HOW SPICE WORKS

- SPICE ≡ Simulation Program for Integrated Circuit Emphasis

- Developed at UCBerkeley

- General purpose circuit simulator for nonlinear DC or transient analysis.

- De-facto circuit simulation tool for IC design. Several variants of the original (public-domain) SPICE program are sold by vendors.

- Built-in models for resistors, capacitors, inductors, mutual inductors, independent sources, dependent sources, transmission lines, BJTs, various FETs etc.

- User specifies the circuit to be simulated, and the type of simulation to be performed via model “cards” in a SPICE “deck”.

- SPICE translates this description and performs the simulation, providing the output in one of several user-specified ways.
• The steps that are followed in this process are outlined below.
  
  – Formulate circuit equations
  – Solve these equations (using algorithms for solving linear algebraic equations)
  – Handle non-linear circuits by linearizing them
  – Handle transient analysis by using Linear Multi-Step methods.

• We will deal with each of these topics in a separate sub-unit.

• This unit deals with formulating circuit equations.
FORMULATING CIRCUIT EQUATIONS

- Utilize equations obtained by applying
  - KVL (Kirchoff Voltage Law)
  - KCL (Kirchoff Current Law)
  - Branch Equations

- Circuit components are modeled mathematically as ideal elements (resistors, capacitors, inductors, current or voltage sources, etc)

- Mathematical equations are in terms of physical quantities like current, voltage, charge etc.

- Reference directions for these elements are as below:

  ![Two-terminal Circuit](image1.png)
  ![Two-port Circuit](image2.png)

- $i$, $v$ are branch currents and voltages respectively.
- **Ideal two-terminal elements are modeled as shown below:**

**Resistor**

Linear

Voltage Controlled

\[ i = \frac{1}{R}v \]

Current Controlled

\[ v = Ri \]

Non-linear

\[ i = i(v) \]

\[ v = v(i) \]

**Capacitor**

Linear

\[ q = Cv, \quad i = \frac{dq}{dt} \rightarrow i = C\frac{dv}{dt} \]

Voltage controlled

Non-linear

\[ q = q(v), \quad i = \frac{dq}{dt} \rightarrow i = \frac{\delta q}{\delta v} \frac{dv}{dt} = C(v)\frac{dv}{dt} \]

Gate-source and gate-drain capacitors in a MOS device are non-linear

**Diode**

\[ v = v(i) \]

is used in SPICE since

\[ i = i(v) \]

has a very high slope for \( v > V_{on} \)
- Controlled sources are treated as two-port elements.

**VCVS**

<table>
<thead>
<tr>
<th>Linear</th>
<th>Non-linear</th>
</tr>
</thead>
<tbody>
<tr>
<td>$v_k = E K v_c$</td>
<td>$v_k = v_k(v_c)$</td>
</tr>
<tr>
<td>$i_c = 0$</td>
<td>$i_c = 0$</td>
</tr>
</tbody>
</table>

**CCCS**

<table>
<thead>
<tr>
<th>Linear</th>
<th>Non-linear</th>
</tr>
</thead>
<tbody>
<tr>
<td>$i_k = F K i_c$</td>
<td>$i_k = i_k(i_c)$</td>
</tr>
<tr>
<td>$v_c = 0$</td>
<td>$v_c = 0$</td>
</tr>
</tbody>
</table>
TOPOLOGICAL EQUATIONS

- Write KCL for each node of the circuit (currents out of a node are positive, currents into a node are negative)

- Write KVL for each branch of the circuit.

- Examples below:

  ![Diagram]

  **KCL**

  \[ \begin{align*}
  &+ v_1 - \\
  \text{node 1}: & i_1 - i_2 - i_3 = 0 \\
  \text{node 2}: & \ldots \\
  \text{node 3}: & \ldots \\
  \text{branch 1}: & v_1 - e_1 + e_2 = 0
  \end{align*} \]

- **Good news!** KVL and KCL can be assembled directly from the input description
MATRIX FORM OF KVL AND KCL

- Actually, KVL and KCL are more convenient to write in matrix form

- Note that IS5 is an independent current source.

- KCL can be written in matrix form as $A \mathbf{i} = \mathbf{0}$, where the matrices $A$ and $i$ are given by:

\[
\begin{bmatrix}
1 & 1 & 1 & 0 & 0 \\
0 & 0 & -1 & 1 & -1
\end{bmatrix}
\begin{bmatrix}
i_1 \\
i_2 \\
i_3 \\
i_4 \\
i_5
\end{bmatrix}
= 
\begin{bmatrix}
0 \\
0
\end{bmatrix}
\]

- The first row represents KCL for node 1, and the second row represents KCL for node 2.
• We could have written KCL for node 0 (the datum node) but it is a linear combination (the sum with sign reversed, in particular) of the rows listed above. Try this out.

• The matrix $A$ is called the reduced incidence matrix.

• Likewise, KVL can be written in matrix form as $v - A^T e = 0$.

$$
\begin{bmatrix}
v_1 \\
v_2 \\
v_3 \\
v_4 \\
v_5
\end{bmatrix} -
\begin{bmatrix}
1 & 0 \\
1 & 0 \\
1 & -1 \\
0 & 1 \\
0 & -1
\end{bmatrix}
\begin{bmatrix}
e_1 \\
e_2
\end{bmatrix} =
\begin{bmatrix}
0 \\
0
\end{bmatrix}
$$

• Note that we could include the datum node in this equation as well. If we did this, however, the equations would be unaffected since $e_0 = 0$ by definition.

• Aside: Telegen’s theorem states that $\sum i_j v_j = 0$.

• In other words, the power dissipated in passive devices is generated from active devices.

• In our matrix notation, this would be $i^T v = 0$
NODE-BRANCH INCIDENCE MATRIX

- The matrix $A$ is called the node-branch incidence matrix or simply the incidence matrix of the circuit.

- Its columns correspond to branches, and rows correspond to nodes. If the datum row is missing, it is also called the reduced incidence matrix.

- It can be constructed by the following rule:

$$A_{ij} = \begin{cases} 
1 & \text{if node } i \text{ is the "+" terminal of branch } j \\
-1 & \text{if node } i \text{ is the "-" terminal of branch } j \\
0 & \text{if node } i \text{ is not connected to branch } j 
\end{cases}$$

- Properties of $A$:
  - $A$ is unimodular ($|a_{ij}| \leq 1$)
  - $A$ has at most 2 non-zero entries per column
  - $A$ has at least 1 non-zero entry per column
  - $A^T \cdot 1 = 0$, where $1$ is a column vector of "1" entries. Thus we designate one node as a datum or ground node, resulting in a reduced incidence matrix. The property listed in this bullet is not true for the reduced incidence matrix.
EQUATION ASSEMBLY FOR LINEAR CIRCUITS

• There are two common techniques to assemble equations:
  – Sparse Tableau Analysis (STA)
  – Nodal Analysis (NA)
  – Modified Nodal Analysis (MNA)

• NA is a precursor of MNA

• Assume that we have $n$ nodes and $b$ branches in our circuit. Given this, let us look at STA, NA and MNA.
SPARSE TABLEAU ANALYSIS

- STA was developed by Brayton, Gustavson and Hachtel.

- Write KCL: $A_i = 0$ ($n$ equations, one per node)

- Write KVL: $v - A^T e = 0$ ($b$ equations, one per branch)

- Write branch equations: $K_i i + K_v v = S$ ($b$ equations, one per branch). These equations reflect the relation between branch currents and voltages.

- So the STA Matrix looks like (dropping the underscore in the representation of matrix or vector quantities henceforth)

$$
\begin{bmatrix}
A & 0 & 0 \\
0 & I & -A^T \\
K_i & K_v & 0
\end{bmatrix} 
\begin{bmatrix}
i \\
v \\
e
\end{bmatrix} = 
\begin{bmatrix}
0 \\
0 \\
S
\end{bmatrix}
$$

- The components along a row have sizes $b$, $b$ and $n$, while the components along a column have sizes $n$, $b$ and $b$ respectively.

- The large matrix is referred to as the Sparse Tableau.
• Note that we have \( n + 2b \) equations in as many unknowns.

• Typically the tableau is very large and also very sparse.

• Advantages of STA:
  - STA can be applied to any circuit
  - STA equations can be assembled directly from input data using a technique called “stamping”. More on this later.
  - STA matrix is typically very sparse.
    * \( A \) or \( A^T \) can have at most \( 2b \) non-zero entries (recall \( A \) is unimodular)
    * \( I \) has \( b \) non-zero entries
    * \( K_i \) and \( K_v \) have \( b \) non-zero entries each.
    * So sparsity is \( \frac{7b}{(n+2b)^2} \approx 1/3 \) to \( 1/2 \) in practice.
    * But special solution techniques must utilized to exploit this sparsity. Specialized data structures and sophisticated programming techniques must be used.
NODAL ANALYSIS

- NA reduces the size of the final matrix, but sparsity is reduced.

- Developed by McCalla, Nagel, Rohrer, Ruehli, Ho.

- Let us understand it using a circuit we saw earlier.

![Diagram of a circuit with nodes and branches]

- Step 1: Write KCL ($n = 2$ equations)
  - $i_1 + i_2 + i_3 = 0$
  - $-i_3 + i_4 - i_5 = 0$

- Step 2: Use branch equations to eliminate branch currents from the result of step 1 (requires voltage controlled devices)
  - $\frac{1}{R_1}v_1 + G_2v_3 + \frac{1}{R_3}v_3 = 0$
\[-\frac{1}{R_3}v_3 + \frac{1}{R_4}v_4 = IS5\]

- **OK**, now we have 2 equations and 3 unknowns. But note that \(v_4\) is a function of \(v_1\) and \(v_3\). So to account for this, we now do step 3.

- **Step 3**: Use KVL to eliminate branch voltages from the result of step 2.
  \[-\frac{1}{R_1}e_1 + G_2(e_1 - e_2) + \frac{1}{R_3}(e_1 - e_2) = 0\]
  \[-\frac{1}{R_3}(e_1 - e_2) + \frac{1}{R_4}e_2 = IS5\]

- Now scrunched this into a matrix:
  \[
  \begin{bmatrix}
  \frac{1}{R_1} + G_2 + \frac{1}{R_3} & -G_2 - \frac{1}{R_3} \\
  -\frac{1}{R_3} & \frac{1}{R_3} + \frac{1}{R_4}
  \end{bmatrix}
  \begin{bmatrix}
  e_1 \\
  e_2
  \end{bmatrix}
  =
  \begin{bmatrix}
  0 \\
  IS5
  \end{bmatrix}
  \]

- This is of the form \(Y_n e = IS\). Note that we have dropped the underscores in our notation, and will henceforth infer matrices and vectors based on context.

- Note that \(Y_n\) is sparse, but significantly less so than the STA matrix. \(Y_n\) is significantly smaller than the STA matrix (which would have been \(12 \times 12\) for this example).
ASSEMBLING NODAL ANALYSIS MATRIX EFFICIENTLY

- Actually, $Y_n$ and $IS$ can be assembled efficiently from the circuit description. This process is referred to as “stamping”.

- Consider a resistor of value $RK$ connected between nodes $N+$ and $N−$ as shown below. Since $i = \frac{e_{n+} - e_{n−}}{RK}$ is positive in sign for the KCL at $N+$, and negative for the KCL at $N−$, we get the following entries in $Y_n$ due to the resistor:

\[
\begin{bmatrix}
N+ & \cdots & N− \\
\frac{1}{RK} & \frac{1}{RK} & \frac{1}{RK} \\
\vdots & \ddots & \vdots \\
\frac{1}{RK} & \frac{1}{RK} & \frac{1}{RK}
\end{bmatrix}
\]

- Note that if $N−$ is the datum node, then only the top left entry survives, the other 3 are not present (since we don’t write KCL for the datum node).
• Now consider a VCCS. Its nodal analysis template or stamp is:

\[
\begin{align*}
V_{C+} & \quad N_+ \\
V_{C-} & \quad N_- \\
& \quad G K V_c
\end{align*}
\]

\[
\begin{bmatrix}
N_+ & \cdots & N_-
\end{bmatrix}
\begin{bmatrix}
G K & -G K \\
- G K & G K
\end{bmatrix}
\]

• Finally, consider an independent current source. Its nodal analysis stamp is:

\[
\begin{align*}
& \quad N_+ \\
& \quad N_- \\
& \quad I K
\end{align*}
\]

\[
\begin{bmatrix}
N_+ & \cdots & N_-
\end{bmatrix}
\begin{bmatrix}
0 & 0 \\
0 & 0 \\
- I K & I K
\end{bmatrix}
\]

• In this case we don’t stamp \( Y_n \), but just stamp \( I S \).
NODAL ANALYSIS REDUX

- Advantages:
  - $Y_n$ can be computed by inspection.
  - $Y_n$ is smaller and fuller than the STA matrix.
  - $Y_n$ has non-zero diagonal elements, and is often diagonally dominant (since any diagonal element is the sum of the conductances of the circuit elements connecting to the corresponding node). We will later see why this is a desirable property.

- Disadvantages:
  - Nodal analysis cannot handle floating (not connected to ground) voltage sources since the current through such a source is indeterminate.
  - For the same reason, NA cannot handle VCVS’s
  - NA as discussed cannot handle CCCS’s and CCVS’s

- This motivates Modified Nodal Analysis (MNA) which we cover next.
MODIFIED NODAL ANALYSIS

- Consider the circuit below:

- Step 1: Write KCL
  - (node 1) $i_1 + i_2 + i_3 = 0$
  - (node 2) $-i_3 + i_4 - i_5 - i_6 = 0$
  - (node 3) $i_6 + i_8 = 0$
  - (node 4) $i_7 - i_8 = 0$

- Step 2: Now use branch equations to eliminate as many branch currents as possible.
  - (node 1) $\frac{1}{R_1}v_1 + G_2v_3 + \frac{1}{R_3}v_3 = 0$
  - (node 2) $-\frac{1}{R_3}v_3 + \frac{1}{R_4}v_4 - i_6 = IS5$
  - (node 3) $i_6 + \frac{1}{R_8}v_8 = 0$
(node 4) \( i_7 - \frac{1}{R_8} v_8 = 0 \)

- Note that \( i_6 \) and \( i_7 \) could not be written in terms of branch voltages.

- **Step 3:** Now write down the **unused branch equations**.
  - \( v_6 = ES6 \)
  - \( v_7 = E7v_3 \)

- **Step 4:** Use KVL to eliminate branch voltages from previous equations:
  - \( \frac{1}{R_1} e_1 + G_2 (e_1 - e_2) + \frac{1}{R_3} (e_1 - e_2) = 0 \)
  - \( -\frac{1}{R_3} (e_1 - e_2) + \frac{1}{R_4} e_2 - i_6 = IS5 \)
  - \( i_6 + \frac{1}{R_8} (e_3 - e_4) = 0 \)
  - \( i_7 - \frac{1}{R_8} (e_3 - e_4) = 0 \)
  - \( (e_3 - e_2) = ES6 \)
  - \( e_4 = E7(e_1 - e_2) \)
• This can be written in matrix form as follows:

\[
\begin{bmatrix}
\frac{1}{R_1} + G_2 + \frac{1}{R_3} & -G_2 - \frac{1}{R_3} & 0 & 0 & 0 & 0 \\
-\frac{1}{R_3} & \frac{1}{R_3} + \frac{1}{R_4} & 0 & 0 & -1 & 0 \\
0 & 0 & \frac{1}{R_8} & -\frac{1}{R_8} & 1 & 0 \\
0 & 0 & -\frac{1}{R_8} & \frac{1}{R_8} & 0 & 1 \\
0 & -1 & 1 & 0 & 0 & 0 \\
E7 & -E7 & 0 & -1 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
e_1 \\
e_2 \\
e_3 \\
e_4 \\
i_6 \\
i_7
\end{bmatrix}
=
\begin{bmatrix}
0 \\
IS5 \\
0 \\
0 \\
ES6 \\
0
\end{bmatrix}
\]

• Which is of the form:

\[
\begin{bmatrix}
Y_r & B \\
C & D
\end{bmatrix}
\begin{bmatrix}
e \\
i
\end{bmatrix}
= MS
\]

• Where:

- \(Y_r\) consists of conductances
- \(B\) represents uneliminated currents
- \(C\) and \(D\) represent unused branch equations
- \(e\) and \(i\) are the node voltages and (some) branch currents that will be solved for.
ASSEMBLING MNA MATRIX EFFICIENTLY

- (Just like we saw in the case of NA) the MNA matrix can be constructed by inspection (stamping).

- Let's see some examples. Consider a floating voltage source. Its MNA template is:

  ![Diagram of floating voltage source]

  \[
  \begin{bmatrix}
  N^+ & N^- & i_k \\
  N^- & 0 & 0 & 1 \\
  branch k & 0 & 0 & 0 & 1 \\
  \end{bmatrix}
  \begin{bmatrix}
  0 \\
  EK \\
  \end{bmatrix}
  \]

- The MNA template for a CCVS is:

  ![Diagram of CCVS]

  \[
  \begin{bmatrix}
  N^+ & N^- & NC^+ & NC^- & i_k & i_j \\
  N^- & 0 & 0 & 0 & 0 & 0 \\
  NC^+ & 0 & 0 & 0 & 0 & 0 \\
  NC^- & 0 & 0 & 0 & 0 & 0 \\
  branch k & 0 & 0 & 0 & 0 & 0 \\
  branch j & 0 & 0 & 0 & 0 & 0 \\
  \end{bmatrix}
  \begin{bmatrix}
  i_j \\
  FK_i_j \\
  i_k \\
  \end{bmatrix}
  \]
• General rules for MNA:
  
  — A branch current is always introduced as an additional variable for a **voltage source** or an **inductor**
  
  — For current sources, resistors, conductances and capacitors, the branch current is introduced only when
    
    * Any circuit element depends on that branch current (see example of CCCS below)
    
    * That branch current is requested as an output in the SPICE deck

• The MNA template for a CCCS is:

  ![MNA Template Diagram]

  • Note that the current of branch $k$ depends on that of branch $j$, hence branch $j$ is present in the MNA based circuit formulation.
MNA REDUX

• Advantages:
  – MNA can be applied to any circuit.
  – MNA matrix can be assembled directly from the SPICE input deck by inspection
  – MNA matrix is similar to the NA matrix $Y_n$, with modifications to handle the deficiencies of NA.

• But the problem is that the MNA matrix may have zeros on the diagonal (see MNA template for CCCS and CCVS - they contribute 0 elements along the diagonal).

• The principal minors of the MNA matrix can be singular.

• Since we will be using Gaussian elimination to solve the MNA matrix, these features are undesirable from
a numerical stability point of view. More on this later...