Devices and Specifications

Overview

There are several basic components that you will need to implement to provide the functional units necessary to carry out instruction-level instructions. There are generally two steps here:

- Study the architecture-level instructions and start putting a design for our to carry out each instruction. There are 32 in total. For each instruction, identify the devices necessary to carry out the instruction and how they will be connected.
- Out of the exercise in the first step, you will have 32 basic data flows that will need to be implemented. Consolidate these into a design for the microarchitecture.
- Start the implementation. Implement each device according to the specification given in this document. Make the necessary connections according to the design.

Generally, it is a good idea to implement an abstract device that has the features and methods for:

- Providing data input ports.
- Providing one or more output latches.
- Implementing the necessary function of the device.
- Reacting to the clock signal.
- Reacting to the control signals.
- Implementing the means to connect its input.

Through inheritance, specialize the abstract device to implement the specific devices outlined in this document. Test each instruction on its own as you implement to manage the software development in a controlled manner.

The list of the devices follows. If you need a device that is not in the list, contact the instructor immediately.

Adder

Number of input ports: 2, 64-bit each
Number of output latches: 1, 64-bit
Control signal: N/A
Function: Adds the signed values present at the input ports and produces a signed value.
Cycles to complete: 1
Area: 400nm^2
Power: 0.5W

Shifter

Number of input ports: 2, 64-bit each
Number of output latches: 1, 64-bit
Control signal: 0x00 for right, 0x01 for left
Function: Shifts the signed value present at the first port by a number of bits equal to the value present at the second port.
Cycles to complete: 1
Area: 200nm^2
Power: 0.5W

Logic

Number of input ports: 2, 64-bit each
Number of output latches: 1, 64-bit
Control signal: 0x00 for NOT, 0x01 for AND, 0x10 for OR, 0x11 for XOR.
Function: Perform bit-wise logical function on the values present at the input ports as specified by the control registers.
Cycles to complete: 1
Area: 600nm^2
Power: 0.75W
Note: In the case if the control signal is 0x00 (NOT operation), the second input port is ignored.

Multiplier

Number of input ports: 2, 64-bit each
Number of output latches: 1, 64-bit
Control signal: N/A
Function: Multiplies the signed values present at the input ports and produces a signed value.
Area: 2000nm^2
Power: 1.5W
Cycles to complete: 3

Divider

Number of input ports: 2, 64-bit each
Number of output latches: 1, 64-bit
Control signal: N/A
Function: Divides the signed value present at the first input port by the value present at the second port and produces a signed value.
Cycles to complete: 8
Area: 5000nm^2
Power: 1.0W

Comparator

Number of input ports: 2, 64-bit each
Number of output latches: 1, 64-bit
Control signal: N/A
Function: Compares the signed value present at the first input port with the value present at the second port. Produces 0 if they are equal, 1 otherwise.
Cycles to complete: 1
Area: 400nm^2
Power: 0.5W

Two's Complement

Number of input ports: 1, 64-bit
Number of output latches: 1, 64-bit
Control signal: N/A
Function: Produces the two's complement (-a) of the value at the input port.
Cycles to complete: 1
Area: 200nm^2
Power: 0.25W

Register File

Number of input ports: 2, 64-bit each
Number of output latches: 2, 64-bit
Control signal: 0x00 NOP, 0x01 R-, 0x10 RR, 0x11 W
Function: Stores an array of 32 registers of 64-bit each. Results varies according to the control signals:

- 0x00 NOP: The register file is idle
- 0x01 R-: Direct the value of the register specified by the value on the first input into the first latch.
- 0x10 RR: Direct the values of the registers specified by the value on the first and second input ports, to the first and second output latch, respectively.
- 0x11 W: Store the value on the first input port into the register specified by the value on the second register.

Cycles to complete: 1
Note: The register file allows up to 2 concurrent reads but only one write. Note that it is an error to specify the same register in an RR operation and should not be allowed.
Area: 20000nm^2
Power: 4W

Multi-ported Register File

Number of input ports: 4, 64-bit each
Number of output latches: 4, 64-bit
Control signal: Expand the list to allow all the operations that you need by extending the control signal schema of the smaller register file.

Cycles to complete: 1
Note: The register file allows up to 4 concurrent reads but only two concurrent writes. Combinations of up to two reads and one write is possible, and so on. Note that it is an error to specify the same register in more than one operation and should not be allowed.
Area: 25000nm^2
Power: 6W

Multiplexer

Number of input ports: 4, 64-bit each
Number of output latches: 1, 64-bit
Control signal: 0x00, 0x01, 0x10 and 0x11
Function: Routes the input on the input port specified by the control signal to the output latch.
Cycles to complete: 0.5
Area: 500nm^2
Power: 0.25W
Note: You can stage two levels of these multiplexers to produce a 16x1 multiplexer. Useful for routing.

Demultiplexer

Number of input ports: 1
Number of output latches: 4
Control signal: 0x00, 0x01, 0x10 and 0x11
Function: Routes the input on the input port to the output latch specified by the control signal.
Cycles to complete: 0.5
Area: 500nm^2
Power: 0.25W
Note: You can stage two levels of these demultiplexers to produce a 1x16 multiplexer. Useful for routing.

Register

Number of input ports: 1, 64-bit
Number of output latches: 1, 64-bit
Control signal: N/A
Function: Holds a value in an individual register outside the purview of the register file.
Cycles to complete: 0.5
Area: 200nm^2
Power: 0.05W

Control Array(m x n)

Number of input ports: n, m-bit wide each.
Number of output latch: 1, m-bit wide.
Control signal: N/A

Function: An array of n registers, each m-bit wide. Used to implement control. At every clock signal, the output latch receives the value in the next register (modulo n). The output of this latch is connected to all the control signals of all devices. This is how data is set to flow and this is where the microarchitecture receives the necessary commands to execute the architecture-level instructions.
Cycle to complete: 1
Area: $1nm^2$/bit
Power: 0.001W/bit

Add4

Number of input ports: 1, 32-bit
Number of output latches: 1, 32-bit
Control signal: N/A
Function: Adds 4 to the unsigned value present at the input port 0 and produces an unsigned value. This is useful for manipulating the program counter.
Cycles to complete: 1
Area: $100nm^2$
Power: 0.1W/bit