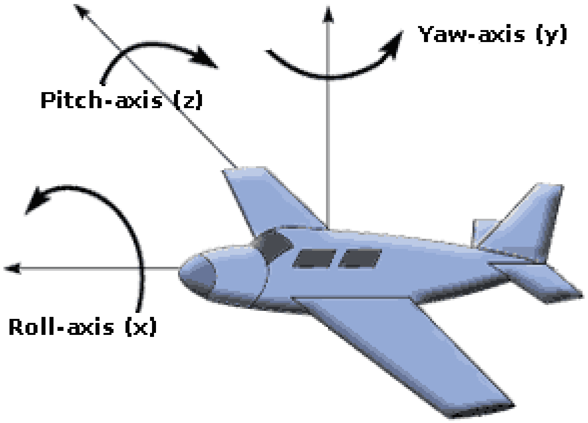
Source:= https://github.com/cyberbotics/webots/tree/released/docs/reference

Inertial unit

The InertialUnit node simulates an *Inertial Measurement Unit* (IMU). The InertialUnit computes and returns its *roll*, *pitch* and *yaw* angles with respect to a global coordinate system defined in the WorldInfo node.

```
from controller import InertialUnit

class InertialUnit (Device):
    def enable(self, samplingPeriod):
    def disable(self):
    def getSamplingPeriod(self):
    def getRollPitchYaw(self):
    def getQuaternion(self):
    def getNoise(self):
    # ...
```



GPS

The GPS node is used to model a Global Positioning Sensor (GPS) which can obtain information about its absolute position from the controller program.

```
from controller import GPS

class GPS (Device):
    def enable(self, samplingPeriod):
    def disable(self):
    def getSamplingPeriod(self):
    def getValues(self):
    def getSpeed(self):
```

COMPASS

A Compass node can be used to model a 1, 2 or 3-axis digital compass (magnetic sensor). The Compass node returns a vector that indicates the north direction specified by the coordinateSystem field of the WorldInfo node.

```
from controller import Compass

class Compass (Device):
   def enable(self, samplingPeriod):
   def disable(self):
   def getSamplingPeriod(self):
   def getValues(self):
   def getLookupTable(self):
   # ...
```

GYRO

The Gyro node is used to model 1, 2 and 3-axis angular velocity sensors (gyroscope). The angular velocity is measured in radians per second [rad/s].

```
from controller import Gyro

class Gyro (Device):
   def enable(self, samplingPeriod):
   def disable(self):
   def getSamplingPeriod(self):
   def getValues(self):
   def getLookupTable(self):
   # ...
```

DISTANCE SENSORS

The DistanceSensor node can be used to model a generic sensor, an infra-red sensor, a sonar sensor, or a laser range-finder. This device simulation is performed by detecting the collisions between one or several sensor rays and objects in the environment.

We have used laser sensors.

```
from controller import DistanceSensor

class DistanceSensor (Device):
   def enable(self, samplingPeriod):
   def disable(self):
   def getSamplingPeriod(self):
   def getValue(self):
   # ...
```