Software Implementation & Integration

# Introduction

This document will describe the implementation of the design described in the document Software Design.

Naming/Java conventions

Since we use Java as our programming language we use the standard Java conventions in our code. These standard conventions include:

Proper indentation and proper use of curly brackets.

To keep the code clean and readable it is important to use proper indentation when nesting code.

Example:

*if( … ) {*

*while( … ) {*

*indentation;*

*}*

*indentation;*

*}*

As is visible we also use the standard conventions for the use of curly brackets, making loops and methods structurized.

Naming conventions

To clearly distinct variables, constants, methods and classes we use the Java conventions for naming them. Constants are written in uppercase letters with an underscore between the words.

Example:

*public static final int MOTOR\_TURN\_SPEED = 120;*

Variables are writing according to the lowerCamelCase convention, where the first word has a first lowercase letter and the other words are written with first upper case letter.

Example:

*public final GyroSensor gyroSensor;*

Methods are also written in lowerCamelCase convention:

*private void run(){*

*indentation;*

*}*

Classes are written according to the UpperCamelCase convention:

*public class MotorLeftState extends MotorState {*

*indentation;*

*}*

User manual

# Timer

Some warnings depend on time. The errors and warnings like gyro does not stabilize, disc did not reach basket and disc took longer than average to reach basket depend on time. In code this is implemented by a Timer class which gives the time between two method calls. However we need to set the thresholds in order to be able to decide whether or not an error or warning should occur. To do this we made a simple app, TimerApp which sorts the discs and measures the time between the moment that the motor started to turn and the time the disc fell onto the lever. The simple app then calculates the average, minimal time and maximum time.

# Conclusion