

GCD Calculator RTL Model

Final Report

Name :- Yogesh Mahajan

Roll No. :- 14D070022

GCD of 8 Numbers (GCD8)

Pseudocode

GCD8

```
when rest =>          --wait till start signal
    init count = 8;
    if(start = '1') then next_state = update; --to set values in
register

when update =>
    set GCD2 outAccepted; --so that gcd2 can go to rst state
    if (counter = '0') then next_state = done; Enable outReg;
    else count = count - 1; next_state = getInput;

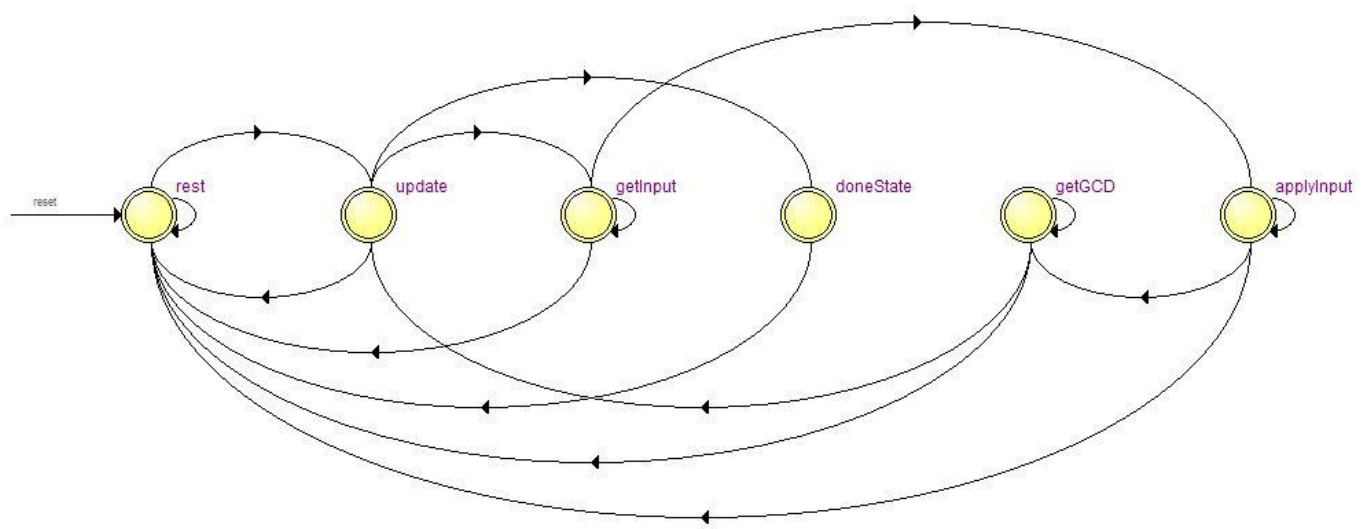
when getInput => --get inputs from external enviornment
    set srdy;
    if(erdy = '1') then
        next_state = applyInputs;
        if(count = 7) then apply input to both areg & breg; --setting
initial GCD = first input
        else apply input only to areg;

when applyInputs => --apply inputs to GCD2
    if(GCD2 is redy) then
        set input redy for GCD2;
        next_state = getGCD;

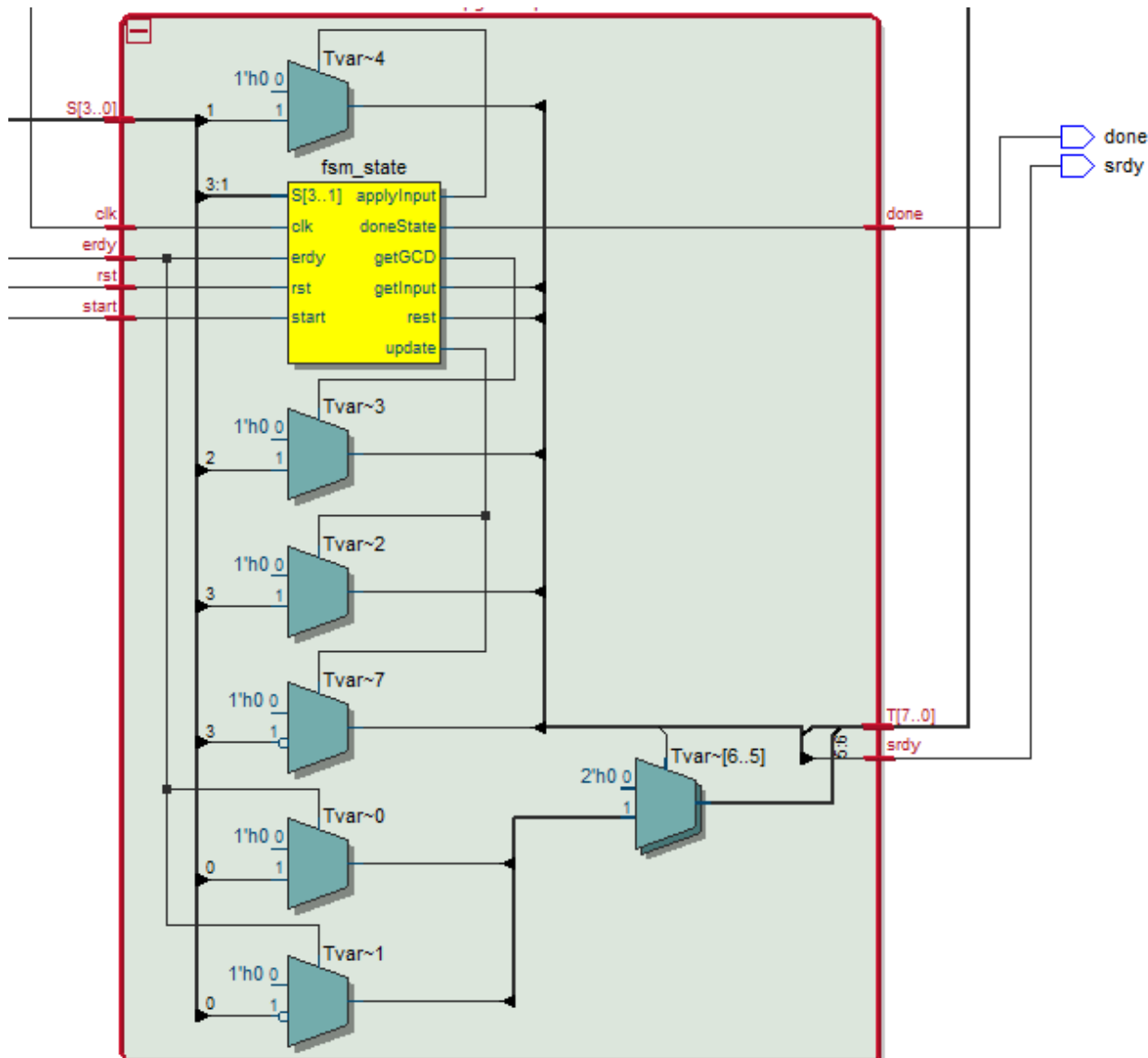
when getGCD => --get GCD of two numbers from GCD2
    if(GCD2 output = '1') then
        store GCD in breg;
        next_state = update;

when done =>
    set done flag;
    next_state = rest;
```

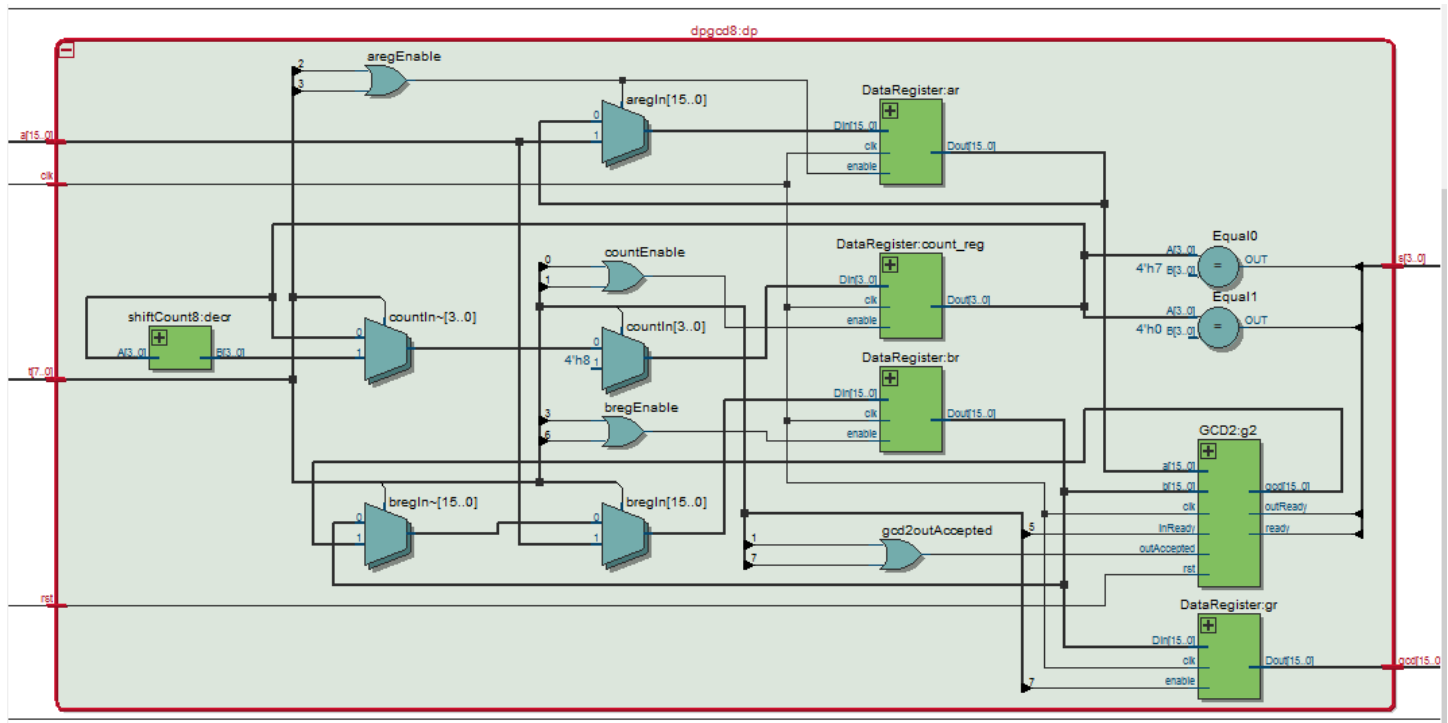
FSM State Diagram



Control Path



Data Path



GCD of 2 Numbers (GCD2)

Pseudocode

GCD2

```

when rest =>
    set ready flag; --GCD2 is ready to accept two numbers
    if(inputReady = '1') then
        next_state = divide;
        store inputs in areg and breg;

when divide => --apply inputs to divider
    if(dividerReady) then
        set dividerInputReady;
        next_state = accOut;

when accOut => --get remainder from divider
    if(dividerOutReady) then
        next_state = update;

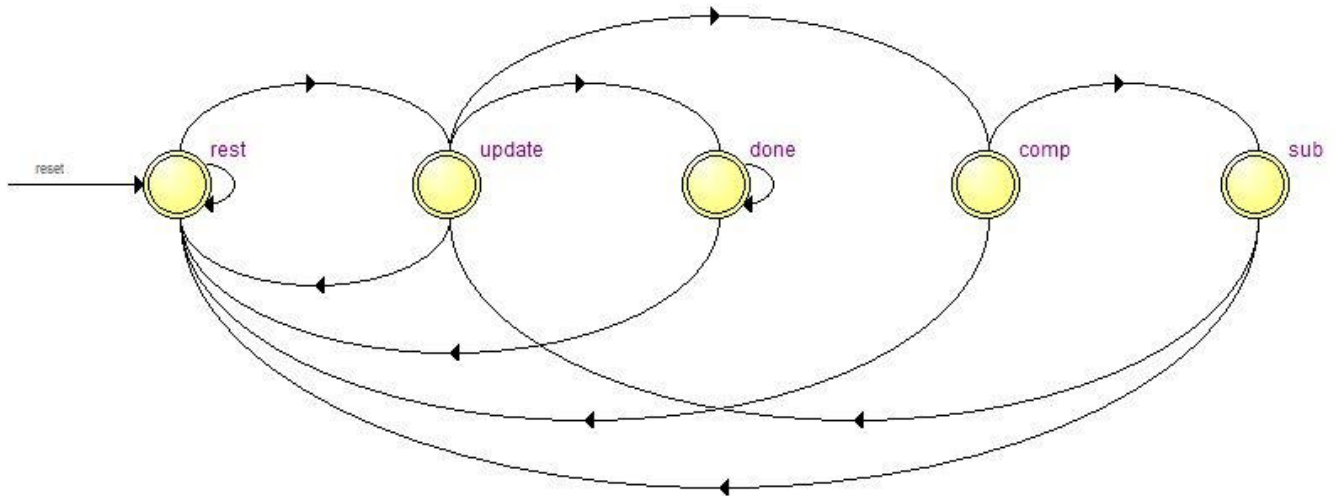
when update =>
    store breg in areg;
    store remainder in breg;
    set GCD2outputAccepted;
    if(breg = 0) then next_state = done;
    else next_state = divide;
    
```

```

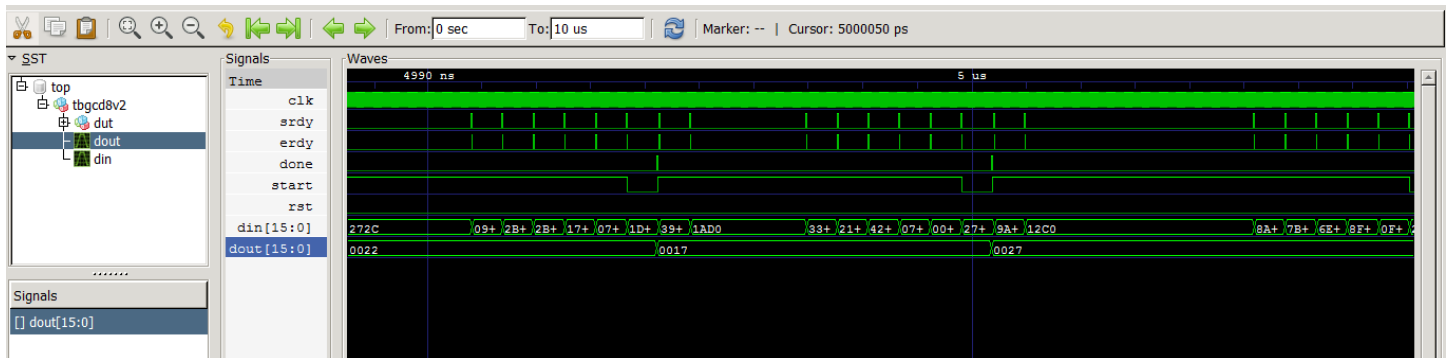
when done =>
  set done flag;
  wait until(outAccepted = '1')  --wait till GCD8 accepts output
  next_state = rest;

```

FSM State Diagram



GHDL Simulation



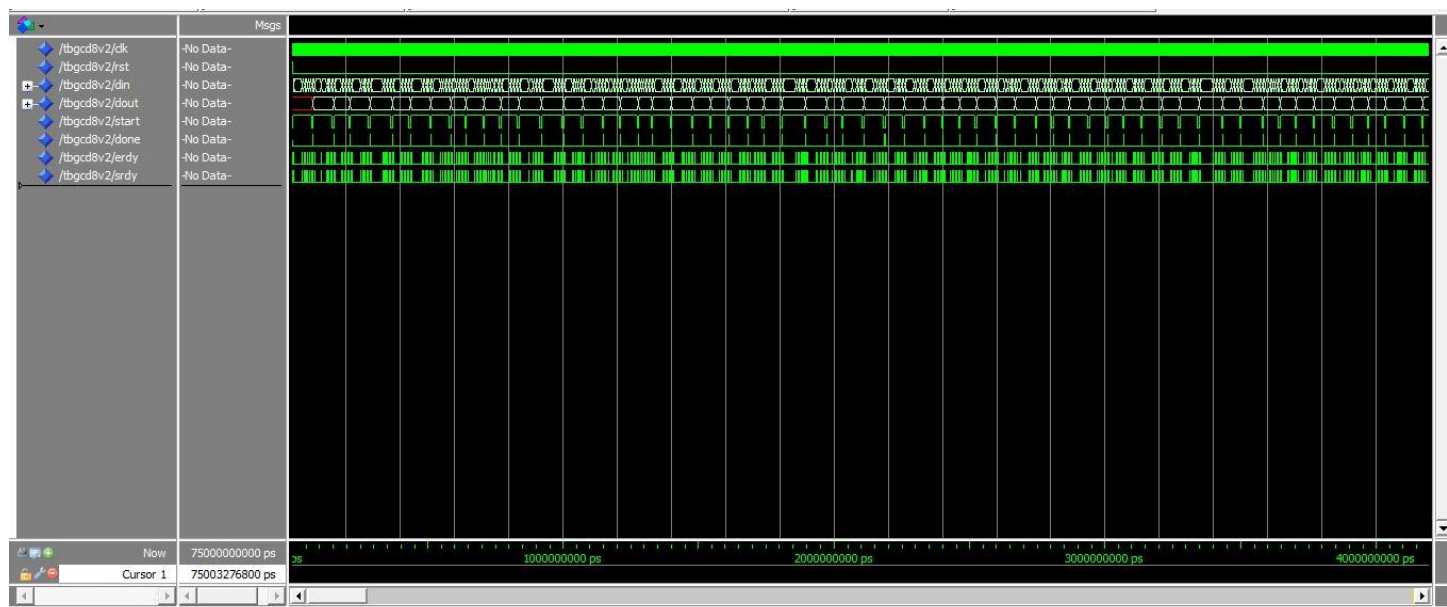
Total 1000 cases

RTL Simulation

Transcripts

```
Transcript
# Loading work.controlpath(benave)
# Loading work.datapath(blahblah)
# Loading work.mux21(structure)
# Loading work.shiftcount16(serial)
# Loading work.lshift16(structure)
# Loading work.sub16bit(structure)
# Loading work.fulladder(fulladder)
# Loading work.halfadder(halfadder)
#
# add wave *
# view structure
# .main_pane.structure.interior.cs.body.struct
# view signals
# .main_pane.objects.interior.cs.body.tree
# run 75 ms
# ** Error: Total Checked 1000 failures 0
# Time: 72102950 ns Iteration: 1 Instance: /tbgcd8v2
VSTM 2>
```

Waveform



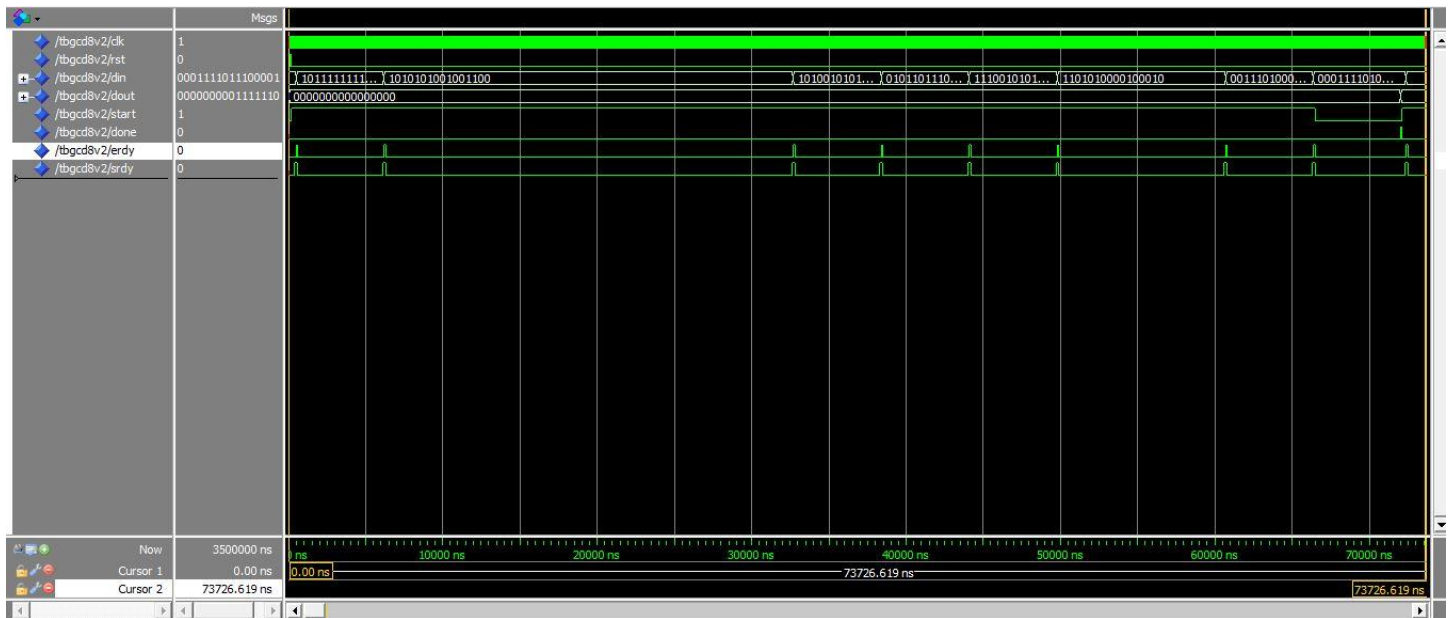
Gate Level Simulation

Transcript

```
Transcript
# Loading maxv.maxv_asyncn_lcell(vital_ie)
# Loading maxv.maxv_lcell_register(vital_ie_reg)
# Loading instances from GCD8_vhd.sdo
# Loading timing data from GCD8_vhd.sdo
# ** Warning: Design size of 34729 statements exceeds ModelSim Altera Starter recommended capacity.
# Expect performance to be adversely affected.
# ** Note: (vsim-3587) SDF Backannotation Successfully Completed.
# Time: 0 ps Iteration: 0 Instance: /tbgcd8v2 File: D:/GCD/gcd8/./tbgcd8v2.vhd
#
# add wave *
# view structure
# .main_pane.structure.interior.cs.body.struct
# view signals
# .main_pane.objects.interior.cs.body.tree
# run 3500 us
# ** Error: Total Checked 50 failures 0
# Time: 3361150 ns Iteration: 1 Instance: /tbgcd8v2
VSIM 2>
```

Total 50 inputs

Waveform



Source Files

Divider

1. divider.vhd
2. dividerComponents.vhd
3. cpDivider.vhd : Control path of divider
4. dpDivider.vhd : Data path of divider
5. tbDivider.vhd : Test bench for divider
6. tbDivider.txt : Trace file for divider test bench
7. tbDivider.py : Python file to create trace file for test bench of divider

GCD2

1. GCD2.vhd
2. cpGCD2.vhd : Control path of GCD2
3. dpGCD2.vhd : Data path of GCD2
4. tbGCD2.vhd : Test bench for GCD2
5. tbGCD2.txt : Trace for test bench of GCD2
6. gcd2.py : Python file to create trace file test bench

GCD8

1. GCD8.vhd
2. cpGCD8.vhd : Control path of GCD8
3. dpGCD8.vhd : Data path of GCD8
4. tbGCD8.vhd : Test bench for GCD8
5. tbGCD8.txt : Trace for test bench of GCD8
6. gcd8.py : Python file to create trace file test bench