

Elib library

Úvod do programování e-Pucku

Ondřej Plátek

Matematicko-fyzikální fakulta – Univerzita Karlova

22. března 2011

Obsah

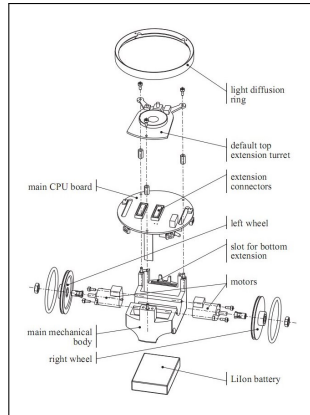
- 1 Proč e-Puck, Proč *Elib* library?
- 2 Použití Epucklib
- 3 TestElib
- 4 Něco složitějšího? Grafika
- 5 Návody tipy

Proč ne vlastní robot?

- Student friendly
- Open software, open hardware
- Komunikace

E-puck

Diferenciální pohon
Processor dsPic → 8kb
RAM
Zajímavé senzory
Pluginy
Baterie:)



Remote control- Je potřeba?

- "Nepříjemné" nahrávání programů
- Ladění, podpora
- Procesní síla
- Standard – Abstrakce

Co za to platíme?

- Nespolehlivost
- Zpoždění
- Nepoužijeme ALU,FFT

- Přesnost
- Musíme "ladit" Bluetooth
- Asynchronní programování

Epucklib BeginEnd funkce

Ukázka kódu

Navázání spojení a poslání příkazu

```
1 //Epuck rur = new Epuck("/dev/rfcomm0","Linux");
2 Epuck rur = new Epuck("Com3","Windows");
3 IAsyncResult ar = ada.BeginMotors(-0.5,0.5, 0.2, ←
   null, null);
4 Console.WriteLine("hned");
5 ada.EndFtion(ar);
6 Console.WriteLine("Blokujici EndFtion");
```

Epucklib callbacky

Ukázka kódu

Co je callback? Co se mu předává?

```
1  if (light[2] < light[5]) {
2      Console.WriteLine("turn around right {0}", ←
        diff_lr);
3      ar = ada.BeginMotors(speed, 0, to, ←
        recGotoLight, ada);
4  } else {
5      //zatocime na druhou stranu ...
```


TestElib vzorové řešení

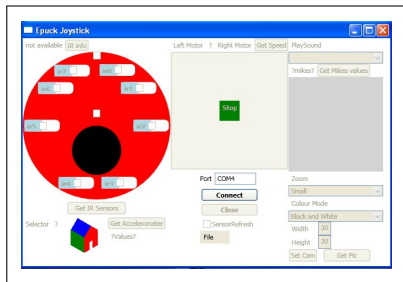
- Rozcestník
- Komentovaný
- Dá se kopírovat

Výsledek

Zde začít a neskončit

Epuck Joystick

Asynchronní volání
Nezamrzá
Dispatcher třída
Wpf
Jen pro Windows:(



Windows vs Unix, .Net vs Mono

- .Net 2.0 a Mono 2.0
- Mono Solution OK
- Microsoft Visual Studio 2005, 2008, 2010 Downgrade
 - *MicrosoftVisualStudioSolutionFile, FormatVersion11.00*
#VisualStudio2010
 - *MicrosoftVisualStudioSolutionFile, FormatVersion10.00*
#VisualStudio2008

Help, dotazy, zdroje

Stránky *Elib* library

<https://code.google.com/p/epuck-lib/>

Kontaktní email

ondrej.platek na gmailu