README for PUMA w/STG Board Simulink Library

Files:

matlab_files_4_qnx.tar
qnx_rtw_files.tar
pumastg.tar
puma.mdl

Instructions:

1. On the QNX target system, create a directory for Matlab source files. Preferably, because the default makefile expects this, it should be `/matlab`. If you do not have root access to create this, the makefile will need to be edited. The directory you choose will be referred to `MATLAB_ROOT` in the rest of this document.
2. On the QNX target system, untar matlab_files_4_qnx.tar in the `MATLAB_ROOT` directory.
3. On the QNX target system, untar qnx_rtw_files.tar in the `MATLAB_ROOT` directory.
4. On the QNX target system, untar pumastg.tar in its own directory. Again, because the default makefile expects it, this directory should be `MATLAB_ROOT/toolbox/pumastg`.
5. On the development system, untar qnx_rtw_files.tar, preferably in the Matlab home directory. If you do not have access to write in the Matlab home directory, untar this file in its own directory. This directory will be referred to as `QNX_RTW_HOME` in the rest of this document.
6. On the development system, untar pumastg.tar, preferably in the Matlab home directory. If you do not have access to write in the Matlab home directory, untar this file in its own directory. This directory will be referred to as `PUMASTG_HOME` in the rest of this document.
7. Start Matlab
8. Go to File->Set Path… and add `QNX_RTW_HOME` and `PUMASTG_HOME`.
9. Open the Simulink Library Browser and look to see if the PUMA w/STG Board Library has been added.
10. Go to `PUMASTG_HOME` in the Matlab command window. Before being able to simulate or compile code, the blocks in the library need to be compiled by the Matlab mex compiler. Do the following:
    a. mex fltrd.c
    b. mex fltwr.c
    c. mex shortwr.c
    d. mex longrd.c
    e. mex robot_init.c
    f. mex robot_power.c

    g. mex robot_zero.c
    h. mex flog.c
    i. mex trajq.c
11. Open puma.mdl. Everything should be set to build. Go to Tools->Real-Time Workshop->Options to look at various settings.
12. To generate code, go to Tools-Real-Time Workshop->Build Model. Or, with the model window active, hit cntl-B. The code will be written to a directory with the name <model>_qnx_rtw, which in this case is puma_qnx_rtw.
13. If the development system is Windows, you will need to strip the extra DOS carriage returns from the code. A utility like `dos2unix` works well.
14. Move the directory to the QNX target system.
15. In the puma_qnx_rtw, there is a makefile, puma.mk. Compile with the command `make -f puma.mk`.
16. This will create an executable named `puma` in the parent directory.
17. Make sure to have the power switch in hand before running. After the executable is started, hit the power switch immediately to start the controller.

Contact:

For any assistance, email me at martin@cs.umn.edu