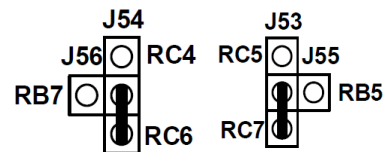


I - La liaison UART

I.a Configuration matérielle

La carte Microchip Explorer 8 comporte une interface USB-to-UART permettant de communiquer avec un PC équipé d'un terminal série. Le microcontrôleur doit être configuré pour être convenablement connecté à cette interface, et vous pourrez alors utiliser son périphérique EUSART (Enhanced Universal Synchronous Asynchronous Receiver Transmitter) intégré via le connecteur micro USB.

Avant toute chose, vérifiez que les cavaliers J53 et J54 de la carte Explorer 8 sont placés comme ci-contre :



Cette configuration connecte la broche RC6 du PIC16F1719 à l'entrée UART RX de l'interface USB-UART, et la broche RC7 à sa sortie UART TX. Une fois ces branchements faits, il ne reste que des configurations logicielles à effectuer dans le microcontrôleur pour établir la connexion.

I.b Configuration logicielle (Microcontrôleur)

La description du module EUSART du microcontrôleur est faite dans [1] p. 354.

Pour faire fonctionner le module EUSART du microcontrôleur, vous devrez effectuer les configurations suivantes :

- 1) Assignment des pins de réception (RX) et de transmission (TX) aux entrées RX/TX du module EUSART. A l'aide du module PPS :
 - a. Connecter RC6 à TX/CK (via le registre RC6PPS)
 - b. Connecter RC7 à RX (via le registre RXPPS)
- 2) Configuration des pins RC6 et RC7 via les registres TRISx, ANSELx, LATx:
 - a. RC6 : Sortie
 - b. RC7 : Entrée numérique
- 3) Configuration de la vitesse de transmission (baudrate) à 9600 bauds à travers trois bits de configuration et un registre 16 bits :
 - a. Bit SYNC (registre TX1STA; on utilisera le module en mode asynchrone, donc SYNC=0)
 - b. Bit BRGH (registre TX1STA)
 - c. Bit BRG16 (registre BAUD1CON)
 - d. Registres SP1BRGH, SP1BRGL (SP1BRG)

Le tableau 31-3 ([1] p. 367) donne la valeur du baudrate en fonction de SYNC, BRGH, BRG16, et SP1BRG ; Le tableau 31-5 ([1] p. 368) donne des combinaisons de configurations pour SYNC, BRGH, BRG16 et SP1BRG pour différentes fréquences d'horloge Fosc et différents baudrates.
- 4) Activation du module EUSART :
 - a. Activation des pins (Bit SPEN, registre RC1STA)
 - b. Activation du *transmitter* (Bit TXEN, registre TX1STA)
 - c. Activation du *receiver* (Bit CREN, registre RC1STA)

5) Utilisation du module EUSART :

- a. La transmission d'1 octet de données se fait en écrivant dans le registre TX1REG
- b. La réception d'1 octet de données se fait en lisant le registre RC1REG

I.c Configuration logicielle (Terminal)

La configuration de la liaison série doit être la suivante (à configurer dans Tera Term) :

- Baudrate : 9600 bauds
- Data length : 8-bit
- Parity : None
- Stop bits : 1
- Flow control : None

II - Exercices

Dans cette série d'exercices, vous utiliserez des transmissions filaires pour faire communiquer des périphériques avec un microcontrôleur. Dans un premier temps, vous établirez une communication que vous pouvez « superviser », entre le terminal d'un PC et le microcontrôleur. Ce genre de liaison permet habituellement de transférer des informations (données, commandes) entre un PC et un système embarqué. Dans un second temps, vous établirez une communication entre le microcontrôleur et un périphérique embarqué sur la carte de développement : ce genre de liaison vous permettra de contrôler le périphérique via le microcontrôleur.

Vous aurez besoin, dans le début de ce TP, d'un émulateur de terminal supportant les connections aux ports série. Téléchargez et installez Tera Term (<https://tera-term.fr.softonic.com/>). Pour son utilisation, référez-vous à l'annexe en fin de sujet.

II.a Liaison UART – transmission de données

Dans ce premier exercice, vous établirez une connexion visant à transmettre des données de la carte Microchip Explorer 8 vers le PC.

- 1) Créer un nouveau projet, nommé TP3a_uart_tx
- 2) Ecrire un programme permettant d'envoyer la chaîne de caractères « Hello, World !\n\r » sur la liaison USB-UART de la carte Explorer 8.
- 3) Vérifier à l'aide de Tera Term que la chaîne de caractères est bien reçue.

II.b Liaison UART – réception de données

Dans ce deuxième exercice, vous programmerez les routines permettant de recevoir et traiter des données envoyées sur l'interface USB-UART de la carte Explorer 8. Puisque ces données peuvent arriver à tout moment, c'est tout naturellement que vous utiliserez des interruptions pour les traiter de façon asynchrone.

Puisque vous avez déjà validé l'envoi de données sur la liaison USB-UART, vous réaliserez un programme « echo », qui reçoit un caractère sur la liaison USB-UART, et le renvoie.¹

¹ Vous noterez que la saisie d'un caractère sur Tera Term n'affiche aucun caractère : les caractères affichés sont uniquement ceux reçus sur la liaison USB-UART. L'objectif du programme « echo » est de : 1. Saisir un caractère dans le terminal ; 2. Le recevoir sur le microcontrôleur ; 3. Le renvoyer sur la liaison USB-UART ; 4. L'afficher sur le terminal.

La réception d'une donnée sur le module EUSART lève le flag d'interruption RCIF. Pour l'exploiter, vous devrez activer le Global Interrupt Enable (GIE), le Peripheral Interrupt Enable (PEIE), et l'Enable spécifique à la réception de données (RCIE).

- 1) Créer un nouveau projet, nommé TP3b_uart_echo
- 2) Ecrire le programme « echo », qui renvoie le caractère reçu sur la liaison USB-UART
- 3) Vérifier à l'aide de Tera Term que l'écho est bien réalisé.

II.c Ecran LCD – Hello, World !

Dans cet exercice, vous apprendrez à afficher des caractères sur l'écran LCD de la carte Explorer 8. Ce périphérique est contrôlé par une liaison SPI. Pour établir cette liaison, vous ne réaliserez pas vous-mêmes les configurations : vous utiliserez des sources (trouvables sur [junia-learning](https://junia-learning.com)) qui vous fourniront des fonctions « haut niveau » vous donnant un accès direct à l'écran LCD.

Avant toute chose, vérifiez que les cavaliers J59, J60 et J61 (sous l'écran LCD) sont branchés.

Téléchargez les sources `lcd.c`, `lcd.h`, `spi.c`, `spi.h`.

Pour utiliser l'écran LCD, vous devrez dans votre programme main :

- 1) Inclure les headers `spi.h` et `lcd.h`
- 2) Appeler les fonctions `SPI_InitializePins()` et `LCD_InitializePins()`
- 3) Appeler les fonctions `SPI_Initialize()` et `LCD_Initialize()`
- 4) Effacer l'écran LCD à l'aide de la fonction `LCD_Clear()`
- 5) Placer le curseur d'écriture de l'écran LCD à l'aide de la fonction `LCD_GoTo(0,0)`
- 6) Ecrire un caractère à l'aide de la fonction `LCD_WriteByte()` ou écrire une chaîne de caractères à l'aide de la fonction `LCD_WriteString()`

- 1) Créer un nouveau projet, nommé TP3c_lcd
- 2) Ecrire un programme qui affiche « Hello, World ! » sur l'écran LCD

II.d Chiffrement/Déchiffrement de code César

Dans ce dernier exercice, vous réaliserez un programme permettant de chiffrer et déchiffrer le code César (chiffrement consistant à décaler chaque caractère d'un message d'un offset constant).

Par exemple, sur l'alphabet 0123456789abcdefghijklmnopqrstuvwxyz, le message « microcontroleur » avec un offset de 1 devient : « njdspdpouspmfvs »

- 1) Créer un nouveau projet, nommée TP3d_cesar
- 2) Réaliser le programme de chiffrement/déchiffrement de code César, répondant aux contraintes suivantes :
 - a. L'utilisateur entre le message à chiffrer/déchiffrer sur le terminal série
 - b. Le message chiffré/déchiffré est renvoyé par la carte Explorer 8 sur la liaison USB-UART, caractère par caractère et s'affiche sur le terminal
 - c. Le potentiomètre de la carte permet de changer la valeur du décalage
 - d. L'appui sur un bouton permet de commuter entre le mode « chiffrement » et le mode « déchiffrement »
 - e. L'écran LCD affiche à tout instant la valeur de l'offset et le mode d'encodage en cours (chiffrement/déchiffrement)
 - f. L'alphabet géré par l'application est 0123456789abcdefghijklmnopqrstuvwxyz
 - g. La transmission par l'utilisateur d'un caractère entre 'a' et 'z' en majuscule l'interprète comme ce même caractère en minuscule
 - h. Les caractères hors alphabet sont ignorés

Références

[1] Fiche technique de la famille de microcontrôleurs PIC16(L)F1717/8/9

http://ww1.microchip.com/downloads/en/DeviceDoc/PIC16F1717_8_9-data-sheet-40001740C.pdf

[2] Microchip Developer Help

<http://microchipdeveloper.com/8bit:peripherals>

[3] MPLABX XC8 User guide

<https://ww1.microchip.com/downloads/en/devicedoc/50002053g.pdf>

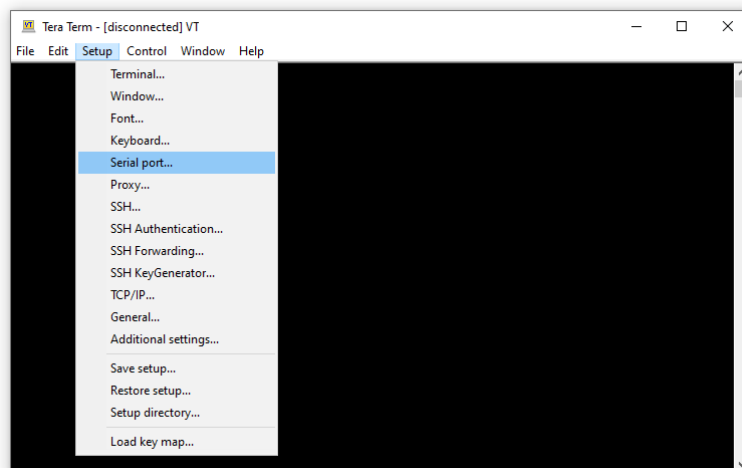
[4] Fiche technique de la carte Microchip Explorer 8

<https://ww1.microchip.com/downloads/en/DeviceDoc/40001812B.pdf>

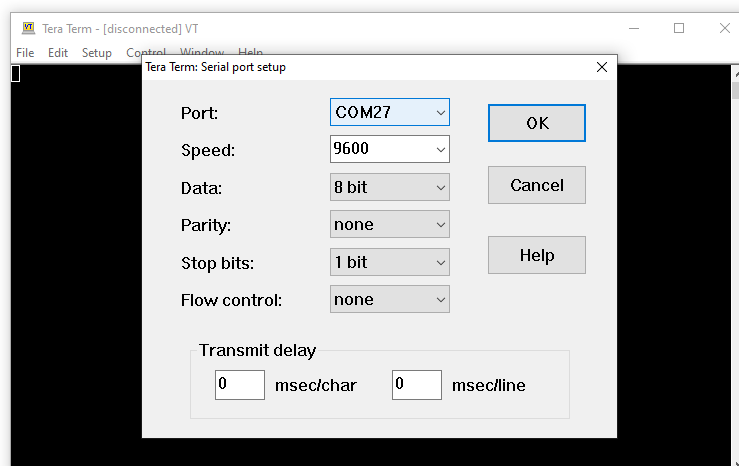
Annexe – Tera Term

Téléchargement et installation : <https://ttssh2.osdn.jp/index.html.en>

Pour configurer la communication série, Setup > Serial port...



Sélectionner le port virtuel COM correspondant à la carte Explorer 8, et configurer les paramètres de la communication :



La liaison série est opérationnelle dès l'appui sur OK.