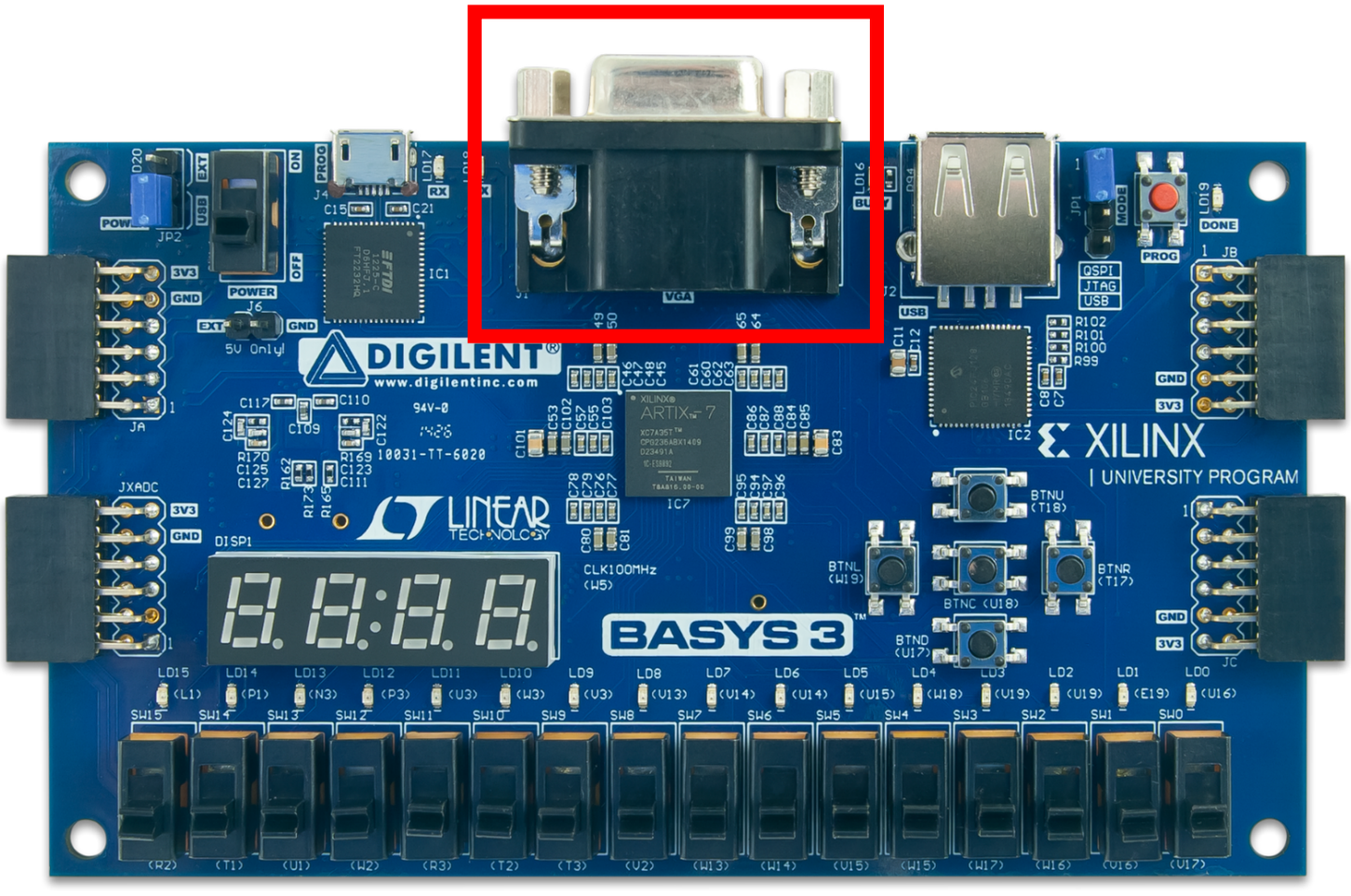
VGA

Hsi-Pin Ma

http://lms.nthu.edu.tw/course/43639
Department of Electrical Engineering
National Tsing Hua University

VGA Port

VGA


VGA
DIGILENT
www.digilentinc.com
BASYS 3
XILINX
UNIVERSITY PROGRAM
LINEAR TECHNOLOGY
JA
JXADC
JB
JC
DISP1
CLK100MHz
(W5)
BTNU (T18)
BTNL (W19)
BTNC (U18)
BTNR (T17)
BTND (U17)
LD15 (L1)
LD14 (P1)
LD13 (N3)
LD12 (P3)
LD11 (U3)
LD10 (W3)
LD9 (V3)
LD8 (V13)
LD7 (V14)
LD6 (U14)
LD5 (U15)
LD4 (W18)
LD3 (V19)
LD2 (U19)
LD1 (E19)
LD0 (U16)
SW15 (R2)
SW14 (T1)
SW13 (U1)
SW12 (W2)
SW11 (R3)
SW10 (T2)
SW9 (T3)
SW8 (V2)
SW7 (W13)
SW6 (W14)
SW5 (V15)
SW4 (W15)
SW3 (W17)
SW2 (W16)
SW1 (V16)
SW0 (V17)

VGA

- VGA = Video Graphics Array
- Introduced by IBM in 1987, and still used today
- Transmitting analog signal

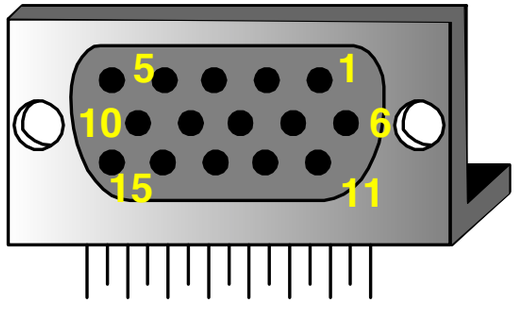
Cathode-Ray Tube Monitor

Video Graphics Array
DE-15 female and male connector

LED Monitor

VGA Video Signal

- A VGA video signal contains 5 active signals (RGBHV)
  - horizontal sync (HS): used for video synchronization in the horizontal direction
  - vertical sync (VS): used for video synchronization in the vertical direction
  - red (R): used to control the red color, 0v (fully off) ~ 0.7v (fully on)
  - green (G): used to control the green color, 0v (fully off) ~ 0.7v (fully on)
  - blue (B): used to control the blue color, 0v (fully off) ~ 0.7v (fully on)

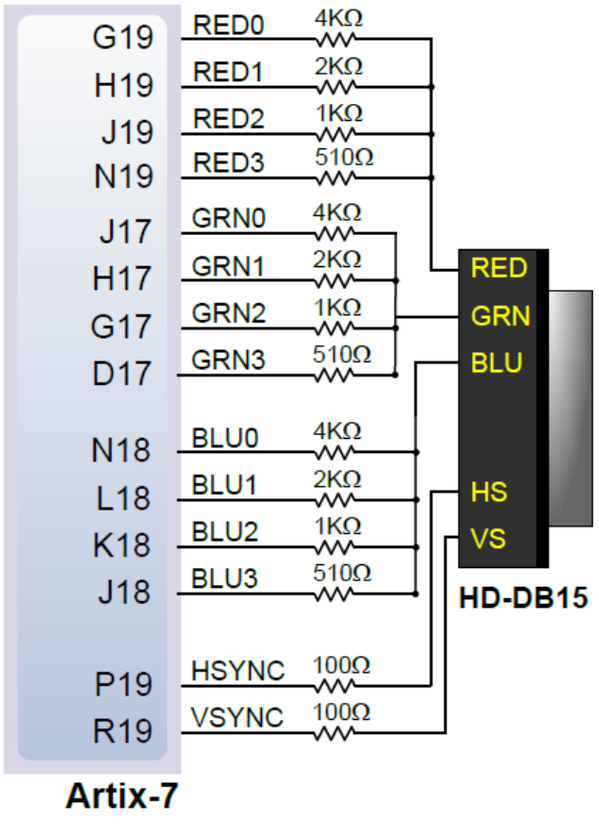


Pin 1: Red	Pin 5: GND
Pin 2: Grn	Pin 6: Red GND
Pin 3: Blue	Pin 7: Grn GND
Pin 13: HS	Pin 8: Blu GND
Pin 14: VS	Pin 10: Sync GND

Basys 3 Control Signals for VGA

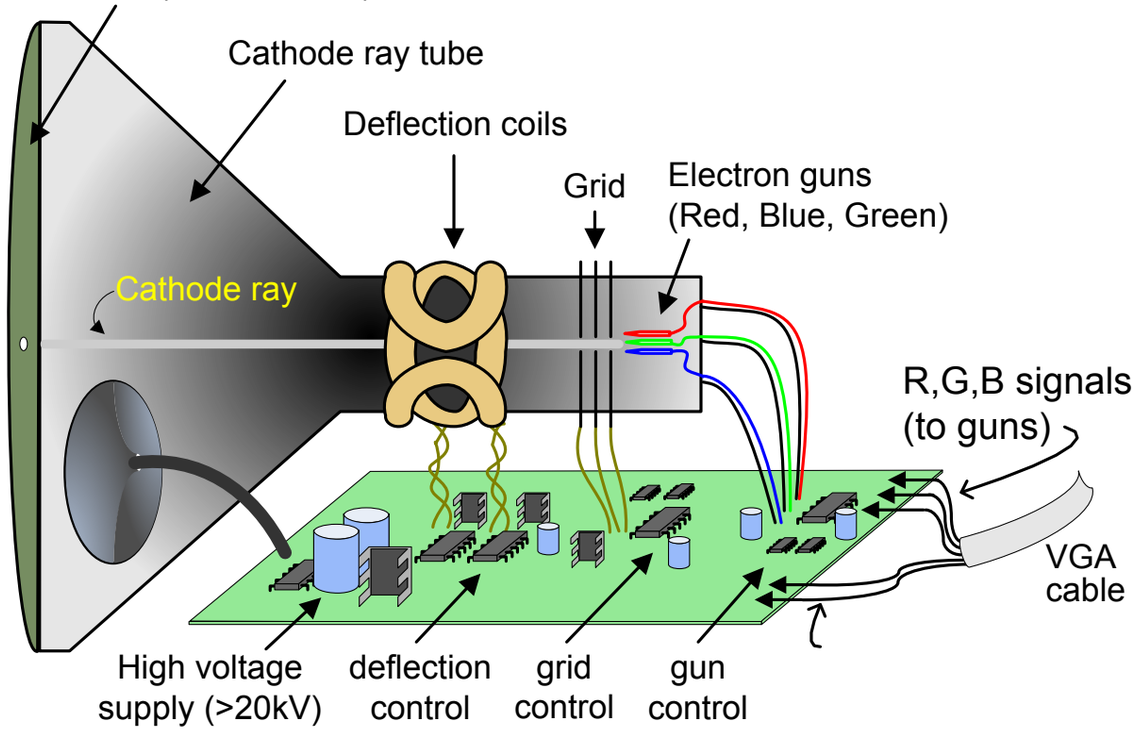
- 14 FPGA pins
  - 4-bits per color (R, G, B)
  - 2 standard sync signals (HS, VS)

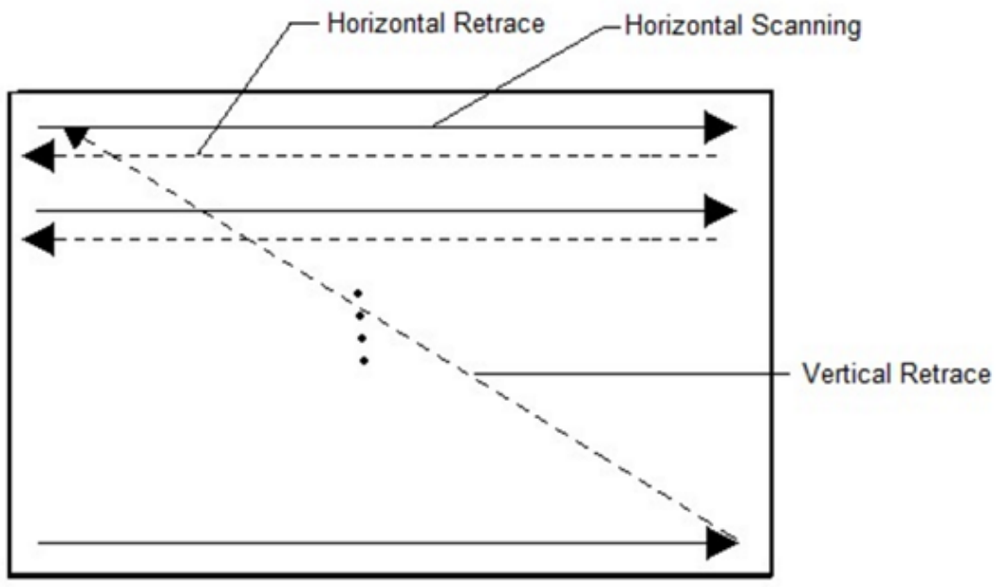
```
##VGA Connector
set_property PACKAGE_PIN G19 [get_ports {vgaRed[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {vgaRed[0]}]
set_property PACKAGE_PIN H19 [get_ports {vgaRed[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {vgaRed[1]}]
set_property PACKAGE_PIN J19 [get_ports {vgaRed[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {vgaRed[2]}]
set_property PACKAGE_PIN N19 [get_ports {vgaRed[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {vgaRed[3]}]
set_property PACKAGE_PIN N18 [get_ports {vgaBlue[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {vgaBlue[0]}]
set_property PACKAGE_PIN L18 [get_ports {vgaBlue[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {vgaBlue[1]}]
set_property PACKAGE_PIN K18 [get_ports {vgaBlue[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {vgaBlue[2]}]
set_property PACKAGE_PIN J18 [get_ports {vgaBlue[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {vgaBlue[3]}]
set_property PACKAGE_PIN J17 [get_ports {vgaGreen[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {vgaGreen[0]}]
set_property PACKAGE_PIN H17 [get_ports {vgaGreen[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {vgaGreen[1]}]
set_property PACKAGE_PIN G17 [get_ports {vgaGreen[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {vgaGreen[2]}]
set_property PACKAGE_PIN D17 [get_ports {vgaGreen[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {vgaGreen[3]}]
set_property PACKAGE_PIN P19 [get_ports hsync]
set_property IOSTANDARD LVCMOS33 [get_ports hsync]
set_property PACKAGE_PIN R19 [get_ports vsync]
set_property IOSTANDARD LVCMOS33 [get_ports vsync]
```



CRT

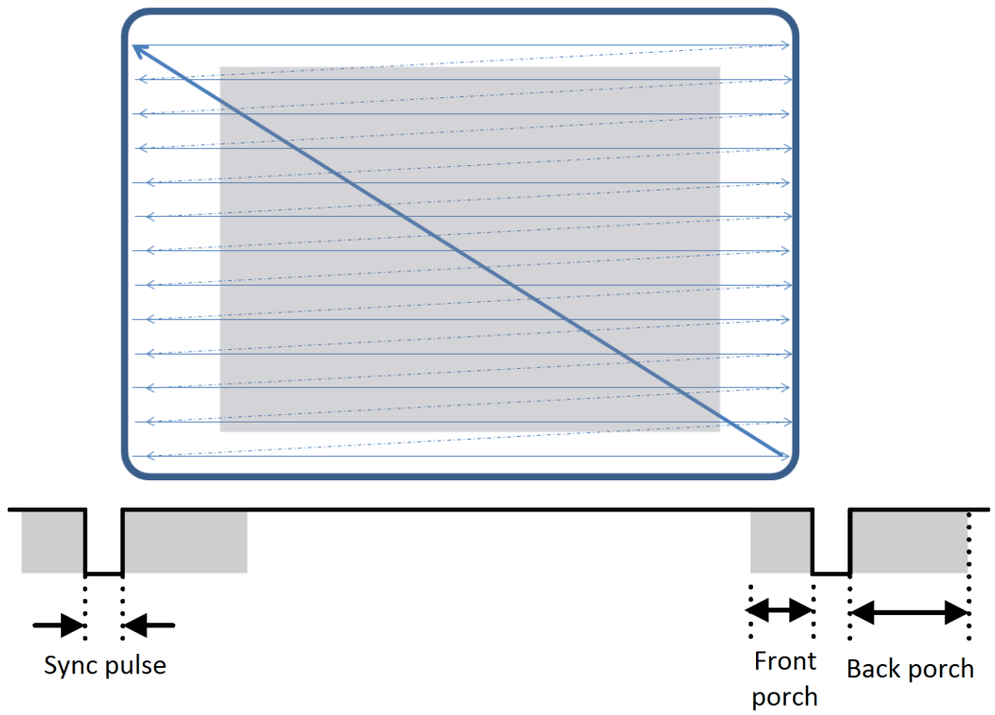
- Cathode Ray Tube (CRT) is a vacuum tube containing one or more electron guns, and a phosphorescent screen is used to view images.





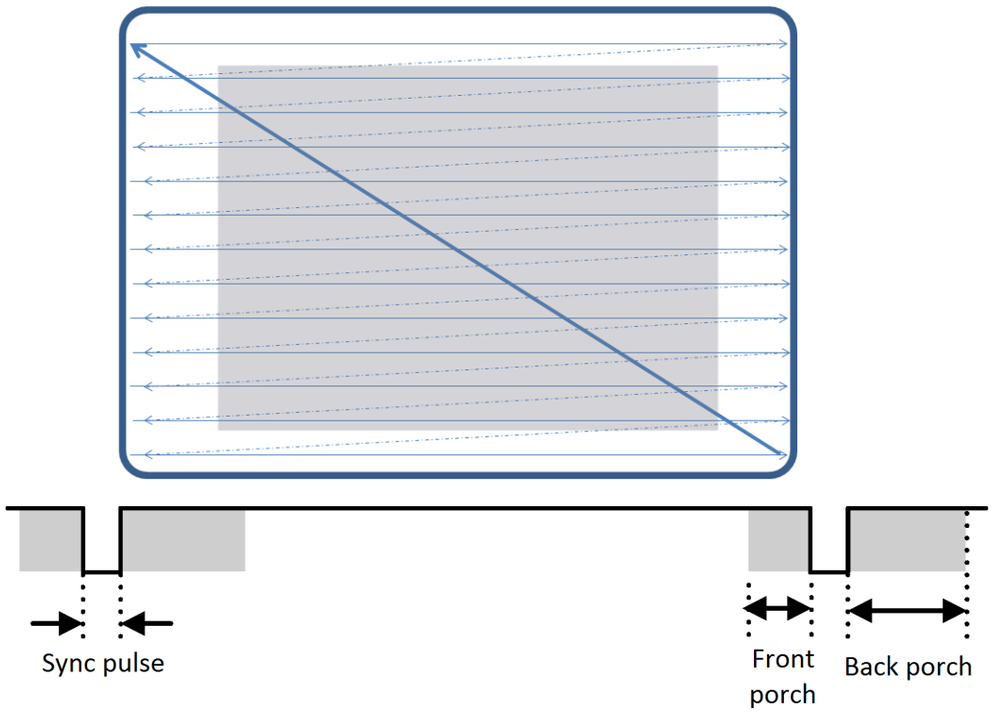
VGA System Timing (1/3)

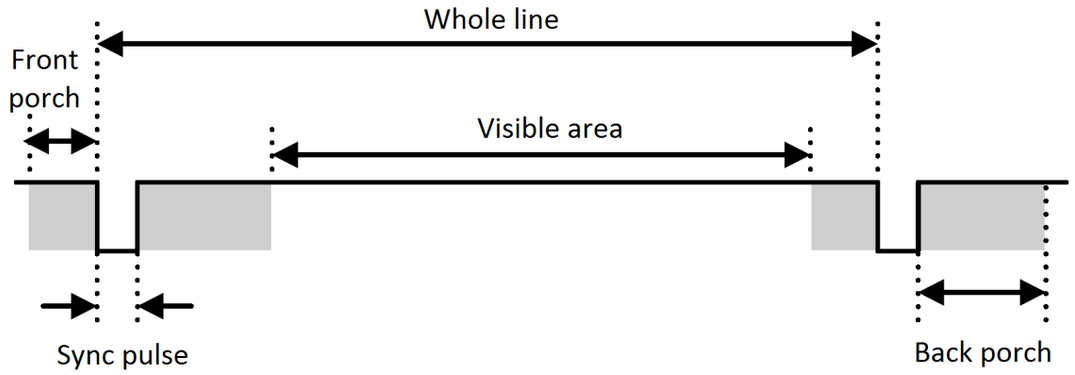
- Whether the information is displayed
  - Displayed: beam moving forward (left to right and top to bottom)
  - Not displayed: the time the beam is reset back to the left or top edge of the display. (Blanking period)
- Display resolution is determined by
  - the size of the beams
  - the frequency at which the beam can be traced across the display
  - the frequency at which the electron beam can be modulated



VGA System Timing (3/3)

- Signal timing for a 640-pixel by 480 rows display using a 25MHz pixel clock





Parameter	Ver. Sync		Hor. Sync	
	Lines	Time(ms)	Pixels	Time(μs)
Visible area	480	15.3	640	25
Front porch	10	0.3	16	0.64
Sync pulse	2	0.064	96	3.8
Back porch	33	1.05	48	1.9
Whole line	525	16.7	800	32

Pixel Clock

- The pixel clock defines the time available to display one pixel of information.
- Example: Suppose we want to display an image with 480 rows and 640 columns, and its refresh rate is 60Hz. The pixel clock which will be delivered to VGA screen must be

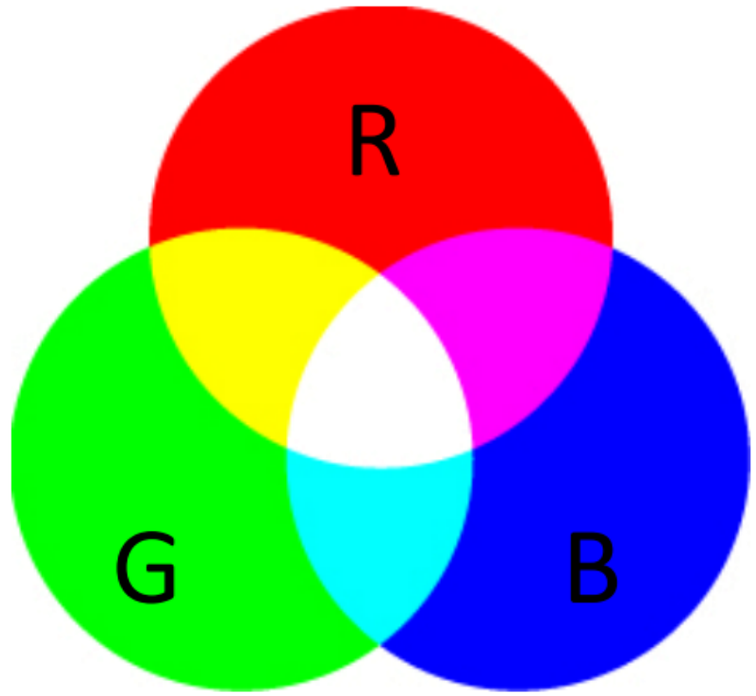
$$800*525*60(\text{frame/sec}) = 25\text{M (pixel/sec)}$$

RGB Bitmap

- A digital color image is composed by a lot of pixels.
- Each pixel contains three R, G, B values to represent the intensity of these three primary colors.

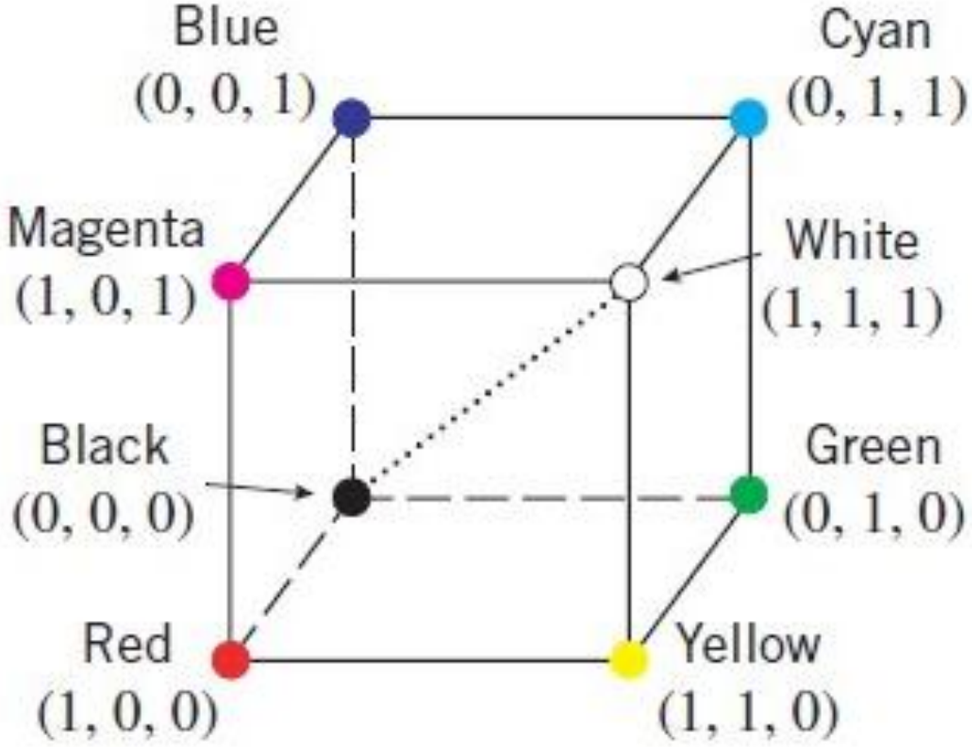
RGB Color Mode

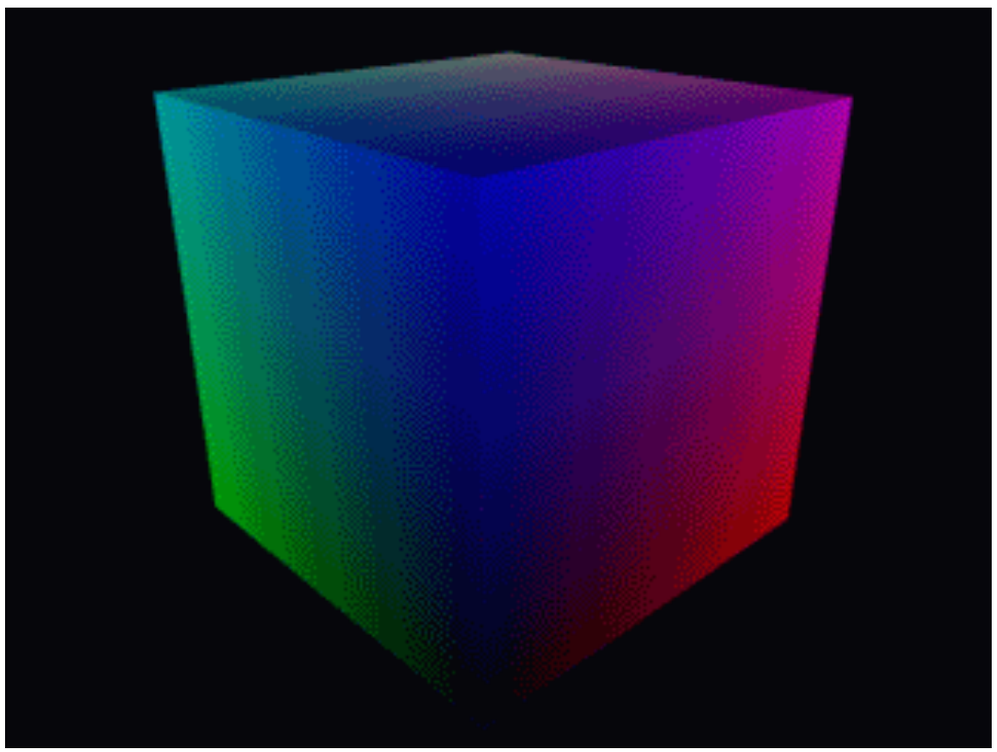
- Three primary colors
  - **Red**
  - **Green**
  - **Blue**
- No one of them can be created as the other two.
- Any other color is a combination



RGB Color Cube

- R, G, and B correspond to three axes in 3D space.
- Normalize the relative amounts of R, G and B so that each value varies between 0 and 1.

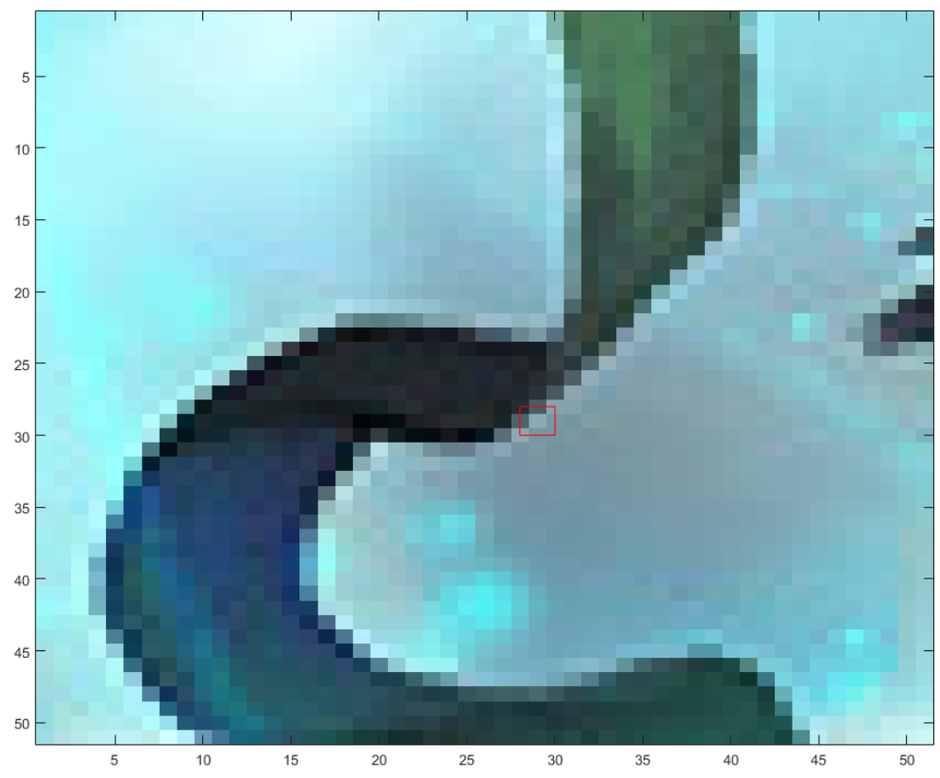




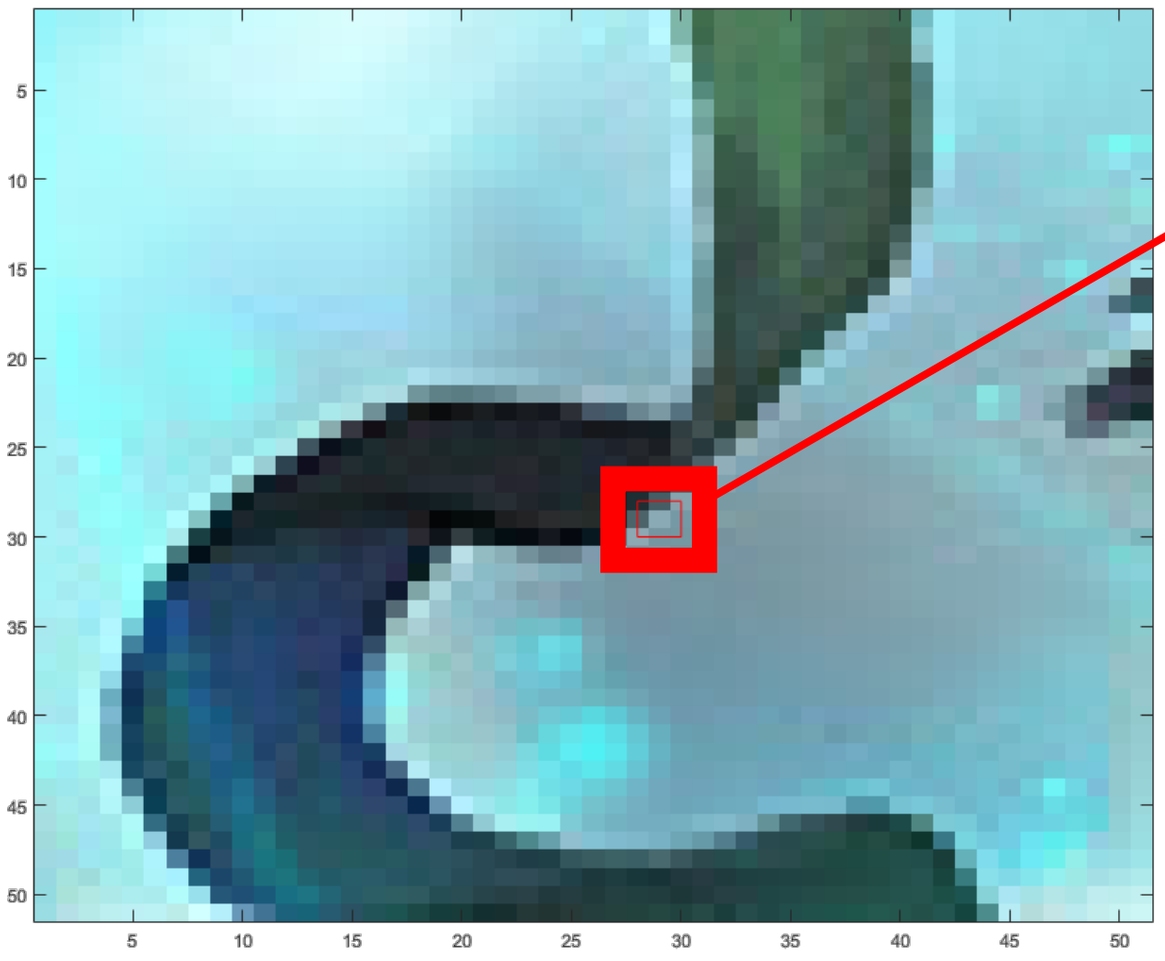
RGB Bitmap Example (1/2)

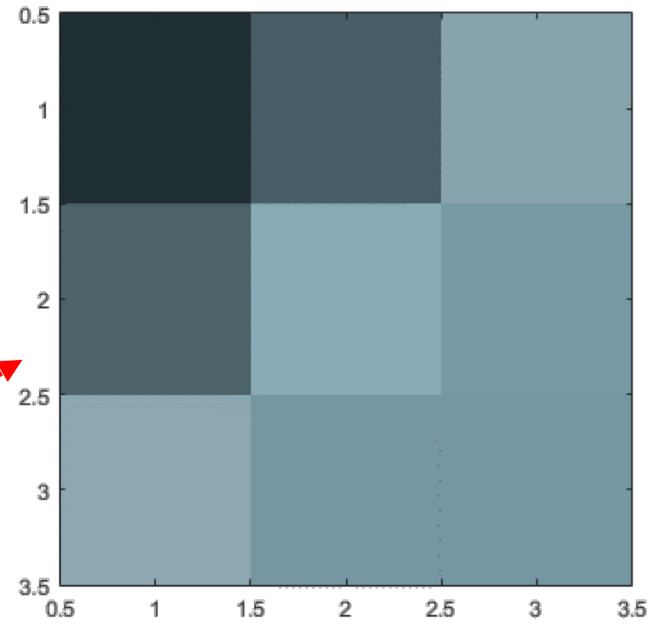
Size: 1215*717

Size: 51*51



RGB Bitmap Example (2/2)



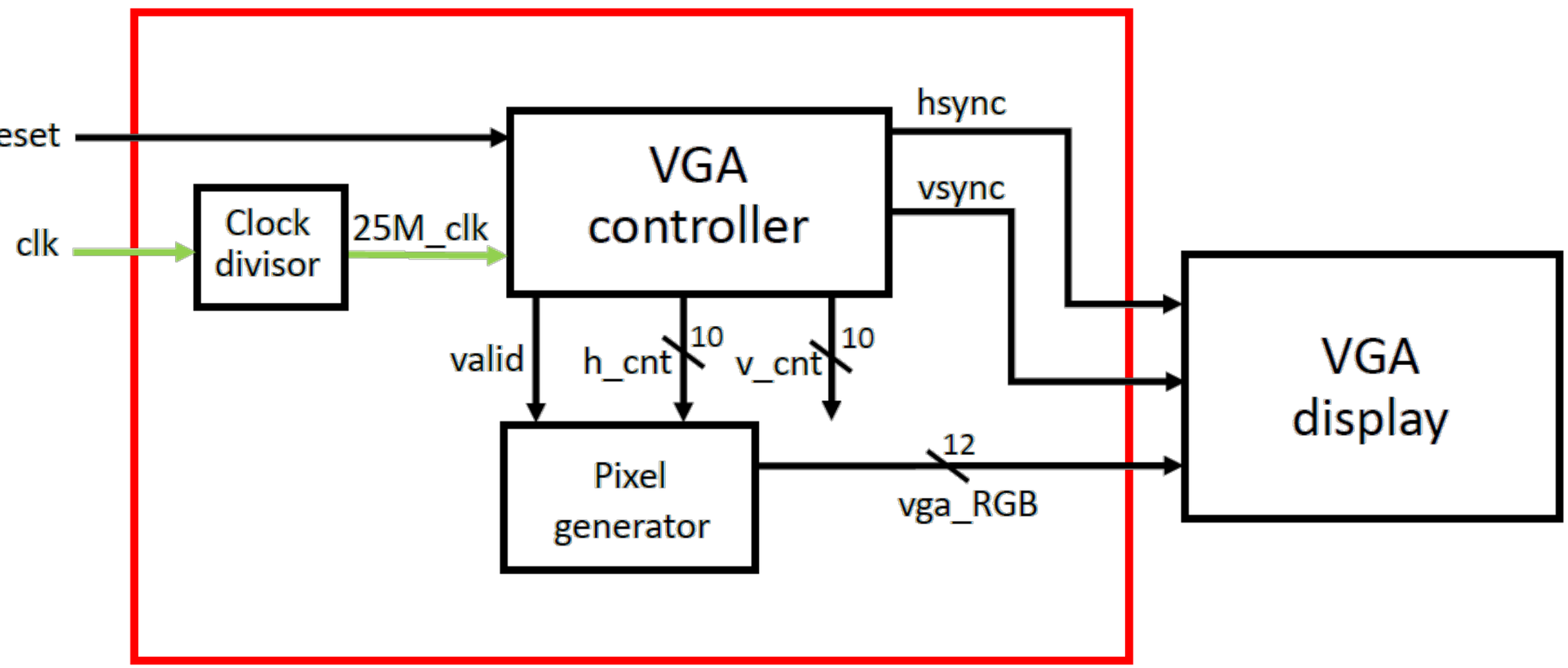


(R,G,B) : range from 0 to 255

(31,46,51)	(70,93,101)	(135,163,174)
(76,99,105)	(138,170,181)	(117,153,165)
(141,168,179)	(118,150,161)	(117,151,161)

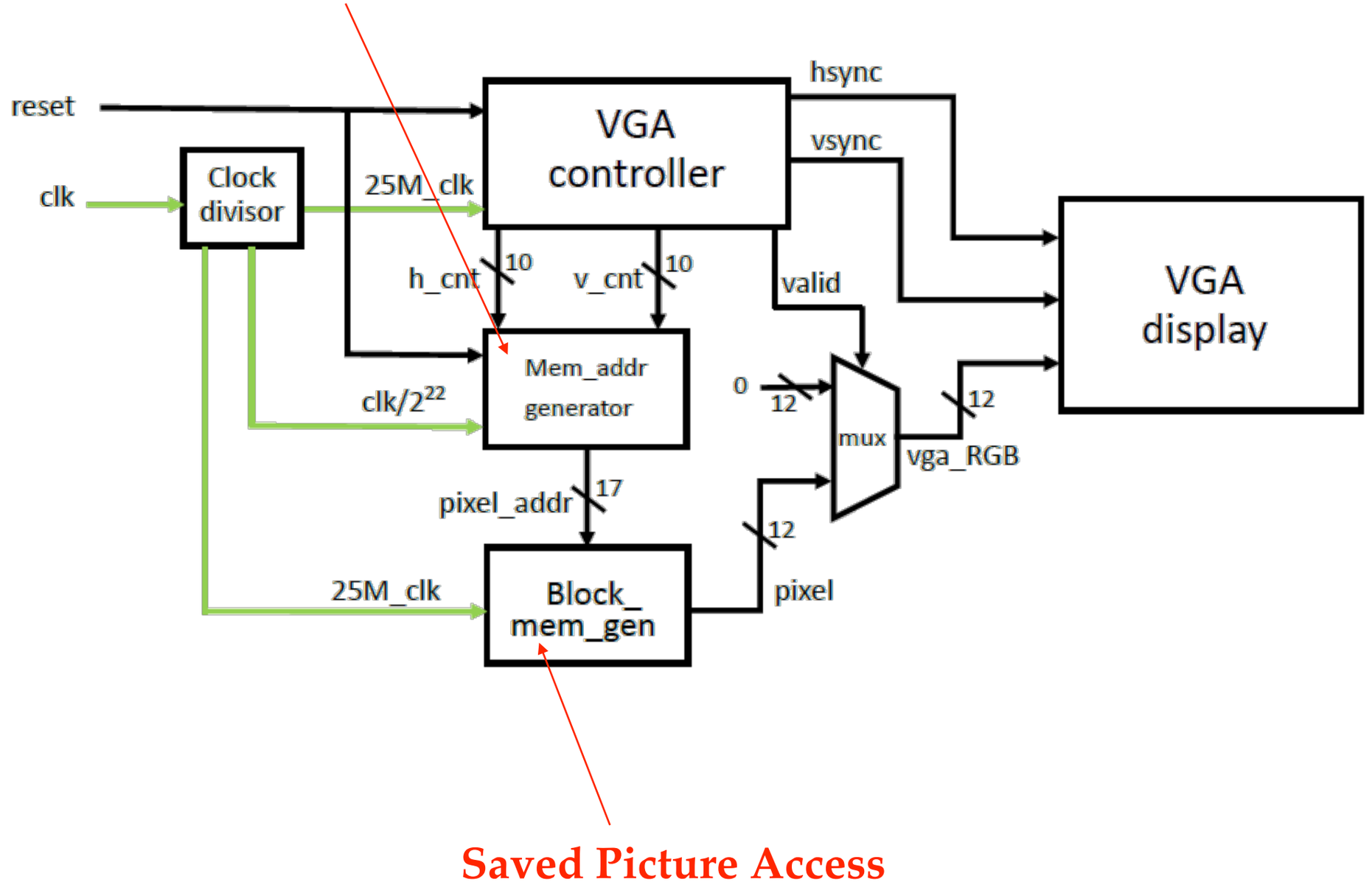
Demo 1: Block Diagram

Top



Demo 2: Block Diagram

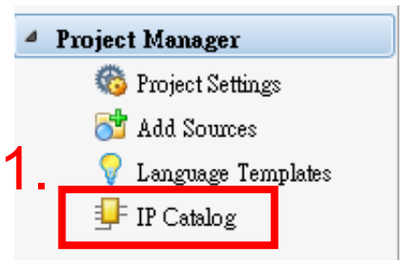
reduce the resolution from 640x480 to 320x240



Saved Picture Access

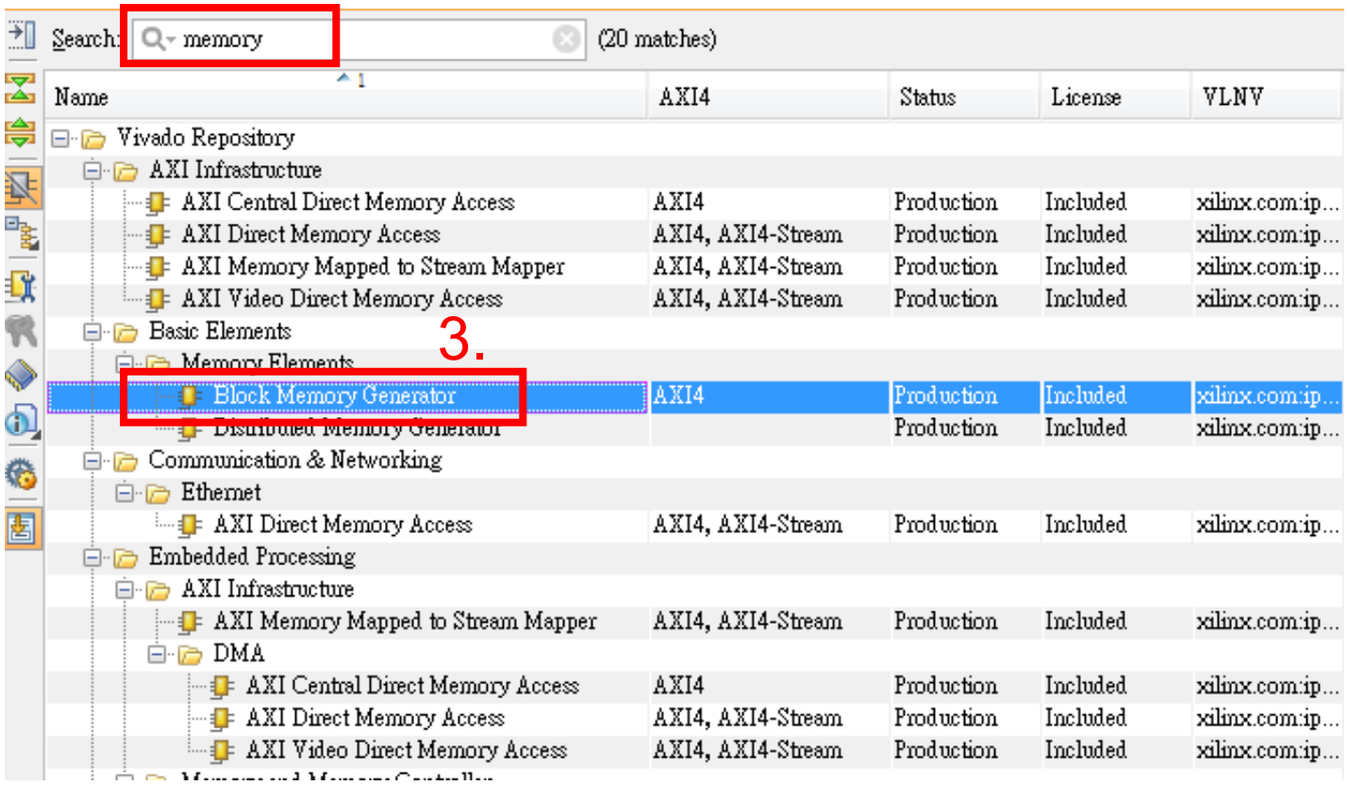
Memory IP (1/5)

1.

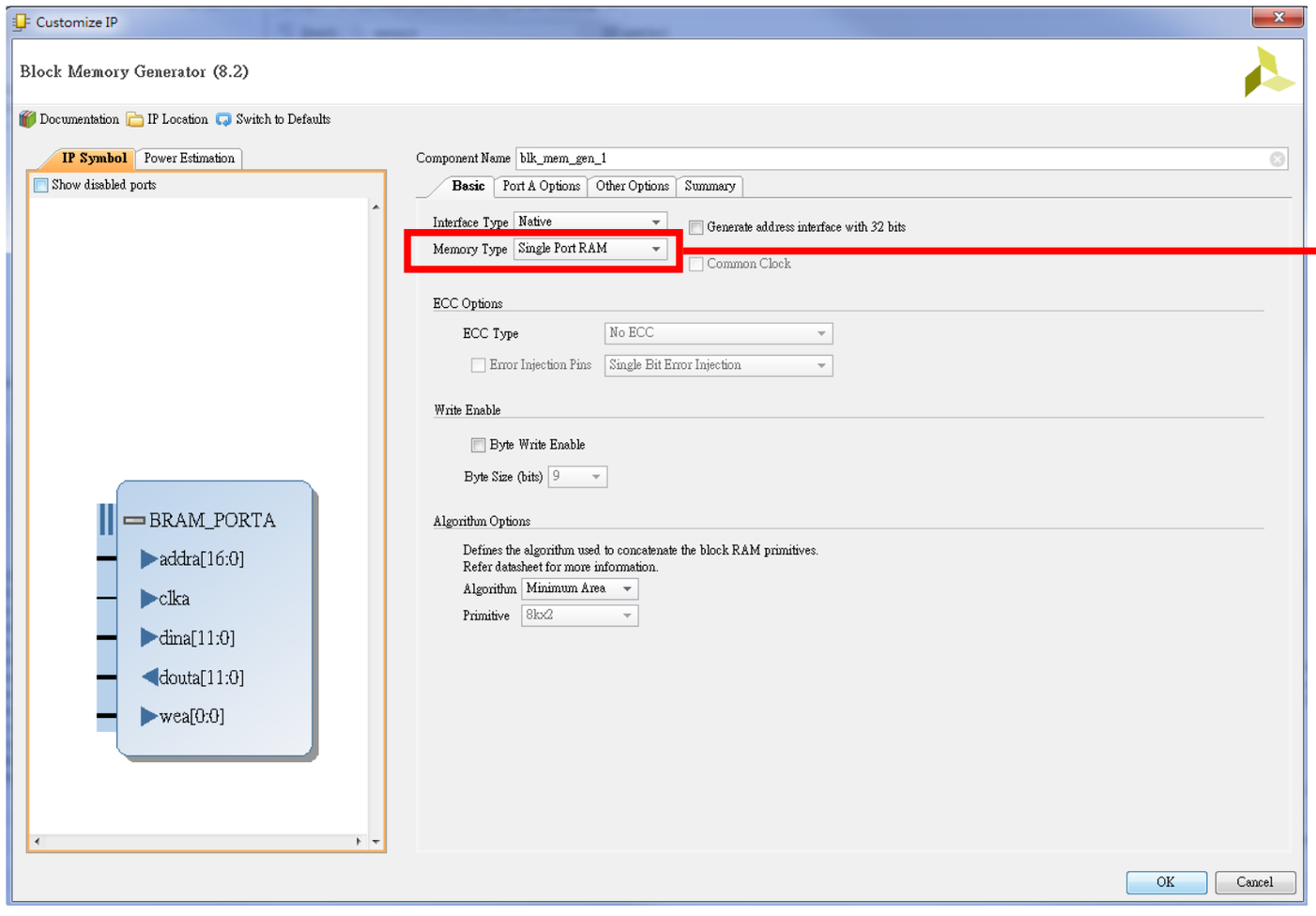


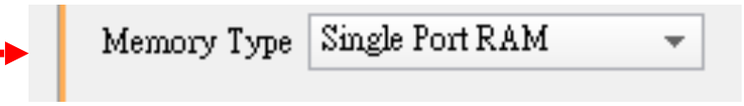
2.

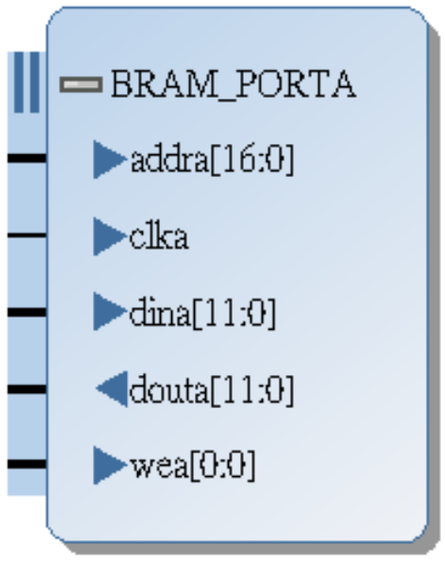
3.



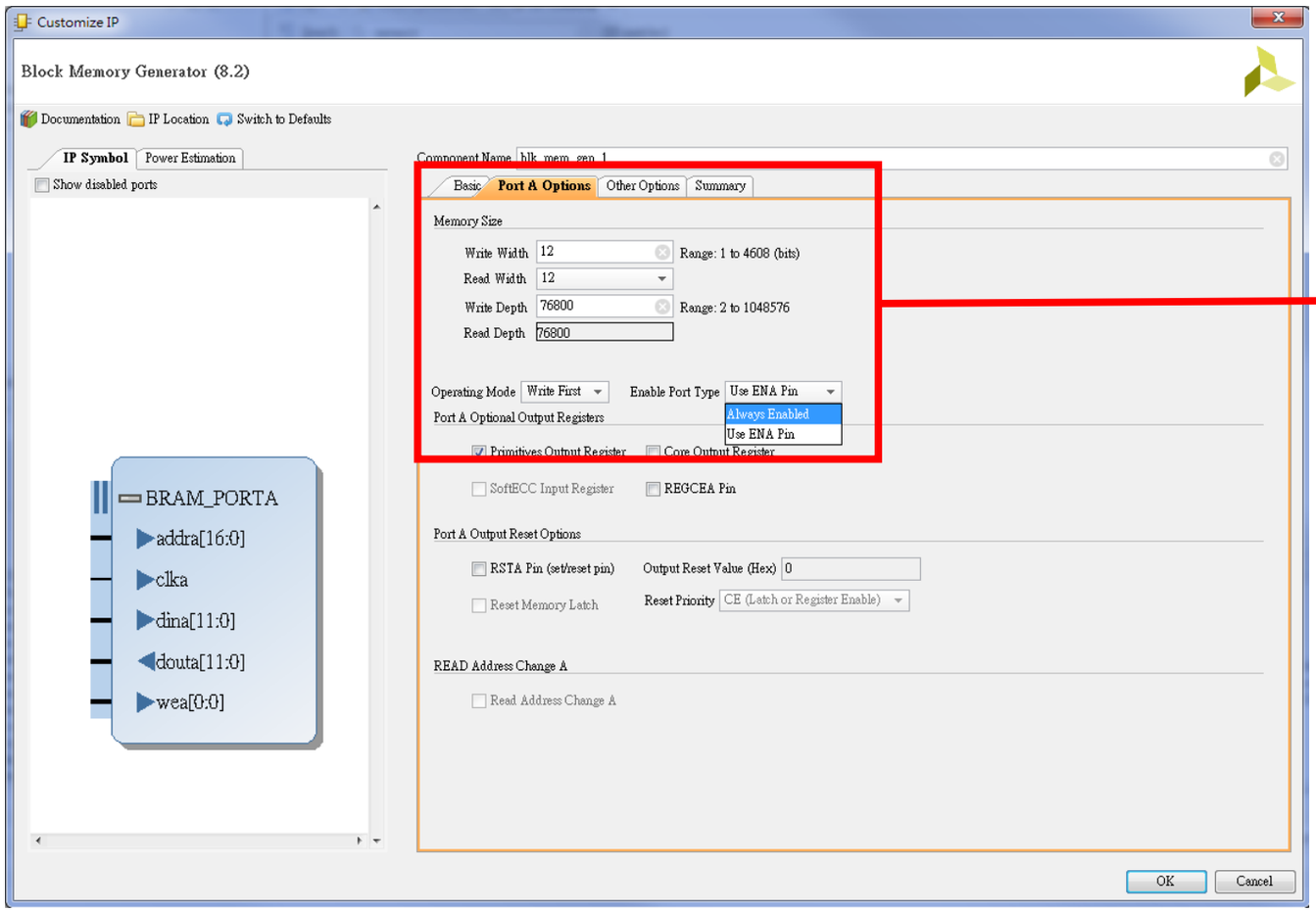
Memory IP (2/5)

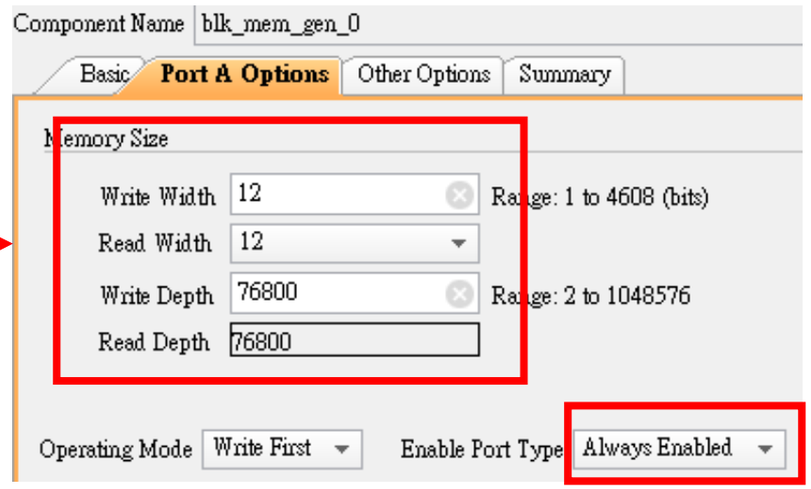

Customize IP
Block Memory Generator (8.2)
Documentation
IP Location
Switch to Defaults
IP Symbol
Power Estimation
Show disabled ports
Component Name
blk_mem_gen_1
Basic
Port A Options
Other Options
Summary
Interface Type
Native
Generate address interface with 32 bits
Memory Type
Single Port RAM
Common Clock
ECC Options
ECC Type
No ECC
Error Injection Pins
Single Bit Error Injection
Write Enable
Byte Write Enable
Byte Size (bits)
9
Algorithm Options
Defines the algorithm used to concatenate the block RAM primitives.
Refer datasheet for more information.
Algorithm
Minimum Area
Primitive
8kx2
OK
Cancel
BRAM_PORTA
addra[16:0]
clka
dina[11:0]
douta[11:0]
wea[0:0]


Memory Type
Single Port RAM


BRAM_PORTA
addra[16:0]
clka
dina[11:0]
douta[11:0]
wea[0:0]

Memory IP (3/5)

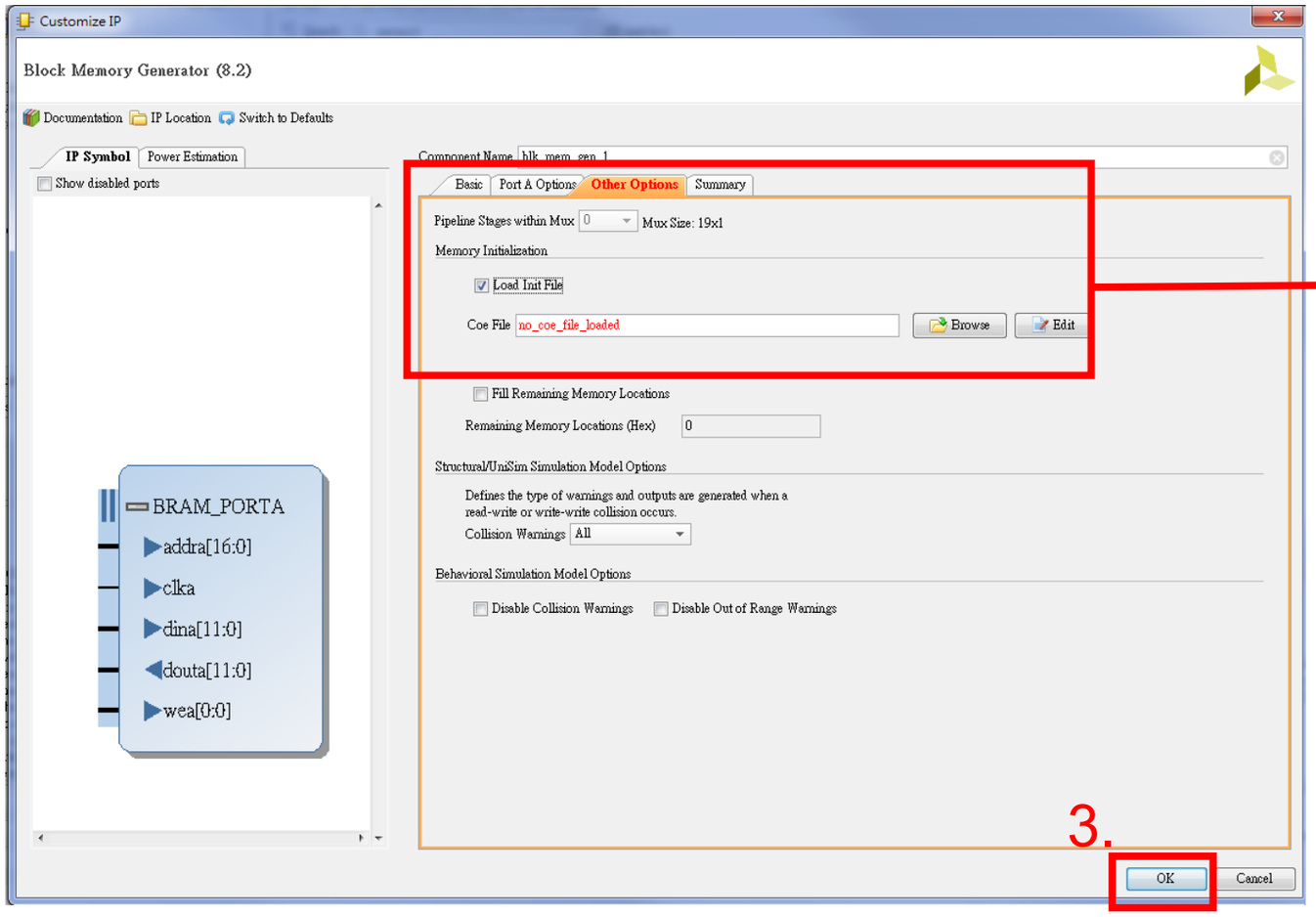


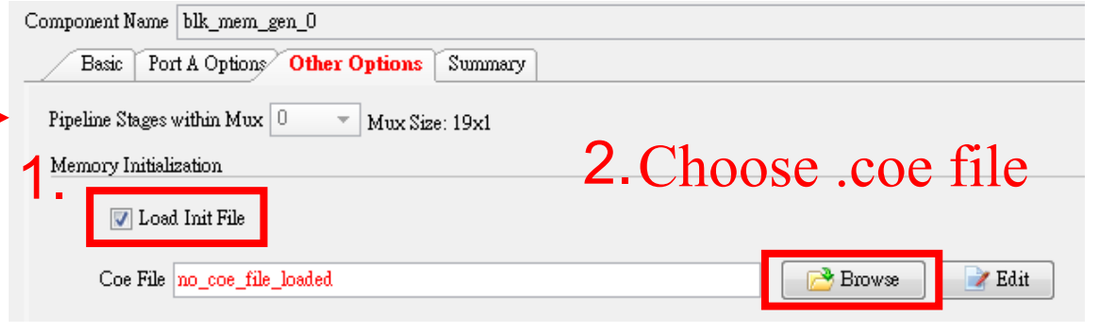


RGB : 12 bits
320x240 : 76800

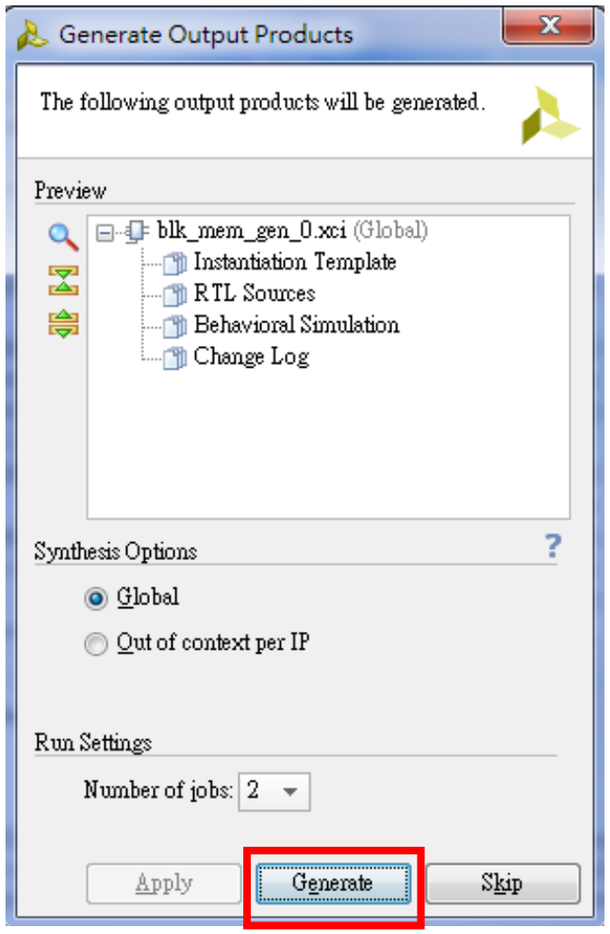
1800kbits of fast block RAM in FPGA

Memory IP (4/5)


Customize IP
Block Memory Generator (8.2)
Documentation
IP Location
Switch to Defaults
IP Symbol
Power Estimation
Show disabled ports
Component Name blk_mem_gen_1
Basic
Port A Options
Other Options
Summary
Pipeline Stages within Mux 0
Mux Size: 19x1
Memory Initialization
Load Init File
Coe File no_coe_file_loaded
Browse
Edit
Fill Remaining Memory Locations
Remaining Memory Locations (Hex) 0
Structural/UniSim Simulation Model Options
Defines the type of warnings and outputs are generated when a read-write or write-write collision occurs.
Collision Warnings All
Behavioral Simulation Model Options
Disable Collision Warnings
Disable Out of Range Warnings
BRAM_PORTA
addra[16:0]
clka
dina[11:0]
douta[11:0]
wea[0:0]
3.
OK
Cancel


Component Name blk_mem_gen_0
Basic
Port A Options
Other Options
Summary
Pipeline Stages within Mux 0
Mux Size: 19x1
1.
Memory Initialization
Load Init File
2. Choose .coe file
Coe File no_coe_file_loaded
Browse
Edit

Memory IP (5/5)


Generate Output Products
The following output products will be generated.
Preview
blk_mem_gen_0.xci (Global)
Instantiation Template
RTL Sources
Behavioral Simulation
Change Log
Synthesis Options
Global
Out of context per IP
Run Settings
Number of jobs: 2
Apply
Generate
Skip

Picture Format Translation (1/2)

amumu.jpg

PicTrans.exe

out.coe

out.coe

```
1  memory_initialization_radix=16;
2  memory_initialization_vector=
3  115,
4  115,
5  115,
6  115,
7  115,
8  115,
9  115,
10 115,
11 115,
12 115,
   ⋮
76797 743,
76798 743,
76799 743,
76800 743,
76801 743,
76802 743;
76803
```

Picture Format Translation (2/2)

- PicTrans.exe:
  - Convert a *.jpg file to a bit map file
- Input:
  - image (*.jpg)
  - the width of the output file
  - the height of the output file
- Output:
  - out.coe

Lab 11: VGA Display

Action Item (1/2)

Modify the Verilog code introduced in class to design a circuit for controlling the VGA display.

- input ports:

```
input  clk;
input  reset;
input  en;
```

- output ports:

```
output  [3:0]vgaRed;
output  [3:0]vgaGreen;
output  [3:0]vgaBlue;
output  hsync;
output  vsync;
```

Action Item (2/2)

- At the beginning or when pressing the **reset** button, the VGA display will show the image (e.g., amumu.jpg) at the origin position. It will stay still until the **en** button is pressed.
- The image will start/resume scrolling down row by row under the frequency of clk divided by 2^{22} (i.e., clk/2^{22}), or pause, depending on whether the number of the **en** button pressed is odd or even.

Example (1/6)

at the beginning or pressing **reset**

(0 row scrolled down)

Example (2/6)

press **en** → start to scroll down

(100 rows scrolled down)

Example (3/6)

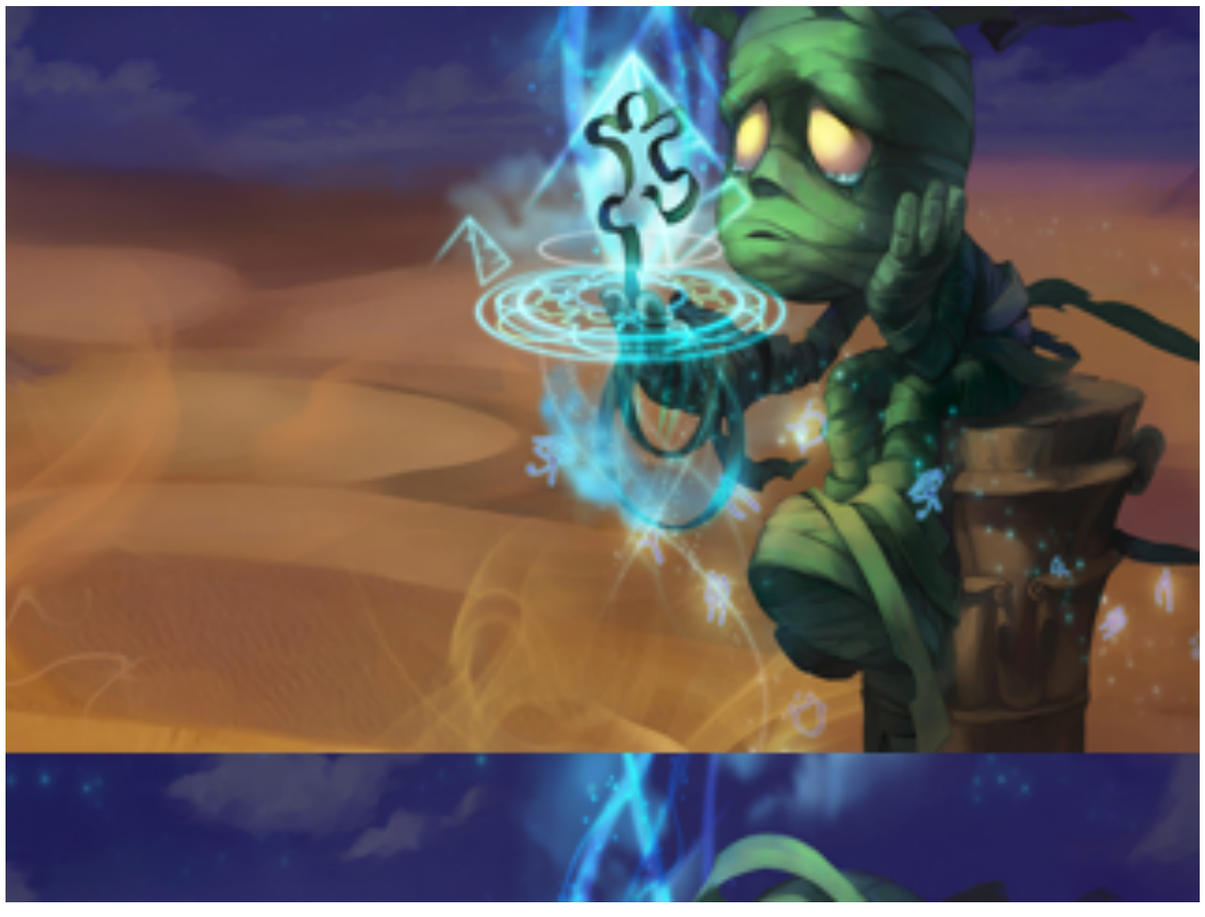
(200 rows scrolled down)

Example (4/6)

(300 rows scrolled down)

Example (5/6)

(400 rows scrolled down)



Example (6/6)

press **en** → pause

(400 rows scrolled down)