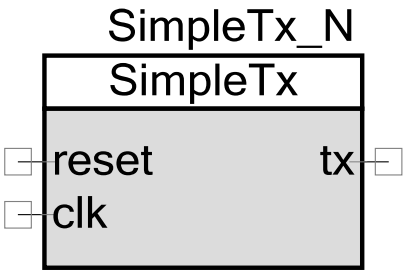
8-Bit Simple UART Tx (SimpleTx)

1.0

Features

- 8-Bit Parallel to Serial Transmitter
- One or Two Stop Bits
- 4 Byte FIFO
- Enable/Disable Modes
- FIFO not Full Status
- Busy Transmit Status



General Description

The SimpleTx is a simple 8-bit UART transmitter used to shift parallel data out serially for PC communication. This component was created to help new users understand out to develop simple components using the shift register in UDB Datapaths.

Input/Output Connections

This section describes the various input and output connections.

clk – Input

Clock input. Set this to same frequency at the desired baud rate.

reset – Input

This signal resets the internal state machine to an idle state.

tx – Output

The tx output is the serial output data. Usually this signal will be connected to an RS-232 transceiver and connected to second computer or PC.

Parameters and Setup

This component does not contain settable parameters in the customizer.

Resources

PSoC UDB resources used.

Resource	Amount Used	Percent PSoC Resource Used
Macrocells	7	3.65%
Unique Product Terms	12	3.13%
Datapaths	1	4.17%
Control Register	1	4.17%
Status Register	1	4.17%

Application Programming Interface

This component does not include APIs for programming but instead provides register definitions so that the user can easily set parameters during program execution time.

Sample Firmware Source Code

The following is a C language example demonstrating the basic functionality of the SimpleTx component. This example assumes the component has been placed in a design with the default name " SimpleTx _1."

Note If you rename your component you must also edit the example code as appropriate to match the component name you specify.

```
#include <project.h>

int main()
{
    uint8 i;
```

```
 /* Enable the SimpleTx with two Stop Bits */
SimpleTx_1_Control_Reg = (SimpleTx_1_CNTL_ENABLE | SimpleTx_1_CNTL_TWO_STOPBIT);

 for(;;)
 {
    /* Send the numbers 0 thru 10 out the serial port. */
    for(i=0; i < 11; i++)
    {
       /* Wait for there to be room in the FIFO */
       while(!(SimpleTx_1_Status_Reg & SimpleTx_1_STATUS_FIFONOTFULL));
           /*Write data to the FIFO*/
       SimpleTx_1_Data_Reg = i;
    }

     /* Wait for all data to be sent, then delay 10 mSec*/
     while(SimpleTx_1_Status_Reg & SimpleTx_1_STATUS_BUSY);
     CyDelay(10);
 }
}
```

Registers

The functions provided support most of the common runtime functions required for most applications. The following register references provide brief descriptions for the advanced user.

SimpleTx_Control_Reg

Bits	7	6	5	4	3	2	1	0
Value	Not Used						Stop Bits	Enable

- Enable: Enable Transmissions (0 = Disable, 1 = Enable).
- Stop Bits: Selects either one or two stop bits.
  - 0: One Stop Bit
  - 1: Two Stop Bits

SimpleTx_Status_Reg

Bits	7	6	5	4	3	2	1	0
Value	Not Used						FIFO not full	Transmitting

- Transmitting: This bit is set if there is data in the FIFO or data in the process of bring shifted out.

- FIFO not full: This status bit tell the programmer if there is room for any more data in the FIFO. If clear, data should not be written to the data register.

SimpleTx _Data_Reg

Bits	7	6	5	4	3	2	1	0
Value	Data[7:0]							

- Data[7:0]: 8-Bit data to be sent.

Component Changes

This section lists the major changes in the component from the previous version.

Version	Description of Changes	Reason for Changes / Impact
1.00		Initial document

© Cypress Semiconductor Corporation, 2014. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

PSoC® is a registered trademark, and PSoC Creator™ and Programmable System-on-Chip™ are trademarks of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein are property of the respective corporations.

Any Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.