Vol. No.6, Issue No. 06, June 2018

www.ijates.com



VHDL implementation of UART Module using FSM

Gaurav Verma¹, Vishal Rajput², Jashandeep Singh³

^{1,2,3} Students, M.Tech VLSI Designing, ACSD, CDAC Mohali

ABSTRACT

With the technology advanced and the rising of SOC (System on Chip) concept resulting in designing of various modules in efficient manner to be embedded in smaller area, consuming less power and improvement in overall integration. UART which is an abbreviation for Universal Asynchronous Receiver Transmitter is a communication protocol majorly finds application in Microcontrollers and other standard data transfer interfaces. UART can be easily implemented by VHDL or Verilog can be demonstrated on a FPGA board for data transmission. This paper presents the basic implementation of UART module using FPGA and its implementation on Spartan 6.

Keywords: Asynchronous, Microcontroller, FPGA

I.INTRODUCTION:

UART stands for Universal Asynchronous Receiver/Transmitter. UART is a circuit that controls the computer interface so that the computer can exchange data with device. UART is a simple Half-duplex, Asynchronous, Serial communication Protocol between two equivalent nodes generally between a processor and a peripheral. The key functionality of UART involves the fact that receiver itself creates the clock by checking the baud rate of the first bit of the frame. Serial transmission of digital information through a single wire or other medium is less costly than parallel transmission through parallel wires. It is used for short distance, low cost data transmission at the cost of low speed. The UART takes bytes of data and transmits individual bits in a sequential fashion. The fundamental concept employed in UART is parallel to Serial data conversion at transmitter and serial to parallel conversion at receiver end. Any node can initiate communication. Its parameter speed is configurable which makes it universal. Each UART contains a shift register which is the fundamental method of conversion between serial and parallel forms.

WORKING OF UART MODULE:

(a) TRANSMITTER

1. UART transmitter receives data in parallel form from the data bus as shown in Figure 1 below.

Vol. No.6, Issue No. 06, June 2018

www.ijates.com



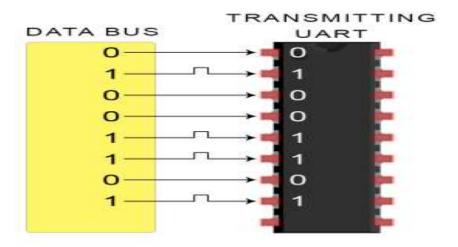


Figure 1

2. The transmitting UART adds the start bit, parity bit, and the stop bit(s) to the data received from the data bus (Figure 2).

TRANSMITTING UART

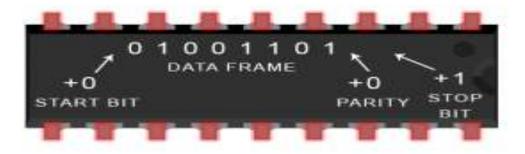


Figure 2

3. The transmitting UART sent the entire frame serially to the receiving UART. The receiving UART samples the data line at the pre-configured baud rate (Figure 3).

Data frame

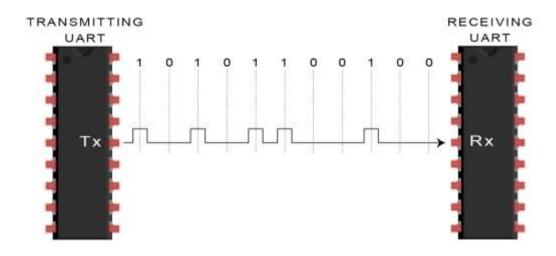


Figure 3

4- The receiving UART discards the start bit, parity bit, and stop bit from the data frame (Figure 4).

RECEIVING UART

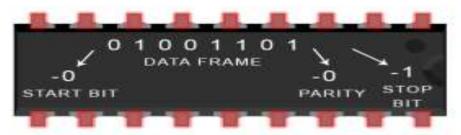


Figure 4

5. The receiving UART converts the serial data back into parallel and transfers it to the data bus on the receiving end (Figure 5).

Vol. No.6, Issue No. 06, June 2018

www.ijates.com



Figure 5

The UART transmitter section includes

- 1. A transmitter hold register (THR)
- 2. A transmitter shift register (TSR).

When the UART is in the FIFO mode, THR is a 16-byte FIFO. Transmitter section control is a function of the UART line control register (LCR). Based on the settings chosen in the LCR, the UART transmitter sends the following to the receiving device:

- 1 START bit
- 5, 6, 7, or 8 data bits (as required)
- 1 PARITY bit (optional)
- 1 STOP bit

Software outputs a byte of data to the THR. The bits are immediately copied into an internal 'shift'-register. The bits are shifted out, one-at-a-time, in sync with a clock-pulse.

(b) RECEIVER

1- The receiving UART discards the start bit, parity bit, and stop bit from the data frame as shown in Figure 6 below.



RECEIVING UART

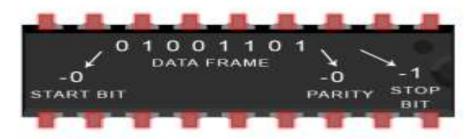


Figure 6

2. The receiving UART converts the serial data back into parallel and transfers it to the data bus on the receiving end (Figure 7).



Figure 7

The UART receiver section includes

- 1. A receiver shift register (RSR)
- 2. A receiver buffer register (RBR).

Receiver section control is a function of the UART line control register (LCR). The UART receiver accepts following from transmitting device:

The receiving UART discards the start bit, parity bit, and stop bit from the data frame

- 1 START bit
- 5, 6, 7, or 8 data bits

Vol. No.6, Issue No. 06, June 2018

www.ijates.com

ISSN 2348 - 7550

- 1 PARITY bit (optional)
- 1 STOP bit

FINITE STATE MACHINE REPRESENTATION

It becomes quite easier for studying and implementation of a module with the use of FSMs. FSMs are symbolic representation of a module with different states and their controlling signals for transitions.

TRANSMITTER STATE DIAGRAM

Figure 8

Figure 8 above shows the Finite State Machine representation of the transmitter section of UART Module designed. Basically it has four states. Idle state is when the transmitter Reset is active and no action is being performed. Then Load_Tx state allows the loading of data bits in register and addition of start and stop bits for completion of frame. Then as UART is ready to transmit, Shift_Tx state is in operation where serial transmission of bits takes place. Finally when all the bits in the frame are sent, the fourth state Stop_Tx is in action where all the registers are cleared and Transmitter is sent back to idle state for next set of data transmission.

Transmitter VHDL Code

```
library IEEE;

use IEEE.STD_LOGIC_1164.ALL;

use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity uarttx1mar is

port

(

clk: in std_logic;

reset: in std_logic;

load: in std_logic;

frame: out std_logic_vector(5 downto 0);

Din: in std_logic_vector (3 downto 0));
```

Vol. No.6, Issue No. 06, June 2018

www.ijates.com

tx_state <= tx_stop;</pre>

ijates ISSN 2348 - 7550

```
end uarttx1mar;
architecture Behavioral of uarttx1mar is
signal tx_regdata : std_logic_vector(3 downto 0);
signal tx_reg : std_logic_vector(5 downto 0);
begin
process (reset, clk, tx_state)
begin
if (reset= '1') then
 tx_regdata <= "1111";
  tx_reg <= "111111";
 tx_state <= tx_idle;</pre>
elsif rising_edge (clk) then
case tx_state is
        when tx_idle =>
  if load= '1' then
  tx_regdata <= Din;</pre>
  tx_state <= tx_load;</pre>
else tx_state <= tx_idle;
  end if:
     when tx_load =>
  tx_state <= tx_shift;</pre>
  tx_reg <= '1' & tx_regdata & '0'; // framing
     when tx_shift =>
tx_reg <= '1' & tx_reg(tx_reg'high downto 1);</pre>
```

Vol. No.6, Issue No. 06, June 2018

www.ijates.com

ijates ISSN 2348 - 7550

when tx_stop =>

tx_state <= tx_idle;
end case;
end if;
frame <= tx_reg;
end process;
end Behavioral;</pre>

RECEIVER STATE DIAGRAM

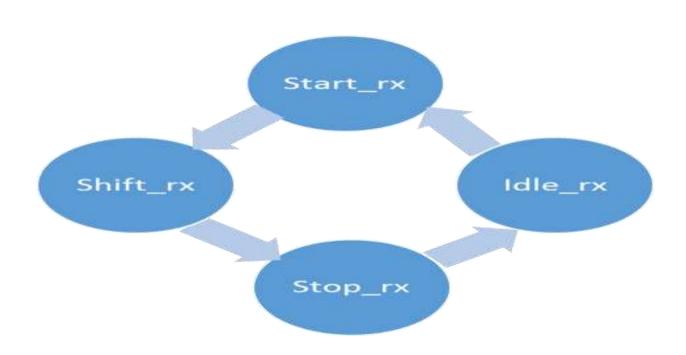


Figure 9

International Journal of Advanced Technology in Engineering and Science Vol. No.6, Issue No. 06, June 2018 ijates www.ijates.com ISSN 2348 - 7550

As in the case of transmitter section, the corresponding receiver section is also designed using four states as shown in figure 9. Here starting from Idle_Tx state where the receiver registers are all cleared. when the receiver starts receiving the bits of frame and first start bit is received, Receiver moves to the Start_Rx start where the start bit is sampled according to baud rate and required no of cycles to receive one bit is determined. The registers are now ready for data reception and receiver is moved to Shift_Rx state where the bits are simultaneously received and shifted in corresponding register. Finally at the completion of all bits of data frame and stop bit the receiver is set to Stop_Rx state and here all the data is moved to output and receiver is finally set to idle_rx state and makes the registers again clear for next set of reception.

Receiver VHDL Code

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity uartrx1mar is
port
clk:in std_logic;
reset:in std_logic;
ready: in std_logic;
rx_reg:inout std_logic_vector(5 downto 0);
Dout : out std_logic_vector (3 downto 0));
end uartrx1mar;
architecture Behavioral of uartrx1mar is
type rx_states is (rx_idle, rx_stop, rx_shift, rx_start);
signal rx_state : rx_states;
begin
process (reset,clk,rx_state)
begin
```

Vol. No.6, Issue No. 06, June 2018

www.ijates.com

```
ijates
ISSN 2348 - 7550
```

```
if (reset='1') then
rx_reg <= (others =>'0');
Dout <= (others =>'0');
framerx \neq (others =>'0');
rx_state <= rx_idle;
elsif rising_edge(clk) then
case rx_state is
     when rx_idle =>
  if (ready='1') then
  framerx <= rx_reg;</pre>
  rx_state <= rx_start;</pre>
else rx_state <= rx_idle;</pre>
  end if;
when rx_start =>
  if (framerx(5)='1') then
  rx_state <= rx_shift;</pre>
  end if;
     when rx_shift =>
  framerx <= '1' & framerx(framerx'high downto 1);</pre>
  rx_state <= rx_stop;</pre>
     when rx_stop =>
  Dout<=framerx;
  rx_state<= rx_idle;
end case;
end if;
```

Vol. No.6, Issue No. 06, June 2018 www.ijates.com ijates ISSN 2348 - 7550

end if;

end process;

end Behavioral;

Output Waveforms:

The following figures from 10 to 26 show the transmission and reception of data bits through frame designed with start and stop bits serially. These were obtained at simulation of VHDL Code separately by combining the transmitter and receiver modules discussed above with a baud rate generator.

			cello con	e	garren.	-71	C25 - 114 C25	40-130-4	< 1 11 11 - 1		Symonope	
Name	Value	2.899,891,00	(2.999,992 pe	2.895,902 ps	(2,999,994.06	2.89.95 m	(Annual or	12,996,997pm	[Lines/reques	D. Williams Co.	1,000,000 ps	13,000,0
in class	10101188					30132100						
la tijena la tojenstrti	0 1 0000223#					80000000						
in tracese in manutautate in esk counts	mart_pat 1					start_bit						
le be indes ■ to detail? (il le to done att	0 10101168 0					20121100						
lig tounts lig tilkd lig tilk bit	11110100001 1 1100000	11111/1000010	1 L1110 300000 mm	1111110000000) LISTE SOME SOM	(111110000E) 130000E	LITTERSENS	111111000E200	111110006308	11111100001001		
		X1: 3,000,000 pe										_

Figure 10

ACTIVE	Voture	5,995,5	Ol ps	5,999,992 pm	5,999,993 ps	5,999,064 pt	5,999,085 ps	5,999,996 m	5,999,997 px	5,999,998 pm	5,995,990 ps	6,000,0
dkir.	1	1000		2010 012200	Transport Children	0.0000000000000000000000000000000000000	100000000000000000000000000000000000000		100000000	A STATE OF THE STATE OF		
reset												
to bytelxical	20103108						30101300					
at tree	#											
to active	\$											
te cenufrei	80000008						93056930					
m ta done	# ·											
lik main to state	start_bit						stiet bit					
elk_counts	111						111					
le tit index							9					
to data(7:0)	20103106						30101300				_	
to dene att	The second	-								-		
e courts	2111818180001	11120	10993011	1113015900011	1113010000011	111101000011.	111101000011	1111019000ETL	1111030000011	1111033006611	1111H10000530	
The click of	+											
	1100000						t190000					

Figure 11

Vol. No.6, Issue No. 06, June 2018

www.ijates.com

iintee
ijates
ISSN 2348 - 7550



Figure 12



Figure 13

Lene	Value	\$4,990,091ps	D4,999,092 pm	Q4,999,993 ps	D4,998,994 pm	P4,994,995 ps	(34,999,996 ps	Q4,990,997 ps	(24,999,990 ps	D4,999,998 pm	25,000,000 ==	25,000,001 ps
in other			In the same			10001000			Transaction of the last		131111111	CALLET TO
reset	6											
Tig to byte(7)18	PRESENTE					10301306						
in states	9											
or to active	1											
to sensibili	10101000					10101500						
m to storie	9		_	_	_		_					
in more to state	data_bis					deta_mi						
in the counts	101100					101100						
in the index	101					ш			_			
to contained lie to door note	10101100					10101306						
lie counts	11110100000	111101000000	111111000000E	11110 200000001	11111110000001	1111918080001	timinuosossi.	T111101000001	1111010000001	1111019000001		
Tile cours	1		The state of the s			-	200000000000000000000000000000000000000		The state of the s	Zaniamont		
The effection	1100000					130000						
111111111111111111111111111111111111111												
		K1: 25,806,000 pm										

Figure 14

Vol. No.6, Issue No. 06, June 2018 www.ijates.com

i	jate	e	S	
ISSN:	2348		7	55

nie .	Value	In section in	G2, 999, 992 pe	12,999,961 ox	III,998,99496	DESCRIPTION OF THE PERSON.	TEC-1000-1000-000	E.988,987ps	32,999,998 px	10,500,000 pc	HLORO, HIED DO	1.080,811
elkon												
incet:												
ta_bate(7:0)	10101100					901815900						
at free	0											
in busine	1					-000						
ta_seruE7XE	Tatattee					10101100						
to done	0					STATE OF THE PARTY						
in main te state	data_hit					deta_91						
de die courts	141110					111100						
in bit meek	111					111						
to data(foli)	1,0191118					10181100						
to done ask	0											
lig county	III.13100000	11.13033000000	11110300000000	1111030000000	11111F30900000	1,111110000000	1111110000000	1111111100000000	1111010000000	1111919000000		
Ug our	A.											
I dk tot	Limeter					1100000						

Figure 15

lane	Value	20,990,901 (6)	31, 909, 992 pa	more, miles	(B) 999, 994 (4)	20,999,999 pm	125,000,000.00	10,000,00714	25,500,000 pa	55,000,000 ps	96,000,000 ps	738,000
in chin	1		OF THE PARTY.		The second second		100000000000000000000000000000000000000			0.000	THE REAL PROPERTY.	100
in reset	W											
tx_byse1700	10101100					30191100					1	
Les al toyo	0					100000						
in to active	1										1	
te sariative	10101100					271111100						
ig te date	0					100000						
lat main, to state	stop_hit					ntop_trik						
lig th rounds	1000118					\$2000 to					1	
till reder	1000					1909					į.	
Mg to data(7.0)	10101100:					10101100					1	
in tedane and	0											
in tourts	11118300000	111203000000	T111030000000	TTTTE:300000000	F1111 30000000	111100000000	11111130000000	1117210800000	1111010000000	1113019000000	1	
lig steat	+										1	
The excess.	3100996					1300000					1	
		311 36,600,000 pc										

Figure 16

Name	Yelse	[39,999,991.pc	39,099,992 ps	39,999,993 pe	(39,099,994 pe	39,099,995 pa	(39,999,995 pe	39,999,997 px	29,999,995 ps	29,099,996 ps	40,000,000 pe	40,000,00
e cto	1											
neset	1											
to Jeylet 7:01	18163109					10301300						
a 4,500	1											
to_active	=											
to_serta@bill	28161100					10101106						
in to done	1											
in reaso, in claim	stop_bis					stop let						
of the reachts	FE67010					900:00:00						
in tot index	1000					5060						
to detail in	19103100					10:001:00						
ig to done and	4											
d courts	11110011111	THE REAL PROPERTY.	1111001515111	1311001111111	1111001111111	INCOME STREET	1111003111111	1111001111111	1111000111111	1113003111111		
A zez	+											
16 ca.se	1766600					1100000						
												_
		T3-≪,000,000 pe										

Figure 17

Vol. No.6, Issue No. 06, June 2018 www.ijates.com

1	ja	te	S
ISSN	23	48	75

50



Figure 18

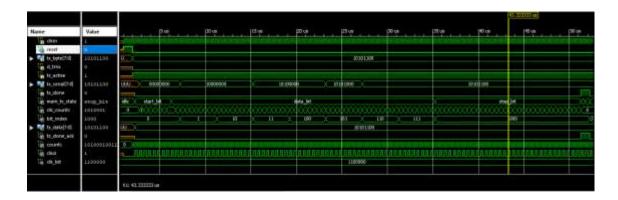


Figure 19



Figure 20

Vol. No.6, Issue No. 06, June 2018 www.ijates.com

ijates ISSN 2348 - 7550

			ALCOHOL: N	removement.		110000000000000000000000000000000000000	-0-0300-	Continues to	roccutor :-	I Company of	100000		227-34
luces	Value	15	MINIST.	pt.990,900		E-990/99444	K.990,995 EF	H. 990, 996 pt.	K,996,907 ==	H. 900,000 pt	K.996,994	7,080,000 ps	7,060.00
m (800)	1											-	
in read	E												
totap	£												
in cotaet:	0												
esili, (MSZ10)	19107039						10000	99					
in a driven	4												
ex beta (7.0)	unneces:						0,000	10-					
e main state	data_Eut						cheta						
ment TO	10101038						10000	Ú1					
In durant	1015						100						
lig na inder													
received_byte(75))	(####PECCII						0.000	110					
in receive ack.	0												
in court	11110000000111100	=00	1119:00:00:00:11	111(11000	1100	[11112:0000001]	111(010000011	[1112000001]	111101000011	1111210000013	111101000011		
lan court	Å.												
ORD CHANGE							100000	n)					

Figure 21



Figure 22



Figure 23

Vol. No.6, Issue No. 06, June 2018 www.ijates.com

ijates ISSN 2348 - 7550



Figure 24

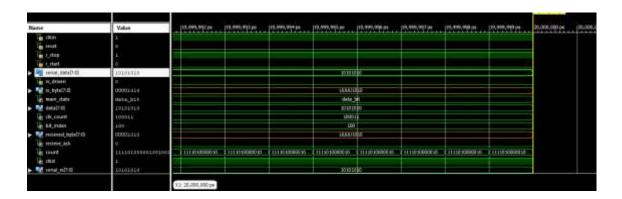


Figure 25

Norre	Value	(30,999,952 pa	50,999,995 pm	30,999,994 pm	50,999,995 pe	30,999,995 pe	30,989,997 pm	(30,999,995 pe	50,999,999 pe	31,000,000 pe	12110
etition.	12.00					11021710		12001000			
m reset	0.										
To t stop	X-1										
g (,dat	4										
Manufaction (7:0)	10212010				18 80 80	ģa —					
in in driven											
▶ 🌠 ro_toyled7t5i	00181010				UE1010	9.7					
in main State	data_bis				699.						
■ *** carta(7:0)	10181010				18 10 10	gail.					
in ck court	111101				15180	A.					
The bit ledes	111				111						
▶ M received Sytel7/6	00181010				VE0000	9.5				5	
Tig repeve ack	# C C C C C C C C C C C C C C C C C C C					ř .					
life court	111223200000011223	1111100000000	1111110000000	111100000000	11111100000000	111101000000	1111111000000	1111010000000	1111010000000		
Illis atkid	1:										
▶ Me sensi relititi	18282830				16 90 10	No.					

Figure 26

Vol. No.6, Issue No. 06, June 2018

www.ijates.com

ISSN 2348 - 7550

II.DISCUSSION:

UART module is implemented using VHDL and verified on FPGA SPARTON 6 kit. 8 bits of data were sent and received successfully. Further detailed investigation can be done to verify the number of clock cycles needed for receiving each bit. Sampling rates and baud rates can be varied for further variety of rates. Further work can be in betterment including more number of states for improved data transmission.

REFERENCES

- [1] FPGA implementation of Advanced UART Controller using VHDL, Arpita Tiwari, Ravi Mohran, Divyanshu Rao, IMPACT: IJRET, Vol.2, Issue 8, August 2014, Pg 23-32
- [2] VHDL Implementation of Universal Asynchronous Receiver Transmitter with Linear Feedback Shift Register, Simarjeet Singh, Amardeep Kaur, IJARCSSE, Volume 7, Issue 5, May 2017
- [3] Designing of UART using VHDL, Renuka Hingmire, Tamanna Jethwa, Aishwarya Askar, Sagar Mahajan, Prashant Shende, ISSN, Volume-3, Issue-3, 2016
- [4] A VHDL Implementation of UART Design with BIST Capability, Mohd Yamani Idris, Mushkuri Yaacob, Zaidi Razak, University of Malaya, Malaysia, Malaysian Journal of Computer Science, Vol. 19 (1), 2006
- [5] Fang Yi-yuan, Chen Xue-jun. "Design and Simulation of UART Serial Communication Module Based on VHDL," Intelligent Systems and Applications (ISA), 2011 3rd International Workshop vol., no., pp.1-4, 28-29 May 2011.
- [6] Naresh Patel, Vatsalkumar Patel and Vikaskumar Patel. "VHDL Implementation of UART with Status Register" 2012 International Conference on Communication Systems and Network Technologies (IEEE).
- [7] Hazim Kamal Ansari, Asad Suhail Farooqi. "Design Of High Speed Uart For Programming FPGA" International Journal Of Engineering And Computer Science Volume 1 Issue 1 Oct 2012
- [8] Ananya Chakraborty, Surbhi, Sukanya Gupta, Swati Deshkar, Pradeep Kumar Jaisal. "Design of UART (Universal Asynchronous Receiver Transmitter) using VHDL", IJCST Vol. 3, Issue 1, Jan. March 2012