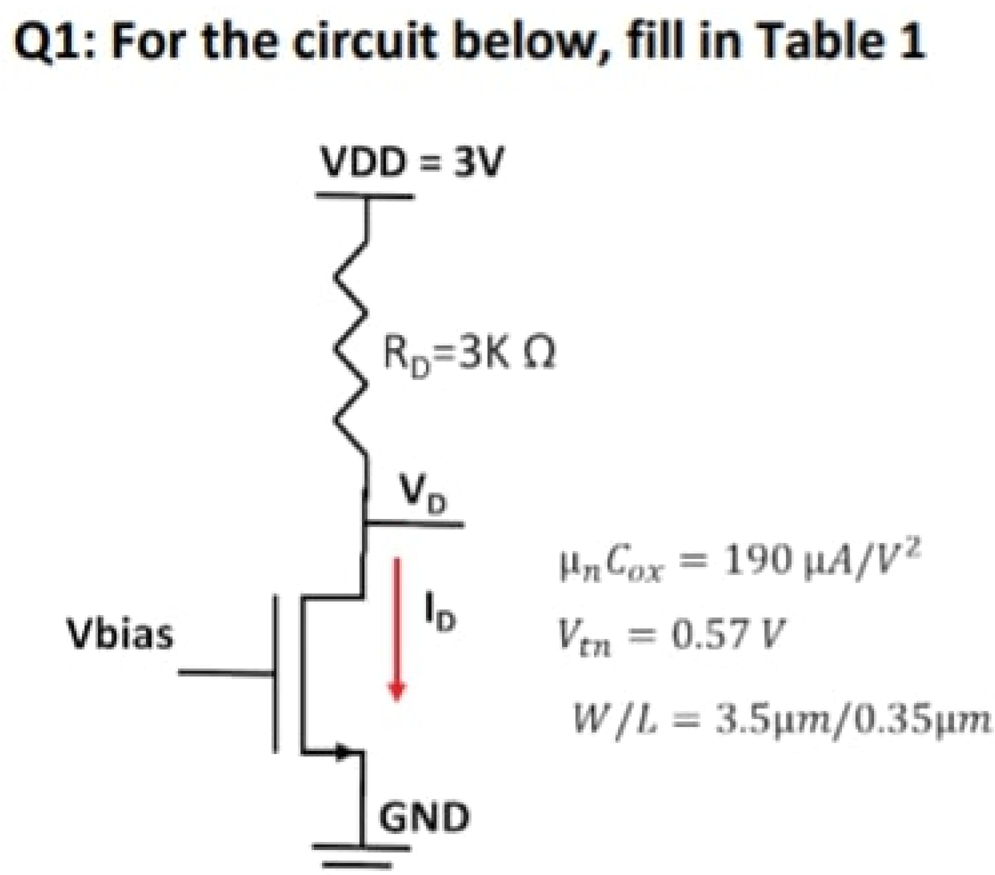
Activity 1

September 5, 2020

1 Activity 1

Class activity. Solve for current.

1.0.1 Problem 1



V_{bias}	MOSFET Operation Mode	I_D (mA)	V_D (V)
0.5	OFF	0	3
1.0	SAT	.175	2.47

V_{bias}	MOSFET Operation Mode	I_D (mA)	V_D (V)
1.2	SAT	.377	1.87
1.5	SAT	.77	.69
2.0	LINEAR	.87	.37
3.0	LINEAR	.93	.21

[49]:
```
import cmath

# saturation
def saturation_current (VGS, VTH, K,fet_mode=""):
    if fet_mode=="sat":
        print("The saturation current is: {} mA".format(0.
 ↪5*K*(VGS-VTH)**2*10**3))
        return 0.5*K*(VGS-VTH)**2
    return None
def drain_voltage (VCC=None, ID=None, RD=None, VGS=None, VTH=None, VDS=None,␣
 ↪fet_mode=''):
    if fet_mode=="sat":
        VDS = round(VCC-ID*RD,3)
        mode_true = VDS > (VGS-VTH)
        print("VDS = {} V, the assumption is {}.".format(VDS, mode_true))
        return VDS
    elif fet_mode=='triode':
        mode_true = VDS < (VGS-VTH)
        print("VDS = {} V, the assumption is {}.".format(VDS, mode_true))
        return VDS
    return None

# Triode
def quadratic_formula (a,b,c):
    d = (b**2) - (4*a*c)
    sol1 = (-b-cmath.sqrt(d))/(2*a)
    sol2 = (-b+cmath.sqrt(d))/(2*a)
    print('The solution are {0} and {1}'.format(sol1,sol2))
    return [sol1, sol2]
def current_post_quad (VCC, VDS, RD):
    ID = (VCC-VDS)/RD
    print('ID = {} A'.format((VCC-VDS)/RD))
    return ID
```

[50]:
```
W = 3.5
L = 0.35
K_prime = 190*10**(-6)
Kn = W/L*K_prime
VGS = 1
VTH = 0.57
```

```
ID = saturation_current(VGS,VTH,Kn,fet_mode='sat')
VCC = 3
RD=3000
#verification
drain_voltage (VCC=VCC, ID=ID, RD=RD, VGS=VGS,VTH=VTH, fet_mode='sat')
```

```
The saturation current is: 0.175655 mA
VDS = 2.473 V, the assumption is True.
```

[50]: 2.473

[51]:
```
W = 3.5
L = 0.35
K_prime = 190*10**(-6)
Kn = W/L*K_prime
VGS = 1.2
VTH = 0.57
ID = saturation_current(VGS,VTH,Kn,fet_mode='sat')
VCC = 3
RD=3000
#verification
drain_voltage (VCC=VCC, ID=ID, RD=RD, VGS=VGS,VTH=VTH, fet_mode='sat')
```

```
The saturation current is: 0.377055 mA
VDS = 1.869 V, the assumption is True.
```

[51]: 1.869

[52]:
```
W = 3.5
L = 0.35
K_prime = 190*10**(-6)
Kn = W/L*K_prime
VGS = 1.5
VTH = 0.57
ID = saturation_current(VGS,VTH,Kn,fet_mode='sat')
VCC = 3
RD=3000
#verification
drain_voltage (VCC=VCC, ID=ID, RD=RD, VGS=VGS,VTH=VTH, fet_mode='sat')

# verification fail -> so it's triode
# VDS
a = Kn/2
b = -1*(1/RD+Kn*(VGS-VTH))
c = VCC/RD
Vd1, Vd2 = quadratic_formula(a=a,b=b,c=c)
VDS = min(abs(Vd1),abs(Vd2))
```

```python
print('The valid VDS here is: {} V'.format(VDS))
# ID
ID = current_post_quad(VCC=VCC, VDS=VDS, RD=RD)
```

```
The saturation current is: 0.821655 mA
VDS = 0.535 V, the assumption is False.
The solution are (0.6939013422053886+0j) and (1.5169758507770679+0j)
The valid VDS here is: 0.6939013422053886 V
ID = 0.0007686995525982037 A
```

[53]:

```python
# input parameter
VCC = 3
RD=3000
W = 3.5
L = 0.35
K_prime = 190*10**(-6)
Kn = W/L*K_prime
VGS = 2
VTH = 0.57

# assumption 1: saturation
ID = saturation_current(VGS,VTH,Kn,fet_mode='sat')
# verification for saturation
drain_voltage (VCC=VCC, ID=ID, RD=RD, VGS=VGS,VTH=VTH, fet_mode='sat')

# assumption 2: triode
# Solving for VDS
a = Kn/2
b = -1*(1/RD+Kn*(VGS-VTH))
c = VCC/RD
Vd1, Vd2 = quadratic_formula(a=a,b=b,c=c)
VDS = min(abs(Vd1),abs(Vd2))
print('The valid VDS here is: {} V'.format(VDS))
# Solving for ID
ID = current_post_quad(VCC=VCC, VDS=VDS, RD=RD)
# verification for triode
VDS = drain_voltage(VDS=VDS, VGS=VGS, VTH=VTH, fet_mode="triode")
```

```
The saturation current is: 1.9426550000000005 mA
VDS = -2.828 V, the assumption is False.
The solution are (0.3706100625820233+0j) and (2.8402671304004334+0j)
The valid VDS here is: 0.3706100625820233 V
ID = 0.0008764633124726589 A
VDS = 0.3706100625820233 V, the assumption is True.
```

[54]:

```python
# input parameter
VCC = 3
```

```
RD=3000
W = 3.5
L = 0.35
K_prime = 190*10**(-6)
Kn = W/L*K_prime
VGS = 3
VTH = 0.57

# assumption 1: saturation
ID = saturation_current(VGS,VTH,Kn,fet_mode='sat')
# verification for saturation
drain_voltage (VCC=VCC, ID=ID, RD=RD, VGS=VGS,VTH=VTH, fet_mode='sat')

# assumption 2: triode
# Solving for VDS
a = Kn/2
b = -1*(1/RD+Kn*(VGS-VTH))
c = VCC/RD
Vd1, Vd2 = quadratic_formula(a=a,b=b,c=c)
VDS = min(abs(Vd1),abs(Vd2))
print('The valid VDS here is: {} V'.format(VDS))
# Solving for ID
ID = current_post_quad(VCC=VCC, VDS=VDS, RD=RD)
# verification for triode
VDS = drain_voltage(VDS=VDS, VGS=VGS, VTH=VTH, fet_mode="triode")
```

```
The saturation current is: 5.609655 mA
VDS = -13.829 V, the assumption is False.
The solution are (0.21051089379563057+0j) and (5.000366299186825+0j)
The valid VDS here is: 0.21051089379563057 V
ID = 0.0009298297020681231 A
VDS = 0.21051089379563057 V, the assumption is True.
```

[55]:

```
# input parameter
VCC = 3
RD=3000
W = 3.5
L = 0.35
K_prime = 190*10**(-6)
Kn = W/L*K_prime
VGS = 3
VTH = 0.57

# assumption 1: saturation
ID = saturation_current(VGS,VTH,Kn,fet_mode='sat')
# verification for saturation
drain_voltage (VCC=VCC, ID=ID, RD=RD, VGS=VGS,VTH=VTH, fet_mode='sat')
```

```
# assumption 2: triode
# Solving for VDS
a = Kn/2
b = -1*(1/RD+Kn*(VGS-VTH))
c = VCC/RD
Vd1, Vd2 = quadratic_formula(a=a,b=b,c=c)
VDS = min(abs(Vd1),abs(Vd2))
print('The valid VDS here is: {} V'.format(VDS))
# Solving for ID
ID = current_post_quad(VCC=VCC, VDS=VDS, RD=RD)
# verification for triode
VDS = drain_voltage(VDS=VDS, VGS=VGS, VTH=VTH, fet_mode="triode")
```

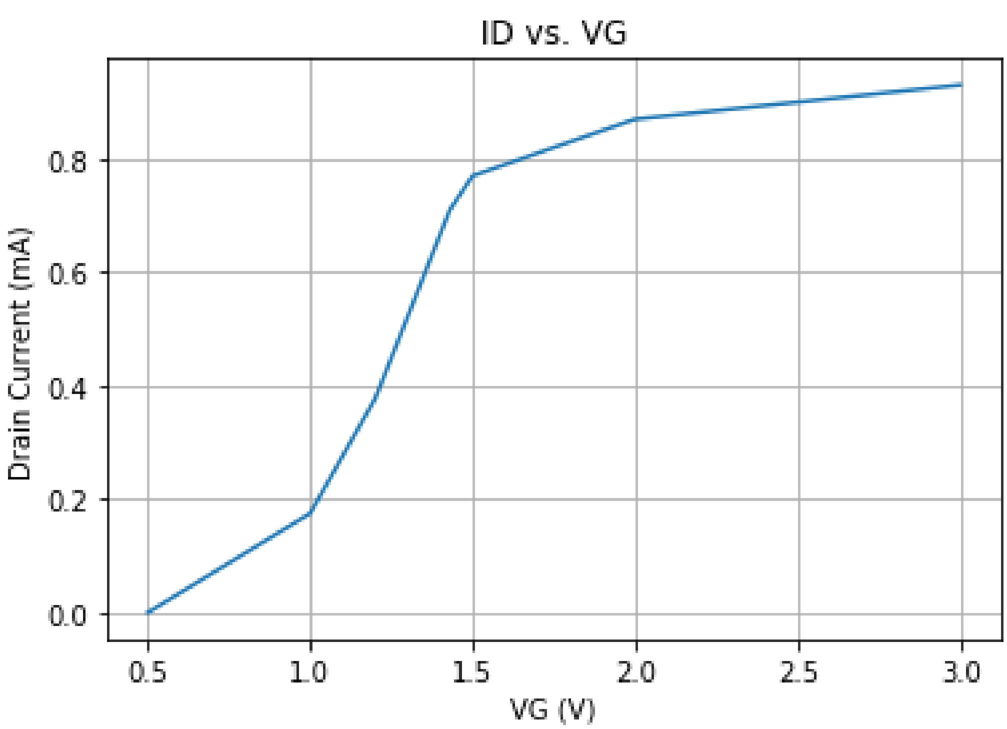
```
The saturation current is: 5.609655 mA
VDS = -13.829 V, the assumption is False.
The solution are (0.21051089379563057+0j) and (5.000366299186825+0j)
The valid VDS here is: 0.21051089379563057 V
ID = 0.0009298297020681231 A
VDS = 0.21051089379563057 V, the assumption is True.
```

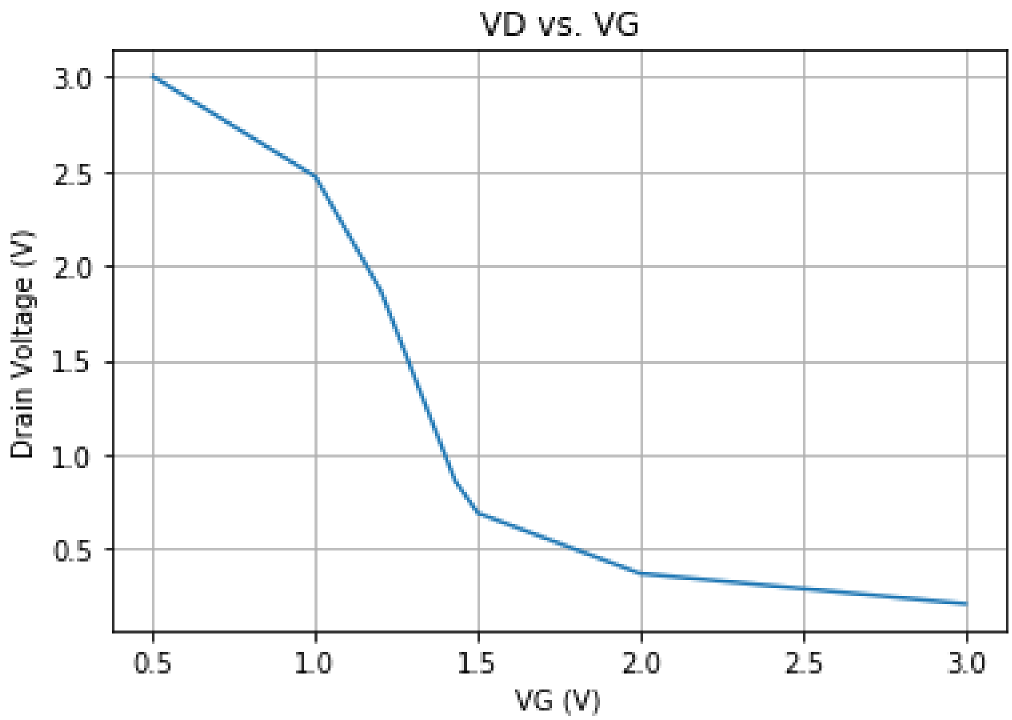
[56]:
```
# Problem: what is the drain voltage when MOSFET transition from SAT -> TRI?
# AKA: What VDS would be to make that happen?
# unknown: VGS, VDS, ID
# known: VDS = VGS + VTH, device in saturation.

# input parameter
VCC = 3
RD=3000
W = 3.5
L = 0.35
K_prime = 190*10**(-6)
Kn = W/L*K_prime
VTH = 0.57

a = Kn/2
b = 1/RD
c = -VCC/RD
Vd1, Vd2 = quadratic_formula(a=a,b=b,c=c)
VDS = abs(Vd2)
print("VDS would be about: {} V".format(Vd2)) #the positive digit one
ID = (VCC - VDS)/RD
print("ID would be about: {} A".format(ID))
VGS = VDS + VTH
print("VGS would be about: {} V".format(VGS))
```

```
The solution are (-1.216308559592374+0j) and (0.8654313666099176+0j)
VDS would be about: (0.8654313666099176+0j) V
```

```
ID would be about: 0.0007115228777966942 A
VGS would be about: 1.4354313666099174 V
```

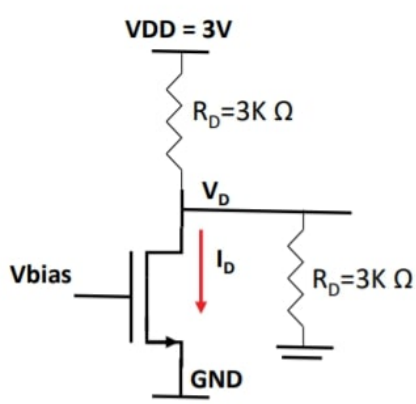
[48]:
```
import matplotlib
import matplotlib.pyplot as plt
import numpy as np

# Data for plotting
def plot_point ( t, data, xlabel, ylabel, title):
    fig, ax = plt.subplots()
    ax.plot(t, data)
    ax.set(xlabel=xlabel, ylabel=ylabel,
           title=title)
    ax.grid()
    plt.show()

VGS = [0.5, 1, 1.2, 1.43, 1.5, 2, 3]
ID = [0,.175,.377,0.71,.77,.87,.93]
VD = [3,2.47,1.87,.86,.69,.37,.21]

# ID vs. VGS
plot_point(t=VGS, data=ID, xlabel = 'VG (V)', ylabel = 'Drain Current (mA)',␣
 ↪title = 'ID vs. VG')
# VD vs. VGS
plot_point(t=VGS, data=VD, xlabel = 'VG (V)', ylabel = 'Drain Voltage (V)',␣
 ↪title = 'VD vs. VG')
```





1.0.2 Problem 2

Q2: Repeat Q1 for the circuit shown below.



V_{bias}	MOSFET oper. Mode Off/SAT/Linear	I_D(mA)	V_D(V)
0.5V			
1.0V			
1.2V			
1.5V			
2V			
3V			

V_{bias}	MOSFET Operation Mode	I_D (mA)	V_D (V)
0.5	OFF	0	1.5
1.0	SAT	.175	1.23
1.2	SAT	.377	.93
1.5	LINEAR	.650	.514
2.0	LINEAR	.783	.325
3.0	LINEAR	.869	.196

[58]:
```
import numpy as np

# Saturation
def drain_voltage (VCC=None, ID=None, RD1=None, RD2=None, VGS=None, VTH=None,␣
↪VDS=None, fet_mode=''):
    if fet_mode=="sat":
        VDS = parallel_resistance(R1=RD1, R2=RD2)*(VCC/RD1-ID)
        mode_true = VDS > (VGS-VTH)
        print("VDS = {} V, the assumption is {}.".format(VDS, mode_true))
        return VDS
    elif fet_mode=='triode':
        mode_true = VDS < (VGS-VTH)
        print("VDS = {} V, the assumption is {}.".format(VDS, mode_true))
        return VDS
    return None
def parallel_resistance (R1, R2):
    return R1*R2/(R1+R2)

# Triode
def quadratic_formula (a,b,c):
    d = (b**2) - (4*a*c)
    sol1 = (-b-cmath.sqrt(d))/(2*a)
    sol2 = (-b+cmath.sqrt(d))/(2*a)
    print('The solution are {0} and {1}'.format(sol1,sol2))
    return [sol1, sol2]
def current_post_quad (VCC, VDS, RD1,RD2):
    ID = VCC/RD1-VDS*(parallel_resistance(R1=RD1,R2=RD2)**(-1))
    print('ID = {} A'.format(ID))
    return ID
```

[42]:
```
W = 3.5
L = 0.35
K_prime = 190*10**(-6)
Kn = W/L*K_prime
VGS = 1
VTH = 0.57
ID = saturation_current(VGS,VTH,Kn,fet_mode='sat')
VCC = 3
```

```
RD=3000
#verification
drain_voltage (VCC=VCC, ID=ID, RD1=RD, RD2=RD,\
                VGS=VGS,VTH=VTH, fet_mode='sat')
```

```
The saturation current is: 0.175655 mA
VDS = 1.2365175 V, the assumption is True.
```

[42]: 1.2365175

[43]:
```
W = 3.5
L = 0.35
K_prime = 190*10**(-6)
Kn = W/L*K_prime
VGS = 1.2
VTH = 0.57
ID = saturation_current(VGS,VTH,Kn,fet_mode='sat')
VCC = 3
RD=3000
#verification
drain_voltage (VCC=VCC, ID=ID, RD1=RD, RD2=RD,\
                VGS=VGS,VTH=VTH, fet_mode='sat')
```

```
The saturation current is: 0.377055 mA
VDS = 0.9344174999999999 V, the assumption is True.
```

[43]: 0.9344174999999999

[38]:
```
W = 3.5
L = 0.35
K_prime = 190*10**(-6)
Kn = W/L*K_prime
VGS = 1.5
VTH = 0.57
ID = saturation_current(VGS,VTH,Kn,fet_mode='sat')
VCC = 3
RD1=3000
RD2=3000
#verification
drain_voltage (VCC=VCC, ID=ID, RD1=RD, RD2=RD,\
                VGS=VGS,VTH=VTH, fet_mode='sat')

# verification fail -> so it's triode
# VDS
a = Kn/2
b = -1*(Kn*(VGS-VTH)+parallel_resistance(R1=RD1, R2=RD2)**(-1))
c = VCC/RD1
```

```
Vd1, Vd2 = quadratic_formula(a=a,b=b,c=c)
VDS = min(abs(Vd1),abs(Vd2))
print('The valid VDS here is: {} V'.format(VDS))
# ID
ID = current_post_quad(VCC=VCC, VDS=VDS, RD1=RD1,RD2=RD2)
```

```
The saturation current is: 0.821655 mA
VDS = 0.2675175 V, the assumption is False.
The solution are (0.5140559592159103+0j) and (2.047698426749002+0j)
The valid VDS here is: 0.5140559592159103 V
ID = 0.0006572960271893932 A
```

[39]:

```
W = 3.5
L = 0.35
K_prime = 190*10**(-6)
Kn = W/L*K_prime
VGS = 2
VTH = 0.57
ID = saturation_current(VGS,VTH,Kn,fet_mode='sat')
VCC = 3
RD1=3000
RD2=3000
#verification
drain_voltage (VCC=VCC, ID=ID, RD1=RD, RD2=RD,\
               VGS=VGS,VTH=VTH, fet_mode='sat')

# verification fail -> so it's triode
# VDS
a = Kn/2
b = -1*(Kn*(VGS-VTH)+parallel_resistance(R1=RD1, R2=RD2)**(-1))
c = VCC/RD1
Vd1, Vd2 = quadratic_formula(a=a,b=b,c=c)
VDS = min(abs(Vd1),abs(Vd2))
print('The valid VDS here is: {} V'.format(VDS))
# ID
ID = current_post_quad(VCC=VCC, VDS=VDS, RD1=RD1,RD2=RD2)
```

```
The saturation current is: 1.9426550000000005 mA
VDS = -1.4139825000000006 V, the assumption is False.
The solution are (0.3252357544620683+0j) and (3.236518631502844+0j)
The valid VDS here is: 0.3252357544620683 V
ID = 0.0007831761636919545 A
```

[40]:

```
W = 3.5
L = 0.35
K_prime = 190*10**(-6)
Kn = W/L*K_prime
```

```python
VGS = 3
VTH = 0.57
ID = saturation_current(VGS,VTH,Kn,fet_mode='sat')
VCC = 3
RD1=3000
RD2=3000
#verification
drain_voltage (VCC=VCC, ID=ID, RD1=RD, RD2=RD,\
               VGS=VGS,VTH=VTH, fet_mode='sat')

# verification fail -> so it's triode
# VDS
a = Kn/2
b = -1*(Kn*(VGS-VTH)+parallel_resistance(R1=RD1, R2=RD2)**(-1))
c = VCC/RD1
Vd1, Vd2 = quadratic_formula(a=a,b=b,c=c)
VDS = min(abs(Vd1),abs(Vd2))
print('The valid VDS here is: {} V'.format(VDS))
# ID
ID = current_post_quad(VCC=VCC, VDS=VDS, RD1=RD1,RD2=RD2)
```

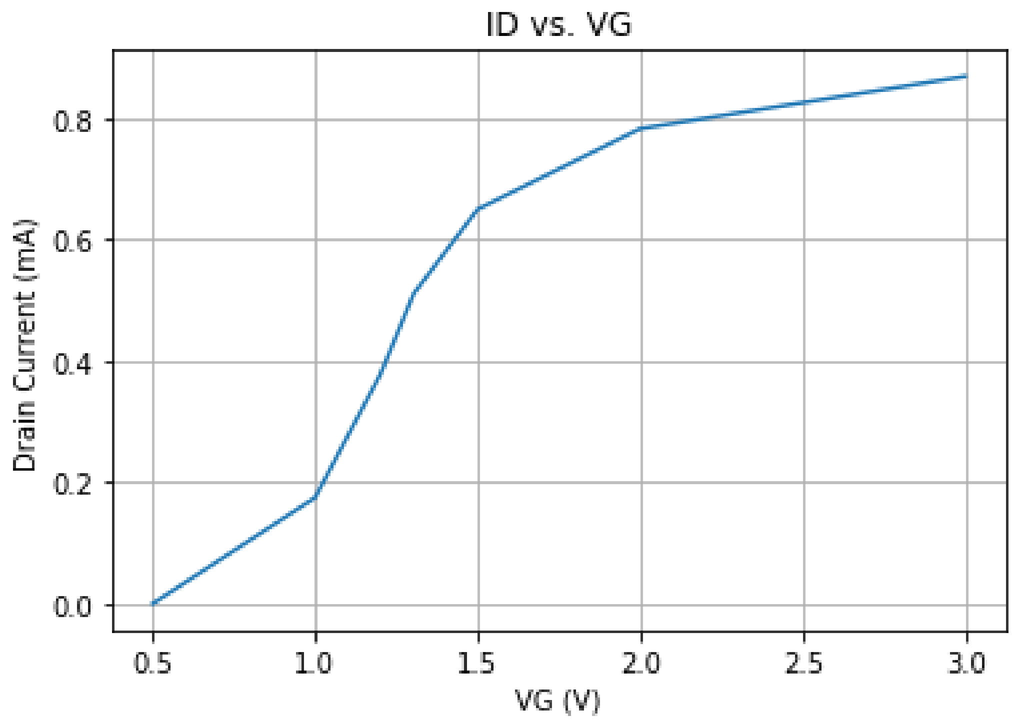
```
The saturation current is: 5.609655 mA
VDS = -6.9144825 V, the assumption is False.
The solution are (0.19618255270043303+0j) and (5.365571833264481+0j)
The valid VDS here is: 0.19618255270043303 V
ID = 0.0008692116315330447 A
```

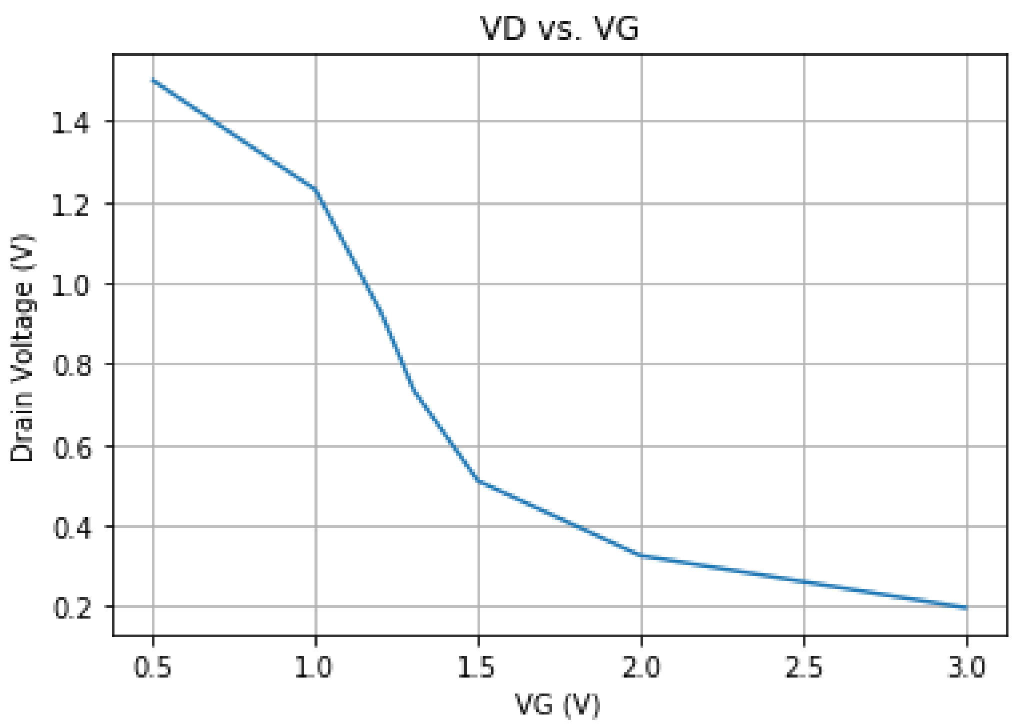
[62]:
```python
# Problem: what is the drain voltage when MOSFET transition from SAT -> TRI?
# AKA: What VDS would be to make that happen?
# unknown: VGS, VDS, ID
# known: VDS = VGS + VTH, device in saturation.

# input parameter
VCC = 3
RD1=3000
RD2=3000
W = 3.5
L = 0.35
K_prime = 190*10**(-6)
Kn = W/L*K_prime
VTH = 0.57
print("Find the point where device transition from saturation to linear␣
 ↪region\n")
a = Kn/2
b = parallel_resistance(R1=RD1, R2=RD2)**(-1)
c = -VCC/RD1
print("This is the 2 possible VD value")
```

```
Vd1, Vd2 = quadratic_formula(a=a,b=b,c=c)

#answer
VDS = abs(Vd2)
print("This is the ID value from the valid VDS value")
ID = current_post_quad(VCC=VCC, VDS=VDS, RD1=RD1,RD2=RD2)
VGS = VDS + VTH

#output
print("VDS would be about: {} V".format(Vd2)) #the positive digit one
print("ID would be about: {} A".format(ID))
print("VGS would be about: {} V".format(VGS))
```

```
Find the point where device transition from saturation to linear region

This is the 2 possible VD value
The solution are (-1.4351955887973646+0j) and (0.7334412028324524+0j)
This is the ID value from the valid VDS value
ID = 0.0005110391981116984 A
VDS would be about: (0.7334412028324524+0j) V
ID would be about: 0.0005110391981116984 A
VGS would be about: 1.3034412028324525 V
```

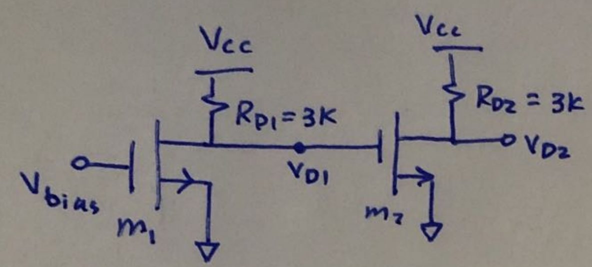
[47]:
```
VGS = [0.5, 1, 1.2, 1.303, 1.5, 2, 3]
ID = [0,.175,.377,0.511,.65,.783,.869]
VD = [1.5,1.23,0.93,.733,.51,.325,.196]

# ID vs. VGS
plot_point(t=VGS, data=ID, xlabel = 'VG (V)', ylabel = 'Drain Current (mA)',␣
 ↪title = 'ID vs. VG')
# VD vs. VGS
plot_point(t=VGS, data=VD, xlabel = 'VG (V)', ylabel = 'Drain Voltage (V)',␣
 ↪title = 'VD vs. VG')
```

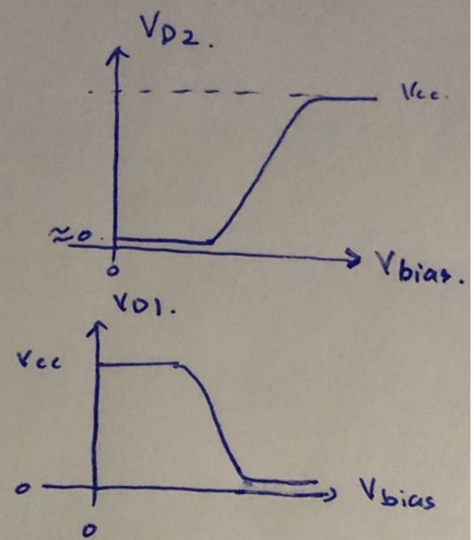

ID vs. VG
Drain Current (mA)
0.0
0.2
0.4
0.6
0.8
0.5
1.0
1.5
2.0
2.5
3.0
VG (V)


VD vs. VG
Drain Voltage (V)
0.2
0.4
0.6
0.8
1.0
1.2
1.4
0.5
1.0
1.5
2.0
2.5
3.0
VG (V)

1.0.3 Problem 3

Q3: w/out math. Do this.





If m_1 input high, V_{D1} Low.

If m_2 input low, V_{D1} High.

∴ If V_{G2} input high, V_{D2} Low

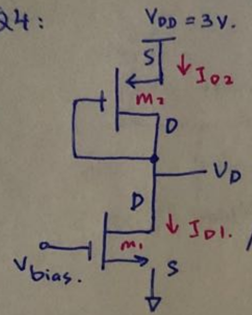
If V_{G2} input low, V_{D2} High

→ If m_2 input high, V_{D2} High.

If m_2 input low, V_{D2} Low.

1.0.4 Problem 4, part 1

Q4:



$\mu_p C_{ox} = 55 \mu A/V^2$

$V_{tp} = -.7V$

$W/L = 10.5/0.35$.

$K_p = 0.0017$

$\mu_n C_{ox} = 190 \mu A/V^2$

$V_{tn} = .57 V$.

$W/L = 3.5/.35$

$K_N = 0.0019$.

$V_{bias} = 1 V$.

$V_{GS} = 1V > V_{tn} \rightarrow$ ON.

Assume saturation.

$I_{D1} = \frac{1}{2} k_N (\underbrace{V_{GS}}_{\text{From } m_1} - V_{tn})^2$

$I_{D2} = \frac{1}{2} k_p (\underbrace{V_{SG}}_{\text{from } m_2} - |V_{tp}|)^2$

$\hookrightarrow V_{SG} = V_{cc} - V_D$.

$I_{D2} = \frac{1}{2} k_p (V_{cc} - V_D - |V_{tp}|)^2$

$I_{D1} = I_{D2}$

$\cancel{\frac{1}{2}} k_N (V_{GS} - V_{tn})^2 = \cancel{\frac{1}{2}} k_p (V_{cc} - V_D - |V_{tp}|)^2$

$\sqrt{\frac{K_n}{K_p}} (V_{GS} - V_{tn}) = V_{cc} - V_D - |V_{tp}|$

$V_D = V_{cc} - |V_{tp}| - \sqrt{\frac{k_n}{K_p}} (V_{GS} - V_{tn})$

$V_D = 1.84\ V$.

$V_D > V_{gs} - V_{th} \rightarrow$ True.

1.0.5 Problem 4 , part 2

$$I_{D2} = \frac{1}{2} K_p (V_{SG} - |V_{Tp}|)^2 \rightarrow m_2 \text{ sat.}$$

$$I_{D1} = \frac{1}{2} K_n (V_{gs} - V_{tn})^2 \rightarrow m_1 \text{ sat.}$$

$$\cancel{\tfrac{1}{2}} K_p (V_{sg} - |V_{Tp}|)^2 = \cancel{\tfrac{1}{2}} K_n . (V_{gs} - V_{tn})^2$$

$$\underbrace{V_{sg}}_{m_2} = |V_{Tp}| + \sqrt{\frac{K_n}{K_p}} (\underbrace{V_{gs}}_{m_1} - V_{tn})$$

$$= \boxed{1.15}$$

$$V_{SD} > (V_{SG} - |V_{Tp}|)$$

$$V_{cc} - V_D > V_{SG} - |V_{Tp}|$$

$$V_D < V_{cc} - \boxed{V_{SG}} + |V_{Tp}|$$

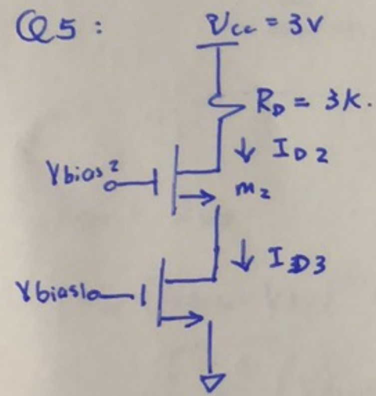
$$V_D < 3 - 1.15 + |V_{Tp}|$$

$$\boxed{V_D < 2.2V}$$ if $m_2 \rightarrow$ sat

$m_1 \rightarrow$ sat.

1.0.6 Problem 5

Q5:



we know	we don't know
$V_{bias1} = 1$	V_{bias2}.

To have m_2 in saturation.

We need $V_{GS} > V_{TH} = .57V$

To have m_2 stay at saturation

we need $V_{GS} < V_{DS} + V_{TH} = 1.43\,V$

$0.57V < V_{GS} = V_{bias2} < 1.43$