5.3. ADDRESSING FORMAT

Addressing modes are various ways of specifying the data. Operands are stored either in the register files or in the memory (on-chip & off-chip).Their addresses are present either directly in the instruction, or indirectly in a CPU register. In either case, the addressing mode should declare if the operand is in the registers or in the memory and provide its address.

(i)Immediate addressing

(ii)Indirect addressing

(iii)Register addressing

(iv)Memory mapped register addressing.

(v)Direct addressing.

(vi)Circular addressing

1. IMMEDIATE ADDRESSING:

Immediate addressing is used to handle constant data .It allows the programmer to operate on an actual value. The data can be either a 16-bit constant or constant length 7.9 or 13. Depending on the length of the data, the addressing mode is referred to as long immediate or short immediate addressing mode .In long immediate addressing the data is contained in apportion of the bits in a single word instruction. At the assembly code level, the developer uses a '#' prefix to specify immediate addressing Example: LD#80h,A : The instruction loads an immediate value 80th in to the accumulator

2. INDIRECT ADDRESSING:

The indirect address mode uses the auxiliary register (ARS) to hold the address of operand in memory. In direct addressing, any location in the 64-k word data memory space can be accessed using a 16- bit address contained in AR. Each auxiliary register (AR0-AR7) provide flexible and powerful indirect addressing .To select a specific auxiliary register , the auxiliary register pointer (arp) is loaded with a value from 0 to 7 for ARO through AR7 respectively . There are seven types of indirect addressing.

1. Auto Increment
2. Auto Decrement

3. Post Indexing By Adding The Contents Of Aro
4. Post Indexing By Subtracting The Contents Of Aro
5. Single Indirect Addressing With No Increment
6. Single Indirect Addressing With No Decrement
7. Bit reversal addressing

3. REGISTER ADDRESSING:

The register addressing mode uses operands in CPU register either explicitly, such as with a direct reference to a specific register, or implicitly, with instruction that intrinsically refers certain registers. That is in this addressing mode the address comes from one of two special purpose memory mapped register in CPU .The block move addresses register (BMAR) and the dynamic bit manipulation register (DBMR). In either case, operand reference is simplified because 16 bit values can be used without specifying a full 16-bit operand address or immediate value.

For example the instruction BLDP, MADD and MADS instruction use the BMAR to address an operand in program memory.

4. MEMORY MAPPED REGISTERS ADDRESSING:

Memory mapped register addressing is used to access efficiently the CPU and on chip peripheral registers. It operates like the direct addressing except that the upper 9-bits of the address that is accessed are assumed to be 0s. This allows us to address the memory mapped register of data page 0 directly without the overhead of changing the DP or auxiliary register. Only the seven lower bits of the complete code, including opcode and operand can be represented using a single 16- bit word.

The following instructions operate in the memory mapped register addressing mode.

LAMM- load accumulator with memory mapped register LMMR-load memory mapped register

SAMM- store accumulator in memory mapped register SMMR-store memory mapped register.

5. DIRECT ADDRESSING MODE:

Direct addressing allows the CPU to access operand by specifying an offset from a base address that is defined in data pointer. DP(data pointer) is a 9- bit field contained in the status register (ST0) .In this mode the address of the operand is obtained by

concatenating the 7- bit data memory address with the 9- bit of the data page pointer . The 16- bit data memory address is placed on an internal direct data memory address bus. Since data pointer is a 9 bit field , it points to one of 512 possible data memory pages and the 7- bit address in the instruction points to one of 128 words within that data memory pages.

6. CIRCULAR ADDRESSING MODE:

Circular addressing is the most sophisticated c5x addressing modes. Any algorithm such as convolution, correlation and FIR filtering can be use circular buffer in memory to implement a sliding window, which contains most recent data to be processed. Five dedicated register are allocated for implementation of circular addressing .The register CBSR1 and CBSR2 are used to load the starting address of circular buffer and the register CBER1 and CBER2 are used to load the end address of circular buffer. The 8- bit CBER enables and disables circular buffer operation. Additionally, one of the auxiliary register (ARS) is used as the pointer in to the circular buffer. To define circular buffer, first we load the start and end addresses in to the corresponding buffer register. Next a value is loaded b/w the start and end register for the circular buffer in to an AR and the corresponding circular buffer enable bit in the CBCR is set. They are

CBSR1- CIRCULAR BUFFER 1 START REGISTER

CBSR2 - CIRCULAR BUFFER 2 START REGISTER.

CBER1- CIRCULAR BUFFER 1 END REGISTER.

CBER2- CIRCULAR BUFFER 2 END REGISTER.

CBCR- CIRCULAR BUFFER CONTROL REGISTER