Tahmini Ders İçeriği
(Tentative Couse Schedule – Syllabus)

TOBB ETÜ
Ekonomi ve Teknoloji Üniversitesi

1. Hafta: Sayısal Sinyaller/Sistemler, İkilik Tabanda Sayılar, Taban Aritmetiği, İşaretli/Eksi Sayıların Gösterimi, Sayısal Tasarım Tarihçesi

2. Hafta: İkili Mantık Aritmetiği ve Kapıları, Bool Cebiri Teorisi ve Tanımları, Bool Fonksiyonları, Kapı-Seviyesinde Yalınlaştırma, Karnough Haritası, Önemsenmeyen Durumlar, NAND, NOR, XOR

3-4. Hafta: FPGA, Birleşik (Combinational) Devreler, Aritmetik Modüller, Decoder, Encoder, Mux, Verilog HDL

5. Hafta: Ardışık (Sequential) Devreler, Mandal (Latch), Flip-Flop, Yazmaçlar (Registers)

Lab Sınavı (265/264L)

6. Hafta: Durum Makinaları, Örnek Tasarımlar, Sayaçlar (Counters)

7. Hafta: FSM Örnekleri

8. Hafta: RTL (Register Transfer Level) ASMD (Algorithmic State Machine and Datapath) Tasarımları

9. Hafta: Durağan Zaman Analizi (Static Timing Analysis)

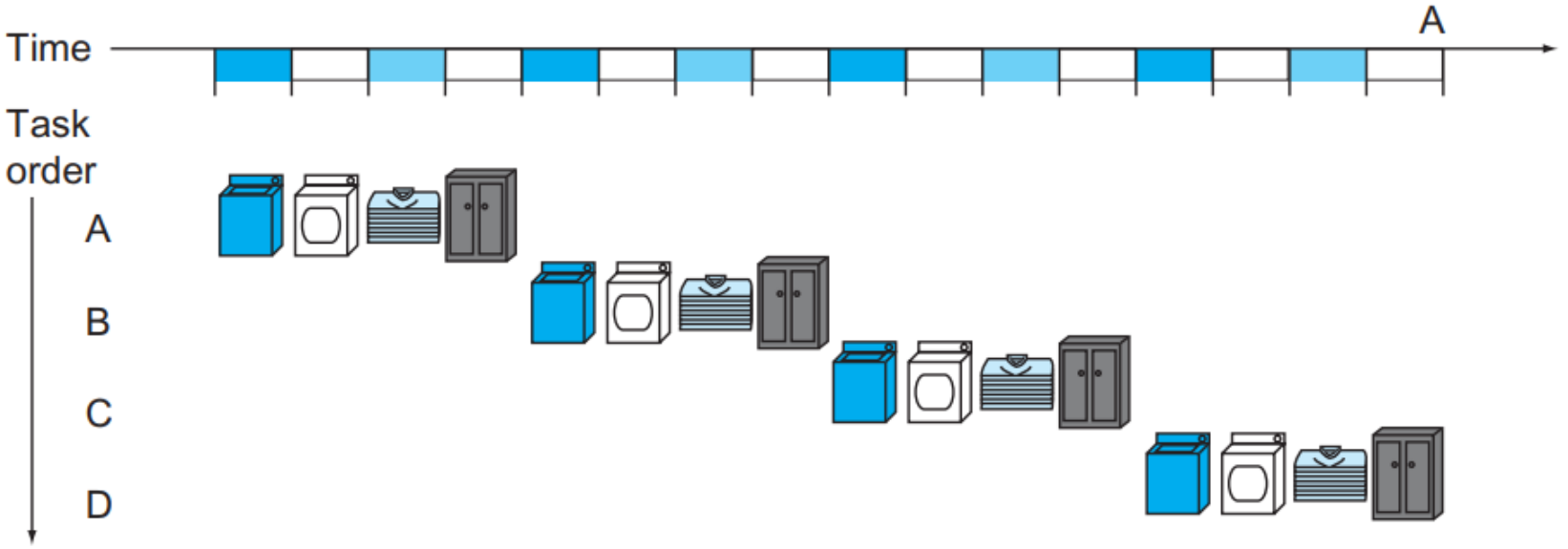
Ara Sınav (265/264)

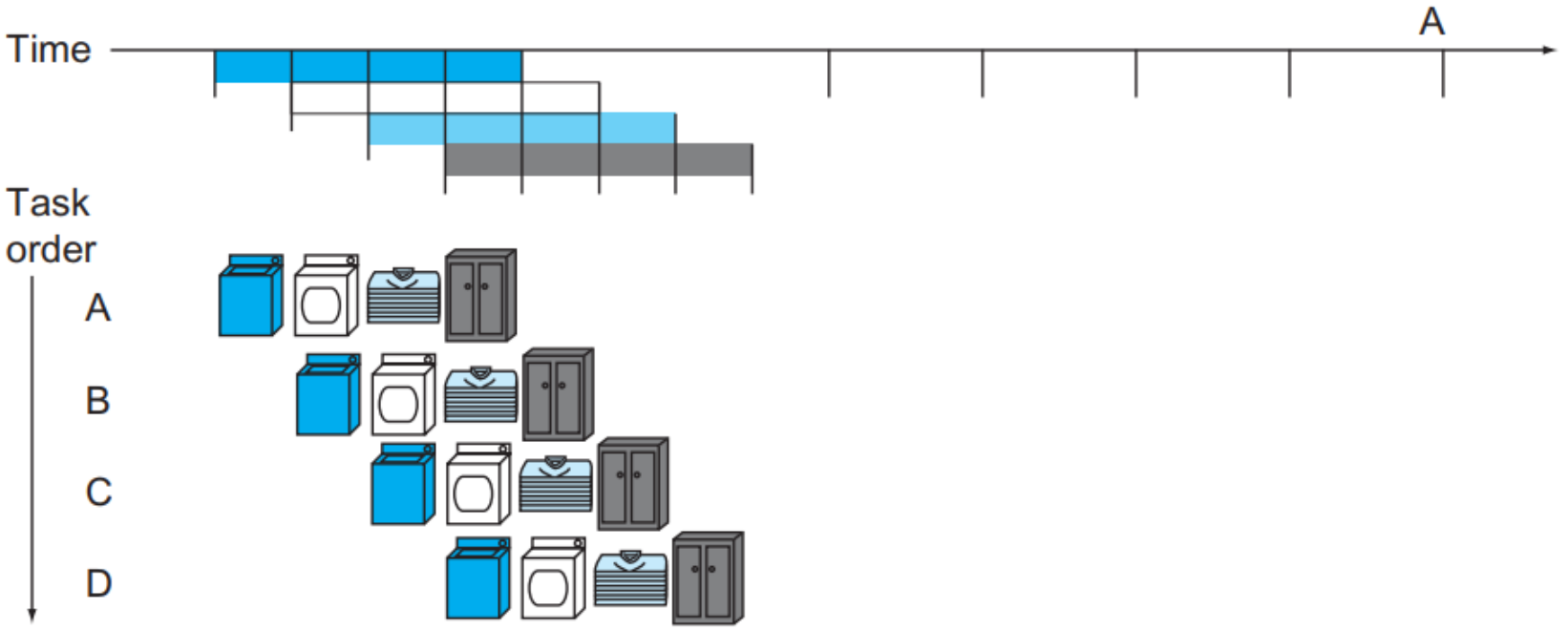
10. Hafta: Bellekler, FPGA'da RAM, OpenRAM

11-12. Hafta: **(21-25 Kasım, 28 Kasım – 2 Aralık)** Boru hattı, FPGA ve ASIC Tasarım Akışları

Final **(16 Aralık)** – Proje Teslimleri **(31 Aralık)**

BORUHATTI (PIPELINING)





Hennessy, Patterson kitabından

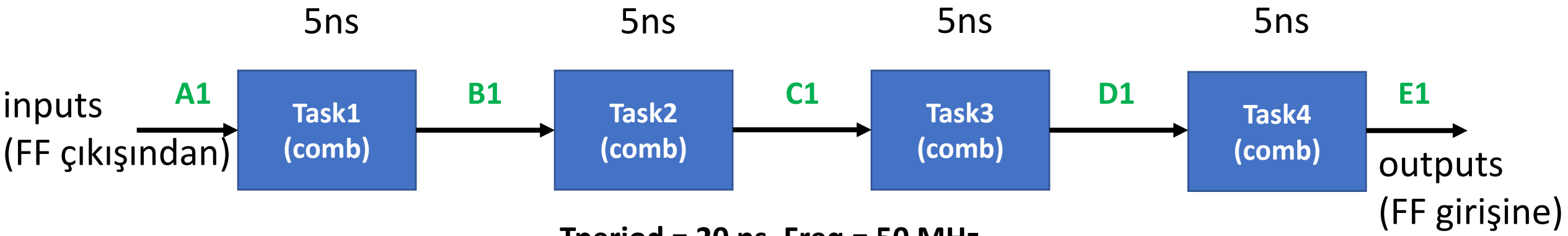
BORUHATTI (PIPELINING)

TOBB ETÜ
Ekonomi ve Teknoloji Üniversitesi

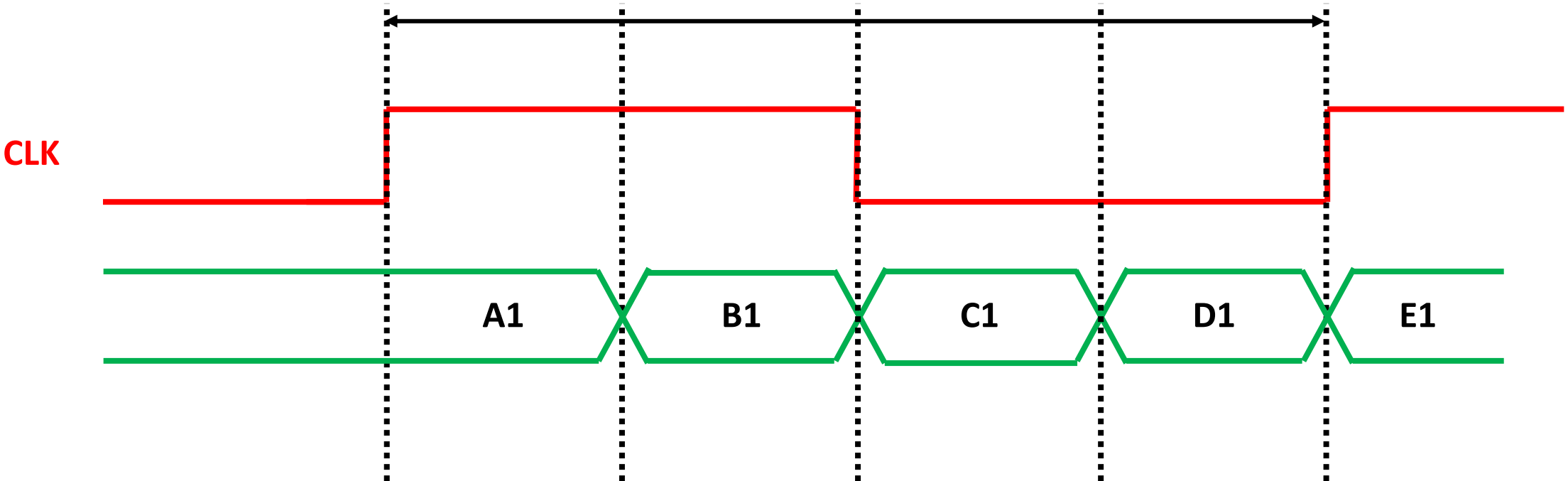
BORUHATTI (PIPELINING)

TOBB ETÜ
Ekonomi ve Teknoloji Üniversitesi

Trouting, Tcq ve Tsetup ignore edilmiştir

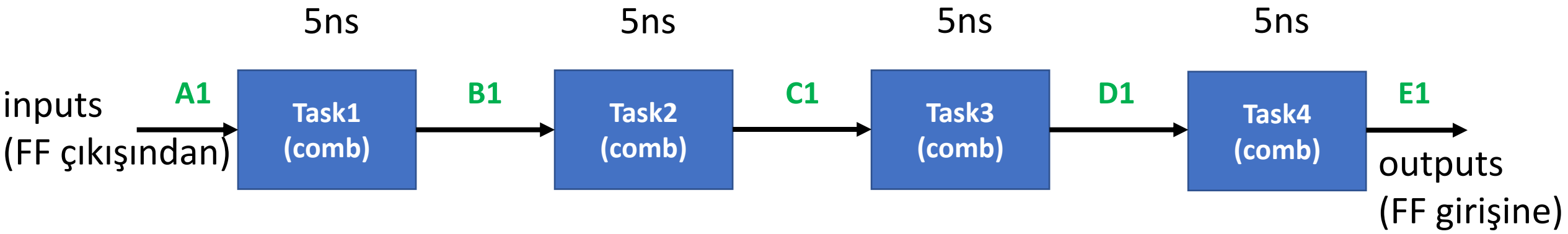


Tperiod = 20 ns, Freq = 50 MHz



BORUHATTI (PIPELINING)

TOBB ETÜ
Ekonomi ve Teknoloji Üniversitesi

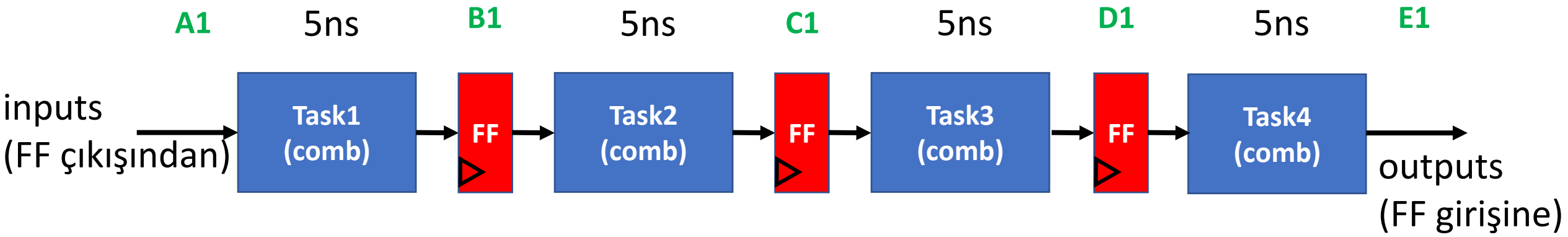


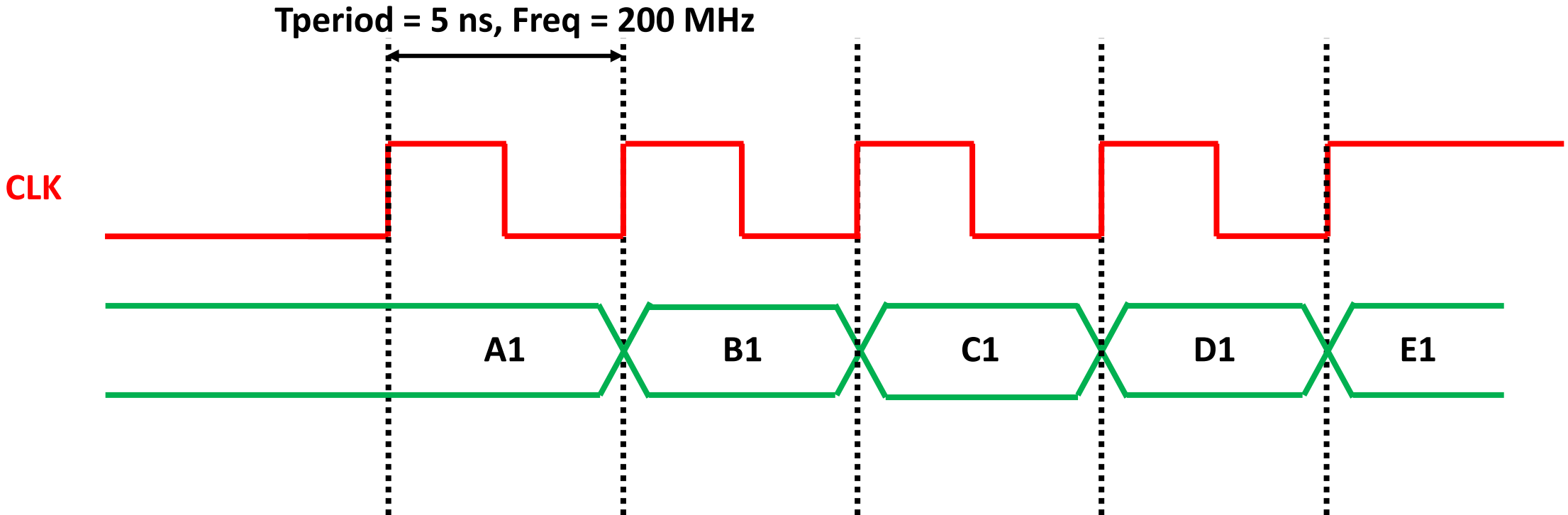
	0	5	10	15	20	25	30	35	40	45
A1	Task1	Task2	Task3	Task4						
A2					Task1	Task2	Task3	Task4		
A3									Task1	
A4										
A5										
A6										
A7										
A8										

BORUHATTI (PIPELINING)

TOBB ETÜ
Ekonomi ve Teknoloji Üniversitesi

A1 girişinin E1'e dönüşmesi yine 20 ns sürdü, yani toplam gecikme azalmadı !

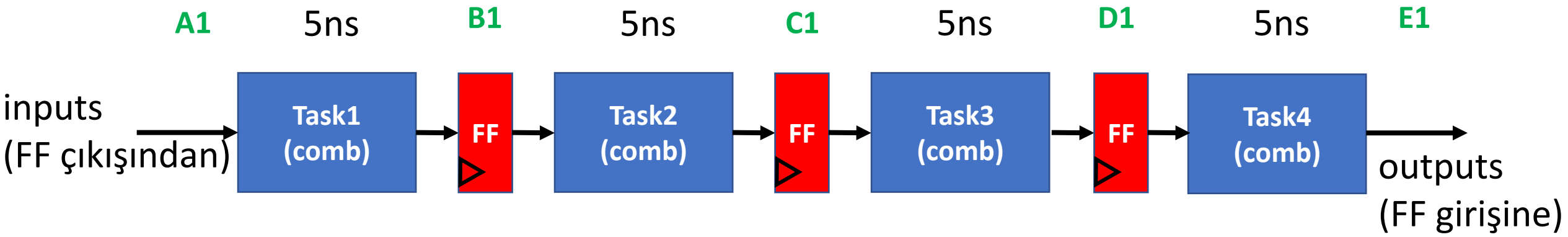




BORUHATTI (PIPELINING)

TOBB ETÜ
Ekonomi ve Teknoloji Üniversitesi

A1 girişinin E1'e dönüşmesi yine 20 ns sürdü, yani toplam gecikme azalmadı !
Ama toplamda sistemin tamamladığı görev, yaptığı işlem artmış oldu !

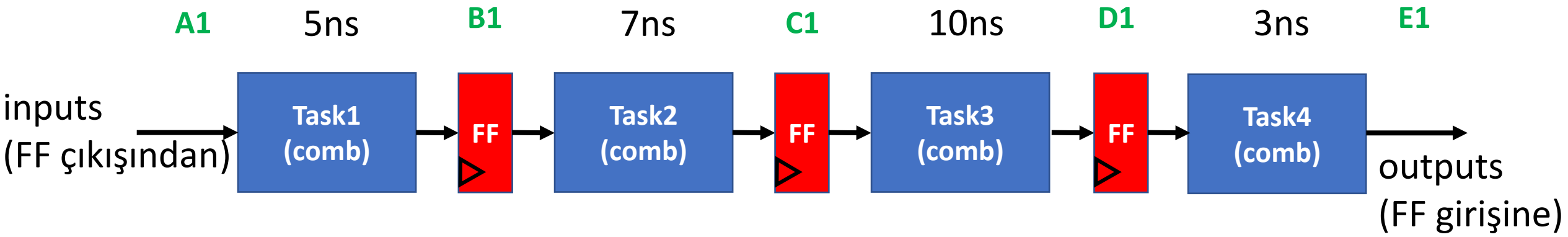


0	5	10	15	20	25	30	35	40	45
A1	Task1	Task2	Task3	Task4					
A2		Task1	Task2	Task3	Task4				
A3			Task1	Task2	Task3	Task4			
A4				Task1	Task2	Task3	Task4		
A5					Task1	Task2	Task3	Task4	
A6						Task1	Task2	Task3	Task4
A7							Task1	Task2	Task3
A8								Task1	Task2

BORUHATTI (PIPELINING)

TOBB ETÜ
Ekonomi ve Teknoloji Üniversitesi

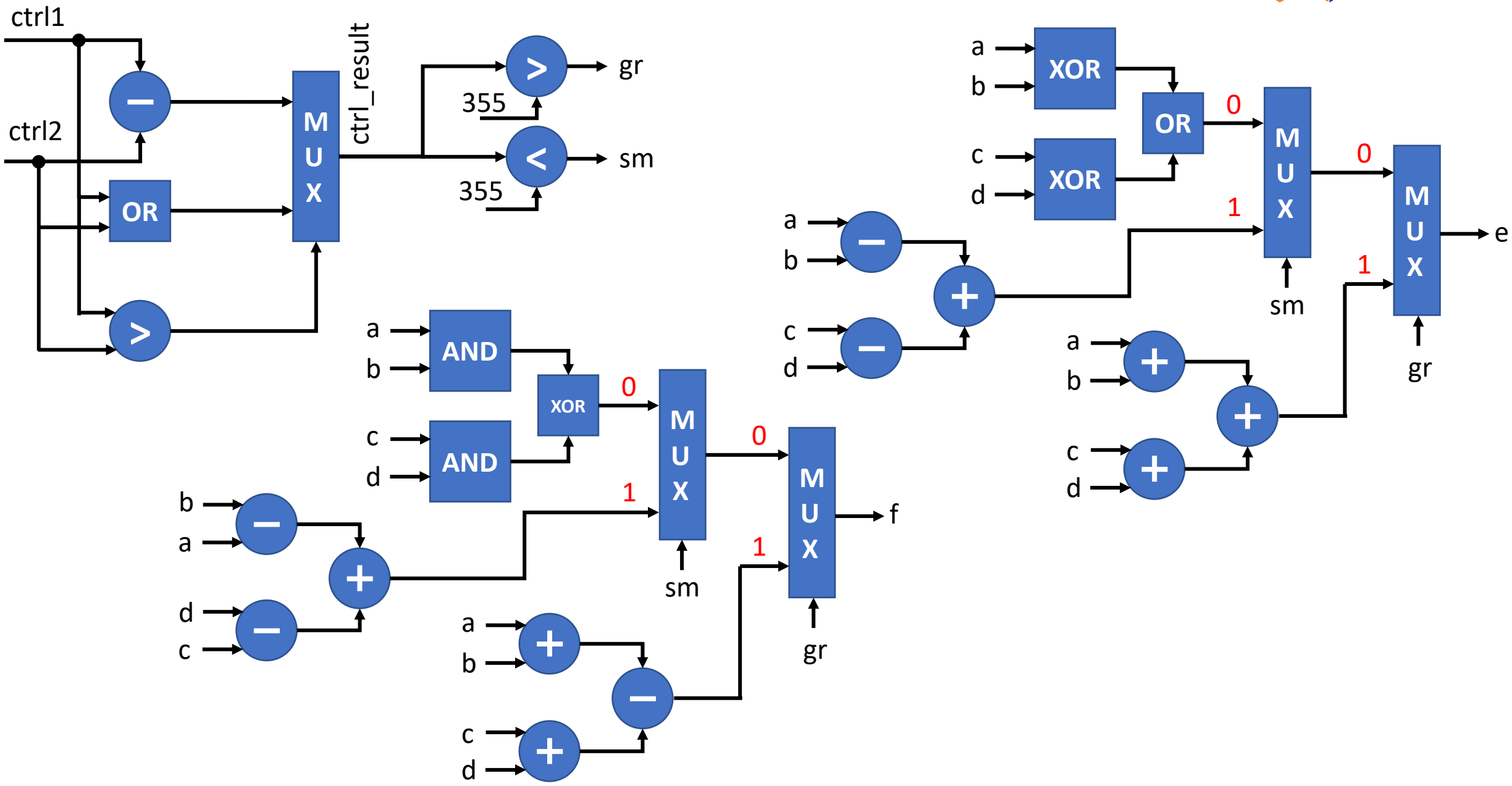
Peki ya bütün tasklar aynı gecikmeye sahip olmasaydı ???
En yavaş task'a göre saat frekansı güncellenir ve verimsizlik meydana gelirdi !



0	10	20	30	40	50	60	70	80	90
A1	Task1	Task2	Task3	Task4					
A2		Task1	Task2	Task3	Task4				
A3			Task1	Task2	Task3	Task4			
A4				Task1	Task2	Task3	Task4		
A5					Task1	Task2	Task3	Task4	
A6						Task1	Task2	Task3	Task4
A7							Task1	Task2	Task3
A8								Task1	Task2

BORUHATTI (PIPELINING) ÖRNEK TASARIM

TOBB ETÜ
Ekonomi ve Teknoloji Üniversitesi



BORUHATTI (PIPELINING) ÖRNEK TASARIM

TOBB ETÜ
Ekonomi ve Teknoloji Üniversitesi

```
module pipeline
(
input clk,
input rst_n,
input [7:0] a_i,
input [7:0] b_i,
input [7:0] c_i,
input [7:0] d_i,
input [7:0] ctrl1_i,
input [7:0] ctrl2_i,
output [9:0] e_o,
output [9:0] f_o
);
```

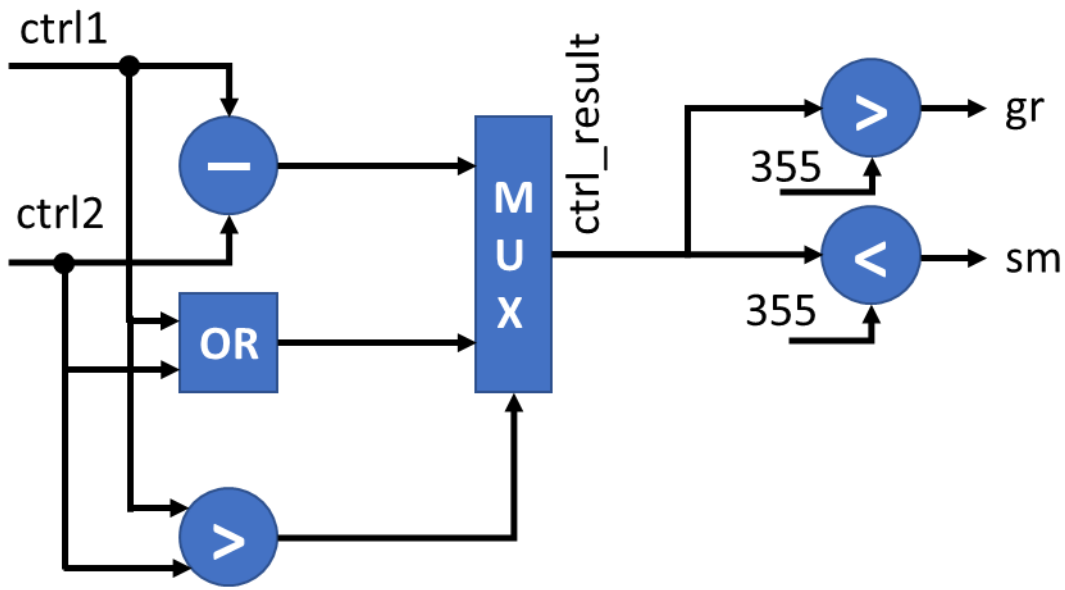
```
reg [9:0] e,f;
reg [7:0] a,b,c,d;
reg [7:0] ctrl1,ctrl2;
reg [8:0] ctrl_result;
reg gr,sm;
```

```
always @(*) begin

    if (ctrl1 > ctrl2)
        ctrl_result = ctrl1 - ctrl2;
    else
        ctrl_result = ctrl1 | ctrl2;

    gr = 0;
    sm = 0;
    if (ctrl_result > 9'd355) begin
        gr = 1;
    end
    else if (ctrl_result < 9'd355) begin
        sm = 1;
    end

end
```



```
always @(posedge clk, negedge rst_n) begin
    if (rst_n == 1'b0) begin
        a       <= {8{1'b0}};
        b       <= {8{1'b0}};
        c       <= {8{1'b0}};
        d       <= {8{1'b0}};
        ctrl1   <= {8{1'b0}};
        ctrl2   <= {8{1'b0}};
        e       <= {10{1'b0}};
        f       <= {10{1'b0}};
    end
    else begin
        a       <= a_i;
        b       <= b_i;
        c       <= c_i;
        d       <= d_i;
        ctrl1   <= ctrl1_i;
        ctrl2   <= ctrl2_i;
        if (gr) begin
            e   <= (a + b) + (c + d);
            f   <= (a + b) - (c + d);
        end
        else if (sm) begin
            e   <= (a - b) + (c - d);
            f   <= (-a + b) + (-c + d);
        end
        else begin
            e   <= {2'b00, (a ^ b) | (c ^ d)};
            f   <= {2'b11, (a & b) ^ (c & d)};
        end
    end
```

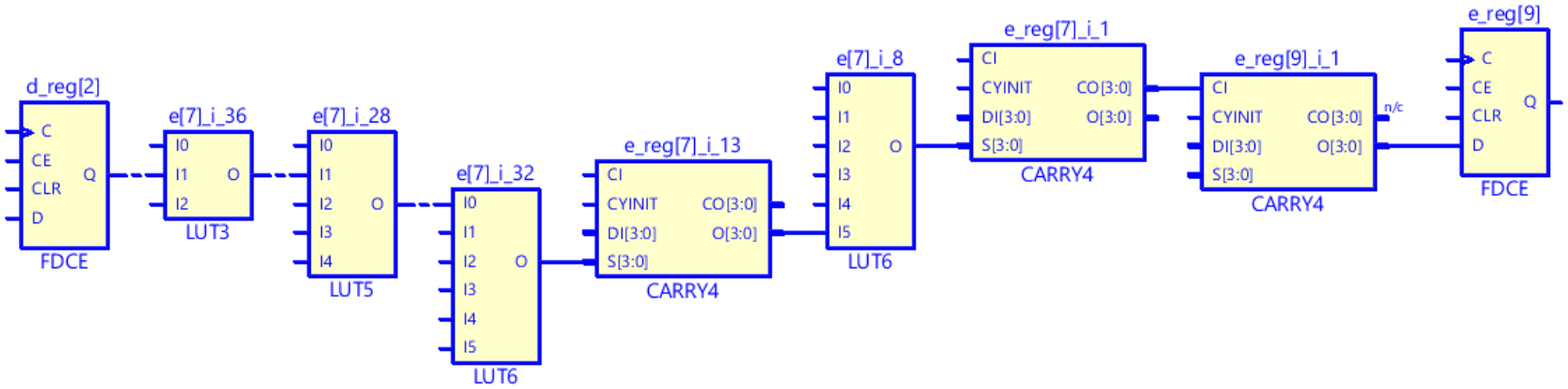
```
assign e_o = e;
assign f_o = f;

endmodule
```

Design Timing Summary

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	-0.244 ns	Worst Hold Slack (WHS):	0.633 ns	Worst Pulse Width Slack (WPWS):	2.000 ns
Total Negative Slack (TNS):	-1.078 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	0.000 ns
Number of Failing Endpoints:	8	Number of Failing Endpoints:	0	Number of Failing Endpoints:	0
Total Number of Endpoints:	20	Total Number of Endpoints:	20	Total Number of Endpoints:	69

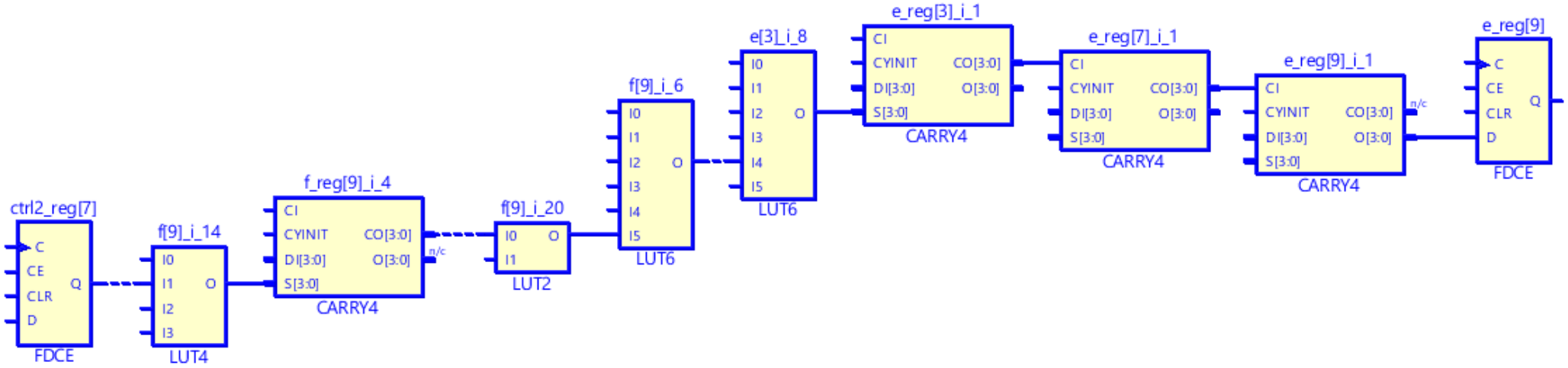
Timing constraints are not met.



Design Timing Summary

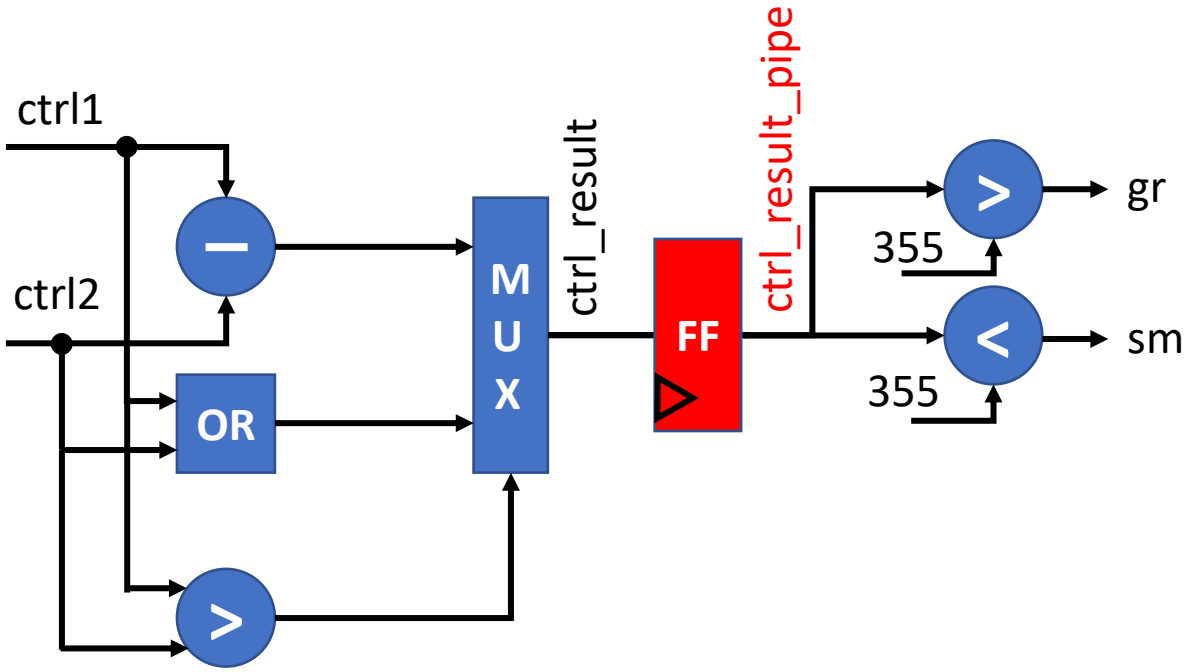
Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	-0.445 ns	Worst Hold Slack (WHS):	0.582 ns	Worst Pulse Width Slack (WPWS):	2.000 ns
Total Negative Slack (TNS):	-1.983 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	0.000 ns
Number of Failing Endpoints:	8	Number of Failing Endpoints:	0	Number of Failing Endpoints:	0
Total Number of Endpoints:	20	Total Number of Endpoints:	20	Total Number of Endpoints:	69

Timing constraints are not met.

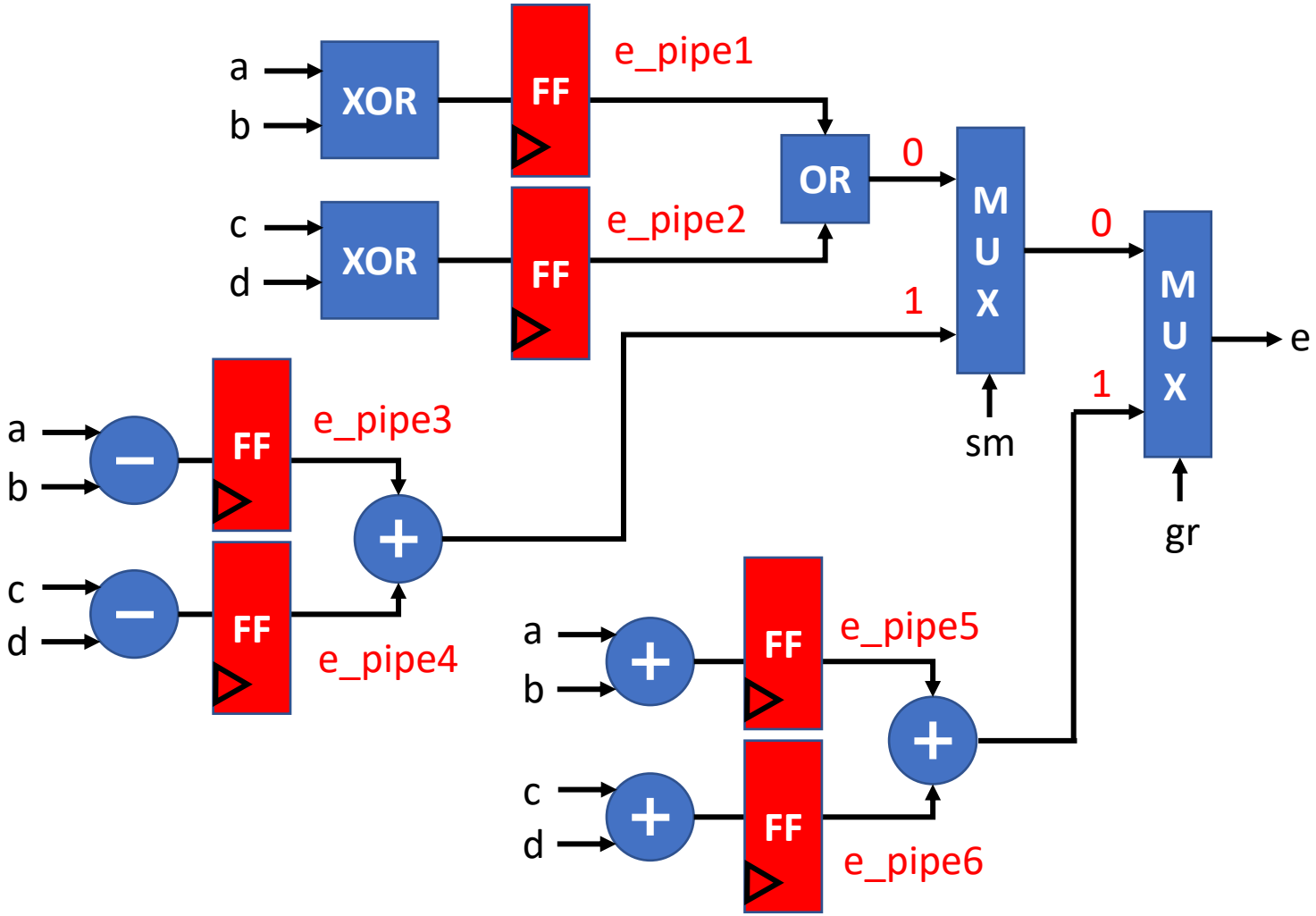


ÇÖZÜM → BORUHATTI (PIPELINING)

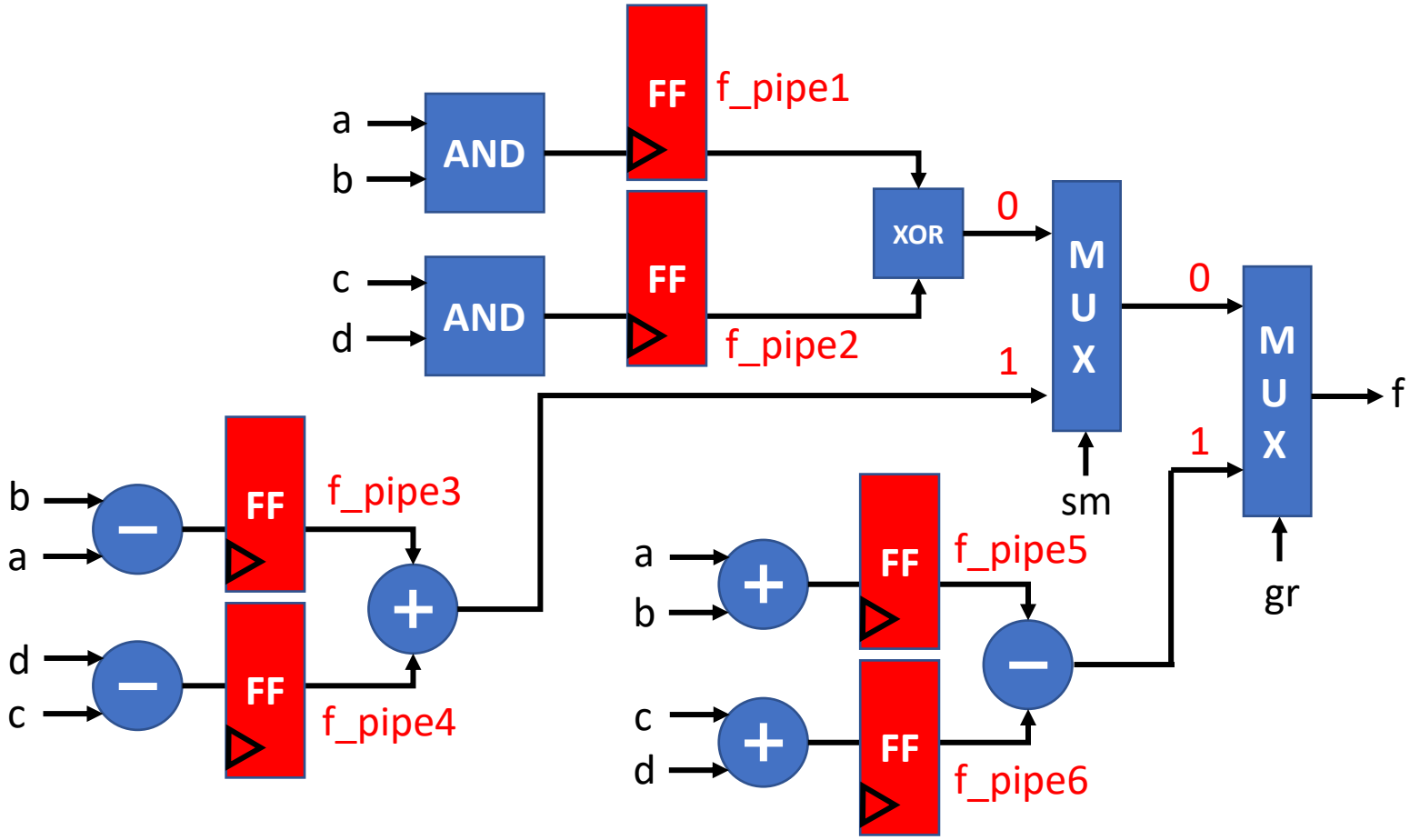
BORUHATTI (PIPELINING) ÖRNEK TASARIM



BORUHATTI (PIPELINING) ÖRNEK TASARIM



BORUHATTI (PIPELINING) ÖRNEK TASARIM



```
// registers for pipeline
reg [8:0] ctrl_result_pipe;
reg [7:0] e_pipe1, e_pipe2;
reg [8:0] e_pipe3, e_pipe4, e_pipe5, e_pipe6;
reg [7:0] f_pipe1, f_pipe2;
reg [8:0] f_pipe3, f_pipe4, f_pipe5, f_pipe6;
```

```
always @(*) begin

    if (ctrl1 > ctrl2)
        ctrl_result = ctrl1 - ctrl2;
    else
        ctrl_result = ctrl1 | ctrl2;

    gr = 0;
    sm = 0;
//    if (ctrl_result > 9'd355) begin
//        gr = 1;
//    end
//    else if (ctrl_result < 9'd355) begin
//        sm = 1;
//    end
    if (ctrl_result_pipe > 9'd355) begin
        gr = 1;
    end
    else if (ctrl_result_pipe < 9'd355) begin
        sm = 1;
    end

end
```

```
always @(posedge clk, negedge rst_n) begin
    if (rst_n == 1'b0) begin
        a               <= {8{1'b0}};
        b               <= {8{1'b0}};
        c               <= {8{1'b0}};
        d               <= {8{1'b0}};
        ctrl1           <= {8{1'b0}};
        ctrl2           <= {8{1'b0}};
        e               <= {10{1'b0}};
        f               <= {10{1'b0}};
        // registers for pipeline
        ctrl_result_pipe    <= {9{1'b0}};
        e_pipe1             <= {8{1'b0}};
        e_pipe2             <= {8{1'b0}};
        e_pipe3             <= {9{1'b0}};
        e_pipe4             <= {9{1'b0}};
        e_pipe5             <= {9{1'b0}};
        e_pipe6             <= {9{1'b0}};
        f_pipe1             <= {8{1'b0}};
        f_pipe2             <= {8{1'b0}};
        f_pipe3             <= {9{1'b0}};
        f_pipe4             <= {9{1'b0}};
        f_pipe5             <= {9{1'b0}};
        f_pipe6             <= {9{1'b0}};
    end
    else begin
```

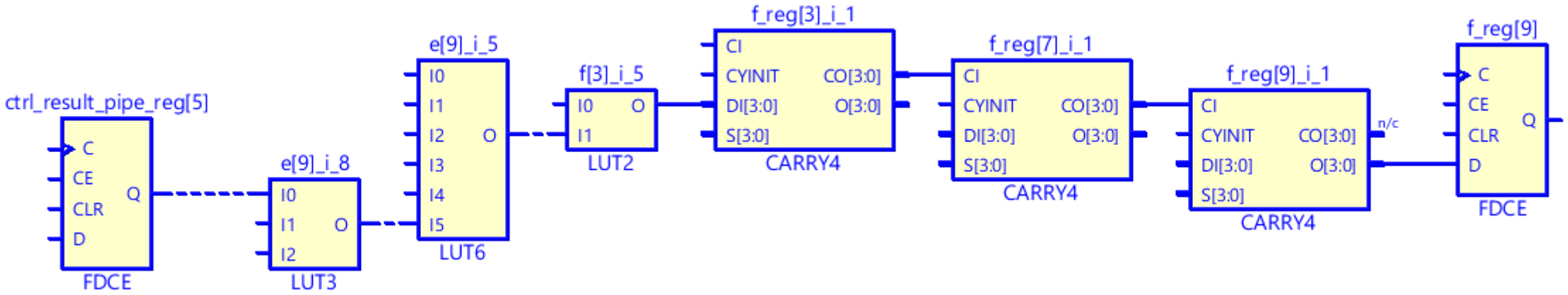
```
else begin
    a       <= a_i;
    b       <= b_i;
    c       <= c_i;
    d       <= d_i;
    ctrl1   <= ctrl1_i;
    ctrl2   <= ctrl2_i;
    // registers for pipeline
    ctrl_result_pipe    <= ctrl_result;
    e_pipe1             <= (a ^ b);
    e_pipe2             <= (c ^ d);
    e_pipe3             <= (a - b);
    e_pipe4             <= (c - d);
    e_pipe5             <= (a + b);
    e_pipe6             <= (c + d);
    f_pipe1             <= (a & b);
    f_pipe2             <= (c & d);
    f_pipe3             <= (-a + b);
    f_pipe4             <= (-c + d);
    f_pipe5             <= (a + b);
    f_pipe6             <= (c + d);
```

```
//          if (gr) begin
//              e   <= (a + b) + (c + d);
//              f   <= (a + b) - (c + d);
//          end
//          else if (sm) begin
//              e   <= (a - b) + (c - d);
//              f   <= (-a + b) + (-c + d);
//          end
//          else begin
//              e   <= {2'b00, (a ^ b) | (c ^ d)};
//              f   <= {2'b11, (a & b) ^ (c & d)};
//          end
        if (gr) begin
            e   <= e_pipe5 + e_pipe6;
            f   <= f_pipe5 - f_pipe6;
        end
        else if (sm) begin
            e   <= e_pipe3 + e_pipe4;
            f   <= f_pipe3 + f_pipe4;
        end
        else begin
            e   <= {2'b00, e_pipe1 | e_pipe2};
            f   <= {2'b11, f_pipe1 ^ f_pipe2};
        end

    end

end
```

BORUHATTI (PIPELINING) ÖRNEK TASARIM



Design Timing Summary

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	0.474 ns	Worst Hold Slack (WHS):	0.227 ns	Worst Pulse Width Slack (WPWS):	2.000 ns
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	0.000 ns
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0	Number of Failing Endpoints:	0
Total Number of Endpoints:	115	Total Number of Endpoints:	115	Total Number of Endpoints:	164

All user specified timing constraints are met.

Normal

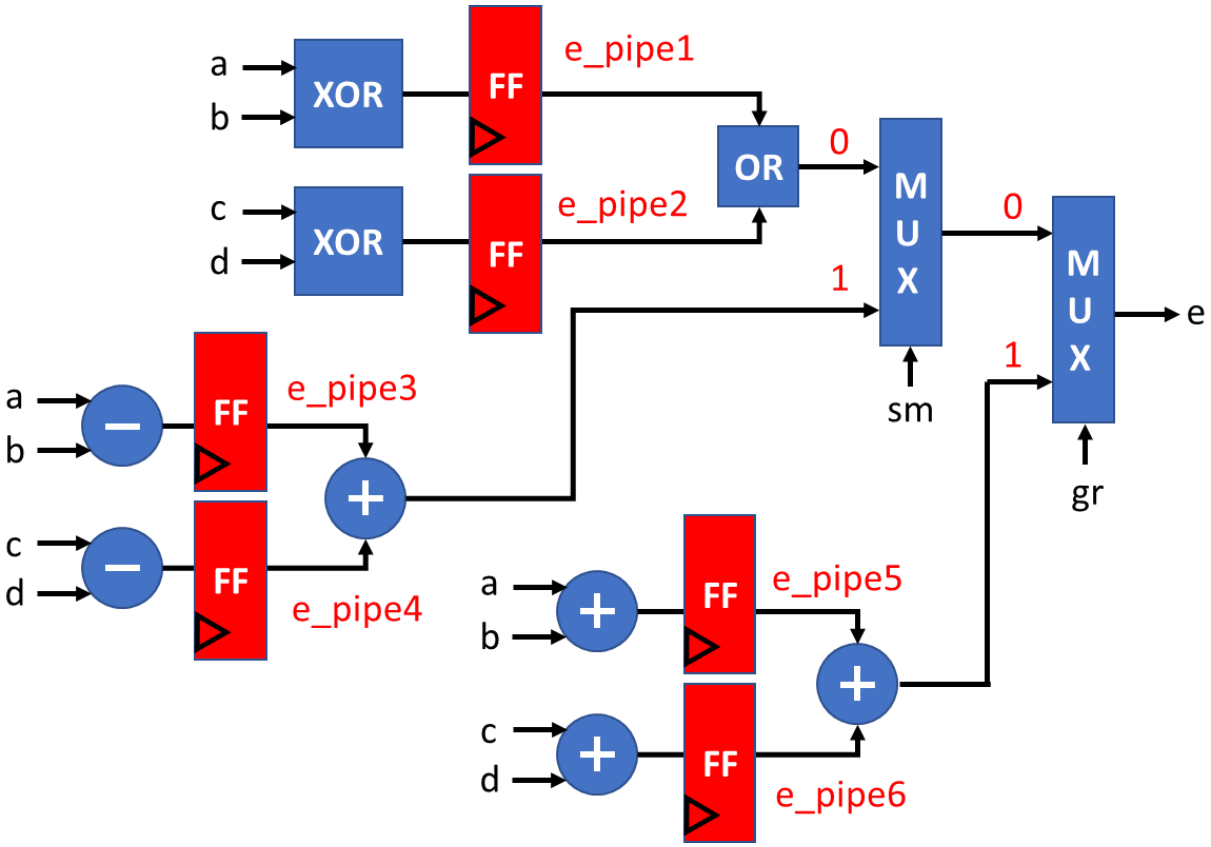
Resource	Utilization	Available	Utilization %
LUT	118	20800	0.57
FF	68	41600	0.16
IO	70	106	66.04
BUFG	1	32	3.13

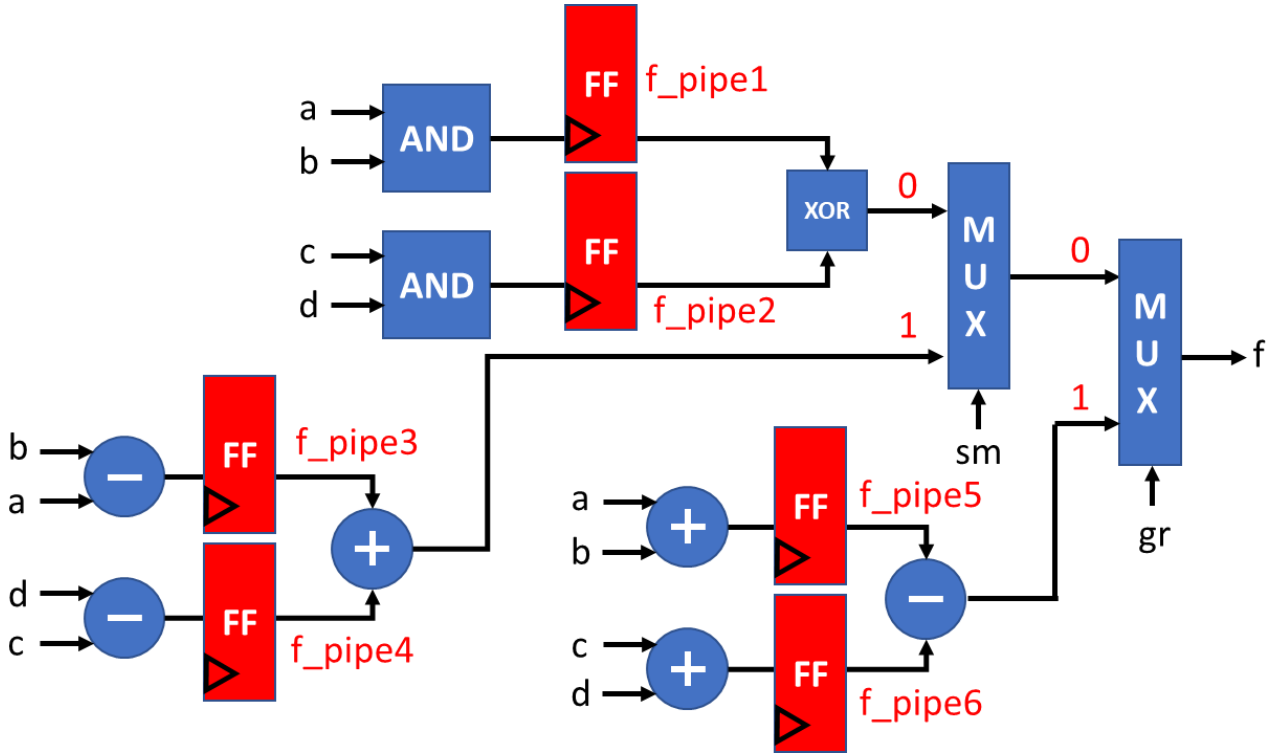
Pipelined

Resource	Utilization	Available	Utilization %
LUT	148	20800	0.71
FF	163	41600	0.39
IO	70	106	66.04
BUFG	1	32	3.13

KAYNAK PAYLAŞIMI (RESOURCE SHARING)

TOBB ETÜ
Ekonomi ve Teknoloji Üniversitesi





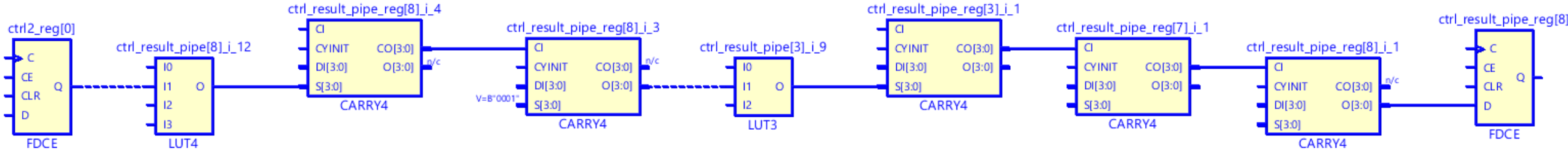
```
// registers for pipeline
reg [8:0] ctrl_result_pipe;
reg [7:0] e_pipe1, e_pipe2;
reg [8:0] e_pipe3, e_pipe4, e_pipe5, e_pipe6;
reg [7:0] f_pipe1, f_pipe2;
//reg [8:0] f_pipe3, f_pipe4, f_pipe5, f_pipe6;
```

```
always @(posedge clk, negedge rst_n) begin
    if (rst_n == 1'b0) begin
        a       <= {8{1'b0}};
        b       <= {8{1'b0}};
        c       <= {8{1'b0}};
        d       <= {8{1'b0}};
        ctrl1   <= {8{1'b0}};
        ctrl2   <= {8{1'b0}};
        e       <= {10{1'b0}};
        f       <= {10{1'b0}};
        // registers for pipeline
        ctrl_result_pipe    <= {9{1'b0}};
        e_pipe1             <= {8{1'b0}};
        e_pipe2             <= {8{1'b0}};
        e_pipe3             <= {9{1'b0}};
        e_pipe4             <= {9{1'b0}};
        e_pipe5             <= {9{1'b0}};
        e_pipe6             <= {9{1'b0}};
        f_pipe1             <= {8{1'b0}};
        f_pipe2             <= {8{1'b0}};
//        f_pipe3             <= {9{1'b0}};
//        f_pipe4             <= {9{1'b0}};
//        f_pipe5             <= {9{1'b0}};
//        f_pipe6             <= {9{1'b0}};
    end
    else begin
```

```
    // registers for pipeline
    ctrl_result_pipe    <= ctrl_result;
    e_pipe1             <= (a ^ b);
    e_pipe2             <= (c ^ d);
    e_pipe3             <= (a - b);
    e_pipe4             <= (c - d);
    e_pipe5             <= (a + b);
    e_pipe6             <= (c + d);
    f_pipe1             <= (a & b);
    f_pipe2             <= (c & d);
      f_pipe3             <= (-a + b);
      f_pipe4             <= (-c + d);
      f_pipe5             <= (a + b);
      f_pipe6             <= (c + d);
```

```
if (gr) begin
    e   <= e_pipe5 + e_pipe6;
    f   <= e_pipe5 - e_pipe6;
end
else if (sm) begin
    e   <= e_pipe3 + e_pipe4;
    f   <= -e_pipe3 - e_pipe4;
end
else begin
    e   <= {2'b00, e_pipe1 | e_pipe2};
    f   <= {2'b11, f_pipe1 ^ f_pipe2};
end
```

KAYNAK PAYLAŞIMI (RESOURCE SHARING)



Design Timing Summary

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	0.680 ns	Worst Hold Slack (WHS):	0.285 ns	Worst Pulse Width Slack (WPWS):	2.000 ns
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	0.000 ns
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0	Number of Failing Endpoints:	0
Total Number of Endpoints:	97	Total Number of Endpoints:	97	Total Number of Endpoints:	146

All user specified timing constraints are met.

Normal

Resource	Utilization	Available	Utilization %
LUT	118	20800	0.57
FF	68	41600	0.16
IO	70	106	66.04
BUFG	1	32	3.13

Pipelined

Resource	Utilization	Available	Utilization %
LUT	148	20800	0.71
FF	163	41600	0.39
IO	70	106	66.04
BUFG	1	32	3.13

Pipelined & Resource Shared

Resource	Utilization	Available	Utilization %
LUT	131	20800	0.63
FF	145	41600	0.35
IO	70	106	66.04
BUFG	1	32	3.13