國立清華大學 電機工程系
112 學年度第一學期

SOC Design Laboratory

Lab#1

學校系所:清大電子所

學號:110063553

中文姓名:張傑閔

英文姓名:JIE-MIN, JHANG

- Brief introduction about the overall system

First, put the written c code into vitis_hls, perform simulation, synthesis, and co-simulation, and then export the rtl as IP.

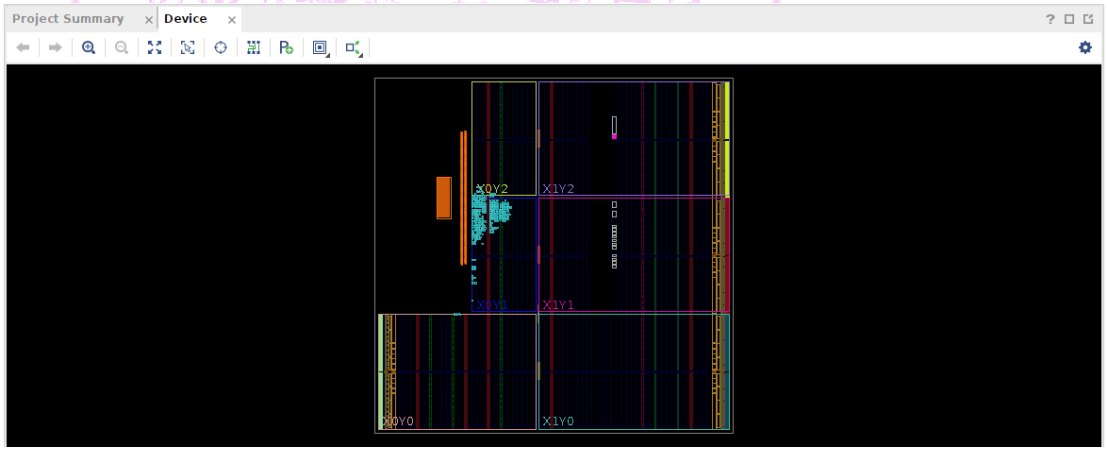
Secondly, go to vivado to introduce the IP and then perform block design. After completion, then use HDL wrapper and finally generate bitstream and feed the bitstream to the FPGA for execution.

- What is observed & learned

I have never learned HLS. This is my first time. I learned that the hardware development process using HLS is to use C language to convert RTL code and send it to FPGA for verification. The purpose is to speed up the hardware development process because traditional ASIC flow needs to go

through multiple verifications, and tape out after the final confirmation that there is no problem.

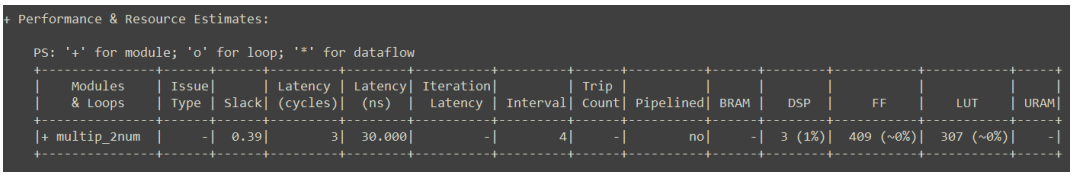
After I executed generate bitstream in vivado, and opened implemented design and observed the following picture. I search the information online and learned that it is a picture of the system automatically configuring (place & route) logic gates on the FPGA.



Screen dump

Performance

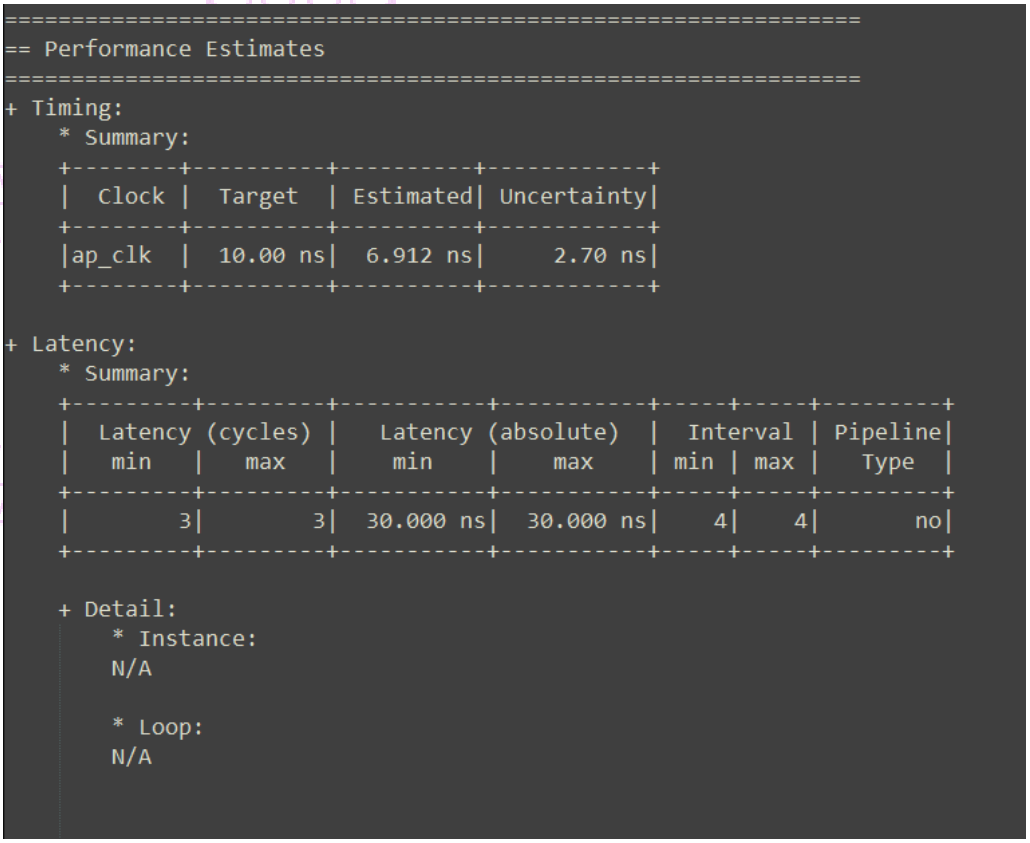
1. Synthesis Summary Report



```
+ Performance & Resource Estimates:

    PS: '+' for module; 'o' for loop; '*' for dataflow
    +--------------+-----+------+--------+-------+---------+---------+------+----------+------+-------+----------+----------+------+
    |    Modules   | Issue|      | Latency | Latency| Iteration|         | Trip |          |      |       |          |          |      |
    |    & Loops   | Type | Slack| (cycles)|  (ns)  |  Latency | Interval| Count| Pipelined| BRAM |   DSP |    FF    |    LUT   | URAM|
    +--------------+-----+------+--------+-------+---------+---------+------+----------+------+-------+----------+----------+------+
    |+ multip_2num |    -|  0.39|       3|  30.000|        -|        4|     -|        no|     -|  3 (1%)|  409 (~0%)|  307 (~0%)|    -|
    +--------------+-----+------+--------+-------+---------+---------+------+----------+------+-------+----------+----------+------+
```

2. Synthesis Detail Report



```
================================================================
== Performance Estimates
================================================================
+ Timing:
    * Summary:
    +--------+----------+----------+------------+
    |  Clock |  Target  | Estimated| Uncertainty|
    +--------+----------+----------+------------+
    |ap_clk  |  10.00 ns|  6.912 ns|     2.70 ns|
    +--------+----------+----------+------------+

+ Latency:
    * Summary:
    +---------+---------+-----------+-----------+-----+-----+---------+
    |  Latency (cycles) |   Latency (absolute)  |  Interval | Pipeline|
    |   min   |   max   |    min    |    max    | min | max |   Type  |
    +---------+---------+-----------+-----------+-----+-----+---------+
    |        3|        3|  30.000 ns|  30.000 ns|    4|    4|       no|
    +---------+---------+-----------+-----------+-----+-----+---------+

    + Detail:
        * Instance:
        N/A

        * Loop:
        N/A
```

Utilization

1. Synthesis Detail Report

```
================================================================
== Utilization Estimates
================================================================
* Summary:
+-----------------+---------+-----+--------+-------+-----+
|       Name      | BRAM_18K| DSP |   FF   |  LUT  | URAM|
+-----------------+---------+-----+--------+-------+-----+
|DSP              |        -|    -|       -|      -|    -|
|Expression       |        -|    -|       -|      -|    -|
|FIFO             |        -|    -|       -|      -|    -|
|Instance         |        0|    3|     309|    282|    -|
|Memory           |        -|    -|       -|      -|    -|
|Multiplexer      |        -|    -|       -|     25|    -|
|Register         |        -|    -|     100|      -|    -|
+-----------------+---------+-----+--------+-------+-----+
|Total            |        0|    3|     409|    307|    0|
+-----------------+---------+-----+--------+-------+-----+
|Available        |      280|  220|  106400|  53200|    0|
+-----------------+---------+-----+--------+-------+-----+
|Utilization (%)  |        0|    1|      ~0|     ~0|    0|
+-----------------+---------+-----+--------+-------+-----+

+ Detail:
    * Instance:
    +----------------------+--------------------+---------+----+-----+-----+-----+
    |        Instance      |       Module       | BRAM_18K| DSP|  FF | LUT | URAM|
    +----------------------+--------------------+---------+----+-----+-----+-----+
    |control_s_axi_U       |control_s_axi       |        0|   0|  144|  232|    0|
    |mul_32s_32s_32_2_1_U1 |mul_32s_32s_32_2_1  |        0|   3|  165|   50|    0|
    +----------------------+--------------------+---------+----+-----+-----+-----+
    |Total                 |                    |        0|   3|  309|  282|    0|
    +----------------------+--------------------+---------+----+-----+-----+-----+

    * DSP:
    N/A

    * Memory:
    N/A
```

* FIFO:
N/A

* Expression:
N/A

* Multiplexer:

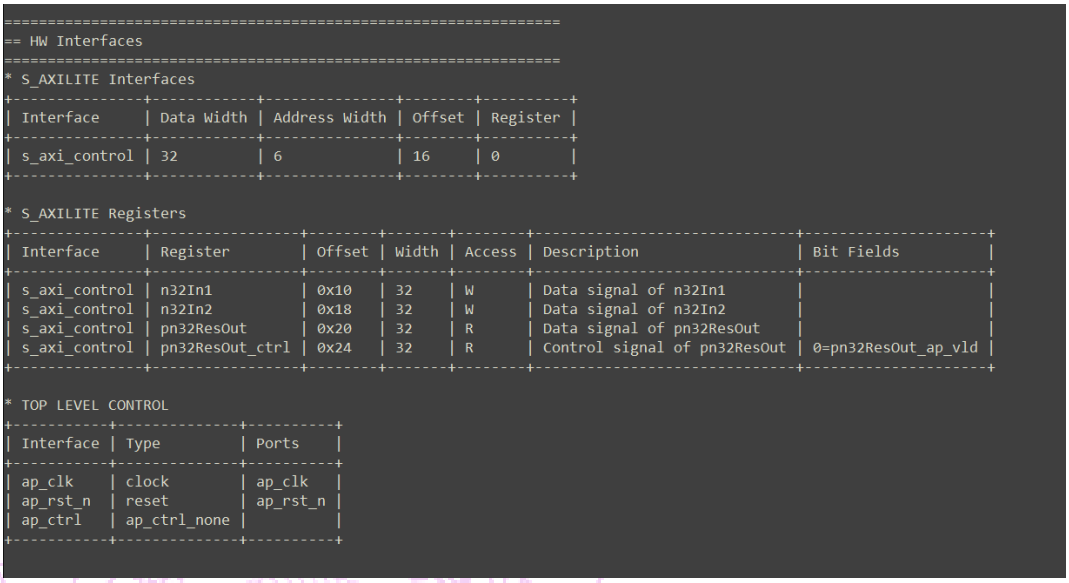
Name	LUT	Input Size	Bits	Total Bits
ap_NS_fsm	25	5	1	5
Total	25	5	1	5

* Register:

Name	FF	LUT	Bits	Const Bits
ap_CS_fsm	4	0	4	0
mul_ln11_reg_71	32	0	32	0
n32In1_read_reg_66	32	0	32	0
n32In2_read_reg_61	32	0	32	0
Total	100	0	100	0

- Interface

1. Synthesis Summary Report

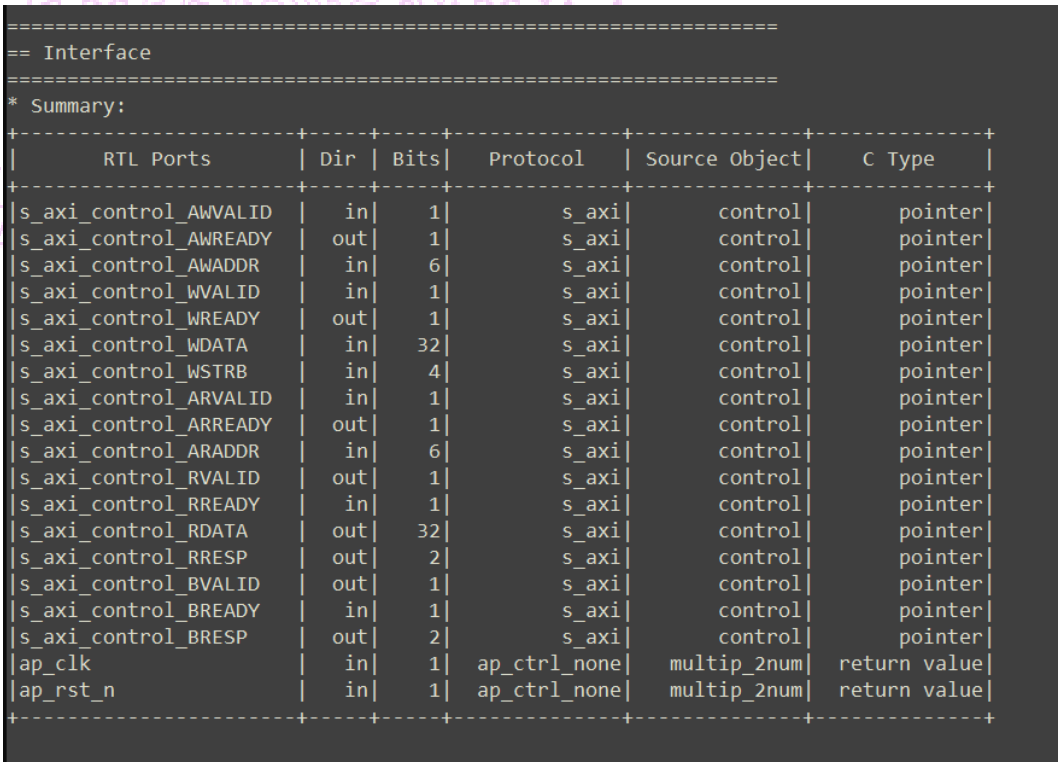


```
================================================================
== HW Interfaces
================================================================
* S_AXILITE Interfaces
+---------------+------------+---------------+--------+----------+
| Interface     | Data Width | Address Width | Offset | Register |
+---------------+------------+---------------+--------+----------+
| s_axi_control | 32         | 6             | 16     | 0        |
+---------------+------------+---------------+--------+----------+

* S_AXILITE Registers
+---------------+-----------------+--------+-------+--------+-----------------------------+---------------------+
| Interface     | Register        | Offset | Width | Access | Description                 | Bit Fields          |
+---------------+-----------------+--------+-------+--------+-----------------------------+---------------------+
| s_axi_control | n32In1          | 0x10   | 32    | W      | Data signal of n32In1       |                     |
| s_axi_control | n32In2          | 0x18   | 32    | W      | Data signal of n32In2       |                     |
| s_axi_control | pn32ResOut      | 0x20   | 32    | R      | Data signal of pn32ResOut   |                     |
| s_axi_control | pn32ResOut_ctrl | 0x24   | 32    | R      | Control signal of pn32ResOut | 0=pn32ResOut_ap_vld |
+---------------+-----------------+--------+-------+--------+-----------------------------+---------------------+

* TOP LEVEL CONTROL
+-----------+--------------+----------+
| Interface | Type         | Ports    |
+-----------+--------------+----------+
| ap_clk    | clock        | ap_clk   |
| ap_rst_n  | reset        | ap_rst_n |
| ap_ctrl   | ap_ctrl_none |          |
+-----------+--------------+----------+
```

2. Synthesis Detail Report



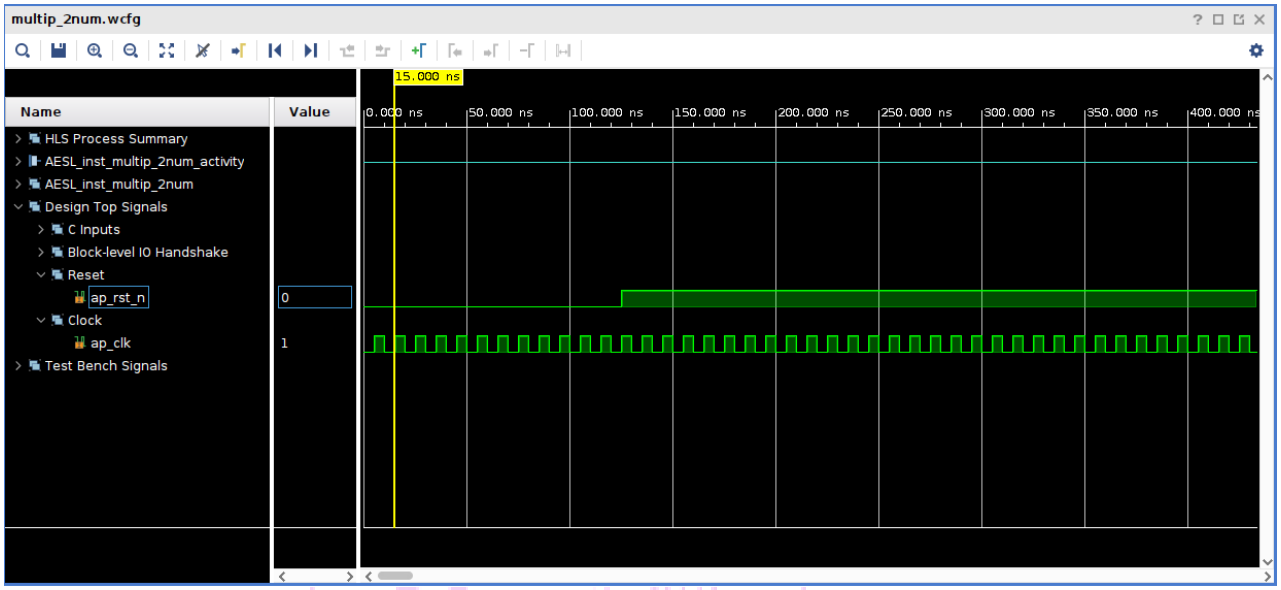
```
================================================================
== Interface
================================================================
* Summary:
+-----------------------+-----+-----+--------------+--------------+--------------+
|       RTL Ports       | Dir | Bits|   Protocol   | Source Object|    C Type    |
+-----------------------+-----+-----+--------------+--------------+--------------+
|s_axi_control_AWVALID  |   in|    1|         s_axi|       control|       pointer|
|s_axi_control_AWREADY  |  out|    1|         s_axi|       control|       pointer|
|s_axi_control_AWADDR   |   in|    6|         s_axi|       control|       pointer|
|s_axi_control_WVALID   |   in|    1|         s_axi|       control|       pointer|
|s_axi_control_WREADY   |  out|    1|         s_axi|       control|       pointer|
|s_axi_control_WDATA    |   in|   32|         s_axi|       control|       pointer|
|s_axi_control_WSTRB    |   in|    4|         s_axi|       control|       pointer|
|s_axi_control_ARVALID  |   in|    1|         s_axi|       control|       pointer|
|s_axi_control_ARREADY  |  out|    1|         s_axi|       control|       pointer|
|s_axi_control_ARADDR   |   in|    6|         s_axi|       control|       pointer|
|s_axi_control_RVALID   |  out|    1|         s_axi|       control|       pointer|
|s_axi_control_RREADY   |   in|    1|         s_axi|       control|       pointer|
|s_axi_control_RDATA    |  out|   32|         s_axi|       control|       pointer|
|s_axi_control_RRESP    |  out|    2|         s_axi|       control|       pointer|
|s_axi_control_BVALID   |  out|    1|         s_axi|       control|       pointer|
|s_axi_control_BREADY   |   in|    1|         s_axi|       control|       pointer|
|s_axi_control_BRESP    |  out|    2|         s_axi|       control|       pointer|
|ap_clk                 |   in|    1|  ap_ctrl_none|   multip_2num|  return value|
|ap_rst_n               |   in|    1|  ap_ctrl_none|   multip_2num|  return value|
+-----------------------+-----+-----+--------------+--------------+--------------+
```

- Co-simulation transcript

```
INFO: [SIM 2] *************** CSIM start ***************
INFO: [SIM 4] CSIM will launch GCC as the compiler.
   Compiling ../../../../hls_Multiplication/Multiplication.cpp in debug mode
   Generating csim.exe
>> Start test!
------------------------
1 * 1 = 1
1 * 2 = 2
1 * 3 = 3
1 * 4 = 4
1 * 5 = 5
1 * 6 = 6
1 * 7 = 7
1 * 8 = 8
1 * 9 = 9
------------------------
2 * 1 = 2
2 * 2 = 4
2 * 3 = 6
2 * 4 = 8
2 * 5 = 10
2 * 6 = 12
2 * 7 = 14
2 * 8 = 16
2 * 9 = 18
------------------------
3 * 1 = 3
3 * 2 = 6
3 * 3 = 9
3 * 4 = 12
3 * 5 = 15
3 * 6 = 18
3 * 7 = 21
3 * 8 = 24
3 * 9 = 27
------------------------
```

```
4 * 1 = 4
4 * 2 = 8
4 * 3 = 12
4 * 4 = 16
4 * 5 = 20
4 * 6 = 24
4 * 7 = 28
4 * 8 = 32
4 * 9 = 36
------------------------
5 * 1 = 5
5 * 2 = 10
5 * 3 = 15
5 * 4 = 20
5 * 5 = 25
5 * 6 = 30
5 * 7 = 35
5 * 8 = 40
5 * 9 = 45
------------------------
6 * 1 = 6
6 * 2 = 12
6 * 3 = 18
6 * 4 = 24
6 * 5 = 30
6 * 6 = 36
6 * 7 = 42
6 * 8 = 48
6 * 9 = 54
------------------------
7 * 1 = 7
7 * 2 = 14
7 * 3 = 21
7 * 4 = 28
7 * 5 = 35
7 * 6 = 42
7 * 7 = 49
7 * 8 = 56
7 * 9 = 63
------------------------
8 * 1 = 8
8 * 2 = 16
8 * 3 = 24
8 * 4 = 32
8 * 5 = 40
8 * 6 = 48
8 * 7 = 56
8 * 8 = 64
8 * 9 = 72
------------------------
9 * 1 = 9
9 * 2 = 18
9 * 3 = 27
9 * 4 = 36
9 * 5 = 45
9 * 6 = 54
9 * 7 = 63
9 * 8 = 72
9 * 9 = 81
------------------------
>> Test passed!
------------------------
INFO: [SIM 1] CSim done with 0 errors.
INFO: [SIM 3] *************** CSIM finish ***************
```

- Co-simulation waveform



- Jupyter Notebook execution results

In [1]:

```
# coding: utf-8

# In[ ]:


from __future__ import print_function

import sys, os

sys.path.append('/home/xilinx')
os.environ['XILINX_XRT'] = '/usr'
from pynq import Overlay

if __name__ == "__main__":
    print("Entry:", sys.argv[0])
    print("System argument(s):", len(sys.argv))

    print("Start of \"" + sys.argv[0] + "\"")

    ol = Overlay("/home/xilinx/jupyter_notebooks/Multip2Num.bit")
    regIP = ol.multip_2num_0

    for i in range(9):
        print("============================")
        for j in range(9):
            regIP.write(0x10, i + 1)
            regIP.write(0x18, j + 1)
            Res = regIP.read(0x20)
            print(str(i + 1) + " * " + str(j + 1) + " = " + str(Res))
    print("============================")
    print("Exit process")
```

```
Entry: /usr/local/share/pynq-venv/lib/python3.8/site-packages/ipykernel_launcher.py
System argument(s): 3
Start of "/usr/local/share/pynq-venv/lib/python3.8/site-packages/ipykernel_launcher.
py"
============================
1 * 1 = 1
1 * 2 = 2
1 * 3 = 3
1 * 4 = 4
1 * 5 = 5
1 * 6 = 6
1 * 7 = 7
1 * 8 = 8
1 * 9 = 9
============================
2 * 1 = 2
2 * 2 = 4
2 * 3 = 6
2 * 4 = 8
2 * 5 = 10
2 * 6 = 12
2 * 7 = 14
2 * 8 = 16
2 * 9 = 18
============================
3 * 1 = 3
3 * 2 = 6
3 * 3 = 9
3 * 4 = 12
3 * 5 = 15
3 * 6 = 18
```

```
3 * 7 = 21
3 * 8 = 24
3 * 9 = 27
============================
4 * 1 = 4
4 * 2 = 8
4 * 3 = 12
4 * 4 = 16
4 * 5 = 20
4 * 6 = 24
4 * 7 = 28
4 * 8 = 32
4 * 9 = 36
============================
5 * 1 = 5
5 * 2 = 10
5 * 3 = 15
5 * 4 = 20
5 * 5 = 25
5 * 6 = 30
5 * 7 = 35
5 * 8 = 40
5 * 9 = 45
============================
6 * 1 = 6
6 * 2 = 12
6 * 3 = 18
6 * 4 = 24
6 * 5 = 30
6 * 6 = 36
6 * 7 = 42
6 * 8 = 48
6 * 9 = 54
============================
7 * 1 = 7
7 * 2 = 14
7 * 3 = 21
7 * 4 = 28
7 * 5 = 35
7 * 6 = 42
7 * 7 = 49
7 * 8 = 56

7 * 9 = 63
============================
8 * 1 = 8
8 * 2 = 16
8 * 3 = 24
8 * 4 = 32
8 * 5 = 40
8 * 6 = 48
8 * 7 = 56
8 * 8 = 64
8 * 9 = 72
============================
9 * 1 = 9
9 * 2 = 18
9 * 3 = 27
9 * 4 = 36
9 * 5 = 45
9 * 6 = 54
9 * 7 = 63
9 * 8 = 72
9 * 9 = 81
============================
Exit process
```