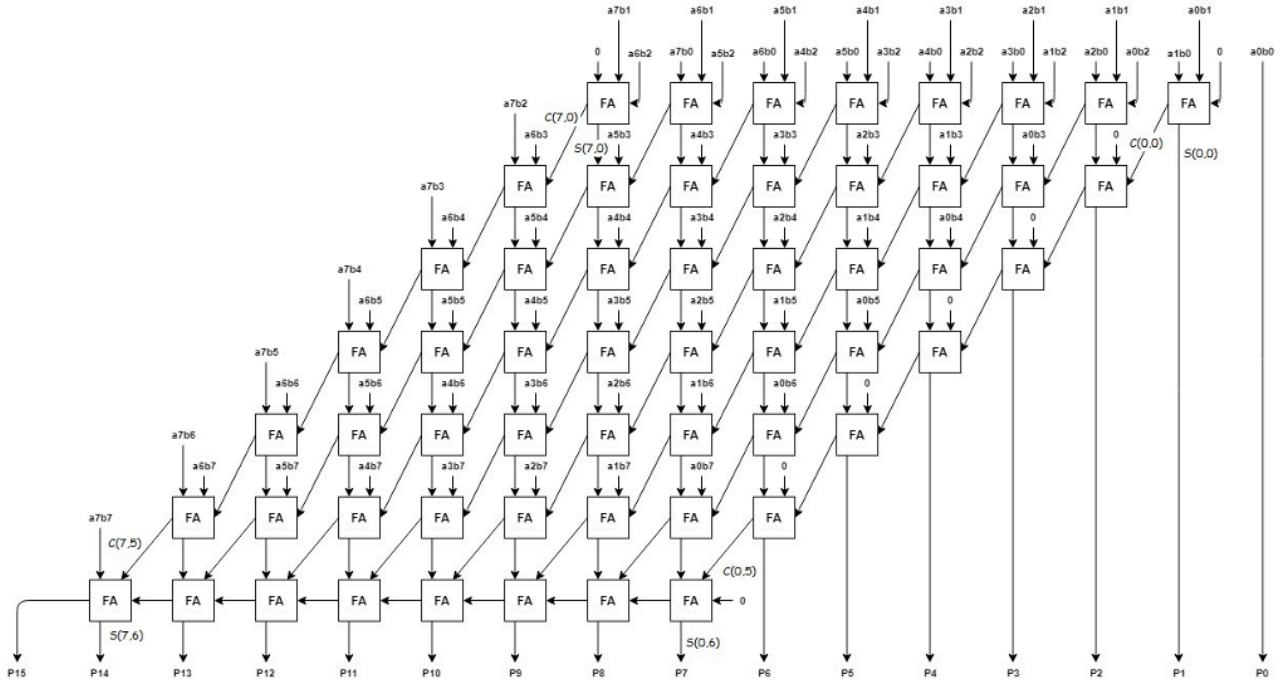
8-BIT CARRY-SAVE MULTIPLIER

NAME	SRN
ANSHUL RS	PES1UG23CS084
ANIRUDH SUDHIR	PES1UG23CS917
ATHREYA SHASHIDHARA	PES1UG23CS120
A.CHARAN KUMAR REDDY	PES1UG23CS101

DESIGN AND IMPLEMENT AN 8-BIT CARRY SAVE MULTIPLIER

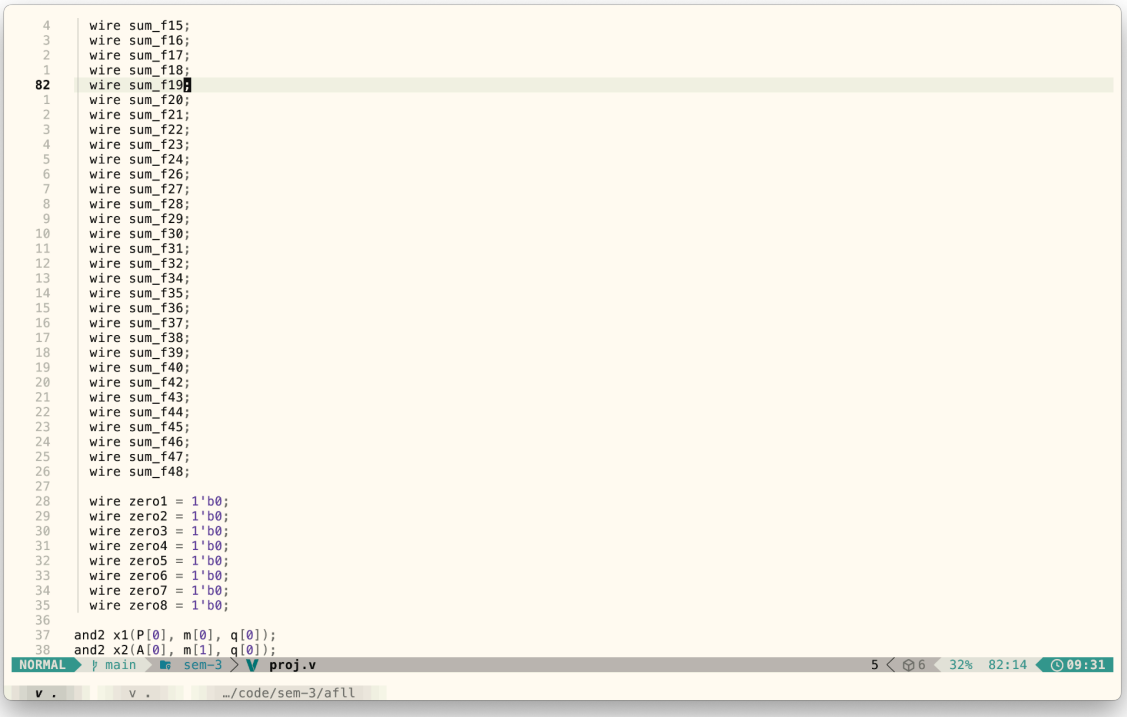
Circuit



proj.v

```
1  module carry_save_mult(input [7:0] m , input [7:0] q, output [15:0] P);
2
3    wire [6:0] A; //q0 series product
4    wire [7:0] B; //q1 series product
5    wire [7:0] C; //q2 series product
6    wire [7:0] D; //q3 series product
7    wire [7:0] E; //q4 series product
8    wire [7:0] F; //q5 series product
9    wire [7:0] G; //q6 series product
10   wire [7:0] H; //q7 series product
11
12   wire c_outf1;
13   wire c_outf2;
14   wire c_outf3;
15   wire c_outf4;
16   wire c_outf5;
17   wire c_outf6;
18   wire c_outf7;
19   wire c_outf8;
20   wire c_outf9;
21   wire c_outf10;
22   wire c_outf11;
23   wire c_outf12;
24   wire c_outf13;
25   wire c_outf14;
26   wire c_outf15;
27   wire c_outf16;
28   wire c_outf17;
29   wire c_outf18;
30   wire c_outf19;
31   wire c_outf20;
32   wire c_outf21;
33   wire c_outf22;
34   wire c_outf23;
35   wire c_outf24;
36   wire c_outf25;
37   wire c_outf26;
38   wire c_outf27;
39   wire c_outf28;
40   wire c_outf29;
41   wire c_outf30;
42   wire c_outf31;
43   wire c_outf32;
     wire c_outf33;
NORMAL  main  sem-3 > proj.v                                  { < 6 < Top  1:1  09:20
```


wire c_outf28;
wire c_outf29;
wire c_outf30;
wire c_outf31;
wire c_outf32;
wire c_outf33;
wire c_outf34;
wire c_outf35;
wire c_outf36;
wire c_outf37;
wire c_outf38;
wire c_outf39;
wire c_outf40;
wire c_outf42;
wire c_outf43;
wire c_outf44;
wire c_outf45;
wire c_outf46;
wire c_outf47;
wire c_outf48;
wire c_outf49;
wire c_outf50;
wire c_outf51;
wire c_outf52;
wire c_outf53;
wire c_outf54;
wire c_outf55;
wire sum_f2;
wire sum_f3;
wire sum_f4;
wire sum_f5;
wire sum_f6;
wire sum_f7;
wire sum_f8;
wire sum_f10;
wire sum_f11;
wire sum_f12;
wire sum_f13;
wire sum_f14;
wire sum_f15;
wire sum_f16;
wire sum_f17;
wire sum_f18;
NORMAL main sem-3 proj.v
32x17 6 16% 43:16 09:30
v . v . …/code/sem-3/afll


wire sum_f15;
wire sum_f16;
wire sum_f17;
wire sum_f18;
wire sum_f19;
wire sum_f20;
wire sum_f21;
wire sum_f22;
wire sum_f23;
wire sum_f24;
wire sum_f26;
wire sum_f27;
wire sum_f28;
wire sum_f29;
wire sum_f30;
wire sum_f31;
wire sum_f32;
wire sum_f34;
wire sum_f35;
wire sum_f36;
wire sum_f37;
wire sum_f38;
wire sum_f39;
wire sum_f40;
wire sum_f42;
wire sum_f43;
wire sum_f44;
wire sum_f45;
wire sum_f46;
wire sum_f47;
wire sum_f48;
wire zero1 = 1'b0;
wire zero2 = 1'b0;
wire zero3 = 1'b0;
wire zero4 = 1'b0;
wire zero5 = 1'b0;
wire zero6 = 1'b0;
wire zero7 = 1'b0;
wire zero8 = 1'b0;
and2 x1(P[0], m[0], q[0]);
and2 x2(A[0], m[1], q[0]);
NORMAL main sem-3 proj.v
5 6 32% 82:14 09:31
v . v . …/code/sem-3/afll

```
    wire zero8 = 1'b0;

    and2 x1(P[0], m[0], q[0]);
    and2 x2(A[0], m[1], q[0]);
121 and2 x3(A[1], m[2], q[0]);
    and2 x4(A[2], m[3], q[0]);
    and2 x5(A[3], m[4], q[0]);
    and2 x6(A[4], m[5], q[0]);
    and2 x7(A[5], m[6], q[0]);
    and2 x8(A[6], m[7], q[0]);

    and2 x9(B[0], m[0], q[1]);
    and2 x10(B[1], m[1], q[1]);
    and2 x11(B[2], m[2], q[1]);
    and2 x12(B[3], m[3], q[1]);
    and2 x13(B[4], m[4], q[1]);
    and2 x14(B[5], m[5], q[1]);
    and2 x15(B[6], m[6], q[1]);
    and2 x16(B[7], m[7], q[1]);

    and2 x17(C[0], m[0], q[2]);
    and2 x18(C[1], m[1], q[2]);
    and2 x19(C[2], m[2], q[2]);
    and2 x20(C[3], m[3], q[2]);
    and2 x21(C[4], m[4], q[2]);
    and2 x22(C[5], m[5], q[2]);
    and2 x23(C[6], m[6], q[2]);
    and2 x24(C[7], m[7], q[2]);

    and2 x25(D[0], m[0], q[3]);
    and2 x26(D[1], m[1], q[3]);
    and2 x27(D[2], m[2], q[3]);
    and2 x28(D[3], m[3], q[3]);
    and2 x29(D[4], m[4], q[3]);
    and2 x30(D[5], m[5], q[3]);
    and2 x31(D[6], m[6], q[3]);
    and2 x32(D[7], m[7], q[3]);

    and2 x33(E[0], m[0], q[4]);
    and2 x34(E[1], m[1], q[4]);
    and2 x35(E[2], m[2], q[4]);
    and2 x36(E[3], m[3], q[4]);
    and2 x37(E[4], m[4], q[4]);
NORMAL  main  sem-3  proj.v                                  5  6  47% 121:19  09:32
 v .    v .    …/code/sem-3/afll
```

```
    and2 x34(E[1], m[1], q[4]);
    and2 x35(E[2], m[2], q[4]);
    and2 x36(E[3], m[3], q[4]);
    and2 x37(E[4], m[4], q[4]);
160 and2 x38(E[5], m[5], q[4]);
    and2 x39(E[6], m[6], q[4]);
    and2 x40(E[7], m[7], q[4]);

    and2 x41(F[0], m[0], q[5]);
    and2 x42(F[1], m[1], q[5]);
    and2 x43(F[2], m[2], q[5]);
    and2 x44(F[3], m[3], q[5]);
    and2 x45(F[4], m[4], q[5]);
    and2 x46(F[5], m[5], q[5]);
    and2 x47(F[6], m[6], q[5]);
    and2 x48(F[7], m[7], q[5]);

    and2 x49(G[0], m[0], q[6]);
    and2 x50(G[1], m[1], q[6]);
    and2 x51(G[2], m[2], q[6]);
    and2 x52(G[3], m[3], q[6]);
    and2 x53(G[4], m[4], q[6]);
    and2 x54(G[5], m[5], q[6]);
    and2 x55(G[6], m[6], q[6]);
    and2 x56(G[7], m[7], q[6]);

    and2 x57(H[0], m[0], q[7]);
    and2 x58(H[1], m[1], q[7]);
    and2 x59(H[2], m[2], q[7]);
    and2 x60(H[3], m[3], q[7]);
    and2 x61(H[4], m[4], q[7]);
    and2 x62(H[5], m[5], q[7]);
    and2 x63(H[6], m[6], q[7]);
    and2 x64(H[7], m[7], q[7]);

    fadder f1(c_outf1, P[1],A[0],B[0],zero1);
    fadder f2(c_outf2,sum_f2,A[1],B[1],C[0]);
    fadder f3(c_outf3,sum_f3,A[2],B[2],C[1]);
    fadder f4(c_outf4, sum_f4, A[3], B[3], C[2]);
    fadder f5(c_outf5, sum_f5, A[4], B[4], C[3]);
    fadder f6(c_outf6, sum_f6, A[5], B[5], C[4]);
    fadder f7(c_outf7, sum_f7, A[6], B[6], C[5]);
    fadder f8(c_outf8, sum_f8, zero2 , B[7], C[6]);
NORMAL  main  sem-3  proj.v                                  5  6  62% 160:19  09:32
 v .    v .    …/code/sem-3/afll
```

```
  and2 x34(E[1], m[1], q[4]);
  and2 x35(E[2], m[2], q[4]);
  and2 x36(E[3], m[3], q[4]);
  and2 x37(E[4], m[4], q[4]);
160 and2 x38(E[5], m[5], q[4]);
  and2 x39(E[6], m[6], q[4]);
  and2 x40(E[7], m[7], q[4]);

  and2 x41(F[0], m[0], q[5]);
  and2 x42(F[1], m[1], q[5]);
  and2 x43(F[2], m[2], q[5]);
  and2 x44(F[3], m[3], q[5]);
  and2 x45(F[4], m[4], q[5]);
  and2 x46(F[5], m[5], q[5]);
  and2 x47(F[6], m[6], q[5]);
  and2 x48(F[7], m[7], q[5]);

  and2 x49(G[0], m[0], q[6]);
  and2 x50(G[1], m[1], q[6]);
  and2 x51(G[2], m[2], q[6]);
  and2 x52(G[3], m[3], q[6]);
  and2 x53(G[4], m[4], q[6]);
  and2 x54(G[5], m[5], q[6]);
  and2 x55(G[6], m[6], q[6]);
  and2 x56(G[7], m[7], q[6]);

  and2 x57(H[0], m[0], q[7]);
  and2 x58(H[1], m[1], q[7]);
  and2 x59(H[2], m[2], q[7]);
  and2 x60(H[3], m[3], q[7]);
  and2 x61(H[4], m[4], q[7]);
  and2 x62(H[5], m[5], q[7]);
  and2 x63(H[6], m[6], q[7]);
  and2 x64(H[7], m[7], q[7]);

  fadder f1(c_outf1, P[1],A[0],B[0],zero1);
  fadder f2(c_outf2,sum_f2,A[1],B[1],C[0]);
  fadder f3(c_outf3,sum_f3,A[2],B[2],C[1]);
  fadder f4(c_outf4, sum_f4, A[3], B[3], C[2]);
  fadder f5(c_outf5, sum_f5, A[4], B[4], C[3]);
  fadder f6(c_outf6, sum_f6, A[5], B[5], C[4]);
  fadder f7(c_outf7, sum_f7, A[6], B[6], C[5]);
  fadder f8(c_outf8, sum_f8, zero2 , B[7], C[6]);
NORMAL  main  sem-3 > proj.v                                  5  6  62% 160:19  09:32
v .   v .   …/code/sem-3/afll
```

```
  fadder f5(c_outf5, sum_f5, A[4], B[4], C[3]);
  fadder f6(c_outf6, sum_f6, A[5], B[5], C[4]);
  fadder f7(c_outf7, sum_f7, A[6], B[6], C[5]);
  fadder f8(c_outf8, sum_f8, zero2 , B[7], C[6]);
199
  fadder f9(c_outf9, P[2], sum_f2, zero3, c_outf1);
  fadder f10(c_outf10, sum_f10, sum_f3, D[0], c_outf2);
  fadder f11(c_outf11, sum_f11, sum_f4, D[1], c_outf3);
  fadder f12(c_outf12, sum_f12, sum_f5, D[2], c_outf4);
  fadder f13(c_outf13, sum_f13, sum_f6, D[3], c_outf5);
  fadder f14(c_outf14, sum_f14, sum_f7, D[4], c_outf6);
  fadder f15(c_outf15, sum_f15, sum_f8, D[5], c_outf7);
  fadder f16(c_outf16,sum_f16,C[7],D[6],c_outf8);

  fadder f17(c_outf17,P[3],sum_f10,zero4,c_outf9);
  fadder f18(c_outf18,sum_f18,sum_f11,E[0],c_outf10);
  fadder f19(c_outf19, sum_f19, sum_f12, E[1], c_outf11);
  fadder f20(c_outf20, sum_f20, sum_f13, E[2], c_outf12);
  fadder f21(c_outf21, sum_f21, sum_f14, E[3], c_outf13);
  fadder f22(c_outf22, sum_f22, sum_f15, E[4], c_outf14);
  fadder f23(c_outf23, sum_f23, sum_f16, E[5], c_outf15);
  fadder f24(c_outf24, sum_f24,D[7],E[6],c_outf16);

  fadder f25(c_outf25, P[4], sum_f18, zero5, c_outf17);
  fadder f26(c_outf26, sum_f26, sum_f19, F[0], c_outf18);
  fadder f27(c_outf27, sum_f27, sum_f20, F[1], c_outf19);
  fadder f28(c_outf28, sum_f28, sum_f21, F[2], c_outf20);
  fadder f29(c_outf29, sum_f29, sum_f22, F[3], c_outf21);
  fadder f30(c_outf30, sum_f30, sum_f23, F[4], c_outf22);
  fadder f31(c_outf31, sum_f31, sum_f24, F[5], c_outf23);
  fadder f32(c_outf32, sum_f32,E[7],F[6],c_outf24);

  fadder f33(c_outf33,P[5],sum_f26,zero6,c_outf25);
  fadder f34(c_outf34,sum_f34,sum_f27,G[0],c_outf26);
  fadder f35(c_outf35, sum_f35, sum_f28, G[1], c_outf27);
  fadder f36(c_outf36, sum_f36, sum_f29, G[2], c_outf28);
  fadder f37(c_outf37, sum_f37, sum_f30, G[3], c_outf29);
  fadder f38(c_outf38, sum_f38, sum_f31, G[4], c_outf30);
  fadder f39(c_outf39, sum_f39, sum_f32, G[5], c_outf31);
  fadder f40(c_outf40, sum_f40, F[7], G[6], c_outf32);

  fadder f41(c_outf41,P[6],sum_f34,zero7,c_outf33);
  fadder f42(c_outf42, sum_f42, sum_f35, H[0], c_outf34);
NORMAL  main  sem-3 > proj.v                                  5  6  78% 199:1  09:32
v .   v .   …/code/sem-3/afll
```

```
  fadder f37(c_outf37, sum_f37, sum_f30, G[3], c_outf29);
  fadder f38(c_outf38, sum_f38, sum_f31, G[4], c_outf30);
  fadder f39(c_outf39, sum_f39, sum_f32, G[5], c_outf31);
  fadder f40(c_outf40, sum_f40, F[7], G[6], c_outf32);
235
  fadder f41(c_outf41,P[6],sum_f34,zero7,c_outf33);
  fadder f42(c_outf42, sum_f42, sum_f35, H[0], c_outf34);
  fadder f43(c_outf43, sum_f43, sum_f36, H[1], c_outf35);
  fadder f44(c_outf44, sum_f44, sum_f37, H[2], c_outf36);
  fadder f45(c_outf45, sum_f45, sum_f38, H[3], c_outf37);
  fadder f46(c_outf46, sum_f46, sum_f39, H[4], c_outf38);
  fadder f47(c_outf47, sum_f47, sum_f40, H[5], c_outf39);
  fadder f48(c_outf48, sum_f48, G[7], H[6], c_outf40);

  fadder f49(c_outf49, P[7] , sum_f42, c_outf41, zero8);
  fadder f50(c_outf50, P[8] , sum_f43, c_outf42, c_outf49);
  fadder f51(c_outf51, P[9] , sum_f44, c_outf43, c_outf50);
  fadder f52(c_outf52, P[10] , sum_f45, c_outf44, c_outf51);
  fadder f53(c_outf53, P[11] , sum_f46, c_outf45, c_outf52);
  fadder f54(c_outf54, P[12] , sum_f47, c_outf46, c_outf53);
  fadder f55(c_outf55, P[13] , sum_f48, c_outf47, c_outf54);
  fadder f56(P[15], P[14], H[7] ,c_outf48, c_outf55);

  endmodule
```

NORMAL main sem-3 proj.v 5 6 92% 235:1 09:33

v . v . …/code/sem-3/afll

proj_tb.v

```
proj.v  proj_tb.v
 24  module proj_tb;
 23
 22  reg [7:0] i0,i1;
 21  wire [15:0] pdt;
 20
 19  carry_save_mult inst1(i0,i1,pdt);
 18
 17
 16  initial
 15  begin
 14  i0=8'b00000000; i1=8'b00000000;
 13  #100 i0=8'b00010111; i1=8'b00001110; //23,14
 12  #100 i0=8'b00001011; i1=8'b00100000; //11,32
 11  #100 i0=8'b00101101; i1=8'b00010011; //45,19
 10  #100 i0=8'b10000001; i1=8'b00101111; //129,47
  9  #100 i0=8'b00100010; i1=8'b01001000; //34,72
  8  #100 i0=8'b01011011; i1=8'b01110100; //92,116
  7  #100 i0=8'b10011101; i1=8'b00101010; //157,42
  6  #100 i0=8'b11100111; i1=8'b00011000; //231,24
  5  #100 i0=8'b11111111; i1=8'b11111111; //231,24
  4  end
  3
  2  initial
  1  begin
 25  $monitor($time," - i0 = %d, i1 = %d, product = %d", i0, i1, pdt);
  1  $dumpfile("proj_tb.vcd");
  2  $dumpvars(0, proj_tb);
  3  end
  4  endmodule
```

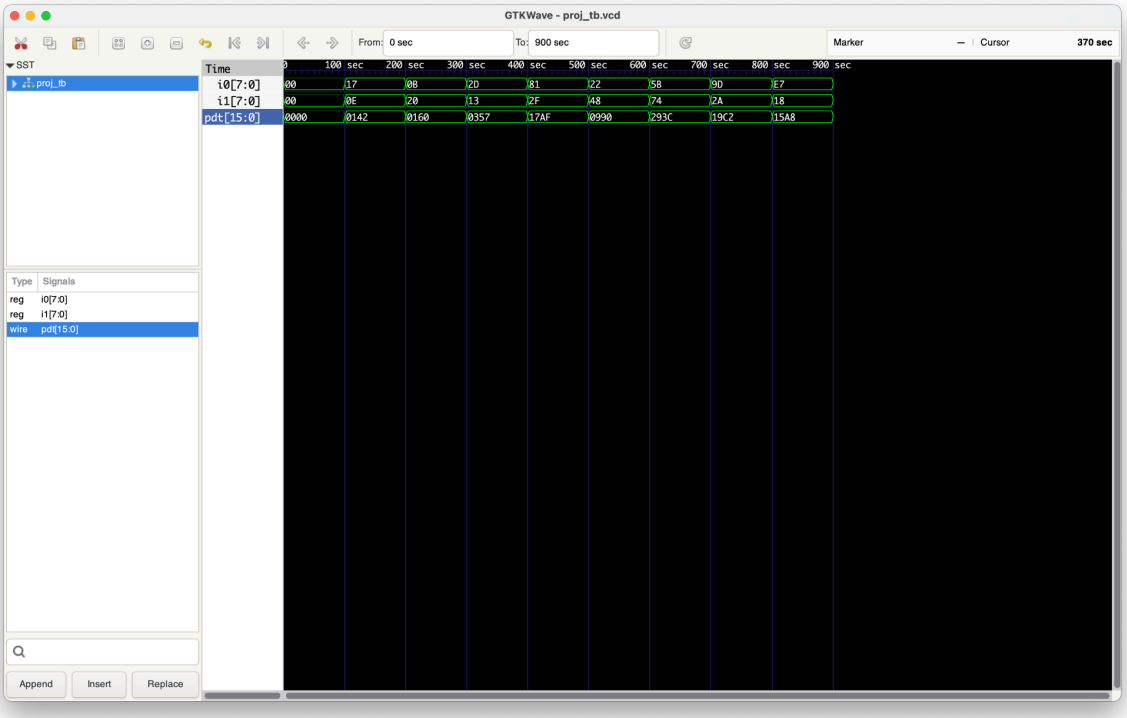
NORMAL main sem-3 proj_tb.v : 6 -1 86% 25:54 09:33

v . v . …/code/sem-3/afll

OUTPUT

```
project-final [main] ϟ make
iverilog -o proj proj_tb.v proj.v proj_lib.v
./proj
VCD info: dumpfile proj_tb.vcd opened for output.
                   0 - i0 =   0, i1 =   0, product =     0
                 100 - i0 =  23, i1 =  14, product =   322
                 200 - i0 =  11, i1 =  32, product =   352
                 300 - i0 =  45, i1 =  19, product =   855
                 400 - i0 = 129, i1 =  47, product =  6063
                 500 - i0 =  34, i1 =  72, product =  2448
                 600 - i0 =  91, i1 = 116, product = 10556
                 700 - i0 = 157, i1 =  42, product =  6594
                 800 - i0 = 231, i1 =  24, product =  5544
                 900 - i0 = 255, i1 = 255, product = 65025
project-final [main] ϟ
```

GTKWAVE


GTKWave - proj_tb.vcd
From: 0 sec
To: 900 sec
Marker
Cursor
370 sec
SST
proj_tb
Time
i0[7:0]
i1[7:0]
pdt[15:0]
100 sec
200 sec
300 sec
400 sec
500 sec
600 sec
700 sec
800 sec
900 sec
00 17 0B 2D 81 22 58 9D E7
00 0E 20 13 2F 48 74 2A 18
0000 0142 0160 0357 17AF 0990 293C 19C2 15A8
Type
Signals
reg i0[7:0]
reg i1[7:0]
wire pdt[15:0]
Append
Insert
Replace