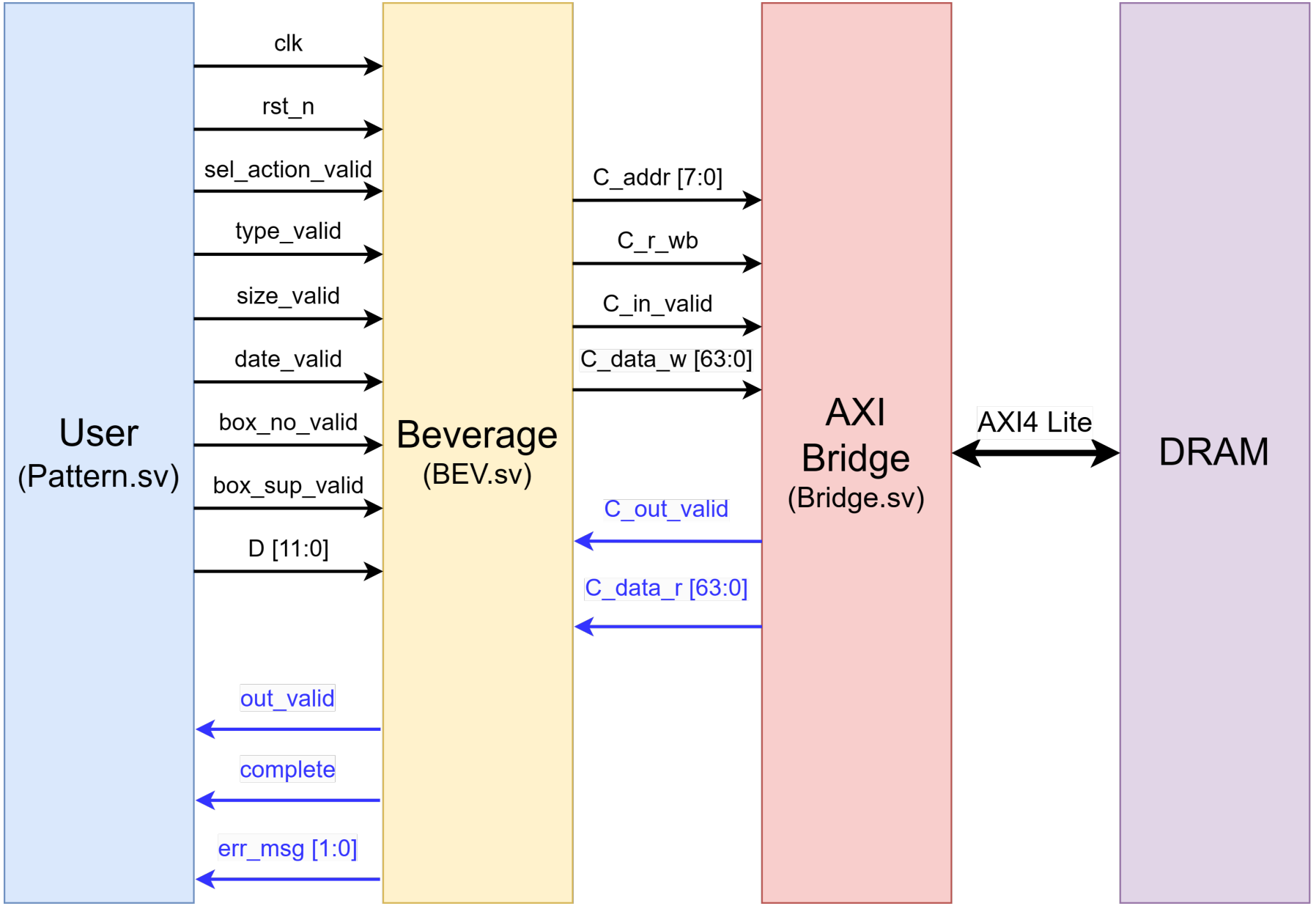
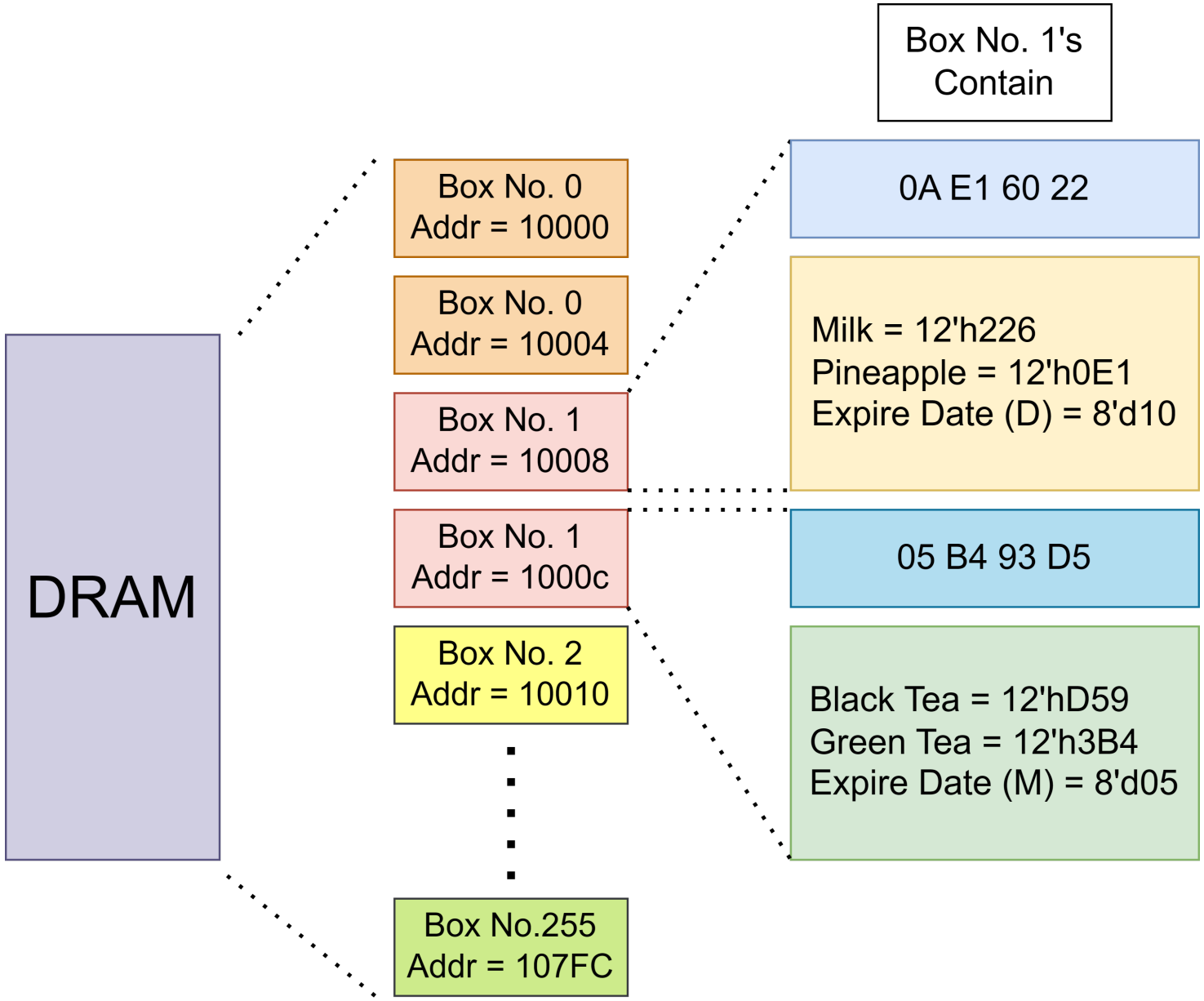
Lab09 Exercise note


User
(Pattern.sv)
clk
rst_n
sel_action_valid
type_valid
size_valid
date_valid
box_no_valid
box_sup_valid
D [11:0]
out_valid
complete
err_msg [1:0]
Beverage
(BEV.sv)
C_addr [7:0]
C_r_wb
C_in_valid
C_data_w [63:0]
C_out_valid
C_data_r [63:0]
AXI
Bridge
(Bridge.sv)
AXI4 Lite
DRAM


DRAM
Box No. 0
Addr = 10000
Box No. 0
Addr = 10004
Box No. 1
Addr = 10008
Box No. 1
Addr = 1000c
Box No. 2
Addr = 10010
Box No.255
Addr = 107FC
Box No. 1's
Contain
0A E1 60 22
Milk = 12'h226
Pineapple = 12'h0E1
Expire Date (D) = 8'd10
05 B4 93 D5
Black Tea = 12'hD59
Green Tea = 12'h3B4
Expire Date (M) = 8'd05

DRAM.dat file example

```
1   @10000
2   05 2A FA F8
3   @10004
4   0A 4D 6B 2F
5   @10008
6   11 19 52 78
7   @1000C
8   0B 2D 58 C2
9   @10010
10  0B 15 7C 35
11  @10014
12  09 71 5C BB
13  @10018
14  1B 49 3C C0
15  @1001C
16  02 03 7B 4D
17  @10020
18  01 FB 84 0E
19  @10024
20  03 42 48 0A
```

@10008

11	19	52	78
Expired Date (Day)	Pineapple Juice	Milk	

@1000C

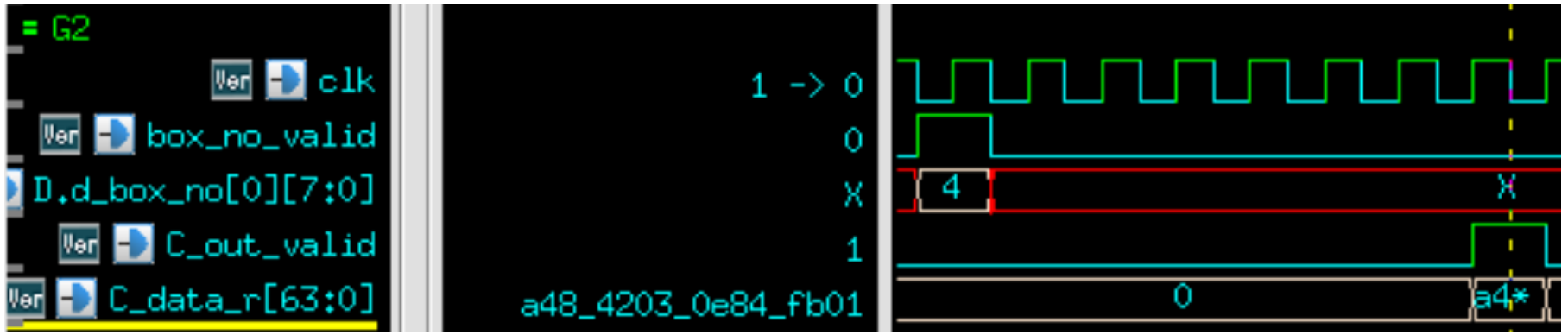
0B	2D	58	C2
Expired Date (Month)	Green Tea	Black Tea	

Category	Amount (Hex.)
Black Tea	C25
Green Tea	82D
Milk	785
Pineapple Juice	219
Expired Date (Month)	0B
Expired Date (Date)	11

DRAM.dat Example (2/2)

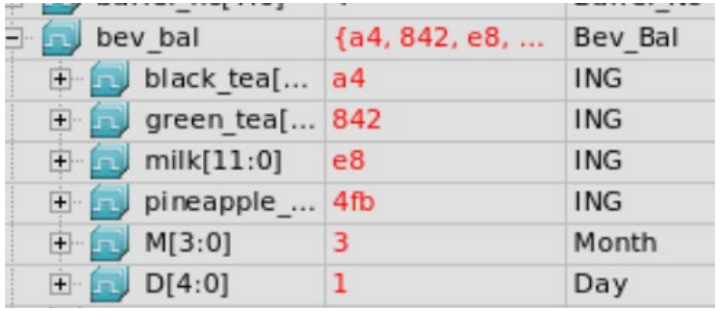
No.0
No.1
No.2
No.3
No.4

```
@10000
05 2A FA F8
@10004
0A 4D 6B 2F
@10008
11 19 52 78
@1000C
0B 2D 58 C2
@10010
0B 15 7C 35
@10014
09 71 5C BB
@10018
1B 49 3C C0
@1001C
02 03 7B 4D
@10020
01 FB 84 0E
@10024
03 42 48 0A
```



Box No.4's Content

Category	Amount (Hex.)
Black Tea	0A4
Green Tea	842
Milk	0E8
Pineapple Juice	4FB
Expired Date (Month)	03
Expired Date (Date)	01



DRAM note

- You may modify the following part in ../00_TESTBED/pseudo_DRAM.sv.

DRAM latency →

```
parameter DRAM_R_latency = 1;
parameter DRAM_W_latency = 1;
parameter DRAM_B_latency = 1;
```

(Will be adjusted to 1~100 during DEMO)

- If you want to initialize dram in pattern, you may use the following code.

Declaration of dram reg array →

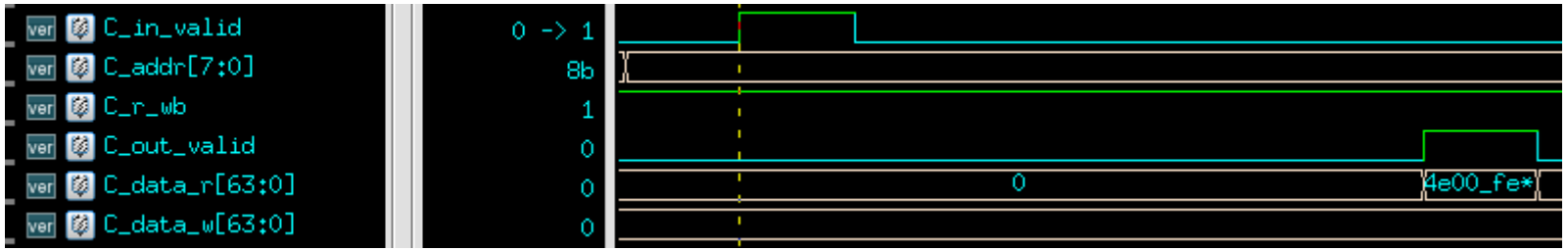
```
parameter DRAM_p_r = "../00_TESTBED/DRAM/dram.dat"

logic [7:0] golden_DRAM[ ((65536+256*8)-1) : (65536+0)] ;

initial $readmemh(DRAM_p_r, golden_DRAM);
```

Bridge

When C_in_valid is high, bridge will check C_r_wb

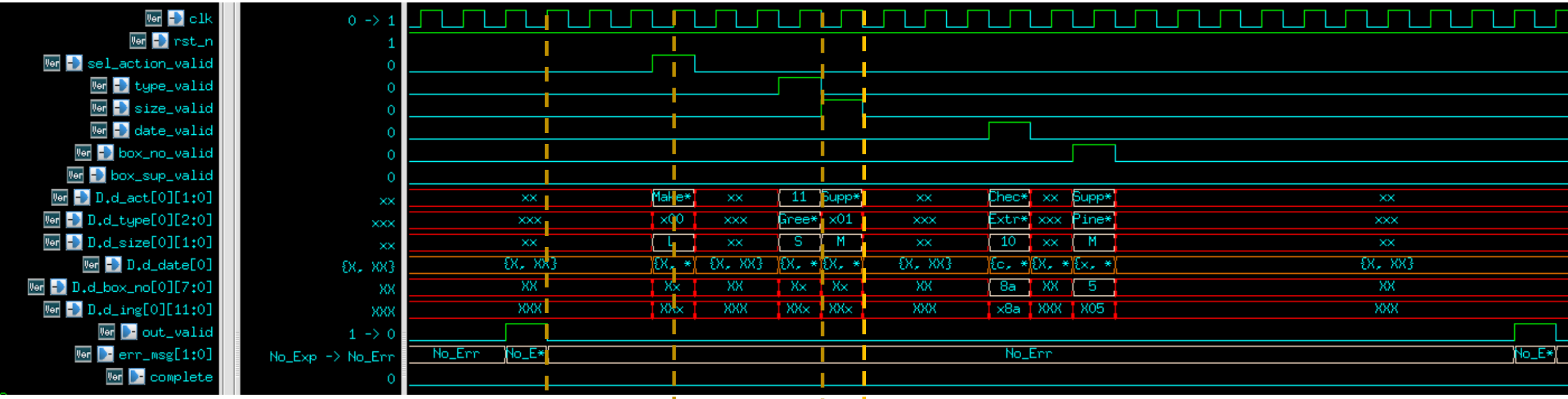


If C_r_wb is 1 (read), then it will base on C_addr to find the corresponding address in dram. When the data from dram is valid, it will pull high C_out_valid and return the value from dram.

If C_r_wb is 0 (write), then it will base on C_addr to find the corresponding address in dram. And then it will write C_data_w to that address. After writing, it will pull high C_out_valid to indicate that the write process is done.

The detail about AXI4 Lite protocol please refer to LAB03 Note.

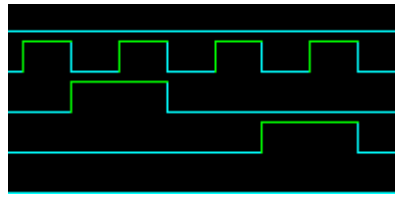
Start of the system



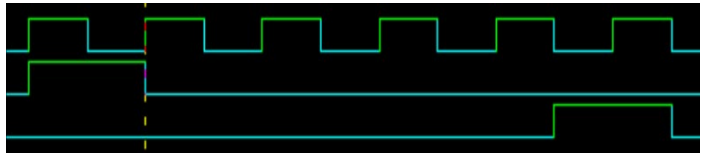
3 cycles

1 cycle

2 cycle



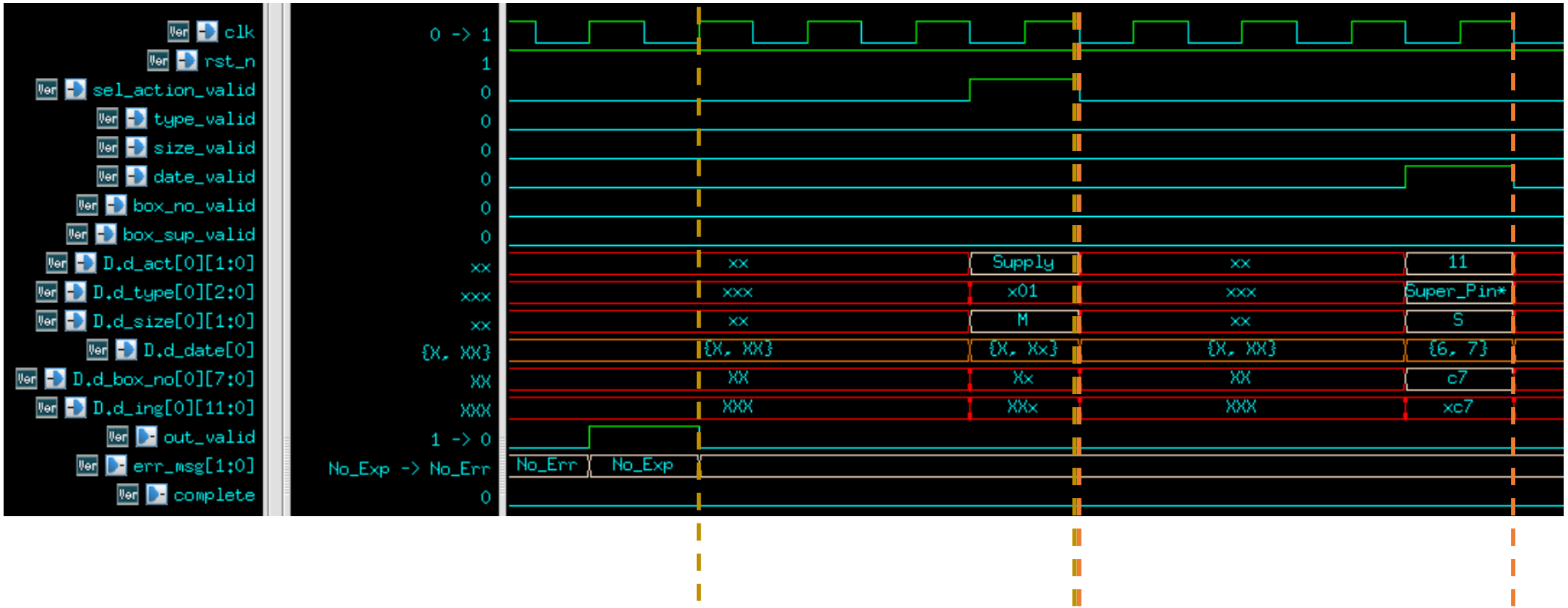
4 cycles



TA will not give the information of the delay between rst_n and the first input valid

1<= Gap <= 4
Order of input valid signal is fixed.
Students can refer to the Lab09 Exercise operations at the page 1.

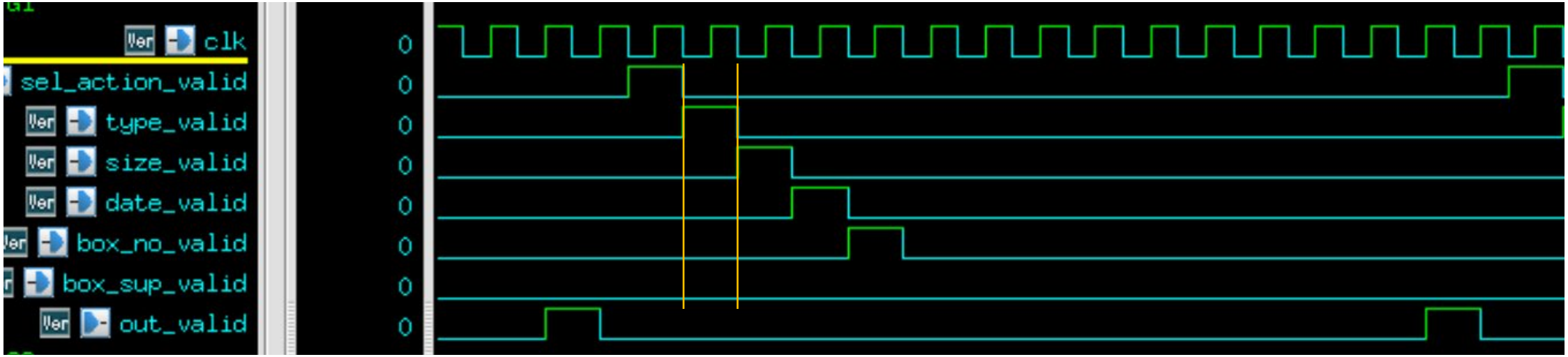
Next operation will be valid **1-4** cycles after out_valid fall.



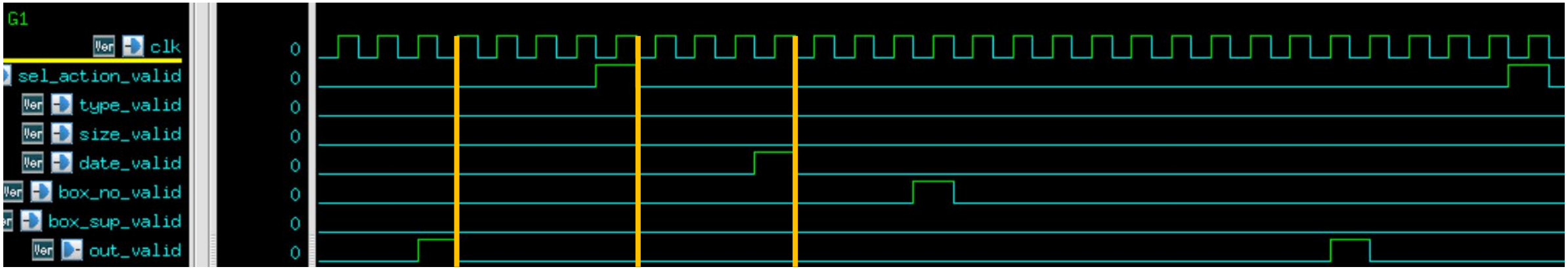
This is the definition of 3 cycles after out_valid fall.

This is the definition of 4 cycles when date_valid = 1 after sel_action_valid = 0

1 cycle between each valid signal



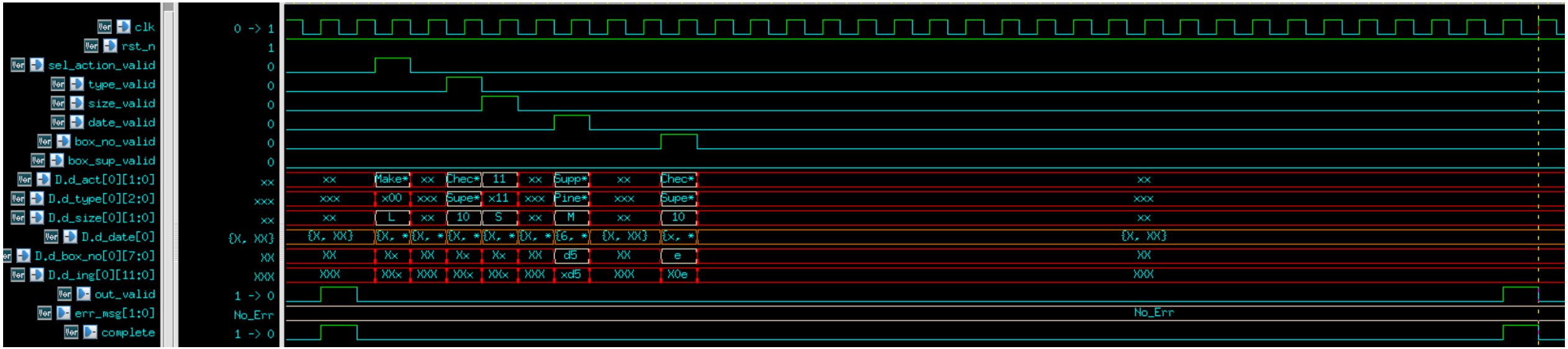
4 cycles between each valid signal



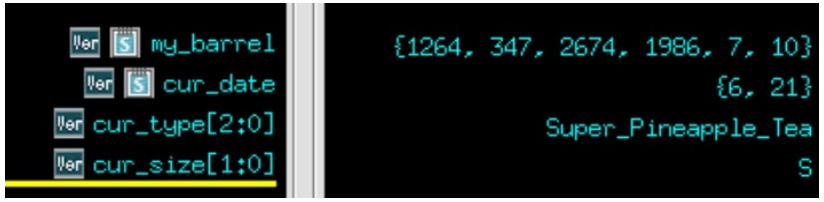
Example

- Case 1 – Make Drink
- Case 2 – Supply
- Case 3 – Check Valid Date

Case 1 – Make Drink



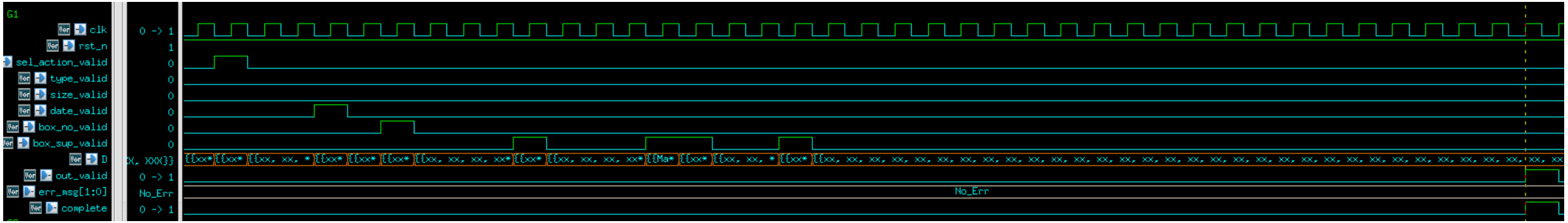
Order in “my_barrel”: Black Tea, Green Tea, Milk, Pineapple, Expired Date (Month), Expire Date (Day)



From the given conditions, it can be determined that the date has not expired, and the ingredient content is sufficient to make the beverage, therefore, there are NO ERRORs this time.
When out_valid=1, set err_msg to “No_Err”, Complete to 1.

my_barrel = Data fetch from Ingredient Box (Black Tea = 1264, Pineapple=1986, Expired Date = 07/10)
cur_date = (Input) Today’s Date = 06/21
cur_type = (Input) Beverage Type = Super Pineapple Tea
cur_size = (Input) Beverage Size = S

Case 2 - Supply



Order: Black Tea, Green Tea, Milk, Pineapple, Expired Date (Month), Expire Date (Day)

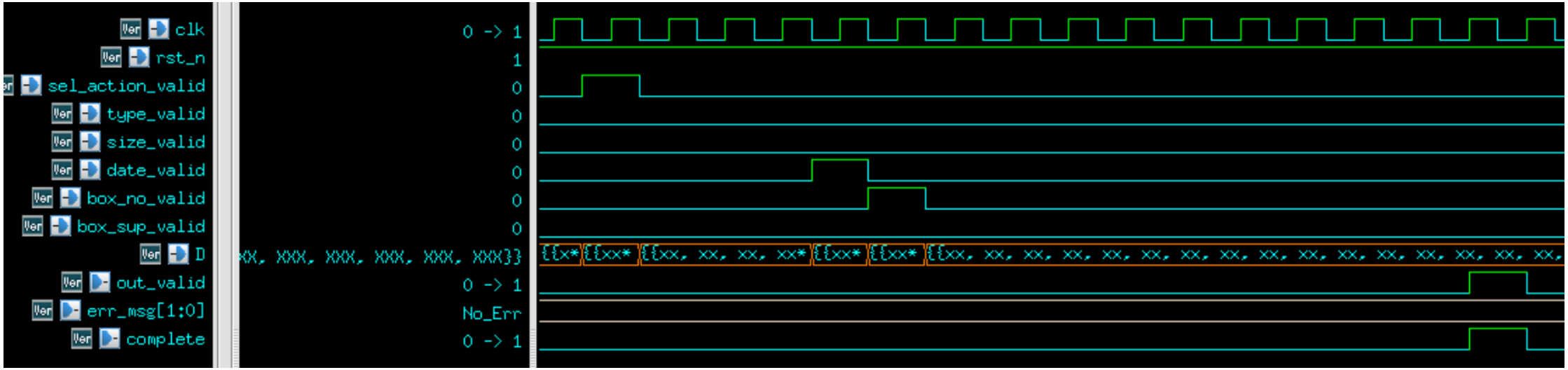
Input: `{87f, ac, 6be, 1d9, 3, 1f}`

Content in Ingredient Box: `{5ee, b4e, 410, 862, 8, 19}`

From the given conditions, it can be determined that the supply ingredient is NOT overflow, therefore, there are NO ERRORs this time.
When out_valid=1, set err_msg to “No_Err”, Complete to 1.

Category	Origin Amount (Hex.)	Supply (Hex.)	Total (Hex.)
Black Tea	5ee	87f	e6d
Green Tea	b4e	ac	bfa
Milk	410	6be	ace
Pineapple Juice	862	1d9	a3b

Case 3 – Check Valid Date



Today's Date: {7, 12}

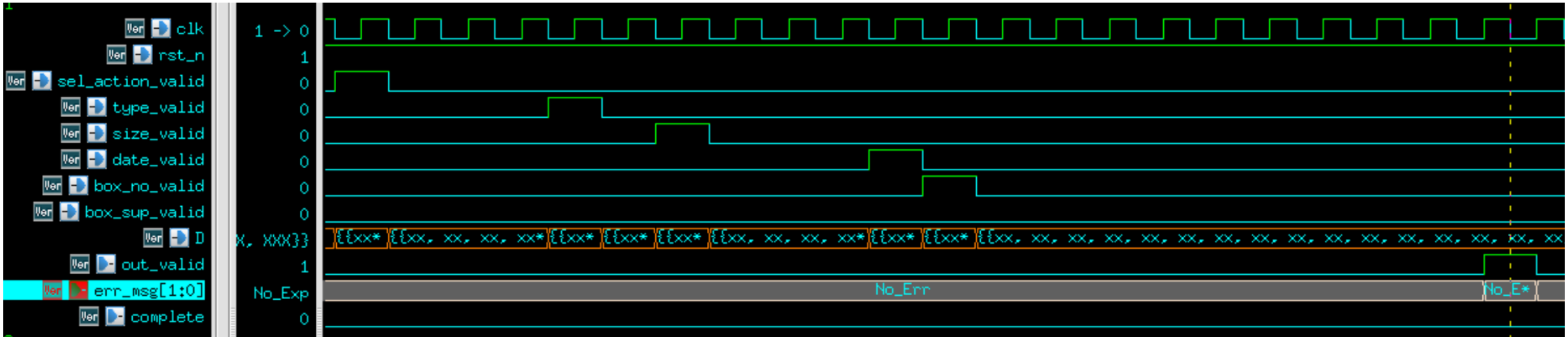
Expired Date of Ingredient Box: , b, 3}

From the given conditions, it can be determined that the expire date is valid.
When out_valid=1, set err_msg to "No_Err", Complete to 1.

Example (Error)

- Case 4 – Make Drink (Expired Data is no longer valid)
- Case 5 – Make Drink (Ingredient not enough)
- Case 6 – Supply (Overflow)
- Case 7 – Check Valid Date (Expired Data is no longer valid)

Case 4 - Make Drink (Expired Data is no longer valid)

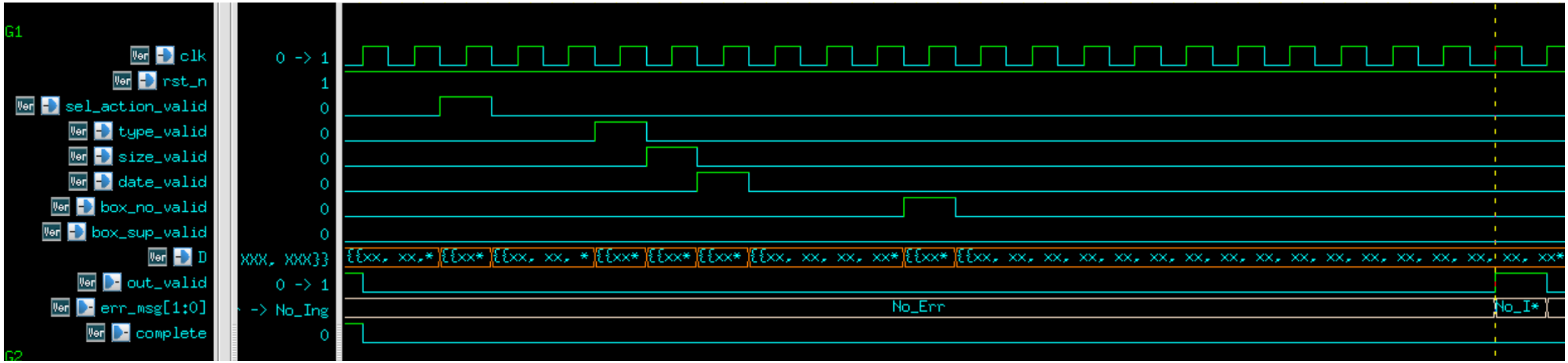


Today's Date: {8, 4}

Expired Date in Ingredient Box: 4, 1c}

From the given conditions, it can be determined that the expire date is not valid.
When out_valid=1, set err_msg to "No_Exp", Complete to 0.

Case 5 - Make Drink (Ingredient not enough)



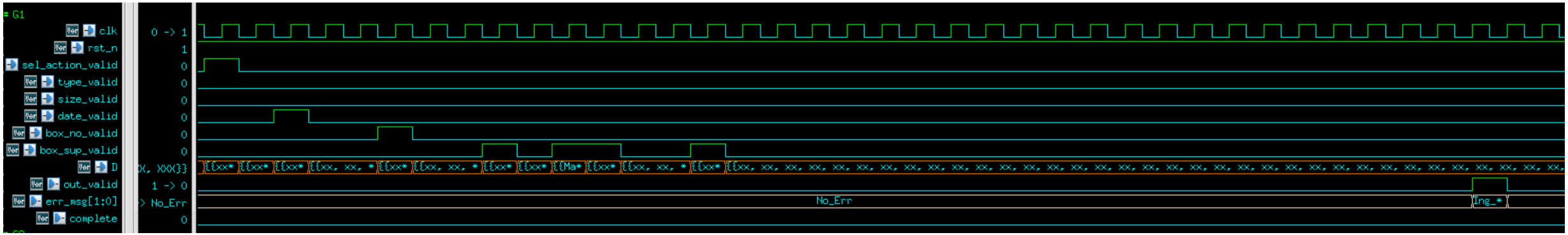
The Large size of Pineapple Juice needs 960 pineapple juice, but pineapple juice only remains 666.

Input: `Pineapple_Juice L`

Ingredient Box: `{f46, 64, d58, 29a, 8, f}`

From the given conditions, it can be determined that the expire date is not valid.
When out_valid=1, set err_msg to “No_Ing”, Complete to 0.

Case 6 – Supply Overflow



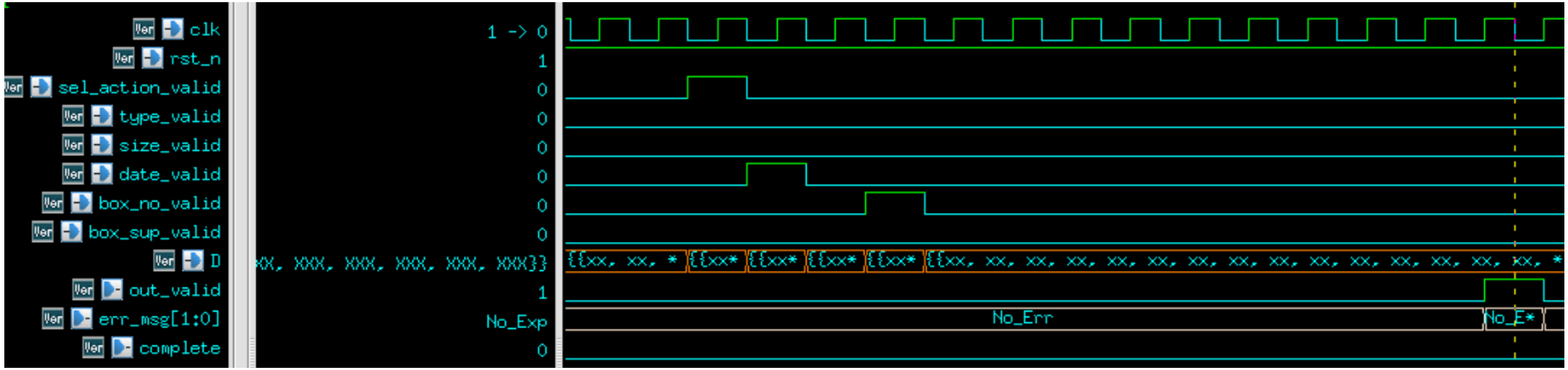
Input: {c5e, 402, 85d, 65a, 5, 9}

Content in Ingredient Box: {fff, 7ed, cac, fff, b, b}

From the given conditions, it can be determined that the supply ingredient is OVERFLOW.
When out_valid=1, set err_msg to “Ing_OF”, Complete to 0.

Category	Origin Amount (Dec.)	Supply (Dec.)	Total (Dec.)
Black Tea	fff	c5e	Overflow!
Green Tea	7ed	402	BEF
Milk	cac	85d	Overflow!
Pineapple Juice	fff	65a	Overflow!

Case 7 – Check Valid Date but No longer Expire



Today's Date: {3, 10}

Expired Date in Ingredient Box: 2, 11}

From the given conditions, it can be determined that the expire date is not valid.
When out_valid=1, set err_msg to "No_Exp", Complete to 0.

Thank YOU