

A3B35ARI - zadání úlohy LEGO Segway

Úvod

Cílem této semestrální úlohy je řízení LEGO robota, v tomto případě ve tvaru dvoukolého vozítka SEGWAY (http://en.wikipedia.org/wiki/Segway_PT), jehož matematický model odpovídá inverznímu kyvadlu.

Pro zjednodušení práce budeme pracovat s již předpřipravenými zdrojovými kódy od Yorihiya Yamamota¹. Ty umožňují velmi efektivní návrh a ověření regulátoru v prostředí Simulinku a následné nahrání řídicího algoritmu do LEGO robota. Veškerá dokumentace je sepsána v [NXTway-GS Model-Based Design.pdf](#) a v [NXTway-GS Building Instructions.pdf](#).

Předpřipravené nástroje ovšem vyžadují mnohé toolboxy, které z licenčních důvodů nemůžeme používat přímo na strojích v laboratořích K23 a K26 nebo na stroji s Matlabem s ČVUT licencí. Proto jsme přistoupili k drobným změnám v úloze, které, alespoň doufáme, nezkomplikují jednoduchost a přímočarost celého řešení.

Cíl práce

1. Sestavte robota podle níže uvedeného návodu a oživte ho poskytnutým testovacím regulátorem. [15%]
2. Navrhněte zpětnovazební vstupně-výstupní regulátor stabilizující robota ve vertikální poloze. Jako výstup systému použijte jeden ze stavů, který bude pro tento úkol vhodný. [25 %]
3. Pomocí metody přímého umístování pólů navrhněte stavový zpětnovazební regulátor, jenž bude stabilizovat jak úhel naklonění robota, tak jeho polohu (v podstatě všechny stavy). [25 %]
4. Rozšiřte bod 3 o sledování reference na úhlovou rychlost koleček. Cílem je bude ujet za 15s co největší vzdálenost, poté zastavit a zůstat ve stabilizované svislé poloze. [35 %]

Do stanoveného termínu nahrajte na moodle podklady, ze kterých bude jasné, jak jste postupovali, jak jste navrhovali který regulátor a k čemu jste dospěli.

Sestavení robota

Detailní návod na sestavení robota je v souboru [NXTway-GS Building Instructions.pdf](#). V tomto bodě je volitelná poloha a spojení kol – je možnost mít široký/úzký rozchod kol a také (ne)spojenou jejich nápravu. Design robota či různé doplňky můžete samozřejmě upravit podle týmové preference :-), pozor

¹ <http://www.mathworks.com/matlabcentral/fileexchange/19147>

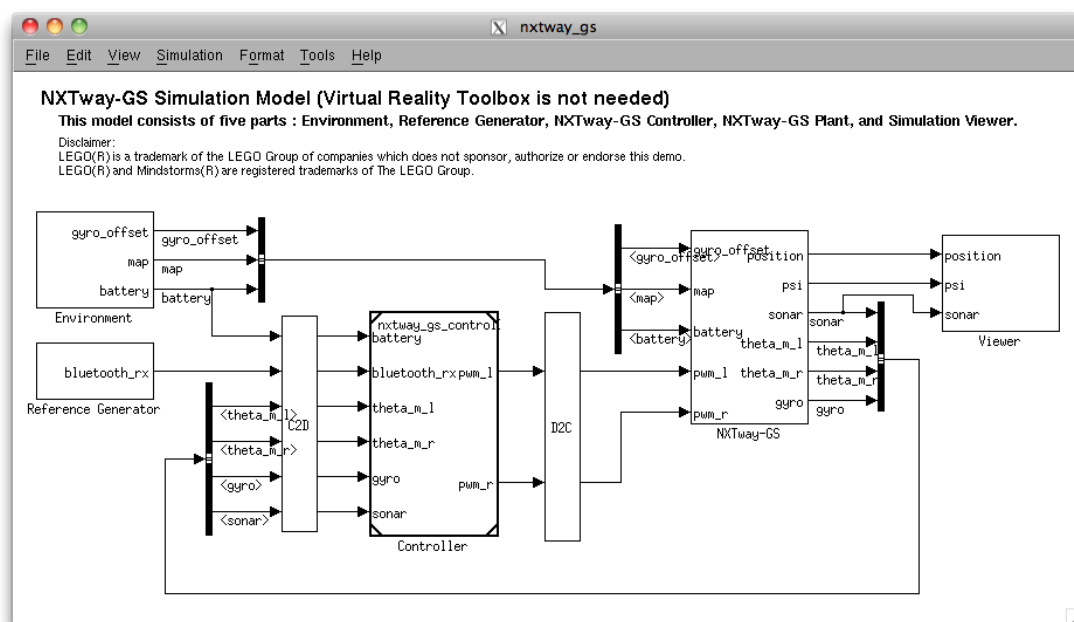
však na to, že větší změnou můžete změnit dynamické vlastnosti robota a nemusí pak odpovídat model.

Pro odzkoušení, zdali je robot dobře sestaven a vše funguje jak má, lze využít aplikaci [testovací nxtway_app.rxe](#), jenž robota stabilizuje ve svislé poloze na místě. Jak ji do robota nahrát je popsáno v podkapitole „Nahrání aplikace do robota“.

Spuštění modelu

Zdrojové kódy pro Matlab a Simulink jsou připraveny v adresáři „nxtway_gs“. Seznam souborů je sepsán v [NXTway-GS Model-Based Design.pdf](#), adresář je doplněn o soubor [compiler_init.m](#), který je nutný pro kompilaci (bude upřesněno níže). Před samotným spuštěním schémat je nutno přidat dodatečné simulinkové knihovny do cesty Matlabu. Tyto knihovny jsou v adresáři „environment“, který stačí přidat do cesty bez podadresářů.

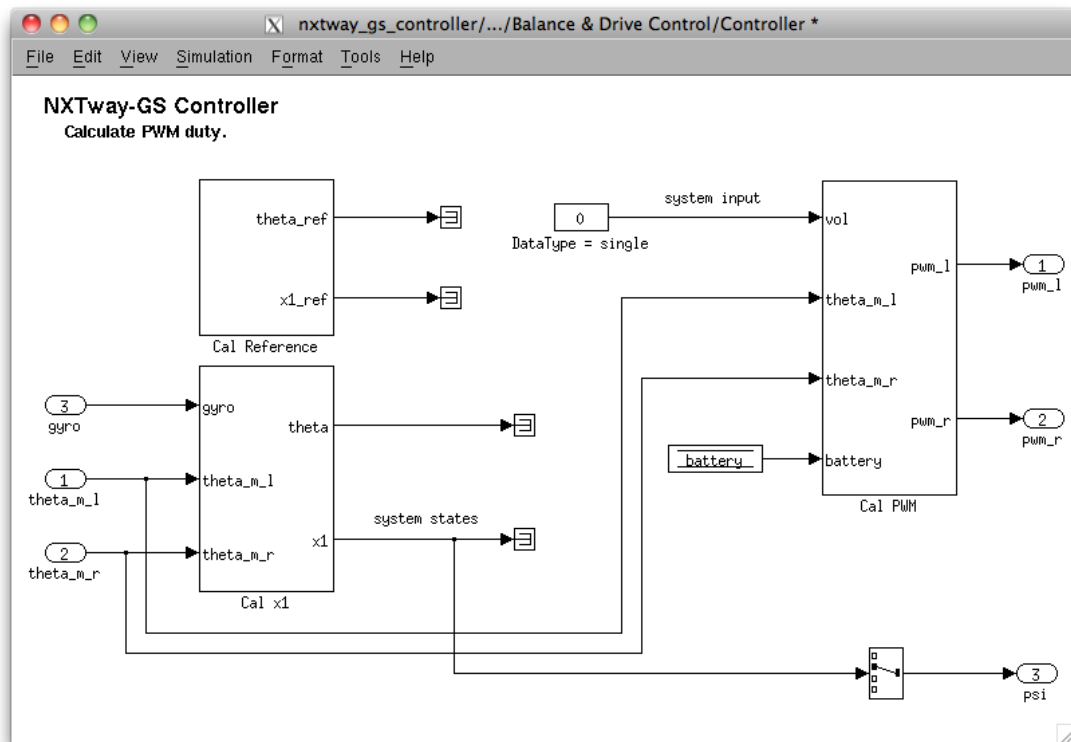
Adresář „nxtway_gs“ obsahuje několik simulinkových schémat. Hlavním schématem je [nxtway_gs.mdl](#) (Obr. 1). Obsahuje celé regulační schéma (systém+regulátor ve smyčce) a tudíž je toto schéma vhodné zejména pro ladění navrženého regulátoru.



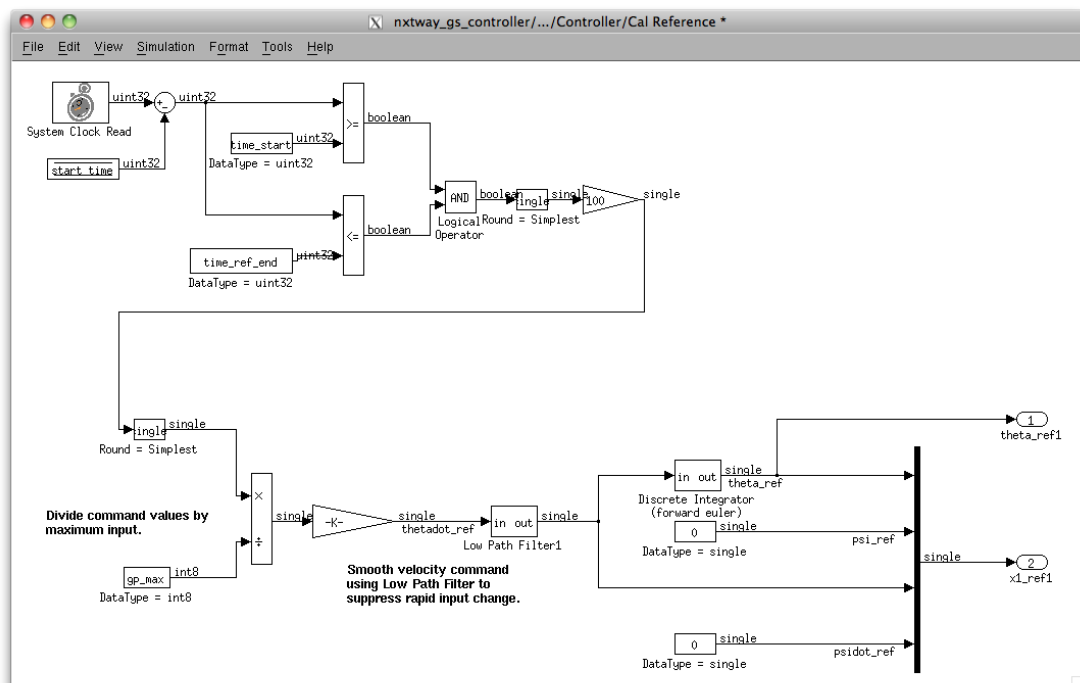
Obr. 1: [nxtway_gs.mdl](#)

Vámi vytvořený regulátor zabudujte do subsystému *Controller/ nxtway_app/ Balance & Driver Control/Balance & Driver Control/Controller* (Obr. 2). Blok *Cal Reference* obsahuje generátor reference na rychlost robota (doba nenulové reference je dána proměnnou *time_ref_end*), blok *Cal x1* je pozorovatel stavů systému a konečně výstup regulátoru se posílá do bloku *Cal PWM*. Veličina *x1* je vektor stavů (postupně úhel otočení koleček *theta*, úhel náklonu robota *psi*, úhlová rychlost koleček *thetadot*, úhlová rychlost robota *psidot*), *vol* je napětí do motorů, *theta_m_l,r* jsou úhly otočení levého resp. pravého kolečka, *gyro* je

aktuální hodnota úhlové rychlosti robota. Blok *Cal_PWM* přepočítává vstupní napětí na PWM signál pro obě kolečka.



Obr. 2: Subsystém, kam je potřeba umístit regulátor



Obr. 3: Cal Reference

Blok generování reference (*Cal Reference*, Obr. 3) generuje po dobu 1s až $\text{time_ref_end}/1000$ s referenci na rychlost a na polohu robota. Jakmile je čas větší než time_start , je požadavek na maximální rychlost, jenž se také přepočítává na polohu, ostatní stavové veličiny mají referenci na nulu. Jakmile je čas větší než time_ref_end , pak je reference na rychlost opět nulová, reference na polohu už se nemění. Rychlé změny požadavku na rychlost je záhodno předfiltrovat (input shaping), jinak může dojít prudkou změnou k destabilizaci robota. Filtr si lze nastavit pomocí konstanty a_r .

Při implementaci regulátoru je nutno dbát na to, aby **regulátor byl diskrétní v čase**, lze například použít blok *Discrete Transfer Fcn*. Dále je potřeba dbát na správné datové typy, které lze v případě potřeby konvertovat pomocí bloku *Data type conversion*.

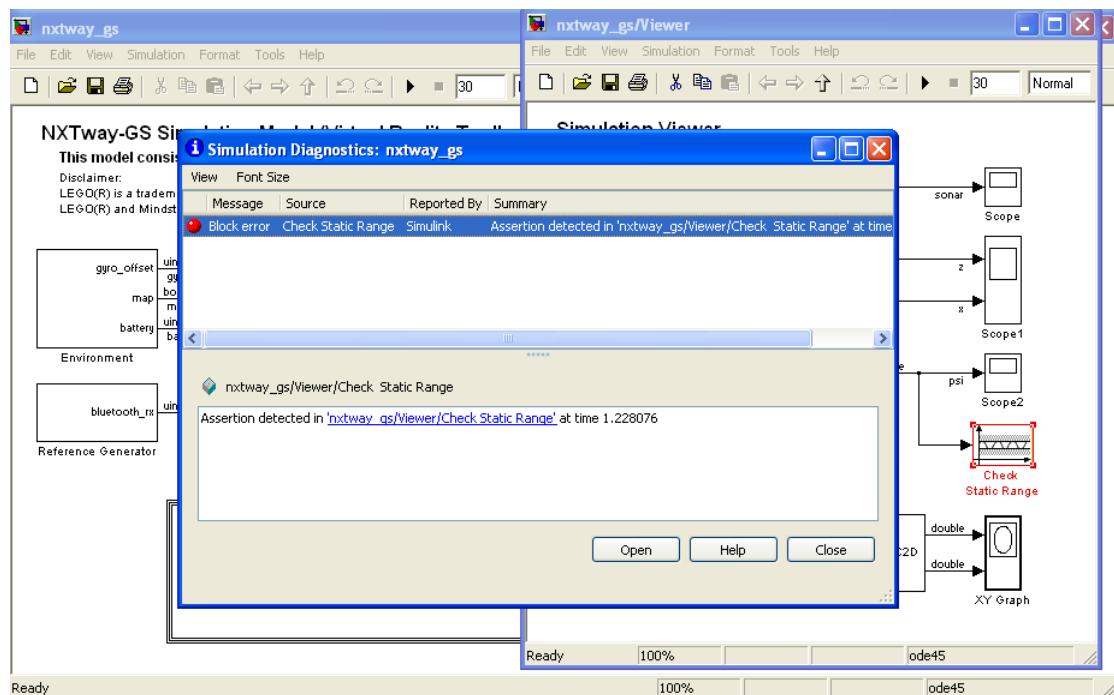
V simulinkovém modelu je obsažen blok *Check static range*, který kontroluje, zda se náklon robota nedostal na víc než 15° . V takovém případě je simulace zastavena a vypsána chybová hláška na Obr. 4 (robot nenávratně padá, navíc přestane platit linearizovaný simulační model). Toto nastane například při spuštění čerstvě staženého modelu, protože poskytnutý simulinkový model neobsahuje regulátor (ten tam musíte nejprve doplnit).

Konstanty a parametry regulátoru doporučujeme zadávat do souboru [param_controller.m](#), jenž se volá automaticky při otevření simulinkových schémat. Vzorkovací perioda pro řízení je specifikována také v [param_controller.m](#) a je definována pomocí proměnné $ts1$. V [param_plant.m](#) je definován model systému - jsou to matice A, B, C, D se stavy popsány výše a vstupem je napětí do motorů.

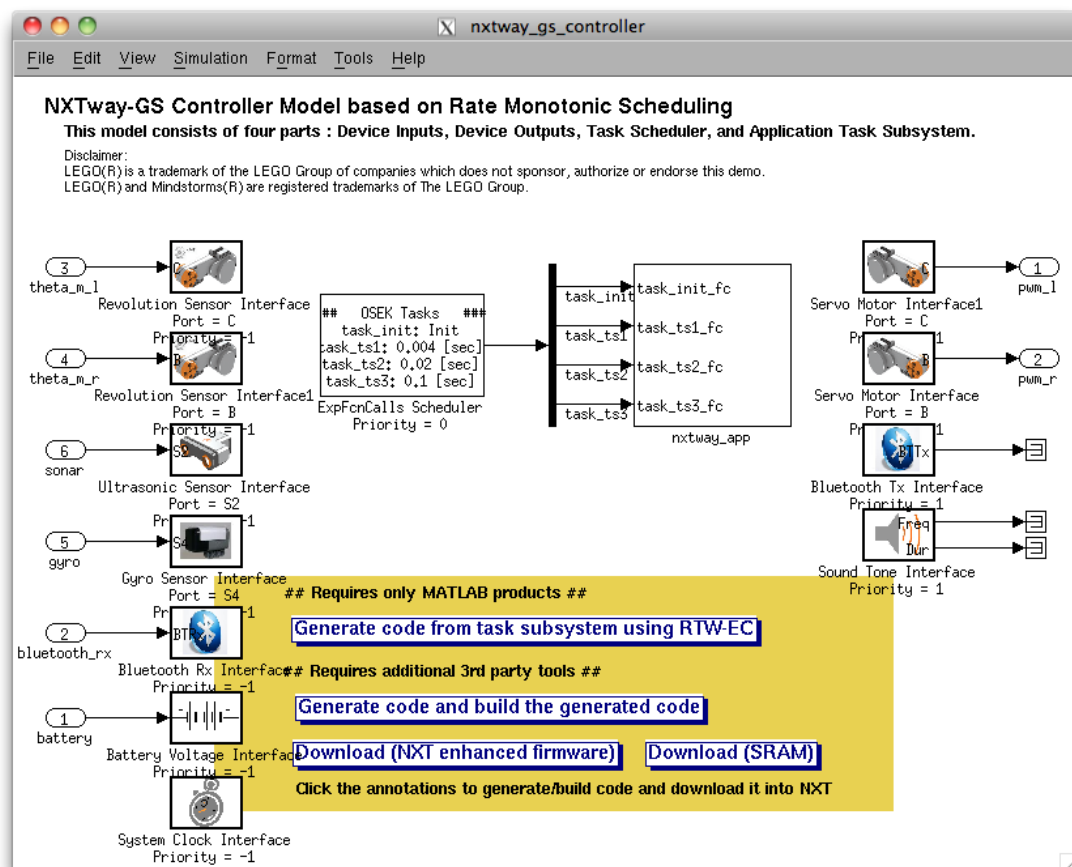
Poznamenejme ještě, že pro hladkou kompilaci je potřeba mít všechny vstupy a výstupy simulinkových bloků zapojeny (minimálně do bloku Terminator).

Simulace na [nxtway_gs.mdl](#) sice projde s několika varováními, ale pak samotná kompilace je na toto velmi citlivá.

Schéma [nxtway_gs_controller.mdl](#), představuje propojení regulátoru s hardwarovými vstupy a výstupy (Obr. 5).



Obr. 4: Detekce „upadnutí“ robota



Obr. 5:nxtway_gs_controller.mdl

Nahrání aplikace do robota aneb simulink2lego

Pokud máte odladěný regulátor připravený k nahrání do robota, je k tomu potřeba nejprve zkompilevat hlavní simulinkové schéma a veškeré zdrojové kódy do programu, který už lego umí spouštět (přípona .rx). Dalším krokem je poté nahrání aplikace do robota.

Kompilace zdrojových kódů

Kompilace probíhá pomocí serverového řešení (kvůli výše zmíněným licenčním podmínkám) a je vykonávána formou nahrání zazipovaného projektu do složky x:\share\lego\in\ (přístupná při přihlášení na počítačích na Karlově náměstí nebo vzdáleně přes webové rozhraní²). Do této složky mají všichni studenti pouze právo zapisovat, nikoliv číst.

Archiv musí obsahovat m-file pojmenovaný *compiler_init.m*, který určuje hlavní simulinkové schéma ke kompilaci (pokud *compiler_init.m* není obsažen v archivu, bude to vypsáno jako chyba v souboru x:\share\lego\out\default_log.html). Soubor by měl obsahovat následující:

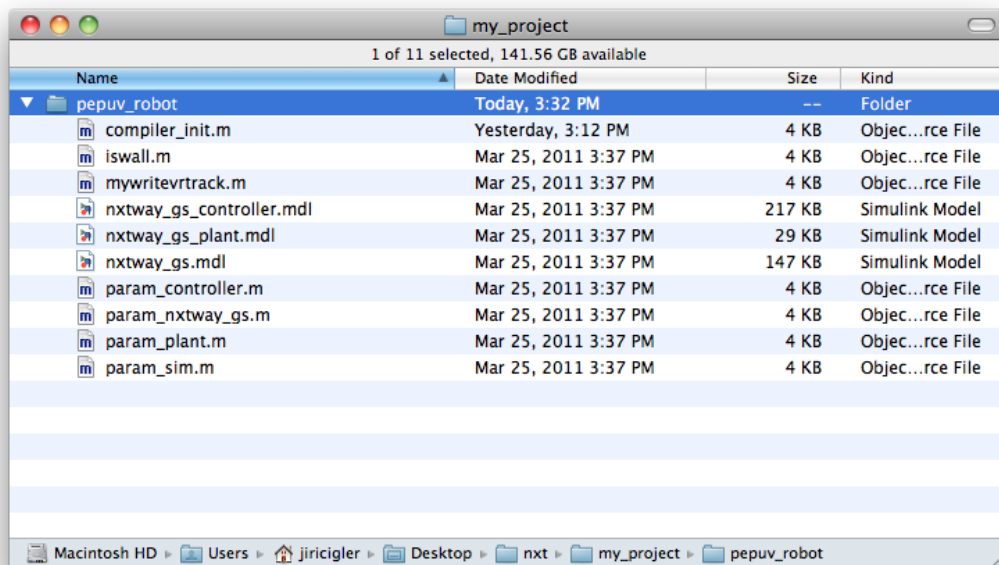
```
mainSimulink='nxtway_gs_controller.mdl';  
key='pepuv_robot';
```

mainSimulink obsahuje název simulinkového modelu, který bude kompilován. Defaultně to je [nxtway_gs_controller.mdl](#), nicméně může to být libovolný model, který obsahuje propojení navrženého regulátoru s hardwarovými vstupy a výstupy robota. Další proměnná *key* určuje prefix v názvu výstupů z kompilace.

Vzor obsahu zip-souboru potřebného ke kompilaci je na Obr. 6.

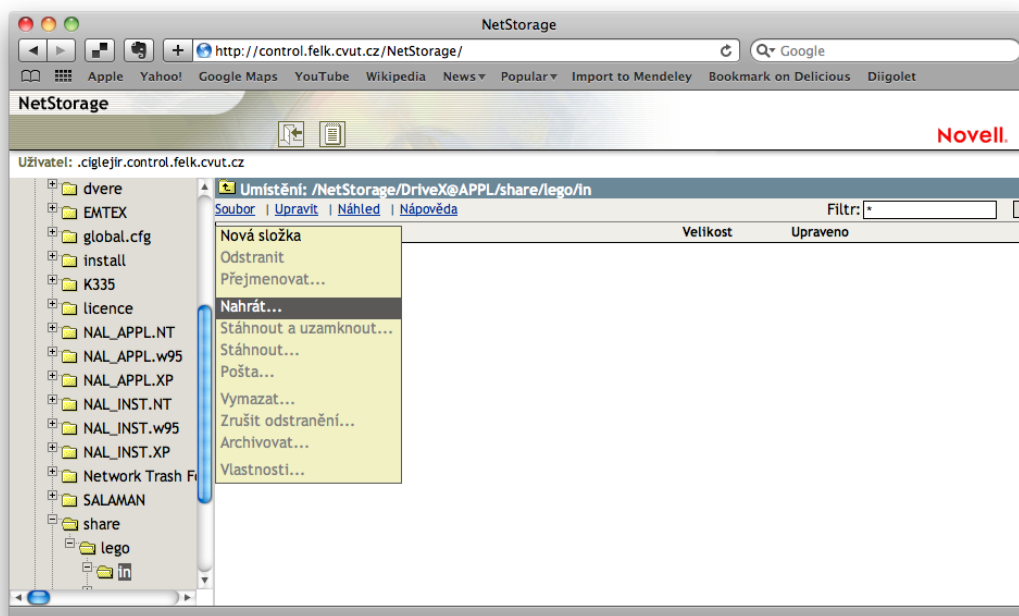
²Přístup k této složce je popsán na:

http://support.dce.felk.cvut.cz/mediawiki/index.php?title=P%C5%99%C3%ADstup_k_s%C3%A4%C5%A5ov%C3%BDm_disk%C5%AFm_na_serveru_Novell&direction=next&oldid=12280

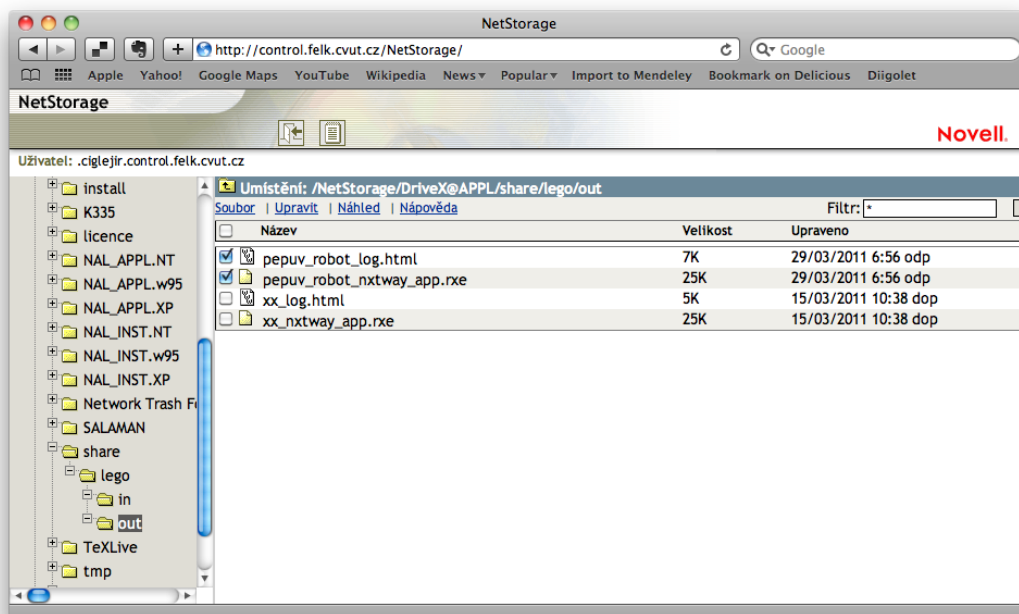


Obr. 6: Ukázka kompletního projektu, jenž se zazipuje a nahraje na server pro následnou kompilaci

Server během několika vteřin po uploadu provede kompilaci modelu, vytvoří záznam z kompilace, spustitelný soubor pro robota a uloží tyto soubory do složky x:\share\lego\out\ na stejném serveru. Soubory v této složce jsou čitelné pro všechny uživatele a jsou průběžně promazávány (momentálně je ve výstupní složce pouze posledních 15 kompilací).



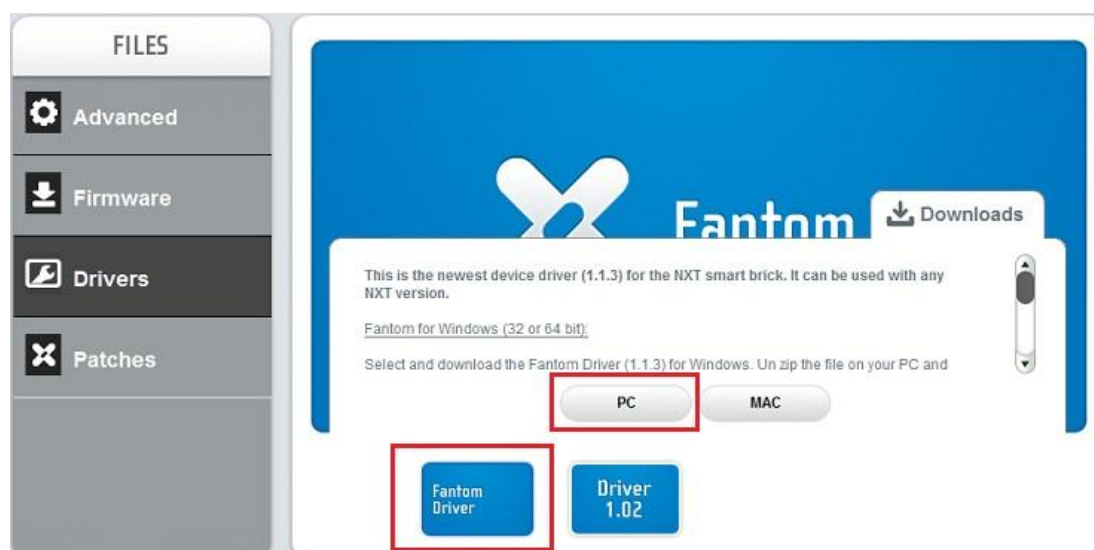
Obr. 7: Nahrání projektu na sdílený disk pomocí webového rozhraní



Obr. 8: Ukázka výstupu kompilace (soubor pepuv_robot_log.html obsahuje průběh kompilace zatímco pepuv_robot_nxtway_app.rxe je aplikace pro robota)

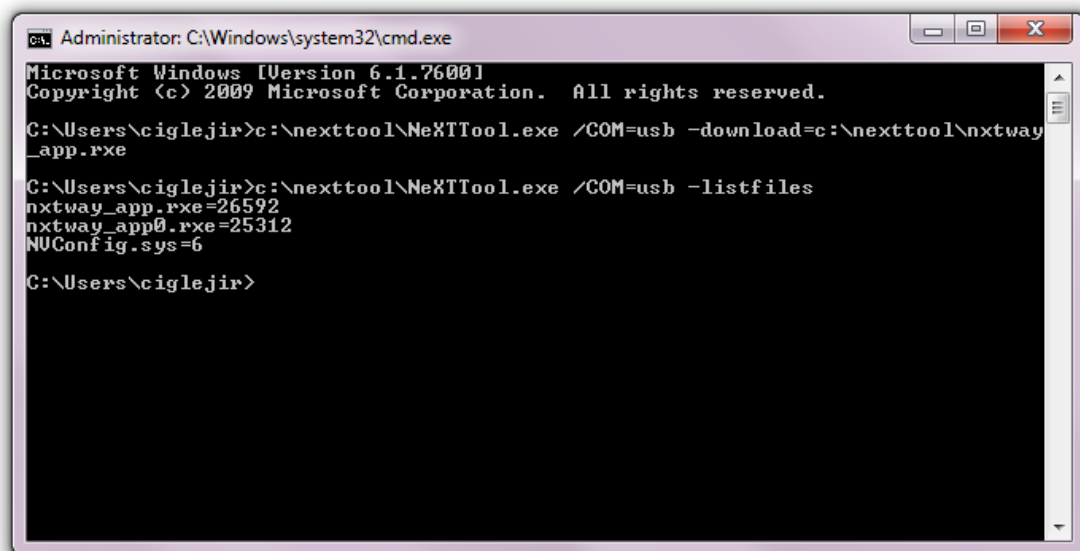
Nahrání aplikace do robota

K nahrání je potřeba mít nainstalovaný Lego Mindstorms NXT Driver Phantom Driver, ten je v učebnách K23 a K26 předinstalován, na jiných počítačích je potřeba jej stáhnout a nainstalovat z <http://mindstorms.lego.com/Support/Updates/>.



Obr. 9: Driver potřebný pro komunikaci s lego robotem

Po připojení LEGO robota k počítači pomocí USB kabelu je možné nahrát zkompilovanou .rxo aplikaci do robota. K tomu lze použít obslužnou aplikaci [NeXTTool.exe](#) podle níže uvedeného obrázku.



```
Administrator: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\ciglejir>c:\nexttool\NeXTTool.exe /COM=usb -download=c:\nexttool\nxtway_app.rxe

C:\Users\ciglejir>c:\nexttool\NeXTTool.exe /COM=usb -listfiles
nxtway_app.rxe=26592
nxtway_app0.rxe=25312
NUConfig.sys=6

C:\Users\ciglejir>
```

Obr. 10:Download aplikace do robota pomocí příkazové řádky ve Windows a ověření, že se to nahrálo

Poznámka k oživení robota: V případě prvního nahrání aplikace do robota bude pravděpodobně potřeba nahrát i nový firmware pomocí sekvence příkazů:

1. Resetujte robota stisknutím tlačítka na zadní straně řídicí jednotky (poblíž USB konektoru). Robot by měl začít tikat.
2. Připojte robota USB kabelem k počítači.
3. Z příkazové řádky zavolejte
./NeXTTool.exe /COM=usb -firmware=lms_arm_nbcnxc_128.rfw
4. Vyjměte a znovu zapojte baterku.

Instalace nexttool na Linuxu

Nainstalujte libusb-dev, libusb-0.1.4, fpc, subversion tedy např
sudo apt-get install libusb-dev libusb-0.1.4 fpc subversion

Stáhněte lokální kopii repository projektu *bricxcc*

```
svn co -r 1 https://bricxcc.svn.sourceforge.net/svnroot/bricxcc/ bricxcc
cd bricxcc
make -f./nexttool.mak
```

Poté je nexttool připraven k nahrání aplikace do robota, je potřeba se zarootovat.
sudo ./NeXTTool /COM=usb -download=nxtway_app.rxe

Instalace nexttool na Mac OS X

Podle <http://bricxcc.sourceforge.net/utilities.html>