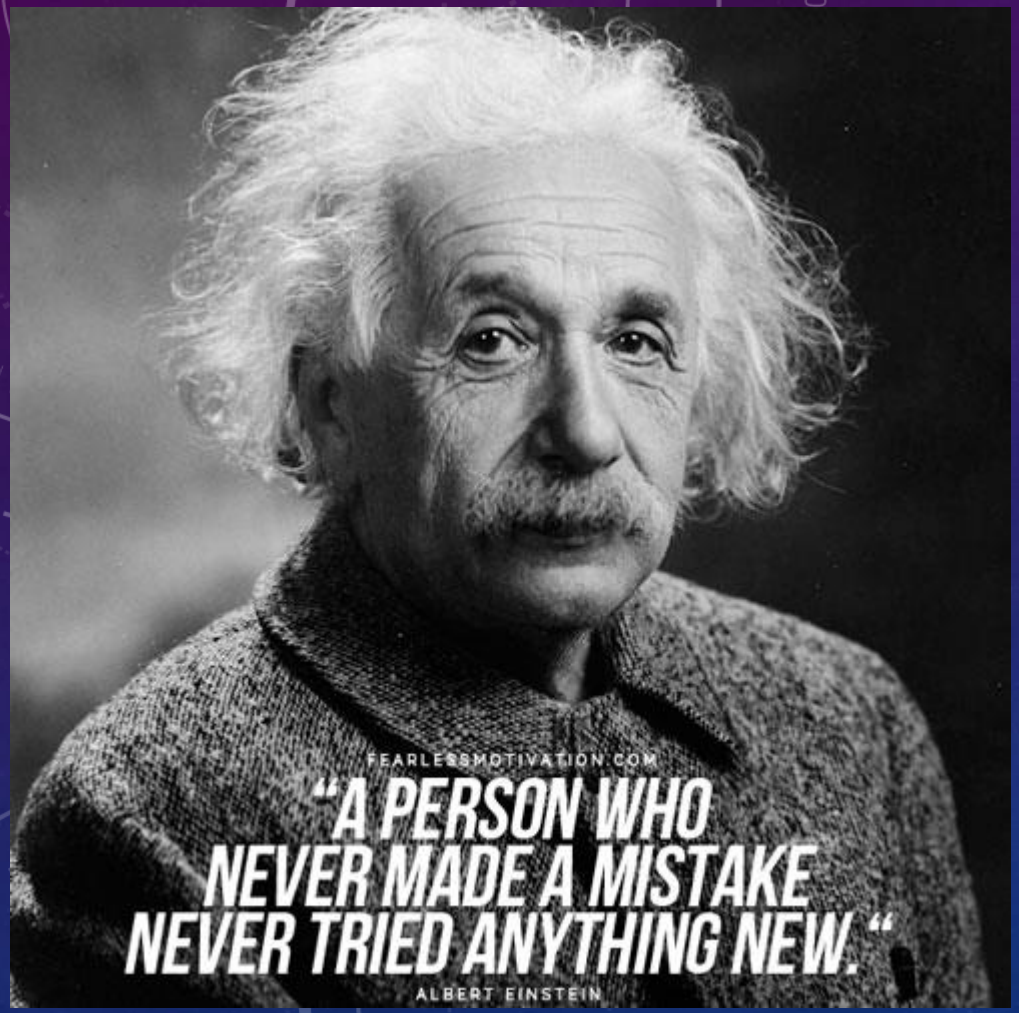
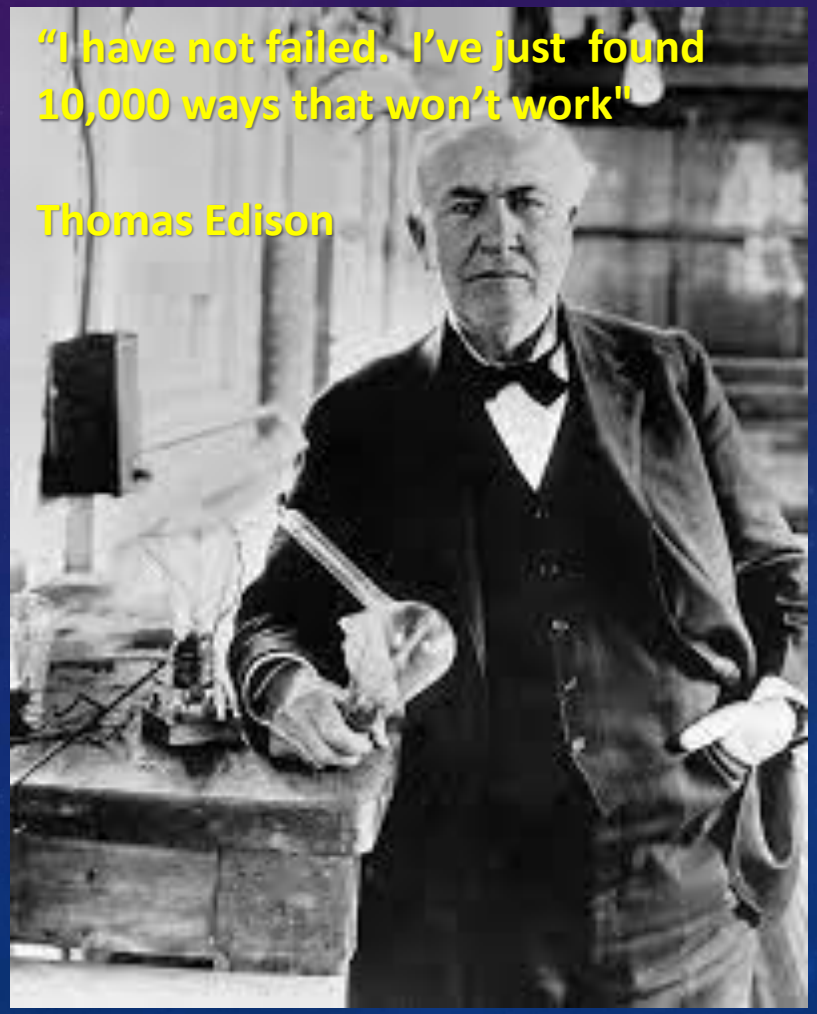
NPN-NMOS TRANSISTOR TRACER

DAVE VE3OOI

MARCH 2023



This Project is a Textbook Example of Making Mistakes



THE PROBLEM

1. You have a load that needs 350mA of current to work properly.
   - ✓ What transistor should you use?
   - ✓ What base resistor do you need to drive 100mA?

2. You need to have a quiescent current of 200mA through a MOSFET.
   - ✓ What bias do you apply to the MOSFET to get 200mA?

3. You need to have a reverse voltage protection diode with lowest voltage drop.
   - ✓ Which diode do you use from a bag of diodes you ordered from Digikey.

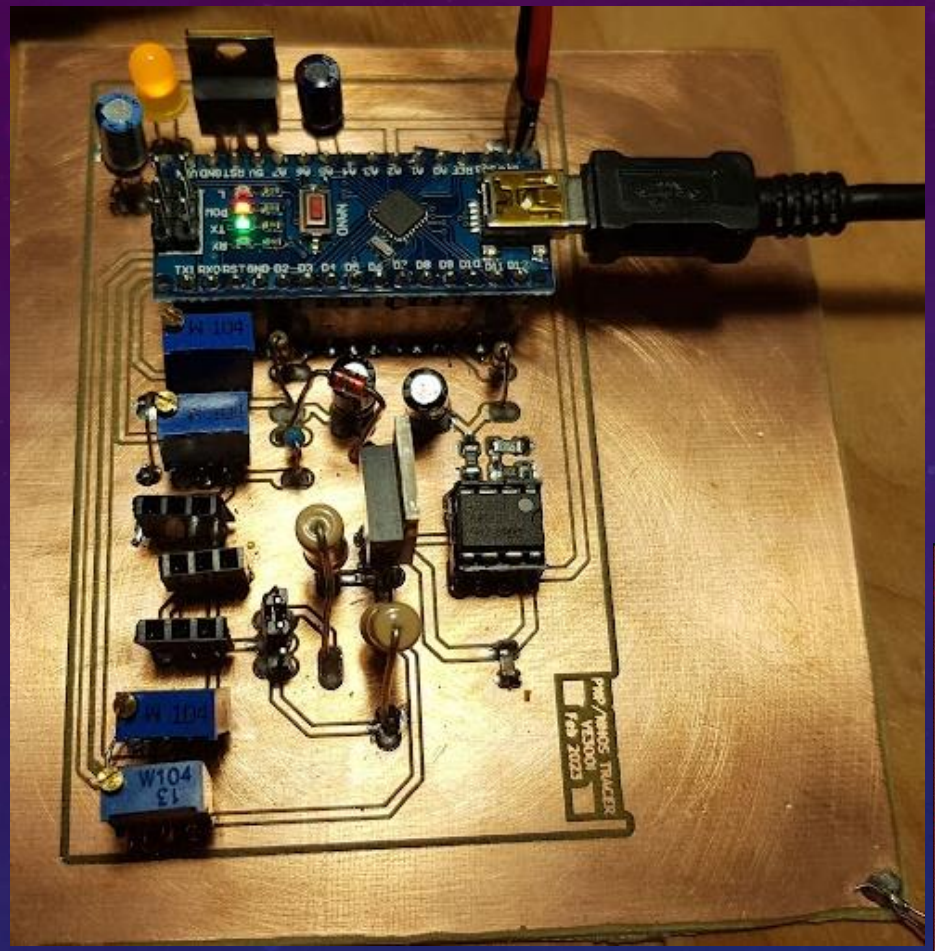
4. You are building a Class A RF amplifier and you need to set the bias point of the transistor so that it 100% linear.
   - ✓ What is the optimum bias point to use?
   - ✓ What base current do you need to get the desired output current?

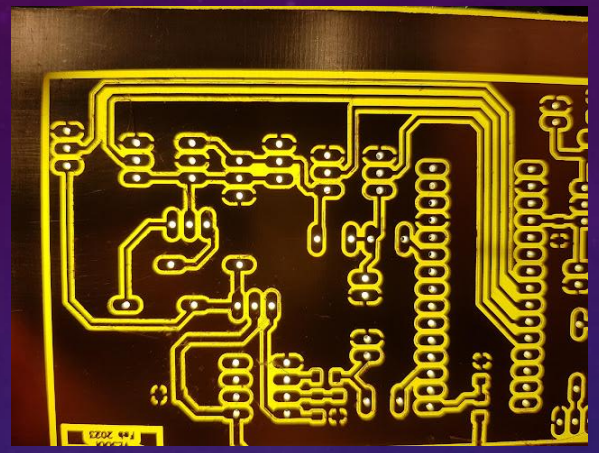
AGENDA

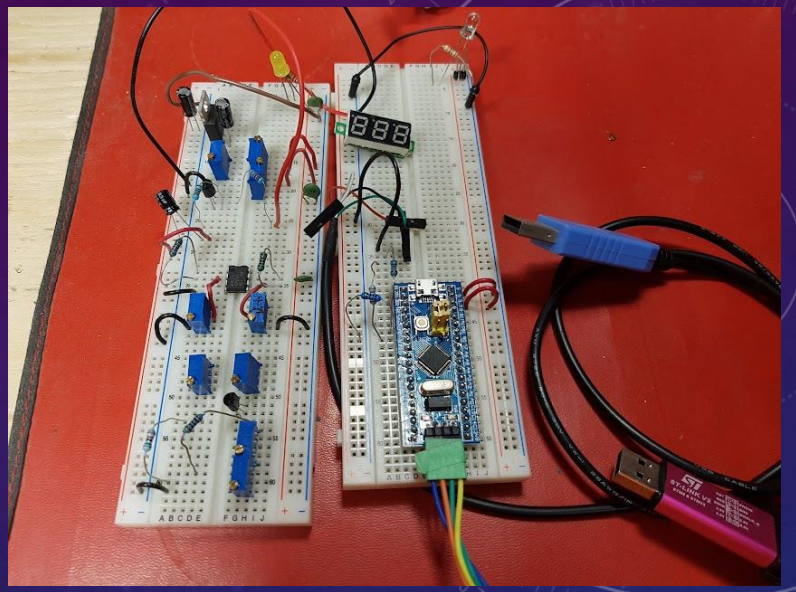
1. The Transistor Tracer Hardware Version 1
2. Python Graphing Program
3. Summary of Various Transistor Curves (Not a Tutorial)
4. Data Sheet Comparison with Transistor Tracer Graphs
   - ✓ Diodes
   - ✓ NPN Transistors
   - ✓ Enhancement Mode MOSFETS
5. Changes in Version 2

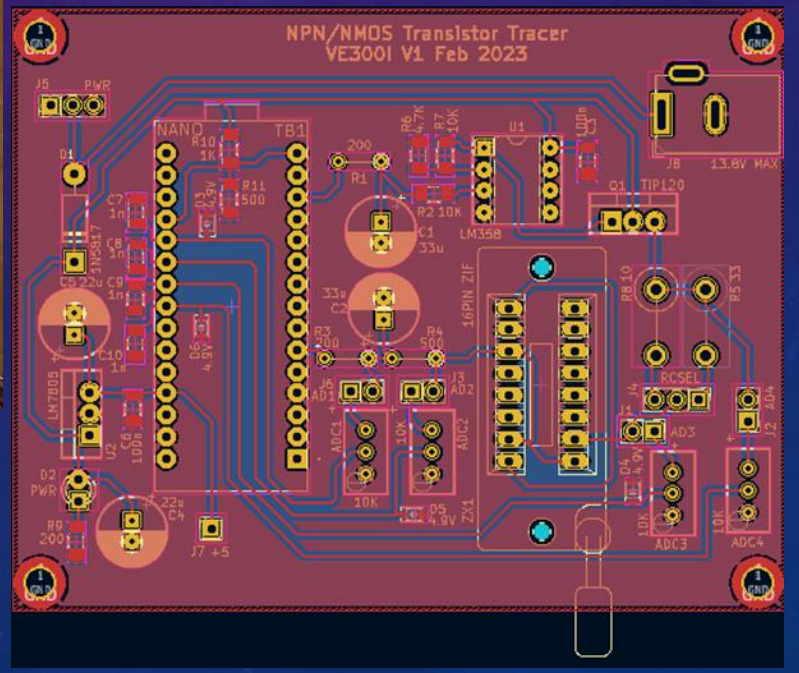
- This is not a Transistor Tutorial. Its assumed you understand how a BJT and MOSFET operates and terms (e.g., Vbe, Vgs, Vce, Vds, etc). Only a short summary if given

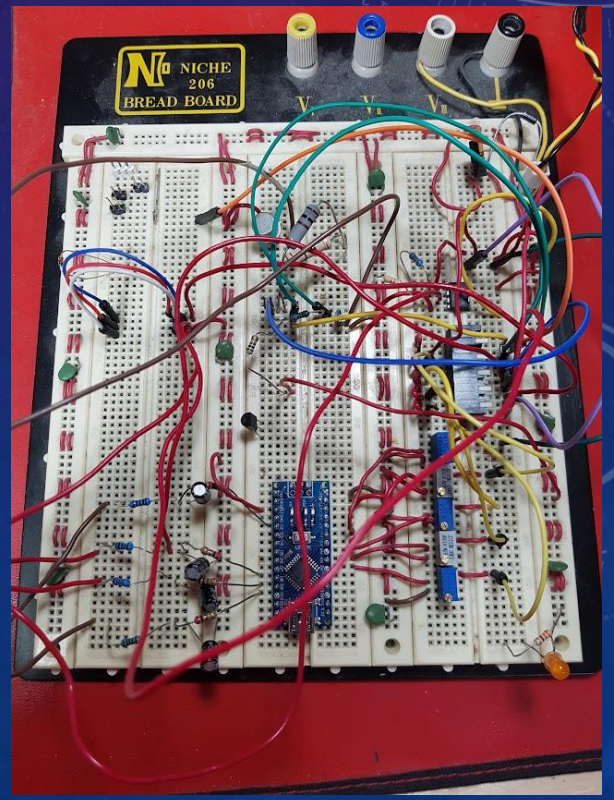
TRANSISTOR TRACER V1 EVOLUTION







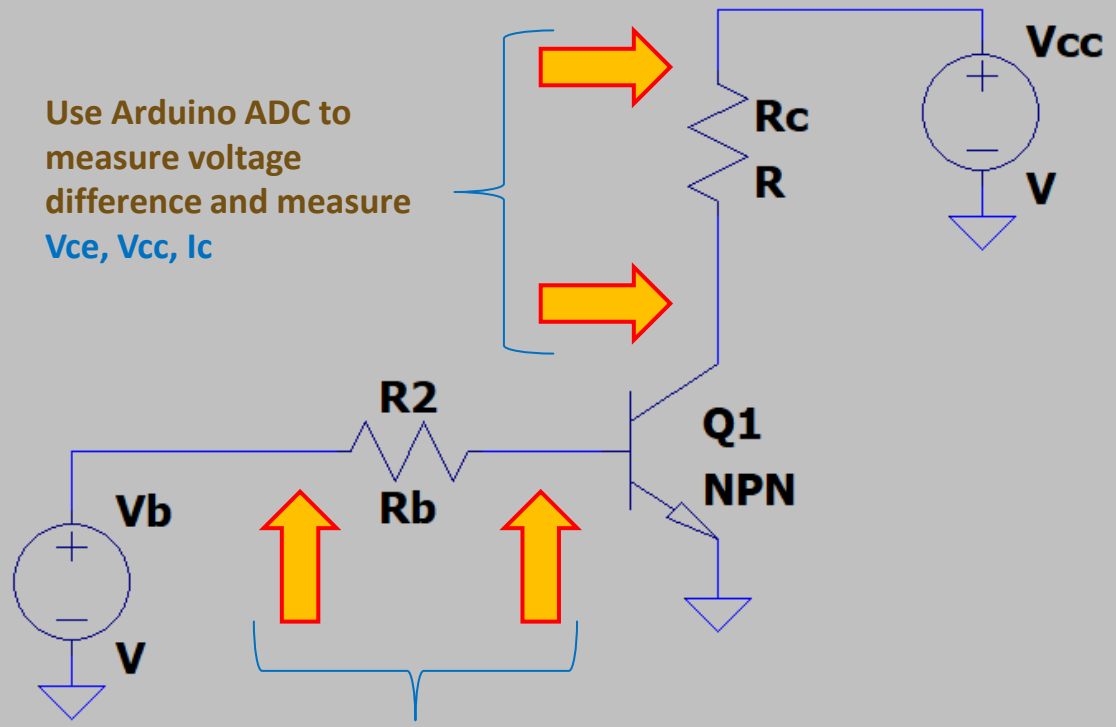




Drop PNP and PCHAN MOS Functionality

HOW IT WORKS

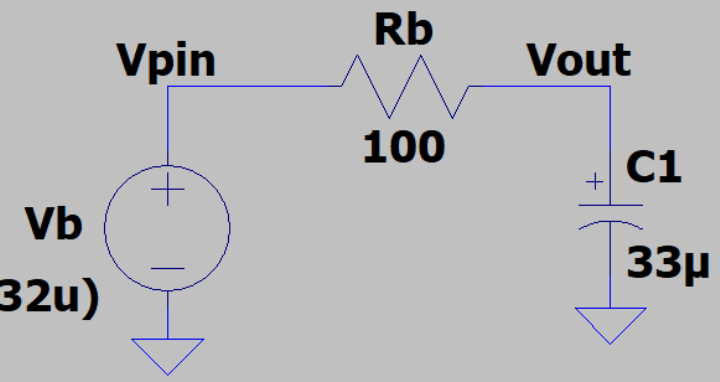
Use Arduino ADC to measure voltage difference and measure Vce, Vcc, Ic

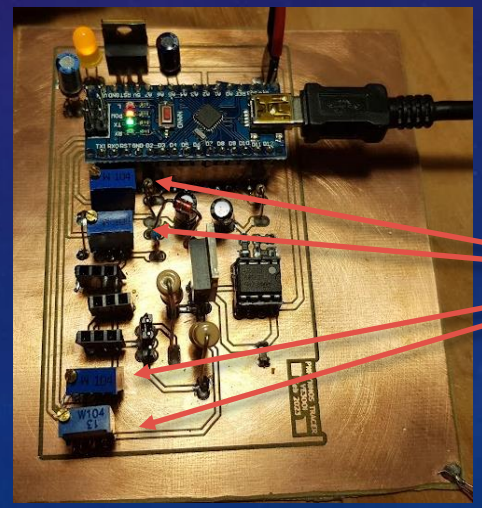


Use Arduino ADC to measure voltage difference and measure Vbe, Ib, Vb

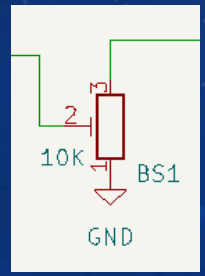
Use Arduino PWM and a LPF to generate a DC voltage

PULSE(0 5 0 1u 1u 16u 32u)

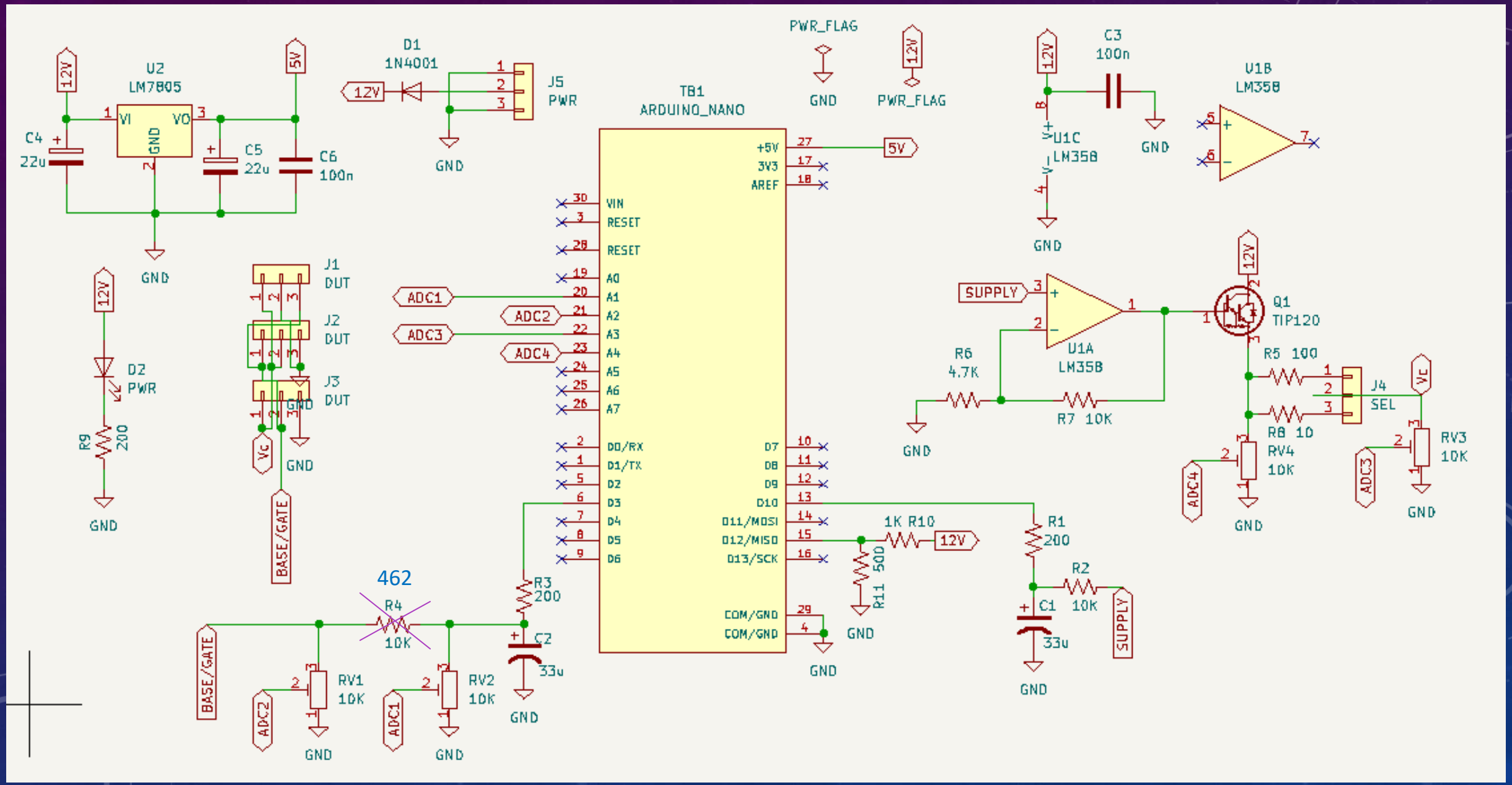




Pots uses as a resistive divider to keep ADC voltages below 5V

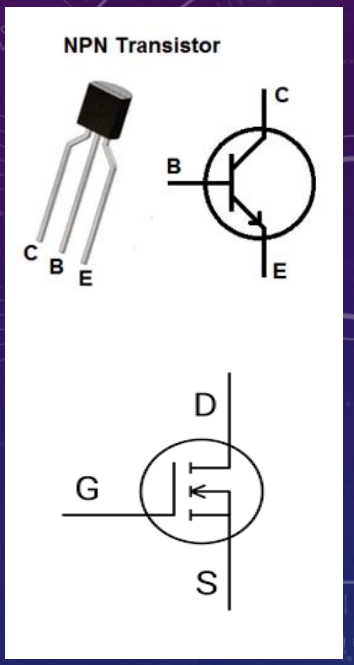


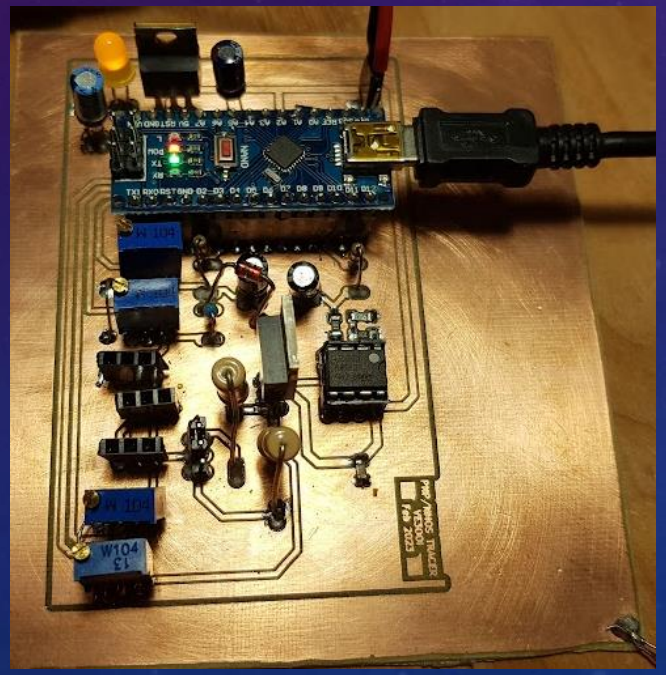
SCHEMATIC

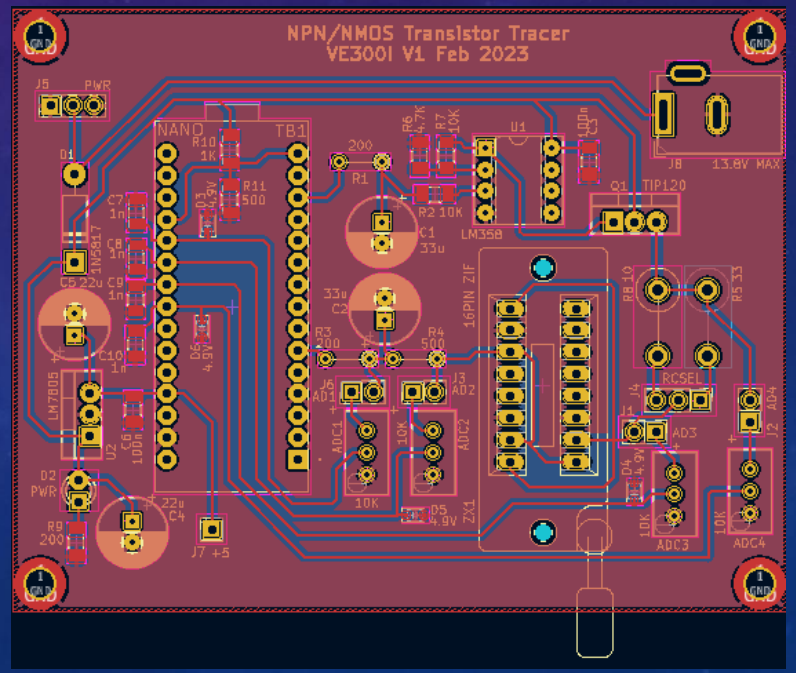

U2
LM7805
12V
5V
GND
C4
22u
C5
22u
C6
100n
D1
1N4001
J5
PWR
PWR_FLAG
TB1
ARDUINO_NANO
+5V
3V3
AREF
VIN
RESET
A0
A1
A2
A3
A4
A5
A6
A7
D0/RX
D1/TX
D2
D3
D4
D5
D6
D7
D8
D9
D10
D11/MOSI
D12/MISO
D13/SCK
COM/GND
U1C
LM358
C3
100n
U1B
LM358
SUPPLY
U1A
LM358
R6
4.7K
R7 10K
Q1
TIP120
R5 100
R8 10
RV4
10K
J4
SEL
RV3
10K
ADC1
ADC2
ADC3
ADC4
J1
DUT
J2
DUT
J3
DUT
Vc
D2
PWR
R9
200
BASE/GATE
462
R4
10K
R3
200
C2
33u
RV1
10K
RV2
10K
1K R10
R11 500
R1
200
R2
10K
C1
33u

VERSION 1: SPEEDS AND FEEDS

- ✓ **NPN** BJTs with EBC, BEC, BCE
- ✓ **NMOS** (N Channel Enhancement Mode MOSFET) with GDS, GSD, DGS
- ✓ Vcc 14V, Max Collector/Drain Voltage: 11@930mA
- ✓ Max Base/Gate voltage: 4.3@7mA





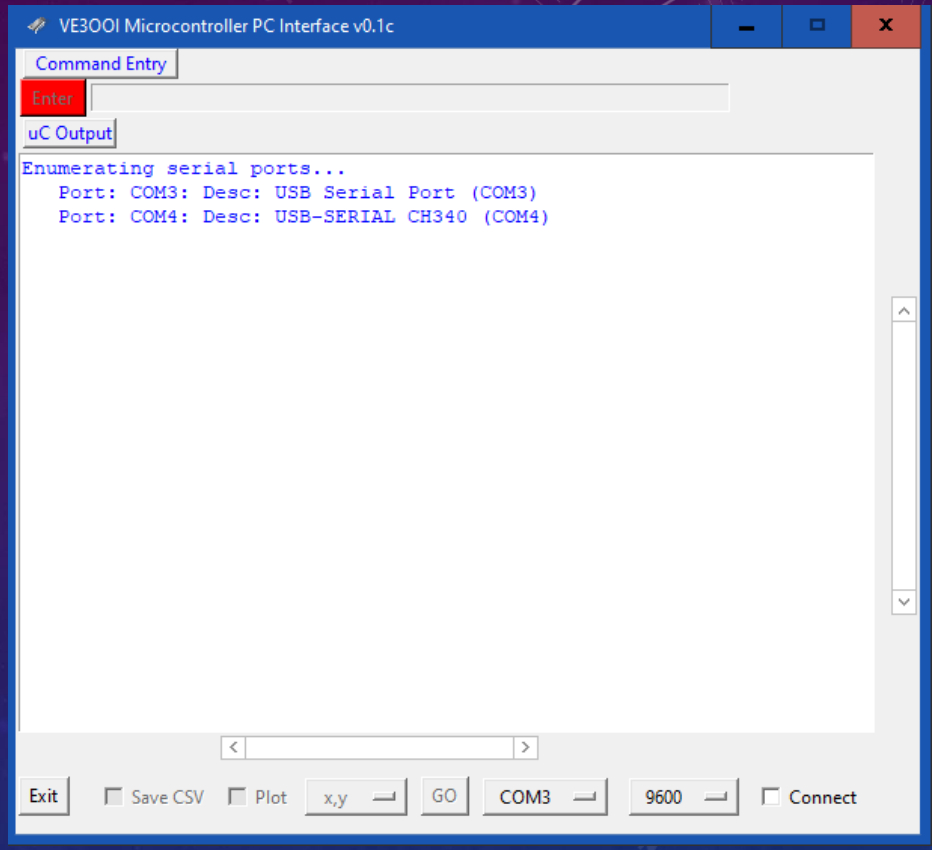


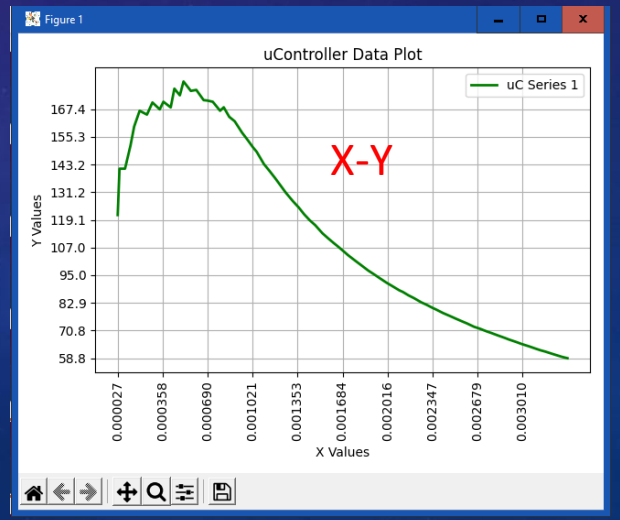
PC INTERFACE – PYTHON MICROCONTROLLER CONSOLE

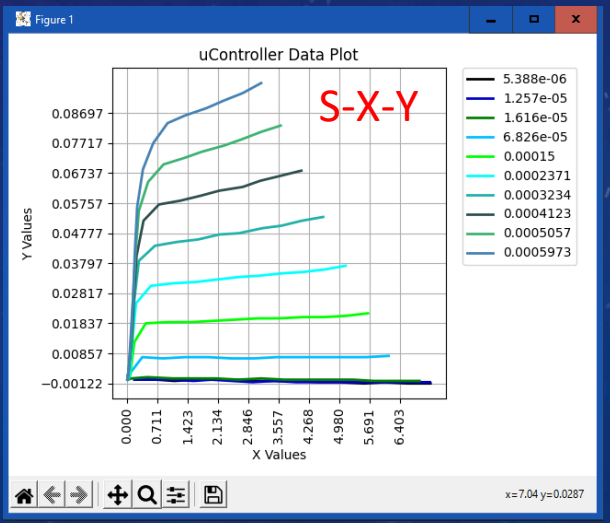
MICROCONTROLLER CONSOLE

- Data acquisition program written in python for PC
  - ✓ https://github.com/drajnauth/uC_Console
  - ✓ Need to install matplotlib, numpy & serial to use Python script
  - ✓ Can run .exe file. Rename .ex_ to .exe 
- Used to acquire and plot data from a microcontroller
- Data is needs to be written to serial console in a specific format
  - Data to be captures has “#” at start of line
  - Finish data set with a “|”
  - Create a new series with a “=“
- Currently supports
  - Plot after all data received: x-y plots, s-x-y series
  - Plot as data arrives (live plot): y, x-y, s-x-y

YOU MUST PROGRAM uC TO SEND DATA





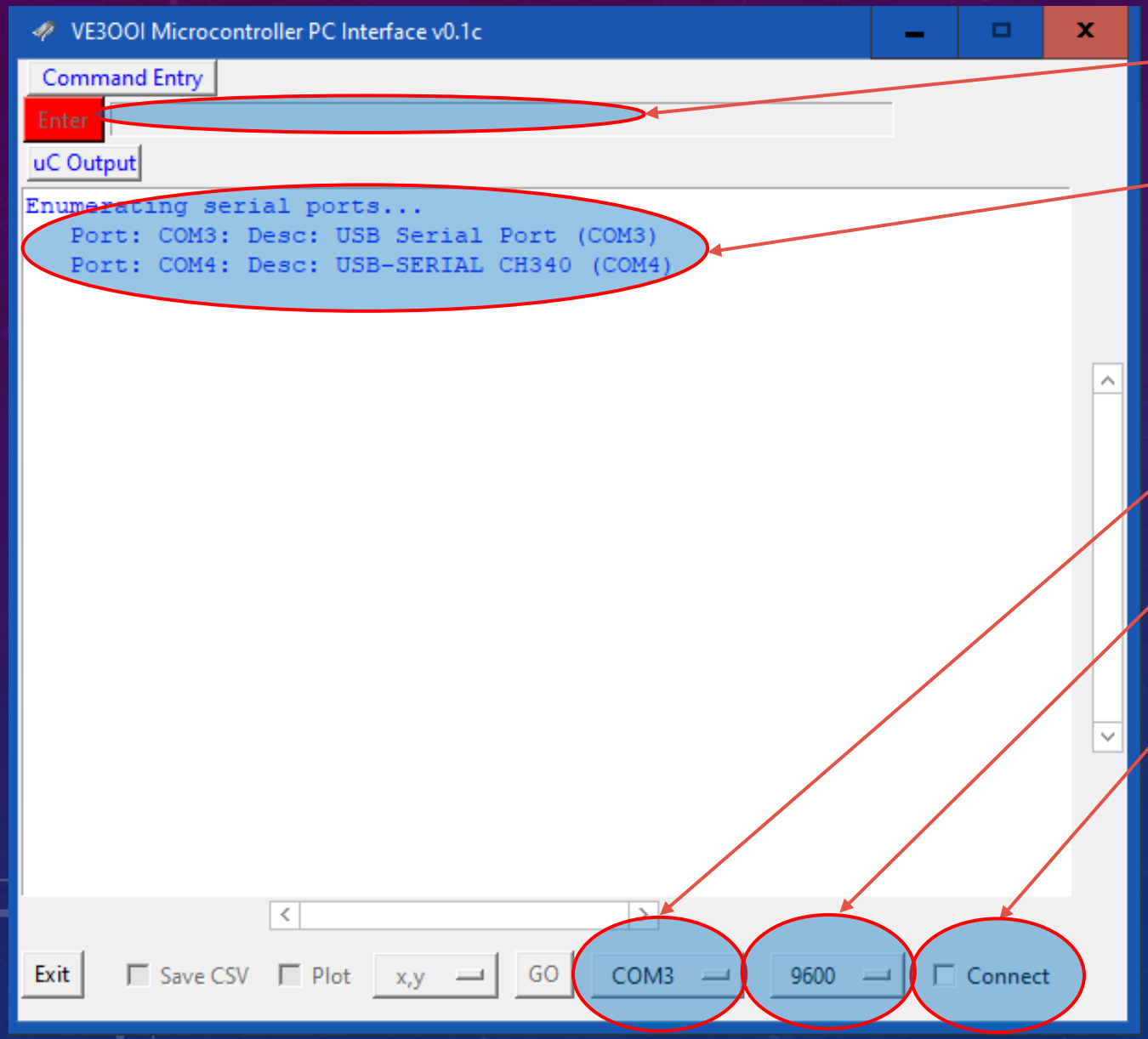


RUNNING THE EXE FILE

Shack_Icon.png	3/30/2023 2:00 PM	PNG File	165 KB
uC_Console_TKv0.1c.ex_	3/30/2023 2:22 PM	EX_ File	31,015 KB
uC_Console_TKv0.1c.py	3/30/2023 2:43 PM	Python File	52 KB
uC_Console_TKv0.1c.pyw	3/30/2023 2:43 PM	Python File (no co...	52 KB

uC_Console_TKv0.1c.exe	3/30/2023 2:21 PM	Application	31,015 KB

MICROCONTROLLER CONSOLE: CONNECTION



Enter command to be sent to uC (same as Arduino Serial Console)

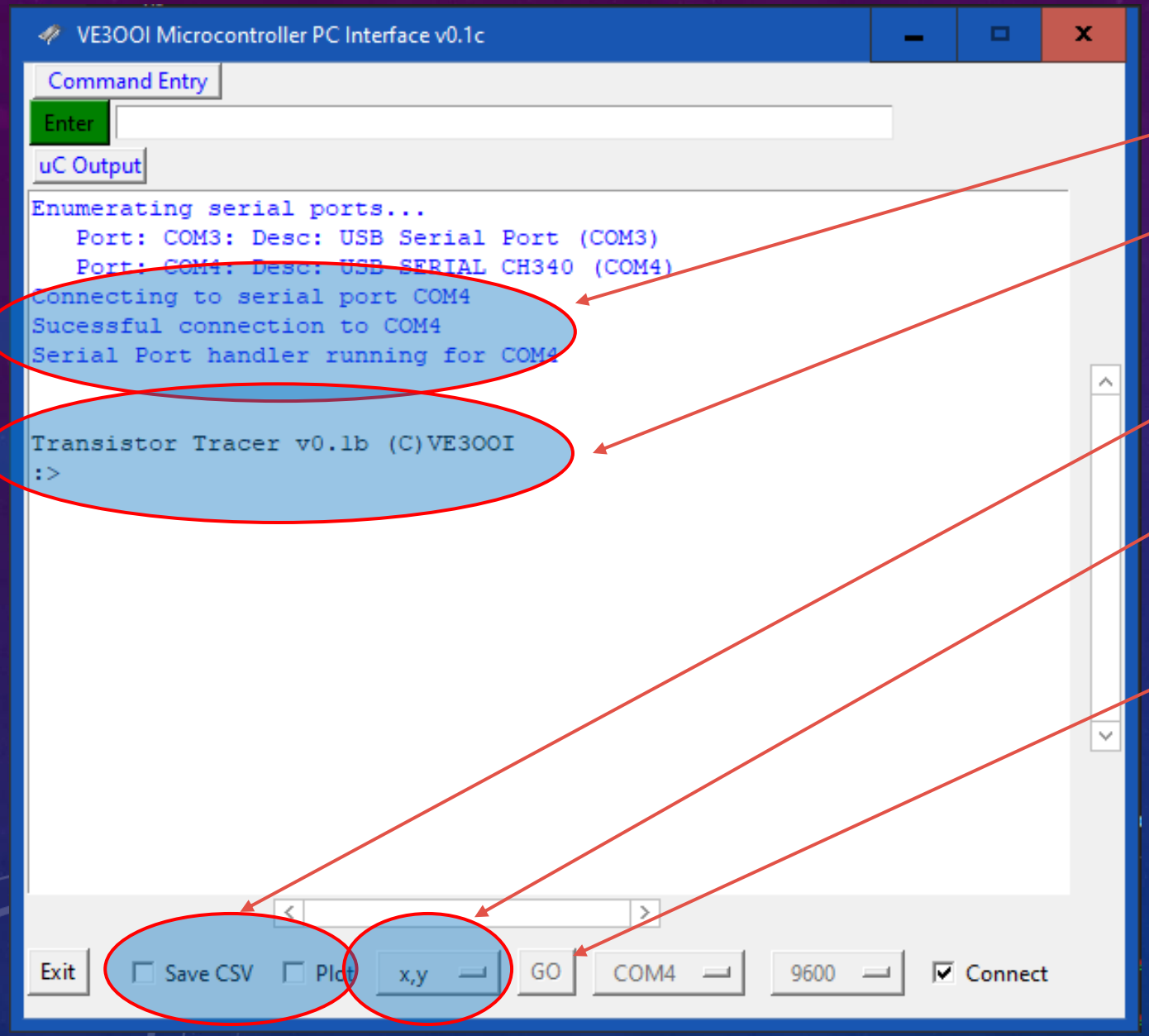
Serial Ports available to connect to

Select com port from drop down

Select baud rate from drop down box

Select the “connect” checkbox to connect to the uC

MICROCONTROLLER CONSOLE: USAGE



Connection successful message

Text from uC is black. Text from program is blue

Select if you want to save data to a csv file or plot the data

Select the type of plot or the data series to save to csv file

When data has been successfully captures, the “GO” button will be enabled and you can press “GO” to generate the plot or save the csv file

HOW IT WORKS

1. The Microcontroller must have a CLI

2. The uC must append “#” to numbers and send other control characters

```
#     // x,y numbers
|     // end control characters
=     // switch control character
      // i.e., new xy plot)
```

```
:> tni32
Rc = 32.00
:> BJT Input Sweep
Sweeping Vb 0.00 V to 4.60 V
Vc/Vd: 4.6000 Vb/Vg 0.0000 - 4.60 inc 0.23
#0.000000000000000000,0.000000000000
#0.188628406524658203 12,0.000000000000
#0.431150627136230468 75,0.000000000000
#0.660199403762817382 81,0.006315693378
#0.714093208312988281 25,0.044209775924
#0.741040134429931640 62,0.077472376823
#0.754513645172119140 62,0.093472108840
#0.781460523605346679 68,0.103577196598
#0.781460523605346679 68,0.111156010627
#0.794933986663818359 37,0.118313789367
#0.808407402038574218 75,0.124629485607
#0.808407402038574218 75,0.130103087425
#0.821880912780761718 75,0.135576665401
#0.821880912780761718 75,0.140208148956
#0.835354328155517578 12,0.145260715484
#0.835354328155517578 12,0.149471163749
#0.835354328155517578 12,0.153260574340
#0.848827743530273437 50,0.157049989700
#0.848827743530273437 50,0.160418343544
#0.862301254272460937 50,0.163365659713
|
:>
```

```
Plotting Enabled for x,y Data
tni32
Rc = 32.00
:> BJT Input Sweep
Sweeping Vb 0.00 V to 4.60 V
Vc/Vd: 4.6000 Vb/Vg 0.0000 - 4.60 inc 0.23
Captured: 0.0, 0.0
Captured: 0.1886284065246582, 0.0
Captured: 0.4311506271362305, 0.0
Captured: 0.6601994037628174, 0.005473584175
Captured: 0.7140932083129883, 0.043788747787
Captured: 0.7410401344299317, 0.077472376823
Captured: 0.7545136451721192, 0.093472118377
Captured: 0.7814605236053467, 0.103998255729
Captured: 0.7814605236053467, 0.111577057838
Captured: 0.7949339866638183, 0.118734836578
Captured: 0.8084074020385742, 0.125050520896
Captured: 0.8084074020385742, 0.130524110794
Captured: 0.8218809127807617, 0.135576665401
Captured: 0.8218809127807617, 0.140629196166
Captured: 0.8353543281555176, 0.145681738853
Captured: 0.8353543281555176, 0.149471163749
Captured: 0.8353543281555176, 0.153681621551
Captured: 0.8488277435302735, 0.1570499897
Captured: 0.8488277435302735, 0.160418367385
Captured: 0.862301254272461, 0.163365659713
Captured 20 data pairs for 0 series.  Press 'GO' to plot
:>
```

HOW IT WORKS

1. The Microcontroller must have a CLI
2. The uC must append “#” to numbers and send other control characters

```
#     // x,y numbers
|     // end control characters
=     // switch control character
      // i.e., new xy plot)
```

```
tno 32
Rc = 32.00
:> BJT Output Sweep
Sweeping Vb 0.00 V to 4.60 V
BJT NPN Output Sweep
Vb 0.0000 - 2.3000
Vc 0.0100 - 4.6000
#0.000000000000,0.000000000000,0.000000000000
#0.000000000000,0.848827743530,-0.000421045351
#0.000000000000,1.724602460861,0.000000000000
=
#0.000013473452,0.000000000000,0.000000000000
#0.000000000000,0.848827743530,0.000421045351
#0.000000000000,1.724602460861,0.000000000000
=
#0.000013473480,0.000000000000,0.000000000000
#0.000013473480,0.835354328155,0.000000000000
#0.000013473480,1.724602460861,-0.000421043491
=
#0.000080840768,0.000000000000,0.000000000000
#0.000040420351,0.026946914196,0.001263136625
#0.000026946902,0.579358625411,0.004631500720
=
#0.000242522192,0.000000000000,0.000000000000
#0.000229048728,0.000000000000,0.000842091083
#0.000161681470,0.107787656784,0.017683911323
=
#0.000431150627,0.000000000000,0.000000000000
#0.000417677164,0.000000000000,0.000842091083
#0.000377256870,0.067367286682,0.019368094444
=
#0.000646725940,0.000000000000,0.000000000000
#0.000633252525,0.000000000000,0.000842091083
#0.000579358673,0.053893828392,0.019789141654
=
|
:>
```

Series Id (not used to plot)

X data point

Y data point

Switch to new XY plot

End of data. Enable plot

CLI MANUAL

ARDUINO SOFTWARE (FIRMWARE)

- This software was compiled using PlatformIO IDE. As is it will NOT compile with Arduino IDE.
- For Arduino IDE, simply rename the main.cpp to an Arduino program name (*.ino). Then create a directory with the same name and store all .cpp and .h files in that directory

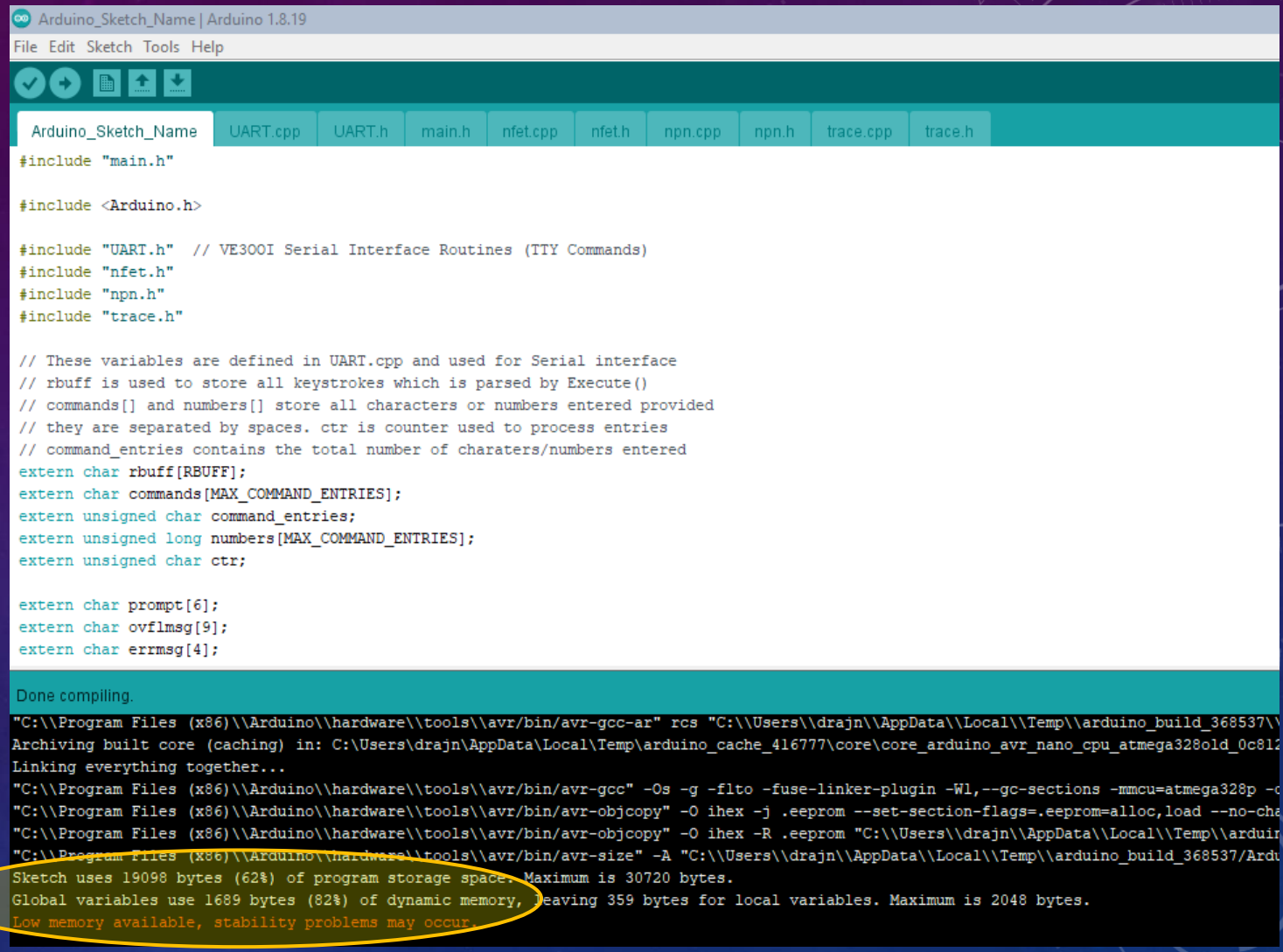
main.cpp	3/10/2023 4:45 PM	CPP File	18 KB
main.h	2/28/2023 9:35 PM	H File	1 KB
nfet.cpp	3/7/2023 7:32 PM	CPP File	5 KB
nfet.h	3/1/2023 5:44 PM	H File	1 KB
npn.cpp	3/6/2023 8:55 PM	CPP File	6 KB
npn.h	3/5/2023 8:14 PM	H File	1 KB
trace.cpp	3/9/2023 6:42 PM	CPP File	13 KB
trace.h	3/10/2023 1:38 PM	H File	4 KB
UART.cpp	2/25/2023 8:49 PM	CPP File	5 KB
UART.h	2/25/2023 8:49 PM	H File	1 KB

Arduino_Sketch_Name	3/27/2023 4:32 PM	File folder

Arduino_Sketch_Name.ino	3/10/2023 4:45 PM	Arduino file	18 KB
main.h	2/28/2023 9:35 PM	H File	1 KB
nfet.cpp	3/7/2023 7:32 PM	CPP File	5 KB
nfet.h	3/1/2023 5:44 PM	H File	1 KB
npn.cpp	3/6/2023 8:55 PM	CPP File	6 KB
npn.h	3/5/2023 8:14 PM	H File	1 KB
trace.cpp	3/9/2023 6:42 PM	CPP File	13 KB
trace.h	3/10/2023 1:38 PM	H File	4 KB
UART.cpp	2/25/2023 8:49 PM	CPP File	5 KB
UART.h	2/25/2023 8:49 PM	H File	1 KB

ARDUINO IDE VS PLATFORM.IO

	Arduino	PlatformIO
RAM	1689	1414
FLASH	23334	19098




```
Linking .pio\build\nanoatmega328\firmware.elf
Checking size .pio\build\nanoatmega328\firmware.elf
Advanced Memory Usage is available via "PlatformIO Home > Project Inspect"
RAM:   [=======   ]  69.0% (used 1414 bytes from 2048 bytes)
Flash: [========  ]  76.0% (used 23334 bytes from 30720 bytes)
========================================================== [SUCCESS] Took 5.92 seconds ==========================================================
*  Terminal will be reused by tasks, press any key to close it.
```

CLI GUIDE: SUMMARY

```
H                              // Help – Command summary

C                              // Display current Calibration
C [R|M|D]                      // Calibrate
D                              // Set default values
E [I|P]                        // Configure EEPROM
M [R]                          // Measure Voltages
P [3|10 [255]                  // Set PWM on pin 3 or pin 10
R                              // Reset
S [V] [B|C] [O] [mV]           // Set voltage on Base or Vcc pin.
S [I] [uA]                     // Set current on Base
S [U] [D|M|U]                  // Set default unit to decimal, Milla or Micro for currents
S [M] [B|C] [min] [max]        // Set min max voltage to sweep
S [R] [B|C] [R]                // Set Rb or Rc resistor values
S [P] [B|C] [n]                // Set number of data point in sweep between max and min
T [NI|NIV|NO|NB|NBC|NR|NG]     // Trace NPN Transistor
T [FI|FIV|FO|FK|FG|FR]         // Trace N Chan MOSFET
T [DI|DO]                      // Trace Diode
```

CLI GUIDE: CALIBRATION

Display current Calibration values

```
C
```

Connect the Tracer to Vcc with empty socket (nothing connected) to determine the voltage sweep range for base/gate voltage and collector/drain voltage. It also calculated the conversion constant to convert collect/drain (pin 10) PWM voltage to actual output Vcc voltage delivered to DUT

```
C M
```

Connect the Tracer to Vcc with empty socket (nothing connected) to determine the voltage sweep range for base/gate voltage and collector/drain voltage. It also calculated the conversion constant to convert collect/drain (pin 10) PWM voltage to actual output Vcc voltage delivered to DUT

```
C D
```

Connect the Tracer to Vcc with empty socket (nothing connected). It generates a PWM voltage on collector/drain pin defined by a calibration voltage constant (see below) and then measures and displays all ADC values 10 times. This is one way to calibrate ADC pots. Press the reset button or enter the R command to stop the display.

```
C R
```

BUILD TIPS

1. **DO NOT CONNECT ARDUINO AND POWER UP BEFORE CALIBRATION IS DONE**
2. Do you best to get parts as labelled.
3. POTS show incorrect values. They should all be 100K
4. The TIP120 can be replaced with a 2SC5706 power transistor
5. The 1N5817 can be replaced with any 1 Amp Shottky Diode
6. The board can be powered from either J8 or J5. Maximum voltage is 14 V. Exceeding this could damage you Arduino
7. R1, R3, R6 and R7 resistors cannot be substituted.
8. Capacitors C1, C2, C7, C8, C9 and C10 cannot be substituted. Other capacitors can used similar values.
9. The D3, D4, D5, and D6 4.9V Zener diodes are optional and are there to protect Arduino ADC from overvoltage
10. Resistors R10 and R11 form a voltage divider to read Vcc voltage. If you changes these values ensure the ratio will allow 14V to delver about 4.8V to the Arduino. Measure and note the values of the resistors you use. You will need to enter them in the Arduino code
11. R4, R5 and R8 set the maximum current to the base or to the collector. You can substitute values here. Keep R4 close the 500R (i.e. within 10 or 20R). R5 and R8 must be 2W resistors. They dissipate several hundred milliamps. Measure and note the values of the resistors you use. You will need to enter them in the Arduino code

12. Solder all SMT parts
13. Solder all through hole components from the underside
14. Arrange the pots such that turning clockwise increases the resistance between the wiper and the grounded leg.
15. Once your finished remove all jumpers from header pins. Do NOT connect the Arduino. Power up and check voltages. Proceed to calibration.

CALIBRATION & CONFIGURATION FOR VERSION 1

1. **DO NOT CONNECT ARDUINO AND POWER UP BEFORE CALIBRATION IS DONE**
2. Before powering up Arduino, you need to calibrate pots to not damage the Arduino.
3. Remove Arduino. Remove jumpers from all trim pots jumper headers pins.
4. Measure the voltage at the +5V terminal pin (J7).
5. Connect a jumper wire from +5V terminal to each pot's jumper "+" pin (J1, J2, J6, J7). "+" is marked on the silkscreen
6. Adjust each pot until you get 1.75V output (exactly) at the pot's wiper. Use one consistent VOM.
7. Insert Arduino and load the code. Power up.
8. Connect via TTY console to Arduino. Enter the C R command to display all ADC values from each pot
9. Note that each pot has ADC1, ADC2, ADC3, ADC4 marked on the silkscreen. Each jumper associate with the pot is marked as AD1, AD2, AD3, AD4. This corresponds to the ADC number on the Arduino.
10. Connect 5V to the same jumper in step 3. Note the ADC number of the jumper and pot.
    a. Adjust the pot's wiper (very fine adjust needed) until the corresponding ADC displayed by the Arduino is **EXACTLY** 364. Repeat for all jumpers/pots.
    b. Enter the R command to reset the Arduino when done
11. Complete and record the following calculations for each pot:
    a. Calculate VOLTAGE_DIVIDER = V(wiper)/V(applied) e.g., 1.75/5 = 0.357
    b. Calculate ADC_CONVERSION = ADC_Value/V(wiper) e.g., 364/1.75 = 208.0
    c. Calculate VOLTAGE_CONVERSION = VOLTAGE_DIVIDER * ADC_CONVERSION e.g., 0.357*208 = 78.0
12. Take the average VOLTAGE_CONVERSION and update the `#define ADC_CONVERSION ((float)78.0)` in the defines.h file
13. You need to measure the resistance of the Base and Collector resistors. Update the following defines in the defines.h file

```
#define RC_RESISTORHIGH 32
#define RC_RESISTORLOW 10
#define RB_RESISTORLOW 462
```

14. Reset the Arduino and measure the max voltage for the base and collectors. Ensure no jumpers are connected. Enter the P 3 255 command and measure and record the voltage after the 200R resistor connected in D3 pin. Reset the Arduino and repeat for pin 10. Enter the P 10 255 and measure and record the voltage after the 200R resistor connected to D10 pin. Update the following defines in the defines.h file

```
1. #define MAX_VC_VOLTAGE 4.6
2. #define MAX_VB_VOLTAGE 4.6
```

15. Compile and load the Arduino program before continuing. The above defines must be in place.
16. Complete the following. DO NO RESET OR POWER OFF ARDUINO UNTIL ALL STEPS ARE COMPLETE. Failure will result in redoing below
    a. Set the base resistor value using the S R B nnnn command where nnnn is the resistor value. E.g., S R B 462
    b. Set the jumper J4 (RCSEL) to the resistor for the sweep. Use the S R C nnnn command to define the Collector resistor nnnn that the jumper selected. E.g., S R C 10
    c. With Vcc applied and nothing connected to the ZIF socket (socket is empty), run the C M command to have the Arduino figure out the max voltages available to it during a sweep. If you change Vcc, this needs to be repeated.
    d. Update the EEPROM with the configuration by entering the E W command. Once completed, the Arduino will power up with these values.
17. Your transistor curve trace (V1) is now ready.

OPERATION: SWEEPS

1. The transistor tracer sweeps a set of PWM voltages.
    1. There is a base/gate PWM voltage (Pin 3) that is applied directly to the transistor under test.
    2. There is a Vcc PWM voltage (Pin 10) that is amplified by an opamp and a high-power transistor
2. Since the base/gate PWM voltage is fed directly to the transistor under test, the base/gate voltage is essentially the output voltage to the transistor
3. However, since the Vcc PWM voltage is amplified, the output voltage fed to the collector/drain of the transistor under test is different.
4. When sweeping Vcc values, only the PWM voltage is used. **When you set voltage ranges, you are setting the PWM voltage that the Arduino outputs.**
5. You can manually set the output voltage supplied to the transistor under test using the S V command. However, this is a one-time setting. Its not used in a sweep. **BE CAREFULL SETTING VOLTAGES MANUALLY** – you could destroy a resistor or the transistor under test if you exceed its maximum ratings

REMEMBER: When you set voltage ranges, you are setting the PWM voltage that the Arduino outputs. You are NOT setting the output voltage fed to the transistor under test

OPERATION: NPN SWEEPS

1. The T command is used to complete Traces. The T N command is used to complete NPN Transistor traces
2. The following traces uses the configured min and max voltages. You can adjust sweep voltages and number of data points (resolution) to be taken during a sweep. Use the S M command to adjust min & max voltages for and S P to set data points.
3. The following below outlines the various traces:

I. Input Trace: T N I
   - ✓ This command sets the Vcc voltage to the maximum available voltage and then sweeps the base voltage from configured min to max at define inc (resolution). It prints out Vbe vs Ic

II. Input Trace: T N I V
   - ✓ This command is the same as above except it prints out Vbe vs Vce

III. Output Trace: T N O
   - ✓ This command sweeps the base voltage from configurated min to max with inc. For each base voltage, it then sweeps Vcc from configured min to max at defined inc. For each base voltage it print out Vce vs Ic in the form Ib, Vce, Ic

IV. Beta Trace: T N B
   - ✓ This command sets the Vcc voltage to the maximum available voltage and then sweeps the base voltage from configured min to max at define inc (resolution). It calculates transistor beta and prints out Ib vs Beta

V. Beta Trace: T N B C
   - ✓ This command is the same as above except it prints out prints out Ic vs Beta

VI. Beta Trace: T N R
   - ✓ This command sets the Vcc voltage to the maximum available voltage and then sweeps the base voltage from configured min to max at define inc (resolution). It calculates transistor re (intrinsic resistance) and prints out Ic vs re

VII. Beta Trace: T N G
   - ✓ This command sets the Vcc voltage to the maximum available voltage and then sweeps the base voltage from configured min to max at define inc (resolution). It calculates transistor transconductance and prints out Ic vs g_m

OPERATION: N-CHANNEL ENHANCEMENT MODE MOSFET SWEEPS

1. The T command is used to complete Traces. The T F command is used to complete N channel Mosfet traces
2. The following traces uses the configured min and max voltages. You can adjust sweep voltages and number of data points (resolution) to be taken during a sweep. Use the S M command to adjust min & max voltages for and S P to set data points.
3. The following below outlines the various traces:

I. Input Trace: T F I
   - ✓ This command sets the Vcc voltage to the maximum available voltage and then sweeps the gate voltage from configured min to max at define inc (resolution). It prints out Vgs vs Id

II. Input Trace: T F I V
   - ✓ This command is the same as above except it prints out Vgs vs Vds

III. Output Trace: T F O
   - ✓ This command sweeps the base voltage from configurated min to max with inc. For each gate voltage, it then sweeps Vcc from configured min to max at defined inc. For each gate voltage it print out Vds vs Id in the form Id, Vgs, Id

IV. Intrinsic Resistance Trace: T F R
   - ✓ This command sets the Vcc voltage to the maximum available voltage and then sweeps the gate voltage from configured min to max at define inc (resolution). It calculates transistor re (intrinsic resistance) and prints out Id vs Rds

V. Transconductance Trace: T F G
   - ✓ This command sets the Vcc voltage to the maximum available voltage and then sweeps the gate voltage from configured min to max at define inc (resolution). It calculates transistor transconductance and prints out Id vs g_m

VI. Transconductance Trace: T F G
   - ✓ This command is the same as above except it it calculates transistor transconductance constant K and prints out Id vs K

OPERATION: DIODE SWEEPS

1. The T command is used to complete Traces. The T D command is used to complete diode traces
2. The following traces uses the configured min and max voltages. You can adjust sweep voltages and number of data points (resolution) to be taken during a sweep. Use the S M command to adjust min & max voltages for and S P to set data points.
3. Note that the diode **MUST** be oriented correctly in the socket.
4. The following below outlines the various traces:

I. Input Trace: T D I
   - ✓ The diode must be connected between the base/gate socket and ground. The cathode is connected to emitter/source socket (ground). For a Zener diode, its reversed, the cathode is connected to the base. This command sets the Vcc voltage to the maximum available voltage and then sweeps the base/gate voltage from configured min to max at define inc (resolution). It prints out Vb vs Id

II. Output Trace: T D O
   - ✓ The diode will need a resistor connected between the cathode and the emitter/source socket (ground). The resistor is selected to NOT exceed the maximum current the diode can handle. The resistor also protect the Arduino pin used for base PWM output (pin 3). Failure to add a resistor could cause damage to the Arduino. This command set the base/gate voltage Arduino pin to zero to sink current. It then sweeps the Vcc voltage from configurated min to max with inc. For each Vcc voltage, it measure the voltage drop across the collector/drain to the base/gate sockets along with current. It print out Vf vs I

Special Jig Required for Output Seep

OPERATION: CUSTOM SWEEPS

1. The S command is used to set trace values.
2. For base/gate voltage there is a min voltage, max voltage
3. Its sets the PWM voltage output from the Arduino. IT DOES NOT SET THE ACTUAL OUTPUT VOLTAGE DELIVERED TO THE DUT
4. Values MUST be entered as milla volts (mV) e.g., 4V entered at 4000
5. The following below outlines the various configuration options:

I. Base/Gate Sweep: S M B xxxx yyyy
   - ✓ This command sets the minimum voltage (xxxx) and the maximum voltage (yyyy) for the base/gate PWM pin. The increment use for trace is automatically calculated to the number of data point defined. Default is min: 0 and maximum determined by C M calibration command.

II. Vcc Sweep: S M C xxxx yyyy
   - ✓ This command sets the minimum voltage (xxxx) and the maximum voltage (yyyy) for the Vcc PWM pin. The increment use for trace is automatically calculated to the number of data point defined. Default is min: 0 and maximum determined by C M calibration command.

You need to enter the E W command to update the EEPROM configuration values or else these definition will be lost at next restart of the Arduino

OPERATION: CUSTOM SWEEPS

1. The S command is used to set trace values.
2. You can configure the number of data point to be taken during a typical sweep. Some sweeps such as the beta, gm, K and Rds and re sweeps typically take 25% of the increment.
3. Example:
   1. If the min is 0.01V and the max is 4.6V
   2. If you set the number of point to 10, then the increment uses during a sweep is (4.6 – 0.01)/20 = 0.45
   3. Therefore, duing a sweep, 10 measurements are taken with each increasing by 0.45V from min to max.
4. The following below outlines the various configuration options:

I. Base/Gate Sweep: S P B xxxx

- ✓ This command sets the number of data points to measure during a sweep of the base/gate voltage. The increment used for sweep is automatically calculated. Default is 10 data points for base/gate.
- ✓ The range of allowed data points is defined in the defines.h file as:

```
#define MIN_NUM_VB_VOLTAGE 5
#define MAX_NUM_VB_VOLTAGE 20
#define NUM_VB_VOLTAGE 10       // Default number of points
```

II. Vcc Sweep: S P C xxxx

- ✓ This command sets the minimum voltage (xxxx) and the maximum voltage (yyyy) for the Vcc PWM pin. The increment use for trace is automatically calculated to the number of data point defined. Default is 15 data points for Vcc.
- ✓ The range of allowed data points is defined in the defines.h file as:

```
#define MIN_NUM_VC_VOLTAGE 5
#define MAX_NUM_VC_VOLTAGE 20
#define NUM_VC_VOLTAGE 15       // Default number of points
```

You need to enter the E W command to update the EEPROM configuration values or else these definition will be lost at next restart of the Arduino

OPERATION: EEPROM MANAGEMENT

The E command is used to manage configuration values stored in the EEPROM.

- Configuration values stored in the EEPROM will be restored whenever the Arduino is power on.
- The following below outlines the various configuration options:

I. Read EEPROM: E R
   - ✓ This command reads and overwrite the current running configuration with the configuration stored in the EEPROM. Any unsaved changes will be lost (i.e., changes to voltage ranges, points, resistor vales, max voltages, etc)

II. Write EEPROM: E W
   - ✓ This command write the current configuration to the EEPROM. It overwrites the EEPROM.

III. Initialize EEPROM: E I
   - ✓ This command sets the default configuration values, calculates the max voltages and then write the default configuration to the EEPROM. Think of this as a factory reset.

OPERATION: MISC MANAGEMENT

There are a few other MISC command that can be used:

I. Set Voltages: S V B xxxx or S V C xxxx
   - ✓ This set the PWM voltage at pin 3 (base voltage) and pin 10 (Vcc voltage). The voltage is given by xxxx and is in milla volts (i.e., mV). If O (Oh and not zero) is added after the B or C then, the Arduino tries to set the output voltage delivered to the transistor under test. Note that this value could be off by up to 0.5 volts.

II. Set PWM: P xx yyy
   - ✓ This command manually sets the PWM output of pin 3 or pin 10 of the Arduino and is used for troubleshooting. Xx is the pin number (3 or 10) and yyy is the PWM value (1-255). A higher PWM values will output a higher voltage.

III. Reset Arduino: M
   - ✓ This reads and display all the voltages and currents. It assumed you have manually set a voltage using the P or V commands

IV. Reset: R
   - ✓ This resets the Arduino. Its not the same as a power on.

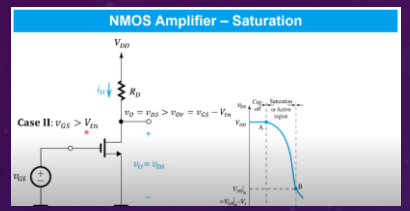
V. Set Defaults: D
   - ✓ This command sets the default parameter to the running config. It does not change the EEPROM config.

VI. Set current units for graph: S U D or S U M or S U U
   - ✓ This command sets units for current values produced. S U D is used for normal decimal, S U M is used for milla and S U U is used for micro

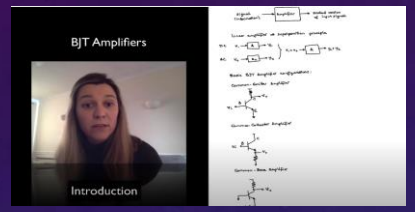
MEASUREMENTS OF NPN BJT TRANSISTORS

IF YOU ARE LOOKING FOR A TUTORIAL…

W2AEW: Back to Basics: Basics of the Bipolar Transistor - NPN and PNP
https://www.youtube.com/watch?v=IrAUC4ZoNYY&t=1263s



Microelectronic Prof. Tony Chan Carusone: Transistor Amplifier Basic Principles
https://www.youtube.com/watch?v=IrAUC4ZoNYY&t=1263s



Mateo Aboy: Series of Videos on BJT and FET Amplifiers
https://www.youtube.com/playlist?list=PLZvLSclgk4yJkRHB_eJ10zo2exTY_irR0

Electronics with Professor Fiore: Semiconductor Devices
https://www.youtube.com/playlist?list=PLxuejeK2BP_dtn1ijj1lZKrIs2gHPPsyp

MOS FET Modification Transistor Curve Tracer HeathKit
https://www.youtube.com/watch?v=rFmDUPVdYSc&t=4953s

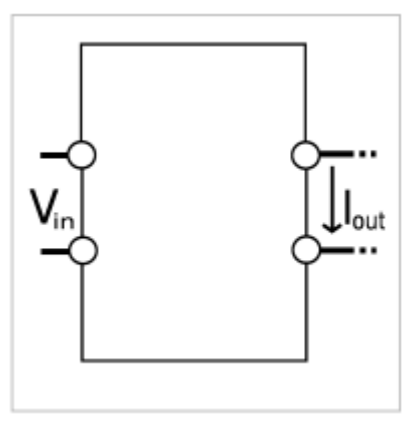
TRANSCONDUCTANCE

Transconductance (for **transfer conductance**), also infrequently called **mutual conductance**, is the electrical characteristic relating the current through the output of a device to the voltage across the input of a device. Conductance is the reciprocal of resistance.

Transconductance is a contraction of *transfer conductance*. The old unit of conductance, the mho (ohm spelled backwards), was replaced by the SI unit, the **siemens**, with the symbol **S** (1 siemens = 1 ampere per volt).

Transconductance is very often denoted as a conductance, g_m, with a subscript, m, for *mutual*. It is defined as follows:

$$g_m = \frac{\Delta I_{\text{out}}}{\Delta V_{\text{in}}}$$



A voltage applied at the input causes a current at the output.
Transconductance is a measure of the slope. It's like a "transfer function"

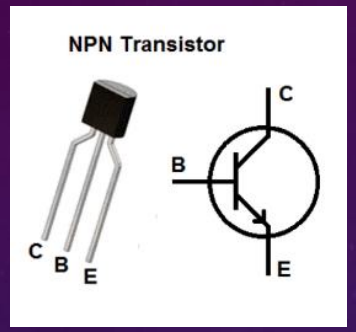
This is a very important concept for MOSFETS

It is used with NPN Transistors, but they are more a current device

A transfer function is a convenient way to represent a linear, time-invariant system in terms of its input-output relationship. It is obtained by applying a Laplace transform to the differential equations describing system dynamics, assuming zero initial conditions. In the absence of these equations, a transfer function can also be estimated from measured input-output data.

TRANSISTOR OPERATION

NPN



Feed Small Current into Base

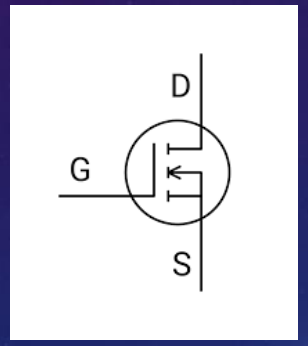
NPN TRANSISTOR

β

Get a LARGE current between Collector and Emitter

Datasheets today use h_{fe} instead of Beta. Beta sounds "cooler"

N-CH MOSFET



Feed Voltage above Threshold

NMOS TRANSISTOR

g_m

Get a LARGE current between Drain and Source

IF YOU ARE LOOKING FOR A TUTORIAL...

W2AEW: Back to Basics: Basics of the Bipolar Transistor - NPN and PNP
https://www.youtube.com/watch?v=IrAUC4ZoNYY&t=1263s

Microelectronic Prof. Tony Chan Carusone: Transistor Amplifier Basic Principles
https://www.youtube.com/watch?v=IrAUC4ZoNYY&t=1263s

Mateo Aboy: Series of Videos on BJT and FET Amplifiers
https://www.youtube.com/playlist?list=PLZvLSclgk4yIkRHB_eJ10zo2exTY_irR0

Electronics with Professor Fiore: Semiconductor Devices
https://www.youtube.com/playlist?list=PLxuejeK2BP_dtn1ijj1lZKrIs2gHPPsyp

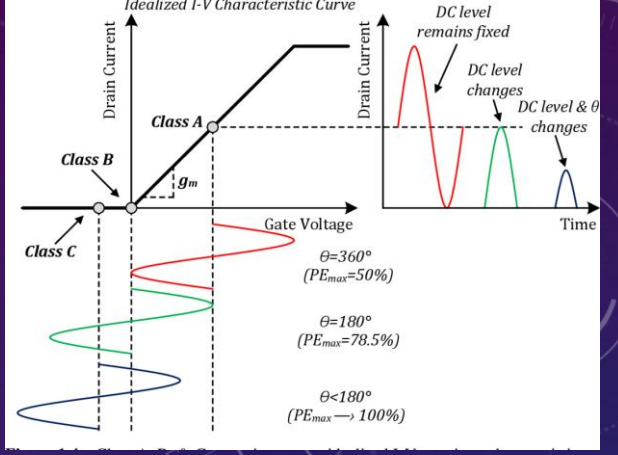
MOS FET Modification Transistor Curve Tracer HeathKit
https://www.youtube.com/watch?v=rFmDUPVdYSc&t=4953s

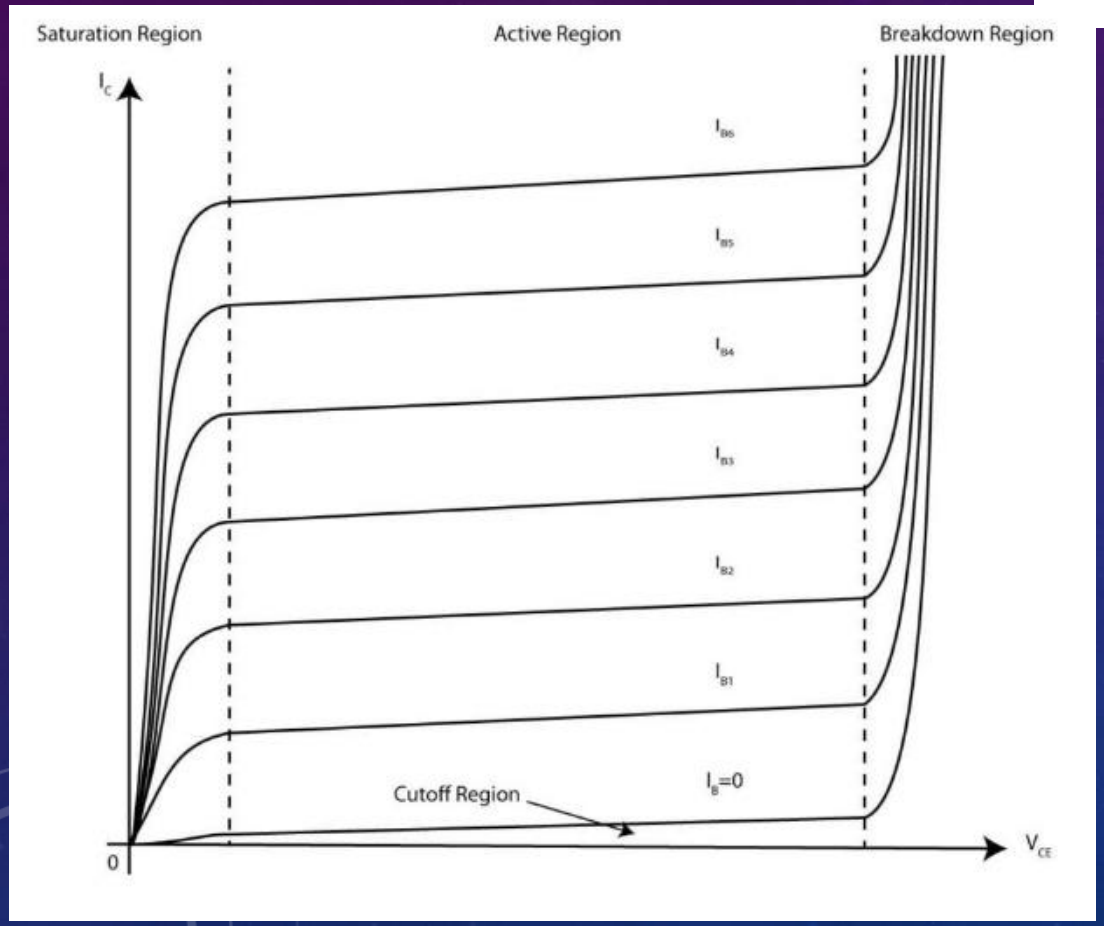
TRANSISTOR REGIONS: VOLTAGE & CURRENT

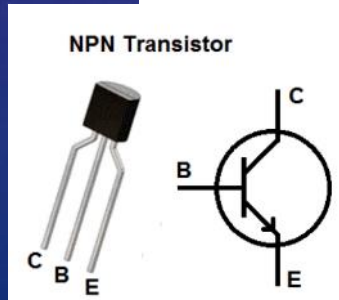
The **Breakdown Region** of a transistor is the region where the collector voltage, Vcc, is so large that the *collector-base diode breaks down, causing a large, undesired collector current to flow*.

When the collector-base voltage is too large, the collector-base diode breaks down, so that the collector conducts electricity. So even though the base of the transistor doesn't receive any current, the transistor still conducts across the collector. This is called the breakdown region.

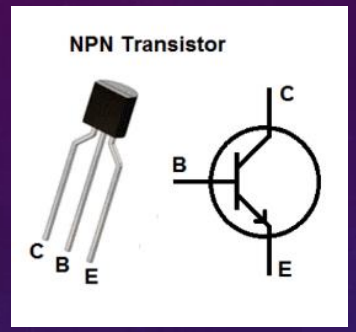
The breakdown region should always be avoided in transistor circuits. This is achieved by not placing too much bias voltage on the collector.







TRANSISTOR REGIONS: CURRENT

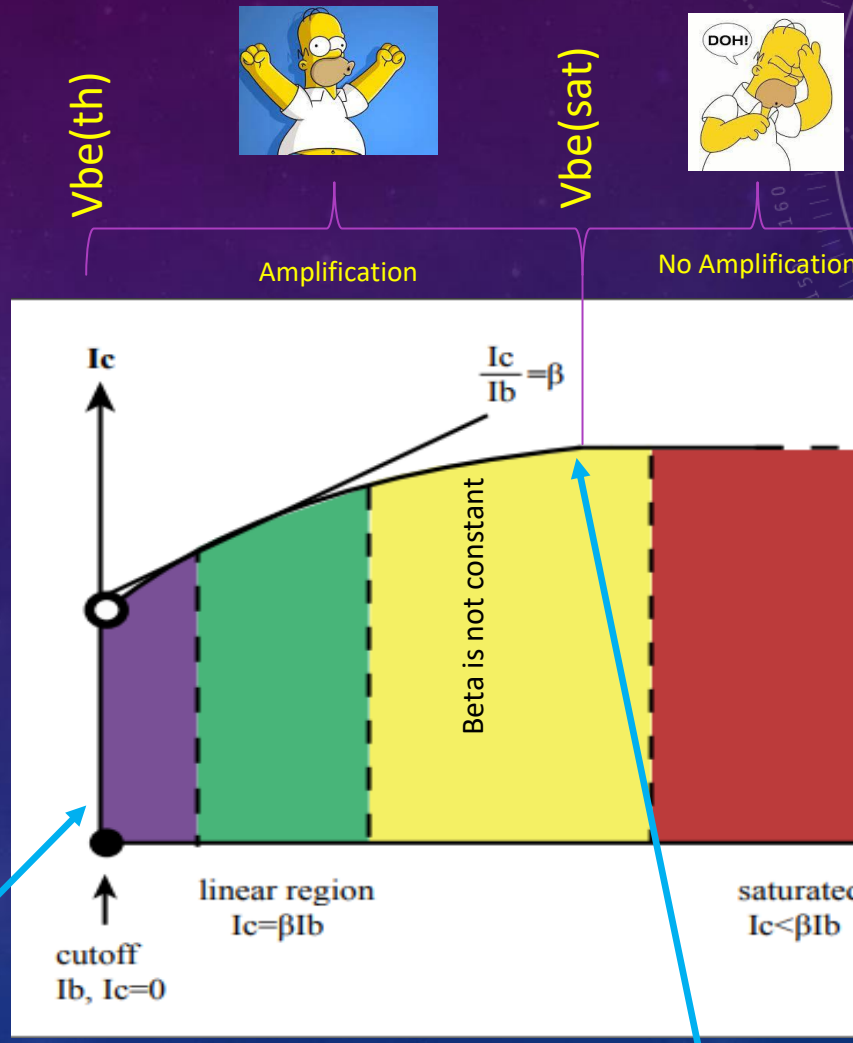


Feed Small Current into Base

TRANSISTOR

β

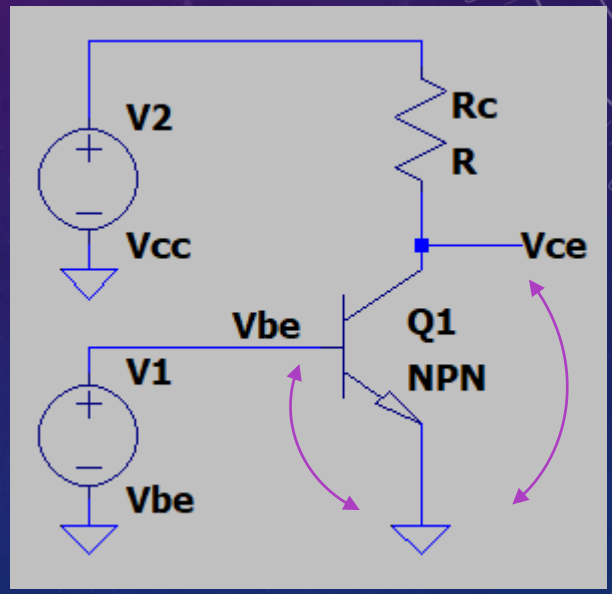
Get a LARGE current between Collector and Emitter



There is "Vbe(th)" for BJT, which is the threshold voltage (actually current) at which device starts conducting. ***Not specified on data sheet!!***

There is Vbe(sat) for BJT, which is the maximum voltage at which collector current is constant.

TRANSISTOR REGIONS: VOLTAGE



NPN PLOTS

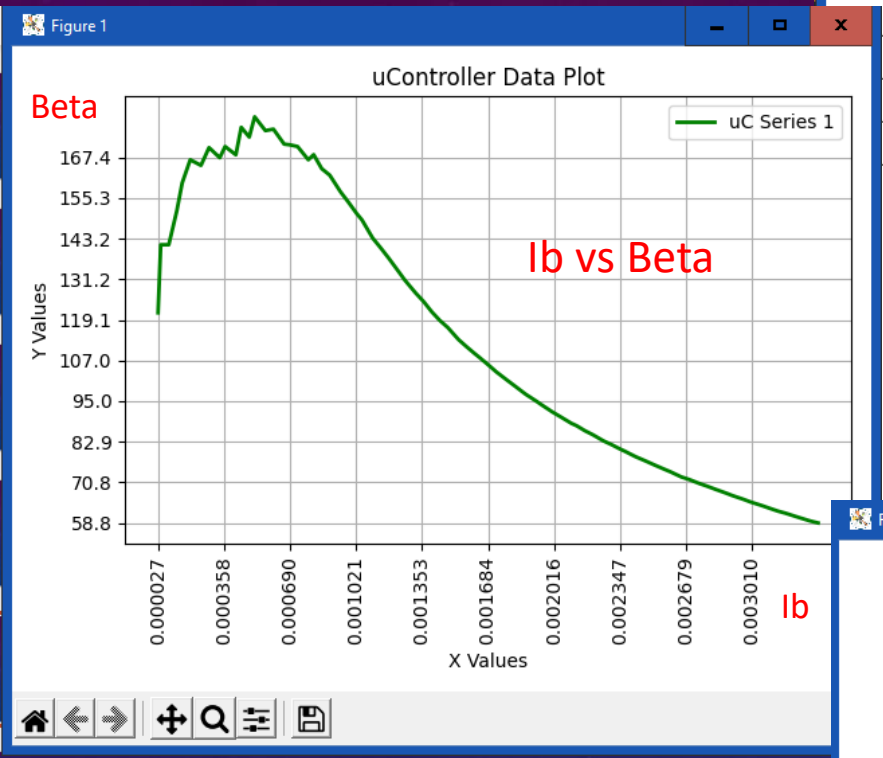


Figure 1
uController Data Plot
Beta
uC Series 1
Ib vs Beta
Y Values
X Values
Ib

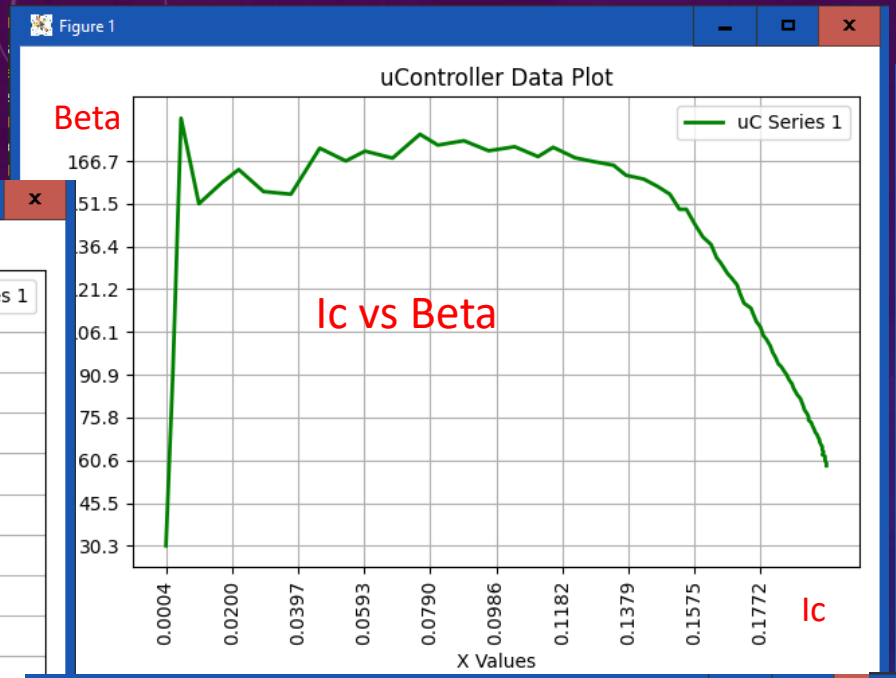


Figure 1
uController Data Plot
Beta
uC Series 1
Ic vs Beta
X Values
Ic

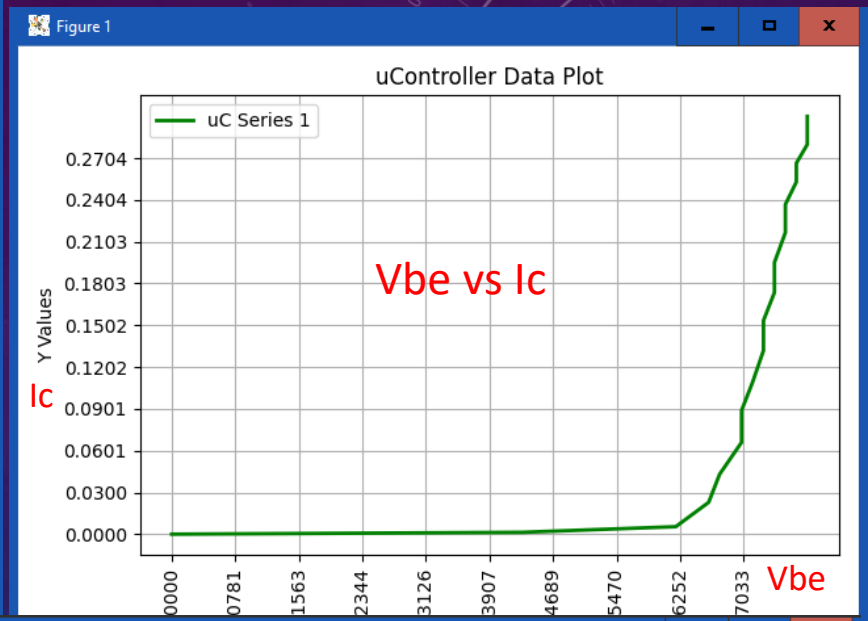


Figure 1
uController Data Plot
uC Series 1
Vbe vs Ic
Y Values
Ic
Vbe

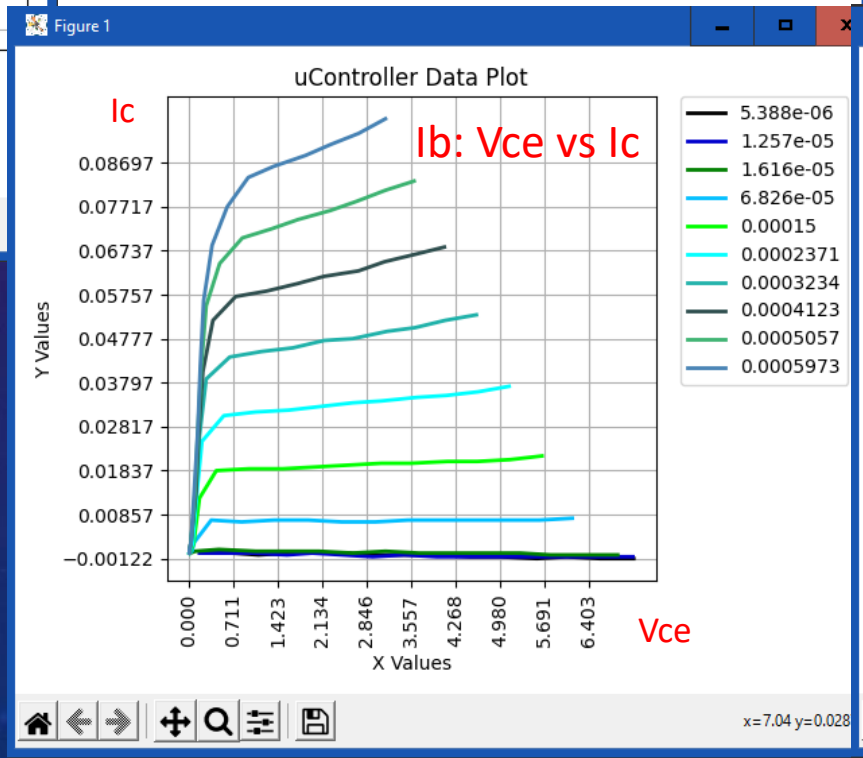


Figure 1
uController Data Plot
Ic
Ib: Vce vs Ic
Y Values
X Values
Vce
5.388e-06
1.257e-05
1.616e-05
6.826e-05
0.00015
0.0002371
0.0003234
0.0004123
0.0005057
0.0005973

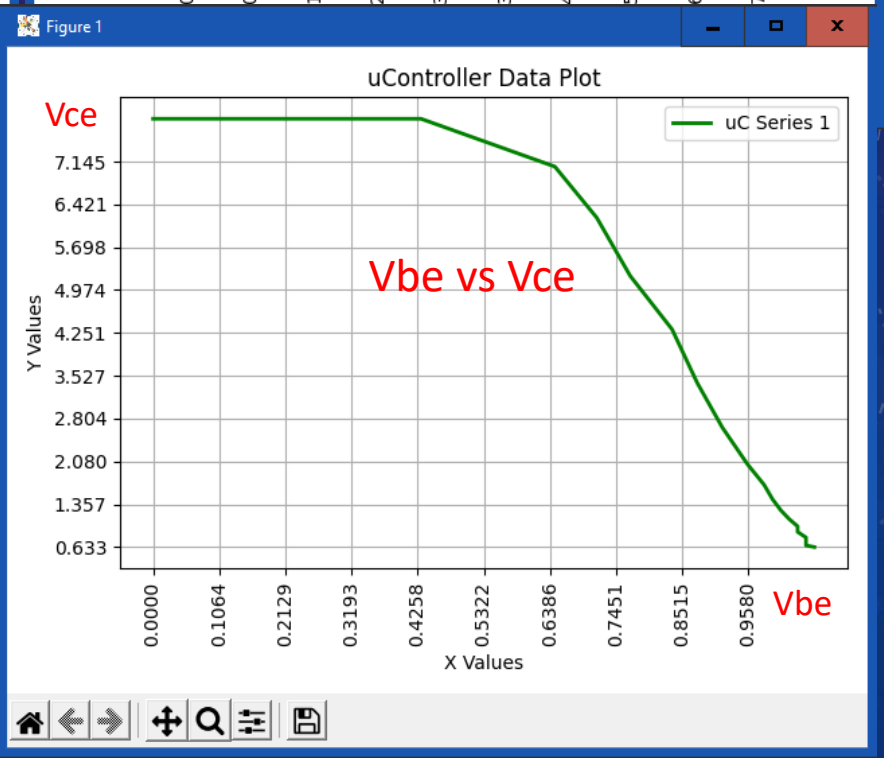
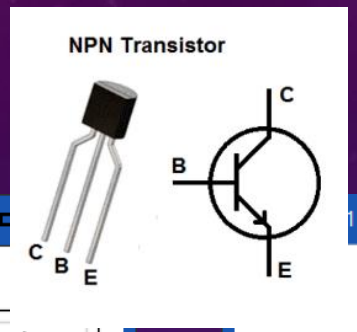


Figure 1
uController Data Plot
Vce
uC Series 1
Vbe vs Vce
Y Values
X Values
Vbe

NPN PLOTS


NPN Transistor
C
B
E
C
B
E

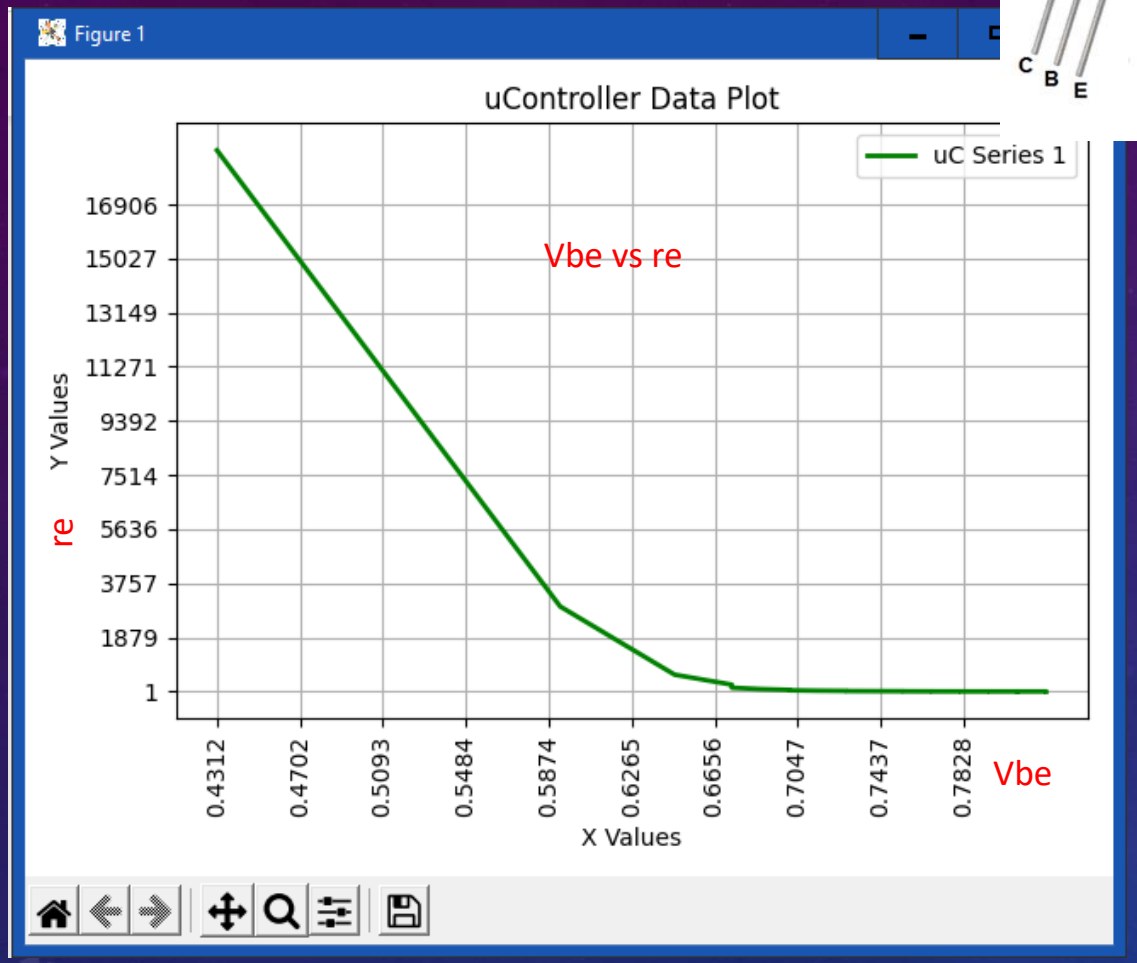
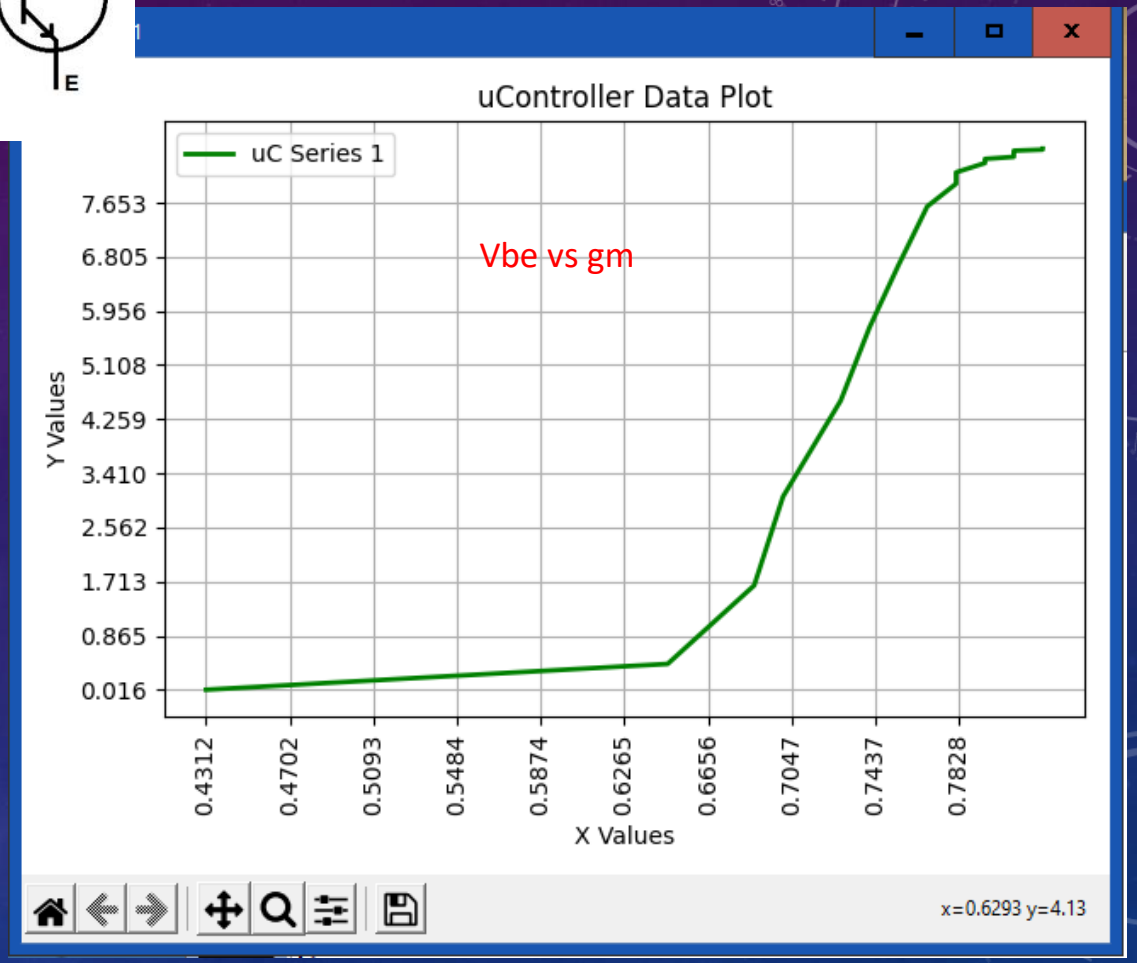
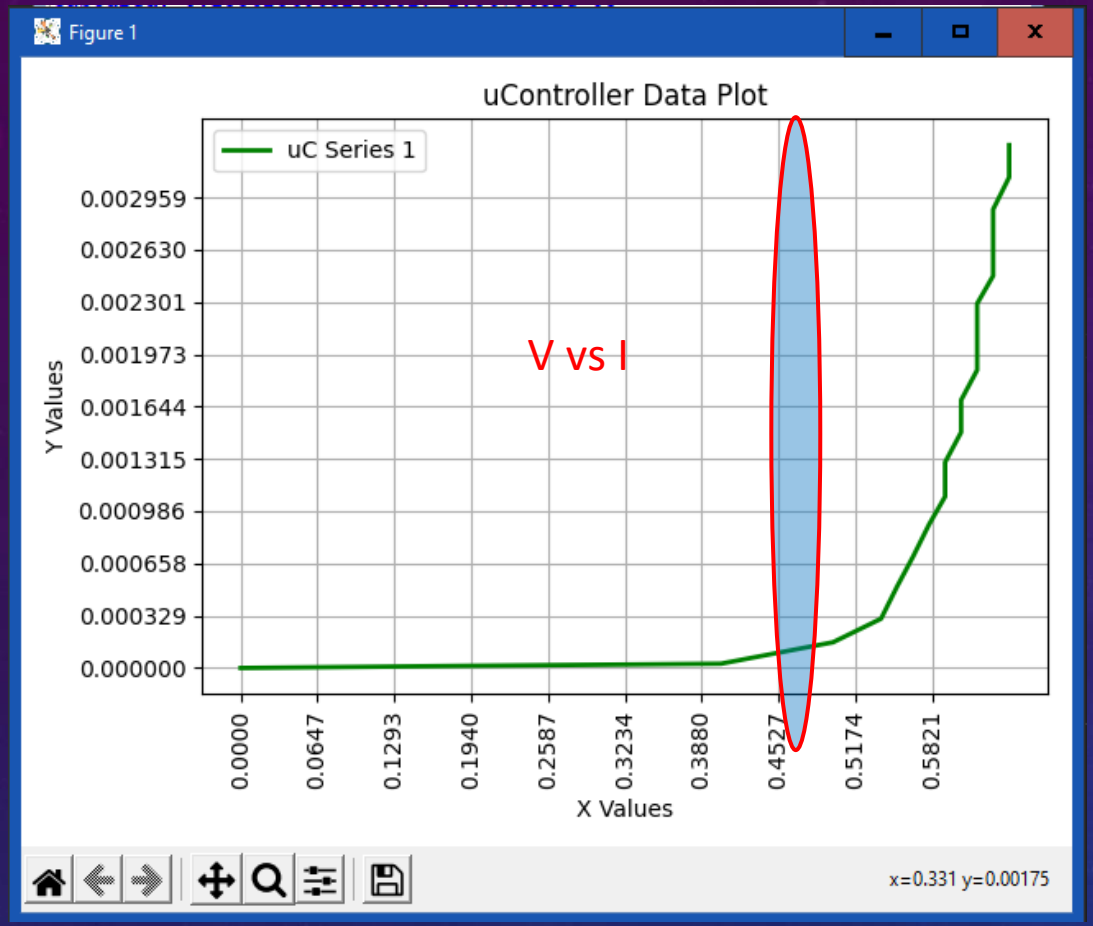


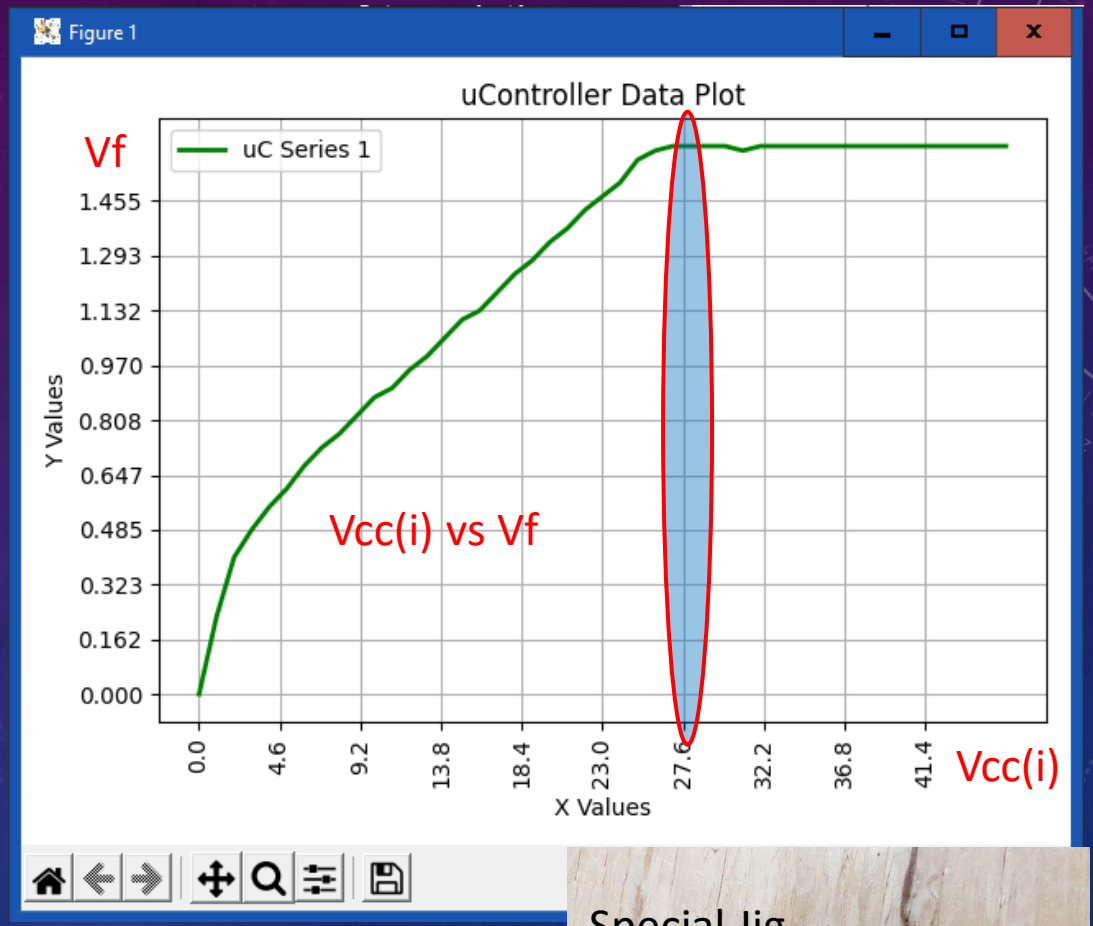
Figure 1
uController Data Plot
uC Series 1
Vbe vs re
16906
15027
13149
11271
9392
7514
5636
3757
1879
1
Y Values
re
0.4312
0.4702
0.5093
0.5484
0.5874
0.6265
0.6656
0.7047
0.7437
0.7828
Vbe
X Values


uController Data Plot
uC Series 1
Vbe vs gm
7.653
6.805
5.956
5.108
4.259
3.410
2.562
1.713
0.865
0.016
Y Values
0.4312
0.4702
0.5093
0.5484
0.5874
0.6265
0.6656
0.7047
0.7437
0.7828
X Values
x=0.6293 y=4.13

MEASUREMENTS OF DIODES AND BJT DEVICES

DIODE PLOTS: 1N914

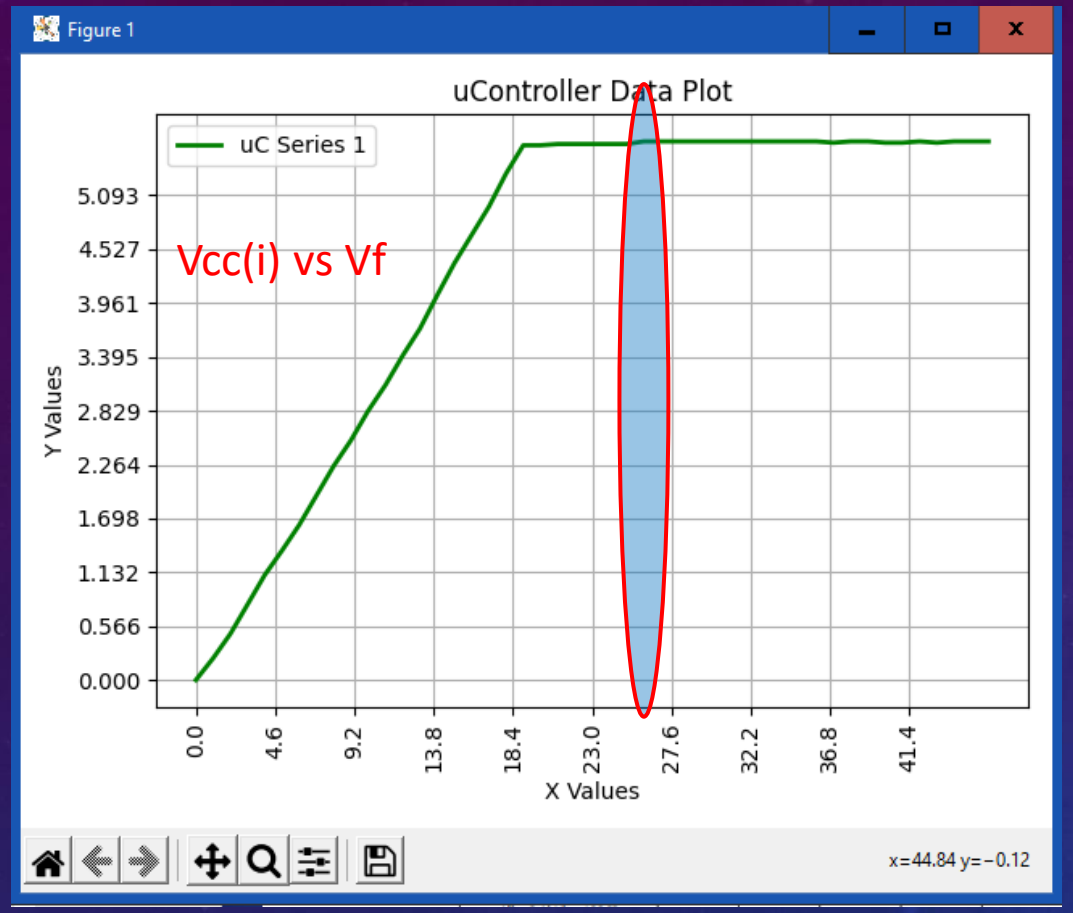





```
Vf: 1.6168
```

PARAMETER	TEST CONDITION	SYMBOL	MIN.	TYP.	MAX.	UNIT
Forward voltage	I_F = 10 mA	V_F			1	V
Breakdown voltage	I_R = 100 µA	$V_{(BR)}$	100			V
Peak reverse current	V_R = 75 V	I_R			5	µA
	V_R = 20 V, T_j = 150 °C	I_R			50	µA
	V_R = 20 V	I_R			25	nA
Diode capacitance	V_R = 0, f = 1 MHz	C_D			4	pF
Reverse recovery time	I_F = 10 mA, I_R = 1 mA, V_R = 6 V, R_L = 100 Ω	t_{rr}			4	ns

DIODE PLOTS: 1N752 (5.6Z)



Vf: 5.6577

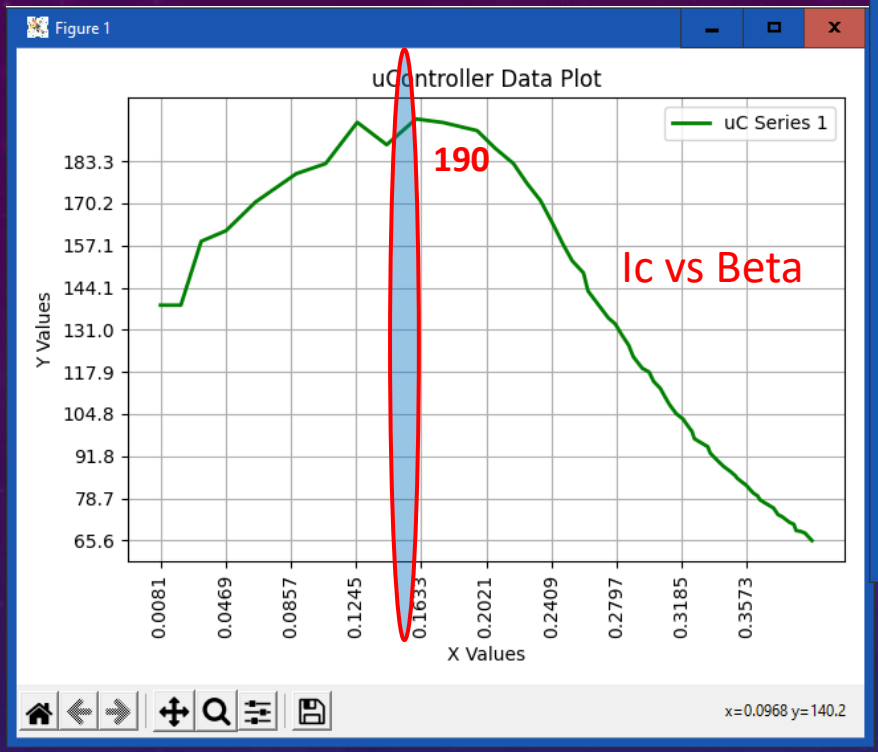
ELECTRICAL CHARACTERISTICS* @ 25°C

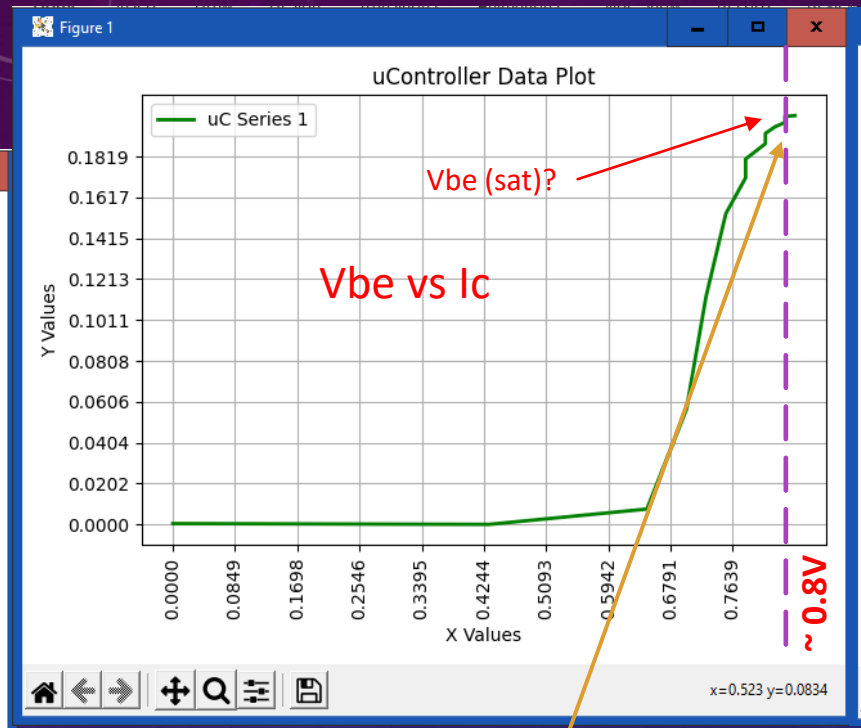
JEDEC TYPE NO. (NOTE1)	NOMINAL ZENER VOLTAGE V_Z @ I_{ZT} (NOTE 2)	ZENER TEST CURRENT I_{ZT}	MAXIMUM ZENER IMPEDANCE Z_{ZT} @ I_{ZT} (NOTE 3)	MAXIMUM REVERSE CURRENT I_R @ V_R = 1 VOLT		MAXIMUM ZENER CURRENT I_{ZM} (NOTE 4)	TYPICAL TEMP COEFF. OF ZENER VOLTAGE α_{VZ}
				@25°C	@+150°C		
	VOLTS	mA	OHMS	µA	µA	mA	%/°C
1N4370	2.4	20	30	100	200	150	-.085
1N4371	2.7	20	30	75	150	135	-.080
1N4372	3.0	20	29	50	100	120	-.075
1N746	3.3	20	28	10	30	110	-.066
1N747	3.6	20	24	10	30	100	-.058
1N748	3.9	20	23	10	30	95	-.046
1N749	4.3	20	22	2	30	85	-.033
1N750	4.7	20	19	2	30	75	-.015
1N751	5.1	20	17	1	20	70	+/-.010
1N752	5.6	20	11	1	20	65	+.030

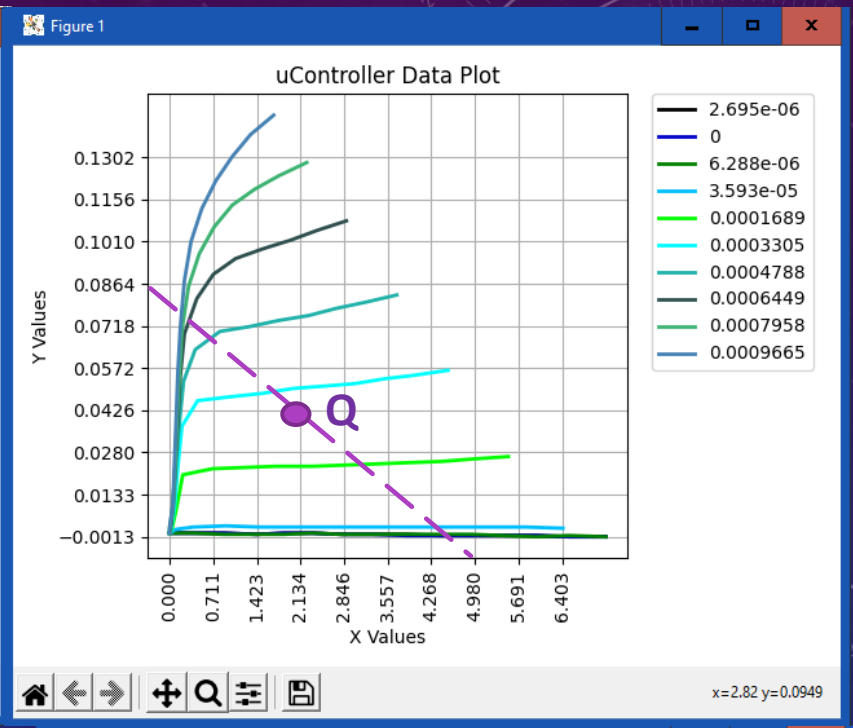

AC/DC ADAPTER
MODEL:A-88
INPUT:AC100-240V 50-60Hz
OUTPUT:DC10V ⎓500mA
CHARGING (RED)
COMPLETED (GREEN)
MADE IN CHINA


DOH!

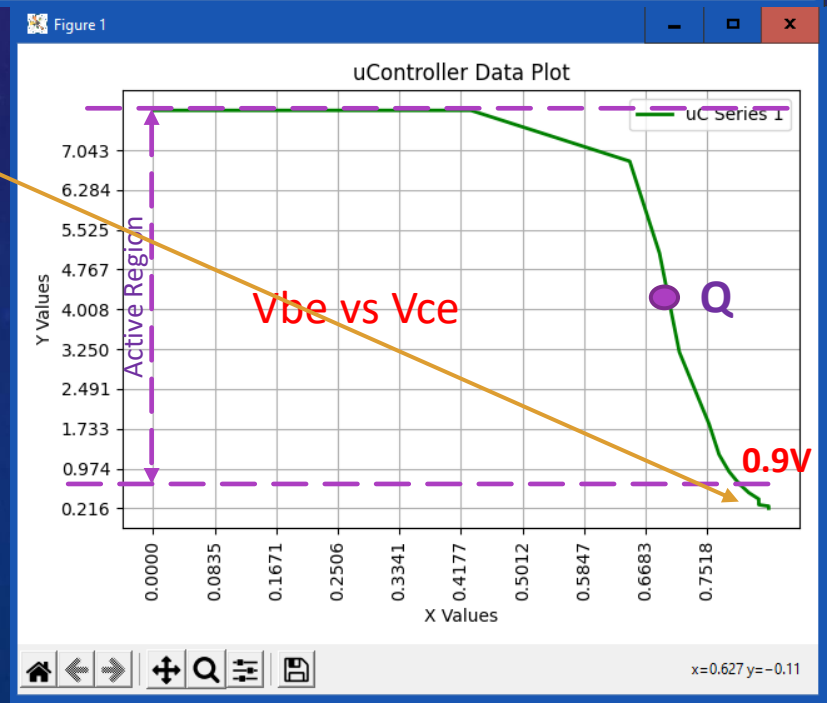
2N2222A





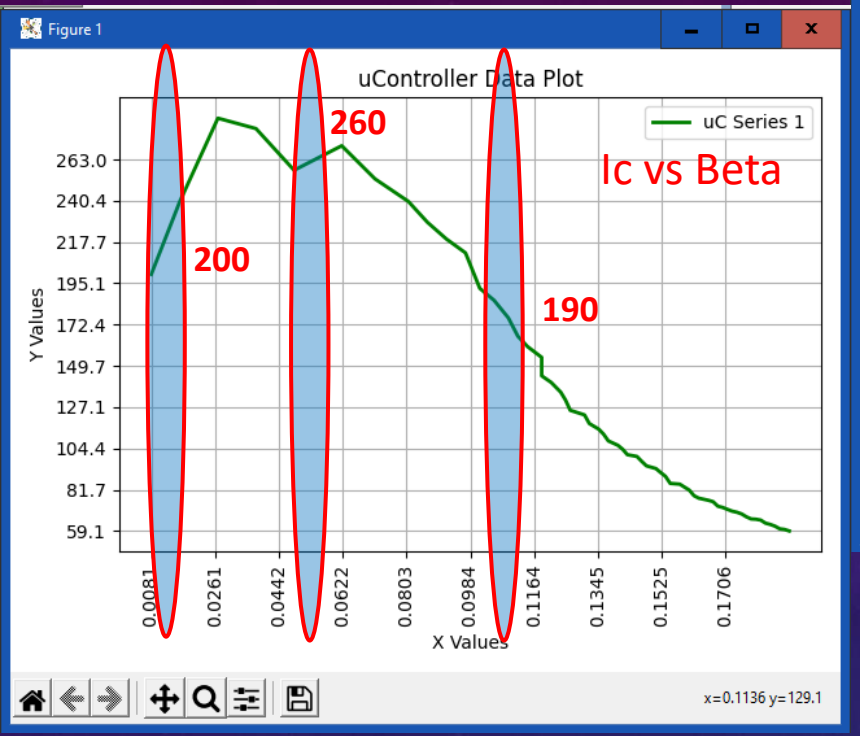


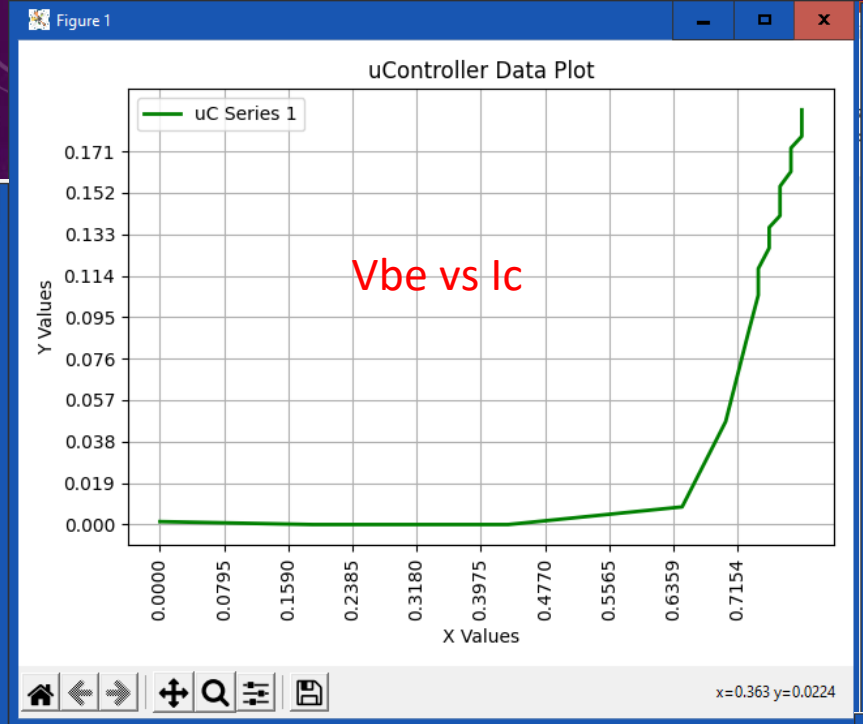
Actual roll over or
Tracer Limitation

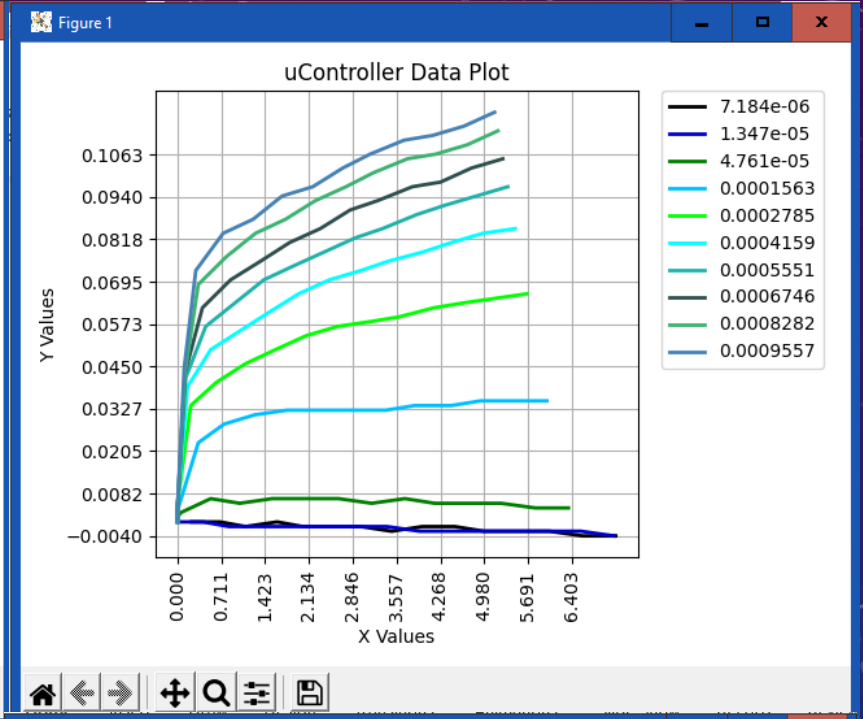


$V_{CE(sat)}$*	Collector-Emitter Saturation Voltage	I_C = 150 mA I_C = 500 mA	I_B = 15 mA I_B = 50 mA			0.3 1	V V
$V_{BE(sat)}$*	Base-Emitter Saturation Voltage	I_C = 150 mA I_C = 500 mA	I_B = 15 mA I_B = 50 mA	0.6		1.2 2	V V
h_{FE}*	DC Current Gain	I_C = 0.1 mA I_C = 1 mA I_C = 10 mA I_C = 150 mA I_C = 500 mA I_C = 150 mA I_C = 10 mA T_{amb} = -55 °C	V_{CE} = 10 V V_{CE} = 10 V V_{CE} = 10 V V_{CE} = 10 V V_{CE} = 10 V V_{CE} = 1 V V_{CE} = 10 V	35 50 75 100 40 50 35		 300 	

2N3904

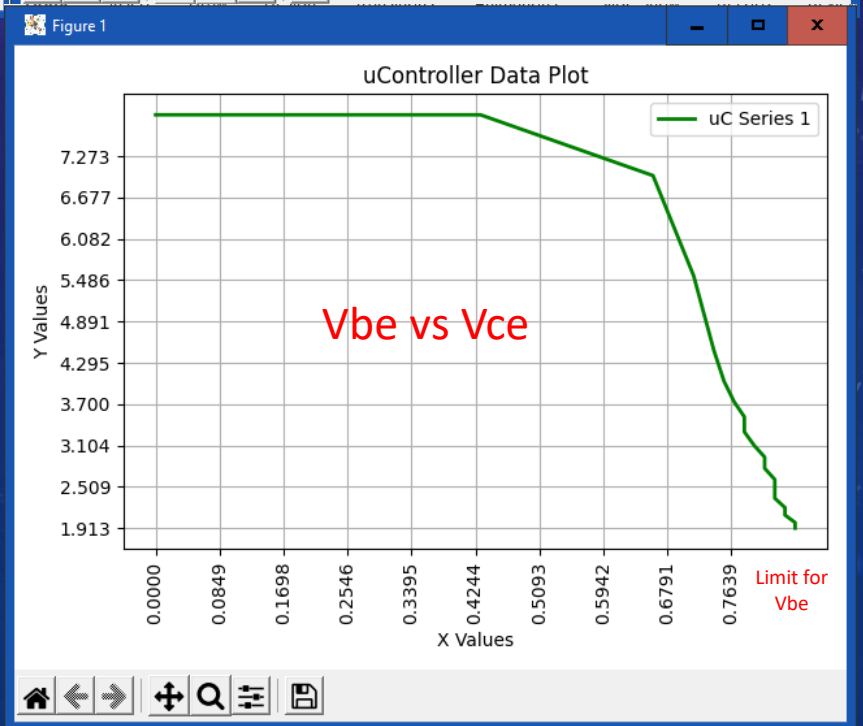




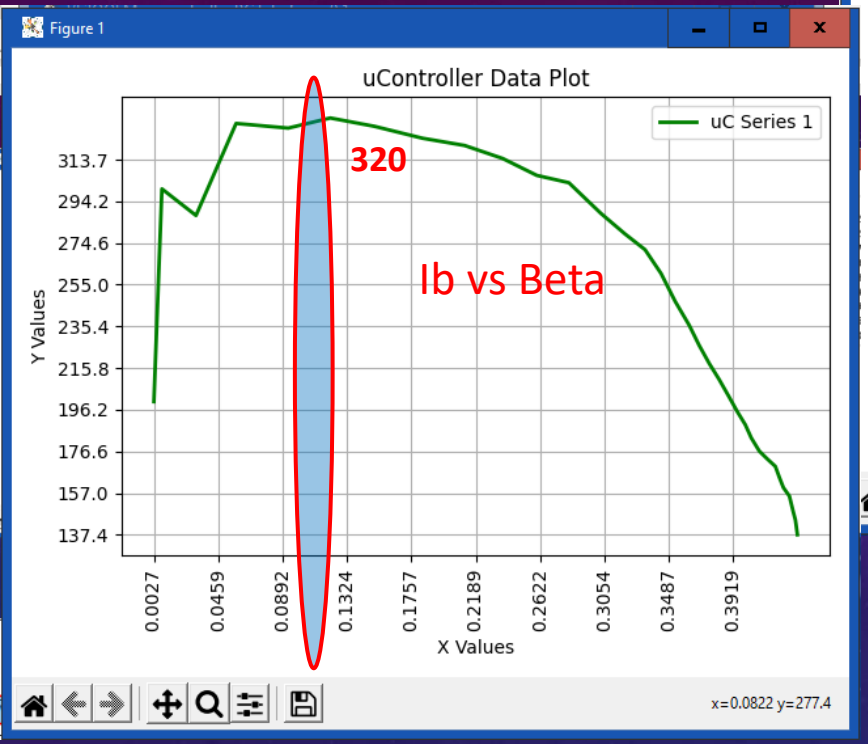


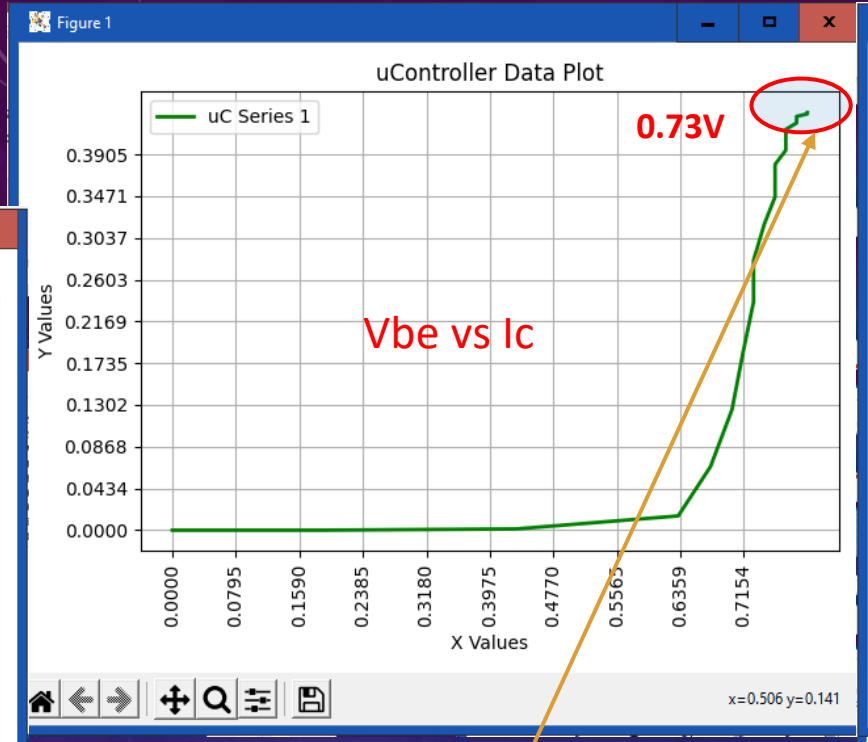
$V_{CE(sat)}$*	Collector-Emitter Saturation Voltage	I_C = 10 mA I_C = 50 mA	I_B = 1 mA I_B = 5 mA			0.2 0.2	V V
$V_{BE(sat)}$*	Base-Emitter Saturation Voltage	I_C = 10 mA I_C = 50 mA	I_B = 1 mA I_B = 5 mA	 0.65		0.85 0.95	V V
h_{FE}*	DC Current Gain	I_C = 0.1 mA I_C = 1 mA I_C = 10 mA I_C = 50 mA I_C = 100 mA	V_{CE} = 1 V V_{CE} = 1 V V_{CE} = 1 V V_{CE} = 1 V V_{CE} = 1 V	60 80 100 60 30		300	

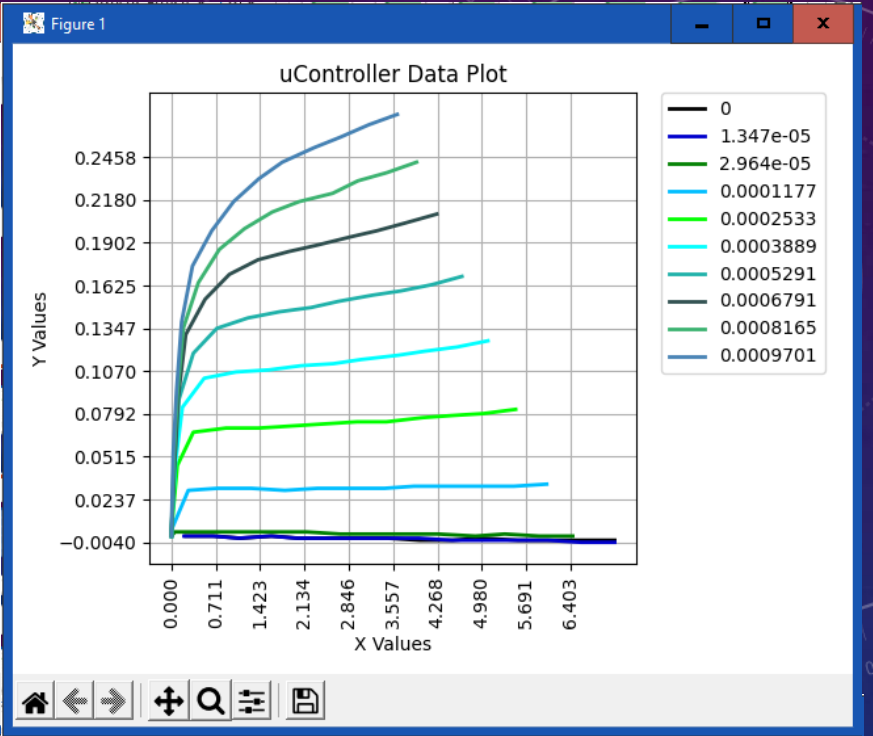
MINIMUM



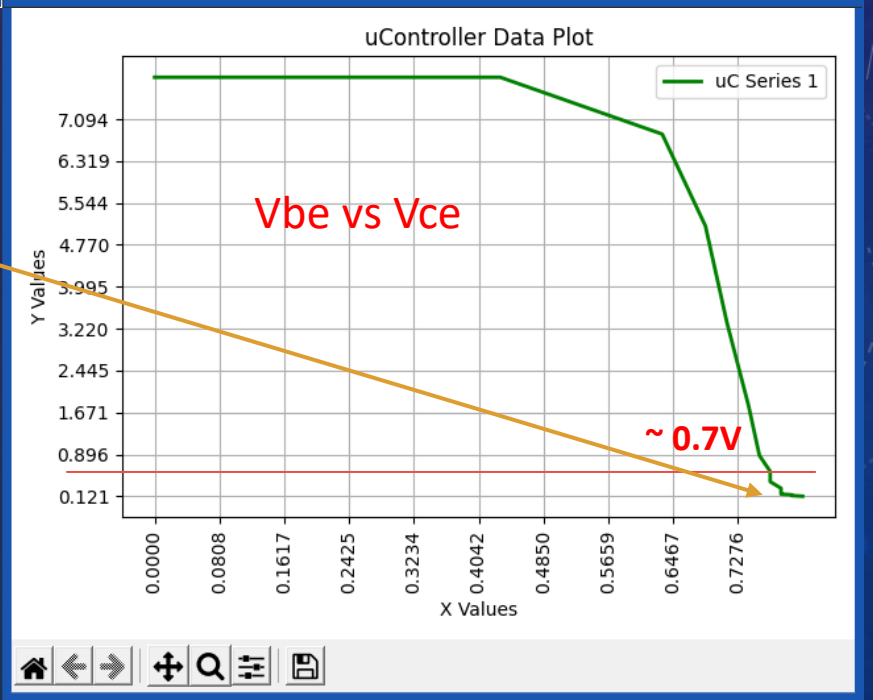
BC337





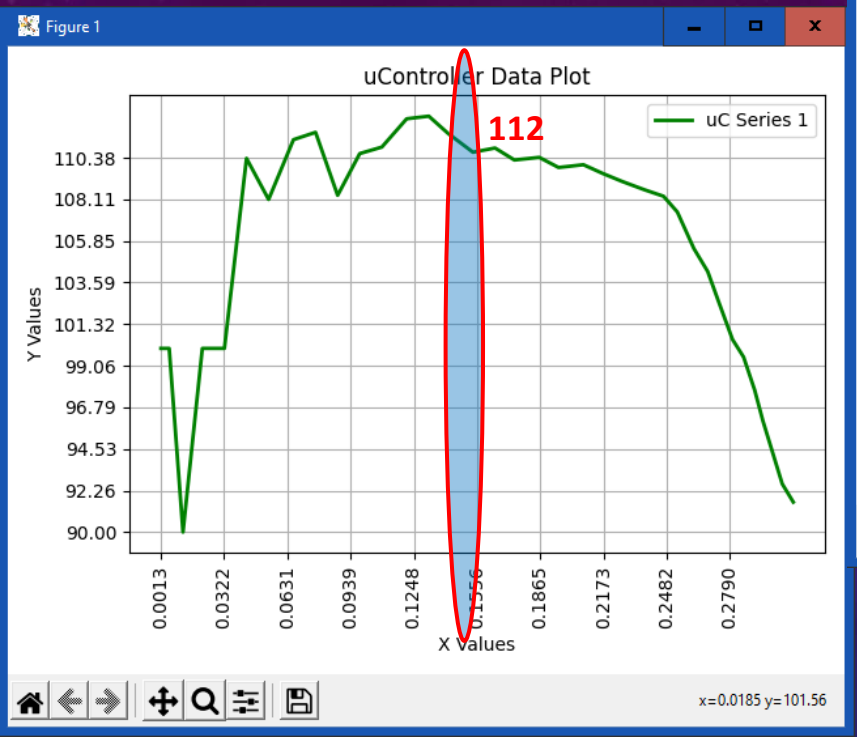


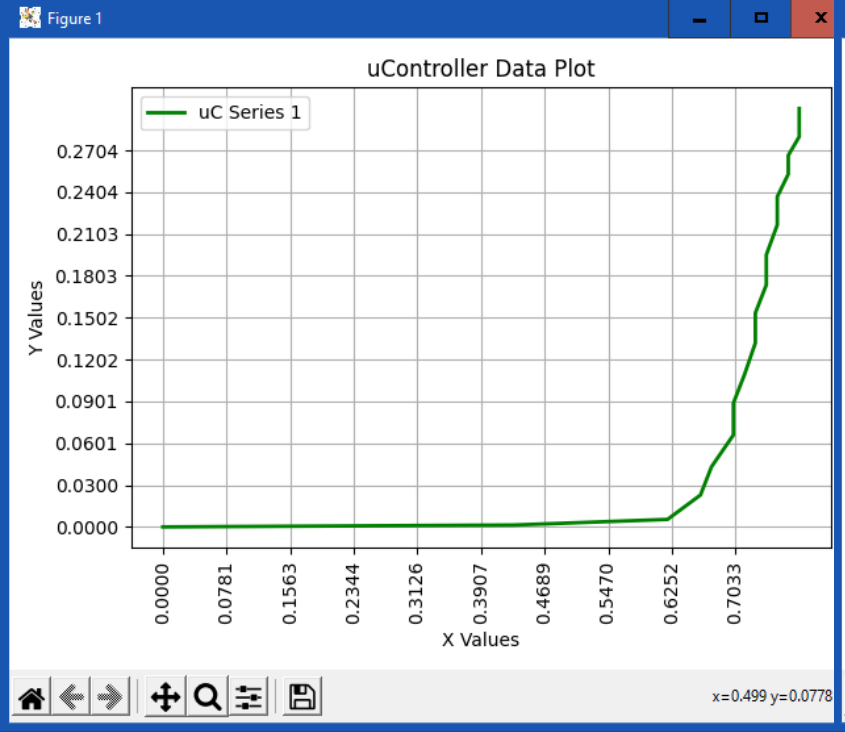
Actual roll over or
Tracer Limitation

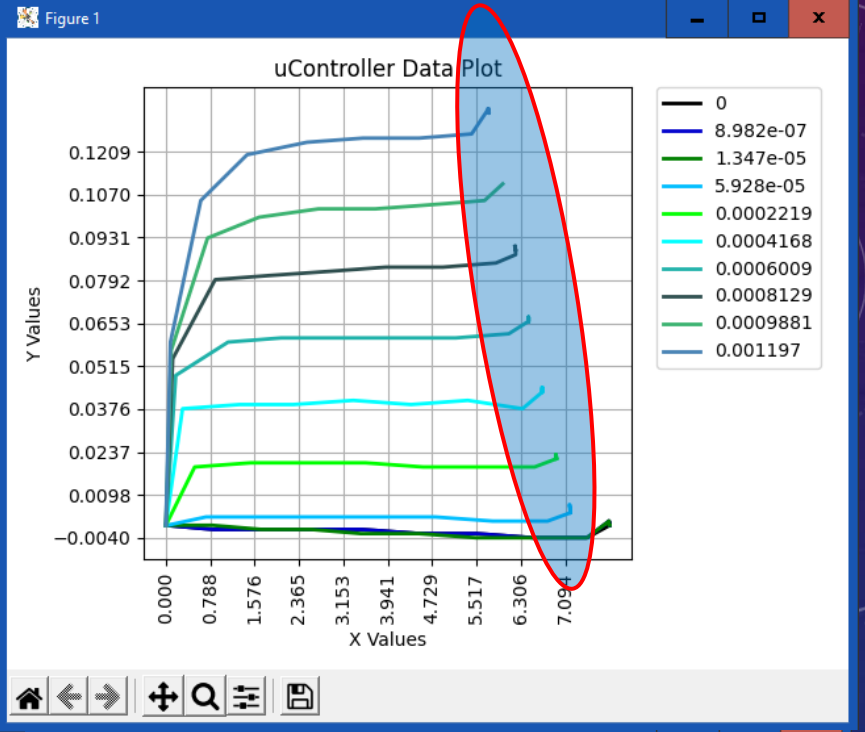


DC Current Gain (I_C = 100 mA, V_{CE} = 1.0 V)	BC337	h_{FE}	100	–	630	–
	BC337–25		160	–	400	
	BC337–40		250	–	630	
(I_C = 300 mA, V_{CE} = 1.0 V)			60	–	–	
Base–Emitter On Voltage (I_C = 300 mA, V_{CE} = 1.0 V)		$V_{BE(on)}$	–	–	1.2	Vdc
Collector–Emitter Saturation Voltage (I_C = 500 mA, I_B = 50 mA)		$V_{CE(sat)}$	–	–	0.7	Vdc

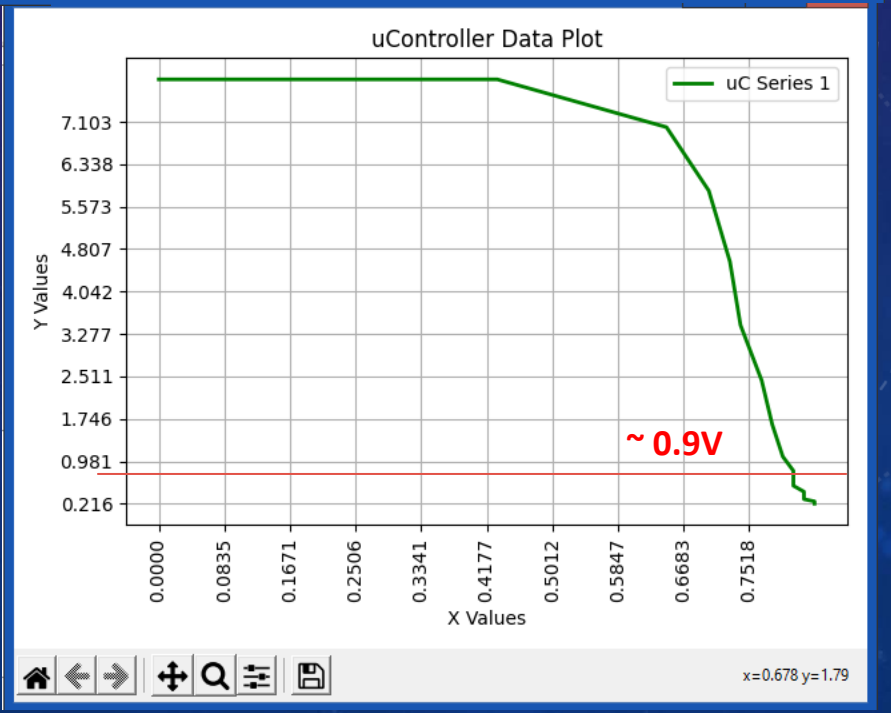
2N2219A







$V_{CE(sat)}$*	Collector-Emitter Saturation Voltage	I_C = 150 mA I_B = 15 mA I_C = 500 mA I_B = 50 mA			0.3 1	V V
$V_{BE(sat)}$*	Base-Emitter Saturation Voltage	I_C = 150 mA I_B = 15 mA I_C = 500 mA I_B = 50 mA	0.6		1.2 2	V V
h_{FE}*	DC Current Gain	I_C = 0.1 mA V_{CE} = 10 V I_C = 1 mA V_{CE} = 10 V I_C = 10 mA V_{CE} = 10 V I_C = 150 mA V_{CE} = 10 V I_C = 500 mA V_{CE} = 10 V I_C = 150 mA V_{CE} = 1 V I_C = 10 mA V_{CE} = 10 V T_{amb} = -55 °C	35 50 75 100 40 50 35		300	



BD139

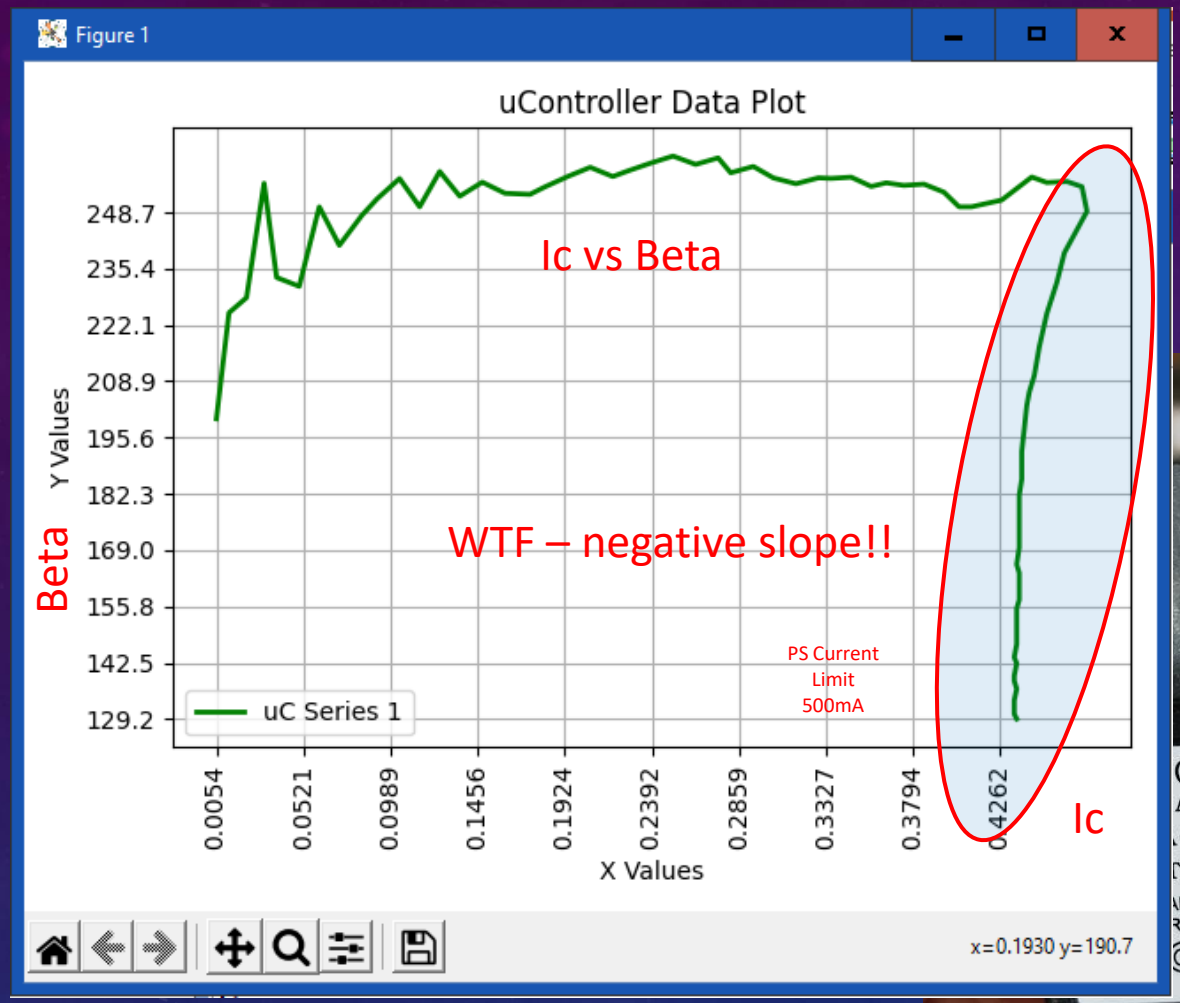


Figure 1
uController Data Plot
Ic vs Beta
WTF – negative slope!!
PS Current Limit 500mA
Beta
Y Values
248.7
235.4
222.1
208.9
195.6
182.3
169.0
155.8
142.5
129.2
uC Series 1
0.0054
0.0521
0.0989
0.1456
0.1924
0.2392
0.2859
0.3327
0.3794
0.4262
X Values
Ic
x=0.1930 y=190.7


C ADAPTER
A-88
C100-240V 50-60Hz
:DC10V ⎓500mA
ARGING
RED)
COMPLETED
(GREEN)
MADE IN CHINA

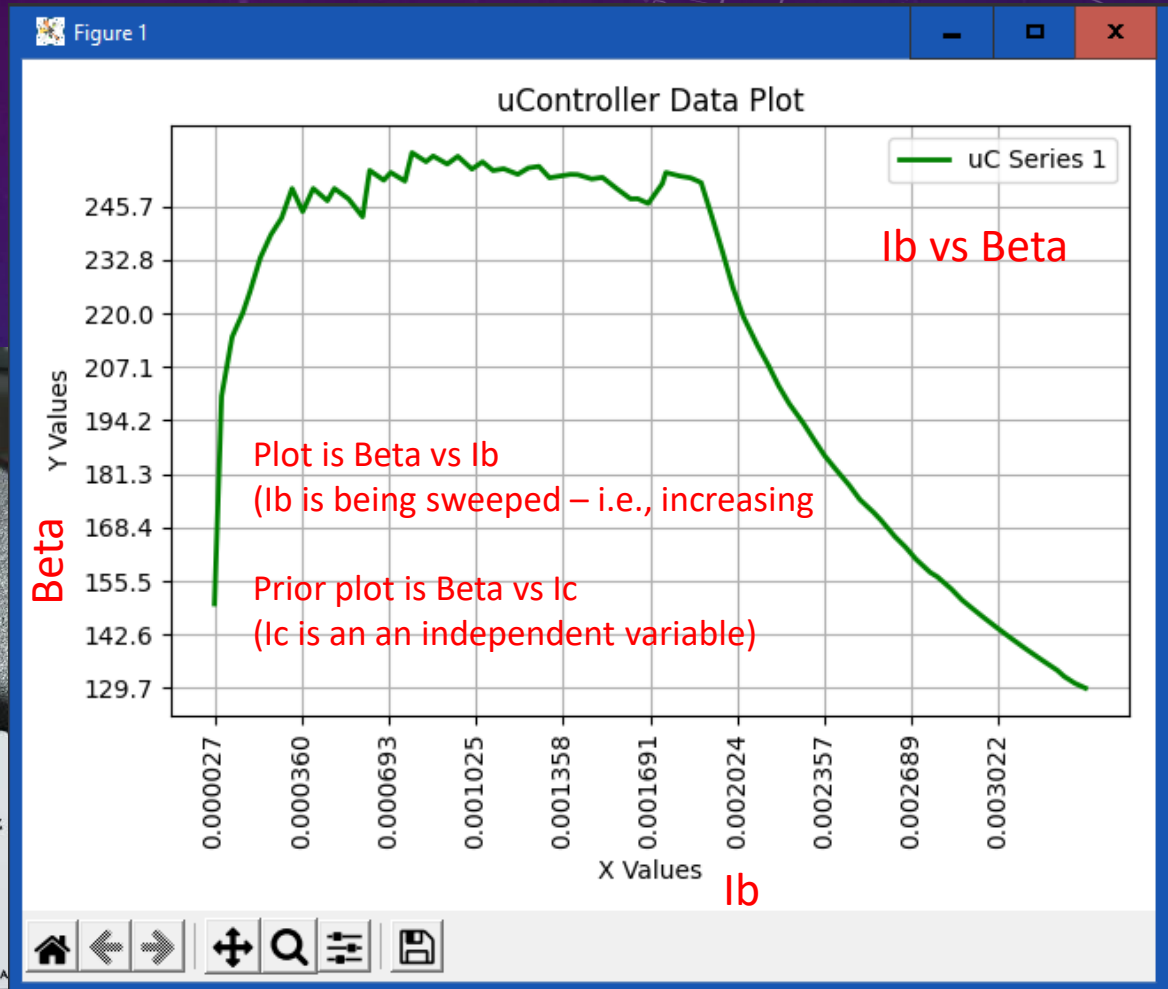
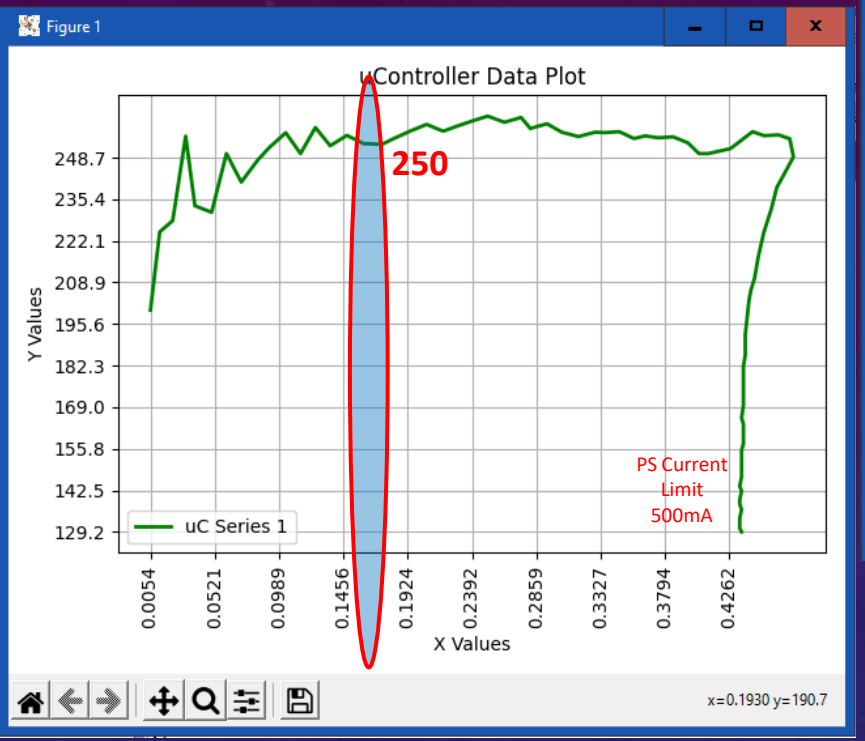
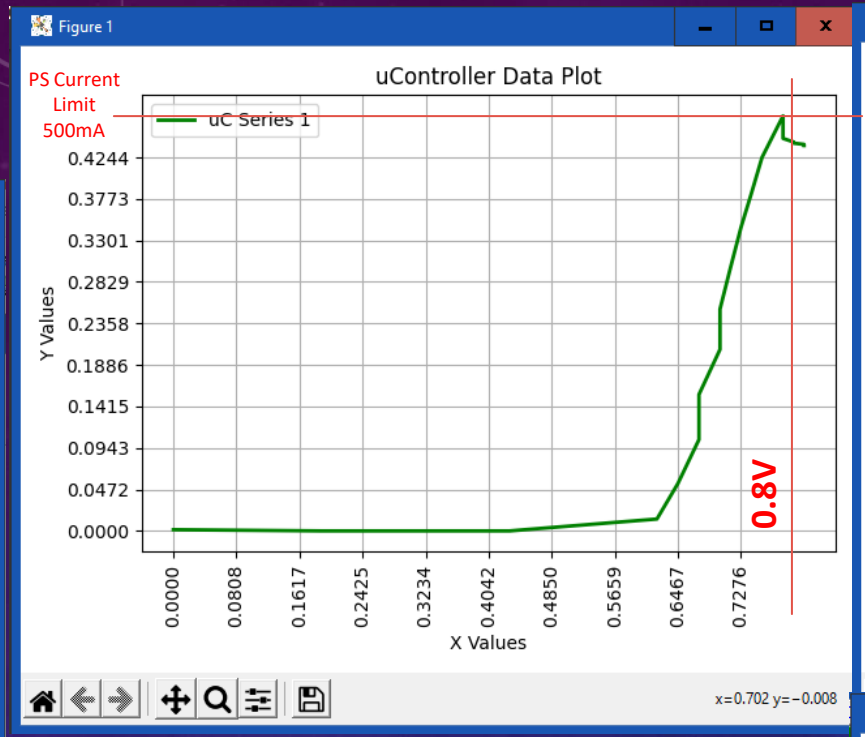
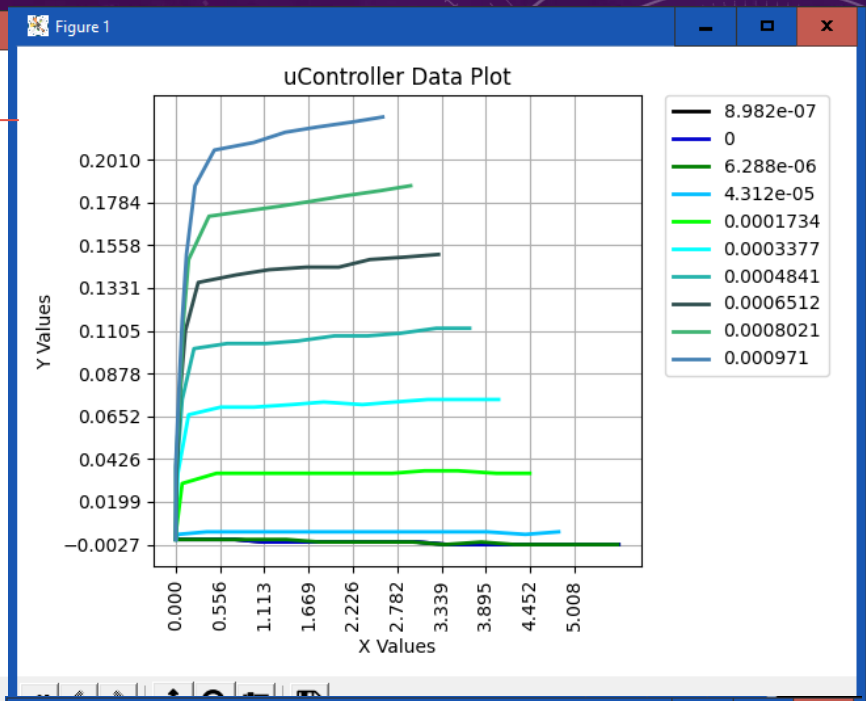


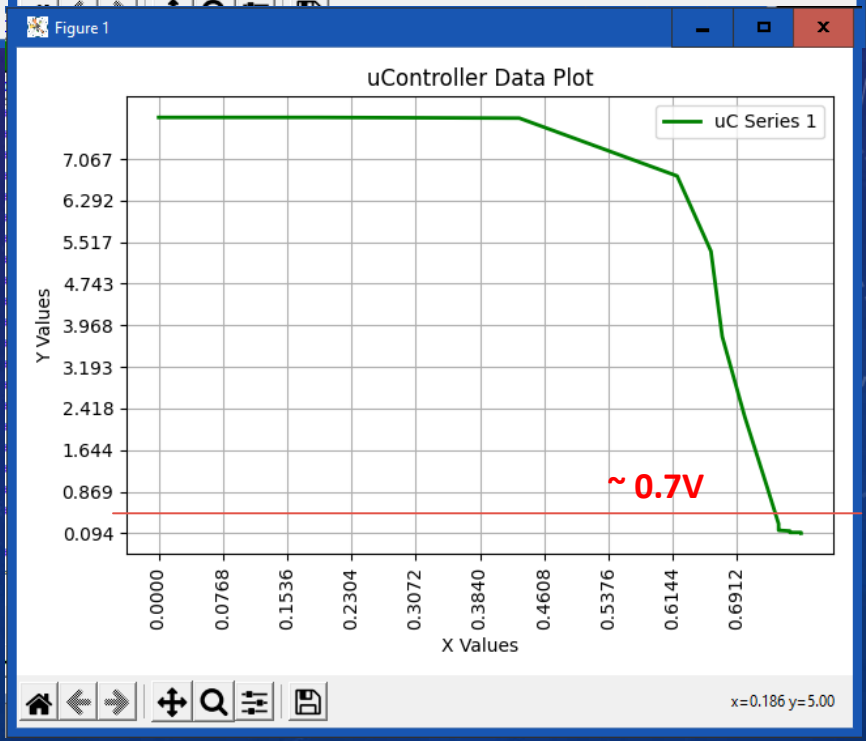
Figure 1
uController Data Plot
uC Series 1
Ib vs Beta
Plot is Beta vs Ib
(Ib is being sweeped – i.e., increasing
Prior plot is Beta vs Ic
(Ic is an an independent variable)
Beta
Y Values
245.7
232.8
220.0
207.1
194.2
181.3
168.4
155.5
142.6
129.7
0.000027
0.000360
0.000693
0.001025
0.001358
0.001691
0.002024
0.002357
0.002689
0.003022
X Values
Ib

BD139



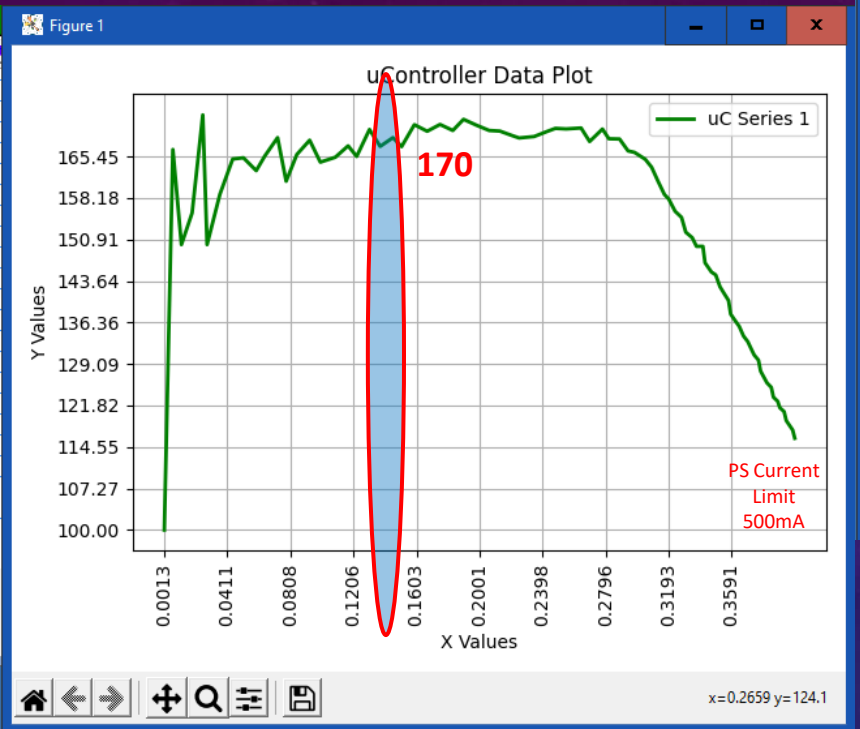


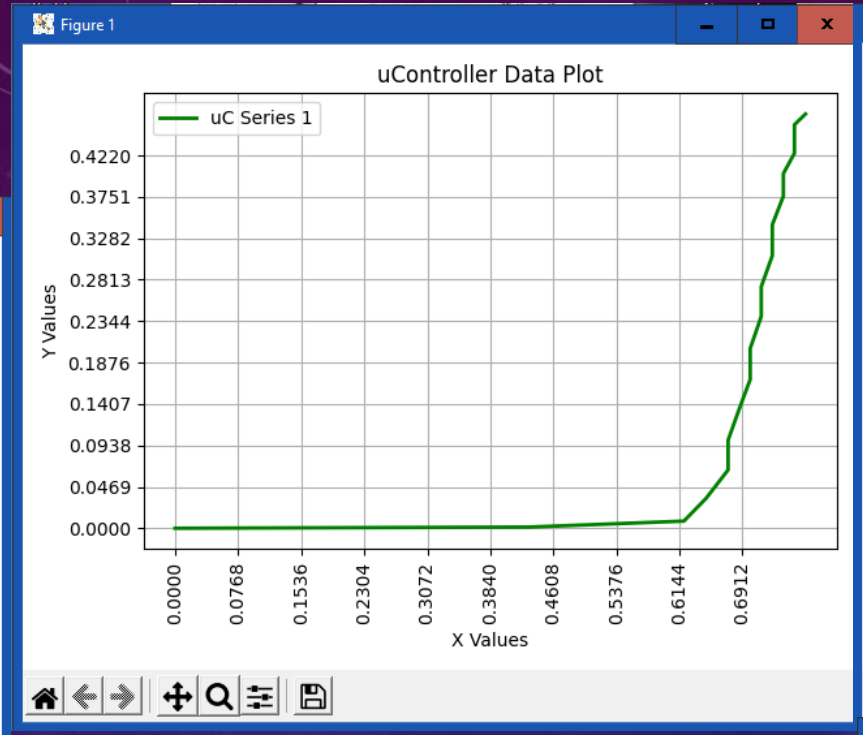


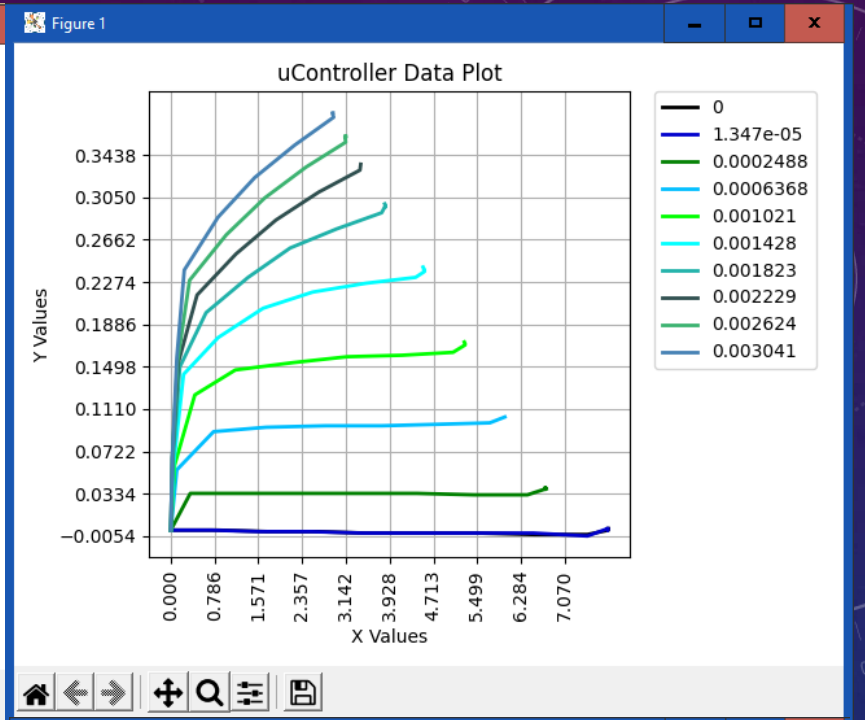


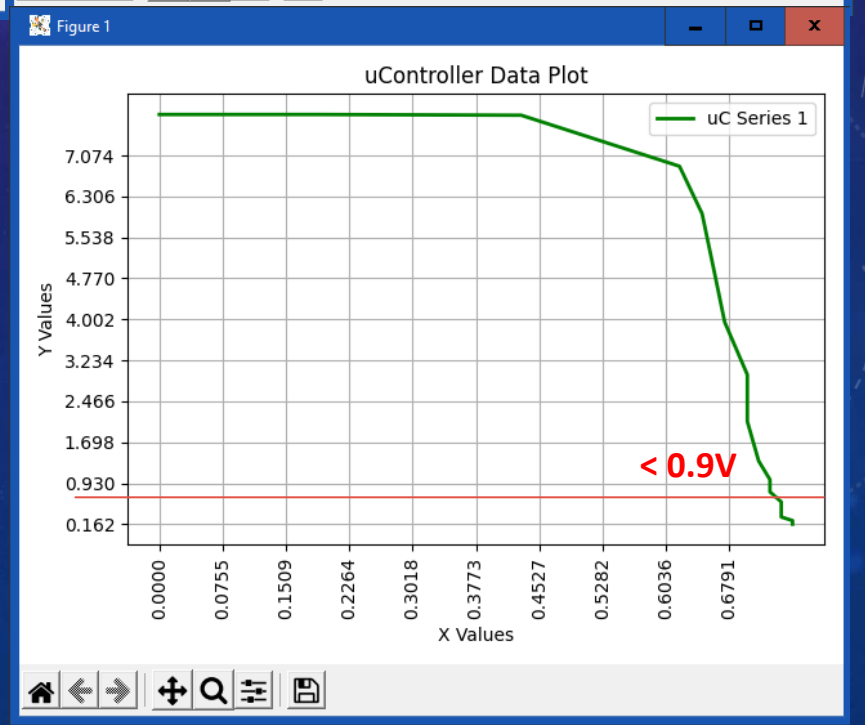
h_{FE1}	DC Current Gain	V_{CE} = 2 V, I_C = 5 mA	25			
h_{FE2}		V_{CE} = 2 V, I_C = 0.5 A	25			
h_{FE3}		V_{CE} = 2 V, I_C = 150 mA	40		250	
V_{CE}(sat)	Collector-Emitter Saturation Voltage	I_C = 500 mA, I_B = 50 mA			0.5	V
V_{BE}(on)	Base-Emitter On Voltage	V_{CE} = 2 V, I_C = 0.5 A			1	V

2N3050



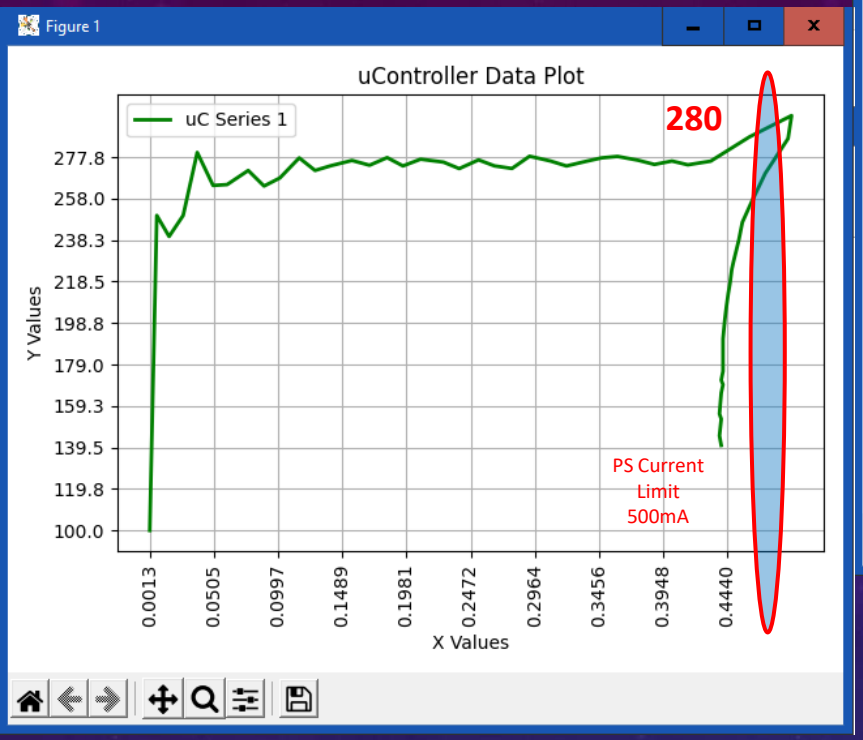


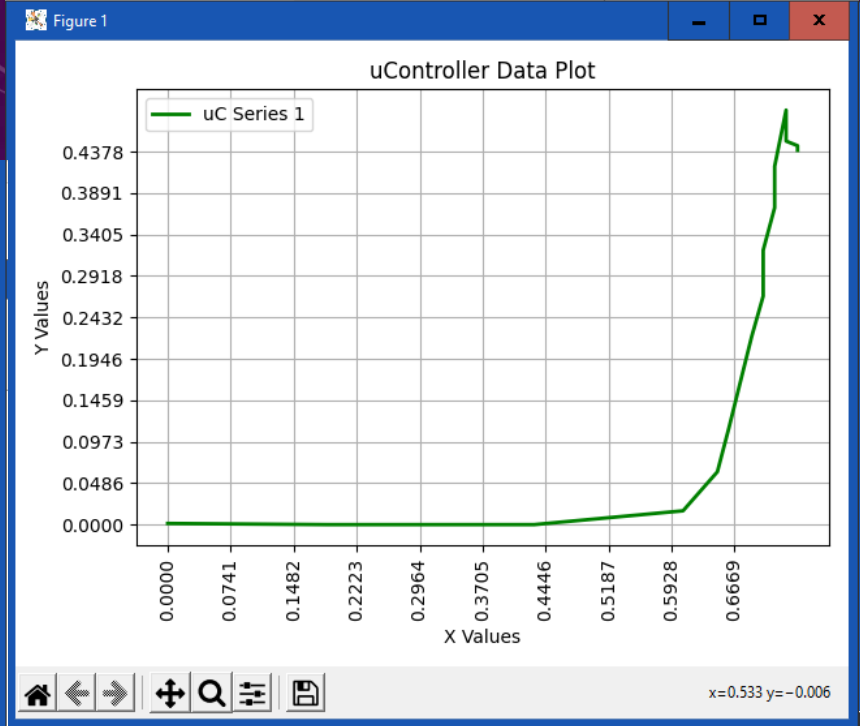


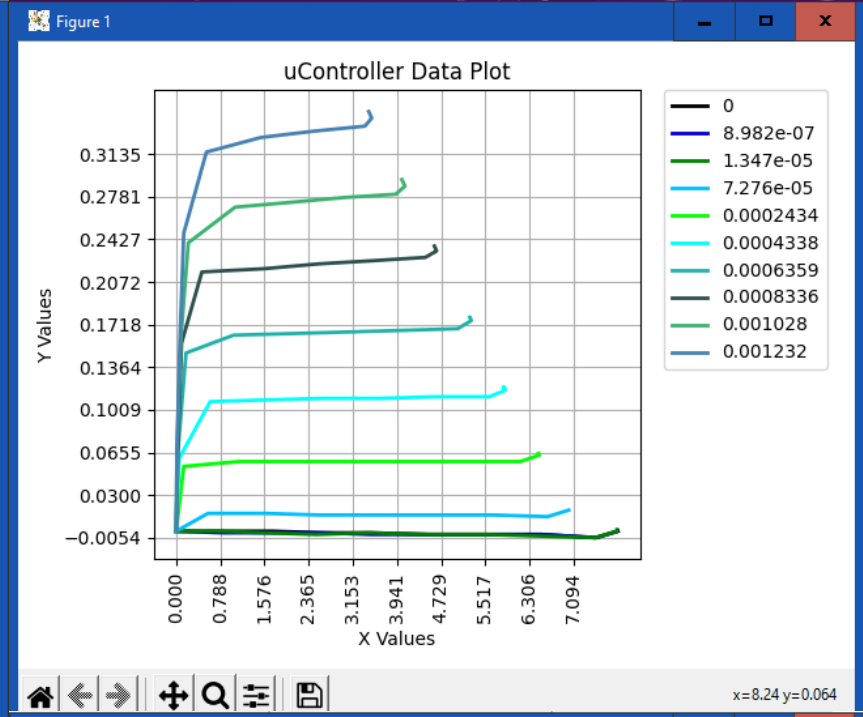


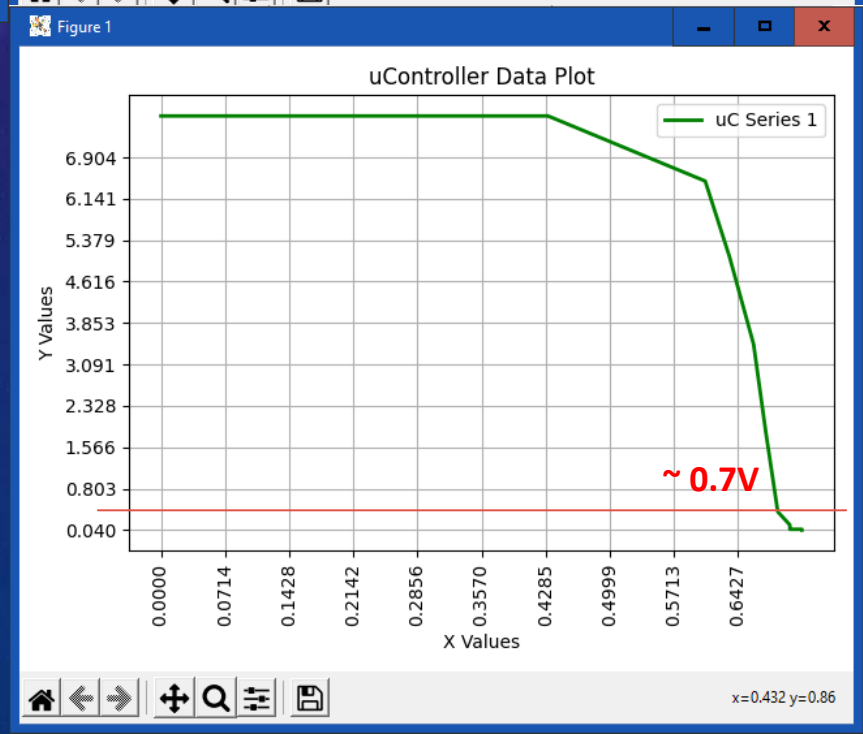
SYMBOL	TEST CONDITIONS	2N3053 MIN	2N3053 MAX	2N3053A MIN	2N3053A MAX	UNITS
$V_{CE(SAT)}$	I_C=150mA, I_B=15mA	-	1.4	-	0.3	V
$V_{BE(SAT)}$	I_C=150mA, I_B=15mA	-	1.7	0.6	1.0	V
$V_{BE(ON)}$	V_{CE}=2.5V, I_C=150mA	-	1.7	-	1.0	V
h_{FE}	V_{CE}=2.5V, I_C=150mA	25	-	25	-	
h_{FE}	V_{CE}=10V, I_C=150mA	50	250	50	250	

2SC5706







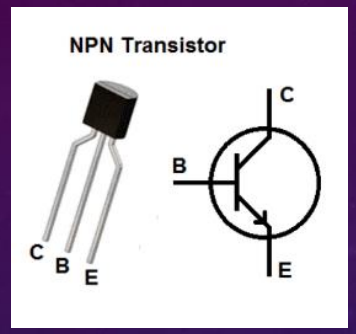


Parameter	Symbol	Conditions	Ratings			Unit
			min	typ	max	
Collector Cutoff Current	I_{CBO}	V_{CB}=(−)40V, I_E=0A			(−)1	μA
Emitter Cutoff Current	I_{EBO}	V_{EB}=(−)4V, I_C=0A			(−)1	μA
DC Current Gain	h_{FE}	V_{CE}=(−)2V, I_C=(−)500mA	200		560	
Gain-Bandwidth Product	f_T	V_{CE}=(−)10V, I_C=(−)500mA		(360)400		MHz
Output Capacitance	Cob	V_{CB}=(−)10V, f=1MHz		(24)15		pF
Collector-to-Emitter Saturation Voltage	V_{CE}(sat)1	I_C=(−)1A, I_B=(−)50mA		(−115)90	(−195)135	mV
	V_{CE}(sat)2	I_C=(−)2A, I_B=(−)100mA		(−255)160	(−430)240	mV
Base-to-Emitter Saturation Voltage	V_{BE}(sat)	V_{CE}=(−)2V, I_B=(−)100mA		(−)0.89	(−)1.2	V

MEASUREMENTS OF N-CHANNEL ENHANCEMENT MODE MOSFETS

TRANSISTOR OPERATION

NPN

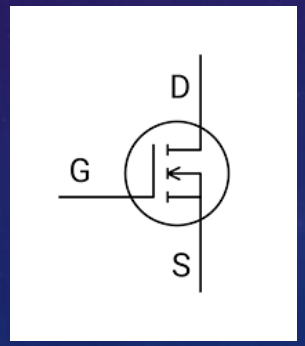


Feed Small Current into Base → NPN TRANSISTOR β

Get a LARGE current between Collector and Emitter

Datasheets today use h_{fe} instead of Beta. Beta sounds "cooler"

N-CH MOSFET



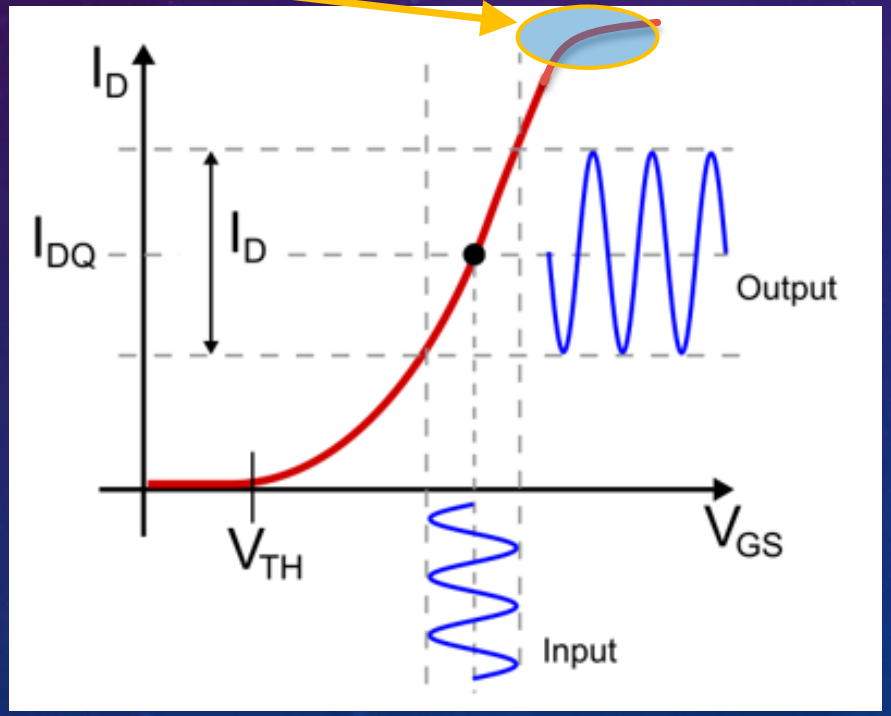
Feed Voltage above Threshold → NMOS TRANSISTOR g_m

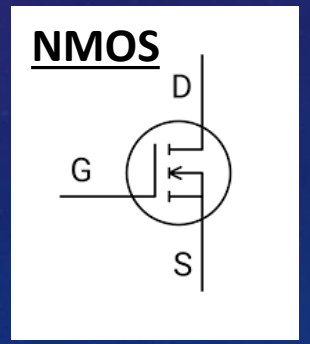
Get a LARGE current between Drain and Source

NMOS MOSFETS

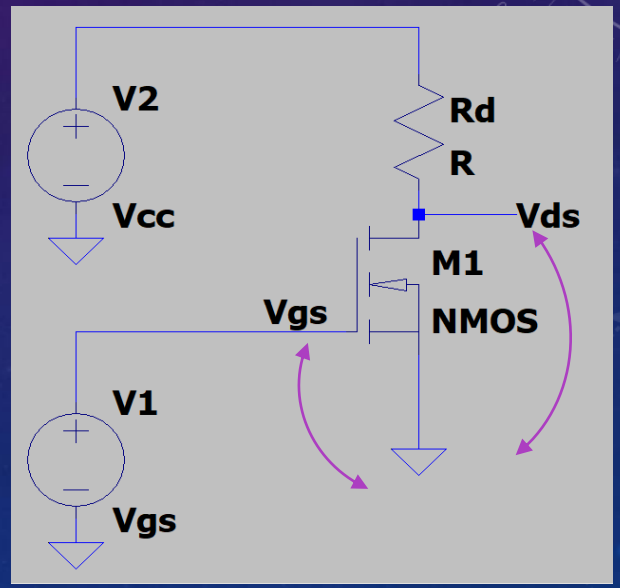
Similar to BJT:

- There is Vbe(turn on) for BJT, there is Vgs(on) or Vgs(th). This is threshold voltage at which device starts conducting
- There is a maximum Vbe or Vgs where current is at maximum and constant. Vbe(sat) and Vgs(*triode?*). ***For MOSFET, Not specified on data sheet!!***





MOSFET REGIONS: VOLTAGE



MOSFETS

- Similar Input and Output charts to BJT:
  - However, the input curve will show saturation if Vgs is high enough
- No concept of Beta (current amplification). MOSFET is a voltage device. Voltage controls current.
- The gain depends on g_m (transconductance) and K
- K is based on channel parameters inside MOSFET

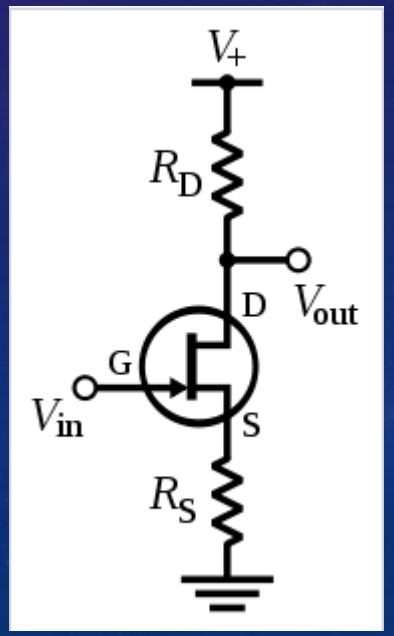
$$I_d = \frac{K}{2}(V_{gs} - V_t)^2 \text{ where } K = \mu C_{ox}\frac{W}{L}$$

$$K = 2\frac{I_d}{(V_{gs}-V_t)^2}$$

$$g_{m(sat.)} = \frac{\partial I_{D(sat.)}}{\partial V_{GS}} = \frac{W\mu_z C_{ox}}{L}(V_{GS} - V_{th})$$

Datasheets typically provide transconductance g_m

	Definition	Expression
Current gain	$A_\mathrm{i} \triangleq \frac{i_\mathrm{out}}{i_\mathrm{in}}$	∞
Voltage gain	$A_\mathrm{v} \triangleq \frac{v_\mathrm{out}}{v_\mathrm{in}}$	$-\frac{g_\mathrm{m} R_\mathrm{D}}{1+g_\mathrm{m} R_\mathrm{S}}$
Input impedance	$r_\mathrm{in} \triangleq \frac{v_\mathrm{in}}{i_\mathrm{in}}$	∞
Output impedance	$r_\mathrm{out} \triangleq \frac{v_\mathrm{out}}{i_\mathrm{out}}$	R_D



NMOS PLOTS

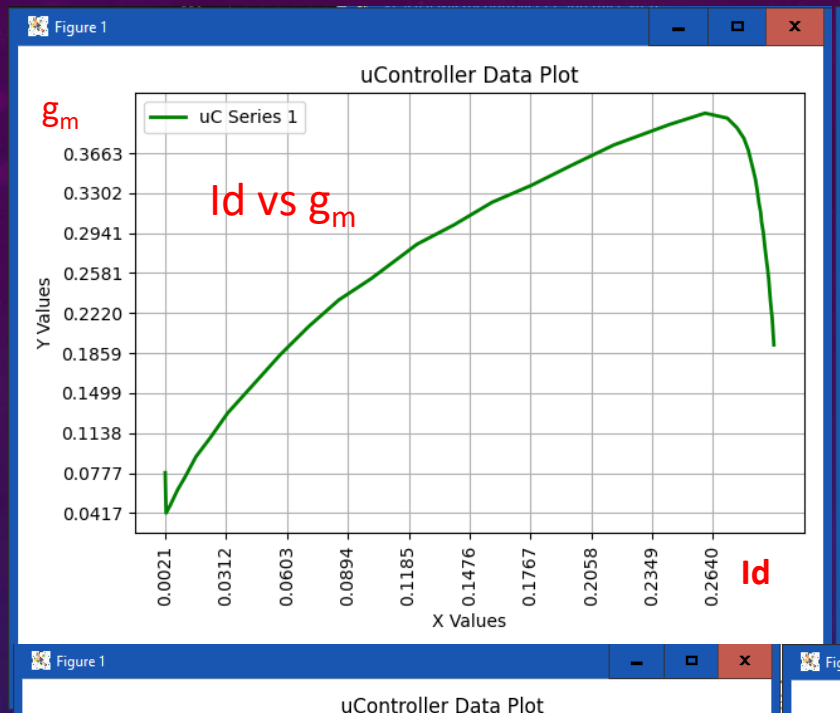


Figure 1
uController Data Plot
g_m
uC Series 1
Id vs g_m
Y Values
0.3663
0.3302
0.2941
0.2581
0.2220
0.1859
0.1499
0.1138
0.0777
0.0417
0.0021
0.0312
0.0603
0.0894
0.1185
0.1476
0.1767
0.2058
0.2349
0.2640
X Values
Id

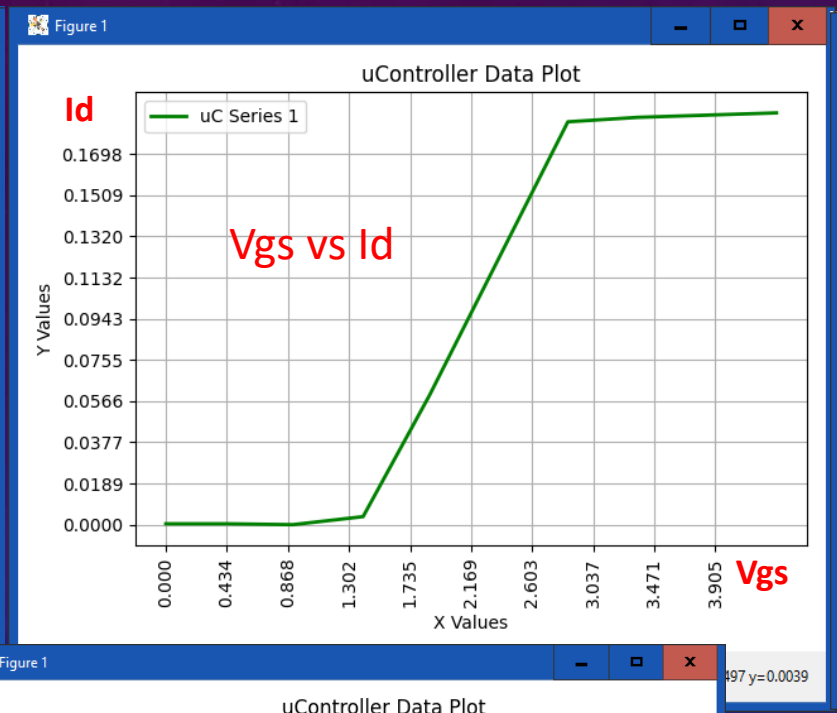


Figure 1
uController Data Plot
Id
uC Series 1
Vgs vs Id
Y Values
0.1698
0.1509
0.1320
0.1132
0.0943
0.0755
0.0566
0.0377
0.0189
0.0000
0.000
0.434
0.868
1.302
1.735
2.169
2.603
3.037
3.471
3.905
X Values
Vgs

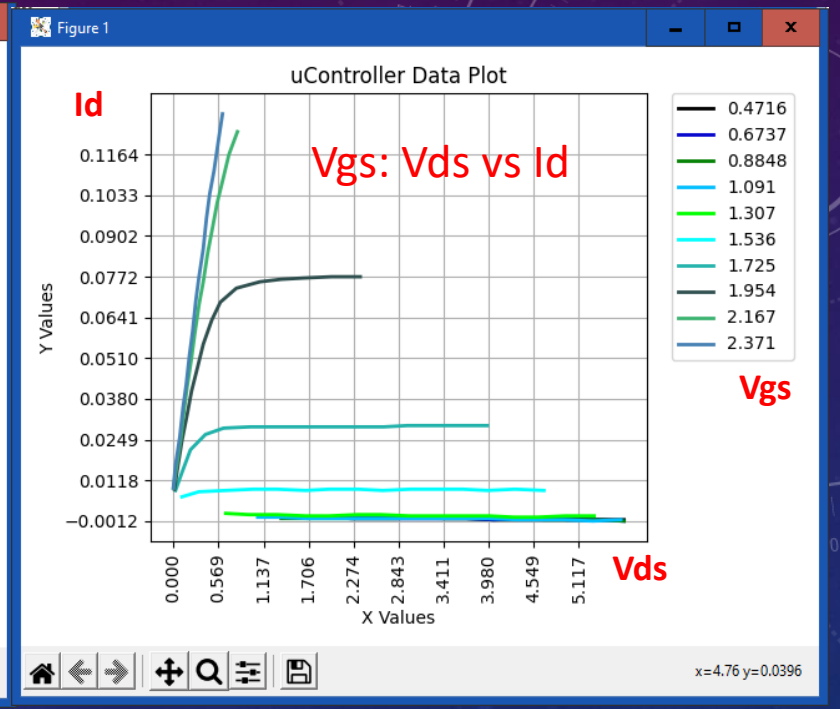


Figure 1
uController Data Plot
Id
Vgs: Vds vs Id
Y Values
0.1164
0.1033
0.0902
0.0772
0.0641
0.0510
0.0380
0.0249
0.0118
−0.0012
0.000
0.569
1.137
1.706
2.274
2.843
3.411
3.980
4.549
5.117
X Values
Vds
0.4716
0.6737
0.8848
1.091
1.307
1.536
1.725
1.954
2.167
2.371
Vgs
x=4.76 y=0.0396

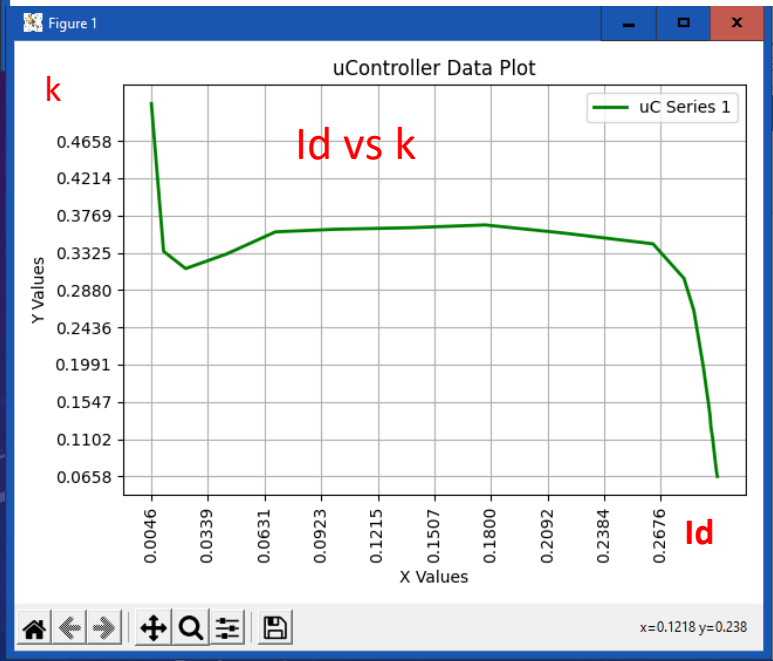


Figure 1
uController Data Plot
k
uC Series 1
Id vs k
Y Values
0.4658
0.4214
0.3769
0.3325
0.2880
0.2436
0.1991
0.1547
0.1102
0.0658
0.0046
0.0339
0.0631
0.0923
0.1215
0.1507
0.1800
0.2092
0.2384
0.2676
X Values
Id
x=0.1218 y=0.238

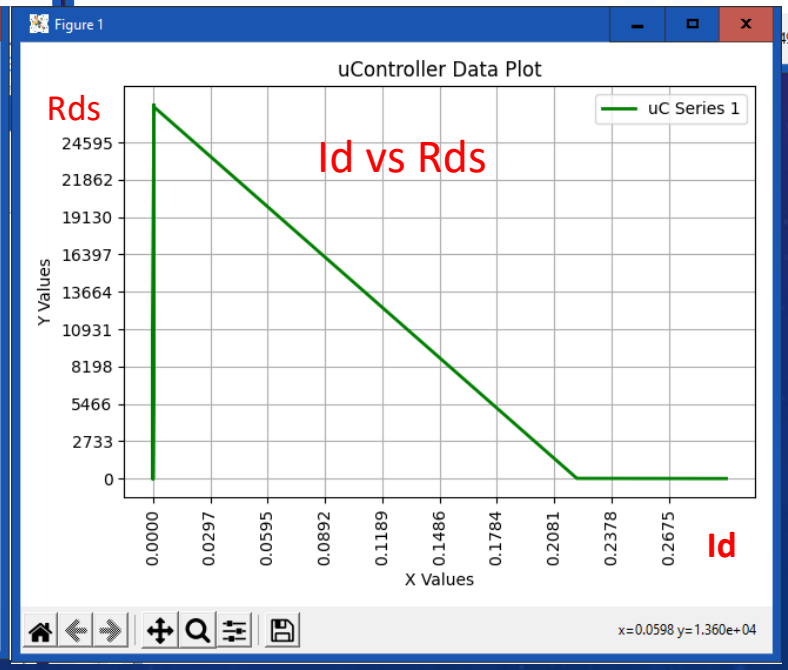
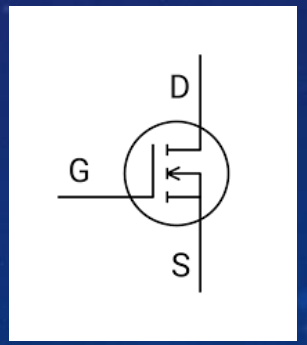
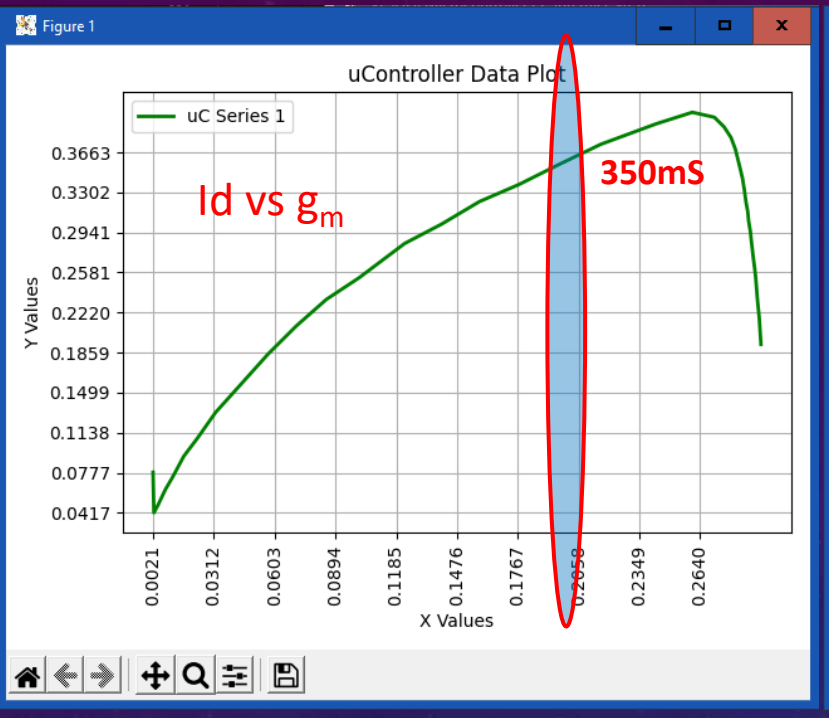
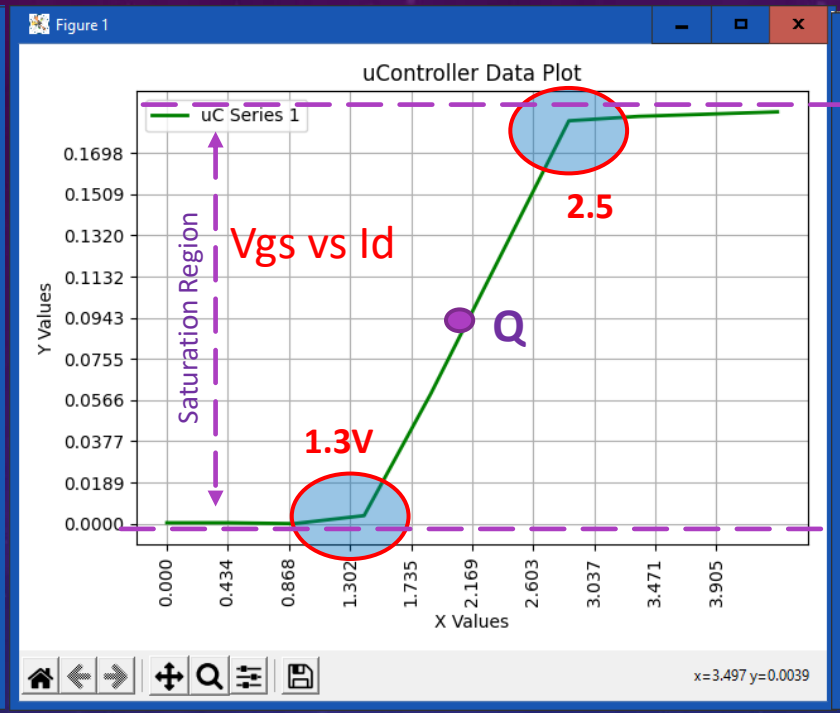


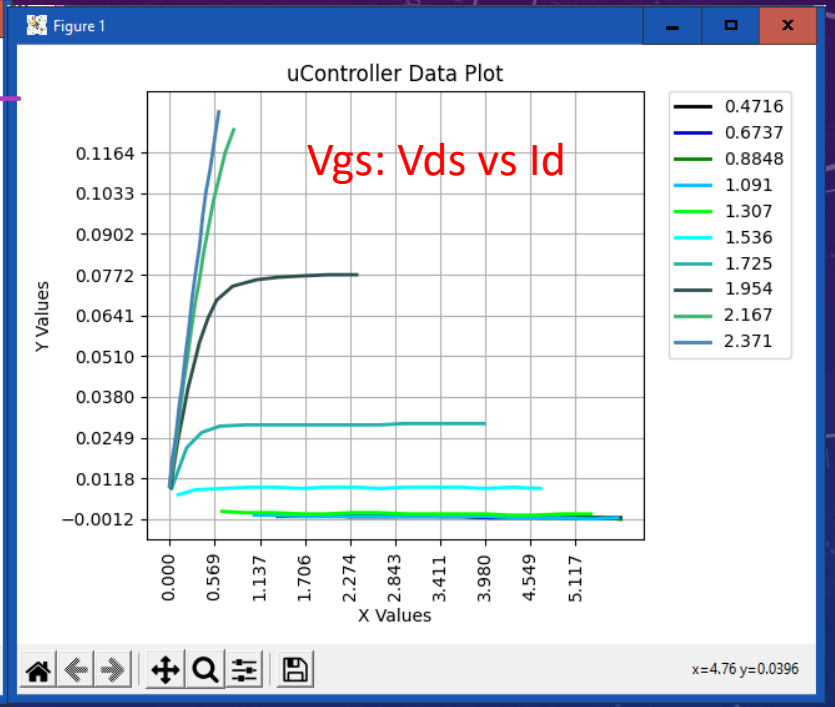
Figure 1
uController Data Plot
Rds
uC Series 1
Id vs Rds
Y Values
24595
21862
19130
16397
13664
10931
8198
5466
2733
0
0.0000
0.0297
0.0595
0.0892
0.1189
0.1486
0.1784
0.2081
0.2378
0.2675
X Values
Id
x=0.0598 y=1.360e+04


D
G
S

2N7000



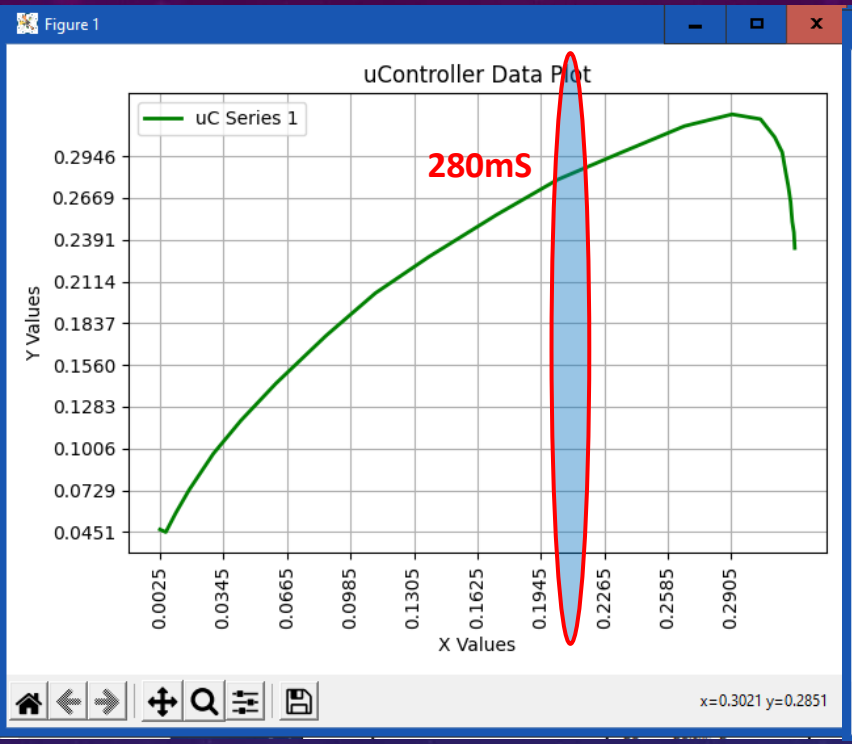


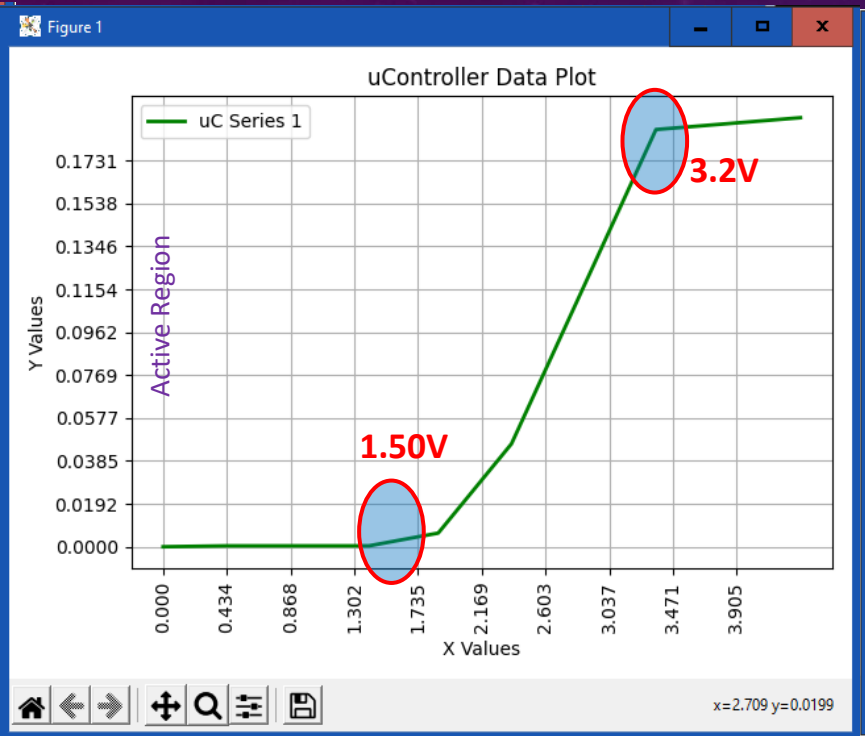


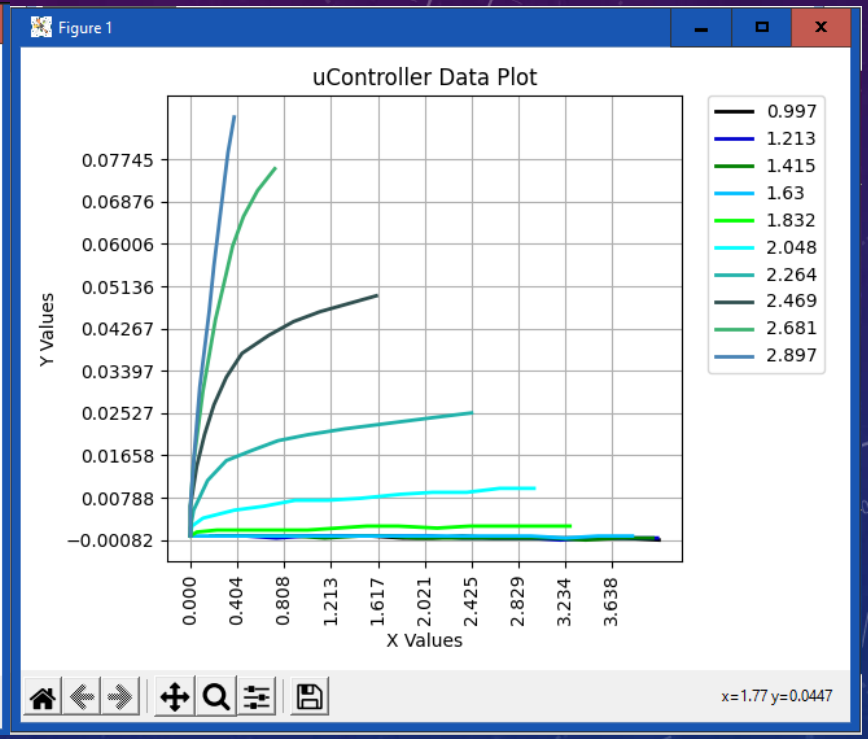
FET K Sweep
Vg 0.00 - 4.60 inc: 0.1150
Getting Vgs(on), Ids(on)
VgControl(on): 1.2650 Vgs(on): 1.2665 Ids(on): 0.0016 Vgs(th): 2.4926 Rds(th): 4.8311 k: 0.0011

On Characteristics							
$V_{GS(th)}$	Gate Threshold Voltage	V_{DS} = V_{GS}, I_D = 1 mA	2N7000	0.8	2.1	3	V
		V_{DS} = V_{GS}, I_D = 250 µA	2N7002 NDS7002A	1	2.1	2.5	
$R_{DS(ON)}$	Static Drain-Source On-Resistance	V_{GS} = 10 V, I_D = 500 mA	2N7000		1.2	5	Ω
		V_{GS} = 10 V, I_D = 500 mA, T_C = 125°C			1.9	9	
g_{FS}	Forward Transconductance	V_{DS} = 10 V, I_D = 200 mA	2N7000	100	320		mS

BS170





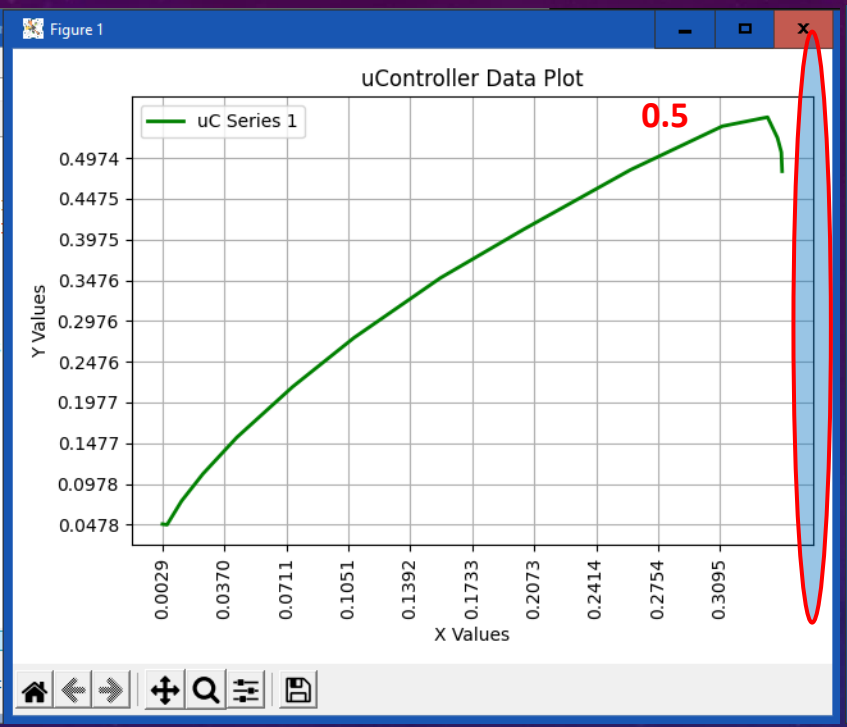


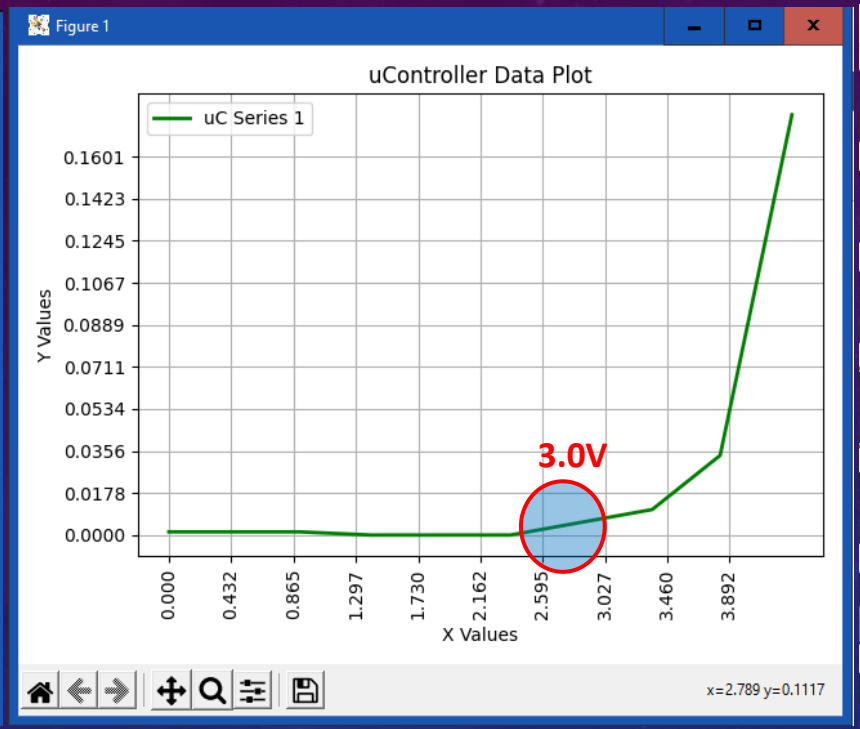
FET K Sweep
Vg 0.00 - 4.60 inc: 0.1150
Getting Vgs(on), Ids(on)
VgControl(on): 1.4950 Vgs(on): 1.5090 Ids(on): 0.0008 Vgs(th): 3.2202 Rds(th): 4.1707 k: 0.0003

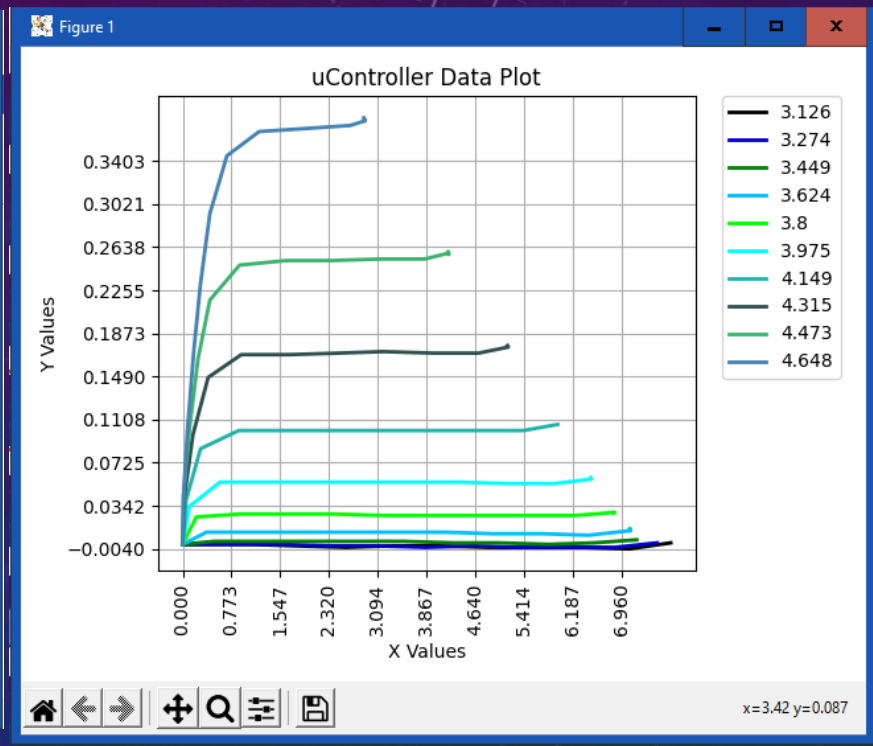
ON CHARACTERISTICS (Note 1)

$V_{GS(th)}$	Gate Threshold Voltage	$V_{DS} = V_{GS}$, $I_D = 1$ mA	All	0.8	2.1	3	V
$R_{DS(ON)}$	Static Drain–Source On–Resistance	$V_{GS} = 10$ V, $I_D = 200$ mA	All	–	1.2	5	Ω
g_{FS}	Forward Transconductance	$V_{DS} = 10$ V, $I_D = 200$ mA	BS170	–	320	–	mS
		$V_{DS} \geq 2\ V_{DS(on)}$, $I_D = 200$ mA	MMBF170	–	320	–	

IRF510





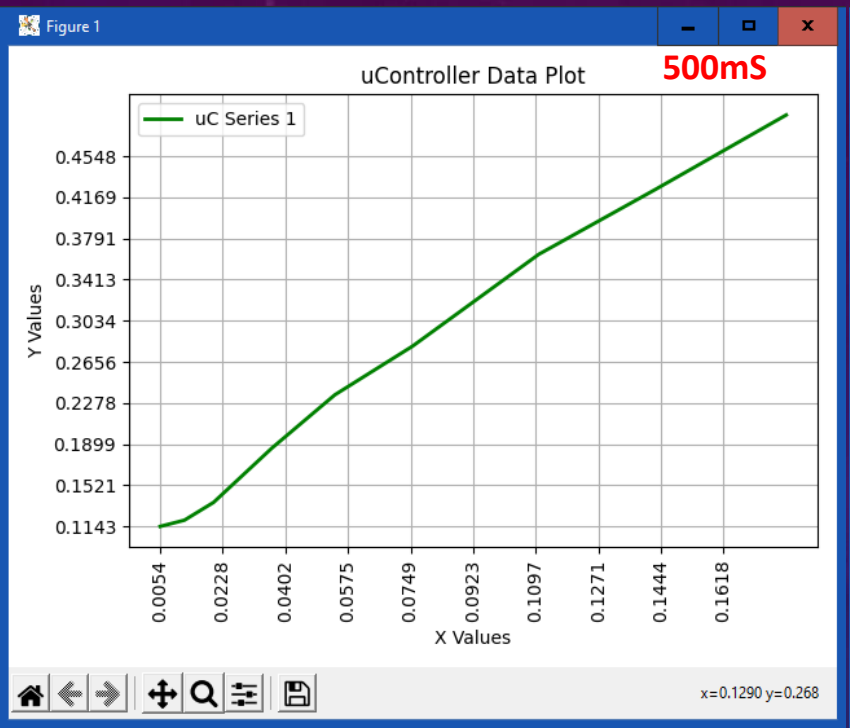


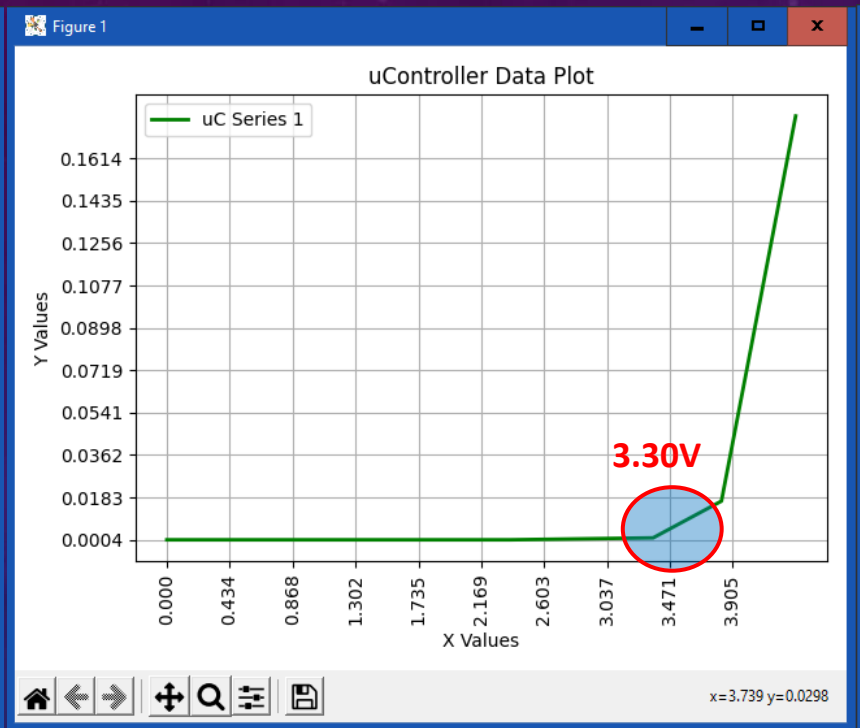
FET K Sweep
Vg 0.00 - 4.60 inc: 0.1150
Getting Vgs(on), Ids(on)
VgControl(on): 2.9900 Vgs(on): 3.0989 Ids(on): 0.0027 Vgs(th): 4.8235 Rds(th): 0.2865 K: 0.0009

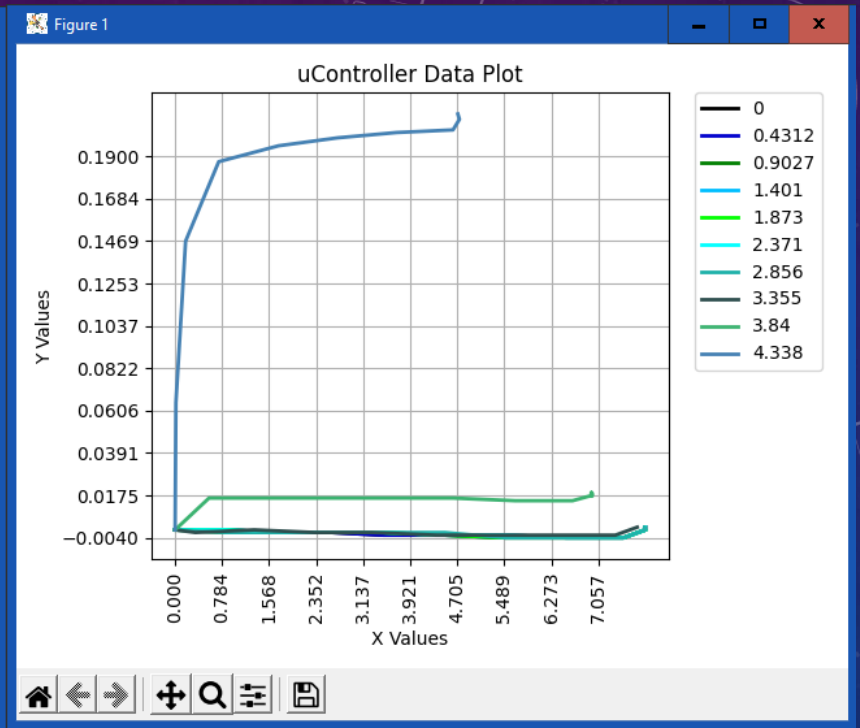
Don't have the drive to measure this MOSFET

Gate-source threshold voltage	$V_{GS(th)}$	V_{DS} = V_{GS}, I_D = 250 µA	2.0	-	4.0	V
Gate-source leakage	I_{GSS}	V_{GS} = ± 20 V	-	-	± 100	nA
Zero gate voltage drain current	I_{DSS}	V_{DS} = 100 V, V_{GS} = 0 V	-	-	25	µA
		V_{DS} = 80 V, V_{GS} = 0 V, T_J = 150 °C	-	-	250	
Drain-source on-state resistance	$R_{DS(on)}$	V_{GS} = 10 V, I_D = 3.4 A [b]	-	-	0.54	Ω
Forward transconductance	g_{fs}	V_{DS} = 50 V, I_D = 3.4 A [b]	1.3	-	-	S

IRFZ20





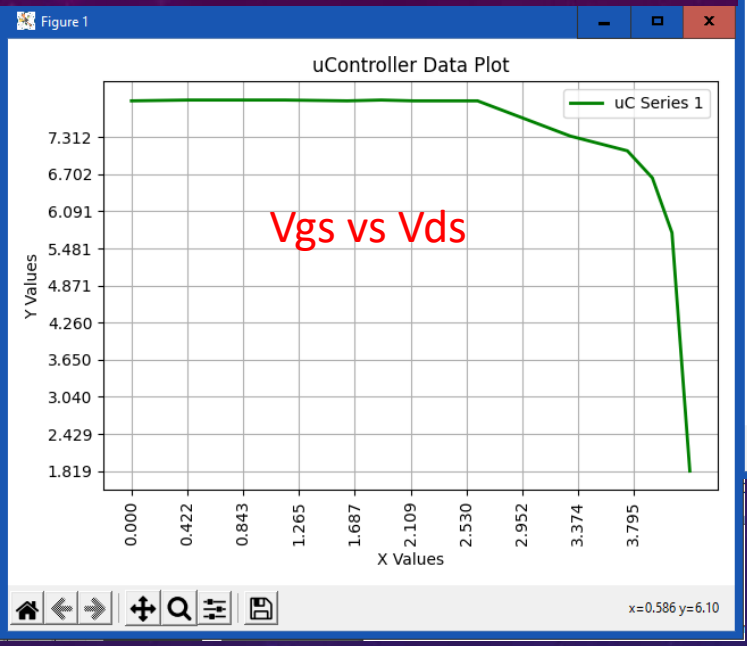


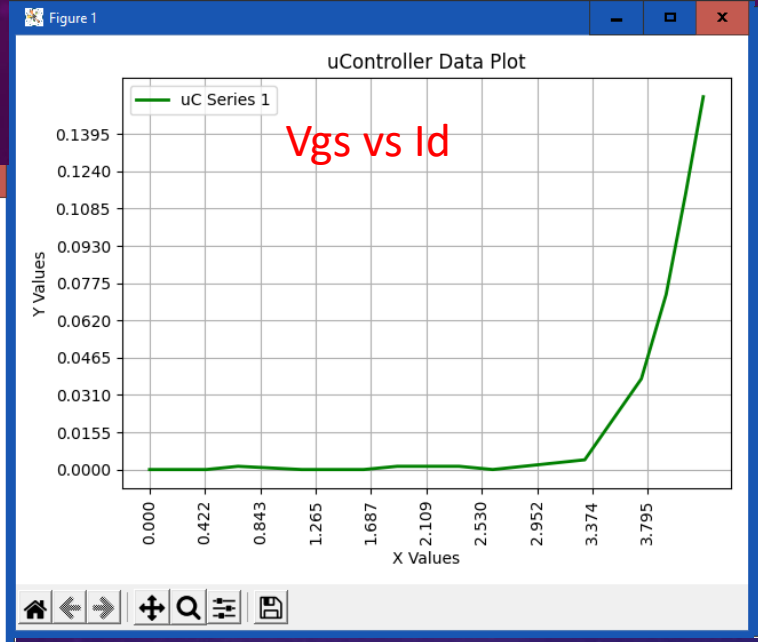
FET K Sweep
Vg 0.00 - 4.60 inc: 0.1150
Getting Vgs(on), Ids(on)
VgControl(on): 3.2200 Vgs(on): 3.3549 Ids(on): 0.0027 Vgs(th): 4.8235 Rds(th): 0.2865 k: 0.0012

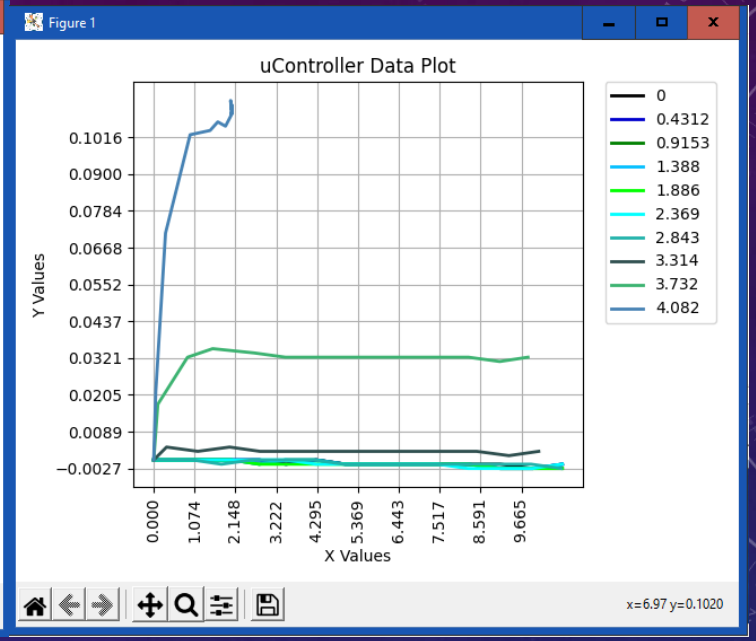
Don't have the drive to measure this MOSFET

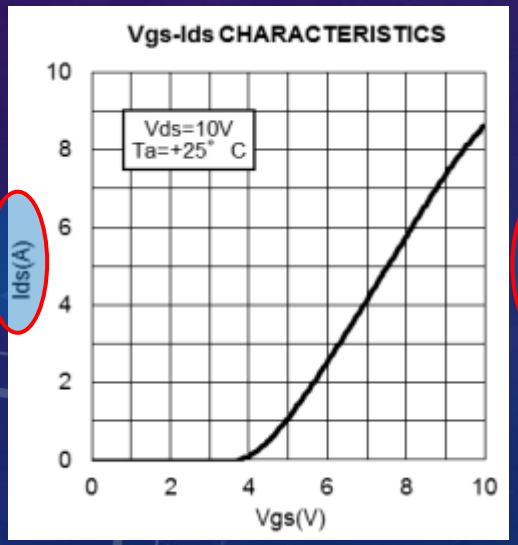
	V_{DS}	V_{GS} = 0 V, I_D = 250 µA	50	-	-	V
Gate-Source Threshold Voltage	$V_{GS(th)}$	V_{DS} = V_{GS}, I_D = 250 µA	2.0	-	4.0	V
Gate-Source Leakage	I_{GSS}	V_{GS} = ± 20 V	-	-	± 500	nA
Zero Gate Voltage Drain Current	I_{DSS}	V_{DS} > Max. Rating, V_{GS} = 0 V	-	-	250	µA
		V_{DS} = Max. Rating x 0.8, V_{GS} = 0 V, T_C = 125 °C	-	-	1000	
On-State Drain Current	$I_{D(on)}$	V_{GS} = 10 V; V_{DS} > $I_{D(on)}$ x $R_{DS(on)}$ max.	-	-	15	A
Drain-Source On-State Resistance[b]	$R_{DS(on)}$	V_{GS} = 10 V; I_D = 10 A	-	0.080	0.10	Ω
Forward Transconductance[b]	g_{fs}	V_{DS} > $I_{D(on)}$ x $R_{DS(on)}$ max., I_D = 9.0 A	5.0	6.0	-	S

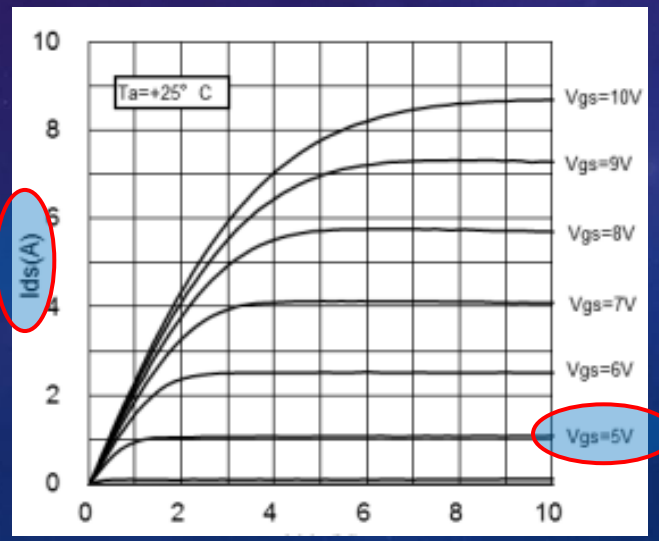
RD16HHF1

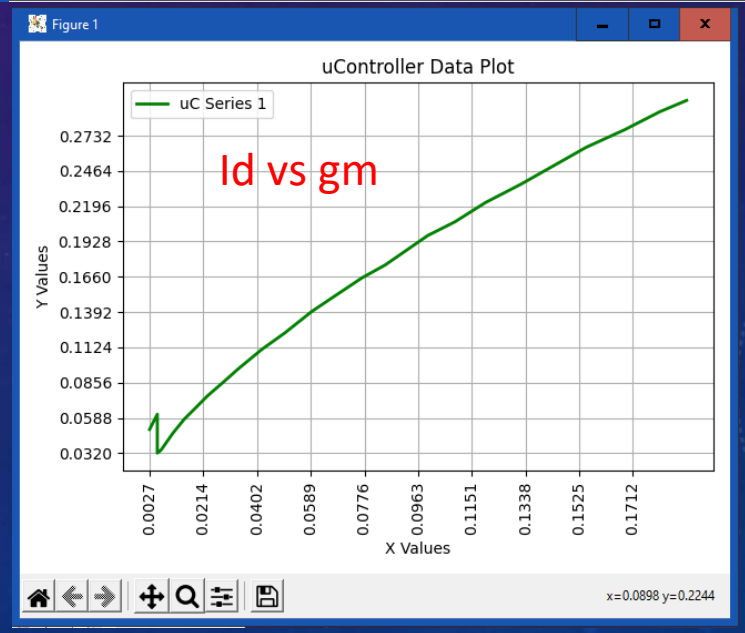








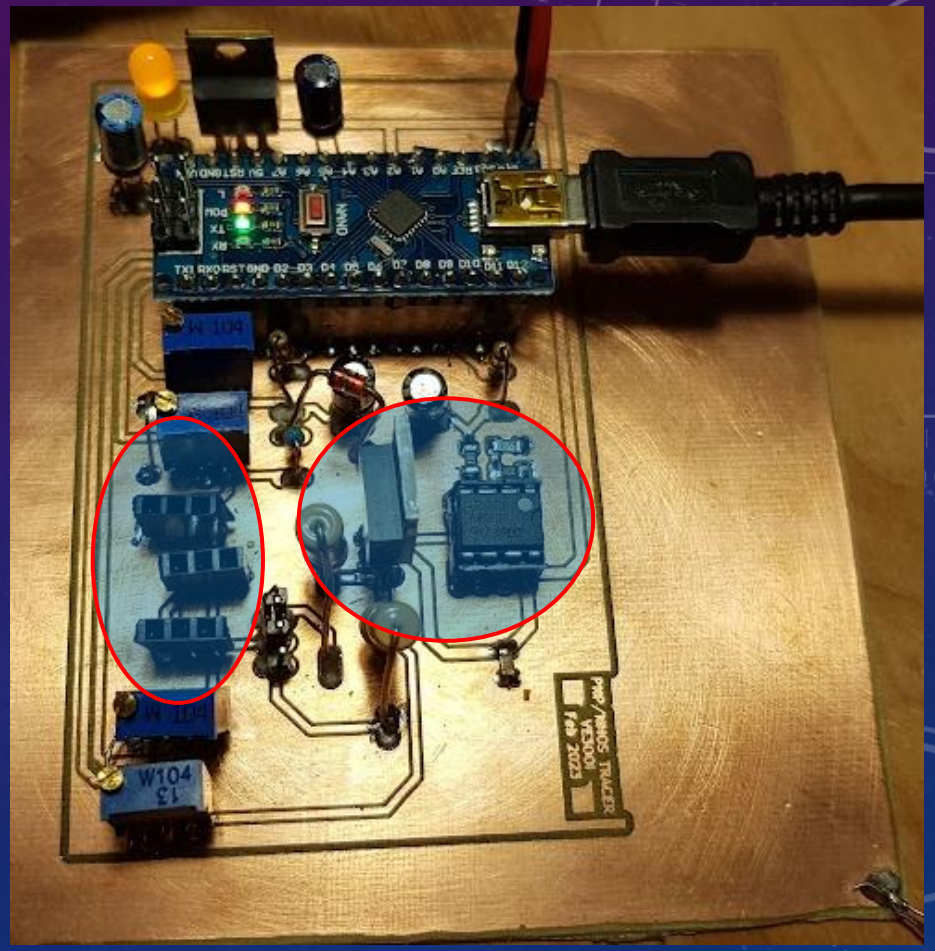




Don’t have the drive to measure this MOSFET

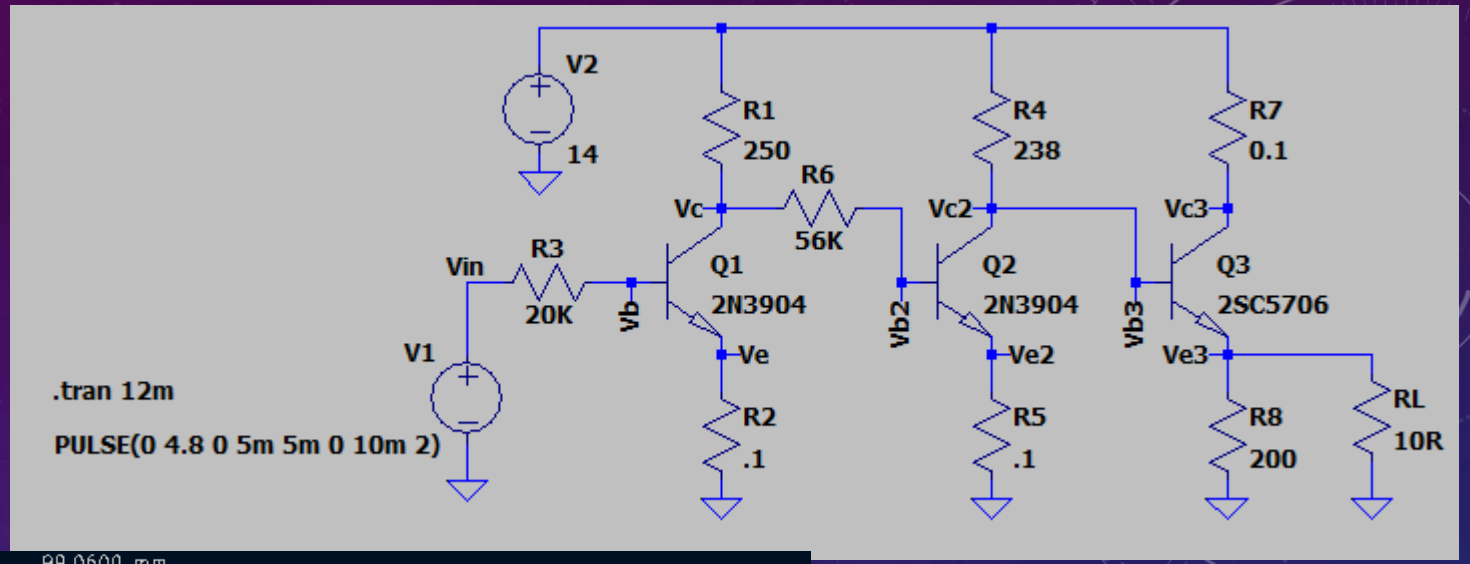
ISSUES

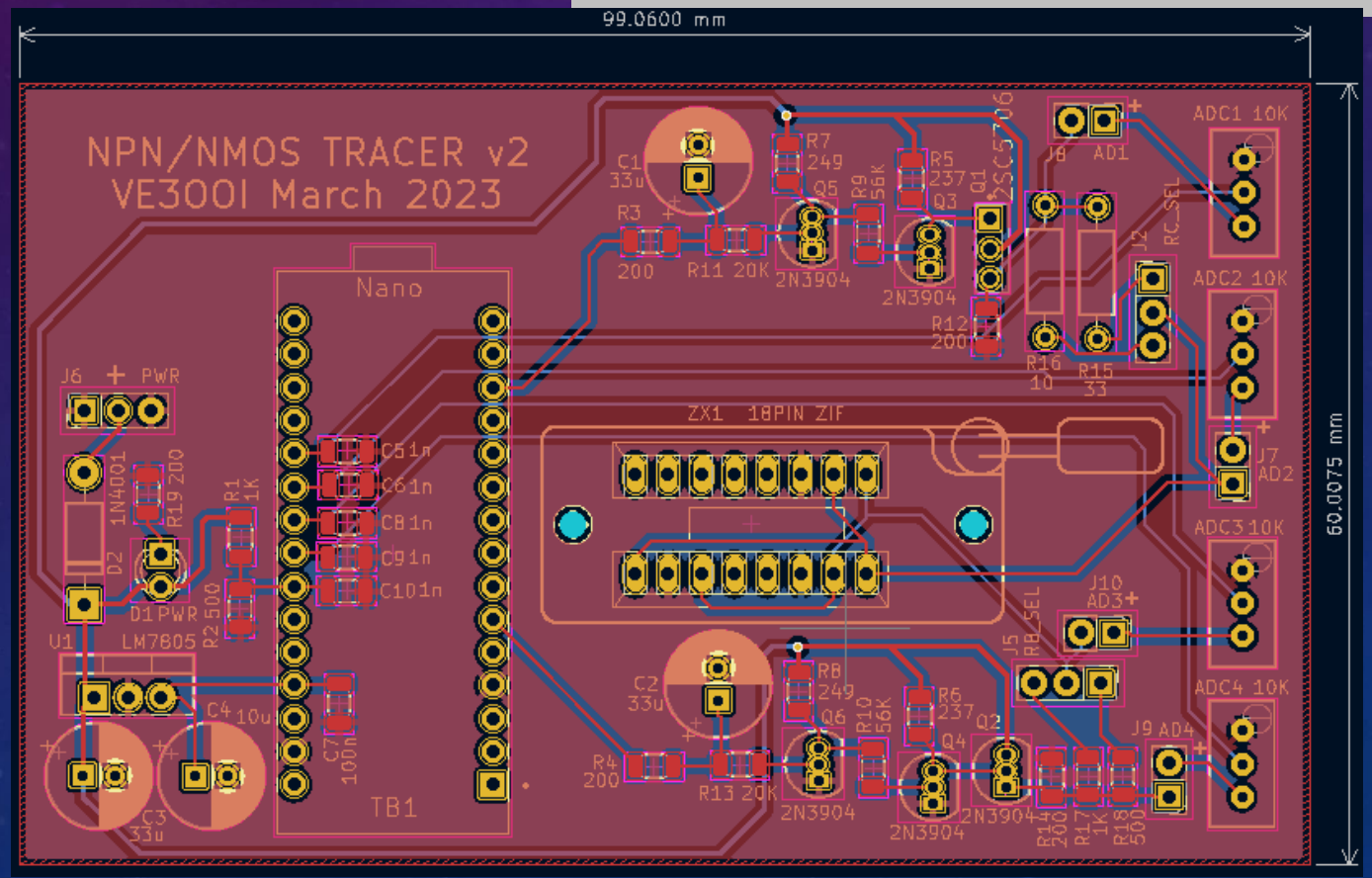
- Need higher current and voltage for Vcc and Vb
  - ❖ Voltage drop from Opamp and TIP120 (~8V)
  - ❖ Using PWM (~4.3V) for Vb
- Need better way to sweep Vcc/Vc
- Need better socket for transistors



VERSION 2

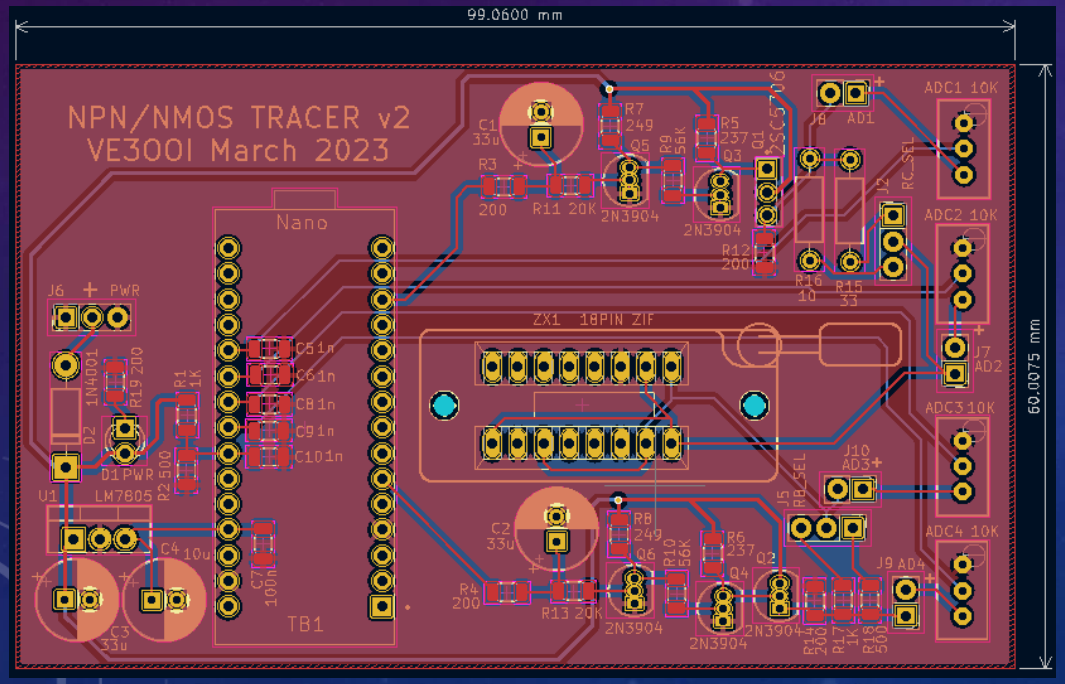
- ZIF Socket
- DC Voltage Amplifier





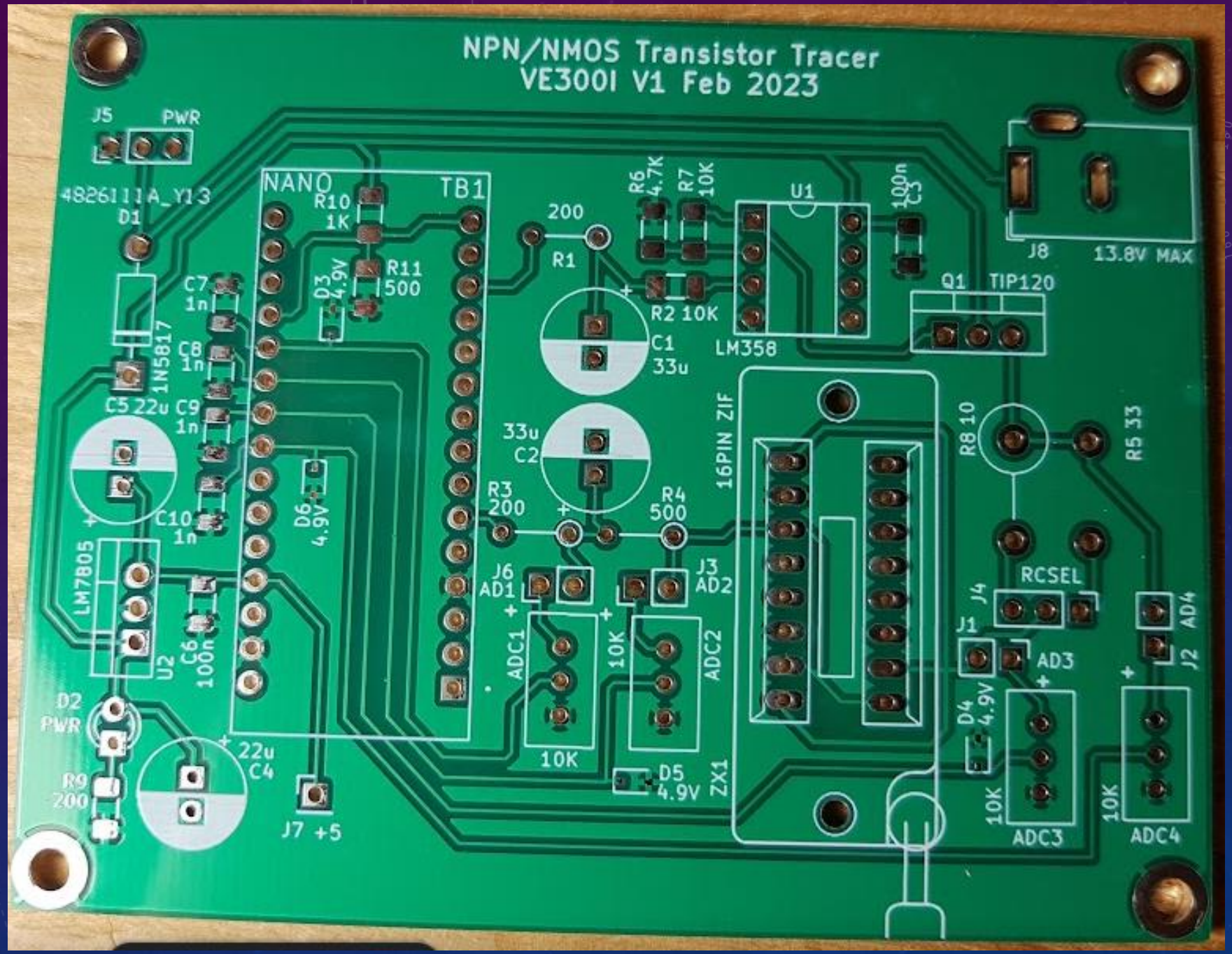
VERSION 2: SPEEDS AND FEEDS

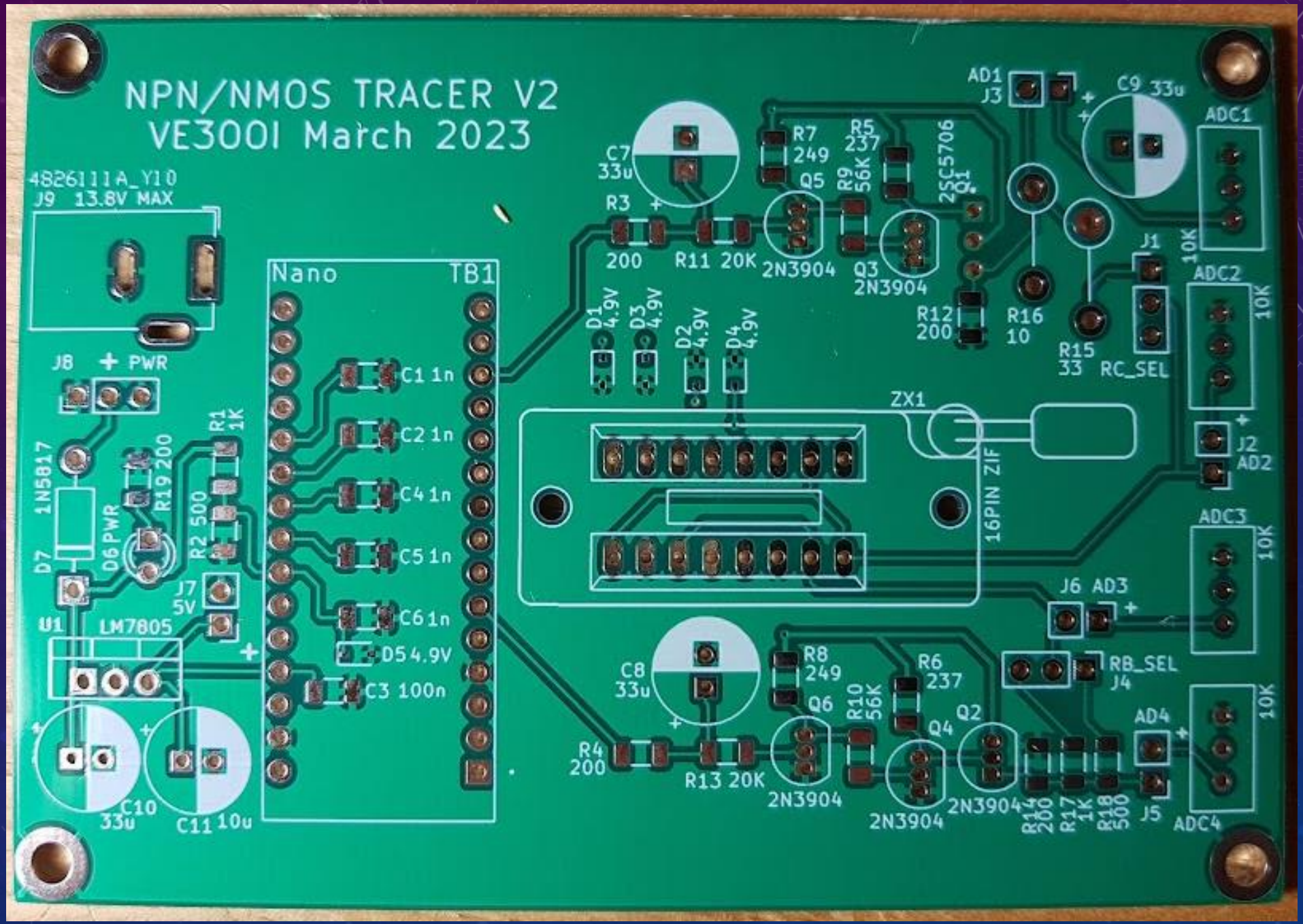
- ✓ **NPN** BJTs with EBC, BEC, BCE
- ✓ **NMOS** (N Channel Enhancement Mode MOSFET) with GDS, GSD, DGS
- ✓ Vcc 14V, Max Collector/Drain Voltage: 11.1@1.7A
- ✓ Vcc 14V, Max Base/Gate voltage: 11.1V@22mA



Version 1:

- ✓ Vcc 14V, Max Collector/Drain Voltage: 10@930mA
- ✓ Max Base/Gate voltage: 4.30@7mA


NPN/NMOS Transistor Tracer
VE3001 V1 Feb 2023
J5
PWR
4826111A_Y13
D1
1N5817
C5 22u
C7 1n
C8 1n
C9 1n
C10 1n
LM7805
U2
C6 100n
D2 PWR
R9 200
22u C4
J7 +5
NANO
TB1
R10 1K
R11 500
D3 4.9V
D6 4.9V
200 R1
R6 4.7K
R7 10K
R2 10K
C1 33u
33u C2
R3 200
R4 500
J6 AD1
J3 AD2
ADC1
ADC2
10K
10K
D5 4.9V
ZX1
U1
LM358
100n C3
J8
13.8V MAX
Q1 TIP120
16PIN ZIF
R8 10
R5 33
RCSEL
J4
J1
AD3
D4 4.9V
10K
ADC3
J2 AD4
10K
ADC4


NPN/NMOS TRACER V2
VE3001 March 2023
4826111A_Y10
J9 13.8V MAX
Nano
TB1
J8 + PWR
1N5817
D7
D6 PWR
R19 200
R1 1K
R2 500
J7 5V
U1
LM7805
C10 33u
C11 10u
C1 1n
C2 1n
C4 1n
C5 1n
C6 1n
D5 4.9V
C3 100n
C7 33u
R3 200
R11 20K
R7 249
R5 237
R9 56K
Q5 2N3904
Q3 2N3904
2SC5706
Q1
D1 4.9V
D3 4.9V
D2 4.9V
D4 4.9V
R12 200
R16 10
R15 33
RC_SEL
J1
AD1
J3
C9 33u
ADC1
ADC2
ADC3
ADC4
10K
J2 AD2
ZX1
16PIN ZIF
J6 AD3
RB_SEL
J4
AD4
J5
C8 33u
R4 200
R13 20K
R8 249
R6 237
R10 56K
Q6 2N3904
Q4 2N3904
Q2 2N3904
R14 200
R17 1K
R18 500


Fin