The Waffle House

System Design

November 16, 2018

Parts List

1-Bit D Flip-Flop: takes in 1-bit input values of clk and D, and when clk is on its positive edge, the 1-bit output value of Q is set equal to D.

8-Bit D Flip-Flop: takes in 1-bit input value of clk and and 8-bit input value of D, and when clk is on its positive edge, the 8-bit output value of Q is set equal to D.

Half-Adder: takes in 1-bit inputs a and b; XORs them into the output sum value, ANDs them into the output c_out value

Full-Adder: takes in 1-bit inputs a, b, & c_in and performs addition on them by instantiating two half-adders; calculates c_out by ORing the respective c_out values of the two half adders

Mux2: takes in two 8-bit value inputs and one 1-bit select input; sets the value of the 8-bit output to the chosen input value based on the given 1-bit select input

Mux10: takes in ten 8-bit value inputs and one 10-bit, one-hot select input; sets the value of the 8-bit output to the chosen input value based on the given one-hot selecte input

AND Gate: takes in two 1-bit values and returns the result of ANDing them

OR Gate: takes in two 1-bit values and returns the result of ORing them

XOR Gate:takes in two 1-bit values and returns the result of XORing them

Input List

int1: the first 8-bit integer used to compute operations in the ALU

int2: the second 8-bit integer used to computer operations in the ALU

rst: a 1-bit value that triggers the ready state when set to 1

opcode: a 3-bit integer that selects the mode of operation to compute

error: a 1-bit integer that determines whether the system is in an error state or not

clk: a 1-bit pulse signal that signals the ALU to perform a computation when it is on its positive edge

Output List

output: the value of the operation that was just performed; also acts as an accumulator

status: signifies whether the ALU is currently in a state that is done or has an error (carry-over warning)

Interface List

carry_out_add: output of 8-bit adder-subtractor module in the addition mode; lines to error wire and status

carry_out_sub: output of 8-bit adder-subtractor module in the subtraction mode; lines to error wire and status

select (s): retrieved from selector based on opcode, rst, and error values; lines to Mux10 part

next_state: Mux10 chooses next state; lines to 8-bit wire

current_state: output of selected calculation; lines to second input of the Mux2 part

first_operation: output of the 1-bit D Flip-Flop determining the completion of the first output; lines into AND gate

last_output: 1-bit wire output from AND gate determining whether the first operation has been completed or not; lines to Mux2 as a selector bit

error: result of there being a carry out during addition or subtraction

one: output of Mux2; the chosen integer between the first integer and the last output to perform operations with

add: 8-bit result of addition; output of 8-bit adder subtractor in add mode; lines to Mux10

sub: 8-bit result of subtraction; output of 8-bit adder subtractor in subtract mode; lines to Mux10

sl: 8-bit result of shifting left behaviorally; lines to Mux10

sr: 8-bit result of shifting right behaviorally; lines to Mux10

_and: 8-bit result of shifting ANDing behaviorally; lines to Mux10

_or: 8-bit result of shifting ORing behaviorally; lines to Mux10

_xor: 8-bit result of shifting XORing behaviorally; lines to Mux10

_not: 8-bit result of shifting NOTing behaviorally; lines to Mux10

Modules List

Test_Bench: the workspace; manipulates values of clk, rst, int1, and int2 to test various features of the ALU

Four_Bit_Adder_Subtractor: instantiates full adders to add two 4-bit integers and receive its carry out

Eight_Bit_Adder_Subtractor: instantiates two *four_bit_adder_subtractor*s to add two 8-bit integers and receive its carry out

Selector: decides on a 10-bit one-hot value based on its given values of the opcode, rst, and error

SHIFT_LEFT: (behavioral) bitwise left shift int1 or accumulator

SHIFT_RIGHT: (behavioral) bitwise right shift int1 or accumulator

AND: (behavioral) logical AND int1 or accumulator with int2

OR: (behavioral) logical OR int1 or accumulator with int2

XOR: (behavioral) logical XOR int1 or accumulator with int2

NOT: (behavioral) logical NOT int1 or accumulator

Modes List (States)

READY: waiting for input; outputs 8-bit 0

ADD: state defining the addition of two 8-bit integers; susceptible to error

SUB: state defining the subtraction of one 8-bit integer from another; susceptible to error

SHIFT_LEFT: state defining the shifting left of an 8-bit integer

SHIFT_RIGHT: state defining the shifting right of an 8-bit integer

AND: state defining the logical ANDing of two 8-bit integers

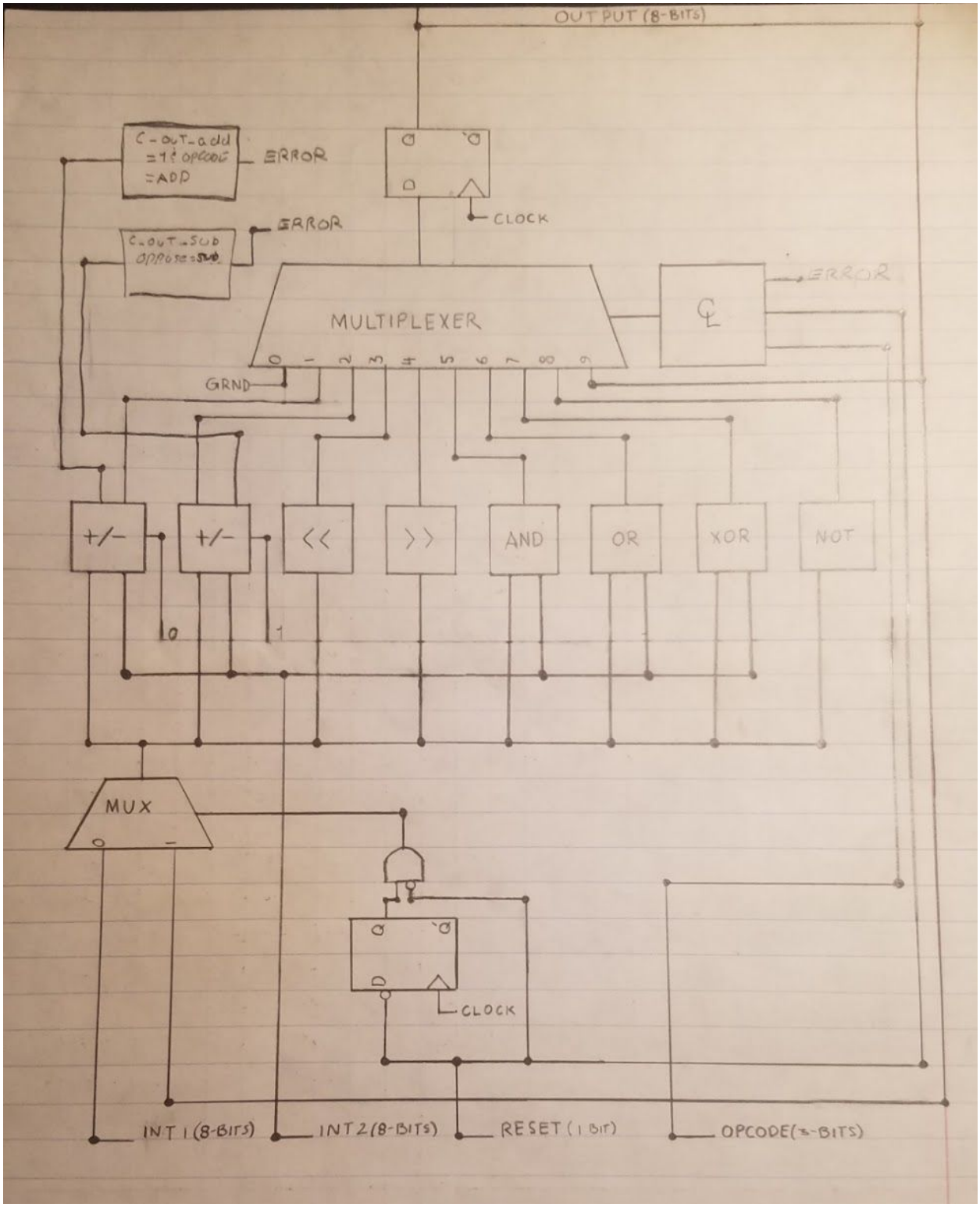
OR: state defining the logical ORing of two 8-bit integers

XOR: state defining the logical XORing of two 8-bit integers

NOT: state defining the logical ANDing of two 8-bit integers

ERROR: state defining the warning of an operation due to carry out (no overflow because the ALU is dealing with unsigned arithmetic); retains the value of the last output

Circuit Diagram



State Machine