我使用nios2開發的lib架構如下:

主程式為 IRIS\_CPU\_V02.c，主程式include一個IRIS\_V02.h,

IRIS\_V02.h裡include其他的.h如下，每個.h有不同的功能:

#include "nios2\_var\_addr.h"

#include "common.h"  
#include "output\_fn.h"  
#include "output\_mode.h"

#include "uart.h"

#include "adda\_config.h"

#include "memory\_manage.h"

#include "eeprom\_v2.h"  
#include "adxl357.h"  
#include "ads122c04\_se.h"

我的主程式裡有一個while loop，會根據uart輸入的指令來決定輸出的函數，目前我在common.h裡宣告了一個函數指標:  
/\*\*\* output function type delaration \*/

typedef void (\*fn\_ptr) (cmd\_ctrl\_t\*, my\_sensor\_t, fog\_parameter\_t);

然後在主程式定義函數如下:

fn\_ptr output\_fn = acq\_rst;

之後在while loop放置 output

while(1) {

get\_uart\_cmd(readDataDynamic(&try\_cnt), &my\_cmd);

cmd\_mux(&my\_cmd);

fog\_parameter(&my\_cmd, &fog\_params);

output\_mode\_setting(&my\_cmd, &output\_fn, &auto\_rst);

if (trigger\_sig == 1) {

g\_time[0] = get\_timer\_int();

update\_sensor\_data(&sensor\_data);

g\_time[1] = get\_timer\_int();

output\_fn(&my\_cmd, sensor\_data, fog\_params);

trigger\_sig = 0;

}

}

目前我是先把output\_fn宣告在output\_fn.h，定義在output\_fn.c，

Output\_fn.h:

#ifndef \_\_OUTPUT\_FN\_H

#define \_\_OUTPUT\_FN\_H

#include "stdio.h"

#include <stdlib.h>  // malloc(), free()

#include <string.h>  // memcpy()

#include "common.h"

#include "adxl357.h"

#define DLY\_NUM 10000

extern alt\_u32 g\_time[12];

/\*\*\* output function type delaration \*/

// typedef void (\*fn\_ptr) (cmd\_ctrl\_t\*, my\_sensor\_t, alt\_u8\*, fog\_parameter\_t);

typedef void (\*fn\_ptr) (cmd\_ctrl\_t\*, my\_sensor\_t, fog\_parameter\_t);

void acq\_rst (cmd\_ctrl\_t\*, my\_sensor\_t, fog\_parameter\_t);

void acq\_fog (cmd\_ctrl\_t\*, my\_sensor\_t, fog\_parameter\_t);

void acq\_imu (cmd\_ctrl\_t\*, my\_sensor\_t, fog\_parameter\_t);

#endif /\* \_\_OUTPUT\_FN\_H \*/

但為了後續方便性，我想要將各個函數另外定義在對字的.h裡，

比如void acq\_imu (cmd\_ctrl\_t\*, my\_sensor\_t, fog\_parameter\_t);我想放在一個acq\_imu.h，然後在output\_fn.h裡include acq\_imu.h。  
這樣可行嗎?