ASCLIN_UART_1
for KIT_AURIX_TC375_LK
UART Communication via ASCLIN module

AURIX™ TC3xx Microcontroller Training
V1.0.0

Scope of work

An ASCLIN module configured for UART communication sends "Hello World!" and receives the string via the internal loopback.

The string "Hello World!" is sent and received via UART through one pin due to the internal loopback. The data can be visualized using an oscilloscope.

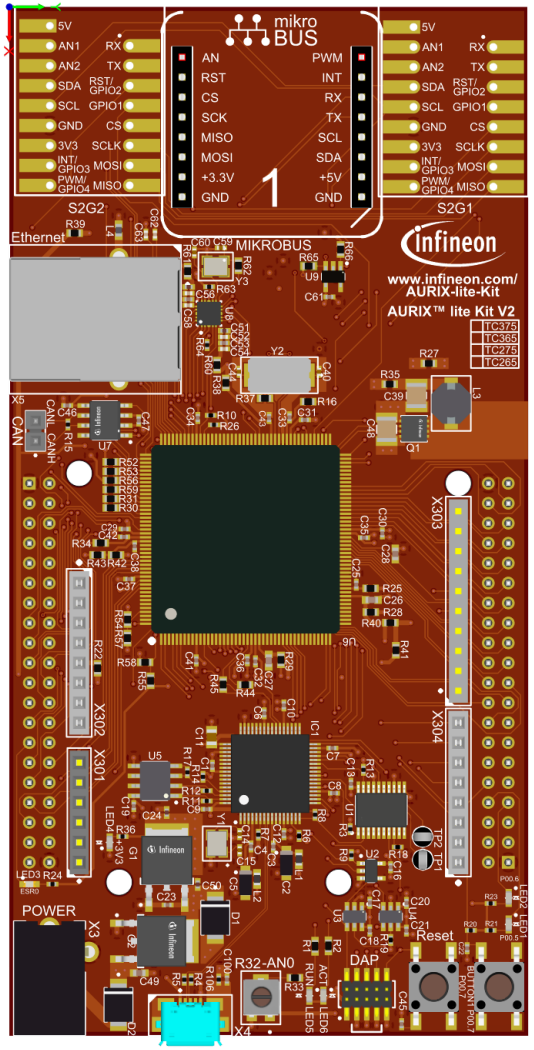
Introduction

- The Asynchronous/Synchronous Interface (ASCLIN) module enables asynchronous/synchronous serial communication with external devices. Among others, it supports asynchronous reception/transmission (UART) for communication.

- For test purposes, the transmit pin (TX) and receive pin (RX) can be shorted internally on-chip (loopback mode).

Hardware setup

This code example has been developed for the board KIT_A2G_TC375_LITE.



Implementation

Configuration of the ASCLIN module:

Configuration of the ASCLIN module for UART communication is done in the setup phase by initializing an instance of the ***IfxAsclin_Asc_Config*** structure with the following parameters:

- ***baudrate*** – structure to set the actual communication speed in bit/s
- ***interrupt*** – structure to set:
  - transmit and receive interrupt priorities (***txPriority***, ***rxPriority***)
  - ***typeOfService*** – defines which service provider is responsible for handling the interrupt, which can be any of the available CPUs, or the DMA
- ***pins*** – structure to set which GPIO port pins are used for the communication
- ***rxBuffer***, ***rxBufferSize***, ***txBuffer***, ***txBufferSize*** – to configure the buffers that will hold the incoming/outgoing data

The function ***IfxAsclin_Asc_initModuleConfig()*** fills the configuration structure with default values and ***IfxAsclin_Asc_initModule()*** initializes the module with the user configuration.

All the above functions can be found in the iLLD header ***IfxAsclin_Asc.h***.

Implementation

The UART send/receive function:

- Sending the string “Hello World!” is implemented inside the function ***send_receive_ASCLIN_UART_message()*** which is called once after the initialization of the ASCLIN module.

- This function calls ***IfxAsclin_Asc_write()*** and ***IfxAsclin_Asc_read()*** which are provided by the iLLD header ***IfxAsclin_Asc.h***.

Implementation

The UART send/receive function:

- The UART frame configured for 115200/8-N-1 consists of different parts:
  - One start bit which is “0”
  - Eight bits of data
  - One stop bit which is “1”

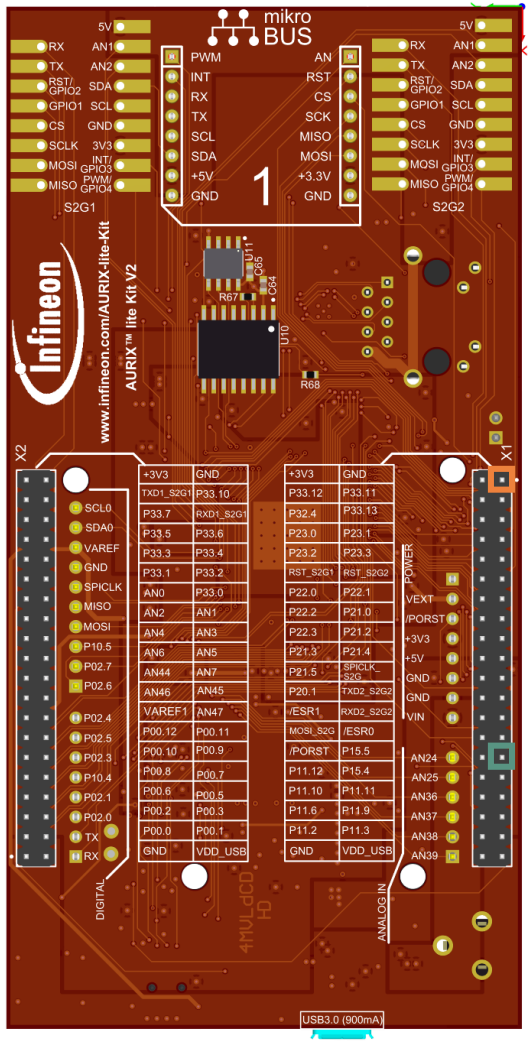
- Each time when the last byte is taken out of the transmit FIFO (size is 16-bytes), the Transmit FIFO Level (TFL) flag is set and the interrupt service routine ***asclin0TxISR()*** is entered. The ISR calls ***IfxAsclin_Asc_isrTransmit()*** which refills the FIFO with the remaining bytes to be transmitted and clears the interrupt flag.

- Each time when an UART byte is received, the Receive FIFO Level (RFL) flag is set and the interrupt service routine ***asclin0RxISR()*** is entered. The ISR calls ***IfxAsclin_Asc_isrReceive()*** which moves the received byte to the global array ***g_ascRxBuffer*** and clears the interrupt flag.

Run and Test

An oscilloscope probe must be connected to the UART TX/RX pin (P15.5) to observe the UART signal.

	X1		
	+3V3	2 | 1	GND
	P33.12	4 | 3	P33.11
	P32.4	6 | 5	P33.13
	P23.0	8 | 7	P23.1
	P23.2	10 | 9	P23.3
RST_S2G1 -	P23.4	12 | 11	P23.5 - RST_S2G2
	P22.0	14 | 13	P22.1
	P22.2	16 | 15	P21.0
	P22.3	18 | 17	P21.2
MDIO -	P21.3	20 | 19	P21.4
	P21.5	22 | 21	P20.10 - SPICLK_S2G
	P20.1	24 | 23	P20.0 - TXD2_S2G2
	/ESR1	26 | 25	P20.3 - RXD2_S2G2
MOSI_S2G -	P20.14	28 | 27	/ESR0
Reset -	/PORST	30 | 29	P15.5
CLK50 -	P11.12	32 | 31	P15.4
RX_D0 -	P11.10	34 | 33	P11.11 - CRS_DV
TX_EN -	P11.6	36 | 35	P11.9 - RX_D1
TX_D1 -	P11.2	38 | 37	P11.3 - TX_D0
	GND	40 | 39	VDD_USB



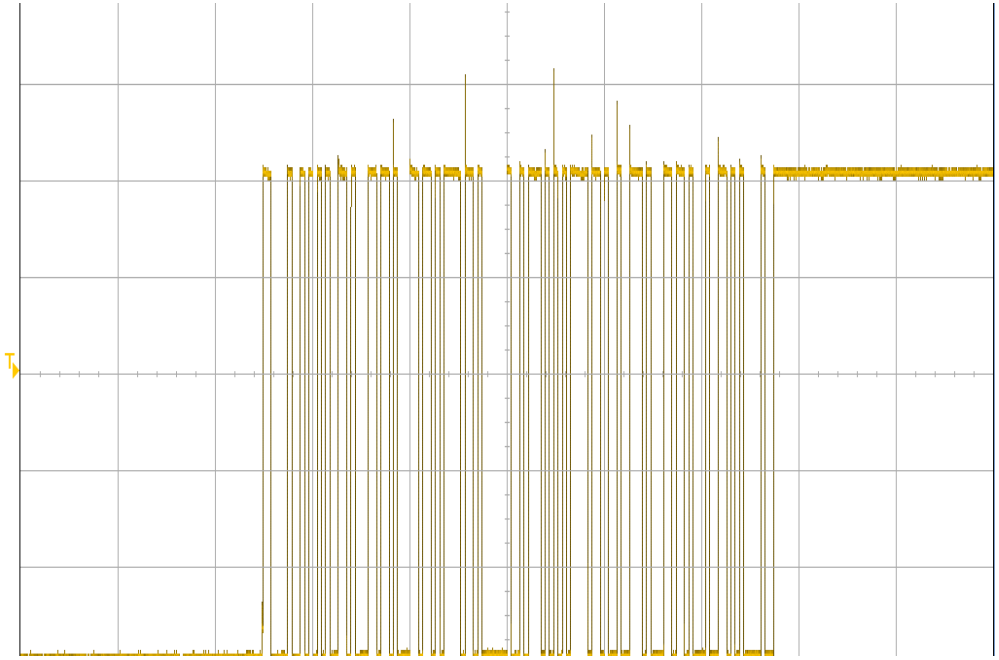
Ground

TX/RX

Run and Test

After code compilation and flashing the device, perform the following steps:

- Connect the oscilloscope probe to the TX/RX pin (P15.5)
- Reset and run the program by pressing the PORST push button
- Check the oscilloscope for the UART signal:



→ "Hello World!"

Run and Test

An additional test without using an oscilloscope can be performed with the debugger.

- Before transmission, the buffer ***g_txData*** is filled with the message "Hello World!" and the buffer ***g_rxData*** is empty.

- After transmission, both buffers should hold the same message:
  - By using the debugger, you can watch the content of both buffers before and after transmission by setting a breakpoint to ***send_receive_ASCLIN_UART_message()***.
  - When reaching this breakpoint, check the content of both buffers (it should be different).
  - After stepping over this function, the content of the buffers must be equal.

References

- AURIX™ Development Studio is available online:
- https://www.infineon.com/aurixdevelopmentstudio
- Use the *„Import...“* function to get access to more code examples.

- More code examples can be found on the GIT repository:
- https://github.com/Infineon/AURIX_code_examples

- For additional trainings, visit our webpage:
- https://www.infineon.com/aurix-expert-training

- For questions and support, use the AURIX™ Forum:
- https://www.infineonforums.com/forums/13-Aurix-Forum

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.

Edition 2021-03
Published by
Infineon Technologies AG
81726 Munich, Germany

Do you have a question about this document?
Email: erratum@infineon.com

Document reference
ASCLIN_UART_1_KIT_TC375_LK

IMPORTANT NOTICE
The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics ("Beschaffenheitsgarantie") .

With respect to any examples, hints or any typical values stated herein and/or any information regarding the application of the product, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

In addition, any information given in this document is subject to customer's compliance with its obligations stated in this document and any applicable legal requirements, norms and standards concerning customer's products and any use of the product of Infineon Technologies in customer's applications.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office (www.infineon.com).

WARNINGS
Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.