SNS COLLEGE OF ENGINEERING
(Autonomous)
DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

Unit 2 ARM Processor and Peripherals

Instructions sets and

INSTRUCTION SETS:

- An **instruction set** in a processor refers to the collection of instructions that the CPU can execute. These instructions define the basic operations that the processor performs, such as arithmetic calculations, data movement, logical operations, and control flow
- The ARM instruction set is 32-bit wide and word-aligned
- Whenever a branch i.e., B instruction is encountered during an ongoing execution then the processor immediately switches to the provided address location and begins to execute the operation from that location.
- This instruction permits forward and backward branches up to 32 MB

Data Processing instructions

The various data processing instructions occur within the general-purpose registers. These instructions include:

1. Arithmetic and logic instructions: These are used to perform various arithmetic and logic operations. Here two source operands are used and the output of the operation is stored within the destination register. The results of arithmetic and logic instructions can be directly written into the Program Counter as it is a general-purpose register.

2. Comparison instructions: The comparison instructions have the same format as that of arithmetic and logic instructions. With this instruction, the operation takes place on two operands but rather than storing the results in registers, the contents of flag registers get updated.

3. Multiply instructions: According to the length of the bits after multiplication, this is divided into two classes, namely,

•32-bit result: It is a normal result and all 32 bits of the result get stored within a register.

•64-bit result: It is considered to be a long result and to store the 64 bit, two separate registers are required.

4. Count leading zero instruction: Through this instruction, the device counts for the total number of zeros from MSB to LSB in the given sequence. The count obtained gets stored in a register.

Load and Store instructions

These instructions are as follows:

1. Load and Store register: Through load register instruction, 8-bit, 16-bit, or 32-bit can be loaded into the register from the memory. While store register instruction transfers the data from the register to memory.

2. Load and Store multiple registers: This instruction permits loading and storing multiple general-purpose registers in block form to or from the memory. Thus, the supportable addressing modes are pre-increment, post-increment, pre-decrement, post-decrement.

3. Swap register and memory content: The swap (SWP) instruction works sequentially in a way that it allows:

•Loading a value from a memory location that is given in the register.

•Further, whatever, content is present within the register is stored in the same memory location.

•Moreover, the value loaded into the memory is also loaded into the register.

Thus, by keeping the register the same for the above two steps, the data existing inside the register and memory location gets interchanged.

- Basic data processing instructions

MOV	Move a 32-bit value	MOV Rd,n	Rd = n
MVN	Move negated (logical NOT) 32-bit value	MVN Rd,n	Rd = ~n
ADD	Add two 32-bit values	ADD Rd,Rn,n	Rd = Rn+n
ADC	Add two 32-bit values and carry	ADC Rd,Rn,n	Rd = Rn+n+C
SUB	Subtract two 32-bit values	SUB Rd,Rn,n	Rd = Rn-n
SBC	Subtract with carry of two 32-bit values	SBC Rd,Rn,n	Rd = Rn-n+C-1
RSB	Reverse subtract of two 32-bit values	RSB Rd,Rn,n	Rd = n-Rn
RSC	Reverse subtract with carry of two 32-bit values	RSC Rd,Rn,n	Rd = n-Rn+C-1
AND	Bitwise AND of two 32-bit values	AND Rd,Rn,n	Rd = Rn AND n
ORR	Bitwise OR of two 32-bit values	ORR Rd,Rn,n	Rd = Rn OR n
EOR	Exclusive OR of two 32-bit values	EOR Rd,Rn,n	Rd = Rn XOR n
BIC	Bit clear. Every '1' in second operand clears corresponding bit of first operand	BIC Rd,Rn,n	Rd = Rn AND (NOT n)
CMP	Compare	CMP Rd,n	Rd-n & change flags only
CMN	Compare Negative	CMN Rd,n	Rd+n & change flags only
TST	Test for a bit in a 32-bit value	TST Rd,n	Rd AND n, change flags
TEQ	Test for equality	TEQ Rd,n	Rd XOR n, change flags

MUL	Multiply two 32-bit values	MUL Rd,Rm,Rs	Rd = Rm*Rs
MLA	Multiple and accumulate	MLA Rd,Rm,Rs,Rn	Rd = (Rm*Rs)+Rn

Arithmetic Operations

* **Operations are:**
  - ADD operand1 + operand2
  - ADC operand1 + operand2 + carry
  - SUB operand1 - operand2
  - SBC operand1 - operand2 + carry -1
  - RSB operand2 - operand1
  - RSC operand2 - operand1 + carry - 1
* **Syntax:**
  - <Operation>{<cond>}{S} Rd, Rn, Operand2
* **Examples**
  - ADD r0, r1, r2
  - SUBGT r3, r3, #1
  - RSBLES r4, r5, #5

LOGICAL INSTRUCTION:

Logical table of and, or, xor, and not gate:

a	b	a AND b	a OR b	a XOR b
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

a	NOT a
0	1
1	0

THANK YOU