**mining\_asic\_top.sv**

**1. Top-Level Module: mining\_top**

* Acts as the **Grand Dispatcher**
* Accepts:
  + A 512-bit block (mining job)
  + A nonce range
* Distributes work across multiple child\_units
* Collects and reports **winning hash and nonce**

**2. Mid-Level Module: child\_unit**

* Acts as the **Child Worker Unit**
* Contains:
  + Multiple hash\_cores (parameterized)
  + Assigns portions of nonce space to each core
* Aggregates and returns hash results to the top level

**3. Leaf-Level Module: hash\_core**

* Acts as the **Grandchild / Compute Core**
* Responsible for:
  + Executing the SHA-256 hash function
  + Accepting job + nonce
  + Returning a 256-bit hash output
* Currently includes a **pipeline stub** — ready for full SHA-256 implementation

**4. Support Module: sha256\_round\_func**

* Implements **1 round** of SHA-256 compression
* Can be used to build a full pipeline or loop

**5. Config Package: mining\_cfg\_pkg**

* Centralizes:
  + Parameters (number of cores, clock target, pipeline depth)
  + Register map (optional for MMIO interface)
  + Helps with easy tuning and synthesis constraints

**Status**

| **Layer** | **Purpose** | **Status** | **Notes** |
| --- | --- | --- | --- |
| mining\_top | Dispatcher + result aggregator | ✅ Complete | Parametrized fan-out logic |
| child\_unit | Job splitter to cores | ✅ Complete | Uses nonce range and distributes to hash\_cores |
| hash\_core | SHA-256 hashing engine | 🔶 Stub | Placeholder pipeline — ready for full logic |
| sha256\_round\_func | Combinational SHA-256 round | ✅ Complete | Can be used to build full pipeline |