module bm(a,b,c,carry,A,B,C,D,E,F,G);

input [2:0]carry;

input c;

input [3:0] a,b;

output A,B,C,D,E,F,G;

wire [3:0]i,j,k,m,o,q;

wire [2:0]l,n,p;

assign A= a[0] & b[0];

assign l[0]= a[1] & b[0];

assign l[1]= a[2] & b[0];

assign l[2]= a[3] & b[0];

assign i[0]= a[0] & b[1];

assign i[1]= a[1] & b[1];

assign i[2]= a[2] & b[1];

assign i[3]= a[3] & b[1];

fa f1(l[0],i[0],carry[0],B,m[0]);

fa f2(l[1],i[1],m[0],n[0],m[1]);

fa f3(l[2],i[2],m[1],n[1],m[2]);

fa f4(c,i[3],m[2],n[2],m[3]);

assign j[0]= a[0] & b[2];

assign j[1]= a[1] & b[2];

assign j[2]= a[2] & b[2];

assign j[3]= a[3] & b[2];

fa f5(j[0],n[0],carry[1],C,o[0]);

fa f6(j[1],n[1],o[0],p[0],o[1]);

fa f7(j[2],n[2],o[1],p[1],o[2]);

fa f8(j[3],m[3],o[2],p[2],o[3]);

assign k[0]= a[0] & b[3];

assign k[1]= a[1] & b[3];

assign k[2]= a[2] & b[3];

assign k[3]= a[3] & b[3];

fa f9(k[0],p[0],carry[2],D,q[0]);

fa f10(k[1],p[1],q[0],E,q[1]);

fa f11(k[2],p[2],q[1],F,q[2]);

fa f12(k[3],o[3],q[2],G,q[3]);

endmodule

module fa(a,b,c,sum,carry);

input a,b,c;

output sum,carry;

assign sum=a^b^c;

assign carry= (a & b) | (b & c) | (c & a);

endmodule