ALU design notes

8-16-2021

After doing some research, I have found my previous suspicions to be true. There are ICS with higher chip complexity that are cheaper than ICS with lower chip complexity. This means that I could possibly reduce the cost of this project by using more complex chips. I will not be re-analyzing the chips I used for the alu because that seems very time consuming, but from here on out I will be looking for the most economical chip option, rather than just using the chips I used for the 8 bit computer.

8-16-2021

It seems that using eeproms to implement the multiplication function is far too expensive, so here are some alternatives

-Using ram chips which are programmed at computer startup  
Pros:  
possibly cheaper  
Cons:  
more complexity  
startup wait time  
  
-Using microprocessor which performs calculations to find answer  
Pros  
possibly cheaper  
Cons:  
Would take longer than standard circuit ←(best for peripheral devices which can afford some delay )  
  
-finding eeprom with 16 address lines  
Pros:  
might be cheaper since everything is packed in the same chip, therefore not needing extra chip casing.  
Also, things are cheaper when bought in bulk, so having more circuitry per chip may make the overall thing cheaper. Or you are buying more complex chips designed for modern computers, which have been simplified ( in terms of design and manufacturing ) so as to make them cheaper for todays demand.

Cons:  
possibly more expensive than other solutions proposed.  
  
-just using multiplication chips, if possible  
pros:  
possibly cheaper  
Cons:  
might not exist  
  
-building discrete logic function yourself  
pros:  
 -maybe cheaper, but I doubt it.

Cons:   
possibly overwhelming complexity

8-13-2021

In the v2 version I have accidentally given the OP1 and OP2 buses 8 bits instead of 16 bits.

8/5/2021

Now making the v2 of my alu, I realize that the previous design neglects to add inverters to the front of the and gates; nullifying the circuitry preceding it. This reveals another advantage of creating a simulation before deploying a prototype; error checking.

2/25/2021

Here is a new idea for the bit-shifter. The alu will accept operand 1 as the value to be shifted and operand 2 as the value to shift by. Starting from the lsb, a nibble of information will be fed from both op1 and op2 into an eeprom, which will be programmed with a proper output for the current power of 2. The results of each eeprom are then fed into another alu which add up the intermediary results of the operation.

I am not sure why I thought the bit shifter was a good idea for binary multiplication, it would only work for powers of two. Very embarrassing!

I wanted to design my own shift register just for fun, so I did.

In the bit shifter only four tri-state buffers are present, but as the connections are

red this symbolizes 16 tri-state buffers.

Also, although the decoders have inverted output, the design calls for non inverted output.

I could only find chips with inverted output.  
  
  
  
  
  
Strangely the chip I used for the flags register does not have any read signals, so I am going to assume that its clock signal works as its enable.   
  
Because the status of the flag register needs to be saved in the case of an interrupt, the registers output will be connected to both the control unit and to the data in portion of the data-bus.  
The ZN flags will not be separate from the CO flags. They will be on the same chip.

I am thinking of representing each module with a different color to indicate that a connection attaches to that module in that way. To distinguish nibbles vs bytes, a connection will either start red or blue and then segment, to be continued in the new color. Since multisim doesn’t allow connections between two differently colored connections, this segment will be empty space.   
  
  
  
  
Here are some color coding ideas:  
  
Green: control unit  
Dark Yellow: ALU  
purple: interrupt module  
orange: data-bus  
  
  
  
I think that I am done with the preliminary design of the ALU, I might need to show my design to a teacher or professional to check that I haven’t left anything out but so far I am feeling good about it.