

嵌入式系統期中考重點整理

makefile 寫法(1)：單純兩個檔案轉譯

備註：

hello.c 與 message.c 為上課使用檔案，屆時考試請注意看題目要求檔案名稱

```
CROSS_PATH = $(HOME)/nds32le-elf-
CROSS_COMPILE = $(CROSS_PATH)gcc
CFLAGS = -c -g
LDFLAGS = -mvh -mcpu=n801 -mmodel=medium -g
LDFLAGS += -lm -Tnds32.ld
SRCS = hello.c message.c
OBJS = $(SRCS:.c=.o)

hello.adx : $(OBJS)
    $(CROSS_COMPILE) $(LDFLAGS) $^ -o $@
.c.o :
    $(CROSS_COMPILE) $(CFLAGS) $^ -o $@
```

makefile 寫法(2)：一個 c 檔案，一個 s 檔案，互相轉譯

備註：

copy.c 與 BlockCopy.s 為上課使用檔案，屆時考試請注意看題目要求檔案名稱

```
CROSS_PATH = $(HOME)/nds32le-elf-
CROSS_COMPILE = $(CROSS_PATH)gcc
CFLAGS = -c -g
LDFLAGS = -mvh -mcpu=n801 -mmodel=medium -g
LDFLAGS += -lm
ASFLAG = -EL -x assembler-with-cpp
CSRCS = copy.c
SSRCS = BlockCopy.s
OBJS = $(SRCS:.s=.o)
OBJS += $(SRCS:.c=.o)

SteCopy.adx : $(OBJS)
    $(CROSS_COMPILE) $(LDFLAGS) $^ -o $@
.c.o :
```

```
$(CROSS_COMPILE) $(CFLAGS) $^ -o $@  
.s.o :  
$(CROSS_COMPILE) $(ASFLGS) $(CFLAGS) $^ -o $@
```

makefile 寫法(3)：數學三角形，三個檔案互相轉譯

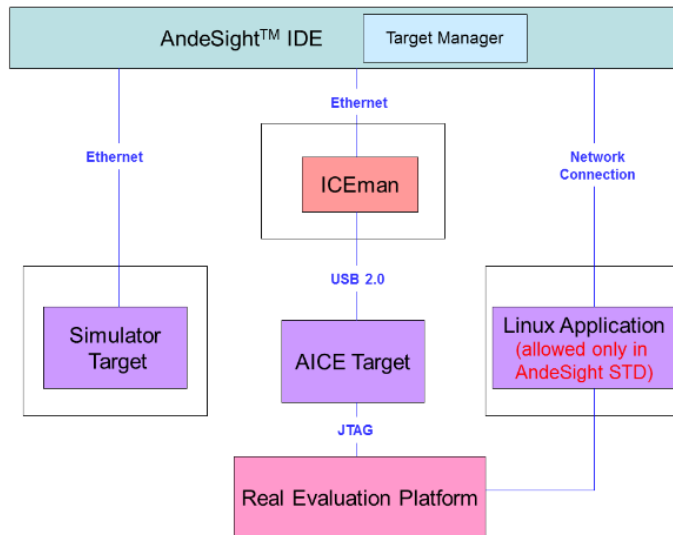
備註：

main.c 與 sidelength.c 與 area_cal.c 為上課使用檔案，屆時考試請注意看
題目要求檔案名稱

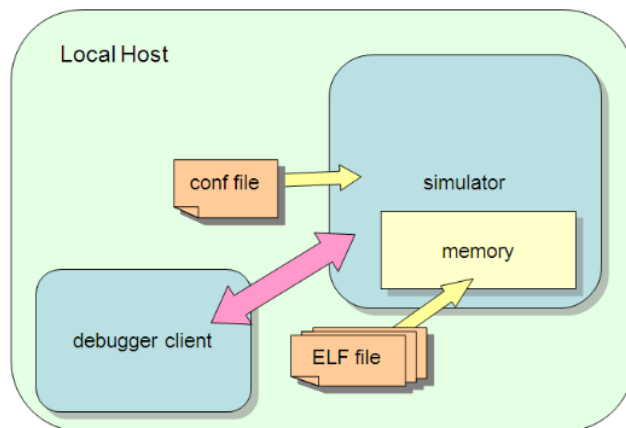
```
CROSS_PATH = $(HOME)/nds32le-elf-  
CROSS_COMPILE = $(CROSS_PATH)gcc  
CFLAGS = -c -g -mcpu=n801  
CFLAGS += -mcpu=n801  
LDFLAGS = -mvh -mcpu=n801 -mcmmodel=medium -g  
LDFLAGS += -lm  
ASFLAG = -EL -x assembler-with-cpp  
SRCS = main.c sidelength.c area_cal.c  
OBS = $(SRCS:.c=.o)  
  
Tria.adx : $(OBS)  
$(CROSS_COMPILE) $(LDFLAGS) $^ -o $@ $(LDFLAGS)  
.c.o :  
$(CROSS_COMPILE) $(CFLAGS) $< -o $@  
depend : $(SRCS)  
$(CC) -M $(CPPFLAGS) $^ > $@  
  
-include depend
```

***只要當 makefile 改變時，需在模擬器中先刪除*.o 與*.adx，再執行 makefile
編譯與執行**

Connection BTW HOST & Target

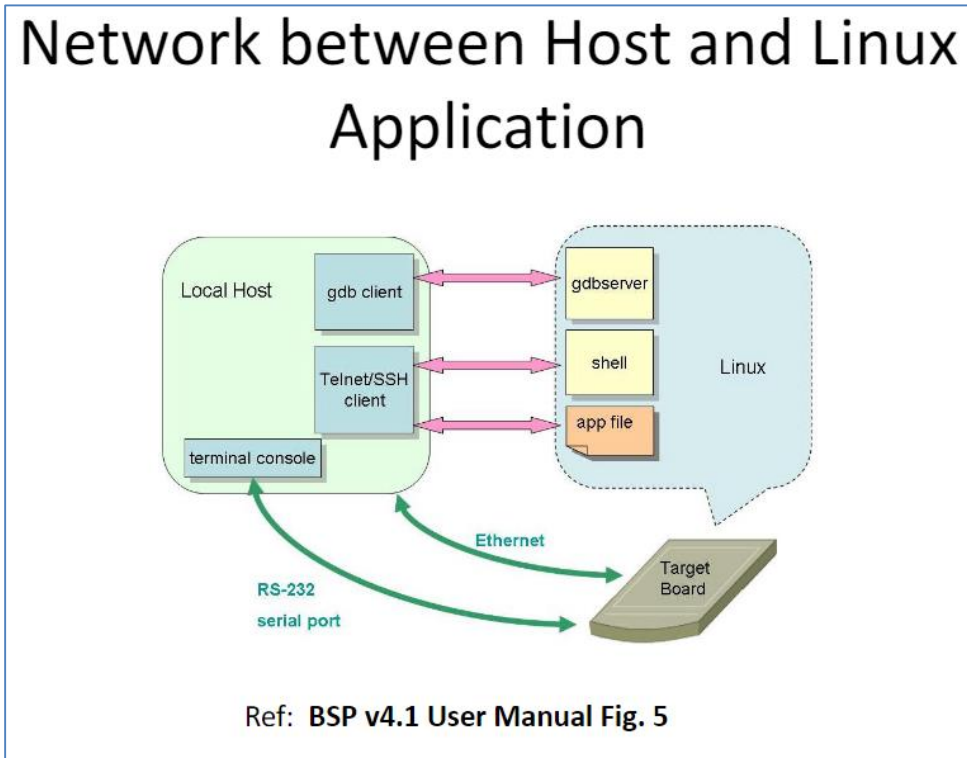


Network Between the Host and the simulator Target



Ref: BSP v4.1 User Manual Fig. 3

※網路與主機之間和 Linux 應用程式運作



1.1 什麼是嵌入式系統？

✚ 它具備以下功能：

- 用來**執行特定功能**
- 以**微電腦與周邊**構成核心
- 需要**嚴格的時序與穩定度**
- **全自動操作循環**

✚ 嵌入式系統裝置：**手機、機上盒、小烏龜(Linux)、交換器(Linux)**、Hy-wire
未來汽車、掃地機器人

Ch02_嵌入式系統開發流程

2.1 嵌入式系統軟體之開發環境

✚ 什麼叫跨平台開發？

Ans：是在**某一部電腦(或開發主機)**編寫程式碼/軟體，編寫完後編譯連結，**不是在原本的裝置執行，而是在另一裝置執行**

✚ 為什麼要採用跨平台開發這麼麻煩？

Ans：因為**嵌入式硬體資源不高，不適合用來開發，也沒編譯器與連結器**，所以把開發功能移到開發主機動作(PC/notebook)

- 介面由半導體廠用來測試用(test port)，測試到底是否正常
- 在嵌入式系統平台上執行由開發工具所建立出來的嵌入式軟體

- 在一般電腦上進行嵌入式系統軟體的開發程序
- 內部網路仿真器-ICE 為嵌入式系統與開發主機轉譯器，轉換成 CPU 能夠接受的指令。因半導體技術成熟，所以除錯功能也在內
 - ① ICE 是個翻譯者，抓值送回開發主機
 - ② 透過串列埠、網路埠、或內部電路仿真器來進行通訊，也會透過網路線 (10M bps)

2.2 嵌入式系統軟體之開發流程

- ✚ 以 C 或 C++ 撰寫，擁有大量程式庫
- ✚ 先將 C 轉換為組合語言，再將組合語言編譯成 obj code，目的為將高階程式語言轉化成機器碼。目前先把高階程式語言轉換成組合語言再轉換成機器碼，且沒有限定用幾種的程式碼(高階與低階都使用)
 - (1) 低階程式語言透過組譯器(Assembler)，翻譯成機器碼(CPU 看得懂的指令)
 - (2) 高階程式語言透過編譯器(Compiler)，翻譯成機器碼(CPU 看得懂的指令)
 - (3).o 檔不是只有機器碼，事實上包含原始程式碼的機器碼之外，也含有要做連結的資訊在裡面
 - (4).bin 檔是把不需要的連結資訊去掉
 - (5).adx 檔等於.exe 檔
 - (6).adx 檔透過 ICE 燒到系統(此系統需含作業系統)
- ✚ Java Virtual Machine 與 Linux 不同

Ch03_GNU

3.1 編譯 makefile 指令

- ✚ make -f makefile

3.2 路徑

- ✚ CROSS_PATH = \$(HOME)/nds32le-elf-
- ✚ CROSS_COMPILE = \$(CROSS_PATH)gcc