**EXPERIMENT 10**

**Aim:** Write a Program using ALP to Simulate Microcontroller interfacing with 7 segment display using http://vlabs.iitb.ac.in/vlabs-dev/labs/8051-Microcontroller-Lab/labs/exp1/simulation.php. Display your SAP ID using this tool

**Submission Sheet**

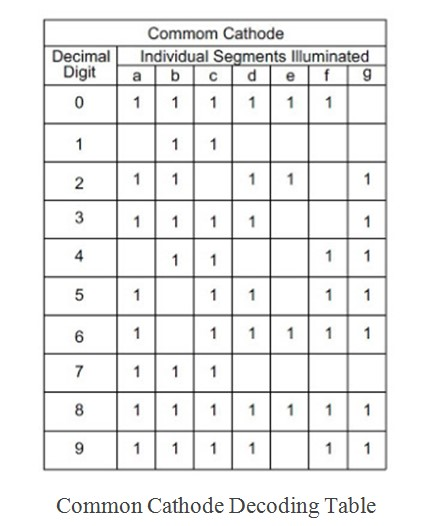
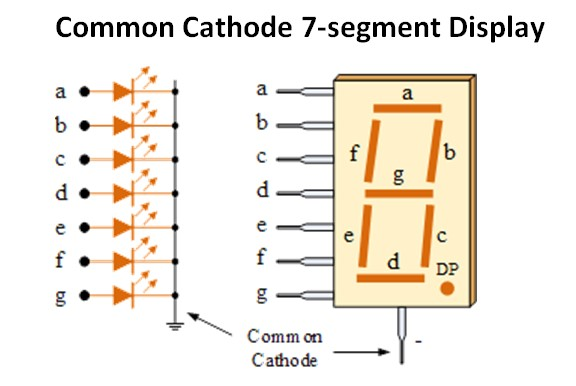
| **SAP ID** | **Name of Student** | **Date of Experiment** | **Date of Submission** | **Remarks** |
| --- | --- | --- | --- | --- |
| 60004190057 | Junaid Girkar | 3-12-2021 | 10-12-2021 |  |

**THEORY**:

The 7-segment display consists of seven LEDs arranged in a rectangular fashion. Each of the seven LEDs is called a segment because when illuminated the segment forms part of a numerical digit (both Decimal and Hex) to be displayed. An additional 8th LED is sometimes used within the same package which is the indication of a decimal point(DP), when two or more 7-segment displays are connected together numbers greater than ten can be displayed.

So by forward biasing the appropriate pins of the LED segments in a particular order, some segments will be glowing and others will remain as it is, allowing the desired character pattern of the number to be generated on the display. This then allows us to display each of the ten decimal digits 0 to 9 on the same 7-segment display.

In the common cathode display, all the cathode connections of the LED segments are joined together to logic “0” or ground. The individual segments are illuminated by application of a “HIGH”, or logic “1” signal via a current limiting resistor to forward bias the individual Anode terminals (a-g).

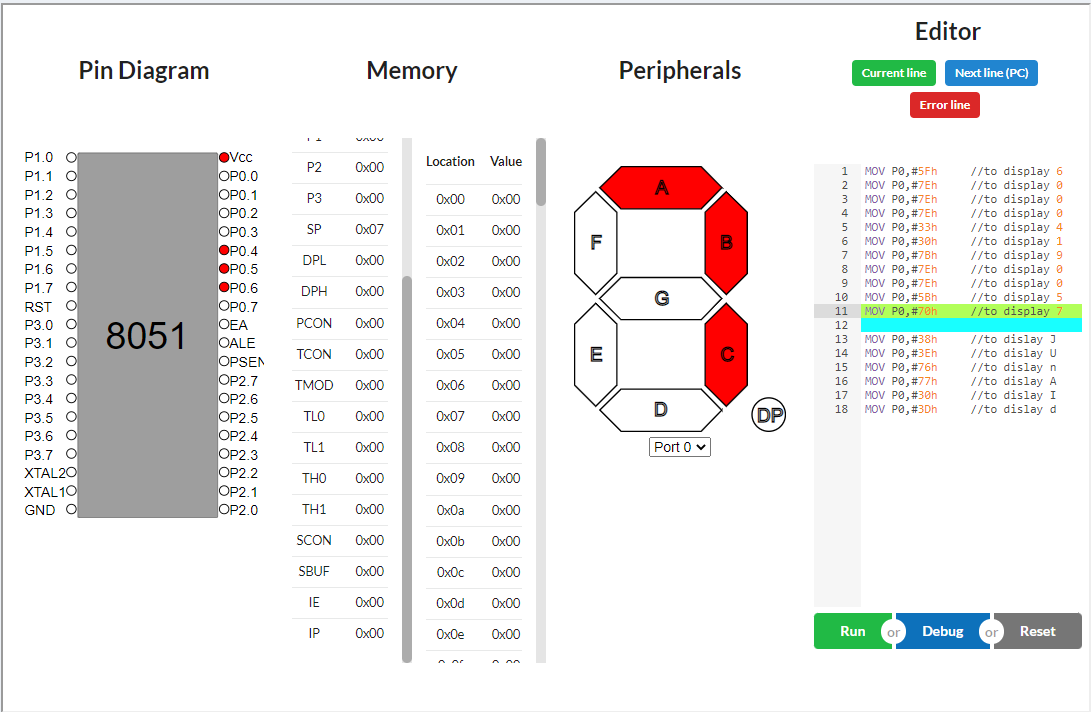
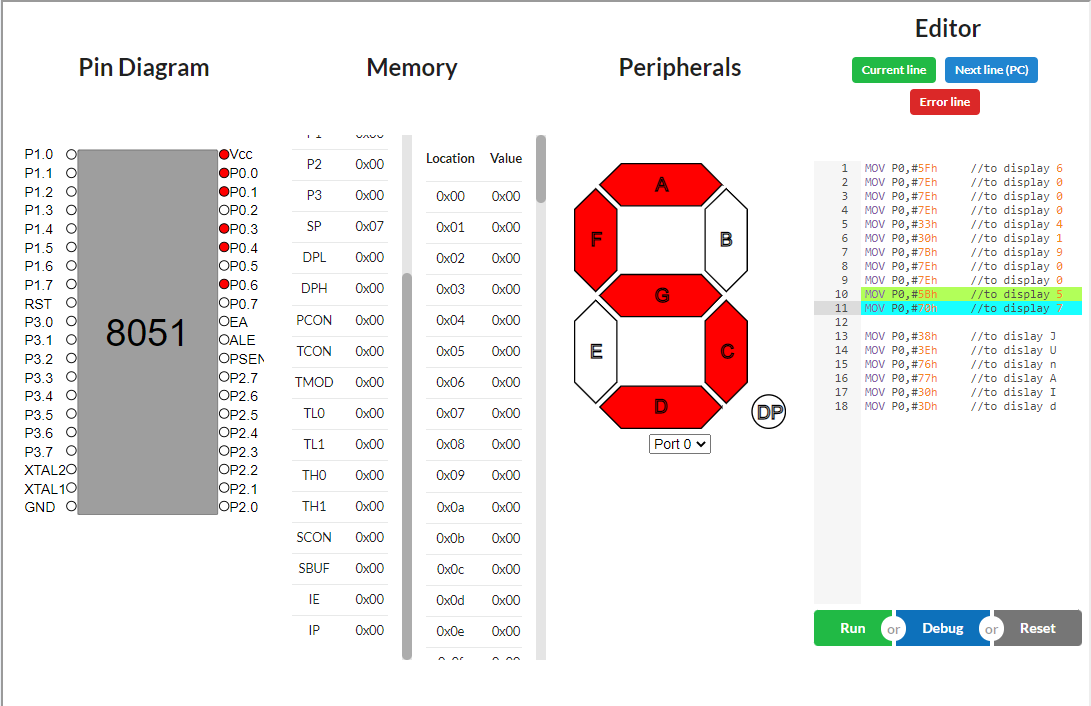
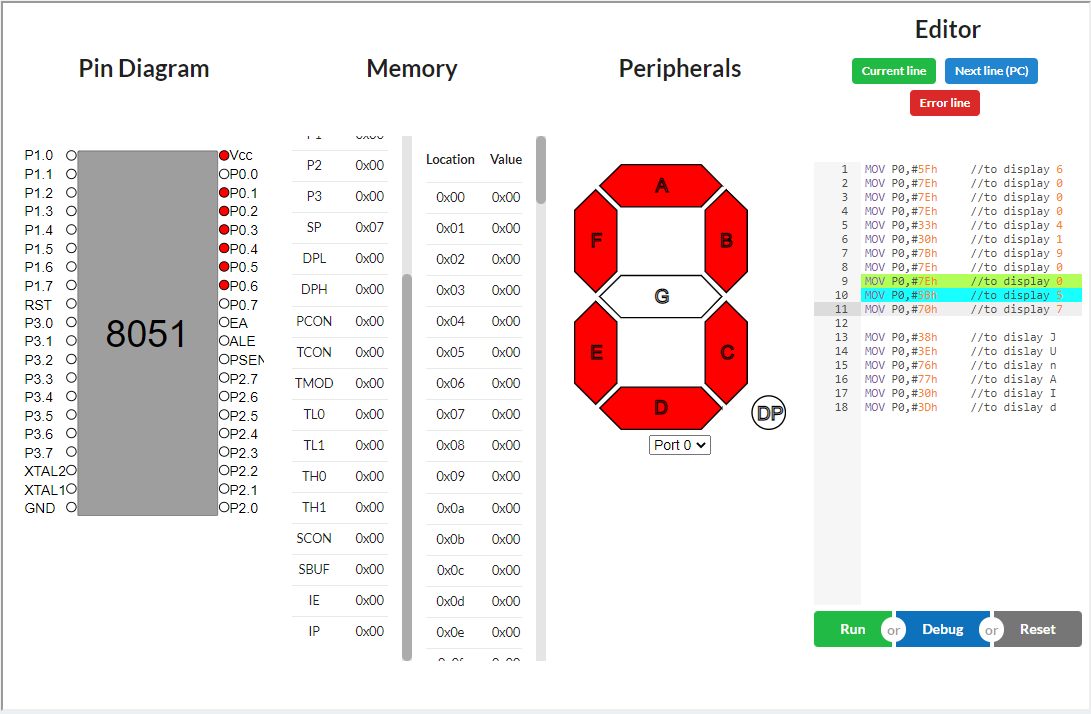
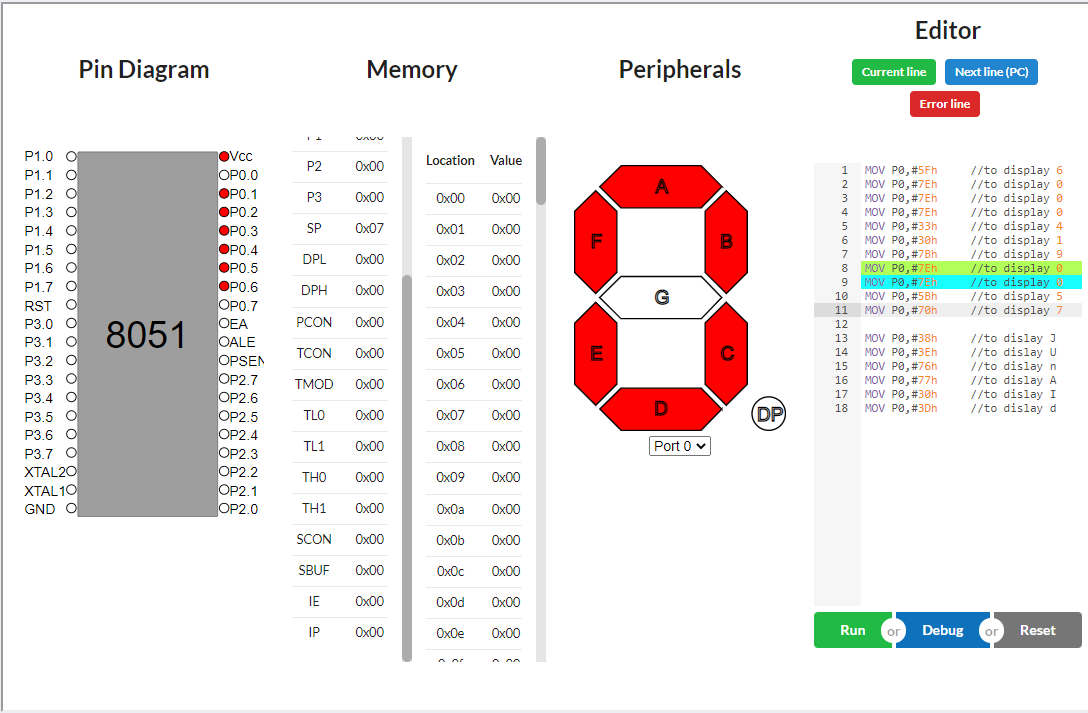
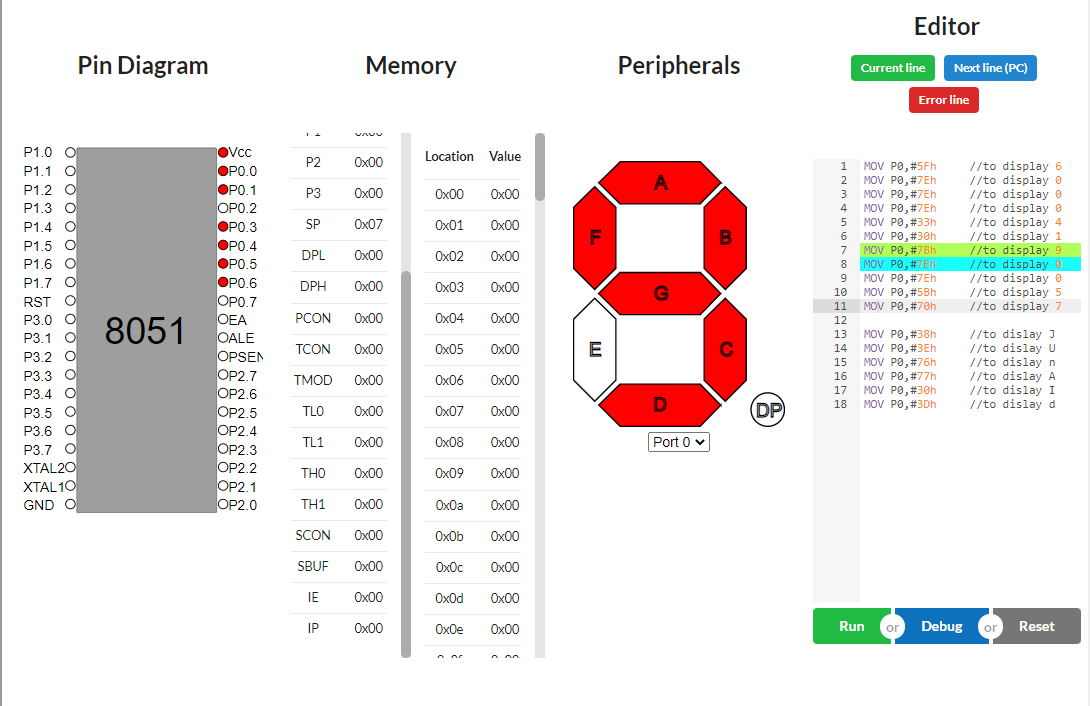
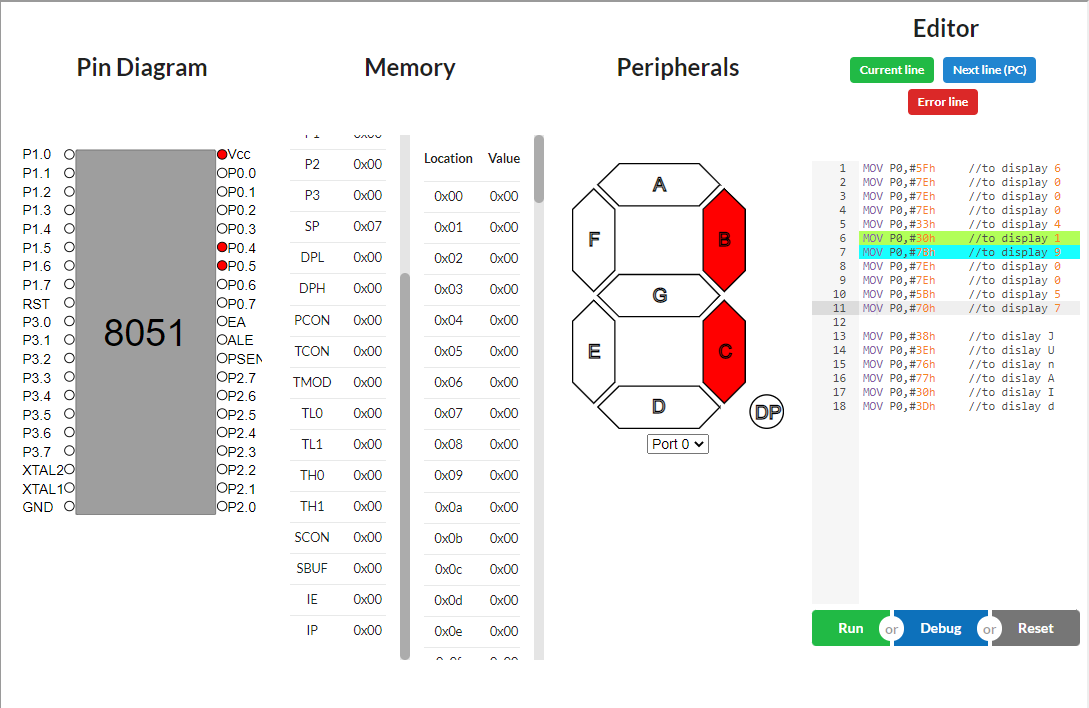
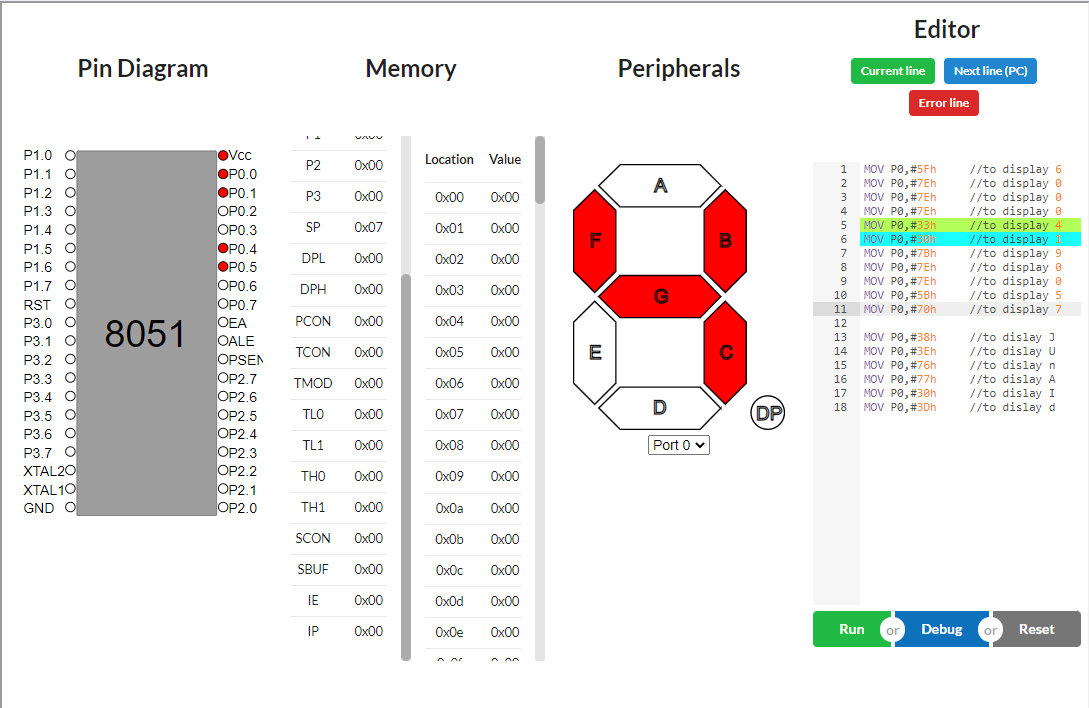
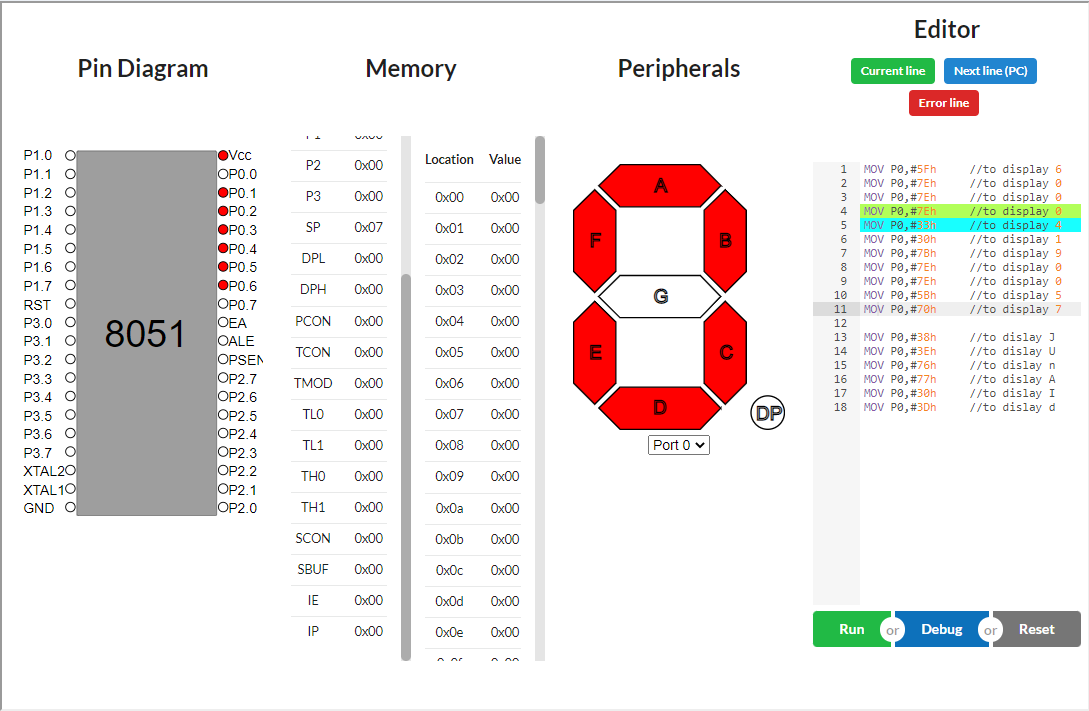
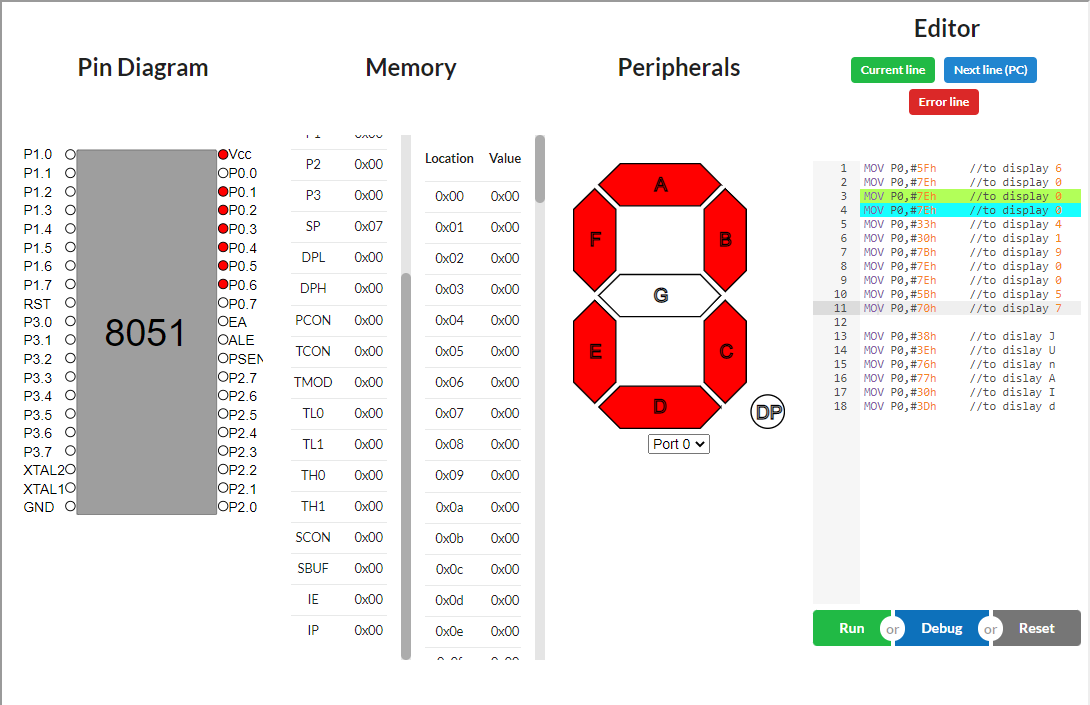
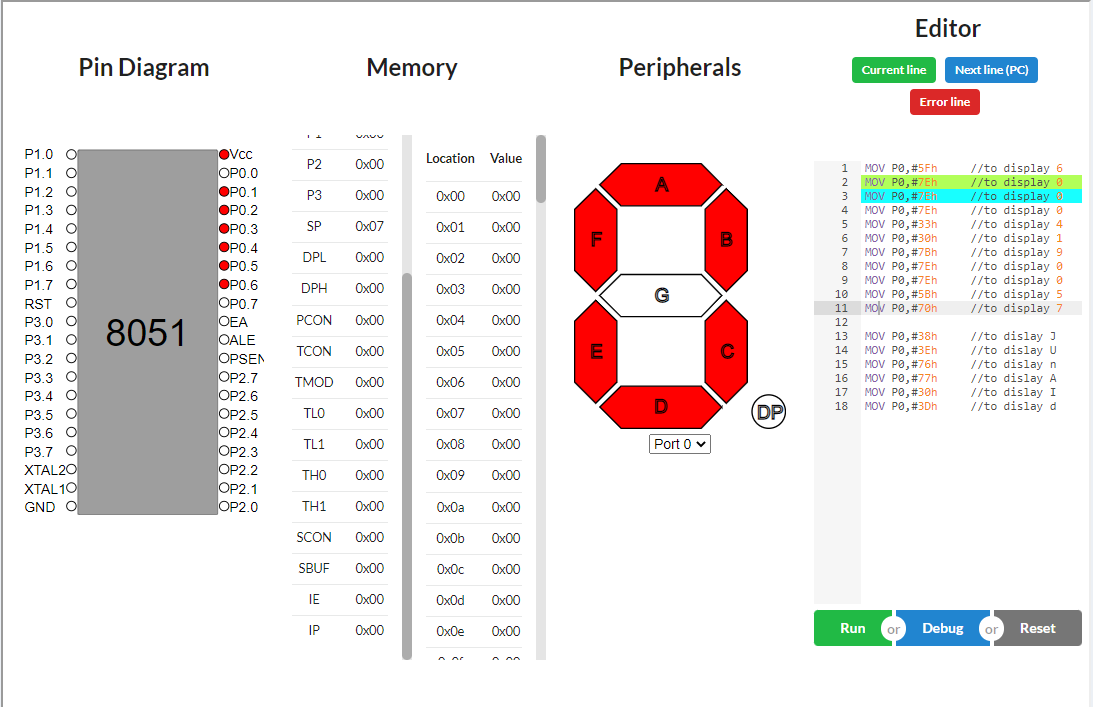
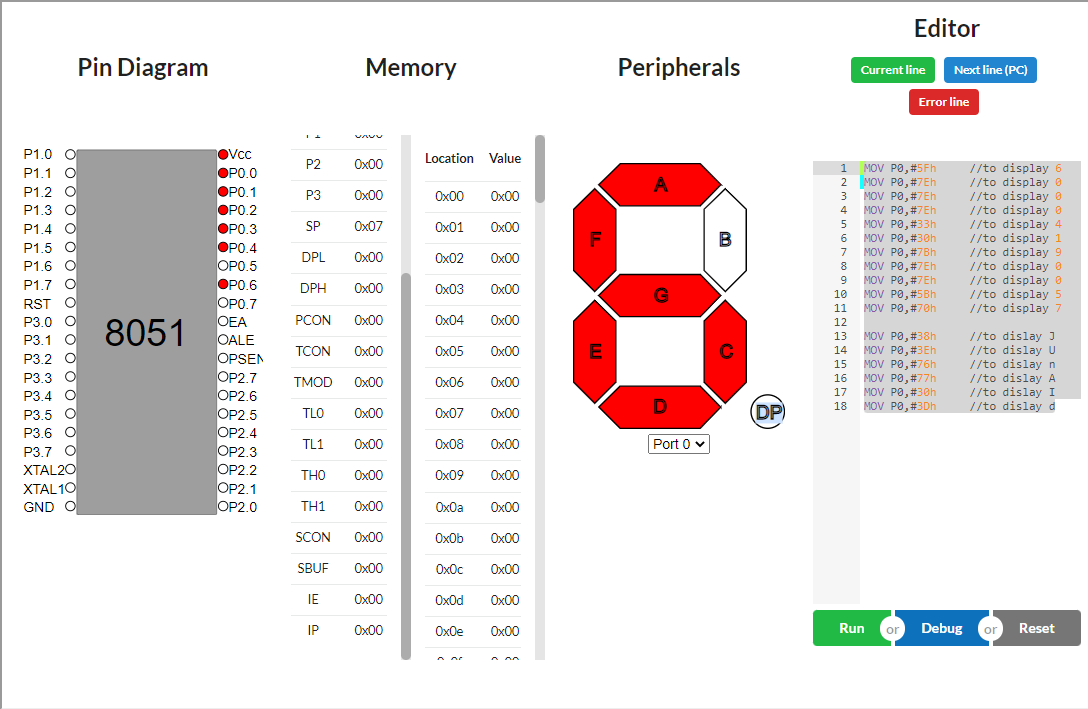


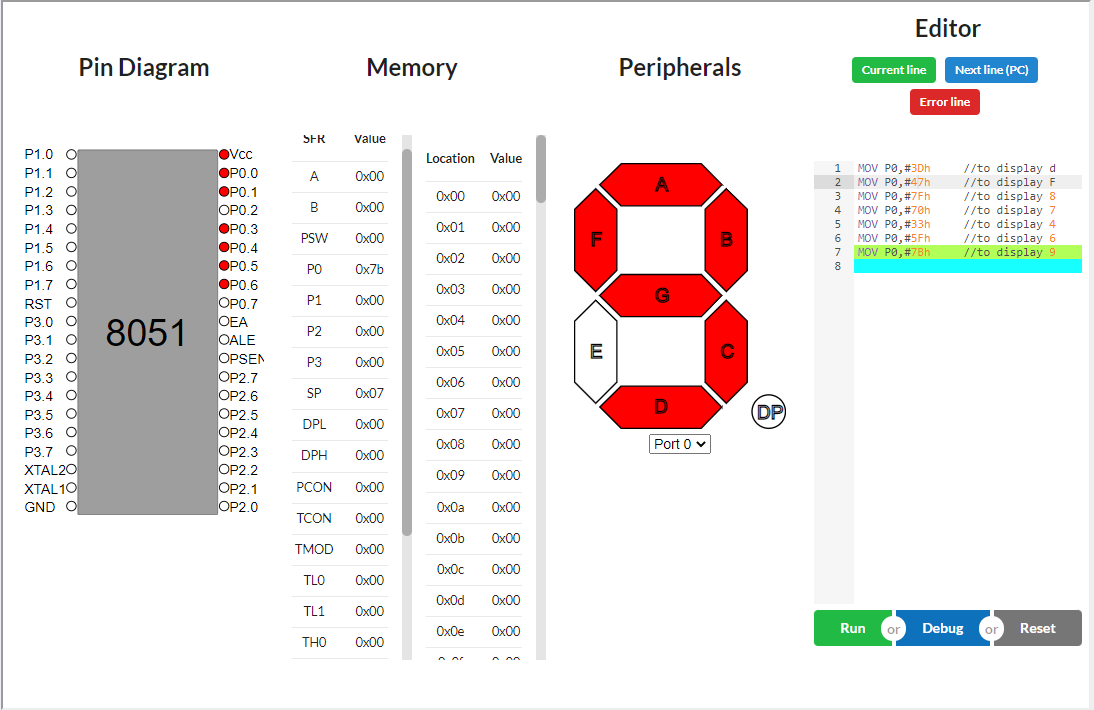
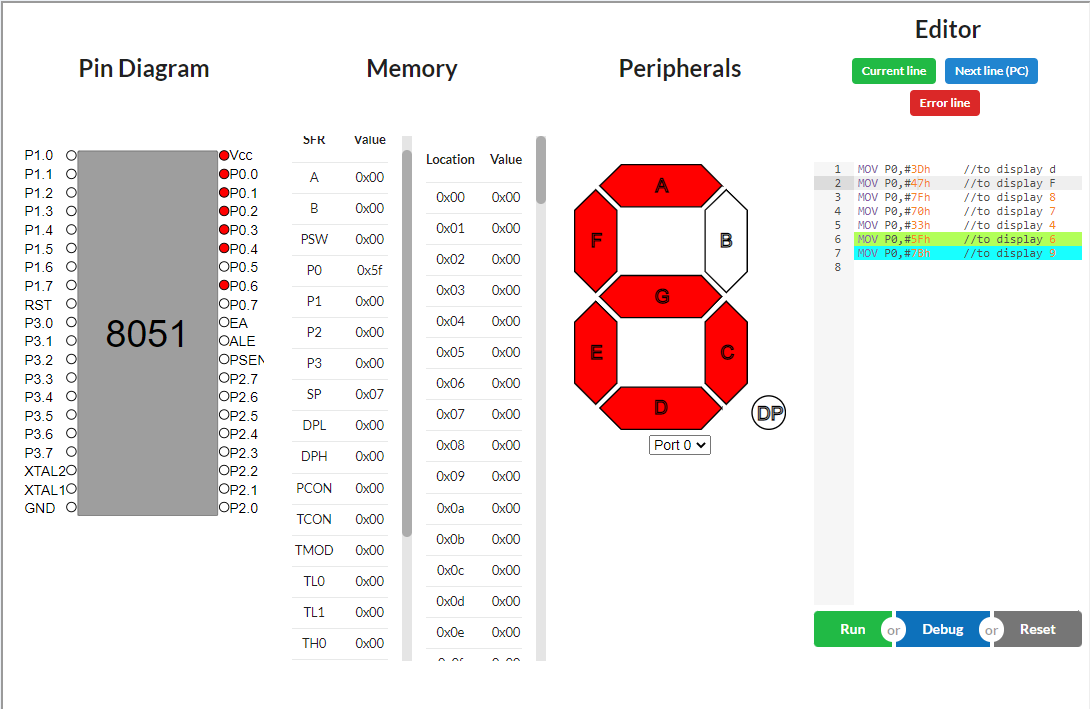
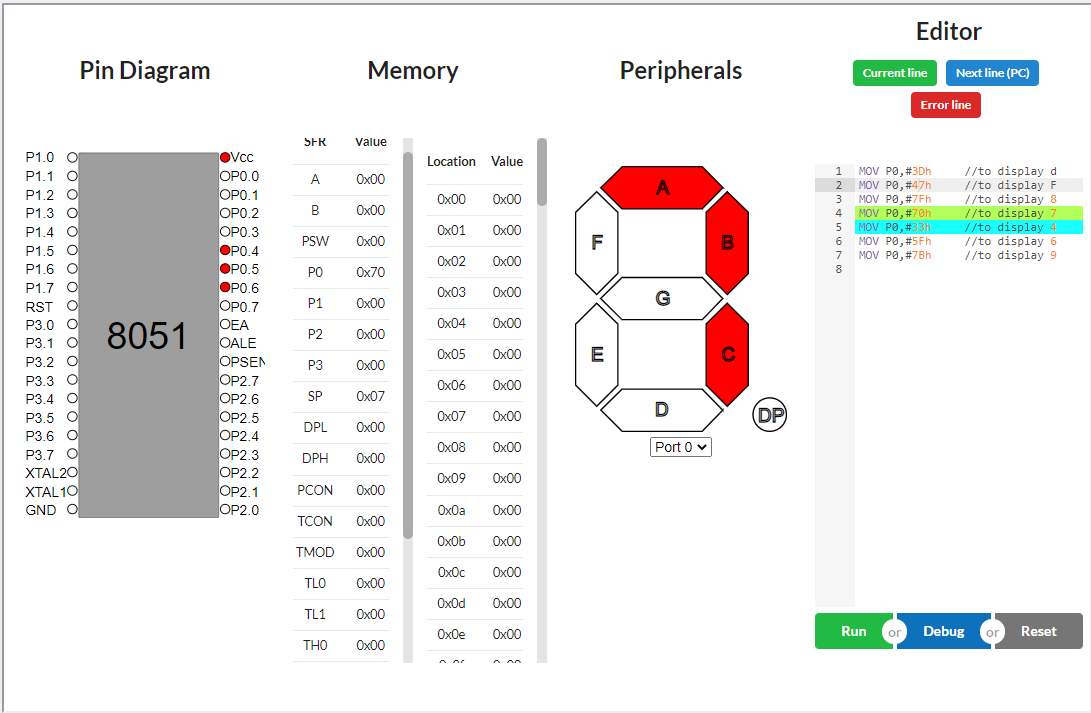
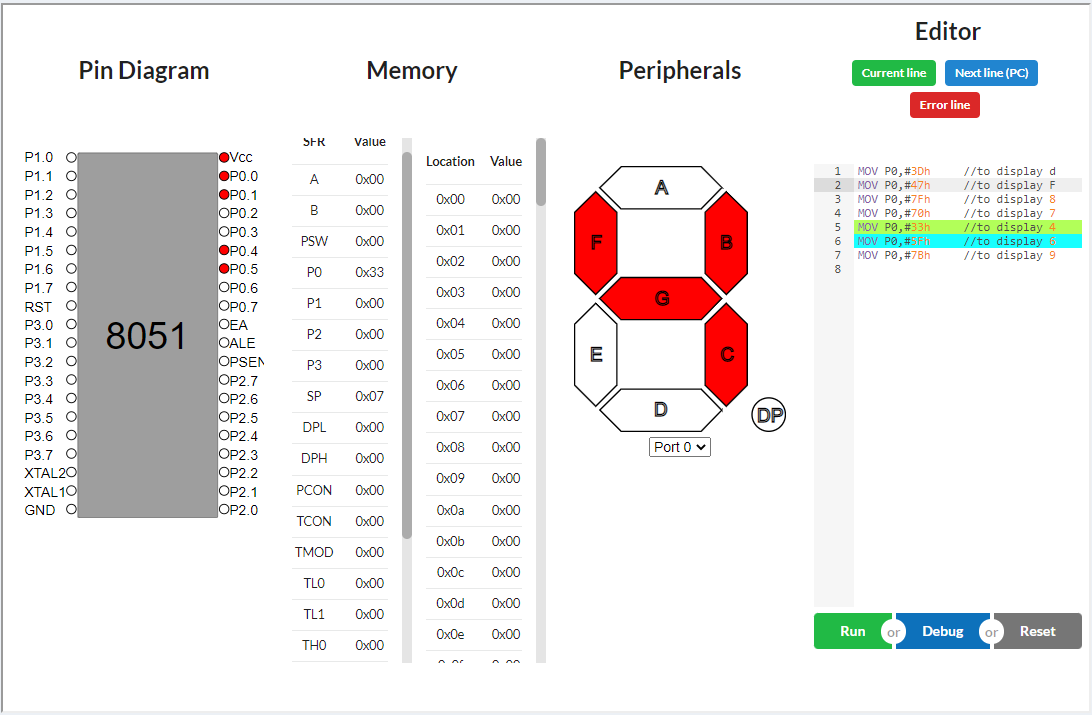
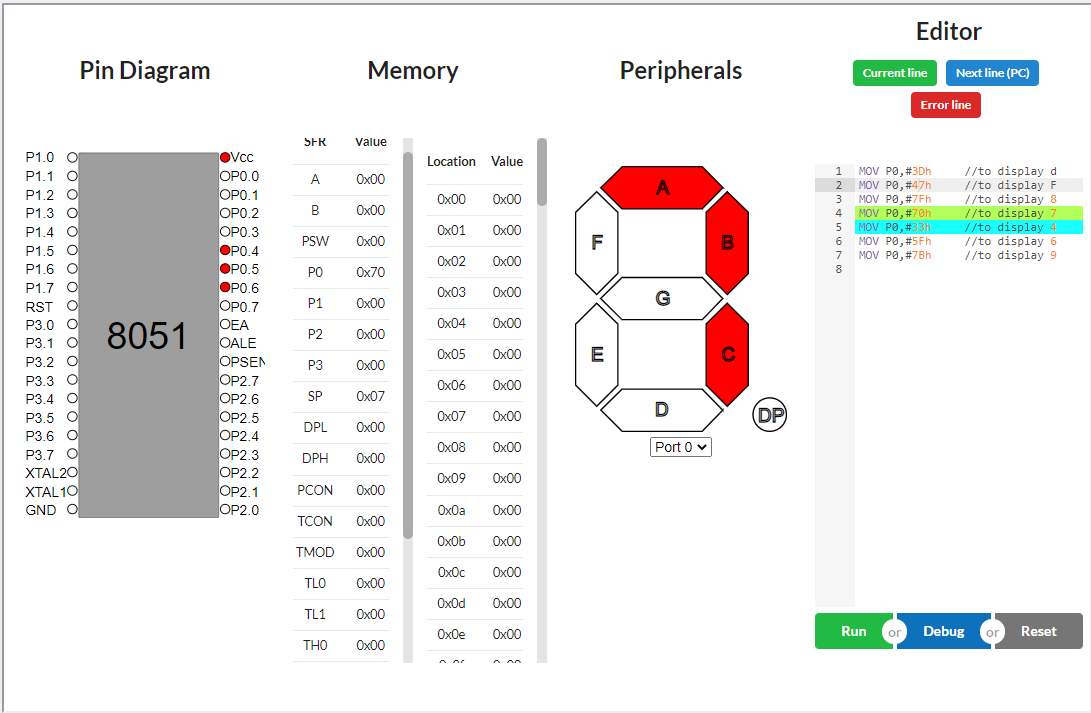
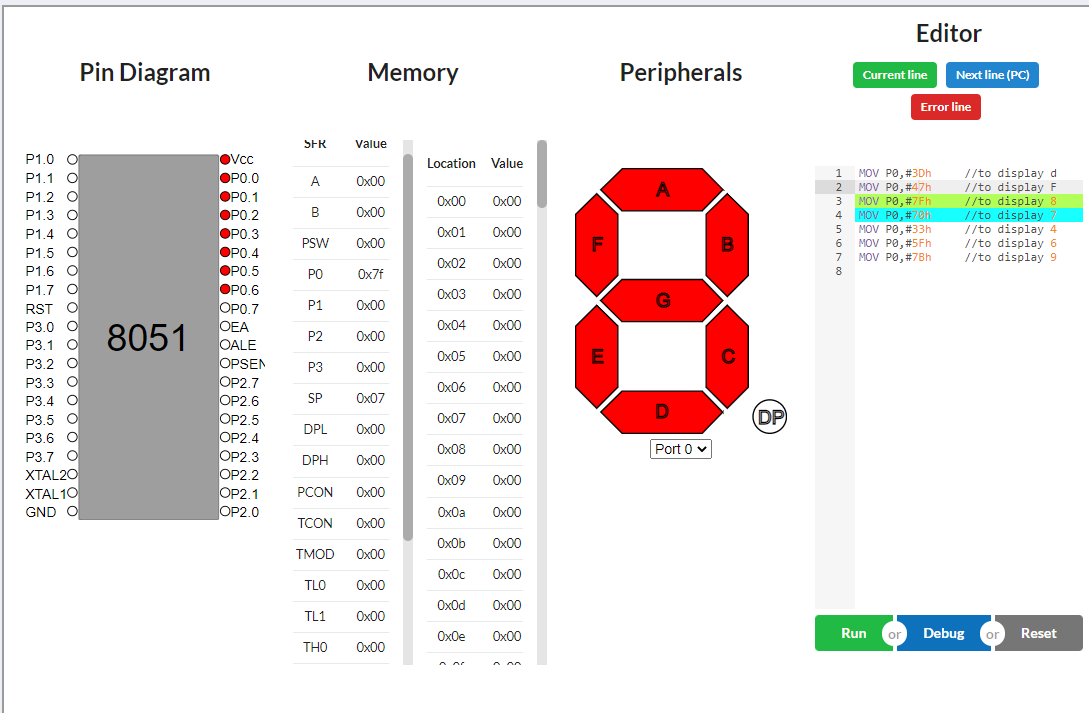
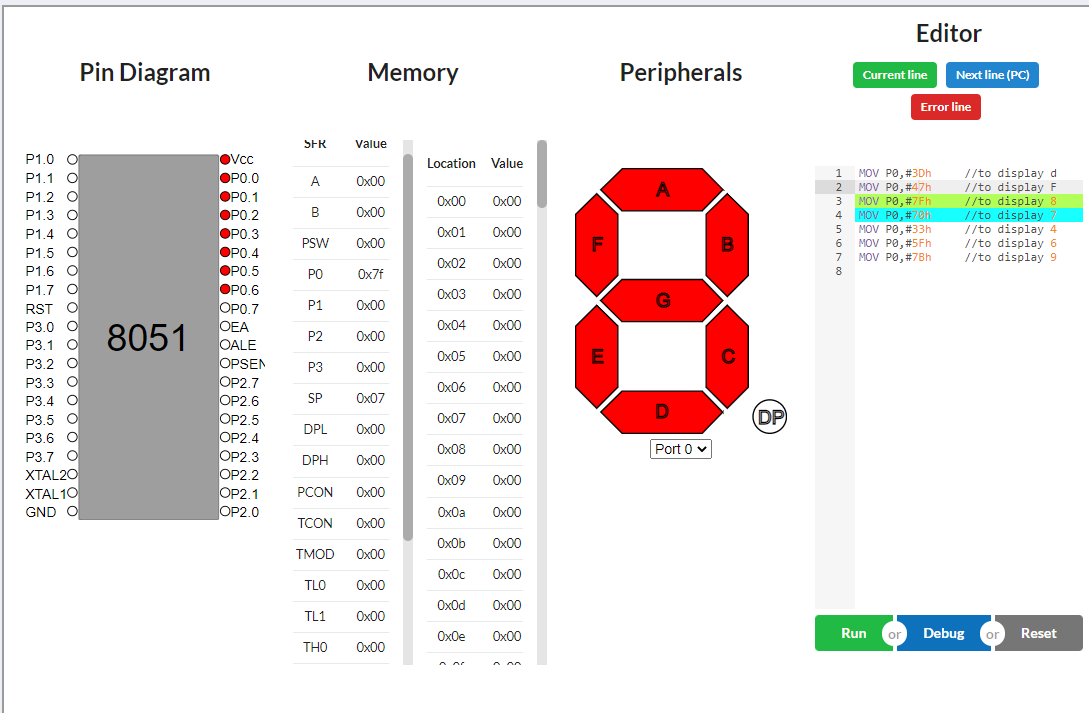
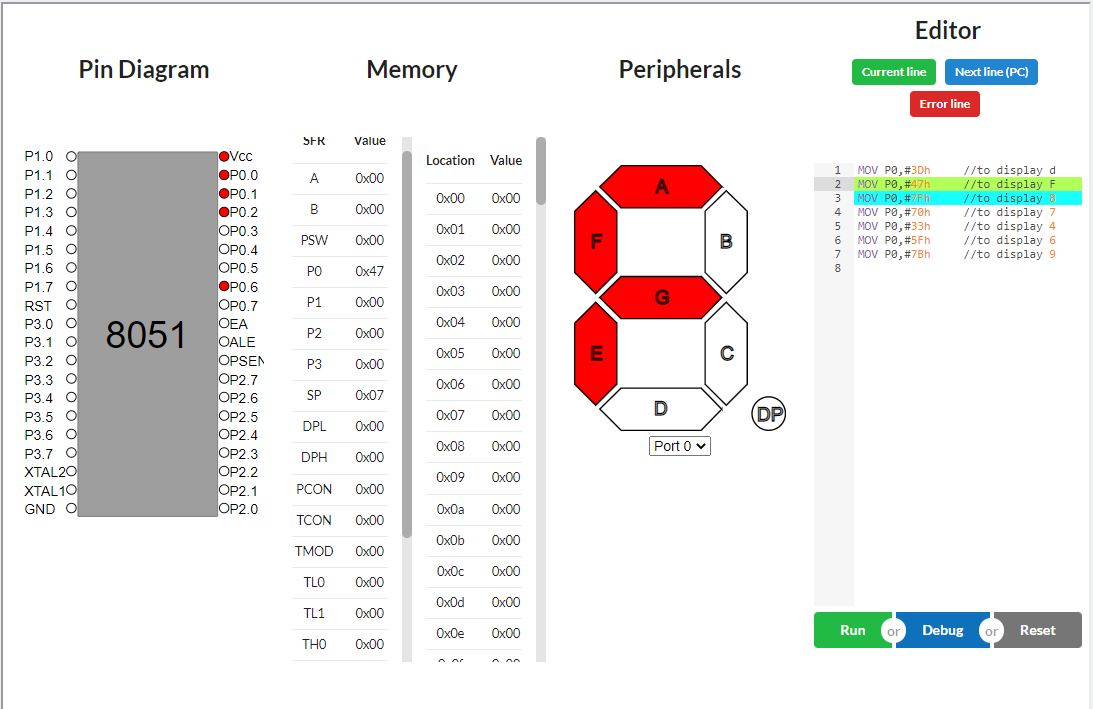
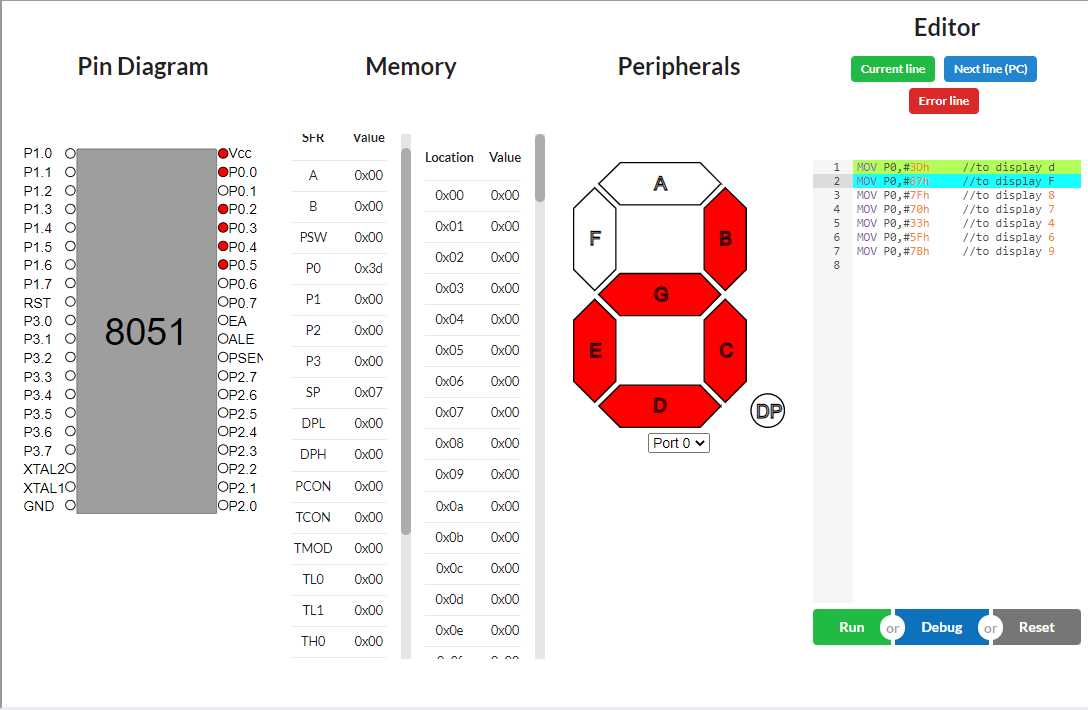
| **CHAR** | **A** | **B** | **C** | **D** | **E** | **F** | **G** | **HEX** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SAP ID = 60004190057** | | | | | | | | |
| **6** | 1 |  | 1 | 1 | 1 | 1 | 1 | 5F |
| **0** | 1 | 1 | 1 | 1 | 1 | 1 |  | 7E |
| **4** |  | 1 | 1 |  |  | 1 | 1 | 33 |
| **1** |  | 1 | 1 |  |  |  |  | 30 |
| **9** | 1 | 1 | 1 | 1 |  | 1 | 1 | 7B |
| **5** | 1 |  | 1 | 1 |  | 1 | 1 | 5B |
| **7** | 1 | 1 | 1 |  |  |  |  | 70 |
| **HEXADECIMAL = DF8874769** | | | | | | | | |
| **D** | 1 | 1 | 1 | 1 | 1 | 1 |  | 7E |
| **F** | 1 |  |  |  | 1 | 1 | 1 | 47 |
| **8** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 7F |
| **7** | 1 | 1 | 1 |  |  |  |  | 70 |
| **4** |  | 1 | 1 |  |  | 1 | 1 | 33 |
| **6** | 1 |  | 1 | 1 | 1 | 1 | 1 | 5F |
| **9** | 1 | 1 | 1 | 1 |  | 1 | 1 | 7B |
| **NAME = JUNAID** | | | | | | | | |
| **J** |  | 1 | 1 | 1 |  |  |  | 38 |
| **U** |  | 1 | 1 | 1 | 1 | 1 |  | 3E |
| **N** | 1 | 1 | 1 |  | 1 | 1 |  | 76 |
| **A** | 1 | 1 | 1 |  | 1 | 1 | 1 | 77 |
| **I** |  | 1 | 1 |  |  |  |  | 30 |
| **D** |  | 1 | 1 | 1 | 1 |  | 1 | 3D |

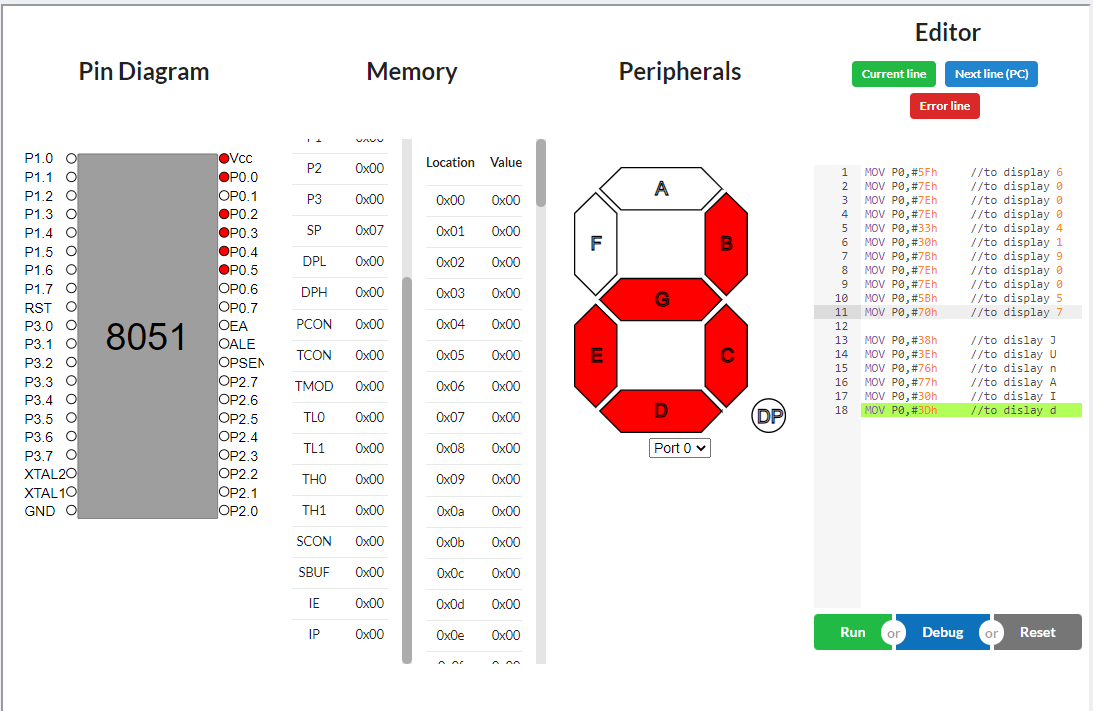
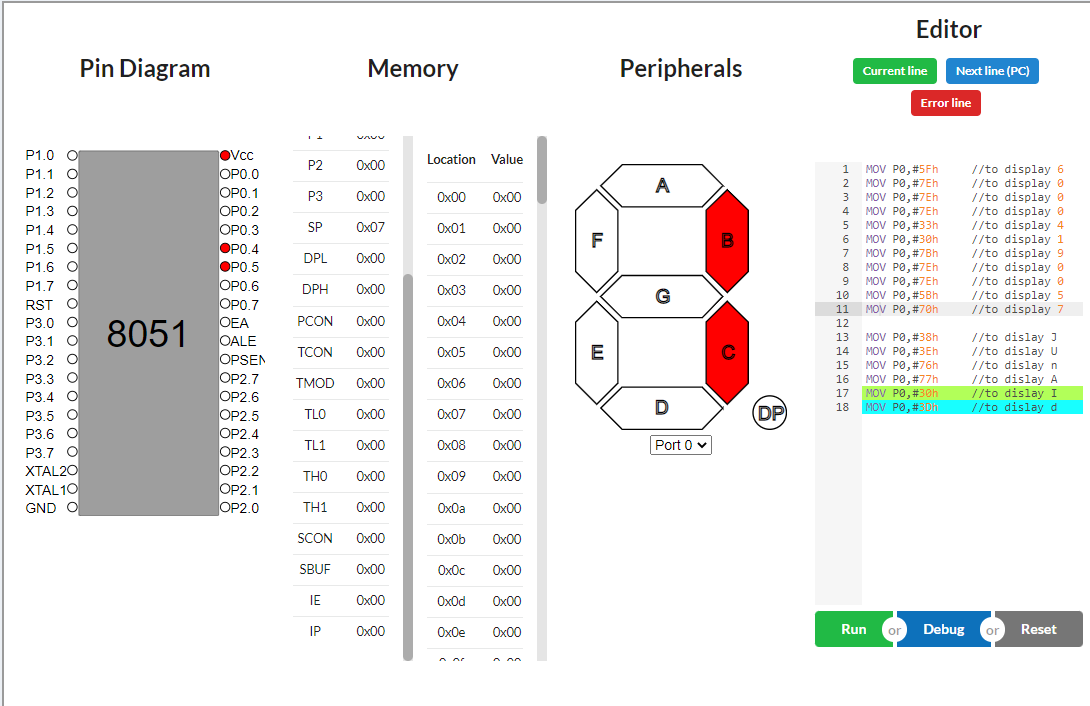
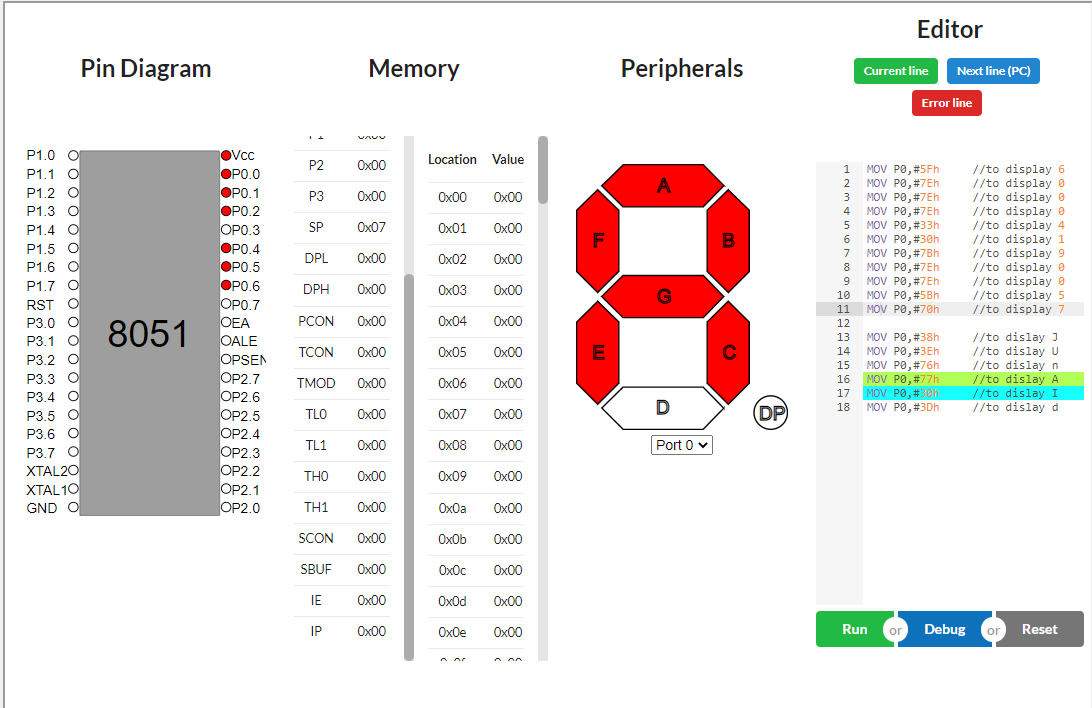
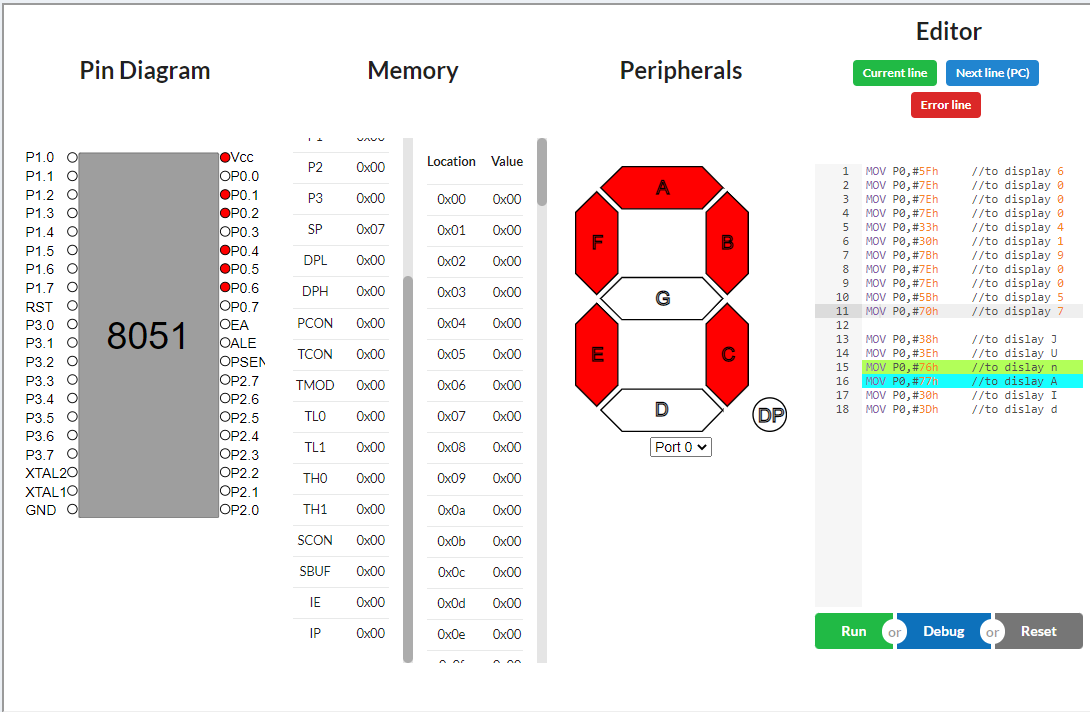
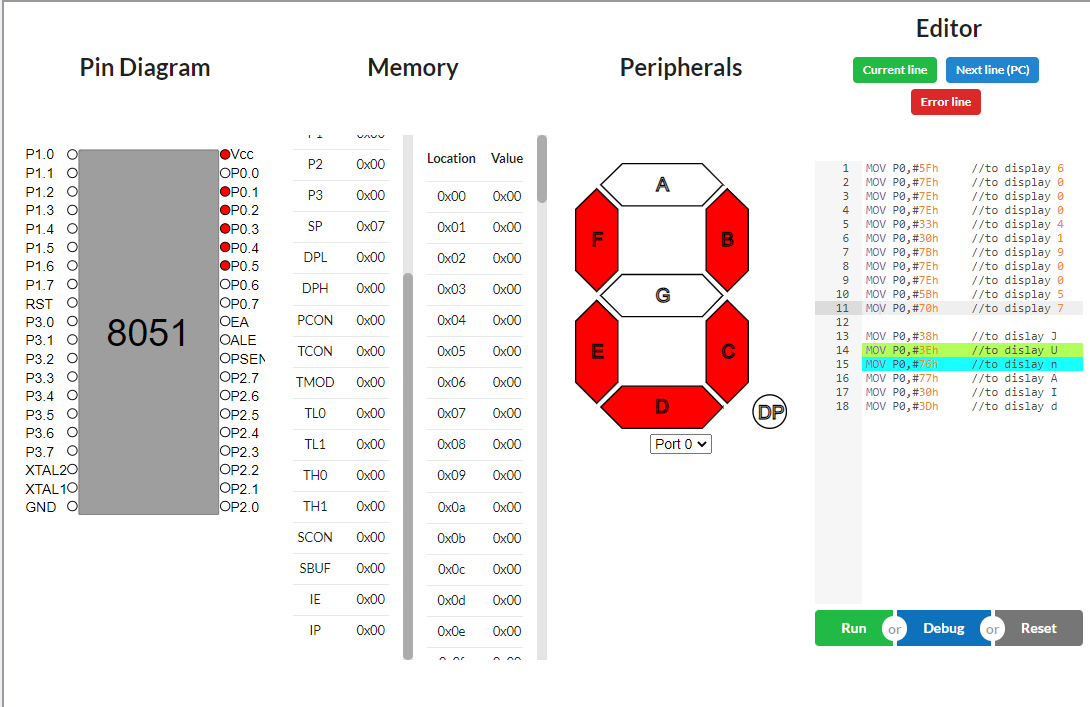
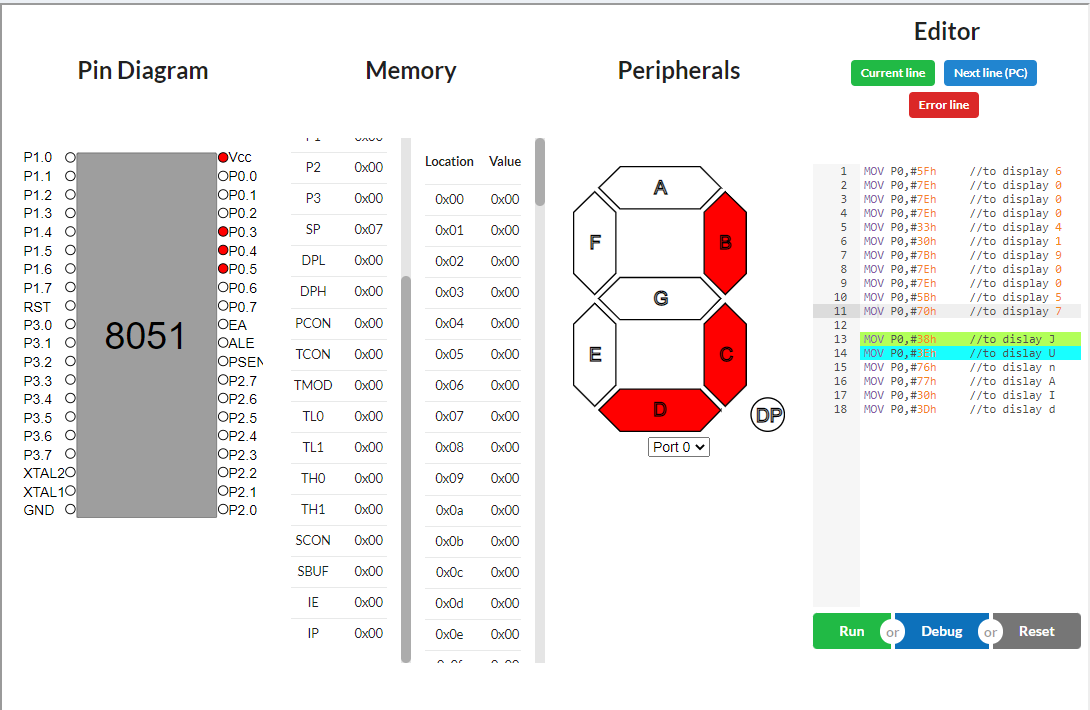
CODE:

| MOV P0,#5Fh //to display 6 MOV P0,#7Eh //to display 0 MOV P0,#7Eh //to display 0 MOV P0,#7Eh //to display 0 MOV P0,#33h //to display 4 MOV P0,#30h //to display 1 MOV P0,#7Bh //to display 9 MOV P0,#7Eh //to display 0 MOV P0,#7Eh //to display 0 MOV P0,#5Bh //to display 5 MOV P0,#70h //to display 7  MOV P0,#3Dh //to display d MOV P0,#47h //to display F MOV P0,#7Fh //to display 8 MOV P0,#7Fh //to display 8 MOV P0,#70h //to display 7 MOV P0,#33h //to display 4 MOV P0,#70h //to display 7 MOV P0,#5Fh //to display 6 MOV P0,#7Bh //to display 9  MOV P0,#38h //to display J MOV P0,#3Eh //to display U MOV P0,#76h //to display n MOV P0,#77h //to display A MOV P0,#30h //to display I MOV P0,#3Dh //to display d |
| --- |

**OUTPUT**:







**CONCLUSION:** We learn about 7 segment display and simulations using the 8051 Microcontroller. We then simulated a few examples by writing the code.