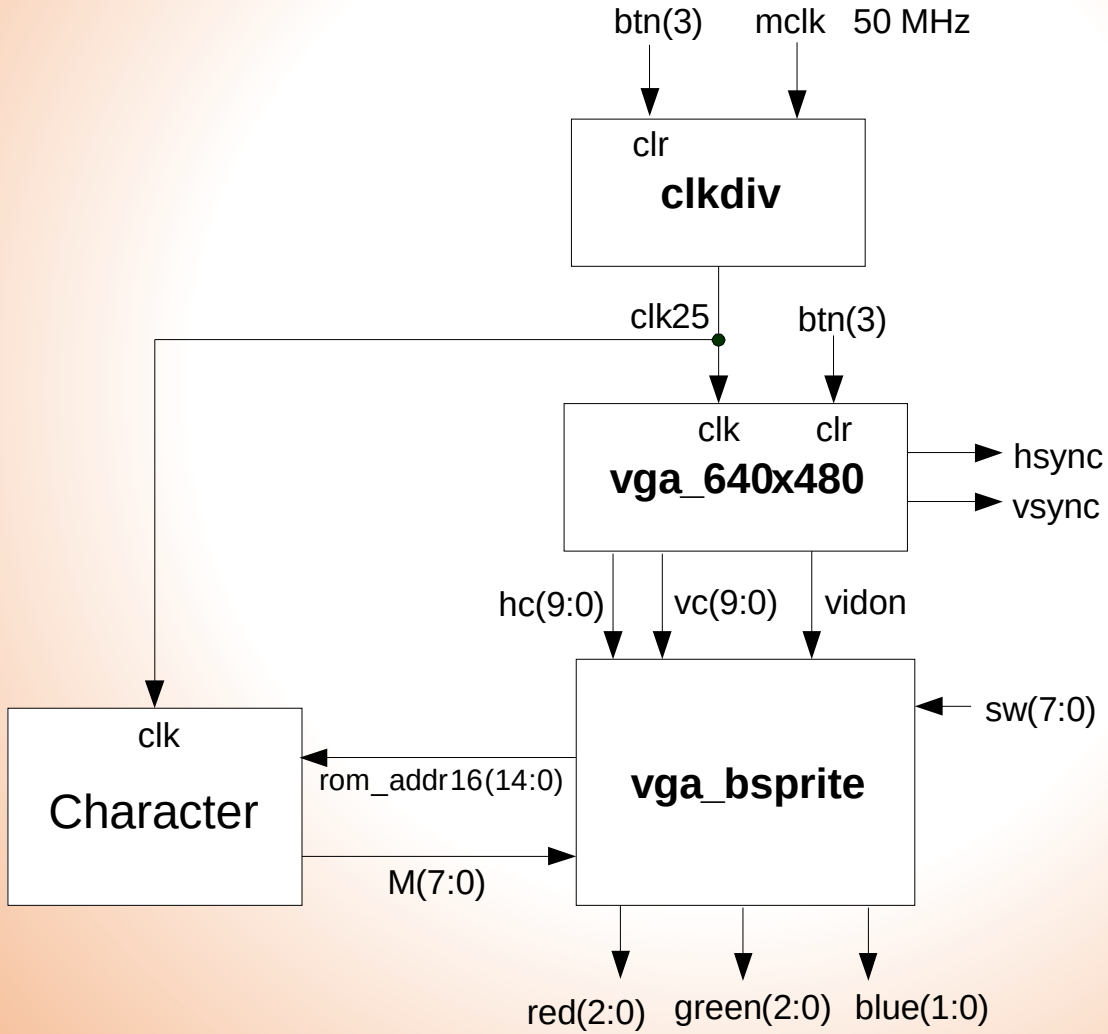
Presented by:
Michael Harris
Eric Yeh

Goal of the Game is to have tom capture jerry by having his image overlap jerry while jerry is trying to run away

Top Level Design



Creating the Characters

We Used Microsoft Pain to remove Our Heads from a picture and placing them over tom and jerry

How to move your character?

Change the values of C and R using the switches to change vbp and hbp

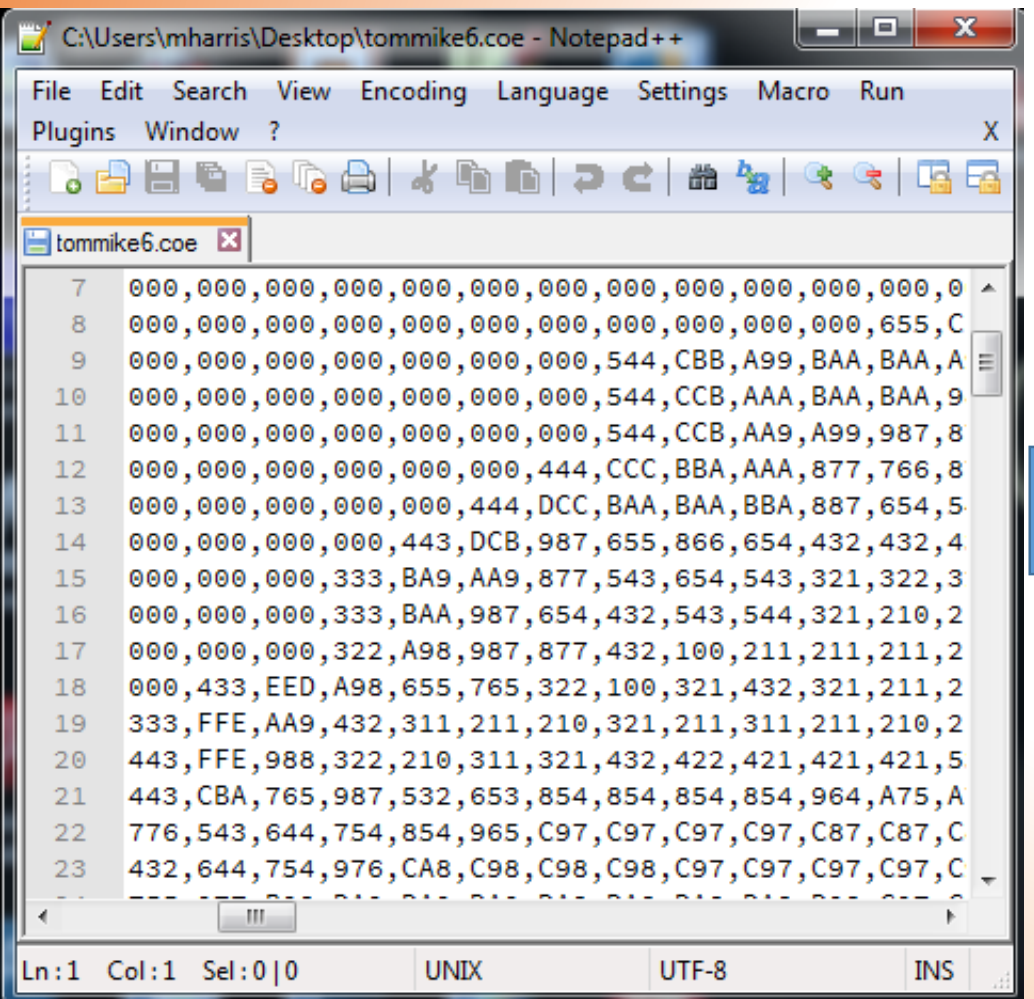
```
constant hbp : std_logic_vector(9 downto 0) := "0010010000";
constant vbp : std_logic_vector(9 downto 0) := "0000011111";
constant w : integer := 130;
constant h : integer := 130;
constant w1 : integer := 640;
constant h1 : integer := 480;

signal xpix1, ypix1, xpix2, ypix2, xpix3, ypix3  : std_logic_vector(9 downto 0);
signal C1, R1, C2, R2, C3, R3 : std_logic_vector(9 downto 0);
signal sprite1on, sprite2on, sprite3on, R, G, B : std_logic;

begin
     C1 <= '0' & sw(3 downto 0) & "00001";
     R1 <= '0' & sw(7 downto 4) & "00001";
     C2 <= '0' & sw(11 downto 8) & "00001";
     R2 <= '0' & sw(15 downto 12) & "00001";
     C3 <= '0' & "0000" & "00001";
     R3 <= '0' & "0000" & "00001";
     ypix1 <= vc - vbp - R1;
     xpix1 <= hc - hbp - C1+1;
     ypix2 <= vc - vbp - R2;
     xpix2 <= hc - hbp - C2+1;
     ypix3 <= vc - vbp - R3;
     xpix3 <= hc - hbp - C3+1;

     sprite1on <= '1' when (((hc >= C1 + hbp) and (hc < C1 + hbp + w))
                  and ((vc >= R1 + vbp) and (vc < R1 + vbp + h))) else '0';
     sprite2on <= '1' when (((hc >= C2 + hbp) and (hc < C2 + hbp + w))
                  and ((vc >= R2 + vbp) and (vc < R2 + vbp + h))) else '0';
     sprite3on <= '1' when (((hc >= C3 + hbp) and (hc < C3 + hbp + w1))
                  and ((vc >= R3 + vbp) and (vc < R3 + vbp + h1))) else '0';

     p1 : process(xpix1, ypix1, xpix2, ypix2, xpix3, ypix3)
     variable rom1_address1, rom1_address2, rom2_address1, rom2_address2, rom3_address1, rom3_address2 : std_logic_vector(19 downto 0);

     begin

          rom3_address1 := (ypix3 & "0000000000") + ("00" & ypix3 & "00000000");

          rom3_address2 := rom3_address1 + ("0000000000" & xpix3);

          rom_3_address <= rom3_address2(18 downto 0);
```

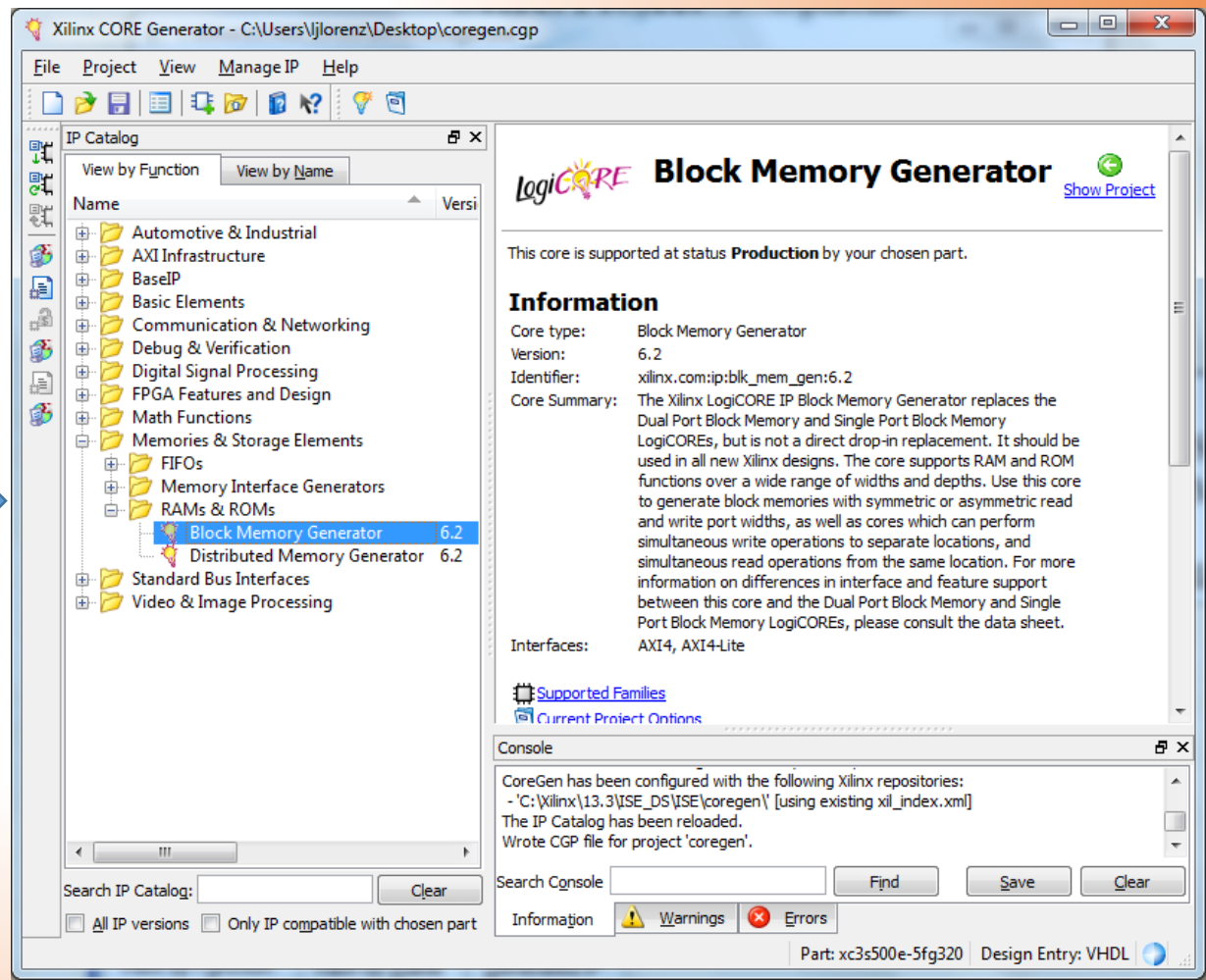
```
Phase  1  : 24426 unrouted;      REAL time: 13 secs

Phase  2  : 7139 unrouted;      REAL time: 13 secs
```

We Use the Matlab code to create the COE file For the Chosen Image

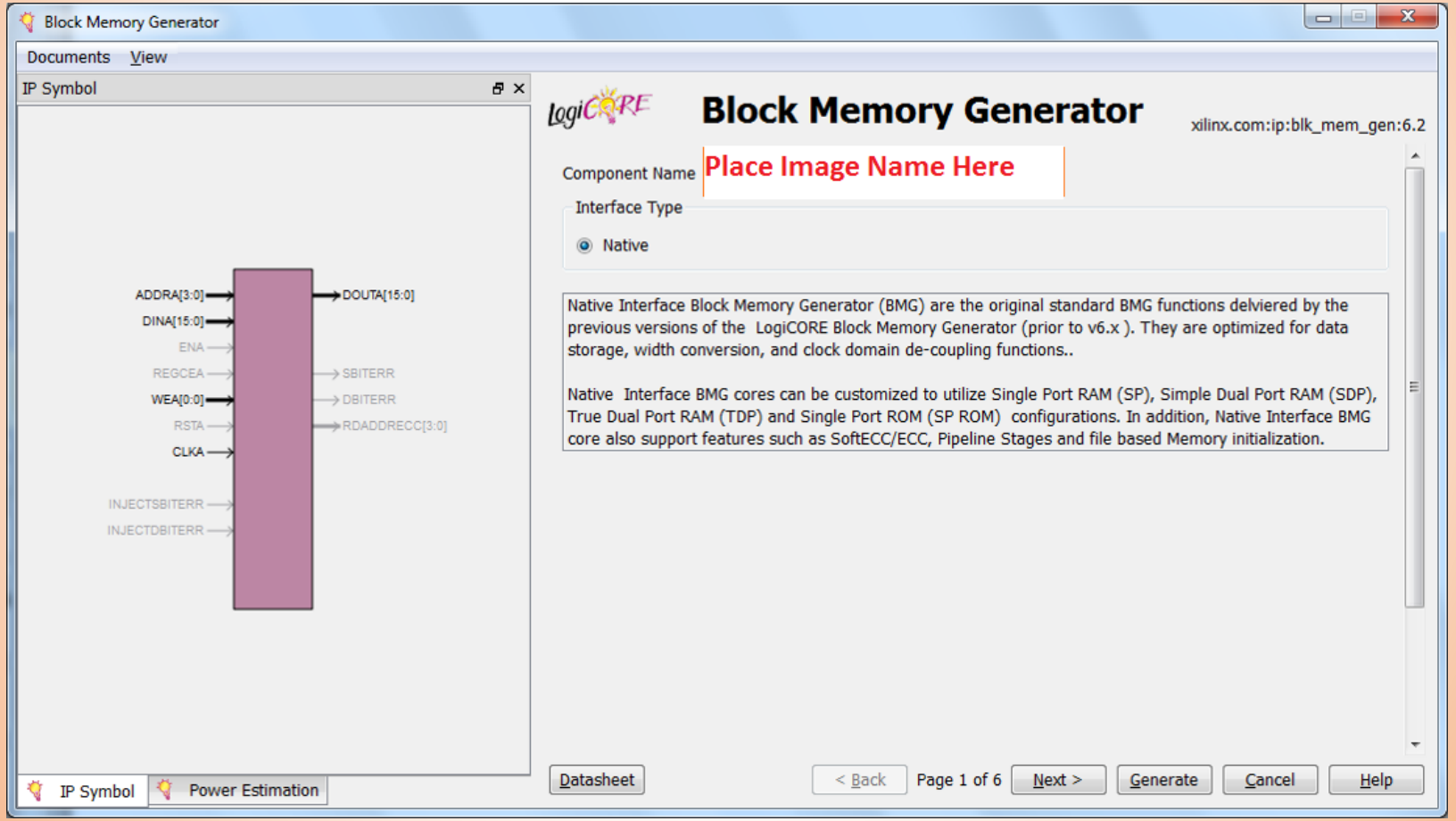
```
% Read the image
img = imread(imgfile);
h = size(img, 1); w = size(img, 2);
% Open the .coe file
s = fopen(outfile,'W');
% Print header
fprintf(s,'%s\n','; VGA Memory Map ');
fprintf(s,'%s\n','; .COE file with hex coefficients ');
fprintf(s,'; Height: %d, Width: %d\n\n', h, w);
fprintf(s,'%s\n','memory_initialization_radix=16;');
fprintf(s,'%s\n','memory_initialization_vector=');
% Convert color channels to binary
R = dec2bin(img(:,:,1)',8);
G = dec2bin(img(:,:,2)',8);
B = dec2bin(img(:,:,3)',8);
% Stitch together the output words
out = bin2dec([ R(:,1:4) G(:,1:4) B(:,1:4) ]);
img2 = img;
for i=1:h-1
   sol = i*w-w+1; % Start of line
   eol = i*w;     % End of line
   % Print out words
   fprintf(s,'%03X,',out(sol:eol,:));
   fprintf(s,'\n');
   % Save new image
   img2(i,:,1) = bin2dec(R(sol:eol,1:4))'*2^4';
   img2(i,:,2) = bin2dec(G(sol:eol,1:4))'*2^4';
   img2(i,:,3) = bin2dec(B(sol:eol,1:4))'*2^4';
end
% Print out the last row
fprintf(s,'%02X,',out(h*w-w+1:end-1,:));
fprintf(s,'%02X;',out(end,:));
img2(h,:,1) = bin2dec(R(h*w-w+1:end,1:4))'*2^4';
img2(h,:,2) = bin2dec(G(h*w-w+1:end,1:4))'*2^4';
img2(h,:,3) = bin2dec(B(h*w-w+1:end,1:4))'*2^4';
% Close the .coe file
fclose(s);
```

We Used the Core Generator to generate Image for the Created .COE file

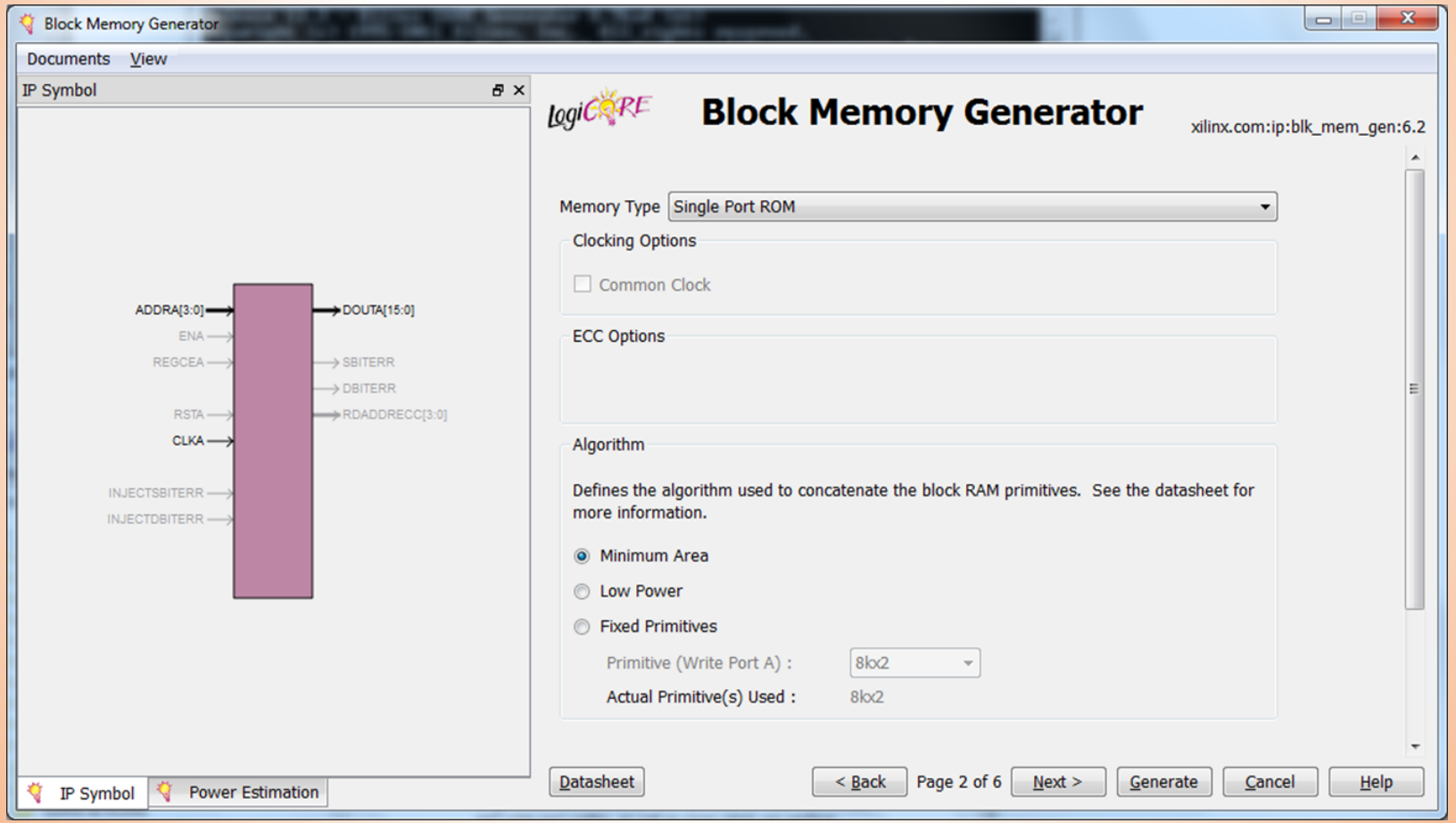




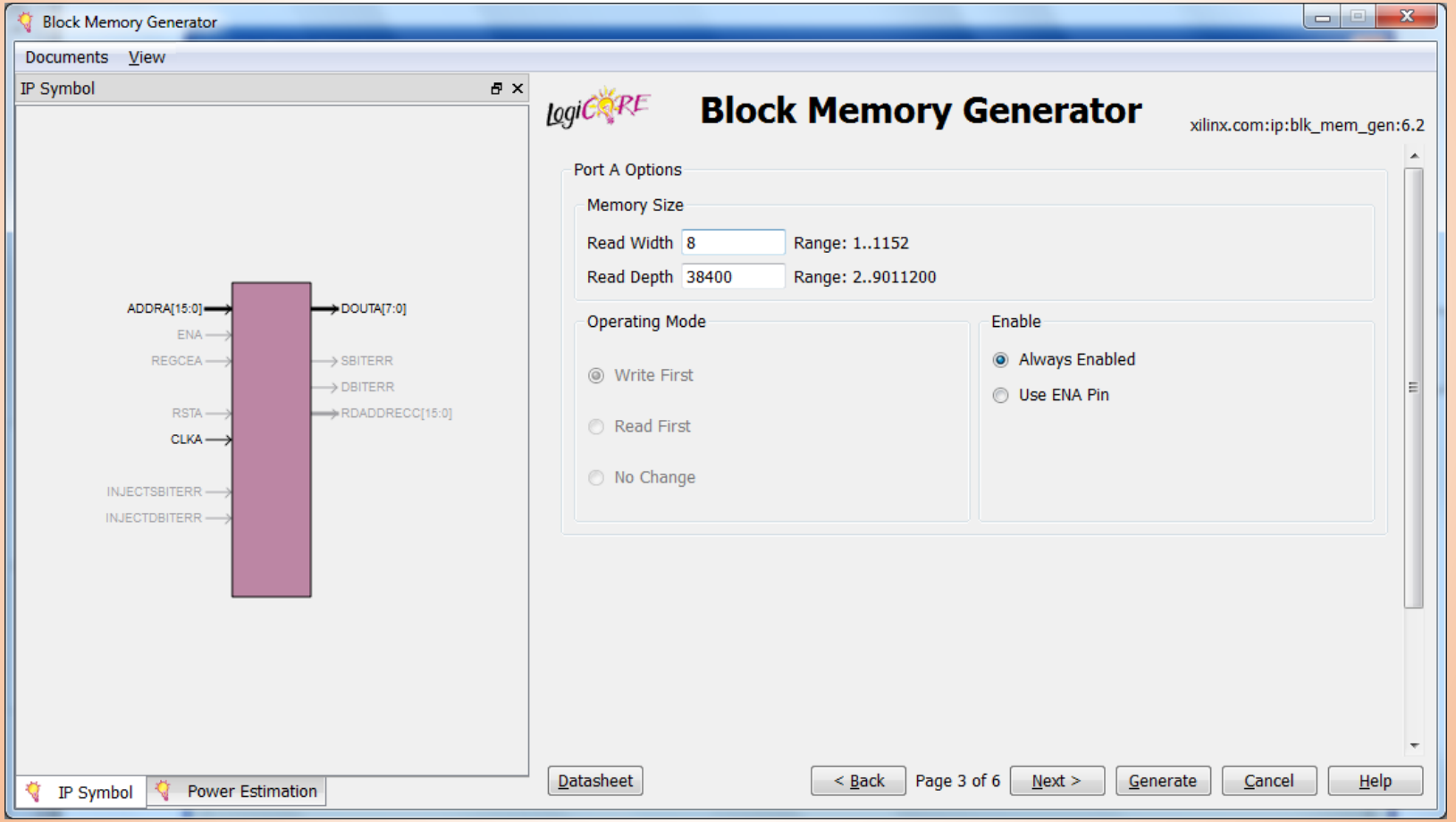
Name The image here



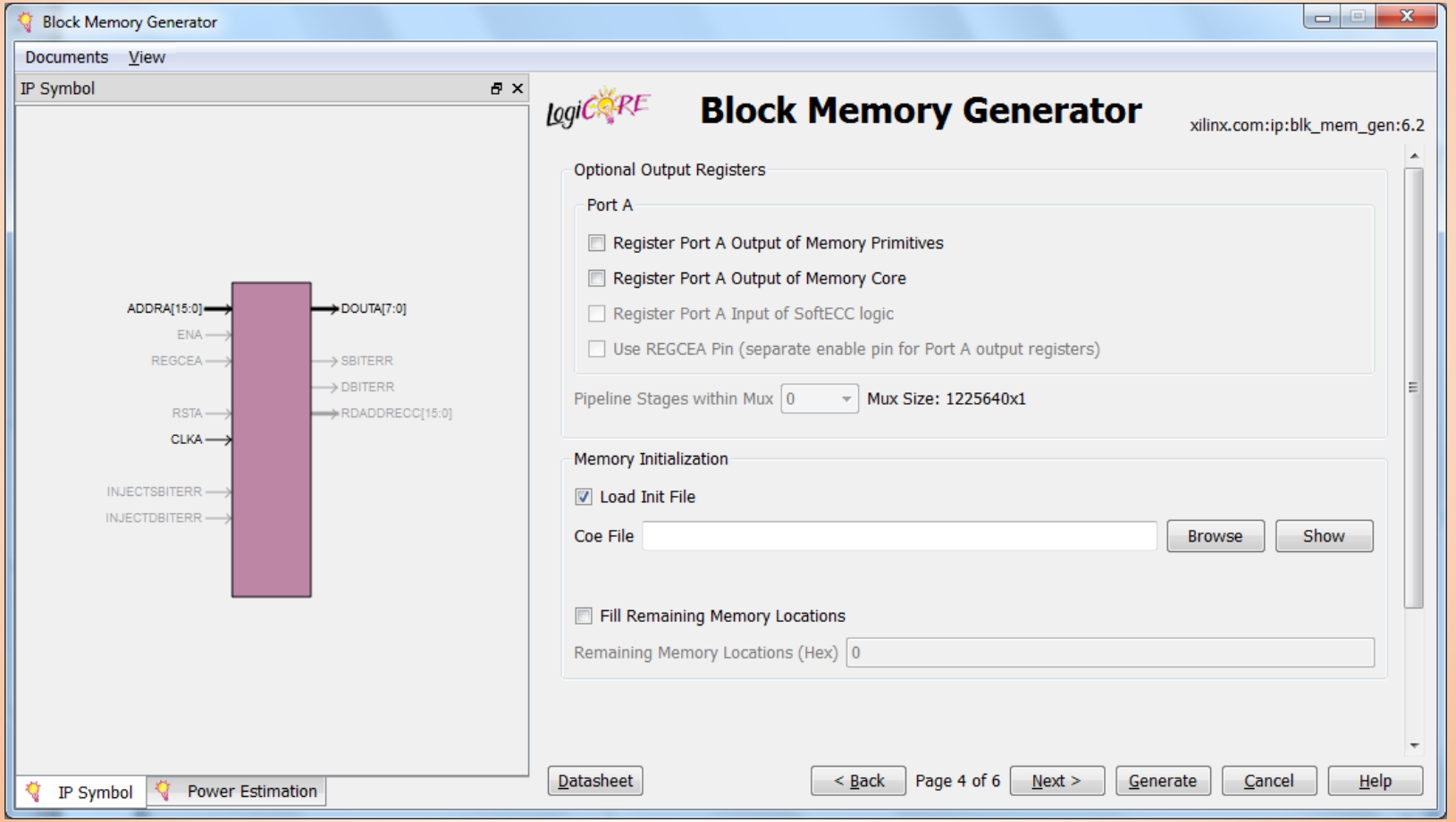
We Use Single Port Rom because it is easy and nice



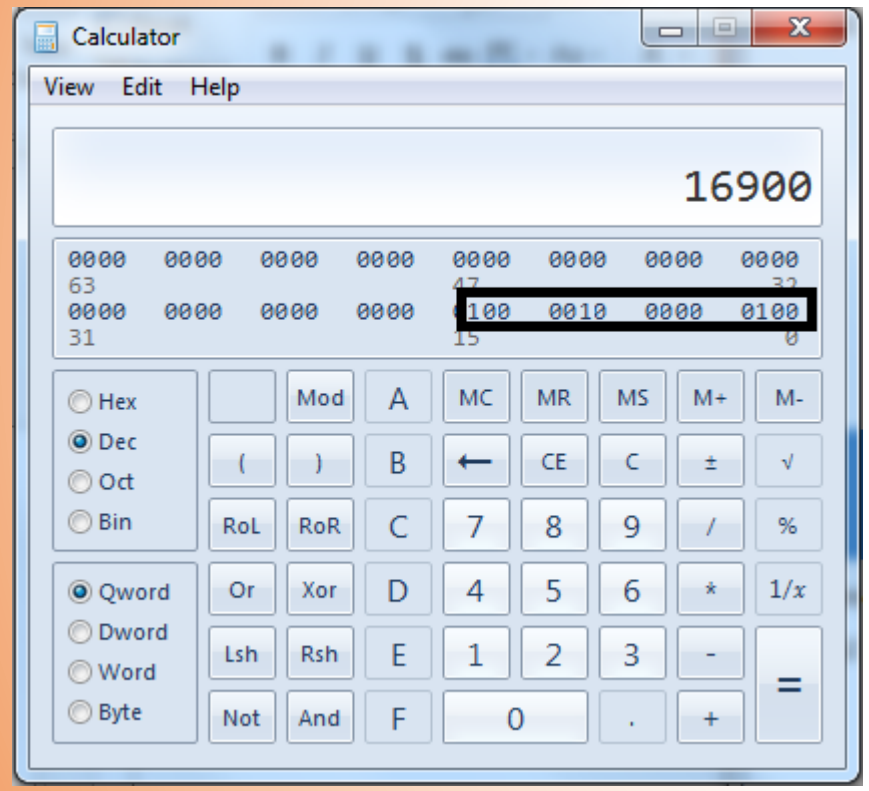
- Read Width is how many color bits for the image (we used 12 bits)
- Read Depth is the multiplication of the dimentions ex for a 130x130 pic the read depth is 130x130=16900



We Browse looking for that coe file then we generate it as a block memory



Finding the Address size for the sprite
Using the windows calculator (programmer mode)



We are using 15 bits for the 130x130 image

So address <=(14 downto 0);

Demo Time