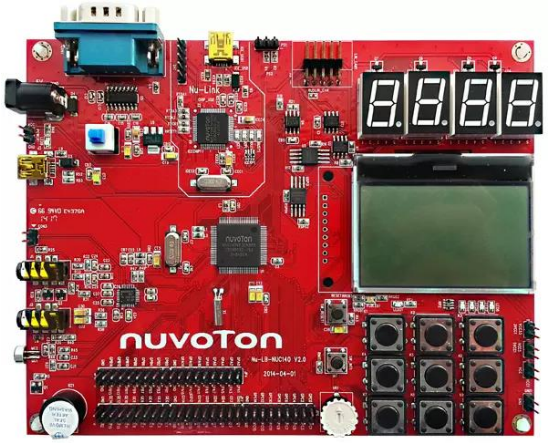
微處理機系統與介面技術 LAB2

系所: 通訊三　　學號:409430043　　姓名: 高司玹

<實驗器材>

Nu-LB-NUC 140 V2.0 開發板　　PL2303TA



<實驗過程與方法>

UART 的工作就是從 CPU 一次接收 8 bits 的資料(parallel)，然後將這些資料 1 次 1bit 的送往周邊設備(serially)。同時，UART 還可以接收周邊設備傳送來的資料，當組成 8 bits 時，再將資料送往 CPU。

在 PC 端，通常可以透過終端管理員或 putty 等軟體透過 UART 與設備溝通。

PC 端要控制 UART，只需要開啟對應的 com port，接著直接呼叫 read()，write()程式即可。而嵌入式系統的 UART 控制，一般會有兩種方式。

1. Interrupt Driven： 使用中斷向量表中的 UART interrupt 實作 UART 功能。
2. Polled I/O：使用 polling 的方式實作 UART 功能。通常是啟動 timer，當 timer interrupt 發生時查詢 UART 對應的 register。

而我們這次的 sample code 就是使用 interrupt driven 的方式，透過觸發中斷(RDA 中斷)並執行中斷後的 call back function 將字印到 terminal 上。

```
200     UART_EnableInt(UART0, (UART_IER_RDA_IEN_Msk | UART_IER_THRE_IEN_Msk | UART_IER_TOUT_IEN_Msk));
201     while(g_bWait);
202
203     /* Disable Interrupt */
204     UART_DisableInt(UART0, (UART_IER_RDA_IEN_Msk | UART_IER_THRE_IEN_Msk | UART_IER_TOUT_IEN_Msk));
205     g_bWait = TRUE;
```

第 200 行就是將 RDA、THRE、TIME_OUT 中斷打開以準備接收輸入的字並觸發中斷

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
DMA_RX_EN	DMA_TX_EN	AUTO_CTS_EN	AUTO_RTS_EN	TIME_OUT_EN	Reserved		LIN_RX_BRK_IEN
7	6	5	4	3	2	1	0
Reserved	WAKE_EN	BUF_ERR_IEN	RTO_IEN	MODEM_IEN	RLS_IEN	THRE_IEN	RDA_IEN

201 行這個變數 g_bWait 初始值為 TRUE，是為了讓上述的中斷功能持續打開。

```
volatile int32_t g_bWait           = TRUE;
```

直到當我們輸入 enter 之後會將其改成 false (call back function 內)，此時就會跑到 204 行將上述中斷功能關閉。

<遇到的問題>

這次 lab 有幾個重點: UART_TEST_HANDLE、EnableInt、DisableInt 的函式簡介跟運算都要了解。

在讀到 main function 時我就有個問題是為何 UART02_IRQHandler 不在 main 內也能順利執行。

而後再助教的細心解說，還有老師上課的補充才知道原來是因為作業系統的原因，當 ISR 發生會中斷原本的程式碼而去執行對應的 call back function。

也遇到一個問題是 Putty 上顯示的字是亂碼，後來才知道是 baud rate 沒有設定到跟 putty 的一樣。

<心得與收穫>

這次學到很多關於中斷、URAT 這種傳輸方法的相關知識，雖然累，但是成功後真的頗有成就感。