Stave Shipment Tests

Dino Tahirovic

January 18, 2018

Abstract

1 Introduction

Stave shipment.

2 System Description

3 Data Acquisition

Currently, I use direct read from MPU registers. I save the raw data as 10 bytes into a buffer, and after 256 entries, the buffer is saved to the SD card. There is also a script readData.cpp that converts binary to human readable format, which can be further analysed (I use MATLAB).

The improvement would be to use FIFO, so that there isnt 50 ms gap when the buffer is written to the SD card.

3.1 Code Installation

Please find the repository here: https://github.com/dinojugosloven/staveShipmentCond The code is in accelSDcard folder (other folders are just my training). After the installation of SparkFun libraries (as given on their site for Razor 14001), the Arduino file should compile.

3.2 Usage

1. Upload the accelSDcard Arduino code. That is the first step. I presume you have downloaded SparkFun and SD libraries for the Arduino and avr-gcc compiler, as advertised on SparkFun site.

2. Sensor sends data into the SD card and create a file in SD card to store raw data (real-time?) Correct.

3. Convert data from binary to human readable format by running readData.cpp Correct.

4. Analyze the converted data by MATLAB (Must date be converted? Can it be raw data? Do I need to write code in MATLAB? Is Excel okay to do the analysis?) Once you have the human readable data, it is up to you. Excel is good (ROOT too), I use Excel to preview the data for the correctness.