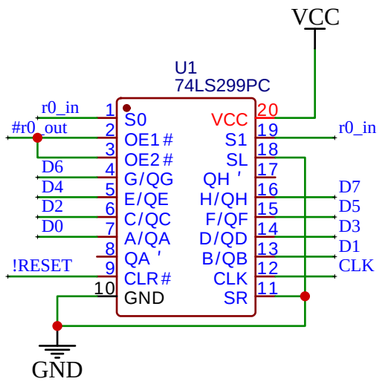
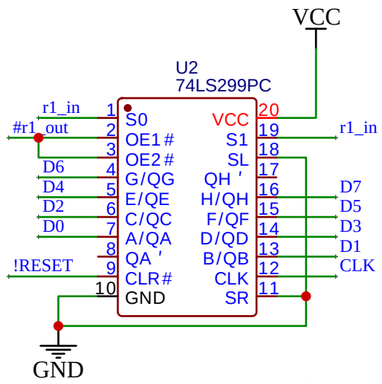
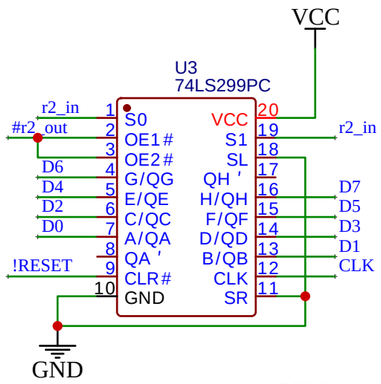
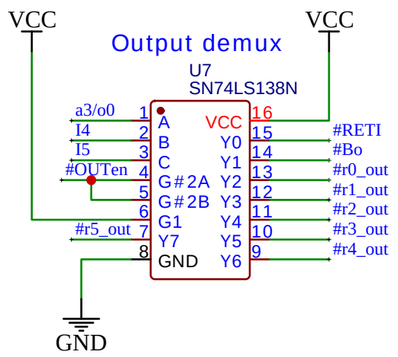
```
# ALU format:
# 0b xxxx a3 a2 a1 a0
# Reg select format:
# 0b i2 i1 o2 o1 o0 i0 x x

# 0b I7 I6 I5 I4 I3 I2 I1 I0

# o0 = XOR(X-sig,x1) = a3
# i0 = XOR(X-sig,a2)
```

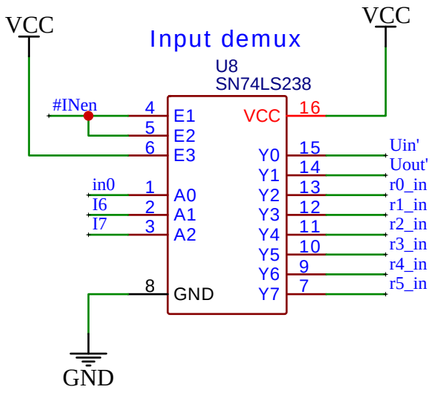
reg_names_OUT = ["CLEAR INT STATUS","B","r0","r1","r2","r3","r4","r5"] reg_names_IN = ["U","_Uout","r0","r1","r2","r3","r4","r5"]

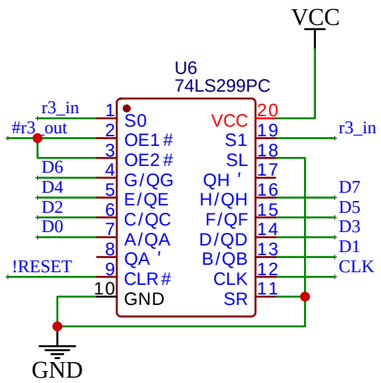


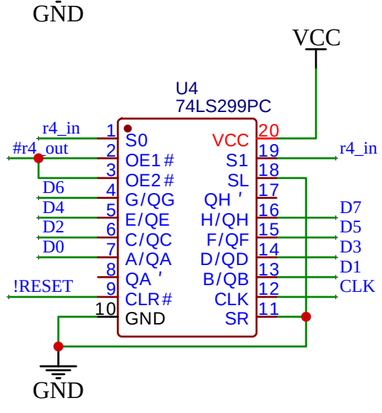


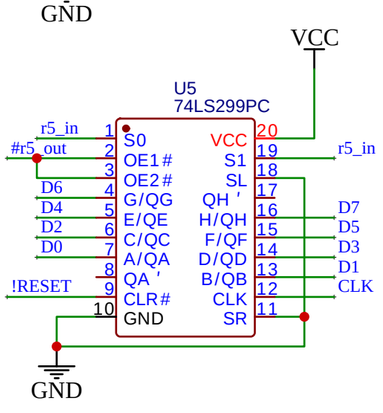


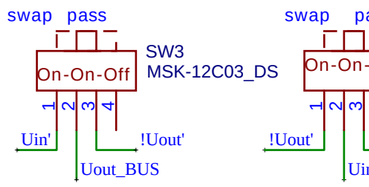


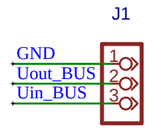


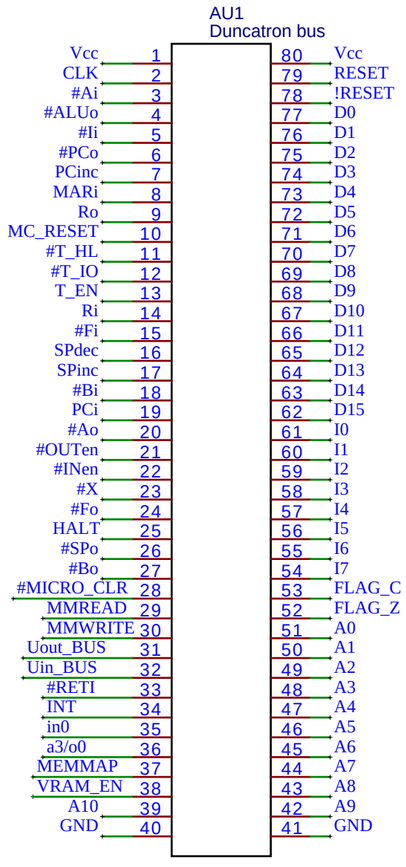




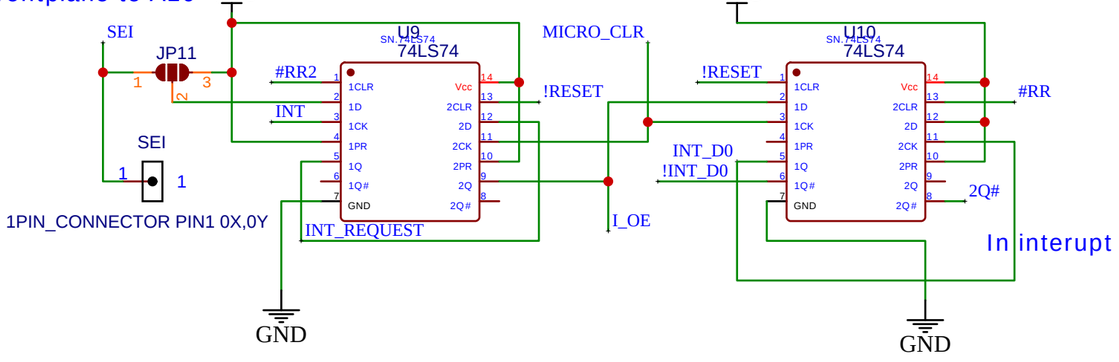


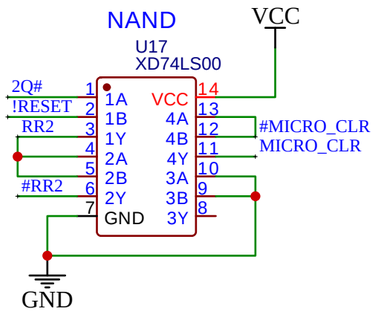


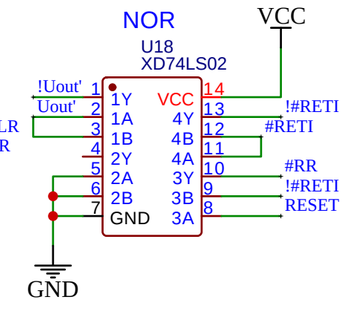


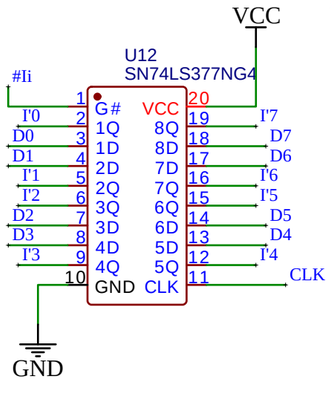


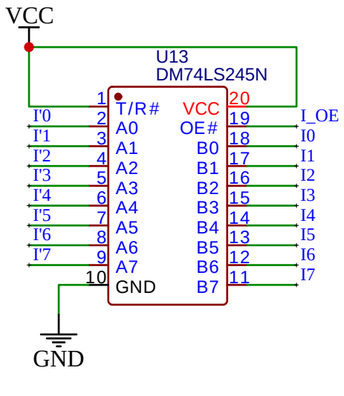
A8/9/10 is not output onto bus in v1.0 of memory map
Could use a wire on backplane from SEI pin to A8/9
or frontplane to A10

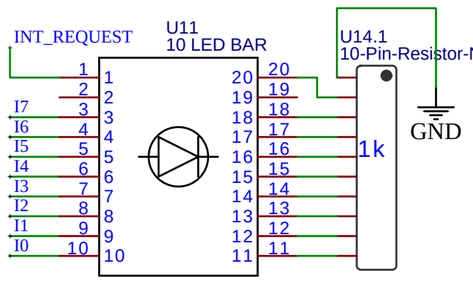








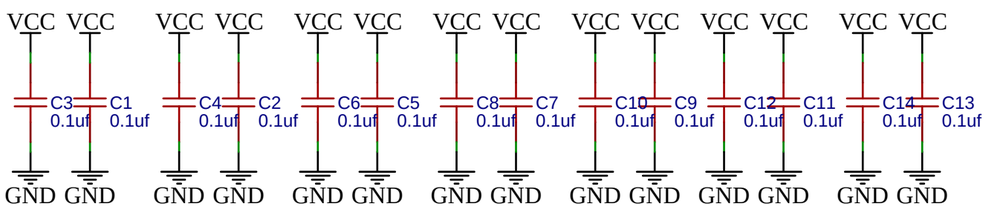


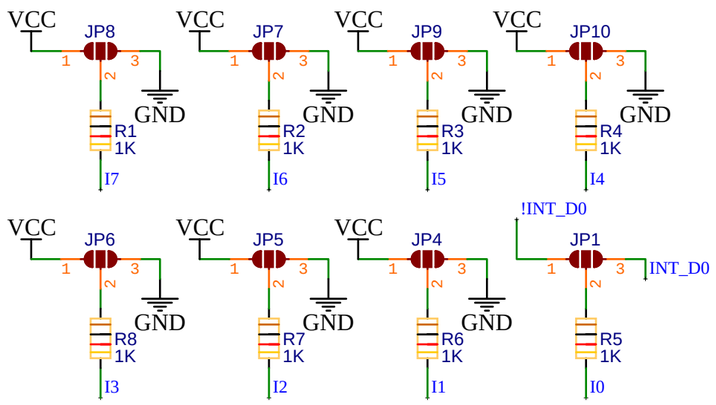


#RR is low when #RETI is low
or #RESET is low

#RR2 is low when 2Q is set
or RESET is high
=NOT(NAND(2Q#,!RESET))

BREAKOUT SEI to a header for
future memory mapping ("Control register")
Bit should be pulled down so interupts
are disabled by default! Means can't interupt
any important init code. OR USE FREE BUS CONN?





This board implements six general purpose hardware registers r0-r6
The instruction register (and interrupt logic)
Interrupts run PUSH_PC+1 and INT in sequence. Opcode can be hardwired with solder jumpers
Transfer register also provided for moving data between hi and low 8-bit buses.
Can also perform a SHIFT RIGHT if X is enabled (HIGH) and LOAD is HIGH
(T_EN=1 and #T_IO =0)
Possess LED bank instruction register (not output selection?)

TITLE: REG schematic	REV: 1.0
EasyEDA	Company: Your Company
	Sheet: 1/1
	Date: 2024-01-03 Drawn By: Duncan Scott