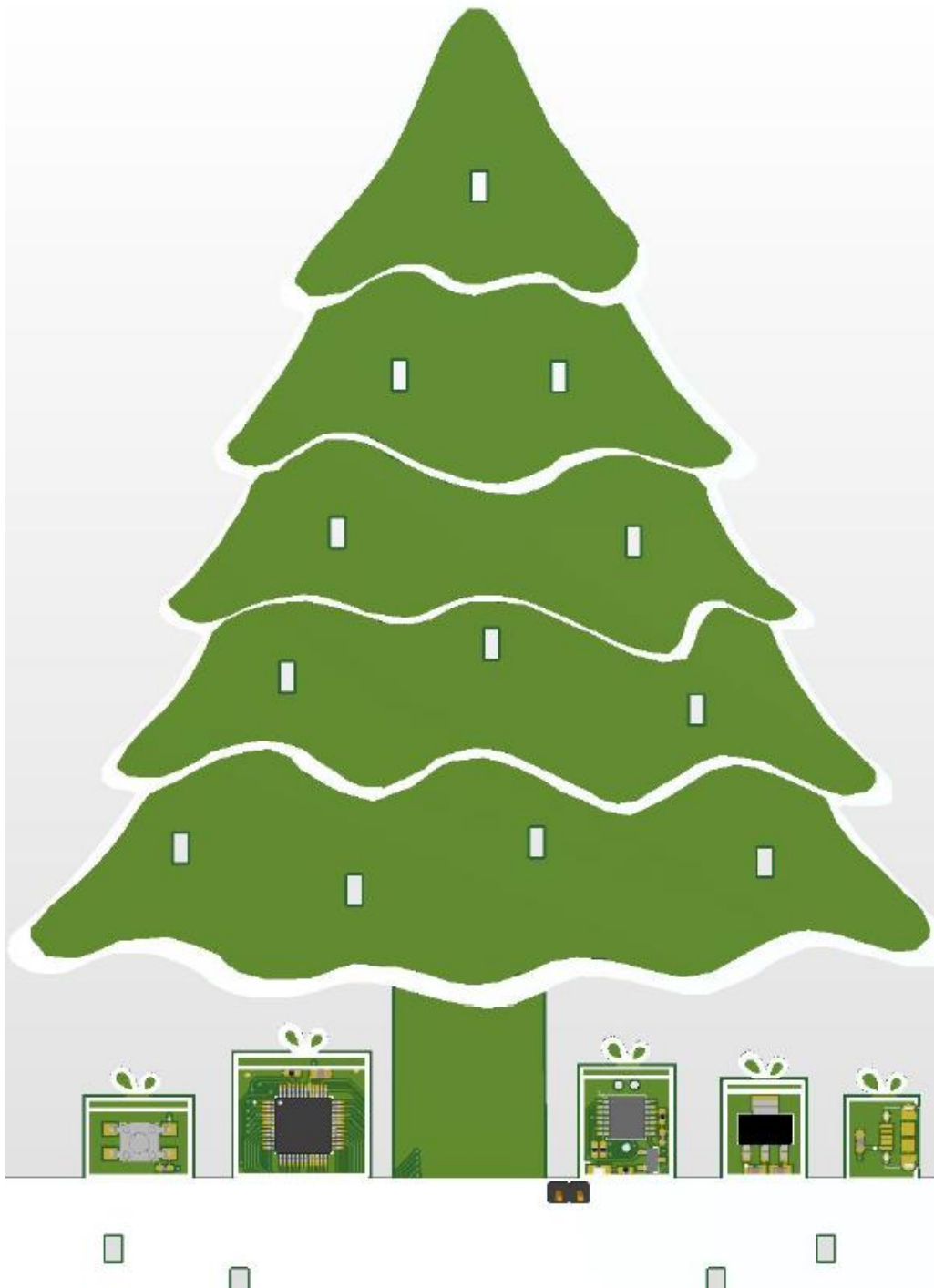


Weihnachtsprojekt



Inhaltsverzeichnis

Idee.....	3
Umsetzung.....	4
Schema	4
Layout	6
Programm.....	9
Fazit	10

Idee

Ich wollte für Weihnachten 2020 ein spezielles Geschenk an Familie und Bekannte machen. Eins das von mir kommt. Gleichzeitig hat JLCPCB einen Wettbewerb gestartet, bei dem man ein PCB designen muss das einen Festlichen Stil hat. Dabei kam ich auf die Idee einen Weihnachtsbaum zu zeichnen unter welchem sich Geschenke befinden und darin sind die grossen Bauteile. Als nächstes entschied ich mich dafür dass LEDs die Weihnachtsbaumkugeln simulieren sollten.

Umsetzung

Schema

Ich habe mir eine Skizze vom Baum gemacht und überlegt wie viele und wie ich die LEDs verteilen will. Optimal sind es bei dieser Grösse 12. Um auch schöne Animationen zu gestalten brauchte ich RGB LEDs und für den optischen Pluspunkt Reverse LEDs, die durch kleine Löcher im PCB, durchscheinen können.

Um nicht Tausend RGB Treiber haben zu müssen, nahm ich gleich einen mit 36 Kanälen. Dieser kann man über I2C ansteuern. Er hat für jeden Kanal ein PWM Register und noch weitere Einstellungen.

Das Interessante an diesem Bauteil ist, dass der Pin für die Slave-Adresse 2 Bits entsprechen.

Table 1 Slave Address (Write only):

Bit	A7:A3	A2:A1	A0
Value	01111	AD	0

AD connected to GND, AD = 00;

AD connected to VCC, AD = 11;

AD connected to SCL, AD = 01;

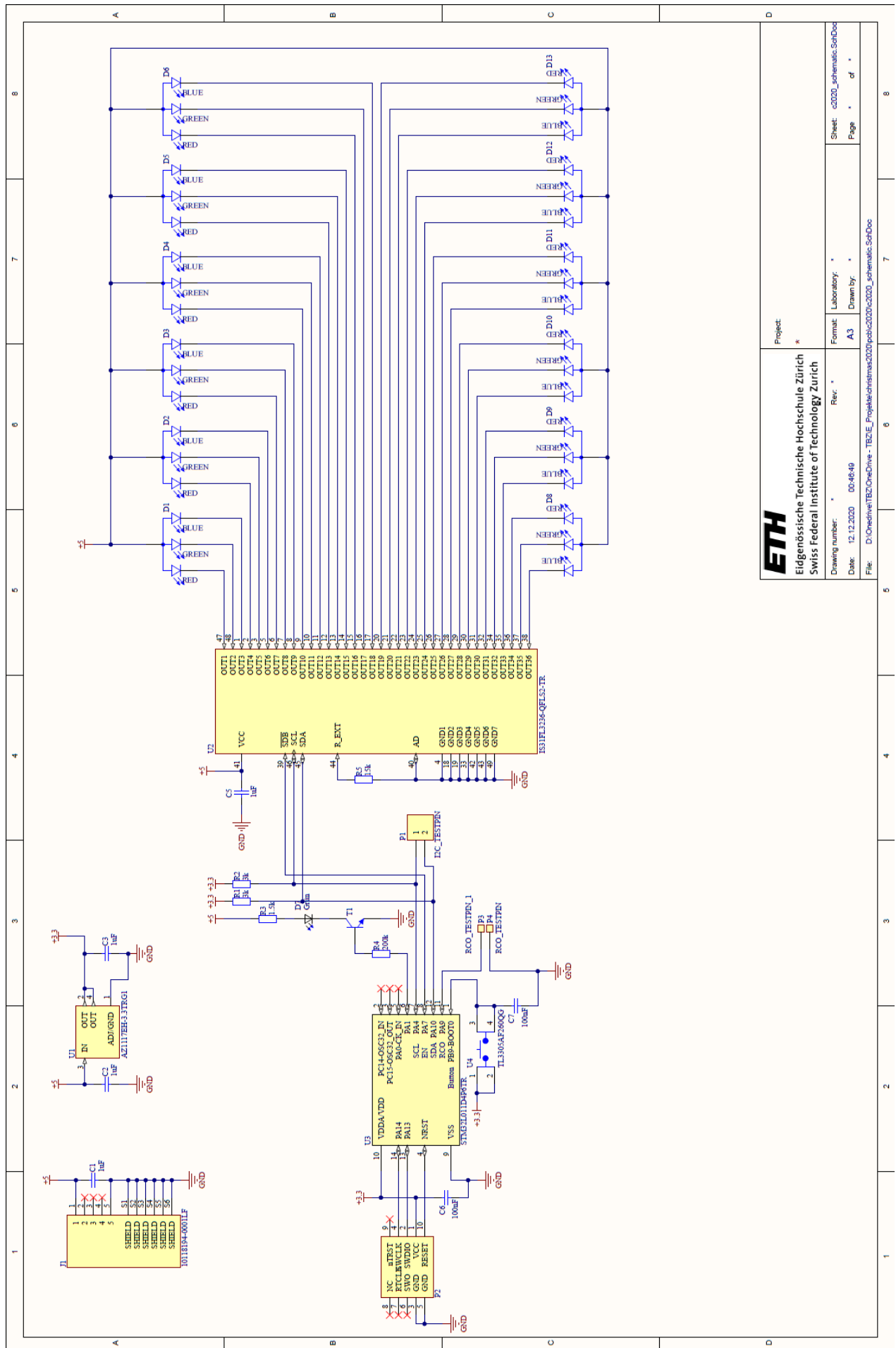
AD connected to SDA, AD = 10;

Das PCB wird über einen USB Mikro Anschluss versorgt. Die heutigen Geräte haben meistens einen USB C Anschluss. Zunächst wollte ich auch eine USB C Buchse nehmen, aber da meine Verwandtschaft eher noch ältere Geräte haben, haben sie auch noch ein paar alte USB Mikro Kabel herumliegen.

Über einen Linearregler kriegt man eine 3.3V Speisung.

Einen Taster ermöglicht das Wechseln zwischen verschiedenen Animationen. Jedoch habe ich nicht gesehen dass der gleiche Pin vom Mikrokontroller, an dem er angeschlossen ist, auch BOOT0 ist. Dieser Fehler hat mich einige Nerven gekostet und konnte es später aber herausfinden. Parallel zum Stütz-Kondensator gibt es jetzt einen 10kOhm Widerstand.

Da ich fast nichts steuern muss, brauchte ich nur einen ganz kleinen Mikrokontroller, welcher ein TSSOP-14 Gehäuse hat. Programmieren tue ich ihn über einen TagConnect Footprint welcher ich zum ersten Mal benutze. Zusätzlich gibt es noch ein LED welches mir noch über das Programm, einen Status geben kann.



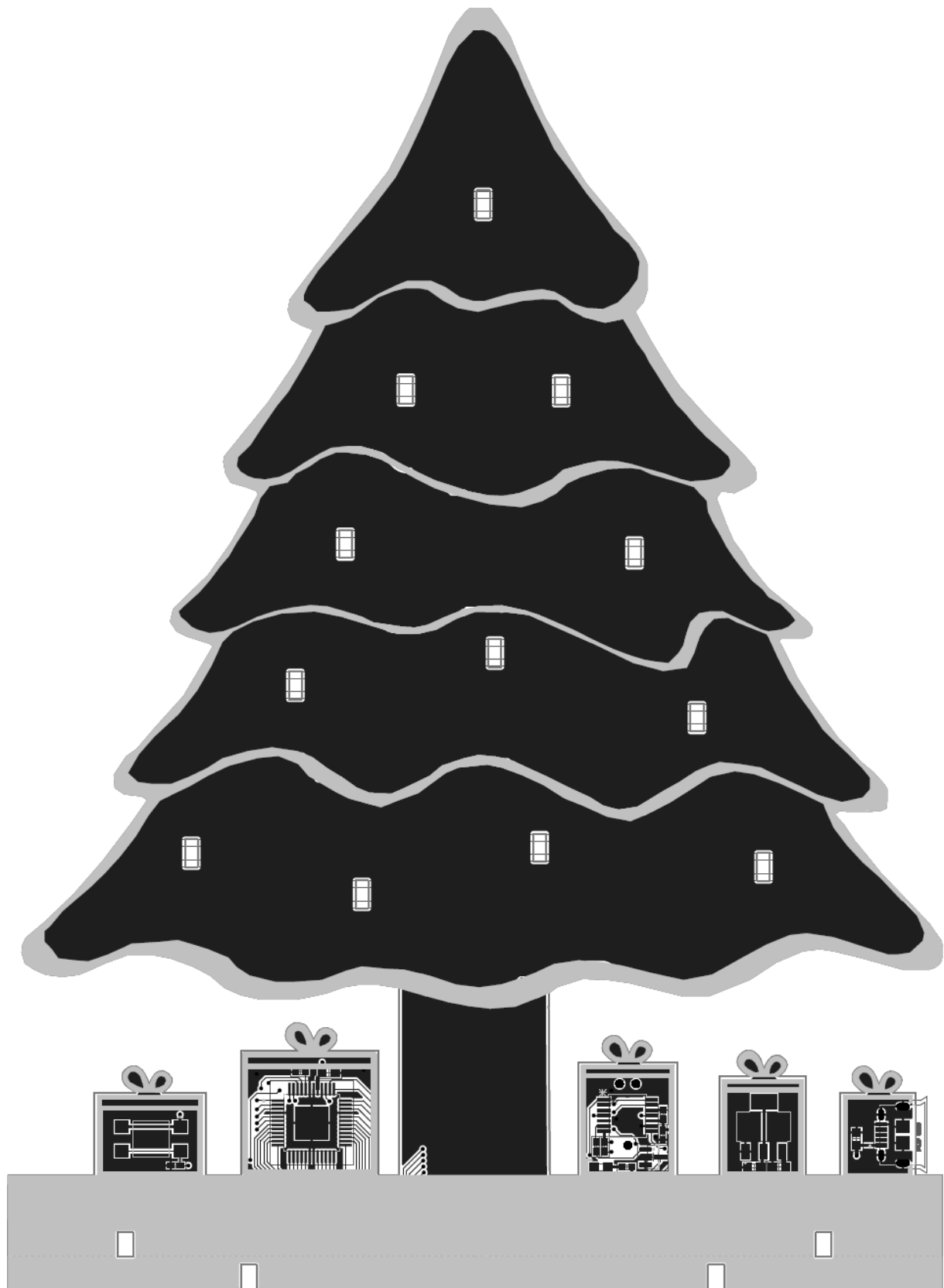
Layout

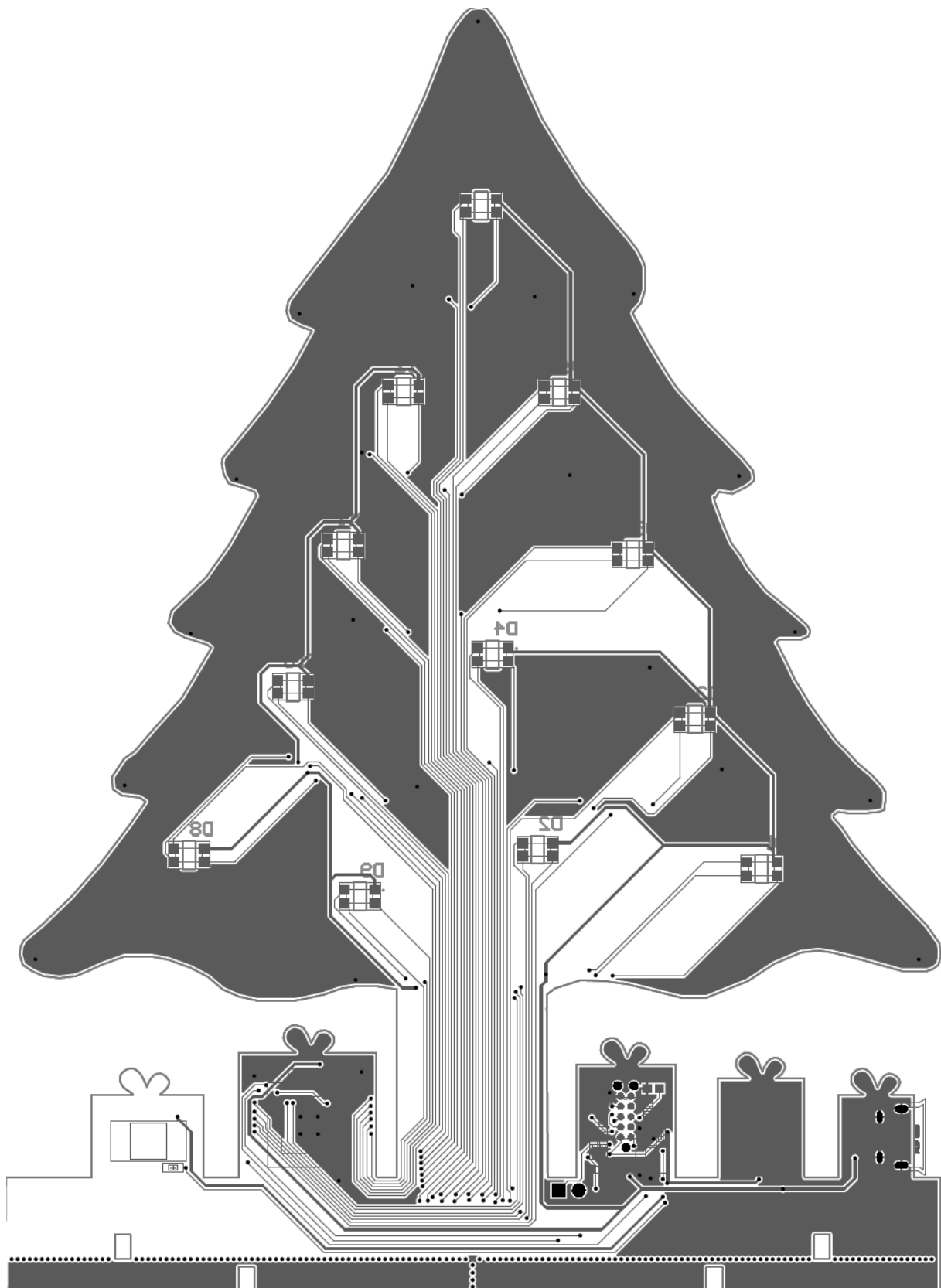
Der grosse Aufwand hier war der Weihnachtsbaum selber. Ich habe mir irgendein Vektorbild von einem Weihnachtsbaum auf Google gesucht und das in Altium importiert. Danach habe ich den Boardrand auf den Baum angepasst. Die Schneespuren, die auf dem Bild waren, habe ich mit dem Polygon nachgefahren und dann auf das Silkscreenlayer verschoben.

Auf diesem Print ist der Baumstamm nicht das unterste Teil. Sondern es gibt noch einen Boden auf dem die Geschenke sich befinden und Leiterbahnen sich verstecken. Der ganze Boden ist mit dem Silkscreen bedeckt.

Damit der Baum auch steht platzierte ich Löcher in einer Linie auf die ganze Breite, damit ich nachher dieses Stück abbrechen kann und überkreuz in den Boden des Baums stecken kann.

Auf der Rückseite des Baumes sind die LEDs versteckt und die grossen Bauteile die grossen Bauteile. Damit man keine Löcher von Vias im Baum hat, habe ich alle unter dem Silkscreen versteckt. Die Leiterbahnen des LED Treibers musste ich sehr eng ziehen, da fast kein Platz vorhanden ist. Der TagConnect Anschluss befindet sich auf der Rückseite.





Programm

Ich hatte zuerst das Problem dass ich den LED Treiber nicht ansprechen konnte. Es lag daran das der Enable-Pin gar nicht Active Low ist, so wie es im Schema gekennzeichnet ist.

Für den Treiber habe ich einzelne Funktionen, bei denen die Slave-Adresse schon fest vorprogrammiert ist.

Register und die Slaveadresse wurden mit Defines bestimmt. Genauso die zugehörigen Register zu den RGB LEDs.

Das LED unterhalb des Mikrokontrollers zeigt den Kommunikationsstatus der I2C Leitung. Blinkt sie einmal, kann er den LED Treiber ansprechen. Bei zweifachen Blinken kann er es nicht. Danach gehen die LEDs an.

Es gibt fünf verschiedene Animationen:

1: RGB

Jeweils vier LEDs haben eine definierte Farbe (RGB). Jede LED hat eine eigene Zählvariable die einen Sinus-Lookuptable folgt. Damit hat man ein schönen blinken

2: Gold

Alle LEDs leuchten in einem Grün-Rot Gemisch, welches ähnlich aussieht wie Gold;

3: Rot

Alle LEDs leuchten rot.

4: Grün

Alle LEDs leuchten grün.

5: Blau

Alle LEDs leuchten blau.

Der Taster habe ich an einen EXTI Eingang gelegt um ihn nachher per Interrupt abzufragen.

Dies erlaubt es mir ununterbrochen die Animationen auszuführen ohne ständig nach dem Taster zu fragen.

Fazit

Ich konnte für eine lange Zeit den Mikrokontroller nicht programmieren. Einerseits da Boot0 nicht mit Ground verbunden wurde und der STlink/v2 nicht auf dem neusten Stand war.

Zuhause habe ich nicht Möglichkeit kleinere Sachen einzulöten. Daher hatte ich sehr Mühe mit dem LED Treiber einlöten. Es kann sein das einzelne Farben von LEDs nicht funktionieren.

Das Projekt hat mir ziemlich Spass gemacht, da es Designtechnisch eine Herausforderung war. Ich war nicht sicher ob die Kanten und Ecken der Schlaufen oder der Baum an sich gut herauskommen.

Bei diesem Projekt konnte ich viel lernen, darunter konnte ich auch lernen wie ich einen EXTI-Interrupt und I2C der HAL Library verwende. Nun weiss ich auch das beim

Alle Dateien findet man in diesem GitHub:

<https://github.com/MXACE/C2020>