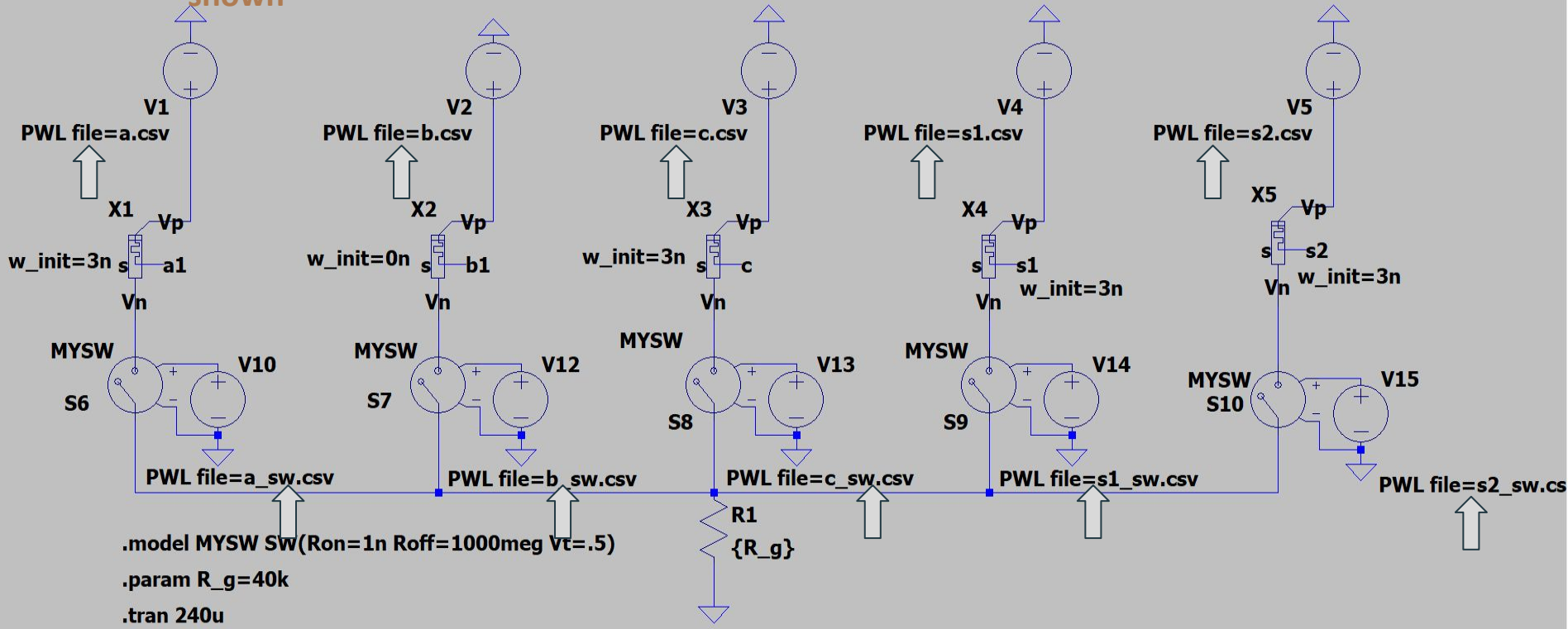
Exercise and Homework 5

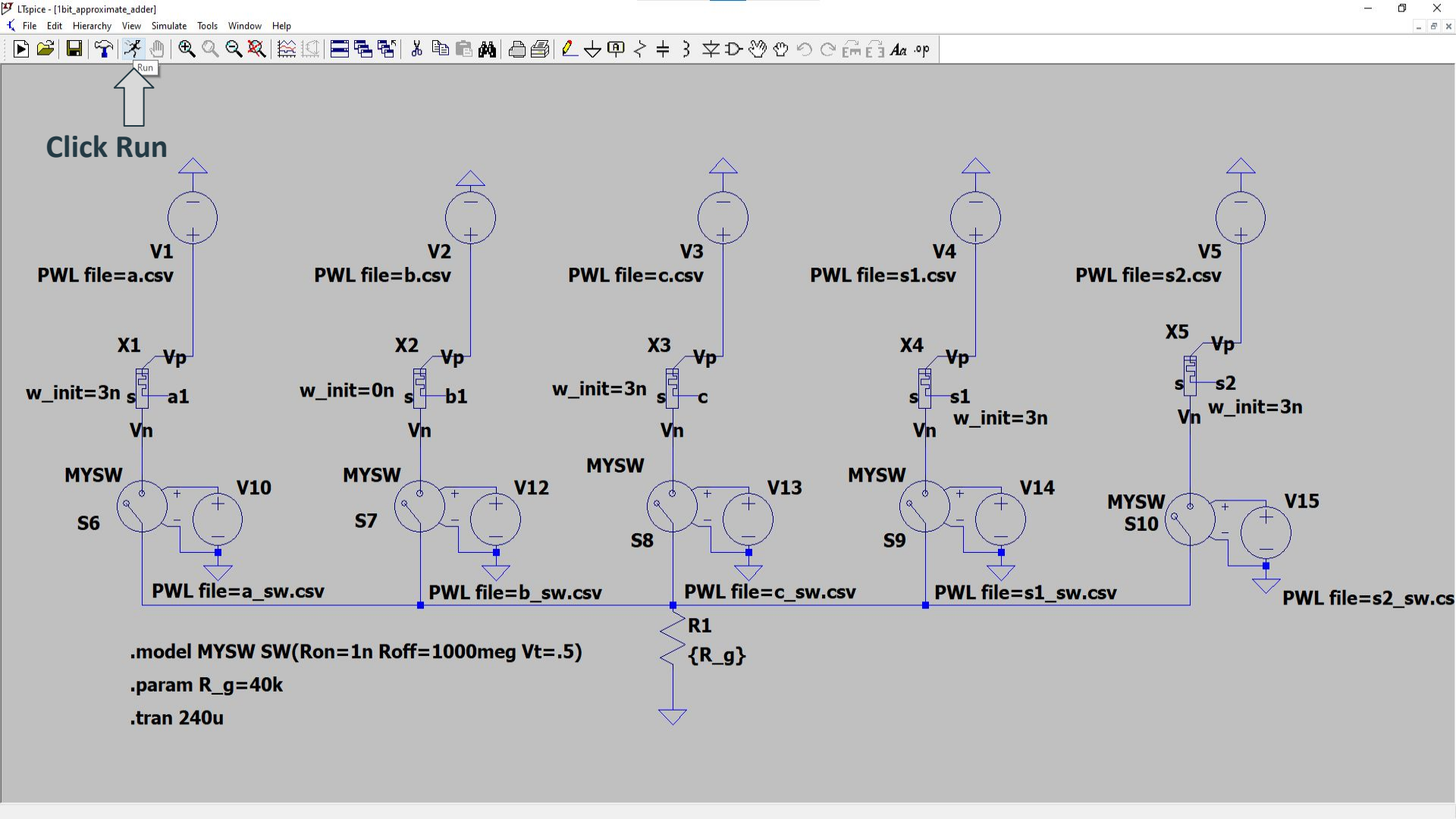
Exercise 5
1-bit Approximate Adder

LTspice - [1bit_approximate_adder]

File Edit Hierarchy View Simulate Tools Window Help

Everything is same as the previous 1-bit exact adder, just added different PWL files as shown




LTspice - [1bit_approximate_adder]
File Edit Hierarchy View Simulate Tools Window Help
Run
Click Run
V1
PWL file=a.csv
V2
PWL file=b.csv
V3
PWL file=c.csv
V4
PWL file=s1.csv
V5
PWL file=s2.csv
X1
Vp
w_init=3n
s
a1
Vn
X2
Vp
w_init=0n
s
b1
Vn
X3
Vp
w_init=3n
s
c
Vn
X4
Vp
s
s1
w_init=3n
Vn
X5
Vp
s
s2
w_init=3n
Vn
MYSW
S6
V10
PWL file=a_sw.csv
MYSW
S7
V12
PWL file=b_sw.csv
MYSW
S8
V13
PWL file=c_sw.csv
MYSW
S9
V14
PWL file=s1_sw.csv
MYSW
S10
V15
PWL file=s2_sw.cs
R1
{R_g}
.model MYSW SW(Ron=1n Roff=1000meg Vt=.5)
.param R_g=40k
.tran 240u

LTspice - 1bit_approximate_adder

File Edit Hierarchy View Simulate Tools Window Help

1bit_approximate_adder 1bit_approximate_adder

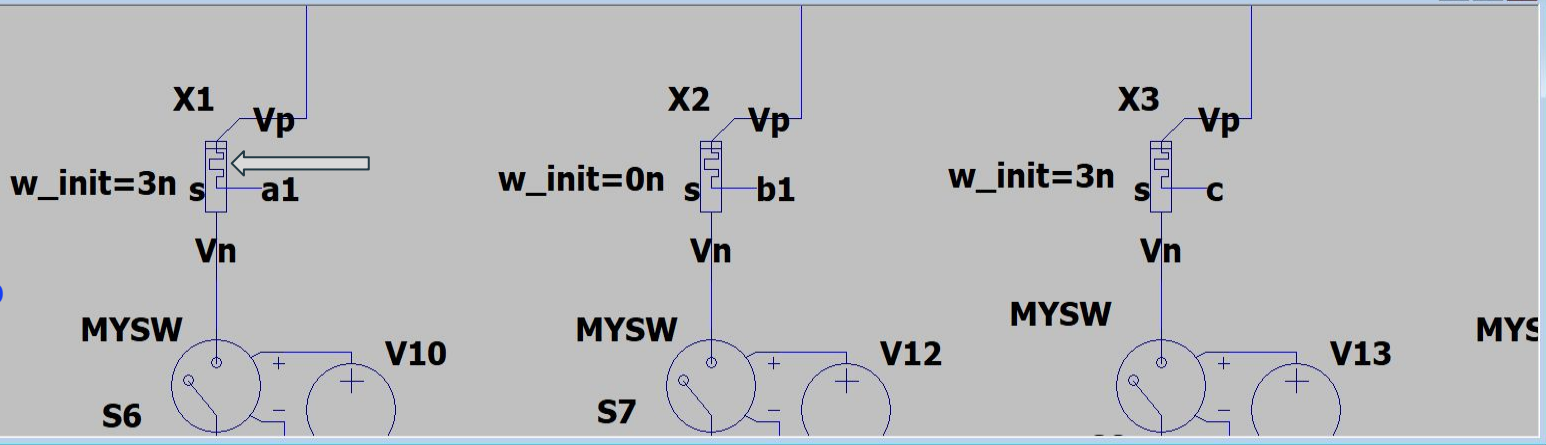
1bit_approximate_adder

V(a1)*Ix(X1:s)+V(N002)*Ix(X1:Vp)+V(N006)*Ix(X1:Vn)

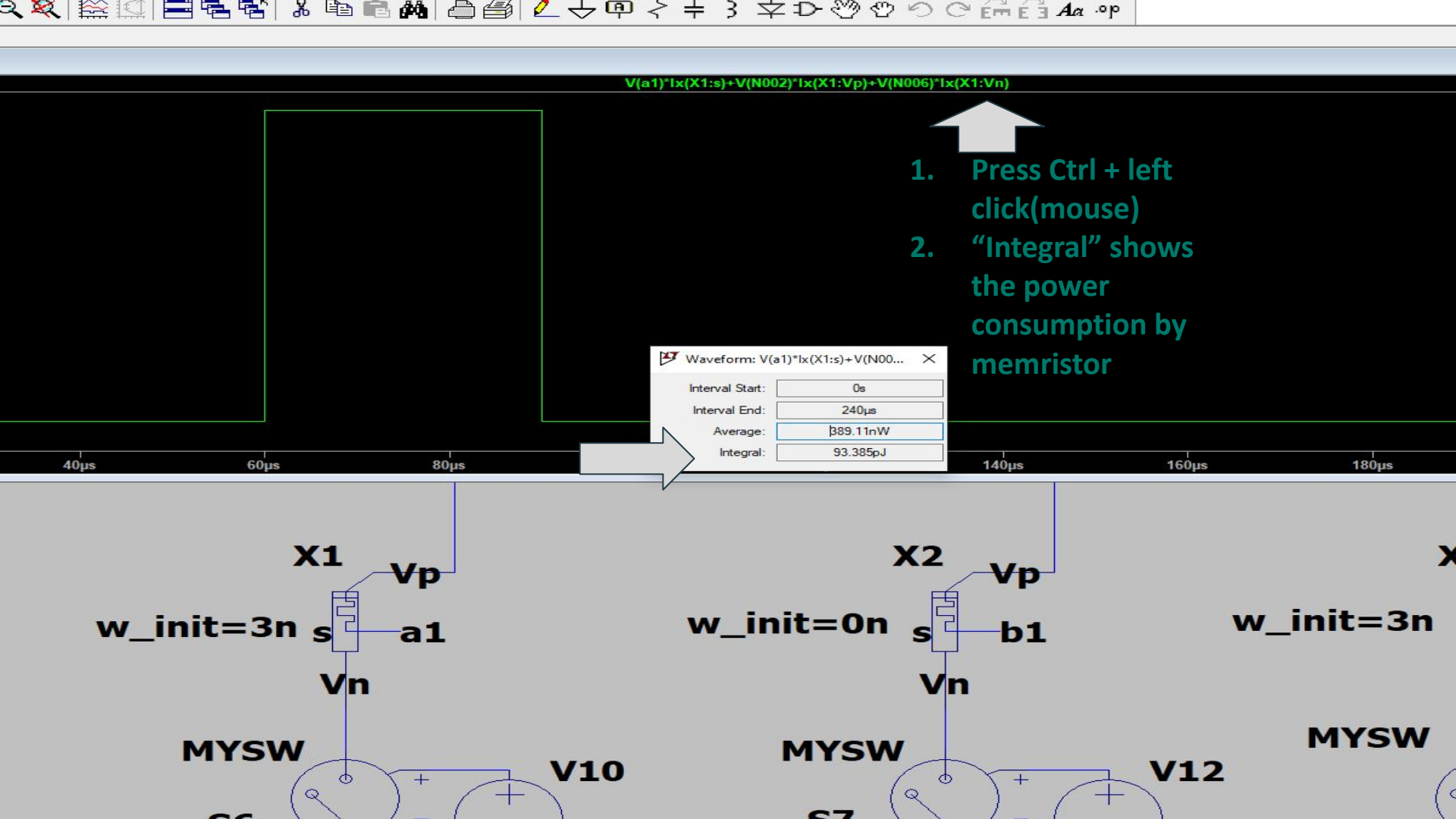
3.3µW
3.0µW
2.7µW
2.4µW
2.1µW
1.8µW
1.5µW
1.2µW
0.9µW
0.6µW
0.3µW
0.0µW

1bit_approximate_adder

1. Place your cursor on a1 meristor
2. Press alt + left click(mouse) to get the above graph



Click to plot V(a1). DC operating point: V(a1) = 1V


V(a1)*Ix(X1:s)+V(N002)*Ix(X1:Vp)+V(N006)*Ix(X1:Vn)
1. Press Ctrl + left click(mouse)
2. “Integral” shows the power consumption by memristor
Waveform: V(a1)*Ix(X1:s)+V(N00...
Interval Start: 0s
Interval End: 240µs
Average: 389.11nW
Integral: 93.385pJ
40µs
60µs
80µs
140µs
160µs
180µs
X1
Vp
w_init=3n
s
a1
Vn
MYSW
V10
X2
Vp
w_init=0n
s
b1
Vn
MYSW
V12
w_init=3n
MYSW

Energy consumption of 1 - bit exact adder of different memristors for different values of a, b, c

Energy consumption in memristors

a	b	c	a1 (pJ)	b1 (pJ)	c (pJ)	s1 (pJ)	s2 (pJ)
0	0	0	368.4	97.664	292.61	575.31	820.33
0	0	1	460.36	83.098	589.28	334.38	577.2
0	1	0	225.2	478.48	135.11	520.98	581.88
0	1	1	154.63	404.52	625.63	381.94	314.33
1	0	0	421.41	375.3	157.12	439.51	676.16
1	0	1	353.78	311.2	662.34	298.78	349.99
1	1	0	489.23	481.12	437.62	380.49	178.42
1	1	1	583.5	409.41	618.22	140.01	141.03
							1.5925nJ

We put the total energy consumption for different bit values in the function “ExactAdder”

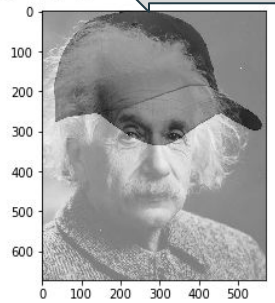
```
def ExactAdder(a, b, c):
    if a==0 and b==0 and c==0:
        s=0
        c_out=0
        energy_consumption = 2153
    elif a==0 and b==0 and c==1:
        s=1
        c_out=0
        energy_consumption = 2043
    elif a==0 and b==1 and c==0:
        s=1
        c_out=0
        energy_consumption = 1941
    elif a==0 and b==1 and c==1:
        s=0
        c_out=1
        energy_consumption = 1881
    elif a==1 and b==0 and c==0:
        s=1
        c_out=0
        energy_consumption = 2069
    elif a==1 and b==0 and c==1:
        s=0
        c_out=1
        energy_consumption = 1976
    elif a==1 and b==1 and c==0:
        s=0
        c_out=1
        energy_consumption = 1966
    elif a==1 and b==1 and c==1:
        s=1
        c_out=1
        energy_consumption = 1892
    return s, c_out, energy_consumption #in pJ
```

Then we calculate the total Energy Consumption for adding the two images

```
add_approx,max_Nbit_adder, total_energy = MyAdder(Y_einstein,Y_cap)
plt.imshow(add_approx, cmap = "gray")
print(max_Nbit_adder.max())
print(Y_einstein.max())
print(Y_cap.max())
print()
print("The total energy consumed for this operation is: "+str(total_energy) + " pJ")
print("or "+str(round(total_energy/10**9,2))+" mJ")
```

```
8.0
255.0
255.0

The total energy consumed for this operation is: 5852819534 pJ
or 5.85 mJ
```



Homework 5

Regulations

Always compress your files in **“.zip”** and no other zipping format!!

Only **one** from the group should upload.

Add the group member names as the name of the zip file.

Make a group of 2 for this assignment.

Tasks

- Make a 3 - bit approximate adder for the first algorithm on LtSpice and upload the circuit in a zip file.
- We have provided different 1-bit algorithms which each have different steps and memristor architecture. Calculate the value of Cout, Sout and Energy consumption of the different 1-bit algorithms to calculate the PSNR and Similarity Index of the Einstein image.

(Note: Some energy consumptions have already been calculated and put in “merge[ABC].csv” files in the algorithms folders)