

Name: Varun Agarwal

Registration Number: 16BEC0450

ECE3003- Microcontroller and its Applications

Lab Slot: L37+L38

Prof. Chitra P

TASK II

Program 1

Write an 8051 ALP for the following

55H AND AAH

Code:

ORG 0000H

XX:MOV A,#55H

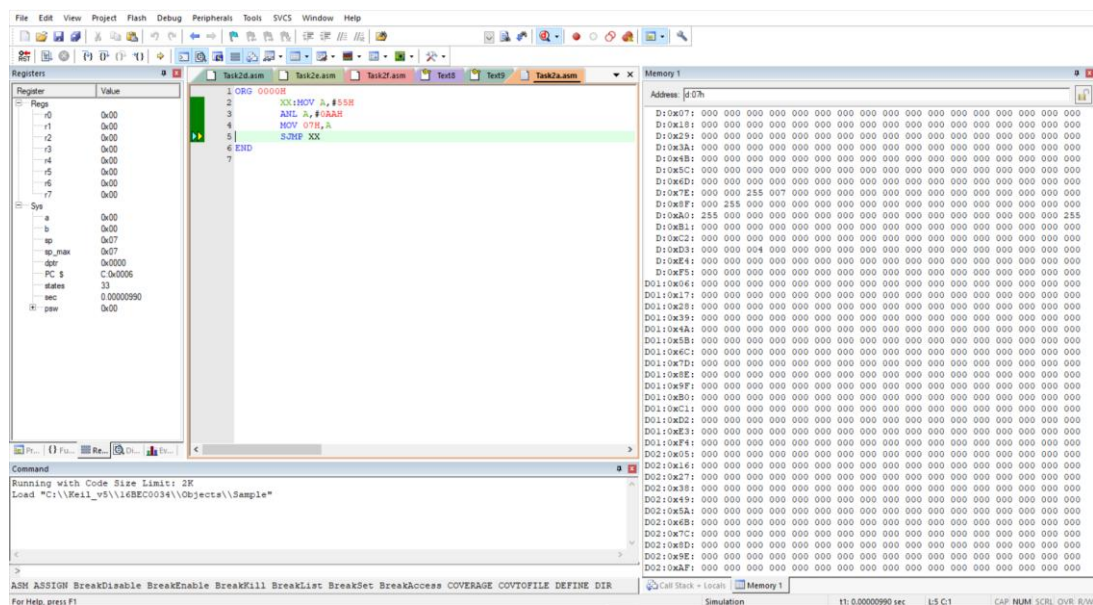
ANL A,#0AAH

MOV 07H,A

SJMP XX

END

OUTPUT :



```
MOV A,#55H
CPL A
ANL A,#0AAH
MOV 07H,A
CLR A
MOV A,#0AAH
CPL A
ANL A,#55H
ORL A,07H
MOV 09H,A
```

END

OUTPUT:

The screenshot displays the Keil uVision IDE interface. The main window shows an assembly program named 'Task2.asm' with the following code:

```
1 ORG 0000H
2 MOV A, #55H
3 CPL A
4 ANL A, #0AAH
5 MOV 07H, A
6 CLR A
7 MOV A, #0AAH
8 CPL A
9 ANL A, #55H
10 ORL A, 07H
11 MOV 09H, A
12 END
13
```

The left pane shows the 'Registers' window with values for R0 through R7, and system registers like SP, PC, and PSW. The right pane shows the 'Memory' window with a list of memory addresses and their contents, mostly zeros. The bottom pane shows the 'Command' window with error messages related to access violations at C:\0x0011.

Program 2

- Write an 8051 ALP for the following TRUTH TABLE

AND LOGIC		
INPUT		OUTPUT
A	B	Y = A.B
0	0	
0	1	
1	0	
1	1	

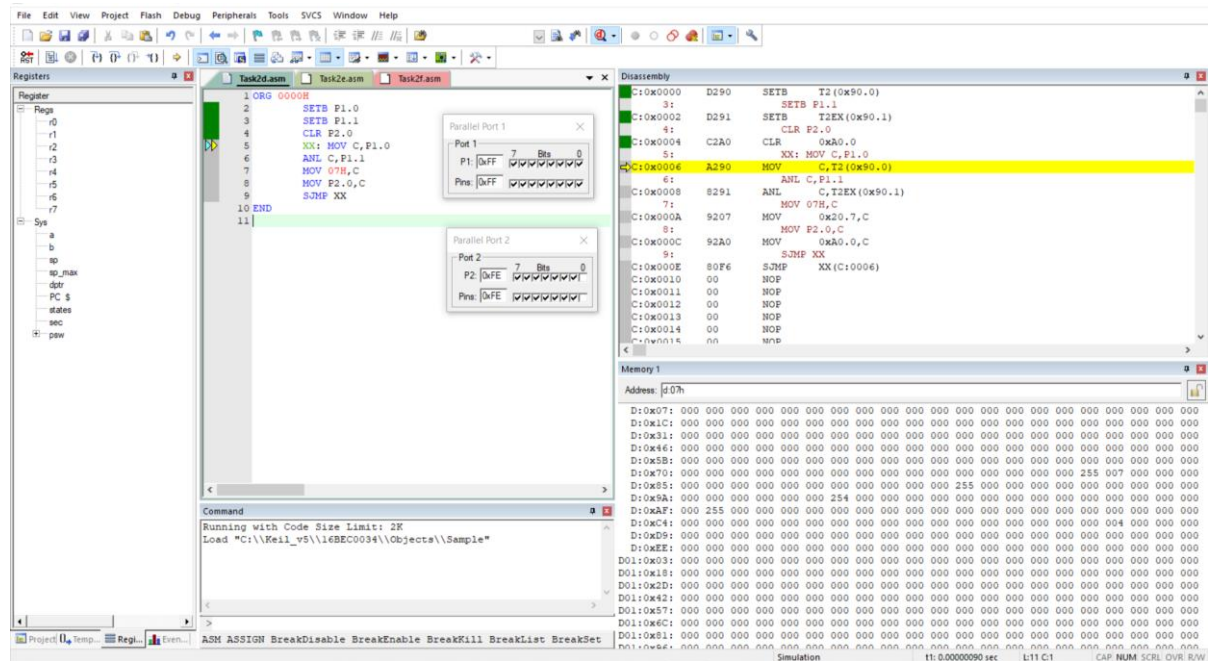
OR LOGIC		
INPUT		OUTPUT
A	B	Y = A+B
0	0	
0	1	
1	0	
1	1	

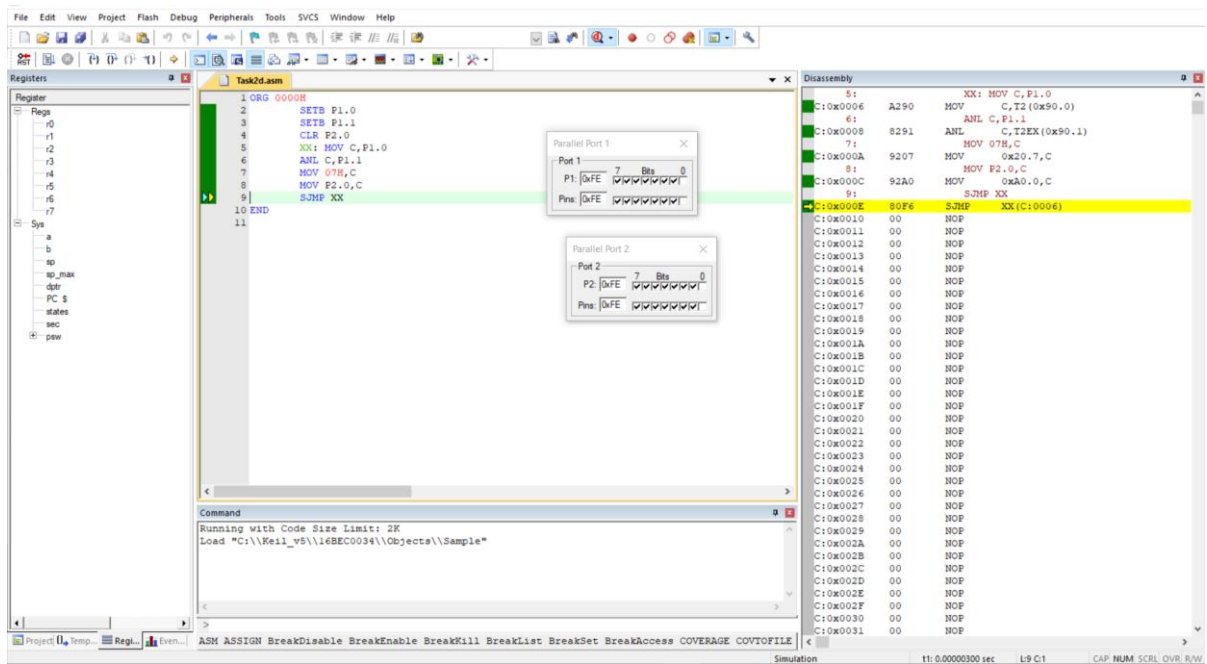
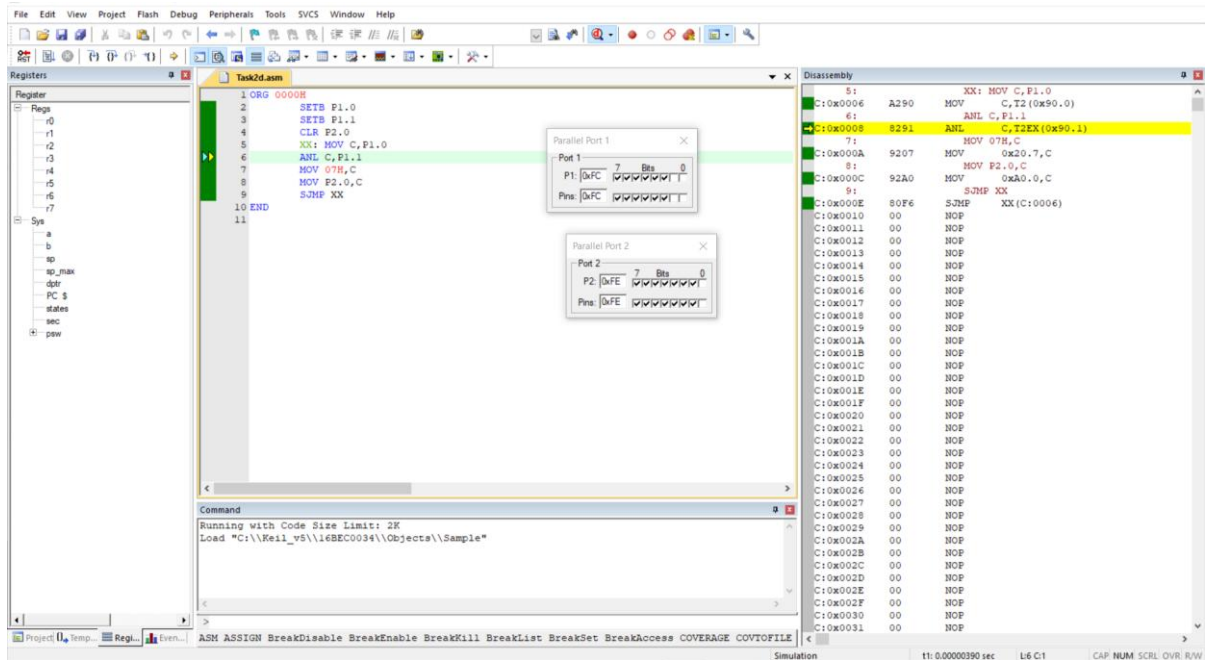
XOR LOGIC		
INPUT		OUTPUT
A	B	Y = A⊕B
0	0	
0	1	
1	0	
1	1	

AND Logic

Code & Output:

```
ORG 0000H  
  
    SETB P1.0  
    SETB P1.1  
    CLR P2.0  
    XX: MOV C,P1.0  
    ANL C,P1.1  
    MOV 07H,C  
    MOV P2.0,C  
    SJMP XX  
  
END
```





OR Logic

Code:

ORG 0000H

SETB P1.0

SETB P1.1

CLR P2.0

XX: MOV C,P1.0

ORL C,P1.1

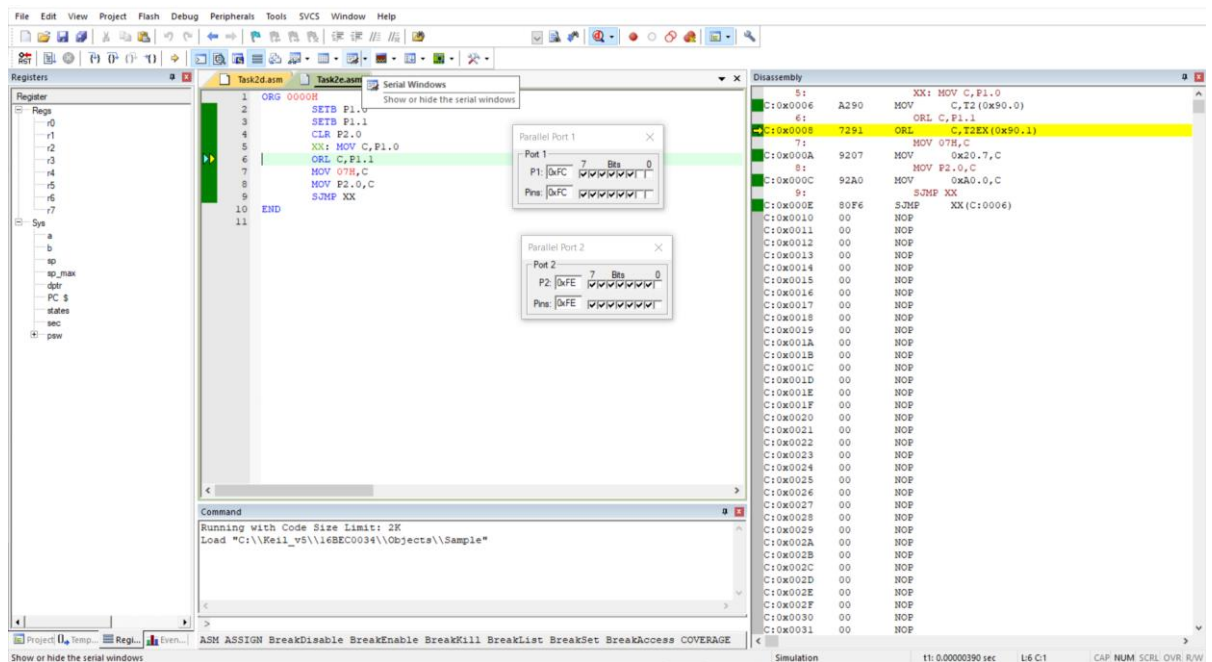
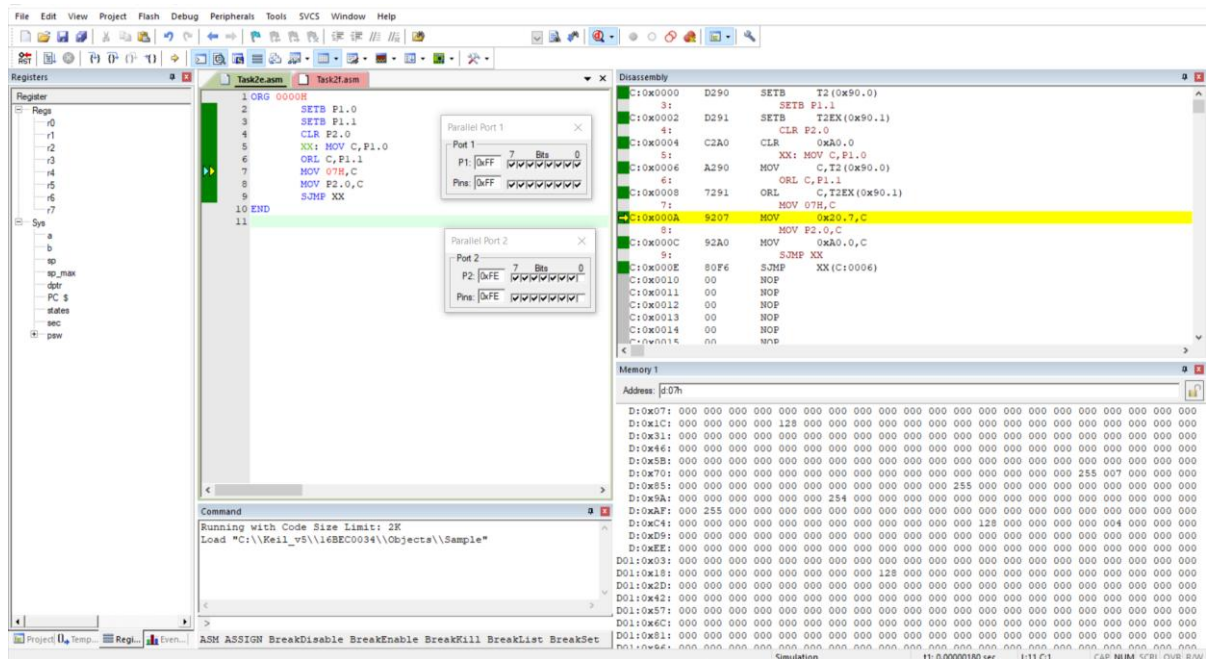
MOV 07H,C

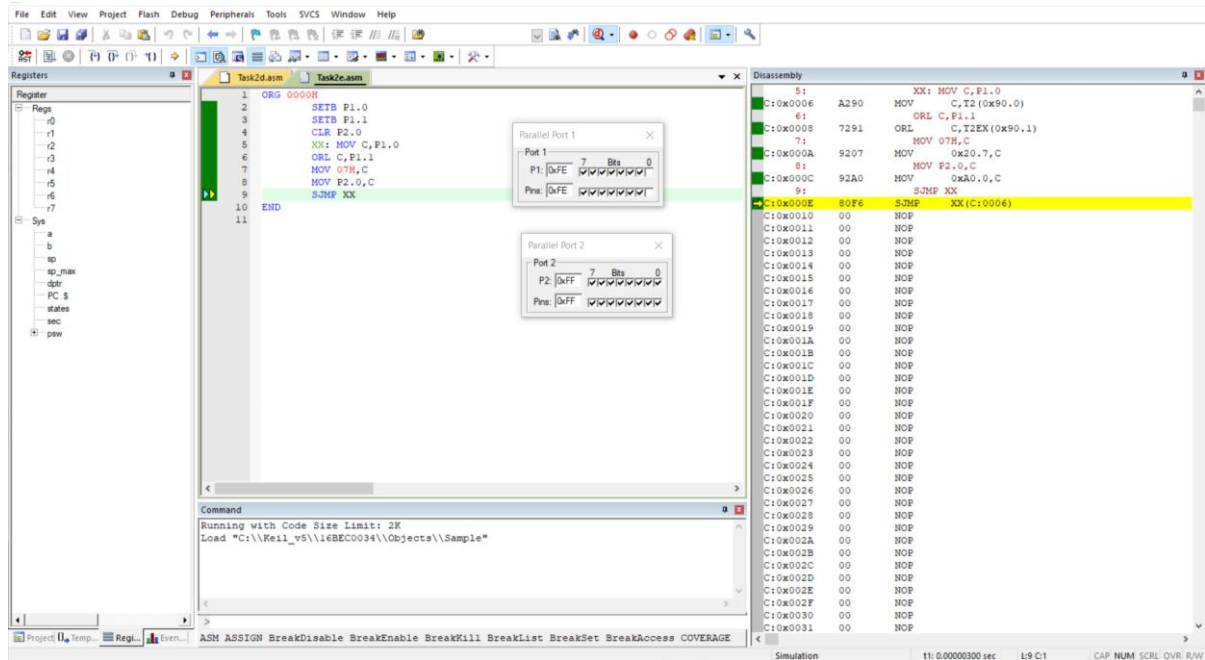
MOV P2.0,C

SJMP XX

END

OUTPUT:





XOR Logic

Code :

```

ORG 0000H
    SETB P0.0
    SETB P0.1
    CLR P1.0
    XX: MOV C,P0.0
    CPL C
    ANL C,P0.1
    MOV 09H,C
    MOV C,P0.1
    CPL C
    ANL C,P0.0
    ORL C,09H
    MOV P1.0,C
    SJMP XX
END
  
```

OUTPUT :

The screenshot shows the AVR Studio IDE with the following components:

- Registers:** A list of registers (r0-r7, sp, sp_max, dptr, PC, S, states, sec, paw) is shown on the left.
- Task2.asm:** The assembly code is displayed in the main window. The code includes instructions like `ORG 0000H`, `SETB P0.0`, `SETB P0.1`, `CLR P1.0`, `XX: MOV C,P0.0`, `CPL C`, `ANL C,P0.1`, `MOV 09H,C`, `MOV C,P0.1`, `CPL C`, `ANL C,P0.0`, `ORL C,09H`, `MOV P1.0,C`, `SJMP XX`, and `END`.
- Parallel Port 0:** A dialog box showing the port configuration with `P0: 0xFC` and `Pins: 0xFC`.
- Parallel Port 1:** A dialog box showing the port configuration with `P1: 0xFE` and `Pins: 0xFE`.
- Disassembly:** The disassembly window on the right shows the compiled instructions, including `XX: MOV C,P0.0`, `A280 MOV C,0x80.0`, `B3 CPL C`, `71 ANL C,P0.1`, `8281 ANL C,0x80.1`, `81 MOV 09H,C`, `9209 MOV 0x21.1,C`, `A281 MOV C,0x80.1`, `B3 CPL C`, `111 ANL C,P0.0`, `8280 ANL C,0x80.0`, `121 ORL C,09H`, `7209 ORL C,0x21.1`, `131 MOV P1.0,C`, `9290 MOV T2(0x90.0),C`, `141 SJMP XX`, `80EE SJMP XX(C:0006)`, and several `NOP` instructions.
- Command:** The command window at the bottom shows the simulation running with a code size limit of 2K and the load path `"C:\\Keil_v5\\16BEC0034\\Objects\\Sample"`.

The screenshot shows the AVR Studio IDE with the following components:

- Registers:** A list of registers (r0-r7, sp, sp_max, dptr, PC, S, states, sec, paw) is shown on the left.
- Task2.asm:** The assembly code is displayed in the main window. The code includes instructions like `ORG 0000H`, `SETB P0.0`, `SETB P0.1`, `CLR P1.0`, `XX: MOV C,P0.0`, `CPL C`, `ANL C,P0.1`, `MOV 09H,C`, `MOV C,P0.1`, `CPL C`, `ANL C,P0.0`, `ORL C,09H`, `MOV P1.0,C`, `SJMP XX`, and `END`.
- Parallel Port 0:** A dialog box showing the port configuration with `P0: 0xFD` and `Pins: 0xFD`.
- Parallel Port 1:** A dialog box showing the port configuration with `P1: 0xFF` and `Pins: 0xFF`.
- Disassembly:** The disassembly window on the right shows the compiled instructions, including `XX: MOV C,P0.0`, `A280 MOV C,0x80.0`, `B3 CPL C`, `71 ANL C,P0.1`, `8281 ANL C,0x80.1`, `81 MOV 09H,C`, `9209 MOV 0x21.1,C`, `A281 MOV C,0x80.1`, `B3 CPL C`, `111 ANL C,P0.0`, `8280 ANL C,0x80.0`, `121 ORL C,09H`, `7209 ORL C,0x21.1`, `131 MOV P1.0,C`, `9290 MOV T2(0x90.0),C`, `141 SJMP XX`, `80EE SJMP XX(C:0006)`, and several `NOP` instructions.
- Command:** The command window at the bottom shows the simulation running with a code size limit of 2K and the load path `"C:\\Keil_v5\\16BEC0034\\Objects\\Sample"`.

File Edit View Project Flash Debug Peripherals Tools SVCS Window Help

Registers

Register

Regs

r0
r1
r2
r3
r4
r5
r6
r7

Sys

a
b
sp_max
dptr
PC \$
states
sec
paw

Task/Lane

1 ORG 0000H
2 SETB P0.0
3 SETB P0.1
4 CLR P1.0
5 XRI MOV C,P0.0
6 CPL C
7 ANL C,P0.1
8 MOV 09H,C
9 MOV C,P0.1
10 CPL C
11 ANL C,P0.0
12 ORL C,09H
13 MOV P1.0,C
14 SJMP XX
15
16 END

Parallel Port 0

Port 0 7 Bits 0
P0: 0xFF
Pins: 0xFF

Parallel Port 1

Port 1 7 Bits 0
P1: 0xFE
Pins: 0xFE

Disassembly

5: XRI MOV C,P0.0
C:0x0004 A280 MOV C,0x80.0
6: CPL C
C:0x0008 B3 CPL C
7: ANL C,P0.1
C:0x0009 8281 ANL C,0x80.1
8: MOV 09H,C
C:0x000B 9209 MOV 0x21.1,C
9: MOV C,P0.1
C:0x000D A281 MOV C,0x80.1
10: CPL C
C:0x000F B3 CPL C
11: ANL C,P0.0
C:0x0010 8280 ANL C,0x80.0
12: ORL C,09H
C:0x0012 7209 ORL C,0x21.1
13: MOV P1.0,C
C:0x0014 9290 MOV T2(0x90.0),C
14: SJMP XX
C:0x0016 80EE SJMP XX(C:0006)
C:0x0018 00 NOP
C:0x0019 00 NOP
C:0x001A 00 NOP
C:0x001B 00 NOP
C:0x001C 00 NOP
C:0x001D 00 NOP
C:0x001E 00 NOP
C:0x001F 00 NOP
C:0x0020 00 NOP
C:0x0021 00 NOP
C:0x0022 00 NOP
C:0x0023 00 NOP
C:0x0024 00 NOP
C:0x0025 00 NOP
C:0x0026 00 NOP
C:0x0027 00 NOP
C:0x0028 00 NOP
C:0x0029 00 NOP
C:0x002A 00 NOP
C:0x002B 00 NOP
C:0x002C 00 NOP
C:0x002D 00 NOP
C:0x002E 00 NOP
C:0x002F 00 NOP

Command

Running with Code Size Limit: 2K
Load "C:\Keil_v5\16BEC0034\Objects\Sample"

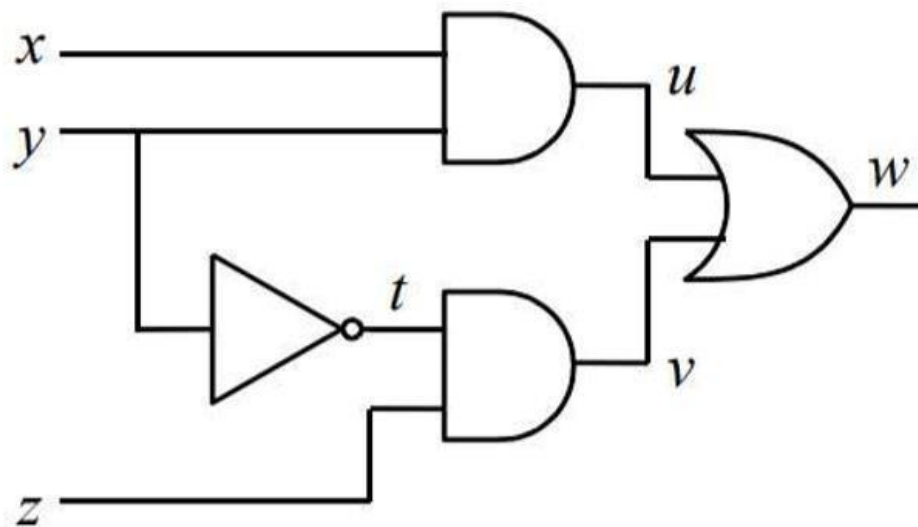
ASM ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE

Simulation t1: 0.00000570 sec L:5 C:1 CAP: NUM SCRL OVR: R/W

Draw the truth table for the following circuits and verify the same by writing the 8051 ALP using Keil IDE.

Program 2A

Write an 8051 ALP for the following circuit



The reduced expression of this logic circuit diagram is:

$$x.y + \bar{y}.z$$

Also

$$u = x.y$$

$$t = \bar{y}$$

$$v = y.z$$

Assume 'x', 'y' and 'z' to be stored in port 0 bits P0.0, P0.1 and P0.2 respectively. And the outputs 'u', 't', 'v' and 'w' in port 1 bits P1.0, P1.1, P1.2 and P1.3 respectively. So the output screenshot and code for the expression will be :

Code :

```
ORG 0000H
SETB P0.0
SETB P0.1
SETB P0.2
CLR P1.0
CLR P1.1
CLR P1.2
CLR P1.3
```

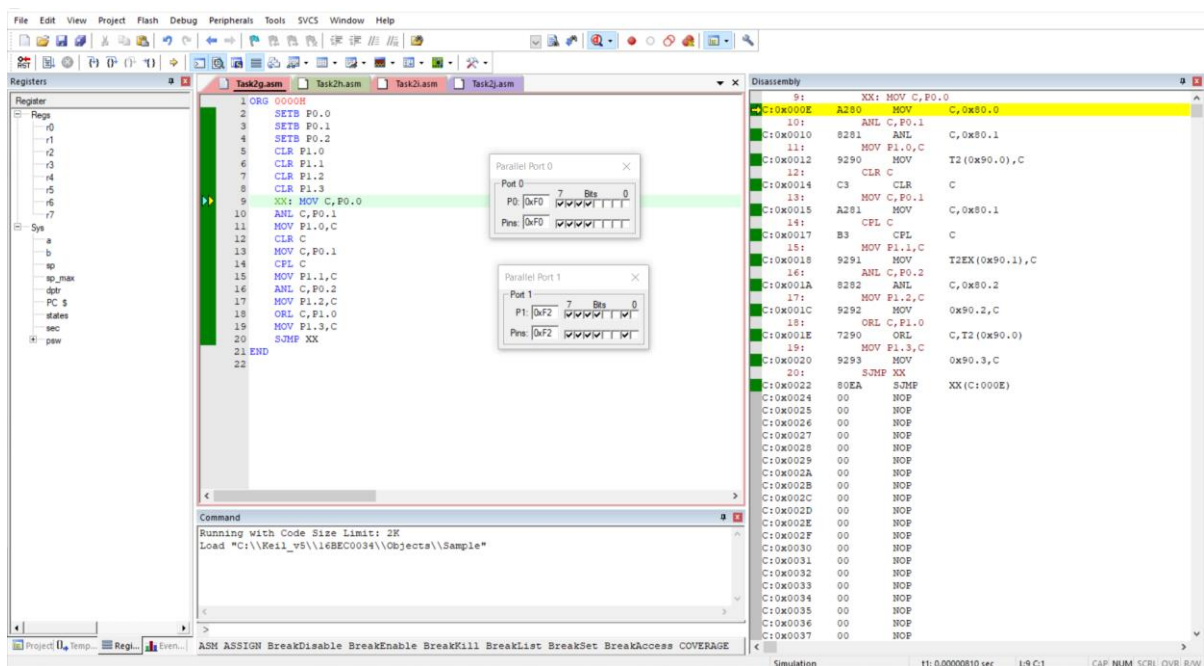
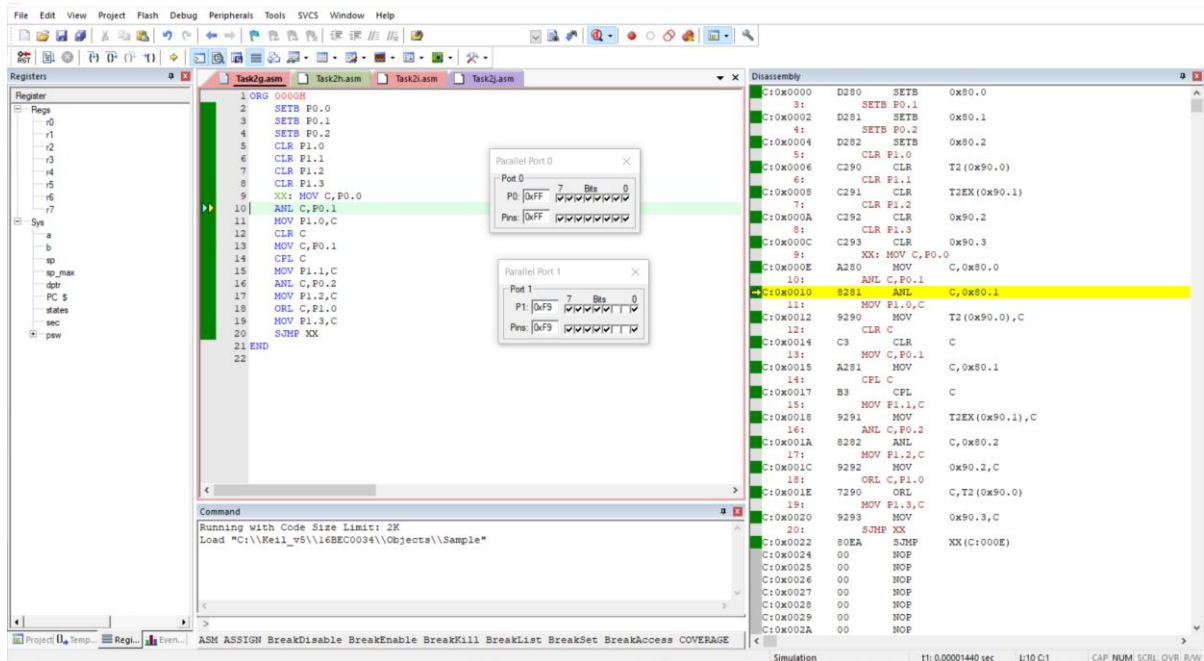
```

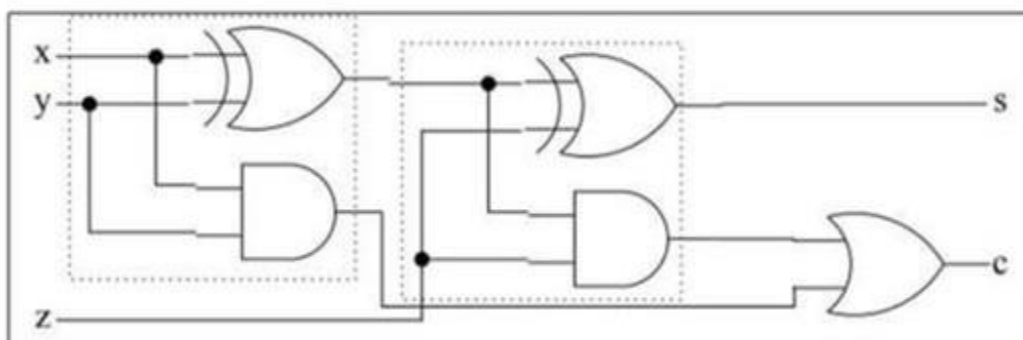
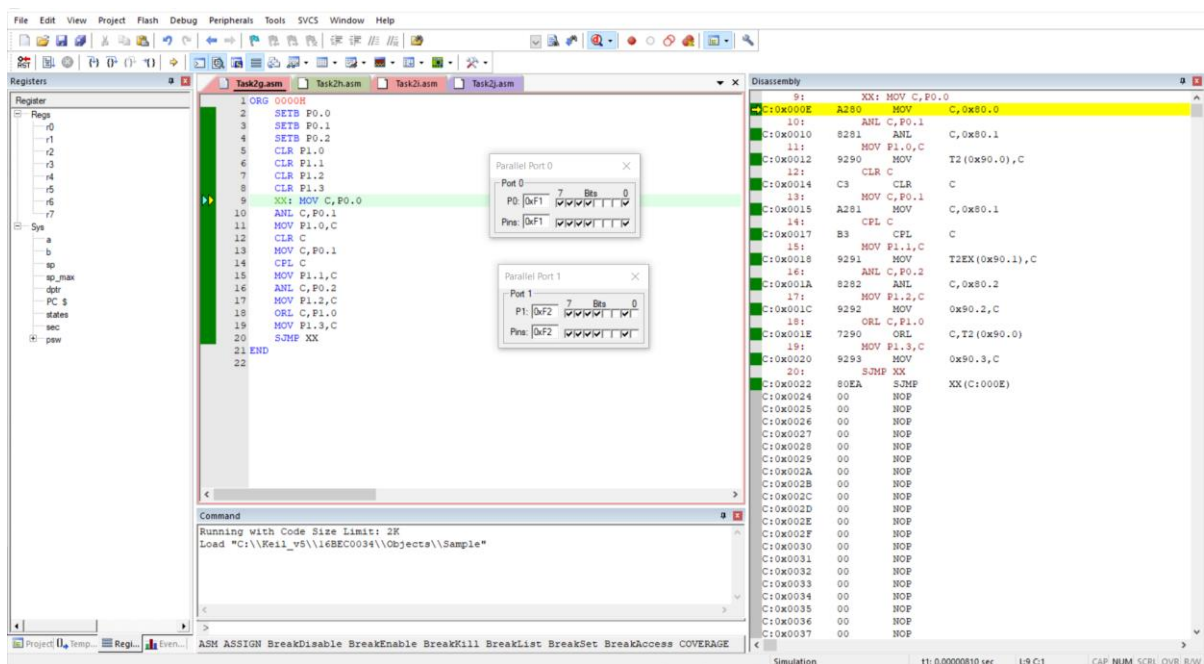
XX: MOV C,P0.0
ANL C,P0.1
MOV P1.0,C
CLR C
MOV C,P0.1
CPL C
MOV P1.1,C
ANL C,P0.2
MOV P1.2,C
ORL C,P1.0
MOV P1.3,C
SJMP XX

```

END

OUTPUT:





The reduced expression of this logic circuit diagram is: (Full adder)

$$s = x \oplus y \oplus z$$
$$c = x.y + (x \oplus y).z$$

Let us assume 'x', 'y' and 'z' to be stored in port 0 bits P0.0, P0.1 and P0.2 respectively. And the outputs 's' and 'c' in port 1 bits P1.0, P1.1 respectively. So the output screenshot and code for the expression will be:

Code & Output:

```
ORG 0000H
    SETB P0.0
    SETB P0.1
    SETB P0.2
    CLR P1.0
    CLR P1.1
    XX: MOV C,P0.0
    CPL C
    ANL C,P0.1
    MOV 07H,C
    CLR C
    MOV C,P0.1
    CPL C
    ANL C,P0.0
    ORL C,07H
    MOV 08H,C
    CLR C
    MOV C,P0.2
    CPL C
    ANL C,08H
    MOV 09H,C
    CLR C
    MOV C,08H
    CPL C
    ANL C,P0.2
    ORL C,09H
    MOV P1.0,C
    CLR C
    MOV C,P0.0
    ANL C,P0.1
    MOV 0AH,C
    CLR C
    MOV C,P0.0
    CPL C
    ANL C,P0.1
    MOV 0BH,C
    CLR C
    MOV C,P0.1
    CPL C
    ANL C,P0.0
    ORL C,0BH
```



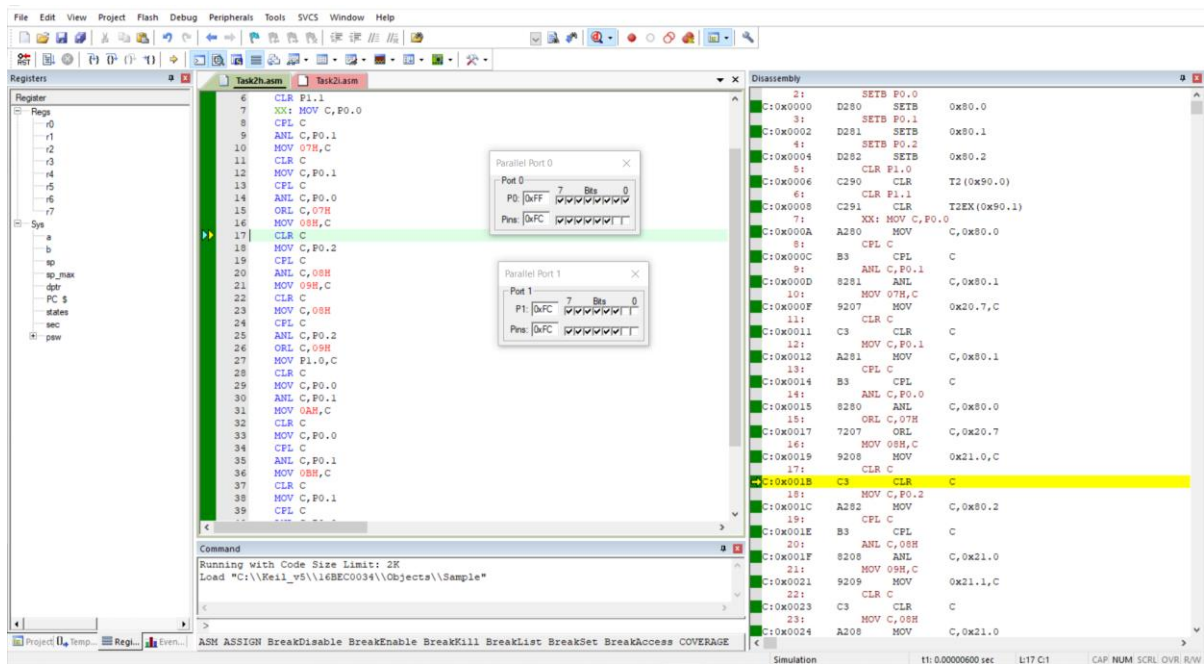
```

ANL C,P0.2
ORL C,0AH
MOV P1.1,C
SJMP XX

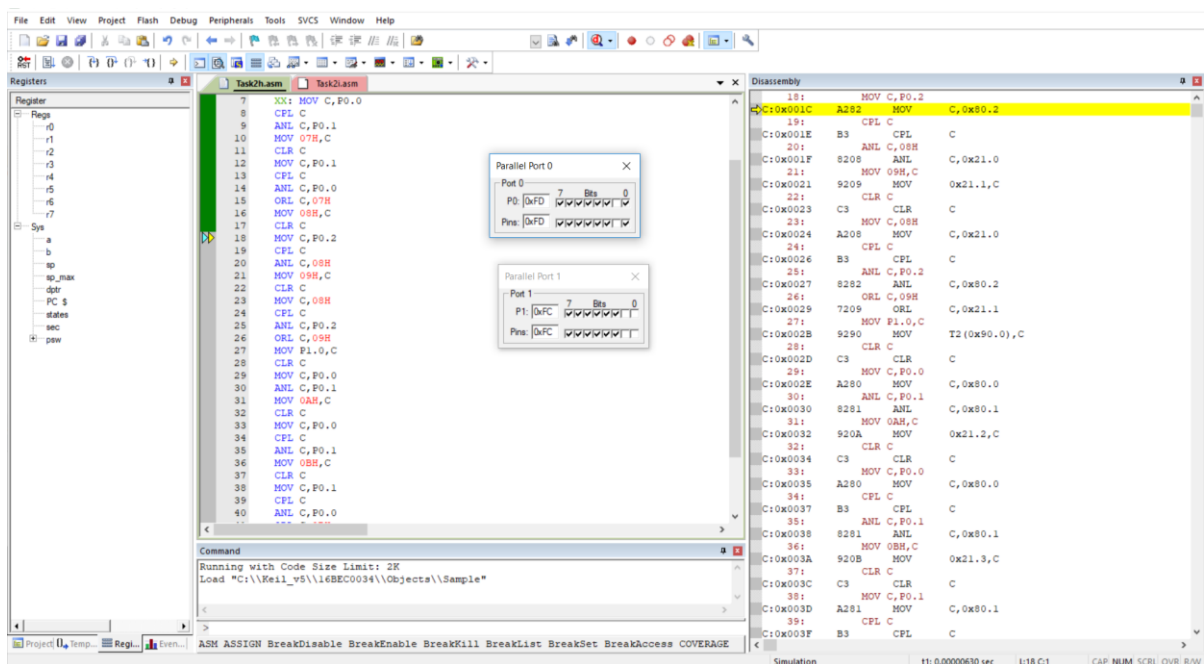
```

END

1. $x=0; y=0; z=0$ so $s=0$ and $c=0$



2. $x=0; y=0; z=1$ so $s=1$ and $c=0$



The screenshot displays the Keil uVision IDE interface. The main window shows the assembly code for `Task2.asm`. The code starts with an origin `ORG 9000H` and includes instructions like `SETB P0.0`, `SETB P0.1`, `SETB P0.2`, `CLR P1.0`, `CLR P1.1`, `MOV C, P0.0`, `ANL C, P0.1`, `MOV 07H, C`, `CLR C`, `MOV C, P0.1`, `CPL C`, `ANL C, P0.0`, `ORL C, 07H`, `MOV 08H, C`, `CLR C`, `MOV C, P0.2`, `CPL C`, `ANL C, 08H`, `MOV 09H, C`, `CLR C`, `MOV C, 08H`, `CPL C`, `ANL C, P0.2`, `ORL C, 08H`, `MOV P1.0, C`, `CLR C`, `MOV C, P0.0`, `ANL C, P0.1`, `MOV 09H, C`, `CLR C`, `MOV C, P0.0`, `CPL C`, and `...`. The command window shows the message: "Running with Code Size Limit: 2K Load 'C:\Keil_v5\116BEC0034\Objects\Sample'".

Two configuration windows are open: "Parallel Port 0" and "Parallel Port 1". Both windows show the "Port 0" and "Port 1" settings. The "Parallel Port 0" window shows "Port 0" with "P0" set to `[x]FD` and "Pins" set to `[x]FD`. The "Parallel Port 1" window shows "Port 1" with "P1" set to `[x]FA` and "Pins" set to `[x]FA`.

The Disassembly window shows the assembly code for `Task2.asm` with the following instructions: `B3: CPL C`, `C:0x000C B3: ANL C, P0.1`, `C:0x000D 8281: ANL C, 0x80.1`, `C:0x000F 101: MOV 07H, C`, `C:0x0010 9207: MOV 0x20.7, C`, `C:0x0011 111: CLR C`, `C:0x0011 C3: CLR C`, `C:0x0012 121: MOV C, P0.1`, `C:0x0012 A281: MOV C, 0x80.1`, `C:0x0013 131: CPL C`, `C:0x0014 B3: CPL C`, `C:0x0015 141: ANL C, P0.0`, `C:0x0015 8280: ANL C, 0x80.0`, `C:0x0016 151: ORL C, 07H`, `C:0x0017 7207: ORL C, 0x20.7`, `C:0x0018 161: MOV 08H, C`, `C:0x0019 9208: MOV 0x21.0, C`, `C:0x001A 171: CLR C`, `C:0x001B C3: CLR C`, `C:0x001C 181: MOV C, P0.2`, `C:0x001D A282: MOV C, 0x80.2`, `C:0x001E 191: CPL C`, `C:0x001F B3: CPL C`, `C:0x0020 201: ANL C, 08H`, `C:0x0021 9208: ANL C, 0x21.0`, `C:0x0022 211: MOV 09H, C`, `C:0x0023 9209: MOV 0x21.1, C`, `C:0x0024 221: CLR C`, `C:0x0025 C3: CLR C`, `C:0x0026 231: MOV C, 08H`, `C:0x0027 A208: MOV C, 0x21.0`, `C:0x0028 241: CPL C`, `C:0x0029 B3: CPL C`, `C:0x002A 251: ANL C, P0.2`, `C:0x002B 8282: ANL C, 0x80.2`, `C:0x002C 261: ORL C, 08H`, `C:0x002D 7209: ORL C, 0x21.1`, `C:0x002E 271: MOV P1.0, C`, `C:0x002F 9290: MOV T2(0x90.0), C`, `C:0x0030 281: CLR C`, `C:0x0031 C3: CLR C`, `C:0x0032 291: MOV C, P0.0`, `C:0x0033 A280: MOV C, 0x80.0`.

The status bar at the bottom shows "ASM ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE".

The screenshot displays the WinBox debugger interface with the following components:

- Menu Bar:** File, Edit, View, Project, Flash, Debug, Peripherals, Tools, SVCS, Windows, Help.
- Toolbars:** Standard Windows toolbars and a WinBox-specific toolbar with icons for file operations, navigation, and debugging.
- Registers Panel:**
 - Register:** Lists registers r0 through r7, Sys, a, b, sp, sp_max, dptr, PC, S, status, sec, and Code.
 - Value:** Shows the current value of the selected register.
- Task2.asm Panel:**
 - Assembly Code:**

```

3 SETB P0.1
4 SETB P0.2
5 CLR P1.0
6 CLR P1.1
7 XK: MOV C, P0.0
8 CPL C
9 ANL C, P0.1
10 MOV 07H, C
11 CLR C
12 MOV C, P0.1
13 CPL C
14 ANL C, P0.0
15 ORL C, 07H
16 MOV 08H, C
17 CLR C
18 MOV C, P0.2
19 CPL C
20 ANL C, 08H
21 MOV 09H, C
22 CLR C
23 MOV C, 08H
24 CPL C
25 ANL C, P0.2
26 ORL C, 09H
27 MOV P1.0, C
28 CLR C
29 MOV C, P0.0
30 ANL C, P0.1
31 MOV 0AH, C
32 CLR C
33 MOV C, P0.0
34 CPL C
35 ANL C, P0.1
36 MOV 0BH, C

```
 - Command Window:**

```

Running with Code Size Limit: 2K
Load "C:\\Reil_v5\\i16BEC0034\\Objects\\Sample"

```
- Disassembly Panel:**
 - Address:** 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 630, 631, 632, 633, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 655, 656, 657, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672, 673, 674, 675, 676, 677, 678, 679, 680, 681, 682, 683, 684, 685, 686, 687, 688, 689, 690, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702, 703, 704, 705, 706, 707, 708, 709, 710, 711, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 724, 725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 737, 738, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 769, 770, 771, 772, 773, 774, 77

5. $x=1; y=0; z=0$ so $s=1$ and $c=0$

Registers

Register

Regs

r0

r1

r2

r3

r4

r5

r6

r7

Sys

a

b

sp

sp_max

dptr

PC

status

sec

psw

Task2.asm

1 ORG 0000H

2 SETB P0.0

3 SETB P0.1

4 SETB P0.2

5 CLR P1.0

6 CLR P1.1

7 XX: MOV C, P0.0

8 CPL C

9 ANL C, P0.1

10 MOV 07H, C

11 CLR C

12 MOV C, P0.1

13 CPL C

14 ANL C, P0.0

15 ORL C, 07H

16 MOV 08H, C

17 CLR C

18 MOV C, P0.2

19 CPL C

20 ANL C, 08H

21 MOV 09H, C

22 CLR C

23 MOV C, 08H

24 CPL C

25 ANL C, P0.2

26 ORL C, 09H

27 MOV P1.0, C

28 CLR C

29 MOV C, P0.0

30 ANL C, P0.1

31 MOV 0AH, C

32 CLR C

33 MOV C, P0.0

34 CPL C

Command

Running with Code Size Limit: 2K

Load "C:\\Keil_v5\\16BEC0034\\Objects\\Sample"

ASM ASSIGN BreakDisable BreakEnable BreakFill BreakList BreakSet BreakAccess COVERAGE

Disassembly

12: MOV C, P0.1

13: A281 MOV C, 0x80.1

14: B3 CPL C

15: 8280 ANL C, P0.0

16: 7207 ORL C, 07H

17: 9208 MOV 0x21.0, C

18: C3 CLR C

19: A282 MOV C, P0.2

20: B3 CPL C

21: 8208 ANL C, 08H

22: 9209 MOV 0x21.1, C

23: C3 CLR C

24: A208 MOV C, 0x21.0

25: B3 CPL C

26: 8282 ANL C, P0.2

27: 7209 ORL C, 09H

28: 9290 MOV T2(0x90.0), C

29: C3 CLR C

30: A280 MOV C, P0.0

31: 8281 ANL C, P0.1

32: 920A MOV 0x21.2, C

33: C3 CLR C

34: A280 MOV C, P0.0

Simulation

t1: 0.00003900 sec L12 C1 CAP. NUM SCRL OVR. RW

6. $x=1; y=0; z=1$ so $s=0$ and $c=1$

Registers

Register

Regs

r0

r1

r2

r3

r4

r5

r6

r7

Sys

a

b

sp

sp_max

dptr

PC

status

sec

psw

Task2.asm

1 ORG 0000H

2 SETB P0.0

3 SETB P0.1

4 SETB P0.2

5 CLR P1.0

6 CLR P1.1

7 XX: MOV C, P0.0

8 CPL C

9 ANL C, P0.1

10 MOV 07H, C

11 CLR C

12 MOV C, P0.1

13 CPL C

14 ANL C, P0.0

15 ORL C, 07H

16 MOV 08H, C

17 CLR C

18 MOV C, P0.2

19 CPL C

20 ANL C, 08H

21 MOV 09H, C

22 CLR C

23 MOV C, 08H

24 CPL C

25 ANL C, P0.2

26 ORL C, 09H

27 MOV P1.0, C

28 CLR C

29 MOV C, P0.0

30 ANL C, P0.1

31 MOV 0AH, C

32 CLR C

33 MOV C, P0.0

34 CPL C

Command

Running with Code Size Limit: 2K

Load "C:\\Keil_v5\\16BEC0034\\Objects\\Sample"

ASM ASSIGN BreakDisable BreakEnable BreakFill BreakList BreakSet BreakAccess COVERAGE

Disassembly

12: MOV C, P0.1

13: A281 MOV C, 0x80.1

14: B3 CPL C

15: 8280 ANL C, P0.0

16: 7207 ORL C, 07H

17: 9208 MOV 0x21.0, C

18: C3 CLR C

19: A282 MOV C, P0.2

20: B3 CPL C

21: 8208 ANL C, 08H

22: 9209 MOV 0x21.1, C

23: C3 CLR C

24: A208 MOV C, 0x21.0

25: B3 CPL C

26: 8282 ANL C, P0.2

27: 7209 ORL C, 09H

28: 9290 MOV T2(0x90.0), C

29: C3 CLR C

30: A280 MOV C, P0.0

31: 8281 ANL C, P0.1

32: 920A MOV 0x21.2, C

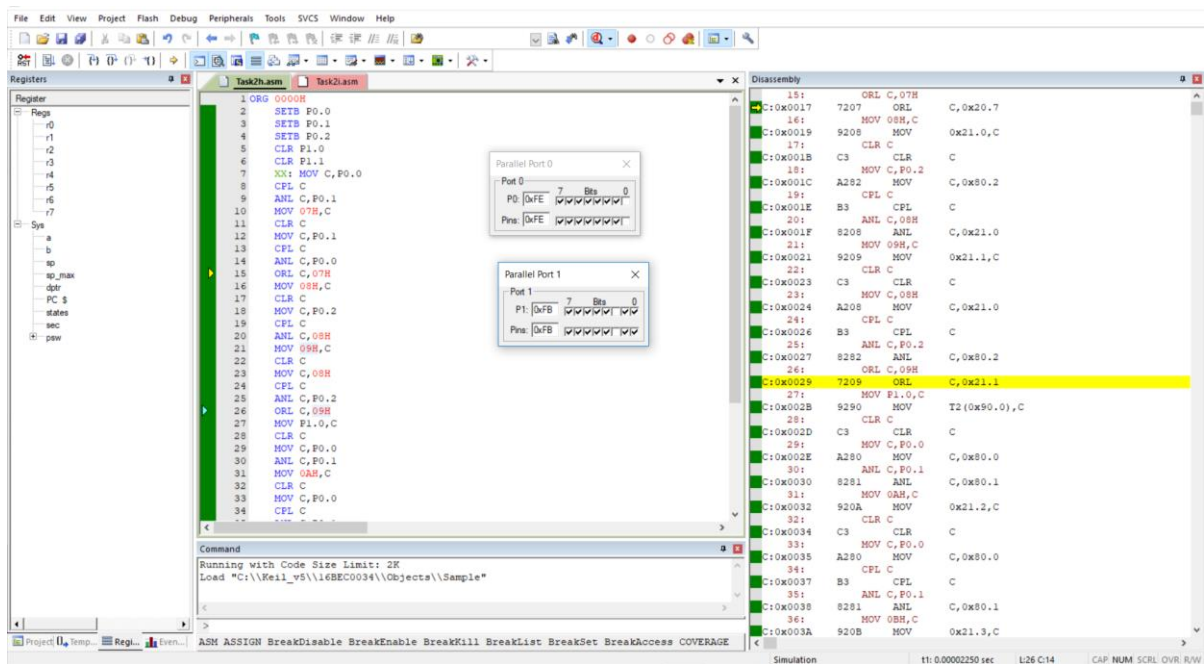
33: C3 CLR C

34: A280 MOV C, P0.0

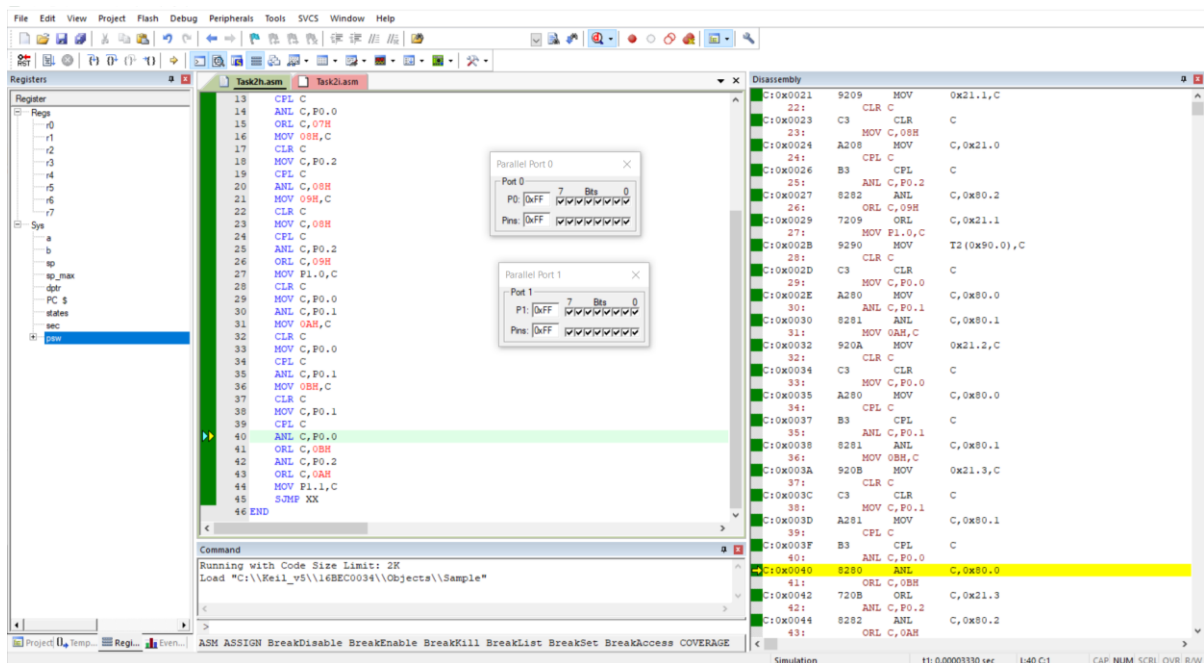
Simulation

t1: 0.00002040 sec L12 C1 CAP. NUM SCRL OVR. RW

7. $x=1; y=1; z=0$ so $s=0$ and $c=1$

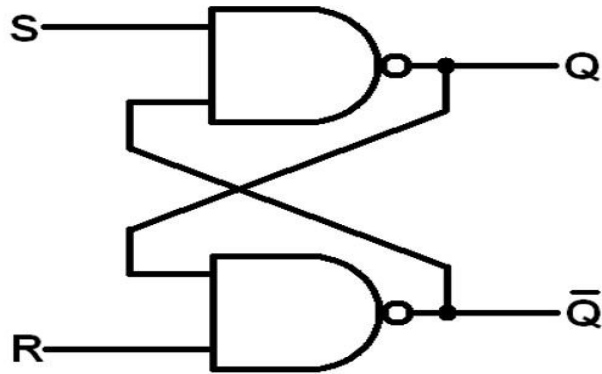


8. $x=1; y=1; z=1$ so $s=1$ and $c=1$



Program 2c

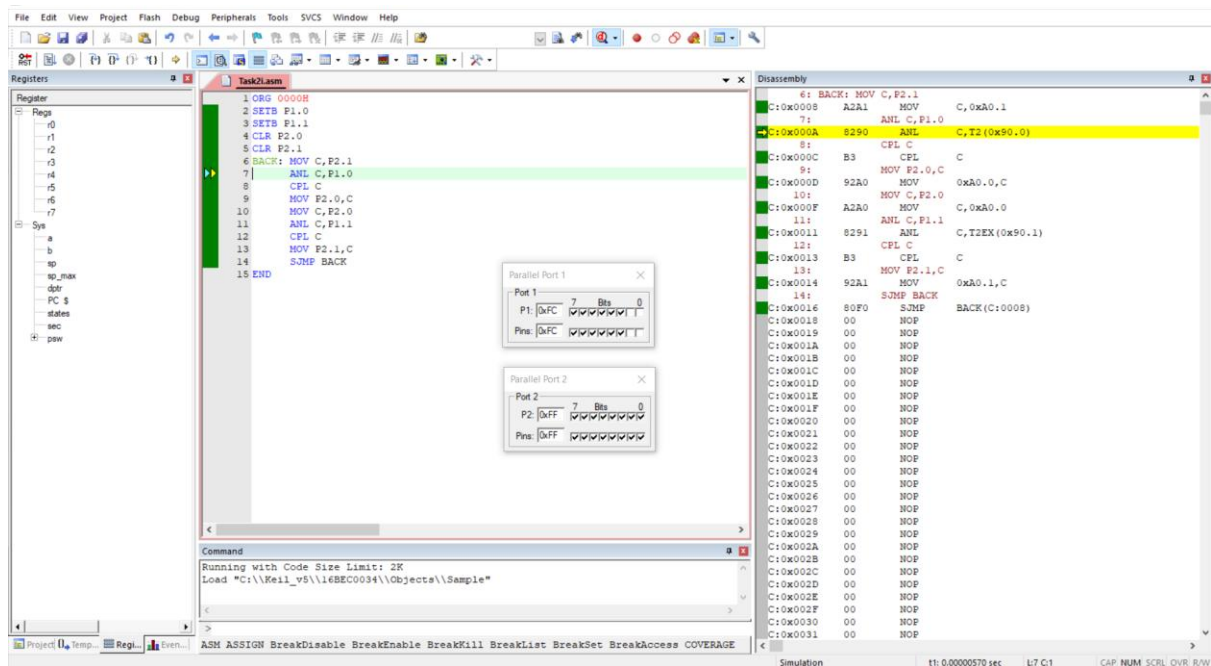
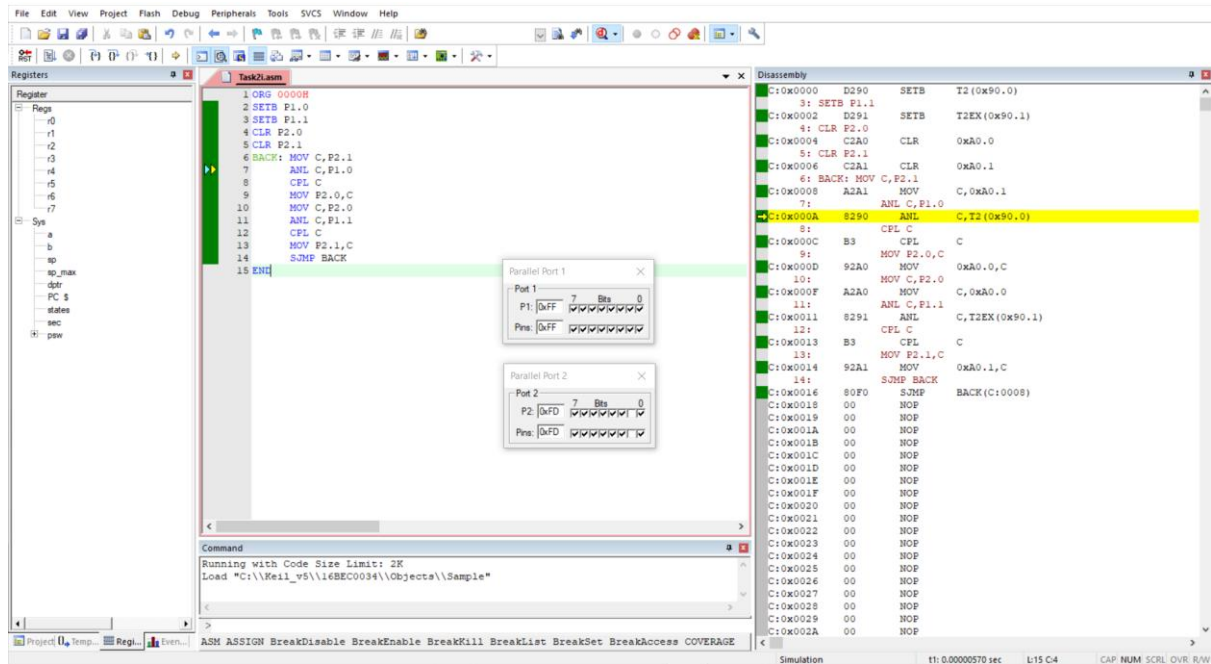
Write an 8051 ALP for the following circuit.



Code:

```
ORG 0000H
SETB P1.0
SETB P1.1
CLR P2.0
CLR P2.1
BACK: MOV C,P2.1
      ANL C,P1.0
      CPL C
      MOV P2.0,C
      MOV C,P2.0
      ANL C,P1.1
      CPL C
      MOV P2.1,C
      SJMP BACK
END
```


OUTPUT:



File Edit View Project Flash Debug Peripherals Tools SVCS Window Help

Registers

Register

Regs

r0
r1
r2
r3
r4
r5
r6
r7

Sys

a
b
sp
sp_max
dptr
PC
status
vec
paw

Task2.asm

```

1 ORG 0000H
2 SETB P1.0
3 SETB P1.1
4 CLR P2.0
5 CLR P2.1
6 BACK: MOV C,P2.1
7 ANL C,P1.0
8 CPL C
9 MOV P2.0,C
10 MOV C,P2.0
11 ANL C,P1.1
12 CPL C
13 MOV P2.1,C
14 SJMP BACK
15 END

```

Parallel Port 1

Port 1

P1: 0xFE 7 Bits 0

Pins: 0xFE 7 Bits 0

Parallel Port 2

Port 2

P2: 0xFD 7 Bits 0

Pins: 0xFD 7 Bits 0

Command

Running with Code Size Limit: 2K
Load "C:\Keil_v5\16BEC0034\Objects\Sample"

ASM ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE

Disassembly

```

6: BACK: MOV C,P2.1
C:0x0008 A2A1 MOV C,0xA0.1
7: ANL C,P1.0
C:0x000A 8290 ANL C,T2(0x90.0)
8: CPL C
C:0x000C B3 CPL C
9: MOV P2.0,C
C:0x000D 92A0 MOV 0xA0.0,C
10: MOV C,P2.0
C:0x000F A2A0 MOV C,0xA0.0
11: ANL C,P1.1
C:0x0011 8291 ANL C,T2EX(0x90.1)
12: CPL C
C:0x0013 B3 CPL C
13: MOV P2.1,C
C:0x0014 92A1 MOV 0xA0.1,C
14: SJMP BACK
C:0x0016 80F0 SJMP BACK(C:0008)
C:0x0018 00 NOP
C:0x0019 00 NOP
C:0x001A 00 NOP
C:0x001B 00 NOP
C:0x001C 00 NOP
C:0x001D 00 NOP
C:0x001E 00 NOP
C:0x001F 00 NOP
C:0x0020 00 NOP
C:0x0021 00 NOP
C:0x0022 00 NOP
C:0x0023 00 NOP
C:0x0024 00 NOP
C:0x0025 00 NOP
C:0x0026 00 NOP
C:0x0027 00 NOP
C:0x0028 00 NOP
C:0x0029 00 NOP
C:0x002A 00 NOP
C:0x002B 00 NOP
C:0x002C 00 NOP
C:0x002D 00 NOP
C:0x002E 00 NOP
C:0x002F 00 NOP
C:0x0030 00 NOP
C:0x0031 00 NOP

```

Simulation t1: 0.0000540 sec L6 C1 CAP_NUM SCRL OVR R/W

File Edit View Project Flash Debug Peripherals Tools SVCS Window Help

Registers

Register

Regs

r0
r1
r2
r3
r4
r5
r6
r7

Sys

a
b
sp
sp_max
dptr
PC
status
vec
paw

Task2.asm

```

1 ORG 0000H
2 SETB P1.0
3 SETB P1.1
4 CLR P2.0
5 CLR P2.1
6 BACK: MOV C,P2.1
7 ANL C,P1.0
8 CPL C
9 MOV P2.0,C
10 MOV C,P2.0
11 ANL C,P1.1
12 CPL C
13 MOV P2.1,C
14 SJMP BACK
15 END

```

Parallel Port 1

Port 1

P1: 0xFD 7 Bits 0

Pins: 0xFD 7 Bits 0

Parallel Port 2

Port 2

P2: 0xFF 7 Bits 0

Pins: 0xFF 7 Bits 0

Command

Running with Code Size Limit: 2K
Load "C:\Keil_v5\16BEC0034\Objects\Sample"

ASM ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE

Disassembly

```

6: BACK: MOV C,P2.1
C:0x0008 A2A1 MOV C,0xA0.1
7: ANL C,P1.0
C:0x000A 8290 ANL C,T2(0x90.0)
8: CPL C
C:0x000C B3 CPL C
9: MOV P2.0,C
C:0x000D 92A0 MOV 0xA0.0,C
10: MOV C,P2.0
C:0x000F A2A0 MOV C,0xA0.0
11: ANL C,P1.1
C:0x0011 8291 ANL C,T2EX(0x90.1)
12: CPL C
C:0x0013 B3 CPL C
13: MOV P2.1,C
C:0x0014 92A1 MOV 0xA0.1,C
14: SJMP BACK
C:0x0016 80F0 SJMP BACK(C:0008)
C:0x0018 00 NOP
C:0x0019 00 NOP
C:0x001A 00 NOP
C:0x001B 00 NOP
C:0x001C 00 NOP
C:0x001D 00 NOP
C:0x001E 00 NOP
C:0x001F 00 NOP
C:0x0020 00 NOP
C:0x0021 00 NOP
C:0x0022 00 NOP
C:0x0023 00 NOP
C:0x0024 00 NOP
C:0x0025 00 NOP
C:0x0026 00 NOP
C:0x0027 00 NOP
C:0x0028 00 NOP
C:0x0029 00 NOP
C:0x002A 00 NOP
C:0x002B 00 NOP
C:0x002C 00 NOP
C:0x002D 00 NOP
C:0x002E 00 NOP
C:0x002F 00 NOP
C:0x0030 00 NOP
C:0x0031 00 NOP

```

Simulation t1: 0.00000660 sec L9 C1 CAP_NUM SCRL OVR R/W