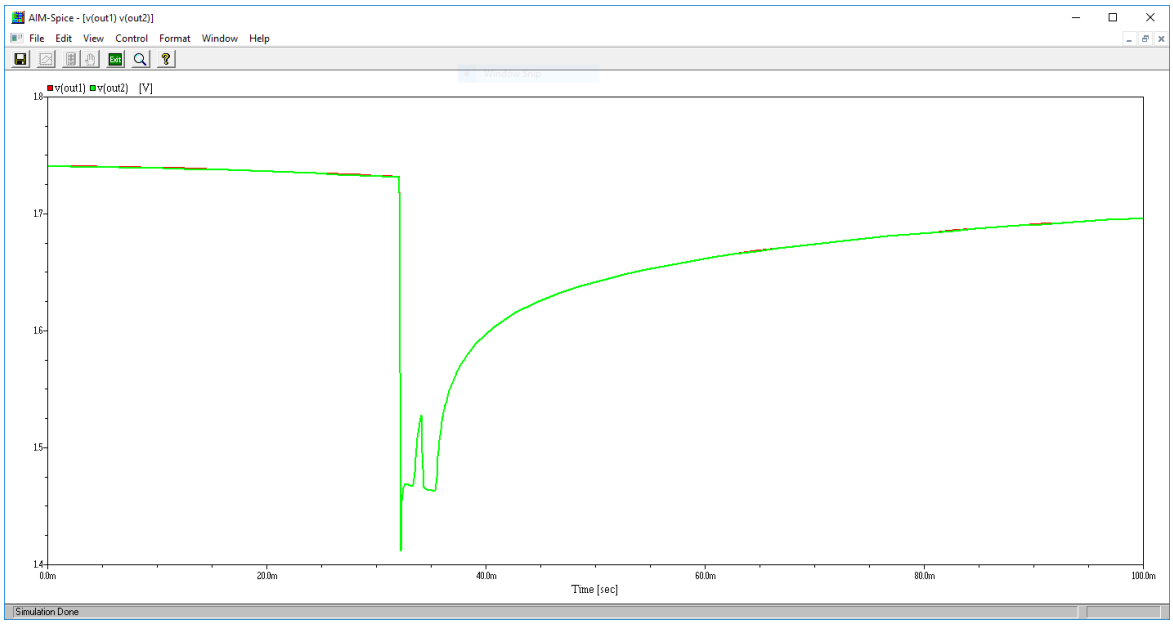

Figure 19: V(out1) and V(out2) shown for 0.1s with 50mA from PD1.

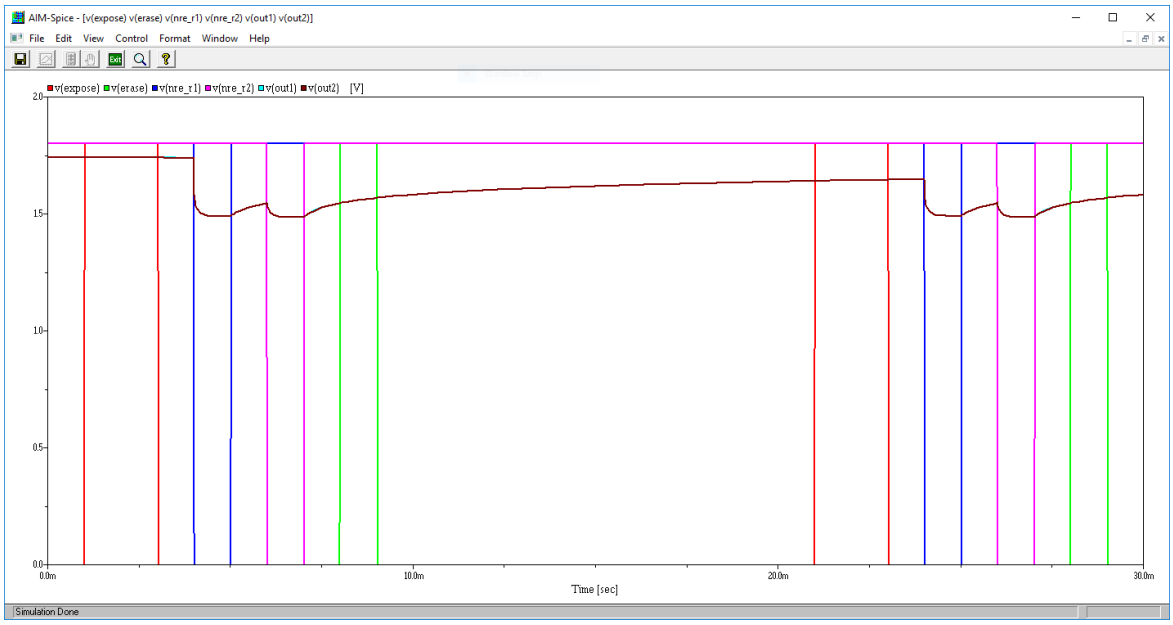

Figure 20: All logic values and `V(out)` shown for 0.03s with 750mA from PD1.

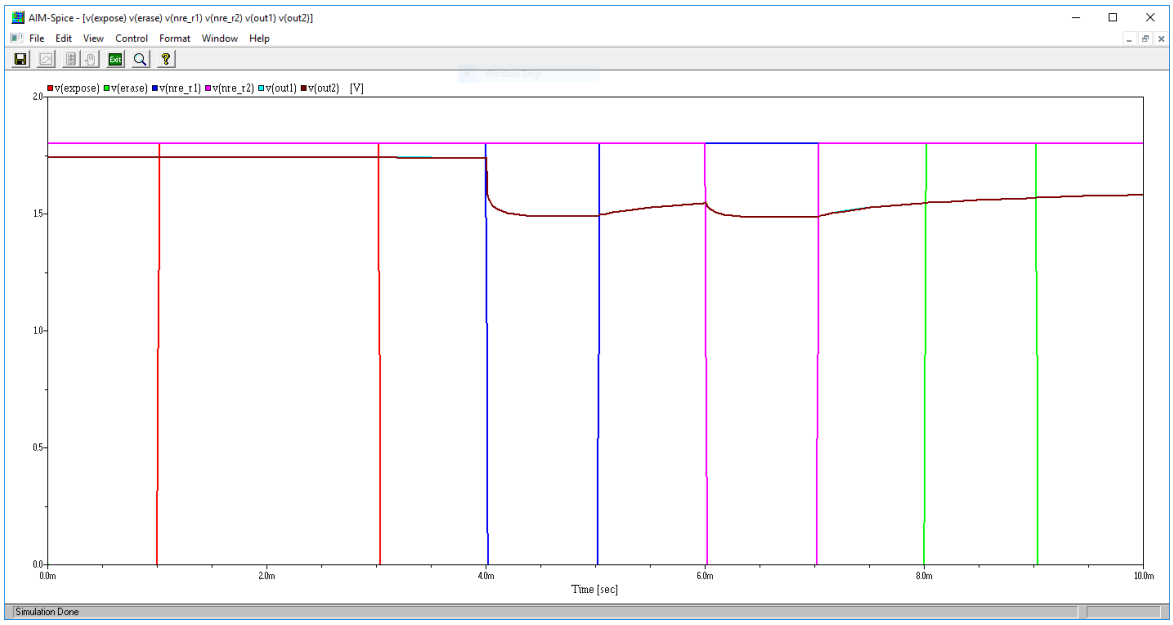

Figure 21: All logic values, `V(out1)` and `V(out2)` shown for 0.01s with 750mA from PD1.

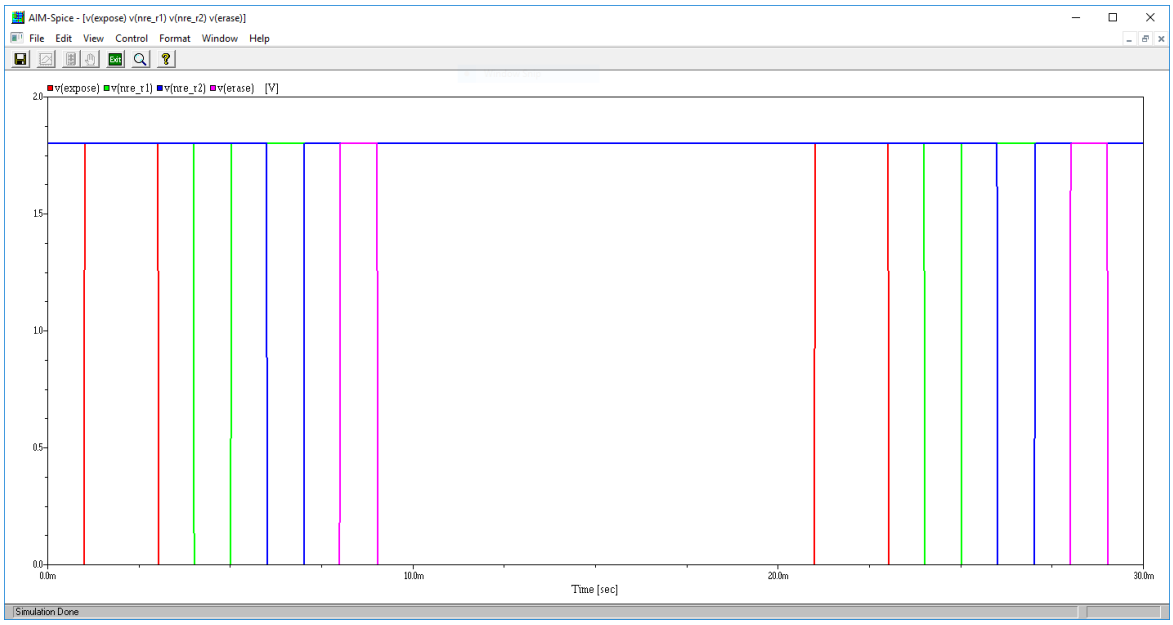

Figure 22: All logic values shown for 0.03s with 50mA from PD1.

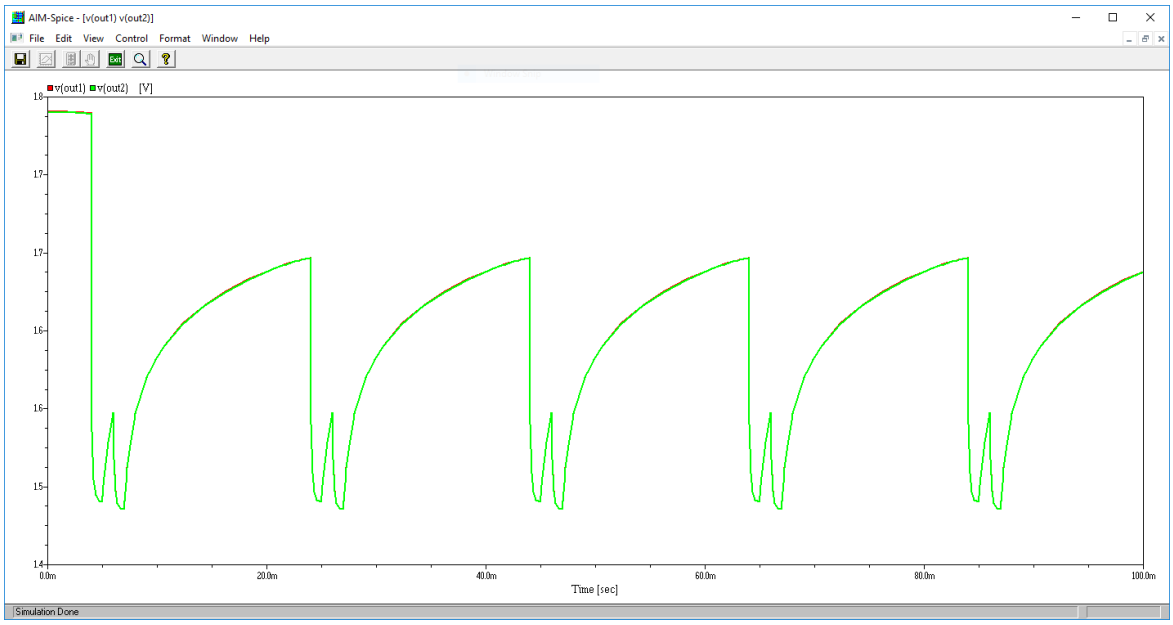

Figure 23: `V(out1)` and `V(out2)` shown for 0.1s with 750mA from PD1.

The inclusion of many graphs show of the circuit in various conditions, such as shorter and longer time periods to better illustrate its behavior.

5 Results and Discussion

5.1 Digital Circuit

The system has been designed and thoroughly tested. The test of both the lower and upper limits shows that the system will not set the exposure time lower or higher than the boundaries it is given. The tests also showed that if both exposure decrease and increase is pressed at the same time, the system behaves as it should and leaves the value as it is. The test of the reset function also behaves as planned. Reset cancels the cycle immediately, so that the system goes back to idle state.

The results from the simulations and testing of the digital system show that the circuit behaves as planned. The digital circuit therefore successfully acts as the digital component of a four pixel digital camera.

5.2 Analog Circuit

The analog circuit works with the digital circuit, and the figures 18, 17 an 22 are included to show the logic applied to the circuit during the simulation. This is a model for the behavior of the digital circuit. The analog circuit simulations show that the circuit works as it should. Since the circuit should be designed for different exposure times and diode currents the maximum and minimum values were tested. In figure 23 the voltage out stabilizes at a value slightly under 1.7 V between each picture. This indicates that the chosen, optimized with the specifications, values work for the purpose. Once the circuit enters exposure, the sequence will continue, thus we did not anticipate the need to test other signals than the ones tested.

6 Conclusion

This report has described how to create a four pixel camera. The report describes how to create a digital circuit, including using a finite state machine, to control an analog circuit. The digital circuit has been simulated using the hardware descriptive language Verilog in the program Active-HDL. The simulations show that the digital circuit behaves as expected. The boundaries of the digital circuit has been tested by trying to exceed the limits of exposure time, in figures 12 and 11 where it was shown that the system is capable of keeping its variables within the given boundaries. A test of trying to increase and decrease the exposure time at the same time was also done, figure 13, to show that the system could handle it. The system was also tested with a reset in the middle of the cycle, figure 14, where it proved to be able to cancel a picture, and return to its idle state.

This report has also described how to create the analog circuit for a pixel camera. Choosing transistor dimensions and capacitor size is an important part of this circuit, and this has been thoroughly described. Simulations has been done in AIM-Spice to show how the system works, based on the input from the digital system. Simulations are useful, because it is easy

to see how the circuit will behave over long and short periods of time. In the simulation upper and lower boundaries were tested, which showed that the circuit worked for both cases. It also showed that although there were some attenuation of the signal initially, it stabilized quickly and would be predictable for the digital designer to work with.

The successful simulations and tests conclude that this report successfully describes the analog and digital circuits needed to create a simple four pixel digital camera.

References

[1] Bjørn B. Larsen. *Project Description – Digital Cameera*. 2019.

[2] *Microelectronic Devices and Circuits*. URL: https://inst.eecs.berkeley.edu/~ee105/fa14/lectures/Lecture13-Small%20Signal%20Model-MOSFET.pdf.

[3] Kenneth W. Martin Tony Chan Carusone David A. Johns. *Analog Integrated Circuit Design*. 2nd ed. John Wiley & Sons, Inc, 2012.

A Verilog Listing

Listing 1: Verilog code for module Timer_counter

```
1  `timescale 1 ms / 1 us // make sure the same timescale is used in
       the whole system
2
3  module Timer_counter (input wire clk, reset, start, input reg [4:0]
       ex_time, output reg ovf5);
4      reg  [4:0] timer;  // timer for exposure time
5      reg keep_counting; // to check if we are in the expose time
6      always @(posedge clk) begin
7          // reset everything
8          if (reset) begin
9              timer <= 0;
10             ovf5 <= 0;
11             keep_counting = 0;
12         end // if reset
13
14         // start will set the timer and that we are in the expose
               time
15         else if (start) begin
16             timer <= ex_time - 1; // set timer to expose time, and
                   subtract one because one cycle has already been done
                    when this is started
17             keep_counting <= 1; // we are in the expose time
18         end // if start
19
20         // count down if we are not finished exposing and still in
               exposure time
21         else if(timer != 0 && keep_counting) timer <= timer - 1;
22
23         // if we are in the exposed time and want to end it
24         else if (timer == 0 && keep_counting) begin
25             ovf5 <= 1; // sent the signal that we are done exposing
26             keep_counting <= 0; // we are no longer in expose time
27         end // if timer
28     end // always
29 endmodule // Timer_counter
```

Listing 2: Verilog code for module CTRL_ex_time

```
1  `timescale 1 ms / 1 us // make sure the same timescale is used in
       the whole system
2
3  module CTRL_ex_time (input wire exp_inc, exp_dec, clk, reset,
       output reg [4:0] ex_time);
4      always @(posedge clk) begin
5          // set the standard exposure time to 15ms
6          if (reset) ex_time <= 15;
7
8          // if both buttons are pressed, the exposure time remains
               unchanged
9          if (exp_inc && exp_dec) ex_time <= ex_time;
10
11         // increase by 1ms if exp_inc is pressed and we are under
               our maximum exposure time of 30ms
12         else if (exp_inc & ex_time<30) ex_time <= ex_time+1;
13
14         // decrease by 1ms if exp_dec is pressed and we are over
               our minimum exposure time of 2ms
15         else if (exp_dec & ex_time>2) ex_time <= ex_time-1;
16
17     end // always
18 endmodule // CTRL_ex_time
```

Listing 3: Verilog code for module FSM_ex_control

```
1
2  `timescale 1ms / 1us // make sure the same timescale is used in the
       whole system
3
4  module FSM_ex_control(input reg init, clk, reset, ovf5, output reg
       NRE_1, NRE_2, ADC, expose, erase, start);
5      parameter [1:0] idle = 2'b00;
6      parameter [1:0] exposure = 2'b01;
7      parameter [1:0] readout = 2'b10;
8
9      reg [1:0] state, next_state; // used to change states
10     reg [4:0] read_counter; // counter for readout state
11
12     // clock signal
13     always @(posedge clk) begin
14
15         // reset should change state to idle
16         if (reset) begin
17             state <= idle; // change state to idle
18             read_counter <= 0; // reset counter for readout state
19         end // if reset
20
21         // change state
22         else state <= next_state;
23
24         // count up while in the readout state
25         if (state == readout) read_counter <= read_counter + 1;
26     end // always clk
27
28     // Finite state machine
29     always @ (*)  begin
30         case (state)
31             // idle state
32             idle : begin
33                 // set characteristic values for this state
34                 erase <= 1;
35                 expose <= 0;
36                 NRE_1 <= 1;
37                 NRE_2 <= 1;
38                 ADC <= 0;
39                 start <= 0;
40
41                 // init begins the expose-readout cycle
42                 if (init) begin
```

```
43                      // set next state and initial values for
                           nextstate
44                      next_state <= exposure;
45                      start <= 1;
46                      expose <= 0;
47                      erase <= 1;
48                  end // if init
49                                          // if nothing is pressed, stay
                                               in idle
50                  else next_state <= state;
51              end // idle state
52
53              //exposure state
54              exposure : begin
55                  // characteristic values for exposure state
56                  erase <= 0;
57                  start <= 0;
58                  expose <= 1;
59
60                  // overflow signal (ovf5) signals that exposure
                       time is done
61                  if (ovf5) begin
62                      next_state <= readout;  // change state to
                           readout
63                      expose <= 0; // turn off exposure
64                  end // if ovf5
65
66                  // stay in exposure if ovf5 signal is not recieved
67                  else next_state <= exposure;
68              end // exposure state
69
70              // readout state
71              readout : begin
72                  // read_counter is started and changes the values
                       for each clock cycle
73                  if (read_counter == 0) NRE_1 <= 0;
74                  if (read_counter == 1) ADC <= 1;
75                  if (read_counter == 2) ADC <= 0;
76                  if (read_counter == 3) NRE_1 <= 1;
77                  if (read_counter == 4) NRE_2 <= 0;
78                  if (read_counter == 5) ADC <= 1;
79                  if (read_counter == 6) ADC <= 0;
80                  if (read_counter == 7) NRE_2 <= 1;
81
82                  // cycle completed
83                  if (read_counter == 8) begin
84                      next_state <= idle; // go back to idle
```

```
85                          erase <= 1; // erase right away, so the system
                                is ready for another picture
86                      end // if read_counter
87                  end // readout state
88              endcase // FSM cases
89          end // always FSM
90  endmodule //FSM_ex_control
```

Listing 4: Verilog code for module RE_control

```
1  `timescale 1 ms / 1 us // make sure the same timescale is used in
       the whole system
2
3  module RE_control ( input reg init , exp_inc, exp_dec, clk , reset
       , output reg NRE_1 , NRE_2 , ADC , expose , erase );
4      reg ovf5 , start ; // overflow and start signal
5      reg [4:0] ex_time ;  // exposure time
6
7      // Exposure time controller
8      CTRL_ex_time Ctrl_ex ( exp_inc, exp_dec, clk , reset , ex_time
           );
9
10     // Exposure time counter
11     Timer_counter Timer (clk , reset , start, ex_time , ovf5);
12
13     // Finite state machine
14     FSM_ex_control FSM (init , clk , reset , ovf5, NRE_1 , NRE_2,
           ADC, expose , erase , start );
15 endmodule // RE_control
```

B AIM-Spice Listing

Listing 5: Code used for simulations in AIM-Spice

```
Analog Circuit

!------------Sizes of transistors and capacitors------------
.param Wmax={5.04u}
.param Lmax={1.08u}
.param Wmin={1.08u}
.param Lmin={0.36u}
.param Cs={1.3p}


.param VDD = 1.8 ! Supply voltage
.param Ipd_1 = 50p ! Photodiode current, range [50 pA, 750 pA]
.param EXPOSURETIME = 30m ! Exposure time, range [2 ms, 30 ms]
.param TRF = {EXPOSURETIME/100} ! Risetime and falltime of EXPOSURE and ERASE
   signals
.param PW = {EXPOSURETIME} ! Pulsewidth of EXPOSURE and ERASE signals
.param PERIOD = {EXPOSURETIME*10} ! Period for testbench sources
.param FS = 1k; ! Sampling clock frequency
.param CLK_PERIOD = {1/FS} ! Sampling clock period
.param EXPOSE_DLY = {CLK_PERIOD} ! Delay for EXPOSE signal
.param NRE_R1_DLY = {2*CLK_PERIOD + EXPOSURETIME} ! Delay for NRE_R1 signal
.param NRE_R2_DLY = {4*CLK_PERIOD + EXPOSURETIME} ! Delay for NRE_R2 signal
.param ERASE_DLY = {6*CLK_PERIOD + EXPOSURETIME} ! Delay for ERASE signal

VDD VDD 0 dc VDD


VEXPOSE EXPOSE 0 dc 0 pulse(0 VDD EXPOSE_DLY TRF TRF EXPOSURETIME PERIOD)
VERASE ERASE 0 dc 0 pulse(0 VDD ERASE_DLY TRF TRF CLK_PERIOD PERIOD)
VNRE_R1 NRE_R1 0 dc 0 pulse(VDD 0 NRE_R1_DLY TRF TRF CLK_PERIOD PERIOD)
VNRE_R2 NRE_R2 0 dc 0  pulse(VDD 0 NRE_R2_DLY TRF TRF CLK_PERIOD PERIOD)


XPIXEL11 VDD EXPOSE ERASE NRE_R1 OUT1 PIXEL
XPIXEL12 VDD EXPOSE ERASE NRE_R1 OUT2 PIXEL
XPIXEL21 VDD EXPOSE ERASE NRE_R2 OUT1 PIXEL
XPIXEL22 VDD EXPOSE ERASE NRE_R2 OUT2 PIXEL
XBUFFER1 OUT1 VDD BUFFER
XBUFFER2 OUT2 VDD BUFFER


.plot V(OUT1) V(OUT2)
.plot V(EXPOSE) V(NRE_R1) V(NRE_R2) V(ERASE)
.plot V(EXPOSE) V(NRE_R1) V(NRE_R2) V(ERASE) V(OUT1) V(OUT2)


!---------------Photodiode
.subckt PhotoDiode VDD N1_R1C1
I1_R1C1  VDD   N1_R1C1   DC  Ipd_1
d1 N1_R1C1 vdd dwell 1
.model dwell d cj0=1e-14 is=1e-12 m=0.5 bv=40
Cd1 N1_R1C1 VDD 30f
```

```
52 .ends
53
54 !---------------One pixel circuit
55 .subckt PIXEL VDD EXPOSE ERASE NRE OUT
56 XPD1 VDD N1 PhotoDiode
57 MN1 N1 EXPOSE N2 0 NMOS W=Wmax L=Lmin
58 MN2 N2 ERASE 0 0 NMOS W=Wmin L=Lmax
59 CS N2 0 Cs
60 MP3 0 N2 N3 VDD PMOS W=Wmax L=Lmin
61 MP4 N3 NRE OUT VDD PMOS W=Wmin L=Lmax
62 .ends
63
64 !---------------Buffer
65 .subckt BUFFER OUT VDD
66 MC1 OUT OUT VDD VDD PMOS W=Wmin L=Lmax
67 CC1 OUT 0 3p
68 .ends
69
70 .include p18_cmos_models.inc
71 .include p18_model_card.inc
```