The goal of this project is to design a chip that could control a bicycle taillight to produce an apparently random flash sequence. The chip should operate from a 3 V supply (two AA batteries in series).

The heart of the chip will be a 5-bit pseudorandom sequence generator (PRSG). This device produces a bit stream that appears random, but repeats after $2^5 - 1 = 31$ bits. A basic block diagram for the PRSG is shown in Fig. 1. PHI is an input clock.

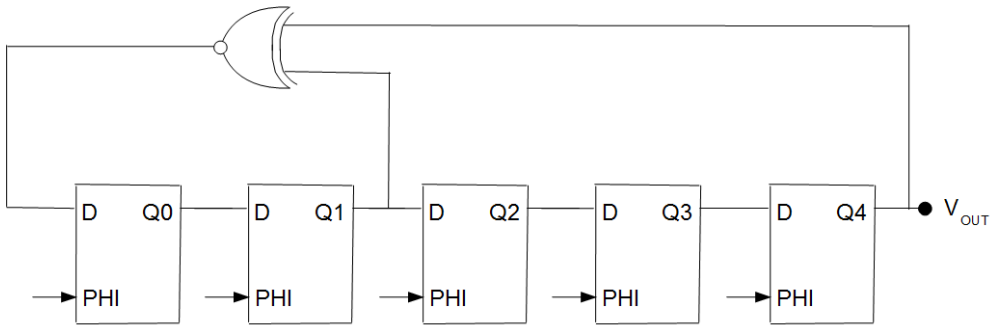

Fig. 1 Pseudorandom sequence generator implemented with shift register and feedback.

The PRSG contains five D flip-flops cascaded to make a shift register. The individual D-flip-flops are to be made using 2-phase dynamic logic. A possible circuit for such a dynamic D-flip-flop is shown in Fig. 2. φ_1 and φ_2 are two-phase non-overlapping clocks. The D flip-flop samples input data when clock φ_1 is high and transfers data to the output when φ_2 goes high.

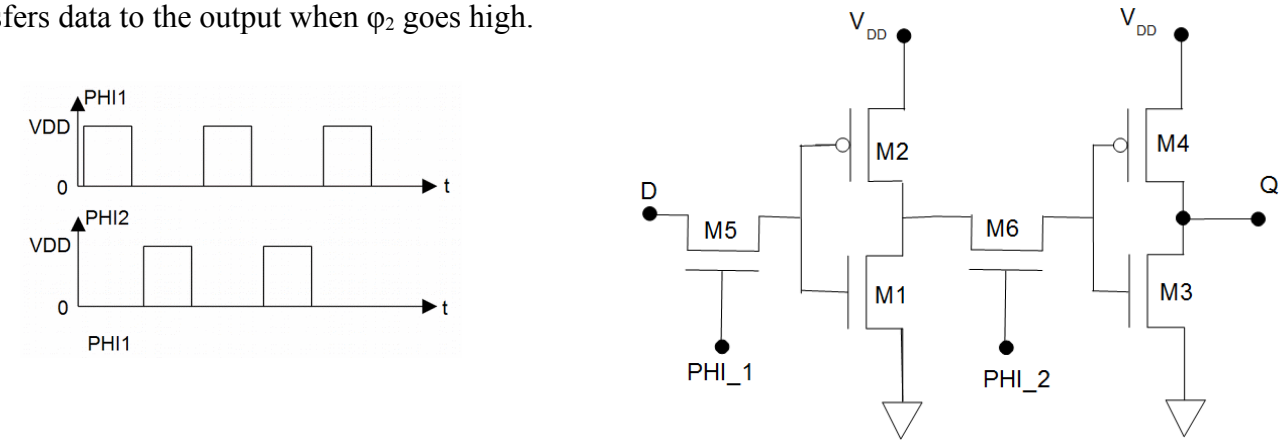

Fig. 2 Shift register stage implemented in 2-phase dynamic logic

The D flip-flop of Fig. 2 stores information on the gate-channel capacitance of the transistors in the inverter stages. The length of time this charge is stored determines how long the flip-flop can hold data. For an SOI implementation the charge storage time is determined by off-state leakage through the pass gates M5 and M6 connecting the output of one inverter to the input of the next stage. Usually one would use full CMOS transmission gates to connect the inverters. In the 4609 CMOS technology the nMOS transistors are likely to have lower off-state leakage than pMOS transistors, so here we will use only nMOS pass gates. Even with nMOS-only pass gates, leakage along the channel edge may be a problem. This problem can be minimized by using a relatively long MOSFET with a double T-gate connection, as shown in Fig. 3. An example layout for this double T-gate MOSFET is included in the *cmos2017.tdb* set-up file for LEdit. T-gate operation will be explained in class. I would like to see some groups try double T-gate MOSFETs for the pass gates, and other groups basic nMOSFETs.

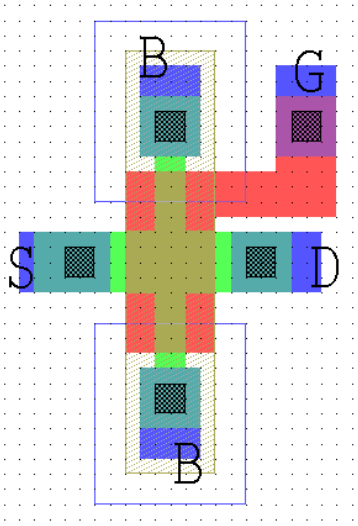

Fig. 3 Double T-gate nMOS layout for nMOS transmission gates M5 and M6 in Fig. 2.

The PRSG requires an XNOR logic block. Baker's textbook (and other sources) present some possible schematics. One simple possibility is shown in Fig. 4.

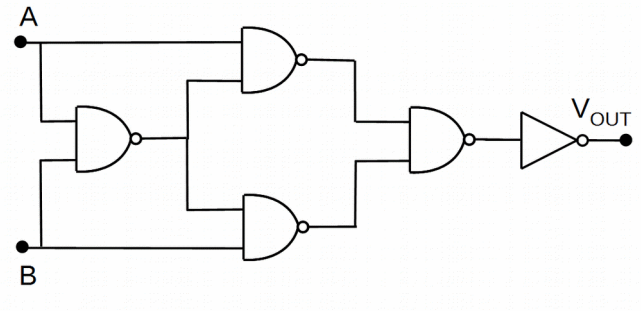

Fig. 4 Simple static XNOR

The PRSG of Fig. 1 can enter a "locked" state if every register holds a logic 1. To avoid entering this state, the PRSG needs a reset. One way to implement the reset is to add an extra input to the XNOR gate capable of forcing its output low, as shown in Fig. 5.

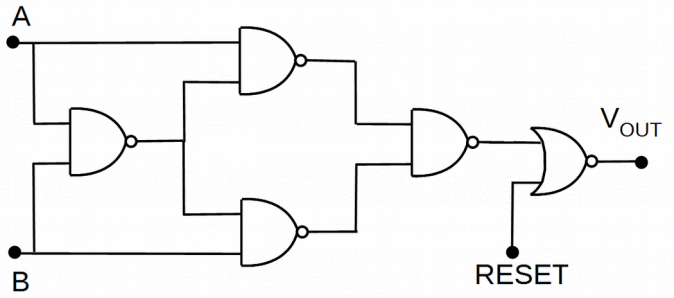

Fig. 5 XNOR with reset

2-phase non-overlapping clocks φ_1 and φ_2 are required to drive the shift register. These clocks can be generated on chip using the circuit of Fig. 6. The inverters at the output of the NAND gates ensure that φ_1 and φ_2 do not overlap.

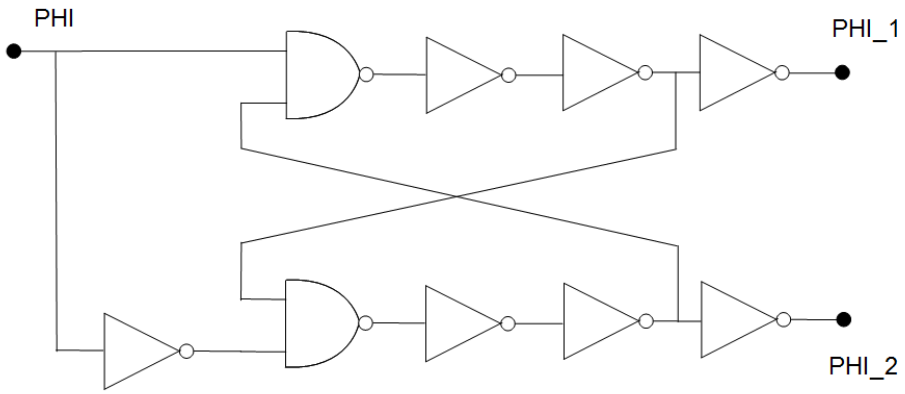

Fig. 6 2-phase clock generator **IMPORTANT NOTE CHANGE!**

A master clock generator is also required. Once again there are many possible astable circuits that can be used for clock generation. The clock generator circuit we will use is shown in Fig. 7.

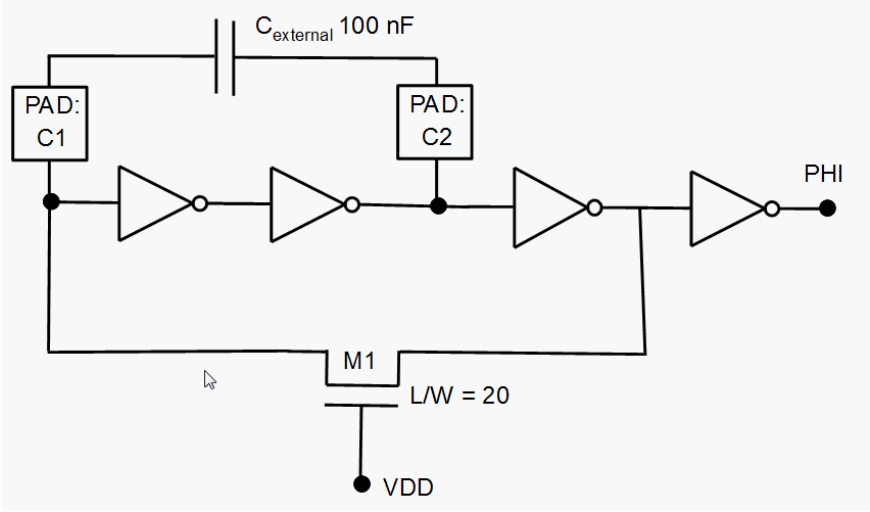

Fig. 7 Master clock generator- external capacitor $C_{external}$ and resistance of M1 determine frequency

$C_{external}$ is an external capacitor connected across two bond pads in the circuit. Values of $C_{external}$ between about 10 and 500 nF will be used for testing (capacitance values larger than about 10 pF are difficult to realize on-chip). For initial testing using an oscilloscope to observe the output waveform a relatively small $C_{external}$ giving a clock frequency of a few kHz is appropriate. For use in a bike taillight, the clock needs to be sufficiently slow that the LED flash rate is approximately 10 Hz, so that a human eye can see the flash. $C_{external}$ can be adjusted to give the flash rate you think will be most visible. For initial simulations, try a $C_{external}$ of 10 nF.

If the clock generator fails, or if we wish to test at higher clock frequencies than can be provided by the oscillator of Fig. 7 an external clock can be injected at bond pad C2. Similarly, two "minipads" pads should be provided to inject 2-phase non-overlapping clocks to the shift register, if necessary. Minipads should be added anywhere else you think it might be desirable to override or test a signal. "Minipads" can be about 10 λ by 10 λ in size, and of course must be in metal.

The complete chip should include an output buffer able to drive a GaP red LED. A typical red LED has a turn-on voltage of about 2 V and draws about 10 mA. A possible SPICE model for such an LED is given below.

```
.model Di_LED D (IS=4e-20 RS=0 BV=5 IBV=100u CJO=0 M=0.5 N=2)
```

The output buffer might consist of a single wide nMOS transistor in “open drain” configuration, as shown in Fig. 8. The driver transistor M1 should be able to sink 10 mA with V_{DS} = 1 V and V_{GS} -= 3V.

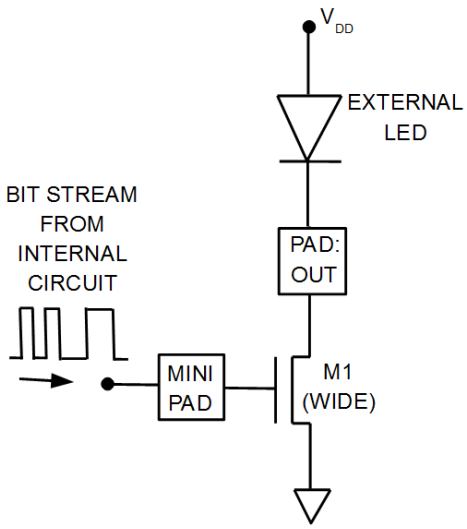

Fig. 8 Open drain output buffer

The complete project must fit in the floor plan shown in Fig. 9 which is equivalent to the cell *frame* in the *cmos2017.tdb* set-up. The height of the project block is fixed at 240 λ, and must not be changed. The vertical position of the bond pad centers must also not be changed. The width of the project can be at most 230 λ. There will be rewards for minimizing width.

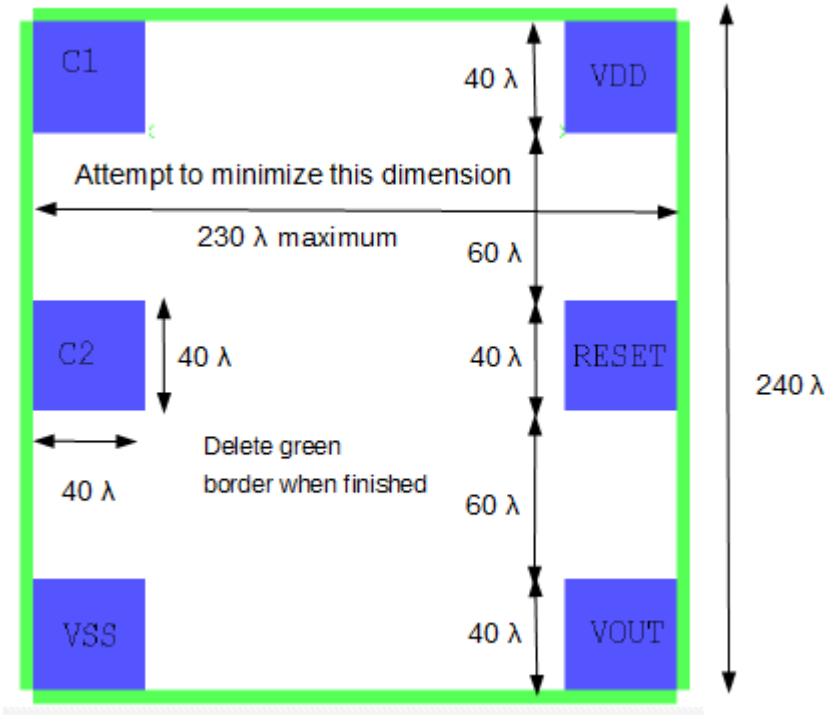

Fig. 9 Probe pad placement and available area for layout

Project Management

It is very important to make an early start on the project. A suggested approach is to draw schematics for each of the following blocks, and then verify the schematics with SPICE simulation:

1. D flip-flop (Fig.)
2. D flip-flop with reset (Fig.)
3. 2-phase clock generator (Fig.)
4. Oscillator (Fig.)
5. XNOR gate
6. Output driver (Fig.)

The schematics must include *L* and *W* for each MOSFET. It may make sense to divide up the design of these blocks amongst the partners in a project group..

Once the individual blocks have been verified, a full schematic for the project should be drawn using the blocks as subcircuits. SPICE should be used to demonstrate that the complete circuit is functional (this full schematic and associated SPICE simulation is due on **January 24** and will be worth 10% of the project grade).

To speed up the design process, LTspice schematics for basic NOR and NAND gate subcircuits are available on the 4609 labs webpage.

Once the schematics for each block have been completed, a rough floorplan for the layout should be sketched on paper showing the location of the different subcircuit blocks and routing of the VSS, VDD, clock and reset lines. Each block or cell can then be laid out in LEdit. Once again it probably makes sense to divide up this work between partners. Remember to DRC frequently as you proceed through layout. Once a cell is completed it should be extracted and verified by LVS against the corresponding schematic.

When all the cells have been laid out they can be assembled to produce a final, complete layout. Input protection should be added to the C1, C2 and reset pads (but **NOT** to VSS or VDD) at this stage. The complete layout should then be extracted and verified by LVS and also SPICE simulation of the extracted netlist. The final layout with proof of verification must be submitted by **February 14**.

A comprehensive report on the design will be due sometime in mid-March. Individual reports are required, even if you worked with a partner. If the chip fabrication is successful a brief report on test results for your design will be due in early April.