

リスト2 反射動作のVerilog HDL 記述

```
//Wait for Start any MODE
ps2:
begin
rHansyaOn <= 8'b0000_0000;
roHansyaSet <= 0;

if(iExtStart) //上位レベルからのスタート信号
begin
case(iMODE)
2'b00: rs <= psR1;
//00 : 外部 (上位レベル) からのパラメータ読み出し命令
2'b01: rs <= psAW1;
//01 : 自動パラメータ設定 (ps0 で行ったのとほぼ同じ)
2'b10: rs <= psEW1;
//10 : 外部 (上位レベル) からのパラメータ書き込み命令
2'b11: rs <= ps2;
//11 : 反射動作待機、センサの入力に対して反射動作をするモード
default: rs <= ps2;
endcase
end

else if (iSensorSet)
//スタート信号が無いときは反射動作を行う
begin
rMODE[0] <= 2'b00; //反射動作パラメータの保存してある
rMODE[1] <= 2'b00; //BlockRAMを読み出しモードに設定
rMODE[2] <= 2'b00;
rs <= (iMODE == 2'b11) ? psH1: ps2;
end
else begin
rs <= ps2;
end
end

// MODE 11 (Hansya Mode)
psH1: //データ読み込み開始
begin

rShinritiAddr[0] <= rSensorData[0];
//センサデータをアドレスにする
rShinritiAddr[1] <= rSensorData[1];
rShinritiAddr[2] <= rSensorData[2];

rMODE[0] <= 2'b00; //00:Readモード
rMODE[1] <= 2'b00;
rMODE[2] <= 2'b00;
```

```
rwShinritiStart[2:0] <= 3'b111; //真理値表のスタート信号

rs <= psH2;
end

psH2: //データ取得
begin
rShinritiData_0 <= wOutputData_0; //真理値表から
rShinritiData_1 <= wOutputData_1; //反射データの取得
rShinritiData_2 <= wOutputData_2;

//反射データと反射判定パラメータの&をとり、真なら反射
rHansyaOn[0] <= |{wOutputData_0 &
{8{rHansyaHantei[0]}} ? 1: 0;
rHansyaOn[1] <= |{wOutputData_1 &
{8{rHansyaHantei[1]}} ? 1: 0;
rHansyaOn[2] <= |{wOutputData_2 &
{8{rHansyaHantei[2]}} ? 1: 0;

rwShinritiStart[2:0] <= 3'b000;

rs <= (rShinritiFin==8'b0000_0111) ? psH3: psH2;
end

psH3: //基準位置と比較
begin

rDaiSy0[0] <= {rShinritiAddr[0] > rKizyun[0]} ? 1: 0;
rDaiSy0[1] <= {rShinritiAddr[1] > rKizyun[1]} ? 1: 0;
rDaiSy0[2] <= {rShinritiAddr[2] > rKizyun[2]} ? 1: 0;

rs <= (rHansyaOn[0] || rHansyaOn[1] || rHansyaOn[2]) ?
psH4: ps2;
end

psH4: //加算か減算かの判定結果を rKagen に代入
begin

rKagen[0] <= {rDaiSy0[0] ^ rDir[0]};
rKagen[1] <= {rDaiSy0[1] ^ rDir[1]};
rKagen[2] <= {rDaiSy0[2] ^ rDir[2]};

rs <= psH5a;
end
```