嵌入式系統實驗, Fall 2016
Lab 4 20161015

指導教授：王勝德 教授　　B03901023 電機三 許秉鈞

實驗目的

跟上次不同，本次使用shell command以及Kernel Module，分別做出LED blink以及hello world的範例。

問題討論

1. Please refer http://codeandlife.com/2012/07/03/benchmarking-raspberry-pi-gpio-speed/ to see the discussions about GPIO speed. What is your choice if you care most about the speed?

Python with RPi.GPIO is the fastest of these, shell is only usable for simple automation tasks, but Perl with BCM 2835 bindings comes close.

2. Why do we need compile the Linux Kernel source at least once before we can build our own kernel module?

因為可以在linux kernel 加入system call，而且這樣才能build更多的api在我們的module之上，（參考老師所說的 hierarchy）。

3. Explain the meaning or purpose of the four values output by the command "cat /proc/sys/kernel/printk".

```
# cat /proc/sys/kernel/printk
7 4 1 7
```

這表示的是原本printk的訊息優先權(printk level), 每個值都有自己的意義.
Messages with a higher priority than console_loglevel will be printed to the console.
Messages without an explicit priority will be printed with priority default_message_level.
Minimum_console_loglevel is the minimum (highest) value to which console_loglevel can be set.
default_console_loglevel is the default value for console_loglevel.

4. Name or illustrate some functionalities of Raspbian OS in the RPi that are implemented by kernel modules.

ref: http://www.makeuseof.com/tag/5-ways-new-raspbian-jessie-makes-raspberry-pi-even-easier-use/

- free operating system based on Debian optimized for the Raspberry Pi hardware.

- 可以支援GUI、Camera Module…etc

5. Explain the meaning of setting "CONFIG_XXX_YYY=m" and "CONFIG_YYY_ZZZ=y" in Linux kernel configurations file .config.

ref: http://tldp.org/HOWTO/SCSI-2.4-HOWTO/kconfig.html

configure 是將 linux kernel 調整成適合目標系統使用的手段。
An option will either indicate some driver is **built into the kernel ("=y")** or will **be built as a module ("=m")** or is not selected. The unselected state can either be indicated by a line starting with "#" (e.g. "# CONFIG_SCSI is not set") or by the absence of the relevant line from the .config file.

—-

[LAB 1-1]

### Why execute the shell command in another terminal?

`ps -eLf | grep thread-ex`

-e: 印出其他user的processes
-L: 挑出特定keyword
-f: 印出uid, pid, parent pid, recent CPU usage, process start time, controlling tty, elapsed CPU usage, and the associated command.

對照關係為：

```
   UID    PID  PPID   LWP  C NLWP STIME   TTY        TIME         CMD
   user  7526  5596  7528  0    3 15:44 pts/0    00:00:00 ./thread-ex
```

ref: https://kb.iu.edu/d/afnv

在另一個terminal執行`ps`的原因，是為了檢查user UID是否有變化。`BLABLABLA| grep thread-ex`則是為了將前面的輸出，grep出含有`thread-ex`的行，其他都捨棄掉，所以才需要加上Pipe後面那串。

### What is the purpose of the command? Note the PID, LWP and NLWP fields

> LWP: Light Weight Process (LWP is the thread id)

`7526`一欄為pid, `5596`一欄為ppid, 後面的`7526`一欄為LWP。
這個指令的目的，是為了檢查`uid`, `pid`以及`parent pid`是否有變化。從結果中看出，除了`grep`指令，`pid`, `ppid`, `NLWP`均同、但`LWP`不一樣。原因是因為`7526`, `7527`代表不同的thread id。`NLWP`代表的是目前有幾個threads，因此數量為3（不包含`PID=7530`那一行）。

### Explain the organization of a Makefile

`makefile`

```
   #The following four macros should be defined.
   TARGET1=thread-ex
   OBJECT1=thread-ex.o
   CC=gcc
   LD_FLAGS= -lpthread
   C_FLAGS=
   # end of user configuration
   all: $(TARGET1)
   %.o : %.c
          $(CC) $(C_FLAGS) -c $< # 真正跑的cmd是這行，其他都是代號
   .PHONY: clean # PHONY代表 偽目標，也就是這些檔案若存在，就clean；不存在也沒關
          係，因為本來就是偽的。
   clean :
          rm -f $(OBJECT1) $(OBJECT2)
```

首先，我已經把`TARGET2, OBJECT2, g++, CXX_FLAGS`都移除，因為這次用C語言寫、不需要管`g++`。

`LD_FLAGS`加上 -lpthread 告訴它要去連結 libpthread，但這次程式不需要，因此在輸出object時我只使用`C_FLAGS`。

### What are the main concepts behind in make mechanisms?

把每個指令打包成一鍵搞定，以處理大型、但編譯或link形式相似的程式。

[LAB 1-2]

### What is the first system call used?

其實就是第一行：

```
   execve("/bin/cat", ["cat", "/proc/cpuinfo"], [/* 69 vars */]) = 0
```

[LAB 3]

1. When does a circuit need a pull-up / down resistor?

當某輸入埠未連接設備或處於高阻抗的情況下，用於保證輸入訊號為預期邏輯電平。

2. What does the C keyword *volatile* mean?

告訴compiler 在***volatile*** 之後所接的變數具有揮發性的特性, 所以與它有關的程式碼請不要作最佳化動作。簡單來說，這個keyword 要加在「希望他因時改變、因此多次賦值、而且不希望編譯器偷懶優化變成只賦值一次。」這樣的情形。

3. How do you implement *winingPiISR()* if you are a wiringPI developer?

This function registers a function to received interrupts on the specified pin. 也就是說當interrupt發生時，function就會被呼叫。這個function要設定夠高的priority、並且讓他不斷擁有full access權限、已獲得所有global variable、才不會接到interrupt時措手不及。