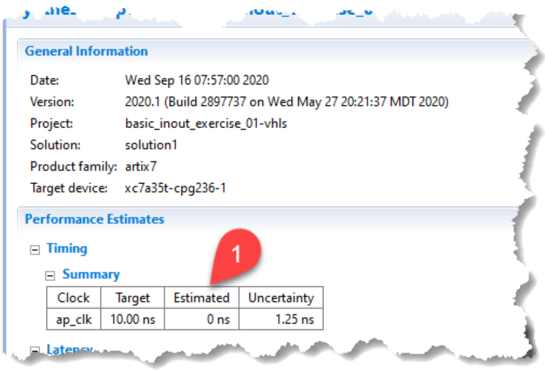
Eercise 1

(Note that the detail of a typical design flow is explained in the first exercise in the previous section)

1- Create a vivado-HLS project with the name of “basic_inout_exercise_01-vhls”
2- Add this top-function to the project

```
void basic_inout_exercise_01(short int sw, short int &led) {
#pragma HLS INTERFACE ap_ctrl_none port=return
#pragma HLS INTERFACE ap_none port=sw
#pragma HLS INTERFACE ap_none port=led
   led = sw;
}
```

3- This would be the HLS report



General Information

Date:	Wed Sep 16 07:57:00 2020
Version:	2020.1 (Build 2897737 on Wed May 27 20:21:37 MDT 2020)
Project:	basic_inout_exercise_01-vhls
Solution:	solution1
Product family:	artix7
Target device:	xc7a35t-cpg236-1

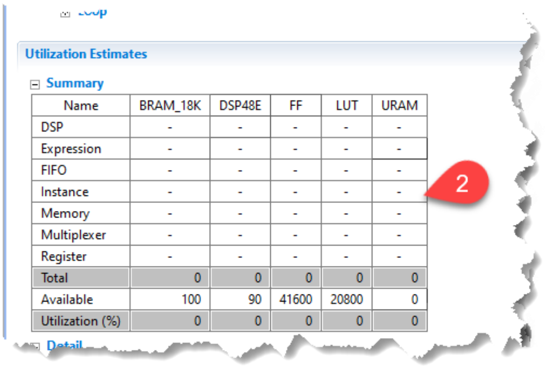
Performance Estimates

Timing

Summary

Clock	Target	Estimated	Uncertainty
ap_clk	10.00 ns	0 ns	1.25 ns

Latency

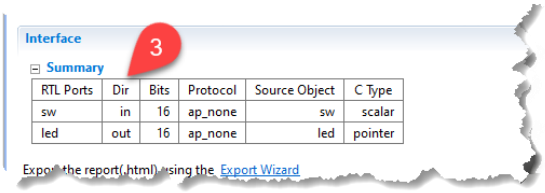


Utilization Estimates

Summary

Name	BRAM_18K	DSP48E	FF	LUT	URAM
DSP	-	-	-	-	-
Expression	-	-	-	-	-
FIFO	-	-	-	-	-
Instance	-	-	-	-	-
Memory	-	-	-	-	-
Multiplexer	-	-	-	-	-
Register	-	-	-	-	-
Total	0	0	0	0	0
Available	100	90	41600	20800	0
Utilization (%)	0	0	0	0	0

Detail



Interface

Summary

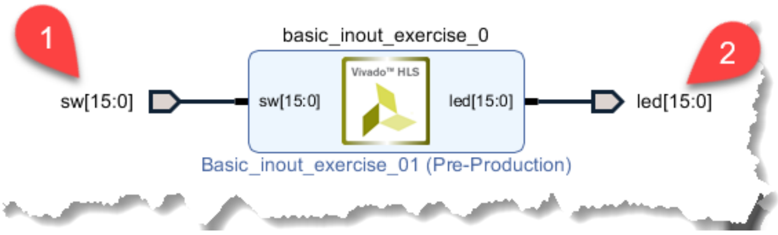
RTL Ports	Dir	Bits	Protocol	Source Object	C Type
sw	in	16	ap_none	sw	scalar
led	out	16	ap_none	led	pointer

Export the report(.html) using the Export Wizard

4- Create a Vivado project with the name of "basic_inout_exercise_01-vivado"

5- Add the Vivado-HLS IP to the Vivado repository

6- Add the design to the Diagram area, make the ports external and change their name as follows



7- Create a constraints file and add the following physical constraints

```
# Switches
set_property PACKAGE_PIN V17 [get_ports {sw[0]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[0]}]
set_property PACKAGE_PIN V16 [get_ports {sw[1]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[1]}]
set_property PACKAGE_PIN W16 [get_ports {sw[2]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[2]}]
set_property PACKAGE_PIN W17 [get_ports {sw[3]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[3]}]
set_property PACKAGE_PIN W15 [get_ports {sw[4]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[4]}]
set_property PACKAGE_PIN V15 [get_ports {sw[5]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[5]}]
set_property PACKAGE_PIN W14 [get_ports {sw[6]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[6]}]
set_property PACKAGE_PIN W13 [get_ports {sw[7]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[7]}]
set_property PACKAGE_PIN V2 [get_ports {sw[8]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[8]}]
set_property PACKAGE_PIN T3 [get_ports {sw[9]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[9]}]
set_property PACKAGE_PIN T2 [get_ports {sw[10]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[10]}]
set_property PACKAGE_PIN R3 [get_ports {sw[11]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[11]}]
set_property PACKAGE_PIN W2 [get_ports {sw[12]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[12]}]
set_property PACKAGE_PIN U1 [get_ports {sw[13]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[13]}]
set_property PACKAGE_PIN T1 [get_ports {sw[14]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[14]}]
set_property PACKAGE_PIN R2 [get_ports {sw[15]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {sw[15]}]


# LEDs
```

```
set_property PACKAGE_PIN U16 [get_ports {led[0]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[0]}]
set_property PACKAGE_PIN E19 [get_ports {led[1]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[1]}]
set_property PACKAGE_PIN U19 [get_ports {led[2]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[2]}]
set_property PACKAGE_PIN V19 [get_ports {led[3]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[3]}]
set_property PACKAGE_PIN W18 [get_ports {led[4]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[4]}]
set_property PACKAGE_PIN U15 [get_ports {led[5]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[5]}]
set_property PACKAGE_PIN U14 [get_ports {led[6]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[6]}]
set_property PACKAGE_PIN V14 [get_ports {led[7]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[7]}]
set_property PACKAGE_PIN V13 [get_ports {led[8]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[8]}]
set_property PACKAGE_PIN V3 [get_ports {led[9]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[9]}]
set_property PACKAGE_PIN W3 [get_ports {led[10]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[10]}]
set_property PACKAGE_PIN U3 [get_ports {led[11]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[11]}]
set_property PACKAGE_PIN P3 [get_ports {led[12]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[12]}]
set_property PACKAGE_PIN N3 [get_ports {led[13]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[13]}]
set_property PACKAGE_PIN P1 [get_ports {led[14]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[14]}]
set_property PACKAGE_PIN L1 [get_ports {led[15]}]
            set_property IOSTANDARD LVCMOS33 [get_ports {led[15]}]
```

8- Generate the bitstream and program the board. Then play with slide-switches and observe their impact on the leds. You should be able to control each LED by its close slide-switch.

Esercise 2

1- Create a vivado-HLS project with the name of “basic_inout_exercise_02-vhls”

2- Add this top-function to the project

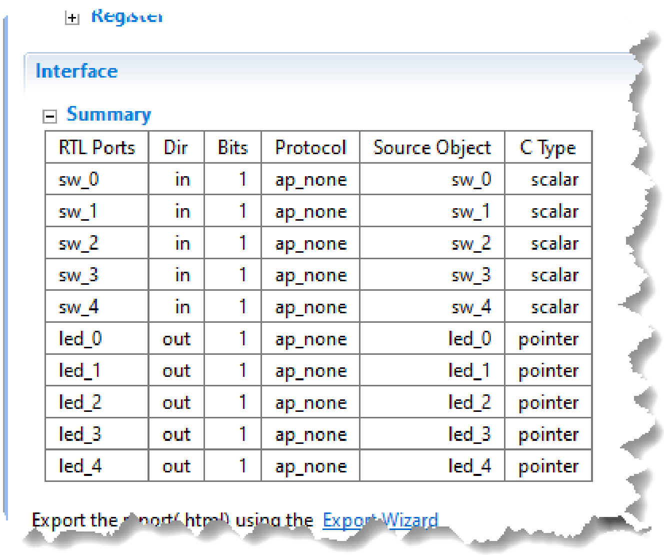
```
void basic_inout_exercise_02(
             bool sw_0,
             bool sw_1,
             bool sw_2,
             bool sw_3,
             bool sw_4,

             bool &led_0,
             bool &led_1,
             bool &led_2,
             bool &led_3,
             bool &led_4) {
#pragma HLS INTERFACE ap_ctrl_none port=return
#pragma HLS INTERFACE ap_none port=sw_0
#pragma HLS INTERFACE ap_none port=sw_1
#pragma HLS INTERFACE ap_none port=sw_2
#pragma HLS INTERFACE ap_none port=sw_3
#pragma HLS INTERFACE ap_none port=sw_4

#pragma HLS INTERFACE ap_none port=led_0
#pragma HLS INTERFACE ap_none port=led_1
#pragma HLS INTERFACE ap_none port=led_2
#pragma HLS INTERFACE ap_none port=led_3
#pragma HLS INTERFACE ap_none port=led_4

      led_0 = sw_0;
      led_1 = sw_1;
      led_2 = sw_2;
      led_3 = sw_3;
      led_4 = sw_4;
}
```

3- The following figure shows the design ports after synthesis



⊞ Register

Interface

⊟ Summary

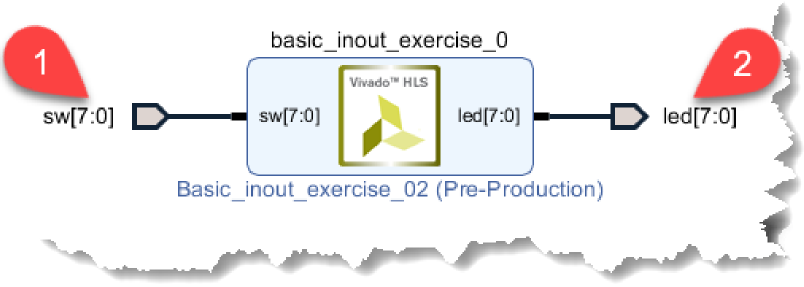
RTL Ports	Dir	Bits	Protocol	Source Object	C Type
sw_0	in	1	ap_none	sw_0	scalar
sw_1	in	1	ap_none	sw_1	scalar
sw_2	in	1	ap_none	sw_2	scalar
sw_3	in	1	ap_none	sw_3	scalar
sw_4	in	1	ap_none	sw_4	scalar
led_0	out	1	ap_none	led_0	pointer
led_1	out	1	ap_none	led_1	pointer
led_2	out	1	ap_none	led_2	pointer
led_3	out	1	ap_none	led_3	pointer
led_4	out	1	ap_none	led_4	pointer

Export the report(.html) using the Export Wizard

4- Create a Vivado project with the name of “basic_inout_exercise_01-vivado”

5- Add the Vivado-HLS IP to the Vivado repository

6- Add the design to the Diagram area, make the ports external and change their name as follows



7- Create a constraints file and add the following physical constraints

```
# Switches
set_property PACKAGE_PIN V17 [get_ports {sw[0]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {sw[0]}]
set_property PACKAGE_PIN V16 [get_ports {sw[1]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {sw[1]}]
set_property PACKAGE_PIN W16 [get_ports {sw[2]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {sw[2]}]
set_property PACKAGE_PIN W17 [get_ports {sw[3]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {sw[3]}]
set_property PACKAGE_PIN W15 [get_ports {sw[4]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {sw[4]}]
set_property PACKAGE_PIN V15 [get_ports {sw[5]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {sw[5]}]
set_property PACKAGE_PIN W14 [get_ports {sw[6]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {sw[6]}]
set_property PACKAGE_PIN W13 [get_ports {sw[7]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {sw[7]}]


# LEDs
set_property PACKAGE_PIN U16 [get_ports {led[0]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {led[0]}]
set_property PACKAGE_PIN E19 [get_ports {led[1]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {led[1]}]
set_property PACKAGE_PIN U19 [get_ports {led[2]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {led[2]}]
set_property PACKAGE_PIN V19 [get_ports {led[3]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {led[3]}]
set_property PACKAGE_PIN W18 [get_ports {led[4]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {led[4]}]
set_property PACKAGE_PIN U15 [get_ports {led[5]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {led[5]}]
set_property PACKAGE_PIN U14 [get_ports {led[6]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {led[6]}]
set_property PACKAGE_PIN V14 [get_ports {led[7]}]
	set_property IOSTANDARD LVCMOS33 [get_ports {led[7]}]
```

8- Generate the bitstream and program the board. Then play with slide-switches and observe their impact on the leds.

Esercise 3

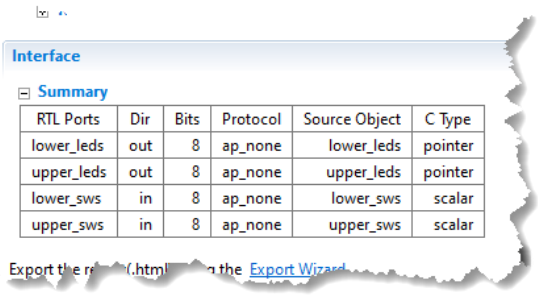
1- Create a new Vivado-HLS project with the name of “basic_inout_exercise_03-vhls”. Choose the “led_controller” as the top-function name.

2- Create a design file and write the following top-function in the file

```
void led_controller(
       char &lower_leds,
       char &upper_leds,
       char  lower_sws,
       char  upper_sws)
{
#pragma HLS INTERFACE ap_ctrl_none port=return
#pragma HLS INTERFACE ap_none port=lower_leds
#pragma HLS INTERFACE ap_none port=upper_leds
#pragma HLS INTERFACE ap_none port=lower_sws
#pragma HLS INTERFACE ap_none port=upper_sws

 lower_leds = lower_sws;
 upper_leds = upper_sws;
}
```

3- Then synthesise the code you should see the hardware ports as follow in the HLS report. It consists of two 8-bit input ports and two 8-bit output ports.



Interface

Summary

RTL Ports	Dir	Bits	Protocol	Source Object	C Type
lower_leds	out	8	ap_none	lower_leds	pointer
upper_leds	out	8	ap_none	upper_leds	pointer
lower_sws	in	8	ap_none	lower_sws	scalar
upper_sws	in	8	ap_none	upper_sws	scalar

Export the r (.html the Export Wizard

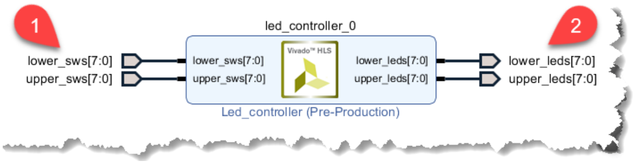
4- Now generate the RTL/IP.

5- Create a new Vivado project

6- Add the generated IP into the Vivado repository

7- Add the IP into the Diagrm tab.

8- Makes all the ports external and change their name as shown in the following figure



9- Create a new constraint file and define these constraints. Please

```
# Switches
set_property PACKAGE_PIN V17 [get_ports {lower_sws[0]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_sws[0]}]
set_property PACKAGE_PIN V16 [get_ports {lower_sws[1]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_sws[1]}]
set_property PACKAGE_PIN W16 [get_ports {lower_sws[2]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_sws[2]}]
set_property PACKAGE_PIN W17 [get_ports {lower_sws[3]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_sws[3]}]
set_property PACKAGE_PIN W15 [get_ports {lower_sws[4]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_sws[4]}]
set_property PACKAGE_PIN V15 [get_ports {lower_sws[5]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_sws[5]}]
set_property PACKAGE_PIN W14 [get_ports {lower_sws[6]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_sws[6]}]
set_property PACKAGE_PIN W13 [get_ports {lower_sws[7]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_sws[7]}]
set_property PACKAGE_PIN V2 [get_ports {upper_sws[0]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_sws[0]}]
set_property PACKAGE_PIN T3 [get_ports {upper_sws[1]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_sws[1]}]
set_property PACKAGE_PIN T2 [get_ports {upper_sws[2]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_sws[2]}]
set_property PACKAGE_PIN R3 [get_ports {upper_sws[3]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_sws[3]}]
set_property PACKAGE_PIN W2 [get_ports {upper_sws[4]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_sws[4]}]
set_property PACKAGE_PIN U1 [get_ports {upper_sws[5]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_sws[5]}]
set_property PACKAGE_PIN T1 [get_ports {upper_sws[6]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_sws[6]}]
set_property PACKAGE_PIN R2 [get_ports {upper_sws[7]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_sws[7]}]


# LEDs
set_property PACKAGE_PIN U16 [get_ports {lower_leds[0]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_leds[0]}]
set_property PACKAGE_PIN E19 [get_ports {lower_leds[1]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_leds[1]}]
set_property PACKAGE_PIN U19 [get_ports {lower_leds[2]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_leds[2]}]
set_property PACKAGE_PIN V19 [get_ports {lower_leds[3]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_leds[3]}]
set_property PACKAGE_PIN W18 [get_ports {lower_leds[4]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_leds[4]}]
set_property PACKAGE_PIN U15 [get_ports {lower_leds[5]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_leds[5]}]
set_property PACKAGE_PIN U14 [get_ports {lower_leds[6]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_leds[6]}]
set_property PACKAGE_PIN V14 [get_ports {lower_leds[7]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {lower_leds[7]}]
set_property PACKAGE_PIN V13 [get_ports {upper_leds[0]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_leds[0]}]
set_property PACKAGE_PIN V3 [get_ports {upper_leds[1]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_leds[1]}]
set_property PACKAGE_PIN W3 [get_ports {upper_leds[2]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_leds[2]}]
set_property PACKAGE_PIN U3 [get_ports {upper_leds[3]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_leds[3]}]
set_property PACKAGE_PIN P3 [get_ports {upper_leds[4]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_leds[4]}]
set_property PACKAGE_PIN N3 [get_ports {upper_leds[5]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_leds[5]}]
set_property PACKAGE_PIN P1 [get_ports {upper_leds[6]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_leds[6]}]
set_property PACKAGE_PIN L1 [get_ports {upper_leds[7]}]
        set_property IOSTANDARD LVCMOS33 [get_ports {upper_leds[7]}]
```

10- Generate the output ptoducts, create the HDL wrapper and finally generate the bitstream.

11- Program the board. You should be able to control each LED with the slide switch close to that.