Analog Computers

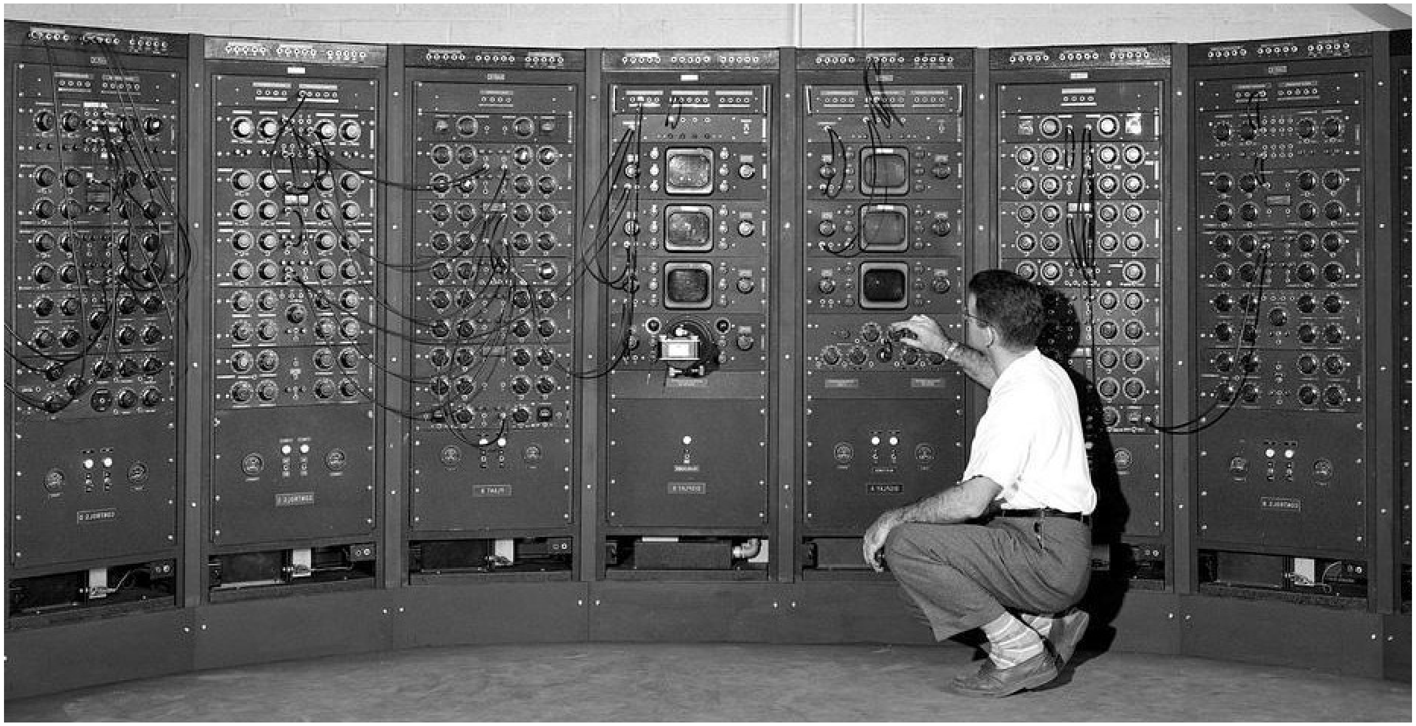
ECE 321: Electronics II

Lecture #11

Please visit Bison Academy for corresponding lecture notes, homework sets, and solutions

Analog Computers

- Electric circuit to implement a differential equation
- Allows you to duplicate the dynamics of an expensive system using an inexpensive circuit



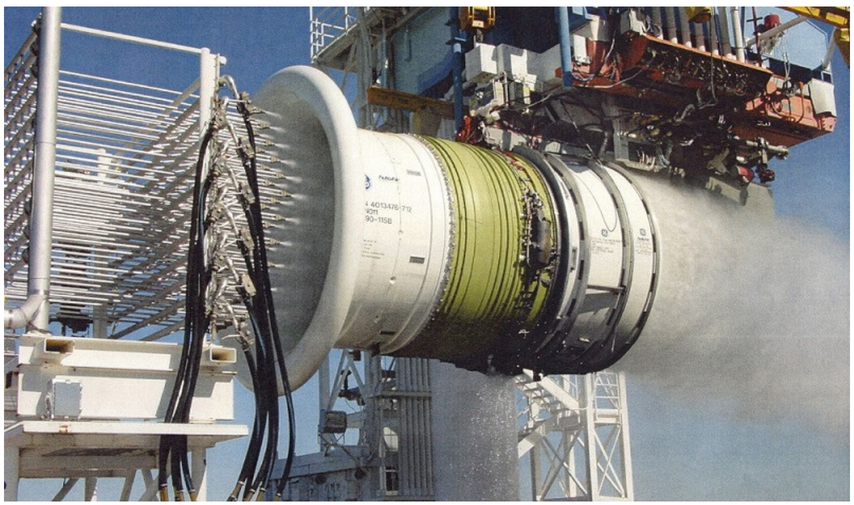
https://images.easytechjunkie.com/1949-electronic-analog-computer.jpg

Analog Computer Application

Jet Engine

https://www.airlineratings.com/did-you-know/how-is-a-jet-engine-tested/

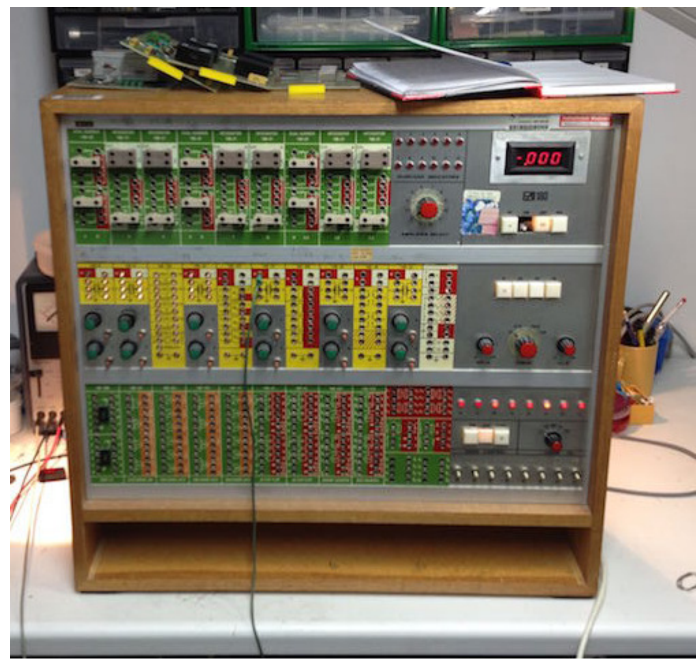
- $10,000 / minute to test
- $4,000,000 if you go unstable



Analog Computer

http://www.analogmuseum.org/english/collection/eai/180/1_small.jpg

- 10 cents / kWh to operated
- Red LED turns on if you go unstable



Analog Computers

Design a circuit to implement a generic proper transfer function

$$Y = \left(\frac{b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + ... + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + a_{n-2}s^{n-2} + ... + a_1 s + a_0} \right) U$$

Solution:

There are many. This is one way to do it. Just to make it more manageable, assume a 3rd-order system

$$Y = \left(\frac{b_2 s^2 + b_1 s + b_0}{s^3 + a_2 s^2 + a_1 s + a_0} \right) U$$

Step 1: Change the problem. Create a dummy state, X

$$X = \left(\frac{1}{s^3+a_2s^2+a_1s+a_0}\right)U$$

$$Y = (b_2s^2 + b_1s + b_0)X$$

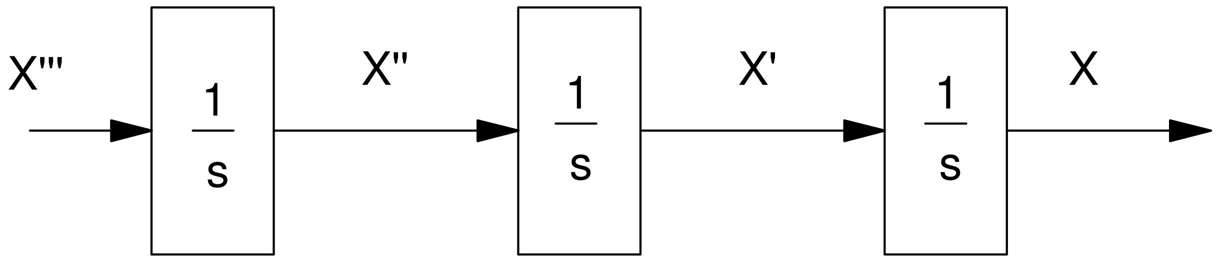
Step 2: Cross multiply and solve for the highest derivative of X:

$$X = \left(\frac{1}{s^3+a_2s^2+a_1s+a_{00}}\right)U$$

$$(s^3 + a_2s^2 + a_1s + a_0)X = U$$

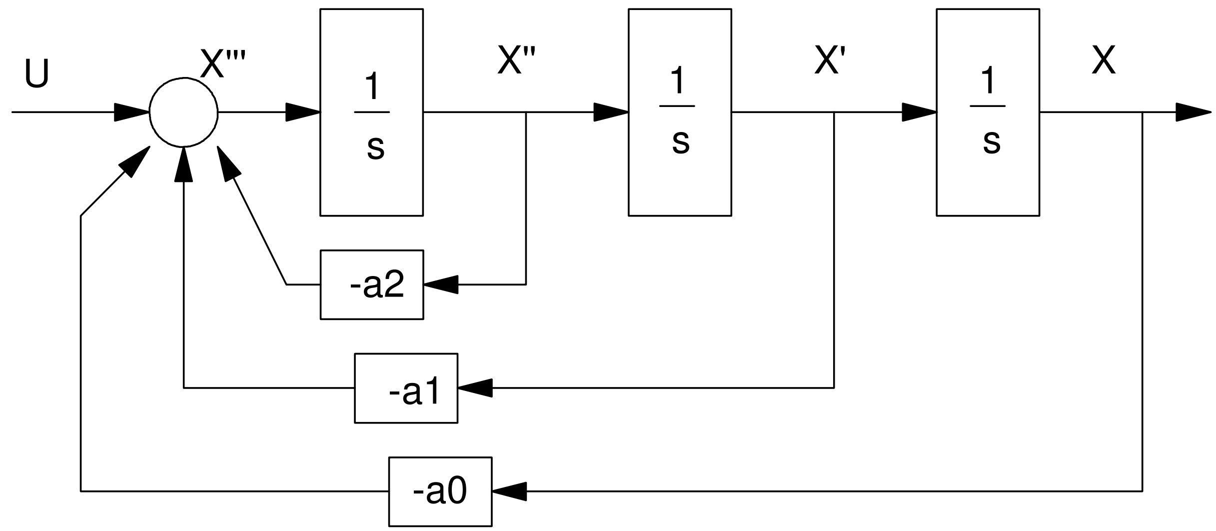
$$s^3X = -(a_2s^2 + a_1s + a_0)X + U$$

Step 3: Given $s^n X$, solve for X by integrating n times (notation: X' means dx/dt)



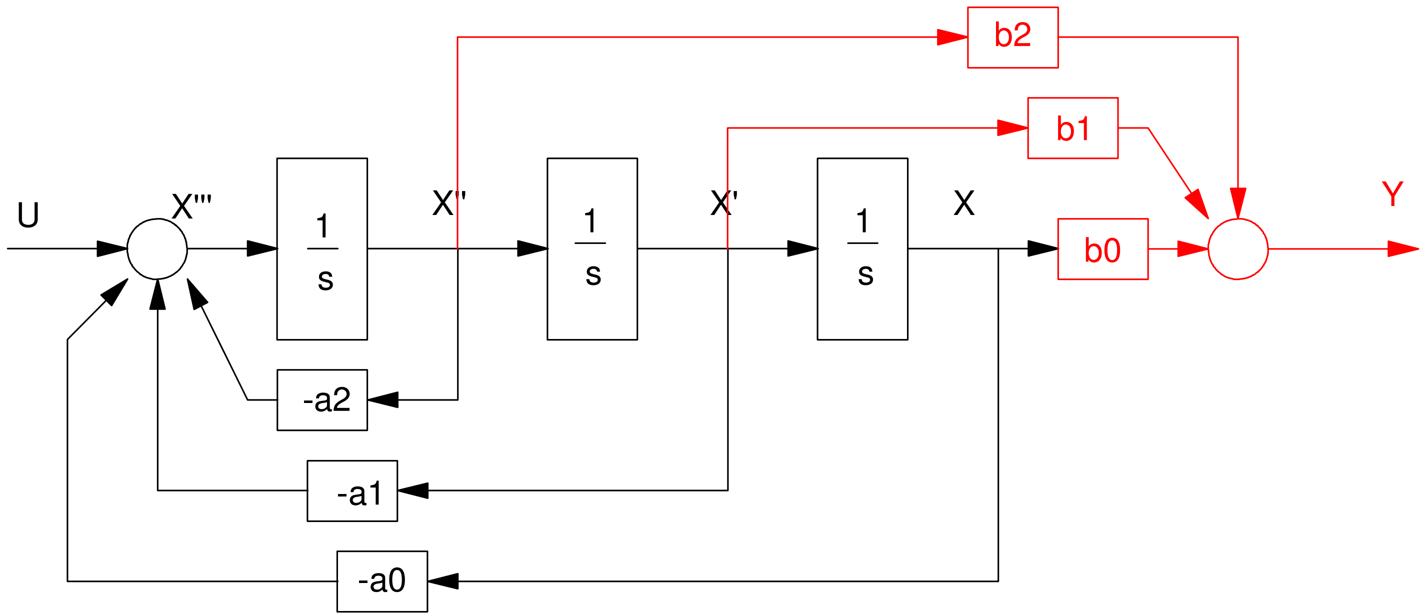
Step 4: Create X''' using the differential equation from step 2:

$$s^3 X = -(a_2 s^2 + a_1 s + a_0)X + U$$

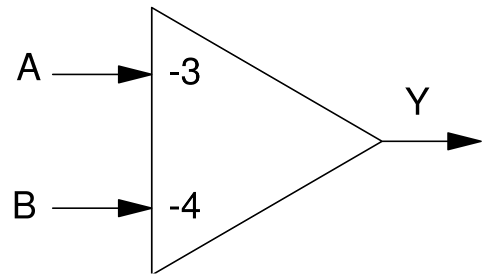


Step 5: Now that you know X and its derivatives, create Y:

$$Y = (b_2 s^2 + b_1 s + b_0)X$$

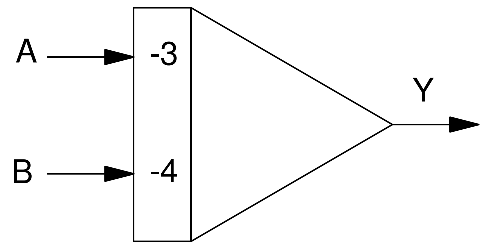


Step 6) Convert to analog computer notation. Here, a triangle means an amplifier:



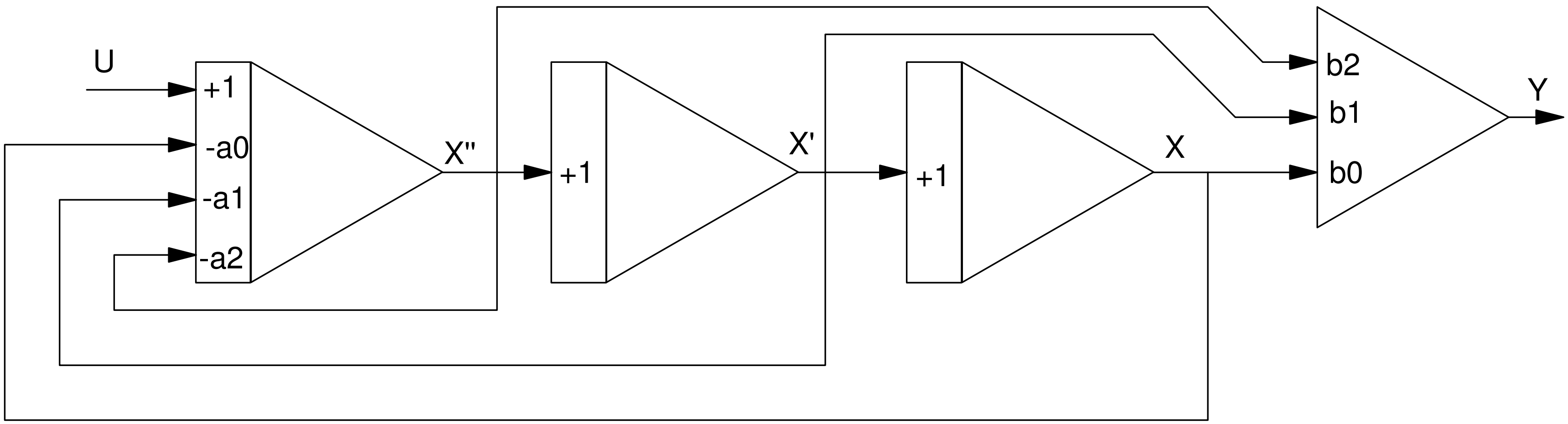
Y = -3A - 4B

whereas a triangle with a box means integrator:



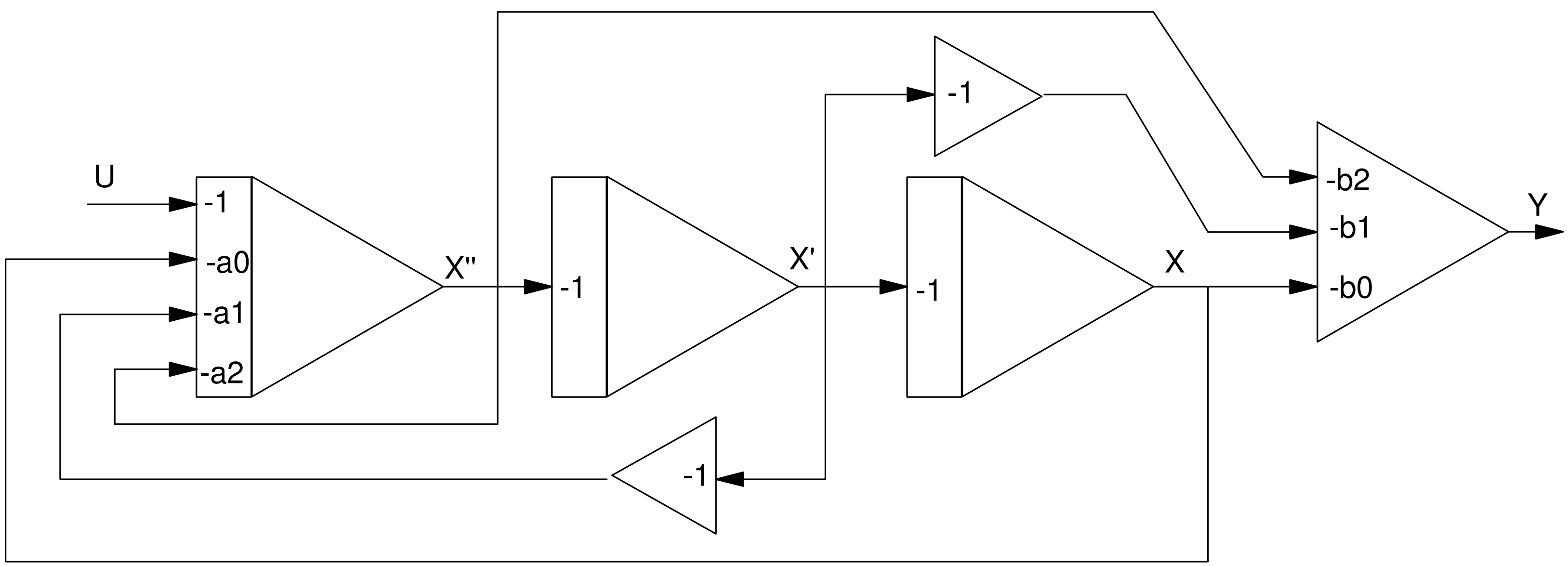
Y = (1/s) (-3A - 4B)

Applying this to the above block diagram:

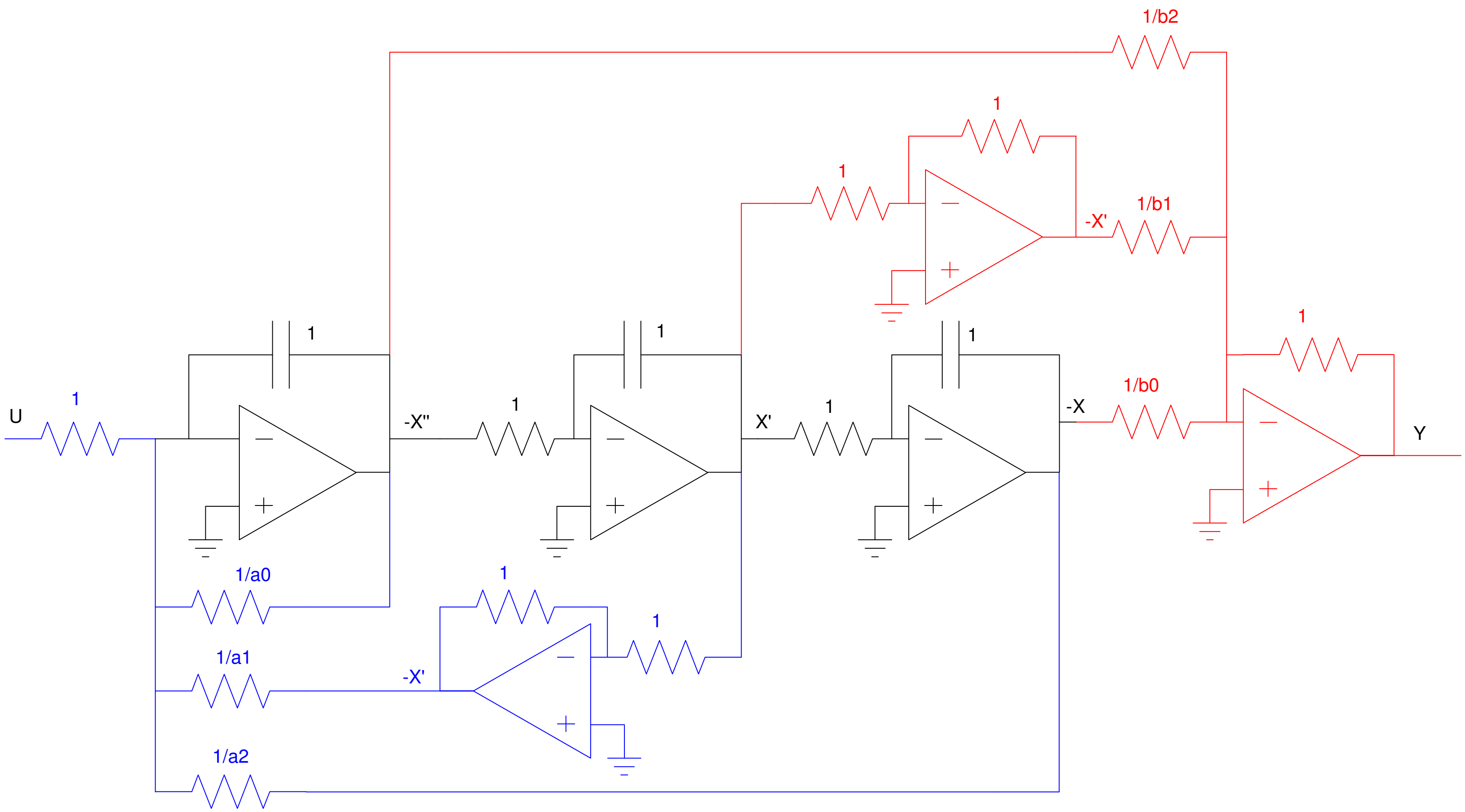


Make all gains negative.

- Keep the sign for all paths from U to Y
- Keep the sign for all loops
- You may have to add inverters



Now convert to an op-amp circuit.



Final Op-Amp Circuit: All units M Ohms and uF.

Sidelight

This technique works well if the poles are close to 1.000

If the poles are not close to 1,

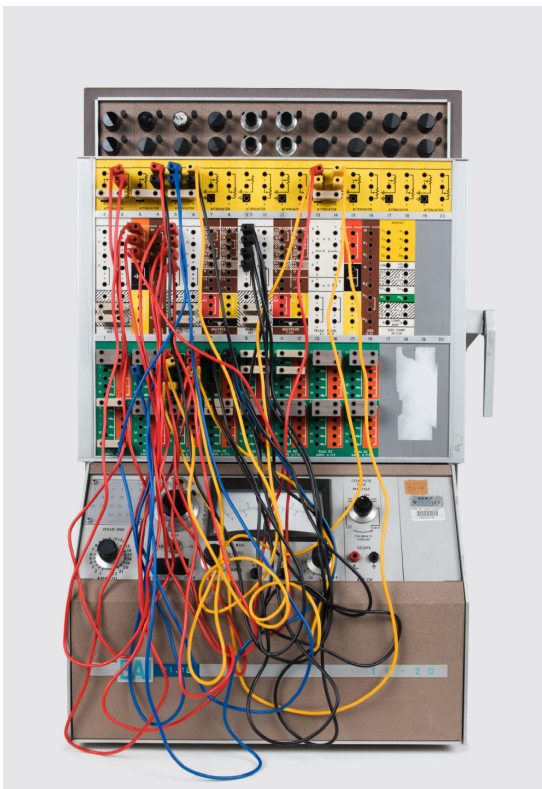
- Scale the poles so that they are close to 1.000
- Design the analog computer using the previous techniques
- Scale the circuit by making C larger (slower) or smaller (faster) to return to the original pole locations.

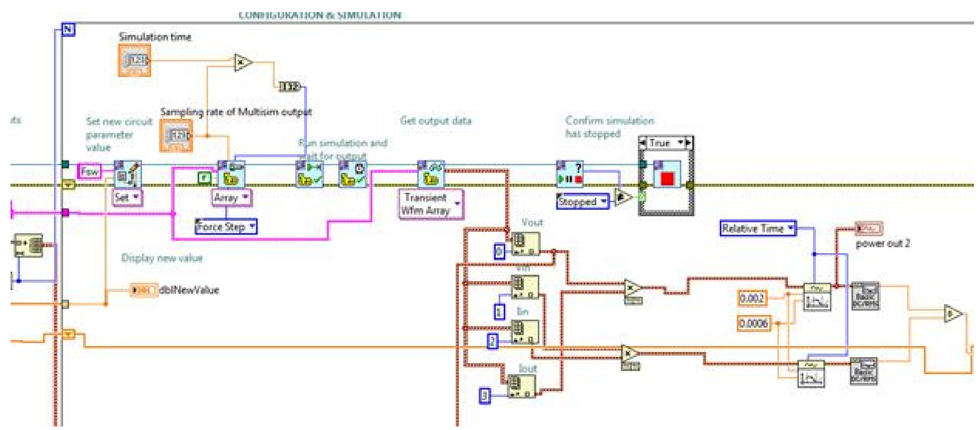
Modern Analog Computers

Anything you can do in hardware you can do in software

Analog computers are now implemented in software

- LabView, Simulink, VisSim





Summary:

Analog computers are just op-amp circuits

- They allow you to implement any differential equation

The heart is the inverting integrating circuit

- Nth-order differential equation requires N integrators

Plus inverting amplifiers and summing junctions

Somewhat dated:

- You can do the same thing with a microcontroller
- Analog computers are tools: for some applications, they work fine. For others, a microcontroller is easier to use.