Lecture 03 (System Architecture)

Won't cover initial few chapters from Seshia book (related to FSM design)

1 Uploading Code to Arduino Board

1. AES takes about 25Kb/s (35 microseconds per byte)
2. If 16 bytes of data is to be sent, then about 0.5ms of overhead is observed
3. This is about 15% overhead total (encryption + decryption) for a powertrain ECU

2 Design Characteristics

1. Interaction with physical world
2. Specific task
3. Real-time (safety-critical)
4. Large numbers
5. Low cost
6. Resource constraints

3 Design Process

1. Highly optimised
   - interactions among different components
   - detailed implementation
2. Concurrency
   - timing
3. Correctness
   - modelling at high and low abstraction levels

3.1 Example of Concurrency + Correctness

1. A drone was storing temperature and pressure value at the same memory location to reduce the memory consumption
2. Processor reads the wrong value - leads to drastic consequences

4 Operating System (types)

1. No OS
2. Real-time operating system (RTOS)
3. Embedded linux distribution - we will use Raspberry Pi boards in this course

5 CPU (types)

1. Atmel AVR Microcontroller
   - 8-bit
   - Arduino uno
2. MicroChip PIC microcontroller
   - 16-bit
3. ARM Cortex-M microcontroller
   - 32-bit
   - smart-watch etc
4. ARM Cortex-A microcontroller
   - 64-bit
   - Raspberry Pi 4

6 System on Chip (SoC)

1. Different components are all integrated on a single chip
2. Has components like DRAM, processor, memory controller, etc
3. Nothing is going to change in the system (components cannot be modified) - helps in optimisations
4. Some more components are peripherals, interrupt controller, timers, power, clocks, boot flash

7 Parallelism

1. Instruction level
   - instruction pipelining
   - superscalar execution
   - OOO execution
2. Data level
   - SIMD
3. Thread level
   - multithreading

8 ISA

1. ARM

- in-order cores
- low power and lesser area

2. Intel Atom
    - in-order execution
    - data-level parallelism