Assignment- 8

1) *Introduction*

In this lab we have built a Receiver in which FPGA board receive 8 *bits data written on "gtkterm screen". Gtkterm is a software which is used for manually providing input so that the "Receiver" can read that data.*

UART (Universal Asynchronous Receiver Transmitter) is used as an interface for communication between Basay 3 *board and PC.*

Some important points regarding UART :

a) UART is a serial interface , so transmitter send data bit by bit and at same time receiver has to capture each bit and convert it into parallel data.

b) Baud rate is a standard feature of URAT which tell us number of bits transfer per second.

c) URAT interface allow use of different Baud rate like 300, 600 , 900, 1200, 9600, 115200 *etc.*

d) Similar to Baud rate there are different variants parameter like 'Data bits' , 'Parity' , 'Start bits' and 'Stop bits'.

Some basic feature of Receiver :

a) Serial − in Parallel − out register

b) Shift the bits serially into a register

c) Read out data in parallel

Some basic feature of Transmitter:

a) Parallel – in Serial – out register

b) Load data in parallel in a register

c) Shift out the bits serially

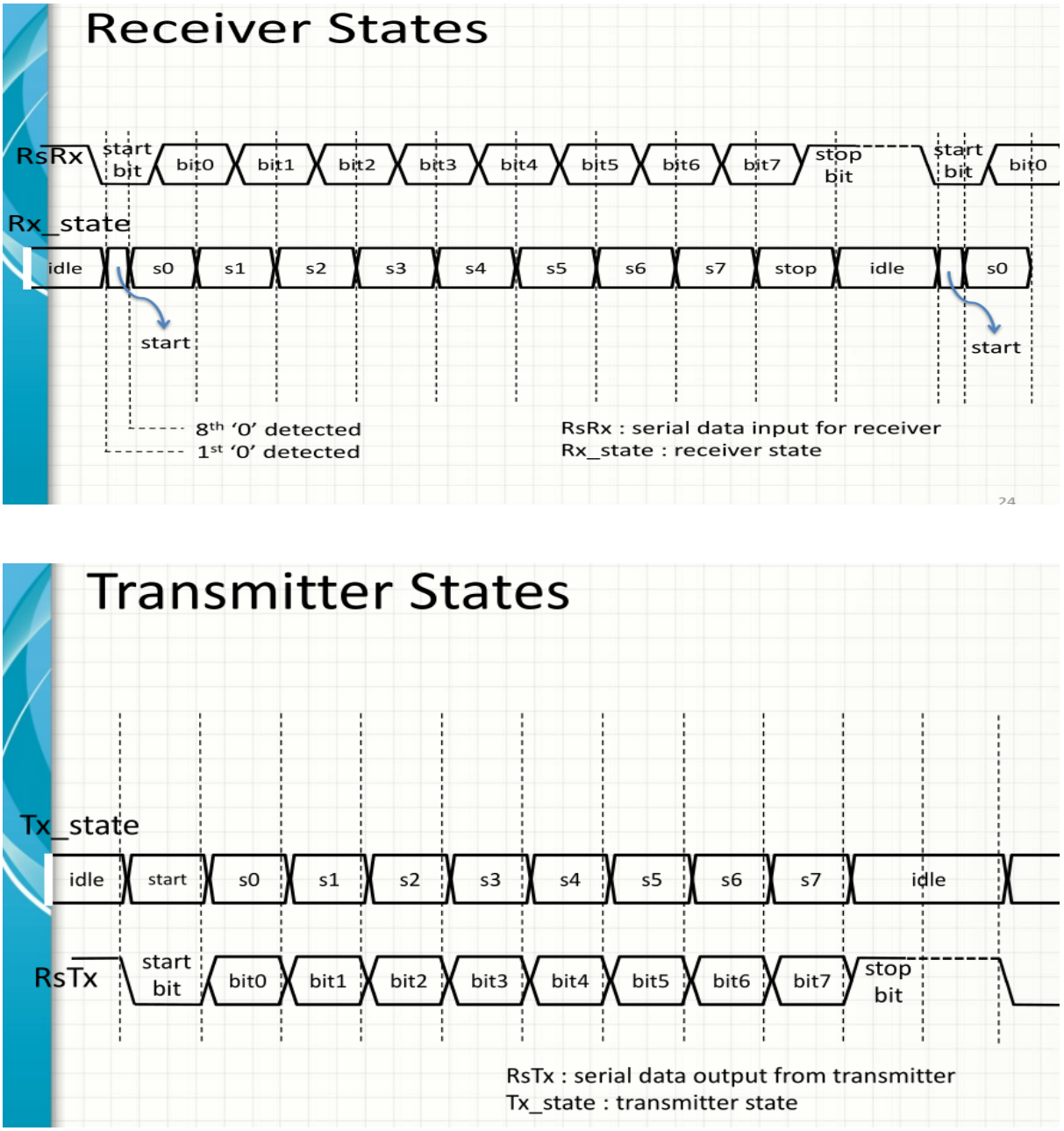


Fig 1 : Basic diagram of Receiver and Transmitter States

2) *Implementation Design*

a) Receiver

In this lab we have implement a Finite State Machine (FSM) /State Machine which is based on bit − data received by Receiver Input pin (rx − in) . We have also used FPGA clock of 100*MHz as clock for the receiver (rx − clock), reset pushButton is also used so that 'idle' state can be reached.*

To sample 8 *bit data we have used counter(temp − count) and with the help of new − clk we are sampling the middle of each data bit .*

Calculation of counter in receiver :

i) Rx − clock frequency = 100 *MHz.*
ii) Baud rate = 9600
iii) Number of clock cycle per bit = 16

As we are using middle data bit sampling method:

$$counter = \frac{(Rx-clock\ Frequency)}{(Baud\ rate)*(No.of\ clock\ cycle\ per\ bit)*2} = \frac{10^8}{(9600*16*2)} = 351$$

b) Transmitter

In this part we have implemented FSM of Transmitter , this FSM has 3 *states which are as follows : Idle, Start, Read*

It Transmit data obtained by receiver to PC "gtkterm" software due to which we can see the "key" pressed on keyboard which was received by the FPGA Board. It uses the same cycle of Receiver .

$To\ sample\ 8\ bit\ data\ we\ have\ used\ counter(clock\ -\ counter)\ and\ with\ the\ help\ of\ tx\ -\ clock\ we\ are\ sampling\ the\ middle\ of\ each\ data\ bit\ .$

$Calculation\ of\ counter\ in\ transmitter$:

$i)\ Tx\ -\ clock\ frequency\ =\ 100\ MHz.$
$ii)\ Baud\ rate\ =\ 9600$

$counter\ =\ clock\ cycle\ for\ consecutive\ 8\ bit\ data\ = \frac{Tx-clock\ frequency}{Baud\ rate\ *2}$

$counter\ =\ \frac{10^8}{9600*2} = 5208$

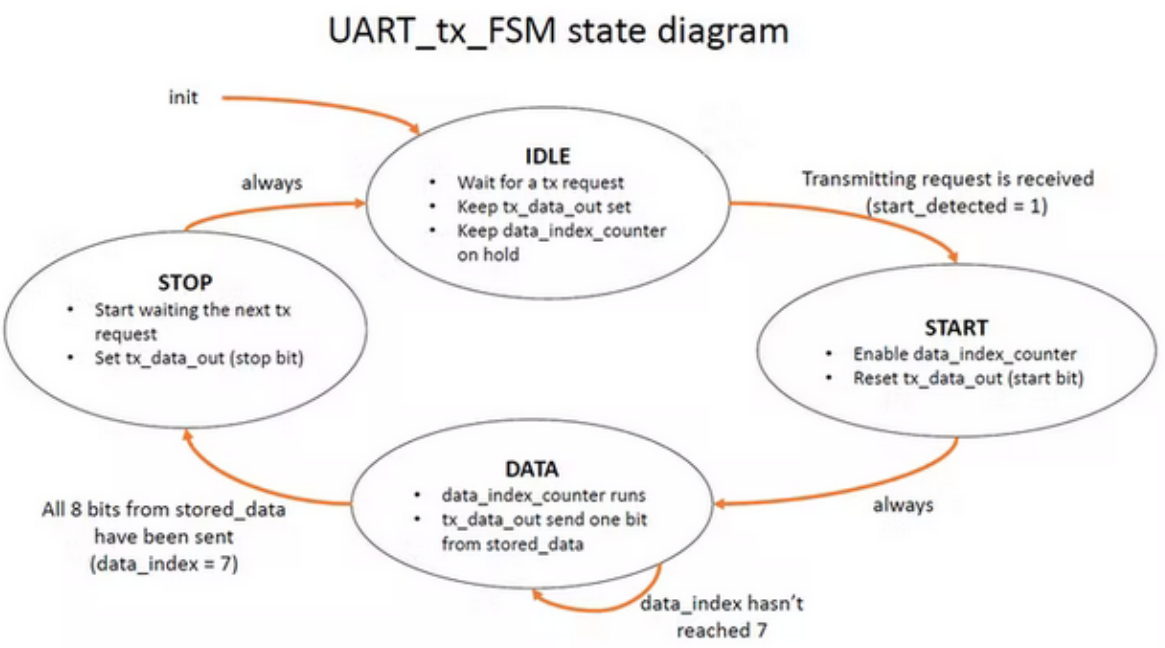
$State\ Diagram\ (FSM)\ \ Description\ of\ Transmitter$:

$3\ States\ of\ FSM:\ Idle,\ Start\ , Read\ -\ and\ -\ Transmit$

$a)\ Idle:\ If\ the\ tx\ -\ start\ =\ '0'\ then\ state\ will\ be\ in\ Idle\ state\ and\ in\ this\ state\ transmitter\ will\ not\ be\ transmit\ any\ bit.\ If\ the\ "tx\ -\ -\ start"\ value\ changes\ from\ 0\ to\ 1\ then\ "Idle"\ state\ will\ transition\ to\ the\ "Start"\ state.$

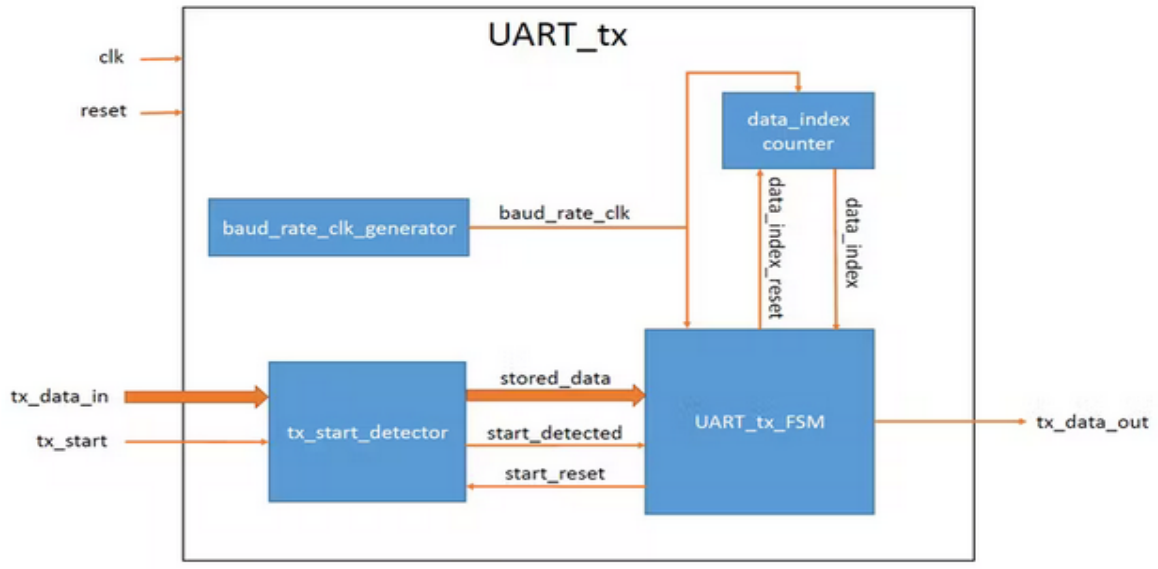
$b)\ Start:\ In\ this\ state\ after\ setting\ "bit\ -\ received"(work\ as\ index)\ to\ '0'\ it\ changes\ to\ Read\ -\ and\ -\ Transmit\ State$

$c)\ Read\ -\ and\ -\ Transmit:\ In\ this\ state\ "tx\ -\ output"\ transmit\ one\ by\ one\ 8\ bit\ data\ which\ it\ extract\ from\ "storing\ -\ vector"\ (in\ which\ Receiver\ value\ was\ stored)\ and\ finally\ move\ to\ "Idle"\ state.$



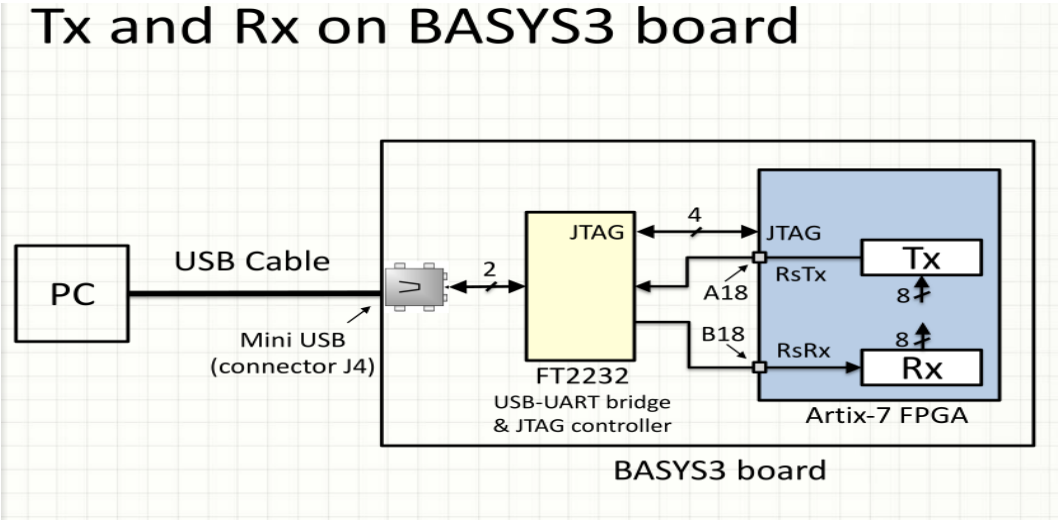
The state diagram of the UART_tx_FSM

Fig 2 : Transmitter FSM diagram



Block diagram of the UART_tx.vhd file

Fig 3 : Block Diagram of Transmitter



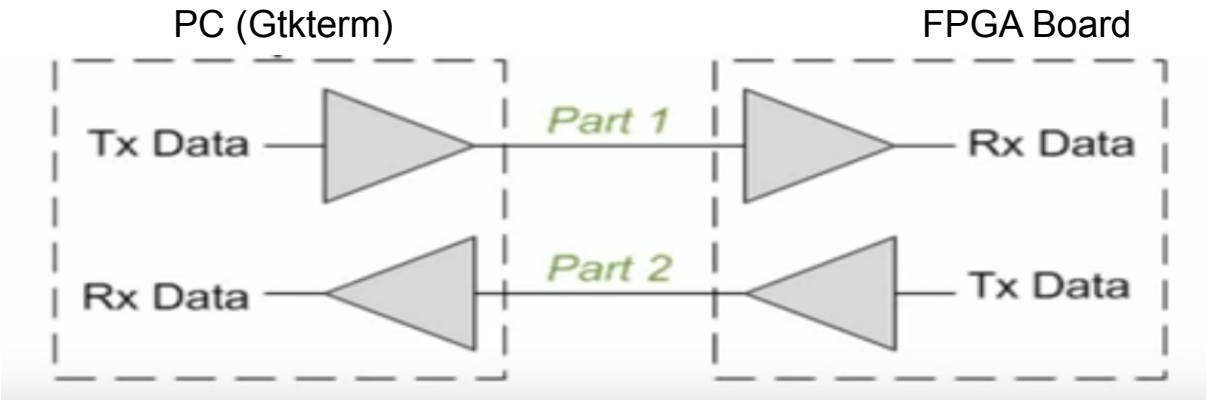
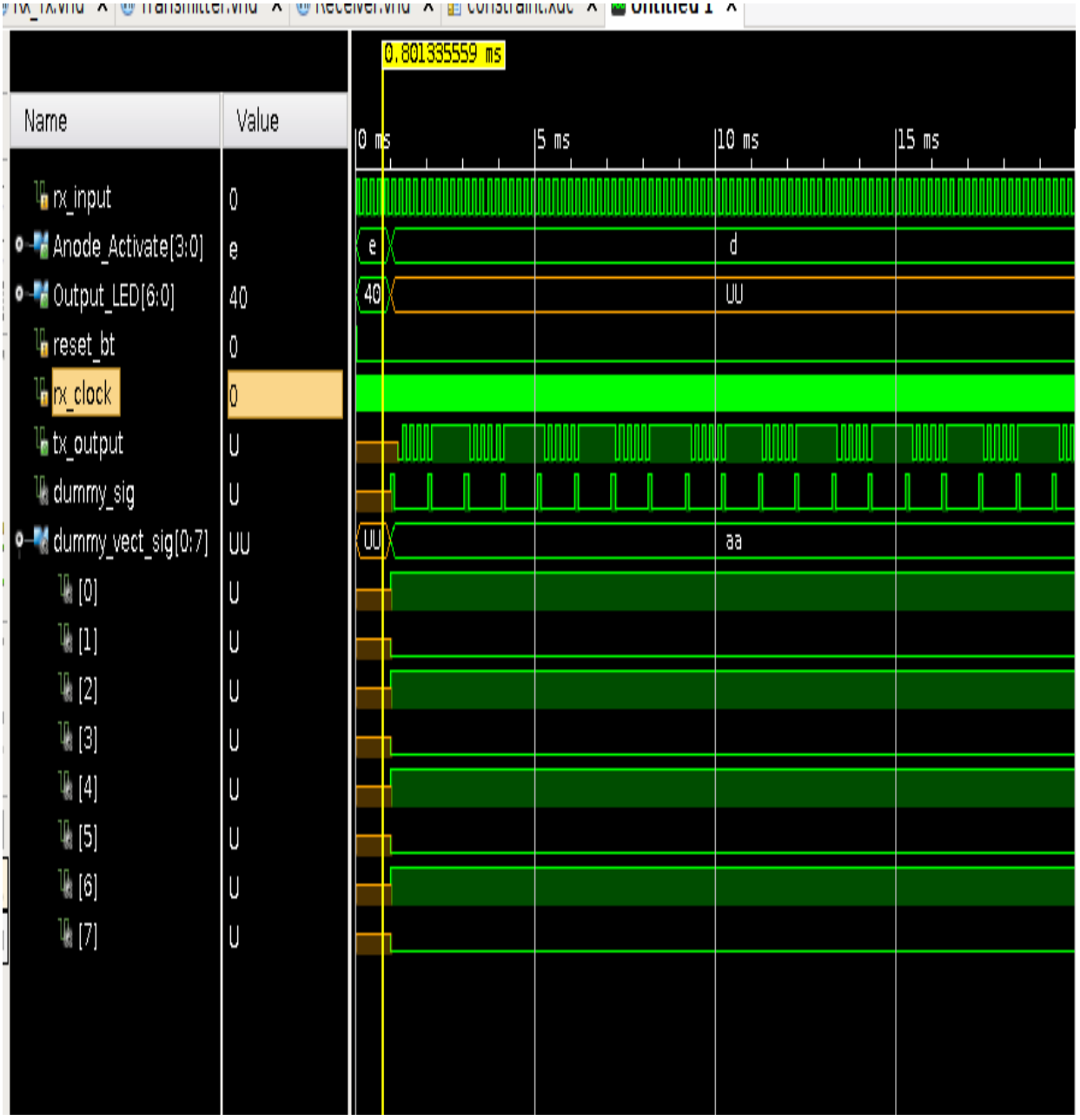
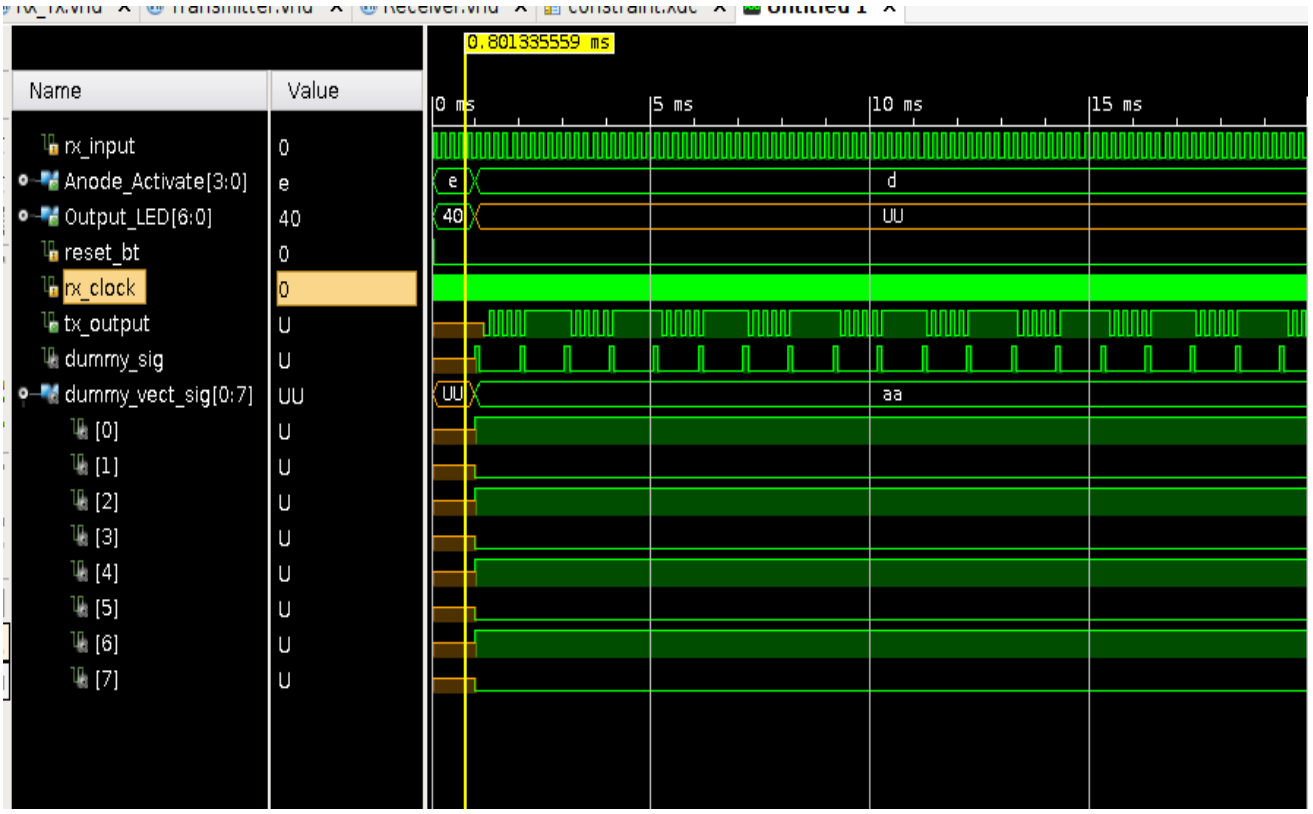


Fig 4: Basic block diagram of Receiver and Transmitter

3) *Simulation Waveform of Receiver – Transmitter*



(a)



(b)

Fig 5 : *Waveform for Receiver − Transmitter at different input*

4) *Digital Circuit for Receiver – Transmitter*

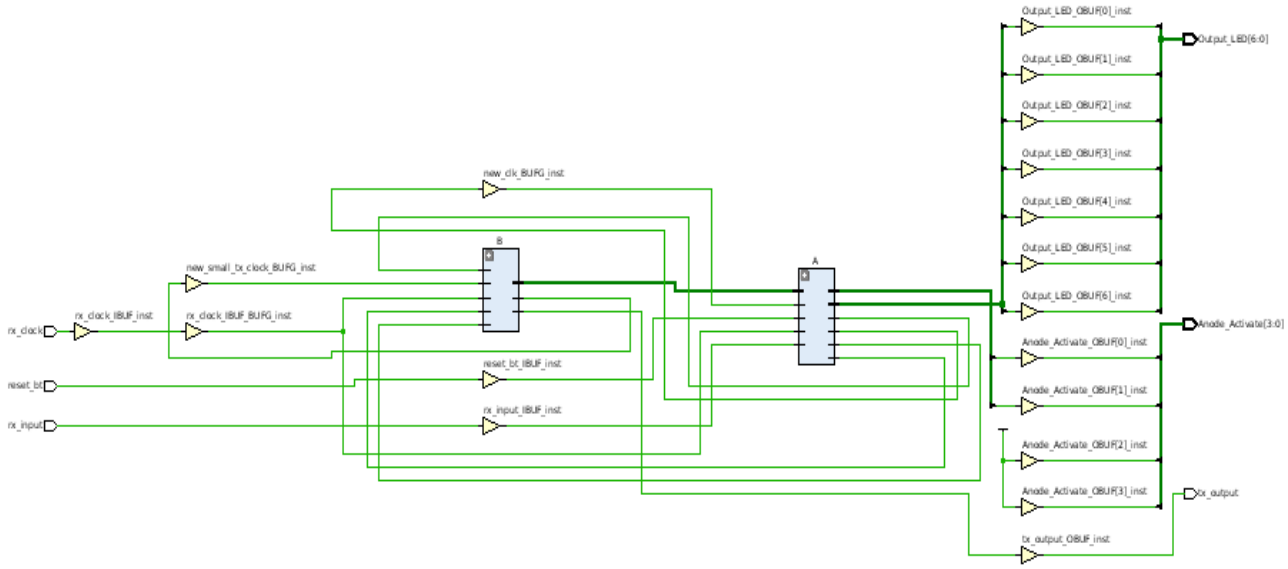


Fig 6: *Complex Digital Circuit for Receiver*

5) *Resource Utilisation*

a) LUT Memory =0
b) LUT logic = 660
c) DSP =0
d) Flip Flops =175
e) BRAM = 0

6) *Some other relevant diagram for resource utilisation*

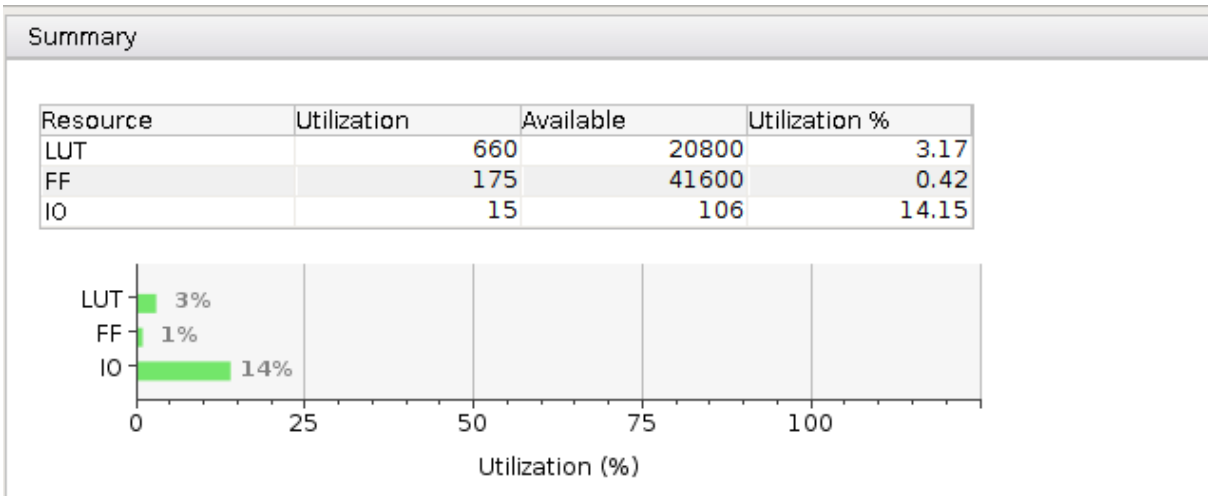


Fig 7 : Summary

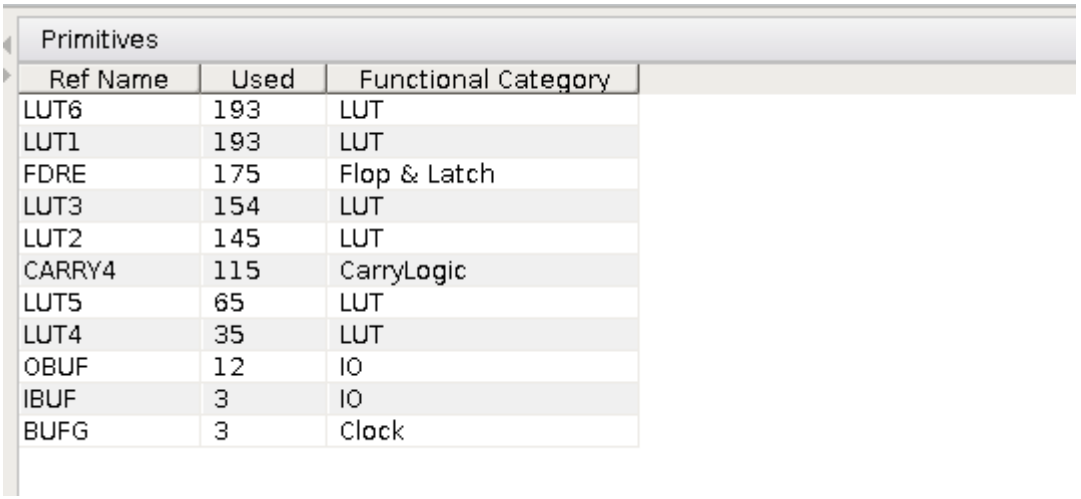


Fig 8 : Primitives

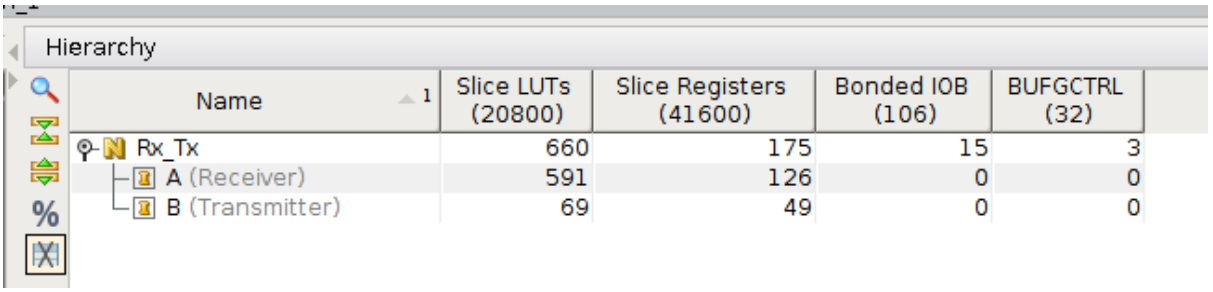
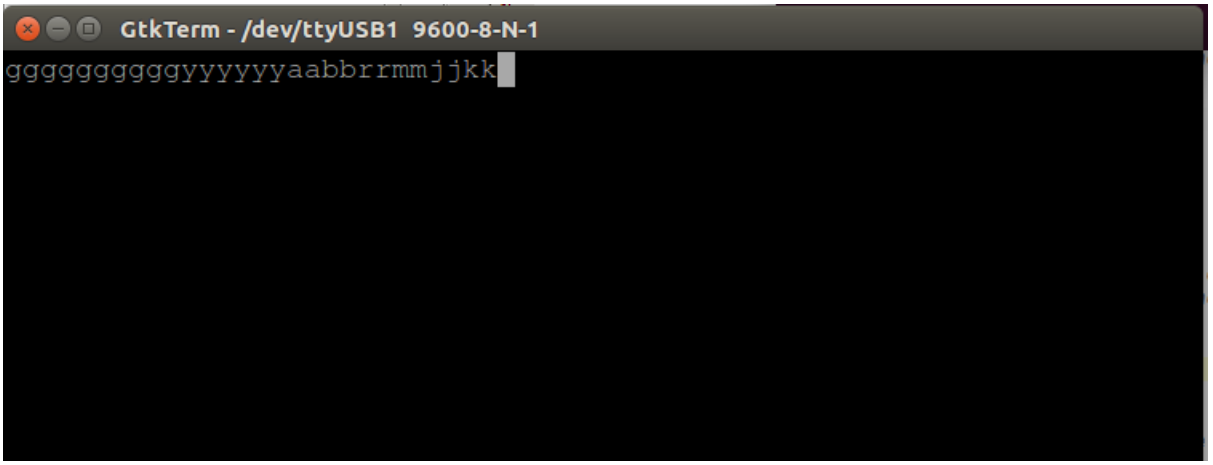
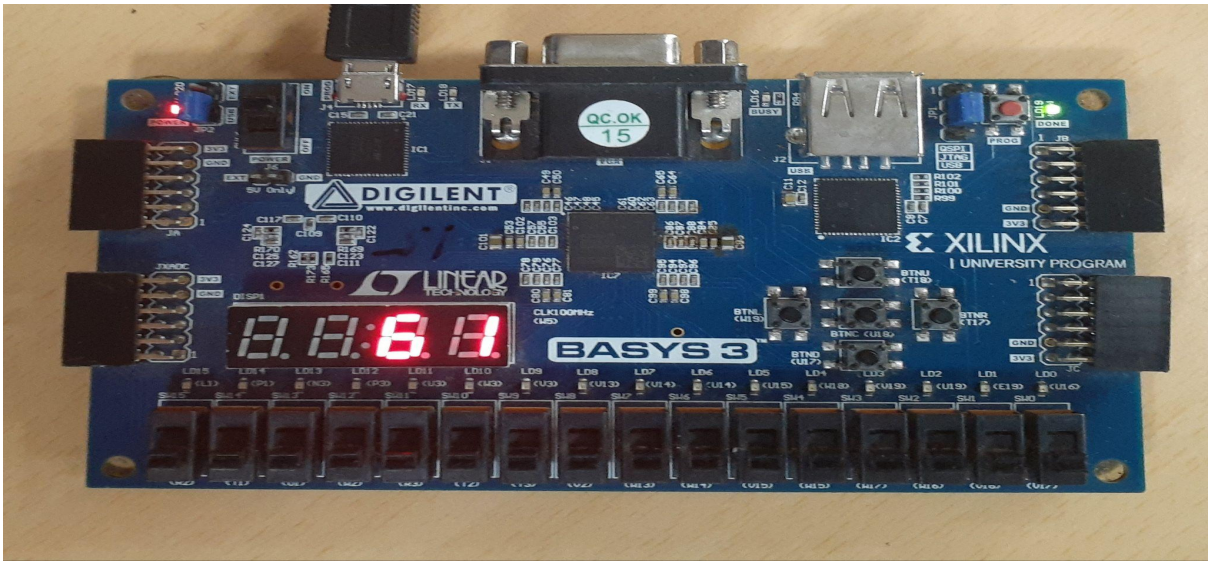


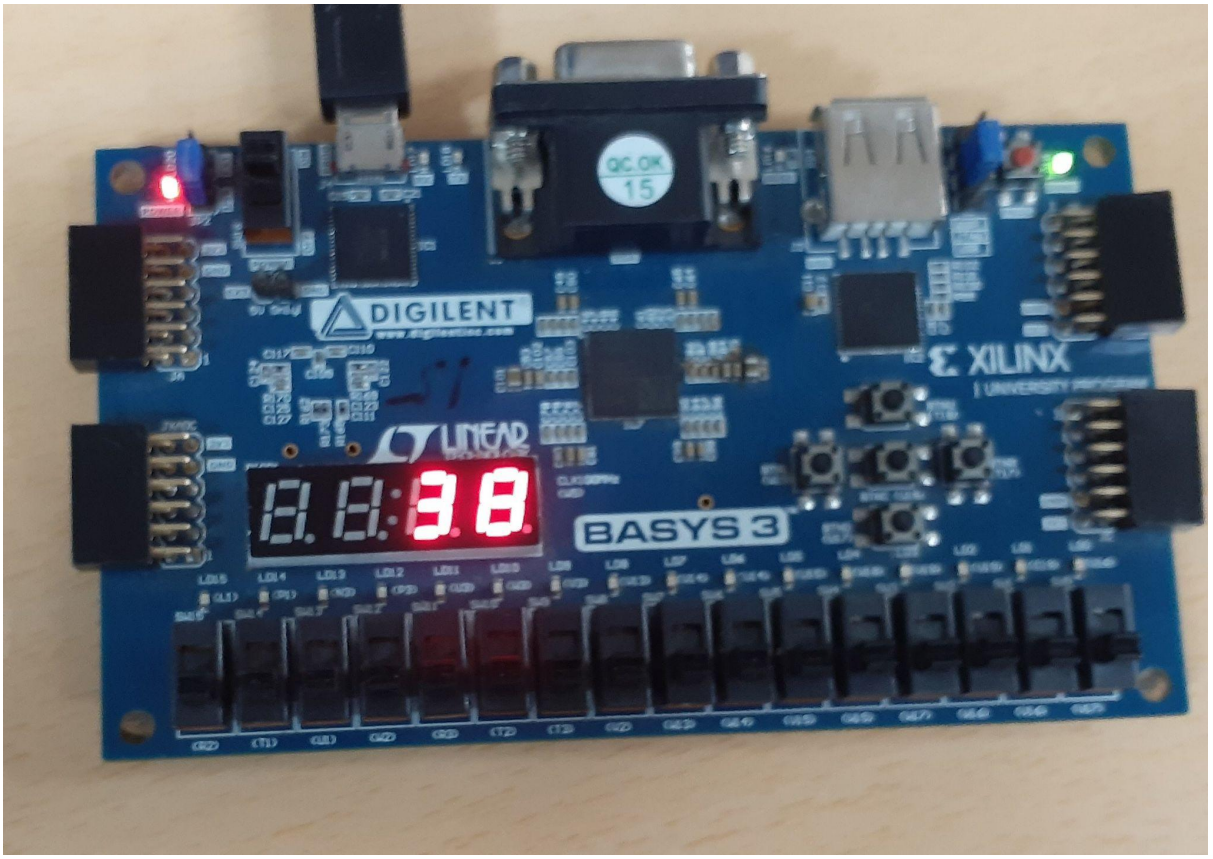
Fig 9 : Hierarchy

7) *Some photographs of FPGA Board and "Gtkterm Screen"*




QC.OK
15
DIGILENT
XILINX
UNIVERSITY PROGRAM
LINEAR
BASYS 3

(a)


QC.OK
15
DIGILENT
XILINX
LINEAR
BASYS 3

(b)