

```

//[サンプル]

// メイン・ルーチン
// *** prototype ***
void sub();

// *** extern ***
extern void psr_write( unsigned char ucPSR );
extern unsigned char psr_read(void);

// *** global ***
volatile unsigned char uc_Val;
volatile unsigned char uc_Read_IL;

/*****
* main
*   Type :      int
*   Ret val :    0
*   Argument : void
*   Function :  main program.
*****/
main()
{
    // enable interrupt
    uc_Val = psr_read();
    uc_Val |= 0x10;
    psr_write( uc_Val );

    sub();

    // disable interrupt
    uc_Val = psr_read();
    uc_Val &= 0xef;
    psr_write( uc_Val );

    sub();

    // read IL
    uc_Val = psr_read();
    uc_Read_IL = ( uc_Val >> 5 );

    sub();

    // set IL 2
    uc_Val = psr_read();
    uc_Val &= 0x1f;
    uc_Val |= ( 2 << 5 );
    psr_write( uc_Val );

    return 0;
}

//PSR アクセス サブ・ルーチン
/*****
;
;
;   psr_write
;
;   PSR write function
;
;   IN
;       %r0    written value
;   OUT
;       None
;
; *****/
;
.text
.align 1
.global psr_write
.type psr_write, @function

```

```
psr_write:
    ld.b [%sp+3], %r0
    reti
```

```
*****
;
;
; psr_read
;
; PSR read function
; This function calls int_psr_read( Vector Table No.31 ).
;
; IN
;     None
; OUT
;     %r0    read value
;
; *****
```

```
.text
.align 1
.global psr_read
.type psr_read, @function
```

```
psr_read:
    int 31
    ret
```

```
*****
;
;
; int_psr_read
;
; PSR read interrupt function
; This function is called by psr_read.
; And this function is allocated to Vector Table No.31.
;
; IN
;     None
; OUT
;     %r0    read value
;
; *****
```

```
.text
.align 1
.global int_psr_read
.type int_psr_read, @function
```

```
int_psr_read:
    ld.ub %r0, [%sp+3]
    reti
```