

# Tvorba modelů s Physiobrary – cirkulace a přenos krevních plynů

## Úvod

Na tomto modelu si vyzkoušíme práci s rozsáhlou knihovnou. Zároveň si ukážeme jednoduchost práce s knihovnami, základy regulace zpětné vazby a expandable konektory.

## Zadání

1. Rozšiřte model cirkulace, který jsme dělali na cvičení o regulaci průtoku (Frank-Starlingův zákon)
2. Vytvořte model transportu CO<sub>2</sub> analogicky k modelu O<sub>2</sub>, který jsme dělali na cvičení.
3. Rozšiřte model transportu CO<sub>2</sub> o regulaci dýchání.
4. Spojte všechny tři subsystémy do jednoho modelu pomocí BusConnectoru (expandable connector).
  - a. Objem kompartmentů *solute*, které nám představují žíly a arterie, v modelech O<sub>2</sub> a CO<sub>2</sub> napojte na objem *veins* a *arteries* v modelu cirkulace.
  - b. Faktor regulace CO<sub>2</sub> (výstup z dělení) přiveďte jako faktor minutového objemu levého srdce.
5. Otestujte funkci následujících situací:
  - a. TBA..

## Physiobrary

Physiobrary je malá knihovna určená k vývoji modelů zejména fyziologických regulací. Její poslední verzi naleznete na <https://github.com/MarekMatejak/Physiobrary>. K tomu aby physiobrary fungovala správně (displayUnits), je třeba přetáhnout jeden soubor (*Physiobrary/Resources/DymolaSettings/displayunit.mos*) do adresáře *insert* v adresáři Dymoly (např. *C:\Program Files\Dymola 2014\insert\*). Správnou funkci ověříme tím, že v *Physiobrary.Types.Examples.Units* když poklepeme na *parametricClass*, tak všechny hodnoty budou právě 1.

## Jednotky

Pozor, některé (většina) jednotek je ze základních SI. Ale my pracujeme s jednotkami odvozenými a přepočítanými. K automatickému přepočtu používáme *displayUnits*, můžeme tak zadávat parametry přímo v cílových jednotkách a zároveň sledovat průběhy. Pozor, počítá se ale stále v základních!!!

Proto není dobré používat Real proměnné, ale vždy jen jednotky odvozené. Balíček *Physiobrary.Types.Examples.Units.ParametricClass* ukazuje jejich použití. K tomu je potom potřeba si dodefinovat některé základní konstanty – nemůžeme již používat normální Real k propojení těchto konektorů. Je to kvůli tomu, aby se nemátly přepočty jednotek.

## Expandable konektor

Je konektor, který obsahuje právě tolik proměnných, kolik do něj zapojíme – rozšiřujeme ho tedy prostým propojením. V tom velmi pomůže prostředí, které si propojení „pamatuje“. Doporučuji následující pořadí:

1. vložit do všech subsystémů expandable konektor (*Physiobrary.Types.BusConnector*)
2. vytvořit celkový model, vložit do něj subsystémy a ty propojit (zde tedy jejich *BusConnectory*)

3. napojovat vstupy a výstupy v jednotlivých subsystémech.

Dodržíme-li toto pořadí, Dymola nám při napojení na BusConnector nabídne seznam již použitých napojení (případně až po kliknutí na <AddVariable>).

### Počítání v ekvilibriu

Physiolibrary je teoreticky schopná počítat modely přímo v ustáleném stavu. Tuto vlastnost pro její složitost teď nebudeme využívat, jen aby vás nemátla celková komplexnost knihovny.

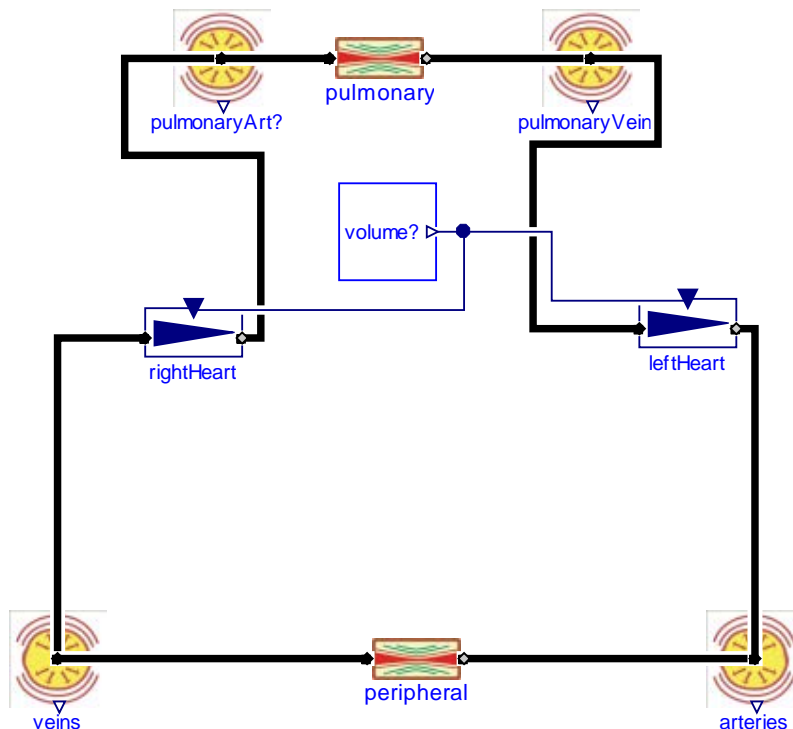
## Subsystémy

Budeme pracovat s několika subsystémy. Bloky cirkulace a přenosu kyslíku jsme se zabývali na cvičení, v této úloze budeme vytvářet systém přenosu CO<sub>2</sub> a doplníme některé regulace.

Jednotlivé bloky si nejdřív sestavíme a otestujeme zvlášť, poté je teprve připojujeme na BusConnector.

### Cirkulace

Jednoduchá cirkulace s pouze minutovými průtoky. Sledujte tlaky v jednotlivých částech řečiště a celkový minutový průtok.



### Tabulka hodnot

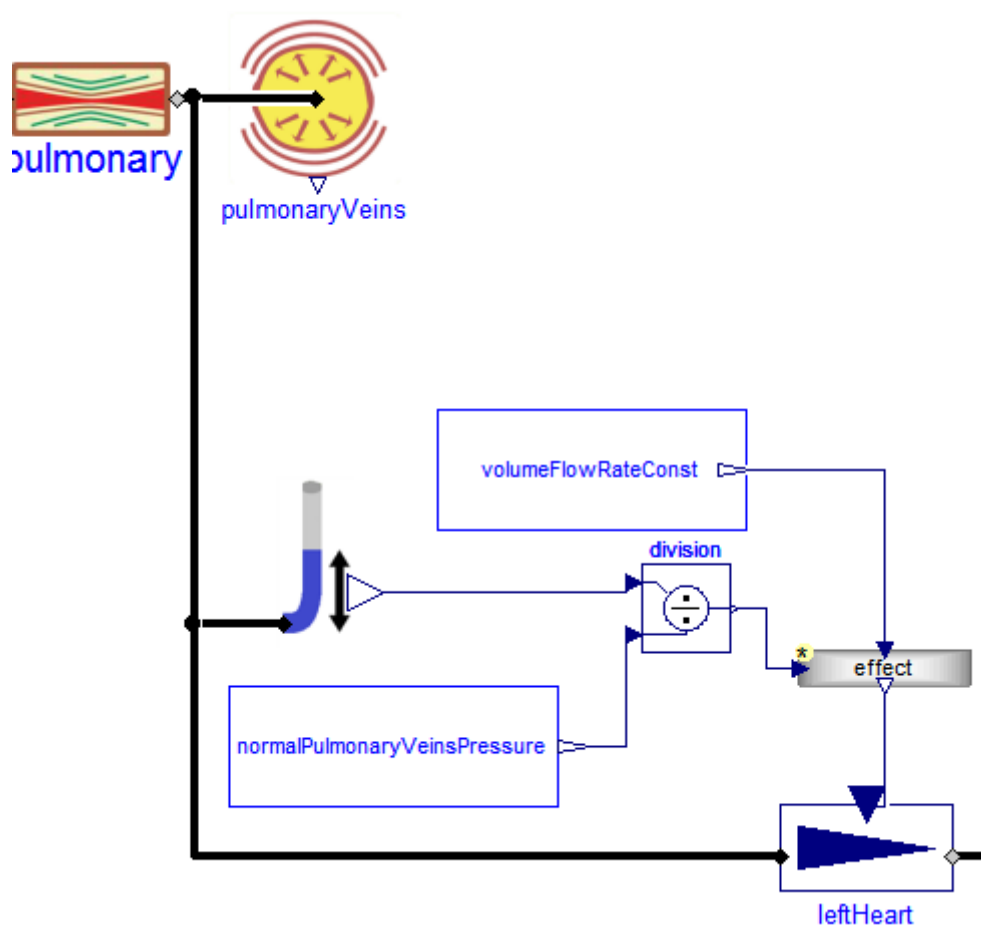
Hydraulic.ElasticBaloon	PulmonaryArteries	0 ml, 6 ml/mmHg, init 100ml
Hydraulic.ElasticBaloon	PulmonaryVeins	0 ml, 80 ml/mmHg, init 400ml
Hydraulic.ElasticBaloon	Arteries	0 ml, 10 ml/mmHg, init 1000ml
Hydraulic.ElasticBaloon	Veins	0 ml, 1750 ml/mmHg, init 3500ml
Hydraulic.Resistor	Pulmonary	558 ml/mmHg/min
Hydraulic.Resistor	Peripheral	57 ml/mmHg/min
Hydraulic.Pump	rightHeart	
Hydraulic.Pump	leftHeart	
Types.Constants. VolumeFlowRateConstatnt		5500 ml/min

--	--	--

## Regulace průtoku krve v oběhu

Duplikujte předchozí model (right-click – new – duplicate class) a upravte model cirkulace tak, aby se srdeční pumpa regulovala pomocí *preload* (tlaku na vstupu) a to jak u systémového, tak pulmonálního okruhu.

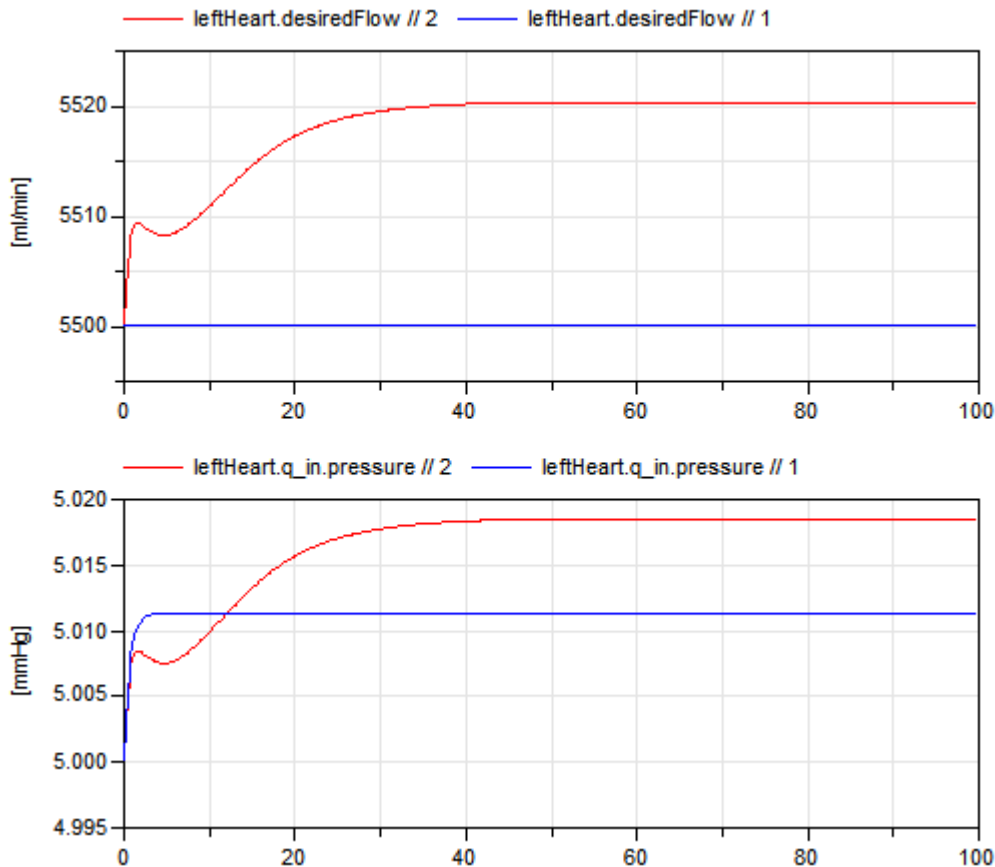
Types.Constants.VolumeFlowRateConst	VolumeFlowRateConst1	5500 ml
Types.Constants.PressureConst	normalPulmonaryVeinsPressure	5 mmHg <sup>1</sup>
Hydraulic.PressureMeasure	pulmPressure	
Blocks.Factors.Effect	Effect	
Modelica.Blocks.Math.Division	Division	
<i>To samé i pro regulaci pravého srdce, akorát jiný parametr:</i>		
Types.Constants.PressureConst	normalSystemicVeinsPressure	2 mmHg <sup>2</sup>



Obrázek 1: Detail řízení levého srdce. Výstup ze senzoru tlaku je podělen normálním tlakem (tj. mezivýsledek je 1) - pozor co s čím dělíte - výsledek je pak pronásoben tokem při normálním tlaku. Klesne-li tedy tlak, klesne i výsledek dělení a tím klesne i celkový průtok pumpy.

<sup>1</sup> Tedy hodnota tlaku na vstupu levého srdce předchozího neregulovaného modelu

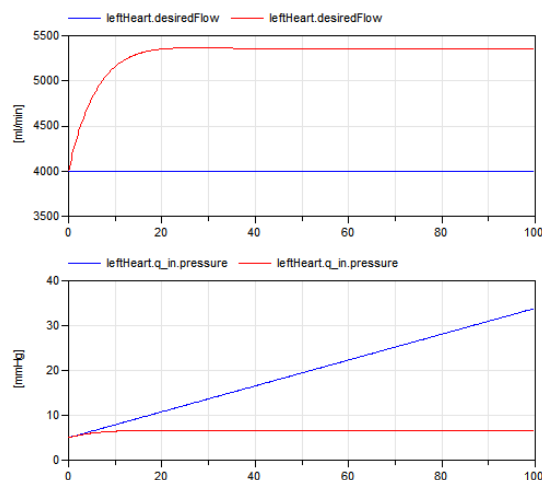
<sup>2</sup> Tedy hodnota tlaku na vstupu pravého srdce předchozího neregulovaného modelu



Obrázek 2: Obvod s regulací (červeně) vs obvod bez regulace. Rozdíl v ustálené hodnotě je dán mírně odlišnou referencí (hodnoty parametrů normalPulmonaryVeinsPressure a normalSystemicVeinsPressure oproti hodnotám v neregulovaném modelu).

## Experiment

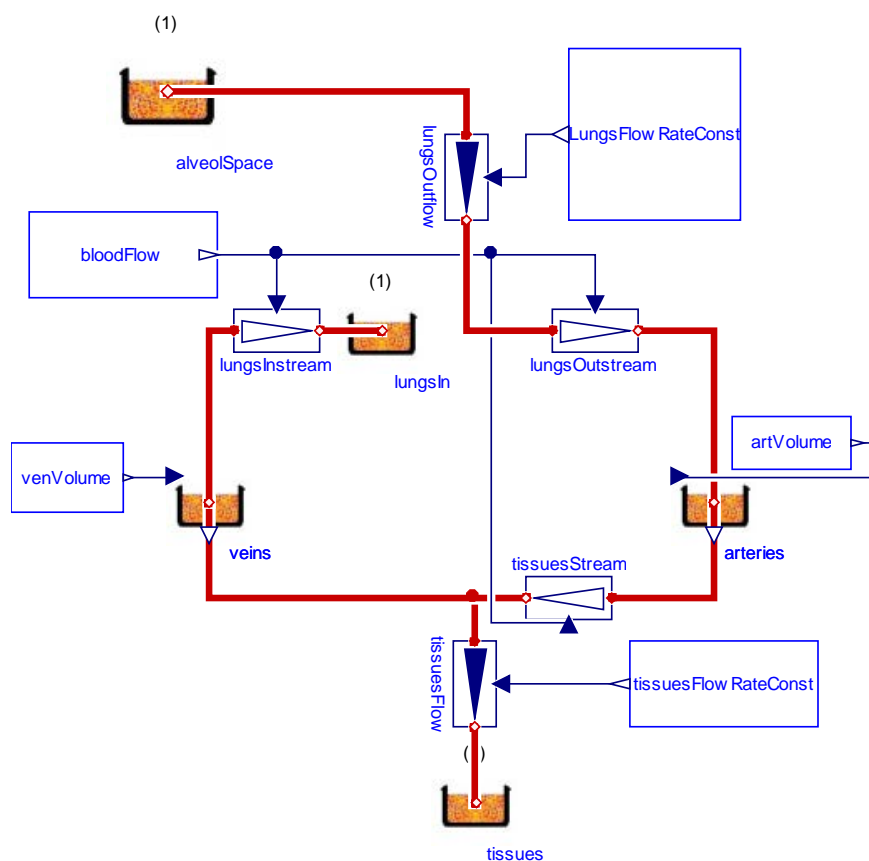
Podstatu a důvod regulace ozřejmíme v následující experimentu – snižte výkon (zde tedy parametr volumeFlowRateConst) **POUZE levého srdce** (například z důvodu infarktu) na 75 % a to jak v minimálním neregulovaném modelu (budete muset každému srdci přiřadit jednu konstantu, ne jednu společnou jako dosud), tak v regulovaném.



Obrázek 3: Výsledek experimentu. Zatímco v regulovaném obvodu stoupne tlak před levým srdcem, tím stimuluje tok a ten se normalizuje, v neregulovaném obvodu sice držíme průtok, ale tlak (a objem) lineárně roste.

## 02

Přenos kyslíku – podobný modelu, který jsme dělali minule. Zde si dávejte pozor na jednotky, jestli je zapisujete správně.



Chemical.UnlimitedStorage	alveolSpace	1 (je to fuk)
Chemical.UnlimitedStorage	lungsIn	1 (je to fuk)
Chemical.UnlimitedStorage	tissues	1 (je to fuk)
Chemical.Substance	Arteries	0.02 mol
Chemical.Substance	Veins	0.02 mol
Chemical.MolarStream	LungStream	
Chemical.MolarStream	lungOutStream	
Chemical.MolarStream	tissuesStream	
Chemical.SoluteFlowPump	lungsOutFlow	
Chemical.SoluteFlowPump	tissuesFlow	
Types.Constants.VolumeFlowRateConst	bloodFlow	5.5 l/min
Types.Constants.MolarFlowRateConst	LungsFlowRate	42 mmol/min
Types.Constants.MolarFlowRateConst	TissuesFlowRate	18 mmol/min
Types.Constants.VolumeConst	VenVolume	3.5 l
Types.Constants.VolumeConst	ArtVolume	1.5 l

## ***CO2***

Sestavte model CO2 podobně jako model O2 s tím rozdílem, že plíce nám oxid uhličitý odebírají a tkáň naopak přidávají. Hodnoty vezměme stejné jako u kyslíku, protože musí probíhat výměna 1:1

### ***Regulace koncentrace oxidu uhličitého***

*TBC..*

### ***Spojení***

Spojte submodely pomocí BusConnectoru.

### ***Regulace průtoku krve v oběhu pomocí koncentrace oxidu uhličitého***

*TBC..*