

Vortex NTNU - Norwegian University of Science and Technology

Benjaminas Visockis, Tina Harildstad, Martynas Smilingis,
Henrik P. Hemnes, Sander A. Hjortland, Benjamin R. Andersen,
Henrik Hammarstrøm, Aleksander Brackenbury, Ronja Kræmer,
Christopher Strøm, Gjermund Bjørnstad, Oliver Rothenbach,
Theo A. Martin, Lars M. Ølstad

August 2023

1 Introduction

Vortex NTNU is a non-profit student organization from the Norwegian University of Science and Technology in Trondheim, Norway. The focus of the organization is to introduce students to the field of marine robotics, a field which is of great importance to Trondheim, and Norway as a whole. Trondheim is home to many marine robotics companies and the fjord is important for Norway's technological development as the first "technological playground" for all kinds of unmanned and autonomous vessels.

Originally Vortex NTNU started in 2016 with an Remotely Operated Vehicle (ROV) competing in the MATE ROV Competition. Three ROVs were developed with that in mind. The third ROV, Manta, was then converted to an Autonomous Underwater Vehicle (AUV) to compete in the RoboSub competition. The next drone, Beluga, was designed with the goal of being an AUV from the start.

Vortex NTNU started work on its first ever Autonomous Surface Vehicle (ASV) in fall of 2021, with both RoboBoat and Njord Challenge in mind. Two years later, around 30 000 hours of design, development, and testing, has resulted in the latest iteration of Freya ASV.

2 Vessel design

This section will cover all hardware design going into the finished vehicle known as Freya ASV, its 3D model depicted in figure 1.

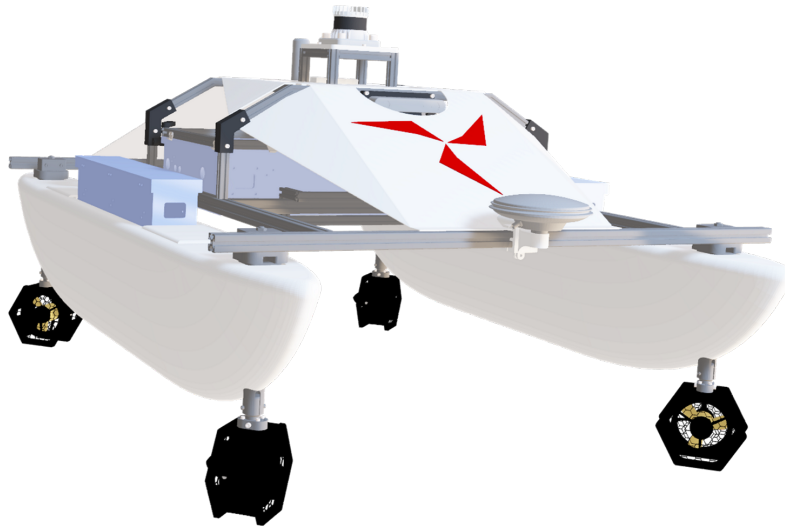


Figure 1: Freya ASV vessel design.

2.1 Hull/Pontoon Design

The initial pontoons, made of Divinycell material, were suited for the RoboBoat competition but lacked the buoyancy needed for the Njord Challenge. In the second iteration, the pontoons were halved and a Divinycell block was inserted for the required buoyancy (see figure 2). The added block, formed from four glued Divinycell plates, was shaped by hand to match the original form. A fiberglass and epoxy coating added strength and protect the Divinycell. A final coat of paint was added for smoothness and safety.

For battery positioning, cutouts in the pontoons lowered the vessel's center of gravity. Drainage holes prevented water buildup, though there was a risk of intake if the pontoons sank too low. Hydrostatic calculations and tests confirmed the waterline would not be high enough for this to happen. Aluminum tubes at each pontoon end serve as mounts for motors and the superstructure.

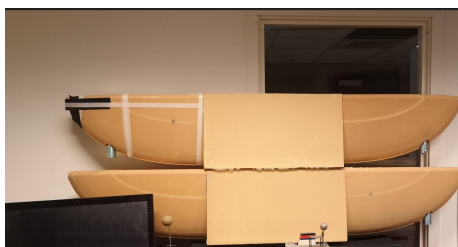


Figure 2: Elongated pontoons.

2.1.1 Materials Used

Divinycell, a durable lightweight PVC foam resistant to water, was best worked with using a jigsaw, handsaw, dremel, and sander, discovered through trial and error. Epoxy proved effective for joining sections. Safety measures, including eye protection and particle face masks, were essential during cutting and sanding, along with suitable attire.

Due to Divinycell's insufficient standalone strength, coating became necessary. Various options were explored, including coatings, epoxy, and fiberglass. Paint and epoxy alone lacked strength, so fiberglass was chosen for its lightweight, strength, and cost-effectiveness. However, fiberglass alone doesn't ensure full waterproofing. Gelcoat is typically used for this, but due to health concerns, a marine-grade paint that bonds well with fiberglass was selected. Though not as resilient as gelcoat, this paint can be easily reapplied to address scratches.

2.2 Superstructure

The superstructure is the construction which holds together the pontoons and acts as the structural skeleton for the ASV. As this is Vortex's first attempt at an ASV, aluminium profiles were the construction material of choice. They offer a high level of modularity and the ability to easily change parts of the design, even after the ASV is produced. Aluminium profiles are also easy to acquire and are relatively cheap. The decision was made to go with 30x30mm aluminium profiles to ensure enough strength to carry a payload of at least 60kg and also withstand the forces created while driving and handling. The decision was based upon a preliminary calculation by hand and then by means of a FEM analysis in our CAD program. The superstructure is connected to the pontoons at four locations. The pontoons and the superstructure have been designed with easy assembly and disassembly in mind. For this reason a single center bolt was used in each of the four corners. The final design of the superstructure was constructed with future proofing and modularity as the main goals. This is why the space between the superstructure is divided into three equal compartments, which can house different payloads and equipment. One is required by the electrical housing and the other two are available for

future projects such as a sonar module or a winch and a moon pool for an ROV or AUV. The compartments could also be used to store extra batteries or other relevant systems.

2.3 Electrical Housing and Battery Compartments

2.3.1 Battery Compartments

For better battery integration, compartments were incorporated into the hull, which lightened the superstructure load and provided additional space for components. This method also shifted the center of gravity downward, enhancing vehicle stability.

The battery compartment measurements aligned with hull specifications: a cutout measuring 10 cm in width, 10 cm in depth, and 60 cm in length inside the pontoon. Battery dimensions posed a subsequent limitation, necessitating meticulous design by the electrical team.

Considering the criteria, three options emerged:

1. Purchase waterproof box: convenient, but unsuitable due to uncertain battery dimensions and unique size needs.
2. Blue Robotics cylinders: cylinders from our other drones could standardize the battery system across vehicles, but square batteries in round compartments wasted space and didn't meet the higher capacity requirements for the ASV.
3. Custom design: decision was made, to opt for custom compartments for the design, constructed from sheets of LEXAN. The material offered both robustness and fire resistance – crucial attributes for battery safety.

After sizing and selecting the material, a simple interlocking jigsaw design was chosen. Lexan was water jet cut due to harmful lasercutting byproducts. Edges were manually refined, then assembled with waterproof polyurethane glue and silicone-sealed for added security.

For battery access, a waterproof and easily removable lid was needed, along with a pressure-based seal. Custom O-rings were crafted using rubber door trim, matched to the compartment edges. Strategically positioned latches created pressure for a tight seal. 3D-printed stiffeners were later added to prevent the lid buckling under pressure, post extensive testing.

2.3.2 Electrical Housing Compartment

The electrical housing design was inspired by the battery module's layout. The electrical housing base consists of four side walls and a bottom plate with a jigsaw pattern at the sides and bottom. Similarly, the lid follows this approach,

with extended plates to cover the base. Connector cutouts were planned in the CAD in conjunction with the electrical team. See figure 3.

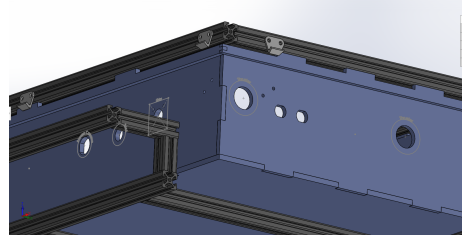


Figure 3: Electrical housing CAD model.

The plates were joined with robust Gorilla glue, offering mechanical strength and saltwater resistance, including UV protection for outdoor use. TEC-7 silicon was applied for extra waterproofing along edges and screws. Screws secured the lid via latches, featuring an EDPM rubber seal, ensuring waterproofing. See figure 4.



Figure 4: Electrical housing lid with glued EDPM seal.

Initially, LEXAN plates were favored for the electrical housing due to availability. The intention was to waterjet cut them, as polycarbonate isn't laser-cut friendly. However, waterjet cutting the LEXAN for battery modules led to excessively rough cuts, complicating waterproofing for the electrical housing. Consequently, the decision shifted to using acrylic plates as seen in figure 5.

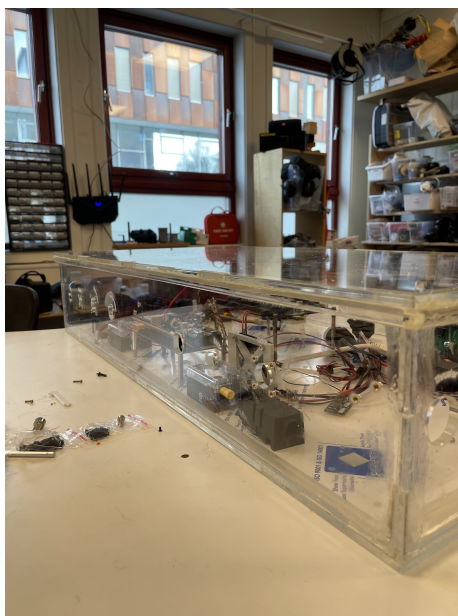


Figure 5: Electrical housing without frame.

The team opted for an aluminum profile frame for the electrical housing (see figure 6). This decision was influenced by the switch from polycarbonate to acrylic and the desire for a simpler attachment to the superstructure. The frame's "cradle" secures the housing in the xy-plane while allowing easy removal if necessary. Additionally, the frame fits well within the superstructure and includes handles for convenient removal.

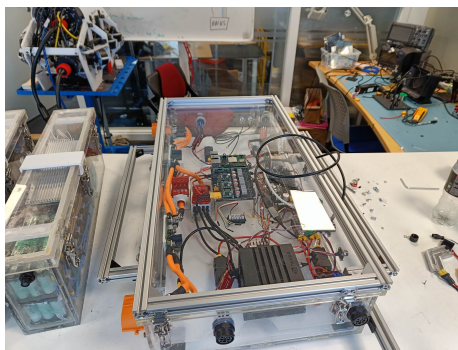


Figure 6: El house with frame and components.

2.3.3 El-house component layout

To strategize the arrangement of components within the El-house, our initial step involved creating basic 3D models for each individual element as can be seen in figure 7. This approach facilitated a clear visualization of the spatial requirements of the El-house. With a focus on both practicality and safety considerations, all components were mounted onto an acrylic plate, elevating them by 1 cm from the floor of the El-house. By adopting this approach, a safeguard was established against potential water leaks into the El-house, ensuring that the components remain slightly elevated from any water intrusion. All the components are mounted to the acrylic plate using threaded inserts and stand-offs.

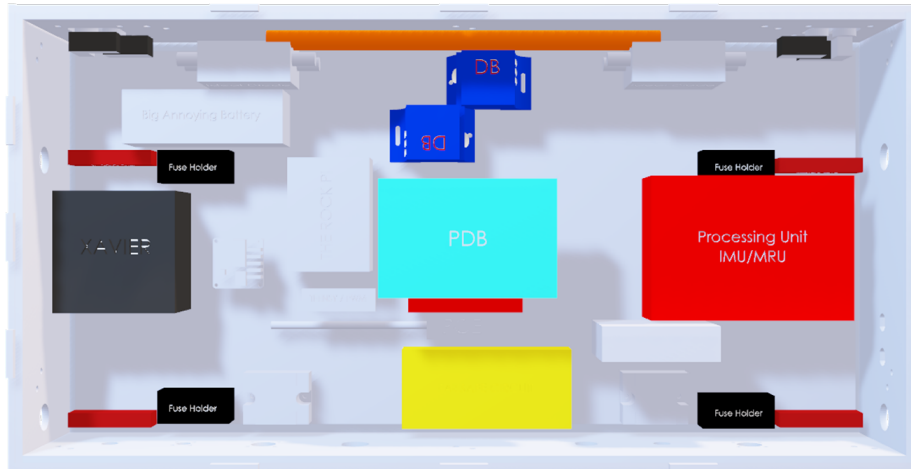


Figure 7: 3D-design of electrical housing layout.

The electrical housing is outfitted primarily with panel connectors sourced from the UTS series by Souriau. These connectors were meticulously chosen to align with the exact requirements of the external components' specifications. This comprehensive selection process encompasses vital factors such as current and voltage compatibility, pin quantity, dimensions, sturdiness, and IP-rating for protection against environmental elements.

2.4 Propulsion System

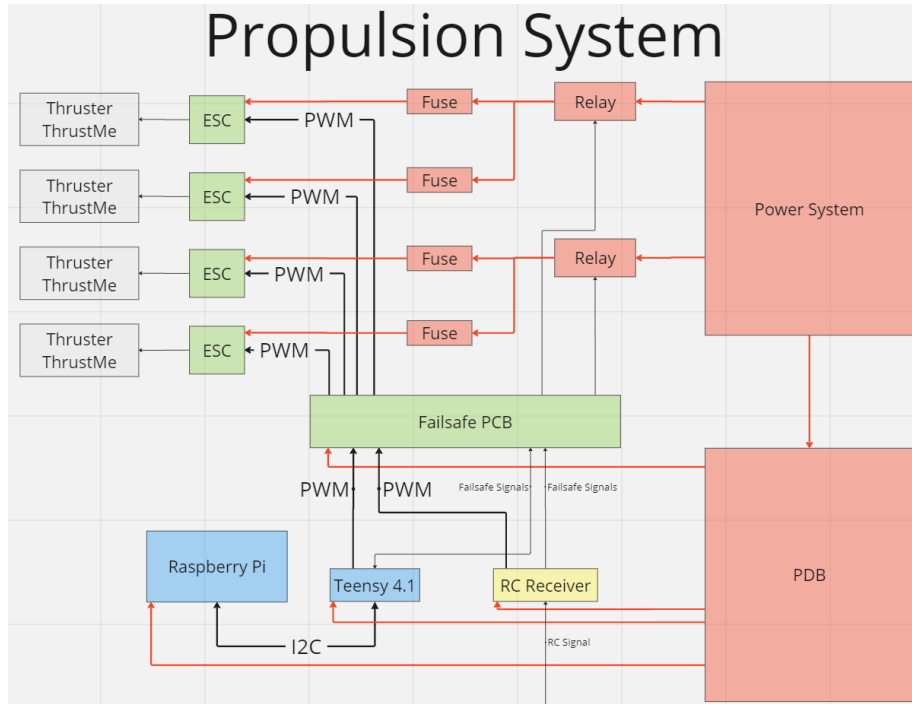


Figure 8: Overview of the propulsion system on board.

The thruster system onboard the ASV serves as a crucial component for propulsion and maneuverability. This section delves into the details of these systems, highlighting their specifications, functionalities, and integration within the broader systems.

ThrustMe Thrusters: An Overview

The ASV is equipped with ThrustMe type thrusters, each featuring a propeller diameter of 100 mm. Every thruster, when factoring in guards and connection shafts, weighs approximately 1 kg. These thrusters function within a voltage range of 12 to 25.2V, displaying optimal performance at 18.5V. At this voltage, the thrusters achieve a maximum forward thrust of 16 kg and a reverse thrust of 13 kg. Their operational range in terms of rotation spans from 4200 rpm under optimal conditions to 200 rpm in challenging environments. One noteworthy feature is the significant current draw; the design anticipates that each thruster can draw up to 60 amperes of current when submerged. Given that each motor has a power consumption of about 1.5 kW, the combined power consumption under extreme conditions is around 6 kW.

Configuration and Structural Integrity

The ASV houses four strategically positioned thrusters. Each thruster adopts a 45° angle; the front pair points outwards, while the rear pair angles inwards. This distinctive arrangement augments the ASV's capability in three Degrees Of Freedom (DOF): forward-backward movement and rotation. While such a configuration might slightly impede efficiency and subject the ASV to increased stress, the vessel's robust structural materials and innovative design counteract these potential pressures. As a result, there's a trade-off between a slight speed decrement and significantly improved control and agility, paving the way for the integration of advanced onboard control systems.

Power and Control: Electric Speed Control Units

Each thruster is powered by an individual Electric Speed Control (ESC) unit. Designed to mirror the thrusters' substantial power requirements and the associated high current draw, the ESCs are capable of managing up to 60A and function within a voltage range of 9-25.2 V. Additionally, to mitigate overheating during high-load operations, each ESC is equipped with an expansive heat sink that remains exposed to fresh air, ensuring efficient thermal regulation.

Safety First: The Failsafe system

Accidents are preventable. This has been a priority in the design and is explained in Section 2.7.2 Fail Safe System.

Control Logic and Integration

Commands for the thrusters are relayed through the ESCs, which interpret 50Hz RC PWM-signals ranging between 1100 to 1900 microseconds. Two predominant modes of control exist: Handheld Radio Control with up to 2 km range and software control executed through the main Micro Controller Unit (MCU).

For the software mode, the high-performance ARM processor-based Teensy 4.1 MCU is employed. This MCU, an advanced iteration with a robust ARM Cortex-M7 processor running at 600 MHz, communicates with a Raspberry Pi 4 via an I2C bus. The Raspberry Pi 4 processes setpoints and, using a low-level thruster interface driver, conveys thrust directives to the Teensy 4.1 via the I2C bus, allowing for bidirectional data exchange. The MCU then modulates these instructions into PWM signals for the ESCs, thereby dictating thruster operations. The Teensy 4.1 is meticulously programmed, encompassing optimized firmware for communication, error mitigation, and accurate PWM signal generation.

2.5 Batteries

The installed battery packs, use lithium-ion battery cells. In the fully-setup configuration, the boat has two battery packs in each battery module, each storing 1.3 kWh of power, resulting in a total capacity of 2.6 kWh. Nominal voltage of each battery pack is 22.2V. The two battery packs are connected in parallel

with additional safety circuitry to avoid rapid voltage balancing between two battery packs. This makes it possible to connect battery packs with different voltages without any issue.

The packs have been produced using 18650 cells from Panasonic welded together using nickle strips as interconnects. There are 120 cells in each battery, configured as 6 cells in series and 20 cells in parallel.

The strips can carry roughly 8 A at max load. This means the interconnects in themselves can carry 160 A in each battery. The C-rating of the batteries is high enough to support even higher currents, but the limiting factor in the battery setup is the connectors between the electrical housing and battery modules, which support 150 A. Each battery is therefore limited to 150A.

The batteries are managed by an off-the-shelf BMS from JIABAIDA. These can be programmed to set current limits, voltage protection, temperature protection etc. All of the parameters can be configured and the data is communicated to the main computer via RS-485 protocol from the BMS. The BMS also ensures that the voltage over the cells are balanced.

The battery module are quite large, measuring in at 540x92x190 mm and weighing 8.5 kg each. In return each pack is capable of delivering 150A and has a capacity of about 62000 mAh, making it possible to power components such as the thrusters at near peak performance. Considering the large size and adequate buoyancy of Freya, the gain in power is worthy a gain in overall weight and surface area.

2.6 Sensors

2.6.1 Exteroceptive Sensors

In order to perceive the world around Freya, a stereoscopic camera and a 360-degree Field of View (FOV) scanning LiDAR has been used. The stereoscopic camera grants a wide field of view and depth estimation, which is not available when using a mono lens camera. The camera also provides crucial color data which can be used in order to differentiate between the different sea markers. The LiDAR is a sensor that is able to provide a discretized scan of the environment. This allows for tracking of objects from all angles of attack which allows for collision avoidance.

2.6.2 GNSS

The onboard GNSS is the Seapath OEM, paired with the MiniMRU from Kongsberg. Two antennas equip the ASV, with one located at the front and the other at the rear. This enables the vessel to determine its heading as well as its position.

2.7 Hardware implementation

This section explains the electrical wiring onboard the ASV. Figure 9 shows an abstract circuit diagram of the contents of the elhousing and its connection to the rest of the ASV.

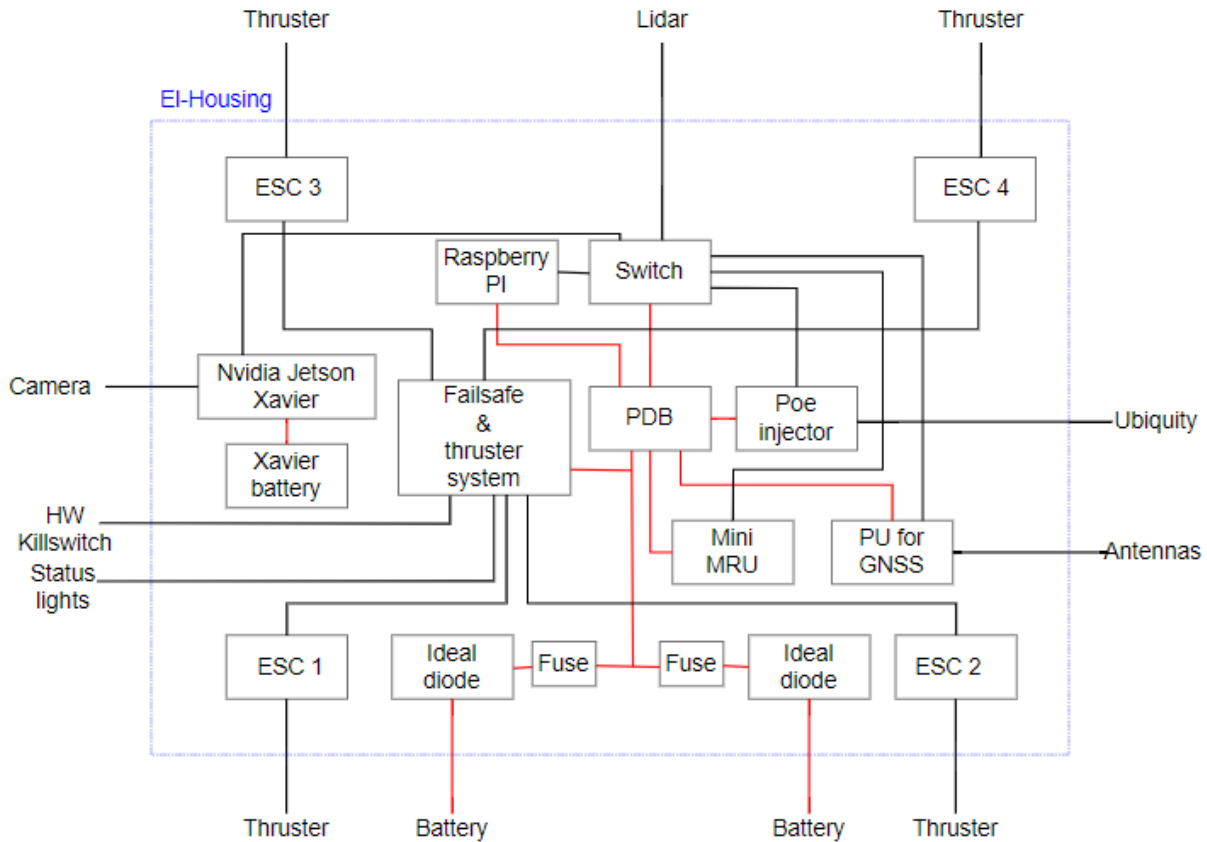


Figure 9: Circuit diagram depicting the el-housing.

2.7.1 Power Distribution Board

The PDB functions as a fuse board and voltage regulator. It powers 3V3, 5V, 12V and 24V systems using Pololu DCDC converters. The PDB has 16 separate fuse channels with LEDs that light up if a fuse burns out. The PDB also features a power sense module that measures input voltage and current draw.

In figure 10, a picture of the PDB can be seen.

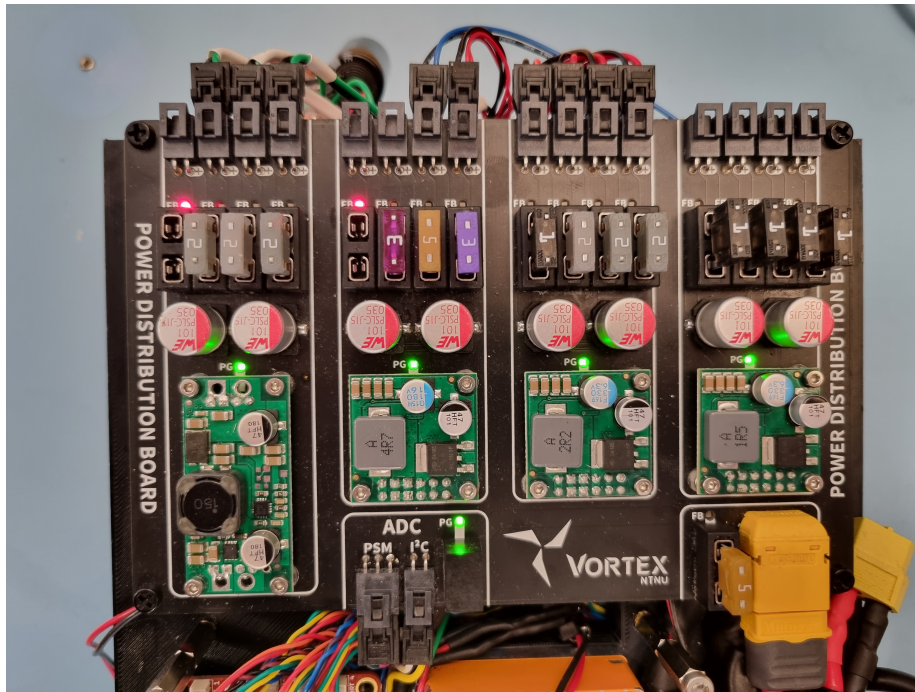


Figure 10: Power distribution board.

Schematics of the PDB can be found in Appendix A.

2.7.2 Fail safe system

The fail safe system ensures that the ASV may operate in a safe manner. This is implemented by controlling two normally open, solid state relays that manage power to the ESCs and thus thrusters. As seen in figure 11, the relays are controlled by a signal that is continuously sent from the Fail Safe PCB and routed through a two channel hardware kill switch consisting of an E-stop button.

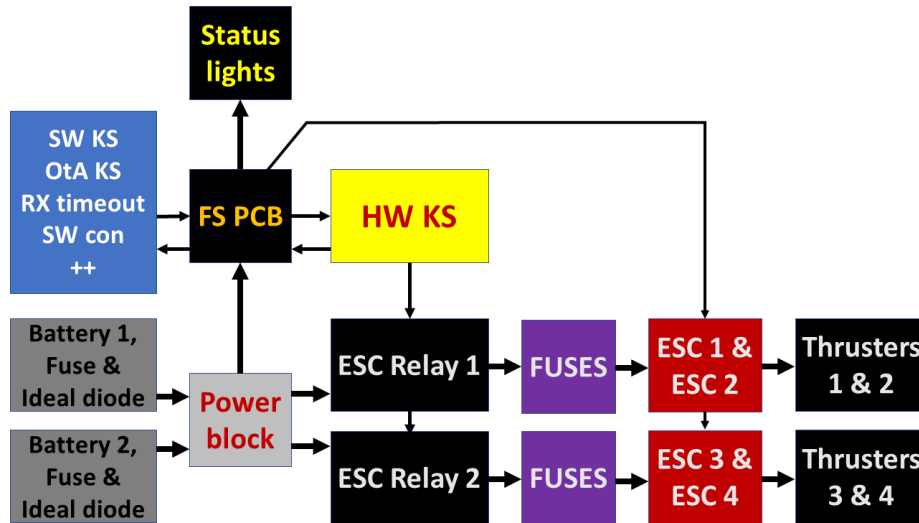


Figure 11: Abstracted schematic of the Fail safe system. Return paths are not shown.

The Fail Safe PCB uses an Arduino Nano to interpret signals from a FrSky ARCHER R10 PRO receiver. "Hardware operation mode", "Over the Air Kill switch", "Radio Timeout" and "Arming"-signals are sent to another Arduino Nano, the FS MCU, that handles the Fail safe logic. The logic consists of listening for kill switch triggers in order to disable thrusters and lock down the system when needed. The FS MCU takes trigger inputs from the Over the Air Kill Switch, Hardware Kill Switch, Radio Timeout and Software Kill Switch. These are indicated with LEDs to ease debugging. To unlock the system, the ASV has to be unarmed with no trigger active. After this the ASV may be rearmed so that the thrusters become operational again.

To limit the severity of one of the Arduino Nanos not working correctly, both Arduinos send a signal into an AND-gate that outputs the final signal controlling the relays.

The FS MCU also controls the Status light, indicating the mode of operation of the ASV. If any Kill Switch trigger is active, or the boat is locked, the status light is red. If the boat is unlocked, armed and manually operated, the status light is yellow. If the boat is unlocked, armed and autonomously operated, the status light is green. If the boat is unlocked, but unarmed, the status light flashes red-yellow or red-green, depending on the selected mode of operation.

The FS PCB switches the control signals for the ESCs using a multiplexer.

Schematics of the Fail Safe PCB can be found in Appendix B.

3 Software

3.1 Software Design

In regards to software, Freya utilizes Robot Operating System (ROS) as the middleware that interfaces between all the systems on the drone. ROS is language agnostic and in the case of the autonomous and perception software stack, packages are written in both python and C++.

3.1.1 LiDAR tracking with DBSCAN and PDAF

DBSCAN is short for density based clustering of applications with noise. The algorithm works by examining points within a specified radius ϵ , if there are more points within the radius than a predefined threshold, a core point is selected. From the core point DBSCAN expands a cluster iteratively until the cluster density is no longer sufficient. Points that do not fall within the predefined radius ϵ are categorized as noise.

Once a cluster of points have successfully been discovered, the position of the cluster is then fed to a tracking algorithm called Probabilistic Data Association Filter (PDAF). PDAF is an extension of the Kalman Filter, and is designed to perform data association between measurements and predicted tracks while taking into account the possibility of false detections, missed detections, and any noise that may degrade the measurements. Once an object is being tracked, its position, orientation, and velocity are fed into guidance and navigation systems of Freya.

3.1.2 Buoy detection with You Only Look Once

The following pipeline was developed to detect a buoy. Firstly, camera data is converted to ROS messages by the use of the ZED-ROS wrapper provided by Stereolabs. The raw image data is fed into a neural-network-based detection algorithm called You Only Look Once (YOLO). YOLO uses a neural network for each picture the algorithm takes in. The neural network used for the detection applications is a complex nonlinear function where you have an input layer which corresponds to all pixels in an image. Based on the pixel colour, position in frame, and intensity, the neural network is able to predict what kind of objects exist in the current image frame. Once an object is detected a bounding box is drawn around the object. The image containing the drawn bounding box is used together with depth data from the ZED2i stereo camera used to compute a position in the world. These positions serve as waypoints for Freya to follow.

3.1.3 ArUco detection with OpenCV

ArUco markers are detected by using the OpenCV library. Once a marker is detected in image frame, the algorithm is able to detect the position of the

object based on the camera calibration parameters. Estimated positions can then be fed to the autonomous systems on Freya.

3.2 Pathfinding

The main component for solving the pathfinding task is Line of Sight (LOS) guidance, which is a known algorithm used for following a given set of coordinates, or waypoints. The algorithm creates a straight line, the LOS vector, between two of the waypoints. The first waypoint is the starting point of the drone, and the second point is the target point to navigate towards. During the next iteration, the previous target point becomes the starting point and the next waypoint in the list becomes the target. The drone is then regulating its heading and speed by minimizing the error between the LOS vector and the drone's current heading, which results in the drone being able to navigate towards the LOS vector, and follow it towards the target.

In order to navigate according to the seamarkers, information about which seamarkers that are detected is stored. The logic for how Freya should behave is then based on which of the seamarkers that are closest to the drone.

3.3 State machine

In order for the drone to behave according to the situation at hand, the ROS framework SMACH is used for defining state machines for each of the separate competition tasks. SMACH easily lets the control system be broken into smaller states, where each state represents a specific action, or behaviour, of the drone. The transitions between states are based on the desired conditions or events, which gives the ability to trigger which behaviour that should be enabled given the current situation. In addition, sub-states are used to define more complex state machines where each task consists of several states. SMACH contributes to having a flexible and adaptive code base, where states easily can be modified without having to rewrite the entire control logic.

3.4 Motion control

Freya will operate in a dynamic marine environment, subject to uncertain disturbances such as waves, currents, and winds. Ensuring stability and robust performance in the face of these uncertainties can be achieved with use of a wide range of control techniques. One such approach is the Linear Quadratic Regulator (LQR), a well-established control algorithm designed to optimize the performance of a linear system against a quadratic cost function. The LQR algorithm aims to minimize the quadratic cost function

$$J = \int (\mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{u}^T \mathbf{R} \mathbf{u}) dt \quad (1)$$

where \mathbf{x} is the state vector and \mathbf{u} is the control input. \mathbf{Q} and \mathbf{R} are weighting matrices that dictate the trade-off between state error and control effort. For Freya, the state vector is defined as

$$\mathbf{x} = [x \ y \ \psi \ u \ v \ r]^T \quad (2)$$

where x and y are the position coordinates of Freya in the world frame, ψ is heading with respect to true north, u and v are the surge and sway velocities in the body frame, and r is the yaw rate. A linear model using these states will be on the form $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$, where the \mathbf{A} matrix describes the state evolution from one timestep to another, while \mathbf{B} defines how control inputs result in a change of the vessel state. However, since the vessel velocity is defined in the body frame, while the position is defined in the world frame, a non-linear transformation between body and world is required to describe the system model. As such, this necessitates a linearization of the vessel dynamics in order to produce the \mathbf{A} and \mathbf{B} matrices required to solve the LQR problem. Defining the matrices \mathbf{M} and \mathbf{D} as

$$\mathbf{M} = \begin{bmatrix} \text{mass} & 0 & 0 \\ 0 & \text{mass} & 0 \\ 0 & 0 & \text{inertia} \end{bmatrix} \quad \mathbf{D} = \begin{bmatrix} d_x & 0 & 0 \\ 0 & d_y & 0 \\ 0 & 0 & d_\psi \end{bmatrix} \quad (3)$$

respectively, the linearized model can be defined as:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & -v & \cos(\psi) & -\sin(\psi) & 0 \\ 0 & 0 & u & \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -d_{1,1}/m_{1,1} & -d_{1,2}/m_{1,1} & -d_{1,3}/m_{1,1} \\ 0 & 0 & 0 & -d_{2,1}/m_{2,2} & -d_{2,2}/m_{2,2} & -d_{2,3}/m_{2,2} \\ 0 & 0 & 0 & -d_{3,1}/m_{3,3} & -d_{3,2}/m_{3,3} & -d_{3,3}/m_{3,3} \end{bmatrix} \quad (4)$$

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1/m_{1,1} & 0 & 0 \\ 0 & 1/m_{2,2} & 0 \\ 0 & 0 & 1/m_{3,3} \end{bmatrix}$$

where $m_{i,j}$ are elements of the mass matrix \mathbf{M} and $d_{i,j}$ are elements of the damping matrix \mathbf{D} . Once the linearized system matrices have been calculated, the control matrix \mathbf{K} can be calculated as

$$\mathbf{K} = (\mathbf{R} + \mathbf{B}^T \mathbf{X} \mathbf{B})^{-1} (\mathbf{B}^T \mathbf{X} \mathbf{A}) \quad (5)$$

where \mathbf{X} is obtain by solving the discrete algebraic Ricatti equation as

$$\mathbf{A}^T \mathbf{X} \mathbf{A} - \mathbf{X} - \mathbf{A}^T \mathbf{X} \mathbf{B} (\mathbf{R} + \mathbf{B}^T \mathbf{X} \mathbf{B})^{-1} \mathbf{B}^T \mathbf{X} \mathbf{A} + \mathbf{Q} = 0 \quad (6)$$

This state formulation with integral effect will allow the surface vessel to reach an arbitrary point in its configuration space, given that no constant disturbances

are present - an assumption that will often be broken in the face of currents, waves and/or wind. As such, the actual state vector used for the LQR controller is defined as

$$\mathbf{x} = [x \ y \ \psi \ u \ v \ r \ ix \ iy]^T \quad (7)$$

where ix and iy are integral states for x and y positions respectively. With this state augmentation, the system matrices will be augmented as follows:

$$\tilde{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{0}_{6 \times 2} \\ \mathbf{I}_{2 \times 2} & \mathbf{0}_{2 \times 2} \end{bmatrix} \quad \text{and} \quad \tilde{\mathbf{B}} = \begin{bmatrix} \mathbf{B} \\ \mathbf{0}_{2 \times 3} \end{bmatrix} \quad (8)$$

Adding integral effect allows the ASV to maintain its setpoint through disturbances, but adds the risk of controller-windup, where the integral effect grows very large. To counteract this, a simple anti-windup feature is added where the integral state is clamped to a predefined range.

3.5 Path Following

The LQR controller is designed to move the ASV to some fixed setpoint. If the distance between the current position of the ASV and the setpoint is too large, the corresponding control outputs may also grow large. To counteract this, and simultaneously smooth the motion of the vessel, an interpolation scheme is used in conjunction with the controller, in a so-called LQR guidance implementation.

The LQR guidance scheme serves as an intermediary layer between higher-level path planning and the lower-level LQR controller. Given a set of waypoints, the LQR guidance system firstly linearly interpolates setpoints between the waypoints such that the distance between any two points does not exceed some defined step size. It then iteratively feeds these setpoints to the LQR controller, while maintaining logic to switch to the next setpoint when the ASV reaches a region of convergence on the setpoint. This region of convergence is also specified a priori.

3.6 Docking

A multi-step strategy, which integrates ArUco detection through a camera and distance estimation via LiDAR, has been employed to enable situational awareness and autopilot during the docking task. Initially, the ArUco detection and tracking algorithm is applied to determine a preliminary bearing and distance to the intended target. Following this, by tracking points using the LiDAR towards this estimated bearing, a much more precise pose (comprising both position and orientation) of the target docking location can be ascertained. Once this pose is transformed into the global reference frame, it can be inputted into the LQR guidance and control system. This allows the system to accurately navigate the vessel to the specified setpoint.

3.7 Collision Avoidance

To solve the collision avoidance task, a decision was taken to utilize an algorithm which would be neither too rudimentary (e.g. stop and wait if there is a dynamic obstacle) nor too advanced (e.g. scenario-based model predictive control) to understand and implement. For this reason, the Velocity Obstacles approach (VO) is used.

The Velocity Obstacle (VO) algorithm implements collision avoidance guidance by identifying potential collisions based on agent velocities which are involved in a probable collision scenario. For any two agents, A and B, a relative velocity is calculated. Using this, the VO determines a "collision cone" representing velocities leading to a collision. To calculate the cone, the estimated positions and velocities of the drone and the obstacle are used, as well as the radius of the vessels (in addition to a buffer on the sizes). To avoid collision, agent A selects a velocity outside this cone. When faced with multiple obstacles, combined VOs identify velocities to avoid. While VO helps in dynamic environments, adaptations are needed for specific constraints like non-holonomic motion.

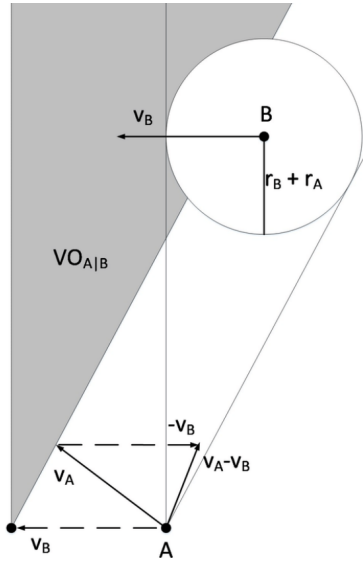


Figure 12: Velocity Obstacles (VO) algorithm visualization.

The difficult part of implementing the VO algorithm is to combine it together with a pathfollowing algorithm (LOS in our case). There needs to be a certain trigger for when the system goes from following the nominal path, to actively avoiding the collision and substantially increasing the cross-track error (transversal distance) to the desired path. The setpoints from the collision

avoidance directly contradicts what the path following system (LOS) is trying to achieve, which, if the systems are not integrated with this in mind, will most certainly lead oscillatory, unstable, and most importantly, unsafe, behaviour of the vessel. The final iteration of the system design, uses a simple solution of "safety zones", where being at a close enough distance to any dynamic object in the tracking zone, will shift the operation mode from "LOS" to "COLAV" (*COL*lision *AV*oidance). The nominal path following is then completely disabled until the immediate danger of a collision gone. This is determined both by the VO cone zone, and a logic scheme implemented according to Convention on the International Regulations for Preventing Collisions at Sea (COLREGS).

3.8 Architecture

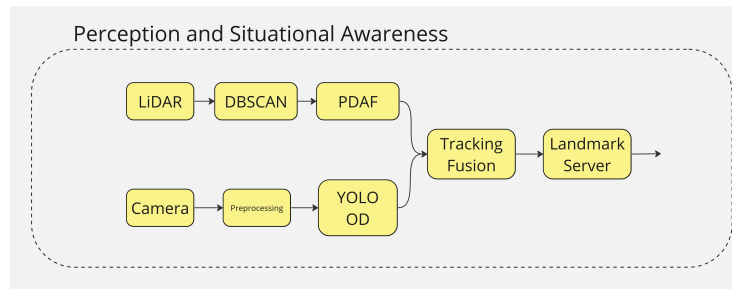


Figure 13: Perception and Situational Awareness software architecture.

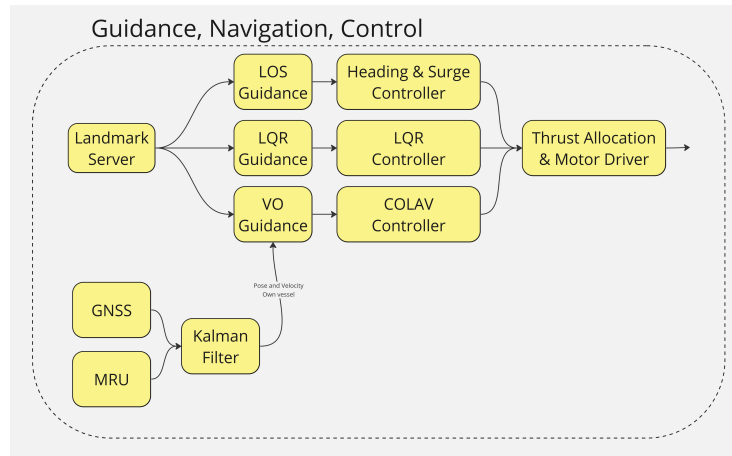


Figure 14: Guidance, Navigation, and Control (GNC) software architecture.

3.9 Implementation

Each of the modules belonging to the perception or control stack is implemented as it's own respective catkin package. To increase the readability and ease of debugging, every module is separated into two parts: one being responsible for the algorithm (eg. for detection) and the other responsible for interacting with the relevant ROS nodes.

4 Testing

4.1 Vessel testing

4.1.1 Waterproof compartments

The team succeeded in making waterproof compartments from scratch. This could not have been done without extensive testing. Both the electrical housing and battery compartments were tested in the same manner, in gradually harsher conditions. Firstly, they were filled with water, to see if any water leaked out through the seal created by the glue and the silicone. After the team was happy with the results, it was time to test the homemade O-ring solution. The lids were attached with the latches, before adding locking pins to ensure that the latches did not loosen during testing. Each compartment was then completely submerged for several minutes as shown in figure 15.

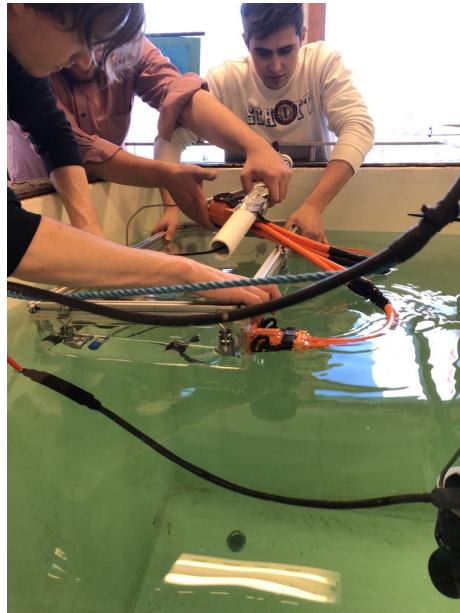


Figure 15: El-house being submerged to find leaks.

Upon inspection, no water had leaked into any of the homemade compartments! This test was not strictly necessary, since the compartments are never meant to be submerged. However, the test proved that in a worst case scenario, essential components would still be kept dry. The battery compartments were stress tested as well. A battery compartment was filled with weights, tied to a rope, and gradually lowered into the pool. At a depth of about 4 meters, one of the long walls buckled, breaking the seal and allowing water to rush in. Until this point, no water had leaked into the compartment. Upon making this discovery, the team added 3D-printed stiffeners to the lid, preventing the same thing from happening again. The test was repeated, and no water leaked in.

The pontoons were also rigorously tested. The first float test was done in a pool where weights were added to test the balance and the maximum payload as shown in figure 16. The key takeaway from this was that it the importance to fill the holes for the batteries to not get weighed down by water.

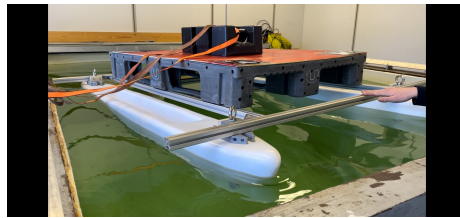


Figure 16: First float test with 76 kg payload.

The ASV was also tested in a wave generator at NTNU (see figure 17). The tested showed that the ASV was very stable sideways and glided nicely over the wave, but it did rock a bit too much in the forward direction. This was corrected by lowering the center of mass.

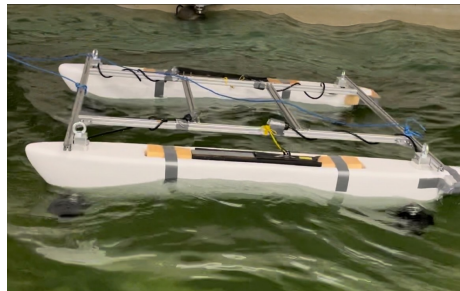


Figure 17: ASV tested in a wave generator pool.

4.2 Hardware testing

4.2.1 Power Distribution Board

The PDB was verified by inputting power and confirming the expected voltage on the output rails, evident from onboard status lights. Testing the fuse status lights involved simulating malfunction by removing fuses, causing the red status lights to illuminate.

4.2.2 Fail Safe PCB

The Fail Safe PCB has been tested by triggering all Kill Switches and verifying that the solid state relays are opened as intended. Similarly it has been verified that the status light switches colours according to specifications. It has also been tested that a catastrophic error in the Fail Safe PCB results in the relays opening and disabling thrusters. This was destructively tested by setting battery voltage on the 5V bus of the Fail Safe PCB.

4.2.3 Batteries

After completing the battery assembly and connecting the BMS, our initial tests began. Using specialized BMS software, cell status data was gathered, including voltage and temperature, which appeared satisfactory. A conservative current limit was set in the BMS and the battery was connected to a suitably rated resistor. By measuring the discharged current with an ammeter, it was confirmed that the BMS achieved an accurate power cutoff at the intended threshold. With the obtained results instilling confidence, the batteries were then connected to the El-house for a comprehensive and successful full test.

4.2.4 Radio system

The range limit has not yet been tested for the radio system, but we have verified that it works in a controlled environment by sending and receiving data over air.

4.2.5 GNSS

The GNSS system has been tested outdoor using a software program provided by Kongsberg.

4.3 Software testing

With regards to detection based algorithms such as DBSCAN, YOLOv5 and Aruco detection, the algorithms were simply tested at the office to have a proof of concept before starting further development. Once a proof of concept had been attained, work towards optimizing the algorithm, and adapting the algorithm to work on the single boards computers aboard Freya was done. Once a implementation was working on Freya, the robustness of the algorithm could

be further tested by using recorded data, or simply dry-testing the vessel. For PDAF both simulated data, as well as live testing at a pool was done in order to validate the algorithm.

4.4 Integrated testing

For testing all of the ASV, Freya was taken out to Nyhavna. The whole process of assembling the parts of Freya was practiced to manually controlling Freya and testing the communication for range between the FrSky antennas and the controller.

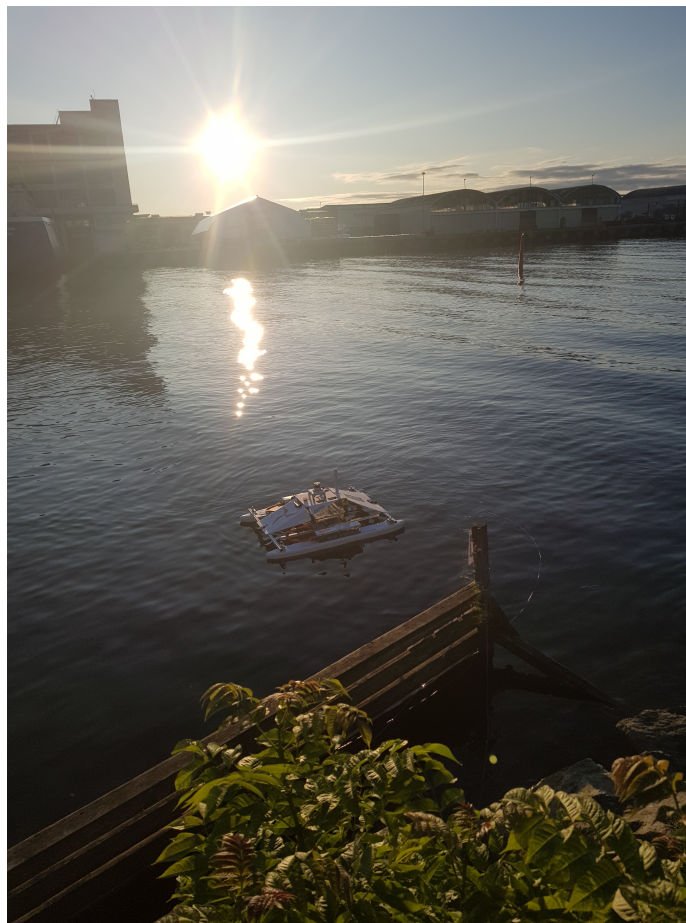
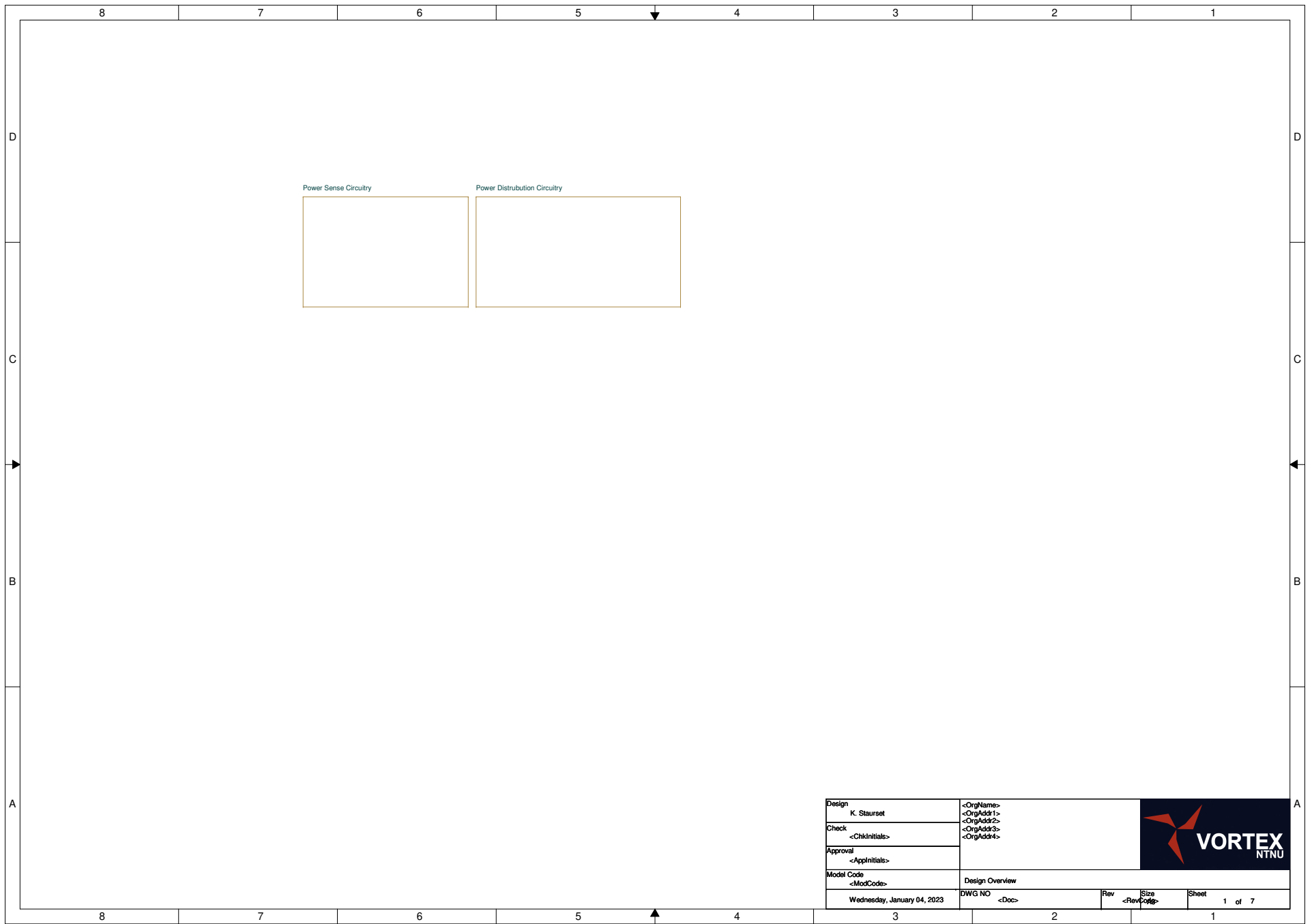



Figure 18: Outside testing of Freya at Nyhavna, Trondheim.

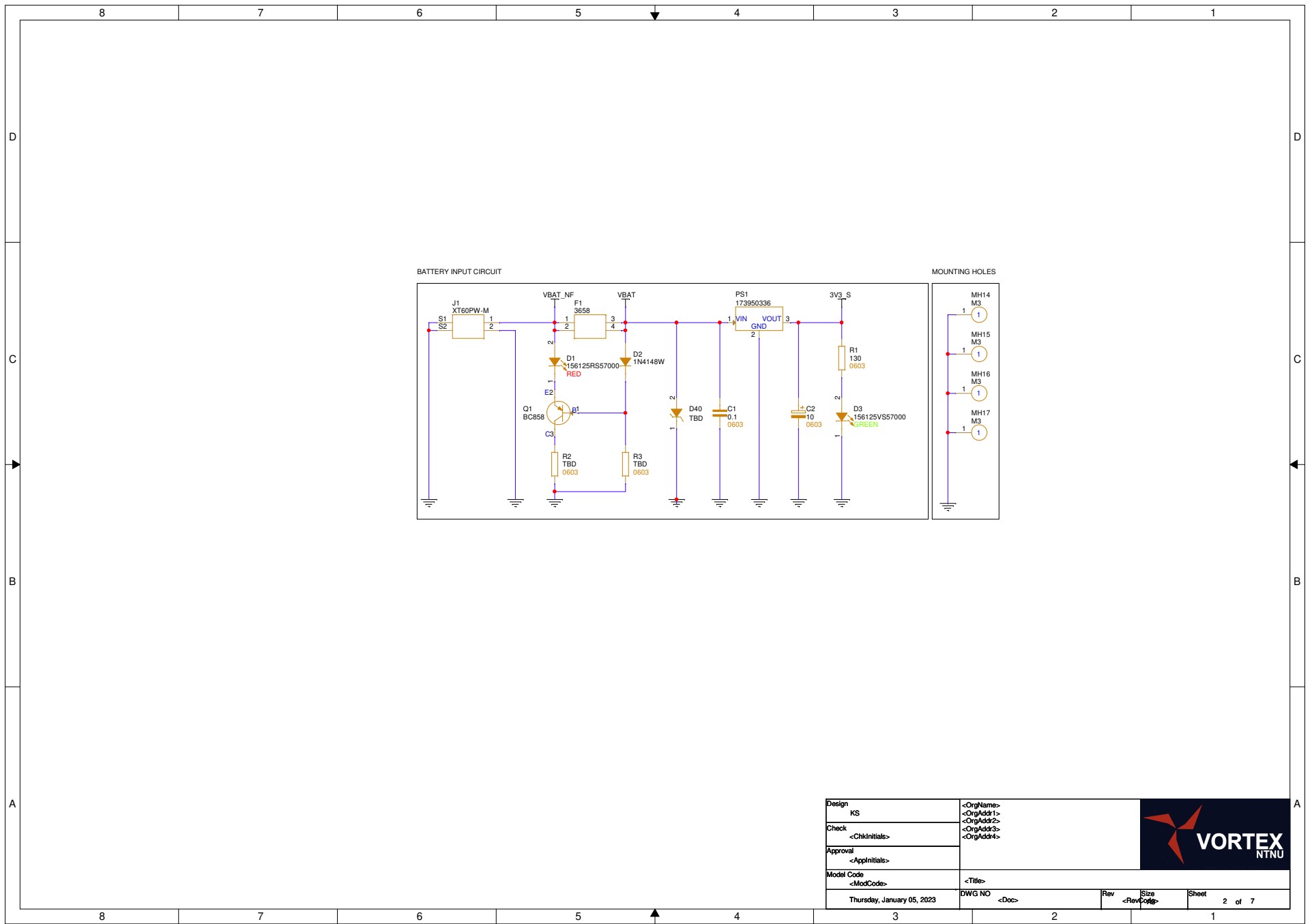
5 Appendix


A Power Distribution Board

The following 7 pages show the schematics of the PDB.

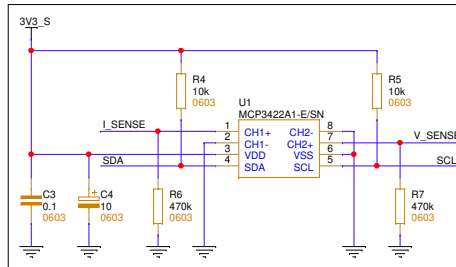


Design	K. Staurset		<OrgName>				
Check	<ChkInitials>		<OrgAddr1> <OrgAddr2> <OrgAddr3> <OrgAddr4>				
Approval	<AppInitials>						
Model Code	<ModCode>		Design Overview				
	DWG NO	<Doc>	Rev	<RevCode>	Size	Sheet	1 of 7

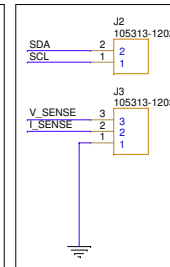



Design	KS	<OrgName>	
Check	<ChkInitials>	<OrgAddr1>	
Approval	<AppInitials>	<OrgAddr2>	
Model Code	<ModCode>	<OrgAddr3>	
	Thursday, January 05, 2023	DWG NO <Doc>	Rev <RevCode>
		Size	Sheet 2 of 7

POWER SENSE CIRCUIT

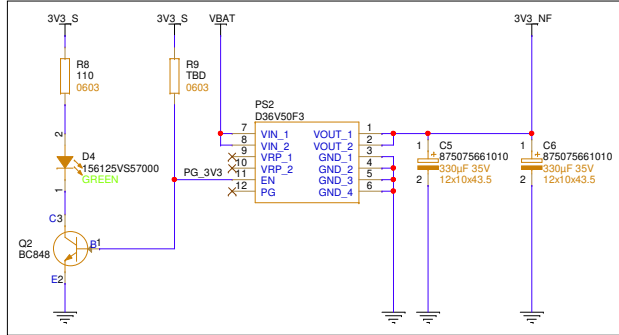


POWER SENSE CONNECTORS

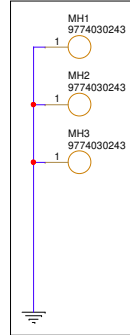


Design	KS	<OrgName>			
Check	<ChkInitials>	<OrgAddr1>			
Approval	<AppInitials>	<OrgAddr2>			
Model Code	<ModCode>	<OrgAddr3>			
	Wednesday, January 04, 2023	DWG NO <Doc>	Rev <RevCode>	Size <Size>	Sheet 3 of 7

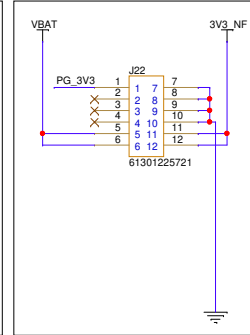
POWER MODULE CIRCUIT



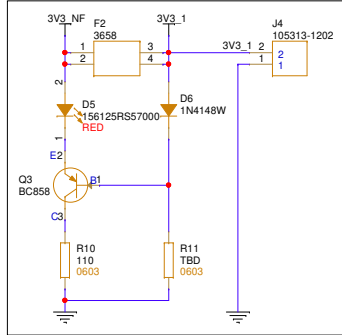
MOUNTING SPACERS



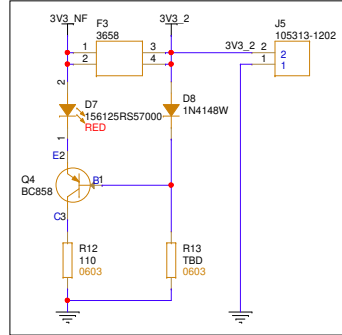
SOCKET HEADERS



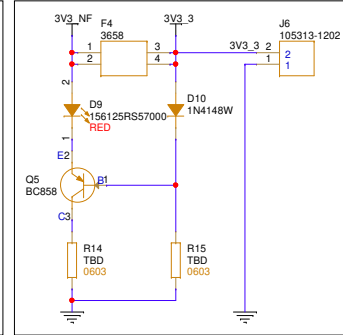
FUSE CIRCUIT 1



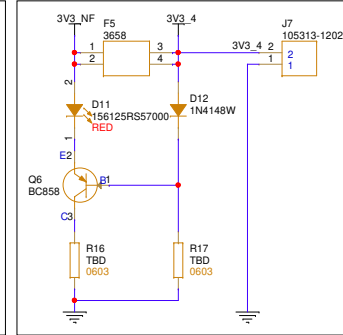
FUSE CIRCUIT 2




FUSE CIRCUIT 3

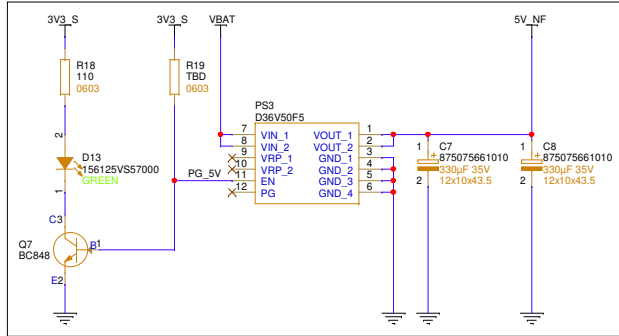


FUSE CIRCUIT 4

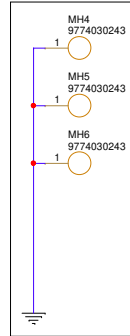


Design	KS	<OrgName>	
Check	<ChkInitials>	<OrgAddr1>	
Approval	<AppInitials>	<OrgAddr2>	
Model Code	<ModCode>	<OrgAddr3>	
	Tuesday, January 10, 2023	DWG NO	<Doc>
		Rev	<RevCode>
		Size	<Size>
		Sheet	4 of 7

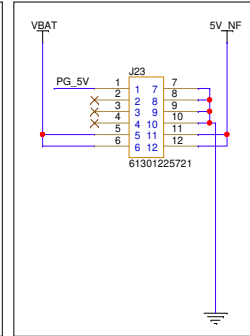
POWER MODULE CIRCUIT



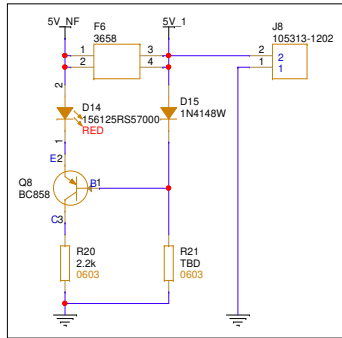
MOUNTING SPACERS



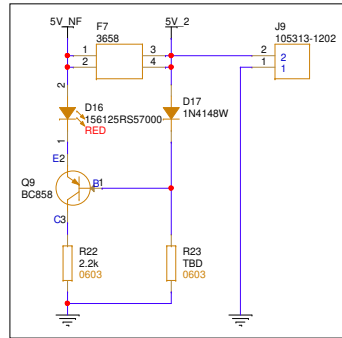
SOCKET HEADERS



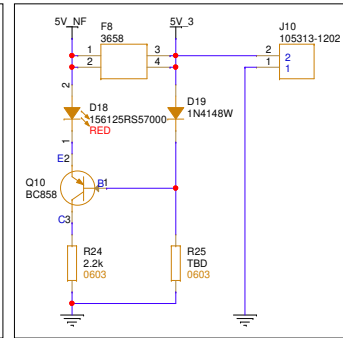
FUSE CIRCUIT 1



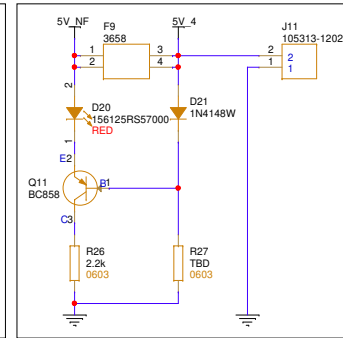
FUSE CIRCUIT 2




FUSE CIRCUIT 3

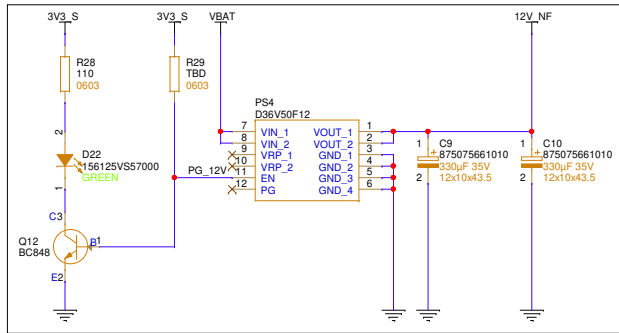


FUSE CIRCUIT 4

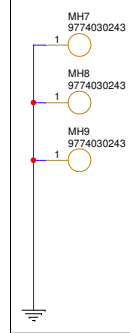


Design	KS	<OrgName>			
Check	<ChkInitials>	<OrgAddr1>			
Approval	<AppInitials>	<OrgAddr2>			
Model Code	<ModCode>	<OrgAddr3>			
	Monday, January 09, 2023	<OrgAddr4>	<Title>	DWG NO	<Doc>
		Rev	Size	Sheet	5 of 7

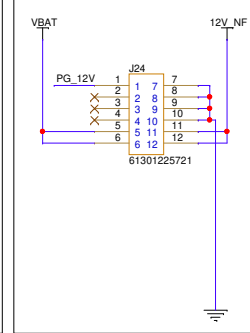
POWER MODULE CIRCUIT



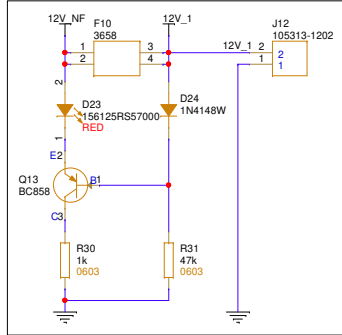
MOUNTING SPACERS



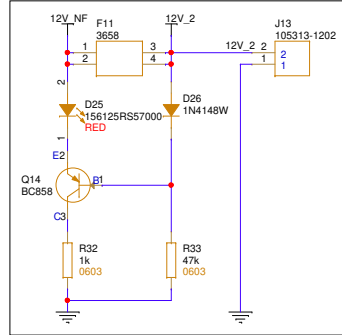
SOCKET HEADERS



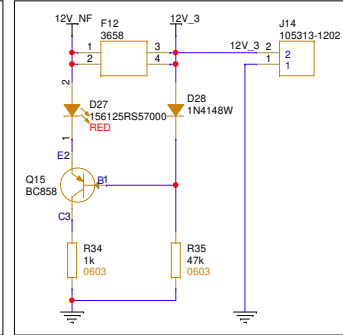
FUSE CIRCUIT 1



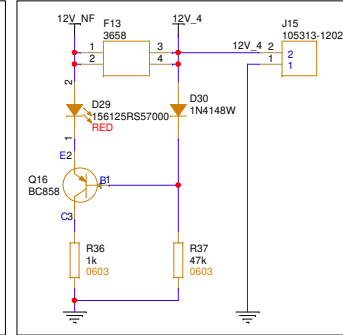
FUSE CIRCUIT 2




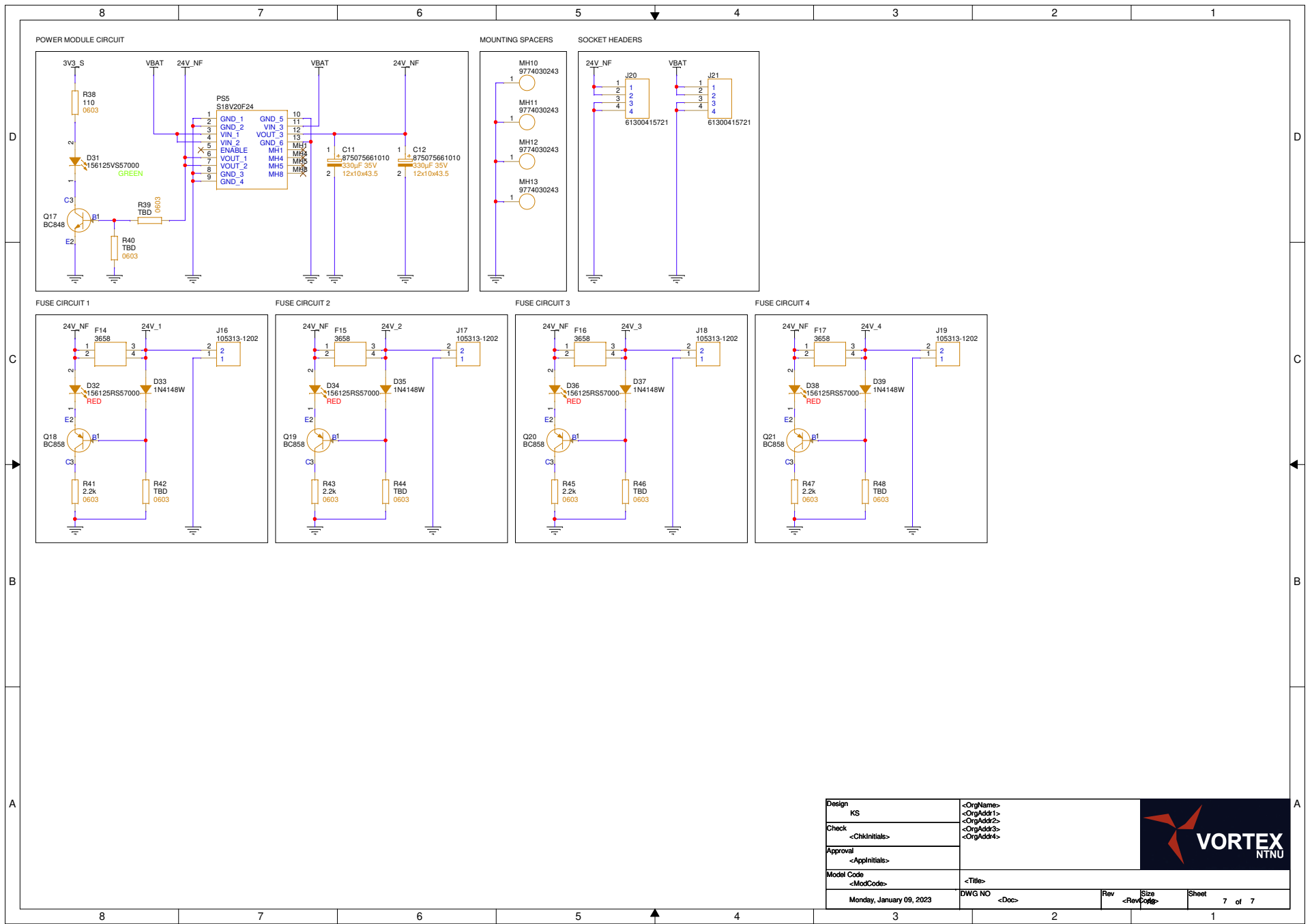
FUSE CIRCUIT 3




FUSE CIRCUIT 4



Design	KS	<OrgName>						
Check	<ChkInitials>	<OrgAddr1>						
Approval	<AppInitials>	<OrgAddr2>						
Model Code	<ModCode>	<OrgAddr3>						
	Tuesday, January 10, 2023	DWG NO	<Doc>	Rev	<RevCode>	Size	Sheet	6 of 7



Design	KS	<OrgName>			
Check	<ChkInitials>	<OrgAddr1>			
Approval	<AppInitials>	<OrgAddr2>			
Model Code	<ModCode>	<OrgAddr3>			
	Monday, January 09, 2023	DWG NO	<Doc>	Rev	<RevCode>
		Size	<Size>	Sheet	7 of 7

B Fail Safe PCB

Figure 19 shows the upper layout of the Fail Safe PCB.

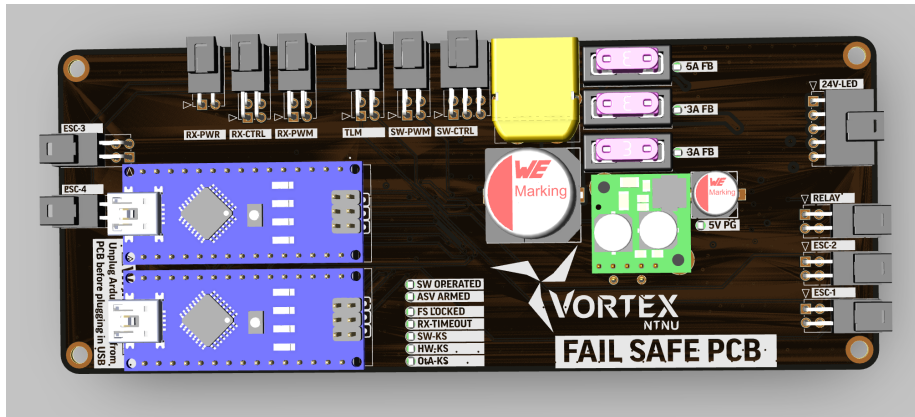
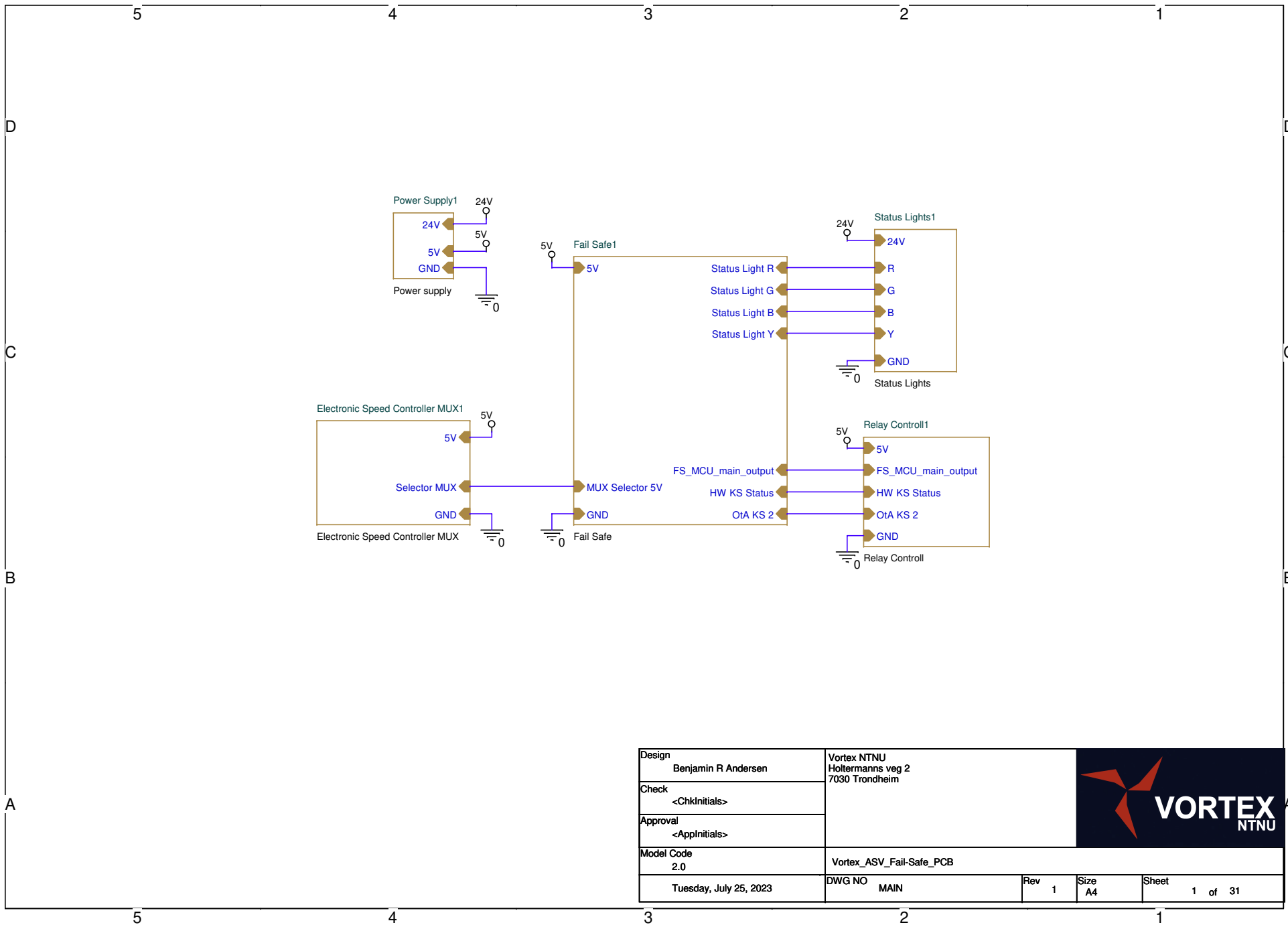

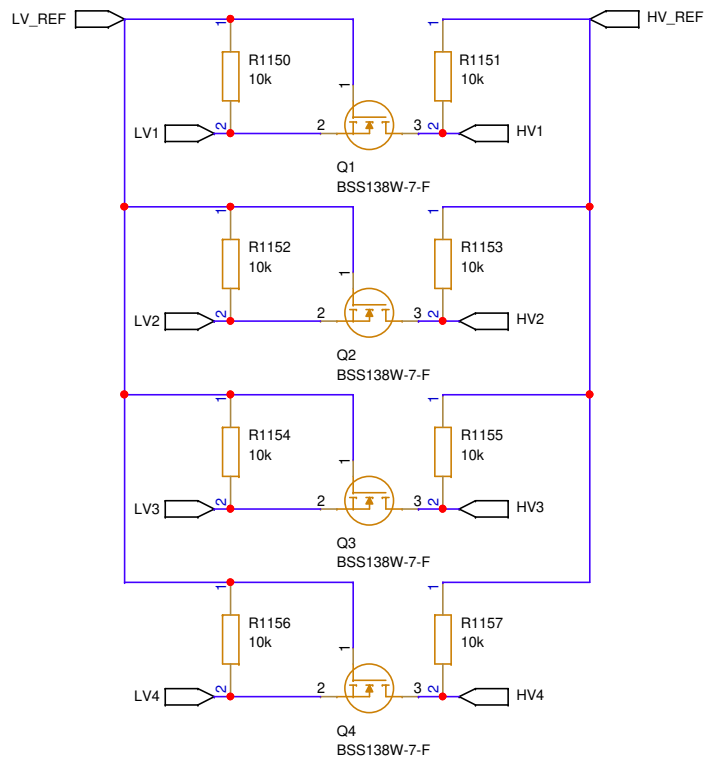



Figure 19: 3d-render of the FS PCB.

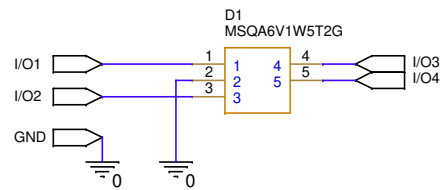
The following 14 pages show the schematics of the Fail Safe PCB.




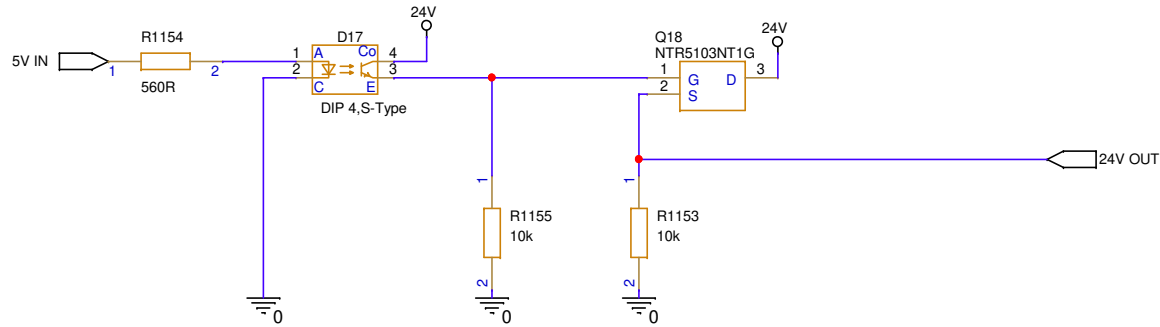
Design	Benjamin R Andersen	Vortex NTNU Holtermanns veg 2 7030 Trondheim						
Check	<ChkInitials>							
Approval	<AppInitials>							
Model Code	2.0	Vortex_ASV_Fail-Safe_PCB						
Tuesday, July 25, 2023	DWG NO	MAIN	Rev	1	Size	A4	Sheet	1 of 31




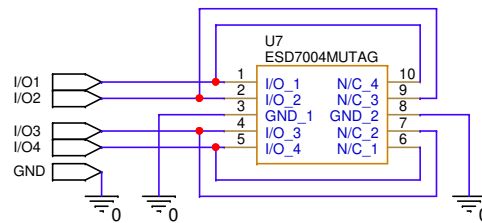
Design Benjamin R Andersen	Vortex NTNU Holtermanns veg 2 7030 Trondheim			
Check <ChkInitials>				
Approval <AppInitials>				
Model Code 1.2	Vortex_ASV_Fail-Safe_PCB			
Tuesday, April 04, 2023	DWG NO Logic Level Converter	Rev 1	Size A4	Sheet 1 of 6




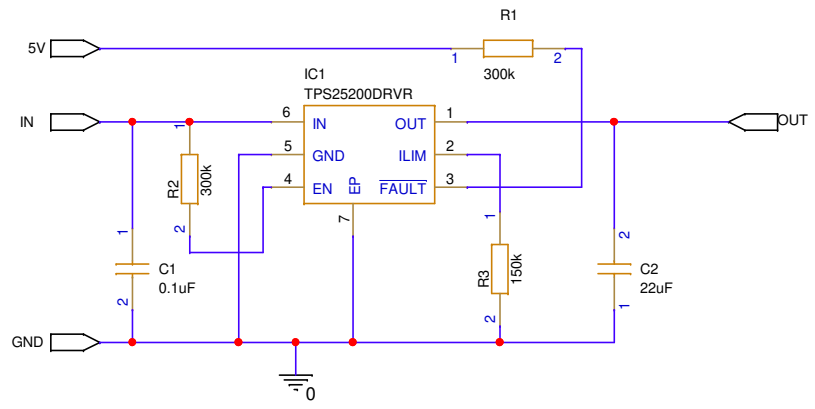
Design Benjamin R Andersen	Vortex NTNU Holtermanns veg 2 7030 Trondheim			
Check <ChkInitials>				
Approval <AppInitials>				
Model Code 1	Vortex_ASV_Fail-Safe_PCB			
Tuesday, April 04, 2023	DWG NO 3V3 TVS diode	Rev 1	Size A4	Sheet 2 of 31




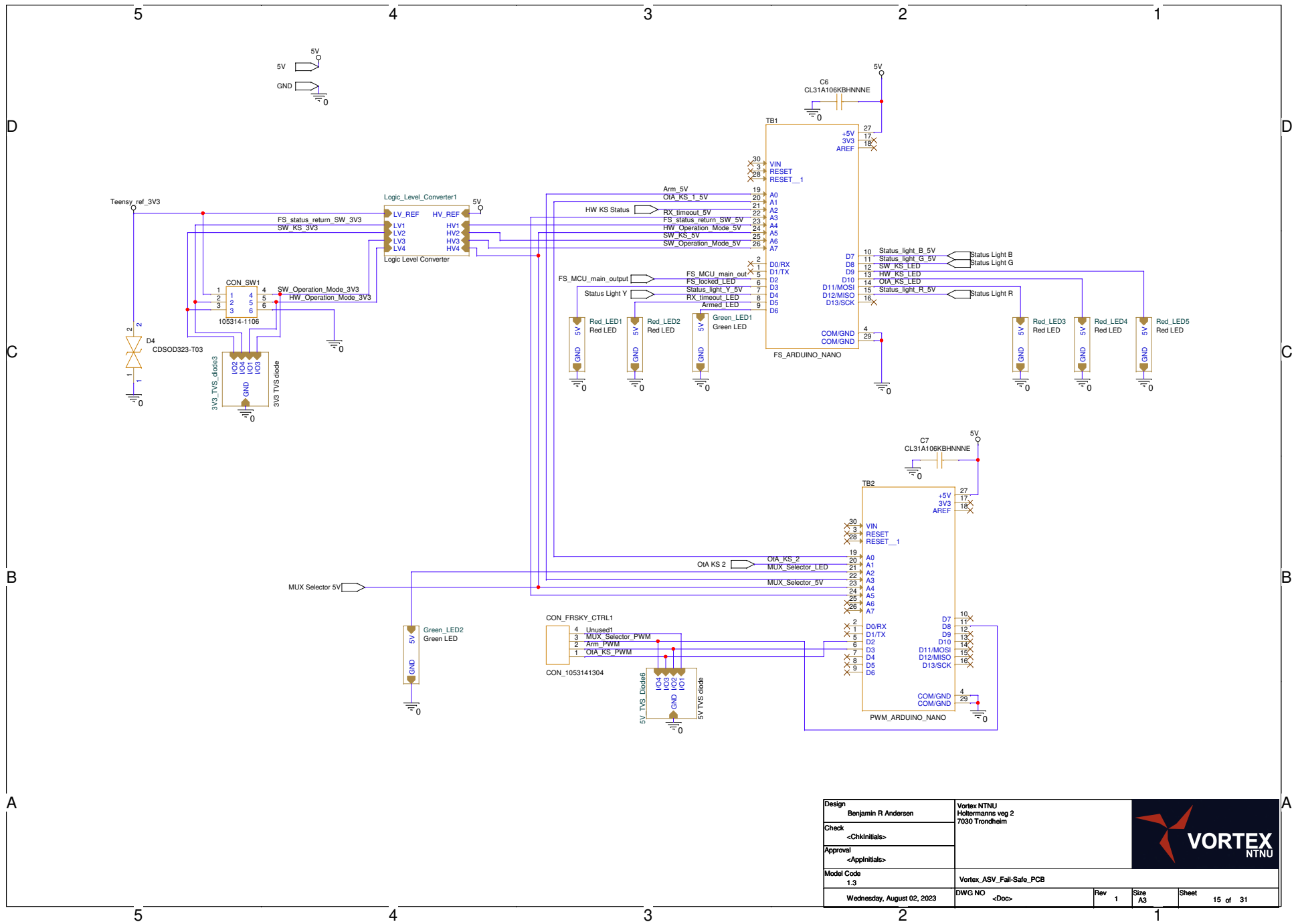
Design	Benjamin R Andersen	Vortex NTNU Holtermanns veg 2 7030 Trondheim			
Check	<ChkInitials>				
Approval	<AppInitials>				
Model Code	1	Vortex_ASV_Fail-Safe_PCB			
Tuesday, April 04, 2023	DWG NO	Status_light_source_follower	Rev 1	Size A4	Sheet 6 of 6




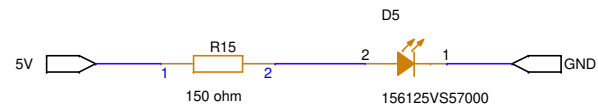
Design Benjamin R Andersen	Vortex NTNU Holtermanns veg 2 7030 Trondheim							
Check <ChkInitials>								
Approval <AppInitials>								
Model Code 1.2	Vortex_ASV_Fail-Safe_PCB							
Monday, January 23, 2023	DWG NO	5V TVS diode	Rev	1	Size	A4	Sheet	11 of 31




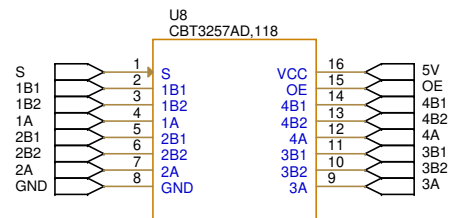
Design	Benjamin R Andersen	Vortex NTNU Holtermanns veg 2 7030 Trondheim						
Check	<ChkInitials>							
Approval	<AppInitials>							
Model Code	0.7	Vortex_ASV_Fail-Safe_PCB						
Friday, August 11, 2023	DWG NO	5V_E-Fuse	Rev	1	Size	A4	Sheet	12 of 31




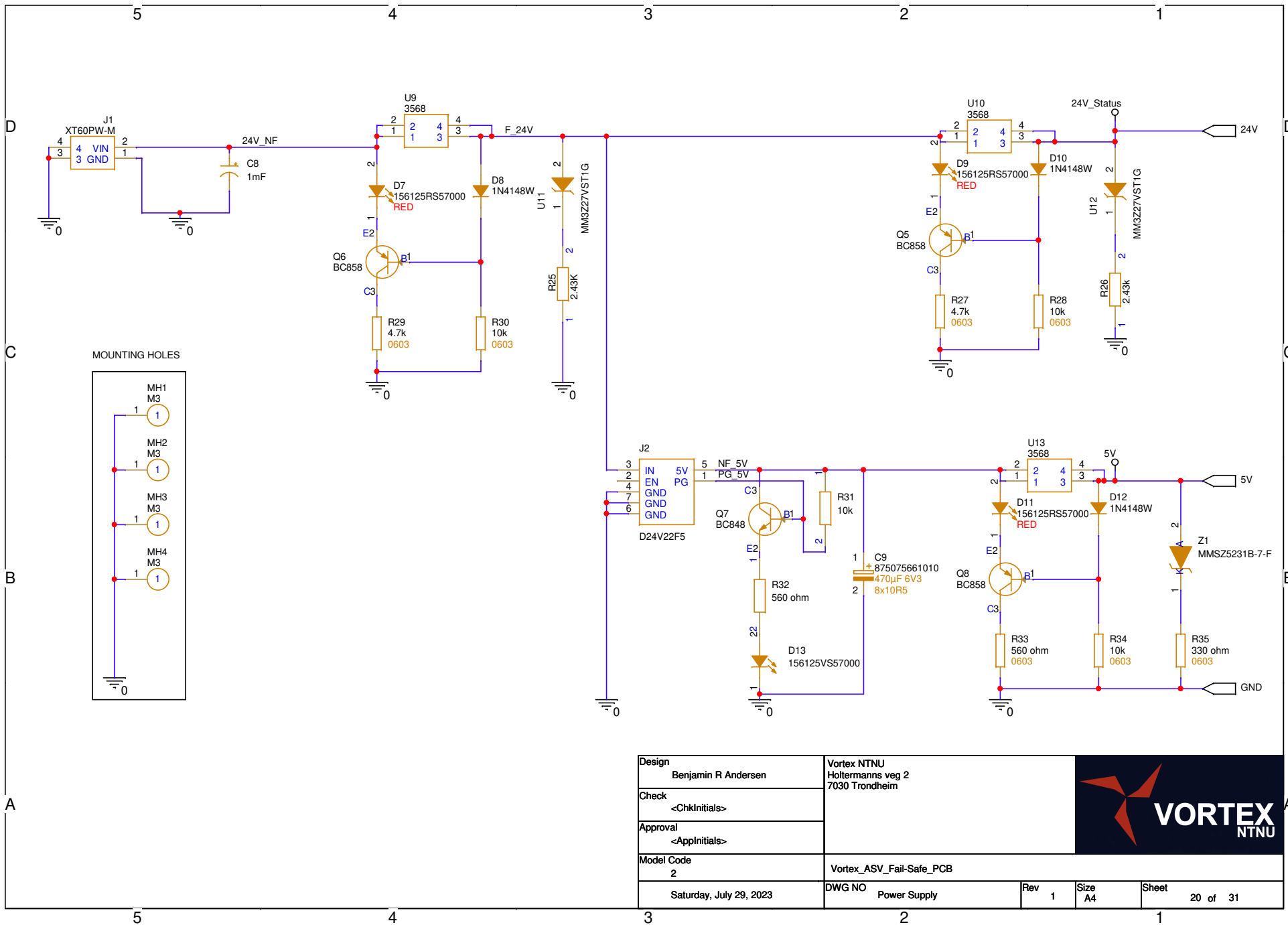
Design	Benjamin R Andersen	Vortex NTNU Høllermanns veg 2 7030 Trondheim	
Check	<ChkInitials>		
Approval	<AppInitials>		
Model Code	1.3	Vortex_ASV_Fail-Safe_PCB	
Wednesday, August 02, 2023	DWG NO <Doc>	Rev 1	Size A3
		Sheet	15 of 31




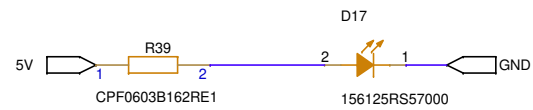
Design Benjamin R Andersen	Vortex NTNU Holtermanns veg 2 7030 Trondheim			
Check <ChkInitials>				
Approval <AppInitials>				
Model Code 0.7	Vortex_ASV_Fail-Safe_PCB			
Tuesday, April 04, 2023	DWG NO Green_LED	Rev 1	Size A4	Sheet 16 of 31




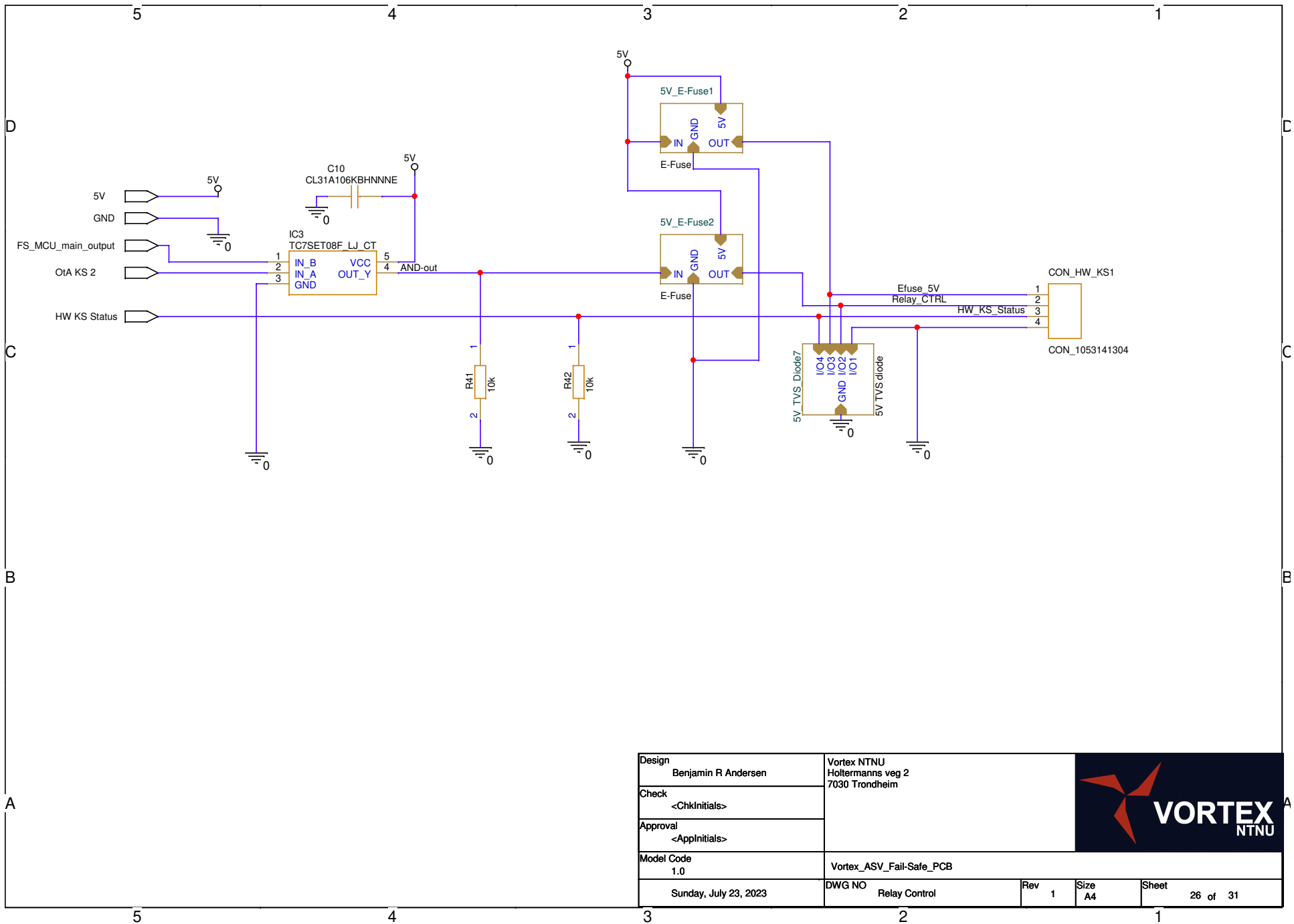
Design	Benjamin R Andersen	Vortex NTNU Holtermanns veg 2 7030 Trondheim						
Check	<ChkInitials>							
Approval	<AppInitials>							
Model Code	1.0	Vortex_ASV_Fail-Safe_PCB						
Monday, January 23, 2023	DWG NO	MUX	Rev	1	Size	A4	Sheet	19 of 31




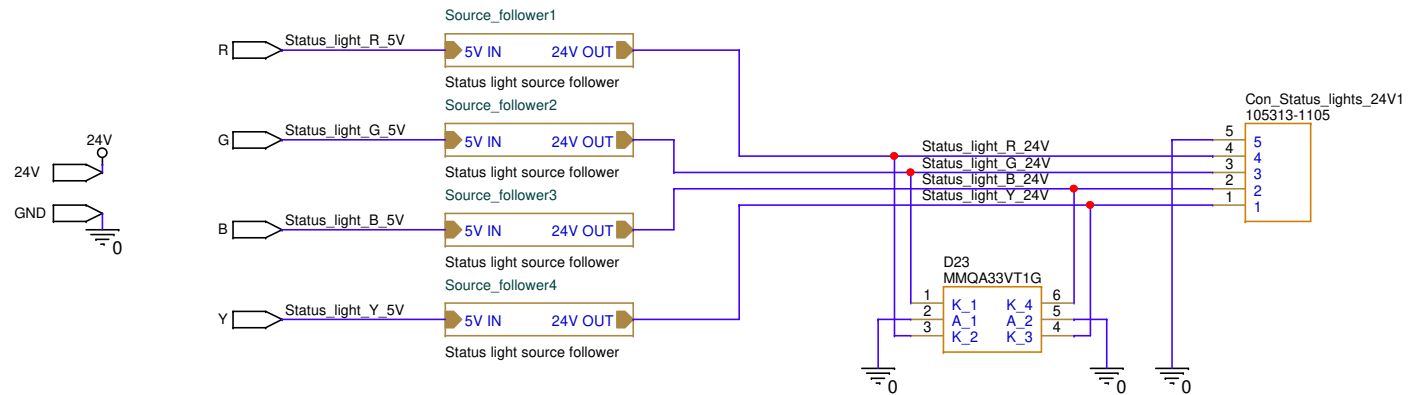
Design	Benjamin R Andersen		Vortex NTNU Holtermanns veg 2 7030 Trondheim	
Check	<Chkinitials>			
Approval	<Appinitials>			
Model Code	2			
	Vortex_ASV_Fail-Safe_PCB			
Saturday, July 29, 2023	DWG NO	Power Supply	Rev	1
	Size	A4	Sheet	20 of 31




Design Benjamin R Andersen	Vortex NTNU Holtermanns veg 2 7030 Trondheim			
Check <ChkInitials>				
Approval <AppInitials>				
Model Code 0.7	Vortex_ASV_Fail-Safe_PCB			
Tuesday, April 04, 2023	DWG NO Red_LED	Rev 1	Size A4	Sheet 24 of 31



Design Benjamin R Andersen	Vortex NTNU Hoftermanns veg 2 7030 Trondheim			
Check <ChkInitials>				
Approval <AppInitials>				
Model Code 1.0	Vortex_ASV_Fail-Safe_PCB			
Sunday, July 23, 2023	DWG NO Relay Control	Rev 1	Size A4	Sheet 26 of 31



Design Benjamin R Andersen	Vortex NTNU Hollermanns veg 2 7030 Trondheim			
Check <ChkInitials>				
Approval <AppInitials>				
Model Code 1.0	Vortex_ASV_Fail-Safe_PCB			
Tuesday, April 04, 2023	DWG NO Status lights	Rev 1	Size A4	Sheet 31 of 31