

MLchip HW2 report

413510028 林柏佑

Code Structure



My code structure is the same as in HW1 because the main task was simply changing the data type between different modules (sc_signal, sc_buffer, and sc_fifo) and making some modifications to implement it.

Code Overview

My code presents a modular implementation of AlexNet, designed to ensure efficient input processing and accurate output generation. Below is a detailed overview of each file's role and functionality.

File Descriptions

layer.h, layer.cpp : module's declaration & definition.

Including convolutional layers, fully connected layers, max pooling function, and softmax function. They encapsulate the computations needed for the network, making the architecture more structured and reusable.

Pattern.h, Pattern.cpp : provide input data & evaluates the correctness of the results
 These files are responsible for providing input data to the network and verifying that the inference results are correct. It ensures that the model receives properly formatted inputs and evaluates the correctness of the outputs.

alexnet.cpp : signal connect between layers & module create

This file serves as the central hub for building and connecting the modules of AlexNet. It initializes and constructs the network, establishes the connections between layers.

Analysis

1. sc_signal

sc_signal is the most fundamental communication mechanism in SystemC, used to simulate signals in hardware.

Characteristics

- **Single writer, multiple readers:** One module can write to sc_signal, while multiple modules can read from it.
- **No buffering:** sc_signal does not store historical data, only the most recent value.
- **Propagation delay:** The value of sc_signal is updated in the next delta cycle of the simulation.
- **Used for simulating signal transmission in hardware.**

Use Cases

- Suitable for simulating clock signals, reset signals, or control signals between modules.
- Ideal for scenarios where multiple modules need to read the same data.

2. sc_buffer

sc_buffer is an extended version of sc_signal, featuring built-in "double-state" behavior.

Characteristics

- **Single writer, multiple readers:** Similar to sc_signal, one module writes, and multiple modules read.
- **Automatic triggering:** A change in value automatically triggers sensitive modules.
- **Double-state behavior:** If the same value is written consecutively, sc_buffer will still trigger sensitive modules.

Use Cases

- Suitable for scenarios where redundant triggers should be avoided, such as simulating state-holding behavior in hardware.

3. `sc_fifo`

`sc_fifo` is a communication mechanism designed to simulate FIFO buffers.

Characteristics

- **Single writer, single reader:** One module writes, and another module reads.
- **First-In-First-Out (FIFO):** Data is read in the same order as it was written.
- **Capacity limit:** The FIFO size can be specified. Writing blocks when full, and reading blocks when empty.
- **Supports both blocking and non-blocking operations:**
 - `write()` and `read()` are blocking operations.
 - `nb_write()` and `nb_read()` are non-blocking operations.

Use Cases

- Suitable for producer-consumer models.
- Ideal for buffering data, such as simulating FIFO buffers in hardware.

Comparison Summary

Feature	<code>sc_signal</code>	<code>sc_buffer</code>	<code>sc_fifo</code>
Multi-reader support	Yes	Yes	No
Multi-writer support	No	No	No
Buffering behavior	None	None	Yes
Data storage	Stores only the latest value	Stores only the latest value	FIFO buffer
Blocking behavior	None	None	Blocks on full write, blocks on empty read
Non-blocking operations	Not supported	Not supported	Supported (<code>nb_write</code> / <code>nb_read</code>)
Suitable scenarios	Control signals, multi-reader scenarios	State retention	Producer-consumer model

Selection Recommendations

- Use `sc_signal` when simulating hardware signal transmission and when multiple modules need to read the same data.
- Use `sc_buffer` when avoiding redundant triggers on sensitive modules is necessary.
- Use `sc_fifo` when simulating FIFO buffers or implementing a producer-consumer model.

Challenges

`sc_buffer` is an extended version of `sc_signal`, so changing the data type between them is straightforward and presents no challenges. However, `sc_fifo` is fundamentally different, which led to some difficulties.

```
sc_vector<sc_fifo<double>> > img_out("img_out", 150528);
sc_vector<sc_fifo<double>> > conv0_out("conv0_out", 193600);
sc_vector<sc_fifo<double>> > max0_out("max0_out", 46656);
sc_vector<sc_fifo<double>> > conv1_out("conv1_out", 139968);
sc_vector<sc_fifo<double>> > max1_out("max1_out", 32448);
sc_vector<sc_fifo<double>> > conv3_out("conv3_out", 43264);
sc_vector<sc_fifo<double>> > conv2_out("conv2_out", 64896);
sc_vector<sc_fifo<double>> > conv4_out("conv4_out", 43264);
sc_vector<sc_fifo<double>> > max2_out("max2_out", 9216);
sc_vector<sc_fifo<double>> > linear0_out("linear0_out", 4096);
sc_vector<sc_fifo<double>> > linear1_out("linear1_out", 4096);
sc_vector<sc_fifo<double>> > linear2_out("linear2_out", 1000);
sc_vector<sc_fifo<double>> > softmax_out("softmax_out", 1000);
sc_vector<sc_fifo<double>> > softmax_linear("softmax_linear", 1000);
```

As picture, I used `sc_fifo` as the data type within an `sc_vector`. Initially, I wrote data to the `sc_vector` twice: once during reset (to initialize all values to zero) and again when each module completed its computation (such as CONV, MP, FC, or SOFTMAX). However, when reading from the `sc_fifo`, I always ended up retrieving a zero, which confused me for a long time.

I eventually realized that in this situation, I needed to read twice to obtain the correct result. To resolve this issue, I decided to remove the reset operation. Without the reset, there is no need for a second read to get the correct answer. With this adjustment, I was finally able to run the model successfully!

Result

sc_signal

```
hw2_1_signal > result_cat_hw2_1_signal.log
1 ./run cat.txt
2
3 SystemC 2.3.3-Accellera --- Mar  4 2025 01:46:38
4 Copyright (c) 1996-2018 by all Contributors,
5 ALL RIGHTS RESERVED
6 Top 100 classes:
7 =====
8   idx |      val | possibility | class name
9   -----
10  285 |  20.21 |    96.38 | Egyptian cat
11  281 |  16.14 |     1.65 | tabby
12  282 |  15.73 |     1.10 | tiger cat
13  287 |  14.79 |     0.43 | lynx
14  728 |  14.41 |     0.29 | plastic bag
15  330 |  12.73 |     0.05 | wood rabbit
16  331 |  12.19 |     0.03 | hare
17  457 |  10.94 |     0.01 | bow tie
18  335 |  10.67 |     0.01 | fox squirrel
19  463 |  10.57 |     0.01 | bucket
20  478 |  10.32 |     0.00 | carton
21  876 |  10.29 |     0.00 | tub
22  622 |  10.18 |     0.00 | lens cap
23  904 |  10.01 |     0.00 | window screen
24  700 |   9.56 |     0.00 | paper towel
25  278 |   9.39 |     0.00 | kit fox
26   8 |   9.29 |     0.00 | hen
```

```
hw2_1_signal > result_dog_hw2_1_signal.log
1 ./run dog.txt
2
3 SystemC 2.3.3-Accellera --- Mar  4 2025 01:46:38
4 Copyright (c) 1996-2018 by all Contributors,
5 ALL RIGHTS RESERVED
6 Top 100 classes:
7 =====
8   idx |      val | possibility | class name
9   -----
10  207 |  16.59 |    38.63 | golden retriever
11  175 |  15.57 |    13.86 | otterhound
12  220 |  15.36 |    11.26 | Sussex spaniel
13  163 |  15.00 |     7.86 | bloodhound
14  219 |  14.59 |     5.22 | cocker spaniel
15  168 |  14.39 |     4.28 | redbone
16  160 |  14.35 |     4.07 | Afghan hound
17  213 |  14.18 |     3.46 | Irish setter
18  291 |  14.10 |     3.19 | lion
19  211 |  13.01 |     1.07 | vizsla
20  244 |  12.81 |     0.88 | Tibetan mastiff
21  216 |  12.69 |     0.78 | clumber
22  200 |  12.46 |     0.62 | Tibetan terrier
23  159 |  12.42 |     0.59 | Rhodesian ridgeback
24  152 |  12.38 |     0.57 | Japanese spaniel
25  167 |  12.01 |     0.39 | English foxhound
26  208 |  11.65 |     0.28 | Labrador retriever
```

sc_buffer

```
hw2_2_buffer > result_cat_hw2_2_buffer.log
1 ./run cat.txt
2
3 SystemC 2.3.3-Accellera --- Mar  4 2025 01:46:38
4 Copyright (c) 1996-2018 by all Contributors,
5 ALL RIGHTS RESERVED
6 Top 100 classes:
7 =====
8   idx |      val | possibility | class name
9   -----
10  285 |  20.21 |    96.38 | Egyptian cat
11  281 |  16.14 |     1.65 | tabby
12  282 |  15.73 |     1.10 | tiger cat
13  287 |  14.79 |     0.43 | lynx
14  728 |  14.41 |     0.29 | plastic bag
15  330 |  12.73 |     0.05 | wood rabbit
16  331 |  12.19 |     0.03 | hare
17  457 |  10.94 |     0.01 | bow tie
18  335 |  10.67 |     0.01 | fox squirrel
19  463 |  10.57 |     0.01 | bucket
20  478 |  10.32 |     0.00 | carton
21  876 |  10.29 |     0.00 | tub
22  622 |  10.18 |     0.00 | lens cap
23  904 |  10.01 |     0.00 | window screen
24  700 |   9.56 |     0.00 | paper towel
25  278 |   9.39 |     0.00 | kit fox
26   8 |   9.29 |     0.00 | hen
```

```
hw2_2_buffer > result_dog_hw2_2_buffer.log
1 ./run dog.txt
2
3 SystemC 2.3.3-Accellera --- Mar  4 2025 01:46:38
4 Copyright (c) 1996-2018 by all Contributors,
5 ALL RIGHTS RESERVED
6 Top 100 classes:
7 =====
8   idx |      val | possibility | class name
9   -----
10  207 |  16.59 |    38.63 | golden retriever
11  175 |  15.57 |    13.86 | otterhound
12  220 |  15.36 |    11.26 | Sussex spaniel
13  163 |  15.00 |     7.86 | bloodhound
14  219 |  14.59 |     5.22 | cocker spaniel
15  168 |  14.39 |     4.28 | redbone
16  160 |  14.35 |     4.07 | Afghan hound
17  213 |  14.18 |     3.46 | Irish setter
18  291 |  14.10 |     3.19 | lion
19  211 |  13.01 |     1.07 | vizsla
20  244 |  12.81 |     0.88 | Tibetan mastiff
21  216 |  12.69 |     0.78 | clumber
22  200 |  12.46 |     0.62 | Tibetan terrier
23  159 |  12.42 |     0.59 | Rhodesian ridgeback
24  152 |  12.38 |     0.57 | Japanese spaniel
25  167 |  12.01 |     0.39 | English foxhound
26  208 |  11.65 |     0.28 | Labrador retriever
```

sc_fifo

hw2_3_fifo > result_cat_hw2_3_fifo.log					hw2_3_fifo > result_dog_hw2_3_fifo.log				
1	./run cat.txt				1	./run dog.txt			
2					2				
3	SystemC 2.3.3-Accellera --- Mar 4 2025 01:46:38				3	SystemC 2.3.3-Accellera --- Mar 4 2025 01:46:38			
4	Copyright (c) 1996-2018 by all Contributors,				4	Copyright (c) 1996-2018 by all Contributors,			
5	ALL RIGHTS RESERVED				5	ALL RIGHTS RESERVED			
6	Top 100 classes:				6	Top 100 classes:			
7	=====				7	=====			
8	idx	val	possibility	class name	8	idx	val	possibility	class name
9	=====				9	=====			
10	285	20.21	96.38	Egyptian cat	10	207	16.59	38.63	golden retriever
11	281	16.14	1.65	tabby	11	175	15.57	13.86	otterhound
12	282	15.73	1.10	tiger cat	12	220	15.36	11.26	Sussex spaniel
13	287	14.79	0.43	lynx	13	163	15.00	7.86	bloodhound
14	728	14.41	0.29	plastic bag	14	219	14.59	5.22	cocker spaniel
15	330	12.73	0.05	wood rabbit	15	168	14.39	4.28	redbone
16	331	12.19	0.03	hare	16	160	14.35	4.07	Afghan hound
17	457	10.94	0.01	bow tie	17	213	14.18	3.46	Irish setter
18	335	10.67	0.01	fox squirrel	18	291	14.10	3.19	lion
19	463	10.57	0.01	bucket	19	211	13.01	1.07	vizsla
20	478	10.32	0.00	carton	20	244	12.81	0.88	Tibetan mastiff
21	876	10.29	0.00	tub	21	216	12.69	0.78	clumber
22	622	10.18	0.00	lens cap	22	200	12.46	0.62	Tibetan terrier
23	904	10.01	0.00	window screen	23	159	12.42	0.59	Rhodesian ridgeback
24	700	9.56	0.00	paper towel	24	152	12.38	0.57	Japanese spaniel
25	278	9.39	0.00	kit fox	25	167	12.01	0.39	English foxhound
26	8	9.29	0.00	hen	26	208	11.65	0.28	Labrador retriever

Result

```

Server_Account mlchip070
Cat - signal (10%) PASS
Dog - signal (10%) PASS
Cat - buffer (10%) PASS
Dog - buffer (10%) PASS
Cat - fifo (10%) PASS
Dog - fifo (10%) PASS
Error_Message No_Error
Submit Date 2025/04/02
Submit Time 22:58:19
Sim_Time - signal (s) 42.92
Sim_Time - buffer (s) 43.16
Sim_Time - fifo (s) 32.14

[Info] Your file will be submitted to: TA folder
[Warning] demo has been submitted.
[Warning] It will overwrite your original file.
[Info] Now submit hw2_mlchip070.tar.gz file to system.
[Success] Copying Successfully.

=====
Submit Report
=====
Result : has been submitted.
Submission time : 2025/04/02 22:58:19
=====

-- Congratulations !! --
-- Submission Successful!! --
--                               --
Please remember to check your submission with ./02_check !!
Please remember to check your submission with ./02_check !!
Please remember to check your submission with ./02_check !!
=====
10:54 mlchip070@ee25[~/hw2/09_SUBMIT]%

```