**Assessment Question**

1. **Describe the Difference between RISC and CISC architectures.**

**RISC (Reduced Instruction Set Computer):** Simplified instructions that execute in a single clock cycle. Examples: ARM, MIPS.

**CISC (Complex Instruction Set Computer):** Complex instructions, which can take multiple cycles to execute. Examples: x86 architecture.

**Key difference:** RISC emphasizes hardware simplicity and software complexity, while CISC focuses on complex hardware to simplify programming.

1. **Explain the Role of the Program Counter (PC) in ARM architecture.**

The PC holds the address of the next instruction to be executed. It gets incremented automatically as instructions execute or updated during branches.

1. **What is the Significance of condition codes in ARM assembly?**

Condition codes determine whether an instruction is executed based on flags (e.g., zero, carry, negative). They enable conditional execution without branching.

1. **How does the ARM pipeline improves performance?**

ARM uses a pipelined architecture where multiple instructions are fetched, decoded, and executed simultaneously. This increases instruction throughput.

1. **Compare Direct and indirect addressing modes in ARM assembly?**

**Direct addressing:** Access memory using an explicit address

**Indirect addressing:** Access memory using a pointer stored in a register.

1. **What is the Difference between LDR and LDM instructions?**

**LDR (Load Register):** Loads a single value from memory into a register.

**LDM (Load Multiple):** Loads multiple values from memory into consecutive registers.

1. **Explain the Use of the Stack Pointer (SP) in subroutine calls?**

The SP manages the stack, storing return addresses, local variables, and register states during subroutine calls.

1. **How are interrupts are handled in ARM architecture?**

ARM uses a vector table to redirect execution to specific interrupt service routines (ISRs). The processor saves the current state before executing the ISR.

1. **What are the** **Advantages of using thumb instructions in ARM?**

Thumb instructions are 16-bit, reducing code size and improving performance in memory-constrained environments.

1. **Write a Code snippet for swapping two numbers without a third variable?**

Assume R0 and R1 hold the two numbers

EOR R0, R0, R1

EOR R1, R0, R1

EOR R0, R0, R1

1. **Define the term"endianess" and its impact on memory storage in ARM?**

Endianess refers to how bytes are ordered in memory (big-endian or little-endian). It impacts how data is interpreted when transferred between systems.

1. **How does the barrel shifter in ARM instruction works:**

The barrel shifter allows shifting or rotating data as part of another instruction, reducing instruction count.

1. **Why pipelining is important in ARM processors?**

Pipelining increases instruction throughput, making the processor more efficient by overlapping instruction execution stages.

1. **Explain How floating-point operations differ from integer operations?**

Floating-point operations involve real numbers and require specialized units for precision, while integer operations handle whole numbers.

1. **What are the Advantages of inline assembly in ARM-based C programming?**

Inline assembly allows developers to write hardware-specific code, optimize performance, and access special processor features directly within C.