Lab #10: Application with FPGA Accelerator

06/07/2018

4190.309A: Hardware System Design
(Spring 2018)

Lab 10: Overview

- Prepare Hardware API for Application
  - DNN Framework requires large matrix-vector multiplier
- Extending Matrix-Vector Multiplier
  - Support matrices whose width or height is larger than 64

MNIST Dataset

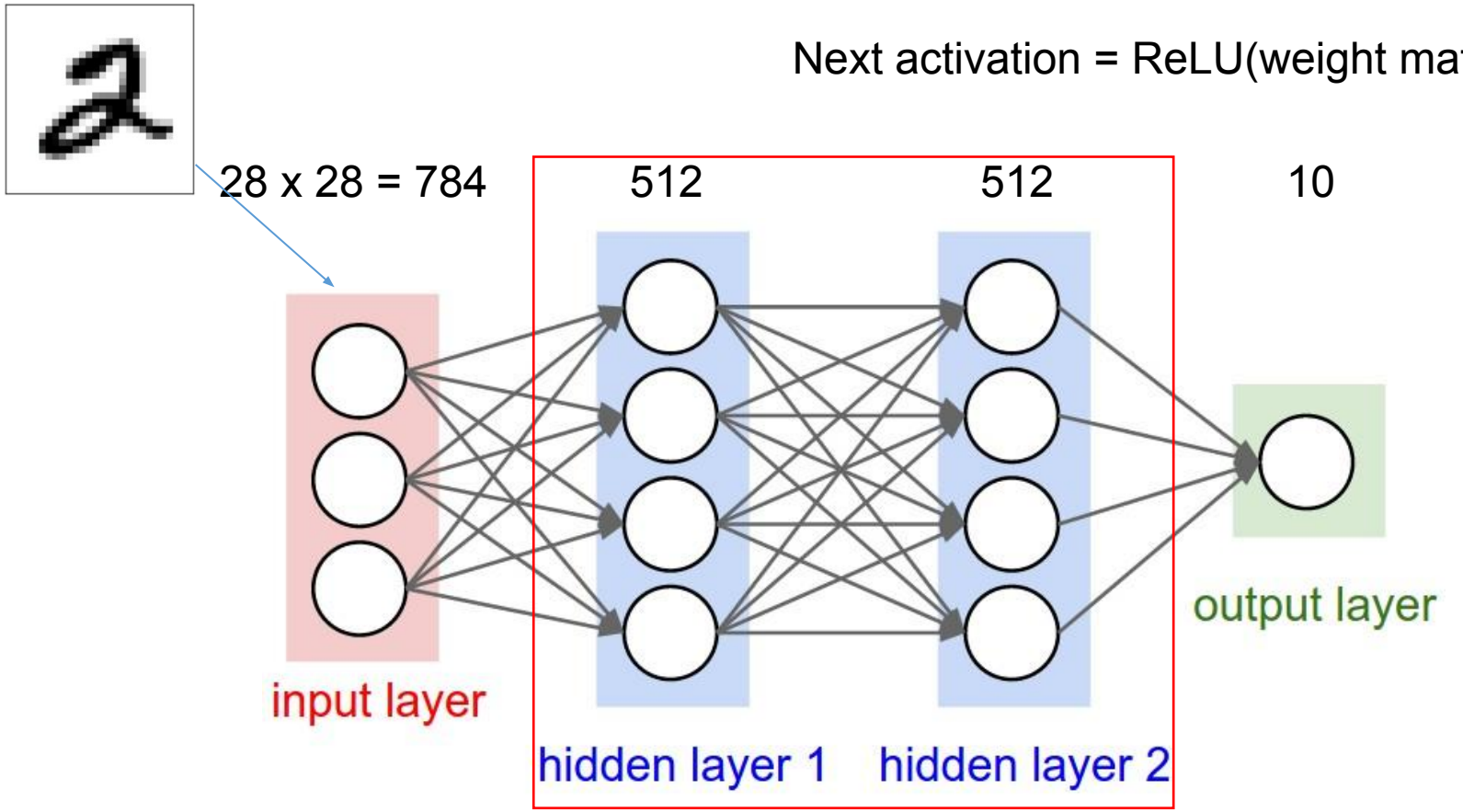
- Handwritten digit database by Yann Lecun
  - World-famous toy problem for recognition

  - 28 x 28 monolithic image
  - 50000 images for training
  - 10000 images for test

  - The state-of-the art result: 99.79 %

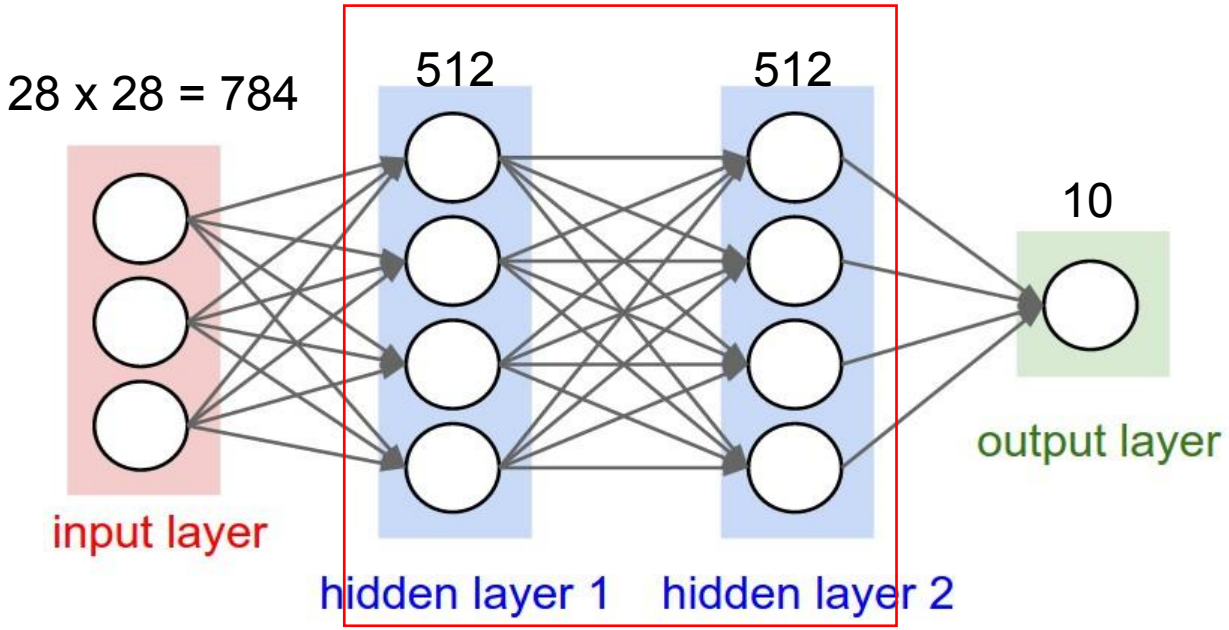
DNN for MNIST

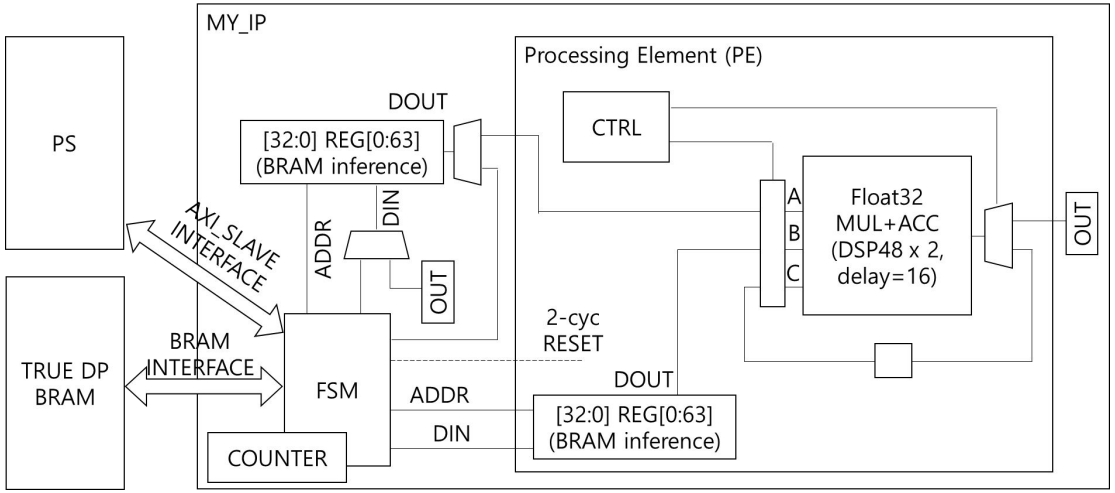
- 3 Layer MLP for MNIST dataset



Problem?

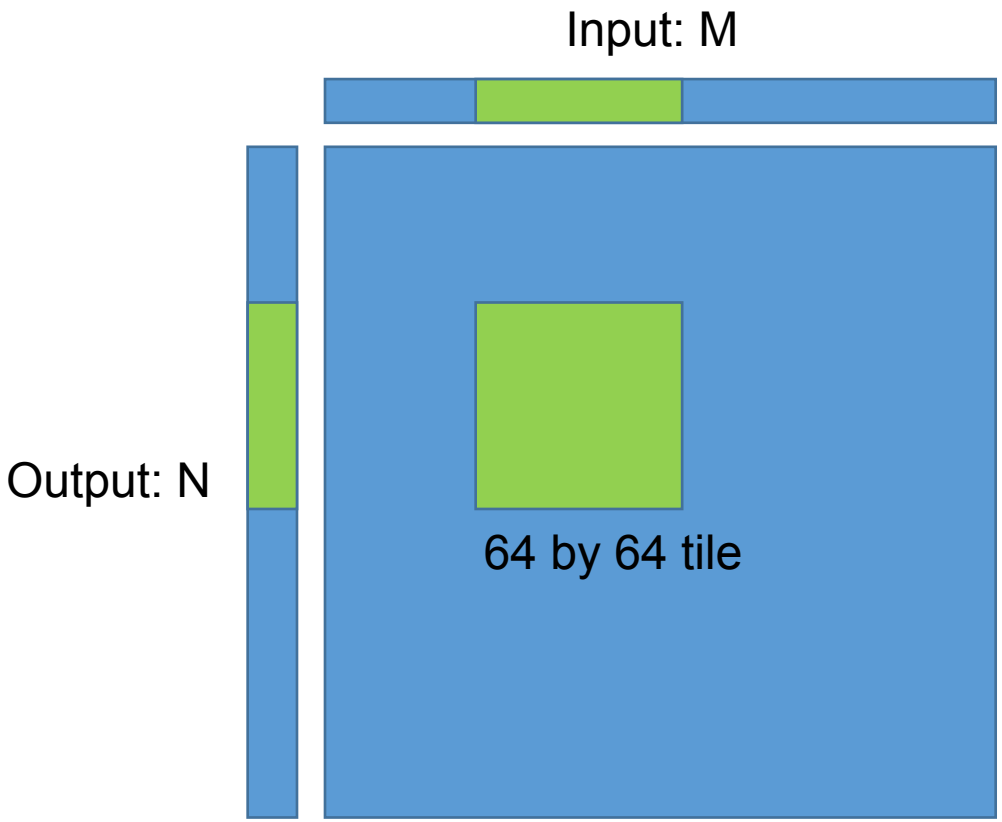
- My IP only supports 64 by 64 matrix-vector multiplication
- DNN requires 512 by 512 matrix-vector multiplication



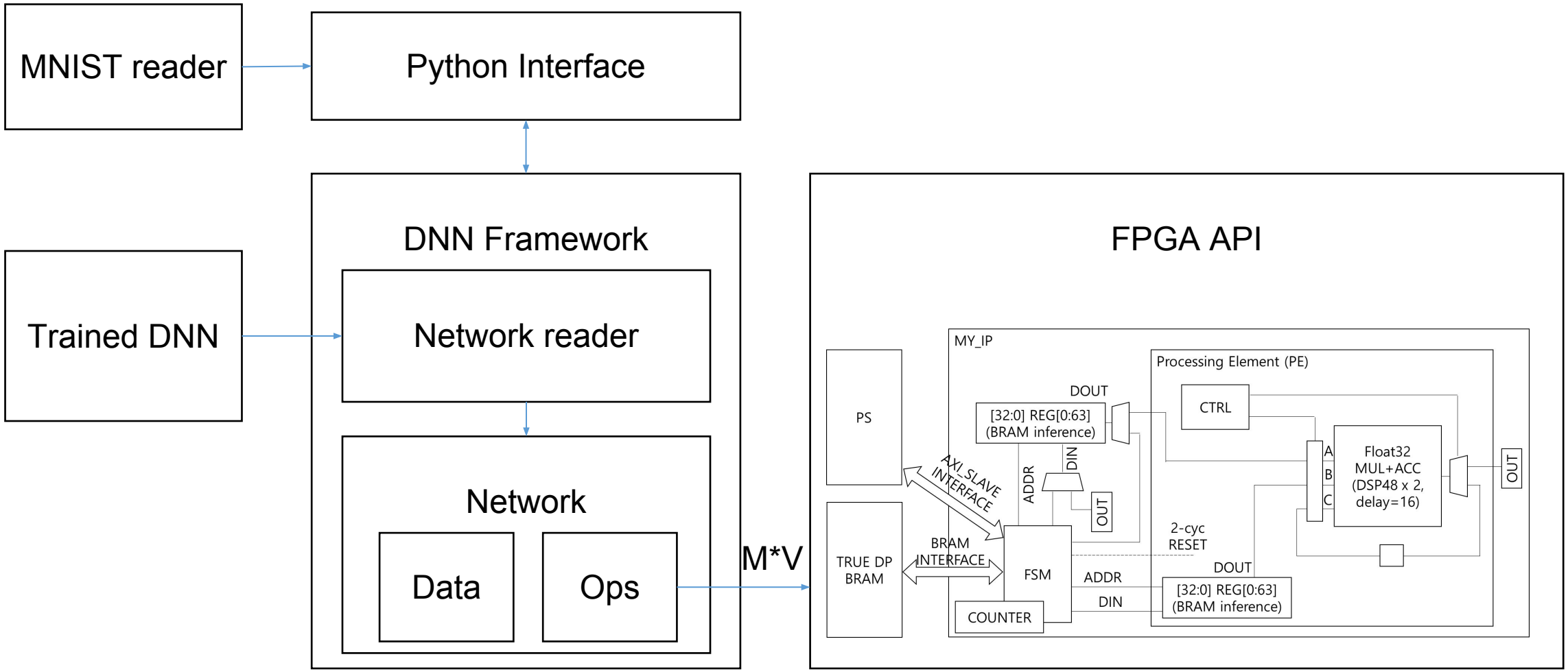


Solution

- Tiling
  - calculate the matrix-vector multiplication by splitting the matrix into small tiles which are supported by the accelerator



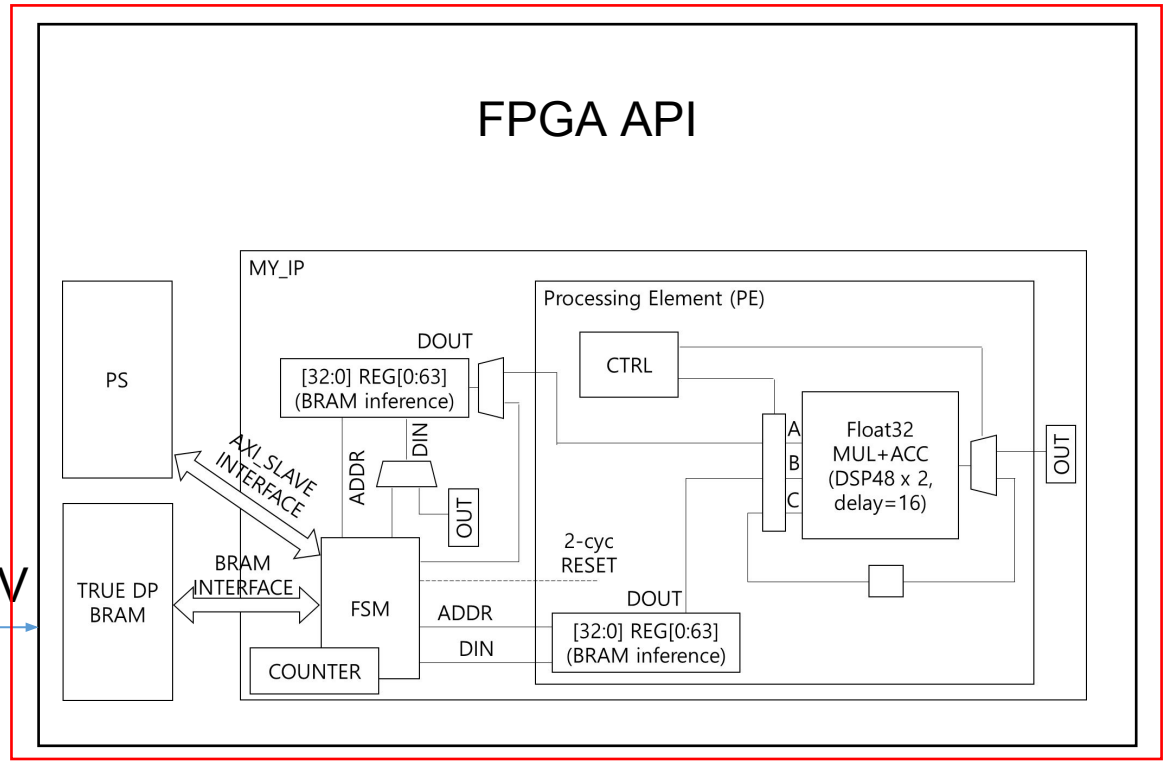
DNN Framework



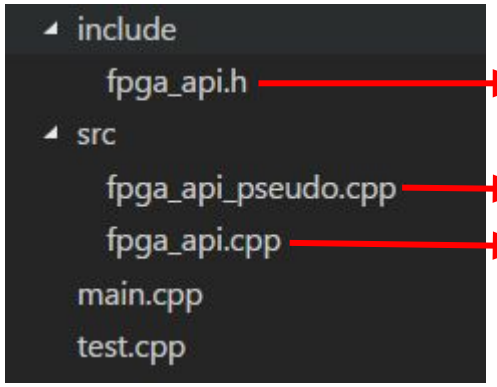
DNN Framework

```
class FPGA
{
private:
    int fd_;
    float* data_;
    unsigned int* api_;

public:
    FPGA(off_t data_addr, off_t api_addr);
    ~FPGA();

    // return internal pointer for the data
    float* matrix(void);
    float* vector(void);

    // perform matrix multiplication and return output array pointer
    const float* run();

    // input vector size: M
    // matrix size: N by M
    // output vector size: N
    // O = M * I
    void largeMV(const float* mat, const float* input,
        float* output, int M, int N);
};
```

M*V



Code Review



FPGA API definition

Code for FPGA emulation (CPU ver. Mat-Vec mul)

Code for real FPGA (Accelerator ver. Mat-Vec mul)

Goal: **Edit fpga_api.cpp** correctly to support large matrix-vector multiplication

Download: git clone https://github.com/K16DIABLO/HSD_LAB10
* You have to change boot loader image and bit (change bit file name to zynq.bit)file for matrix multiplication

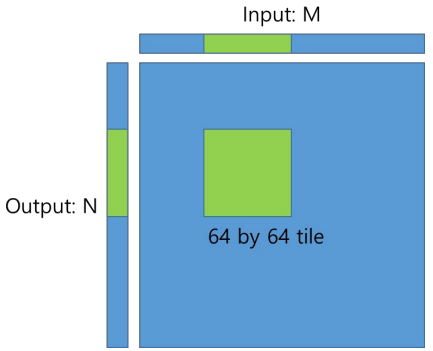
Practice: sudo g++ -I./include [test.cpp or main.cpp] ./src/fpga_api.cpp -o run.exe && sudo ./run.exe

Examination: sudo g++ -I./include [test.cpp or main.cpp] ./src/fpga_api_pseudo.cpp -o run.exe && sudo ./run.exe

Goal

- fpga_api.cpp

```
void FPGA::largeMV(const float* large_mat, const float* input,
        float* output, int M, int N)
{
    float* vec = this->vector();
    float* mat = this->matrix();

    // your code here
}
```



- Run main.cpp run file and show that the output result is correct (100 pts)
  - Compress fpga_api.cpp and a screenshot of terminal output into "L10.zip" and submit it.
  - Due : 6/12 (Tue) 11:59 PM

index	CPU	FPGA
0	29.571375	29.571379 %
1	29.198685	29.198683 %
2	26.202524	26.202526 %
3	25.465837	25.465832 %
4	32.068344	32.068344 %
5	27.400455	27.400452 %
6	27.644554	27.644562 %
7	26.395996	26.395996 %
8	25.957373	25.957371 %
9	28.004372	28.004375 %
10	25.970810	25.970808 %
11	25.726191	25.726189 %
12	25.495695	25.495695 %
13	25.319040	25.319042 %
14	28.147896	28.147898 %
15	29.486328	29.486328 %
16	26.942915	26.942913 %
17	27.058779	27.058783 %
18	24.272160	24.272163 %
19	28.154261	28.154255 %
20	25.774593	25.774593 %
21	28.984474	28.984474 %
22	26.771263	26.771263 %
23	28.975391	28.975395 %
24	26.334663	26.334669 %
25	26.550322	26.550322 %
26	25.457830	25.457832 %
27	24.677580	24.677582 %
28	30.304474	30.304474 %
29	25.673882	25.673885 %
30	30.522638	30.522636 %
31	26.585258	26.585266 %
32	25.778601	25.778599 %
33	28.370478	28.370480 %
34	30.132090	30.132082 %
35	27.529270	27.529263 %
36	23.902092	23.902086 %
37	29.841188	29.841188 %
38	29.740511	29.740515 %
39	26.522060	26.522057 %
40	28.214262	28.214264 %
41	25.827162	25.827164 %
42	29.654276	29.654274 %
43	25.854925	25.854927 %
44	28.670488	28.670486 %

Running IP for Neural Network

Goal

- Connect to server, run DNN and check that HW works correctly.
  - Submit “L11.**pdf**” (containing a screenshot only) on eTL
  - Due : 6/17 (Sun) 11:59 PM
- Server running time : 6/10(Sun) 9:00 AM ~ 6/15(Fri) 11:59 PM
- To avoid multiple accesses at a time, please write down your team number and class at link below before you connect.
  - https://docs.google.com/spreadsheets/d/1loOxgpYF1Fr-jsu55gj9mkVkjdaVBOLiaTn_tFY8g0w/edit?usp=sharing
- We will notify how to connect to the server later.

Term Project 2

- File Submission Due : 6/17 (Sun) 11:59 PM
- We will check your Demo by : 6/17 (Sun) 4:00 PM
  - If you cannot make the time, please send us a short video.

- Term project 1&2 report due : 6/17 (Sun) 11:59 PM
  - Briefly describe how your code works in 1 ~ 2 pages.
  - Check eTL for where to submit.

- For delayed submissions : score * (1 - 0.1 * delayed_dates).

Board Return

- Due : 6/24(Sun) 9:00 AM
  - Put your boards and SD card in your cabinet.
  - Please let SD card visible.
  - Make sure that the USB cable and power cable are in the box.