

Kardiovaskulární reakce na ortostatickou zátěž

K dispozici máte přichystaný soubor MOS4_uloha.mo

Na cvičení jsme si udělali

Vyrobíme si bloky:

- Zdroj tlaku (a)
- Rezistenci (b)
- Chlopeň (c)
- Elastický kompartment (d)
- Zdroj Elasticity (e)

Zdroj tlaku

Bude obsahovat v sobě parametr *tlak*, který bude přiveden na jeho *konektor*.

Rezistence

Jako odpor v elektrotechnice, čili 2 konektory a pro ně patřičné Kirchhoffovy zákony.

Chlopeň

Dioda, rovněž bude respektovat KZ. Tento blok bude trochu složitější. Bude obsahovat booleovskou proměnnou *off* (`boolean off`) a pomocnou proměnnou *s*. V sekci rovnic definujeme, že proměnná *off* nabývá hodnoty `true`, pokud ($s < 0$). Události, na kterou by měla dioda reagovat, pokud bude *off* = `true`, pak tlak bude roven proměnné *a* a tok bude roven 0. V opačném případě je tomu naopak. Tedy tlak nulový a tok roven *s*.

Elastický kompartment

Modeluje objem, který se akumuluje v rozmezí systolického a diastolického tlaku.

Rovnice:

- Derivace objemu je rovna tlaku
- Transmurální tlak = objem * elastance (výstup *Et* z bloku HeartElasticity)
- Rozdíl tlaku vnitřního (na konektoru) a externího tlaku je právě onen transmurální (rozdílový) tlak
- Deklarujte parametr externího tlaku!

Inicializační rovnice:

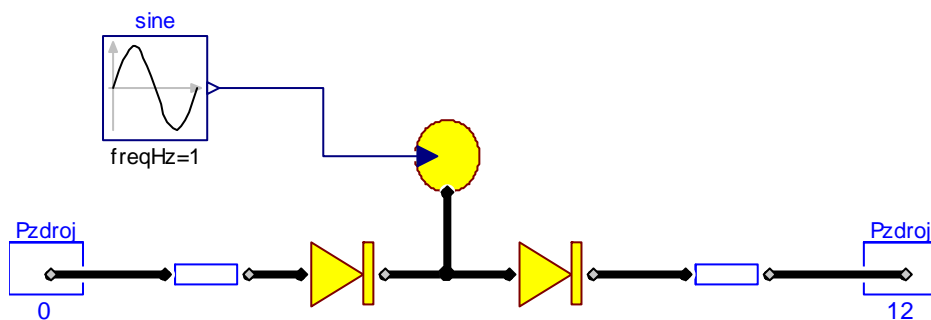
Objem = `initObjem` // nezapomente deklarovat parametr `initObjem`!

Zdroj elasticity

Průběh elasticity je složitá funkce, nyní brutálně zjednodušíme a nahradíme sinusoidou.

Model

Spojte **a-b-c-d-c-b-a**, kde do *d* (kompartment) bude vstupovat řídicí vstup z bloku elasticity.



Úkol na doma1 – do 26. 10.

Otestujte, zda se model chová tak jak má! Zajímá nás časový interval do cca 20s. Teče proud správným směrem? Chlopně se zavírají a otvírají tak jak mají?

Nechová? Je to tím, že tam máme 2 chyby. Najděte je a opravte! Využijte blokovost, otestujte si jednotlivé bloky zvlášť. Ořežte model na minimum a prozkoušejte, že se chová podle očekávání.

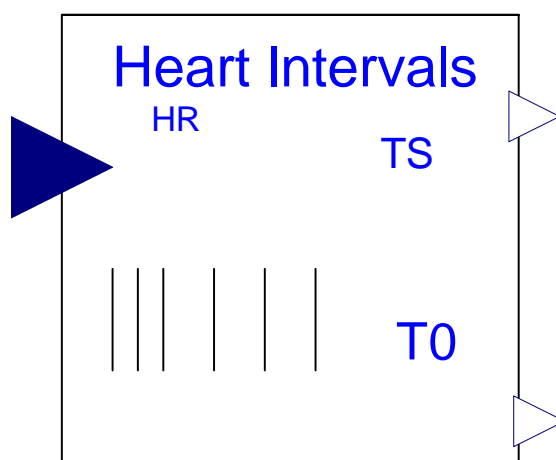
Úkol na doma – do 2. 11.

Vyzkoušíte si:

- Tvorbu vlastního diskrétního konektoru
- Použití when
- Použití if
- Logicky přemýšlet ☺

Nyní se pokusíme zapojit zdroj skutečné elasticity do komory. Zdroj máte předpřipravený v balíčku, nicméně chybí jeho ovládání – před každým beatem je nutné ho naplnit časem počátku následující systoly a její délkou.

Vytvořte si blok, který bude provádět výpočet srdečních intervalů. Bude opatřen jedním vstupem (*RealInput*) a dvěma výstupy (*DiscreteRealOutput*). Např. obrázek níže. HR je spojitá proměnná vyjadřující srdeční frekvenci. Na to musí bloček reagovat a neustále vypočítávat nové intervaly.



Obrázek 1: Ukázka možného vzhledu ikony HeartIntervals

- RealInput použijete ze standard Modelica Library
- DiscreteRealOutput vyrobíte z RealOutput z Modelica Library. Zkopírujte (*duplicate*) si ho do svého projektu a duplikát pojmenujte. V textovém režimu pak v něm doplňte klíčové slovo `discrete` před slovo `output`.
 - (To znamená, že si vyrobíme bloček, který je stejný jako RealOutput, akorát že jeho proměnná není *Real*, ale *discrete Real* a umístíme si ho do našeho package MOS5)
- Legenda:
 - HR "Heart rate (beats/min)"
 - TSyst "delka systoly v sec"
 - T0(start=0) "delka systoly v sec"

Dále je nutné provést deklaraci těchto proměnných:

- `discrete Real TPulsePrev "delka predchoziho srdecního cyklu v s ec";`
- `Boolean b;`
- `XXX TPulse;`

Jakého typu bude TPulse? Doplňte správně místo XXX

Nyní se dostáváme k sekci inicializačních rovnic, která bude mít 3 rovnice:

- `TPulse=60/HR;`
- `TPulsePrev=TPulse;`
- `TSyst=0.3*(TPulse^0.5);`

Poznámka: nastavení počáteční hodnoty v rovnicích v sekci *initial equation* je ekvivalentní použití přímo v deklaraci, např. `discrete Real TPulsePrev(start = TPulse)`.

Než se pustíme do rovnic, tedy sekce *equation*, je důležité pochopit rozdíl mezi `if` a `when`.

Opakování z minulé přednášky ☺

- `If` používáme k rozhodnutí, kterou ze dvou (i více) alternativ programu vykonat. Vždy se ale vykoná jen jedna alternativa či jedna cesta.
- `When` používáme k jednorázovému vykonání části programu nad jeho rámec. Pokud je tedy splněna podmínka, vykoná se „obsah“ v této části.

První rovnice pro booleovskou proměnnou `b` bude vyhodnocovat podmínku:

- `time-pre(T0) je větší než pre(TPulse)`

Její výsledek bude `true` nebo `false`. Proto jsme použili DiscreteRealOutput → abychom mohli získat hodnoty `T0` v předchozím stavu (`pre(T0)`). Pomocí operátoru `pre` rozlišujeme minulou a budoucí hodnotu (stále to jsou rovnice).

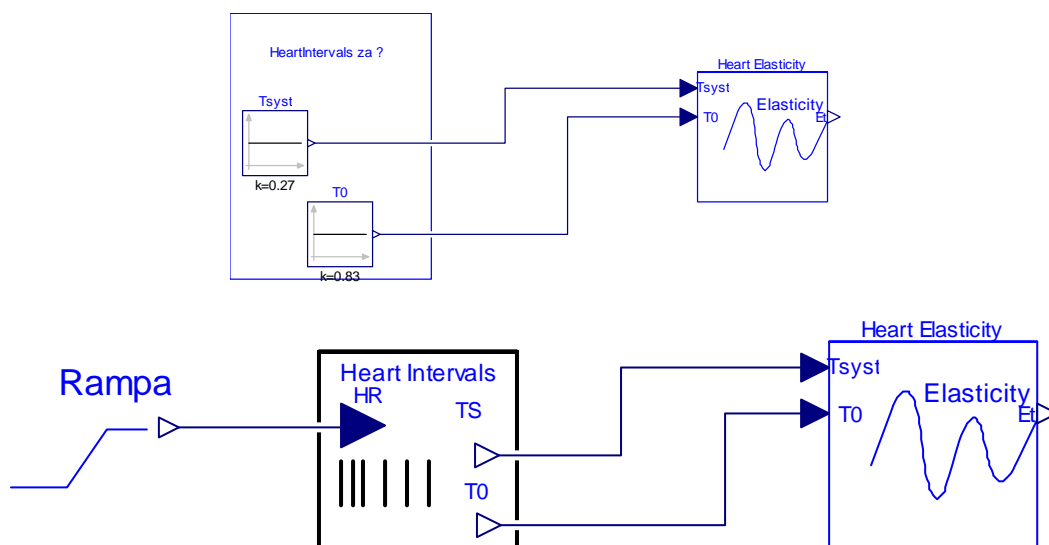
Co vrací `pre(TPulse)`? Odpověď uveďte do zprávy.

Nyní potřebujeme reagovat na událost, kdy dojde k přechodu proměnné **b** z false na true (z 0 na 1). Bude tedy **reagovat** pouze na **náběžnou hranu** proměnné **b** (čili použijete if, nebo when?). Viz číslicová technika a honění jedniček a nul... A pouze při této změně se vykonají tyto rovnice:

- $T0 = \text{time};$
- $TPulse = 60/HR;$
- $TPulsePrev = \text{pre}(TPulse);$
- $TSyst = 0.3 * (\text{pre}(TPulse)^{0.5});$

Použijeme if nebo when? Na co reaguje when? Jak bude vypadat podmínka? Dokážete popsat smysl operátorů `pre()`?

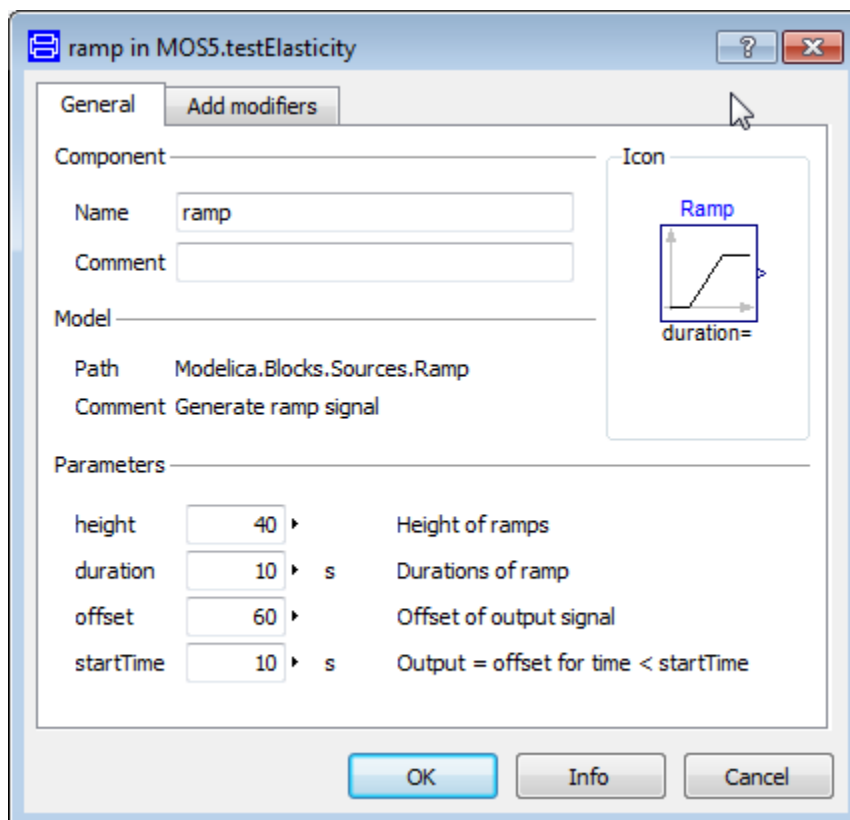
Blok je hotový a nyní jej zapojíme do systému.



Obrázek 2: TestHeartElasticity – před a po

Nezapomeňte na vstup vyrobeného bloku (HR) připojit zdroj rychlosti srdečního tepu. Budeme simulovat klidný tep 60 bpm, který se v 10 vteřině začne zvedat a ve 20 vteřině dosáhne 100 bpm, kde setrvá až do 30s.

Pro otestování můžete zpočátku použít rampu, pak bychom ale rádi viděli vlastní implementaci (zde použijete if, nebo when?) – pro upřesnění povinně.



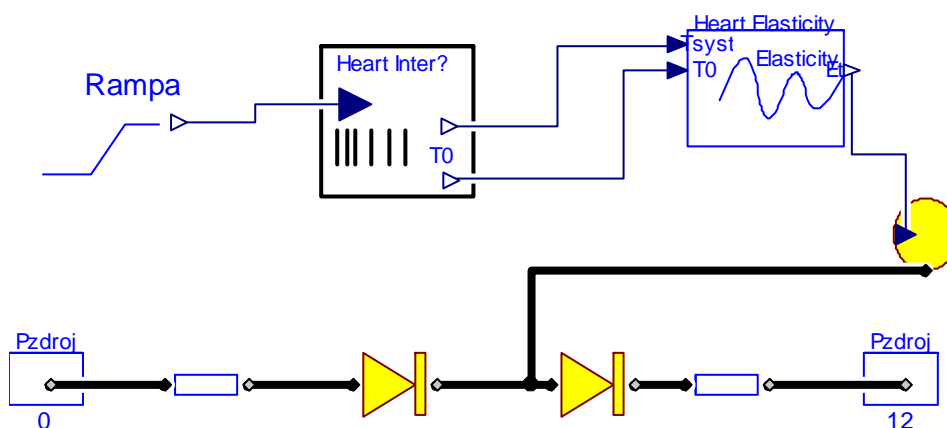
Obrázek 3: Parametry

Blok HR

Pomocí *if a time* si rozdělíte čas do intervalů 0, 10 a 20. V některých blocích bude hodnota konstantní, jinde bude určitou rychlostí (tu si spočtete, ať už předem, nebo dynamicky) stoupat (trik: místo derivace můžete použít znovu *time*, tentokrát ale jako hodnotu). Poté je opět fixována na určitou hodnotu.

Zapojení do obvodu

Nyní to všechno spojte dohromady, jako je na obrázku 4.



Obrázek 4: Celkový obvod

Simulace

- Pozorujte průběhy v Elastický kompartment (d).
 - konkrétně elastanci. Čím je tento průběh omezen?
 - Objem. Okomentujte jeho průběh. Co když nastavím pTransmural na 0?
- Jak se mění proměnné ve zdrojích tlaku - Zdroj tlaku (a)?
- Jakým způsobem dochází k uzavírání a otevírání chlopní?
- Pozorujte vygenerovaný průtok

Pokud to není jasné..

ČTĚTE chybové hlášky – většinou se vám snaží něco sdělit!

Případnou potřebnou nápovědu můžete žádat na jezekfi1@fel.cvut.cz, pokud myslíte, že to bude lépe probrat na konzultaci, obraťte se jak na jezekfi1@fel.cvut.cz, tak i na tomas.krocek@gmail.com

Bonus

- Tentokrát není, čímž zvyšujeme váhu těch minulých.

Bonus bonusu

Pokud to zvládnete celé udělat v OpenModelice, máte u nás +1b ZCELA navíc. V tom případě nepoužívejte packages