Microprocessors Lab.

ARYTHMETIC OPERATIONS AND STACK

Arythmetic Operations

- Addition: **ADD**

Adds two arguments:

$$A = A + x$$

The first argument is always the Accumulator register, the second can be one of the following:

- Register (R0 to R7),
- Direct address of a memory,
- Indirect address „pointed" by R0 or R1 (@Ri),
- 8-bit data.

The result is stored in the Accumulator

Arythmetic Operations

- Subtraction: **SUBB**

Subtracts two arguments taking into account the carry flag (C) :

$$A = A - x - C$$

The first argument is always the Accumulator register, the second can be one of the following:

- Register (R0 to R7),
- Direct address of a memory,
- Indirect address „pointed" by R0 or R1 (@Ri),
- 8-bit data.

The result is stored in the Accumulator

Arythmetic Operations

- Multiplication **MUL**

Multiplies values stored in Accumulator and B register. The result can be higher than the 8-bit number range. Therefore, the result is stored in two registers i.e., A (lower 8 bits) and B (higher 8 bits).

Arythmetic Operations

- Division **DIV**

Divides values stored in Accumulator (denominator) and B register (nominator). The result is stored in two registers i.e., A and B . The Accumulator holds the integer part of the result. The B register holds the remainder after division.

Stack – LIFO que

- Stack pointer SP – memory address where the stack begins.
  - Default: SP=07H. The data is stored from address 08H.
  - Changing the stack pointer to memory address 60H:

    MOV SP,#60H ;
  - Storing data on the stack:

    PUSH direct;
  - Retrieving data from the stack:

    POP direct;
  - PUSH and POP operations use direct addressing only

Number input

LCALL WAIT_KEY;

- Waits for a key from the matrix keyboard.
- **This subroutines stores the value of the pressed key into Accumulator and modifies the PSW register.**

Binary-Coded Decimal format

- BCD format
  - class of binary encodings of decimal numbers where each decimal digit is represented by a fixed number of bits
  - unpacked BCD uses a full byte for each digit
  - *packed* BCD encodes two decimal digits within a single byte.
    - Example: 91 encoded in packed BCD - 10010001B = 91H