CS2610: Computer Organization and Architecture

Lab 4: Report

Devansh Singh Rathore(111701011), Himanshu Jain(111701013)

Objective

To implement exponent equalization procedure of single precision floating point numbers in verilog.

Problem

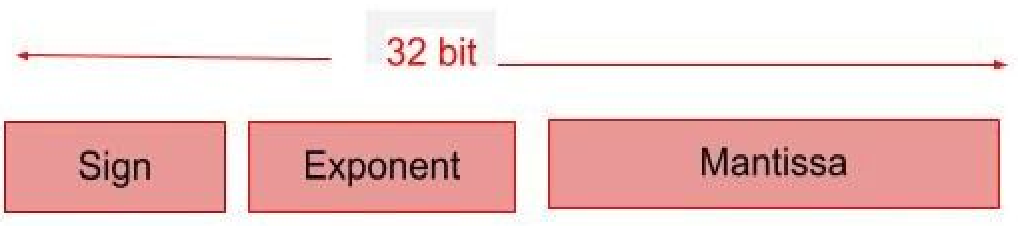
You are given two numbers A and B in IEEE 754 format.In Verilog, implement exponent equalization procedure and perform addition of two given numbers.

Design

IEEE-754 format breaks up a floating point number x into three distinctive parts:

•sign bit sx(sx=0 for positive numbers, sx=1 for negative)
•biased exponent ex(8 bits in the 32 bit format, 11 bits in the 64 bit format, offset by 127 or 1023, respectively)
•a non-negative mantissa mx, which stores the most significant bits of the number. The number is then given by x = (-1) $^{s_x} m_x 2^{e_x}$

Adding (and, subtracting) two floating point numbers:
1.) Make sure the (biased) exponents are the same. If they're not the same, adapt the smaller to the larger. This involves right-shifting the bits in the full mantissa of the number with the smaller exponent.
2.) Add the (full) mantissas by converting them into two's complement format, taking the sign bit into account.
3.) Format result, taking into account overflow (may need to increment exponent) or mantissa starting with 0's (shift result left and decrement exponent accordingly)



Single Precision IEEE 754 Floating Point Standard

Implementation

Code for the Verilog program.

```
module main(input [31:0] a,b,output reg [31:0] out);

    wire c;
    reg [23:0] a1;
    reg [23:0] b1;
    reg [24:0] c1;
    wire [7:0] big,smal;
    wire [7:0] diff

    comp C1 (a[30:23] ,b[30:23], big,smal,c);
    subtractor S1(big,smal, diff);

    initial
     begin
        a1[23]=0;
        b1[23]=0;
        c1[24] = 0;
    end

    always@*
    begin
        if(c==1'b0)
        begin
           out[31] = a[31];
           b1[22:0] = b[22:0] >> 1;
           b1[22] = 1;
           b1[22:0] = b1[22:0]>>(diff-1);
           if(b[31]^a[31]==1)
           begin
            a1 = a[22:0]-b1[22:0];
           end
           else
           begin
            a1 = a[22:0] + b1[22:0];
           end
           if(a1[23]==1)
           begin
            out[22:0] = a1[23:1];
            out[30:23] = big+1;
```

```
   end
   else
   begin
    out[22:0] = a1[22:0];
    out[30:23] = big;
   end
end
else
begin
if(diff==0)
begin
 out[31] = b[31];
 a1[23] = 1;
 b1[23] = 1;
 if(a[31]^b[31]==1)
 begin
  if(a[22:0]>b[22:0])
  begin
   a1[23] = 0;
   a1[22:0] = a[22:0]-b[22:0];
   out[30:23] = big;
   while(a1[23]!=1)
   begin
    a1 = a1<<1;
    out[30:23] = out[30:23]-1;
   end
   out[22:0] = a1[22:0];
  end
  else
  begin
   a1[23] = 0;
   a1[22:0] = b[22:0]-a[22:0];
   out[30:23] = big;
   while(a1[23]!=1)
   begin
    a1 = a1<<1;
    out[30:23] = out[30:23]-1;
   end
   out[22:0] = a1[22:0];
  end
 end
 else
 begin
  a1[22:0] = a[22:0];
  b1[22:0] = b[22:0];
  c1 = a1+b1;
```

```
          out[30:23] = big+1;
          out[22:0] = c1[24:2];
         end
        end
        else
        begin
           out[31] = b[31];
           a1 = a[22:0] >> 1;
           a1[22] = 1;
           a1 = a1[22:0]>>diff-1;
           if(a[1]^b[1]==1)
           begin
            b1 = a1[22:0]-b[22:0];
           end
           else
           begin
            b1 = a1[22:0] + b[22:0];
           end
           if(b1[23]==1)
           begin
            out[22:0] = b1[23:1];
           end
           else
           begin
            out[22:0] = b1[22:0];
           end
        end
      end
    end

endmodule
```

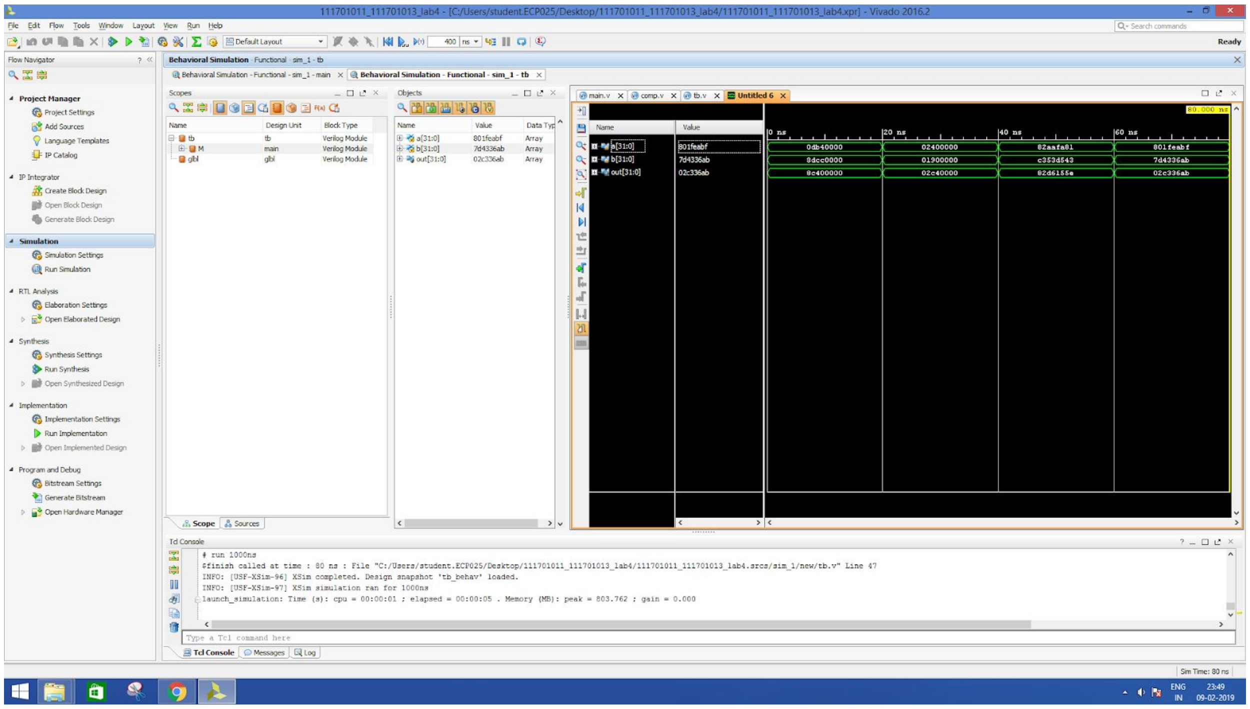
Test Bench

```
module tb();

  reg [31:0] a, b;
  wire [31:0] out;

  main M(
    .a(a),
    .b(b),
      .out(out)
    );

    initial begin
      a = 32'b00001101101101000000000000000000;
      b = 32'b10001101110011000000000000000000;
      #20
      a = 32'b00000010010000000000000000000000;
      b = 32'b00000001100100000000000000000000;
      #20
      a = 32'b110000010101010101111101010000001;
      b = 32'b111000011010100111101010101000011;
      #20
      a = 32'b110000000000111111110101010111111;
      b = 32'b001111101010000110011011010101011;
      #20
      $finish;
    end

  endmodule
```

Experiments

This section should contain the following details for every circuit.

1. Simulation snapshots:



111701011_111701013_lab4 - [C:/Users/student.ECP025/Desktop/111701011_111701013_lab4/111701011_111701013_lab4.xpr] - Vivado 2016.2

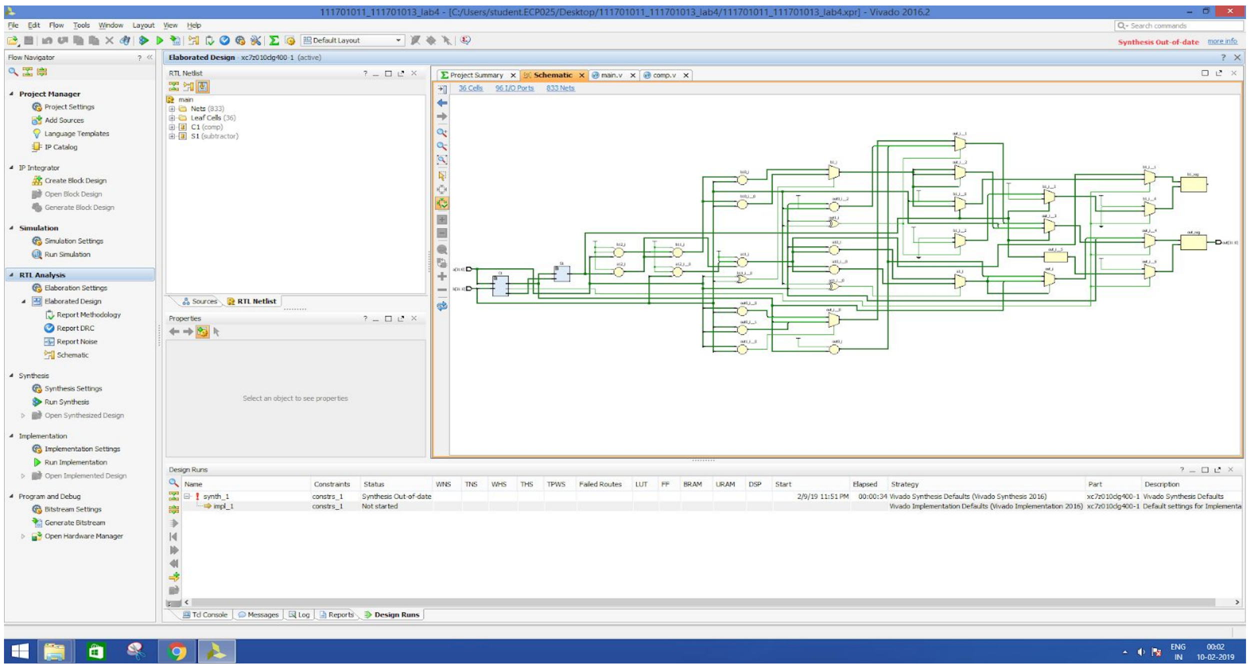
Name	Value	0 ns	20 ns	40 ns	60 ns
a[31:0]	801feabf	0db40000	02400000	82aafa81	801feabf
b[31:0]	7d4336ab	8dcc0000	01900000	c353d543	7d4336ab
out[31:0]	02c336ab	8c400000	02c40000	82d6155e	02c336ab

Tcl Console

```
# run 1000ns
$finish called at time : 80 ns : File "C:/Users/student.ECP025/Desktop/111701011_111701013_lab4/111701011_111701013_lab4.srcs/sim_1/new/tb.v" Line 47
INFO: [USF-XSim-96] XSim completed. Design snapshot 'tb_behav' loaded.
INFO: [USF-XSim-97] XSim simulation ran for 1000ns
launch_simulation: Time (s): cpu = 00:00:01 ; elapsed = 00:00:05 . Memory (MB): peak = 803.762 ; gain = 0.000
```

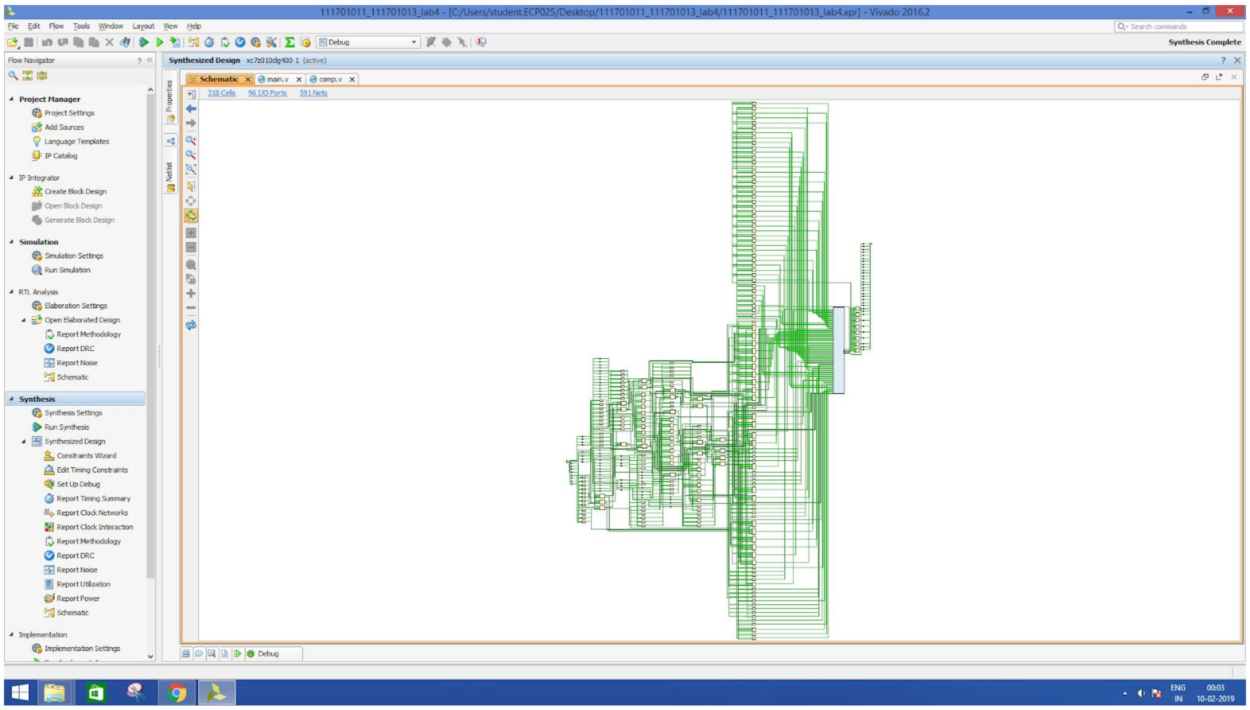
Sim Time: 80 ns

2. Schematic diagram:

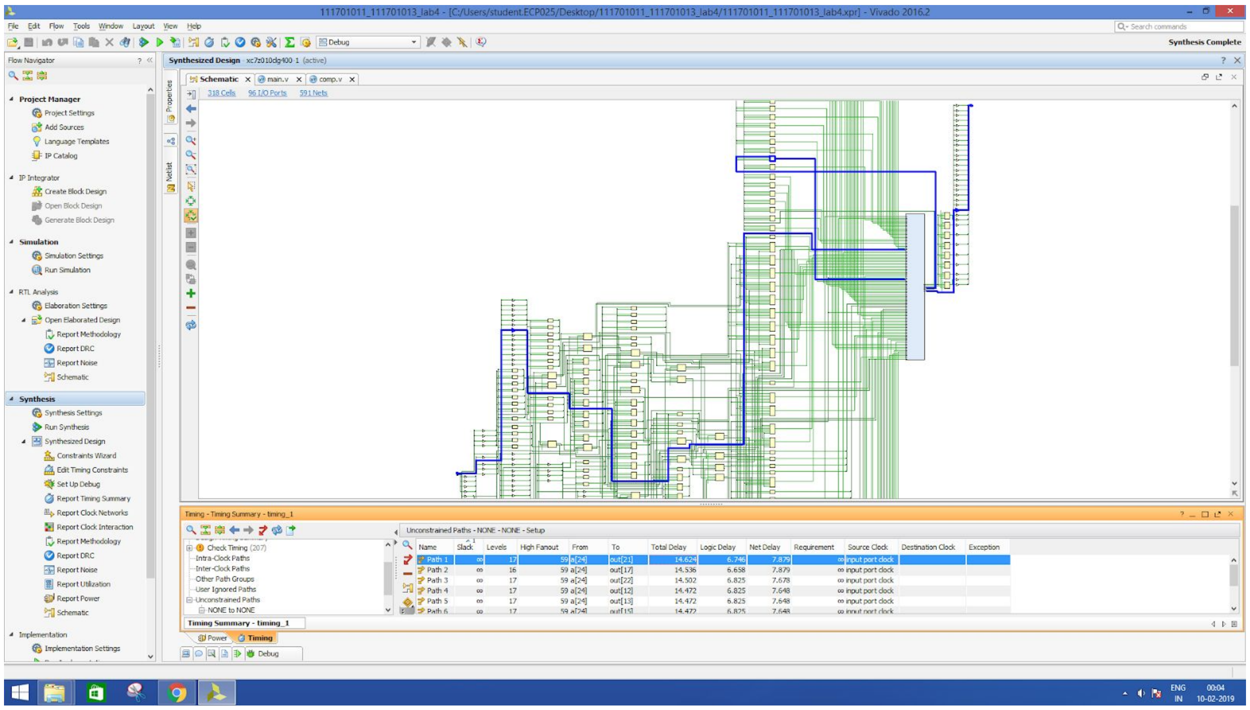

111701011_111701013_lab4 - [C:/Users/student.ECP025/Desktop/111701011_111701013_lab4/111701011_111701013_lab4.xpr] - Vivado 2016.2
Flow Navigator
Project Manager
RTL Analysis
Synthesis
Implementation
Program and Debug
Elaborated Design - xc7z010clg400-1 (active)
RTL Netlist
Schematic
36 Cells
96 I/O Ports
833 Nets
Synthesis Out-of-date
Select an object to see properties

Design Runs

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Strategy	Part	Description
synth_1	constrs_1	Synthesis Out-of-date												2/9/19 11:51 PM	00:00:34	Vivado Synthesis Defaults (Vivado Synthesis 2016)	xc7z010clg400-1	Vivado Synthesis Defaults
impl_1	constrs_1	Not started														Vivado Implementation Defaults (Vivado Implementation 2016)	xc7z010clg400-1	Default settings for Implementa


111701011_111701013_lab4 - [C:/Users/student.ECP025/Desktop/111701011_111701013_lab4/111701011_111701013_lab4.xpr] - Vivado 2016.2
Synthesis Complete
Synthesized Design - xc7z010clg400-1 (active)
Schematic
318 Cells
96 I/O Ports
591 Nets

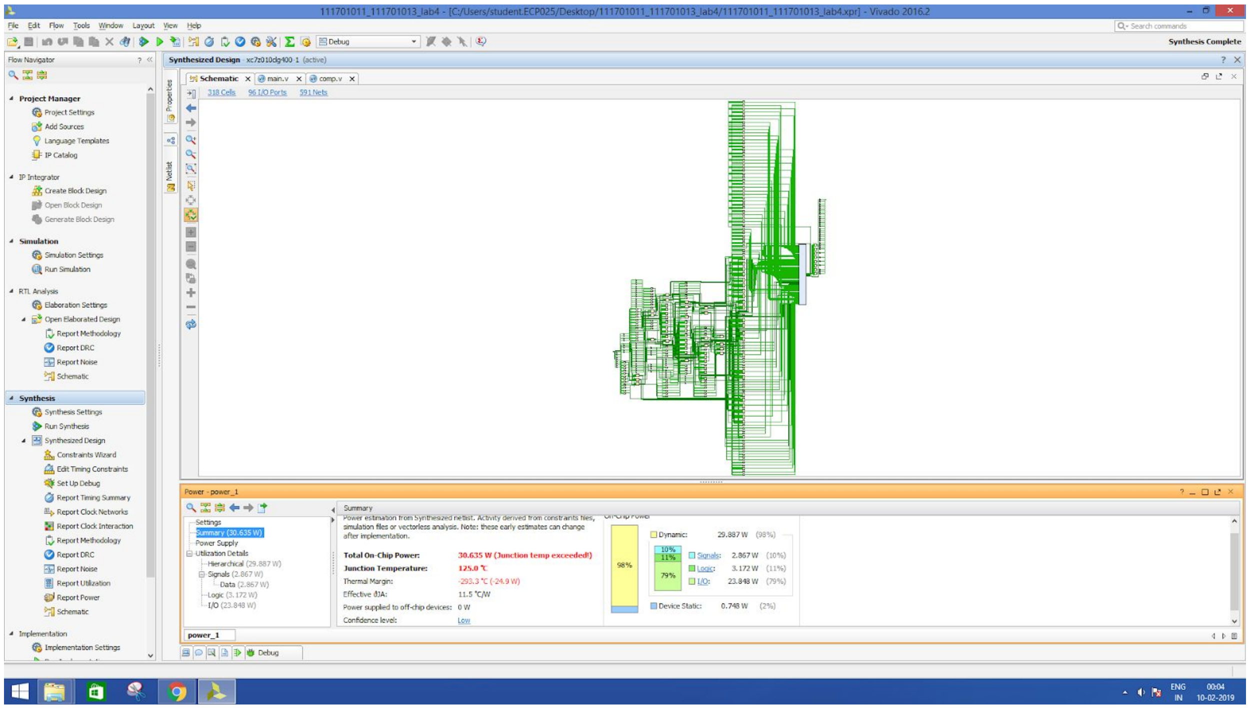
3. Synthesis report.



Unconstrained Paths - NONE - NONE - Setup

Name	Slack	Levels	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source Clock	Destination Clock	Exception
Path 1	∞	17	59	a[24]	out[21]	14.624	6.746	7.879	∞	input port clock		
Path 2	∞	16	59	a[24]	out[17]	14.536	6.658	7.879	∞	input port clock		
Path 3	∞	17	59	a[24]	out[22]	14.502	6.825	7.678	∞	input port clock		
Path 4	∞	17	59	a[24]	out[12]	14.472	6.825	7.648	∞	input port clock		
Path 5	∞	17	59	a[24]	out[13]	14.472	6.825	7.648	∞	input port clock		
Path 6	∞	17	59	a[24]	out[15]	14.472	6.825	7.648	∞	input port clock		

Timing Diagram



Summary

Power estimation from Synthesized netlist. Activity derived from constraints files, simulation files or vectorless analysis. Note: these early estimates can change after implementation.

Total On-Chip Power:	**30.635 W (Junction temp exceeded!)**
Junction Temperature:	**125.0 °C**
Thermal Margin:	-293.3 °C (-24.9 W)
Effective ϑJA:	11.5 °C/W
Power supplied to off-chip devices:	0 W
Confidence level:	Low

Dynamic:	29.887 W	(98%)
Signals:	2.867 W	(10%)
Logic:	3.172 W	(11%)
I/O:	23.848 W	(79%)
Device Static:	0.748 W	(2%)

Power Results

Observations

Dynamic Power used : 29.887W
Static Power used : 0.748W
Total On Chip Power used : 30.635W(Junction Temp. exceeded)
Static Power is 2% of the total power while dynamic power is 98%.
In dynamic power,
79% is used for I/O operations = 23.848W
11% is used in logic = 3.172W
10% is used in signals = 2.867W
Junction Temperature was 125.0 ° C
Total Delay = 14.624

Key observations in trends of power consumption and timings:

1. Power consumption is higher as compared to general implementations because of use of more complex component structure. Moreover, there was a increase in parallel component in circuit (as can be seen from dynamic(98%) to static(2%) ratio), which again causes rise in power consumption.
2. Junction temperature was very high (exceeded the limit), because of increase in power consumption.
3. One of the reason could be because of use of large(32) bit registers in equations and carrying out its complex calculations in a nested conditional structure, the junction temperature as well as the power consumption increased by a large margin.
4. Total delay was much higher than total delays in general cases, due to use of heavy complex circuit design.
5. Dynamic power consumption increased due to increase in number of transitions in the logic, Input/Output in the circuit.
6. Inconsistencies: Junction Temperature exceeded the limit, which might be caused due to use of extensive I/O, transitions in logical circuit.

Conclusions

We analyzed exponent equalization and addition arithmetic procedure of single precision floating point numbers, which took high power consumption (30.635W), with junction temperature exceeded, and total time delay of 14.624.

References

[1] https://www.ias.ac.in/article/fulltext/reso/021/01/0011-0030