## コンピュータアーキテクチャ論 Ex04

S1260027 Shunsuke Onuki

課題 4-1: ALU のシミュレーション

課題 4-2: レジスタファイルのシミュレーション

課題 4-1: ALU のシミュレーション

ALU全体が正しく動作しているのかを確認するため、シミュレーション結果と期待値を比べる。以下の表を参考にシミュレーションを行う。

|           | AND: ALU制 | 御入力: 0000     |           | 加算: ALU制御入力: 0010 |           |         |           |  |  |  |
|-----------|-----------|---------------|-----------|-------------------|-----------|---------|-----------|--|--|--|
| А         | В         | ゼロ判定期待値 結果期待値 |           | A                 | В         | ゼロ判定期待値 | 結果期待値     |  |  |  |
| 0000 0000 | 0000 0000 | 1             | 0         | 0000 0000         | 0000 0000 | 1       | 0         |  |  |  |
| OFOF OFOF | F0F0 F0F0 | 1             | 0         | OFOF OFOF         | F0F0 F0F0 | 0       | FFFF FFFF |  |  |  |
| F0F0 F0F0 | OFOF OFOF | 1             | 0         | 7FFF FFFF         | 0000 0001 | 0       | 8000 0000 |  |  |  |
| FFFF FFFF | FFFF FFFF | 0             | FFFF FFFF | FFFF FFFF         | 0000 0001 | 1       | 0         |  |  |  |
| D8A3 B8D4 | 63B9 D6F2 | 0             | 40A1 90D0 | D8A3 B8D4         | 63B9 D6F2 | 0       | 3C5D 8FC6 |  |  |  |
|           | OR: ALU制  | 御入力: 0001     |           | 減算: ALU制御入力: 0110 |           |         |           |  |  |  |
| А         | В         | ゼロ判定期待値       | 結果期待値     | А                 | В         | ゼロ判定期待値 | 結果期待値     |  |  |  |
| 0000 0000 | 0000 0000 | 1             | 0         | 0000 0000         | 0000 0000 | 1       | 0         |  |  |  |
| OFOF OFOF | F0F0 F0F0 | 0             | FFFF FFFF | 0000 0000         | 0000 0001 | 0       | FFFF FFFF |  |  |  |
| F0F0 F0F0 | OFOF OFOF | 0             | FFFF FFFF | 0000 0001         | 8000 0001 | 0       | 8000 0000 |  |  |  |
| EFFE FFFF | FEFF FFFF | 0             | FEFF FFFF | 0000 0001         | 0000 0001 | 1       | 0         |  |  |  |
| FFFF FFFF | ,         |               |           |                   |           |         |           |  |  |  |

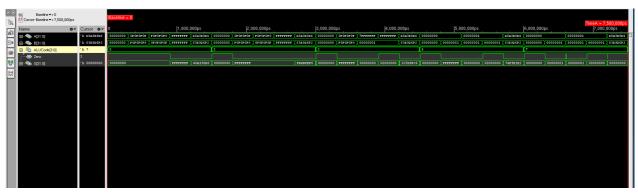
| Set on less than: ALU制御入力: 0111 |  |   |  |  |  |  |  |  |  |  |
|---------------------------------|--|---|--|--|--|--|--|--|--|--|
| В                               | ゼロ判定期待値                                  | 結果期待値   |  |  |  |  |  |  |  |  |
| 0000 0000                       | 1  | 0   |  |  |  |  |  |  |  |  |
| 0000 0001                       | 0  | 1   |  |  |  |  |  |  |  |  |
| 8000 0001                       | 0  | 1   |  |  |  |  |  |  |  |  |
| 0000 0001                       | 1  | 0   |  |  |  |  |  |  |  |  |
| 63B9 D6F2                       | 1  | 0   |  |  |  |  |  |  |  |  |
|                                 | B<br>0000 0000<br>0000 0001<br>8000 0001 | B     ゼロ判定期待値       0000 0000     1       0000 0001     0       8000 0001     0       0000 0001     1 |  |  |  |  |  |  |  |  |

## Testfixture.verilog

```
// Verilog stimulus file.
// Please do not create a mo
                                                     A[31:0] = 32'h000000000;
// Default verilog stimulus.
                                                     B[31:0] = 32'h000000000;
initial
begin
                                                     ALUCode[3:0] = 4'b0010;
   A[31:0] = 32'h00000000;
                                                  #300
                                                     A[31:0] = 32'h0F0F0F0F;
   B[31:0] = 32'h00000000;
                                                     B[31:0] = 32'hF0F0F0F0;
  ALUCode[3:0] = 4'b0000;
                                                  #300
   A[31:0] = 32'h0F0F0F0F;
                                                     A[31:0] = 32'h7FFFFFFF;
   B[31:0] = 32'hF0F0F0F0;
                                                     B[31:0] = 32'h00000001;
#300
   A[31:0] = 32'hF0F0F0F0;
                                                  #300
                                                     A[31:0] = 32'hFFFFFFFF;
   B[31:0] = 32'h0F0F0F0F;
                                                     B[31:0] = 32'h00000001;
#300
   A[31:0] = 32'hFFFFFFFF;
                                                     #300
                                                     A[31:0] = 32'hD8A3B8D4;
   B[31:0] = 32'hFFFFFFFF;
                                                     B[31:0] = 32'h63B9D6F2;
   A[31:0] = 32'hD8A3B8D4;
                                                  #300
   B[31:0] = 32'h63B9D6F2;
                                                     A[31:0] = 32'h000000000;
#300
                                                     B[31:0] = 32'h000000000;
   A[31:0] = 32'h000000000;
                                                     ALUCode[3:0] = 4'b0110;
   B[31:0] = 32'h000000000;
                                                  #300
   ALUCode[3:0] = 4'b0001;
                                                     A[31:0] = 32'h000000000;
#300
   A[31:0] = 32'h0F0F0F0F;
                                                     B[31:0] = 32'h000000001;
   B[31:0] = 32'hF0F0F0F0;
                                                  #300
                                                     A[31:0] = 32'h00000001;
#300
  A[31:0] = 32'hF0F0F0F0F0;
                                                     B[31:0] = 32'h80000001;
   B[31:0] = 32'h0F0F0F0F;
                                                  #300
#300
                                                     A[31:0] = 32'h00000001;
   A[31:0] = 32'hFFFFFFFF;
                                                     B[31:0] = 32'h00000001;
   B[31:0] = 32'hFFFFFFFF;
                                                     #300
#300
                                                     A[31:0] = 32'hD8A3B8D4;
   A[31:0] = 32'hD8A3B8D4;
                                                     B[31:0] = 32'h63B9D6F2;
   B[31:0] = 32'h63B9D6F2;
#300
                                                  #300
```

```
A[31:0] = 32'h000000000;
  B[31:0] = 32'h000000000;
  ALUCode[3:0] = 4'b0111;
#300
  A[31:0] = 32'h000000000;
  B[31:0] = 32'h00000001;
  A[31:0] = 32'h00000001;
  B[31:0] = 32'h80000001;
#300
  A[31:0] = 32'h00000001;
  B[31:0] = 32'h000000001;
#300
  A[31:0] = 32'hD8A3B8D4;
  B[31:0] = 32'h63B9D6F2;
#300
$finish;
end
```

## 結果





|                         | 2,0         | 00,000ps  |          |   | 3,000,000 | ps       |   | 4,000,000ps             |   |  |
|-------------------------|-------------|---|----------|---|-----------|----------|---|-------------------------|---|--|
| 00000000                | OF OF OF OF | FOFOFOFO  | PPEPPPPP | D8A3B8D4                                    | 00000000  | OFOFOFOF | TEFFFEFF.   | 77777777                | DSA3B8D4                                    |  |
| 00000000                | FOFOFOFO    | OFOFOFOF  | PPPPPPPP | 68B9D6F2                                    | 00000000  | POPOPOPO | 00000001  |                         | 63B9D6F2                                    |  |
| 1                       |             |   |          |   | 2         |          |   |                         |   |  |
| 1                       | 0           | 0   | 0        | 0   | 1         | 0        | 0   | 1                       | 0   |  |
| 00000000                | 77777777    |   |          | <b>РВВВРЕР</b> 6                            | 00000000  | PPPPPPP  | 80000000  | 00000000                | 3cSp8rc6                                    |  |
| Zeroが1<br>なので<br>SはOになる |             | FOFOFOFO<br>OF<br>OFOFOFOF<br>は<br>FFFFFFFF<br>LUCodeが<br>R回路の挙 |          | D8A3B8D4<br>or<br>63B9D6F2<br>は<br>FBBBFEF6 |           |          | 7FFFFFFF<br>+<br>00000001<br>は<br>80000000<br>odeが10(2)<br>1路の学動を | Zeroが1<br>なので<br>SはOになる | D8A3B8D4<br>+<br>63B9D6F2<br>id<br>3C5D8FC6 |  |

|                         | 5,0  | 00,000ps   |                         |  | 6,000,000               | os   | ı  | TimeA = 7,500,000ps<br> 7,000,000ps |  |  |
|-------------------------|--|--|-------------------------|--|-------------------------|--|--|-------------------------------------|--|--|
| 00000000                |  | 00000001   |                         | D8A3B8D4                                     | 00000000                |  | 00000001   |                                     | D8A3B8D4   |  |
| 00000000                | 00000001   | 80000001   | 00000001                | 68B9D6F2                                     | 60000000                | 00000001   | 80000001   | 00000001                            | 63B3D6F2   |  |
| 6                       |  |  |                         |  | 7                       |  |  |                                     |  |  |
| 1                       | 0  | 0  | 1                       | 0  | 1                       | 0  | 0  | 1                                   | 1  |  |
| 00000000                | 77777777   | 80000000   | 00000000                | 74E9E1E2                                     | 00000000                | 00000001   | 00000001   | 00000000                            | 00000000   |  |
| Zeroが1<br>なので<br>SはOになる | 負の符号が<br>付かない<br>ので<br>繰り下げで<br>00000000<br>-<br>00000001<br>は<br>FFFFFFFF | 負の符号が<br>付かない<br>ので<br>繰り下げで<br>00000001<br>-<br>80000001<br>は<br>80000000 | Zeroが1<br>なので<br>SはOになる | D8A3B8D4<br>-<br>63B9D6F2<br> d=<br>74E9E1E2 | Zeroが1<br>なので<br>SはOになる | 00000000<br>より<br>00000001<br>は大きい<br>ので<br>Sは1になる | 00000001<br>より<br>80000001<br>は大きい<br>ので<br>Sは1になる | Zeroが1<br>なので<br>SはOになる             | D8A3B8D4<br>より<br>63B9D6F2<br>は<br>小さいので<br>Sは0になる |  |
| Ø                       | ALUCod<br>時は減算回  | eが110(2)<br>路の挙動を3   | する                      |  |                         |  | odeが111(2<br>回路の挙動:                                |                                     |  |  |

理想値の表と見比べると正しい値が出力されていることがわかる。

## 課題 4-2: レジスタファイルのシミュレーション レジスタファイルの動作を確認するためテストベンチ(testfixture)を作成し、動作を確認する

```
Testfixture
```

```
// Verilog stimulus file.
// Please do not create a module in this file.
// Default verilog stimulus.
initial
begin
   CK = 1'b0;
   CLR = 1'b1;
   WriteEN = 1'b1;
   ReadReg1[4:0] = 5'b00001;
   ReadReg2[4:0] = 5'b00010;
   #100
   CLR = 1'b0;
   #50
   WriteData[31:0] = 32'b00000001001001100000000000100111;
   WriteReg[4:0] = 5'b00001;
   #150
   WriteReg[4:0] = 5'b00010;
   WriteData[31:0] = 32'b00000000000000000100000100001;
   #150
   WriteEN = 1'b0;
   #150
   $finish;
end
   always #50 CK = ~CK;
```

|   |                             | ⊞* ¶ ReadReg2[4:0] | ⊟்¶்டை ReadData2[31:0] | i¶a. ReadReg1[4:0] | FactData[31:0] | WriteEN | <mark>I</mark> D₁ CLR | <mark>N</mark> • CK | 🖶 🜇 WriteReg[4:0] | ⊕¶a. WriteData[31:0] | Name ••   | [4] Baseline ▼= 0<br>☐ Cursor-Baseline ▼= 600,000ps |
|---|-----------------------------|--------------------|------------------------|--------------------|----------------|---------|-----------------------|---------------------|-------------------|----------------------|-----------|---|
|   |                             | ነ  02              | ነት 00002021            | 'ት 01              | 'h 01260027    | 0       | 0                     | ц                   | 'ሕ 02             | ነት 00002021          | Cursor •  |   |
| <u></u>   | CLRが1<br>なので<br>も書き込まれな     | 02                 | 0000000                | 01                 | 00000000       | _       | -                     | 0                   | ХХ                | ххххххх              | 0         | Baseline = 0  |
| CLRが1<br>なので<br>も書き込まれば                           |                             |                    | 0                      |                    | 0              | _       | -                     | _                   |                   |                      |           | 0   |
| W W   | CKが0<br>なので<br>も書き込まわ       |                    |                        |                    |                | -       | 0                     | 0                   |                   |                      | 100,000ps |   |
| CKが1<br>WriteENが1<br>でも<br>teDataとWrit<br>に何もないのつ | 9                           |                    | ххххххх                |                    | хххххх         | _       | 0                     | _                   | Œ                 | 01260027             |           |   |
| Write<br>Reg Write                                | CKが0<br>なので<br>書き込まれな       |                    |                        |                    |                | _       | 0                     | 0                   |                   |                      | 200,000ps |   |
| RegとReadRe<br>同じなので<br>DataがReadDo                | 書き込み                        |                    |                        |                    | 01260027       | _       | 0                     | _                   |                   |                      |           |   |
| g 1 ½ Writ<br>dta1 K Write                        | CKがO<br>なので<br>き書き込まれな      |                    |                        |                    | 7              | _       | 0                     | 0                   | 02                | 00002021             | 300,000ps |   |
| eRegとReadRe<br>同じなので<br>PDataがReadBe<br>書き込まれる    | () <b>書</b> き込み             |                    | 00002071               |                    |                | _       | 0                     | _                   |                   |                      |           |   |
| tφ07<br>leg2ti<br>lata2に<br>S                     | CKが0<br>なので<br>も書き込まれ       |                    | ı                      |                    |                | _       | 0                     | 0                   |                   |                      | 400,000ps |   |
| 何も書き込まなので   | یار<br>WriteEN <i>t</i> j(0 |                    |                        |                    |                | 0       | 0                     | _                   |                   |                      |           |   |
| き込まれない<br>WriteENが0<br>なので何も書き込まれない               |                             |                    |                        |                    |                | 0       | 0                     | 0                   |                   |                      | 500,000ps | TimeA = 6   |
| nau   | WriteENがO<br>なので<br>何も書き込ま  |                    |                        |                    |                | 0       | 0                     | _                   |                   |                      |           | TimeA = 600,000ps                                   |