MICROCHIP

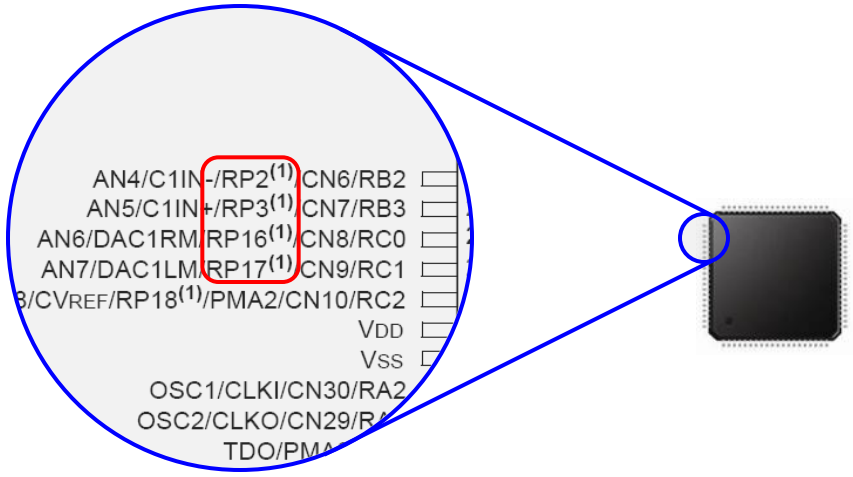
Regional Training Centers

Section 11
PPS and UART Module

MICROCHIP
Regional Training Centers

What's PPS

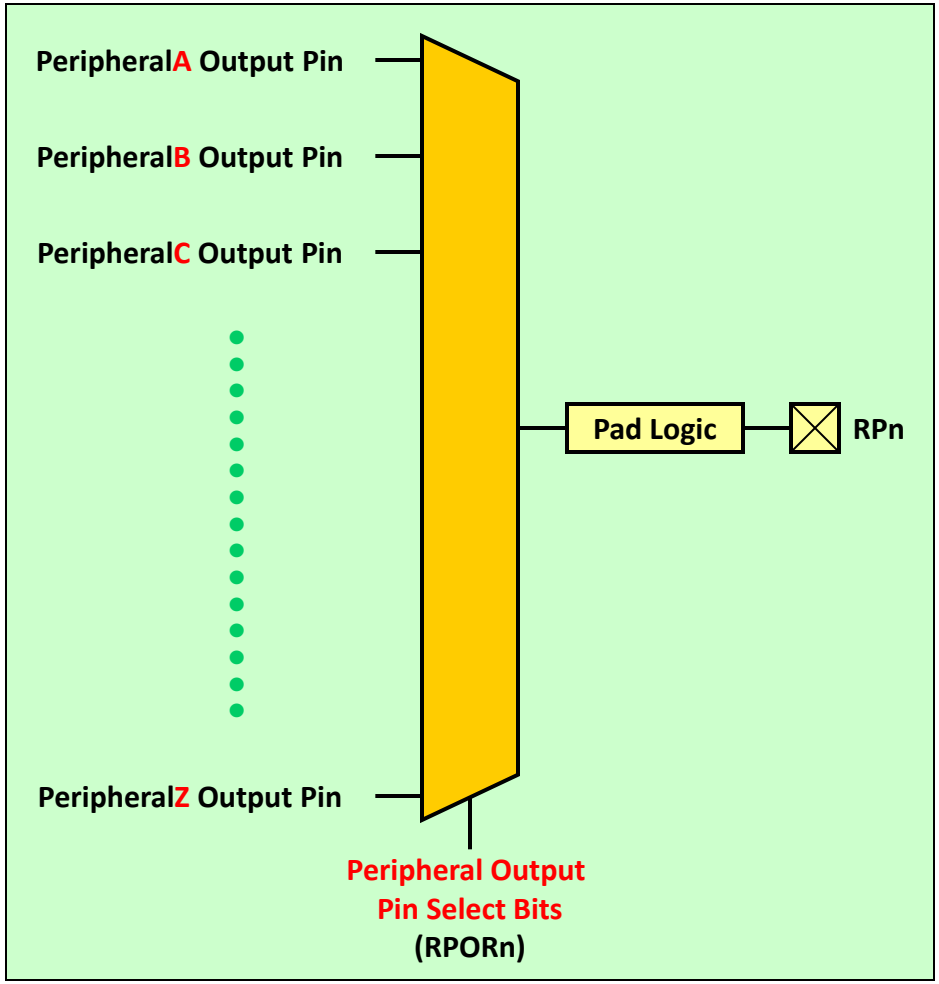
- PPS：Peripheral Pin Select,周邊接腳選擇
  一種可以重新改變周邊訊號接腳位置的功能。讓使用者可以自行安排(Remapping)周邊訊號接腳位置,達到最彈性化的接腳使用。
- 在接腳上有標示RPn/RPIn的, 都是可以Remapping的接腳。可已透過設定將UART,SPI,etc..的訊號接腳透過該Pin連接。
- PPS支援大部分的數位週邊,如:
  UART,SPI,OC,IC等。但需要較複雜狀態處理的數位週邊及類比功能則不支援,其接腳為固定不可變更的,如:
  I²C,USB,PMP及類比數輸入等。



MICROCHIP
Regional Training Centers

PPS's Output

- 輸出多工器的輸入端連接到各個周邊的輸出訊號,輸出則接到特定接腳(RPx)。
- 每個接腳(RPx)都有專屬的多工器,可單獨指定其所要接的輸出訊號。
- PPS輸出的指定原則:
  指定接腳要使用哪個周邊的輸出。

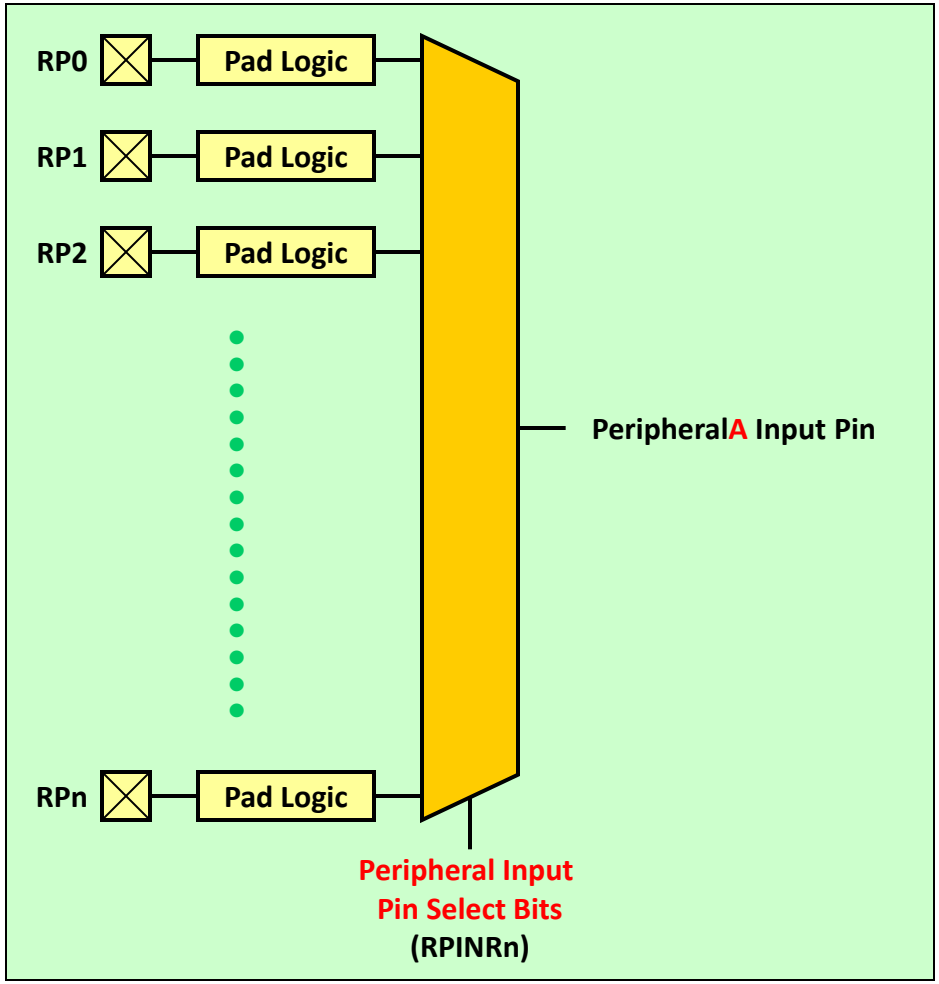


PPS基本輸出架構

MICROCHIP
Regional Training Centers

PPS's Input

- 輸入多工器的輸入端連接到各個輸入接腳(RPx, RPIx), 輸出則接到某個周邊的輸入接腳。
- 每個周邊都有專屬的多工器,可單獨指定其所要連接的接腳(RPx, RPIx) 。
- PPS輸入的指定原則:
  指定周邊的輸入要使用哪個接腳。
  RPx 可作為輸出或輸入用;
  RPIx只能作為輸入。



PPS基本輸入架構

MICROCHIP
Regional Training Centers

支援PPS功能的周邊

- External Interrupts
- Timer External Clocks
- Input Captures
- Output Compares
- PWM Fault Input pins
- SPI,UART Functions
- QEI Inputs
- Data Converter Interface
- CAN

Table 30-1: Selectable Input Sources (Maps Input to Function)

Input Name(1)	Function Name	Register	Configuration Bits
External Interrupt 1	INT1	RPINR0	INT1R<4:0>
External Interrupt 2	INT2	RPINR1	INT2R<4:0>
Timer2 External Clock	T2CK	RPINR3	T2CKR<4:0>
Timer3 External Clock	T3CK	RPINR3	T3CKR<4:0>
Input Capture 1	IC1	RPINR7	IC1R<4:0>
Input Capture 2	IC2	RPINR7	IC2R<4:0>
Input Capture 7	IC7	RPINR10	IC7R<4:0>
Input Capture 8	IC8	RPINR10	IC8R<4:0>
Output Compare Fault A	OCFA	RPINR11	OCFAR<4:0>
PWM1 Fault	$\overline{\text{FLTA1}}$	RPINR12	FLTA1R<4:0>
PWM2 Fault	$\overline{\text{FLTA2}}$	RPINR13	FLTA2R<4:0>
QEI Phase A	QEA	RPINR14	QEAR<4:0>
QEI Phase B	QEB	RPINR14	QEBR<4:0>
QEI Index	INDX	RPINR15	INDXR<4:0>
UART1 Receive	U1RX	RPINR18	U1RXR<4:0>
UART1 Clear t			
SPI1 Data Inpu			
SPI1 Clock Inp			
SPI1 Slave Sel			

Note 1: Unles

Table 30-2: Output Selection for Remappable Pin (RPn)

Function	RPnR<4:0>	Output Name
NULL	00000	RPn tied to default port pin
U1TX	00011	RPn tied to UART1 Transmit
$\overline{\text{U1RTS}}$	00100	RPn tied to UART1 Ready to Send
SDO1	00111	RPn tied to SPI1 Data Output
SCK1OUT	01000	RPn tied to SPI1 Clock Output
SS1OUT	01001	RPn tied to SPI1 Slave Select Output
OC1	10010	RPn tied to Output Compare 1
OC2	10011	RPn tied to Output Compare 2
UPDN	11010	RPn tied to QEI direction (UPDN) status

MICROCHIP
Regional Training Centers

PPS Lock, Unlock

- PPS 設定前必須先進行解鎖,設定完後再上鎖。以避免非預期的動作更動PPS的設定。任何位元的非預期改變都會造成 MCU Reset。
- 解鎖動作必須將OSCCON的IOLOCK清除為0;

  OSCCONbits.IOLOCK = 0;

  鎖定動作必須必須將OSCCON的IOLOCK設為1。

  OSCCONbits.IOLOCK = 1;
- PPS 的鎖定模式,在Configuration Bit中設定

  IOL1WAY:IOLOCK One-Way Set Enable bit

  IOL1WAY=1,PPS的Mapping只能被設定一次。

  IOL1WAY=0,PPS的Mapping可以重複的設定。

MICROCHIP
Regional Training Centers

MPLAB C30對PPS的支援

- Output Mapping

  iPPSOutput(*pin* , *fn*);

  pin:PPS的接腳編號

  OUT_PIN_PPS_RPx

  fn:周邊模組輸出訊號

  OUT_FN_PPS_NULL , OUT_FN_PPS_CxOUT ,
  OUT_FN_PPS_UxTX , OUT_FN_PPS_SDOx ,
  OUT_FN_PPS_OCx ,

- Input Mapping

  iPPSInput(*fn* , *pin*);

  fn:周邊模組輸入訊號

  IN_FN_PPS_INTx , IN_FN_PPS_ICx , IN_FN_PPS_OCFx ,
  IN_FN_PPS_UxRX , IN_FN_PPS_SDIx ,

  pin:PPS的接腳編號

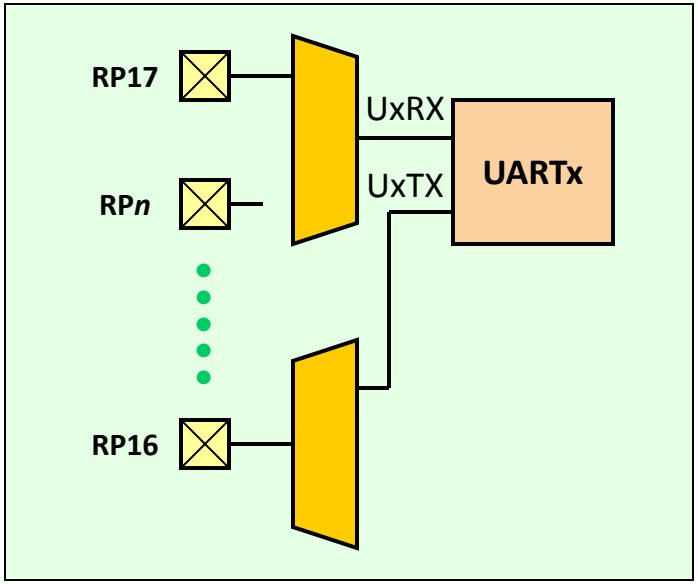
  IN_PIN_PPS_RPx

PPS UART Mapping Example

- PPS設定UART2, Tx, Rx的Mapping:

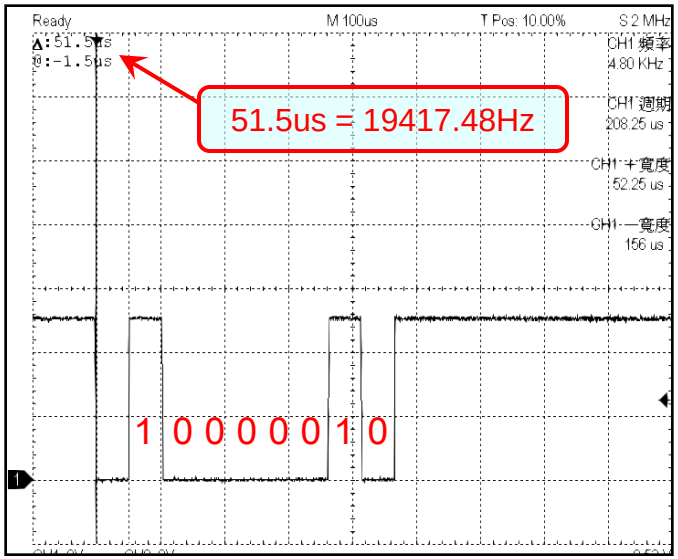
  Ex:

```
#include <PPS.h>
int main( void )
{
    ...
    ...
    ...
    ...
    PPSUnLock;
    iPPSOutput( OUT_PIN_PPS_RP16 , OUT_FN_PPS_U2TX );
    iPPSInput( IN_FN_PPS_U2RX , IN_PIN_PPS_RP17 );
    PPSLock;
    ...
}
```

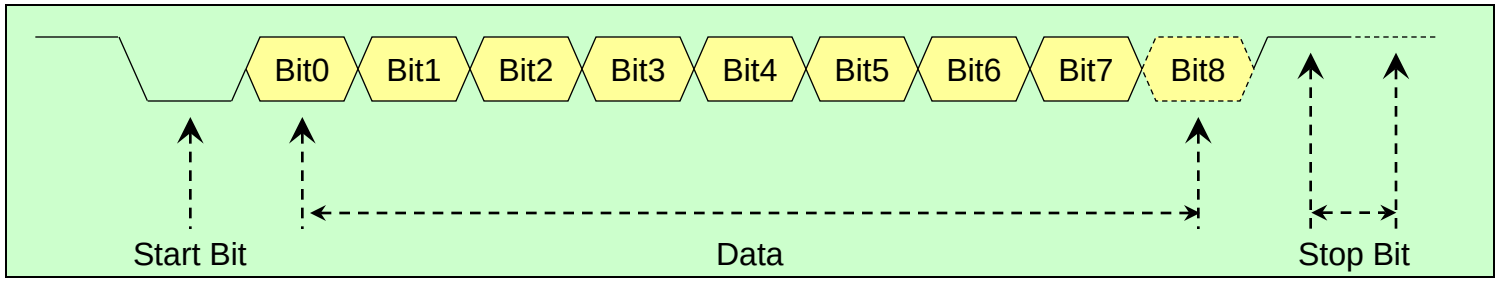


What's UART

- UART:Universal Asynchronous Receiver Transmitter,泛用非同步接收傳送模組。
  一個以串列方式進行資料交換與溝通的通訊模組。
- UART 傳輸的資料由LSB先傳。格式如下,包含一個起始位元,八或九位元的資料,一至二個停止位元。

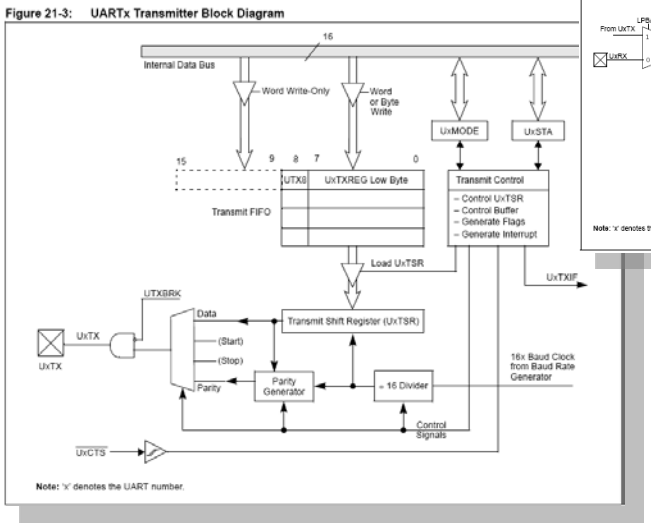


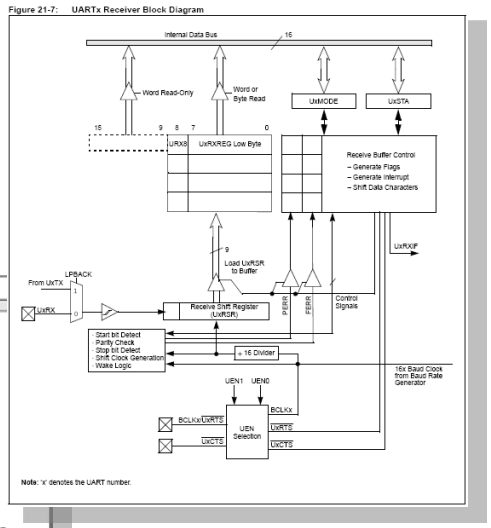
UART傳送範例'A'(0x41),19200 bps



PIC24 UART Tran./Rece. Block

- PIC24的UART Module支援,8 or 9-Bits Data, 同位檢查及硬體流量控制(DTS,RTS)。
- 採全雙工, 非同步通訊。支援RS-232, RS-485, IrDA及LIN Bus。
- 四層深度的輸出入緩衝器(Buffer) 。
- 具有錯誤檢查機制
  (Parity Error, Frame Error, Overflow, etc..),
  Loopback及
  Address Detect
  (RS-485), Auto Baud
  Detect(LIN bus)功能。

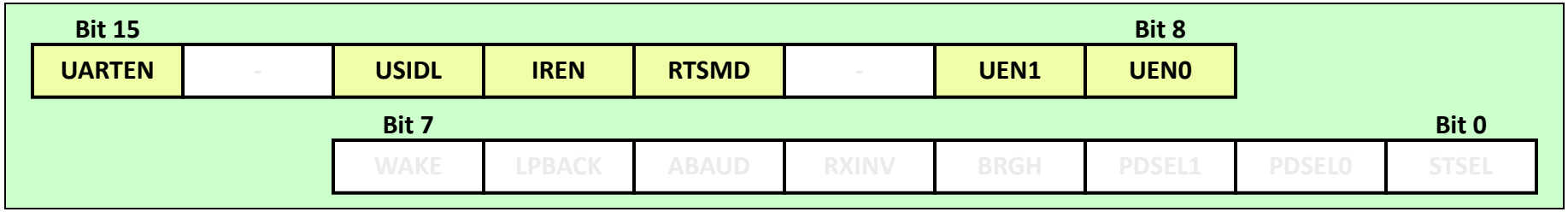




MICROCHIP
Regional Training Centers

控制暫存器 UxMODE

UxMODE



- UARTEN:開關。
- USIDL:IDLE模式是否關閉。
- IREN:IrDA Support。
- RTSMD:UxRTS是否為流量控制功能。
- UEN<1:0>:UART模式選擇

  00:UxTx , UxRx , 01:UxTx , UxRx , UxRTS

  10:UxTx , UxRx , UxCTS , UxRTS

  11:UxTx , UxRx , BCLKx

MICROCHIP
Regional Training Centers

控制暫存器 UxMODE

UxMODE

Bit 15							Bit 8
UARTEN	-	USIDL	IREN	RTSMD	-	UEN1	UEN0

Bit 7							Bit 0
WAKE	LPBACK	ABAUD	RXINV	BRGH	PDSEL1	PDSEL0	STSEL

- WAKE:啟用喚醒功能。
- LPBACK:啟用迴路測試模式。
- ABAUD:啟用自動鮑率偵測。
  BRGH:High Baud Secect。
- PDSEL<1:0>:同位檢查與資料格式設定。
  00 = 8Bit,No Parity , 01 = 8Bit Even ,
  10 = 8Bit Odd , 11 = 9Bit No Parity
- STSEL:停止位元數量選擇。
  0 = 1Bit , 1 = 2Bits

MICROCHIP
Regional Training Centers

控制及狀態暫存器 UxSTA

UxSTA

Bit 15							Bit 8
UTXISEL1	UTXINV	UTXISEL0	-	UTXBRK	UTXEN	UTXBF	TRMT

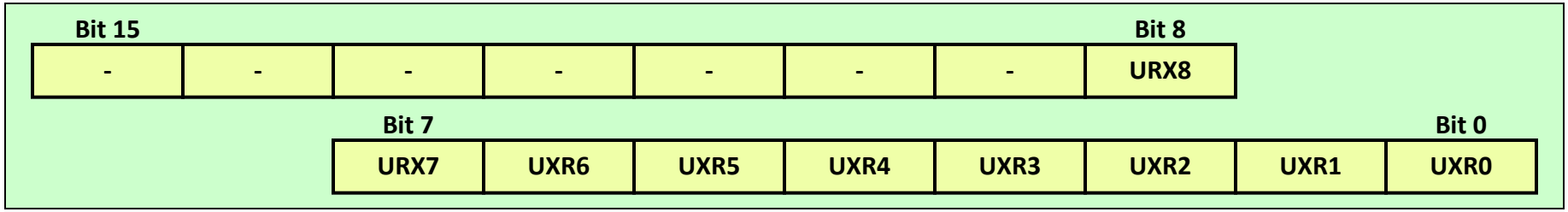
Bit 7							Bit 0
URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA

- UTXISEL<1:0>:傳送中斷模式設定(預設0x00)
- 00 = Character -> Transmit Shift Register
- UTXEN:啟動傳送功能。
- UTXBF:傳送暫存器是否已滿？
- URXISEL<1:0>:接收中斷模式設定(預設0x00)
- PERR:同位檢查錯誤?
- FERR:框架錯誤?
- OERR:接收暫存器是否溢位？
- URXDA:是否接收到資料?

MICROCHIP
Regional Training Centers

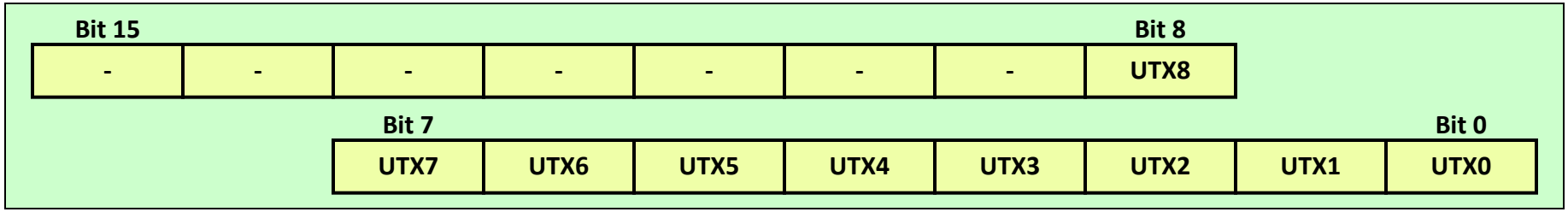
接收與傳送暫存器

UxRXREG



- URX<8:0>:接收暫存器

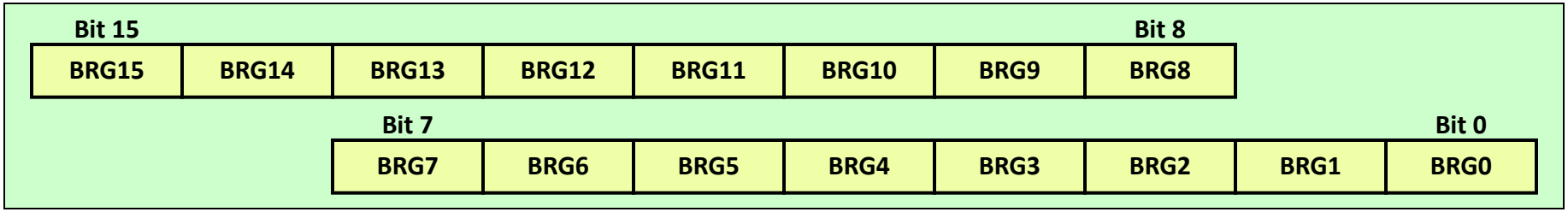
UxTXREG



- UTX<8:0>:傳送暫存器

鮑率產生暫存器UxBRG

UxBRG



- BRG<15:0>:鮑率產生暫存器

MICROCHIP
Regional Training Centers

UART鮑率計算

- PIC24的鮑率(Baud Rate)的計算分為高低兩種速率,透過UxMODE中BRGH來設定。BRGH=0為低速;反之則為高速。
- 低速率時(BRGH=0),計算方式如下:

$$UxBRG = \frac{SystemClock}{16 \times IdeaBaulRate} - 1 \qquad ActualBaudRate = \frac{SystemClock}{16 \times (UxBRG + 1)}$$

- 高速率時(BRGH=1),計算方式如下:

$$UxBRG = \frac{SystemClock}{4 \times IdeaBaulRate} - 1 \qquad ActualBaudRate = \frac{SystemClock}{4 \times (UxBRG + 1)}$$

MICROCHIP
Regional Training Centers

MPLAB C30對UART的支援

- MPLAB C30中可用的UART Function:
  OpenUARTx( ); // 設定工作模式,開啟UART。
  BusyUARTx( ); // 測試UART是否傳送資料中?
  WriteUARTx( ); // 寫入資料,準備傳送。
  DataRdyUARTx( ); // 測試UART是否已經接收到資料?
  ReadUARTx( ); // 讀取接收到的資料 。
  putsUARTx( ); // 循序寫入資料,準備傳送。
  getsUARTx( ); // 循序讀取接收到的資料。
  ConfigIntUARTx( ); // 設定UART的中斷優先權,中斷啟用。
  ……

MICROCHIP
Regional Training Centers

UART Example

- 簡單的UART初始化範例

```
PPSUnLock;
PPSInput( PPS_UxRX , PPS_RP11 );
PPSOutput( PPS_RP12 ,PPS_UxTX );
PPSLock;
OpenUARTx( UART_EN & … );
```

- 簡單的接收範例

```
while( !DataRdyUARTx( ) );
RxData = ReadUARTx( );
```

- 簡單的傳送範例

```
while(BusyUARTx( ) );
WriteUARTx( TxData );
```

- 簡單的UART中斷開啟範例

```
ConfigIntUART2( UART_RX_INT_EN & ... );
```

MICROCHIP
Regional Training Centers

Lab10 UART

- 利用Lab7的程式,將UART的功能加入。初始化UART2,設定為19200,N,8,1。
- 建立UART 接收的中斷服務常式,並開啟UART的接收中斷,設定優先權。優先權設定為預設值”4” 。
- 設定PPS,指定UART2的傳送接腳(Tx)為RP10(Pin31),接收(Rx)為RP17(Pin32)。
- 閱讀UART Function的說明文件,了解UART Function的始用方法。了解如何使用OpenUARTx( ), ConfigIntUARTx( ), ReadUARTx( ), WriteUARTx( ), etc.. 。
- 閱讀PPS Function的說明文件,了解PPS Function的始用方法。 了解如何使用iPPSInput( ), iPPSOutput( ), PPSLock, PPSUnlock。
- 使用PIC18F14K50作USB to UART的Bridge, 設定PC上的超級終端機, 透過PC觀察輸出結果。

Lab10 UART Step1

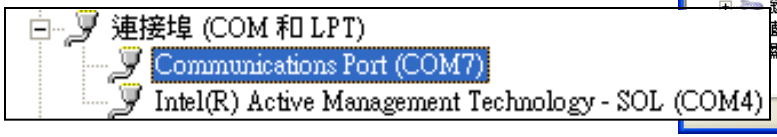
- 回憶下Lab6中燒錄Secondary MCU(PIC18F14K50)的步驟, 用同樣步驟, 把USB to UART的功能, 燒錄進去。這支程式, 在燒錄成功後, 會看到D1, D2兩顆LED一起閃爍。

  (Path: 2013 Winter Elite\PIC18F14K50 Hex Files\ Device - CDC - Serial Emulator.Hex)

- 使用Mini USB Cable連接PC與Module上的CON1。此時PC會出現"找到新硬體"的訊息, 接著請安裝該裝置的驅動程式,

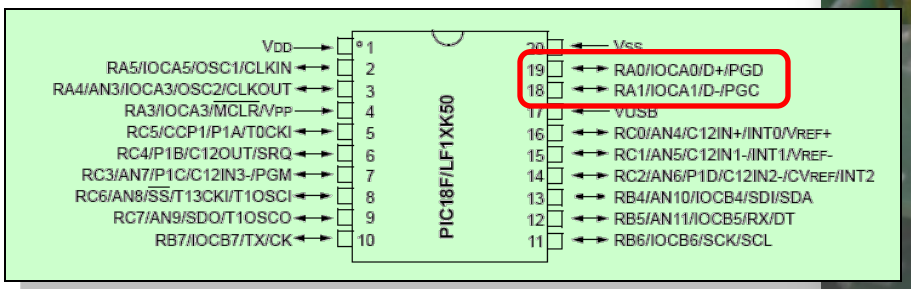
  (Path: 2013 Winter Elite\PIC18F14K50 Project\ Project\Device - CDC - Serial Emulator\inf\)

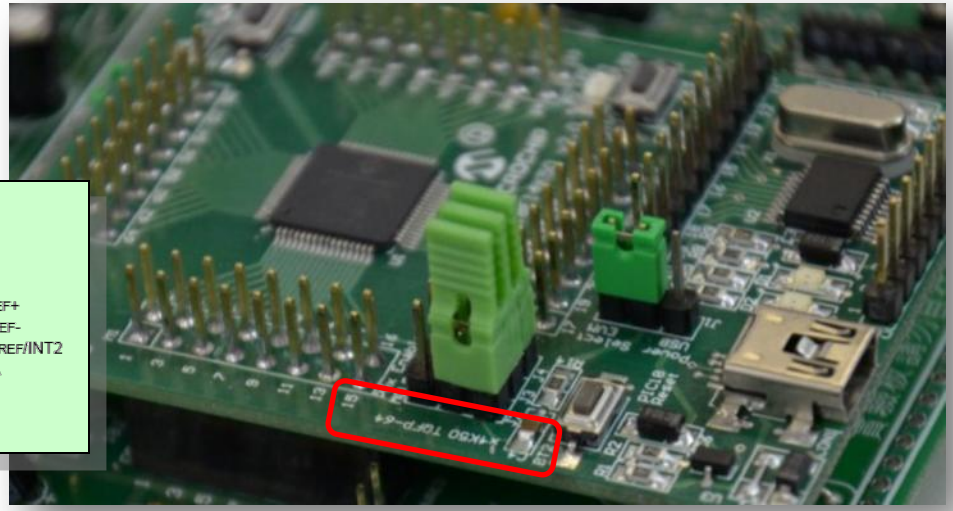
  安裝完後會多出一個COM Port,我們將透過這個COMPort來與PIC24的UART溝通。



PIC18F14K50的Debug與D+, D-

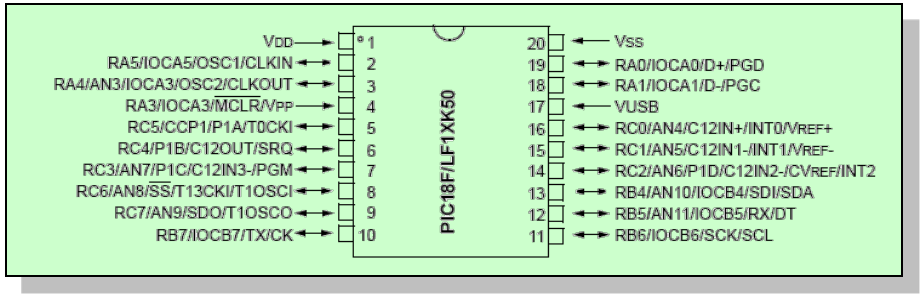
- PIC18F14K50屬於小型封裝的MCU, 所以接腳功能的共用情況很高。USB的訊號腳 D+, D-就剛好跟燒錄/除錯用的腳位PGD, PGC打架。
- 因此燒錄時, 必須先把Mini USB的Cable移除, 以免燒錄失敗。
- 燒錄完成後, 記得把PIM上J2, J3, J4調整回TQFP64的設定, 以免燒錄訊號影響USB的連接。

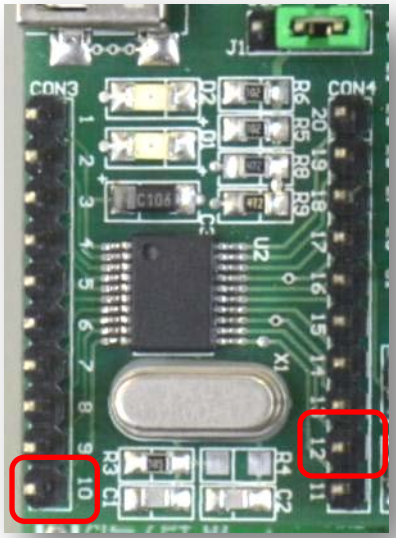




Serial Emulator

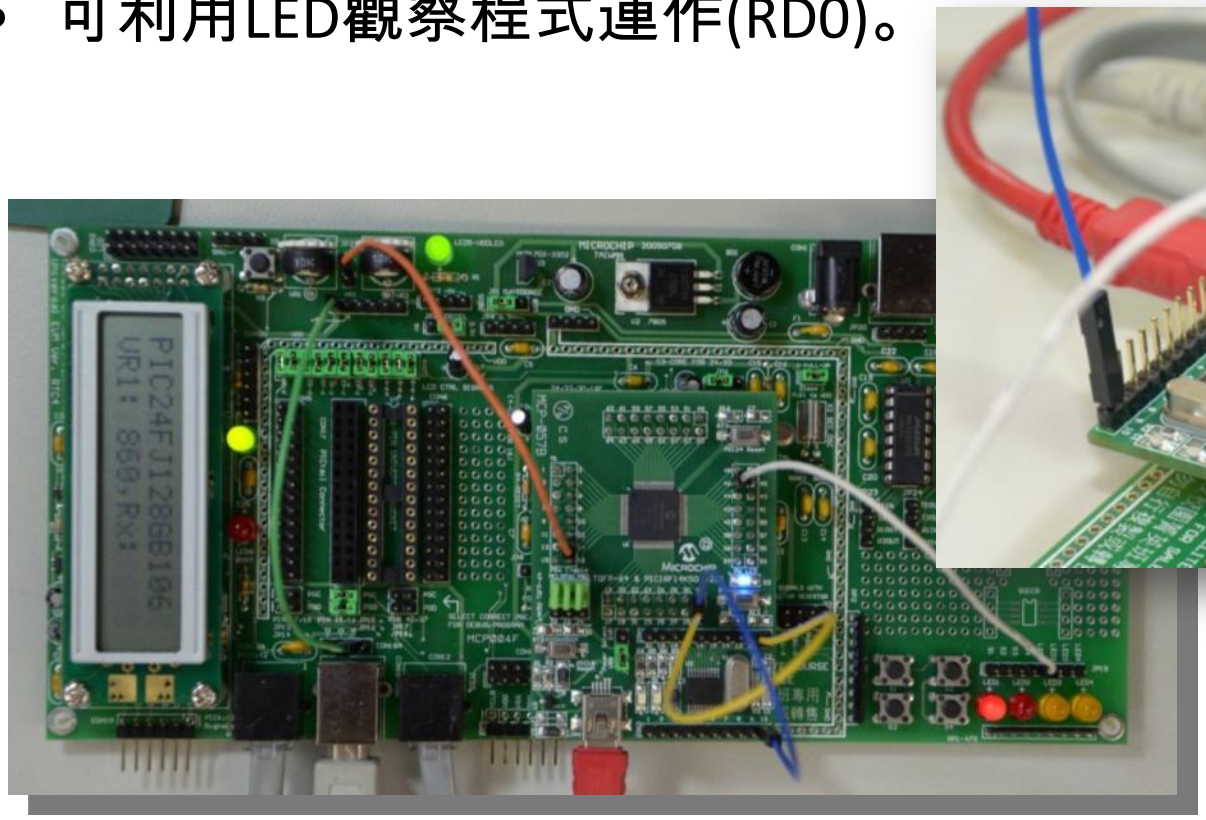
- PIC18F14K50-I/PT內建有USB的Module, 此程式利用USB Module實作了一個CDC類別的USB to UART Bridge。
- Tx腳為Pin10, Rx腳Pin12。D1, D2分別會在傳送與接收資料時閃爍, 指示目前資料傳輸的情形。

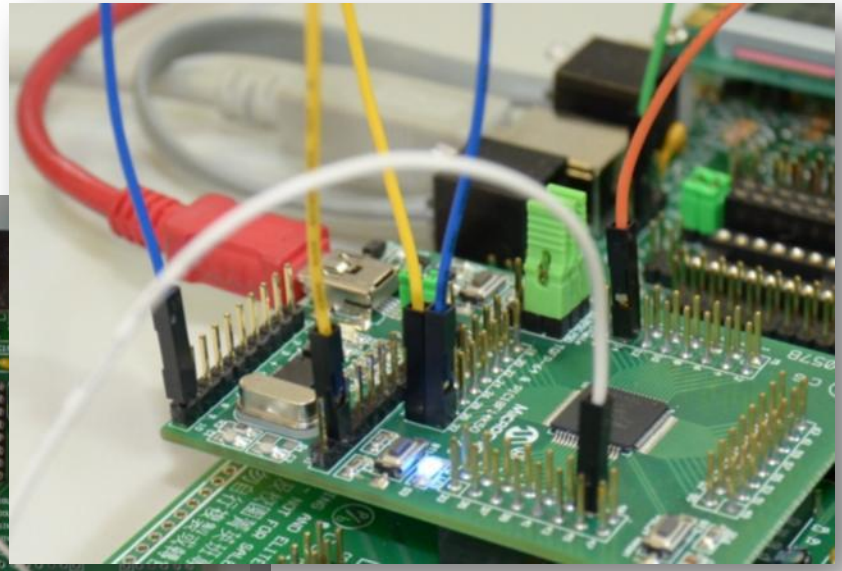




Lab10 UART Step2

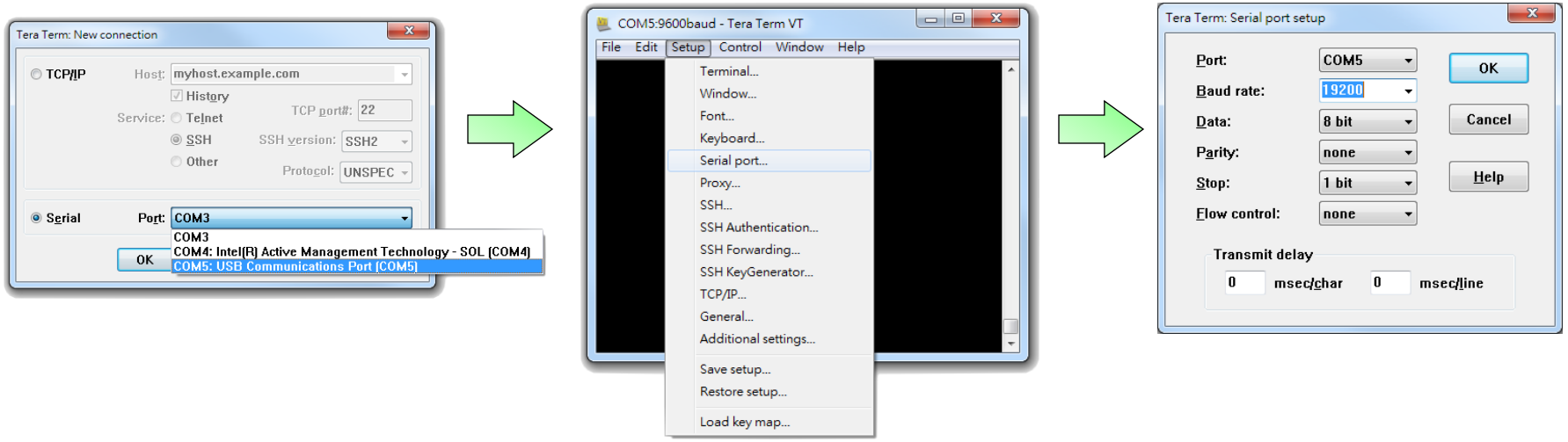
- 連接UART2 Tx, Rx與USB to UART Bridge的Tx, Rx。
  (PIC24 Pin31<-> PIC18 Pin12), (PIC24 Pin32<->PIC18 Pin10) 。
- 可利用LED觀察程式運作(RD0)。





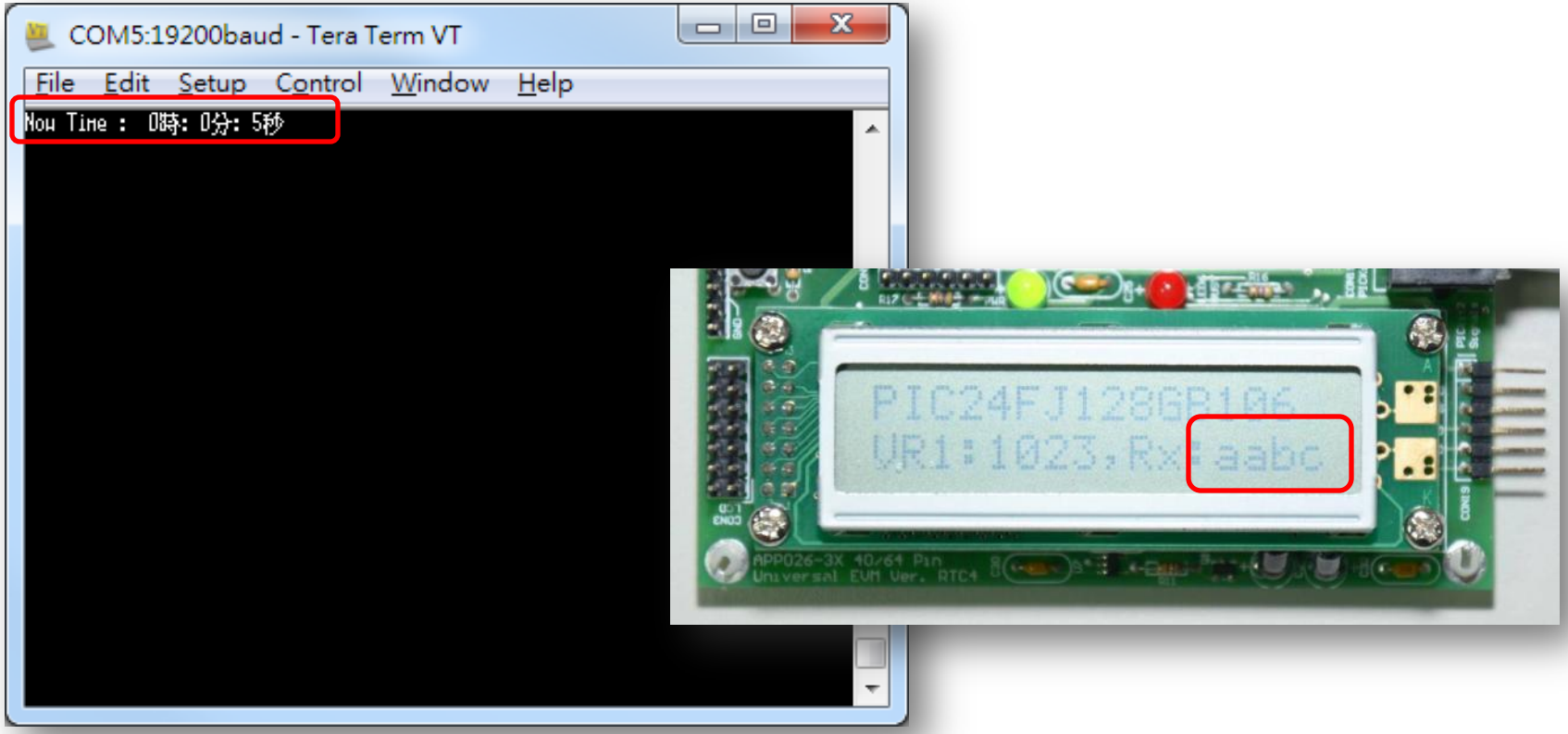
Tera Term

- 由於Windows 7不再提供超級終端機, 因此必須先安裝替代軟體, 才可以監控COM Port的狀態。
- 請先安裝Tera Term, 然後開啟然後Tera Term, 指定COM Port, 設定為19200,N,8,1。



Lab9 UART Step3

- 超級終端機會看到輸出文字,每秒鐘更新一次。
- 在超級終端機上按任意鍵,會在APP026-3x LCD Module上面看到文字輸出。



Lab10 UART Step4

- 鮑率如何計算?

  老話一句,自己算太累了,請Compiler幫我們算。

```
#define SystemFrequency 80000000L / 2
#define UART2Period ( ( SystemFrequency / UART2Baud ) / 16 ) – 1
#define UART2Baud 19200L
```

  SystemFrequency:系統頻率(Fcy), FCY = FOSC / 2 (dsPIC33/PIC24F/24H/24E)。
  Datasheet中有說明,鮑率的計算公式。

- 設定High Baud Mode時(UxMODEbits.BRGH = 0)

  define UART2Period ((SystemFrequency / UART2Baud) / 16) – 1

- 設定Low Baud Mode時(UxMODEbits.BRGH = 1)

  define UART2Period ((SystemFrequency / UART2Baud) / 4) – 1

MICROCHIP
Regional Training Centers

字串轉換

- 範例中的相關函式

```
unsigned char UARTString1[ 100 ];
sprintf( ( char * ) UARTString1 , “Now Time %2d時:%2d分:%2d秒\r” , Hours , Mins , Secs );
 // 格式化輸出函數,功能與printf相同,唯一不同的是,
 // print輸出的目標是stdout(UART),sprintf則是輸出到一個字串陣列中。

UART_PutRAMString( UARTString1 )
{
        while( *String != 0x00 )
        {
                while( BusyUART2( ) );
                WriteUART2( *String++ );
        }
}
// 將字串陣列的字元,逐一透過UART2傳送出去,直到字串結尾(0x00) 。
```

鮑率計算的誤差

- UART模組鮑率計算誤差?
  假設System Clock為16MHz。預設鮑率115,200 bps。計算鮑率與誤差。
  低速率時(BRGH=0),鮑率與誤差計算:

$$\frac{16MHz}{16\times 115200}-1=7.68\cong 8 \qquad \frac{16MHz}{16\times(8+1)}\cong 111111 \qquad \frac{111111-115200}{19200}\times 100\%\cong -3.55\%$$

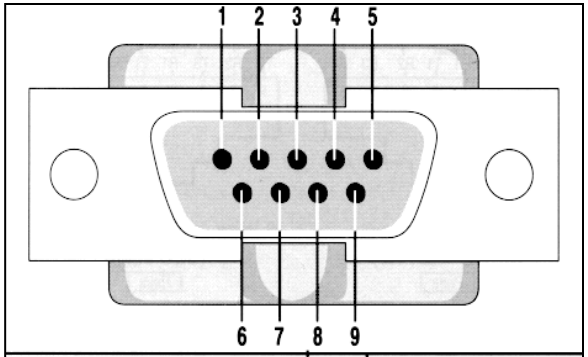
  高速率時(BRGH=1),鮑率與誤差計算:

$$\frac{16MHz}{4\times 115200}-1=33.7\cong 34 \qquad \frac{16MHz}{4\times(34+1)}\cong 114286 \qquad \frac{114286-115200}{115200}\times 100\%\cong -0.79\%$$

- UART誤差容忍度約為3%,在範例中,以低速率設定時,產生的誤差過高。此時就算UART的硬體設定無誤,也會因為過高的誤差導致UART動作不正常。

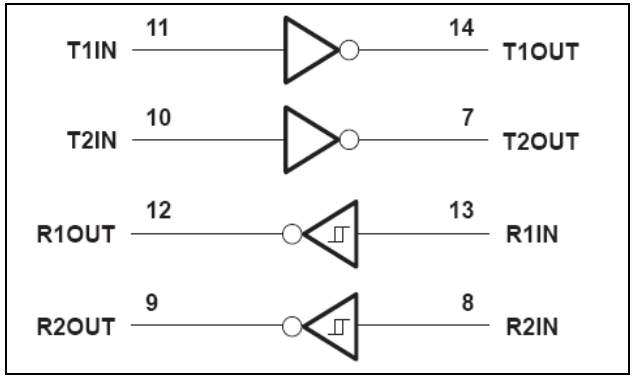
UART 與 MAX232 的連接

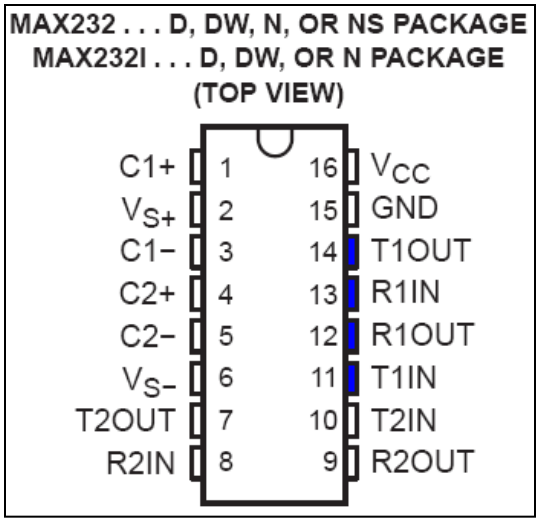
- 也可以使用MAX232來與PC的COM Port連接。
- PC上的RS232所使用邏輯系統與MCU不同。MCU使用正邏輯系統。其邏輯"1"的電壓準位為0.8 V_{DD} ~ V_{DD};邏輯"0"則為V_{SS} ~ 0.2V_{DD}。RS232,使用負邏輯,邏輯"1"的電壓準位為-3~-12V;邏輯"0"則為+3 ~ +12V。
- MCU與RS232因為邏輯系統與電壓位準均不同,因此無法直接連接,必須透過MAX232作電壓位準的調整。



Pin	Signal	Pin	Signal
1	Data Carrier Detect	6	Data Set Ready
2	Received Data	7	Request to Send
3	Transmitted Data	8	Clear to Send
4	Data Terminal Ready	9	Ring Indicator
5	Signal Ground		

http://www.aggsoft.com/rs232-pinout-cable/serial-cable-connections.htm





UART與MAX232的連接

- UART,MAX232與RS232連接器之間的線路連接可參考下圖。在APP026-3x上,MAX232的電源,及電容等均以事先連接完成。但訊號線須由使用者自行以杜邦線連接適當訊號。
- 先使用PPS指定UART2的Rx(RP17,Pin31),Tx(RP10,Pin32)位置,再利用杜邦線連接PIC24與MAX232, (Rx-R1OUT) , (Tx-T1IN)。
- 接著,利用杜邦線連接MAX232與DB9 Connecter, (R1IN- Pin3), (T1OUT-Pin2) 。
- CTS (Clear to Send),RTS (Request to Send)為進行流量控制時所必須使用之訊號。如果需要流量控制時,則必須將CTS,RTS依參考線路所示連接。