Oregon TECH

An printed circuit board designed to hold all of the components necessary to create an Arduino Uno using Mentor PADS software. This project included designing a usable schematic, and ensuring that the final PCB was error free and ready for submission to a fabrication house.

Arduino Uno Printed Circuit Board

An Oregon Institute of Technology engineering project

Michael Hernandez

Arduino Uno Printed Circuit Board Project

Oregon Institute of Technology

EE 485: Printed Circuit Board Design

Instructors: Jason Sullins and Allan Douglas

Student: Michael Hernandez

Summer 2019

Table of Contents

Lab 1 6
Purpose 6
Part 1: Familiarization with the Arduino Uno 6
ATMEGA328 Microcontroller 6
Power Supply 7
FT231XS USB-to-Serial Converter 9
Part 2: Creating your PartQuest Accounts and Links for Download 10
DigiKey – PartsQuest 11
ODA Starter Library 12
Part 3: Adding all Active Components and Connectors 13
A special note about the redesign of the MIC5205 17
Conclusion 17
Lab 2 18
Purpose 18
Part 1: Completing the Schematic 18
Part 2: Packaging the Schematics 19
Part 3: Generate a Bill of Materials 20
Conclusion 21
Lab 3 23
Purpose 23
Part 1: PADS Layout and Integration 23
Forward & Back Annotation 23
Errors preventing forward annotation 25
Part 2: Define Electrical Layers and the PCB Outline 26
Part 3: Placement of all Electrical Components 28
Conclusion 29
Lab 4 31
Objectives 31
Redesigned Board 31
Part 1: Modify the Layer Stack-Up and create planes. 32
Part 2: Power and Ground Routing 34
Part 3: Signal Routing 34
A note about routing for this project 35
Part 4: Run PADs DRCs and Check Plane Layer Thermals. 36
Conclusion 41
Lab 5 42

Introduction 42
Part 1: Renumber Reference Designators, Verify Silkscreen, and Back Annotate Schematics. 42
Part 2: Generate and Review Gerber files. 43
Part 3: Generate NC Drill CAM document 45
Part 4: Add Board Outlines and Generate Gerber Files 46
Part 5: ODB++ 48
Conclusion 49
Appendix A 50

Table of Figures

Figure 1 - Schematic for the ATMEGA328 Chip 7
Figure 2 - Schematic for the power supply 9
Figure 3 - FT231XS USB-to-serial converter 10
Figure 4 - Account profile screen with Mentor Graphics 11
Figure 5 - Account profile screen with Digi-Key 11
Figure 6 - ATMEGA328 successfully downloaded 12
Figure 7 - ATMEGA328 placed onto the schematic 13
Figure 8 - Redesign of the ATMEGA328 component 14
Figure 9 - Redesign of the FT231XS component 15
Figure 10 - Redesign of the LM1117 component 15
Figure 11 - Redesign of the MIC5205 component 16
Figure 12 - Initial schematic once all of the modified components have been added 17
Figure 13 - Completed Schematic 19
Figure 14 - Screen to select which parameters to varify 20
Figure 15 - No errors in this schematic 20
Figure 16 - Project Integration Screen 24
Figure 17 - Dispersed Components 25
Figure 18 - Library Services 26
Figure 19 - Board Thickness 26
Figure 20 - Layer 1 27
Figure 21 - Layer 2 27
Figure 22 - Layer 3 27
Figure 23 - Layer 4 27
Figure 24 - 2X2 board layout with component decals 28
Figure 25 - Component pads have been placed to untangle the rats nets as much as possible 29
Figure 26 - Redesigned Layout 32
Figure 27 - Creating a copper plane 33
Figure 28 - Top view of the completed board 34
Figure 29 - Bottom view of the completed board. 35
Figure 30 - Example demonstrating the effects of error checking being disabled. Vias could be laid on top of pads or traces, and traces could overlap. 36
Figure 31 - A clearance report revealed a few errors, most of which were fixed except for errors resulting from the IC pads being to close together which would need to be changed in one of the document settings 37
Figure 32 - Connectivity report revealing no errors 37
Figure 33 - Fabrication report revealing no errors 37
Figure 34 - Results of the plane report 38
Figure 35 - Via connecting the net to the +3.3V plane 39
Figure 36 - Via connecting the +5.0V node to the power plane 40
Figure 37 - Via connecting the net to the DGND plane 40
Figure 38 - The eco toolbar is greyed out 42
Figure 39 - Name and number on top layer 42

Figure 40 - Back annotation is disabled 43
Figure 41 - Top Layer 44
Figure 42 - Bottom Layer 44
Figure 43 - DGND Layer 44
Figure 44 - Power Plane Layer 44
Figure 45 - Pastemask Top Layer 45
Figure 46 - Silkscreen Top Layer 45
Figure 47 - NC Drill Drawing 46
Figure 48 - Board Outlines 47
Figure 49 - Zipped gerber files; click on the image to open the files. 47
Figure 50 - ODB++ Error 48
Figure 51 - ODB++ 11 Viewer 49

Lab 1

Purpose

This lab exercise is designed to gain familiarity with the schematic and layout of the Sparkfun Redboard variant of the Arduino Uno as well as to learn the basics of how to create a new project using the various pieces of software that will be used throughout the duration of this project. Specifically, the software that will be covered in this lab are as follows:

- Learning how to create a new project in DxDesigner
- Learning how to import components with PartQuest
- Learning how to modify imported schematic symbols

Part 1: Familiarization with the Arduino Uno

The schematic to be used for the duration of this project is to be downloaded from the Sparkfun website. Please see Appendix A for a copy of the schematic. Part 1 will analyze all three portions of the schematic. The ATMEGA328 microcontroller, the power supply, and the FT231XS USB to serial converter.

ATMEGA328 Microcontroller

The ATmega328 is a single-chip microcontroller created by Atmel in the megaAVR family. It has a modified Harvard architecture 8-bit RISC processor core. The pin used in the Sparkfun Redboard variant uses a 32-pin chip. The datasheet for this chip can be found at: http://ww1.microchip.com/downloads/en/DeviceDoc/ATmega48A-PA-88A-PA-168A-PA-328-P-DS-DS40002061A.pdf.

The crystal oscillator is necessary for clock operations in the microcontroller; it feeds into the clock multiplexer along with four other components. The low power crystal oscillator is not capable of driving other clock inputs, however the full swing crystal oscillator is. The crystal oscillator is connected to the XTAL 1 and XTAL 2 pins on the ATMEGA328 microcontroller. The full swing oscillator has rail-to-rail swing at the XTAL 2 output pin. Either a quartz crystal or a ceramic resonator may be used for the crystal oscillator.

Test points are for checking against a ground reference to see if the test point (usually on a power node) has been shorted out.

The In-Circuit Serial Programming (ICSP) connector allow for in-circuit serial programming. In other words, the data is moved into a microcontroller via serial communication; the microcontroller then executes these instructions. In the ATMEGA328 board there are six pins – four pins that are used for data transfer, a 5V VCC input, and a ground pin. In this microcontroller, the MOSI is the master output/slave input pin, the MISO pin is the master

input/slave output pin, , and the SCK pin is the master clock output/slave clock output pin. These three pins are associated with registers DDB 3/4/5 respectively.

An LED is connected to the microcontroller so that the microcontroller can be visually inspected for shorts. If the LED is on then the Arduino has power and is functioning properly. The resistor connected in series with the LED is to reduce the input voltage to the LED, thereby protecting the LED from burning out.

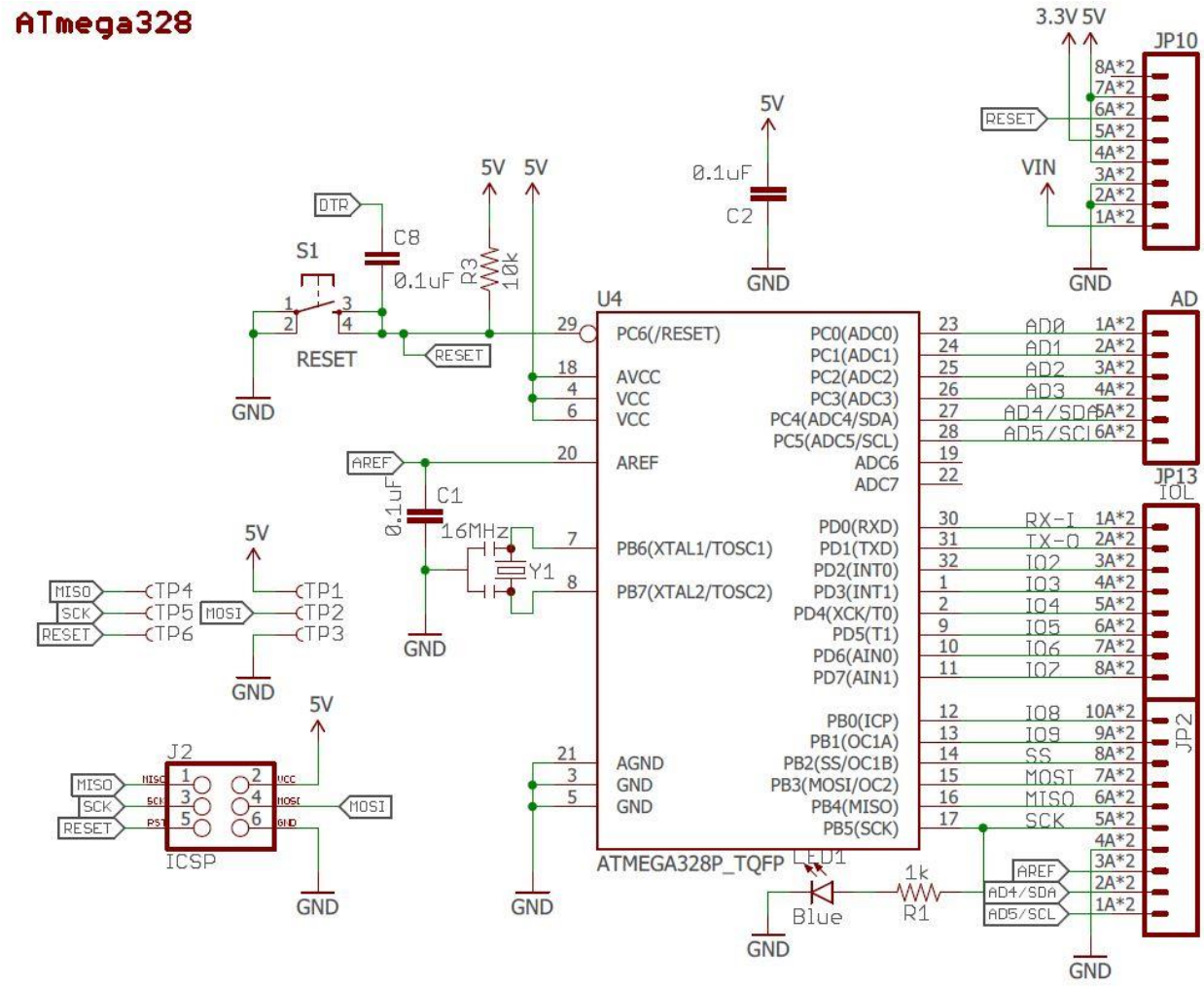


Figure 1 - Schematic for the ATMEGA328 Chip

Power Supply

The LM1117 IC3 is a current limiting device which limits the current on the microcontroller and all pins to 150mA regardless of how many pins are in use. The datasheet for the LM1117 can be found at http://www.ti.com/lit/ds/symlink/lm1117.pdf.

D1 is an MBRA140 Schottky power rectifier whose function is to act as a low voltage, high frequency switch. The datasheet for the MBRA140 can be found at https://www.onsemi.com/pub/Collateral/MBRA140T3-D.PDF.

LED2 is connected to 5 volts as a means of visually checking to determine if the device is powered on.

U2 is an MIC5205 150 mA Low-Noise LDO Regulator. Its primary purpose is to maintain a 150mA output current in order to power the FT231 USB-to-Serial Converter. Two capacitors, C6 and C7 are required on the output pin of U2 to prevent current oscillation. The datasheet can be found at http://ww1.microchip.com/downloads/en/DeviceDoc/20005785A.pdf.

Q1 is a Schottky diode connected in parallel with a MOSFET transistor. An example of this is the MAX17521 60V, 1A, Dual-Output, High-Efficiency, Synchronous Step-Down DC-DC Converter. The MAX17521 uses peak-current-mode control. Each output can be operated in the pulse-width modulation (PWM) or pulse-frequency modulation (PFM) control schemes. The datasheet for MAX17521 can be found at https://datasheets.maximintegrated.com/en/ds/MAX17521.pdf.

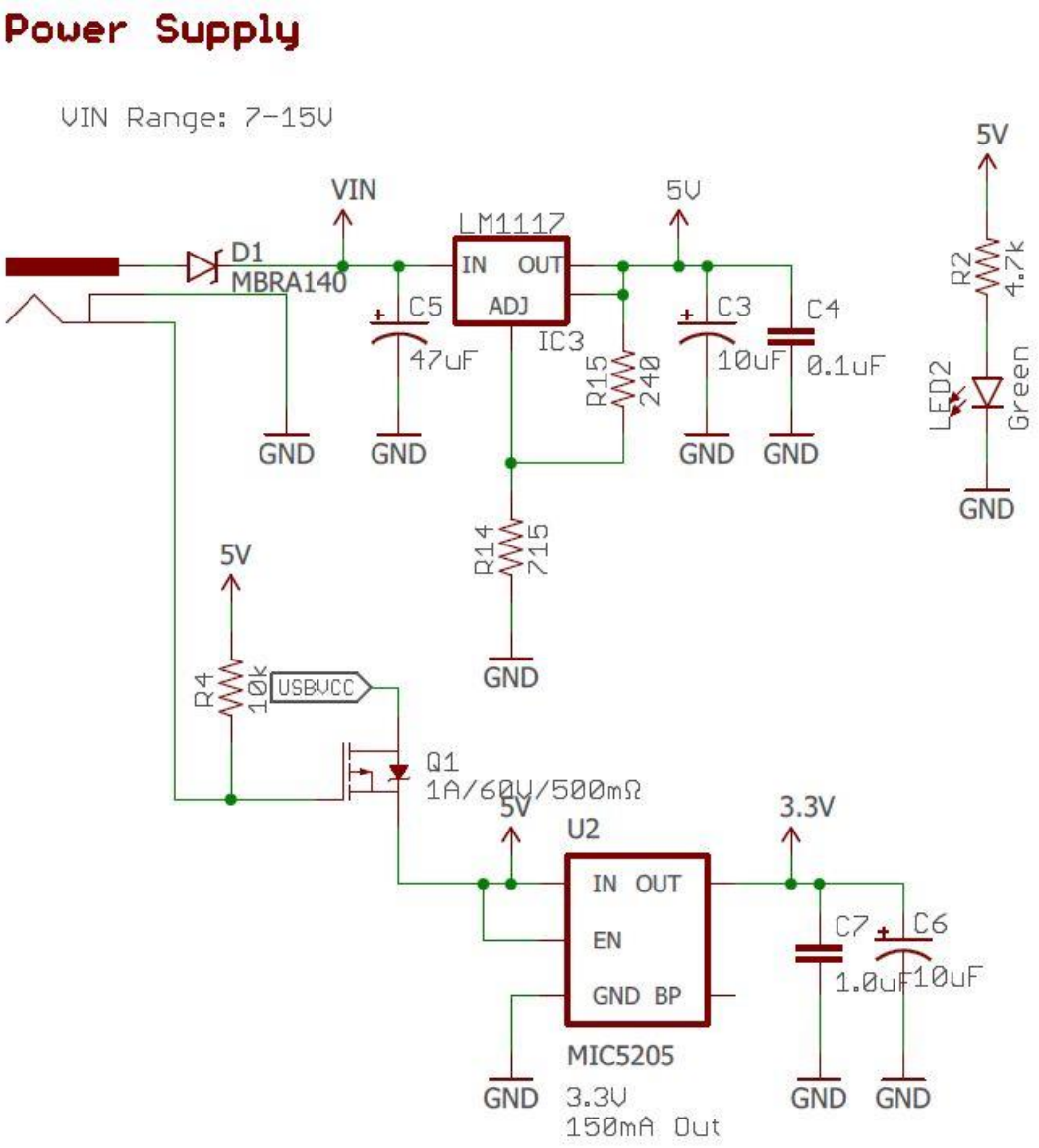


Figure 2 - Schematic for the power supply

FT231XS USB-to-Serial Converter

U1 is the FT231XS package developed by Future Technology Devices International. It is a USB to serial UART interface. It is a convenient chip because the entire USB protocol handled on the chip so no USB specific firmware programming required. The datasheet for the FT231X is available at https://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT231X.pdf.

F1 is a resettable fuse with a maximum voltage of 6V, and a maximum current of 500mA. The function of the fuse is to protect the USB port. One such fuse is the 0805 SMT fuse from Bourns. The datasheet is available at https://www.sunrom.com/get/285900.

Q2 is the 2N7002PW 60 V, 310 mA N-channel Trench MOSFET. The purpose of Q2 is to act as a high-speed switch between the FT231XS and the ATMEGA328 reset pin. The datasheet is available at https://static6.arrow.com/aropdfconversion/81cb195590ad52e7f88c5301252a6a7779066d32/1782814141267262n7002pw.pdf.

Resistors 8 and 9 are to lower the voltage between the transmit and reset pins on the FT231XS. The resistors have values of 81KΩ and 91KΩ respectively, causing a substantial voltage drop prior to entering the chip.

The reset portion of the ATMEGA328 contains a switch which pulls the voltage low from the 5-volt VCC. An inverting gate negates the input signal prior to entering the reset pin. The C8 capacitor converts the level-triggered signal DTR into an edge-triggered signal and has the effect of level shifting it to within the operating voltage range of the MCU. Data Terminal Ready (DTR) is a signal line used for hardware flow-control in serial protocols. The DTR signal coming from the FT231XS leaves the chip as an inverted signal and goes through the Q2 switch prior to entering the ATMEGA328. The 3.3V and 5V differential voltages on either side of the 2N7002PW act as voltage triggers; the addition of a voltage signal from the FT231XS chip to the 3.3V side of the 2N7002PW pushes the current forward into the ATMEGA328.

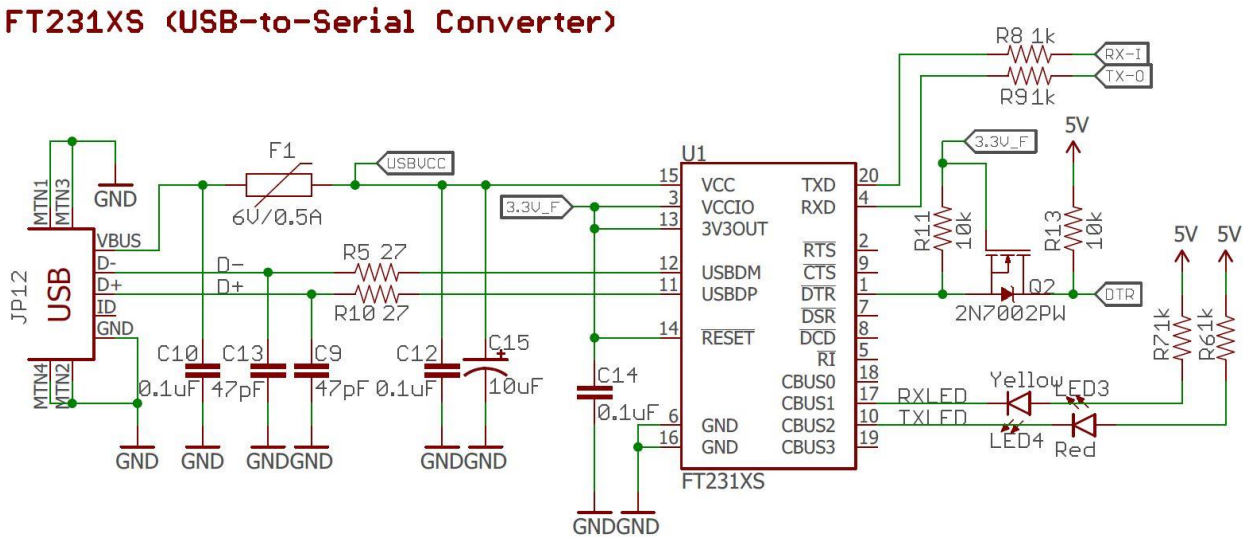


Figure 3 - FT231XS USB-to-serial converter

Part 2: Creating your PartQuest Accounts and Links for Download

This section of the lab is designed to introduce the student to the PartQuest utility from DigiKey, as well as to create a starter library that will be used in subsequent labs.

DigiKey – PartsQuest

For this portion of the lab accounts with Mentor Graphics and DigiKey are to be created. After accounts are successfully, the Atmel ATMEGA328P-AUR component from PartsQuest is to be downloaded.

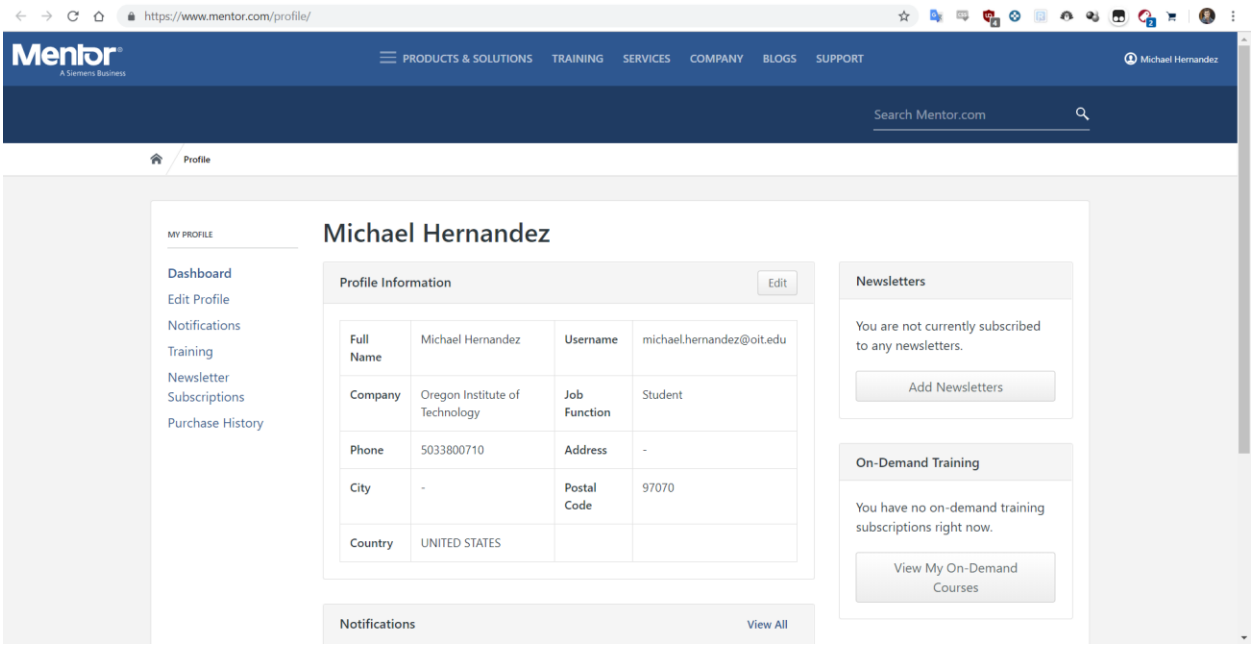


Figure 4 - Account profile screen with Mentor Graphics

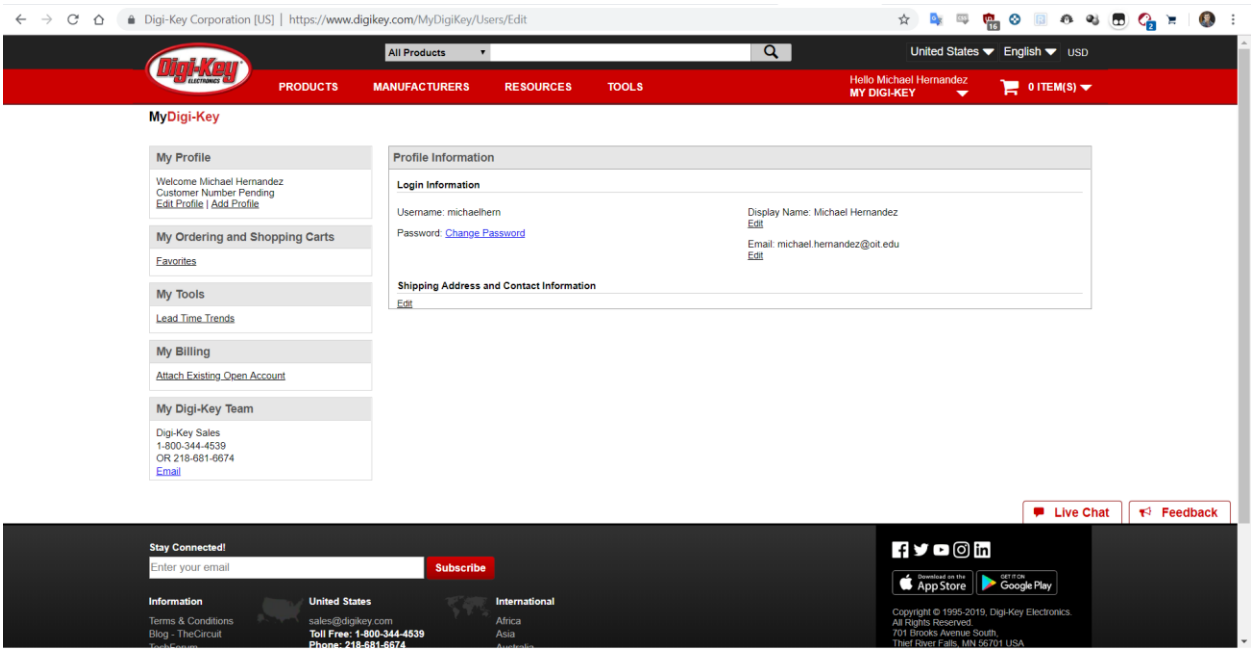


Figure 5 - Account profile screen with Digi-Key

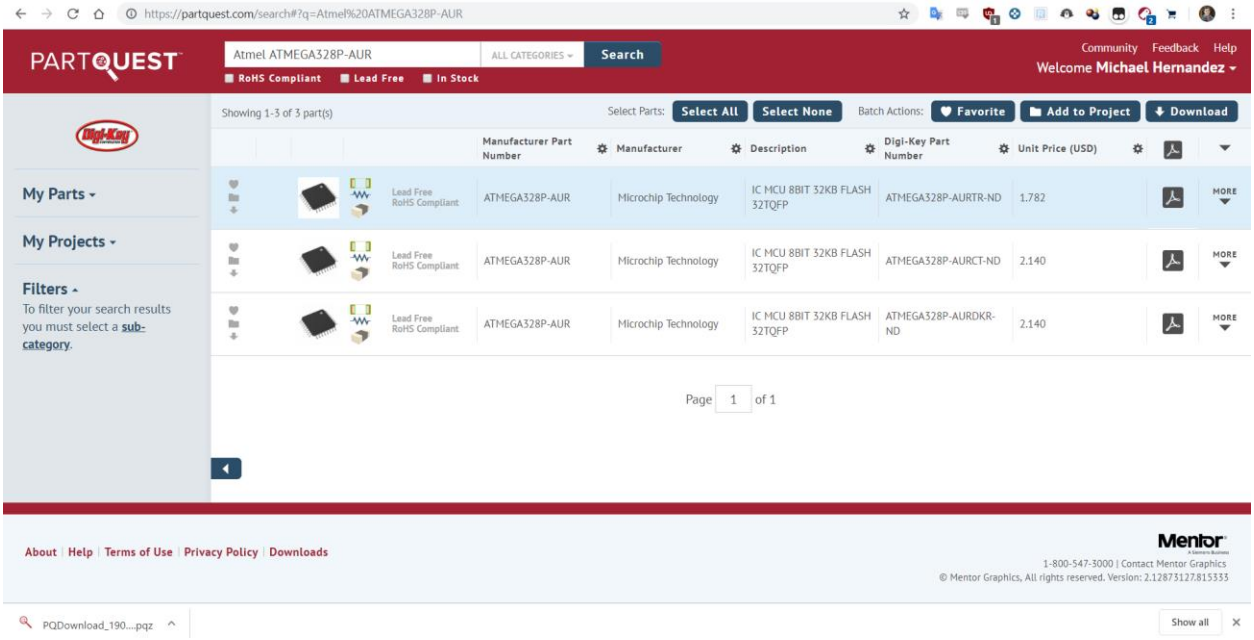


Figure 6 - ATMEGA328 successfully downloaded

ODA Starter Library

For this portion of the lab, the ODA Starter Library for PADS Standard Plus is to be downloaded for later use.

Once the library has been downloaded, and components successfully imported into PADS, the ATMEGA328 IC is to be placed on the schematic as seen below.

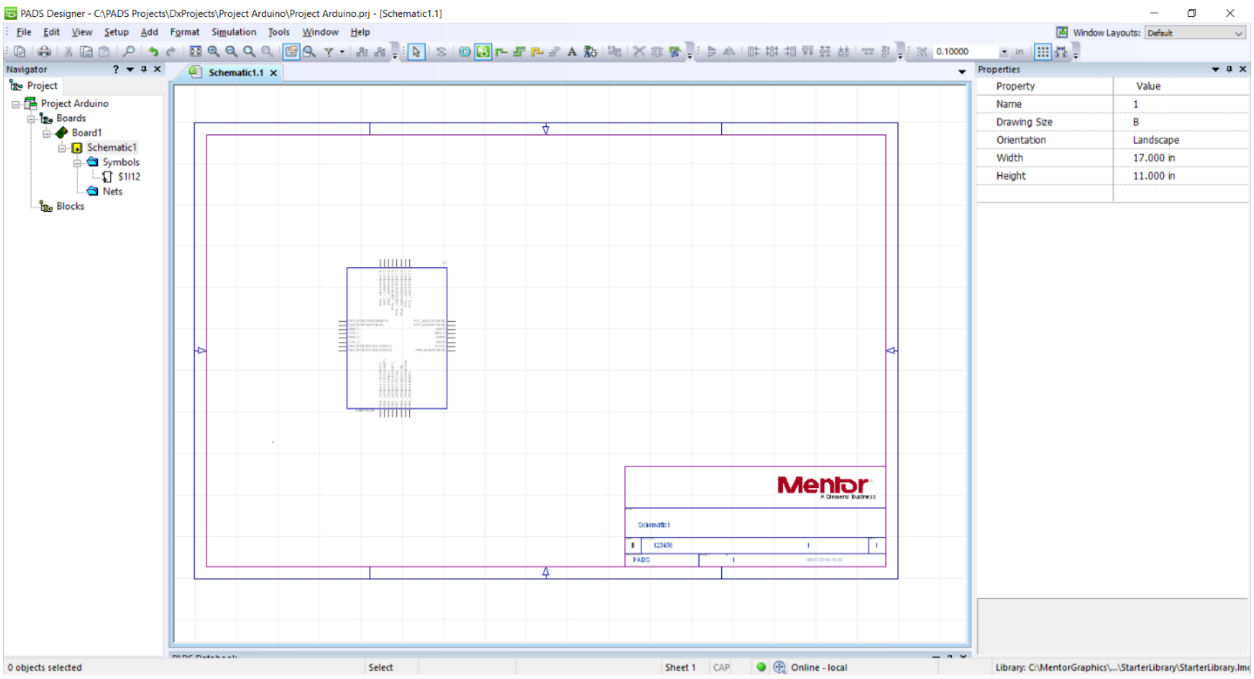


Figure 7 - ATMEGA328 placed onto the schematic

Part 3: Adding all Active Components and Connectors

Once part 2 is completed, part three may be started. The following components are to be downloaded from Partquest and imported into PADS: IC3, U1, U2, Q1, Q2, micro USB, J2, JP10, JP12, IOL, and AD. These components should reflect the components found on the schematic in Appendix A.

Once all components are successfully imported into PADS they need to be modified to match the layout of the Redboard schematic.

In order to correctly reflect the schematic that is being used as a guide for this project, several components needed to be redesigned using the component editor tool in PADS. The end result can be seen in the following figures.

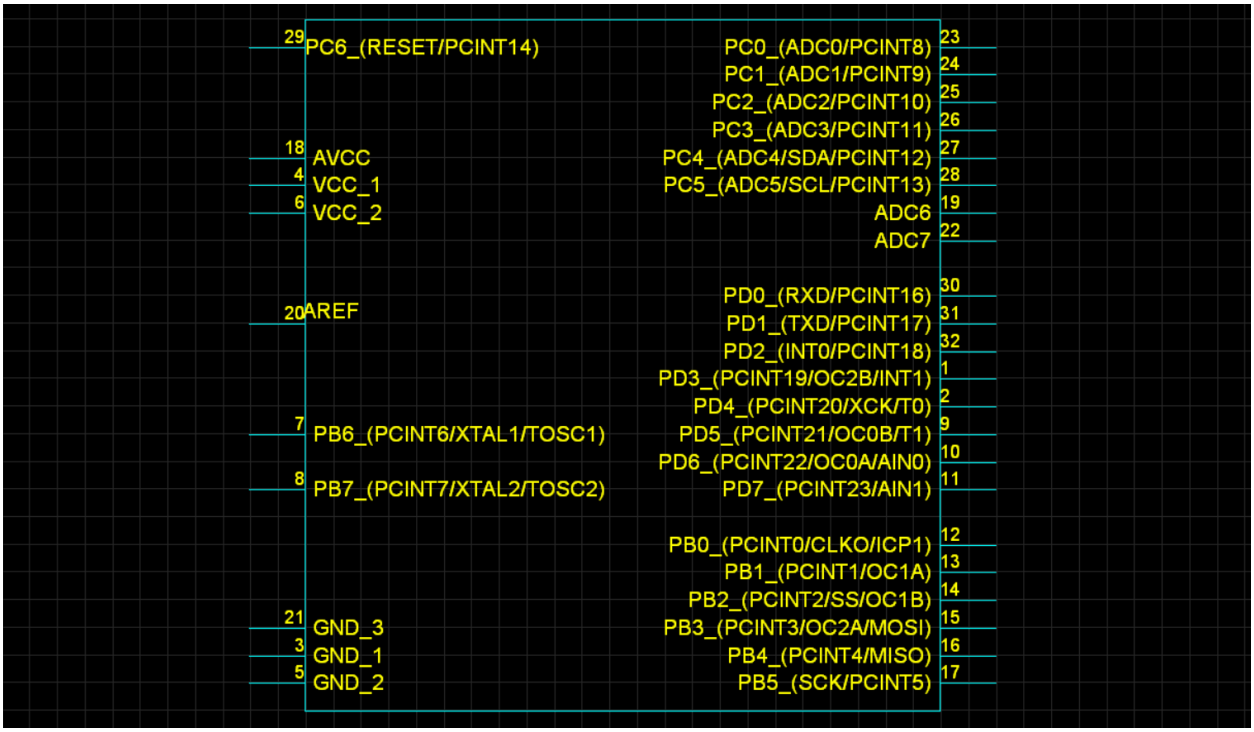


Figure 8 - Redesign of the ATMEGA328 component

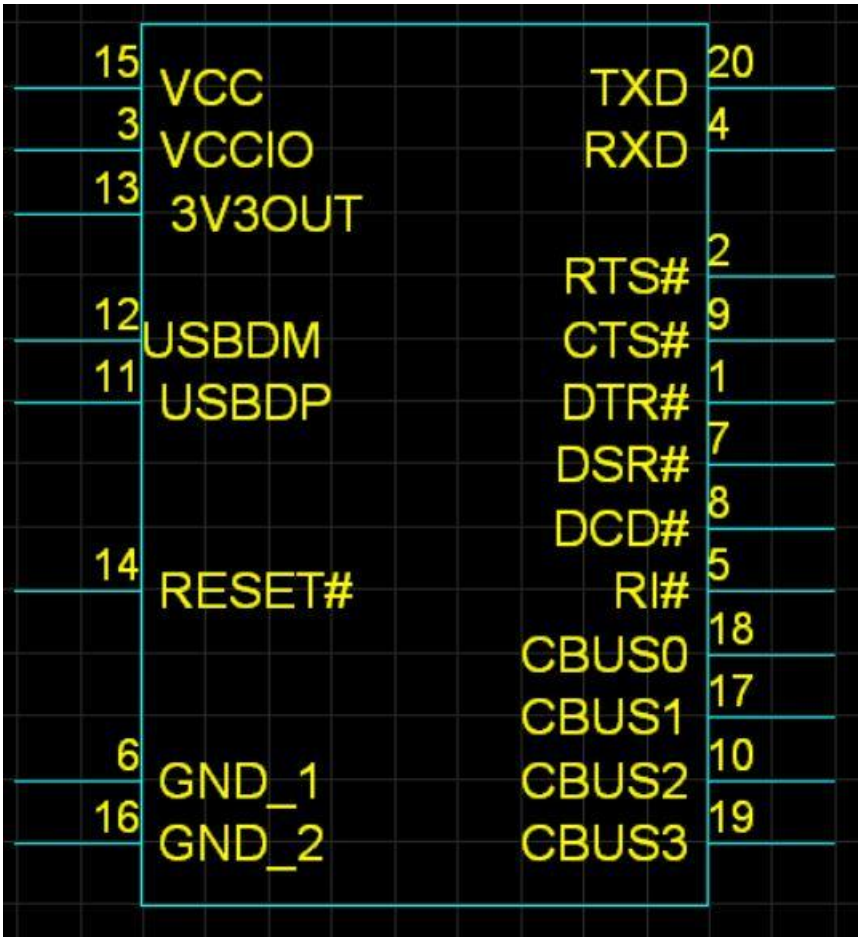


Figure 9 - Redesign of the FT231XS component

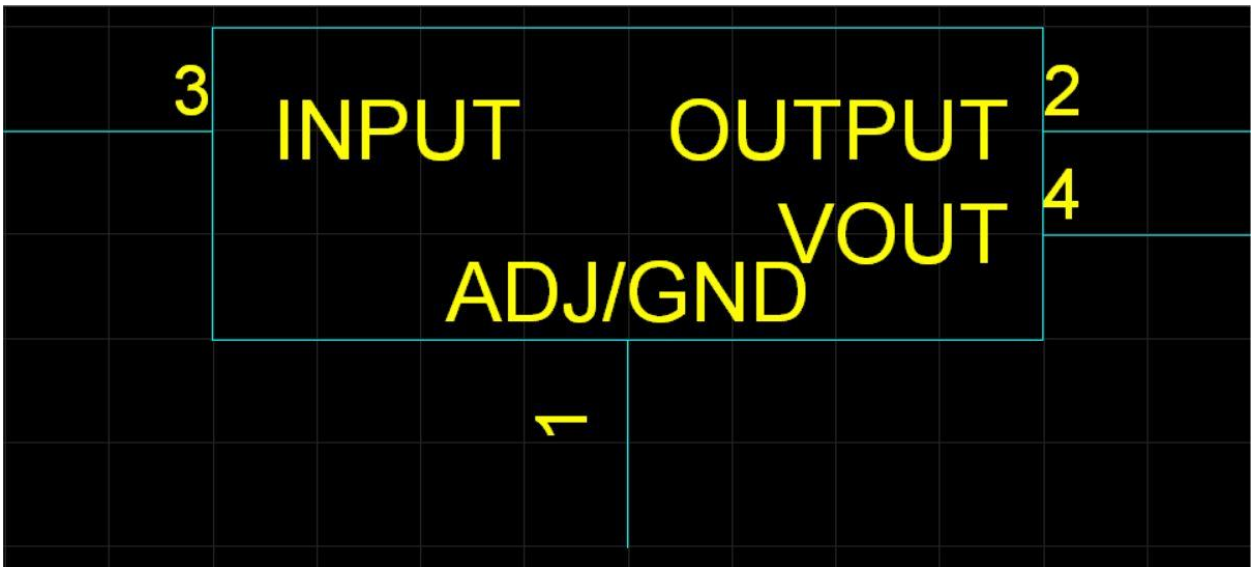


Figure 10 - Redesign of the LM1117 component

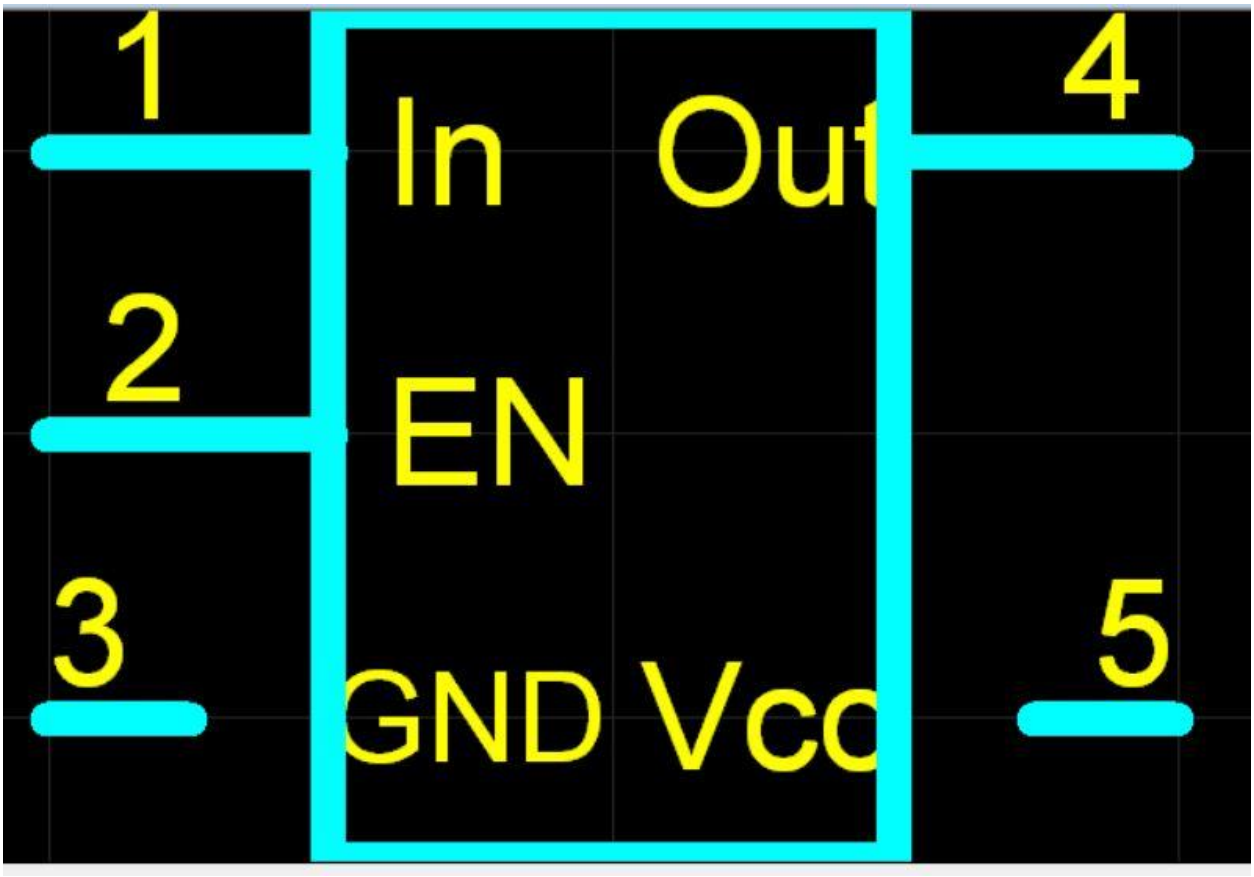


Figure 11 - Redesign of the MIC5205 component

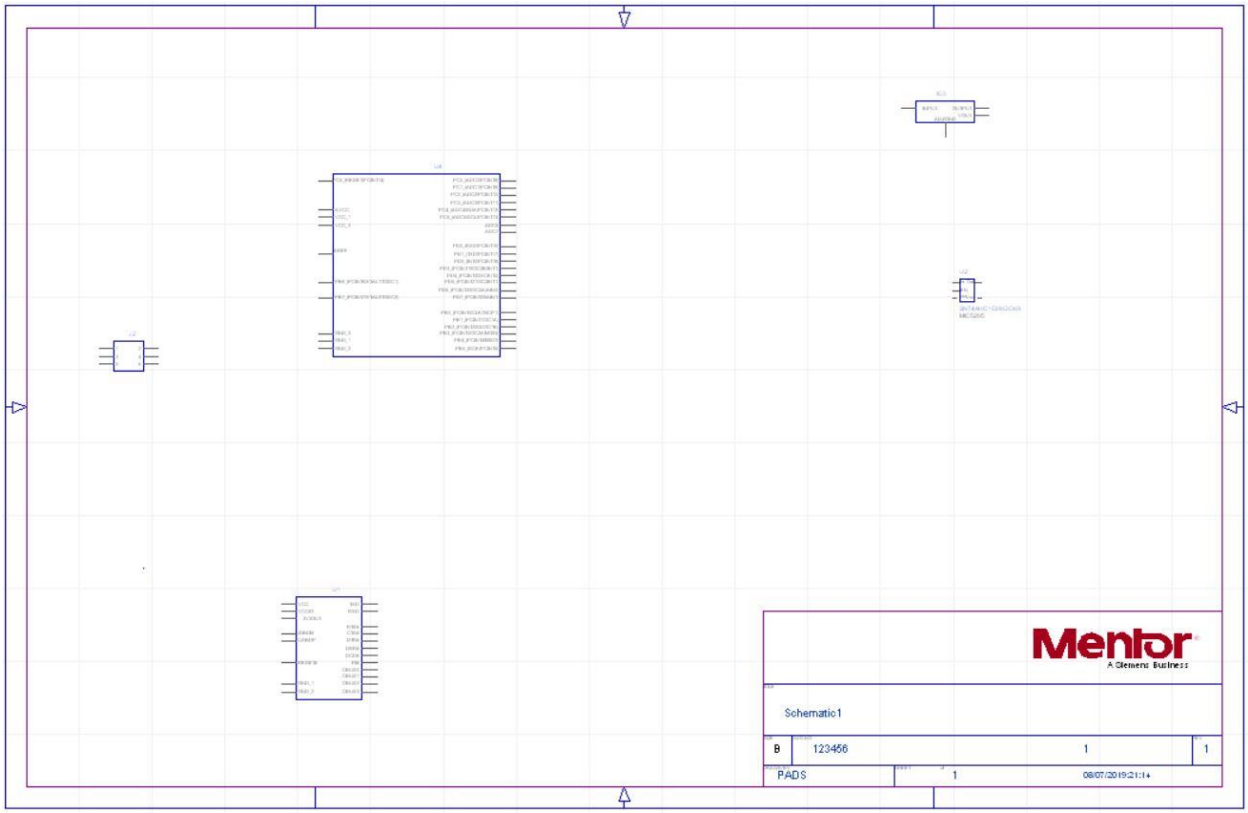


Figure 12 - Initial schematic once all of the modified components have been added

A special note about the redesign of the MIC5205

When the MIC5205 was downloaded from Partquest and imported, an error message appeared in PADS stating that the component already existed. Therefore, it was not possible to import the MIC5205 into PADS. Several names were given as alternative components to be used. Many of them had minor variations that would change the functionality of the component, for example an inverting gate coming out of pin 4. Also, all of the pins needed to be relabeled from A, B, etc. to be In, En, and Out. The component that was chosen to be replaced was the SN74AHC1G08DCKR.

Conclusion

This lab introduced the concepts of using Partquest to find parts not natively in PADS. The use of PADS software for schematic design, and how to use PADS to modify existing schematic components. An existing schematic for an Arduino Uno was analyzed in detail, particularly the active components that will be used for creating the device. This project will attempt to faithfully recreate the Sparkfun Arduino Uno schematic for purposes of PCB design.

Lab 2

Purpose

The purpose of this lab is to complete the schematic in PADS in order to get it to a point where it is possible to begin routing. A bill of materials will also be generated.

Part 1: Completing the Schematic

The remainder of the schematic started in lab 1 is to be completed to match the schematic in appendix A. Table 1 outlines the components to be used in the schematic.

SparkFun Redboard Components	Alternatives
ATMEGA328P	Import from PartQuest, then modify the symbol
J2 Connector	Use ODA 15-91-0060
JP10 Connector	Use ODA 15-91-0080
AD Connector	Use ODA 15-91-0060
IOL Connector	Use ODA 15-91-0080
JP2 Connector	Use ODA 15-91-6102
Power Connector	Use ODA 55932-0210
LM1117 Regulator	Use ODA LM1117MP-ADJ/NOPB
U2 Regulator	Use ODA LM2937IMP-3.3/NOPB
Q1 P-Channel FET	Use ODA ZXM61P03F
Q2 N-Channel FET	Use ODA 2N7002
FT231X	Use SOP63R5P600_20_D865N decal, create custom symbol
USB Connector	Use ODA 54819-0572
Reset Button	Use ODA EVQ-P2602W
16MHz Crystal	Use ODA ATS160SM-T and two CAP-0402-NPO-22pF-50V-5TOL
500mA Fuse	Use ODA 0460_500
47pF Capacitor	Use ODA CAP-0603-NPO-47pF-50V-5TOL
0.1uF Capacitor	Use ODA CAP-0603-X7R-100nF-16V-10TOL
10uF Capacitor	Import from PartQuest, DIGIKEY_478_8193_2_ND
1uF Capacitor	Use ODA CAP-0603-X5R-1uF-6V3-10TOL
Green LED	Use ODA HLMP-6500-F0011
Red LED	Use ODA HLMP-6000-E0011
Yellow LED	Use ODA HLMP-6400-F0011
Blue LED	Use a green LED instead

Table 1 - Components to be used in the schematic

The schematic that was created closely represents the SparkFun schematic. A few minor changes were included. A custom logo was inserted into the schematic. The USB-to-Serial converter portion was broken up into two different portions, the USB and the FT321XS portions. Each net was labeled with a tag manually drawn.

The resulting schematic can be seen in figure 13.

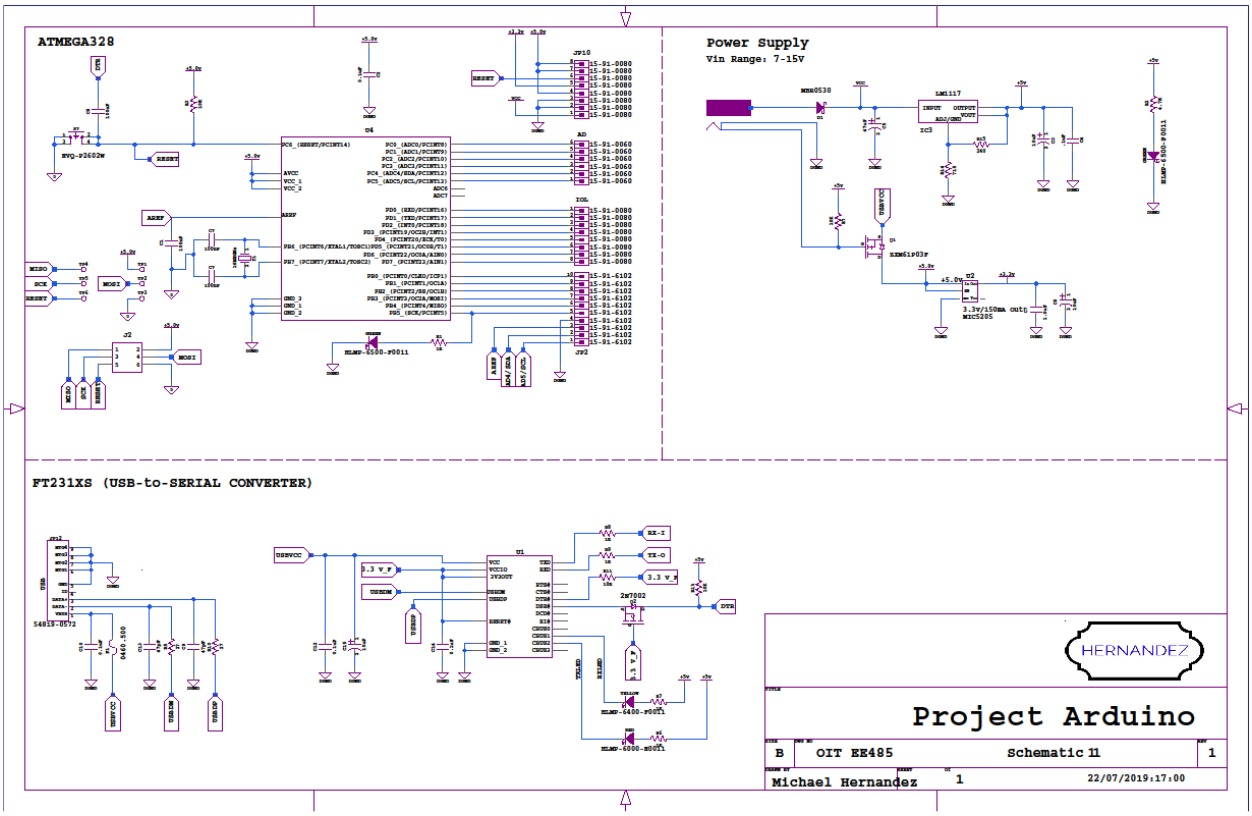


Figure 13 - Completed Schematic

Part 2: Packaging the Schematics

The packager application is to be ran to ensure that there are no errors or warnings.

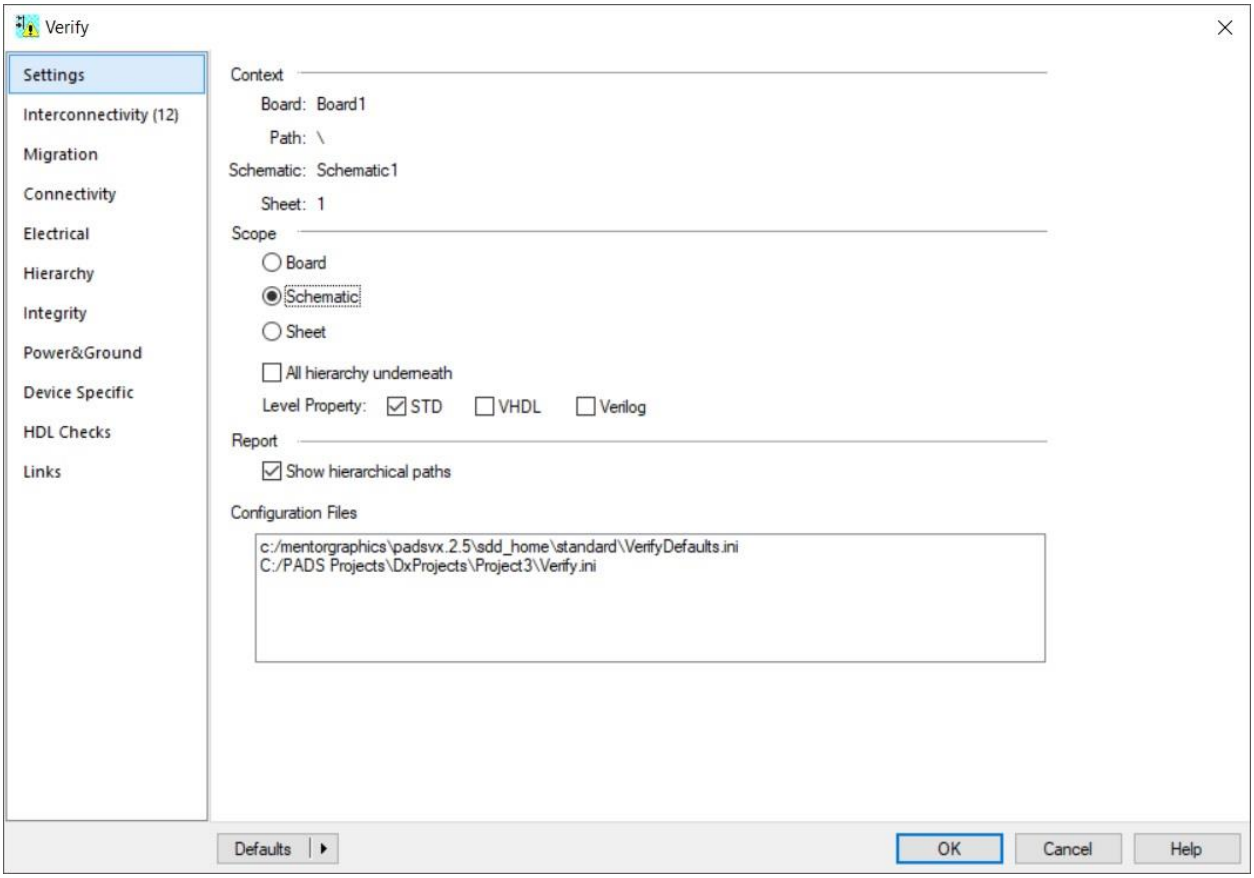


Figure 14 - Screen to select which parameters to varify

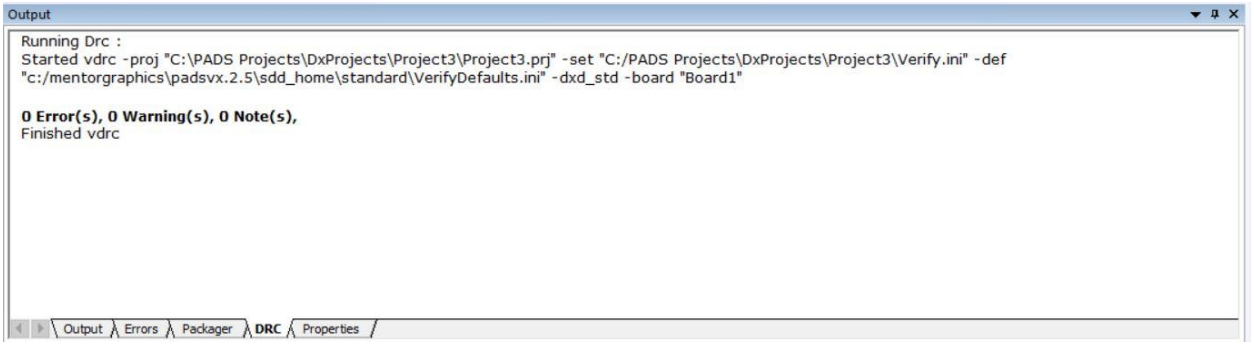


Figure 15 - No errors in this schematic

Part 3: Generate a Bill of Materials

A bill of materials needs to be generated. The output file will be in the form of an excel spreadsheet, although it can be saved as a .txt file as well. The bill of materials is generated

under tools → part lister. The parameters that are to be listed as well as the format of the output file can be configured in the window that appears.

#	QTY	Part Number	Value	Ref Designator
1	9	CAP-0603-X7R-100NF-16V-10TOL	.1u	C1-C9
2	1	CAP-0603-X5R-1UF-6V3-10TOL	1u	C10
3	3	TAJT106K010	10u	C11,C13,C14
4	1	TAJT106K010	47u	C12
5	2	CAP-0603-NPO-47PF-50V-5TOL	47p	C15,C16
6	2	HLMP-6500-F0011	GREEN	CR1,CR2
7	1	HLMP-6400-F0011	YELLOW	CR3
8	1	HLMP-6000-E0011	RED	CR4
9	1	MBR0530		D1
10	1	0460_500		F1
11	1	DIGIKEY_ATMEGA328P_AURTR_ND		IC1
12	1	DIGIKEY_768_1129_2_ND		IC2
13	2	15-91-0060		J1,J2
14	2	15-91-0080		J3,J4
15	2	15-91-0080	IOL	J3,J4
16	1	15-91-6102		J5
17	1	54819-0572		JZ
18	1	ZXM61P03F		Q1
19	1	2N7002		Q2
20	3	RES-0402-10K-1TOL	10K	R1-R3
21	5	RES-0402-1K-1TOL	1K	R4-R8
22	1	RES-0402-10K7-1TOL		R9
23	1	RES-0402-240K-1TOL	240	R10
24	1	RES-0402-715R-1TOL	715	R11
25	1	RES-0603-4K7-1TOL	4.7K	R12
26	2	RES-0402-27R-5TOL	27	R13,R14
27	1	EVQ-P2602W		S1
28	6	TP104-01-00		TP1-TP6
29	1	LM2937IMP-3_3/NOPB		U1
30	1	LM1117MP-ADJ/NOPB		U2
31	1	ATS160SM-T	16MHz	Y1

Table 2 - Bill of Materials

Conclusion

The schematic is created by selecting components, usually imported from a source such as partquest, and connecting them in an easy to read format. It is important to select the

components that will be used to create the PCB board as those will be listed in the resulting bill of materials. Once all components of the schematic are connected appropriately and laid out in an easy to read manner, it is then necessary to run a check to ensure that no errors were made in designing the schematic.

Lab 3

Purpose

This lab has three objectives. The schematic components and design netlist need to be integrated onto the board. The electrical layers need to be defined as well as the outline of the PCB. Finally, all of the components need to be placed onto the board for routing.

Part 1: PADS Layout and Integration

Under setup → project integration, the screen shown in Figure 16 - Project Integration Screen will appear. Clicking on the yellow button should solve any warnings that appear. When all of the buttons turn green, the components will appear on the layout. Once the components have been placed on the layout, they are to be spread out so that the rats nest can be seen. This can be easily done using the disperse components feature located under tools → disperse components. The resulting dispersed layout can be seen in Figure 17 - Dispersed Components.

Forward & Back Annotation

Forward annotation is the process of updating a layout to reflect changes made in the schematic. It is to be used whenever a change is made to the schematic and the change needs to be reflected in the layout.

Back annotation is the process of updating the schematic to reflect changes made to the layout. It is to be used whenever a change is made to the layout and the change needs to be reflected in the schematic.

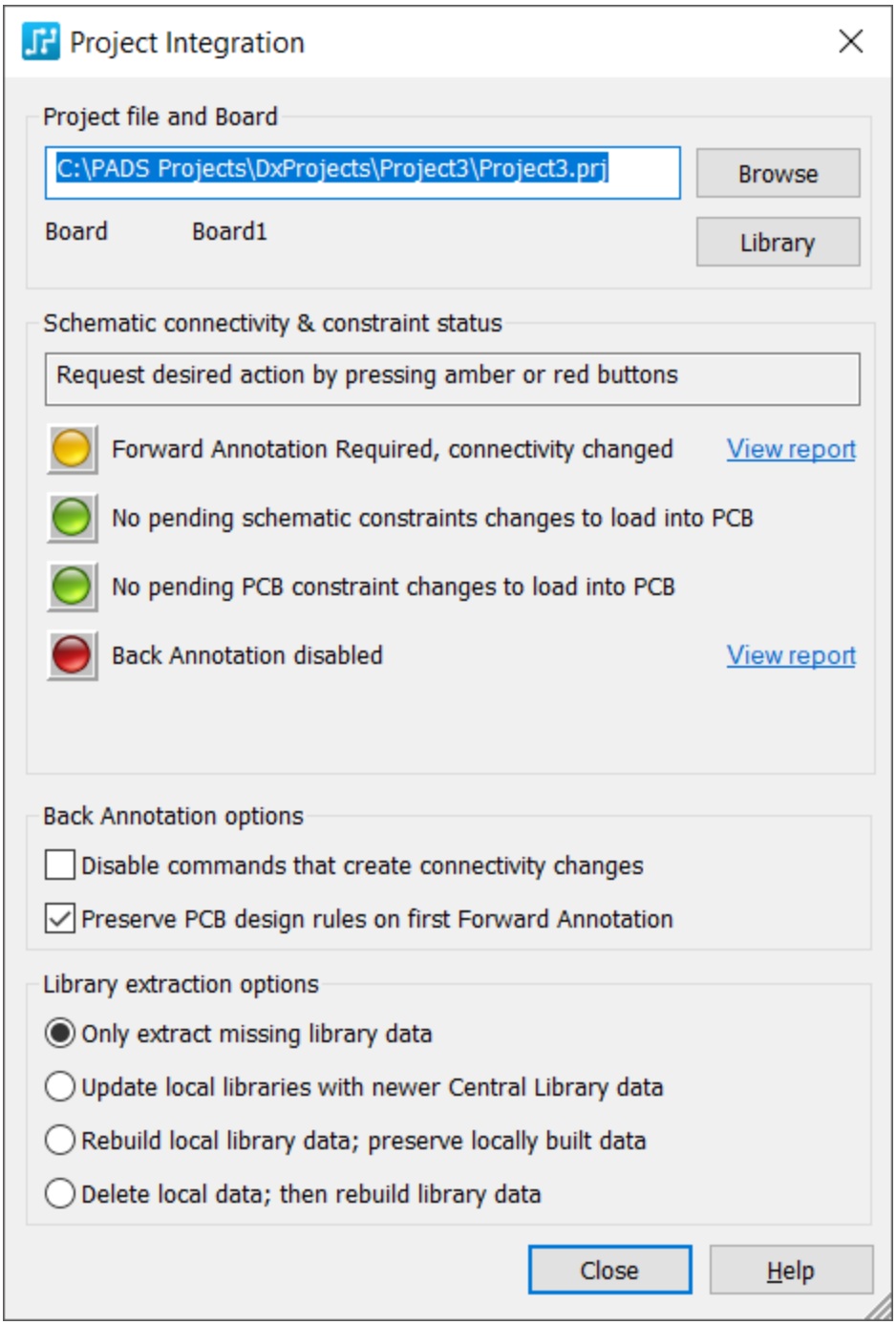


Figure 16 - Project Integration Screen

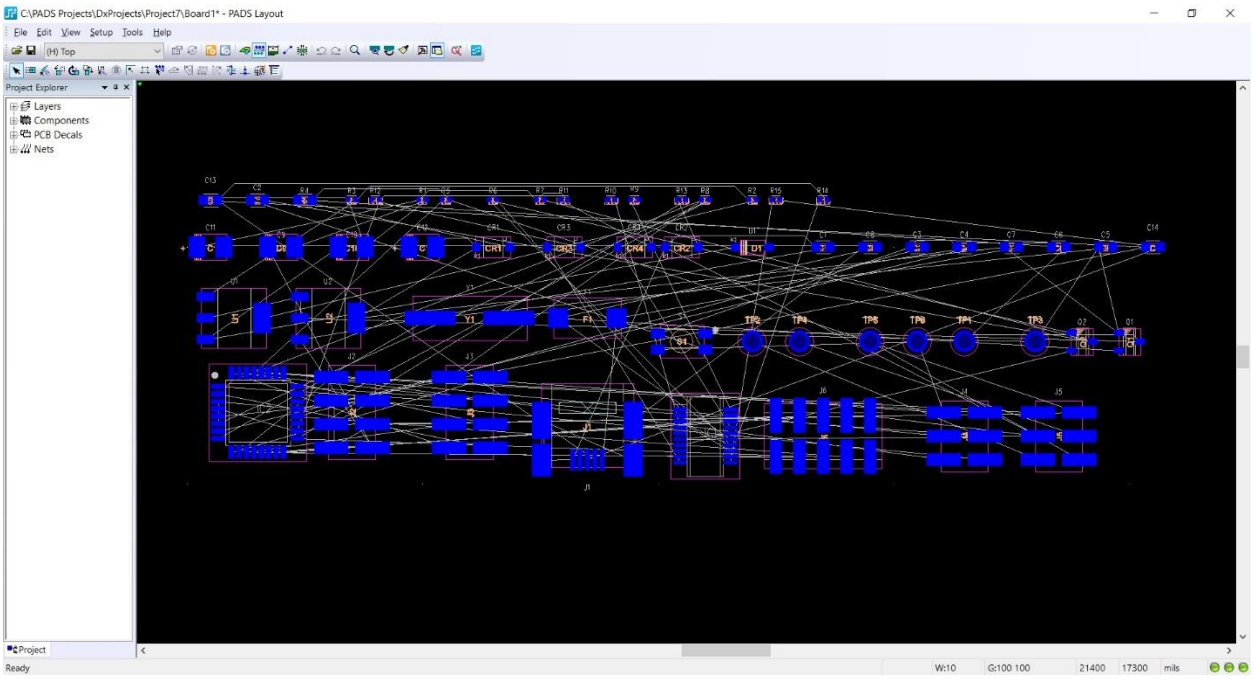


Figure 17 - Dispersed Components.

Errors preventing forward annotation

If a component is missing the decal needed for the layout diagram, an error will appear which states “ERROR: Can not make new PCB item. iCDB counterpart identifier: [two nine digit numbers will appear here]”. Following this, each missing pin will be identified. In order to solve this, the decal must be added to the component within the parts manager library. First open the parts manager library. Then add the .edx file for the component with the missing decal. Third, add the component when the component is added, the screen in Figure 18 - Library Serviceswill appear. Select yes and click through all of the prompts that appear. At this point the decal can now be added to the library without any issues for the component.

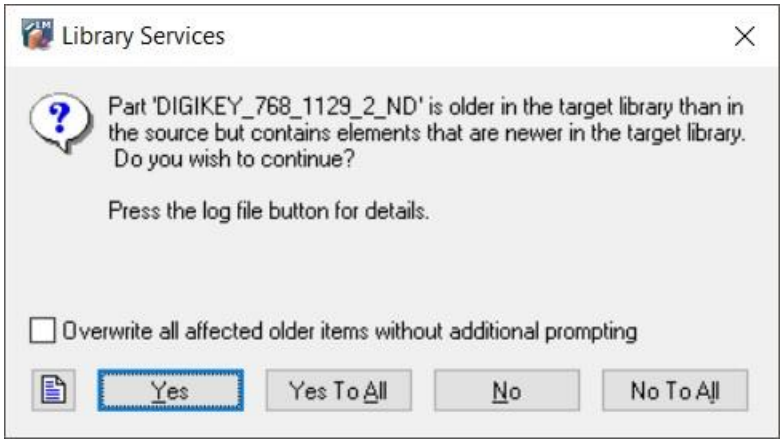


Figure 18 - Library Services

Part 2: Define Electrical Layers and the PCB Outline

This board is to have four layers. Layer 1 is to be for placing components, layer 2 is to be the ground plane, layer 3 is for the +3.3V and +5 volt nets, and layer 4 is for routing. All unused layers are to be disabled.

Next the thickness is to be adjusted so that there is 1 Oz of copper on each layer. The thickness of the two middle layers is to be adjusted so that the board is 62 mils thick.

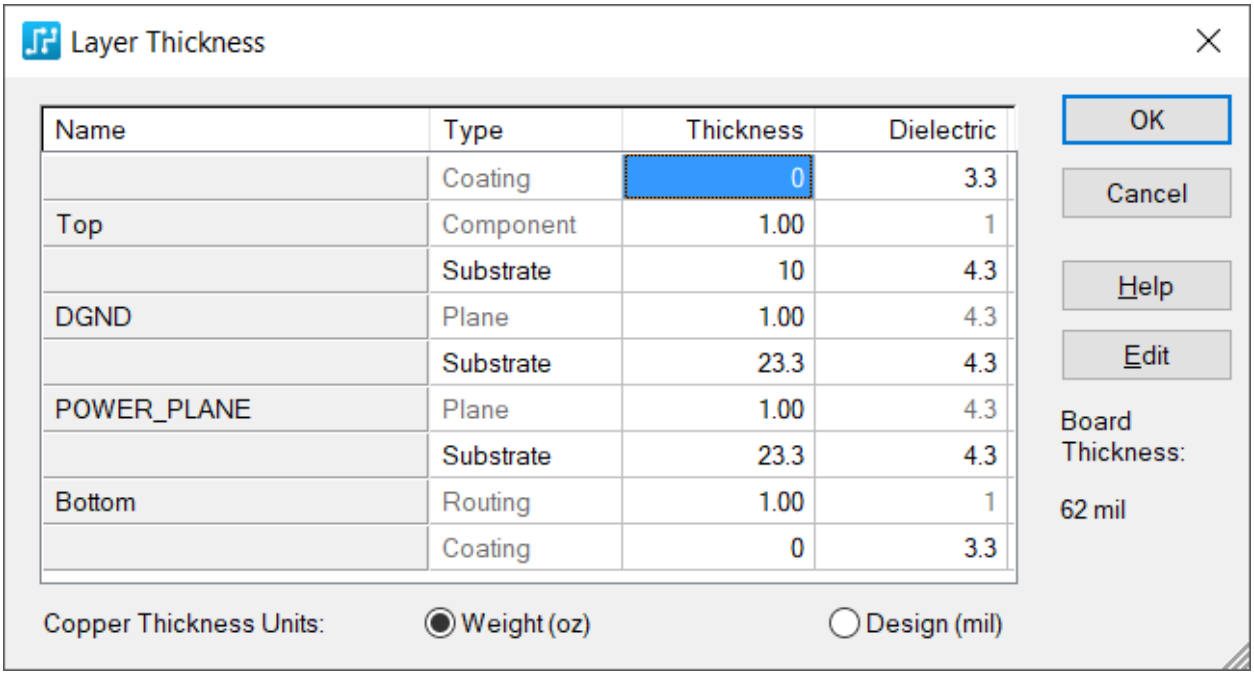


Figure 19 - Board Thickness

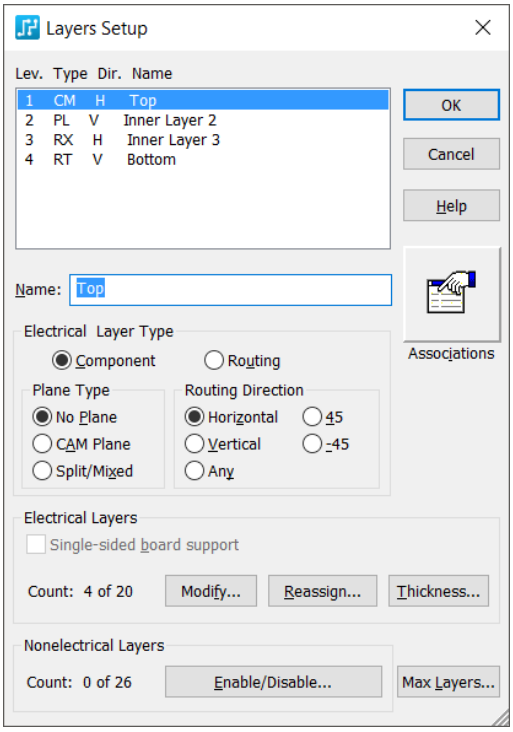


Figure 20 - Layer 1

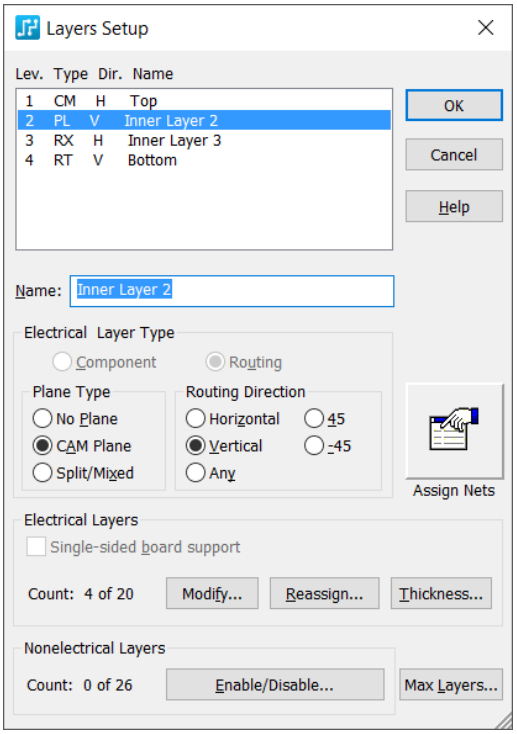


Figure 21 - Layer 2

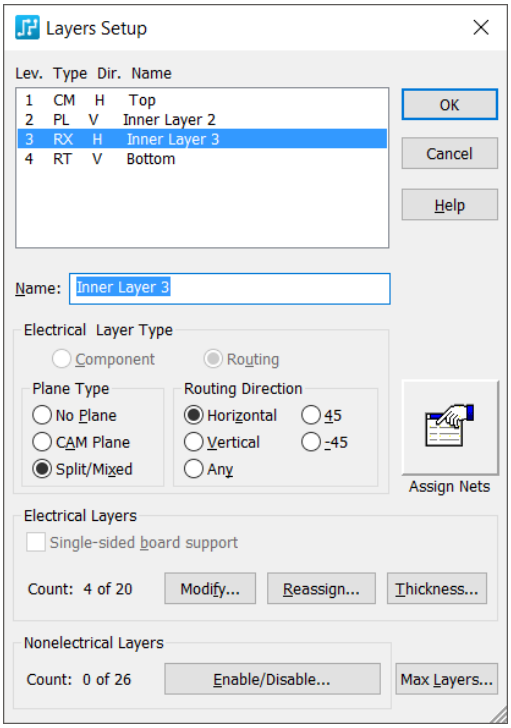


Figure 22 - Layer 3

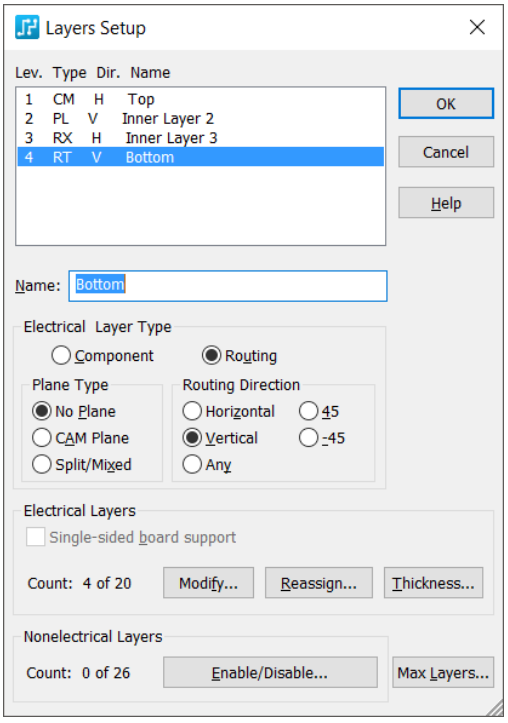


Figure 23 - Layer 4

Once the planes have been defined, the outline needs to be set. A 2x2 outline is to be created for this board to reflect the Redboard Arduino Uno.

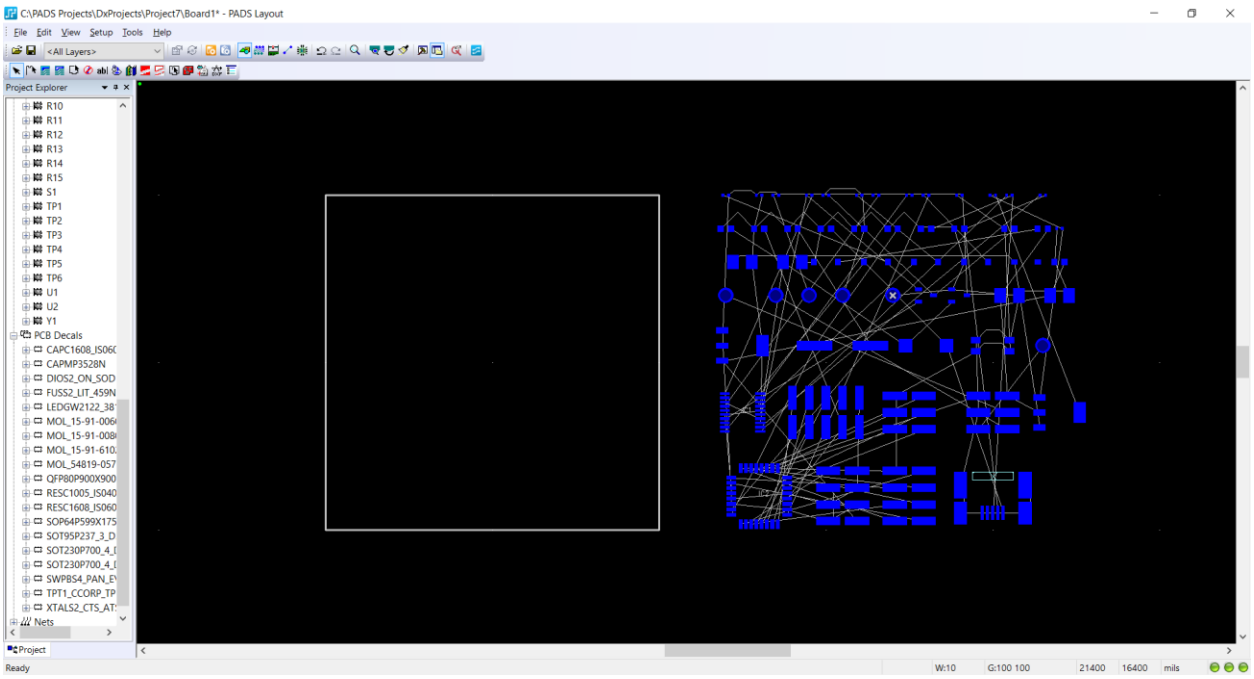


Figure 24 - 2X2 board layout with component decals

Part 3: Placement of all Electrical Components

For part 3, all capacitors and resistors are to be flipped to the bottom side. To do this, the components are to be selected. Once they are selected, they can be flipped to the other side by right clicking and clicking “Flip Side”, or by hitting Ctrl + F.

Once the components are switched to their appropriate sides, they will need to be positioned on the board. First all of the active components are to be placed on the board. The connectors should be placed on the edges of the board. Then the active components need to be placed somewhere in the middle of the board. Finally, the passive components such as resistors and capacitors need to be placed. During placement, the components need to be placed in such a way as to try to make routing in the next step easier.

On the board, components on the top layer appear in blue, while components on the bottom layer appear in red.

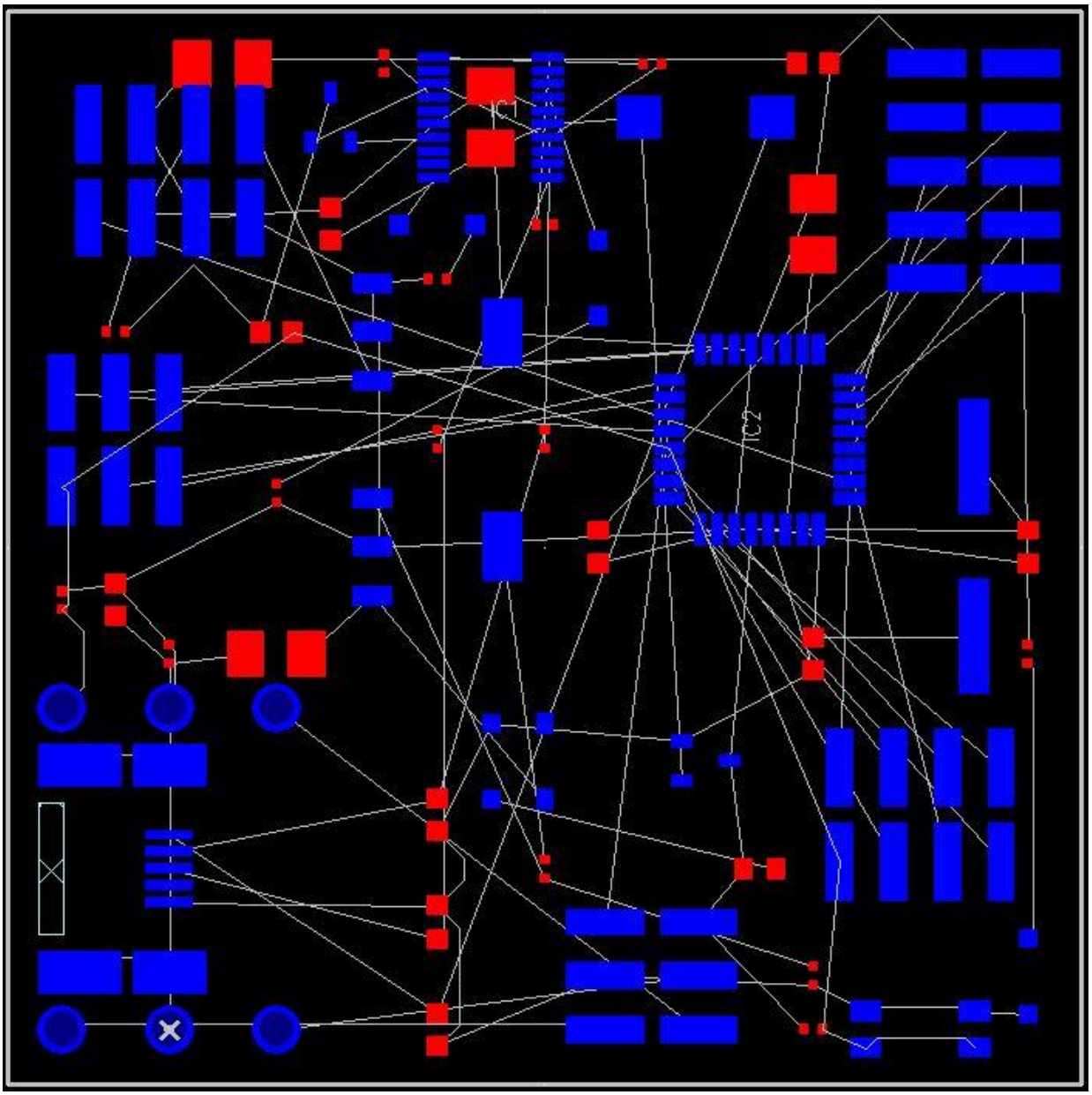


Figure 25 - Component pads have been placed to untangle the rats nets as much as possible.

Conclusion

Placing the layout requires that forward annotation work correctly. If any decals are missing from parts imported from partquest then this issue will need to be solved prior to laying out components. Once forward annotation is finished, all components will be imported into PADS Projects. The next step is to define the layers of the board and set the board to the desired thickness. Once this is completed, the physical outline of the board needs to be laid out. Finally

the components need to be placed within the outlined area in a manner that minimizes tangling of the rats nest for later routing.

Lab 4

Objectives

The purpose of this lab is mainly to route all of the components on the PCB. Additionally, this lab is designed to give practice changing views to verify plane connections, and to run design rule checks to ensure that the design is complete.

Redesigned Board

The layout of the components needed to be redone in order to properly route all of the components later on.

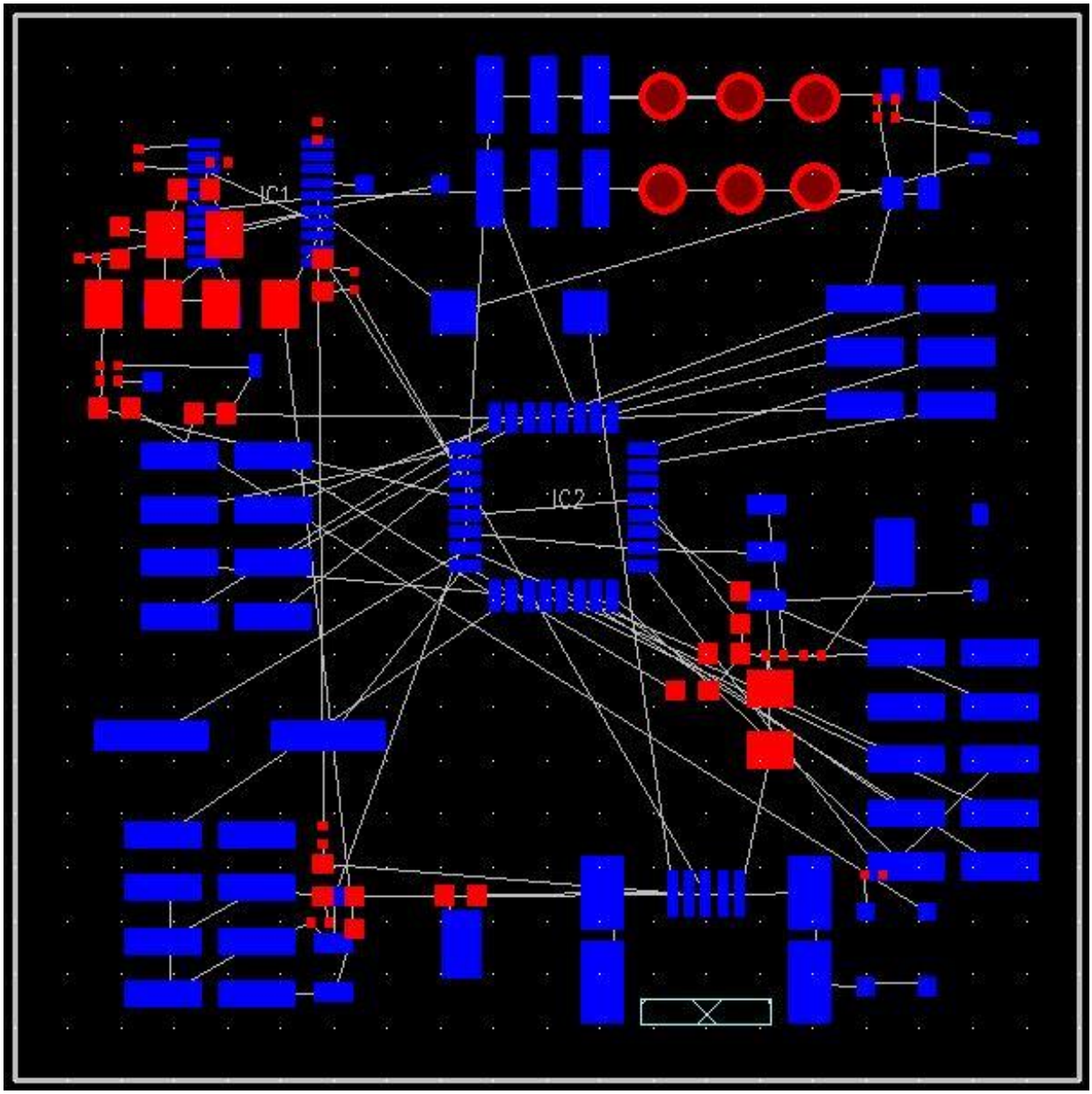


Figure 26 - Redesigned Layout

Part 1: Modify the Layer Stack-Up and create planes.

Layer 3 is to be changed to be a split/mixed plane; this is accomplished under the layer definition screen. Separate plane areas are to be created for both the +5.0V and the +3.3V sections.

Once this has been done, a copper plane is to be poured on the DGND layer, as well as the power plane for the +3.3V and +5.0V areas. To do this, select the board outline, then select “select

shape", right click the board outline again, and select "create copper plane" a screen should appear as shown in Figure 27. The net assignment is to be set to the desired node.

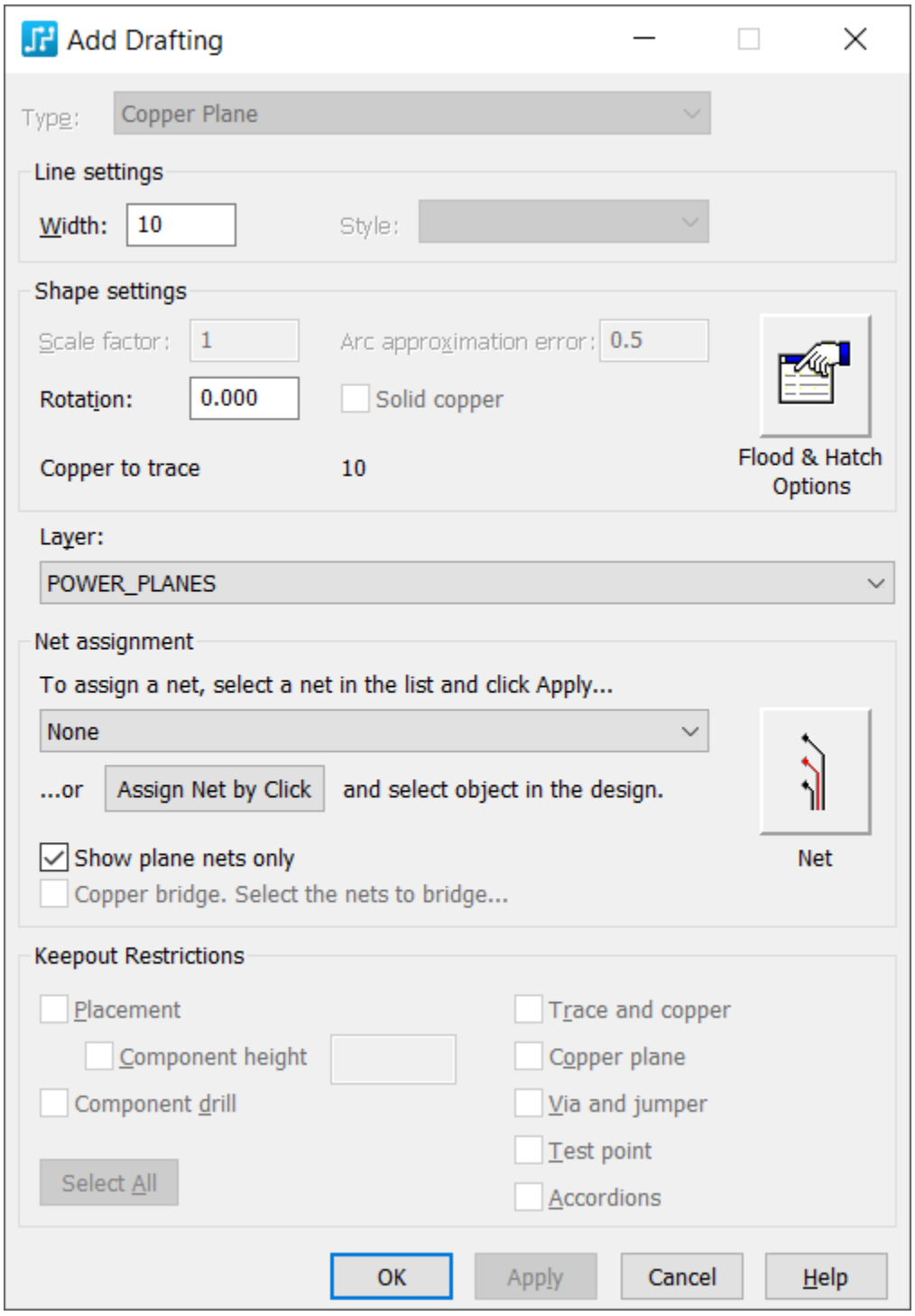


Figure 27 - Creating a copper plane

Part 2: Power and Ground Routing

Vias are to be used to connect the DGND, +3.3V, and +5.0V planes to components that necessitate them. Additionally, the pour manager is to be used to flood the planes with copper.

Due to issues with the program's ability to connect with Mentor's servers, this step was not able to be completed.

Part 3: Signal Routing

All of the required traces are to be connected according to the rat's nest. Vias are to be used to connect the top and bottom sides of the board. Right angles are to be avoided whenever possible.

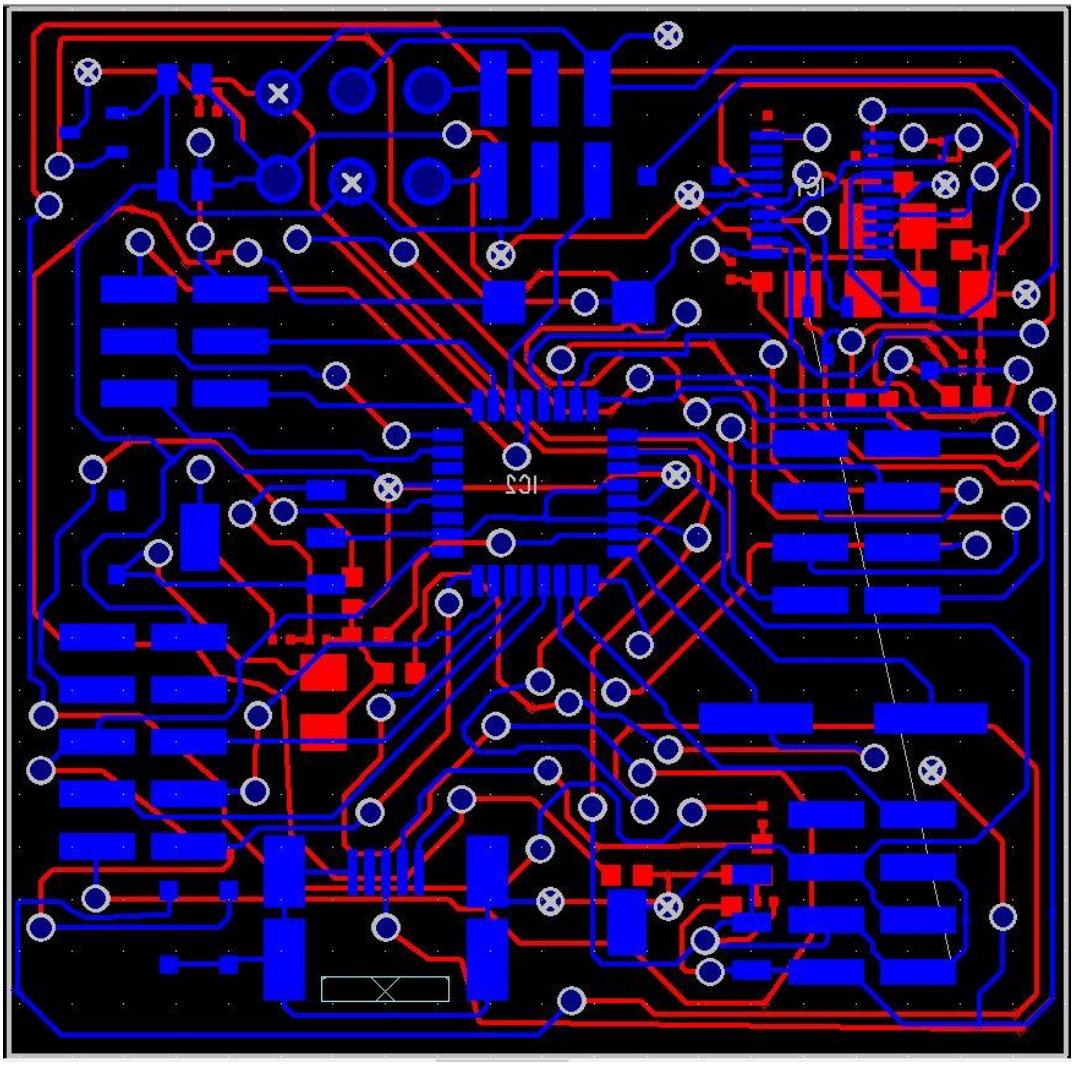


Figure 28 - Top view of the completed board

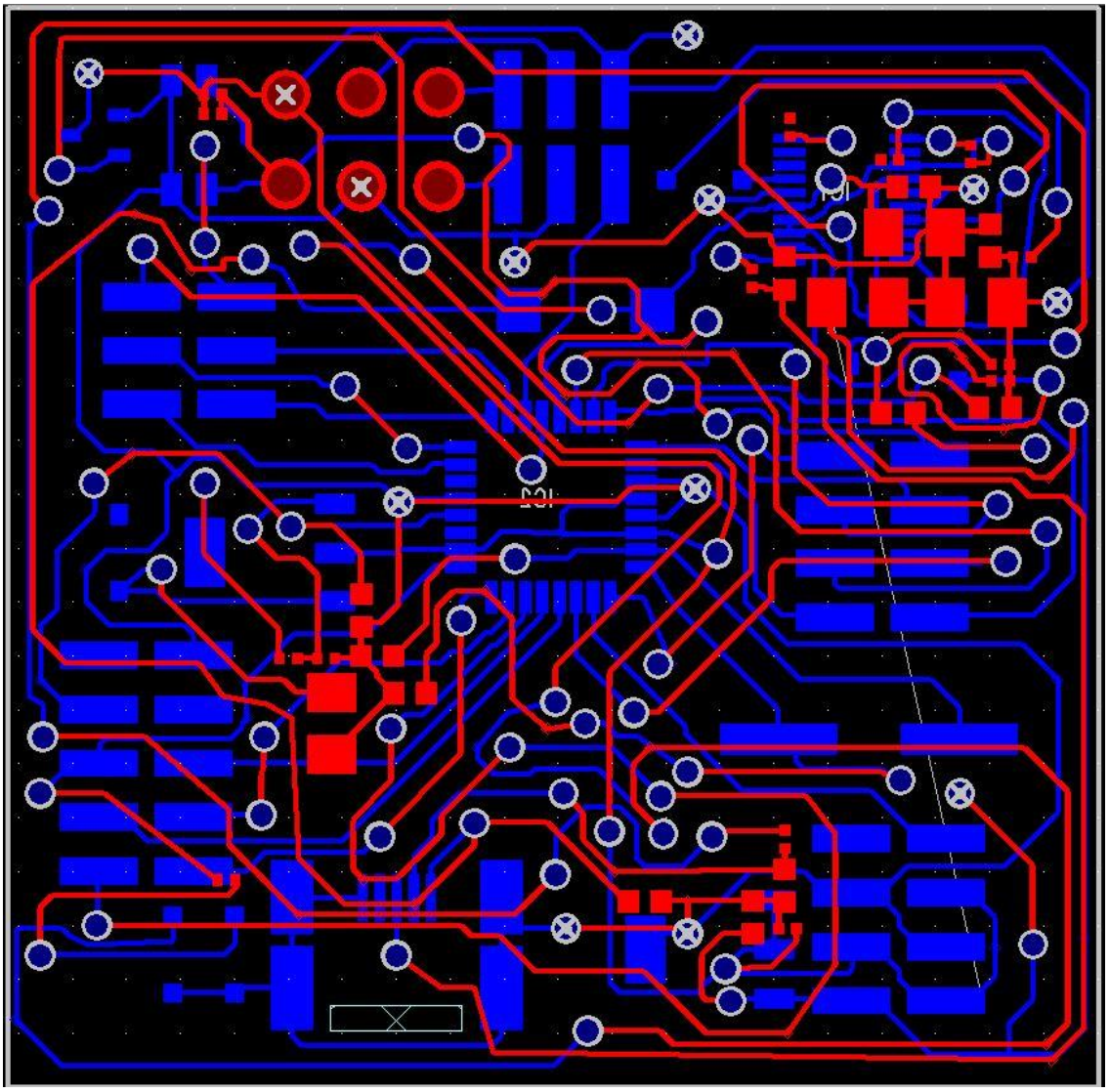


Figure 29 - Bottom view of the completed board.

A note about routing for this project

Technical difficulties were encountered starting when the routing was approximately halfway completed. The issue appears to be that PADS is unable to connect to Mentor's servers. PADS layout could still be used; however, PADS designer was unable to open the schematic for this board. Once this bug was encountered error checking as well as dynamic routing were disabled, and traces and vias could be laid on top of each other or over the top of pads (see Figure 30, below). Due to this error, the traces were laid using "eyeball" approximations. Because certain features were disabled, when components needed to be moved, several right angles in the traces were unavoidable.

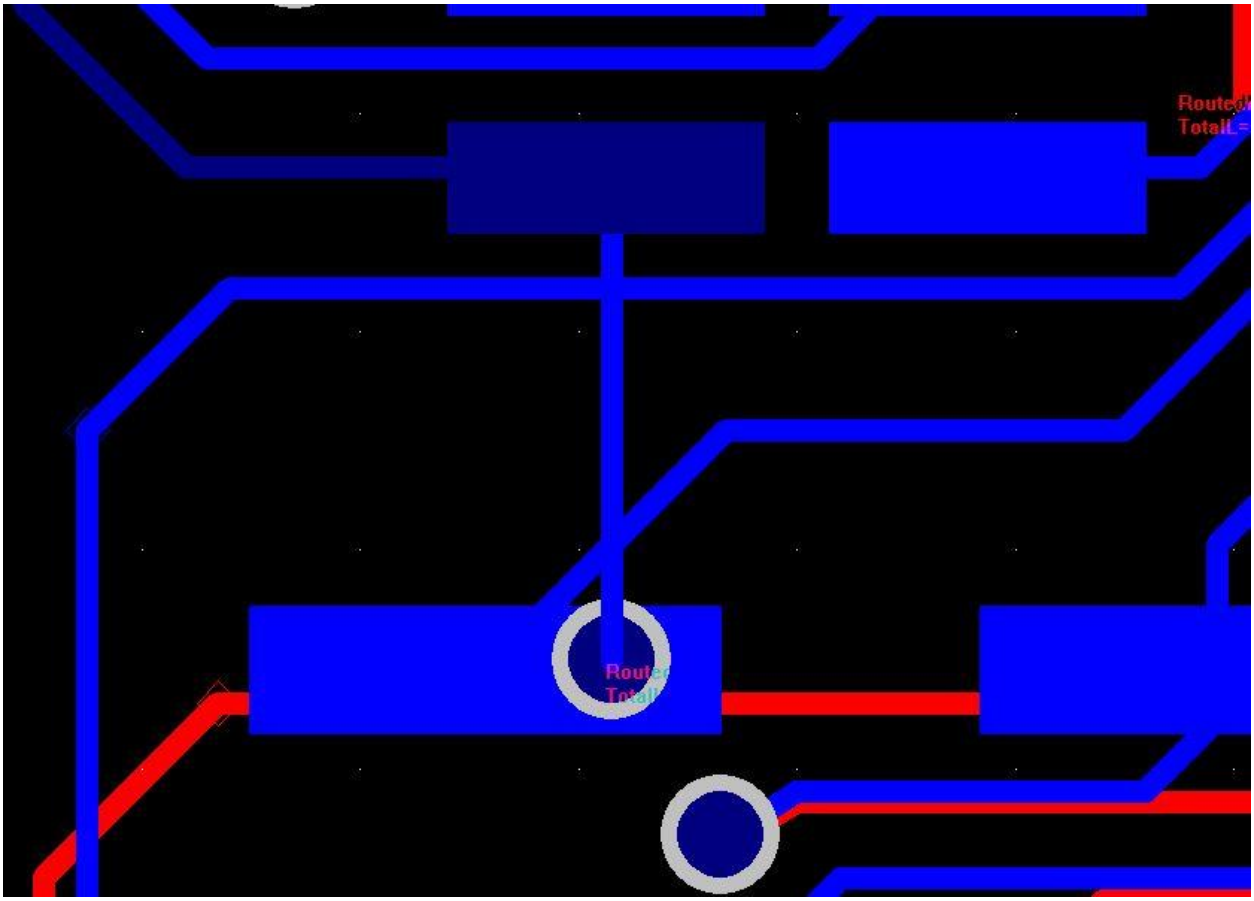


Figure 30 - Example demonstrating the effects of error checking being disabled. Vias could be laid on top of pads or traces, and traces could overlap.

Part 4: Run PADs DRCs and Check Plane Layer Thermals.

The design of the PCB is to be verified using the verify design tool in PADS. A clearance check, connectivity check, plane check, and fabrication check are to be ran to ensure completion of the board. During this step a number of routes and vias needed to be moved in order to have the appropriate clearences.

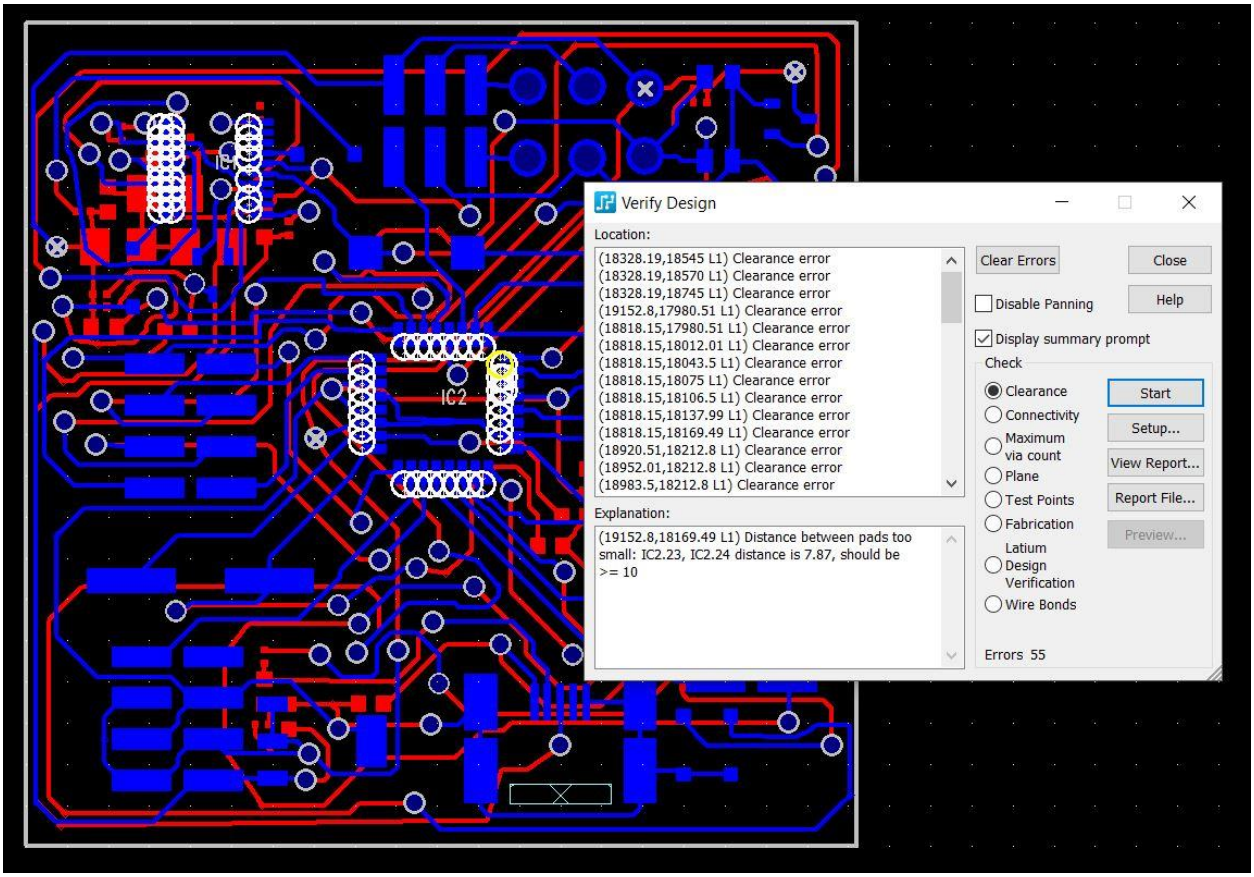


Figure 31 - A clearance report revealed a few errors, most of which were fixed except for errors resulting from the IC pads being to close together which would need to be changed in one of the document settings.

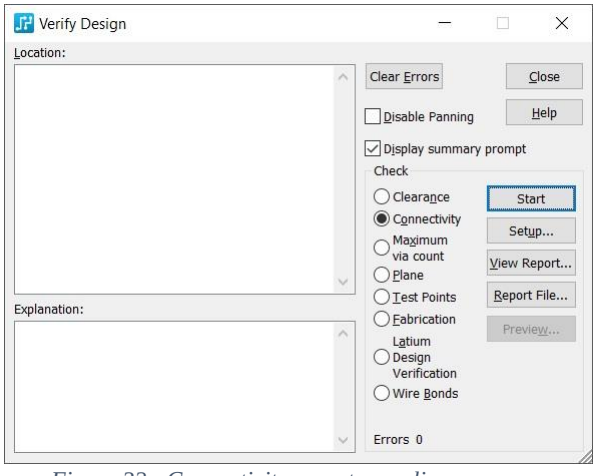


Figure 32 - Connectivity report revealing no errors

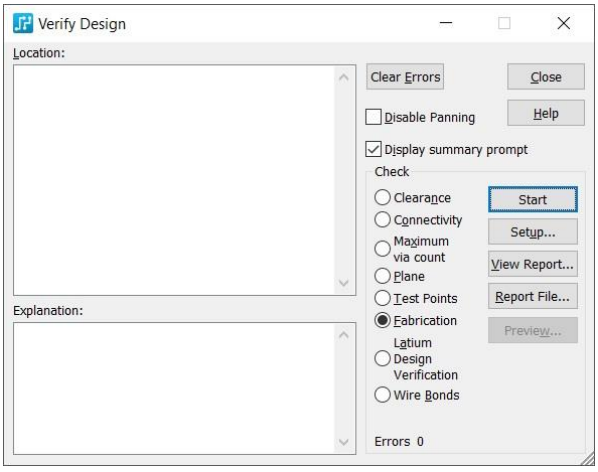


Figure 33 - Fabrication report revealing no errors

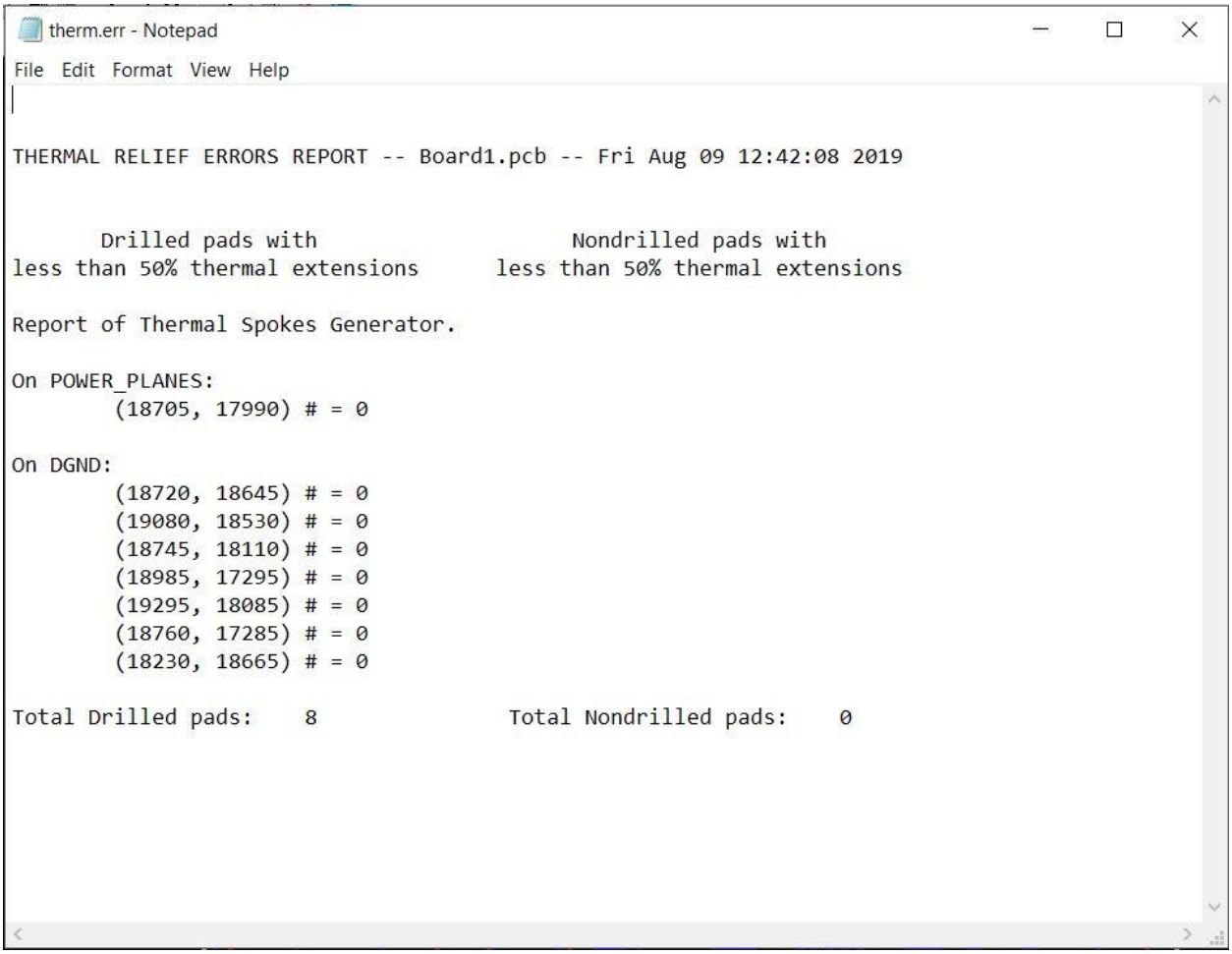


Figure 34 - Results of the plane report

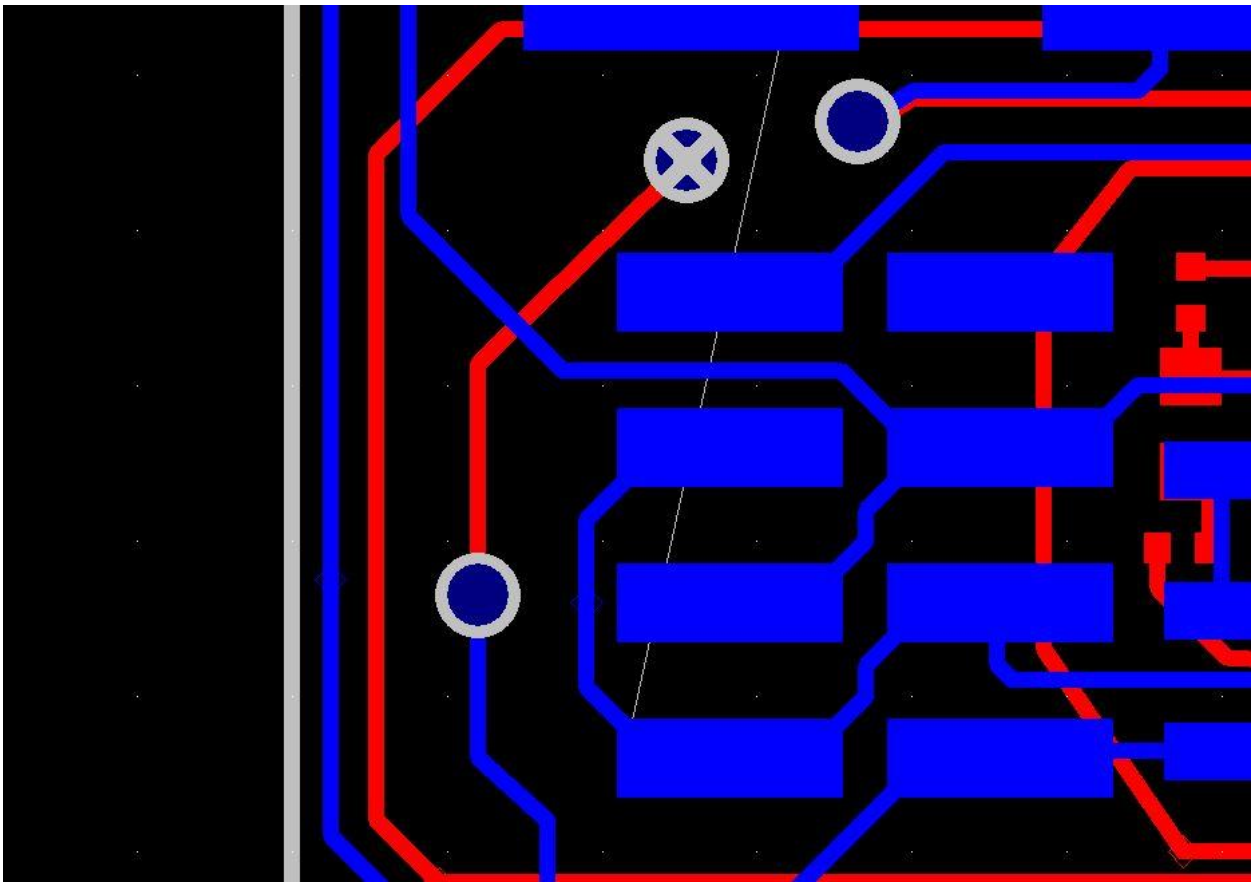


Figure 35 - Via connecting the net to the +3.3V plane

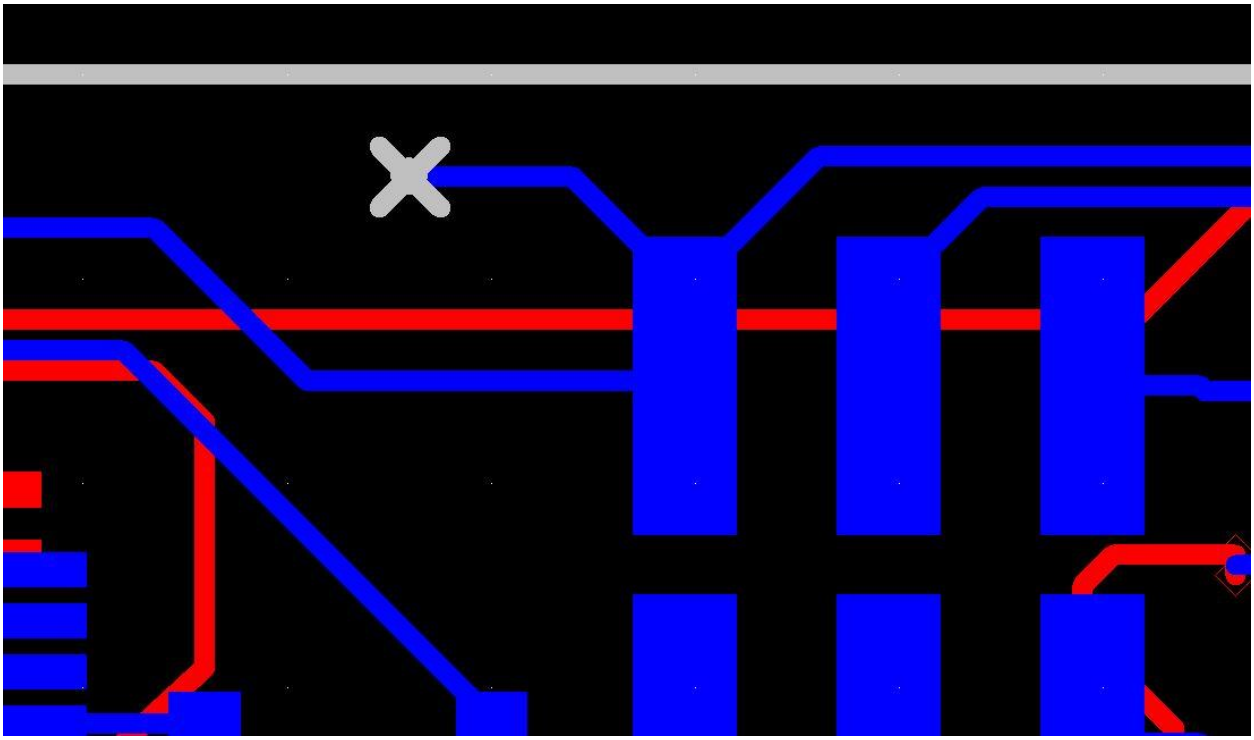


Figure 36 - Via connecting the +5.0V node to the power plane

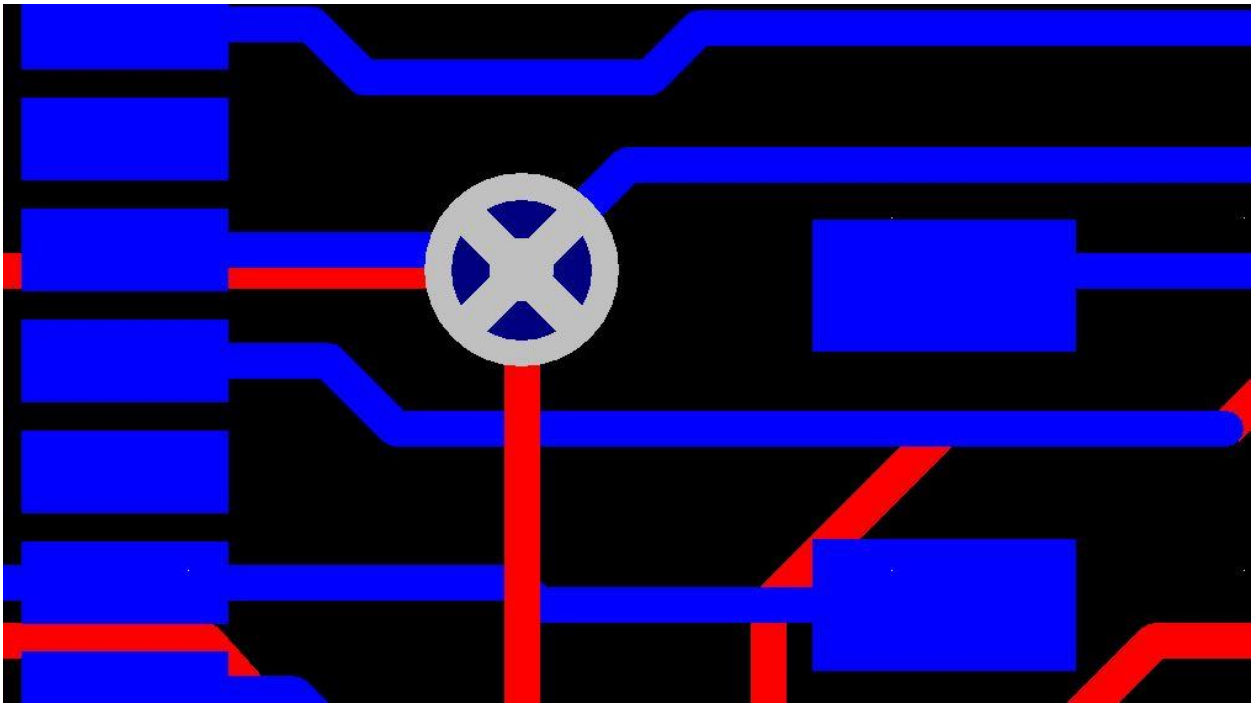


Figure 37 - Via connecting the net to the DGND plane

Conclusion

The layout of the board was completed in lab 3, but needed to be changed in order to make routing feasible using only two layers while staying within the 2”X2” parameters for the board. Vias needed to be added to the board in order to connect the correct nodes to their respective power and ground planes. About halfway through the project the program encountered a problem where it was unable to connect to Mentor’s servers, and therefore the schematic could no longer be referenced for the duration of routing. During this time certain features such as dynamic routing and smoothing were disabled, increasing the difficulty of the completing the project.

Lab 5

Introduction

The purpose of this lab is to put the finishing touches on the PCB and to get it ready to be sent over to a fabrication plant. This lab will generate gerber files which may be used to have a board fabricated.

Part 1: Renumber Reference Designators, Verify Silkscreen, and Back Annotate Schematics.

The cell size in the eco tool bar is to be renumbered to create a 1” x 1” cell size. This step cannot be completed however due to the eco toolbar being greyed out.

Figure 38 - The eco toolbar is greyed out

The designer’s name, student ID number, course number, and term/year are to be added onto the silk screen layer on the primary side of the board.

Figure 39 - Name and number on top layer

Due to errors connecting to Mentor’s servers, back annotation is not possible for this project.

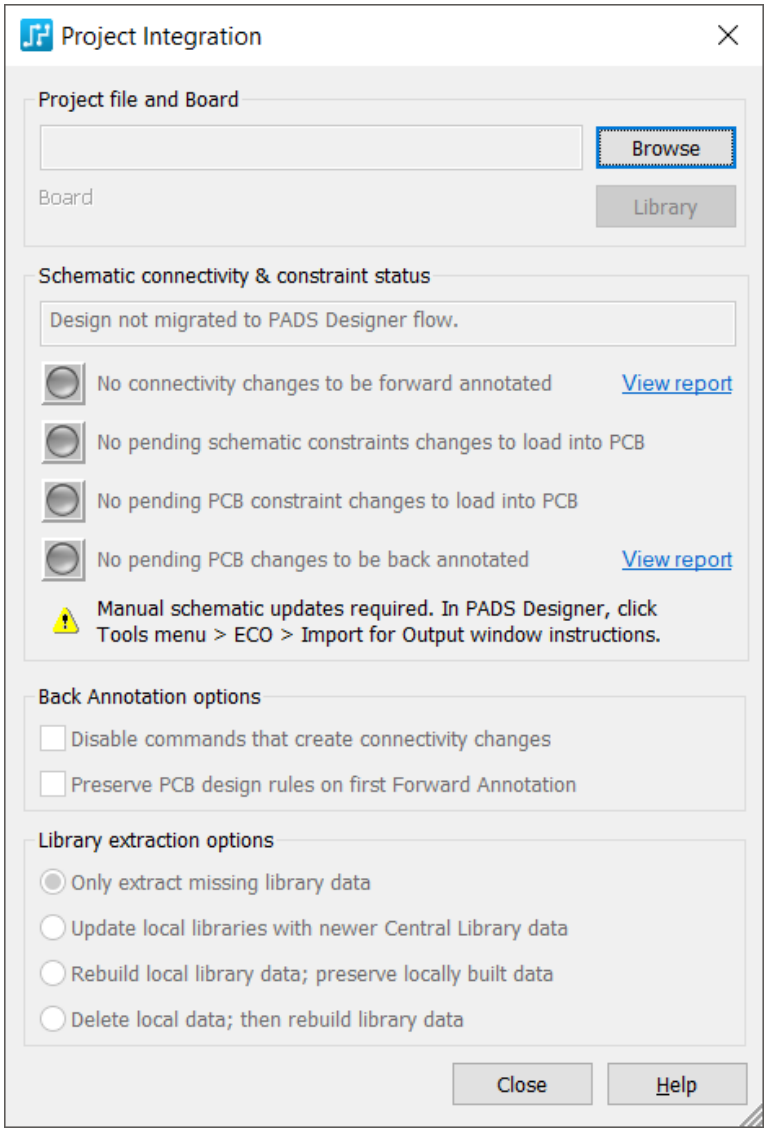


Figure 40 - Back annotation is disabled

Part 2: Generate and Review Gerber files.

Gerber files can be generated under file > CAM. A screenshot for each layer is to be taken.

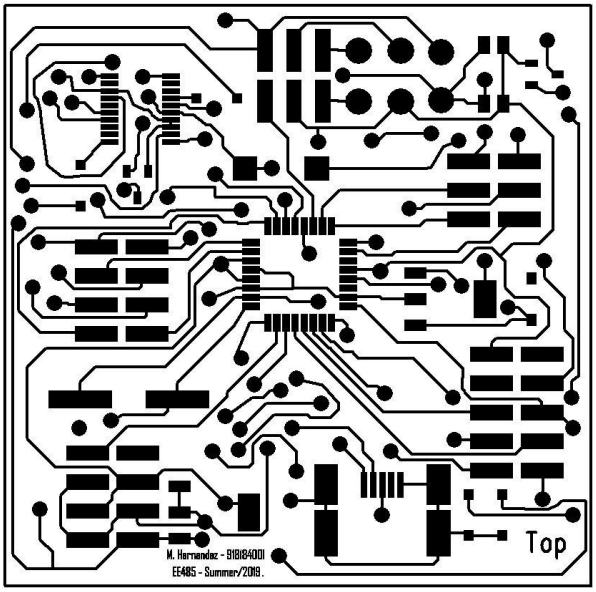


Figure 41 - Top Layer

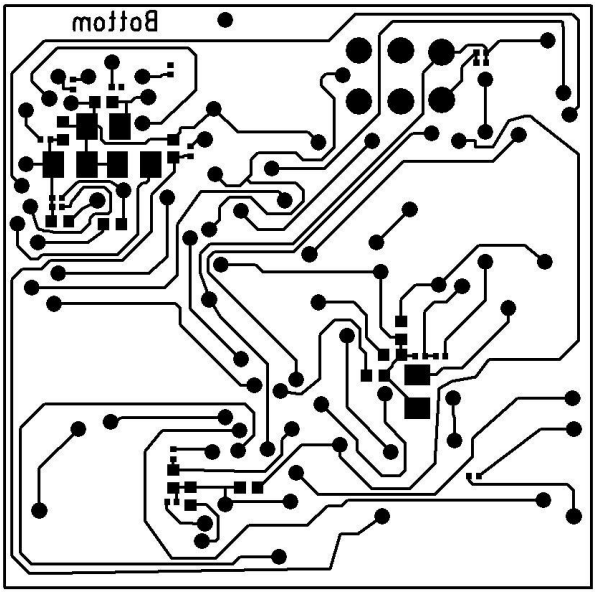


Figure 42 - Bottom Layer

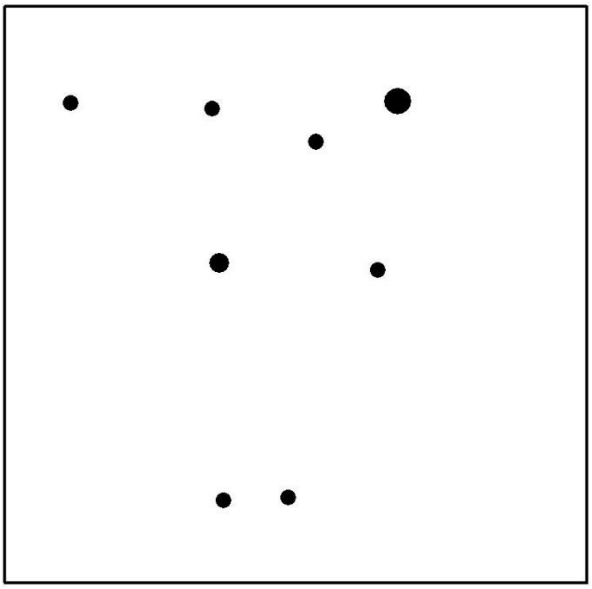


Figure 43 - DGND Layer

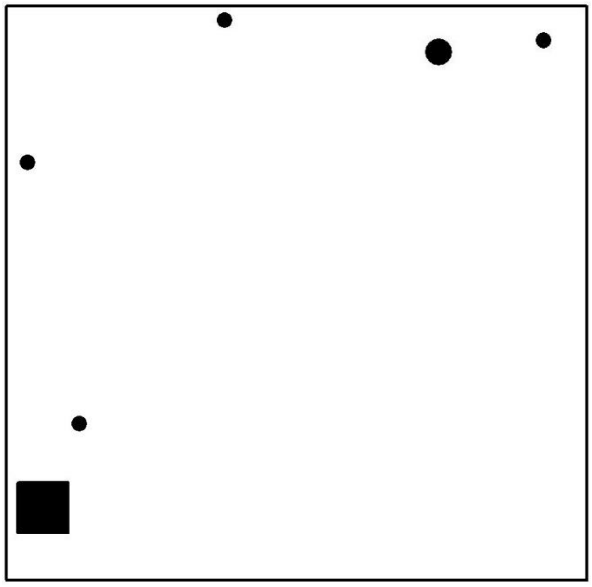


Figure 44 - Power Plane Layer

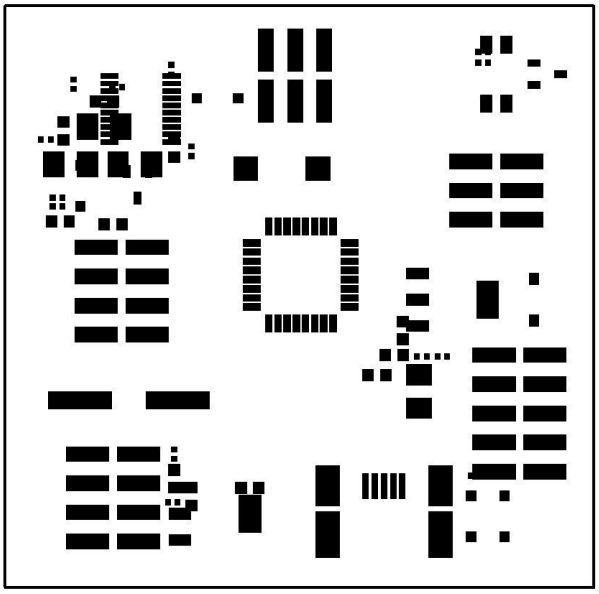


Figure 45 - Pastemask Top Layer

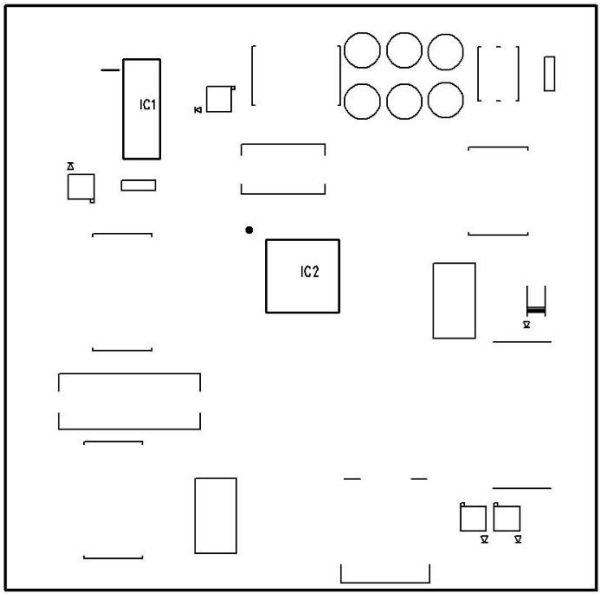


Figure 46 - Silkscreen Top Layer

Part 3: Generate NC Drill CAM document

The NC Drill file is to be generated, along with a drill drawing.

The drawing can be seen in Figure 47. A few of the drill holes are on the edge of the board.

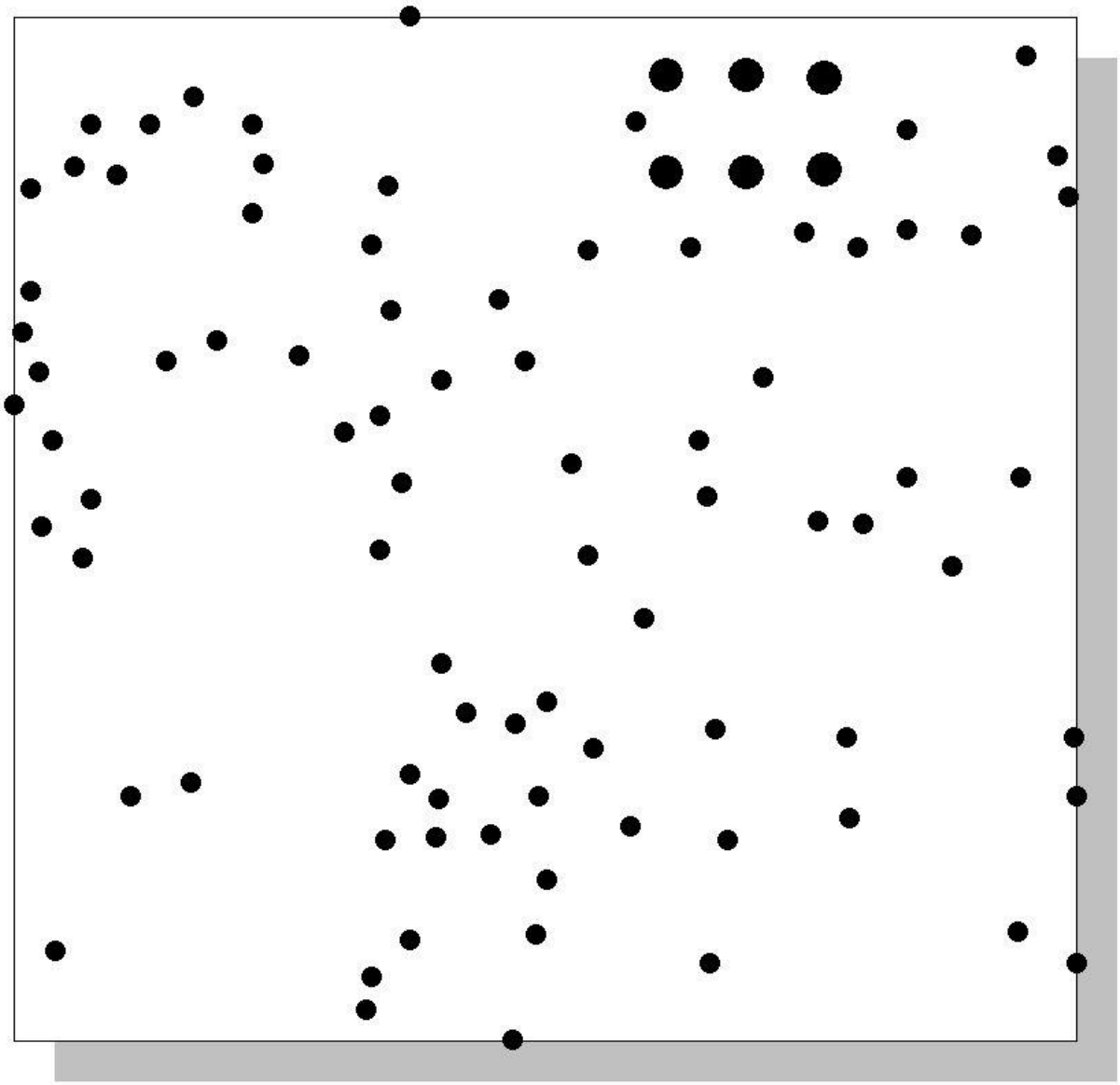


Figure 47 - NC Drill Drawing

Part 4: Add Board Outlines and Generate Gerber Files

Board outlines are to be added by using the menu as seen in the figure below. Once the gerber files are created, they are to be zipped into a single file for submission to a board fabrication house, such as Sunstone circuits (https://www.sunstone.com/).

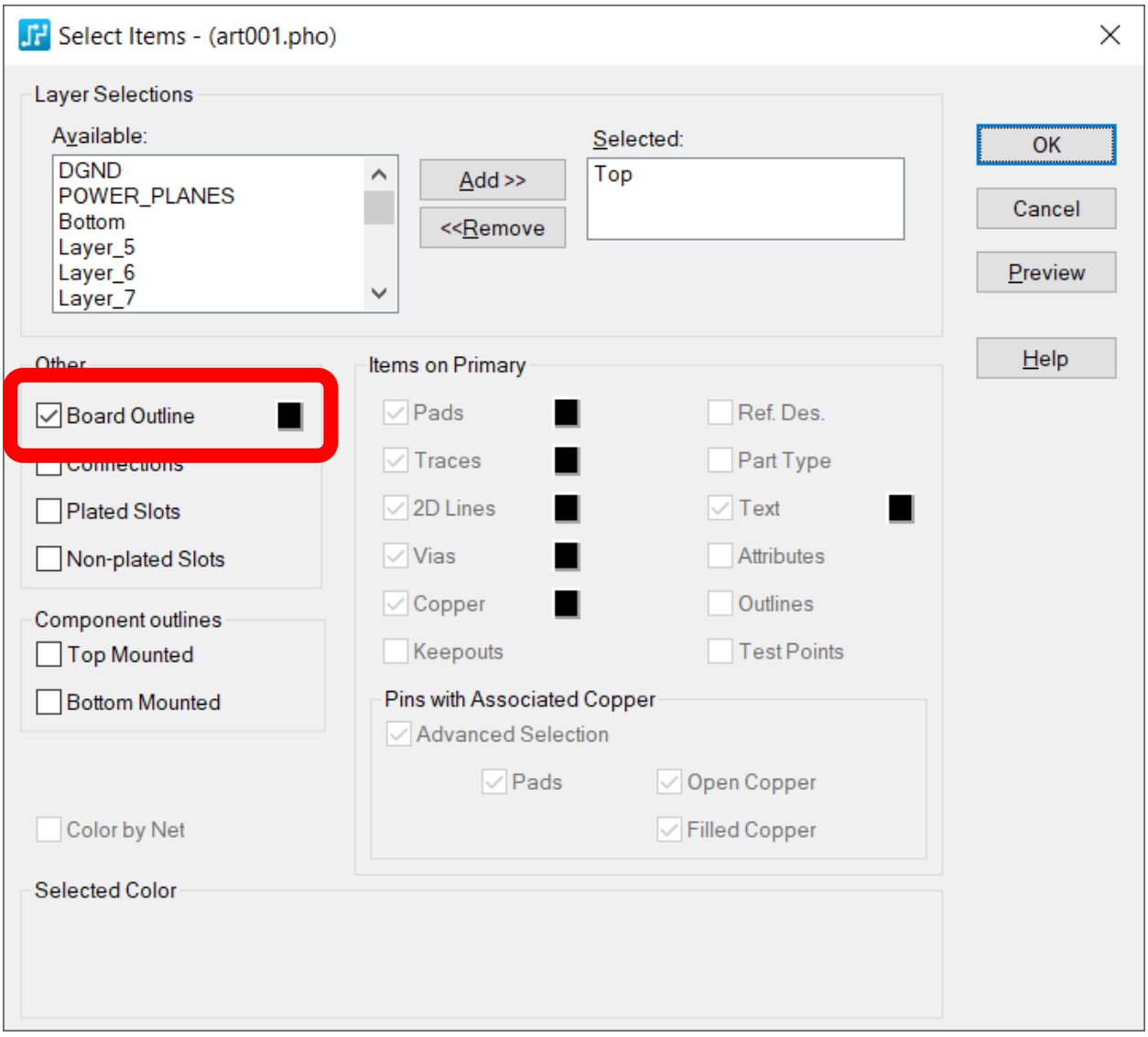
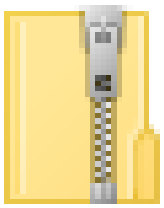


Figure 48 - Board Outlines



Project Arduino.zip

Figure 49 - Zipped gerber files; click on the image to open the files.

Part 5: ODB++

ODB++ files are to be exported for the PCB design.

An error was encountered which prevented the export of an ODB++ type document (see below), however an ASCII type file was able to be exported.

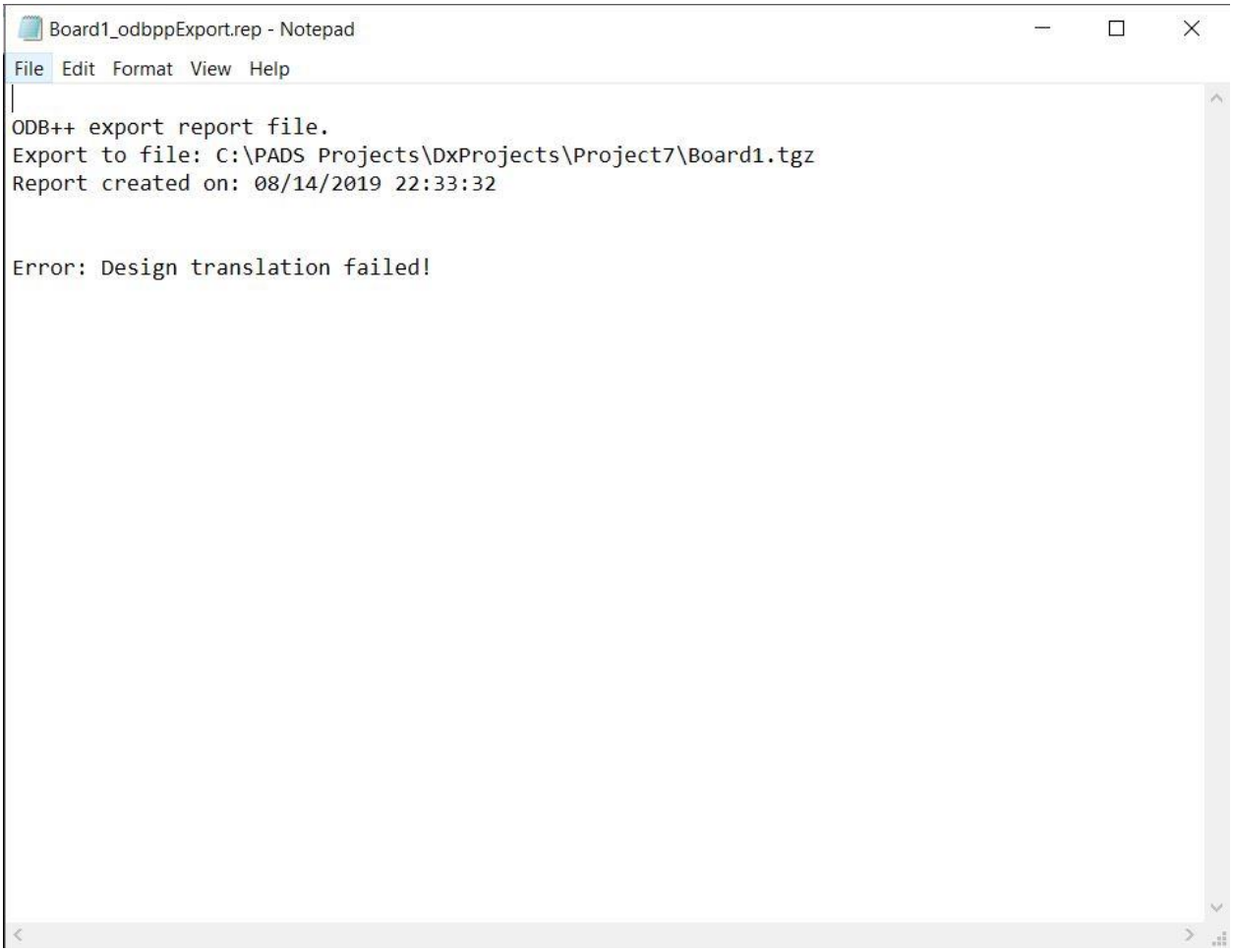


Figure 50 - ODB++ Error

Had the ODB++ file been exported, it would have been viewed using another product from Mentor called ODB++ Viewer 11, which can be seen in Figure 51. Had the file been able to be exported it would have been able to be referenced with the viewer by the directory that the file was saved to.

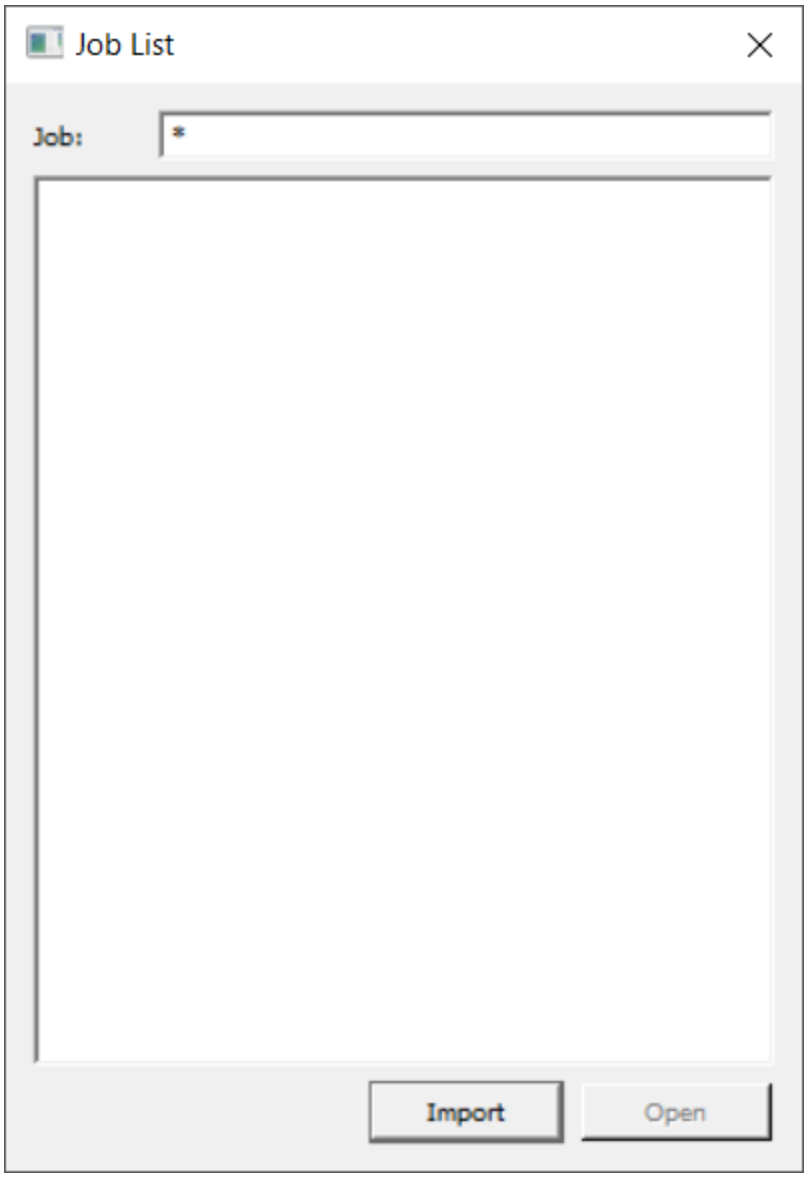
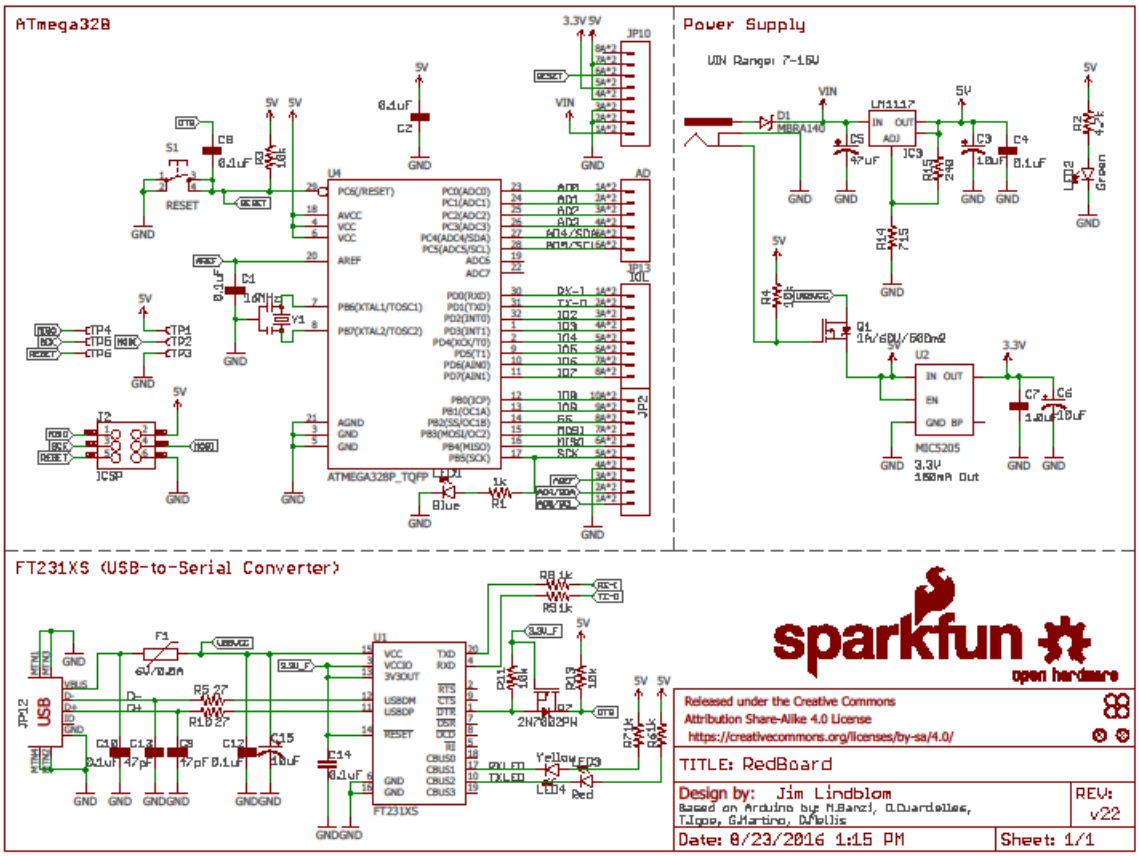


Figure 51 - ODB++ 11 Viewer

Conclusion

Text and images can be added to the silkscreen in order to make the board easier to work with later on after it has been fabricated. Gerber files where created so that the information for the board can be submitted to the fabrication house. There are a number of files that need to be prepared, including the top and bottom layers, the soldermask, the silkscreen, the pastemask, the ground plane and power planes. When the files are prepared, they must then be zipped together in order to be submitted. ODB++ is the latest standard which contains all of the information for the board in a single file.

Appendix A



Arduino Uno schematic

Downloaded from https://www.sparkfun.com/products/12757