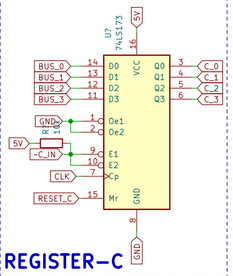
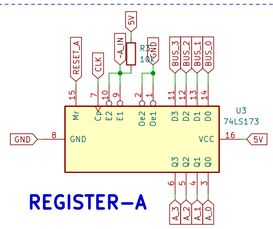
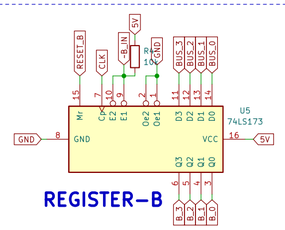
REGISTER–C



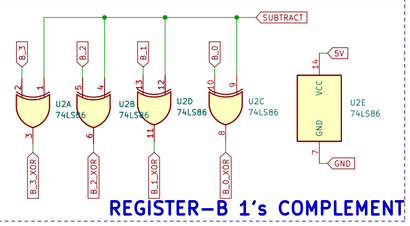
REGISTER–A



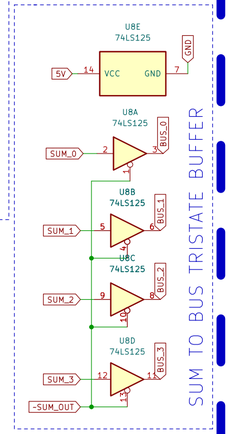
REGISTER–B



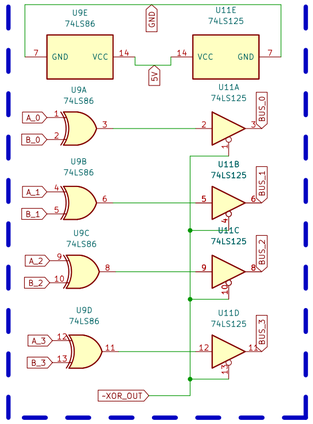
REGISTER–B 1's COMPLEMENT



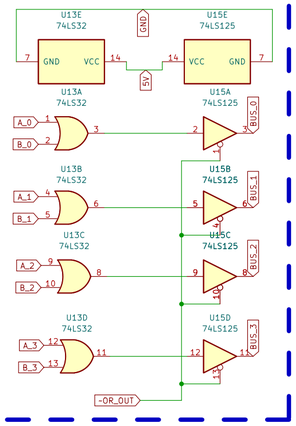
SUM TO BUS TRISTATE BUFFER



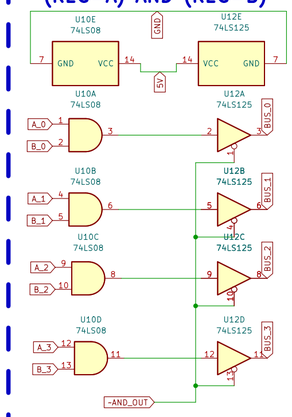
(REG–A) XOR (REG–B)



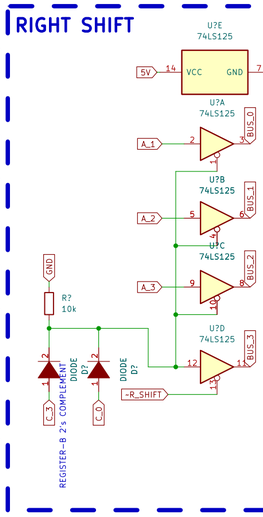
(REG–A) OR (REG–B)



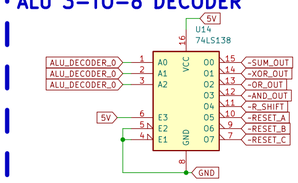
(REG–A) AND (REG–B)



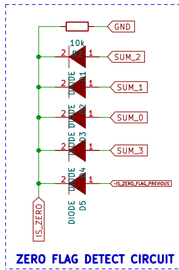
RIGHT SHIFT



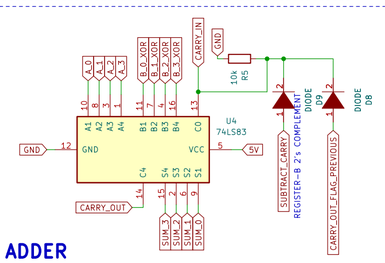
ALU 3–TO–8 DECODER



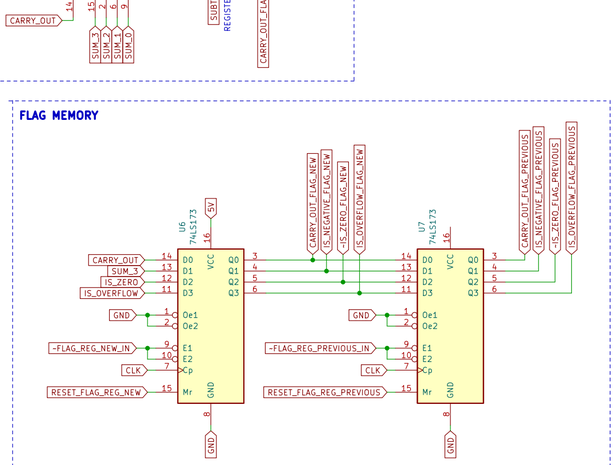
ZERO FLAG DETECT CIRCUIT



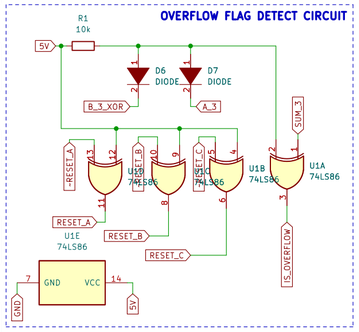
BINARY ADDER



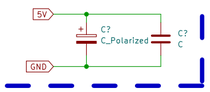
FLAG MEMORY



OVERFLOW FLAG DETECT CIRCUIT



INPUT CAPACITOR



Logic Gate	Symbol	Description	Boolean
AND		Output is at logic 1 when, and only when all its inputs are at logic 1 otherwise the output is at logic 0	X = A•B
OR		Output is at logic 1 when one or more are at logic 1 If all inputs are at logic 0 output is at logic 0	X = A+B
NAND		Output is at logic 0 when and only when all its inputs are at logic 1 otherwise the output is at logic 1	$X = \overline{A \cdot B}$
NOR		Output is at logic 0 when one or more of its inputs are at logic 1 If all the inputs are at logic 0 the output is at logic 1	$X = \overline{A+B}$
XOR		Output is at logic 1 when one and Only one of its inputs is at logic 1. Otherwise it is logic 0.	X = A⊕B
XNOR		Output is at logic 0 when one and only one of its inputs is at logic1.Otherwise it is logic 1. Similar to XOR but inverted.	$X = \overline{A \oplus B}$
NOT		Output is at logic 0 when its only input is at logic 1, and at logic 1 when its only input is at logic 0 That's why it is called and INVERTER	$X = \overline{A}$

Designed by Erdem Canaz
EE348– 4 Bit computer project
Sheet: /
File: ALU design.kicad_sch

Title: ALU design

Size: A2 | Date: 2023–03–19 | **Rev: v0**
KiCad E.D.A. kicad (6.0.11) | Id: 1/1