## Lab 5: Communication Peripherals

## Activity 1: UART configuration

Aim: Learn to how to configure UART communication peripheral on STM32L4 DISCO board using STM32CubeIDE.

Objectives:

- 1- Learn how to configure the UART
- 2- Use DMA to transmit data through UART
- 3- Testing the functionality by TeraTerm (Serial Comm software).

**Step 1:** Create a project in STM32CubeIDE. In Pinout & Configuration, Expand 'Connectivity' menu and select 'UART2' and in Mode window select 'Asynchronous' in Mode tab then relocate the green highlighted pins by mouse while holding **CTRL** key on keyboard to Pin PD5 for Tx and PD6 for RX alternatively Select the PD5 as UART2\_Tx and PD6 as UART2\_RX. Note the parameter setting in configuration window.



Step 2: In configuration window select 'DMA setting' window 'Add' DMA channel for 'UART2 TX' and select 'Normal' mode for DMA request setting.



Step 3: In configuration window select 'parameter setting' and keep the default settings.



Basic Parameters Baud Rate Word Length Parity Stop Bits

115200 Bits/s 8 Bits (including Parity) None 1

**Step 4:** Generate the code and add the code section given below:

Declare global variable:

```
/* USER CODE BEGIN PV */
uint8_t Message_Tx[]={"UART 2 STM3214756VG test message\r\n"};
 /* USER CODE BEGIN 2 */
/**
 * @brief Send an amount of data in DMA mode.
  * @note When UART parity is not enabled (PCE = 0), and Word Length is
configured to 9 bits (M1-M0 = 01),
 *
           the sent data is handled as a set of u16. In this case, Size must
indicate the number
 *
           of u16 provided through pData.
 * @param huart UART handle.
  * @param pData Pointer to data buffer (u8 or u16 data elements).
 * @param Size Amount of data elements (u8 or u16) to be sent.
 * @retval HAL status
if (HAL_UART_Transmit_DMA(&huart2, Message_Tx, sizeof(Message_Tx)/sizeof(Message_Tx[0])-1) != HAL_OK)
  {
    Error Handler();
  }
```

Run and build the project.

Todo: State what other function are available for UART peripherals.

**Step 5:** Open **Device manger** to know the serial port number for 'STMicrelectronics STLink Virtual COM Port'.

봂	🛃 Device Manager –						
<u>F</u> ile	<u>A</u>	ction <u>V</u> iew <u>H</u> elp					
	-						
v					^		
	>	Audio inputs and outputs			- 10		
1	5 4	Batteries					
	5	Bluetooth					
	, (	Cameras					
	, [	Computer					
	,	Digital Media Devices					
	, .	Disk drives					
	>	Display adapters					
	> 1	Firmware					
	> 📕	Human Interface Devices					
	> =	IDE ATA/ATAPI controllers					
	> 🗵	Keyboards					
	> (	Mice and other pointing devices					
	>	Monitors					
	>	P Network adapters					
	>	Portable Devices					
	~ Ű	Ports (COM & LPT)					
		Standard Serial over Bluetooth link (COM4)					
		STMicroelectronics STLink Virtual COM Port (COM7)					
	> 🖪	Print queues					
	> [	Processors					
	> 🚦	Sensors					
	>	Software components					
	>	Software devices					
	5 1	Sound video and dame controllers			~		

**Step 5:** <u>Install</u> and Open 'TeraTerminal' software and select the 'Serial' option and in port menu select 'STMicrelectronics STLink Virtual COM Port'.

OTCHIP	Hos <u>t</u> :	myhost.exar	nple.com
		⊠ Hist <u>o</u> ry	
	Service:	⊖ Te <u>I</u> net	TCP <u>p</u> ort#: 22
		● <u>s</u> sh	SSH version: SSH2
		○ Other	IP version: AUTO
● S <u>e</u> rial	Po <u>r</u> t:	COM7: STMi	croelectronics STLink Virtu

In TeraTerminak software click on Menu 'Setup' and select 'Serial Port'.



Configure the speed to '115200' and verify the other parameters to STM32L4 UART parameters and finally click on 'New Setting'.

Port:		COM7	$\sim$	New setting
Sp <u>e</u> ed:		115200	~	<u></u>
<u>D</u> ata:		8 bit	$\sim$	Cancel
P <u>a</u> rity:		none	$\sim$	
<u>S</u> top bits	:	1 bit	$\sim$	<u>H</u> elp
<u>F</u> low cor	ntrol:	none	$\sim$	
	0	msec/ <u>c</u> har	0	msec/line
Device Fr	iendly Na stance ID	ame: STMicroe ): USB\VID_04	electron 83&PID	ics STLink Virtual COM F _374B&MI_02\6&329DAEI

Press **Reset Button number of times** on STM32L4 DISCO kit and view the transmitted message.

💆 COM25 - Tera Term VT					_	×	
<u>F</u> ile	<u>E</u> dit	<u>S</u> etup	C <u>o</u> ntrol	<u>W</u> indow	<u>H</u> elp		1
UART 2 UART 2 UART 2 UART 2 UART 2 UART 2	STH321 STH321 STH321 STH321 STH321 STH321 STH321	4756VG to 4756VG to 4756VG to 4756VG to 4756VG to 4756VG to 4756VG to	est nessage est nessage est nessage est nessage est nessage est nessage				^

**Step 6:** Change the UART2\_TX pin to 'PA2' and generate the code and run the project. Attach logic analyzer 'D0' to pin 'PA2' and GND to GND pins. Click on 'Add protocol decoder' and from 'decoder selector' window double click on 'UART' protocol.



**Step 6:** Now click on 'UART' on left and look into 'Tx menu' for 'D0'. From **Data format** drop down list select 'ascii'.

Transie Sessie	on 1 - PulseView		- 0 )
	Run 🕺 Session 1 🛛		
Session 1	Name UART	đ×	Decoder Selector
-	Color	7. I G samples V 2 MHz V	٩
	UART 🗢 00 µs	+468200 µs +468300 µs +468400 µs +468600 µs +468600 µs +468700 µs	Decoder Name
	RX (UART receive line)		RGB LED ( RGB LED string decoder ( RTC-8564 Epson RTC-8564 JE/NB S/PDIF Sony/Philips Digital Interf
	beta bits 8 - Parity none -	2009 €02080008 0000000 €0000008 €0000008 0000000 0000000 €00 5 70 \$ << >73 75 << >75 72 75 << >75 73 75 <	SD card (SD         Secure Digital card (SD           SD card (SP         Secure Digital card (SPI           SDA2506         Siemens SDA 2506-5           SPI         Serial Peripheral Interface
	Stop bits 1.0 Bit order Isb-first •		SPI flash/EE SPI flash/EEPROM chips SSI32 Synchronous Serial Interf. ST7735 Sitronix ST7735 Stepper motor Stepper motor position /
	Data format Invert RX		SWD Serial Wire Debug SWIM STMB SWIM bus 755xx RFID T55xx
			TDM audio Time division multiplex TI TCA6408A Texas Instruments TCA64 TI TLC5620 Texas Instruments TLC56
			Timing Timing calculation with UART Universal Asynchronous USB packet Universal Serial Bus (LS/ USB PD USB Power Delivery
			USB request Universal Serial Bus (Ls) Universal Asynchronous Receiver Transmitter (uat) Asynchronous serial Aus UART (Universal Asynchronous Receiver Transmitter) is a simple serial communication protocol which allows two
			This decoder should work on all "UART-like" async protocols with one start bit (0), 5-9 databits, an (optional) parity and one or more stop bits (1), in this order.
			It can be run on one signal line (RX or TX) only or on two lines (RX + TX).
			There are various standards for the physical layer specification of the sionals_including RS232_(TTL) HART_RS485_
	<	>	

**Step 7:** Press the reset button on STM32L4 Disco and zoom in the data area and view the output as below.

Session 1 - PulseView	- 0 ×
Kun 🔊 Session 1 🔝	
Sesson 1 df x	Decoder Selector & X
📄 🔻 🛅 👻 🧟 🔲 🗓 🦷 👘 👘 👘 💦 Saleae Logic 🔍 🚀 🖉 G samples 🔹 12 MHz 🔹 🦇	9
+1595500 µs +1596500 µs +1596500 µs +1597500 µs +15977500 µs	Decoder Name
	SPI         Serial Reipheral Interface           SPI flast/EE         SPI flast/EEROM chips           S322         Synchronous Serial Interf           SPT flast/EEROM chips         Stronis ST735           Stronis ST735         Stronis ST735           SWIM         Stronis Depart motor position /           SWIM         Strik Swim Debug           SWIM         Stronis ST55xx           TDM daulo         Trade Stronis
	TI CG4600         Texas instruments ICG460.           TI LC5620         Texas instruments ICS620           Timing calculation with fr         URAPT           Universal Synchronous B         USB Power Delivery           USB Power Delivery         USB Power Delivery           USB request         Universal Senal Bus (LS/F           USB regregat         Universal Senal Bus (LS/F           USB signalit.         Users (LS/F           Viegand         Uricersal Senal Bus (LS/F           Viegand         Uricersal Senal Bus (LS/F           Viegand         Users (LS/F           US Cox IX2444M/P         XP           X20         Zlog 280 CPU
	Universal Asynchronous Receiver/ Transmitter (uart) Asynchronous, serial bus.
8- 8-	UART (Universal Asynchronous Receiver A Transmitter) is a simple serial communication protocol which allows two devices to talk to each other.
	This decoder should work on all "UART-like" async protocols with one start bit (0), 5-9 databits, an (optional) parity bit, and one or more stop bits (1), in this order.
	It can be run on one signal line (RX or TX) only, or on two lines (RX + TX).
۲ ۲	layer specification of the sionals, including RS232, (TTL) HART, RS4RS
■ P Type here to search O III O ≪ C O @ □ 4 E T to	へ 電 🦟 印) ENG 11:31 22/10/2021 <b>見</b>

To do: View the data protocol for even and odd parity.