

```

module voter(in1, in2, in3, out, error);
    input [1:0] in1, in2, in3;
    output [1:0] out;
    output error;

    wire [1:0] in_bits [0:3];
    wire [1:0] inequal [0:2];
    wire e [0:2];
    wire ne [0:2];
    wire sel [0:3];

    buf buff_1 [1:0] (in_bits[0], in1);
    buf buff_2 [1:0] (in_bits[1], in2);
    buf buff_3 [1:0] (in_bits[2], in3);
    buf buff_4 [1:0] (in_bits[3], 2'bxx);

    genvar i;
    generate for (i = 0; i < 3; i = i + 1) begin: xor_loop
        xor ieq [1:0] (inequal[i], in_bits[i], in_bits[(i + 1) % 3]);
    end
endgenerate

    genvar j;
    generate for (j = 0; j < 3; j = j + 1) begin: error_loop
        or (e[j], inequal[j][0], inequal[j][1]);
        not (ne[j], e[j]);
    end
endgenerate

    nor (error, ne[0], ne[1], ne[2]);

    buf (sel[0], ne[0]);
    and (sel[1], ne[1], e[0]);
    and (sel[2], ne[2], e[1], e[0]);
    and (sel[3], e[2], e[1], e[0]);

    genvar k;
    generate for (k = 0; k < 4; k = k + 1) begin: out_loop
        buf1 tribuff [1:0] (out, in_bits[k], sel[k]);
    end
endgenerate
endmodule

```

برای حل این سوال، می‌خواهیم از generate استفاده کنیم. بنابراین در ابتدای کار سه ورودی in را در آرایه in\_bits قرار می‌دهیم تا بتوانیم از طریق نمایه به آنها دسترسی داشته باشیم. همچنین عضو چهارم آرایه را نیز xx در نظر می‌گیریم تا در صورت وقوع error آن را خروجی دهیم.

توجه کنید که XOR دو بیت اگر آن دو برابر باشند برابر صفر و در غیر این صورت برابر ۱ خواهد بود. ابتدا به صورت بیت به بیت XOR هر دو ورودی را محاسبه می‌کنیم و در inequal قرار می‌دهیم. سپس بیت‌های آن را or می‌کنیم. به این صورت سه عدد یک بیتی متناظر با هر ترکیب دوتایی در e بدست خواهد آمد. هر یک از این اعداد اگر برابر ۱ باشد به این معنی است که آن دو عدد حداقل در یک بیت با یکدیگر اختلاف دارند و اگر برابر صفر باشد به معنی این است که آن دو عدد کاملاً یکسان هستند. بنابراین nor نقیض این سه بیت مقدار error را مشخص می‌کند. (ne نقیض e است و همچنین  $e[i]$  نمایان گر این است که آیا عدد  $i$ ام و  $i+1$ ام نابرابر هستند یا خیر)

حال با استفاده از tri state buffer می‌توانیم خروجی out را مشخص کنیم. ابتدا باید مشخص کنیم که کدام عدد با خروجی متصل شود. اگر عدد اول با عدد دوم برابر باشد، عدد  $i$ ام با  $i+1$ ام برابر باشد، عدد  $i$ ام را به خروجی وصل می‌کنیم.