Static CMOS Design

The most widely used logic style is static complementary CMOS. The static CMOS style is really an extension of the static CMOS inverter to multiple inputs. The primary advantage of the CMOS structure is robustness (i.e., low sensitivity to noise), good performance, and low power consumption with no static power dissipation. Most of those properties are carried over to large fan-in logic gates implemented using a similar circuit topology.

The complementary CMOS circuit style falls under a broad class of logic circuits called static circuits in which at every point in time (except during the switching transients), each gate output is connected to either VDD or Vss via a low-resistance path. Also, the outputs of the gates assume at all times the value of the Boolean function implemented by the circuit (ignoring, once again, the transient effects during switching periods). This is in contrast to the dynamic circuit class, which relies on temporary storage of signal values on the capacitance of high-impedance circuit nodes. The latter approach has the advantage that the resulting gate is simpler and faster. Its design and operation are however more involved and prone to failure due to an increased sensitivity to noise. The design of various static circuit flavors includes complementary CMOS, ratioed logic (pseudo-NMOS and DCVSL), and pass transistor logic.

- **Complementary CMOS**

A static CMOS gate is a combination of two networks, called the pull-up network (PUN)and the pull-down network (PDN) (Figure 1). The figure shows a generic N input logic gate where all inputs are distributed to both the pull-up and pull-down networks. The function of the PUN is to provide a connection between the output and VDD anytime the output of the logic gate is meant to be 1 (based on the inputs). Similarly, the function of the PDN is to connect the output to VSS when the output of the logic gate is meant to be 0. The PUN and PDN networks

are constructed in a mutually exclusive fashion such that one and only one of the networks is conducting in steady state. In this way, once the transients have settled, a path always exists between VDD and the output F, realizing a high output ("one"), or, alternatively, between VSS and F for a low output ("zero"). This is equivalent to stating that the output node is always a low- impedance node in steady

state.

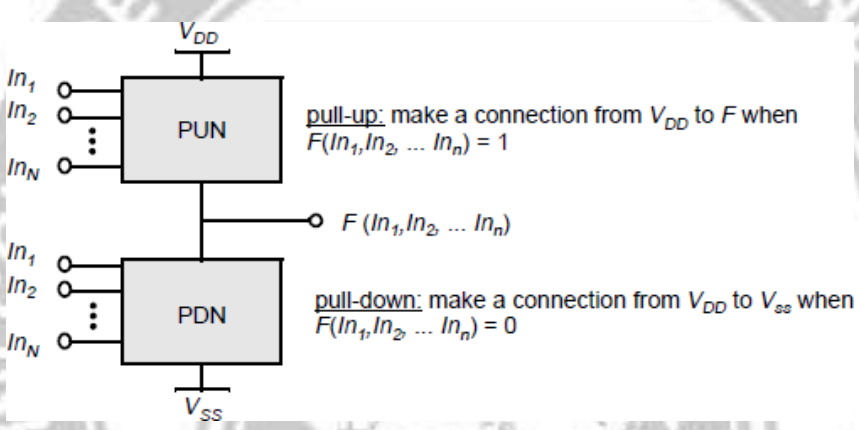

Figure 2.1.1: Complementary logic gate as a combination of a PUN (pull-up network) and a PDN (pull-down network)

[Source :Neil H.E. Weste, David Money Harris —CMOS VLSI Design: A Circuits and Systems Perspective…]

In constructing the PDN and PUN networks, the following observations should bekept in mind:

- A transistor can be thought of as a switch controlled by its gate signal. An NMOS switch is on when the controlling signal is high and is off when the controlling signal is low. A PMOS transistor acts as an inverse switch that is on when the controlling signal is low and off when the controlling signal is high.
- The PDN is constructed using NMOS devices, while PMOS transistors are used in the PUN. The primary reason for this choice is that NMOS transistors produce

"strong zeros," and PMOS devices generate "strong ones". To illustrate this, consider the examples shown in Figure 2. In Figure 2.a, the output capacitance is initially charged to VDD. Two possible discharge scenarios are shown. An NMOS device pulls the output all the way down to GND, while a PMOS lowers the output no further than |VTp| — the PMOS turns off at that point, and stops contributing discharge current. NMOS transistors are hence the preferred devices in the PDN. Similarly, two alternative approaches to charging up a capacitor are shown in Figure 2.b, with the output initially at GND. A PMOS switch succeeds in charging the output all the way to VDD, while the NMOS device fails to raise the output above VDD-VTn. This explains why PMOS transistors are preferentially used in a PUN.

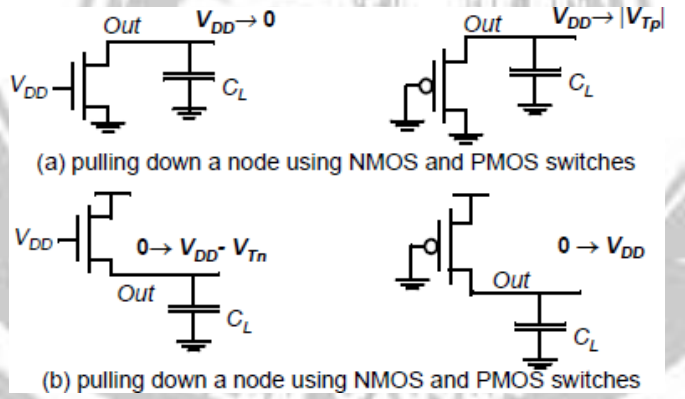

Figure 2.1.2: Simple examples illustrate why an NMOS should be used as a pull-down, and a PMOS should be used as a pull-up device.

[Source :Neil H.E. Weste, David Money Harris —CMOS VLSI Design: A Circuits and Systems Perspective…]

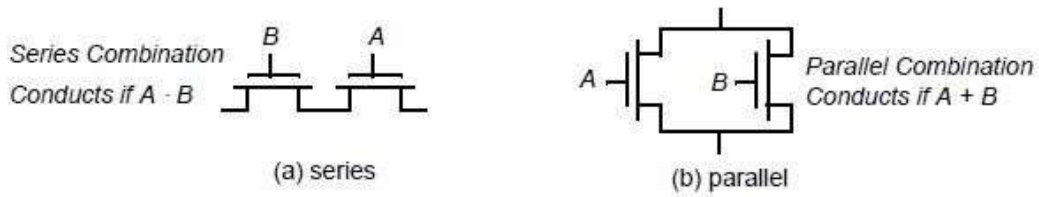

Figure 2.1.3: NMOS logic rules — series devices implement an AND, and parallel devices implement an OR.

[Source :Neil H.E. Weste, David Money Harris ―CMOS VLSI Design: A Circuits and Systems Perspective…]

A set of construction rules can be derived to construct logic functions (Figure 4). NMOS devices connected in series corresponds to an AND function. With all the inputs high, the series combination conducts and the value at one end of the chain is transferred to the other end. Similarly, NMOS transistors connected in parallel represent an OR function. A conducting path exists between the output and input terminal if at least one of the inputs is high. Using similar arguments, construction rules for PMOS networks can be formulated. A series connection of PMOS conducts if both inputs are low, representing a NOR function (A.B = A+B), while PMOS transistors in parallel implement a NAND (A+B = A· B.

- Using De Morgan's theorems ((A + B) = A· B and A· B = A + B), it can be shown that the pull-up and pull-down networks of a complementary CMOS structure are dual networks. This means that a parallel connection of transistors in the pull-up network corresponds to a series connection of the corresponding devices in the pull-down network, and vice versa. Therefore, to construct a CMOS gate, one of the networks (e.g., PDN) is implemented using combinations of series and parallel devices. The other network (i.e.,PUN) is obtained using duality principle by walking the hierarchy, replacing series sub- nets with parallel sub-nets, and parallel sub-nets with series sub-nets. The complete CMOS gate is constructed by combining the PDN with the PUN.

- The complementary gate is naturally inverting, implementing only functions such as NAND, NOR, and XNOR. The realization of a non-inverting Boolean function (such as AND OR, or XOR) in a single stage is not possible, and requires the addition of an extra inverter stage.
- The number of transistors required to implement an N-input logic gate is 2N.

ROHINI COLLEGE OF ENGINEERING AND TECHNOLOGY
PALKULAM, KANYAKUMARI
OBSERVE OPTIMIZE OUTSPREAD