

การทดลองที่ 4 การ Implement

จุดประสงค์

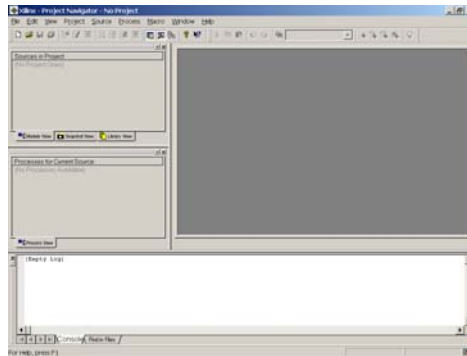
1. เพื่อให้นักศึกษาได้รู้จักวิธีการใช้โปรแกรม Xilinx WebPACK สำหรับการ Implement ภาษา VHDL คำสั่ง

การทดลองนี้เป็นการทดลองที่จะฝึกให้นักศึกษาได้รู้จักวิธีการใช้โปรแกรม Xilinx WebPACK สำหรับการ Implement ภาษา VHDL ลงชิพซีพีแอลดี

ให้นักศึกษาศึกษาใช้ วิชาซีแอลโมเดล วงจร Full Adder จากแลปที่ 1 มาทำการทดลอง โดยทำตามลำดับขั้นดังนี้

สร้างโปรเจคใหม่

1. เมื่อเปิด (start) โปรแกรม) Xilinx WebPACK 4.2 → WebPACK Project Navigator โปรแกรมจะเข้าสู่หน้าจอ Xilinx-Project Navigator



ให้เลือกคำสั่ง file -> new project เพื่อสร้างโปรเจคใหม่ เมื่อปรากฏหน้าต่าง New Project ให้ทำดังนี้

1. เลือก project Location

2. พิมพ์ชื่อโปรเจค

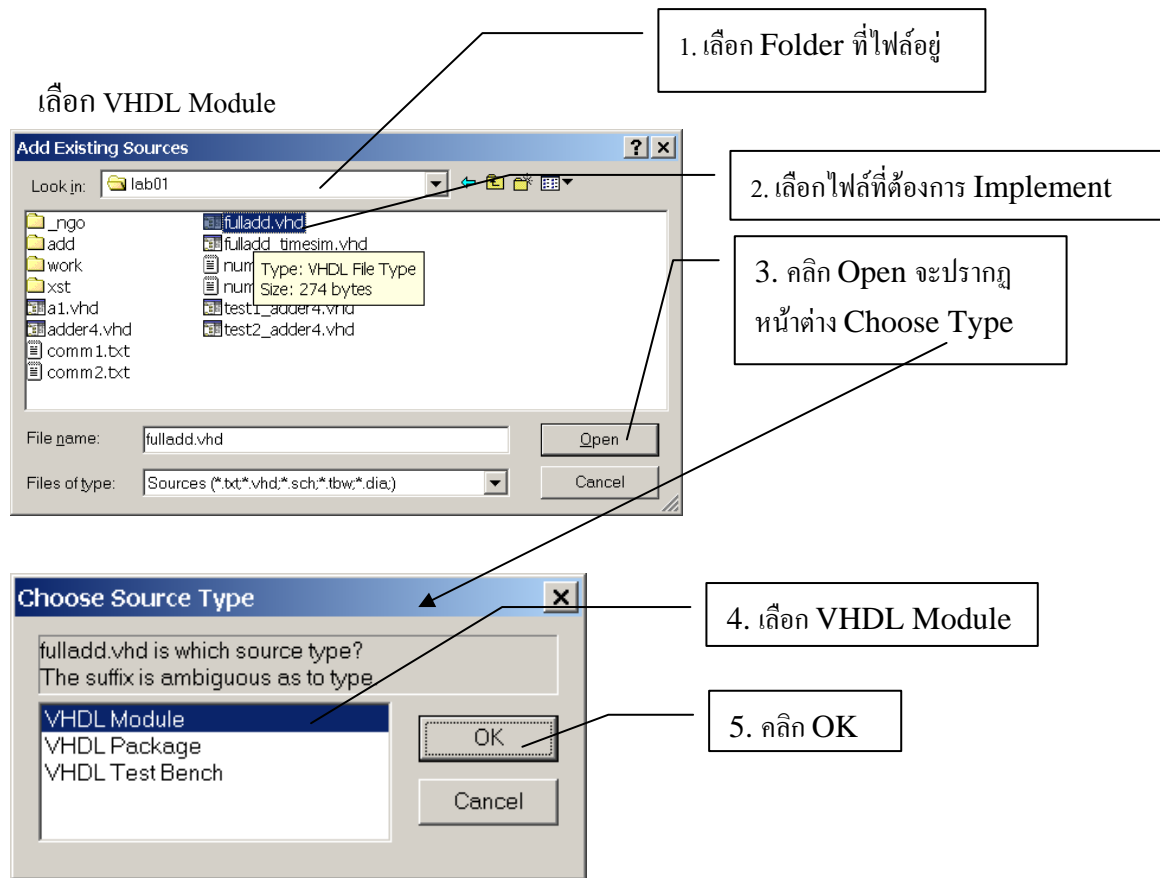
3. คลิกเลือก XC9500 CPLDs

4. คลิกเลือก XC9572 PC44

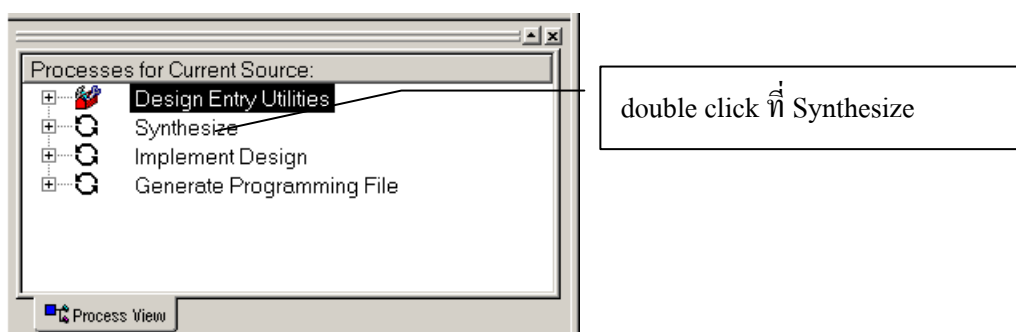
5. คลิกเลือก XST VHDL

Property Name	Value
Device Family	XC9500 CPLDs
Device	XC9572 PC44
Design Flow	XST VHDL

2. กำหนดไฟล์ VHDL โดยใช้คำสั่ง Project -> Add source โดยคลิกขวาที่พื้นที่ Sources in Project window เลือก add source -> fulladd.vhd

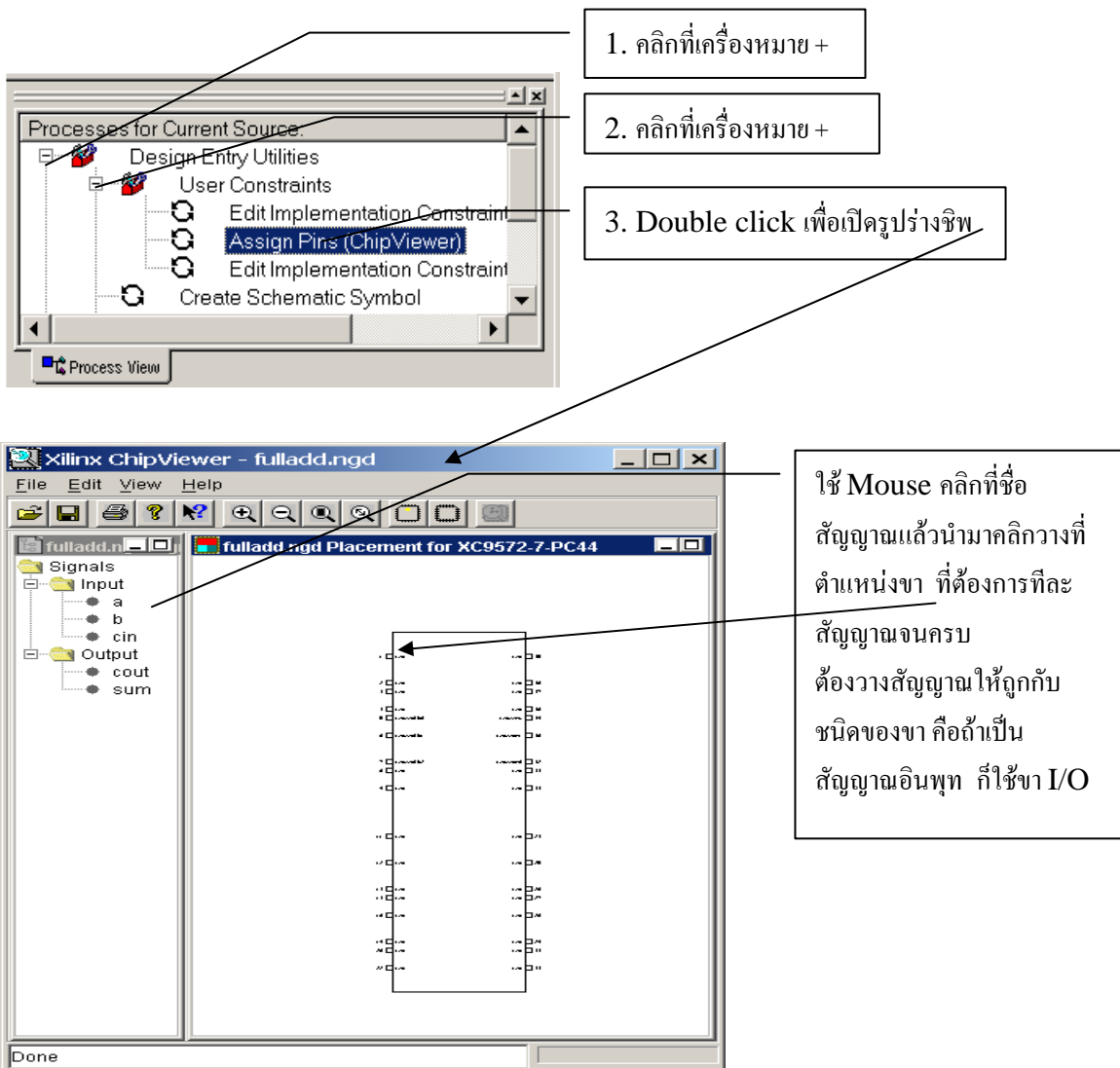


3. Synthesize โดย double click ที่ Synthesize ใน Processes for Current Source



4. สร้างไฟล์ fulladd.ucf เพื่อกำหนดค่า โดยเปิด Design Entry Utilities, เปิด User Constraints

4.1 Double click ที่ Assign Pins (Chip Viewer)



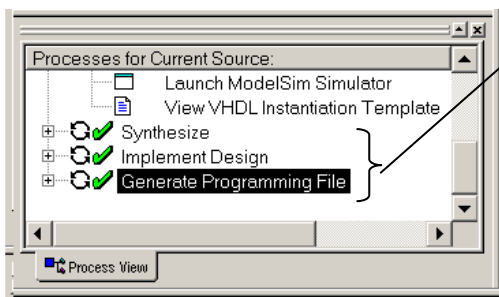
เมื่อกำหนดตำแหน่งสัญญาณครบแล้ว Save จะได้ไฟล์ Fulladd.ucf ดังตัวอย่างนี้

```
// Template UCF file created by the Project Navigator
NET a LOC=P26;
NET b LOC=P27;
NET cin LOC=P28;
NET cout LOC=P2;
NET sum LOC=P1;
```

สัญญาณ a

ขาไอซี
หมายเลข 26

5. Implement ลงชิพสร้างไฟล์ JEDEC โดย Double click Implement Design และ Generate Programming File ใน Processes for Current Source

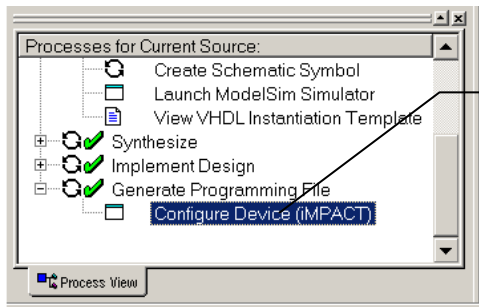


Double click หรือ คลิกขวา แล้วเลือก RUN

6. โปรแกรมลงชิพ

6.1 ต่อสาย Interface ระหว่างพอร์ตขนานกับบอร์ดทดลอง ป้อนไฟเข้าบอร์ดทดลอง

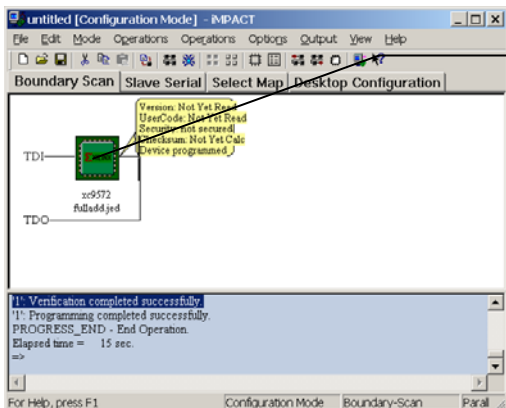
6.2 คลิกที่ Configure Device (iMPACT)



Double click หรือ คลิกขวา แล้วเลือก RUN

6.3 เมื่อปรากฏหน้าต่าง Configuration Mode ในขณะนี้โปรแกรมทำการตรวจสอบว่ามีชุด Download อยู่หรือไม่ บางครั้งถ้าโปรแกรมตรวจสอบผ่านทางพอร์ตอนุกรม เมื่อไม่พบจะขึ้นหน้าต่างเพื่อให้เราเลือกพอร์ต ให้คลิกเลือกเป็นพอร์ตขนาน

6.4 เมื่อโปรแกรมพร้อมจะ Download จะปรากฏรูปตัว IC ให้คลิกขวาที่ตัว IC แล้วเลือกคำสั่งโปรแกรม หรือเมื่อคลิกซ้ายที่ตัว IC แล้วเมนู Operations -> Program และติ๊ก Erase before Programming และ Verify โปรแกรมก็จะทำการโปรแกรมชิพ เมื่อเสร็จแล้วชิพจะเป็นวงจรที่เราออกแบบ พร้อมทดสอบการทำงาน
หมายเหตุบางครั้ง การโปรแกรมอาจไม่ได้ผลให้ลองทำให้



Double click หรือ คลิกขวา แล้วเลือก RUN

ให้ทำการ Implement วงจรในการทดลองที่ 3 ทั้งหมดโดยกำหนดตำแหน่งสัญญาณเป็นดังนี้

วงจรถอดรหัส 7-Segment

```
NET segm<0> LOC=P1;
NET segm<1> LOC=P2;
NET segm<2> LOC=P3;
NET segm<3> LOC=P4;
NET segm<4> LOC=P5;
NET segm<5> LOC=P6;
NET segm<6> LOC=P7;
NET segm<7> LOC=P8;
NET bin<3> LOC=P25;
NET bin<2> LOC=P26;
NET bin<1> LOC=P27;
NET bin<0> LOC=P28;
```

วงจรถอดรหัสแบบ 2-to-4 Line Decoder

```
NET y<0> LOC=P11;
NET y<1> LOC=P12;
NET y<2> LOC=P13;
NET y<3> LOC=P14;
NET s<1> LOC=P26;
NET s<0> LOC=P27;
NET g LOC=P28;
```

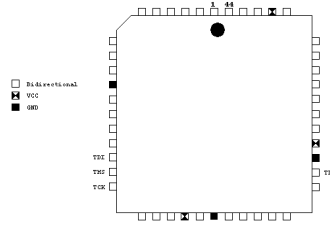
วงจรมัลติเพล็กซ์แบบ 4-line multiplexer

```
NET y LOC=P14;
NET x<3> LOC=P22;
NET x<2> LOC=P24;
NET x<1> LOC=P25;
NET x<0> LOC=P26;
NET s<1> LOC=P27;
NET s<0> LOC=P28;
```

วงจรเข้ารหัส 4 Line Priority Encoder

```
NET y<0> LOC=P12;
NET y<1> LOC=P13;
NET eo LOC=P14;
NET xi<3> LOC=P25;
NET xi<2> LOC=P26;
NET xi<1> LOC=P27;
NET xi<0> LOC=P28;
```

ตำแหน่งสัญญาณของ XC9572



ตาราง ตำแหน่ง

ขาสัญญาณของ XC9572

Pin	FB	Mac	Function	Pin	FB	Mac	Function
1	1	2	I/O	23			GND
2	1	5	I/O	24	4	2	I/O
3	1	6	I/O	25	4	5	I/O
4	1	8	I/O	26	4	8	I/O
5	1	9	I/O/GCK1	27	4	9	I/O
6	1	11	I/O/GCK2	28	4	11	I/O
7	1	14	I/O/GCK3	29	4	14	I/O
8	1	15	I/O	30			TDO
9	1	17	I/O	31			GND
10			GND	32			VCCIO
11	3	2	I/O	33	4	15	I/O
12	3	5	I/O	34	4	17	I/O
13	3	8	I/O	35	2	2	I/O
14	3	9	I/O	36	2	5	I/O
15			TDI	37	2	6	I/O
16			TMS	38	2	8	I/O
17			TCK	39	2	9	I/O/GSR
18	3	11	I/O	40	2	11	I/O/GTS2
19	3	14	I/O	41			VCCINT
20	3	15	I/O	42	2	14	I/O/GTS1
21			VCCINT	43	2	15	I/O
22	3	17	I/O	44	2	17	I/O

FB : Function Block number

Mac : Macrocell

Pin : Pin number