



# Modelování v mechatronice

Robert Grepl

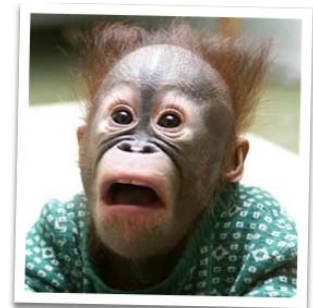
Institute of Solid Mechanics, Mechatronics and Biomechanics  
Faculty of Mechanical Engineering, Brno University of Technology



## V této přednášce...

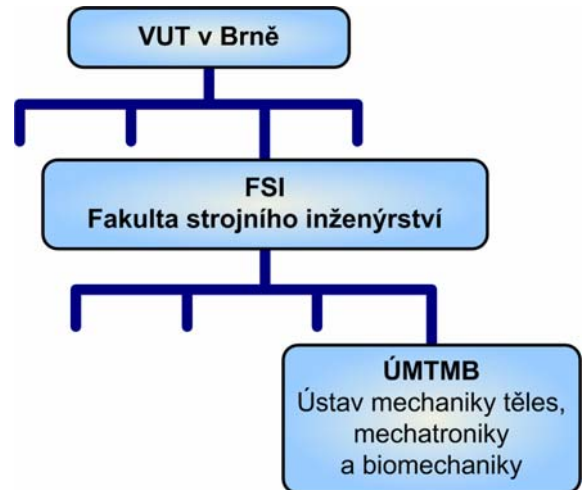
bych rád – a to velmi neuspořádaným až chaotickým způsobem – přiblížil odpovědi na následující otázky:

- Co je to mechatronika?
- Co je to model? (v našem kontextu)
- K čemu se mi může model hodit?
- Co to je Simulink a jak se v tom kliká?
- Co bylo dřív, vejce nebo slepice?



## Kontext: Mechanika těles

- **Statika**
  - most: jaké jsou síly v pilířích?
  - ... ale ... [[Tacoma Bridge](#)]
- **Kinematika**
  - ruka robota
- **Dynamika**
  - síly a momenty vs. kinematika
- **Pružnost – pevnost**
  - „jak se to deformuje a kdy to praskne?“
- **Mechatronika – dynamika + řízení + hardware + ...**



## Kontext: Počítačové modelování

- = řešení úloh Statiky, Kinematiky, Dynamiky,... na počítači
- složitější reálné problémy – nelze řešit „ručně“ obtížně

### K čemu mi to je ?

- pochopení chování systému
- předvídání chování
- návrh řízení

### Nástroje:

- algebraické rovnice
- diferenciální rovnice
  - obyčejné (ODE)
  - parciální (PDE)
- implementované v **software** (Matlab, Simulink, Ansys, ...)

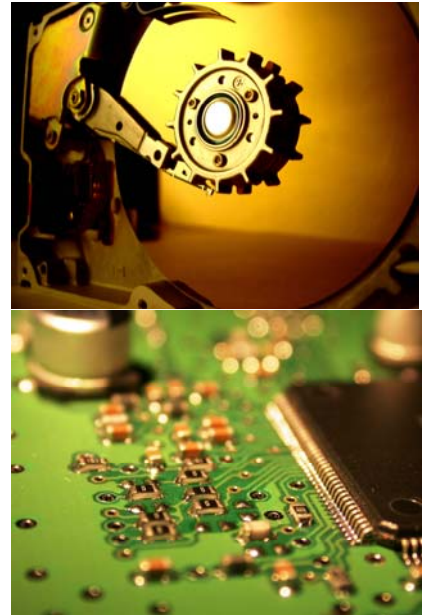
# Co je to „Mechatronika“?

Mechatronika je moderní technický přístup, který integruje:

- přesnou mechaniku
- elektrotechniku, elektroniku
- a počítačové řízení.

Příklady mechatronických systémů:

- ABS v automobilu (mechanika, elektronika, řízeno počítačem)
- další řízené části automobilů, strojů, letadel
- robotické aplikace
- ...

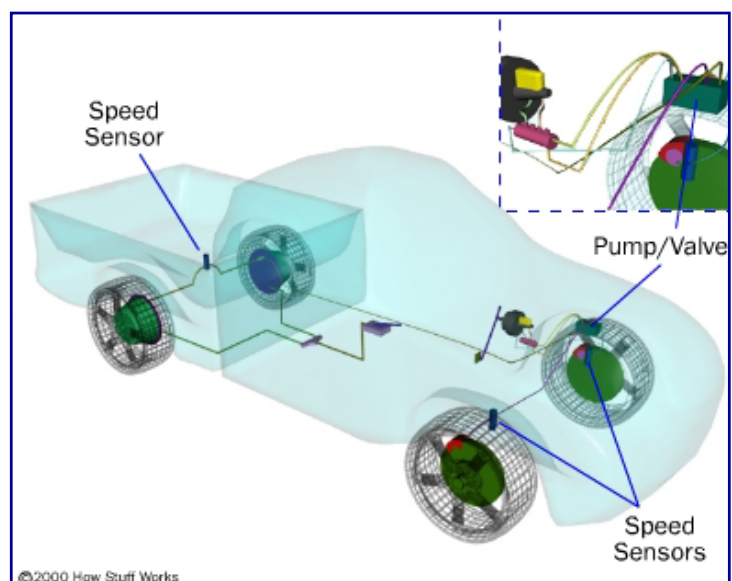


## Mechatronika: principy

Počítačové řízení lepší než lidské?

### ABS – anti block system

- přesná mechanika
- elektronické řízení
- „inteligence“

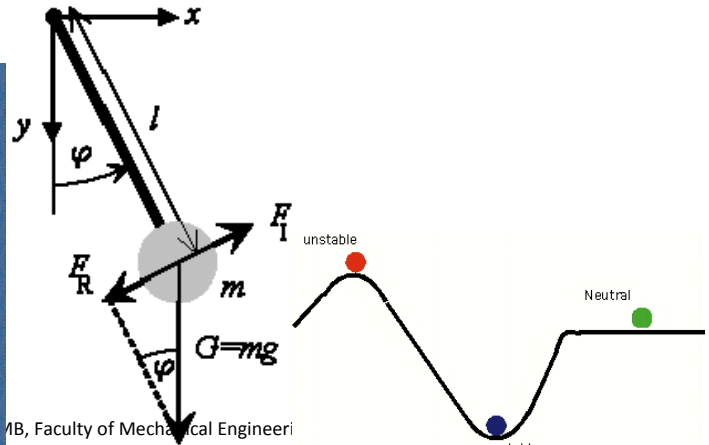
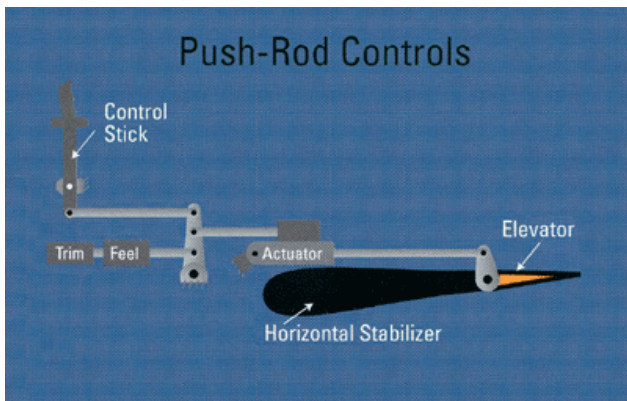


# Mechatronika: principy

## Stabilní nebo nestabilní ?

### Klasická koncepce řízení letadla

- stabilní vs. nestabilní návrh

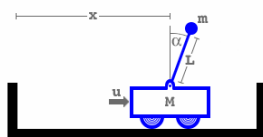
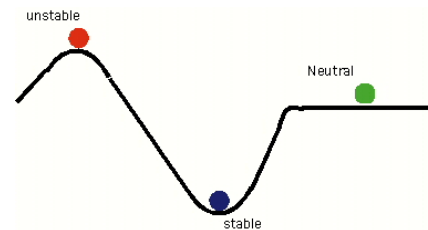


# Mechatronika: principy

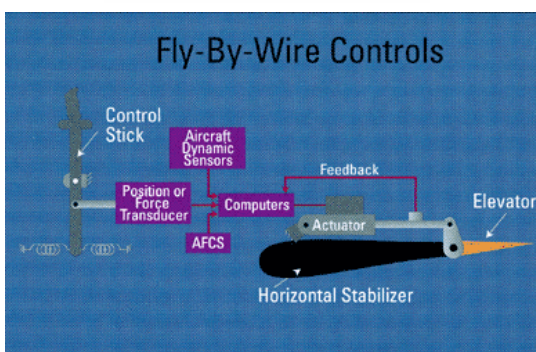
## Stabilní nebo nestabilní ?

### Moderní koncepce řízení letadla

- stabilní vs. **nestabilní** návrh
- fly by wire
- zvýšení manévrovatelnosti díky počítač. řízení

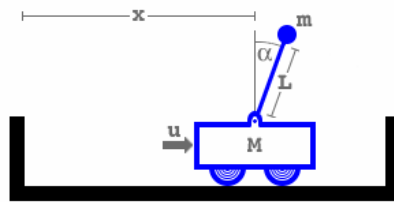


video: [cobra](#)

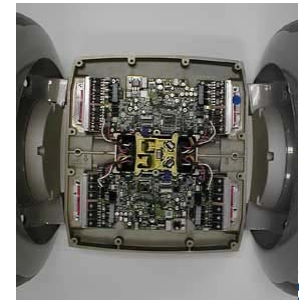


# Mechatronika: příklad - Segway

- Segway, ... a mnoho dalších
- princip řízení – problem „inverted pendulum“
- mechatronika =
  - železo
  - el. motor + elektronika
  - HW a v něm řízení

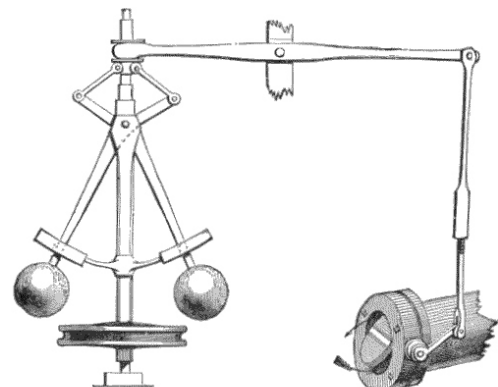
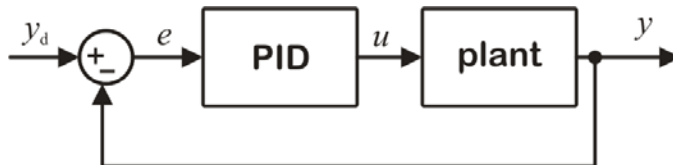


video [Segway](#)

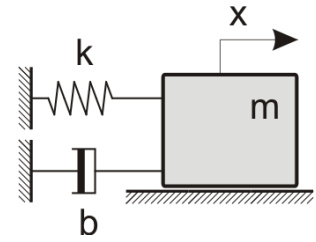


## Základní pojem řízení: zpětná vazba (feedback)

- základní věc, která (mimo kontejnerů) umožňuje náš blahobyt
- funguje všude: příroda, mezilidské vztahy, mechatronika



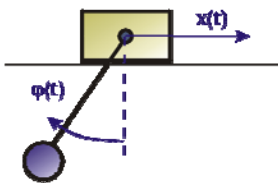
# Odbočka k mocným (nástrojům) tohoto světa



- **Task 1:**  
Sestavte model (rovnici) fyzikálního systému podle obrázku.
- **Task 2:**  
Sestavte model (výpočtový) tohoto systému v prostředí Matlab/Simulink.
- **Task 3:**  
Navrhněte a odsimulujte řízení polohy  $x$  pomocí vnější síly  $u=F$ .

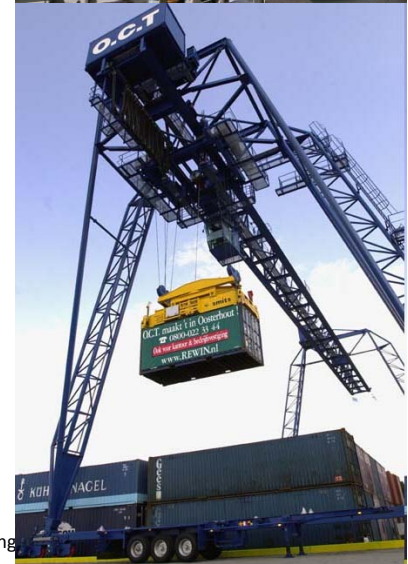
## K čemu může model být – příklad: řízení jeřábu

- motivace  
= transport břemene bez kývání
- jednoduchý dynamický model (konst.  $r$ )



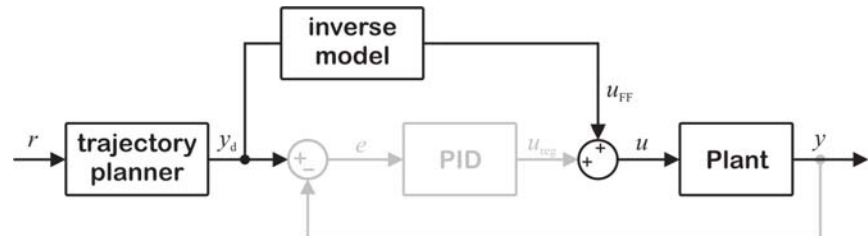
$$\frac{r}{\cos \varphi} \ddot{\varphi} + g \tan \varphi = \ddot{x}$$

- $x$  je vstupem do systému, „kinematické buzení“  
(pojezd vozíku řídíme jako polohu  $x(t)$ )
- požadavek: chceme řídit BEZ sensoru náklonu



## K čemu může model býti – příklad: řízení jeřábu

- základní myšlenka feedforward (oproti feedback)
  - naplánujeme trajektorii
  - použijeme inv. model pro výpočet akce  $u$
  - případné chyby (modelu, vnější poruchy) doladíme zpětnou vazbou



- extrémní varianta: nepoužijeme zpětnou vazbu.
  - Bude to fungovat???

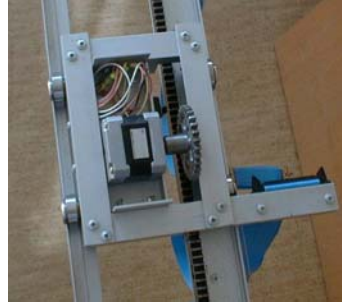
[video Crane](#)

## Model (Simulink) vs. realita

- 1. Mám vytvořený fungující model „soustava + řízení“ v Simulinku.
- 2. Mám otestovány různé detaily jako:
  - sample time
  - omezení akční veličiny
  - ...
- 3. Chci ale řídit reálný stroj.
  - ? vzít rovnice a přepsat to do C
  - ... jinak.

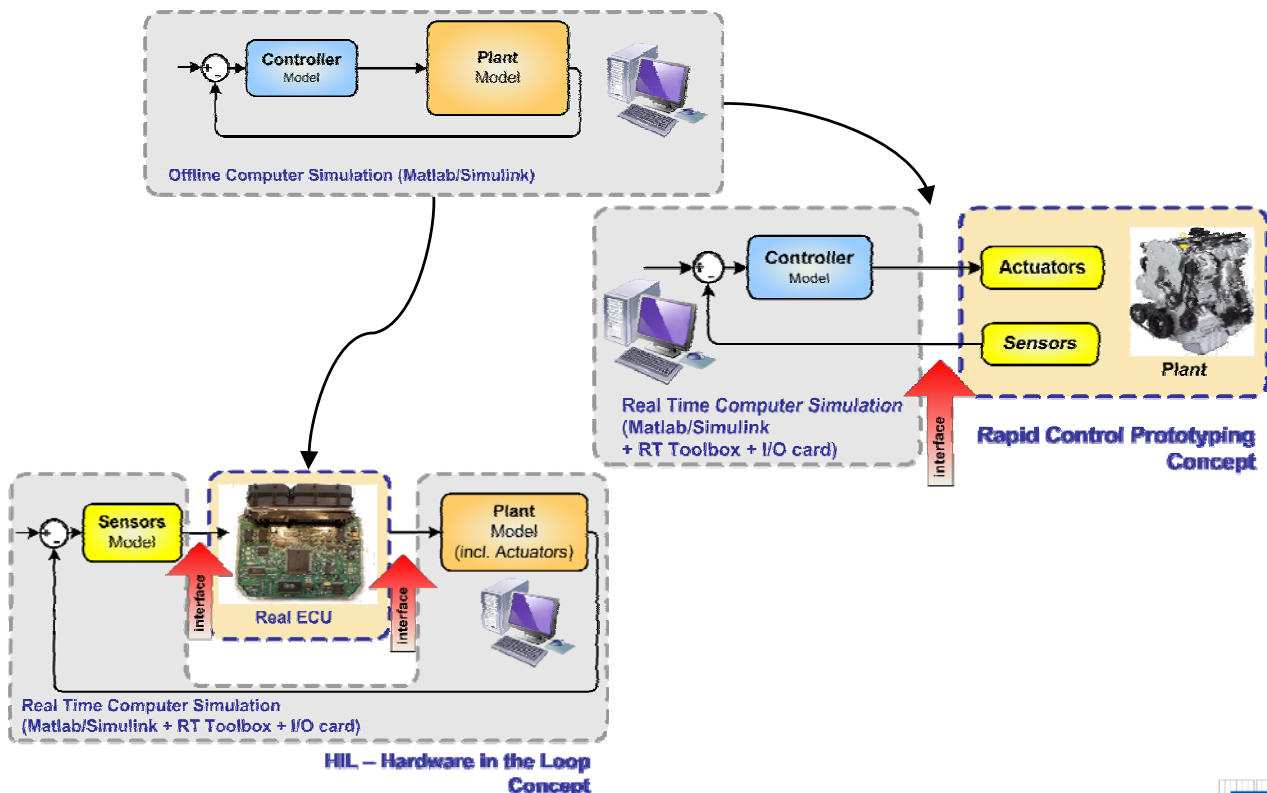
# Model (Simulink) vs. realita

## Historická odbočka: řízení modelu jeřábu



- Cíl: realizovat řízení co nejjednodušeji a nejlevněji.
- HW = staré PC + paralelní port
- SW = MS DOS + kód v Pascalu
  - proč MS DOS?
    - přerušení IRQ0 – obsluha klávesnice, periodické spouštění, nastaveno na 10kHz
    - přímý přístup na paralelní port
    - MS DOS lze provozovat jako RTOS

## Rapid Control Prototyping & Hardware In the Loop





# HW :: MF 624

- I/O

- 8x 14bit A/D (+10V, 1.6(1 ch.)-3.7(8ch.)us conversion time)
- 8x 14bit D/A (+10V, 10mA max., 31us settling time)
- 8x Dout, 8x Din (TTL compatible)
- 4x čítač/časovač (32-bit, použití PWM)
- 4x vstup pro enkodér



Adapter  
Humusoft  
MF624 (auto)

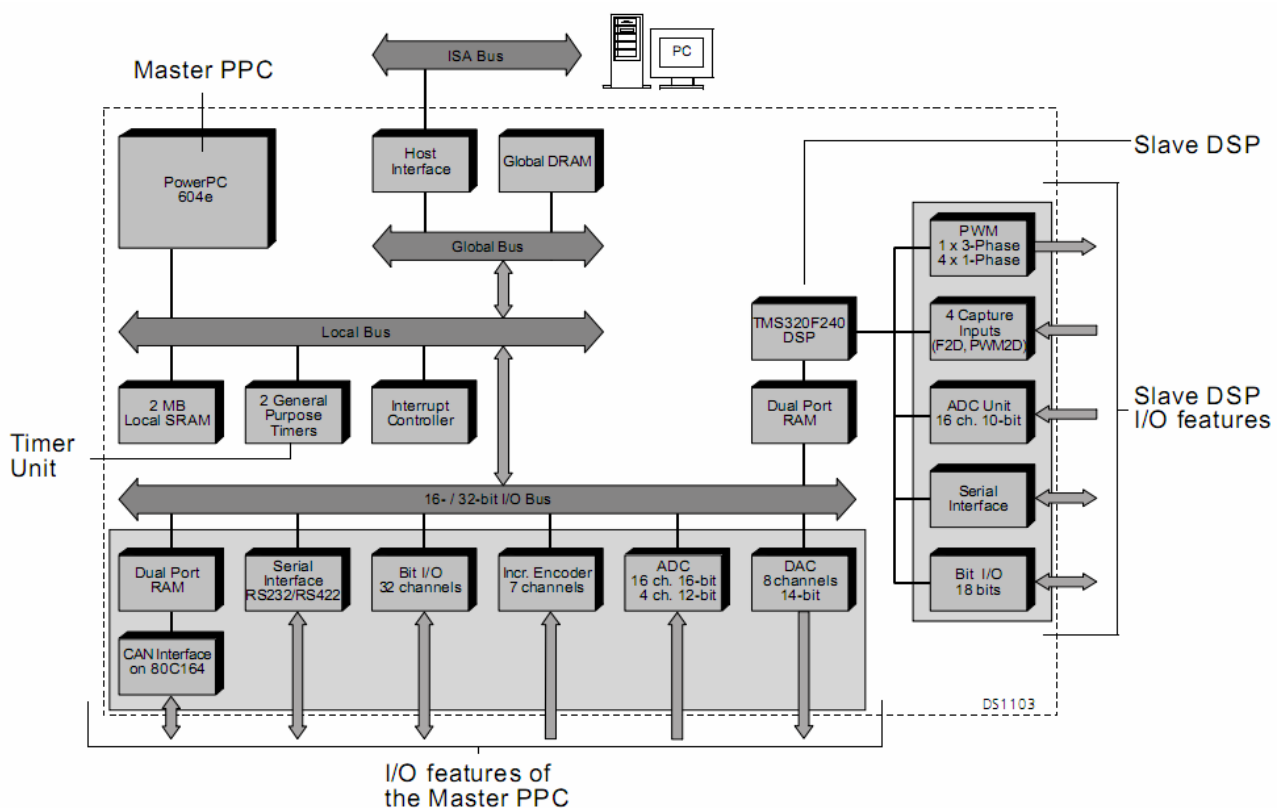
- napětí vyvedeno na kartě

- omezený proudový odběr (do SN = 624100149 resp. nad SN 149)
- +5V (100mA / 250mA max.)
- +12V (nejištěno / 150mA max.)

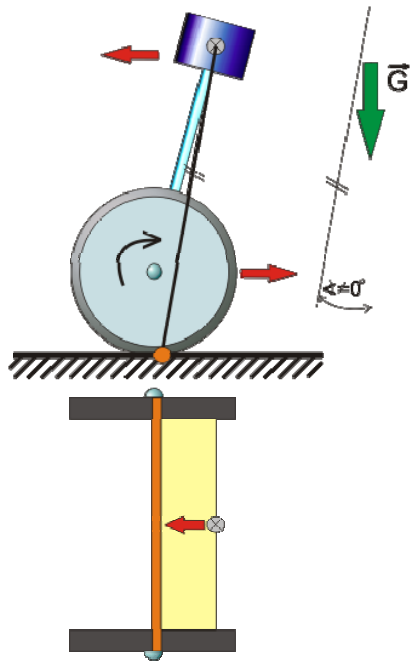
- používá konektory Canon (2 x DB-37 F)



## HW :: dSPACE Single-board DS1103 PPC Controller Board

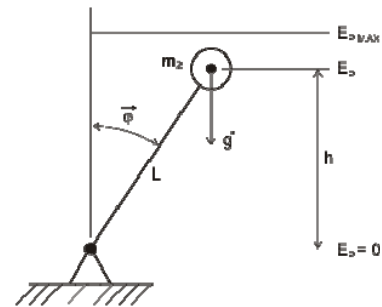
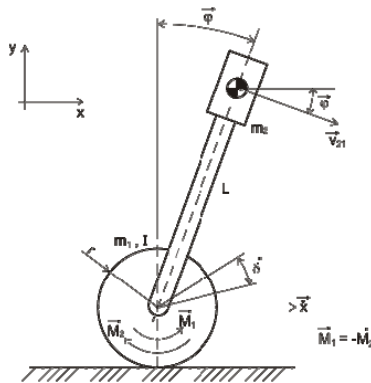


## Příklad použití RTT: Keywatko I



- osa těžiště-kontakt kola s podložkou svírá s vektorem gravitačního zrychlení nenulový úhel
- těžiště mimo oblast stability - nerovnovážený stav
- snaha vrátit těžiště zpět do stabilní oblasti

## Příklad použití RTT: Keywatko I



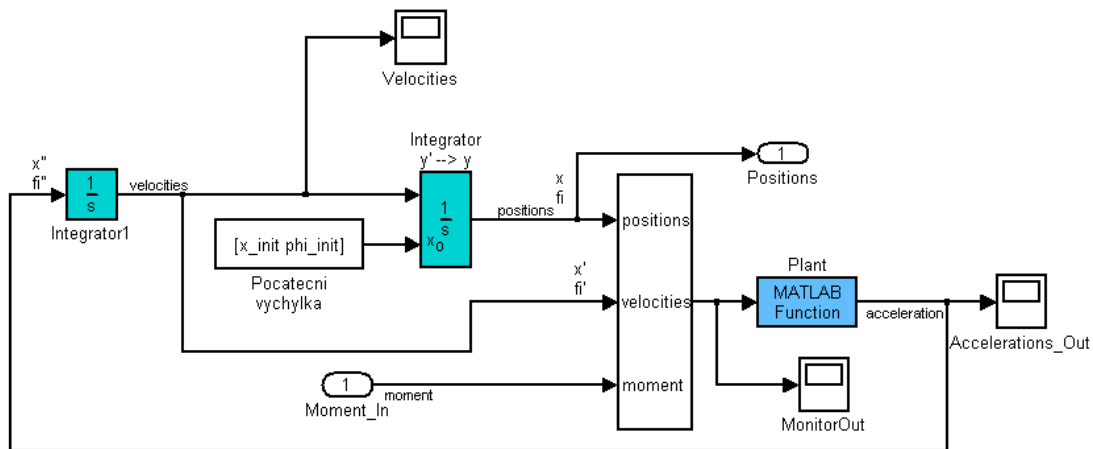
Sestavení pohybových rovnic

$$\ddot{x} \left( 1 + \frac{m_1}{m_2} + \frac{I}{m_2 r^2} \right) + \dot{\varphi} L \cos \varphi = \frac{M}{m_2 r} + L \dot{\varphi}^2 \sin \varphi$$

$$\ddot{x} \cos \varphi + \ddot{\varphi} L = \frac{M}{m_2 L} + g \sin \varphi$$

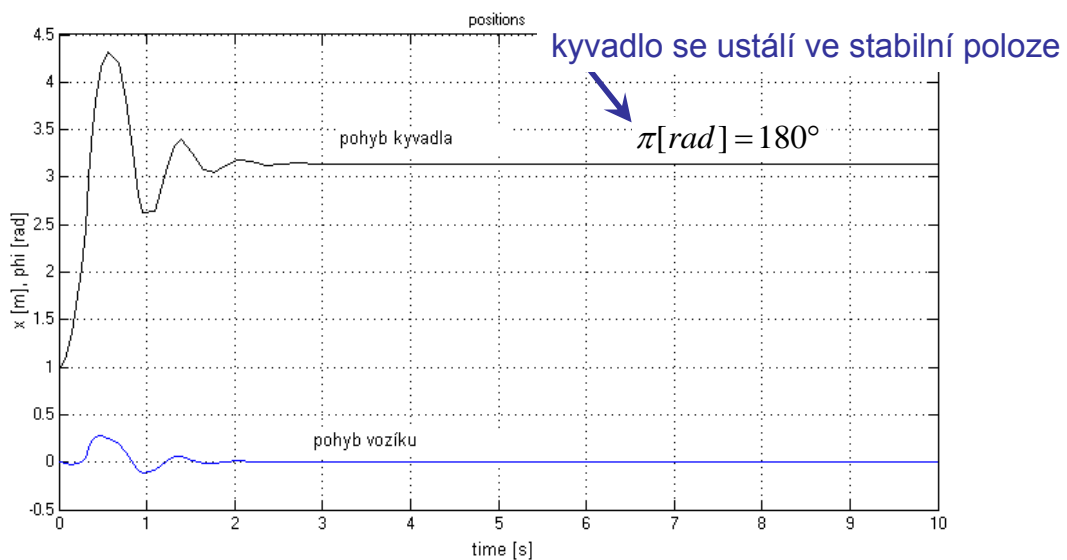
# Příklad použití RTT: Keywatko I

## Sestavení matematického modelu v Simulinku

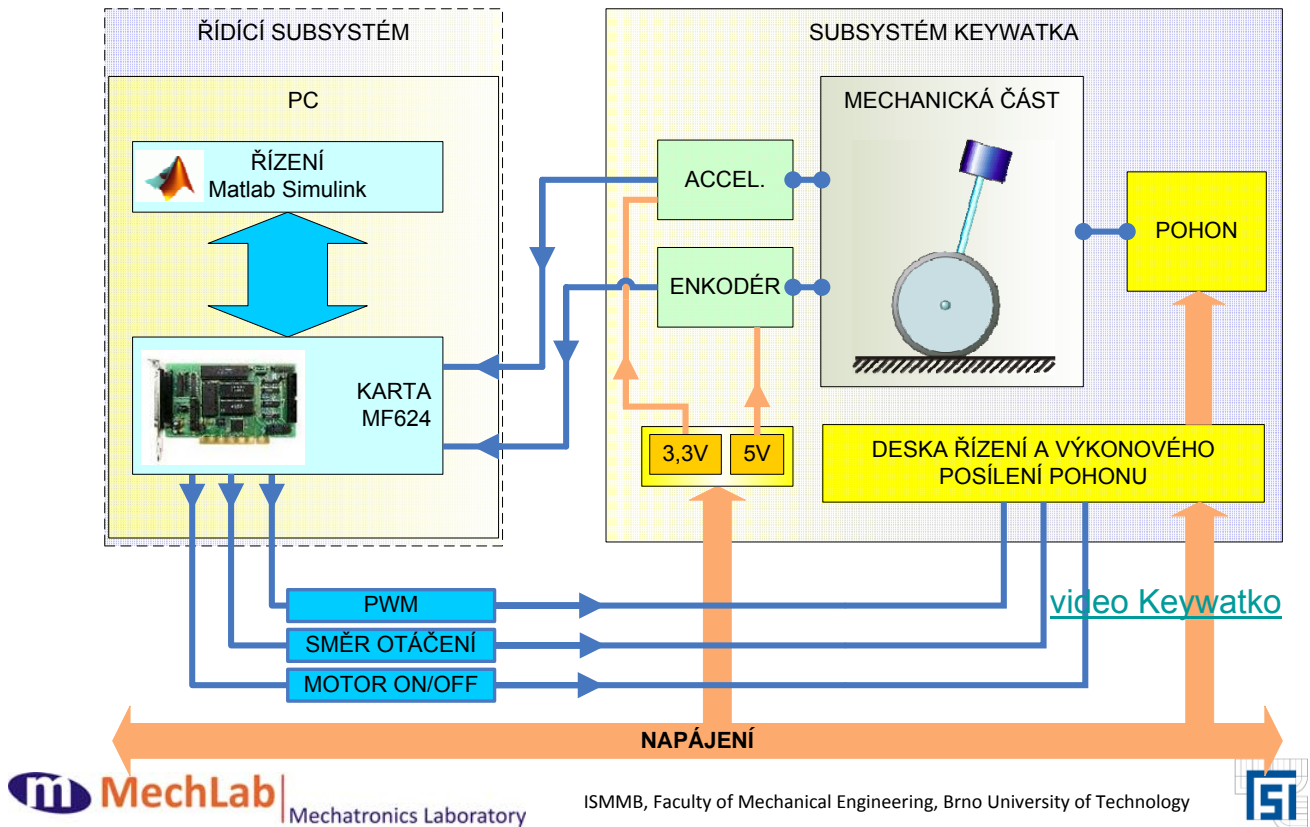


# Příklad použití RTT: Keywatko I

- ověření chování modelu bez zpětné vazby
- výchylka kyvadla není omezena



## KEYWATKO BLOKOVÉ SCHÉMA



## Ne(o)patrné shrnutí:

- Někdy může být výhodné navrhnout systém jako nestabilní a následně ho stabilizovat řízením.
- Některé věci by ani nebylo možné bez řízení realizovat (parní lokomotiva).
- Mechatronika je sloučenina několika různých věcí. K této sloučenině přistupujeme komplexně.
- Model = rovnice se používá při řízení i studiu chování systému.
- Rovnici je nutno nějak řešit – třeba v Simulinku.
- Rapid Control Prototyping – „technologie“ kterak z modelu v počítači (Simulink) rychle přejít k řízení reality.
- Co bylo dřív, vejce nebo slepice?

## A závěrem přednášky...

odpověď na závažný dotaz „co bylo dřív, zda vejce nebo slepice“:

Samostudium viz

<http://www.youtube.com/watch?v=PifReSsx14E>



Děkuji za pozornost.