Creation of PSoC First Touch Kit and Arduino Shield Component Libraries using DipTrace

Brett Donlon

November 11, 2011

Design Team 1

Abstract

This application note serves as a reference for DipTrace Schematic and PCB Design software users who wish to create printed circuit boards containing PSoC First Touch Kit boards and/or Arduino shields. This document will explain how to create a component library for both the First Touch Kit and Arduino shields as well as provide general guidance on the use of DipTrace's Component Editor.

Introduction

DipTrace is electronic design automation software that can be used to create schematic diagrams and lay out printed circuit boards (PCBs). DipTrace is easy to use and is able to create professional looking PCBs. Full versions of the software are available on their website, diptrace.com, as well as a 30-day full featured trial and a freeware version limited to 300 pins. DipTrace covers every step in the creation of a printed circuit board, from the creation of the components and patterns, to the schematic capture and board layout.

DipTrace offers standard component libraries which contain over 100,000 parts. Discrete components and many frequently used electronic components from well-known companies are included in the long list of libraries.

Objective

Despite the fact that there is an existing Cypress component library, there is no existing PSoC First Touch Kit component. In fact there are no components available in DipTrace for any of the PSoC development kits. There is no existing component library for Arduino either. In this application note, creation of these components using DipTrace's Component Editor will be discussed in detail.

Measurements

PSoC First Touch Kit

Prior to using the DipTrace Component Editor, measurements of your component need to be taken. Most important are the spacing between pins and spacing between pin arrays if there are multiple pin arrays. The physical dimensions of the kit are something that should be taken into account when completing the actual layout of the printed circuit board but is of little importance in the component creation stage. The PSoC First Touch Kit is shown in Figures 1 and 2 (Top and Bottom Views).

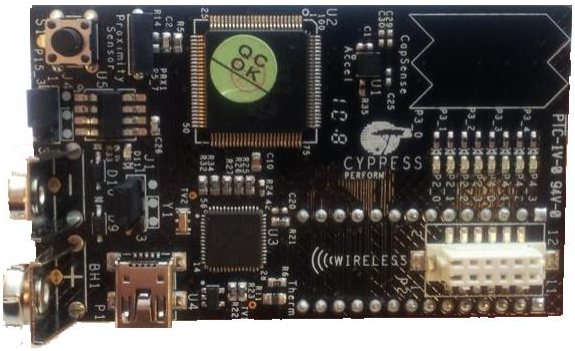


Figure 1: PSoC First Touch Kit (Top View)

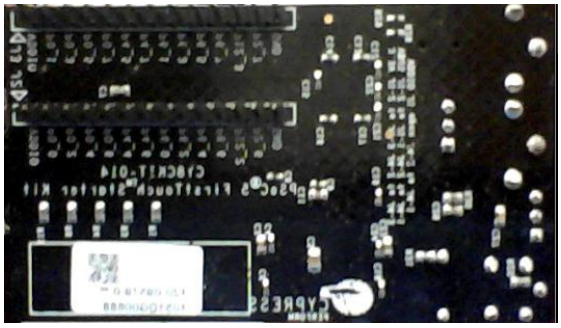


Figure 2: PSoC First Touch Kit (Bottom View)

The PSoC First Touch Kit has 28 General Purpose I/O (GPIO) pins. These pins are arranged in two separate pin arrays, each containing 14 pins. The individual pin spacing for both of these pin arrays is 0.1 inches or 100 mils (Units will be discussed in a later section). These two pin arrays are spaced 0.5 inches from each other. The figure below illustrates the dimensions for the First Touch kit pin spacing.

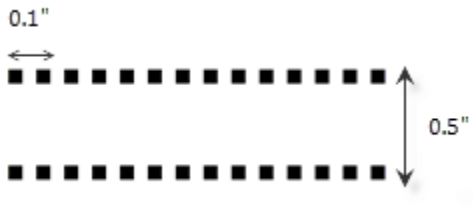


Figure 3: PSoC First Touch Kit Pin Spacing

Arduino Shields

Arduino users form a very large community, spanning many different applications and projects. There have been many different shields created and many different layouts exist. This application will cover the creation of a component that will fit the general layout of Arduino Uno and Duemilanove boards. Shields such as the Arduino Ethernet Shield use this layout. Figure 4 below of the Ethernet shield illustrates this layout.

Figure 4: Arduino Ethernet Shield

There are five headers: two 6-pin headers, two 8-pin headers and a 2x3 header which is located on the underside of the board. The solder joints for this header can be seen directly above the micro SD card slot on the right side of the board.

The individual pin spacing for all the pins on the board is 0.1 inches. The spacing between the two 6-pin headers is 0.2 inches. The spacing between the two 8-pin headers is 0.16 inches. The 2x3 header also known as the In-Circuit Serial Programming (ICSP) header's first row of three pins is perfectly in line with the first pin of both the 6-pin and 8-pin header at the right side of the board (when referring to Figure 4). The 2x3 header is spaced 0.89 inches from the 6-pin header and 0.81 inches from the 8-pin header. For a detailed illustration of the pin spacing for the Arduino shield, refer to Figure 5 below.

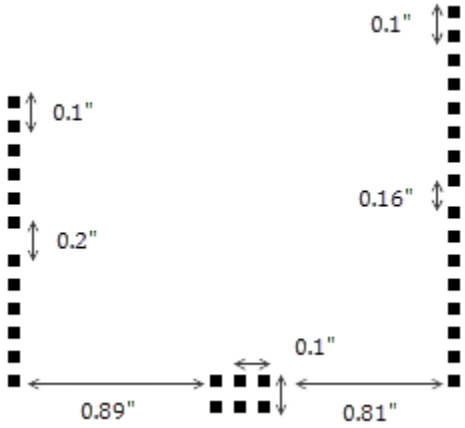


Figure 5: Arduino Shield Pin Spacing

Component Editor Overview

Once the measurements of the components are taken, it is necessary to become familiar with DipTrace's Component Editor. First open up the Component Editor (this Application Note was done using the 30 day full featured trial but this note can be used for the 300 pin limit freeware as well because neither component has more than 300 pins). Upon opening the software, the screen showing should be what is seen in Figure 6 on the following page.

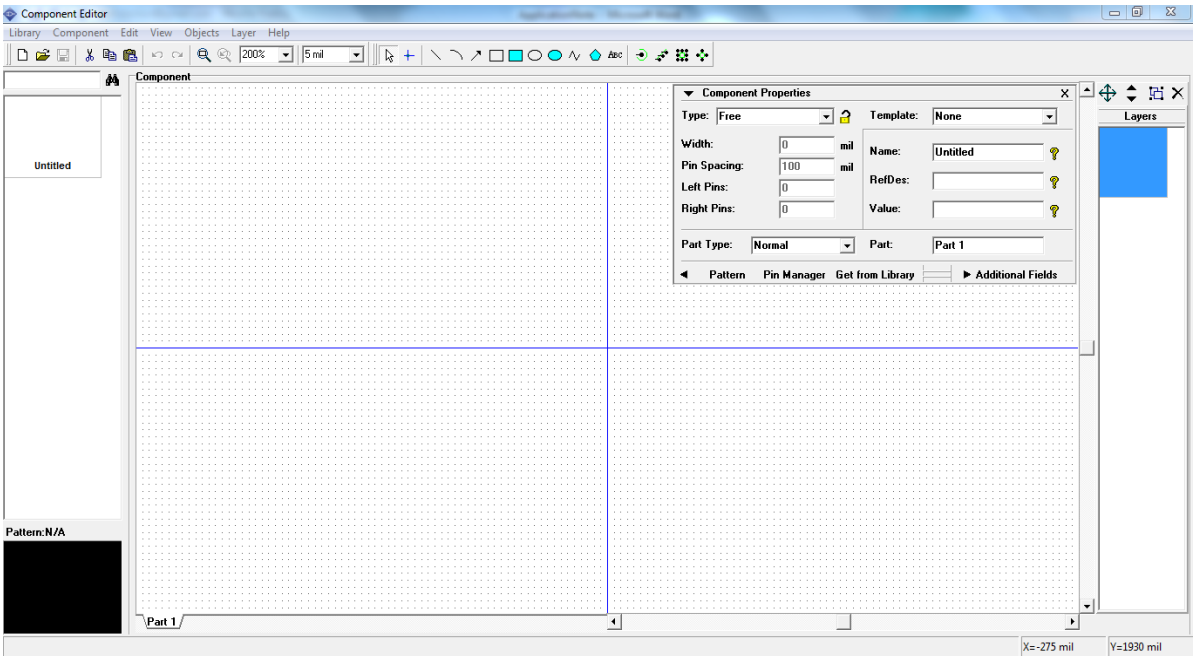


Figure 6: Initial Screen for DipTrace Component Editor

Navigation

DipTrace Component Editor can be navigated through with ease using the toolbars and drop down menus seen in Figure 6.

Menus

Below is a brief overview of each drop down menu.

Library menu: Open, save, import, export, and create new files.

Component menu: Add components and parts to the current library. These components can be taken from existing libraries or be created. Pin manager can also be accessed where component pin settings such as name, number, length, type, and electrical properties, can be configured.

Edit menu: Cut, copy, paste, undo, redo, etc. Lock, rotate, and flip objects.

View menu: Select which toolbars to display, toggle grid on/off, show/hide pin numbers, display origin, and choose units for component design.

Objects menu: Place shapes, pins, and pin arrays.

Layer menu: Move, delete, merge, and change layers.

Help menu: Contains a tutorial in case troubles arise during PCB layout.

Toolbars

In addition to the menus, component creation can also be guided by the toolbars available. Three toolbars exist in DipTrace Component Editor: Component Properties, Standard, and Objects. Following is a brief description of each.

Component Properties Toolbar: Type, pin spacing, number of pins, name, value, and reference designation can be specified using the Component Properties toolbar.

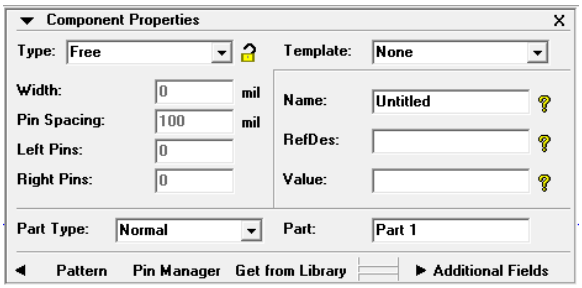


Figure 7: Component Properties Toolbar

Standard Toolbar: Standard operations such as open, save, cut, copy and paste can be performed using the Standard toolbar. Grid size and zoom properties can also be controlled.

Figure 8: Standard Toolbar

Objects Toolbar: The objects toolbar provides most of the same functions as the Objects menu.

Figure 9: Objects Toolbar

Organization

DipTrace Component Editor allows for the creation of several components within a particular component library. Furthermore, components can be broken down into individual parts. The components contained in a library are shown on the left side of the screen when using DipTrace Component Editor. At the bottom of the screen, there is a tab for each individual part that may be used to form an entire component. The advantages of creating individual parts within a component is the ability to display the reference designation, value, and pin names for each individual part that makes up the component.

Component Creation

Initial Setup

With measurements taken and knowledge on the software, component creation can begin. With the initial screen open, first the units and grid size need to be specified. The units used in this application note will be mils (1 inch = 1000 mil). In the View menu, select units, and then mil. Next the grid size will be set. Again go to the View menu, then grid size and select 10 mil. If it is not present, a custom grid size can be created by selecting "Customize Grid" in the View menu.

PSoC First Touch Kit Component

As mentioned earlier, there exists a Cypress library in DipTrace. This component can be saved either in the existing library or in a library of its own. To save to the Cypress library, click Open either in the Library drop down menu or on the Standard toolbar and navigate to the directory containing the Cypress component library. It is important to recognize the difference between component and pattern libraries. The file extension of a component library is *.eli while a pattern library has an extension of *.lib. Once the cypress.eli component library is opened, go to the Component drop down menu and click "Add New to Library." A new blank workspace will appear on the screen. A new component library can be created by simply going to Library>New.

In the Component Properties toolbox, the type of the component must be specified. Choices include Free, 2 sides, IC-2 sides, and IC-4 sides. The PSoC First Touch kit has two sides of 14 pins each so the type "2 sides" can be selected. Using the Component Properties toolbar, the pin spacing and width can be configured using the measurements taken

earlier in the application note. The name and reference designation can also be entered at this point. The following two figures are the Component Properties toolbar and screen shot of the pins on the workspace to this point.

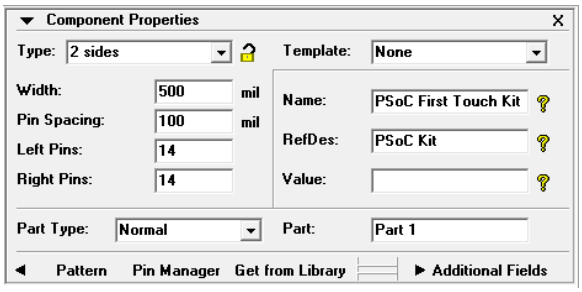


Figure 10: Component Properties Toolbar for PSoC First Touch Kit

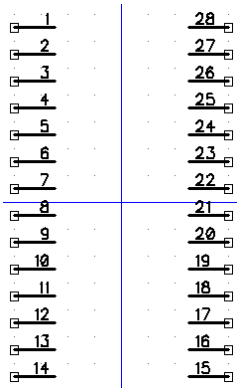


Figure 11: 2 sides Component Type (28 pins)

Now that the layout of the pins is set, Pin Manager can assist in naming and classifying the pins. Located on the Component Properties toolbar is the Pin Manager tool. Click on it and the window will open containing information about each pin. Name, number, x and y coordinates, length, type, and electric characteristics can be specified.

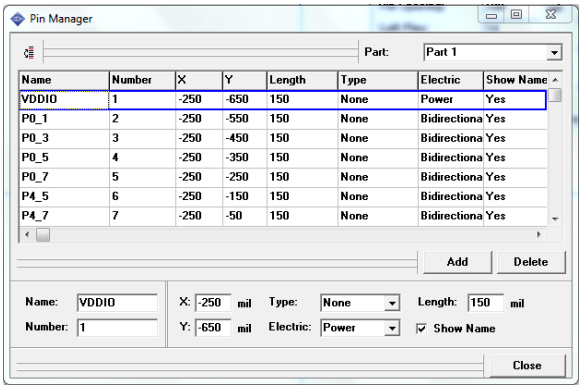


Figure 12: Pin Manager for First Touch Kit

Refer to the appendix for a table containing the corresponding names and electric characteristics for all of the PSoC First Touch Kit pins. Finally, due to the way the pins were specified (i.e. VDDIO pin 1 and P0_1 pin 2 and so on) and the configuration needed so that the pins match when plugging the board in, the component needs to be rotated such that pin 1 is in the bottom left corner of the component and pin 28 is in the bottom right hand corner of the component.

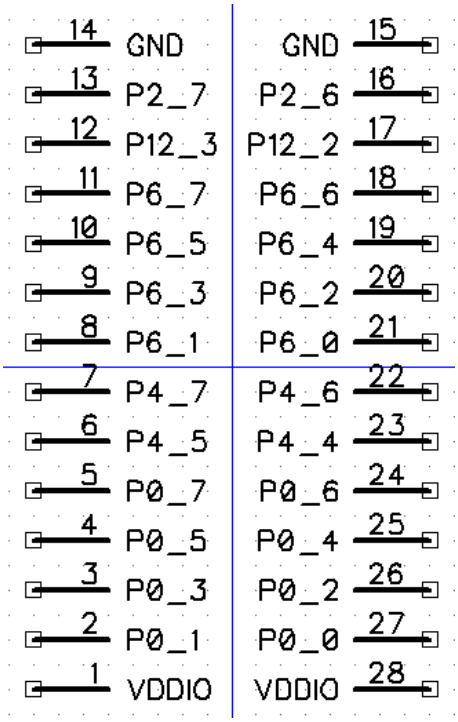


Figure 13: Final PSoC First Touch Kit Component

Now that the component has been created and is properly configured, the object should be locked so no pins are unintentionally moved at any point. This can be done by selecting all the pins and going to the Edit drop down menu and selecting "Lock Selected." Now the pins are unable to be moved.

The final step in finishing up the component will be to attach a pattern. After schematic capture is completed, the corresponding pattern for a component provides the footprint on the PCB board for the silkscreen, etc. Pattern creation for PSoC First Touch Kit and Arduino shields is discussed in another Application Note from ECE480 Design Team 1. Attaching a pattern is done by selecting Pattern on the bottom of the Component Properties toolbar. The Attached Pattern window will pop up on the screen.

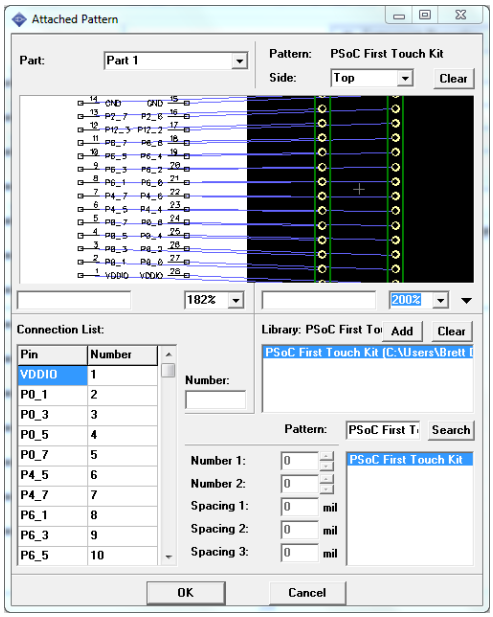


Figure 14: Attached Pattern window

The library must be selected from the proper directory and within that library the proper pattern can be selected and attached to the component.

Arduino Shield Component

For the Arduino Shield component creation, a new library needs to be created. For this component, each individual header will be created within the library and then copied and placed in one component to form the entire shield. This will demonstrate more capabilities within DipTrace Component Editor. Using the measurements from before, all of the headers can be created. The individual pin spacing is 100 mils.

First, start with the two 6-pin headers. The headers are one row of pins so the previous method of using the 2 sides component type is not applicable here. Keep the type as Free. Pins or pin lines will be placed to form the headers. Using the Objects menu or Objects toolbar, place six individual pins or a 6-pin line in the workspace. Ensure the spacing between each pin is 100 mils by rolling the pointer over each pin and referring to the coordinates at the bottom of the screen. This can also be verified in Pin Manager.

In Pin Manager, the pins need to be named and the electric characteristics need to be specified as was done for the PSoC First Touch Kit. Refer to the appendix for the pin names corresponding to the pin numbers for all of the headers. Repeat this process for the other 6-pin header and the two 8-pin headers remembering to "Add New to Library" each time. Rotate the 8-pin headers such that pin 1 is on the bottom for each of these components. This is important for when the individual headers are brought together to form the Arudino shield.

For the ICSP header, use the 2 sides component type with three pins on the left

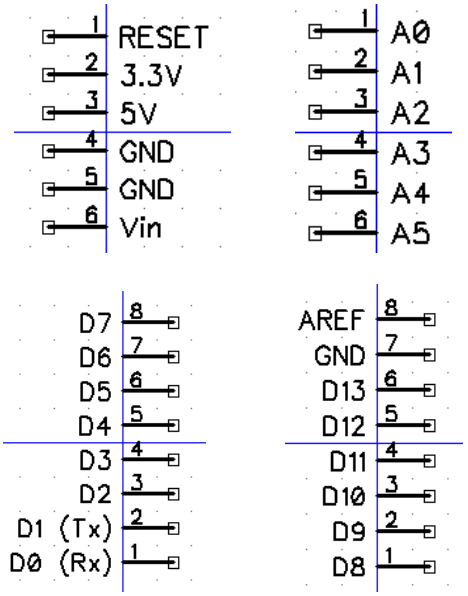
and three pins on the right. The width is 100 mil and the pin spacing is 100 mil. Be sure that these values are changed in the Component Properties toolbar. The ICSP pin numbering convention is a little different than what the DipTrace 2 sides component type uses.

5 3 1
6 4 2

Figure 15: ICSP Pin Numbering

This can be changed by simply double-clicking on each individual pin and changing the pin numbers accordingly. Then in Pin Manager the proper names can be assigned. Note: For the ICSP Header, it may be best to not show the pin names due to the compact pin spacing.

With each component for all of the individual headers including spacing, pin names, etc., completed, the Arduino shield component can be created using all of the individual headers.



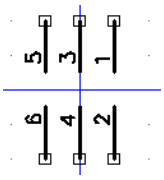


Figure 16: Individual Arduino Headers

This can be done by simply selecting the header from the component window and pasting it into a new component window within the Arduino library. Do this for each header until all the headers are in the Arduino shield component window. Refer back to the measurements taken to ensure that all the headers are properly spaced from one another. Because each component was created separately and all of the components have pins numbered 1, 2, 3, etc., the pin numbers will have to be changed. Again, refer to the appendix for the pin numbering scheme. Remember to lock the pins in place and attach the proper pattern (If this was not done for each individual header, it may be a good idea to go back and do this). The final Arduino shield component appears in Figure 17 below.

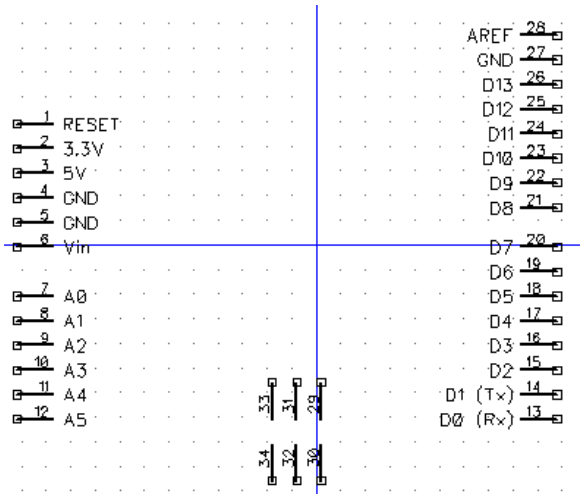


Figure 17: Final Arduino Shield Component

Note: ICSP pin names are not showing because of space constraints.

Observations

DipTrace Component Editor allows for customization of your components ranging from the display, to the physical dimensions, and even the electric characteristics. This note was prepared using certain numbering conventions, grid size, and units but the user can choose what works best for them and their project.

Conclusion

Due to the popularity and adaptability of PSoC and Arduino, and the potential for an expanding user community, it may be of interest to design a PCB with either the PSoC First Touch Kit or an Arduino shield on board. This application note can help guide a user in DipTrace Component Editor through the process of creating these components.

References

Cypress Semiconductor. "PSoC® 5 FirstTouch™ Starter Kit Guide." *Cypress.com*. Cypress Semiconductor Corporation. Web. 10 Nov. 2011. <http://www.cypress.com/?docID=27015>.

DipTrace Professional PCB Design Tool, www.diptrace.com

Appendix

PSoC First Touch Kit Pin Management		
Number	**Name**	**Electric**
1	VDDIO	Power
2	P0_1	Bidirectional
3	P0_3	Bidirectional
4	P0_5	Bidirectional
5	P0_7	Bidirectional
6	P4_5	Bidirectional
7	P4_7	Bidirectional
8	P6_1	Bidirectional
9	P6_3	Bidirectional
10	P6_5	Bidirectional
11	P6_7	Bidirectional
12	P12_3	Bidirectional
13	P2_7	Bidirectional
14	GND	Power
15	GND	Power
16	P2_6	Bidirectional
17	P12_2	Bidirectional
18	P6_6	Bidirectional
19	P6_4	Bidirectional
20	P6_2	Bidirectional
21	P6_0	Bidirectional
22	P4_6	Bidirectional
23	P4_4	Bidirectional
24	P0_6	Bidirectional
25	P0_4	Bidirectional
26	P0_2	Bidirectional
27	P0_0	Bidirectional
28	VDDIO	Power

Arduino Power Header Pin Management		
Number	**Name**	**Electric**
1	RESET	Bidirectional
2	3.3V	Power
3	5V	Power
4	GND	Undefined
5	GND	Undefined
6	Vin	Power

Arduino Analog Header Pin Management		
Number	**Name**	**Electric**
1	A0	Bidirectional
2	A1	Bidirectional
3	A2	Bidirectional
4	A3	Bidirectional
5	A4	Bidirectional
6	A5	Bidirectional

Arduino D0-D7 Digtial Header Pin Management		
Number	**Name**	**Electric**
1	D0 (Rx)	Bidirectional
2	D1 (Tx)	Bidirectional
3	D2	Bidirectional
4	D3	Bidirectional
5	D4	Bidirectional
6	D5	Bidirectional
7	D6	Bidirectional
8	D7	Bidirectional

Arduino D8-D13 Digital Header Pin Management		
Number	**Name**	**Electric**
1	D8	Power
2	D9	Power
3	D10	Bidirectional
4	D11	Bidirectional
5	D12	Bidirectional
6	D13	Bidirectional
7	GND	Bidirectional
8	AREF	Bidirectional

Arduino ICSP Header Pin Management		
Number	**Name**	**Electric**
1	MISO	Bidirectional
2	5V	Power
3	SCK	Bidirectional
4	MOSI	Bidirectional
5	GND	Power
6	RESET	Bidirectional

Arduino Shield Pin Management		
Number	**Name**	**Electric**
1	RESET	Bidirectional
2	3.3V	Power
3	5V	Power
4	GND	Undefined
5	GND	Undefined
6	Vin	Power
7	A0	Bidirectional
8	A1	Bidirectional
9	A2	Bidirectional
10	A3	Bidirectional
11	A4	Bidirectional
12	A5	Bidirectional
13	D0 (Rx)	Bidirectional
14	D1 (Tx)	Bidirectional
15	D2	Bidirectional
16	D3	Bidirectional
17	D4	Bidirectional
18	D5	Bidirectional
19	D6	Bidirectional
20	D7	Bidirectional
21	D8	Power
22	D9	Power
23	D10	Bidirectional
24	D11	Bidirectional
25	D12	Bidirectional
26	D13	Bidirectional
27	GND	Bidirectional
28	AREF	Bidirectional
29	MISO	Bidirectional
30	5V	Power
31	SCK	Bidirectional
32	MOSI	Bidirectional
33	GND	Power
34	RESET	Bidirectional