(پروژه آزمایشگاه مدار منطقی)

نحوه اجرا کد:

ابتدا یک ماژول به نام main طراحی میکنیم که شامل ۷ ورودی و یک خروجی که همان IEN هست میشود همچنین یک ورودی clk هم داریم .

module main\_module(

input wire clk,

input wire x1,

input wire x2,

input wire x3,

input wire x4,

input wire x5,

input wire x6,

input wire x7,

output wire IEN

);

سپس ۳ وایر یا همان سیم رابط در کد تعریف کرده ایم زیرا در این گذر پاسخ نهایی برخی از گیت های منطقی به گیت دیگر باید وارد و محسابات لازم صورت بگیرد برای همین نیاز داریم که در این کد ۳وایر برای کد تعریف کنیم.

wire p1, p2, p3,p4;

wire xor\_result;

wire jk\_Q;

به ترتیب به با استفاده از دستور assign وایر های تعریف شده را مقدار دهی کرده و طبق مدار خواسته شده گیت های منطقی انرا تعیین میکنیم.

assign p1 = x1 & x2 & x3;

assign p2 = p1 & x4;

assign p3 = p1 & x5;

assign p4 = x6 & x7;

assign xor\_result = p3 ^ p4;

بعد از مقدار دهی اولیه نیاز به یک ماٰژول به نام ماژول j-k flip\_flop داریم که برای طراحی این ماژول نیاز به ۳وایر j,k,clk و یک reg به نام Q داریم زیرا مقدار ان قرار است در always مقدار دهی شود که در نهایت به IEN متصل میشد.

module j\_k\_flip\_flop(

input wire J,

input wire K,

input wire CK,

output reg Q,

);

با استفاده از دستور alway و لبه بالا رونده کلارک شروع به مقدار دهی کردن Q میکنیم .

J=1 & k=0 🡺 q=1

J=0 & k=1 🡺 q=0

J=1 & k=1 🡺 q= ~q

always @(posedge CK) begin

if (J & ~K)

Q <= 1'b1;

else if (~J & K)

Q <= 1'b0;

else if (J & K)

Q <= ~Q;

در نهایت با استفاده از ماژول jk ومقدار دهی کردن اولیه ان در نهایت پاسخ نهایی را به IEN متصل میکنیم.

j\_k\_flip\_flop jk\_ff(

.J(p2),

.K(xor\_result),

.CK(clk),

.Q(jk\_Q),

);

assign IEN = jk\_Q;