# UART-to-UART Bridge Using Low-Memory MSP430™ MCUs

## Introduction

The universal asynchronous receiver transmitter (UART) interface enables serial communication between the MSP430™ microcontroller (MCU) and another device, such as a personal computer (PC), host MCU, or host processor. Both devices must operate at the same baud rate to communicate. Common baud rates range from 1200 baud to 115200 baud but can reach up to 921600 baud. Basically, a higher baud rate means that the data is sent and received faster. Some designs may require connecting two devices with different baud rates. If these baud rates are fixed, a UART-to-UART bridge is needed to translate the baud rates. The MSP430FR2000 MCU can be used as a low-cost UART-to-UART bridge by using its enhanced universal serial communication interface (eUSCI) UART module and its Timer module. To get started, download project files and a code example demonstrating this functionality.

## Implementation

Figure 1 shows the block diagram for the UART-to-UART bridge. The MSP-TS430PW20 target development board was used for connecting the peripherals to the MSP430FR2000 MCU. First, ensure that jumpers JP14 and JP15 are populated (leave JP13 unpopulated), that jumper J16 is set to UART, and that jumpers JP11, JP17, and JP18 are all removed. These jumper settings allow the backchannel UART interface on the MSP-FET programmer tool to simulate the target device, which has the lower baud rate. Using jumper wires, connect J4.14 (P2.0) to JP11.3, and connect J4.13 (P2.1) to JP11.4. To simulate the host device, which has the higher baud rate, use a USB serial adapter that features USB-to-TTL-level conversion and a maximum baud rate greater than 1 Mbaud. Connect the TX signal to J4.16 (P1.6), connect the RX signal to J4.15 (P1.7), and connect the GND signal to J2.2 (GND).

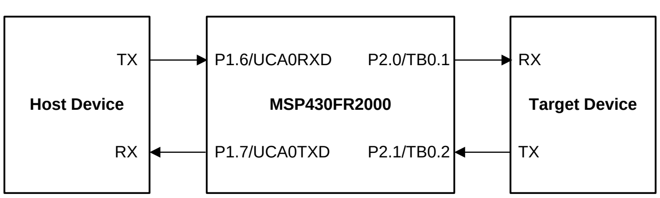


**Figure 1. UART-to-UART Bridge Block Diagram**

Using a PC, open a new serial connection with a terminal program like Tera Term, and connect to the back-channel UART interface on the MSP-FET by selecting the COM port called *MSP Application UART1*. In the first terminal window, change the baud rate to 9600. Next, open another serial connection, and connect to the UART interface on the USB serial adapter by selecting the appropriate COM port. In the second terminal window, change the baud rate to 921600. These two terminal windows will simulate the host and target devices. To demonstrate the functionality of the UART-to-UART bridge, enter a character in either terminal, press *Enter*, and it will be displayed in the other terminal.

In the firmware, the main code initializes the digitally controlled oscillator (DCO), the hardware UART pins and eUSCI UART module, and the software UART pins and Timer module. Then, the central processing unit (CPU) goes to sleep by entering low-power mode 0 (LPM0). When the MCU receives hardware or software UART interrupts, the CPU wakes up, enters active mode, captures the UART data, and then transmits the received data as quickly as possible before going back to sleep.

Figure 2 shows the flowchart for the hardware and software UART code.

When UART data is received by the hardware UART RX pin (P1.6), the UCRXIFG interrupt flag is set, and the data bits are read from the UART RX buffer, UCRXBUF. Next, the Timer module is started and then delayed repeatedly to send the start bit, the eight data bits, and the stop bit using the software UART TX pin (P2.0).

The software UART RX pin (P2.1) is initially configured as general purpose input-output (GPIO) that provides an interrupt on the falling edge of the input signal.

When UART data is received by P2.1, the falling edge of the start bit triggers this interrupt flag. Next, the Timer module is started and then delayed repeatedly to read the eight data bits in the middle of each bit (see Figure 3). The data bits are placed in the UART TX buffer and sent using the hardware UART TX pin (P1.7).

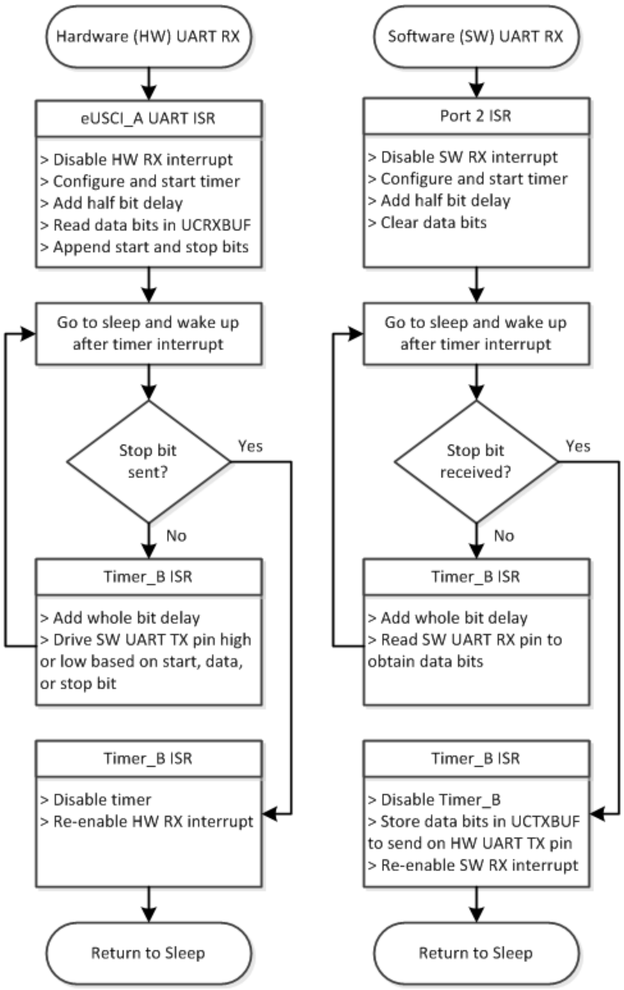


**Figure 2. Hardware and Software UART RX and TX Code Flow**

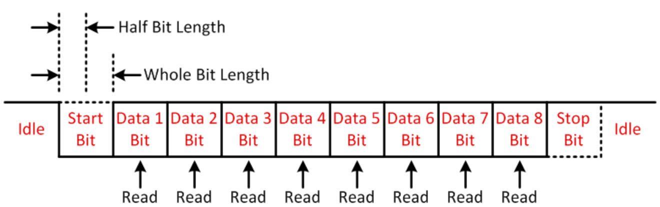


**Figure 3. UART Packet and Timings for Reading Software UART Data**

## Performance

The firmware supports half-duplex UART communication only, which means one direction at a time. It also supports UART packets with 8 data bits, least significant bit (LSB) first, no parity bit, and one stop bit. To maximize the performance of the MSP430FR2000 MCU, always connect the hardware UART interface to the device with the higher baud rate.

When two consecutive UART packets are received by the hardware UART interface, the first packet is processed while the next packet is stored in UCRXBUF. For more than two consecutive packets, these will be skipped because there is only one receive buffer. This limitation does not affect the software UART interface.

To change the baud rate of the hardware UART interface, see the *MSP430FR4xx and MSP430FR2xx Family User's Guide* for the proper configuration. To change the baud rate of the software UART interface, change the *WHOLE_BIT* definition in the code, which equals the subsystem master clock (SMCLK) divided by the baud rate. The *HALF_BIT* definition is just half this value. Figure 3 shows these bit length delays. In this example, SMCLK operates at 16 MHz. After changing these definitions, rebuild the code. Table 1 lists the maximum baud rates supported by both interfaces in this example.

**Table 1. Maximum Baud Rates**

| UART Interface | Maximum Baud Rate |
|---|---|
| Hardware | 921600 |
| Software | 38400 |

Table 2 lists the delay between receiving and sending the UART packets. To reduce code size, the same Timer initialization function is used by the software UART receive and transmit code, which delays by half bit length. Lower baud rates increase the delay.

**Table 2. Delay Between UART Packets**

| UART Packet Flow | RX Baud Rate | TX Baud Rate | Delay (µs) | Average Delay (µs) |
|---|---|---|---|---|
| HW RX to SW TX | 921600 | 38400 | 21 | 22 |
| SW RX to HW TX | 38400 | 921600 | 23 | |
| HW RX to SW TX | 921600 | 9600 | 60 | 58.5 |
| SW RX to HW TX | 9600 | 921600 | 57 | |

LPM0 reduces power consumption while the MCU is not receiving or transmitting data. Other LPMs may achieve lower power consumption, but they may require an external crystal oscillator and may limit the maximum baud rate due to increased wake-up times.

To implement more advanced features such as adding support for flow control, multi-byte receive buffers, odd or even parity bits, multiple stop bits, and CRC error detection, it may be necessary to upgrade to the 1KB MSP430FR2100 MCU.

### Device Recommendations

The device used in this example is part of the MSP430 Value Line Sensing portfolio of low-cost MCUs, designed for sensing and measurement applications. This example can be used with the devices shown in Table 3 with minimal code changes. For more information on the entire Value Line Sensing MCU portfolio, visit www.ti.com/MSP430ValueLine.

**Table 3. Device Recommendations**

| Part Number | Key Features |
| --- | --- |
| MSP430FR2000 | 0.5KB FRAM, 0.5KB RAM, eComp |
| MSP430FR2100 | 1KB FRAM, 0.5KB RAM, 10-bit ADC, eComp |
| MSP430FR2110 | 2KB FRAM, 1KB of RAM, 10-bit ADC, eComp |
| MSP430FR2111 | 3.75KB FRAM, 1KB RAM, 10-bit ADC, eComp |

MSP430 is a trademark of Texas Instruments.

## IMPORTANT NOTICE FOR TI DESIGN INFORMATION AND RESOURCES

Texas Instruments Incorporated ('TI") technical, application or other design advice, services or information, including, but not limited to, reference designs and materials relating to evaluation modules, (collectively, "TI Resources") are intended to assist designers who are developing applications that incorporate TI products; by downloading, accessing or using any particular TI Resource in any way, you (individually or, if you are acting on behalf of a company, your company) agree to use it solely for this purpose and subject to the terms of this Notice.

TI's provision of TI Resources does not expand or otherwise alter TI's applicable published warranties or warranty disclaimers for TI products, and no additional obligations or liabilities arise from TI providing such TI Resources. TI reserves the right to make corrections, enhancements, improvements and other changes to its TI Resources.

You understand and agree that you remain responsible for using your independent analysis, evaluation and judgment in designing your applications and that you have full and exclusive responsibility to assure the safety of your applications and compliance of your applications (and of all TI products used in or for your applications) with all applicable regulations, laws and other applicable requirements. You represent that, with respect to your applications, you have all the necessary expertise to create and implement safeguards that (1) anticipate dangerous consequences of failures, (2) monitor failures and their consequences, and (3) lessen the likelihood of failures that might cause harm and take appropriate actions. You agree that prior to using or distributing any applications that include TI products, you will thoroughly test such applications and the functionality of such TI products as used in such applications. TI has not conducted any testing other than that specifically described in the published documentation for a particular TI Resource.

You are authorized to use, copy and modify any individual TI Resource only in connection with the development of applications that include the TI product(s) identified in such TI Resource. NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE TO ANY OTHER TI INTELLECTUAL PROPERTY RIGHT, AND NO LICENSE TO ANY TECHNOLOGY OR INTELLECTUAL PROPERTY RIGHT OF TI OR ANY THIRD PARTY IS GRANTED HEREIN, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information regarding or referencing third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of TI Resources may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

TI RESOURCES ARE PROVIDED "AS IS" AND WITH ALL FAULTS. TI DISCLAIMS ALL OTHER WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED, REGARDING TI RESOURCES OR USE THEREOF, INCLUDING BUT NOT LIMITED TO ACCURACY OR COMPLETENESS, TITLE, ANY EPIDEMIC FAILURE WARRANTY AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

TI SHALL NOT BE LIABLE FOR AND SHALL NOT DEFEND OR INDEMNIFY YOU AGAINST ANY CLAIM, INCLUDING BUT NOT LIMITED TO ANY INFRINGEMENT CLAIM THAT RELATES TO OR IS BASED ON ANY COMBINATION OF PRODUCTS EVEN IF DESCRIBED IN TI RESOURCES OR OTHERWISE. IN NO EVENT SHALL TI BE LIABLE FOR ANY ACTUAL, DIRECT, SPECIAL, COLLATERAL, INDIRECT, PUNITIVE, INCIDENTAL, CONSEQUENTIAL OR EXEMPLARY DAMAGES IN CONNECTION WITH OR ARISING OUT OF TI RESOURCES OR USE THEREOF, AND REGARDLESS OF WHETHER TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You agree to fully indemnify TI and its representatives against any damages, costs, losses, and/or liabilities arising out of your non-compliance with the terms and provisions of this Notice.

This Notice applies to TI Resources. Additional terms apply to the use and purchase of certain types of materials, TI products and services. These include; without limitation, TI's standard terms for semiconductor products http://www.ti.com/sc/docs/stdterms.htm), evaluation modules, and samples (http://www.ti.com/sc/docs/sampterms.htm).

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265