DCS Lab4

Sequential Circuit

Sequential Circuit

- Different from combinational circuit, sequential circuit will synthesize registers which can store previous data.

O

```
always @(posedge clk or negedge rst_n) begin
    if(!rst_n)
        a <= 0;
    else
        a <= a + 1;
end
```

← Sequential circuit

X

```
assign a = a + 1;

always @(*) begin
  a = a + 1;
end

always @(*) begin
  if(a > b) a = b;
  else a = a;
end
```

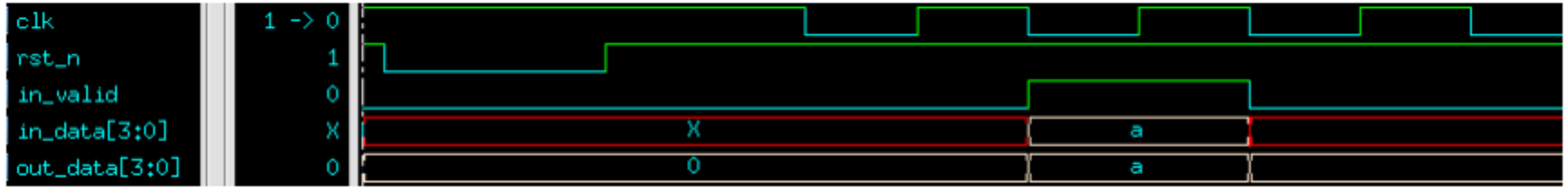
← Combinational circuit

Waveform Difference

- Combinational Circuit

```
always@(*)begin
    if(in_valid) out_data = in_data;
    else out_data = 0;
end

assign out_data = (in_valid) ? in_data : 0;
```



Waveform Difference

- Sequential Circuit

```
always@(posedge clk or negedge rst_n)begin
    if(!rst_n)
        out_data <= 0;
    else begin
        if(in_valid) out_data <= in_data;
        else out_data <= 0;
    end
end
```

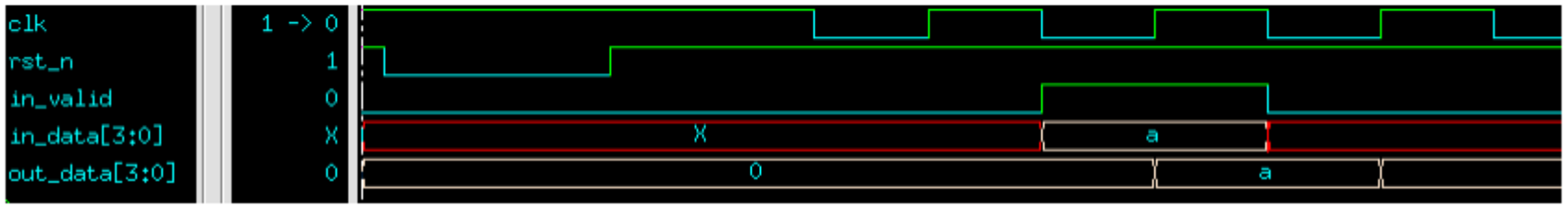
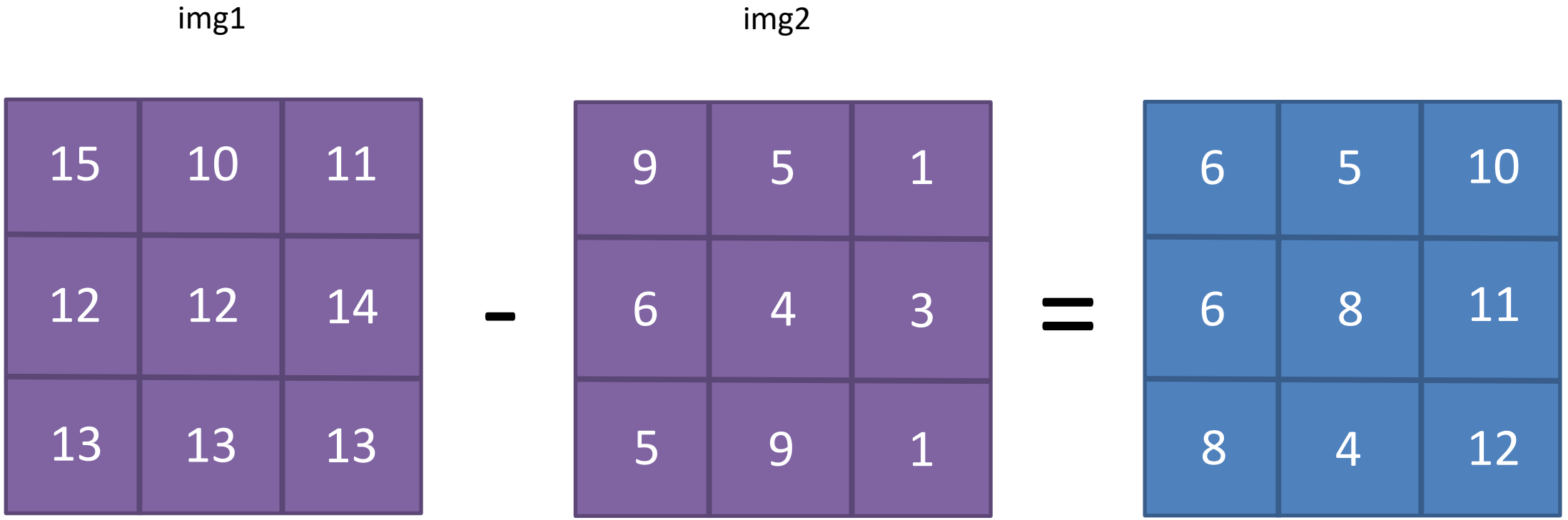


Image Subtract

- Residual Image
- Show objects in the image (image – background)

Image Subtract

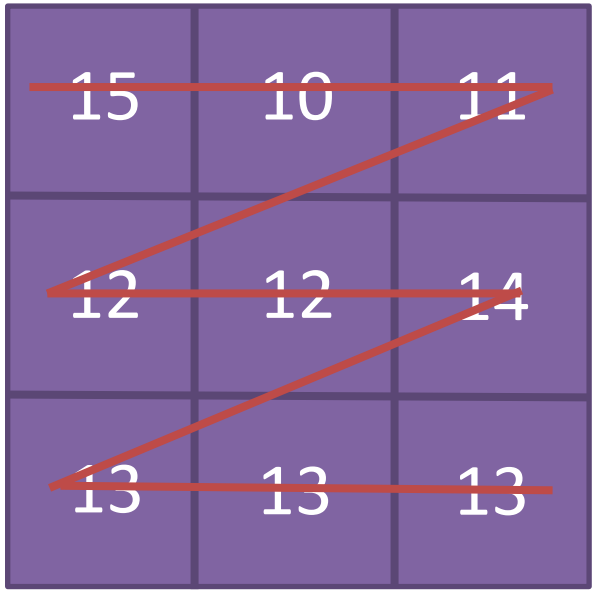


Seq.sv

Input Signal	Bit Width	Definition
clk	1	Clock
rst_n	1	Asynchronous active-low reset
in_valid	1	High when input is valid
in_image	4	Pixel values of two images (Raster Scan Order)

Output Signal	Bit Width	Definition
out_valid	1	High when output is valid
out_diff	4	Pixel values of difference => img1 – img2 (Raster Scan Order)

Raster Scan Order



Spec of 01_RTL

- for_loop is forbidden
- All output should be reset after the reset signal is asserted
- The **out_valid** should not be raised when **in_valid** is high.
- When out_valid is low, all output should be reset to 0
- The out_valid should be high within 100 cycles after in_valid pull low
- Cannot name your registers or wires with those words, such as *error*, *congratulation*, *latch* or *fail*, which will cause demo fail

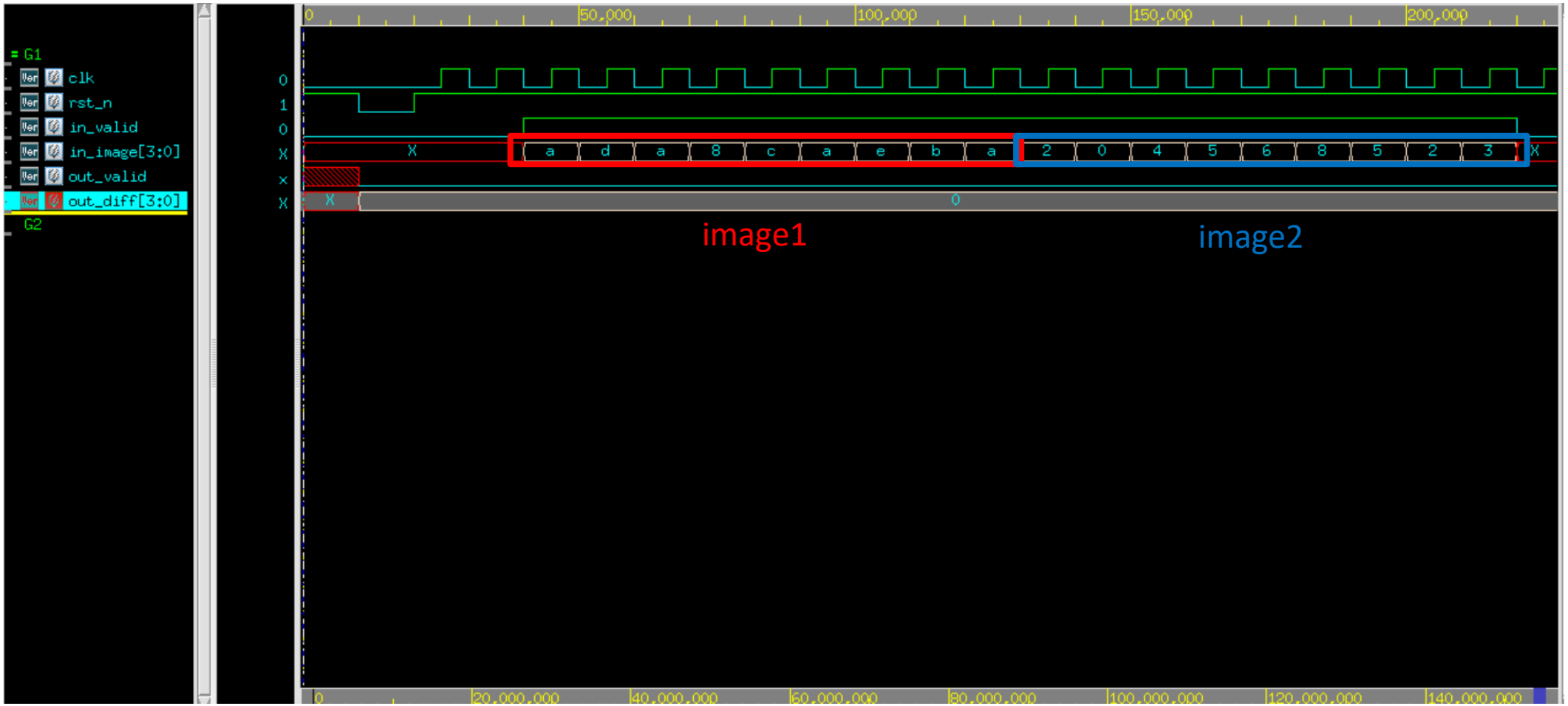
Spec of 02_SYN & 03_GATE

- The synthesis result (syn.log) cannot include any error and latch
- The slack in the end of timing report should be non-negative and MET
- 03_GATE simulation need pass and without timing violation warning

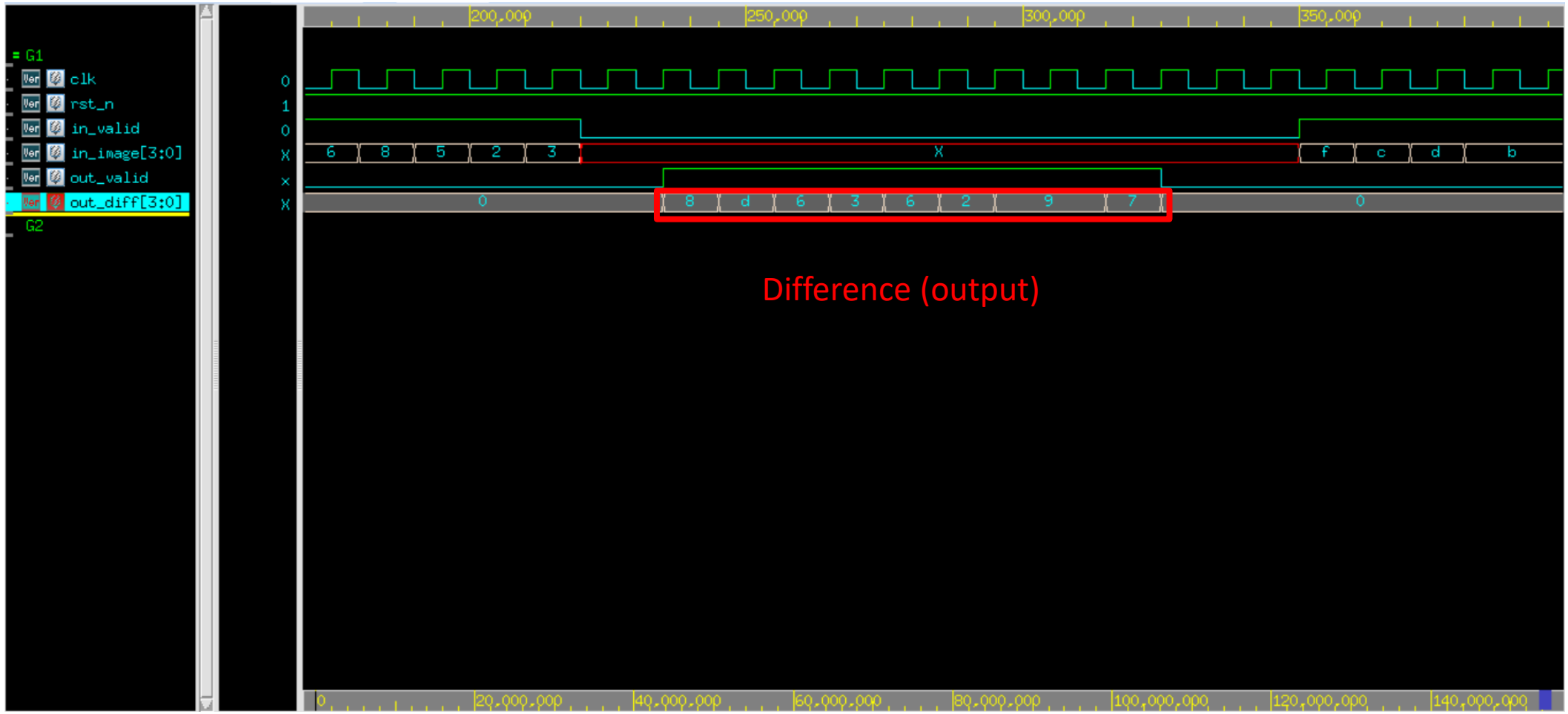
Waveform


G1
clk
rst_n
in_valid
in_image[3:0]
out_valid
out_diff[3:0]
G2
0
1
0
X
x
X
0
50,000
100,000
150,000
200,000
X
a
d
a
8
c
a
e
b
a
2
0
4
5
6
8
5
2
3
X
All outputs are reset
0
20,000,000
40,000,000
60,000,000
80,000,000
100,000,000
120,000,000
140,000,000

Waveform


G1
clk
rst_n
in_valid
in_image[3:0]
out_valid
out_diff[3:0]
G2
image1
image2

Waveform


G1
clk
rst_n
in_valid
in_image[3:0]
out_valid
out_diff[3:0]
G2
0
1
0
X
x
X
200,000
250,000
300,000
350,000
6
8
5
2
3
X
f
c
d
b
0
8
d
6
3
6
2
9
7
0
Difference (output)
0
20,000,000
40,000,000
60,000,000
80,000,000
100,000,000
120,000,000
140,000,000

Command

- tar -xvf ~dcsta01/Lab04.tar
- 02_SYN need
  - No Latch
  - No error
  - No timing violation (MET)
- 03_GATE
  - Simulation need pass and no timing violation

DEMO1: 3/21 17:30:00

DEMO2: 3/22 23:59:59