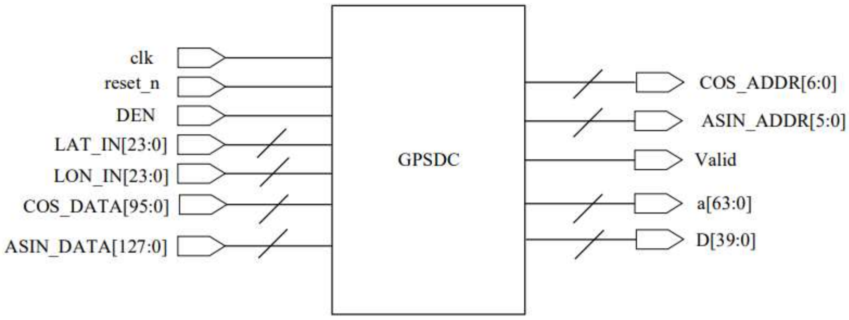
(3) Real-Time GPS Distance Calculator

甲、摘要

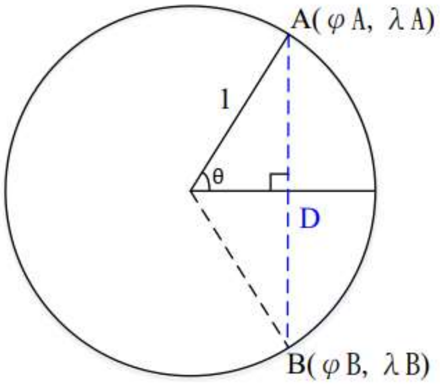
利用硬體涉及 GPS 可減少軟體分析所耗費的大量時間，可用於即時系統上面，算出上一筆輸入與目前輸入的距離並輸出。



圖一、系統方塊圖

乙、想法

利用半正矢定理(Haversine Formula)，並使用查表(三角函數)搭配線性內插來估算圓座標兩點間的距離



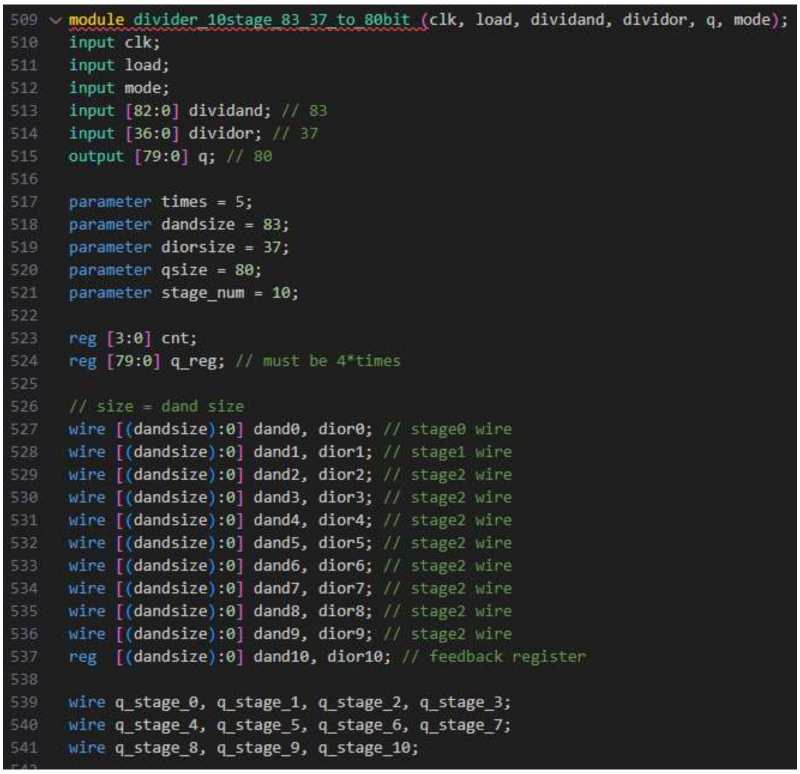
$$\mathrm{haversin}(\theta) = \sin^2(\frac{\theta}{2})$$

$$a = \sin^2\left(\frac{(\varphi B - \varphi A) * rad}{2}\right) + \cos\varphi A * \cos\varphi B * \sin^2\left(\frac{(\lambda B - \lambda A) * rad}{2}\right)$$

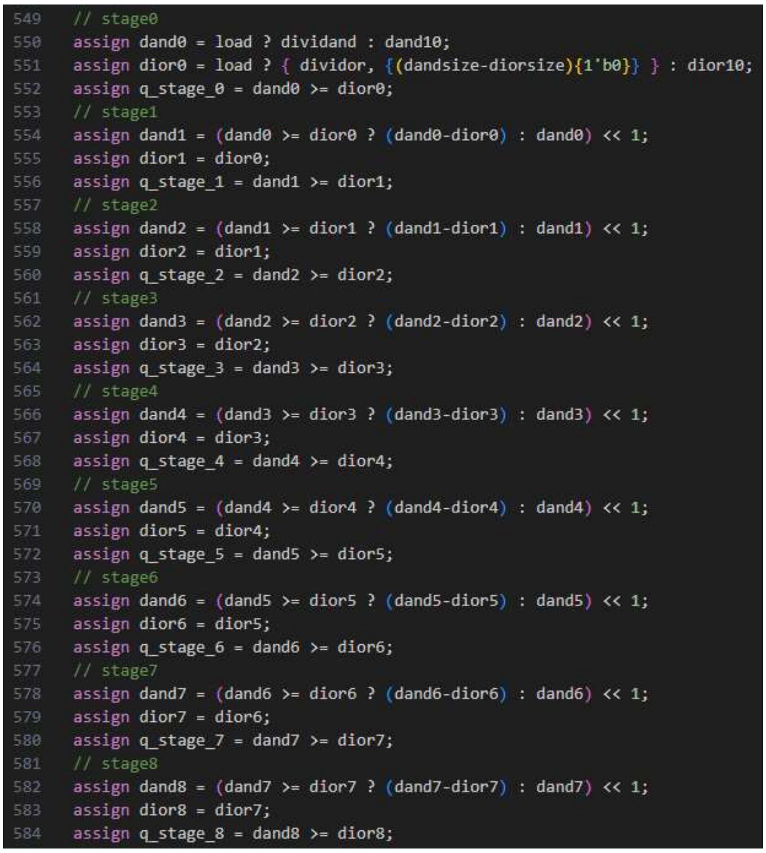
$$D=R*\sin^{-1}(\sqrt{a})$$

圖二、半正矢定理

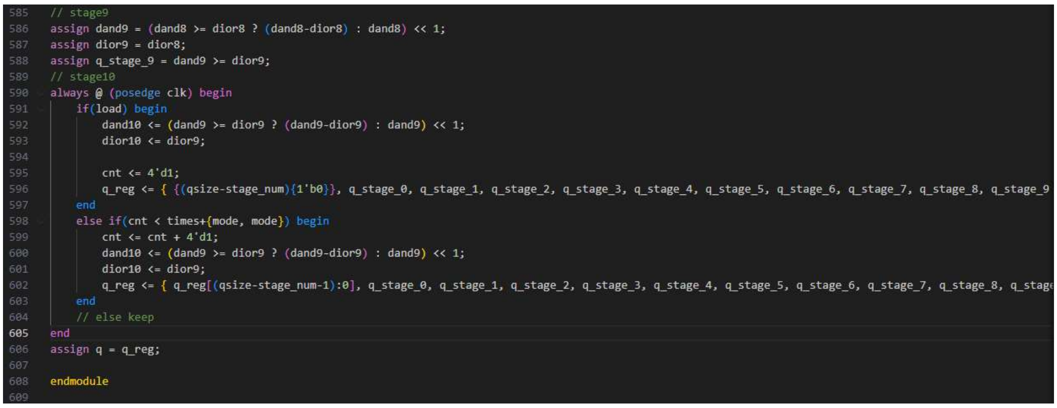
丙、結果



```
module divider_10stage_83_37_to_80bit (clk, load, dividand, dividor, q, mode);
input clk;
input load;
input mode;
input [82:0] dividand; // 83
input [36:0] dividor; // 37
output [79:0] q; // 80

parameter times = 5;
parameter dandsize = 83;
parameter diorsize = 37;
parameter qsize = 80;
parameter stage_num = 10;

reg [3:0] cnt;
reg [79:0] q_reg; // must be 4*times

// size = dand size
wire [(dandsize):0] dand0, dior0; // stage0 wire
wire [(dandsize):0] dand1, dior1; // stage1 wire
wire [(dandsize):0] dand2, dior2; // stage2 wire
wire [(dandsize):0] dand3, dior3; // stage2 wire
wire [(dandsize):0] dand4, dior4; // stage2 wire
wire [(dandsize):0] dand5, dior5; // stage2 wire
wire [(dandsize):0] dand6, dior6; // stage2 wire
wire [(dandsize):0] dand7, dior7; // stage2 wire
wire [(dandsize):0] dand8, dior8; // stage2 wire
wire [(dandsize):0] dand9, dior9; // stage2 wire
reg  [(dandsize):0] dand10, dior10; // feedback register

wire q_stage_0, q_stage_1, q_stage_2, q_stage_3;
wire q_stage_4, q_stage_5, q_stage_6, q_stage_7;
wire q_stage_8, q_stage_9, q_stage_10;
```

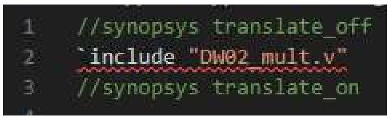


```
// stage0
assign dand0 = load ? dividand : dand10;
assign dior0 = load ? { dividor, {(dandsize-diorsize){1'b0}} } : dior10;
assign q_stage_0 = dand0 >= dior0;
// stage1
assign dand1 = (dand0 >= dior0 ? (dand0-dior0) : dand0) << 1;
assign dior1 = dior0;
assign q_stage_1 = dand1 >= dior1;
// stage2
assign dand2 = (dand1 >= dior1 ? (dand1-dior1) : dand1) << 1;
assign dior2 = dior1;
assign q_stage_2 = dand2 >= dior2;
// stage3
assign dand3 = (dand2 >= dior2 ? (dand2-dior2) : dand2) << 1;
assign dior3 = dior2;
assign q_stage_3 = dand3 >= dior3;
// stage4
assign dand4 = (dand3 >= dior3 ? (dand3-dior3) : dand3) << 1;
assign dior4 = dior3;
assign q_stage_4 = dand4 >= dior4;
// stage5
assign dand5 = (dand4 >= dior4 ? (dand4-dior4) : dand4) << 1;
assign dior5 = dior4;
assign q_stage_5 = dand5 >= dior5;
// stage6
assign dand6 = (dand5 >= dior5 ? (dand5-dior5) : dand5) << 1;
assign dior6 = dior5;
assign q_stage_6 = dand6 >= dior6;
// stage7
assign dand7 = (dand6 >= dior6 ? (dand6-dior6) : dand6) << 1;
assign dior7 = dior6;
assign q_stage_7 = dand7 >= dior7;
// stage8
assign dand8 = (dand7 >= dior7 ? (dand7-dior7) : dand7) << 1;
assign dior8 = dior7;
assign q_stage_8 = dand8 >= dior8;
```

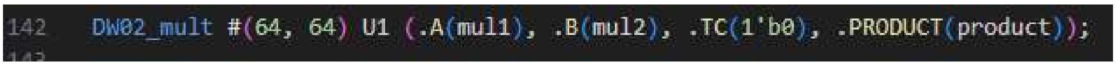


```verilog
// stage9
assign dand9 = (dand8 >= dior8 ? (dand8-dior8) : dand8) << 1;
assign dior9 = dior8;
assign q_stage_9 = dand9 >= dior9;
// stage10
always @ (posedge clk) begin
    if(load) begin
        dand10 <= (dand9 >= dior9 ? (dand9-dior9) : dand9) << 1;
        dior10 <= dior9;

        cnt <= 4'd1;
        q_reg <= { {(qsize-stage_num){1'b0}}, q_stage_0, q_stage_1, q_stage_2, q_stage_3, q_stage_4, q_stage_5, q_stage_6, q_stage_7, q_stage_8, q_stage_9
    end
    else if(cnt < times+{mode, mode}) begin
        cnt <= cnt + 4'd1;
        dand10 <= (dand9 >= dior9 ? (dand9-dior9) : dand9) << 1;
        dior10 <= dior9;
        q_reg <= { q_reg[(qsize-stage_num-1):0], q_stage_0, q_stage_1, q_stage_2, q_stage_3, q_stage_4, q_stage_5, q_stage_6, q_stage_7, q_stage_8, q_stage
    end
    // else keep
end
assign q = q_reg;

endmodule
```

圖三四五、使用客製化多級除法器

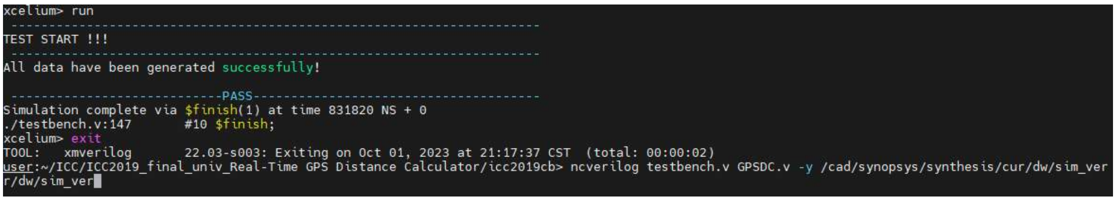


```verilog
//synopsys translate_off
`include "DW02_mult.v"
//synopsys translate_on
```



```verilog
DW02_mult #(64, 64) U1 (.A(mul1), .B(mul2), .TC(1'b0), .PRODUCT(product));
```

圖六七、使用 synopsys DW IP



```
xcelium> run
 ----------------------------------------------------------------------
TEST START !!!
 ----------------------------------------------------------------------
All data have been generated successfully!

 ---------------------------PASS--------------------------------------
Simulation complete via $finish(1) at time 831820 NS + 0
./testbench.v:147        #10 $finish;
xcelium> exit
TOOL:    xmverilog       22.03-s003: Exiting on Oct 01, 2023 at 21:17:37 CST  (total: 00:00:02)
user:~/ICC/ICC2019_final_univ_Real-Time GPS Distance Calculator/icc2019cb> ncverilog testbench.v GPSDC.v -y /cad/synopsys/synthesis/cur/dw/sim_ver
r/dw/sim_ver
```

圖八、驗證結果正確且達 Rank A