Welcome

- Over the next few lessons, we will look in detail at the code that implements the ESC toolbox
- This understanding is not critical to be able to use the toolbox, but it will give insight into how it works
- In this lesson, we study `simCell.m`

Preliminaries

- We look first at the function header
- Defines inputs and outputs

```
% function [vk,rck,zk,OCV] = simCell(ik,T,deltaT,model,z0,iR0,h0)
%   ik - current, where (+) is discharge
%   T  - temperature (degC) - can be different at every point in time
%   deltaT - sampling interval in data (s)
%   model - standard model structure
%   z0 - initial SOC
%   iR0 - initial resistor currents as column vector
%   h0 - initial hysteresis state
%
%   vk - cell voltage at each time instant
%   rck - resistor currents through R-C branches
%   zk - cell states of charge
%   OCV - cell OCV at each time instant
function [vk,rck,hk,zk,sik,OCV] = simCell(ik,T,deltaT,model,z0,iR0,h0)
  ik = ik(:); % Force data to be column vector(s)
```


Retrieve model parameter values

- The next part of the function retrieves model parameter values
- Also computes A_{RC} factor `RCfact` and applies η to $i[k]$

```
  % Get model parameters from model structure
  RCfact = exp(-deltaT./abs(getParamESC('RCParam',T,model)))';
  Q = getParamESC('QParam',T,model);
  etaParam = getParamESC('etaParam',T,model);
  G = getParamESC('GParam',T,model);
  M = getParamESC('MParam',T,model);
  M0 = getParamESC('M0Param',T,model);
  RParam = getParamESC('RParam',T,model);
  R0Param = getParamESC('R0Param',T,model);

  etaik = ik; etaik(ik<0) = etaParam*ik(ik<0); % modify input current
```

Simulate state equations of model

- Next, simulate SOC, R–C currents, hysteresis

```
  % Simulate SOC states
  zk = z0-cumsum([0;etaik(1:end-1)])*deltaT/(Q*3600);
  if any(zk>1.1), warning('Current may have wrong sign as SOC > 110%'); end

  % Simulate the dynamic states of the model
  rck = zeros(length(RCfact),length(etaik)); rck(:,1) = iR0;
  hk=zeros([length(ik) 1]); hk(1) = h0; sik = 0*hk;
  fac=exp(-abs(G*etaik*deltaT/(3600*Q)));
  for k = 2:length(ik),
    rck(:,k) = diag(RCfact)*rck(:,k-1) + (1-RCfact)*etaik(k-1);
    hk(k)=fac(k-1)*hk(k-1)-(1-fac(k-1))*sign(ik(k-1));
    sik(k) = sign(ik(k));
    if abs(ik(k))<Q/100, sik(k) = sik(k-1); end
  end
  rck = rck';
```


Compute voltage

- Finally, the code computes the voltage estimate and returns

```
 % Compute output equation
  OCV = OCVfromSOCtemp(zk,T,model);
  vk = OCV + M*hk + M0*sik - rck*RParam' - ik.*R0Param;
return % from simCell.m
```


Summary

- You have now learned how `simCell.m` works
- First, retrieves model parameter values we will need
- Next, computes model states $z[k]$, $i_R[k]$, $s[k]$, and $h[k]$
- Finally, combines states to produce voltage estimate