TLE9893_2QKW62S_UART_PRINTF

About this document

Scope and purpose

The aim of this guide is to present the scope, the implementation, the algorithm and a demonstration of the **TLE9893_2QKW62S_UART_PRINTF** example code for the TLE989x Infineon Embedded Power ICs based on Arm® Cortex® M3. This example code can be found in the Keil µVision Pack Installer.

The full functionalities and characteristics of the embedded power devices are described in the datasheets and user's manual. Please refer to these documents for more detailed information. Furthermore, a low level (line-by-line) description of the code is not the aim of this document, although occasionally some codeblocks might be reported if necessary to the comprehension.

Note: The following information is given as a hint for the implementation of the system only and shall not be regarded as a description or warranty of a certain functionality, condition or quality of the referred devices or presented software example.

Intended audience

Design engineers, system engineers, embedded power designers

Table of contents

1 Introduction

A string HelloWorld is written to the standard output. A user-defined stdout target is implemented in the UART module. The redirection is established via the compiler-I/O-STDOUT RTE setting.

The pin P1.1 used as output is the debug pin on the USB device.

Figure 1 shows the transmission of the first character ('H' = 0x46 = 'b01001000') of the "HelloWorld" string including start and stop bit.
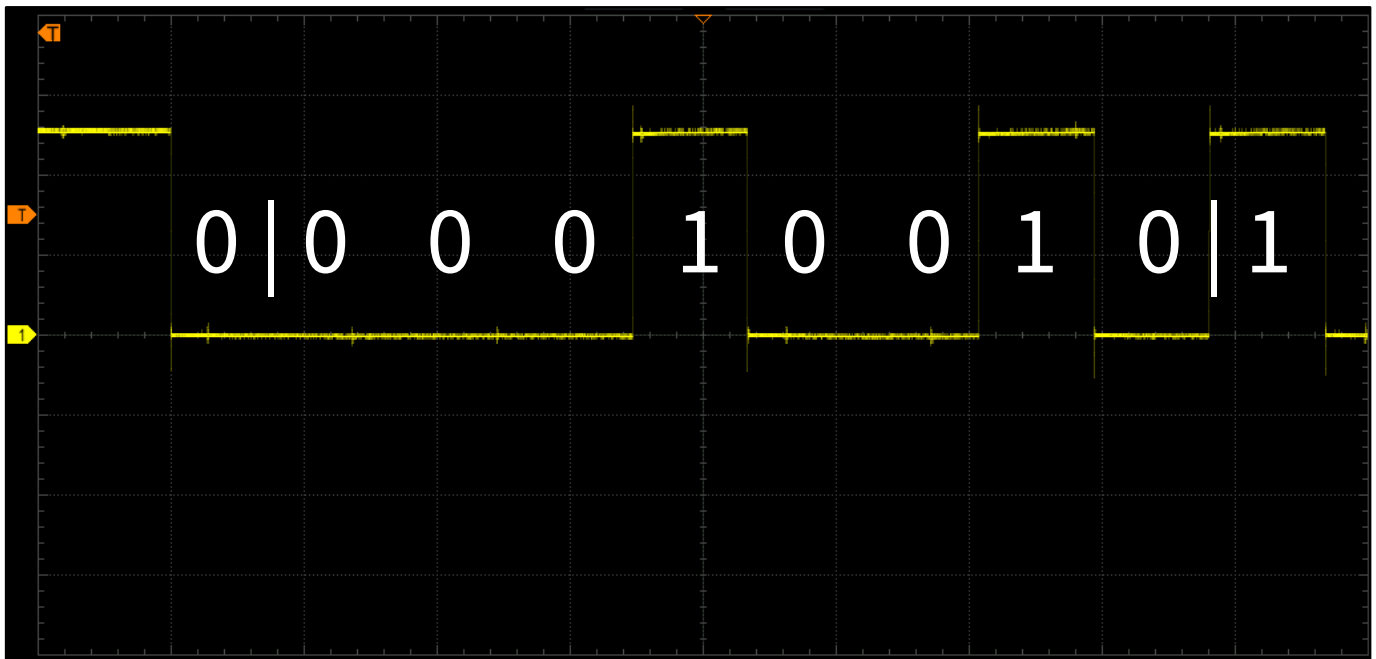

Figure 1 Capture of pin P1.1

2 Hardware

This chapter shows how to run the TLE9893_2QKW62S_UART_PRINTF example with the TLE988X/TLE989X evaluation board. For this the project must be opened and compiled.

Figure 2 shows the TLE988X/TLE989X evaluation board. The application code must be loaded via a debugger (e.g. ULINK or J-Link) to the board. The board must be powered with 12V (red and black connections).
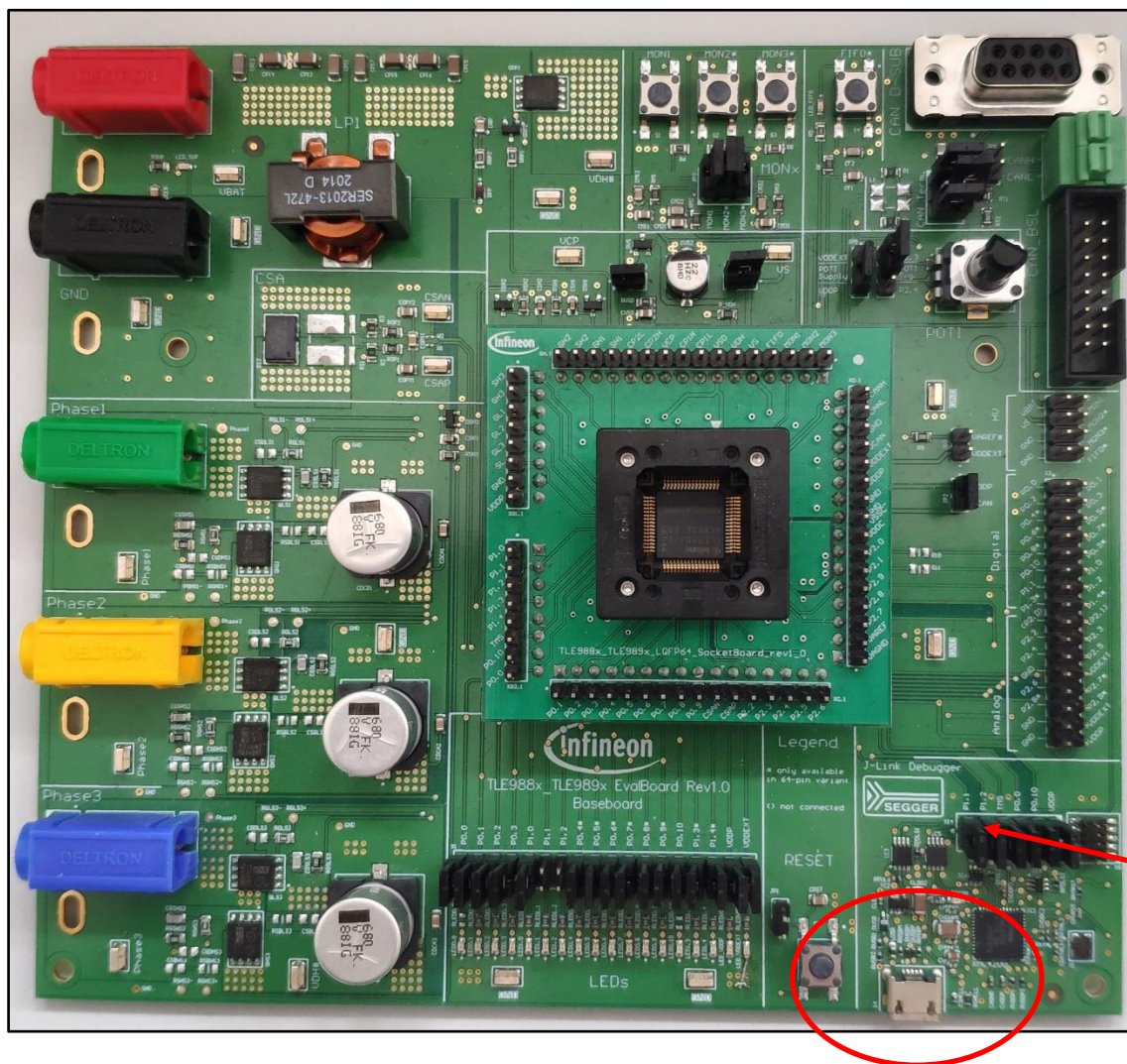

Figure 2 TLE988X/TLE989X evaluation board

The output pin P1.1 can be tapped on the red arrow.

Alternatively, a USB connection can be established to a local PC, which emulates a virtual COM port. The relevant COM device number can be identified via the Device Manager on Windows systems or the dmesg tool on Unix based operating systems.

In order to show the output on a command console, free tools like Putty or TeraTerm can be used. The UART1 in this example is configured with:

- a transmission baud rate of 115200,

- 8 data bits,

- 1 stop bit,

- no parity and no flow control.

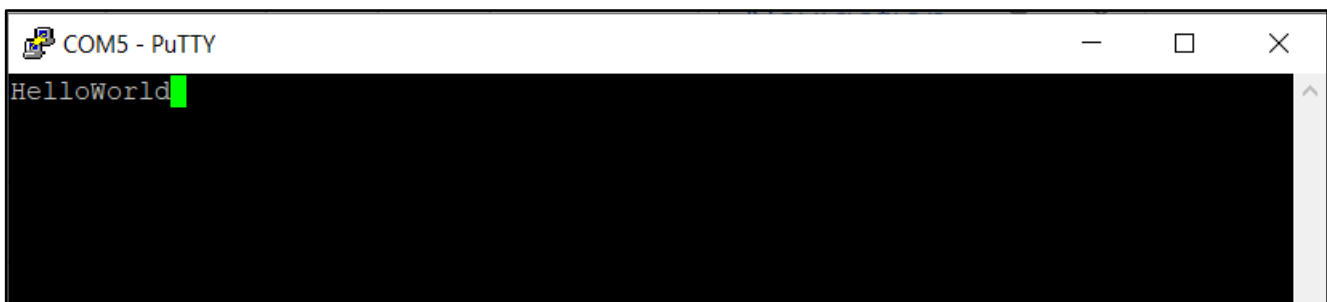Figure 3 shows the output of the "HelloWorld" string on a Putty command console.

Figure 3 Output of HelloWorld on a Putty console

3 Implementation

This chapter shows the process to follow to get a working TLE9893_2QKW62S_UART_PRINTF example.

3.1 Get the example via the Pack Installer for Keil

Open the Pack Installer within the Keil IDE. See Figure 4 below.

Choose the appropriate device (here TLE9893_2QKW62S) on the left-hand side. On the right-hand side, select the tab Examples, where you can access the TLE9893_2QKW62S_UART_PRINTF example.
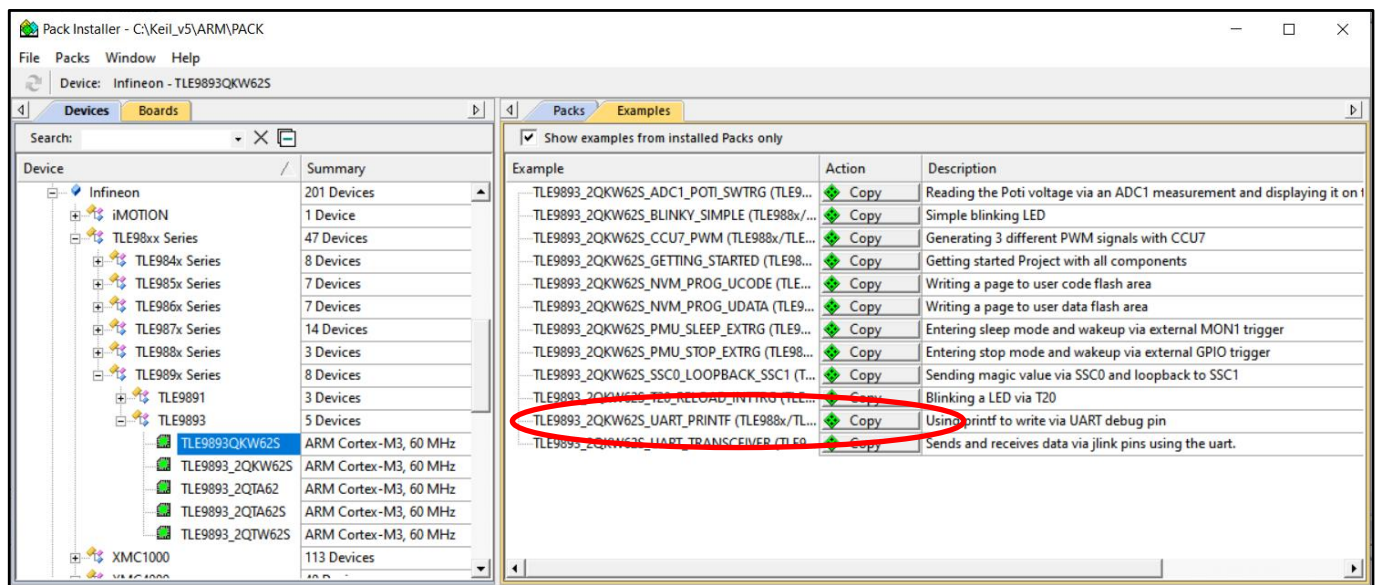
Clicking on "Copy" will copy the example on your computer and open it.

Figure 4 Keil Pack Installer

In order to redirect the stdout functions - the printf call in the example, adjust the runtime environment setting for the compiler within the Keil IDE. Select the option "User" under Compiler -> I/O -> STDOUT (see Figure 5 RTE settings for stdout).

In the TLE9893_2QKW62S_UART_PRINTF example, only the debug output is shown. Therefore, only the stdout is set. In case stdin, stderr or file access is necessary, this can be set in the Manage Run-Time Environment window as well.
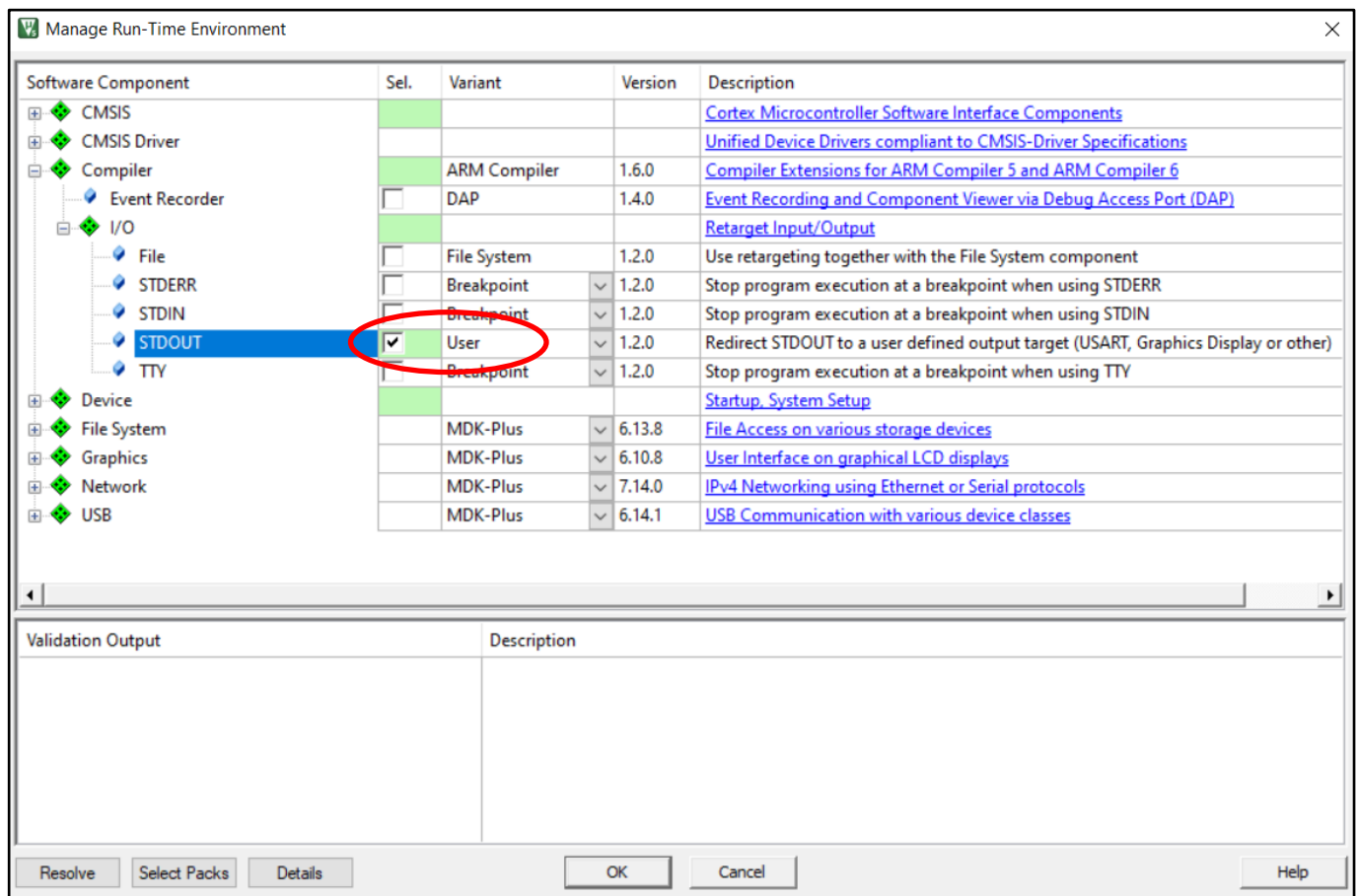
Figure 5 RTE settings for stdout

3.2 Configuration

In order to see the configured output pin for the LED, start the tool Config Wizard can be started. It is available within the Keil IDE through a shortcut in the Tools menu.

The Config Wizard opens and shows an overall status of the current pin configuration. In Figure 6, the pin P1.1 is configured as the debug output.
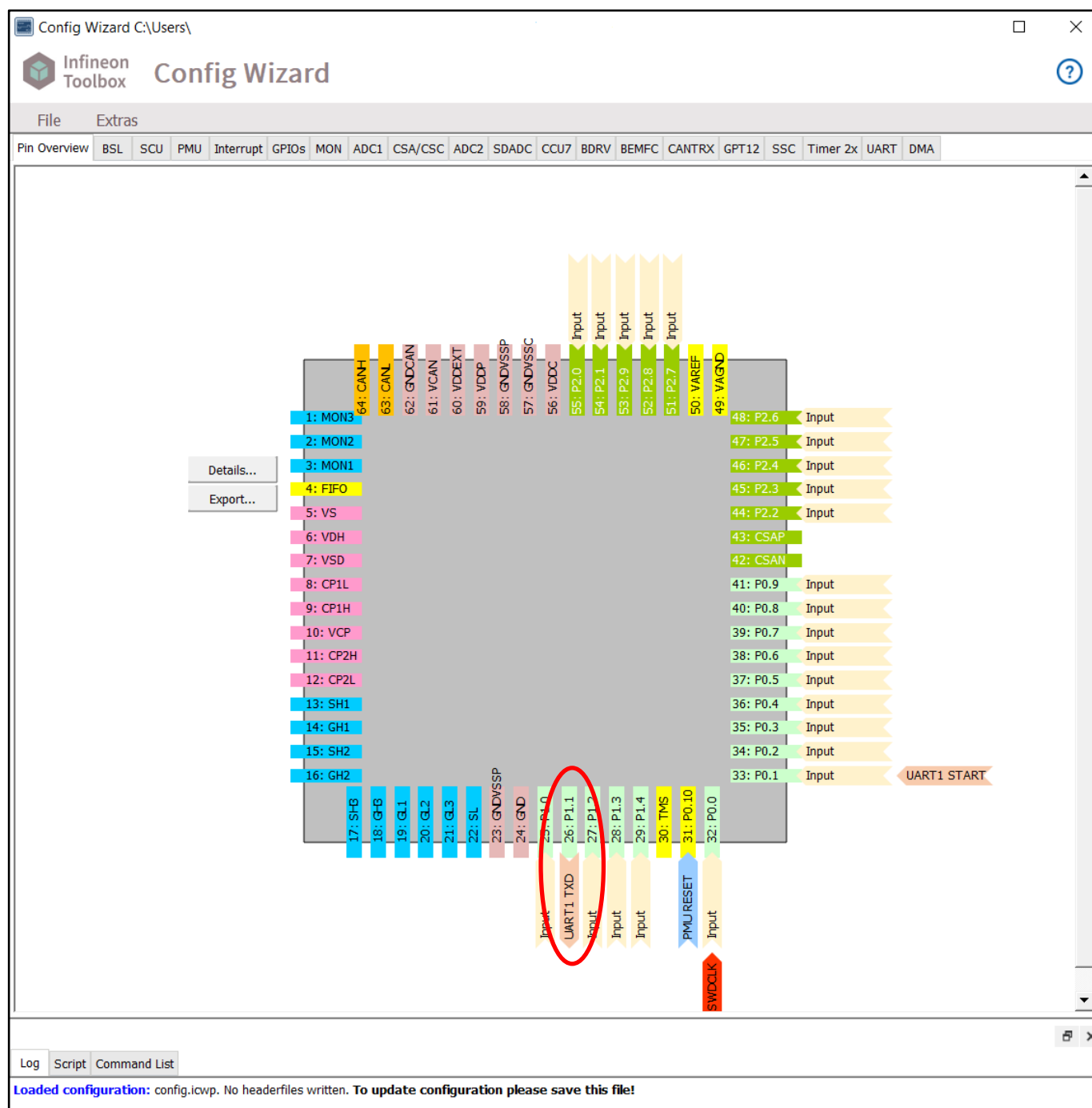
Figure 6 Config Wizard pin overview

In order to configure the UART module for the TLE9893_2QKW62S_UART_PRINTF example, select the UART tab.

Enable the UART1 module. Next, select the 8-bit UART mode with variable baudrate. The baudrate is set to 115200 in the blue box Baudrate Generator Settings. This is one of the common speed settings for the UART. In the pink box Transmission Settings, select the pin P1.1. See Figure 7 Config Wizard, module UART for more details.
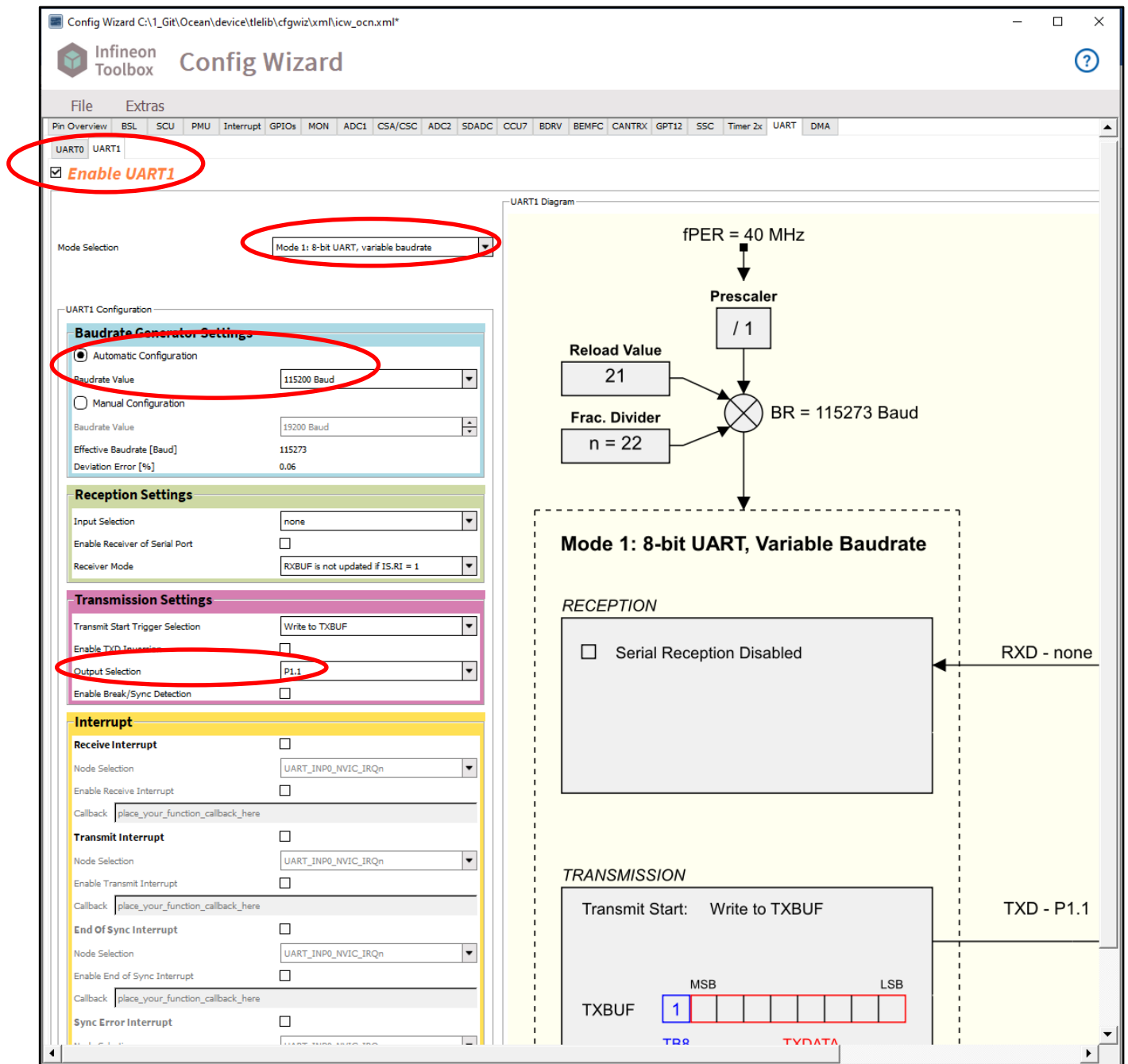
Figure 7 Config Wizard, module UART1

Finally, save your configuration to take these changes into account (File -> Save).

3.3   Sample code

Figure 8 shows the application code of the TLE9893_2QKW62S_UART_PRINTF example.

It executes the printf call (included by stdio.h) outside the endless loop during startup (line 87). However, this can also be put in the for-loop to get a continuous output of the string "HelloWorld".

```
83      /* Clear bridge driver status flags */
84      BDRV->STSCLR.reg = 0xFFFFFFFFU;
85
86      /* Print HelloWorld to the stdout */
87      printf("HelloWorld");
88
89      for (;;)
90      {
91        /* Main watchdog service */
92        (void) PMU_serviceFailSafeWatchdog();
93      }
94  }
```

Figure 8 TLE9893_2QKW62S_UART_PRINTF application code

References

See the code examples at **www.infineon.com**

Revision history

Document version	Date of release	Description of changes
1.0	2021-04-28	Initial version
1.1	2022-10-13	Editorial changes

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