

# Week 3 report

Javier Torres

January 16-20, 2017

## Contents

<b>1</b>	<b>Advance Status</b>	<b>1</b>
<b>2</b>	<b>Comments and Thoughts</b>	<b>1</b>
<b>3</b>	<b>Next Steps</b>	<b>2</b>

## 1 Advance Status

**Stepper motor** The problem with the motor controller was solved by using an A4988 chip (that was stored in the laboratory) instead of the Phidget board. There was no problem interfacing this chip with the MCU.

**Sensor bracket** The bracket was designed, printed and mounted with no problems. A more finished version of it will be produced next week.

**Code Scalability** The source code of the project was modified in order to achieve an easier scalability i.e. now the number of sensors, degrees per reading and motor steps can be modified by adjusting the corresponding parameter on the source code header file.

**Demonstration** In order to demonstrate and test all the components working together, the whole project was mounted on a breadboard. Only one issue was encountered, everything else worked properly.

## 2 Comments and Thoughts

**About the bracket prototype** Two different versions of the bracket were printed. Both with the "Draft" settings to evaluate their functionality. As one of them worked well for the project, a more finished version of it will be designed and printed.

**About the starting position** During the tests, a design problem was encountered: the system has no means to establish the starting position consistently. If the power is turned off in the middle of a scan, when the power is turned back on, the home position will be the point where it was turned off, e.g. if the power is turned off when the system is scanning in the 90 position, the next time It will scan from 90 to 270 instead of 0 to 180, potentially damaging the hardware. This situation should be resolved as soon as possible. A possible solution is to place a photo interrupter at the 0 position. By reading its status the MCU will have a means to determine if the sensor is in the home position.

**About the motor controller** Two possible solutions for the motor control problem were evaluated:

- Use an USB host chip to interface the MCU and Phidget (like the FTDI's VNC1L chip)
- Use a stepper motor controller chip (like the Allegro's A4988 chip)

Looking for any chip similar to the mentioned above, I found an A4988 on the lab, so I implemented that solution.

### 3 Next Steps

**Consistently setting the starting position** As mentioned in the "Comments and Thoughts" section, a solution for this problem will be implemented as soon as possible.

**PCB design** In order to have a more stable and compact solution, I will design a PCB to replace the circuit assembled on the breadboard.

**Prototype enclosure** Once all the electronic components are mounted on the PCB, I will start working on a case to mount everything.