

HUNT ENGINEERING Chestnut Court, Burton Row, Brent Knoll, Somerset, TA9 4BP, UK Tel: (+44) (0)1278 760188, Fax: (+44) (0)1278 760199, Email: sales@hunteng.co.uk http://www.hunteng.co.uk http://www.hunteng.co.uk





Using Different Versions of ISE

Document version 1.0 R.Williams 05/05/05

HUNT ENGINEERING is a trading style of HUNT ENGINEERING (U.K.) Ltd, Co Reg No 3333633 Directors P.Warnes & N.J.Warnes. Reg'd office 34 & 38 North St, Bridgwater, Somerset TA6 3YD. VAT Reg'd No GB 515 8449 31 This documentation and the product it is supplied with are Copyright HUNT ENGINEERING 2005. All rights reserved. HUNT ENGINEERING maintains a policy of continual product development and hence reserves the right to change product specification without prior warning.

WARRANTIES LIABILITY and INDEMNITIES

HUNT ENGINEERING warrants the hardware to be free from defects in the material and workmanship for 12 months from the date of purchase. Product returned under the terms of the warranty must be returned carriage paid to the main offices of HUNT ENGINEERING situated at BRENT KNOLL Somerset UK, the product will be repaired or replaced at the discretion of HUNT ENGINEERING.

Exclusions - If HUNT ENGINEERING decides that there is any evidence of electrical or mechanical abuse to the hardware, then the customer shall have no recourse to HUNT ENGINEERING or its agents. In such circumstances HUNT ENGINEERING may at its discretion offer to repair the hardware and charge for that repair.

Limitations of Liability - HUNT ENGINEERING makes no warranty as to the fitness of the product for any particular purpose. In no event shall HUNT ENGINEERING'S liability related to the product exceed the purchase fee actually paid by you for the product. Neither HUNT ENGINEERING nor its suppliers shall in any event be liable for any indirect, consequential or financial damages caused by the delivery, use or performance of this product.

Because some states do not allow the exclusion or limitation of incidental or consequential damages or limitation on how long an implied warranty lasts, the above limitations may not apply to you.

TECHNICAL SUPPORT

Technical support for HUNT ENGINEERING products should first be obtained from the comprehensive Support section <u>www.hunteng.co.uk/support/index.htm</u> on the HUNT ENGINEERING web site. This includes FAQs, latest product, software and documentation updates etc. Or contact your local supplier - if you are unsure of details please refer to <u>www.hunteng.co.uk</u> for the list of current re-sellers.

HUNT ENGINEERING technical support can be contacted by emailing support@hunteng.demon.co.uk, calling the direct support telephone number +44 (0)1278 760775, or by calling the general number +44 (0)1278 760188 and choosing the technical support option.

Document History

| 1.0 | 05/05/05 | First Written |
|-----|----------|---------------|
| | | |

TABLE OF CONTENTS

| INTRODUCTION | 5 |
|---|---|
| HOW TO USE THIS DOCUMENT | |
| UPGRADING TO THE LATEST VERSION OF ISE | |
| CONTINUING TO USE A PREVIOUS VERSION OF ISE | |
| CREATING NEW PROJECTS | |
| UPGRADING EXISTING PROJECTS TO ISE 7.1 | 7 |
| CREATING PROJECTS FOR ISE 4 | |
| CONVERTING EXAMPLE1 FOR YOUR MODULE TYPE | 9 |
| Step 1: Copying Example1 from the HUNT ENGINEERING CD | |
| Step 2: Creating a New Project File | |
| Step 3: Adding VHDL Source to the New Project | |
| Step 4: Adding User Constraints | |
| Step 5: Adding a Simulation Test-Bench | |
| Step 6: Setting Project Build Options | |
| Step 7: Building the Project | |
| CREATING PROJECTS FOR ISE 5 | |
| Converting Example1 for your Module Type | |
| Step 1: Copying Example1 from the HUNT ENGINEERING CD | |
| Step 2: Creating a New Project File | |
| Step 3: Adding VHDL Source to the New Project | |
| Step 4: Adding User Constraints | |
| Step 5: Adding a Simulation Test-Bench | |
| Step 6: Setting Project Build Options | |
| Step 7: Building the Project | |
| CREATING PROJECTS FOR ISE 6 | |
| CONVERTING EXAMPLE1 FOR YOUR MODULE TYPE | |
| Step 1: Copying Example1 from the HUNT ENGINEERING CD | |
| Step 2: Creating a New Project File | |
| Step 3: Adding VHDL Source to the New Project | |
| Step 4: Adding User Constraints | |
| Step 5: Adding a Simulation Test-Bench | |
| Step 6: Setting Project Build Options | |
| Step 7: Building the Project | |
| CREATING PROJECTS FOR ISE 7 | |
| Converting Example1 for your Module Type | |
| Step 1: Copying Example1 from the HUNT ENGINEERING CD | |
| Step 2: Creating a New Project File | |
| Step 3: Adding VHDL Source to the New Project | |
| Step 4: Adding User Constraints | |
| Step 5: Adding a Simulation Test-Bench | |
| Step 6: Setting Project Build Options | |
| Step 7: Building the Project | |

For users of FPGA modules there are many example projects provided on the HUNT ENGINEERING CD. These projects are intended to be the starting point when making a new FPGA design with any of the HUNT ENGINEERING FPGA modules. All of the example projects are designed to be used with the standard tool-set provided by Xilinx (there is a separate application note 'Using non ISE development tools' available for users working with development tools such as Leonardo Spectrum or Synplicity).

The standard FPGA design tool from Xilinx is the integrated design environment 'ISE'. Over time Xilinx make improvements to the ISE product and release major version changes. As each new version of design tools is released HUNT ENGINEERING adapt the standard FPGA module examples so they continue to work as expected.

Each time a new version of ISE is released, users may either continue working with their current version of tools, or upgrade to the new version.

Whether continuing with their current version of ISE or upgrading, this document describes what must be done to ensure that the HUNT ENGINEERING FPGA examples continue to work as expected. In addition, this document describes how to make a brand new FPGA project in ISE.

The following three sections describe how to use this document, depending on whether you are upgrading ISE, continuing with your existing version, or making a new project from scratch.

Upgrading to the Latest Version of ISE

The latest version of ISE is version 7.1. The FPGA module examples on the current HUNT ENGINEERING CD are written to use ISE 7.1. For users upgrading to ISE 7.1 this simply means that any of the examples on the *latest* CD can be copied and used without any issue of upgrading the project. In this situation therefore, the remainder of this document does not apply.

However, for users who are upgrading to ISE 7.1, any existing project based around FPGA modules will need to be upgraded.

To upgrade an existing FPGA module project to ISE 7.1 you will need to work through the section 'Upgrading Existing Projects to ISE 7.1'

Continuing to Use a Previous Version of ISE

The latest version of ISE and HUNT ENGINEERING CD examples is version 7.1. If you are continuing to use a version of ISE earlier than ISE 7 and have an existing project then you may continue development without needing to work through the remainder of this document.

If however, you wish to work with any of the new example projects on the current HUNT ENGINEERING CD you will need to work through one of the following sections depending on your current tool version. The section that applies will be one of the following, 'Creating Projects in ISE 4', 'Creating Projects in ISE 5', 'Creating Projects in ISE 6' or 'Creating Projects in ISE 7'.

Creating New Projects

To create a brand new FPGA project in ISE, you will need to work through the appropriate section from either 'Creating Projects in ISE 4', 'Creating Projects in ISE 5', 'Creating Projects in ISE 6' or 'Creating Projects in ISE 7', depending on the version of ISE you are using. You will also need to refer to the 'Making your own FPGA design' section of the User Manual for the FPGA module you are using for module specific information.

This section describes how to upgrade an existing FPGA module project to work with the latest version of ISE, version 7.1.

The first step in upgrading to ISE 7.1 is to ensure you have installed the latest service pack. In the initial release of ISE 7, the project file import process for converting old projects failed to correctly read the 'Hierarchy Separator' setting in the 'Synthesis Options' of the existing project. This was fixed in Service Pack 1. Please ensure you have completed the installation of the latest service pack before continuing through this section.

The next step in upgrading an existing project is to replace the contents of the 'Common' directory with the latest FPGA support from HUNT ENGINEERING. Using either the latest HUNT ENGINEERING CD or by visiting the User Area of the HUNT ENGINEERING web-site <u>www.hunteng.co.uk</u> download the IP for the version of FPGA module you are using to a directory on your local drive. With this done, replace the contents of the FPGA projects 'Common' directory with the latest 'Common' FPGA IP.

Now you are ready to open the existing project in ISE. Open the ISE 7.1 Project Navigator and click 'File \rightarrow Open Project...'. Select the project file you wish to open.

You should then see a window similar to that shown below.

| Update P | roject 🔀 | | | |
|----------|---|--|--|--|
| ? | D:\fpga5v1\Example1\ISE\Ex1_Fpga5v.npl | | | |
| ч | This project was generated by a previous version of Project Navigator and must be updated to the new project format. Once this project has been updated you will no longer be able to open it in earlier versions of Project Navigator. | | | |
| | The existing project will automatically be archived and stored at D:\fpga5v1\Example1\ISE\Ex1_Fpga5v_jse6_bak.zip in the event you later decide to return to the older version. | | | |
| | Please note that starting in 7.1i, the new extension for the project file will be .ise. This project will be converted to Ex1_Fpga5v.ise. | | | |
| | Do you want to update the project? | | | |
| | Yes No Help | | | |

This window indicates that the project needs updating. Also note that the project file extension changes from '.npl' to '.ise' with version 7. Click 'Yes' to continue.

You will now be able to continue development on the converted project in ISE 7.1.

For each FPGA module type, each project contained on the CD provides all of the required design elements to ensure correct operation on that chosen module type. These elements include a Hardware Interface Layer that is used to correctly control external devices, user constraints information to control design timing and pin location and example VHDL to provide a structured starting point.

For users of ISE 4 design tools this document must be followed in order to convert the newer project format used on the HUNT ENGINEERING CD to the correct ISE 4 format. This section describes how to build an ISE 4 project from scratch, using the design source provided on the CD. This process is necessary as the standard XILINX tool version has moved on from ISE 4 and XILINX provide no mechanism for converting post ISE 4 projects back to ISE 4.

This document uses Example1 as the starting point for the creation of a new project. It is important to start from one of the standard examples provided on the HUNT ENGINEERING CD as this give the correct starting point for FPGA development with your FPGA module.

Please note: if you are using this section in order to create a brand new project with functionality that does not match any of the standard CD examples, then you must still start from Example1. Once you have created a correct project based around Example1 you may then insert your own unique code into the User-Ap entity, removing all unwanted logic. When doing this, you will need to refer to the relevant information in the 'Making your own FPGA design' section of your FPGA User Manual.

Converting Example1 for your Module Type

The HUNT ENGINEERING CD provides support for many different FPGA module types. For each module there is always a standard Example1 project provided on the CD. This example is the 'Getting Started' example for each module type and should always be used as the first step in FPGA development.

This section describes how to convert Example1 for the HERON-FPGA5 as an example of the conversion process. Although the design source varies for each module type, the principles shown here are the same regardless of type.

Step 1: Copying Example1 from the HUNT ENGINEERING CD

The first step in the conversion process is to copy Example1 onto your local hard drive. Make a directory on your local drive in which to store Example1.

On the HUNT ENGINEERING CD, all FPGA examples are provided below the 'fpga' directory. The 'fpga' directory is divided into sub-directories that reflect the name of the module type. In this document, Example1 for the HERON-FPGA5 is to be converted, so the directory 'fpga/fpga5v1/' will contain the appropriate project information. Identify the appropriate CD directory according to your module type.

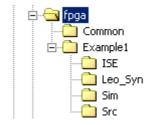
With the correct CD directory located you will need to copy Example 1 to your local drive. This can be done in one of two ways. The first method is to copy the Example 1 project by hand for the module type you are using. The second method is to unzip the ZIP file for that module type onto your hard drive.

When using the first approach you will need to copy the 'Common' directory and 'Example1' directory from the CD to a chosen directory on your local drive. The 'Example1' directory contains all design source that forms Example1 and the 'Common' directory provides design source common to all designs made for that module.

Please note, when copying files from your CD by hand you may need to edit the file attributes after copying. This is because on some operating systems Read-Only files on the CD will become Read-Only files on your local drive. All files in the 'Example1' directory should be set to Read-Write while all files in the 'Common' directory must remain Read-Only.

When using the second approach you will be unzipping all examples for that module, including Example1. When unzipping all project examples, the read-write attributes will automatically be set correctly.

The picture below shows a new directory on the local drive named 'fpga'. In the 'fpga' directory are the copied 'Common' and 'Example1' directories.



Step 2: Creating a New Project File

With the Example1 example directory and Common directory copied onto your Local Drive you can now begin building an ISE 4 project.

Open the ISE 4 Project Navigator if it is not already open. Please note, in the remainder of this section an example conversion is shown for Example 1 for the HERON-FPGA5 where this conversion is performed in ISE 4.1. If you are using a different ISE 4 version then although the windows shown may not perfectly match, the conversion process is still the same.

Next, delete the project file for the Example1 project in the new directory you have created on your local drive. The project file will be located in the 'ISE' sub-directory of the 'Example1' directory and will have the extension '.ise'. This existing project file must be removed as it will not be useable in ISE 4 and must be replaced with an appropriately constructed project file.

With the existing file deleted, select the menu item 'File \rightarrow New Project...' to begin creating a new project. The following window should be displayed.

| New Project | | | x |
|--|---|-------------|---|
| Project <u>N</u> ame: Ex1_Fpga5v | Project <u>L</u> ocatior c:\fpga5v1\Ex | | |
| Project Device Options: Property Name | | Value | |
| Device Family | | Virtex2 | |
| Device | | xc2v1000 | |
| Package | | fg456 | |
| Speed Grade | | -4 | |
| Design Flow | | XST VHDL | |
| | OK | Cancel Help | |

Enter the correct project name in the 'Project Name' field. For Example1 this should be set to be the same project name as the Example1 project file on the HUNT ENGINEERING CD (minus the .ise extension).

Next, enter the correct location into the 'Project Location' field. This field should match the location of the 'Example1/ISE' directory you have made on your local drive.

Then set the Project Device Options to correctly reflect the appropriate device information according to the module type you are using. If you are unsure of the correct information refer to the User Manual for that module type. Ensure that the Design Flow is 'XST VHDL'.

When you have correctly defined all fields click 'OK'.

Step 3: Adding VHDL Source to the New Project

The next step is to add design source, starting with the top level of the hierarchy. For all FPGA module projects the top level of the design is always contained in the file 'top.vhd'. 'top.vhd' is always provided as part of the 'Common' directory for each module type.

Select the menu item 'Project \rightarrow Add Source...'. The following window will be displayed where you will need to navigate to the 'Common' directory that was created on your local drive. Select the file 'top.vhd' and click 'Open'.

| Add Existing Sources | | | ? × |
|---|---|-------------------|-----|
| Look in: 🔂 Common | | - 🗢 🗈 💣 🎟 | |
| Fpga5v_tpl.ucf HE_CONV.vhd HE_RWCLK.vhd HE_SDRAM.vhd HE_USER.vhd HE_USER.vhd HE_WR_6F.vhd | SIM_MSG.vhd SIM_RD_6F.vhd SIM_SDRAM.vhd SIM_WR_6F.vhd SIM_WR_6F.vhd SYN_V2_RD_6F.vhd | JUSER_AP_TPL.vhd | |
| File name: TOP.vhd | | <u></u> pe | n |
| Files of type: Sources | (*.txt;*.vhd;*.vhdl;*.v;*.abl;* | f.xco;*.sc ▼ Cano | |

The following window will then appear. Select 'VHDL Module' and click OK to add 'top.vhd' to the project.

| Choose Source Type | × |
|---|--------|
| TOP.vhd is which source type? The suffix is ambiguous as to type | |
| VHDL Module VHDL Package VHDL Test Bench | Cancel |
| | Help |

With this done, the Module View window will now look similar to the picture below, depending on the particular source files that are required by the module type you are using:

| Sources in Project: |
|----------------------------------|
| 🚎 🧧 Ex1_Fpga5v |
| Ė~ 🛄 xc2v1000-4fg456 |
| 🗄 🐨 📝 top-rtl (\\Common\TOP.vhd) |
| |
| 🚽 📝 he_rwclk |
| |
| 🚽 📝 he_user |
| 🚽 📝 he_wr_6f |
| user_ap |
| |
| |

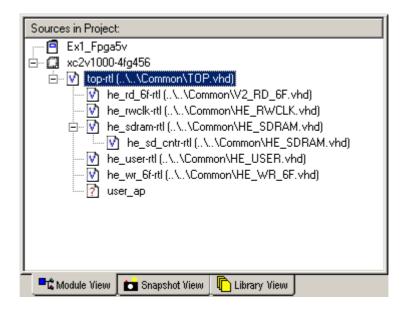
In the case of the FPGA5v1 example conversion, after adding 'top.vhd' we can now see 6 more design entities are needed. One of these entities is the User-Application level of the design which contains the VHDL that makes this example the getting started example, Example1. The other entities shown are part of the Hardware Interface Layer and are found in the 'Common' directory.

First add the appropriate files from the 'Common' directory in order to replace the red question mark icons with correct entities. To do this, you will again need to use the 'Add Source...' menu item. Navigate to the 'Common' directory and select the files that have names that match the entities in the Module View of your new project.

Please note: the HE_RD_6F entity is provided in the 'Common' directory in the source file 'V2_RD_6F'.

When adding each file, the correct Source Type will be 'VHDL Module' as was selected when adding 'top.vhd'.

When you have added all required Hardware Interface Layer components the Module View should look similar to the picture below for the FPGA5v1 conversion:



Next you will need to navigate to the 'Src' directory of Example1 and add the User_Ap file to the project:

| Add Existing | Sources | <u>?</u> × |
|-------------------------|--|------------|
| Look in: 🔁 | Src 💌 🗢 🖆 🎫 - | |
| HSB1.vhd | | |
| HSB1.vhd | | |
| TB_EX1.vh | | |
| | | |
| | | |
| | | |
| , File <u>n</u> ame: | User_Ap1.vhd | n |
| | | <u> </u> |
| Files of <u>type</u> : | Sources (*.txt)*.vhd)*.vhd)*.v;*.abl;*.xco;*.sc 💌 🛛 Cano | :el |

After the 'user_ap' entity has been added to the project you will see which other source files are required below this entity. Add the appropriate source files from the 'Src' directory.

After you have added all of required VHDL source files from the 'Src' directory there may still be red question marks against entities in the hierarchy. These entities will correspond to Core Gen components that are placed in the Example1 'ISE' directory. In the case of Example1 there is one Core Gen component called 'fifo15x32' that must be added to the project from the 'ISE' directory.

With all relevant VHDL source added to the project the Module View should look similar to the picture shown below for the FPGA5v1 conversion:

| Sources in Project: | | |
|--|--|--|
| 🚎 🖻 Ex1_Fpga5v | | |
| Ė∽ 🛄 xc2v1000-4fg456 | | |
| config (\Src\User_Ap1.vhd) | | |
| 🖻 📝 [top-rtl (\\Common\TOP.vhd] | | |
| ├── 📝 he_rd_6f-rtl (\\Common\V2_RD_6F.vhd) | | |
| | | |
| 📄 🖳 🙀 he_sdram-rtl (\\Common\HE_SDRAM.vhd) | | |
| he_sd_cntr-rtl (\\Common\HE_SDRAM.vhd) | | |
| ├── 💇 he_user-rtl (\\Common\HE_USER.vhd) | | |
| ├── 💇 he_wr_6f-rtl (\\Common\HE_WR_6F.vhd) | | |
| 🖻 🖓 user_ap-example1 (\Src\User_Ap1.vhd) | | |
| fifo15x32 (fifo15x32.xco) | | |
| hsb1-rtl (\Src\HSB1.vhd) | | |
| | | |
| 🗖 📲 Module View 💼 Snapshot View 🖺 Library View | | |

Step 4: Adding User Constraints

The next step is to add user design constraints to the project. Using 'Project \rightarrow Add Source...' add the user constraints file contained in the 'ISE' directory of the project.

| Add Existing 9 | Sources | <u>?</u> × |
|--------------------|---|------------|
| Look jn: 🔁 | ISE 💌 🗲 🖻 📸 🎫 | |
| projnav | | |
| Ex1_Fpgas | | |
| 🗒 fifo15x32. | | |
| | | |
| | | |
| I | | |
| File <u>n</u> ame: | Ex1_Fpga5v.ucf Ope | n |
| Files of type: | Sources (*.txt;*.vhd;*.xco;*.sch;*.tbw;*.bmm;* Cano | ;el |
| | | 111 |

Select the UCF file and click 'Open'. The following window will appear to allow you to specify the design file to which the user constraints should be associated. Highlight 'top' and click 'OK'.

| Associate with Source | × |
|--|-------------------|
| Associate Ex1_Fpga5v.ucf with th affects. | ne source that it |
| he_sd_cntr top | OK |
| he_rd_6f he_rwclk | Cancel |
| he_wr_6f user_ap | Help |

Step 5: Adding a Simulation Test-Bench

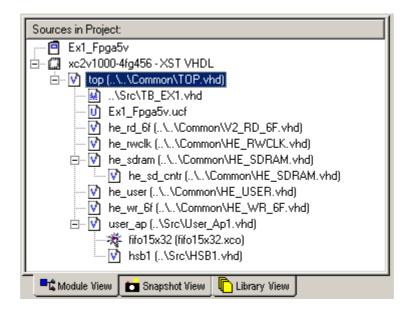
Typically, a project will also include a test-bench for simulation. For Example1 there is a test-bench provided in the 'Src' directory of the project. This file must also be added to the project using 'Project \rightarrow Add Source...'. Navigate to the 'Src' directory, highlight the file that beings 'TB_' and click 'Open'.

| Add Existing | Sources | ? × |
|-------------------------|--|------|
| Look jn: 🔁 |) Src 💌 🗢 🖆 🎫 | |
| HSB1.vhd | | |
| 📕 sim_fifo.vł | | |
| TB_EX1.vh | | |
| | | |
| | | |
| | | |
| , File <u>n</u> ame: | TB_EX1.vhd Ope | |
| nie <u>n</u> ame. | TB_EX1.vhd | in i |
| Files of type: | Sources (*.txt;*.vhd;*.vhd;*.v;*.abl;*.xco;*.sc 💌 Cano | el |
| | | // |

The following window will appear where you need to specify the source type. Select 'VHDL Test Bench' and click 'OK'.

| Choose Source Type | × |
|---|--------|
| TB_EX1.vhd is which source type? The suffix is ambiguous as to type. | ? |
| VHDL Module VHDL Package | ОК |
| VHDL Test Bench | Cancel |
| | Help |
| | |

At this point the Module View should look similar to the picture shown below for the FPGA5v1 conversion process.



Step 6: Setting Project Build Options

The last step in creating a new project is to apply the necessary project settings that will allow the design to be built correctly.

Ensure the file 'top.vhd' is highlighted in the Module View, and then right click on 'Synthesize' in the Process View. Select 'Properties...' to open the following window. With the 'Synthesis Options' tab at the front, check that the settings match those shown below.

| Property Name | Value |
|--------------------------------|--------------|
| Optimization Goal | Speed 💌 |
| Optimization Effort | Normal |
| Synthesis Constraints File | |
| Use Synthesis Constraints File | |
| Keep Hierarchy | |
| Global Optimization Goal | AllClockNets |
| Generate RTL Schematic | Yes |
| Read Cores | |
| Write Timing Constraints | |
| Cross Clock Analysis | |
| Hierarchy Separator | _ |
| Bus Delimiter | |
| Slice Utilization Ratio | 100 |
| Case | Lower |
| VHDL Work Directory | ./xst |
| VHDL INI File | |

Next bring the 'HDL Options' tab to the front and check that the settings match those in the picture below.

| Property Name | Value |
|---------------------------------|----------|
| FSM Encoding Algorithm | Auto 🔽 |
| RAM Extraction | V |
| RAM Style | Auto |
| ROM Extraction | |
| ROM Style | Auto |
| Mux Extraction | Yes |
| Mux Style | Auto |
| Decoder Extraction | |
| Priority Encoder Extraction | Yes |
| Shift Register Extraction | |
| Logical Shifter Extraction | |
| XOR Collapsing | |
| Resource Sharing | |
| Complex Clock Enable Extraction | |
| Multiplier Style | Auto |

Next bring the 'Xilinx Specific Options' tab to the front. Change the 'Number of Clock Buffers' item to '0', and set the 'Pack I/O Registers into IOBs' item to 'No'. With this done, check that the settings match those in the picture below.

| Synthesis Options HDL Options Xilinx Specific Op | · |
|--|-------|
| Property Name | Value |
| Add I/O Buffers | V |
| Max Fanout | 500 |
| Number of Clock Buffers | 0 |
| Register Duplication | |
| Equivalent Register Removal | |
| Register Balancing | No |
| Move First Flip-Flop Stage | N/A |
| Move Last Flip-Flop Stage | N/A |
| Slice Packing | |
| Pack I/O Registers into IOBs | No |
| | |

Click on the 'OK' button at the bottom of the window to apply the new settings.

Next the Translate Properties must be checked. With the file 'top.vhd' still highlighted in the Module View right click on 'Translate' in the Process View. Select 'Properties...' to open the following window. Tick the item 'Allow Unmatched LOC Constraints', check all other settings match, and click 'OK'.

| Property Name | Value |
|--|-----------|
| Use LOC Constraints | |
| Netlist Translation Type | Timestamp |
| Macro Search Path | |
| Create I/O Pads from Ports | |
| Allow Unexpanded Blocks | |
| User Rules File for Netlister Launcher | |
| Allow Unmatched LOC Constraints | ব |
| Preserve Hierarchy on Sub Module | |
| | |

Next open the Process Properties window for the Map process. Check that the settings match those in the picture below.

| Property Name | Value |
|--|-------------------|
| Trim Unconnected Signals | <u>v</u> |
| Replicate Logic to Allow Logic Level Reduction | V |
| Allow Logic Optimization Across Hierarchy | |
| Map to Input Functions | 4 |
| Optimization Strategy (Cover Mode) | Area |
| Generate Detailed MAP Report | |
| Use Guide Design File (.ncd) | |
| Guide Mode | None |
| Use RLOC Constraints | |
| Pack I/O Registers/Latches into IOBs | Default (For I/O) |
| Disable Register Ordering | |
| CLB Pack Factor Percentage | 100 |
| Tri-state Buffer Transformation Mode | Off |
| Perform Timing-Driven Packing | |
| Map Slice Logic into Unused Block RAMs | |

Next check the Place and Route process properties. For Example1 the 'Place & Route Effort Level (Overall)' typically needs to be set to 'Normal'. Make this change and check all other settings match the picture below and click 'OK'.

| Process Properties | × | |
|--|------------------------|--|
| Place & Route Properties | | |
| Property Name | Value | |
| Place & Route Effort Level (Overall) | Normal 🔽 | |
| Placer Effort Level (Overrides Overall Level) | N/A | |
| Router Effort Level (Overrides Overall Level) | N/A | |
| Extra Effort (Highest PAR level only) | N/A | |
| Starting Placer Cost Table (1-100) | 1 | |
| Place And Route Mode | Normal Place and Route | |
| Guide File | | |
| Guide Mode | N/A | |
| Use Timing Constraints | | |
| Use Bonded I/Os | | |
| Generate Detailed PAR Report | | |
| Generate Post-Place & Route Static Timing Report | V | |
| Generate Post-Place & Route Simulation Model | | |
| | | |
| OK Cancel | Default Help | |

Next check the 'Generate Programming File' properties. With the 'General Options' tab at the front, tick the 'Create ASCII Configuration File' and then check that all other settings match for this tab.

| Process Properties | × |
|---|---------------------|
| Startup options Readback options | Encryption options |
| General Options Cor | nfiguration options |
| Property Name | Value |
| Run Design Rules Checker (DRC) | <u>।</u> |
| Create Bit File | V |
| Create Binary Configuration File | |
| Create ASCII Configuration File | |
| Create IEEE 1532 Configuration File | |
| Enable BitStream Compression | |
| Enable Debugging of BitStream | |
| Enable Cyclic Redundancy Checking (CRC) | |
| | |
| OK Cancel | Default Help |

Bring the Configuration Options to the front. Select 'Pull Up' for the 'Unused IOB Pins' item. With this done check the settings match those shown in the picture below.

| Startup options Readback options | | | Encryption options | \$ | |
|---------------------------------------|-------------|---------------------|--------------------|---------|--|
| General Options Configuration options | | nfiguration options | | | |
| Property Name | | | Value | | |
| Configuration Rate | | | 4 | | |
| Configuration Clk (Configura | tion Pins) | | Pull Up | | |
| Configuration Pin M0 | | | Pull Up | | |
| Configuration Pin M1 | | | Pull Up | | |
| Configuration Pin M2 | | | Pull Up | Pull Up | |
| Configuration Pin Program | | | Pull Up | | |
| Configuration Pin Done | | Pull Up | | | |
| Configuration Pin Powerdown | | Pull Up | | | |
| JTAG Pin TCK | | Pull Up | | | |
| JTAG Pin TDI | | Pull Up | | | |
| JTAG Pin TDO Float | | | | | |
| JTAG Pin TMS Pull Up | | | | | |
| Unused IOB Pins Pull Up | | • | | | |
| UserID Code (8 Digit Hexadecimal) | | 0×FFFFFFFF | | | |
| Reset DCM if SHUTDOWN 8 | AGHIGH pe | erformed | | | |
| Disable Bandgap Generator | for DCMs to | o save power | | | |
| DCI Update Mode | | | Continuous | | |

Next check the settings for the 'Startup Options' tab against those shown in the picture below.

| Process Properties | | × |
|----------------------------|------------------|--------------------|
| General Options | Con | figuration options |
| Startup options | Readback options | Encryption options |
| Proper | ty Name | Value |
| FPGA Start-Up Clock | | CCLK 🔽 |
| Enable Internal Done Pipe | | |
| Done (Output Events) | | Default (4) |
| Enable Outputs (Output Eve | nts) | Default (5) |
| Release Write Enable (Outp | ut Events) | Default (6) |
| Release DLL (Output Events | 5) | Default (NoWait) |
| Match Cycle | | Default (NoVVait) |
| Drive Done Pin High | | |
| | | |
| 0 | K Cancel | Default Help |

Next check the 'Readback Options' against the picture below.

| Process Properties | | | × | | | |
|---------------------------------|------------------------------------|-------------|-----------------------------|----------------------------|-----|--|
| General Option | ns | Cor | nfiguration options | | | |
| Startup options | Readback options Encryption option | | | | | |
| Prop | erty Name | | Value | | | |
| Security | | | Enable Readback and 💌 | | | |
| Create ReadBack Data Fil | | | | Create ReadBack Data Files | | |
| Allow SelectMAP Pins to Persist | | Persist N/A | | | | |
| Create Logic Allocation Fi | ile N/A | | reate Logic Allocation File | | N/A | |
| Create Mask File | N/A | | N/A | | | |
| | | | | | | |
| | ОК | Cancel | Default Help | | | |

Finally check the 'Encryption Options' tab against the picture shown below.

| Process Properties | | | | ر | | |
|-------------------------------|-------------------------------------|------------------|-----------------|------------|--|--|
| General Option | General Options Confi | | | ons | | |
| Startup options | Readback options Encryption options | | | on options | | |
| Property Name | | | V | alue | | |
| Encrypt Bitstream | | | | | | |
| Key 0 (Hex String) | | | | | | |
| Key 1 (Hex String) | | | | | | |
| Key 2 (Hex String) | | | | | | |
| Key 3 (Hex String) | | | | | | |
| Key 4 (Hex String) | | | | | | |
| Key 5 (Hex String) | ey 5 (Hex String) | | | | | |
| Input Encryption Key File | | | | | | |
| Location of Key 0 in Sequ | Location of Key 0 in Sequence None | | | | | |
| Location of Key 1 in Sequ | | | | | | |
| Location of Key 2 in Sequence | | in Sequence None | | | | |
| Location of Key 3 in Sequence | | in Sequence None | | | | |
| Location of Key 4 in Sequ | ocation of Key 4 in Sequence None | | | | | |
| Location of Key 5 in Sequ | ocation of Key 5 in Sequence None | | | | | |
| Starting Key | arting Key None | | | | | |
| Starting CBC Value (Hex) | Starting CBC Value (Hex) | | | | | |
| | | | | | | |
| | ОК | Cancel | <u>D</u> efault | Help | | |

When all tabs have been checked, click 'OK' to enter the new settings.

Step 7: Building the Project

At this point the project is ready to be built. All of the required design files have been added to the project and the project build settings have been checked. Example1 should now build as far as bitstream generation without error.

Although each example project supplied on the HUNT ENGINEERING CD will differ from Example1, the process to create a new project for ISE 4 is the same. The routine described in this section can therefore be repeated for each of the standard examples provided on the CD. For each FPGA module type, each project contained on the CD provides all of the required design elements to ensure correct operation on that chosen module type. These elements include a Hardware Interface Layer that is used to correctly control external devices, user constraints information to control design timing and pin location and example VHDL to provide a structured starting point.

For users of ISE 5 design tools this document must be followed in order to convert the newer project format used on the HUNT ENGINEERING CD to the correct ISE 5 format. This section describes how to build an ISE 5 project from scratch, using the design source provided on the CD. This process is necessary as the standard XILINX tool version has moved on from ISE 5 and XILINX provide no mechanism for converting post ISE 5 projects back to ISE 5.

This document uses Example1 as the starting point for the creation of a new project. It is important to start from one of the standard examples provided on the HUNT ENGINEERING CD as this give the correct starting point for FPGA development with your FPGA module.

Please note: if you are using this section in order to create a brand new project with functionality that does not match any of the standard CD examples, then you must still start from Example1. Once you have created a correct project based around Example1 you may then insert your own unique code into the User-Ap entity, removing all unwanted logic. When doing this, you will need to refer to the relevant information in the 'Making your own FPGA design' section of your FPGA User Manual.

Converting Example1 for your Module Type

The HUNT ENGINEERING CD provides support for many different FPGA module types. For each module there is always a standard Example1 project provided on the CD. This example is the 'Getting Started' example for each module type and should always be used as the first step in FPGA development.

This section describes how to convert Example1 for the HERON-FPGA5 as an example of the conversion process. Although the design source varies for each module type, the principles shown here are the same regardless of type.

Step 1: Copying Example1 from the HUNT ENGINEERING CD

The first step in the conversion process is to copy Example1 onto your local hard drive. Make a directory on your local drive in which to store Example1.

On the HUNT ENGINEERING CD, all FPGA examples are provided below the 'fpga' directory. The 'fpga' directory is divided into sub-directories that reflect the name of the module type. In this document, Example1 for the HERON-FPGA5 is to be converted, so the directory 'fpga/fpga5v1/' will contain the appropriate project information. Identify the appropriate CD directory according to your module type.

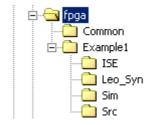
With the correct CD directory located you will need to copy Example 1 to your local drive. This can be done in one of two ways. The first method is to copy the Example 1 project by hand for the module type you are using. The second method is to unzip the ZIP file for that module type onto your hard drive.

When using the first approach you will need to copy the 'Common' directory and 'Example1' directory from the CD to a chosen directory on your local drive. The 'Example1' directory contains all design source that forms Example1 and the 'Common' directory provides design source common to all designs made for that module.

Please note, when copying files from your CD by hand you may need to edit the file attributes after copying. This is because on some operating systems Read-Only files on the CD will become Read-Only files on your local drive. All files in the 'Example1' directory should be set to Read-Write while all files in the 'Common' directory must remain Read-Only.

When using the second approach you will be unzipping all examples for that module, including Example1. When unzipping all project examples, the read-write attributes will automatically be set correctly.

The picture below shows a new directory on the local drive named 'fpga'. In the 'fpga' directory are the copied 'Common' and 'Example1' directories.



Step 2: Creating a New Project File

With the Example1 example directory and Common directory copied onto your Local Drive you can now begin building an ISE 5 project.

Open the ISE 5 Project Navigator if it is not already open. Please note, in the remainder of this section an example conversion is shown for Example 1 for the HERON-FPGA5 where this conversion is performed in ISE 5.2. If you are using a different ISE 5 version then although the windows shown may not perfectly match, the conversion process is still the same.

Next, delete the project file for the Example1 project in the new directory you have created on your local drive. The project file will be located in the 'ISE' sub-directory of the 'Example1' directory and will have the extension '.ise'. This existing project file must be removed as it will not be useable in ISE 5 and must be replaced with an appropriately constructed project file.

With the existing file deleted, select the menu item 'File \rightarrow New Project...' to begin creating a new project. The following window should be displayed.

| New Project | | | | × |
|-------------------------------------|---|----------|------|---|
| Project <u>N</u> ame: Ex1_Fpga5v | Project <u>L</u> ocation c:\fpga5v1\Ex | | | |
| Project Device <u>O</u> ptions: | | | | |
| Property Name | | Value | | |
| Device Family | | Virtex2 | | |
| Device | | xc2v1000 | | |
| Package | | fg456 | | |
| Speed Grade | | -4 | | |
| Design Flow | | XST VHDL | | |
| | OK | Cancel | Help | |

Enter the correct project name in the 'Project Name' field. For Example1 this should be set to be the same project name as the Example1 project file on the HUNT ENGINEERING CD (minus the .ise extension).

Next, enter the correct location into the 'Project Location' field. This field should match the location of the 'Example1/ISE' directory you have made on your local drive.

Then set the Project Device Options to correctly reflect the appropriate device information according to the module type you are using. If you are unsure of the correct information refer to the User Manual for that module type. Ensure that the Design Flow is 'XST VHDL'.

When you have correctly defined all fields click 'OK'.

Step 3: Adding VHDL Source to the New Project

The next step is to add design source, starting with the top level of the hierarchy. For all FPGA module projects the top level of the design is always contained in the file 'top.vhd'. 'top.vhd' is always provided as part of the 'Common' directory for each module type.

Select the menu item 'Project \rightarrow Add Source...'. The following window will be displayed where you will need to navigate to the 'Common' directory that was created on your local drive. Select the file 'top.vhd' and click 'Open'.

| Add Existing Sources | | | ? × |
|---|---|-------------------|-----|
| Look in: 🔂 Common | | - 🗢 🗈 💣 🎟- | |
| Fpga5v_tpl.ucf HE_CONV.vhd HE_RWCLK.vhd HE_SDRAM.vhd HE_USER.vhd HE_USER.vhd HE_WR_6F.vhd | SIM_MSG.vhd SIM_RD_6F.vhd SIM_SDRAM.vhd SIM_WR_6F.vhd SIM_WR_6F.vhd SYN_V2_RD_6F.vhd | JUSER_AP_TPL.vhd | |
| File name: TOP.vhd | | <u>0</u> pe | n |
| Files of type: Sources | (*.txt;*.vhd;*.vhdl;*.v;*.abl;* | f.xco;*.sc 💌 Cano | el |

The following window will then appear. Select 'VHDL Module' and click OK to add 'top.vhd' to the project.

| Choose Source Type | × |
|---|--------|
| TOP.vhd is which source type? The suffix is ambiguous as to type | |
| VHDL Module VHDL Package VHDL Test Bench | Cancel |
| | Help |

With this done, the Module View window will now look similar to the picture below, depending on the particular source files that are required by the module type you are using:

| Sources in Project: |
|----------------------------------|
| 🚎 🧧 Ex1_Fpga5v |
| Ė~ 🛄 xc2v1000-4fg456 |
| 🗄 🐨 📝 top-rtl (\\Common\TOP.vhd) |
| |
| 🚽 📝 he_rwclk |
| |
| 🚽 📝 he_user |
| 🚽 📝 he_wr_6f |
| user_ap |
| |
| |

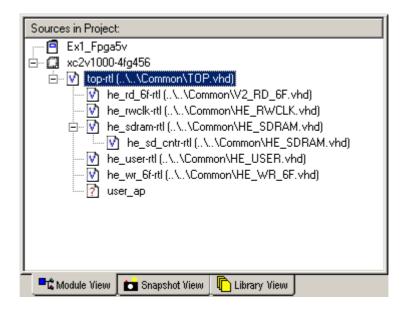
In the case of the FPGA5v1 example conversion, after adding 'top.vhd' we can now see 6 more design entities are needed. One of these entities is the User-Application level of the design which contains the VHDL that makes this example the getting started example, Example1. The other entities shown are part of the Hardware Interface Layer and are found in the 'Common' directory.

First add the appropriate files from the 'Common' directory in order to replace the red question mark icons with correct entities. To do this, you will again need to use the 'Add Source...' menu item. Navigate to the 'Common' directory and select the files that have names that match the entities in the Module View of your new project.

Please note: the HE_RD_6F entity is provided in the 'Common' directory in the source file 'V2_RD_6F'.

When adding each file, the correct Source Type will be 'VHDL Module' as was selected when adding 'top.vhd'.

When you have added all required Hardware Interface Layer components the Module View should look similar to the picture below for the FPGA5v1 conversion:



Next you will need to navigate to the 'Src' directory of Example1 and add the User_Ap file to the project:

| Add Existing | Sources | <u>?</u> × |
|-------------------------|--|------------|
| Look in: 🔁 | Src 💌 🗢 🖆 🎫 - | |
| HSB1.vhd | | |
| HSB1.vhd | | |
| TB_EX1.vh | | |
| | | |
| | | |
| | | |
| , File <u>n</u> ame: | User_Ap1.vhd | n |
| | | <u> </u> |
| Files of <u>type</u> : | Sources (*.txt)*.vhd)*.vhd)*.v;*.abl;*.xco;*.sc 💌 🛛 Cano | :el |

After the 'user_ap' entity has been added to the project you will see which other source files are required below this entity. Add the appropriate source files from the 'Src' directory.

After you have added all of required VHDL source files from the 'Src' directory there may still be red question marks against entities in the hierarchy. These entities will correspond to Core Gen components that are placed in the Example1 'ISE' directory. In the case of Example1 there is one Core Gen component called 'fifo15x32' that must be added to the project from the 'ISE' directory.

With all relevant VHDL source added to the project the Module View should look similar to the picture shown below for the FPGA5v1 conversion:

| Sources in Project: | | |
|--|--|--|
| 🚎 🖻 Ex1_Fpga5v | | |
| Ė.∽ 🛄 xc2v1000-4fg456 | | |
| 🖻 _ config (\Src\User_Ap1.vhd) | | |
| 🖻 📝 [top-rtl (\\Common\TOP.vhd)] | | |
| | | |
| 📝 he_rwclk-rtl (\\Common\HE_RWCLK.vhd) | | |
| 🖨 - 📝 he_sdram-rtl (\\Common\HE_SDRAM.vhd) | | |
| he_sd_cntr-rtl (\\Common\HE_SDRAM.vhd) | | |
| he_user-rtl (\\Common\HE_USER.vhd) | | |
| he_wr_6f-rtl (\\Common\HE_WR_6F.vhd) | | |
| 🖻 🖓 user_ap-example1 (\Src\User_Ap1.vhd) | | |
| | | |
| hsb1-rtl (\Src\HSB1.vhd) | | |
| | | |
| 📃 📲 Module View 🚺 💼 Snapshot View 🖺 Library View | | |

Step 4: Adding User Constraints

The next step is to add user design constraints to the project. Using 'Project \rightarrow Add Source...' add the user constraints file contained in the 'ISE' directory of the project.

| Add Existing Sources | <u>? ×</u> |
|---|------------|
| Look in: 🔁 ISE 💌 🖛 🗈 📸 🏢 | - |
| | |
| Ex1_Fpga5v.ucf fifo15x32.vhd | |
| ■ fifo15x32.xco | |
| | |
| | |
| | |
| File name: Ex1_Fpga5v.ucf | ben |
| Files of type: Sources (*.txt;*.vhd;*.xco;*.sch;*.tbw;*.bmm;* | ncel |

Select the UCF file and click 'Open'. The following window will appear to allow you to specify the design file to which the user constraints should be associated. Highlight 'top' and click 'OK'.

| Associate Ex1_Fpga5v.ucf with affects. | the source tha |
|---|----------------|
| he_sd_ontr | ОК |
| top he_rd_6f | Cancel |
| he_rwclk he_wr_6f user_ap | Help |

Step 5: Adding a Simulation Test-Bench

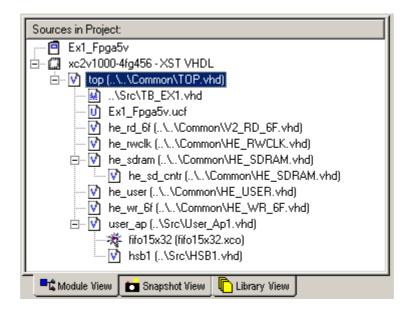
Typically, a project will also include a test-bench for simulation. For Example1 there is a test-bench provided in the 'Src' directory of the project. This file must also be added to the project using 'Project \rightarrow Add Source...'. Navigate to the 'Src' directory, highlight the file that beings 'TB_' and click 'Open'.

| Add Existing | Sources | ? × |
|------------------------|--|----------|
| Look in: 🔁 |) Src 💽 🗢 🛍 🖽 | - |
| HSB1.vhd | | |
| sim_fifo.vl | nd | |
| | | |
| File <u>n</u> ame: | TB_EX1.vhd | pen |
| Files of <u>type</u> : | Sources (*.txt;*.vhd;*.vhd;*.v;*.abl;*.xco;*.sc 💌 Ca | incel |

The following window will appear where you need to specify the source type. Select 'VHDL Test Bench' and click 'OK'.

| Choose Source Type | × |
|---|--------|
| TB_EX1.vhd is which source type? The suffix is ambiguous as to type. | ? |
| VHDL Module VHDL Package | ОК |
| VHDL Test Bench | Cancel |
| | Help |
| | |

At this point the Module View should look similar to the picture shown below for the FPGA5v1 conversion process.



Step 6: Setting Project Build Options

The last step in creating a new project is to apply the necessary project settings that will allow the design to be built correctly.

Ensure the file 'top.vhd' is highlighted in the Module View, and then right click on 'Synthesize' in the Process View. Select 'Properties...' to open the following window. With the 'Synthesis Options' tab at the front, check that the settings match those shown below.

| Property Name | Value |
|--------------------------------|--------------|
| Optimization Goal | Speed 💌 |
| Optimization Effort | Normal |
| Synthesis Constraints File | |
| Use Synthesis Constraints File | |
| Keep Hierarchy | |
| Global Optimization Goal | AllClockNets |
| Generate RTL Schematic | Yes |
| Read Cores | |
| Write Timing Constraints | |
| Cross Clock Analysis | |
| Hierarchy Separator | _ |
| Bus Delimiter | |
| Slice Utilization Ratio | 100 |
| Case | Lower |
| VHDL Work Directory | ./xst |
| VHDL INI File | |

Next bring the 'HDL Options' tab to the front and check that the settings match those in the picture below.

| Property Name | Value |
|---------------------------------|----------|
| FSM Encoding Algorithm | Auto 🔽 |
| RAM Extraction | N |
| RAM Style | Auto |
| ROM Extraction | |
| ROM Style | Auto |
| Mux Extraction | Yes |
| Mux Style | Auto |
| Decoder Extraction | |
| Priority Encoder Extraction | Yes |
| Shift Register Extraction | |
| Logical Shifter Extraction | |
| XOR Collapsing | |
| Resource Sharing | |
| Complex Clock Enable Extraction | |
| Multiplier Style | Auto |

Next bring the 'Xilinx Specific Options' tab to the front. Change the 'Number of Clock Buffers' item to '0', and set the 'Pack I/O Registers into IOBs' item to 'No'. With this done, check that the settings match those in the picture below.

| Synthesis Options HDL Options Xilinx Specific Optio | |
|---|-------|
| Property Name | Value |
| Add I/O Buffers | |
| Max Fanout | 500 |
| Number of Clock Buffers | 0 |
| Register Duplication | |
| Equivalent Register Removal | |
| Register Balancing | No |
| Move First Flip-Flop Stage | N/A |
| Move Last Flip-Flop Stage | N/A |
| Slice Packing | |
| Pack I/O Registers into IOBs | No |
| | |

Click on the 'OK' button at the bottom of the window to apply the new settings.

Next the Translate Properties must be checked. With the file 'top.vhd' still highlighted in the Module View right click on 'Translate' in the Process View. Select 'Properties...' to open the following window. Tick the item 'Allow Unmatched LOC Constraints', check all other settings match, and click 'OK'.

| Property Name | Value |
|--|-----------|
| Use LOC Constraints | |
| Netlist Translation Type | Timestamp |
| Macro Search Path | |
| Create I/O Pads from Ports | |
| Allow Unexpanded Blocks | |
| User Rules File for Netlister Launcher | |
| Allow Unmatched LOC Constraints | |
| Preserve Hierarchy on Sub Module | |
| | |

Next open the Process Properties window for the Map process. Check that the settings match those in the picture below.

| Property Name | Value |
|--|-------------------|
| Trim Unconnected Signals | <u>v</u> |
| Replicate Logic to Allow Logic Level Reduction | <u>ح</u> |
| Allow Logic Optimization Across Hierarchy | |
| Map to Input Functions | 4 |
| Optimization Strategy (Cover Mode) | Area |
| Generate Detailed MAP Report | |
| Use Guide Design File (.ncd) | |
| Guide Mode | None |
| Use RLOC Constraints | |
| Pack I/O Registers/Latches into IOBs | Default (For I/O) |
| Disable Register Ordering | |
| CLB Pack Factor Percentage | 100 |
| Tri-state Buffer Transformation Mode | Off |
| Perform Timing-Driven Packing | |
| Map Slice Logic into Unused Block RAMs | |

Next check the Place and Route process properties. For Example1 the 'Place & Route Effort Level (Overall)' typically needs to be set to 'Normal'. Make this change and check all other settings match the picture below and click 'OK'.

| Process Properties | X | |
|--|------------------------|--|
| Place & Route Properties | | |
| Property Name | Value | |
| Place & Route Effort Level (Overall) | Normal 🔽 | |
| Placer Effort Level (Overrides Overall Level) | N/A | |
| Router Effort Level (Overrides Overall Level) | N/A | |
| Extra Effort (Highest PAR level only) | N/A | |
| Starting Placer Cost Table (1-100) | 1 | |
| Place And Route Mode | Normal Place and Route | |
| Guide File | | |
| Guide Mode | N/A | |
| Use Timing Constraints | | |
| Use Bonded I/Os | | |
| Generate Detailed PAR Report | | |
| Generate Post-Place & Route Static Timing Report | | |
| Generate Post-Place & Route Simulation Model | | |
| | | |
| OK Cancel | <u>D</u> efault Help | |

Next check the 'Generate Programming File' properties. With the 'General Options' tab at the front, tick the 'Create ASCII Configuration File' and then check that all other settings match for this tab.

| rocess Properties | × | |
|---|---------------------|--|
| Startup options Readback options | Encryption options | |
| General Options Cor | nfiguration options | |
| Property Name Value | | |
| Run Design Rules Checker (DRC) | <u>v</u> | |
| Create Bit File | | |
| Create Binary Configuration File | | |
| Create ASCII Configuration File | | |
| Create IEEE 1532 Configuration File | | |
| Enable BitStream Compression | | |
| Enable Debugging of BitStream | | |
| Enable Cyclic Redundancy Checking (CRC) | | |
| | | |
| OK Cancel | Default Help | |

Bring the Configuration Options to the front. Select 'Pull Up' for the 'Unused IOB Pins' item. With this done check the settings match those shown in the picture below.

| Startup options | Readba | ck options | Encryption options | | |
|--|------------|------------|--------------------|---------|--|
| General Options Conf | | | figuration options | | |
| Property Name | | | Value | | |
| Configuration Rate | | | 4 | | |
| Configuration Clk (Configura | tion Pins) | | Pull Up | | |
| Configuration Pin M0 | | | Pull Up | | |
| Configuration Pin M1 | | | Pull Up | | |
| Configuration Pin M2 | | Pull Up | Pull Up | | |
| Configuration Pin Program | | | Pull Up | Pull Up | |
| Configuration Pin Done | | Pull Up | | | |
| Configuration Pin Powerdown | | Pull Up | | | |
| JTAG Pin TCK | | Pull Up | | | |
| JTAG Pin TDI | | Pull Up | | | |
| JTAG Pin TDO | | Float | | | |
| JTAG Pin TMS | | Pull Up | | | |
| Unused IOB Pins | | Pull Up 💌 | | | |
| UserID Code (8 Digit Hexadecimal) | | | 0×FFFFFFFF | | |
| Reset DCM if SHUTDOWN & AGHIGH performed | | | | | |
| Disable Bandgap Generator for DCMs to save power | | | | | |
| DCI Update Mode | | Continuous | | | |

Next check the settings for the 'Startup Options' tab against those shown in the picture below.

| Process Properties | | × | |
|-----------------------------|----------------------------------|---------------------|--|
| General Options | Cor | nfiguration options | |
| Startup options | Startup options Readback options | | |
| Proper | ty Name | Value | |
| FPGA Start-Up Clock | | CCLK | |
| Enable Internal Done Pipe | | | |
| Done (Output Events) | Default (4) | | |
| Enable Outputs (Output Eve | ents) Default (5) | | |
| Release Write Enable (Outp | put Events) Default (6) | | |
| Release DLL (Output Events) | | Default (NoWait) | |
| Match Cycle | Default (NoWait) | | |
| Drive Done Pin High | | | |
| | | | |
| 0 | K Cancel | Default Help | |

Next check the 'Readback Options' against the picture below.

| Process Properties | | | × | | | |
|-------------------------------|---------------------|-----------|---------------------|----------------------------|--|--|
| General Options Configuration | | | nfiguration options | | | |
| Startup options | Readbac | k options | Encryption options | | | |
| Prop | erty Name | | Value | | | |
| Security | Enable Readback and | | | | | |
| Create ReadBack Data Fil | Files | | | Create ReadBack Data Files | | |
| Allow SelectMAP Pins to | Persist N/A | | N/A | | | |
| Create Logic Allocation Fi | ile N/A | | | | | |
| Create Mask File | N/A | | N/A | | | |
| | | | | | | |
| | ОК | Cancel | Default Help | | | |

Finally check the 'Encryption Options' tab against the picture shown below.

| Process Properties | | | | 2 | | |
|------------------------------------|------------------------|--------|------------------|--------------------|--|--|
| General Options Con | | | nfiguration opti | figuration options | | |
| Startup options | Readback options Encry | | | on options | | |
| Property Name | | | V | alue | | |
| Encrypt Bitstream | | | | | | |
| Key 0 (Hex String) | | | | | | |
| Key 1 (Hex String) | | | | | | |
| Key 2 (Hex String) | | | | | | |
| Key 3 (Hex String) | | | | | | |
| Key 4 (Hex String) | | | | | | |
| Key 5 (Hex String) | | | | | | |
| Input Encryption Key File | | | | | | |
| Location of Key 0 in Sequence | | None | | | | |
| Location of Key 1 in Sequence N | | None | None | | | |
| Location of Key 2 in Sequence None | | | | | | |
| Location of Key 3 in Sequence None | | | | | | |
| Location of Key 4 in Sequence None | | | | | | |
| Location of Key 5 in Sequence None | | | | | | |
| Starting Key | tarting Key None | | | | | |
| Starting CBC Value (Hex) | | | | | | |
| | | | | | | |
| | ОК | Cancel | <u>D</u> efault | Help | | |

When all tabs have been checked, click 'OK' to enter the new settings.

Step 7: Building the Project

At this point the project is ready to be built. All of the required design files have been added to the project and the project build settings have been checked. Example1 should now build as far as bitstream generation without error.

Although each example project supplied on the HUNT ENGINEERING CD will differ from Example1, the process to create a new project for ISE 5 is the same. The routine described in this section can therefore be repeated for each of the standard examples provided on the CD. For each FPGA module type, each project contained on the CD provides all of the required design elements to ensure correct operation on that chosen module type. These elements include a Hardware Interface Layer that is used to correctly control external devices, user constraints information to control design timing and pin location and example VHDL to provide a structured starting point.

For users of ISE 6 design tools this document must be followed in order to convert the newer project format used on the HUNT ENGINEERING CD to the correct ISE 6 format. This section describes how to build an ISE 6 project from scratch, using the design source provided on the CD. This process is necessary as the standard XILINX tool version has moved on from ISE 6 and XILINX provide no mechanism for converting post ISE 6 projects back to ISE 6.

This document uses Example1 as the starting point for the creation of a new project. It is important to start from one of the standard examples provided on the HUNT ENGINEERING CD as this give the correct starting point for FPGA development with your FPGA module.

Please note: if you are using this section in order to create a brand new project with functionality that does not match any of the standard CD examples, then you must still start from Example1. Once you have created a correct project based around Example1 you may then insert your own unique code into the User-Ap entity, removing all unwanted logic. When doing this, you will need to refer to the relevant information in the 'Making your own FPGA design' section of your FPGA User Manual.

Converting Example1 for your Module Type

The HUNT ENGINEERING CD provides support for many different FPGA module types. For each module there is always a standard Example1 project provided on the CD. This example is the 'Getting Started' example for each module type and should always be used as the first step in FPGA development.

This section describes how to convert Example1 for the HERON-FPGA5 as an example of the conversion process. Although the design source varies for each module type, the principles shown here are the same regardless of type.

Step 1: Copying Example1 from the HUNT ENGINEERING CD

The first step in the conversion process is to copy Example1 onto your local hard drive. Make a directory on your local drive in which to store Example1.

On the HUNT ENGINEERING CD, all FPGA examples are provided below the 'fpga' directory. The 'fpga' directory is divided into sub-directories that reflect the name of the module type. In this document, Example1 for the HERON-FPGA5 is to be converted, so the directory 'fpga/fpga5v1/' will contain the appropriate project information. Identify the appropriate CD directory according to your module type.

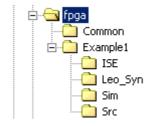
With the correct CD directory located you will need to copy Example 1 to your local drive. This can be done in one of two ways. The first method is to copy the Example 1 project by hand for the module type you are using. The second method is to unzip the ZIP file for that module type onto your hard drive.

When using the first approach you will need to copy the 'Common' directory and 'Example1' directory from the CD to a chosen directory on your local drive. The 'Example1' directory contains all design source that forms Example1 and the 'Common' directory provides design source common to all designs made for that module.

Please note, when copying files from your CD by hand you may need to edit the file attributes after copying. This is because on some operating systems Read-Only files on the CD will become Read-Only files on your local drive. All files in the 'Example1' directory should be set to Read-Write while all files in the 'Common' directory must remain Read-Only.

When using the second approach you will be unzipping all examples for that module, including Example1. When unzipping all project examples, the read-write attributes will automatically be set correctly.

The picture below shows a new directory on the local drive named 'fpga'. In the 'fpga' directory are the copied 'Common' and 'Example1' directories.



Step 2: Creating a New Project File

With the Example1 example directory and Common directory copied onto your Local Drive you can now begin building an ISE 6 project.

Open the ISE 6 Project Navigator if it is not already open. Please note, in the remainder of this section an example conversion is shown for Example 1 for the HERON-FPGA5 where this conversion is performed in ISE 6.2. If you are using a different ISE 6 version then although the windows shown may not perfectly match, the conversion process is still the same.

Next, delete the project file for the Example1 project in the new directory you have created on your local drive. The project file will be located in the 'ISE' sub-directory of the 'Example1' directory and will have the extension '.ise'. This existing project file must be removed as it will not be useable in ISE 6 and must be replaced with an appropriately constructed project file.

With the existing file deleted, select the menu item 'File \rightarrow New Project...' to begin creating a new project. The following window should be displayed.

| Project <u>N</u> ame: Ex1_Fpga5v | Project Location: c:\fpga5v1\Example1\ISE | |
|-------------------------------------|---|--|
| | | |
| | | |
| | the second se | |
| | Level module for the Project | |
| Top-Level Module Ty | | |
| Top-Level Module Ty | pe: | |
| | pe: | |
| Top-Level Module Ty | pe: | |

Enter the correct project name in the 'Project Name' field. For Example1 this should be set to be the same project name as the Example1 project file on the HUNT ENGINEERING CD (minus the .ise extension).

Next, enter the correct location into the 'Project Location' field. This field should match the location of the 'Example1/ISE' directory you have made on your local drive. Next, ensure the 'Top-Level Module Type' field is set to HDL.

With these steps completed click on the 'Next >' button at the bottom of the window, to display the following:

| Device Fam | ily | Virtex2 |
|-------------|---------------------|--------------------|
| Device | | xc2v1000 |
| Package | | fg456 |
| Speed Grad | e | -4 |
| Top-Level N | 1odule Type | HDL |
| Synthesis T | ool | XST (VHDL/Verilog) |
| Simulator | | Other |
| Generated S | Simulation Language | VHDL |
| | | |

Fill out the appropriate device information according to the module type you are using. If you are unsure of the correct information refer to the User Manual for that module type. When you have done this click the 'Next >' button.

The following window will then be displayed. Simply click 'Next >'.

| ew Proje | ct | | | | | |
|----------------|-------------------------|------------------------|-------------------|------------|----------------------|---------|
| Creat | e a New Source | | | | | |
| 1 | Source File | 1 | уре | | New Source Remove | |
| Creat | e a new source to add t | o the project (optiona |) Only one new : | | be specified | now |
| Additi comm | onal new sources can b | e added after project | creation using th | e "Project | ->New Source | ," " |
| | | | | | | |
| | | < <u>B</u> ack | <u>N</u> ext > | Cance | | Help |

The following window will then be displayed. Again, click 'Next >'.

| | Source File | Туре | Copy to Projec | Add Source |
|--------------------|--|---|---|-------------------------------------|
| 1 2 | | | | |
| 3 | | | | Remove |
| 4 | | | | • |
| | | | | |
| | | | | |
| d exis | ting sources to the pr | oject (optional). Addit | ional sources can be ac | lded after project |
| ld exis eation | ting sources to the pro using the "Project->4 | oject (optional). Addit Add Source'' or ''Proj | ional sources can be ac ect->Add Copy of Sourc | lded after project e'' commands. |
| ld exis reation | ting sources to the pro using the "Project->4 | oject (optional). Addit Add Source'' or ''Proj | ional sources can be ac ect->Add Copy of Sourc | lded after project e'' commands. |

Finally, the following window is displayed providing a summary of all the project information you have entered. Click 'Finish' to make the new project file.

| N | ew Project Information | × |
|---|---|---|
| | Project Navigator will create a new Project with the following specifications: | |
| | Project: Project Name: Ex1_Fpga5v Project Location: c:\fpga5v1\Example1\ISE Project Type: HDL Device: Device: Device Family: Virtex2 Device: xc2v1000 Package: fg456 Speed Grade: -4 | |
| | : Top-Level Module Type: HDL Synthesis Tool: XST (VHDL/Verilog) Simulator: Other Generated Simulation Language: VHDL | |
| | < <u>B</u> ack Finish Cancel Help | _ |

Step 3: Adding VHDL Source to the New Project

The next step is to add design source, starting with the top level of the hierarchy. For all FPGA module projects the top level of the design is always contained in the file 'top.vhd'. 'top.vhd' is always provided as part of the 'Common' directory for each module type.

Select the menu item 'Project \rightarrow Add Source...'. The following window will be displayed where you will need to navigate to the 'Common' directory that was created on your local drive. Select the file 'top.vhd' and click 'Open'.

| Add Existing Sources | | | ? × |
|---|---|--------------------|-----|
| Look in: 🔂 Common | | - 🗢 🗈 📥 | |
| Fpga5v_tpl.ucf HE_CONV.vhd HE_RWCLK.vhd HE_SDRAM.vhd HE_USER.vhd HE_USER.vhd HE_WR_6F.vhd | SIM_MSG.vhd SIM_RD_6F.vhd SIM_SDRAM.vhd SIM_WR_6F.vhd SIM_WR_6F.vhd SYN_V2_RD_6F.vhd | USER_AP_TPL.vhd | |
| File name: TOP.vhd | | <u> </u> | en |
| Files of type: Sources | (*.txt;*.vhd;*.vhdl;*.v;*.abl; | *.xco;*.sc 💌 🛛 Car | |

The following window will then appear. Select 'VHDL Design File' and click OK to add 'top.vhd' to the project.

| Choose Source Type | × |
|---|--------|
| TOP.vhd is which source type? The suffix is ambiguous as to type | |
| VHDL Design File VHDL Test Bench File | OK |
| | Cancel |
| | Help |
| | |

With this done, the Module View window will now look similar to the picture below, depending on the particular source files that are required by the module type you are using:

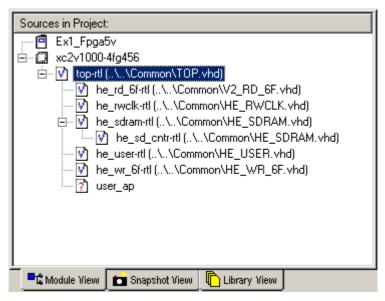
| Sources in Project: | | | |
|--------------------------------|--|--|--|
| 🚎 🧧 Ex1_Fpga5v | | | |
| 🖮 🛄 xc2v1000-4fg456 | | | |
| 🖮 📝 top-rtl (\\Common\TOP.vhd) | | | |
| | | | |
| 🚽 📝 he_rwclk | | | |
| | | | |
| 🚽 📝 he_user | | | |
| 🚽 📝 he_wr_6f | | | |
| user_ap | | | |
| | | | |
| | | | |

In the case of the FPGA5v1 example conversion, after adding 'top.vhd' we can now see 6 more design entities are needed. One of these entities is the User-Application level of the design which contains the VHDL that makes this example the getting started example, Example1. The other entities shown are part of the Hardware Interface Layer and are found in the 'Common' directory.

First add the appropriate files from the 'Common' directory in order to replace the red question mark icons with correct entities. To do this, you will again need to use the 'Add Source...' menu item. Navigate to the 'Common' directory and select the files that have names that match the entities in the Module View of your new project.

Please note: the HE_RD_6F entity is provided in the 'Common' directory in the source file 'V2_RD_6F'.

When adding each file, the correct Source Type will be 'VHDL Design File' as was selected when adding 'top.vhd'. When you have added all required Hardware Interface Layer components the Module View should look similar to the picture below for the FPGA5v1 conversion:



Next you will need to navigate to the 'Src' directory of Example1 and add the User_Ap file to the project:

| Add Existing | Sources | <u>?</u> × |
|-------------------------|--|------------|
| Look in: 🔁 | Src 💌 🗢 🖆 🎫 - | |
| HSB1.vhd | | |
| HSB1.vhd | | |
| TB_EX1.vh | | |
| | | |
| | | |
| | | |
| , File <u>n</u> ame: | User_Ap1.vhd | n |
| | | <u> </u> |
| Files of <u>type</u> : | Sources (*.txt)*.vhd)*.vhd)*.v;*.abl;*.xco;*.sc 💌 🛛 Cano | :el |

After the 'user_ap' entity has been added to the project you will see which other source files are required below this entity. Add the appropriate source files from the 'Src' directory.

After you have added all of required VHDL source files from the 'Src' directory there may still be red question marks against entities in the hierarchy. These entities will correspond to Core Gen components that are placed in the Example1 'ISE' directory. In the case of Example1 there is one Core Gen component called 'fifo15x32' that must be added to the project from the 'ISE' directory.

With all relevant VHDL source added to the project the Module View should look similar to the picture shown below for the FPGA5v1 conversion:

| Sources in Project: | | |
|--|--|--|
| 🚎 🖻 Ex1_Fpga5v | | |
| È 🛄 xc2∨1000-4fg456 | | |
| | | |
| 🖻 📝 [top-rtl (\\Common\TOP.vhd)] | | |
| 📝 he_rd_6f-rtl (\\Common\V2_RD_6F.vhd) | | |
| | | |
| 🖨 – 📝 he_sdram-rtl (\\Common\HE_SDRAM.vhd) | | |
| he_sd_cntr-rtl (\\Common\HE_SDRAM.vhd) | | |
| 💟 he_user-rtl (\\Common\HE_USER.vhd) | | |
| 🛐 _he_wr_6f-rtl (\\Common\HE_WR_6F.vhd) | | |
| i⊟ 💟 user_ap-example1 (\Src\User_Ap1.vhd) | | |
| - 🏄 fifo15x32 (fifo15x32.xco) | | |
| 🔤 📝 hsb1-rtl (\Src\HSB1.vhd) | | |
| | | |
| 📃 📲 Module View 🚺 💼 Snapshot View 🖺 Library View | | |

Step 4: Adding User Constraints

The next step is to add user design constraints to the project. Using 'Project \rightarrow Add Source...' add the user constraints file contained in the 'ISE' directory of the project.

| Add Existing 9 | Sources | <u>?</u> × |
|------------------------|--|------------|
| Look jn: 🔂 | ISE 💽 🗲 🖆 🎫 - | |
| Ex1_Fpga5 | 5v | |
| Ex1_Fpga5 | | |
| 📑 fifo15x32.0 | | |
| if o15x32. | | |
| 🛃 fifo15x32.: | xco | |
| | | |
| | | |
| | | |
| File <u>n</u> ame: | Ex1_Fpga5v.ucf | n |
| Files of <u>type</u> : | Sources (*.txt)*.vhd)*.vhd)*.v;*.abl;*.xco;*.sc 💌 Cano | el |
| , nee or gype. | | |

Select the UCF file and click 'Open'. The following window will appear to allow you to specify the design file to which the user constraints should be associated. Highlight 'top' and click 'OK'.

| Associate with Source | × |
|--|----------------------|
| Associate Ex1_Fpga5v.ucf with I affects. | he source that it |
| top he_sdram he_sd_cntr he_user he_rd_6f he_wr_6f | OK Cancel Help |

Step 5: Adding a Simulation Test-Bench

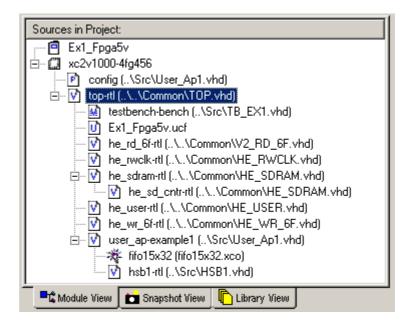
Typically, a project will also include a test-bench for simulation. For Example1 there is a test-bench provided in the 'Src' directory of the project. This file must also be added to the project using 'Project \rightarrow Add Source...'. Navigate to the 'Src' directory, highlight the file that beings 'TB_' and click 'Open'.

| Add Existing | Sources | <u>?</u> × |
|-------------------------|--|------------|
| Look jn: 🔁 |) Src 💌 🗢 🖻 📸 🎟 - | |
| HSB1.vhd | | |
| 📕 sim_fifo.vl | | |
| TB_EX1.vt | | |
| | | |
| | | |
| | | |
| , File <u>n</u> ame: | TB_EX1.vhd Ope | |
| r lie <u>H</u> ame. | | :r1 |
| Files of type: | Sources (*.txt;*.vhd;*.vhd;*.v;*.abl;*.xco;*.sc 🔽 🛛 Cano | el |
| | | 11. |

The following window will appear where you need to specify the source type. Select 'VHDL Test Bench File' and click 'OK'.

| Choose Source Type | × |
|--|--------|
| TB_EX1.vhd is which source type The suffix is ambiguous as to type. | |
| VHDL Design File VHDL Test Bench File | OK |
| | Cancel |
| | Help |
| | |

At this point the Module View should look similar to the picture shown below for the FPGA5v1 conversion process.



Step 6: Setting Project Build Options

The last step in creating a new project is to apply the necessary project settings that will allow the design to be built correctly.

Ensure the file 'top.vhd' is highlighted in the Module View, and then right click on 'Synthesize' in the Process View. Select 'Properties...' to open the following window. With the 'Synthesis Options' tab at the front, check that the settings match those shown below.

| Process Properties | | |
|---|---------------------|--|
| Synthesis Options HDL Options Xilinx Specific Options | | |
| Property Name | Value | |
| Optimization Goal | Speed | |
| Optimization Effort | Normal | |
| Synthesis Constraints File | | |
| Use Synthesis Constraints File | | |
| Library Search Order | | |
| Keep Hierarchy | No | |
| Global Optimization Goal | AllClockNets | |
| Generate RTL Schematic | Yes | |
| Read Cores | | |
| Cores Search Directories | | |
| Write Timing Constraints | | |
| Cross Clock Analysis | | |
| Hierarchy Separator | _ | |
| Bus Delimiter | \diamond | |
| Slice Utilization Ratio | 100 | |
| Case | Maintain | |
| Work Directory | ./xst | |
| HDL INI File | | |
| Verilog 2001 | | |
| Verilog Include Directories | | |
| Custom Compile File List | | |
| Other XST Command Line Options | | |
| | | |
| ОК | Cancel Default Help | |

Next bring the 'HDL Options' tab to the front and check that the settings match those in the picture below.

| rocess Properties | | | | |
|--|------------|--|--|--|
| Synthesis Options HDL Options Xilinx Specifi | ic Options | | | |
| Property Name | Value | | | |
| FSM Encoding Algorithm | Auto | | | |
| Case Implementation Style | None | | | |
| FSM Style | LUT | | | |
| RAM Extraction | N | | | |
| RAM Style | Auto | | | |
| ROM Extraction | <u> </u> | | | |
| ROM Style | Auto | | | |
| Mux Extraction | Yes | | | |
| Mux Style | Auto | | | |
| Decoder Extraction | | | | |
| Priority Encoder Extraction | Yes | | | |
| Shift Register Extraction | | | | |
| Logical Shifter Extraction | | | | |
| XOR Collapsing | | | | |
| Resource Sharing | <u> </u> | | | |
| Multiplier Style | Auto | | | |

Next bring the 'Xilinx Specific Options' tab to the front. Change the 'Number of Clock Buffers' item to '0', and set the 'Pack I/O Registers into IOBs' item to 'No'. With this done, check that the settings match those in the picture below.

| Synthesis Options HDL Options Xilinx Specific Options | |
|---|-------|
| Property Name | Value |
| Add I/O Buffers | |
| Max Fanout | 500 |
| Number of Clock Buffers | 0 |
| Register Duplication | |
| Equivalent Register Removal | |
| Register Balancing | No |
| Move First Flip-Flop Stage | N/A |
| Move Last Flip-Flop Stage | N/A |
| Pack I/O Registers into IOBs | No |
| Slice Packing | |
| Convert Tristates To Logic | No |
| Optimize Instantiated Primitives | |

Click on the 'OK' button at the bottom of the window to apply the new settings.

Next the Translate Properties must be checked. With the file 'top.vhd' still highlighted in the Module View right click on 'Translate' in the Process View. Select 'Properties...' to open the following window. Tick the item 'Allow Unmatched LOC Constraints', check all other settings match, and click 'OK'.

| P | rocess Properties | × |
|---|--|--------------|
| | Translate Properties | |
| | Property Name | Value |
| | Use LOC Constraints | |
| | Netlist Translation Type | Timestamp |
| | Macro Search Path | |
| | Create I/O Pads from Ports | |
| | Allow Unexpanded Blocks | |
| | User Rules File for Netlister Launcher | |
| | Allow Unmatched LOC Constraints | |
| | Preserve Hierarchy on Sub Module | |
| | Other Ngdbuild Command Line Options | |
| | | |
| | | |
| | | |
| | OK Cancel | Default Help |

Next open the Process Properties window for the Map process. Check that the settings match those in the picture below.

| Property Name | Value |
|--|------------------------|
| Perform Timing-Driven Packing and Placement | |
| Map Effort Level | N/A |
| Trim Unconnected Signals | |
| Replicate Logic to Allow Logic Level Reduction | |
| Allow Logic Optimization Across Hierarchy | |
| Map to Input Functions | 4 |
| Optimization Strategy (Cover Mode) | Area |
| Generate Detailed MAP Report | |
| MAP Guide Design File (.ncd) | |
| MAP Guide Mode | None |
| Convert Guide File to 6.1 i Format | |
| Use RLOC Constraints | |
| Pack I/O Registers/Latches into IOBs | For Inputs and Outputs |
| Disable Register Ordering | |
| CLB Pack Factor Percentage | 100 |
| Tri-state Buffer Transformation Mode | Off |
| Map Slice Logic into Unused Block RAMs | |
| Other Map Command Line Options | |

Next check the Place and Route process properties. For Example1 the 'Place & Route Effort Level (Overall)' typically needs to be set to 'Medium'. Make this change and check all other settings match the picture below and click 'OK'.

| Property Name Value | | |
|--|------------------------|--|
| Place & Route Effort Level (Overall) | Medium 🔽 | |
| Placer Effort Level (Overrides Overall Level) | None | |
| Router Effort Level (Overrides Overall Level) | None | |
| Extra Effort (Highest PAR level only) | N/A | |
| Starting Placer Cost Table (1-100) | 1 | |
| Place And Route Mode | Normal Place and Route | |
| PAR Guide Design File (.ncd) | | |
| PAR Guide Mode | None | |
| Convert Guide File to 6.1 i Format | | |
| Use Timing Constraints | | |
| Use Bonded I/Os | | |
| Generate Asynchronous Delay Report | | |
| Generate Post-Place & Route Static Timing Report | | |
| Generate Post-Place & Route Simulation Model | | |
| Other Place & Route Command Line Options | | |

Next check the 'Generate Programming File' properties. With the 'General Options' tab at the front, tick the 'Create ASCII Configuration File' and then check that all other settings match for this tab.

| Process Properties | × | | |
|---|--------------------|--|--|
| Startup Options Readback Options | Encryption Options | | |
| General Options Con | figuration Options | | |
| Property Name | Value | | |
| Run Design Rules Checker (DRC) | | | |
| Create Bit File | | | |
| Create Binary Configuration File | | | |
| Create ASCII Configuration File | | | |
| Create IEEE 1532 Configuration File | | | |
| Enable BitStream Compression | | | |
| Enable Debugging of BitStream | | | |
| Enable Cyclic Redundancy Checking (CRC) | | | |
| Other Bitgen Command Line Options | | | |
| | | | |
| | | | |
| | | | |
| OK Cancel | Default Help | | |

Bring the Configuration Options to the front. Select 'Pull Up' for the 'Unused IOB Pins' item. With this done check the settings match those shown in the picture below.

| Configuration Pin M0 Pull Up Configuration Pin M1 Pull Up Configuration Pin M2 Pull Up Configuration Pin Program Pull Up Configuration Pin Program Pull Up Configuration Pin Done Pull Up Configuration Pin Powerdown Pull Up JTAG Pin TCK Pull Up JTAG Pin TDI Pull Up JTAG Pin TD0 Pull Up JTAG Pin TMS Pull Up Unused IOB Pins Pull Up UserID Code (8 Digit Hexadecimal) 0xFFFFFFFF | Startup Options Readback Options | | | Encryption Options |
|---|--|----------|------------|-----------------------|
| Configuration Rate4Configuration RatePull UpConfiguration Pin M0Pull UpConfiguration Pin M1Pull UpConfiguration Pin M2Pull UpConfiguration Pin ProgramPull UpConfiguration Pin ProgramPull UpConfiguration Pin PowerdownPull UpConfiguration Pin TCKPull UpJTAG Pin TCKPull UpJTAG Pin TDIPull UpJTAG Pin TMSPull UpUnused IOB PinsPull UpUserID Code (8 Digit Hexadecimal)OxFFFFFFFF | General Option | ns | | Configuration Options |
| Configuration Clk (Configuration Pins)Pull UpConfiguration Pin M0Pull UpConfiguration Pin M1Pull UpConfiguration Pin M2Pull UpConfiguration Pin ProgramPull UpConfiguration Pin DonePull UpConfiguration Pin PowerdownPull UpJTAG Pin TCKPull UpJTAG Pin TD0Pull UpJTAG Pin TD0Pull UpJTAG Pin TD0Pull UpJTAG Pin TD0Pull UpJTAG Pin TMSPull UpUnused IOB PinsPull UpUserID Code (8 Digit Hexadecimal)OxFFFFFFFF | Prope | rty Name | | Value |
| Configuration Pin M0 Pull Up Configuration Pin M1 Pull Up Configuration Pin M2 Pull Up Configuration Pin Program Pull Up Configuration Pin Program Pull Up Configuration Pin Done Pull Up Configuration Pin Powerdown Pull Up JTAG Pin TCK Pull Up JTAG Pin TDI Pull Up JTAG Pin TD0 Pull Up JTAG Pin TMS Pull Up Unused IOB Pins Image: Pull Up UserID Code (8 Digit Hexadecimal) 0xFFFFFFFF | Configuration Rate | | | 4 |
| Configuration Pin M1 Pull Up Configuration Pin M2 Pull Up Configuration Pin Program Pull Up Configuration Pin Done Pull Up Configuration Pin Powerdown Pull Up JTAG Pin TCK Pull Up JTAG Pin TDI Pull Up JTAG Pin TD0 Pull Up JTAG Pin TMS Pull Up Unused IOB Pins Full Up UserID Code (8 Digit Hexadecimal) 0xFFFFFFFF | Configuration Clk (Configuration Pins) | | Pull Up | |
| Configuration Pin M2 Pull Up Configuration Pin Program Pull Up Configuration Pin Done Pull Up Configuration Pin Powerdown Pull Up JTAG Pin TCK Pull Up JTAG Pin TDI Pull Up JTAG Pin TDO Pull Up JTAG Pin TDO Pull Up JTAG Pin TDO Pull Up JTAG Pin TMS Pull Up Unused IOB Pins Pull Up UserID Code (8 Digit Hexadecimal) OxFFFFFFFF | Configuration Pin M0 | | | Pull Up |
| Configuration Pin Program Pull Up Configuration Pin Done Pull Up Configuration Pin Powerdown Pull Up JTAG Pin TCK Pull Up JTAG Pin TDI Pull Up JTAG Pin TDO Pull Up JTAG Pin TDO Pull Up JTAG Pin TDO Pull Up JTAG Pin TMS Pull Up Unused IOB Pins Pull Up UserID Code (8 Digit Hexadecimal) OxFFFFFFFF | Configuration Pin M1 | | | Pull Up |
| Configuration Pin Done Pull Up Configuration Pin Powerdown Pull Up JTAG Pin TCK Pull Up JTAG Pin TDI Pull Up JTAG Pin TD0 Pull Up JTAG Pin TMS Pull Up Unused IOB Pins Pull Up UserID Code (8 Digit Hexadecimal) 0xFFFFFFFF | Configuration Pin M2 | | | Pull Up |
| Configuration Pin Powerdown Pull Up JTAG Pin TCK Pull Up JTAG Pin TDI Pull Up JTAG Pin TDO Pull Up JTAG Pin TMS Pull Up JITAG Pin TMS Pull Up Unused IOB Pins Pull Up UserID Code (8 Digit Hexadecimal) OxFFFFFFFF | Configuration Pin Program | | | Pull Up |
| JTAG Pin TCK. Pull Up JTAG Pin TDI Pull Up JTAG Pin TDO Pull Up JTAG Pin TMS Pull Up Unused IOB Pins Pull Up UserID Code (8 Digit Hexadecimal) OxFFFFFFF | Configuration Pin Done | | | Pull Up |
| JTAG Pin TDI Pull Up JTAG Pin TDO Pull Up JTAG Pin TMS Pull Up Unused IOB Pins Pull Up UserID Code (8 Digit Hexadecimal) OxFFFFFFF | Configuration Pin Powerdov | vn | | Pull Up |
| JTAG Pin TDO Pull Up JTAG Pin TMS Pull Up Unused IOB Pins Pull Up UserID Code (8 Digit Hexadecimal) OxFFFFFFF | JTAG Pin TCK | | | Pull Up |
| JTAG Pin TMS Pull Up Unused IOB Pins Pull Up UserID Code (8 Digit Hexadecimal) 0xFFFFFFF | JTAG Pin TDI | | | Pull Up |
| Unused IOB Pins Pull Up UserID Code (8 Digit Hexadecimal) 0xFFFFFFF | JTAG Pin TDO | | Pull Up | |
| UserID Code (8 Digit Hexadecimal) 0xFFFFFFF | JTAG Pin TMS | | | Pull Up |
| | Unused IOB Pins | | | Pull Up |
| | UserID Code (8 Digit Hexadecimal) | | | 0xFFFFFFFF |
| Reset DCM if SHUTDOWN & AGHIGH performed | Reset DCM if SHUTDOWN & AGHIGH performed | | | |
| | Disable Bandgap Generator for DCMs to save power | | | |
| DCI Update Mode Quiet(Off) | DCI Update Mode | | Quiet(Off) | |

Next check the settings for the 'Startup Options' tab against those shown in the picture below.

| rocess Properties | | |
|--------------------------------|------------------|-----------------------|
| General Options | | Configuration Options |
| Startup Options | Readback Options | Encryption Options |
| Property N | lame | Value |
| FPGA Start-Up Clock | | CCLK 🗾 |
| Enable Internal Done Pipe | | |
| Done (Output Events) | | Default (4) |
| Enable Outputs (Output Events) | | Default (5) |
| Release Write Enable (Output E | | Default (6) |
| Release DLL (Output Events) | | Default (NoWait) |
| Match Cycle | | Auto |
| Drive Done Pin High | | |
| | | |
| | OK Cancel | Default Help |

Next check the 'Readback Options' against the picture below.

| Process Properties | | × | |
|------------------------------|------------------|--------------------------|--|
| General Option | s | Configuration Options | |
| Startup Options | Readback Options | Encryption Options | |
| Proper | Property Name | | |
| Security | | Enable Readback and Reco | |
| Create ReadBack Data File | | | |
| Allow SelectMAP Pins to Pe | ersist | N/A | |
| Create Logic Allocation File | | N/A | |
| Create Mask File | | N/A | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| [| OK Cancel | Default Help | |

Finally check the 'Encryption Options' tab against the picture shown below.

| General Options | | Configuration Options | |
|------------------------------|------------------|-----------------------|--|
| Startup Options | Readback Options | Encryption Options | |
| Property | Name | Value | |
| Encrypt Bitstream | | | |
| Key 0 (Hex String) | | | |
| Key 1 (Hex String) | | | |
| Key 2 (Hex String) | | | |
| Key 3 (Hex String) | | | |
| Key 4 (Hex String) | | | |
| Key 5 (Hex String) | | | |
| Input Encryption Key File | | | |
| Location of Key 0 in Sequenc | e | None | |
| Location of Key 1 in Sequenc | e | None | |
| Location of Key 2 in Sequenc | e | None | |
| Location of Key 3 in Sequenc | e | None | |
| Location of Key 4 in Sequenc | e | None | |
| Location of Key 5 in Sequenc | e | None | |
| Starting Key | | None | |
| Starting CBC Value (Hex) | | | |
| | | | |
| | | | |

When all tabs have been checked, click 'OK' to enter the new settings.

Step 7: Building the Project

At this point the project is ready to be built. All of the required design files have been added to the project and the project build settings have been checked. Example1 should now build as far as bitstream generation without error.

Although each example project supplied on the HUNT ENGINEERING CD will differ from Example1, the process to create a new project for ISE 6 is the same. The routine described in this section can therefore be repeated for each of the standard examples provided on the CD. For each FPGA module type, each project contained on the CD provides all of the required design elements to ensure correct operation on that chosen module type. These elements include a Hardware Interface Layer that is used to correctly control external devices, user constraints information to control design timing and pin location and example VHDL to provide a structured starting point.

This section describes how to build an ISE 7.1 project from scratch, using the design source provided on the CD.

This document uses Example1 as the starting point for the creation of a new project. It is important to start from one of the standard examples provided on the HUNT ENGINEERING CD as this give the correct starting point for FPGA development with your FPGA module.

Please note: if you are using this section in order to create a brand new project with functionality that does not match any of the standard CD examples, then you must still start from Example1. Once you have created a correct project based around Example1 you may then insert your own unique code into the User-Ap entity, removing all unwanted logic. When doing this, you will need to refer to the relevant information in the 'Making your own FPGA design' section of your FPGA User Manual.

Converting Example1 for your Module Type

The HUNT ENGINEERING CD provides support for many different FPGA module types. For each module there is always a standard Example1 project provided on the CD. This example is the 'Getting Started' example for each module type and should always be used as the first step in FPGA development.

This section describes how to convert Example1 for the HERON-FPGA5 as an example of the conversion process. Although the design source varies for each module type, the principles shown here are the same regardless of type.

Step 1: Copying Example1 from the HUNT ENGINEERING CD

The first step in the conversion process is to copy Example1 onto your local hard drive. Make a directory on your local drive in which to store Example1.

On the HUNT ENGINEERING CD, all FPGA examples are provided below the 'fpga' directory. The 'fpga' directory is divided into sub-directories that reflect the name of the module type. In this document, Example1 for the HERON-FPGA5 is to be converted, so the directory 'fpga/fpga5v1/' will contain the appropriate project information. Identify the appropriate CD directory according to your module type.

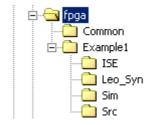
With the correct CD directory located you will need to copy Example 1 to your local drive. This can be done in one of two ways. The first method is to copy the Example 1 project by hand for the module type you are using. The second method is to unzip the ZIP file for that module type onto your hard drive.

When using the first approach you will need to copy the 'Common' directory and 'Example1' directory from the CD to a chosen directory on your local drive. The 'Example1' directory contains all design source that forms Example1 and the 'Common' directory provides design source common to all designs made for that module.

Please note, when copying files from your CD by hand you may need to edit the file attributes after copying. This is because on some operating systems Read-Only files on the CD will become Read-Only files on your local drive. All files in the 'Example1' directory should be set to Read-Write while all files in the 'Common' directory must remain Read-Only.

When using the second approach you will be unzipping all examples for that module, including Example1. When unzipping all project examples, the read-write attributes will automatically be set correctly.

The picture below shows a new directory on the local drive named 'fpga'. In the 'fpga' directory are the copied 'Common' and 'Example1' directories.



Step 2: Creating a New Project File

With the Example1 example directory and Common directory copied onto your Local Drive you can now begin building an ISE 7 project.

Open the ISE 7 Project Navigator if it is not already open. Please note, in the remainder of this section an example conversion is shown for Example 1 for the HERON-FPGA5 where this conversion is performed in ISE 7.1, with Service Pack 1 installed. If you are using a different ISE 7 version then although the windows shown may not perfectly match, the conversion process is still the same.

Next, delete the project file for the Example1 project in the new directory you have created on your local drive. This must be done, as we are about to create a new project file.

With the existing file deleted, select the menu item 'File \rightarrow New Project...' to begin creating a new project. The following window should be displayed.

| w Project | |
|-------------------------------------|--|
| Enter a Name and Location for the | Project |
| Project <u>N</u> ame: Ex1_Fpga5∨ | Project <u>L</u> ocation: c:\fpga5v1\Example1\ISE |
| Select the type of Top-Level modul | le for the Project |
| Top-Level Module Type: | |
| | |
| HDL | • |
| HDL | • |
| HDL | • |

Enter the correct project name in the 'Project Name' field. For Example1 this should be set to be the same project name as the Example1 project file on the HUNT ENGINEERING CD (minus the .ise extension).

Next, enter the correct location into the 'Project Location' field. This field should match the location of the 'Example1/ISE' directory you have made on your local drive. Next, ensure the 'Top-Level Module Type' field is set to HDL.

With these steps completed click on the 'Next >' button at the bottom of the window, to display the following:

| Property Name evice Family | Value |
|-------------------------------|--------------------|
| Device | xc2v1000 |
| Package | fg456 |
| Speed Grade | -4 |
| | |
| Top-Level Module Type | HDL |
| Synthesis Tool | XST (VHDL/Verilog) |
| Simulator | ISE Simulator |
| Generated Simulation Language | VHDL |
| | |

Fill out the appropriate device information according to the module type you are using. If you are unsure of the correct information refer to the User Manual for that module type. When you have done this click the 'Next >' button. The following window will then be displayed. Simply click 'Next >'.

| ew Proje | ct | | | | | 2 |
|----------|---|----------------------|----------------------|-------------|----------------|-----------|
| | | | | | | |
| - Create | e a New Source | | | | | |
| | | | | | | |
| | Source File | | Туре | | New Source | |
| | | | | | | |
| | | | | | Remove | |
| | | | | | | |
| | | | | | | |
| L | | | | | | |
| Create | | a the project (aptic | nal). Only one new | | - he specified | |
| Additio | e a new source to add t onal new sources can b | e added after proj | ect creation using t | he "Project | ->New Source | now. " |
| comm | | | - | · | | |
| Existin | ng sources can be adde | d on the next pag | e. | | | |
| | | | - | | | |
| | | | | | | |
| | | | | | | |
| | | | Next > | | | |
| | | < <u>B</u> ack | | Cance | | Help |

The following window will then be displayed. Again, click 'Next >'.

| | Source File | Туре | Copy to Projec | | d Source |
|-------------------|--|---|---|-------------------------|----------------|
| 1 2 | | | | | source |
| 3 | | | | | Remove |
| 4 | | | <u> </u> | - | |
| | | | | | |
| | | | | | |
| | | | | | |
| t exist na the | ting sources to the proje "Project->Add Source" | ect (optional). Addit '' or ''Proiect->Add | ional sources can be a Copy of Source'' comr | idded after j nands. | project creati |
| d exis ng the | ting sources to the proje "Project->Add Source" | ect (optional). Addit '' or ''Project->Add | ional sources can be a Copy of Source'' comm | idded after j nands. | project creati |
| d exis ng the | ting sources to the proje "Project->Add Source" | ect (optional). Addit ' or ''Project->Add | ional sources can be a Copy of Source'' comm | idded after j nands. | project creati |

Finally, the following window is displayed providing a summary of all the project information you have entered. Click 'Finish' to make the new project file.

| Project Navigator will create a new Project with the following specifications: | |
|---|---------|
| Project: Project Name: Ex1_Fpga5v Project Location: c:\fpga5v1\Example1\ISE Project Type: HDL | |
| Device: Device Family: Virtex2 Device: xc2v1000 Package: fg456 Speed Grade: -4 | |
| Top-Level Module Type: HDL Synthesis Tool: XST (VHDL/Verilog) Simulator: ISE Simulator Generated Simulation Language: VHDL | |
| < <u>B</u> ack Finish Canc | el Help |

Step 3: Adding VHDL Source to the New Project

The next step is to add design source, starting with the top level of the hierarchy. For all FPGA module projects the top level of the design is always contained in the file 'top.vhd'. 'top.vhd' is always provided as part of the 'Common' directory for each module type.

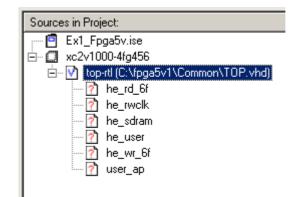
Select the menu item 'Project \rightarrow Add Source...'. The following window will be displayed where you will need to navigate to the 'Common' directory that was created on your local drive. Select the file 'top.vhd' and click 'Open'.

| Add Existing Sou | rces | | | <u>? ×</u> |
|---|---|--|-----------------|--------------|
| Look jn: | 🔁 Common | • | 수 🗈 💣 🎫 | |
| History Desktop My Documents My Computer | Fpga5v_tpl.ucf HE_CONV.vhd HE_RWCLK.vhd HE_SDRAM.vhd HE_USER.vhd HE_USER.vhd HE_WR_6F.vhd SIM_MSG.vhd SIM_RD_6F.vhd SIM_SDRAM.vhd SIM_SDRAM.vhd SIM_SDRAM.vhd SIM_SDRAM.vhd SIM_V2_RD_6F.vhd | l d d | | |
| My Network P | File <u>n</u> ame: | TOP.vhd | • | <u>O</u> pen |
| My Howont 1 | Files of <u>type</u> : | Sources (*.txt;*.vhd;*.vhd);*.v;*.abl; | *.xco;*.sch;* 💌 | Cancel |

The following window will then appear. Select 'VHDL Design File' and click OK to add 'top.vhd' to the project.

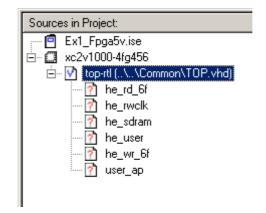
| Choose Source Type | × |
|--|--------|
| TOP.vhd is which source type? The suffix is ambiguous as to type. | |
| VHDL Design File VHDL Test Bench File | OK |
| | Cancel |
| | Help |
| Apply to all .vhd files. | |

With this done, the Module View window may look something like the picture below.



In this picture we can see that the path shown for 'top.vhd' is absolute. That is, it shows the complete path. The project needs to be changed to use relative path information. To do this, left click on the entity 'top'. With top selected, right click to bring up a menu and select 'Toggle Paths'.

The Module View should now look similar to the picture below, depending on the particular source files that are required by the module type you are using. Note, the path information has now changed to reflect a relative path to the Common directory.



In the case of the FPGA5v1 example conversion, after adding 'top.vhd' we can now see 6 more design entities are needed. One of these entities is the User-Application level of the design which contains the VHDL that makes this example the getting started example, Example1. The other entities shown are part of the Hardware Interface Layer and are found in the 'Common' directory.

First add the appropriate files from the 'Common' directory in order to replace the red question mark icons with correct entities. To do this, you will again need to use the 'Add Source...' menu item. Navigate to the 'Common' directory and select the files that have names that match the entities in the Module View of your new project.

Please note: the HE_RD_6F entity is provided in the 'Common' directory in the source file 'V2_RD_6F'.

When adding each file, the correct Source Type will be 'VHDL Design File' as was selected when adding 'top.vhd'.

When you have added all required Hardware Interface Layer components the Module View should look similar to the picture below for the FPGA5v1 conversion:

| Sources in Project: |
|--|
| Ex1_Fpga5v.ise xc2v1000-4fg456 vc1000-4fg456 v top-rtl (\\Common\TOP.vhd) v he_rd_6f-rtl (\\Common\Y2_RD_6F.vhd) v he_sdram-rtl (\\Common\HE_SDRAM.vhd) v he_sd_cntr-rtl (\\Common\HE_USER.vhd) v he_user-rtl (\\Common\HE_USER.vhd) v he_wr_6f-rtl (\\Common\HE_WR_6F.vhd) v user_ap |
| 📲 Module View 💼 Snapshot View 🖺 Library View |

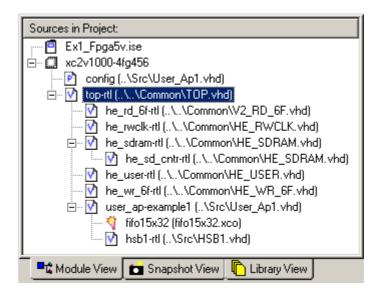
Next you will need to navigate to the 'Src' directory of Example1 and add the User_Ap file to the project:

| Add Existing Sou | rces | | | ? × |
|--|-------------------------|---|---------------|--------------|
| Look jn: | 🔁 Src | • | 🗢 🗈 💣 🎟 • | |
| History History Desktop My Documents My Computer | HSB1.vhd | | | |
| | , File <u>n</u> ame: | User_Ap1.vhd | • | <u>O</u> pen |
| My Network P | Files of type: | Sources (*.txt;*.vhd;*.vhd;*.v;*.abl;*. | xco;*.sch;* 💌 | Cancel |

After the 'user_ap' entity has been added to the project you will see which other source files are required below this entity. Add the appropriate source files from the 'Src' directory.

After you have added all of required VHDL source files from the 'Src' directory there may still be red question marks against entities in the hierarchy. These entities will correspond to Core Gen components that are placed in the Example1 'ISE' directory. In the case of Example1 there is one Core Gen component called 'fifo15x32' that must be added to the project from the 'ISE' directory.

With all relevant VHDL source added to the project the Module View should look similar to the picture shown below for the FPGA5v1 conversion:



Step 4: Adding User Constraints

The next step is to add user design constraints to the project. Using 'Project \rightarrow Add Source...' add the user constraints file contained in the 'ISE' directory of the project.

| Add Existing Sou | rces | | | ? × |
|---|---|---|---------------|--------------|
| Look jn: | 🔁 ISE | • | 🕂 🖻 📩 🎟 | |
| History Desktop My Documents My Computer | projnav xmsgs Fx1_Fpga5v.uc fifo15x32.edn fifo15x32.vhd | | | |
| | File <u>n</u> ame: | Ex1_Fpga5v.ucf | • | <u>O</u> pen |
| My Network P | Files of <u>type</u> : | Sources (*.txt;*.vhd;*.vhd];*.v;*.abl;*.; | xco;*.sch;* 💌 | Cancel |

Select the UCF file and click 'Open'.

The following window will appear to allow you to specify the design file to which the user constraints should be associated. Highlight 'top' and click 'OK'.

| Associate with Source | × |
|--|------------------|
| Associate Ex1_Fpga5v.ucf with the affects. | e source that it |
| top he_wr_6f | OK |
| he_rwclk he_rd_6f | Cancel |
| he_user he_sdram | Help |

Step 5: Adding a Simulation Test-Bench

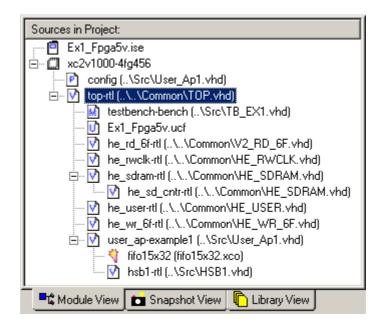
Typically, a project will also include a test-bench for simulation. For Example1 there is a test-bench provided in the 'Src' directory of the project. This file must also be added to the project using 'Project \rightarrow Add Source...'. Navigate to the 'Src' directory, highlight the file that beings 'TB_' and click 'Open'.

| Add Existing Sou | rces | | <u>? ×</u> |
|------------------------------------|--|---|--------------|
| Look jn: | Ga Src | - 🖬 🎦 🗢 💽 | |
| History Desktop My Documents | HSB1.vhd sim_fifo.vhd TB_EX1.vhd User_Ap1.vhd | | |
| My Computer | | | |
| | ' File <u>n</u> ame: | TB_EX1.vhd | <u>O</u> pen |
| My Network P | Files of type: | Sources (*.txt;*.vhd;*.vhdl;*.v;*.abl;*.xco;*.sch;* | Cancel |

The following window will appear where you need to specify the source type. Select 'VHDL Test Bench File' and click 'OK'.

| Choose Source Type | × | | |
|---|--------|--|--|
| TB_EX1.vhd is which source type? The suffix is ambiguous as to type. | | | |
| VHDL Design File VHDL Test Bench File | OK | | |
| WIDE rest bench lie | Cancel | | |
| | Help | | |
| Apply to all .vhd files. | | | |

At this point the Module View should look similar to the picture shown below for the FPGA5v1 conversion process.



Step 6: Setting Project Build Options

The last step in creating a new project is to apply the necessary project settings that will allow the design to be built correctly.

Ensure the file 'top.vhd' is highlighted in the Module View, and then right click on 'Synthesize' in the Process View. Select 'Properties...' to open the following window.

With the 'Synthesis Options' tab at the front, set 'Optimization Effort' to High and set 'Hierarchy Separator' to '_'. Having done this check that the settings match those shown below.

| Property Name | Value | | |
|--------------------------------|-----------------|--------------|----------|
| Optimization Goal | Speed | - | |
| Optimization Effort | | High | |
| Enable Auto Floorplanning | | No | |
| Use Synthesis Constraints File | | | |
| Synthesis Constraints File | | | |
| Library Search Order | | | |
| Keep Hierarchy | | No | |
| Global Optimization Goal | | AllClockNets | |
| Generate RTL Schematic | | Yes | |
| Read Cores | | | |
| Cores Search Directories | | _ | |
| Write Timing Constraints | | | |
| Cross Clock Analysis | | | |
| Hierarchy Separator | | _ | |
| Bus Delimiter | | ♦ | |
| Slice Utilization Ratio | | 100 | |
| Case | | Maintain | |
| Work Directory | | ./xst | |
| HDL INI File | | — | |
| Verilog 2001 | | | |
| Verilog Include Directories | | | |
| Custom Compile File List | | | |
| Other XST Command Line Options | | | |
| <u>P</u> ropert | ı display level | Advanced | _ |

Next bring the 'HDL Options' tab to the front and check that the settings match those in the picture below.

| Synthesis Options | HDL Options | Xilinx Specific Options | |
|----------------------------|-------------|-------------------------|----------|
| Property Name | | | Value |
| FSM Encoding Algorithm | | | Auto |
| Safe Implementatio | n | | No |
| Case Implementatio | in Style | | None |
| FSM Style | | | LUT |
| RAM Extraction | | | |
| RAM Style | | | Auto |
| ROM Extraction | | | |
| ROM Style | | | Auto |
| Mux Extraction | | | Yes |
| Mux Style | | | Auto |
| Decoder Extraction | | | |
| | | Yes | |
| Shift Register Extraction | | | |
| Logical Shifter Extraction | | | |
| XOR Collapsing | | | |
| Resource Sharing | | | |
| Multiplier Style | | Auto | |
| | | | |
| | Prop | erty display level | Advanced |

Next bring the 'Xilinx Specific Options' tab to the front. Change the 'Number of Clock Buffers' item to '0', and set the 'Pack I/O Registers into IOBs' item to 'No'. With this done, check that the settings match those in the picture below.

| Synthesis Options HDL Options Xilinx Specific Options | Value |
|---|----------|
| Property Name | |
| Add I/O Buffers | |
| Max Fanout | 500 |
| Number of Clock Buffers | 0 |
| Register Duplication | |
| Equivalent Register Removal | |
| Register Balancing | No |
| Move First Flip-Flop Stage | N/A |
| Move Last Flip-Flop Stage | N/A |
| Pack I/O Registers into IOBs | No |
| Slice Packing | |
| Convert Tristates To Logic | Yes |
| Use Clock Enable | Yes |
| Use Synchronous Set | Yes |
| Use Synchronous Reset | Yes |
| Optimize Instantiated Primitives | |
| | |
| Property display level | Advanced |

Click on the 'OK' button at the bottom of the window to apply the new settings.

Next the Translate Properties must be checked. With the file 'top.vhd' still highlighted in the Module View right click on 'Translate' in the Process View. Select 'Properties...' to open the following window. Tick the item 'Allow Unmatched LOC Constraints', check all other settings match, and click 'OK'.

| Process Properties | × | | | |
|--|--------------|--|--|--|
| Translate Properties | | | | |
| Property Name | Value | | | |
| Use LOC Constraints | | | | |
| Netlist Translation Type | Timestamp | | | |
| Macro Search Path | | | | |
| Create I/O Pads from Ports | | | | |
| Allow Unexpanded Blocks | | | | |
| User Rules File for Netlister Launcher | | | | |
| Allow Unmatched LOC Constraints | | | | |
| Preserve Hierarchy on Sub Module | | | | |
| Other Ngdbuild Command Line Options | | | | |
| Property display level | | | | |
| OK Cancel | Default Help | | | |

Next open the Process Properties window for the Map process. Check that the settings match those in the picture below.

| Property Name | Value | |
|--|------------------------|--|
| Perform Timing-Driven Packing and Placement | | |
| Map Effort Level | N/A | |
| Extra Effort | N/A | |
| Starting Placer Cost Table (1-100) | N/A | |
| Register Duplication | N/A | |
| Trim Unconnected Signals | | |
| Replicate Logic to Allow Logic Level Reduction | | |
| Allow Logic Optimization Across Hierarchy | | |
| Map to Input Functions | 4 | |
| Optimization Strategy (Cover Mode) | Area | |
| Generate Detailed MAP Report | | |
| MAP Guide Design File (.ncd) | | |
| MAP Guide Mode | None | |
| Use RLOC Constraints | | |
| Pack I/O Registers/Latches into IOBs | For Inputs and Outputs | |
| Disable Register Ordering | | |
| CLB Pack Factor Percentage | 100 | |
| Tri-state Buffer Transformation Mode | Off | |
| Map Slice Logic into Unused Block RAMs | | |
| Other Map Command Line Options | | |
| | | |
| Property display level | Advanced | |

Next check the Place and Route process properties. For Example1 the 'Place & Route Effort Level (Overall)' typically needs to be set to 'Medium'. Make this change and check all other settings match the picture below and click 'OK'.

| Property Name | Value |
|--|------------------------|
| Place & Route Effort Level (Overall) | Medium 🔽 |
| Placer Effort Level (Overrides Overall Level) | None |
| Router Effort Level (Overrides Overall Level) | None |
| Extra Effort (Highest PAR level only) | N/A |
| Starting Placer Cost Table (1-100) | 1 |
| Place And Route Mode | Normal Place and Route |
| PAR Guide Design File (.ncd) | |
| PAR Guide Mode | None |
| Use Timing Constraints | |
| Use Bonded I/Os | |
| Generate Asynchronous Delay Report | |
| Generate Post-Place & Route Static Timing Report | |
| Generate Post-Place & Route Simulation Model | |
| Other Place & Route Command Line Options | |
| <u>P</u> roperty display level | Advanced |

Next check the 'Generate Programming File' properties. With the 'General Options' tab at the front, tick the 'Create ASCII Configuration File' and then check that all other settings match for this tab.

| Process Properties | × | | | |
|---|---------------------|--|--|--|
| Startup Options Readback Options | Encryption Options | | | |
| General Options Cor | ifiguration Options | | | |
| Property Name | Value | | | |
| Run Design Rules Checker (DRC) | | | | |
| Create Bit File | V | | | |
| Create Binary Configuration File | | | | |
| Create ASCII Configuration File | | | | |
| Create IEEE 1532 Configuration File | | | | |
| Enable BitStream Compression | | | | |
| Enable Debugging of Serial Mode BitStream | | | | |
| Enable Cyclic Redundancy Checking (CRC) | | | | |
| Other Bitgen Command Line Options | | | | |
| Property display level | | | | |
| OK Cancel <u>D</u> efault Help | | | | |

Bring the Configuration Options to the front. Select 'Pull Up' for the 'Unused IOB Pins' item. With this done check the settings match those shown in the picture below.

| Startup Options Readback Options | | | Encryption Options |
|--|-----------------|-----------|---------------------|
| General Option | ns | Cor | nfiguration Options |
| Property Name | | | Value |
| Configuration Rate | | | 4 |
| Configuration Clk (Configur | ation Pins) | | Pull Up |
| Configuration Pin M0 | | | Pull Up |
| Configuration Pin M1 | | | Pull Up |
| Configuration Pin M2 | | | Pull Up |
| Configuration Pin Program | | | Pull Up |
| Configuration Pin Done | | | Pull Up |
| Configuration Pin Powerdown | | | Pull Up |
| JTAG Pin TCK | | | Pull Up |
| JTAG Pin TDI | | | Pull Up |
| JTAG Pin TDO | | | Pull Up |
| JTAG Pin TMS | | | Pull Up |
| Unused IOB Pins | | | Pull Up |
| UserID Code (8 Digit Hexadecimal) | | | 0xFFFFFFFF |
| Reset DCM if SHUTDOWN & AGHIGH performed | | | |
| Disable Bandgap Generator for DCMs to save power | | | |
| DCI Update Mode | | | Quiet(Off) |
| | | | |
| | Property displa | y level 🛛 | dvanced |

Next check the settings for the 'Startup Options' tab against those shown in the picture below.

| General Option | s | Configuration Options | | |
|-----------------------------------|------------------|-----------------------|--|--|
| Startup Options | Readback Options | Encryption Options | | |
| Prope | rty Name | Value | | |
| FPGA Start-Up Clock | | CCLK 🗸 | | |
| Enable Internal Done Pipe | | | | |
| Done (Output Events) | | Default (4) | | |
| Enable Outputs (Output Eve | Default (5) | | | |
| Release Write Enable (Outp | Default (6) | | | |
| Release DLL (Output Events) | | Default (NoWait) | | |
| Match Cycle | | Auto | | |
| Drive Done Pin High | | | | |
| Property display level Advanced 💌 | | | | |
| | | | | |

Next check the 'Readback Options' against the picture below.

| Process Properties | | × |
|------------------------------|------------------------|-------------------------|
| General Option | ns Co | nfiguration Options |
| Startup Options | Readback Options | Encryption Options |
| Prope | erty Name | Value |
| Security | | Enable Readback and F 💌 |
| Create ReadBack Data Fil | es | |
| Allow SelectMAP Pins to F | Persist | N/A |
| Create Logic Allocation File | N/A | |
| Create Mask File | N/A | |
| | Property display level | Advanced 💌 |
| | OK Cancel | Default Help |

Finally check the 'Encryption Options' tab against the picture shown below.

| Process Properties | | | | |
|-------------------------------|------------------|--------|---------------------|------|
| General Options Cor | | | nfiguration Options | |
| Startup Options | Readback Options | | Encryption Options | |
| Property Name | | | Value | |
| Encrypt Bitstream | | | | |
| Key 0 (Hex String) | | | | |
| Key 1 (Hex String) | | | | |
| Key 2 (Hex String) | | | | |
| Key 3 (Hex String) | | | | |
| Key 4 (Hex String) | | | | |
| Key 5 (Hex String) | | | | |
| Input Encryption Key File | | | | |
| Location of Key 0 in Sequence | | | None | |
| Location of Key 1 in Sequence | | | None | |
| Location of Key 2 in Sequence | | | None | |
| Location of Key 3 in Sequence | | | None | |
| Location of Key 4 in Sequence | | | None | |
| Location of Key 5 in Sequence | | | None | |
| Starting Key | | | None | |
| Starting CBC Value (Hex) | | | | |
| | | | | |
| Property display level | | | Ndvanced | • |
| | OK | Cancel | <u>D</u> efault | Help |

When all tabs have been checked, click 'OK' to enter the new settings.

Step 7: Building the Project

At this point the project is ready to be built. All of the required design files have been added to the project and the project build settings have been checked. Example1 should now build as far as bitstream generation without error.

Although each example project supplied on the HUNT ENGINEERING CD will differ from Example1, the process to create a new project for ISE 7 is the same. The routine described in this section can therefore be repeated for each of the standard examples provided on the CD.