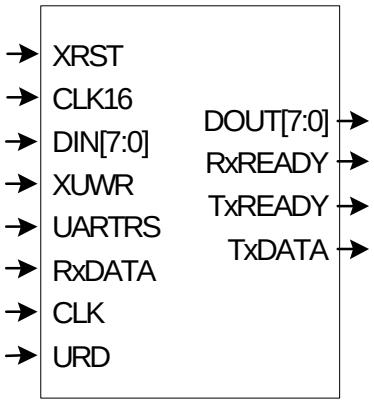
Function Description

The C_UART megafunction is a generic universal asynchronous receiver/transmitter (UART) and can be used to implement a peripheral data communication device. The designer can program the megafunction with an 8-bit CPU.

Features

- 8 bit characters
- TxC / RxC (16 times the desired output baud rate)
- 1 start bit / 1 stop bit
- Polling and interrupt modes
- Flexibility for adding other features
- The C_UART was developed in VHDL

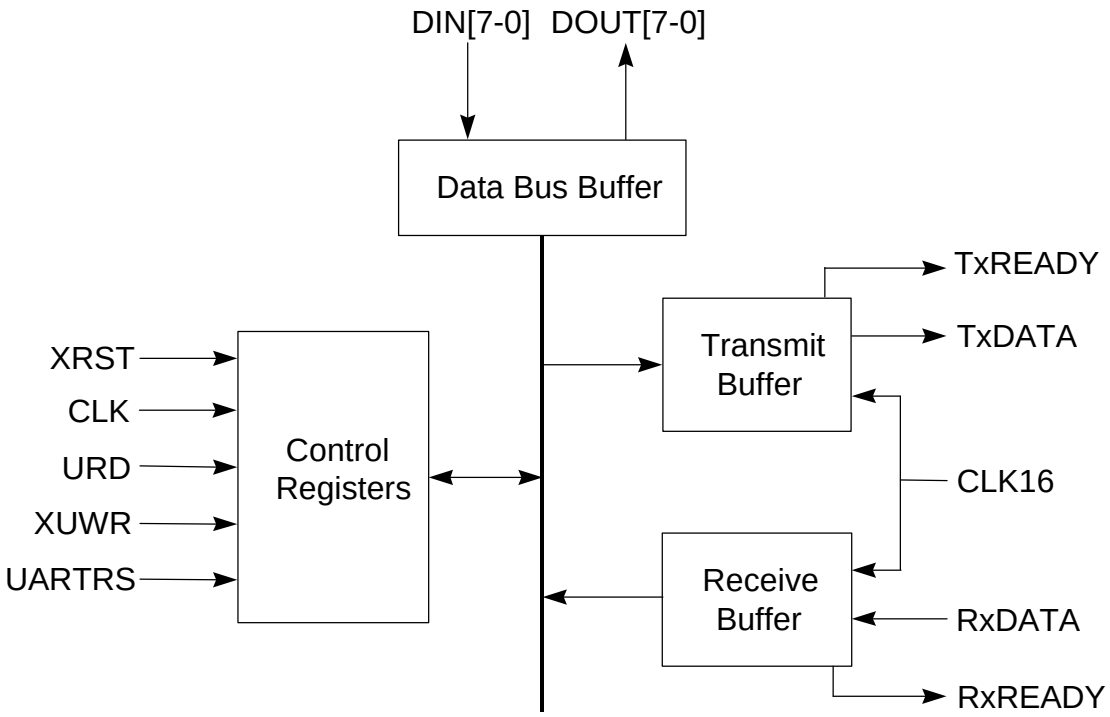
Symbol



Pin Description

Name	Type	Polarity	Description
CLK	In	Rising	System Clock
XRST	In	Low	Master Reset
CLK16	In	Rising	Transmitter/Receiver Master Clock
XUWR	In	Rising	Decoded UART Write Control for Transmitter Register
URD	In	Falling	Decoded UART Read
UARTRS	In	-	Register Select (0 = Status; 1 = RX_Register)
DIN[7:0]	In	-	Input Data Bus
RxDATA	In	-	Receiver Data
DOUT[7:0]	Out	-	Output Data Bus
TxDATA	Out	-	Transmitter Data
TxREADY	Out	Low	Transmitter Empty (ready to accept new character)
RxREADY	Out	Low	Receiver Ready (character ready to be transferred to CPU)

Block Diagram



Signal Definitions

UART INTERFACE SIGNALS FOR CPU

The UART interfaces to a generic CPU via two 8-bit data buses, a register select line, two interrupt request lines, two decoded R/W lines, and a Master Reset line. These signals permit the CPU to take complete control of the UART.

UART Data Buses (Din[7:0],Dout [7:0])

Data lines allow data transfers between CPU and UART (DIN) and vice-versa (DOUT).

XRST

The Master Reset (active low) resets all registers.

XUWR

With data present on the DIN[7:0] pins, a rising edge of XUWR latches the data in the Transmit Register and starts the trasmit sequence.

URD

After the input data stream is received, this signal (active-low) tells the receiver that the CPU has read the data, and also re-initializes the receiver section setting of RxREADY.

UARTRS (Register Select)

This pin controls the DOUT[7:0] lines. A high level is used to select the receive buffer, and a low level to select status register.

CLK (System clock)

A rising edge on this clock latches the contents of the RxREADY and TxREADY lines in the Status Register. It should be synchronous with the CPU clock.

RxREADY(Rx Buffer Status)

This line could be used as a receive interrupt (active-low). A low level on this line indicates that the data has been received and the Status Register bit 0 is low. This interrupt can be cleared by reading data or by resetting the UART.

TxREADY(Tx Buffer Status)

This line could be used as a transmit interrupt (active-low). A low level on this line indicates that the data has been transmitted, Tx buffer is empty and the Status Register bit 1 is low. This interrupt can be cleared by writing data or by resetting the UART.

CLOCK INPUT (CLK16)

A single clock input is provided for transmitting and receiving data. This clock frequency should be 16 times the data rate.

SERIAL INPUT/OUTPUT LINES

RxDATA(Receive data)

The Receive Data line is an input through which data is received in serial format. Synchronization with a clock for detection of data is accomplished internally.

TxDATA(Trasmit data)

The Transmit Data output line transfers serial data to other peripherals.

Modeling Notes

Data and Status Registers

R/W RX_TX_REG UART Data Register

7	6	5	4	3	2	1	0
d	d	d	d	d	d	d	d

The Data Register is used for reading data from the receiver or writing data to the transmitter. This register is cleared during Reset. The UART Data Register is located in the Control Registers Block.

R UARTRD UART STATUS Register

7	6	5	4	3	2	1	0
0	0	0	0	0	0	n	n

The Status Register maintains information about current operational status of the UART. During Reset bit 0 of the register is set, and bit 1 is cleared. The UART Status Register is located in the Control Registers Block.

bit 0 RxREADY	0	RxDATA RECEIVED (rx_reg full)
	1	Rx BUFFER EMPTY
bit 1 TxREADY	0	Tx BUFFER EMPTY
	1	Tx BUFFER FULL

Transmit Sequence

Figures 1 and 2 show the transmitting sequence of the C_UART. For the purposes of these figures, TxDATA and RxDATA are connected to an external signal called IN_TxDATA.

After reset and with TxREADY going 'low' signaling that the transmit buffer is empty a new sequence can start by bringing XUWR low. XUWR in this example is driven from external signals CS_UART + WRITE (both active low).

The 'AA' value on DIN is stored in the transmit buffer on the rising edge of XUWR.

The serial data is transmitted (TxDATA) with a baud-rate equal to CLK16/16 and fed back to the receiver section of the UART (RxDATA). A low level on RxREADY signals when the data has been received, and it appears on DOUT.

Note that UARTRS is 'high' which selects the Receive Data Register to appear on DOUT.

A low level on TxREADY completes the transmitting sequence, and indicates that the transmit buffer is empty and ready for a new transmission to start.

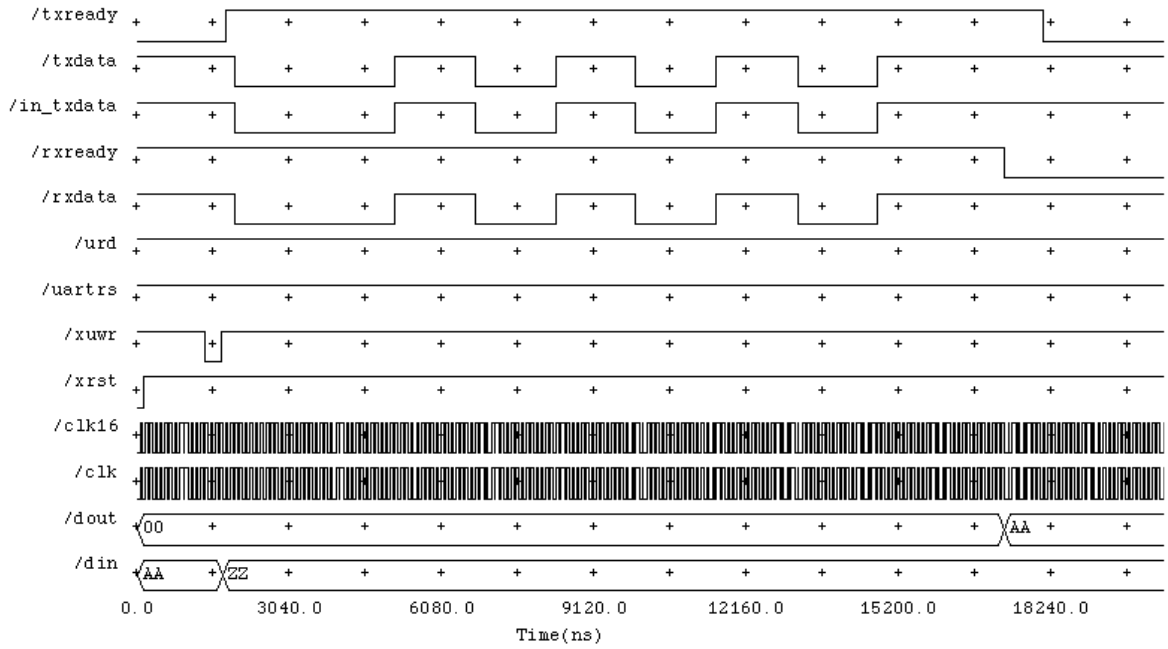


Figure 1 - Transmit/Receive Sequence

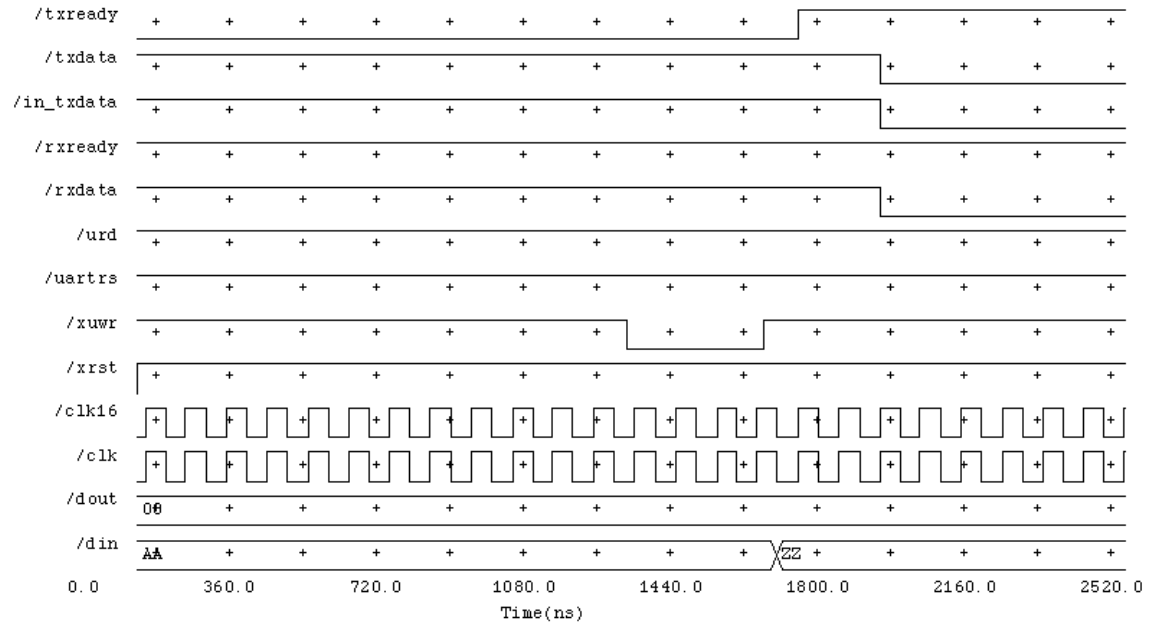


Figure 2 - Transmit/Receive Sequence (Zoomed in)

Receive Sequence

Figure 3 shows the receiving sequence of the C_UART. For the purposes of these figures, TxDATA and RxDATA are connected to an external signal called IN_TxDATA.

The rising edge of CLK continuously stores the value of TxREADY and RxREADY signals in bit 0 and bit1 respectively of the Status Register. While UARTRS is 'low' DOUT will show the contents of the Status Register.

With the falling edge of URD, RxREADY goes 'high' indicating that the receiver is ready. Note that DOUT goes to '01' once RxREADY goes 'high'. URD in this example is generated from external signals CS_UART + READ (both active low).

Note: Since there is no overrun flag, the user must insure that the Receive Buffer Register is read before a new data stream arrives. The received data will be retained in the Receive Buffer Register until the new data arrives.

A rising edge of XUWR latches DIN in theTransmit Register and a new transmission starts (data latched is "55"). Note that DOUT is now "03" since TxREADY went 'high'.

UARTRS going 'high' selects again the Receive Data Buffer to appear on DOUT (AA) which was the previous data received. Once RxReady goes 'low' the new received data (55) is loaded into the Receive Data and the receiving sequence is complete.

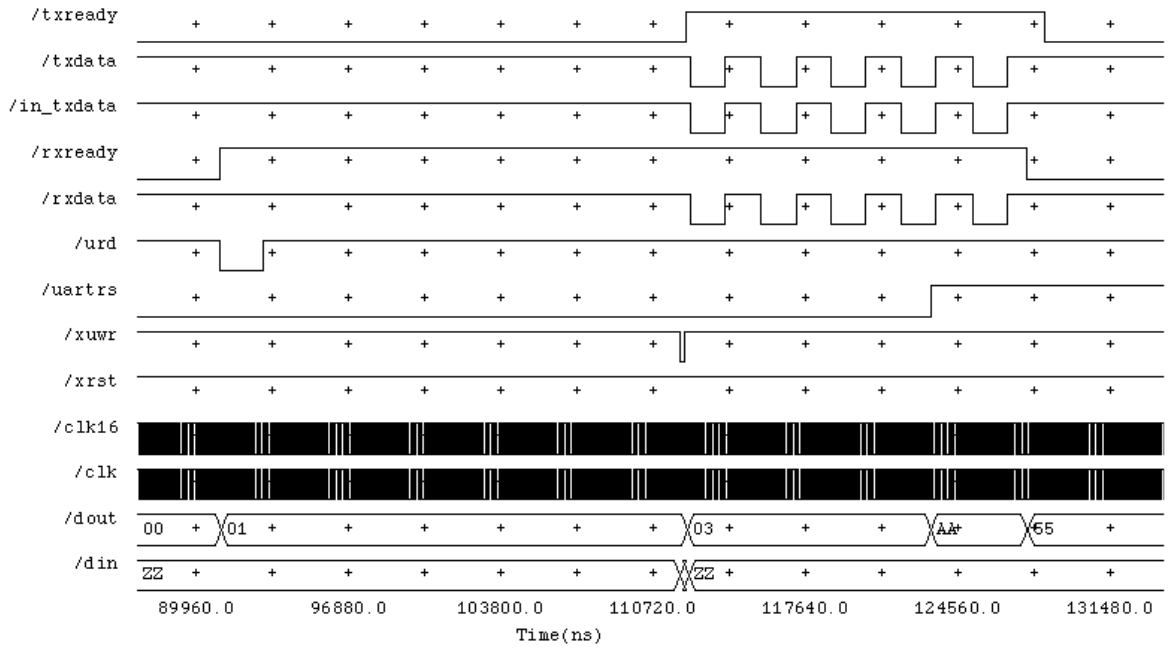


Figure 3 - Receive Sequence

Applications

The C_UART megafunction is used in serial data communications applications.

Megafunction Modifications

The C_UART megafunction can have the following functions customized:

- Character length
- Parity
- Start/Stop bits

Please contact CAST directly for any required modifications.

Device Utilization & Performance

Target Device	Speed Grade	Utilization		Performance	Availability
		LCs	EABs	F_{max}	
EPM7096	-7	68	-	62 MHz	**Now**
EPM9320	-15	68	-	40 MHz	**Now**
EPF6016	-2	112	-	76 MHz	**Now**
EPF8282	-2	96	-	67 MHz	**Now**
EPF10K10	-3	87	-	84 MHz	**Now**
EPF10K10A	-1	87	-	172 MHz	**Now**

Deliverables

Encrypted Licenses

- Post-synthesis AHDL
- Assignment & Configuration
- Symbol file
- Include file
- Graphic Design file of test circuit
- Vectors for testing the functionality of the megafunction

VHDL Source Licenses

- VHDL RTL source code
- Testbench
- Example testbench wrapper for post-route simulation
- Vectors for testbench
- Expected results for testbench
- Simulation script
- Synthesis script

CAST, Inc.

24 White Birch Drive
Pomona, New York 10907 USA
Phone: +1 914-354-4945 Fax: +1 914-354-0325
E-Mail: info@cast-inc.com URL : www.cast-inc.com

The C_UART megafunction is licensed from Moxsyn S.r.l.