ECE 482

Final Project

Due: 3:00 PM, December 11, 2024

I. Background

Face detection is a fundamental computer vision application, widely used in areas like security, user authentication, social media, and human-computer interaction. It involves identifying and locating human faces within digital images or video streams. A crucial component of face detection systems is the image comparator, which compares visual features between images to determine if they represent the same subject. At the input to an image comparator, each image is represented by a binary data stream. For a black and white image, each pixel is represented by 1 bit whose value denotes black or white. Gray scale images are more widely used today; there, 8 bits are used to represent each pixel's intensity. In this project, you will develop an image comparator that compares two images. Pixel-level information will be encoded in 4 bits to limit the size of the circuit to be designed. You will be given data streams for two 4×4 bit images. Your circuit should be able to compare the two line by line (i.e., 4 bits at a time) and for each line, calculate the number of bits that are different. The results of that comparison are to be sent off chip, presumably to another chip that will ascertain if the two images should be classified as different or the same.

II. Image Comparator

A. Overview

A block diagram of the image comparator module is shown in Figure 1. The signal flow is from left to right.

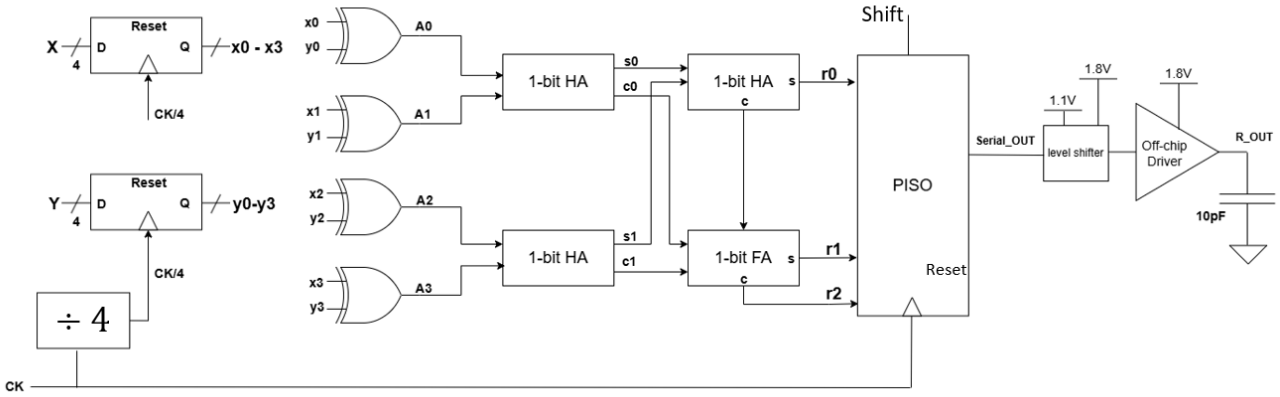

Fig. 1: Block diagram of the Image Comparator

The data enter the image comparator in parallel form. 4 bits of image X will be compared with the corresponding 4 bits of the reference image, Y. The master clock for the entire chip is denoted **CK**. The image data are input to the image comparator at a lower clock frequency, ¼ that of the master clock. The lower frequency clock ("**CK/4**") is generated by an on-chip frequency divider. The comparison of 4 bits from image X and 4 bits from image Y is effected by 4 XOR gates.

The outputs of those comparisons are denoted A0-A3. Those 4 bits are input to an adder module, composed of 3 half-adders and 1 full-adder. The output of the adder module is a 3-bit binary number between 000 and 100 (i.e., 0 to 4), which indicates the number of pixels that differ between the image (X) and the reference (Y). The result of the comparison will be serialized before being transmitted off-chip. The 3-bit parallel data are converted to serial form by a parallel-in serial-out shift register (PISO). Often, data are transmitted between chips serially to reduce the package pin count.

The output circuitry is described in Section II.D.

All registers used in this design are resettable. The circuit should be in a standby state whenever the signal **Reset** is high. **Reset** is an external signal. Optionally, you may operate the registers in your circuit with a clock that is inactivated whenever **Reset** = 1 ("gated clock").

B. Shift Register

The parallel-in serial-out shift register, "PISO," of Fig. 1 is shown in detail in Fig. 2; it includes an additional control signal named **Shift**. When **Shift** = 0, the PISO loads new data, i.e., the bits $r_2r_1r_0$ are loaded into the register. When **Shift** = 1, the data are shifted through the shift register at the clock frequency (**CK**), and output as serial data. The signal **Shift** must be generated on-chip and properly timed with respect to the clock edges (**CK**).

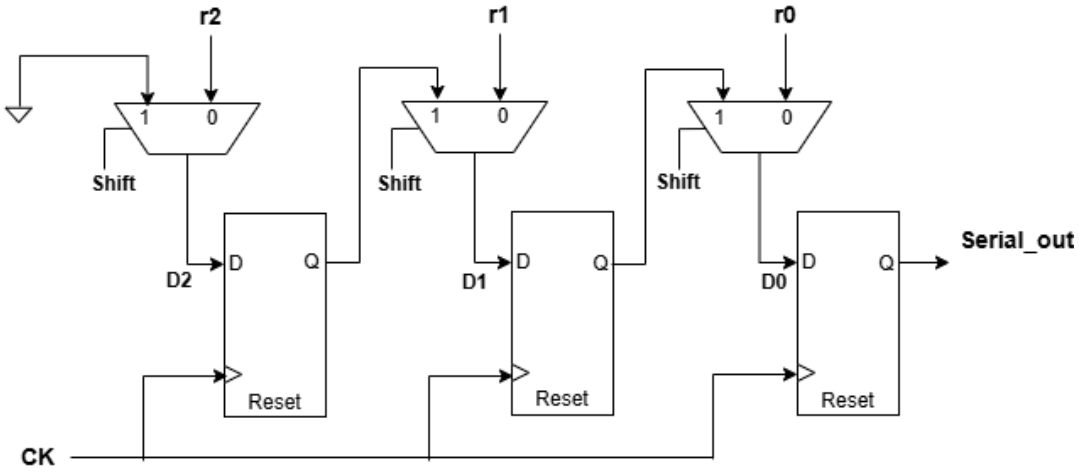

Fig. 2: Block diagram of the PISO shift register

C. Timing diagram

An example timing diagram is provided in Fig. 3. The signals labeled in red are provided to you in the testbench, and the signals in blue are to be generated on-chip by your circuit. The timing diagram shows an *example* of a functional design, i.e., one that provides the correct outputs; your circuit should have that same property. Furthermore, all correct timing diagrams will have a **Shift** signal that operates at the **CK** frequency and with a 75% duty cycle, as shown in the figure. However, it is not required that the arrival time of the zero pulse (**Shift** = 0) be identical to that shown in the figure; the phase of the **Shift** signal may be different from what is shown, and this will impact the circuit latency and/or timing margin. Latency refers to the number of clock cycles that pass before valid data appear at the output. Latency is measured starting from the first rising edge of **CK** after **Reset** goes low. (For this design, latency will be affected by the time offset between the **Reset** signal and the low-frequency clock (**CK/4**), which is non-deterministic. Therefore, the latency of your design might need to be expressed as range.) Low latency is generally considered good, but this project does not have a specified requirement. Additionally, the example timing

diagram represents a design without a gated clock, and the timing diagram will look different if clock gating is used. Finally, we note that **Reset** is an asynchronous signal, and it may switch at random times, not necessarily at the time instances shown in the figure.

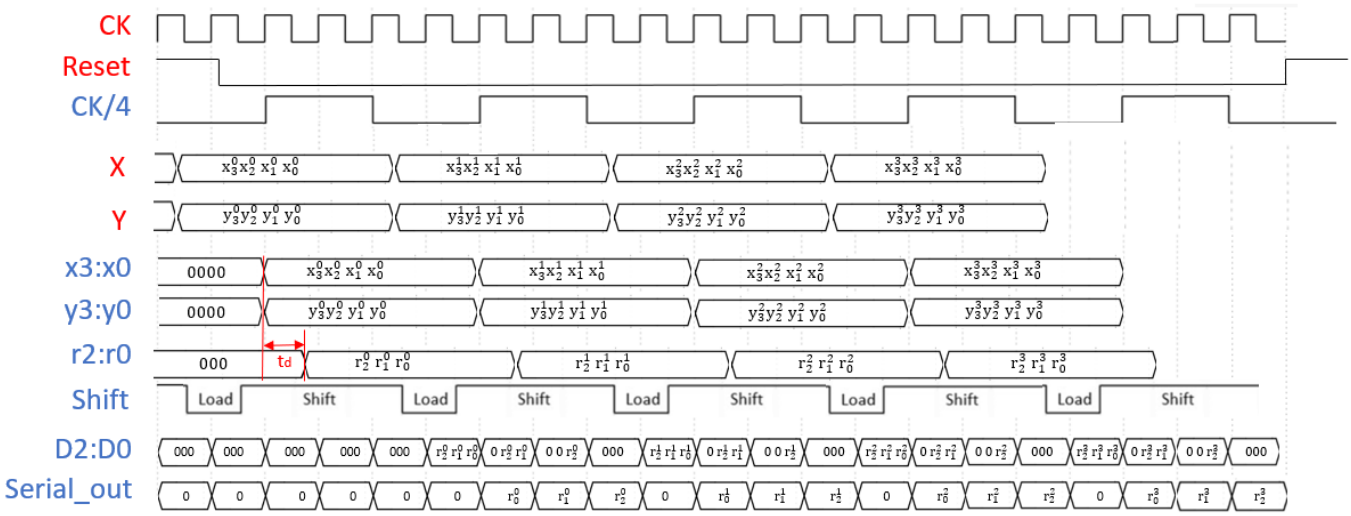

Fig. 3. Timing diagram for the system.

D. Output Circuitry

Low-speed off-chip drivers normally contain "IO" transistors that operate at a higher supply voltage ("VDDIO"). (For an advanced process node, anything below a few Gb/s is considered low-speed.) Your circuit should transmit the data off-chip using a simple single-ended CMOS driver, such as an inverter. You will need to insert a level shifter circuit between the PISO and the off-chip driver to shift the output high voltage from the core VDD up to VDDIO.

The off-chip load is represented by C_{OUT} = 10 pF and the driver must be sized large enough such that the output can swing between 0.1·VDDIO and 0.9·VDDIO within one clock period. If needed, you may use a superbuffer to drive the off-chip load; in such a design, the output stage is called the off-chip driver and the preceding stages are called the pre-driver. (If you do any manual analysis, remember that quantities such as t_{p0}, $C_{g,min}$, and γ have different values for IO and core inverters.)

The data that appears at the output should be identical to that at the output of the PISO; check that it is not inverted.

III. **Specifications and Test Bench**

You will design the circuit using the 45-nm GPDK. The supply voltage (VDD) for the core circuitry is 1.1 V; the supply voltage for the output driver is 1.8 V (VDDIO).

You must use the provided test bench, it includes the clock (**CK**), **Reset**, X data and Y data. **The test bench is to be downloaded from: ECE482_Project_StarterFiles.zip on Canvas**. You will be given another test bench 10 days before the project is due, it will contain X and Y data that are different from those in the original test bench, and you will use that second test bench to validate your design.

The circuit must be functional over all the process corners (TT, SS, SF, FS, FF). Verification of the circuit performance must use a netlist that is extracted from layout.

The CK signal in the test bench should be connected to a 1 µm × 1 µm square of Metal 5 at a single location of your choosing within the chip. From there, you must route it to all the registers on chip. If needed, you may insert clock buffers.

The target clock frequency (CK) for this design is 2.4 GHz. f_{CK} is an editable parameter in the test bench. If you are unable to achieve proper circuit functionality at all process corners at the target frequency, you may decrease the value of f_{CK}. The number of points deducted for failing to achieve the design target will be inversely proportional to how well the source of that shortcoming is analyzed in your report, and whether it is the result of simply starting the project too late. Conversely, extra credit will be awarded to any team that demonstrates full functionality of their circuit at a clock frequency above 2.4 GHz.

Power and ground must enter your design at Metal 9 at 1 or 2 fixed points on the periphery of your chip. It is best practice to use the low-resistance top-level metals to create low resistance buses and meshes for the power and ground nets. The recommended way to do upper-level power routing is illustrated in Figure 4. As shown, VDD and VDDIO have a horizontal Metal 9 bus on the top and bottom edge of the layout, respectively. Typically, this bus has near-maximum width. Because the largest number of circuit blocks are connected to the VDD net, it is also routed across the central part of the layout on vertical buses that are connected to the main bus. The VSS net is routed to horizontal buses branching out from the bond-pads, and like VDD, it also is routed to vertical buses on Metal 8. The illustrated power/ground routing provides a spatially-uniform low impedance. Notice that each metal layer is used primarily for routing in either a horizontal or vertical direction; this minimizes routing complexity. The thickness and spacing within the metal mesh will depend on DRC rules. Your finished layout will include supply connections on lower-level metals and to the various subcircuit blocks. It is recommended that you connect the various subcircuits to the top-level power/ground mesh toward the end of the design process.

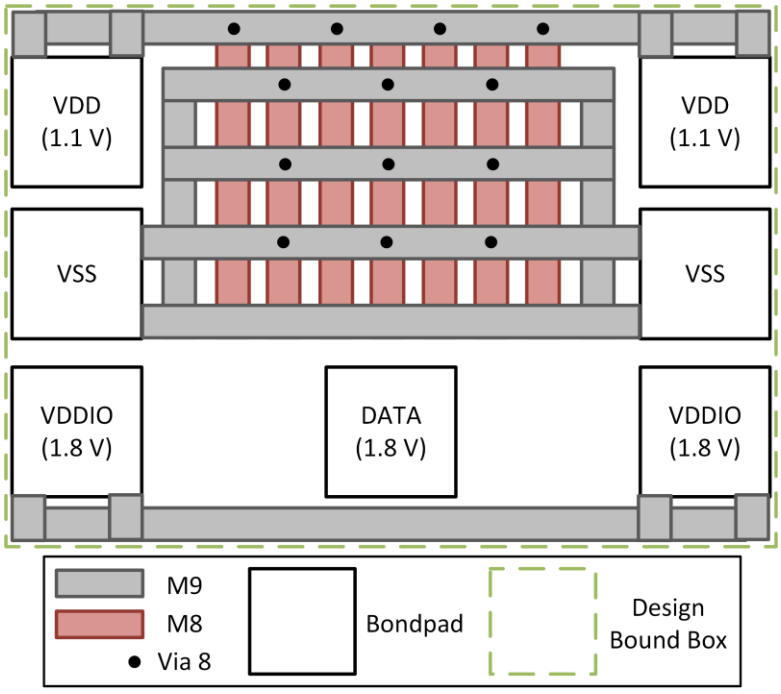

Fig. 4: Top-side view of top-level routing for Power/Ground. The so-called bond-pads can be drawn as 10 µm × 10 µm squares of Metal 9.

IV. **Requirements**

The design objectives are listed below, in order of decreasing priority.

1. Design a circuit that has the required functionality at all process corners.
2. Achieve correct operation at the target clock frequency, f_{CK} = 2.4 GHz.
3. Minimize the average power consumption of the entire circuit.
4. Execute a compact and neat layout.

In your final report, use a table to list the average power values of your design. Separate the contributions of the 1.1 and 1.8 V supplies; that is, report the power value for each supply domain.

The final set of simulations are to be performed using a netlist that was RC extracted from the layout of the complete design. Those are the simulation results that will be included in your report. RC simulations of a top-level design often have a lengthy run time, due to the large number of nets in the extracted circuit. Therefore, it is recommended that students first verify the functionality of the design (and ensure sufficient design margin) with C-only extraction before proceeding to RC extraction.

You must demonstrate that your circuit has the correct functionality over all the process corners (TT, SS, SF, FS, FF) using the testbench provided to you. Power is to be simulated using typical component models (TT process corner).

V. **Report**

Each team's report must conform to the following outline.

i. Description of your overall design approach. Starting from the top level, describe your architecture choices and justify those based on performance metrics such as power, area, delay, etc. Include diagrams of your schematic as each portion of the design is presented, descending through the hierarchy. This section should include schematics down to the logic gate level. Justify/explain why you chose to use a particular type of logic gate or register. The level of detail in this section should be such that another student in ECE482 could replicate the design. **Be certain to highlight any clever aspects of your design rather than assuming the grader will identify those features on his/her own.**
   Within the diagrams, blocks should be used to represent repeated circuit parts. Do not cut and paste a Cadence screenshot. All figures should have informative captions.

ii. Transistor-level rendition of any combinational logic gate that is other than complementary static CMOS. Transistor-level schematics of registers. Table listing all the transistor sizes. Do not cram all the requested schematics onto a single page. Drawings should be clear and text labels readable.

iii. A completed copy of the table shown below:

1	Maximum f_{CK} for correct functionality at all process corners	
2	Average power drawn from VDD at MIN(f_{CK} listed in row 1, 2.4 GHz)	
3	Core supply figure-of-merit: Power / frequency (same power and frequency as in row 2)	
4	Average power drawn from VDDIO at MIN(f_{CK} listed in row 1, 2.4 GHz)	
5	IO supply figure-of-merit: Power / frequency (same power and frequency as in row 4)	

6	Area of Top-level layout (µm×µm)	
7	Latency (measured in number of **CK** cycles)	

iv. Demonstration of circuit functionality for the TT, SS, SF, FS and FF process corners.
v. Discussion of results. If something didn't work as expected, explain/analyze that result.
vi. Top view of the entire layout. Discussion of layout strategy.
vii. Describe how the workload was shared across the team.

Your report must be typeset using a word processing program. Do not use font sizes smaller than 10 point. **Make certain that all figures/plots are legible, have informative captions, and are labeled completely.**

A .pdf of the report, the project Cadence library and HSPICE netlists will be submitted to Canvas as a .zip file. Within this .zip, include a README that points out the location of the top-level design and simulation netlists that are included in the report. If your report is clearly written and if there is evidence that your design operates correctly and meets specifications, there will be no need for the instructional staff to look at the project files; however, if the report is poorly written or the design appears flawed, the TAs may need to access the project files to determine partial credit.

VI. **Grading**

75% of your grade will be based on how close the design comes to meeting all the objectives. 25% of your grade will be based on the clarity and literary quality of your report. Do not limit yourself to a maximum score of 75 points by turning in a sloppy or cryptic or poorly organized report. It is recommended that you complete the design **at least** one week prior to the due date so that you have sufficient time to write a professional-quality report. Homework assignments in November will be relatively short to free up time to work on the project. Take advantage of that opportunity.

Each team will hand in one report. In the vast majority of cases, each member of the team will receive the same grade for the report. At the end of the semester, you will fill out a form to evaluate the efforts of your teammates; one needs to be classified as satisfactory by a majority of his/her teammates to receive full credit.

Project teams

Group	Team members	Netid
1	Chang, Kelly	kellyhc2
	Kim, Andrew	akim229
	Koseli, Mahir	mkoseli2
	Wang, Yibai	yibaiw2
2	Danthinne, Simon	simoned2
	Hsiao, Ming-Yan	myhsiao2
	Mukherjee, Jayanto	jayanto2
	Yang, Peidong	peidong5
3	Chen, Ting-Yu	tingyuc4
	Dilbaghi, Dhruv	dhruvd4
	Koo, Isaac Koo Hern En	ikoo2
	Murali, Pavvithra	pm44
4	Baala, Raj	kbaala2
	Lin, Chien-Ming	cmlin3
	Wang, Robert	rlwang2
	Yang, Xianhui	xianhui2
5	Deng, Runbo	runbod2
	Fu, Ryan	ryfu2
	Li, Howard	zl114
	Roberts, Alex	asr9
6	Hsieh, Pei-Chin	peichin2
	Hsu, Wei-Chun	weichun4
	Liu, Shu-Wei	shuweil3
	Zhang, Zhuoer	zhuoer3
7	Chen, Wei-Jen	weijenc2
	Freeman, Danny	djf6
	Subramanian, Sujay	sujays3
	Zhou, Ethan	yz69
8	Bekdache, Omar	omarb3

	Li, Harbin	hdli2
	Wang, Joey	joeyw4
	Zhang, Haochen	hz39
9	Khandelwal, Ananya	ananya15
	Kim, Ben	kijungk3
	Shaheen, Kadin	kadinas2
	Yuan, Muye	muyey2
10	Ding, Donny	xuanzhe2
	Panchmia, Dhruv	dhruvp4
	Singh, Rawnie	rawnies2
	Wu, Stanley	zaizhou3
	Wu, PoHao	pohao2
11	Arshad, Zoya	zarsha3
	Liu, Yening	yeningl2
	Sun, Ruize	ruize2
	Tian, Xiaoyang	xt12
	Yan, Ruofei	ruofeiy3
12	Cheng, Ryan	chingc5
	Kothary, Aaditya	kothary3
	Liu, Junmin	junminl2
	Santhanam, Shreya	shreya37
	Wen, Tinhsu	ttwen2
13	Lee, Ying-Ching	yl192
	Nathan, Jay	jayrn2
	Tang, Eric	leweit2
	Thirasuntrakul, Phuvit	phuvitt2
	Yan, Jie	jiey7
14	Abrams, Sloan	sloanaa2
	Chen, Raymond	rc18
	Choi, Jiho	jihoc3
	Huo, Zixiao	zixiaoh3

	Tsao, Fang-Chi	ftsao2