

System Level Error Handling

Circuit-Level EDAC

- Disadvantages
 - Expensive in power and mass if used to protect all circuits
 - Can be defeated by multiple-bit transient errors
- Advantages
 - Much faster response than system-level EDAC
 - Some protection is gained by simply following good design practices
 - e.g., not allowing lock-up states

System Level EDAC

- Advantages
 - Required for hard-failure handling, so is already a part of the system
 - Relies on inherent redundancy in system, high-level error checking, and some EDAC hardware, so additional required hardware is minimized
- Disadvantages
 - Detection time may be longer than for circuit level EDAC
 - Recovery time may be problem in critical mission phases

System-Level Error Checking Mechanisms

- Natural error checking mechanisms
 - e.g., fire a thruster, check for spacecraft attitude change
- Checking mechanisms arising from multiple subsystems
 - e.g., command a module to power on, check its current draw and temperature
- Explicitly added checking mechanisms
 - Watchdog timers
 - Handshake protocols for command acknowledgement
 - Monitors, e.g., thruster on-time monitor

Error: SEU Causes FSM Jump to Legal State

- Will not be detectable by illegal state detection logic
 - Will probably cause the circuit to operate incorrectly
 - Will probably be detectable at system level only based on incorrect module operation
- => If the circuit operation is critical, e.g., could cause loss of mission, design so that this cannot happen

Error: SEU Causes FSM Jump to Illegal State

- Essentially two choices to handle this:
 - Provide homing sequence from illegal state to a legal state
 - Becomes an erroneous jump to legal state
 - No indication there was a problem
 - Detect with illegal state detection logic
 - Three choices to handle this:
 - Transfer to legal state
 - Reset circuit and start again
 - Stop, raise flag, handle at system level
 - Will get some indication there was a problem
- Recovery success depends on nature of system
- Suppress FSM outputs in illegal states

System-Level Error Handling Mechanisms Also Handle Transient Error Effects

Transient Error Effect	System Response
Command Rejection	Command Retry
Telemetry or Data Corruption	Data Filtering, also required to handle system noise
State Machine Lock-up, e.g., detected by multiple command rejections	Indistinguishable from hard error

Impact of Adding EDAC to Common FSM Types

FSM Type	Protecting with EDAC
Binary	High encoding efficiency => smallest EDAC impact Potentially few illegal states => fairly easy to detect Full decoding eliminates effects of illegal states
One-hot	Poor encoding efficiency => greatest EDAC impact Many illegal states => complex circuit to detect Full decoding defeats advantage of easy state decoding

Conclusion: Optimal State Machine

- Binary state machine may be optimal for highly reliable systems
 - Most amenable to the addition of EDAC circuitry if necessary because of high encoding efficiency
 - Full state decoding protects against erroneous outputs
 - Easier to detect illegal states

Conclusion: Overall EDAC Scheme

- Overall EDAC scheme will also use system level EDAC
 - Will be there for hard failures, anyhow
 - Must consider system response to defeated circuit-level EDAC
 - Some critical circuits may require system intervention if errors are detected in them