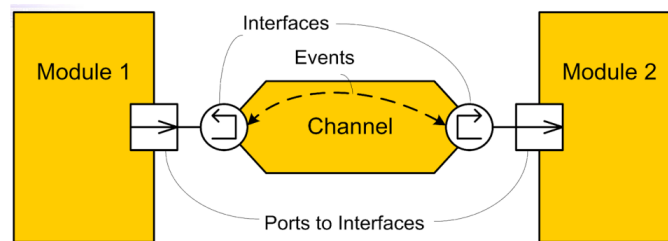


# Machine Learning Intelligent Chip Design

## Homework2 Channel and Interface

### Description

In SystemC, **interface** is an abstract class that inherits from `sc_interface`. Interface can be used to define communication protocols between different modules. It only describes the communication methods and protocols and does not involve specific data transmission. A SystemC **channel** is a class that implements one or more SystemC interface classes and inherits from either `sc_channel` or `sc_prim_channel`. The following figure shows some primitive channels and interfaces defined by SystemC library:



#### ❖ Channel

- ❖ `sc_signal<T>`
- ❖ `sc_signal_resolved`
- ❖ `sc_signal_rv<W>`
- ❖ `sc_buffer<T>`
- ❖ `sc_fifo<T>`
- ❖ `sc_mutex`
- ❖ `sc_semaphore`

#### ❖ Interface

- ❖ `sc_fifo_in_if`
- ❖ `sc_fifo_out_if`
- ❖ `sc_mutex_if`
- ❖ `sc_semaphore_if`
- ❖ `sc_signal_in_if`
- ❖ `sc_signal_out_if`

### Implementation Details

The purpose of this homework assignment is to give students an opportunity to practice using **sc\_signal**, **sc\_buffer**, and **sc\_fifo** to establish a communication mechanism between different modules. You are required to divide the AlexNet model implemented in HW1 into several sub-modules (not necessarily distinguished by layer) and then connect them using these three SystemC channels. The result of the model's execution should be identical to that of HW1.

## Implement Notes

---

- **Module Boundaries:**

Ensure that each submodule has clearly defined boundaries and that any communication and interaction between them is clearly specified. This helps reduce coupling between modules, making the code easier to understand and debug.

- **Module functionality:**

Each submodule should have specific functionality, and these functionality should be as independent and reusable as possible. This makes the code more modular and individual modules can be easily replaced or modified when needed without affecting other modules.

- **Testing and Validation:**

Each submodule is unit tested to ensure it functions properly and as expected. In addition, integration testing should be performed on the entire system to verify the interfaces and interactions between modules.

## Submission Guidelines

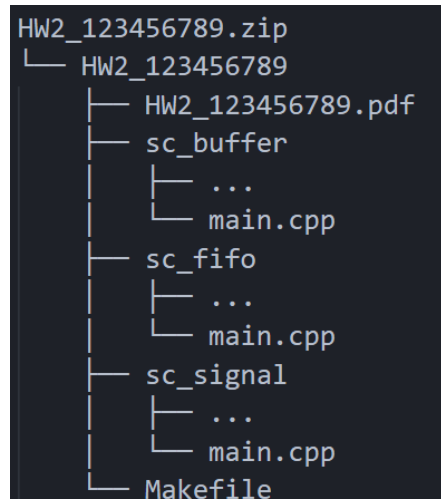
---

- **Please compress a folder named HW<ID>\_<student-ID> into a zip file with the same name and upload it to E3.**

- **The folder should include:**

- Report (Name: HW<ID>\_<student ID>.pdf)
- Codes
- Makefile

- **Example:**



```
HW2_123456789.zip
├── HW2_123456789
│   ├── HW2_123456789.pdf
│   ├── sc_buffer
│   │   ├── ...
│   │   └── main.cpp
│   ├── sc_fifo
│   │   ├── ...
│   │   └── main.cpp
│   ├── sc_signal
│   │   ├── ...
│   │   └── main.cpp
│   └── Makefile
```

- **You don't need to upload parameters.**
- **Ensure that your code is well-commented and organized for clarity and understanding.**
- **Plagiarism is forbidden, otherwise you will get 0 point!!!**

## Deliverables

---

- **SystemC Implementation:**

Use SystemC to implement the AlexNet architecture and connect modules through `sc_signal`, `sc_buffer` and `sc_fifo`.

- **Report:**

A brief report containing

- Simulation results demonstrate the predicted output for the provided input data.
- Your implementation approach, challenges faced, and any observations or insights gained during the implementation and simulation process.