☆Tip：利用word查找功能输入关键词可快速找到需要的文段。

1.ISE各个窗口的名称

Start panel:没什么新奇，它就是start，功能一目了然。

Design panel:view pan(这里有implementation,simulation两个check box），Hierarchy Pane（这里显示相应design view下的文件结构），process pan（在hierarchy pan里面选择不同source，这里内容会有不同）

Files pane:project里面所有文件的列表。

Library Panel:library和source等的依赖关系在这里一目了然。

Errors/Warning Panel:明显是用来显示warning和error的。最好常开着。

Work space：打开的text editor所在的地方。

Design Summary/Report Viewer ：ise还提供了生成很多设计相关内容报告的功能，还没用过。

上述panel在，view->panel里面可以check打开或关闭。

其他， PlanAheadTM, ISim,third-party text editors, XPower Analyzer, and iMPACT这些会在独立窗口中打开。

1. "The Top module has not been specified" error

按照ise quick start新建好project，添加source文件，然后用simulator进行了仿真。一切顺利，但是要进行pin map的时候，遇到了这个问题。

问题的起因不明，但是xilinx的论坛上提到按下面操作可以解决问题：

a）fill in the top module content

b）uncheck the “manual compile order”

第一个不知道怎么操作，但是第二个可行，在project视图下，其实是有两个 module的，一个显示为你前面选择的器件名，另一个是你前面填的project名。在“器件名”上右键，uncheck “manual compile order”就可以进行synthesize-XST了。

没有进行synthesize-XST是不能通过constraint editor来进行pin map的。

1. 解决ISE找不到pdf阅读器的问题

第二个是建立名为gmake的软连接到make，第一个是建立名为acroread的软连接到evince。因为ISE默认使用gmake和acroread。

# sudo ln -s /usr/bin/evince /usr/bin/acroread

# sudo ln -s /usr/bin/make /usr/bin/gmake

1. Pin assignment

设计输入差不多完成时需要对设计进行pin assignment。对CPLD，可以用PACE；对于FPGA，可以用PlanAhead。

PlanAhead有使用视频：<http://www.xilinx.com/products/design>

PlanAhead中进行Pin assignment的具体步骤：

1）window->I/O ports，选择具体的输入输出，在site的下拉菜单中选择location。

2）bus或者是单个输入输出，右键，在菜单中选择“configure i/O ports”，来设置端口电平，拉低还是拉高等。

设置完成不要忘记保存。

3）unused pin是generate programming file的一个property，修改即可。

另外，使用PlanAhead还可以进行partial reconfiguration设计。

1. 使用ISE工作流程

Design Entry->Design synthesis->Design implementation->Device programming->Debug hardware

1）Design Entry，设计输入：包括新建project，添加或新建 source，设计规则输入。

2）Design synthesis，设计综合：将逻辑转化为xilinx FPGA相关的综合net\_list。包括设置timing constraints， pin map等。

3）Design implementation，设计执行：将综合后netlist翻译成FPGA内部代码，将所有功能单元map到实际硬件（physical representation）。

4）Device programming，烧写程序：将implementation完成后，生成bit文件，使用impack将bit文件下载到FPGA。

5）Debug hardware，板上调试：测试烧写好的程序。

前仿真：对Project中的source进行逻辑验证。

后仿真：拿综合了FPGA内部延迟等信息的文件，进行时序逻辑等验证。

1. 使用Chip Scope六步骤：

1）生成一个integrated controller core, ICON.

使用core generator。

2）生成一个或多个integrated logical analyzer, ILA.

使用core generator。

3）链接ILA和ICON，并添加到project，使之成为project的一部分。

主要有以下内容：

a.add source，将ILA和ICON的core加入project。

b.为ILA声明一个36位的bus。如，wire [35:0] ILAControl

c.ICON module例化，每个project只能有一个ICON module。 icon i\_icon(.control0(ILAControl))

d.ILA module例化，根据项目需要，可以放多个ILA module。 ila i\_ila(.control(ILAControl), .clk(Clock),.trig0(/\*the trigger \*/,.data(/\*the data will be captured when triggered/\*))

4）将包含有ILA，ICON的project综合及布线 。

5）程序烧写。

6）使用ChipScope软件链接ILA并使用ILA。（ChipScope需要通过ICON与ILA通信）

找个地记一下：新系统下，在pin assignment的时候，发现go ahead没法使用。

错误：

PlanAhead/lib/lnx32.o/librdi\_commontasks.so: libncurses.so.5: cannot open shared object file: No such file or directory

发现这个错误是因为Ncurses平台，所以，又装了下ncurses，具体方式：sudo apt-get install libncurses5-dev