

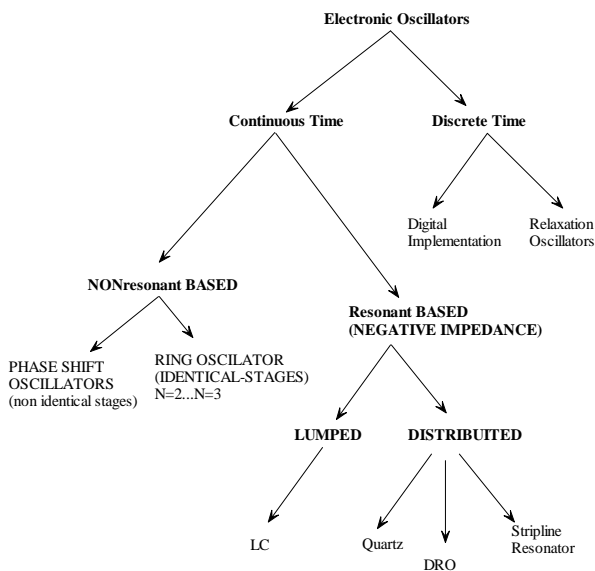
## Tutorial ADS2006- CMOS INVERTER OSCILLATORS

**Abstract**— This paper describes how to use ADS from Agilent to design an inverter oscillators as clock reference generator.

### I. INTRODUCTION

Oscillators are the key building of integrated transceivers. An oscillator classification can be based on one of the basic properties of the oscillator as frequency, tuning range, noise performance or functionality (single or multiphase outputs).

A possible classification of various oscillator are represented in fig.1 [1]



We begin this series of tutorial by considering as a first example the discrete time oscillators and by showing their implementation in ADS2006.

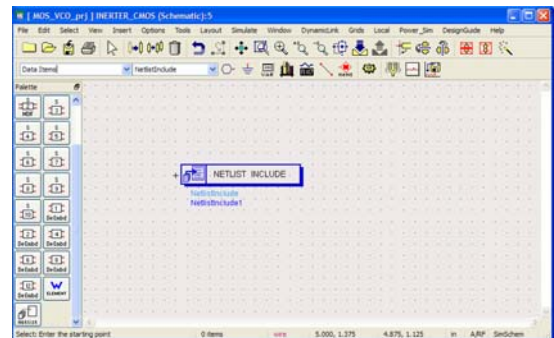
### II. CONTINUOUS TIME NON RESONANT BASED OSCILLATORS: AN EXAMPLE TO A CLOCK GENERATOR GENERAL

It is well know as the CMOS technology are advantageous for the design and implementation of integrated systems thanks to their relative low cost and the reduced dimensions that allows to integrate in small areas a great number of transistor. In this context oscillator can be employed to create a

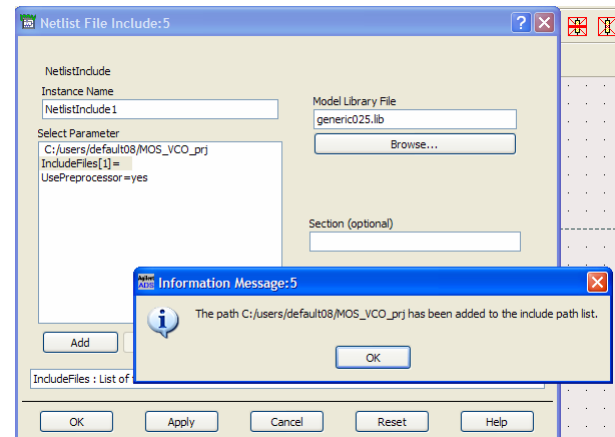
clock signal reference to synchronize all digital or mixed signal circuits.

The simples example of a CMOS clock generator can be achieved by using a odd number of CMOS cascade inverters. For the CAD systems there are a lot of design kit that contain mathematical models and layout check-rules to design a system. In ADS you can model transistors putting in general models the parameter for the technology. As an example the parameters for a generic 0.25 $\mu$ m technology can be downloaded here click on generic0.25.lib.

To include the file into design; open ADS, create a new schematic and save them as **INVERTER-CMOS**. By library **Data-Items** select **NETLIST INCLUDE** as show in fig2



Double click in **NETLIST INUDE** and click the voice **IncludeFiles[1]**; then select with Browse the directory which contain the file generic0.25.lib and click on ok.

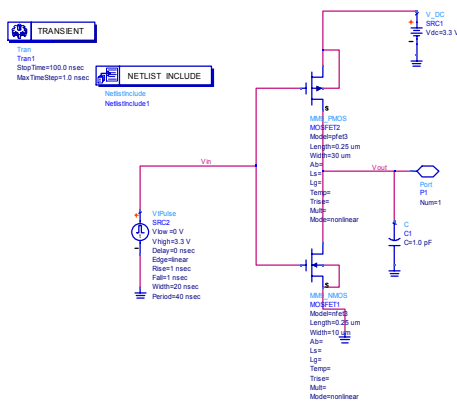


Now we can start to design and simulate a single inverter. From library **DEVICES-MOS** select **NMOS** and **PMOS** transistor and

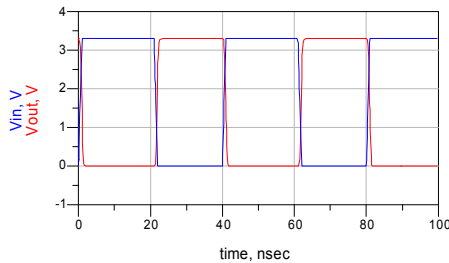
digit **nfet3** for NMOS and **pfet3** for PMOS. Choose dimension for transistors and connect the terminal with wire. In the same way add the DC-Bias from **Sources-Time Domain** library and the signal source.

From **Lumped Components** library you can select the capacitive load. **By Simulation-Transient** library select Transient and put into schematic also the names for the input and output nodes.

At the end the windows will be as



The simulation results are



The simplest ring oscillator contains an odd number of cascade inverters; the output of the last stage is connected to the input of first stage; in this way the the oscillation frequency depend by the delay time introduced by each inverter and is not externally controllable.

For a single CMOS inverter the delay time is computed after considering the charge and discharge time of the capacitor.

The rise time  $\tau_R$  can be expressed as

$$\tau_R = \frac{4C}{K_P \left(\frac{W}{L}\right)_P V_{DD}}$$

where  $C$  is the load capacitance  $K_P$  is the intrinsic conductance of the p-MOS and depend by  $\mu_n, C_{OX}$ .

The fall time has the same expression but for the N-MOS transistor

$$\tau_F = \frac{4C}{K_N \left(\frac{W}{L}\right)_N V_{DD}}$$

The frequency of oscillation is simply defined as

$$\begin{aligned} f_{osc} &= \frac{1}{N(\tau_R + \tau_F)} \rightarrow \\ &= \frac{1}{N \left( \frac{4C}{K_P \left(\frac{W}{L}\right)_P V_{DD}} + \frac{4C}{K_N \left(\frac{W}{L}\right)_N V_{DD}} \right)} \\ &= \frac{1}{N \left( \frac{4C \left( K_P \left(\frac{W}{L}\right)_P + K_N \left(\frac{W}{L}\right)_N \right)}{K_N K_P \left(\frac{W}{L}\right)_P \left(\frac{W}{L}\right)_N V_{DD}} \right)} \\ &= \frac{K_N K_P \left(\frac{W}{L}\right)_P \left(\frac{W}{L}\right)_N V_{DD}}{N \left( 4C \left( K_P \left(\frac{W}{L}\right)_P + K_N \left(\frac{W}{L}\right)_N \right) \right)} \end{aligned}$$

From the file .lib it is possible to estimate the parameters for transistors N and P

$$\mu_N \approx 112^{-6} \text{ and } \mu_P \approx 23^{-6}$$

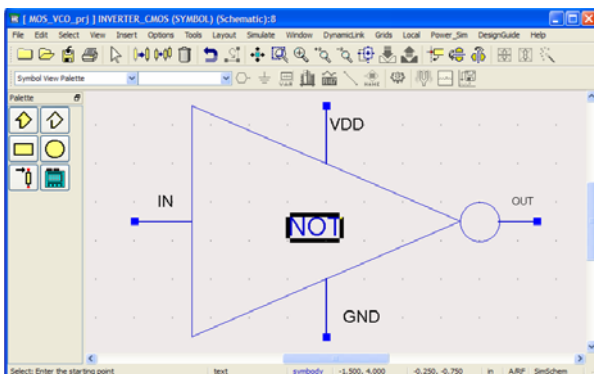
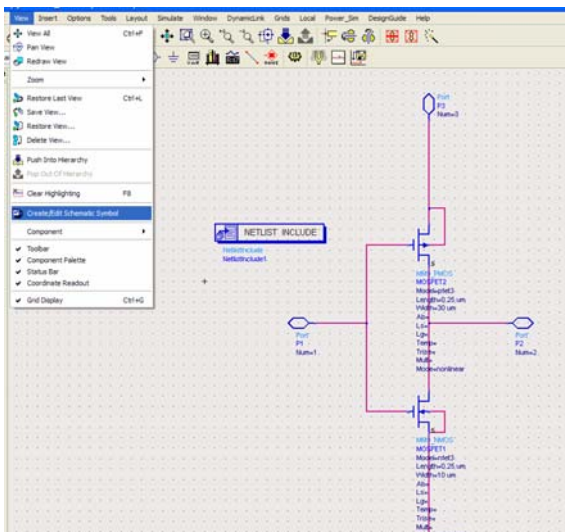
Then employing the previous relationship will be found, neglecting the parasite capacitances of transistors

$$f_{osc} = \frac{1}{N(\tau_R + \tau_F)} \approx 281 \text{ MHz}$$

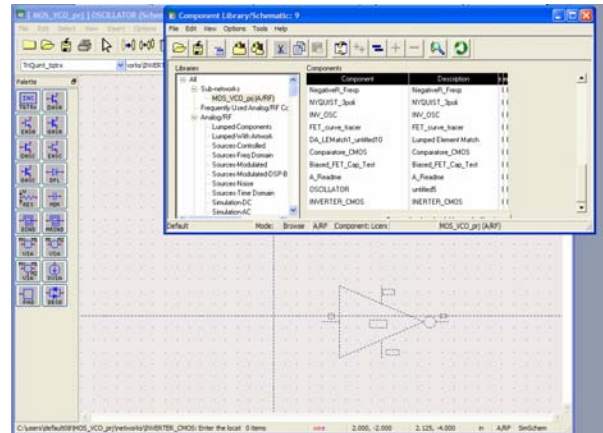
Results obtained with simulators (Transient and HB) obviously are better accurate.

### III. HIERARCHY DESIGN AND SIMULATIONS

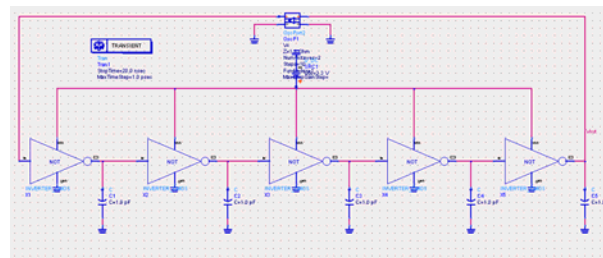
To keep clean the design it is better to create a symbol for the sub circuit and compose the full system instantiating the sub modules. To create the symbol you need first to define the ports for your sub circuit; in this case you can add 4 ports for Input, output, VDD and Ground. Then you can click inView and Create/Edit symbol; ADS create a default symbol that represent your network. At this point you can change the shape of symbol add names and save.



In a New schematic Windows drag the symbol that you created; you can find them setting the library of the project and setting the subcircuit

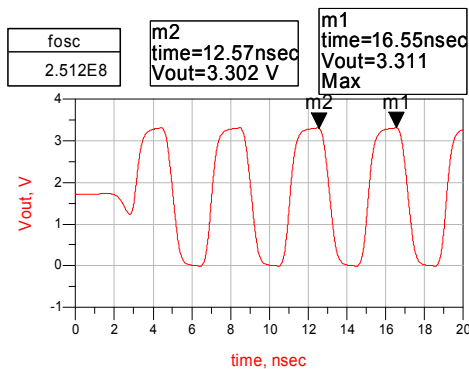


Then you can construct the full schematic for the oscillator

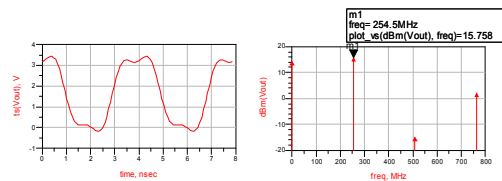


The first simulation is achieved through the Transient analysis. Transient analysis compute all the voltages nodes and currents for the circuit. The Osc Port2 taken from the Simulation-HB library it is necessary to introduce initial conditions different from zero and to see the start-up. The output can be see plotting the  $V_{out}$  versus Time and show how the waveform for the circuit is a square wave. The output frequency can be computet through the equation the equation editor by defining the Eqn fosc . The results show a oscillator frequency of **251MHz**.

$$\text{Eqn } f_{osc} = 1 / (\text{indep}(m1) - \text{indep}(m2))$$



Then run simulation and plot results in the window



#### IV. HOMEWORKS

Try to change W/L for the transistor and the number of stages to design a fixed frequency oscillator of: 150MHz ; 300MHz; 500MHz.

#### HB-Analysis.

A kind of frequency analysis most useful to study the behaviour of the system is based on Harmonic Balance analysis. The oscillator is a non linear system in the steady state condition but can be considered as a linear at the start-up ref[2] The harmonic Balance algorithm [3] is based on the idea to separate the parts of the systems that depend on amplitude to parts that depend uniquely on frequency. In this way it is possible to estimate with better precision the frequency and the output power of the waveform generated. To analyse the circuit with HB select the Harmonic Balance simulator by library Simulation-HB. Select and insert the Osc-Port device between input and output, then select the frequency of analysis (near to the target frequency) and activate the oscillator analysis using Osc-Port. For more detail you can see the Agilent website.

#### REFERENCES

- [1] Johan van der Thang, Dieter Kasperkovitz, Arthur van Roermud; "High Frequency Oscillator Design for Integrated Transceivers"; 2003 Kluwer Academic Publishers
- [2] M.Monni, G.Martines, "A Novel Approach to Determine the Start-Up Conditions in Microwave Negative Impedance Oscillators", European Microwave Conference, in EuMW2007, pp 1397-1400, Munich October 8-12, 2007.
- [3] Rowan Gilmore, Les Besser "Practical RF Circuit Design for Modern Wireless Systems Vol.2" Artech House, Boston 2003.

