**8 Bit Kogge-Stone Adder**

Cannon Wurster A16675388

Chin-Chou Wu A16408078

The design of robust adder circuits is important as they are a crucial building block for more complex circuits like multipliers and even entire processors. Creating fast yet efficient adder digital circuits is an ongoing design challenge as process technology improves and the circuits that use them get faster. Over the years, there have been many designs that aim to improve on certain performance parameters like frequency, delay, power consumption, area, and number of transistors. However, not all these parameters can be optimized in one design as often optimizing one parameter hinders another.

With the design of the first full adder block, engineers have been working to optimize it for various parameters and applications. By cascading full adder blocks it was found early on that the critical input delay was from carry in to carry out. So, many of the early optimizations to the full adder circuit involved finding a method to have a faster carry delay.

One of the first optimizations was the carry skip adder which allowed for the carry to skip over certain bits where it wasn’t needed. The circuit makes the decision to skip by introducing logic that can tell whether the carry is propagated or terminated at each bit.

After this, it was found that the carry into each bit could be calculated based on the original inputs before it by using generate and propagate logic. This is when the carry-lookahead adder was designed. This adder design is able to compute 4 carry bits into each of their respective full adder blocks simultaneously. It achieves this by using high valency cells which can take in more than 2 inputs at a time. With this design, the delay was drastically reduced but at the expense of power and number of transistors.

With the discovery that carry bits could be computed without the previous one as seen with the carry lookahead adder, engineers began exploring ways to improve upon this lookahead method. The tree adder was the product of this exploration. Tree adders are able to perform lookaheads on the lowest level lookaheads recursively. This recursive lookahead gives O log(N) delay. There are many forms of tree adders but the one implemented in this project was the Kogge-Stone Adder.

The Kogge-Stone adder is a variation of the tree adder which uses generate and propagate logic to perform recursive carry lookahead. The most basic cell utilized in the design is the generate or G cell. Shown in **Figure 2**, this cell combines two logic gates into one static CMOS circuit. This cell is then used to create the PG cell shown in **Figure 1**. The PG cell combines the G cell with an AND gate to produce both generate and propagate logic in one block. In both blocks, the transistors were sized to have equal drive strength in the pull up and pull down networks. By keeping the transistors small we were able to mitigate increased power consumption and delay due to increased capacitance. These 2 cells are then used in **Figure 3** to create the full 8 bit adder. Standard AND and OR gates are used for the full adder to generate initial P and G values while XOR gates are used to generate final sum values.

From testing, it was found that 00000001 + 01111111 = 10000000 was the critical input to the adder with a propagation delay of 178.463ps. This input case is plotted in Figure 5 where it can be seen that only SUM<7> is driven high while SUM<8> and SUM<6:0> are driven low which shows the adder functions properly for the worst case input. We also tested the delay to carry out by inputting 00000001 + 11111111 to get a carry our of 1 followed by a sum of 00000000. It was found that this produced a smaller delay of 165.763ps further verifying the critical input discovered previously.

Given the delay through the critical input, it was found that the circuit could operate at a maximum frequency of 3.633GHz which can be seen in **Figure 5** where the circuit is operating correctly at this frequency. Shown in the same figure is the current waveform used to calculate the average power clipped over 10 iterations. We clipped time from 0.3ns to 6.5ns, and subtracted average power consumed by the Synthesized Adder at 3.633GHz from overall average power consumption of both Adders at 3.633GHz to calculate average power consumption of our Custom Design at max frequency of 3.633GHz. It’s because our Custom Design uses both “VDD!” as well as “VDDcustom!” while Synthesized Adder only uses “VDD!” as a power source. It was found that our Custom Design consumed a reasonable 437.154uW of power. Furthermore, we used the same metric of subtraction to calculate power consumption of our Custom Design at 1GHz, clipped from 0.3ns to 21ns, and got result of 158.718uW.

Several things to note here are, first, the first 0.3ns is the initialization time for both adders so we didn’t include it in our calculation. Second, 6.5ns and 21ns are best representations of around 10 operations at each of the frequencies so we used those time durations to calculate each average power consumption of our Custom Design, and for Synthesized adder we used 10ns and 21ns respectively. Third, we calculated the average power consumption of the Synthesized Adder by ignoring our Custom Design (including the Adder specifically and flip flops in the Custom Adder block) in the testbench schematic. Fourth, the max operational frequency of the Synthesized Adder is found to be 2.302GHz.

The results for maximum frequency, power consumption and energy per operation at max frequency and at 1GHz of our Custom 8 bit Kogge-Stone Adder are all shown in **Figure 6**, as well as those of the given Synthesized Adder.

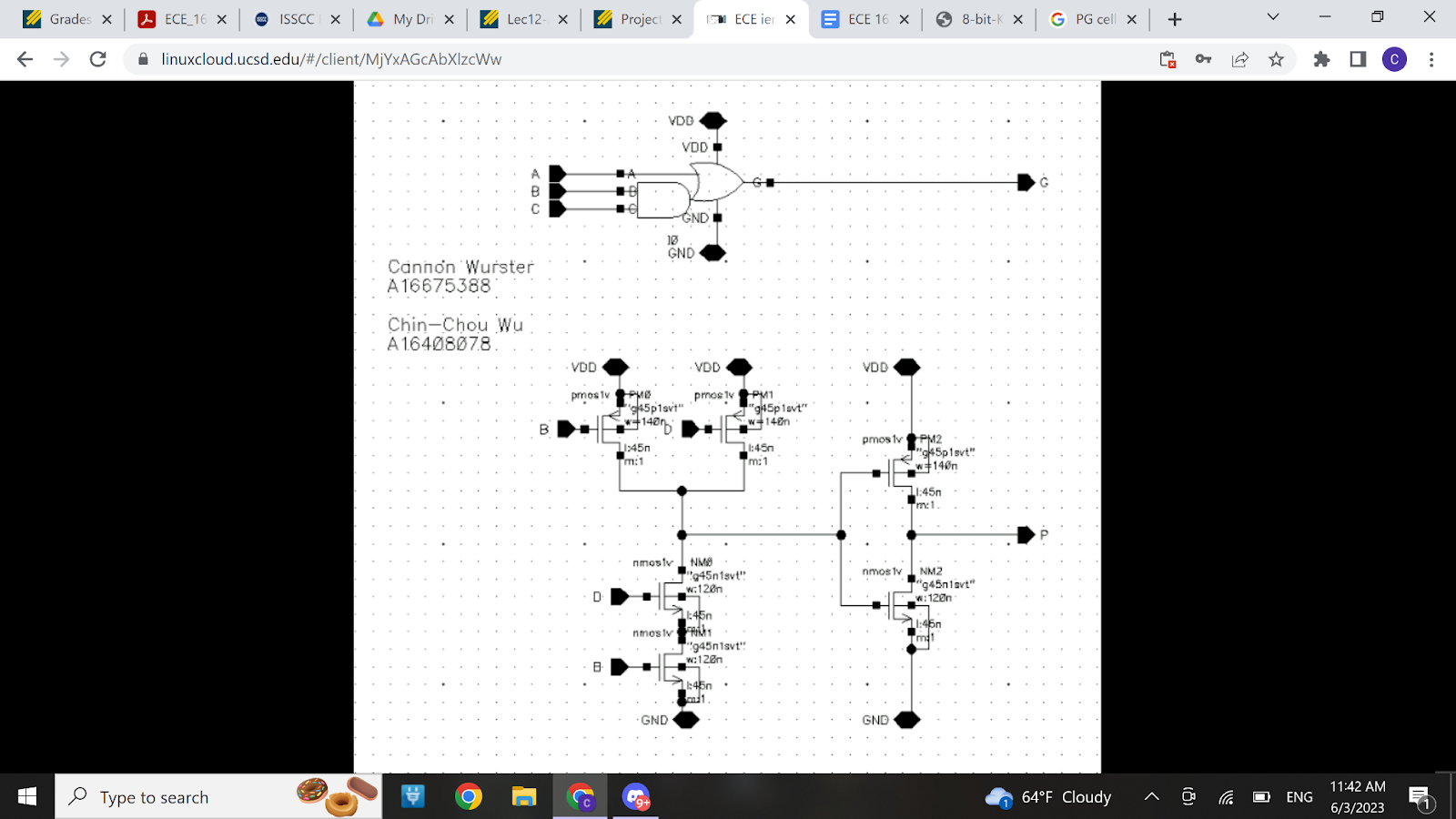
An innovative aspect of our work was that we designed the circuit to be able to handle a carry in although not required for the project. For purposes of testing the carry in was tied low; however, we did verify it functioned correctly in our own testing not reported. In the future, the adder can be further optimized by tuning the transistor sizes and optimizing how the inputs are placed in relation to the output.

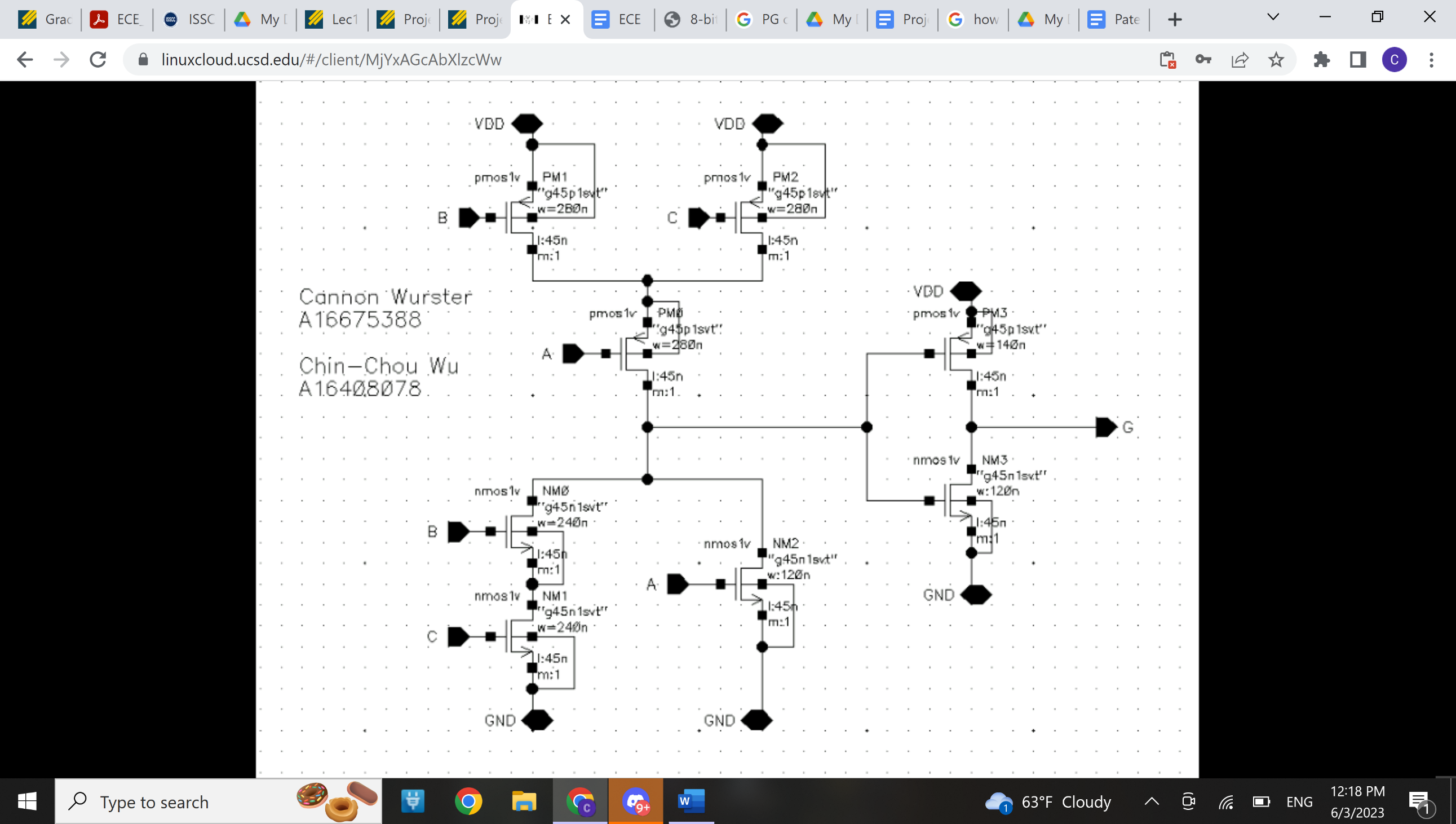
The design of the G cell and PG cell was completed by Cannon. He also assembled and wired these cells together in the full 8 bity Kogge-Stone adder. Chin-Chou completed testing on the circuit which included performing power and max frequency calculations on both the custom and synthesis adders. This report is a culmination of both of their work and is written in parallel between them.

In total, the Kogge-Stone adder proves to be major improvement on the Carry Lookahead adder as delay was drastically reduced allowing the circuit to operate at about double the frequency than the PnR adder. Although there is increased complexity and more transistors, the Kogge-Stone adder has only marginally worse power consumption. This could be due to the smaller transistor sizes which allow for relatively low capacitance.

*References:*

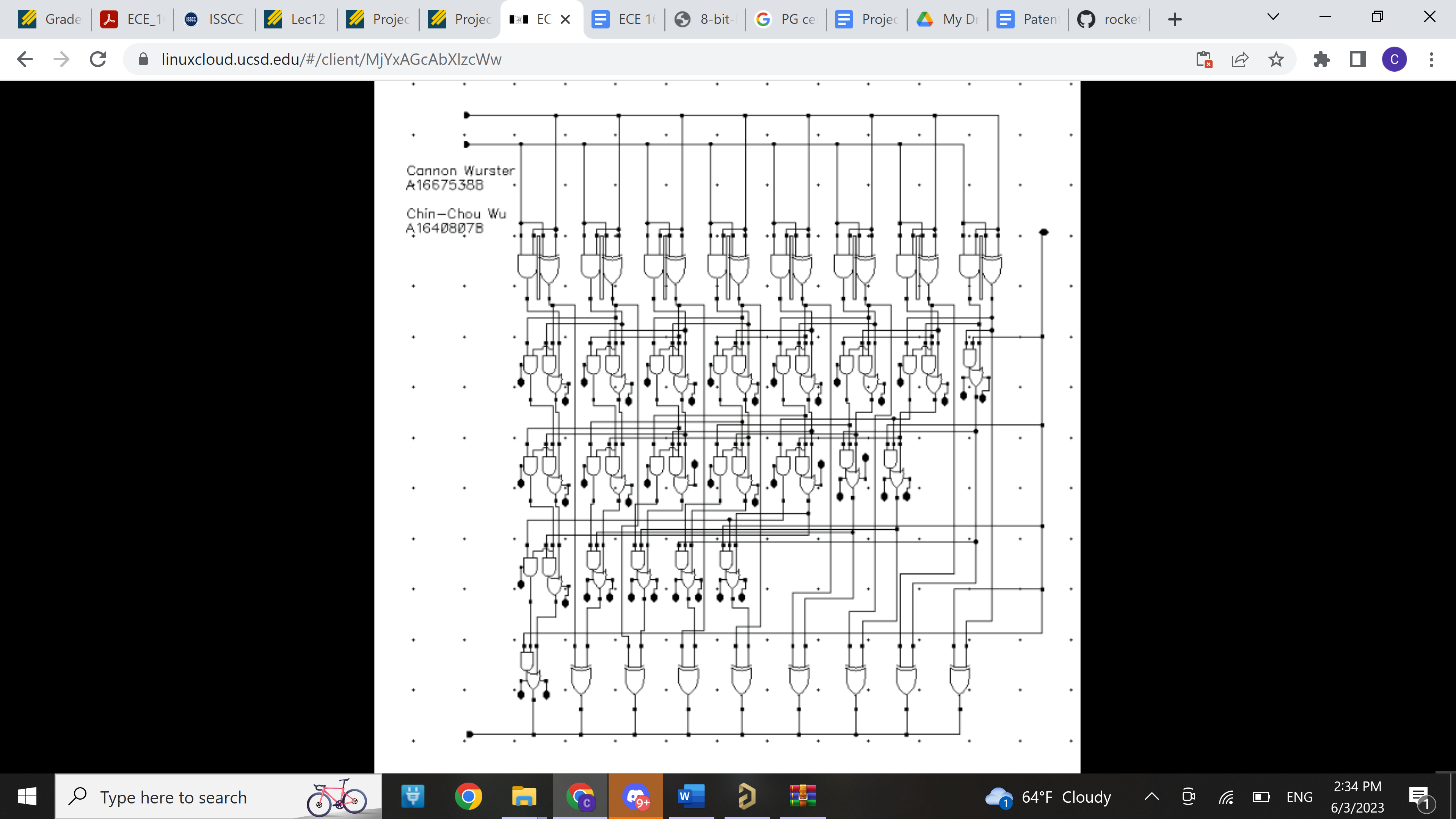
[1] ECE 165 Lecture 12, Patrick Mercier





**Figure 1 |** PG Cell Schematic

**Figure 2 |** G Cell Schematic

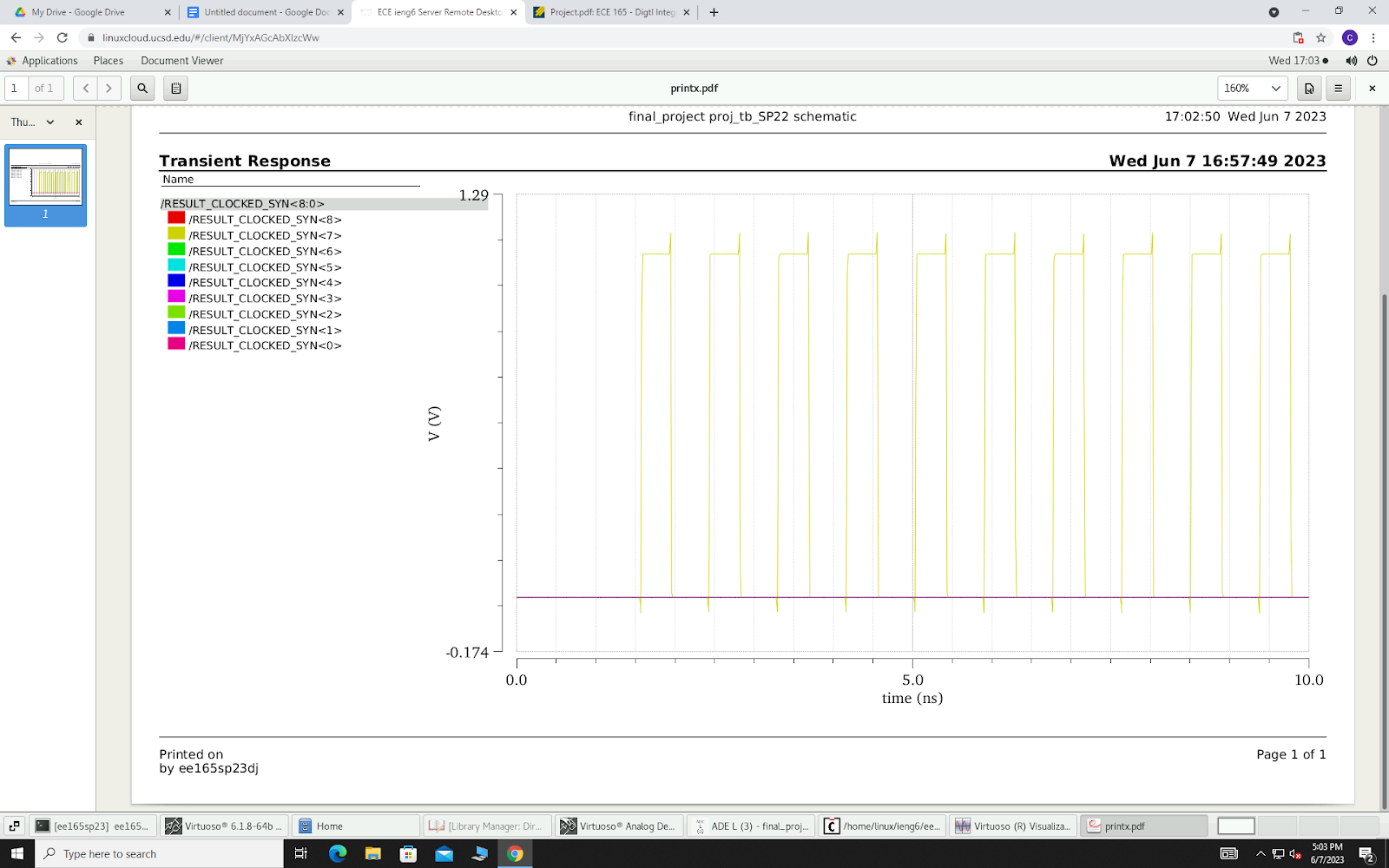


**Figure 3 |** Insert caption here

**Figure 3 |** Full Kogge-Stone Adder Schematic

A picture containing text, line, screenshot, plot

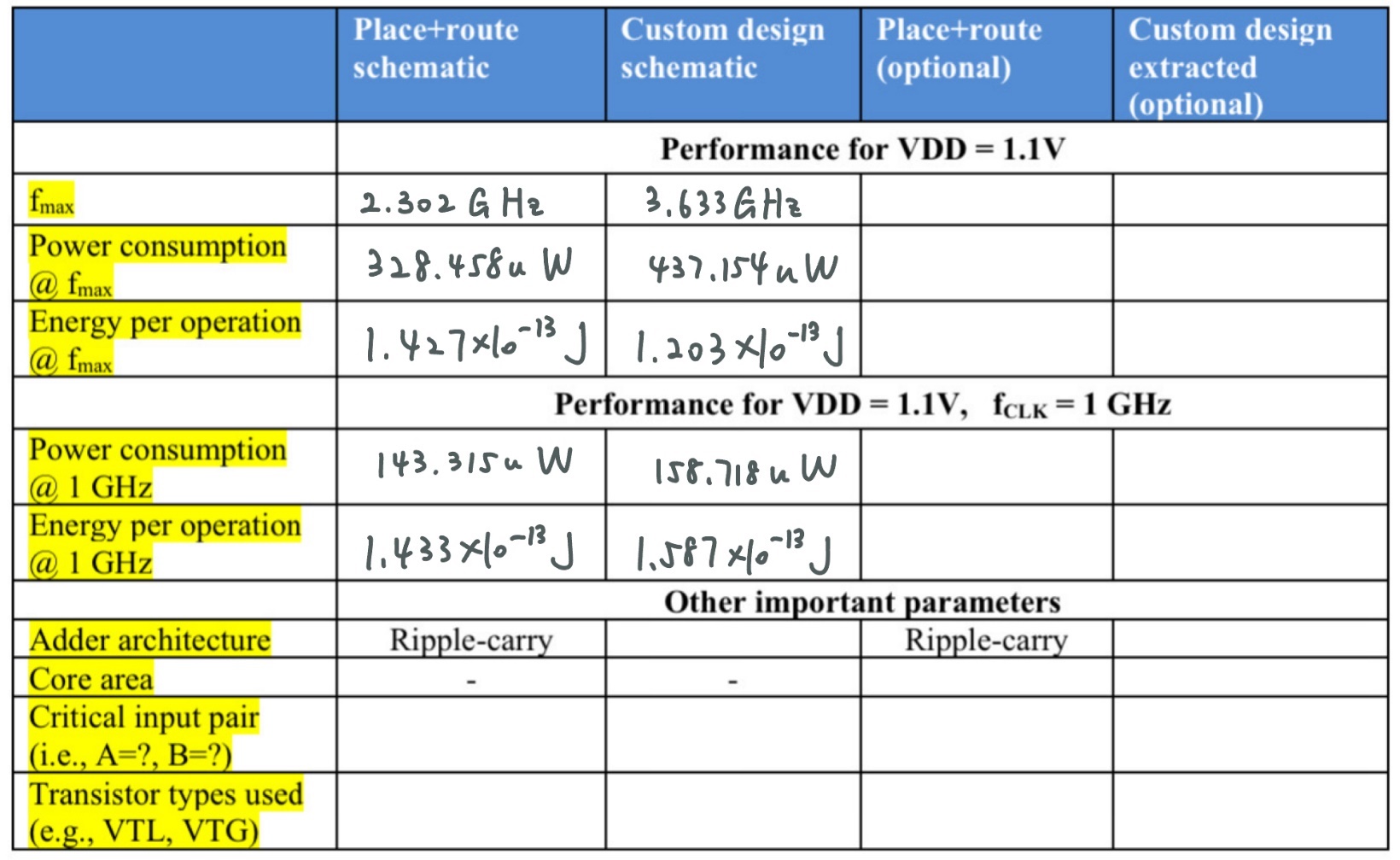
Description automatically generated



A picture containing text, plot, screenshot, font

Description automatically generated

**Figure 5 |** Custom Adder With Critical Input at 3.633 GHz (Top), Synthesis Adder with Critical Input at 2.302 GHz (Middle), and Custom Adder Current Over 10 Cycles at 3.633 GHz With Critical Inputs (Bottom).



**Figure 6 |** Table of Results

Normal Threshold

Normal Threshold

00000001

+ 01111111

00000001

+ 01111111

Kogge-Stone