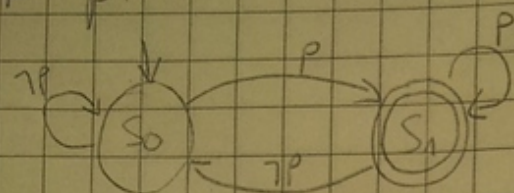
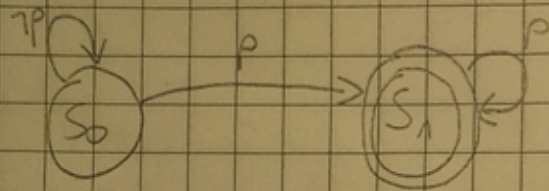


1) a)  $GF\ p$



b)  $FG\ p$



2)  $E(GF\ p)$

- pripada  $CTL^*$  - unificirana struktura  $CTL(E)$  i  $LTL(GF)$
- govori da postoji put na kojem je  $p$  istinit beskonačno često
- nije  $LTL$  jer ne vrijedi za sve putove modela

3) a)  $AF\ p = A(TU\ p)$

b)  $EG\ p = \neg(A(TU\ \neg p))$

```

1 typedef enum (s1, s2, s3) stateT
module kiple (clk)
input clk;
reg x, y, z;
wire rand;
stateT reg state;
initial begin
    state = s1;
    x = 1;
    y = 1;
    z = 0;
end
assign rand = $ND(0, 1);
always @(posedge clk) begin
    case (state)
        s1: begin
            if (rand == 1)
                begin
                    y = 0;
                    z = 1;
                    state = s2;
                end
            else
                begin
                    x = 0;
                    y = 0;
                    z = 1;

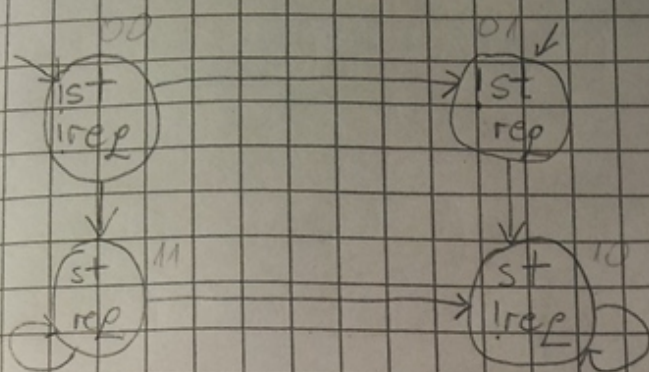
```

```

                state = s3;
            end;
        end;
        s2: begin
            if (rand == 1)
                begin
                    x = 0;
                    state = s3;
                end;
            end;
        s3: begin
            if (rand == 1)
                begin
                    x = 1;
                    y = 1;
                    z = 0;
                    state = s1;
                end;
            end;
        endcase;
    end;
endmodule

```





SPEC  $E \neq AG (st=1 \ \& \ rep = \text{nešto})$

↳ za  $rep=0$  zadovoljava, za  $rep=1$  ne zadovoljava

6) a)  $x =$

int  $x=1$

active prototype A1()

```
{
  x--;
  do
  :: (x >= 4) -> break;
  :: true -> x++;
  od;
```

active prototype A2()

```
{
  do
  :: x++ -> x--;
  od
}
```

b) FSM C

$S = \{ S_0 S_0, S_0 S_1, S_0 S_2, S_1 S_0, S_1 S_1, S_1 S_2 \}$

$S_0 = \{ S_0 S_1 \}$

$L = \{ x--, x \geq 4, x++ \}$

$F = \{ S_0 S_0, S_1 S_0, S_2 S_0, S_2 S_1 \}$

$T = \{ S_0 S_0, x++, S_0 S_1$

$S_1 S_0, x--, S_0 S_0$

$S_0 S_0, x--, S_1 S_0$

$S_0 S_1, x--, S_1 S_1$

$S_1 S_0, x++, S_1 S_0$

$S_1 S_1, x++, S_1 S_1$

$S_1 S_0, x \geq 4, S_2 S_0$

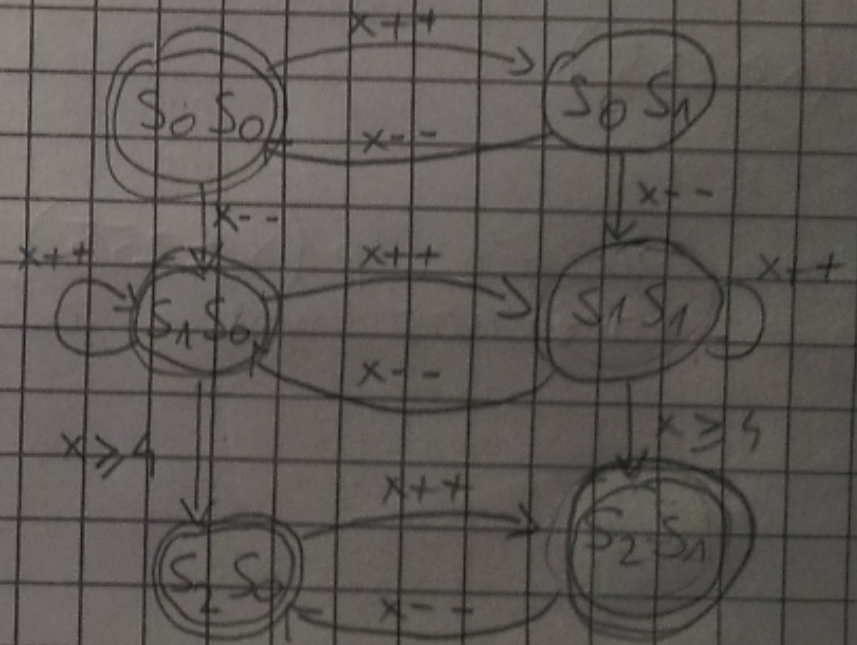
$S_1 S_1, x \geq 4, S_2 S_1$

$S_2 S_0, x++, S_2 S_1$

$S_2 S_1, x--, S_2 S_0$

$S_0 S_1, x--, S_0 S_0$

$S_1 S_1, x--, S_1 S_0$





c) never  $\{ \quad \} \wedge \Diamond [ \Box p^* ]$

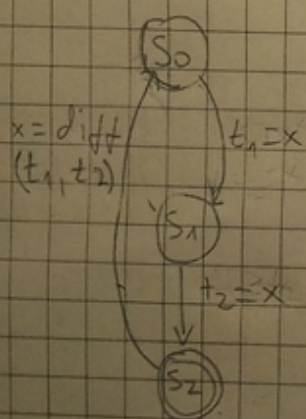
```

TO-init:
do
  :: p → goto accept_S4
  :: 1 → goto TO-init
od;
accept_S4:
do
  :: p → goto accept_S4
od;
}

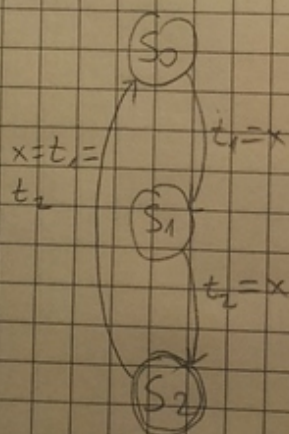
```

d) Postoji li proces moze razlikovati između  $x++$  i  $x--$

7)  $A_1$



$A_2$



b) assert  $(x > 255)$

c) uvrucimo uvjet  $p(x \leq 255)$  u never  $\{ \text{loc} \mid Gp \}$ . Ukoliko  $x$  dosegne vrijednost vecu od 255 spin ce vratiti gresku.



$$8) f(xy) + f(xy) = f(x) + f(y)$$

$$x f_x + x' f_{x'} = x [f(1+y) + f(y)] + x' [f(y) + f(0)] =$$

$$x f(1) + x f(y) + x' f(y) + x' f(0) = x f(1) + x' f(0) + x f(y) + x' f(y)$$

$$f(x) + f(y) (x+x') = f(x) + f(y)$$

$$9) a) \text{ and } (f, g) = \text{ite}(f, g, 0)$$

$$b) \text{ or } (f, g) = \text{ite}(f, 1, g)$$

$$c) \text{ nand } (f, g) = \text{ite}(f, \bar{g}, 1)$$

$$d) \text{ xor } (f, g) = \text{ite}(f, \bar{g}, g)$$

00	0
01	1
10	1
11	0

$$12) F = \bar{S}x + Sy$$

$$13) M = \bar{S}_1 \bar{S}_2 a + \bar{S}_1 S_2 b + S_1 \bar{S}_3 c + S_1 S_3 d$$

