Complete 64-bit ALU Unit

A complete 64-bit ALU unit built using sixty-four 1-bit ALU units and an ALU Control Unit.

VERILOG CODE :

```verilog
    reg[63:0] A;
    reg[63:0] B;
    wire[63:0] Result;
    reg CarryIn;
    wire Zero, CarryOut;
    CU inst(Op, ALU_Op, funct7, funct3);
    ALU alu1(Result, Zero, CarryOut, A, B, CarryIn, Op);
    initial begin
    for(j = 0; j < 4; j++) begin
    #20 ALU_Op = 2'b00;
        CarryIn = Op[2];
        funct7 = 7'b0100111;
        funct3 = 3'b111;
        $display("Case 0: Add(ALU_Op code)\n\n");
    #20    ALU_Op = 2'b01;
        CarryIn = Op[2];
        funct7 = 7'b0000000;
        funct3 = 3'b111;
        $display("Case 1: Subtract(ALU_Op code)\n\n");
    #20 ALU_Op = 2'b10;
        CarryIn = Op[2];
        funct7 = 7'b0000000;
        funct3 = 3'b000;
        $display("Case 2: Add(R-type)\n\n");
    #20 ALU_Op = 2'b10;
        CarryIn = Op[2];
        funct7 = 7'b0100000;
        funct3 = 3'b000;
        $display("Case 3: Subtract(R-type)\n\n");
    #20 ALU_Op = 2'b10;
        CarryIn = Op[2];
        funct7 = 7'b0000000;
        funct3 = 3'b111;
        $display("Case 4: And(R-type)\n\n");
    #20 ALU_Op = 2'b10;
        CarryIn = Op[2];
        funct7 = 7'b0000000;
        funct3 = 3'b110;
        $display("Case 5: Or(R-type)\n\n");
    end
```

```verilog
    #20 ALU_Op = 2'b10;
        CarryIn = Op[2];
        funct7 = 7'b0100000;
        funct3 = 3'b000;
        $display("Case 3: Subtract(R-type)\n\n");
    #20 ALU_Op = 2'b10;
        CarryIn = Op[2];
        funct7 = 7'b0000000;
        funct3 = 3'b111;
        $display("Case 4: And(R-type)\n\n");
    #20 ALU_Op = 2'b10;
        CarryIn = Op[2];
        funct7 = 7'b0000000;
        funct3 = 3'b110;
        $display("Case 5: Or(R-type)\n\n");
    end
    end
    initial begin
        A = 1;                              //A= 1
        B = -64'h1;                         //B = -1
    #120
        A = 64'h7fffffffffffffff;                        //A = max
        B = 1;                                           //B = 1
    #120
        A = 64'h7fffffffffffffff;           //A = max
        B = 64'h8000000000000000;           //b = min
    #120
        A = 0;
        B = 0;
    end
    initial begin
    #1;
        for(i = 0; i < 24; i++) begin
    #20    $display("ALU_Op = %3b\t\tfunct7 = %8b\t\tfunct3 = %4b\n\nOpcode = %b\n", ALU_Op, funct7, funct3, Op);
            $display("Time = %4d\nA = %10h\t\tB = %10h\t\tCarry In = %3b\n\nResult = %10h\t\tZero = %3d\t\tCarry Out = %b\n\n\n", $time, A[63
:0], B[63:0], CarryIn,  Result[63:0], Zero, CarryOut);
        end
    end
endmodule
```

OUTPUT :

```
File Edit View Search Terminal Help

ALU_Op =  00          funct7 =  0100111          funct3 =  111

Opcode = 0010

Time =   21
A = 0000000000000001          B = ffffffffffffffff          Carry In =   x

Result = 0000000000000000          Zero =   1          Carry Out = 1



Case 1: Subtract(ALU_Op code)


ALU_Op =  01          funct7 =  0000000          funct3 =  111

Opcode = 0110

Time =   41
A = 0000000000000001          B = ffffffffffffffff          Carry In =   0

Result = 0000000000000002          Zero =   0          Carry Out = 0


Case 2: Add(R-type)


ALU_Op =  10          funct7 =  0000000          funct3 =  000

Opcode = 0010

Time =   61
A = 0000000000000001          B = ffffffffffffffff          Carry In =   1

Result = 0000000000000000          Zero =   1          Carry Out = 1



Case 3: Subtract(R-type)
```

```
File Edit View Search Terminal Help
Case 0: Add(ALU_Op code)


ALU_Op =  00          funct7 =  0100111          funct3 =  111

Opcode = 0010

Time =  141
A = 7fffffffffffffff          B = 0000000000000001          Carry In =   0

Result = 8000000000000000          Zero =   0          Carry Out = 0



Case 1: Subtract(ALU_Op code)


ALU_Op =  01          funct7 =  0000000          funct3 =  111

Opcode = 0110

Time =  161
A = 7fffffffffffffff          B = 0000000000000001          Carry In =   0

Result = 7ffffffffffffffe          Zero =   0          Carry Out = 1



Case 2: Add(R-type)


ALU_Op =  10          funct7 =  0000000          funct3 =  000

Opcode = 0010

Time =  181
A = 7fffffffffffffff          B = 0000000000000001          Carry In =   1

Result = 8000000000000000          Zero =   0          Carry Out = 0
```

Case 3: Subtract(R-type)

ALU_Op = 10   funct7 = 0100000   funct3 = 000

Opcode = 0110

Time = 321
A = 7fffffffffffffff   B = 8000000000000000   Carry In = 0

Result = ffffffffffffffff   Zero = 0   Carry Out = 0

Case 4: And(R-type)

ALU_Op = 10   funct7 = 0000000   funct3 = 111

Opcode = 0000

Time = 341
A = 7fffffffffffffff   B = 8000000000000000   Carry In = 1

Result = 0000000000000000   Zero = 1   Carry Out = 0

Case 5: Or(R-type)

ALU_Op = 10   funct7 = 0000000   funct3 = 110

Opcode = 0001

Time = 361
A = 0000000000000000   B = 0000000000000000   Carry In = 0

Result = 0000000000000000   Zero = 1   Carry Out = 0

Case 0: Add(ALU_Op code)

ALU_Op = 00   funct7 = 0100111   funct3 = 111

Opcode = 0010

Time = 381
A = 0000000000000000   B = 0000000000000000   Carry In = 0

Result = 0000000000000000   Zero = 1   Carry Out = 0

Case 1: Subtract(ALU_Op code)

ALU_Op = 01   funct7 = 0000000   funct3 = 111

Opcode = 0110

Time = 401
A = 0000000000000000   B = 0000000000000000   Carry In = 0

Result = 0000000000000000   Zero = 1   Carry Out = 1

Case 2: Add(R-type)

ALU_Op = 10   funct7 = 0000000   funct3 = 000

Opcode = 0010

Time = 421
A = 0000000000000000   B = 0000000000000000   Carry In = 1

Result = 0000000000000000   Zero = 1   Carry Out = 0