

메모 (Technical)

Document No.: MJL-LD-AN-9
 Author: 양창우 [cwyang@mjl.com]
 Version: 1.0
 Date: 2000년 9월 26일
 Subject: **MAX+plus II 에서 PLL 쓰기 (ACEX 1K 를 중심으로...)**

ALTERA Device 에서 지원되는 PLL 의 기능은 Clock Lock 과 Clock Boost 이다... Clock Lock 은 말그대로 Global Clock Pin 으로 입력되는 신호의 Clock Skew 를 최대한 억제하여 내부 Device 에 신호를 공급하기 위하여 사용을 하는것이고, Clock Boost 는 입력신호를 Multiply 하여 Device 내부로 공급할때 사용을 한다...

PLL 이 지원되는 ALTERA Device 는 APEX 20KE, FLEX 10KE, ACEX 1K/2K 가 있는데, 그중 ACEX 1K 시리즈는 PLL 의 여러기능중 Clock Lock 과 Clock Boost 를 지원하고, 그중 Clock Boost 는 2X 만을 지원한다... 또한 Speed Grade -1 과 -2 에서만 PLL 기능을 사용할수 있다...

아래의 내용은 ACEX 1K 에서 2x Clock Boost 기능을 VHDL 을 사용하여 활용하는 방법에 관한 것이다...

그림 1 은 여기서 예로들 VHDL Coding 을 Schematic 으로 표시한 것이다...

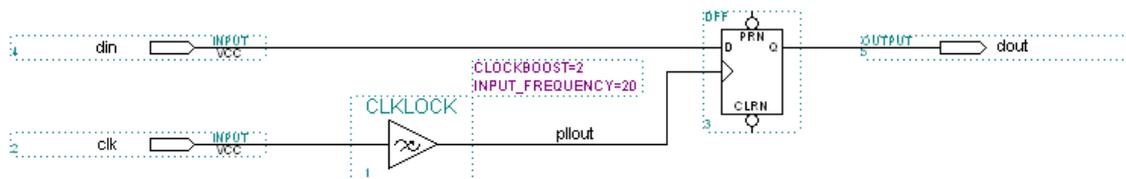


Figure 1 Sample Schematic

위의 그림 1 을 VHDL 로 Coding 하면 다음과 같다...

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;

LIBRARY altera;
USE altera.maxplus2.ALL;

entity pll is
port
```

```

        ( din : in std_logic;
          clk : in std_logic;
          dout : out std_logic
        );
end pll;

architecture pll_arch of pll is

signal pllout : std_logic;

begin

u1: clklock
    GENERIC MAP (clockboost => 2, input_frequency => "20")
    port map (inclk => clk, outclk => pllout);

process (pllout)
begin
    if (pllout'event and pllout='1') then
        dout <= din;
    end if;
end process;

end pll_arch;

```

VHDL Source 를 살펴보면, PLL 기능은 ALTERA 에서 제공하는 Device Specific Function 이므로 Block Instantiation 기법을 이용하여 기능구현을 한다... 그러므로 Architecture 구문내에 Component 선언을 하여야한다... 하지만 이미 ALTERA 에서 이러한 기능을 제공하기위해 ALTERA 라는 이름의 Library 를 제공하므로

```

LIBRARY altera;
USE altera.maxplus2.ALL;

```

이 두줄로 간단히 해결이 된다... (일부러 Component 선언을 할 필요가 없다...) 참고로 ALTERA Library 중 Maxplus2 Package 내에는 CLKLOCK 라는 이름으로 PLL Component 가 선언이 되어있다...

```

component clklock
    generic ( input_frequency : STRING;
              clockboost : INTEGER);
    port ( inclk: in STD_LOGIC;
           outclk: out STD_LOGIC);
end component;

```

사용자는 이미 선언되어 있는 내용을 기반으로 VHDL Source 내에 Port Map 만 기술해주면 된다...

```

u1: clklock
    GENERIC MAP (clockboost => 2, input_frequency => "20")
    port map (inclk => clk, outclk => pllout);

```

여기서 한가지 주의깊게 보아야 할부분이 있는데, 바로 GENERIC MAP 부분이다... 이부분에 기술된 값에 따라 PLL 의 동작특성이 결정된다... PLL 의 사용시에는 clockboost 와 input_frequency 라는 두가지 변수가 사용된다... 첫번째로 clockboost 에는 1 과 2 값이 올수 있는데, 1 을 대입하면 1x inclk 즉 Clock Lock 기능으로 사용된다는 뜻이고 2 를 대입하면 2x inclk 로서 Clock Boost 기능을 사용한다는 뜻이 된다... 위의 Source 에서는 Clock Boost 로 사용하였다... 두번째로 input_frequency 는 CLKLOCK 의 inclk 에 입력되는 CLK 값을 MHz 단위로 표현한 문자열이다... 문자열이라고 하는데, 반드시 쌍따옴표로 묶어주어야 Error 가 생기지 않는다... 위의 예에서는 Input 주파수를 20MHz 라고 가정하였다...

나머지 부분은 일반적인 D type Flip-Flop 을 기술한 부분이다... CLKLOCK 의 출력신호(pllout)를 가지고 Sequential Process 구문을 작성하였다...

위의 VHDL Code 를 가지고 MAX+plus II 9.64 에서 ACEX 1K30TC144-1 로 Device 를 선정하고 Compile 과정을 수행하여 아래 그림 2 와 같은 Timing Simulation 결과를 추출하였다...

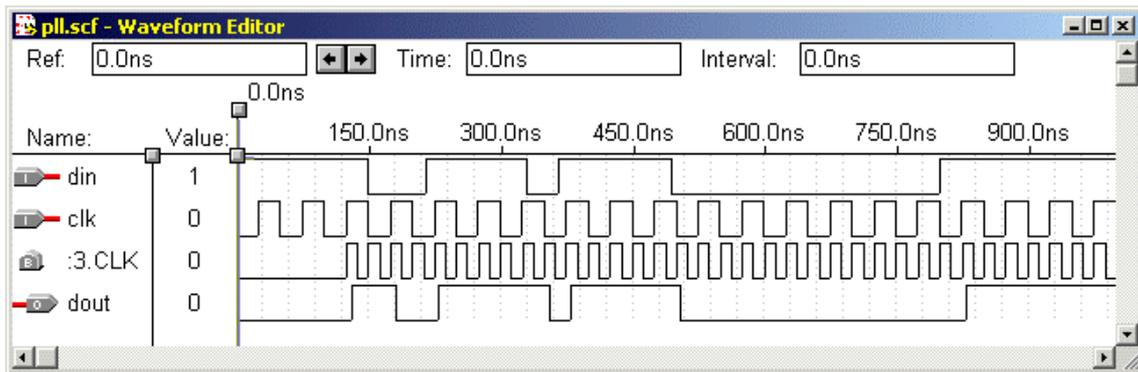


Figure 2 PLL (CLKLOCK)의 Timing Simulation 결과

화면의 Netname 중 :3.CLK 는 그림 1 및 VHDL Source 에서 기술한 pllout signal (CLKLOCK 의 outclk pin)의 내부 Node Name 이다... 여기서 한가지 주의할 사항은 clk 에 신호가 들어오는 즉시 2x clk 가 되는것이 아니라 어느정도의 interval 을 가지고 실제 동작을 한다는 점이다... (바로 동작되기를 바라는건 말도안되는 너무 무리한 요구임을 말한다...)

참고문헌

- MAX+plus II On-Line Help 중 **Megafunctions/LPM** 의 **clklock** (Phase-Locked Loop)

Revision History

- 2000-9-26 - Ver 1.0: Initialize Release...