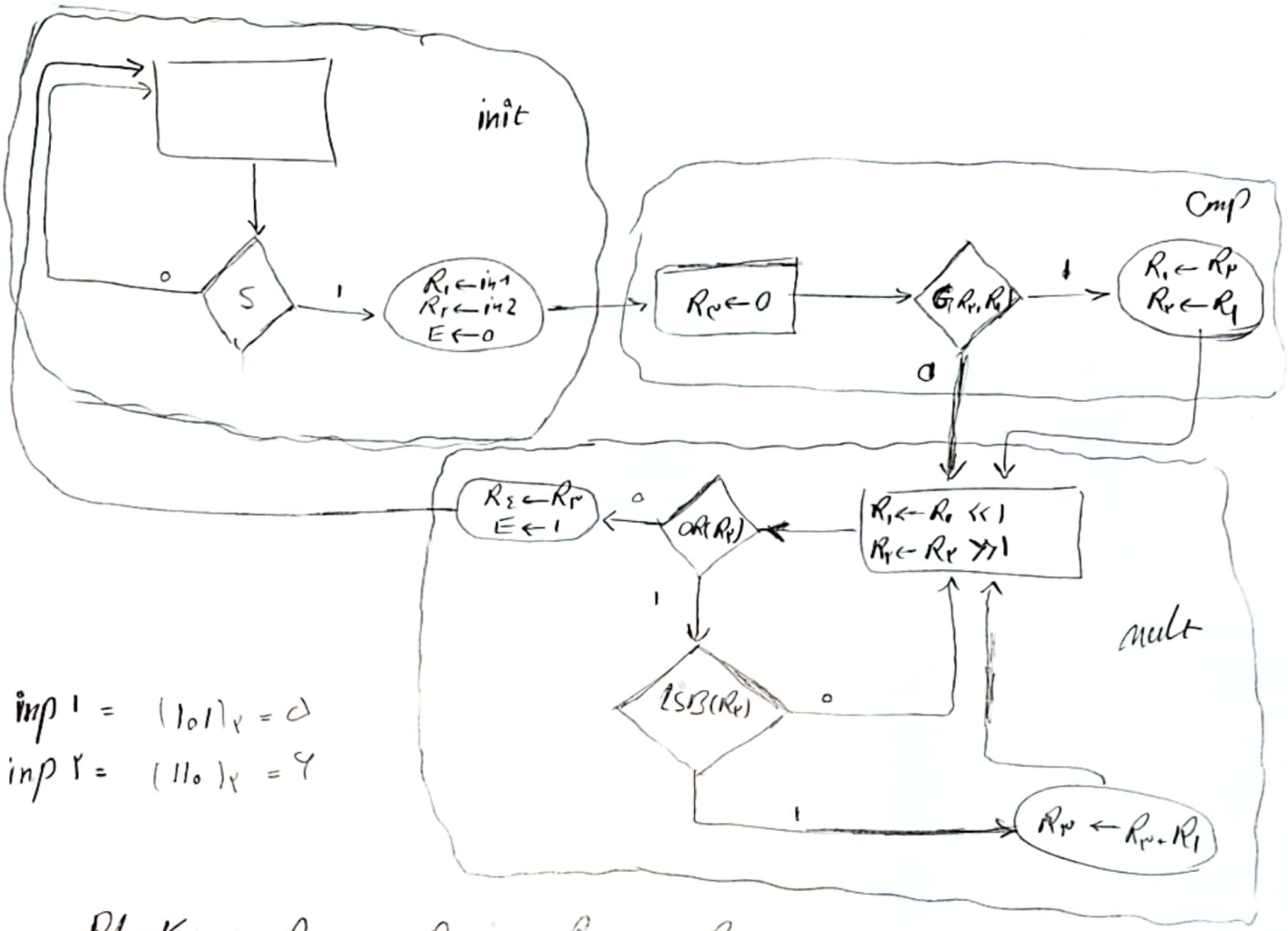
OSD - multiplier



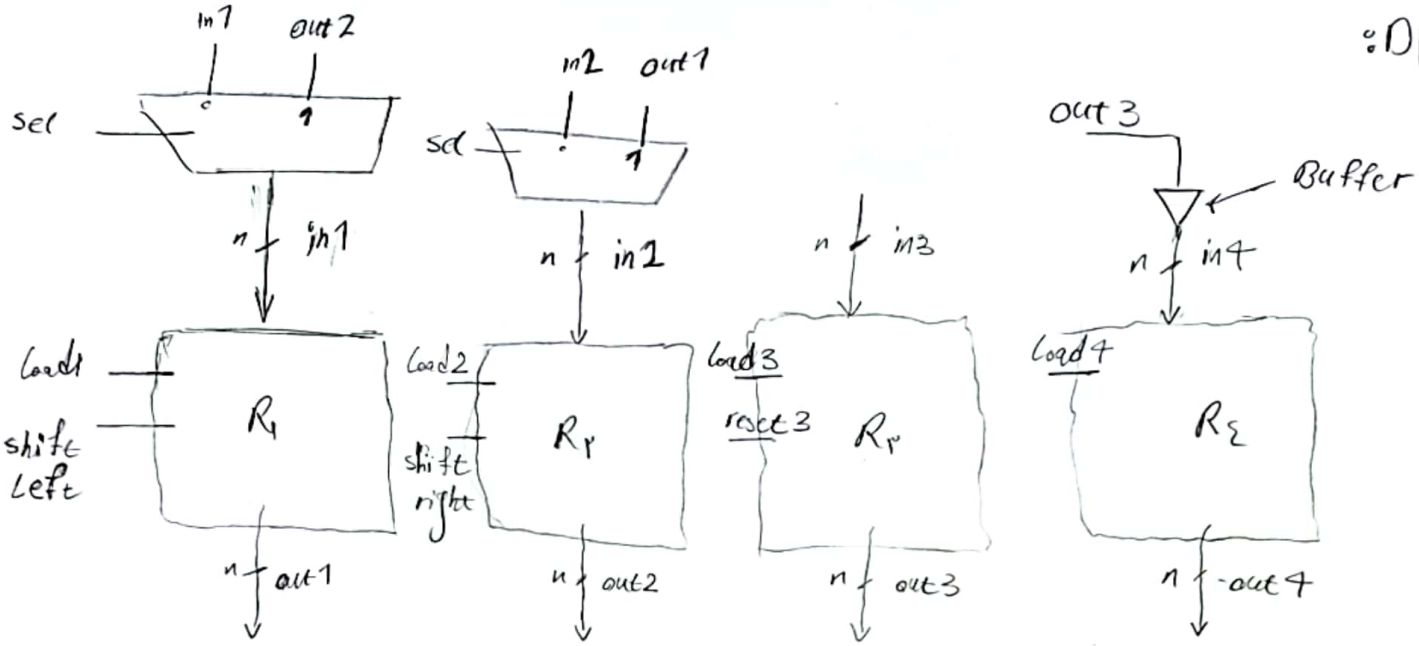
inp 1 = $(101)_2 = 5$

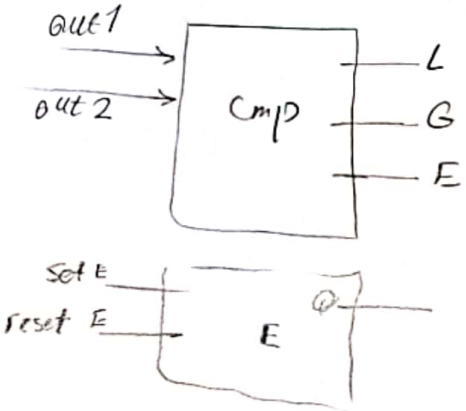
inp 2 = $(110)_2 = 6$

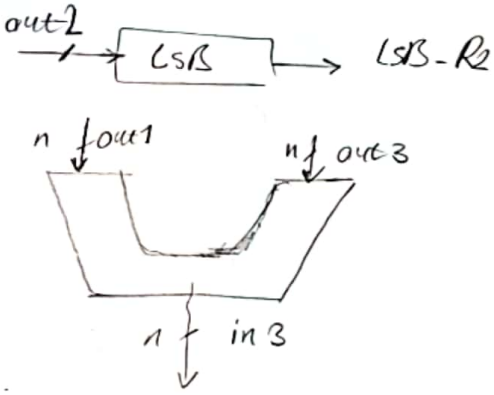
Block	R_1	R_2	R_3	R_4	ready/E
init	-	0	-	-	1
Cmp	1.1	110	-	-	0
mult	110	101	0	-	0
mult	1100	10	110	-	0
mult	11000	1	110	-	0
mult	110000	0	1110	-	0
init	110000	0	1110	1110	1

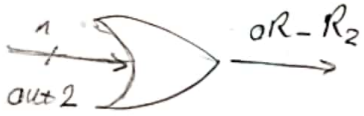
$5 \times 6 = 30 = (11110)_2$

: DP

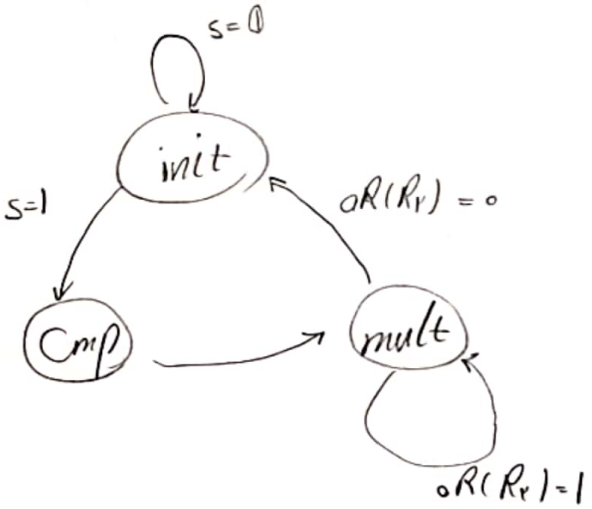








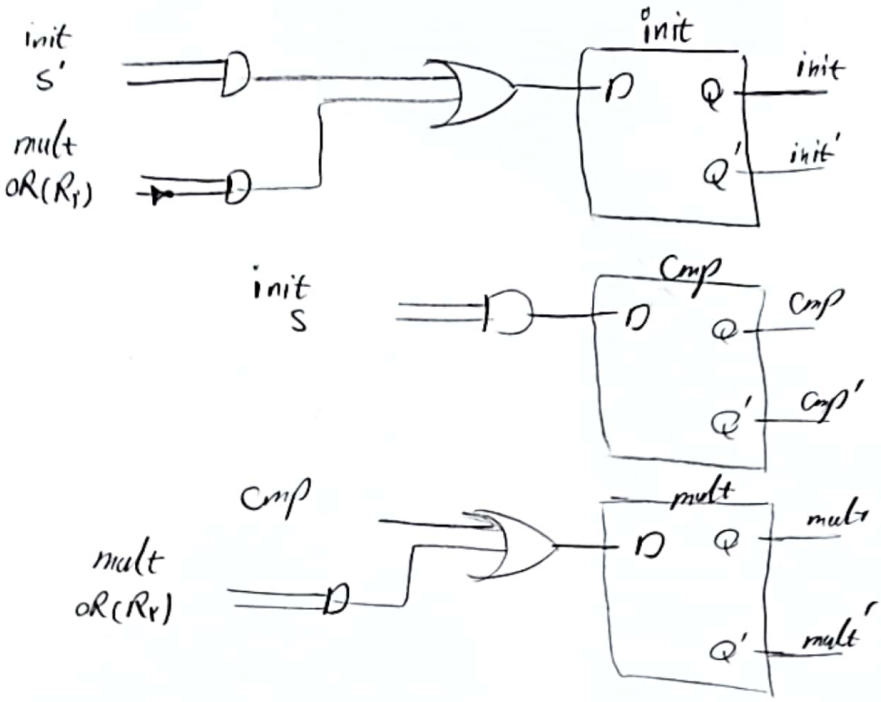
: CU



$D_{init} = init \cdot s' + mult \cdot oR'(R_1)$

$O_{cmp} = init \cdot s$

$D_{mult} = Cmp + mult \cdot oR(R_1)$



: CS

Sel = init'

shift left = mult

shift left = mult

reset 3 = Cmp

Load1 = init.s + Cmp · G(Rf, Rf)

Load2 = init.s + Cmp · G(Rf, Rf)

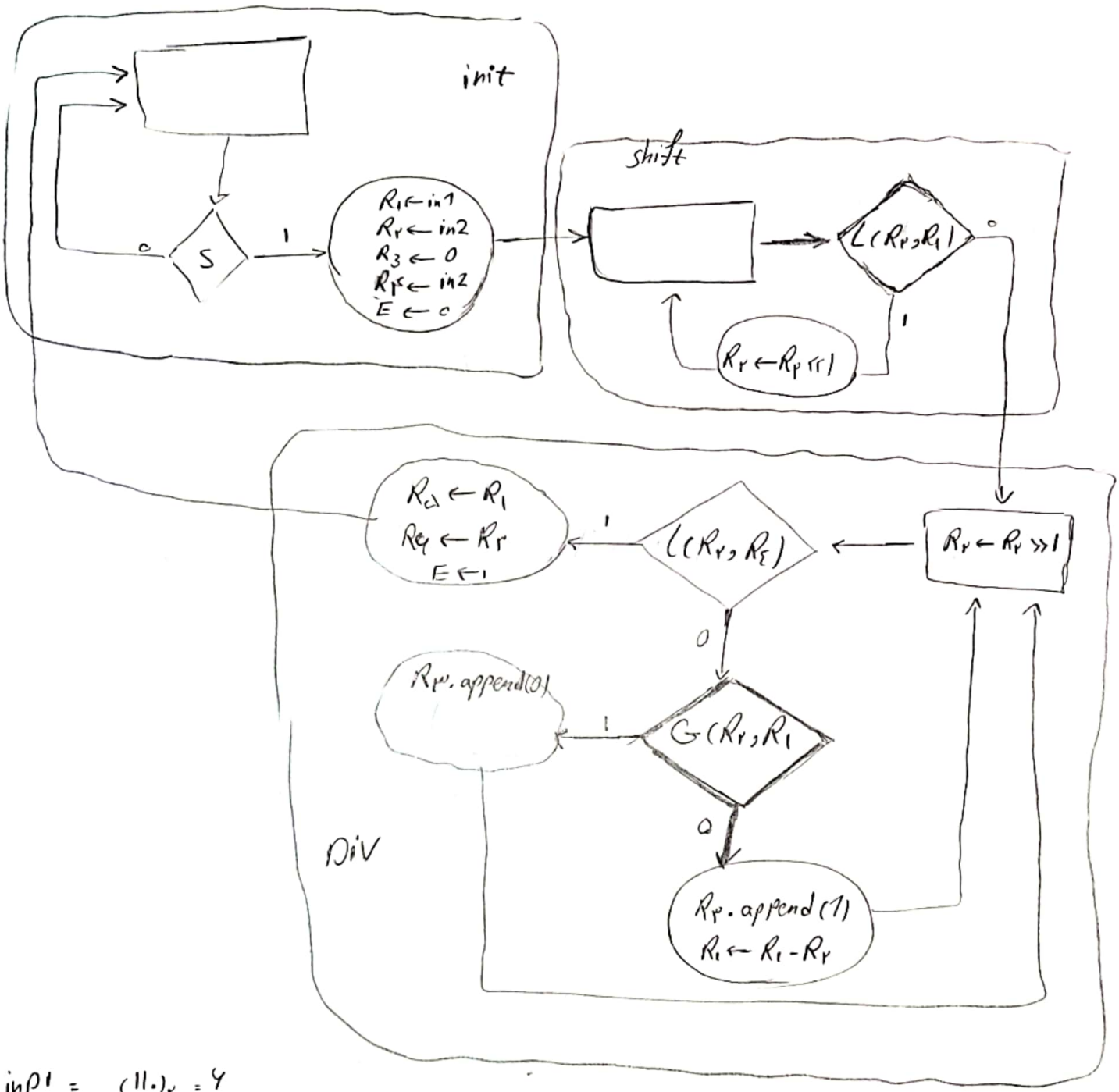
Load3 = mult . oR(Rf) . LSB(Rf)

Load4 = mult. oR'(Rf)

Set E = mult. oR'(Rf)

reset E = init . S

OSD - devider



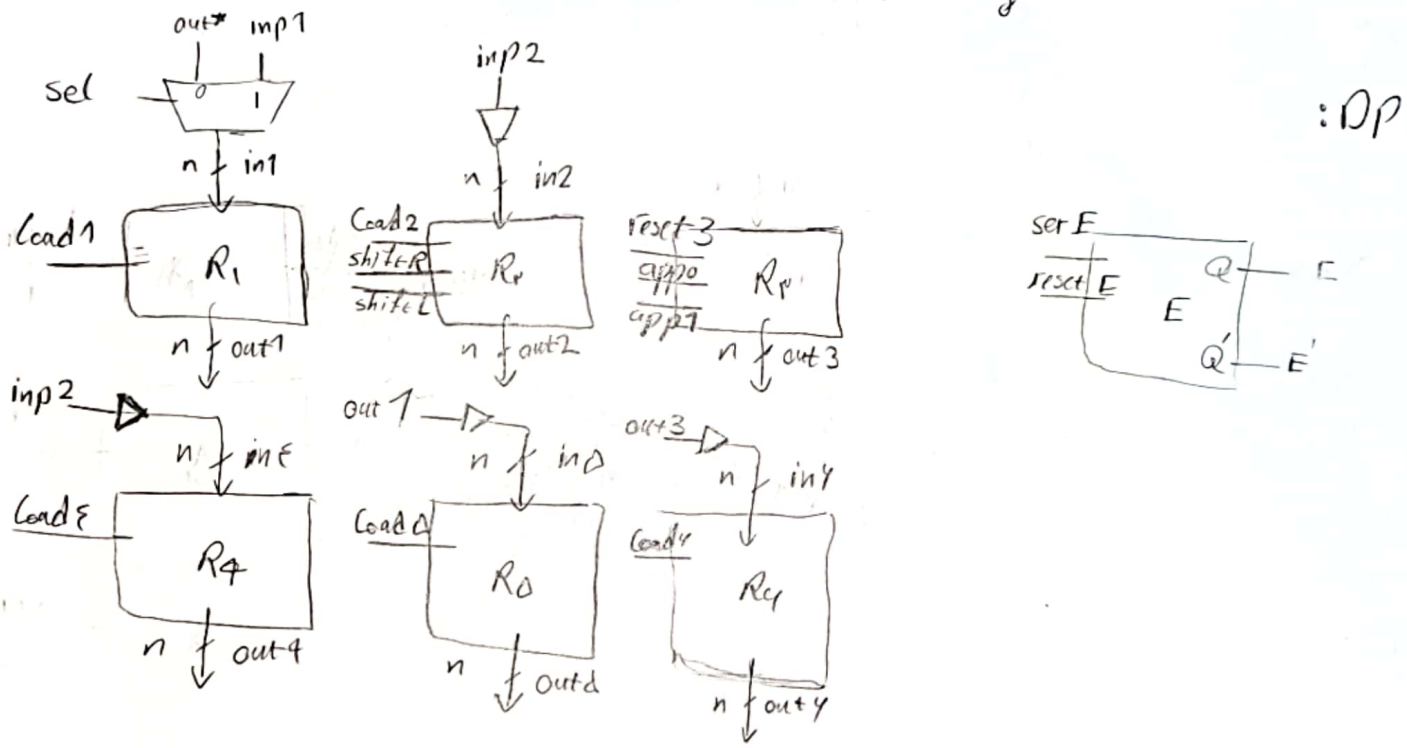
inp1 = $(110)_2 = 6$

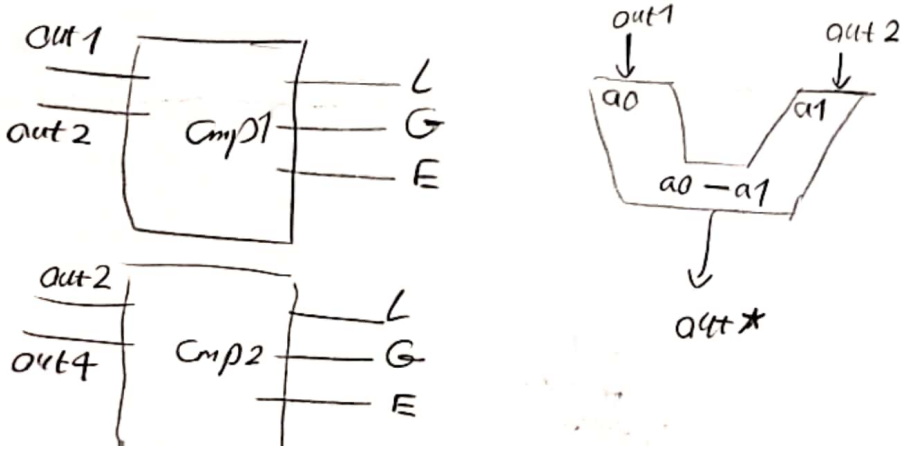
inp2 = $(101)_2 = 5$

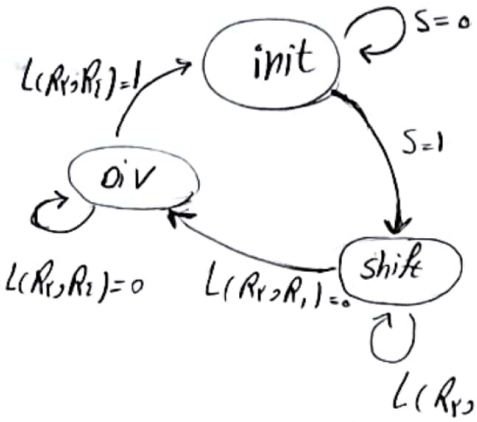
* append0 : یعنی بیت 0 را به راست عدد اضافه می‌کند (app0)
* appendA1 : " " " " " 1 " " (app1)
* منظور از ← باضرب و بیشتر بخاطر جدا کردن نام‌گذاری کم‌دا استفاده شده ، یعنی اصلا نیازی به آن‌ها نداریم و فقط جهت واضح‌تر شدن نام‌گذاری کم‌دا آن‌ها را گذاشته‌ایم ! « فقط ارزش نمایشی دارد »

Block	R_1	R_2	R_3	R_4	R_5	R_6	ready/E
init	-	0	-	-	-	-	1
shift	110	101	0	101	-	-	0
shift	110	1010	0	1a1	-	-	0
Div	110	1010	0	101	-	-	0
Div	110	101	0	101	-	-	0
Div	1	10	1	101	-	-	0
init	1	1	1	101	1	1	1

remainder quotient



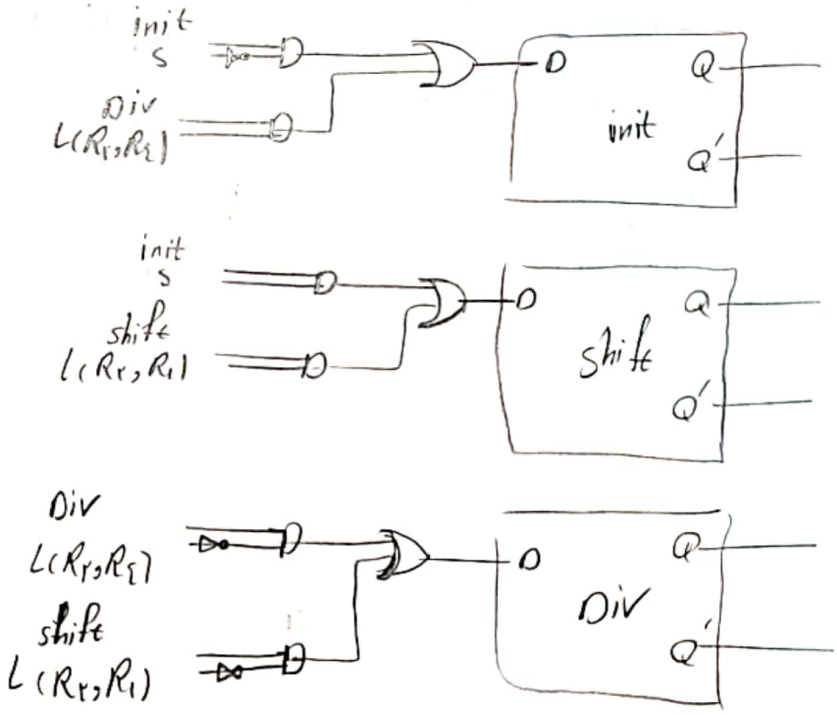




$$D_{init} = init \cdot S' + Div \cdot L(R_1, R_1)$$

$$D_{shift} = shift \cdot L(R_1, R_1) + init \cdot S$$

$$D_{Div} = Div \cdot L'(R_1, R_1) + shift \cdot L'(R_1, R_1)$$



: CS

Set E = Div. $L(R_7, R_8)$

Reset E = init . S

Load 1 = init . S + Div. $L'(R_7, R_8) . G'(R_7, R_1)$

Load 2 = init . S

Load 4 = init . S

Load 5 = Div. $L(R_7, R_8)$

Load 6 = Div. $L(R_7, R_{16})$

Shift R = Div

Shift L = Shift. $L(R_7, R_1)$

App 0 = Div. $L'(R_7, R_5) . G'(R_7, R_1)$

App 1 = Div . $L'(R_7, R_{16}) . G(R_7, R_1)$

Reset 3 = init . S