FERMILAB-SLIDES-20-092-AD

Fermilab U.S. DEPARTMENT OF ENERGY Office of Science

Project 1 - R&D of Motion Table for DPA Experiment

Intern: Antonio Huanay

Mentor: Katsuya Yonehara

CCI Poster Presentation Summer 2020

This manuscript has been authored by Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the U.S. Department of Energy, Office of Science, Office of High Energy Physics.

Displacement Per Atom Experiment

- Purpose of experiment is to measure DPA cross section of target by using a 120 GeV/c proton beam at M03 experimental hall in FTBF (Fermilab Test Beam Facility)

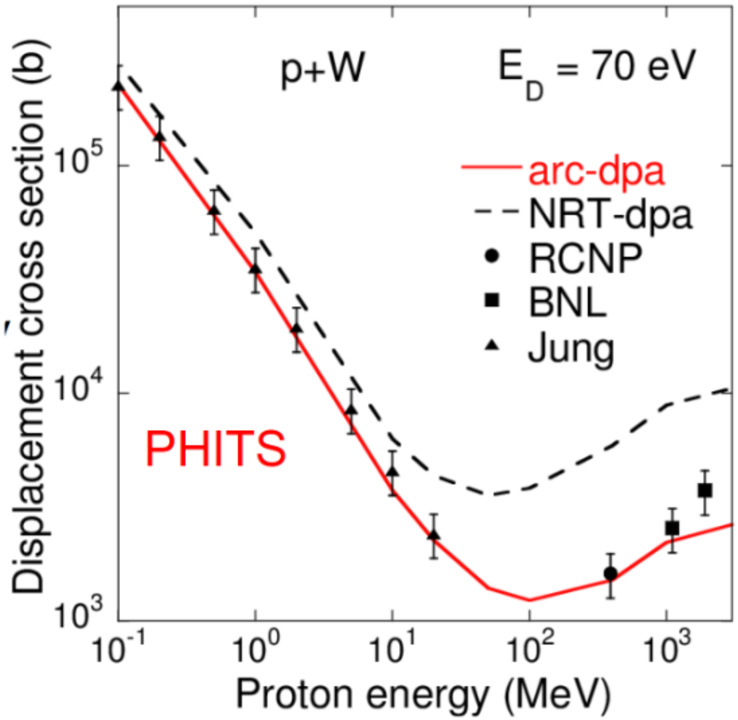


Figure 1: DPA cross section as a function of proton energy. Points are measurements. A red line is a prediction from the latest DPA model.

- To move the target in position, a motion table is needed to be placed in M03 experimental area

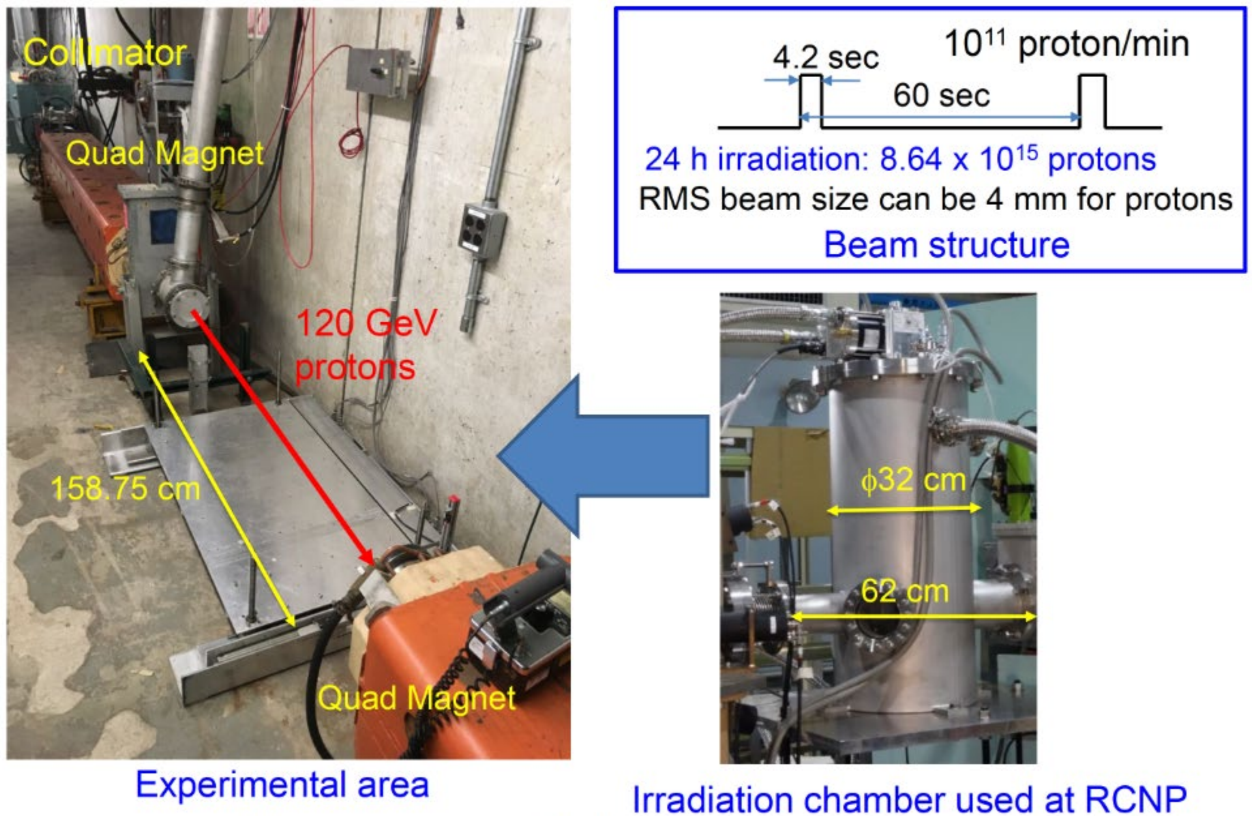


Figure 2: Picture of the M03 experimental hall and the dimension of possible target for the DPA experiment at Fermilab.

Motion Table Requirements

- Needs to hold a 100 kg target moving in horizontal direction by crossing beam line. Total stroke is 400 mm and an accuracy of 1 mm. Speed of motion table is not required. Figure 3 shows the THK motion table

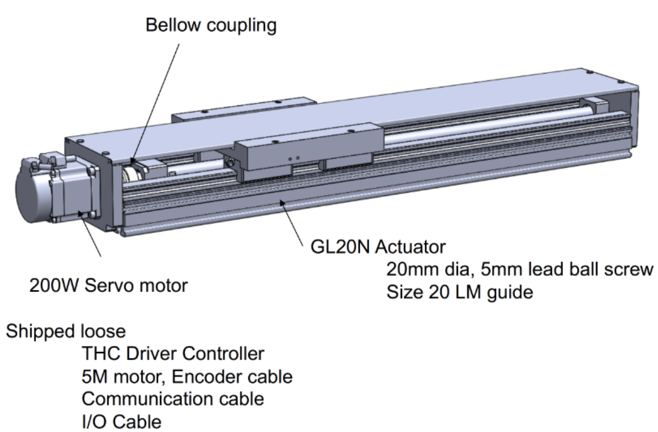
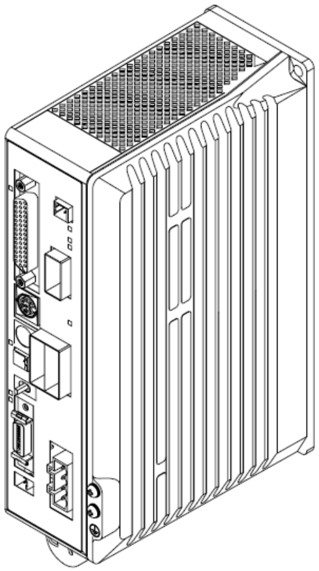


Figure 3: Purchased motion table from THK.



THC controller for the actuator

Project Mission

- The project's objective is to control the motion table remotely with the use of Python in Raspberry Pi 3B+ connected to the actuator.

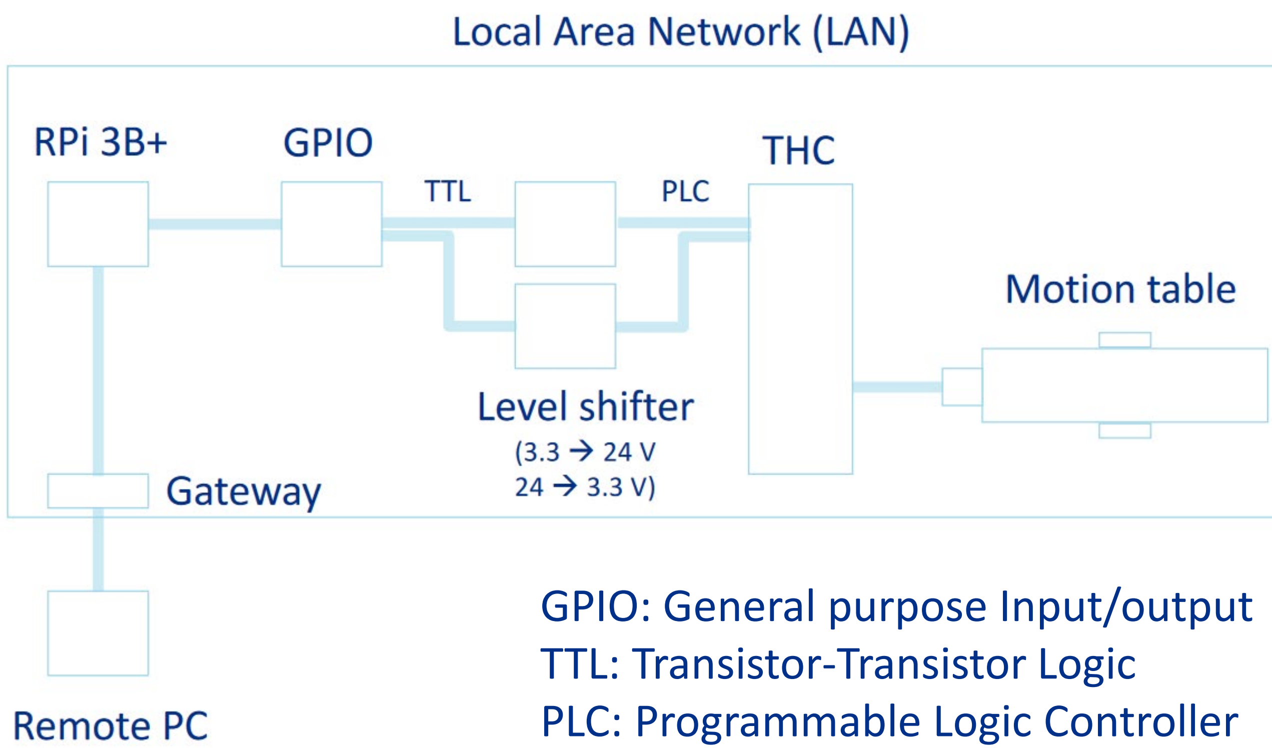
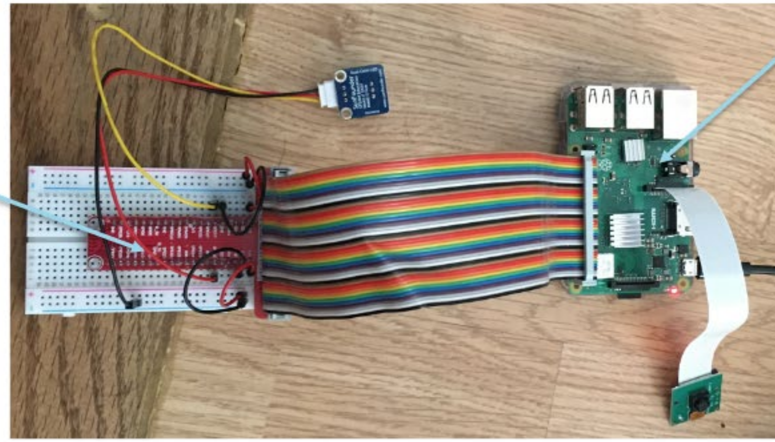


Figure 5

Fermilab

Raspberry Pi 3B+

GPIO board

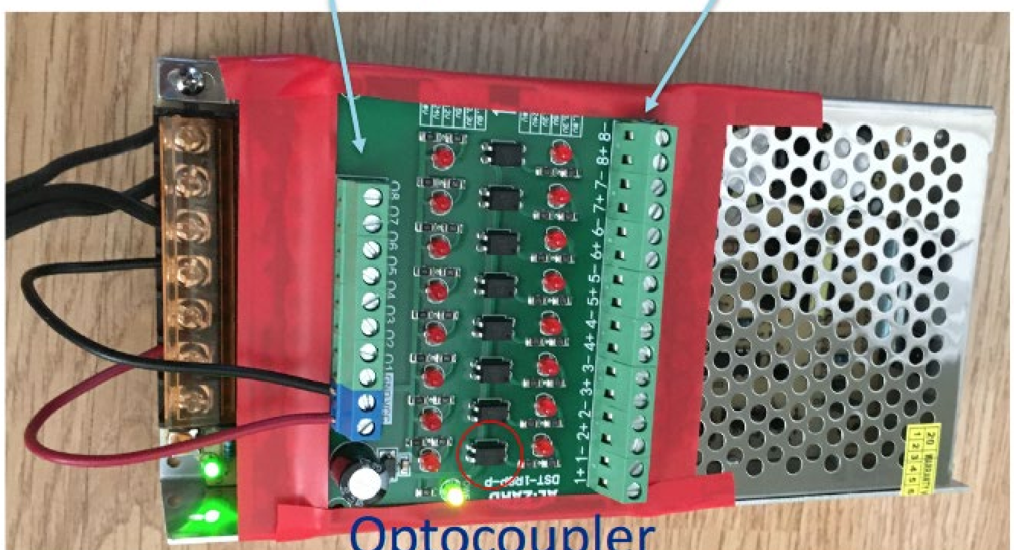


8ch PLC output

8ch TTL input

120 VAC

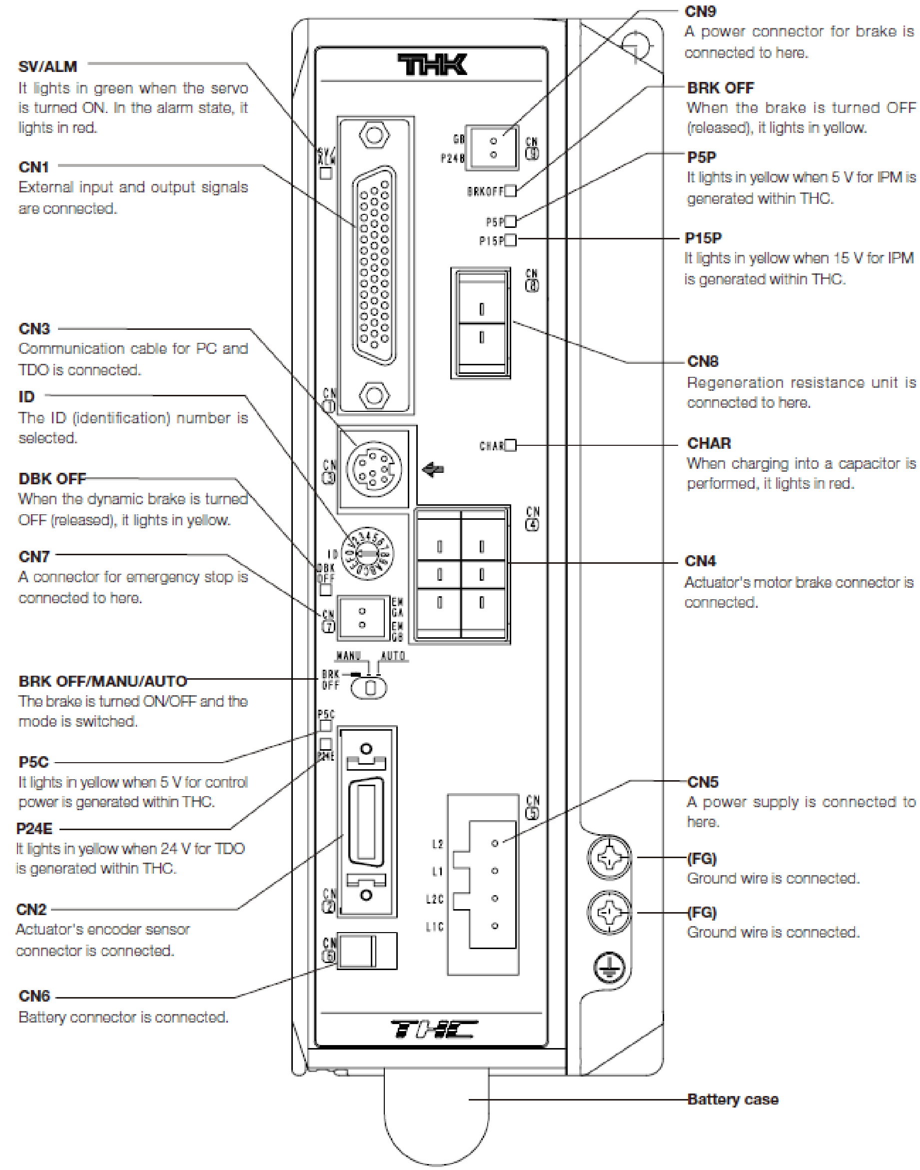
24 Vcc



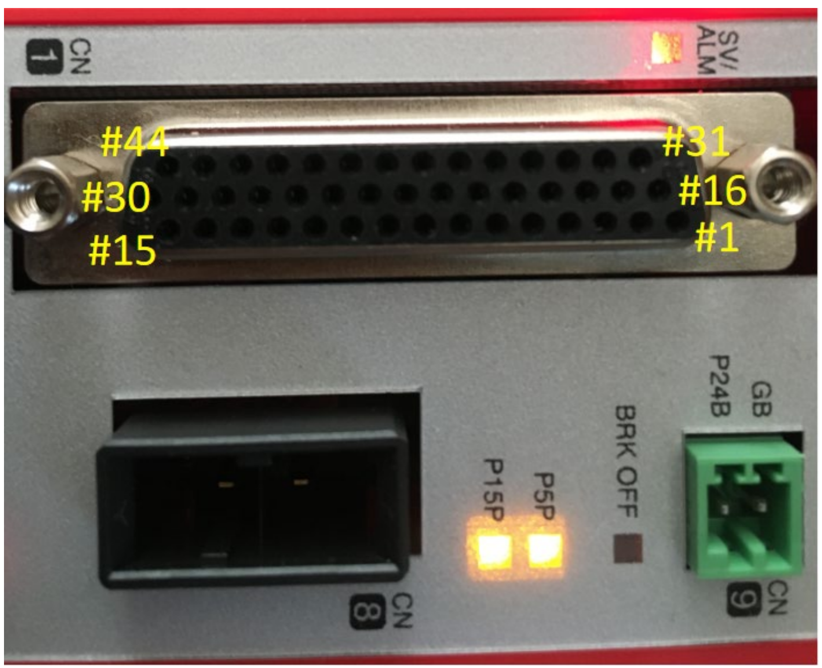
Optocoupler

Function

- By connecting CN1 to Raspberry Pi, it is possible to send signals to certain pins in the controller to move the actuator. This way it can be controlled remotely



- Each pin has a specific function. When a combinations of pins are on, the actuator will move.



Pin number	Input/ output	Function mode 1 External input instruction	
1,2	--	P24O	
3	Input	PI 0	Instruction position number
4	Input	PI 1	Instruction position number
5	Input	PI 2	Instruction position number
6	Input	PI 3	Instruction position number
7	Input	PI 4	Instruction position number
8	Input	PI 5	Instruction position number
9	Input	MODE	External input instruction mode
10	Input	JOG/INCHING	Manual switching during external input instruction
11	Input	JOG P	Moving direction + during external input instruction
12	Input	JOG N	Moving direction − during external input instruction
13	Input	STRT/PWRT	Current position write during start/external input instruction
14	Input	MANU	Operation mode
15	Input	HOME	Zero return
16	Input	PAUSE	Pause
17	Input	REST	Alarm reset
18	Input	SV-ON	Servo on
19	Output	PO 0	End position number
20	Output	PO 1	End position number
21	Output	PO 2	End position number
22	Output	PO 3	End position number
23	Output	PO 4	End position number
24	Output	PO 5	End position number
25	Output	MOVE	Moving
26	Output	MODE S	Current operation mode
27	Output	P AREA	Position area
28	Output	MANU S	Operation mode status
29	Output	HEND	Zero return completed
30	Output	INPS	Positioning completed
31	Output	WEND	Writing completed
32	Output	SVRDY	Operation preparations completed
33	Output	BALM	Voltage reduction in battery
34	Output	ALM	Alarm
35	--	---	
36	--	---	
37	--	---	
38	--	---	
39	--	---	
40	--	---	
41,42	--	FG	
43,44	--	GO	
case		FG	

Fermilab

Using Oscilloscope to verify signals from output-code

- In order to know if the program works and sends a signal through Raspberry Pi, an oscilloscope was used to verify the output of Rpi

*Actuator_code_2.py - /home/pi/Actuator_code_2

File Edit Format Run Options Window Help

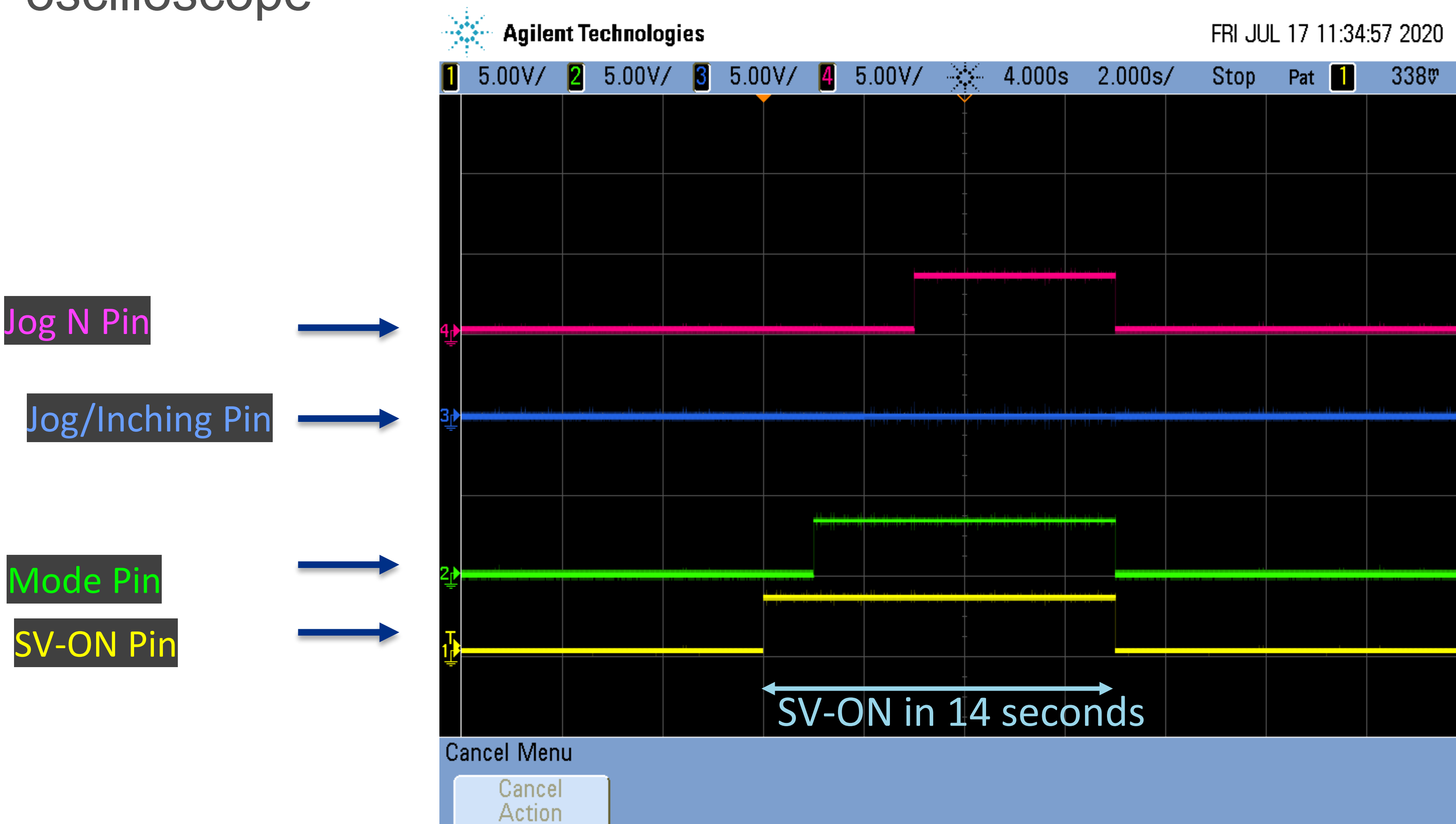
```python
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)

#Name - GPIO - THC Pin: Input List

#SV-ON - 4 - 18     used
#Mode - 19 - 9 used
#Mode S - 16 - 26
#Zero return - 17 - 15
#Jog/Inching - 21 - 10  used
#JOG N - 26 - 12   used
#JOG P - 12 - 11 used

#Turn Servo On
sv = 4
GPIO.setup(sv,GPIO.OUT)
GPIO.output(sv,True)

#Waiting time 10ms
time.sleep(0.01)
```

*Actuator_code_2.py - /home/pi/Actuator_code_2.py (3.5

File Edit Format Run Options Window Help

```python
time.sleep(0.01)
mode = 19
GPIO.setup(mode,GPIO.OUT)
GPIO.output(mode, True)

time.sleep(0.01)

#Turn JOG/Inch OFF
ji = 21
GPIO.setup(ji, GPIO.OUT, initial = GPIO.LOW)

#Waiting time 10ms
time.sleep(0.01)

#Turn JOG N ON (Default speed is 10 mm or 1cm per second)
jn = 26
GPIO.setup(jn, GPIO.OUT)
GPIO.output(jn, True)

#Waiting time
time.sleep(15)

#Turn JOG N OFF
GPIO.output(jn,False)
```

- What this code does is to send instructions to the controller as signals, without receiving any input from the controller yet.
- The picture below are the signals from the code read by oscilloscope



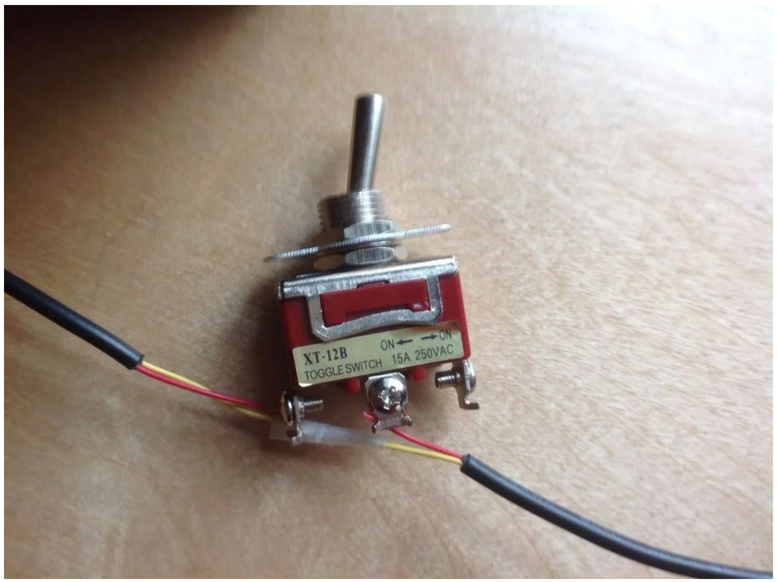
Output-Input code

- This code reads input from controller from a specific pin and determines whether it is on or off. This has been tested with a pulse generator connected to raspberry pi simulating the controller output

```python
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)

modes = 16
GPIO.setup(modes, GPIO.IN)

time.sleep(0.1)

from datetime import datetime
t1 = datetime.now()
while (datetime.now()-t1).seconds <= 20:
    while GPIO.input(modes) != 0:
        print("Signal on")
```

```python
from datetime import datetime
t1 = datetime.now()
while (datetime.now()-t1).seconds <= 20:
    while GPIO.input(modes) != 0:
        print("Signal on")

    else:
        while GPIO.input(modes) != 1:
            print("signal off")
    print(datetime.now())

GPIO.cleanup()
```

- At the time of testing, channels of level shifter did not work, so 8 manual switches and 2 relays were added to the circuit to replace channels of level shifter.



Manual Switch

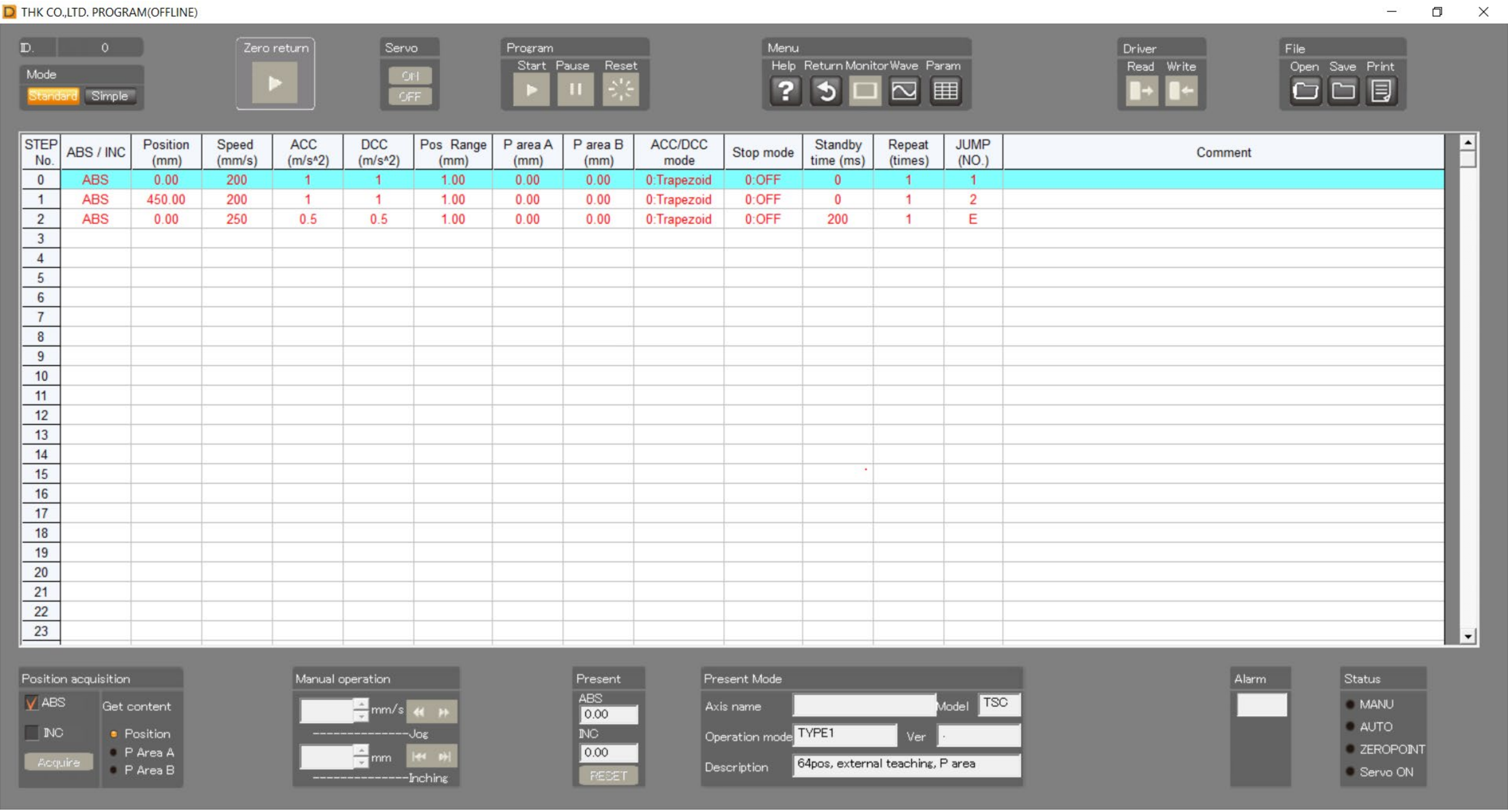
Relay switch controlled remotely

Test

- At testing with different switches, we turned on manually pins SV-ON, MODE, performed Zero-return, PI 0, and Start. In addition we turned off Jog/inching and sent signals to Jog P and Jog N to move the actuator. In the end, this did not work.

- Then we tried another method. We controlled manually only pins START and PAUSE. The programming was through D-STEP, and we were able to move the actuator

D-STEP



Fermilab

Actuator moving

What could be improved in the future?

- Electronics improvement to make the controller read signals
- Testing in laboratory

Project 2 - Making a power supply that produces 1000 A with ripple of 50 parts per million.

Intern: Antonio Huanay

Mentor: Chris Jensen

CCI for Summer 2020

Project's Objective

- Purpose of project is to show what power supply type should be used to achieve 50 parts per million (ppm) of current ripple
  - Current ripple is the amount of change in the current over time when it should be stable
- This is necessary to create a stable magnetic field so that target particle in beam stays in the correct position
- Six simulations of power supplies were done: Two and three phase bridge power supplies with various filters
  - Filters are components added to reduce the amount of current at higher frequencies while keeping the steady state current the same
- All simulations were done in LTSpice XVII

Praeg Filter

- This circuit is one example of a filter. It is commonly used in high current power supplies.
- The efficiency in reducing current ripple will be shown.

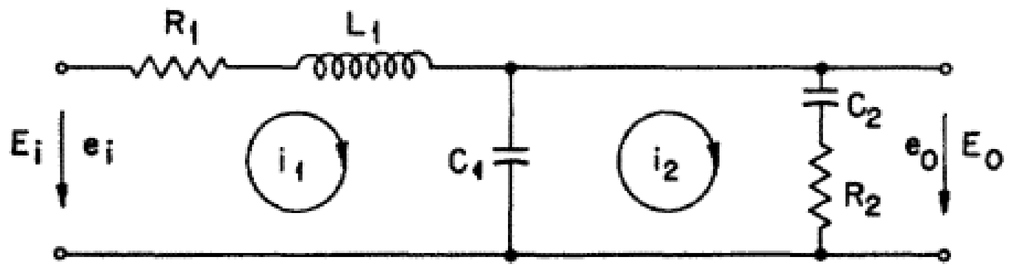


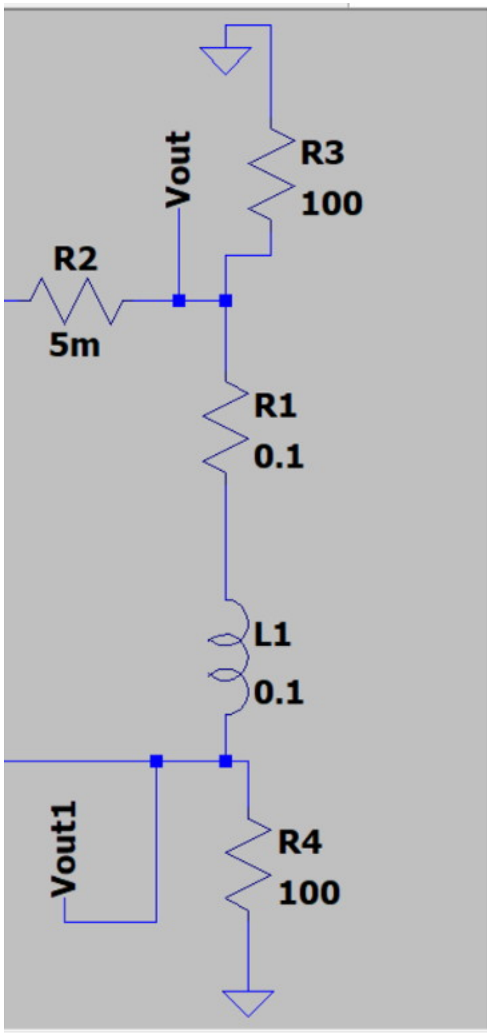
Fig. 3. Improved *LCR* low-pass filter.

From paper: A High-Current Low-Pass Filter for Magnet Power Supplies by Walter F. Praeg, 1970

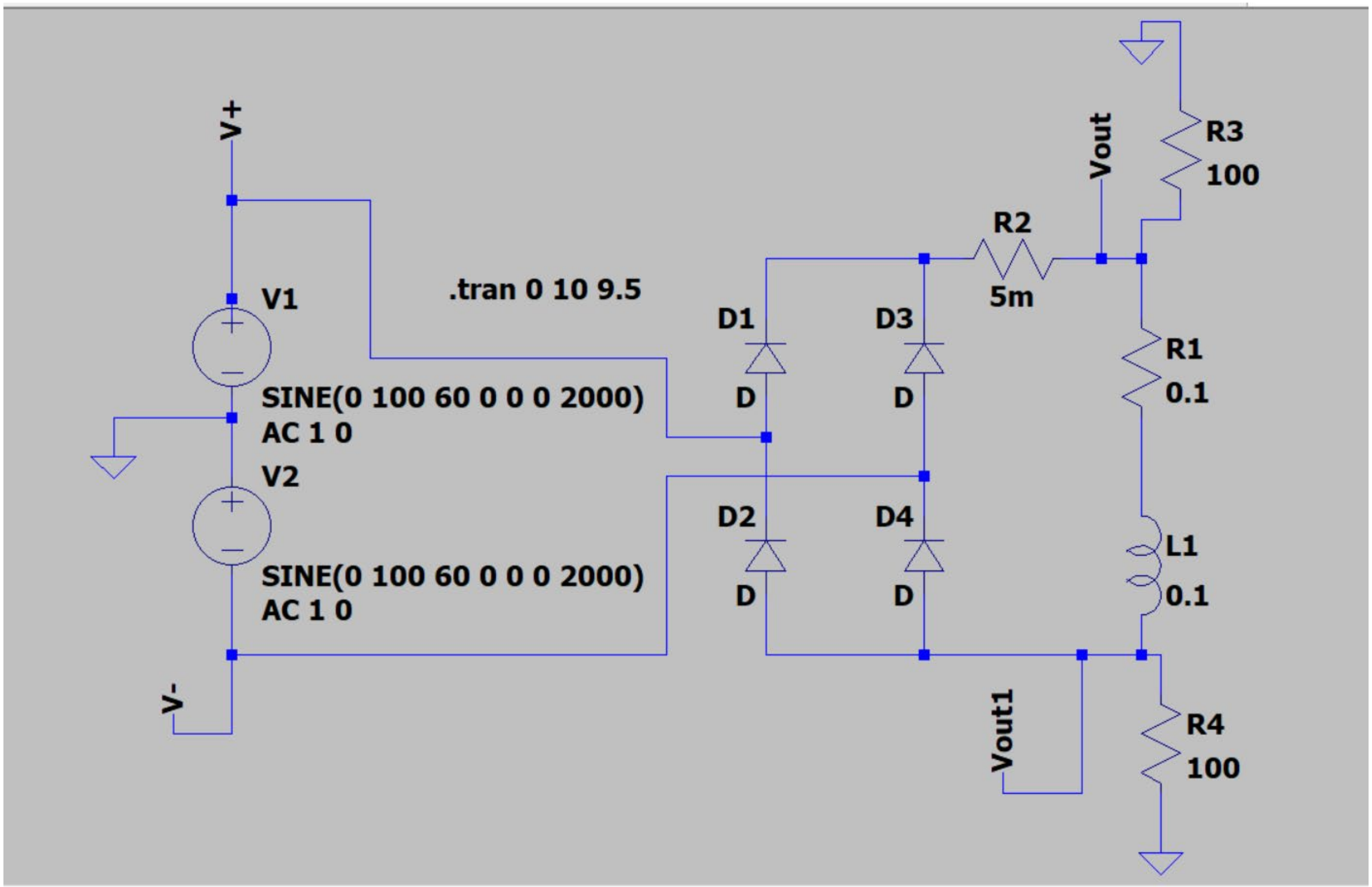
General Description of Magnet Load

In the picture:

- R1 and L1 represent a magnet load
- R2 represents basic resistance of wiring
- R4 and R3 are balancing resistors so that voltage on the magnet is balanced about ground.
- Vout and Vout1 were created to measure difference in voltage

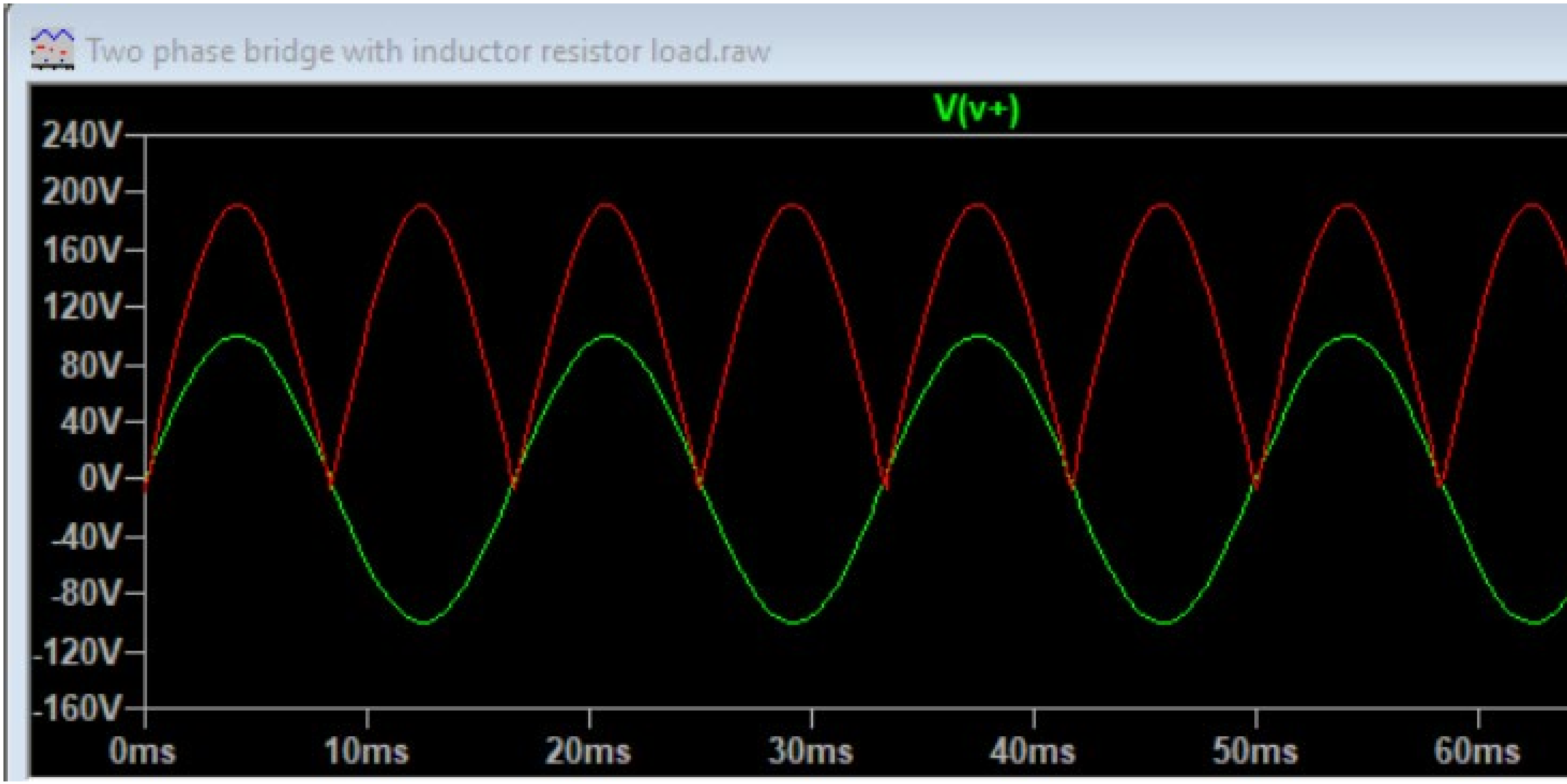


Two Phase Bridge Power Supplies with Load only (Resistor and Inductor)



Simulated Voltages

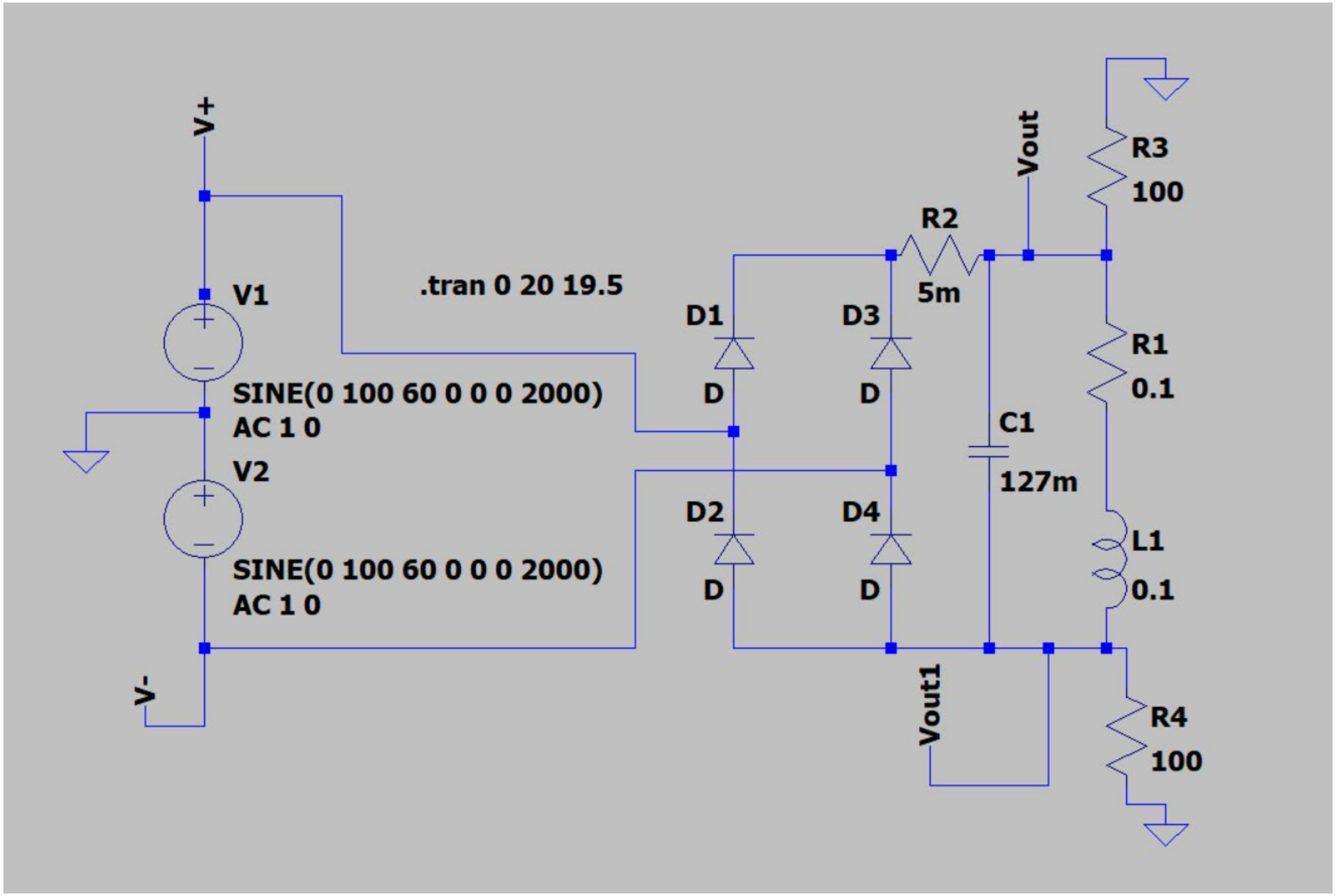
- Red line represents output voltage V(out) – V(out1)
- Green line represents input voltage V(V+)



Two Phase Bridge Power Supplies with Load only (Resistor and Inductor) - Measurements

	2 phase bridge with load only (R & L)
Current Percentage Ripple	0.16 % (1600 ppm)
Peak to Peak current	1.94 A
Average Current	1.170 kA
Voltage Ripple [V(out) – V(vout1)]	197.5 V
Avg Volt [V(out) – V(vout1)]	116.33 V

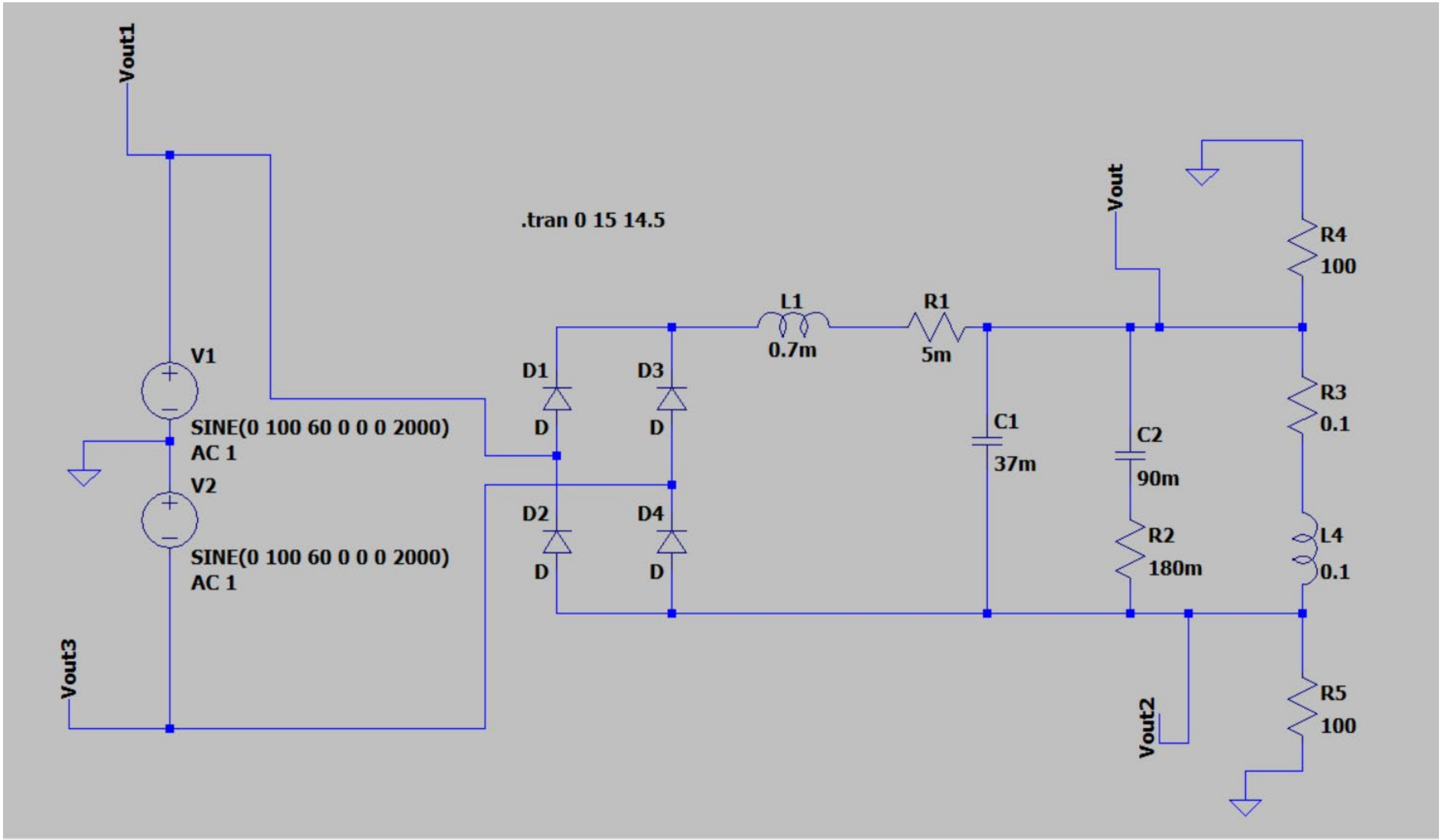
Two Phase Bridge Power Supply with Resistor, Capacitor, and Inductor



Two Phase Bridge Power Supply with Resistor, Capacitor, and Inductor - Measurements

	2 phase Bridge with R, C and L C = 127mF
Current Percentage Ripple	0.041 % (410 ppm)
Peak to Peak current	641.47 mA
Average Current	1.55 kA
Voltage Ripple [V(out) – V(vout1)]	56 V
Avg Volt [V(out) – V(vout1)]	156.67 V

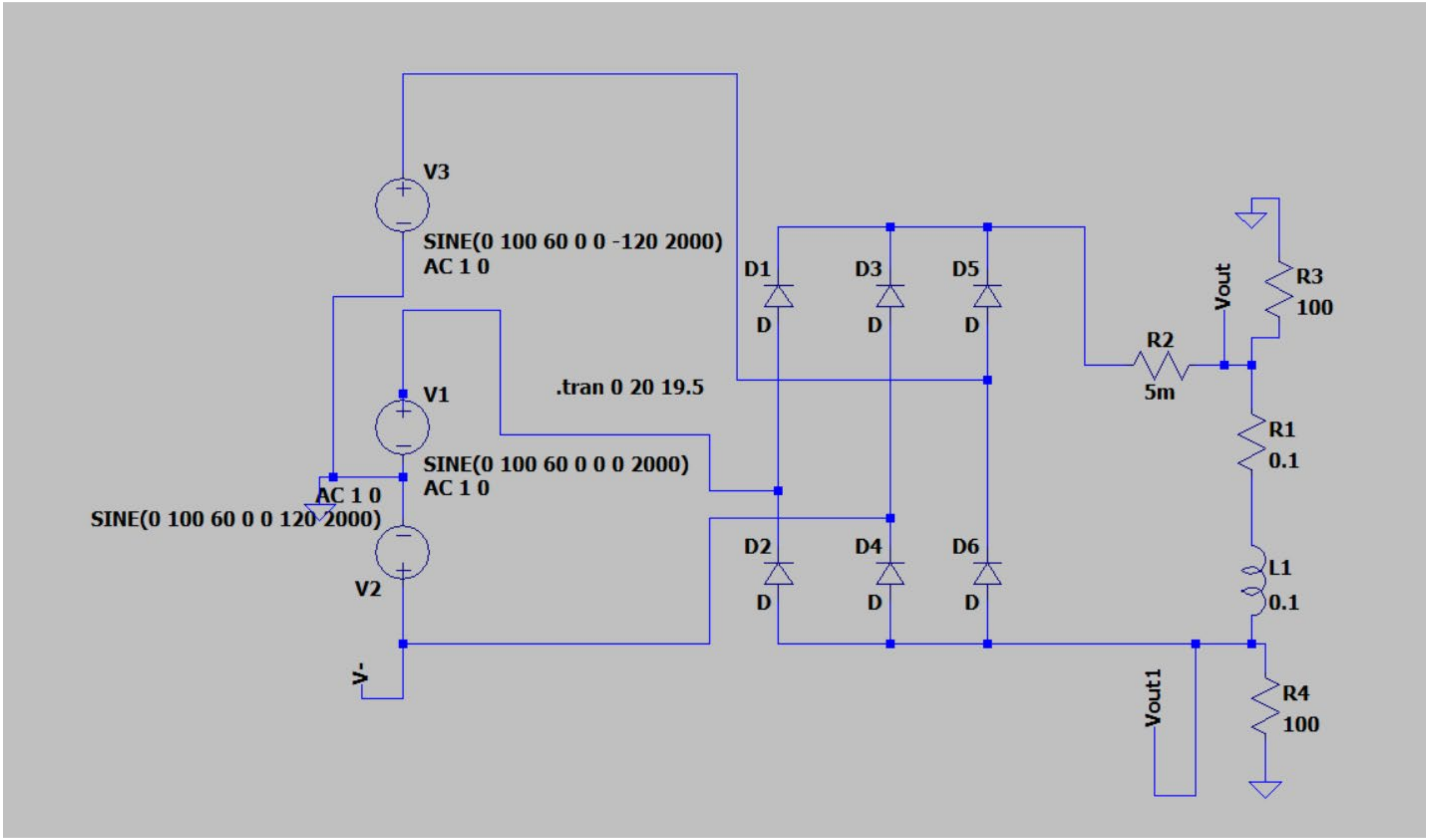
Two Phase Bridge Power Supply with Praeg Filter



Two Phase Bridge Power Supply with Praeg Filter - Measurements

	2 phase Bridge with Praeg filter C = 127mF
Current Percentage Ripple	0.011 % (110 ppm)
Peak to Peak current	123.24 mA
Average Current	1.1602 kA
Voltage Ripple [V(out) – V(vout1)]	11 V
Avg Volt [V(out) – V(vout1)]	117.19

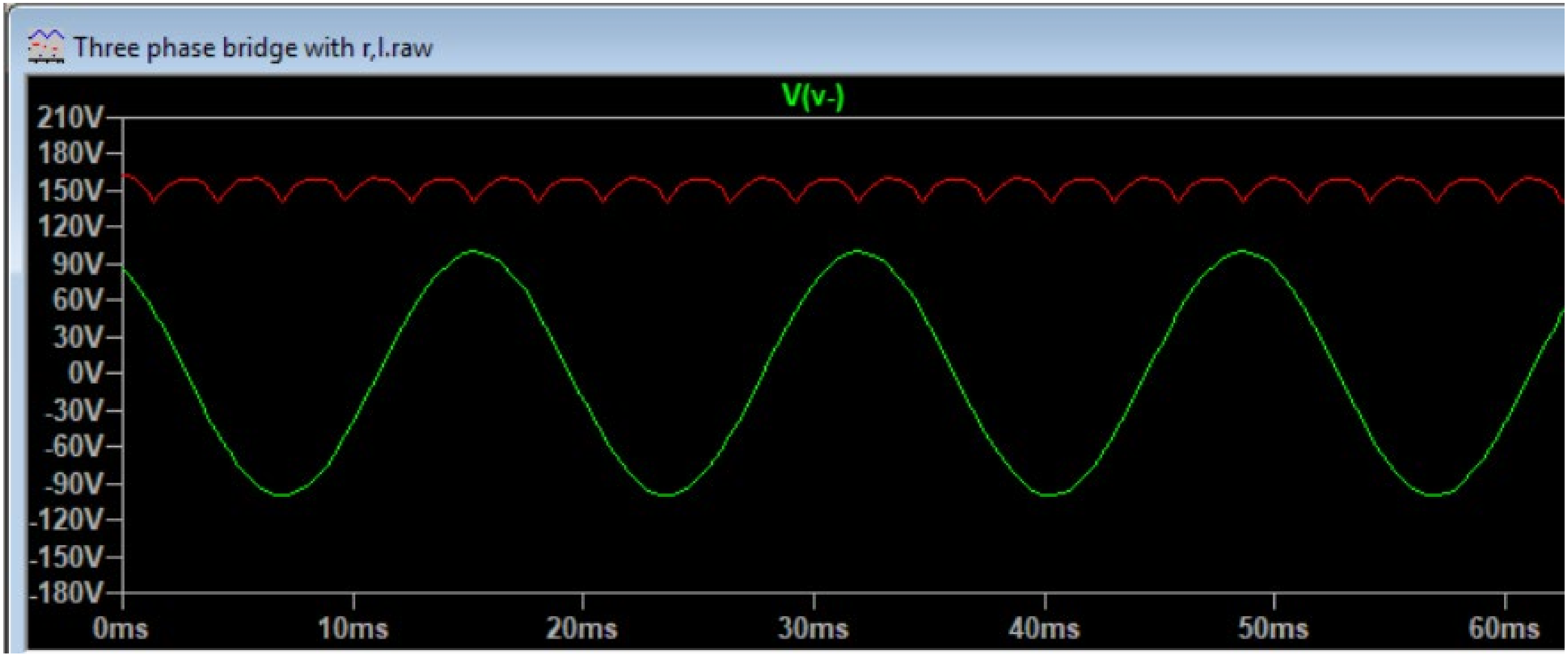
Three Phase Bridge Power Supply with Load only (Resistor and Inductor)



Fermilab

Simulated Voltages

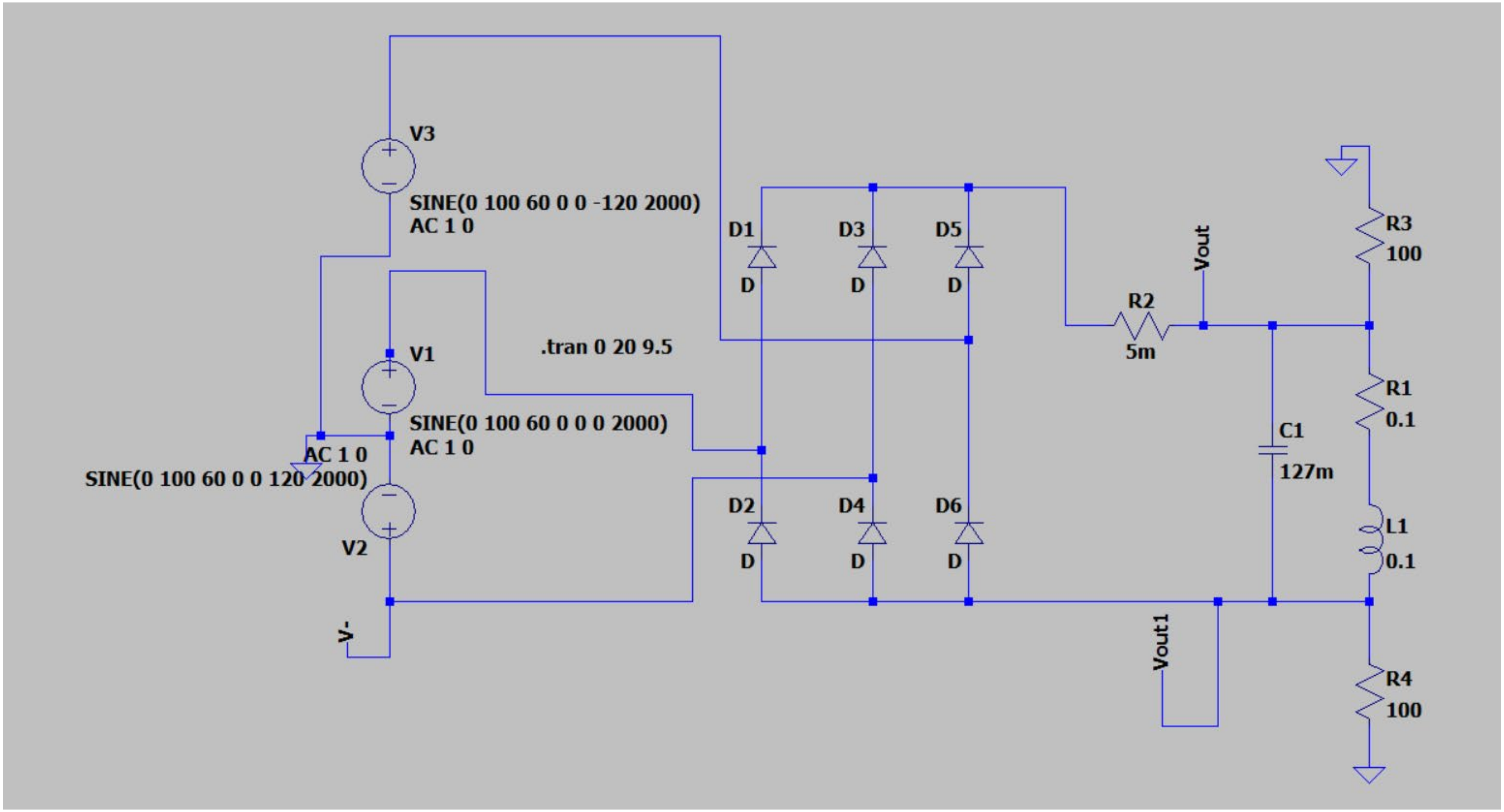
- Red line represents output voltage V(out) – V(out1)
- Green line represents input voltage V(V+)



Three Phase Bridge Power Supply with Load only (Resistor and Inductor) - Measurements

	3 phase bridge with R & L
Current Percentage Ripple	0.0044% (40 ppm)
Peak to Peak current	67.60 mA
Average Current	1.5361 kA
Voltage Ripple [V(out) – V(vout1)]	19.3 V
Avg Volt [V(out) – V(vout1)]	154.16

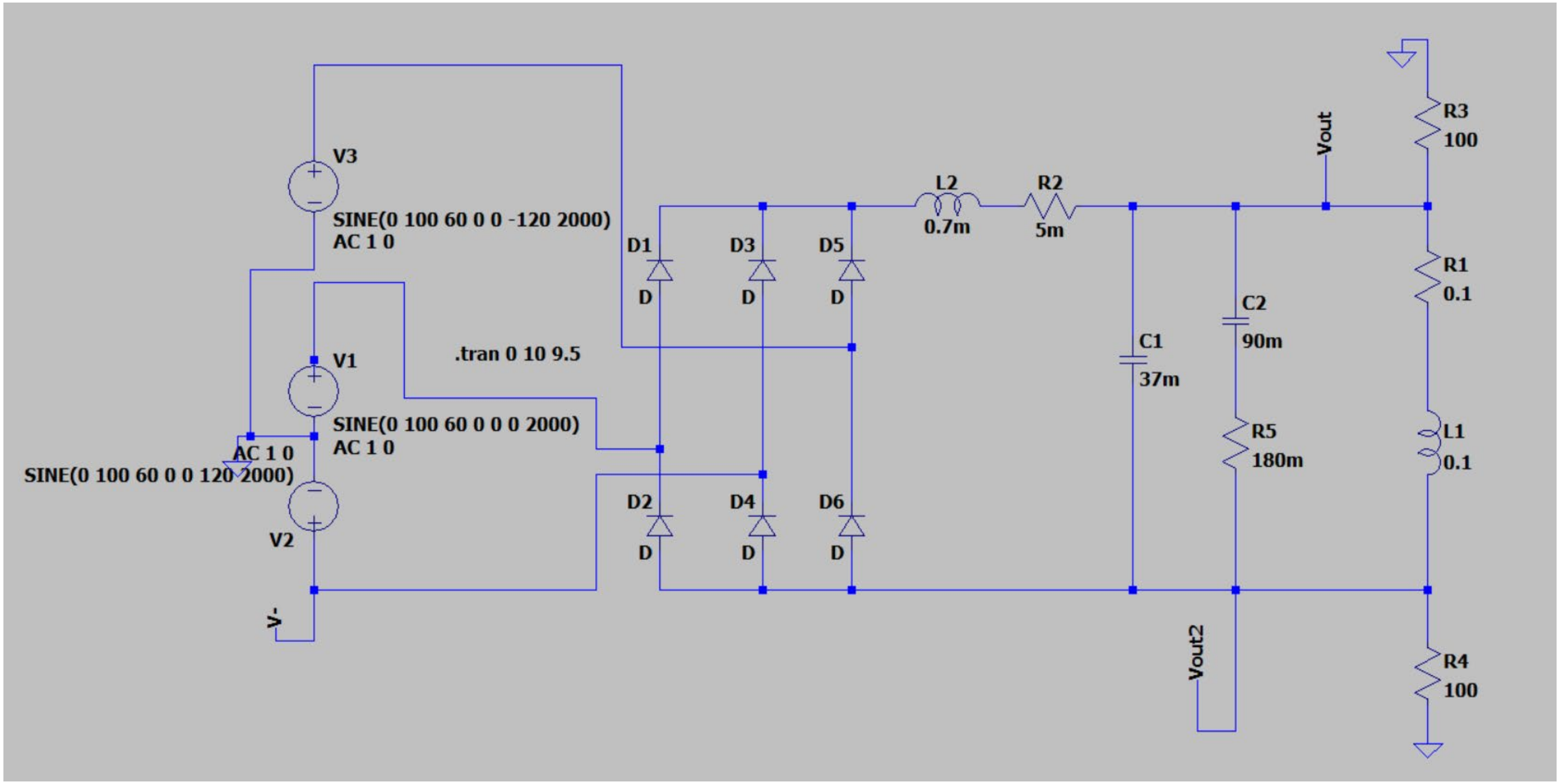
Three Phase Bridge Power Supply with Resistor, Inductor, and Capacitor



Three Phase Bridge Power Supply with Resistor, Inductor, and Capacitor - Measurements

	3 phase bridge with R,L & C C = 127mF
Current Percentage Ripple	0.0021% (21 ppm)
Peak to Peak current	32.48 mA
Average Current	1.542 kA
Voltage Ripple [V(out) – V(vout1)]	8.8 V
Avg Volt [V(out) – V(vout1)]	155.16

Three Phase Bridge Power Supply with Praeg Filter



Fermilab

Three Phase Bridge Power Supply with Praeg Filter - Measurements

	3 phase bridge with praeg filter
Current Percentage Ripple	0.000037% (.4 ppm)
Peak to Peak current	566 uA
Average Current	1.5217 kA
Voltage Ripple [V(out) – V(vout1)]	95 mV
Avg Volt [V(out) – V(vout1)]	153.7

Circuit Measurements for 2 phase bridge power supplies

	2 phase bridge with load only (R & L)	2 phase Bridge with R, C and L C = 127mF	2 phase Bridge with Praeg filter C = 127mF
Current Percentage Ripple	0.16 %	0.041 %	0.011 %
Peak to Peak current	1.94 A	641.47 mA	123.24 mA
Average Current	1.170 kA	1.55 kA	1.1602 kA
Voltage Ripple [V(out) – V(vout1)]	197.5 V	56 V	11 V
Avg Volt [V(out) – V(vout1)]	116.33 V	156.67 V	117.19

Fermilab

Circuit Measurement for 3 Phase Bridge Power Supplies

	3 phase bridge with R & L	3 phase bridge with R,L & C C = 127mF	3 phase bridge with praeg filter
Current Percentage Ripple	0.0044%	0.0021%	0.000037%
Peak to Peak current	67.60 mA	32.48 mA	566 uA
Average Current	1.5361 kA	1.542 kA	1.5217 kA
Voltage Ripple [V(out) – V(vout1)]	19.3 V	8.8 V	95 mV
Avg Volt [V(out) – V(vout1)]	154.16	155.16	153.7

Conclusion

- As shown, 3 phase bridge power supplies achieve the goal of reducing the current ripple below 50 parts per million.
- For this reason, these types of power supplies are used at Fermilab

Power supply at site

- Three phase bridge power supply with 75 V, 1000 A, and 50 ppm current ripple

Fermilab