XILINX®

BASIC TRAINING

# Foundation Basic FPGA Design Flow Tutorial

서울시 마포구 서교동 444-17 덕화빌딩3층
㈜현명전자 응용기술부
Tel 02-3141-0147 / Fax 02-3141-0149
hmkor@hmelec.co.kr

$X_S\ A_T\ C_E\ T_P$

# Contents

개요 (FPGA TUTORIAL OVERVIEW) ..... 3

설계 전 준비사항 (DESIGN PREPARATION) ..... 4

프로젝트 열기 (OPENING A PROJECT) ..... 5

PROJECT MANAGER GUI ..... 7

DESIGN ENTRY ..... 8

WATCH HDL DESIGN ..... 9

LOGIBLOX ..... 11

디자인의 합성 (SYNTHESIZING THE DESIGN) ..... 13

EXPRESS CONSTRAINTS GUI ..... 15

HDL DESIGN 표현하기(IMPLEMENTING THE HDL DESIGN) ..... 17

WATCH SCHEMATIC DESIGN ..... 19

CNT60 SCHEMATIC MODULE 보기 (VIEWING THE CNT60 SCHEMATIC MODULE) ..... 21

HDL DESIGN MODULES 보기 (VIEWING HDL DESIGN MODULES ) ..... 22

STATE MACHINE DESIGN MODULES 보기 (VIEWING STATE MACHINE DESIGN MODULES ) ..... 24

LOGIBLOX DESIGN MODULES 보기 (VIEWING LOGIBLOX DESIGN MODULES) ..... 26

XILINX 공통 라이브러리에 기반한 설계모듈 보기 (VIEWING DESIGN MODULES BASED ON XILINX UNIFIED LIBRARIES COMPONENTS) ..... 28

IMPLEMENTING THE SCHEMATIC DESIGN ..... 30

FLOW ENGINE ..... 33

VERSIONS TAB ..... 34

IMPLEMENTATION 보고서 살펴보기 (REVIEWING IMPLEMENTATION REPORTS) ..... 35

타이밍 시뮬레이션 수행 (PERFORMING TIMING SIMULATION) ..... 36

SCRIPT 파일을 사용한 시뮬레이션 (SCRIPT FILE METHOD OF SIMULATION) ..... 37

DOWNLOADING THE DESIGN ..... 38

# 개요 (FPGA Tutorial Overview)

이 문서는 Foundation F1.5 FPGA 설계의 흐름을 간략히 설명하고 프로젝트를 열고 설계를 시작해서 Xilinx Device를 프로그램하는 마지막 단계까지의 설계흐름을 안내한다. 하나의 완성된 설계를 가지고 시작하여, 실제 자신의 설계단계에서 사용할 여러 가지 도구들의 사전검토로 설계흐름을 통하여 나아갈 수 있다. 당신이 본 교재에서 선택한 프로젝트에 따라 Schematic flow (WATCH_SC)나 HDL flow (WATCHVHD or WATCHVER)중 하나를 사용할 수 있다. 이 문서는 각각의 단계별로 설계의 도입부가 약간씩 다르기 때문에 각 흐름별로 분리되어 있는 설계 도입의 관점에서 접근한다. 이문서는 10분 ~ 20분이면 끝마칠 수 있다.

좀더 자세한 교재는 *Foundation Series Quickstart Guide F1.5*에서 찾아볼 수 있다. 심화학습 교재는 설계시작에서 검증 그리고 마지막 단계인 Xilinx FPGA내로 Implementing까지를 통해 설계방향을 제시한다.

CPLD 설계 흐름에 관련된 정보는 CPLD Design Flows라는 Help문서를 참고할 것.

## 설계 전 준비사항 (Design Preparation)

이 교재에서는 Watch라고 불리는 Stopwatch 디자인을 사용한다. 이 디자인은 Foundation Software를 설치할 때 동일 프로젝트명으로 기본적으로 설치되어진다. 만약 이 프로젝트가 설치되어 있지 않다면, CDROM에 있는 Foundation 설치 프로그램을 수행하여 Sample project를 설치한다.

Sample 프로젝트가 설치되었는지를 확인하려면, Foundation이 설치된 위치가 C:\FNDTN일 때 C:\FNDTN\ACTIVE\PROJECTS\WATCH, (WATCH_SC, WATCHVHD, WATCHVER)라는 이름의 디렉토리를 찾아보면 된다. 총 3개의 Watch 프로젝트이다.

WATCH_SC Schematic 기반의 프로젝트

WATCHVHD VHDL 기반의 프로젝트

WATCHVER Verilog 기반의 프로젝트

본 교재에서는 이들 세 개의 프로젝트중 하나를 사용하면 된다. 이 교재 전체에서 디자인은 일반적으로 Watch라는 프로젝트에서 인용한다.

---

**Note:** Watch 디자인의 VHDL / Verilog판을 수행하기 위해서는 유효한 Synopsys 합성용 License가 포함된 Foundation Express와 Base Express 패키지가 필요하다.

Watch 디자인은 XC4003E 디바이스에 맞추어져 있다. Xchecker또는 JTAG 케이블을 사용하여 Xilinx Demonstration Board로 설계한 내용을 다운로드 할 수 있다. (*Foundation Series Quickstart Guide 1.5*내에서 찾을 수 있는 In-Depth Tutorial중 Hardware Verification부분에서 Download를 하는 방법을 설명한 부분을 찾아볼것.)

또한 *Foundation Series Quick Start Guide 1.5*내에 있는 In-Depth Tutorial에는 Solution 프로젝트로서 WATCH 프로젝트를 사용한다. 우리는 추후 실제 설계시 참고용으로 이 디자인을 보관해둘 것을 권고한다.

WATCH 디렉토리내의 파일들을 온전하게 보존하기 위해, 다른 디렉토리를 선택하여 WATCH 프로젝트를 복사한다. 우선, WATCH 프로젝트를 열고 새로운 이름을 주어서 다른 디렉토리에 복사를 한후, 작업 디렉토리내에서 이름을 바꾼 프로젝트를 열면된다.

# 프로젝트 열기 (Opening a Project)

Foundation내의 모든 도구들은 Xilinx FPGA와 CPLD의 설계에 도움을 주도록 만들어져 있다. 이들 도구들은 Project Manager라고 불리는 단일 프로그램상에서 모두 사용할 수 있다.

1. Project Manager를 시작하려면, Windows 데스크 탑에 있는 Xilinx Foundation Project Manager 아이콘을 더블클릭 하거나 또는 **시작**메뉴에서 **프로그램(P) -> Xilinx Foundation Series -> Xilinx Foundation Project Manager**를 선택한다.

   Getting Started라는 대화상자가 열린다. 이 대화상자에서 열기를 원하는 프로젝트를 선택한다.

2. **More Projects** 버튼을 클릭하여 3개의 WATCH 프로젝트인 WATCH_SC, WATCHVHD, WATCHVER중 (C:\fndtn\active\projects 디렉토리에 위치한) 하나를 Open Project 대화상자의 왼쪽편의 목록 중에서 선택한다.

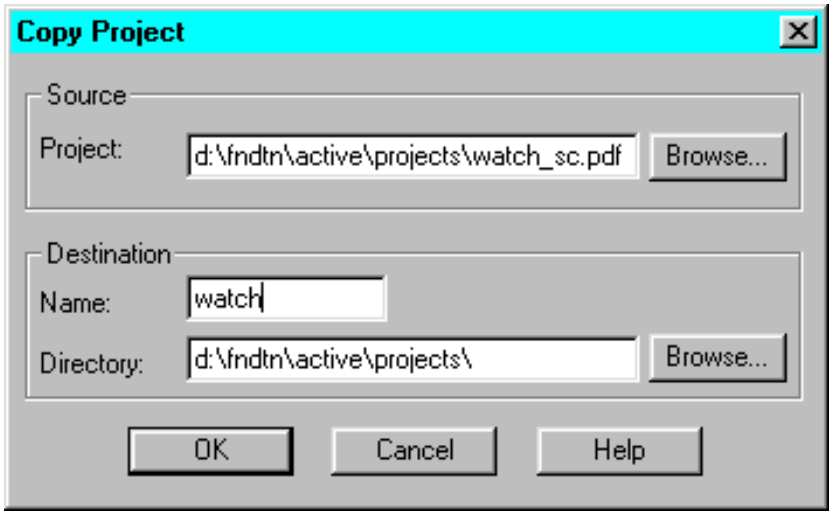
3. WATCH 프로젝트를 열려면, 프로젝트명을 더블클릭 하거나 혹은 목록을 선택 후 **Open** 버튼을 클릭한다.

**프로젝트 복사하기**

1. Project Manager가 열린 후, **File -> Copy Project**를 선택한다.

   (본 문서와 함께 사용하기 위해서는 프로젝트를 복사해야 한다.)

2. Copy Project 대화상자의 Destination Name 필드에다가 watch혹은 자신이 원하는 다른 이름을 입력한다.




3. 새 작업영역에 프로젝트를 복사하려면 **OK**를 클릭한다.

**복사된 프로젝트 열기**

1. **File -> Open Project**를 선택하여 Project 목록부분을 스크롤하여 Watch 프로젝트를 선택한다.

2. **Open**을 클릭하여 Watch 프로젝트를 연다.
   Watch 프로젝트가 열리면서 Project Manager에 프로젝트가 로드 되고 내용을 볼 수 있다.

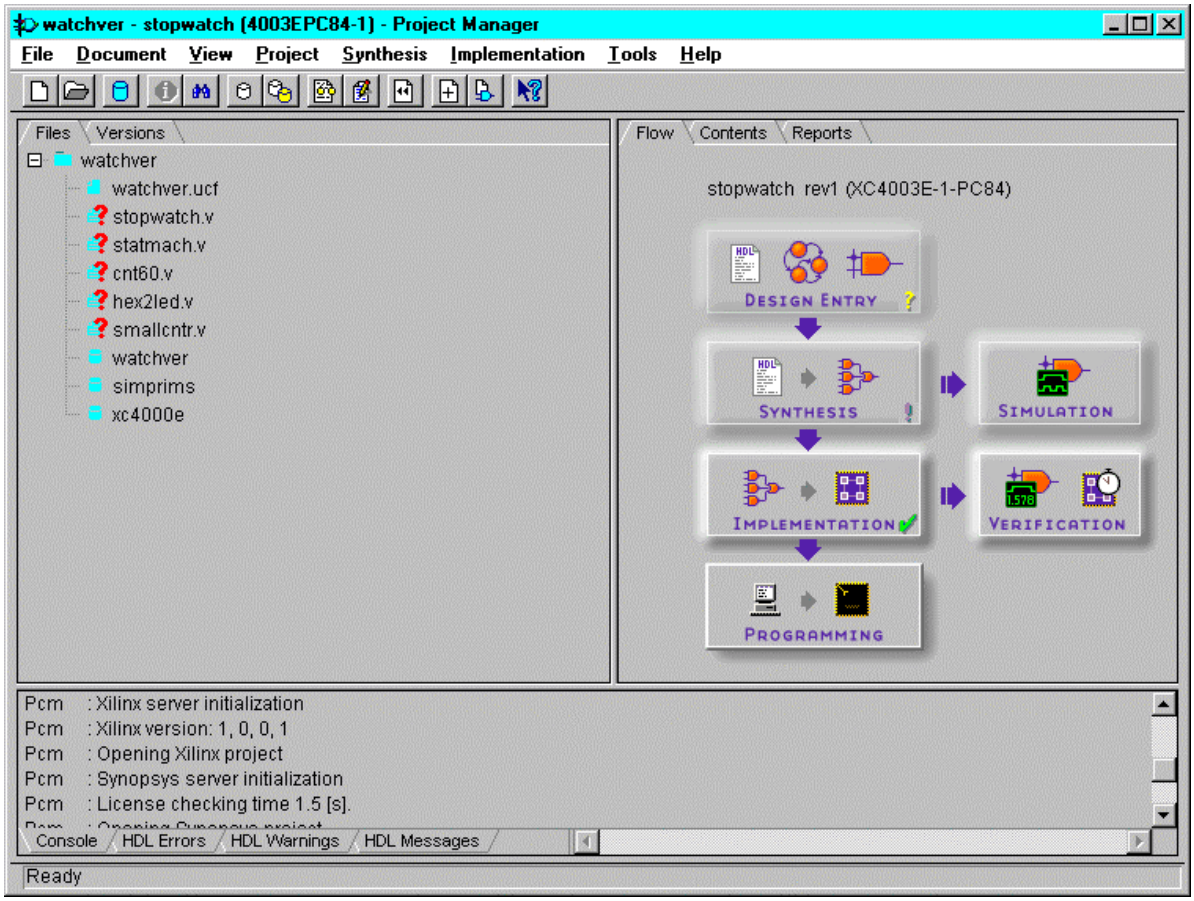
## Project Manager GUI

Project Manager는 세 개의 섹션으로 구분된다.

The Hierarchy Browser (왼쪽) 두개의 탭 (File탭과 Versions탭)으로 구분되고 프로젝트 파일들과 라이브러리들 그리고 Implementation 버전들이 계층구조로 표시된다.

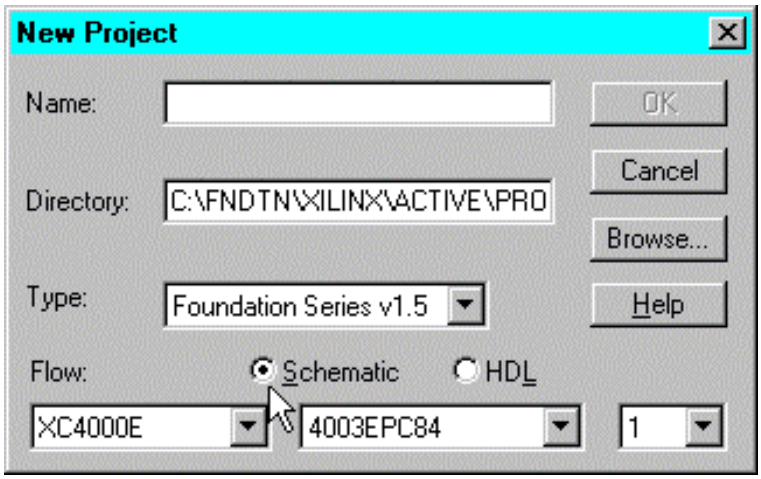
The Project Flowchart (오른쪽) 여러 개의 탭 (Flow, Contents, Reports)으로 구분되며 플로우차트가 포함된 Flow탭은 설계 전반을 통해 도움을 준다. 만약 Schematic 프로젝트를 열었다면 프로젝트 플로우차트에서 SYNTHESIS 버튼은 보이지 않을 것이다.

The Console area (아래쪽) 오류, 경고 및 관련된 정보들을 표시한다.



## Design Entry

디자인 엔트리는 별개의 두가지 프로젝트가 존재한다. (Foundation Schematic과 HDL) 최상위 단계의 디자인 Source (Schematic또는 HDL)는 Xilinx Foundation Series 도구를 통해서 설계방향을 정의할 수 있다. 이는 사용자가 사용할 프로젝트를 생성할 때 지정할 수 있다. 한번 프로젝트를 생성하면, 프로젝트 설계방향을 변경할 수 없다.




이 교재의 사용을 위해 선택한 프로젝트에 따라 사용자는 Schematic flow (WATCH_SC)나 HDL flow (WATCHVHD, WATCHVER)를 사용할 수 있다. 이 문서는 각각의 단계별로 설계의 도입부가 약간씩 다르기 때문에 각 흐름별로 분리되어 있는 설계 도입의 관점에서 접근한다.

사용자가 선택한 디자인에 적합한 Flow를 시작하려면, 아래의 두 항목 중 선택한 부분으로 넘어간다.

Schematic-based flow (Watch_sc)

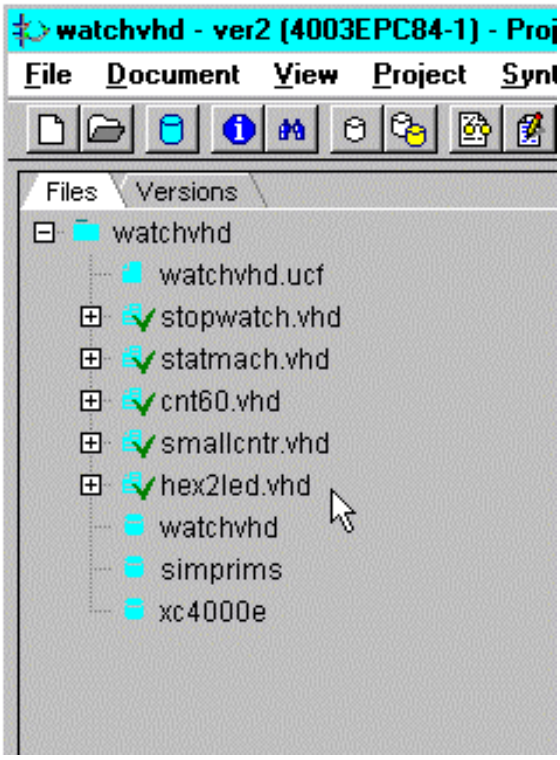
HDL-based flow (Watchvhd or Watchver)

---

**Note:** 사용자는 HDL 기반의 Flow를 따라 하기 위해서는 반드시 유효한 Synopsys 합성용 License가 포함된 Foundation Base Express나 Foundation Express 패키지를 가지고 있어야 한다.

## Watch HDL Design

Watch HDL 디자인은 계층구조로 된 HDL 기반의 디자인이다. 이는 최상위 파일이 저수준의 설계 모듈이 포함된 HDL파일임을 의미한다. Watch 디자인에서 저수준의 모듈은 HDL 파일들과 LogiBLOX 모듈들이다.



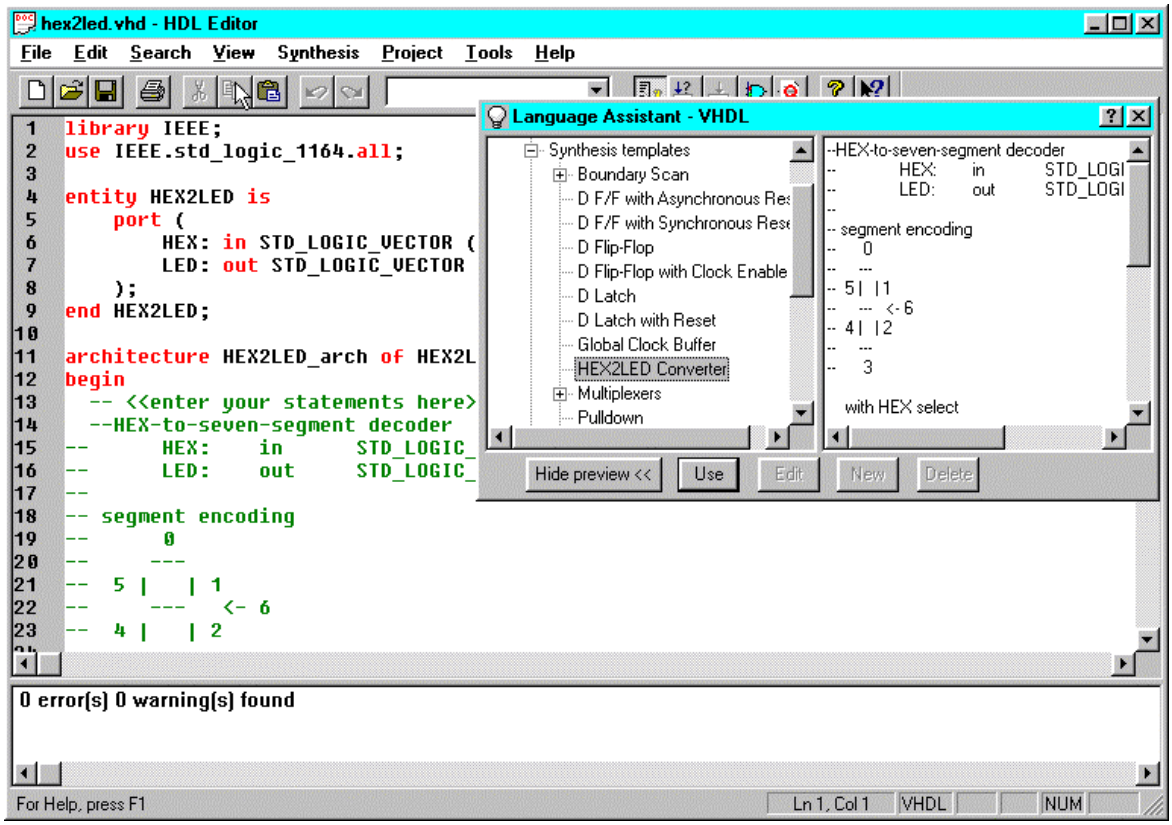

Project Manager의 Files 탭에서 사용자는 Project의 부분으로 되어있는 HDL 파일들을 볼 수 있다.

1. HDL 편집기로 파일을 불러들이려면 Hex2led.vhd (VHDL)나 Hex2led.v (Verilog)를 더블클릭을 한다.

   HDL 편집기는 HDL 파일들을 손쉬운 도입 구조를 제공한다. 이 편집기는 키워드, 주석, 변수등이 각기 다른색으로 표현된다. 사용자는 또한 HDL 편집기에서 직접 **Synthesis -> Check Syntax**를 선택하여 파일의 문법을 검사할 수 있다.

2. **Tools -> Language Assistant** 선택

   Language Assistant 창을 열면 일반적으로 사용되는 HDL구문들을 위한 HDL 템플리트가 보인다. Language Assistant는 현재 사용자가 선택하여 사용중인 언어에 근거하여 VHDL이나 Verilog 템플리트를 제공한다. HEX2LED라는 이HDL File에서 생성된 Code는 Synthesis 템플리트에서 바로 제공된다.




3. Language Assistant는 오른쪽 상단 구석에 있는 **X** 표시를 클릭하면 닫힌다.

4. HDL 편집기를 닫는다.

# LogiBLOX

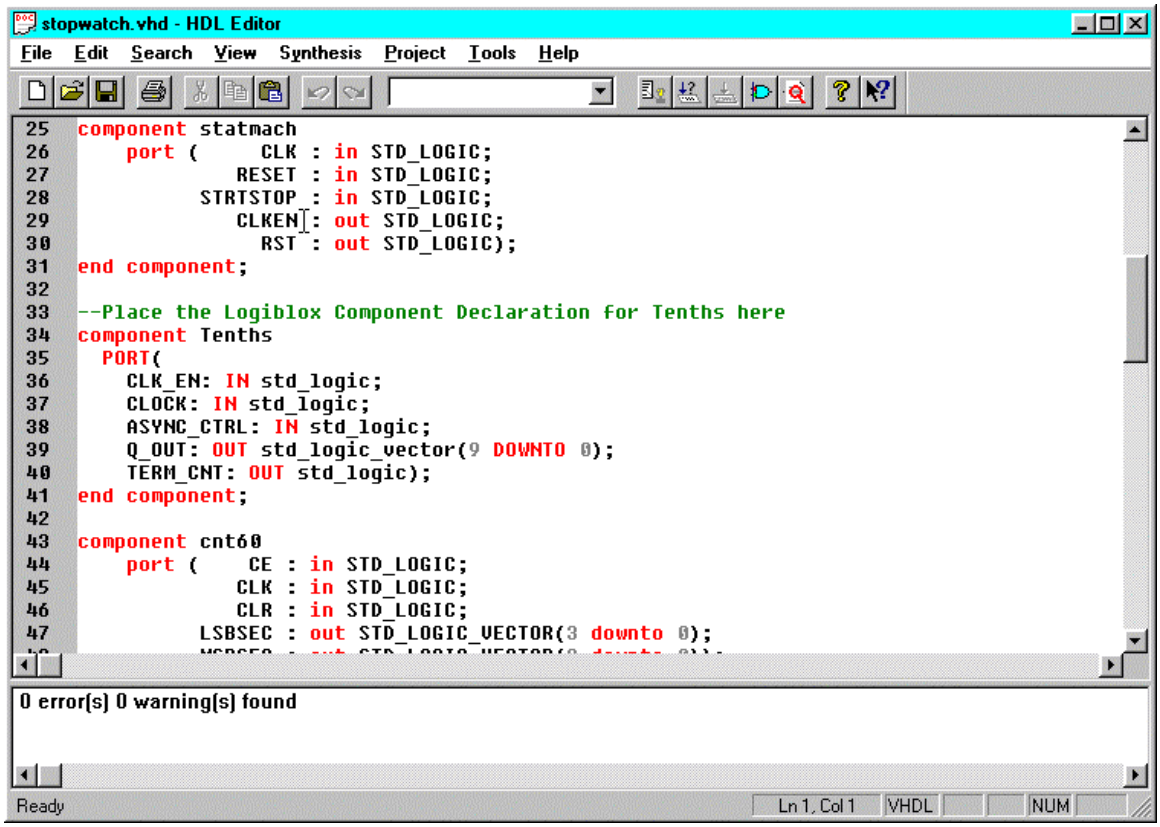
LogiBLOX은 사용자가 고유한 라이브러리 컴포넌트를 만들기 위해 사용하는 그래픽한 대화형 도구이다. LogiBLOX은 Fast Carry Logic이나 CLB RAM, ROM같은 Xilinx Device에 특화된 구조적인 면의 장점을 가지는 각Device구조에 최적화 된 컴포넌트를 생성한다.

1. Project Manager에서 stopwatch.vhd이나 stopwatch.v를 더블 클릭하여 HDL 편집기를 연다.

   이 파일은 이 프로젝트의 최상위 수준의 HDL이다.

2. Tenths라는 컴포넌트 선언문이 보일 때까지 HDL 파일을 아래로 스크롤 한다.

   Tenths 컴포넌트는 LogiBLOX 모듈이다.

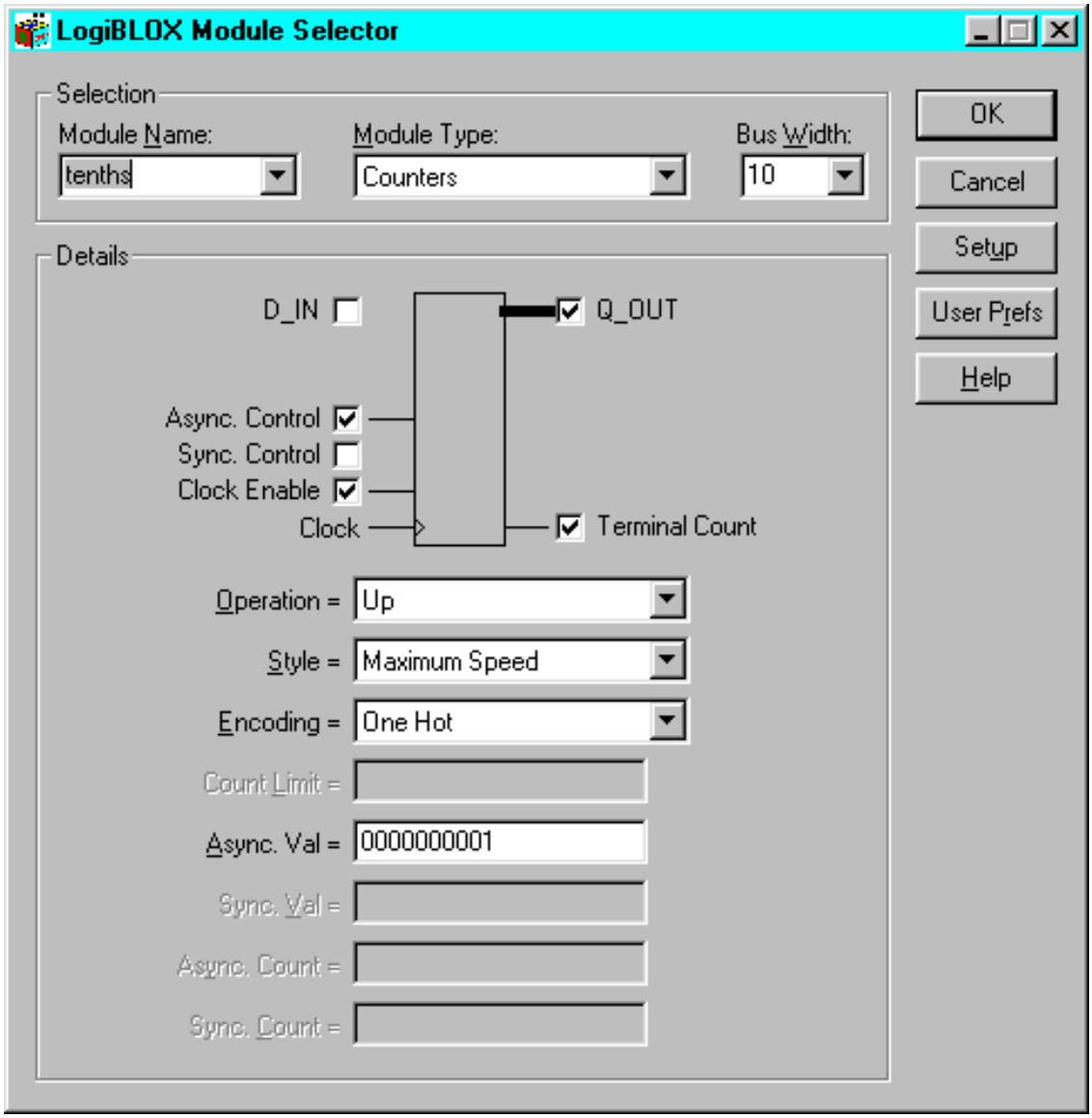



3. Synthesis menu에서 LogiBLOX을 선택한다.

   사용자가 고유한 LogiBLOX 모듈을 만들 때 사용하는 LogiBLOX 모듈 선택 GUI가 열린다. 이 GUI내에서 사용자는 모듈 명, 형태, 기능 등을 선택한다.

4. Tenths 컴포넌트를 선택하기 위해 Module Name 필드의 풀-다운 메뉴를 사용한다.

   이단계는 생성되어져 있는 Tenths 컴포넌트를 선택하는 표시창을 연다. 사용자는 이제 10비트 One-hot방식으로 인코딩된 계수기인 Tenths 모듈을 볼 수 있다. 추가적으로, LogiBLOX은 HDL 코드안으로 이 모듈을 삽입하는데 도움을 주는 템플리트 파일을 생성한다. 이들 템플리트 파일들은 VHDL용으로 <*module*>.vhi, Verilog용으로는 <*module*>.vei라고 불리운다. 한번 템플리트 파일이 생성되면 Stopwatch HDL 파일내의 Tehths 컴포넌트 선언과 삽입은 완전히 끝난다. You have the option now to open these files in the HDL Editor to see what they look like.




5. LogiBLOX Module Selector를 닫기 위해 **Cancel**을 클릭한다.

   본 교재용으로 LogiBLOX 모듈이 이미 생성되어 있기 때문에 사용자는 **OK**대신 **Cancel**을 클릭해야 한다.

6. HDL 편집기를 닫는다.

# 디자인의 합성 (Synthesizing the Design)

사용자는 아래에 나타난 방법중 하나를 선택하여 Foundation Project Manager에서 설계한 내용을 합성을 할 수 있다.

Files 탭에 있는 Local menu

Project Manager의 Synthesis 메뉴 아래의 Menu 선택

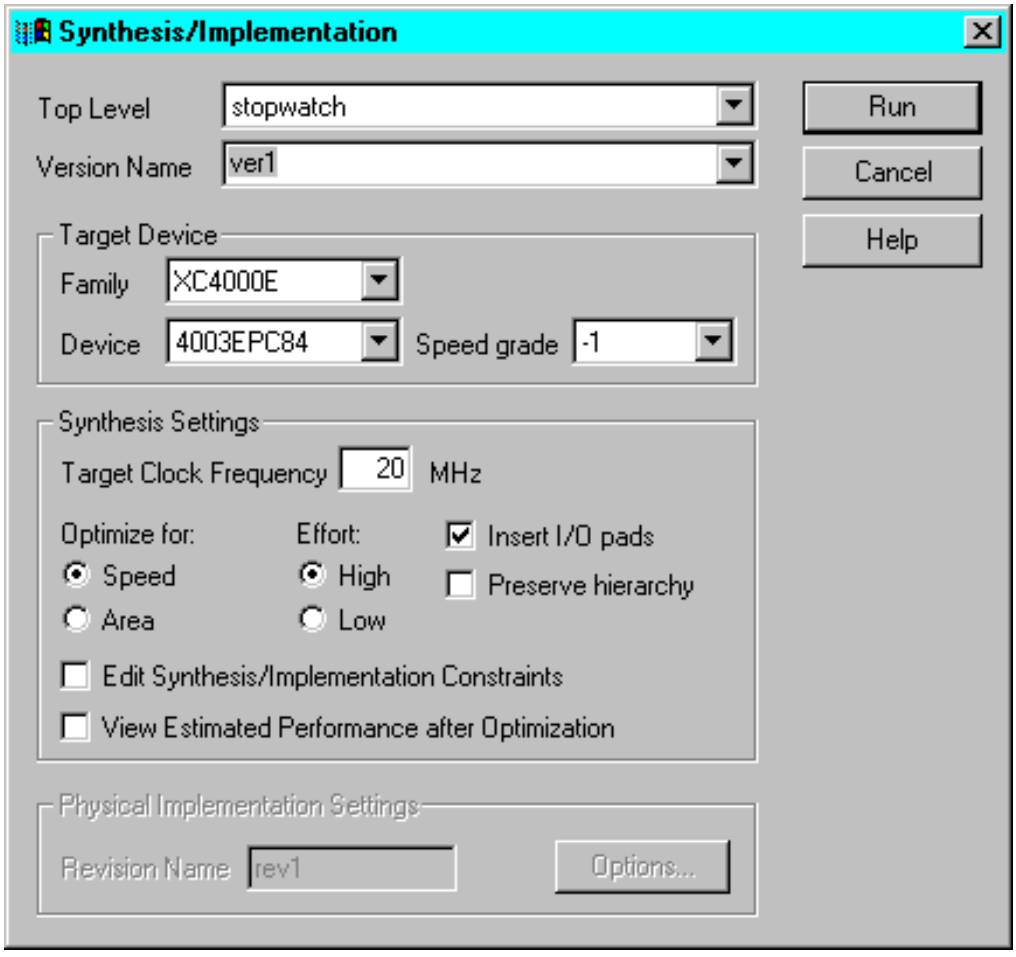
Flow Diagram상의 Synthesis 버튼

이 교재에서는 Flow Diagram상의 Synthesis 버튼을 사용한다.

1. 합성단계의 초기화를 하기위해, Project Manager의 Flow 탭에 있는 **Synthesis** 버튼을 클릭한다.

사용자는 Synthesis/Implementation 대화상자를 볼 수 있다. 최상위 단계의 Entity 명으로서 stopwatch로 되었는지 검사하고 Version Name 필드에 version name을 입력한다. Version Name은 사용자가 원하는 아무 이름이나 할 수 있다.

이 대화상자에서 또한 사용자는 Device를 선택하고, 시스템 클럭 주파수와 최적화 전략, I/O Pad의 삽입여부를 선택할 수 있다.



2. 만약 사용자가 Foundation Express 라이센스를 가지고 있다면, Edit Synthesis/Implementation Constraints와 View Estimated Performance after Optimization 항목을 선택할 수 있다.

   이부분은 사용자가 design constraint들의 입력과 Express Constraints GUI에서 합성후 결과 보기를 허용한다.

**Note:** Foundation Base Express 라이센스는 Express Constraints GUI를 위한 허가부분이 빠져있다. Base 사용자는 UCF 파일이나 Xilinx Constraints Editor를 통해서 Constraints를 입력할 수 있다.

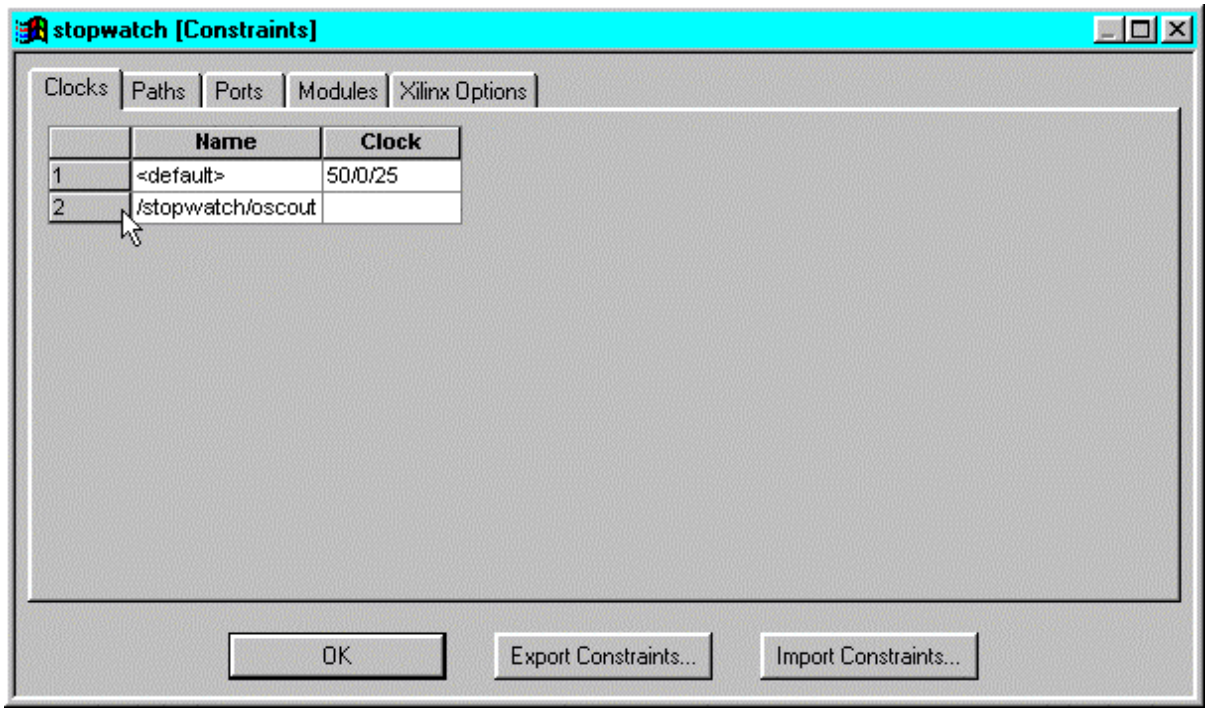
3. 합성을 시작하려면 **Run**을 클릭한다.

만약 Foundation Base Express를 사용한다면, implement your design부분으로 이동한다.
만약 Foundation Express를 사용한다면, Express Constraints GUI부분으로 이동한다. (바로 다음)

## Express Constraints GUI

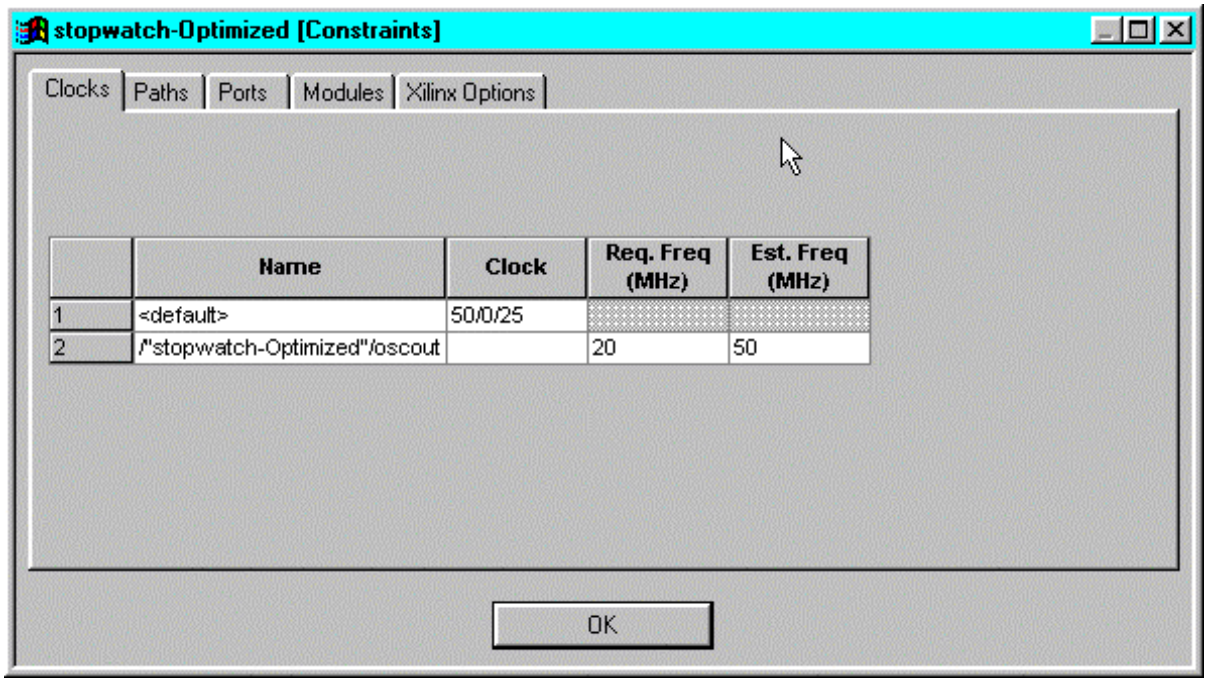
**Note:** Express Constraints GUI 는 Foundation Express 사용자들에게만 적용된다. 그것은 Foundation Base Express 사용자들에게는 적용이 되질 않는다.

디자인이 정밀하게 구성이 된 후에, Express Constraints GUI 가 열릴 것이다. 이 GUI 안에서, 여러분은 디자인의 timing constraints 을 조정할 수 있으며, global buffers 와 I/O registers, 그리고 pin location constraints를 설정하는 그러한, 구조적인 특징을 조절할 수 있다. 이 GUI에 대한 좀 더 자세한 정보에 대해서는 online help를 참조하길 바란다.




여러분이 이 GUI에 여러분 자신이 충분히 익숙해졌을 때, 오른쪽 위 상단 구석의 **X**를 한번 누름으로써 이 GUI가 닫힌다. 논리합성(Synthesis)은 최적화단계를 가지고 계속 진행되어질 것이다.

최적화단계가 완료되면, Time Tracker's GUI가 열리며, 여러분은 추정되어진 디자인의 성능결과를 볼 수 있을 것이다.



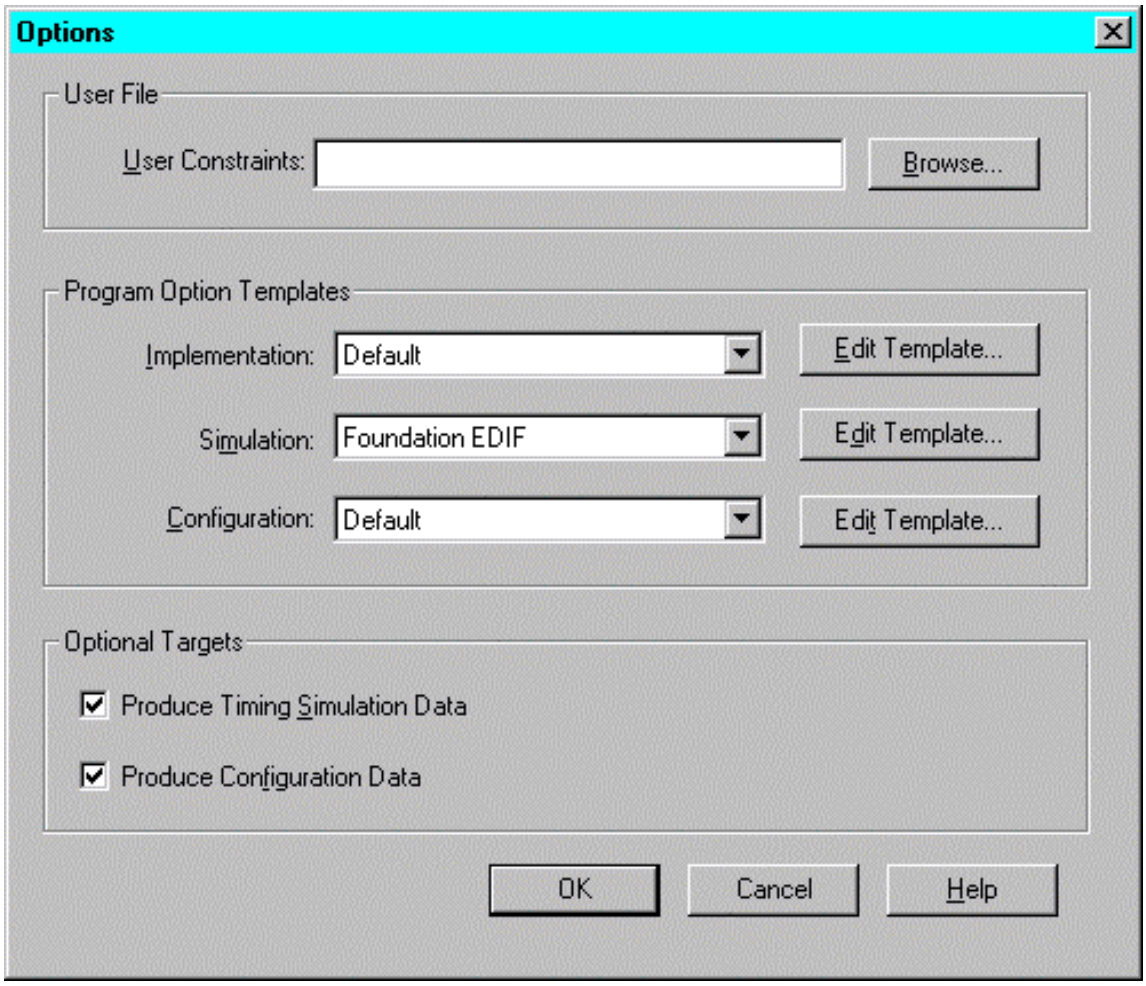

계속 진행할 준비가 됐으면 이 GUI를 닫는다. 논리합성이 완료된다.

## HDL Design 표현하기(Implementing the HDL Design)

Design implementation은 여러분의 디자인을 translating, mapping, placing, 그리고 routing 하는 처리 과정이다. Xilinx Implementation Tools 이런 처리 과정을 구성한다.

1. Implementation Options설정하기 위해, Project Manager로 부터 **Implementation's Options** 을 선택한다.

   Options dialog box 열리고 이 대화상자로부터 여러분은 다양한 design implementation options을 선택할 수 있다.




2. Optional Targets section아래에 있는 두개의 checkboxes 가 check 되어질 것이다.

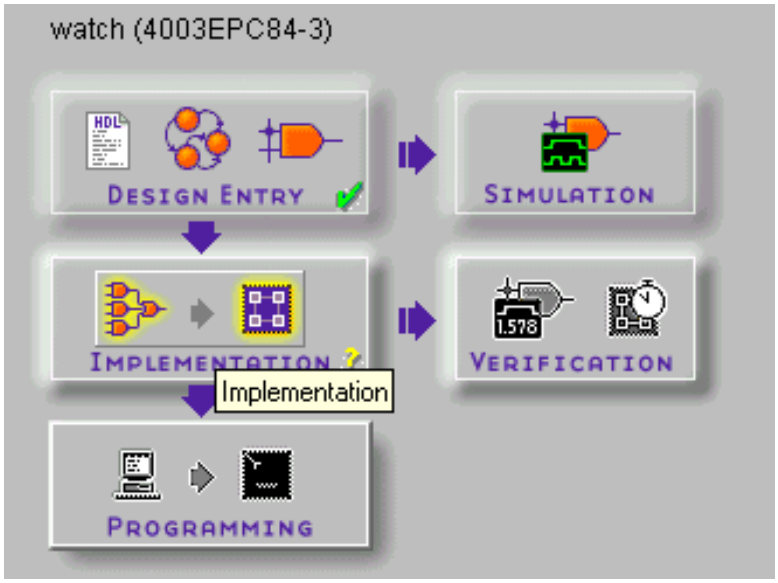
   Produce Timing Simulation Data checkbox는 tool이 timing simulation netlist를 만들 것임을 알려 준다. Produce Configuration Data checkbox는 tool이 후에 디바이스에 download 되어질 수 있는 하나의 .bit 파일을 만들 것 임을 알려 준다.

3. Options dialog box 안에 있는 **OK** 버튼을 한번 누른다.

4. Implementation을 시작하기 위해, Project Manager 의 Flow tab안에 있는 **Implementation** button을 한번 누른다.

   만약 tool이 netlist가 up-to-date가 되지 않았다는 경고를 여러분에게 알리면, netlist를 update 하기 위해 **Yes** 버튼을 한번 누른다.

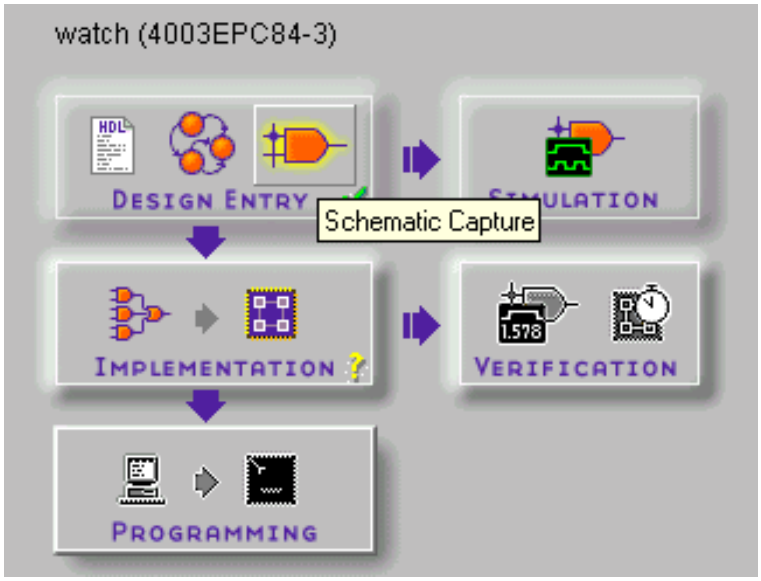
   이것은 Flow Engine을 불러서 implementation을 시작한다.


watch (4003EPC84-3)
HDL
DESIGN ENTRY
SIMULATION
IMPLEMENTATION
Implementation
1.578
VERIFICATION
PROGRAMMING

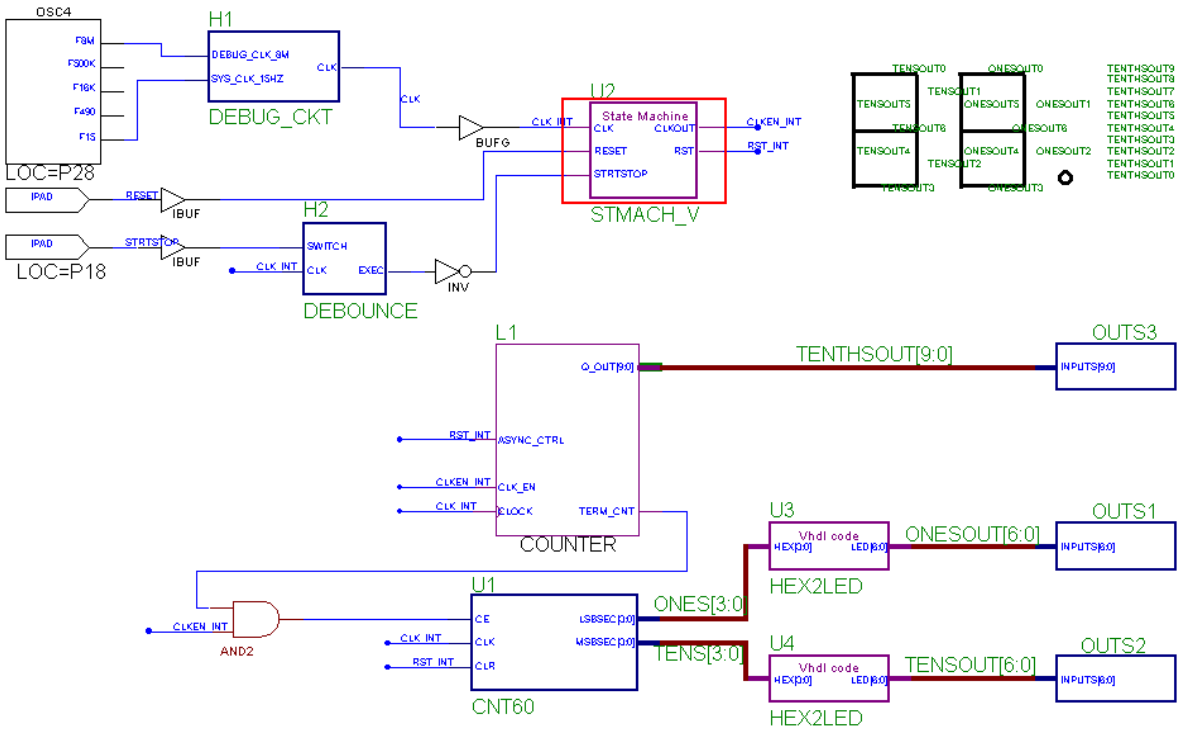
# Watch Schematic Design

Watch design은 계층적인 스키메틱 구조의 디자인이다. 이것은 top-level file이 lower-level design modules을 많이 가지고 있는 schematic file 임을 의미한다. 이러한 모듈 중 약간이 이 tutorial에서 표현되어질 것이다.

Schematic Capture tool을 열고 Watch schematic을 보기 위해, Flow diagram의 Design Entry section 안에 있는 **Schematic Capture** button 을 한번 누른다.




여러 개의 디자인 모듈이 아래 schematic에 놓여져 있다. 이러한 모듈은 design entry의 다양한 방법들로 만들어 졌다. 이러한 design entry 방법들 각각이 이 tutorial에서 간략하게 소개되어질 것이다.



**See Also**

CNT60 Schematic Module 보기 (Viewing the CNT60 Schematic Module )

HDL Design Macro보기 (Viewing HDL Design Macros )

LogiBLOX Design Module 보기 (Viewing LogiBLOX Design Modules )

State Machine Design Macro 보기 (Viewing State Machine Design Macros)

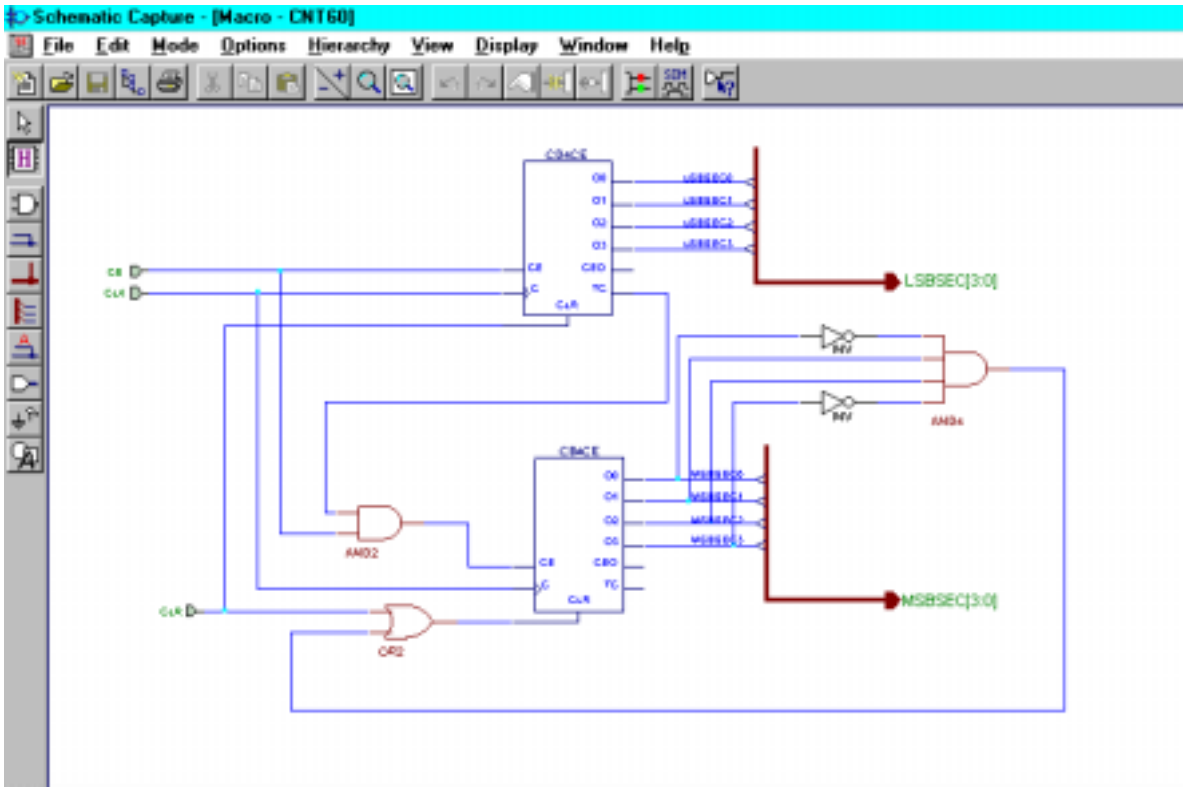
Xilinx Unified Libraries Component에 기초를 둔 디자인 매크로 보기 (Viewing Design Macros Based on Xilinx Unified Libraries Components )

여러분의 디자인을 지금 implement 하자.

## CNT60 Schematic Module 보기 (Viewing the CNT60 Schematic Module)

CNT60 module은 schematic-based module이다. 이것이 의미하는 것은 module에 해당하는 file이 schematic 이라는 것이다. schematic안의 Library component는 Xilinx Unified Libraries에서 나오는 것들이다.

1. design module안으로 들어 가기 위해, vertical tool bar에 있는 icon을 한 번 누르고, design module을 두 번 누른다.




2. macro 밖으로 나오기 위해서는, Hierarchy Push/Pop icon 을 다시 누르고, CNT60 schematic sheet상의 빈공간 아무 곳이나 두 번 누른다.

schematic-based macros 외에 다른 형태의 design macro들이 있다. 다른 형태의 design macro을 보기위해, 아래 여러분이 선택하는 Macro를 한 번 누른다.

HDL Macros

State Machine Macros

LogiBLOX Modules

Macros Based on Xilinx Unified Libraries Components

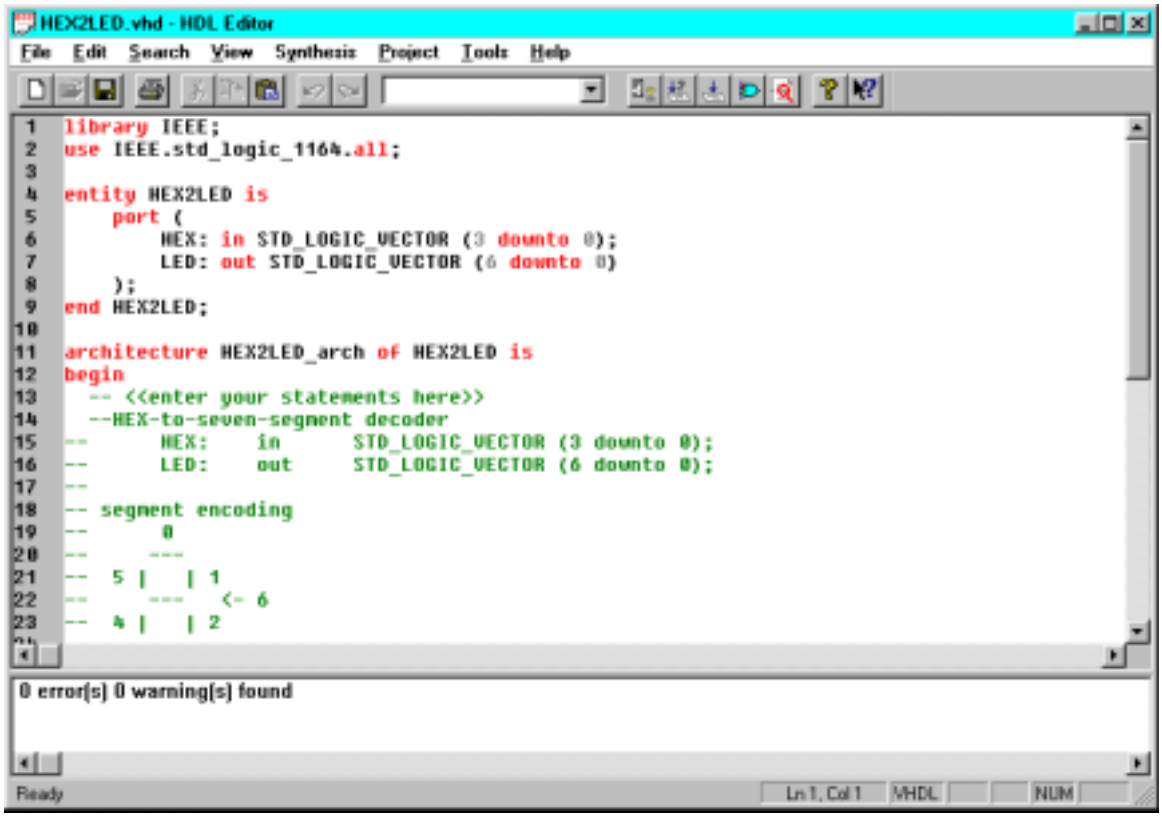
## HDL Design Modules 보기 (Viewing HDL Design Modules )

이 예제에서 사용된 hex2led module이 HDL에 기초를 두고 있다.

**Note:** Foundation은 세가지 다른 HDL 언어(VHDL, Verilog, ABEL)를 지원하고 있다. VHDL 과 Verilog은 Foundation Express 과 Foundation Base Express packages에서만 사용이 가능하다.

1. Watch schematic을 열고, vertical toolbar안에 있는 icon을 한 번 누르고, design module을 두 번 누르면 design module안에 들어 갈 수 있다.

   아래의 그림은 HDL Editor안의 hex2led VHDL file 을 보여 준다.

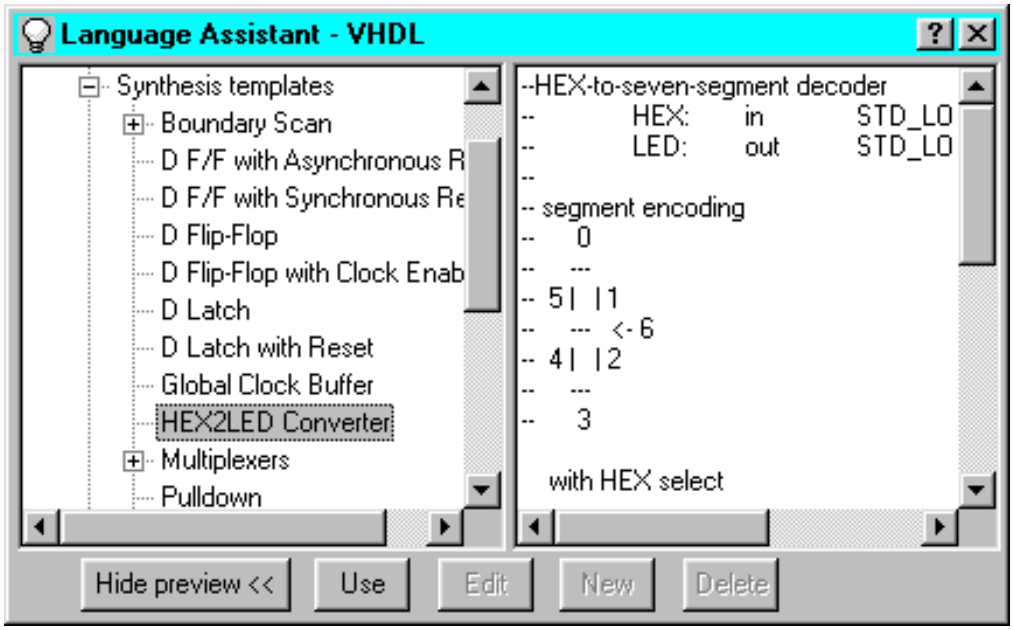



   HDL Editor는 HDL을 위한 쉬운 entry mechanism을 제공한다. HDL Editor는 keyword, comment, value을 구별된 색깔로 눈에 띄게 한다.

   더욱이, 여러분은 HDL Editor(Synthesis menu)로부터 디자인을 논리합성을 할 수 있을 뿐만 아니라 schematic상에 사용할 수 있는 macro symbol을 만들 수 있습니다.

2. **Tools메뉴에서 Language Assistant**를 선택한다.

   이것은 constructs 와 components에 공통으로 사용되는 HDL template를 보여 주는 Language Assistant가 열린다.




Xilinx architecture-specific templates가 또한 Synthesis Templates section에 보여진다. 위에 보여지는 HEX2LED HDL ffile은 Synthesis Template에서 직접 나온 것이다.

3. 상단 오른쪽 구석의 **X**를 한 번 누르면 HDL Editor를 빠져 나온다.

다른 형태의 design macro을 보기위해, 아래 여러분이 선택하는 Macro를 한 번 누른다.

Schematic-Based Macros

State Machine Macros

LogiBLOX Modules

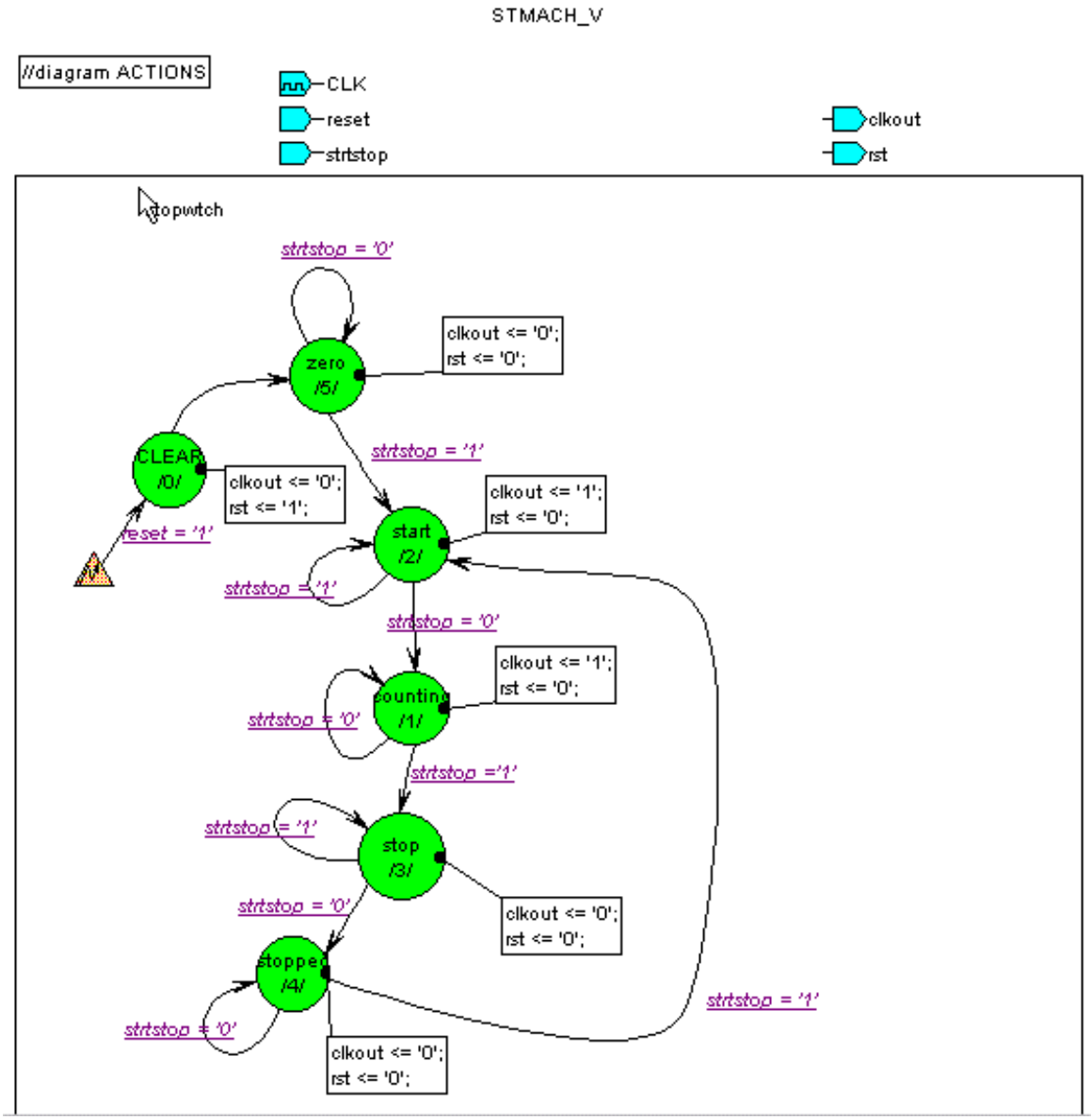
Macros Based on Xilinx Unified Libraries Components

## State Machine Design Modules 보기 (Viewing State Machine Design Modules )

다음 module은 state machine design에 기초를 두고 있다.

1. Watch schematic을 열고, vertical toolbar 안에 있는 icon을 한 번 누르고, design module을 두 번 누르면 design module안에 들어 갈 수 있다.

   아래의 그림은 State Editor안에 있는 Stopwatch state machine을 보여 준다.




표준 bubble diagram format을 사용하여 여러분의 state machine을 도형적으로 그리기 위해서는 State Editor을 사용하라. State Editor는 여러분이 그린 도형적인 표현을 취하여, VHDL, Verilog 또는 ABEL file를 만든다. 그러면 HDL code가 회로 합성되어진다. transition conditions과 actions은 target language의 문법으로 쓰여 진다.

2. target language 의 문법을 설정하기 위해, **Synthesis -> Configuration**을 선택한다.

   State Editor내에서, 여러분은 schematic상에서 사용할 수 있는 macro symbol을 만들 수 있다.

3. 상단 오른쪽 구석의 **X**를 한 번 누르면 State Editor를 빠져 나온다.

다른 형태의 design macro을 보기위해, 아래 여러분이 선택하는 Macro를 한 번 누른다

Schematic-Based Macros

HDL Macros

LogiBLOX Modules

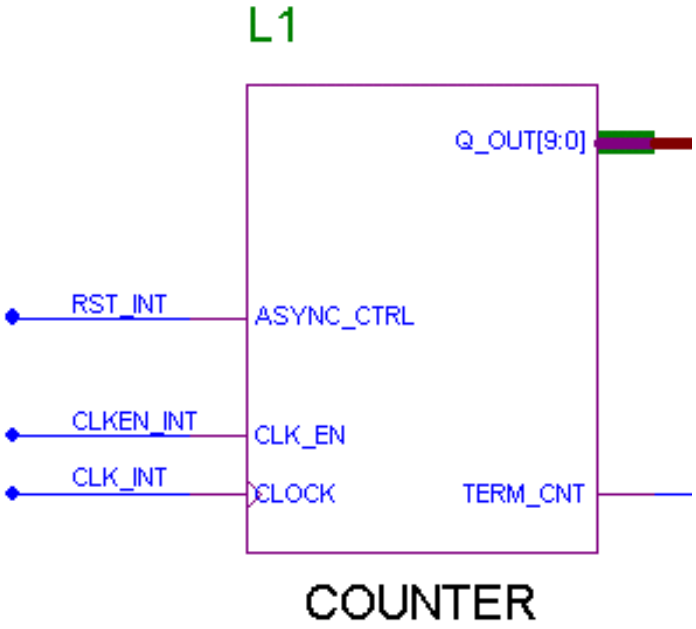
Macros Based on Xilinx Unified Libraries Components

## LogiBLOX Design Modules 보기 (Viewing LogiBLOX Design Modules)

다음 module은 LogiBLOX module이다. LogiBLOX은 여러분 자신의 library component를 만들 수 있는 graphical interactive tool이다. LogiBLOX은 Fast Carry Logic, CLB RAM, ROM과 같은 그러한 Xilinx-specific architectural features의 장점을 취할 수 있는 architecture-optimized component를 만들 수 있다.

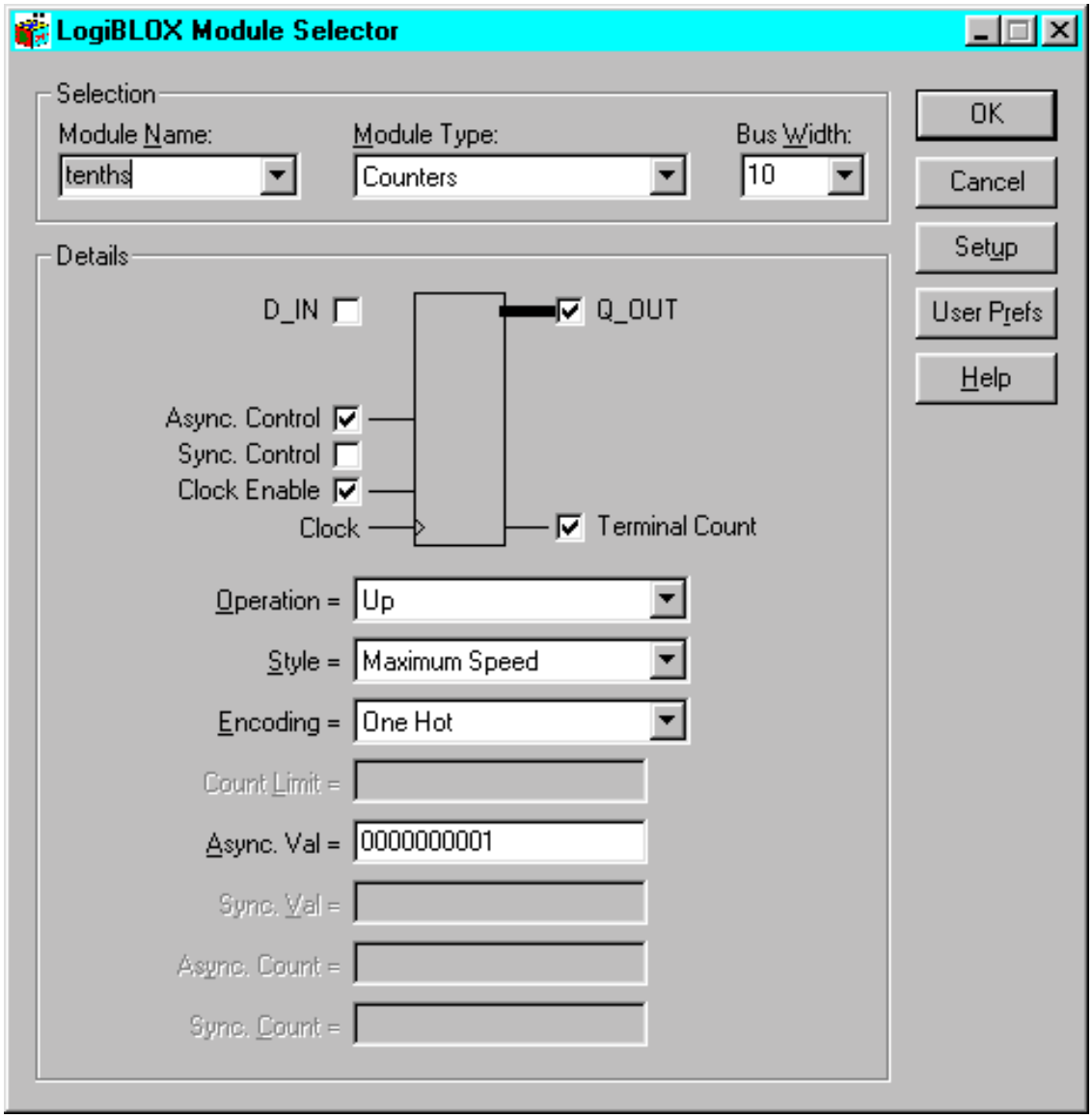
1. Watch schematic 열어 놓고, Tenths라 불리는 L1 LogiBLOX counter를 위치시킨다.

   여러분은 schematic상에 놓여진 LogiBLOX module안으로 들어 갈 수가 없다. 왜냐하면 해당되는 netlist가 binary netlist이기 때문이다. 이 netlist는 implementation시 합병되어 사용되어 진다.




2. LogiBLOX Module Selector GUI (where you create the LogiBLOX module)를 열기 위하여, Schematic Capture tool로부터 **Options → LogiBLOX** 를 선택한다.

   이 GUI 안에서, 여러분은 module name, type, 그리고 function을 선정한다.




3. Tenths component를 선택하기 위해, Module Name field에 있는 pull-down menu를 사용하라.

   이 단계는 Tenths component를 만들기 위한 선택 사항들을 보여 준다. 여러분은 지금 Tenths module이 10-bit one-hot encoded counter라는 것을 알 수 있을 것이다. LogiBLOX Module Selector GUI에서 module이 만들어 질 때, symbol이 자동적으로 만들어지고 여러분은 만들어진 symbol을 schematic상에 위치시킬 수 있다.

4. LogiBLOX Module Selector를 닫기 위해, **Cancel** 을 한 번 누른다.

다른 형태의 design macro을 보기위해, 아래 여러분이 선택하는 Macro를 한 번 누른다.

Schematic-Based Macros

HDL Macros

State Machine Macros

Macros Based on Xilinx Unified Libraries Components

## Xilinx 공통 라이브러리에 기반한 설계모듈 보기 (Viewing Design Modules Based on Xilinx Unified Libraries Components)

추가적으로 당신이 사용하는 모듈을 만들 때 Xilinx의 통일된 라이브러리를 사용할 수 있다. 일반적으로 대부분의 회로와 회로에 사용되는 마크로는 컴퍼넌트 형태로 만들어진다.

Xilinx의 통일된 라이브러리에서는 다음과 같은 컴퍼넌트를 얻을 수 있다.

AND 나 OR 같은 단순한 컴퍼넌트.

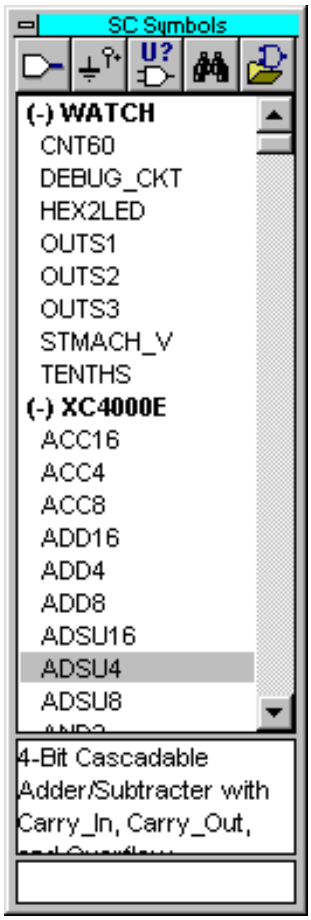
adders, counters, multiplexer등과 같이 좀더 복잡한 함수의 컴퍼넌트.

컴퍼넌트는 RAM, 내부 oscillators, 그리고 global Set/Reset 처럼 Xilinx의 구조적인 특성을 가질 수 있다.

Watch 회로를 보면 서로 다른 몇 개의 모듈을 볼 수 있다. Watch회로에서 AND2,OSC4,IPADs, IBUFs는 Xilinx에서 제공되는 통일된 라이브러리의 일부이다.

1. icon 을 클릭하면 SC Symbols 툴 박스가 아래와 같이 나타날 것이다. SC Symbols 박스에는 이 프로젝트에서 사용 가능한 모든 라이브러리가 보여진다.

2. 스크롤바을 이용하여 모든 컴퍼넌트를 찾아 볼 수 있다.




3. 컴퍼넌트를 이용 할 때는 컴퍼넌트를 선택한 뒤 회로영역으로 이동하여 클릭하면 된다. 컴퍼넌트에 대한 자세한 설명은 *Xilinx Libraries Guide*에 기술되어 있다.

4. 회로를 닫고자 할 때는 우측상단의 **X** 표시를 클릭하면 된다.

**Note**: If the tool prompts you to save changes, **DO NOT** save the changes. Click **No**.

디자인 툴을 바꾸고자 할 때는, 다음과 같은 디자인 방법으로 넘어간다.

Schematic-Based Macros

HDL Macros

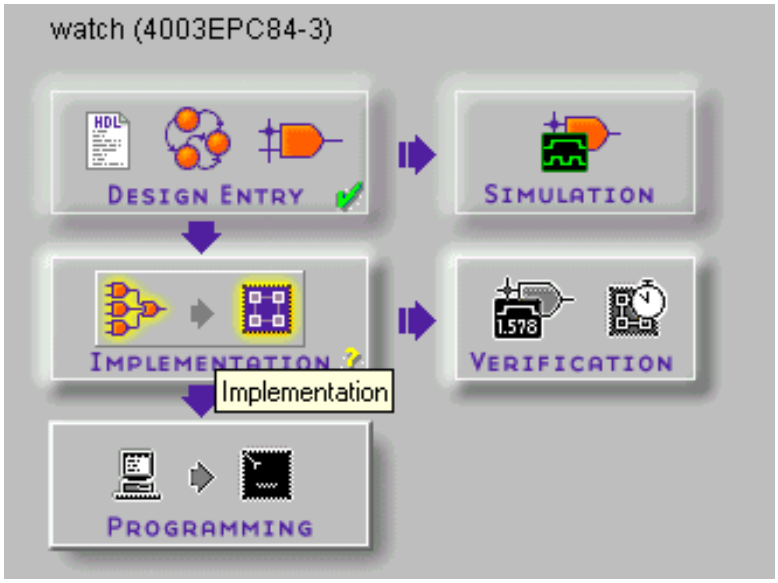
State Machine Macros

LogiBLOX Modules

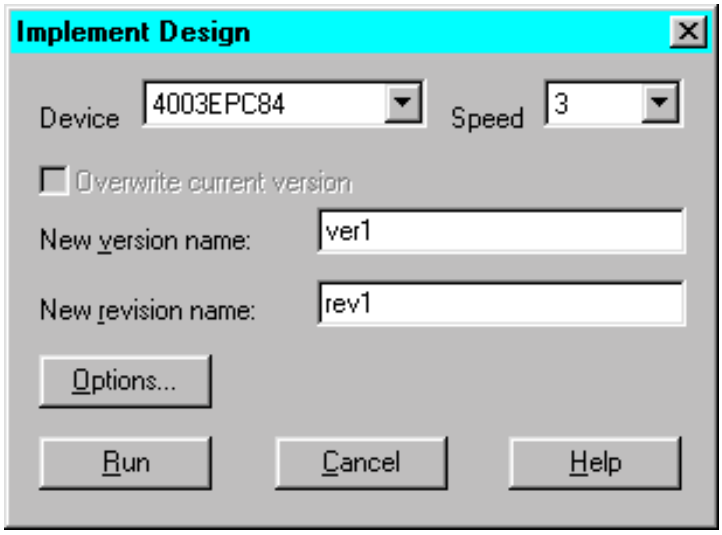
## Implementing the Schematic Design

Design implementation은 translating, mapping, placing과 routing으로 이루어진다. Xilinx Implementation Tools (or the Flow Engine) 은 이 과정을 수행한다.

1. Implementation의 시작은 Foundation Project Manager에서 **Implementation** 버튼을 클릭하면 된다. 만약 툴에서 넷 리스트가 업데이트 되지 않았다는 경고가 나오면, **Yes**를 클릭하여 넷 리스트를 업데이트 하면 된다.

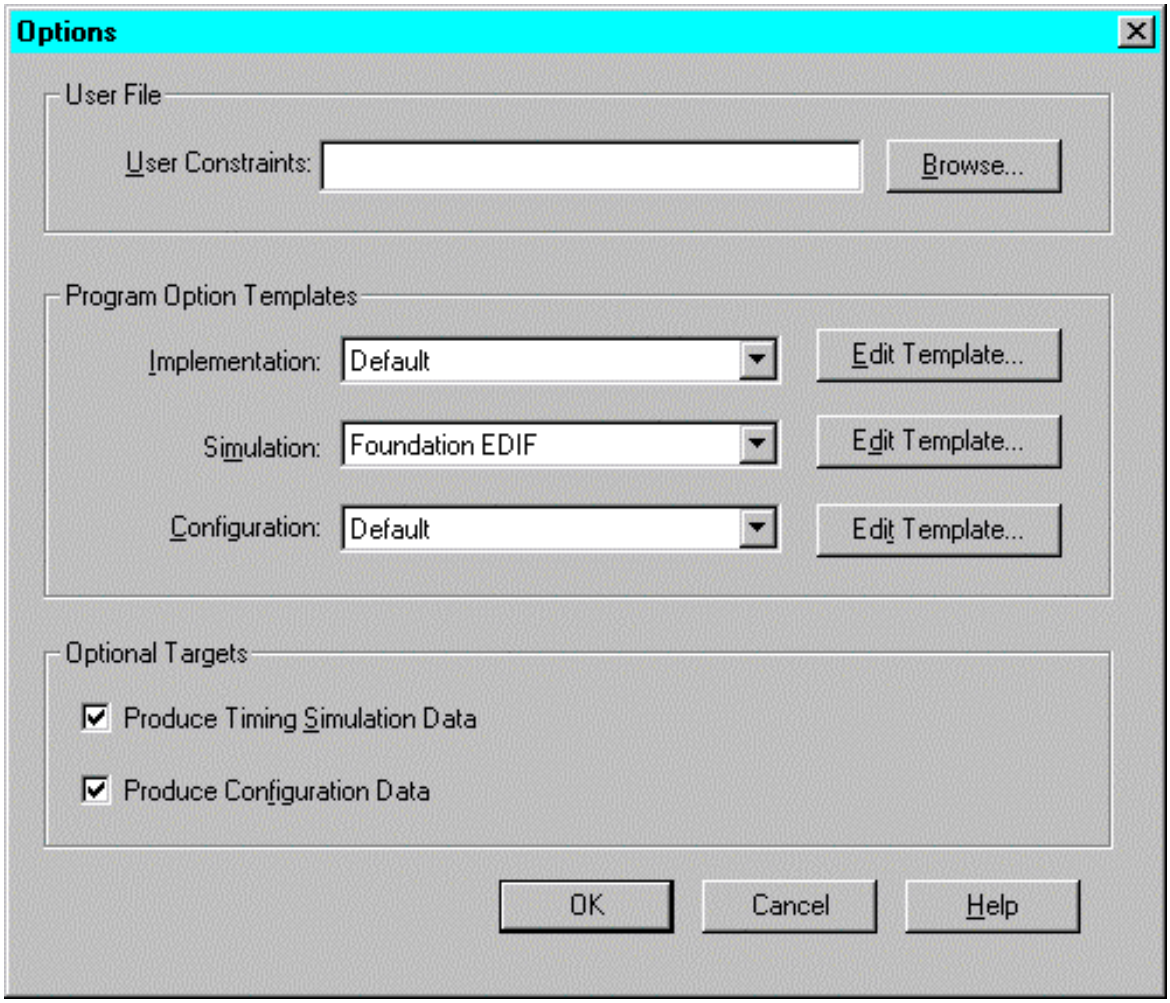



Implement Design이 다음과 같이 나타난다. 여기에서 당신이 원하는 디바이스나 스피드, 버전, 리비젼등을 설정할 수 있고 Options을 이용할 수도 있다. 여기에서는 디바이스가 XC4003EPC84-3로 설정되어 있다.

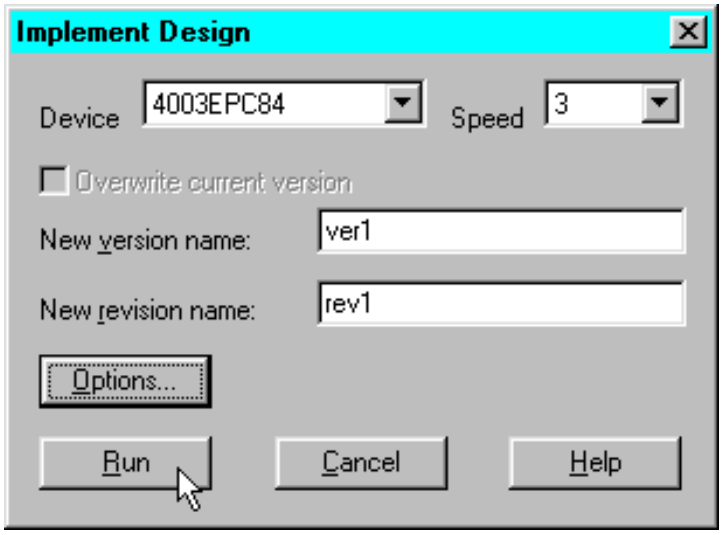



2. **Options** 을 클릭하면 다음과 같은 박스가 나타난다.

   이 박스에서 사용자가 원하는 implementation options을 지정할 수 있다.

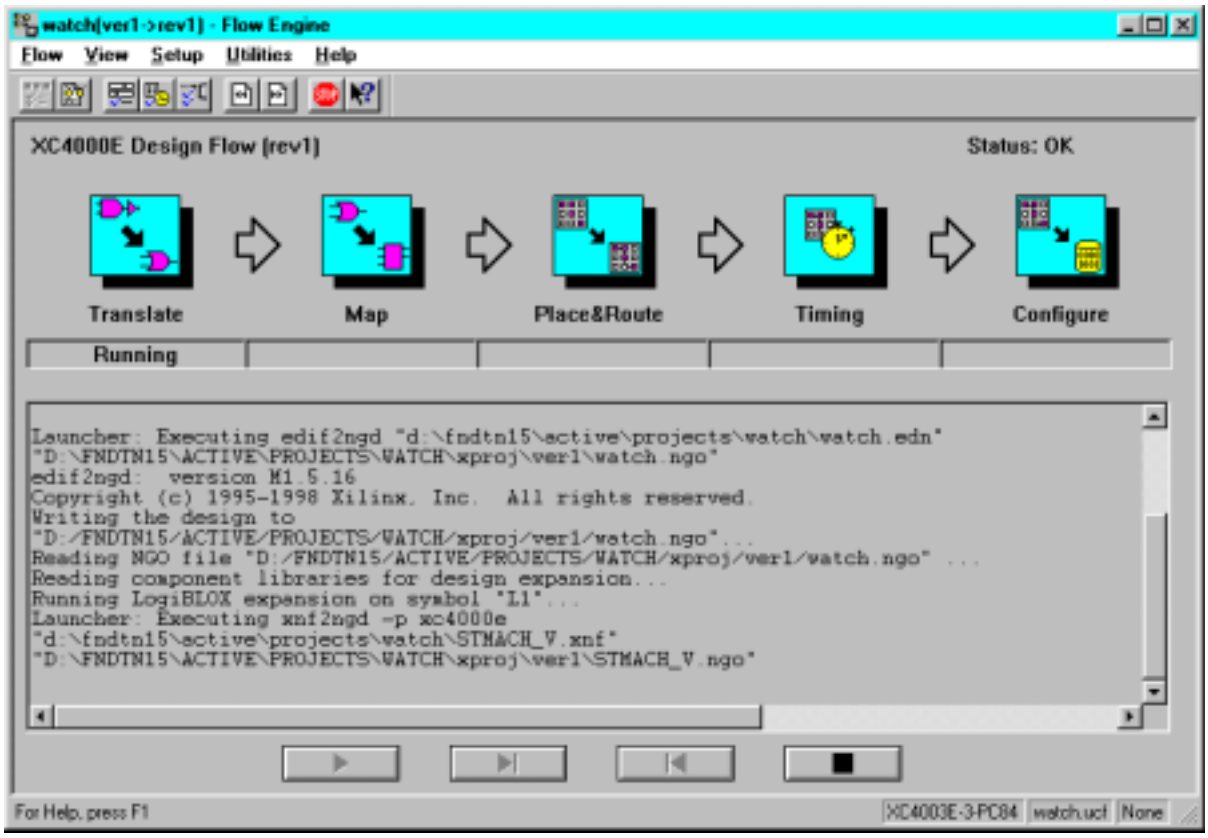



3. Optional Targets을 보면 두개의 박스가 모두 클릭 되어 있을 것이다. Produce Timing Simulation Data는 타이밍 시뮬레이션을 위한 넷 리스트를 만드는 것이고 Produce Configuration Data는 차후에 디바이스에 다운로드 할 수 있는 .bit 파일을 만드는 것이다. 또 다른 옵션들은 상단의 options template을 이용할 수 있다.
4. **Edit Template**를 클릭하여 다른 형태의 옵션을 줄 수도 있다.
5. Options의 박스에서 **Ok**을 클릭한다.
6. Implement Design에서 **Run** 을 클릭하면 Implementation이 시작된다.


Implement Design
Device
4003EPC84
Speed
3
Overwrite current version
New version name:
ver1
New revision name:
rev1
Options...
Run
Cancel
Help

## Flow Engine

implementation이 시작되면 Flow Engine 화면이 나타난다. Flow Engine은 디자인을 implementation 한다. 디자인의 흐름과 상태는 그래픽적으로 나타나며, 진행되는 상태에 대한 기술은 하단의 콘솔 박스에 표시된다.



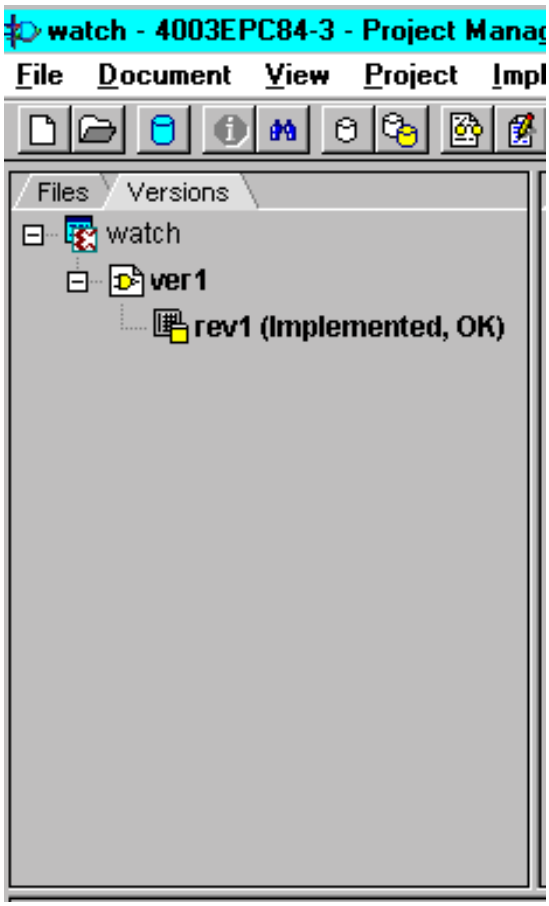

implementation이 끝나면, Flow Engine은 자동적으로 닫힌다. Implementation이 성공적으로 또는 실패로 끝났다는 화면이 나타날 것이다. 추가적으로 Implementation의 상태가 프로젝트 메니저의 하단에 있는 콘솔박스에 나타난다.

## Versions Tab

파운데이션 프로젝트 메니저는 프로젝트를 버전과 리버젼으로 관리한다. 버전은 입력되는 넷 리스트에 변화가 있을 때 다시 만들어진다. 즉 회로상의 로직이 더해지거나 빠지거나 또는 HDL 소스가 수정되거나 하는 등의 소스 디자인에 변화가 발생할 때 새로운 버전으로 만들어 진다. 리버전은 주어진 버전에서 새로운 옵션이나 새로운 P&R레벨을 이용하여 implementation을 실행하였을 경우에 새로운 리버젼이 만들어 진다. 파운데이션은 현재 선택된 프로젝트의 각 리버젼 단위로 파일이나 리포트 등을 관리한다.

**Note**: 디자인 소스가 각 버전별로 저장되어 있지는 않다. 단지 각 버전의 넷 리스트와 컴파일이 된 결과만이 저장된다.

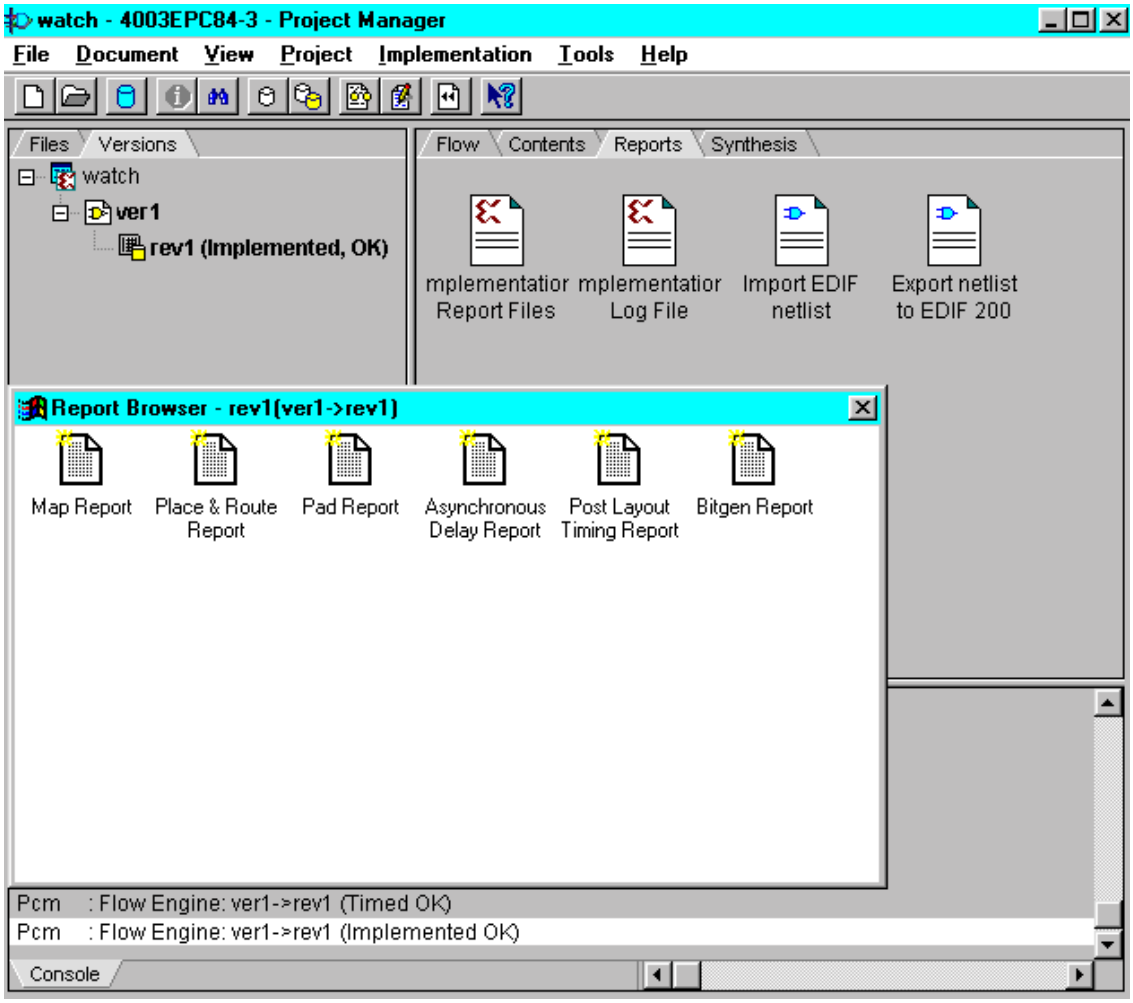
프로젝트 메니저의 버전 탭을 누르면 모든 버전과 리버젼이 계층적으로 나타난다. 가장 최근의 리버젼이 굵게 표시된다.



## Implementation 보고서 살펴보기 (Reviewing Implementation Reports)

다음과 같이 주어진 리버젼에 대한 implementation 리포트를 볼 수 있다.

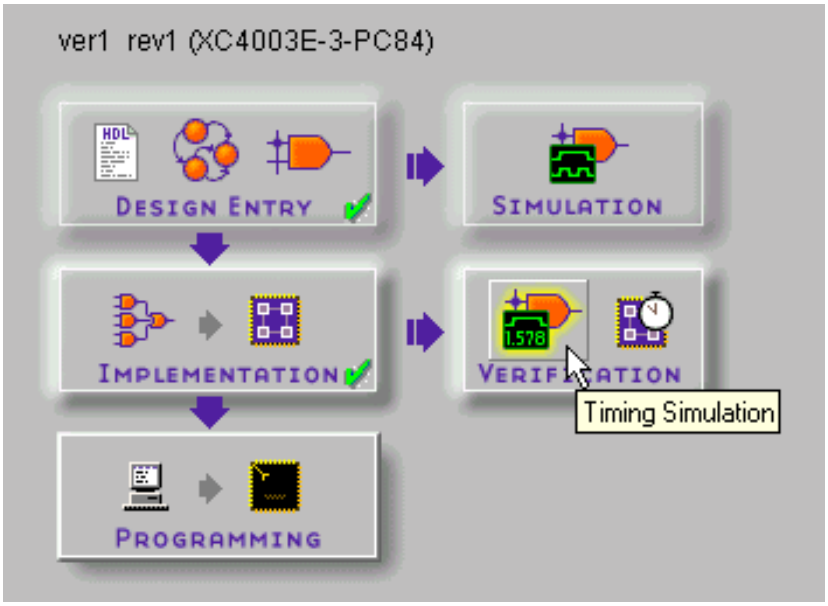
1. 필요한 리버젼을 선택하고 프로젝트메니저의 우측에 있는 Reports tab을 클릭한다.
2. Implementation Report Files을 선택하고 더블 클릭한다. 이 과정을 거치면 모든 implementation report들이 나타난다.
3. 필요한 리포트를 선택하여 보면 된다.



## 타이밍 시뮬레이션 수행 (Performing Timing Simulation)

가장 타이밍이 좋지 않은 경우의 타이밍 정보를 이용하여 타이밍 시뮬레이션을 이용하면 디자인을 수정할 수 있다.

Verification section의 timing simulation을 클릭하면 타이밍 시뮬레이터를 불러낼 수 있다. 이것은 라우팅된 넷 리스트를 읽어서 파운데이션 로직 시뮬레이터를 동작 시킨다.




사용자는 implementation하기 전과(Functional Simulation) ,implementation한 이후에(Timing Simulation) 시뮬레이션을 사용할 수 있다. 이 두 가지는 다음 부분을 제외하고는 모두 동일하게 사용할 수 있다.

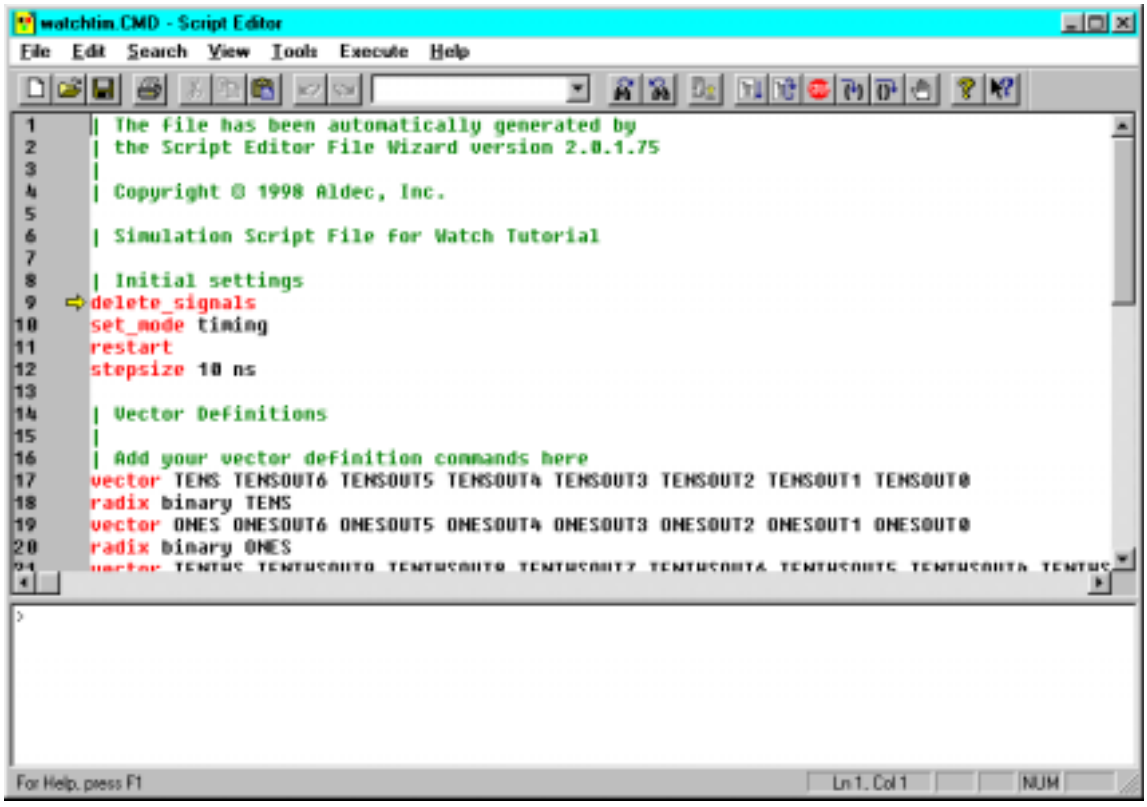
펑션 시뮬레이션은 P&R하기 이전의 넷 리스트를 기본으로 한다. 즉 타이밍 정보를 전혀 고려하지 않고, 순수한 함수적인 로직의 특성만을 시뮬레이션 한다.

타이밍 시뮬레이션은 딜레이를 기본으로 한 타이밍 정보를 가진 넷 리스트,즉 P&R된 이후의 넷 리스트를 가지고 시뮬레이션을 한다.

## Script 파일을 사용한 시뮬레이션 (Script File Method of Simulation)

파운데이션에는 매우 다양하게 시뮬레이션 입력을 정의하는 방법을 지원한다. 하지만 이 매뉴얼에서는 스크립터를 이용하여 시뮬레이션 하는 방법 만을 기술 한다. 시뮬레이션의 보다 기초적인 다른 사용방법은 *"Foundation Series Quickstart Guide 1.5"*를 참조하면 된다.

1. 시뮬레이터에서 **Tools Script Editor**를 실행한다.
2. 스크립트 파일의 프롬프트가 나타나면 watchtim.cmd을 선택한다.
3. **OK**를 클릭.




스크립트 파일은 시뮬레이션 입력이나 출력하고자 하는 파형과 진보적인 시뮬레이션을 포함할 수 있다. 스크립트 에디터는 텍스트 파일을 보거나 수정할 수 있고 예약된 단어에 대하여는 빨간색, 부가기술에 대하여는 녹색으로 표현된다.

4. 시뮬레이션을 실행 할 때는 **Execute-Go**를 선택한다. 스크립트 파일이 시뮬레이션 될 때 그 결과는 시뮬레이터의 Waveform Viewer에 나타나므로 확인 하면 된다.
5. 출력된 결과 파형을 가지고 회로를 검사하여 본다.
6. 시뮬레이터를 닫는다.
7. 스크립터 에디터를 닫는다.

## Downloading the Design

스톱워치 디자인을 하드웨어 디버거를 이용하여 자이링스 데모보드의 XC4003E에 다운로드 할 수 있다. 하드웨어 디버거 메뉴로 들어가기 위하여 **Programming** 버튼을 클릭한다.

동일하게 **Tools->Device Programming->Hardware Debugger**을 이용하여서도 동작 시킬 수 있다.

좀더 자세하게 다운로드에 대하여 알고자 하시면 *Foundation Series Quickstart Guide 1.5*을 참고하시면 된다.

**이것으로 기본적인 사용자 가이드는 마무리가 된다.**