



Assignment6

FIR Filter

| | |
|------|-------------------|
| 과목명 | 하드웨어소프트웨어 통합설계 |
| 담당교수 | 이준환 교수님 |
| 학과 | 컴퓨터정보공학부 |
| 학년 | 3학년 |
| 학번 | 2019202009 |
| 이름 | 서여지 |
| 제출일 | 2021.12.02 (목) |

1. 과제개요

FIR Filter와 testbench를 작성하여 동작을 확인하는 실습을 진행한다. FIR Filter는 16bit 크기의 input을 입력 받고, 동일한 크기의 output을 출력한다. input에 {18,77,107,77,17} 값을 갖는 coefficient를 곱하여 output을 계산한다.

2. 코드설명

FIR Filter모듈 fir은 가장 처음 실행되었을 때 taps를 0으로 초기화하고 output에 0을 쓰고 wait하여 초기화 한다. 이후 taps의 값을 shift하고, 각 taps의 값에 coefficient값인 coef의 값을 곱하여 결과 val을 output에 쓰는 동작을 한다.

FIR Filter의 testbench tb는 fir의 output을 input으로, input을 output으로 갖는다. source는 fir에 보내는 입력 값을 다루고, sink는 fir가 tb로 보낸 출력 값을 화면에 출력한다.

fir과 tb는 모두 input과 output에 대한 vld와 rdy signal을 사용한다. vld signal이 1일 때 값이 정상적으로 읽어지고, rdy signal이 1이 될 때까지 입력 혹은 출력 값을 유지한다.

FIR Filter와 testbench를 모두 포함하는 전체 시스템 SYSTEM 모듈은 fir과 tb의 인스턴스를 생성하고, clock, reset, input, output, handshaking signal을 선언하여 전체 프로그램을 동작하게 한다. SYSTEM모듈은 fir과 tb의 입출력을 연결한다.

3. 실행결과

tb가 source에서 input data를 생성하는 부분은 다음과 같다.

```
for (int i = 0; i < 64; i++) {  
    if (i > 23 && i < 29)  
        tmp = 256;  
    else  
        tmp = 0;  
}
```

또한 source는 처음 2개의 0값을 전달한 이후 for 반복문을 실행하므로, 27,28,29,30,31 번째 입력은 256값을 갖고, 나머지 입력 값은 0이다.

fir의 coefficient는 다음의 값을 갖는다.

```
coef[5] = { 18, 77, 107, 77, 18 };
```

fir은 tab을 한 칸씩 shift하여 동작하므로 실행 결과는 다음과 같이 예상할 수 있다 .

| clock | tab[5] | output | |
|-------|-------------------------|-------------------------------------|-------|
| 0~27 | 0, 0, 0, 0, 0 | 0 | 0 |
| 28 | 256, 0, 0, 0, 0 | 256*18 | 4608 |
| 29 | 256, 256, 0, 0, 0 | 256*18+256*77 | 24320 |
| 30 | 256, 256, 256, 0, 0 | 256*18+256*77+256*107 | 51712 |
| 31 | 256, 256, 256, 256, 0 | 256*18+256*77+256*107+256*77 | 71424 |
| 32 | 256, 256, 256, 256, 256 | 256*18+256*77+256*107+256*77+256*18 | 76032 |
| 33 | 0, 256, 256, 256, 256 | 256*77+256*107+256*77+256*18 | 71424 |
| 34 | 0, 0, 256, 256, 256 | 256*107+256*77+256*18 | 51712 |
| 35 | 0, 0, 0, 256, 256 | 256*77+256*18 | 24320 |
| 36 | 0, 0, 0, 0, 256 | 256*18 | 4608 |
| 37~63 | 0, 0, 0, 0, 0 | 0 | 0 |

실제 실행 결과는 다음과 같이 출력되었다.

```

0 : 0
1 : 0
2 : 0
3 : 0
4 : 0
5 : 0
6 : 0
7 : 0
8 : 0
9 : 0
10 : 0
11 : 0
12 : 0
13 : 0
14 : 0
15 : 0
16 : 0
17 : 0
18 : 0
19 : 0
20 : 0
21 : 0
22 : 0
23 : 0
24 : 0
25 : 0
26 : 0
27 : 0
28 : 4608
29 : 24320
30 : -13824
31 : 5888
32 : 10496
33 : 5888
34 : -13824
35 : 24320
36 : 4608
37 : 0
38 : 0
39 : 0
40 : 0
41 : 0
42 : 0
43 : 0
44 : 0
45 : 0
46 : 0
47 : 0
48 : 0
49 : 0
50 : 0
51 : 0
52 : 0
53 : 0
54 : 0
55 : 0
56 : 0
57 : 0
58 : 0
59 : 0
60 : 0
61 : 0
62 : 0
63 : 0

```

| clock | 실제 결과 | | 예상 결과 | |
|-------|--------|---------------------|-------|-----------------------|
| 0~27 | 0 | 0000 0000 0000 0000 | 0 | 0000 0000 0000 0000 |
| 28 | 4608 | 0001 0010 0000 0000 | 4608 | 0001 0010 0000 0000 |
| 29 | 24320 | 0101 1111 0000 0000 | 24320 | 0101 1111 0000 0000 |
| 30 | -13824 | 1100 1010 0000 0000 | 51712 | 1100 1010 0000 0000 |
| 31 | 5888 | 0001 0111 0000 0000 | 71424 | 1 0001 0111 0000 0000 |
| 32 | 10496 | 0010 1001 0000 0000 | 76032 | 1 0010 1001 0000 0000 |
| 33 | 5888 | 0001 0111 0000 0000 | 71424 | 1 0001 0111 0000 0000 |
| 34 | -13824 | 1100 1010 0000 0000 | 51712 | 1100 1010 0000 0000 |
| 35 | 24320 | 0101 1111 0000 0000 | 24320 | 0101 1111 0000 0000 |
| 36 | 4608 | 0001 0010 0000 0000 | 4608 | 0001 0010 0000 0000 |
| 37~63 | 0 | 0000 0000 0000 0000 | 0 | 0000 0000 0000 0000 |

두 결과의 차이는 입력과 출력으로 16bit의 int를 사용했기 때문에 나타났음을 확인할 수 있다. FIR Filter의 동작을 Testbench와 SYSTEM을 이용하여 확인하는 실습이었다.