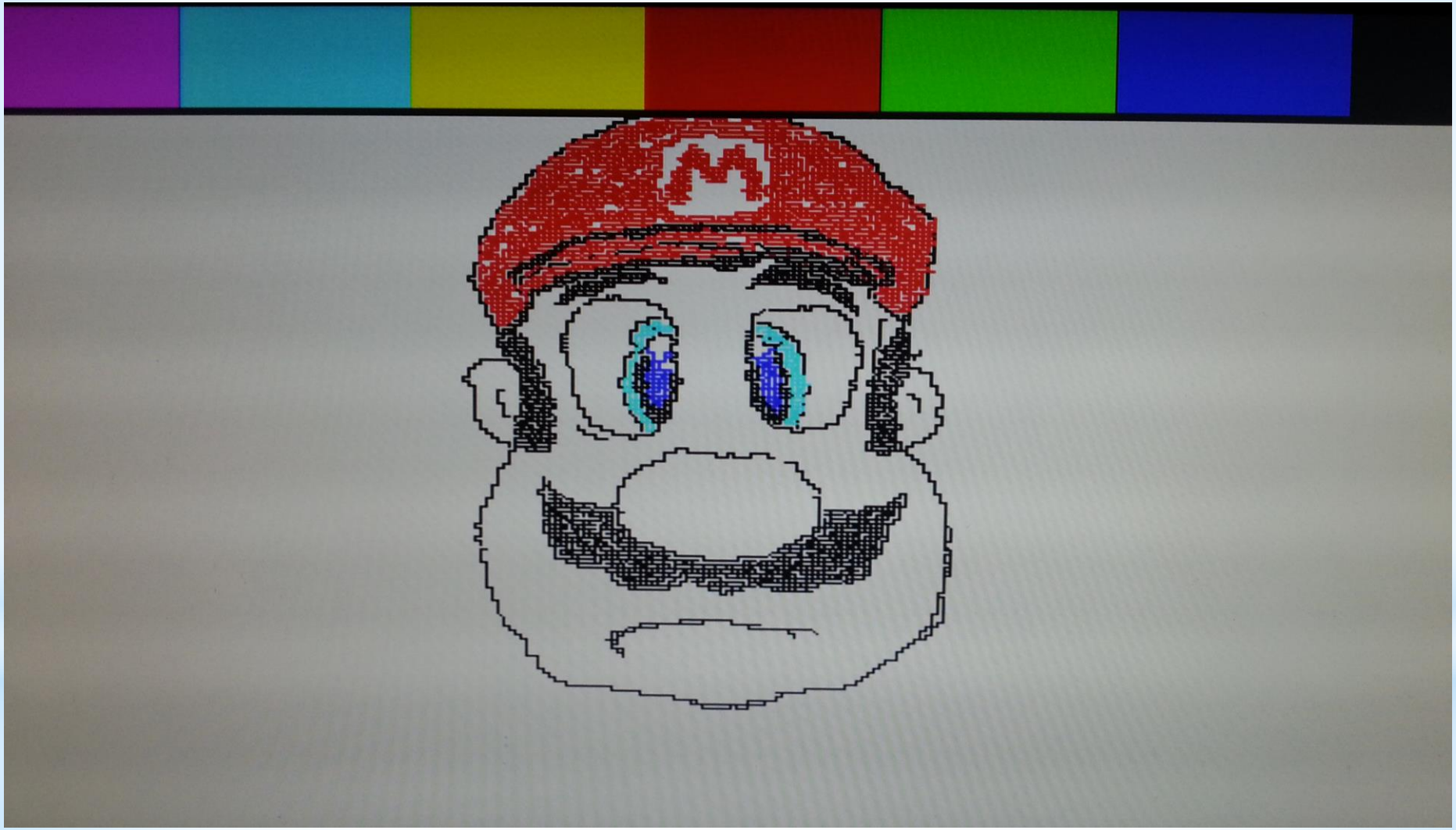
Oakland University
Electrical and Computer Engineering Department
ECE 378 – Final Project

Paint Tool

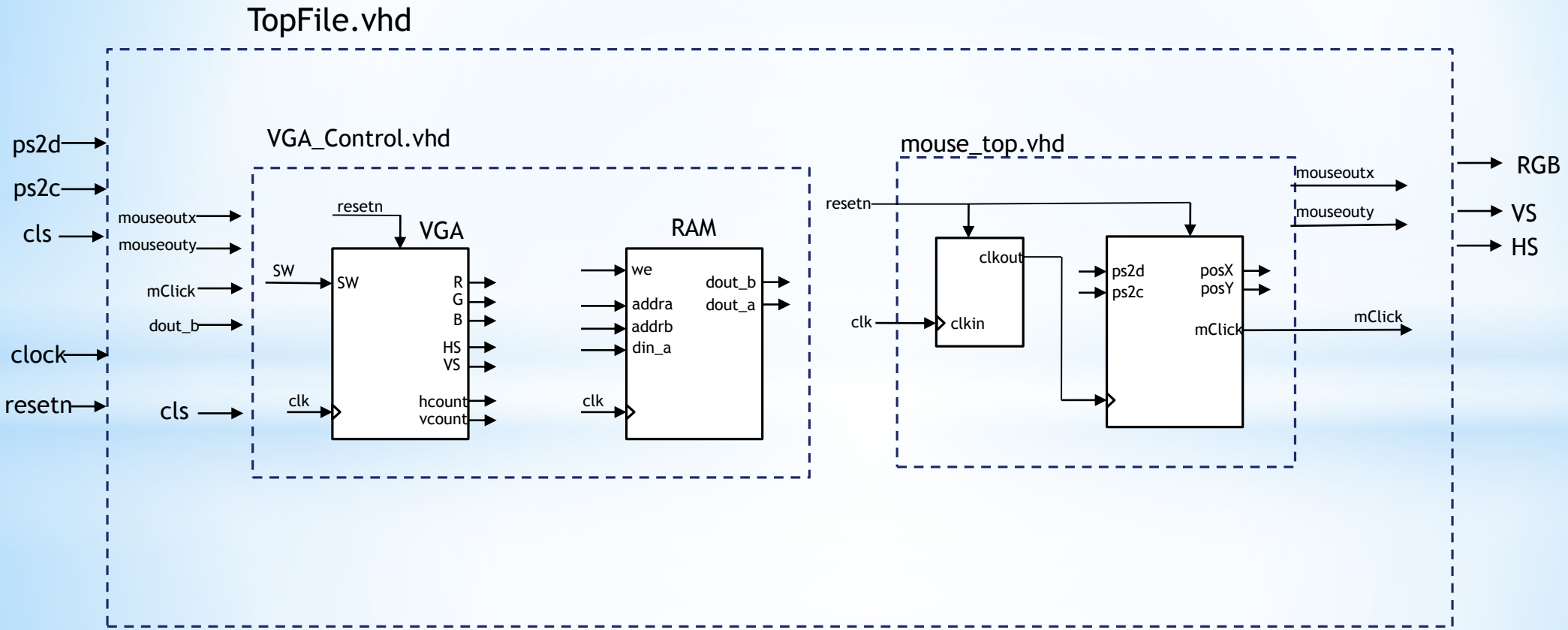
Duncan Taylor
Daniel Wolfe
Cabir Yavuz

04.14.2015

Paint Tool

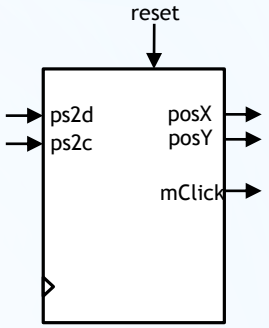


Datapath Circuit



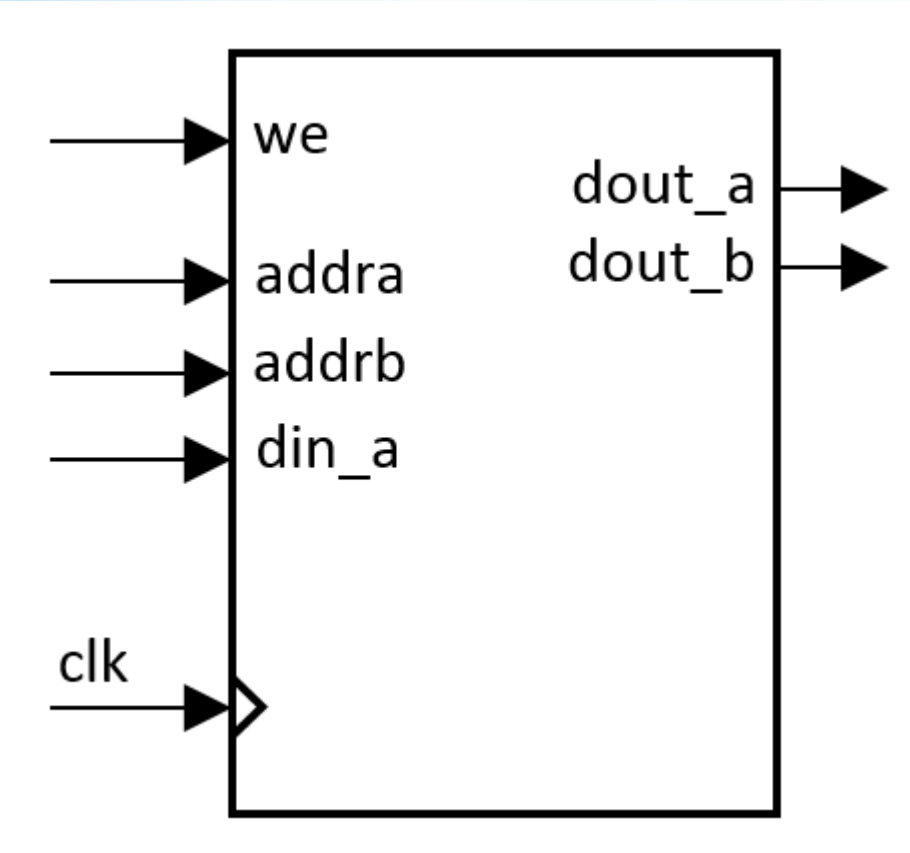
PS2 Mouse - Received Data Structure

Register 1	1	P	y7	y6	y5	y4	y3	y2	y1	y0	0
Register 2	1	P	x7	x6	x5	x4	x3	x2	x1	x0	0
Register 3	1	P	yv	xv	ys	xs	1	0	R	L	0



PS2 Input → Four Directional Input → Mouse Cursor Coordinates (PosX and PosY)

Dual Port Ram Memory




```
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
entity xilinx_dual_port_ram_sync is
   generic(
      ADDR_WIDTH: integer:=19;
      DATA_WIDTH:integer:=4
   );
   port(
      clk: in std_logic;
      we: in std_logic;
      addr_a: in std_logic_vector(ADDR_WIDTH-1 downto 0);
      addr_b: in std_logic_vector(ADDR_WIDTH-1 downto 0);
      din_a: in std_logic_vector(DATA_WIDTH-1 downto 0);
      dout_a: out std_logic_vector(DATA_WIDTH-1 downto 0);
      dout_b: out std_logic_vector(DATA_WIDTH-1 downto 0)
   );
end xilinx_dual_port_ram_sync;

architecture beh_arch of xilinx_dual_port_ram_sync is
   type ram_type is array (0 to 2**ADDR_WIDTH-1)
        of std_logic_vector (DATA_WIDTH-1 downto 0);
   signal ram: ram_type;
   signal addr_a_reg, addr_b_reg:
          std_logic_vector(ADDR_WIDTH-1 downto 0);
begin
   process(clk)
   begin
     if (clk'event and clk = '1') then
        if (we = '1') then
           ram(to_integer(unsigned(addr_a))) <= din_a;
        end if;
        addr_a_reg <= addr_a;
        addr_b_reg <= addr_b;
     end if;
   end process;
   dout_a <= ram(to_integer(unsigned(addr_a_reg)));
   dout_b <= ram(to_integer(unsigned(addr_b_reg)));
end beh_arch;
```

Assigning the RAM Addresses

Address Width = log2 (640 * 480) = 19 Bits

Data Width = 3 Bits

Address “a” is used to write, so it is set to the mouse cursor

```
addr_a <= conv_std_logic_vector(conv_integer(mouseposx) * 640 + conv_integer(mouseposy), 19);
```

Address “b” is used to read, so it cycles through every pixel with hcount and vcount

```
addr_b <= conv_std_logic_vector(conv_integer(hcount) * 640 + conv_integer(vcount), 19);
```

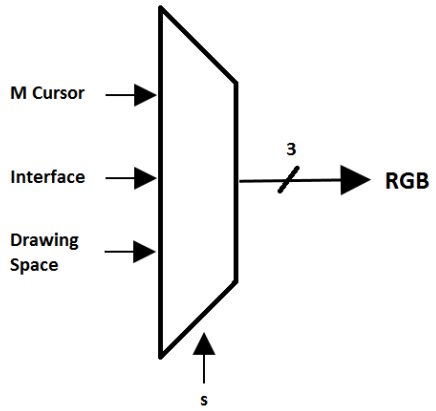
VGA - MUX

Mouse Cursor

Interface

Drawing Space

```
if  (hcount > (mouseposx - mousew)) and (hcount < (mouseposx + mousew)) and (vcount > (mouseposy - mouseh)) and (vcount < (mouseposy + mouseh)) then
   s <= "1010";
elsif hcount <= 80 and vcount < 51 then
   s <= "1001";
elsif hcount > 80 and hcount <= 160 and vcount < 51 then
   s <= "1000";
elsif hcount > 160 and hcount <= 240 and vcount < 51 then
   s <= "0111";
elsif hcount > 240 and hcount <= 320 and vcount < 51 then
   s <= "0110";
elsif hcount > 320 and hcount <= 400 and vcount < 51 then
   s <= "0101";
elsif hcount > 400 and hcount <= 480 and vcount < 51 then
   s <= "0100";
elsif hcount > 480 and hcount <= 560 and vcount < 51 then
   s <= "0011";
elsif hcount > 560 and hcount <= 640 and vcount < 51 then
   s <= "0010";
elsif (hcount > wall_l) and (hcount < (640-wall_l)) and (vcount > wall_t) and (vcount < (wall_t + wall_k)) then
   s <= "0001";
else
   s <= "0000";
end if;
```




```
with s select
   SW <= mousecolor when "1010",
         "000000000101" when "1001",
         "000000000011" when "1000",
         "000000000110" when "0111",
         "000000000100" when "0110",
         "000000000010" when "0101",
         "000000000001" when "0100",
         "000000000000" when "0011",
         "000000000111" when "0010",
         "000000000000" when "0001",
         ("000000000" & colorinb) when others;
```