NFPA - FPVC Hydraulic Bike

A Baccalaureate thesis submitted to the
Department of Mechanical and Materials Engineering
College of Engineering and Applied Science
University of Cincinnati

in partial fulfillment of the
requirements for the degree of

Bachelor of Science

in Mechanical Engineering Technology

by

Ian Gilbertsen
Brandon Walsh
Nathan Blickensderfer
Kirk Stephenson
Zachary Selby

April 2022

Thesis Advisor:
Professor Muthar Al-Ubaidi, Ph.D.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

A hydraulic vehicle is one which uses fluid, specifically hydraulic oil, to transform mechanical energy provided by a rider into fluid energy, then back into mechanical energy to propel the vehicle. This is done using hydraulic pumps, motors, valves, and tubing along with mechanical components such as sprockets, shafts, and wheels. This can be accomplished using driven components such as electric motors or via human energy, thus our vehicle is designed to be powered with human energy as is for a standard bicycle.

Special Thanks

Our team would like to formally thank everyone who has helped us design, build, and test our bike. Special thanks to our mentor, Dan Turner of GPM Controls for his guidance and attendance at our weekly meetings. We would also like to thank Professor Abigail Yee for her lectures on hydraulic fundamentals, mainly on pumps, motors, and valves. Thanks to Bob and Jim Sheaf at CFC Industrial training. Through their time and knowledge, our team learned how to bend our circuit tubes.

PROBLEM DEFINITION

PROBLEM STATEMENT

The goal of this challenge is to design and build a vehicle which is powered by hydraulics and human input. This design project is a submission to the Fluid Power Vehicle Challenge, sponsored by the National Fluid Power Association (NFPA). Specifically, the University of Cincinnati is in the Danfoss Competition, as opposed to the Norgren Power Competition. This challenge is an annual event where schools across the country design, build, and race their hydraulic vehicles. The competition itself is comprised of three separate challenges/races including an endurance, sprint, and efficiency race.

1. **Sprint Race**
   a. This event will demonstrate the ability of the vehicle to move a distance where the weight of the vehicle is proportional to the human propulsion.
2. **Durability Race**
   a. This event will demonstrate the reliability, safety, replicability, and durability of the fluid power system design and assembly.
3. **Efficiency Challenge**
   a. This event will demonstrate the ability of the vehicle to effectively store and most efficiently use the smallest amount of stored energy to propel the unassisted vehicle the greatest distance proportional to the vehicle's weight.

DESIGN OVERVIEW

Vehicle Concept

What separates our design from previous years is a new concept involving the use of multiple hydraulic pumps and motors as well as sophisticated electronic controls to regulate vehicle performance. Our bike uses two hydraulic gear pumps, one of larger displacement and one of smaller displacement, as well as two hydraulic gear motors, one smaller and one larger, to achieve multiple flow and pressure combinations in the system. The objective here is to create a more comfortable ride by adjusting torque and rpm ratios when necessary. Variable torque and rpm ratios also aid in vehicle acceleration and energy optimization. A complex control system is used to automatically switch which pumps and motors are actively working in the system. More specifically, feedback from inductive sensors and pressure transducers effectively create an automatic transmission. We created a touch screen HMI to interact with the vehicle as well as display performance data in the form of gauges, indicators, and text boxes.

Hydraulic System

Per competition rules and regulations, our vehicle has four drive modes: direct drive, accumulator charge, accumulator drive, and regenerative braking. Each drive mode can be thought of as its own flow path. Flow paths (drive modes) are determined and changed using directional valves placed throughout the circuit. Each drive mode serves a specific purpose; however, all drive modes are part of one, all-encompassing system. Figure 1 below shows our entire hydraulic circuit including all pumps, motors, valves, pressure transducers, accumulator, and manifold. Table 1 below lists all components of our circuit according to the items labeled in the circuit.

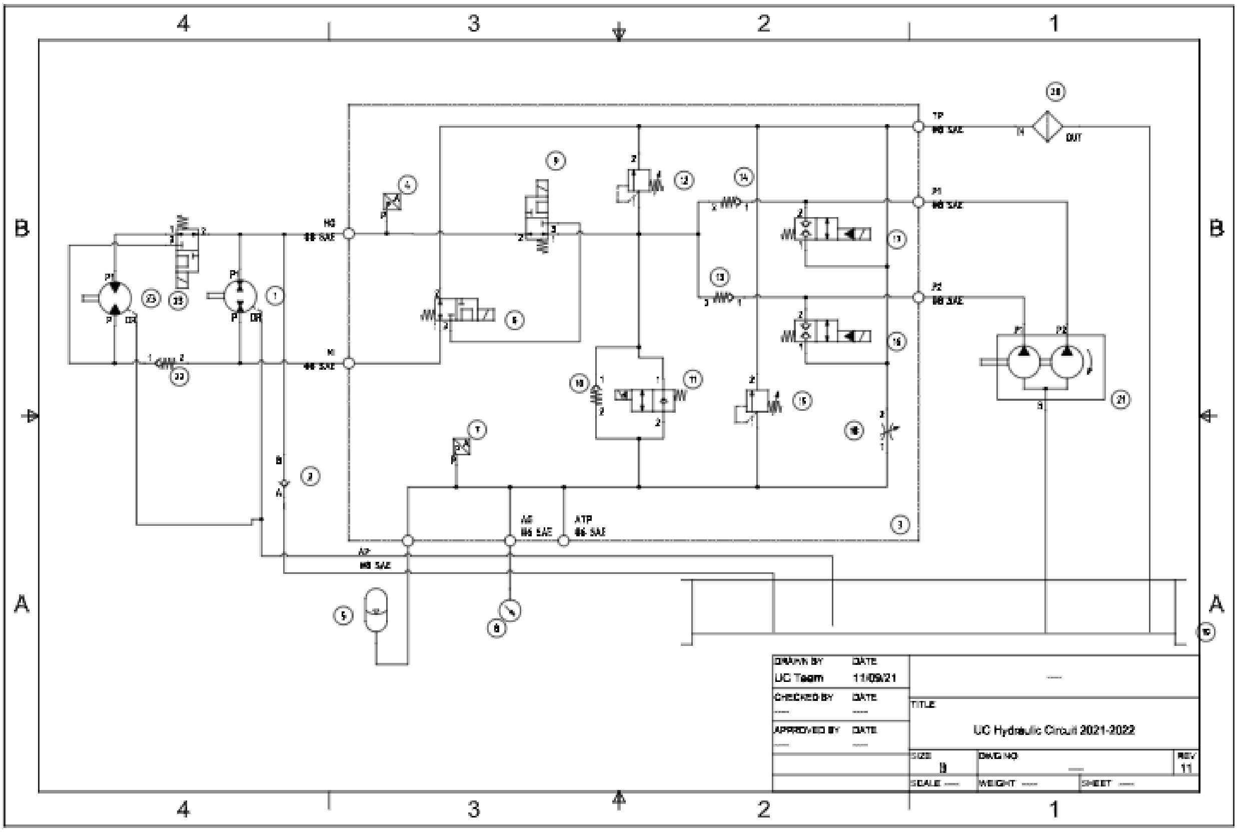

Figure 1: Hydraulic Circuit Schematic

ITEM	QTY	MODEL CODE	DESCRIPTION	MANUFACTURER
1	1	ALM1A-R-7-E2	GEAR MOTOR (.317 CIR)	MARZOCCHI
2	1	CV3-10	CHECK VALVE SIZE 10 (EXTERIOR, SPRING REMOVED)	EATON
3	1	N/A	MANIFOLD BLOCK	SUNSOURCE/SOURCE FLUID POWER
4, 7	2	HDA-847K-G-3000-404	ANALOG PRESSURE TRANSDUCER	HYDAC
5	1	N/A	CARBON FIBER ACCUMULATOR (1 GALLON)	N/A
6, 9, 23	3	SV1-10-3	3/2 DIRECTIONAL	EATON
8	1	N/A	ACCUMULATOR PRESSURE GAUGE	N/A
10, 13, 14	3	CV3-8	CHECK VALVE SIZE 8	EATON
11	1	SVP08-NCR	2/2 POPPET VALVE FOR ACC	DANFOSS
12, 15	2	RV10-10	DIRECT RELIEF	EATON
16, 17	2	SBV1-10-C	2/2 BI-POPPET NC	EATON
18	1	NV1-8	NEEDLE VALVE	EATON
19	1	N/A	RESERVOIR	N/A
20	1	N/A	FILTER	N/A
21	1	ALPA2A-D-13 + ALPP2-D-6-FA	TANDEM PUMP (.567 AND .25 CIR)	MARZOCCHI
22	1	CV3-10	CHECK VALVE SIZE 10 (EXTERIOR)	EATON
23	1	SNM2NN	GEAR MOTOR (.513 CIR)	DANFOSS

Table 1: Circuit Component list

Direct Drive

The main operating mode on our vehicle is direct drive. In this mode, the rider pedals the vehicle the same as when riding a standard bike. The rider is directly turning the tandem pump via chain and the pump draws fluid from the reservoir and sends it straight to the motors. The pumps create negative pressure on the intake side which draws the fluid in. The other end of the pump creates positive pressure which drives the fluid through the system. The motors receive the fluid and rotate. This power is translated to the back wheel via chain and propels the vehicle forward. In this mode, maximum torque is needed at the motors to get the vehicle rolling, so we start with only the small pump actively in the system to create the greatest pressure. Both motors are active initially to produce the greatest torque to the wheel as there is high pressure and maximum surface area. At 1.5 mph, the small pump is taken out of the circuit and the large pump is put in to increase flow rate. At this same limit, the larger of the two motors is also removed from the circuit. Greater flow from the pump and a smaller motor allows for increased motor and wheel rpm which accelerates the vehicle. At 5.5 mph, the small pump is put back into the system to increase the flow rate further. With both pumps actively working and only our small motor, the vehicle can accelerate even further to a maximum velocity of 12mph for this drive mode. The velocity limits for pump and motor switching were acquired through copious testing to find the maximum vehicle speed from each pump and motor combination.

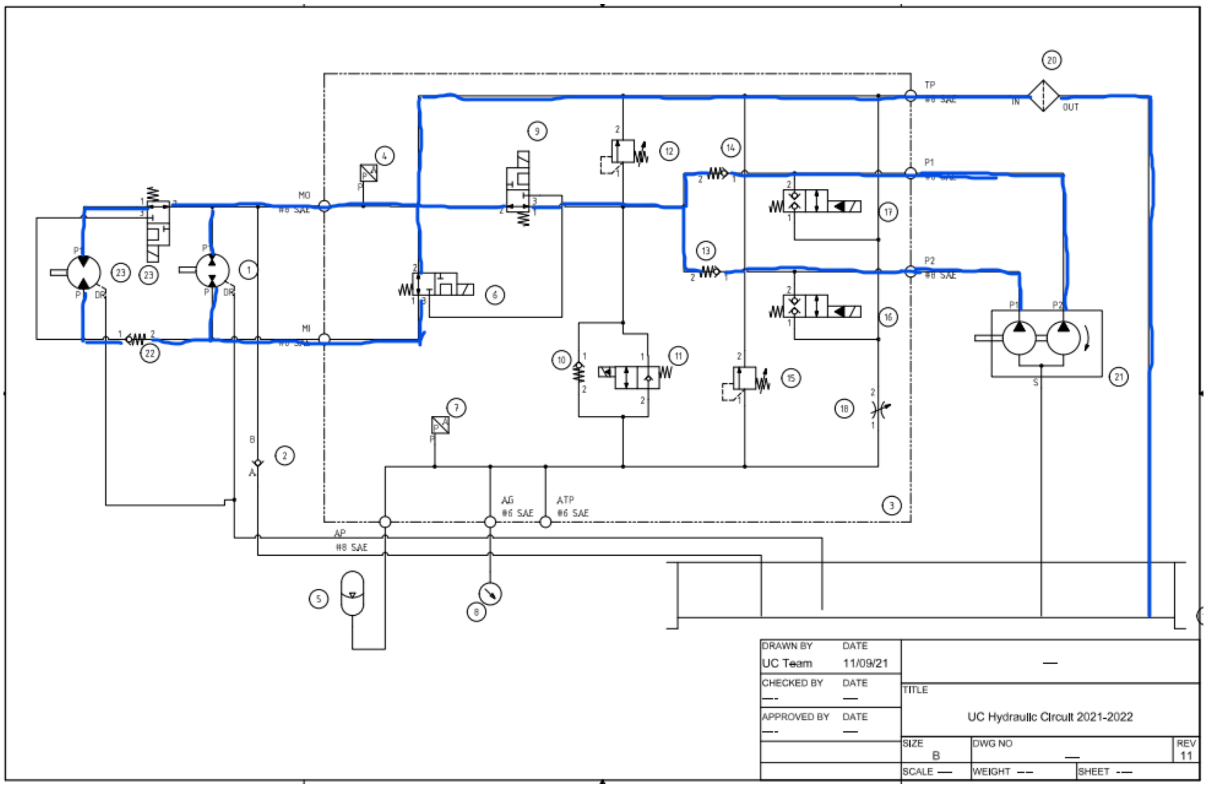

Figure 2: Direct Drive Flow Path (Starts at item 21)

Accumulator Charge

One particularly important component in our hydraulic circuit is the accumulator. This is component 5 in Figure 3 and table 1 below. It is a 1-gallon bladder accumulator made of carbon fiber. The accumulator is used to store energy which can be released and sent to the motors to propel the vehicle without the need for pedaling. In this mode, fluid is routed directly from the pumps to the accumulator. To store energy in the accumulator, the bike is propped up, so the back wheel is off the ground and a rider turns the pedals just like in direct drive. But in this case, the fluid is routed directly from the pumps to the accumulator instead of from the pumps to the motors. The accumulator is rated for 3000psi however our system is designed for a maximum pressure of 2500psi for safety reasons. This energy is stored until the rider releases the fluid using a normally closed button mounted on the handlebars of the bike. Like direct drive, we implemented limits via programming to switch which pump is actively working throughout the charge process. The order in which pumps are switched is opposite of direct drive. We begin with both pumps working to achieve the greatest flow rate and fill the accumulator faster. As fluid builds in the accumulator the pressure in the accumulator increases making it harder and harder to drive fluid in. Once we hit 900psi, the small pump is removed from the circuit. This decreases the flow rate but increases the pressure input as only the large pump is working. This pressure increase helps to overcome the back pressure from the accumulator and drive more fluid in. At 1000psi, the large pump is replaced by the small pump which provides the greatest pressure input to the system. The small pump is used the rest of the way to 2500psi. Changing pumps and therefor pressure, decreases the amount of force the rider must exert on the pedals making it easier to charge the accumulator especially at the top end of the charge.

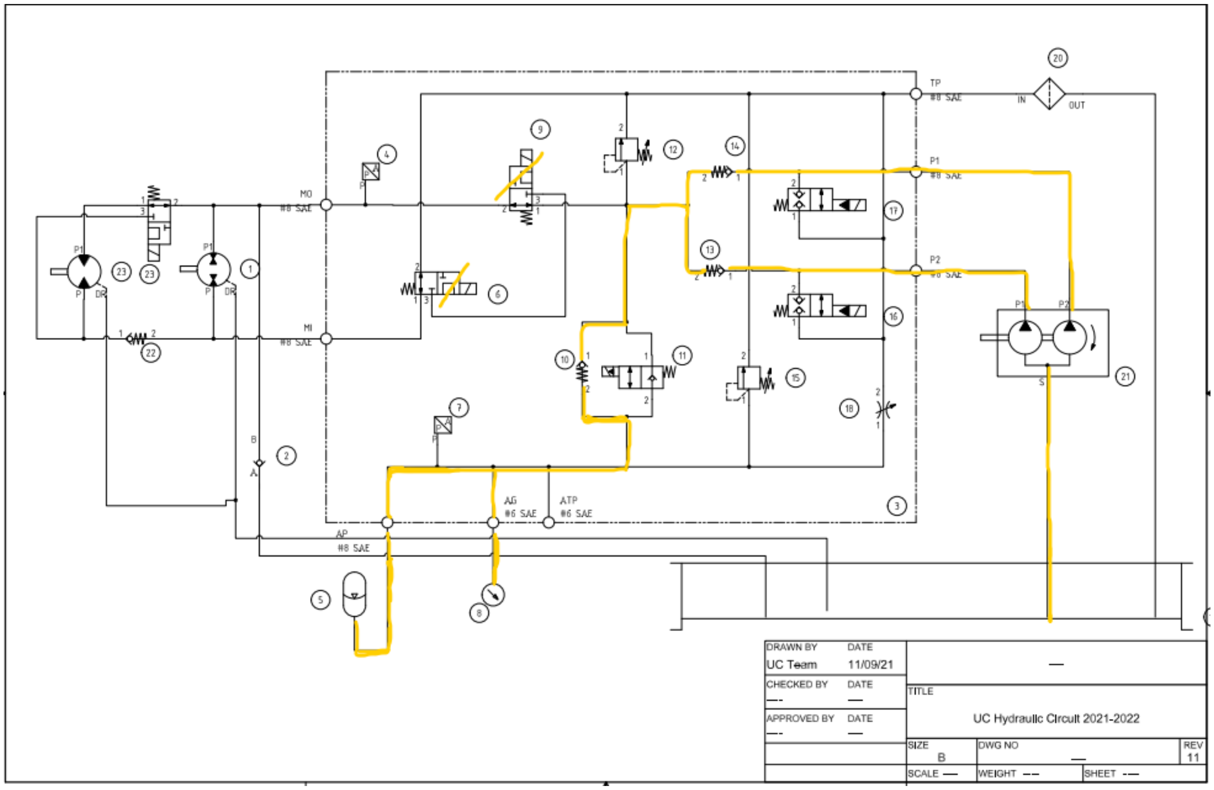

Figure 3: Accumulator Charge Flow Path (Starts at item 21)

Accumulator drive

As mentioned above, stored energy can be released from the accumulator to propel the vehicle without the need for rider input. Once the accumulator has been charged, the rider switches into the accumulator charge mode. The fluid is released with a push button that is directly connected to valve 11 as shown in Figure 4. In previous year's designs, the fluid would be released all at once to the motors which equates in the greatest speed possible. However, this drive mode is used mainly for the efficiency challenge in which distance from a single charge is the main objective. To increase the distance traveled the rider will use a concept we created called accumulator pulse. The rider can press the button and release a burst of pressure from the accumulator, but upon release of the button the accumulator is closed again. The vehicle will coast for a distance proportional to the rider's momentum due to the burst of energy. As the vehicle slows down, the rider can "pulse" the accumulator again and then coasts again. Doing this until all fluid from the accumulator has been used increases the distance traveled from one charge resulting in greater efficiency.

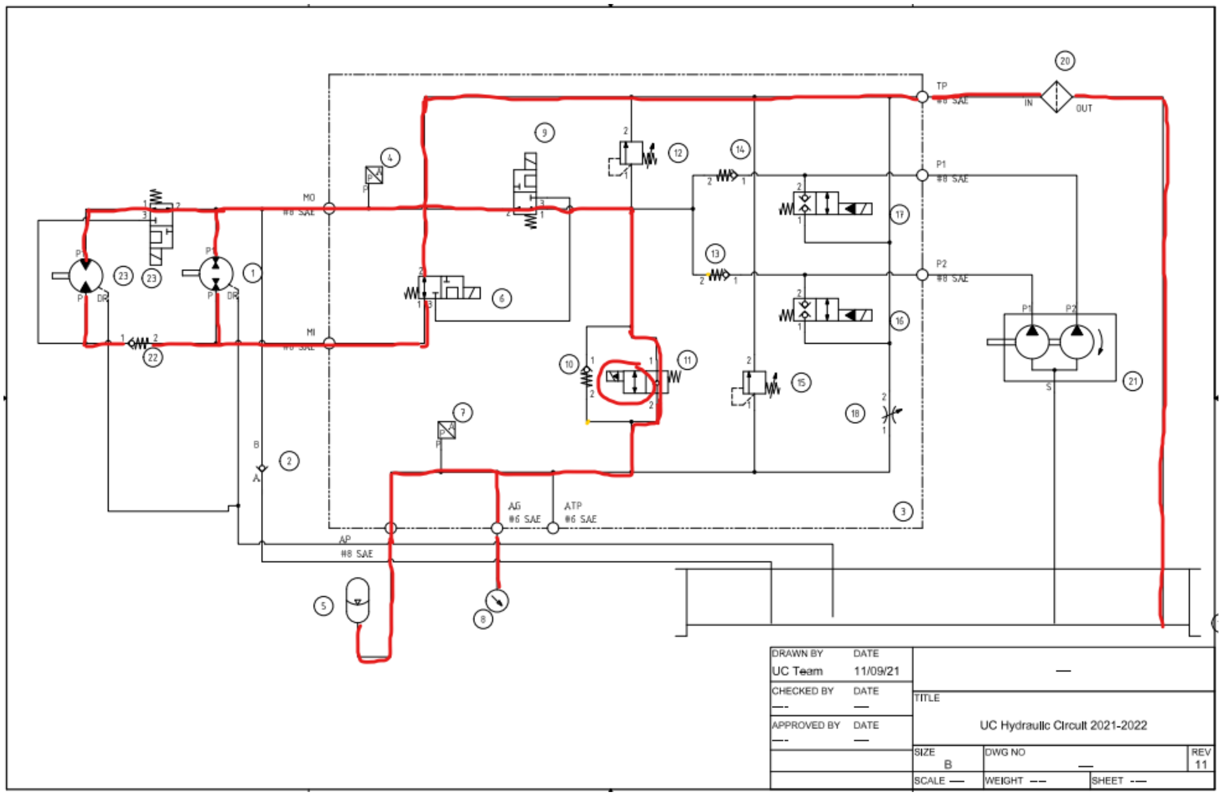

Figure 4: Accumulator Drive Flow Path (Starts at item 5)

Regenerative Braking

The last drive mode available on our vehicle uses a concept called regenerative braking. This concept is also used in modern day electric vehicles. The objective is to regain some energy in the system while the vehicle slows down to a stop. To use this concept, the rider will get the vehicle up to its maximum speed using either direct drive or accumulator charge to build up momentum. At the vehicle's top speed, the rider will switch into the regenerative braking mode which effectively turns the motor into a pump. Because the vehicle has momentum, the motor will continue to rotate with the wheel and draw fluid from the reservoir using a designated flow path. Because the motor is now acting as a pump, it will route the fluid back to the accumulator. As this is happening, the vehicle will slow down and eventually come to a stop hence the name "regenerative braking". We only use our small motor in this drive mode to create high pressure and drive as much fluid into the accumulator as possible. This concept is a requirement for the fluid power vehicle challenge; however, it greatly improves the performance of the vehicle as well.

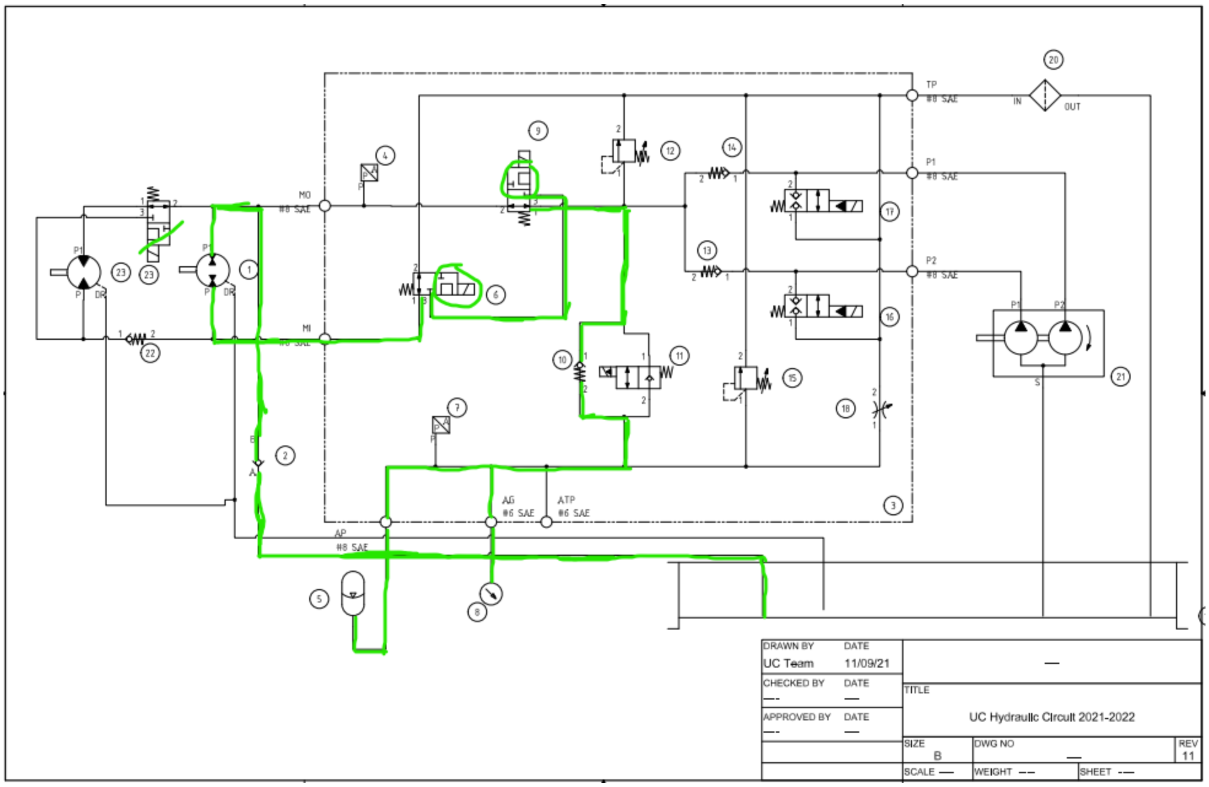

Figure 5: Regenerative Braking Flow Path (Starts at item 19)

Hydraulic Manifold

Another key feature of our hydraulic circuit is the use of a custom hydraulic manifold. A hydraulic manifold is a block of either steel or aluminum with various machined channels to route fluid as well as machined cavities where cartridge valves are inserted. The manifold is the heart of the vehicle. Just as the human heart routes blood to different areas of the body, the hydraulic manifold is home to most of the valves in the system which route the fluid to different areas of the bike dependent on the drive mode selected by the rider. This year, we partnered with VEST Inc, SunSource, and Source Fluid Power to design our own manifold specifically for our unique circuit and vehicle design. We, the students, utilized cutting edge technology provided by VEST Inc to place our valves, ports, and channels where we felt was optimal. The design was then given to Source Fluid Power where it was fabricated and sent back to us. Internal, machined flow paths can be seen in Figure 7 and an external view of all valves and sensors is shown in Figure 8. Hydraulic manifolds allow for a cleaner, more compact circuit which saves space and weight on the vehicle. Manifolds make troubleshooting and maintenance easier as all valves are in one central area as well as reduce the number of opportunities for active leaks in the circuit. Custom designing our own manifold allowed us to strategically place ports and valves to decrease external tubing complexity.

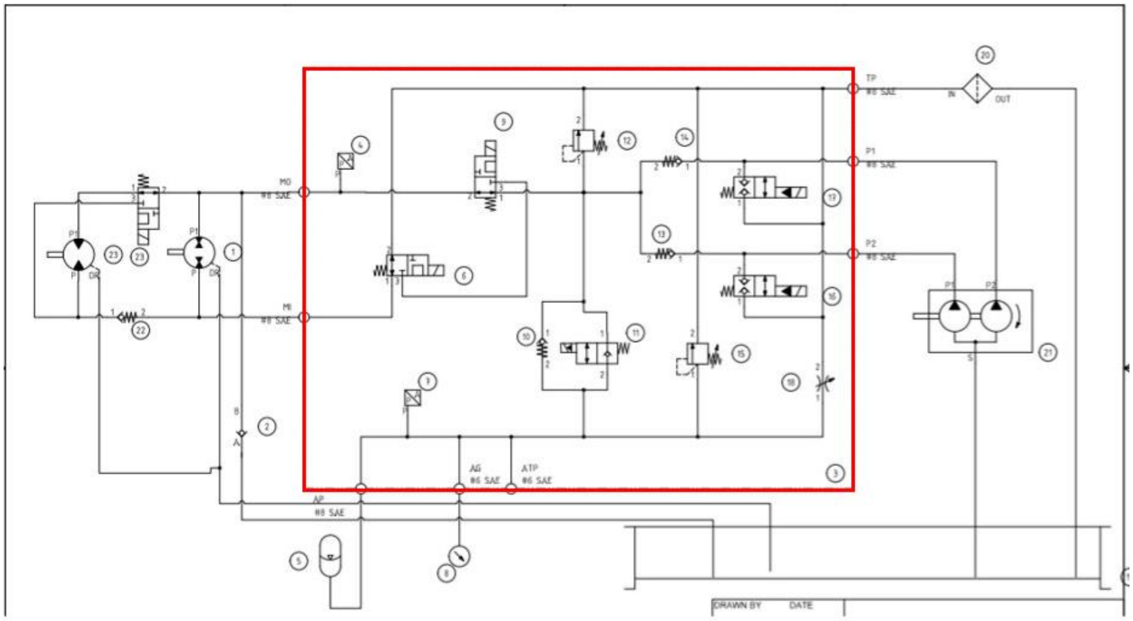

Figure 6: Circuit with Manifold Components Outlined in Red

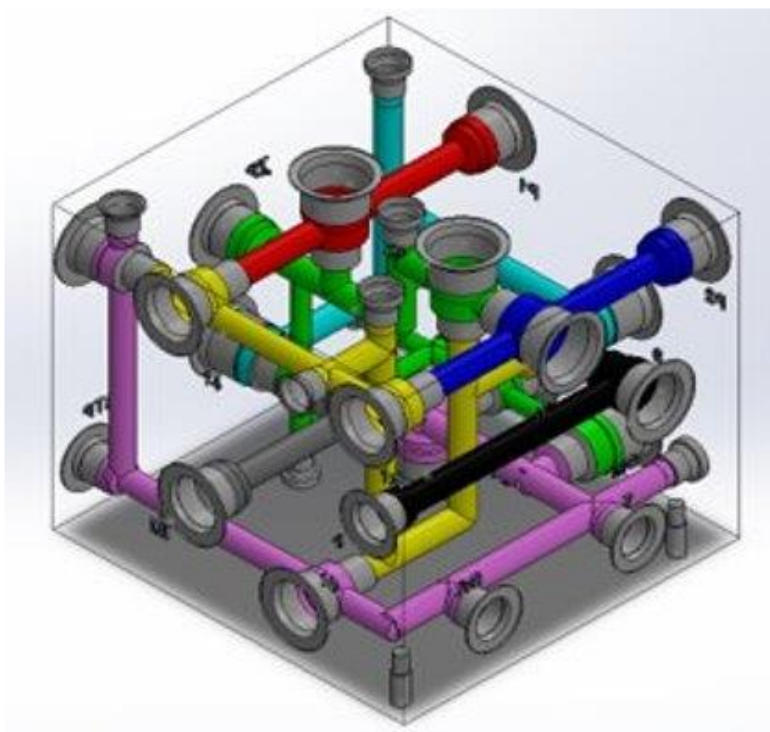

Figure 7: Internal Manifold Machining

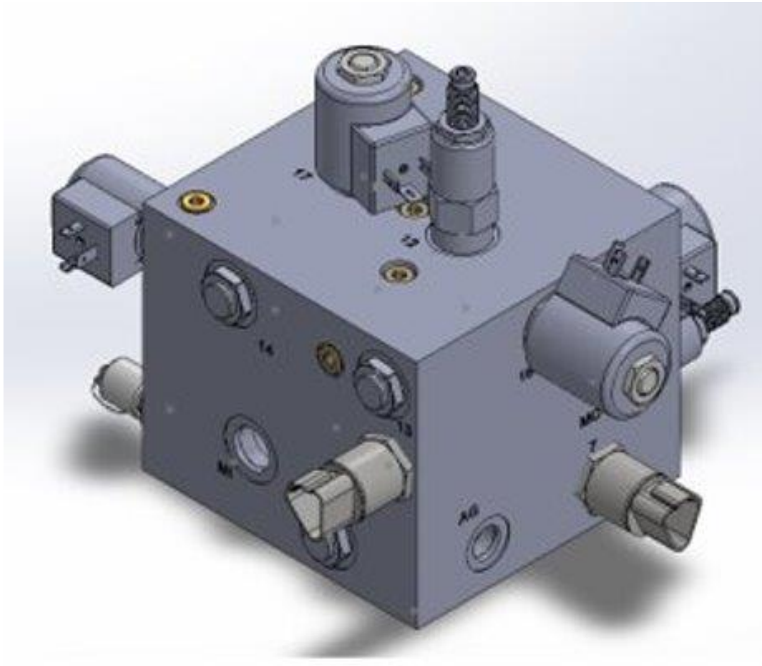

Figure 8: External Manifold Components

Pneumatic System

The goal of our pneumatic system is to design and create a pneumatic scissor jack. The purpose of this scissor jack is to lift the bike up and stabilize it for storing and testing purposes. The components we used to build this circuit include a pneumatic receiver, regulator, 5-way 3-position valve, and finally a pneumatic piston. The pneumatic circuit showing how each of these components are configured can be seen in Figure 9. Once we did our research on which components, we ordered these from Norgren and McMaster-Carr. We also bought a scissor jack from Lowes. Calculations were made to find the optimal positioning within the scissor jack to get the most displacement with the piston while getting the piston to lift with optimal enough force to lift the bike and any additional weight that may have been added afterwards. These calculations can be seen in Figure 11. Bike Schmeat. Once the parts were received, fabrication began on the scissor jack to allow the pneumatic piston to fit within the side of the jack. The rest of the components were assembled onto the bike. Each component was attached together with red nylon pneumatic hosing that was rated to 190 psi. This was much more than needed. The final assembly can be seen in Figure 10.

The overall design worked out well. Once the receiver was pressurized with air the regulator would allow 60 pounds of pressure to pass through to the 5-way 3-position valve. This valve would hold the pressure till the user manually pushes the lever down would then fill the side of the piston with the pressurized air, thus shortening the stroke of the piston and lifting the bike. The closing of the piston raises the scissor jack, while extending the piston lowers the scissor jack assembly, thus raising or lowering the back wheel if needed.

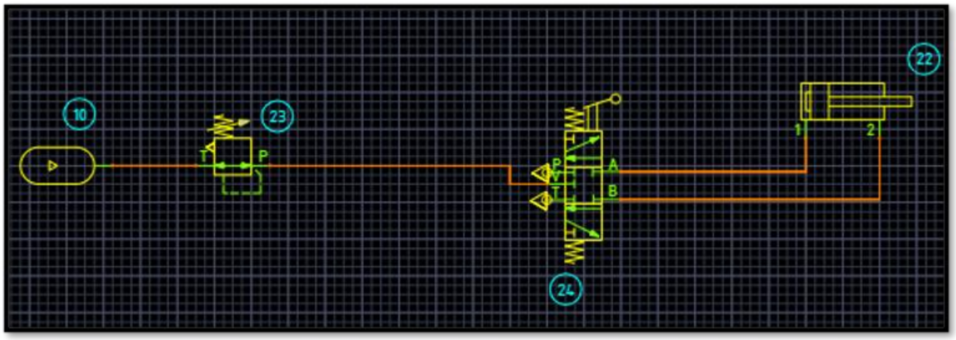

Figure 9: Pneumatic Circuit

Figure 10: Physical Pneumatic Assembly

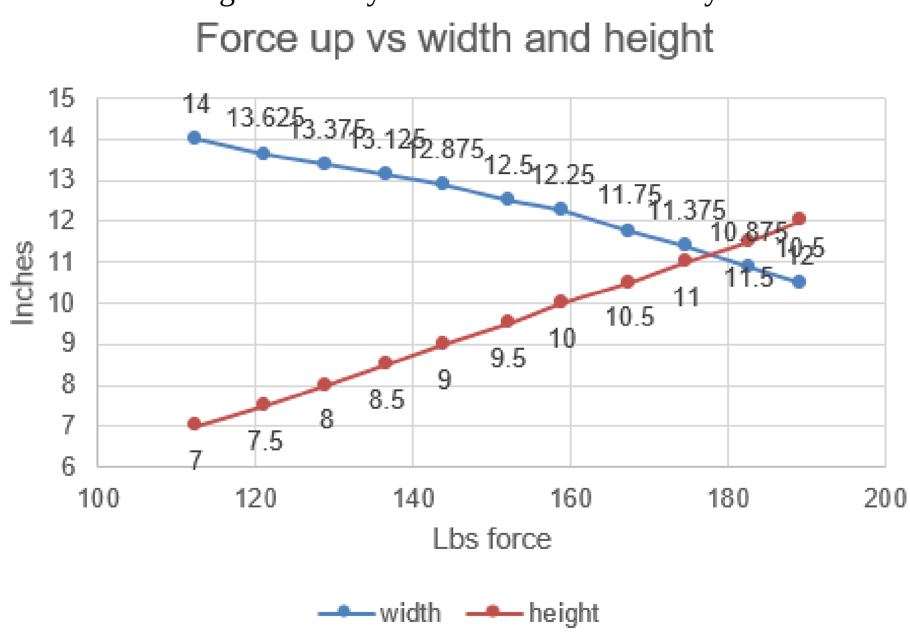

Figure 11: Force Up vs Width & Height

Electronic System

For the electronics this year, the team wanted to create a more complex system that would adapt based on the current state of the bike. The team purchased the first controls package option offered by Iowa Fluid Power this year that included a touchscreen display and a programmable logic controller. The touchscreen display was designed by the team in JMobile to show useful information to the rider for data collection purposes as well as optimize the rider's experience. The screen displayed pressure at the motors, pressure at the accumulator, velocity of the bike, input and output horsepower, active pumps in the system, active motors in the system, and buttons to change between drive modes.

For the PLC, a code was created by the team in the CoDeSys software that had the core functions of the bike. With this code, the team programmed the function of each of the buttons on the touchscreen, sent information to the touchscreen to display, and achieved this by sending electrical signals. The code was used to create an “automatic transmission” of sorts by activating different pumps and motors to optimize the rider’s experience. In direct drive, the smallest pump is active with both motors, then once the bike reaches 1.5 MPH, the small pump is kicked out and the large pump is active with the small motor only. Once the bike reaches 5.5 MPH, the small and large pumps are active with the small motor. This is done by tracking the speed with an inductive sensor on the motor gear, counting teeth as they pass and using a conversion with wheel circumference to calculate the MPH. For accumulator charge, pumps are changed off pressure readings from the accumulator pressure transducer. To start, both pumps are active, then at 900 PSI, the small pump is kicked out and the large pump is active. Then at 1000 PSI, the large pump is kicked out and the small pump is active. Doing this makes pedaling fluid into the accumulator much easier for the rider and allows the team to charge the accumulator to 3000 PSI in about three minutes and thirty seconds. In accumulator drive, both motors are active from 3000 PSI until 2200 PSI, then the large motor is kicked out and the small motor is active to utilize the flow more efficiently and increase speed. Accumulator drive pressure limits are also tracked with the accumulator pressure transducer. For output horsepower, the inductive sensor on the motor gear calculates its speed and the motor pressure transducer finds the pressure, then both are used to calculate the horsepower and it is displayed on the screen. For the input horsepower, the inductive sensor on the pedal gear calculates its speed and the pressure transducer at the motor calculates the pressure, then both were used to calculate the input horsepower of the rider.

As stated previously, the sensors used along with the code (seen in the appendix) create an automatic transmission for the bike, allowing for a smoother ride by adapting the hydraulic components to the current state of the bike. These controls were purchased with the intent of reprogramming them each year for the use of teams to follow this one.

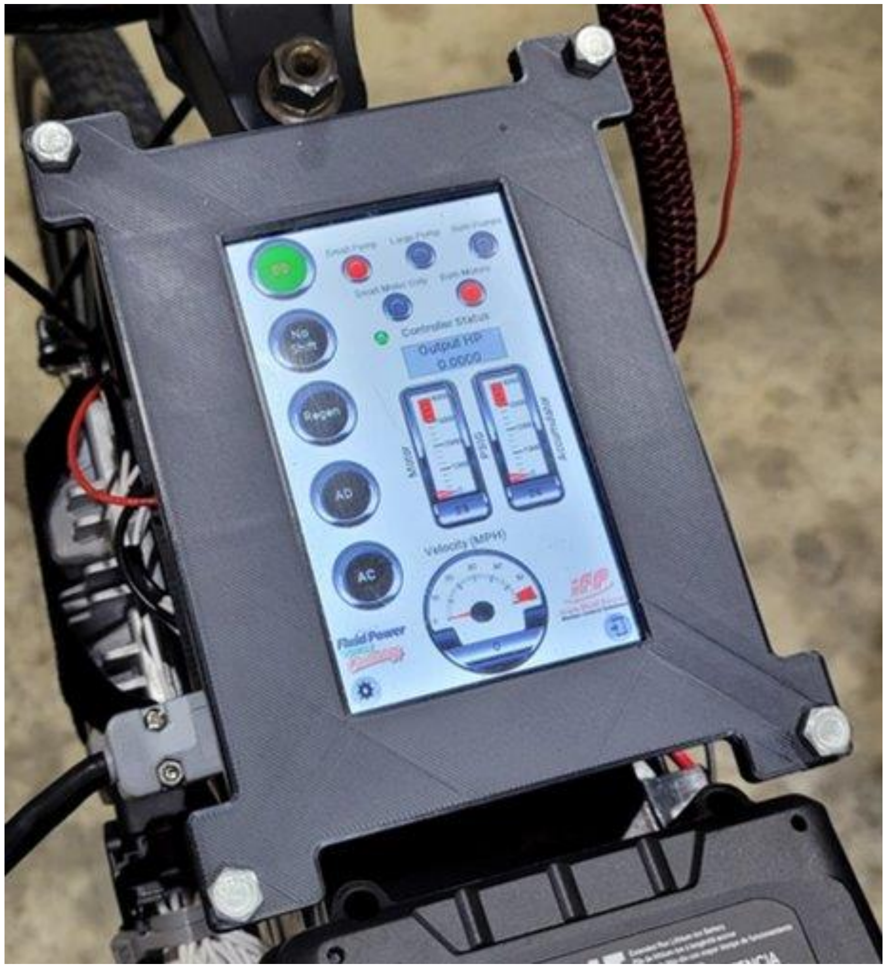

Figure A: Touchscreen Display

Frame and Fabrication Design

For the frame itself, we wanted to create a custom frame by using parts from a donor bike that we received. We specifically looked for a donor bike with 24” wheels and an American bottom bracket which will allow more space to weld since it is designed to be larger. We designed the frame, brackets, reservoir, and back dropouts using SolidWorks. The final design can be seen in Figure 13. Then we used Ansys Mechanical R2 to test the deformation and stress analysis of the frame which showed that it would be more than sufficient for our application. This can be seen in Figure 14. We also made sure that the frame was structurally sound by adding a safety factor of 10 to the structural analysis (Advised by our mentor). We constructed the frame out of a513 carbon steel hollow tubing with 1/8”, ¼", and ½" plate for the reservoir and brackets. We were able to use Hirschberg Steel for material and Iron Belle Metal Design for fabrication. Some things that we wanted to improve on last year was to make the reservoir apart of the frame. The reason behind this is because it would allow more space to put other components, save weight, and will allow easy port access. We also designed the dropouts where the back wheel is to have a slotted hole instead of a normal hole. This would allow the bike tire to be moved up a couple inches or back a couple if needed. Finally, we kept with the design from last year to keep a back plate for additional component placement. The final fabricated frame can be seen in Figure 15.

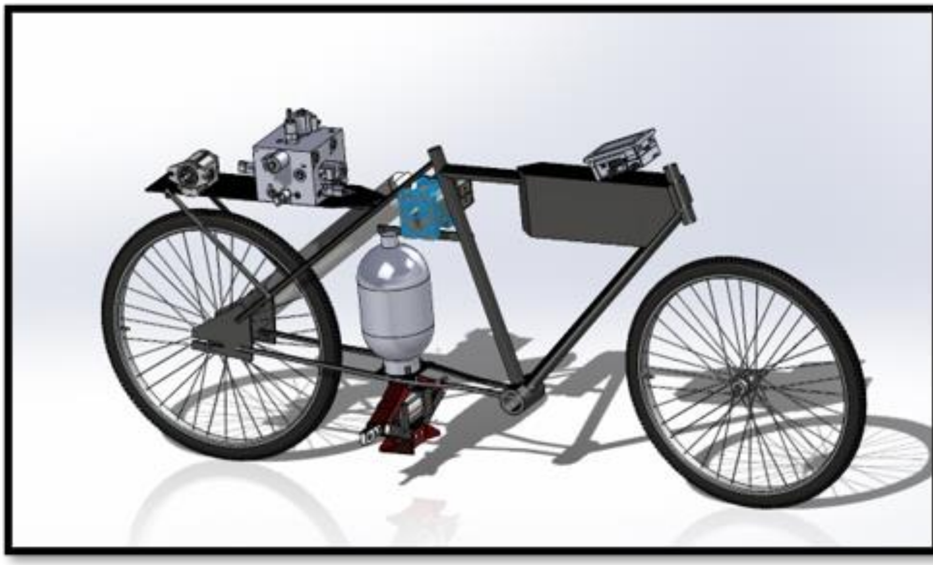

Figure 13: Final Frame Design in Solidworks 3D

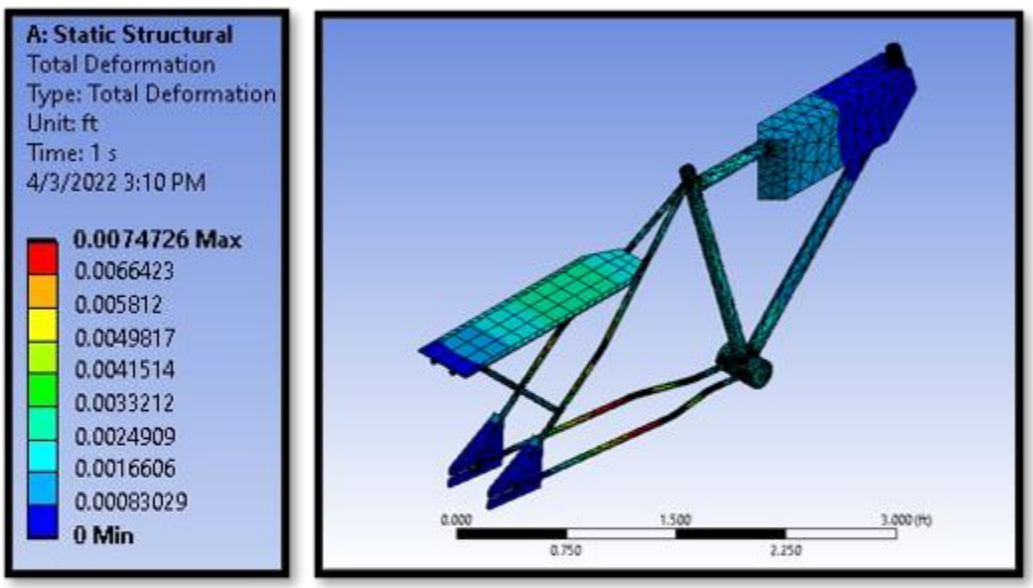

Figure 14: Total Deformation Analysis with Ansys Mechanical R2

Figure 15: Final Fabricated Frame Design

Gearing Design

Our gearing design consists of many sprockets. Starting off we have a sprocket at the pedals that has 40 teeth which then goes up to our tandem pump sprocket with 10 teeth. This makes our gear ratio 4:1. We wanted to make sure we could get enough flow out of the pumps and to have a greater gear ratio would make the pump revs higher. This can be seen in Figure 16. Next going from the back wheel sprocket, we have an 18 teeth sprocket which then goes to 2 other motor sprockets with 18 teeth. These Motor are offset to make sure there is chain engagement. An idler gear sprocket was also added to the system later to help aid in sprocket engagement. This can be seen in Figures 17 and 18. We used a standard ANSI roller chain with a ½ inch pitch constructed of 41 steel to make sure that it would handle any torsional forces and would comply with the gears that we purchased for the pump and motors.


Figure 16: Peddle Sprocket going up to Tandem Sprocket


Figure 17: Back Wheel Sprocket

Figure 18: Motor Sprocket

DESIGN ALTERNATIVES AND SELECTION

Most teams at the FPVC in Ames, Iowa built three wheeled bikes such as a tricycle or a recumbent bike. The purpose for this specific build is because it adds more stability to the bike, and it can get lower to the ground making the center of gravity better. There are some drawbacks to using one of these types of bikes though. Some drawbacks include more friction loss, more weight, and a larger turning radius compared to a two-wheel bike. Therefore, our team decided to go with a standard two-wheel bike instead of a three-wheeled bike. Our design was way lighter than the competitors, only being about 187lbs. Our turning radius was much smaller which helped on the endurance races. Finally, even though our two-wheel design created opportunities for balancing issues, we were able to strategically place components in a way that stabilized the bike.

ENGINEERING CALCULATIONS

List of equations

Grade to Radians:

$$Radians = Tan^{-1}\left(\frac{Grade}{100}\right)$$

Pull of Bike (Pull1):

$$Pull\,1 = Sin(Radians) \cdot Weight$$

Pull from Weight (Pull2):

$$Pull\,2 = \cos(Radians) \cdot Weight \cdot Rolling\ Resis\tan c\,e$$

Total Uphill Pull:

$$Total\ Pull = Pull\,1 + Pull\,2$$

Torque Required:

$$\tau = Uphill\ Pull \cdot r$$

Wheel RPM:

$$RPM = \frac{(336 \cdot Desired\ MPH)}{d}$$

Motor CIR^M Calculated:

$$CIR = \frac{(\tau \cdot 2\pi)}{CIR\ Chosen}$$

Adjusted PSI:

$$PSI\ Adjusted = \frac{(\tau \cdot 2\pi)}{CIR\ Chosen}$$

Motor CIR^M Calculated:

$$CIR\ Available = \frac{CIR}{Motor\ Effciency}$$

Gallons/Minute of Flow:

$$GPM = \frac{(CIR \cdot REq'd \cdot RPM)}{231}$$

Motor HP Req'd:

$$HP = \frac{(GPM \cdot PSI\ Adujested)}{1714}$$

Motor HP Req'd (Check):

$$HP = \frac{(\tau \cdot RPM)}{63025}$$

Motor Efficiency:

$$Effciency = \frac{(HP\ Check)}{Motor\ Effciency}$$

Pump RPM:

$$Pump\ RPM = Pedaling\ RPM \cdot Gear\ Ratio$$

Pump CIR^P Req'd:

$$CIR = \frac{(GPM \cdot 231)}{RPM}$$

Pump CIR Available:

$$CIR\ Available = \frac{(CIR\ Chosen)}{Pump\ Efficiency}$$

Loading Conditions – required yield and tensile strength of material

After the main frame design was completed in Solidworks 3D, it was imported into a static structural project file in Ansys Mechanical R2. Once imported, we assigned the material of the frame to be a513 carbon steel. The tensile strength of this steel is 87,000 psi and the required yield is 72,000 psi. After assigning the material, a mesh was created to position the nodes around the main body of the frame. This can be seen in Figure 19. Then, loads were placed onto the bike frame. These loads consisted of one pushing down on the head tube, one pushing down on the seat tube, one pushing on the insides of the bottom bracket, on the bike plate, and finally loads on the back dropouts. After the loads were set in the correct positions with their correct magnitudes, resulting fixed supports were added below the head tube and the dropouts. Next, we made the software want to check for total deformation, total stress, and total strain. The results of these tests can be seen in Figures 19, 20, and 21. These results are with a safety factor of 10 added to the system. Overall, the results prove our design is sound.

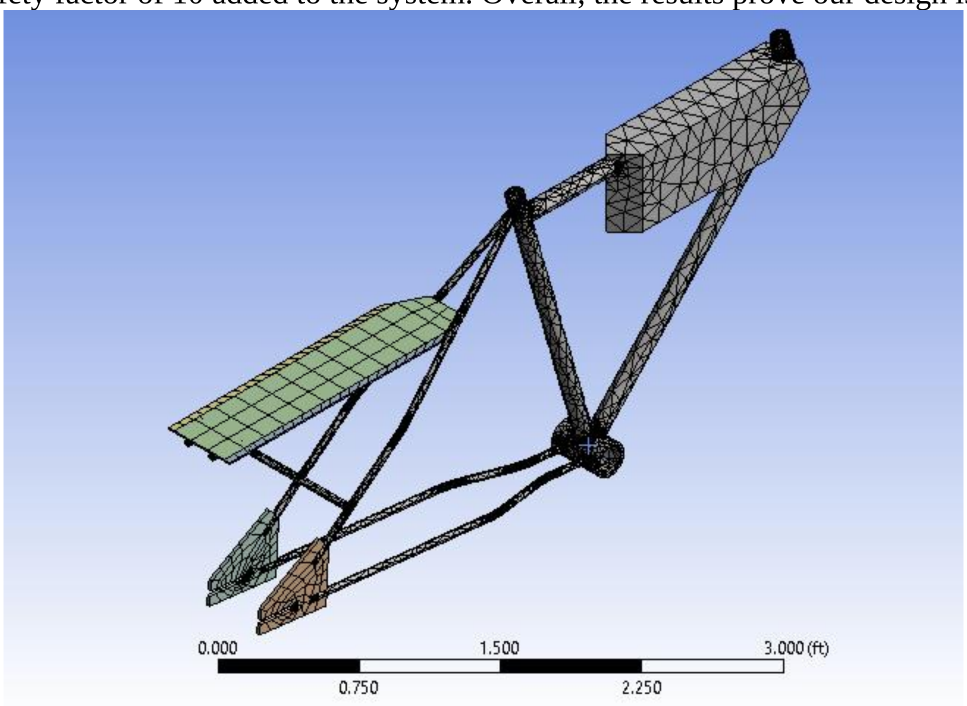

Figure 19: Frame Mesh

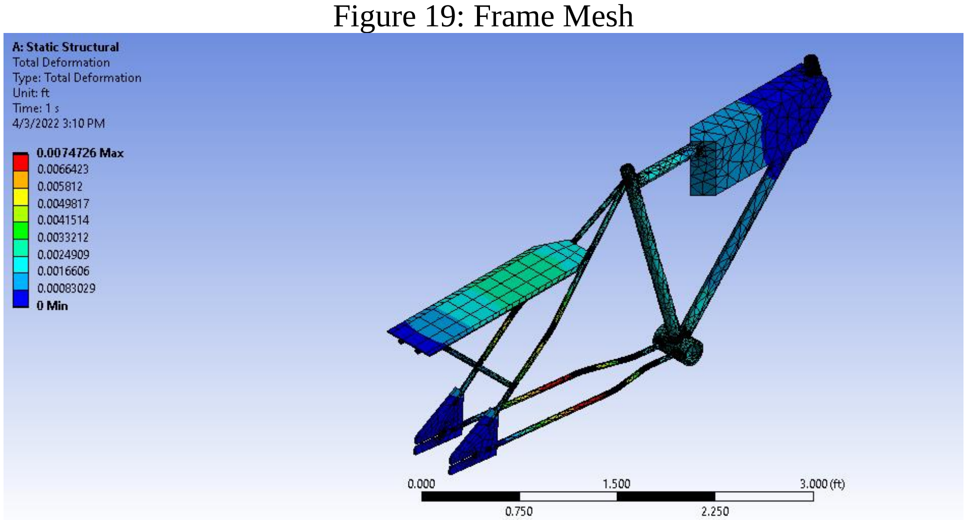

Figure 20: Total Deformation

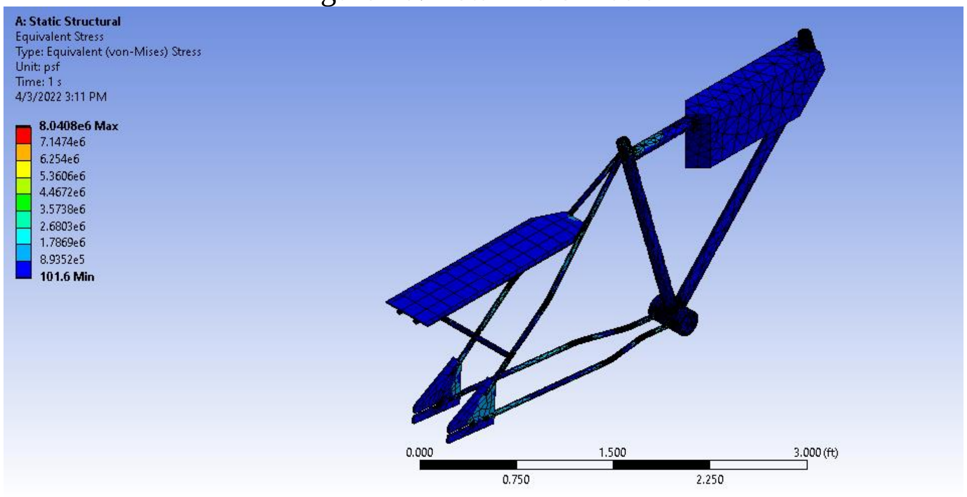

Figure 21: Frame Stress

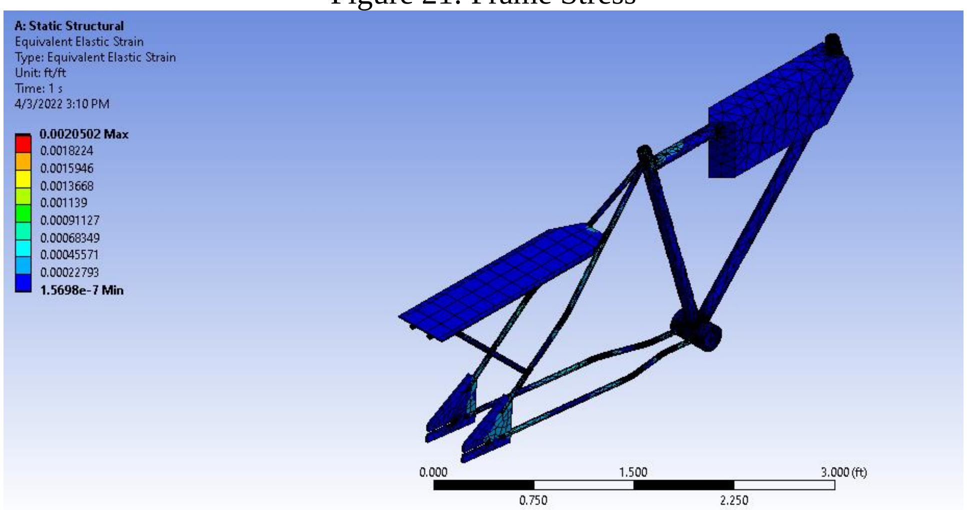

Figure 22: Frame Strain

Material Selection

The material we decided to use for our bike frame was 513 carbon steel. We wanted to have strength within our frame, so we chose steel over aluminum. Steel is heavier than aluminum, but we made sure when making our BOM we went with lighter pieces that either had thin thicknesses or thin pipe walls. Another reason why we went with steel is because it is easier to weld to. Figure 23 shows our fabricated frame and the steel that we used for it.

Figure 23: Material Use for Frame Fabrication

Factors of Safety

A safety factor of 10 was chosen for the frame to avoid failure in the event of unexpected loads on the seat of the bike. We used a safety factor of 10 because while talking to our mentor Dan Turner, he advised us whenever there is a human involved with designed machinery, a safety factor of 10 is necessary. Other safety features on our vehicle include 3D printed chain/gear covers, multiple pressure relief valves, an accumulator discharge valve, and an easily accessible electrical kill switch. Furthermore, automatic valve activation allows the rider to keep their hands on the handlebars and their eyes on the road. The display was strategically placed in front of the rider and set at a 15-degree angle for easy verification of system properties. Lastly, we designed a 5th mode called "no shift" which essentially acts as a neutral position in the event of an unexpected loss of power.

MANUFACTURING DRAWINGS

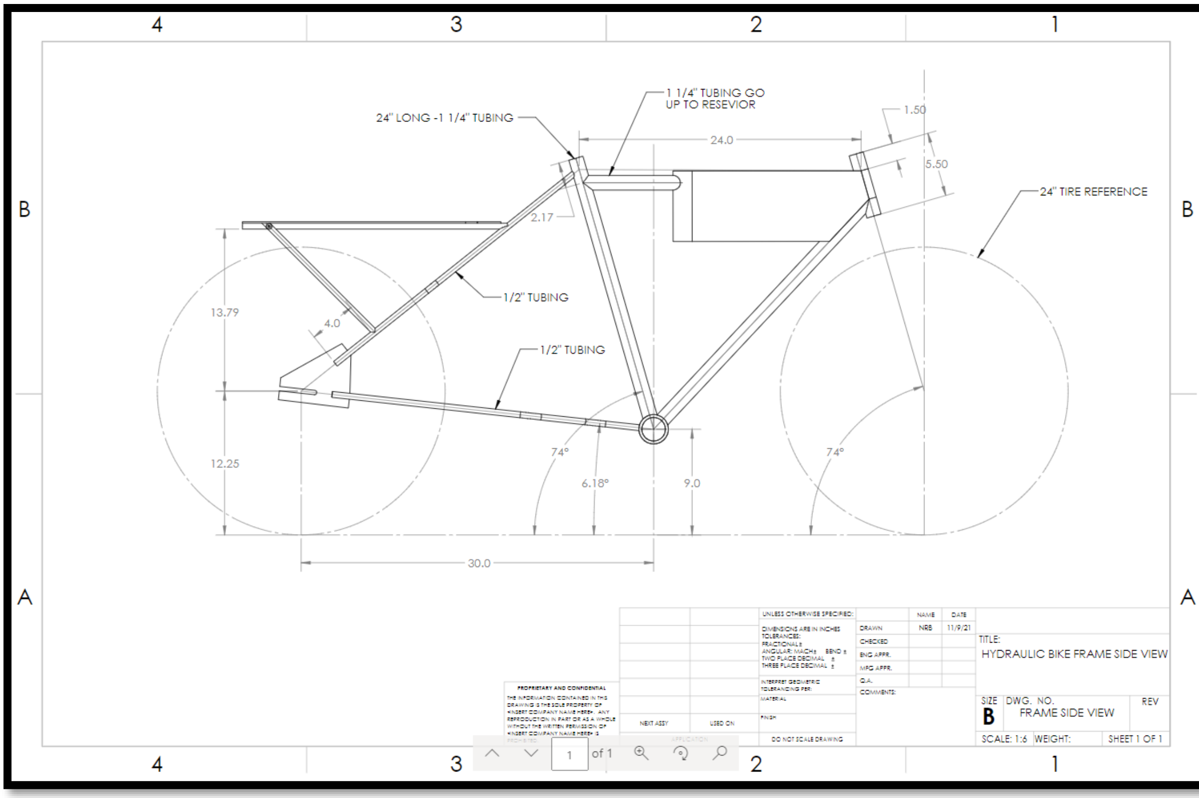

Figure 24: Final Frame Drawing

BILL OF MATERIAL

Item	Quantity	Value	Extended Value	Supplier
1.25" Od x 0.12 wall tube x 20'	1	$ 90.00	$ 90.00	David Hirschberg Steel
0.5" OD x 0.065 wall tube x 20'	1	$ 32.00	$ 32.00	
0.25" plate 4' x 4'	1	$ 260.00	$ 260.00	
1/8" sheet 4' x 8'	1	$ 220.00	$ 220.00	
0.5" plate 1' x 1'	1	$ 50.00	$ 50.00	
Weld in port #8	4	$ 12.63	$ 50.52	Turner Hydraulics
Weld in port #12	5	$ 11.08	$ 55.40	

Table 2: Frame BOM

Item	Quantity	Value	Extended Value	Supplier
Check Valve, size 10	1	$ 14.00	$ 14.00	Sunsource
Check Valve, size 8	2	$ 15.00	$ 30.00	
Coil, 24VDC DIN, J type	2	$ 26.00	$ 52.00	
Electrical Connector, DIN, 12-24VDC	5	$ 8.00	$ 40.00	
Flow control, needle valve	1	$ 26.00	$ 26.00	
Line body, VC10-2, Aluminum SAE -8	1	$ 23.00	$ 23.00	
Line body, VC10-3, Aluminum SAE -6	1	$ 63.00	$ 63.00	
Proportional, throttle, NC	1	$ 163.00	$ 163.00	
Relief, direct acting	1	$ 41.00	$ 41.00	
Solenoid, 2 pos, 2 way, bipoppit, NC	2	$ 65.00	$ 130.00	
Solenoid, 2 pos, 3 way spool 1-2/1-3	1	$ 45.00	$ 45.00	
Accumulator Test point, SAE -6 Male	1	$ 15.00	$ 15.00	
Motor, aluminum 1/2" shaft, 5.2c^3	1	$ 167.65	$ 167.65	IFP
3 way, 2 pos valve, NC	1	$ 77.16	$ 77.16	
Line body, 90 deg, SAE 8 ports T-11A	1	$ 29.24	$ 29.24	
24 VDC coil with connector	1	$ 23.81	$ 23.81	
Tandem pump	1	$ 512.00	$ 512.00	Turner Hydraulics

Table 3: Hydraulic Circuit BOM

Item	Quantity	Value	Extended Value	Supplier
5" TFT Touchscreen 800x480	1	$1,094.28	$ 1,094.28	IFP
Jmobile programming software	1	$ 295.75	$ 295.75	
Plug-in module for CAN fieldbus	1	$ 194.35	$ 194.35	
Mobile PLC Controller	1	$ 372.54	$ 372.54	
Wiring harness	1	$ 62.52	$ 62.52	
PCAN USB adapter	1	$ 225.00	$ 225.00	
DB9 Serial Y adapater	1	$ 10.39	$ 10.39	
Pressure transducer, 3000 psi, SAE 6	2	$ 70.30	$ 140.60	
Cable Deutsch connector, 3 wire	2	$ 19.88	$ 39.76	
Inductive sensor, 8mm dist, 3 wire	2	$ 48.68	$ 97.36	
Batteries and charger (24V)	1	$ 139.00	$ 139.00	Misc
Battery Adapter	1	$ 19.00	$ 19.00	

Table 4: Electronics BOM

Item	Quantity	Value	Extended Value	Supplier
Non-repairable reservior	1	-	-	Norgren
2" storke cylinder	1	-	-	
T connector (pneumatics)	1	$ 40.55	$ 40.55	McMaster
Air valve NPT	1	$ 3.65	$ 3.65	
Reducer 3/8" Male	1	$ 4.12	$ 4.12	
Ball valve 1/4" NPT Female	1	$ 13.37	$ 13.37	
Muffler 1/4 NPT	1	$ 3.79	$ 3.79	
Muffler 3/8 BSPP Male	3	$ 6.03	$ 18.09	
Directional valve, 5 way, 3 pos	1	$ 206.91	$ 206.91	
Brushing adapter	6	$ 3.33	$ 19.98	
Adapter 5/16" tube	10	$ 2.05	$ 20.50	
Pneumatic tubing	3	$ 5.64	$ 16.92	
Pneumatic line fasteners	1	$ 10.09	$ 10.09	Misc
Scissor jack for pneumatics	1	$ 31.98	$ 31.98	

Table 5: Pneumatics BOM

Item	Quantity	Value	Extended Value	Supplier
10 tooth sprocket	1	$ 15.71	$ 15.71	McMaster
18 tooth sprocket	1	$ 25.04	$ 25.04	
Chain with 1/2" pitch	10	$ 4.64	$ 46.40	
Chain connecting link	2	$ 0.95	$ 1.90	
Chain adding connecting link	2	$ 2.15	$ 4.30	
Chain adding link	2	$ 0.69	$ 1.38	
Thread tape	1	$ 4.29	$ 4.29	
Locktite	1	$ 15.33	$ 15.33	
Electrical wiring	1	$ 44.10	$ 44.10	Misc
Electrical tools	1	$ 58.90	$ 58.90	
Paint	1	$ 35.37	$ 35.37	
Nuts and washers for wheel	1	$ 9.91	$ 9.91	
Bike frame to donate parts	1	$ 245.78	$ 245.78	
Tires and intertubes	1	$ 130.29	$ 130.29	
Cable ties and mounting tape	1	$ 10.75	$ 10.75	
3 piece crank replacement	1	$ 45.00	$ 45.00	
Shop towels	1	$ 10.11	$ 10.11	
3 piece crank and hand grips	1	$ 208.00	$ 208.00	

Table 6: Additional Components BOM

BUILD AND TEST

DISCUSSION OF THE MANUFACTURING PROCESSES UTILIZED

Our vehicle was manufactured in many ways. The frame was fabricated by Jordan Graff, owner of Iron Belle Metal Design in Cincinnati, OH. We designed the frame in its entirety and sent the drawings to Jordan for welding. The only components of the frame that were not custom designed were the front forks, handlebars, headset, and bottom bracket which we were able to salvage from a standard bicycle. We did, however, do a fair amount of welding ourselves after the frame was initially complete to add supports and brackets for certain components. Some components and/or supports were attached to the vehicle via fasteners such as nuts and bolts. The hydraulic manifold was manufactured by Source Fluid Power using a CNC machine. The hydraulic tubing was bent by hand and done completely by the team with oversight from Bob Sheaf, owner of CFC Industrial Training in Fairfield, OH. All valves were inserted by hand and by the team as well. All electronics were wired by the team as well.

TEST PROCEDURE AND CRITERIA

Vehicle testing started in early March 2022 once all components were received and assembled. Testing mainly consisted of verifying all drive modes for proper operation, ensuring all sensors were operational and calibrated, testing certain vehicle concepts and gathering vehicle performance data. To test direct drive operation, multiple team members took turns taking laps on the bike. The rider would check for vehicle comfort, changes on the display, and automatic valve activation. The rest of the team would look for active leaks in the circuit, changes in components indicating poor security, and tension on pump and motor chains. To test accumulator charge, multiple riders took turns filling the accumulator completely on their own. The rider would feel overall comfort and ease of pedaling throughout while ensuring that the proper pump(s) were always active. To test accumulator drive, multiple riders would ride the vehicle with energy stored in the accumulator. Again, the rider would ensure that the proper valves were switching as intended and monitor the pressure gauges to ensure consistent release. Regenerative braking testing included riding the vehicle in direct drive and switching into regen mode and verifying a safe stopping distance and the amount of pressure built up over that distance.

One vehicle concept we tested was keeping the accumulator open while in direct drive. The idea here was to allow any excess pressure to build in the accumulator, then once there was a drop in pressure, likely from the rider not pedaling, the pressure in the accumulator would be released and fill in the gaps. We verified that this concept did assist in a smoother ride during direct drive. Another concept that required lots of testing was finding the optimal accumulator pre-charge. The accumulator is filled partially with nitrogen gas which allows the fluid to be released. This value affects how soon and easily the fluid will begin to fill the accumulator. We tested multiple different levels ranging from 1200psi to 500psi and settled on 850psi which yields the optimal performance for all drive modes.

This year, we had the time and capability to capture useful vehicle performance data such as velocity, input and output HP, gear rpm, and pressure in multiple locations in our system. Using these outputs, we can calculate other useful values such as output torque and acceleration. We wanted to ensure that teams succeeding ours would have useful data other than just the race results to compare with and ensure their design is improving year over year.

TEST RESULTS AND FINDINGS

Direct Drive Data					
Rider	Top speed (mph)	Time to top speed (s)	Acceleration (fps^2)	500ft time (s)	Max HP
1	12.00	28.11	0.39	45.21	0.09
2	11.00	50.00	0.20	51.62	0.09
3	10.00	38.00	0.21	60.90	0.09
Average	11.00	38.70	0.27	52.58	0.09

Table 7: Direct Drive Performance Data

Regenerative Braking Data			
Rider	Speed (mph)	Pressure (psi)	Distance After Charge (ft)
1.00	12.00	941.00	27.00
2.00	11.00	934.00	9.00
3.00	10.00	930.00	3.00
Average	11.00	935.00	13.00

Table 8: Regenerative Braking Performance Data

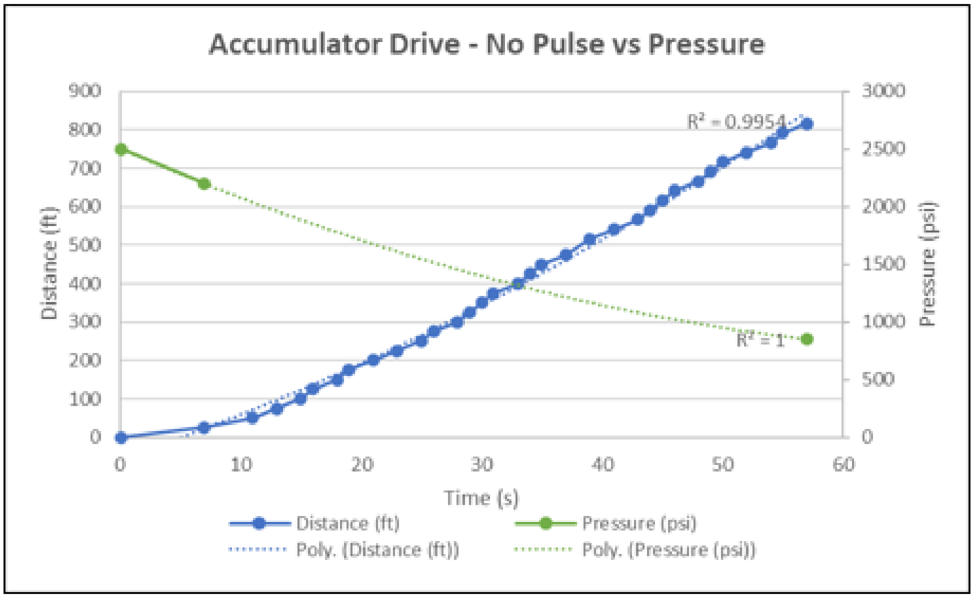

Figure 25: Accumulator Drive Distance and Pressure vs Time

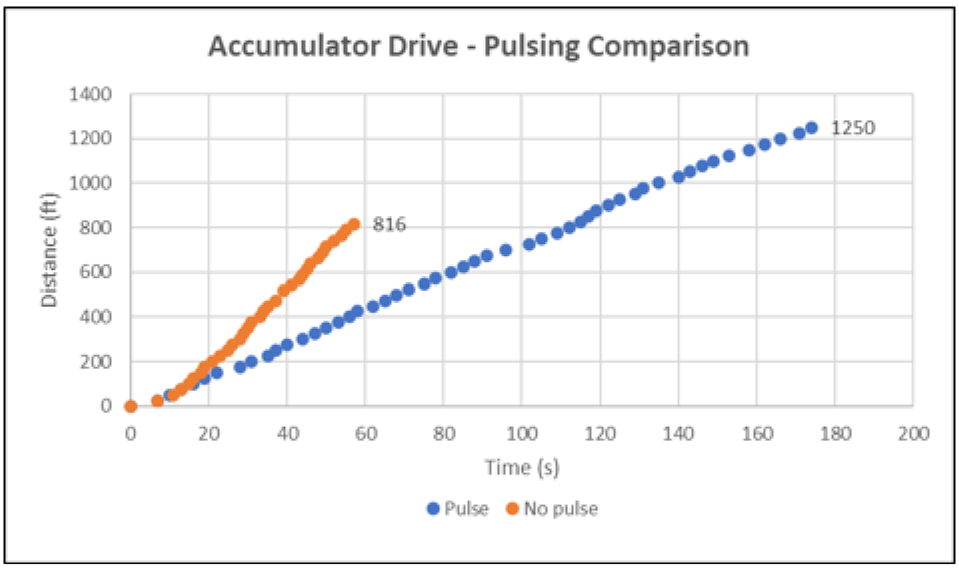

Figure 26: Accumulator Drive Pulse method vs Full Flow Method

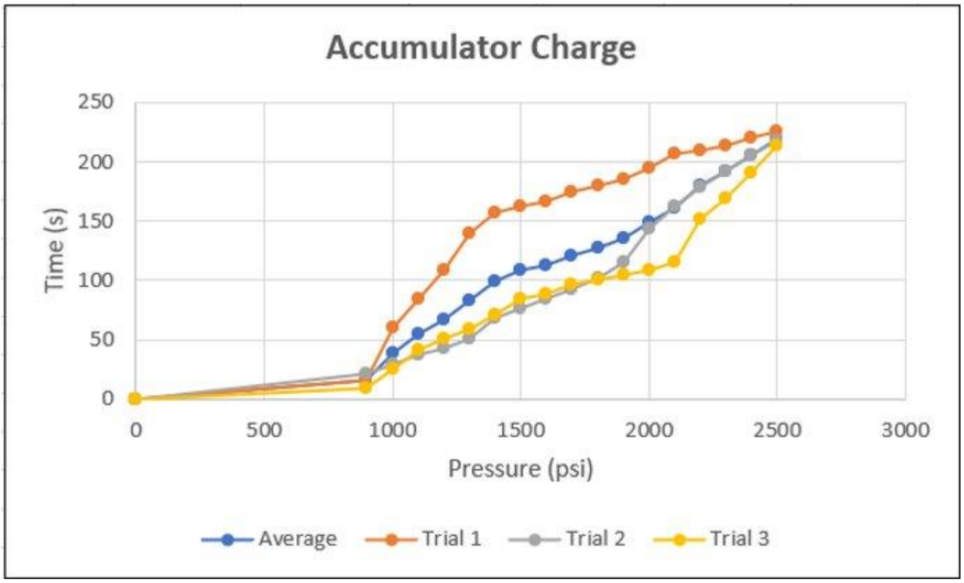

Figure 27: Accumulator Charge Time vs Pressure

The tables and Figures above were created during the data collection process using our pressure transducers and inductive sensors. Table 7 shows useful direct drive performance data which allowed us to gauge how well we would do in the FPVC sprint race. Table 8 shows regenerative braking trials performed by multiple riders. The important thing to note here is that we can travel some distance just using the energy stored from regen braking. This is an improvement on last year as their vehicle could not build up enough pressure to get moving again. Figure 25 illustrates distance traveled versus pressure in the accumulator over time using full flow (not pulsing). There is an indirect relationship between pressure and distance. The objective of the efficiency challenge is to go as far as possible with one full charge of the accumulator. Because of this we implemented the accumulator pulse method, and its benefit can be seen in Figure 25. You can see that not pulsing the accumulator allows for greater acceleration, but less overall distance traveled. Pulsing the accumulator allows us to conserve the amount of energy in the accumulator and travel further off one full charge. This increases the efficiency of our vehicle by 60% when using the accumulator. Lastly, Figure illustrates the time it takes to charge the accumulator to 2500psi. The pressure initially spikes to 850psi as this is our accumulator pre-charge, then rises slowly from there. Differences in slope are due to different riders switching mid-charge. Ultimately, regardless of the order in which riders change, the accumulator can be fully charged under 4 minutes which is significantly less than the 10 minutes allotted for charging during the competition.

PROJECT MANAGEMENT

To complete this project in time for the competition and to allow for a sufficient testing period, we utilized Microsoft Teams Tracker. This is a project management tool that allows users to create various tasks and assign due dates, team members, and "buckets" to each one. The program also provides useful charts to display how well tasks are met. Figure 28 shows the final breakdown of how well we were able to complete all tasks created. We created 157 tasks through the design and build process and completed all of them. They were not all completed by their respective due dates, but they were all completed before the competition. These tasks were assigned to different areas of the bike or "buckets" which made it easy to understand how much work was left for each area. The green bars illustrate the number of completed tasks. Overall, we were able to complete the vehicle on time and test the vehicle sufficiently before going to the competition.

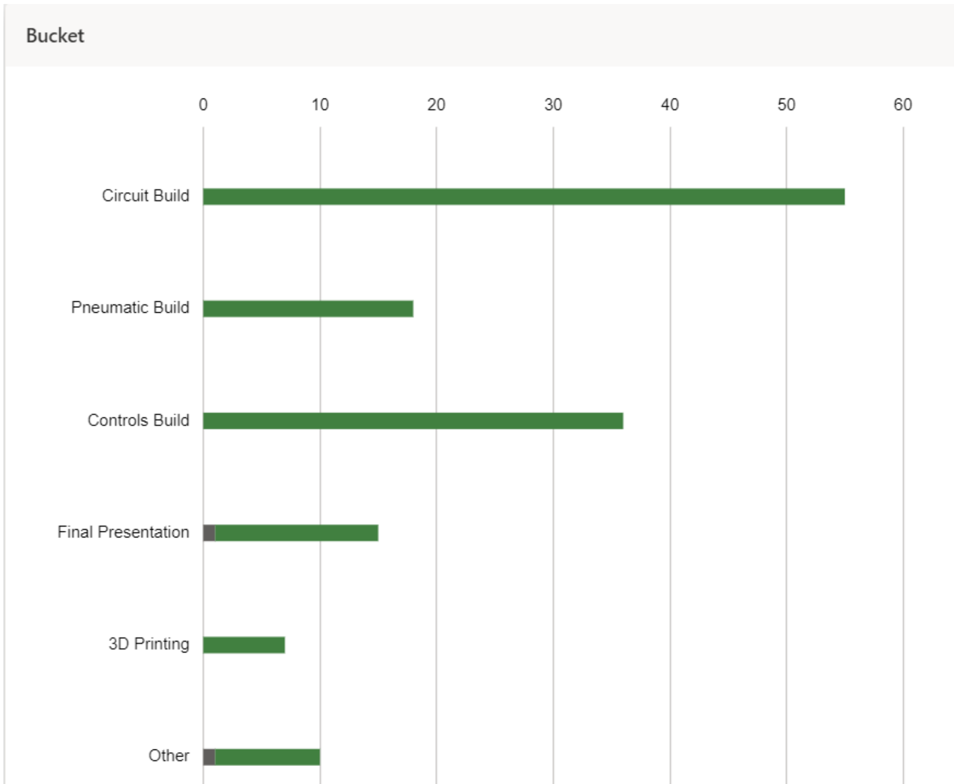

Figure 28: Microsoft Teams Tracker Output

CONCLUSIONS

Overall, this year's bike was one of the most achieving bikes in University of Cincinnati's history with the Fluid Power Vehicle Challenge. We met all vehicle improvement goals including achieving different vehicle speeds with the use of a tandem pump, implemented a pneumatic component, improved the regenerative braking circuit, improved component placement, and created more advanced controls including automatic valve activation. We were also to complete all the races and achieve satisfactory results per the competition goals. We Received 3rd place in the efficiency challenge, 3rd place in the endurance challenge, and 4th place in the sprint race. We also received awards for best presentations, best use of electronics, and won 3rd place overall out of 12 total teams. The judges found our design to be creative and innovative, which boosted our overall score.

Suggestions for Improvement

- Research gear ratios to increase tandem pump performance.
- Ensure motor chain cannot loosen under high pressure from accumulator.
  - Frame design and component placement can affect this.
- Find ways to totally disengage motor from back wheel to create little to no resistance (clutch)
  - Should be able to use pneumatics for this.
- Design hard pipes to have fewer joints and elbows. This will decrease flow resistance.
- Improve pneumatic kickstand idea. Make a new design or find a way to locate the stand below the center of gravity.
- Improve the regenerative braking circuit by separating this circuit from the main circuit.
  - This can be done with a separate, smaller wheel that is only used during regen.
  - This could also be done as it is in our current design but using a small gear pump that is specifically for regen and not a part of the main circuit.
- Look into a recumbent style bike design. There may be a way to reduce the weight of the vehicle, achieve a tight turn radius, and get enough power for adequate speed.
  - If using this style, ensure Ackerman steering mechanism is correct (wheel alignment for turning).
- Find a way to set different pre-charges or a variable pre charge in accumulator.
  - Optimal pre charge various between drive modes so being able to change it would be beneficial.

APPENDIX

```


IF
    PLC_PRG.x_Initialized = TRUE                        (* see if initialization routine was completed *)
        THEN
            controller_on       := TRUE;                (* set a bit that we will use to send to the screen to confirm controller is active *)
ELSE

            controller_on       := FALSE;               (* set that bit to false so we can see the failure on the controller *)
END_IF

(*new step*)

IF
    inputs.r_transducer_1_PSI < 250                     (* if the transducer is below 250 PSI we set a bit to note a warning on the screen *)
        THEN
            low_pressure_motor := TRUE;
ELSE
            low_pressure_motor := FALSE;
END_IF

(*new step*)
IF
    inputs.r_transducer_2_PSI < 250                     (* if the transducer is below 250 PSI we set a bit to note a warning on the screen *)
        THEN
            low_pressure_acc    := TRUE;
ELSE
            low_pressure_acc    := FALSE;
END_IF


(*Button Steps*)
(*Neutral / No Shift*)
IF
    CAN0_rx.screen_send_shift_1 = FALSE     (*DD*)
        AND
    CAN0_rx.screen_send_shift_2 = FALSE     (*Regen*)
        AND
    CAN0_rx.screen_send_AD = FALSE          (*added test button 3 AD*)
        AND
    CAN0_rx.screen_send_AC = FALSE          (*added test button 4 AC*)
        THEN
    enable_no_shift   := TRUE;

ELSIF
    CAN0_rx.screen_send_shift_1 = TRUE
        OR
    CAN0_rx.screen_send_shift_2 = TRUE
        OR
    CAN0_rx.screen_send_AD = TRUE           (*added test button 3 AD*)
```

```
    CAN0_rx.screen_send_shift_1 = TRUE
        OR
    CAN0_rx.screen_send_shift_2 = TRUE
        OR
    CAN0_rx.screen_send_AD = TRUE           (*added test button 3 AD*)
        OR
    CAN0_rx.screen_send_AC = TRUE           (*added test button 4 AC*)
        THEN
    enable_no_shift := FALSE;

END_IF

(*next button*)

IF
    enable_no_shift                 = TRUE
        AND
    CAN0_rx.screen_send_shift_1     = FALSE
        AND
    CAN0_rx.screen_send_shift_2     = FALSE
        AND
    CAN0_rx.screen_send_AD          =FALSE      (*added test button 3 AD*)
        AND
    CAN0_rx.screen_send_AC          =FALSE      (*added test button 4 AC*)
        THEN
            enable_output_0 := TRUE;        (*Smaller Pump 1, solenoid 17*)
            enable_output_1 := TRUE;        (*Larger Pump 2, solenoid 16*)
            enable_output_2 := FALSE;       (*3/2 to motor, solenoid 9*)
            enable_output_3 := FALSE;       (*3/2 from motor, solenoid 6*)
            enable_output_4 := FALSE;       (*Accumulator bi-poppet, solenoid 11*)
            enable_output_5 := FALSE;       (*Larger motor (external), solenoid 25*)
            enable_output_6 :=FALSE;        (*Accumulator bi-poppet, solenoid 11 ONLY FOR AD BUTTON*)
            indicator_light_motor1 := FALSE;
            indicator_light_motor12 := TRUE;
            indicator_light_pump1 := FALSE;     (*indicator for pump 1*)
            indicator_light_pump2 := FALSE;     (*indicator for pump 2*)
            indicator_light_pump12 := FALSE;    (*indicator for pumps 1 and 2*)

ELSIF                   (*DD Mode*)
    CAN0_rx.screen_send_no_shift = FALSE
        AND
    CAN0_rx.screen_send_shift_1     = TRUE
        AND
    CAN0_rx.screen_send_shift_2     = FALSE
        AND
    CAN0_rx.screen_send_AD          =FALSE      (*added test button 3 AD*)
        AND
    CAN0_rx.screen_send_AC          =FALSE      (*added test button 4 AC*)
        THEN
            enable_output_2 := FALSE;       (*3/2 to motor, solenoid 9*)
            enable_output_3 := FALSE;       (*3/2 from motor, solenoid 6*)
```

```
        THEN
            enable_output_2 := FALSE;           (*3/2 to motor, solenoid 9*)
            enable_output_3 := FALSE;           (*3/2 from motor, solenoid 6*)
            enable_output_4 := TRUE;            (*Accumulator bi-poppet, solenoid 11*)
            enable_output_6 :=FALSE;            (*Accumulator bi-poppet, solenoid 11 ONLY FOR AD BUTTON*)

            IF
                (*inputs.r_power_equ < 0.005(*0.29*)*)                      (*took 75% of theoretical max power for small pump*)
                inputs.r_wheelMPH < 1.5
            THEN
                enable_output_0 := FALSE;           (*Smaller Pump 1, solenoid 17*)
                enable_output_1 := TRUE;            (*Larger Pump 2, solenoid 16*)
                enable_output_5 := TRUE;            (*Larger motor (external), solenoid 25*)
                indicator_light_pump1 := TRUE;      (*indicator for pump 1*)
                indicator_light_pump2 := FALSE;     (*indicator for pump 2*)
                indicator_light_pump12 := FALSE;    (*indicator for pumps 1 and 2*)
                indicator_light_motor1 := FALSE;
                indicator_light_motor12 := TRUE;
            ELSIF
                (*inputs.r_power_equ > 0.005 (*0.29*) (*AND inputs.r_power_equ < 0.06(*0.65*)*)*)
                inputs.r_wheelMPH > 1.5 AND inputs.r_wheelMPH < 5.5
            THEN
                enable_output_0 := TRUE;            (*Smaller Pump 1, solenoid 17*)
                enable_output_1 := FALSE;           (*Larger Pump 2, solenoid 16*)
                enable_output_5 := TRUE;            (*Larger motor (external), solenoid 25*)
                indicator_light_pump1 := FALSE;     (*indicator for pump 1*)
                indicator_light_pump2 := TRUE;      (*indicator for pump 2*)
                indicator_light_pump12 := FALSE;    (*indicator for pumps 1 and 2*)
                indicator_light_motor1 := TRUE;
                indicator_light_motor12 := FALSE;
            ELSIF
                (*inputs.r_power_equ > 0.06 (*0.65*)                        (*took 75% of theoretical max power for large pump*)*)
                inputs.r_wheelMPH > 5.5
            THEN
                enable_output_0 := FALSE;           (*Smaller Pump 1, solenoid 17*)
                enable_output_1 := FALSE;           (*Larger Pump 2, solenoid 16*)
                enable_output_5 := TRUE;            (*Larger motor (external), solenoid 25*)
                indicator_light_pump1 := FALSE;     (*indicator for pump 1*)
                indicator_light_pump2 := FALSE;     (*indicator for pump 2*)
                indicator_light_pump12 := TRUE;     (*indicator for pumps 1 and 2*)
                indicator_light_motor1 := TRUE;
                indicator_light_motor12 := FALSE;
            END_IF
(*    END_IF
Delay(IN := FALSE);    (*reset timer*)*)


ELSIF                   (*Regen*)
    CAN0_rx.screen_send_no_shift = FALSE
        AND
    CAN0_rx.screen_send_shift_1 = FALSE
```

```
ELSIF                   (*Regen*)
    CAN0_rx.screen_send_no_shift = FALSE
        AND
    CAN0_rx.screen_send_shift_1   = FALSE
        AND
    CAN0_rx.screen_send_shift_2   = TRUE
        AND
    CAN0_rx.screen_send_AD        =FALSE        (*added test button 3 AD*)
        AND
    CAN0_rx.screen_send_AC        =FALSE        (*added test button 4 AC*)
        THEN
            enable_output_0 := TRUE;            (*Smaller Pump 1, solenoid 17*)
            enable_output_1 := TRUE;            (*Larger Pump 2, solenoid 16*)
            enable_output_2 := TRUE;            (*3/2 to motor, solenoid 9*)
            enable_output_3 := TRUE;            (*3/2 from motor, solenoid 6*)
            enable_output_4 := FALSE;           (*Accumulator bi-poppet, solenoid 11*)
            enable_output_5 := TRUE;            (*Larger motor (external), solenoid 25*)
            enable_output_6 :=FALSE;            (*Accumulator bi-poppet, solenoid 11 ONLY FOR AD BUTTON*)
            indicator_light_motor1 := TRUE;
            indicator_light_motor12 := FALSE;
            indicator_light_pump1 := FALSE;     (*indicator for pump 1*)
            indicator_light_pump2 := FALSE;     (*indicator for pump 2*)
            indicator_light_pump12 := FALSE;    (*indicator for pumps 1 and 2*)

ELSIF                   (*AD Mode*)
    CAN0_rx.screen_send_no_shift = FALSE
        AND
    CAN0_rx.screen_send_shift_1   = FALSE
        AND
    CAN0_rx.screen_send_shift_2   = FALSE
        AND
    CAN0_rx.screen_send_AD        =TRUE         (*added test button 3 AD*)
        AND
    CAN0_rx.screen_send_AC        =FALSE        (*added test button 4 AC*)
        THEN
            enable_output_0 := TRUE;            (*Smaller Pump 1, solenoid 17*)
            enable_output_1 := TRUE;            (*Larger Pump 2, solenoid 16*)
            enable_output_2 := FALSE;           (*3/2 to motor, solenoid 9*)
            enable_output_3 := FALSE;           (*3/2 from motor, solenoid 6*)
            enable_output_4 := FALSE;           (*Accumulator bi-poppet, solenoid 11*) (*controlled by physical button input*)
            enable_output_6 :=TRUE;             (*Accumulator bi-poppet, solenoid 11 ONLY FOR AD BUTTON*)
            indicator_light_pump1 := FALSE;     (*indicator for pump 1*)
            indicator_light_pump2 := FALSE;     (*indicator for pump 2*)
            indicator_light_pump12 := FALSE;    (*indicator for pumps 1 and 2*)

            IF
                inputs.r_transducer_2_PSI > 2200 (* Unsure of value, change at later time*)
            THEN
                enable_output_5 := FALSE;           (*Larger motor (external), solenoid 25*)
```

```
            indicator_light_pump12 := FALSE;    (*indicator for pumps 1 and 2*)

            IF
                inputs.r_transducer_2_PSI > 2200 (* Unsure of value, change at later time*)
            THEN
                enable_output_5 := FALSE;           (*Larger motor (external), solenoid 25*)
                indicator_light_motor1 := FALSE;
                indicator_light_motor12 := TRUE;
            ELSIF
                inputs.r_transducer_2_PSI < 2200
            THEN
                enable_output_5 := TRUE;            (*Larger motor (external), solenoid 25*)
                indicator_light_motor1 := TRUE;
                indicator_light_motor12 := FALSE;
            END_IF

ELSIF                   (*AC Mode*)
    CAN0_rx.screen_send_no_shift = FALSE
        AND
    CAN0_rx.screen_send_shift_1   = FALSE
        AND
    CAN0_rx.screen_send_shift_2   = FALSE
        AND
    CAN0_rx.screen_send_AD        =FALSE        (*added test button 3 AD*)
        AND
    CAN0_rx.screen_send_AC        =TRUE         (*added test button 4 AC*)
        THEN
            enable_output_2 := TRUE;            (*3/2 to motor, solenoid 9*)
            enable_output_3 := FALSE;           (*3/2 from motor, solenoid 6*)
            enable_output_4 := FALSE;           (*Accumulator bi-poppet, solenoid 11*)
            enable_output_5 := FALSE;           (*Larger motor (external), solenoid 25*)
            enable_output_6 :=FALSE;            (*Accumulator bi-poppet, solenoid 11 ONLY FOR AD BUTTON*)
            indicator_light_motor1 := FALSE;
            indicator_light_motor12 := TRUE;

    Delay(IN :=TRUE, PT :=T#3S);
        IF  Delay.Q THEN
        Delay(IN:=TRUE);

            IF
                inputs.r_transducer_2_PSI < 900
            THEN
                enable_output_0 := FALSE;           (*Smaller Pump 1, solenoid 17*)
                enable_output_1 := FALSE;           (*Larger Pump 2, solenoid 16*)
                indicator_light_pump1 := FALSE;     (*indicator for pump 1*)
                indicator_light_pump2 := FALSE;     (*indicator for pump 2*)
                indicator_light_pump12 := TRUE;     (*indicator for pumps 1 and 2*)
            ELSIF
                inputs.r_transducer_2_PSI > 900 AND inputs.r_transducer_2_PSI < 1000
            THEN
                enable_output_0 := TRUE;            (*Smaller Pump 1, solenoid 17*)
```

```
            inputs.r_transducer_2_PSI > 900 AND inputs.r_transducer_2_PSI < 1000
        THEN
            enable_output_0 := TRUE;            (*Smaller Pump 1, solenoid 17*)
            enable_output_1 := FALSE;           (*Larger Pump 2, solenoid 16*)
            indicator_light_pump1 := FALSE;     (*indicator for pump 1*)
            indicator_light_pump2 := TRUE;      (*indicator for pump 2*)
            indicator_light_pump12 := FALSE;    (*indicator for pumps 1 and 2*)
        ELSIF
            inputs.r_transducer_2_PSI > 1000
        THEN
            enable_output_0 := FALSE;           (*Smaller Pump 1, solenoid 17*)
            enable_output_1 := TRUE;            (*Larger Pump 2, solenoid 16*)
            indicator_light_pump1 := TRUE;      (*indicator for pump 1*)
            indicator_light_pump2 := FALSE;     (*indicator for pump 2*)
            indicator_light_pump12 := FALSE;    (*indicator for pumps 1 and 2*)
        END_IF
    ELSIF  NOT(Delay.Q) THEN
    RETURN;
  END_IF
    Delay(IN :=FALSE);    (*reset timer*)

ELSE
        enable_output_0 := FALSE;           (*Smaller Pump 1, solenoid 17*)
        enable_output_1 := TRUE;            (*Larger Pump 2, solenoid 16*)
        enable_output_2 := FALSE;           (*3/2 to motor, solenoid 9*)
        enable_output_3 := FALSE;           (*3/2 from motor, solenoid 6*)
        enable_output_4 := FALSE;           (*Accumulator bi-poppet, solenoid 11*)
        enable_output_5 := TRUE;            (*Larger motor (external), solenoid 25*)
        enable_output_6 :=FALSE;            (*Accumulator bi-poppet, solenoid 11 ONLY FOR AD BUTTON*)
        indicator_light_motor1 := TRUE;
        indicator_light_motor12 := FALSE;
        indicator_light_pump1 := TRUE;      (*indicator for pump 1*)
        indicator_light_pump2 := FALSE;     (*indicator for pump 2*)
        indicator_light_pump12 := FALSE;    (*indicator for pumps 1 and 2*)
END_IF
  (*(*IF
      enable_output_5 = TRUE
  THEN
      indicator_light_motor1 := TRUE;
      indicator_light_motor12 := FALSE;
  ELSIF
      enable_output_5 = FALSE
  THEN
      indicator_light_motor1 := FALSE;
      indicator_light_motor12 := TRUE;
  END_IF*)
END_IF*)

(*

```


```
        indicator_light_pump12 := FALSE;    (*indicator for pumps 1 and 2*)
END_IF
  (*(*IF
      enable_output_5 = TRUE
  THEN
      indicator_light_motor1 := TRUE;
      indicator_light_motor12 := FALSE;
  ELSIF
      enable_output_5 = FALSE
  THEN
      indicator_light_motor1 := FALSE;
      indicator_light_motor12 := TRUE;
  END_IF*)
END_IF*)

(*

```


```
IF CAN0_rx.screen_button_input_6 = TRUE THEN
enable_output_6 := TRUE;
ELSE
enable_output_6 :=FALSE;
END_IF

IF CAN0_rx.screen_button_input_7 = TRUE THEN
enable_output_7 := TRUE;
ELSE
enable_output_7 :=FALSE;
END_IF
*)
```