

Homework 3: Combinational Chips (Arithmetic+ALU)

Objective:

Build the gates described in Chapter 2 (see table below), which will test your understanding of Boolean Logic and Arithmetic, building the gates that do so.

Grading method:

If the chip passes *all* the tests specified in the supplied test script, it receives 60% of the grade. 30% goes to it being well built (the lowest number of chips to implement), with the remaining 10% going towards documentation provided for each chip. Generally speaking, we prefer implementations that *use as few chip parts as possible*, even if it implies a less efficient chip design (in terms of # of AND/OR/NOT chips). Higher-level chips are considered as one chip part (ex. Mux, DMux, Or8Way, etc.)

What do you turn in?

What do you turn in?

A Word document (or PDF) with screen shots of each of the working (or not) logic gates. You should also upload the documentation.pdf file (see Documentation Instructions for guidelines on how to do this) per Project Submission Guidelines.

NOTE: the HDL code you write for the ALU (ALU.hdl) will be used in BOTH test scripts (ALU-basic.tst and ALU.tst). Please take a screen shot of the output from both tests. This is to maximize partial credit for you in case you aren't able to get the status output flags working.

<i>Chip</i>	<i>Working?</i>	<i>Well built?</i>
HalfAdder	/ 10	/ 5
FullAdder	/ 10	/ 5
Add16	/ 10	/ 5
Inc16	/ 10	/ 5
ALU without status outputs (ALU-basic.tst)	/ 15	/ 5
ALU + full test (ALU.tst)	/ 5	/ 5
Subtotal	/ 60	/ 30
Documentation	/	10

See <http://nand2tetris.org/02.php> for some tips/resources/tools (note that the assignment on the website is substantially different from the assignment that is described above, if you need clarification email your instructor. You will be graded based on this documents requirements).