

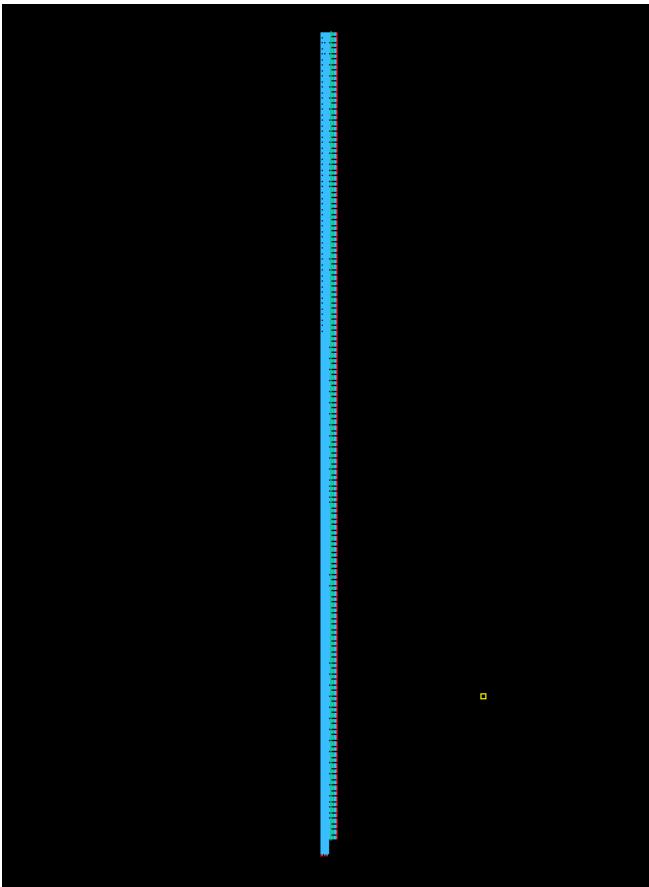
# **SRAM Predecoder and Row Decoder Design in 16nm FinFET**

**Nandini Kumawat**

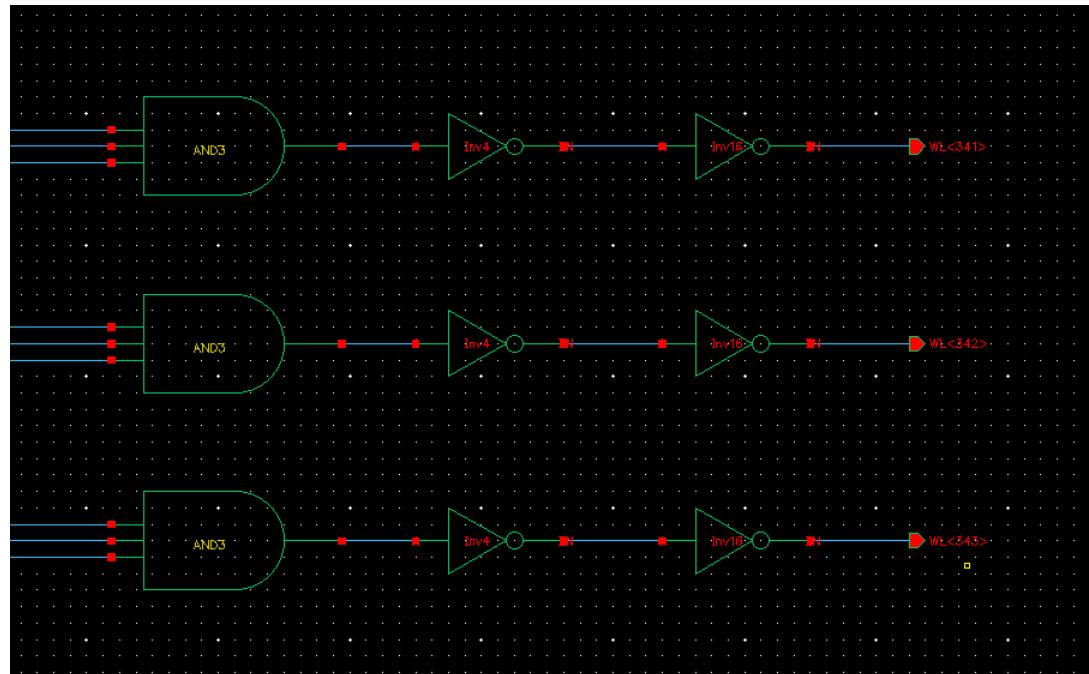
**December 8, 2025**



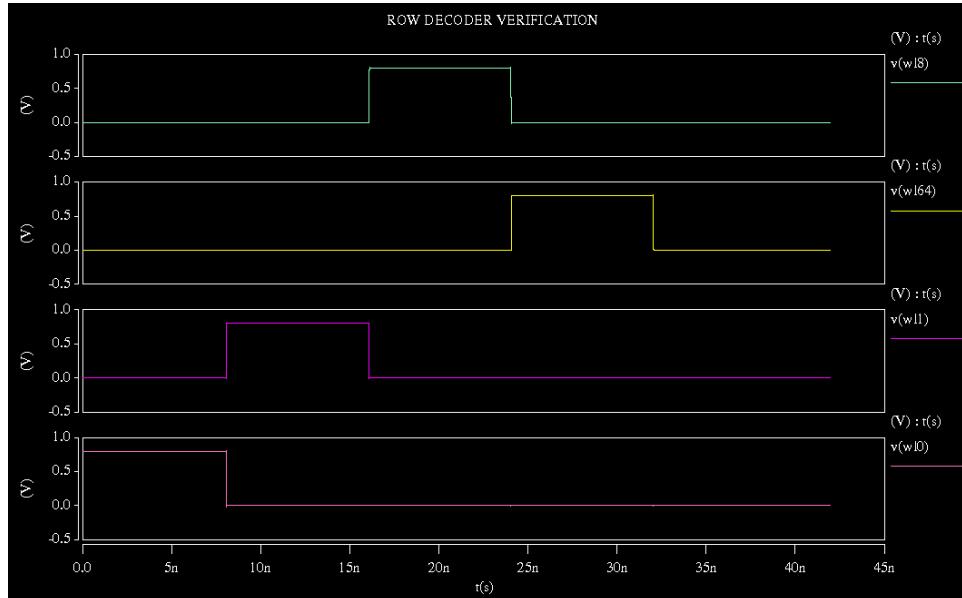
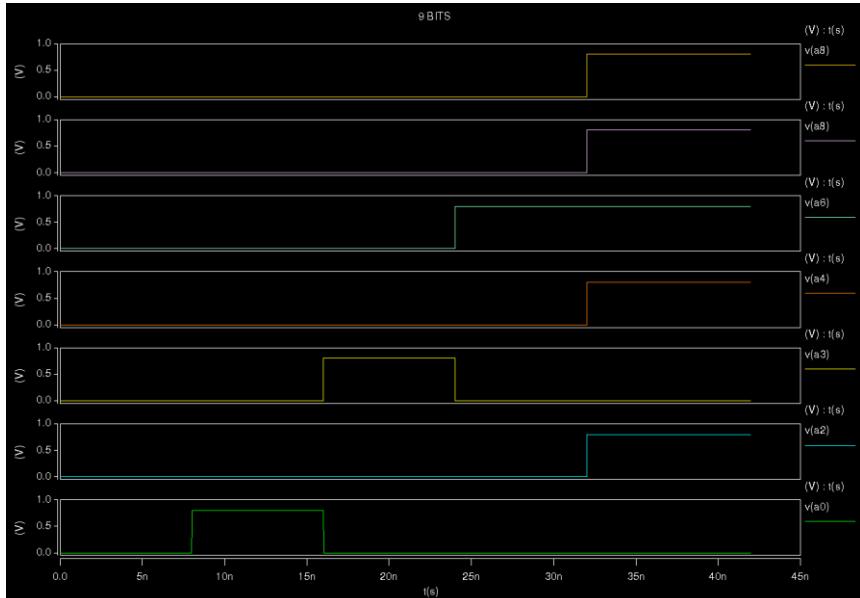
# Automated Schematic using SKILL



- SKILL-based schematic generation for predecoder and WL driver connectivity
- Consistent net naming and instance naming for scaling
- Reduced manual errors in repeated structures



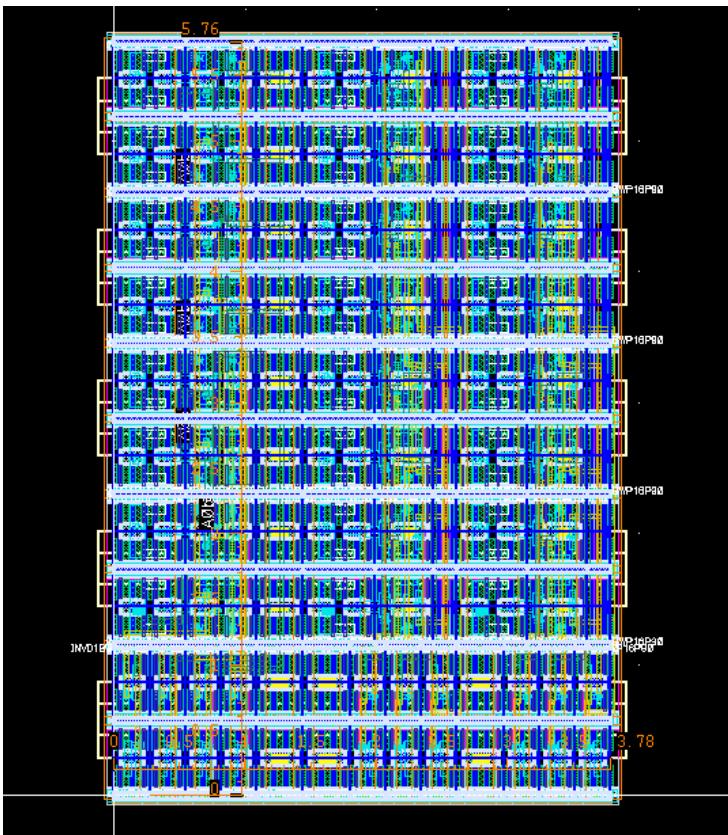
# Verification of Wordline Signals



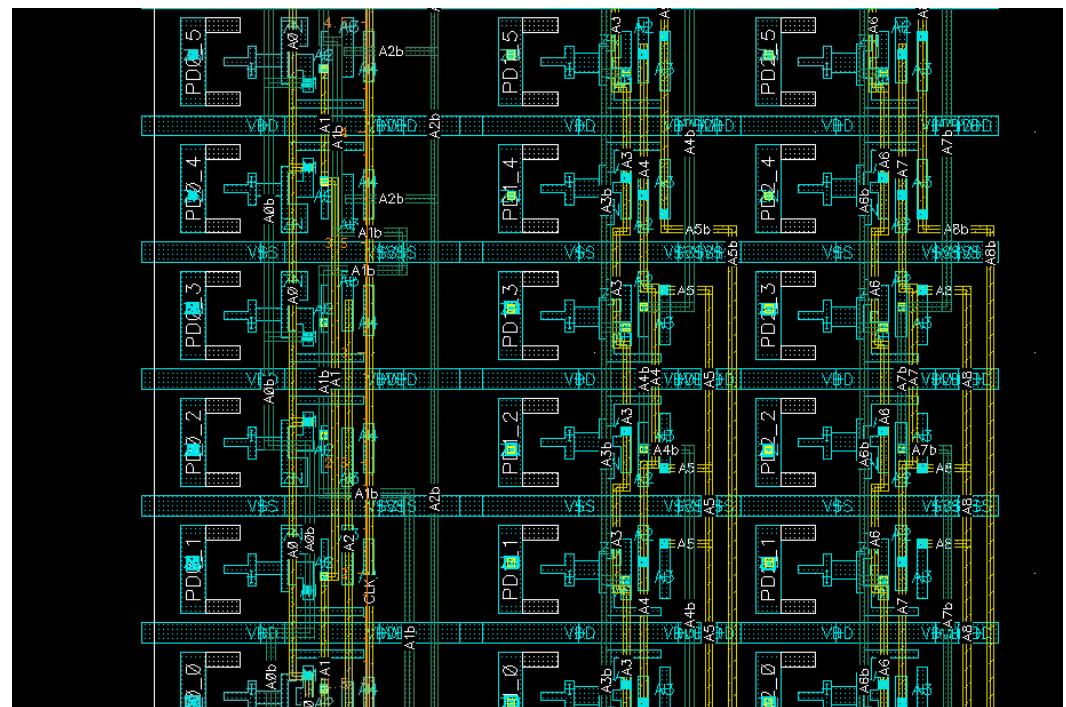
- Verified WL functional correctness using signal tracing and net checks
- Confirmed one-hot behavior across WLs for address patterns
- Checked continuity from predecoder outputs to WL driver inputs



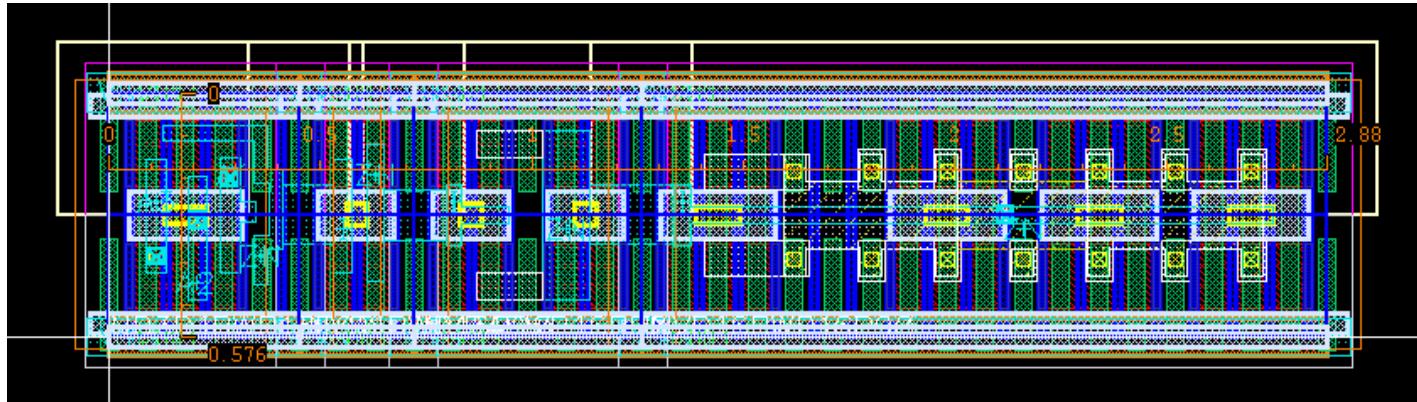
# Predecoder Layout + Signal Tracing



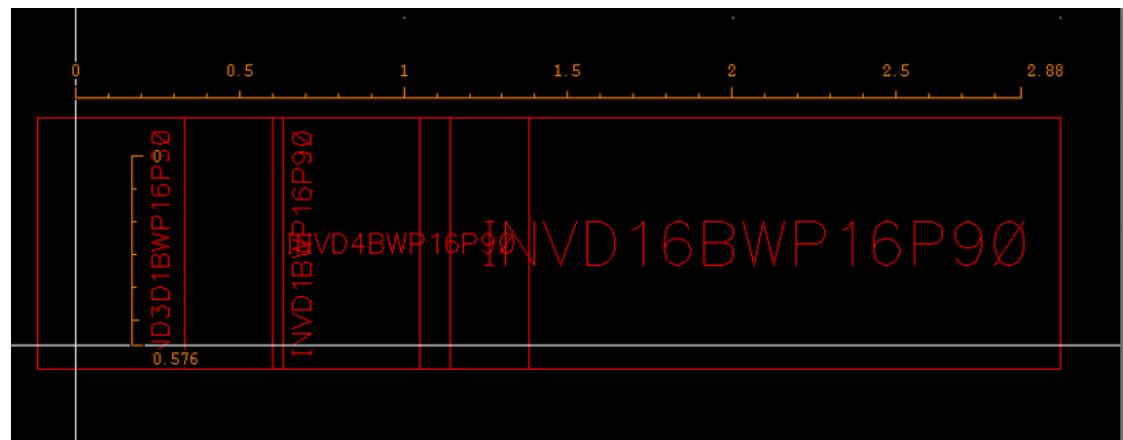
- Predecoder layout completed with hierarchical integration approach
  - Traced critical nets end-to-end for label and connectivity sanity
  - Focus on routability and clean pin accessibility



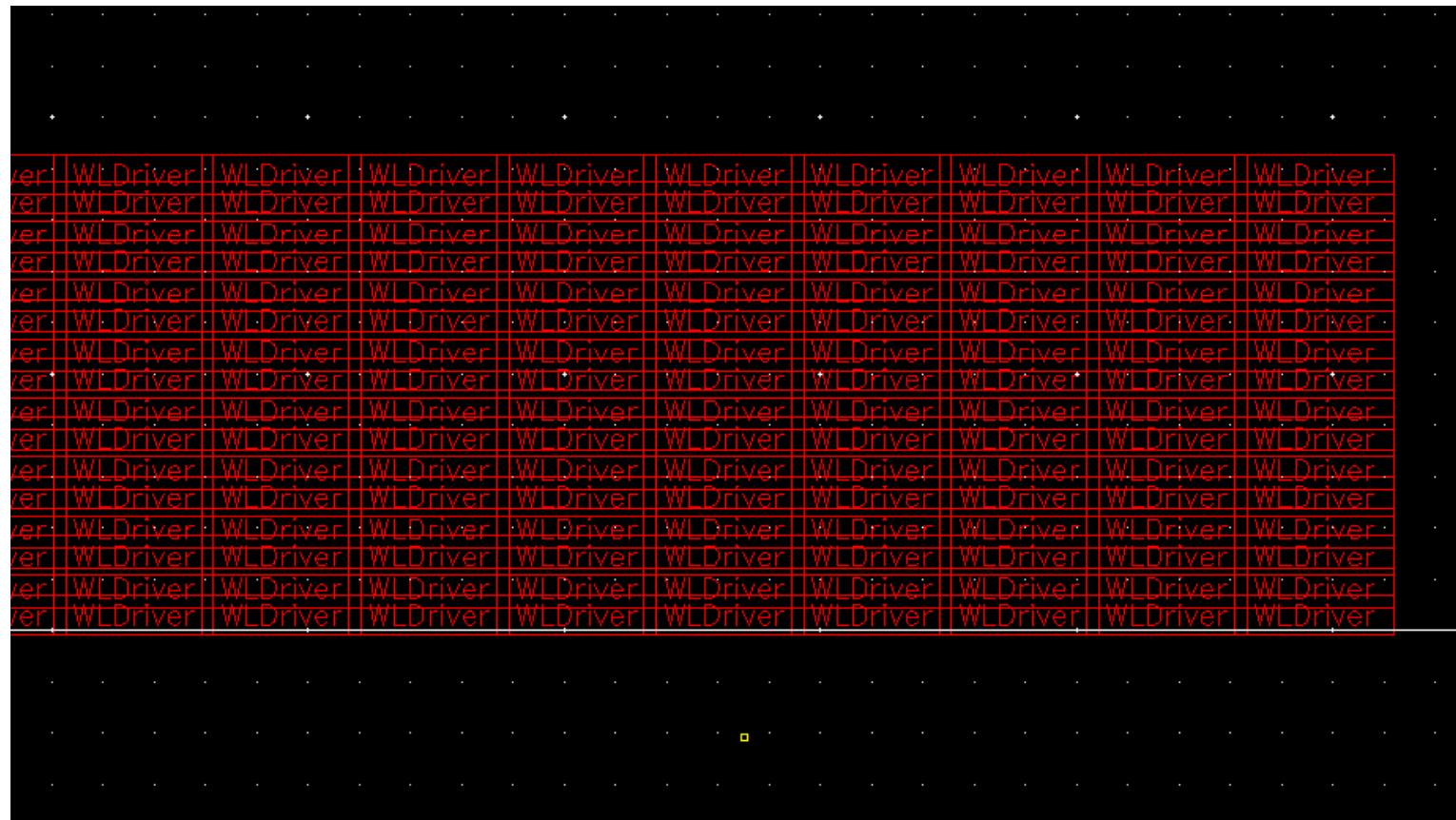
# Wordline Driver and Sizing



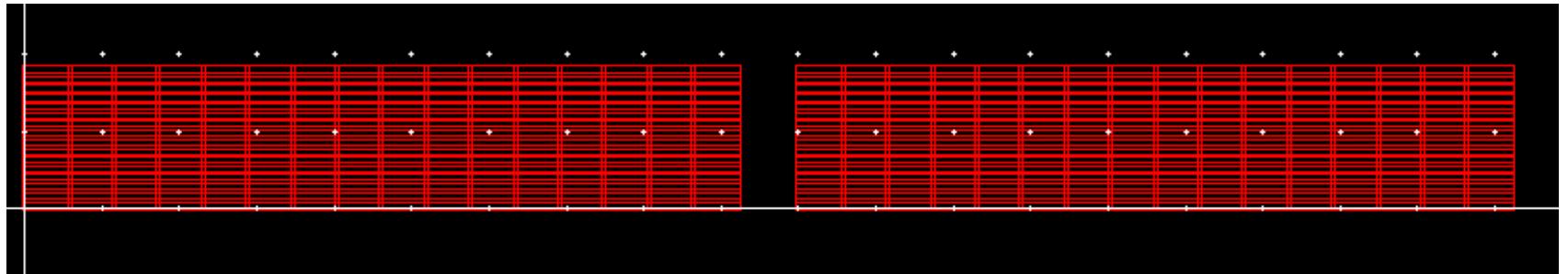
- WL driver sized to match load of 256 SRAM cells
- Physical sizing aligned to 32 poly-pitch width for row integration
- Driver strength chosen to meet WL RC and edge-rate needs under full row load



# 16X16 Array of Wordline Drivers (SKILL)

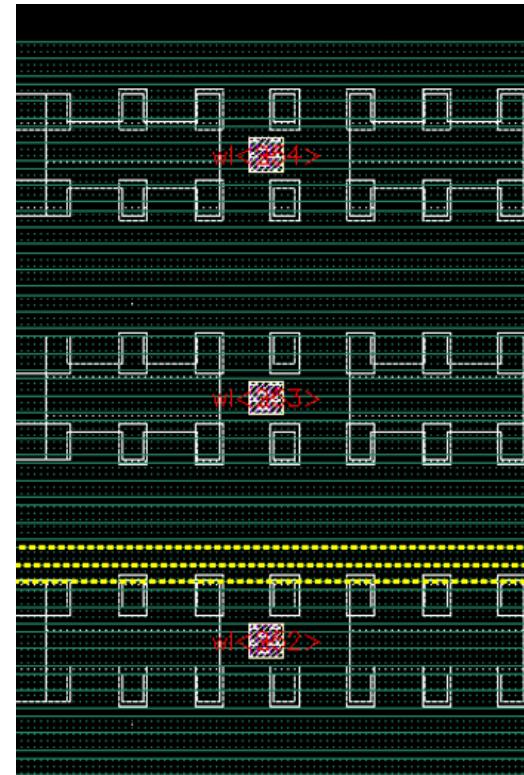


*Generated 16x16 WL driver array using SKILL for repeatability*

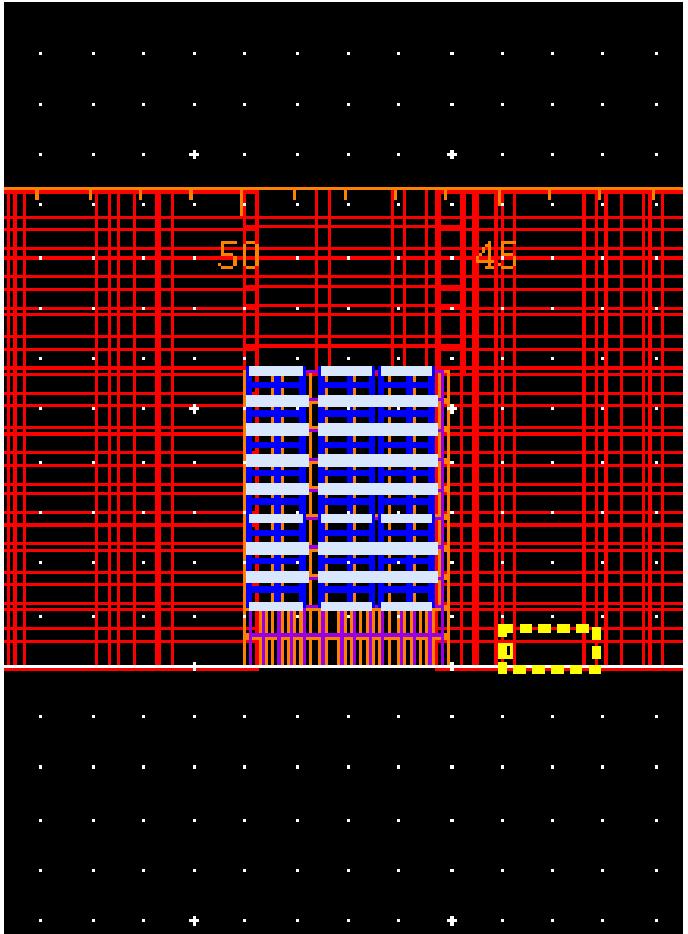


- Structured pin grid for clean top-level routing access
- Consistent pin ordering for WL indexing and debug
- Reduced routing ambiguity during integration

16 X 16 X 2 = 8X8X8 Matrix  
Pin creation and placing was done  
using Cadence SKILL.

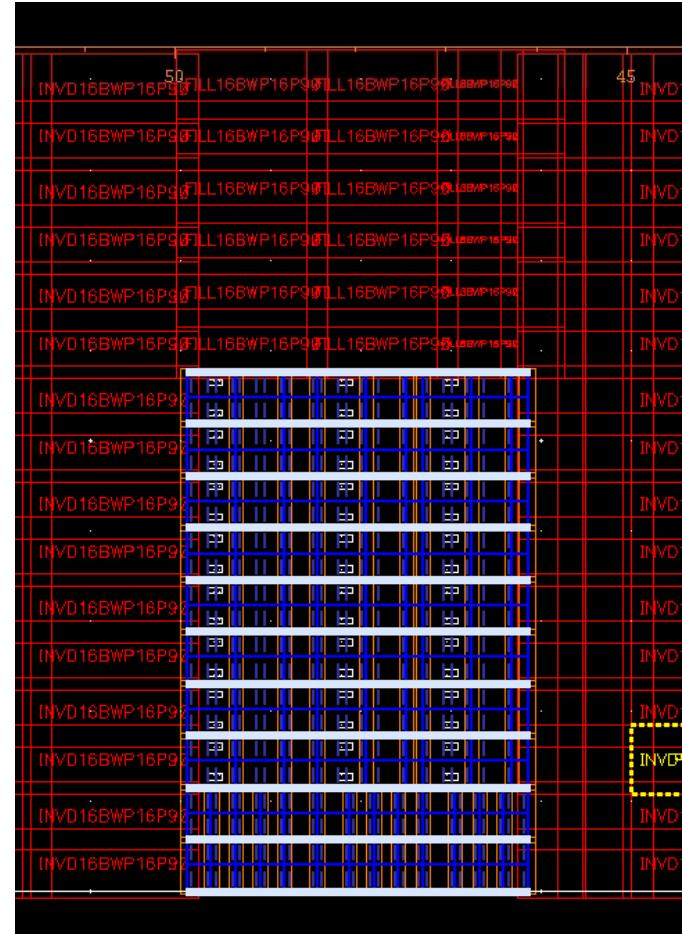


# Placement of WL Drivers and Predecoder

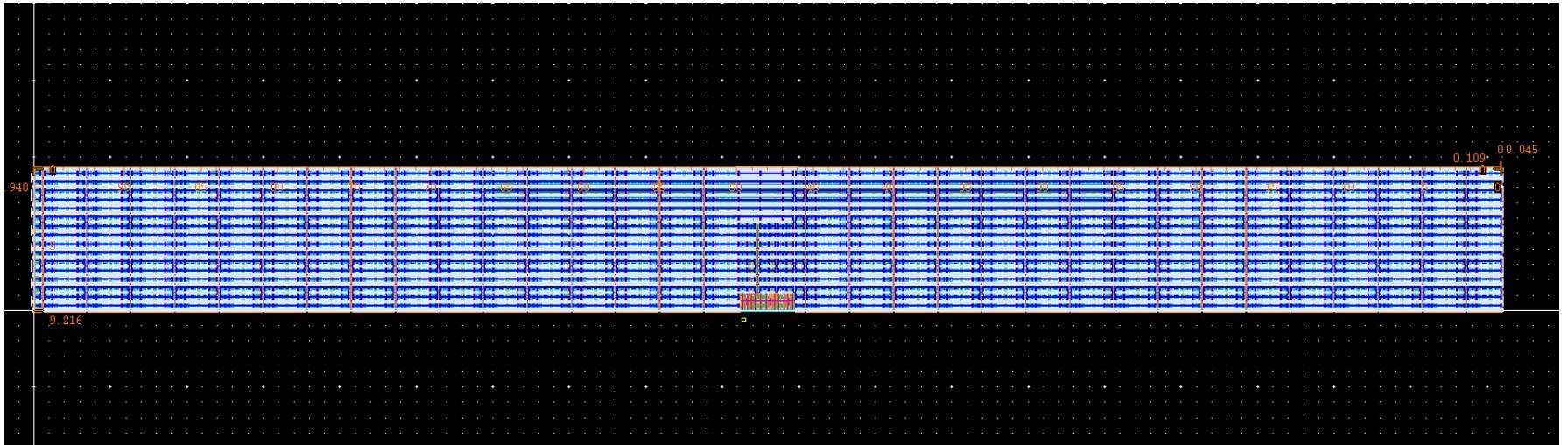


*Co-placed  
predecoder and  
WL drivers to  
minimize  
wirelength and  
placement  
optimized for  
dense packing and  
routing lanes*

*Inserted Fillers*



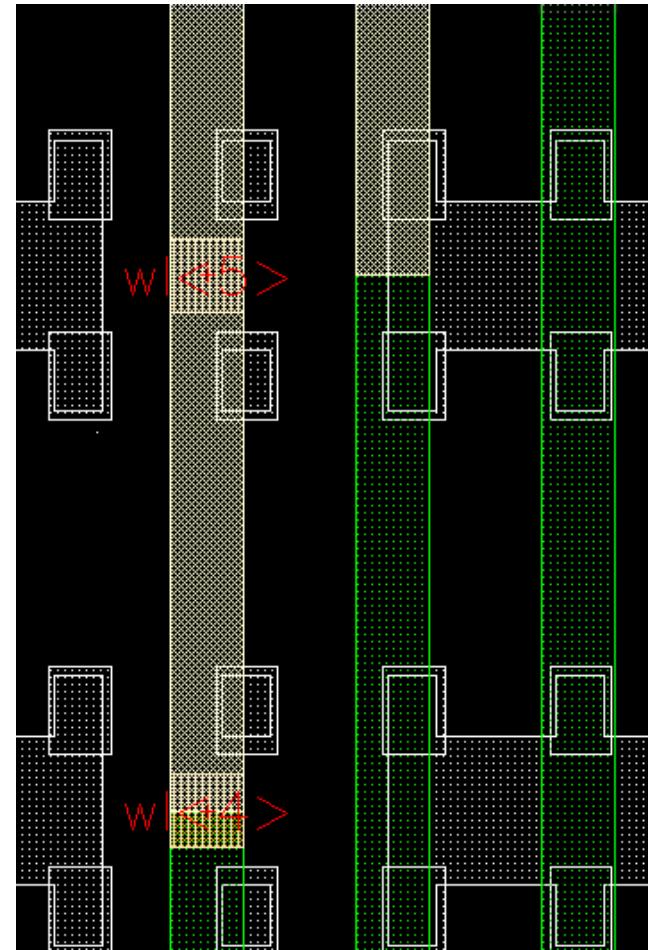
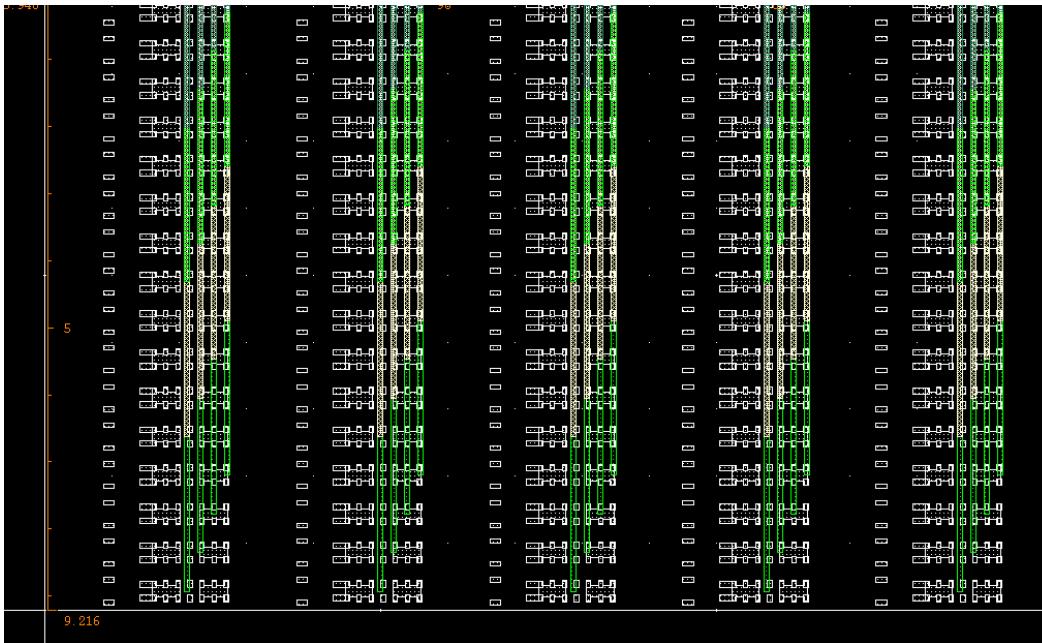
# All Cells Placed



- *Complete placement of required standard cells*
- *WL driver chain implemented as NAND3 + INV1 + INV4 + INV16*
- *No filler cells inside the WL driver chain to maximize density and reduce extra parasitics*



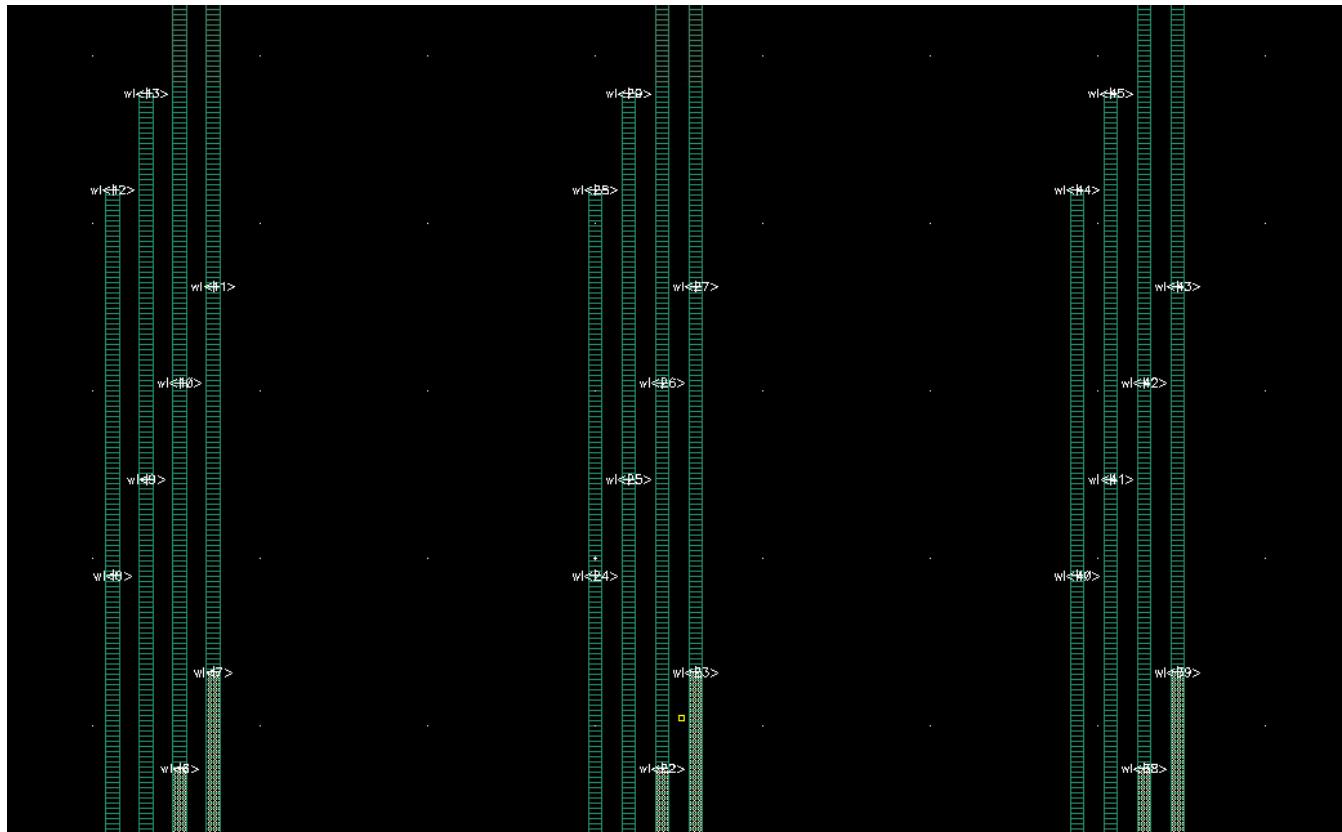
# Strategic Routing of WL Outputs



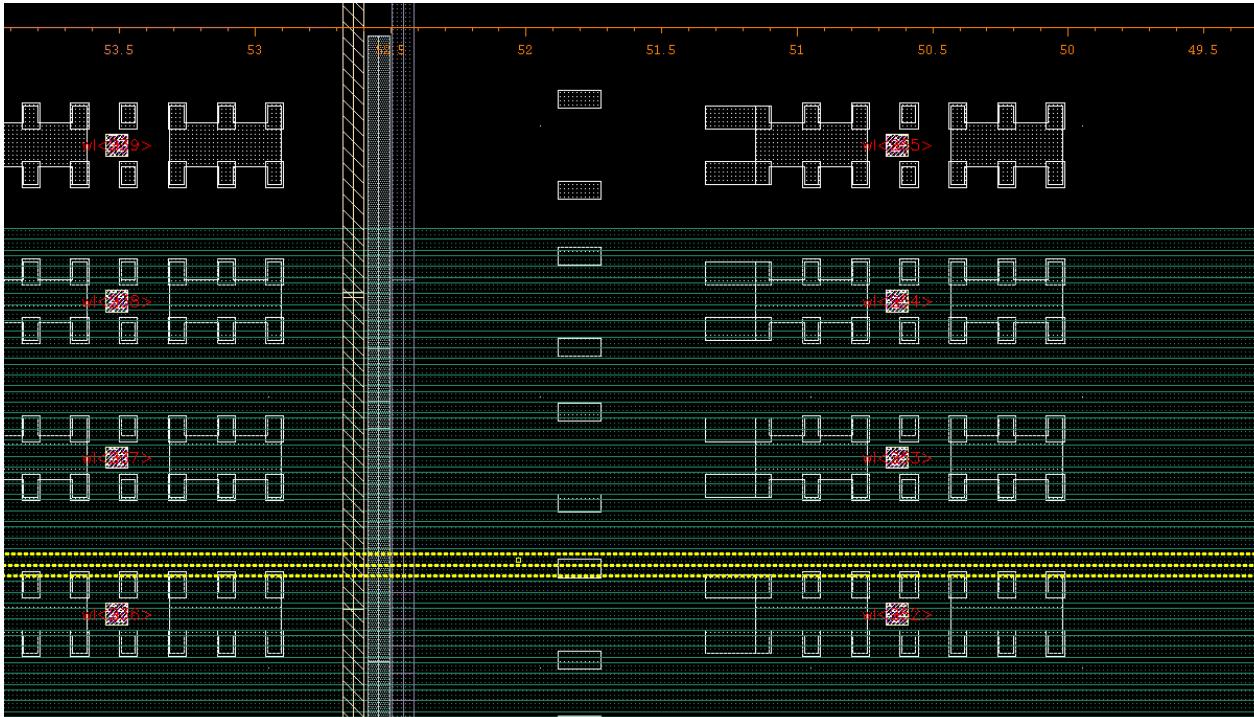
- Routed predecoder outputs with consistent layer usage and via strategy
- Kept WL output paths uniform to limit skew across rows
- Prioritized clean access to WL driver inputs



- *WL0 to WL255 naming applied consistently across hierarchy*
  - *Labeling supports debug, and integration with SRAM array*
  - *Ensured readable WL annotation in top-level layout*
  - *Used different metal layers for first 4 outputs in an array and moved to a different metal layer every 4 cells in a column to avoid overlap.*



# WL Drivers Strategic Routing

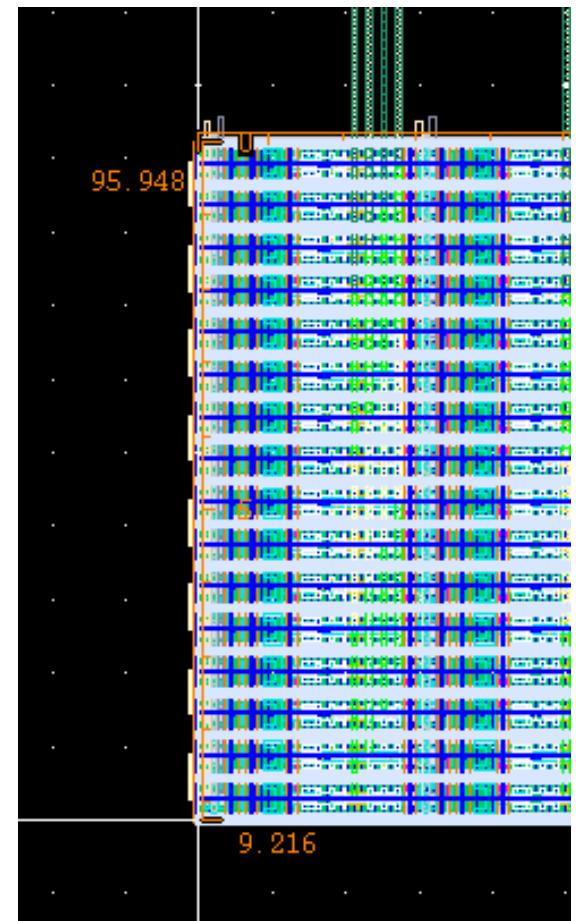
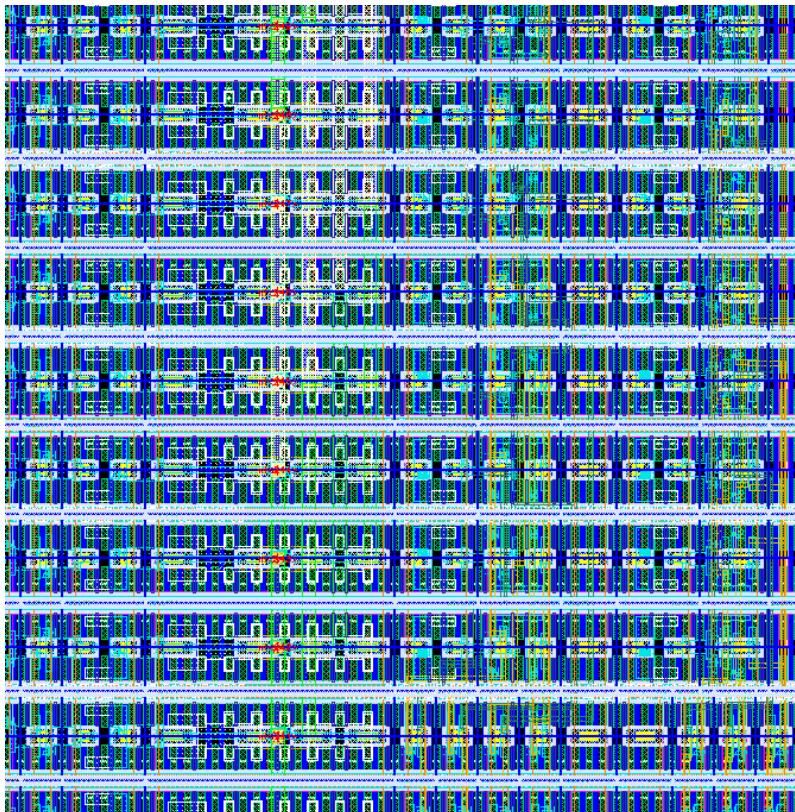


- Routed driver outputs with shortest, widest practical paths where needed
- Focus on minimizing WL series resistance and via count
- Avoided detours to keep WL latency controlled



# Dense Placement and Area

- *Dense standard-cell style packing for area efficiency*
- *Kept routing congestion manageable through structured placement*
- *Density-first layout choices while maintaining signal traceability*



$$\text{Area} = 9.216 \times 95.948 = 884.717568 \mu\text{m}^2$$



## **Google Drive link to all the SKILL Files, Schematic and Layout**

Schematics:

/home/kumaw010/tsmcN16\_Nandini/NandiniSRAM/rowDecoder512  
/home/kumaw010/tsmcN16\_Nandini/NandiniSRAM/subblock1  
/home/kumaw010/tsmcN16\_Nandini/NandiniSRAM/subblock2  
/home/kumaw010/tsmcN16\_Nandini/NandiniSRAM/subblock3

Layout:

/home/kumaw010/tsmcN16\_Nandini/NandiniSRAM/PredecoderComplete  
/home/kumaw010/tsmcN16\_Nandini/NandiniSRAM/WLDriver  
/home/kumaw010/tsmcN16\_Nandini/NandiniSRAM/rowDecoder512\_complete5  
/home/kumaw010/tsmcN16\_Nandini/NandiniSRAM/wl\_driver\_array\_16x16



