Pratik Rajput, MTS Design Enablement Engineer
GLOBALFOUNDRIES
GlobalFoundries 22FDX® body bias design exploration with Genus for optimal power/performance tradeoff
April 11-12, 2017
Agenda

• GLOBALFOUNDRIES Overview
• What is 22FDX® Technology?
• Body-Biasing: A New Dimension in Design Closure
• Automation of Design Planning
• Analyze Library Assignment Flow Details
• Results
• Conclusion
The GLOBALFOUNDRIES Story

Building an industry leader

- Acquires Chartered Semiconductor

- Fab 8 delivers initial silicon 1st time right
- Dresden ships 250,000th 32nm HKMG wafer

- Launches 14nm FinFET technology
- Collaborates w/ Samsung on 14nm manufacturing

- GF created 2012
- Announces 7nm FinFET & 12FDX™
- Launches: SiGe 8XP & 5PAx RF technologies
- 22FDX®-MRAM technology
- FDXcelerator™ partner program

- Acquires IBM Microelectronics business
- Launches: 22FDX Platform FX-14™ ASIC

- Largest privately held semiconductor company
- Invests in Worldwide Capacity Growth to Meet Customer Demand

- ~8X capacity increase since 2009
Manufacturing Capacity

- Burlington, Vermont
- Singapore
- East Fishkill, New York
- Dresden, Germany
- Malta, New York

TECHNOLOGY NODES
- 350–90nm
- 180–40nm
- 90–22nm
- 28, 22, 12nm
- 14, 7nm

CAPACITY (WAFERS/MONTH)
- 40k (200mm)
- 68k (300mm) 93k (200mm)
- Up to 20k (300mm)
- Up to 80k (300mm)
- Up to 60k (300mm)

Up to 7.7M Wafers/Year
200mm equivalents
GLOBALFOUNDRIES Roadmap

Markets: Servers, high performance computing and graphics, high-end smartphone, core networking

High Performance Computing
- 7nm FinFET
- 14nm FinFET

Wireless, Battery-Powered Computing
- 12FDX™
- 22FDX®
- 28nm
- 40/55nm

Premium Tier
- Features: High-performance, high-cost

Markets: Low & mid-end smartphones, wireless, IoT, autonomous vehicles, mobile camera

Volume Tier
- Features: Low-power, cost-effective-performance, RF, embedded memory
22FDX® is the First Technology to demonstrate 0.4v operation capability at >500Mhz on an ARM A7 Processor.

Total Power (mw)

Freq. (MHz)

22FDX@0.4v

92% Less Power

22FDX

47% Less Power

14nm FinFET

50% Faster + 18% Less Power

28SLP-HKMG

Source: Verisilicon

22FDX®, ARM® Cortex-A7 Implementation
FDX™: The simple solution for advanced node performance

Next node performance with 40 percent fewer masks

- **Performance**
  - 22FDX
  - 16/14nm FinFET
  - 12FDX
  - 10nm FinFET

- **Relative # of Masks**
  - 22FDX
  - 16/14nm FinFET
  - 12FDX
  - 10nm FinFET

**FinFET performance & power**
- Superior Analog/RF
- Performance on demand

**Source:** Based on GF internal assessments

- **40% fewer masks**
- Lower mask cost
- Reduced cycle time
Example: Remote Security Camera Application

22FDX Delivers:
- Optimization for Max performance and Minimal power
  - FBB for max performance
  - RBB for minimal power (low leakage and dynamic power)
- RF integration for reduced BOM cost
Agenda

• GLOBALFOUNDRIES Overview
• What is 22FDX Technology?
• Body-Biasing: A New Dimension in Design Closure
• Automation of Design Planning
• Analyze Library Assignment Flow Details
• Results
• Conclusion
22FDX Technology

What is 22FDX technology?
• It is the new 22nm Fully Depleted Silicon-on-Insulator (FDSOI) technology from GLOBALFOUNDRIES

Advantages:
• Lower Leakage due to insulator layer
• Enables Body Bias (BB) with minimal leakage impact
• FDSOI variability is smaller across die due to lower doping effort

Effects of Body Biasing in Bulk Transistor and FDSOI Transistor

Bulk versus FDSOI
RBB versus FBB

• Bias voltage is applied to P-well and N-well

• Reverse Body Bias (RBB)
  • nMOS neg. substrate voltage, pMOS pos. substrate voltage
  • raising VT of these devices

• Forward Body Bias (FBB)
  • nMOS pos. substrate voltage, pMOS neg. substrate voltage
  • lowering VT of these devices
Agenda

• GLOBALFOUNDRIES Overview
• What is 22FDX Technology?
• Body-Biasing: A New Dimension in Design Closure
• Automation of Design Planning
• Analyze Library Assignment Flow Details
• Results
• Conclusion
22FDX Technology

Power/Performance Trade-off

- **Leakage Power** vs. **Max Frequency**
  - **Reverse Body-bias (RBB)**
  - **Forward Body-bias (FBB)**

- **FBB and RBB are different devices**

**FDSOI: Fully Depleted Silicon-on-Insulator**

- *-2V to +2V Body-Biasing*

**SLVT/LVT**
- Lowest VT
- Optimized for FBB
- Highest Performance

**RVT/HVT**
- Mid-Range VT
- Optimized for RBB
- Balance of low leakage and high performance

**Minimum Leakage in Standby Mode**

**Maximum Performance Operating Mode**

**Relative Fmax**
22FDX Body-Biasing

A New Dimension in Design Closure

• Body-Biasing offers an additional option to tune cell performance or power:
  • Same implementation can be timed with different Bias voltages resulting in different performance results
  • Different Body-Biasing domains on one chip are enabling new design architectures and design styles

• PVT + BIAS $\rightarrow$ PVTB
22FDX Body-Biasing

Multi-Bias Domain - Example

- The RTL can be dissected into modules for better BB optimization:
  - Optimize modules rather than the whole design with module-specific BB
  - Does NOT require level shifters or isolation cells (same VDD) – Just spacing rule
  - Proper design planning along with Body Biasing will give optimal PPA

<table>
<thead>
<tr>
<th>Module</th>
<th>VNW bias</th>
<th>VPW bias</th>
<th>VDD</th>
</tr>
</thead>
<tbody>
<tr>
<td>OR1200_TOP</td>
<td>0V</td>
<td>0V</td>
<td>VDD(+/-10%)</td>
</tr>
<tr>
<td>OR1200_CPU</td>
<td>0V</td>
<td>-1V</td>
<td>VDD(+/-10%)</td>
</tr>
<tr>
<td>OR1200_DU</td>
<td>1V</td>
<td>-2V</td>
<td>VDD(+/-10%)</td>
</tr>
</tbody>
</table>
22FDX Body-Biasing

Multi-Bias Domain - Floorplan

- **NET_BIAS0_VPW**
- **NET_BIAS0_VNW**
- **NET_BIAS1_VPW**
- **NET_BIAS1_VNW**
- **NET_BIAS2_VPW**
- **NET_BIAS2_VNW**

Common VDD and VSS for all 3 BIAS domains

S = Spacing between BIAS areas
Agenda

• GLOBALFOUNDRIES Overview
• What is 22FDX Technology?
• Body-Biasing: A New Dimension in Design Closure
• Automation of Design Planning
• Analyze Library Assignment Flow Details
• Results
• Conclusion
Hierarchical blocks of SoC designs with varying range of operating frequencies would benefit from optimal choice of BB and library modes (RBB/FBB)

An automated solution for BB voltages and library mode selection will provide best possible PPA while reducing the design turn around time

analyze_library_assignment provides a cockpit solution to quickly make choice on critical design parameters like Power, Performance and Area
Agenda

• GLOBALFOUNDRIES Overview
• What is 22FDX Technology?
• Body-Biasing: A New Dimension in Design Closure
• Automation of Design Planning
• Analyze Library Assignment Flow Details
• Results
• Conclusion
Automatic Bias Selection Flow

- Read libraries with all bias voltages
- each bias voltage is represented by a separate library_domain
- Read HDL & Apply constraints
- Set analyze_library_order attribute
- Preview which hierarchy instances to analyze and decide on selection criteria
- Synthesize targeting first library_domain
- Store Power Performance Area (PPA) for each hierarchy instance
- Iterate through each library domain in analyze_library_order attribute
- When finished, the tool will select a bias voltage for each hierarchical instance based up on selection criteria
- The utility has the option to reselect new bias voltages if the selection criteria changes.
22FDX : Analyze Library Assignment Flow

• Analyze_library_assignment usage

Usage: analyze_library_assignment -outdir <string> [-preview] [-threshold <integer>] [-default_selection <string>]
[-tradeoff_range_percentage <integer>] [-slack <integer>] [-express] [-reselect] [<string>]

-outdir <string>:
  CSV results files written to this directory

-[preview]:
  Preview mode. Do sanity checking and report what analysis would be done, but stop before doing the analysis.

-[threshold <integer>]:
  Generic area threshold required for analysis. Only instances over this generic area threshold will be analyzed.

-[default_selection <string>]:
  Default selection criteria if analyze_library_selection_criteria attribute is not set on an instance.
  Possible values are best_area, best_total_power, best_leakage_power, best_dynamic_power,
  area_tradeoff_leakage, area_tradeoff_power, leakage_tradeoff_area, power_tradeoff_area. If unspecified, best_area is used.

-[tradeoff_range_percentage <integer>]:
  The percentage increase in area or power that is allowed. Allowed power is least power + tradeoff_range_percentage.
  Default if unspecified is 15%.

-[slack <integer>]:
  Slack number (in ps) used as passing criteria. Default is 0 ps.

-[express]:
  Express synthesis

-[reselect]:
  Read the CSV files from outdir, and reselect the library domain for each instance based upon the new criteria.

-<string>:
  The design or hier-instance under which all hier instances will be analyzed. If this option is not supplied, only the top design will be analyzed.
22FDX : Analyze Library Assignment Flow

• **Analyze_library_assignment Associated Attributes**

  • Attribute to specify the library_domains that will targeted during the analysis. The first library domain the list will targeted first, then the second, etc.

    ```
    attribute name: analyze_library_order
    category: enhancement ()
    object type: root
    access type: read-write
    data type: string
    default value:
    help: List of library sets to target during the analysis, in the order they will be run.
    ```

  • Attribute to specify the selection criteria for each hierarchical instance that is analyzed. If this attribute is not defined on an instance, the default selection criteria specified on the command line will be used.

    ```
    attribute name: analyze_library_selection_criteria
    category: enhancement ()
    object type: instance
    access type: read-write
    data type: string
    default value:
    help: Selection criteria to be used for this particular instance. Possible values are best_area, best_total_power, best_leakage_power, best_dynamic_power, area_tradeoff_leakage, area_tradeoff_power, leakage_tradeoff_area and power_tradeoff_area.
    ```
Bias Libraries Setup Script Example

```bash
set libs(dom_0P00V_0P00V_rbb) {
    GFVAR_LIB_NAME(hvt)_SSG_0P72V_0P00V_0P00V_0P00V_M40C.lib
    GFVAR_LIB_NAME(rvt)_SSG_0P72V_0P00V_0P00V_0P00V_M40C.lib 
}
set libs(dom_0P00V_0P00V_fbb) {
    GFVAR_LIB_NAME(lvt)_SSG_0P72V_0P00V_0P00V_0P00V_M40C.lib
    GFVAR_LIB_NAME(slvt)_SSG_0P72V_0P00V_0P00V_0P00V_M40C.lib 
}
set libs(dom_0P00V_M1P00V_rbb) {
    GFVAR_LIB_NAME(hvt)_SSG_0P72V_0P00V_0P00V_M1P00V_M40C.lib
    GFVAR_LIB_NAME(rvt)_SSG_0P72V_0P00V_0P00V_M1P00V_M40C.lib 
}
set libs(dom_0P00V_M1P00V_fbb) {
    GFVAR_LIB_NAME(lvt)_SSG_0P72V_0P00V_0P00V_M1P00V_M40C.lib
    GFVAR_LIB_NAME(slvt)_SSG_0P72V_0P00V_0P00V_M1P00V_M40C.lib 
}
set libs(dom_1P00V_M2P00V_rbb) {
    GFVAR_LIB_NAME(hvt)_SSG_0P72V_0P00V_1P00V_M2P00V_M40C.lib
    GFVAR_LIB_NAME(rvt)_SSG_0P72V_0P00V_1P00V_M2P00V_M40C.lib 
}
set libs(dom_1P00V_M2P00V_fbb) {
    GFVAR_LIB_NAME(lvt)_SSG_0P72V_0P00V_1P00V_M2P00V_M40C.lib
    GFVAR_LIB_NAME(slvt)_SSG_0P72V_0P00V_1P00V_M2P00V_M40C.lib 
}
foreach i {dom_0P00V_0P00V_rbb dom_0P00V_0P00V_fbb dom_0P00V_M1P00V_rbb dom_0P00V_M1P00V_fbb
dom_1P00V_M2P00V_rbb dom_1P00V_M2P00V_fbb} {
    create_library_domain $i
    set_attr library $libs($i) [find /-library_domain $i] } 

VDD    VSS    VNW    VPW    Temp
0p72V  0p00V  0p00V  M1p00V  M40c
```
Bias Power Libraries Setup Script Example

# reading power libraries is an important step to ensure power analysis is done at the appropriate PVT corner.

```
set libs(dom_0P00V_0P00V_rbb__power) {
    GFVAR_LIB_NAME(hvt)_FFG_0P88V_0P00V_0P00V_0P00V_125C.lib
    GFVAR_LIB_NAME(rvt)_FFG_0P88V_0P00V_0P00V_0P00V_125C.lib
}

set libs(dom_0P00V_0P00V_fbb__power) {
    GFVAR_LIB_NAME(lvt)_FFG_0P88V_0P00V_0P00V_0P00V_125C.lib
    GFVAR_LIB_NAME(slvt)_FFG_0P88V_0P00V_0P00V_0P00V_125C.lib
}

set libs(dom_0P00V_M1P00V_rbb__power) {
    GFVAR_LIB_NAME(hvt)_FFG_0P88V_0P00V_0P00V_M1P00V_125C.lib
    GFVAR_LIB_NAME(rvt)_FFG_0P88V_0P00V_0P00V_M1P00V_125C.lib
}

set libs(dom_0P00V_M1P00V_fbb__power) {
    GFVAR_LIB_NAME(lvt)_FFG_0P88V_0P00V_0P00V_M1P00V_125C.lib
    GFVAR_LIB_NAME(slvt)_FFG_0P88V_0P00V_0P00V_M1P00V_125C.lib
}

set libs(dom_1P00V_M2P00V_rbb__power) {
    GFVAR_LIB_NAME(hvt)_FFG_0P88V_1P00V_0P00V_M2P00V_125C.lib
    GFVAR_LIB_NAME(rvt)_FFG_0P88V_1P00V_0P00V_M2P00V_125C.lib
}

set libs(dom_1P00V_M2P00V_fbb__power) {
    GFVAR_LIB_NAME(lvt)_FFG_0P88V_1P00V_0P00V_M2P00V_125C.lib
    GFVAR_LIB_NAME(slvt)_FFG_0P88V_1P00V_0P00V_M2P00V_125C.lib
}

foreach i {dom_0P00V_0P00V_rbb dom_0P00V_0P00V_fbb dom_0P00V_M1P00V_rbb dom_0P00V_M1P00V_fbb
    dom_1P00V_M2P00V_rbb dom_1P00V_M2P00V_fbb} {
    # create power library domain
    create_library_domain ${i}_power
    set_attr library $libs(${i}_power) [find / -library_domain ${i}_power]
    set_attr power_library $libs(${i}_power) [find / -library_domain $i]
}
```
• Genus Synthesis Tool Run Script Example

```tcl
source library_setup.tcl
read_hdl <hdl_files>
elaborate
check_design -unresolved
# apply timing constraints
read_sdc <sdc_file>
define_cost_group -name I2C -design $DESIGN
define_cost_group -name C2O -design $DESIGN
define_cost_group -name I2O -design $DESIGN
path_group -from [all::all_seqs] -to [all::all_outs] -group C2O -name C2O
path_group -from [all::all_inps] -to [all::all_seqs] -group I2C -name I2C
path_group -from [all::all_inps] -to [all::all_outs] -group I2O -name I2O
# source the analyze_library_assignment procedure
source analyze_library_assignment.etf
# define the library order
set_attr analyze_library_order {}
  dom_0P00V_0P00V_rbb dom_0P00V_0P00V_fbb 
  dom_0P00V_M1P00V_rbb dom_0P00V_M1P00V_fbb 
  dom_1P00V_M2P00V_rbb dom_1P00V_M2P00V_rbb 
}  
# optionally define a selection criteria on specific instances
set_attribute analyze_library_selection_criteria best_leakage_power [find / -instance or1200_cpu]
# run analyze_library_assignment. Use -preview to see what instances will be analyzed before running.
analyze_library_assignment -threshold 300 -slack -10 
  -default_sel best_total_power 
  -outdir myodir.genus [find_unique_design]
```
Agenda

• GLOBALFOUNDRIES Overview
• What is 22FDX Technology?
• Body-Biasing: A New Dimension in Design Closure
• Automation of Design Planning
• Analyze Library Assignment Flow Details
• Results
• Conclusion
**22FDX : Analyze Library Assignment Results**

- Table written for each instance analyzed showing the PPA achieved for each library set targeted.

<table>
<thead>
<tr>
<th>or1200_cpu</th>
<th>WNS</th>
<th>Area</th>
<th>Leakage (nW)</th>
<th>Dynamic (nW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>dom_0P00V_0P00V_rbb</td>
<td>0.0</td>
<td>17443.9</td>
<td>8827431.3</td>
<td>27912797.5</td>
</tr>
<tr>
<td>dom_0P00V_0P00V_fbb</td>
<td>0.0</td>
<td>14543.3</td>
<td>22365217.9</td>
<td>27671021.7</td>
</tr>
<tr>
<td>dom_0P00V_M1P00V_rbb</td>
<td>-104.7</td>
<td>20787.5</td>
<td>7524229.3</td>
<td>32266200.8</td>
</tr>
<tr>
<td>dom_0P00V_M1P00V_fbb</td>
<td>0.1</td>
<td>14522.9</td>
<td>48642126.9</td>
<td>32573354.0</td>
</tr>
<tr>
<td>dom_1P00V_M2P00V_rbb</td>
<td>-676.3</td>
<td>27454.5</td>
<td>3872347.3</td>
<td>36007518.9</td>
</tr>
<tr>
<td>dom_1P00V_M2P00V_fbb</td>
<td>0.0</td>
<td>14093.3</td>
<td>128142848.7</td>
<td>46680329.9</td>
</tr>
</tbody>
</table>

- A table summarizing the library selection is written last.

**NOTE:** Library selection results shown in the following table:

<table>
<thead>
<tr>
<th>Instance</th>
<th>Selection Criteria</th>
<th>Library Domain Selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>or1200_cpu</td>
<td>best_total_power</td>
<td>dom_0P00V_0P00V_rbb</td>
</tr>
<tr>
<td>or1200_du</td>
<td>best_total_power</td>
<td>dom_0P00V_M1P00V_rbb</td>
</tr>
<tr>
<td>or1200_tt</td>
<td>best_total_power</td>
<td>dom_0P00V_M1P00V_rbb</td>
</tr>
</tbody>
</table>

**Diagram:**

- Total Power (nW) ranging from 0 to 200
- Leakage and Dynamic power distribution indicated by shading.
• How to Reselect with New Selection Criteria

The – reselect switch will recalculate based upon new selection criteria.
A design does not need to be loaded because the data will be extracted from the CSV files written to
the output directory.

```bash
legacy_genus:// analyze_library_assignment -slack -10 -default_sel best_leakage_power -outdir myodir.genus -reselect
```

<table>
<thead>
<tr>
<th>Design</th>
<th>Selection Criteria</th>
<th>Library Domain Selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>or1200_cpu</td>
<td>best_leakage_power</td>
<td>dom_0P00V_0P00V_rbb</td>
</tr>
<tr>
<td>or1200_cpu</td>
<td>best_leakage_power</td>
<td>dom_0P00V_M1P00V_rbb</td>
</tr>
<tr>
<td>or1200_cpu</td>
<td>best_leakage_power</td>
<td>dom_0P00V_M1P00V_rbb</td>
</tr>
</tbody>
</table>

**NOTE:** Library selection results shown in the following table:

<table>
<thead>
<tr>
<th>Design</th>
<th>Selection Criteria</th>
<th>Library Domain Selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>or1200_cpu</td>
<td>best_leakage_power</td>
<td>dom_0P00V_0P00V_rbb</td>
</tr>
<tr>
<td>or1200_cpu</td>
<td>best_leakage_power</td>
<td>dom_0P00V_M1P00V_rbb</td>
</tr>
<tr>
<td>or1200_cpu</td>
<td>best_leakage_power</td>
<td>dom_0P00V_M1P00V_rbb</td>
</tr>
</tbody>
</table>

---

**Total Power (nW)**

<table>
<thead>
<tr>
<th>Design</th>
<th>WNS</th>
<th>Area</th>
<th>Leakage(nW)</th>
<th>Dynamic(nW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>dom_0P00V_0P00V_rbb</td>
<td>0.0</td>
<td>17443.9</td>
<td>8827431.3</td>
<td>27912797.5</td>
</tr>
<tr>
<td>dom_0P00V_0P00V_fbb</td>
<td>0.0</td>
<td>14543.3</td>
<td>22365217.9</td>
<td>27671021.7</td>
</tr>
<tr>
<td>dom_0P00V_M1P00V_rbb</td>
<td>-104.7</td>
<td>28787.5</td>
<td>7524229.3</td>
<td>32266200.8</td>
</tr>
<tr>
<td>dom_0P00V_M1P00V_fbb</td>
<td>0.1</td>
<td>14522.9</td>
<td>48642126.9</td>
<td>32573354.0</td>
</tr>
<tr>
<td>dom_1P00V_1P00V_rbb</td>
<td>-676.3</td>
<td>27454.5</td>
<td>3872347.3</td>
<td>36007518.9</td>
</tr>
<tr>
<td>dom_1P00V_1P00V_fbb</td>
<td>0.0</td>
<td>14093.3</td>
<td>128142848.7</td>
<td>46680329.9</td>
</tr>
</tbody>
</table>

**NOTE:** Total Power (nW) calculated according to the design and library domain selected.
A design is needed in memory if the user wants to use the `analyze_library_selection_criteria` instance attribute. Any DB file from the output directory can be read before applying the attribute.

```bash
genus:/> set_attribute analyze_library_selection_criteria best_area [find / -instance or1200_cpu]
Setting attribute of instance 'or1200_cpu': 'analyze_library_selection_criteria' = best_area
genus:/> analyze_library_assignment -slack -10 -default_sel best_leakage_power -outdir myodir -reselect
```

### Library Assignment Results

<table>
<thead>
<tr>
<th>Instance</th>
<th>WNS</th>
<th>Area (μm²)</th>
<th>Leakage (nW)</th>
<th>Dynamic (nW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>dom_0P00V_0P00V_rbb</td>
<td>0.0</td>
<td>17443.9</td>
<td>8827431.3</td>
<td>27912797.5</td>
</tr>
<tr>
<td>dom_0P00V_0P00V_fbb</td>
<td>0.0</td>
<td>14543.3</td>
<td>22365217.9</td>
<td>27671021.7</td>
</tr>
<tr>
<td>dom_0P00V_M1P00V_rbb</td>
<td>-104.7</td>
<td>20787.5</td>
<td>7524229.3</td>
<td>32266200.8</td>
</tr>
<tr>
<td>dom_0P00V_M1P00V_fbb</td>
<td>0.1</td>
<td>14522.9</td>
<td>48642126.9</td>
<td>32573354.0</td>
</tr>
<tr>
<td>dom_1P00V_M2P00V_rbb</td>
<td>-676.3</td>
<td>27454.5</td>
<td>3872347.3</td>
<td>36007518.9</td>
</tr>
<tr>
<td>dom_1P00V_M2P00V_fbb</td>
<td>0.0</td>
<td>14093.3</td>
<td>128142848.7</td>
<td>46688329.9</td>
</tr>
</tbody>
</table>

**NOTE:** Library selection results shown in the following table:

<table>
<thead>
<tr>
<th>Instance</th>
<th>Selection Criteria</th>
<th>Library Domain Selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>or1200_cpu</td>
<td>best_area</td>
<td>dom_0P00V_0P00V_rbb</td>
</tr>
<tr>
<td>or1200_cpu</td>
<td>best_area</td>
<td>dom_0P00V_0P00V_fbb</td>
</tr>
<tr>
<td>or1200_du</td>
<td>best_leakage_power</td>
<td>dom_0P00V_M1P00V_rbb</td>
</tr>
<tr>
<td>or1200_tt</td>
<td>best_leakage_power</td>
<td>dom_0P90V_M1P80V_rbb</td>
</tr>
</tbody>
</table>
Selection of Area/Power Tradeoff Criteria

- The user can tradeoff power and area using the area_tradeoff_power, area_tradeoff_leakage, power_tradeoff_area and leakage_tradeoff_area options, along with the -tradeoff_range_percentage switch.
- In this example, a library set with lower area was chosen because the user allowed power to increase by 45% over the minimum.

```
legacy_genus:/> analyze_library_assignment -slack -10 -default_sel area_tradeoff_power -tradeoff_range_percentage 45 -outdir myodir -reselect

<table>
<thead>
<tr>
<th>Instance</th>
<th>Selection Criteria</th>
<th>Library Domain Selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>or1200_cpu</td>
<td>area_power_tradeoff</td>
<td>dom_0P00V_0P00V_fbb</td>
</tr>
<tr>
<td>or1200_cpu</td>
<td>area_power_tradeoff</td>
<td>dom_0P00V_M1P00V_fbb</td>
</tr>
<tr>
<td>or1200_cpu</td>
<td>area_power_tradeoff</td>
<td>dom_1P00V_M2P00V_fbb</td>
</tr>
<tr>
<td>or1200_du</td>
<td>area_power_tradeoff</td>
<td>dom_0P00V_M1P00V_rbb</td>
</tr>
<tr>
<td>or1200_tt</td>
<td>area_power_tradeoff</td>
<td>dom_0P00V_M1P00V_rbb</td>
</tr>
</tbody>
</table>
```

NOTE: Library selection results shown in the following table:

<table>
<thead>
<tr>
<th>Instance</th>
<th>Selection Criteria</th>
<th>Library Domain Selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>or1200_cpu</td>
<td>area_power_tradeoff</td>
<td>dom_0P00V_0P00V_fbb</td>
</tr>
<tr>
<td>or1200_cpu</td>
<td>area_power_tradeoff</td>
<td>dom_0P00V_M1P00V_fbb</td>
</tr>
<tr>
<td>or1200_cpu</td>
<td>area_power_tradeoff</td>
<td>dom_1P00V_M2P00V_fbb</td>
</tr>
<tr>
<td>or1200_du</td>
<td>area_power_tradeoff</td>
<td>dom_0P00V_M1P00V_rbb</td>
</tr>
<tr>
<td>or1200_tt</td>
<td>area_power_tradeoff</td>
<td>dom_0P00V_M1P00V_rbb</td>
</tr>
</tbody>
</table>

22FDX : Analyze Library Assignment Results

- Selection of Area/Power Tradeoff Criteria

- The user can tradeoff power and area using the area_tradeoff_power, area_tradeoff_leakage, power_tradeoff_area and leakage_tradeoff_area options, along with the -tradeoff_range_percentage switch.
- In this example, a library set with lower area was chosen because the user allowed power to increase by 45% over the minimum.
Selection of Area/Power Tradeoff Criteria (Continue)

The behavior of these “tradeoff” options can be described as follows.

- `area_tradeoff_power` – choose the least area within a total power range
- `area_tradeoff_leakage` – choose the least area within a leakage power range
- `power_tradeoff_area` – choose the least total power within an area range
- `leakage_tradeoff_area` – choose the least leakage power within an area range

The “range” is calculated as the minimum power or area that meets timing, with the `tradeoff_range_percentage` number specified on the command line.

For example, if you are using “leakage_tradeoff_area”, and the minimum area that meets timing for is 10000, and the range percentage specified is 20%, then the utility will choose the library domain with the minimum leakage power, within an area range of 10000 to 12000 (10000 + 10000*0.20)
• Additional files written by analyze_library_assignment

• The following files are written into the output directory
  - start.db – a Genus DB file written by the utility before any synthesis is executed.
  - <domain_name>.db – a Genus DB file written after synthesis is executed targeting the specified library domain.
  - <instance_name>.csv – a CSV file with the PPA results for the specified instance.
  - domain_selection.tcl – a TCL script that applies the selected library_domain to each instance analyzed.
    ▪ set_attribute library_domain dom_0P00V_0P00V_rbb [find / -instance or1200_cpu]
    ▪ set_attribute library_domain dom_0P00V_M1P00V_rbb [find / -instance or1200_du]
    ▪ set_attribute library_domain dom_0P00V_M0P00V_rbb [find / -instance or1200_tt]
  - final_synth.tcl – a very simple Genus script that applies the selected library_domain settings and runs synthesis.
    ▪ if [llength [find / -design *]] {rm /des*/*}
    ▪ read_db myodir.genus.2/start.db
    ▪ source myodir.genus.2/domain_selection.tcl
    ▪ syn_gen; syn_map; syn_opt
Agenda

• GLOBALFOUNDRIES Overview
• What is 22FDX Technology?
• Body-Biasing: A New Dimension in Design Closure
• Automation of Design Planning
• Analyze Library Assignment Flow Details
• Results
• Conclusion
Bias Voltage design with GLOBALFOUNDRIES’ 22FDX technology is a new dimension in Power, Performance and Area tuning.

The GENUS based Automatic Body Bias Solution With GlobalFoundries 22 FDX technology greatly enhances the Designer’s Capabilities to meet the Design Targets and significantly reduce the turn around time.

The Utility with Design Example will be available for download @GLOBALFOUNDRIES FoundryView.