Process and Device Simulation, Model Extraction and Circuit Simulation of Hybrid CMOS-like Inverters Using a-Si:H and Pentacene TFTs

Introduction

In this article we demonstrate the process and device simulation of a-Si:H TFT and Pentacene based OTFT and we subsequently use the resulting device simulation data to perform SPICE model extraction. As a test case of a hybrid circuit realization, a CMOS-like inverter and a ring oscillator are presented. Extracted SPICE models for NMOS a-Si:H TFT and pentacene PMOS TFT are used to simulate the test circuits. An amorphous silicon TFT is used in place of the NMOS devices and a Pentacene based TFT is used in place of the PMOS devices in the hybrid inverter circuit.

Traditionally logic circuits developed on TFTs primarily use one type of device; either an NMOS a-Si:H TFT or a PMOS organic device. The a-Si:H and pentacene PMOS TFTs have only recently been integrated into a CMOS-like structure.

First we tested the performance of a bootstrapped inverter using only NMOS a-Si:H TFT as in reference [1]. The ID-VG and ID-VD data for the NMOS a-Si:H TFT is taken from the TCAD simulation and is used in Utmost IV for SPICE model extraction.
Process Device Simulation and Model Extraction

Silvaco provides a complete solution of tools that support TCAD to Tape-out. Silvaco Athena and Atlas TCAD tools were used to model the TFT devices. The a-Si NTFT input TCAD deck used for process and device simulation, and for Utmost IV data set generation is given below:

```plaintext
go athena
# ATHENA TFT formation -> ATLAS id/Vd and Id/Vg extraction example

# mesh definition
line x loc=0 spac=0.2
line x loc=1.0 spac=0.1
line x loc=3.0 spac=0.25
line y loc=0 spac=0.1
line y loc=0.2 spac=0.1

# start with glass substrate
initialize oxide

# deposit thin metal as gate
deposit aluminum thick=0.3 spaces=2

# MIS insulator
deposit oxide thick=0.1 divisions=4
deposit nitride thick=0.1 divisions=4

# active amorphous silicon layer
deposit silicon thick=0.15 divisions=15 c.phos=2.1e14
deposit silicon thick=0.05 divisions=5 c.phosphor=1e20

# deposit source/drain contacts
deposit aluminum thick=0.2 divisions=2

# pattern source/drain regions. Recess into active layer.
deposit barrier thick=0.1 div=2
etch barrier p1.x=1.0 right
etch aluminum dry thick=0.20
etch silicon dry thick=0.10
etch barrier all

# Passivation Layer
deposit oxide thick=0.2 divisions=2

# mirror to get full structure
structure mirror right

# define electrodes for ATLAS
electrode name=source x=0 y=-1.0
electrode name=gate x=3
electrode name=drain x=6 y=-1.0

structure outf=n_tft.str
*tonyplot n_tft.str

go atlas simflags="-P 1"

mesh infile=n_tft.str width=20

material material=silicon mun=20 mup=1.5 nc300=2.5e20 \nv300=2.5e20 eg300=1.8 taun0=1e-8 taup0=1e-8
```
defects nta=1.e21 ntd=1.e21 wta=0.05 wtd=0.05 \\ nga=1.0e16 ngd=1.0e16 ega=0.6 egd=0.6 wga=0.3 wgd=0.3 \\ sigtae=1.e-17 sigtah=1.e-15 sigtde=1.e-15 sigtdh=1.e-17 \\ siggae=2.e-16 siggah=2.e-15 siggde=2.e-15 siggdh=2.e-16

interface qf=3e10
contact name=gate aluminum
models srh fermi temp=300 bbt.std bb.b=1.3e7
method climit=1e-4 itlimit=100 maxtraps=10

# ID vs VD Curves

# set gate biases with Vds=0.0
solve init
solve vgate=1 outf=solve_tmp1
solve vgate=3 outf=solve_tmp2
solve vgate=5 outf=solve_tmp3
solve vgate=7 outf=solve_tmp4
solve vgate=9 outf=solve_tmp5
solve vgate=11 outf=solve_tmp6
solve vgate=13 outf=solve_tmp7

log outf = idvd_tcad.log

# load in temporary files and ramp VDs
load infile=solve_tmp1
solve name=drain vdrain=0 vfinal=2 vstep=0.1
solve name=drain vdrain=3 vfinal=15 vstep=1

load infile=solve_tmp2
solve name=drain vdrain=0 vfinal=2 vstep=0.1
solve name=drain vdrain=3 vfinal=15 vstep=1

load infile=solve_tmp3
solve name=drain vdrain=0 vfinal=2 vstep=0.1
solve name=drain vdrain=3 vfinal=15 vstep=1

load infile=solve_tmp4
solve name=drain vdrain=0 vfinal=2 vstep=0.1
solve name=drain vdrain=3 vfinal=15 vstep=1

load infile=solve_tmp5
solve name=drain vdrain=0 vfinal=2 vstep=0.1
solve name=drain vdrain=3 vfinal=15 vstep=1

load infile=solve_tmp6
solve name=drain vdrain=0 vfinal=2 vstep=0.1
solve name=drain vdrain=3 vfinal=15 vstep=1

load infile=solve_tmp7
solve name=drain vdrain=0 vfinal=2 vstep=0.1
solve name=drain vdrain=3 vfinal=15 vstep=1

log off
tonypplot idvd_tcad.log

# ID vs VG Curves
# set gate biases with Vgs=0.0
solve init
solve vdrain=0.1 outf=solve_tmp1
solve vdrain=10  outf=solve_tmp12

log outf = idvg_tcad.log

# load in temporary files and ramp Vds
load infile=solve_tmp1
solve name=gate vgate=0 vfinal=-20 vstep=-0.5
solve name=gate vgate=-20 vfinal=20 vstep=0.5

load infile=solve_tmp12
solve name=gate vgate=0 vfinal=-20 vstep=-0.5
solve name=gate vgate=-20 vfinal=20 vstep=0.5

log off
tonyplot idvg_tcad.log

go utmost4 simflags="-V 1.8.6.R"

script begin

var mydir, allDataSets, dataSets, dataSet1, dataSet2, width, project, modelLibrary, netlist, optimSetup, error

mydir = Utmost4.openCurrentWorkingDirectory();

allDataSets = []; width = 20e-6;

dataSet1 = mydir.dcImportFromTCADLogfile("idvd_tcad.log");
dataSet1.setKeys (27, "tcad", "tcad", "tcad", "ntft_W20", "idvd");
dataSet1.addPlot ("idvd", "XY(LIN LIN)", "vdrain", "idrain", "");
dataSet1.addAttribute ("W", width);
allDataSets.push (dataSet1);


dataSet2 = mydir.dcImportFromTCADLogfile("idvg_tcad.log");
dataSet2.setKeys (27, "tcad", "tcad", "tcad", "ntft_W20", "idvg");
dataSet2.addPlot ("idvg", "XY(LIN LOG)", "vgate", "idrain", "");
dataSet2.addAttribute ("W", width);
allDataSets.push (dataSet2);

mydir.saveDataSets ("datasets_ntft_W20.uds", allDataSets);

script end
run script
quit
After completion of this deck the ID-VD and ID-VG dataset is converted in Utmost IV data format and the “datasets_ntft_W20.uds” file is generated. To extract the a-Si NTFT model, the output file (datasets_ntft_W20.uds) is imported into the Utmost IV database. In a similar way other TFT datasets can be generated and imported into Utmost IV. An additional script is necessary to convert the data to an Utmost IV format. The required TCAD input deck is shown below:

```
go utmost4 simflags="-V 1.8.6.R"
script begin
var mydir, allDataSets, dataSet1, dataSet2, width, project, modellibrary, netlist, optimSetup, error
mydir = Utmost4.openCurrentWorkingDirectory ();
allDataSets = [];
width = 20e-6;
dataSet1 = mydir.dcImportFromTCADLogfile ("idvd_tcad.log");
dataSet1.setKeys (27, "tcad", "tcad", "tcad", "ntft_W20", "idvd");
dataSet1.addPlot ("idvd", "XY(LIN LIN)", "vdrain", "idrain", ");
dataSet1.addAttribute ("W", width);
allDataSets.push (dataSet1);
dataSet2 = mydir.dcImportFromTCADLogfile ("idvg_tcad.log");
dataSet2.setKeys (27, "tcad", "tcad", "tcad", "ntft_W20", "idvg");
dataSet2.addPlot ("idvg", "XY(LIN LOG)", "vgate", "idrain", ");
dataSet2.addAttribute ("W", width);
allDataSets.push (dataSet2);
mydir.saveDataSets ("datasets_ntft_W20.uds", allDataSets);
script end
run script
```

In the above script, the following line converts the ID-VD data into the Utmost IV format:

```
dataSet1 = mydir.dcImportFromTCADLogfile ("idvd_tcad.log");
```

The following line converts the ID-VG data into the Utmost IV format:

```
dataSet2 = mydir.dcImportFromTCADLogfile ("idvg_tcad.log");
```

Finally all of the Utmost IV data is stored as a collection into a single file using the following line:

```
mydir.saveDataSets ("datasets_ntft_W20.uds", allDataSets);
```

This last step performs an automatic conversion of the TCAD datasets into an Utmost IV dataset. This could also be use for other TFT devices or any other TCAD simulated device when exporting to an Utmost IV database format. The detailed explanation of the a-Si TFT model extraction methodology can be found in the Silvaco application note “Utmost-IV Delivers Full Capability of RPI TFT Models” [2].
The ID-VD characteristics of a-Si NTFT after optimization in Utmost IV is shown in Figure 1 below:

![Figure 1. ID-VD characteristics of a-Si NTFT.](image)

**Circuit Simulation Using the Extracted SPICE Model**

The a-Si NTFT model was exported and used for circuit simulation of an NMOS a-Si:H TFT bootstrapped inverter and a five-stage ring oscillator, shown in figure 2 and 3 respectively. The NTFT and PTFT symbols were created in Gateway. Circuit simulation for the five-stage ring oscillator using an NMOS a-Si:H TFT bootstrapped inverter was performed in SmartSpice using the extracted NMOS a-Si:H TFT model.

![Figure 2. NMOS a-Si:H TFT bootstrapped inverter.](image)
At a 20V supply voltage, the output waveform for the five stage ring oscillator using the NMOS a-Si:H TFT bootstrapped inverter is shown in Figure 4. The resulting output waveform is similar to the waveform found in reference [1].
Using A Hybrid Inverter

TFT circuits in all NMOS (or PMOS) like topology have a large static power dissipation due to the existence of a direct path from supply to ground. Such power dissipation would prevent these circuits from being used in battery operated portable systems. Thus, the obvious choice is to integrate the a-Si:H NMOS TFT with pentacene PMOS TFT in a complete CMOS structure. This was first shown by Bonse et al [3] and CMOS TFT Op-Amps in hybrid TFT technology have been demonstrated in reference [4]. To realize the hybrid inverter circuit we performed a process and device simulation of pentacene based PTFT and an NMOS a-Si:H TFT. We converted the TCAD data to Utmost IV format (from .log file to .uds file), and performed model extraction in Utmost IV. For the pentacene based PTFT, a UOTFT model (level=37) was used. For the NMOS a-Si:H TFT an RPI a-Si TFT model (level=35), also available in Utmost IV, was used. Figure 5 illustrates the ID-VD characteristics of the OTFT after optimization in Utmost IV.

The extracted SPICE models of the a-Si NTFT and Pentacene based PTFT are then used in the hybrid inverter circuit and the five stage ring oscillator using this same hybrid inverter.

The schematics for the hybrid inverter and ring oscillator circuits are shown in Figure 6 and Figure 7, respectively. The output waveform of the five stage ring oscillator using the hybrid inverter is shown in Figure 8.

![Figure 5. ID-VD characteristics of Pentacene based TFT.](image-url)
Figure 6. Hybrid inverter schematic.

Figure 7. Five-stage ring oscillator using hybrid inverter.
Using a-Si Only Approach for CMOS-like Inverters

Figure 9 shows the schematic of a CMOS-like inverter using a-Si NTFT and a-Si PTFT model. Figure 10 illustrates the five stage ring oscillator circuit using a CMOS-like inverter in only a-Si technology. The output waveform of this ring oscillator is shown in Figure 11.

Figure 8. Output of five stage ring oscillator using hybrid inverter.

Figure 9. a-Si TFT based CMOS-like inverter schematic.
Conclusion

TCAD simulation of a-Si TFT and pentacene based TFT was performed and the output data was then converted to Utmost IV format. Using this data, SPICE models for a-Si TFT (level=35) and organic TFT (using UOTFT model level=37) were extracted and used successfully to simulate a five stage ring oscillator using an NMOS a-Si:H TFT bootstrapped inverter. Similarly, a five stage ring oscillator using a hybrid inverter was also simulated successfully. Using the Silvaco tool set we successfully demonstrated that process device simulation tools can be used for SPICE model extraction, which can in turn then be used for circuit simulation and analysis.
Author’s note
It is possible to perform TCAD mixed-mode simulation using Silvaco’s Atlas software, but for larger circuits, Gateway and SmartSpice with the extracted SPICE model of the TCAD device would be a more suitable approach.

References
[2] Umost IV Delivers Full Capability of RPI TFT Models (http://www.silvaco.com/content/appNotes/analog/1-026_UTMOST-IV_RPI.pdf )