**MRAM Subsystem and Architecture Research**

**(For Academic and Research Use Only – Not for Commercial Purposes)**

*This document is intended solely for academic study and internal research purposes.  
It may not be used, distributed, or referenced for any commercial development, marketing, or licensing without explicit permission from the author.*

**Overview**

This research outlines the architecture and design of a high-efficiency **MRAM subsystem**, developed to interface MRAM macros with AMBA-compliant SoC platforms. The design addresses the unique characteristics of MRAM—particularly its **non-volatility**, **multi-cycle latency**, and **access alignment constraints**—while ensuring compatibility with standard system bus protocols.



**Architectural Highlights and Technical Features**

**1. Protocol-Compliant Bus Interface**

* Full AMBA protocol support, including burst, single, and misaligned transaction management.
* Dynamic backpressure and ready signaling ensure stable handshaking with system masters.

**2. Read Performance Optimization**

* Prefetch logic coupled with a FIFO buffer masks MRAM’s inherent latency.
* Threshold-based release mechanism provides configurable balance between performance and power.

**3. Robust Error Management**

* Hardware-level detection of:
  + Address misalignment
  + Invalid partial writes
* Generates bus error responses and system interrupts for graceful exception handling.

**4. Autonomous Power-On Control**

* Built-in FSM handles MRAM power-up, register configuration, and readiness check.
* Removes dependency on firmware or MCU-side initialization code.

**5. Testability and Diagnostic Support**

* Integrated error counters for single- and multi-bit failure analysis.
* Software-triggered reset and BIST (Built-In Self-Test) status accessible via memory-mapped registers.

**6. Modular and Scalable Architecture**

* Internal architecture includes:
  + Protocol control state machines
  + MRAM command generation unit
  + Data alignment logic
* Parameterized for reuse across SoC generations and technology nodes.

**PPA-Oriented Design Considerations**

| **Category** | **Design Impact** |
| --- | --- |
| **Power** | ✔ No refresh cycles; ideal for always-on low-power domains |
| **Performance** | ✔ FIFO prefetch hides latency and improves throughput |
| **Area** | ❗ Slightly increased logic due to error handling and control FSMs, balanced by MRAM’s in-place memory integration |