

---

# **Motors with Analog Inputs**

**ECE 476 Advanced Embedded Systems**

**Jake Glower - Lecture #11**

Please visit [Bison Academy](#) for corresponding  
lecture notes, homework sets, and solutions

---

## Introduction:

The previous lecture looked at motors with digital inputs.

These were fairly low power:

- Stepper Motor: 5V @ 3A = 15W
- BLDC: 12.0V @ 10A = 120W
- Digital Servo Motor: 5V @ 3A = 15W

Larger motors tend to have analog inputs.

This lecture looks at driving two types of motors with a Pi-Pico:

- DC Servo Motors
  - Two Leads
  - 6W & 100W
- 3-Phase AC Synchronous Motors
  - Three Leads
  - 300W

# DC Servo Motors

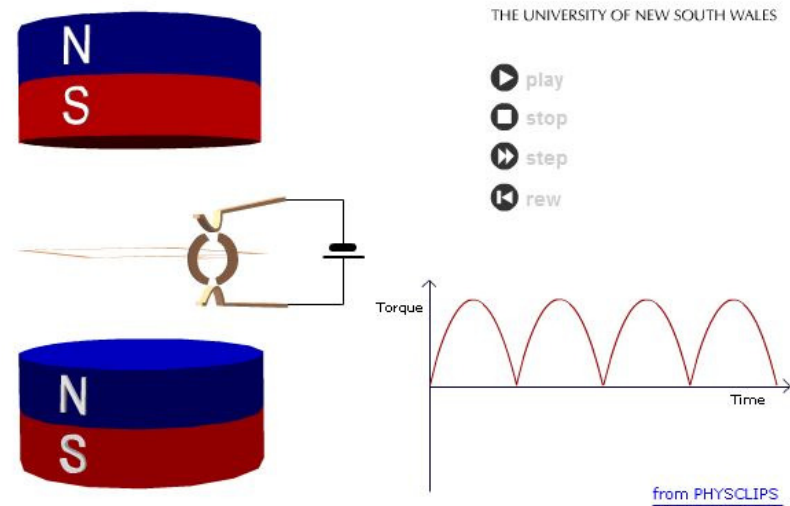
<https://www.animations.physics.unsw.edu.au/jw/electricmotors.html>

DC servo motors are what most people think of

- It has two leads (DC)
- Apply a DC voltage and it spins
- More voltage = spins faster
- Reverse voltage = spin opposite way

These are also DC generators

- Termed *dynamos*.
- Motors and generators are one and the same



---

# 6W DC Servo Motor

Available from Amazon

Gearhead motor

- Output of the motor drives a gear
- Slows the motor's speed
- Increases the motor's torque

Motor Input:

- 12V
- up to 500mA

Needs a buffer circuit



Greartisan DC 12V 550RPM Gear Motor High Torque Electric Micro Speed Reduction Geared Motor Centric Output Shaft 37mm Diameter Gearbox

[Visit the Greartisan Store](#)

4.5 ★★★★★ 1,153 ratings | [Search this page](#)

**Amazon's Choice** in Automotive Replacement Gear Kit Motors by Greartisan

100+ bought in past month

\$14<sup>99</sup>

**prime** Two-Day

FREE Returns

**Coupon:** ☐ Apply 5% coupon [Shop items >](#) | [Terms](#)

**Earn 5% back (\$0.74 in rewards)** on the amount charged to your Prime Visa.

Color: **550 RPM**



Size: **12 Volt**

**12 Volt** 24 Volt 24 volt

---

# Unidirectional Hardware: NPN Switch

Assume the motor spins in just one direction

- Input is 0V to 12V DC

An NPN transistor allows a Pico to drive the motor

- Any NPN capable of 12V & 500mA
  - Zetex 1051A works
  - gain > 300
  - max(I<sub>c</sub>) = 3A

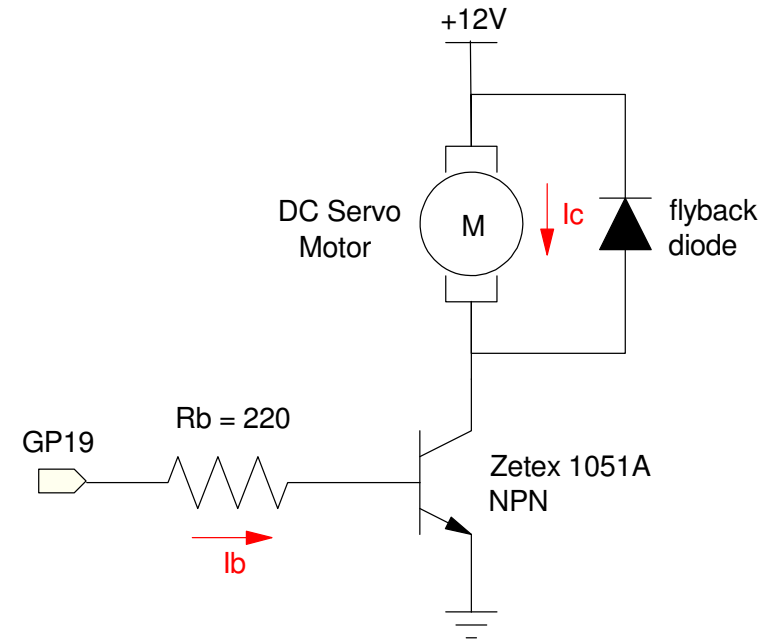
Calculations for R<sub>b</sub>:

$$h_{fe} \cdot I_b > I_c$$

$$12mA > I_b > \left( \frac{500mA}{300} \right) = 2.67mA$$

$$R_b = \left( \frac{3.3V - 0.7V}{I_b} \right)$$

$$216\Omega < R_b < 975\Omega$$



---

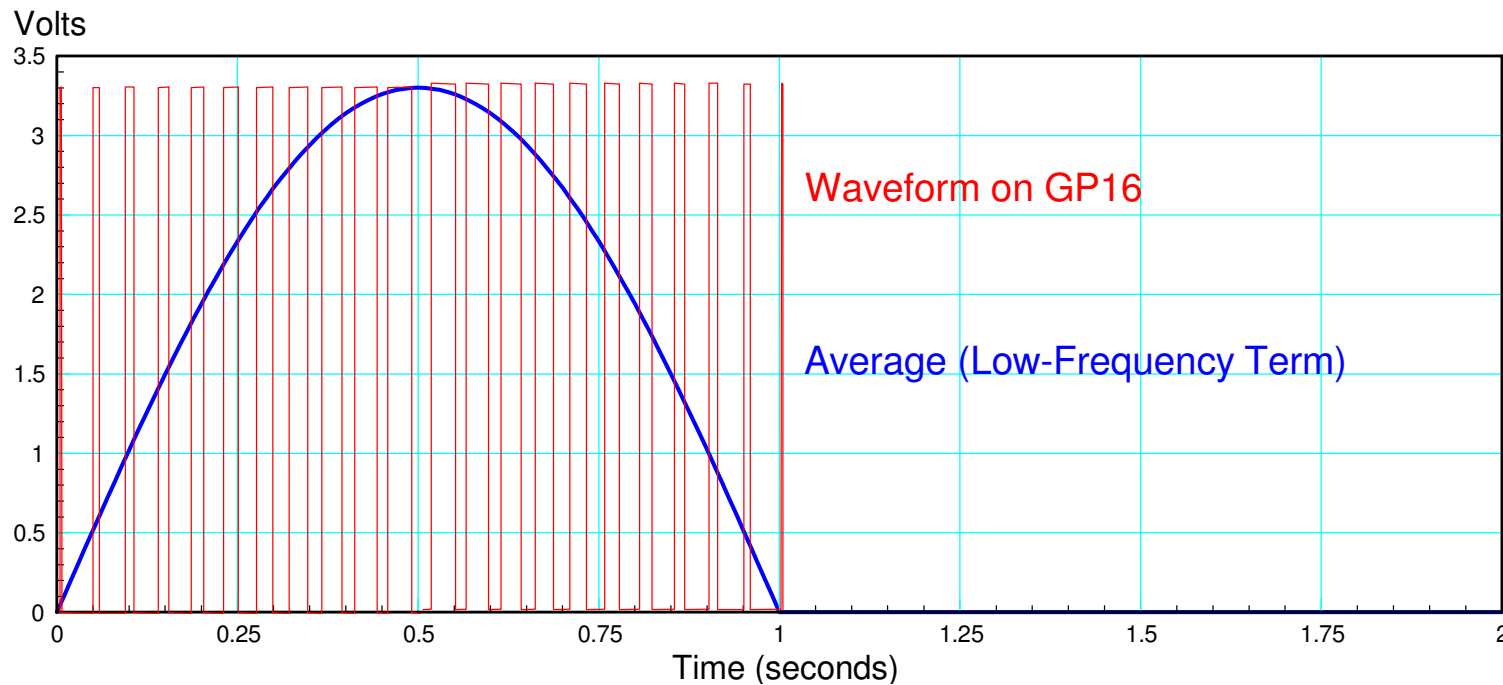
# Speed Control with an NPN Switch

The input to the BJT needs to be 0V or 3.3V

- 0V = off (0%)
- 3.3V = saturated (100%)

To generated outputs between 0% and 100%, use PWM

- Pulse Width Modulation



---

# PWM Code

## 20kHz PWM

- Outside audio range
- Motor's inductance filters out harmonics
- (current is smooth even though voltage is not)

## Look-Up Table to generate waveform

- Half-wave rectified sine wave

```
from machine import Pin, PWM
from time import sleep_ms
from math import sin, pi

Aout = Pin(16, Pin.OUT)
Aout = PWM(Pin(16))
Aout.freq (20_000)

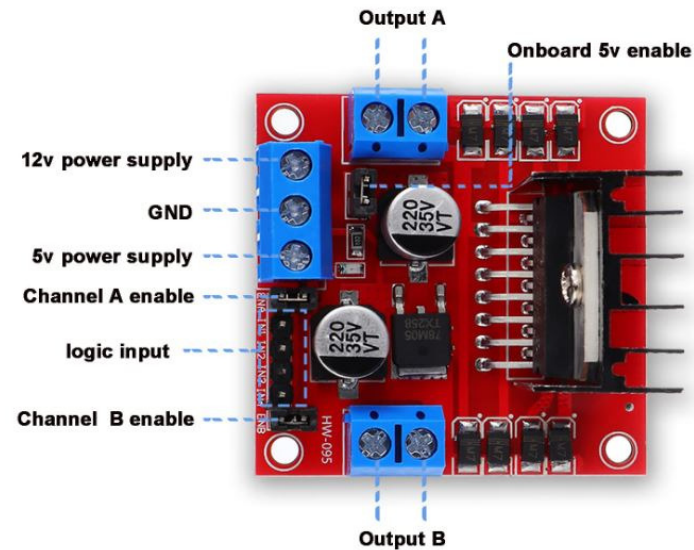
Table = []
for i in range(0,100):
    Table.append(int(65535*sin(i*pi/100)))
for i in range(0,100):
    Table.append(0)

i = 0
while(1):
    i = (i + 1) % 200
    Aout.duty_u16(Table[i])
    sleep_ms(10)
```

---

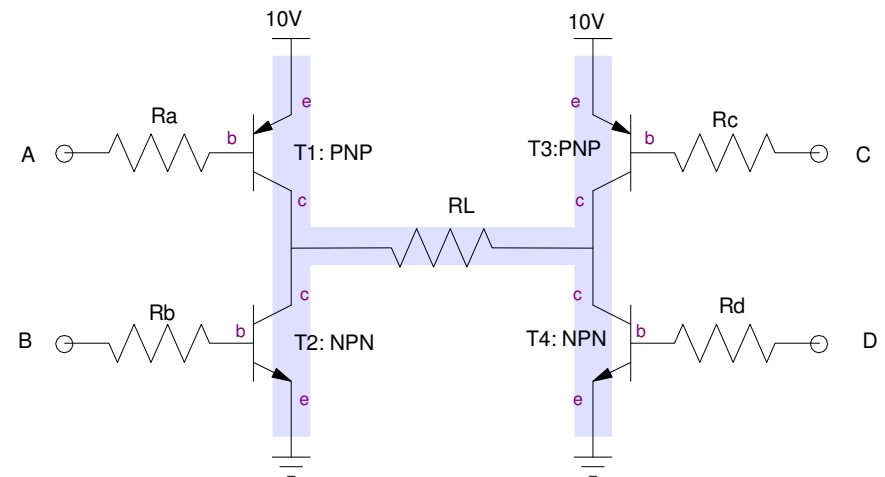
# Bidirectional Hardware: H-Bridge

- search L298N on Amazon
- 5V to 35V
- Up to 2A per channel
- Dual H-bridge
  - Can drive two DC motors
- \$2.50 each



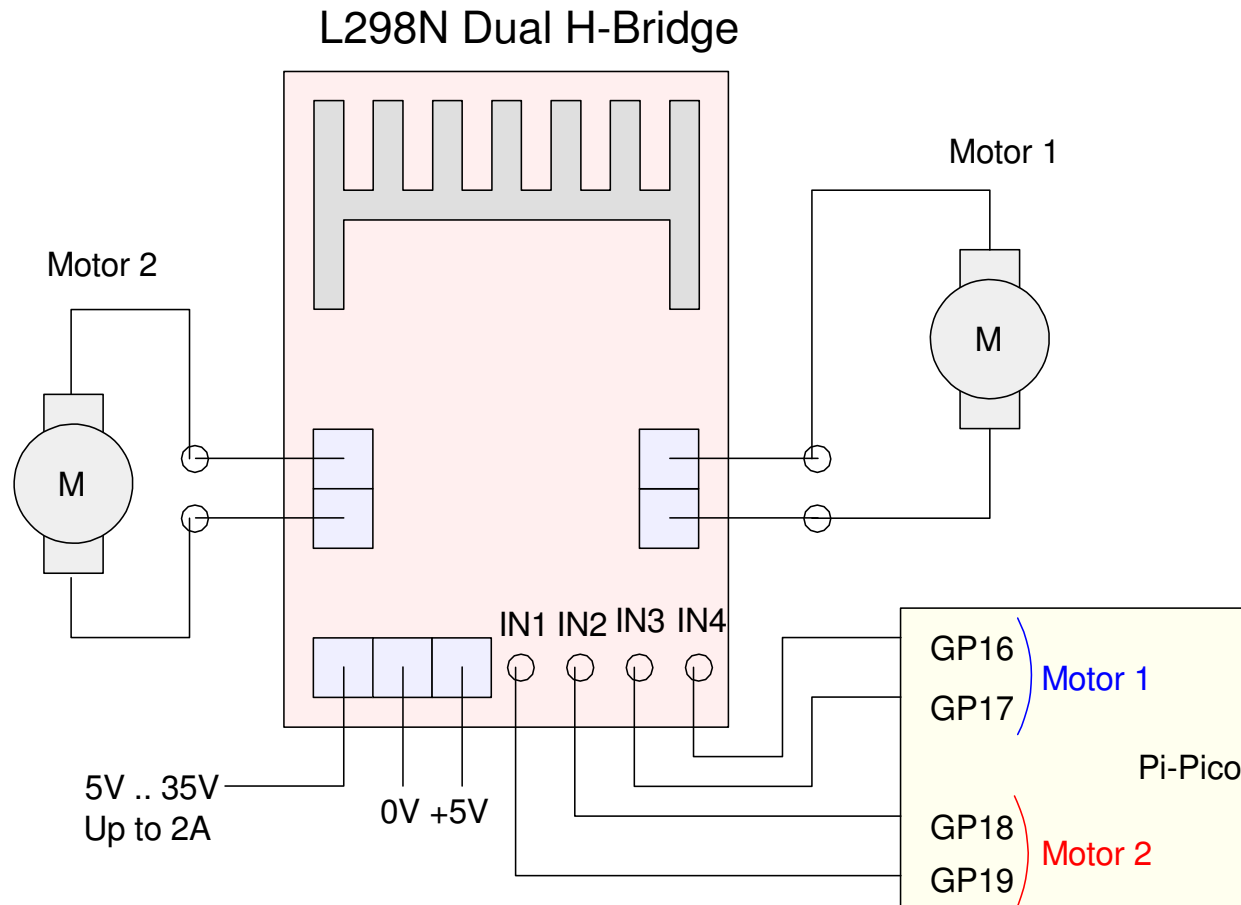
H-bridges allow you to output

- Positive voltages, and
- Negative voltages
- Using just a single power supply



---

The hardware connection is similar to what was used for a stepper motor:



Two DC motors can be driven with a dual H-bridge driver

---

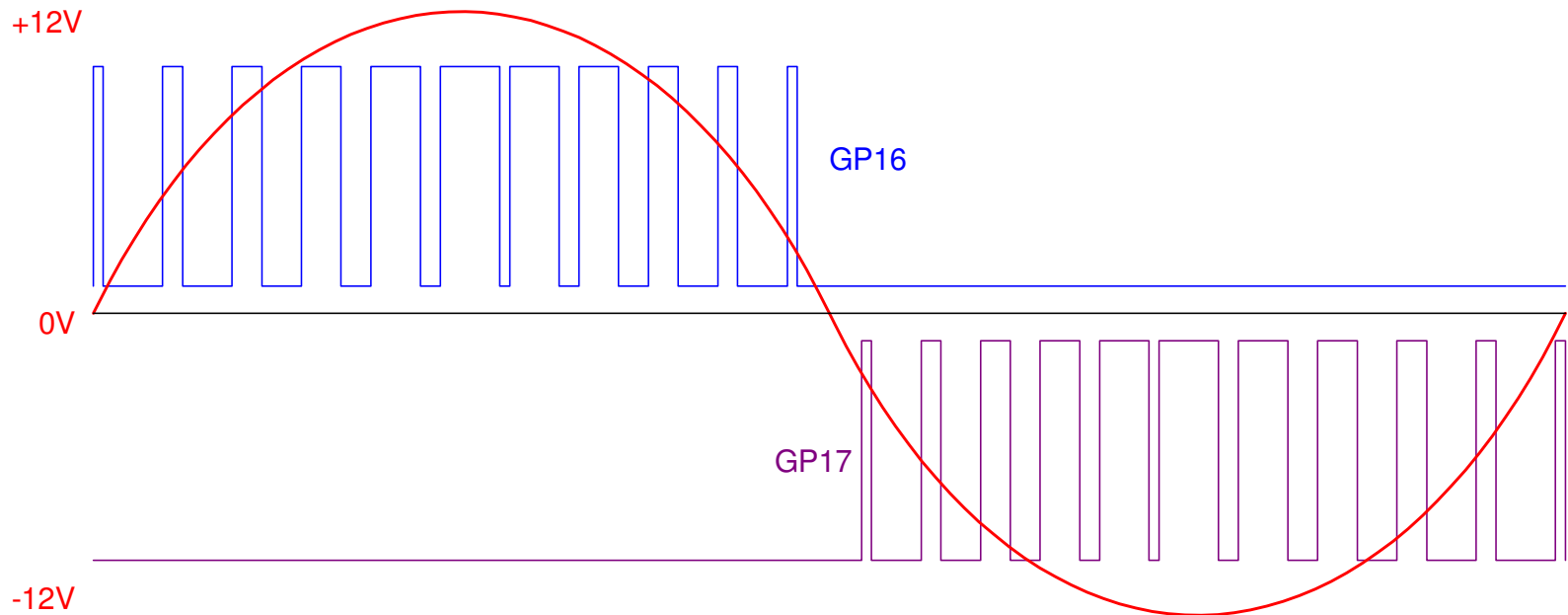
---

## H-Bridge & Software:

PWM again sets the speed of the motor

Direction set by which pin gets the PWM signal

	GP16	GP17
Forward	PWM	0V
Reverse	0V	PWM



---

## H-Bridge Code:

Almost the same as before  
if-else controls direction

- CW: PWM on GP16
- CCW: PWM on GP17

```
from machine import Pin, PWM
from time import sleep_ms
from math import sin, pi

Aout = Pin(16, Pin.OUT)
Aout = PWM(Pin(16))
Aout.freq (20_000)

Aout = Pin(17, Pin.OUT)
Aout = PWM(Pin(16))
Aout.freq (20_000)

Table = []
for i in range(0,100):
    Table.append(int(65535*sin(i*pi/100)))
for i in range(0,100):
    Table.append(0)

i = 0
while(1):
    i = (i + 1) % 200
    if(Table[i] > 0):
        Aout.duty_u16(Table[i])
        Bout.duty_u16(0)
    else:
        Aout.duty_u16(0)
        Bout.duty_u16(-Table[i])
    sleep_ms(10)
```

---

## DC Servo Motors (100W - 1000W)

Larger DC motors are available

- 1000W = \$86

These need

- More voltage (36V for this motor)
- More current (30A)

Replace the hardware with something that can take 30A and you're good to go



Brushed DC Electric Motor, 36V 1000W MY1020 Type Motor for Electric Scooter 3000 RPM Electric Motor for Go Cart E-Bike Scooter

Brand: YIYIBYUS

5.0 ★★★★★ 1 rating | [Search this page](#)

-10% \$86<sup>00</sup>

Typical price: ~~\$96.00~~

Or \$14.33 /mo (6 mo). Select from 2 plans

**Coupon:** ☐ Apply 5% coupon [Shop items >](#) | [Terms](#)

Earn 5% back (\$4.30 in rewards) on the amount charged to your Prime Visa.

Brand	YIYIBYUS
Speed	3000 RPM
Voltage	36 Volts
Horsepower	1000 Watts
Manufacturer	YIYIBYUS

# Example: Power MOSFET

Search Digikey for

- MOSFET
- N-Channel
- 30A+
- Through Hole

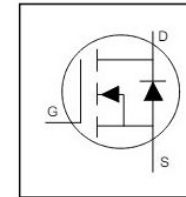
results in 1,252 options.

Selecting an IRFB7545PBF Mosfet

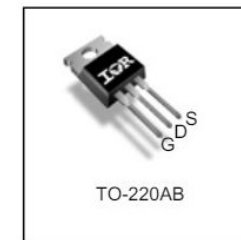
- $V_{ds(max)} = 60V$
- $R_{ds(on)} = 5.9m\Omega @ 57A @ 10V$
- $4010pF @ 25V$
- $V_{gs(th)} (max) = 3.7V$
- $I_{ds(max)} = 95A$

Strong<sup>IR</sup>FET™  
IRFB7545PbF

HEXFET® Power MOSFET



$V_{DSS}$	60V
$R_{DS(on)}$ typ.	4.9m $\Omega$
max	5.9m $\Omega$
$I_D$	95A



## Application

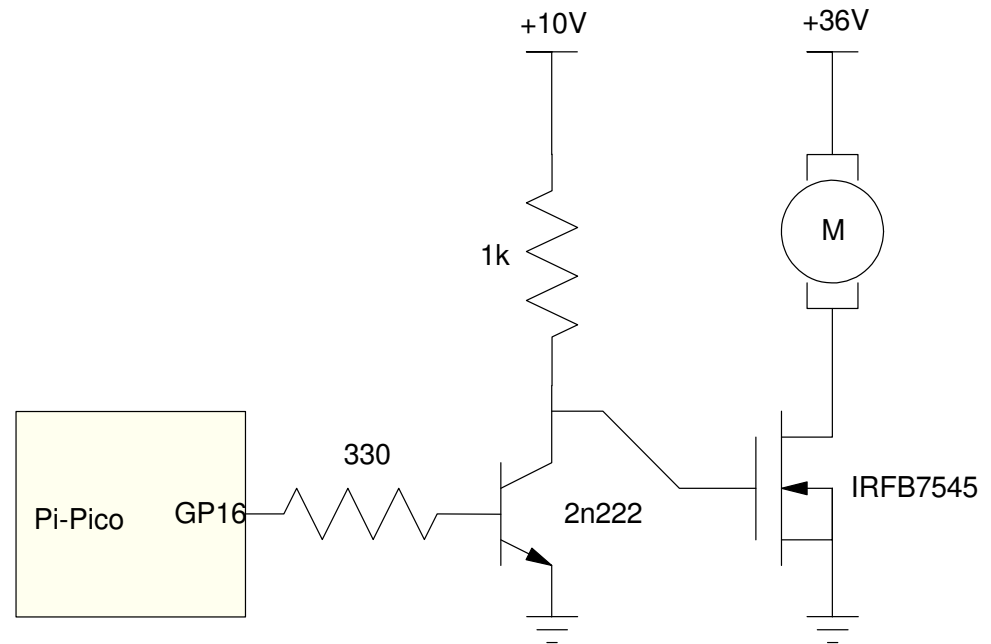
- Brushed motor drive applications
- BLDC motor drive applications
- Battery powered circuits
- Half-bridge and full-bridge topologies
- Synchronous rectifier applications
- Resonant mode power supplies
- OR-ing and redundant power switches
- DC/DC and AC/DC converters
- DC/AC inverters

---

## Translation:

- If you set  $V_{gs} < 3.7V$ , the MOSFET turns off (good)
- If you set  $V_{gs} = 10V$ , the on resistance is  $5.9m\Omega$  (good)
- At max current (30A)
  - $R_{ds} = 0.0059 \Omega$
  - $V_{ds} = 0.177V$  (good)
  - $P = 5.31W$
- This Mosfet can handle 95A
  - overkill

Mosfets will almost always out-perform BJT transistors.



---

## Max Switching Time

The capacitances sets this

- $C_{gs} < 4010\text{pF}$

The RC time constant is

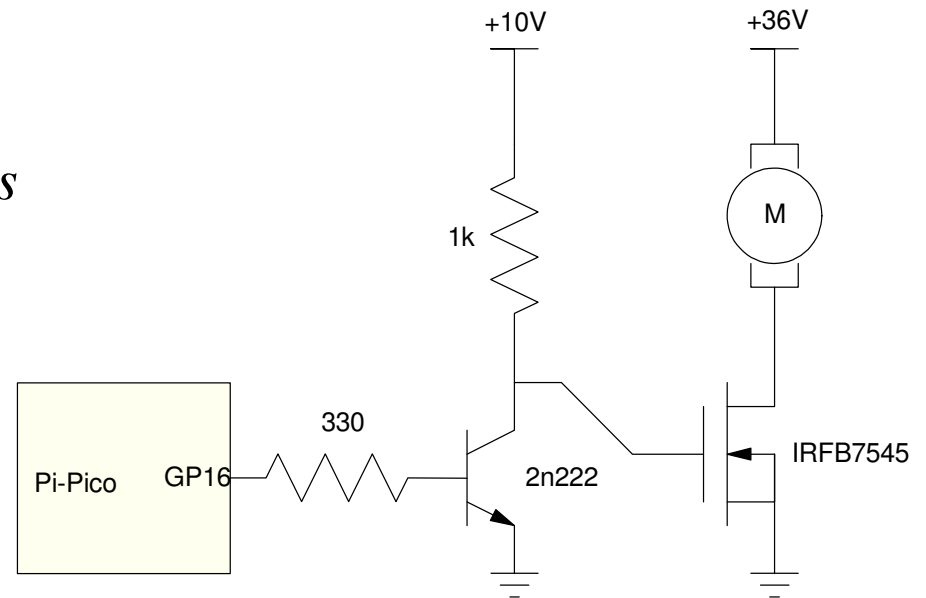
$$t = RC = (1k\Omega)(4010\text{pF}) = 4.01\mu\text{s}$$

$T(\text{on}) \approx 3$  time constants

- $12.03\mu\text{s}$
- $f(\text{max}) = 1/121\mu\text{s} = 83.1\text{kHz}$

20kHz PWM should be fine.

- Same code as before
- 100Hz PWM also works



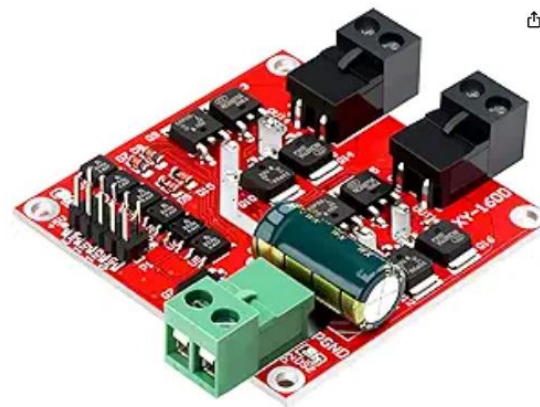
# Bidirectional Motor

Higher-power drivers are available

- \$16: 160W driver
- \$19: 1200W driver

Same PWM output from Pi-Pico

- The 1200W driver uses a switch for direction
  - 0V = CW
  - 3.3V = CCW



DROK DC Motor Driver, L298 Dual H Bridge Motor Speed Controller DC 6.5V-27V 7A PWM Motor Regulator Board 12V 24V Electric Motor Control Module Industrial 160W with Optocoupler Isolation

Visit the DROK Store  
★★★★☆ 134

Amazon's Choice Overall Pick

100+ bought in past month

\$15<sup>99</sup>

Get Fast, Free Shipping with Amazon Prime  
FREE Returns

Available at a lower price from other sellers that may not offer free Prime shipping.

Style: 160W

160W	360W
\$15.99	\$16.79



Roll over image to zoom in

DC Motor Speed Controller, Brush Motor Driver Controls Module DC 9V-60V 12V 24V 36V 48V 60V Motor Pulse Width Modulator Regulator 20A 1200W PWM Monitor Dimmer Governor with Switch & Knob +1

Visit the SDRGEEK Store  
★★★★☆ 391 | Search this page

Amazon's Choice in Electrical Motor Controls by SDRGEEK

100+ bought in past month

\$18<sup>90</sup>

Get Fast, Free Shipping with Amazon Prime  
FREE Returns

Get \$10 off instantly: Pay \$8.90 ~~\$18.90~~ upon approval for the Amazon Store Card. No annual fee.

## About this item

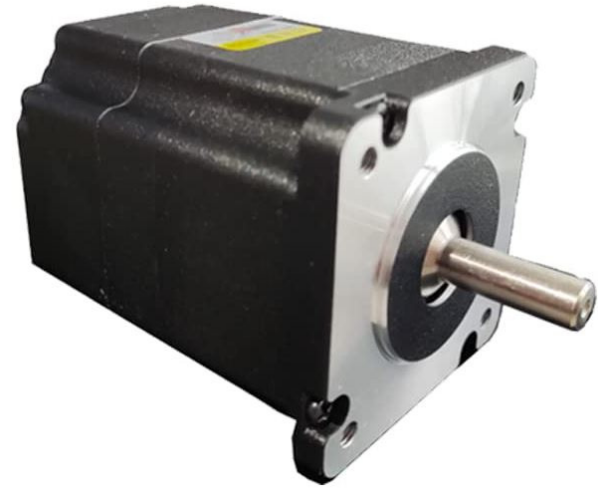
- Parameters: motor speed controller input voltage range is 9-60V, output current range is 0-20A, continuous power is 1200W.
- Application: the dc motor driver can be used to brush motor speed regulation, light dimming regulation in the DC circuit. Note: The motor cannot be used in electric vehicles.
- Speed Control: our motor control board can regulate motor speed by potentiometer; what's more, it support clockwise/anticlock-wise rotation adjustment.
- Easy Wiring: thick red wire for the positive of the power supply, and thick black for the negative; thick blue wire for the motor positive, and the thick green for the motor negative.
- PWM: the advantage of using a pulse width modulation (PWM) method for dimming / speed regulation is that the energy of the power supply can be fully utilized and the circuit is highly efficient.

---

## 3-Phase AC Synchronous Motors (100W - 400W)

Larger motors tend to be 3-phase AC motors

- No commutator
  - Reduced size
  - Improved efficiency
  - Extended life
  - Less (no) RF interference
  - Lower cost
  - \$39 = 300W AC motor (top)
  - \$29 = 1089W AC motor (bottom)



Note on size:

- 300W = Tour-de-France athlete
  - Average output of world-class cyclist
  - 1000W for short bursts



## 3-Phase Motor Drivers

These motors require a 3-phase inputs

- Drivers available from Amazon
- Called ESC controllers

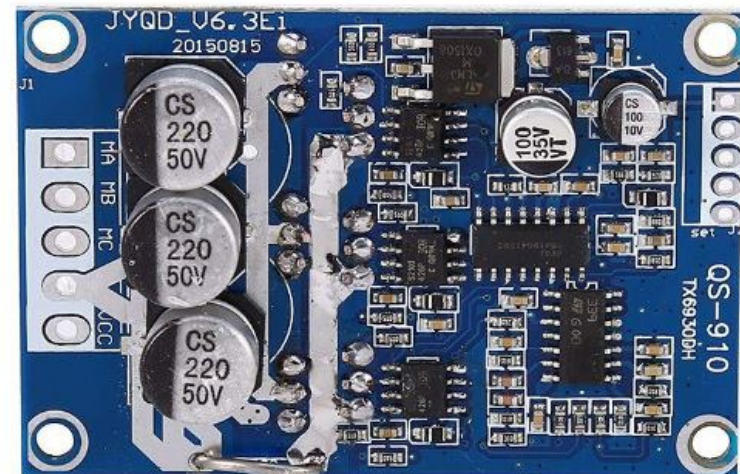
Top Board:

- 6V-80V, 1600W
- PWM input,
- I2C input, or
- 0-5V analog input



Bottom Board: Input can be

- 12V-36V, 300W
- 0-5V analog input



---

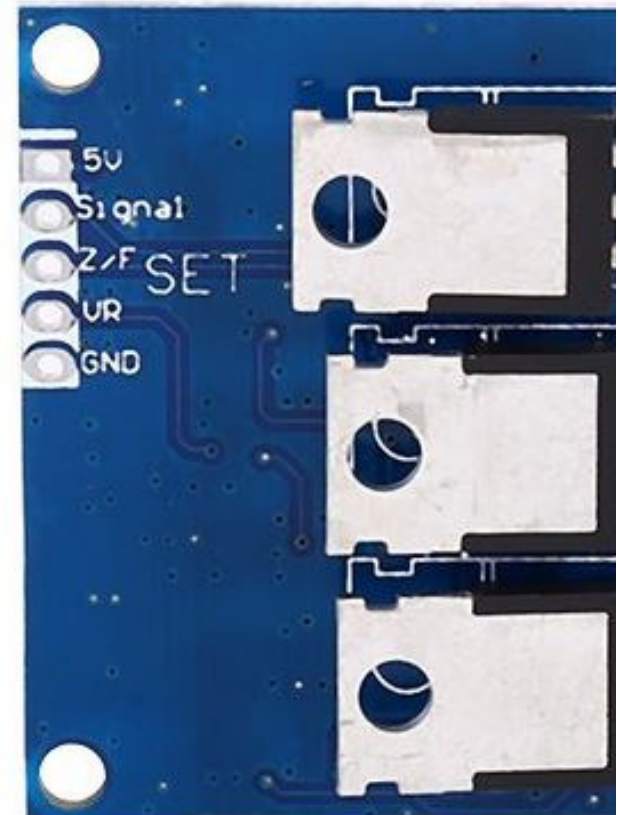
## Pi-Pico Signals

With the latter board, there are five inputs:

- 5V: 5V output from ESC control
- Signal:
  - Square wave output to Pi-Pico.
  - Frequency indicates the speed of the motor
- Z/F: Direction control.
  - 0V = CW
  - 5V = CCW (3.3V logic compatible)
- VR: 0V to 5V speed control (analog)
- GND: Common ground for all components

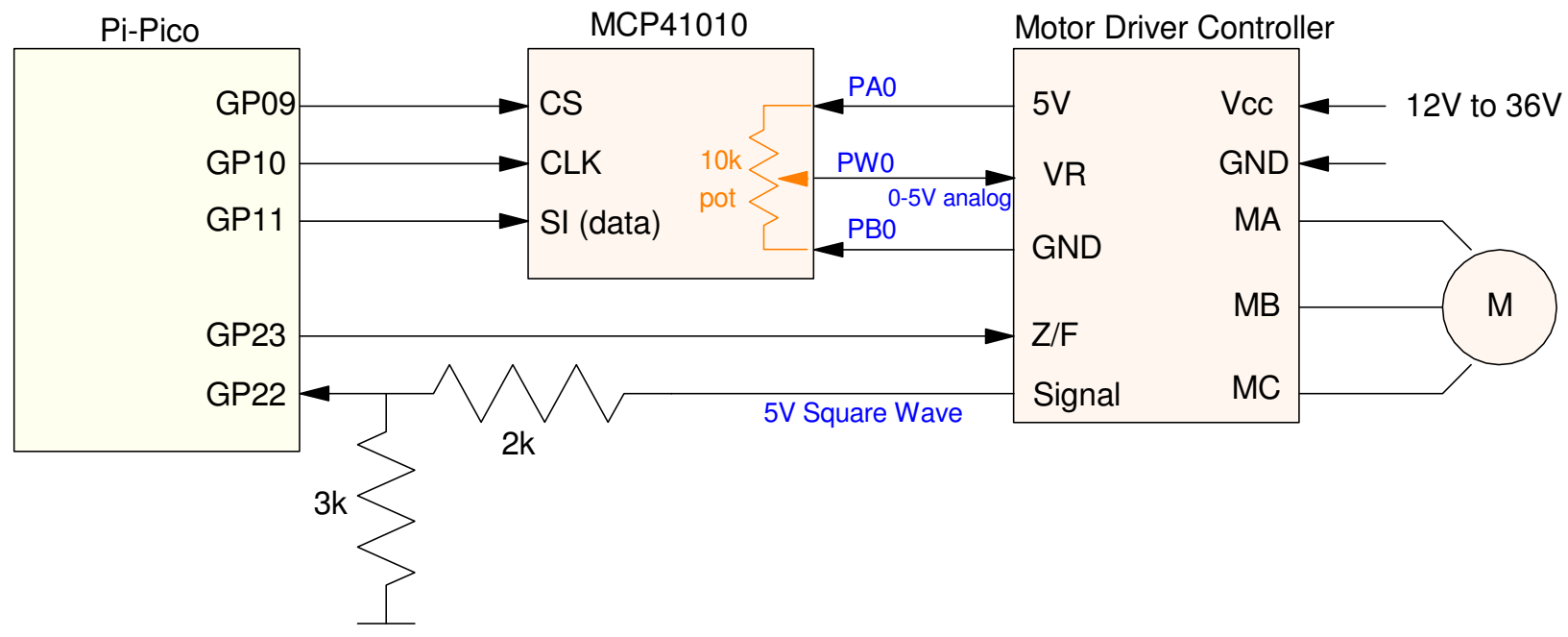
### Speed Control

- 0-5V signal on VR
- Normally set by a potentiometer
- Wiper output = 0V to 5V
  - Pi-Pico drives VR to set the motor speed



## Option 1: Digital Potentiometer

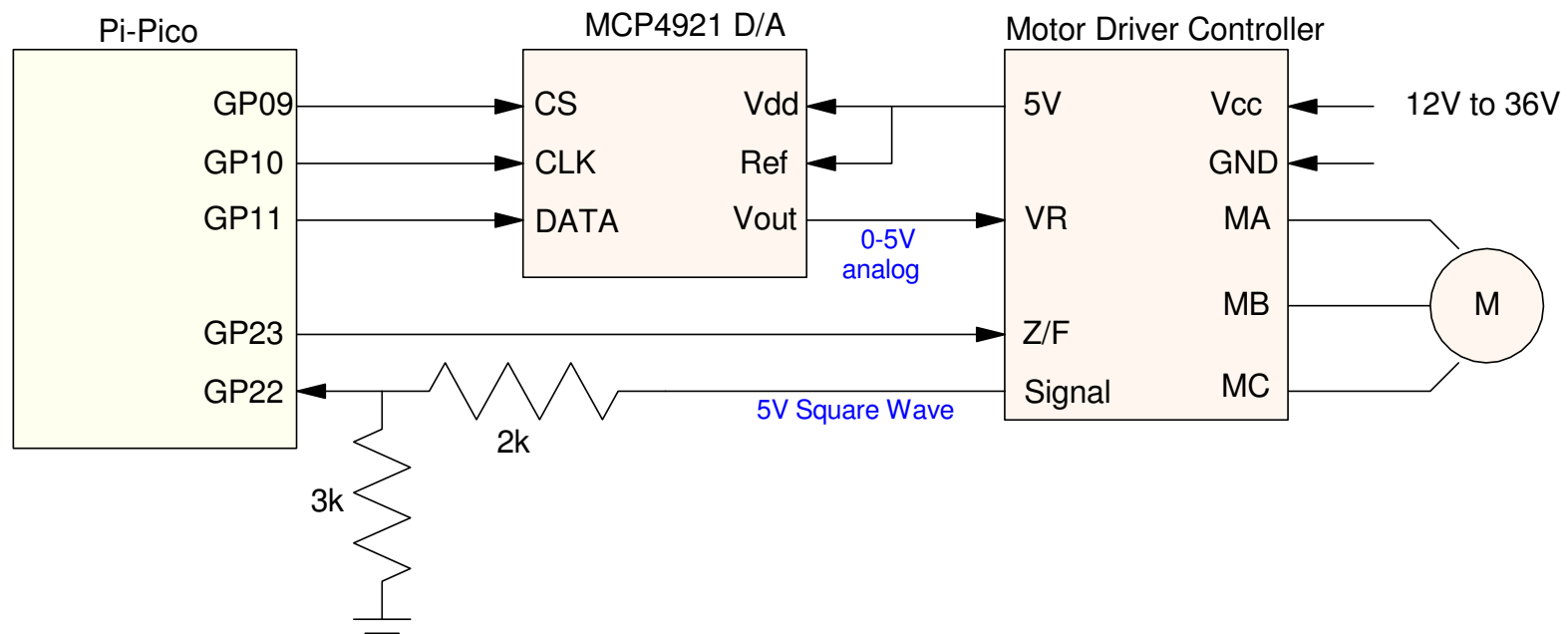
- Replace the potentiometer driving VR with a digital pot
  - ex: MCP41010 digital pot
  - Gives 256 speed settings
- SPI data interface to the digital pot
  - *Signal* reports the speed of the motor to the Pi-Pico
  - Two resistors drop this voltage down to 3.0V



## Option 2: D/A

Drive VR with a D/A chip

- MCP4921 (lecture #9).
- SPI data interface
- Results in 4095 speeds (12-bit D/A)



## Option #3: PWM Output

- Built in feature of the Pi-Pico
- Need to increase voltage from 3.3V to 5V
- Need to remove chatter (1kHz PWM)

A 2nd-order lowpass filter with a corner at 100 rad/sec works

- Reduces the 1st harmonic at 1kHz to 1.3mVpp

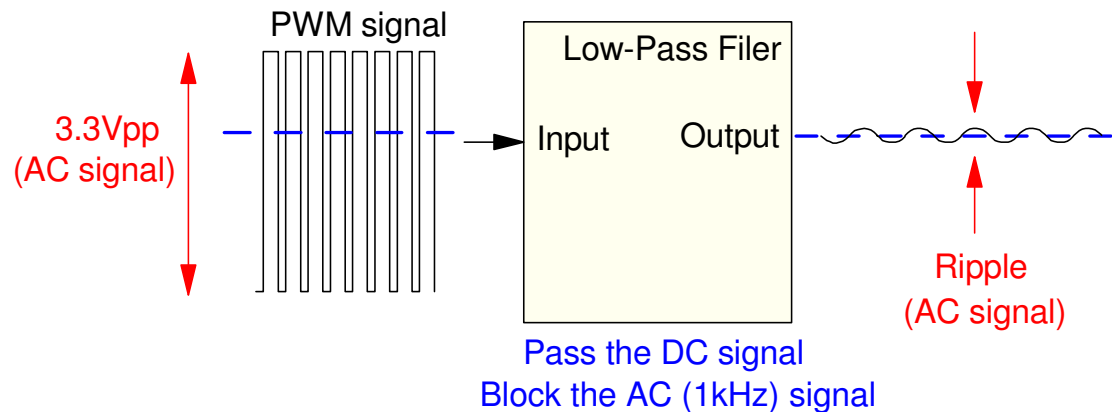
$$k = 1.56 = 1 + \frac{R_9}{R_8}$$

$$3 - k - 2 \cos \theta$$

$$\theta = 43.9^\circ$$

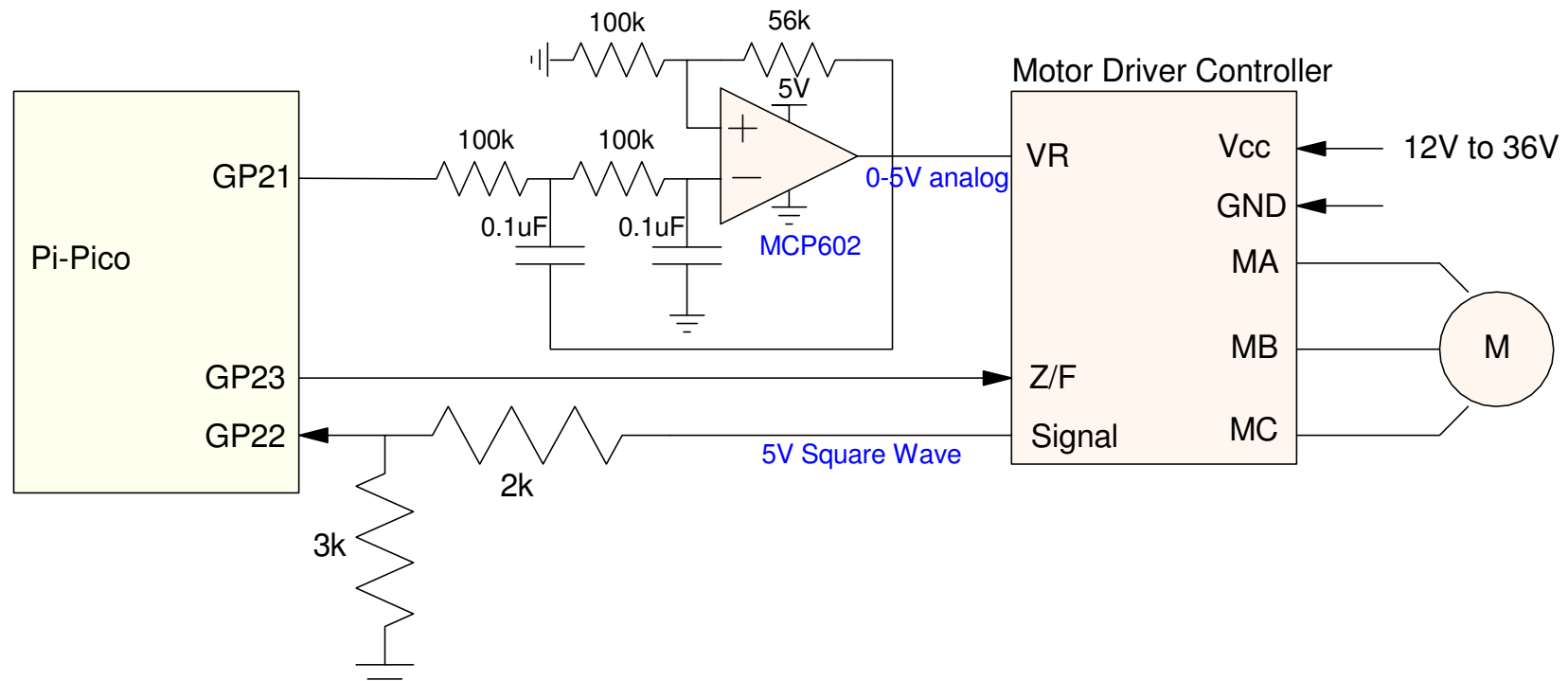
$$\frac{1}{RC} = 100$$

$$V_2 = \left( \frac{k \cdot 100^2}{s^2 + 144s + 100^2} \right) V_0$$



If GP21 is a 3.3V<sub>pp</sub> sine wave at 1kHz

$$V_2 \approx \left( \frac{1.56 \cdot 100^2}{s^2 + 144s + 100^2} \right)_{s=j2000\pi} \cdot 3.3V_{pp} \approx 1.3mV_{pp}$$



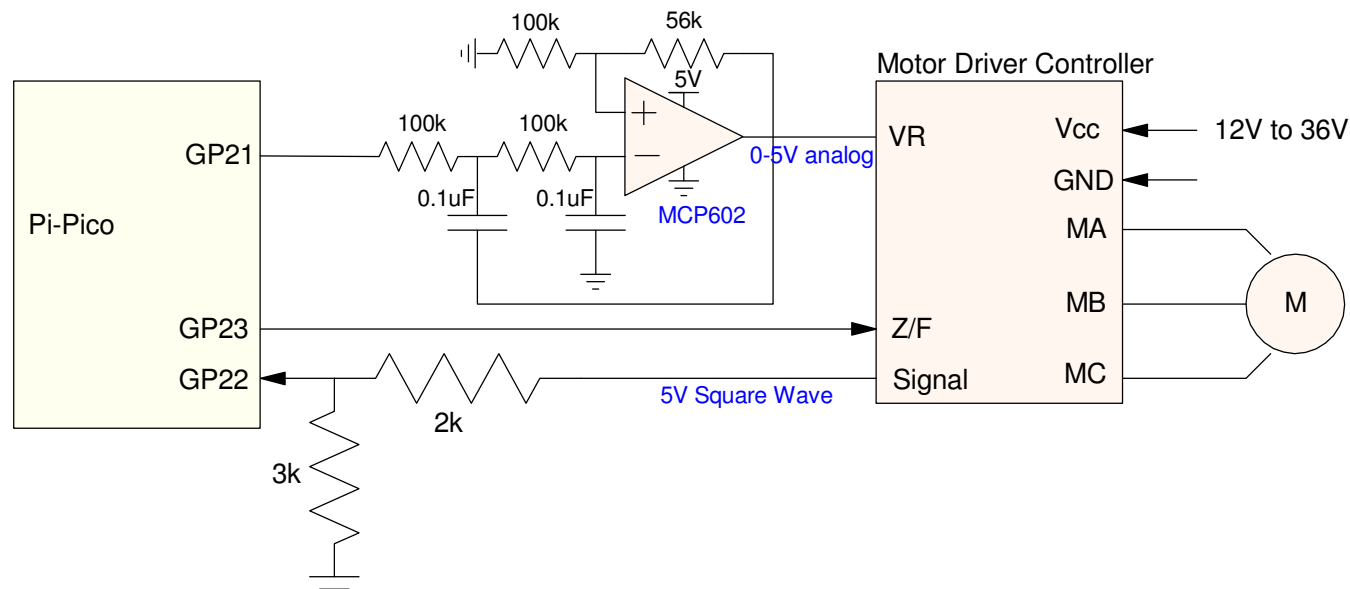
## Option #4: PWM Input

Eventually, someone will come up with an ESC controller with a PWM input.

- This will simplify the hardware somewhat.
- Until then, you can get by with any of these three options

For the rest of this lecture, we'll go with option #3

- PWM and low-pass filter



---

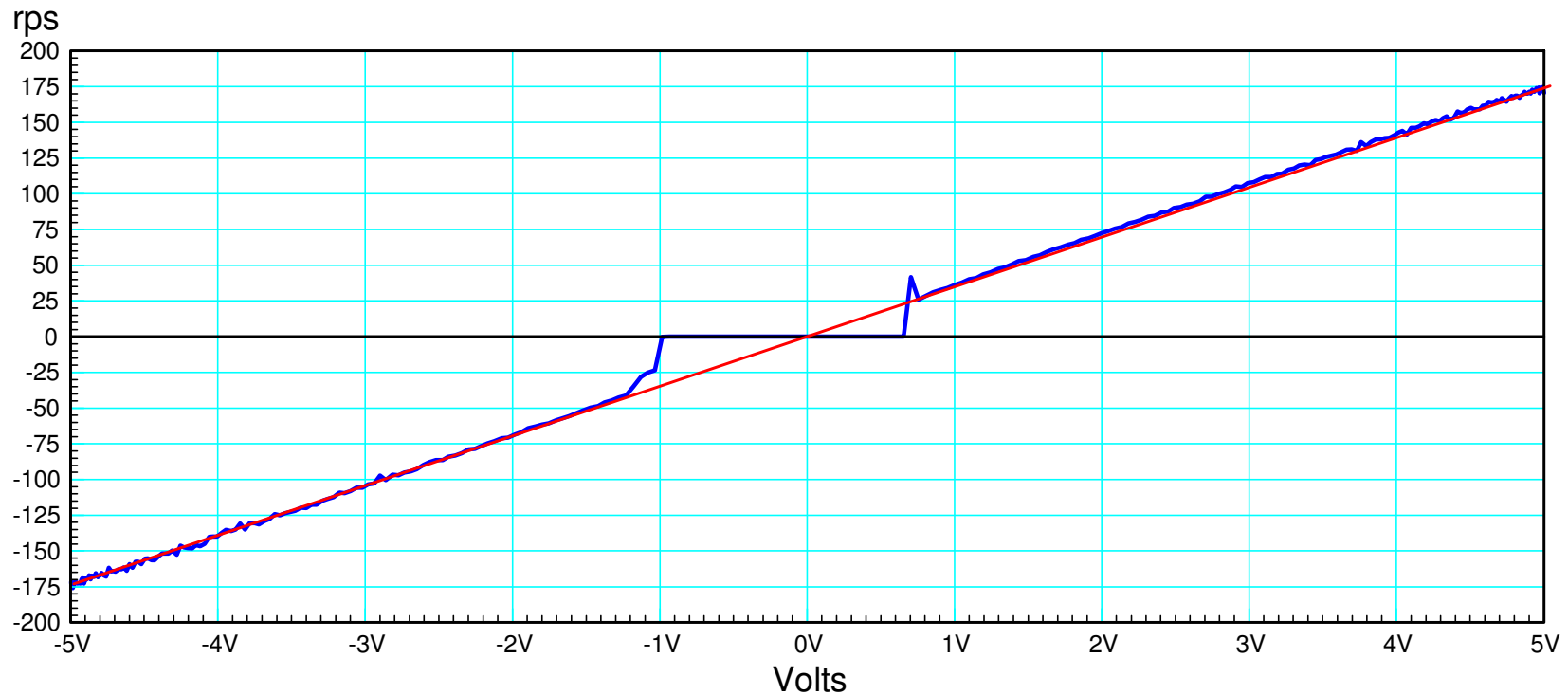
# Speed vs. Voltage Test

Vary VR from -100% to +100% speed

- Z/F sets direction

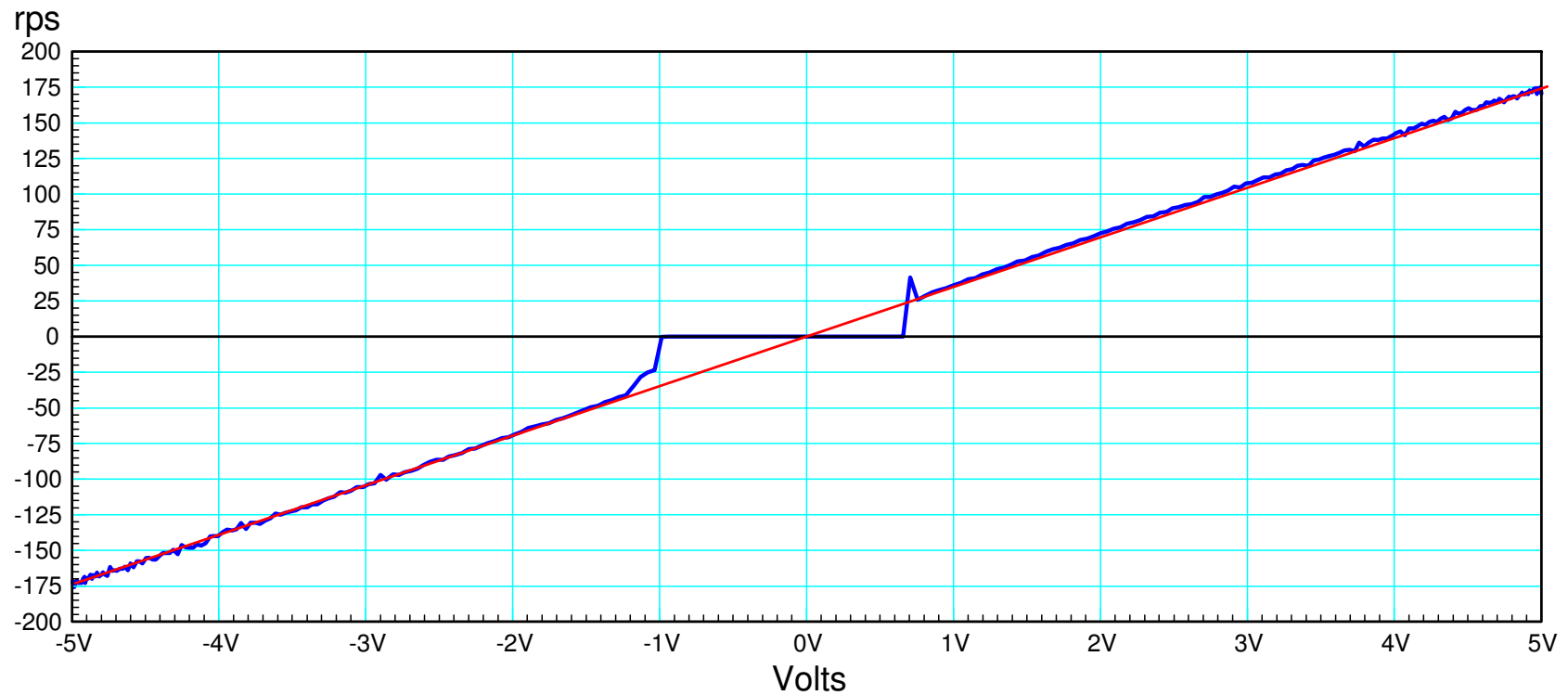
Pause 100ms to speed to stabilize and record motor speed

- GP22 records the speed by measuring the frequency from *Signal*



## Speed vs. Voltage Results:

- $\omega \approx \left(\frac{175}{5}\right) V$  rps
- There is a dead-zone
  - Probably good
  - No back emf to limit current at 0 rps



---

## Code: Speed Test

Only the main routine is shown:

- V varies from -5V to +5V
- Edge interrupts count rising edges on GP22
  - *Signal* from the ESC controller
- The interrupt records the period in the variable, dT

The frequency in Hz (proportional to motor speed) is then found as

$$\text{Hz} = 1,000,000 / \text{dT}$$

```
t = 0
dt = 0.1
kv = 65535 / 5
Hz = 0

while(t < 64):

    V = 5*sin(t/10)

    if(V > 0):
        Vout.duty_u16(int(V*kv))
        Dir.value(0)
    else:
        Vout.duty_u16(int(-V*kv))
        Dir.value(1)

    Hz = 1_000_000 / dT

    print(V, Hz)
    t += dt
    sleep(dt)
```

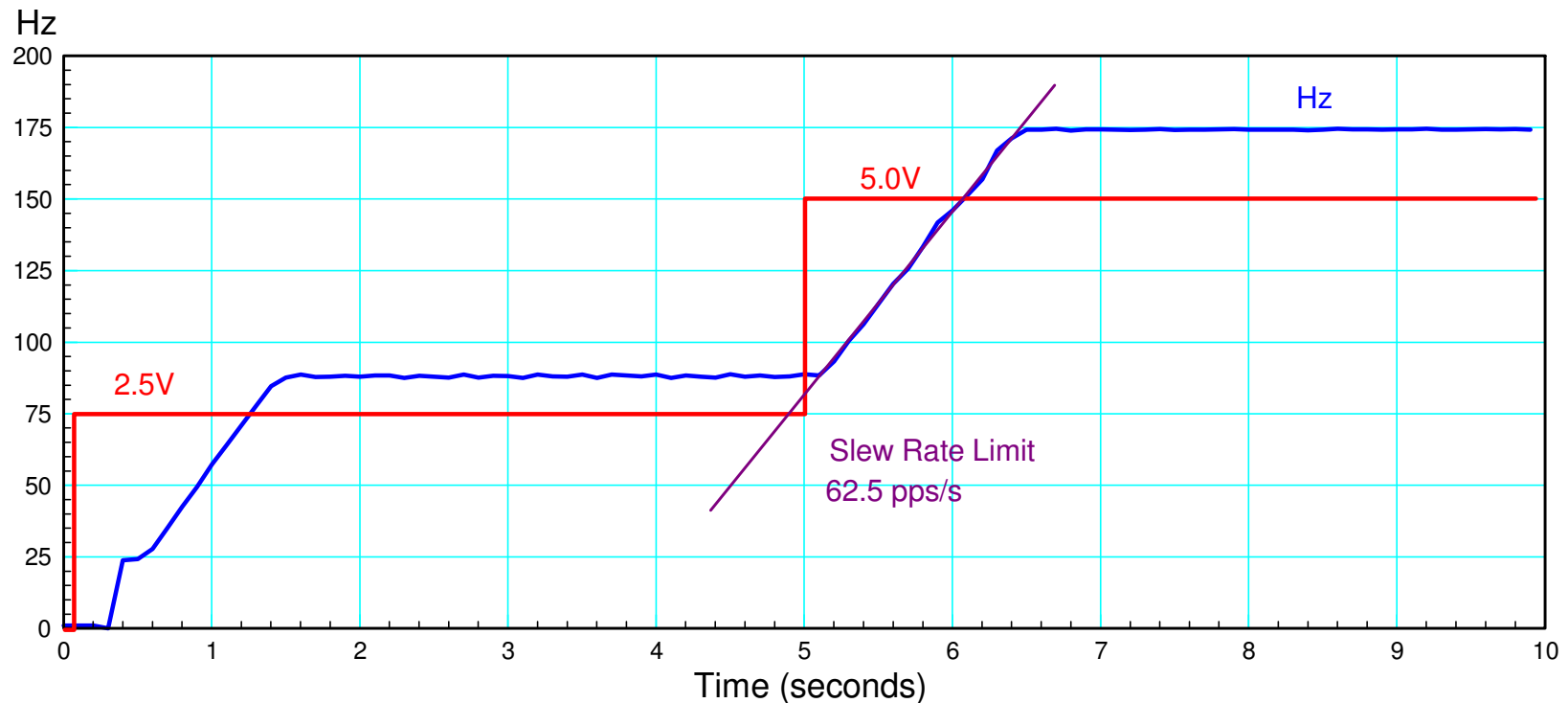
---

## Step Response:

Apply a step change in the input, VR & record the speed

Result:

- Dead-zone < 25Hz
- Slew-rate limit of 62.5Hz/sec (probably good)



---

## Code:

Same as before

- Start at 0V
- Step to 2.5V at t=0
- Step to 5.0V at t=5

```
t = 0
dt = 0.1
kv = 65535 / 5
Hz = 0

while(t < 10):
    if(t < 5):
        V = 2.5
    else:
        V = 5

    if(V > 0):
        Vout.duty_u16(int(V*kv))
        Dir.value(0)
    else:
        Vout.duty_u16(int(-V*kv))
        Dir.value(1)

    Hz = 1_000_000 / dt

    print(V, Hz)
    t += dt
    sleep(dt)
```

Code for a step response

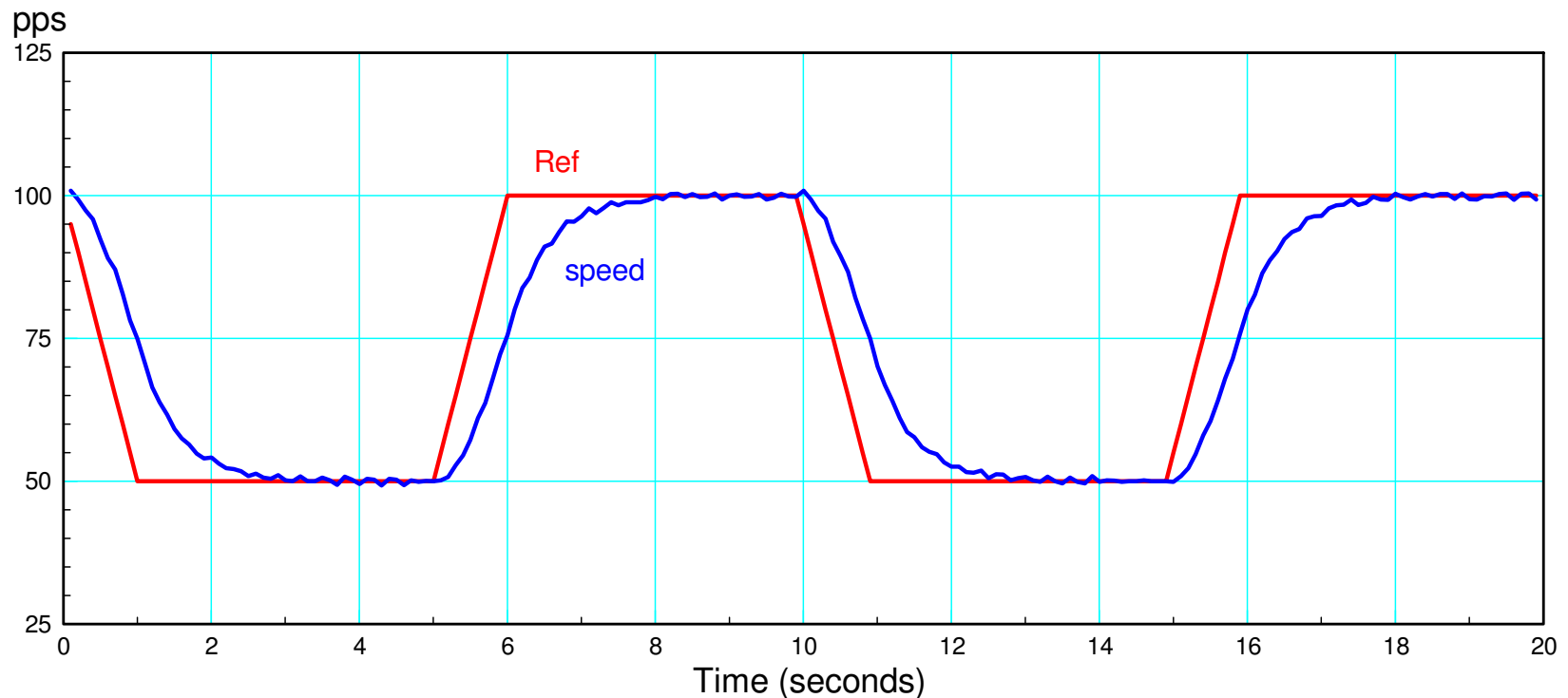
---

---

# Motor Speed Control: I Control

Use feedback to regulate the motor speed

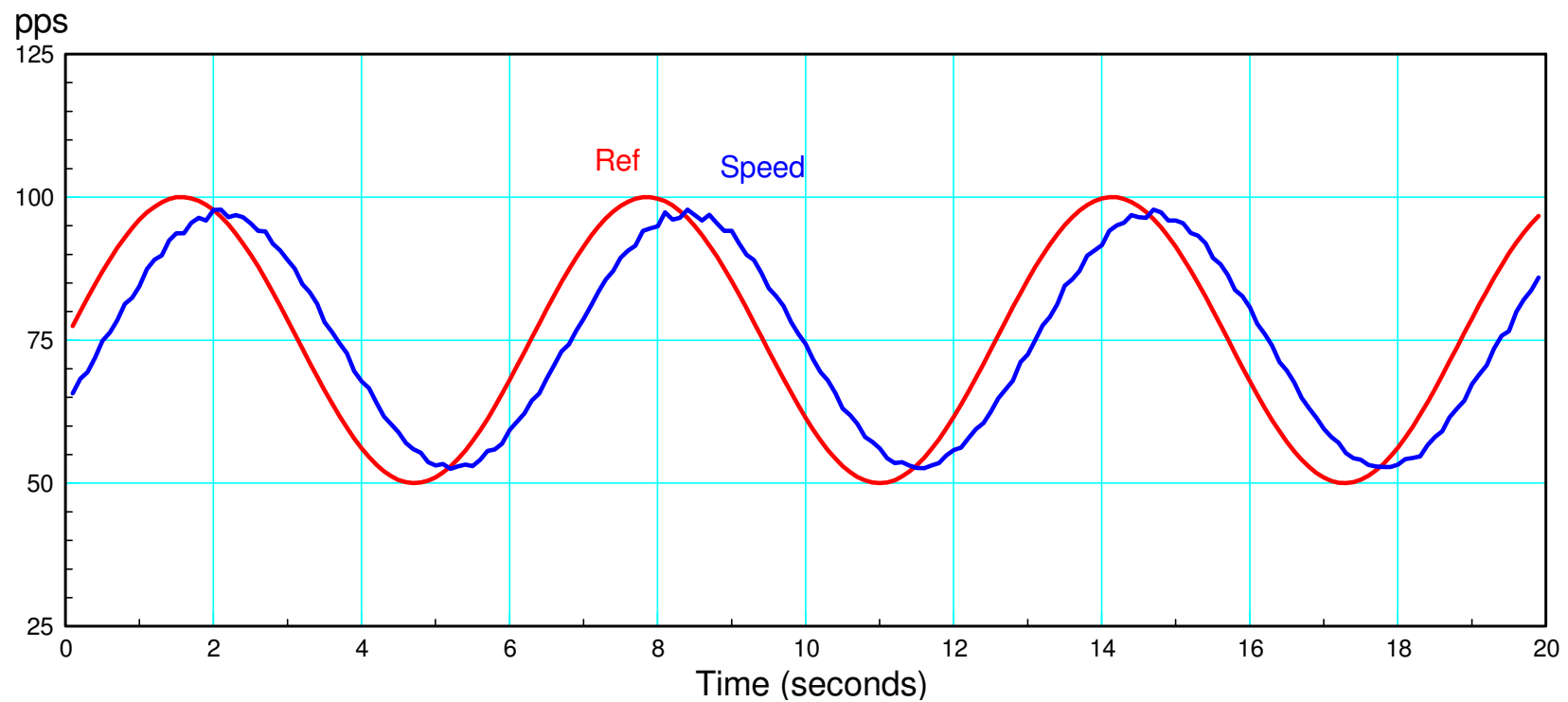
- $V = 0.05 \int (Ref - \omega) dt$
- Step response: results in perfect tracking (after 2 seconds ish)



---

## Tracking a sine wave

- 1 rad/sec
- Time delay (output lags)
- Slight drop in gain



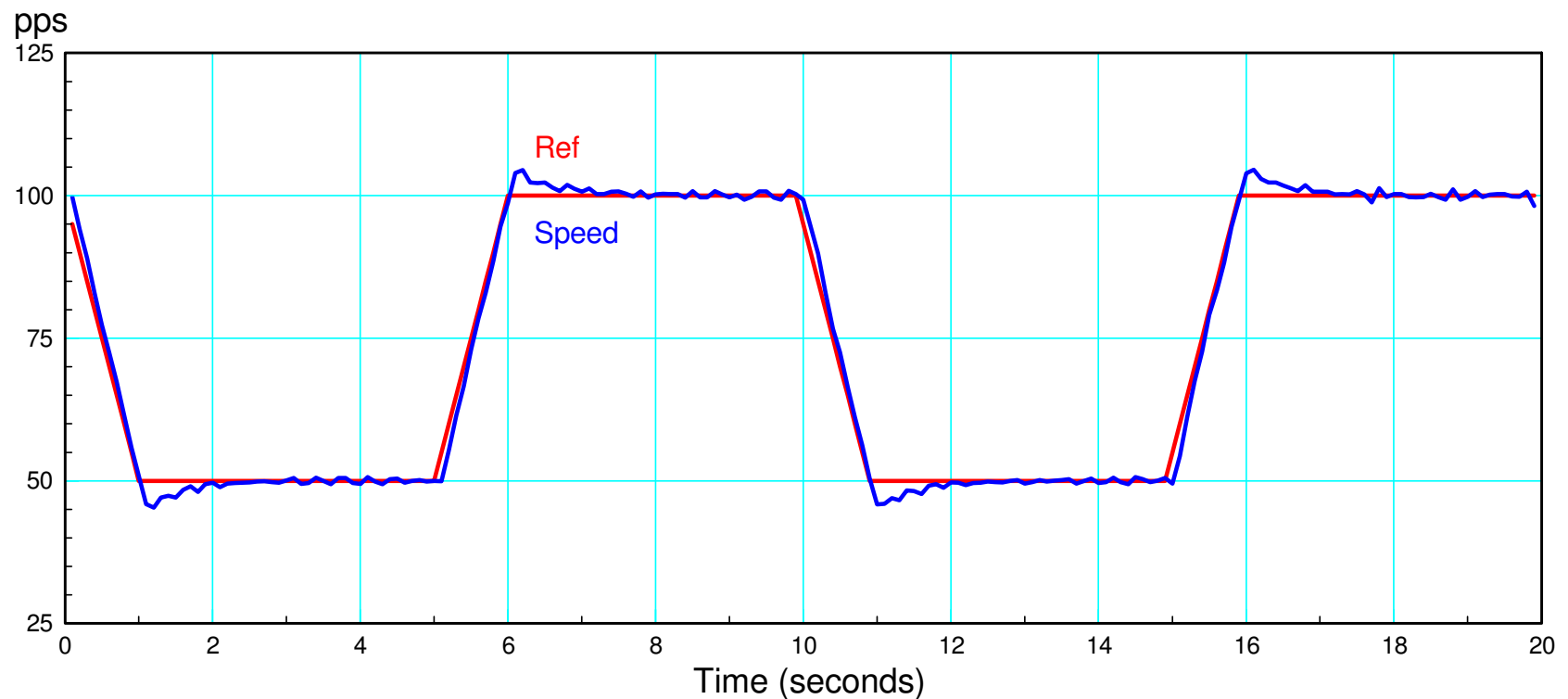
---

## I + Feedforward Control

Add a term: pre-calculate the voltage based upon speed

$$\omega \approx \left( \frac{175}{5} \right) V$$

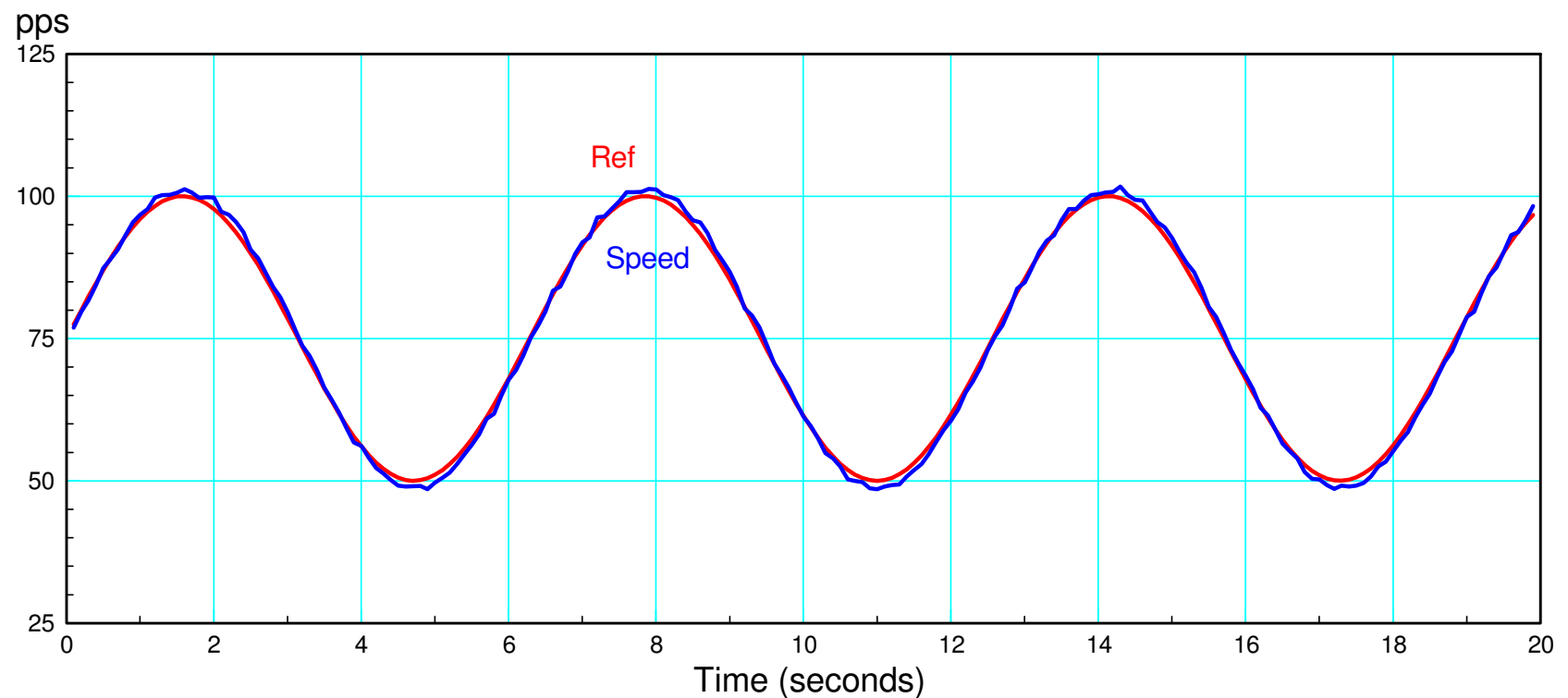
$$V = \left( \frac{5}{175} \right) Ref + 0.05 \int (Ref - \omega) dt$$



---

## I + Feedforward results in almost perfect tracking

- Feedforward term determines V
  - in theory
- Feedback adjusts for any errors



---

## Code:

Adding a feedforward term to the integral control requires just one extra line of code

But results in much better tracking

```
while(t < 20):  
  
    if(sin(pi*t/5) > 0):  
        Ref = min(50, Ref - 50*dt)  
    else:  
        Ref = max(100, Ref + 50*dt)  
  
    Hz = 1000000/dT  
  
    I += 0.05*(Ref - Hz)*dt  
    V = I + Ref*5/175  
  
    if(V > 0):  
        Vout.duty_u16(int(V*kv))  
        Dir.value(0)  
    else:  
        Vout.duty_u16(int(-V*kv))  
        Dir.value(1)  
  
    print(t, Ref, Hz, V)  
    t += dt  
    sleep(dt)  
  
print('Stop')  
Vout.duty_u16(0)
```

---

---

## Summary

Motors with analog inputs can be driven fairly easily with a Pi-Pico and a PWM output.

The main thing you need is

- An H-bridge for DC motors or
- An ESC controller for an AC motor.

---

# References

## Pi-Pico and MicroPython

- [https://github.com/geeekpi/pico\\_breakboard\\_kit](https://github.com/geeekpi/pico_breakboard_kit)
- [https://micropython.org/download/RPI\\_PICO/](https://micropython.org/download/RPI_PICO/)
- <https://learn.pimoroni.com/article/getting-started-with-pico>
- <https://www.w3schools.com/python/default.asp>
- <https://docs.micropython.org/en/latest/pyboard/tutorial/index.html>
- <https://docs.micropython.org/en/latest/library/index.html>
- <https://www.fredscave.com/02-about.html>

## Pi-Pico Breadboard Kit

- <https://wiki.52pi.com/index.php?title=EP-0172>

## Other

- <https://docs.sunfounder.com/projects/sensorkit-v2-pi/en/latest/>
  - <https://electrocredible.com/raspberry-pi-pico-external-interrupts-button-micropython/>
  - <https://peppe8o.com/adding-external-modules-to-micropython-with-raspberry-pi-pico/>
  - <https://randomnerdtutorials.com/projects-raspberry-pi-pico/>
  - <https://randomnerdtutorials.com/projects-esp32-esp8266-micropython/>
-