



EXAMENSARBETE INOM ELEKTROTEKNIK,  
GRUNDNIVÅ, 15 HP  
*STOCKHOLM, SVERIGE 2017*

# **Evaluation of MEMS accelerometer and gyroscope for orientation tracking nutrunner functionality**

## **Utvärdering av MEMS accelerometer och gyroskop för rörelseavläsning av skruvdragare**

**ERIK GRAHN**



# **Evaluation of MEMS accelerometer and gyroscope for orientation tracking nutrunner functionality**

Utvärdering av MEMS accelerometer och gyroskop för  
rörelseavläsning av skruvdragare

Erik Grahn

Examensarbete inom Elektroteknik,  
Grundnivå, 15 hp  
Handledare på KTH: Torgny Forsberg  
Examinator: Thomas Lind  
TRITA-STH 2017:115

KTH  
Skolan för Teknik och Hälsa  
141 57 Huddinge, Sverige



## **Abstract**

In the production industry, quality control is of importance. Even though today's tools provide a lot of functionality and safety to help the operators in their job, the operators still is responsible for the final quality of the parts. Today the nutrunners manufactured by Atlas Copco use their driver to detect the tightening angle. Therefore the operator can influence the tightening by turning the tool clockwise or counterclockwise during a tightening and quality cannot be assured that the bolt is tightened with a certain torque angle. The function of orientation tracking was desired to be evaluated for the Tensor STB angle and STB pistol tools manufactured by Atlas Copco. To be able to study the orientation of a nutrunner, practical experiments were introduced where an IMU sensor was fixed on a battery powered nutrunner. Sensor fusion in the form of a complementary filter was evaluated. The result states that the accelerometer could not be used to estimate the angular displacement of tightening due to vibration and gimbal lock and therefore a sensor fusion is not possible. The gyroscope by itself can be used to provide the angular displacement around every axis with high accuracy without taking into account the gimbal lock phenomena or external forces in the form of vibration of the tool. The gyroscope provided data with a probability to measure  $\pm 1^\circ$  in future tightenings by 69,76%. The gyroscope provided data with high accuracy and stability and can be used in real world application and production for true angle functionality of the tools.

## **Keywords**

MEMS, IMU, accelerometer, gyroscope, nutrunner, orientation tracking



## **Sammanfattning**

I produktionsindustrin är kvalitetskontroll av stor betydelse. Även om dagens verktyg innehåller mycket funktionalitet och säkerhet för att hjälpa operatörer i jobbet, är operatören fortfarande ansvarig för den slutliga kvaliteten. Idag använder Atlas Copcos skruvdragare motorns vridmoment för att göra den slutliga åtdragningen. Därav kan operatören påverka åtdragningen genom att vrinda verktyget medurs eller moturs under en åtdragning och kvaliteten kan inte säkerställas att bulten dras med ett visst vridmoment. Funktion för rörelseavläsning var önskvärd att utvärderas för Tensor STB-vinkel- och STB-pistolverktygen tillverkade av Atlas Copco. För att kunna studera orienteringen hos en skruvdragare introducerades praktiska experiment där en IMU-sensor fixerades på en batteridriven skruvdragare. En Sensorfusion i form av ett komplementärt filter utvärderades. Resultaten visar att accelerometern inte kunde användas för att uppskatta vinkelförskjutningen av en åtdragning på grund av vibration och gimballås och därav kan inte fusionen heller nyttjas. Gyroskopet i sig kan användas för att ge vinkelförskjutningen runt varje axel med hög noggrannhet utan att ta hänsyn till gimballåsfenomen eller yttre krafter i form av vibration från verktyget. Gyroskopet gav data med en sannolikhet att mäta  $\pm 1^\circ$  i avvikelse från ett förbestämt värde i framtida åtdragningar med 69,76%. Vidare utvärdering av gyroskop och implementation av detta borde göras innan detta skulle anses kunna användas i en riktig applikation för rörelseavläsning under en åtdragning.

## **Nyckelord**

MEMS, IMU, accelerometer, gyroskop, skruvdragare, rörelseavläsning



## **Acknowledgements**

I would like to express my appreciation for my supervisor at KTH, Torgny Forsberg for valuable feedback of the report. I would also like to thank my industrial supervisors, Stefan Olofsson, Wilhelm Nyberg and Anton Lagerholm for all support and giving me the opportunity to investigate this topic at Atlas Copco.



## Nomenclature

### Abbreviations

MEMS	Micro Electro-Mechanical Systems
IMU	Inertial measurement unit
EKF	Extended Kalman filter
STB	Battery powered nutrunner manufactured by Atlas Copco.
DPS	Degrees per second.
SPI	Serial Peripheral Interface

### Symbols

$\phi$	Angle around X axis. Also referred to as Roll
$\psi$	Angle around Z axis. Also referred to as Yaw
$\theta$	Angle around Y axis. Also referred to as Pitch
Yaw	Angle around Z-axis.
Pitch	Angle around Y axis.
Roll	Angle around X axis.



# Contents

1	Introduction.....	1
1.1	Problem statement.....	1
1.2	Objectives .....	1
1.3	Delimitations .....	2
2	Background and theory.....	3
2.1	True angle .....	3
2.2	Inertial measurement unit.....	4
2.2.1	Basic principle of accelerometer .....	4
2.2.2	Basic principle of gyroscope .....	5
2.3	Attribute of IMU's.....	6
2.3.1	Euler angles to determine rotation .....	6
2.3.2	Quaternions .....	8
2.4	Error characteristics in MEMS gyroscope and accelerometers .....	9
2.4.1	Constant bias .....	9
2.4.2	Thermo-mechanical white noise .....	9
2.4.3	Flicker noise and bias stability .....	9
2.4.4	Temperature effects .....	9
2.4.5	Calibration errors.....	9
2.4.6	Summary of error characteristics .....	9
2.5	Sensor fusion.....	10
2.5.1	Complementary filter .....	10
2.5.2	Kalman filter .....	11
2.5.3	Complementary filter vs Kalman filter.....	11
3	Methodology.....	13
3.1	IMU calibration .....	13
3.2	Static analysis.....	14
3.3	Dynamic analysis .....	15
3.3.1	Tightening program STB angle.....	16
3.3.2	Tightening program STB pistol .....	16
3.3.3	Design of the experiments .....	17
4	Results .....	19
4.1	Result of static experiments .....	19
4.2	Results of dynamic experiments.....	24
5	Analysis and Discussion.....	27
5.1	Static analysis .....	27

5.2	Dynamic analysis .....	27
5.3	Statistical analysis of the tightenings .....	28
5.4	Economic and environmental aspects.....	29
6	Conclusion.....	31
6.1	Future work .....	31
7	References .....	33
	Appendix.....	35
A.	Datasheet of the LSM6DSM sensor.....	35
B.	Tightenings.....	37
C.	Code used for experiments .....	41





# 1 Introduction

In the manufacturing industry, quality control is of importance. Even though today's tools provide a lot of functionality and safety to help the operators in their job, the operator still are responsible for the final quality of the parts. Atlas Copco battery powered nutrunners use different tightening programs based on the type of job to be performed. The nutrunners are equipped with an intelligent control system and software. Today the tool uses its driver to perform the final torque angle of a tightening. Therefore the operator can influence the tightening by turning the tool clockwise or counterclockwise during a tightening and quality cannot be assured that the bolt is tightened with a certain torque angle. A technique called *true angle* was introduced in a corded tool Tensor STR that is able to compensate the tightening angle for movement of the tool during a tightening. The function of true angle is desired to be implemented in the Tensor STB angle and STB pistol tools in the future and Atlas Copcos vision is for it to be platform independent so that it can be used in their complete series of cordless tools.

Atlas Copco is a leading manufacturer of innovative tools and assembly solutions. Their tools are used in industries of construction of airplanes, appliances, electronics or in environmental work where cutting, bending or shaping metal is required. They supply a comprehensive range of high productivity assembly tools with intelligent controllers and a sharp focus on ergonomics. Their products range from low torque MicroTorque tools providing 0,5 Ncm of torque, to high torque Tensor ST tools, offering up to 4,000 Nm.

Atlas Copco wants to implement the functionality of motion tracking with the help of MEMS gyroscope and accelerometers in their next generation of battery powered nutrunners primarily used in production industries.

## 1.1 Problem statement

The thesis aims to test and evaluate MEMS (micro electro-mechanical systems) based IMU (inertial measurement unit) to achieve the requirements by Atlas Copco of accuracy within  $\pm 1^\circ$ , flexibility and innovation in their next generation of handheld battery powered tools to track motion and movements caused by the operator during a tightening. The thesis will explore the applicability of MEMS-based IMU's as a navigation solution for orientation tracking of nutrunners in bolt tightening processes.

## 1.2 Objectives

The IMU are going to be fixed on the nutrunner and be able to detect the amount of motions caused by the operator during a tightening process. Properties to be tested and verified are:

- Accuracy
- Stability

Based on the produced data, additional software implementations will be designed and added in the form of filtering, calibration, control and adjustment to make proper use of the provided data from the IMU. The goal is to evaluate the usage of MEMS based IMU in a handheld nutrunner tool used in industry to provide relevant data depending on angular rate and motion of the tool.

### **1.3 Delimitations**

The thesis will not be focusing on implementation of different functions depending on the data provided by the IMU. The work primarily intends to investigate the use of MEMS based gyroscope and accelerometers in handheld nutrunners for angle compensation. It will not investigate the estimation of position movement that can be done by gyroscopes and accelerometers rather just the angular displacement in a three dimensional space.

## 2 Background and theory

The following section will brief the reader about the nutrunner tools that Atlas Copco wants to implement true angle functionality into, three dimensional orientation theory and also basic principles of MEMS based gyroscopes and accelerometers.

The main advantage of battery tools is their superior flexibility due to the absence of cables. This, in turn, improves the efficiency of the operator. Battery assembly tools also contribute to improving safety in the workplace, since there are no air hoses or electric cables to become entangled or jammed. Tensor ETV STB and ETP STB are ergonomically designed battery powered nutrunners by Atlas Copco. The ETV STB is an angle tool with a torque range from 2 up to 150 Nm and the ETP STB is a pistol tool that operates in a torque range of 4 to 12 Nm. Both nutrunners offers flexibility advantages for the operator which, in turn, raises operator efficiency.

### 2.1 True angle

The definition of true angle by Atlas Copco is stated below and the principle can also be observed in figure 1 and 2.

*"Eliminating operator influence, TrueAngle® uses a gyro signal to read operator movement, increasing the accuracy of your angle reading. Enabling handheld angle controlled tightenings while removing operator influence. This allows you to narrow down your angle window, and detect the most common assembly problems such as wrong or missing components and damaged threads. Avoid premature shut-offs and detect re-hits. All this will increase your quality and lower costs connected to scrapping and rework".*

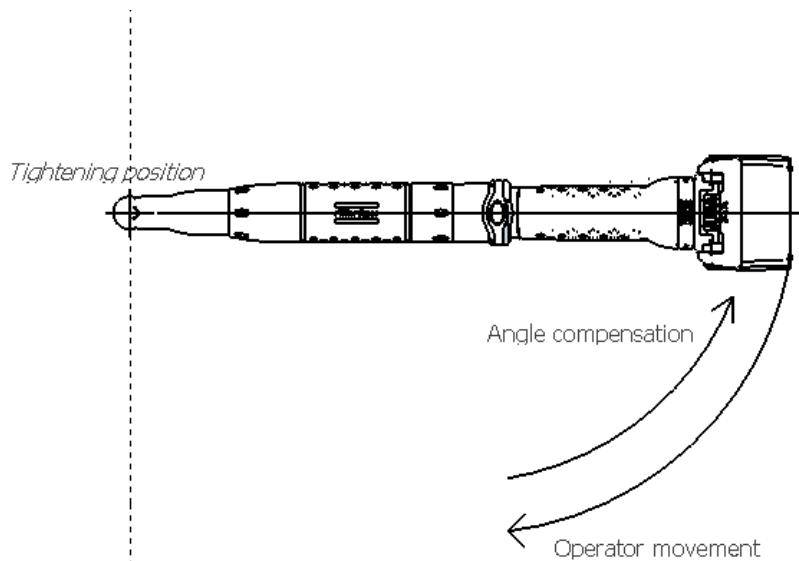
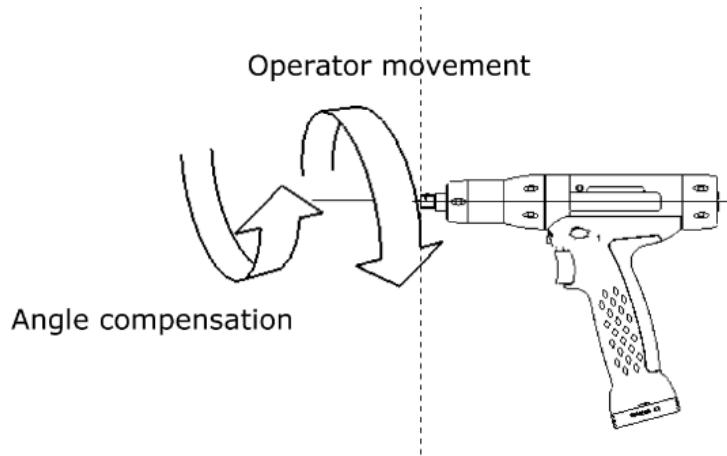


Figure 1. Basic illustration of the principle of true angle of a STB tool seen from top view.



*Figure 2. Basic illustration of the principle of true angle of a STB pistol tool seen from side view.*

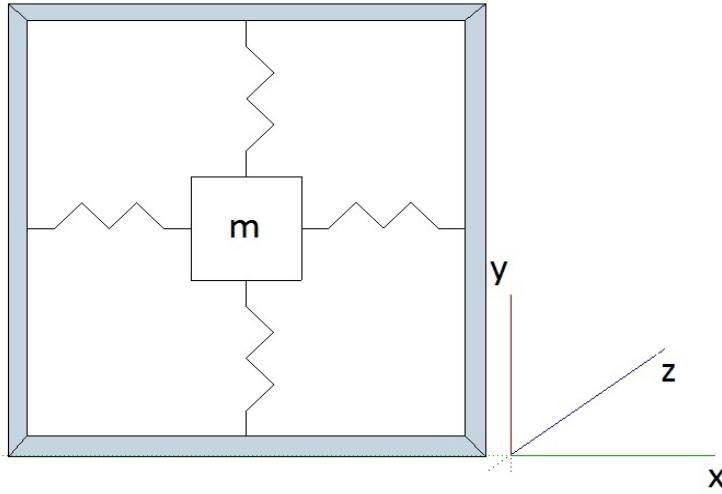
## 2.2 Inertial measurement unit

Typically three orthogonal rate-gyroscopes and three orthogonal accelerometers that are measuring angular velocity and linear acceleration in a three dimensional space are contained in an IMU. In relation to a known starting point, orientation and velocity, gyroscopes and accelerometers are used to track the position and orientation of an object [1]. Functionality provided by gyroscopes have been around for 150 years where one important application has been navigation of ships. Accelerometers have been around for 90 years and used in aircrafts and dynamometers [2]. Recent advances in MEMS technology have made it possible to keep tracking of human motion and orientation of devices with the possibility to manufacture smaller and lighter sensors [1].

### 2.2.1 Basic principle of accelerometer

For simplicity, MEMS accelerometer in one dimension can basically be seen as a process operating of Newton's second law where a mass is attached to a spring within a reference frame, see figure 3.

$$F = k * x = m * \ddot{x} \quad (1)$$



*Figure 3. Illustration of the basic principle of Newton's second law in two dimensions where a mass **m** is attached to springs within a reference frame.*

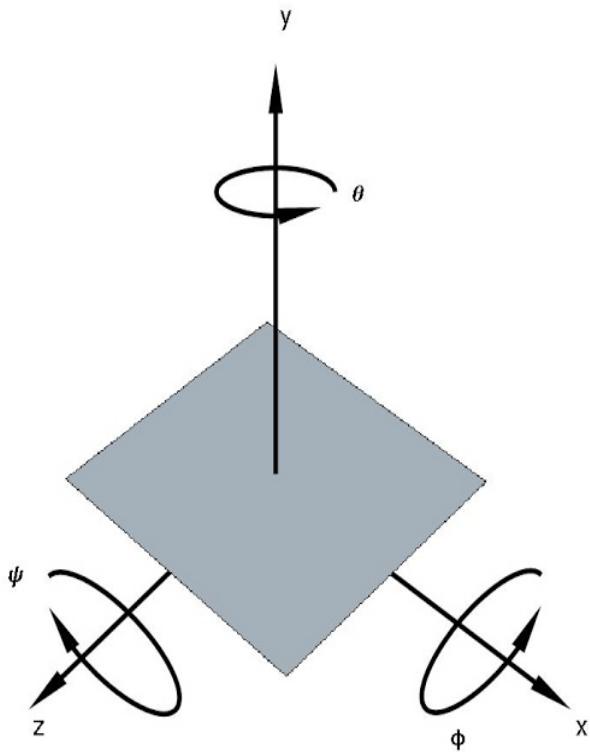
There are two main principles of MEMS accelerometers where one is to measure the displacement of the mass and the other one measures the change in frequency of a vibrating element (mass) caused by change of tension. The accelerometer measures linear acceleration and by integrate the signal twice we can obtain the position.

#### 2.2.2 Basic principle of gyroscope

MEMS gyroscope make use of the experience called Coriolis effect, which states that in a frame of reference rotating at angular velocity  $\omega$ , a mass **m** moving with velocity **v** experience a force:

$$F_c = -2 * m(\boldsymbol{\omega} \times \mathbf{v}) \quad (2)$$

To measure the Coriolis effect, MEMS gyroscope contains vibrating masses which vibrates along a drive axis. A secondary vibration is induced along the perpendicular sense axis which displaces the mass from its original path when the gyroscope is rotated [1]. The working principles of a gyroscope where the rotation around respective axes is introduced can be observed in figure 4. The gyroscope introduces capacitance changes to detect this displacements. Based on this, the angular velocity of the IMU can be measured and by integrate the signal we can obtain orientation.



*Figure 4. Illustration of the basic principle of a gyroscope and how the rotations ( $\phi$ ,  $\theta$ ,  $\psi$ ) occur around respective axes.*

### 2.3 Attribute of IMU's

Today IMU's often includes both three-axis gyroscopes and three-axis accelerometers which provides the angular velocities  $\omega_x$ ,  $\omega_y$  and  $\omega_z$  (gyroscope) and the linear accelerations  $a_x$ ,  $a_y$  and  $a_z$  (accelerometer). To calculate the attitude angles (rotations around the axis) of the IMU, the measured angular velocities and linear accelerations are utilized. The angle around Z-axis cannot be determined by the accelerometer since rotation along that axis will not cause any changes in relation to gravity [3].

There are essentially two representations of orientation tracking that are widely used in motion tracking [4].

1. Euler angles
2. Quaternions

#### 2.3.1 Euler angles to determine rotation

Euler angles are relatively easy to understand but it has singularity problem when the yaw angle is  $90^\circ$ , called gimbal lock and requires knowledge about rotation sequences [5]. Describing the theory about Euler angles and gimbal lock is out of scope in this thesis but will just brief the reader about methods for rotation tracking around respective axis.

The rotation matrices of respective axis can be carried out for moving from the inertial frame to the body frame:

Angle around X-axis, also referred to as Roll ( $\phi$ ) is measured by the accelerometer and given by the rotation matrix:

$$Rx(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \quad (3)$$

where

$$\phi = \arctan\left(-\frac{a_x}{a_z}\right) \quad (4)$$

Angle around Y-axis, also referred to as Pitch ( $\theta$ ) is measured by the accelerometer and given by the rotation matrix:

$$Ry(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \quad (5)$$

where

$$\theta = \arctan\left(\frac{-a_y}{\sqrt{a_x^2 + a_z^2}}\right) \quad (6)$$

Angle around Z-axis, also referred to as Yaw ( $\psi$ ) is measured by the gyroscope and given by the rotation matrix:

$$Rz(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (7)$$

$$\psi = -\arctan\left(\frac{\omega_x}{\omega_y}\right) \quad (8)$$

As can be observed in equation 6, the denominator of the pitch is defined to always be positive, so the equation itself only provides [-90, 90] range, which is exactly what is expected for the pitch angle. In contrast, the roll equation provides [-180, 180] range. It is important to take into account that when the pitch angle is 90°, the rotation around the X-axis (roll) is directly aligned with the gravity vector, thus cannot be measured anymore. This is what is called Gimbal lock.

### 2.3.2 Quaternions

Quaternions provide a measurement technique that does not suffer from gimbal lock but are more complex than Euler angles [4]. A quaternion is a four-element vector and describes a hyper-sphere that is composed of one real element and three complex elements [ $q_0 iq_1 jq_2 kq_3$ ]. A rotation matrix can be represented as a quaternion.

$$C_t^f = \begin{bmatrix} c_1 & c_4 & c_7 \\ c_2 & c_5 & c_8 \\ c_3 & c_6 & c_9 \end{bmatrix} \quad (9)$$

$$= \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (10)$$

Where  $q_0, q_1, q_2$  and  $q_3$  are the quaternion components that satisfy:

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \quad (11)$$

From the angular velocity measurements the quaternion components are computed using following equation:

$$\dot{q} = \frac{1}{2} Q * \omega \quad (12)$$

$$= \begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix} * \begin{bmatrix} 0 \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (13)$$

$$\Psi = \text{Atan2}(2q_1 q_2 - 2q_0 q_3, 2q_0^2 + 2q_1^2 - 1) \quad (14)$$

$$\Theta = -\sin^{-1}(2q_1 q_3 + 2q_0 q_2) \quad (15)$$

$$\Phi = \text{Atan2}(2q_2 q_3 - 2q_0 q_1, 2q_0^2 + q_3^2 - 1) \quad (16)$$

Notice the Atan2 function is the implementation used in software to be able to calculate the arctangent of all four quadrants.

## 2.4 Error characteristics in MEMS gyroscope and accelerometers

Both gyroscope and accelerometers suffer from the same error characteristics but the difference is the significance of the errors due to the integration part of the signals [1][3][6][7].

### 2.4.1 Constant bias

When the gyroscope is not undergoing any rotation it outputs an average constant bias  $\epsilon$  which causes an angular error:

$$\theta(t) = \epsilon \cdot t \quad (17)$$

with time due to integration of the signal. The bias of the accelerometer when double integrated, causes an error in position which grows quadratically with time:

$$s(t) = \epsilon * \frac{t^2}{2} \quad (18)$$

### 2.4.2 Thermo-mechanical white noise

The output of a MEMS gyroscope are interfered with thermal noise and is usually called *Angle random walk* which can be referred to as a sequence of uncorrelated random variables that usually are defined with units  $^\circ/\sqrt{h}$ . The accelerometer is also interfered with thermal noise and is referred to as *Velocity random walk* and are usually defined in the unit  $m/s/\sqrt{h}$ .

### 2.4.3 Flicker noise and bias stability

The bias of a MEMS IMU drift over time due to flicker noise in the electronics. The noise is also referred to as pink noise and the bias fluctuations in gyroscopes are usually modelled as angle random walk while in accelerometers as bias random walk. Flicker noise tends to be overshadowed by white noise at high frequencies.

### 2.4.4 Temperature effects

Changes in the environmental temperature can cause movement in the bias of the sensor and grows linearly with time. Most IMU's today contains internal temperature sensors to make up for the induced bias effects.

### 2.4.5 Calibration errors

This refers to manufacturing errors such as scale factors, alignments and linearities.

### 2.4.6 Summary of error characteristics

Since all these errors described above is factored in the integration steps, the position and orientation errors increases over time. For MEMS, the impact of Angle random walk or bias errors due to uncompensated temperature fluctuations or initial bias estimation is usually the most critical sources of errors [1].

## 2.5 Sensor fusion

The fusion of MEMS sensors is necessary to achieve better performance at the system level in terms of accuracy, resolution, stability and response time. It takes advantage of different and complementary information coming from various sensors and uses a set of adaptive algorithms for prediction and filtering [8]. By combining the data provided by several sensors, a more accurate and reliable information can be collected than would be possible if the sensors were used individually. It can be compared to humans who are combining information provided by the five body senses (sight, smell, sound, taste and touch) to create a more dynamic model of the world.

For IMU's there are essentially two methods that are widely used to apply a sensor fusion.

1. Complementary filter
2. Kalman filter

### 2.5.1 Complementary filter

A complementary filter is a simple and common way to fuse the gyroscope and accelerometer to obtain accurate pitch, roll and yaw attitude outputs. It is developed in such a way that the strength of one sensor will be used to overcome the weaknesses of the other sensor, which is complementary to each other and hence the name of the filter. It consists of a low-pass filter for the accelerometer and a high-pass filter for the gyroscope [9]. The filter is easier to implement and easier to understand compared to a Kalman filter that is often used for inertial navigation system. The principles of the complementary filter can be observed in figure 5.

$$\theta = \beta * \theta_{gyro} + (1 - \beta) * \theta_{acc} \quad (19)$$

where  $\beta$  is the filter coefficient:

$$\beta = \frac{\text{time constant}}{\text{time constant} + \Delta T} \quad (20)$$

Where  $\Delta T$  is the sample rate of the IMU and the *time constant* is usually tuned based on practical experiments and analysis of the specific application.  $\theta$  is the value in terms of degrees. The *time constant* is usually between 2 and 6 seconds for aerospace and flight applications [16]. For the low-pass filter, the signals that are much longer than the time constant can pass the filter unaltered while the signals shorter than the time constant are filtered out. The opposite is also true for the high-pass filter.

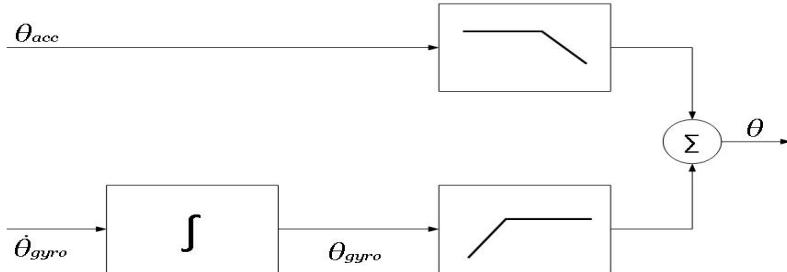


Figure 5. Block diagram of a simple sensor fusion-algorithm based on complementary filter.

### 2.5.2 Kalman filter

The theory and math behind the Kalman filter is beyond the scope of this thesis and can be explored in Welch's and Bishop's paper [11], but the basic idea behind the filter is that they are ideal for systems that are undergoing continuous changes. The filter is a set of mathematical equations that provides an efficient computational means to estimate the state of a process in a way that minimizes the mean squared error. It can support estimations of past, present and even future states. By taking into account the present estimated state and the information of the system constraints and dynamics, the prediction of the next state can be performed the next time step. Dealing with IMU's the process cannot be characterized as linear and therefore the Kalman filter needs to be extended [8]. In non-linear processes, the Extended Kalman filter (EKF) is therefore used.

### 2.5.3 Complementary filter vs Kalman filter

Previous work that has been done on the subject of Complementary filters compared to Kalman filters shows that some researcher advocates the use of Kalman filters to be able to estimate orientation and rotation with reliable accuracy due to reduction of bias drift [8], [12], [13]. While other studies states that the mathematical algorithm of Kalman Filter is too complex due to long computational time and states that the angle estimation actually can be achieved using fewer sensors and much simpler algorithms [14], [15]. In a study 2016 they concluded they could measure desired position level of  $0^\circ$  to  $100^\circ$ , consistently without any issues due to bias drift of the sensors with the help of a complementary filter [9]. Another study concludes that the complementary filter could achieve attitude filtering performance of the same quality as a full extended Kalman filter [10]. The Complementary filter has been proven to be faster than Kalman filters and therefore the power consumption is also lower [10], [17], [18], which is an important factor for battery powered devices.



### 3 Methodology

True angle is based on motion of short distances during short time intervals. The data from the gyroscope on a short term is more trustable than the long term and will not impact the bias drift significantly [18]. The implementation method of sensor fusion was decided to be complementary filter. Since the effort and complexity of Kalman filters would not provide that much of improvement in performance according to the studies presented in section 2.4.3. The use of complementary filter requires the input of the gyroscope's angular rate to be integrated. This to emerge an attitude angle before it feed into high pass filter to reduce the effect of bias drift.

True angle is based on motion, which can be directly translated into degrees. Therefore the data provided by the IMU-sensor will be converted to roll, pitch and yaw. The angular velocity and linear acceleration would not provide information that is directly applicable to true angle estimations for analyzing aspects of STB tools. Depending on the position of how and where the IMU is fixed on the tools, the orientation of the sensor will impact which axes in the local frame the orientation occurs around. The function of true angle is to detect small angle displacements caused by the operator during a tightening and compensate for that. During a tightening when the tool experiences a certain torque, it switches to another tightening program. The program switching can be used to initialize the IMU sensor at the present orientation and from there be able to detect angular displacement of  $\pm 30^\circ$ . Importance is not the orientation in relation to the real world environment instead the angular displacement from an initialized position. Euler angles was the choice of implementation over quaternions since the complexity of total 3D orientation is not of interest and would provide a more complex implementation and makes maintenance work more difficult in the future. Careful observations and analysis about gimbal lock has to be made to verify the consequences.

To be able to study the orientation of a nutrunner, practical experiments were introduced where an IMU-sensor was fixed on a battery powered nutrunner. For the experiments and evaluation, a LSM6DSM MEMS sensor fabricated by STMicroelectronics were used. The raw data of the gyroscope and accelerometer needed to be converted from 16-bit data to angular velocity respective linear acceleration. This was done by integration in relation to the sample rate and the sensitivity of the sensor, which can further be explored in the datasheet in appendix A. The nutrunner tools have powerful batteries that are charged regularly and therefore the power consumption is not an important factor considering the IMU-sensor. The sample rate should be the highest possible to ensure highest accuracy. The sensor was configured with the highest sample rate of 6,66 kHz for all measurements. The  $\beta$  filter coefficient was set to 0,95 as a starting point, which was based on the presentation by Esfandyari, De Nuccio and Xu in their article [8].

#### 3.1 IMU calibration

The gyroscope had a constant bias that was integrated for every time step and drifted significantly over time. The constraints of the  $\pm 3$  dps for zero-rate level was used to remove the bias drift from the static analysis. The same was done to the accelerometer with constraints of  $\pm 40$  mg for zero-g level to remove bias constant

errors when no external forces impacted the sensor. To remove the constant bias error when both the gyroscope and accelerometer was experience a change in linear acceleration or angular velocity, further calibration was made. 100 samples were collected for every axis and averaged to make a bias offset based on the article presented by Esfandyari, De Nuccio and Xu in [19]:

$$\text{Gyroscope bias offset}(X, Y, Z) = \sum_{i=1}^{100} (\text{angular velocity}(X_i, Y_i, Z_i)) / 100 \quad (21)$$

$$\text{Accelerometer bias offset}(X, Y, Z) = \sum_{i=1}^{100} (\text{linear acceleration}(X_i, Y_i, Z_i)) / 100 \quad (22)$$

The bias offset was used to calibrate the readings of angular velocity or linear acceleration as close to 0 as possible by subtracting the offset from the readings of the IMU.

$$\text{Angular velocity}(X, Y, Z) - \text{Gyroscope bias offset}(X, Y, Z) \quad (23)$$

$$\text{Linear acceleration}(X, Y, Z) - \text{Accelerometer bias offset}(X, Y, Z) \quad (24)$$

The purpose of the experiment was to achieve high accuracy estimation of the positioning angle by implementing digital complementary filter. This to be able to analyze angular displacement.

The practical experiments was categorized into two different sections:

1. Static analysis
2. Dynamic analysis

The yaw cannot be measured by the same principles as the pitch and roll, because of the non-existent gravity vector as reference. The accelerometer is not capable of measure the yaw and can therefore not be used to determine an angle, thus a sensor fusion is not possible. The gyroscope can be used to measure yaw, but not in the relation to the surrounding environment. The gyroscope will always be initialized to 0 dps independent of the orientation of the sensor. Therefore the yaw will be 0° and can only measure the angle in reference of the local frame and not according to any angle in relation of the surrounding environment. Although, a sensor fusion to measure yaw is still possible to perform with a gyroscope and a magnetometer in fusion.

The data from the IMU was sent to a computer for logging through a SPI-bus.

### 3.2 Static analysis

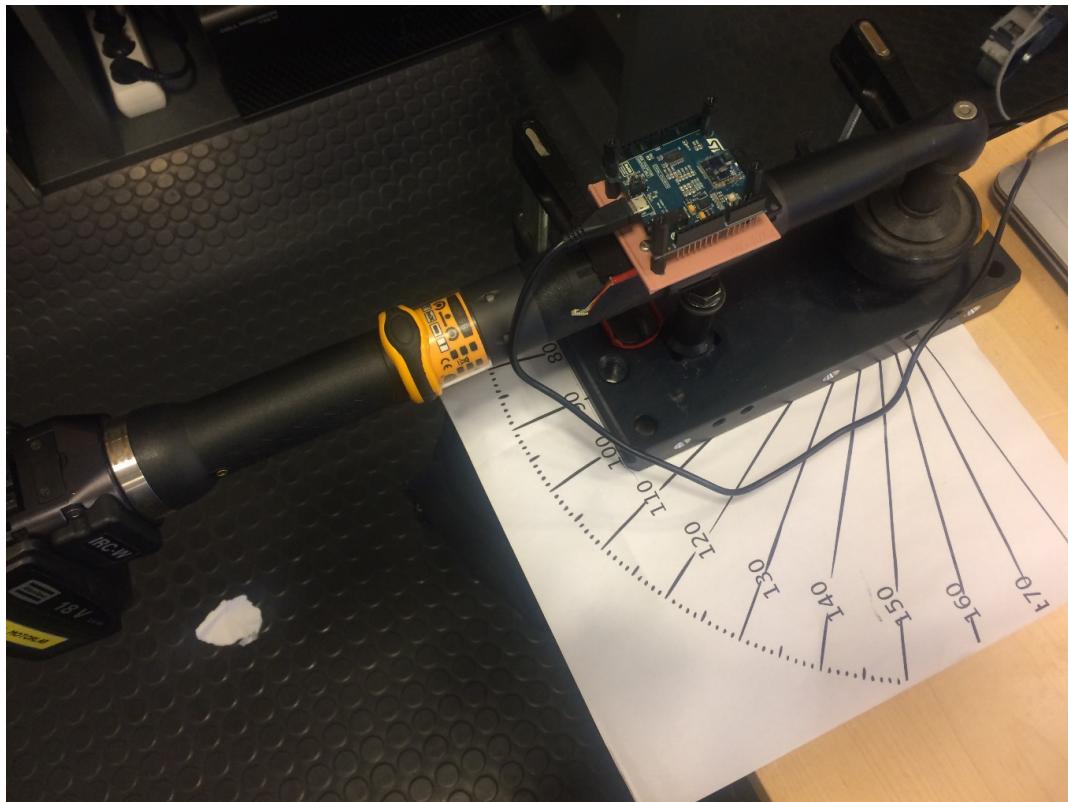
The static analysis was made to verify the precision and stability in terms of degrees (roll, pitch, yaw) from each axes of the sensor when the IMU-sensor was in a steady state position where no rotations or physical acceleration was introduced. The measurements can be observed in figures 8, 9, 10, 11, 12, 13, 14, 15, 16 in section 4.1. The measurements were made at 0° and ±30° rotation around every axis for one minute to be able to determine the accuracy and stability that covers the entire angular resolution of true angle. The accelerometer was configured with the lowest range of ±2g, which means that 16-bit resolution is defined in the range of ±2g and

has a zero-g level offset accuracy of  $\pm 40$  mg. The gyroscope was configured with  $\pm 2000$  dps and have a zero-rate level bias offset of  $\pm 3$  dps.

These experiments were performed to verify the stability and accuracy of the IMU to be able to proceed to the dynamic experiments without performance issues.

### 3.3 Dynamic analysis

The dynamic analysis consisted of experiments where the IMU-sensor was fixed on the STB tools during tightenings. The setup of the experiments can be observed in figure 6 and 7. The ETV STB is an angle type of tool and the sensor was fixed in a position that it measures yaw when rotations is introduced around the tightening axis, see figure 6. The roll and pitch was not possible to measure during a tightening of the ETV STB tool. The ETP STB is a pistol type of tool and the sensor was fixed in a position that it measures roll when rotations is introduced around the tightening axis. The yaw and pitch was not possible to measure during a tightening of the ETP STB tool, see figure 7. During the dynamic experiments, the pitch was therefore never measured and could not be obtained for the dynamic analysis. The setup consisted of a mechanical fixed frame that made it possible for the tools to perform an angular displacement of  $10^\circ$  before it collided into the frame, see figure 7.



*Figure 6. The setup of the experiment of ETV STB tool.*



Figure 7. The setup of the experiment of ETP STB tool.

### 3.3.1 Tightening program STB angle

The tightening program that were used for the experiments of the ETV STB tool consisted of two steps. The first step was to tighten the bolt until the tool experienced a torque of 7,00 Nm and then switched to step two. Step two was based purely on angle, where the driver turns additional 90°.

### 3.3.2 Tightening program STB pistol

The tightening program that were used for the experiments of the ETP STB tool also consisted of two steps. The first step was to tighten the bolt until the tool experienced a torque of 6,00 Nm and then switched to step two. Step two was based purely on angle, where the driver turns additional 20°.

### 3.3.3 Design of the experiments

The gyroscope provided data as angular velocity in the form of milli degrees per second (mdps) and to be able to obtain the angular displacement, an integration has to be invoked.

$$\text{Angular displacement} = \text{Angular displacement} + \text{angular velocity} * dt \quad (25)$$

Where  $dt$  is the sample period of the gyroscope ( $\frac{1}{6,66 \text{ kHz}}$ ).

The experiments of the dynamic analysis was designed to make an angular displacement of the tool by  $10.00^\circ$  during a tightening and then analyze the readings of the sensor to verify the accuracy. The experiment was designed to be performed in four different configurations:

- |                       |                     |
|-----------------------|---------------------|
| 1. Accelerometer: 2g  | Gyroscope: 2000 dps |
| 2. Accelerometer: 2g  | Gyroscope: 125 dps  |
| 3. Accelerometer: 16g | Gyroscope: 2000 dps |
| 4. Accelerometer: 16g | Gyroscope: 125 dps  |

This was to analyze the maximum and minimum parameters of respective sensor in every combination to evaluate which configuration provide most accuracy during a tightening operation.

For the yaw experiment, only two different configurations was made, since it only uses the gyroscope:

Gyroscope: 2000 dps

Gyroscope: 125 dps

Every configuration was performed with 30 tightenings to collect significant amount of data to be able to do a statistical analysis of the accuracy of the sensor. If gimbal lock would occur when the X-axis is in line with the gravity vector, no further experiments will be performed with different configurations of the accelerometer.

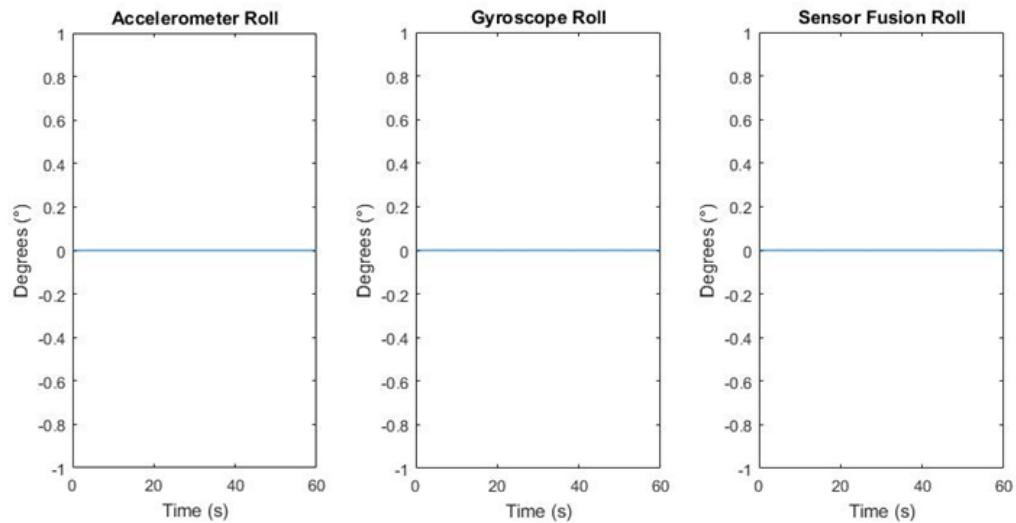
A statistical analysis was perform to verify which configuration provided data with highest probability for measurement within the range of  $9^\circ$  to  $11^\circ$  ( $\pm 1^\circ$ ) of 30 measurements of respective configuration. According to Armin Halilovic, Associate professor of mathematics at Royal Institute of Technology in Sweden, 30 samples of every configuration should be collected to be able to perform analysis with adequate reliability.



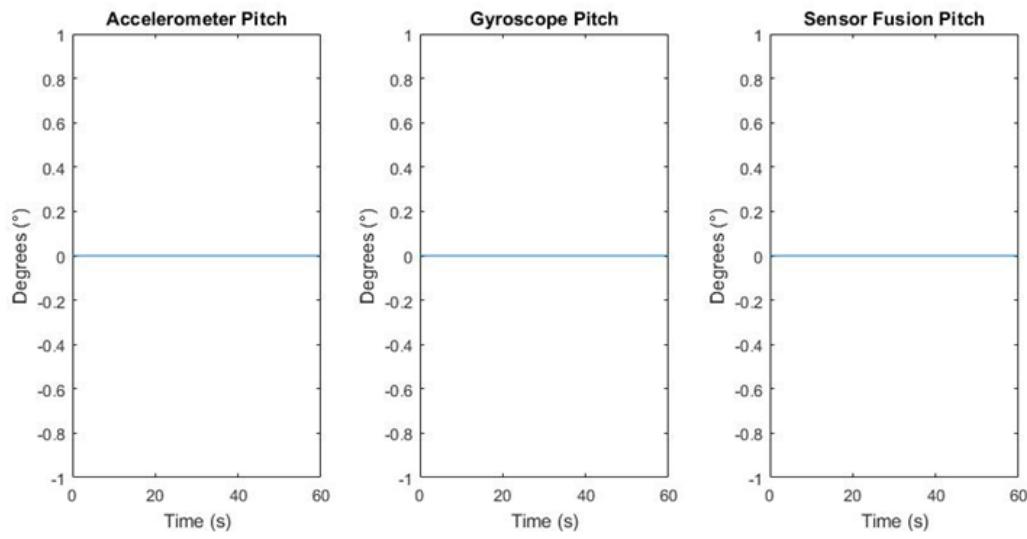
## 4 Results

### 4.1 Result of static experiments

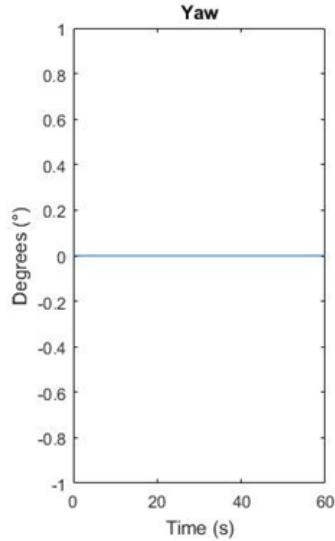
The results of the static analysis is presented in figure 8, 9, 10, 11, 12, 13 ,14 ,15, 16 where the measurements took place for one minute with  $0^\circ$  and  $\pm 30^\circ$  to verify stability and accuracy.



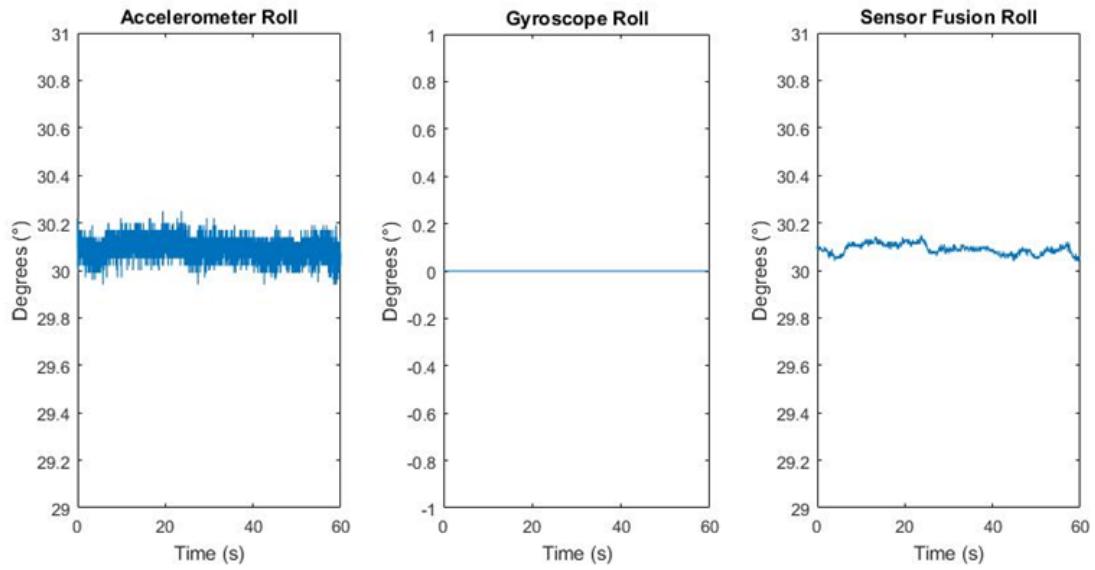
*Figure 8. Static measurements of the rotation around the X-axis (roll) for one minute when no rotation was introduced. It can be observed that the gyroscope did not drift due to the scaling of zero-rate level of  $\pm 3$  dps and accelerometer did not fluctuate due to the scaling of zero-g level offset accuracy of  $\pm 40$  mg. The sensor fusion is stable around  $0^\circ$  due to no changes of acceleration or angular velocity.*



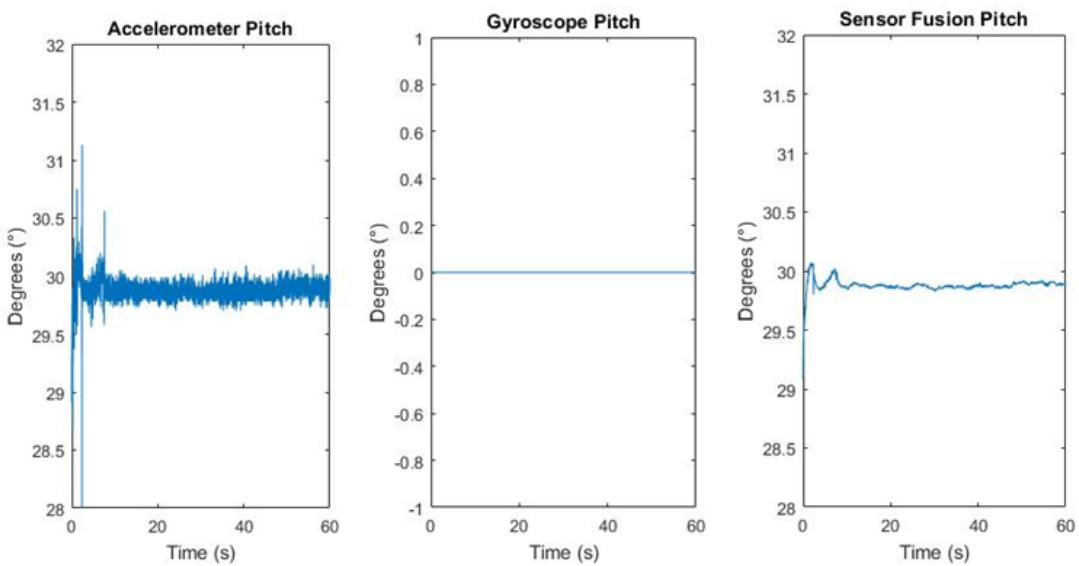
*Figur 9. Static measurements of the rotation around the Y-axis (pitch) for one minute when no rotation was introduced. It can be observed that the gyroscope did not drift due to the scaling of zero-rate level of  $\pm 3$  dps and accelerometer did not fluctuate due to the scaling of zero-g level offset accuracy of  $\pm 40$  mg. The sensor fusion is stable around  $0^\circ$  due to no changes of acceleration or angular velocity.*



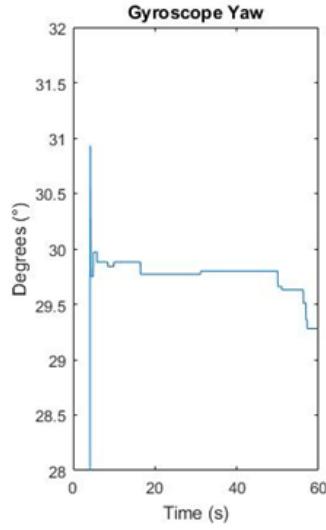
*Figur 10. Static measurements of the rotation around the Z-axis (yaw) for one minute when no rotation was introduced. It can be observed that the gyroscope did not drift due to the scaling of zero-rate level of  $\pm 3$  dps. A sensor fusion was not necessary to perform since the accelerometer could not be used.*



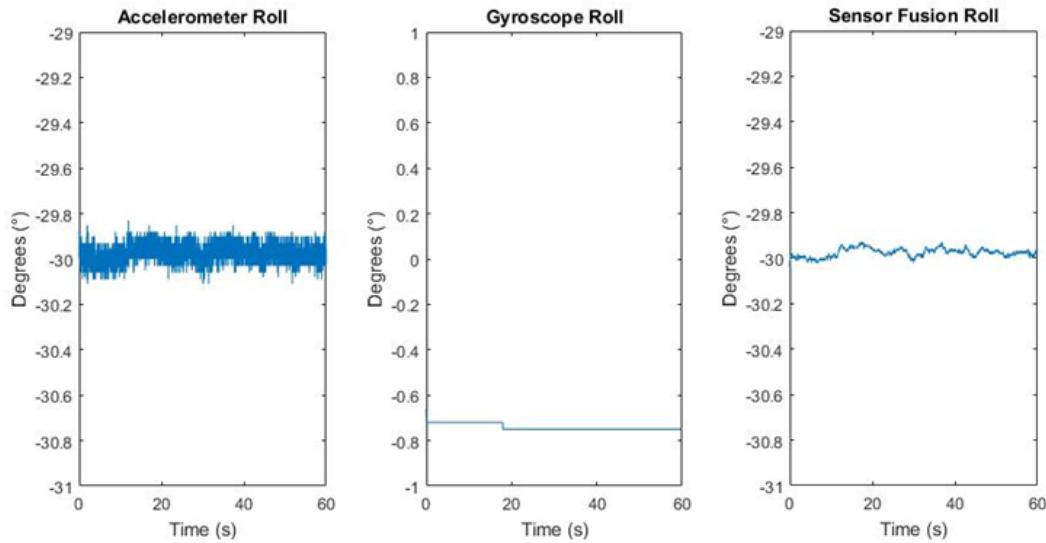
*Figure 11.* Static measurements of the rotation around the X-axis (roll) for one minute when a tilt of  $30^\circ$  was introduced. It can be observed that the gyroscope did not drift due to the scaling of zero-rate level of  $\pm 3$  dps but the accelerometer did fluctuate due to the external forces caused by the tilt. The sensor fusion smoothens out the fluctuations caused by the accelerometer and provides a more stable roll data.



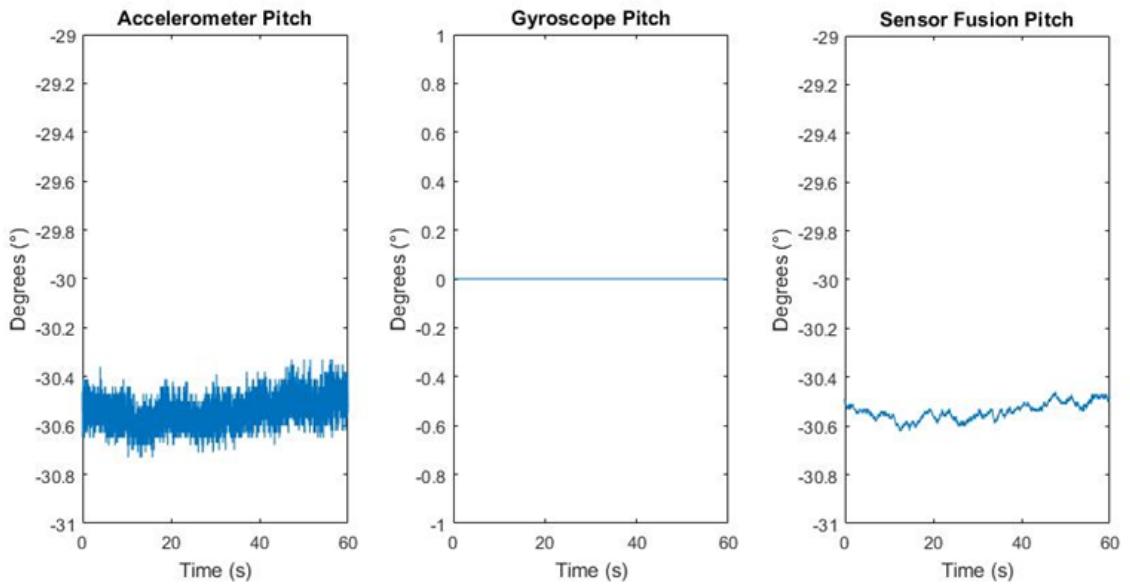
*Figure 12.* Static measurements of the rotation around the Y-axis (pitch) for one minute when a tilt of  $30^\circ$  was introduced. It can be observed that the gyroscope did not drift due to the scaling of zero-rate level of  $\pm 3$  dps but the accelerometer did fluctuate due to the external forces caused by the tilt. The sensor fusion smoothens out the fluctuations caused by the accelerometer and provides a more stable pitch data.



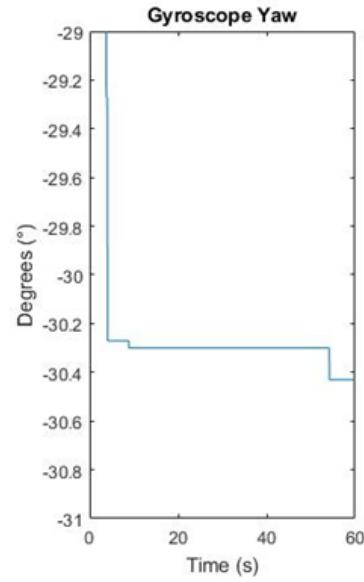
*Figure 13. Static measurements of the rotation around the Z-axis (yaw) for one minute when  $30^\circ$  of rotation was introduced. It can be observed that the gyroscope has smaller fluctuations due to the small constant bias error that are integrated. A sensor fusion was not necessary to perform since the accelerometer could not be used.*



*Figure 14. Static measurements of the rotation around the X-axis (roll) for one minute when a tilt of  $-30^\circ$  was introduced. It can be observed that the gyroscope had small angular displacement due to external impact that occurred during measurement. The accelerometer did fluctuate due to the external forces caused by the tilt. The sensor fusion smoothens out the fluctuations caused by the accelerometer and provides a more stable roll data.*



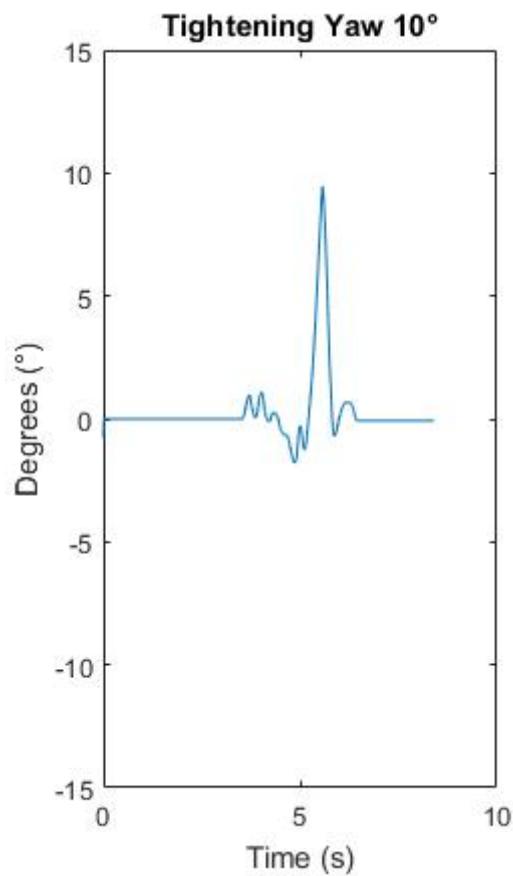
*Figure 15.* Static measurements of the rotation around the Y-axis (pitch) for one minute when a tilt of  $-30^\circ$  was introduced. It can be observed that the gyroscope did not drift due to the scaling of zero-rate level of  $\pm 3$  dps but the accelerometer did fluctuate due to the external forces caused by the tilt. The sensor fusion smoothens out the fluctuations caused by the accelerometer and provides a more stable pitch data.



*Figure 16.* Static measurements of the rotation around the Z-axis (yaw) for one minute when  $-30^\circ$  of rotation was introduced. It can be observed that the gyroscope has smaller fluctuations due to the small constant bias error that are integrated. A sensor fusion was not necessary to perform since the accelerometer could not be used.

## 4.2 Results of dynamic experiments

Analysis of experiments where the IMU-sensor was fixed on a STB angle and STB pistol tool during a tightening to verify the accuracy of the accelerometer and gyroscope in their different configurations. In figure 17, a sample of a tightening with a STB angle tool where yaw was the angular displacement to obtain. In figure 18, a sample of a tightening with a STB angle tool where roll was the angular displacement to obtain. A total of 30 tightenings in every configuration was made based on the information provided in section 3.3.3.



*Figure 17. Illustration of a yaw measurement during a tightening of  $10^\circ$  with a STB tool. A total of 30 tightenings were made.*

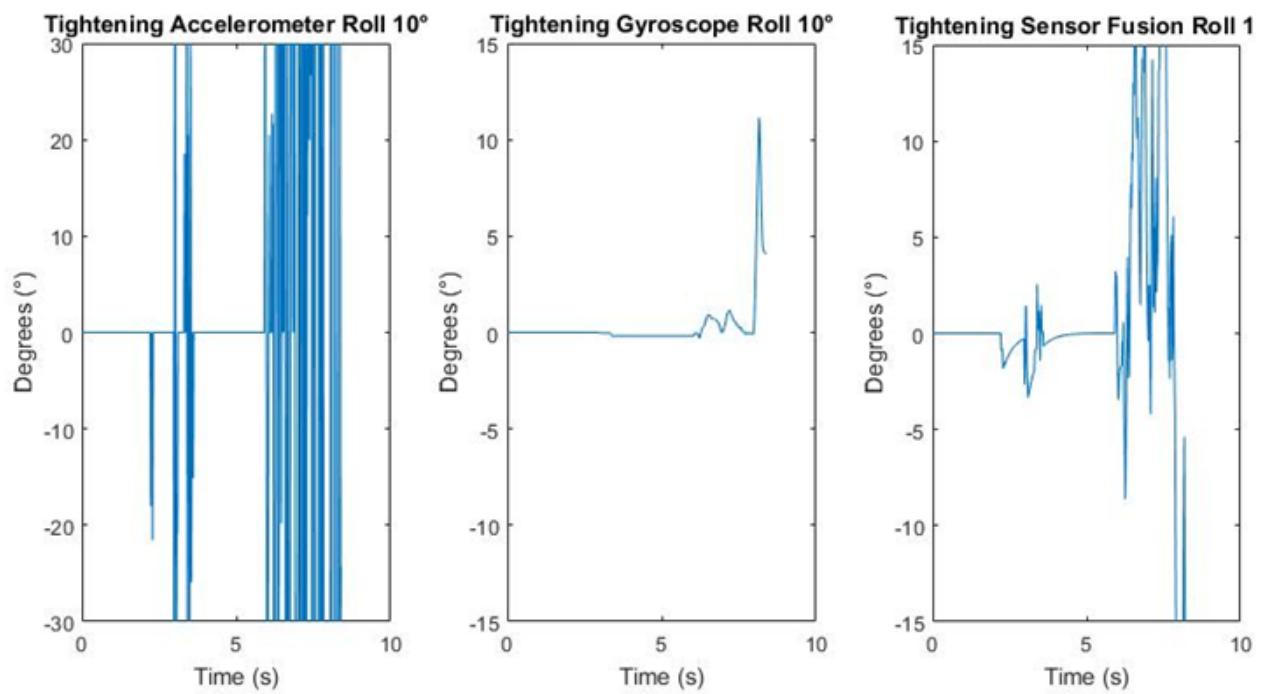


Figure 18. Illustration of a roll measurement during a tightening of  $10^\circ$  with a STB pistol tool. A total of 30 tightenings were made.

All tightening with different configurations are presented in tables in appendix B.



## 5 Analysis and Discussion

Full source code of the implementation of the IMU can be examined in appendix C.

### 5.1 Static analysis

To be able to measure the angular displacement of the yaw, the sensor needed to be turned  $\pm 30^\circ$  from its initial position which was always  $0^\circ$  independent of the orientation of the sensor, see figure 13 and 16. The transient in the beginning of the measurement can be observed during the turning of the sensor and it cannot be categorized as a pure static measurement since rotation was involved.

The accelerometer was implemented with the constraints of  $\pm 40$  mg at zero-g level and therefore it did not result in any fluctuations during static measurements of  $0^\circ$  as can be observed for both yaw, roll and pitch in figures 8,9 and 10. As soon as tilt or rotation was performed on the sensor, the constraints of zero-g level did not have any impact on the sensor and therefore fluctuations during measurements of static  $\pm 30^\circ$  was introduced on the accelerometer which can be observed in figures 11-16. Even though calibration was made to reduce the offset of the gyroscope and accelerometer there are still small fluctuations around zero-g level and zero-rate level which will make the sensor more sensitive to angular displacement in a certain rotation. If for example the gyroscope outputs a zero-rate level of -100 it needs less rotation in the negative direction for the angular displacement to increase. The sensor fusion reduces the fluctuations provided by the accelerometer and the accuracy and stability by the IMU is within a range of  $\pm 1^\circ$ .

Depending on the application that the IMU is used in, calibration of the accelerometer can be adapted. Further calibration could have been made to the accelerometer to make it suit tightening applications which can be explored in the *Parameters and calibration of a low-g 3-axis accelerometer*, Application note AN4508, by STMicroelectronics [20].

### 5.2 Dynamic analysis

During the dynamic experiments of roll, it was noticed that the sensor fusion was not going to work since the accelerometer introduces gimbal lock if the X-axis is aligned with the gravity vector when rotations around that axis was performed, see figure 18. Since the sensor fusion could not be used no further filter coefficients  $\beta$  was evaluated and no further configurations of the accelerometer was tested but  $2g$ . The roll could instead be measured by the same principles as the yaw for determine angular displacement. It was also noticed that the tool itself impacts the accelerometer axis as external forces in form of vibration. Therefore the accelerometer did not provide data with high reliability, see figure 18. Instead, the true angle function should only involve the gyroscope since it is not sensitive to vibration and external forces in the same manner as the accelerometer, see figure 18 of roll measurement. If the gyroscope is the only sensor that is used, there is no need to apply quaternions over Euler angles to remove the phenomena of gimbal lock, because the accelerometer has to be excluded from the measurements of angular displacement.

Since the accelerometer readings has to be excluded from the function of true angle, there is no need to perform a sensor fusion. Neither complementary filter nor Kalman filter would be possible to use since the gyroscope is the only sensor that can be used because it is not sensitive to external forces and vibration.

### 5.3 Statistical analysis of the tightenings

A statistical analysis was performed to verify which configuration that provided data with highest probability for measurement within the range of  $9^\circ$  to  $11^\circ$  ( $\pm 1^\circ$ ) of 30 measurements of respective configuration. The data was based only by the readings of the gyroscope.

The average of every sample in respective configuration was calculated:

$$\mu = \frac{1}{n} \sum_{i=1}^n x_i \quad (26)$$

The standard deviation was calculated:

$$\sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \mu)^2} \quad (27)$$

Thereafter, the probability of obtaining measurements between  $9,00^\circ$  and  $11,00^\circ$  according to the formula of normal distribution:

$$P(9 < X < 11) = F(11) - F(9) \quad (28)$$

$$= \Phi\left(\frac{11,00 - \mu}{\sigma}\right) - \Phi\left(\frac{9,00 - \mu}{\sigma}\right) \quad (29)$$

*Table 1. Compilation of the statistical analysis*

Sensitivity	$\mu$	$\sigma$	$P$
Yaw 125 dps	10,66667	0,632664	0,6976
Yaw 2000 dps	11,06793	0,766683	0,4606
Roll 125 dps	10,99867	1,181378	0,4545
Roll 2000 dps	13,612	1,36711	0,02832

The conclusion of the statistical analysis is that the configuration of the lowest sensitivity of 125 dps provides most accurate readings, see table 1. The probability that the angular displacement of yaw measures between 9,00° and 11,00° in future tightenings is 69,76% and 45,45% for roll measurements. A reason why the STB pistol tool provided a lower accuracy might have been that the tool made a faster motion and collided with the setup frame with harder impact than the STB angle tool. Although, without the frame it would be hard to verify the accuracy by determine the angular displacement in free space. In industrial applications, the tool will not be mounted in a fixed frame and therefore never collide with a frame to impact the IMU. The probability of 98,07% for yaw measurements with 125 dps configuration is achieved between 12° and 9°. The probability of 98,00% for roll measurements with 125 dps configuration is achieved between 13,5° and 7,7°.

During the tightenings when the tools experienced a torque of 7,0 Nm for STB or 6,0 Nm for STB pistol, it made an angular displacement in the positive direction of roll and yaw if the operator did not withstand the force of the torque. The measurements were therefore only based on the orientation in the positive direction. But it can be assumed that the readings in the negative direction of the roll or yaw would be equally accurate. If the IMU is mounted in another position of the tool it depends on the direction of the sensor axes to determine if it is roll, pitch or yaw to be measured when used in a tightening.

The achieved results is only based on the tightening program that were used for the experiments. Different tightening programs might impact the gyroscope in another way and has to be considered. If the probability that the readings will be  $\pm 1^\circ$  in relation to a predetermined value during tightenings is higher, better quality control can be assured from industries that are users of Atlas Copco nutrunners in their production.

#### **5.4 Economic and environmental aspects**

Compared to other newly produced MEMS sensors on the market, the LSM6DSM manufactured by STMicroelectronics is one of the cheapest due to its price about 3\$ and would therefore be favored by Atlas Copco for mass production of tools if desired accuracy can be achieved. Implementation of true angle in STB tools would ensure higher quality of tools by Atlas Copco which can provide a broader custom base in the future. Better quality can also be assured by the industries that uses the tools which can make them more competitive on the market.

The legal requirements that exist for the tools are that the batteries to be collected. Atlas Copco has environmental management systems according to ISO 14001 implemented in a large part of the organization. The implementation of true angle might not have an environmental impact globally but as industries will be able to ensure quality productions it can lead to a world with lesser manufacturing errors and quality issues.



## 6 Conclusion

Using the accelerometer to estimate the angular displacement is hard and might not work in the application of tightenings. The tool vibrates too much during a tightening making the accelerometer provide irrelevant data that makes the angular displacement not match with the reality. Therefore the use of sensor fusion in form of complementary filter or Kalman filter is not suitable for true angle functionality. The gyroscope by itself can be used to provide the angular displacement around every axis with high accuracy without take into account the gimbal lock phenomena.

The gyroscope provided data with an probability to measure  $\pm 1^\circ$  in future tightenings by 69,76% for yaw measurements and 45,45% for roll measurements. Before implementation of true angle into the STB tools, it has to be evaluated more and reach a higher probability of measure  $\pm 1^\circ$  in future tightenings.

The evaluation of MEMS based IMU has proven to work to determine the angular displacement in tightening applications. Although, it needs more experiments and analyzing to optimize the using of the IMU to work with high reliability in real world applications of tightenings. Implement further filtering by software or use the inbuilt hardware filters should be the next step for evaluation of the gyroscope in tightening applications.

### 6.1 Future work

If the IMU in the future would be mounted inside the tool, it has to be analyzed if there is going to be any heat that will have an impact on the angular rate sensitivity or zero-rate level on the gyroscope. The vibration caused by the tool itself with different tightening programs will also need to be analyzed carefully and may place the sensor on rubber cushion to remove as much vibration as possible.

To improve the accuracy and readings by the gyroscope for the specific true angle application, further investigation of the properties and parameters that can be configured in software for the gyroscope can be made. STMicroelectronics provide drivers that offer the user the ability to configure the IMU in a multitude of configurations.



## 7 References

- [1] Woodman, O. J. (2007). An introduction to inertial navigation, University of Cambridge. UCAM-CL-TR-696, ISSN 1476-2986
- [2] M. Perlmutter, L. Robin, (2012). "High-performance, Low Cost Inertial MEMS: a Market in Motion!" IEEE, 978-1-4673-0387-3/12.
- [3] M. Pedley. (march, 2013). Tilt Sensing Using a Three-Axis Accelerometer. Free-scale Semiconductor. Rev. 6, AN3461
- [4] Won, S. H. P., et al. (2009). "A Fastening Tool Tracking System Using an IMU and a Position Sensor With Kalman Filters and a Fuzzy Expert System." IEEE Transactions on Industrial Electronics **56**(5): 1782-1792.
- [5] Won. S, Melek. W, Golnaraghi. F, (2009), "A Fastened Bold Tracking System for a Hand-held Tool Using an Inertial Measurement Unit and a Triaxial Magnetometer", IEEE, 978-1-4244-4649-0
- [6] Bagchi, S. (2016). A Study on Positioning for Industrial Tools using Inertial Sensors. School of Industrial Engineering and Management. Stockholm, Kungliga Tekniska Högskolan.
- [7] R. Rasoulzadeh, A. M. S. (2016). Implementation of A Low- Cost Multi- IMU Hardware by Using a Homogenous Multi-Sensor Fusion. 2016 4th International Conference on Control, Instrumentation, and Automation (ICCIA). Qazvin Islamic Azad University, Qazvin, Iran.
- [8] J. Esfandyari, R. D. N., G. Xu. "Solutions for MEMS sensor fusion." from [http://www.mouser.se/applications/sensor\\_solutions\\_mem/](http://www.mouser.se/applications/sensor_solutions_mem/).  
[Retrieved 2017-04-25]
- [9] N. H. Ariffin, B. Bais., N. Arsal (2016). Low Cost MEMS Gyroscope and Accelerometer Implementation Without Kalman Filter For Angle Estimation. International Conference on Advances in Electrical, Electronic and System Engineering. Putrajaya, Malaysia. 978-1-5090-2889-4/16 IEEE.
- [10] T. Walter, JR. Higgins, (May, 1975). "A comparison of complementary and kalman filtering." IEEE Tranactions On Aerospace And Electronic Systems Vol. AES-11, No. (3).
- [11] Welch. G, Bishop. G. (2006). An Introduction to the Kalman Filter. University of North Carolina at Chapel Hill, Chapel Hill, NC 27599-3175.
- [12] G. Yanning, H. Fei, D. Shaohe, M. Guangfu and Z. Liangkuan, (2015) "Performance Analysis of MEMS Gyro and Improvement using Kalman Filter," IEEE 34th Chinese Control Conference, July, pp. 1934- 1768.

- [13] M.T. Leccadito, (2013), “An Attitude Heading Reference System using a Low Cost Inertial Measurement Unit,” M.S. thesis, Dept. Elect. and Computer Eng., Virginia Commonwealth University, Richmond, Virginia, August.
- [14] B. McCarron, (2013), “Low-Cost IMU Implementation via Sensor Fusion Algorithms in the Arduino Environment,” Bachelor of Science, Department of the Aerospace Engineering, Polytechnic State University, San Luis Obispo, California, June.
- [15] Pascoal, I. Kaminer and P. Oliveira, (2000), “Navigation System Design using Time-Varying Complementary Filters,” IEEE Transactions on Aerospace and Electronic Systems, vol. 36, October, pp. 1099- 1114.
- [16] A Complementary Filter for Attitude Estimation of a Fixed-Wing UAV  
Mark Euston, Paul Coote, Robert Mahony, Jonghyuk Kim and Tarek Hamel
- [17] Anthony, Y.M. and W.T. Ang, (September, 2004) “An Efficient Real-Time Human Posture Tracking Algorithm using Low-Cost Inertial and Magnetic Sensors,” Proceeding of IEEE International Conference on Intelligent Robots and Systems, vol. 34, pp. 2967- 2972
- [18] P. Gui, L. Tang., S. Mukhopadhyay (2015). MEMS based IMU for tilting measurement - comparison of complementary and Kalman filter based data fusion IEEE 10th Conference on Industrial Electronics and Applications (ICIEA), 978-1-4799-8389-6/15
- [19] J. Esfandyari, R. De Nuccio, G. Xu, (November 15, 2010), ” Introduction to MEMS gyroscopes”, STmicroelectronics,  
<http://electroiq.com/blog/2010/11/introduction-to-mems-gyroscopes/>.  
[Retrieved 2017-05-15]
- [20] STMicroelectronics, (June, 2014), “Parameters and calibration of a low-g 3-axis accelerometer”, Application note AN4508, DocID026444 Rev 1

## Appendix

### A. Datasheet of the LSM6DSM sensor

Symbol	Parameter	Test conditions	Min.	Typ. <sup>(1)</sup>	Max.	Unit
LA_FS	Linear acceleration measurement range			±2		<i>g</i>
				±4		
				±8		
				±16		
G_FS	Angular rate measurement range			±125		dps
				±245		
				±500		
				±1000		
				±2000		
LA_So	Linear acceleration sensitivity <sup>(2)</sup>	FS = ±2		0.061		mg/LSB
		FS = ±4		0.122		
		FS = ±8		0.244		
		FS = ±16		0.488		
G_So	Angular rate sensitivity <sup>(2)</sup>	FS = ±125		4.375		mdps/LSB
		FS = ±245		8.75		
		FS = ±500		17.50		
		FS = ±1000		35		
		FS = ±2000		70		
G_So%	Sensitivity tolerance <sup>(3)</sup>	at component level		±1		%
LA_SoDr	Linear acceleration sensitivity change vs. temperature <sup>(4)</sup>	from -40° to +85°		±0.01		%/°C
G_SoDr	Angular rate sensitivity change vs. temperature <sup>(4)</sup>	from -40° to +85°		±0.007		%/°C
LA_TyOff	Linear acceleration zero-g level offset accuracy <sup>(5)</sup>			±40		mg
G_TyOff	Angular rate zero-rate level <sup>(5)</sup>			±3		dps
LA_OffDr	Linear acceleration zero-g level change vs. temperature <sup>(4)</sup>			±0.1		mg/ °C
G_OffDr	Angular rate typical zero-rate level change vs. temperature <sup>(4)</sup>			±0.015		dps/°C

Symbol	Parameter	Test conditions	Min.	Typ. <sup>(1)</sup>	Max.	Unit
Rn	Rate noise density in high-performance mode <sup>(6)</sup>			3.8		mdps/ $\sqrt{\text{Hz}}$
RnRMS	Gyroscope RMS noise in normal/low-power mode <sup>(7)</sup>			75		mdps
An	Acceleration noise density in high-performance mode <sup>(8)</sup>	FS = $\pm 2 \text{ g}$		90		$\mu\text{g}/\sqrt{\text{Hz}}$
		FS = $\pm 4 \text{ g}$		90		
		FS = $\pm 8 \text{ g}$		90		
		FS = $\pm 16 \text{ g}$		130		
RMS	Acceleration RMS noise in normal/low-power mode <sup>(9)(10)</sup>	FS = $\pm 2 \text{ g}$		1.8		mg(RMS)
		FS = $\pm 4 \text{ g}$		2.0		
		FS = $\pm 8 \text{ g}$		2.4		
		FS = $\pm 16 \text{ g}$		3.0		
LA_ODR	Linear acceleration output data rate			1.6 <sup>(11)</sup> 12.5 26 52 104 208 416 833 1666 3332 6664		Hz
G_ODR	Angular rate output data rate			12.5 26 52 104 208 416 833 1666 3332 6664		
Vst	Linear acceleration self-test output change <sup>(12)(13)(14)</sup>		90		1700	mg
	Angular rate self-test output change <sup>(15)(16)</sup>	FS = 245 dps	20		80	dps
		FS = 2000 dps	150		700	dps
Top	Operating temperature range		-40		+85	°C

## B. Tightenings

Yaw 2000 dps

Nr of tightenings	Expected angular displacement	Tightening angular displacement
1	10°	11,83°
2	10°	11,87°
3	10°	10,62°
4	10°	12,37°
5	10°	10,14°
6	10°	10,19°
7	10°	11,34°
8	10°	11,99°
9	10°	10,70°
10	10°	11,17°
11	10°	10,93°
12	10°	10,96°
13	10°	10,36°
14	10°	12,41°
15	10°	11,27°
16	10°	11,00°
17	10°	10,58°
18	10°	11,28°
19	10°	10,18°
20	10°	11,47°
21	10°	9,23°
22	10°	11,20°
23	10°	11,34°
24	10°	11,55°
25	10°	10,77°
26	10°	9,95°
27	10°	12,30°
28	10°	11,20°
29	10°	11,30°
30	10°	10,43°

Yaw 125 dps

Nr of tightenings	Expected angular displacement	Tightening angular displacement
1	10°	9,46°
2	10°	11,16°
3	10°	10,44°
4	10°	10,53°
5	10°	10,73°
6	10°	10,98°
7	10°	10,23°
8	10°	9,61°
9	10°	9,60°
10	10°	10,28°
11	10°	11,13°
12	10°	10,44°
13	10°	9,85°
14	10°	10,17°
15	10°	10,89°
16	10°	10,23°
17	10°	10,40°
18	10°	11,39°
19	10°	10,92°
20	10°	10,77°
21	10°	10,84°
22	10°	10,15°
23	10°	11,86°
24	10°	10,47°
25	10°	11,25°
26	10°	11,17°
27	10°	10,75°
28	10°	11,98°
29	10°	10,67°
30	10°	11,65°

Roll 125 dps

Nr of tightenings	Expected angular displacement	Tightening angular displacement
1	10°	9,24°
2	10°	10,45°
3	10°	8,93°
4	10°	8,52°
5	10°	11,16°
6	10°	12,04°
7	10°	9,91°
8	10°	12,93°
9	10°	11,98°
10	10°	12,46°
11	10°	9,28°
12	10°	11,07°
13	10°	11,50°
14	10°	12,08°
15	10°	11,78°
16	10°	12,14°
17	10°	9,90°
18	10°	11,21°
19	10°	12,79°
20	10°	11,45°
21	10°	10,64°
22	10°	10,86°
23	10°	10,63°
24	10°	10,92°
25	10°	12,33°
26	10°	10,82°
27	10°	11,36°
28	10°	11,99°
29	10°	10,28°
30	10°	9,31°

Roll 2000 dps

Nr of tightenings	Expected angular displacement	Tightening angular displacement
1	10°	12,15°
2	10°	12,91°
3	10°	13,21°
4	10°	15,61°
5	10°	12,36°
6	10°	12,19°
7	10°	13,34°
8	10°	14,11°
9	10°	12,84°
10	10°	14,45°
11	10°	12,54°
12	10°	14,73°
13	10°	15,27°
14	10°	14,77°
15	10°	10,45°
16	10°	11,31°
17	10°	12,70°
18	10°	14,00°
19	10°	15,01°
20	10°	14,18°
21	10°	12,56°
22	10°	14,04°
23	10°	15,13°
24	10°	14,47°
25	10°	14,69°
26	10°	14,38°
27	10°	10,99°
28	10°	13,87°
29	10°	15,02°
30	10°	15,08°

### C. Code used for experiments

```
void Sensor_Handler( void *handleGyro, void *handleAcc)
{
    uint8_t who_am_i;
    float odr;
    float fullScale;
    uint8_t id;
    uint8_t idAcc;
    SensorAxes_t angular_velocity;
    SensorAxes_t acceleration;
    uint8_t status;
    uint8_t statusAcc;
    int32_t d1, d2;
    float gyrZ = 0;
    float gyrX = 0;
    float gyrY = 0;
    static int counter = 0;
    static int test = 0;
    static float angle = 0;
    static float yaw = 0;
    static float fusionRoll = 0;
    static float fusionPitch = 0;
    static float gyroAngle = 0;
    float rollAcc;
    static float pitchAcc = 0;
    static float rollAngle;
    static float pitchAngle;
    static float gyroAngleX = 0;
    static float gyroAngleY = 0;
    static float gyroAngleZ = 0;
    static int calibration_counter = 0;
    static int calibration_arrayZ[100];
    static int calibration_arrayX[100];
    static int calibration_arrayY[100];
    static int gyroZOffset = 0;
    static int gyroXOffset = 0;
    static int gyroYOffset = 0;
    static int test2 = 0;
    static int counter2 = 0;
```

```

/*******************************/
***** ACCELEROMETER TEST *****
BSP_ACCELERO_Get_Instance( handleAcc, &idAcc );

BSP_ACCELERO_IsInitialized( handleAcc, &statusAcc );

if ( statusAcc == 1 )
{
    if (BSP_ACCELERO_Get_Axes( handleAcc, &acceleration ) == COMPONENT_ERROR )
    {
        acceleration.AXIS_X = 0;
        acceleration.AXIS_Y = 0;
        acceleration.AXIS_Z = 0;
    }

    if(acceleration.AXIS_Y < 40 && acceleration.AXIS_Y > -40)
    {
        acceleration.AXIS_Y = 0;
    }

    if(acceleration.AXIS_Z < 40 && acceleration.AXIS_Z > -40)
    {
        acceleration.AXIS_Z = 0;
    }

    if(acceleration.AXIS_X < 40 && acceleration.AXIS_X > -40)
    {
        acceleration.AXIS_X = 0;
    }
}

rollAcc = atan2(-(float)acceleration.AXIS_X, (float)acceleration.AXIS_Z)*180/M_PI;

pitchAcc
atan2((float)acceleration.AXIS_Y,sqrt((float)acceleration.AXIS_X*(float)acceleration.AXIS_X
(float)acceleration.AXIS_Z*(float)acceleration.AXIS_Z ))*180/M_PI;
=+
***** /
```

```

BSP_GYRO_Get_Instance( handleGyro, &id );

BSP_GYRO_IsInitialized( handleGyro, &status );

if( status == 1 )
{
    if( BSP_GYRO_Get_Axes( handleGyro, &angular_velocity ) == COMPONENT_ERROR )
    {
        angular_velocity.AXIS_X = 0;
        angular_velocity.AXIS_Y = 0;
        angular_velocity.AXIS_Z = 0;
    }
}

/*
 * Counter to remove the transients that appear during startup of the sensor, before calibration
 * takes place
 */
if(calibration_counter < 100 && counter2 >= 10)
{
    calibration_arrayZ[calibration_counter] = angular_velocity.AXIS_Z;
    calibration_arrayX[calibration_counter] = angular_velocity.AXIS_X;
    calibration_arrayY[calibration_counter] = angular_velocity.AXIS_Y;
}
if(calibration_counter == 100 )
{
    for(int i = 0; i < 100; i++)
    {
        gyroZOffset += calibration_arrayZ[i];
        gyroXOffset += calibration_arrayX[i];
        gyroYOffset += calibration_arrayY[i];
    }

    gyroZOffset /= 100;
    gyroXOffset /= 100;
    gyroYOffset /= 100;
}

/*
 * Counter to remove the transients that appear during startup of the sensor, before calibration
 * takes place
 */
if(calibration_counter > 100)
{
}

```

```

counter2 = 100;
angular_velocity.AXIS_Z -= gyroZOffset;
angular_velocity.AXIS_X -= gyroXOffset;
angular_velocity.AXIS_Y -= gyroYOffset;

// Constraints for zero-rate level of 3 dps
if(((angular_velocity.AXIS_X >= 3000) || (angular_velocity.AXIS_X
<= -3000)) && counter >= 10)
{
    gyrX = (float)angular_velocity.AXIS_X / 1000;
}
else
    gyrX = 0;

if(((angular_velocity.AXIS_Y >= 3000) || (angular_velocity.AXIS_Y
<= -3000)) && counter >= 10)
{
    gyrY = (float)angular_velocity.AXIS_Y / 1000;
}
else
    gyrY = 0;

if(((angular_velocity.AXIS_Z >= 3000) || (angular_velocity.AXIS_Z
<= -3000)) && counter >= 10)
{
    gyrZ = (float)angular_velocity.AXIS_Z / 1000;
}

}

else
    gyrZ = 0;

counter++;
if(counter >= 10)
    counter = 10;

/*****************/
// SENSOR FUSION ROLL

gyroAngleY = gyroAngleY + (gyrY * dt);
gyroAngleX = gyroAngleX + (gyrX * dt);

yaw = yaw + gyrZ*dt;
fusionRoll = (fusionRoll + (gyrY * dt)) * 0.95 + rollAcc * 0.05;

```

```

fusionPitch = (fusionPitch + (gyrX * dt)) * 0.95 + pitchAcc * 0.05;

/***********************/

//BSP_GYRO_Get_ODR( handleAcc, &odr );
//floatToInt( odr, &d1, &d2, 3 );

// sprintf( dataOut, "FS[%d]: %d.%03d dps\n", (int)id, (int)d1, (int)d2 );

if(SendOverUSB) // Write data on the USB
{
    // fusionRoll, fusioPitch, Yaw
    //sprintf( dataOut, "\n\r%d, %d, %d",
(int)(fusionRoll*100), (int)(fusionPitch*100), (int)(yaw*100));

    // roll, pitch, yaw BARA GYRO
    //sprintf( dataOut, "\n\r%d, %d, %d",
(int)(gyroAngleY*100), (int)(gyroAngleX*100), (int)(yaw*100));

    // fusionRoll, accRoll, gyroRoll
    sprintf( dataOut, "\n\r%d, %d, %d, %d,
%d", (int)(gyroAngleY*100) , (int)(rollAcc*100),(int)(fusionRoll*100) , (int)(gyrY*dt*100),
(int)(angular_velocity.AXIS_Y));

    // fusionPitch, accPitch, gyroPitch
    //sprintf( dataOut, "\n\r%d, %d, %d, %d",
(int)(fusionPitch*100), (int)(pitchAcc*100), (int)(gyroAngleX*100),
(int)(angular_velocity.AXIS_X));

    // yaw
    //sprintf( dataOut, "\n\r%d, %d, %d", (int)(yaw*100),
(int)(gyrZ*dt), (int)(angular_velocity.AXIS_Z));

    //sprintf( dataOut, "ODR[%d]: %d.%03d Hz\n",
(int)id, (int)d1, (int)d2 );

    //FILE *f = fopen("c:\\test.txt", "w");
    //printf("%f\n", gyroAngle);
    //fprintf(f, "%f\n", gyroAngle);
    CDC_Fill_Buffer(( uint8_t * )dataOut, strlen( dataOut ));

    //fclose(f);
    if ( verbose == 1 )
    {
        if ( BSP_GYRO_Get_WhoAmI( handleGyro,
&who_am_i ) == COMPONENT_ERROR )
        {
            sprintf( dataOut, "WHO AM I address[%d]: ER-
ROR\n", id );
        }
        else
        {
    
```

```

        sprintf(  dataOut, "WHO AM I address[%d]:\n",
ox%o2X\n", id, who_am_i );
    }

    CDC_Fill_Buffer(( uint8_t * )dataOut, strlen( dataOut
));

    if ( BSP_GYRO_Get_ODR( handleGyro, &odr ) ==
COMPONENT_ERROR )
{
    sprintf( dataOut, "ODR[%d]: ERROR\n", id );
}
else
{
    floatToInt( odr, &d1, &d2, 3 );
    sprintf( dataOut, "ODR[%d]: %d.%o3d Hz\n", (int)id,
(int)d1, (int)d2 );
}

    CDC_Fill_Buffer(( uint8_t * )dataOut, strlen( dataOut
));

    if ( BSP_GYRO_Get_FS( handleGyro, &fullScale ) ==
COMPONENT_ERROR )
{
    sprintf( dataOut, "FS[%d]: ERROR\n", id );
}
else
{
    floatToInt( fullScale, &d1, &d2, 3 );
    sprintf( dataOut, "FS[%d]: %d.%o3d dps\n", (int)id,
(int)d1, (int)d2 );
}

    CDC_Fill_Buffer(( uint8_t * )dataOut, strlen( dataOut
));
}

}

else if(SD_Log_Enabled) // Write data to the file on the SDCard
{
    uint8_t size;
    size = sprintf(dataOut, "%d\t%d\t%d\t",
(int)angular_velocity.AXIS_X, (int)angular_velocity.AXIS_Y, (int)angular_velocity.AXIS_Z);
    res = f_write(&MyFile, dataOut, size, (void *)&byteswritten);
}

    calibration_counter = 110;
}
counter2++;
calibration_counter++;

}
}

```



TRITA STH 2017:115