**AHSANULLAH UNIVERSITY OF SCIENCE AND TECHNOLOGY**



**PROJECT**

**Course No : EEE 4134**

**Course Title : VLSI-I Lab**

**Name of the Project : Secure CMOS logic through logic encryption**

**Group 1**

**1. Aalolika Roy Chowdhury Student ID: 19.02.05.111**

**2. Tonima Rani Ghosh Student ID: 19.02.05.112**

**3. Raihan Ahmed Student ID: 19.02.05.114**

**4. Md. Idrak Efaz Student ID: 19.02.05.121**

**5. Sumaia Akter Ritu Student ID: 19.02.05.123**

**Section: [B2] Date of Submission:**

**Year: 4th  21st August, 2023**

**Semester: 1st**

**Abstract:**

In the present world, the semiconductor industry is growing rapidly where hardware security is the primary concern to prevent piracy. Many fabless companies send their ICs to the foundries that have advanced fabrication capabilities. But there comes a risk of IP piracy, reverse engineering, overproduction and malicious tampering of IC for Trojan insertion as the IC design flow is known to the attackers. An efficient method to protect hardware is logic encryption where the original functionality is accessible by the authorized persons only. In our proposed circuits, a novel transistor-level method logic encryption for CMOS gates is ensured along with power, cost, performance, and reliability optimizations. We designed secure OR, AND, NOR, NAND, XOR and XNOR circuits where the correct output depends on the logic level of key. Faulty outputs will be provided by the encrypted circuits during the application of an incorrect key pattern.

**Background:**

When the chips enter in the supply chain of IC, they can be attacked by the hackers through reverse engineering to access the design or specific secrets from a design. Annually 4 billion dollars is lost in the semiconductor industry due to these problems. So, the IP vendors have to face difficulties to save the intellectual properties from piracy, overproduction and reverse engineering. Many logic encryption techniques have been applied by the researchers to protect hardware from piracy but those methods increase some circuit parameters like area, power, delay, and energy. In our circuits, these disadvantages were eliminated. Area, power, delay, and energy are reduced by an average of 42.94%, 37.37%, 26.79%, and 50.96% respectively in the proposed encrypted key gates. Moreover, other logic encryption methods have a constant output like as logic high (1) or logic low (0) whereas our proposed gates don’t provide any continuous or constant output which helps to prevent piracy. Without the correct key, the attacker will not be able to access the circuit netlist by reverse engineering because a Trojan will not be inserted into the netlist’s internal node.

**Result (Data Representation):**

**Truth Table of Proposed Gates**

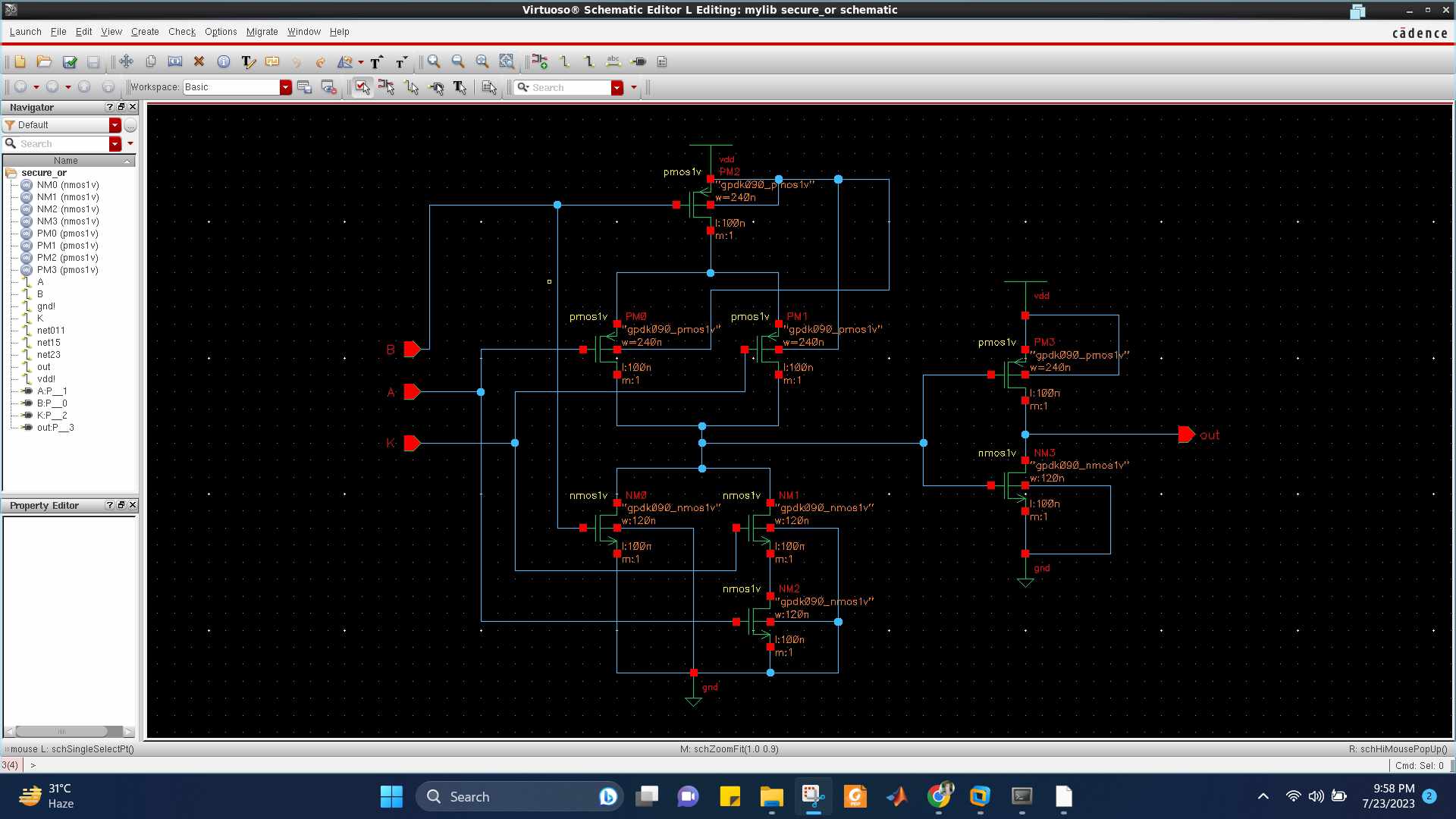
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Key** | **Inputs** |  | **Outputs** |  |  |  |  |  |
| **K** | **A** | **B** | **AND** | **OR** | **XOR** | **NAND** | **NOR** | **XNOR** |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |

**Comparison Table**

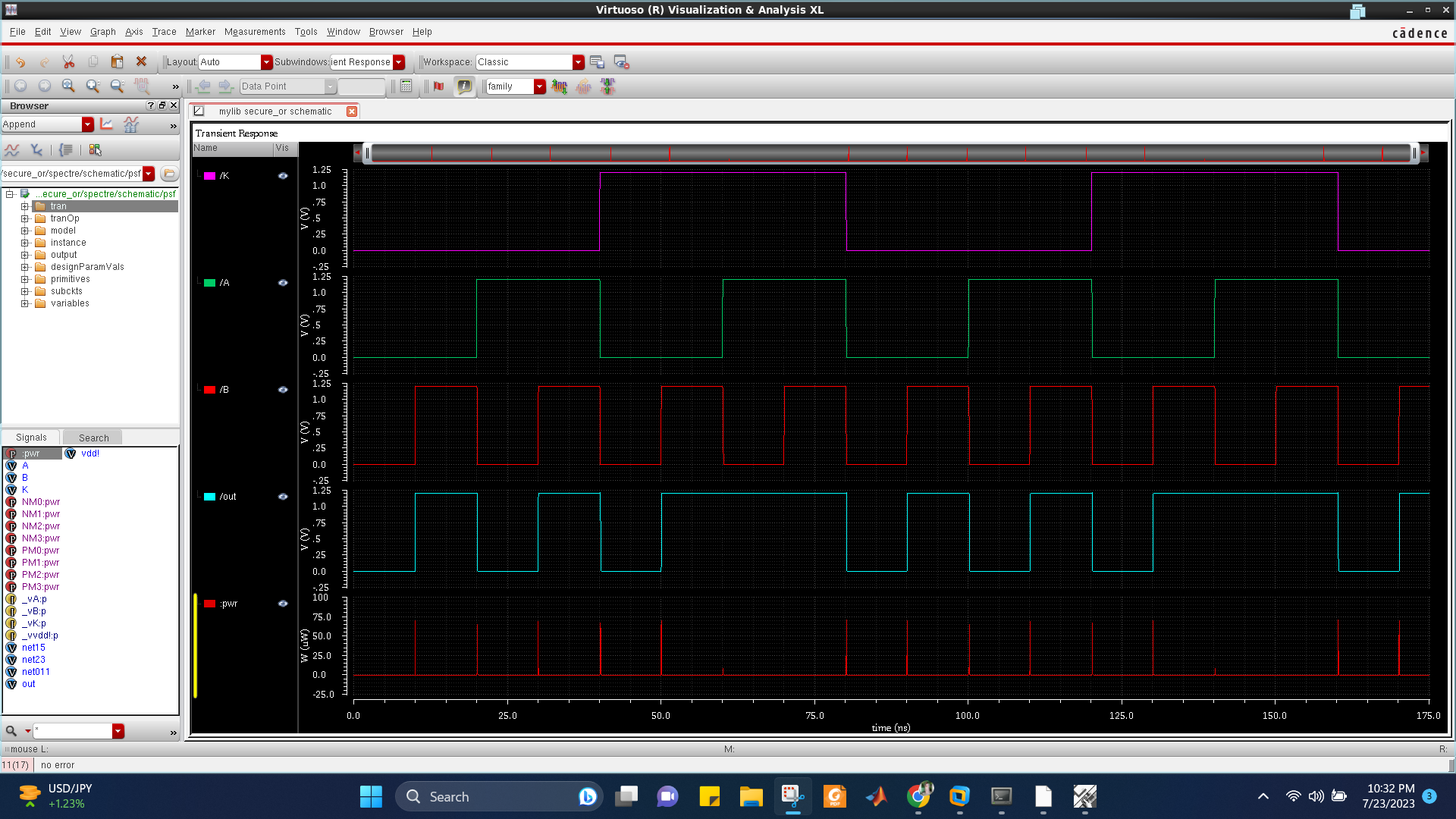
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Gate no | Propagation delay | Average Power | Power delay Product | Cell area | No. transistors | No. of DRC | No. LVS Mismatches |
| NAND | 143.8E-12 | 107.7E-9 | 1.587E-17 | 9.6288 micrometer | 6 | 0 | 0 |
| AND | 165.1E-12 | 188.0E-9 | 3.103E-17 | 19.516 micrometer | 8 | 0 | 0 |
| NOR | 40.83E-12 | 155.3E-9 | 6.341E-18 | 31.014  micrometer | 6 | 0 | 0 |
| OR | 56.16E-12 | 280.4E-9 | 1.574E-17 | 36.636  micrometer | 8 | 0 | 0 |
| XOR | 11.17E-12 | 361.9E-9 | 4.042E-18 | 37.0821 micrometer | 14 | 0 | 0 |
| XNOR | 10.24E-12 | 311.0E-9 | 3.184E-18 | 33.7800 micrometer | 14 | 0 | 0 |

**Result (Pictorial Representation):**

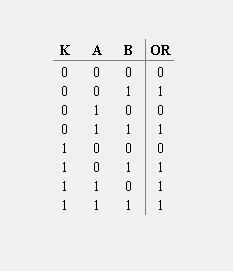
**Secure OR Schematic**

****

**Secure OR Waveform**

****

**Secure OR Truth Table**

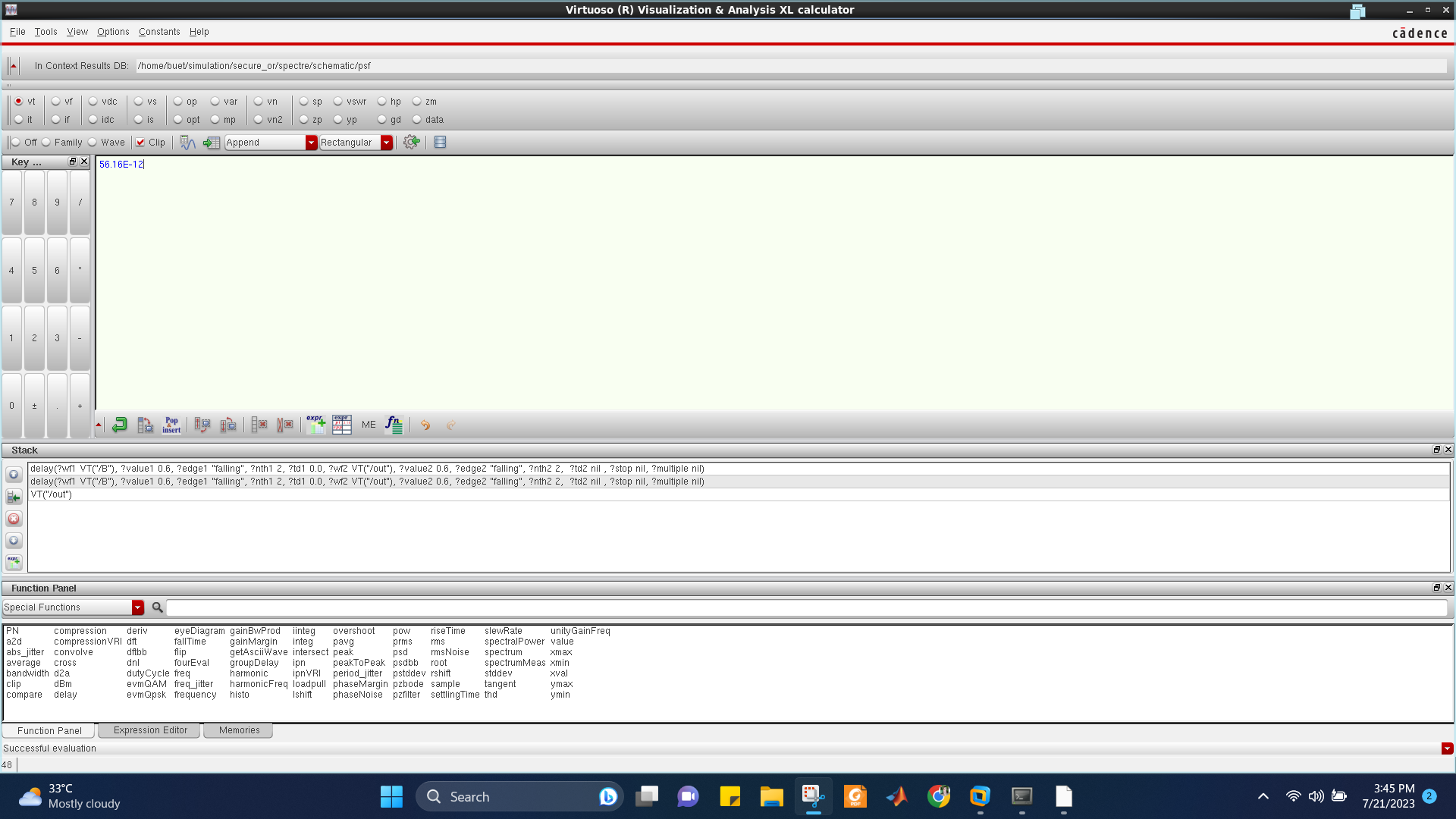
****

From the schematic diagram, we get the output function to be,

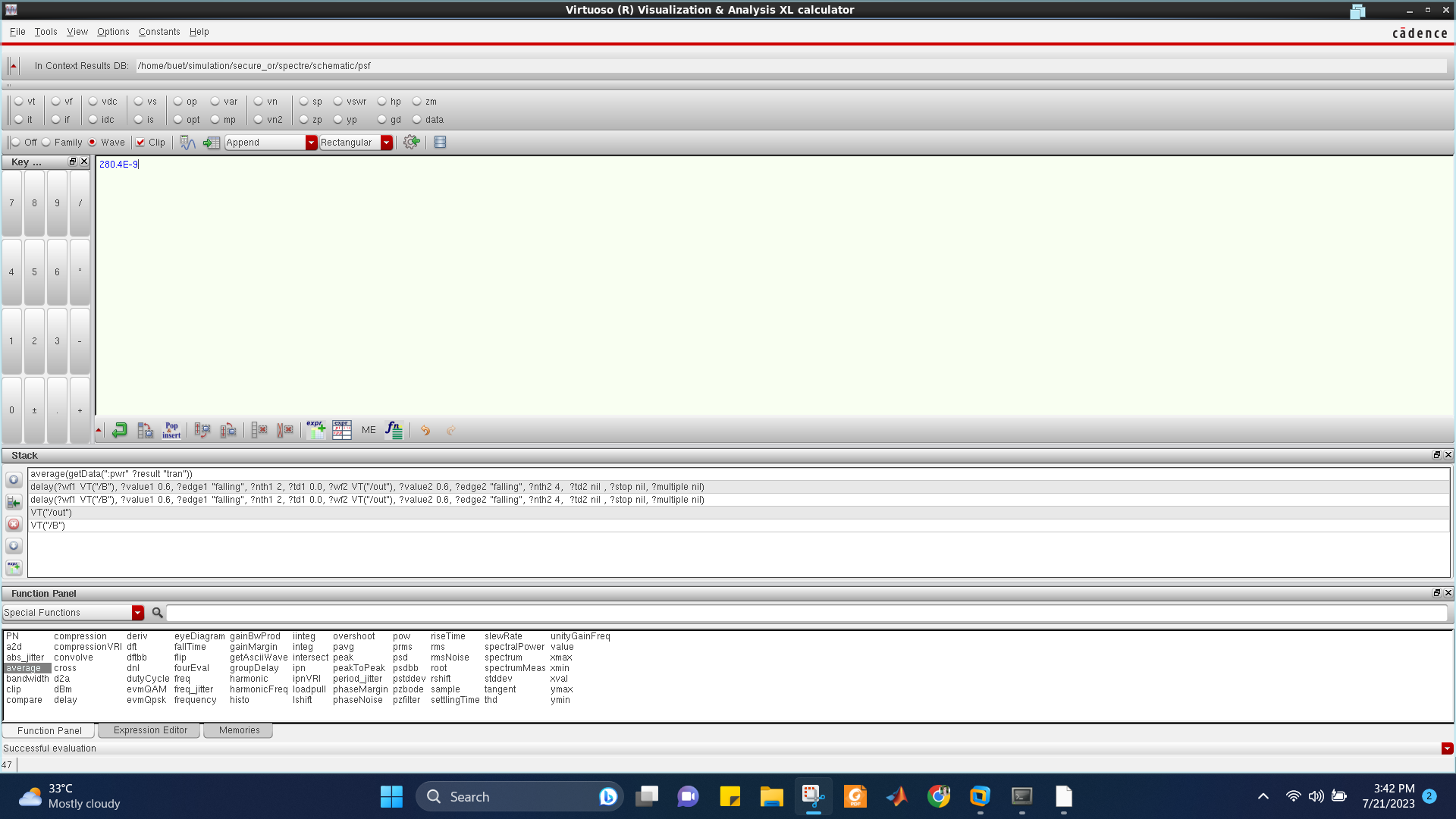
OR = AK + B

When the logic level of K is 1, we get the output as OR = A + B , which is the expected output of OR gate. Whereas, if K = 0, the output will not provide the expected values of OR gate. In this way, the privacy of this gate is ensured as without the particular value of K, the accurate result of OR gate will not be found.

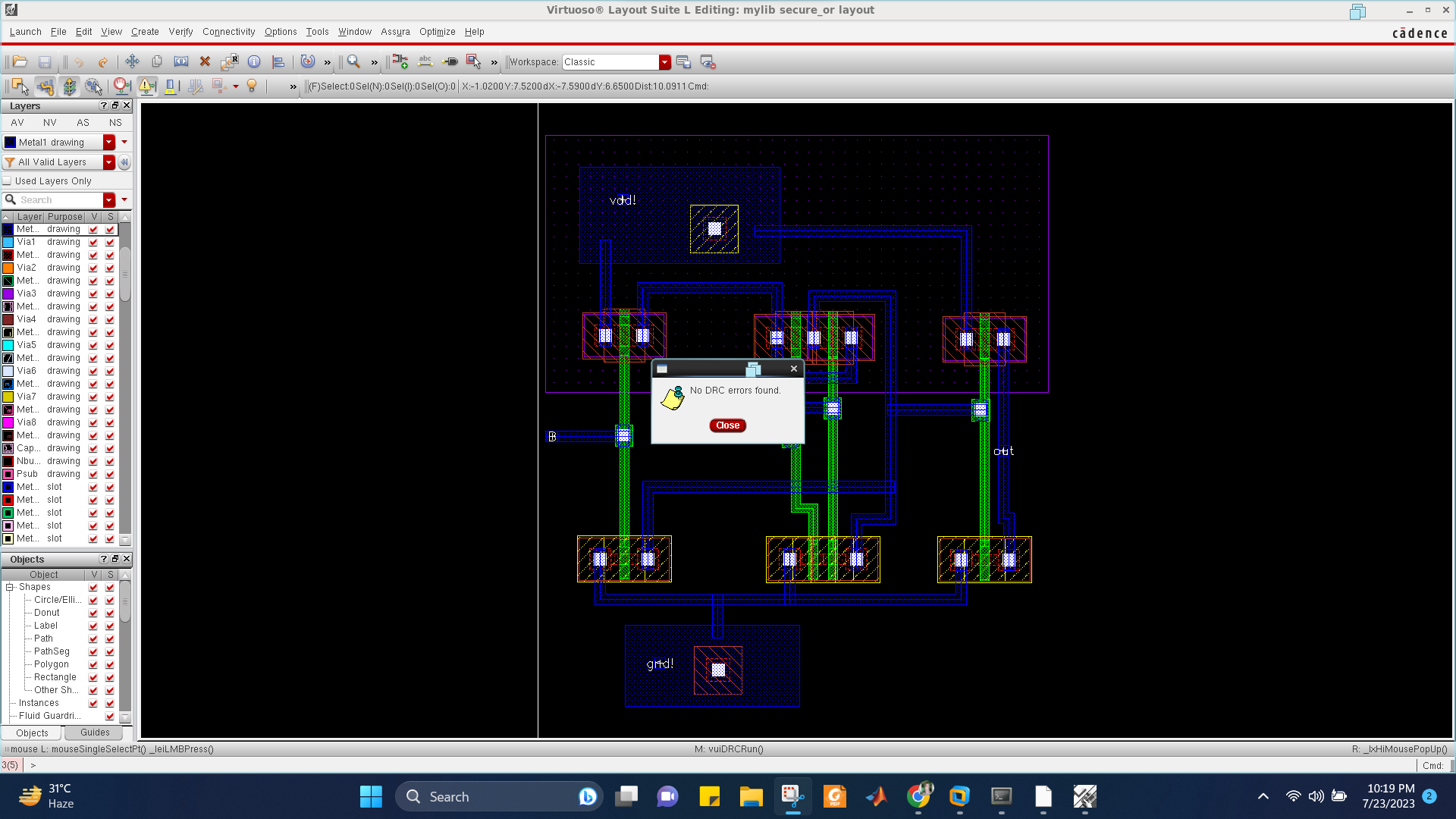
**Secure OR Propagation Delay**

****

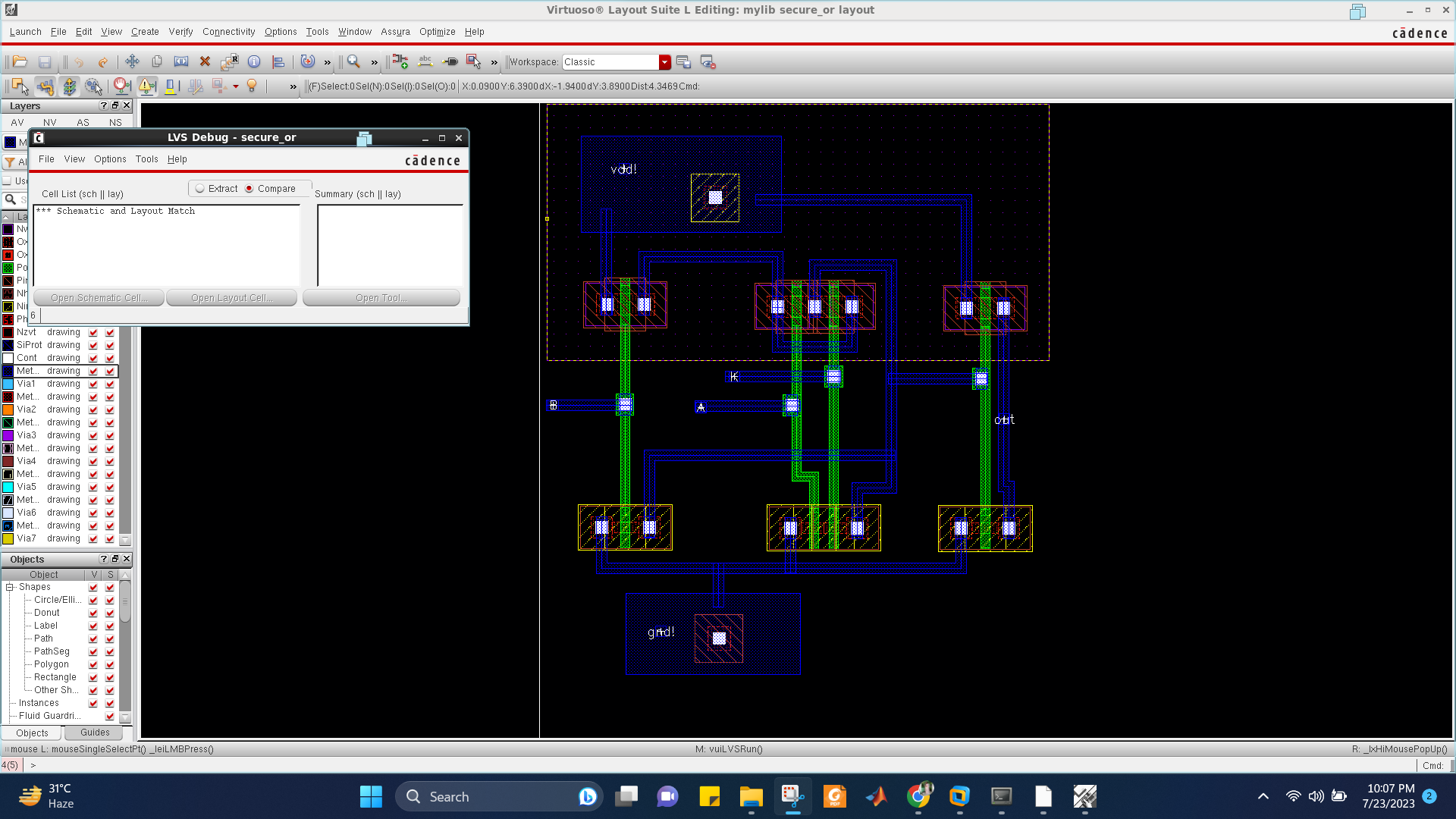
**Secure OR Average Power**

****

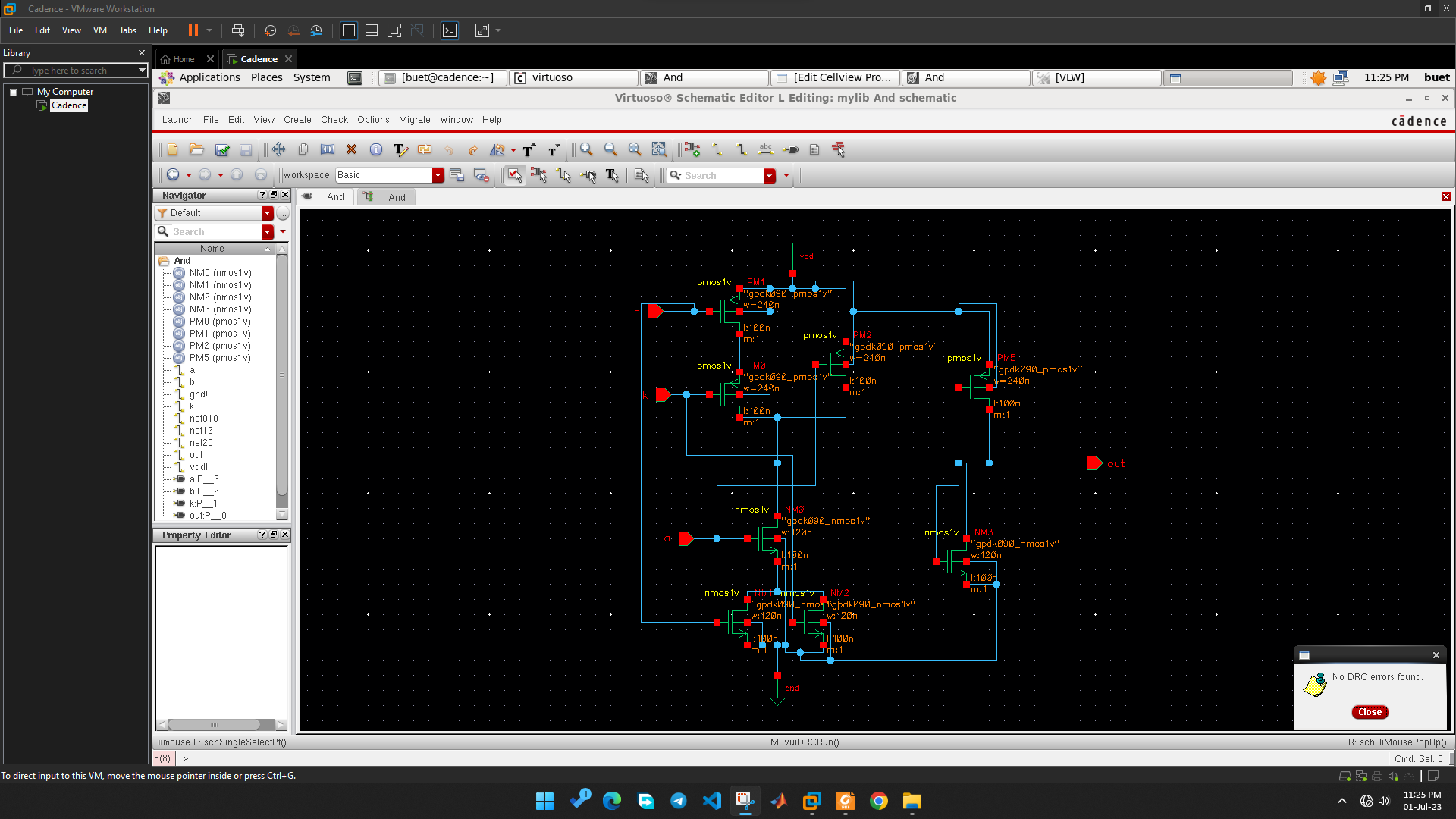
**Secure OR Layout DRC**

****

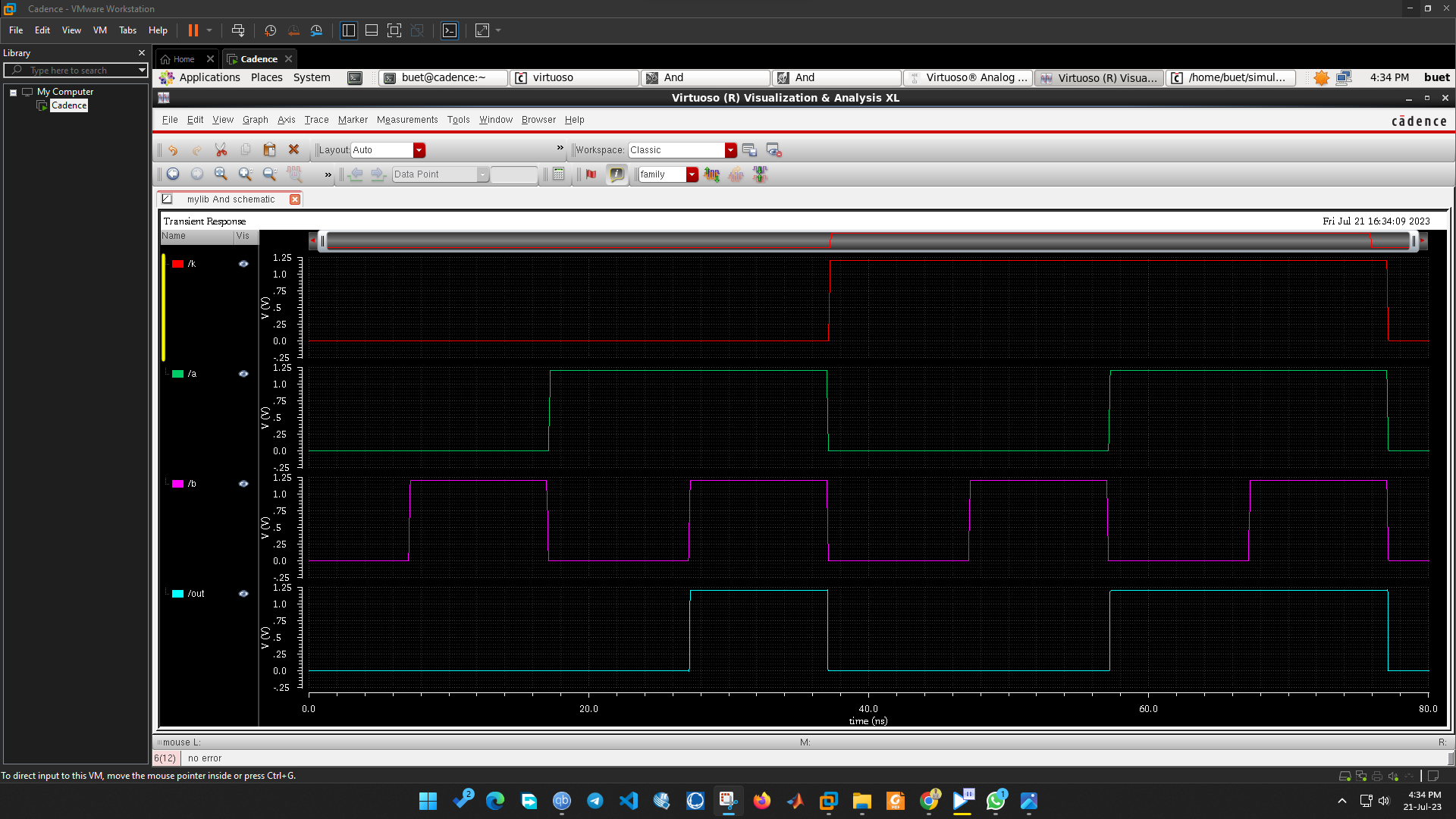
**Secure OR Layout LVS**

****

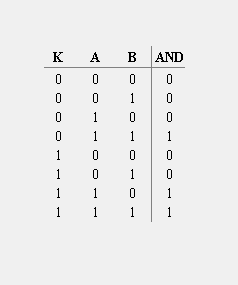
**Secure AND Schematic**

****

**Secure AND Waveform**

****

**Secure AND Truth Table**

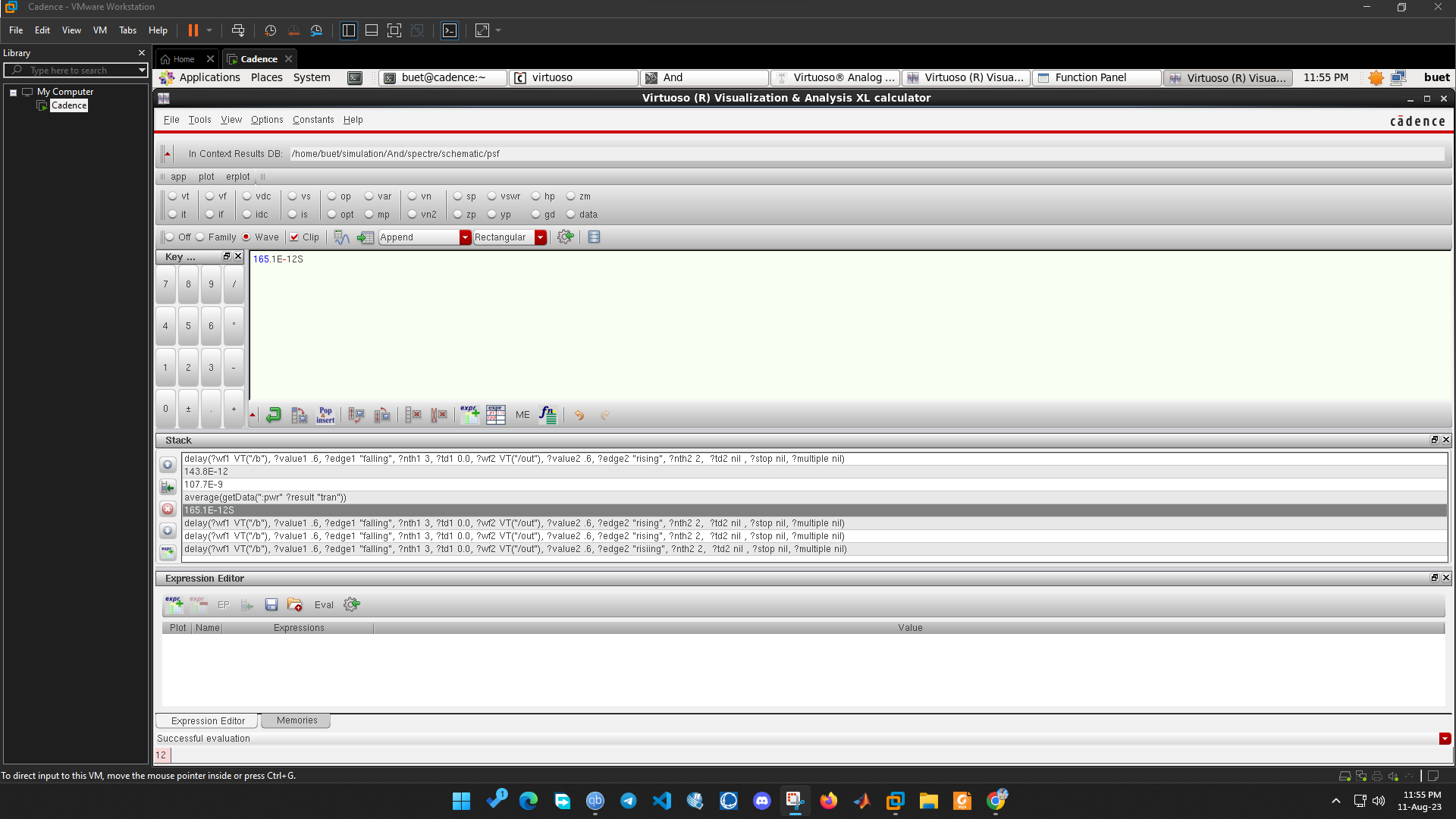
****

From the schematic diagram, we get the output function to be,

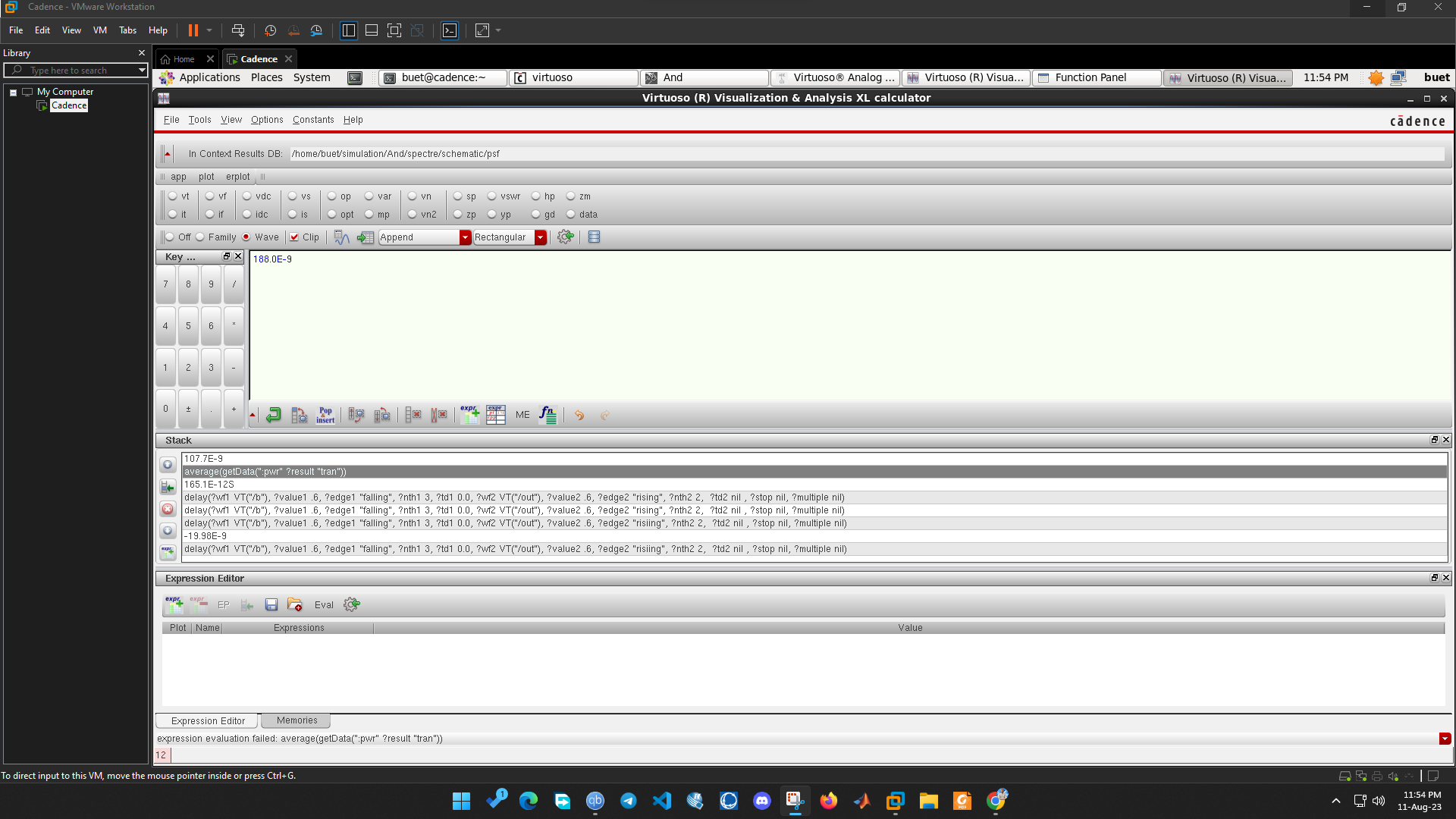
AND = A(B+K)

When the logic level of K is 0, we get the output as AND = A.B , which is the expected output of AND gate. Whereas, if K = 1, the output will not provide the expected values of AND gate. In this way, the privacy of this gate is ensured as without the particular value of K, the accurate result of AND gate will not be found.

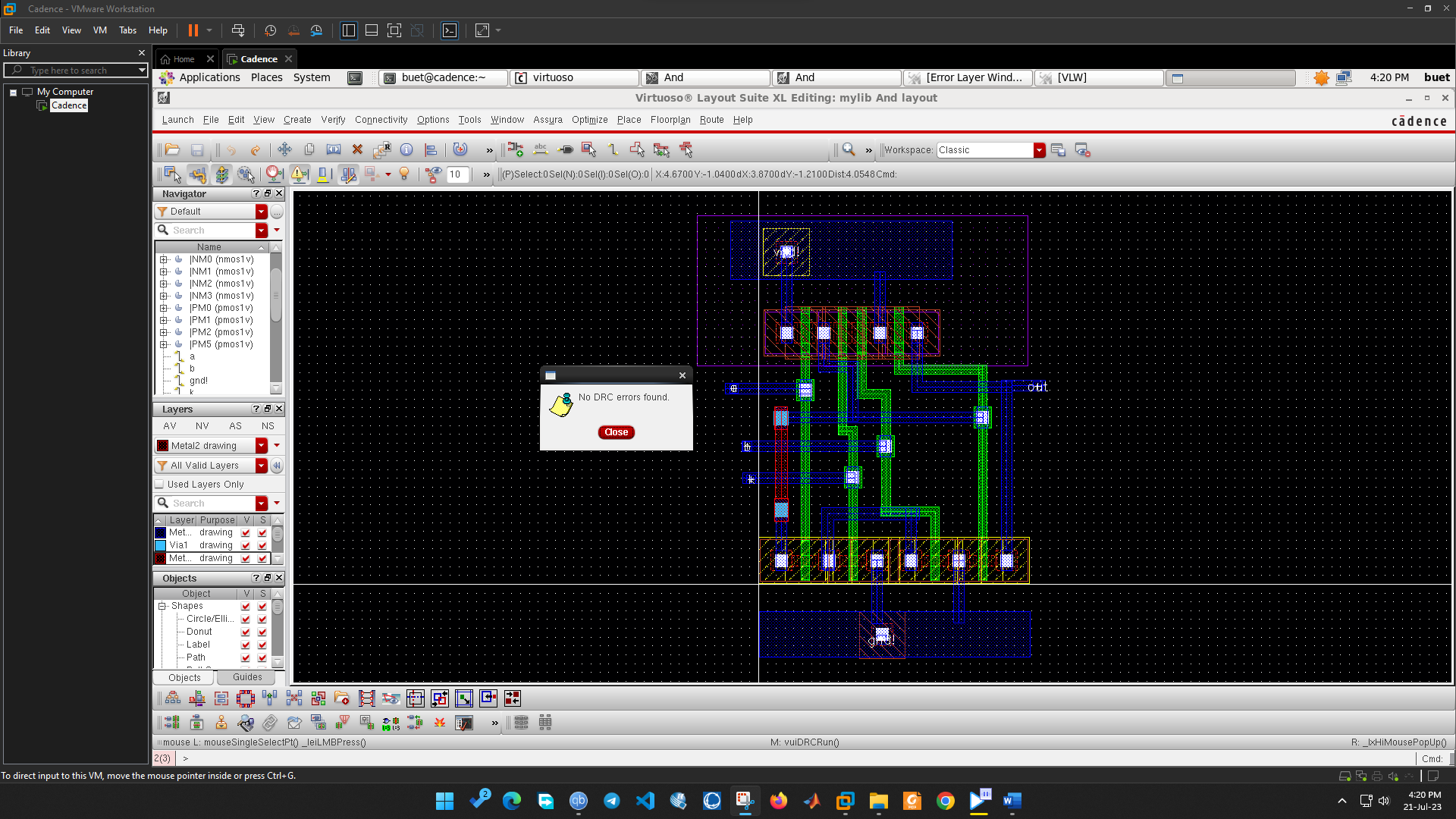
**Secure AND Propagation Delay**

****

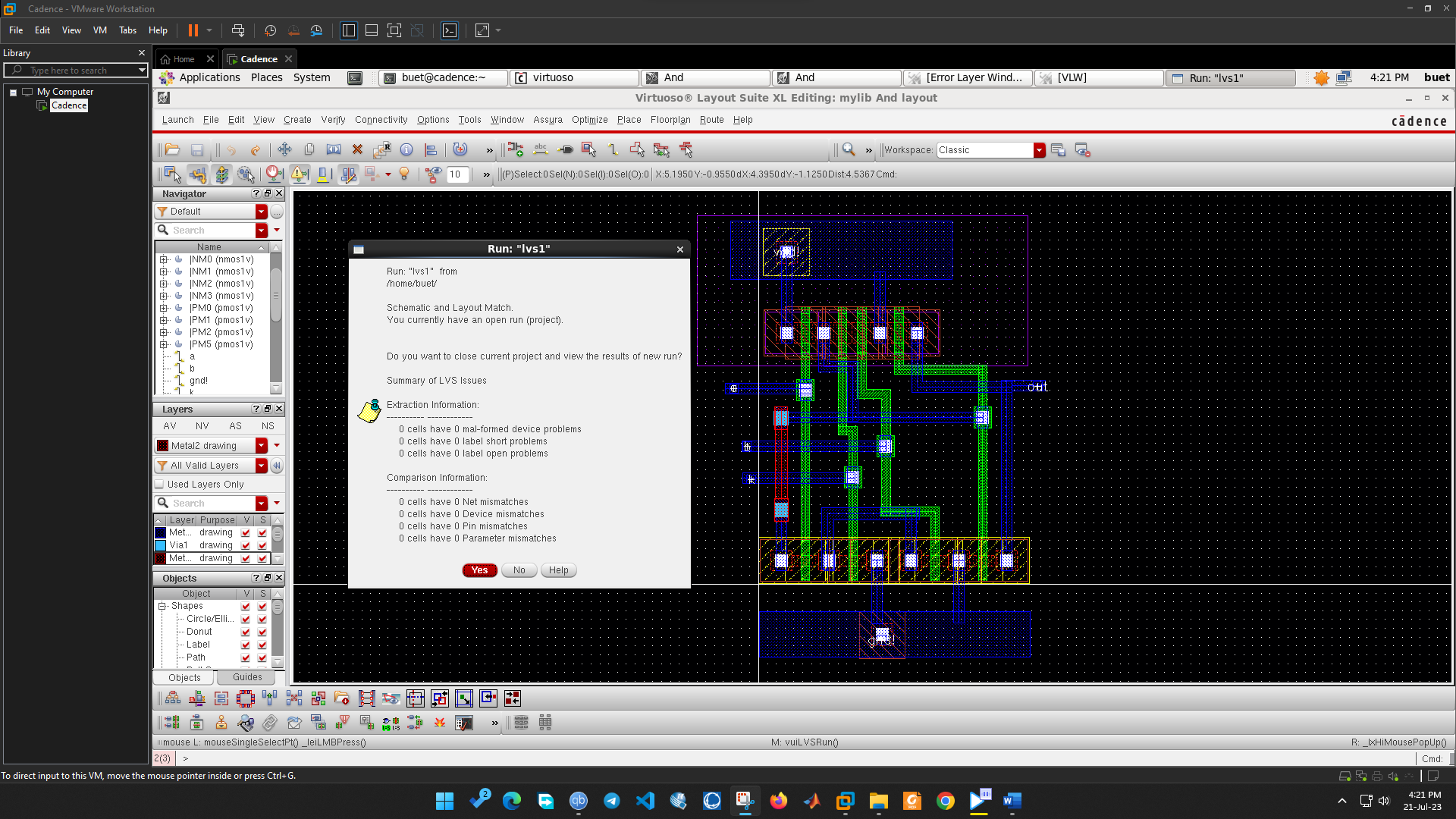
**Secure AND Average Power**

****

**Secure AND Layout DRC**

****

**Secure AND Layout LVS**

****

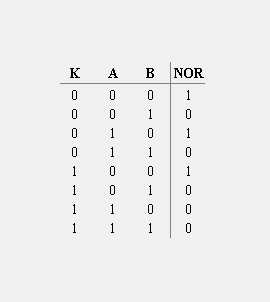
**Secure NOR Schematic**

****

**Secure NOR Waveform**

****

**Secure NOR Truth Table**

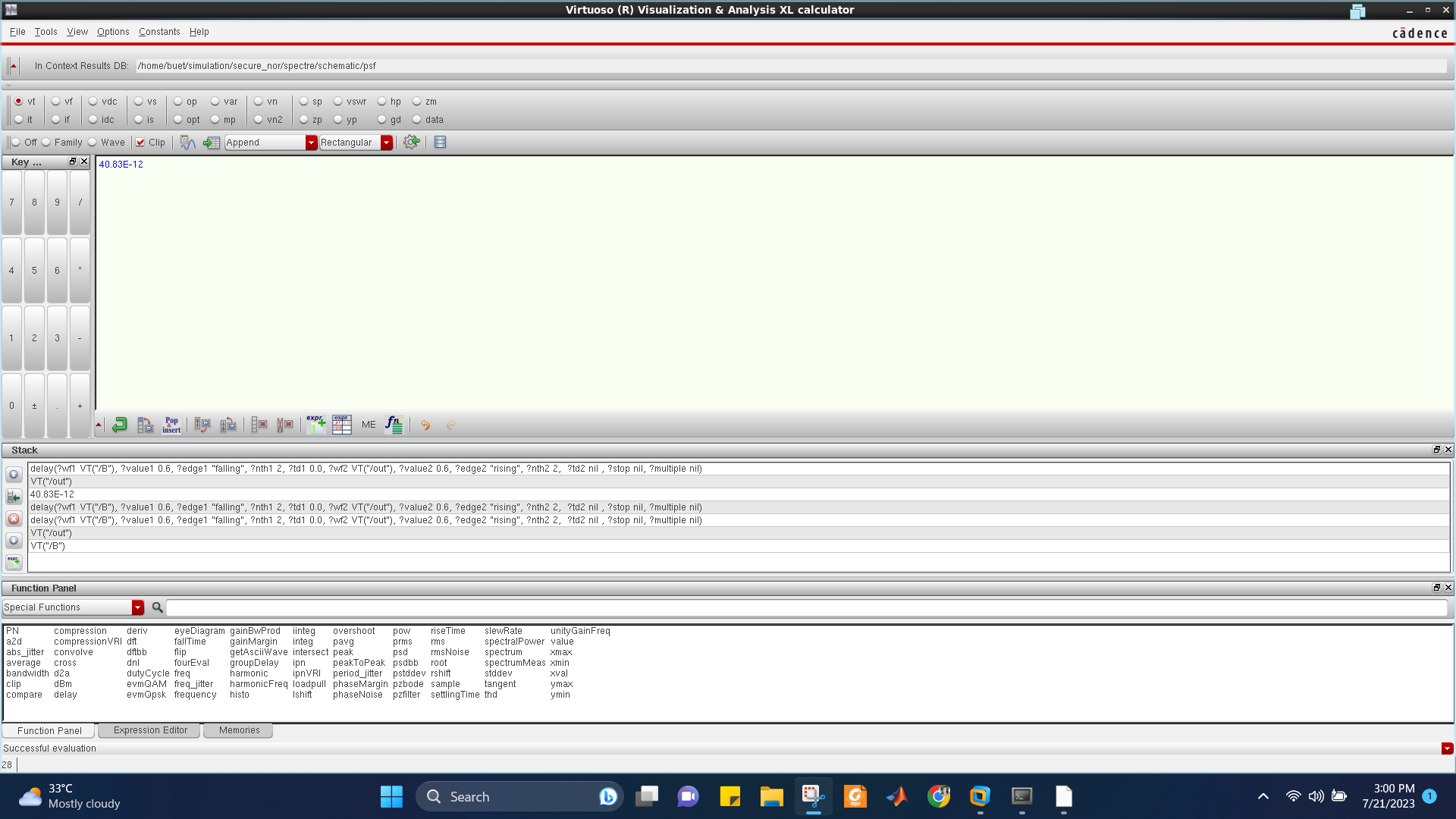
****

From the schematic diagram, we get the output function to be,

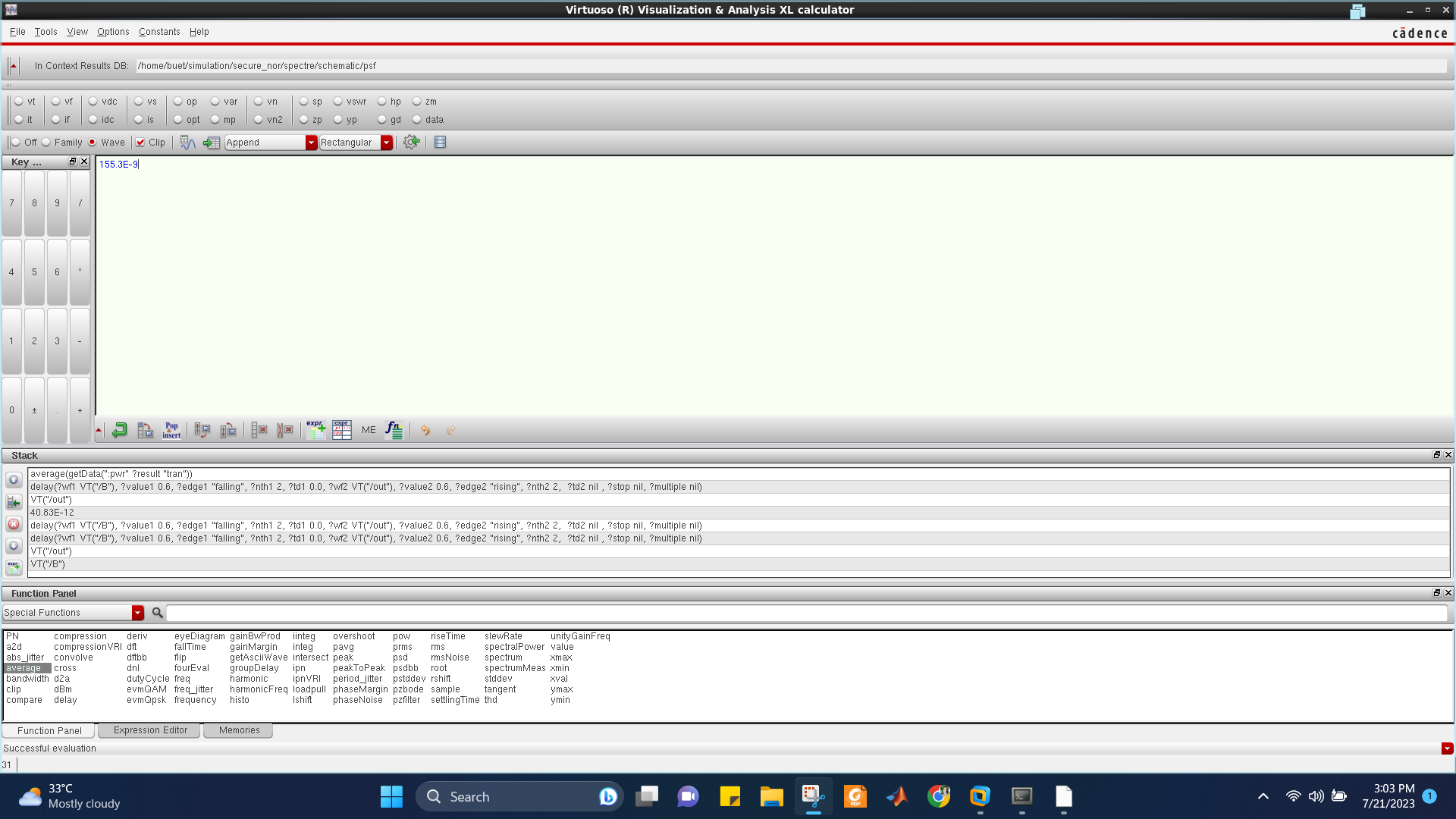
NOR =

When the logic level of K is 1, we get the output as NOR = , which is the expected output of NOR gate. Whereas, if K = 0, the output will not provide the expected values of NOR gate. In this way, the privacy of this gate is ensured as without the particular value of K, the accurate result of NOR gate will not be found.

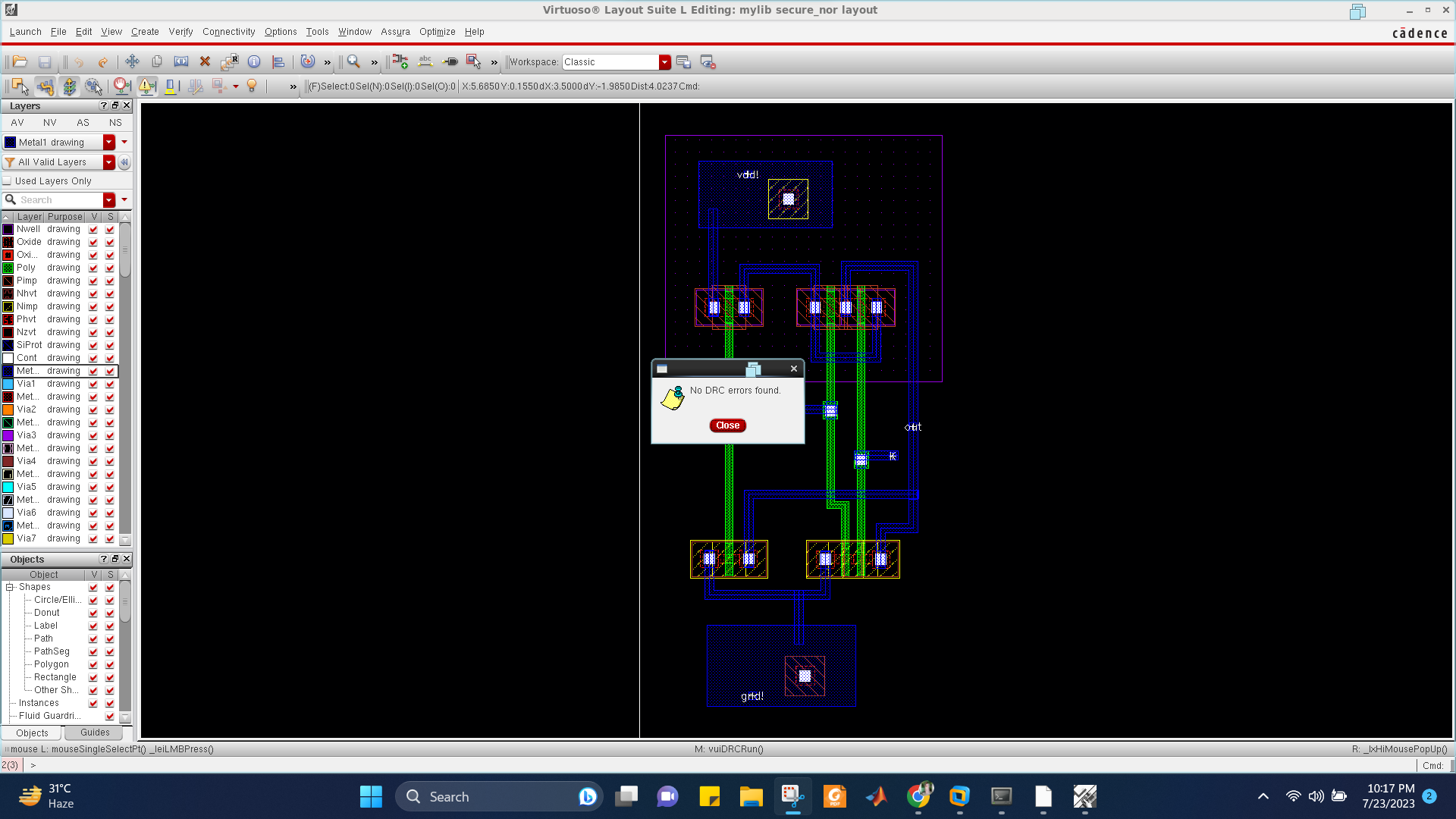
**Secure NOR Propagation Delay**

****

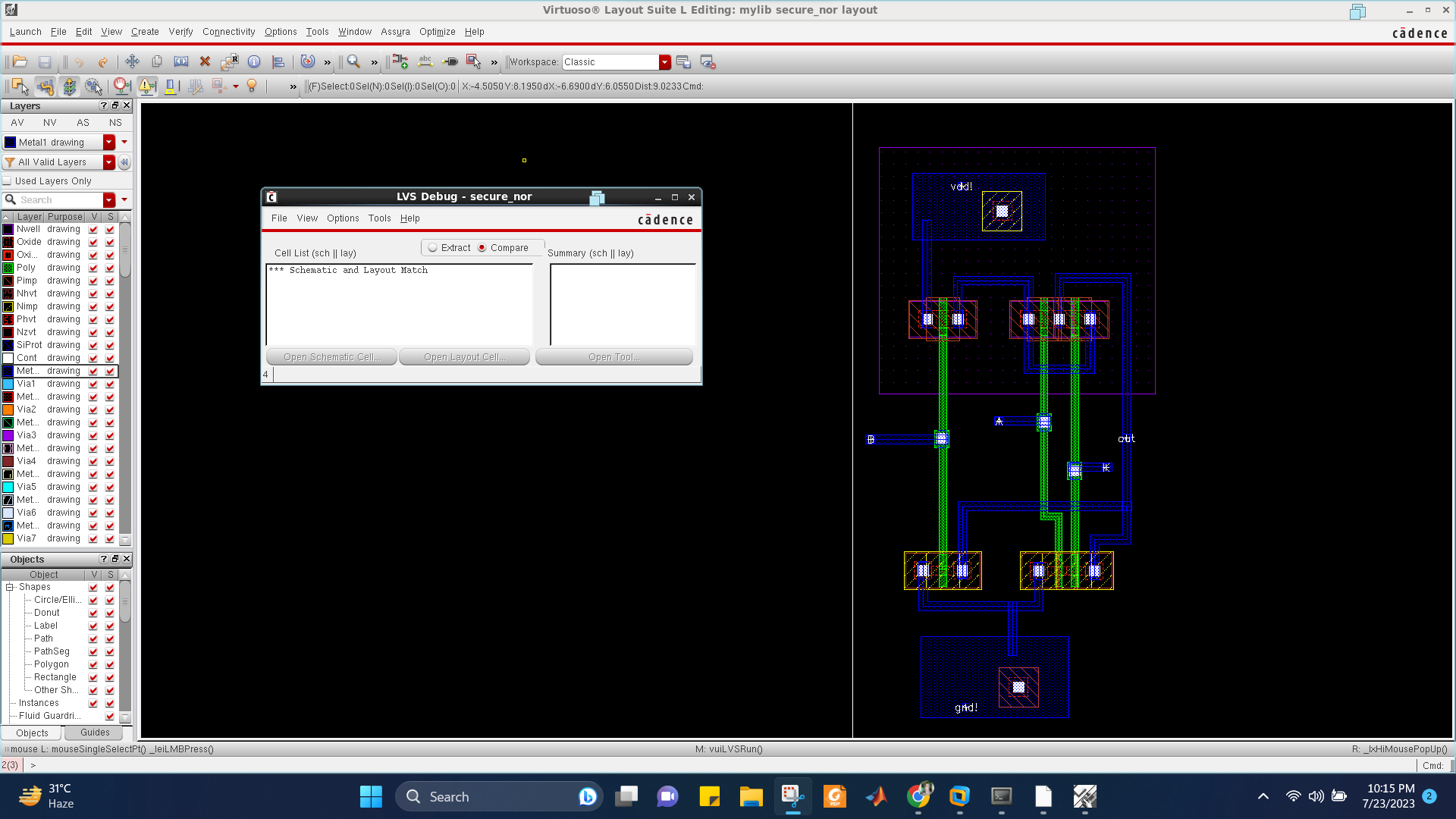
**Secure NOR Average Power**

****

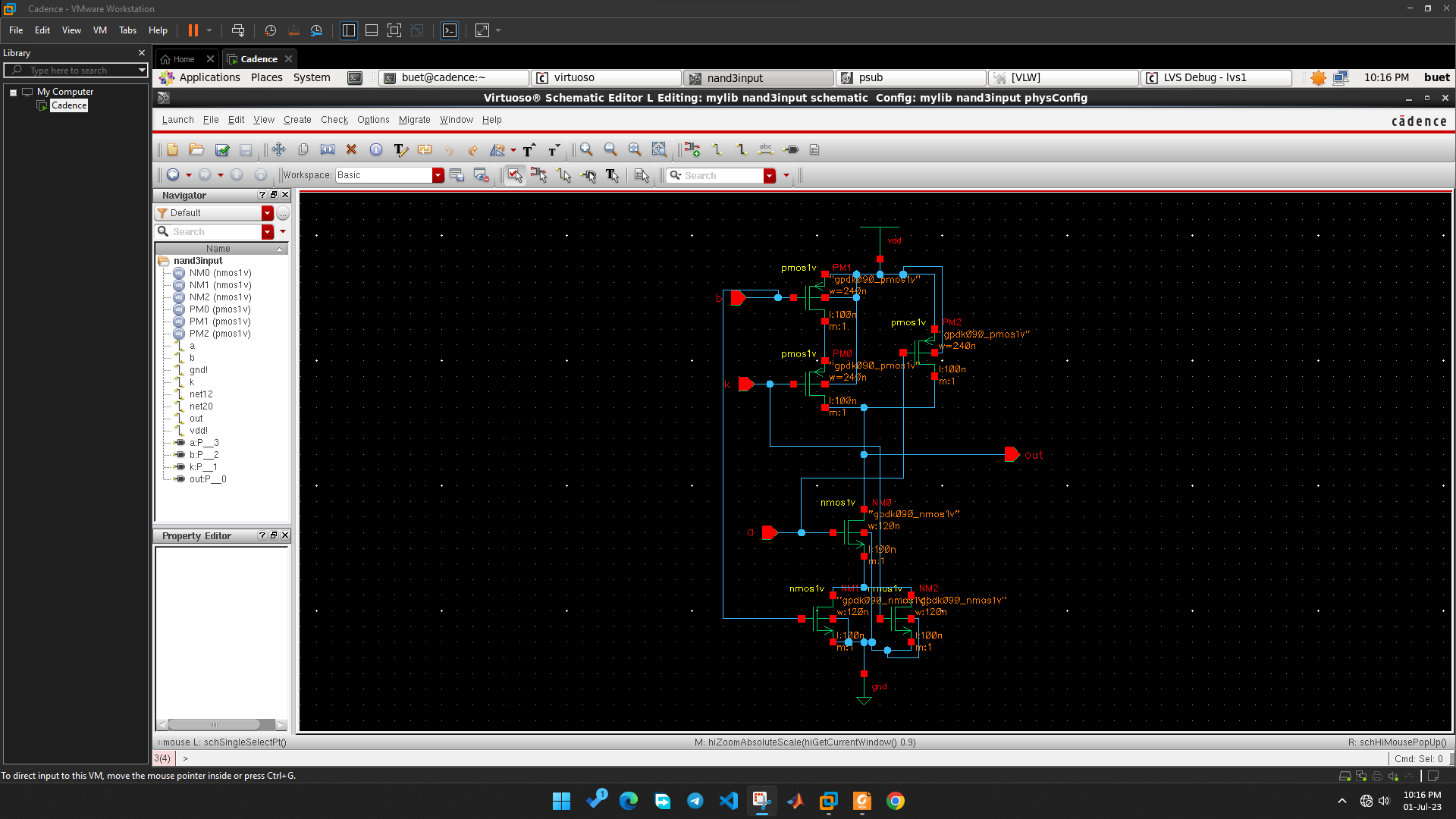
**Secure NOR Layout DRC**

****

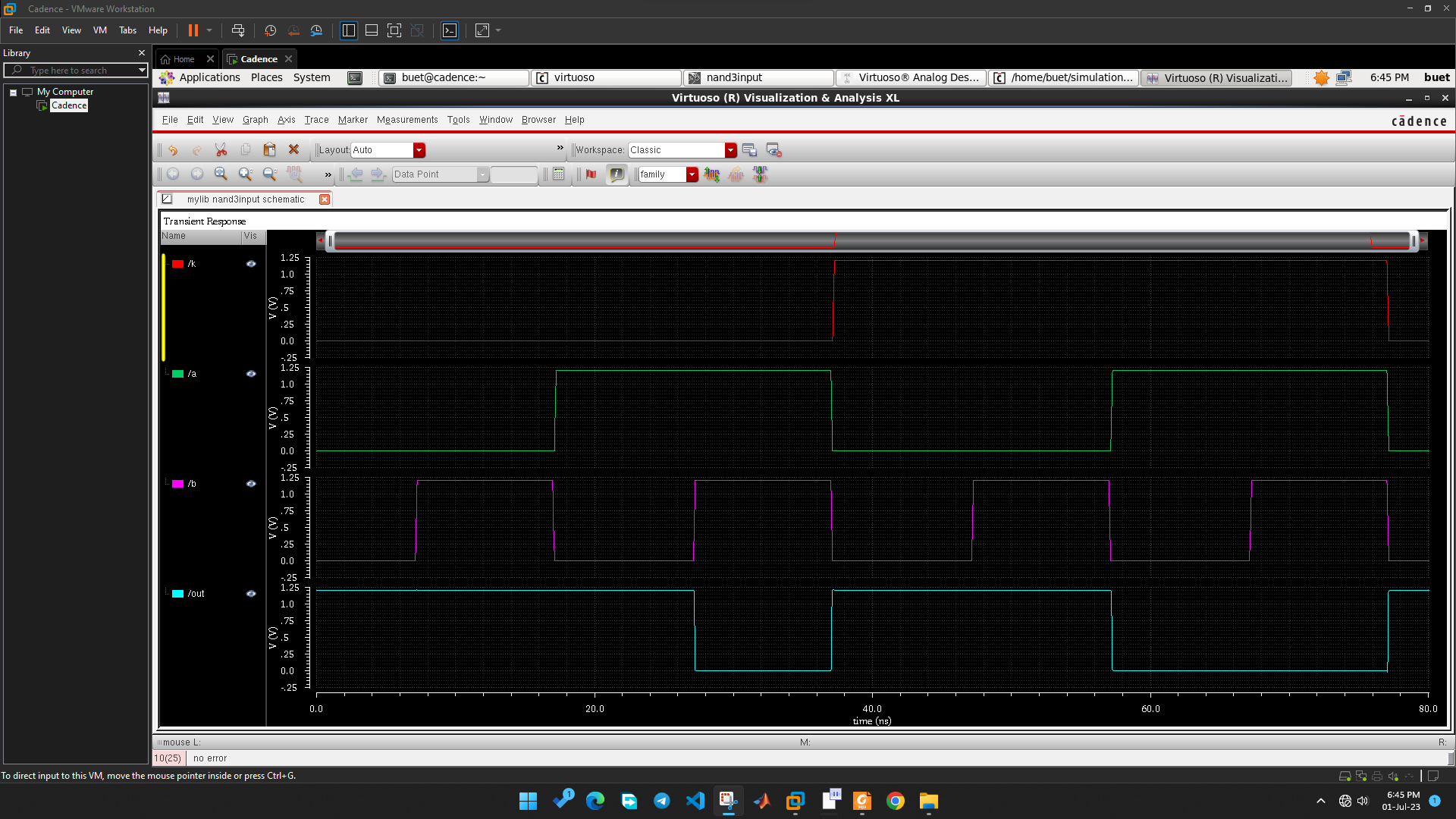
**Secure NOR Layout LVS**

****

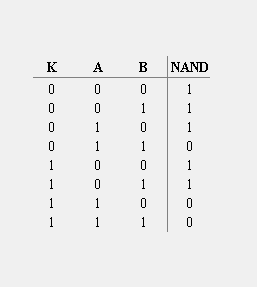
**Secure NAND Schematic**

****

**Secure NAND Waveform**

****

**Secure NAND Truth Table**

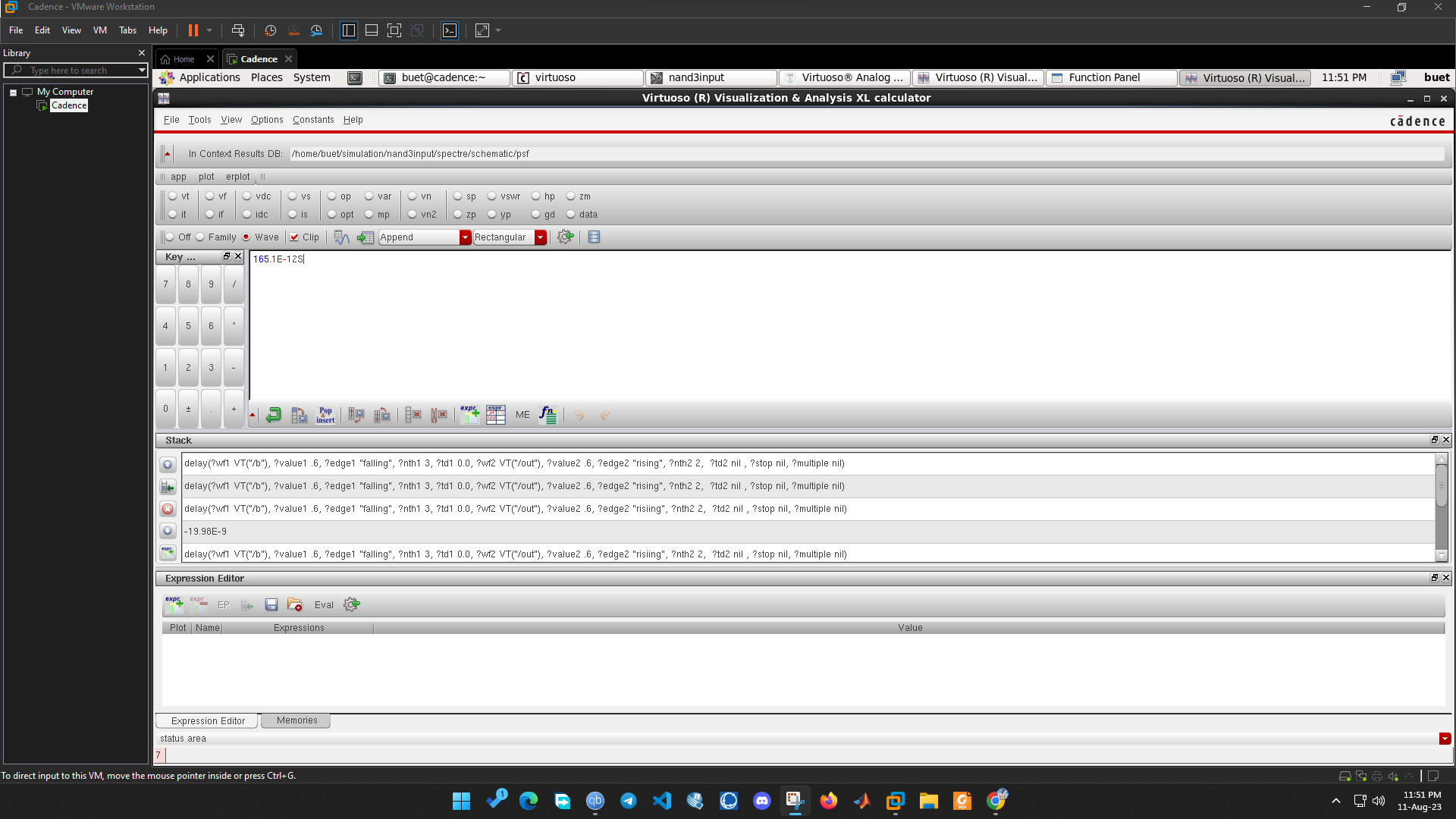
****

From the schematic diagram, we get the output function to be,

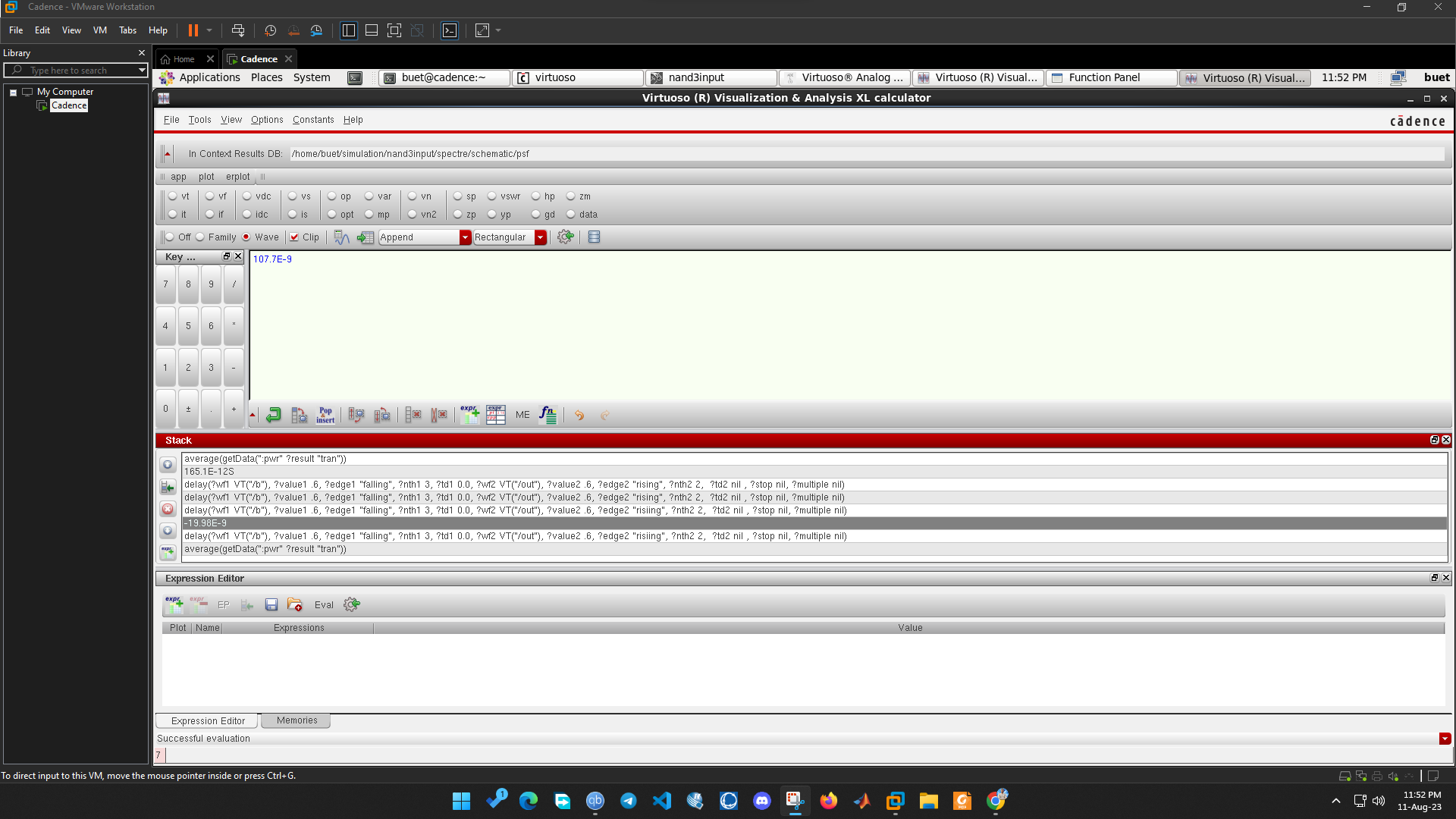
NAND =

When the logic level of K is 0, we get the output as NAND = , which is the expected output of NAND gate. Whereas, if K = 1, the output will not provide the expected values of NAND gate. In this way, the privacy of this gate is ensured as without the particular value of K, the accurate result of NAND gate will not be found.

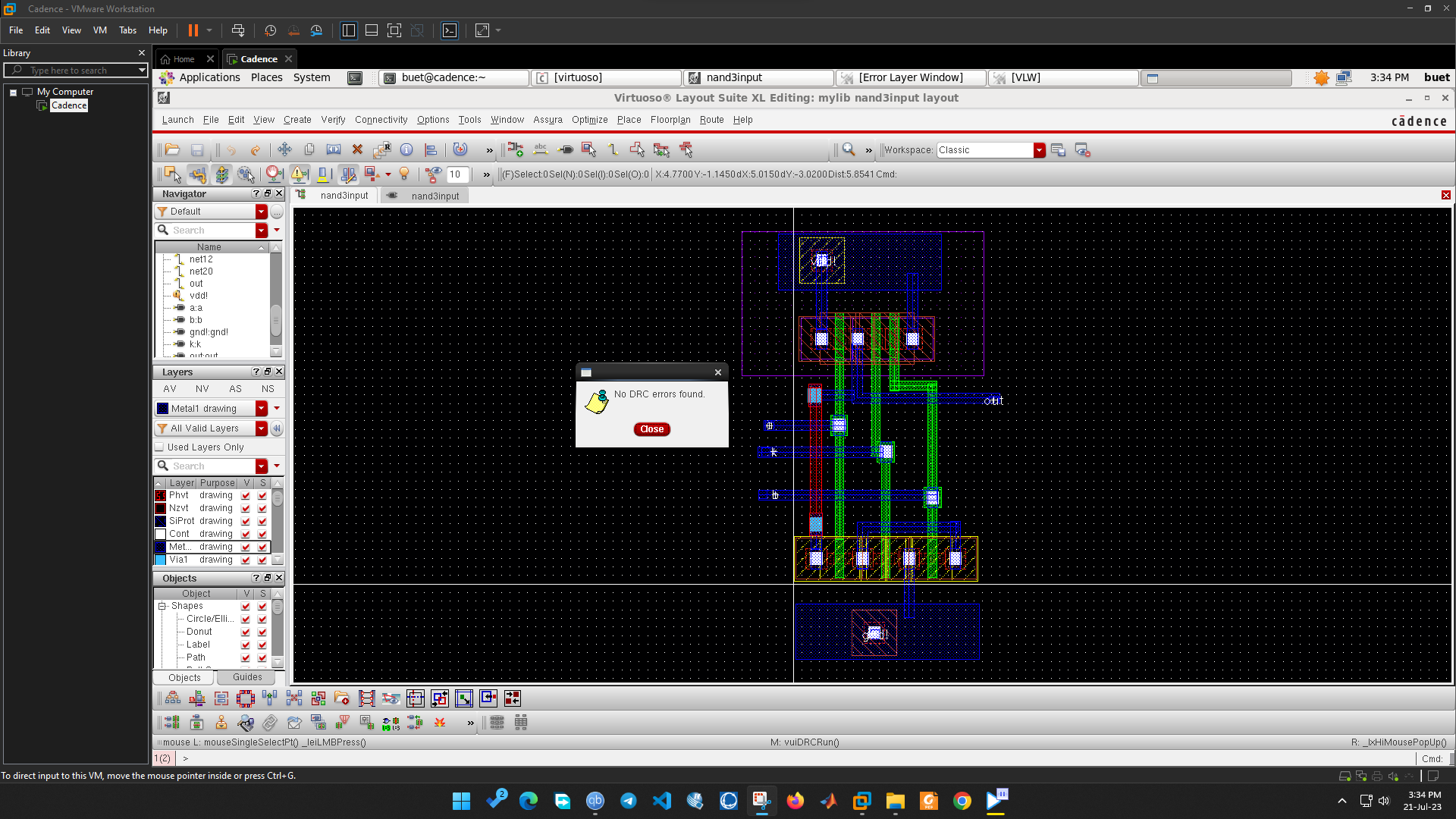
**Secure NAND Propagation Delay**

****

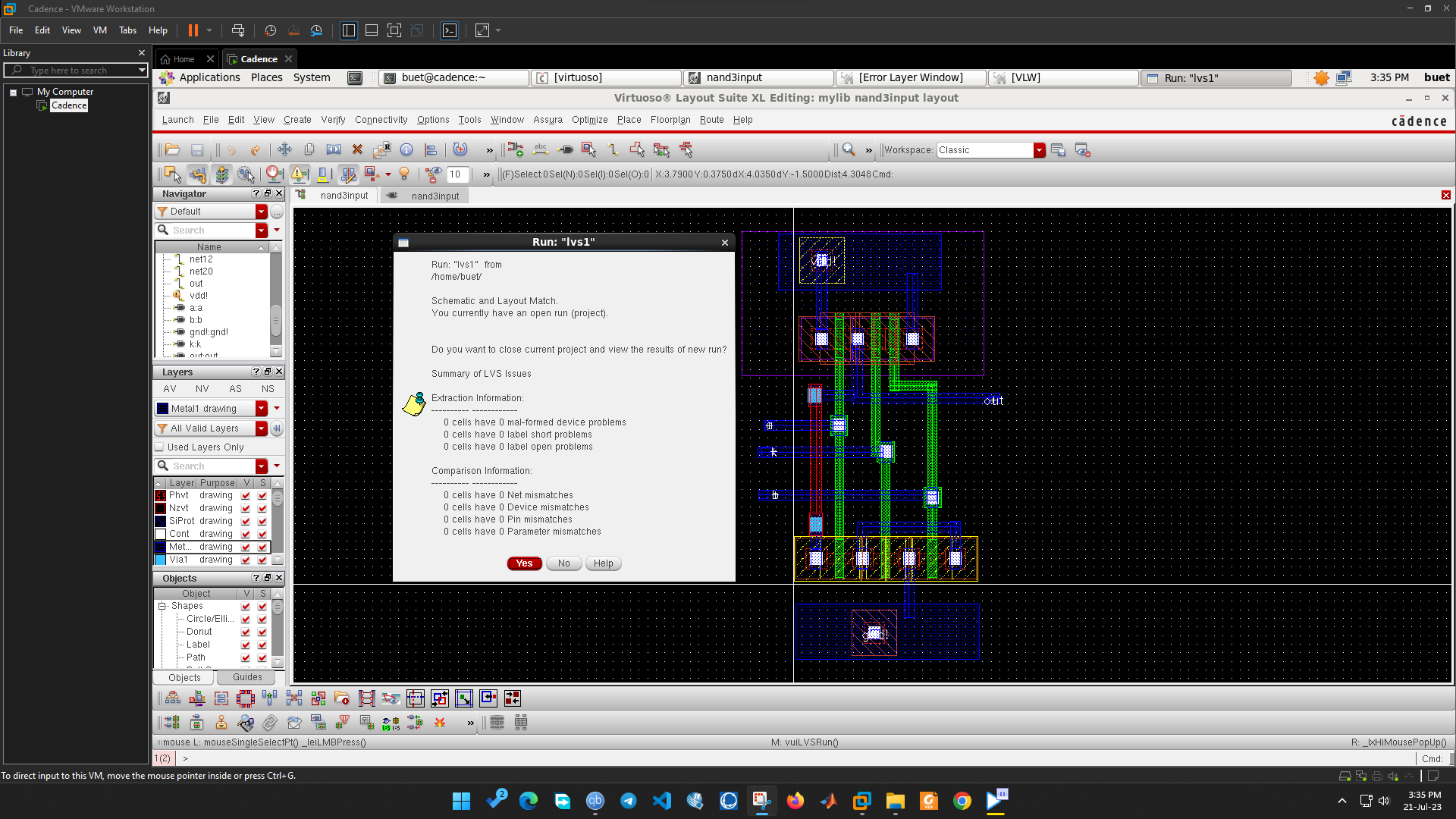
**Secure NAND Average Power**

****

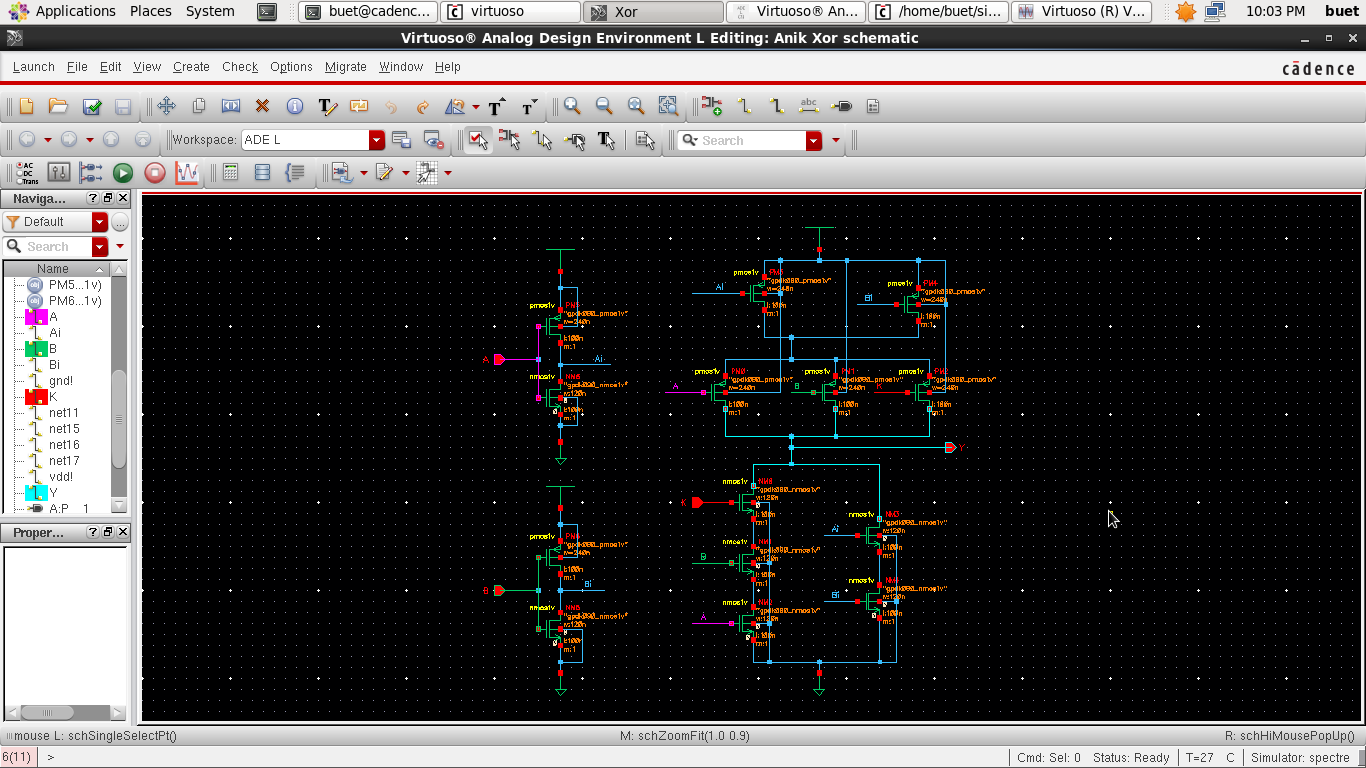
**Secure NAND Layout DRC**

****

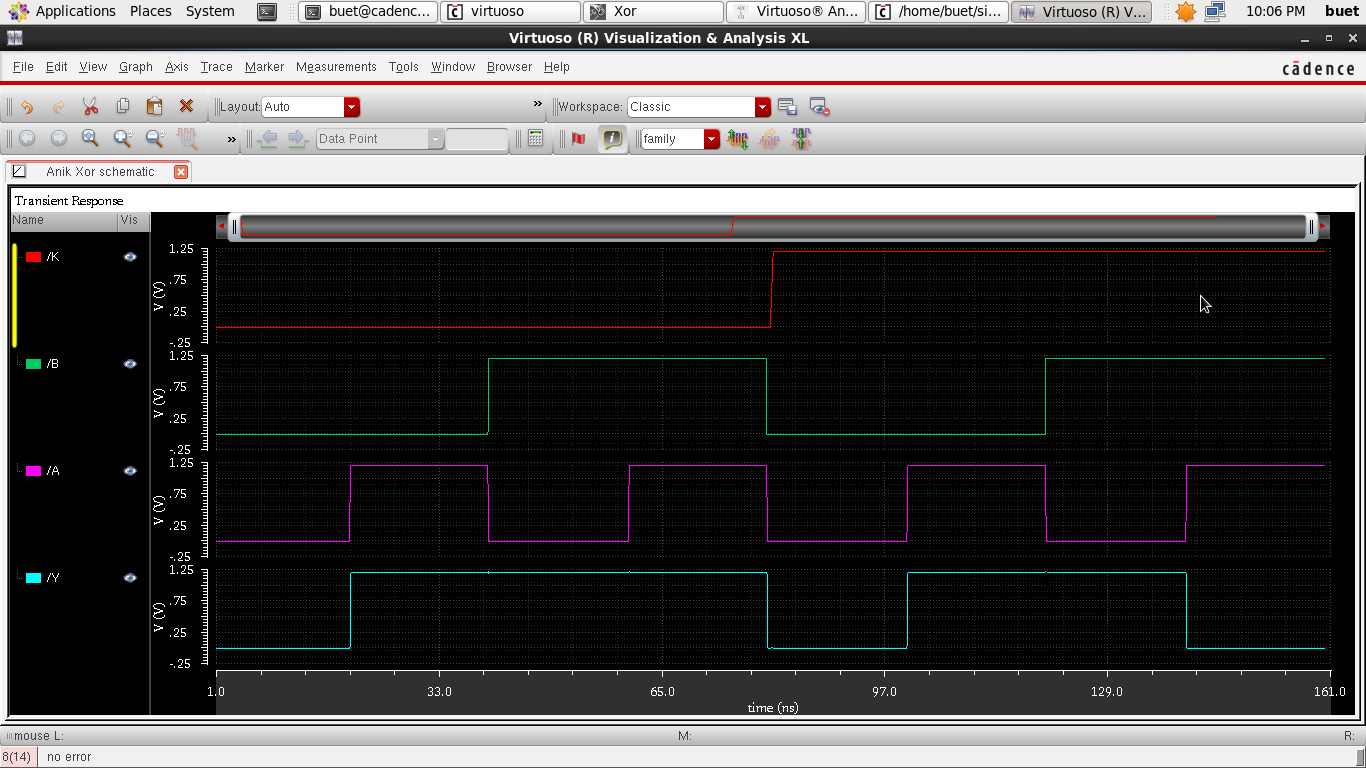
**Secure NAND Layout LVS**

****

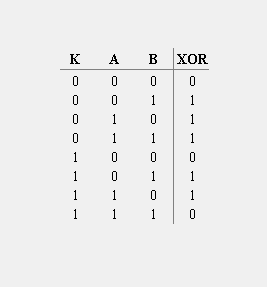
**Secure XOR Schematic**



**Secure XOR Waveform**



**Secure XOR Truth Table**

****

From the schematic diagram, we get the output function to be,

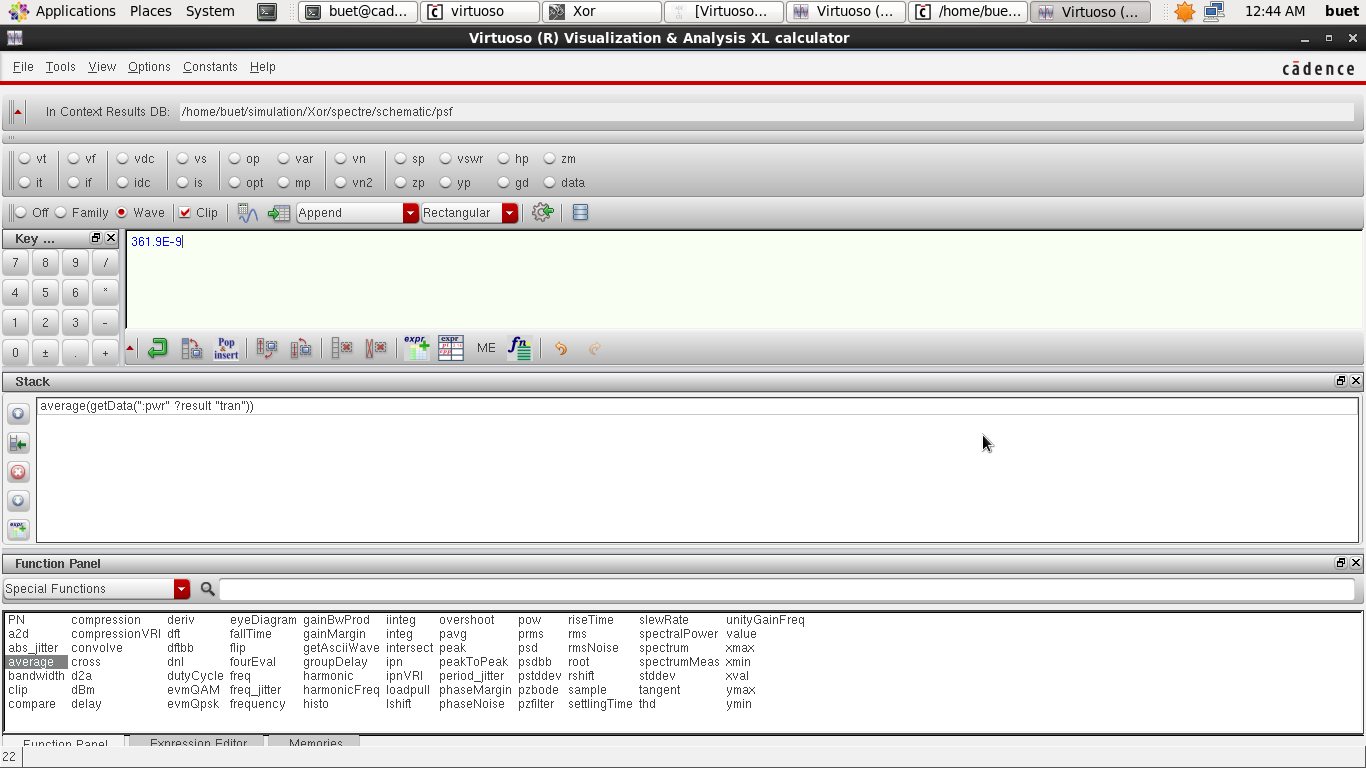
XOR =

When the logic level of K is 1, we get the output as XOR = , which is the expected output of XOR gate. Whereas, if K = 0, the output will not provide the expected values of XOR gate. In this way, the privacy of this gate is ensured as without the particular value of K, the accurate result of XOR gate will not be found.

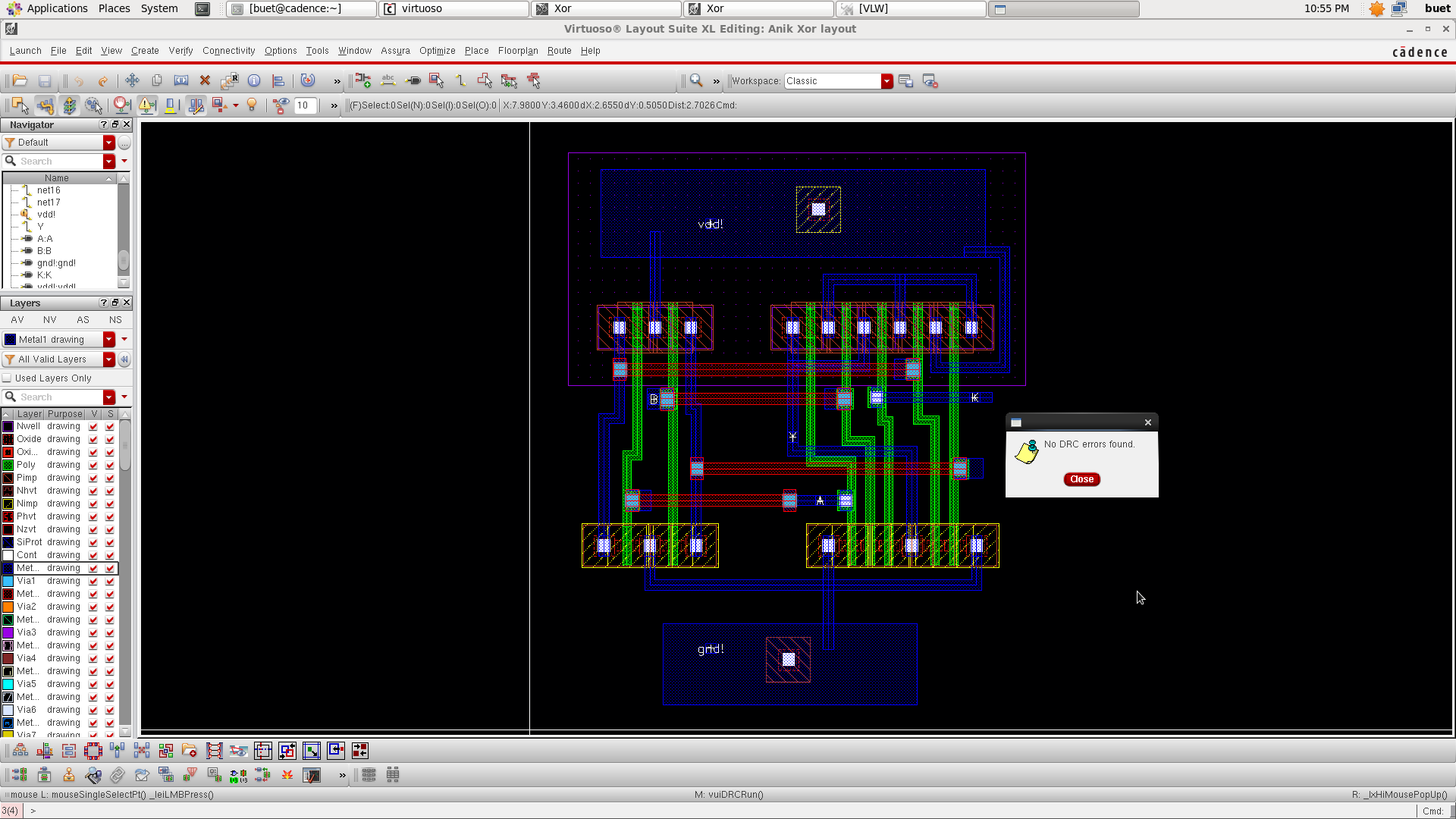
**Secure XOR Propagation Delay**



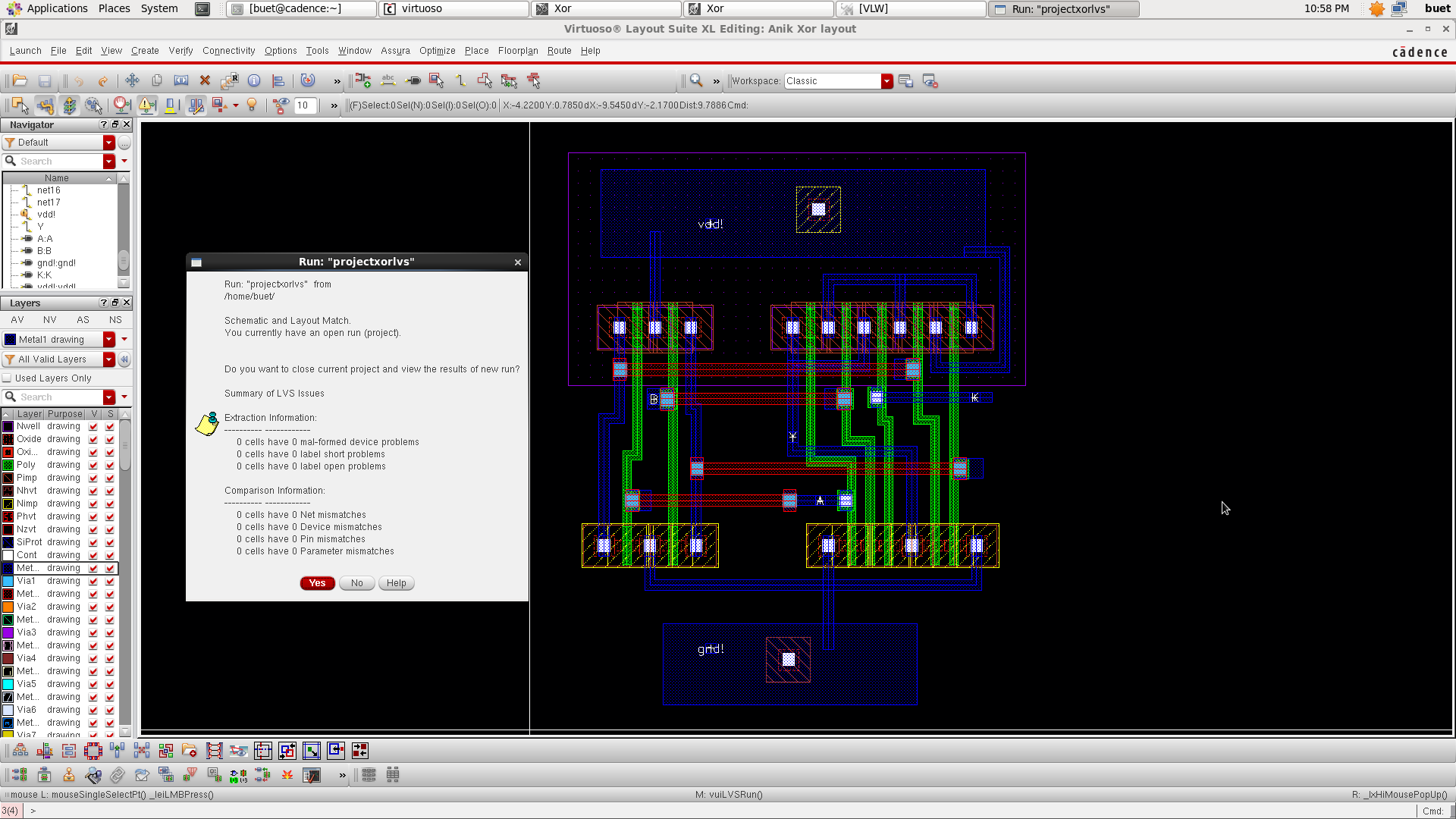
**Secure XOR Average Power**



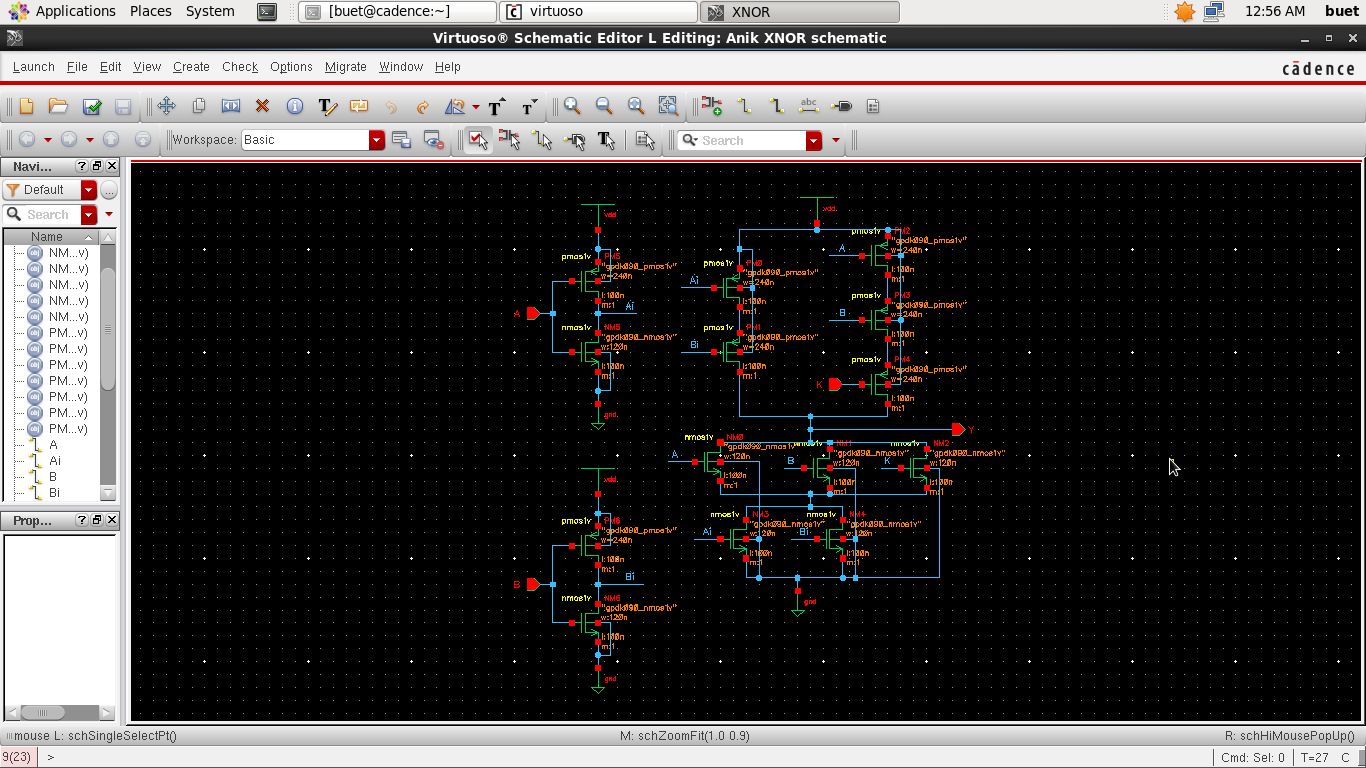
**Secure XOR Layout DRC**



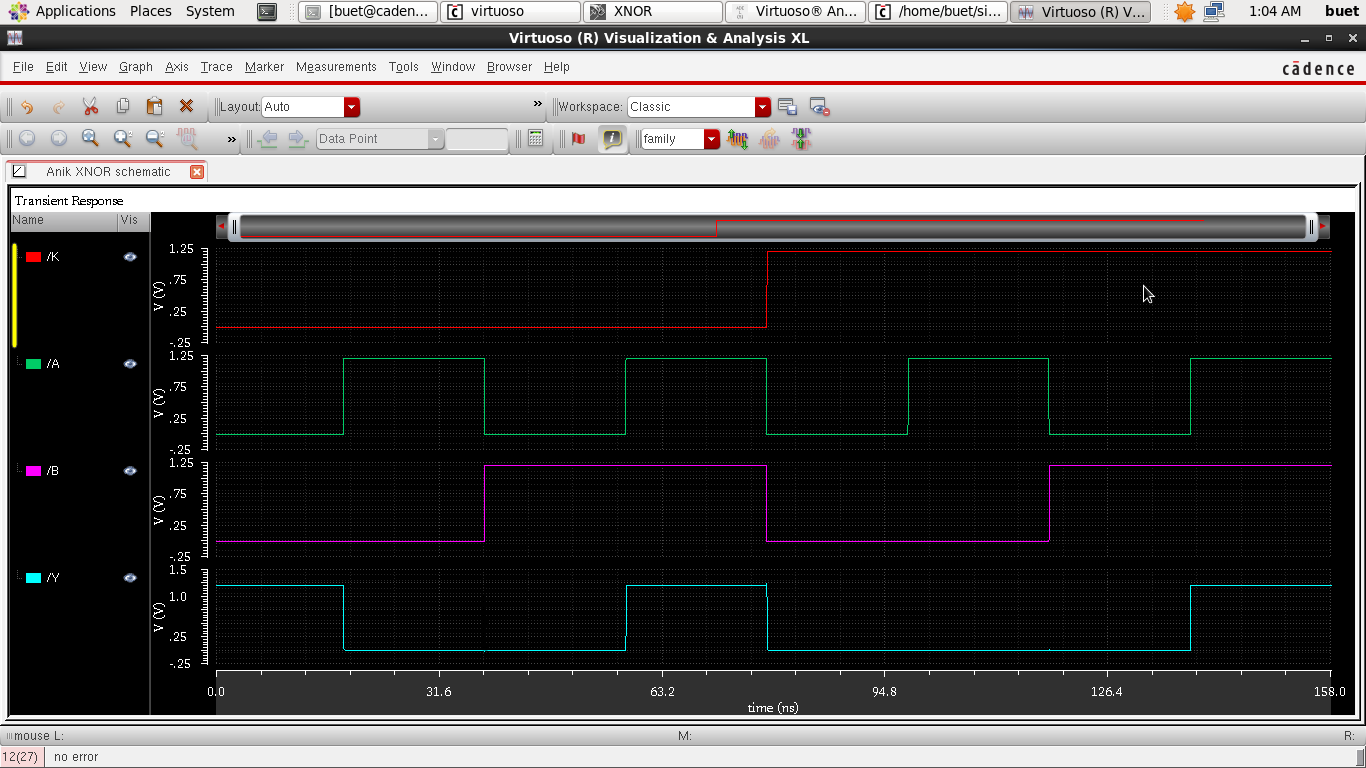
**Secure XOR Layout LVS**



**Secure XNOR Schematic**



**Secure XNOR Waveform**

****

**Secure XNOR Truth Table**

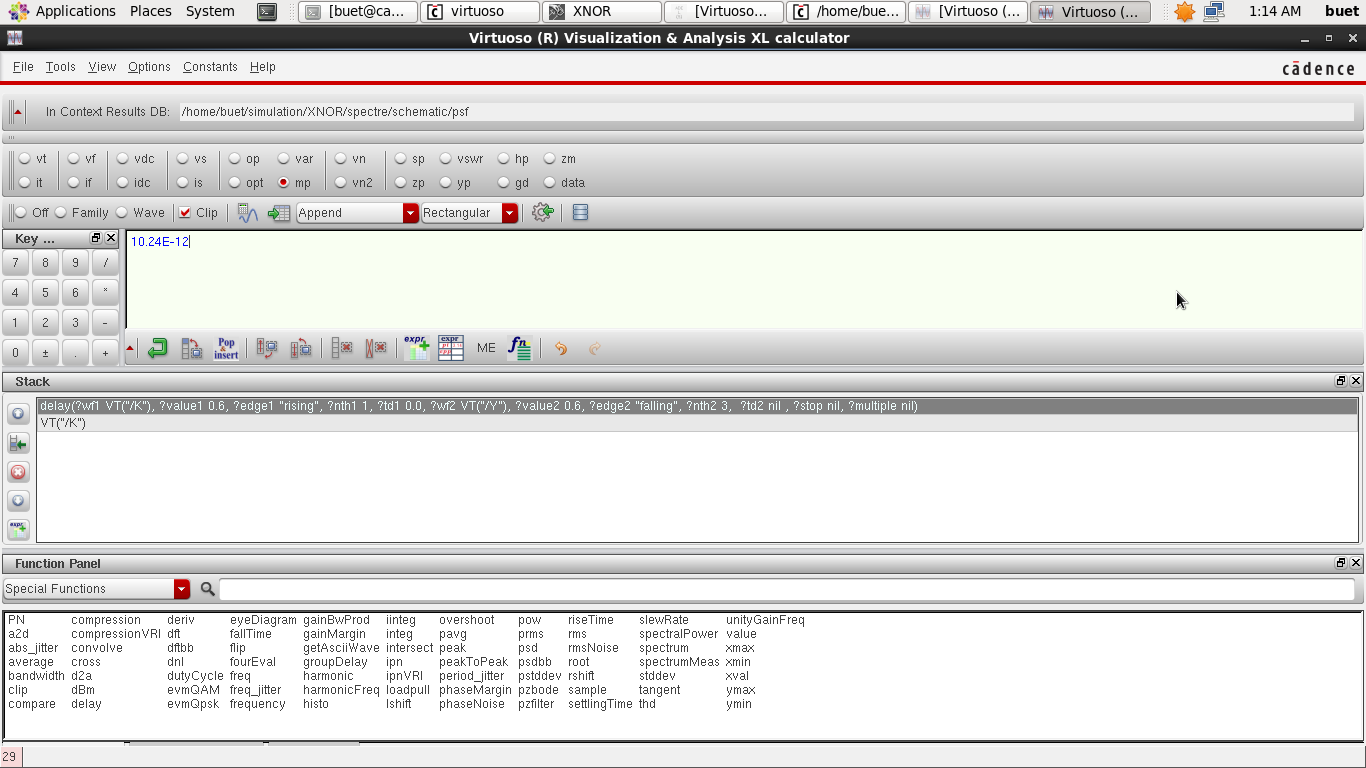
****

From the schematic diagram, we get the output function to be,

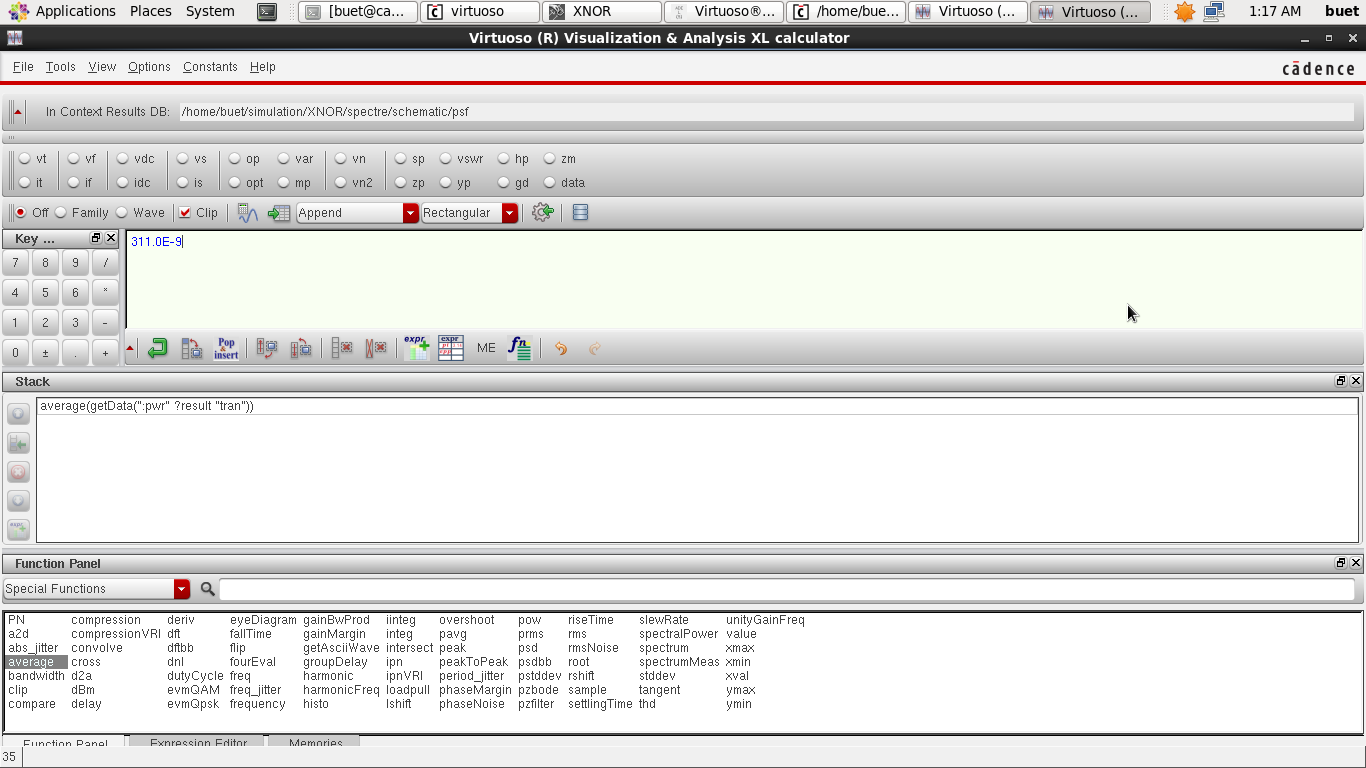
XNOR =

When the logic level of K is 0, we get the output as XNOR = , which is the expected output of XNOR gate. Whereas, if K = 1, the output will not provide the expected values of XNOR gate. In this way, the privacy of this gate is ensured as without the particular value of K, the accurate result of XNOR gate will not be found.

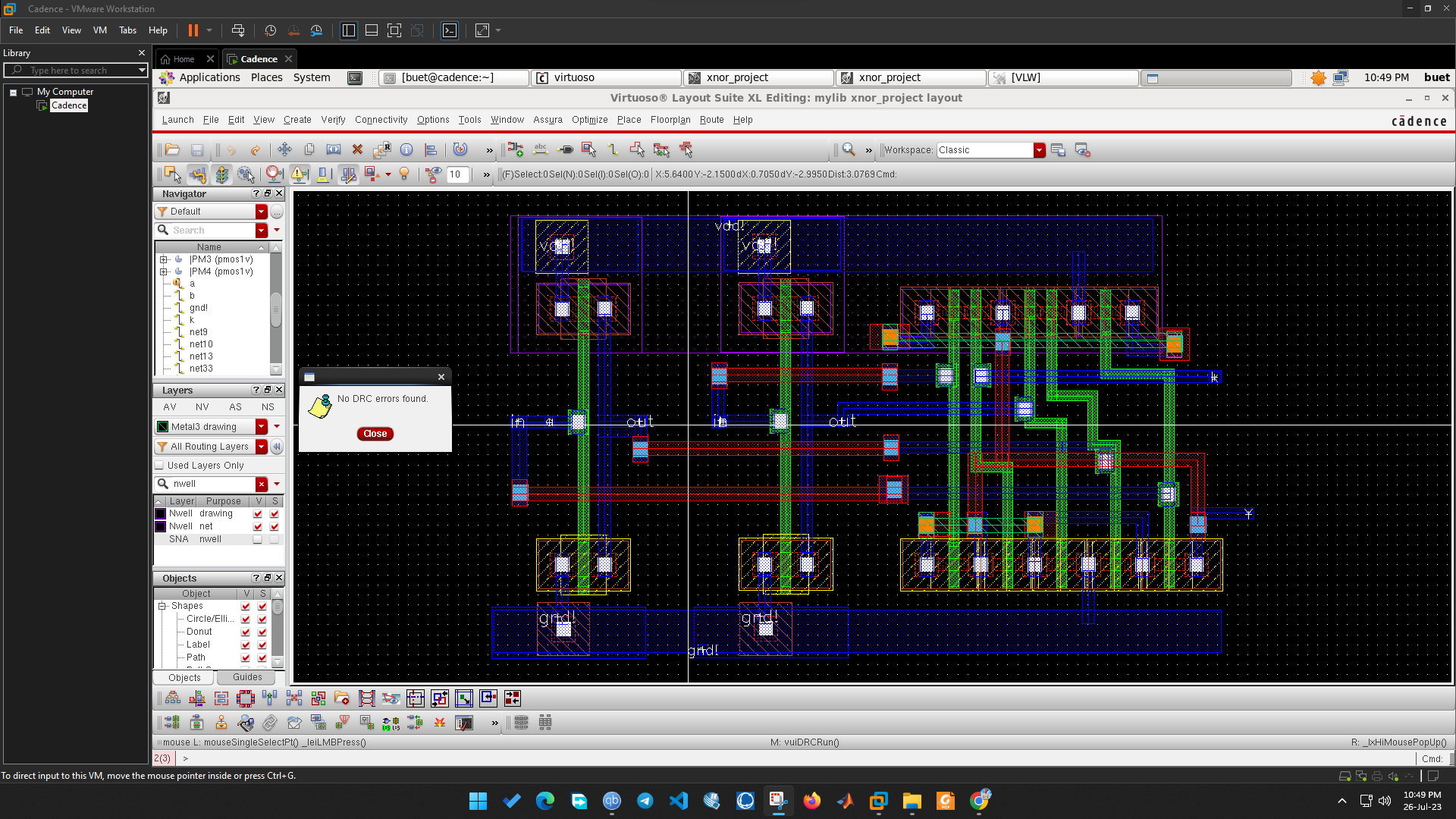
**Secure XNOR Propagation Delay**



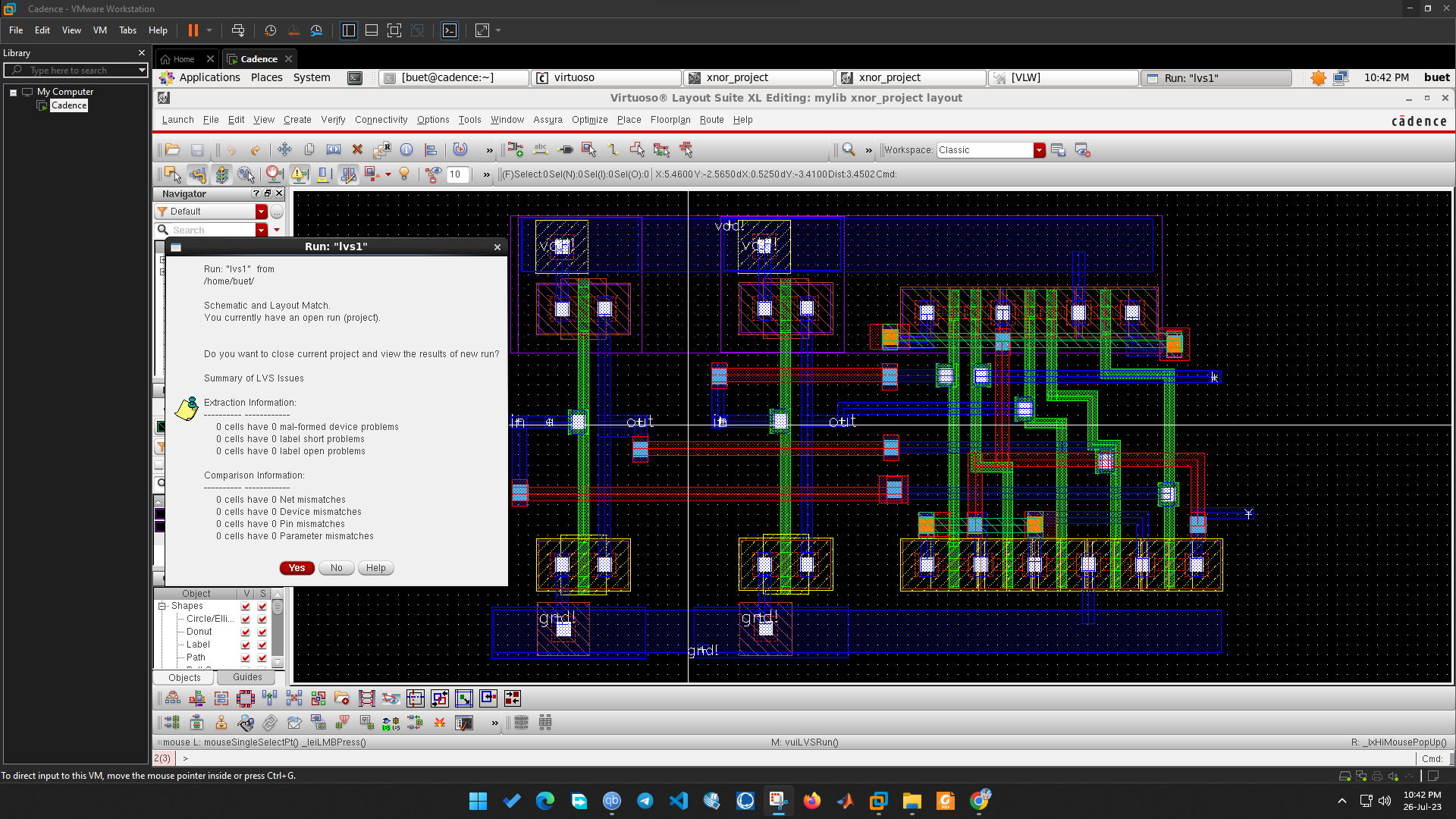
**Secure XNOR Average Power**



**Secure XNOR Layout DRC**

****

**Secure XNOR Layout LVS**

****

The proposed circuits are modeled using 90nm technology library in Cadence Virtuoso Tool. They have different logical behavior from each other. Circuit bafflement will confuse the attacker to figure out the gate functionalities. The most significant advantage of these circuits is, the original netlist cannot be obtained by the hackers as they will not insert a trojan into the netlist’s internal node. It is difficult to decrypt the circuit as the key is in-built. The inputs of the gates can be re-arranged in any manner and the functionality of the circuits would change accordingly. Theses circuits can be used for security purpose in different aspects like changing design structure, hiding circuit functionality, masking output probability etc. The introduction of multi-gate functionality increases the hardware security. They are also able to minimize energy (power delay product) consumption which is one of the major advantages over other existing topologies.

**Future Aspects:**

Day by day, the necessity of hardware security is increasing because of the rise in intellectual property piracy. Therefore, the significance of logic encryption to protect circuits as well as specific secrets of design is increasing rapidly. Though, the proposed methodology for logic encryption is executed here for some simple logic gates (OR, AND, NOR, NAND, XOR, XNOR) designed with CMOS, it can be used for various complex circuits for maintaining the privacy along with efficient designing of circuits.

**Conclusion:**

In this project, a new topology for CMOS gates at the transistor level is designed regarding the necessity of hardware security. The gates have an efficient structure to prevent intellectual property piracy, IC counterfeiting and reverse engineering. It was also observed that the proposed circuits reduce the area, delay, save power and energy compared to the existing methodologies of logic encryption. Thus, a trade-off between circuit performance for security purpose and design efficiency is ensured.

**Reference:**

Chandra, S.S., Kannan, R.J., Balaji, B.S. *et al.* Efficient design and analysis of secure CMOS logic through logic encryption. *Sci Rep* 13, 1145 (2023). https://doi.org/10.1038/s41598-023-28007-2