

---

# Testing the Encoder

## Introduction

This journal covers tests of the quadraturdecoder software the group made

## Considerations

According to the protocol, the encoder has to provide a position and a velocity for both the motor and the system as a whole. The data will be send to the Tiva, and the data format has to be 12 bit signed.

Previously tests regarding the encoder has shown an inconsistency with it's position counting. The problem is "drifting", where the position, in certain outer cases, would jump between two "ticks"<sup>1</sup> which translate to 1/12 of a degree. The porpuse of this journal is to oberserve, whether or not this error has been corrected, and in case it hasn't, if it even is a problem.

not sure this is relevant for the consideration

## Test specification and produce execute

The experiment is conducted by using the following equipment:

- Osicloscope
- 2x powersupplies
- a FPGA
- Pan tilt system
- PWM controller

For the first experiemnt to determine whether or not the decoder experience drifting, the following produce has been followed: First the setup has to be configured, with the PWM controller to 12V, the Hall sensor Vcc to 3.4V and the LOGIC\_VCC to 5V.

## Results

However, at first, there turned out to be a problem with the PWm controller and the encoder frequency. It had a disruptive influence on the encoder frequency in the sense that the frequency jumped between 24Hz to 500Hz, with the same motor velocity. This should not have happened. The frequency should be steady on one frequency for each motor velocity. Thereforth the it has been decided to make an experiemnt to figure out, what exactly it is, that is distrutive.

First the motor was set to 6V without the use of the PWM controller. This resulted in a frequency that was stable at 157Hz and outputtet an expected signal. It continued to do so with other motor velocities. However, as stated

---

<sup>1</sup>Exsample: If the encoder stops at postion 245, the position counter might jump between 245 and 246

---

above, if the mottor was controlled through the PWM controller, the frequency jumped and the signal was very noisy. We switched the frequency of the PWM and that worked.

Type	Test1	Test2	Test3	Test4
------	-------	-------	-------	-------

Secondly we had to check whether or not the software could handle

## **Conclusion**