```c
//Tilak Poojary
//NNM24EE127
//20/8/2025
//EXP1B & To familiarize logical and arithmetic instructions Logical instructions


#include<reg51.h>
unsigned char idata *internalmemory = 0x60, *R4 = 0x04, *R2=0x02;
unsigned char variable, temp, temp1;
void main(void)
{
  //LOGICAL AND OPERATION

  ACC = 0x78;                       //storing data in accumulator
  *internalmemory = ACC & 0x0F;     //in internal memory storing the lower nibble of the data in accumulator using logical AND opertaor ACC & 0X0F GIVES 0X08 AS RESULT WHICH IS STORED IN INTERNAL MEMORY AS IT IS THE LOWER NIBBLE.

  //LOGICAL OR

  *internalmemory++;                //incremented memory location from 0x60 to 0x61

  ACC = 0xF8;                        //storing value or data 0xF8 that is F8 in accumulator
  *R2=ACC;
  *internalmemory = ACC | 0XFF;     //performed logical OR and storing the value after operation to internal memory 0x61. The value is 0xFF

  //LOGICAL XOR

   *internalmemory++;                //incremented memory location from 0x61 to 0x62
     *R4=0x55;                       //storing value or data 0x55 in the register R4
  *internalmemory = *R4 ^ 0xFF;      //complemented the bits in register R4 using XOR logical operation and we stored the result that is 0xAA in internal memory 0x62

  //LOGICAL NOT

  *internalmemory++;                //incremented memory location from 0x62 to 0x63
  ACC=0x55;                         //storing value 0x55 in the accumulator
  *internalmemory = ~ ACC;          //did complement using LOGICAL NOT to the value in Accumulator and stored i internal memory 0x63 the result 0xAA

  //SHIFT OPERATOR

  *internalmemory++;                        //incremented memory location from 0x63 to 0x64
  ACC=0x78;                                 //storing value 0x78 in accumulator
  *internalmemory = ACC & 0xF0;             //doing logical AND operatio on Accumulator and 0xF0 to get the higher nibble and store the result 0x70 in internalmemory 0x64
  *internalmemory = *internalmemory>>4;     //using right shift shifted the value 70 to 07 by shiffting 4 tims towards right and is stored in intrnal memory 0x64 itself by overwriting

  //SWAPPIG of nibble of variable and store result in internal memory

  *internalmemory++;                       //incrementing the internal memory location from 0x63 to 0x64
  variable=0x48;                           //storing value 0x48 to the variable
  temp=(variable & F0)>>4;                  //variable value 0x48 is anded with F0 to get only higher nibble 0x40 then shiffted 4 times right side to get it at lower nibble which will be 0x04 and stored in temp
  temp1=(variable & 0F)<<4;                 //variable value 0x48 is anded with 0F to get only lower nibble 0x08 then shiffted 4 times left side to get it at higher nibble which is 0x80 and stored in temp1
  *internalmemory=temp|temp1;              //temp is ored with temp1 to get 0x84 that will be stored in internalmemory

}
```