

# **ECE-388**

Final Project: Pattern Generator

**Marcel Vieira**

December 8, 2020

I Marcel Vieira certify that this lab report consists only of work that is my own.

# Contents

<b>1</b>	<b>Abstract</b>	<b>3</b>
<b>2</b>	<b>Problem Statement</b>	<b>3</b>
<b>3</b>	<b>Procedures</b>	<b>3</b>
3.1	Materials . . . . .	3
3.2	Test Process . . . . .	3
<b>4</b>	<b>Discussion</b>	<b>14</b>
<b>5</b>	<b>Conclusion</b>	<b>16</b>
<b>6</b>	<b>Reflection</b>	<b>16</b>
<b>7</b>	<b>Appendix</b>	<b>18</b>

# List of Figures

1	Pattern Gen Multisim V1 . . . . .	5
2	Pattern Gen Eagle Schematic V1 . . . . .	6
3	Pattern Gem BOM 1 . . . . .	6
4	Prototype Output . . . . .	8
5	Pattern Gen Code V1 PG 1 . . . . .	9
6	Pattern Gen Code V1 PG 2 . . . . .	10
7	Schematic V5 . . . . .	12
8	Layout V5 . . . . .	12
9	Schematic V5 . . . . .	13
10	Layout V5 . . . . .	13
11	Schematic V1 . . . . .	18

12	Layout V1 . . . . .	18
13	Schematic V2 . . . . .	19
14	Schematic V3 . . . . .	19
15	Layout V3 . . . . .	20
16	Layout V4 . . . . .	20

# **1 Abstract**

This document describes the process taken in planning designing and constructing a pattern generator. The steps detailed includes identification and defining the problem, creating a bill of materials, creating simulations, prototyping, and drafting PCB design layouts. This document also details the problems encountered along with how they were solved or proposed solutions. Due to the project being incomplete this only details up to the creation of the PCB and the creation of a demonstrational prototype.

## **2 Problem Statement**

Use surface mount components and a ATMEGA 328pb to generate binary patterns at either 5V or 3.3V, with a user interface that allows a user to change the pattern as well as the logic level across 8 independent channels.

## **3 Procedures**

### **3.1 Materials**

- Multisim
- Eagle
- Web Browser
- Atmel Studio

### **3.2 Test Process**

In order to begin the process of creating a pattern generator the problem must be properly defined to set the guidelines for what the solution will be. To define the problem the constraints must be

laid out in an Engineering note book and clarified with the TAs and professor. They required that the ATMEGA328PB chip be used, and all components that are chosen must be surface mount. These two constraints are not enough to clearly define the problem so more must be added such as an ability to switch logic levels and a user interface that allows people to interact with the board and adjust the the output pattern. Now the problem can be defined with a problem statement, in this case it is done with the one above.

Begin brain storming possible ways to resolve the problem. The ATMEGA328PB is capable of outputting digital signals in patterns so one of the biggest issues is changing the 5V is outputs from 5V to 3.3V when desired. This could be done with a dc converter, using resistors or diodes to drop the voltage, or using MOSFETS to toggle the flow of electricity form whatever logic source is selected. The second issue is in getting the 3.3V form the 5V that will come into the board. This could be done using a voltage divider, a DC to DC converter, or a Zener diode. The third issue was how was the user interface going to be made. The interface could be made either using hard ware switches, buttons, and rotary encoders or use serial port to type in the pattern. The last problem was how was information going to be shown to the user? The information could be shown using as series of LEDs or on a screen.

The next step is to decide which brainstormed solutions will be used to solve the problem. For this project the user interface will use a serial input with a screen for the display. For the power logic level selection a Zener diode will be used to get the 3.3V with a sp3t switch to switch between the power states which will include 5V 3.3V and 0V. The voltage that comes out of the switch then will be passed to MOSFETS that will be toggled by the ATMEGA328PB chip.

Now that the plan has been completed the parts must be selected and tested in Multisim (figure 1). For this case the displays cannot be properly tested so they will be tested physically later on. The MOSFETS when input in Multisim showed that the were capable of using 5V at the gate to toggle both 5V and 3.3V. The Zener diode also showed it was a valid solution to getting 3.3V.

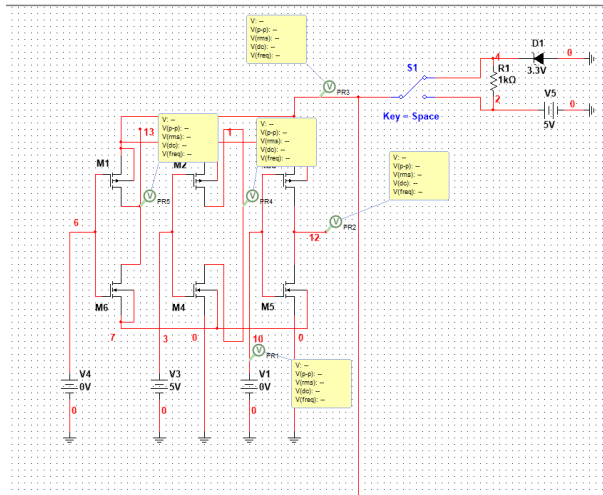


Figure 1: Pattern Gen Multisim V1

Then the parts that are required to build the circuit are searched in LCSC and Digikey. When a part was found it was inserted into Eagle (figure 2). Once all the parts were placed in Eagle a bill of materials is exported and opened in excel in order to add purchase information and remove irrelevant information (figure 3).

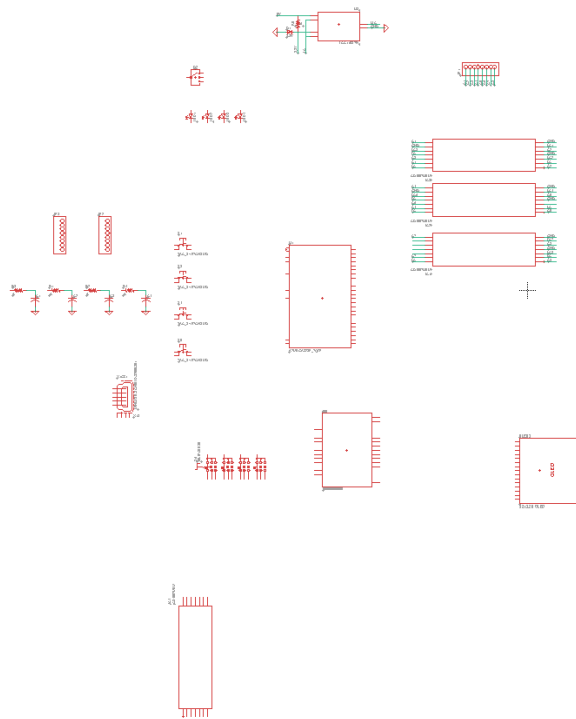


Figure 2: Pattern Gen Eagle Schematic V1

Qty	Value	Device	Package	Part	Description	Source	Price for 1	Price for 10	Price for 100
4	1uF	C-UC0805	C0805	C1,C2,C3,C4	CAPACITOR, American symbol	Digkey	0.15	1.05	6.93
4		LEDCHP-LED0805	CHP-LED0805	LED1,LED2,LED3,LED4	LED	Digkey	0.27	2.27	22.79
3		PNH0-USB-3.3MM	3MM	JP1,JP2,JP3	PN HEADER	Digkey	0.05	0.41	54.2
1	CUS-13	SWITCH-SPST-080	SWITCH-SPST_080		Single-Pole, Throw (SPST) Switch	Digkey	0.72	6.96	54.88
1	422KCV3	ZENER-DIODES0723	0723	D1	Z Diode	Digkey	0.33	7.33	73.33
5	5.1k	R-US-0805	0805	R1,R2,R3,R4,R5	RESISTOR, American symbol	Digkey	0.1	1.10	11.10
1	32x18mm	OLED-L98W450C	OLED-V6-283275	OLED1	Micro OLED	Stock Sheet			
1	DRL1-H9418P20	DRL1-H9418P20	56		Rotary Encoder	Digkey	1.16	10.75	79.55
1	ATMEGA328P-TQFP	ATMEGA328P-TQFP	TQFP32-08	U1	Popular 328P in QFP	Digkey	1.34	13.4	132
4	CD4051UBM	CD4051UBM	SONC277600K175	IC1,IC2,IC3,IC4	Texas Instruments CD4051UBM Inverter, 3.3V, 14-Pin SOT	Digkey	0.65	3.83	28.58
1	MC72200-105	MC72200-105	SONC277600K2-112		MC72200 Series 5.5V, 4-Channel USB to UART Serial Convert	Digkey	1.96	19.4	147
1	MINI-USB-SCHIEP-3-12005-301	MINI-USB-SCHIEP-3-12005-301	X1		MINI USB B Connector	Stock Sheet			
4	TAC_SWITCH0805	TACTLS-0805CH	15,15,15,15		Momentary Switch	Stock Sheet			
1	1CC1100TR	1CC1100TR	SONC2549952N47	4N	Solid state relay	Digkey	3.84	38.40	384.00

Figure 3: Pattern Gem BOM 1

After the parts were selected and the circuit was successfully simulated it was time to begin construction of the prototype. The prototype allows for the circuit to be tested prior to having everything on the PCB. The use of a prototype especially one on a bread board allows for parts to be more easily isolated and replaced. The prototype is constructed based on the circuit design used in Multisim but the prototype is broken into sections to test the functionality of all the individual smaller parts prior to testing the circuit in its entirety. The prototype consisted of a voltage regulator, a switching circuits made with 2 PMOSs and an inverter, and a set of CMOS inverters.

The first test conducted was connect a voltmeter to the the voltage regulator and check if the proper voltages are produced. The expected values were 5V at the input and 3.3V at the output. This ensures that the voltage does not damage the circuit and sets a base line for the voltage to expect in the inverters.

The second test is to put 5V and 3.3V to the switching circuit use the voltmeter to check if it is outputting the correct input depending on what signal is sent to it. If a high is put into the circuit it should output 3.3V, if a low is put into the circuit it should output 5V.

The final test for the first prototype was to check if the CMOS inverters could produce the expected signal. To test this a clock pulse is put into the inverter and the response as well as the input are measured and compared with an oscilloscope. The result showed be the opposite of the input with the exception of the high being what voltage is put to the sources of the PMOSs in the CMOS inverters.



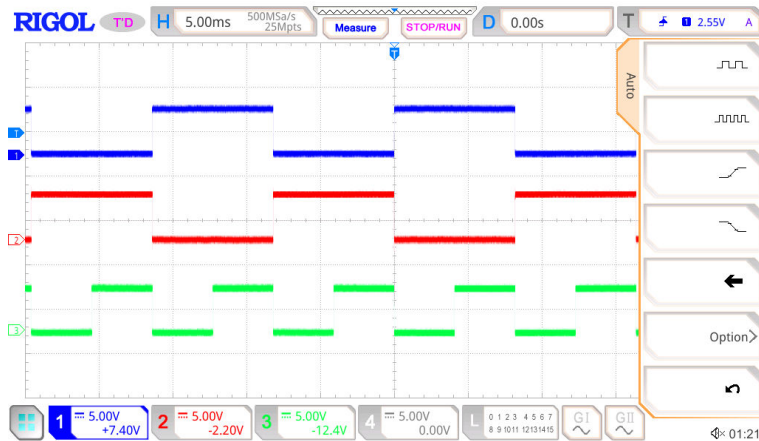


Figure 4: Prototype Output

After taking notes on the first prototype the second was built. This prototype is focused on the user interface. This prototype consists of a rotary encoder with a built in button, several LEDs, and a 328pb. For this prototype code was written in Atmel Studio. The code for the prototype used various interrupts with 2 pin change interrupts being used to determine the direction the rotary encoder is turned on. There is also another pin change interrupts that is used to change the button press. This user interface was tested using debugging and visual observation to tell whether it is operational.

```

1  /*
2   * Pattern Generator.c
3   *
4   * Created: 10/30/2020 12:13:11 PM
5   * Author : pphantom
6   */
7
8  #define F_CPU 16000000
9  #include <avr/io.h>
10 #include <stdio.h>
11 #include <avr/interrupt.h>
12 #include <util/atomic.h>
13 #include <util/delay.h>
14
15 int rotorCheck();
16
17 static uint8_t temp = 0;
18 static uint8_t i = 0;
19 static uint8_t turned = 0;
20 static uint8_t clockwise = 0;
21 static uint8_t mode = 0;
22 static uint8_t buttonPress = 0;
23 static uint8_t bit = 0;
24 static uint8_t channelNum = 0;
25 static uint8_t voltage = 0;
26 static uint8_t channel[8] = {0x65,0x81,0xff,0xf6,0x82,0x80,0x34,0x75};
27
28 int main(void)
29 {
30     cli();
31
32     DDRA = 0b11111110;
33     PORTA = 0b11111111;
34
35     DDRC = 0b00000000;
36     PORTC = 0b11111111;
37
38     DDD = 0b01111111;
39     PORTD = 0b01111111;
40
41     DDRE = 0b11111111;
42     PORTE = 0b11111110;
43
44     PCICR = 0b00000111;
45     PCMSK0 = 0b00000001;
46     PCMSK1 = 0b00000001;
47     PCMSK2 = 0b10000000;
48
49     OCR1A = 0x3C08;
50     TCCR1B = (1 << WGM12);
51     TIMSK1 = (1 << OCIE1A);
52     TCCR1B = (1 << CS12) | (1 << CS10);
53
54     sei();
55
56     while (1)
57     {
58         switch (mode)
59         {
60             case 0:
61                 channel[channelNum] = (channel[channelNum] + rotorCheck()) & 0b00111111;
62                 PORTD--(channel[channelNum]);
63                 break;
64
65             case 1:
66                 channelNum = (channelNum + rotorCheck()) % 8;
67                 PORTD--(channelNum);
68                 break;
69
70             case 2:
71                 voltage = (voltage + rotorCheck()) % 2;
72                 if (voltage == 0)
73                     break;
74
75         }
76     }
77 }

```

Figure 5: Pattern Gen Code V1 PG 1

```

76         }
77         else
78         {
79             PORTD = 0b11011111;
80         }
81         break;
82     }
83 }
84 }
85
86
87
88 int motorCheck()
89 {
90     if(turned)
91     {
92         turned = 0;
93         if(clockwise)
94         {
95             return 1;
96         }
97         else
98         {
99             return -1;
100         }
101     }
102     return 0;
103 }
104
105
106
107
108 ISR(PCINT0_vect)
109 {
110     if(!turned)
111     {
112         clockwise = 1;
113     }
114     turned = 1;
115     _delay_ms(150);
116 }
117
118
119
120 ISR(PCINT1_vect)
121 {
122     if(!turned)
123     {
124         clockwise = 0;
125     }
126     turned = 1;
127     _delay_ms(150);
128 }
129
130
131
132 ISR(PCINT2_vect)
133 {
134     if (!buttonPress)
135     {
136         mode = (mode + 1) % 3;
137     }
138     buttonPress = ~buttonPress;
139 }
140
141
142
143 ISR(TIMER1_COMPA_vect)
144 {
145     cli();
146     i=0;
147     temp = 0;
148 }

```

Figure 6: Pattern Gen Code V1 PG 2

The user interface should have LEDs that start off displaying the current binary signal, this is referred to as mode 1. In mode 1 the rotary encoder changes the binary value going up when turned clockwise and down when turned counter clockwise. When the button is pressed it enters mode 2 which allows the user to view the the current channel number in binary displayed on the LEDs. Mode 2 also allows for the user to switch between channels using the rotary switch. When the button is pressed again it goes into mode 3 which simulates the display of the voltage settings which is adjustable with the rotary encoder as well.

Once the final design has been decided on the the design must properly be translated into a schematic that will be uploaded to Github and checked by Tim Chase. The schematic must contain all parts that are to be used, in this instance that would be the micro USB, the CMOS chips, the serial chip, the LEDs, the OLED display, and the rotary encoder. Power must flow in the schematic from the left to right and be clearly labeled. The schematic must finally pass all ERC checks with no errors or only errors approved by Tim Chase.

Once the schematic is approved by Tim Chace the board lay out can then be worked on. The board layout must include all parts, ground planes, fiducials to show the alignment of the board, and must pass all DRC checks in the file Dirty8TS.dru. On the back of the board the board must have a name, date, version number, and the Git repository link printed on the back of the board.

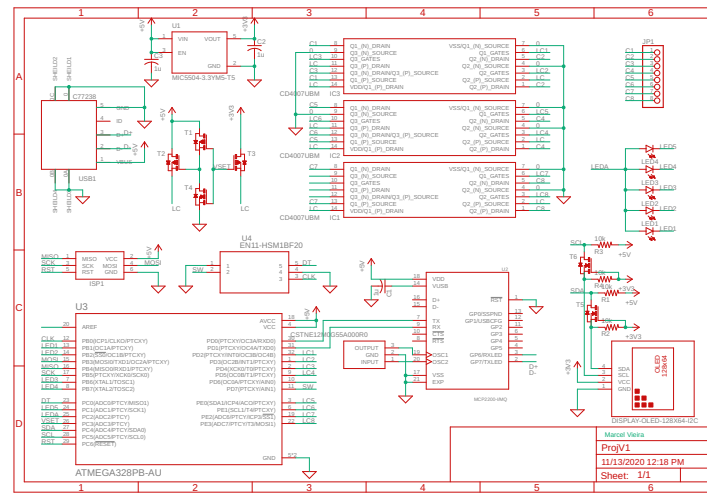


Figure 7: Schematic V5

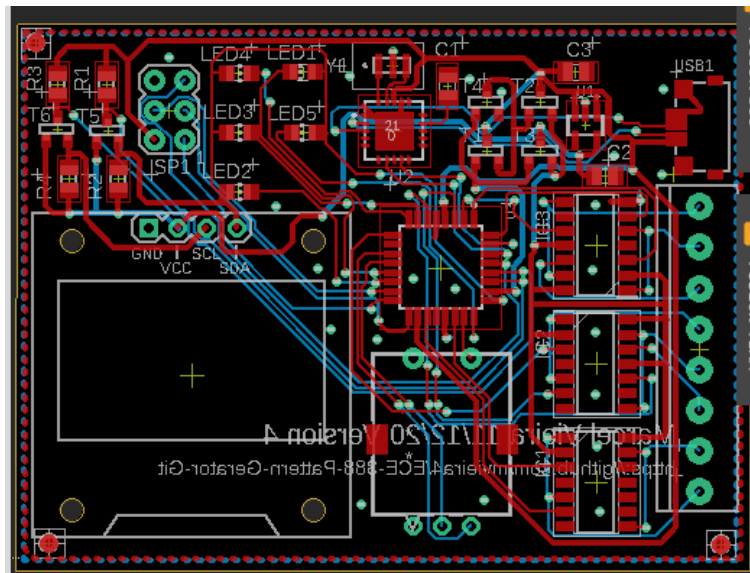


Figure 8: Layout V5

After the board received approval from Tim Chase the Gerber files were sent to JCL PCB to design create the board. Now that the boards were being manufactured an acceptance test is needed to determine whether or not the end result performs to a desirable standard. In this case the test will be how well does the device hold up to the problem statement. Does it have 8 independent channels that can be changed, does it have a UI that makes sense, and can it switch voltages? This will be tested using an a multimeter and a digital discovery which can read voltage patterns in both 5V and 3.3V.

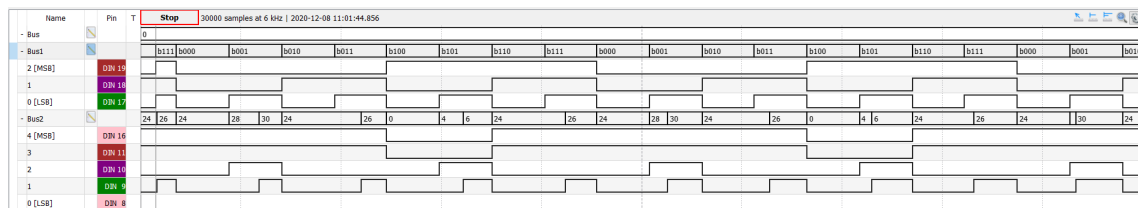


Figure 9: Schematic V5

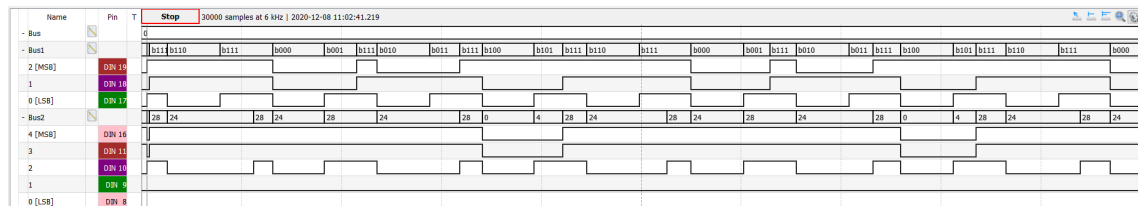


Figure 10: Layout V5

Because of complications with the manufacturer the stencil arrived in time but the PCBs did not making board population impossible. To work around this a fully functional prototype was created that contains similar parts to show how the board was intended to work. After it is built test the prototype in place of the final board.

## 4 Discussion

Originally the circuit was going to use a voltage divider to drop the voltage to 3.3V, and N-type MOSFETS to toggle to logic voltages on and off but they presented several issues in the Multisim simulations. For the voltage divider applying the 3.3V to a load changed the resistance on altering the voltage and because the resistance was going to be difficult to predict the Zener diode was used. For the N-type MOSFETS they worked perfectly but each of the MOSFETS was their own chip, but after a suggestion by Tim Chace CMOS inverters were used instead which proved to be just as effective and required 3 times less chips saving space. These test processes lead to the revision of the bill of materials 3 times.

For prototyping all tests ran as expected proving that all systems individually are operational. This is with the exception of the linear regulator, the MCP2200 serial chip, and the OLED display. For the display and serial chip they have not come in time to be properly tested, but the voltage regulator was tested and failed. The failure at first was due to the wrong chip being supplied without a part number which meant that the wrong data sheet was used to wire the chip. This in turn damaged the chip and because there was only one further tests could not be done. An attempt was made to test a complete prototyping but there were complications with the 328pb which made it unable to be identified by any computer. This resulted in the 328pb not being able to be programmed. This caused the loss of another test session. Fortunately Ben Vial was able to repair the board so even if the issue occurs again there is a known solution for the problem now.

In creating the first schematic a new USB and an OLED were added to the schematic that originally only contained parts. After wiring the schematic several issues presented themselves.

First the resonator that was attached to the previous schematic was the wrong size. The second issue was that there were no power labels connected. Once these corrections were made even more issues appeared. These issues were brought up by Ben Vial that let students know that the OLED display runs on 3Vs while the 328pb runs on 5V in order. To eliminate issues with communication between the 2 chips he recommended incorporate a MOSFET level shifter. After fixing this the USB had to be replaced with one that had a similar foot print to what was in the stock sheet.

Once the schematic was all set all focus was paced on completing the layout. The issues that were worked on for the layout focused more on labeling and having correct foot prints for all components. Fiducials were also added to the layout to indicate proper orientation. While replacing one of the components in the layout it disappeared but still appeared in the schematics the rest of the time working on the boards were centralized around attempting to resolve the issues. After the third attempt the entirety of the board layout was populated by the correct parts.

After the board was cleared to send to the manufacturer by Tim the Gerbers were sent out. The stencil came in on time but the supplier sent the stencil but not the PCB. Without the PCBs I could not build and test the finished product. To resolve this problem a fully functional prototype was built in its place.

Due to not having enough time to work with the OLED or serial chip the prototypes UI only used the rotary encoder and LEDs. Although it was not ideal it was still in scope for what was acceptable.

To test the prototype for the demo the first step was to use the digital discovery to show the pattern it generated. The first 3 channels in the demo were used to number and keep track of the order of each bit in the pattern.

For the 5V setting the pattern show was perfect exactly matching what was input. When it came to using the UI that worked exactly as intended. The UI allowed for the changing of channels, changing of the selected channel pattern, and changing of voltage.

For the 3V setting the pattern shown was similar but did not match what was expected. This is hypothesized to be because of the final board being intended to have a 3.3V regulator while this



## User Acceptance Test Template

Project Name: Pattern Generator

Project Sponsor:

Service Owner:

Project Manager: Marcel Vieira

Document Date: 12/08/20



### 1. UAT Scope (In Scope – Out of Scope)

UAT - In Scope	UAT - Out of Scope
Changes Channel Changes Voltage Changes Pattern	Delay Frequency Usability

### 2. UAT Assumptions and Constraints

#### UAT Assumptions

Board will turn on  
No connections will burnout  
Everything is operating a correct power  
No components are damaged  
Digital discovery not damaged  
Digital discovery will be able to read pattern in both settings

#### UAT Constraints

There must be more than 3 channels  
First 3 channels will start with clock pattern to pace remaining channels  
Channels must be connected to digital discovery input

### 3. UAT Risks

Description	Probability High Med Low	Impact High Med Low	Mitigation
Disconnect peripherals Wrong code may be uploaded Digital discovery not reading 3V	Low Low High	High Low High	Secure connections Upload code prior Secure connections

### 4. UAT Team Roles & Responsibilities

Name	Roles	Responsibilities
Marcel Vieira	All roles	Make project a success

### 5. UAT Entry Criteria

ID	Criteria
5.1	Connected to computer Connected to digital discovery Code is running Multimeter connected Waveform software must be open and configured

### 6. UAT Requirements-Based Test Cases

ID	Test Cases
6.1	Start board: Expected: First channel oscillates every 4 bits, the second oscillates every 2, and the 3 oscillates every bit

## User Acceptance Test Template

Project Name: Pattern Generator

Project Sponsor:

Service Owner:

Project Manager: Marcel Vieira



Document Date: 12/08/20  
Should start in Voltage selection mode with 5 LEDs blinking

## 7. UAT Test Results

ID	Test Cases	Pass/Fail	Tested By	Date Tested
	Mode Switch Case: Push button Expected: Mode change indicated by change in LED behavior	Pass	Marcel	12/08/20
	Voltage Switch Case: Enter voltage mode Turn nob twice clockwise Turn nob twice counterclockwise Expected: Mode should switch from 5 to 3 to 5 to 3 to 5 This will be indicated by the number of blinking light going from 5 to 3 as in the order above Oscillation pattern should remain the same	Fail	Marcel	12/08/20
	Patter Switch Case: Enter pattern mode 4 times Turn nob clockwise 4 times Turn nob counterclockwise Expected: The oscillation pattern should change The binary value represented by the lights goes up The oscillation pattern then returns to normal The binary value represented by the light goes back to normal	Pass	Marcel	12/08/20
	Channel Switch Case: Enter channel mode Turn nob less than 8 times Enter pattern mode Turn nob in any direction 2 times Expected: The light will move into a new position The oscillation pattern of the selected channel should change	Pass	Marcel	12/08/20

## 8. Document Signatures

Role	Name	Signature	Date
Service Owner			
Project Manager	Marcel Vieira		12/08/20
Others (if needed)	<i>Add more rows if needed.</i>		

prototyping uses 3V. This presents an issue because the logic range for the digital discovery is from 3V to 5V which means that because there wont be a consistent 3V.

## **5 Conclusion**

For the bill of materials it is completed but it still may change because no solution has been found to electronically switch between the sources being used for the logic levels. There are still issues with the voltage regulator and a full prototype replicating the entire system must still be tested. The schematics and board layout have been completed and have been approved. The board design was sent to the manufacturer but the PCBs did not arrive in time. To be able to demo the project still a fully functional prototype was built and tested. Although there were issues with testing the 3V setting the 5V setting worked flawlessly.

## **6 Reflection**

The planning and simulation had no major issues. There only three minor issues. The first was that it was unclear early on what the pattern generator was and what it was supposed to do. The second was that simulating the circuits proved difficult at first because the necessary chips were not in Multisim so they had to be built out of equivalents that had to be found in Multisim. The third issue was that no clear way to use the MOSFETS to switch between logic sources was found.

In the prototyping stage one of the major issues was a lack of parts to work with which caused a lot of time to be lost and a lot of confusion. This problem was difficult to avoid due to the current pandemic and the restrictions placed as a consequence. Another issue encountered with prototyping was time management. Due to issues with the regulator a majority of lab time was spent around trouble shooting it. Because so much time was lost the rest of the test were done quickly and pictures were not taken. This is why there are no images of the first and second prototyping stages.

The process of creating a final schematic and board lay out came with unexpected issues. One

of the major issues was communication. Due to the time constraints being so tight it was difficult to make changes and get the revised quickly. The biggest issue was due to a lack of understanding of what was on the write up that detailed the requirements of the schematic and layout. To solve this issue other students were consulted and they were able to clarify what was needed. In the future other students will be contacted sooner in the event of confusion.

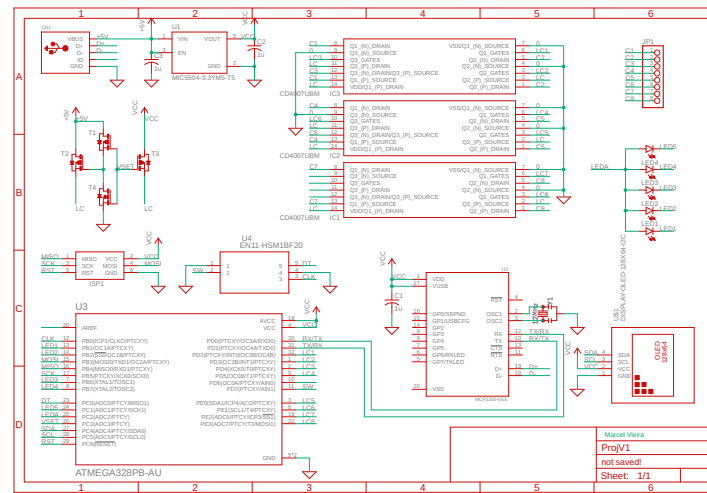
Creating the final board had many issues. One of the first issues was that the PCBs did not arrive on time. This was due to not getting the board design out in time. This happened because the Github repository did not clearly show which design was the final design. This caused confusion and lead to the board getting approved later than it should have been. To improve this it would have been best to better label the final design version. Had the PCBs and stencil arrived sooner the pick and place would have been used to populate a majority of the board and the OLED and pins would have been hand soldered.

Because the PCB did not arrive a fully functional prototyping was built but this prototype had a 3V regulator instead of a 3.3V regulator. This meant that the digital discovery used to test it could not accurately predict highs. Had the final PCB been constructed this would have most likely not have been the case. This could have been resolved by using an oscilloscope or having a 3.3V regulator.

For the project in its entirety the major issues had to due with a lack of time. This was because too much time was spent in the early stages of design and the equipment that was needed to test the design were on campus. This could have been resolved with more of a realization of time sensitivity due to the pandemic.

Communication also posed and issue which drained plenty of time as well. One major solution to this was to take more time and talk to more students about the formats required for the Github. Also there was an issue were Github notifications were being sent directly to spam which could have been avoided by simply checking spam and making sure that all communications went through efficently.

## 7 Appendix



11/10/2020 12:00 PM C:\Users\pphantom\OneDrive\Umass\ECE-388 Work\Lab\ECE-388-Proj-V1\ECE-388 Pattern Generator\Git\ProjV1.sch (Sheet: 1/1)

Figure 11: Schematic V1

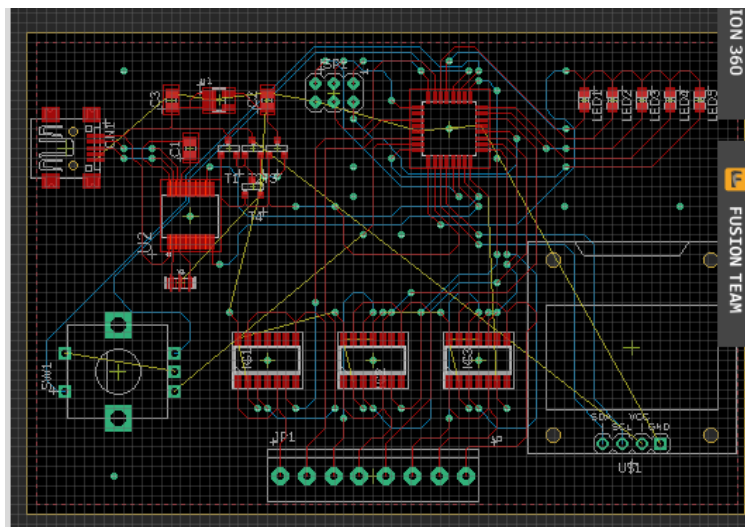
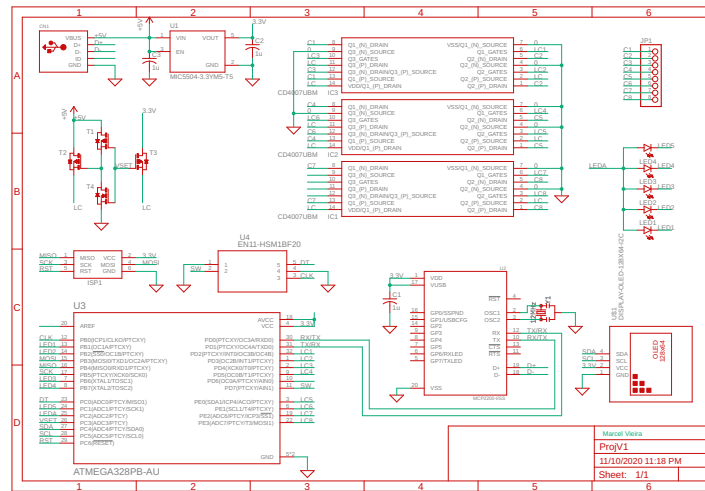
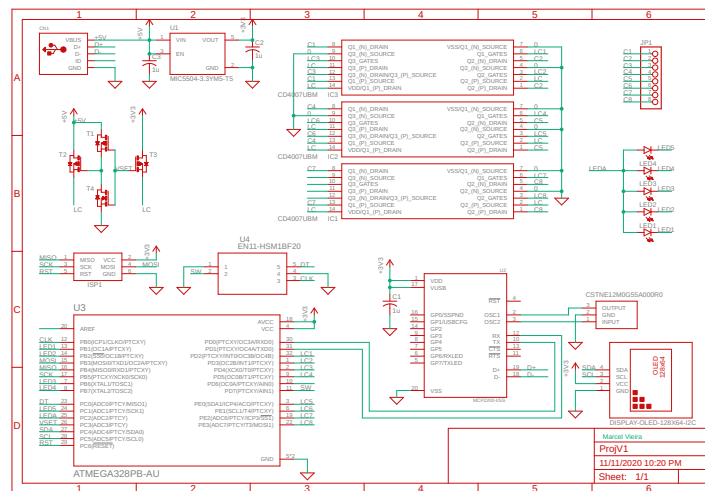


Figure 12: Layout V1



11/10/2020 11:33 PM C:\Users\phantom\OneDrive\Umass\ECE-388 Work\Lab\ECE-388-Proj-V1\ECE-388 Pattern Generator Git\ProjV1.sch (Sheet: 1/1)

Figure 13: Schematic V2



11/12/2020 7:11 AM C:\Users\phantom\OneDrive\Umass\ECE-388 Work\Lab\ECE-388-Proj-V1\ECE-388 Pattern Generator Git\ProjV1.sch (Sheet: 1/1)

Figure 14: Schematic V3

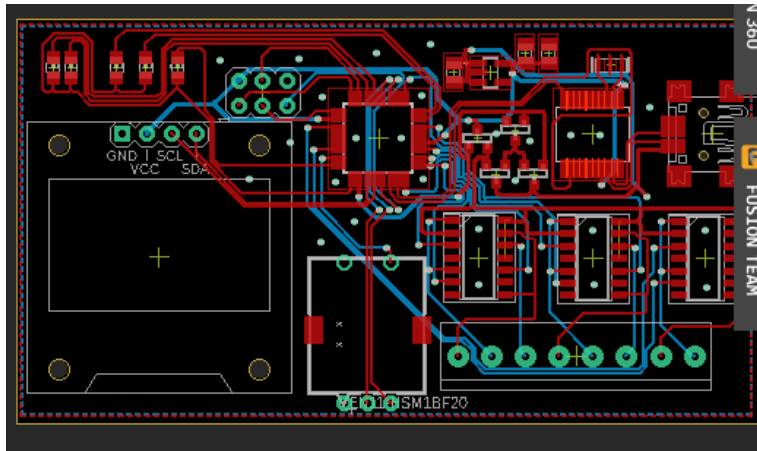


Figure 15: Layout V3

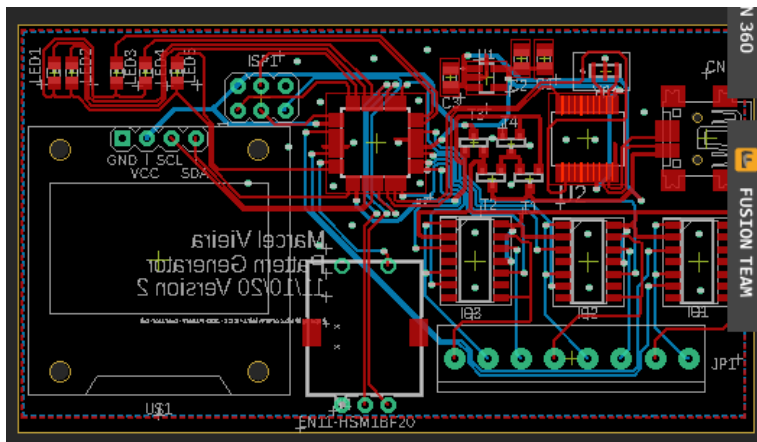


Figure 16: Layout V4

# PROJECT POST-MORTEM

**Project Title:** Pattern Generator

**Date Prepared:** 12/08/20

## Project Overview:

The goal of this project was to create a PCB that used the ATMEGA 328PB chip to create binary sequences at both 3.3V and 5V with a user interface that at least made sense. There were several ways that were going to be implemented to do each of these tasks but for the project to be considered success full this meant that there must be multiple independent channels with there own adjustable binary sequences. This also meant that there had to be a user interface capable of allowing the end user to adjust the binary sequence for each of the channels as well as adjust the logic level that all channels were using. This was tested by making sure that the input received from each of the channels matched what was being indicated by the end user and that when the channel was adjusted the binary sequence of only that specific channel changed.

## Key Accomplishments:

For this project, the early stages of designing and planning went very well with plenty of redundancy in place. The project was designed with two possible display methods the OLED display and a set of individual LEDs, and two sets of input methods serial and a rotary encoder. This allowed for flexibility and adaptability in the firmware, if one of these failed in hardware or presented to much of a challenge to program there was an alternative route so it would help guarantee the project would be functional at the end. The firmware it's self-worked well with little to no issues this was mostly do to planning in advanced what the general structure of the code looked like in advance. This structure essentially had 3 modes a mode to change the voltage, a mode to change the selected channel, and one to change the pattern output. This worked very well with the rotary encoder which had a built-in button which allow modes to be changed with a button press and the values in each mode to be shifted by turning the nob. One of the most key success was the use of CMOS inverters to toggle all the channels at a specified logic level. This method works flawlessly in prototyping and was a key part in making sure the there was correct output.

## Key Problem Areas:

One of the first major obstacles was with getting parts for the first prototype because this project in its early stages required the use of oscilloscopes which I only had access to in lab which made my window for prototyping even smaller. There were issues with getting parts form suppliers hence issues getting those parts for prototyping this set me back a lab session and a half for prototyping with only 4 labs in total allotted for prototyping. Because the circuit could not finalize without prototyping, I had to use 2 extra lab sessions to get the prototyping finalized to begin work on finalizing designs. The second issue was caused mainly by rushing to make up for lost time which led to a lack of understanding of what was required and what formats to use when submitting assignments. No matter how many times I looked at the write up I could not understand what exactly was expected of me because I was not taking my time or talking to other students about the criteria. This caused major setbacks due to designs not getting approved and ended up causing me to miss the deadline to get my board fabricated by the school. By the time I was able to get approved it was too late I was able to get a board sent out and get credit, but I was not able to get my PCBs in time to populate and test them correctly. To at least allow for my design to be demoed I had



## PROJECT POST-MORTEM

to build a fully functional prototype and buy a digital discovery to demo it but the prototype used a 3V instead of a 3.3V which caused issues reading the output in 3V mode making the results for that portion of the demo inconclusive.

### Post Project Tasks/Future Considerations

A lot of the issues with trying to complete this project involved poor time management and loss of time. To prevent this in the future I would suggest making a schedule ahead of time to avoid spending too much time on small details like was the case in the early development stages with the design and prototyping. Making a schedule also would have allowed for the planning of setbacks. Another major adjustment would be to reduce the amount of time spent reworking elements of the project that already worked. The next adjustment is to establish better communications so when something is wrong it can be more quickly addressed. The biggest and most necessary adjustment would be to take my time reading the requirements more carefully and be less shy about asking questions.

## PROJECT POST-MORTEM

### Lessons Learned:

Category	Lesson Learned	Achieved?	Comments
Project Planning	Product concept was appropriate to Business Objectives	Y	
	Project Plan and Schedule were well-documented, with appropriate structure and detail	N	Not much documentation on deadlines
	Project Schedule encompassed all aspects of the project	Y	
	Tasks were defined adequately	Y	
	Stakeholders (e.g., Sponsor, Customer) had appropriate input into the project planning process	N/A	
	Requirements were gathered to sufficient detail	Y	
	Requirements were documented clearly	Y	
	Specifications were clear and well-documented	Y	
	Test Plan was adequate, understandable, and well-documented	Y	
	External dependencies were identified, agreements signed	Y	
	Project budget was well defined	N	No budget was defined
	End of Phase Criteria were clear for all project phases	Y	
	Project Plan had buy-in from the stakeholders	N/A	
	Stakeholders had easy access to Project Plan and Schedule	N/A	
Project Execution	Project stuck to its original goals	Y	
	Changes in direction that did occur were of manageable frequency and magnitude	Y	
	Project baselines (Scope, Time, Cost, Quality) were well-managed (e.g., changed through a formal Change Control Process)	Y	
	Design changes were well-controlled	N	Some changes were made despite working wasting time
	Basic project management processes (e.g., Risk Management, Issue Management) were adequate	N	The risk of getting parts and boards from suppliers was not fully considered causing mismanagement of time
	Project tracked progress against baselines and reported accurate status	Y	
	Procurement (e.g., RFP, Contract with vendor) went smoothly	N	Vender delivered stencil but did not deliver PCB or certain parts
	Contracted vendor provided acceptable deliverables of appropriate quality, on time, and within budget	N	Products were not delivered on time some products did not arrive at all
	Stakeholders were satisfied with the information they received	N/A	

## PROJECT POST-MORTEM

Category	Lesson Learned	Achieved?	Comments
	The project had adequate Quality Control	Y	
	Requirements – specifications – Test Plan were well-managed (e.g., Requirements Management System was used)	Y	
<b>Human Factors</b>	Project Manager reported to the appropriate part of the organization	Y	
	Project Manager was effective	N	I could have made better use of time and did not complete my goal
	Project Team was properly organized and staffed	N/A	
	Project Manager and staff received adequate training	Y	
	Project Team's talent and experience were adequate	Y	
	Project team worked effectively on project goals	Y	
	Project team worked effectively with outside entities	N	I struggle with communicating with Tim effectively and it cause setbacks
	There was good communication within the Project Team	N/A	
	Management gave this project adequate attention and time	N	This project required lab equipment to complete effectively so it need a little bit more time for prototyping
	Resources were not over-committed	N	
	Resources were consistently committed to project aims	N	Not enough resources
	Functional areas cooperated well	Y	
	Conflicting departmental goals did not cause problems	Y	
	Authority and accountability were well defined and public	Y	
<b>Overall</b>	Initial cost and schedule estimates were accurate	N/A	
	Product was delivered within amended schedule	N	
	Product was delivered within amended budget	N	
	Overall Change Control was effective	N	
	External dependencies were understood and well-managed	N	
	Technology chosen was appropriate	Y	
	The project was a technological success	N	
	Customer's needs/requirements were met	N/A	
	Customer was satisfied with the product	N/A	
	Project Objectives were met	N	Not completed even though functional prototype works as expected
	Business Objectives were met	N/A	

## PROJECT POST-MORTEM

### Project Close Acceptance:

---

Project Manager Signature

---

Marcel Vieira

---

Project Manager Name

---

12/08/20

---

Date

---

Sponsor Signature

---

Sponsor Name

---

Date