



Indian Institute of Technology Bombay

Department of Electrical Engineering

EE-309: Microprocessors

Assignment 1

Submission Deadline: 08 October 2018 (Monday) 23:55 Hrs (**Firm**)

Statement: Design a scaled down version of 8085 microprocessor, say Mini-8085 which is suppose to implement the following instructions. Hardware flow chart method is a well-structured method to design microprocessors. Therefore, use hardware flow chart method to design Mini-8085. It should be microcode-based architecture, i.e., use control store (CS) to store encoded control signals. Provide level 2 flow chart, Datapath organization and controller organization including the layout of control store and complete control words along with decode logic.

It should be submitted as *hand written document*.

Instruction Set

1. MOV Rg, Rg
2. MOV Rg, M
3. MOV M, Rg
4. MVI $Rg, D08$ *mov the data in next pc to the reg*
5. LXI $Rp/SP, D16$ *The instruction loads 16-bit data in the register pair designated in the*
6. LDA $D16$ *Load accumulator. (this instruction copies the data from a given 16 bit address to the accumulator)*
7. STA $D16$ *the content of accumulator are copied into the memory location.*
8. ADC Rg *Add register to the accumulator with carry*
9. ACI $D08$ *Add the immediate to the accumulator with carry*
10. SBB Rg *Subtract the source and borrow from the accumulator*
11. ANA Rg *logical AND operation is performed*
12. CMP Rg *compare register with accumulo*

- 13. JMP *D16* unconditionally jump
- 14. JC *D16* conditional jump
- 15. CALL *D16* Text
- 16. CZ *D16* conditional call
- 17. RET Return from subroutine unconditionally
- 18. RZ conditional call
- 19. RST *n* Text

Notations:

Rg – Register (A, B, . . .)

Rp – Register Pair (BC, DE, . . .)

SP – Stack pointer

D08: 8 bit data

D16: 16-bit data/address

M – Memory

n – RST level

Please refer to the following book for the further details of these instructions and encoding of the instructions

**Ramesh Gaonkar, *Microprocessor Architecture, Programming, and Applications with 8085*, PRI
Publisher**