

NI Circuit Design Suite

Getting Started with NI Circuit Design Suite

January 2007
374482A-01

Worldwide Technical Support and Product Information

ni.com

National Instruments Corporate Headquarters

11500 North Mopac Expressway Austin, Texas 78759-3504 USA Tel: 512 683 0100

Worldwide Offices

Australia 1800 300 800, Austria 43 0 662 45 79 90 0, Belgium 32 0 2 757 00 20, Brazil 55 11 3262 3599, Canada 800 433 3488, China 86 21 6555 7838, Czech Republic 420 224 235 774, Denmark 45 45 76 26 00, Finland 385 0 9 725 725 11, France 33 0 1 48 14 24 24, Germany 49 0 89 741 31 30, India 91 80 41190000, Israel 972 0 3 6393737, Italy 39 02 413091, Japan 81 3 5472 2970, Korea 82 02 3451 3400, Lebanon 961 0 1 33 28 28, Malaysia 1800 887710, Mexico 01 800 010 0793, Netherlands 31 0 348 433 466, New Zealand 0800 553 322, Norway 47 0 66 90 76 60, Poland 48 22 3390150, Portugal 351 210 311 210, Russia 7 495 783 68 51, Singapore 1800 226 5886, Slovenia 386 3 425 42 00, South Africa 27 0 11 805 8197, Spain 34 91 640 0085, Sweden 46 0 8 587 895 00, Switzerland 41 56 200 51 51, Taiwan 886 02 2377 2222, Thailand 662 278 6777, United Kingdom 44 0 1635 523545

For further support information, refer to Appendix A, “Technical Support and Professional Services”. To comment on National Instruments documentation, refer to the National Instruments Web site at ni.com/info and enter the info code feedback.

© 2007 National Instruments Corporation. All rights reserved.

Important Information

Warranty

The media on which you receive National Instruments software are warranted not to fail to execute programming instructions, due to defects in materials and workmanship, for a period of 90 days from date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace software media that do not execute programming instructions if National Instruments receives notice of such defects during the warranty period. National Instruments does not warrant that the operation of the software shall be uninterrupted or error free.

A Return Material Authorization (RMA) number must be obtained from the factory and clearly marked on the outside of the package before any equipment will be accepted for warranty work. National Instruments will pay the shipping costs of returning to the owner parts which are covered by warranty.

National Instruments believes that the information in this document is accurate. The document has been carefully reviewed for technical accuracy. In the event that technical or typographical errors exist, National Instruments reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult National Instruments if errors are suspected. In no event shall National Instruments be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, NATIONAL INSTRUMENTS MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF NATIONAL INSTRUMENTS SHALL BE LIMITED TO THE AMOUNT THEREFORE PAID BY THE CUSTOMER. NATIONAL INSTRUMENTS WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. This limitation of the liability of National Instruments will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against National Instruments must be brought within one year after the cause of action accrues. National Instruments shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's failure to follow the National Instruments installation, operation, or maintenance instructions; owner's modification of the product; owner's abuse, misuse, or negligent acts; and power failure or surges, fire, flood, accident, actions of third parties, or other events outside reasonable control.

Copyright

Under the copyright laws, this publication may not be reproduced or transmitted in any form, electronic or mechanical, including photocopying, recording, storing in an information retrieval system, or translating, in whole or in part, without the prior written consent of National Instruments Corporation.

National Instruments respects the intellectual property of others, and we ask our users to do the same. NI software is protected by copyright and other intellectual property laws. Where NI software may be used to reproduce software or other materials belonging to others, you may use NI software only to reproduce materials that you may reproduce in accordance with the terms of any applicable license or other legal restriction.

BSIM3 and BSIM4 are developed by the Device Research Group of the Department of Electrical Engineering and Computer Science, University of California, Berkeley and copyrighted by the University of California. The ASM51 cross assembler bundled with Multisim MCU is a copyrighted product of Metalink Corp. (www.metaice.com). MPASM™ macro assembler and related documentation and literature is reproduced and distributed by Electronics Workbench under license from Microchip Technology Inc. All rights reserved by Microchip Technology Inc. MICROCHIP SOFTWARE OR FIRMWARE AND LITERATURE IS PROVIDED "AS IS," WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL MICROCHIP BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY ARISING OUT OF OR IN CONNECTION WITH THE SOFTWARE OR FIRMWARE OR THE USE OF OTHER DEALINGS IN THE SOFTWARE OR FIRMWARE.

Trademarks

National Instruments, NI, ni.com, and LabVIEW are trademarks of National Instruments Corporation. Refer to the *Terms of Use* section on ni.com/legal for more information about National Instruments trademarks. Ultiboard is a registered trademark and Multisim and Electronics Workbench are trademarks of Electronics Workbench. Other product and company names mentioned herein are trademarks or trade names of their respective companies.

Members of the National Instruments Alliance Partner Program are business entities independent from National Instruments and have no agency, partnership, or joint-venture relationship with National Instruments.

Patents

For patents covering National Instruments products, refer to ni.com/patents.

Some portions of this product are protected under United States Patent No. 6,560,572.

WARNING REGARDING USE OF NATIONAL INSTRUMENTS PRODUCTS

(1) NATIONAL INSTRUMENTS PRODUCTS ARE NOT DESIGNED WITH COMPONENTS AND TESTING FOR A LEVEL OF RELIABILITY SUITABLE FOR USE IN OR IN CONNECTION WITH SURGICAL IMPLANTS OR AS CRITICAL COMPONENTS IN ANY LIFE SUPPORT SYSTEMS WHOSE FAILURE TO PERFORM CAN REASONABLY BE EXPECTED TO CAUSE SIGNIFICANT INJURY TO A HUMAN.

(2) IN ANY APPLICATION, INCLUDING THE ABOVE, RELIABILITY OF OPERATION OF THE SOFTWARE PRODUCTS CAN BE IMPAIRED BY ADVERSE FACTORS, INCLUDING BUT NOT LIMITED TO FLUCTUATIONS IN ELECTRICAL POWER SUPPLY, COMPUTER HARDWARE MALFUNCTIONS, COMPUTER OPERATING SYSTEM SOFTWARE FITNESS, FITNESS OF COMPILERS AND DEVELOPMENT SOFTWARE USED TO DEVELOP AN APPLICATION, INSTALLATION ERRORS, SOFTWARE AND HARDWARE COMPATIBILITY PROBLEMS, MALFUNCTIONS OR FAILURES OF ELECTRONIC MONITORING OR CONTROL DEVICES, TRANSIENT FAILURES OF ELECTRONIC SYSTEMS (HARDWARE AND/OR SOFTWARE), UNANTICIPATED USES OR MISUSES, OR ERRORS ON THE PART OF THE USER OR APPLICATIONS DESIGNER (ADVERSE FACTORS SUCH AS THESE ARE HEREAFTER COLLECTIVELY TERMED "SYSTEM FAILURES"). ANY APPLICATION WHERE A SYSTEM FAILURE WOULD CREATE A RISK OF HARM TO PROPERTY OR PERSONS (INCLUDING THE RISK OF BODILY INJURY AND DEATH) SHOULD NOT BE RELIANT SOLELY UPON ONE FORM OF ELECTRONIC SYSTEM DUE TO THE RISK OF SYSTEM FAILURE. TO AVOID DAMAGE, INJURY, OR DEATH, THE USER OR APPLICATION DESIGNER MUST TAKE REASONABLY PRUDENT STEPS TO PROTECT AGAINST SYSTEM FAILURES, INCLUDING BUT NOT LIMITED TO BACK-UP OR SHUT DOWN MECHANISMS. BECAUSE EACH END-USER SYSTEM IS CUSTOMIZED AND DIFFERS FROM NATIONAL INSTRUMENTS' TESTING PLATFORMS AND BECAUSE A USER OR APPLICATION DESIGNER MAY USE NATIONAL INSTRUMENTS PRODUCTS IN COMBINATION WITH OTHER PRODUCTS IN A MANNER NOT EVALUATED OR CONTEMPLATED BY NATIONAL INSTRUMENTS, THE USER OR APPLICATION DESIGNER IS ULTIMATELY RESPONSIBLE FOR VERIFYING AND VALIDATING THE SUITABILITY OF NATIONAL INSTRUMENTS PRODUCTS WHENEVER NATIONAL INSTRUMENTS PRODUCTS ARE INCORPORATED IN A SYSTEM OR APPLICATION, INCLUDING, WITHOUT LIMITATION, THE APPROPRIATE DESIGN, PROCESS AND SAFETY LEVEL OF SUCH SYSTEM OR APPLICATION.

Documentation Conventions

When this guide refers to a toolbar button, an image of the button appears in the left column.

This guide uses the convention **Menu/Item** to indicate menu commands. For example, “**File/Open**” means choose the **Open** command from the **File** menu.

This guide uses the convention of an arrow (➤) to indicate the start of procedural information.

Getting Started with NI Circuit Design Suite uses the construction CTRL-KEY and ALT-KEY to indicate when you need to hold down the “Ctrl” or “Alt” key on your keyboard and press another key.

Table of Contents

1. Introduction to NI Circuit Design Suite

- 1.1 NI Circuit Design Suite Product Line 1-1
- 1.2 The Tutorials 1-1

2. Multisim Tutorial

- 2.1 Overview 2-1
- 2.2 Schematic Capture 2-2
 - 2.2.1 Opening and Saving the File 2-3
 - 2.2.2 Placing the Components 2-3
 - 2.2.3 Wiring the Circuit 2-7
- 2.3 Simulation 2-9
 - 2.3.1 Virtual Instrumentation 2-9
 - 2.3.2 Analysis 2-10
 - 2.3.3 The Grapher 2-12
 - 2.3.4 The Postprocessor 2-12
- 2.4 Reports 2-12
 - 2.4.1 Bill of Materials 2-13

3. Ultiboard Tutorial

- 3.1 Opening the Tutorial 3-2
- 3.2 Creating a Board Outline 3-3
- 3.3 Placing Components 3-5
 - 3.3.1 Dragging Components from Outside the Board Outline 3-6
 - 3.3.2 Dragging Components from the Parts Tab 3-7
 - 3.3.3 Placing the Tutorial Components 3-7
 - 3.3.4 Placing Parts from the Database 3-8
 - 3.3.5 Moving Components 3-10
- 3.4 Placing Traces 3-11
 - 3.4.1 Placing a Manual Trace 3-12
 - 3.4.2 Placing a Follow-me Trace 3-14
 - 3.4.3 Placing a Connection Machine Trace 3-15

3.5	Auto Part Placement	3-16
3.6	Autorouting Traces	3-17
3.7	Other Features	3-18
3.8	Preparing for Manufacturing/Assembly	3-19
3.8.1	Cleaning up the Board	3-19
3.8.2	Adding Comments.	3-19
3.8.3	Exporting a File	3-20
3.9	Viewing Designs in 3D	3-20

4. Multisim MCU Module Tutorial

4.1	Overview	4-1
4.2	About the Tutorial	4-2
4.2.1	Understanding the Assembly Program	4-4
4.3	Advanced Features.	4-7
4.3.1	Debug View Overview.	4-7
4.3.2	Adding a Breakpoint	4-9
4.3.3	Break and Step	4-10
4.3.4	Break and Step Out.	4-12
4.3.5	Break and Step Into	4-12
4.3.6	Break and Step Over.	4-12
4.3.7	Run to Cursor	4-12

Appendix A - Support and Services

A.1	Technical Support and Professional Services	A-1
-----	---	-----

Chapter 1

Introduction to NI Circuit Design Suite

Some of the features described in this book may not be available in your edition of NI Circuit Design Suite. Refer to the release notes for a list of the features in your edition.

The following are described in this chapter.

Subject	Page No.
NI Circuit Design Suite Product Line	1-1
The Tutorials	1-1

1.1 NI Circuit Design Suite Product Line

National Instruments Circuit Design Suite is a suite of EDA (Electronics Design Automation) tools that assists you in carrying out the major steps in the circuit design flow.

Multisim is the schematic capture and simulation program designed for schematic entry, simulation, and feeding to downstream steps, such as PCB layout. *Multisim* also includes mixed analog/digital simulation capability.

The *Multisim MCU Module* adds microcontroller co-simulation.

Ultiboard, fed from Multisim, is used to design printed circuit boards, perform certain basic mechanical CAD operations, and prepare them for manufacturing. Ultiboard also provides automated parts placement and layout.

1.2 The Tutorials

This book contains the following step-by-step tutorials:

- **Multisim Tutorial** — introduces you to Multisim and its many functions.
- **Ultiboard Tutorial** — shows you how to place the components and traces for the circuit

described in the Multisim Tutorial chapter. You will also learn how to autoplacement parts and then autoroute them.

- **Multisim MCU Module Tutorial** — leads you through the process of simulating and debugging a circuit that contains a microcontroller.

For more detailed information on the features discussed in these chapters, refer to the *Multisim User Guide*, the *Ultiboard User Guide*, or the *Multisim MCU Module User Guide*.

Chapter 2

Multisim Tutorial

This chapter contains a tutorial that introduces you to Multisim and its many functions. Some of the features described in this chapter may not be available in your edition of Multisim. Refer to the release notes for a list of the features in your edition. The following are described in this chapter.

Subject	Page No.
Overview	2-1
Schematic Capture	2-2
Opening and Saving the File	2-3
Placing the Components	2-3
Wiring the Circuit	2-7
Simulation	2-9
Virtual Instrumentation	2-9
Analysis	2-10
The Grapher	2-12
The Postprocessor	2-12
Reports	2-12
Bill of Materials	2-13

2.1 Overview

This tutorial leads you through the circuit design flow, from schematic capture, through simulation and analysis. After following the steps outlined on the following pages, you will have designed a circuit that samples a small analog signal, amplifies it and then counts the occurrences of the signal on a simple digital counter.



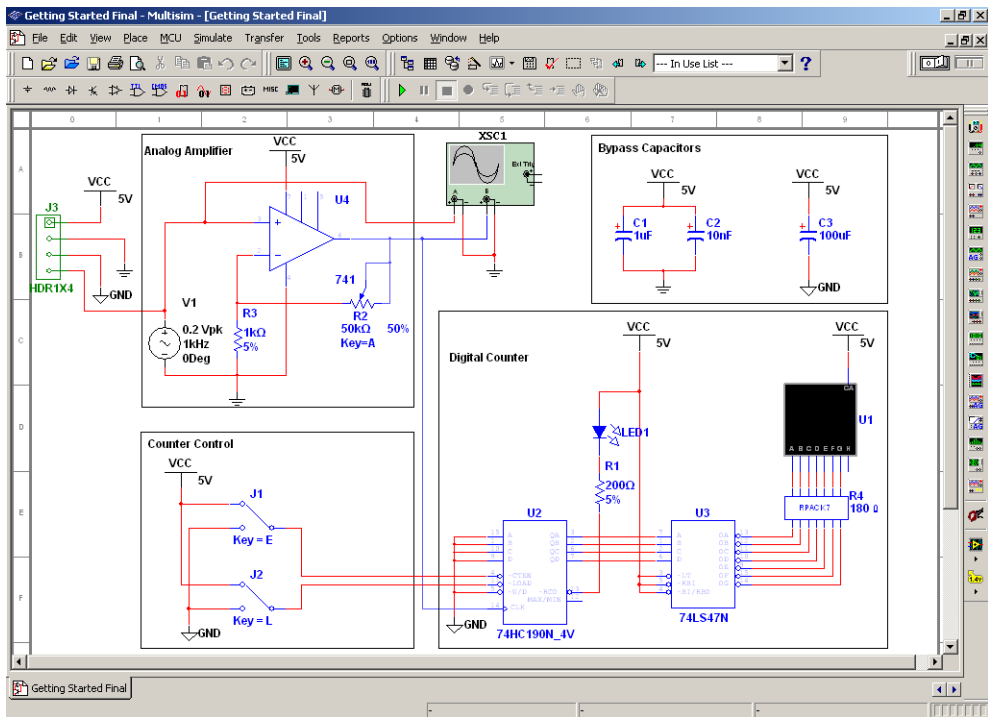
Helpful tips are indicated by the presence of an icon in the left column, as in:

You can access the online help at any time by pressing F1 on your keyboard, or by clicking on the **Help** button in a dialog box.

When you get to the wiring section of this tutorial, you can carry on with the circuit you created in the component placement section, or open the file *Getting Started 1.ms10* (which has all components properly placed) from the *Getting Started* folder (found inside the *samples* folder) and proceed. When you arrive at the simulation section, you can carry on with the circuit you wired, or open the file *Getting Started 2.ms10* (which has all components properly wired).

2.2 Schematic Capture

In this section, you will place and wire the components in the circuit shown below.



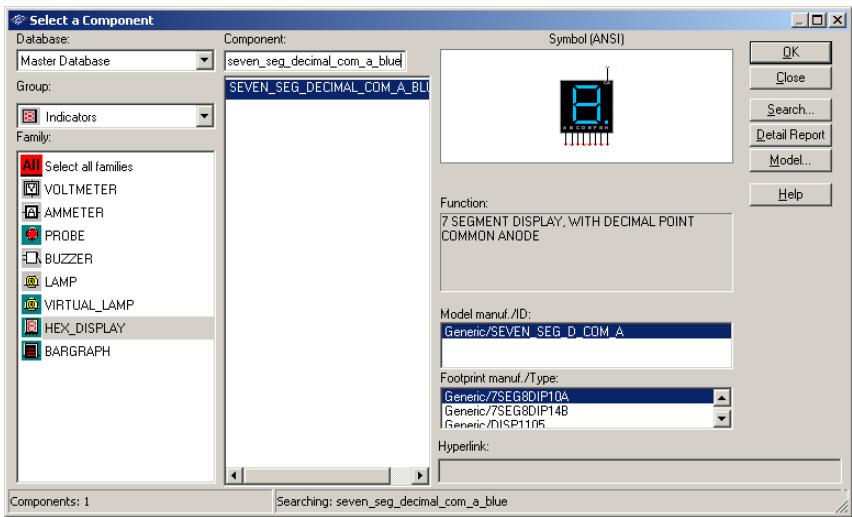
2.2.1 Opening and Saving the File

- To launch Multisim:
 1. Select **Start > All Programs > National Instruments > Circuit Design Suite 10.0 > Multisim**. A blank file opens on the workspace called "Circuit1".
- To save the file with a new name:
 1. Select **File/Save As** to display a standard Windows Save dialog.
 2. Navigate to the location where you wish the file to reside, enter `MyGettingStarted` as the filename, and click the **Save** button.
-  To guard against accidental loss of data, set up a timed auto-backup of the file in the **Save** tab of the **Preferences** dialog box.
- To open an existing file:
 1. Select **File/Open**, navigate to the location where the file resides, highlight the file, and click on the **Open** button.
-  To view files from earlier versions of Multisim, select the desired version in the **Files of Type** drop-down in the **Open** dialog.

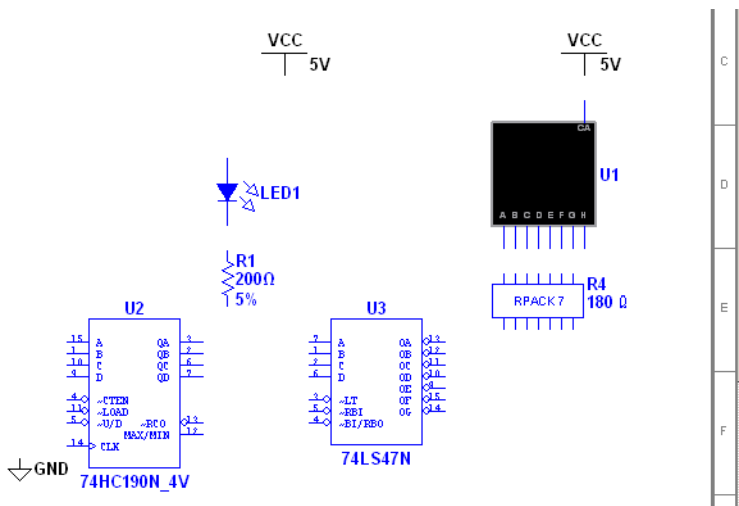
2.2.2 Placing the Components

- To start placing components:
 1. Open `MyGettingStarted.ms10` as described above.
 2. Select **Place/Component** to display the **Select a Component** browser, navigate to the 7-segment LED display as shown below and click **OK**. The component appears as a "ghost" on the cursor.
-  Once you have selected the desired **Group** and **Family**, start typing the component's name in the browser's **Component** field. As you type, the string appears in the **Searching** field at

the bottom of the browser. In the example below, type “seven_seg_decimal_com_a_blue”. Matches are displayed as you type.



3. Move the cursor to the bottom-right of the workspace and left-click to place the component. Note that the Reference Designator for this component is "U1".
4. Place the remaining components in the Digital Counter area as shown below.



Note When placing resistors, inductors, or capacitors (RLC components), the **Select a Component** browser has slightly different fields than for other components. When placing any of these components, you can choose any combination of: the

component's value (for example, the resistance value); type (for example, carbon film); tolerance; footprint and manufacturer. If you are placing a component that will be ultimately exported to PCB layout, and become part of a **Bill of Materials**, you must be careful that the combination of values that you select in the **Select a Component** dialog box are available in a real-world, purchaseable component.



When placing RLC components, type the value of the device that you want to place in the field at the top of the **Component** list. The value does not need to appear in the list to be placed on the schematic.

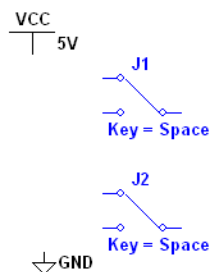


While placing the 200-ohm resistor, rotate it to a vertical orientation by pressing CTRL-R on your keyboard.



Reference Designators (for example, U1, U2) are assigned in the order the components are placed. If you place components in a different order than in the original circuit, the numbering will differ. This will not affect the operation of the circuit in any way.

5. Place the parts in the Counter Control section. After placement, right-click on each of the SPDT switches and select **Flip Horizontal**.

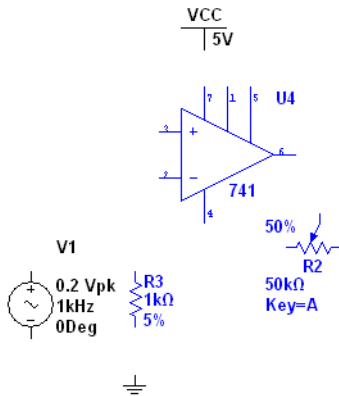


The SPDT switches are in the Basic Group; Switch Family



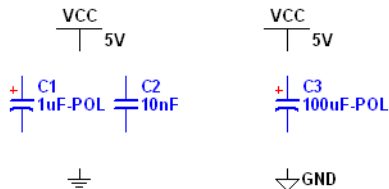
When a part is on the workspace and you want to place the same part again, highlight it and select **Edit/Copy**, then **Edit/Paste**. You can also select it from the **In Use List** and click to place it on the workspace.

6. Place the parts in the Analog Amplifier section as shown below, rotating as needed.

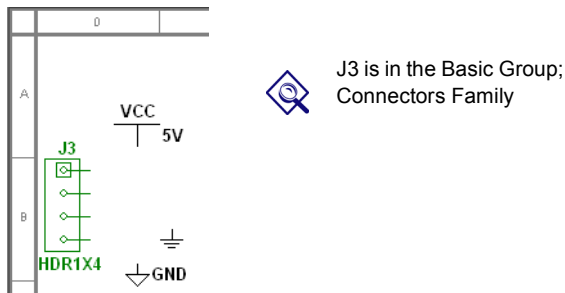


After you place the AC voltage signal source, double-click on it. Change the **Voltage (Pk)** to 0.2 V and click **OK** to close the dialog.

7. Place the parts in the Bypass Capacitors section as shown below.



8. Place the header and associated parts as shown below.



Once you have wired a circuit, you can drop two-pinned passive components like resistors directly onto a wire. The connection is automatically made by Multisim.

2.2.3 Wiring the Circuit

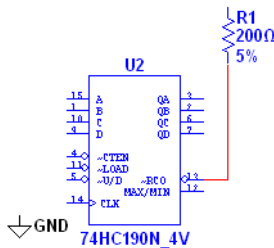
All components have pins that you use to wire them to other components or instruments. As soon as your cursor is over a pin, Multisim knows you want to wire and the pointer changes to a crosshair.



You can wire the circuit that you placed on the workspace or you can use Getting Started 1.ms10 from the Getting Started folder (found inside the samples folder).

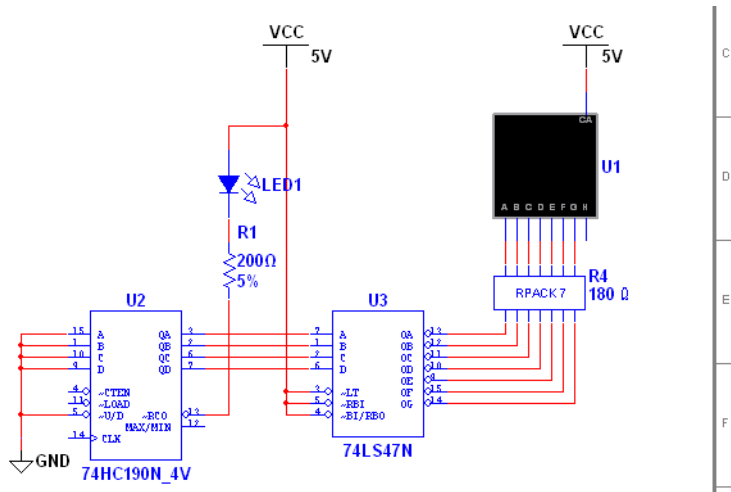
➤ To wire the circuit:

1. Click on a pin on a component to start the connection (your pointer turns into a crosshair) and move the mouse. A wire appears, attached to your cursor.
2. Click on a pin on the second component to finish the connection. Multisim automatically places the wire, which conveniently snaps to an appropriate configuration, as shown below. This feature saves a great deal of time when wiring large circuits.



3. You can also control the flow of the wire by clicking on points as you move the mouse. Each click "fixes" the wire to that point.

4. Finish wiring the Digital Counter section as shown below.

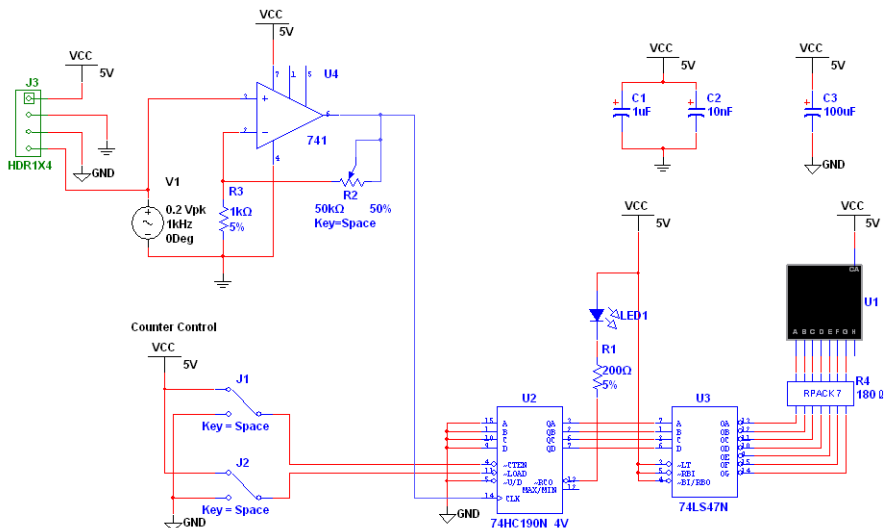


Use *Bus Vector Connect* to wire multi-pinned devices like U3 and R4 together. Refer to the *Multisim User Guide* for details.



Virtual Wiring - To avoid clutter, you could use virtual connections between the Counter Control and Digital Counter sections. When two nets have the same net name, they are virtually connected.

5. Finish wiring the circuit as shown below.



2.3 Simulation

Simulating your circuits with Multisim catches errors early in the design flow, saving time and money.

2.3.1 Virtual Instrumentation

In this section, you will simulate the circuit with the virtual oscilloscope.



You can also use `Getting Started 2.ms10` from the `Getting Started` folder (found inside the `samples` folder).

1. J1, J2 and R2 are interactive components.

Set up the interactive keys for J1, J2 and R2 by double-clicking on each. In the **Key** field, enter "E" for J1, "L" for J2, and "A" for R2.

Press "E" to enable the counter, or just click on the widened switch arm that appears when you hover the cursor over J1.

2. Select **Simulate/Instruments/Oscilloscope** to place the oscilloscope on the workspace. Wire the instrument as shown in step 4.

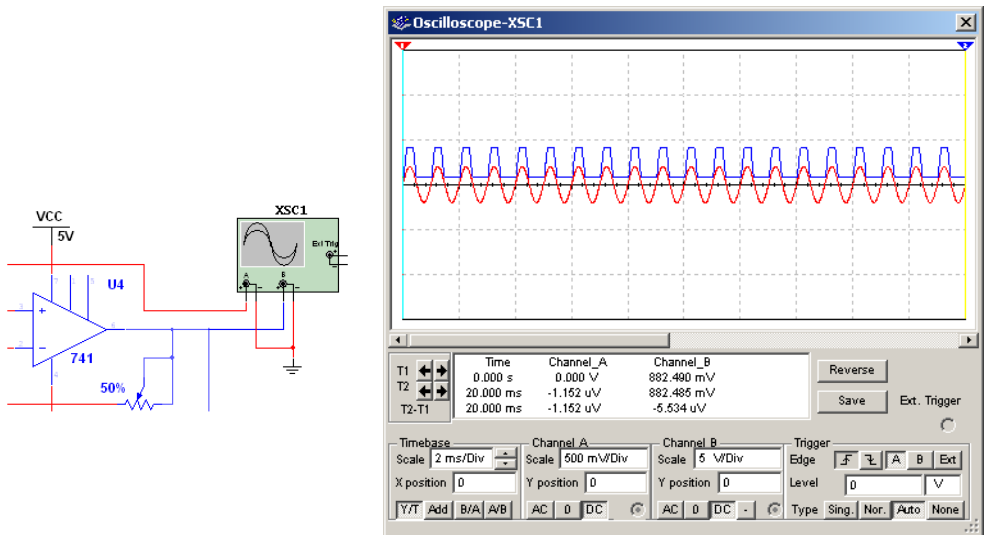


To easily differentiate between traces on the oscilloscope, right-click on the wire connected to the scope's "B" input and select **Segment Color** from the pop-up. Select a color that differs from the wire connected to the "A" input, for example blue. (Simulation cannot be running when changing wire color or performing any other editing function).



3. Double-click on the scope's icon to show the instrument face. Select **Simulate/Run**. The output of the opamp appears on the scope.

4. Adjust the Timebase to 2mS/Div and Channel A's Scale to 500mV/Div. You will see the following displayed on the scope.



As the circuit simulates, the 7-segment display counts up and the LED flashes at the end of each count cycle.

5. Press "E" on your keyboard while the simulation is running to enable or disable the counter. Enable is Active Low.

Press "L" to load zeros into the counter. Load is Active Low.

Press "Shift-A" to observe the effect of changing the potentiometer's setting. Repeat, pressing "A".



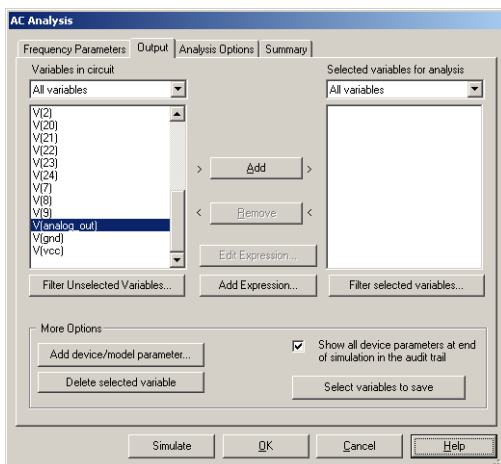
Instead of pressing the above-mentioned keys, you can directly manipulate the interactive components on the schematic with your mouse.

2.3.2 Analysis

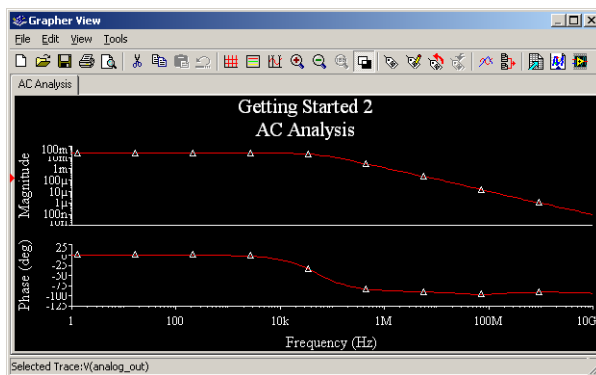
In this section, you will use **AC Analysis** to verify the frequency response of the amplifier.

- To perform an **AC Analysis** at the output of the opamp:
 1. Double-click on the wire that is attached to pin 6 of the opamp, and change the net name to "analog_out" in the **Net** dialog box.

2. Select **Simulate/Analyses/AC Analysis** and click on the **Output** tab.



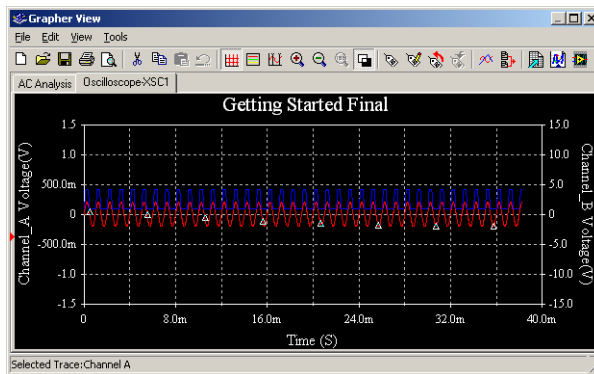
3. Highlight **V (analog_out)** in the left column and click **Add**. **V (analog_out)** moves to the right column.
4. Click **Simulate**. The results of the analysis appear in the **Grapher**.



2.3.3 The Grapher

The **Grapher** is a multi-purpose display tool that lets you view, adjust, save and export graphs and charts. It is used to display the results of all Multisim analyses in graphs and charts and a graph of traces for some instruments (for example the results of the oscilloscope).

- To view results of a simulation on the **Grapher**:
 1. Run the simulation as described earlier.
 2. Select **View/Grapher**.



2.3.4 The Postprocessor

The **Postprocessor** lets you manipulate the output from analyses performed on a circuit and plot the results on a graph or chart. Types of mathematical operations that can be performed on analysis results include arithmetic, trigonometric, exponential, logarithmic, complex, vector and logic.

2.4 Reports

Multisim allows you to generate a number of reports: Bill of Materials (BOM), Component Detail Report, Netlist Report, Schematic Statistics, Spare Gates and the Cross Reference Report. This section uses the **BOM** as an example for the tutorial circuit.

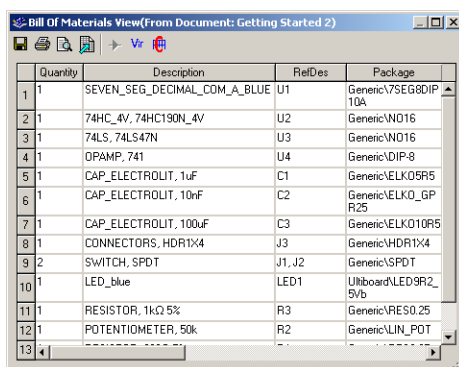
2.4.1 Bill of Materials

A bill of materials lists the components used in your design and therefore provides a summary of the components needed to manufacture the circuit board. Information provided includes:

- quantity of each component needed
- description, including the type of part (example: resistor) and value (example: 5.1 kohm)
- Reference Designator of each component
- package or footprint of each component

➤ To create a **BOM** (bill of materials) for your circuit:

1. Click the **Reports** menu and choose **Bill of Materials** from the menu that appears.
2. The report appears, looking similar to this:



	Quantity	Description	RefDes	Package
1	1	SEVEN_SEG_DECIMAL_COM_A_BLUE	U1	Generic\7SEG8DIP 10A
2	1	74HC_4V, 74HC190N_4V	U2	Generic\N016
3	1	74LS_74LS47N	U3	Generic\N016
4	1	OPAMP, 741	U4	Generic\DIP-8
5	1	CAP_ELECTROLIT, 1uF	C1	Generic\ELK05R5
6	1	CAP_ELECTROLIT, 10nF	C2	Generic\ELK0_GP R25
7	1	CAP_ELECTROLIT, 100uF	C3	Generic\ELK010R5
8	1	CONNECTORS, HDR1x4	J3	Generic\HDR1x4
9	2	SWITCH, SPDT	J1, J2	Generic\SPDT
10	1	LED_blue	LED1	Ultiboard\LED9R2_5Vb
11	1	RESISTOR, 1kQ 5%	R3	Generic\RES0.25
12	1	POTENTIOMETER, 50k	R2	Generic\LIN_POT
13				



➤ To print the **Bill of Materials**, click the **Print** button. A standard Windows print screen appears, allowing you to choose the printer, number of copies, and so on.



➤ To save the **Bill of Materials** to a file, click the **Save** button. A standard Windows file save dialog box appears, allowing you to specify the path and file name.

Because the **Bill of Materials** is primarily intended to assist in procurement and manufacturing, it includes only “real” parts — it excludes parts that are not real or able to be purchased, such as sources or virtual components.



➤ To see a list of components in your circuit that are not “real” components, click the **Virtual** button. A separate window appears, showing these components only.

Detailed information on this and other reports can be found in the *Multisim User Guide*.

Chapter 3

Ultiboard Tutorial



The tutorial in this chapter places the components and traces for the circuit described in the Multisim Tutorial chapter.

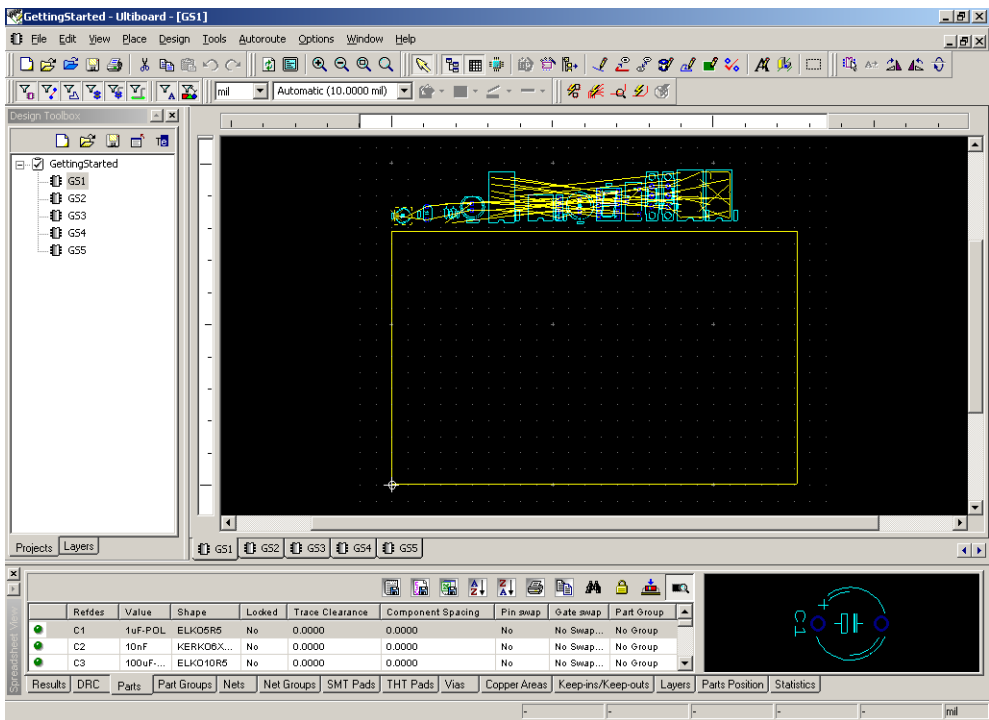
For instructions on exporting a design from Multisim to Ultiboard, refer to the *Multisim User Guide*, the *Ultiboard User Guide*, or the helpfiles.

The following are described in this chapter.

Subject	Page No.
Opening the Tutorial	3-2
Creating a Board Outline	3-3
Placing Components	3-5
Dragging Components from Outside the Board Outline	3-6
Dragging Components from the Parts Tab	3-7
Placing the Tutorial Components	3-7
Placing Parts from the Database	3-8
Moving Components	3-10
Placing Traces	3-11
Placing a Manual Trace	3-12
Placing a Follow-me Trace	3-14
Placing a Connection Machine Trace	3-15
Auto Part Placement	3-16
Autorouting Traces	3-17
Other Features	3-18
Preparing for Manufacturing/Assembly	3-19
Cleaning up the Board	3-19
Adding Comments	3-19
Exporting a File	3-20
Viewing Designs in 3D	3-20

3.1 Opening the Tutorial

- To open the tutorial file:
 1. Select **Start > All Programs > National Instruments > Circuit Design Suite 10.0 > Ultiboard** to launch Ultiboard.
 2. Select **File/Open**.
 3. Navigate to the quickstart folder found in your NI Circuit Design Suite installation directory (for example, ... \samples\Getting Started), select **Getting Started.ewprj** and click **Open**. The project file is loaded into Ultiboard.



4. To select a design (for example, GS1) either click on its tab, or click on its name in the **Projects** tab of the **Design Toolbox**.

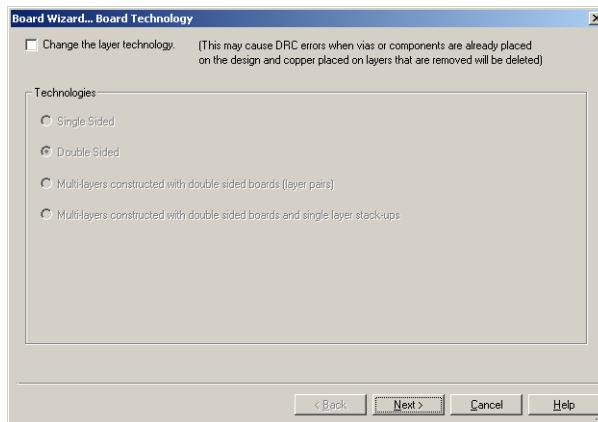
3.2 Creating a Board Outline

You can create a board outline in one of the following ways:

- draw a board outline using the drawing tools
- import a DXF file
- use the **Board Wizard**.

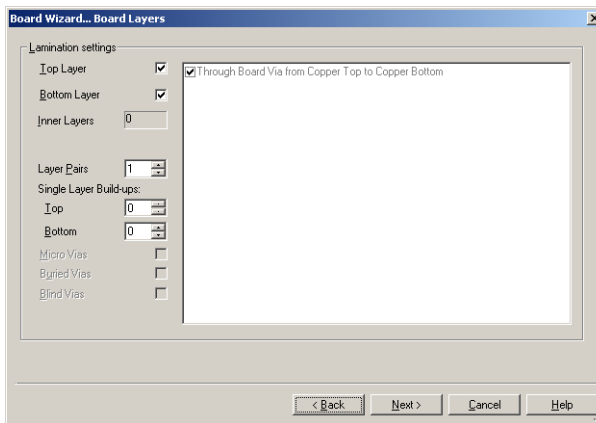
➤ To experiment with the **Board Wizard**:

1. Double-click on **Board Outline** in the **Layers** tab.
2. Click on the existing board outline in the **Quick1** design and press **DELETE**.
3. Choose **Tools/Board Wizard**.



4. Enable the **Change the layer technology** option to make the other options available.
5. Choose **Multi-layers constructed with double sided boards and single layer stack ups**, and click **Next**.

6. The next dialog box allows you to define the **Lamination Settings** for the board. (For this tutorial you will not change settings).



7. Click **Next**.

In the **Shape of Board** dialog box:

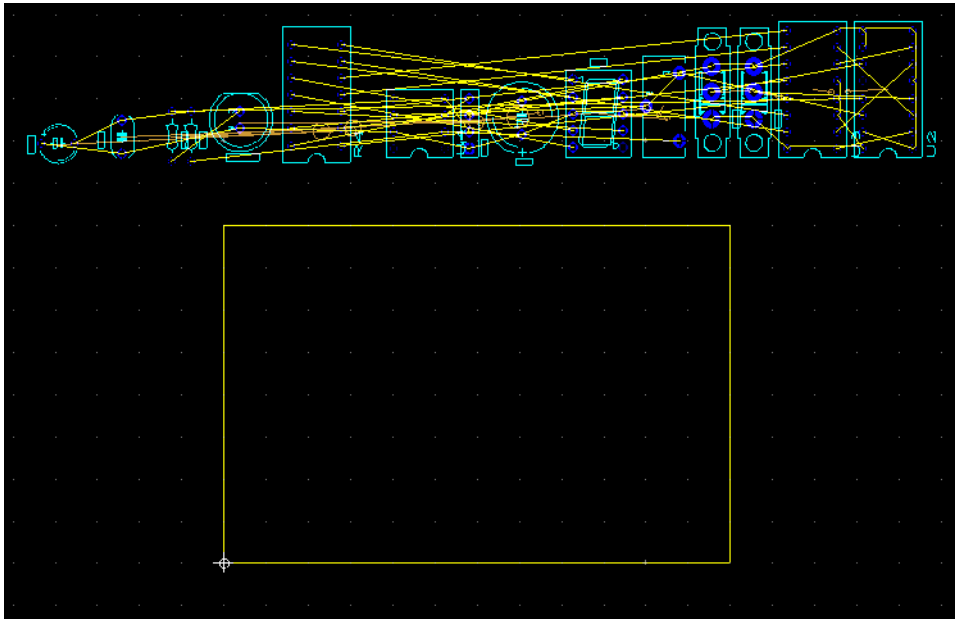
- make sure the **Reference Point** is set to **Left-Bottom** for **Alignment**
- make sure the **Rectangular** option is selected
- set the **Width** to 3000 and the **Height** to 2000 (a more suitable size for the components in this design)
- set the **Clearance** to 5.00000. This is the distance from the edge of the board that is to be kept free of any other elements.

8. Click **Finish**. The board outline is placed on your design.

Note For complete details on the **Board Wizard**, refer to the *Ultiboard User Guide*.

- To move the board outline:
 1. Double-click on **Board Outline** in the **Layers** tab.
 2. Click anywhere on the board outline in the workspace and drag the board to a location just below the row of components.
- To change the reference point:
 1. Select **Design/Set Reference Point**. The reference point is attached to your cursor.

2. Move the cursor to the lower-left corner of the board outline and click to place it.



3.3 Placing Components

You can place components on your GS1 design file in several different ways:

- select one or more components from outside the board outline and drag them into place
- use the **Parts** tab in the **Spreadsheet View** to locate components and place them
- select parts from the database.



You can use the **Place/Unplace Components** command to quickly remove all non-locked components from the PCB and experiment with a different placement technique.

3.3.1 Dragging Components from Outside the Board Outline

By default, components are placed outside the board outline when you open a netlist from Multisim or another schematic capture program. Before you begin, double-click the **Copper Top** layer in the **Design Toolbox** to make it the active layer.

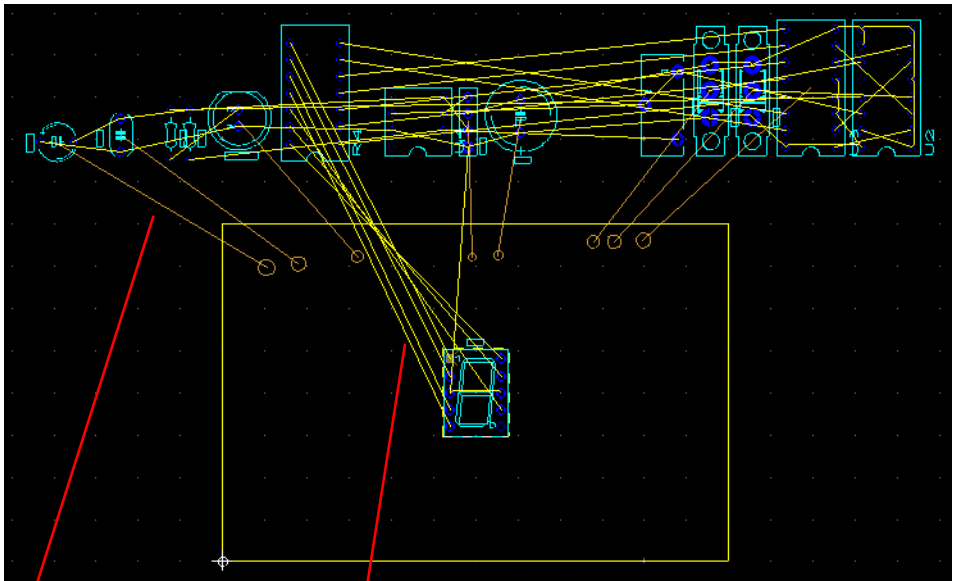
- To drag U1 from outside the board outline:

1. Find U1 in the collection of components outside the board outline. To make this easier, zoom in (press F8) until you can see U1.



You can also search for a part with the **Edit/Find** command. While this command works much like a Find function in other applications, it also allows you to search for a part by name, number, shape, value, or by all variables. Refer to the *Ultiboard User Guide* for details.

2. Click on U1 (the 7-segment display) and drag it to the center of the board.



Force Vector. For details, refer to the *Ultiboard User Guide*.

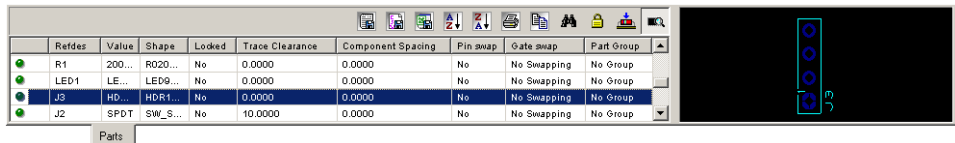
Ratsnest. Refer to the *Ultiboard User Guide* for details.

U1 remains selected. This is an important point for Ultiboard that holds throughout the application — you need to explicitly end any particular action. In this case, simply clicking somewhere else de-selects the component. Right-clicking also ends the current action.

- Go to the **Parts** tab in the **Spreadsheet View** and scroll to U1. You will notice that the green light beside the component is slightly brighter — this indicates that the component has been placed.

3.3.2 Dragging Components from the Parts Tab

- To drag components from the **Parts** tab:
 - In the **Parts** tab, scroll down until you see J3.



- Click on J3 and drag it from the **Parts** tab onto the workspace. J3 is attached to your mouse pointer.
- Drop J3 on the left side of the board, roughly in the middle. As before, in the **Parts** tab J3's green light is slightly brighter, indicating that the component has been placed.

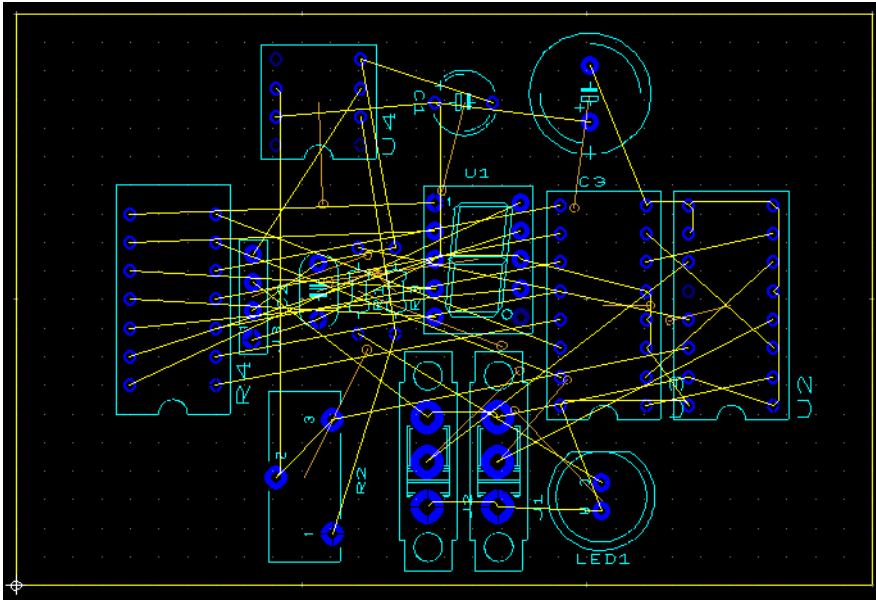


For even more rapid placement of parts, in the **Parts** tab select an unplaced part (its green light is dim) and click the **Start Placing the Unpositioned Parts** button. Ultiboard systematically goes through the list of components in the **Parts** tab, selecting each one and attaching it to your mouse pointer so you can place it, then selecting the next one on the list.

3.3.3 Placing the Tutorial Components

Using any method or combination of methods, make your layout look like the illustration below. You can also simply open the next design file in the project, GS2, which has already been set up this way.

Your design should look like this:



3.3.4 Placing Parts from the Database

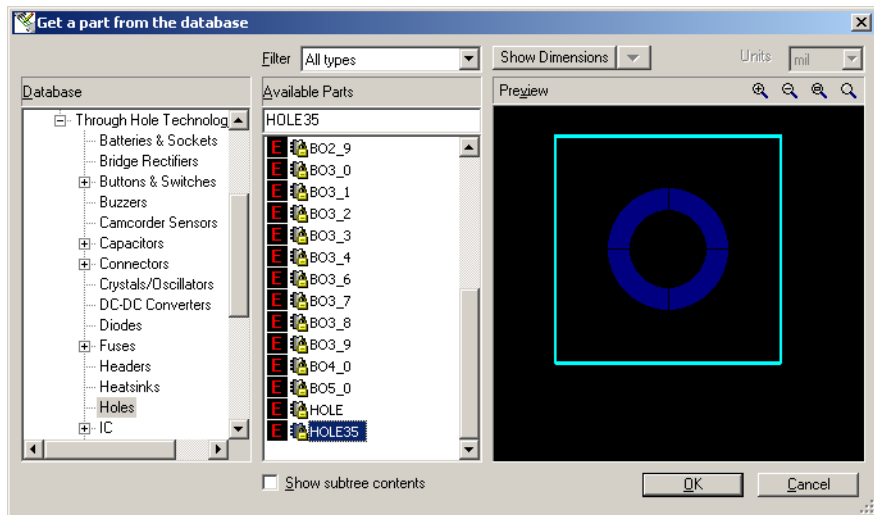
In addition to placing parts imported as part of your design file, you can place parts directly from the database. The following uses this method to place the mounting holes.

➤ To place parts from the database:



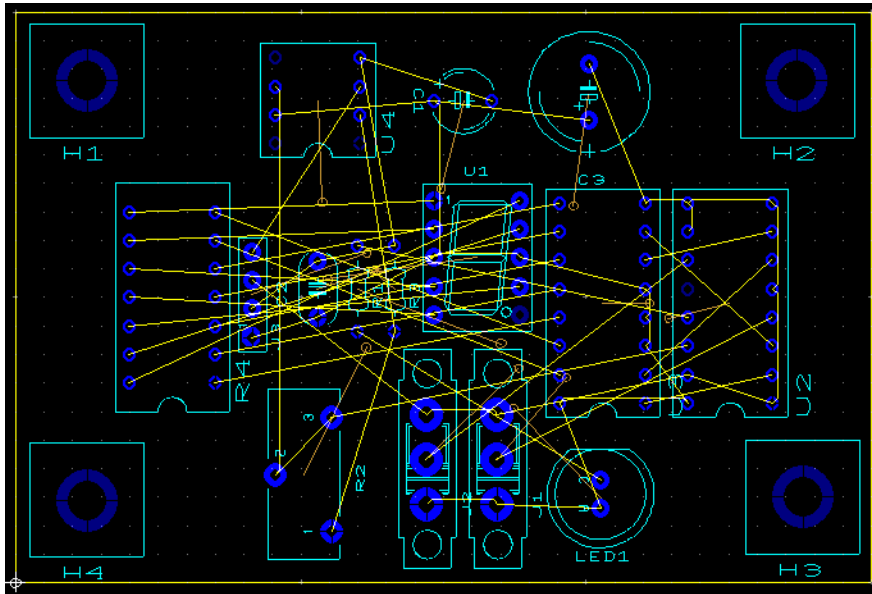
1. Choose **Place/From Database**. The **Get a part from the Database** dialog box opens:
2. In the **Database** panel, expand the **Through Hole Technology Parts** category and navigate to the **Holes** category. The parts appear in the **Available Parts** panel.

3. In the **Available Parts** panel, select the Hole35 part. The part displays in the **Preview** panel.



4. Click **OK**. The **Get a part from the database** dialog box disappears, and you are prompted to enter the **RefDes** and **Value**.
5. Enter the hole's reference designator (H1) and value (HOLE) and click **OK**.
6. Move the pointer over the board. The part is attached to the pointer.
7. When the hole is in position in the top-left corner, click to drop it on the board.
8. The **Enter Reference Designation for Component** dialog box reappears, with the reference designator automatically incremented to H2. Click **OK** to place the next mounting hole in the top right corner, and repeat to place H3 in the bottom right corner, and H4 in the

bottom left corner. Click **Cancel** to stop, and click **Cancel** again to close the **Get a part from the database** dialog box.



3.3.5 Moving Components

You can use the same methods for moving components as you do for placing them. To select a component already on the board, simply click on it. To specify the X/Y coordinates to which the selected part is to move, press the * key on the numeric keypad. Alternatively, in the **Parts** tab, select a placed component (indicated by a bright green light beside it) and drag it to a new location.



The component's label is a separate element from its footprint. When selecting a component on the board, be sure to select the whole component, not just the label.



Once a component is selected, you can also move it around on the board by pressing the arrow keys on your keyboard.

You can also select a group of components and move them together. To do this, you can do one of the following:

- hold down the SHIFT key and click on more than one component
- drag a box around several components.

All the selected components will move together when you drag the cursor.



These are temporary groups — once you select another component, the group connection is lost. To make a group that remains until you remove it, you can use the **Group Editor**. For details, refer to the *Ultiboard User Guide*.

Another option for moving components is to use the **Edit/Align** commands to align the edges of selected components or to space them relative to each other.

- Use the **Edit/Align** commands to align the mounting holes you just placed:
 1. Select H1 and hold down the SHIFT key to select H2.
 2. Choose **Edit/Align/Align Top**. If H2 was not originally placed exactly in line with H1, you will see it move.
 3. Click on an empty space on the board, then select H2 and H3.
 4. Choose **Edit/Align/Align Right**.
 5. Continue in this manner to align the bottoms of H3 and H4, and the left sides of H1 and H4.

3.4 Placing Traces

You have the following options for placing traces:

- manual trace
- follow-me trace
- connection machine trace.

A manual trace is placed exactly as you specify, even running through a component if that is the path you set out. A follow-me trace automatically draws a legal trace between the pins you select with your mouse movements — you can move from pin to pin, leaving a legal trace. A connection machine trace automatically joins two pins by the most efficient route, though you have the option of changing it.

As you place a trace, and before you click to fix it in place, you can always remove a segment by backing up over it. Each time you click while placing a manual trace, or each time a follow-me trace or connection machine trace changes direction, a separate segment of that trace is created. When performing operations on traces, be sure to select either the appropriate segment or, if you wish, the whole trace.

3.4.1 Placing a Manual Trace

You can continue with the design you have been working on, or open GS3. Be sure you are on the **Copper Top** layer before beginning.



If necessary, press F7 to show the whole design.

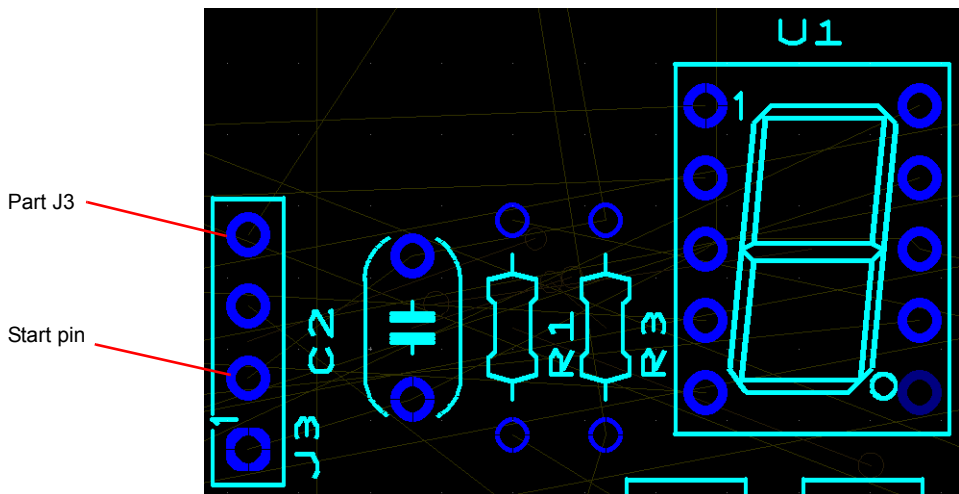
- To place a trace manually:

1. Choose **Place/Line**.



The **Line** command is used to create a line on any layer. The results differ depending on the layer selected. For example, if the selected layer is silkscreen, you will create a line on the silkscreen layer of the PCB. If the selected layer is a copper layer, then the “line” is actually a trace.

2. Locate J3, toward the left-hand part of the board. Find the start pin shown below:

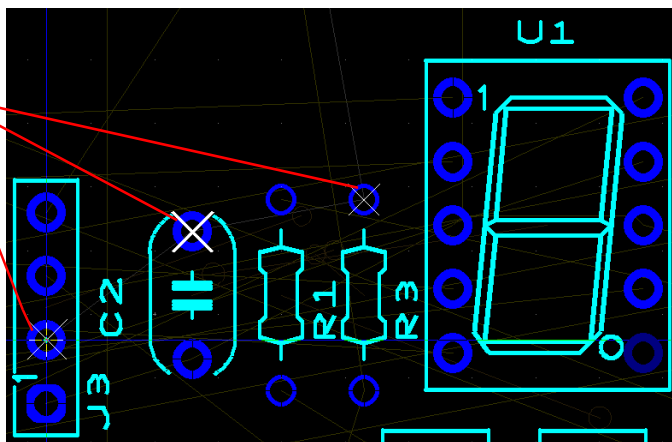


If you have trouble locating the component, use the **Find** function of the **Parts** tab. Select the component in the **Parts** tab, then click the **Find and select the part** button. The component is shown in the workspace. If necessary, zoom in further using F8.

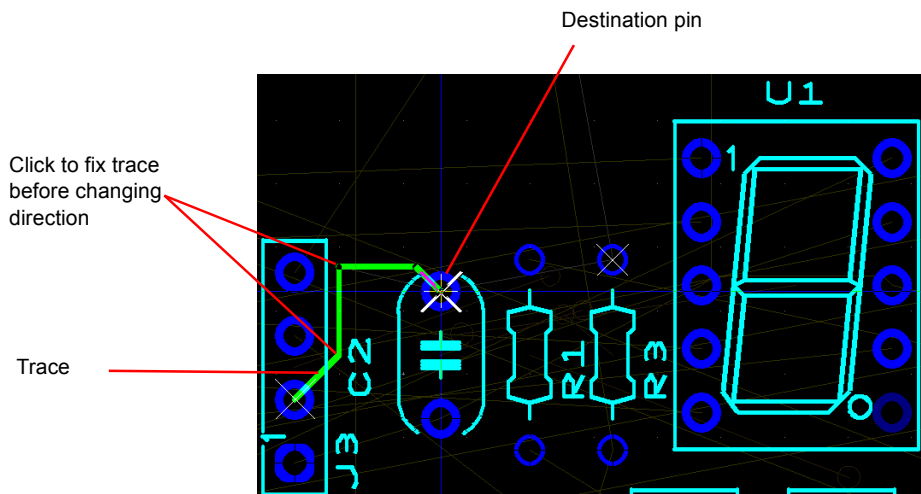
3. Click on the pin specified in the above step. Ultiboard highlights all the pins that are part of the same net as the pin you clicked on. (The color of the highlighting can be changed in

the **Colors** tab of the **Preferences** dialog box). This is how you know where to connect to pins to match the connectivity from your schematic.

Pins that are part of the same net are indicated with an X.



4. Move the cursor in any direction. A green line (the trace) is attached to the selected pin. Each time you click you anchor the trace segment.
5. Click on the destination pin.



6. Right-click to stop placing traces.

3.4.2 Placing a Follow-me Trace

➤ To place a follow-me trace:



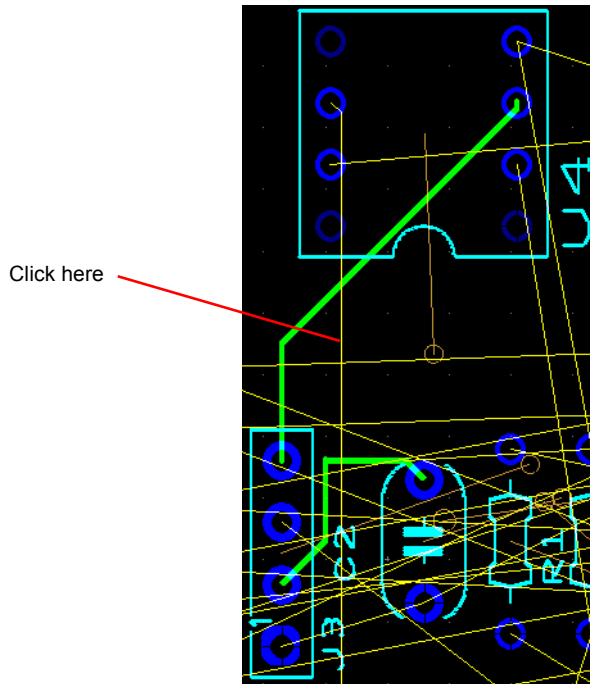
1. Choose **Place/Follow-me**.
2. Click on the top pin of J3.
3. Click on the second pin in the right column of U4.
4. Ultiboard draws the connection for you.



You do not need to click exactly on a pin — you can also start by clicking on a ratsnest line.

3.4.3 Placing a Connection Machine Trace

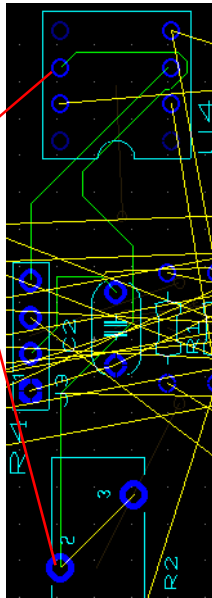
- To place a **Connection Machine** trace:
1. Choose **Place/Connection Machine**.
 2. Click on the segment of the ratsnest indicated below.



3. Move your cursor — Ultiboard suggests various traces routed around obstacles.

4. When you see the route you want, click to fix the trace. You don't have to click on the ratsnest or the destination pin.

New trace segments appear between these pins



5. Right-click to end trace placement.

3.5 Auto Part Placement

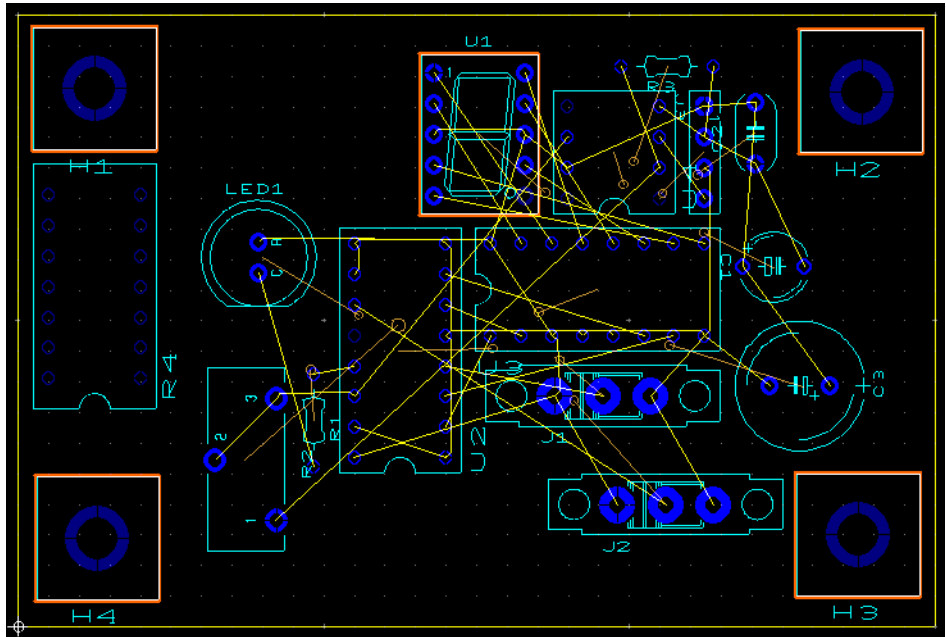
As well as placing parts as described earlier in this chapter, you can use Ultiboard's advanced automatic part placement functionality.



Before autoplacing parts, pre-place and lock any components that you do not wish to be moved during the autoplacement process. (The mounting holes and U1 in GS5 have been pre-placed and locked). For details on locking parts, refer to the *Ultiboard User Guide*.

- To autoplace the parts in `Getting Started.ewprj`:
 1. Open the GS5 design in Ultiboard.

2. Select **Autoroute/Start Autoplacement**. The parts are placed on the circuit board.

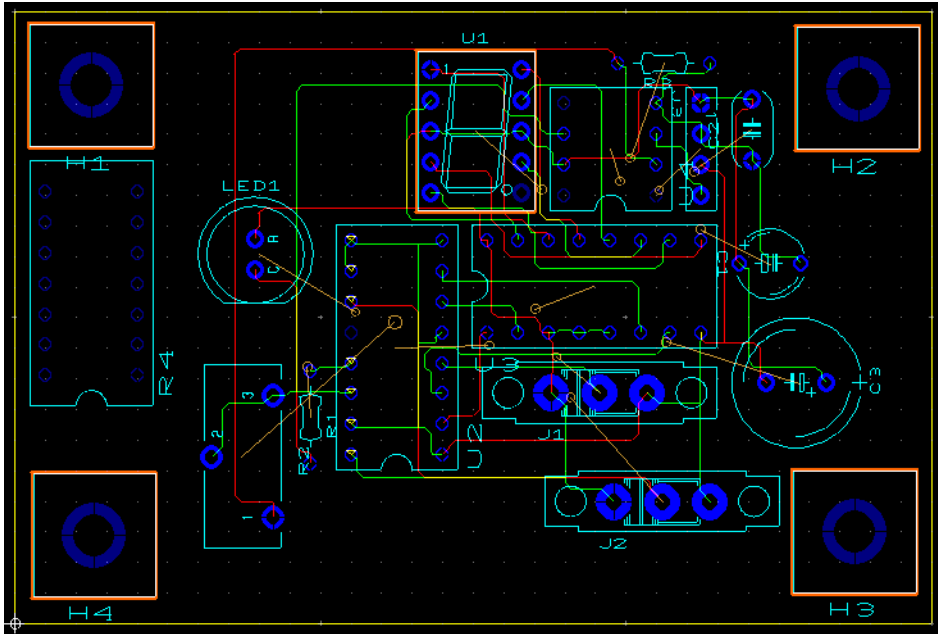


3.6 Autorouting Traces

You can place traces in Ultiboard using the methods described earlier in this chapter, or automatically route the traces as described below.

- To autoroute the traces in `Getting Started.ewprj`:
 1. Select **Autoroute/Start/Resume Autorouter**. The workspace goes to Autorouting Mode and trace autorouting begins.
 As autorouting proceeds, you will see traces being placed on the board. When autorouting is complete, Autorouting Mode closes and you are returned to the workspace.

2. Optionally, select **Autoroute/Start Optimization** to optimize the placement of the traces.



The autorouter can be stopped at any time and you can make manual changes as desired. When you restart the autorouter, it will continue with the changes you made. Remember to lock any traces that you have placed manually and do not wish to be moved by the autorouter.



Use the **Routing Options** dialog box to modify autoplacement and autorouting options. Refer to the *Ultiboard User Guide* for details.

3.7 Other Features

Ultiboard has many other useful features, including:

- **Net Bridges** — use to make connections between different nets (e.g., digital and analog grounds) without losing the properties of either net.
- **PCB Transmission Line Calculator** — use to calculate the following parameters for typical printed circuit board trace geometries:
 - Characteristic Impedance (Z_0)
 - Per unit length Capacitance (C_0)
 - Per unit length Inductance (L_0)
 - Propagation Delay (tpd)

- **PCB Differential Impedance Calculator** — performs calculations for two traces that carry signals that are exactly equal and opposite (a differential pair); use to calculate the following parameters for differential pairs:
 - Characteristic Impedance (Z_0)
 - Per unit length Capacitance (C_0)
 - Per unit length Inductance (L_0)
 - Propagation Delay (tpd)
 - Differential Impedance (Z_{diff})

3.8 Preparing for Manufacturing/Assembly

Ultiboard can produce many different output formats to support your production and manufacturing needs. This section explains the functions performed to output your board for production and documentation purposes.

3.8.1 Cleaning up the Board

Before sending the board for manufacturing, you should clean up any open trace ends (trace segments that do not have any terminating connections in the design) and unused vias that have been left on the board.

- To delete open trace ends, make sure the GS4 design is open and choose **Edit/Copper Delete/Open Trace Ends**. This deletes all open trace ends in the design.
- To delete any unused vias, make sure the design is open and choose **Design/Clean Unused Vias** to delete all vias that do not have any trace segments or copper areas connected to them.

3.8.2 Adding Comments



Comments can be used to show engineering change orders, to facilitate collaborative work among team members, or to allow background information to be attached to a design.

You can "pin" a comment to the workspace, or directly to a component. When a component with an attached comment is moved, the comment also moves.

For details, refer to the *Ultiboard User Guide*.

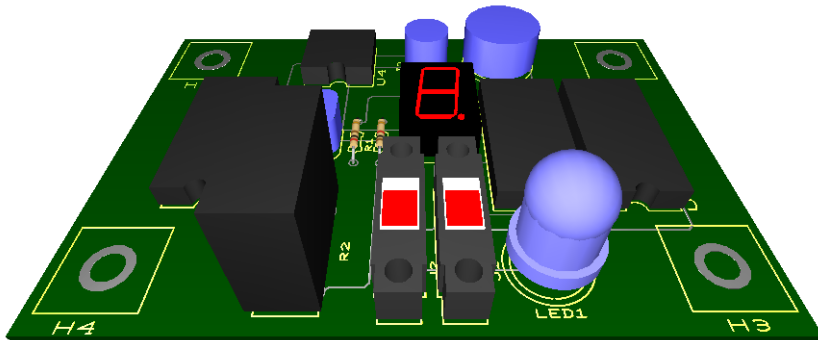
3.8.3 Exporting a File

Exporting a file refers to producing an output from Ultiboard in a format that can be understood by the board manufacturer. An exported file contains complete information describing how a finished board is to be manufactured. Files that can be exported include Gerber RS-274X and RS-274D files.

For complete details, refer to the *Ultiboard User Guide*.

3.9 Viewing Designs in 3D

Ultiboard lets you see what the board looks like in three dimensions at any time during the design. For complete details, refer to the *Ultiboard User Guide*.



You can use the **Internal View** to look between the layers of a multi-layer PCB. For details, refer to the *Ultiboard User Guide*.

Chapter 4

Multisim MCU Module Tutorial

The tutorial in this chapter leads you through the process of simulating and debugging a circuit that contains a microcontroller.

The following are described in this chapter.

Subject	Page No.
Overview	4-1
About the Tutorial	4-2
Understanding the Assembly Program	4-4
Advanced Features	4-7
Debug View Overview	4-7
Adding a Breakpoint	4-9
Break and Step	4-10
Break and Step Out	4-12
Break and Step Into	4-12
Break and Step Over	4-12
Run to Cursor	4-12

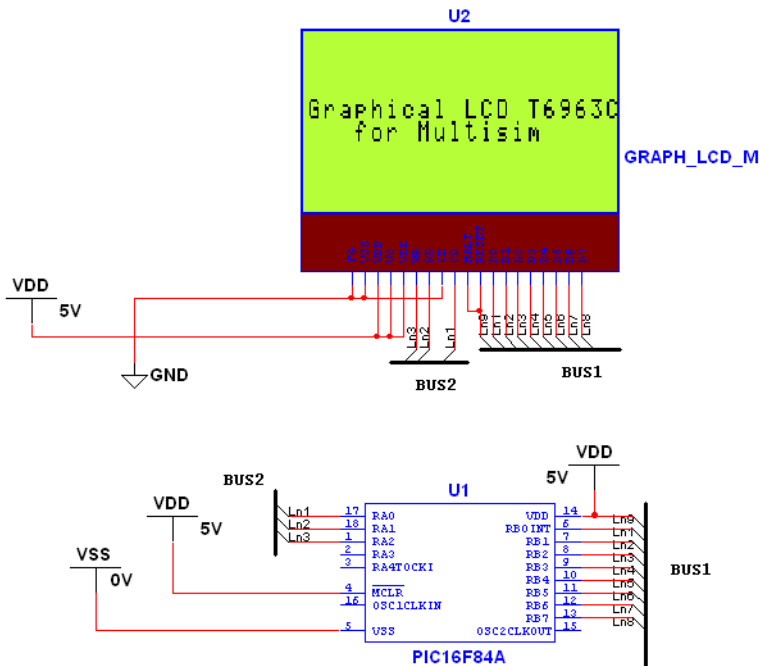
4.1 Overview

The files used for this tutorial install with your NI Circuit Design Suite software at
...\\samples\\Getting Started.

This tutorial uses Getting Started MCU.ms10, which accesses the contents of folder LCDWorkspace as required.

The LCD Graphical Display circuit example demonstrates the use of a PIC microcontroller to control a graphical LCD display component in Multisim based on a combination of the Toshiba T6963C controller and an external display RAM. To control the LCD display, the microcontroller sends signals to the LCD through the LCD's data and control lines. A

software program written for the microcontroller determines the logic behind setting the lines on its pins to high or low to send commands and data to the LCD display.



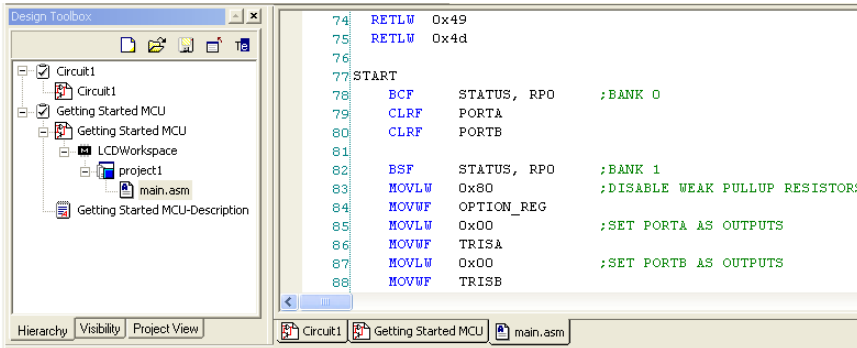
4.2 About the Tutorial

The data lines of LCD U2 are connected to pins RB0 – RB7 on microcontroller U1. The control lines of the LCD are connected to RA0 - RA2 on the microcontroller. The MCU U1 communicates with the LCD U2 via these wires. Data is sent to U2 in parallel and signals on the control lines determine the timing and type of data being sent (i.e., address or data).

The LCD Graphical Display can operate in three modes: text mode, graphical mode and a combination text and graphical mode. This example demonstrates the controlling of the LCD Graphical Display in a combination text and graphical mode. The software that the MCU runs is contained in an MCU workspace that displays in the **Design Toolbox** as LCDWorkspace. The workspace contains one project project1 that consists of a single source code file main.asm.

➤ To view the file:

1. Double-click on `main.asm` in the **Design Toolbox**. A tab appears in the schematic capture workspace called `main.asm` that displays the assembly program.



➤ To display the line numbers, select **MCU/Show Line Numbers**.

➤ To run this circuit:



1. Select **Simulate/Run**. If you did not build your program beforehand, a dialog box displays stating that the configuration is out of date and asks if you would like to build it. Click **Yes**. The results of the build display in the **Results** tab of the **Spreadsheet View**. If there are no errors or warnings, the program built successfully. (The sample program should not contain any errors.)

The program displays the line “Graphical LCD T6963C for Multisim” characters in text mode; the LCD then switches to graphical mode and draws an inverted “V” dot-by-dot on top of the text.

Once the lines are drawn, the text scrolls right and then left. This is achieved by moving the start address of the text buffer of the LCD display. This also demonstrates that there are two buffers in the LCD, one for storing graphics and another for storing text. Other features of the LCD such as text flashing and erasing of characters are also demonstrated.

The LCD display program continues to cycle through each of these effects.



➤ To stop the simulation, select **Simulate/Stop**.

4.2.1 Understanding the Assembly Program

Constants and data

To make the program easier to understand, the LCD display commands and temporary buffers for storing addresses and data in the MCU are predefined in constants at the start of the program:

```
DATA_BUFFER      EQU      0x20
DATA_BUFFER2     EQU      0x21
CMD_BUFFER       EQU      0x22
REF_BUFFER       EQU      0x24

ADDR_INDEX       EQU      0x25      ;STARTING ADDRESS IN EEPROM
ADDR_L           EQU      0x26      ;STARTING ADDRESS L
ADDR_H           EQU      0x27      ;STARTING ADDRESS H
COUNTER_INDEX    EQU      0x29      ;COUNTER
BIT_INDEX        EQU      0x2A      ;BIT INDEX

CMD_SET_CURSOR   EQU      21H      ;SET CURSOR
CMD_TXHOME       EQU      40H      ;SET TXT HM ADD
CMD_TXAREA       EQU      41H      ;SET TXT AREA
CMD_GRHOME       EQU      42H      ;SET GR HM ADD
CMD_GRAREA       EQU      43H      ;SET GR AREA
CMD_OFFSET       EQU      22H      ;SET OFFSET ADD
CMD_ADPSSET      EQU      24H      ;SET ADD PTR
CMD_SETDATA_INC  EQU      0C0H      ;WRITE DATA AND INCREASE ADP
CMD_AWRON        EQU      0B0H      ;SET AUTO WRITE MODE
CMD_AWROFF       EQU      0B2H      ;RESET AUTO WRITE MODE
```

The text to be displayed on the LCD display is stored in data tables for some microcontrollers, but there is no PIC assembly instruction that allows you to directly address a data value in the program memory space. Instead, you can load literal values into the W register so you can write a routine that returns a value in your string based on an index. The RETLW instruction loads a constant value into the W register and executes a RETURN in one instruction.

The TXPRT routine retrieves the text data to be displayed on the LCD display. The character codes for the LCD display are defined in the T6963C controller reference manual (e.g. 0x27 is the code for the letter “G”, 0x52 for “r”, and so on:

```
; DATA
DATA_NUM      EQU      23H
TXPRT          ; Text data "Grapical LCD T6963C   for Multisim"
    ADDWF     PCL, 1
    RETLW     0x27
    RETLW     0x52
    RETLW     0x41
    RETLW     0x50
    RETLW     0x48
    RETLW     0x49
    RETLW     0x43
    RETLW     0x41
    RETLW     0x4c
    RETLW     0x00
    RETLW     0x2C
    ...
```

Initialization

The initialization code begins at the `START` label as shown in the excerpt below. The pins in the microcontroller are set up as output pins, and the values are reset. The LCD display component is initialized by the microcontroller and set to graphical and text mode. The home addresses for the internal graphical and text buffers in the LCD display component are set to `0x0000` and `0x2941` respectively, which determines where on the display the LCD starts to display the buffer data. Finally, the control signals are set up for the proper read/write operation on the LCD display.

```
START
    BCF     STATUS, RPO           ;BANK 0
    CLRF    PORTA
    CLRF    PORTB

    BSF     STATUS, RPO           ;BANK 1
    MOVLW   0x80                  ;DISABLE WEAK PULLUP RESISTORS
    MOVWF   OPTION_REG
    MOVLW   0x00                  ;SET PORTA AS OUTPUTS
    MOVWF   TRISA
    MOVLW   0x00                  ;SET PORTB AS OUTPUTS
    MOVWF   TRISB

    BCF     STATUS, RPO           ;BANK 0
    MOVLW   0x0F                  ; 1111 no commands ready
    MOVWF   PORTA

;1 SET DISPLAY MODE to GRAPH + TEXT mode, cursor off
    MOVLW   0x9C
    MOVWF   CMD_BUFFER
    CALL    CMD
...
```

Drawing Text and Graphics

The rest of the program sends commands to the LCD graphical display via the control lines through MCU pins RA0 to RA2 and data through the data lines:

```
;5 write_string
    MOVLW    0x7D
    MOVWF    DATA_BUFFER
    MOVLW    0x29
    MOVWF    DATA_BUFFER2    ; external CG start at: 1400h
    CALL     DT2
    MOVLW    CMD_ADPSET
    MOVWF    CMD_BUFFER
    CALL     CMD

    MOVLW    CMD_AWRON
    MOVWF    CMD_BUFFER
    CALL     CMD

    MOVLW    0x00                ; Initial the counter
    MOVWF    ADDR_INDEX

LOOP_READ_DATA2
    MOVF     ADDR_INDEX,0    ; STARTING data ADDRESS
    CALL     TXPRT

    MOVWF    DATA_BUFFER    ; LOAD CHAR data TO W
    CALL     ADT

    INCF     ADDR_INDEX,1

    MOVF     ADDR_INDEX, 0
    SUBLW    DATA_NUM        ; 35 chars
    BTFSS    STATUS, Z
    GOTO     LOOP_READ_DATA2

    MOVLW    CMD_AWROFF
    MOVWF    CMD_BUFFER
    CALL     CMD

...
```

For example, the above excerpt from the main loop in the program sends the characters defined in the TXPRT subroutine to be displayed in text mode on the graphical LCD.

The following sets the LCD to auto write mode:

```
MOVLW    CMD_AWRON
MOVWF    CMD_BUFFER
CALL     CMD
```

At this point, the program starts counting, and executes through the loop LOOP_READ_DATA2 35 times. This loop calls TXPRT to retrieve the text data and load it into the W register. It then calls to the subroutine ADT, which calls SEND_DATA, which writes the values in the W register to port B, to be sent to the data lines of the LCD display. Once the data is sent, the proper value on port A of the microcontroller is sent to the control pins of the LCD display to let it know that the data is ready to be read. The subroutines all return at the end to the instruction just after the call to them and the same thing happens until all 35 characters have been transmitted. The final three instructions in the excerpt turn off the auto write mode in the LCD display after exiting the loop:

```

MOVLW  CMD_AWROFF
MOVWF  CMD_BUFFER
CALL   CMD

```

The next few instructions draw the horizontal and sloped lines in graphical mode:

```

;6 draw wave once
MOVF  ADDR_L, 0
BTFS  STATUS, Z
CALL  DRAW_WAVE

```

4.3 Advanced Features

This section provides a step-by-step walkthrough of the Multisim MCU module's debugging features. It is important to follow the steps exactly as scripted, otherwise, the descriptions will no longer apply. Once you understand the breakpoint and single stepping features you can explore the possibilities of advanced MCU debugging.

4.3.1 Debug View Overview

To write a program for a microcontroller either in C or assembly, you create source code files (.asm, .inc, .c, .h) as part of the MCU workspace, which can in turn be edited in the source code view.

➤ To access the source code view:

1. Double-click on the file item (e.g., main.asm) shown in the MCU workspace hierarchy in the **Design Toolbox**.

During simulation, additional debugging information displays to help you understand what is happening inside the MCU. For example, you can switch between viewing events happening in the high level source and at the assembly instruction level which also displays the actual opcodes for each instruction that are being executed by the MCU.

The source code view is not capable of displaying all this extra information. Instead, each MCU component in the circuit design has its own **Debug View** that displays debugging information.

➤ To access the **Debug View**:

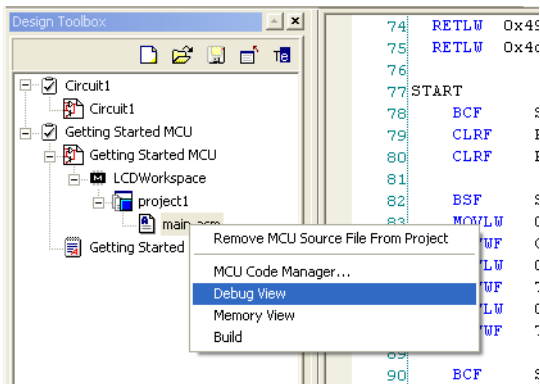
1. Select **MCU/MCU PIC 16F84A U1/Build**.

Note The **Debug View** is available only after you have successfully built your code, so the preceding step is only necessary once.

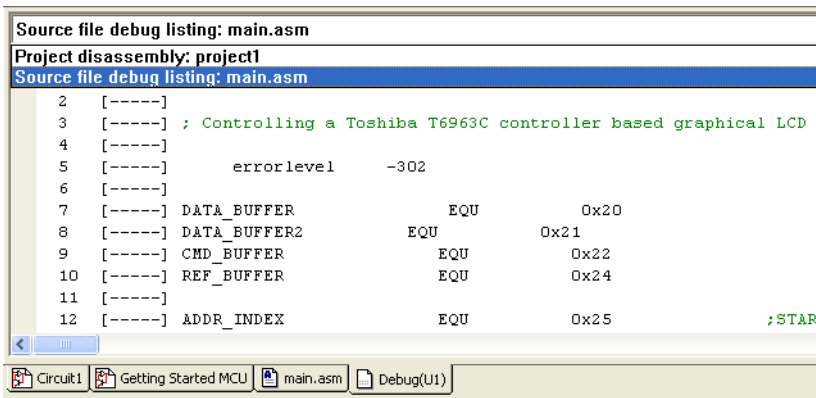
2. Select **MCU/MCU PIC 16F84A U1/Debug View**.

Or

Use the right-click context menu on an item in the MCU workspace of the **Design Toolbox**.



Another tab opens in the schematic capture workspace called **Debug(<reference designator of MCU>)**, in this case **Debug(U1)**.



Use the drop-down list at the top of the **Debug View** to select between the disassembly instructions generated internally by Multisim or the listing file generated by the assembler or compiler (the format of the listing file is dependant on the tool that you choose to build your code).

In the LCD graphical display example, the code was written in assembly and built by the Microchip assembly tools. The Microchip assembler generates a listing file (.lst) that contains all of the opcodes generated for each assembly instruction. The debug listing view displays information from this listing file. Multisim generates the disassembly format using its internal disassembler to disassemble the opcode instructions into assembly instructions.

This format is not necessary for this example since the debug listing contains all of the information needed. In cases where an MCU project loads only the machine code (.hex) file, the disassembly view shows the disassembled opcode instructions so that you can see what's happening in the MCU. Since no listing file for MCU projects of this type is available, the disassembly view is very useful.

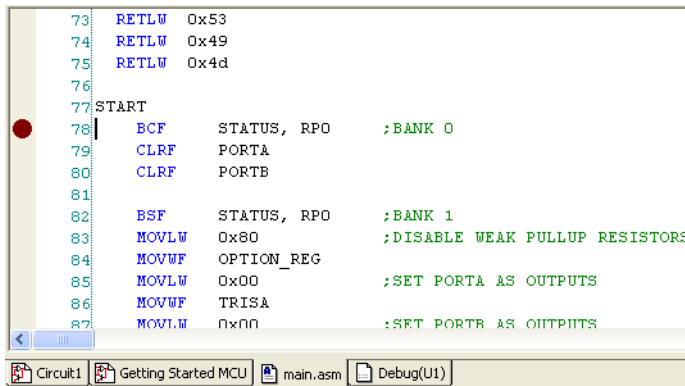
4.3.2 Adding a Breakpoint

You can add breakpoints in the source code view when simulation has stopped, as well as during simulation. You can add breakpoints to a microcontroller in two ways.

One way is to add them in the source code view. In this example, the `main.asm` tab in the schematic capture workspace is the only source code view available.

Note If your MCU design contains more than one file, there will be a source code view for each of your source code files.

You can also set a breakpoint in the **Debug View** window. You can set breakpoints in the disassembly view or the debug listing view, but for this example, you will only use the debug listing view.



- To add a breakpoint in the source code view:
 1. Build the MCU workspace.
 2. Open the **Debug View** for U1.
 3. Double-click on `main.asm` in the **Design Toolbox**.
 4. Scroll to the line just below the `START` label: `BCF STATUS, RPO`.

5. Double-click on the first (grey) column on the left side of the `main.asm` window next to the line `BCF STATUS, RP0`. A red circle appears at that location indicating that a breakpoint has been set at that line.



6. Select **Simulate/Run**. The simulation automatically pauses at the breakpoint that you have just set. The **Debug View** automatically jumps into focus with a yellow arrow showing where the MCU program execution is paused.

➤ To remove the breakpoint:

1. Double-click on the breakpoint in the **Debug View** or the `main.asm` source code view.

Or



Select **MCU/Remove all breakpoints** to remove all breakpoints.

Note You can add and remove breakpoints in the **Debug View** in the same manner as the source code view.

4.3.3 Break and Step



1. Select **MCU/Remove all breakpoints** to remove all breakpoints.
2. Go to the circuit design view (the **Getting Started MCU** tab) and select **Simulate/Run**. The words “Graphical LCD T6963CC for Multisim” start to display on the graphical LCD component.



3. Select **Simulate/Pause**.

- Go to the **Debug View** for U1 and notice that the line of code in the debug listing view where the MCU has stopped its execution is indicated by a yellow arrow in the left-most column.

MCU Memory View :: U1 :: GET

Registers

Name	Hex	Bit-7	Bit-6
Working register	00	0	0

Memory View

Name	Address	Hex	Bit-7	Bit-6
INDF	00	00	0	0
TMR0	01	1A	0	0

IRAM

EEPROM

IRAM

PC: 0193

ROM	00	01	02	03
0000	2834	0000	0000	00
0008	0000	0000	0000	00
0010	0782	3427	3452	34
0018	3441	344C	3400	34
0020	3416	3419	3416	34

Source file debug listing: main.asm

```

599 [-----]
600 [-----] READ_DATA
601 [0018A]    CALL SET_PORT_B_INPUT
602 [0018B]    MOVLW    0x04                ; 0100 r
603 [0018C]    MOVWF    PORTA
604 [0018D]    MOVF     PORTB,0              ; GET
605 [0018E]    MOVWF    DATA_BUFFER
606 [0018F]    RETURN
607 [-----]
608 [-----]
609 [-----] SEND_DATA
610 [00190]    CALL     SET_PORT_B_OUTPUT
611 [00191]    MOVF     DATA_BUFFER, 0      ;output d
612 [00192]    MOVWF    PORTB
613 [00193]    MOVLW    0x0A                ; 1010 w
614 [00194]    MOVWF    PORTA
615 [00195]    BSF      PORTA, 2
616 [00196]    RETURN
617 [-----]
618 [-----] SEND_CMD
619 [00197]    CALL     SET_PORT_B_OUTPUT
620 [00198]    MOVF     CMD_BUFFER, 0        ;output cm
621 [00199]    MOVWF    PORTB
622 [0019A]    MOVLW    0x0B                ; 1011 w
623 [0019B]    MOVWF    PORTA
624 [0019C]    BSF      PORTA, 2
625 [0019D]    RETURN
626 [-----]

```

- Select **MCU/MCU PIC16F84A U1/Memory View** to view the current state of the memory inside the microcontroller U1. Notice that the value of the program counter PC in the **IRAM** section is one higher than the address value of the line the yellow arrow is pointing to. (In the above figure, the address in the **Debug View** is 192 and the PC value in the **Memory View** is 193.)

Note If the MCU has not finished executing the current command when you pause the simulation, the value in the program counter will be the same as the address value.

You can also look at the other sections of the **Memory View** to see the values inside the other parts of memory in the microcontroller.



- Click the **Step into** button in the **Simulation** tool bar.
- The current instruction is executed and the simulation pauses at the next instruction.
- Select **Simulate/Stop**.



4.3.4 Break and Step Out



1. Place a breakpoint in the `SEND_DATA` subroutine at `MOVWF PORTB`.
2. Select **Simulate/Run**. The simulation pauses at the breakpoint.
3. Click the **Step out** button in the **Simulation** toolbar to step out of the `SEND_DATA` subroutine.
4. The simulation executes all of the remaining instructions in the `SEND_DATA` subroutine and pauses at the first instruction after the call to the `SEND_DATA` subroutine.

4.3.5 Break and Step Into



1. Select **MCU/Remove all breakpoints**.
2. Place a breakpoint at the call to `SEND_DATA` where you had just stepped out of just above the yellow arrow.
3. Select **Simulate/Run**. The simulation pauses at breakpoint that you just placed.
4. Click the **Step Into** button on the **Simulation** toolbar. The simulation pauses inside the `SEND_DATA` subroutine.

4.3.6 Break and Step Over



1. Select **Simulate/Run**. The simulation pauses at the same breakpoint that you set previously at the call to the subroutine `SEND_DATA`.
2. Click the **Step Over** button on the **Simulation** toolbar. The entire `SEND_DATA` subroutine is executed and the simulation pauses at the instruction after the `CALL SEND_DATA` instruction.

4.3.7 Run to Cursor



1. Select **MCU/Remove all breakpoints**.
2. Click on a line inside the `SEND_DATA` subroutine since we know that this subroutine will be called again to send data to the LCD display.
3. Click the **Run to Cursor** button in the **Simulation** toolbar. The simulation runs until the MCU hits the instruction that you clicked on inside the `SEND_DATA` subroutine. It then pauses and places the yellow arrow next to that line.

Appendix A - Support and Services

A.1 Technical Support and Professional Services

Visit the following sections of the National Instruments web site at ni.com for technical support and professional services:

- **Support** — online technical support resources at ni.com/support include the following:
 - **Self-Help Resources** — For answers and solutions, visit the award-winning National Instruments web site for software drivers and updates, a searchable KnowledgeBase, product manuals, step-by-step troubleshooting wizards, thousands of example programs, tutorials, application notes, instrument drivers, and so on.
 - **Free Technical Support** — All registered users receive free Basic Service, which includes access to hundreds of Application Engineers worldwide in the NI Discussion Forums at ni.com/forums. National Instruments Application Engineers make sure every question receives an answer. For information about other technical support options in your area, visit ni.com/services or contact your local office at ni.com/contact.
- **Training and Certification** — Visit ni.com/training for self-paced training, eLearning, virtual classrooms, interactive CDs, and Certification program information. You also can register for instructor-led, hands-on courses at locations around the world.
- **System Integration** — If you have time constraints, limited in-house technical resources, or other project challenges, National Instruments Alliance Partner members can help. To learn more, call your local NI office or visit ni.com/alliance.

If you searched ni.com and could not find the answers you need, contact your local office or NI corporate headquarters. Phone numbers for our worldwide offices are listed in the front of this manual. You can also visit the Worldwide Offices section of ni.com/niglobal to access the branch office web sites, which provide up-to-date contact information, support phone numbers, email addresses, and current events.

Index

Numerics

3D designs in Ultiboard 3-20

A

advanced MCU debugging 4-7

analysis 2-10

assembly program 4-4

autoplacement 3-16

autorouting 3-17

B

bill of materials 2-13

board clean-up 3-19

board outline 3-3

BOM 2-13

break and step 4-10

break and step into 4-12

break and step out 4-12

break and step over 4-12

breakpoint 4-9

C

comments 3-19

connection machine trace 3-15

D

dragging components 3-6, 3-7

E

exporting files from Ultiboard 3-20

F

follow-me trace 3-14

G

grapher 2-12

M

manual trace 3-12

manufacturing/assembly 3-19

MCU debugging overview 4-7

MCU tutorial 4-2

MCU tutorial overview 4-1

moving components in Ultiboard 3-10

Multisim tutorial overview 2-1

O

opening Multisim files 2-3

opening Ultiboard tutorial 3-2

P

placing components in Multisim 2-3

placing components in Ultiboard 3-5, 3-7

placing traces in Ultiboard 3-11

placing Ultiboard dB parts 3-8

postprocessor 2-12

products 1-1

R

reports 2-12

run to cursor 4-12

S

saving Multisim files 2-3

schematic capture 2-2

simulation 2-9

T

tutorial descriptions 1-1

two-pinned passives

dropping directly onto a wire 2-6

V

virtual instruments 2-9

W

wiring components in Multisim 2-7