

1. It's better to clear all the RAM in the program start. As you emulate in the ICE. If the RAM not used. It will be easy to find out the value still keep "0h".
2. Be careful not to over used the data area which is used as LCD display RAM
3. Don't over use the data RAM with the stack. All stacks are shared with RAM. So as arrange the RAM register must be every careful about do not over used with the stack. EMC suggest set the stack point as "0Ch" in the program initialization that will give programmer the best utilized range. By the way . Check the SPW as more as possible will let you know the stack # you used. If you clear all the memory in the program start. You could also check the stack memory to find out the most stack # you used. (the stack that is not used will keep "0".)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	Stack 0				Stack 1				Stack 2				Stack 3			
10	Stack 4				Stack 5				Stack 6				Stack 7			
20	Stack 8				Stack 9				Stack 10				Stack 11			
30	Stack 12				TA(L,M,H)				TB(L,M,H)				DP(L,M,H)			SPW

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20																
30																
40	Stack 0				Stack 1				Stack 2				Stack 3			
50	Stack 4				Stack 5				Stack 6				Stack 7			
60	Stack 8				Stack 9				Stack 10				Stack 11			
70	Stack 12				TA(L,M,H)				TB(L,M,H)				DP(L,M,H)			SPW

[illegible]

50															
60															
70															
80															
90															
A0															
B0															
C0	Stack 0				Stack 1				Stack 2				Stack 3		
D0	Stack 4				Stack 5				Stack 6				Stack 7		
E0	Stack 8				Stack 9				Stack 10				Stack 11		
F0	Stack 12				TA(L,M,H)				TB(L,M,H)				DP(L,M,H)		
													SPW		

Over 256 nibble RAM only could be shared with LCD display RAM

4. Be careful to switch the RAM bank. User must keep in mind which bank you are now. EMC suggest keep in the bank 0 always. If you switch to other bank, you must return to bank 0 as you finish all the things you want to do in bank x.
5. As you switch to bank x for example. But the interrupt happen. And you must to use some register in some fixed RAM bank. This will happen random or unpredictable. So we suggest you use the following description in the start and end of interrupt call.

```

***** interrupt subroutine example *****
      ORG 000h
FLAG      RES    01h;
P93_FLAG  EQU    (FLAG+0)*10h+1;
;
      ORG 010h
ST_HL     RES    02h;
ST_ACC    RES    01h;
;
;
; ***** PUSH ACC / HL REGISTER VALUE INTO ST_ACC AND ST_HL
;
PUSH_REG:  CLR P93_F/10H,P93_F;
           TFP P9,3;
           B PU_R;
           SET P93_F/10H,P93_F;
PU_R:      CLP P9,3;
           STA ST_ACC;
           EXHL ST_HL;
           RET;
;

```

```

;
;
; ***** POP ACC / HL REGISTER VALUE FROM ST_ACC AND ST_HL
;
POP_REG:    LDA ST_ACC;
            LDHL ST_HL;
            TF P93_F/10H,P93_F;
            B PO_R;
            SEP P9,3;
PO_R:      RET;
;
;

; Interrupt service routine for timer A
TAI:      LCALL PUSH_REG;
            :
            LCALL POP_REG;
;
*****

```