

# HELLO, FPGA

Document my FPGA learning journey

APRIL 2023, 4 BY ACKYE

## Smart ZYNQ (SP&SL Edition) Project 6 Use ZYNQ's PS to light up the LED light connected to the PL end AXI\_GPIO\_IP method (full picture and text)

The LED lights lit before are lit through PL resources, that is, through the logic of the FPGA to light the LED lights, this project will try to call ZYNQ's PS resources for the first time to achieve a point LED light operation, because the LED resources on the board are connected to the PL end, so here is a linkage, with PS resources to light the PL end LED lights

(This method will occupy PL resources, generally recommended to use EMIO method, will save resources)

This article is demonstrated on vivado2018.3, please research for other versions

**(Note: The content of this section applies to the boards of Smart ZYNQ SP and SL Edition, if it is Smart ZYNQ Standard Edition, please refer to the corresponding board directory)**

### 1. Create a Vivado project

1) Specific steps to create a VIVADO project, open the software and select Create Project, as shown in the following figure

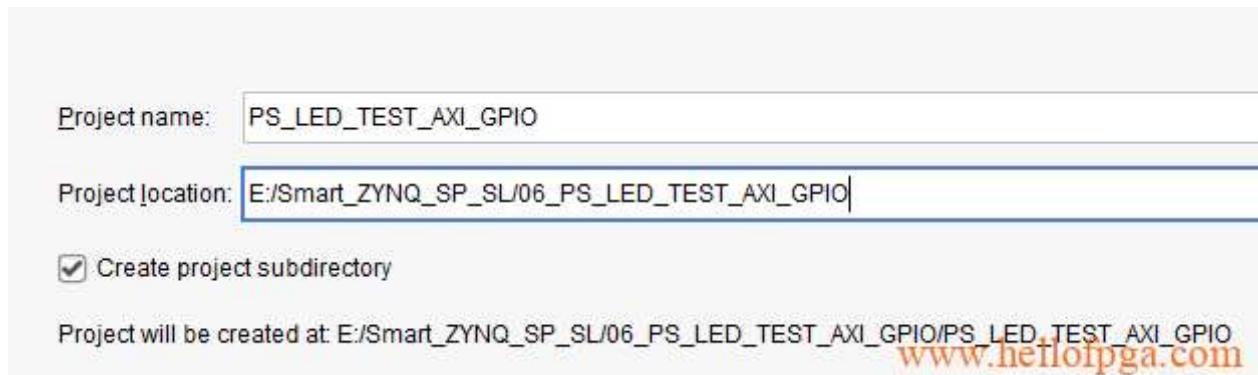
# Quick Start

[Create Project >](#)

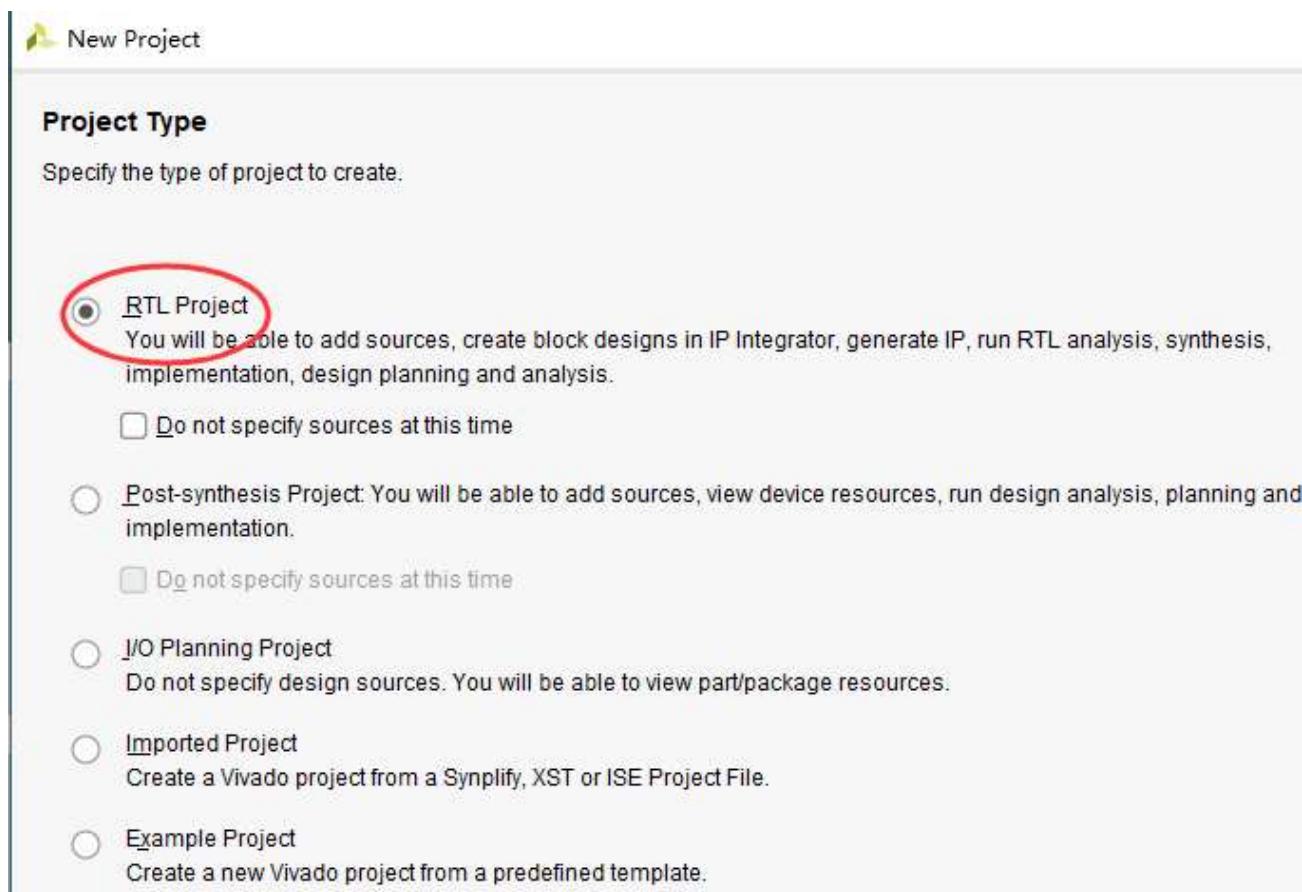
[Open Project >](#)

[Open Example Project >](#)

- 2) Click NEXT and enter the project name in the second dialog box "Project name" that appears; Select the save path in Project location; Check "Create project subdirectory" and click "Next" **Note, all paths cannot appear Chinese name**



- 3) Click the RTL PROJECT option and click NEXT



- 4) Step <>: The Add Sources option is left blank, NEXT

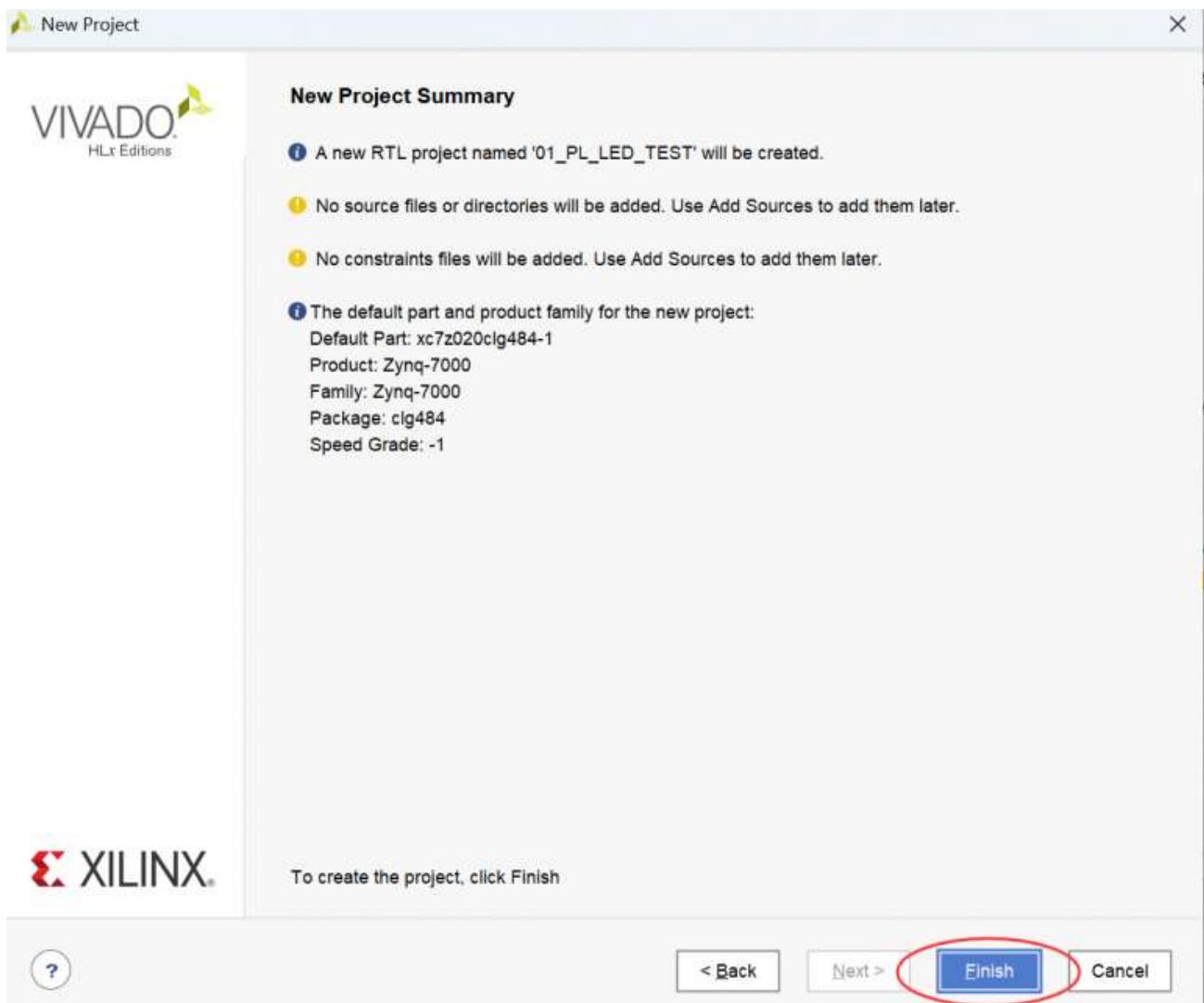
5) Step <>: The Add Constraints option is left blank, NEXT

6) Select the chip model XC7Z020CLG484-1

The screenshot shows the 'New Project' dialog with the 'Default Part' tab selected. The search bar at the top contains the query 'xc7z020'. A red circle labeled '1' highlights the search bar. Below the search bar is a table listing various Xilinx parts. A red circle labeled '2' highlights the row for 'xc7z020clg484-1'. At the bottom right of the dialog, there are buttons for '?', '< Back', 'Next >', 'Finish', and 'Cancel'. A red circle labeled '3' highlights the 'Finish' button. The URL 'www.hellofpga.com' is visible in the bottom right corner of the dialog.

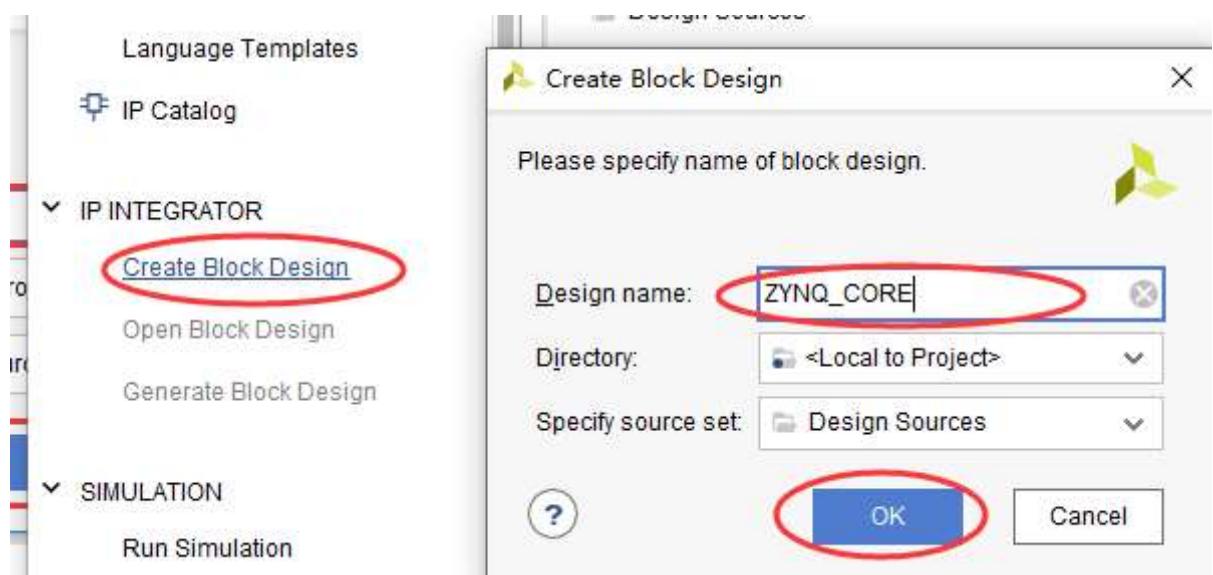
Part	I/O Pin Count	Available IOBs	LUT Elements	FlipFlops	Block RAMs	Ultra RAMs	DSPs	Gb Tr
xc7z020clg400-3	400	125	53200	106400	140	0	220	0
xc7z020clg400-2	400	125	53200	106400	140	0	220	0
xc7z020clg400-1	400	125	53200	106400	140	0	220	0
xc7z020clg484-3	484	200	53200	106400	140	0	220	0
xc7z020clg484-2	484	200	53200	106400	140	0	220	0
xc7z020clg484-1	484	200	53200	106400	140	0	220	0
xc7z020clg400-1L	400	125	53200	106400	140	0	220	0
xc7z020iclg484-1L	484	200	53200	106400	140	0	220	0

7) Confirm the selected information Click "Finish" to complete the project creation of vivado

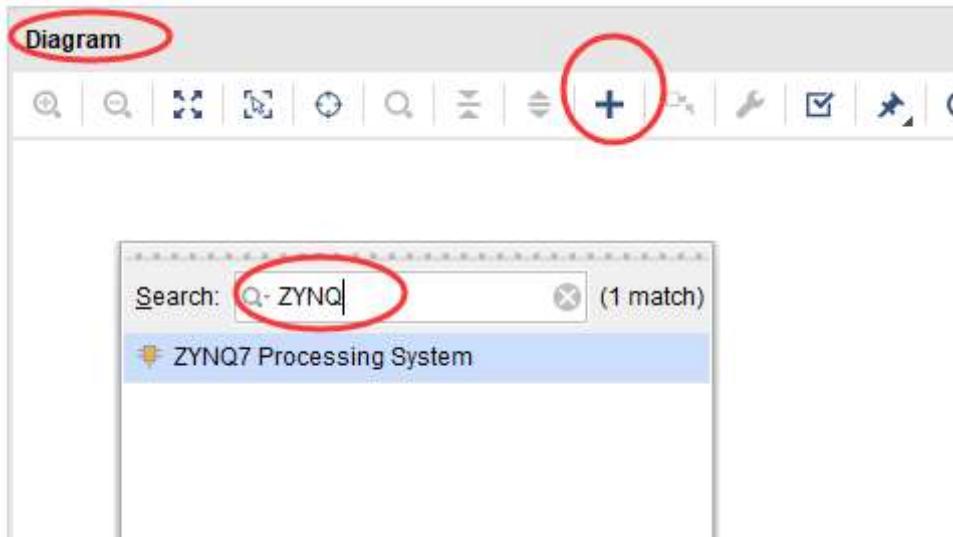


## 2 Create a BLOCK design

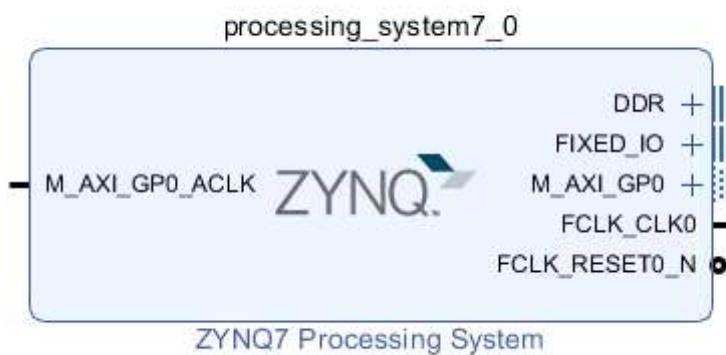
1) IP INTEGRATOR→Create Block Design, enter the design name in the pop-up dialog box, and finally click "OK", as shown in the figure below



2) IN THE WINDOW ON THE RIGHT, CLICK THE PLUS SIGN, SEARCH FOR ZYNQ IN THE SELECTION BOX, AND FIND ZYNQ7 PROCESSING SYSTEM, DOUBLE-CLICK AND OPEN



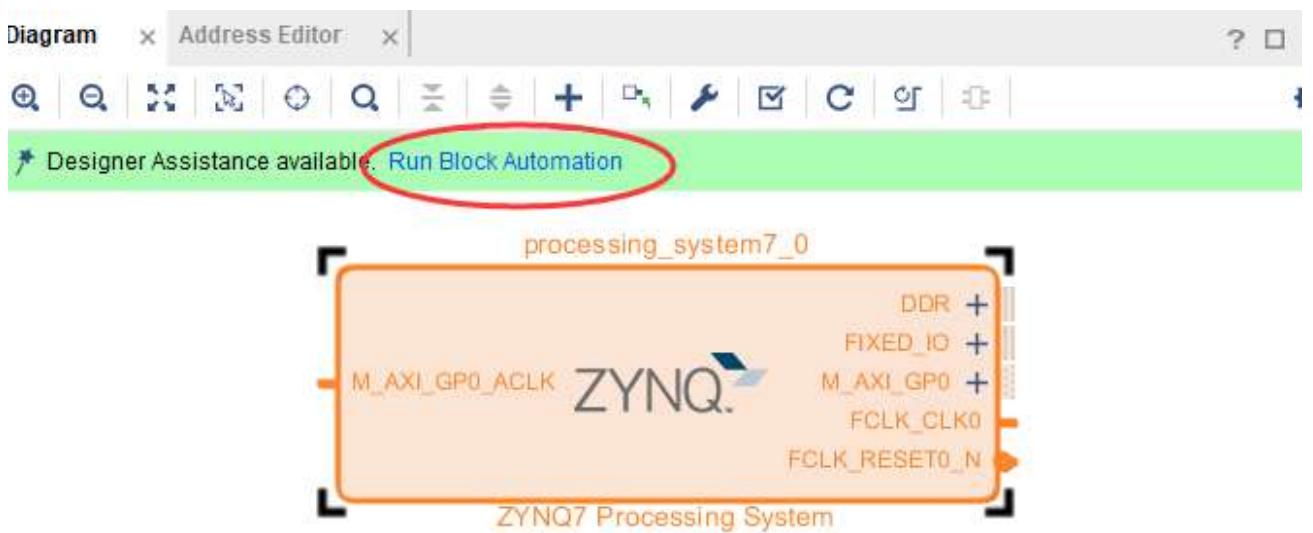
3) The software automatically generates a zynq block as shown in the figure below, next to do some corresponding settings, double-click the ZYNQ core in the figure below



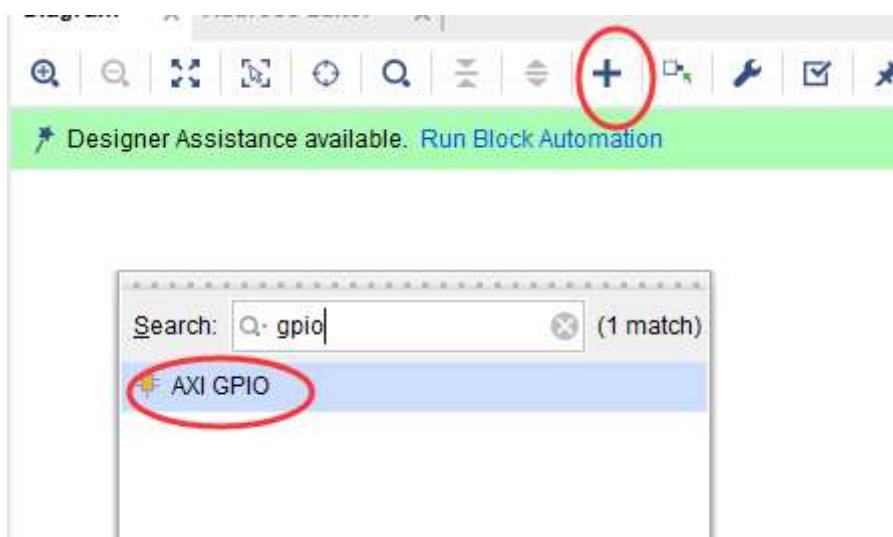
4) Find DDR Configuration → DDR Controller Configuration → DDR3 in the pop-up window in turn, select the corresponding DDR3 according to the DDR on your board in the Memory Part drop-down menu, the model used in this experiment: MT41K256M16RE-125, select 16bit of data width and finally click "OK", as shown in the figure below.

Name	Select	Description
Memory Type	DDR 3	Type of memory interface. Refer to UG
Memory Part	MT41K256M16 R...	Memory component part number. For u
Effective DRAM Bus Width	16 Bit	Data width of DDR interface, not includ
ECC	Disabled	Enables error correction code support.
Burst Length	8	Minimum number of data beats the cor
DDR	533.333333	Memory clock frequency. The allowed f
Internal Vref	<input type="checkbox"/>	Enables internal voltage reference sou
Junction Temperature (C)	Normal (0-85)	Intended operating temperature range.
Memory Part Configuration		
Training/Board Details		
Additive Latency (cycles)	0	Additive Latency (cycles). Increases the
Enable Advanced options		

Finally, click on "Run Block Automation" as shown in the image below. Keep the default in the pop-up options and click "OK" to complete the configuration of the ZYNQ7 Processing System



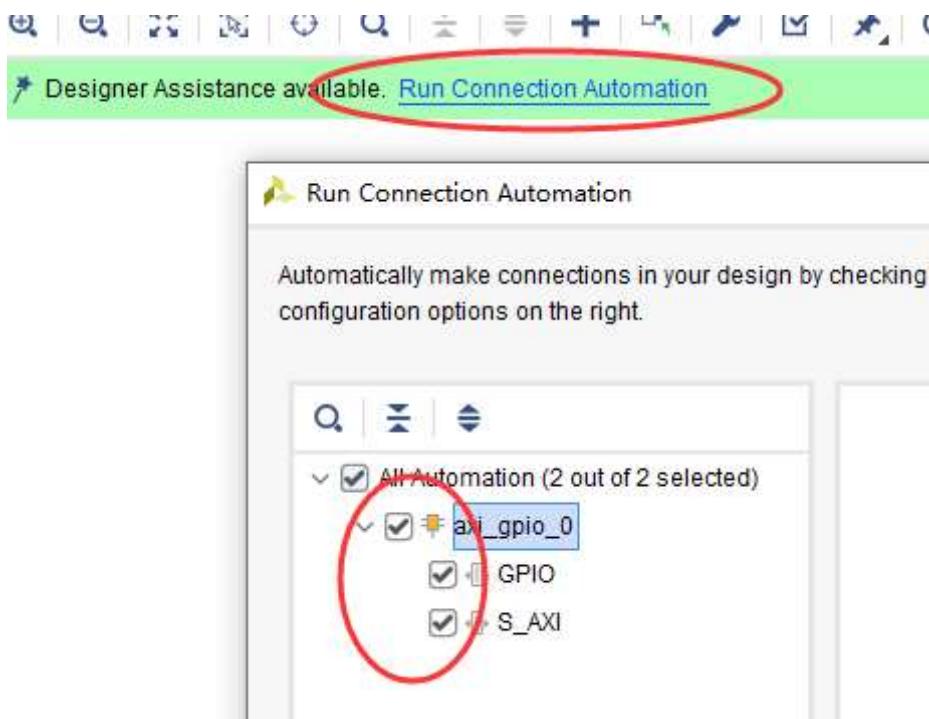
- 5) Use the same method to increase GPIO resources Click the plus sign, search for GPIO in the settings and add, then double-click and open the generated GPIO resources for settings



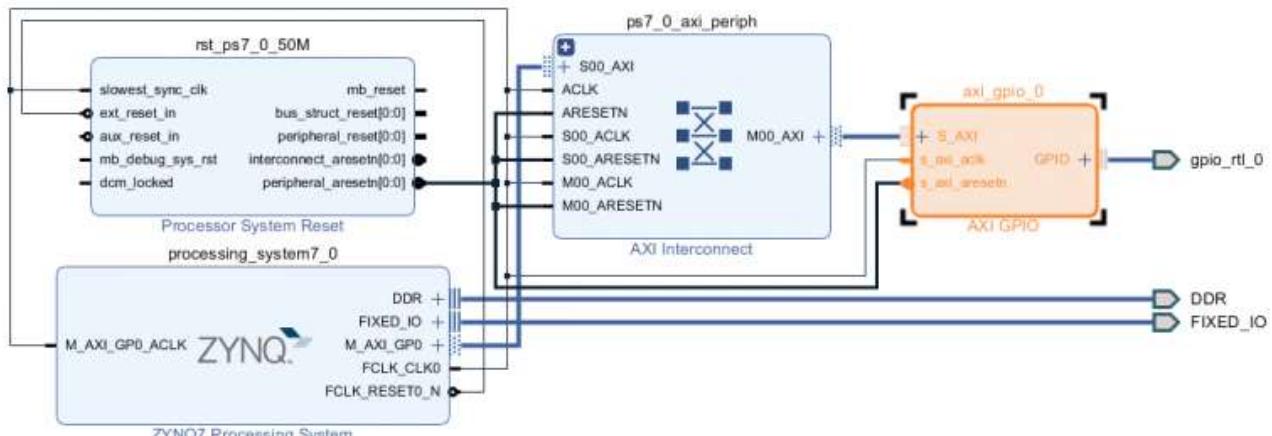
- 6) In the GPIO settings bar, change the input and output properties to All Output ALL Outputs, because there are only 2 LED lights on our board, so set the GPIO Width width to 2 bits, and click OK



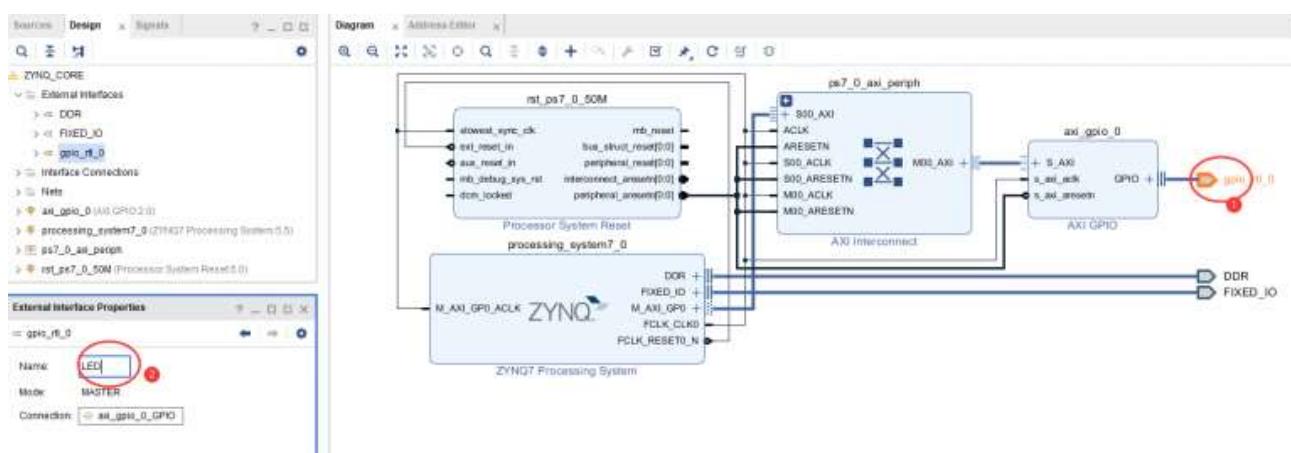
- 7) Click Run Connection Automation above, and check both the GPIO and the tick in front of the S\_AXI, as shown in the following figure click OK



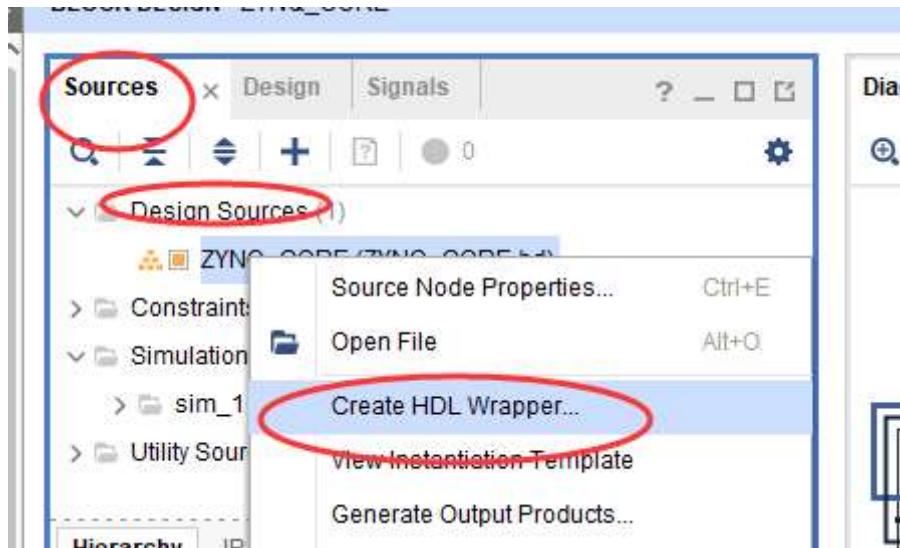
8) The software will automatically help us complete all the necessary connections and add the necessary modules



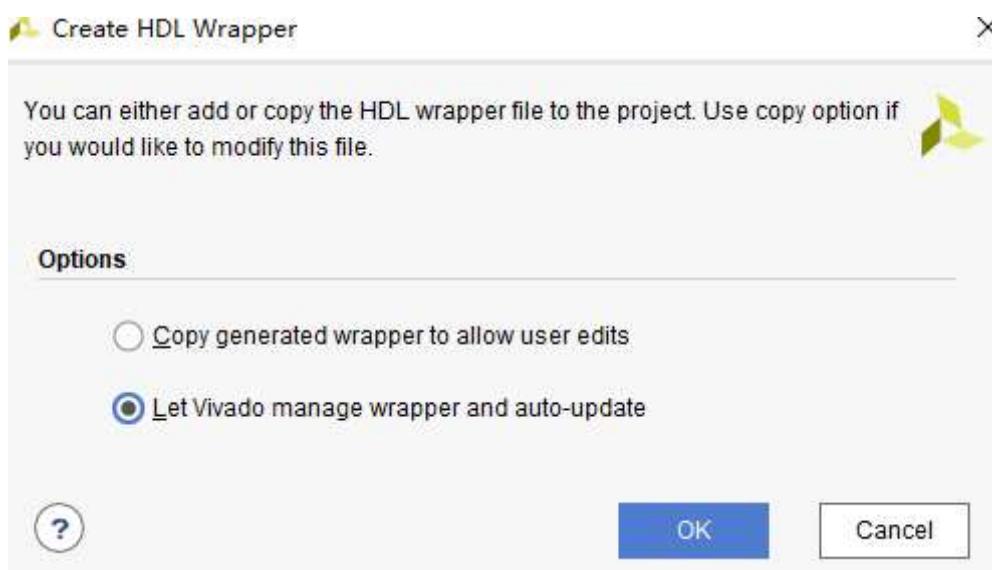
9) Double-click the pin of the GPIO output in the block diagram and rename it to LED to facilitate software programming later



10) source→Design Source, right-click on the BLOCK project we created, and click create HDL wrapper as shown in the figure below.



Keep the default in the pop-up dialog

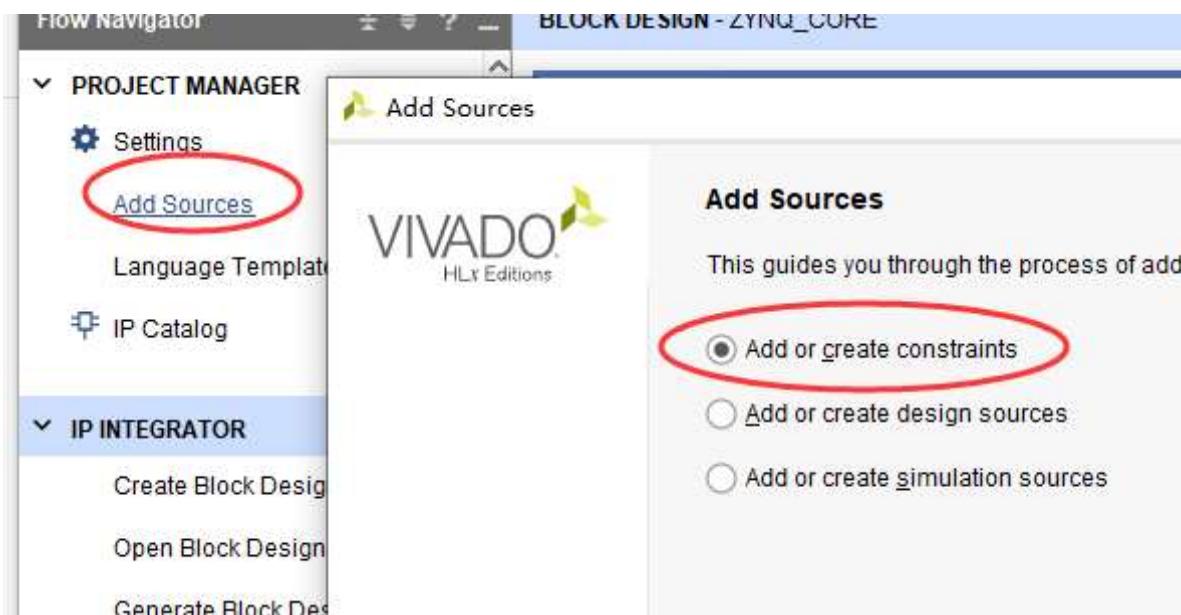


The software automatically generates HDL files for us

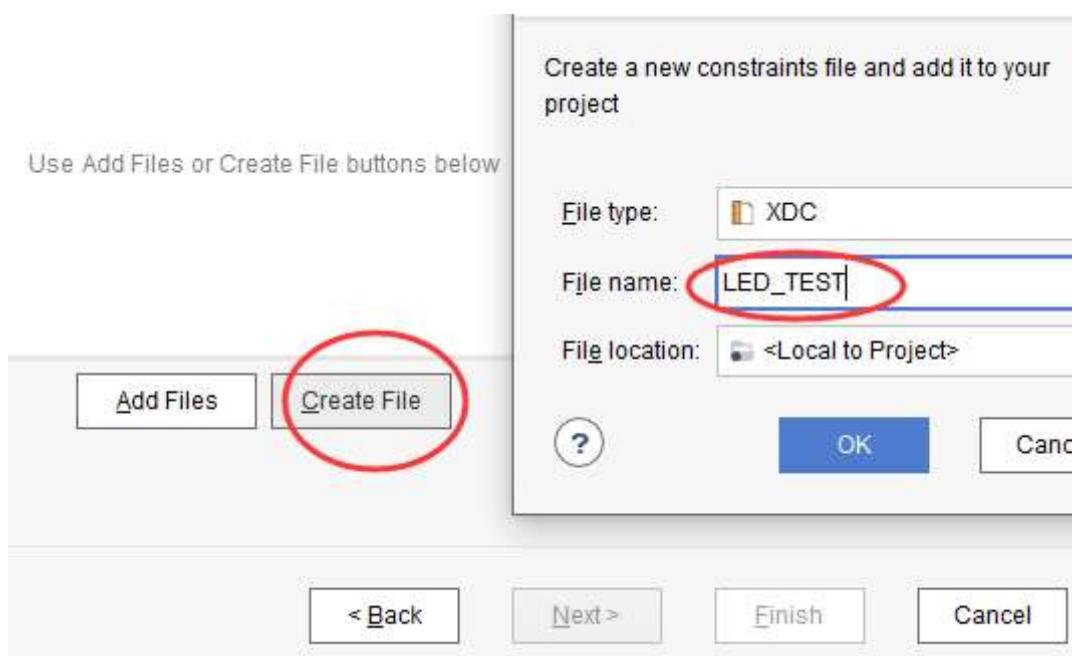


### 3. 创建约束文件，并且定义管脚

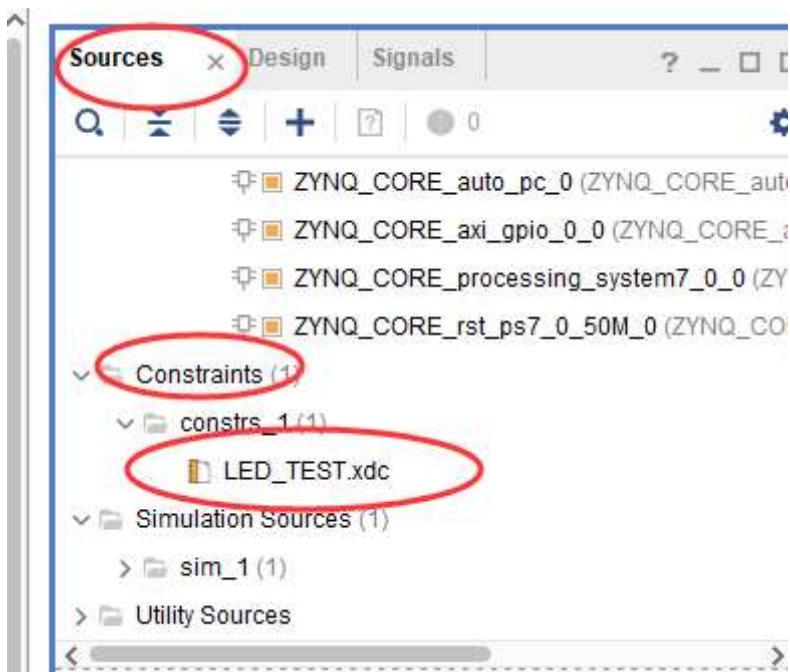
- 1) Add Source → Add or create constraints 点Next



因为这个项目没有创建过约束文件 所以这里创建一个约束文件，并在File name 里设置约束文件的名称，并且点击FINISH 完成约束文件的创建



- 2) Sources → Constraints 里找到刚才创建的约束文件 双击并打开该XDC约束文件



在约束文件里面复制下面代码来对输出的GPIO进行管脚（所有的管脚转接板上丝印都有实际标注对应的IO）

```
set_property IOSTANDARD LVCMOS33 [get_ports LED_tri_o[0]]  
set_property IOSTANDARD LVCMOS33 [get_ports LED_tri_o[1]]  
  
set_property PACKAGE_PIN P20 [get_ports LED_tri_o[0]]  
set_property PACKAGE_PIN P21 [get_ports LED_tri_o[1]]
```

## 4.生成bit文件

按下绿色箭头对工程进行编译

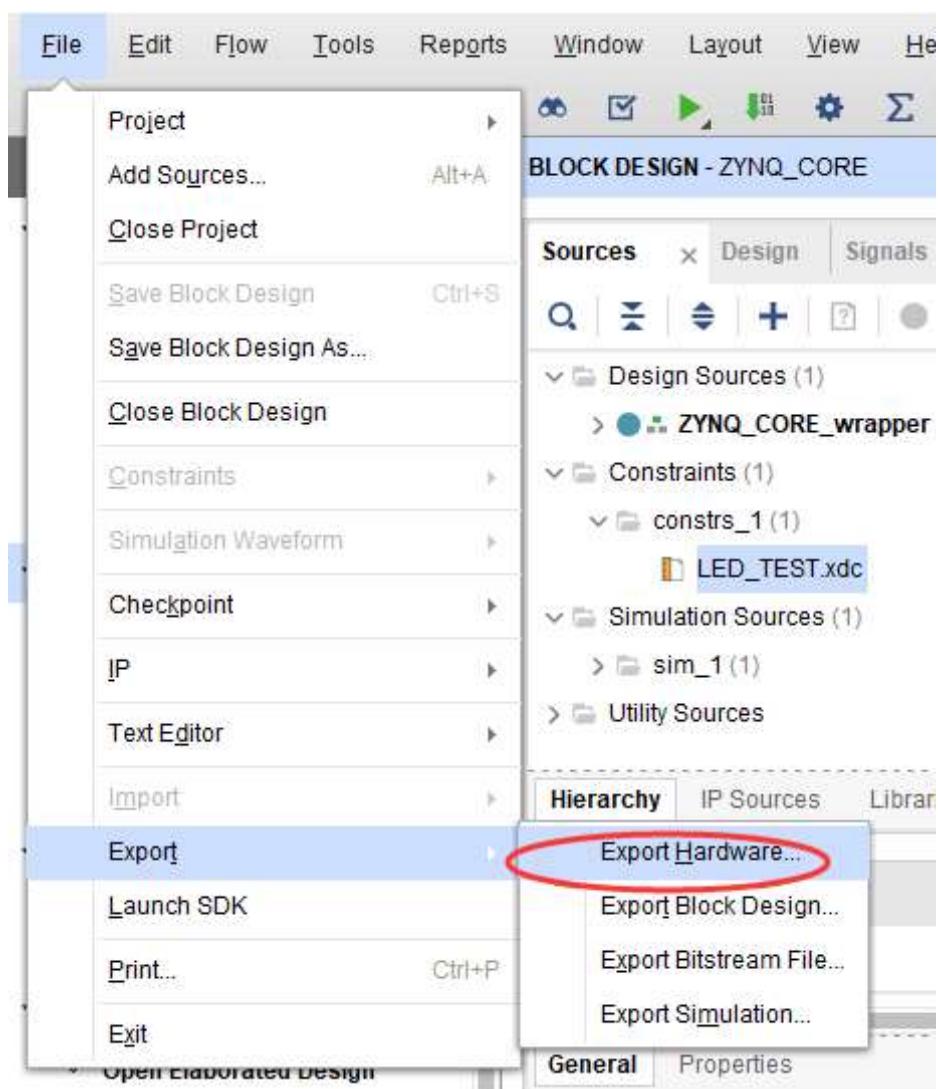


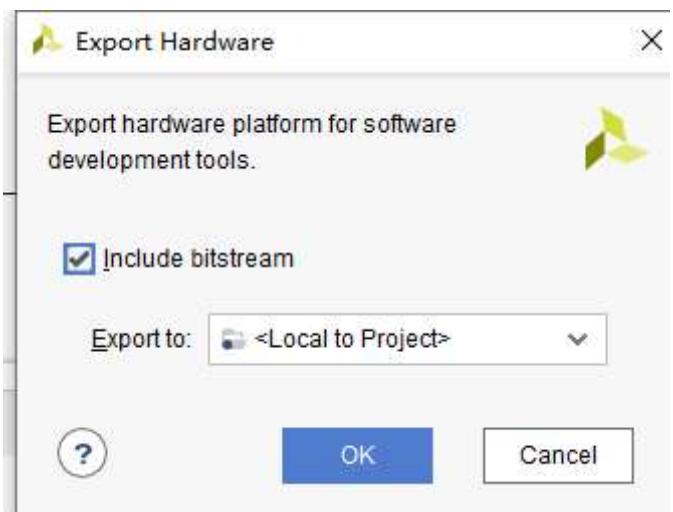
按下Generate Bitstream 完成综合以及生成bit文件



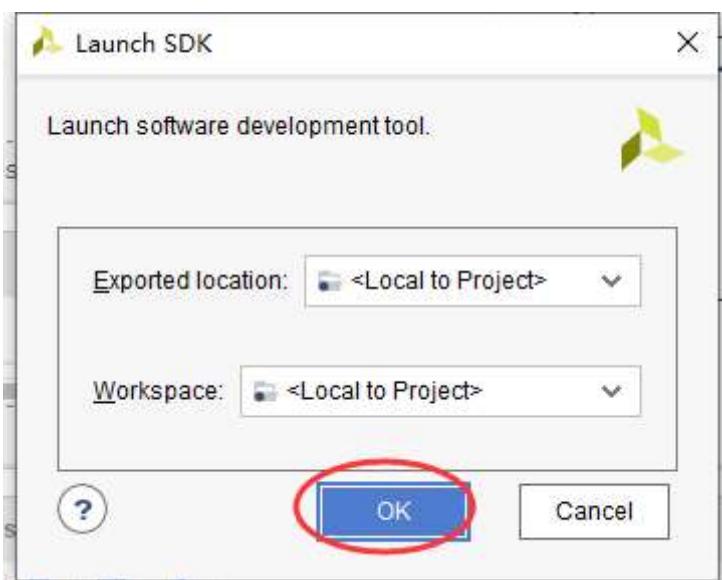
## 5.SDK程序编写

1) File→Export→Export hardware...，在弹出的对话框中勾选“include bitstream”，点击“OK”确认，如下图所示。

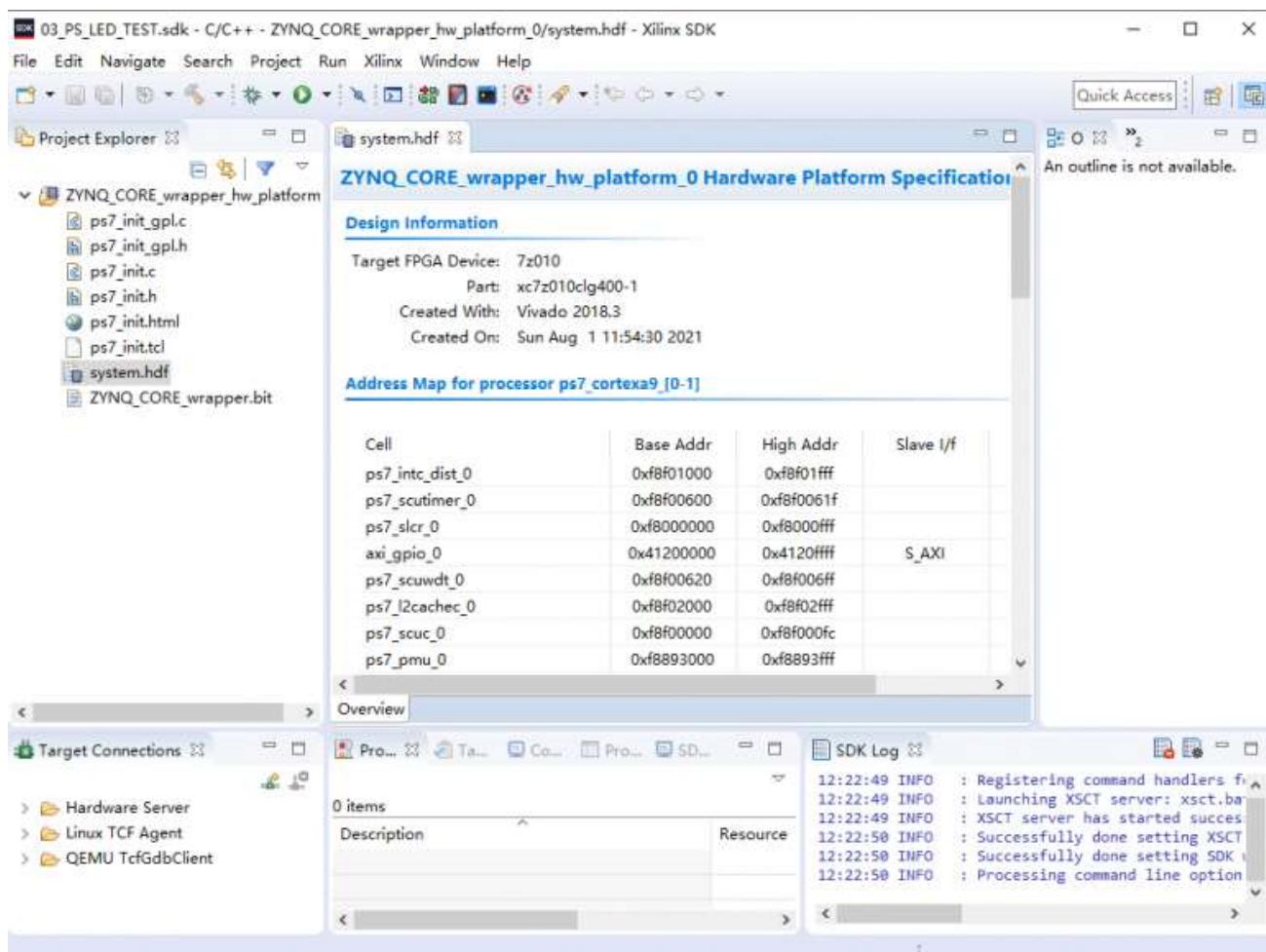




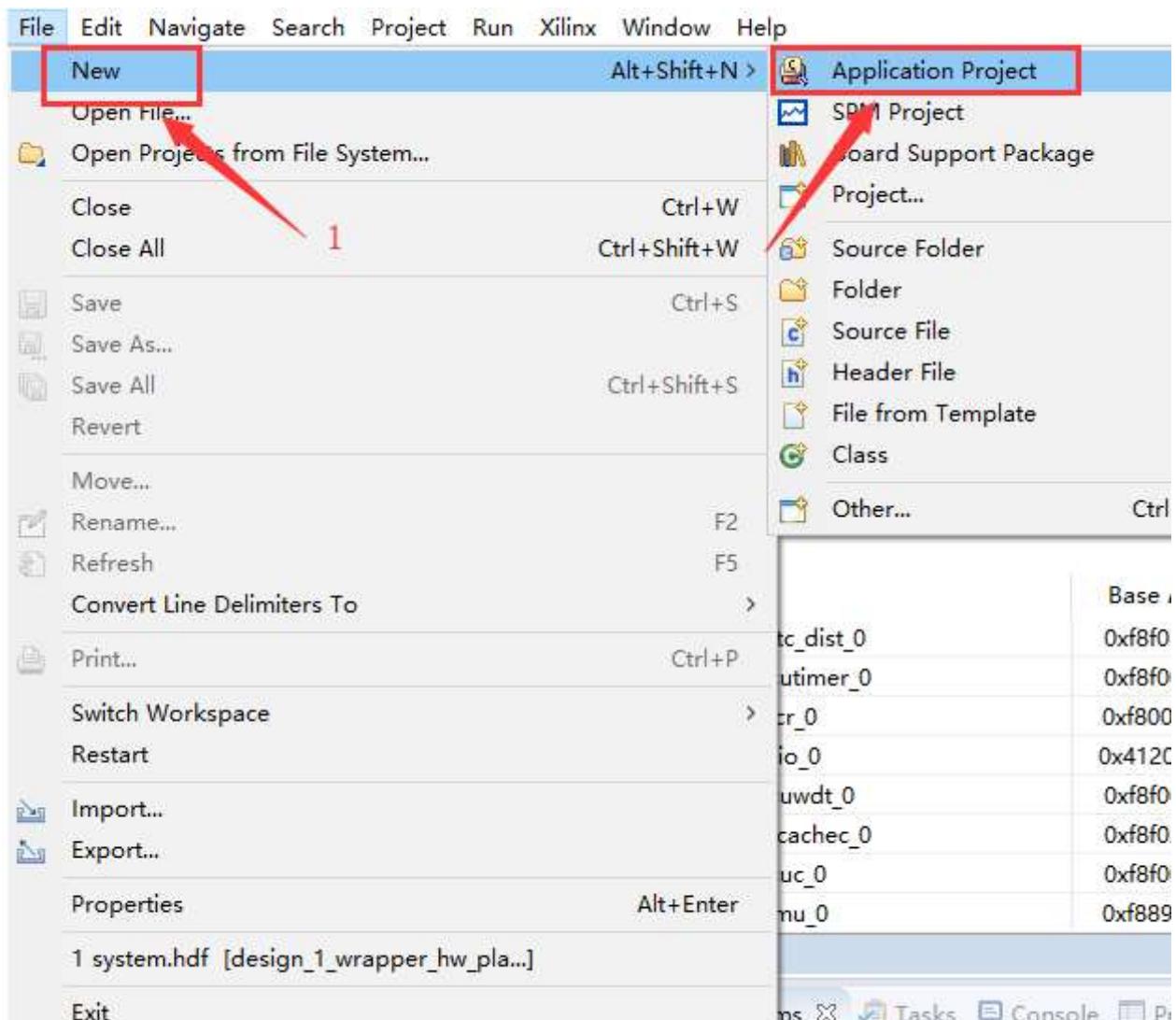
2) File→Launch SDK，在弹出的对话框中，保存默认，点击“OK”，如下图所示。



系统将自动打开SDK开发环境

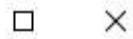


3) 新建一个工程 file→new→Application Project, 来新建一个“Application Project”，如下图所示。



4) 在新建工程名中输入自己的工程名称，点击NEXT

SDK New Project



## Application Project



Create a managed make application project.

Project name: **LED\_CODE**

Use default location

Location: E:\Tiny\_ZYNQ\06\_PS\_LED\_TEST\_AXI\_GPIO\PS\_LED\_AXI\_G

[Browse...](#)

Choose file system: default

OS Platform: standalone

### Target Hardware

Hardware Platform: ZYNQ\_CORE\_wrapper\_hw\_platform\_0 [New...](#)

Processor: ps7\_cortexa9\_0

### Target Software

Language:  C  C++

Compiler: 32-bit

Hypervisor Guest: N/A

Board Support Package:  Create New **LED\_CODE\_bsp**

Use existing



< Back

Next >

Finish

Cancel

[www.hellofpga.com](http://www.hellofpga.com)

5) Select the empty project and click Finish

## Templates

Create one of the available templates to generate a fully-functioning application project.



### Available Templates:

Dhrystone

**Empty Application**

Hello World

lwIP Echo Server

lwIP TCP Perf Client

lwIP TCP Perf Server

lwIP UDP Perf Client

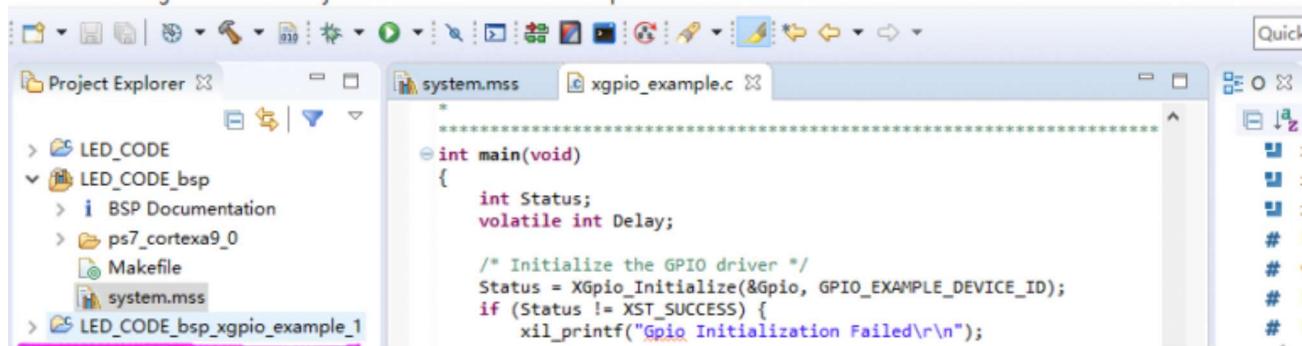
lwIP UDP Perf Server

Memory Tests

A blank C project.

- 4      5      6      7      8      9      10      11      12      13      14      15      16      17      18      19      20      21      22      23      24      25      26      27      28      29      30      31      32      33      34      35      36      37      38      39      40      41      42      43      44      45      46      47      48      49      50      51      52      53      54      55      56      57      58      59      60      61      62      63      64      65      66      67      68      69      70      71      72      73      74      75      76      77      78      79      80      81      82      83      84      85      86      87      88      89      90      91      92      93      94      95      96      97      98      99      100      101      102      103      104      105      106      107      108      109      110      111      112      113      114      115      116      117      118      119      120      121      122      123      124      125      126      127      128      129      130      131      132      133      134      135      136      137      138      139      140      141      142      143      144      145      146      147      148      149      150      151      152      153      154      155      156      157      158      159      160      161      162      163      164      165      166      167      168      169      170      171      172      173      174      175      176      177      178      179      180      181      182      183      184      185      186      187      188      189      190      191      192      193      194      195      196      197      198      199      200      201      202      203      204      205      206      207      208      209      210      211      212      213      214      215      216      217      218      219      220      221      222      223      224      225      226      227      228      229      230      231      232      233      234      235      236      237      238      239      240      241      242      243      244      245      246      247      248      249      250      251      252      253      254      255      256      257      258      259      260      261      262      263      264      265      266      267      268      269      270      271      272      273      274      275      276      277      278      279      280      281      282      283      284      285      286      287      288      289      290      291      292      293      294      295      296      297      298      299      300      301      302      303      304      305      306      307      308      309      310      311      312      313      314      315      316      317      318      319      320      321      322      323      324      325      326      327      328      329      330      331      332      333      334      335      336      337      338      339      340      341      342      343      344      345      346      347      348      349      350      351      352      353      354      355      356      357      358      359      360      361      362      363      364      365      366      367      368      369      370      371      372      373      374      375      376      377      378      379      380      381      382      383      384      385      386      387      388      389      390      391      392      393      394      395      396      397      398      399      400      401      402      403      404      405      406      407      408      409      410      411      412      413      414      415      416      417      418      419      420      421      422      423      424      425      426      427      428      429      430      431      432      433      434      435      436      437      438      439      440      441      442      443      444      445      446      447      448      449      450      451      452      453      454      455      456      457      458      459      460      461      462      463      464      465      466      467      468      469      470      471      472      473      474      475      476      477      478      479      480      481      482      483      484      485      486      487      488      489      490      491      492      493      494      495      496      497      498      499      500      501      502      503      504      505      506      507      508      509      510      511      512      513      514      515      516      517      518      519      520      521      522      523      524      525      526      527      528      529      530      531      532      533      534      535      536      537      538      539      540      541      542      543      544      545      546      547      548      549      550      551      552      553      554      555      556      557      558      559      550      551      552      553      554      555      556      557      558      559      560      561      562      563      564      565      566      567      568      569      570      571      572      573      574      575      576      577      578      579      580      581      582      583      584      585      586      587      588      589      580      581      582      583      584      585      586      587      588      589      590      591      592      593      594      595      596      597      598      599      590      591      592      593      594      595      596      597      598      599      600      601      602      603      604      605      606      607      608      609      610      611      612      613      614      615      616      617      618      619      610      611      612      613      614      615      616      617      618      619      620      621      622      623      624      625      626      627      628      629      630      631      632      633      634      635      636      637      638      639      630      631      632      633      634      635      636      637      638      639      640      641      642      643      644      645      646      647      648      649      640      641      642      643      644      645      646      647      648      649      650      651      652      653      654      655      656      657      658      659      650      651      652      653      654      655      656      657      658      659      660      661      662      663      664      665      666      667      668      669      660      661      662      663      664      665      666      667      668      669      670      671      672      673      674      675      676      677      678      679      670      671      672      673      674      675      676      677      678      679      680      681      682      683      684      685      686      687      688      689      680      681      682      683      684      685      686      687      688      689      690      691      692      693      694      695      696      697      698      699      690      691      692      693      694      695      696      697      698      699      700      701      702      703      704      705      706      707      708      709      7010      7011      7012      7013      7014      7015      7016      7017      7018      7019      7010      7011      7012      7013      7014      7015      7016      7017      7018      7019      7020      7021      7022      7023      7024      7025      7026      7027      7028      7029      7020      7021      7022      7023      7024      7025      7026      7027      7028      7029      7030      7031      7032      7033      7034      7035      7036      7037      7038      7039      7030      7031      7032      7033      7034      7035      7036      7037      7038      7039      7040      7041      7042      7043      7044      7045      7046      7047      7048      7049      7040      7041      7042      7043      7044      7045      7046      7047      7048      7049      7050      7051      7052      7053      7054      7055      7056      7057      7058      7059      7050      7051      7052      7053      7054      7055      7056      7057      7058      7059      7060      7061      7062      7063      7064      7065      7066      7067      7068      7069      7060      7061      7062      7063      7064      7065      7066      7067      7068      7069      7070      7071      7072      7073      7074      7075      7076      7077      7078      7079      7070      7071      7072      7073      7074      7075      7076      7077      7078      7079      7080      7081      7082      7083      7084      7085      7086      7087      7088      7089      7080      7081      7082      7083      7084      7085      7086      7087      7088      7089      7090      7091      7092      7093      7094      7095      7096      7097      7098      7099      7090      7091      7092      7093      7094      7095      7096      7097      7098      7099      70100      70101      70102      70103      70104      70105      70106      70107      70108      70109      70100      70101      70102      70103      70104      70105      70106      70107      70108      70109      70110      70111      70112      70113      70114      70115      70116      70117      70118      70119      70110      70111      70112      70113      70114      70115      70116      70117      70118      70119      70120      70121      70122      70123      70124      70125      70126      70127      70128      70129      70120      70121      70122      70123      70124      70125      70126      70127      70128      70129      70130      70131      70132      70133      70134      70135      70136      70137      70138      70139      70130      70131      70132      70133      70134      70135      70136      70137      70138      70139      70140      70141      70142      70143      70144      70145      70146      70147      70148      70149      70140      70141      70142      70143      70144      70145      70146      70147      70148      70149      70150      70151      70152      70153      70154      70155      70156      70157      70158      70159      70150      70151      70152      70153      70154      70155      70156      70157      70158      70159      70160      70161      70162      70163      70164      70165      70166      70167      70168      70169      70160      70161      70162      70163      70164      70165      70166      70167      70168      70169      70170      70171      70172      70173      70174      70175      70176      70177      70178      70179      70170      70171      70172      70173      70174      70175      70176      70177      70178      70179      70180      70181      70182      70183      70184      70185      70186      70187      70188      70189      70180      70181      70182      70183      70184      70185      70186      70187      70188      70189      70190      70191      70192      70193      70194      70195      70196      70197      70198      70199      70190      70191      70192      70193      70194      70195      70196      70197      70198      70199      70200      70201      70202      70203      70204      70205      70206      70207      70208      70209      70200      70201      70202      70203      70204      70205      70206      70207      70208      70209      70210      70211      70212      70213      70214      70215      70216      70217      70218      70219      70210      70211      70212      70213      70214      70215      70216      70217      70218      70219      70220      70221      70222      70223      70224      70225      70226      70227      70228      70229      70220      70221      70222      70223      70224      70225      70226      70227      70228      70229      70230      70231      70232      70233      70234      70235      70236      70237      70238      70239      70230      70231      70232      70233      70234      70235      70236      70237      70238      70239      70240      70241      70242      70243      70244      70245      70246      70247      70248      70249      70240      70241      70242      70243      70244      70245      70246      70247      70248      70249      70250      70251      70252      70253      70254      70255      70256      70257      70258      70259      70250      70251      70252      70253      70254      70255      70256      70257      70258      70259      70260      70261      70262      70263      70264      70265      70266      70267      70268      70269      70260      70261      70262      70263      70264      70265      70266      70267      70268      70269      70270      70271      70272      70273      70274      70275      70276      70277      70278      70279      70270      70271      70272      70273      70274      70275      70276      70277      70278      70279      70280      70281      70282      70283      70284      70285      70286      70287      70288      70289      70280      70281      70282      70283      70284      70285      70286      70287      70288      70289      70290      70291      70292      70293      70294      70295      70296      70297      70298      70299      70290      70291      70292      70293      70294      70295      70296      70297      70298      70299      702910      702911      702912      702913      702914      702915      702916      702917      702918      702919      702910      702911      702912      702913      702914      702915      702916      702917      702918      702919      702920      702921      702922      702923      702924      702925      702926      702927      702928      702929      702920      702921      702922      702923      702924      702925      702926      702927      702928      702929      702930      702931      702932      702933      702934      702935      702936      702937      702938      702939      702930      702931      702932      702933      702934      702935      702936      702937      702938      702939      702940      702941      702942      702943      702944      702945      702946      702947      702948      702949      702940      702941      702942      702943      702944      702945      702946      702947      702948      702949      702950      702951      702952      702953      702954      702955      702956      702957      702958      702959      702950      702951      702952      702953      702954      702955      702956      702957      702958      702959      702960      702961      702962      702963      702964      702965      702966      702967      702968      702969      702960      702961      702962      702963      702964      702965      702966      702967      702968      702969      702970      702971      702972      702973      702974      702975      702976      702977      702978      702979      702970      702971      702972      702973      702974      702975      702976      702977      702978      702979      702980      702981      702982      702983      702984      702985      702986      702987      702988      702989      702980      702981      70298

8) This automatically creates a reference routine



The screenshot shows the Xilinx Vivado IDE interface. On the left, the Project Explorer window displays a project named "LED\_CODE" with a sub-project "LED\_CODE\_bsp". Inside "LED\_CODE\_bsp", there are files like "BSP Documentation", "ps7\_cortexa9\_0", "Makefile", and "system.mss". A file named "LED\_CODE\_bsp\_xgpio\_example\_1" is highlighted with a pink underline. On the right, the main workspace shows two tabs: "system.mss" and "xgpio\_example.c". The "xgpio\_example.c" tab contains C code for a main routine that initializes a GPIO driver and prints a message if initialization fails.

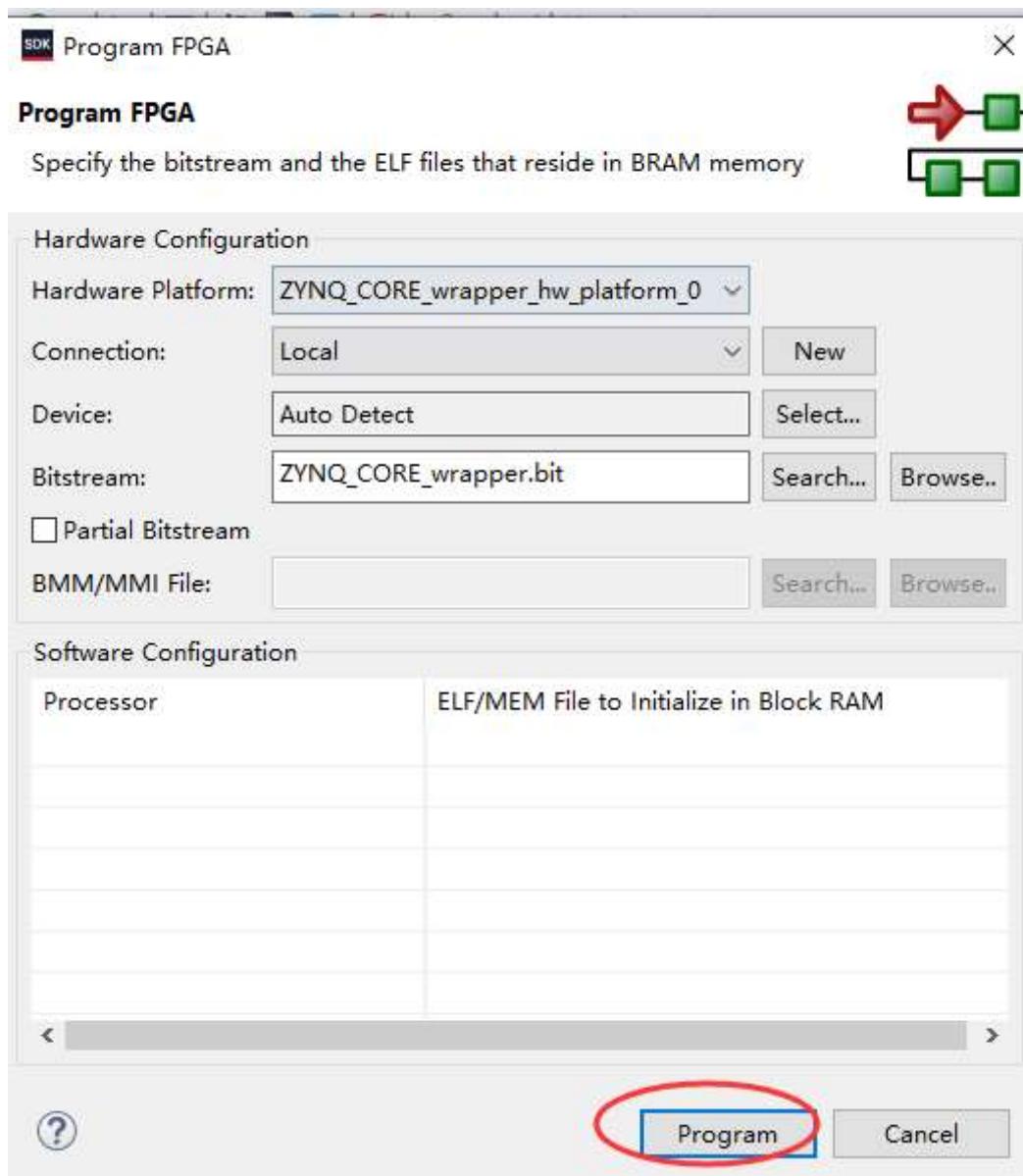
```
int main(void)
{
    int Status;
    volatile int Delay;

    /* Initialize the GPIO driver */
    Status = XGpio_Initialize(&Gpio, GPIO_EXAMPLE_DEVICE_ID);
    if (Status != XST_SUCCESS) {
        xil_printf("Gpio Initialization Failed\r\n");
    }
}
```

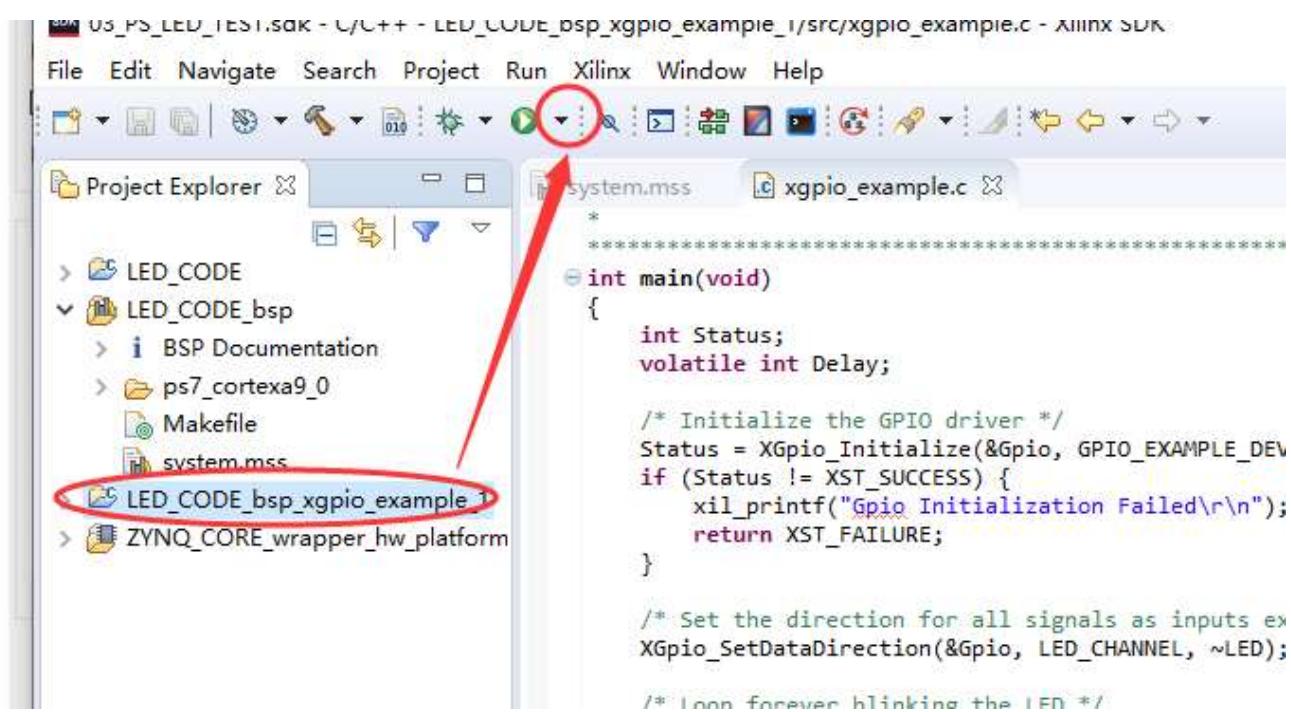
Note, you can directly download the reference design to see the effect, or you can refer to the code to add to your own main. Here we choose to download the reference code directly to see the effect

## 6. Download to the board for verification

Select the hardware platform in the project, right-click → Program FPGA, select Default in the pop-up dialog box, and click "program" to complete the Program work in the FPGA PL part

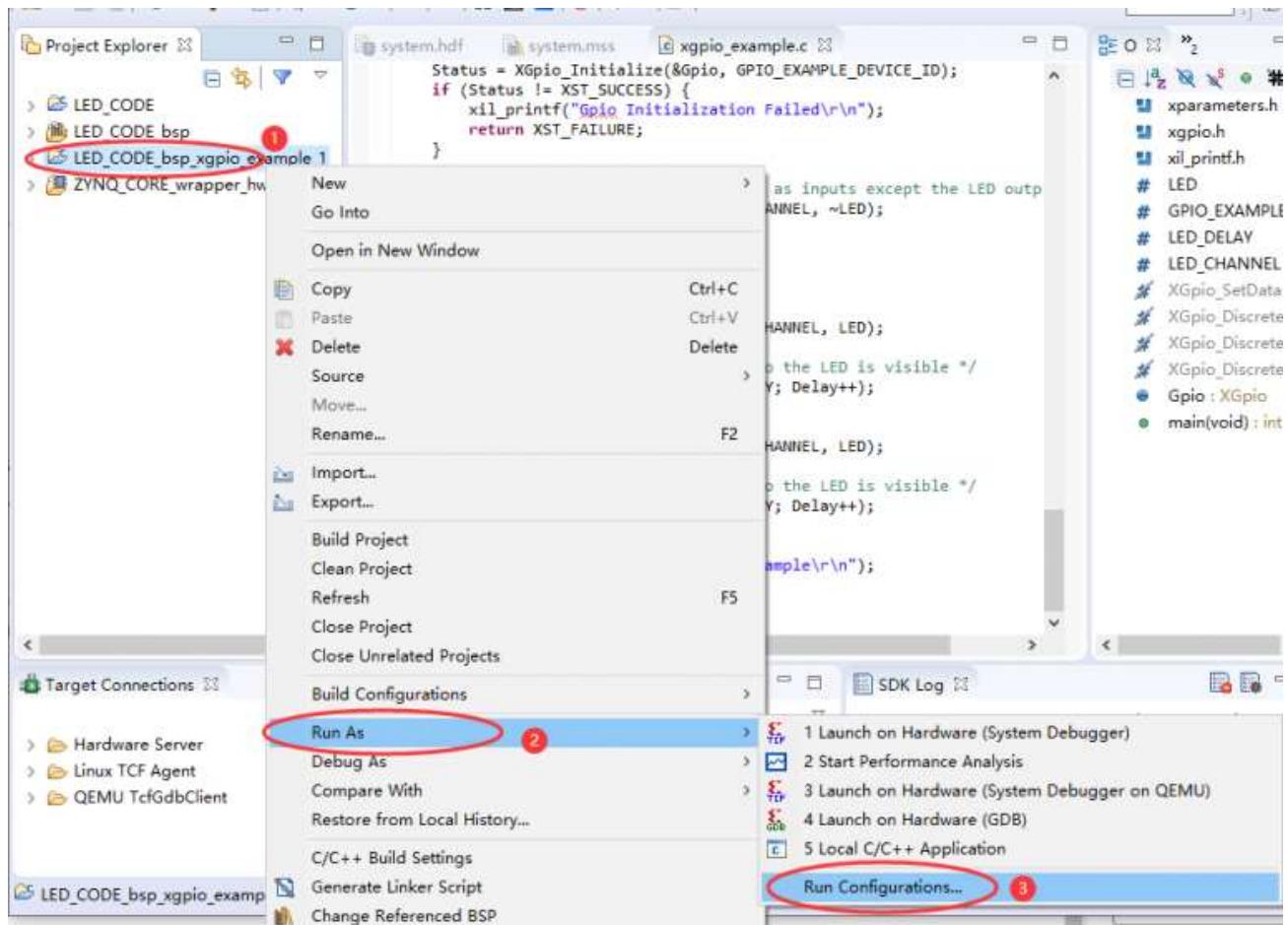


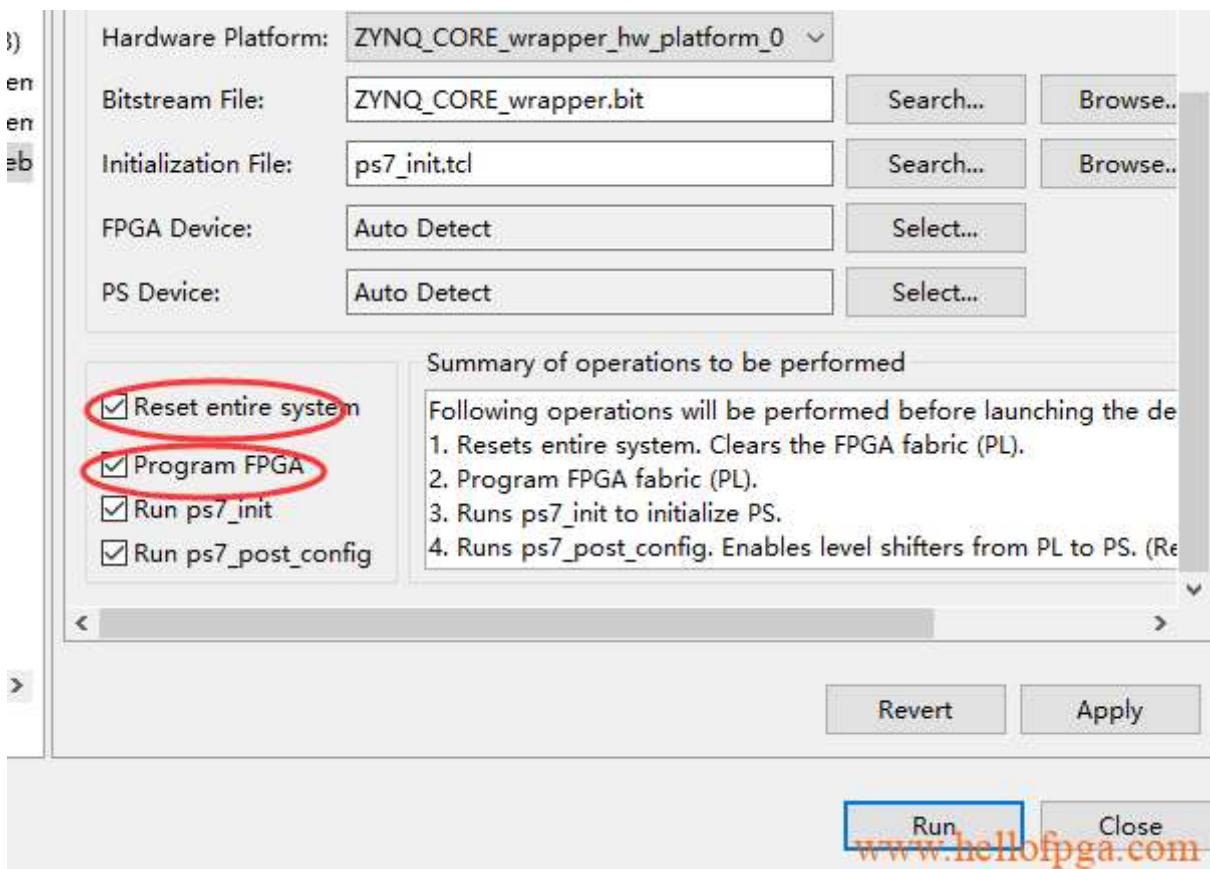
- 2) Select the GPIO project we generated, expand the icon to the right of the green arrow (RUN), and select Run As→1 Launch on Hardware (System Debugger)



You can see the LED1 light on the board blinking

Note: If an error pops up when RUN, you can follow the following operations to set up and then DEBUG





Then click APPLY and then select Run As→1 Launch on Hardware (System Debugger) to see if the download is successful

## Code interpretation

XGpio\_SetDataDirection 设置GPIO为输入/输出

XGpio\_DiscreteWrite 置位GPIO

XGpio\_DiscreteClear 拉低GPIO

#define LED 0x01

继续看代码，LED在代码中的定义为 0X01 即代表 LED 在LED\_CHANNEL中代表最低位 BIT0

看懂了上面这些 就很容易看懂程序里的代码

XGpio\_DiscreteClear(&Gpio, LED\_CHANNEL, LED); 代表拉低GPIO LED\_CHANNEL中的 第0位

XGpio\_DiscreteWrite(&Gpio, LED\_CHANNEL, LED); 代表拉高GPIO LED\_CHANNEL中的 第0位

XGpio\_SetDataDirection(&Gpio, LED\_CHANNEL, ~LED); 设置为输出模式

for (Delay = 0; Delay < LED\_DELAY; Delay++); 这是一个耗费系统资源的delay函数的简写

所以整个程序的效果就是LED 1 不停的点亮熄灭，反复循环。

如果想要将1个灯改成2个灯 只需要将

#define LED 0x01 更改为 #define LED 0x03即可， 07代表 0000\_0111即使 bit0 bit1 bit2都置1， 修改之后 3个灯就能同步进行变换

看懂这些 可以自己创建一个main函数 然后在里面参考这个例子写驱动LED的代码（本文中介绍一下如果利用参考设计来创建工程）

完整工程如下：(板子有多个版本，请按照对应板子的芯片型号进行下载测试)

[06\\_PS\\_LED\\_TEST\\_AXI\\_GPIO\\_XC7Z020](#)

**Download**

The following website is my own compilation of the different points of MIO EMIO AVIGPIO, if you are interested, you can take a look

[http://www.hellofpga.com/index.php/2021/08/08/zynq\\_emio\\_mio\\_axi\\_gpio/](http://www.hellofpga.com/index.php/2021/08/08/zynq_emio_mio_axi_gpio/)

In addition, if you want to know about EMIO to drive GPIOs, you can see the next project (Engineering 7)

 **SMART ZYNQ SP & SL**