🧱 Static Arrays

🔹 Fixed size arrays

Size defined at compile time. Example: int my_array[5]; ✅ Use when size is known.

🔄 Dynamic Arrays

🔹 Size can change at runtime

Declared with empty brackets []. Example: int my_array[]; my_array = new[10]; ✅ Use for flexible storage.

🧩 Packed vs Unpacked Arrays

🔷 Packed Array

Bit-level, all dimensions packed. Example: logic [3:0][7:0] data; 🔶 Unpacked Array: Regular multi-dimensional.

🗂️ Associative Arrays

🔑 Key-value storage

Key can be int, string, enum. Example: int my_map[string]; my_map["temp"] = 100; ✅ Best for sparse data.

🌀 Queues

🔸 Built-in list type

Auto-resizing, operates like FIFO. Declare: int my_queue[$]; ✅ Ideal for streaming.

🎯 Summary

✨ SystemVerilog Array Types

Static: Fixed, no resize. Dynamic: Runtime, resizable. Associative: Sparse, resizable. Queue: FIFO, resizable.

💡 Packed vs Unpacked

💡 Packed = Bit-level

💡 Unpacked = Multi-dim structure.

📣 Final Slide: Master These!

📚 Arrays are everywhere!

Mastering them leads to clean, efficient code! Drop a 💻 if you're learning SystemVerilog!