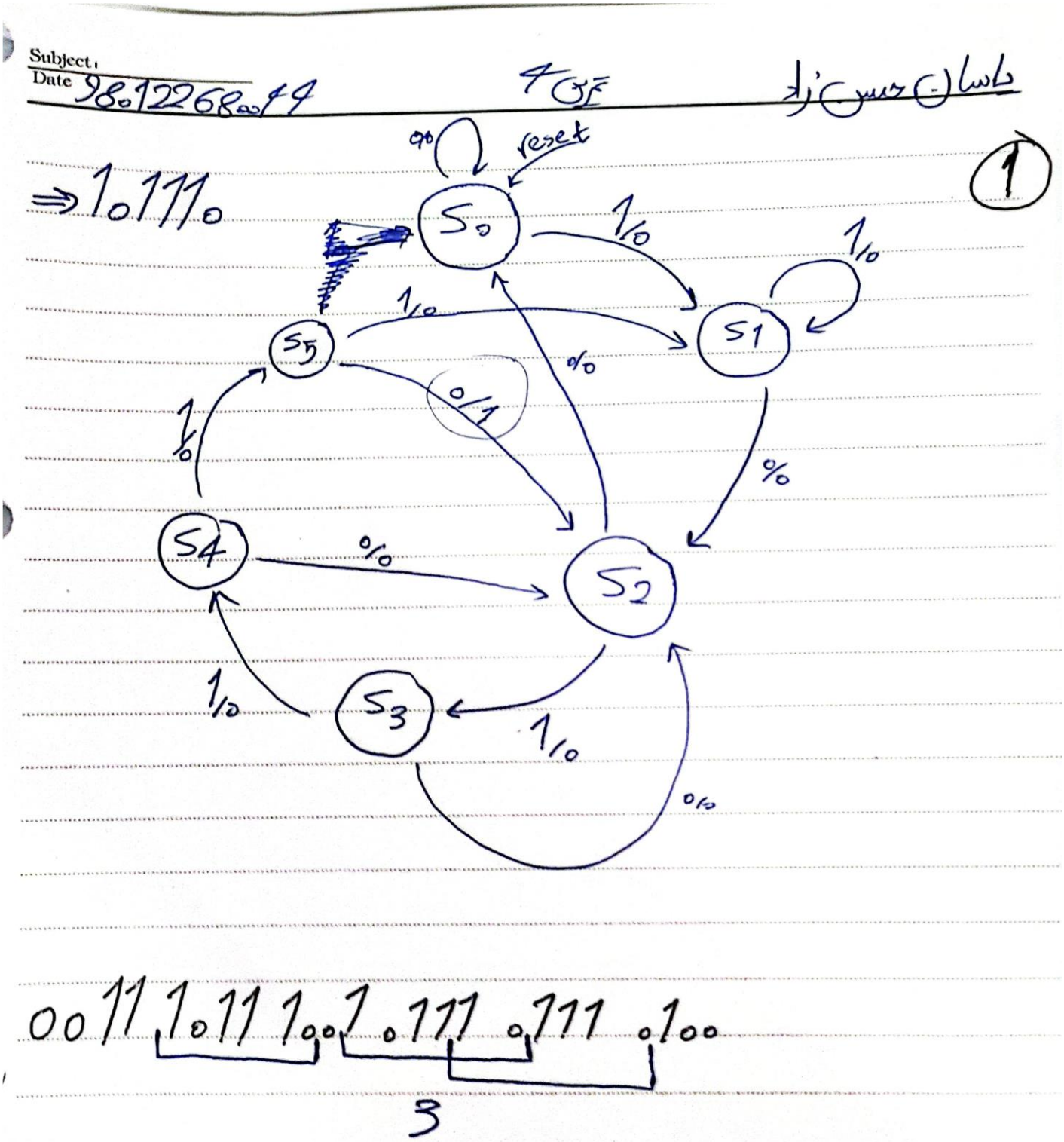
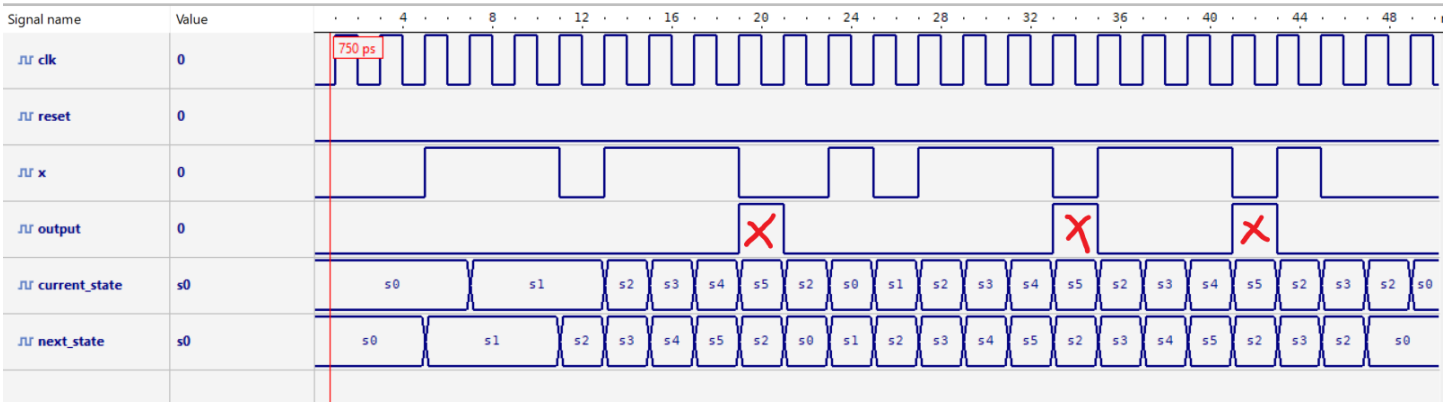
یاسان حسن زاد 980122680044

. 1

: sequence detector

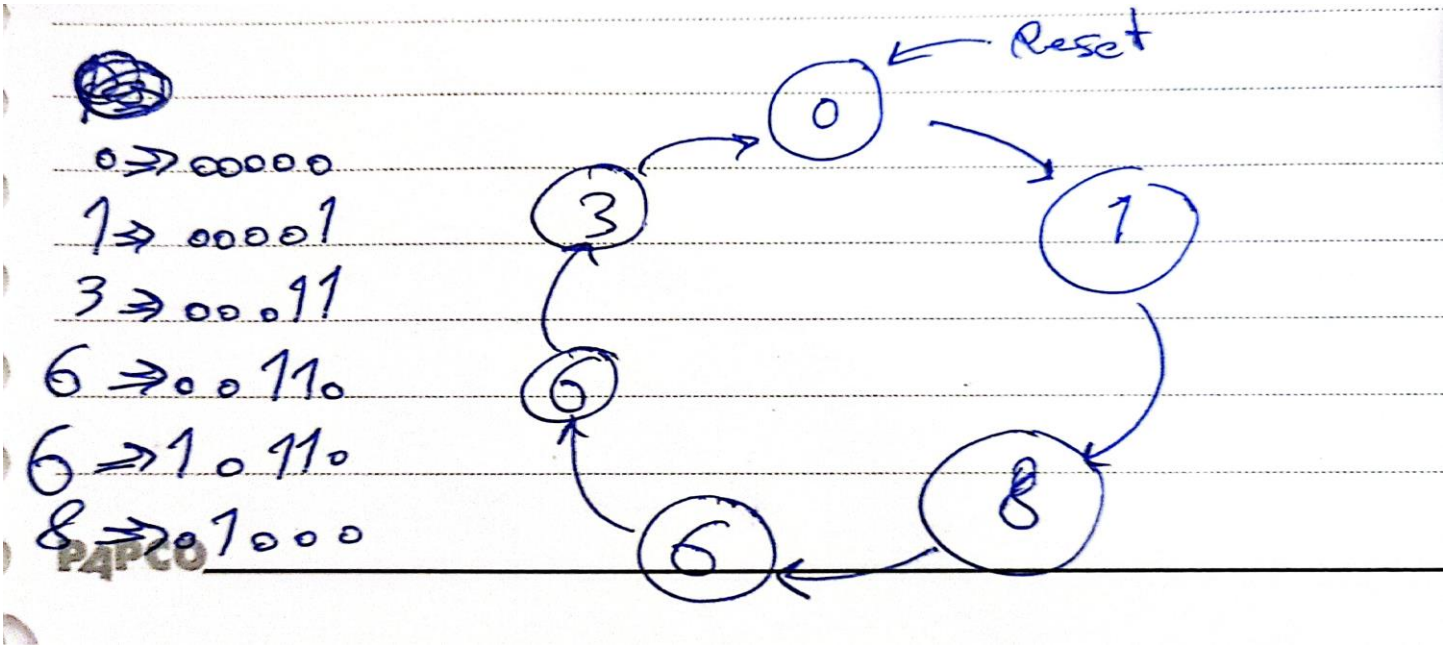


: Tesetbench



. 2

: sequence detector



: Hand Coding

```
architecture T42 of T42 is
subtype T42_state is std_logic_vector(4 downto 0);
signal current_state , next_state : T42_state;

constant s0 : T42_state := "00000";
constant s1 : T42_state := "00001";
constant s3 : T42_state := "00011";
constant s61 : T42_state := "00110";
constant s62 : T42_state := "10110";
constant s8 : T42_state := "01000";


begin
```

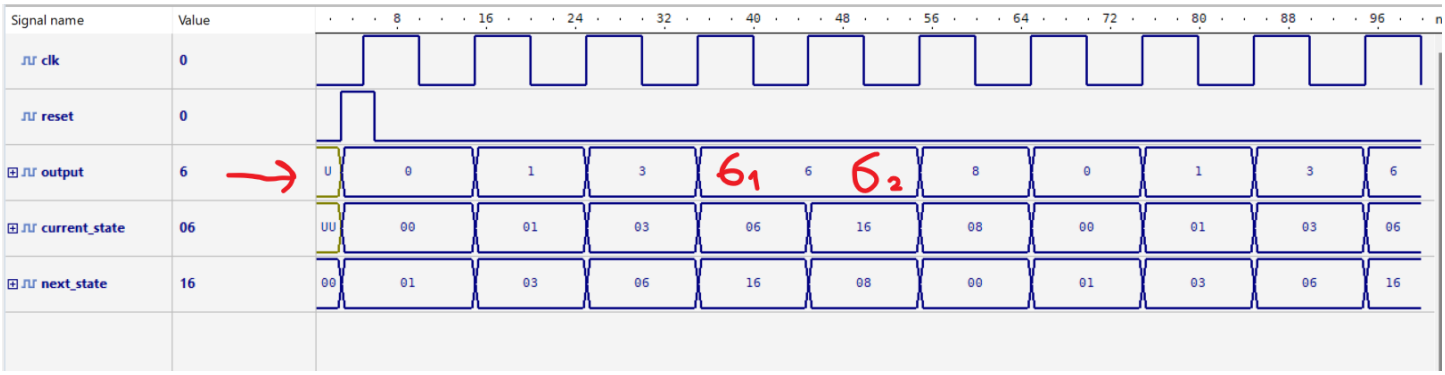
: REG & CMB & OUTPUT

```
REG : process(clk,reset)
begin
    if reset = '1' then
        current_state <= s0;
    elsif clk'event and clk ='1' then
        current_state <= next_state;
    end if;
end process;

CMB : process(current_state)
begin
    case current_state is
        when s0 => next_state <= s1;
        when s1 => next_state <= s3;
        when s3 => next_state <= s61;
        when s61 => next_state <= s62;
        when s62 => next_state <= s8;
        when s8 => next_state <= s0;
        when others => next_state <= s0;
    end case;
end process;

output <= current_state(3 downto 0);
 -- enter your statements here --
```

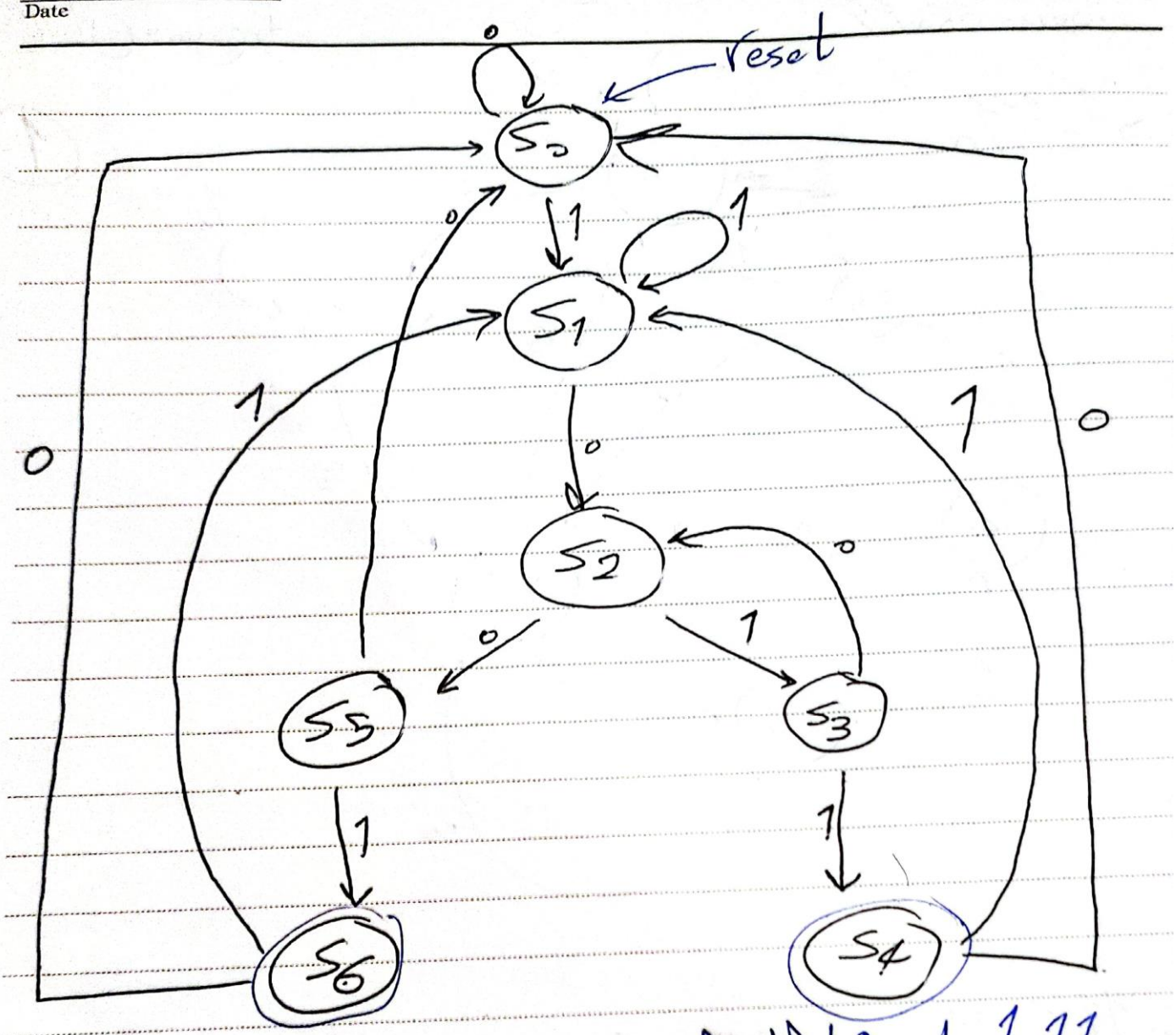
: Tesetbench



. 3

: sequence detector

Subject:
Date



1001 DatintoReg

RegtoDataout 1011

X = 00 1 0 0 1 00 1 0 1 0 11 00

: Datapath & outputs

```
DP_R : process(clk , reset)
begin
     dataout_reg <= (others => 'Z');
               valid_reg <= '0';
     if reset = '1' then
          current_reg <= (others => '0');
          valid <= '0';
     elsif clk'event and clk ='1' then
          if current_state = s6 then
               current_reg <= datain;
          end if;

          if current_state = s4 then
               dataout_reg <= current_reg;
               valid_reg <= '1';
          else
               dataout_reg <= (others => 'Z');
               valid_reg <= '0';
          end if;
     end if;

end process;

dataout <= dataout_reg;
valid <= valid_reg;
```

: Tesetbench