

TI DSP, MCU 및 Xilinx Zynq FPGA 프로그래밍 전문가 과정

2018-05-30 (64 회차)

강사 - Innova Lee(이상훈)

gcccompil3r@gmail.com

학생 - 정유경

ucong@naver.com

- Vivado로 Hardware 설계하기 -

1. home 에 zynq_zybo > lab6 > hardware 폴더를 만든다
2. vivado 구동
바탕화면에 바로가기 안 만든 사람들은 직접 구동시키자
Xilinx > Vivado > 2017.1 > bin > ./vivado
3. 새로운 프로젝트를 생성한다
Name: 'driver_lab'
Directory: home > 내이름 > lab7 > hardware
Create project subdir 체크
4. RTL Project
Do not specify sources 체크
5. Default Part에서 Board명 선택
'Zybo Z7-10' 클릭
6. Create Block Design
'system'
7. Diagram에서 Add IP 'ZYNQ7 Processing Sys' 추가
Run Block Automation > 그냥 OK
ZYNQ7 더블 클릭 > MIO config > IO Peripherals > USB0 체크 해제

Clock Config > FCLK_CLK0 / (Clock Source) IO PLL / (Req Freq)100 체크
확인

8. Diagram에서 Add IP 'AXI GPIO' 추가
왼쪽 아래 Block Properties > 이름변경 'led_pin'
더블클릭 > IP Config > GPIO Width 1 설정
Run Connection Automation > GPIO와 S_AXI체크
Interface > Custom으로 변경
9. Run Connection Automation > OK
블록 다이어그램을 보기좋게 정리하고 저장한다
10. Address Editor 탭에서
Offset Address 0x4120_0000확인
11. Diagram 탭에서
Validate Design(F6) 클릭
12. 왼쪽 창에서 Design Sources > system 우클릭 > Generate Output
Products
Syntesis 옵션: Global
Run Settings: on local host / jobs 4
13. 다시 우클릭 > Create HDL Wrapper
Let Vivado manage Wrapper and 자동 업데이트 선택

14. Add Sources

add or create constraints > Create File

Name: 'lab(마음대로)' > Finish

15. 방금 만든 파일에 핀 설정을 작성

```
set_property -dict { PACKAGE_PIN V15 IOSTANDARD LVCMOS33 }  
[get_ports {led_pin}];
```

16. Run Synthesis > Run Implementaion

17. 하단 창에서 I/O Ports > GPIO_8845 > gpio_rtl_tri_io[0]를
V15 / LVCMOS33 으로 변경

18. Design 바뀌었으므로 아까 작성한 constraints가서 Reload

(set_property IOSTANDARD LVCMOS33 {get_ports {gpio_rtl_tri_io[0]}}; 가
자동으로 추가되었음을 확인

19. Generate Bitstream

20. File > Export HW

include Bit Stream 체크 > OK

처음에 설정한 Hardware경로에 총 9개 파일 생성된걸 확인한당

```
(cd ~/zynq_zybo/lab6/hardware/driver_lab)
```

- Peta Linux 준비하기 -

1. Peta Linux 생성

cd ~/zynq_zybo/lab6 (hardware의 상위폴더)에서 다음을 입력
petalinux-create -t project -s ~/fpga_test/ZYBO_petalinux_v2015_4.bsp
(bsp파일 경로 찾아서 입력)
lab6 밑에 두개의 폴더(hardware, ZYBO_petalinux_v2015_4) 확인

2. New Apps 생성

cd ~/zynq_zybo/lab6/ZYBO_petalinux_v2015_4
petalinux-create -t apps --name gpio-dev-mem-test
6개 파일 생성

3. Software 생성

cd ~/zynq_zybo/lab6
petalinux-create -t project -n software --template zynq
3개 파일 생성

4. 리눅스 시스템 환경설정

cd ~/zynq_zybo/lab6/hardware/driver_lab/driver_lab.sdk
petalinux-config --get-hw-description -p ~/zynq_zybo/lab6/software
설정할것 없으므로 바로 Exit > No

5. 리눅스 빌드

cd ~/zynq_zybo/lab6/software

petalinux-build

6. BOOT.BIN 생성

cd ~/zynq_zybo/lab6/software/images/linux
petalinux-package --boot --fsbl zynq_fsbl.elf --fpga
~/zynq_zybo/lab6/hardware/driver_lab.runs/impl_1/system_wrapper.bit --
u-boot

7. Bootgen 에러

구글에 검색 (경로주의) 하여 다음을 실행
source ~/Xilinx/SDK/2017.1/settings64.sh
source ~/Xilinx/Vivado/2017.1/settings64.sh

8. 6번 을 재실행하여 에러 메시지가 없어진 것 확인
Binary is Ready!!

<FIN>