

嵌入式軟體設計與實習

Creator/PreSOC-PXA270 實驗模組

Lab Unit 04 – UART & Interrupt

教育部顧問室嵌入式軟體聯盟
<http://esw.cs.nthu.edu.tw>

Summary

Unit Code	Lab-PXA270-04-UART&Interrupt	Ver./Date	Version 1.0.50 2012-11-15
Category	Non-OS 進階		
Unit Title	Creator PreSOC/PXA270 UART & Interrupt		
Tasks	(1)瞭解Creator PXA270 EVM之UART運作與程式設計 (2)熟悉Creator PXA270 EVM應用Interrupt機制控制UART之程式設計		
Resources	Hardware	■ MicroTime Creator-PXA270 EVM & Power Supply ■ RS232 Null Modem Cable ■ RJ45 Cable	
	Software	■ Cygwin與ARM toolchains for PXA270 ■ Host連線程式：超級終端機 或 putty ■ TFTP Server	
Timing	(A) Introduction – 20 minutes (B) Demonstration – 15 minutes (C) Hands-on Practicing – 30 minutes (D) Exercises- 45 minutes		
Duty T. A.			

嵌入式軟體設計與實習

❖ Creator/PreSOC-PXA270 Lab-U04

- Introduction to this Lab Practicing Unit
- Demonstration
- Hands-on Practicing
- Exercises
- References and Further Reading

嵌入式軟體設計與實習

❖ Creator/PreSOC-PXA270 Lab-U04

- ✓ Introduction to this Lab Practicing Unit
- Demonstration
- Hands-on Practicing
- Exercises
- References and Further Reading

Introduction

❖ 資料傳輸類型

- ➡ 以使用的時脈分為同步與非同步傳輸
 - 同步傳輸 (synchronous communication)
 - 傳送與接收端使用共同的外部時脈信號
 - 非同步傳輸 (asynchronous communication)
 - 傳送與接收端個別使用自己的時脈信號
- ➡ 以資料的處理方式分為串列與並列傳輸
 - 並列傳輸 (parallel communication)
 - 一次傳輸多個位元，將一筆資料的多個位元一次傳送
 - 例：PC的印表機埠、Data Bus
 - 串列傳輸 (serial communication)
 - 一次傳輸一個位元，將一筆資料的多個位元依序傳送
 - 例：PC的COM port、USB

Introduction

❖ 通訊傳輸用詞

- ➡ 鮑率 (Baud Rate)
 - 資料通訊之單位，表每秒傳輸的symbol數量
 - 若1個bit為1個symbol，鮑率等同位元率(一秒傳輸位元數)
 - PXA270鮑率等同於位元率
 - PXA270 U-boot程式預設鮑率為9600 Baud (bps)

❖ UART

- ➡ Universal Asynchronous Receiver/Transmitter
- ➡ 為將序列傳輸與並列傳輸所用資料格式作傳輸轉換的電腦硬體
- ➡ 常用通訊標準介面有 RS-232、RS-422等

Introduction

❖ UART傳輸資料格式

Start	Data 0	Data 1	Data N	Parity	Stop
-------	--------	--------	-------	--------	--------	------

- ➡ 開始位元 (Start Bit)
 - 通知一次封包傳輸的開始，為邏輯0
- ➡ 資料位元 (Data Bit)
 - 傳輸時實際資料所在的區塊
 - 資料位元個數可為5至8個，通常為7或8個
 - 後面可選擇是否附上 Parity bit以作為簡單的錯誤檢查
- ➡ 結束位元 (Stop Bit)
 - 通知一次封包傳輸完成，為邏輯1
 - 個數可有1或2個，越多則不同時脈的同步化越具彈性

Introduction

❖ PXA270 UART控制器

- ➡ 擁有可程式化的鮑率產生器
- ➡ UART與CPU的互動方式
 - Polling
 - Interrupt
- ➡ 可產生多組中斷事件觸發
- ➡ 配備收發各可容納64 字元之FIFO緩衝記憶體
- ➡ FIFO緩衝記憶體存取方式
 - 軟體 (polling、interrupt)
 - DMA (Direct Memory Access)
- ➡ 相容於 16550A / 16750工業標準

Introduction

❖ PXA270 UART控制器

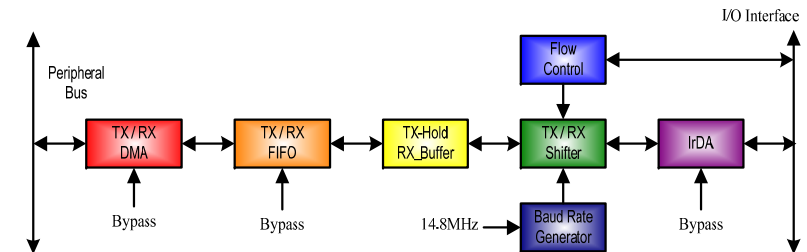
➡ 三種UART控制器

- FFUART (Full Function UART, 全功能UART)
 - 支援數據機控制能力
 - 最大傳輸速率為 230.4 Kbps (bits/s)
- BTUART (BlueTooth UART, 藍芽UART)
 - 高速UART, 可與藍牙模組連接
 - 最大傳輸速率達 921.6 Kbps (bits/s)
- STUART (Standard UART, 標準UART)
 - 不支援數據機控制能力
 - 最大傳輸速率為 230.4 Kbps (bits/s)

Introduction

❖ PXA270 UART控制器

➡ UART控制器運作方塊圖



Introduction

❖ PXA270 UART控制器

➡ PXA270 UART FIFO緩衝記憶體

- 輸入FIFO共64x11位元
- 輸出FIFO共64x8位元
- UART可選擇是否啟動FIFO
- 當啟動FIFO後, 接收到的資料會先放至輸入FIFO內, 欲傳送的資料則會先放至輸出FIFO中

Introduction

❖ PXA270 UART控制器

➡ PXA270 UART控制暫存器

■ 控制暫存器記憶體映對表

- FFUART Base位址為 0x40100000

UART Register Addresses (Base + offset)	DLAB Bit Value	Register Accessed
Base	0	Receive Buffer (read-only)
Base	0	Transmit Buffer (write-only)
Base + 0x04	0	Interrupt Enable (read/write)
Base + 0x08	X	Interrupt Identification (read-only)
Base + 0x08	X	FIFO Control (write-only)
Base + 0x0C	X	Line Control (read/write)
Base + 0x10	X	Modem Control (read/write)
Base + 0x14	X	Line Status (read-only)
Base + 0x18	X	Modem Status (read-only)
Base + 0x1C	X	Scratch Pad (read/write)
Base + 0x20	X	Infrared Selection (read/write)
Base	1	Divisor Latch Low (read/write)
Base + 0x04	1	Divisor Latch High (read/write)

Introduction

❖ PXA270 UART控制器

➡ PXA270 UART控制暫存器

- Receive Buffer Register (RBR)
 - 存放UART接收之資料
 - 啟動FIFO時，RBR會被寫入FIFO的頭一筆資料
- Transmit Holding Register (THR)
 - 存放UART準備傳輸之資料
 - 啟動FIFO時，寫入THR的資料會放入FIFO的最末端
- UART鮑率 Divisor Latch Register (DL)
 - 作為UART鮑率產生器的除頻輸入，含DLH與DLL二Register
 - Divisor Latch Register – High (DLH)：代表除頻輸入高8位元
 - Divisor Latch Register – Low (DLL)：代表除頻輸入低8位元

Introduction

❖ PXA270 UART控制器

➡ PXA270 UART控制暫存器

- Line Status Register (LSR)
 - 顯示UART狀態
 - bit0 (DR)為 Data Ready，代表是否收到資料並存於RBR
 - 資料從RBR被讀取後，DR會恢復為 0

31-8	7	6	5	4	3	2	1	0
Reserved	FIFOE	TEMT	TDRQ	BI	FE	PE	OE	DR

Introduction

❖ PXA270 UART控制器

➡ PXA270 UART控制暫存器

- Line Control Register (LCR)
 - 作資料傳輸設定
 - DLAB為 1可存取DLL與DLH，0則可存取RBR與THR
 - WLS設定傳輸資料的長度，3為 8-bit資料

31-8	7	6	5	4	3	2	1	0
Reserved	DLAB	SB	STKYP	EPS	PEN	STB	WLS	

Introduction

❖ PXA270 UART控制器

➡ PXA270 UART控制暫存器

- Interrupt Identification Register (IIR)
 - 顯示UART中斷的類型
 - nIP為0代表有interrupt等待處理，1代表沒有
 - FIFOES為3代表有啟動FIFO，0代表未啟動FIFO
 - IID為1代表資料傳送中斷，2代表資料接收中斷

31-8	7	6	5	4	3	2	1	0
Reserved	FIFOES		EOC	ABL	TOD	IID		nIP

Introduction

❖ PXA270 UART控制器

➡ PXA270 UART控制暫存器

■ Interrupt Enable Register (IER)

- 控制UART interrupt的啟動設定
- RAVIE決定資料接收中斷能否被觸發
- TIE決定資料傳送中斷能否被觸發，在THR被寫入或IIR被讀取後會被清為0
- RTOIE致能 FIFO receive time-out interrupt
- UUE為1代表UART可使用，0則為不可使用

31-8	7	6	5	4	3	2	1	0
Reserved	DMAE	UUE	NRZE	RTOIE	MIE	RLSE	TIE	RAVIE

Introduction

❖ PXA270 UART控制器

➡ PXA270 UART控制暫存器

■ FIFO Control Register (FCR)

- 設定FIFO緩衝區暫存器之相關設定
- TRFIFOE為1時，啟動FIFO（進入FIFO mode）
- ITL設定觸發 UART Interrupt與DMA Request的 threshold

31-8	7	6	5	4	3	2	1	0
Reserved	ITL	BUS	TRAIL	TTL	RESETTF	RESETRF	TRFIFOE	

Introduction

❖ PXA270 UART鮑率設定

➡ 鮑率產生器

- 使用14.857MHz的時脈源
- 以DLH與DLL產生鮑率值

➡ 自動鮑率偵測

➡ 手動調整鮑率

- 調整 DLH與 DLL暫存器
- 使用計算法計算鮑率
 - 計算公式 $14.7456M / (16 * (DLH @ DLL))$ (單位bps)
 - 實際鮑率 $14.857MHz / (16 * (DLH @ DLL))$ (單位bps)
 - 適用於 28.8Kbps以上的鮑率。低於28.8Kbps的鮑率若實際產生鮑率與目標鮑率相差1%以上，傳輸會產生問題

Introduction

❖ PXA270 UART鮑率設定

➡ 手動調整鮑率

- 使用查表法決定鮑率

Required Frequency(MHz)	Required Baud Rate	DLH@DLL	Actual Baud Rate	Actual Frequency(MHz)
14.7456	300	3072	302	14.857
14.7456	1200	768	1209	14.857
14.7456	2400	384	2418	14.857
14.7456	4800	192	4836	14.857
14.7456	9600	96	9675	14.857
14.7456	19200	48	19345	14.857
14.7456	38400	24	38690	14.857
14.7456	57600	16	58035	14.857
14.7456	115200	8	116070	14.857

Introduction

❖ PXA270 中斷控制器

➡ 採用2階層中斷架構

■ Primary

- 以 Processor Device 為來源所發出的中斷

■ Secondary

- 以周邊設備的內部事件為來源所發出的中斷
- 多個 secondary source 會應對至同一個 primary source
- UART 控制器為一 primary source，其中有接收、傳輸等 secondary source

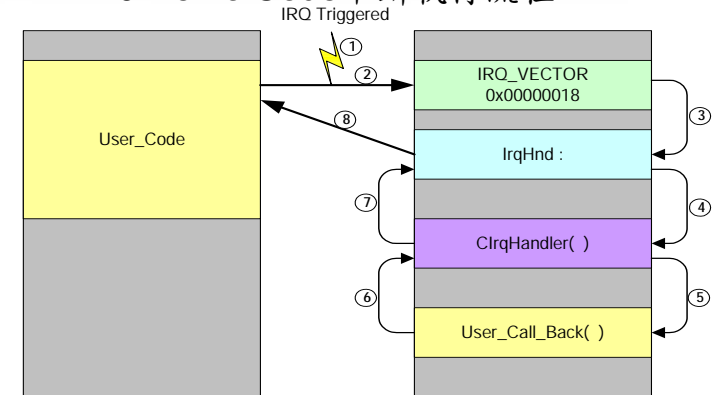
➡ 中斷 Level 分為 IRQ 與 FIQ 兩種

➡ 中斷 mask 決定中斷之觸發與否

Introduction

❖ PXA270 中斷控制器

➡ PXA270 Demo Code 中斷執行流程



Introduction

❖ PXA270 UART 中斷

➡ UART 中斷分為五種，觸發時以 IIR 判斷之

■ Receive Line Status Error Interrupt

- 最高優先權 (優先權 0)
- 當 LSR 有任何 Error bit 被設為 1 時觸發

■ Receive Data Interrupt

- 次高優先權 (優先權 1)
- 在未啟動 FIFO 時，RBR 被寫入時觸發
- 在啟動 FIFO 時，輸入 FIFO 內資料量到達 threshold 觸發

■ Receive Time-out Interrupt

- 優先權 2
- 僅在啟動 FIFO 可觸發
- 當有資料在輸入 FIFO 內，且最近一次接收與最久之前的讀取發生超過一定時間 (4 個資料接收的時間)

Introduction

❖ PXA270 UART 中斷

➡ UART 中斷分為五種，觸發時以 IIR 判斷之

■ Transmit Data Interrupt

- 優先權 3
- 啟動 FIFO 時，輸出 FIFO 至少半空才可觸發
- 未啟動 FIFO 時，THR 已將前一筆資料送出才可觸發

■ Modem Interrupt

- 最低優先權 (優先權 4)
- 不討論

嵌入式軟體設計與實習

- ❖ Creator/PreSOC-PXA270 Lab-U04
 - ➡ Introduction to this Lab Practicing Unit
 - ✓ **Demonstration**
 - ➡ Hands-on Practicing
 - ➡ Exercises
 - ➡ References and Further Reading

Demonstration

- ❖ The above operation procedures will now be demonstrated:
 - ➡ The operations and outcome messages at host computer will be broadcast to your computer.
 - ➡ The outcome on EVM will be shown at demo desk.
- ❖ Part A - 程式說明
- ❖ Part B - 下載程式碼與編譯
- ❖ Part C - 超級終端機操作與執行程式

Demonstration

- ❖ Part A - 程式說明
 - ➡ 程式使用方式
 - 本程式為將使用者所從鍵盤輸入或複製貼上至實驗板的內容呈現在終端機上
 - 程式分為 polling與 interrupt兩種模式，使用 ctrl + A鍵在兩模式之間做交換，切換之後螢幕會在新的一行顯示代表 Polling的 'P'或代表 Interrupt的 'I'
 - Interrupt模式中，每輸入一個文字，LCD上會顯示觸發中斷的訊息

Demonstration

- ❖ Part A - 程式說明
 - ➡ 變數
 - FFLCR
 - 映對至FFUART的LCR
 - 資料型態：unsigned long (U32)、Macro
 - #define FFLCR (*(volatile unsigned long*)0x4010000C)
 - FFLSR
 - 映對至FFUART的LSR
 - 資料型態：unsigned long、Macro
 - #define FFLSR (*(volatile unsigned long*)0x40100014)

Demonstration

❖ Part A - 程式說明

➡ 變數

■ FFRBR

- 在LCR的DLAB為0且被讀取時映對至FFUART的RBR
- 資料型態：unsigned long、Macro
- #define FFRBR (*(volatile unsigned long*)0x40100000)

■ FFTHR

- 在LCR的DLAB為0且被寫入時映對至FFUART的THR
- 資料型態：unsigned long、Macro
- #define FFTHR (*(volatile unsigned long*)0x40100000)
- 將資料存入後UART即會自動傳送

Demonstration

❖ Part A - 程式說明

➡ 變數

■ FFDLL

- 在LCR的DLAB為1時映對至FFUART的DLL
- 資料型態：unsigned long、Macro
- #define FFDLL (*(volatile unsigned long*)0x40100000)

■ FFDLH

- 在LCR的DLAB為1時映對至FFUART的DLH
- 資料型態：unsigned long、Macro
- #define FFDLH (*(volatile unsigned long*)0x40100004)

Demonstration

❖ Part A - 程式說明

➡ 函數

■ int uart_init()

- 初始化UART軟體模組，使用其他UART函數前須呼叫此函數
- 參數：無
- 回傳：0為初始化失敗，1為初始化成功

■ int uart_on(int channel, termios_t *termios)

- 設定channel代表的UART控制器環境與註冊ISR
- 參數：channel，UART控制器編號，UART0為FFUART
UART1為BTUART
UART2為STUART
termios，UART傳輸環境參數 (baud rate, etc.)
- 回傳：1代表設置成功，0為參數錯誤

Demonstration

❖ Part A - 程式說明

➡ 函數

■ int uart_put_char(int channel, char ch)

- 將單筆8-bit資料放至程式的傳送buffer內並觸發interrupt傳送
- 參數：channel，UART控制器編號
ch，欲傳送的資料
- 回傳：1代表傳送成功，0代表buffer溢位或參數錯誤

■ int uart_put_string(int channel, char* pstring)

- 將多筆8-bit資料放至程式的傳送buffer並觸發interrupt傳送
- 參數：channel，UART控制器編號
pstring，欲傳送的資料起始位址
- 回傳：1代表傳送成功，0代表buffer溢位或參數錯誤

Demonstration

❖ Part A - 程式說明

➡ 函數

- int uart_get_char(int channel, char* ch)
 - 讀取程式的接收buffer，並將讀取值存入ch中，接收buffer中的資料為透過interrupt傳入
 - 參數：channel，UART控制器編號
ch，欲將讀取資料放置之處的記憶體位址
 - 回傳：1為讀取成功，0代表buffer是空的或參數錯誤
- int uart_off(int channel)
 - 關閉channel所代表的UART控制器
 - 參數：channel，UART控制器編號
 - 回傳：1代表關閉成功，0為關閉失敗

Demonstration

❖ Part A - 程式說明

➡ 函數

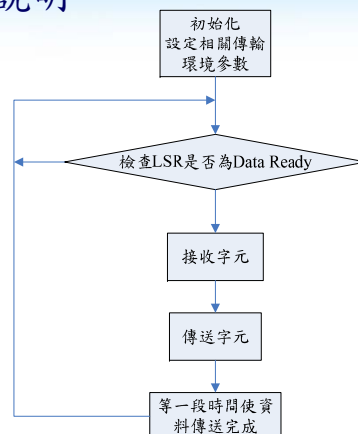
- void serial_isr(U16 irq, struct irq_desc, void *pVoid)
 - UART Interrupt的ISR，ISR參數與回傳資料型態為固定格式
 - 參數：irq，ISR所註冊的interrupt 編號
desc，註冊isr之相關 structure
pVoid，傳予此ISR所使用的參數
 - 回傳：無

Demonstration

❖ Part A - 程式說明

➡ 程式流程

■ Polling

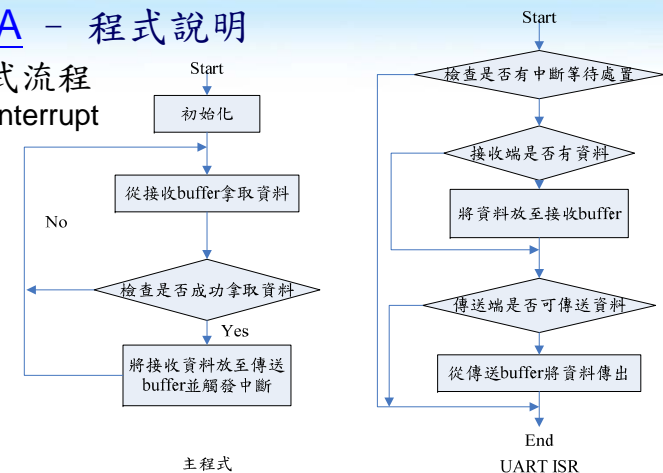


Demonstration

❖ Part A - 程式說明

➡ 程式流程

■ Interrupt



Demonstration

❖ Part B – 下載程式碼與編譯

- ➡ 執行 cygwin，做以下步驟
- ➡ 1. 於 cygwin 下進入本次實習的資料夾下
 - 鍵入「cd /usr/var/pxa270_lab_material/Lab04」
 - 若以上目錄不存在，則將 pxa270_lab_material 資料夾複製到「C:\cygwin\usr\var」目錄下再重複以上步驟
 - 註：在 cygwin 下鍵入第一個字母再按「tab 鍵」會自動出現符合的目錄或檔案名稱，鍵入 ls 則顯示當下目錄的所有資料

Demonstration

❖ Part B – 下載程式碼與編譯

- ➡ 2. 建立本次實習程式碼
 - 鍵入「./set_env.sh」
- ➡ 3. 檢查本次實習程式碼的檔案是否完整
 - 鍵入「./check_file.sh」
- ➡ 4. 進入資料夾 Code，此為本次實習程式碼所在之目錄
 - 鍵入「cd Code」
- ➡ 5. 進行編譯
 - 鍵入「make clean;make」
 - 若程式沒有錯誤則會產生檔案「lab.bin」，若程式有錯誤則會有錯誤訊息顯示，修改後再進行步驟 5

Demonstration

❖ Part C – 電腦端操作以執行程式

- ➡ 1. 進入 Windows 下的資料夾
C:\cygwin\usr\var\pxa270_lab_material\Lab04\Code
- ➡ 2. 設置好 PXA270 實驗板，先不要開啟電源，開啟「C:\TFTP」目錄下的「TFTPD.exe」
- ➡ 3-A. 開啟與設定超級終端機
 - 開始→所有程式→附屬應用程式→通訊→超級終端機
 - Win7 使用者參照 3-B putty 的設定操作

Demonstration

❖ Part C – 電腦端操作以執行程式

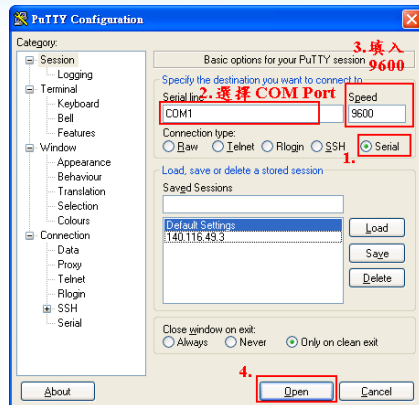
➡ 3-A. 開啟與設定超級終端機



Demonstration

❖ Part C –電腦端操作以執行程式

➡ 3-B. putty設定操作



Demonstration

❖ Part C –電腦端操作以執行程式

➡ 4. 下載編譯好的程式至實驗板上執行

- (1) 開啟Creator PXA270電源
- (2) 在超級終端機畫面按任意鍵停止倒數
- (3) 在超級終端機畫面鍵入 printenv檢查環境變數
serverip=AAA.BBB.CCC.XXX
ipaddr=AAA.BBB.CCC.YYY
兩個變數IP位址前三段須相同，最後一段不可相同
Ex : serverip=140.116.49.167
ipaddr=140.116.49.170

若原設定確認無誤，跳至步驟(7)

Demonstration

❖ Part C –電腦端操作以執行程式

➡ 4. 下載編譯好的程式至實驗板上執行

- (4) 設定 serverip，設定的位址須與連接的電腦相同
鍵入「setenv serverip 電腦IP位址」
- (5) 設定 ipaddr
鍵入「setenv ipaddr IP位址」
- (6) 儲存設定的變數
鍵入「saveenv」
- (7) 用 tftp將 lab.bin傳到 PXA270實驗板上
鍵入「tftp a0000000 lab.bin」
- (8) 執行程式
鍵入「go a0000000」

Demonstration

❖ Part C –電腦端操作以執行程式

➡ 4. 下載編譯好的程式至實驗板上執行

- 5. 執行結果
 - 鮑率設定為9600，複製26個字母
“abcdefghijklmnopqrstuvwxyz”至終端機，在 Polling與
Interrupt模式下貼上後所產生的結果

```
Load address: 0xa0000000
Loading: #####
done
Bytes transferred = 48804 (bea4 hex)
## Starting application at 0xa0000000 ...
```

```
Pacegikmopr tvxz      Polling
labcdefghijklmnopqrstuvwxyz Interrupt
```


嵌入式軟體設計與實習

❖ Creator/PreSOC-PXA270 實驗模組-U04

Q&A



嵌入式軟體設計與實習

❖ Creator/PreSOC-PXA270 Lab-U04

- Introduction to this Lab Practicing Unit
- Demonstration
- ✓ **Hands-on Practicing**
- Exercises
- References and Further Reading

Hands-on Practicing

❖ Based on the introduction and demo, please do the following practicing:

- Part A – 擷取程式碼與編譯
- Part B – 下載與執行程式，並複製一段文字串在 Polling 與 Interrupt 兩個模式分別貼上，觀察產生的現象，試解釋其原因。

嵌入式軟體設計與實習

❖ Creator/PreSOC-PXA270 Lab-U04

- Introduction to this Lab Practicing Unit
- Demonstration
- Hands-on Practicing
- ✓ **Exercises**
- References and Further Reading

Exercises

❖ Creator/PreSOC-PXA270 Lab-U04

- Introduction to this Lab Practicing Unit
- Demonstration
- Hands-on Practicing
- Exercises
- ✓ **References and Further Reading**

References

❖ 嵌入式微處理器系統 (以Creator-PXA270為例)

- 作者：新華電腦股份有限公司
- 出版：新華電腦股份有限公司



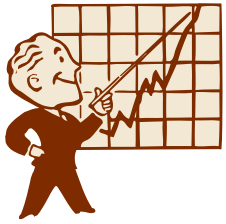
References

❖ PXA27x Processor Family Developer's Manual

- 出版：Intel
- 參閱章節：第10章UART

嵌入式軟體設計與實習

❖ Creator/PreSOC-PXA270 實驗模組-U04



The End