I. NEARLAB – NES

II. TECHNICAL

III. REPORTS

Title:	How to install Xsens drivers for Simulink
Report Number:	190528
Author:	Stefano Dalla Gasperina
Author's Email:	stefano.dallagasperina@polimi.it
Date:	2019.05
Brief Description	This guide explains how to build Xsens MTw Awinda Simulink blocks (SDK: 4.3.0, Ubuntu 18.04, MTw Awinda Station)
Revision:	1.0

I. INTRODUCTION

This guide walks you through the procedure of installing and building Xsens drivers for MATLAB/Simulink in Linux environment.

The procedure explained in this report is for the following versions: **Ubuntu 18.04** and **Xsens MTSDK 4.3.0**. Most of the report was written with the information found on the internet. You can find the links to some of these online pages in the references section.

The guide can be divided in different phases:

1. Downloading and unpacking
2. Installing Xsens MTSDK 4.3.0

   https://www.xsens.com/mt-software-suite-mtw-awinda/
3. Compiling Xsens block from source
4. Building Simulink model and testing

Let us start by installing some dependencies. It is possible that this list is not as exhaustive as it should be. If at any point during this process you encounter an error a particular program cannot be find, more than likely you just need to apt-get it.

(NOTE USE THE WORD FILE IF YOU FACE PROBLEMS IN COPY PASTING COMMANDS LONGER THAN ONE LINE. YOU MAY NEED TO REPLACE THE APASTRPHS)

```
sudo apt-get install build-essential liblapack-dev libboost-all-dev libusb-1.0-0-dev
sudo apt install chrpath
```

In Ubuntu 18.04 you may find some problems with liblapack.so.3gf dependencies.

To install this package, you may be required to visit https://pkgs.org/download/liblapack.so.3gf or to run:

```
sudo ln -s /usr/lib/lapack/liblapack.so.3 /usr/lib/lapack/liblapack.so.3gf
sudo ln -s /usr/lib/liblapack.so.3 /usr/lib/liblapack.so.3gf
```

According to gcc and g++ compilers, if you use Ubuntu from 14.10., you need to downgrade your gcc and/or g++ compiler version to 4.8. To compile the Xsens examples please install the missing compilers and update compiler options so as to use g++ and gcc 4.8 with higher priority.

First erased the current update-alternatives setup for gcc and g++:

```
sudo update-alternatives --remove-all gcc
sudo update-alternatives --remove-all g++
```

It seems that both gcc-4.8 and g++-4.8 can be installed by default. However, we can explicitly install the following packages:

```
sudo apt-get install gcc-4.8 g++-4.8
```

Symbolic links cc and c++ are installed by default. We will install symbol links for gcc and g++, then link cc and c++ to gcc and g++ respectively. (Note that the 10, 20 and 30 options are the priorities for each alternative.)

Please **replace x.x** with you current gcc/g++ compiler version

```
sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-x.x 10
sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-4.8 20

sudo update-alternatives --install /usr/bin/g++ g++ /usr/bin/g++-x.x 10
sudo update-alternatives --install /usr/bin/g++ g++ /usr/bin/g++-4.8 20
```

The last step is configuring the default commands for gcc, g++. It's easy to switch between 4.8 and x.x interactively:

```
sudo update-alternatives --config gcc
sudo update-alternatives --config g++
```

II. DOWNLOADING AND UNPACKING

From a workspace folder of your choice download the following package, unpack it and run the following commands to install Xsens MT Software Development Kit (MTSDK).

https://www.xsens.com/download/MT/4.3/MT_Software_Suite_Linux_4.3.tar.gz

This guide has been tested only with version 4.3, however it would work in beta version also with MTSDK 4.6.0

https://www.xsens.com/software-downloads?hsCtaTracking=29a46ff7-74cc-45c8-9eb6-019d4718e5a1%7Ceec53702-e1d1-4be0-a0c4-5c1504d888e2

```
cd MT_Software_Suite_Linux_4.3
sudo chmod +x mtsdk_linux_4.3.sh
sudo ./mtsdk_linux_4.3.sh
```

When asked, type the following directory for installation:

```
/usr/local/lib/xsens
```

Run example to check if installation was successful:

Copy Xsens-rules to folder /etc/udev/rules.d (Xsens rules can be found in the shared folder)

```
sudo cp xsens.rules /etc/udev/rules.d
```

To build the example enter the /examples/mtsdk directory and run:

```
make src_cpp
```

This will result in a single executable named "example" in /examples/mtsdk/src_cpp

```
cd /examples/mtsdk/src_cpp
sudo ./example
```

To use libusb, run make with the HAVE_LIBUSB=1

Move installed files to correct folders (**TODO in a more fashion way**)

```
cd /usr/local/lib/xsens/include
sudo cp * -r /usr/local/include
```

```
cd /usr/local/lib/xsens/lib
sudo cp * /usr/local/lib
```

If shared library is not found, try adding /usr/local/lib to matlab path:

Matlab Mainpage -> Set Path -> Add Folder -> /usr/local/lib

III. REFERENCES