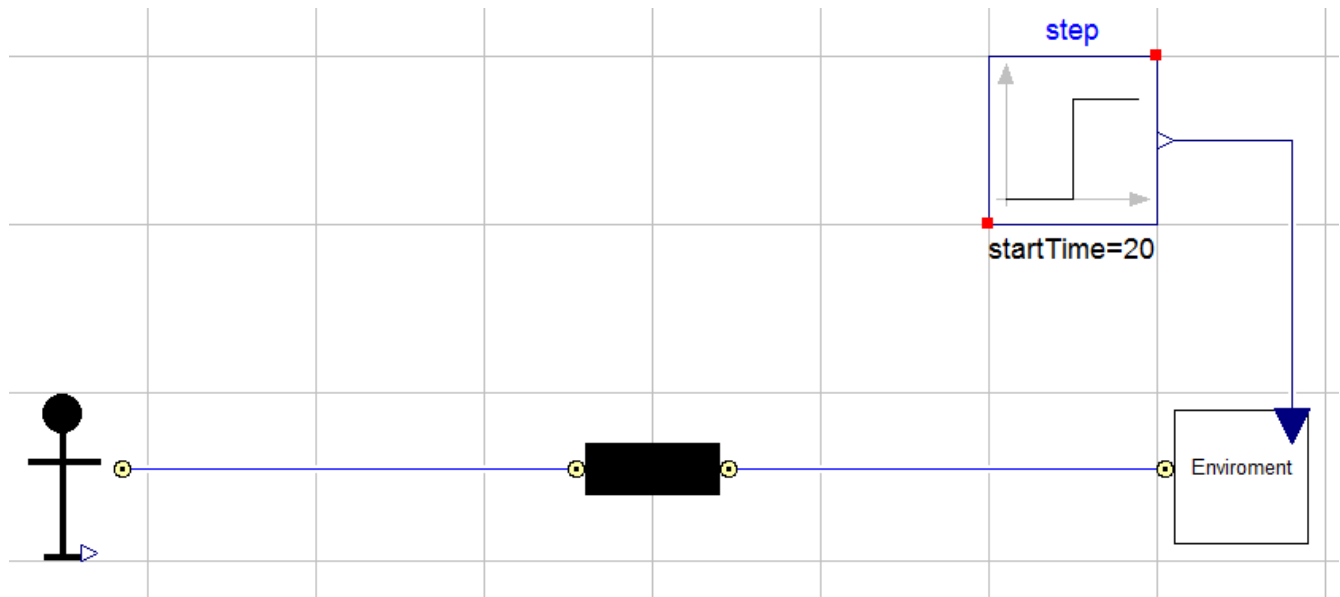


Úloha MOS poslední - do 20. 12. 2011 - Bus connectivity a dědičnost (upraveno)

Zadání

Vyjdeme z minulé úlohy Regulace tělesné teploty, konkrétně **první části bez regulace**, který může vypadat třeba takto.



Obrázek 1: Model jednoduchých ztrát

Zde vytáhnete z bločků *Tělo*, *Kondukcni_ztraty* a *Prostředí* parametry do nového bločku *ControlPanel*.

V bloku *těla* nastavujeme parametry:

- hmotnost m
- Vlastní teplo vzniklé bazálním metabolismem $vlastniTeplo$

a budeme chtít číst

- aktuální teplotu těla (v [°C])

V bloku *kondukcni ztraty* nastavujeme:

- koeficient tepelné vodivosti (konduktanci) k .

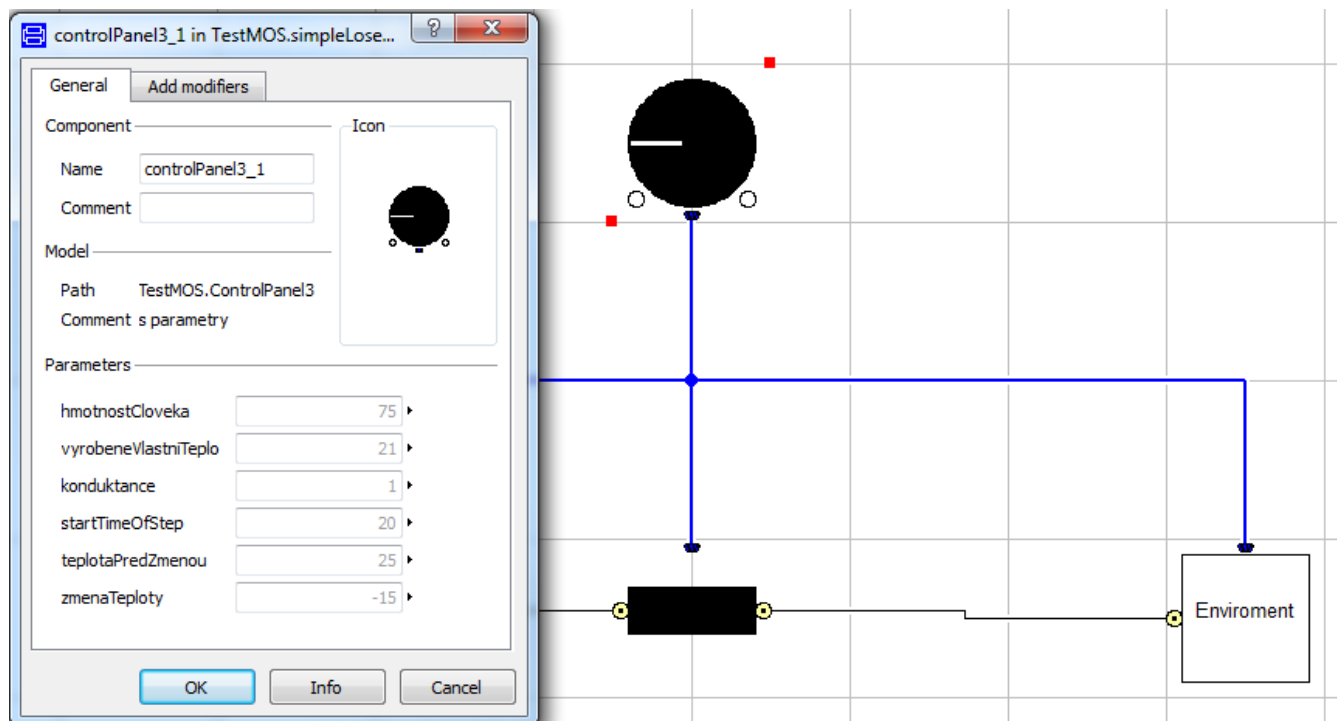
V bloku *prostředí* chceme nastavovat stejné parametry, jako v bloku *Step*. Tedy:

- `offset`,
- `startTime`
- `height` (změna signálu po uplynutí `startTime`). Viz `Modelica.Blocks.Sources.Step`.

Prvek *ControlPanel* tedy obsahuje parametry m , *vlastní teplo*, k , *offset*, *startTime*, *height* a přepočítává aktuální teplotu těla na Kelviny. Propojení *ControlPanelu* s ostatními bločky přes *busConnector* (= expandable connector), který bude dědit svou ikonku z jiného konektoru. **Doporučujeme vyrobit nejdříve *ControlPanel* s parametry m , *vlastní teplo* a k . Otestovat a v případě úspěchu implementovat *Step*, znova ověřit a nakonec implementujete převod teploty.**

Doporučený postup

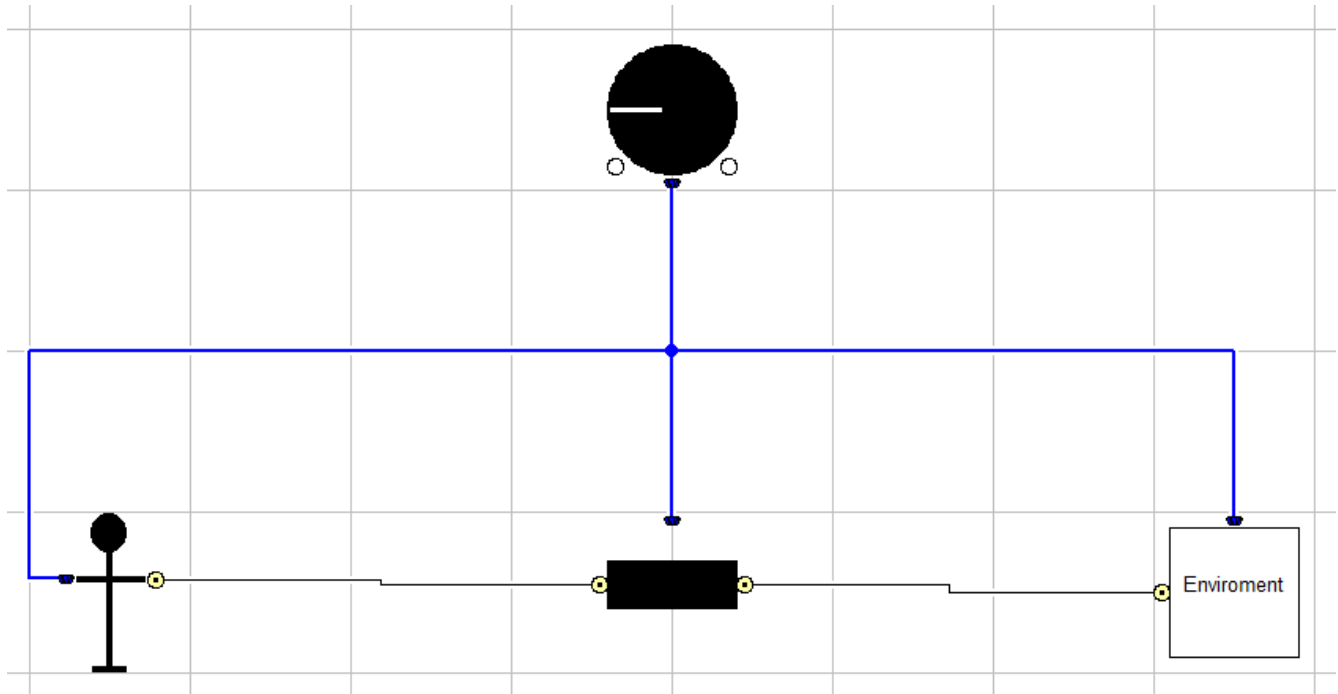
1. Vytvořte si nový konektor, kterému nakreslíte ikonku sběrnice.
2. Vytvořte si nový expandable connector, který zdědí ikonku z konektoru výše.
3. Upravte bločky Tělo a Kondukční_ztráty tak, aby parametry, které nastavujete v *ControlPanelu* byly vyjádřeny jako *RealInput* a ty budou připojeny do *busConnectoru*.
4. Skryjte konektory *RealInput* a *RealOutput* tím, že je definujete jako *protected*.
5. Jednotlivé bločky pak spojte pomocí *busConnectoru*. Akauzální konektory budou zvlášť.
6. Přidejte do celkového modelu bus a propojte ho se vstupem *kondukčních ztrát*.
7. Vytvořte bloček *ControlPanel* a vložte do něj bus. Bloček pak vložte ho do celkového modelu a propojte s jednotlivými *busConnectory*.
8. V bločku *ControlPanel* nastavte proměnné, využít můžete například bloku *Constant* z knihovny *Modelica.Blocks.Sources*. Nebo místo nich použijte *RealOutput* a implementaci jako parametry umožníte jejich nastavování přímo z vlastností.



Obrázek 2: controlPanel s parametry

9. V bločku *ControlPanel* přepočtete teplotu těla z $^{\circ}\text{C}$ na $^{\circ}\text{Kelviny}$. Buď v sekci *algorithm* nebo v sekci *equation*.

Schéma



Obrázek 3: Konečné schéma modelu

Černě jsou akauzální vodiče (vedou teplo a tepelný tok). Modře jsou “řídící” signály z busConnectoru. Na této úrovni stačí graficky propojit bločky. Expandable connector se o dosazení co a kam postará sám.

Bonusová úloha

Stáhněte si model Hummod. Čím je dán tlak v systémových artériích? Popište mechanismus a dohromady alespoň 6 proměnných a parametrů, které jsou za to odpovědné. (+1 BonusBod).

Poznámky

Když mluvíme o bločcích, většinou nemáme na mysli doslovně abyste použili *block*. Naopak, používejte *model*. *Block* znamená, že třída má jasné kauzální proměnné – všechny musí být definované jako *input*, *output*, nebo *protected* (soukromé). Většinou nám to je jedno, proto používejme obecný *model*.

Modelica podporuje vícenásobnou dědičnost, to jest můžeme dědit z libovolného množství tříd najednou. Ne vždy to však je praktické. Dědičnost implementujeme klíčovými slovy *extends* (dědí) a *partial* („chovná“ třída – určena ke zdědění, sama o sobě nemá smysl). Abychom mohli z nějaké třídy (modelu, konektoru...) dědit, nemusí být nutně *partial*.

Dědí se vše – deklarace proměnných, rovnice, dokonce i anotace. Toho využíváme v této úloze.

Expandable konektor (=bus konektor) implementujeme klíčovým slovem *expandable* před *connector*. Konektor nemusí obsahovat žádné proměnné (naopak se to nedoporučuje), zato se dokáže sám rozšířit o připojené konektory.

Expandable konektor můžeme napojovat pouze na libovolné (RealInput, PressureFlow...) další konektory „rovníci“ *connect()*. Nemůžeme z něj ale vytažovat proměnné stylem, jakým jsme byli

zvyklí u klasických konektorů, tj. např. konektor `l.teplota` nám nebude povoleno. Místo toho ho musíme napojit na existující konektor, například `RealInput`, `RealOutput`, nebo jakýkoli jiný. Z toho již můžeme vyčítat a nastavovat proměnné klasicky.

Expandable connector aka Bus konektor, aka též `SignalBus` je určen pro kauzální přenos hodnot, to jest všechny jeho proměnné musí být na jedné straně vstupem a na druhé výstupem. Pro další podrobnosti o expandable konektoru se podívejte do specifikace jazyka `modelica` na www.Modelica.org.

Pokud v části deklarace proměnných napíšeme klíčové slovo *protected*, pak všechny proměnné pod ním až do začátku bloku `equation` (nebo do dalšího klíčového slova *public*, které definuje opak), jsou deklarovány jako `protected`. Není pak možno k nim zvnějšku přistupovat.

Odevzdání

Odevzdávejte do **21. 12. 2011 11:59** do systému.

Nápověda

Krátkou nápovědu si poptáte tradičně na jezekfil@fel.cvut.cz, mezi 14 a 20. 12. se obračejte spíše na tomas.krocek@gmail.com. Někdy bývá lepší si to vysvětlit osobně, neváhejte nás kontaktovat. Konzultace není sprosté slovo!!!