

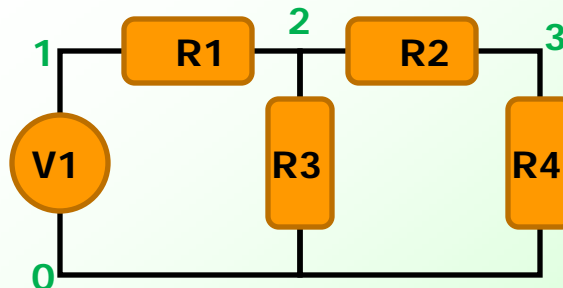
Quick SPICE Introduction

- SPICE = Simulation Program with Integrated Circuit Emphasis
- Netlist = text-based description of circuit
 - netlist does not use symbols or graphical elements
 - makes it simple to learn
 - but hard to visualize → usually need a companion schematic
 - netlist describes
 - circuit elements (resistors, capacitors, etc.)
 - power supplies, input voltages, bias currents, etc.
 - connections between circuit elements
 - analysis method; defines data to be calculated
- Example netlist

Example 1

```
V1 1 0 10      ← DC voltage supply
R1 1 2 10      ← Resistive elements
R2 2 0 30
R3 2 3 200
R4 3 0 4k
.PROBE        ← Analysis statements
.TRAN 1 200 0 1
.END
```

nodes, values



Op Amp SPICE Simulation

- Subcircuits: circuit block referenced as a single element

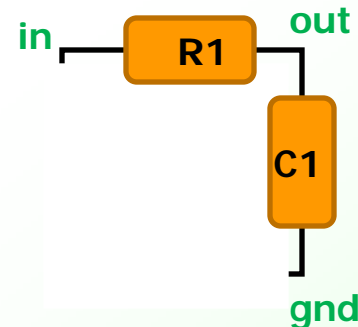
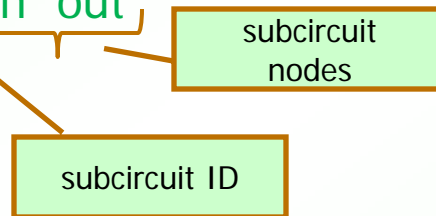
- element line description

- EX: `Xopamp1 vin vref vdd gnd vout opamp741`



- subcircuit definition

```
.SUBCKT lpfiler in out
R1 in out 1k
C1 out gnd 1u
.ENDS
```



In most SPICE simulators, can use numbers or text strings to define nodes

PSpice requires ground defined as node 0.

- Subcircuits for opamp simulation

- commercial opamps often provide a model subcircuit netlist

- include opamp model subcircuit in your SPICE file
 - “call” opamp model with single subcircuit element line

Example Inverting Amp Using Subcircuits

* **Main circuit**

XOP467 Vpos Vneg VDD GND Vout OP467

*above line calls the subcircuit; node order defined in subcircuit

R1 Vin Vneg 100

R2 Vneg Vout 100k

* define sine input (0V offset, 1mV amplitude, 1kHz frequency) at Vin

VS Vin agnd AC 1 SIN(0V 1mV 1KHZ)

* define positive and negative supply and analog ground

Vposupply VDD GND 10V

Vagnd agnd GND 5V

* connect Vneg op-amp input to agnd

V1 Vpos agnd 0V

Vg GND 0 0V

* OP467 SPICE Macro-model

```

*          non-inverting input
*          | inverting input
*          | | positive supply
*          | | negative supply
*          | | output
*          | |

```

```
.SUBCKT OP467 1 2 99 50 27
```

<long subcircuit omitted>

.ENDS

* define analysis type: .AC for frequency response, .TRAN for time response

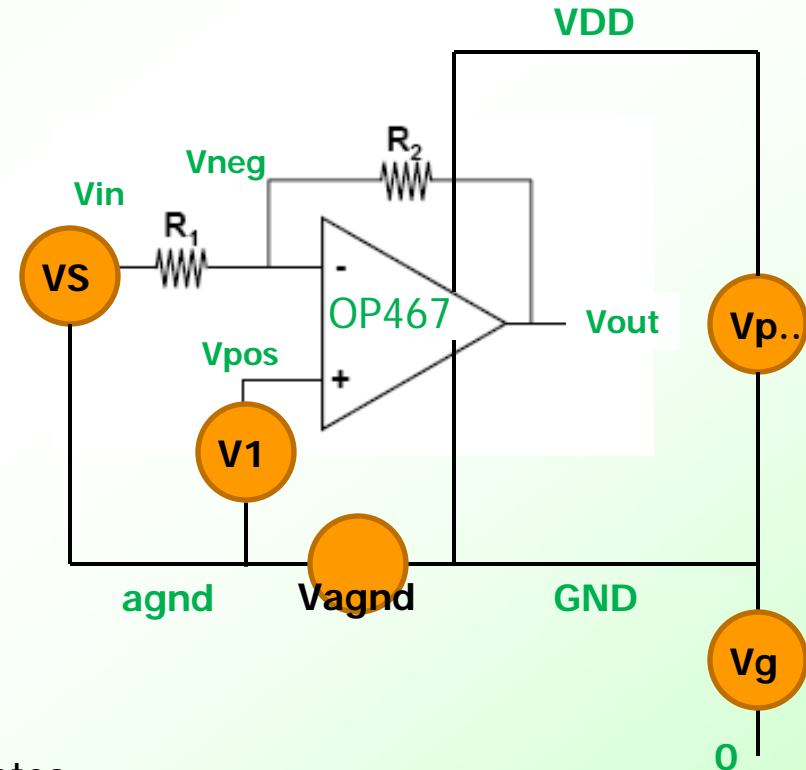
.AC DEC 5 10 1MEG

*.tran 3m

.PROBE

.END

* End Main circuit



Notes

* = comment line

first line of netlist is always circuit description

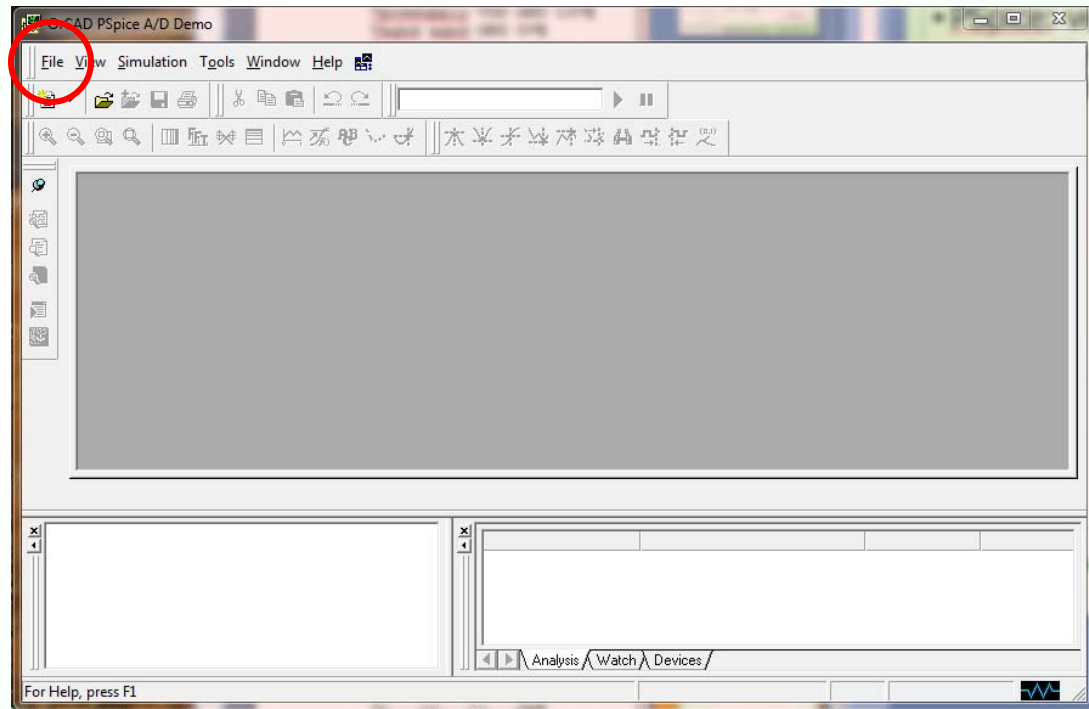
always end with .END

SPICE Analysis Types

- **.AC**: sweeps frequency for all defined AC sources
 - used for frequency response analysis, most common type for ECE445
 - must have at least one AC source in circuit
 - Example AC source: `Vs 1 0 AC 1`
 - source name, output node, reference node, AC (not DC), amplitude
 - Example **.AC** analysis statement
 - `.AC DEC 10 100 1e6`
 - DEC 10 = logarithmic range with 10pts per decade
 - 10 = starting frequency, 1e6 = stop frequency (1MHz)
- **.TRAN**: analyze circuit over time
 - generally for time-varying sources like sinusoidal signals
 - Example SIN source: `Vs 1 0 SIN(2V 10mV 1k)`
 - 2V = DC offset, 10mV = AC amplitude, 1k = frequency
 - Example **.TRAN** analysis statement
 - `.TRAN 20u 20m`
 - 20u = step time, 20m = stop time
- More details? <http://www.uta.edu/ee/hw/pspice/>

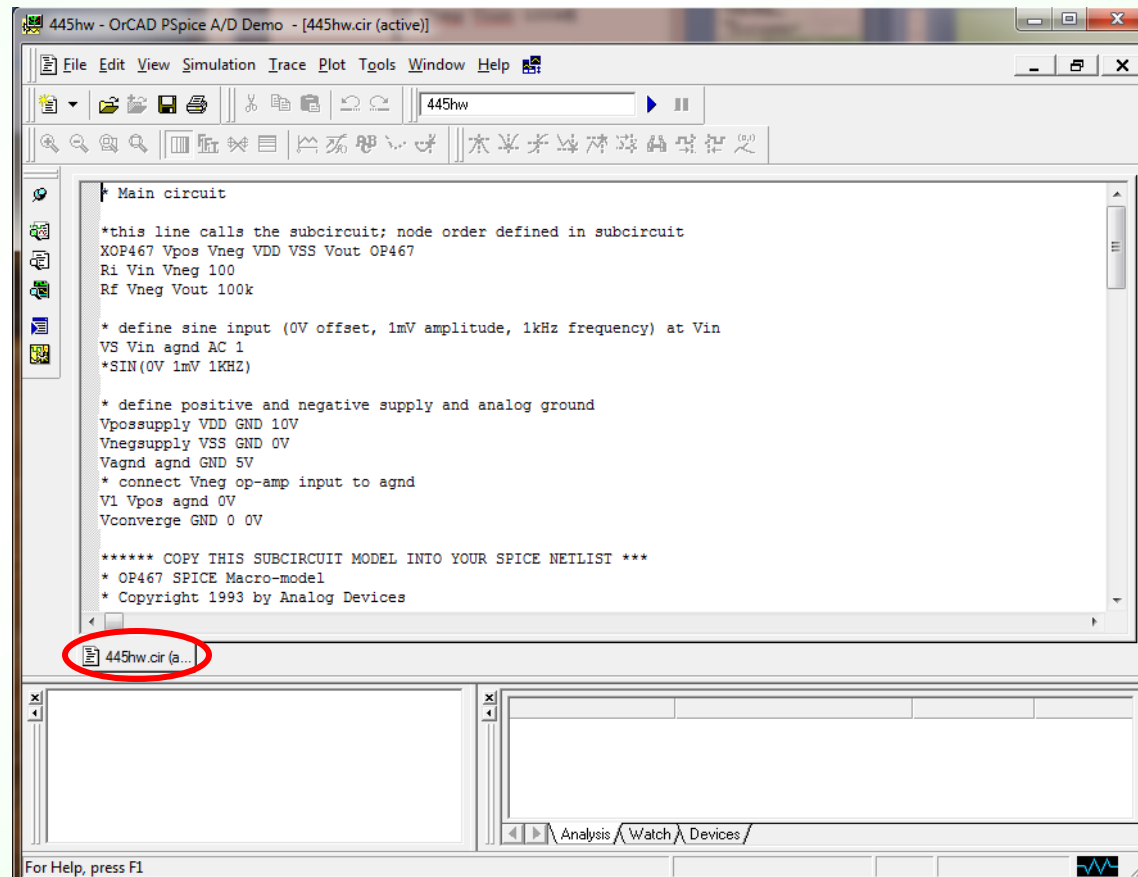
OrCAD PSpice

- PSpice: version of SPICE by Cadence
 - available in DECS labs
 - free student version available through the internet
- Create new file
 - File >> New >> Text File



OrCAD PSpice

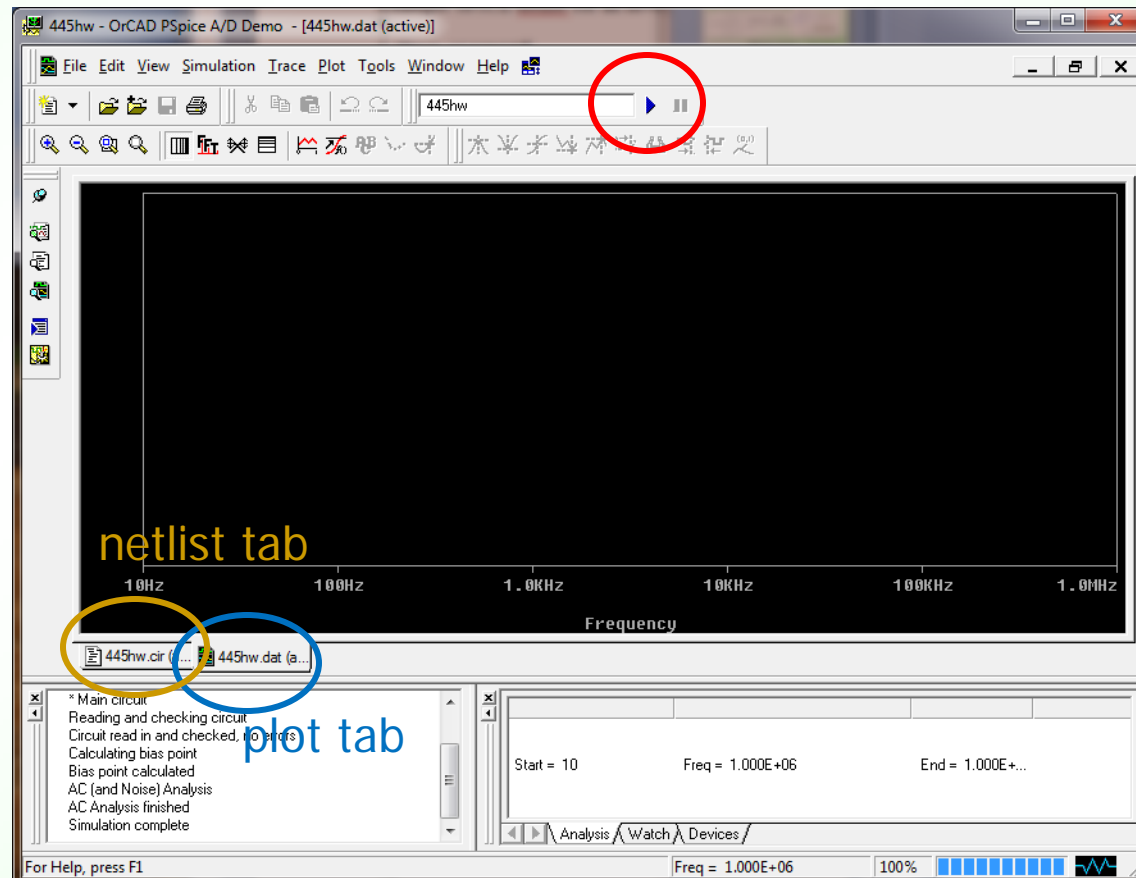
- PSpice: version of SPICE by Cadence
 - available in DECS labs
 - free student version available through the internet
- Create new file
 - File >> New >> Text File
- Type in netlist
 - save as .cir file



OrCAD PSpice

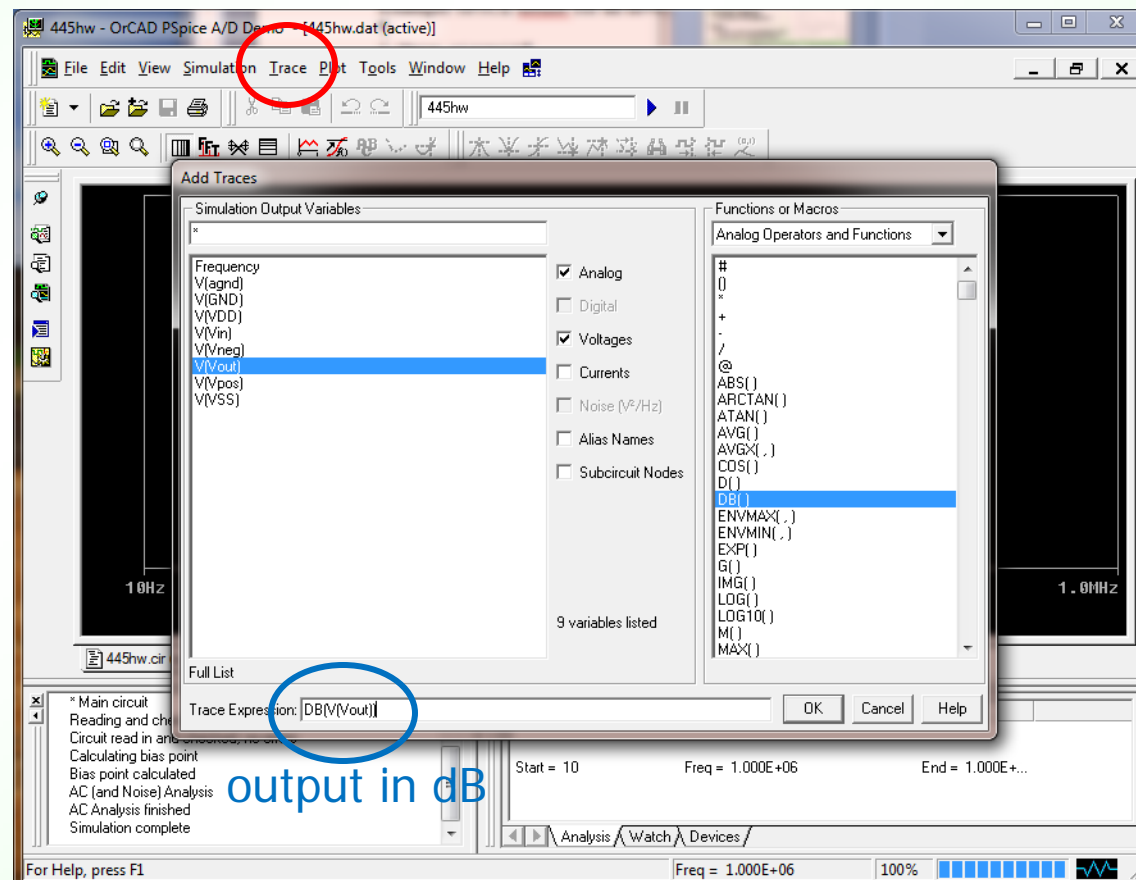
- PSpice: version of SPICE by Cadence
 - available in DECS labs
 - free student version available through the internet
- Create new file
 - File >> New >> Text File
- Type in netlist
 - save as .cir file
- Run simulation
 - type specified in netlist
 - check .out file if errors

```
* define analysis type: .AC for frequency response,
.Probe
.AC DEC 5 10 1MEG
*.TRAN 3m
.END
```



OrCAD PSpice

- PSpice: version of SPICE by Cadence
 - available in DECS labs
 - free student version available through the internet
- Create new file
 - File >> New >> Text File
- Type in netlist
 - save as .cir file
- Run simulation
 - type specified in netlist
 - check .out file if errors
- Add trace
 - Trace >> Add Trace
 - DB(V(Vout))



output in dB

OrCAD PSpice

- PSpice: version of SPICE by Cadence
 - available in DECS labs
 - free student version available through the internet
- Create new file
- Type in netlist
- Run simulation
- Add trace
 - Trace >> Add Trace
 - DB(V(Vout))
- Add cursor
 - Trace >> Cursor >> Display
 - -3dB frequency = 30.7kHz

• Questions?

