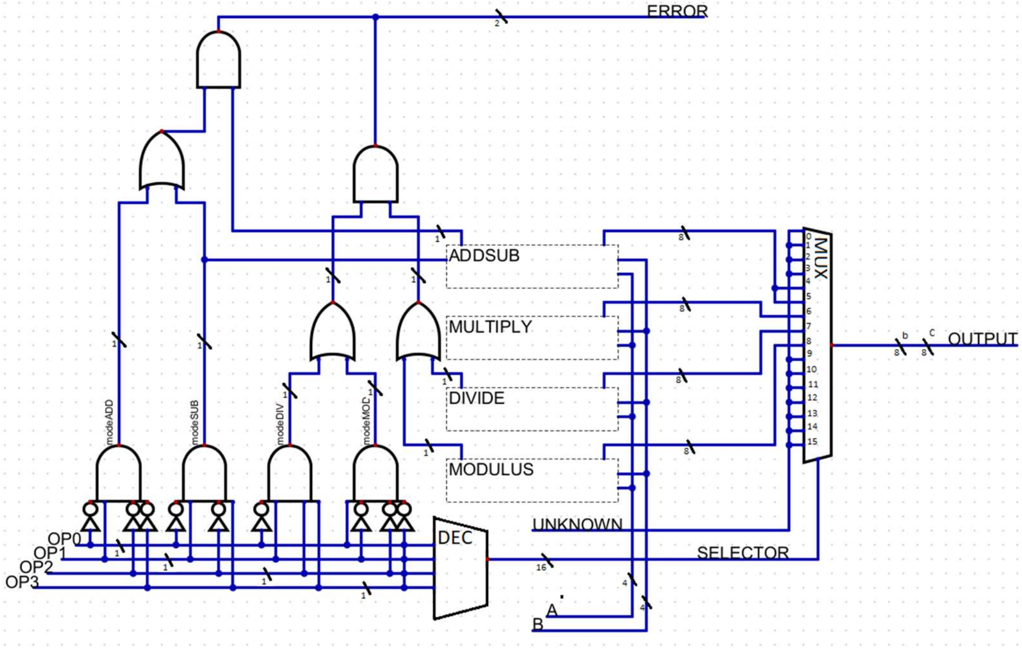
Digital Logic Project 2 System Design

Mason Doherty | The Verilogians | 3/3/2024

Top Level Diagram (Sample Code)



System Design (Sample Code):

2.1 Inputs

These are the inputs to the circuit. They can be found in the top left corner of the diagram.

- A: is a 4-bit input used for the calculations inside the modules (ADDSUB, MULTIPLY, DIVIDE, MODULUS)
- B: is a 4-bit input used for the calculations inside the modules (ADDSUB, MULTIPLY, DIVIDE, MODULUS)
- Opcode: is a 4-bit input used to determine the mode and is fed into the MUX to create a one hot number.

2.2 Outputs

These are the final outputs of the circuit. This includes the channel selected in MUX and the error code.

- C: is an 8-bit output that comes from the output of the MUX.
- error: is a 2-bit output needed to determine if there was an error in
- the calculations of ADDSUB, DIVISION, OR MODULUS.

2.3 Interfaces

The Wires and the Registers

1. INPUTS:

- A: 4-bit wire that carries the input A to the modules.
- B: 4-bit wire that carries the input B to the modules.
- Opcode: Wires that carry the opcode to the mode gates and the decoder.

2. OUTPUTS:

- error: 2-bit wire that carries the error to the output.
- C: 8-bit wire that carries the final output.
- outputADDSUB: 8-bit wire that carries the output of the ADDSUB module to the error gates.
- outputDIV: 8-bit wire that carries the output of the DIVIDE module to the error gates.
- outputMOD: 8-bit wire that carries the output of the MODULUS module to the error gates.
- outputMUL: 8-bit wire that carries the output of the MULTIPLY module to the error gates.

3. INTERNAL WIRES:

- clk: used for the clock not implemented in this project.
- rst: reset not used in this project.
- channels: 16 8-bit channels for the MUX.
- select: 16-bit one hot number that is carried to the MUX.
- b: 8-bit output of the MUX.
- unknown: 8-bit wire that connects to the unused MUX channels.

4. ERRORS:

- ADDerror: 1-bit output of the ADDSUB module that tells if there was an error.
- DIVerror: 1-bit output of the DIVISION module that tells if there was an error.
- MODerror: 1-bit output of the MODULUS module that tells if there was an error.

5. MODES:

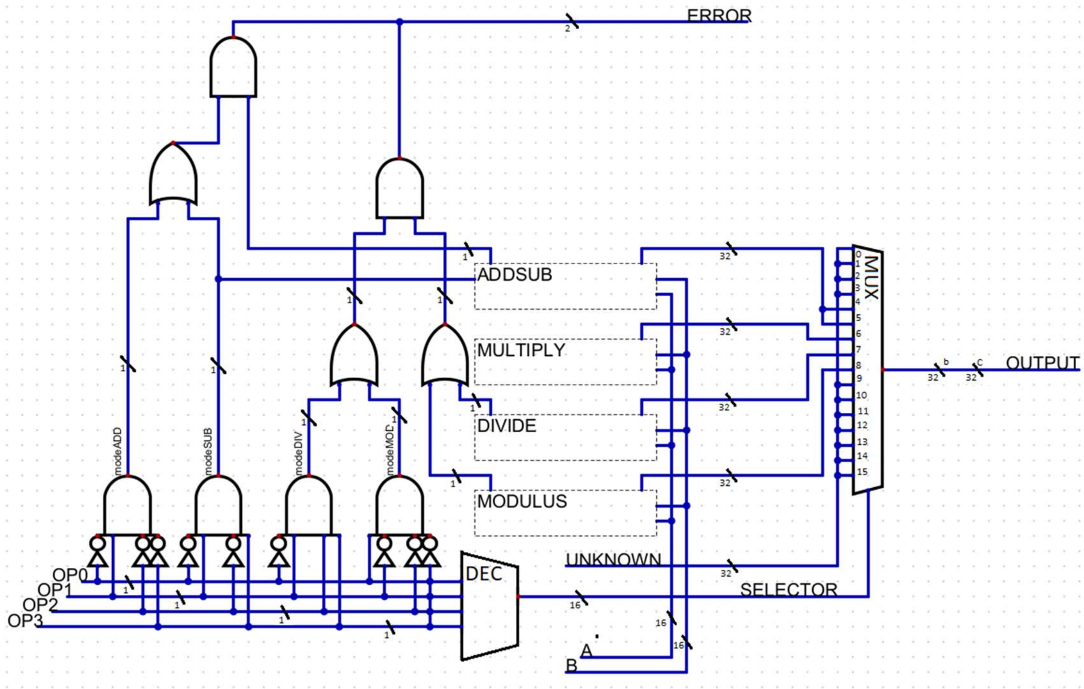
- modeADD: 1-bit wire coming out of the AND gate that determines whether the ADD error should be sent to output.
- modeSUB: 1-bit wire coming out of the AND gate t that determines whether the SUB error should be sent to output and changes the ADDSUB module to subtraction.
- modeDIV: 1-bit wire coming out of the AND gate that determines whether the DIV error should be sent to output.
- modeMOD: 1-bit wire coming out of the AND gate that determines whether the MOD error should be sent to output.

2.4 Parts

These are the modules that perform the calculations.

- DEC: Takes in the 4-bit opcode and creates a 16bit one hot number.
- MUX: Takes in 16 8-bit numbers (unknown, outputADDSUB, outputDIV, outputMOD, outputMUL) and the 16-bit Selection and outputs an 8-bit value, “b”.
- ADDSUB: Takes in the 4-bit A and B and outputs 8-bit outputADDSUB, and 1-bit ADD error.
- MULTIPLY: Takes in the 4-bit A and B and outputs 8-bit outputMUL.
- DIVISION: Takes in the 4-bit A and B and outputs 8-bit outputDIV, and 1-bit DIVerror.
- MODULUS: Takes in the 4-bit A and B and outputs 8-bit outputMOD, and 1-bit MODerror

Top Level Diagram (Expanded Combinational Logic)



System Design (Expanded Combinational Logic):

2.1 Inputs

These are the inputs to the circuit. They can be found in the top left corner of the diagram.

- A: is a 16-bit input used for the calculations inside the modules (ADDSUB, MULTIPLY, DIVIDE, MODULUS)
- B: is a 16-bit input used for the calculations inside the modules (ADDSUB, MULTIPLY, DIVIDE, MODULUS)
- Opcode: is a 4-bit input used to determine the mode and is fed into the MUX to create a one hot number.

2.2 Outputs

These are the final outputs of the circuit. This includes the channel selected in MUX and the error code.

- C: is a 32-bit output that comes from the output of the MUX.
- error: is a 2-bit output needed to determine if there was an error in
- the calculations of ADDSUB, DIVISION, OR MODULUS.

2.3 Interfaces

The Wires and the Registers

1. INPUTS:

- A: 16-bit wire that carries the input A to the modules.
- B: 16-bit wire that carries the input B to the modules.
- Opcode: Wires that carry the opcode to the mode gates and the decoder.

2. OUTPUTS:

- error: 2-bit wire that carries the error to the output.
- C: 32-bit wire that carries the final output.
- outputADDSUB: 32-bit wire that carries the output of the ADDSUB module to the error gates.
- outputDIV: 32-bit wire that carries the output of the DIVIDE module to the error gates.
- outputMOD: 32-bit wire that carries the output of the MODULUS module to the error gates.
- outputMUL: 32-bit wire that carries the output of the MULTIPLY module to the error gates.

3. INTERNAL WIRES:

- clk: used for the clock not implemented in this project.
- rst: reset not used in this project.
- channels: 16 32-bit channels for the MUX.
- select: 16-bit one hot number that is carried to the MUX.
- b: 32-bit output of the MUX.
- unknown: 32-bit wire that connects to the unused MUX channels.

4. ERRORS:

- ADDerror: 1-bit output of the ADDSUB module that tells if there was an error.
- DIVerror: 1-bit output of the DIVISION module that tells if there was an error.
- MODerror: 1-bit output of the MODULUS module that tells if there was an error.

5. MODES:

- modeADD: 1-bit wire coming out of the AND gate that determines whether the ADD error should be sent to output.
- modeSUB: 1-bit wire coming out of the AND gate t that determines whether the SUB error should be sent to output and changes the ADDSUB module to subtraction.
- modeDIV: 1-bit wire coming out of the AND gate that determines whether the DIV error should be sent to output.
- modeMOD: 1-bit wire coming out of the AND gate that determines whether the MOD error should be sent to output.

2.4 Parts

These are the modules that perform the calculations.

- DEC: Takes in the 4-bit opcode and creates a 16bit one hot number.
- MUX: Takes in 16 32-bit numbers (unknown, outputADDSUB, outputDIV, outputMOD, outputMUL) and the 16-bit Selection and outputs a 32-bit value, “b”.
- ADDSUB: Takes in the 16-bit A and B and outputs 32-bit outputADDSUB, and 1-bit ADD error.
- MULTIPLY: Takes in the 16-bit A and B and outputs 32-bit outputMUL.
- DIVISION: Takes in the 16-bit A and B and outputs 32-bit outputDIV, and 1-bit DIVerror.
- MODULUS: Takes in the 16-bit A and B and outputs 32-bit outputMOD, and 1-bit MODerror.

3.4 Question:

If the bit-size of all other components and lines are being updated, why is the bit-size of the opcode unchanged?

- The opcode remains unchanged because even though the size of the inputs and outputs have changed, the quantity has not. An 8-bit system requires the same number of inputs and outputs as a 32-bit system.