



UNIVERSITY COLLEGE TATI (UC TATI)

FINAL EXAMINATION QUESTION BOOKLET

COURSE CODE	: DEI 2103
COURSE	: FUNDAMENTAL OF MICROCONTROLLER
SEMESTER (SESSION)	: 2 – 2024/2025
DURATION	: 3 HOURS

Instructions:

1. This booklet contains **4** questions. Answer **ALL** .
2. All answers should be written in the answer booklet.
3. Write legibly and draw sketches wherever required.
4. If in doubt, raise your hand and ask the invigilator.

DO NOT OPEN THIS BOOKLET UNTIL YOU ARE TOLD TO DO SO

THIS BOOKLET CONTAINS 10 PRINTED PAGES INCLUDING COVER PAGE

QUESTION 1

- a) List FOUR (4) assembly language instructions that involve with IO programming. (4 marks)
- b) Describe the different between 8031 and 8051 microcontroller family. (3 marks)
- c) Describe the definition for the followings:
 - i. On-chip memory. (3 marks)
 - ii. External RAM. (3 marks)
- d) Describe the difference between Assembly Language and High-Level Language. (3 marks)
- e) Explain the difference between the following two instructions: (4 marks)
 - i. MOV A,40H
 - ii. MOV A,#40H

QUESTION 2

- a) List any FOUR (4) type of addressing modes of 8051 with one example of each. (4 marks)
- b) Name FOUR (4) assembler directive of Assembly Language Instructions. (4 marks)
- c) Describe the meaning of following instructions and its addressing mode. (4 marks)
- i) MOV A, #0F0H
 - ii) SETB P1.0
 - iii) ADD A, R4
 - iv) SWAP A
- d) Explain what is the value in SP registers during power up reset. (3 marks)
- e) Explain the process of assembling and running 8051 programs. (4 marks)
- f) Produce assembly language program to perform BCD addition process of immediate data 29H and 18H respectively. (6 marks)

QUESTION 3

- a) List any THREE (3) unconditional jump instructions. (3 marks)
- b) Describe the definition for the followings:
- i) Define Byte (DB). (2 marks)
 - ii) Define Word (DW). (2 marks)
- c) Describe the definition for the followings:
- i. Unsigned addition. (2 marks)
 - ii. Unsigned subtraction. (2 marks)
 - iii. Signed number. (2 marks)
- d) Write an assembly program for the following steps (Table 1): (6 marks)

Step	Description
1	Move immediate hex 30 into Accumulator A.
2	Shift the contents of accumulator one bit left.
3	Move immediate hex 18 into B register.
4	Multiply content of A by B register.

Table 1

- e) Write and assemble a program subroutine to create a delay about 1.8ms. The crystal frequency is 11.0592 MHz and the machine cycle is 1.085 us. (4 marks)
- f) Simplify program below using call and return instruction by creating subroutine. (3 marks)

```

START:    ORG      0000H
          MOV      R1,#23H
DELAY1:   NOP
          DJNZ     R1,DELAY1
          MOV      R1,#24H

          NOP
          DJNZ     R1,DELAY2
          SJMP     START
          END

```

QUESTION 4

- a) List any FOUR (4) logical instructions. (4 marks)
- b) If initial content of accumulator is 44 H, find out the new content of accumulator after execution of the instruction RR A. (2 marks)
- c) Write a program to get data from P1 then send it to P0, P2 and P3 respectively. (5 marks)
- d) Explain the operation of the following program; (6 marks)
- ```
ORG 0000h
MOV 20H,#55H
MOV 30H,#11H
MOV R0,#20H
MOV R1,#30H

MOV A,@R0
ADD A,@R1
MOV P1,A
SETB P1.0
NOP
END
```
- e) Produce assembly language program to multiply two 8-bit numbers stored at 26h and 27h in internal RAM. Store result in 22h and 23h. (6 marks)
- f) Develop assembly language program for 8051 to perform the conversion of an 8-bit BCD number of 34hex to two ASCII. Store result in R0 and R1. (6 marks)

-----End of question-----

## FUNDAMENTAL OF MICROCONTROLLER (DEI2103)

## APPENDICE

## 8051 Microcontroller Instruction Set

Instruction Opcodes in Hexadecimal Order

| Hex Code | Number of Bytes | Mnemonic | Operands            |
|----------|-----------------|----------|---------------------|
| 00       | 1               | NOP      |                     |
| 01       | 2               | AJMP     | code addr           |
| 02       | 3               | LJMP     | code addr           |
| 03       | 1               | RR       | A                   |
| 04       | 1               | INC      | A                   |
| 05       | 2               | INC      | data addr           |
| 06       | 1               | INC      | @R0                 |
| 07       | 1               | INC      | @R1                 |
| 08       | 1               | INC      | R0                  |
| 09       | 1               | INC      | R1                  |
| 0A       | 1               | INC      | R2                  |
| 0B       | 1               | INC      | R3                  |
| 0C       | 1               | INC      | R4                  |
| 0D       | 1               | INC      | R5                  |
| 0E       | 1               | INC      | R6                  |
| 0F       | 1               | INC      | R7                  |
| 10       | 3               | JBC      | bit addr, code addr |
| 11       | 2               | ACALL    | code addr           |
| 12       | 3               | LCALL    | code addr           |
| 13       | 1               | RRC      | A                   |
| 14       | 1               | DEC      | A                   |
| 15       | 2               | DEC      | data addr           |
| 16       | 1               | DEC      | @R0                 |
| 17       | 1               | DEC      | @R1                 |
| 18       | 1               | DEC      | R0                  |
| 19       | 1               | DEC      | R1                  |
| 1A       | 1               | DEC      | R2                  |
| 1B       | 1               | DEC      | R3                  |
| 1C       | 1               | DEC      | R4                  |
| 1D       | 1               | DEC      | R5                  |
| 1E       | 1               | DEC      | R6                  |
| 1F       | 1               | DEC      | R7                  |
| 20       | 3               | JB       | bit addr, code addr |
| 21       | 2               | AJMP     | code addr           |
| 22       | 1               | RET      |                     |
| 23       | 1               | RL       | A                   |
| 24       | 2               | ADD      | A, #data            |
| 25       | 2               | ADD      | A, data addr        |

| Hex Code | Number of Bytes | Mnemonic | Operands            |
|----------|-----------------|----------|---------------------|
| 26       | 1               | ADD      | A, @R0              |
| 27       | 1               | ADD      | A, @R1              |
| 28       | 1               | ADD      | A, R0               |
| 29       | 1               | ADD      | A, R1               |
| 2A       | 1               | ADD      | A, R2               |
| 2B       | 1               | ADD      | A, R3               |
| 2C       | 1               | ADD      | A, R4               |
| 2D       | 1               | ADD      | A, R5               |
| 2E       | 1               | ADD      | A, R6               |
| 2F       | 1               | ADD      | A, R7               |
| 30       | 3               | JNB      | bit addr, code addr |
| 31       | 2               | ACALL    | code addr           |
| 32       | 1               | RETI     |                     |
| 33       | 1               | RLC      | A                   |
| 34       | 2               | ADDC     | A, #data            |
| 35       | 2               | ADDC     | A, data addr        |
| 36       | 1               | ADDC     | A, @R0              |
| 37       | 1               | ADDC     | A, @R1              |
| 38       | 1               | ADDC     | A, R0               |
| 39       | 1               | ADDC     | A, R1               |
| 3A       | 1               | ADDC     | A, R2               |
| 3B       | 1               | ADDC     | A, R3               |
| 3C       | 1               | ADDC     | A, R4               |
| 3D       | 1               | ADDC     | A, R5               |
| 3E       | 1               | ADDC     | A, R6               |
| 3F       | 1               | ADDC     | A, R7               |
| 40       | 2               | JC       | code addr           |
| 41       | 2               | AJMP     | code addr           |
| 42       | 2               | ORL      | data addr, A        |
| 43       | 3               | ORL      | data addr, #data    |
| 44       | 2               | ORL      | A, #data            |
| 45       | 2               | ORL      | A, data addr        |
| 46       | 1               | ORL      | A, @R0              |
| 47       | 1               | ORL      | A, @R1              |
| 48       | 1               | ORL      | A, R0               |
| 49       | 1               | ORL      | A, R1               |
| 4A       | 1               | ORL      | A, R2               |

## FUNDAMENTAL OF MICROCONTROLLER (DEI2103)

| Hex Code | Number of Bytes | Mnemonic | Operands        |
|----------|-----------------|----------|-----------------|
| 4B       | 1               | ORL      | A,R3            |
| 4C       | 1               | ORL      | A,R4            |
| 4D       | 1               | ORL      | A,R5            |
| 4E       | 1               | ORL      | A,R6            |
| 4F       | 1               | ORL      | A,R7            |
| 50       | 2               | JNC      | code addr       |
| 51       | 2               | ACALL    | code addr       |
| 52       | 2               | ANL      | data addr,A     |
| 53       | 3               | ANL      | data addr,#data |
| 54       | 2               | ANL      | A,#data         |
| 55       | 2               | ANL      | A,data addr     |
| 56       | 1               | ANL      | A,@R0           |
| 57       | 1               | ANL      | A,@R1           |
| 58       | 1               | ANL      | A,R0            |
| 59       | 1               | ANL      | A,R1            |
| 5A       | 1               | ANL      | A,R2            |
| 5B       | 1               | ANL      | A,R3            |
| 5C       | 1               | ANL      | A,R4            |
| 5D       | 1               | ANL      | A,R5            |
| 5E       | 1               | ANL      | A,R6            |
| 5F       | 1               | ANL      | A,R7            |
| 60       | 2               | JZ       | code addr       |
| 61       | 2               | AJMP     | code addr       |
| 62       | 2               | XRL      | data addr,A     |
| 63       | 3               | XRL      | data addr,#data |
| 64       | 2               | XRL      | A,#data         |
| 65       | 2               | XRL      | A,data addr     |
| 66       | 1               | XRL      | A,@R0           |
| 67       | 1               | XRL      | A,@R1           |
| 68       | 1               | XRL      | A,R0            |
| 69       | 1               | XRL      | A,R1            |
| 6A       | 1               | XRL      | A,R2            |
| 6B       | 1               | XRL      | A,R3            |
| 6C       | 1               | XRL      | A,R4            |
| 6D       | 1               | XRL      | A,R5            |
| 6E       | 1               | XRL      | A,R6            |
| 6F       | 1               | XRL      | A,R7            |
| 70       | 2               | JNZ      | code addr       |

| Hex Code | Number of Bytes | Mnemonic | Operands            |
|----------|-----------------|----------|---------------------|
| 71       | 2               | ACALL    | code addr           |
| 72       | 2               | ORL      | C,bit addr          |
| 73       | 1               | JMP      | @A+DPTR             |
| 74       | 2               | MOV      | A,#data             |
| 75       | 3               | MOV      | data addr,#data     |
| 76       | 2               | MOV      | @R0,#data           |
| 77       | 2               | MOV      | @R1,#data           |
| 78       | 2               | MOV      | R0,#data            |
| 79       | 2               | MOV      | R1,#data            |
| 7A       | 2               | MOV      | R2,#data            |
| 7B       | 2               | MOV      | R3,#data            |
| 7C       | 2               | MOV      | R4,#data            |
| 7D       | 2               | MOV      | R5,#data            |
| 7E       | 2               | MOV      | R6,#data            |
| 7F       | 2               | MOV      | R7,#data            |
| 80       | 2               | SJMP     | code addr           |
| 81       | 2               | AJMP     | code addr           |
| 82       | 2               | ANL      | C,bit addr          |
| 83       | 1               | MOVC     | A,@A+PC             |
| 84       | 1               | DIV      | AB                  |
| 85       | 3               | MOV      | data addr,data addr |
| 86       | 2               | MOV      | data addr,@R0       |
| 87       | 2               | MOV      | data addr,@R1       |
| 88       | 2               | MOV      | data addr,R0        |
| 89       | 2               | MOV      | data addr,R1        |
| 8A       | 2               | MOV      | data addr,R2        |
| 8B       | 2               | MOV      | data addr,R3        |
| 8C       | 2               | MOV      | data addr,R4        |
| 8D       | 2               | MOV      | data addr,R5        |
| 8E       | 2               | MOV      | data addr,R6        |
| 8F       | 2               | MOV      | data addr,R7        |
| 90       | 3               | MOV      | DPTR,#data          |
| 91       | 2               | ACALL    | code addr           |
| 92       | 2               | MOV      | bit addr,C          |
| 93       | 1               | MOVC     | A,@A+DPTR           |
| 94       | 2               | SUBB     | A,#data             |
| 95       | 2               | SUBB     | A,data addr         |
| 96       | 1               | SUBB     | A,@R0               |

## FUNDAMENTAL OF MICROCONTROLLER (DEI2103)

*8051 Microcontroller Instruction Set*

| Hex Code | Number of Bytes | Mnemonic | Operands              |
|----------|-----------------|----------|-----------------------|
| 97       | 1               | SUBB     | A,@R1                 |
| 98       | 1               | SUBB     | A,R0                  |
| 99       | 1               | SUBB     | A,R1                  |
| 9A       | 1               | SUBB     | A,R2                  |
| 9B       | 1               | SUBB     | A,R3                  |
| 9C       | 1               | SUBB     | A,R4                  |
| 9D       | 1               | SUBB     | A,R5                  |
| 9E       | 1               | SUBB     | A,R6                  |
| 9F       | 1               | SUBB     | A,R7                  |
| A0       | 2               | ORL      | C,bit addr            |
| A1       | 2               | AJMP     | code addr             |
| A2       | 2               | MOV      | C,bit addr            |
| A3       | 1               | INC      | DPTR                  |
| A4       | 1               | MUL      | AB                    |
| A5       |                 | reserved |                       |
| A6       | 2               | MOV      | @R0,data addr         |
| A7       | 2               | MOV      | @R1,data addr         |
| A8       | 2               | MOV      | R0,data addr          |
| A9       | 2               | MOV      | R1,data addr          |
| AA       | 2               | MOV      | R2,data addr          |
| AB       | 2               | MOV      | R3,data addr          |
| AC       | 2               | MOV      | R4,data addr          |
| AD       | 2               | MOV      | R5,data addr          |
| AE       | 2               | MOV      | R6,data addr          |
| AF       | 2               | MOV      | R7,data addr          |
| B0       | 2               | ANL      | C,bit addr            |
| B1       | 2               | ACALL    | code addr             |
| B2       | 2               | CPL      | bit addr              |
| B3       | 1               | CPL      | C                     |
| B4       | 3               | CJNE     | A,#data,code addr     |
| B5       | 3               | CJNE     | A,data addr,code addr |
| B6       | 3               | CJNE     | @R0,#data,code addr   |
| B7       | 3               | CJNE     | @R1,#data,code addr   |
| B8       | 3               | CJNE     | R0,#data,code addr    |
| B9       | 3               | CJNE     | R1,#data,code addr    |
| BA       | 3               | CJNE     | R2,#data,code addr    |
| BB       | 3               | CJNE     | R3,#data,code addr    |
| BC       | 3               | CJNE     | R4,#data,code addr    |

| Hex Code | Number of Bytes | Mnemonic | Operands            |
|----------|-----------------|----------|---------------------|
| BD       | 3               | CJNE     | R5,#data,code addr  |
| BE       | 3               | CJNE     | R6,#data,code addr  |
| BF       | 3               | CJNE     | R7,#data,code addr  |
| C0       | 2               | PUSH     | data addr           |
| C1       | 2               | AJMP     | code addr           |
| C2       | 2               | CLR      | bit addr            |
| C3       | 1               | CLR      | C                   |
| C4       | 1               | SWAP     | A                   |
| C5       | 2               | XCH      | A,data addr         |
| C6       | 1               | XCH      | A,@R0               |
| C7       | 1               | XCH      | A,@R1               |
| C8       | 1               | XCH      | A,R0                |
| C9       | 1               | XCH      | A,R1                |
| CA       | 1               | XCH      | A,R2                |
| CB       | 1               | XCH      | A,R3                |
| CC       | 1               | XCH      | A,R4                |
| CD       | 1               | XCH      | A,R5                |
| CE       | 1               | XCH      | A,R6                |
| CF       | 1               | XCH      | A,R7                |
| D0       | 2               | POP      | data addr           |
| D1       | 2               | ACALL    | code addr           |
| D2       | 2               | SETB     | bit addr            |
| D3       | 1               | SETB     | C                   |
| D4       | 1               | DA       | A                   |
| D5       | 3               | DJNZ     | data addr,code addr |
| D6       | 1               | XCHD     | A,@R0               |
| D7       | 1               | XCHD     | A,@R1               |
| D8       | 2               | DJNZ     | R0,code addr        |
| D9       | 2               | DJNZ     | R1,code addr        |
| DA       | 2               | DJNZ     | R2,code addr        |
| DB       | 2               | DJNZ     | R3,code addr        |
| DC       | 2               | DJNZ     | R4,code addr        |
| DD       | 2               | DJNZ     | R5,code addr        |
| DE       | 2               | DJNZ     | R6,code addr        |
| DF       | 2               | DJNZ     | R7,code addr        |
| E0       | 1               | MOVX     | A,@DPTR             |
| E1       | 2               | AJMP     | code addr           |
| E2       | 1               | MOVX     | A,@R0               |

## FUNDAMENTAL OF MICROCONTROLLER (DEI2103)

| Hex Code | Number of Bytes | Mnemonic | Operands    |
|----------|-----------------|----------|-------------|
| E3       | 1               | MOVX     | A,@R1       |
| E4       | 1               | CLR      | A           |
| E5       | 2               | MOV      | A,data addr |
| E6       | 1               | MOV      | A,@R0       |
| E7       | 1               | MOV      | A,@R1       |
| E8       | 1               | MOV      | A,R0        |
| E9       | 1               | MOV      | A,R1        |
| EA       | 1               | MOV      | A,R2        |
| EB       | 1               | MOV      | A,R3        |
| EC       | 1               | MOV      | A,R4        |
| ED       | 1               | MOV      | A,R5        |
| EE       | 1               | MOV      | A,R6        |
| EF       | 1               | MOV      | A,R7        |
| F0       | 1               | MOVX     | @DPTR,A     |
| F1       | 2               | ACALL    | code addr   |
| F2       | 1               | MOVX     | @R0,A       |
| F3       | 1               | MOVX     | @R1,A       |
| F4       | 1               | CPL      | A           |
| F5       | 2               | MOV      | data addr,A |
| F6       | 1               | MOV      | @R0,A       |
| F7       | 1               | MOV      | @R1,A       |
| F8       | 1               | MOV      | R0,A        |
| F9       | 1               | MOV      | R1,A        |
| FA       | 1               | MOV      | R2,A        |
| FB       | 1               | MOV      | R3,A        |
| FC       | 1               | MOV      | R4,A        |
| FD       | 1               | MOV      | R5,A        |
| FE       | 1               | MOV      | R6,A        |
| FF       | 1               | MOV      | R7,A        |

## FUNDAMENTAL OF MICROCONTROLLER (DEI2103)

# ASCII TABLE

| Decimal | Hex | Char                   | Decimal | Hex | Char    | Decimal | Hex | Char | Decimal | Hex | Char  |
|---------|-----|------------------------|---------|-----|---------|---------|-----|------|---------|-----|-------|
| 0       | 0   | [NULL]                 | 32      | 20  | [SPACE] | 64      | 40  | @    | 96      | 60  | `     |
| 1       | 1   | [START OF HEADING]     | 33      | 21  | !       | 65      | 41  | A    | 97      | 61  | a     |
| 2       | 2   | [START OF TEXT]        | 34      | 22  | "       | 66      | 42  | B    | 98      | 62  | b     |
| 3       | 3   | [END OF TEXT]          | 35      | 23  | #       | 67      | 43  | C    | 99      | 63  | c     |
| 4       | 4   | [END OF TRANSMISSION]  | 36      | 24  | \$      | 68      | 44  | D    | 100     | 64  | d     |
| 5       | 5   | [ENQUIRY]              | 37      | 25  | %       | 69      | 45  | E    | 101     | 65  | e     |
| 6       | 6   | [ACKNOWLEDGE]          | 38      | 26  | &       | 70      | 46  | F    | 102     | 66  | f     |
| 7       | 7   | [BELL]                 | 39      | 27  | '       | 71      | 47  | G    | 103     | 67  | g     |
| 8       | 8   | [BACKSPACE]            | 40      | 28  | (       | 72      | 48  | H    | 104     | 68  | h     |
| 9       | 9   | [HORIZONTAL TAB]       | 41      | 29  | )       | 73      | 49  | I    | 105     | 69  | i     |
| 10      | A   | [LINE FEED]            | 42      | 2A  | *       | 74      | 4A  | J    | 106     | 6A  | j     |
| 11      | B   | [VERTICAL TAB]         | 43      | 2B  | +       | 75      | 4B  | K    | 107     | 6B  | k     |
| 12      | C   | [FORM FEED]            | 44      | 2C  | ,       | 76      | 4C  | L    | 108     | 6C  | l     |
| 13      | D   | [CARRIAGE RETURN]      | 45      | 2D  | -       | 77      | 4D  | M    | 109     | 6D  | m     |
| 14      | E   | [SHIFT OUT]            | 46      | 2E  | .       | 78      | 4E  | N    | 110     | 6E  | n     |
| 15      | F   | [SHIFT IN]             | 47      | 2F  | /       | 79      | 4F  | O    | 111     | 6F  | o     |
| 16      | 10  | [DATA LINK ESCAPE]     | 48      | 30  | 0       | 80      | 50  | P    | 112     | 70  | p     |
| 17      | 11  | [DEVICE CONTROL 1]     | 49      | 31  | 1       | 81      | 51  | Q    | 113     | 71  | q     |
| 18      | 12  | [DEVICE CONTROL 2]     | 50      | 32  | 2       | 82      | 52  | R    | 114     | 72  | r     |
| 19      | 13  | [DEVICE CONTROL 3]     | 51      | 33  | 3       | 83      | 53  | S    | 115     | 73  | s     |
| 20      | 14  | [DEVICE CONTROL 4]     | 52      | 34  | 4       | 84      | 54  | T    | 116     | 74  | t     |
| 21      | 15  | [NEGATIVE ACKNOWLEDGE] | 53      | 35  | 5       | 85      | 55  | U    | 117     | 75  | u     |
| 22      | 16  | [SYNCHRONOUS IDLE]     | 54      | 36  | 6       | 86      | 56  | V    | 118     | 76  | v     |
| 23      | 17  | [END OF TRANS. BLOCK]  | 55      | 37  | 7       | 87      | 57  | W    | 119     | 77  | w     |
| 24      | 18  | [CANCEL]               | 56      | 38  | 8       | 88      | 58  | X    | 120     | 78  | x     |
| 25      | 19  | [END OF MEDIUM]        | 57      | 39  | 9       | 89      | 59  | Y    | 121     | 79  | y     |
| 26      | 1A  | [SUBSTITUTE]           | 58      | 3A  | :       | 90      | 5A  | Z    | 122     | 7A  | z     |
| 27      | 1B  | [ESCAPE]               | 59      | 3B  | ;       | 91      | 5B  | [    | 123     | 7B  | {     |
| 28      | 1C  | [FILE SEPARATOR]       | 60      | 3C  | <       | 92      | 5C  | \    | 124     | 7C  |       |
| 29      | 1D  | [GROUP SEPARATOR]      | 61      | 3D  | =       | 93      | 5D  | ]    | 125     | 7D  | }     |
| 30      | 1E  | [RECORD SEPARATOR]     | 62      | 3E  | >       | 94      | 5E  | ^    | 126     | 7E  | ~     |
| 31      | 1F  | [UNIT SEPARATOR]       | 63      | 3F  | ?       | 95      | 5F  | _    | 127     | 7F  | [DEL] |