Control System and User Interface for Hydraulic Bike

Project # 2

Design Review

TA: Jacob Bryan

ECE 445

February 23, 2015

Table of Contents

1. Introduction 3
1.1 Statement of Purpose 3
1.2 Objectives 3
1.2.1 Goals 3
1.2.2 Benefits 3
1.2.3 Functions and Features 3
2. Context for Project 4
2.1 Competition Overview 4
2.2 Mechanical System Description 4
2.3 Hydraulic Circuits to Support 5
3. Design 6
3.1 Block Diagram 6
3.2 Circuit Schematic and Software Flow Chart 6
3.3 User Interface 7
3.4 Calculation 8
3.5 Block Descriptions 9
User Interface Module 9
Control System Module 9
Power Module 10
Sensors Module 10
3.6 Requirements and Verification for the User Interface Module 11
4. Cost and Schedule 12
4.1 Cost Analysis 12
4.1.1 Labor 12
4.1.2 Parts 12
4.1.3 Grand Total 12
4.2 Schedule 13
5. Ethics 14
5.1 Safety Statement 14
5.2 Application of IEEE Code of Ethics 15

1. Introduction

1.1 Statement of Purpose

Parker-Hannifin, a fluid power systems company, hosts an annual competition for the design of a chainless bicycle. A control system, user interface, and electrical power are needed for UIUC's entry into this competition. Previous control systems for these bicycles used either commercially-available systems[1] or discrete logic[2]. A new and unique approach to this problem is a custom-built control system and user interface. This has the possibility of being tailored to the end users, who are both accessible and on campus.

Automation and control systems are of personal interest, so working on the control system and operator interface for a real product would be a significant personal and professional benefit.

1.2 Objectives

1.2.1 Goals

- Electro-mechanically control flow of fluid in hydraulic circuits
- React to user input and give feedback to rider
- Stay within competition guidelines
- Help create a competitive or winning hydraulic bicycle
- No control failures during competition (happened in 2013-2014)

1.2.2 Benefits

- Automatic control of electromechanical valves improves bicycle's ease of use (versus manual valves)

1.2.3 Functions and Features

- Require user interaction to change hydraulic circuits (for safety of rider)
- Interpret sensor data and ensure operation is within safe parameters
- Visually display current accumulator pressures
- Support functioning of mechanical systems on bicycle, in addition to above
    - Power and control valves
    - Work around physical constraints on bike for control system and user interface for both dimensions and placement

[1] Team using Parker-Hannifin IQAN system: https://engineering.purdue.edu/ABE/InfoFor/CurrentStudents/SeniorProjects/2012/GeskeLamneckSparenbergEtAl

[2] Team using discrete logic (page 19): http://deepblue.lib.umich.edu/bitstream/handle/2027.42/86206/ME450?sequence=1

2. Context for Project

2.1 Competition Overview

The 2014/2015 Chainless Challenge is a competition sponsored by Parker-Hannifin with the goal being to develop a fluid-powered bicycle, preferably without a chain. The competition consists of an endurance race, an efficiency race, and a sprint race. Points are awarded for placing in races as well as for various design features.

2.2 Mechanical System Description

A brief explanation of the mechanical system that is being controlled is necessary to give context to the rest of this document. Fluid power is provided by the rider and is input at the pedals. The fluid power can either be stored in the accumulators or sent directly to the hydraulic motor, as determined by the valves in the manifold. The hydraulic circuits to support this are listed below in section 1.3.1. The accumulators have valves that can either be setup to accept incoming fluid or expel outgoing fluid to the motor. Incoming fluid can either come from the rider (charge paths) or from the motor (regenerative braking path). Figure 1 is an overview of how components are connected.

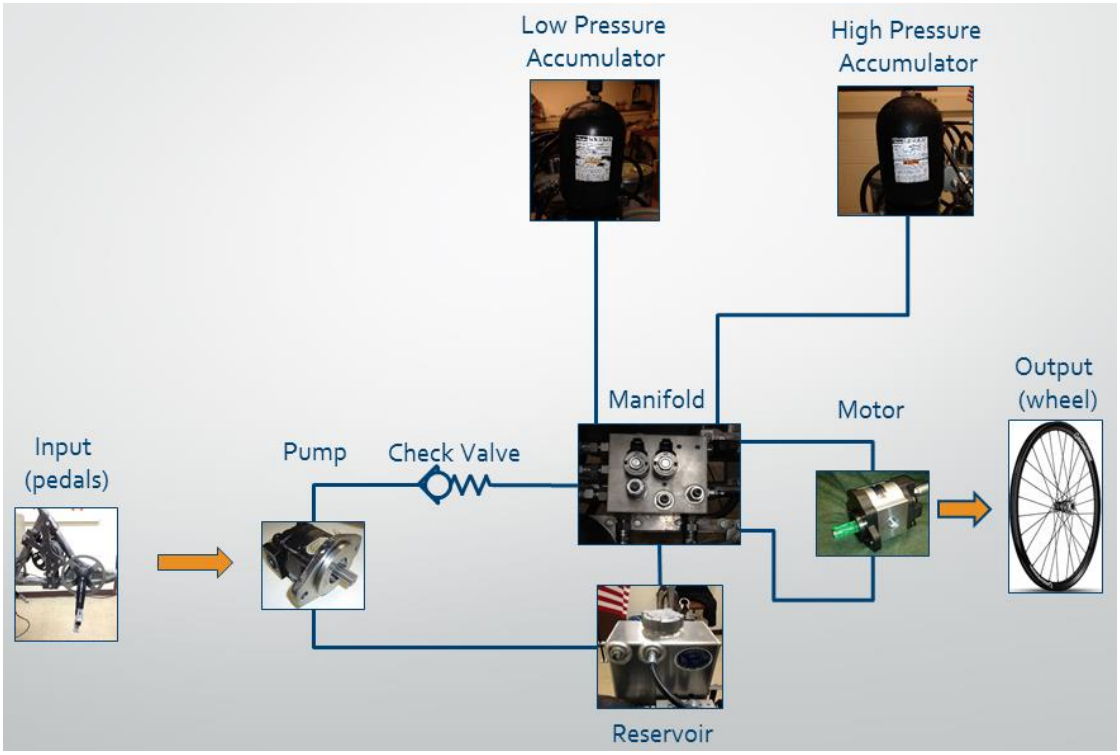


Figure 1. Overview of mechanical components.
Blue lines represent hosing for fluid.

2.3 Hydraulic Circuits to Support

There are 7 hydraulic circuits that need to be supported by the electrical system, listed below. A PDF is attached at the end which shows the various fluid paths in the latest mechanical design.

1. Direct Drive Path (Default)
2. Coasting Path
3. Charge Low Pressure Accumulator and Coast
4. Charge Low and High Pressure Accumulators
5. Discharge Low Pressure Accumulator
6. Discharge High Pressure Accumulator
7. Regenerative Braking Path

3. Design

3.1 Block Diagram

Figure 2 shows the electrical component of the hydraulic bicycle, which consists of three subsystems: control system, user interface, and power management.

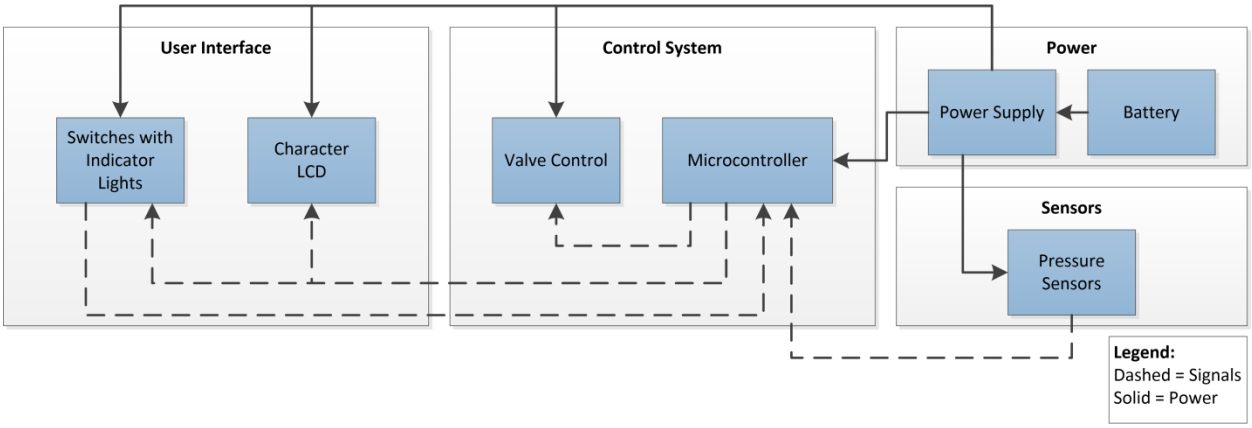


Figure 2. Block diagram of electrical system.

3.2 Circuit Schematic and Software Flow Chart

Circuit schematic attached at end. The software flow chart for the control system is shown in Figure 3.

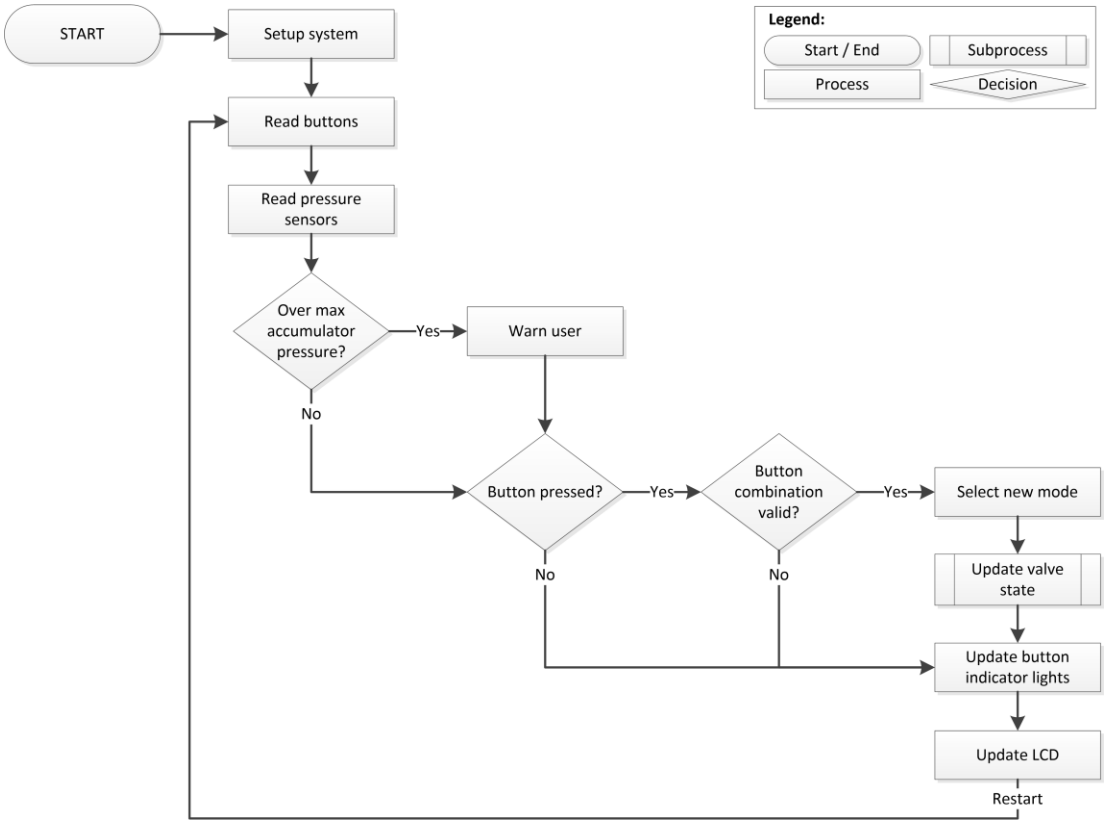


Figure 3. Software flow chart for control system.

3.3 User Interface

Based on the need to switch between the 7 hydraulic circuits, or modes, I designed a user interface with Ian O'Leary from the MechSE team working on this project. We wanted to avoid having 7 buttons as it is hard to place 7 buttons in accessible places on bike handlebars. Since the charging and discharging states had variations for each accumulator combination, we decided to use a selector switch for to choose the accumulator and then have one button each for charge and discharge. Electrical power is not intended to be needed other than power up and power down, so it is slightly less accessible. Indicator lights near the buttons provide feedback when a button is pressed.

A photo of the prototyped user interface is in Figure 4. It was discovered that acrylic is extremely reflective, so the lights were turned off to take the photo.

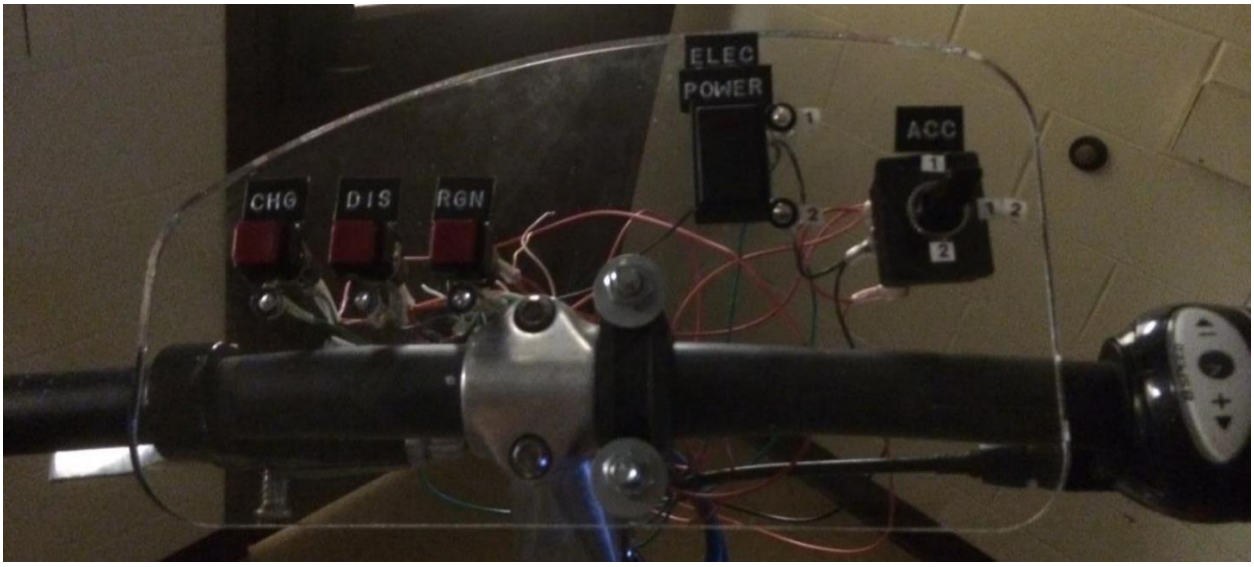


Figure 4. User interface prototype.

Future iterations of the user interface will include more user feedback via one or two character LCD(s). The accumulator pressures and current mode are data that will be shown.

3.4 Calculation

I examined the current that will need to be drawn from the microcontroller in order to turn on a relay and its associated LED. This is necessary because not only does each microcontroller pin have a current limit, but the overall current limit is less than the sum of the pins' current limits.

The relay coil resistance is 125Ω ± 12.5 from the datasheet. The relay will be operated at 5VDC. This results in equation (1).

$$I_{relay,max} = \frac{5V}{112.5\Omega} = 44.4\text{mA} \quad (1)$$

The LED current is dependent on the V_F of the LED and the resistance in series. For an average LED and resistor in the schematic, $V_{F,min}$ = 2V. An LED current of 10mA is desired. The combined current is calculated in equation (2).

$$I_C = 44.4 + 10 = 55.44\text{mA} \quad (2)$$

The combined current of 54.4mA is greater than the current most microcontroller pins can handle. Specifically, this number is above the 20mA recommended pin current and the 40mA max current per pin for the ATmega328-PU.

A transistor circuit acting as a switch (with current-limiting resistors) solves this problem by requiring a small amount of input current relative to the output current, related by the DC current gain, h_{FE}. The datasheet for the NPN transistor being used (SS8050) states $h_{FE} > 45$ for I_C = 5mA. h_{FE} increases with higher I_C, but we can do the calculations with the overly conservative value h_{FE} = 45 and still achieve an acceptable result. Equation (3) is the equation for DC current gain, which is then solved for I_B in equation (4).

$$h_{FE} = \frac{I_C}{I_B} = 45 \quad (3)$$

$$\therefore I_B = \frac{55.44}{45} = 1.23\text{mA} \quad (4)$$

A current draw of 1.23mA for each of the 8 relay/LED indicator circuits is under the current recommendation of 20mA per pin and the max pin source current limitation for any group of ports (the lowest of which is 100mA for 9 ports), acccording to the datasheet. Thus, switching the relays can be done with the microcontroller if a transistor circuit is added as a buffer.

3.5 Block Descriptions

User Interface Module

The user interface module is the main way to facilitate user interaction and display information to the user. The user needs to be able to select between 6 operating modes, view the operating mode on the Character LCD, and view sensor data on the Character LCD, at the very least. The user interface will be mounted on the handlebars for accessibility and convenience. Future addition of features will most impact the user interface.

3.5.1 Switches with Indicator Lights

Three buttons with indicator lights will be used to select Charge, Discharge, and Regenerative Braking. A 3-position selector switch will select between charging/discharging the low and/or high pressure accumulators when the Charge or Discharge modes are selected. All buttons will be momentary, since if 'toggle' functionality is desired, it can be achieved in software. Switch circuitry will include pull-up resistors. Inputs and outputs will likely be communicated with the microcontroller via an I2C/TWI port expander IC.

3.5.2 Character LCD

The Character LCD block will be composed of at least one 20x4 character LCD and will display the current pressure of both accumulators and text indicating the current state. If a PID controller or similar functionality is later added to the project, the character LCD might also show suggestions for the best mode as based on accumulator pressures, race type, etc. Character LCD(s) will likely communicate with the microcontroller via I2C/TWI using a commercially-available "LCD backpack".

Control System Module

The control system's microcontroller is responsible for sending/receiving all signals to other modules.

3.5.3 Microcontroller

The microcontroller is responsible for reading the button state on the user interface, sending display data to the character LCD(s), controlling the indicator lights, reading the pressure sensors, and turning on/off the valves. For rapid prototying, an ATmega 328P is currently in use since it is supported by the Arduino environment (the 328P is used in the Arduino UNO) and it is available in a 28-pin DIP. The microcontroller is not physically size-constrained and is not very power-constrained, as discussed in the Battery block description. Memory is the biggest constraint, especially if a PID controller with state suggestions is to be implemented and the ASCII text for the LCD is stored. Extra EEPROM or similar non-volatile memory could be added over an I2C/TWI bus if memory became an issue.

3.5.4 Valve Control

The electromechanical valves require 12V at 1.2A in order to actuate. Since an ATmega 328P can only source 200mA across all pins, either transistors or small power relays are required. Power relays were chosen because they effectively isolate the input from the output. Since turning on the valves will create a momentary drop in supply voltage, future separation of the power supplies may be desired. Thus, power relays were chosen. The power relays currently in use require 80mA to turn on, so a transistor is being used to drive them. All outputs for valves will also have indicator LEDs to aide in troubleshooting.

Power Module

Power from the battery will be stepped down to the necessary voltages and currents.

3.5.5 Power Supply

The power supply will step down to 5 VDC for all components except for the valve outputs and will pass the 12V battery voltage to the valves (through the relays). While the valves do not have a polarity, the 5VDC circuitry can be damaged by reversed-polarity, so a protection circuit will be added. A power PMOS is currently being used for this purpose.

3.5.6 Battery

The battery is required to be 12V and will need to power the entire system for 1 day of the competition. While exact power usage is unknown, it has been estimated that 20Ah is the bare minimum, based on 8 hours usage at 2.5 Amps. A lead-acid battery will likely be used due to cost vs capacity vs volume.

Sensors Module

Pressure sensors are currently the only sensors anticipated.

3.5.7 Pressure Sensors

Pressure sensors attached to the accumulators will provide feedback on accumulator pressures. The pressure sensors that have been selected are Parker IQAN-SP pressure sensors, which take 5 VDC input and return an analog voltage from 0 to 4.5 VDC.

3.6 Requirements and Verification for the User Interface Module

Block	Requirement	Verification	Points
1. Switches with indicator lights	**a.** HIGH or LOW button state is read over I2C **b.** Indicator lights are controlled over I2C	**a.** Connect I2C master to bus and read from the I2C port expander with all buttons unpressed. Press one button at a time, continuing to read the state over I2C. Data should indicate logic LOW when buttons are unpressed and logic HIGH when buttons are pressed. 5 button presses and depresses should be properly registered. **b.** Start with all indicator lights off. Connect I2C master to bus and write to I2C port expander the data required to turn on each indicator light, then turn off each indicator light. Only the indicator lights specified in the I2C packets should change state on each write. Each indicator light should be tested twice in this manner.	15
2. Character LCD	**a.** Character LCD displays ASCII text from microcontroller	**a.** Connect I2C master to bus and send ASCII text to LCD backpack over bus. Ensure ASCII text sent fills character LCD and that all characters are displayed where expected.	5
3. Micro controller	**a.** Microcontroller can communicate with all I2C slaves **b.** Output a low signal 25% below the turn-on voltage (1VDC) of the valve control circuit (ie. 0.75VDC) **c.** Output a high signal 25% above above the saturation voltage (1.2VDC) of the valve control circuit (ie. 1.5VDC)	**a.** Connect microcontroller to I2C bus as master. Verify microcontroller can perform all I2C tasks mentioned elsewhere in this table – read switch states, control indicator lights and read pressure sensors. **b.** Program the microcontroller to send a low signal. Check voltage is 25% below threshold voltage of transistor (0.75V) relative to ground using DMM in parallel with microcontroller. **c.** Program the microcontroller to send a low signal. Check voltage is 25% above the threshold voltage of transistor (1.5V) relative to ground using DMM in parallel with microcontroller.	30
4. Valve control	**a.** Turn on output given an input 25% above transistor saturation voltage (1.2VDC) **b.** Turn off output given an input 25% below transistor turn-on voltage (1VDC)	**a.** Connect a variable power supply to the input of the circuit. Sweep the voltage from 25% below $V_{saturation}$ (0.9V) to 25% above $V_{saturation}$ (1.5V). Relay should turn on and indicator should light. **b.** Connect a variable power supply to the input of the circuit. Sweep the voltage from 25% above $V_{turn\text{-}on}$ (1.25V) to 25% below $V_{turn\text{-}on}$ (0.75V). Relay and indicator should turn off.	20
5. Power supply	**a.** Provide 5 ± 0.25 VDC at 2 ± 0.25 A **b.** If input voltage is connected with reversed polarity, output 0 V +0.1 -0.	**a.** Place DMM in parallel with power supply output. Measure voltage difference – should be between 4.75 and 5.25 V. Next, add a constant-current load to the power supply output and set it to 2 ± 0.1 A. Let run for 1 hour. At the end of the hour, if the DMM still is within the voltage requirements, power supply passes. Do not use heatsinks or airflow beyond normal application. **b.** Disconnect power supply output from rest of circuits. Connect battery in reverse polarity. Connect DMM to power supply output and measure voltage – should be within 0 V and 0.1 V.	5
6. Battery	**a.** Check battery can last 1 day at event. Expected use is 2.4A for more than 6 hours.	**a.** Charge battery to full capacity. Discharge battery with constant current load of 2.4 ± 0.1 A and time how long it takes to discharge to 11 ± 0.5 V. Must take more than 6 hours. Recharge battery after full discharge to prevent permanent damage.	10
7. Pressure sensors	**a.** Check pressure reading shows atmospheric pressure ± 20% when fluid is at atmospheric pressure **b.** Check pressure readings increase as each accumulator is charged **c.** Check pressure readings decrease as each accumulator is charged	**a.** Discharge both accumulators. Disconnect hosing to each accumulator at manifold end. Let fluid equalize with room temperature. Read pressure sensor output with microcontroller and verify that it reads atmospheric pressure ± 20%. **b.** Set assembled and otherwise functional bike to charge desired accumulator. Pedal bike to charge accumulator and read pressure sensor output with microcontroller to verify pressure is increasing. Test other accumulator. **c.** Set assembled and otherwise functional bike to discharge desired accumulator. Read pressure sensor output with microcontroller to verify pressure is decreasing as bike moves forward. Test other accumulator.	15

4. Cost and Schedule

4.1 Cost Analysis

4.1.1 Labor

Name	Hourly Rate	Total Hours Invested	Total Labor
Iain	$30.00	400	$30,000

4.1.2 Parts

Item	Part Number	Quantity	Cost
Microcontroller	ATMEGA328P-PU	1	$3.74
Power switch	100SP1T1B4M2QE	1	$2.11
Fuse and holder	Unknown	1	$5.00
Reverse polarity transistor	IRF9540PBF	1	$1.72
Capacitors, 10uF ±20%*	ESK106M050AC3AA	2	$0.32
Voltage regulator	LM7805-ACT	1	$0.67
LEDs (red, green, blue)*	Cree C503B-XXX	11	$1.65
Status LED transistor*	Low current NPN, ie. SS8050	1	$0.25
Valve control transistors*	SS8050DBU	9	$2.61
Resistors ±5%*	CF14JT*xxxx*	23	$2.30
Terminal block header, 2 pos	284518-2	17	$11.39
Terminal block plug, 2pos	284507-2	17	$21.69
Ceramic resonator*	CSTLS16M0X53-B0	1	$0.40
Capacitors, 22pF ±20%*	FK28C0G1H220J	2	$0.58
Pushbutton switch*	B3F-1000	1	$0.35
Diode*	1N4004	9	$0.99
Relay*	T77S1D10	9	$20.52
Character LCD, 20x4	Adafruit 198	1	$17.95
I2C port expander	MCP23017-E/SP	1	$1.44
I2C LCD backpack	Adafruit 292	1	$10.00
I2C ADC	ADS1015	1	$9.95
Illuminated pushbuttons	Adafruit 1439	4	$6.00
Selector switch	AutomationDirect GCX3370-22	1	$14.50
Battery	UB12220	1	$43.96
Wire	--	1	$10.00
		TOTAL	$190.09

* Equivalent part available free from ECE Service Shop.

4.1.3 Grand Total

Section	Total
Labor	$30,000
Parts	$190.09
Grand Total	***$30,190.09***

4.2 Schedule

Week	Task	Responsibility
2/9	Test individual subcircuits (done) Finalize proposal Prepare mock design review Finish basic functions prototype	Iain
2/16	Order parts Finalize PCB (and order?) Wire and mount user interface	Iain
2/23	Order PCB, if not already Mount PCB on bike, if ready Design review Bicycle testing, if PCB ready	Iain
3/2	Mount PCB on bike Test bike Create backup set of electronics for competition (1/3)	Iain
3/9	Create backup set of electronics for competition (2/3)	Iain
3/16	Create backup set of electronics for competition (3/3)	Iain
3/23	Ship bike	Iain
3/30	Individual progress report due 4/1 Shoot video for mock demo and demo	Iain
4/6	Create presentation for mock demo Compete from 4/7 to 4/9	Iain
4/13	Improve on mock demo presentation for demo	Iain
4/20	Create final presentation	Iain
4/27		Iain
5/4	Final papers due 5/6 Lab checkout on 5/7	Iain

5. Ethics

5.1 Safety Statement

This project both interacts with a human being and controls the bike that a human being rides. Moreover, the system controls the flow of hydraulic fluid that is used to store energy and can be discharged by actuating the correct electro-mechanical valves.

An alternative to an electromechanical control system is a system that is entirely mechanically-actuated with levers. However, depending on the order the valves are opened and closed, the user could release stored energy in an unintended way (ie. discharge both accumulators instead of just one). Since we anticipate the bicycle rider needing to change states several times during a race, while steering a 200 lb bike, the risk is relatively large. An electrical control system turning on and off electromechanical valves eliminates this risk and can be programmed to reject unsafe changes in valves, based on the sensor and user input it receives. In this way, it reduces the risk of operator error by limiting the choices of valve states. However, the electrical control system processes data faster than a human can react, so if it outputs an unsafe combination of valves, the bicycle may abruptly accelerate or decelerate without warning to the user. In addition, care must be taken that loss of power to the control system or all valves does not create an unsafe condition.

With this in mind, the valves and hydraulic circuit have been designed by the Mechanical Engineering team so that their default state (that in which no valves are powered/actuated) is the Direct Drive circuit, which does not involve storing or releasing energy. The user will have the ability to remove power from the control system and the valves in order to return to this state if the control system fails, via a power switch on the user interface. In addition, a safety relay has been added. One side of the load goes to the battery and the other goes to the other relay loads (see schematic, attached). In this way, IFF the safety relay is on, the valves can be actuated by turning on the other 8 valve relays. If the safety relay is off, turning on the other relays will not actuate the valves. When designing the microcontroller code that switches valve states, care will be taken to examine undesirable combinations of valves that may occur when switching between states.

5.2 Application of IEEE Code of Ethics

This project is closely related to several items of the IEEE Code of Ethics, namely:

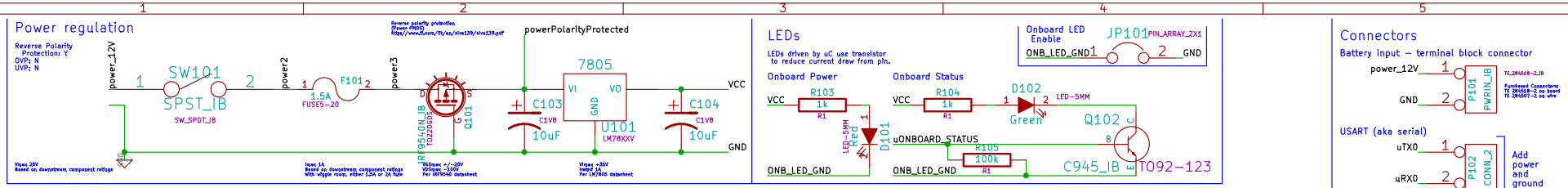
> 1. to accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to disclose promptly factors that might endanger the public or the environment;
>
> 3. to be honest and realistic in stating claims or estimates based on available data;
>
> 5. to improve the understanding of technology; its appropriate application, and potential consequences;
>
> 7. to seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others;

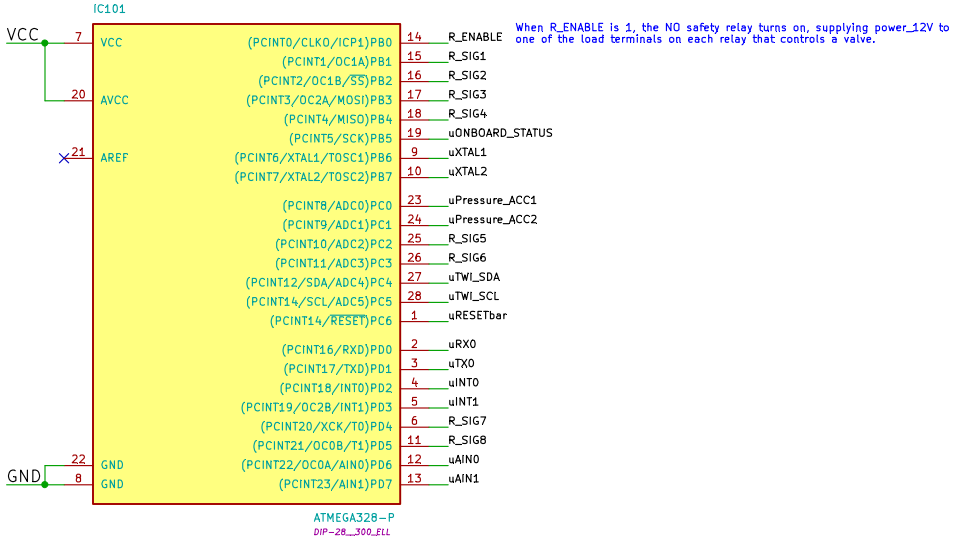
Item 1 is very relevant, for the reasons discussed in the previous section, Safety Statement.

Item 3 is relevant because making false claims regarding the capabilities of the control system or user interface could lead the rider to trust the control system and user interface with their life in a situation they would not have been comfortable with when presented unbiased facts. Making false claims also creates moral and legal liability for negative outcomes of using the user interface and control system.

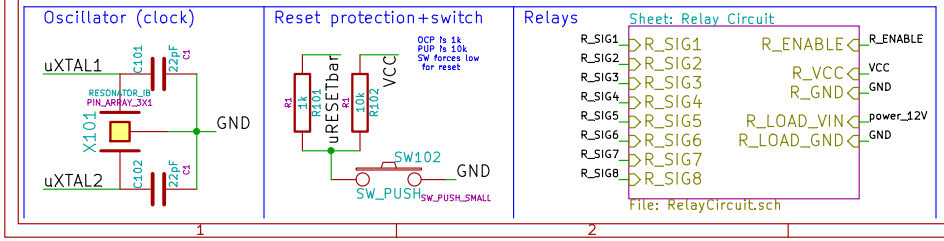
Item 5 is important – equipment that fails could have somewhat catastrophic failures and it is important to understand the technical limitations of the technology being used, how to use it safely within those limitations, etc.

Item 7 is important because I am working with a team of mechanical engineers, and while they do not contribute directly to the reports, they create content I choose to include in my reports. Their work must be understood as their own with proper citations.





When R_ENABLE is 1, the NO safety relay turns on, supplying power_12V to one of the load terminals on each relay that controls a valve.



Notes

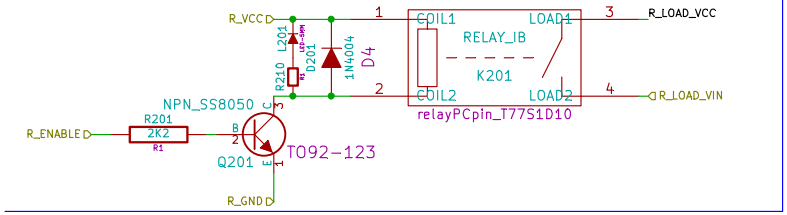
1. User interface will likely communicate with microcontroller entirely over TWI/I2C.
2. Proportional valves will need to be controlled with a remote DAC over TWI/I2C, since there are no DACs on the ATmega328P.

Senior Design Project	
ECE 445	
Iain Brearton	
File: ElectronicControls.sch	
Sheet: /	
Title: Electronic Controls	
Size: USLetter Date: 23 feb 2015	**Rev: 0.6**
KiCad E.D.A. eeschema (2013-07-07 BZR 4022)-stable	Id: 1/2

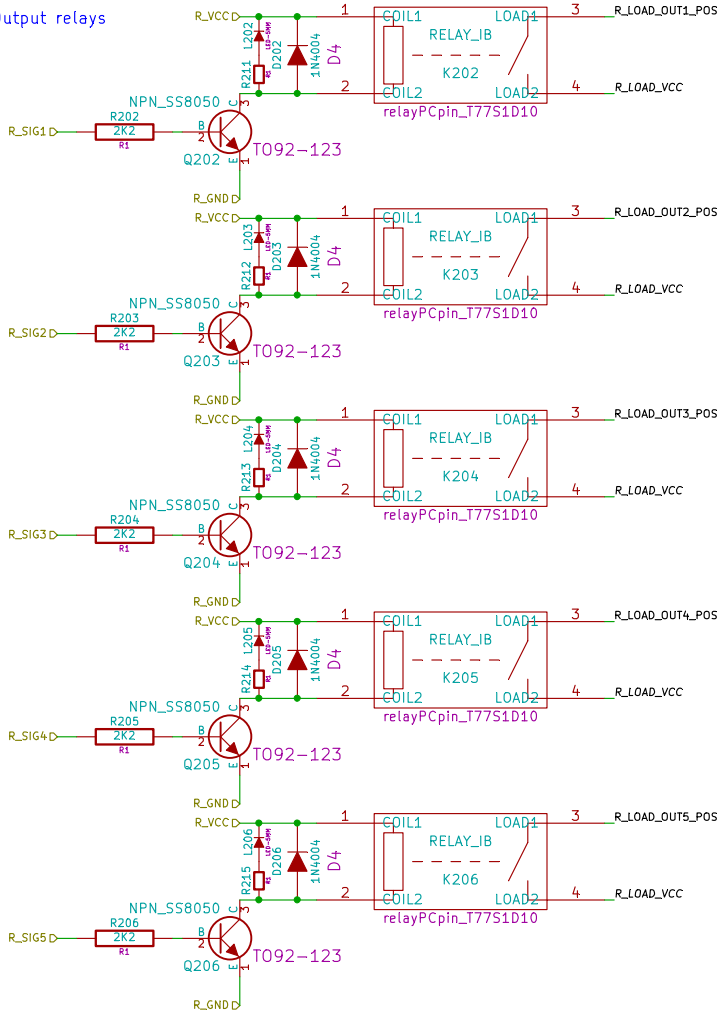
Based on schematic at http://playground.arduino.cc/uploads/Learning/Arduino_Opto_relay.pdf
Since there's only one power source, isolation may not provide much protection. Opto-isolation removed for initial revision.

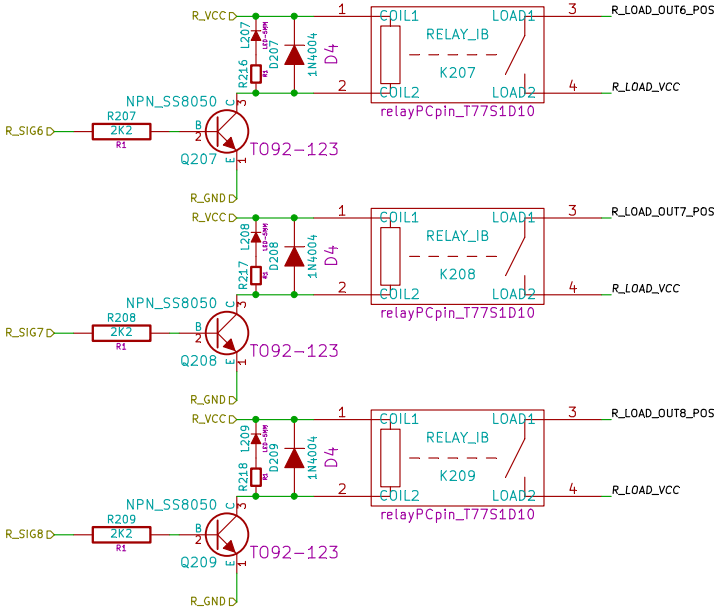
Safety relay

A NO relay enables the passage of 12V to the R_LOAD_VCC pins on each of the other relays. The relay must receive power from the microcontroller in order for any of the valves to be actuated. The mechanical system's state with no valves actuated is "Direct Drive", the failsafe state desired by the MechSE engineers. Note that a loss of power to the control system also returns the mechanical system to this failsafe state.

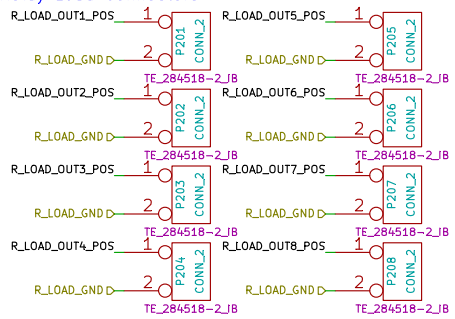


Output relays

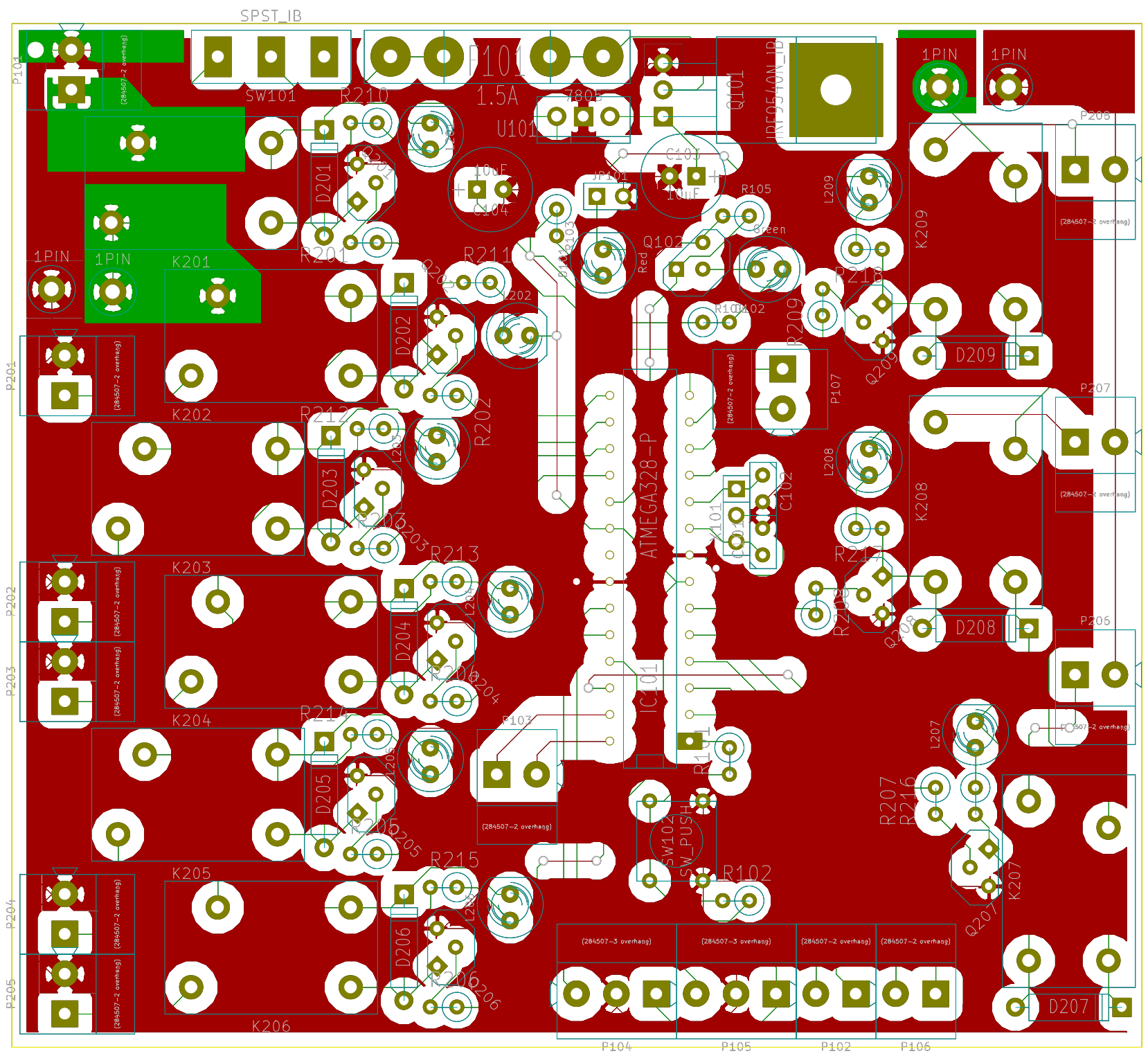




Relay Load Connectors



Senior Design Project	
ECE 445	
Iain Brearton	
File: RelayCircuit.sch	
Sheet: /Relay Circuit/	
Title: Relay Circuit	
Size: USLetter Date: 23 feb 2015	**Rev: 0.6**
KiCad E.D.A. eeschema (2013-07-07 BZR 4022)-stable	Id: 2/2

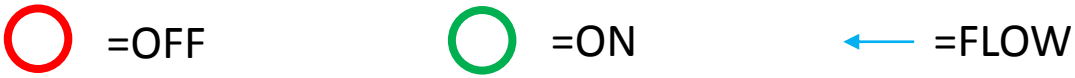


PCB trace and via sizing done using KiCAD Pcb Calculator

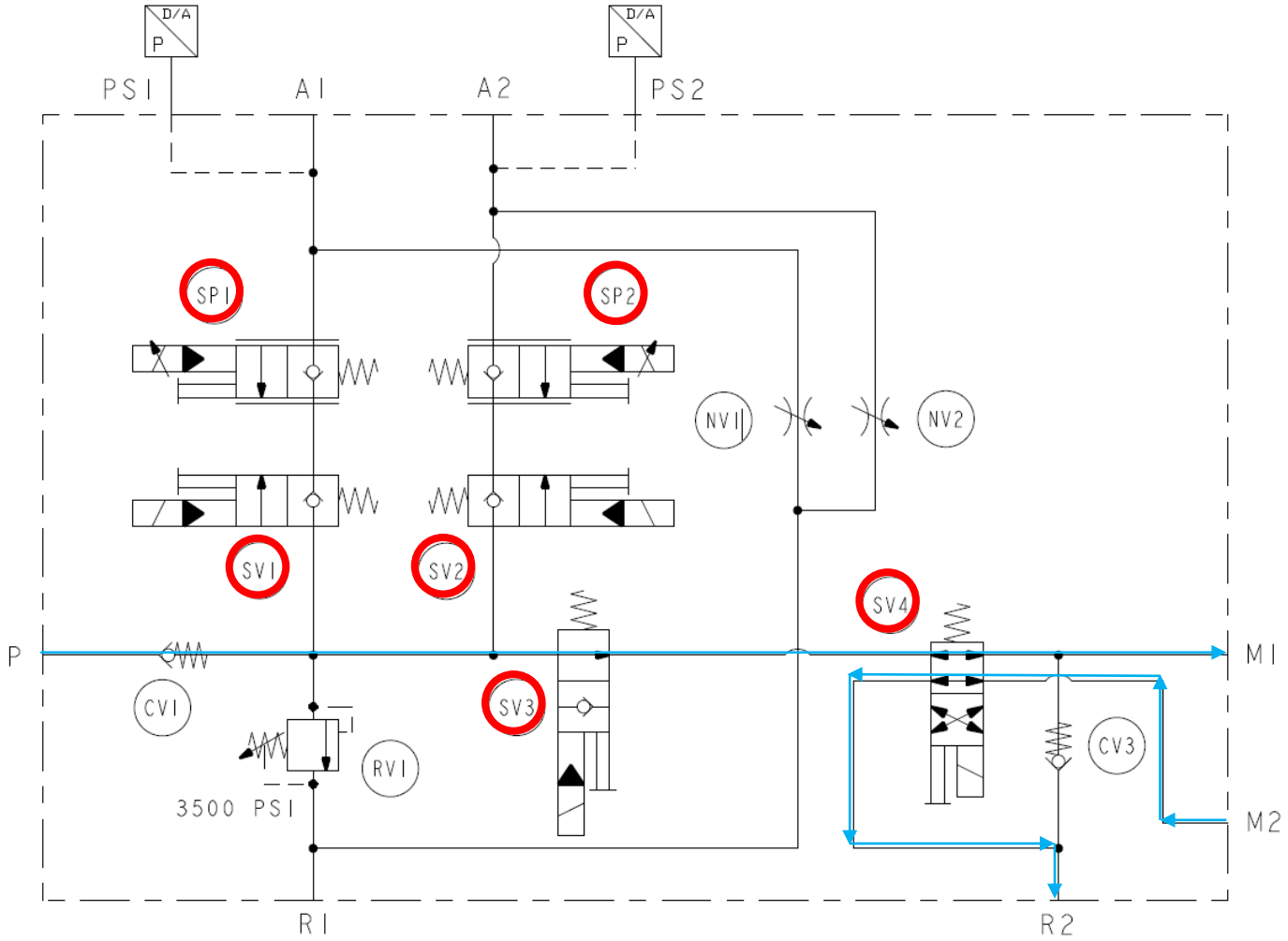
Senior Design Project	
ECE 445	
Iain Brearton	
File: ElectronicControls.kicad_pcb	
Sheet: 1/1	
Title: Electronic Controls	
Size: USLetter Date: 23 feb 2015	Rev: 0.6
KiCad E.D.A. pcbnew (2013-07-07 BZR 4022)-stable	Id: 1/1

Updated Manifold 2014-2015

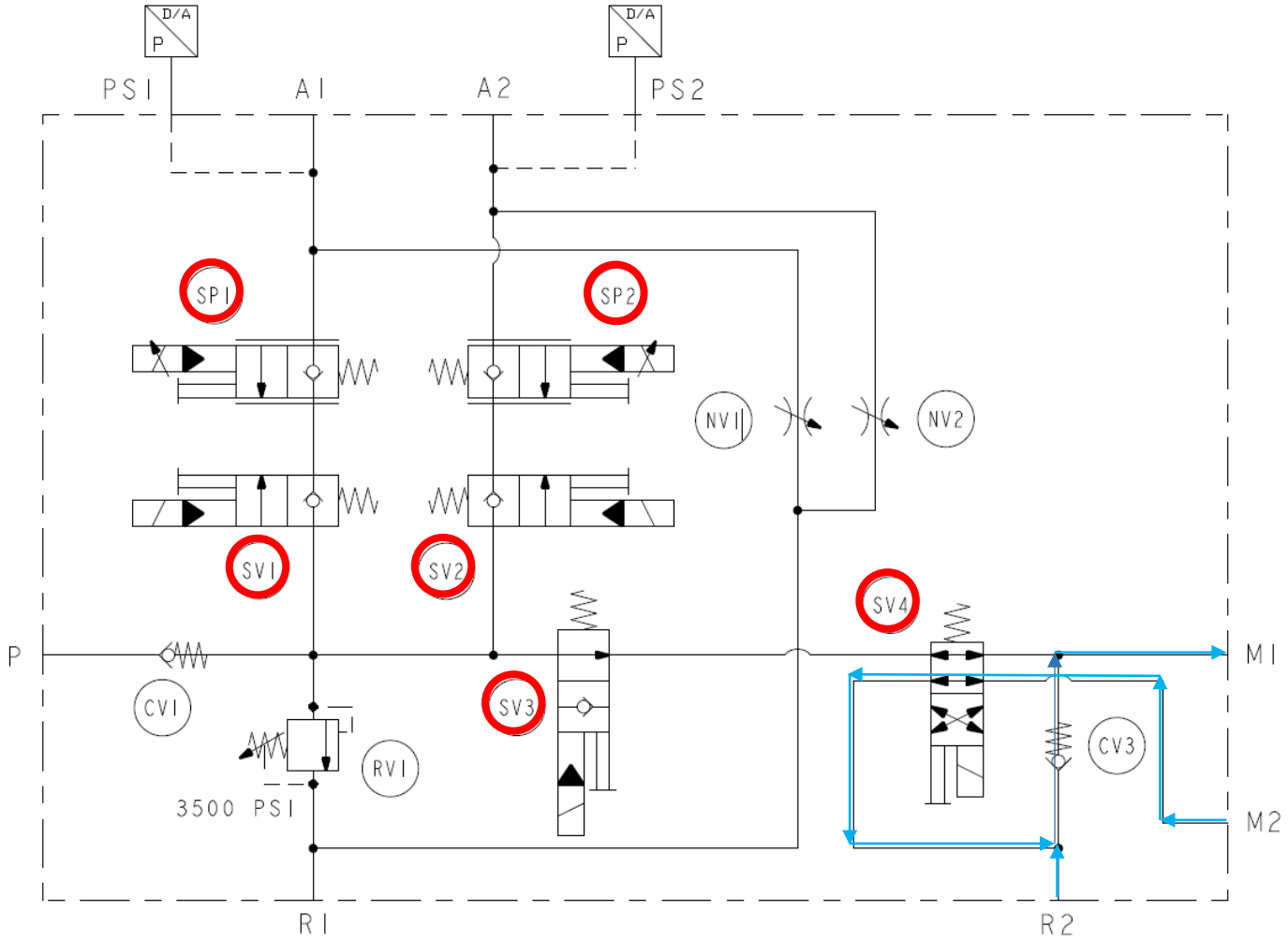
Made by Ian O'Leary for bicycle control team



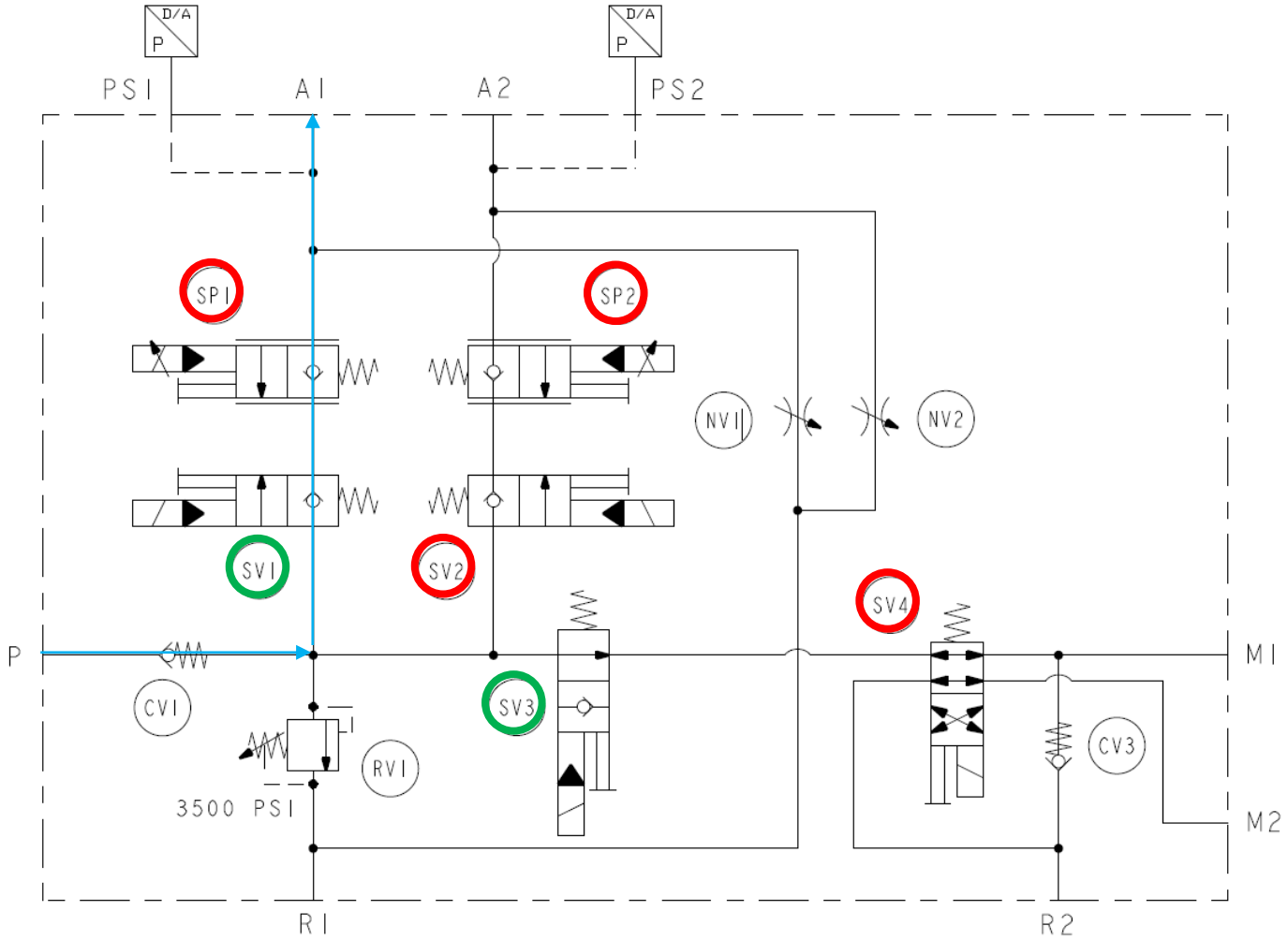
Direct Drive



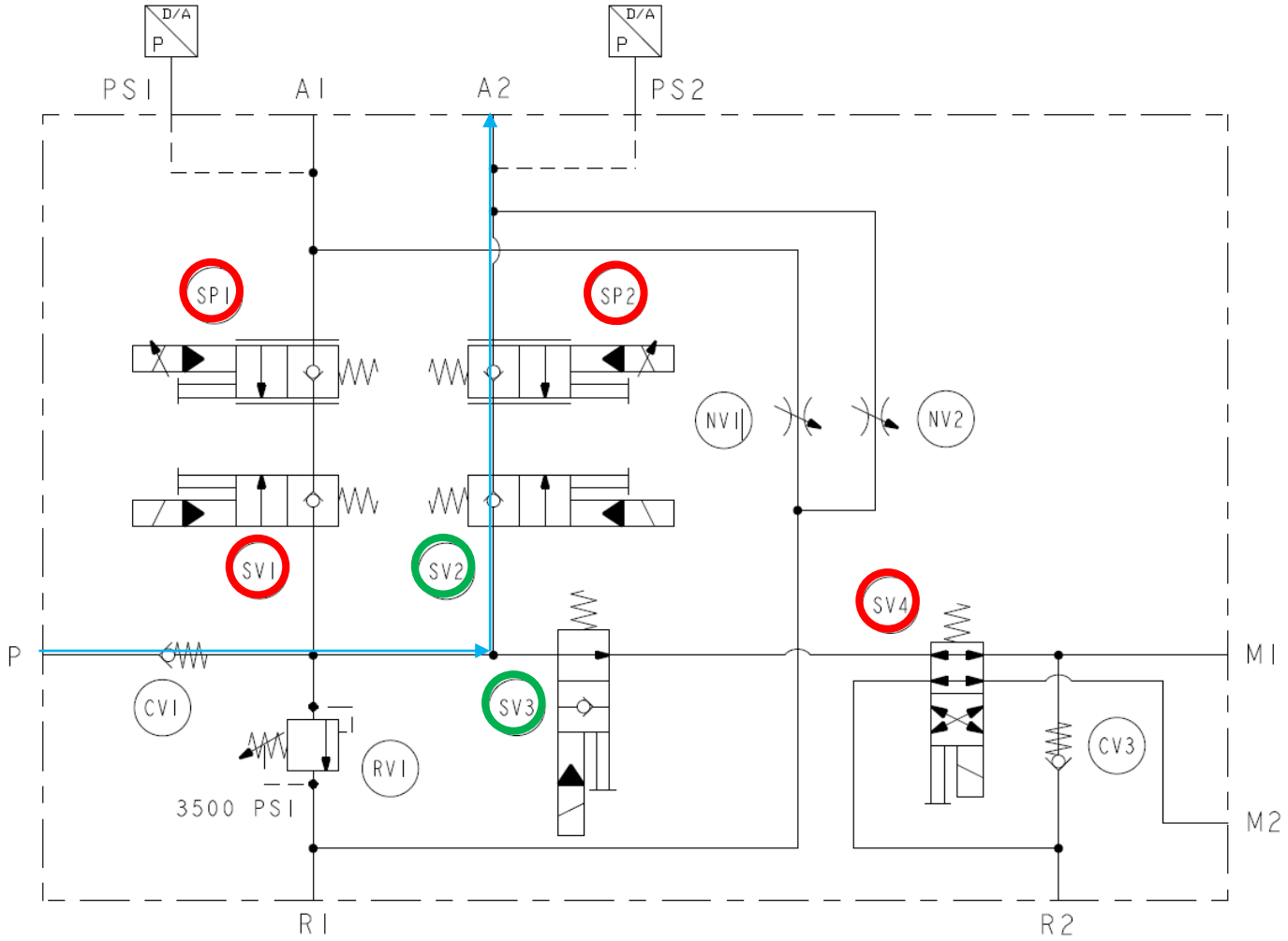
Coasting


D/A
P
PS1
A1
A2
PS2
SP1
SP2
NV1
NV2
SV1
SV2
SV4
P
M1
CV1
SV3
CV3
RV1
3500 PSI
M2
R1
R2

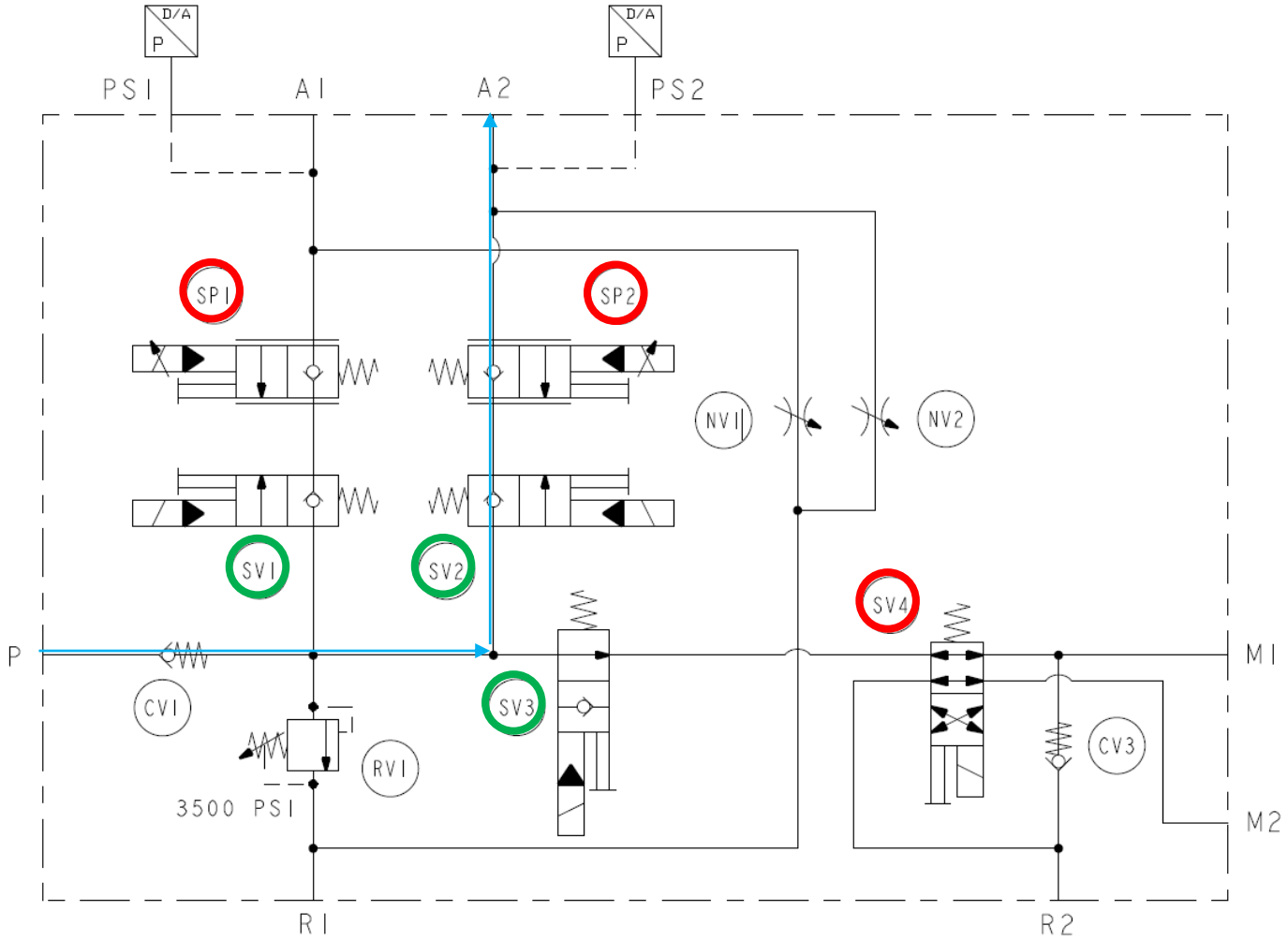
Charge A1


D/A
P
PS1
A1
A2
PS2
SP1
SP2
NV1
NV2
SV1
SV2
SV4
P
CV1
SV3
M1
RV1
CV3
3500 PSI
M2
R1
R2

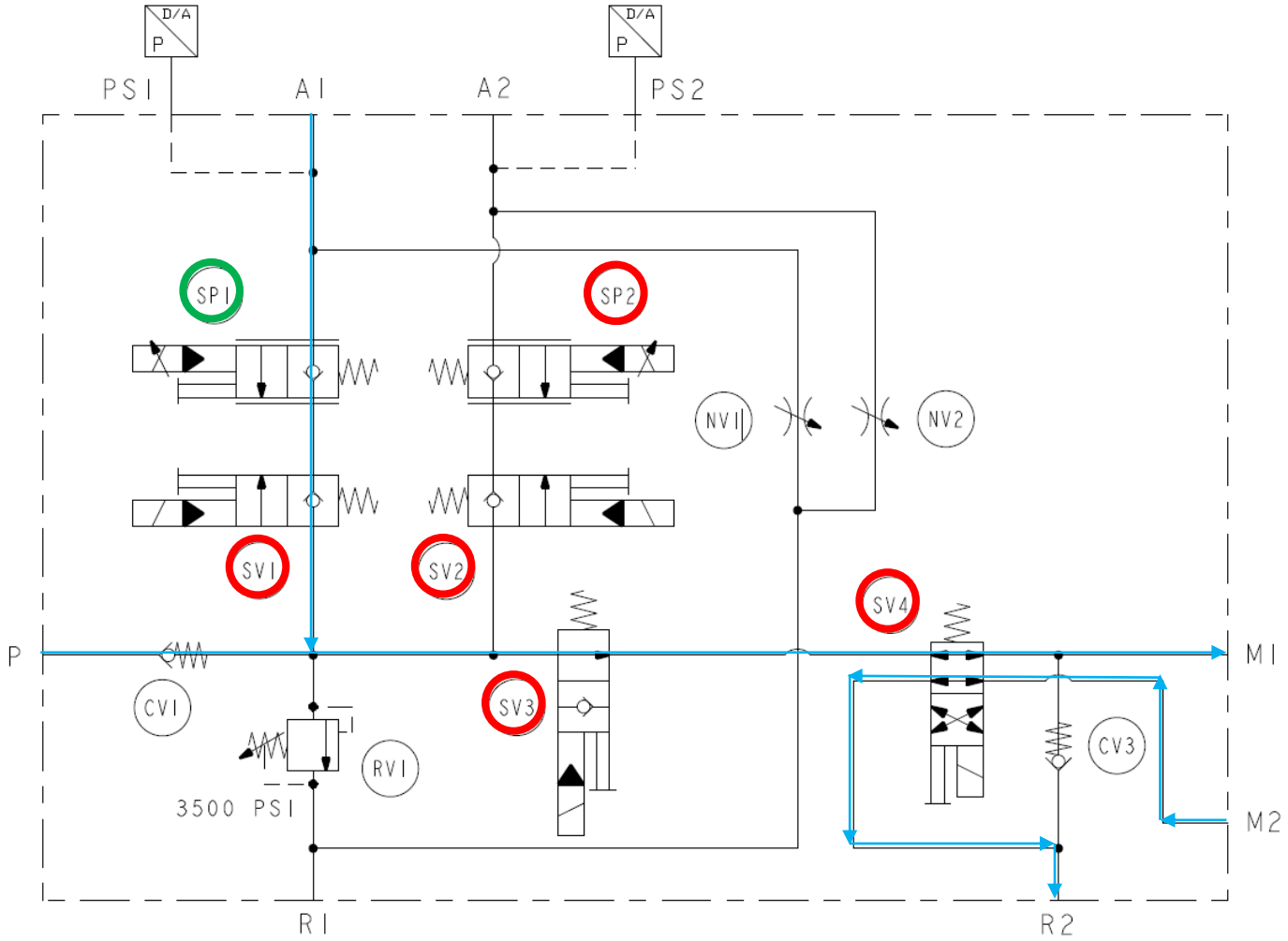
Charge A2


D/A
P
PS1
A1
A2
PS2
SP1
SP2
NV1
NV2
SV1
SV2
SV4
P
M1
CV1
SV3
CV3
RV1
3500 PSI
M2
R1
R2

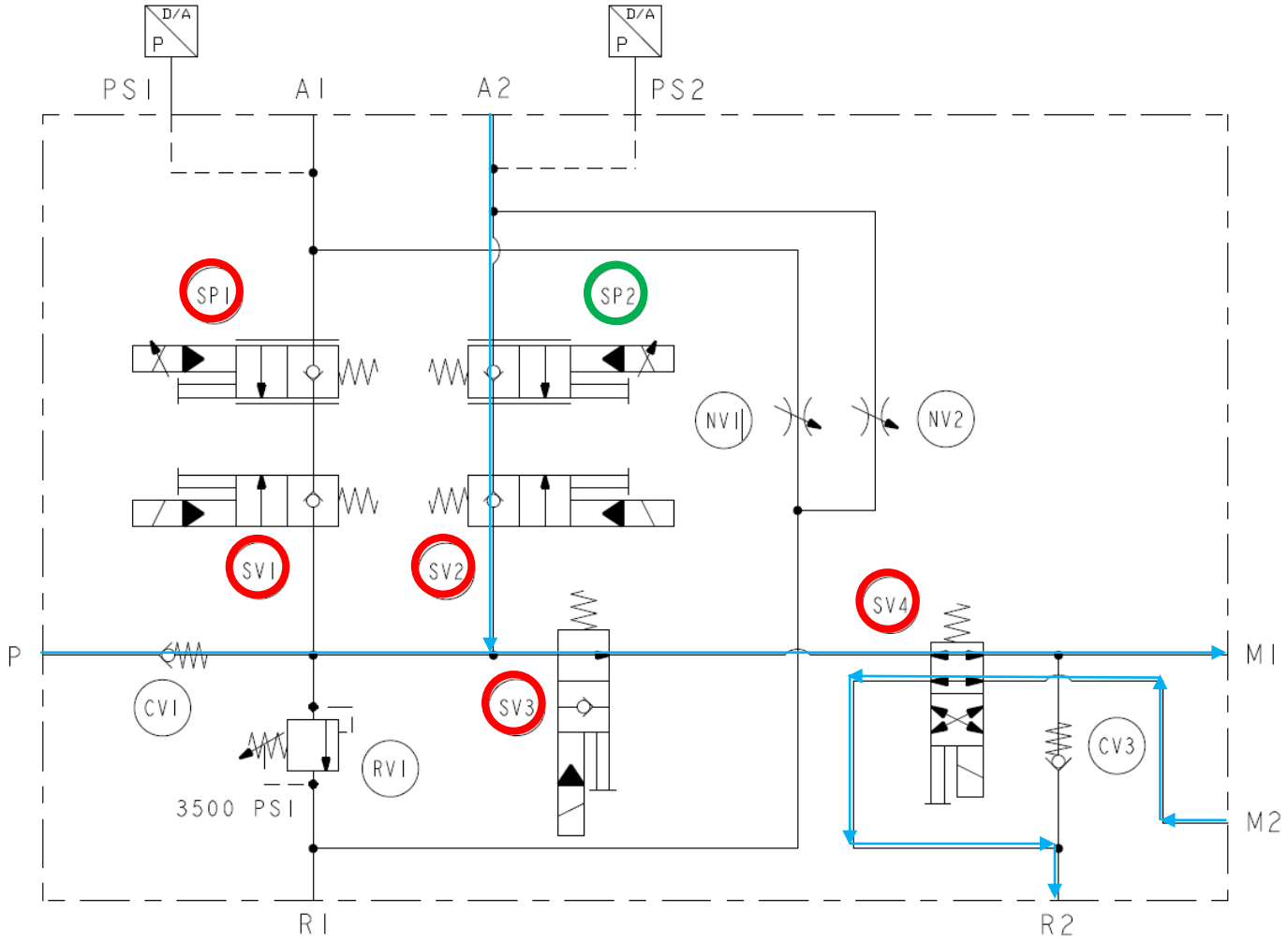
Charge A1 & A2


D/A
P
PS1
A1
A2
PS2
SP1
SP2
NV1
NV2
SV1
SV2
SV4
P
M1
CV1
SV3
RV1
CV3
3500 PSI
M2
R1
R2

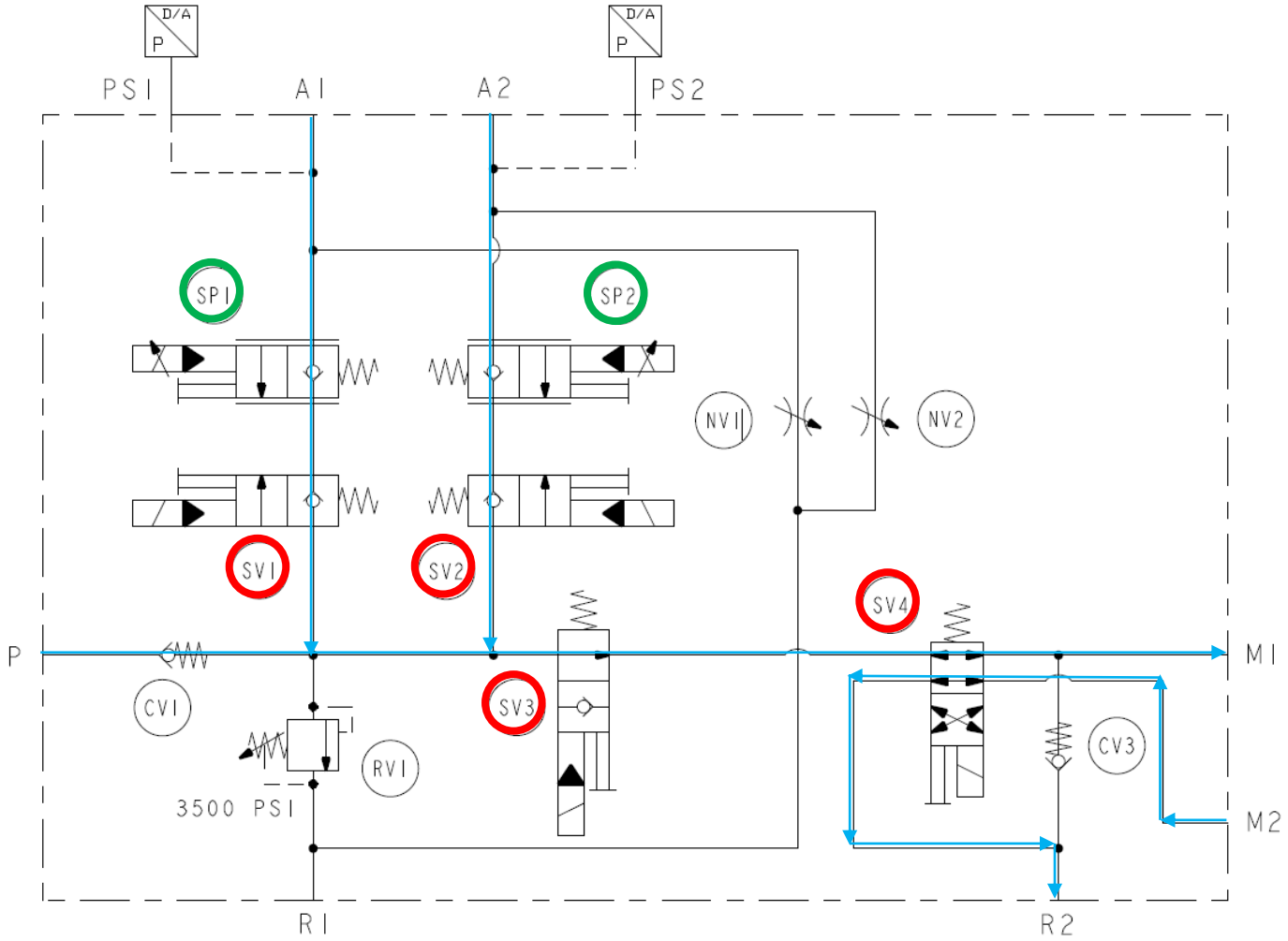
Discharge A1 (Optional Pedaling)



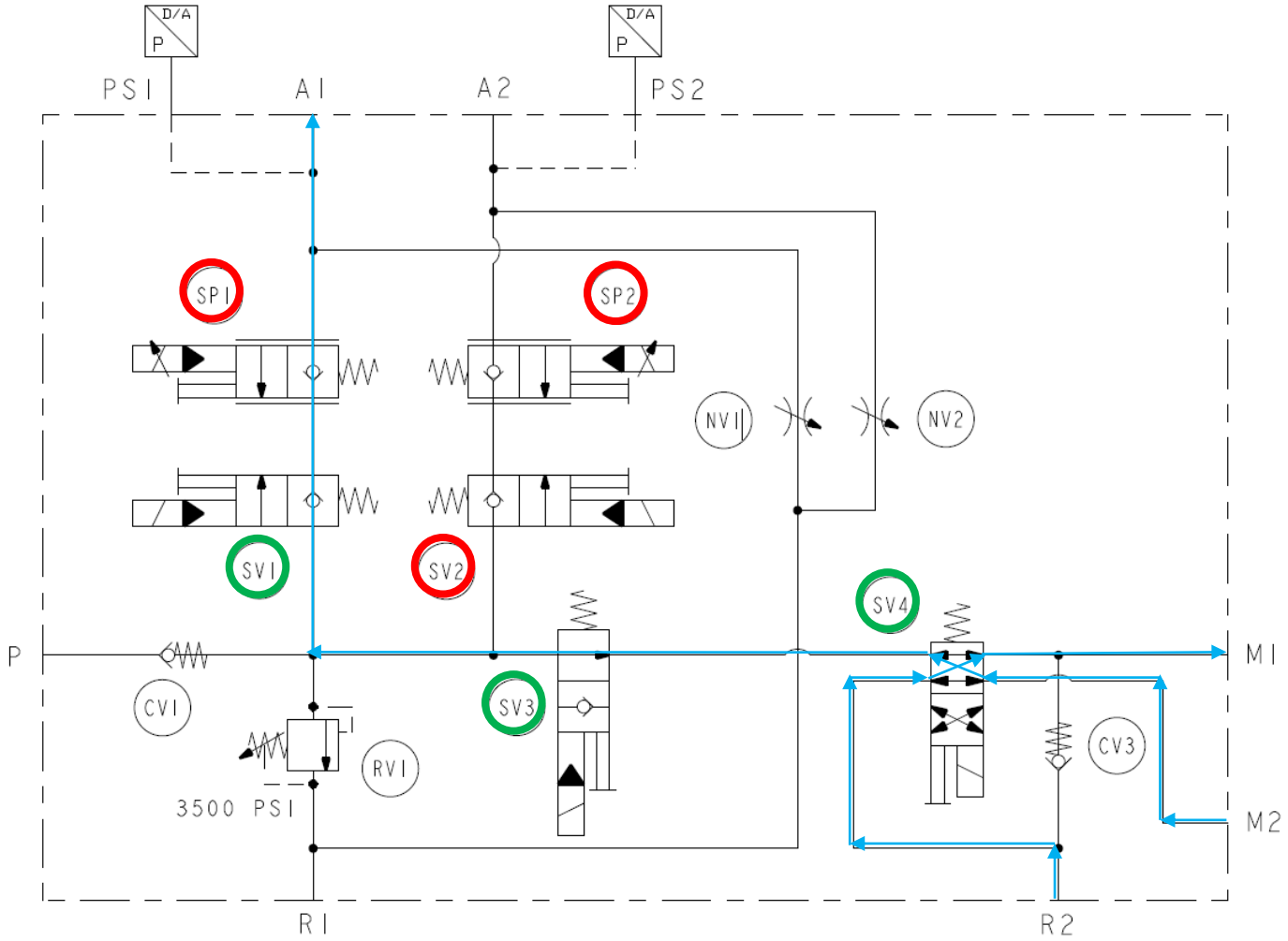
Discharge A2 (Optional Pedaling)



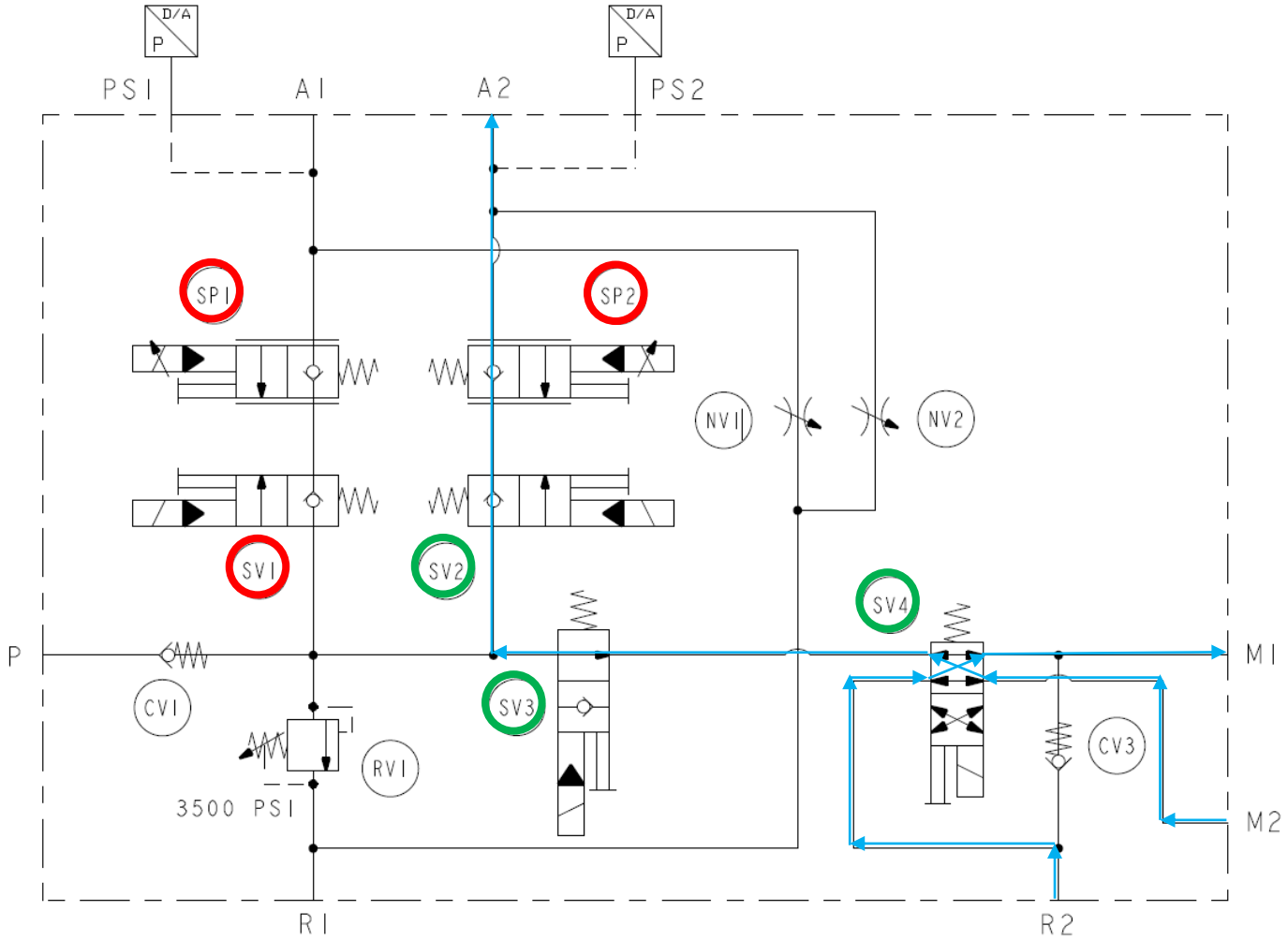
Discharge A1 & A2 (Optional Pedaling)



Regenerative Breaking A1



Regenerative Breaking A2



Regenerative Breaking A2

