



2023-2024 Yaz Dönemi BİL265/264 Proje

Hazırlayanlar:

Ad Soyad : Tuğberk Aksu No: 201101075

Ad Soyad : Ali Osman Çıtak No: 211101076

Projemizin amacı verilog dilinde BASYS3 fpga kullanarak bir saat oluşturmaktır. Saniye, dakika, saat, gün, ay, yıl değerleri hesaplanmaktadır. Dakika ve saat fbga üzerindeki 7 segment display’de gösterilmekte. Saniye ise ledler üzerinde görünmektedir. Projemiz 5 dosyadan oluşmaktadır. saatt.v BASYS3.xdc uartrx.v uartrx.v debounce.v dosyalarımızdır. saat.v ana modülümüzdür. Bu modül ana sayma işlemini yapmakta aynı zamanda uart modüllerinin de entegresini barındırmaktadır. Uart dosyalarımızı internette kaynak taraması yaparak bulunmuştur içerikleri tamamen bize ait değildir.

saat.v

```
reg [55:0] time_date_str;
// Otomatik saat ve tarih sayımı
always @(posedge clk2) begin
    if (counter) begin
        time_date_str = {year3, year2, year1, year0, month1, month0, day1, day0, hour1, hour0, min1, min0, sec1, sec0};
        sec0 <= sec0 + 1;
        if (sec0 == 9) begin
            sec0 <= 0;
            sec1 <= sec1 + 1;
            if (sec1 == 5) begin
                sec1 <= 0;
                min0 <= min0 + 1;
                if (min0 == 9) begin
                    min0 <= 0;
                    min1 <= min1 + 1;
                    if (min1 == 5) begin
                        min1 <= 0;
                        hour0 <= hour0 + 1;
                        if (hour0 == 9) begin
                            hour0 <= 0;
                            hour1 <= hour1 + 1;
                            end
                            if (hour1 == 2 && hour0 == 3) begin
                                hour1 <= 0;

```

Bu kod bloğu ana sayma mantığımızı barındırır. Örnek olarak saniyenin birler basamağı 9 olduğunda saniyenin onlar basamağının değişmesi vs.

```
//saati hızlandırma veya normal saniyeye döndürme
always @(posedge clk)begin
    if (clk_fast)begin
        count2=count2+10;
        if (count2 >= 5000)begin
            clk2=~clk2;
            count2 = 1;
        end
    end else begin
        count2=count2+1;
        if (count2 >= 50000000)begin
            clk2=~clk2;
            count2 = 1;
        end
    end
end

always @(posedge clk)begin
    count3=count3+1;
    if (count3 == 50000)begin
        clk3=~clk3;
        count3 = 1;
    end
end
end
```

Buradaki kod bloğu saatimizin hızını artırmakta kullandığımız kod bloğumuz.

```

reg counter=1;
reg stopped=0;
always @(posedge clk3 or posedge stop) begin
    if(stop)begin
        if(!stopped)begin
            counter=0;
            stopped=1;
        end else begin
            counter=1;
            stopped=0;
        end
    end
end
end

```

Bu kod bloğu saati durdurmamızı sağlayan modül.

```

// LED'lere saniye göstermek
always @(posedge clk) begin
    led[5:0] <= {sec1[2:0], sec0};
    //$display("tarih: %d%d %d%d %d

end

```

Bu kod bloğu saniyeyi ledlere atayan modülümüzdür.

```

reg [5:0] sayac; // İki durumlu sayaç
reg [7:0] temp_value; // Geçici değer

always @(posedge clk) begin
    if (sayac == 0) begin
        // Saniyenin birler hanesi
        case (day1)
            4'd0: temp_value <= 8'h30; // '0'
            4'd1: temp_value <= 8'h31; // '1'
            4'd2: temp_value <= 8'h32; // '2'
            4'd3: temp_value <= 8'h33; // '3'
            4'd4: temp_value <= 8'h34; // '4'
            4'd5: temp_value <= 8'h35; // '5'
            4'd6: temp_value <= 8'h36; // '6'
            4'd7: temp_value <= 8'h37; // '7'
            4'd8: temp_value <= 8'h38; // '8'
            4'd9: temp_value <= 8'h39; // '9'
            default: temp_value <= 8'h30; // Varsayılan olarak '0'
        endcase
    end
end

```

Bu kod bloğu uarttx için data_in değerlerini temp_value'ya hexadecimal olarak hangi sayıya denk geldiğini test edip atıyoruz.

```
always @(posedge clk3)begin
  if(count==0)begin
    case(hour1)
      4'h0: ss_out = 7'b1000000; // 0
      4'h1: ss_out = 7'b1111001; // 1
      4'h2: ss_out = 7'b0100100; // 2
      4'h3: ss_out = 7'b0110000; // 3
      4'h4: ss_out = 7'b0011001; // 4
      4'h5: ss_out = 7'b0010010; // 5
      4'h6: ss_out = 7'b0000010; // 6
      4'h7: ss_out = 7'b1111000; // 7
      4'h8: ss_out = 7'b0000000; // 8
      4'h9: ss_out= 7'b0010000; // 9
    endcase
    en_out = 4'b0111;
    count = 1;
  end
end
```

Bu kod bloğu saatler ve dakikaları 7 segment displaye atama yapmaktadır.

BASYS3.xdc

```
set_property IOSTANDARD LVCMOS33 [get_ports {led[1]]}
set_property PACKAGE_PIN U19 [get_ports {led[2]]}
set_property IOSTANDARD LVCMOS33 [get_ports {led[2]]}
set_property PACKAGE_PIN V19 [get_ports {led[3]]}
set_property IOSTANDARD LVCMOS33 [get_ports {led[3]]}
set_property PACKAGE_PIN W18 [get_ports {led[4]]}
set_property IOSTANDARD LVCMOS33 [get_ports {led[4]]}
set_property PACKAGE_PIN U15 [get_ports {led[5]]}
set_property IOSTANDARD LVCMOS33 [get_ports {led[5]]}
set_property PACKAGE_PIN U14 [get_ports {led[6]]}
set_property IOSTANDARD LVCMOS33 [get_ports {led[6]]}
set_property PACKAGE_PIN V14 [get_ports {led[7]]}
set_property IOSTANDARD LVCMOS33 [get_ports {led[7]]}
set_property PACKAGE_PIN V13 [get_ports {led[8]]}
set_property IOSTANDARD LVCMOS33 [get_ports {led[8]]}
set_property PACKAGE_PIN V3 [get_ports {led[9]]}
set_property IOSTANDARD LVCMOS33 [get_ports {led[9]]}
set_property PACKAGE_PIN W3 [get_ports {led[10]]}
set_property IOSTANDARD LVCMOS33 [get_ports {led[10]]}
set_property PACKAGE_PIN U3 [get_ports {led[11]]}
set_property IOSTANDARD LVCMOS33 [get_ports {led[11]]}
set_property PACKAGE_PIN P3 [get_ports {led[12]]}
set_property IOSTANDARD LVCMOS33 [get_ports {led[12]]}
set_property PACKAGE_PIN N3 [get_ports {led[13]]}
set_property IOSTANDARD LVCMOS33 [get_ports {led[13]]}
set_property PACKAGE_PIN P1 [get_ports {led[14]]}
set_property IOSTANDARD LVCMOS33 [get_ports {led[14]]}
set_property PACKAGE_PIN L1 [get_ports {led[15]]}
set_property IOSTANDARD LVCMOS33 [get_ports {led[15]]}
```

Bu kod saniyeler için ledlere atama yapılan kod bloğu.

```
set_property PACKAGE_PIN V17 [get_ports {sw[0]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {sw[0]}]
set_property PACKAGE_PIN V16 [get_ports {sw[1]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {sw[1]}]
set_property PACKAGE_PIN W16 [get_ports {sw[2]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {sw[2]}]
set_property PACKAGE_PIN W17 [get_ports {sw[3]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {sw[3]}]
set_property PACKAGE_PIN W15 [get_ports {sw[4]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {sw[4]}]
set_property PACKAGE_PIN V15 [get_ports {sw[5]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {sw[5]}]
set_property PACKAGE_PIN W14 [get_ports {sw[6]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {sw[6]}]
```

Bu kod seven segment display için değerlerin atama yapıldığı kod bloğu.