

Write the ASM codes for the following instructions:

1. Read two data (1 byte) from the last and first memory location of MPU 8086 and store their sum in the last even memory location of MPU 8086.
2. Read two numbers (1 byte) from the first two memory location of a fixed port address and write the result in the last memory location of MPU 8086.
3. Read two data (1 byte) from the last and first memory location of MPU 8086 and write their sum in the last even memory location of a fixed port address.
4. Read two data (1 byte) from the last and first memory location of a variable port address and write their sum in the last even memory location of a fixed port address.
5. Read three numbers (1 byte) from three consecutive memory locations of MPU 8086 and write the result in a variable port address (any).
6. Read two data from the last and first memory location of MPU 8086 and store their sum in the last even memory location of MPU 8086, if the sum is greater than 5.
7. Read two data from the last and first memory location of MPU 8086 and perform the subtraction operation. Store the result in the last memory location of a variable port address if the result is positive, otherwise store in the first memory location of a fixed port address.
8. Read two positive numbers from two consecutive memory location of MPU 8086 and store the larger number in a fixed port address.
9. Read the data using AL register and check whether it's positive or negative.
10. Read the data using CL register and check whether it's odd or even.
11. Read the data using AL register and check whether it's palindrome or not.
12. Read the data using BL register from the last memory location of MPU 8086 and count the number of 1's in BL register.
13. Read the data using BL register from the last memory location of MPU 8086 and count the number of 0's in BL register.
14. Multiply the number stored in AL register by 2, 4, 8, and 16.
15. Divide the number stored in AX register by 2, 4, 8, and 16.
16. Check whether the lower nibble of AL is greater than 9 or not.
17. Check whether the higher nibble of CH is greater than 9 or not.
18. Set the MSB/LSB of BL register.
19. Clear the MSB/LSB of CL register.
20. Change the sign bit of the data stored in BL register.
21. Change the bits of the AL register those are evenly indexed (0-6), given that indexing start from 0.
22. Change the bits of the BL register those are oddly indexed (1-7), given that indexing start from 0.
23. Reverse the data or number stored in AL/AX register.
24. Clear the CL register using XOR op-code.
25. Clear the even indexed bits of AL register.
26. Set the odd indexed bits of CL register.
27. Exchange the nibbles of AL register without using XCHG op-code.
28. Check whether lower nibble of AL register is equal to higher nibble of AL.
29. Add two bytes, if the result is zero then store the result in DL register, otherwise store it in DH register.
30. Add two bytes, if the result contains even number of 1's then store the result in DL register, otherwise store it in DH register.
31. Read two words and add. Store the result in DX register if unsigned overflow occurs, otherwise store it in CX register.
32. Read two words and add. Store the result in DX register if signed overflow occurs, otherwise store it in CX register.
33. Add two bytes and check the result stored in AL for an auxiliary carry or half carry.
34. Read data (1 byte) from three consecutive memory locations of MPU 8086 and store the largest number (unsigned/signed) in DL register.
35. Read data (1 byte) from three consecutive memory locations of MPU 8086 and store the smallest number (unsigned/signed) in DL register.
36. Convert the ASCII value of the decimal digits (0-9) to the numerical value (0-9).