Mechatronics Fall 2023 Intro Project

Brendon Swierczewski

Introduction

The product I developed for the mechatronics intro project is a versatile and interactive LED light that can be controlled in terms of brightness and ON/OFF state. The LED is brightness controlled through serial input to an Arduino Uno R3 and toggled ON/OFF using a push button.

Functionality

The way this system works is quite simple. The push button is connected to the Uno's built in +5V supply and GND, while the output is connected to a digital pin (in my setup PD4). A 10kΩ resistor is connected between the push button and GND. Without the resistor, regardless of button state change, the digital pin would always read LOW as the +5V would be shorted to ground. With the resistor in place, when the button is pressed the digital pin reads HIGH. The anode of the LED is connected to a PWM digital pin (in my setup PD5) and the cathode is connected to ground. A 330Ω resistor is placed between PD5 and the anode of the LED. This is done to ensure that a safe current passes through the LED, preventing it from burning out. I would like to point out that the placement of the 330Ω can also be placed between the cathode of the LED and GND, as the current through the LED will be the same regardless of resistor placement (KVL).

The initial brightness and LED state are set to 0 in my code. PD4 is configured as an Input pin and PD5 as an Output pin. My program reads the current state of the button and saves the value to a variable called lastButtonState. The program then reads the button state once again and saves the value to currentButtonState. If there is a difference between the current and previous button state, then a button press has occurred. If the LED was off the program sets the LED state to the current brightness value selected. If the LED was on then the LED state is set to 0. Setting the LED state I use an analogWrite command because I am writing to the PWM pin, and reading the button state I use a digitalRead command because it is either HIGH or LOW. Once the button state checks are done, the program reads a character from the serial monitor. Depending on the input the program will adjust the LED's brightness or display the current brightness.

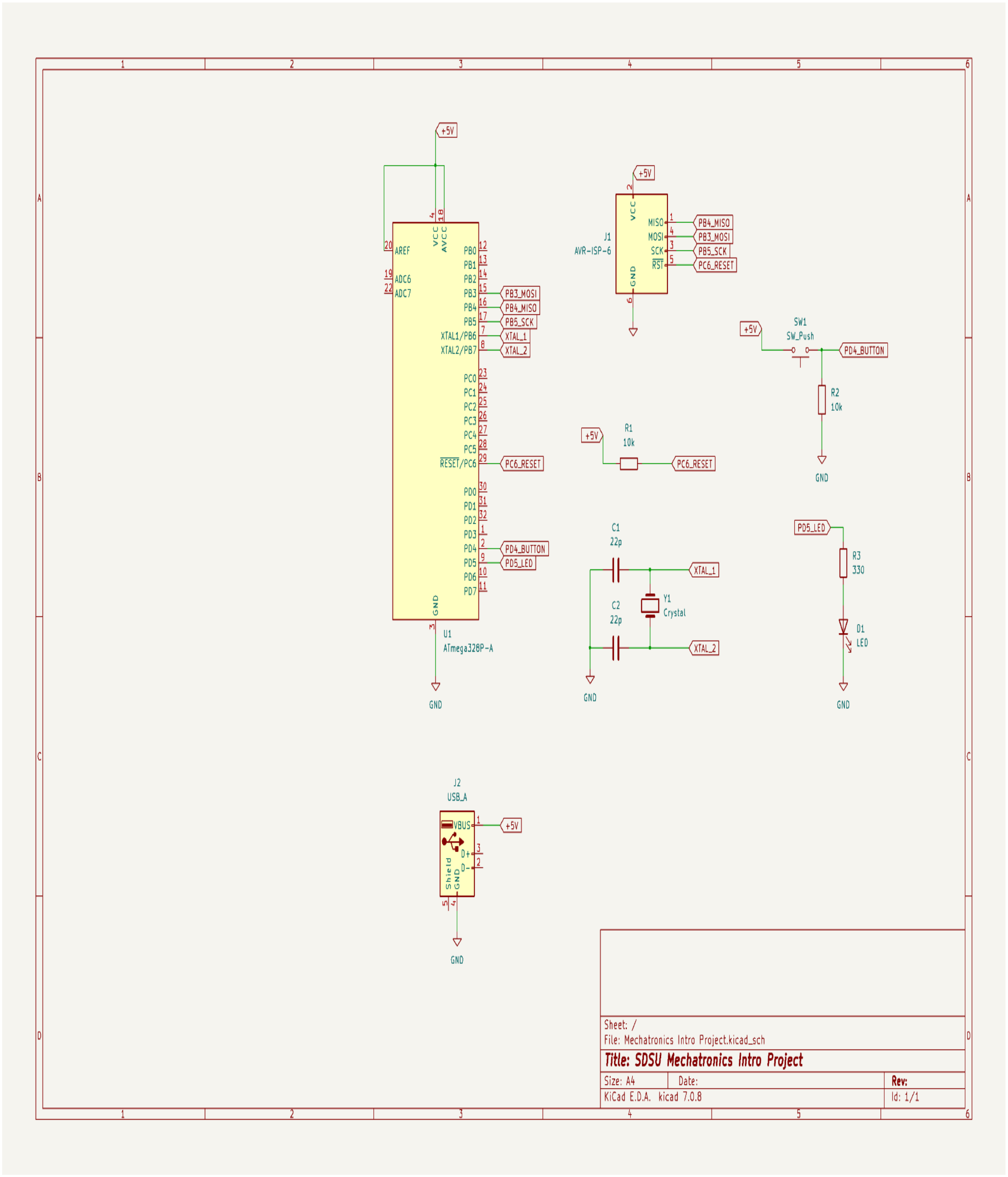
Challenges

Throughout the design process I came across a few challenges. When initially designing the user input, I was having the problem that the button would turn on when a new brightness input was set with the LED off. I fixed this problem by having the conditional statements for each input only change a variable called brightness and not write the brightness. The initial condition check of the button push either turns the button off or sets the button to the brightness variable's value. Another problem I had right at the beginning was my LED was only turning on past 50% brightness. The problem was that I used digitalWrite commands to the LED and not analogWrite.

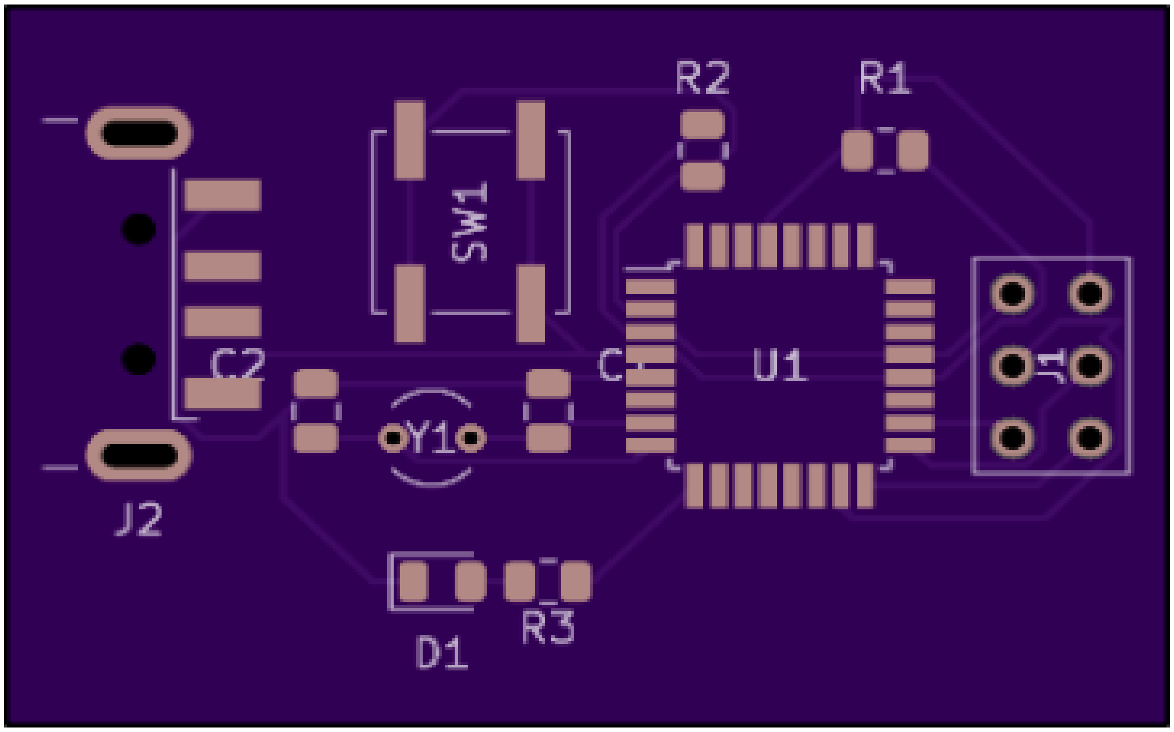
Conclusion

Overall I found this project to be very enjoyable. I was not aware of Tinkercad's ability to simulate circuits with code for the selected microcontroller and I will definitely be taking advantage of that in the future. Furthermore, I have not had any experience using a PCB design software like Kicad and I really enjoyed being able to learn how the program works. I feel more confident in my abilities to transfer a circuit to PCB.

Schematic

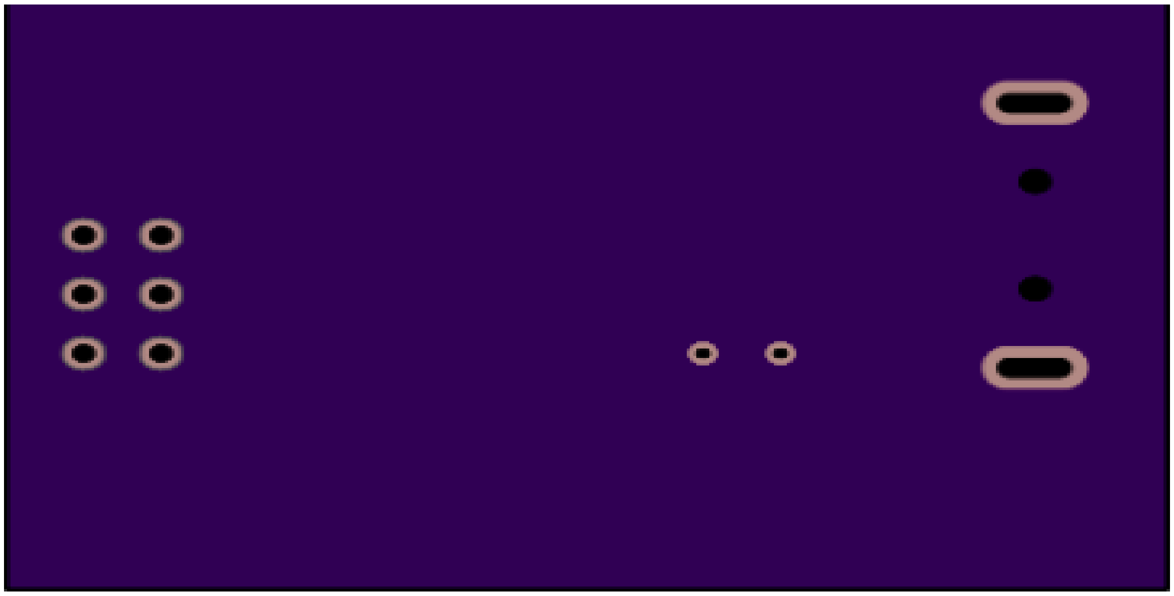

+5V
U1
ATmega328P-A
VCC
AVCC
AREF
ADC6
ADC7
PB0
PB1
PB2
PB3
PB4
PB5
XTAL1/PB6
XTAL2/PB7
PC0
PC1
PC2
PC3
PC4
PC5
RESET/PC6
PD0
PD1
PD2
PD3
PD4
PD5
PD6
PD7
GND
PB3_MOSI
PB4_MISO
PB5_SCK
XTAL_1
XTAL_2
PC6_RESET
PD4_BUTTON
PD5_LED
J1
AVR-ISP-6
MISO
MOSI
SCK
RST
SW1
SW_Push
R1
10k
R2
10k
R3
330
C1
22p
C2
22p
Y1
Crystal
D1
LED
J2
USB_A
VBUS
D+
D-
Shield
GND
Sheet: /
File: Mechatronics Intro Project.kicad_sch
Title: SDSU Mechatronics Intro Project
Size: A4
Date:
Rev:
KiCad E.D.A. kicad 7.0.8
Id: 1/1

PCB Front



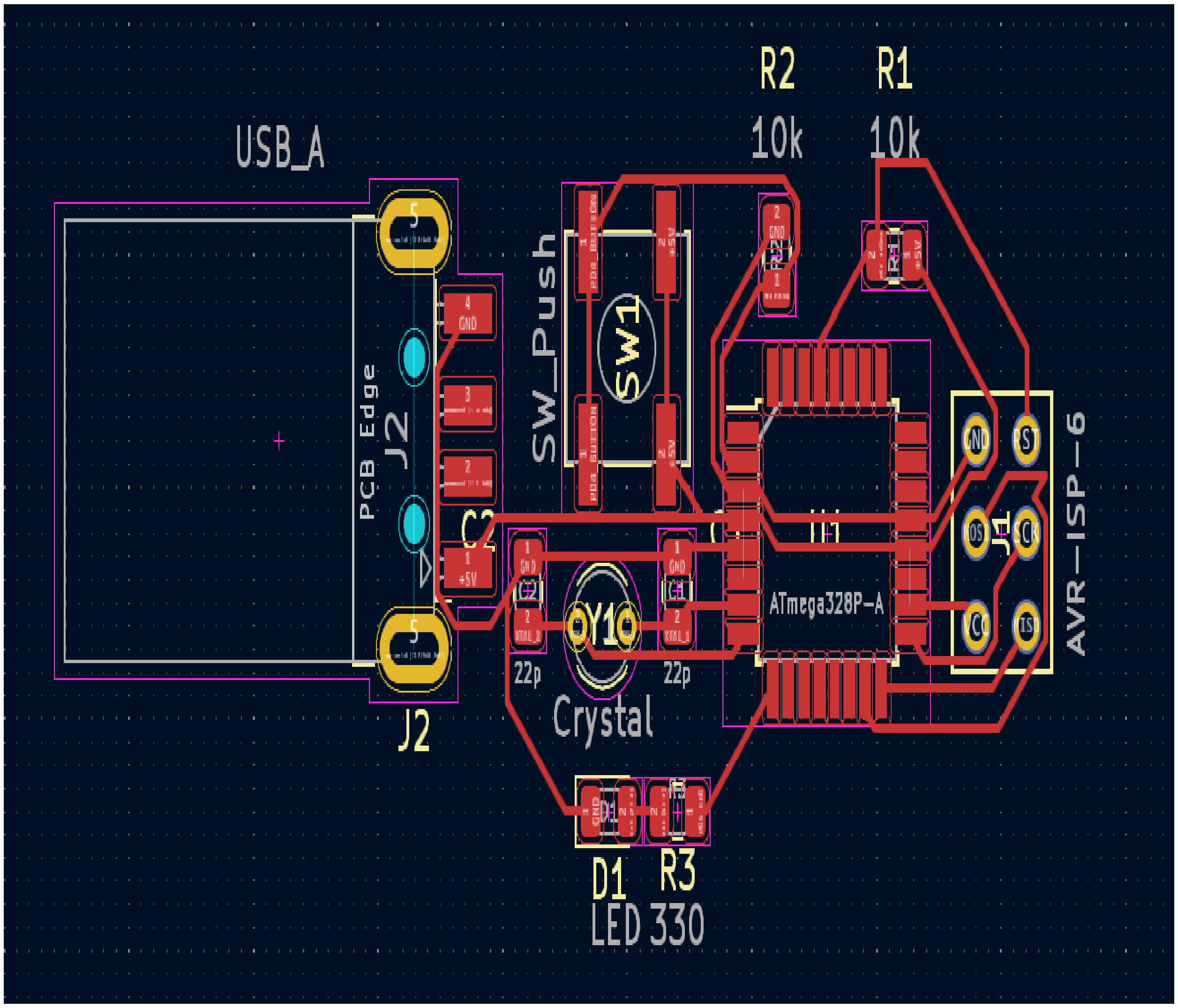
Top

PCB Back



Bottom

KiCad PCB


R2
10k
R1
10k
USB_A
SW_Push
SW1
PCB Edge
J2
C2
C1
Y1
22p
22p
Crystal
U1
ATmega328P-A
J1
AVR-ISP-6
GND
RST
MOSI
SCK
VCC
MISO
J2
D1
R3
LED 330