



Malaga University
Electronic Technology Department



ETS Telecommunications Engineering

Bachelor's Degree in Telecommunications Systems Engineering

microcontrollers

Course 2022/2023

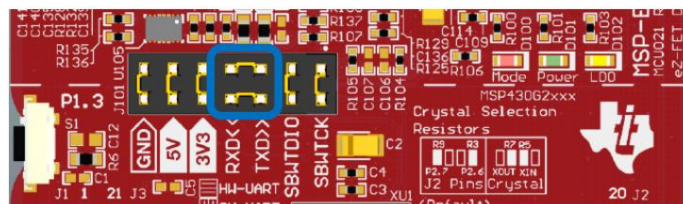
Practice 5: UART

(v2)

1. DESCRIPTION

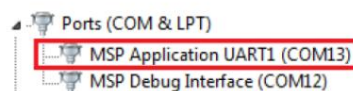
Practice 5 of the subject presents the basic concepts of the **UART** of the microcontroller. It will be developed on the *LaunchPad* and BBP boards, using the 4 Sx buttons and the 4 LEDx of the same ($x = 3..6$). It must be taken into account that several buttons will never be pressed simultaneously, so it is not necessary to manage this situation.

To carry out this exercise, the UART of the microcontroller must be connected to the PC, which is done by turning the RXD and TXD jumpers of connector J101 of the *LaunchPad board* 90°, as can be seen in the following figure:

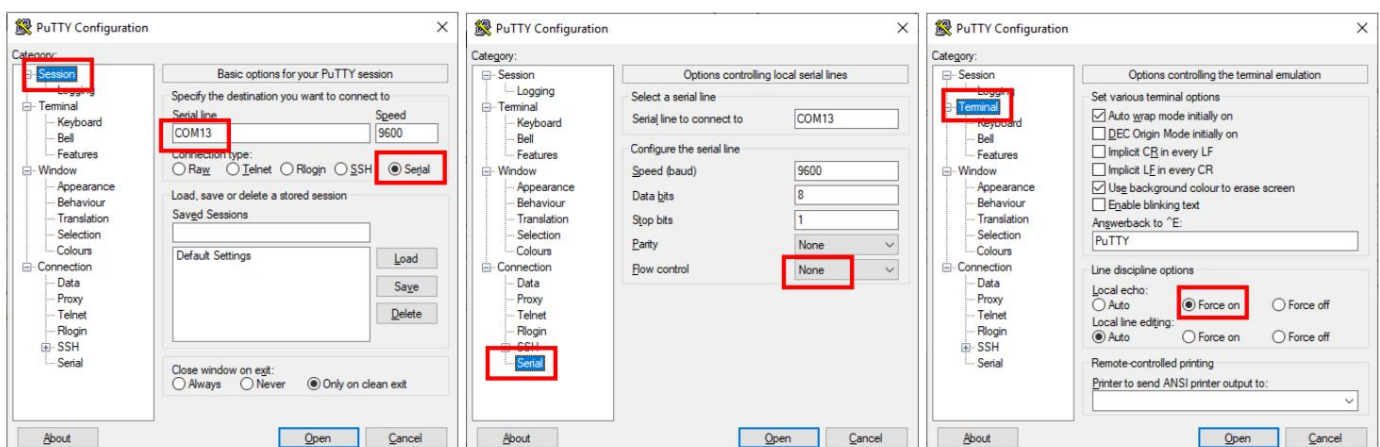


Likewise, a virtual serial port (named "MSP Application UART1") will appear on the PC, which will be connected to the UART of the microcontroller, and which can be identified using the Configuration Manager.

Windows devices, as shown in the following figure:



Finally, to have a terminal on the PC that can send and receive characters to the UART of the microcontroller, the Putty application is needed, available in the Virtual Campus of the subject. The configuration of the putty application is done through the following options:



IMPORTANT NOTE: In order to develop this practice it is necessary that the first instructions that are executed in the "main" function of the program are:

```
WDTCTL = (WDTPW|WDTHOLD);           // Stop the watchdog
DCOCTL = 0;                          // Set the frequency to 1MHz
BCSCTL1=CALBC1_1MHZ;
DCOCTL=CALDCO_1MHZ;
```

To implement this practice, the modules described below must be developed.

Q5.1

Description:

When the character 'x' is received it will switch the LEDx.

Considerations:

- The management of the UART will be carried out by interruptions.
- The UART will use the clock signal SMCLK at 1MHz, and will be set to 9600bps, 8 bits per character, without parity and 1 stop bit.

Q5.2

Description:

When the Sx button is pressed the character 'x' will be sent

Considerations:

- The management of the buttons and the UART will be carried out by interruptions.
- The pushbuttons will have a 32ms debounce, implemented through the watchdog with the signal SMCLK clock at 1MHz.
- The UART will use the clock signal SMCLK at 1MHz, and will be set to 9600bps, 8 bits per character, without parity and 1 stop bit.

Q5.3

Description:

An inverted echo of 4 characters will be implemented, so that once any 4 characters are received they will be sent in the reverse order in which they were received.

Considerations:

- The management of the UART will be carried out by interruptions.
- The UART will use the clock signal SMCLK at 1MHz, and will be set to 9600bps, 8 bits per character, without parity and 1 stop bit.