

Modul 5

Pemrograman Dasar PS

1. Tujuan

- Memahami SDK 2016.4 dan membuat project baru.
- Membuat user application.

2. Materi

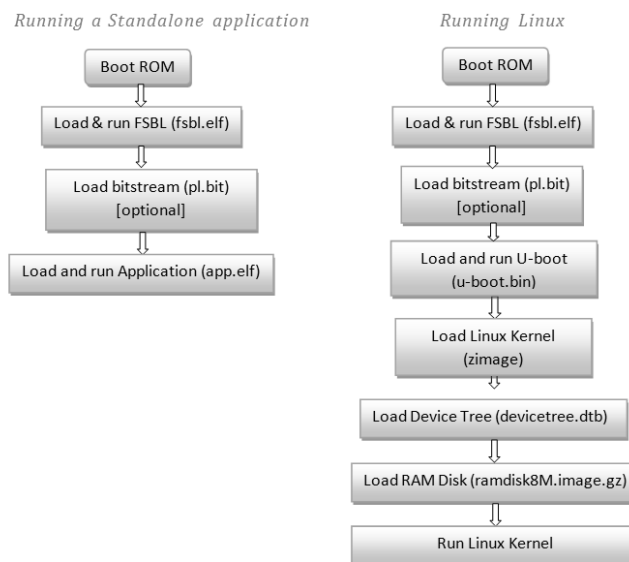
a. Xilinx SDK

Xilinx SDK merupakan software development kit yang digunakan untuk membuat program untuk Zynq PS. Xilinx SDK menggunakan IDE berbasis Eclipse. Pada Xilinx SDK terdapat library- library untuk driver IP core Xilinx. Xilinx SDK juga digunakan untuk membuat FSBL.

b. FSBL

First Stage Boot Loader merupakan bootloader yang memiliki fungsi untuk men-load file bitstream FPGA (PL) dan user application atau 2nd stage boot loader. Gambar 1 menggambarkan proses boot dari Zynq. Terdapat dua jenis boot yaitu untuk standalone application dan running linux.

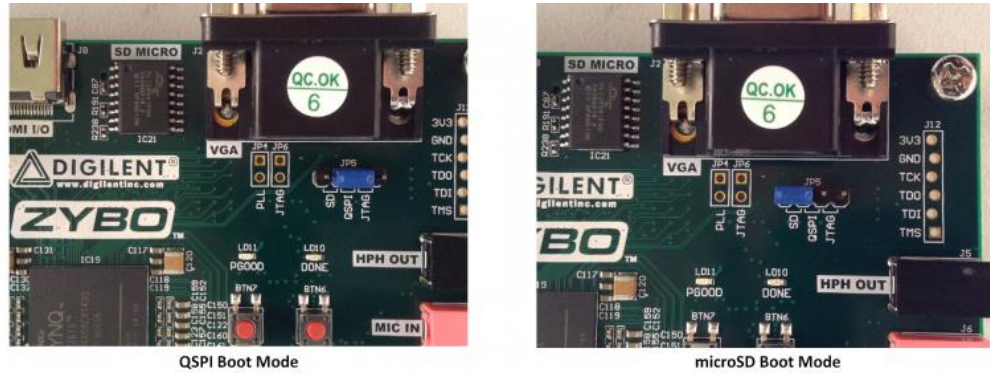
- Standalone application:** BootROM merupakan hardcode bootloader yang terdapat pada Zynq chip. BootROM akan men-load dan memanggil FSBL. FSBL kemudian akan men-load bitstream PL (optional) dan user application (e.g. led_blink.elf).
- Running Linux:** BootROM akan men-load dan memanggil FSBL. FSBL kemudian akan men-load bitstream PL (optional) dan U-boot. U-boot men-load compressed Linux kernel, device tree, dan ramdisk. U-boot kemudian menjalankan compressed Linux Image (zimage).



Gambar 1. Proses boot dari Zynq

c. Boot Image

Boot image untuk Zynq merupakan file yang bernama BOOT.bin. File ini berisi FSBL, bitstream, user application, atau U-boot. BOOT.bin dapat disimpan pada QSPI flash memory atau pada microSD card (format FAT). Konfigurasi boot dilakukan dengan setting jumper JP5 seperti pada gambar 2.



Gambar 2. Jumper untuk boot dari QSPI flash atau microSD card

BOOT.bin merupakan bagian penting dari Zynq, tetapi BOOT.bin **tidak** perlu selalu dibuat ketika kita memprogram PS. Di QSPI flash Zybo berisi factory default boot image. Boot image ini telah berisi FSBL, sehingga kita bisa men-load bitstream dan user application melalui JTAG. File bitstream dan user application yang di-load melalui JTAG sifatnya volatile, sehingga jika power dimatikan, maka akan hilang. BOOT.bin biasanya dibuat ketika akan membuat produk akhir, sehingga jika power dimatikan, maka tidak akan hilang.

3. Tutorial

Pada tutorial ini, kita akan melakukan beberapa hal sebagai berikut:

- Membuat design hardware (Vivado) untuk test LED dan serial port.
- Membuat program untuk test LED dan serial port.
- Membuat boot image.
- Upload boot image ke board ZYBO.

Sebelum memulai tutorial pastikan beberapa hal berikut ini untuk **menghindari error** saat membuat project:

- Disarankan untuk menginstall Vivado dan SDK pada folder yang tidak memiliki whitespace pada nama folder-nya (e.g. hindari menginstall di C:\Program Files, karena memiliki whitespace).
- Disarankan untuk **tidak** membuat project pada folder read-only (e.g. folder Documents, C:\Users).
- Pastikan Xilinx SDK telah terinstall. Cara ceknya bisa dari start menu windows cari Xilinx Design Tools, SDK 20xx.x.

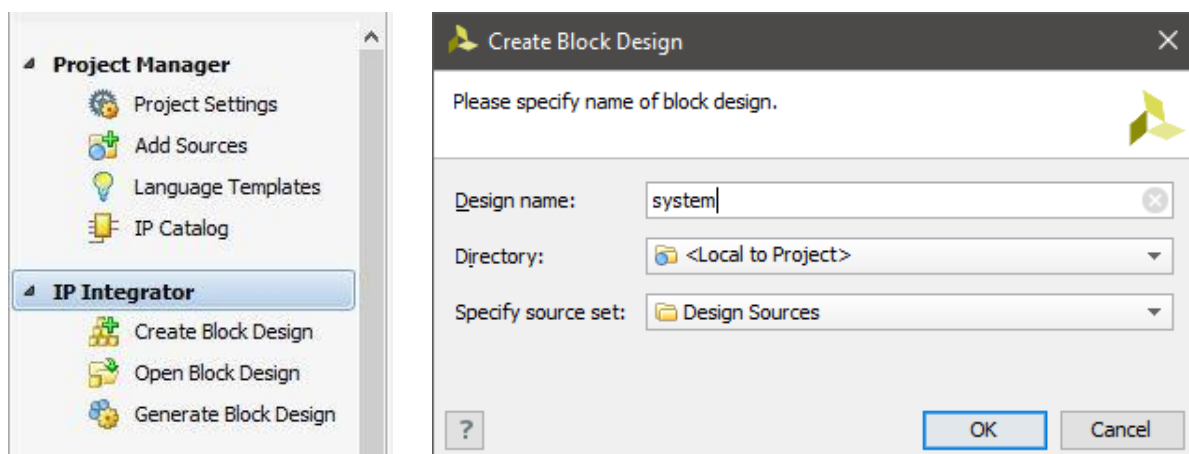
Beberapa file yang dibutuhkan untuk tutorial ini yaitu sebagai berikut:

- ZYBO_master.xdc
- ZYBO_C.tcl

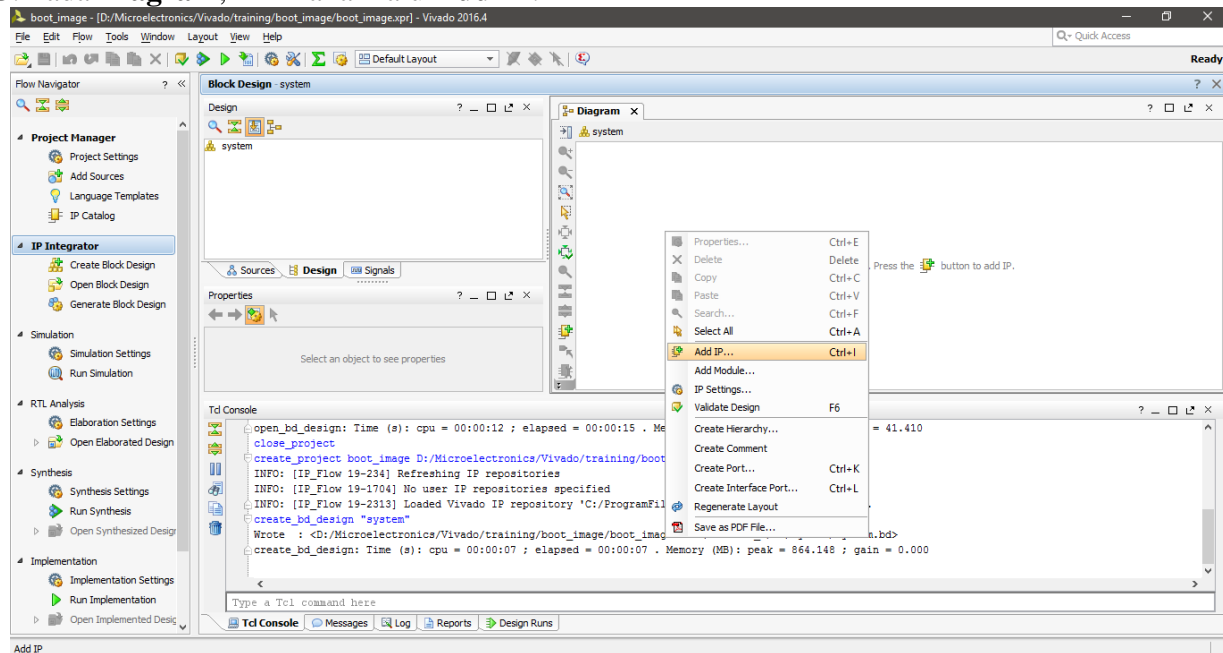
Membuat design hardware (Vivado) untuk test LED dan serial port.

1. Buat project baru untuk Zybo (detail step-by-step untuk membuat project baru dapat dilihat pada Modul 1).

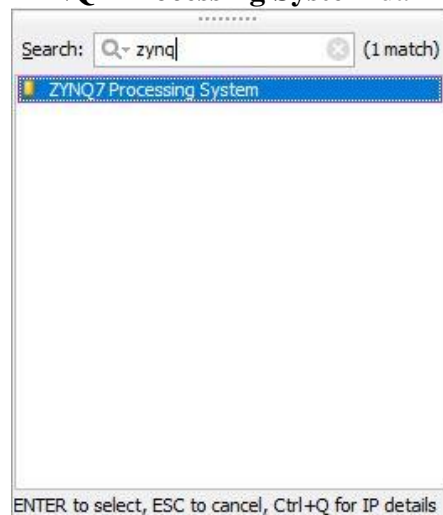
2. Buat block design dengan cara klik **Create Block Design** pada IP Integrator. Beri nama sebagai **system**.



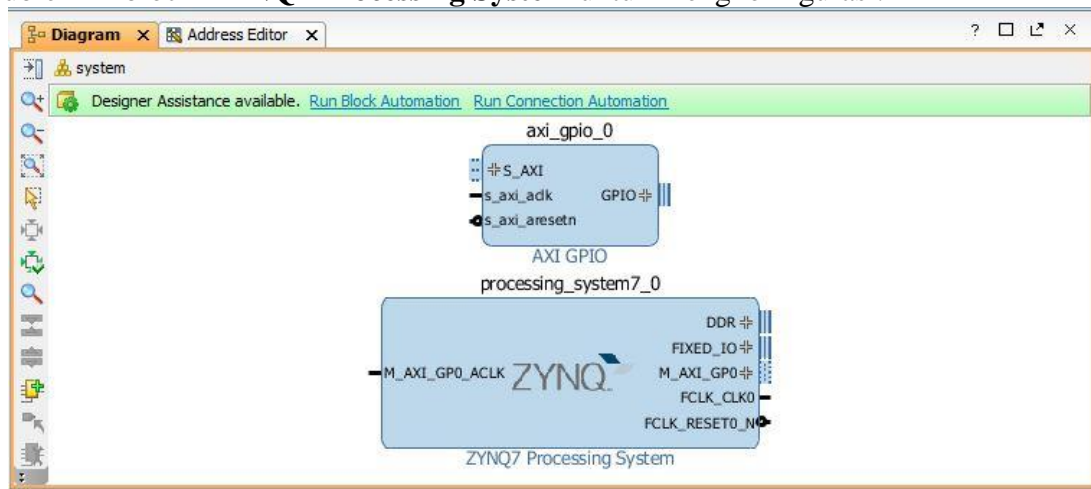
3. Pada **Diagram**, klik kanan lalu **Add IP**.



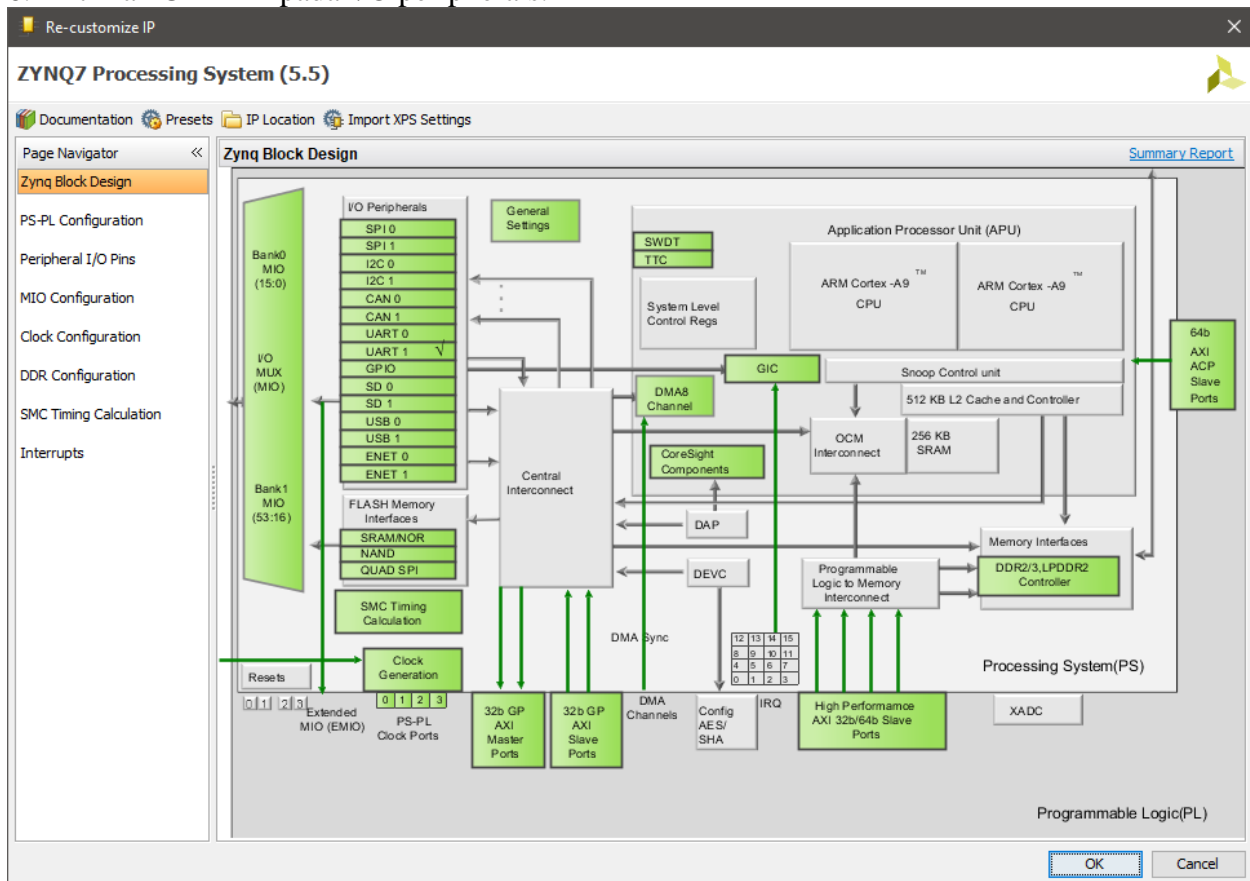
4. Add dua IP core berikut ini, **ZYNQ7 Processing System** dan **AXI GPIO**.

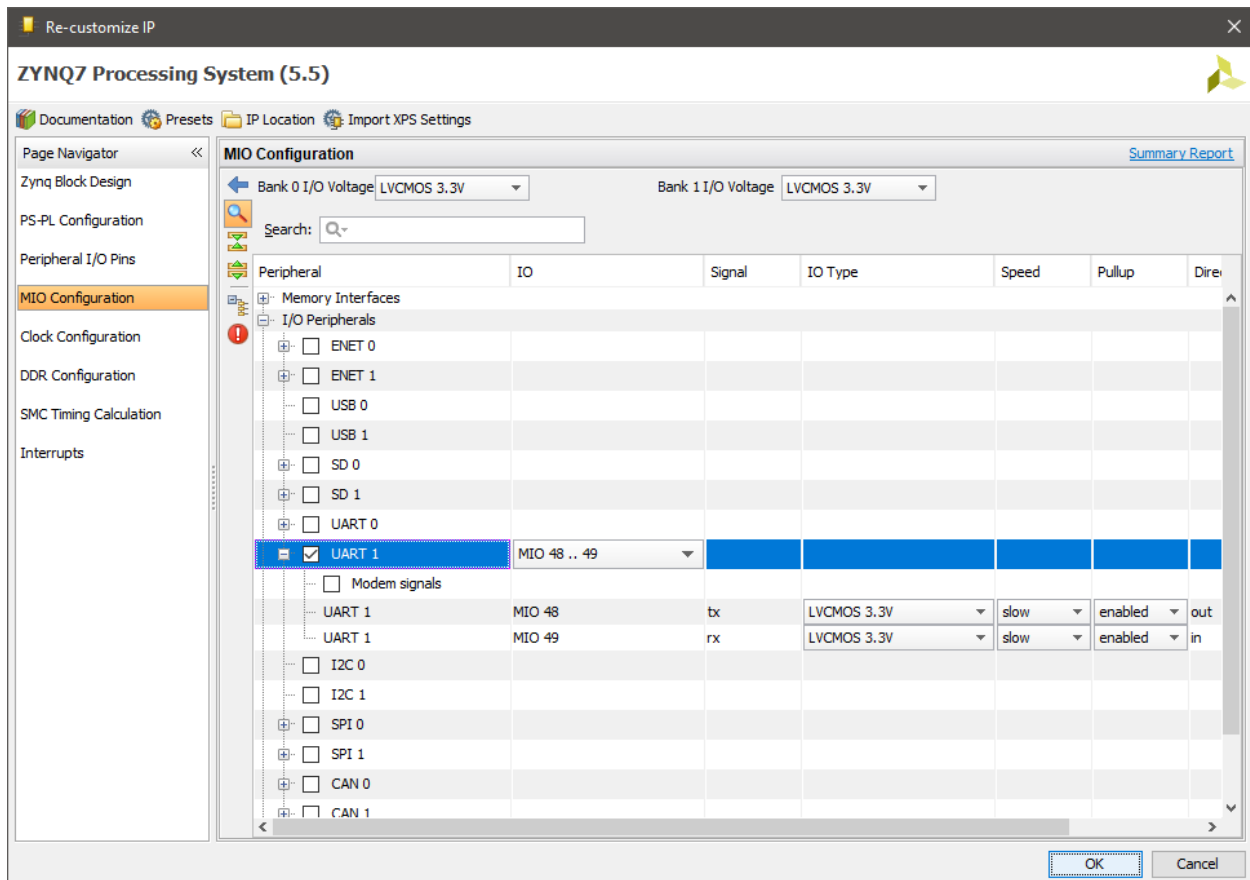


5. Double klik block **ZYNQ7 Processing System** untuk mengkonfigurasi.

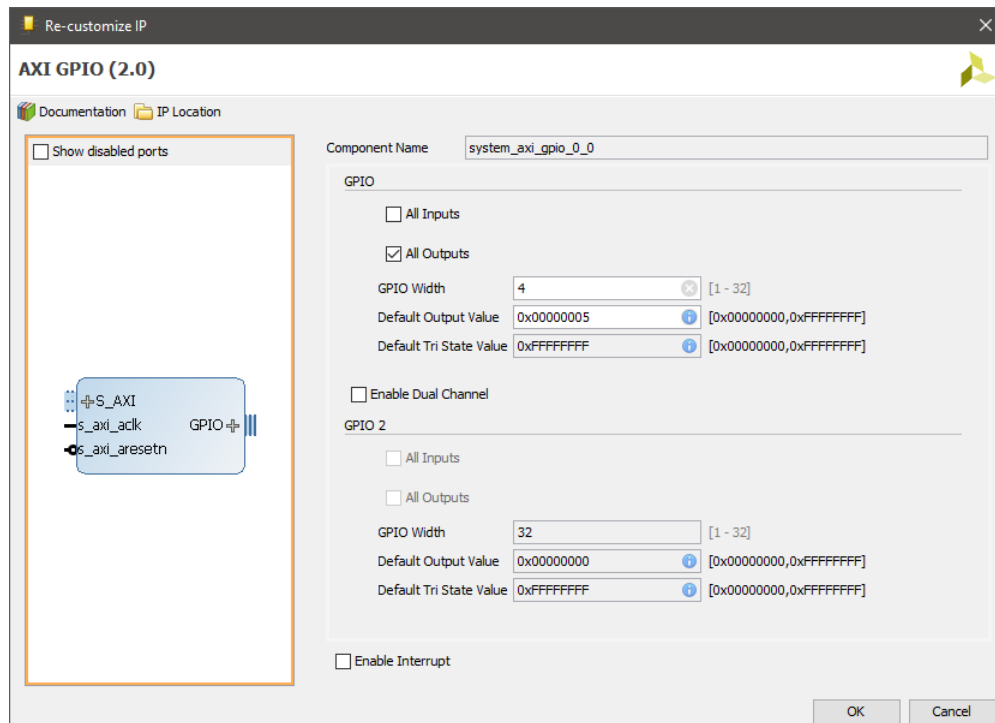


6. Aktifkan UART 1 pada I/O peripherals.

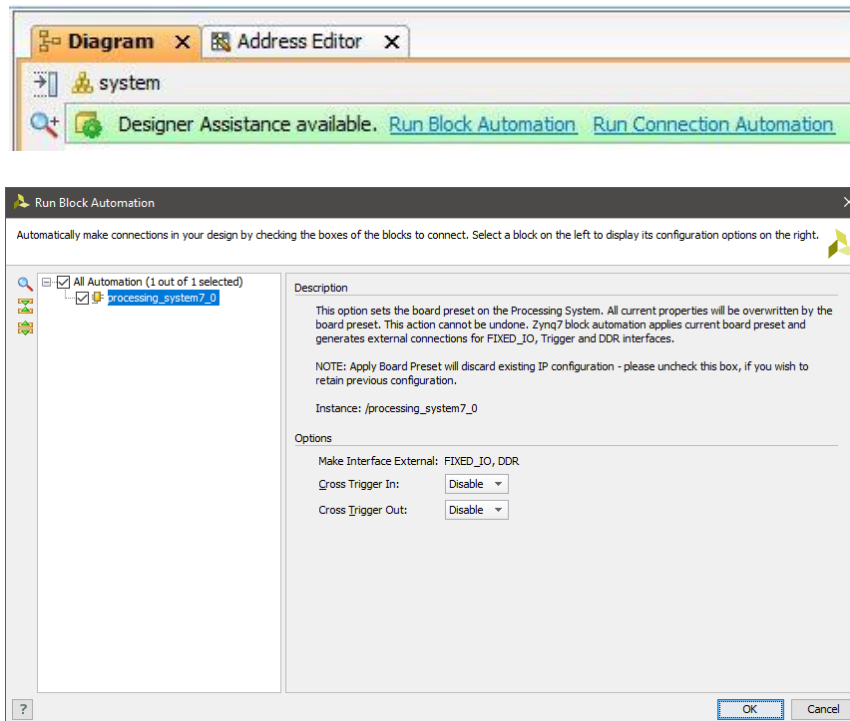




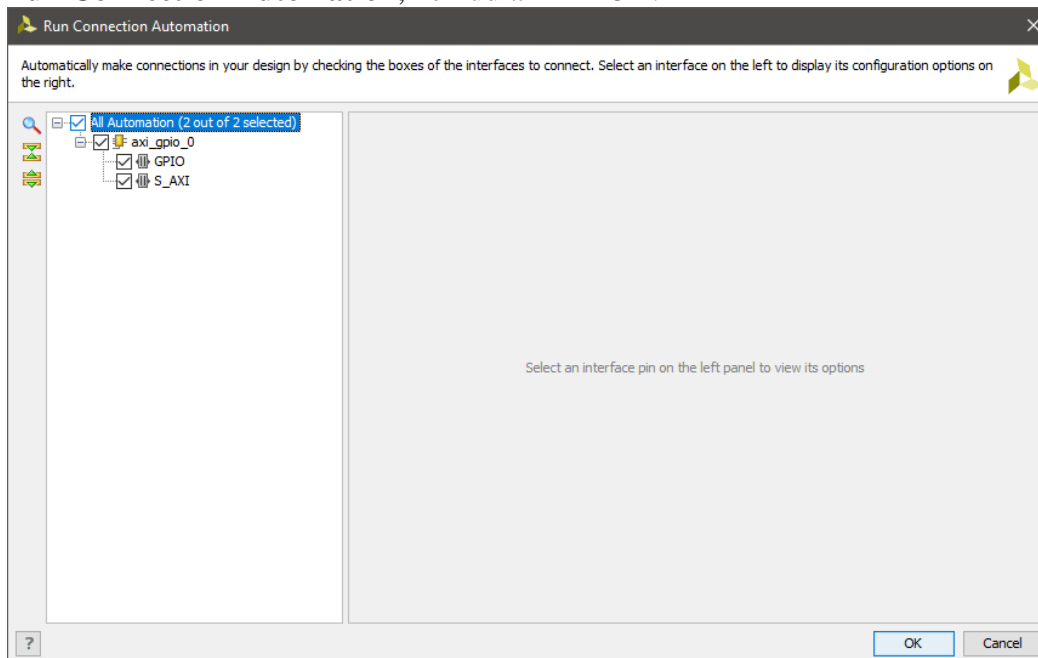
7. Double klik **IP AXI GPIO**, kemudian konfigurasi sebagai **All Outputs**, dan set default value menjadi **0x00000005**.



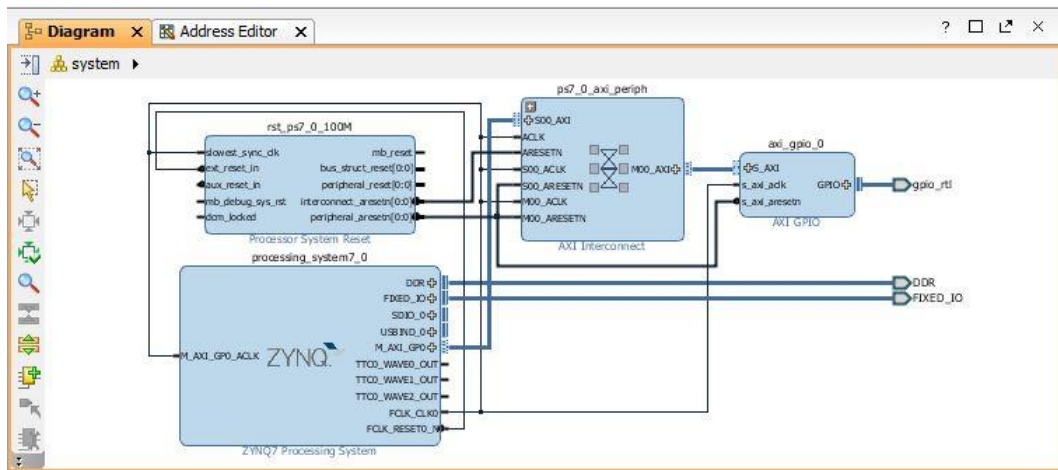
8. Klik **Run Block Automation**, kemudian klik OK.



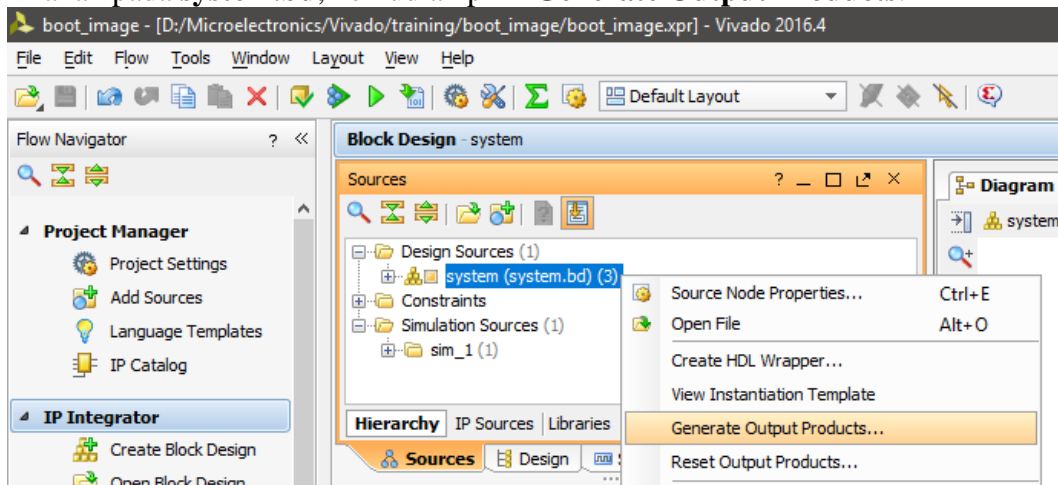
9. Klik **Run Connection Automation**, kemudian klik OK.



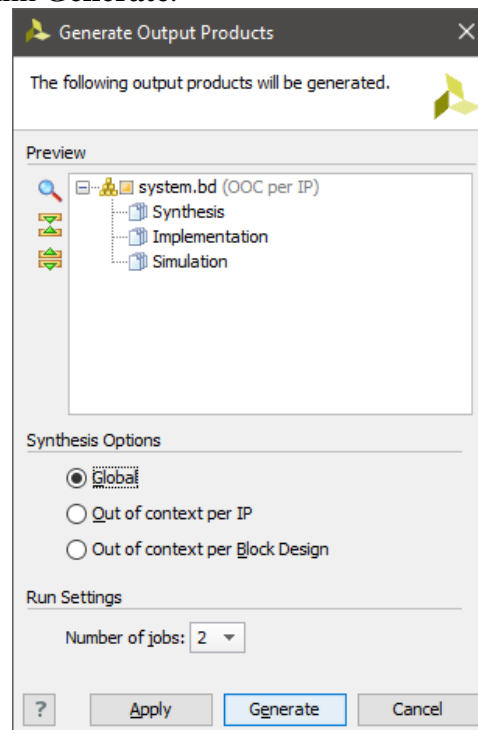
10. Hasil akhir block design seperti ini:



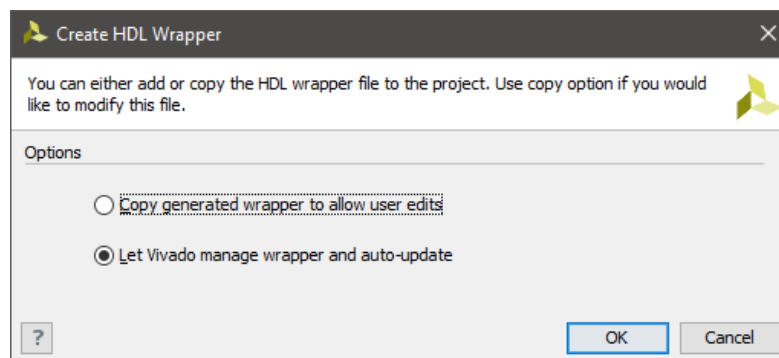
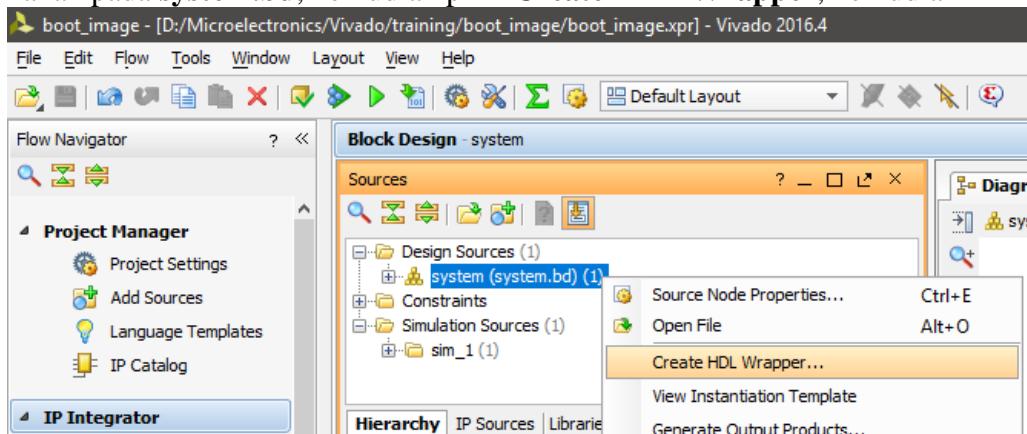
11. Klik kanan pada **system.bd**, kemudian pilih **Generate Output Products**.



12. Pilih **Global**, kemudian klik **Generate**.



13. Klik kanan pada **system.bd**, kemudian pilih **Create HDL Wrapper**, kemudian klik OK.



14. Add constraint file **ZYBO_master.xdc** (detail step-by-step untuk menambahkan constraint file dapat dilihat pada Modul 1). Kemudian ubah constraint untuk LED seperti ini:

```
#LEDs
#IO L23P T3 35
set_property PACKAGE_PIN M14 [get_ports {gpio_rtl_tri_o[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {gpio_rtl_tri_o[0]}]

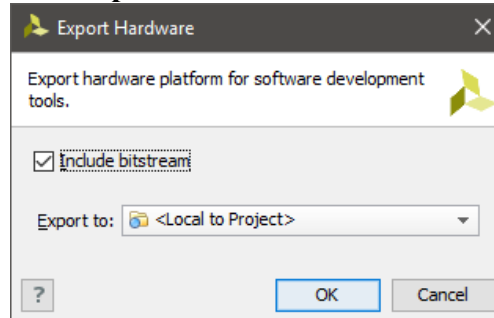
#IO_L23N_T3_35
set_property PACKAGE_PIN M15 [get_ports {gpio_rtl_tri_o[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {gpio_rtl_tri_o[1]}]

#IO_0_35
set_property PACKAGE_PIN G14 [get_ports {gpio_rtl_tri_o[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {gpio_rtl_tri_o[2]}]

#IO_L3N_T0_DQS_AD1N_35
set_property PACKAGE_PIN D18 [get_ports {gpio_rtl_tri_o[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {gpio_rtl_tri_o[3]}]
```

15. Lakukan synthesis, implementation, dan generate bitstream.

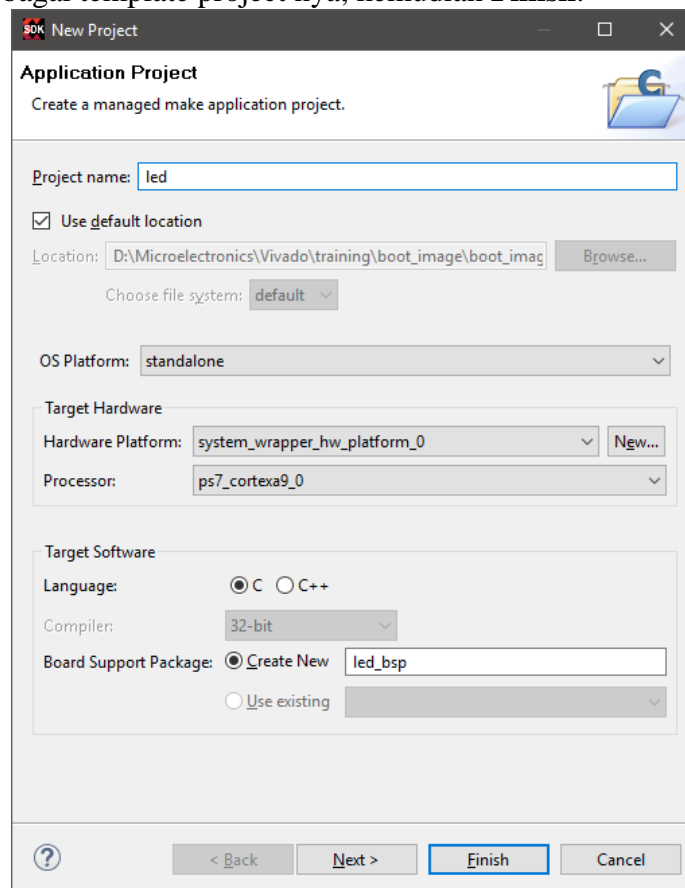
16. Pilih menu **File** → **Export** → **Export Hardware**, aktifkan **Include Hardware**, lalu klik OK.



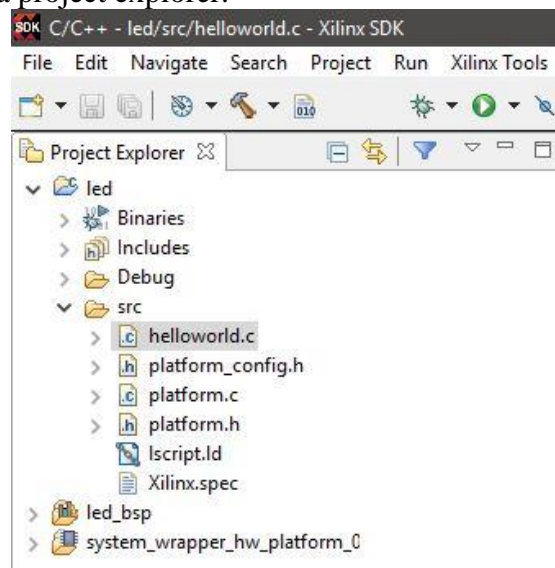
17. Launch SDK dengan cara, **File** → **Launch SDK**, lalu klik OK.

Membuat program untuk test LED dan serial port.

1. Pilih menu **File** → **New** → **Application Project**. Masukan nama project kemudian, **Next**, lalu pilih **Hello World** sebagai template project nya, kemudian **Finish**.



2. Buka **helloworld.c** pada project explorer.



3. Buat kode program seperti ini:

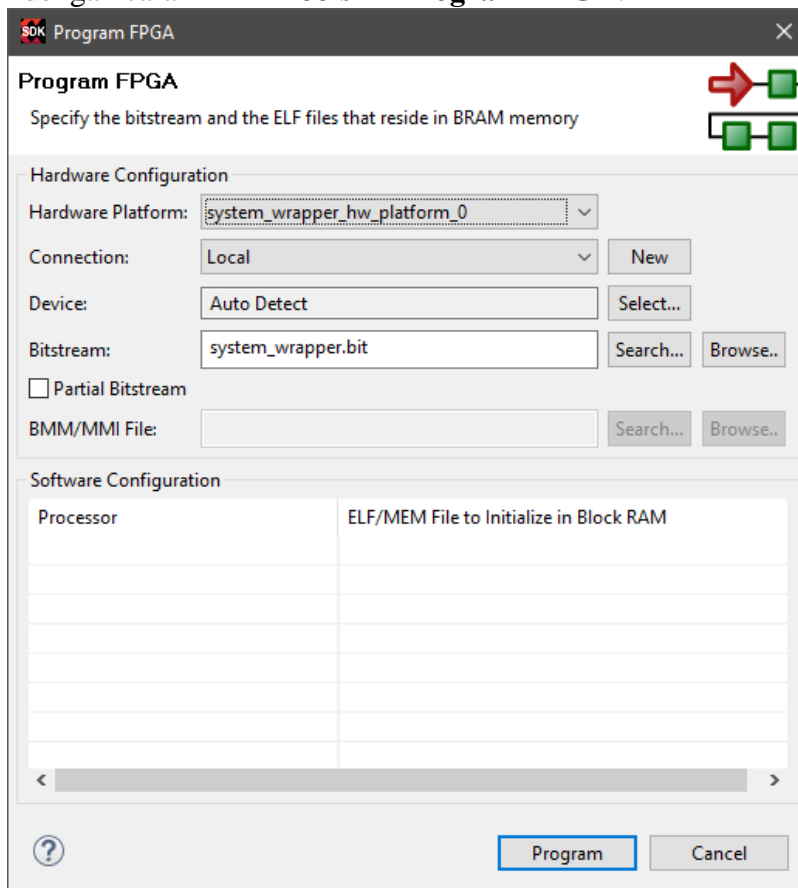
```
#include <stdio.h>
#include <stdint.h>
#include "sleep.h"

uint32_t *led_p = (uint32_t *)0x41200000;

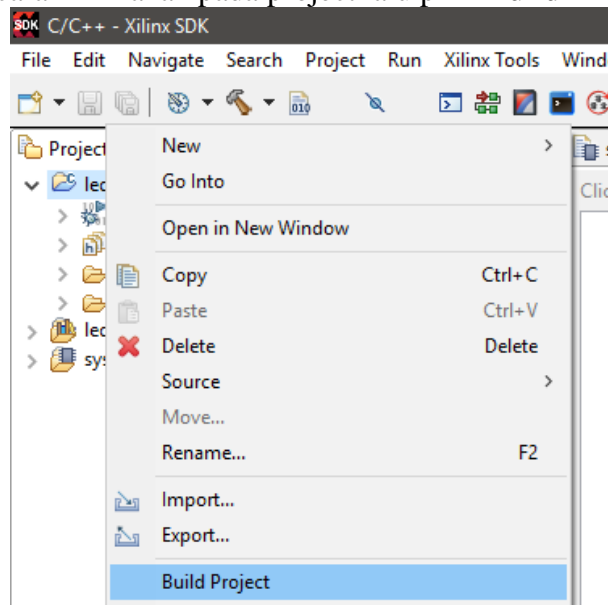
int main()
{
    while (1)
    {
        printf("0xA\n");
        *(led_p+0) = 0xA;
        sleep(1);
        printf("0x5\n");
        *(led_p+0) = 0x5;
        sleep(1);
    }

    return 0;
}
```

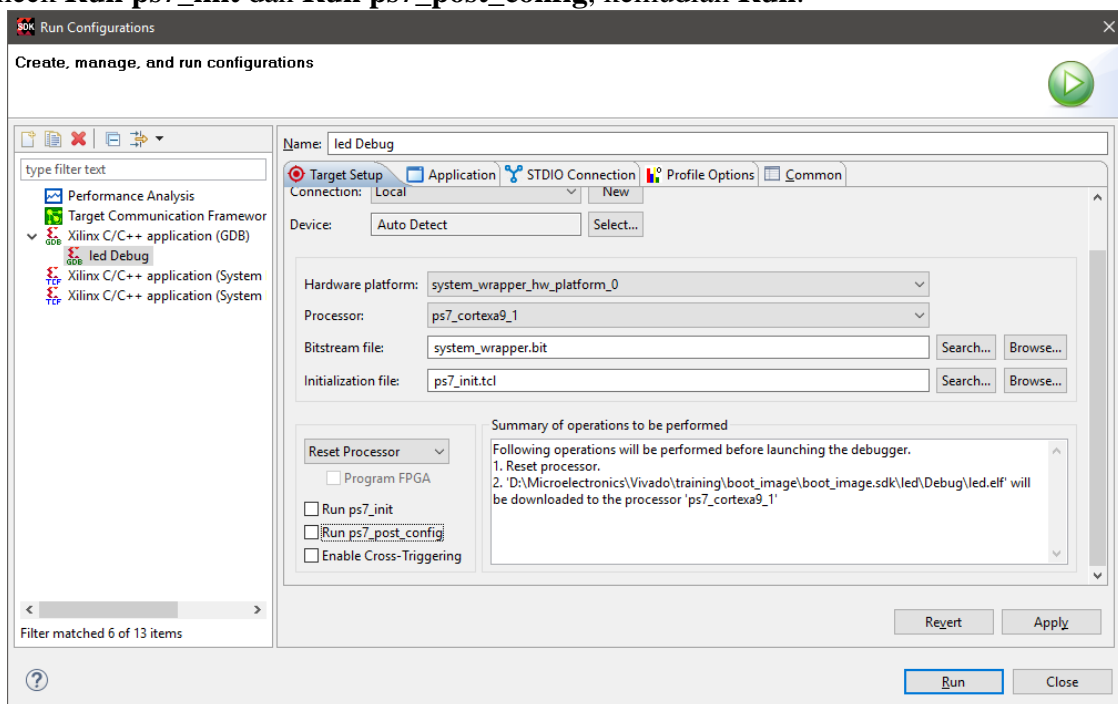
4. Program FPGA dengan cara **Xilinx Tools** → **Program FPGA**.



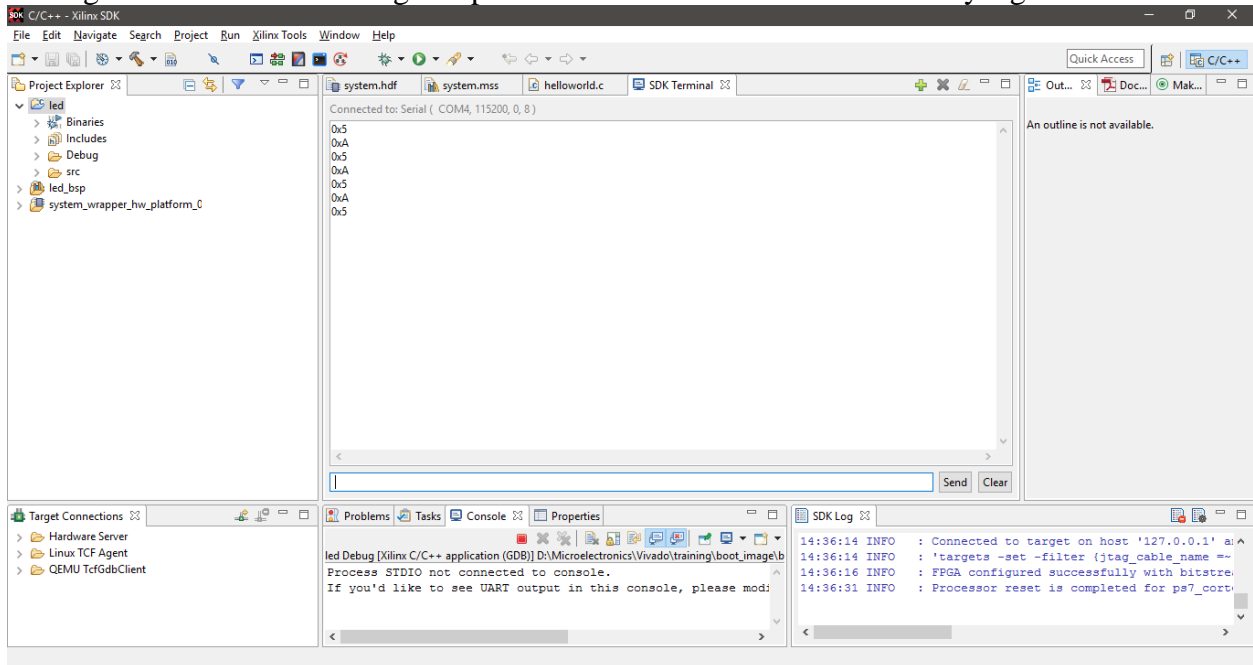
5. Build project dengan cara klik kanan pada project lalu pilih **Build Project**.



7. Dari menu **Run**, pilih **Run Configurations**, double klik **Xilinx C/C++ application (GDB)**. Uncheck **Run ps7_init** dan **Run ps7_post_config**, kemudian **Run**.



8. Program LED sudah running dan pada SDK terminal terlihat data serial yang diterima.



9. Klik tombol ini untuk **stop** program.

