**Objective**
- Develop high-confidence (embedded) systems
- Target multicore SBCs (Intel Joule, Edison, etc)
- Mixed criticalities: Timeliness & Safety
- Predictable, Secure, Fault Tolerant

**Applications**
- Healthcare
- Avionics
- Automotive
- Factory Automation
- Robotics
- Space exploration
- IoT, Industry 4.0 “Smart factories”

**Approach**
- Quest-V for multicore processors
- Distributed system on a chip
- Time as a first-class resource
- Sandboxes sub-components using hardware-assisted memory virtualization (e.g., Intel EPTs)

**Example Mixed Criticality System**

**Isolation**
- Memory virtualization using shadow paging isolates sandboxes and their components
- Dedicated physical cores assigned to sandboxes
- I/O passthrough for direct device access
- Hardware performance monitoring for shared cache and bus isolation
- Monitors blacklist illegal port-based I/O device accesses
- EPTs prevent unauthorized memory-mapped device access

**Predictability**
- Virtual CPUs (VCPUs) for time budgeted real-time execution of threads and system events (e.g., interrupts)
- Sandbox kernels perform local scheduling on assigned cores
- Avoid VM-Exits to Monitor – eliminate cache/TLB flushes

**Recent Developments**
- Real-time fault detection and recovery (e.g., Triple Modular Redundancy of replicated sandboxes)
- MARACAS – memory-aware, real-time aware, cache-aware scheduling for multicores
- QduinoMC – Quest-Arduino real-time multithreaded (& multicore) API
- World’s first secure & smart 3D printer

**Conclusions**
- Quest-V real-time chip-level separation kernel
- Uses hardware virtualization for time and space isolation of guest services
- Eliminates overheads of a hypervisor

**Quest Website:** [www.questos.org](http://www.questos.org)