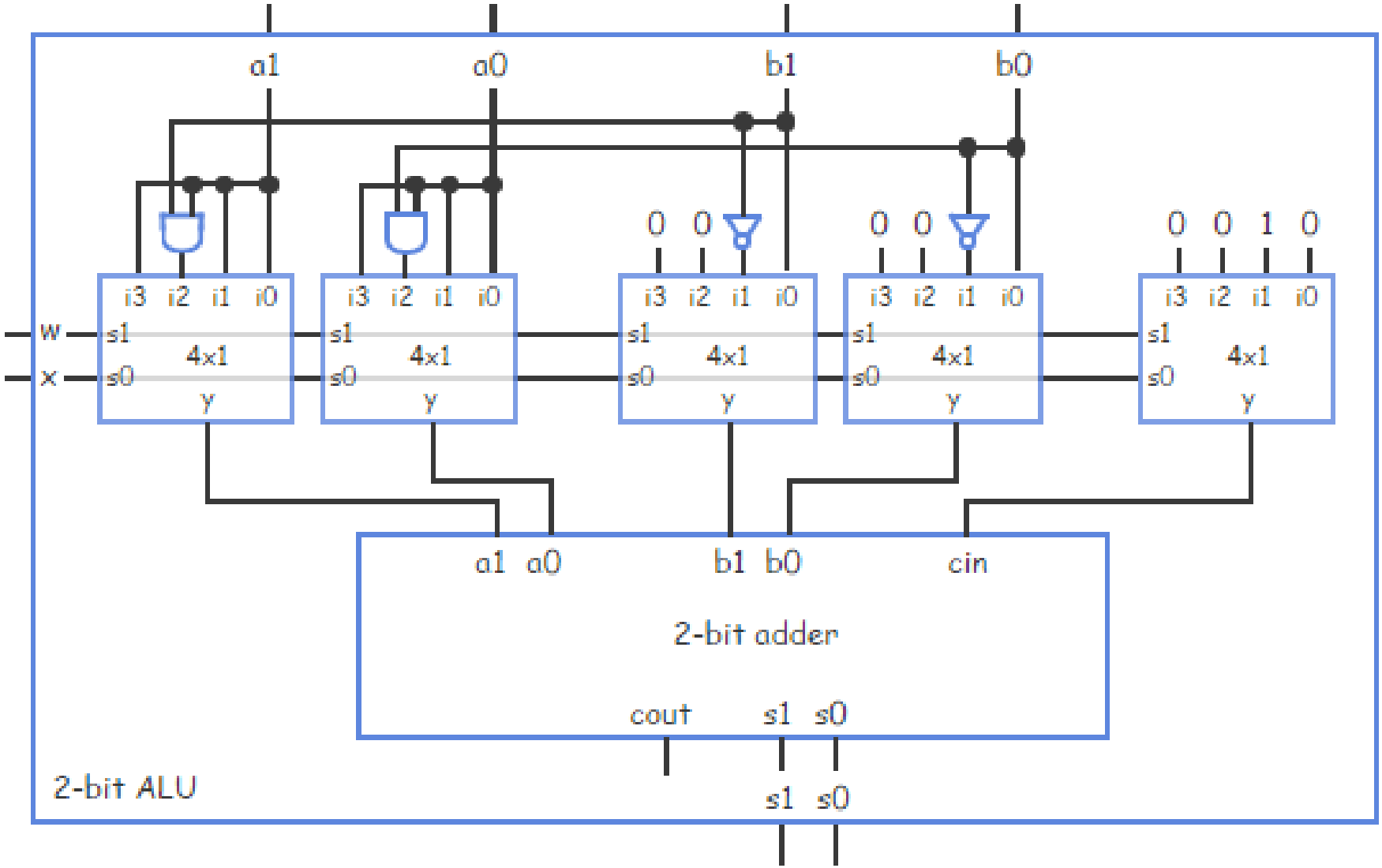
Introduction to Digital Logic

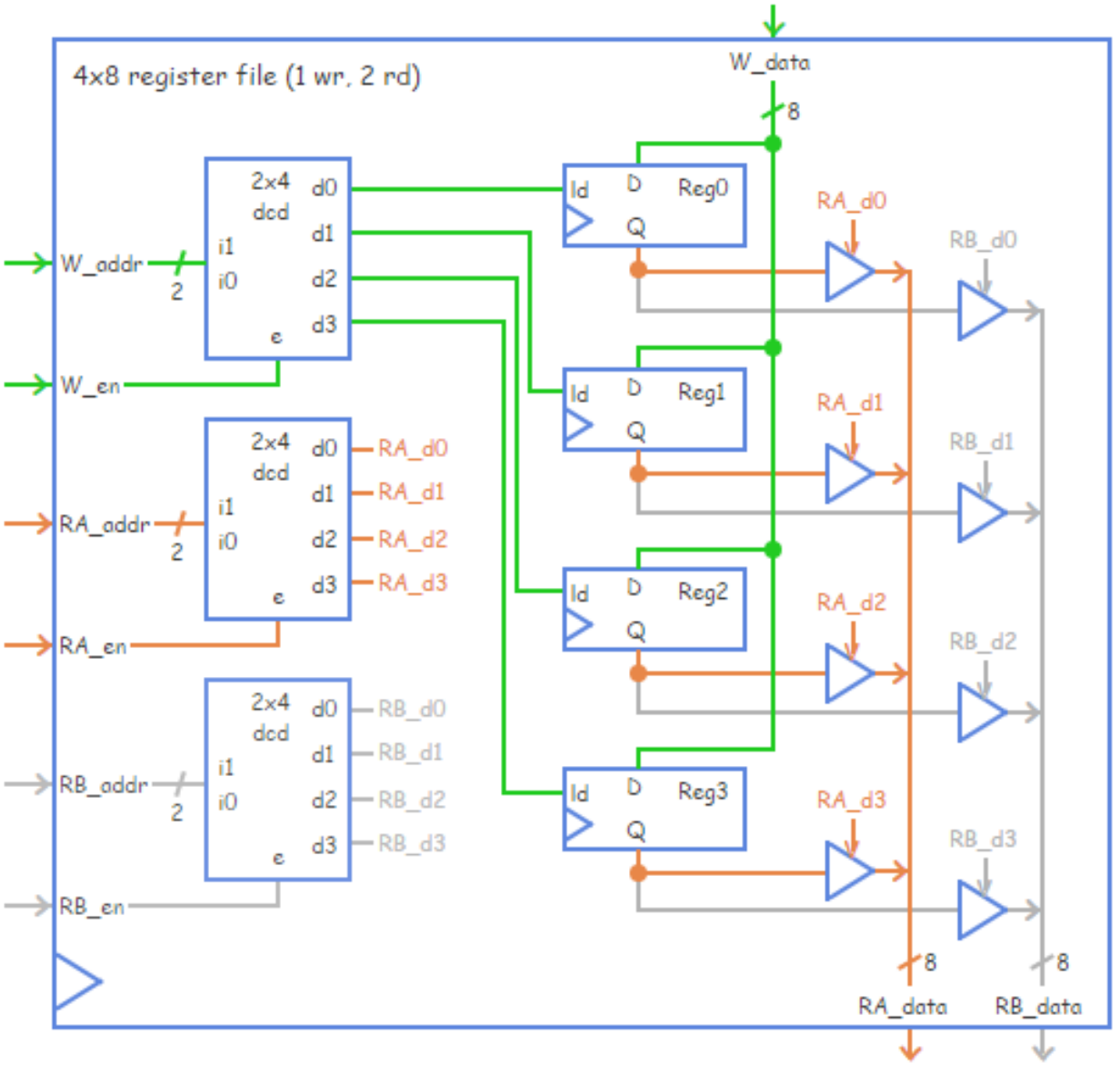
- ALUs

Doug Shook

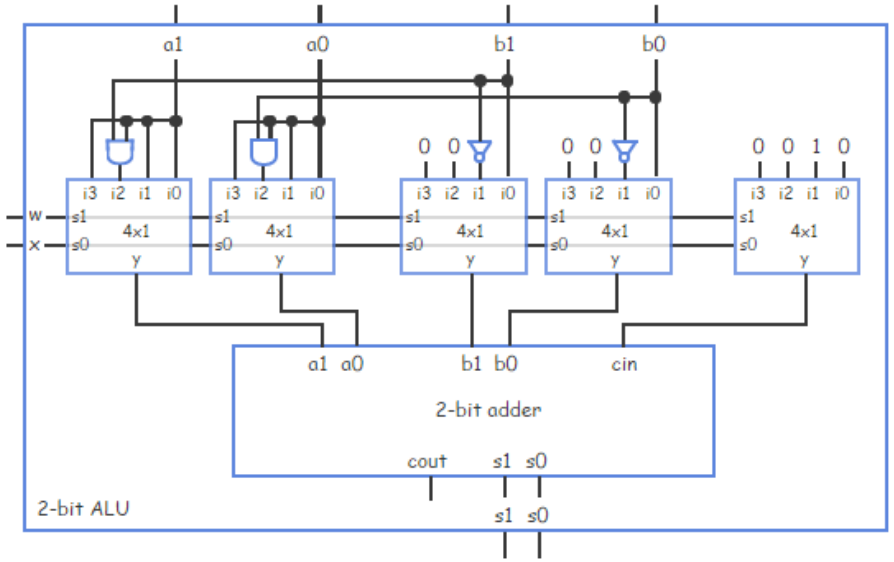
ALU

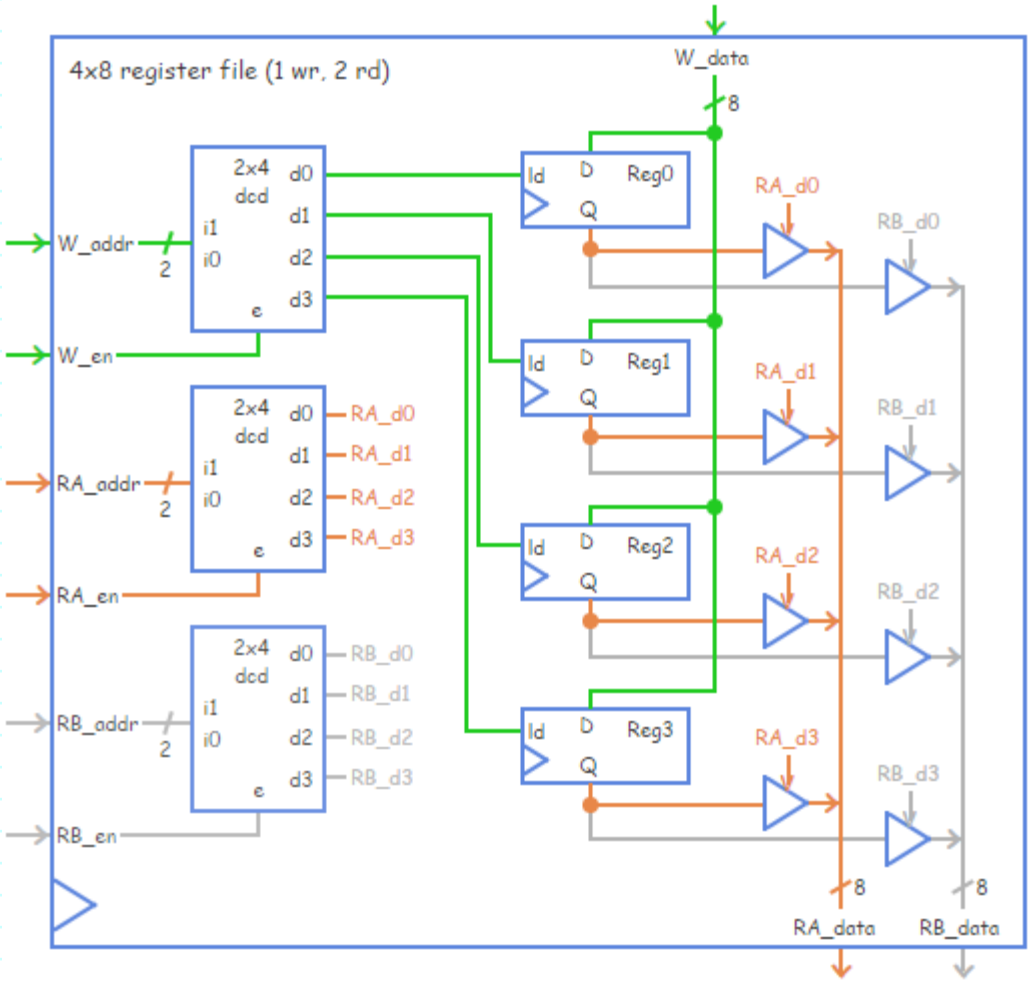

a1
a0
b1
b0
0 0
0 0
0 0 1 0
i3 i2 i1 i0
w
x
s1
s0
4x1
y
a1 a0
b1 b0
cin
2-bit adder
cout
s1 s0
2-bit ALU
s1 s0

Register File



CPU


a1
a0
b1
b0
0 0
0 0
0 0 1 0
i3 i2 i1 i0
s1
s0
4x1
y
w
x
a1 a0
b1 b0
cin
2-bit adder
cout
s1 s0
2-bit ALU


4x8 register file (1 wr, 2 rd)
W_data
8
W_addr
2
W_en
RA_addr
RA_en
RB_addr
RB_en
2x4
dcd
i1
i0
e
d0
d1
d2
d3
ld
D
Q
Reg0
Reg1
Reg2
Reg3
RA_d0
RA_d1
RA_d2
RA_d3
RB_d0
RB_d1
RB_d2
RB_d3
RA_data
RB_data

Instructions

Instruction (first three bits)	Operation
000 - nop	Do nothing
001 - load	reg[destination] = dataIn
010 - increment	reg[destination] = reg[destination]++
011 – decrement	reg[destination] = reg[destination]--
100 – shift left	reg[destination] = shift left(reg[source])
101 – shift right	reg[destination] = shift right(reg[source])
110 – addition	reg[destination] = reg[source] + dataIn
111 – subtraction	reg[destination] = reg[source] - dataIn