2507/302
MICROCONTROLLER TECHNOLOGY
June/July 2018
Time: 3 hours

THE KENYA NATIONAL EXAMINATIONS COUNCIL

DIPLOMA IN AERONAUTICAL ENGINEERING (AVIONICS OPTION)

MODULE III

MICROCONTROLLER TECHNOLOGY

3 hours

INSTRUCTIONS TO CANDIDATES

You should have the following for this examination:
Answer booklet;
Non-programmable scientific calculator;
Intel 8051 Microcontroller instrument set.
This paper consists of ***EIGHT*** *questions.*
Answer any ***FIVE*** *of the* ***EIGHT*** *questions in the answer booklet provided.*
All questions carry equal marks.
Maximum marks for each part of a question are as shown.
Candidates should answer the questions in English.

This paper consists of 9 printed pages.

Candidates should check the question paper to ascertain that all the pages are printed as indicated and that no questions are missing.

Turn over

1. (a) Perform the following:

(i) $(1110\ \ 1010\ \ 0111\ \ 1111)_2$ into hexadecimal;

(ii) $(11\ 0011)_2$ into decimal;

(iii) $(1010111.101)_2$ into octal;

(iv) -27_{10} into 8 - bits two's compliment. (8 marks)

(b) Perform the following operations:

(i) $(1101)_2 \times (111)_2$;

(ii) $(1001)_2 \div (10)_2$;

(iii) $23\,D_{16} - IF\,E_{16}$. (6 marks)

(c) Draw a labelled flowchart of micro-controller instruction cycle. (6 marks)

2. (a) Draw programmable logic controller (PLC) symbols for each of the following components:

(i) normally closed contact;

(ii) output coil. (2 marks)

(b) Draw a labelled block diagram of a PLC and state the function of each block. (9 marks)

Power

CPU

Input

Memory

output

data P

Program device

(c) A digital control system has two inputs X and Y and three outputs P, Q and R. The relationship between the inputs and outputs are as follows:

Output P indicates the absence of both inputs X and Y;
Output Q indicates the presence of either input;
Output R indicates the presence of both inputs.

(i) Draw a truth table to represent these functions;

(ii) Write the Boolean expressions for the functions;

P x+y = Q x×y = R

(iii) Write instruction list program for output Q. (9 marks)

0 0 P
0 1 = Q
1 0 = Q
1 1 — R

3. (a) Define each of the following terms as used in process control:

(i) dead time;

(ii) lag compensation;

(iii) transient response. (3 marks)

(b) Describe the function(s) of the following Supervisory Control and Data Acquisition (SCADA) components:

(i) Supervisory computes;

(ii) Remote Terminal Units (R.T.U); ->

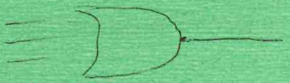
(iii) Human - Machine Interface (H.M.I); -> Display ✓

(iv) Actuator; (8 marks)

(c) Draw ladder programs for the following:

(i) A lamp is switched ON if there is an input from sensor A or sensor B or sensor C. ✓

(ii) A battery back-up relay to maintain an output ON, even if the input ceases.



(9 marks)

4. (a) State **four** features of the 8051 microcontroller. (4 marks)

(b) With the aid of a one line instruction, describe the following addressing modes in 8051 microcontroller:

(i) direct;

(ii) register indirect;

(iii) index register. (9 marks)

(c) Write a microcontroller assembly program to subtract 8 - bit numbers 3B H from 4F H and exchange the nibbles and multiply the result by 14 H. (7 marks)

5. (a) With an aid of an example, describe three microcontroller instruction groups. (9 marks)

Data handling

Alu ->

Control flow ->

Compressio

(b) For each of the following 8051 microcontroller pins, state their function:

(i) $\overline{RD}$;

(ii) $\overline{WR}$; - write

(iii) $\overline{INT1}$;

(iv) TXD; - Transmit.

(v) RXT. - Re (5 marks)

(c) Draw the memory maps for the intel 8051 microcontroller and describe each memory type. (6 marks)

6. (a) Describe each of the modes of 8051 Timer operations:

(i) mode 0;

(ii) mode 1;

(iii) mode 2;

(iv) mode 3. (4 marks)

(b) Table 2 shows intel 8051 microcontroller program segment:

Table 2

Instruction
MOV 45 H; # 20 H MOV A, # 45 H MOV R O, A MOV A, @ RO MOV R1 # 09 H ADD A, R 1

(i) Draw its trace table;

(ii) Determine the total number of bytes for the machine program. (10 marks)

(c) Write an assembly program to generate a BCD up counter and send each count to port 1. (6 marks)

7. (a) State:

(i) **three** merits of using robots in industries;

(ii) **three** types of sensors used in robotics. ✓ (6 marks)

(b) A 12 – bit BCD digital - to - analog converter has a step - size of 12 mV. Determine its:

(i) full scale output;

(ii) percentage resolution. (6 marks)

(c) Figure 1 shows an 8051 microcontroller connected to a digital ramp Analogue to Digital Converter (ADC) for data acquisition.

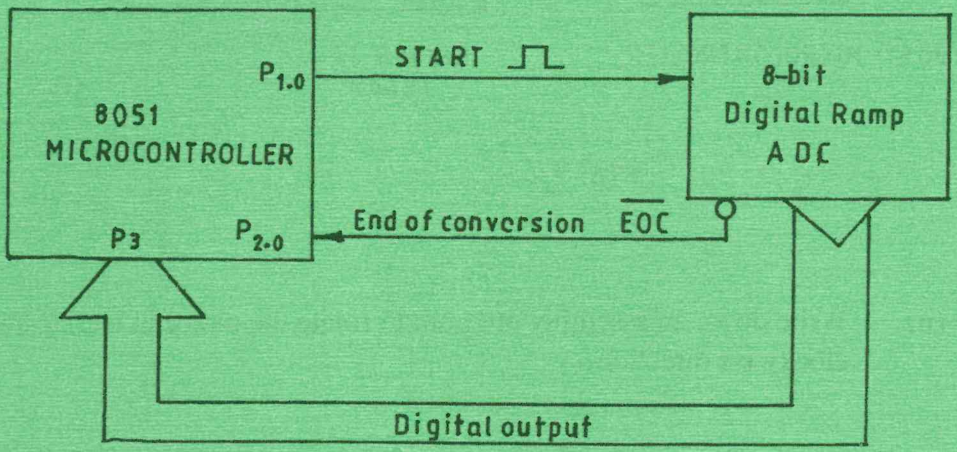


Fig. 1

Write an assembly language program to control the ADC. (8 marks)

8. (a) Describe the function of the following microcontroller software development tools:

(i) assembler;

(ii) compiler;

(iii) linker;

(iv) editor. (8 marks)

(b) (i) Figure 2 shows a stepper motor interfaced to an intel 8051 microcontroller. Describe how the motor is controlled to step in clockwise direction.

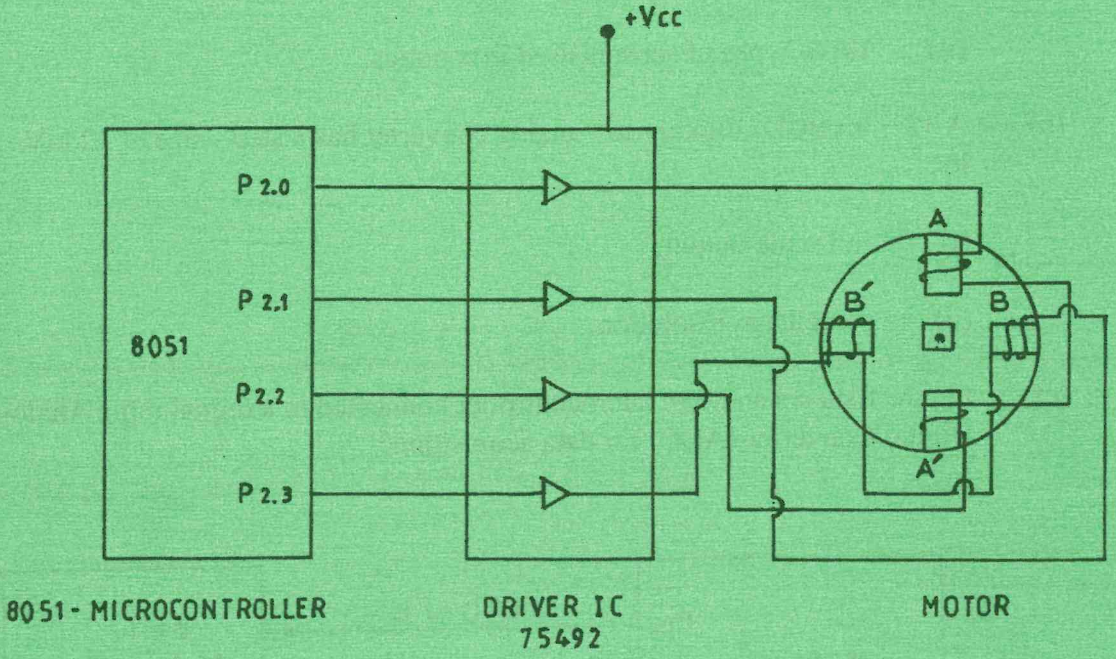


Fig. 2

(ii) Write down an assembly program to rotate the motor in b (i) 6 steps in clockwise direction. (12 marks)

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
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
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

8051 OpCodes en Hexadecinal.

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
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
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
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
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

Instruction Opcodes in Hexadecimal Order

THIS IS THE LAST PRINTED PAGE.