Cover

Stewart Platform Prototype Description

in list Epics

Description Edit

Stewart Platform Prototype - Summer 2020

Aims

Before building the main test-rig a smaller prototype of a stewart platform would be nice to build and evaluate.

- Get a feel for how to control a stewart platform.
- Get to know tools to control it
  - MATLAB
    - Simulink model of SP (Stewart Platform)
    - PID-control of SP
      - Feedback from sensors
        - Accelerator
        - Gyroscope
    - Inverse-kinematics of SP
      - Accuracy of models
  - MATLAB Support Packages on RaspberryPi/Arduino
    - Usage and installation
  - MATLAB Simulink on Raspberry Pi /Arduino
  - Optional: Use Arduino instead of RPi
  - Optional: MATLAB GUI interface for manual control of SP

Goals

- Investigate the tools in MATLAB, RaspberryPi, (Arduino?)
  - Evaluate feasibility of tool on main test-rig.
  - Document how to use and integrate into main test-rig.
- Try out the math/modelling behind the controls
  - Simulated (using CAD-model and/or Simulink-model)

    - On the prototype (with feedback from accelerometer)
- Learn tools for tuning the controls (libraries in Matlab)
- Write code that can be reused on the main test-rig (if possible).
- Write test cases that can be reused for main rest-rig.
- Develop tools and routines for distance collaborative work.

Time-plan

1st sprint: 6-12th July

2nd sprint: 13-19 July

3nd sprint: 20-26 July

4th sprint: 27th July - 2nd August

5th sprint: 3-9th August

6th sprint: 10-16th August

7th sprint: 17-23 August

8th sprint: 24th-30 August

Project management

Github

HK-SOCKETSENSE-CODE

To get the Github repository

Mac/Linux:
- open terminal, cd into directory where you want to put the repository
- `git clone https://github.com/axelzedigh/HK-SOCKETSENSE-CODE.git`

To get newest changes from the github

- cd into the repository directory
- `git pull`

To work on code on your own branch

- cd into repository directory
- create new branch from master:
  `git branch master %your_name%_%date%_%time%__%issue%`
  e.g. `git branch master axel_20200706_0915__issue_1`

You will now have your own branch to work on.

To send code to github

- Checkout your branch: `git checkout %your_branch_name%`
- Add all new files/lines of code to future commit: `git add *`
- Commit (-m "Commit message") e.g. :
  `git commit -m "Message describing what I've done`

- Push code to Github: `git push`

Merge branches into master branch

Will be done manually by Axel (will review code before it is merged).

Trello

Socket Sense - Summer Project

- Add tasks to backlog

The build

"Stakeholder" Requirements

Shalls
- Use Raspberry Pi or Arduino Uno
- Use Matlab and it's support packages
- Stewart Platform in a portable size
- 20x20x20cm - 40x40x40cm (LxWxH) in total size
- Able to disassemble and send the prototype easily by mail
- Able to control each joint via laptop
- Able to control platform position (inverse kinematics)
- Able to calibrate and get feedback from the position using sensors
- Not not expensive

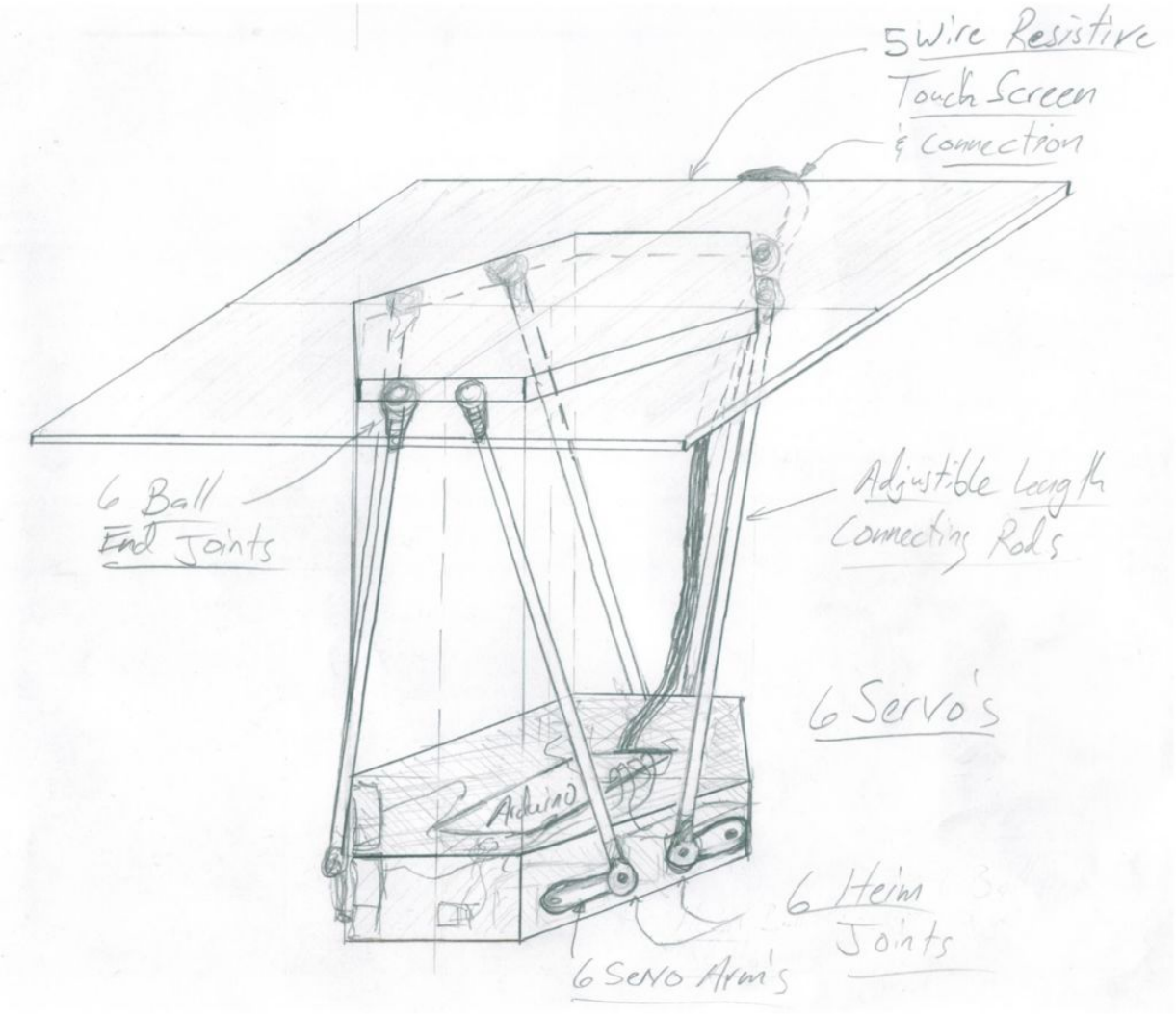
Would likes
- GUI interface in MATLAB where your're able to control platform
- CAD model of prototype (to be used when simulating control)
- Able to control platform via relays (instead via laptop)
- Alternative via keyboard/gamepad connected to microcontroller/laptop.
- Store and visualize positional data of servos and accelerometer using MATLAB and/or InfluxDB/Graphana.
- Possible to steer the prototype via internet (SSH and/or browser)

Other projects to base this on

The project is based on the following project:

PID Controlled Ball Balancing Stewart Platform

Sketch



The prototype will differ in that the positional sensing will be done with other sensor than touch screen.

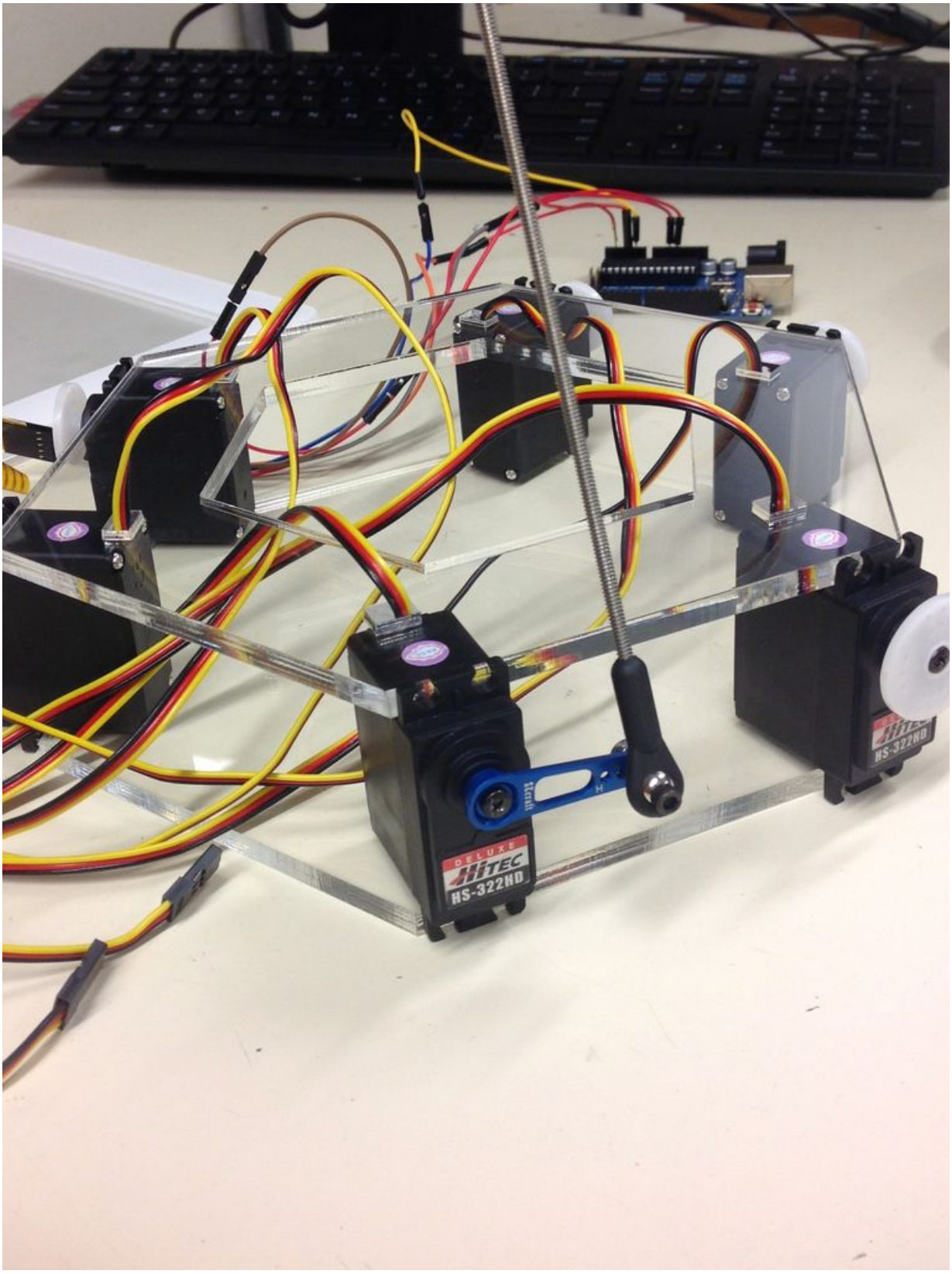
Material list

- A few sheets (thick enough of fastening servos to it) of either:
  - Acrylic
  - Plywood
  - 3D-printed parts
- 6 small servos (e.g. SM-S2309S)
- Servo arms with multiple holes for adjustability
- 6 threaded rods (adjustable)
- 12 end-connectors (rod ends/rod end bearings)
- Microcontroller (Pi or Arduino)
- Breadboard
- Electrical components (resistors, capacitors, inductors)
- Power source to servos, microcontroller.

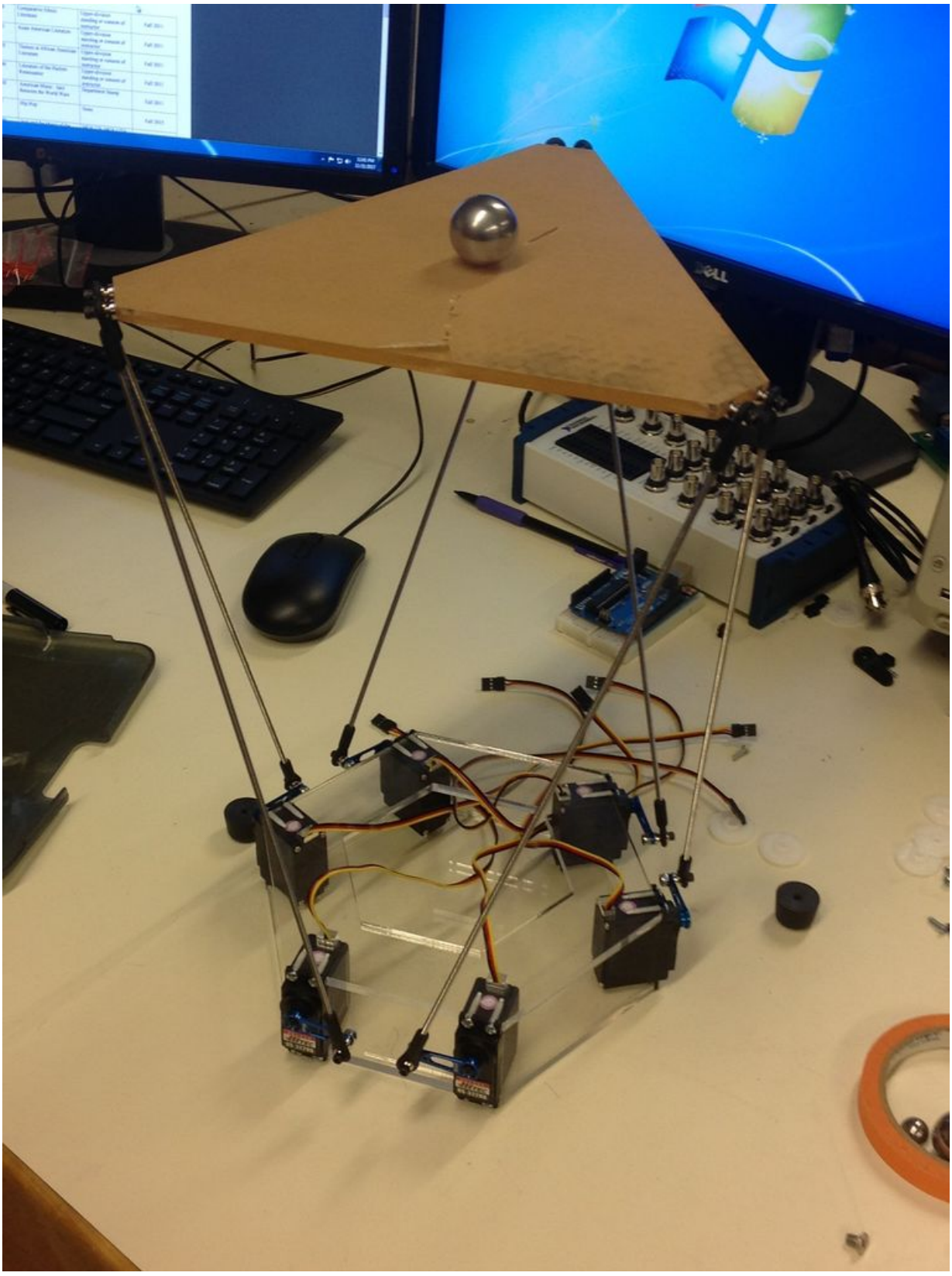
- Positional sensor (accelerometer/gyroscope)
- Screws

The Design

Servo base plates



Top platform



Phases

1. Planning

- Write requirements
- Write test cases
- Decide design of system

- Decide materials and components
- First model of system by hand and in Simulink/MATLAB

2. Get materials

- Where to get material and order them.

3. Prepare materials

- 3D-print, cut etc.

4. Assemble materials

- Use proper tools

5. Code

- Collaborate using github

6. Test and evaluate

- Run test cases. Sort out bugs.

7. Fine-tune

- Iterative process: tune, calibrate and redo previous processes if needed.

Attachments

Component list ↗
Comment - Remove

Created: 10 Jul at 09:57
Modified: just now by Axel Zedigh

Add an attachment

Activity

Show Details