UART Module

Baragan Andrei Iustin

January 11, 2022

Contents

1 Introduction **1**

2 Objectives **2**

3 Bibliography study **2**
3.1 Start Bit . . . 2
3.2 Data Frame . . . 3
3.3 Stop Bit . . . 3
3.4 ADVANTAGES AND DISADVANTAGES OF UARTS . . . 3

4 Design and Implementation **3**
4.1 Transmitter . . . 3
4.2 Receiver . . . 4

5 Testing **5**

6 Testing **5**

7 Testing **6**

8 Conclusion **6**

1 Introduction

A universal asynchronous receiver-transmitter (UART) is a computer hardware device for asynchronous serial communication in which the data format and transmission speeds are configurable. It sends data bits one by one, from the least significant to the most significant, framed by start and stop bits so that precise timing is handled by the communication channel. The electric signaling levels are handled by a driver circuit external to the UART. Two common signal levels are RS-232, a 12-volt system, and RS-485, a 5-volt system. Early teletypewriters used current loops.

It was one of the earliest computer communication devices, used to attach teletypewriters for an operator console. It was also an early hardware system for the Internet.

A UART is usually an individual (or part of an) integrated circuit (IC) used for serial communications over a computer or peripheral device serial port. One or more UART peripherals are commonly integrated in microcontroller chips. Specialised UARTs are used for automobiles, smart cards and SIMs.

A related device, the universal synchronous and asynchronous receiver-transmitter (USART) also supports synchronous operation.

Transmitting and receiving UARTs must be set for the same bit speed, character length, parity, and stop bits for proper operation. The receiving UART may detect some mismatched settings and set a "framing error" flag bit for the host system; in exceptional cases, the receiving UART will produce an erratic stream of mutilated characters and transfer them to the host system.

Typical serial ports used with personal computers connected to modems use eight data bits, no parity, and one stop bit; for this configuration, the number of ASCII characters per second equals the bit rate divided by 10.

Some very low-cost home computers or embedded systems dispense with a UART and use the CPU to sample the state of an input port or directly manipulate an output port for data transmission. While very CPU-intensive (since the CPU timing is critical), the UART chip can thus be omitted, saving money and space. The technique is known as bit-banging. [3, Wikipedia Description of UART]

2 Objectives

The objective of this project is to prototype a functional UART Receiver Transmitter using vhdl language.The connection between the receiver and the transmitter will be made by a queue that will take the data from the receiver and will transmit them to the transmitter.

The UART module will be implemented on a basys3 board and a arduino mega board will be used to test it.

3 Bibliography study

In UART communication, two UARTs communicate directly with each other. The transmitting UART converts parallel data from a controlling device like a CPU into serial form, transmits it in serial to the receiving UART, which then converts the serial data back into parallel data for the receiving device. Only two wires are needed to transmit data between two UARTs. Data flows from the Tx pin of the transmitting UART to the Rx pin of the receiving UART:

[2, UARTinfo]

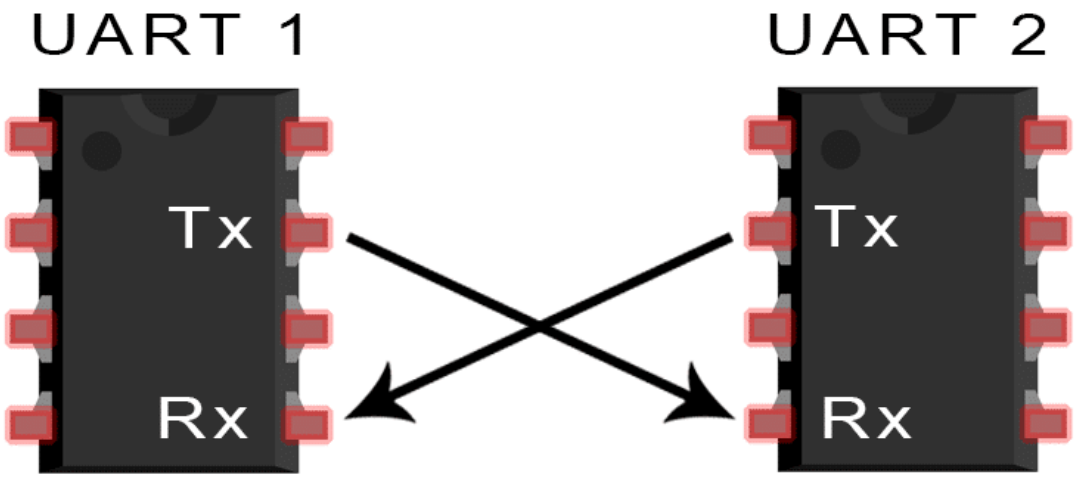

Figure 1: UART Connection

3.1 Start Bit

The UART data transmission line is normally held at a high voltage level when it's not transmitting data. To start the transfer of data, the transmitting UART pulls the transmission line from high to low for one clock cycle. When the receiving UART detects the high to low voltage transition, it begins reading the bits in the data frame at the frequency of the baud rate.

[2]

3.2 Data Frame

The data frame contains the actual data being transferred. It can be 5 bits up to 8 bits long if a parity bit is used. If no parity bit is used, the data frame can be 9 bits long. In most cases, the data is sent with the least significant bit first.[2]

3.3 Stop Bit

To signal the end of the data packet, the sending UART drives the data transmission line from a low voltage to a high voltage for at least two bit durations.[2]

3.4 ADVANTAGES AND DISADVANTAGES OF UARTS

No communication protocol is perfect, but UARTs are pretty good at what they do. Here are some pros and cons to help you decide whether or not they fit the needs of your project:

Advantages:

- Only uses two wires
- No clock signal is necessary
- The structure of the data packet can be changed as long as both sides are set up for it
- Well documented and widely used method

Disadvantages:

- The size of the data frame is limited to a maximum of 9 bits
- Doesn't support multiple slave or multiple master systems
- The baud rates of each UART must be within 10% of each other

[1]

References

[1] *Analogue-Dialogue : UART*. URL: https://www.analog.com/en/analog-dialogue/articles/uart-a-hardware-communication-protocol.html.

[2] *Circuit : UART*. URL: https://www.circuitbasics.com/basics-uart-communication/.

[3] *Wikipedia/UART*. URL: https://en.wikipedia.org/wiki/Universal_asynchronous_receiver-transmitter.

4 Design and Implementation

The Design of Transmitter and Receiver was inspired from this code but it differs a lot.

4.1 Transmitter

This is basically the diagram of the transmitter:

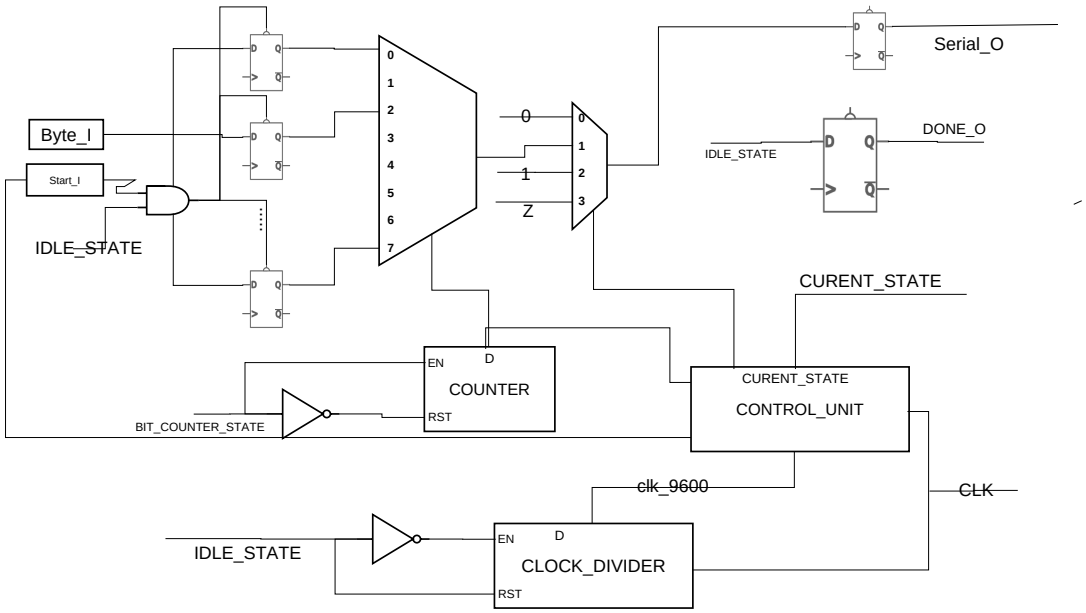

Figure 2: Transmitter Diagram

The tranmitter "motor" is that Control Unit that has 4 states:

1. IDLE STATE when the transmitter wait to receive a new byte to transmit.In this state the Counter and the Clock Divider are off and the output is high value.
2. START BIT STATE .In this state the output is low , th clock divider is started and the control unit wait for clk_9600 signal to move to the next state In this states the clock Divider starts.
3. BIT TRANSMITTER STATE .In this state the counter starts and the output has the value of the bistable indicated by counter.The counter use clk_9600 signal clock and when all values were transmitted ,the control unit switches to the next state.
4. STOP BIT STATE.In this state the output is high value , the counter is off .The control unit wait for clk_9600 signal to switch back to IDLE STATE

4.2 Receiver

The receiver is designed in a very similar way to the transmitter, with small differences. As with the transmitter, the receiver motor is the control unitand it has 4 states:

1. IDLE STATE.In this state the receiver is waiting for the first low signal to switch to Start Bit State.The counter and clock divider are also stopped.
2. START BIT STATE.This state lasts only half of clk_9600 signal clock period , and because of that all next States are half a clock ahead, so all reading will be done at half the transmitter clock.If during the state the signal becomes high the receiver switch back to IDLE STATE.In this states the clock Divider starts.

3. BIT TRANSMITTER STATE .In this state the counter starts and every input received is stored where the counter indicate.After received of 8 inputs , the control unit it switch to STOP BIT STATE.

4. STOP BIT STATE.This states verify the stop bit (HIGH VALUE).If it's not high then the Receiver it returns to IDLE STATE , and if it is , it set the done flag and retun to IDLE STATE.

5 Testing

Lower are the test for UART MODULE . In these tests some data are transmitted by the transmitter and the receiver receive the information with a small delay.

The bue signals are the principal one .It show the data transmitted and the data received in time .

The red signal are specific receiver signals like the state , or the bit index (generated by the counter), the clk_9600 signal generated by clock divider(one at 4 normal clk).

The red signal are specific transmitter signals like the state , or the bit index (generated by the counter), the clk_9600 signal generated by clock divider(one at 4 normal clk).

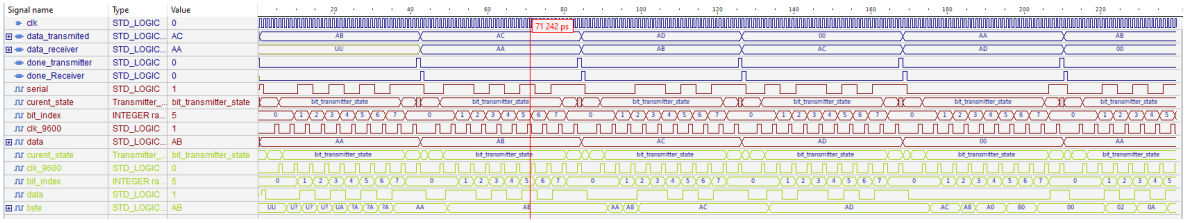

Figure 3: UART MODULE Test

6 Testing

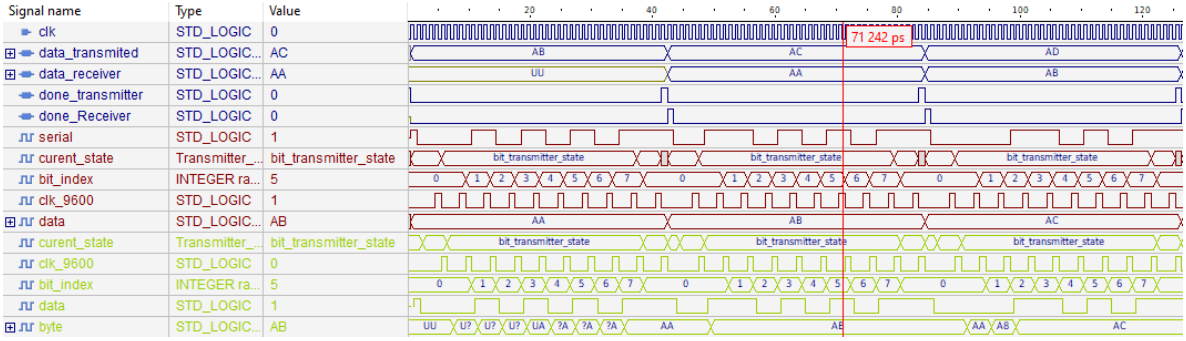

Figure 4: UART MODULE Test subsection1

7 Testing

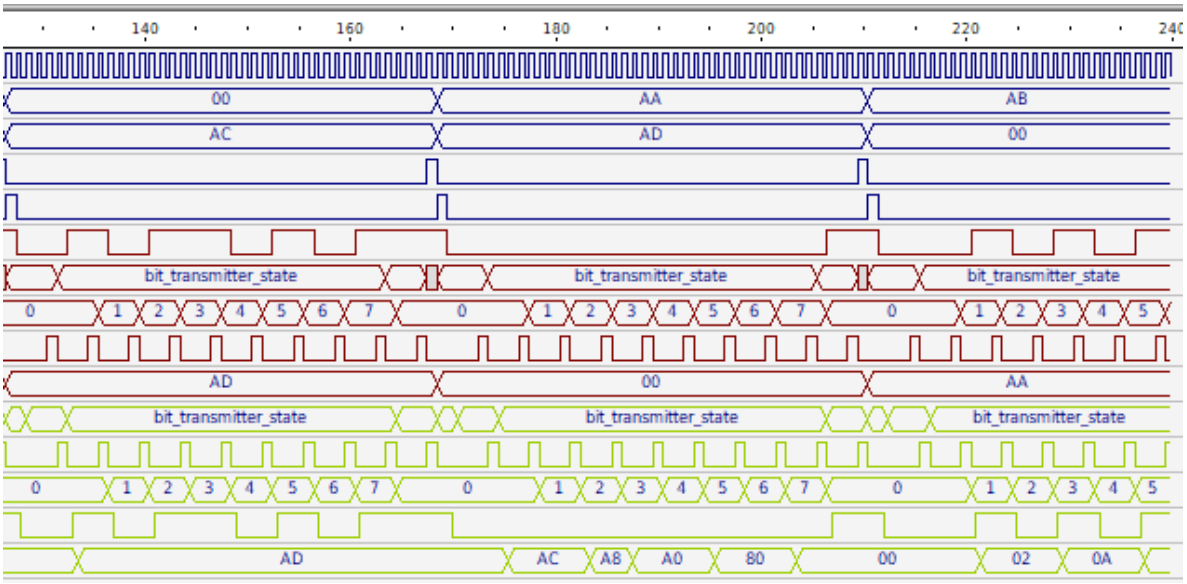

Figure 5: UART MODULE Test subsection2

8 Conclusion

This UART is a configurable programmable logic component that accommodates communication through a simple asynchronous serial interface. It allows a user to specify the system clock, baud rate, data length, parity scheme, and oversampling.