Instruction for Running the MATLAB Code for the Li/SVO-CFx Battery Thermo-Electrical Model

Tool Reference

RST Reference Number: RST24ES04.01

Date of Publication: 09/22/2025

Recommended Citation: U.S. Food and Drug Administration. (2025). *Computational Model for Li/SVO-CFx Batteries used in Implantable Medical Devices* (RST24ES04.01). https://cdrh-rst.fda.gov/computational-model-lisvo-cfx-batteries-used-implantable-medical-devices

For more information

Catalog of Regulatory Science Tools to Help Assess New Medical Devices

Disclaimer

About the Catalog of Regulatory Science Tools

The enclosed tool is part of the Catalog of Regulatory Science Tools, which provides a peer-reviewed resource for stakeholders to use where standards and qualified Medical Device Development Tools (MDDTs) do not yet exist. These tools do not replace FDA-recognized standards or MDDTs. This catalog collates a variety of regulatory science tools that the FDA's Center for Devices and Radiological Health's (CDRH) Office of Science and Engineering Labs (OSEL) developed. These tools use the most innovative science to support medical device development and patient access to safe and effective medical devices. If you are considering using a tool from this catalog in your marketing submissions, note that these tools have not been qualified as Medical Device Development Tools and the FDA has not evaluated the suitability of these tools within any specific context of use. You may request feedback or meetings for medical device submissions as part of the Q-Submission Program.

For more information about the Catalog of Regulatory Science Tools, email RST_CDRH@fda.hhs.gov.

Instruction for Running the MATLAB Code for the Li/SVO-CFx Battery Thermo-Electrical Model

Overview:

This Matlab code, **Thermal_EC_Model_LiCFxSVO_ICD_V1.m**, is designed to simulate the thermoelectrical dynamics of Li/SVO-CFx (Lithium/silver vanadium oxide and carbon-monofluoride) batteries commonly used in Implantable Cardioverter Defibrillators (ICDs). This simulation tool predicts the battery depth of discharge, surface temperature, and terminal voltage during its operation under load conditions corresponding to typical ICD power requirements. The model simulates the battery voltage and capacity over its lifetime. The user can adjust the inputs and parameters to investigate their impact on the battery longevity.

Prerequisites:

MATLAB Installation: Ensure that MATLAB is installed on your system to run the code. This model in developed with MATLAB 2022b and is being tested with higher versions too.

Code Overview:

The code does not require any specific MATLAB toolbox.

It consists of three for loops, two of them for generating two current inputs and the third one for solving the thermoelectrical dynamic model. The last for loop utilizes a forward Euler solver to calculate various states and output for simulating the battery performance under realistic loads. The states are the battery state of charge **(x1)**, the transient voltage which is the voltage drop across the RC pair in the model **(x2)**, and the battery surface temperature **(x3)**. The inputs are the ambient temperature **()** and the device load which includes the housekeeping current **(u)** and the defibrillation current **(u_shock)**. The model output is the battery terminal voltage versus time.

Parameters and Initial Values:

The parameters and initial values are defined at the beginning of the script, including body temperature, simulation time, time intervals, cell capacity, initial state of charge (SOC), load continuous current, and shock pulsation.

Here is a list of all variables (model parameters and inputs):

BodyTemperature = 37; In degrees Celsius

ambientT = BodyTemperature+273.15; In degrees Kelvin

Q = 2000; (Battery nominal capacity in mili Amper hour (mAh))

th(1,1) = 1/((Q/1000)*3600); (1/the battery capacity in Amper Second)

th(1,2) = 0.0042; 1/RC = 1/ (0.1ohm * 2400 Farad) = 0.0042*
*Q_battery (A.s) = CV = C*3v = 7200 A.s =>>> C = 2400F >>> th2 = 1/(0.1*2400) = 0.0042 [1/ohm.Farad]

th(1,3) = 0.0036; Thermal time-constant =hA/mCp = 13*5e-4/30e-3*60 = 0.0036 [1/second]

th(1,4) = 0.0427; % 1/mCp m=30g, Cp=13J/s-m^2-k

th(1,5) = 1/2400; % 1/C , C=2400 Farad

th(1,6) = 0.27; % [Ω] Ohmic resistance

** th(1,7) to th(1,11) are the depth of discharge-dependent entropy coefficients of the thermal model obtained from the RST tool reference [5]*

dt = 10; The simulation time intervals in seconds

Battery_life_simulation_time = 6; Years of the battery life during the device operation

initSOC = 0.95; Initial state of charge of the battery in decimal percentage (x1)

x2 = 0; Initial voltage drop [volt] across the RC pair in the Equivalent Circuit Model (ECM) (when the battery is not initially under the load, then x2 is equal to 0 [volt])

x3 = 37+273.15; Initial battery surface temperature in Kelvin

u = -25e-6; Constant load current (housekeeping) in Amper

u_shock = -3; Defibrillation current for charging the defibrillation capacitor in the device

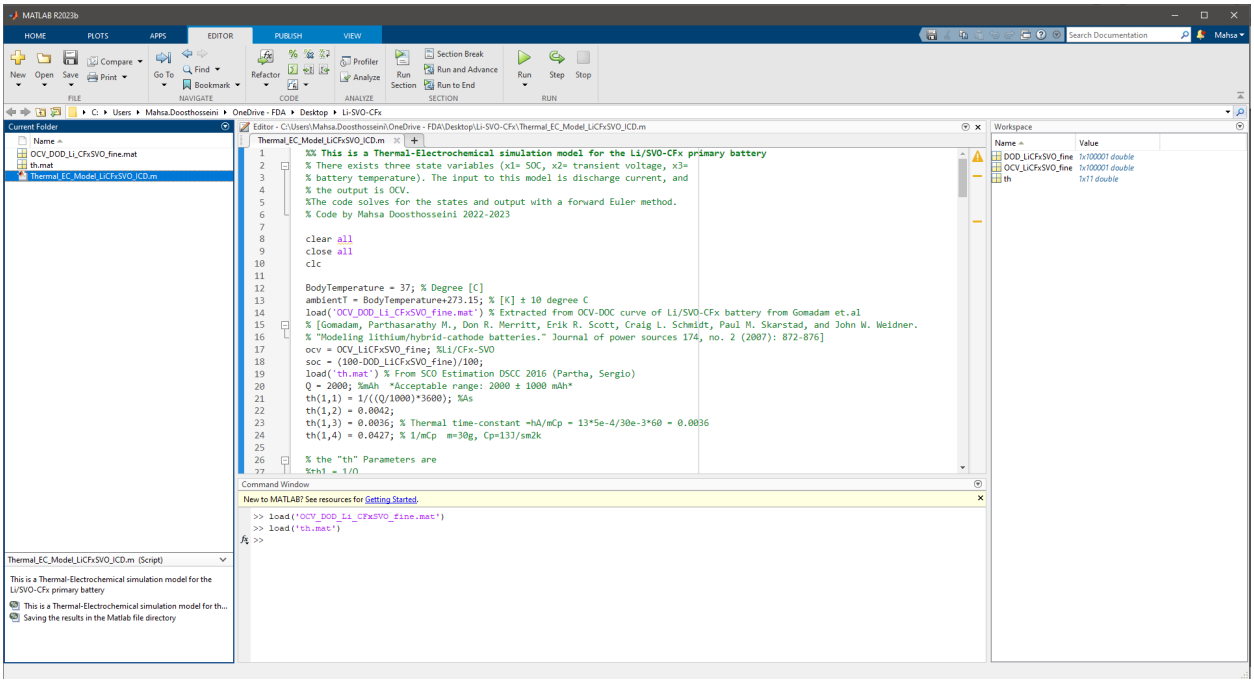
The code requires two provided supplemental data files to run. One file (**OCV_DOD_Li_CFxSVO_fine.mat**) contains OCV-DOD (Open Circuit Voltage – Depth of discharge) characteristic curve data, and the other data file is the (**th.mat**). The OCV_DOD curve illustrates the relationship between the battery's open circuit voltage and its depth of discharge which is a unique feature of the battery for assessing its capacity, efficiency, and overall health.

The data for this curve is measured in a work done by Gomadam et. al. which is the RST tool reference [1].

The th.mat file contains 11 parameters in which the first 6 parameters, (th(1,1) to th(1,6)) are the battery electrical and thermal parameters explained above in the code and the remaining 5 parameters , (th(1,7) to th(1,11)), are the entropy coefficients obtained from the work done by Mendoza et. al in the RST tool reference [5].

Execution:

Save the **Thermal_EC_Model_LiCFxSVO_ICD_V1.m, OCV_DOD_Li_CFxSVO_fine.mat and th.mat** files in a folder on your computer. Open the **Thermal_EC_Model_LiCFxSVO_ICD_V1.m** in MATLAB. Ensure that the MATLAB working directory corresponds to the folder where you have stored the files. Run the script, and it will load the data files, then generate the input current pulsation intervals and solve the model's differential equations. The screenshot below shows the MATLAB script. Lines 12 to 24 are some of the adjustable parameters within the model, which users can modify in the script as detailed in the following section.



Results, including battery terminal voltage, depth of discharge (DOD), surface temperature, discharge current, and OCV-DOD characteristic curve, are plotted and stored in the MATLAB file directory.

Adjustments:

Users can modify battery capacity (Q), initial SOC, simulation device life-time, ambient temperature, load current magnitude (u and u_shock), timing and number of shocks in each year,

and the time interval between shocks in the code to observe their impact on simulation results and to predict the battery lifetime. Adjusting these parameters should be within the ranges provided below for meaningful results:

Q = 2000± 1000 mAh

0.6 ≤ initSOC ≤ 1

Device_LifeTime = 6.5; % ± 3 [years]

ambientT = BodyTemperature+273.15 ± 10 degree C

u = -25e-6 ± 25e-6 Amp

u_shock = -3± 2 Amp

```
% Shock Pulses Configuration
config = struct();
config(1).shocks = 2; % Number of shocks in the first year
config(1).first_shock_time_sec = 2*30*24*3600; % Time in seconds for the first shock in the first year
config(2).shocks = 1; % Number of shocks in the second year
config(2).first_shock_time_sec = 2*30*24*3600; % Time in seconds for the first shock in the second year
config(3).shocks = 3; % Number of shocks in the third year
config(3).first_shock_time_sec = 2*30*24*3600; % Similarly, set for each year as needed
% ... and so on for each year you want to configure

% Pulse width and interval settings (global or per year)
pulse_width_sec = 10; % Pulse width in seconds
interval_between_shocks_sec = 30*24*3600; % Interval between shocks in seconds
```

Important Notes:

Ensure adherence to the provided comments and parameter ranges.

Changing the battery chemistry will change the model parameters and the characteristic curve of the cell. Therefore, the model needs to be modified accordingly for different battery chemistries. To apply this model to a different lithium battery chemistry, the initial stage involves conducting an OCV-DOD characteristic curve experiment and collecting data to generate the curve specific to the new battery chemistry. Subsequently, the model needs to be parameterized for the new

chemistry, with optimization to enhance alignment with actual data. The final phase would be to subject the model to accuracy assessment techniques and comparison with real experimental data, to ensure precise prediction of battery performance. For further details, please refer to the publication link provided in the RST.