S.O.T.A

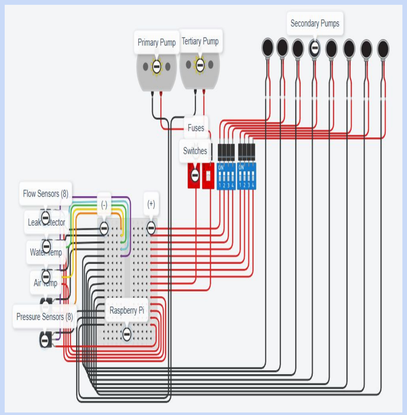
Sorbent On-Site Testing Apparatus

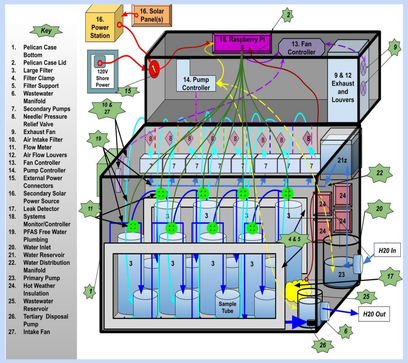
Trey Lundberg Chase White Kelvin Huang Amory Quinn

SDSU Mechanical Engineering

Problem

The Naval Information Warfare Systems Command is developing a testing apparatus as part of the NESDI program for the purpose of identifying Per- and polyfluoroalkyl substances (PFAS), primarily for possibly contaminated water supplies found in naval facilities and ships. PFAS is a synthetic chemical which has been linked to an increased risk in cancer amongst those exposed, and has become an increasingly problematic issue that requires action.

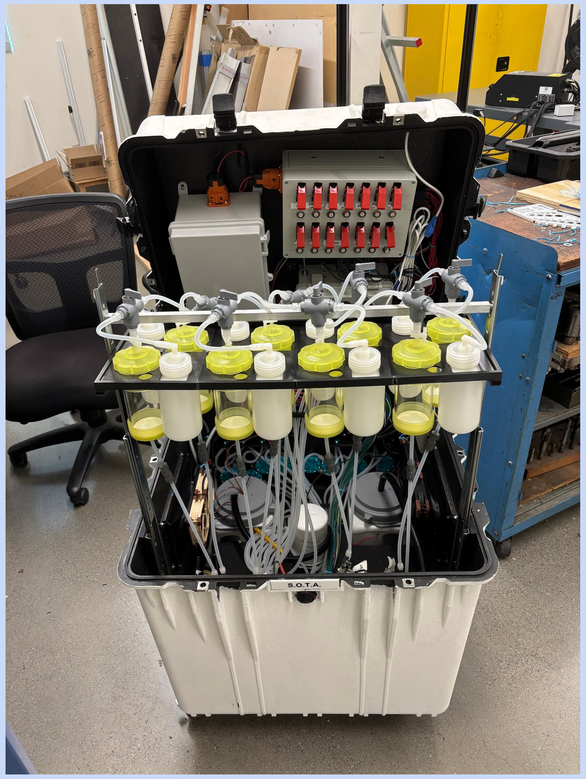




Solution

A compact system which incorporates hydraulics, electronics, and additive-manufactured components has been developed with remote-access and solar-power capabilities to provide reliable testing data regarding the concentration of PFAS materials in a given water source. The developed apparatus is not only durable, but is wholly contained within a portable and relatively lightweight mobile case (Not including the external solar panels and power bank).

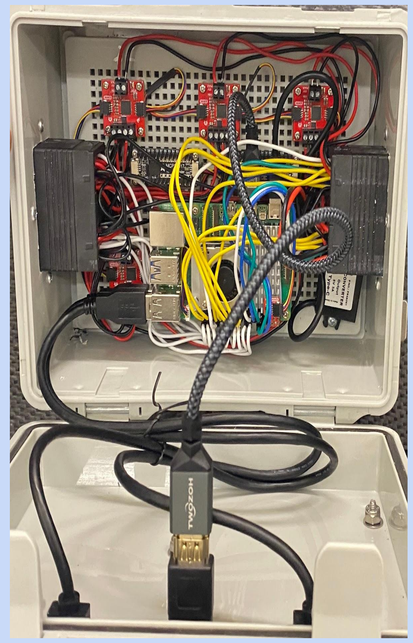
Final Design



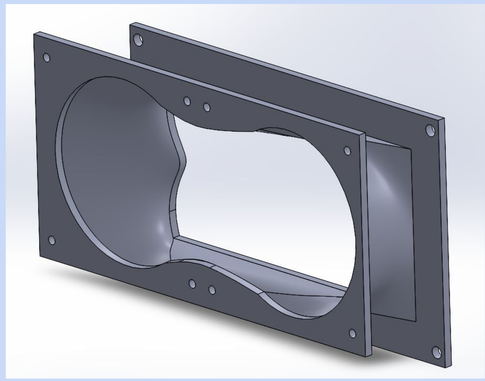
Python Code

```
# I2C addresses for the motor drivers
MOTOR_DRIVER_ADDRESSES = [0x5D, 0x5E, 0x5F, 0x61]

# Initialize the motor drivers
motor_drivers = []
for address in MOTOR_DRIVER_ADDRESSES:
    motor_driver = qwiic_scmd.QwiicScmd(address)
    if motor_driver.connected:
        motor_driver.begin()
        motor_driver.enable()
        motor_drivers.append(motor_driver)
        print(f"Motor driver at address {hex(address)} initialized.")
    else:
        print(f"Motor driver at address {hex(address)} not found.")

# Function to set pump flow rate (0 to 255, with 255 as maximum speed)
def set_pump_flow_rate(motor_driver, pump_channel, flow_rate):
    flow_rate = max(0, min(flow_rate, 255))  # Ensure flow_rate is within
valid range
    motor_driver.set_drive(pump_channel, 0, flow_rate)  # 0 for forward
direction
    print(f"Pump on driver {hex(motor_driver.address)} channel {pump_channel}
set to flow rate {flow_rate}.")

# Function to stop a pump
def stop_pump(motor_driver, pump_channel):
    motor_driver.set_drive(pump_channel, 0, 0)
    print(f"Pump on driver {hex(motor_driver.address)} channel {pump_channel}
stopped.")

# Set desired flow rate (adjust as needed)
desired_flow_rate = 128  # Value between 0 (stopped) and 255 (maximum speed)
```

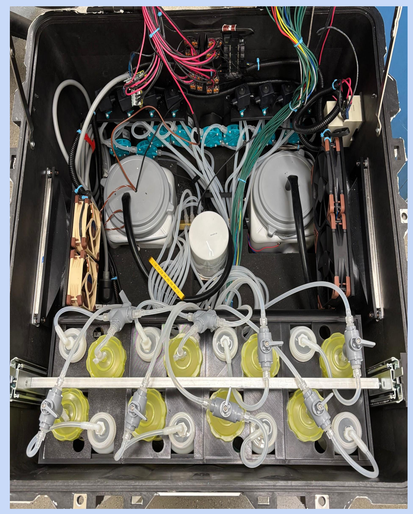
Electrical



CAD Modeling



Hydraulic



Fall 2024