**Food Dispenser**

ELEC 402: Synthesized Verilog Project

Xingwei Su

72979917

**Report from RTL Compiler**

Food Dispenser FSM used 157 Cells and has timing slack of 31ps which meet both cells requirement and the timing requirement.

文本, 信件

描述已自动生成

表格

描述已自动生成

**Visual Waveforms Comparison**

To match the timing slack, the main clock frequency from 50Mhz to 10Mhz. Thus, state will change much slower this time. Figures below demonstrate that the waveform before and after mapping perform the same.

图形用户界面

中度可信度描述已自动生成

Figure 1: Reset & Initialized Test (Before Mapped)

表格

描述已自动生成

Figure 2: Reset & Initialized Test (After Mapped)

图形用户界面, 图示

中度可信度描述已自动生成

Figure 3: Refill Test (Before Mapped)

表格

低可信度描述已自动生成

Figure 4: Refill Test (After Mapped)

图示

描述已自动生成

Figure 5: Playfunction Test (Before Mapped)

图片包含 图形用户界面

描述已自动生成

Figure 6: Playfunction Test (After Mapped)

电脑萤幕画面

中度可信度描述已自动生成

Figure 7: Daily Limit Test (Before Mapped)

绿色的钟表

中度可信度描述已自动生成

Figrue 8: Daily Limit Test (After Mapped)

图表

描述已自动生成

Figure 9: New Day Reset (Before Mapped)

电脑萤幕画面

中度可信度描述已自动生成

Figure 10: New Day Reset (After Mapped)

图形用户界面

描述已自动生成

Figure 11: Autofeeding (Before Mapped)

电脑萤幕画面

中度可信度描述已自动生成

Figure 12: Autofeeding (After Mapped)

电脑萤幕画面

描述已自动生成

Figure 13: Warning (Before Mapped)

图形用户界面

中度可信度描述已自动生成

Figure 14: Warning (After Mapped)

Latency Demonstrate:

图片包含 图形用户界面

描述已自动生成

Figure 15: Delay of FSM

When mapping the code to Verilog, the latency simulation is performed. In this case FSM Food Dispenser have 0.1 ns latency from the clock signal. However, the state transition performs the same with the previous system Verilog code, thus the mapped Verilog is satisfied the timing requirement.

**Mapped Verilog generated by RTL Compiler**

From the previous system Verilog code, I found there is a few latches and thus delete one of the always block and add reset feature to the other. No functional code being added. If needed, please refer to .sv file inside .zip.

// Generated by Cadence Encounter(R) RTL Compiler RC14.13 - v14.10-s027\_1

// Verification Directory fv/food\_dispenser\_fsm

module food\_dispenser\_fsm(reset, clk, timesup, food\_weight,

     set\_food\_weight, refill\_detector, cap\_detector,

     play\_function\_pedal, initialize\_flag, newday, food\_gate, warning,

     play\_function\_flag, play\_function\_fail\_flag);

  input reset, clk, timesup, refill\_detector, cap\_detector,

       play\_function\_pedal, initialize\_flag, newday;

  input [6:0] food\_weight, set\_food\_weight;

  output food\_gate, warning, play\_function\_flag,

       play\_function\_fail\_flag;

  wire reset, clk, timesup, refill\_detector, cap\_detector,

       play\_function\_pedal, initialize\_flag, newday;

  wire [6:0] food\_weight, set\_food\_weight;

  wire food\_gate, warning, play\_function\_flag, play\_function\_fail\_flag;

  wire [3:0] state;

  wire [3:0] play\_function\_counter;

  wire n\_0, n\_1, n\_2, n\_3, n\_4, n\_5, n\_6, n\_7;

  wire n\_8, n\_9, n\_10, n\_11, n\_12, n\_13, n\_14, n\_15;

  wire n\_16, n\_17, n\_18, n\_19, n\_20, n\_21, n\_22, n\_23;

  wire n\_24, n\_25, n\_26, n\_27, n\_28, n\_29, n\_30, n\_31;

  wire n\_32, n\_33, n\_34, n\_35, n\_36, n\_37, n\_38, n\_39;

  wire n\_40, n\_41, n\_42, n\_43, n\_44, n\_45, n\_46, n\_47;

  wire n\_48, n\_49, n\_50, n\_51, n\_52, n\_53, n\_54, n\_55;

  wire n\_56, n\_57, n\_58, n\_59, n\_60, n\_61, n\_62, n\_63;

  wire n\_64, n\_65, n\_66, n\_67, n\_68, n\_69, n\_70, n\_71;

  wire n\_72, n\_73, n\_74, n\_75, n\_76, n\_77, n\_78, n\_79;

  wire n\_80, n\_81, n\_82, n\_83, n\_84, n\_85, n\_86, n\_87;

  wire n\_88, n\_89, n\_90, n\_91, n\_92, n\_93, n\_94, n\_95;

  wire n\_96, n\_97, n\_98, n\_99, n\_100, n\_101, n\_102, n\_103;

  wire n\_104, n\_105, n\_106, n\_107, n\_108, n\_109, n\_110, n\_111;

  wire n\_112, n\_113, n\_114, n\_116, n\_117, n\_118, n\_119, n\_120;

  wire n\_121, n\_122, n\_123, n\_124, n\_125, n\_126, n\_127, n\_128;

  wire n\_129, n\_130, n\_131, n\_132, n\_133, n\_134, n\_135, n\_136;

  wire n\_137, n\_138, n\_139, n\_140, n\_141, n\_142, n\_143, n\_144;

  wire n\_145;

  DFFSNQ\_X1 warning\_reg(.SN (1'b1), .CLK (clk), .D (n\_145), .Q

       (warning));

  NAND2\_X1 g3725(.A1 (n\_144), .A2 (n\_83), .ZN (n\_145));

  NAND4\_X1 g3726(.A1 (n\_143), .A2 (n\_73), .A3 (n\_92), .A4 (n\_53), .ZN

       (n\_144));

  AOI22\_X1 g3727(.A1 (n\_142), .A2 (n\_90), .B1 (n\_11), .B2 (n\_3), .ZN

       (n\_143));

  AND2\_X1 g3728(.A1 (n\_141), .A2 (n\_121), .Z (n\_142));

  AOI22\_X1 g3729(.A1 (n\_140), .A2 (n\_123), .B1 (n\_139), .B2

       (food\_weight[2]), .ZN (n\_141));

  AOI22\_X1 g3730(.A1 (n\_137), .A2 (n\_44), .B1 (n\_127), .B2 (n\_138), .ZN

       (n\_140));

  AOI21\_X1 g3731(.A1 (n\_122), .A2 (n\_138), .B (n\_137), .ZN (n\_139));

  OAI21\_X1 g3732(.A1 (n\_133), .A2 (food\_weight[0]), .B (n\_136), .ZN

       (n\_137));

  AOI21\_X1 g3733(.A1 (n\_134), .A2 (n\_35), .B (n\_135), .ZN (n\_136));

  AOI21\_X1 g3735(.A1 (n\_132), .A2 (food\_weight[0]), .B

       (food\_weight[1]), .ZN (n\_135));

  XNOR2\_X1 g3742(.A1 (n\_128), .A2 (set\_food\_weight[3]), .ZN (n\_134));

  DFFRNQ\_X1 \state\_reg[0] (.RN (n\_130), .CLK (clk), .D (n\_129), .Q

       (state[0]));

  NAND2\_X1 g3739(.A1 (n\_131), .A2 (set\_food\_weight[1]), .ZN (n\_133));

  AOI21\_X1 g3740(.A1 (set\_food\_weight[1]), .A2 (set\_food\_weight[0]), .B

       (n\_131), .ZN (n\_132));

  DFFRNQ\_X1 \state\_reg[1] (.RN (n\_130), .CLK (clk), .D (n\_126), .Q

       (state[1]));

  OR4\_X1 g3741(.A1 (n\_119), .A2 (n\_75), .A3 (n\_86), .A4 (n\_51), .Z

       (n\_129));

  XOR2\_X1 g3745(.A1 (n\_124), .A2 (set\_food\_weight[2]), .Z (n\_131));

  NAND2\_X1 g3747(.A1 (n\_125), .A2 (n\_127), .ZN (n\_128));

  NOR4\_X1 g3737(.A1 (n\_88), .A2 (n\_118), .A3 (n\_108), .A4 (n\_93), .ZN

       (n\_126));

  AOI22\_X1 g3756(.A1 (n\_120), .A2 (n\_19), .B1 (n\_116), .B2

       (set\_food\_weight[2]), .ZN (n\_125));

  AOI21\_X1 g3752(.A1 (n\_123), .A2 (n\_122), .B (n\_114), .ZN (n\_124));

  AOI22\_X1 g3754(.A1 (n\_120), .A2 (food\_weight[3]), .B1 (n\_99), .B2

       (food\_weight[4]), .ZN (n\_121));

  NAND2\_X1 g3743(.A1 (n\_117), .A2 (n\_107), .ZN (n\_119));

  INV\_X1 g3748(.I (n\_117), .ZN (n\_118));

  NAND2\_X1 g3749(.A1 (n\_87), .A2 (n\_113), .ZN (n\_117));

  INV\_X1 g3765(.I (n\_123), .ZN (n\_116));

  DFFSNQ\_X1 \play\_function\_counter\_reg[3] (.SN (1'b1), .CLK (clk), .D

       (n\_110), .Q (play\_function\_counter[3]));

  INV\_X1 g3763(.I (n\_127), .ZN (n\_114));

  OAI21\_X1 g3766(.A1 (set\_food\_weight[3]), .A2 (set\_food\_weight[2]), .B

       (n\_109), .ZN (n\_123));

  INV\_X1 g3767(.I (n\_122), .ZN (n\_120));

  AOI21\_X1 g3751(.A1 (n\_97), .A2 (n\_7), .B (play\_function\_flag), .ZN

       (n\_113));

  NAND3\_X1 g3764(.A1 (n\_111), .A2 (n\_112), .A3 (n\_4), .ZN (n\_127));

  OAI21\_X1 g3768(.A1 (n\_112), .A2 (n\_15), .B (n\_111), .ZN (n\_122));

  DFFSNQ\_X1 \play\_function\_counter\_reg[2] (.SN (1'b1), .CLK (clk), .D

       (n\_102), .Q (play\_function\_counter[2]));

  DFFSNQ\_X1 \play\_function\_counter\_reg[0] (.SN (1'b1), .CLK (clk), .D

       (n\_96), .Q (play\_function\_counter[0]));

  DFFSNQ\_X1 \play\_function\_counter\_reg[1] (.SN (1'b1), .CLK (clk), .D

       (n\_104), .Q (play\_function\_counter[1]));

  DFFRNQ\_X1 \state\_reg[2] (.RN (n\_130), .CLK (clk), .D (n\_94), .Q

       (state[2]));

  DFFSNQ\_X1 play\_function\_flag\_reg(.SN (1'b1), .CLK (clk), .D (n\_95),

       .Q (play\_function\_flag));

  DFFSNQ\_X1 food\_gate\_reg(.SN (1'b1), .CLK (clk), .D (n\_100), .Q

       (food\_gate));

  OAI21\_X1 g3744(.A1 (n\_85), .A2 (n\_103), .B (n\_77), .ZN (n\_110));

  DFFRNQ\_X1 \state\_reg[3] (.RN (n\_130), .CLK (clk), .D (n\_106), .Q

       (state[3]));

  INV\_X1 g3775(.I (n\_111), .ZN (n\_109));

  NAND2\_X1 g3750(.A1 (n\_107), .A2 (n\_105), .ZN (n\_108));

  AOI22\_X1 g3776(.A1 (n\_82), .A2 (set\_food\_weight[4]), .B1 (n\_46), .B2

       (n\_98), .ZN (n\_111));

  NAND4\_X1 g3777(.A1 (n\_68), .A2 (n\_61), .A3 (n\_105), .A4 (n\_65), .ZN

       (n\_106));

  OAI22\_X1 g3778(.A1 (n\_37), .A2 (n\_103), .B1 (n\_24), .B2 (n\_101), .ZN

       (n\_104));

  OAI22\_X1 g3753(.A1 (n\_79), .A2 (n\_103), .B1 (n\_78), .B2 (n\_101), .ZN

       (n\_102));

  INV\_X1 g3786(.I (n\_89), .ZN (n\_100));

  AOI22\_X1 g3788(.A1 (n\_99), .A2 (set\_food\_weight[4]), .B1 (n\_91), .B2

       (n\_98), .ZN (n\_112));

  DFFSNQ\_X1 play\_function\_fail\_flag\_reg(.SN (1'b1), .CLK (clk), .D

       (n\_81), .Q (play\_function\_fail\_flag));

  NOR4\_X1 g3769(.A1 (n\_66), .A2 (n\_28), .A3 (n\_33), .A4 (n\_34), .ZN

       (n\_97));

  NAND2\_X1 g3771(.A1 (n\_84), .A2 (n\_63), .ZN (n\_96));

  OAI22\_X1 g3772(.A1 (n\_41), .A2 (n\_80), .B1 (n\_1), .B2 (n\_69), .ZN

       (n\_95));

  NAND3\_X1 g3774(.A1 (n\_71), .A2 (n\_74), .A3 (n\_59), .ZN (n\_94));

  AOI21\_X1 g3783(.A1 (n\_39), .A2 (n\_47), .B (n\_60), .ZN (n\_93));

  NAND3\_X1 g3785(.A1 (n\_91), .A2 (n\_90), .A3 (n\_17), .ZN (n\_92));

  AOI22\_X1 g3787(.A1 (food\_gate), .A2 (n\_49), .B1 (n\_88), .B2 (n\_32),

       .ZN (n\_89));

  NAND3\_X1 g3757(.A1 (n\_87), .A2 (play\_function\_flag), .A3 (n\_45), .ZN

       (n\_107));

  OAI22\_X1 g3800(.A1 (n\_43), .A2 (n\_72), .B1 (n\_30), .B2 (n\_70), .ZN

       (n\_86));

  XOR2\_X1 g3755(.A1 (n\_58), .A2 (play\_function\_counter[3]), .Z (n\_85));

  OAI21\_X1 g3781(.A1 (n\_8), .A2 (n\_76), .B (play\_function\_counter[0]),

       .ZN (n\_84));

  NAND2\_X1 g3782(.A1 (warning), .A2 (n\_56), .ZN (n\_83));

  NOR2\_X1 g3790(.A1 (n\_99), .A2 (set\_food\_weight[5]), .ZN (n\_82));

  OAI21\_X1 g3797(.A1 (n\_55), .A2 (n\_80), .B (n\_52), .ZN (n\_81));

  XOR2\_X1 g3779(.A1 (n\_57), .A2 (n\_78), .Z (n\_79));

  NAND2\_X1 g3789(.A1 (play\_function\_counter[3]), .A2 (n\_76), .ZN

       (n\_77));

  NOR2\_X1 g3791(.A1 (state[3]), .A2 (n\_40), .ZN (n\_75));

  AOI21\_X1 g3792(.A1 (n\_73), .A2 (n\_23), .B (n\_87), .ZN (n\_74));

  NAND3\_X1 g3793(.A1 (n\_64), .A2 (n\_72), .A3 (play\_function\_pedal), .ZN

       (n\_105));

  NAND3\_X1 g3794(.A1 (n\_67), .A2 (n\_70), .A3 (state[3]), .ZN (n\_71));

  AOI21\_X1 g3795(.A1 (n\_54), .A2 (state[3]), .B (n\_76), .ZN (n\_69));

  NAND2\_X1 g3796(.A1 (n\_42), .A2 (n\_67), .ZN (n\_68));

  NAND4\_X1 g3799(.A1 (n\_18), .A2 (n\_29), .A3 (n\_16), .A4 (n\_20), .ZN

       (n\_66));

  NAND2\_X1 g3801(.A1 (n\_50), .A2 (n\_62), .ZN (n\_103));

  NAND2\_X1 g3803(.A1 (n\_64), .A2 (newday), .ZN (n\_65));

  NAND2\_X1 g3809(.A1 (n\_36), .A2 (n\_62), .ZN (n\_63));

  NAND3\_X1 g3811(.A1 (n\_60), .A2 (n\_26), .A3 (play\_function\_flag), .ZN

       (n\_61));

  NAND3\_X1 g3813(.A1 (n\_64), .A2 (n\_2), .A3 (refill\_detector), .ZN

       (n\_59));

  INV\_X1 g3818(.I (n\_99), .ZN (n\_91));

  NAND2\_X1 g3784(.A1 (n\_57), .A2 (play\_function\_counter[2]), .ZN

       (n\_58));

  NAND4\_X1 g3798(.A1 (n\_55), .A2 (n\_54), .A3 (n\_48), .A4 (n\_53), .ZN

       (n\_56));

  NAND2\_X1 g3806(.A1 (play\_function\_fail\_flag), .A2 (n\_21), .ZN (n\_52));

  INV\_X1 g3807(.I (n\_76), .ZN (n\_101));

  NOR3\_X1 g3810(.A1 (n\_50), .A2 (n\_31), .A3 (n\_22), .ZN (n\_51));

  NAND4\_X1 g3815(.A1 (n\_48), .A2 (n\_9), .A3 (n\_6), .A4 (n\_130), .ZN

       (n\_49));

  OAI21\_X1 g3816(.A1 (n\_12), .A2 (n\_38), .B (n\_48), .ZN (n\_47));

  AOI22\_X1 g3819(.A1 (n\_46), .A2 (set\_food\_weight[4]), .B1 (n\_27), .B2

       (set\_food\_weight[6]), .ZN (n\_99));

  NAND4\_X1 g3773(.A1 (n\_5), .A2 (n\_138), .A3 (n\_44), .A4

       (food\_weight[0]), .ZN (n\_45));

  INV\_X1 g3833(.I (n\_64), .ZN (n\_43));

  NOR2\_X1 g3802(.A1 (n\_70), .A2 (n\_48), .ZN (n\_42));

  NAND2\_X1 g3804(.A1 (n\_70), .A2 (n\_88), .ZN (n\_41));

  AOI21\_X1 g3805(.A1 (n\_10), .A2 (food\_weight[5]), .B (food\_weight[6]),

       .ZN (n\_90));

  OAI21\_X1 g3808(.A1 (n\_88), .A2 (state[3]), .B (n\_53), .ZN (n\_76));

  AOI21\_X1 g3812(.A1 (n\_39), .A2 (n\_38), .B (n\_88), .ZN (n\_40));

  XOR2\_X1 g3817(.A1 (n\_25), .A2 (play\_function\_counter[1]), .Z (n\_37));

  NOR2\_X1 g3820(.A1 (n\_54), .A2 (play\_function\_counter[0]), .ZN (n\_36));

  INV\_X1 g3825(.I (n\_80), .ZN (n\_62));

  NAND2\_X1 g3827(.A1 (n\_34), .A2 (n\_33), .ZN (n\_35));

  NOR2\_X1 g3828(.A1 (n\_31), .A2 (reset), .ZN (n\_32));

  NOR2\_X1 g3829(.A1 (n\_31), .A2 (state[1]), .ZN (n\_87));

  INV\_X1 g3830(.I (n\_30), .ZN (n\_67));

  NOR3\_X1 g3834(.A1 (n\_54), .A2 (state[2]), .A3 (state[3]), .ZN (n\_64));

  AOI22\_X1 g3840(.A1 (n\_14), .A2 (set\_food\_weight[1]), .B1 (n\_44), .B2

       (set\_food\_weight[2]), .ZN (n\_29));

  XNOR2\_X1 g3841(.A1 (n\_27), .A2 (food\_weight[5]), .ZN (n\_28));

  INV\_X1 g3860(.I (n\_31), .ZN (n\_60));

  INV\_X1 g3856(.I (n\_54), .ZN (n\_26));

  NOR2\_X1 g3821(.A1 (n\_25), .A2 (n\_24), .ZN (n\_57));

  NOR2\_X1 g3822(.A1 (n\_50), .A2 (state[3]), .ZN (n\_73));

  NAND2\_X1 g3823(.A1 (state[2]), .A2 (n\_22), .ZN (n\_23));

  NAND2\_X1 g3824(.A1 (n\_88), .A2 (n\_53), .ZN (n\_21));

  NAND2\_X1 g3826(.A1 (state[3]), .A2 (n\_53), .ZN (n\_80));

  NAND2\_X1 g3831(.A1 (n\_88), .A2 (n\_39), .ZN (n\_30));

  AOI22\_X1 g3837(.A1 (n\_19), .A2 (food\_weight[2]), .B1 (n\_13), .B2

       (set\_food\_weight[0]), .ZN (n\_20));

  AOI22\_X1 g3838(.A1 (n\_138), .A2 (set\_food\_weight[3]), .B1 (n\_17), .B2

       (set\_food\_weight[4]), .ZN (n\_18));

  AOI22\_X1 g3839(.A1 (n\_15), .A2 (food\_weight[3]), .B1 (n\_98), .B2

       (food\_weight[4]), .ZN (n\_16));

  NOR2\_X1 g3858(.A1 (n\_27), .A2 (set\_food\_weight[6]), .ZN (n\_46));

  NOR2\_X1 g3843(.A1 (n\_14), .A2 (set\_food\_weight[1]), .ZN (n\_34));

  NOR2\_X1 g3842(.A1 (n\_13), .A2 (set\_food\_weight[0]), .ZN (n\_33));

  INV\_X1 g3846(.I (n\_50), .ZN (n\_12));

  INV\_X1 g3851(.I (n\_10), .ZN (n\_11));

  NAND2\_X1 g3855(.A1 (state[0]), .A2 (n\_0), .ZN (n\_55));

  NAND2\_X1 g3859(.A1 (n\_39), .A2 (state[1]), .ZN (n\_9));

  NOR2\_X1 g3862(.A1 (n\_48), .A2 (state[1]), .ZN (n\_8));

  XNOR2\_X1 g3835(.A1 (food\_weight[6]), .A2 (set\_food\_weight[6]), .ZN

       (n\_7));

  NAND2\_X1 g3857(.A1 (n\_6), .A2 (state[1]), .ZN (n\_54));

  NAND2\_X1 g3861(.A1 (state[2]), .A2 (n\_48), .ZN (n\_31));

  NOR4\_X1 g3814(.A1 (food\_weight[4]), .A2 (food\_weight[1]), .A3

       (food\_weight[6]), .A4 (food\_weight[5]), .ZN (n\_5));

  NOR3\_X1 g3832(.A1 (refill\_detector), .A2 (timesup), .A3 (newday), .ZN

       (n\_72));

  NAND4\_X1 g3836(.A1 (play\_function\_counter[1]), .A2

       (play\_function\_counter[2]), .A3 (play\_function\_counter[3]), .A4

       (play\_function\_counter[0]), .ZN (n\_70));

  NAND2\_X1 g3854(.A1 (set\_food\_weight[3]), .A2 (set\_food\_weight[2]),

       .ZN (n\_4));

  NOR2\_X1 g3844(.A1 (state[1]), .A2 (initialize\_flag), .ZN (n\_38));

  NOR2\_X1 g3849(.A1 (food\_weight[6]), .A2 (food\_weight[5]), .ZN (n\_3));

  NOR2\_X1 g3853(.A1 (state[2]), .A2 (reset), .ZN (n\_53));

  NOR2\_X1 g3848(.A1 (state[0]), .A2 (state[1]), .ZN (n\_88));

  NOR2\_X1 g3850(.A1 (refill\_detector), .A2 (cap\_detector), .ZN (n\_22));

  NAND2\_X1 g3852(.A1 (set\_food\_weight[6]), .A2 (set\_food\_weight[5]),

       .ZN (n\_10));

  NAND2\_X1 g3845(.A1 (state[1]), .A2 (play\_function\_counter[0]), .ZN

       (n\_25));

  NAND2\_X1 g3847(.A1 (state[0]), .A2 (state[1]), .ZN (n\_50));

  INV\_X1 g3877(.I (reset), .ZN (n\_130));

  INV\_X1 g3872(.I (food\_weight[1]), .ZN (n\_14));

  INV\_X1 g3871(.I (food\_weight[4]), .ZN (n\_17));

  INV\_X1 g3868(.I (play\_function\_counter[2]), .ZN (n\_78));

  INV\_X1 g3880(.I (food\_weight[2]), .ZN (n\_44));

  INV\_X1 g3874(.I (newday), .ZN (n\_2));

  INV\_X1 g3873(.I (food\_weight[3]), .ZN (n\_138));

  INV\_X1 g3863(.I (state[2]), .ZN (n\_39));

  INV\_X1 g3879(.I (set\_food\_weight[4]), .ZN (n\_98));

  INV\_X1 g3878(.I (set\_food\_weight[3]), .ZN (n\_15));

  INV\_X1 g3865(.I (state[3]), .ZN (n\_48));

  INV\_X1 g3869(.I (play\_function\_counter[1]), .ZN (n\_24));

  INV\_X1 g3867(.I (play\_function\_flag), .ZN (n\_1));

  INV\_X1 g3875(.I (set\_food\_weight[2]), .ZN (n\_19));

  INV\_X1 g3866(.I (state[0]), .ZN (n\_6));

  INV\_X1 g3870(.I (food\_weight[0]), .ZN (n\_13));

  INV\_X1 g3864(.I (state[1]), .ZN (n\_0));

  INV\_X1 g3876(.I (set\_food\_weight[5]), .ZN (n\_27));

endmodule