**ASSESSMENT 24**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | 13-06-2020 | **Name:** | Sheela Golasangi |
| **Course:** | VLSI | **USN:** | 4AL16EC068 |
| **Topic:** | Digital VLSI design virtual lab | **Semester & Section:** | VIII  ‘B’ |
| **Github Repository:** | Sheela-Course |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| MOSFET The Metal Oxide Semiconductor Field Effect Transistor, or **MOSFET** for short, has an extremely high input gate resistance with the current flowing through the channel between the source and drain being controlled by the gate voltage. Because of this high input impedance and gain, MOSFETs can be easily damaged by static electricity if not carefully protected or handled.  **MOSFET’s** are ideal for use as electronic switches or as common-source amplifiers as their power consumption is very small. Typical applications for metal oxide semiconductor field effect transistors are in Microprocessors, Memories, Calculators and Logic CMOS Gates etc.  The metal–oxide–semiconductor field-effect transistor (MOSFET) is a transistor used for amplifying or switching electronic signals. In MOSFETs, a voltage on the oxide-insulated gate electrode can induce a conducting channel between the two other contacts called source and drain. The channel can be of n-type or p-type, and is accordingly called an nMOSFET or a pMOSFET. Figure 1 shows the schematic diagram of the structure of an nMOS device before and after channel formation.    Figure shows symbols commonly used for MOSFETs where the bulk terminal is either labeled (B) or implied (not drawn).    Fig. : Circuit symbols for nMOS and pMOS respectively Output characteristics MOSFET output characteristics plot ID versus VDS for several values of VGS.    The characteristics of an nMOS transistor can be explained as follows. As the voltage on the top electrode increases further, electrons are attracted to the surface. At a particular voltage level, which we will shortly define as the threshold voltage, the electron density at the surface exceeds the hole density. At this voltage, the surface has inverted from the p-type polarity of the original substrate to an n-type inversion layer, or inversion region, directly underneath the top plate as indicated in Fig. 1(b). This inversion region is an extremely shallow layer, existing as a charge sheet directly below the gate. In the MOS capacitor, the high density of electrons in the inversion layer is supplied by the electron–hole generation process within the depletion layer. The positive charge on the gate is balanced by the combination of negative charge in the inversion layer plus negative ionic acceptor charge in the depletion layer. The voltage at which the surface inversion layer just forms plays an extremely important role in field-effect transistors and is called the threshold voltage Vtn. The region of output characteristics where VGStn and no current flows is called the cutt-off region. When the channel forms in the nMOS (pMOS) transistor, a positive (negative) drain voltage with respect to the source creates a horizontal electric field moving the electrons (holes) toward the drain forming a positive (negative) drain current coming into the transistor. The positive current convention is used for electron and hole current, but in both cases electrons are the actual charge carriers. If the channel horizontal electric field is of the same order or smaller than the vertical thin oxide field, then the inversion channel remains almost uniform along the device length. This continuous carrier profile from drain to source puts the transistor in a bias state that is equivalently called either the non-saturated, linear, or ohmic bias state. The drain and source are effectively short-circuited. This happens when VGS > VDS + Vtn for nMOS transistor and VGS < VDS +Vtp for pMOS transistor. Drain current is linearly related to drain-source voltage over small intervals in the linear bias state.  But if the nMOS drain voltage increases beyond the limit, so that VGS < VDS + Vtn, then the horizontal electric field becomes stronger than the vertical field at the drain end, creating an asymmetry of the channel carrier inversion distribution shown in Figure 4.    Fig. : Channel pinchoff for (a) nMOS and (b) pMOS transistor devices.  If the drain voltage riseswhile the gate voltage remains the same, then VGD can go below the threshold voltage in the drain region. There can be no carrier inversion at the drain-gate oxide region, so the inverted portion of the channel retracts from the drain, and no longer “touches” this terminal. The pinched-off portion of the channel forms a depletion region with a high electric field. The n-drain and p-bulk form a pn junction. When this happens the inversion channel is said to be “pinched-off” and the device is in the saturation region. The characteristics can be loosely modelled by the following equations.    **Transfer characteristics**  The transfer characteristic relates drain current (ID) response to the input gate-source driving voltage (VGS). Since the gate terminal is electrically isolated from the remaining terminals (drain, source, and bulk), the gate current is essentially zero, so that gate current is not part of device characteristics. The transfer characteristic curve can locate the gate voltage at which the transistor passes current and leaves the OFF-state. This is the device threshold voltage (Vtn). Figure 5 shows measured input characteristics for an nMOS and pMOS transistor with a small 0.1V potential across their drain to source terminals.    The transistors are in their non-saturated bias states. As VGS increases for the nMOS transistor in Figure 5a, the threshold voltage is reached where drain current elevates. For VGS between 0V and 0.7V, ID is nearly zero indicating that the equivalent resistance between the drain and source terminals is extremely high. Once VGS reaches 0.7V, the current increases rapidly with VGS indicating that the equivalent resistance at the drain decreases with increasing gate-source voltage. Therefore, the threshold voltage of the given nMOS transistor is about Vtn ≈ 0.7V. The pMOS transistor input characteristic in Figure 5b is analogous to the nMOS transistor except the ID and VGS polarities are reversed.  **CMOS Inverter**  The inverter is universally accepted as the most basic logic gate doing a Boolean operation on a single input variable. Fig.1 depicts the symbol, truth table and a general structure of a CMOS inverter. As shown, the simple structure consists of a combination of anpMOS transistor at the top and a nMOS transistor at the bottom.    CMOS is also sometimes referred to as complementary-symmetry metal–oxide–semiconductor. The words "complementary-symmetry" refer to the fact that the typical digital design style with CMOS uses complementary and symmetrical pairs of p-type and n-type metal oxide semiconductor field effect transistors (MOSFETs) for logic functions. Two important characteristics of CMOS devices are high noise immunity and low static power consumption. Significant power is only drawn while the transistors in the CMOS device are switching between on and off states. Consequently, CMOS devices do not produce as much waste heat as other forms of logic, for example transistor-transistor logic (TTL) or NMOS logic, which uses all n-channel devices without p-channel devices.  **4:1 MUX**  A multiplexer or mux is a combinational circuits that selects several analog or digital input signals and forwards the selected input into a single output line. A multiplexer of 2n inputs has n selected lines, are used to select which input line to send to the output.    Figure shows how a 4:1 MUX can be constructed out of two 2:1 MUXs.    **Logic Gates**    **Static logic** is a design methodology in integrated circuit design where there is at all times some mechanism to drive the output either high or low. For example, in many of the popular logic families, such as TTL and traditional CMOS, there is always a low-impedance path between the output and either the supply voltage or the ground. The most widely used logic style is static CMOS. A static CMOS gate is a combination of two networks, called the pull-up network (PUN) and the pull-down network (PDN). 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 (based on the inputs). The PUN and PDN networks are constructed in a mutually exclusive fashion such that, one and only one of these networks is conducting in the steady state.    **Dynamic logic** is a design methodology in integrated circuit design in that it uses a clock signal in its implementation of combinational logic circuits. In dynamic logic, there is not always a mechanism driving the output high or low. In the most common version of this concept, the output is driven high or low during distinct parts of the clock cycle. Dynamic logic requires a minimum clock rate fast enough that the output state of each dynamic gate is used before it leaks out of the capacitance holding that state. The basic construction of a dynamic logic gate is shown in fig.2. The PDN (pull-down network) is constructed exactly as in complementary CMOS. The operation of this circuit is divided into two major phases: precharge and evaluation, with the mode of operation determined by the clock signal CLK.    Precharge: When CLK = 0, the output node Out is precharged to VDD by the PMOS transistor Mp. During that time, the evaluate NMOS transistor Me is off, so that the pull-down path is disabled. The evaluation FET eliminates any static power that would be consumed during the precharge period (this is, static current would flow between the supplies if both the pulldown and the precharge device were turned on simultaneously).  Evaluation: For CLK = 1, the precharge transistor Mp is off, and the evaluation transistor Me is turned on. The output is conditionally discharged based on the input values and the pull-down topology. If the inputs are such that the PDN conducts, then a low resistance path exists between Out and GND and the output is discharged to GND. If the PDN is turned off, the precharged value remains stored on the output capacitance CL, which is a combination of junction capacitances, the wiring capacitance, and the input capacitance of the fan-out gates. During the evaluation phase, the only possible path between the output node and a supply rail is to GND. Consequently, once Out is discharged, it cannot be charged again till then next precharge operation. The inputs to the gate can therefore make at most one transition during evaluation. |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | 13-06-2020 | **Name:** | Sheela Golasangi |
| **Course:** | Java Tutorial for Complete Beginners | **USN:** | 4AL16EC068 |
| **Topic:** | Programming core java   1. The Equals Method 2. Inner Classes 3. Enum Types: Basic and Advanced Usage 4. Recursion: A Useful Trick Up Your Sleeve 5. Serialization: Saving Objects to Files 6. Serializing Arrays 7. The Transient Keyword and More Serialization 8. Passing by Value | **Semester & Section:** | VIII  ‘B’ |
| **Github Repository:** | Sheela-Course |  |  |
| **AFTERNOON SESSION DETAILS** | | | |
| The method determines whether the Number object that invokes the method is equal to the object that is passed as an argument. Syntax public boolean equals(Object o) Parameters Here is the detail of parameters −   * Any object.  Return Value  * The method returns True if the argument is not null and is an object of the same type and with the same numeric value. There are some extra requirements for Double and Float objects that are described in the Java API documentation.  Example publicclassTest{  publicstaticvoid main(Stringargs[]){  Integer x =5;  Integer y =10;  Integer z =5;  Short a =5;  System.out.println(x.equals(y));  System.out.println(x.equals(z));  System.out.println(x.equals(a));  }  } Nested Classes In Java, just like methods, variables of a class too can have another class as its member. Writing a class within another is allowed in Java. The class written within is called the nested class, and the class that holds the inner class is called the outer class.  Syntax  Following is the syntax to write a nested class. Here, the class Outer\_Demo is the outer class and the class Inner\_Demo is the nested class.  class Outer\_Demo {  class Inner\_Demo {  }  }  Nested classes are divided into two types −   * Non-static nested classes − These are the non-static members of a class. * Static nested classes − These are the static members of a class.  Java Transient Keyword **Java transient** keyword is used in serialization. If you define any data member as transient, it will not be serialized.  Let's take an example, I have declared a class as Student, it has three data members id, name and age. If you serialize the object, all the values will be serialized but I don't want to serialize one value, e.g. age then we can declare the age data member as transient. Example of Java Transient Keyword In this example, we have created the two classes Student and PersistExample. The age data member of the Student class is declared as transient, its value will not be serialized. Object references are passed by value All object references in Java are passed by value. This means that a copy of the value will be passed to a method. But the trick is that passing a copy of the value also changes the real value of the object. To understand why, start with this example:  publicclassObjectReferenceExample{  publicstaticvoid main(String...doYourBest){  Simpsonsimpson=newSimpson();  transformIntoHomer(simpson);  System.out.println(simpson.name);  }  staticvoidtransformIntoHomer(Simpsonsimpson){  simpson.name ="Homer";  }  }  classSimpson{  String name;  }  What do you think the simpson.name will be after the transformIntoHomer method is executed?  In this case, it will be Homer! The reason is that Java object variables are simply references that point to real objects in the memory heap. Therefore, even though Java passes parameters to methods by value, if the variable points to an object reference, the real object will also be changed. | | | |