```
module cpld1(f4m,
p3, p2, p4,
p12, p10, p8, p6, p16, p14);

    input f4m, p3, p2, p4;
    output p12, p10, p8, p6, p16, p14;

    reg[15:0] cnt;
    reg[1:0] sel;
    reg flag, flag2;

    assign {p12, p10, p8, p6} =
    (sel == 0) ? cnt[15:12]:
    (sel == 1) ? cnt[11:8]:
    (sel == 2) ? cnt[7:4]:
    cnt[3:0];

    assign {p16, p14} = sel;

    always@(posedge f4m)
        if (p3)
            cnt = 0;
        else
            if (p2)
                flag = 1;
            else
                if (flag)
                begin
                    flag = 0;
                    if (cnt == 16'h9999)
                        cnt = 0;
                    else if (cnt[11:0] == 12'h999)
                        cnt = cnt + 16'h0667;
                    else if (cnt[7:0] == 8'h99)
                        cnt = cnt + 16'h0067;
                    else if (cnt[3:0] == 4'h9)
                        cnt = cnt + 16'h0007;
                    else
                        cnt = cnt + 1;
                end

    always@(posedge f4m)
        if (p4)
            flag2 = 1;
        else
            if (flag2)
            begin
                flag2 = 0;
                sel = sel + 1;
            end

endmodule

```

```
module cpld2(f4m,
p10, p4, p2, p14, p13,
p7, p3, p5, p9, p15, p19, p16, p18, p6, p20, p1);

    input f4m, p10, p4, p2, p14, p13;
    output p7, p3, p5, p9, p15, p19, p16, p18, p6, p20, p1;
    wire en;

    assign p6 = (t == 18'h1ffff);
    assign p20 = (t == 18'h2ffff);
    assign p1 = (!p13) & (!p14);
    assign en = !(p2 & p4 & p10);

    reg[17:0] t;
    always@(posedge f4m)
        if (~en) t = 0;
        else t = t + (t != 18'h3ffff);

    reg[16:0] c;
    always@(posedge f4m)
        if (~en) c = c + 1;

    assign {p15, p19, p16, p18} =
    (c[16:15] == 0) ? 4'b1110:
    (c[16:15] == 1) ? 4'b1101:
    (c[16:15] == 2) ? 4'b1011:
    4'b0111;

    assign {p7, p3, p5, p9} =
    ({c[16:15], p2, p4, p10} == 5'b00011) ? 0:
    ({c[16:15], p2, p4, p10} == 5'b00101) ? 4:
    ({c[16:15], p2, p4, p10} == 5'b00110) ? 8:

    ({c[16:15], p2, p4, p10} == 5'b01011) ? 1:
    ({c[16:15], p2, p4, p10} == 5'b01101) ? 5:
    ({c[16:15], p2, p4, p10} == 5'b01110) ? 9:

    ({c[16:15], p2, p4, p10} == 5'b10011) ? 2:
    ({c[16:15], p2, p4, p10} == 5'b10101) ? 6:

    ({c[16:15], p2, p4, p10} == 5'b11011) ? 3:
    ({c[16:15], p2, p4, p10} == 5'b11101) ? 7:
    0;

endmodule

```

```
module cpld3(clr,
p19, p21, p5, p22, p7, p20,
p11, p12, p14, p15, p16, p17, p8, p4, p6);

    input clr, p19, p21, p5, p22, p7, p20;
    output p11, p12, p14, p15, p16, p17, p8, p4, p6;
    wire clk;

    reg ip5, ip21;
    always@(posedge p22) ip5 = p5;
    always@(posedge p22) ip21 = p21;

    assign clk = p20 ? ip21 & p22 : p19;
    assign p8 = p20 ? ip5 & p22 : p7;
    assign p4 = p20 ? 0 : p22;
    assign p6 = p20 ? 1 : p7;

    reg[5:0] hour;

    assign {p11, p12, p14, p15, p16, p17} = hour;

    always@(posedge clr or posedge clk)
        if (clr)
            hour = 0;
        else
            if (hour == 6'h23) hour = 0;
            else if (hour == 6'h19) hour = 6'h20;
            else if (hour == 6'h09) hour = 6'h10;
            else hour = hour + 1;

endmodule

```