# Modelica Chemical Library - User’s Guide

Marek Mateják

(first draft)

## Abstract

New library for chemical calculations is released. It is based on equilibrating the electro-chemical potentials of the substances as following the modern theory of physical chemistry. The library is very general. The chemical solution has full thermodynamic state with possibility to connect it with thermal, mechanical and electrical components of Modelica Standard Library 3.2.1. This paper show how to implement the lead-acid battery, the hydrogen burning engine and the chloride shift membrane channel in cellular membrane of human red cells. All these examples uses only very small amount of base components such as component for chemical solution, component for chemical substance or component for chemical reaction. Behind these components are only fundamental laws of thermodynamics and physical chemistry. The most of parameters are properties of the chemical substances, so the user does not set the dissociation coefficients of the reactions, because they can be calculated from Gibbs energies of the chemical reaction, which can be expressed from formation Gibbs energies of the substances using Hess’ law.

## Introduction

The Chemical library becomes from very successive library for physiological calculations called Physiolibrary. This library allows to move different substances in different direction through membrane at the same time, which was not possible using Modelica.Fluid package because stream constructs move the substances only in direction of the main stream. We used the Physiolibrary to implement the biggest model of human physiology of the year 2010 called HumMod. Now we called the model Physiomodel and we are continuing the extended it in more detailed microscopic and chemical levels. The macroscopic processes and regulations of human physiology are already validated by experiments on animals and human. However the chemical processes in the models was until today a black-boxes with inputs and outputs defined more with empirical relationships than with strict physical chemistry theory. Using this empirical behavior it was well formed expectations for the elementary processes, which can be theoretically described in more details using last modern fundamental relations of thermodynamics and physical chemistry.

This new chemical library solves the more detailed chemical environment of human cells and cellular chemical processes, where the Physiolibrary failed. For example the equilibration of osmolarities on capillary membrane works well, because of small amount of impermeable proteins. However, we found that in cellular membrane these relations was not in good agreement with measured data. These data shows Raftos et al. for human blood red cells, called erythrocytes, where total molarity of plasma is around 289 mmol/L and molarity of intracellular space is 207 mmol/L at osmotic equilibrium. The explanation of these disproportions was hidden in physical chemistry and when we equilibrate the chemical potential of water instead of osmolarity the capillary membrane and also this cellular membrane reach the expected values as measured in experiments. The other problem with old Physiolibrary approach was, that it does not automatically calculate the membrane equilibriums. The very specific blocks for calculating the Donnan’s equilibriums at glomerular membrane was created to reach expected concentrations of electrolytes at semipermeable membrane. However, there was not generated any membrane electric potential, which is the result of this electrolyte’s equilibrium. Even the neural information was spread by simplification describing of amplitudes of frequencies on specific neural drives. Now the new approach can automatically solve the Donnan’s equilibria on semipermeable membrane together with Nernst membrane potential as a simple consequence of the equilibrated electro-chemical potentials of the permeable substances.

We realized during building the new Modelica support of these chemical processes that they are general enough to describe much more than the equilibrium on the erythrocyte cellular membrane. The result is a library, which allows to create any type of chemical reaction in any type of solution at any conditions. We made it in one hand with studding thermodynamics and physical chemistry relations behind. In Modelica it seems really that they can be only rewritten the selected base definitions from these theoretical approach. For example the definition of electro-chemical potential at chemical substance component, the thermodynamic relations at chemical solution component and the simple equilibration of electro-chemical potentials in all components of chemical processes. And all starts to work above our expectations.

Now we define only tree models of substances: ideal gas, incompressible substance and subunit substance. However the user can very easy redefine their own model of substance just by inserting the right expression for pure substance activity coefficient, molar volume, free molar formation entropy and free molar formation enthalpy based on substance data, temperature, pressure, electric potential and ionic strength. As object-oriented design allows also the definition of substance data is the part of the substance model. So the user can design the substance model based on totally different substance data, because this record is used only in the previously mentioned functions of the model, which should be also redefined in new substance model.

In our examples we will work only with ideal gas substances, which will be all gas substances, and with incompressible substances, which will be all liquid or solid substances. The definition data are molar mass of the substance, number of charge of the substance, molar heat capacity of the substance, free formation enthalpy (as known tabulated value for many chemical substances) and free formation Gibbs energy (as tabulated value). And the parameter density only for incompressible substance. So the full definition of ideal gas substance is only to create the record of the five parameters. This way was already defined more than 35 real chemical substances in example package of this chemical library. The usage of these predefined substances data are very simple. It the parameter dialog of the chemical substance can be selected the right record with this data as shown in Figure 1.

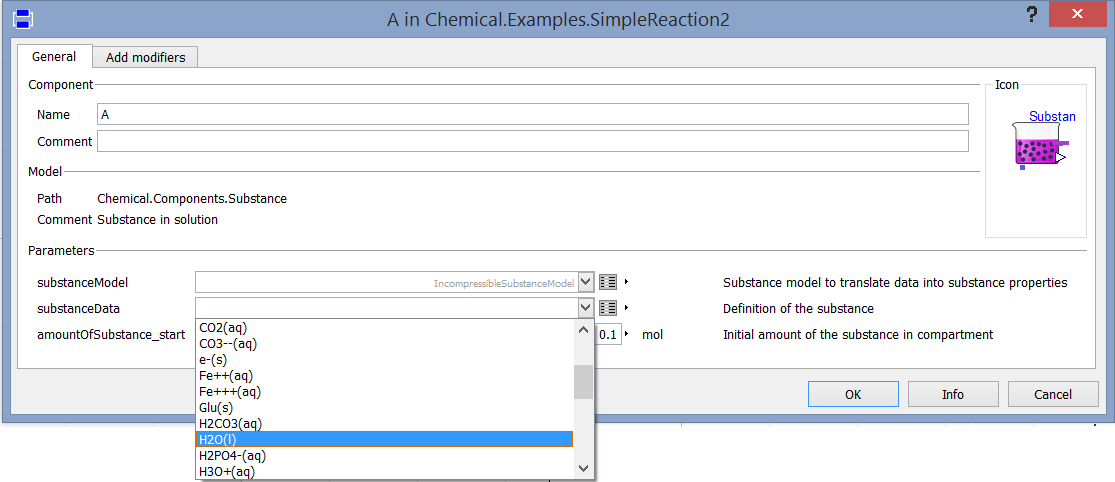


Figure 1, Setting of the predefined chemical substance, where (s) means solid phase, (aq) means dissolved in water, and (g) means gas phase and (l) means liquid phase.

This setting is typically the most important setting of each chemical model. All equilibrium coefficients are automatically solved using the selected substance data. As a result for example the chemical reaction component need to define only the stoichiometry coefficient and from the connected substances it reach the equilibrium at the right equilibrium coefficient.

## Example of lead-acid battery

The lead-acid electrochemical cells are characterized with two chemical reactions:

PbO2(cathode) + HSO4-(liquid)+ 3 H+(liquid) +2 e-(cathode) <-> PbSO4(cathode) + 2 H2O(liquid)

Pb (anode) + HSO4- (liquid) <-> PbSO4(anode) + H+(liquid) +2 e-(anode)

Building of one cell of lead-acid battery using the library starts with definition of three solutions. Two for electrodes and one for acid solution. This can be done by drag and drop of the library class Components.SimpleSolution class into the diagram. The first instance we called “cathode” and the second “solution” and the last “anode” as in Figure 2A. We set the parameter ElectricalGround to “false” for all of these solutions. Now we can to specify the chemical substances inside the chemical solutions. We drag and drop of the library class Components.Substance into the “solution” as chemical substances H2O(l), H+(aq), HSO4-(aq) representing the liquid aqueous solution of sulfuric acid, into the “cathode” PbSO4(s) and PbO2(s) representing as the elements of positive electrode, and into “anode” the substances Pb(s) and PbSO4(s) representing the elements of negative electrode, see Figure 2B. As was mentioned for all this substances must be selected appropriate substance data definition, e.g. Examples.Substances.Water\_liquid, Examples.Substances.Lead\_solid, and so on. The last substance very special substance is an electron. This class is called Components.Electrone and it must be added for each electrode as in Figure 2C to translate electron flows from the chemical reaction to the electric current. Each of these substances must be connected to the appropriate solution using solution port as expressed in Figure 2B,C. Having all substances it is possible to implement the chemical reactions. Drag and drop of library class Components.Reaction for both chemical reactions and setting they parameters into appropriate number of reactants and products and their stoichiometry allows to connect each substance with the reaction as expressed in Figure 2D. This setting can be done using parameter dialog of the cathode chemical reaction as there are four types of substrates (nS=4) with stoichiometric coefficients 1 for first and second reactant, 3 for third reactants and 2 for forth reactants (s={1,1,3,2}) and there are two types of products (nP=2) with stoichiometry 1 for PbSO4 and 2 for water (p={1,2}) following the chemical scheme of the first chemical reaction above. After setting the number of reactants and products there is possible to connect the substances with reaction. Each instance of reaction was an array of connectors for substrates and an array of connectors for products and the user must be very careful to connect each element of this array with another substance in the same order as defined stoichiometric coefficients. This means that for example the water must be connected in index 2 of products of the first chemical reaction, because the second product was stoichiometry 2 as defined for H2O. The second chemical reaction must be set analogically as nS=2, nP=3, p={1,1,2} with connections of substance ports of Pb to substrate[1], HSO4 to substrate[2], PbSO4 to product[1], H+ to product[2] and e- to procuct[3] as graphically represented in Figure 2D.

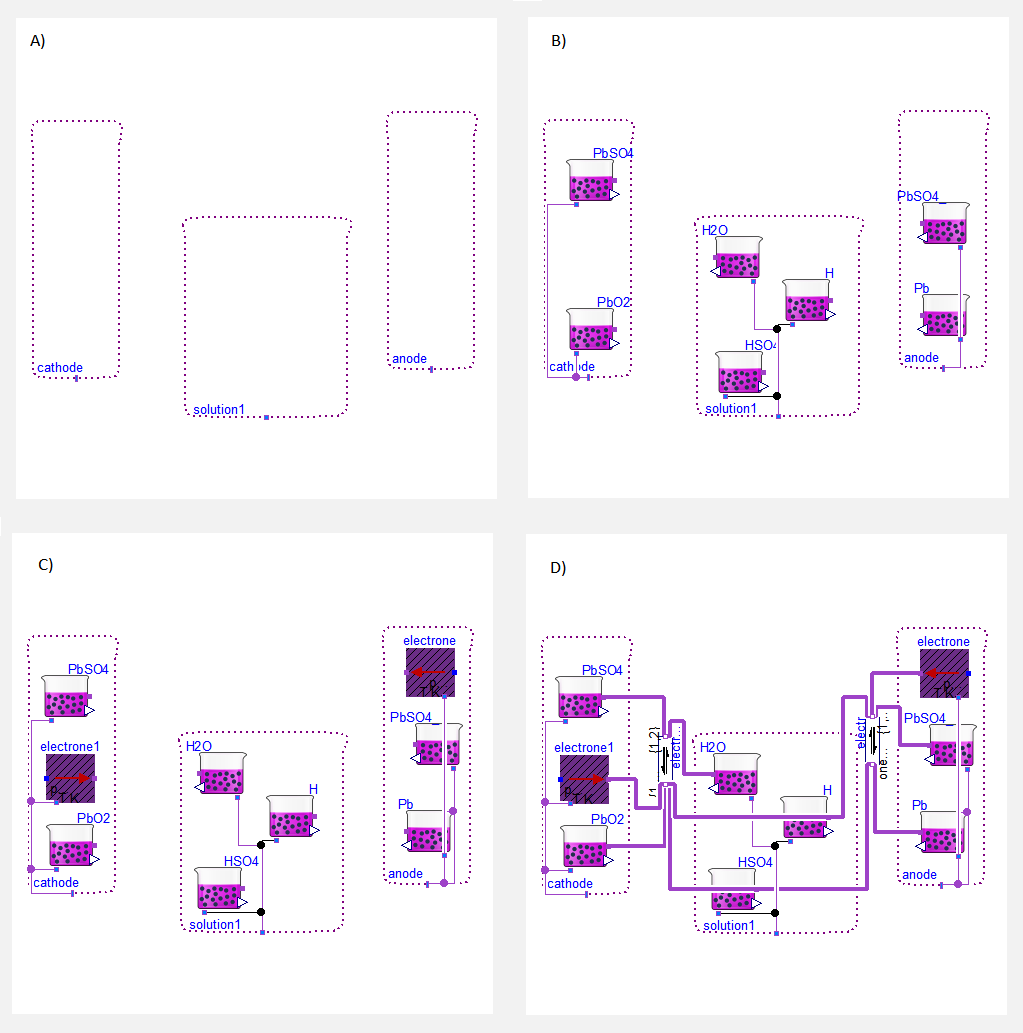
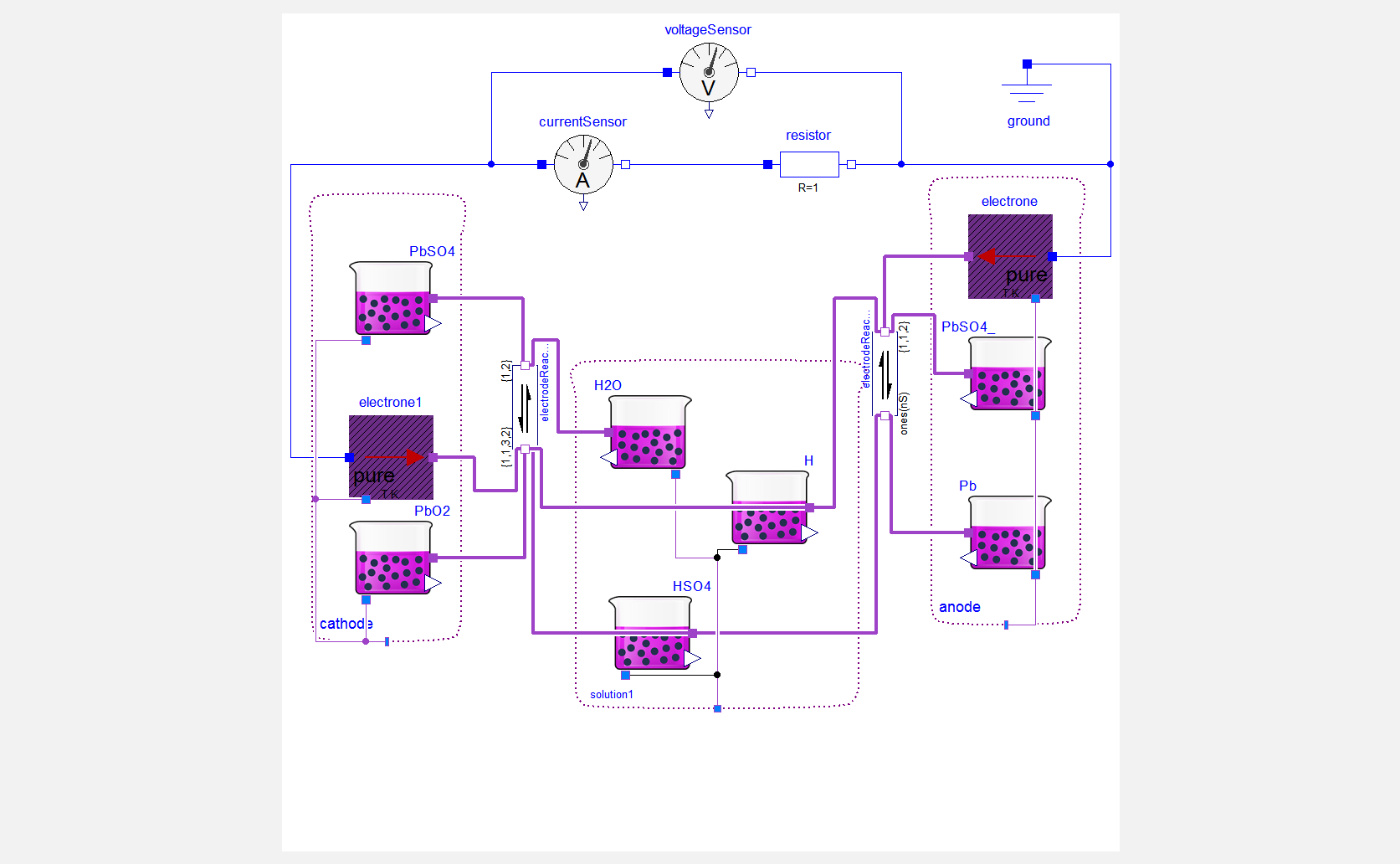
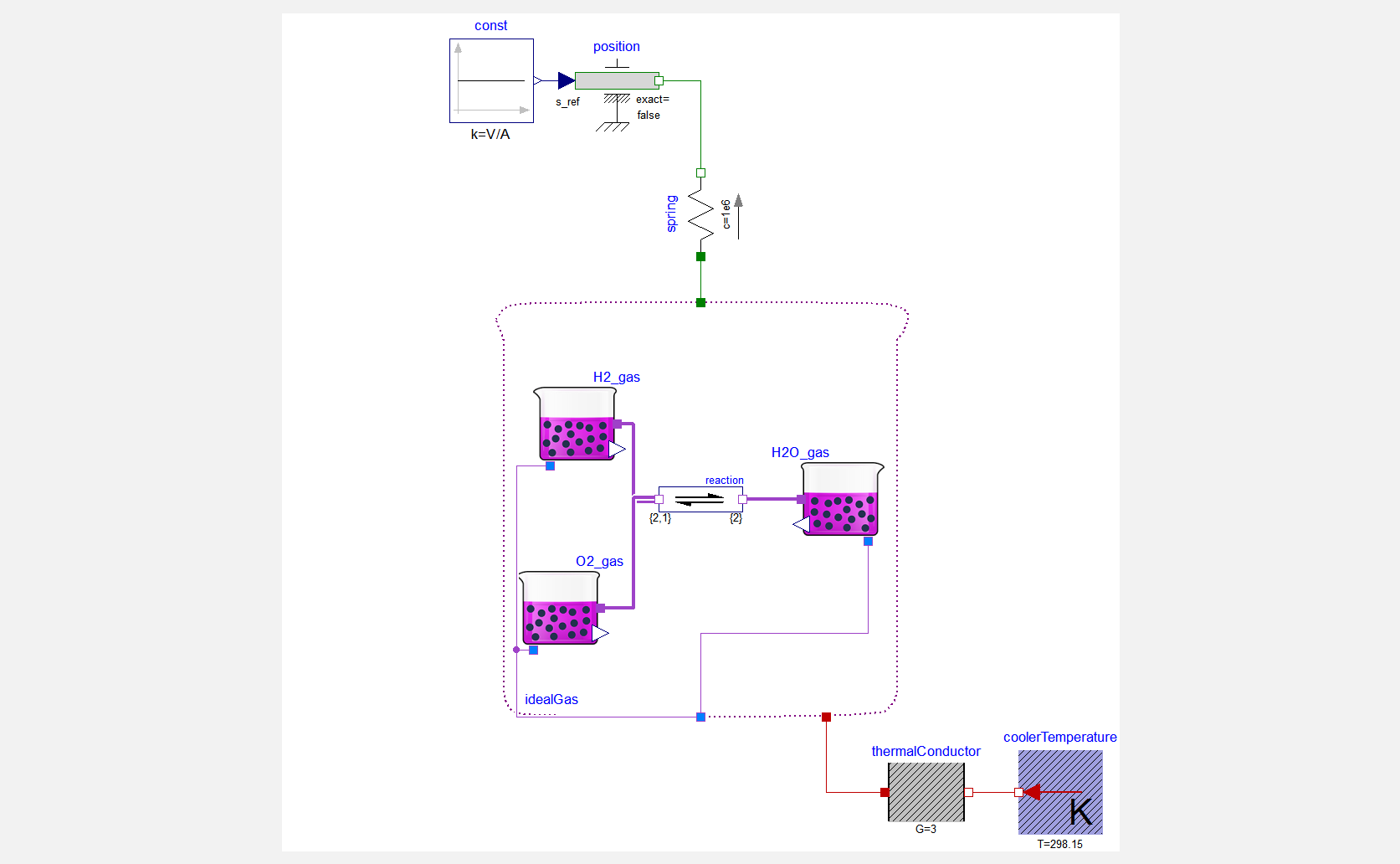


Figure 2, Building of one electro-chemical cell of lead-acid battery in four steps: A) adding solution components, B) adding substance components, C) adding electron components, D) adding reaction components

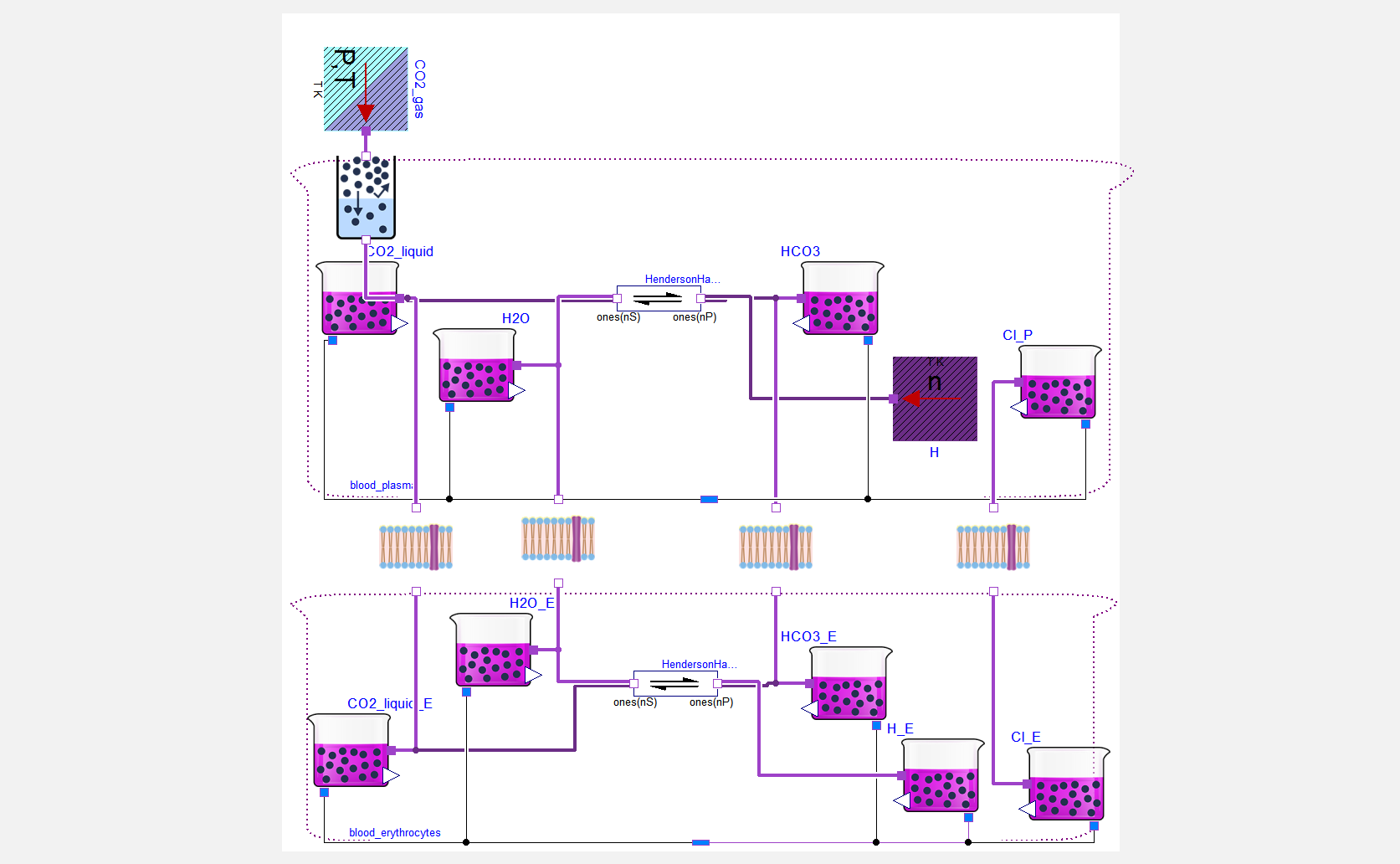
Now, the electrochemical cell is already implemented. However the simulation need the initial state of substances, which for the fully charged battery means that almost all elements of cathode is PbO2 and almost all elements anode is Pb. At this state can be the sulfuric acid very concentrated, what increases the effectivity of the electrochemical cell. To set this state it is possible just double-click on PbO2 and Pb and set the amount e.g. 1mol, the same as the total amount of “cathode” and “anode” components. These total amounts of solutions must be equal or greater than sum of all substances inside. To set them the user must double click on the border of solution and using the parameter dialog rewrite the value of amountOfSolution\_start also to 1mol for both electrodes. If we went to examine also mass of volume properties of electrodes, there must be set the initial values of these physical quantities. However they in this example do not have impact to generated electric currents or electric potentials, so without we can ignore them this time without loss of generality. To set the pure concentrated sulfuric acid we can set the amount of SO4 and H+ also to 1 mol, what makes the total amount of substances in liquid “solution” 2 mol (solution1. amountOfSolution\_start=2). This fully charged ideal state is ready to simulate.



## Example of hydrogen burning engine



## Example of chloride shift



## Results

## Discussion