```verilog
module Plants(
    input Rungame,
    input Framerate,
    input INIT,
    input [11:0] INITPosition,
    input FirstPlant,
    input ThreePixFrameClk,
    input ResetPlant,
    input clk,
    output [11:0] Row,
    output [11:0] Col
    );
    wire dtc1, dtc2;

    wire Reset = (Col == 4079); //when to reset the tree to the other side
    wire [11:0] PlantPosition = INIT ? INITPosition : 12'b001011010101;
    countUD4L plantcount1 (.Up(1'b0), .Dw(ThreePixFrameClk & Rungame),
.LD(INIT | Reset | ResetPlant), .Reset(1'b0), .Q(PlantPosition[3:0]), .clk(clk),
.DTC(dtc1), .Qout(Col[3:0]));  //counters for the columns of the plants
    countUD4L plantcount2 (.Up(1'b0), .Dw(ThreePixFrameClk & Rungame & dtc1),
.LD(INIT | Reset | ResetPlant), .Reset(1'b0), .Q(PlantPosition[7:4]), .clk(clk),
.DTC(dtc2), .Qout(Col[7:4]));
    countUD4L plantcount3 (.Up(1'b0), .Dw(ThreePixFrameClk & Rungame & dtc1 & dtc2),
.LD(INIT | Reset | ResetPlant), .Reset(1'b0), .Q(PlantPosition[11:8]), .clk(clk),
        .Qout(Col[11:8]));   //the Q bits reset the plants to 680

    //All these numbers are WTR 232
    parameter pos1 = 156, pos2 = 160, pos3 = 164, pos4 = 168, pos5 = 172, pos6 =
176, pos7 = 180, pos8 = 184;
    parameter pos9 = 188, pos10 = 192, pos11 = 196, pos12 = 200, pos13 = 216, pos14
= 232, pos15 = 236;

    //This randomly choses a row to start off on
    wire [3:0]rand;
    wire getRandRow = (Col == 680);
    LFSR randNumGen (.clk(clk), .GetNum(getRandRow), .Q(rand));

    wire initHold;
    FDRE #(.INIT(1'b0) ) ff1 (.C(clk), .R(Reset), .CE(INIT|Reset), .D(INIT),
.Q(initHold)); //if we started with an initial position (only the first plant) it
should hold until the next value

    assign Row = FirstPlant & initHold ? pos8:
                        (rand == 0) ? pos1  :
                        (rand == 1) ? pos2  :
                        (rand == 2) ? pos3  :
```

```verilog
                     (rand == 3)  ? pos4  :
                     (rand == 4)  ? pos5  :
                     (rand == 5)  ? pos6  :
                     (rand == 6)  ? pos7  :
                     (rand == 7)  ? pos8  :
                     (rand == 8)  ? pos9  :
                     (rand == 9)  ? pos10 :
                     (rand == 10) ? pos11 :
                     (rand == 11) ? pos12 :
                     (rand == 12) ? pos13 :
                     (rand == 13) ? pos14 :
                     (rand == 14) ? pos15 : pos8;   //to further randomize

endmodule
```