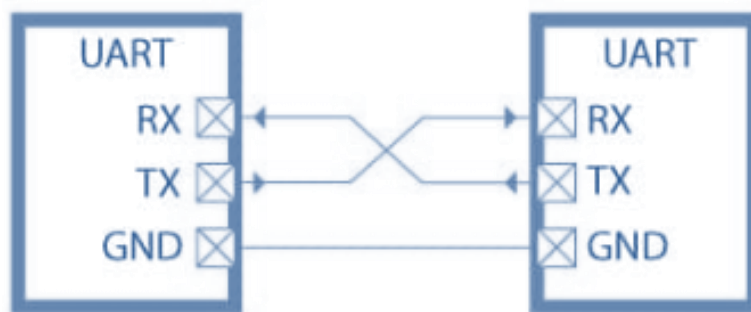


UART Specification

Introduction

- There are many serial communication protocols as I2C, UART and SPI.
- A Universal Asynchronous Receiver/Transmitter (UART) is a block of circuitry responsible for implementing serial communication.

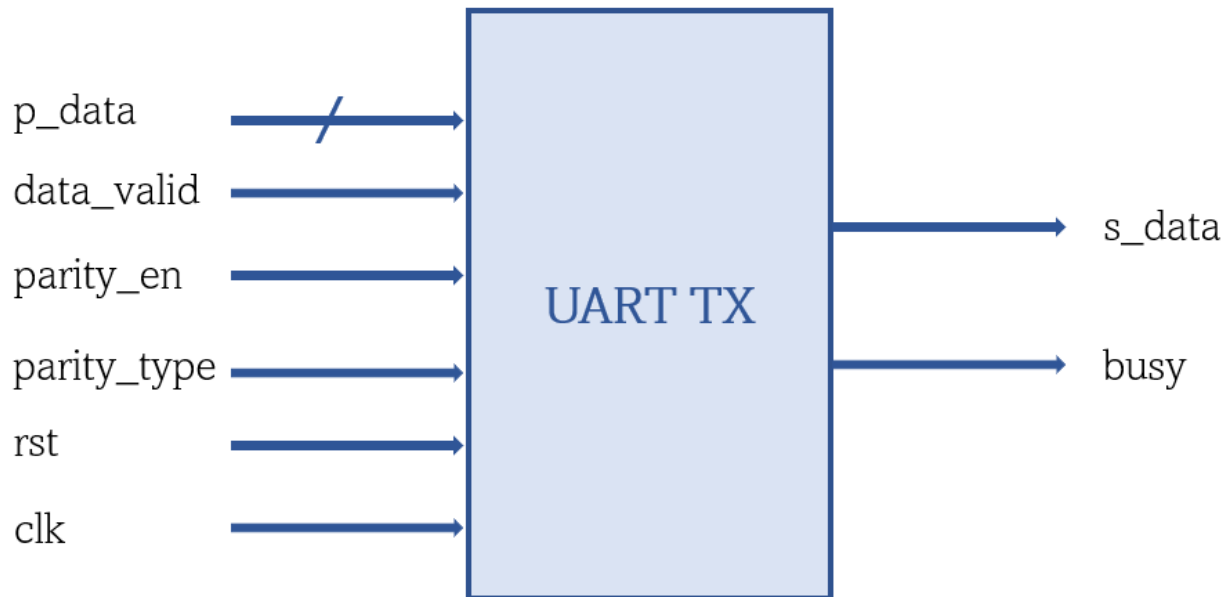
UART is Full Duplex protocol (data transmission in both directions simultaneously).



- Transmitting UART converts parallel data from the master device (e.g., CPU) into serial form and transmit in serial to receiving UART.
- Receiving UART will then convert the serial data back into parallel data for the receiving device.

UART Transmitter

Block Diagram



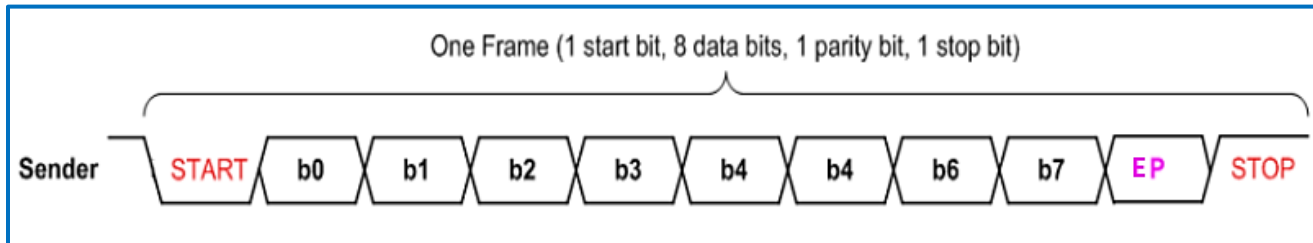
Specifications

1. UART TX receives the new data on **p_data** Bus only when **data_valid** Signal is **high**.
2. Registers are cleared using asynchronous active low reset.
3. **data_valid** is high for only 1 clock cycle.
4. **busy** signal is **high** as long as UART_TX is transmitting the frame, otherwise **low**.
5. UART_TX couldn't accept any data on **p_data** during UART_TX processing, however **data_valid** get high.
6. **s_data** is high in the **IDLE** case (No transmission).
7. **parity_en** (Configuration)
 - 0: To disable frame parity bit
 - 1: To enable frame parity bit
8. **parity_type** (Configuration)
 - 0: Even parity bit
 - 1: Odd parity bit
9. 200 MHz clock frequency.

All Expected Output Frames

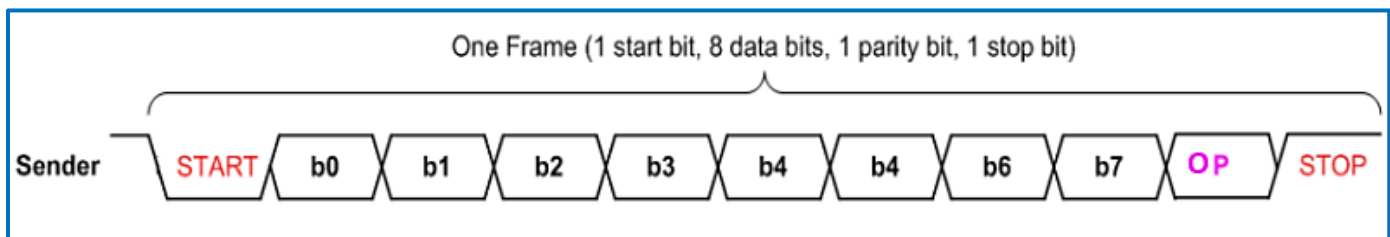
1. Data Frame (in case of Parity is enabled & Parity Type is even)

- One start bit (1'b0)
- Data (LSB first or MSB, 8 bits)
- Even Parity bit
- One stop bit



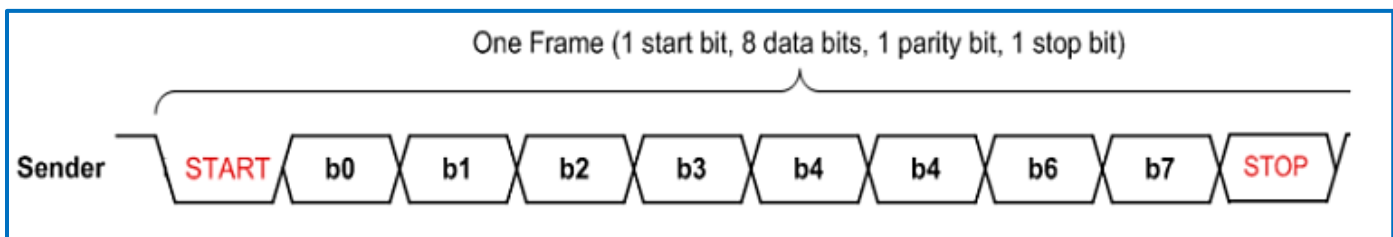
2. Data Frame (in case of Parity is enabled & Parity Type is odd)

- One start bit (1'b0)
- Data (LSB first or MSB, 8 bits)
- Odd Parity bit
- One stop bit



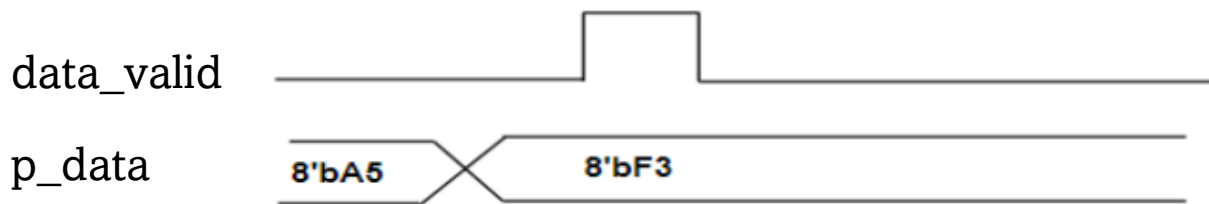
3. Data Frame (in case of Parity is not Enabled)

- One start bit (1'b0)
- Data (LSB first or MSB, 8 bits)
- One stop bit

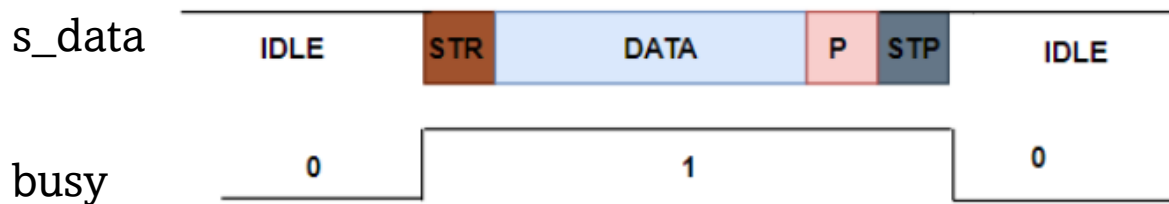


Waveforms

1. Expected Input

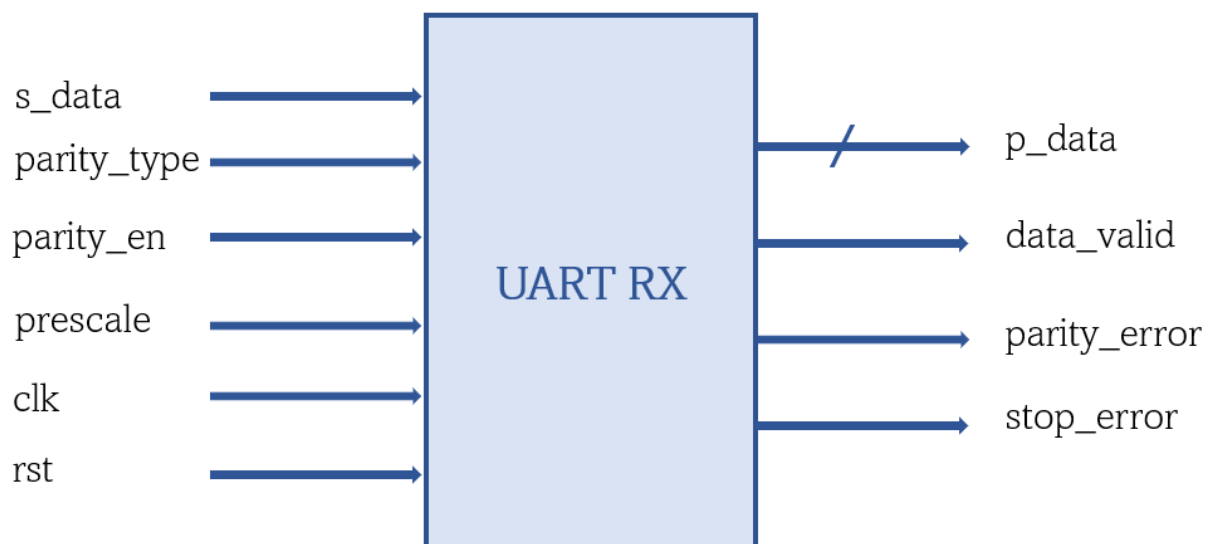


2. Expected Output



UART Receiver

Block Diagram

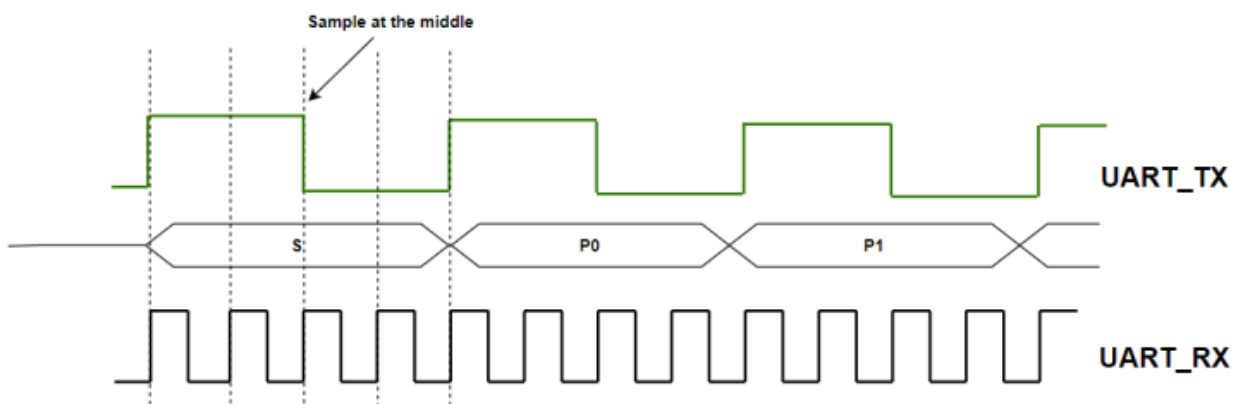


Specifications

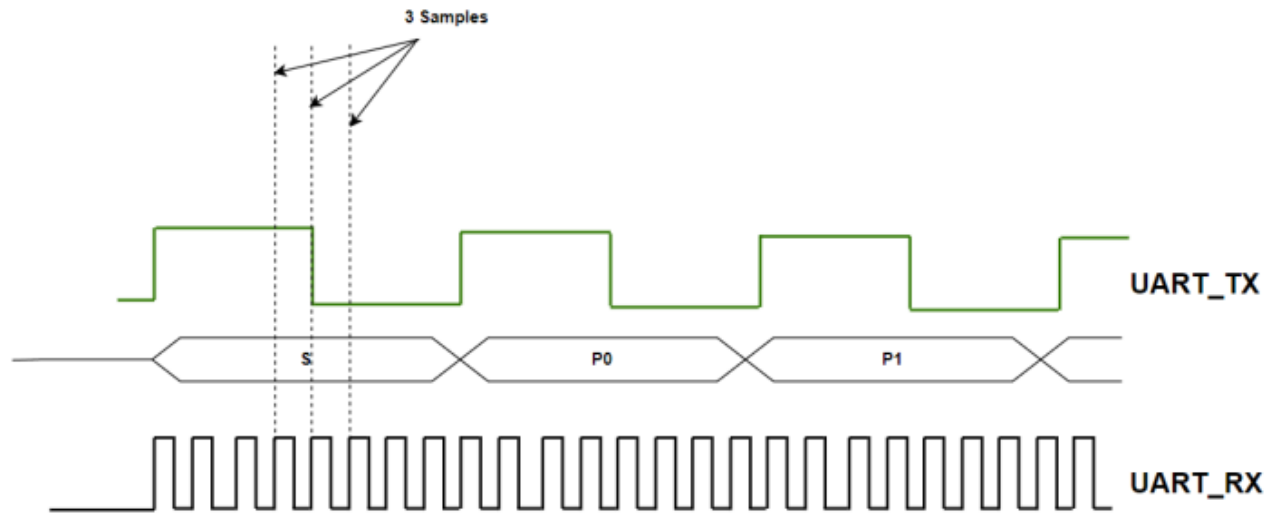
1. UART TX receives a UART frame on `s_data`.
2. UART_RX supports **oversampling** by 8.
3. `s_data` is high in the **IDLE** case (No transmission).
4. `parity_error` signal is **high** when the calculated parity bit does not equal the received frame parity bit as this means that the frame is corrupted.
5. `stop_error` signal is **high** when the received stop bit does not equal 1 as this means that the frame is corrupted.
6. DATA is extracted from the received frame and then sent through `p_data` bus associated with `data_valid` signal **only** after checking that the frame is received correctly and not corrupted. (`parity_error=0` && `stop_error=0`).
7. UART_RX can accept consequent frames.
8. Registers are cleared using asynchronous active low reset.
9. `parity_en` (**Configuration**)
 - 0: To disable frame parity bit
 - 1: To enable frame parity bit
10. `parity_type` (**Configuration**)
 - 0: Even parity bit
 - 1: Odd parity bit

Oversampling

1. Oversampling by 4: This means that the clock speed of UART_RX is 4 times the speed of UART_TX.



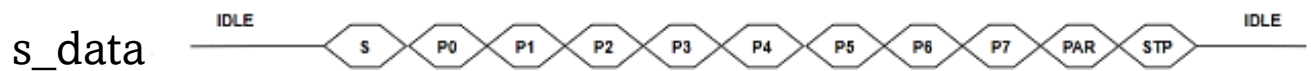
2. Oversampling by 8: This means that the clock speed of UART_RX is 8 times the speed of UART_TX.



Waveforms

1. Expected Input

In case of one frame



In case of multiple frames



2. Expected Output

