

Adim : Mohamed Ibrahim Abdi

No: 376203

Ders sorumlusu: Dr.Ibrahim Savran.

Santral Devresi

“8 girişi olan ve 8 çıkış olan,arada 4 ara kablosu olan santral Devre tasarlamaktayım.

8 giriş 4 ara kablolar hangisini boş ise bağlayıp çıkışa aktarıyor “

Library IEEE, std;

USE IEEE.std_logic_1164.all;

entity Tsantral is

Port(A: in std_logic_vector(7 downto 0);

B: out std_logic_vector(7 downto 0);

k, h: in integer;

t: in integer;

s: in std_logic);

end Tsantral;

Architecture Struct of Tsantral is

Begin --mimari

-- eCok: eGiris PORT MAP (A, S, kk);

--eTek: eCikis PORT MAP (B, T, kk);

process (s)

Type tSyc is array (0 to 3) of integer;

Type tInt10 is array (0 to 9) of integer;

variable sSyc: tSyc := (0, 0 , 0, 0);

```

variable mSyc: tSyc := (0, 0 , 0, 0);
variable bsyc: tSyc := (0, 1, 2, 3);

variable pA, pB: std_logic_vector(7 downto 0) := "00000000";

--variable pK: std_logic_vector(3 downto 0) := "0000";
variable ns : integer := 0 ;
--variable pH: std_logic_vector(3 downto 0) := "0000";
variable n,m, nc : integer := 0;
variable pH, pK, pT : tInt10;

```

```

Begin
if (rising_edge(s)) then
if n < 10 then
pH(n) := h;
pK(n) := k;
pT(n) := t;
n := n + 1;
end if;
if m < n then
if pA(pK(m)) /= '1' and pB(pH(m)) /= '1' and nc < 4 then
pA(pK(m)) := '1';
pB(pH(m)) := '1';
B(pH(m)) <= A(pK(m));
sSyc(bsyc(nc)) := pT(m);
mSyc (bsyc(nc)) := m;
nc := nc + 1;
m := m + 1;
end if;
end if;
if sSyc(0) /= 0 then
sSyc(0) := sSyc(0) -1;

```

if sSyc(0) = 0 then

pA(pK(mSyc (0))) := '0';

pB(pH(mSyc (0))) := '0';

B(pH(mSyc (0))) <= 'Z';

mSyc (0) := -1;

end if;

end if;

if sSyc(1) /= 0 then

sSyc(1) := sSyc(1) -1;

if sSyc(1) = 0 then

pA(pK(mSyc (1))) := '0';

pB(pH(mSyc (1))) := '0';

B(pH(mSyc (1))) <= 'Z';

mSyc (1) := -1;

end if;

end if;

if sSyc(2) /= 0 then

sSyc(2) := sSyc(2) -1;

if sSyc(0) = 0 then

pA(pK(mSyc(2))) := '0';

pB(pH(mSyc(2))) := '0';

B(pH(mSyc (2))) <= 'Z';

mSyc (2) := -1;

end if;

end if;

if sSyc(3) /= 0 then

sSyc(3) := sSyc(3) -1;

```
if sSyc(3) = 0 then
```

```
    pA(pK(mSyc (3))) := '0';
```

```
    pB(pH(mSyc (3))) := '0';
```

```
    B(pH(mSyc (3))) <= 'Z';
```

```
    mSyc (3) := -1;
```

```
end if;
```

```
end if;
```

```
end if; -- rising
```

```
end Process;
```

```
--Process( pK(0) )
```

```
--begin
```

```
--IF pK(0) = '0' Then
```

```
--End IF;
```

```
-- End Process;
```

```
-- Component eCikis is
```

```
-- port(j2:out std_logic_vector(7 downto 0);
```

```
--  t2: in std_logic_vector(2 downto 0);
```

```
--  k3: in std_logic_vector(3 downto 0);
```

```
--      j2 <= K3 when "11",
```

```

--      K3 when "10",
--      K3 when "01",
--      K3 when "00",
--      K3 when "others";
--End eCikis;

--Signal kk: std_logic_vector(3 downto 0);

--Begin

-- egir: eGiris port map(A,k,t,kk)
-- ecik: eCikis port map(B,h,t,kk)

--      b(2) <= A(5)
--      wait for 6 sn -- fist
----      b(2) <= A(6)
---      b(1) <= A(3) after 1 sn -- second
--      b(3) <= A(7) after 1 sn
--      wait for 19 sn
--      b(3) <= A(5) -- third
--      b(4) <= A(2) after 1 sn
--      b(1) <= A(1) after 1 sn
--      b(7) <= A(4) after 1 sn
--      Wait for 7 sn
--      b(5) <= A(0)
--      b(3) <= A(3) after 1 sn
--      Wait for 12 sn;

--      b(1) <= A(1) after 1 sn
--      b(7) <= A(4) after 1 sn
--      Wait for 7 sn

```

-- b(5) <= A(0)
-- b(3) <= A(3) after 1 sn
-- Wait for 12 sn;

end Struct;

ilk istek olan (7,5,6) geldi zaten ilk istek olduğu için ara hatlar, kaynak ve hedef boşta. 6 sn boyunca bu bağlantı gerçekleşir.

1 sn sonra: ilk isteğin 5 saniyesi kaldı.

1. istek (6,2,10) geldi ancak B(2) hedefi ilk. istek tarafından kullanılıyor bağlantı için minimum 6 sn beklemeli.

1 sn sonra ilk. isteğin 5 saniyesi kaldı.

2. istek olan (3,1,2) geldi; A(3), B(1) boş olduğu için bir sonraki boş ara hat olan 1 kullanılarak bağlantı 2 saniye boyunca gerçekleşir.

1 sn sonra 0. isteğin 4 saniyesi, 2. isteğin 1 saniyesi kaldı.

3. istek olan (7,3,20) geldi; B(3) ve A(7) boş bir sonraki ara hat olan 2 kullanılarak bağlantı 20 saniye boyunca gerçekleştirilir.

1 sn sonra 0. isteğin 3 saniyesi, 3. isteğin 19 saniyesi kaldı ve 2. istek tamamlandı.

4. İstek (5,3,12); bağlantı gerçekleşemez çünkü 0. istek sonlanmadığından A(5) kaynağı; 3. istek sonlanmadığından B(3) hedefi meşguldür. 4. istek de bekleyen listesine alınır. Böylece bekleyen listesinde 1. istek ve 4. istek vardır.

1 sn sonra 0. isteğin 2 saniyesi, 3. isteğin 18 saniyesi kalmıştır

5. istek olan (2,4,10) gelir; B(4) A(2) ve 1. ara hat kullanılarak bağlantı 10 saniye boyunca gerçekleştirilir.

1 sn sonra 0. isteğin 1 saniyesi, 3. isteğin 17 saniyesi, 5. isteğin 9 saniyesi kalmıştır.

6. istek olan (1,1,10) gelir. A(1), B(1) boş olduğu için bağlandı boştaki son hat olan 3. ara hat kullanılarak 10 saniye boyunca gerçekleştirilir.

Bu bağlantı ile birlikte bütün ara hatlar dolmuştur. İşleyen bağlantılar dikkate alındığında bir sonraki bağlantının gerçekleşebilmesi için bir ara hattın serbest bırakılması gerekir. İşleyen bağlantılardan en erken 0. istek 1 saniye sonra sonacaktır. 1 saniye sonra 0. istek tamamlandı. 3. isteğin 15 saniyesi, 5. isteğin 7 saniyesi, 6. isteğin 8 saniyesi kalmıştır.

7. istek gelir ancak tamamlanan 0. işlem ara hattı sonraki saniye bırakacağından beklemeye geçer.

1 saniye sonra. 8. istek gelir ve 0. istek ara hattı bırakmıştır. Bu durumda:

1. istek B(2) için, 4. istek A(5) için 0. isteğin sonlanmasını beklerken, 7. istek de ara hat için beklemektedir. Dolayısıyla 1,4,7,8 istekleri 0. isteğin tamamlanmasıyla işleme alınabilecek hale gelmiştir. Bunların içinden en az sürede işlemini bitirecek olan 8. istek öne geçerek 0. ana hattı, B(5) ve A(0)' ı ele geçirir.

1.sn sonra 3. isteğin 14 saniyesi, 5. isteğin 6 saniyesi, 6. isteğin 7 saniyesi 8. isteğin 2 saniyesi kalmıştır. Ara hatların boşalması beklenirken 9. istek olan (3,3,3) gelir. Ancak hem B(3) dolu olduğundan hem de ara hatlar boşalmadığı için bekleme sırasına alınır. Yeni bekleme listesi şöyle olur: 1,4,7,9. istekler.

2.sn sonra 3. isteğin 12 saniyesi, 5. isteğin 4 saniyesi, 6. isteğin 5 saniyesi kalmıştır ve 8. istek tamamlanmıştır.

Bu durumda 0. ara hat, B(5) ve A(0) serbest kalır. 8. isteğin tamamlanmasıyla birlikte 1,7. istekler işleme alınabilir duruma gelir. Bunlar arasından en kısa sürecek olan 1.istek, 0. ara hat işlemin tamamlanmasından 1 saniye sonra serbest kalacağından 1 saniye sonra işleme alınır. Ve tüm ara hatlar yeniden dolmuş olur. 3. isteğin 11 saniyesi, 5. isteğin 3 saniyesi, 6. isteğin 4 saniyesi,1. isteğin 10 saniyesi var. Bu durumda 7. istek ve 4,9. istekler ayrı olarak değerlendirilmelidir. Çünkü 7. istek ara hattın boşalmasını beklerken 4 ve 9. istekler B(3)'ün boşalmasını beklemektedir. 3 saniye sonra 5. isteğin sonlanır ve 1 saniye sonrasında boş ara hat oluşur ve 7. istek işleme alınır. Bu durumda 3. isteğin 7 saniyesi, 1. isteğin 6 saniyesi ve 7. isteğin 20 saniyesi var. Ve 6. istek tamamlandı Bu durumdan 1 saniye sonra 6. istek ara hattı serbest bırakır. Ara hat boşalmasına rağmen 4 ve 9 sırada beklemeye devam eder çünkü bu iki isteğin bağlanması için 3. isteğin tamamlanması gerekir. 5 saniye sonra 1. istek ve 6 saniye sonra 3. istek tamamlanır. 3. İsteğin tamamlanmasıyla 1 saniye sonra B(3) serbest kalır ve 4 ve 9 arasından daha kısa olan 9. istek işleme alınır. Bu durumda 7. isteğin 12 saniyesi, 9. isteğin 3 saniyesi vardır. 3 saniye sonunda 9. istek tamamlanır ancak B(3) hedefi 1saniye sonra serbest kalır B(3) hedefi serbest kalır ve 4. istek 12 saniye boyunca gerçekleştirilir ve bütün bağlantılar gerçekleşir.

ÖRNEK DO-SCRIPT DOSYASI

```
# Do script
quit -sim
vcom projec1.vhd
vsim -gui work.tsantral
add wave *
force -freeze sim:/tsantral/s 0 0, 1 {50 ps} -r 100
force -freeze sim:/tsantral/A 11111111 0
force -freeze sim:/tsantral/k 7 0
force -freeze sim:/tsantral/h 5 0
force -freeze sim:/tsantral/t 6 0
force -freeze sim:/ tsantral /k 6 0
force -freeze sim:/ tsantral /h 2 0
force -freeze sim:/ tsantral /t 10 0
force -freeze sim:/tsantral/k 3 0
force -freeze sim:/tsantral/h 1 0
force -freeze sim:/tsantral/t 2 0
run force -freeze sim:/tsantral/k 7 0
```

force -freeze sim:/tsantral/h 3 0

force -freeze sim:/tsantral/t 20 0 run

force -freeze sim:/tsantral/k 5 0

force -freeze sim:/tsantral/h 3 0

force -freeze sim:/tsantral/t 12 0

force -freeze sim:/tsantral/k 2 0

force -freeze sim:/tsantral/h 4 0

force -freeze sim:/tsantral/t 10 0

force -freeze sim:/tsantral/k 1 0

force -freeze sim:/tsantral/h 1 0

force -freeze sim:/tsantral/t 10 0 run

force -freeze sim:/tsantral/k 4 0

force -freeze sim:/tsantral/h 7 0

force -freeze sim:/tsantral/t 20 0

run

force -freeze sim:/tsantral/k 0 0

force -freeze sim:/tsantral/h 5 0

force -freeze sim:/tsantral/t 3 0 run

force -freeze sim:/tsantral/k 3 0

force -freeze sim:/tsantral/h 3 0

force -freeze sim:/tsantral/t 3 0