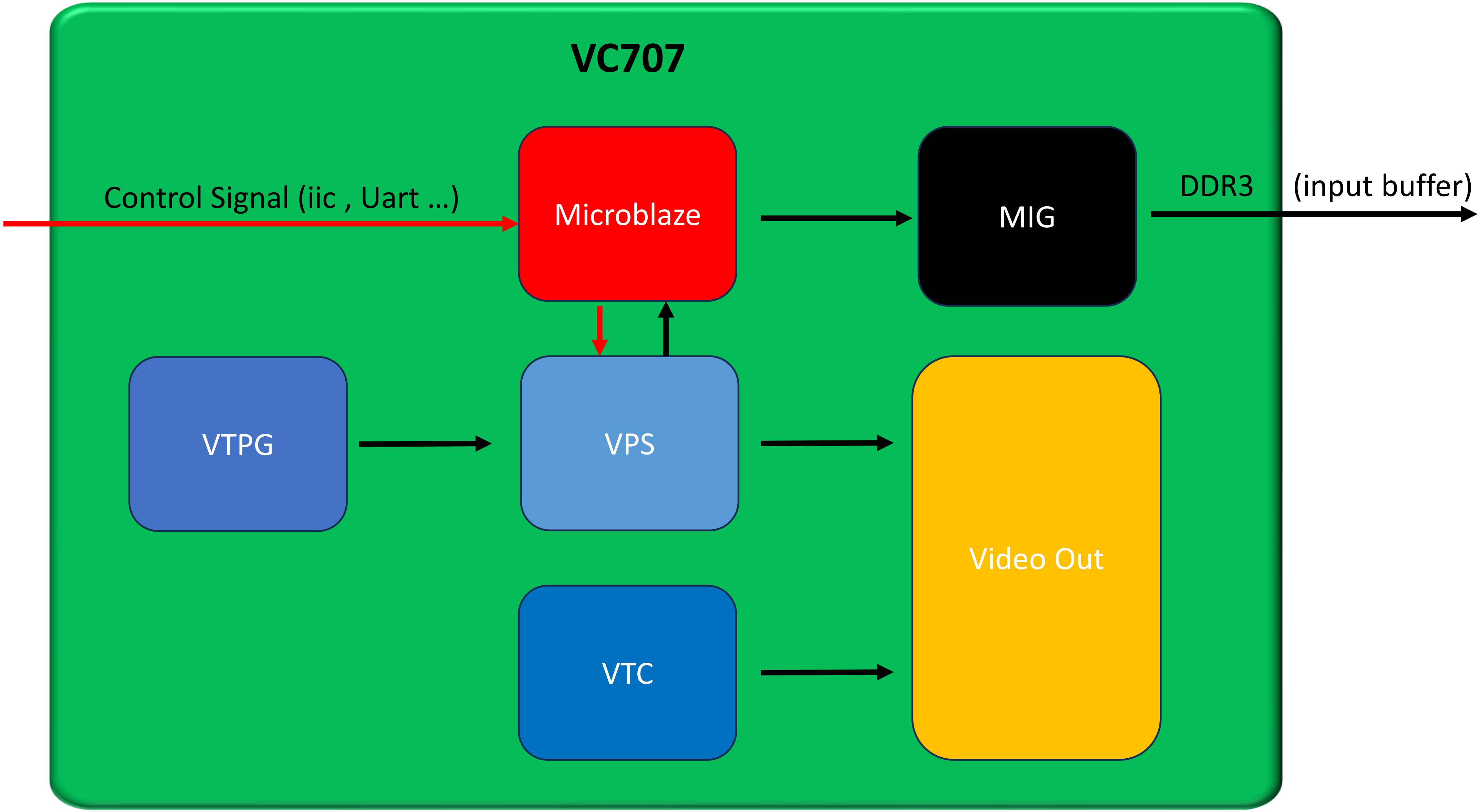
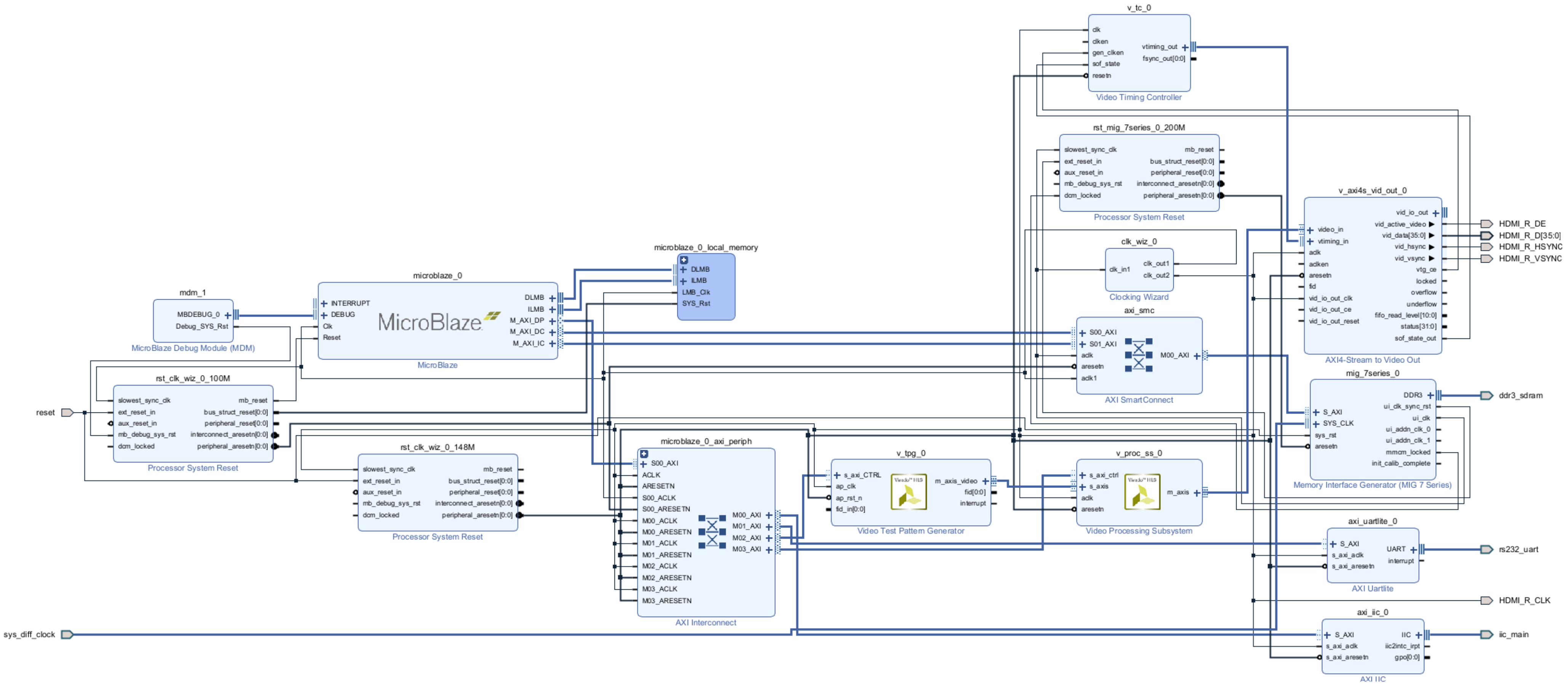
VC707 HDMI Output Example

XILINX

HDMI_output Block Diagram

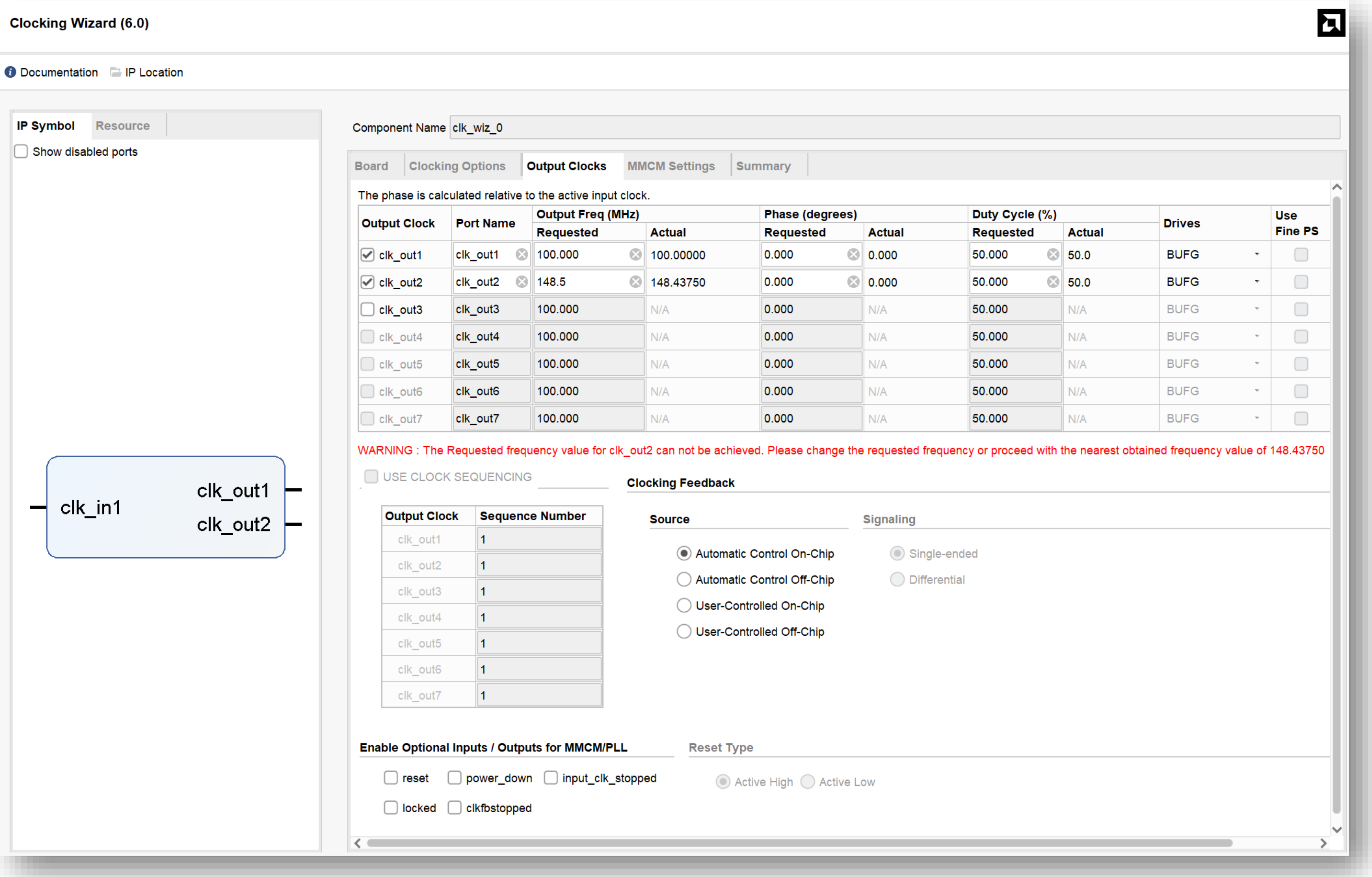


Block Design



Clocking Wizard Setting

輸出100MHz 給Microblaze , 輸出148.5MHz 給影像解析度使用



Clocking Wizard (6.0)

□ Show disabled ports

Component Name clk_wiz_0

The phase is calculated relative to the active input clock.

Output Clock	Port Name	Output Freq (MHz)		Phase (degrees)		Duty Cycle (%)		Drives	Use Fine PS
		Requested	Actual	Requested	Actual	Requested	Actual		
☑ clk_out1	clk_out1	100.000	100.00000	0.000	0.000	50.000	50.0	BUFG	□
☑ clk_out2	clk_out2	148.5	148.43750	0.000	0.000	50.000	50.0	BUFG	□
□ clk_out3	clk_out3	100.000	N/A	0.000	N/A	50.000	N/A	BUFG	□
□ clk_out4	clk_out4	100.000	N/A	0.000	N/A	50.000	N/A	BUFG	□
□ clk_out5	clk_out5	100.000	N/A	0.000	N/A	50.000	N/A	BUFG	□
□ clk_out6	clk_out6	100.000	N/A	0.000	N/A	50.000	N/A	BUFG	□
□ clk_out7	clk_out7	100.000	N/A	0.000	N/A	50.000	N/A	BUFG	□

WARNING : The Requested frequency value for clk_out2 can not be achieved. Please change the requested frequency or proceed with the nearest obtained frequency value of 148.43750

□ USE CLOCK SEQUENCING

Output Clock	Sequence Number
clk_out1	1
clk_out2	1
clk_out3	1
clk_out4	1
clk_out5	1
clk_out6	1
clk_out7	1

Clocking Feedback

Source

● Automatic Control On-Chip
○ Automatic Control Off-Chip
○ User-Controlled On-Chip
○ User-Controlled Off-Chip

Signaling

● Single-ended
○ Differential

Enable Optional Inputs / Outputs for MMCM/PLL

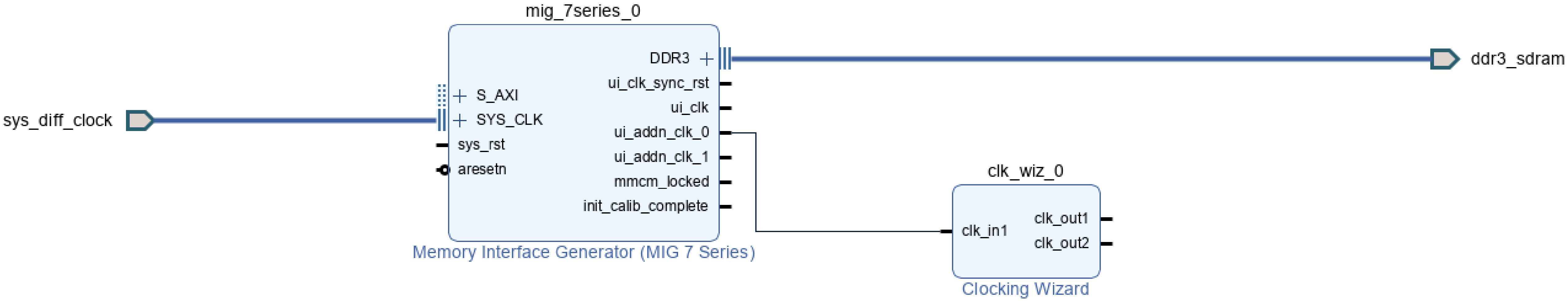
□ reset □ power_down □ input_clk_stopped
□ locked □ clkfbstopped

Reset Type

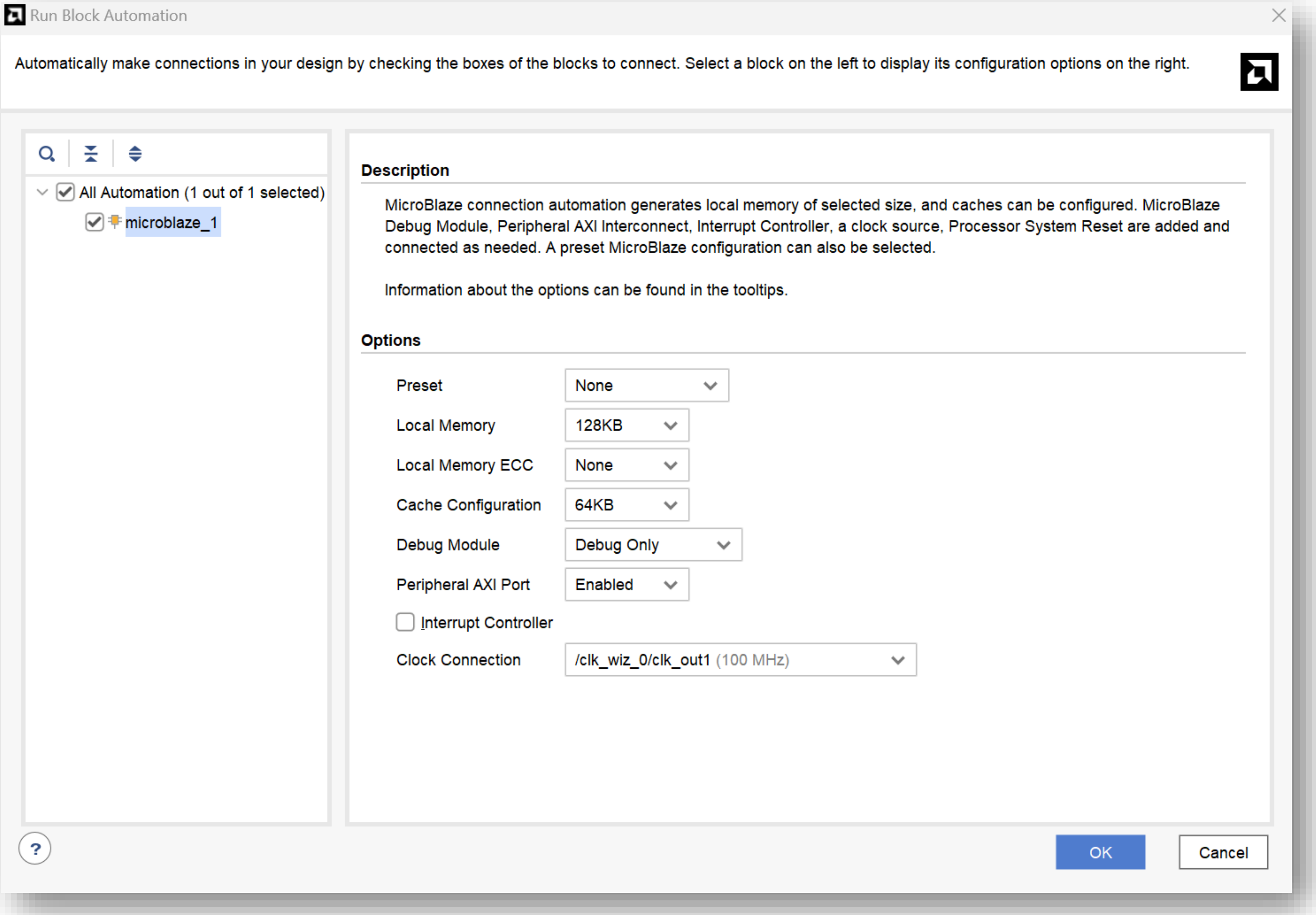
● Active High ○ Active Low

Clocking Wizard Setting

輸出100MHz 給Microblaze , 輸出148.5MHz 給影像解析度使用



Microblaze Setting



Automatically make connections in your design by checking the boxes of the blocks to connect. Select a block on the left to display its configuration options on the right.

☑ All Automation (1 out of 1 selected)

☑ microblaze_1

Description

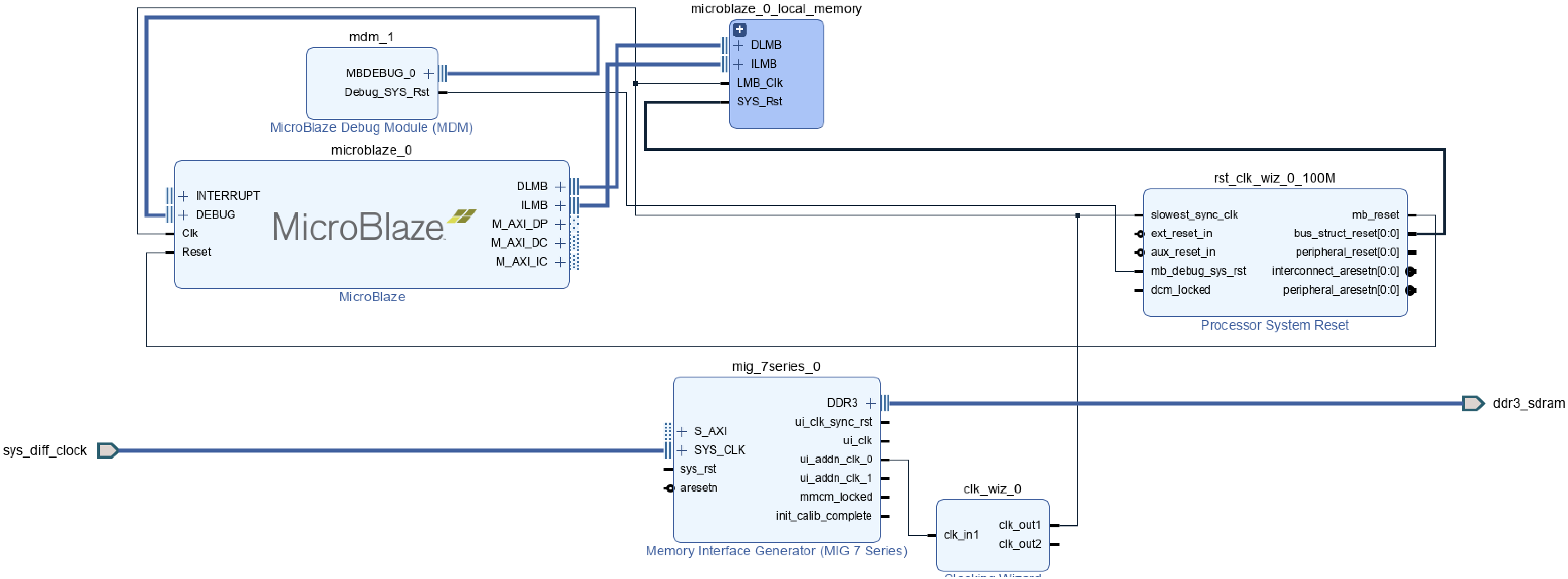
MicroBlaze connection automation generates local memory of selected size, and caches can be configured. MicroBlaze Debug Module, Peripheral AXI Interconnect, Interrupt Controller, a clock source, Processor System Reset are added and connected as needed. A preset MicroBlaze configuration can also be selected.

Information about the options can be found in the tooltips.

Options

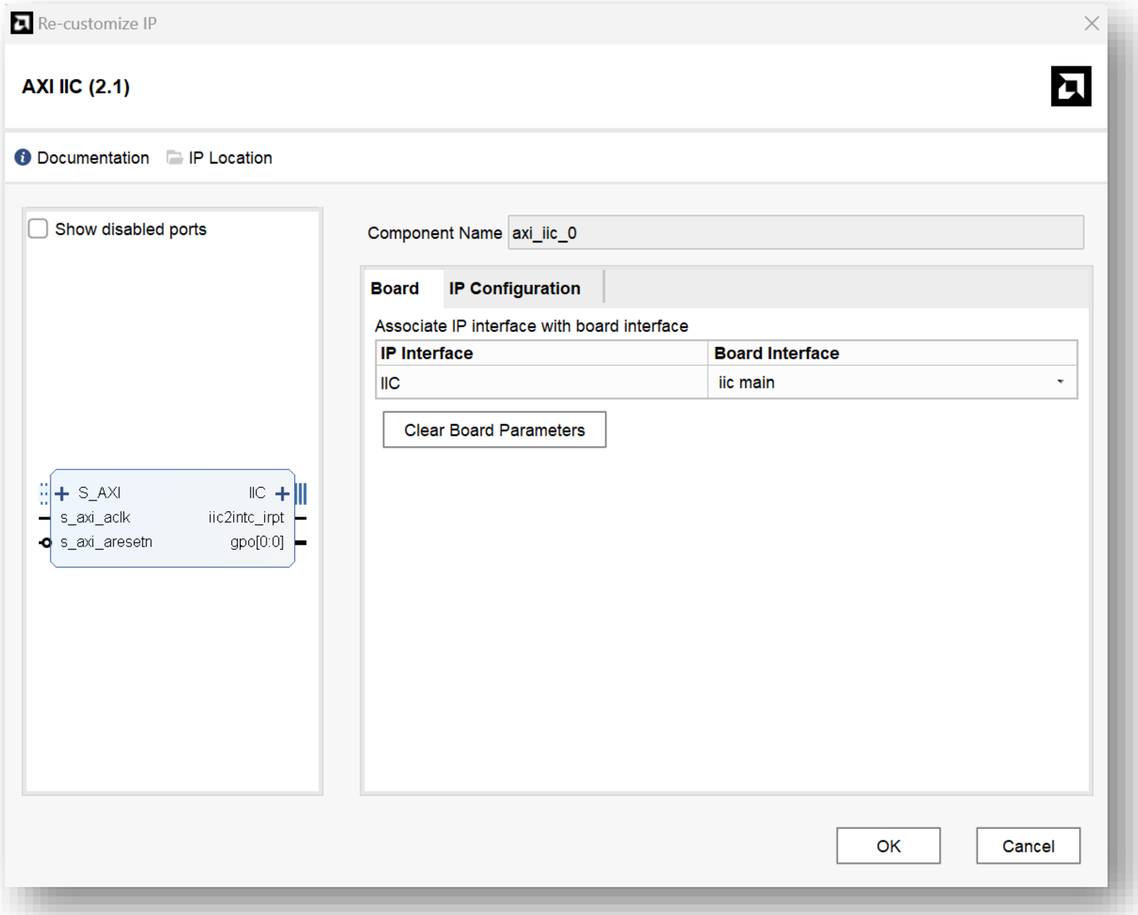
Preset	None
Local Memory	128KB
Local Memory ECC	None
Cache Configuration	64KB
Debug Module	Debug Only
Peripheral AXI Port	Enabled
□ Interrupt Controller	
Clock Connection	/clk_wiz_0/clk_out1 (100 MHz)

Microblaze Setting



Iic & uart Setting

Iic與uart直接使用開發版上的配置

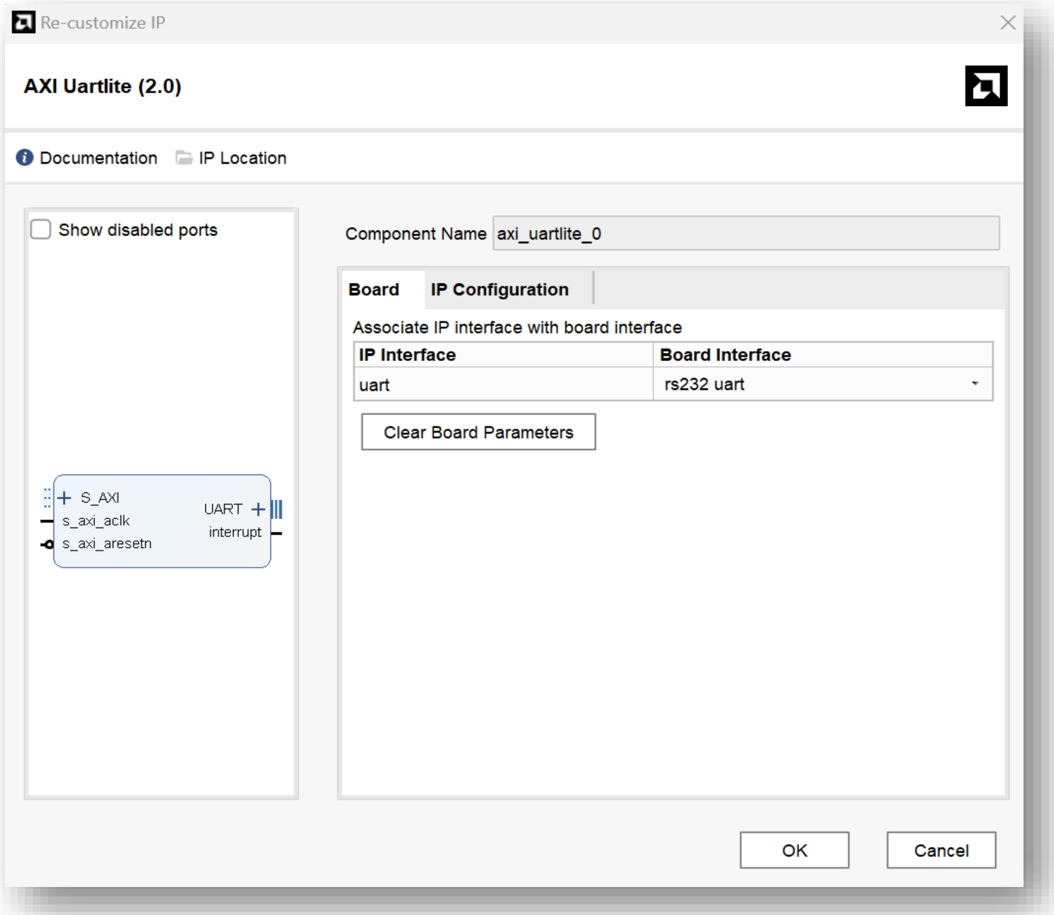

Re-customize IP

AXI IIC (2.1)

Component Name axi_iic_0

Associate IP interface with board interface

IP Interface	Board Interface
IIC	iic main


Re-customize IP

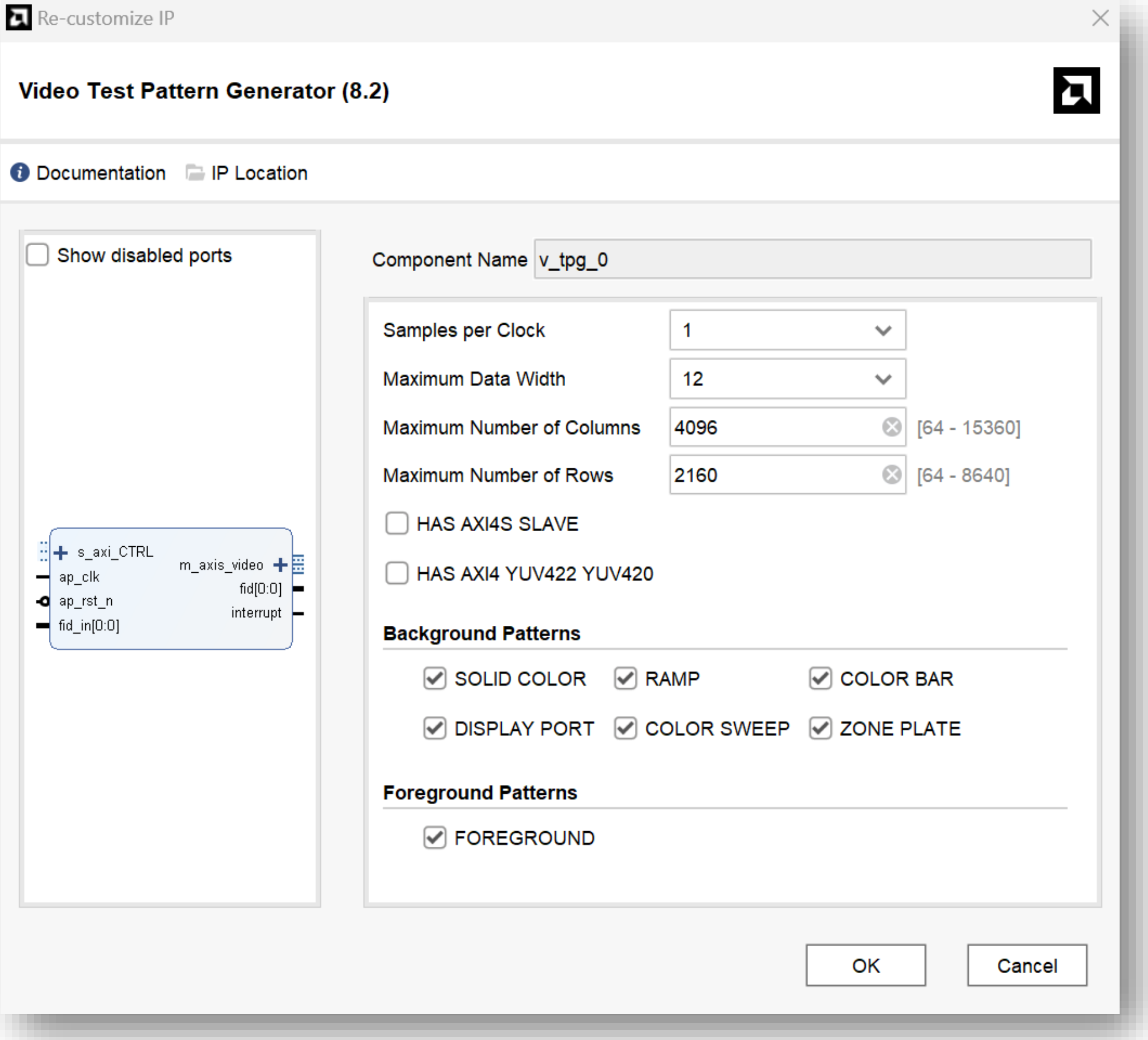
AXI Uartlite (2.0)

Component Name axi_uartlite_0

Associate IP interface with board interface

IP Interface	Board Interface
uart	rs232 uart

VTPG Setting



Re-customize IP

Video Test Pattern Generator (8.2)

Documentation   IP Location

☐ Show disabled ports

Component Name v_tpg_0

Samples per Clock: 1

Maximum Data Width: 12

Maximum Number of Columns: 4096 [64 - 15360]

Maximum Number of Rows: 2160 [64 - 8640]

☐ HAS AXI4S SLAVE

☐ HAS AXI4 YUV422 YUV420

Background Patterns

☑ SOLID COLOR ☑ RAMP ☑ COLOR BAR

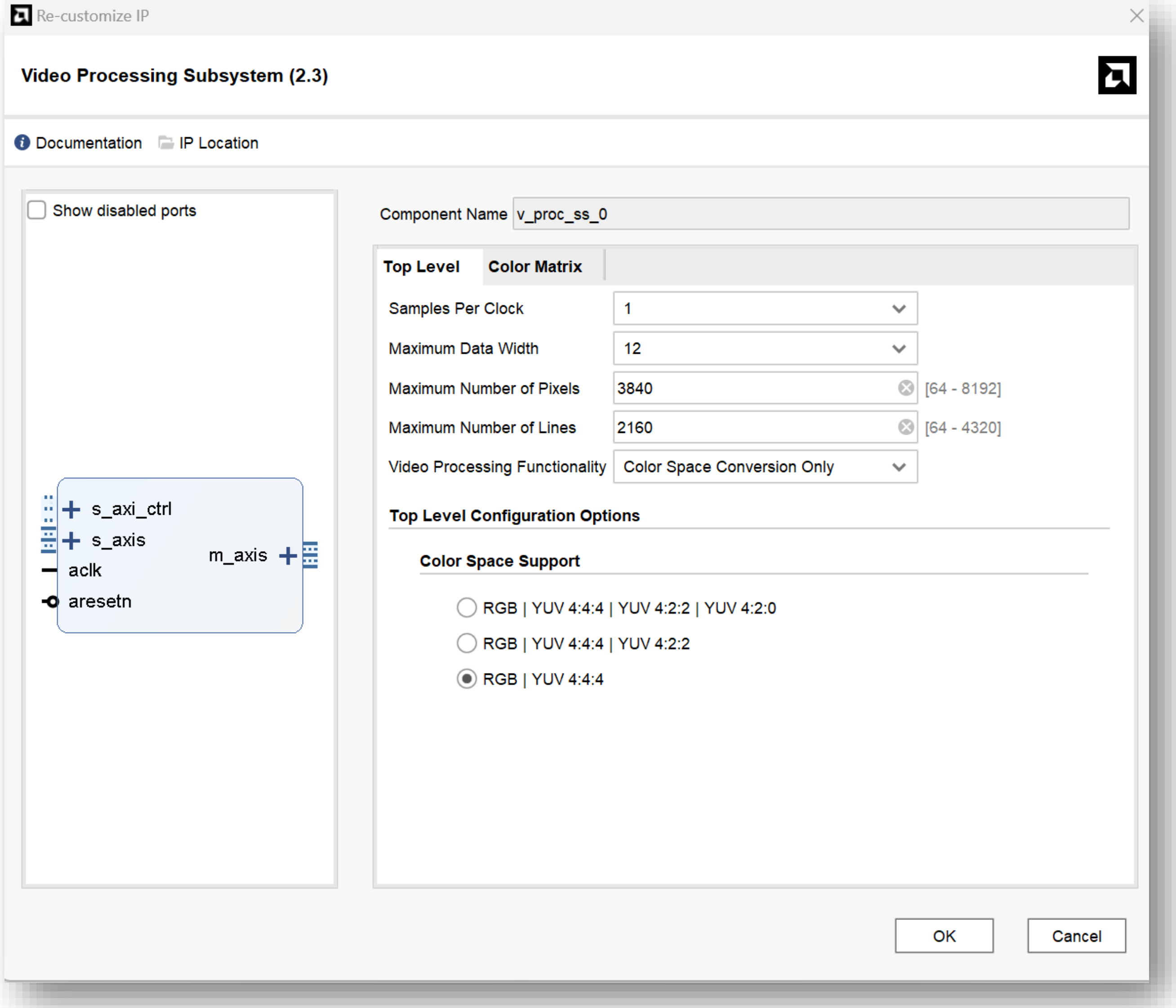
☑ DISPLAY PORT ☑ COLOR SWEEP ☑ ZONE PLATE

Foreground Patterns

☑ FOREGROUND

Video Processing Setting

設定影像格式 R.G.B 4:4:4



Video Processing Subsystem (2.3)

□ Show disabled ports

Component Name v_proc_ss_0

Samples Per Clock	1	
Maximum Data Width	12	
Maximum Number of Pixels	3840	[64 - 8192]
Maximum Number of Lines	2160	[64 - 4320]
Video Processing Functionality	Color Space Conversion Only	

Top Level Configuration Options

Color Space Support

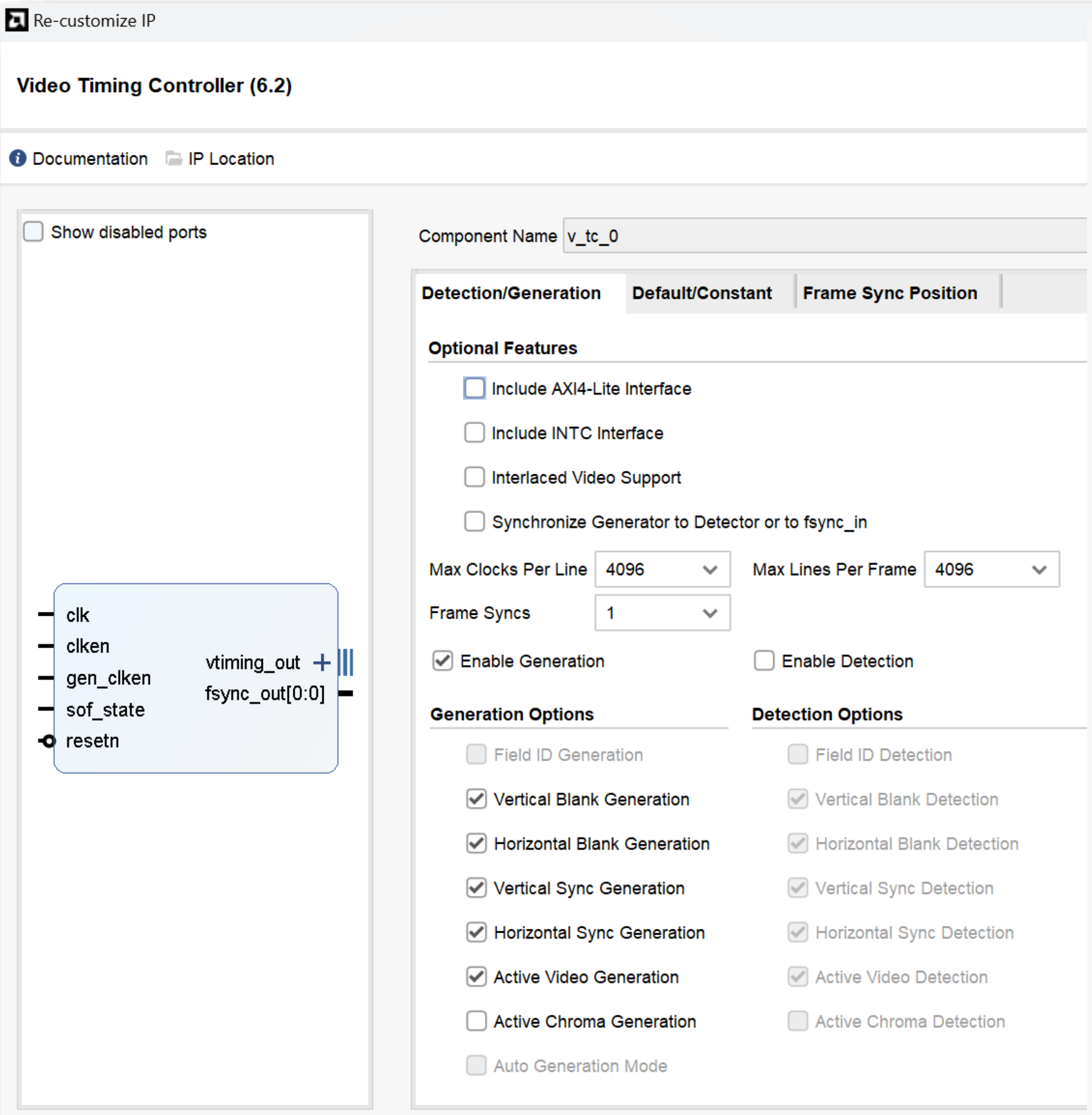
○ RGB | YUV 4:4:4 | YUV 4:2:2 | YUV 4:2:0

○ RGB | YUV 4:4:4 | YUV 4:2:2

● RGB | YUV 4:4:4

Video Timing Controller Setting

設定解析度時序 1080p



Re-customize IP

Video Timing Controller (6.2)

Documentation IP Location

□ Show disabled ports

Component Name v_tc_0

Detection/Generation | Default/Constant | Frame Sync Position

Optional Features

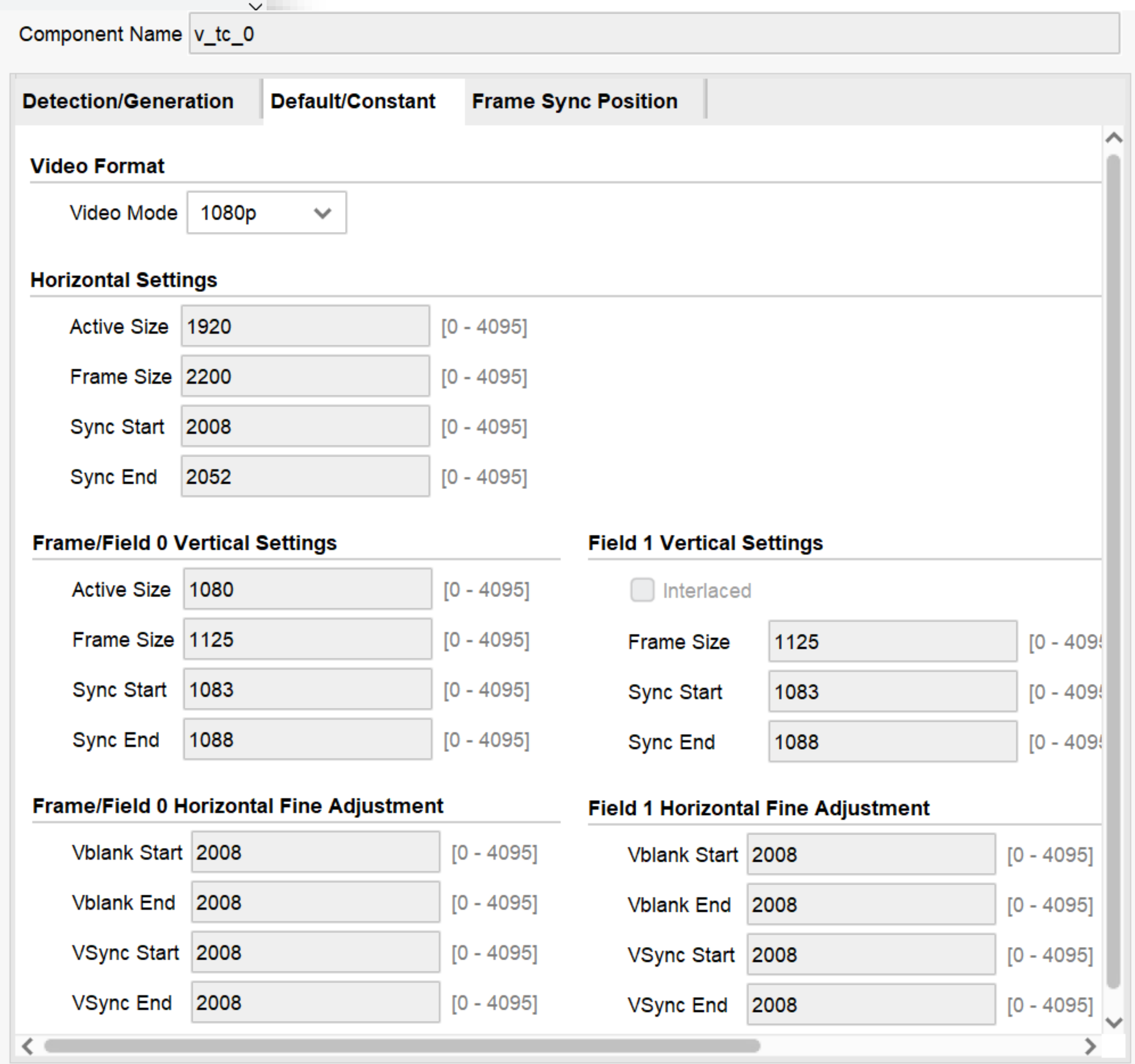
- □ Include AXI4-Lite Interface
- □ Include INTC Interface
- □ Interlaced Video Support
- □ Synchronize Generator to Detector or to fsync_in

Max Clocks Per Line: 4096 Max Lines Per Frame: 4096

Frame Syncs: 1

☑ Enable Generation □ Enable Detection

Generation Options	Detection Options
□ Field ID Generation	□ Field ID Detection
☑ Vertical Blank Generation	☑ Vertical Blank Detection
☑ Horizontal Blank Generation	☑ Horizontal Blank Detection
☑ Vertical Sync Generation	☑ Vertical Sync Detection
☑ Horizontal Sync Generation	☑ Horizontal Sync Detection
☑ Active Video Generation	☑ Active Video Detection
□ Active Chroma Generation	□ Active Chroma Detection
□ Auto Generation Mode	



Component Name v_tc_0

Detection/Generation | Default/Constant | Frame Sync Position

Video Format

Video Mode: 1080p

Horizontal Settings

Active Size	1920	[0 - 4095]
Frame Size	2200	[0 - 4095]
Sync Start	2008	[0 - 4095]
Sync End	2052	[0 - 4095]

Frame/Field 0 Vertical Settings

Active Size	1080	[0 - 4095]
Frame Size	1125	[0 - 4095]
Sync Start	1083	[0 - 4095]
Sync End	1088	[0 - 4095]

Field 1 Vertical Settings

□ Interlaced

Frame Size	1125	[0 - 409
Sync Start	1083	[0 - 409
Sync End	1088	[0 - 409

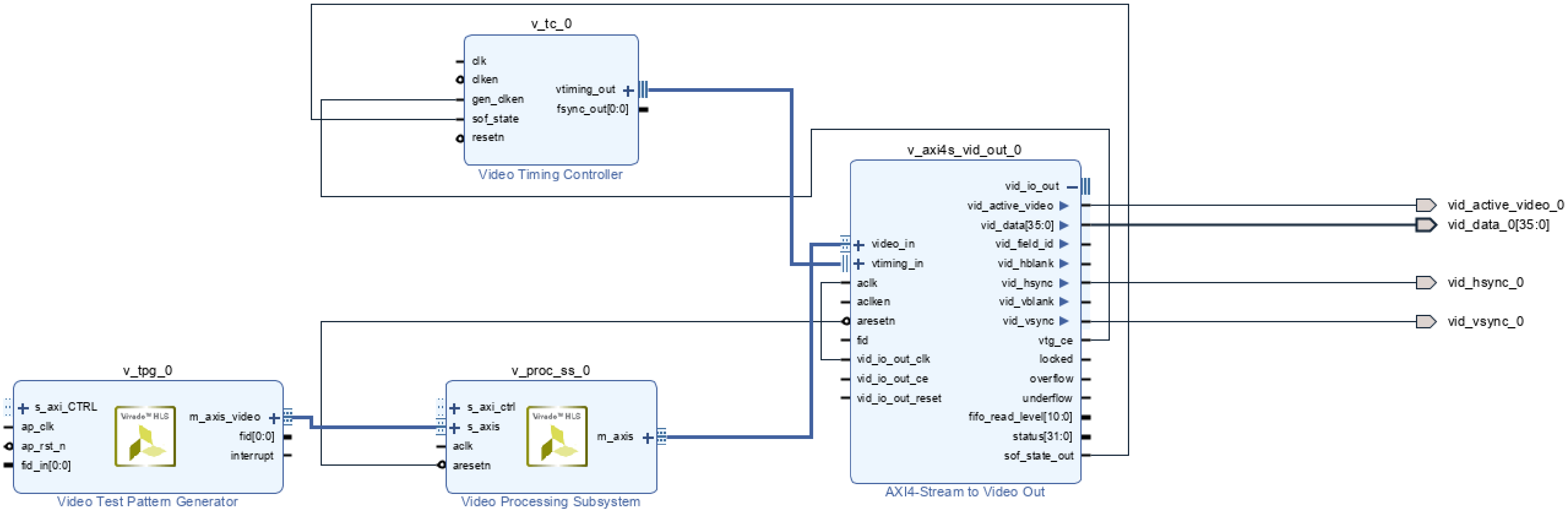
Frame/Field 0 Horizontal Fine Adjustment

Vblank Start	2008	[0 - 4095]
Vblank End	2008	[0 - 4095]
VSync Start	2008	[0 - 4095]
VSync End	2008	[0 - 4095]

Field 1 Horizontal Fine Adjustment

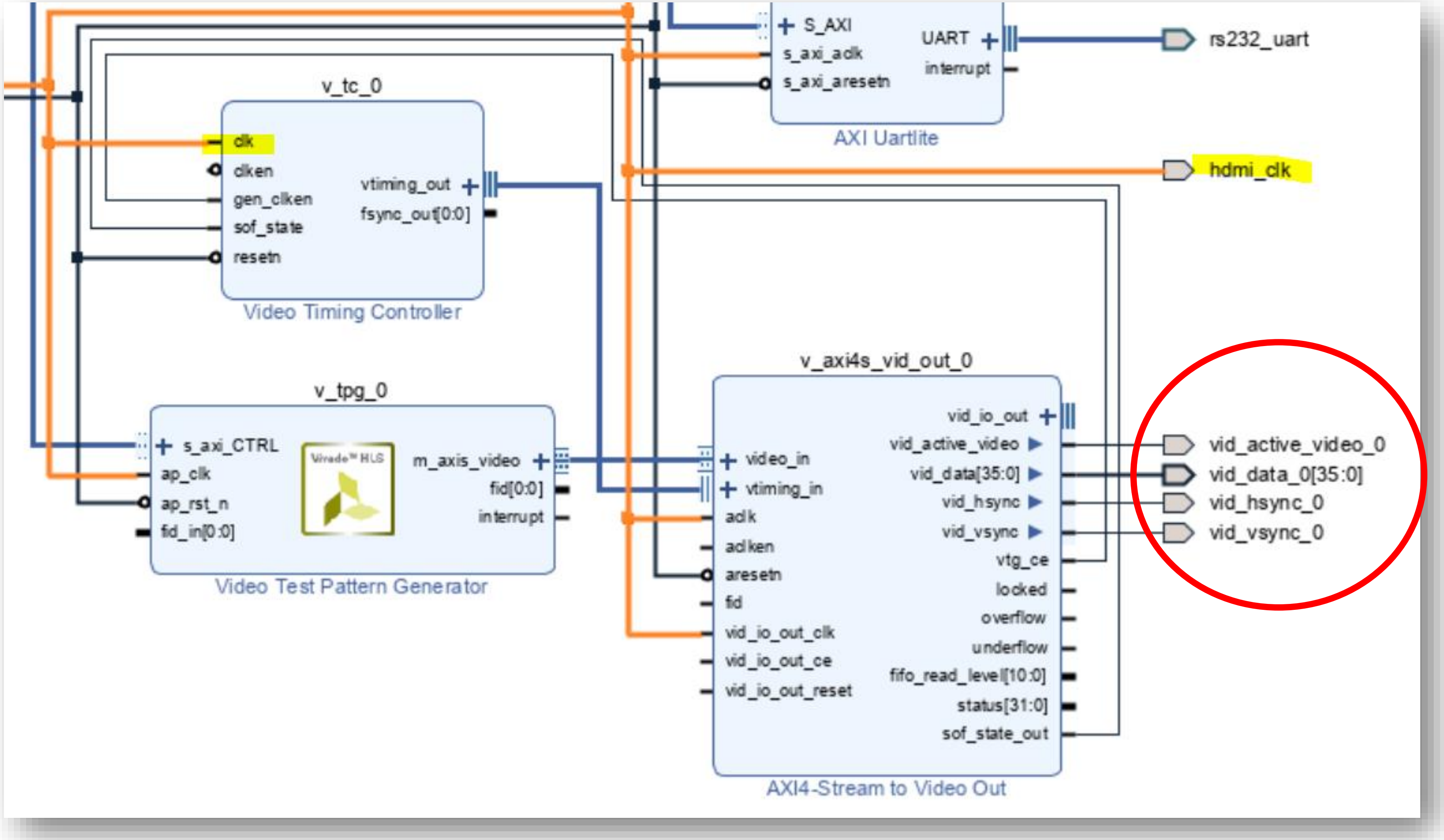
Vblank Start	2008	[0 - 4095]
Vblank End	2008	[0 - 4095]
VSync Start	2008	[0 - 4095]
VSync End	2008	[0 - 4095]

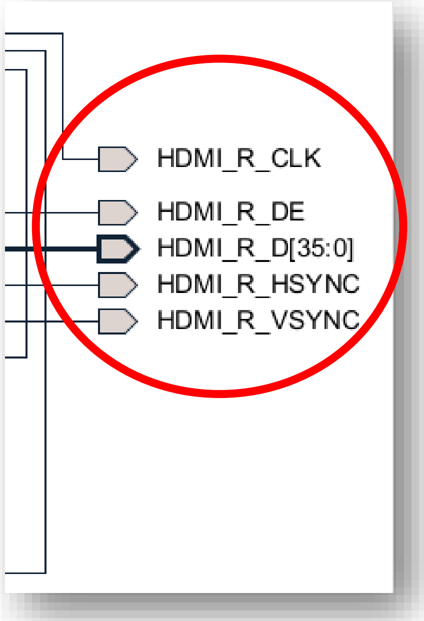
Video data connecting Setting



Video data connecting Setting

輸出一個hdmi_clk(148.5MHz)





IO Constraints

```
set_property PACKAGE_PIN AM22 [get_ports HDMI_R_D[0]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[0]]
set_property PACKAGE_PIN AL22 [get_ports HDMI_R_D[1]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[1]]
set_property PACKAGE_PIN AJ20 [get_ports HDMI_R_D[2]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[2]]
set_property PACKAGE_PIN AJ21 [get_ports HDMI_R_D[3]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[3]]
set_property PACKAGE_PIN AM21 [get_ports HDMI_R_D[4]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[4]]
set_property PACKAGE_PIN AL21 [get_ports HDMI_R_D[5]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[5]]
set_property PACKAGE_PIN AK22 [get_ports HDMI_R_D[6]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[6]]
set_property PACKAGE_PIN AJ22 [get_ports HDMI_R_D[7]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[7]]
set_property PACKAGE_PIN AL20 [get_ports HDMI_R_D[8]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[8]]
set_property PACKAGE_PIN AK20 [get_ports HDMI_R_D[9]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[9]]
set_property PACKAGE_PIN AK23 [get_ports HDMI_R_D[10]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[10]]
set_property PACKAGE_PIN AJ23 [get_ports HDMI_R_D[11]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[11]]
set_property PACKAGE_PIN AN21 [get_ports HDMI_R_D[12]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[12]]
set_property PACKAGE_PIN AP22 [get_ports HDMI_R_D[13]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[13]]
set_property PACKAGE_PIN AP23 [get_ports HDMI_R_D[14]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[14]]
set_property PACKAGE_PIN AN23 [get_ports HDMI_R_D[15]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[15]]
set_property PACKAGE_PIN AM23 [get_ports HDMI_R_D[16]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[16]]
set_property PACKAGE_PIN AN24 [get_ports HDMI_R_D[17]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[17]]
set_property PACKAGE_PIN AY24 [get_ports HDMI_R_D[18]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[18]]
set_property PACKAGE_PIN BB22 [get_ports HDMI_R_D[19]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[19]]
set_property PACKAGE_PIN BA22 [get_ports HDMI_R_D[20]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[20]]
set_property PACKAGE_PIN BA25 [get_ports HDMI_R_D[21]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[21]]
set_property PACKAGE_PIN AY25 [get_ports HDMI_R_D[22]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[22]]
set_property PACKAGE_PIN AY22 [get_ports HDMI_R_D[23]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[23]]
```

```
set_property PACKAGE_PIN AY23 [get_ports HDMI_R_D[24]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[24]]
set_property PACKAGE_PIN AV24 [get_ports HDMI_R_D[25]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[25]]
set_property PACKAGE_PIN AU24 [get_ports HDMI_R_D[26]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[26]]
set_property PACKAGE_PIN AW21 [get_ports HDMI_R_D[27]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[27]]
set_property PACKAGE_PIN AV21 [get_ports HDMI_R_D[28]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[28]]
set_property PACKAGE_PIN AT24 [get_ports HDMI_R_D[29]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[29]]
set_property PACKAGE_PIN AR24 [get_ports HDMI_R_D[30]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[30]]
set_property PACKAGE_PIN AU21 [get_ports HDMI_R_D[31]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[31]]
set_property PACKAGE_PIN AT21 [get_ports HDMI_R_D[32]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[32]]
set_property PACKAGE_PIN AW22 [get_ports HDMI_R_D[33]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[33]]
set_property PACKAGE_PIN AW23 [get_ports HDMI_R_D[34]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[34]]
set_property PACKAGE_PIN AV23 [get_ports HDMI_R_D[35]]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_D[35]]


set_property PACKAGE_PIN AU23 [get_ports HDMI_R_CLK]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_CLK]
set_property PACKAGE_PIN AP21 [get_ports HDMI_R_DE]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_DE]
set_property PACKAGE_PIN AT22 [get_ports HDMI_R_VSYNC]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_VSYNC]
set_property PACKAGE_PIN AU22 [get_ports HDMI_R_HSYNC]
set_property IOSTANDARD LVCMOS18 [get_ports HDMI_R_HSYNC]
```

Microblaze Controll

Vc707_hdmi.c

```c
#include <stdio.h>
#include "platform.h"
#include "xil_printf.h"
#include "xstatus.h"
#include "sleep.h"
#include "xiic_l.h"
#include "xil_io.h"
#include "xil_types.h"
#include "xv_tpg.h"
#include "xvidc.h"
//#include "xvprocss.h"

#define PAGE_SIZE   16
#define IIC_BASE_ADDRESSXPAR_IIC_0_BASEADDR
#define EEPROM_TEST_START_ADDRESS0x80
#define IIC_SWITCH_ADDRESS 0x74
#define IIC_ADV7511_ADDRESS 0x39

XV_tpg tpg;
XV_tpg_Config *tpg_config;
//XVprocSs scaler_new_inst;
//XVprocSs csc_new_inst;

typedef u8 AddressType;

typedef struct {
          u8 addr;
          u8 data;
          u8 init;
} HDMI_REG;
#define NUMBER_OF_HDMI_REGS  40
```

Vc707_hdmi.c

```c
HDMI_REG hdmi_iic[NUMBER_OF_HDMI_REGS] = {

{ 0x15, 0x00, 0x00 },
{ 0x16, 0x00, 0x20 },
{ 0x41, 0x00, 0x10 },
{ 0x48, 0x00, 0x08 },
{ 0x55, 0x00, 0x00 },
{ 0x56, 0x00, 0x28 },
{ 0x98, 0x00, 0x03 },
{ 0x9A, 0x00, 0xE0 },
{ 0x9C, 0x00, 0x30 },
{ 0x9D, 0x00, 0x61 },
{ 0xA2, 0x00, 0xA4 },
{ 0xA3, 0x00, 0xA4 },
{ 0xAF, 0x00, 0x06 },
{ 0xBA, 0x00, 0x60 },
{ 0xE0, 0x00, 0xD0 },
{ 0xF9, 0x00, 0x00 },
// HDTV YCbCr (16to235) to RGB (16to235)
//color space conversion
/**/{ 0x18, 0x00, 0xAC },
/**/{ 0x19, 0x00, 0x53 },
/**/{ 0x1A, 0x00, 0x08 },
/**/{ 0x1B, 0x00, 0x00 },
/**/{ 0x1C, 0x00, 0x00 },
/**/{ 0x1D, 0x00, 0x00 },
/**/{ 0x1E, 0x00, 0x19 },
/**/{ 0x1F, 0x00, 0xD6 },
/**/{ 0x20, 0x00, 0x1C },
/**/{ 0x21, 0x00, 0x56 },
/**/{ 0x22, 0x00, 0x08 },
/**/{ 0x23, 0x00, 0x00 },
/**/{ 0x24, 0x00, 0x1E },
/**/{ 0x25, 0x00, 0x88 },
/**/{ 0x26, 0x00, 0x02 },
/**/{ 0x27, 0x00, 0x91 },
/**/{ 0x28, 0x00, 0x1F },
/**/{ 0x29, 0x00, 0xFF },
/**/{ 0x2A, 0x00, 0x08 },
/**/{ 0x2B, 0x00, 0x00 },
/**/{ 0x2C, 0x00, 0x0E },
/**/{ 0x2D, 0x00, 0x85 },
/**/{ 0x2E, 0x00, 0x18 },
/**/{ 0x2F, 0x00, 0xBE }

};
```

Vc707_hdmi.c

```
u8 EepromIicAddr;/* Variable for storing Eeprom IIC address */
int IicLowLevelDynEeprom();
u8 EepromReadByte(AddressType Address, u8 *BufferPtr, u8 ByteCount);
u8 EepromWriteByte(AddressType Address, u8 *BufferPtr, u8 ByteCount);

//HDMI IIC
int IicLowLevelDynEeprom()
{
  u8 BytesRead;
  u32 StatusReg;
  u8 Index;
  int Status;
  u32 i;
  u8 channel[1] = {0x20};

  Status = XIic_DynInit(IIC_BASE_ADDRESS);
  if (Status != XST_SUCCESS) {return XST_FAILURE;}
  xil_printf("\r\nAfter XIic_DynInit\r\n");
  while (((StatusReg = XIic_ReadReg(IIC_BASE_ADDRESS,
          XIIC_SR_REG_OFFSET)) &
          (XIIC_SR_RX_FIFO_EMPTY_MASK |
          XIIC_SR_TX_FIFO_EMPTY_MASK |
          XIIC_SR_BUS_BUSY_MASK)) !=
          (XIIC_SR_RX_FIFO_EMPTY_MASK |
          XIIC_SR_TX_FIFO_EMPTY_MASK)) {
}

  EepromIicAddr = IIC_SWITCH_ADDRESS;
  XIic_DynSend(IIC_BASE_ADDRESS, EepromIicAddr,
      channel, sizeof(channel), XIIC_STOP);
```

Vc707_hdmi.c

```
EepromIicAddr = IIC_ADV7511_ADDRESS;
  for ( Index = 0; Index < NUMBER_OF_HDMI_REGS; Index++)
  {
    EepromWriteByte(hdmi_iic[Index].addr, &hdmi_iic[Index].init, 1);
  }

  for ( Index = 0; Index < NUMBER_OF_HDMI_REGS; Index++)
  {
    BytesRead = EepromReadByte(hdmi_iic[Index].addr, &hdmi_iic[Index].data, 1);
    for(i=0;i<1000;i++) {};// IIC delay
    if (BytesRead != 1) {return XST_FAILURE;}}
  return XST_SUCCESS;
}
```

Vc707_hdmi.c

```
u8 EepromReadByte(AddressType Address, u8 *BufferPtr, u8 ByteCount)
{
  u8 ReceivedByteCount;
  u8 SentByteCount;
  u16 StatusReg;

  /*
   * Position the Read pointer to specific location in the EEPROM.
   */
  do {StatusReg = XIic_ReadReg(IIC_BASE_ADDRESS, XIIC_SR_REG_OFFSET);
    if (!(StatusReg & XIIC_SR_BUS_BUSY_MASK)) {
          SentByteCount = XIic_DynSend(IIC_BASE_ADDRESS, EepromIicAddr,
                    (u8 *) &Address, sizeof(Address), XIIC_REPEATED_START);
    }

  } while (SentByteCount != sizeof(Address));
  /*
   * Receive the data.
   */
  ReceivedByteCount = XIic_DynRecv(IIC_BASE_ADDRESS, EepromIicAddr,
                                          BufferPtr, ByteCount);

  /*
   * Return the number of bytes received from the EEPROM.
   */

  return ReceivedByteCount;

}
```

Vc707_hdmi.c

```
u8 EepromWriteByte(AddressType Address, u8 *BufferPtr, u8 ByteCount)
{
  u8 SentByteCount;
  u8 WriteBuffer[sizeof(Address) + PAGE_SIZE];
  u8 Index;
  /*
   * A temporary write buffer must be used which contains both the address
   * and the data to be written, put the address in first based upon the
   * size of the address for the EEPROM
   */
  if (sizeof(AddressType) == 2) {
          WriteBuffer[0] = (u8) (Address >> 8);
          WriteBuffer[1] = (u8) (Address);
  } else if (sizeof(AddressType) == 1) {
          WriteBuffer[0] = (u8) (Address);
          EepromIicAddr |= (EEPROM_TEST_START_ADDRESS >> 8) & 0x7;
  }
  /*
   * Put the data in the write buffer following the address.
   */
  for (Index = 0; Index < ByteCount; Index++) {
          WriteBuffer[sizeof(Address) + Index] = BufferPtr[Index];
  }
  /*
   * Write a page of data at the specified address to the EEPROM.
   */
  SentByteCount = XIic_DynSend(IIC_BASE_ADDRESS, EepromIicAddr,WriteBuffer, sizeof(Address) + ByteCount,XIIC_STOP);
  /*
   * Return the number of bytes written to the EEPROM.
   */
  return SentByteCount - sizeof(Address);}
```

Vc707_hdmi.c

```
XV_tpg tpg;

void ConfigTpg() {
XV_tpg_Initialize(&tpg, 0);
XV_tpg_DisableAutoRestart(&tpg);
XV_tpg_Set_height(&tpg, 1080);
XV_tpg_Set_width(&tpg, 1920);
XV_tpg_Set_colorFormat(&tpg, XVIDC_CSF_RGB);
XV_tpg_Set_bckgndId(&tpg, XTPG_BKGND_COLOR_BARS);
XV_tpg_Set_ovrlayId(&tpg, 1);
XV_tpg_Set_boxSize(&tpg, 100);
XV_tpg_Set_motionSpeed(&tpg, 10);
XV_tpg_EnableAutoRestart(&tpg);
XV_tpg_Start(&tpg);
}
```

Vc707_hdmi.c

```c
int main()
{
    int Status;

    init_platform();

    print("Hello World\n\r");
    print("Successfully ran Hello World application");

    Status = IicLowLevelDynEeprom();
    if (Status != XST_SUCCESS) {
    xil_printf("ADV7511 IIC programming FAILED\r\n");
    return XST_FAILURE;
      }
    xil_printf("ADV7511 IIC programming PASSED\r\n");



    print("----------------------------------\n\r");
    print(" ADV7511 HDMI Output Demo\n\r");
    print("----------------------------------\n\r");

    print("\n\r");
    print("TPG Configuration\n\r");
    ConfigTpg();

    //InitVprocSs_CSC();
return 0;
}
```

HDMI Output

Music
Pictures
Videos
Windows (C:)
Network
3:21 PM
52 items | 1 item selected
VC707_HDMI 10/24/2023 2:56 PM File folder
vc707_ibert 9/22/2023 9:32 AM File folder
vck190_bit 4/20/2023 3:12 PM File folder
vcu118_bit 12/9/2022 10:21 AM File folder
vcu128_bit 2/20/2023 6:17 PM File folder
XVES_0019 9/26/2023 1:41 PM File folder
BASE ON USER
BASE ON USER
BASE ON USER
BASE ON USER