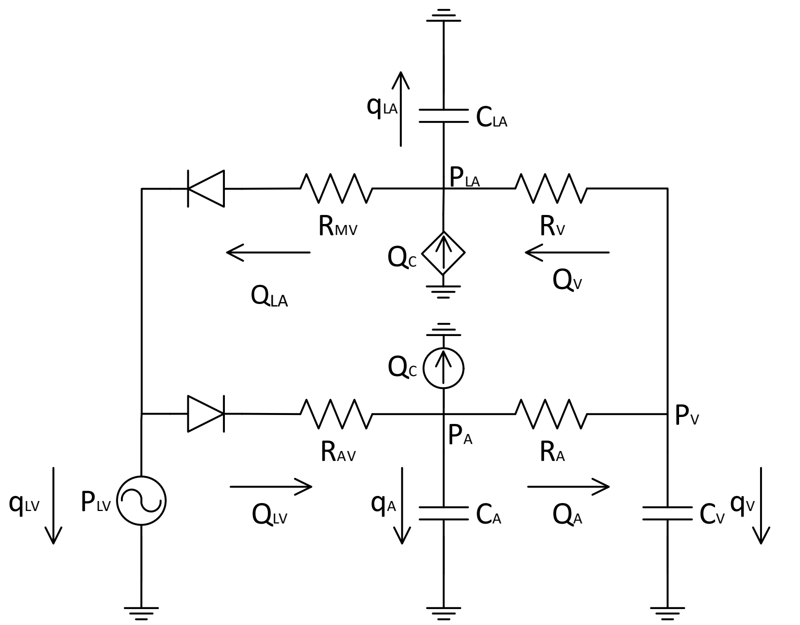
I. ELECTRICAL ANALOGUE CIRCUIT of the CARDIVASCULAR SYSTEM



In the circuit:

- P_{LV} is shown as an AC voltage source. But, in class, we will consider it to be a variable capacitance, $e_{LV}(t)$.
- The coronary blood flow between the aorta and the venous system is shown as Q_C. You can ignore it at this time.
- Aortic and mitral valve resistances are labeled R_{AV} and R_{MV}, respectively. In your report, label them R_a and R_m.
- To avoid writing the capacitance and resistance in your equations as fractions, use their reciprocals as

$$\text{Elastance} \stackrel{\text{def}}{=} e = \frac{1}{C} \text{ and}$$

$$\text{Conductance} \stackrel{\text{def}}{=} \text{g} = \frac{1}{\text{R}}, \text{respectively.}$$

- There are 4 node pressures, which you should use as the state variables, because the capacitance equation yields their derivatives readily, as long as the capacitances are constant (which they are for all pressures except for P_{LV}).

$$\dot{P}_i = \frac{1}{C_i} q_i = e_i q_i$$

- Since e_{LV} is variable, the above capacitance equation won't apply to it. Instead, you will use the chain rule on the LV Elastance equation:

$$P_{LV} = e_{LV}(V_{LV} - V_0) \quad (1)$$

$$\dot{P}_{LV} = \frac{d}{dt}[e_{LV}(V_{LV} - V_0)]$$

$$\dot{P}_{LV} = \dot{e}_{LV}(V_{LV} - V_0) + e_{LV}\dot{V}_{LV} \quad (2)$$

From (1)

$$V_{LV} - V_0 = \frac{P_{LV}}{e_{LV}} \quad (3)$$

Substitue (3)into (2)

$$\dot{P}_{LV} = P_{LV}\frac{\dot{e}_{LV}}{e_{LV}} + e_{LV}\dot{V}_{LV} \quad (4)$$

Also , cardiac volume is given by Kirchoff's current applide to P_{LV} node

$$\dot{V}_{LV} = -q_{LV} = -(Q_{LA} - Q_{LV})$$

$$= -\left(\frac{P_{LA} - P_{LV}}{R_m} - \frac{P_{LV} - P_A}{R_a}\right)$$

$$= -[g_m(P_{LA} - P_{LV}) - g_a(P_{LV} - P_A)]$$

$$= -[g_a P_A - (g_m + g_a)P_{LV} + g_m P_{LA}] \quad (5a)$$

Do not mistake this with cardiac outflow, which only happens during systole when $g_m = 0$

$$Q_{LV} = \frac{P_{LV} - P_A}{R_a} = g_a(P_{LV} - P_A) \quad (5b)$$

Substitue (5a) into (4)

$$\dot{P}_{LV} = P_{LV}\frac{\dot{e}_{LV}}{e_{LV}} - e_{LV}[g_a P_A - (g_m + g_a)P_{LV} + g_m P_{LA}]$$

$$\dot{P}_{LV} = \left[\frac{\dot{e}_{LV}}{e_{LV}} - (g_m + g_a)e_{LV}\right] P_{LV} + e_{LV}(g_a P_A g_m P_{LA}) \qquad (6)$$

- As seen in (5b), the expression for the blood flow, Q_{ij}, across two nodes, i and j, can be found from the pipe equation (Ohm's law). You have 4 of those.

$$Q_{ij} = \frac{1}{R_{ij}}\left(P_i - P_j\right) = g_{ij}\left(P_i - P_j\right)$$

- The expression for the blood flow, q_i, over the capacitance , C_i, can be obtained from Kirchoff's current law. You'll have 4 of those, one for each capacitance including the variable capacitance for the left ventricle:

$$q_i = Q_{into\ node} - Q_{away\ from\ node}$$

- Using these 12 equations (4 P_i, 4 Q_i, 4 q_i) , you can write the state dynamic equations

$$\dot{P}_{LV} = a_{11}P_{LV} + a_{12}P_A + a_{13}P_V + a_{14}P_{LA}$$

$$\dot{P}_{LV} = a_{21}P_{LV} + a_{22}P_A + a_{23}P_V + a_{24}P_{LA}$$

$$\dot{P}_{LV} = a_{31}P_{LV} + a_{32}P_A + a_{33}P_V + a_{34}P_{LA}$$

$$\dot{P}_{LV} = a_{41}P_{LV} + a_{42}P_A + a_{43}P_V + a_{44}P_{LA}$$

- And express the 4-state system dynamics in state-space form:

$$\begin{bmatrix} \dot{P}_{LV} \\ \dot{P}_A \\ \dot{P}_V \\ \dot{P}_{LA} \end{bmatrix} = [A] \begin{bmatrix} P_{LV} \\ P_A \\ P_V \\ P_{LA} \end{bmatrix} \text{ subject to } \mathrm{P_i}(t = 0) = P_{i,0}$$

Here, $[A]$ is the 4x4 State Matrix formed of elements a_{ij}, which you should obtain in order to write the Matlab code and set-up the Simulink program.

II. MATLAB CODE

Start by writing an m.file that you will embed in a user-defined function box in Simulink

- Start by assigning values to system conductances and elastances. Use Khoo references and, if necessary, search the net for normal physiologic values.
- Enter the conductance and elastance values in your m.file as vector strings, G and E, respectively.

```
% System Constants
>>  g_A = G(1,1); g_a = G(2,1); g_m = G(3,1); g_V = G(4,1);
>>  e_A = E(1,1); e_V = E(2,1); e_LA = E(3,1);
```

- Define the state vector P

```
% System Constants
>>  P_LV = P(1); P_A = P(2); P_V = P(3); P_LA = P(4);
```

- Design the User-defined Function Box.

  Assign the states vector P, conductance and Elastance vectors G and E, the variable LV elastance **e_t** and its derivative **e_dot** be your inputs.

  Let the derivative of the state vector, **P_dot**, and the volumetric LV variables **Q_LV**, **V_LV** and **q_LV** be your outputs.

```
>> function [P_dot, , Q_LV, V_LV, q_LV] = cardiovascular_model(P,G,E,e_t,e_dot_t)
```

- Initilize pressure gradients across the aortic and mitral valves

```
>>  p1 = 0; p2 = 0;
```

- Initilize the Flow Vector

```
>>Q = zeros(2,1);
```

- Design your valves (Diodes)

```
%Mitral Valve
>>  if ((P(4) - P(1)) > 0)    %  if P_LA > P_LV
>>      p1 = P(4) - P(1);     %  then ΔP= P_LA - P_LV
>>  end
%Aortic Valve
>>  if ((P(1) - P(2)> 0)      %  if P_LV > P_A
>>      p2 = P(1) - P(2);     %  then ΔP= P_LV – P_A
>>  end
```

- Define the Flow Vector

```
>>  Q = [g_m*p1; g_a*p2];
```
%Q=$\begin{bmatrix} Q(1,1) \\ Q(2,1) \end{bmatrix}$=$\begin{bmatrix} Q_{LA} \\ Q_{LV} \end{bmatrix}$

- Initilize the Cardiac Phase

```
>>  phase = 0;
```

- Let Matlab decide which phase were are in by looking at the flow and open or close the valves

```
>>  if  Q(2,1) > 0 && Q(1,1) == 0        % Q_LV > 0  and Q_LA = 0
>>          phase = 1;                    % Ejection phase
>>          g_m=0                         %Close mitral, leave aortic open
>>   end
```

- Write code for the other phases yourself

- Define your State Matrix here

```
>>  A= [...]
```

- Define the state equations

```
>> P_dot= A*P;
```

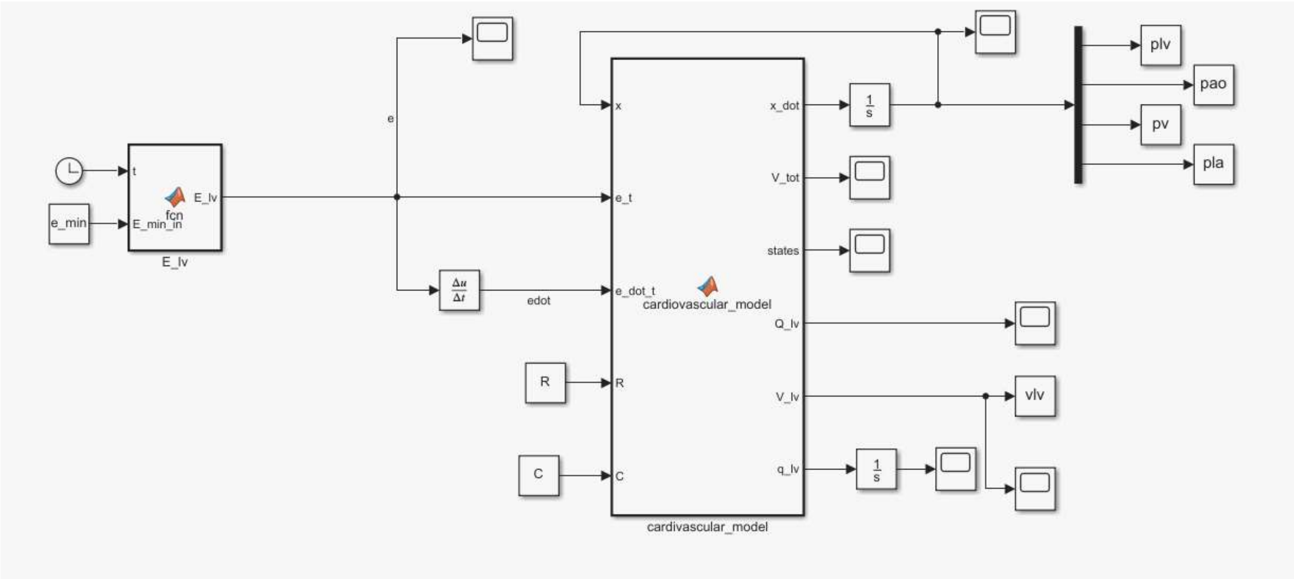
- Define, initialize and calculate Cardiac Output

```
>> Q_LV = 0;

>>  if phase == 1                        %if heart is ejecting
>>      Q_LV = g_a*(P(1) - P(2));        %Equation (5)
>>  end
```

- Define all other outputs here

```
>>  V_lv = LVP*(1/e_t) + V0;
>>  q_lv = -(LVP - AOP)/Ra + (LAP - LVP)/Rm ;

>>end                        %This closes the User-defined Function Box
```

- The Simulink file should look like this:



From the diagram, you may notice that

- V_{LV} can be calculated two different ways. What are they?
- The derivative of the State Vector, x_dot (designated P_dot in your file) is muxed. To observe each pressure individually, the vector has to be demuxed.
- The time-varying elastance of the left ventricle needs to be defined as an external input into the user-defined function box . Below, is a guide for you to write this function as an m.file.

YOUR HOMEWORK CHALLENGE: REDUCE THIS to a 3-STATE SYSTEM by ELIMINATING ONE of the STATES (P_V).

YOU CAN DO THIS USING THE FACT THAT TOTL VOLUME ın the CIRCUIT MUST REMAIN CONSTANT.

ADD THE FORMULA of V_{TOTAL} to the OUTPUT and PLOT IT to SHOW that IT REMAINS CONSTANT.

III. ELASTANCE FUNCTION

The time-varying LV elastance can be expressed as a function of its minimum (end-diastolic) and maximum (end-systolic) values as

$$\boldsymbol{e_{LV}}(t) = (e_{max} - e_{min})\boldsymbol{e_n}(\boldsymbol{t}) + e_{min} \qquad (1)$$

where the activation function $e_n(t)$ is given in terms of the dimensionless time, t_n,

$$t_n = \frac{t}{t_{es}}$$

The systolic duration, t_{es}, is a function of the cardiac period, t_c

$$t_{es} = 0.2 + 0.15t_c$$

$$t_c = \frac{60}{HR}$$

and the heart rate, HR, is given in beats per minute.

According to (1), any definition of the activation function, $e_n(t)$, must satisfy the boundary conditions

$$e_n(t) = \begin{cases} 1 & t = t_{es} \\ 0 & t = t_c \end{cases}$$

To-date, it has not been possible to express $e_n(t)$ analytically, although many empirical formulations exist, which are constructed by curve-fitting experimental P_{LV}- V_{LV} data (collected from animal experiments using a conductance catheter). Below, are given three equations for you to develop a Matlab code and generate the Elastance curve for a single cardiac cycle.

I. The first empirical formula is of the form

$$\boldsymbol{e_n}(\boldsymbol{t}) = \frac{a(t_n)^{\alpha}}{b + c(t_n)^{\beta} + d(t_n)^{\gamma} + e(t_n)^{\delta}} \qquad (2)$$

where

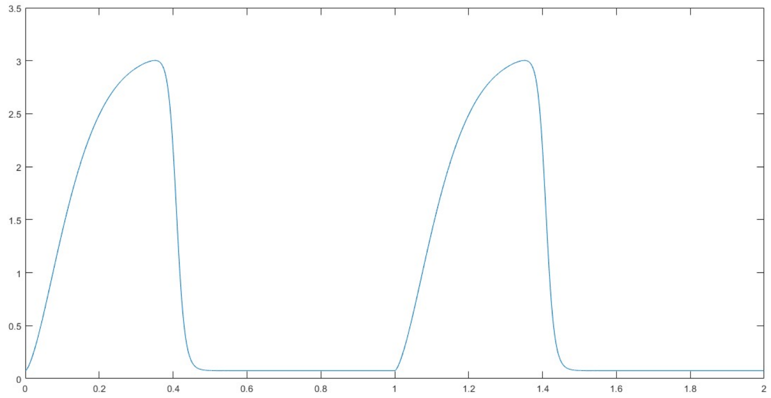
$a = 461.4567, b = 126.2886, c = 333.3319$, d $= 0.3789$, e=1
$\alpha = 1.5, \beta = 1.9, \gamma = 37, \delta = 38.9$

II. The second empirical formula is given by

$$e_n = 2.14\left(\frac{t_n}{a}\right)^{\alpha} \frac{\left[1+\left(\frac{t_n}{b}\right)^{\beta}\right]}{\left[1+\left(\frac{t_n}{c}\right)^{\gamma}\right]} \tag{2}$$

where a=0.7, b=1.17, c=0.6 and $\alpha = 1.5, \beta = 1.9, \gamma = 37$

These formulas should result in a curve similar to this:

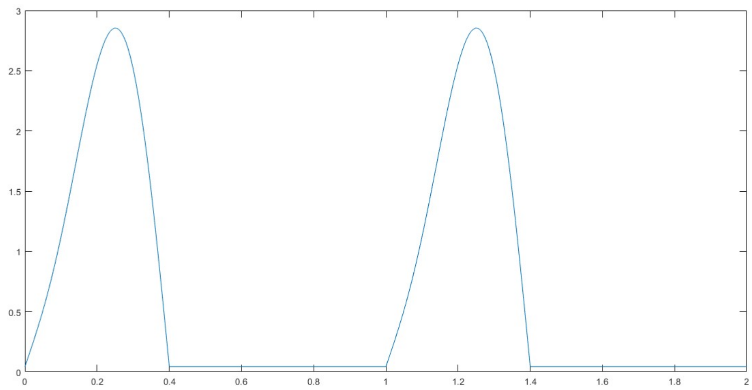


III. The third formulation is the simplest but the least accurate:

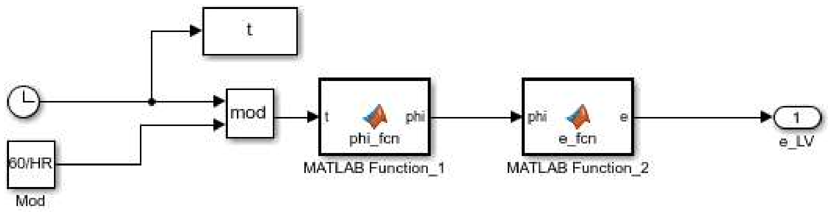
$$t_{es} = 0.400\ sec$$

$$e_{LV} = 0.043\ (1-\varphi) + 2.83\ \varphi$$

$$\varphi = 0.9\ sin\left(\pi \frac{t}{t_{es}}\right) - 0.25\ sin\left(2\pi \frac{t}{t_{es}}\right)$$



This is its output. You can easily see that some of the details are lost. And below is the Simulink file to create the last function.



t
mod
60/HR
Mod
t
phi
phi_fcn
MATLAB Function_1
phi
e
e_fcn
MATLAB Function_2
1
e_LV