Test IMU CHIP ROBOTICS BNO080

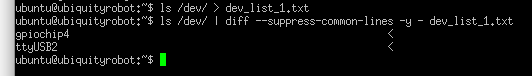
Buy BNO080 from ChipRobotics at <https://chiprobotics.com/>

<https://chiprobotics.gitbook.io/chip-robotics/sensors/imu-sensor>

To get the visualization program to run on Linux using CHIP ROBOTICS IMU

1. Download Linux version from <https://docs.chiprobotics.com/sensors/imu-sensor/software>
2. Extract file to Desktop
3. $ cd /Desktop
4. $ chmod +x IMU\_Sensor\_App-x86\_64\_v1.0.AppImage # to update the permissions
5. $ ./IMU\_Sensor\_App-x86\_64\_v1.0.AppImage # to run the application
6. Point the IMU to the East and press RESET IMU. This will trigger the IMU to report 0 when pointing East (-90) South

Setup udev rule for BNO080 on RPi (lawntractor)

* PuTTy into RPi (tractor)
* Store list of devices connected before plugging in the Teensy
* $ ls /dev/ > dev\_list\_1.txt
* Then run this after you plug it in
* $ ls /dev/ | diff --suppress-common-lines -y - dev\_list\_1.txt
* 

| ubuntu@ubiquityrobot:~$ dmesg | grep "ttyUSB2"  [ 998.396272] usb 1-1.3.3: FTDI USB Serial Device converter now attached to ttyUSB2  ubuntu@ubiquityrobot:~$ dmesg | grep "1-1.3.3"  [ 998.225046] usb 1-1.3.3: new full-speed USB device number 14 using dwc\_otg  [ 998.382659] usb 1-1.3.3: New USB device found, idVendor=0403, idProduct=6015, bcdDevice=10.00  [ 998.382679] usb 1-1.3.3: New USB device strings: Mfr=1, Product=2, SerialNumber=3  [ 998.382693] usb 1-1.3.3: Product: FT230X Basic UART  [ 998.382707] usb 1-1.3.3: Manufacturer: FTDI  [ 998.382721] usb 1-1.3.3: SerialNumber: DM03OZST  [ 998.391229] ftdi\_sio 1-1.3.3:1.0: FTDI USB Serial Device converter detected  [ 998.391487] usb 1-1.3.3: Detected FT-X  [ 998.396272] usb 1-1.3.3: FTDI USB Serial Device converter now attached to ttyUSB2  ubuntu@ubiquityrobot:~$ |
| --- |

* Use FileZilla; Edit /etc/udev/rules.d/05-serial.rules
* Copy/Add a line and update: Vendor, product, serial and use the name chipr\_imu

SUBSYSTEM=="tty", ATTRS{idVendor}=="0403" , ATTRS{idProduct}=="6015" , ATTRS{serial}=="DM03OZST", SYMLINK+="chipr\_imu"

* Save the file, close FileZilla
* If interested, $ udevadm info --query=all --name=/dev/ttyUSB2 --attribute-walk
* $ sudo reboot # to check the rules
* $ ls /dev # look for reference “chipr\_imu”

{issue: setting udev rule like this did not work. ttyUSB2 basically disappeared. Removing the rule and rebooting allowed ttyUSB2 to re-appear. “gpiochip4” appears to work as a sym link. The command $ udevadm info --query=all --name=/dev/gpiochip4 –attribute-walk returns similar results when the device is connected to different USB ports.

Test the ROS driver from: <https://github.com/chiprobotics>

Installing ROS library for CHIP ROBOTICS BNO080 IMU (not BNO085)

* <https://github.com/chiprobotics/chip_imu_driver>
* $ sudo apt-get install ros-noetic-rviz-imu-plugin
* $ sudo apt-get install ros-noetic-serial
* $ cd catkin\_ws/src
* $ git clone<https://github.com/chiprobotics/chip_imu_driver.git>
* $ cd ..
* $ catkin\_make

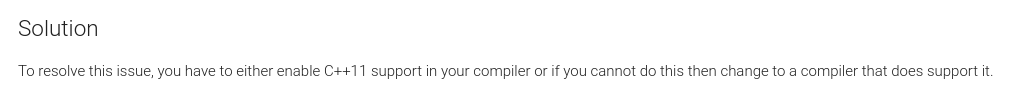
Fails

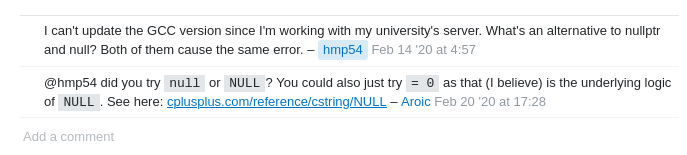
| In file included from /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:1:0:  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/include/chip\_imu\_driver/imu\_serial.h:44:9: warning: scoped enums only available with -std=c++11 or -std=gnu++11  enum class State {  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/include/chip\_imu\_driver/imu\_serial.h:56:33: warning: non-static data member initializers only available with -std=c++11 or -st d=gnu++11  const uint16\_t HEADER = 0xAAAA;  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/include/chip\_imu\_driver/imu\_serial.h:58:37: warning: non-static data member initializers only available with -std=c++11 or -st d=gnu++11  const int32\_t MESSAGE\_LEN = 19;  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/include/chip\_imu\_driver/imu\_serial.h:59:36: warning: non-static data member initializers only available with -std=c++11 or -st d=gnu++11  const int32\_t HEADER\_LEN = 2;  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/include/chip\_imu\_driver/imu\_serial.h:60:32: warning: non-static data member initializers only available with -std=c++11 or -st d=gnu++11  const int32\_t CS\_LEN = 1;  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/include/chip\_imu\_driver/imu\_serial.h:61:67: warning: non-static data member initializers only available with -std=c++11 or -st d=gnu++11  const int32\_t DATA\_FIELD\_LEN = MESSAGE\_LEN - HEADER\_LEN - CS\_LEN;  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp: In constructor ‘imu\_sensor::IMUSerial::IMUSerial(const char\*, int)’:  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:14:17: error: ‘nullptr’ was not declared in this scope  serial\_(nullptr),  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:15:16: error: ‘State’ is not a class or namespace  state\_(State::WaitHeader),  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp: In member function ‘void imu\_sensor::IMUSerial::readAndParse()’:  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:106:19: error: ‘b’ does not name a type  for (auto b : bytes) {  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:109:5: error: expected ‘;’ before ‘}’ token  }  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:109:5: error: expected primary-expression before ‘}’ token  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:109:5: error: expected ‘;’ before ‘}’ token  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:109:5: error: expected primary-expression before ‘}’ token  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:109:5: error: expected ‘)’ before ‘}’ token  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:109:5: error: expected primary-expression before ‘}’ token  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp: In member function ‘void imu\_sensor::IMUSerial::parse(uint8\_t)’:  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:123:18: error: ‘State’ is not a class or namespace  case State::WaitHeader:  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:126:30: error: ‘State’ is not a class or namespace  state\_ = State::WaitData;  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:132:18: error: ‘State’ is not a class or namespace  case State::WaitData:  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:136:30: error: ‘State’ is not a class or namespace  state\_ = State::WaitCS;  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:140:18: error: ‘State’ is not a class or namespace  case State::WaitCS:  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:153:26: error: ‘State’ is not a class or namespace  state\_ = State::WaitHeader;  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:157:26: error: ‘State’ is not a class or namespace  state\_ = State::WaitHeader;  ^  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp: In member function ‘uint8\_t imu\_sensor::IMUSerial::calcCS(const std::vector<unsigned char>&)’:  /home/ubuntu/catkin\_ws/src/chip\_imu\_driver/src/imu\_serial.cpp:170:29: warning: range-based ‘for’ loops only available with -std=c++11 or -std=gnu++11  for (uint8\_t byte : data) {  ^  chip\_imu\_driver/CMakeFiles/imu\_serial.dir/build.make:62: recipe for target 'chip\_imu\_driver/CMakeFiles/imu\_serial.dir/src/imu\_serial.cpp.o' failed  make[2]: \*\*\* [chip\_imu\_driver/CMakeFiles/imu\_serial.dir/src/imu\_serial.cpp.o] Error 1  CMakeFiles/Makefile2:3289: recipe for target 'chip\_imu\_driver/CMakeFiles/imu\_serial.dir/all' failed  make[1]: \*\*\* [chip\_imu\_driver/CMakeFiles/imu\_serial.dir/all] Error 2  Makefile:138: recipe for target 'all' failed  make: \*\*\* [all] Error 2  Invoking "make -j2" failed  ubuntu@ubiquityrobot:~/catkin\_ws$ |
| --- |

Looking here: <https://stackoverflow.com/questions/10033373/c-error-nullptr-was-not-declared-in-this-scope-in-eclipse-ide>

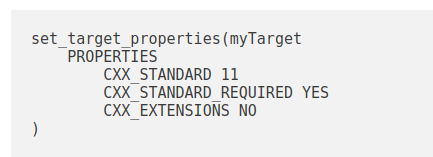
<https://stackoverflow.com/questions/60195415/error-nullptr-was-not-declared-in-this-scope>

| ubuntu@ubiquityrobot:~/catkin\_ws$ gcc -v  Using built-in specs.  COLLECT\_GCC=gcc  COLLECT\_LTO\_WRAPPER=/usr/lib/gcc/arm-linux-gnueabihf/5/lto-wrapper  Target: arm-linux-gnueabihf  Configured with: ../src/configure -v --with-pkgversion='Ubuntu/Linaro 5.4.0-6ubuntu1~16.04.12' --with-bugurl=file:///usr/share/doc/gcc-5/README.Bugs --enable-languages=c,ada,c++,java,go,d,fortran,objc,obj-c++ --prefix=/usr --program-suffix=-5 --enable-shared --enable-linker-build-id --libexecdir=/usr/lib --without-included-gettext --enable-threads=posix --libdir=/usr/lib --enable-nls --with-sysroot=/ --enable-clocale=gnu --enable-libstdcxx-debug --enable-libstdcxx-time=yes --with-default-libstdcxx-abi=new --enable-gnu-unique-object --disable-libitm --disable-libquadmath --enable-plugin --with-system-zlib --disable-browser-plugin --enable-java-awt=gtk --enable-gtk-cairo --with-java-home=/usr/lib/jvm/java-1.5.0-gcj-5-armhf/jre --enable-java-home --with-jvm-root-dir=/usr/lib/jvm/java-1.5.0-gcj-5-armhf --with-jvm-jar-dir=/usr/lib/jvm-exports/java-1.5.0-gcj-5-armhf --with-arch-directory=arm --with-ecj-jar=/usr/share/java/eclipse-ecj.jar --enable-objc-gc --enable-multiarch --enable-multilib --disable-sjlj-exceptions --with-arch=armv7-a --with-fpu=vfpv3-d16 --with-float=hard --with-mode=thumb --disable-werror --enable-multilib --enable-checking=release --build=arm-linux-gnueabihf --host=arm-linux-gnueabihf --target=arm-linux-gnueabihf  Thread model: posix  gcc version 5.4.0 20160609 (Ubuntu/Linaro 5.4.0-6ubuntu1~16.04.12)  ubuntu@ubiquityrobot:~/catkin\_ws$ |
| --- |





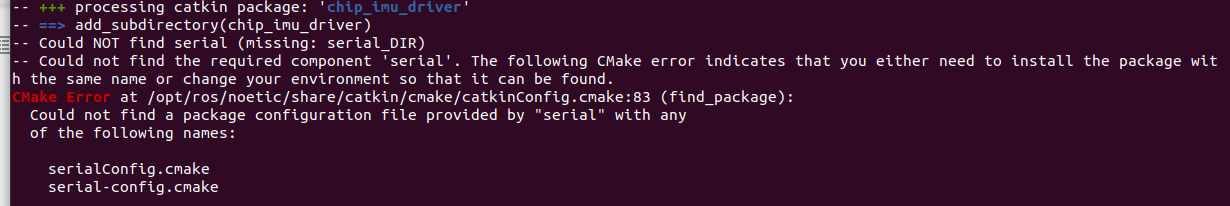
<https://crascit.com/2015/03/28/enabling-cxx11-in-cmake/>



Current command:

| set\_target\_properties(chip\_imu\_driver\_node PROPERTIES OUTPUT\_NAME chip\_imu\_driver\_node PREFIX "") |
| --- |

| set\_target\_properties(chip\_imu\_driver\_node  PROPERTIES  OUTPUT\_NAME chip\_imu\_driver\_node PREFIX ""  CXX\_STANDARD 11  CXX\_STANDARD\_REQUIRED YES  CXX\_EXTENSIONS NO  ) |
| --- |

First issue - getting “serial on Ubuntu 20 / Noetic laptop” 

| Fixing the serial error   * + $ cd   + $ sudo apt-get install ros-noetic-serial     Attempting to build rosserial locally   * $ cd catkin\_ws/src * $ git clone <https://github.com/ros-drivers/rosserial.git> * $ cd .. * $ catkin\_make     $ rosdep update # did not help, same error |
| --- |

Steps needed to complete the process, but not yet attempted because of errors above

* $ source ./devel/setup.bash
* $ rospack find chip\_imu\_driver
* $ roslaunch chip\_imu\_driver example\_rviz.launch

7/31/21

| al@al-ThinkPad-W530:~$ sudo apt-get install ros-noetic-serial  Reading package lists... Done  Building dependency tree  Reading state information... Done  E: Unable to locate package ros-noetic-serial |
| --- |

<https://wiki.ros.org/rosserial>

| Install the binaries  <https://lkk62367.gitbooks.io/ros-serial/content/install_rosserial_package.html>   * $ sudo apt-get install ros-indigo-rosserial-arduino * $ sudo apt-get install ros-indigo-rosserial     Install rosserial locally (although this seemed to work - rospack find rosserial did not work. I was able to install binaries and get the Arduino example to work)   * $ cd catkin\_ws/src * $ git clone <https://github.com/chiprobotics/chip_imu_driver.git> * $ cd .. * $ catkin\_make * $ catkin\_make install * $ source catkin\_ws/install/local\_setup.bash * $ source /home/al/catkin\_ws/install/local\_setup.bash * $ rospack find rosserial |
| --- |

Code for IMU - have not looked at it yet

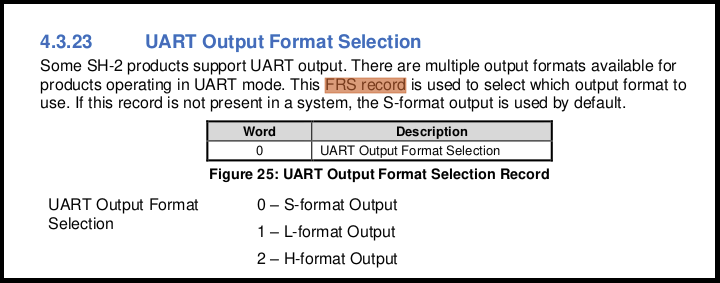
<https://github.com/jerabaul29/Artemis_MbedOS_recipes/tree/main/recipes/recipe_IMU_quaternions>

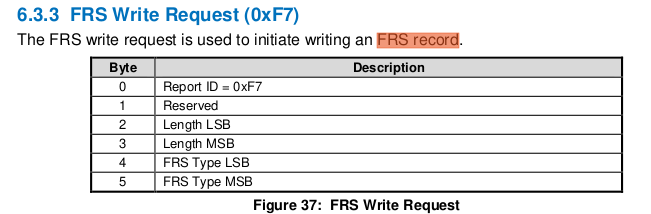
Background notes on BNO085

7/31/21

Based on looking at the ROS Package program the IMU is looking for a Header 16-bit packet header, always equal to 0xAAAA which is defined in 3.1.3 H-format Output Packet

It seems an FRS record us used and needs to be “2”





API - one command mentioned

https://docs.chiprobotics.com/sensors/imu-sensor/api

9/5/21

$ roslaunch chip\_imu\_driver example.launch

Change

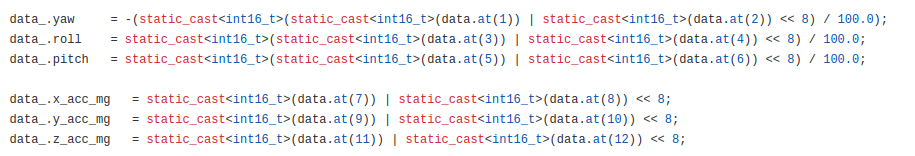
<!-- <arg name="port" default="/dev/ttyUSB0" /> -->

<arg name="port" default="/dev/gpiochip0" /> # or whatever will work

PARAMETERS \* /chip\_imu\_driver\_node/port: /dev/ttyUSB0

Need to update launch file with correct USB





Recompiling the package after making changes:

* $ roscd
* $ cd ..
* $ catkin\_make
* $ source ./devel/setup.bash
* $ roslaunch chip\_imu\_driver example.launch

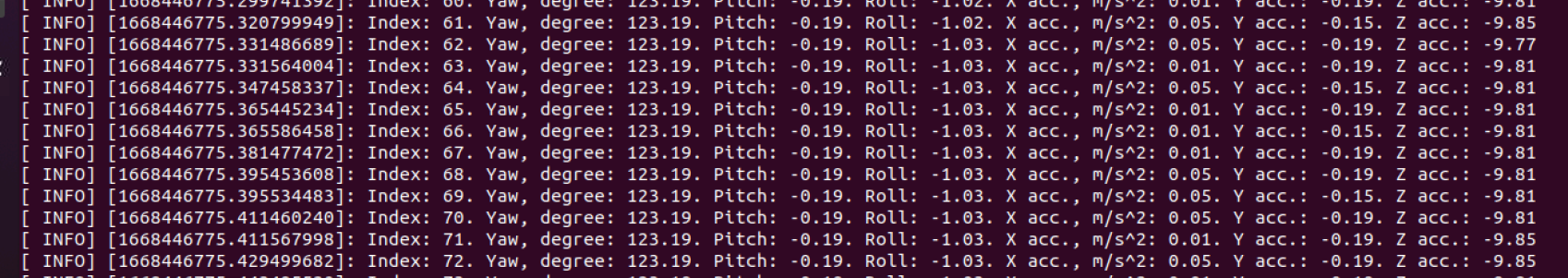
$ roslaunch chip\_imu\_driver example.launch port:=/dev/ttyUSB1

$ rosrun beginner\_tutorials ros\_yaw\_from\_imu.py

11/14/22 - running Chip Robotics IMU

<!-- <arg name="port" default="/dev/ttyUSB0" /> -->

<arg name="port" default="/dev/gpiochip0" />



I would expect Yaw to be 180 on a 360 degree basis

But the range appears to be -180 to 0 to +180

<https://stackoverflow.com/questions/10697844/how-to-deal-with-the-discontinuity-of-yaw-angle-at-180-degree/31240443#31240443>

<https://answers.ros.org/question/141366/convert-the-yaw-euler-angle-into-into-the-range-0-360/>

$ ls /dev/ | diff --suppress-common-lines -y - dev\_list\_1.txt

* ls /dev/ | diff --suppress-common-lines -y - dev\_list\_1.txt
* usb 8-1.1.2: can't set config #1, error -32

11/21/22 - trying

* $ cd /Desktop
* $ ./IMU\_Sensor\_App-x86\_64\_v1.0.AppImage # to run the application
* Execute the application
* Connect on /dev/ttyUSB1

|  |
| --- |

* Find ros\_yaw\_from\_imu.py and copy into /home/tractor/catkin\_ws/src/beginner\_tutorials/scripts
* $ cd /home/tractor/catkin\_ws/src/beginner\_tutorials/scripts
* $ chmod +x ros\_yaw\_from\_imu.py
* $ roslaunch chip\_imu\_driver example.launch port:=/dev/ttyUSB1
* $ rosrun beginner\_tutorials ros\_yaw\_from\_imu.py
* The printed output “yaw” will range from 3.14 to -3.14
* If East is set at 0, By the right hand rule, the yaw component of orientation increases as the child frame rotates counter-clockwise, yaw is zero when pointing east. East (0) to North(1.57) to West(+3.14) and then South(-1.57)