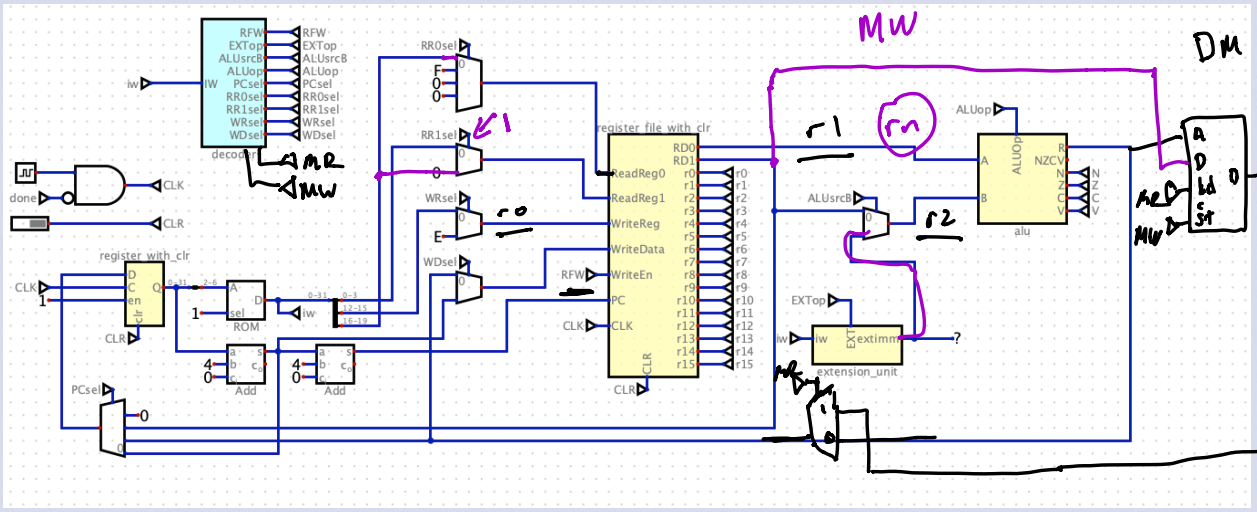
SDT (ldr/str)
Multiply (mul)
Shifting (lsl, lsr)
Condition execution (Control Unit)

Decoder
add(r)
addi
mov(r)
mov (i)
:
:

ldr(r)	ldr r0, [r1, r2]
ldr(i)(dn) ←	ldr r0, [r1, #-8]
ldr (i) (up) ←	ldr r0, [r1, #16]

SDT



ldr r0, [r1, r2] (+)

19	16	15	12	3	0
cond	01	I	P	U	B	W	L	rn	rd		rm

0 1 1 1 1 0 0 1

input bits

str r0, [r1, r2] X

str r0, [r1]

str r0, [r1, #0]] OK

rd

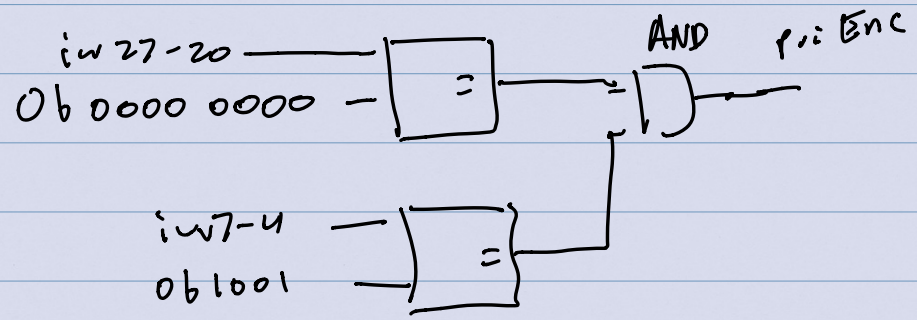
RFW = 0

MW = 1

Multiply

Need to modify data path to select rd, rs

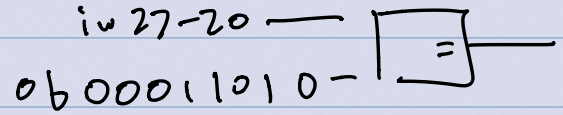
Decoder



LSL / LSR

Decoder

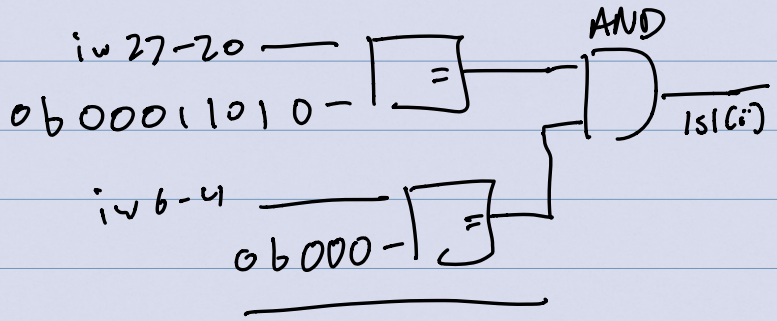
mov(r) + Shift field



lsl(r) ⟶ lsl r0, r1, r2

r0 = r1 << r2

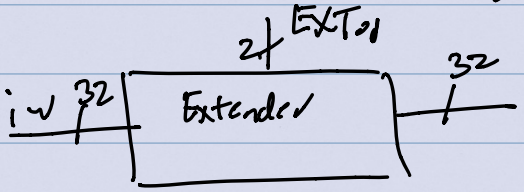
lsl(i) ⟶ lsl r0, r1, #2



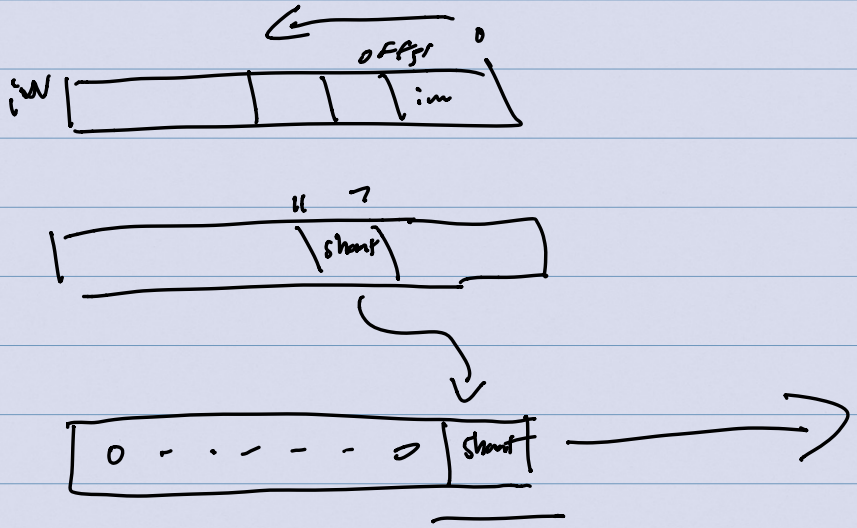
isI(5)
0b0001
4 bits

Datapath

We need bits 11-7 to go to Extender.



00	8	bit	zero extend	✓
01	12	bit	zero extend	✓
10	24	bit	sign extend and x4	✓
11	5	bit	shift, and zero extend	



Control Unit

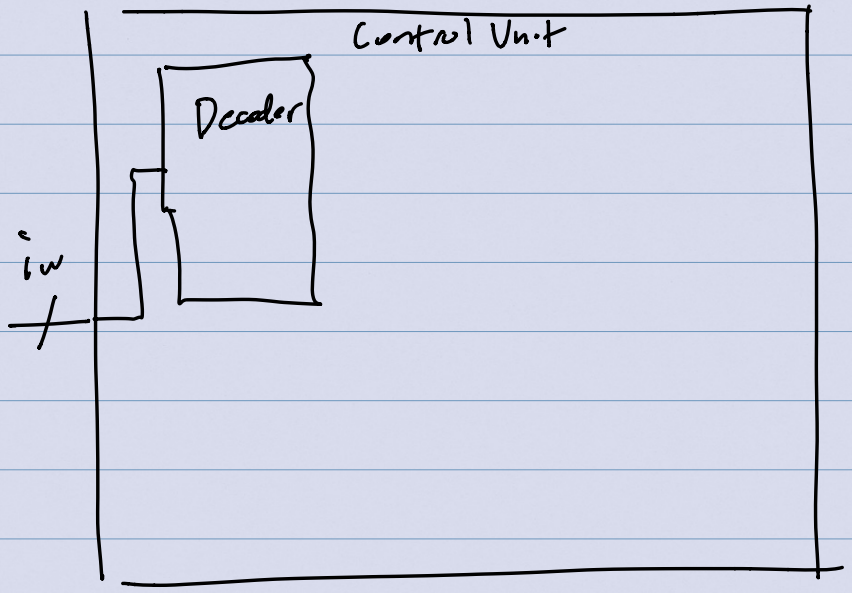
Conditional Execution

beq, bne, blt, bge

cmp r0, #0 → N Z C V → CPSR reg

→ beq foo ←

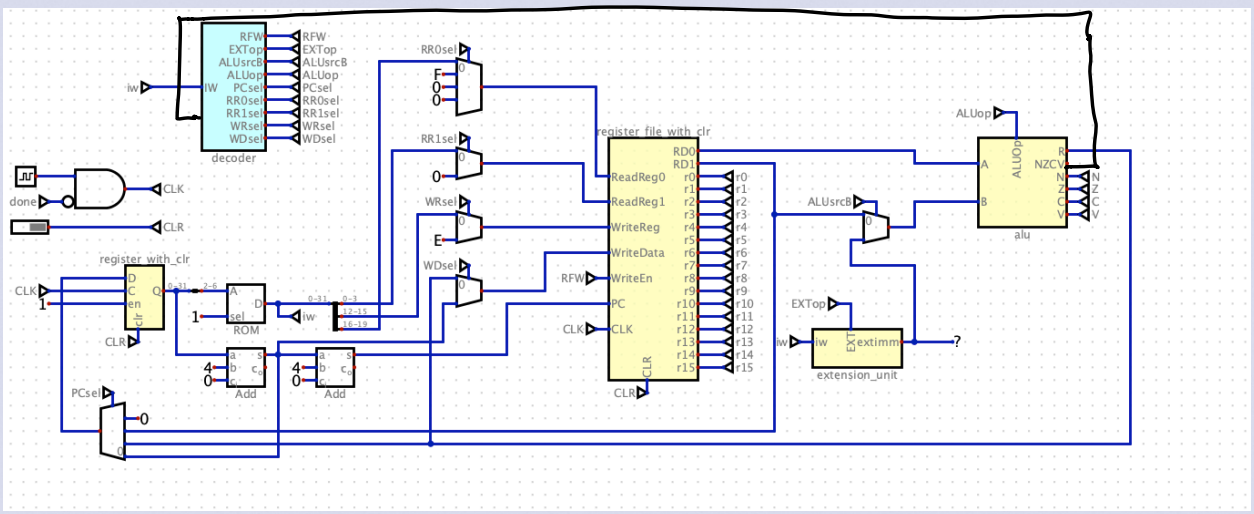
→

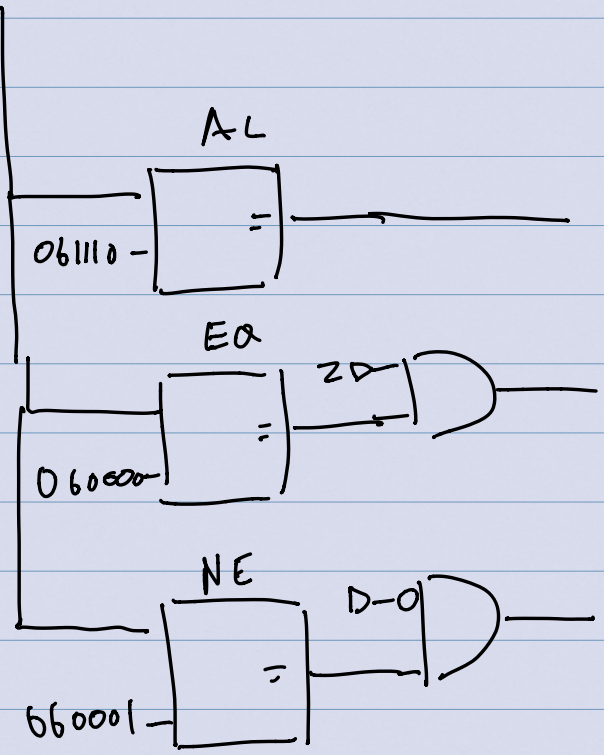


Conditional Execution:

1) On cmp store NZCV bits into CPSR reg in Control unit

2) On execution of conditional branch
check cond code and cond cond
if true → take branch
if false → go to PC+4


decoder
register_with_clr
ROM
Add
Add
register_file_with_clr
ALUop
alu
ALUsrcB
EXTop
extension_unit
RR0sel
RR1sel
WRsel
WDsel
RFW
PCsel
CLK
CLR
done


AL
06111 0
EQ
2D
0 60000
NE
D-Q
660001