Difference between Software architecture and Hardware architecture and instructions set architecture of embedded systems :

Hardware Architecture:

-This is the blueprint for the physical components of the embedded system. It defines the building blocks like the processor, memory (RAM and ROM), communication interfaces (USB, SPI), and peripherals (sensors, actuators).

-Hardware architecture focuses on how these components connect and interact with each other to achieve the overall functionality of the system.

-Decisions made during hardware architecture design impact factors like performance, power consumption, and cost.

Software Architecture:

-This blueprint focuses on the software side of the system. It defines the organization of the software components, their communication protocols, and the overall data flow.

-Software architecture includes the embedded operating system (if used), device drivers for peripherals, and the application itself.

-A well-defined software architecture promotes modularity, maintainability, and easier integration of future functionalities.

Instruction Set Architecture :

-This acts as a bridge between hardware and software. It defines the set of instructions the processor can understand and execute.

-The ISA essentially specifies a language that the software uses to communicate with the hardware.

-Different processor families (e.g., ARM, x86) have their own unique ISAs. Understanding the ISA is crucial for programmers to write efficient code that leverages the processor's capabilities.