NEST Docking System

Mississippi State University IMPRESS Lab

Center for Advanced Vehicular Systems

Table of Contents

System Overview

The Nest docking station is a system designed by Mississippi State University's Information Processing and Sensing lab. The Nest was designed to assist in performing regular, precise drone flights. These flights are necessary to gather accurate data for IMPRESS lab's smartphone GNSS program. The Nest system can be divided into its software and hardware components. The hardware system can be further divided into the following subsystems: structure, motion, electrical, and control. The software system can be divided into the following subsystems: server, app, and windows application.

NEST Docking Station

Hardware

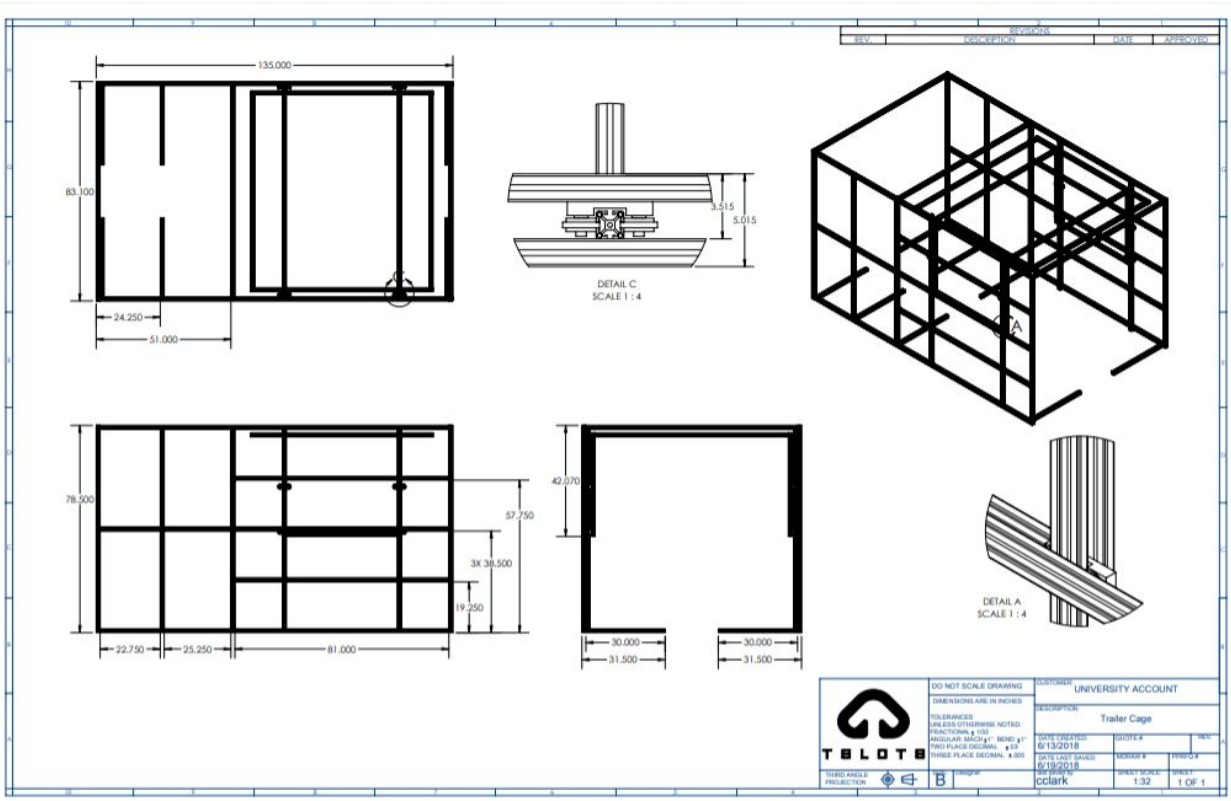
Structure

Trailer

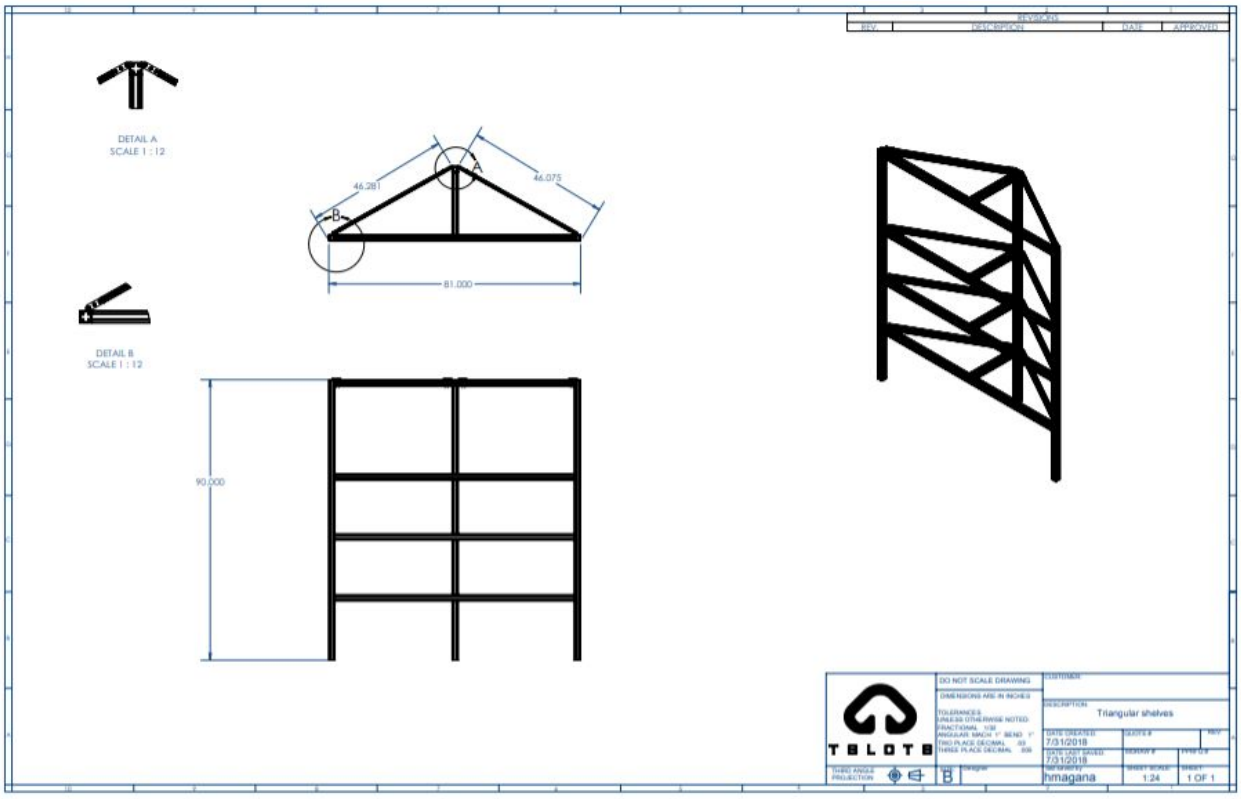
Information about the trailer initially bought and modified will go here. [1]

Cage

The interior of the trailer is reinforced by an aluminium cage. Other structural and motion components of the NEST are attached to the cage. A drawing for the cage is below.



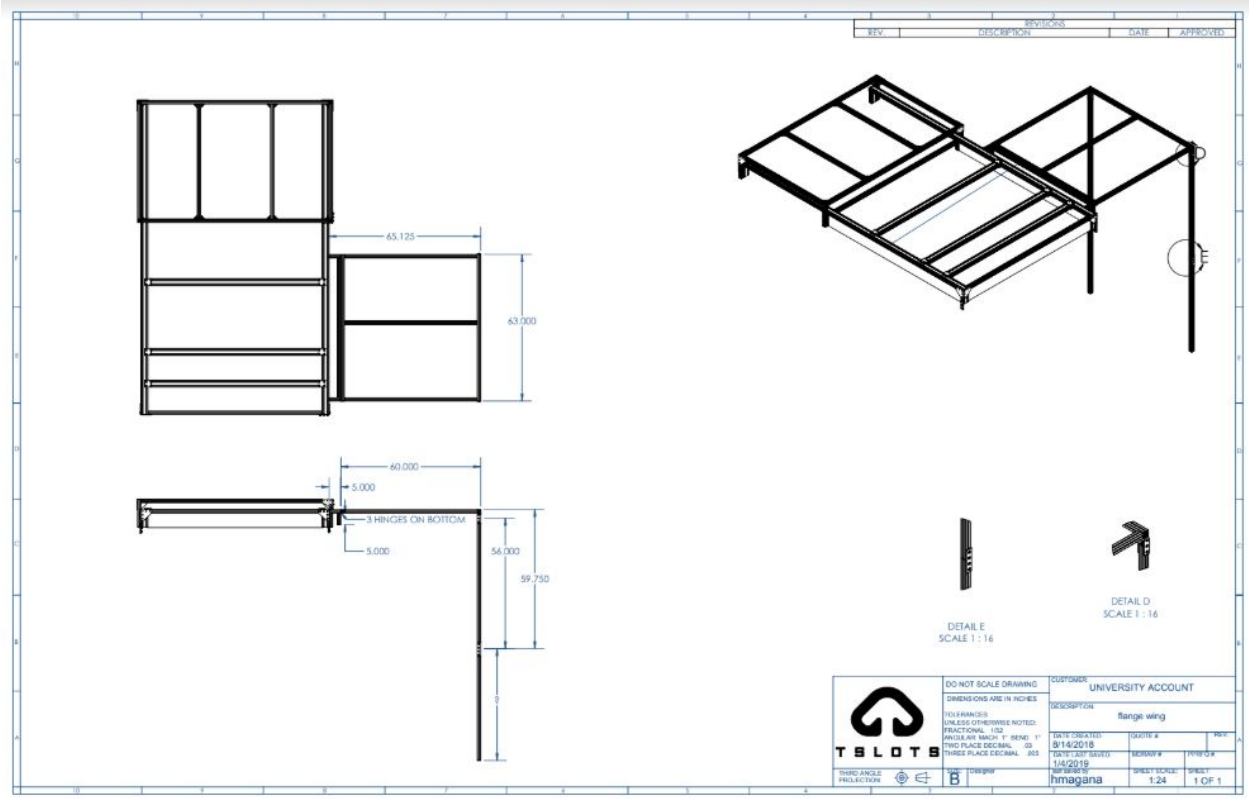
Cage Drawing



Shelves Drawing

Folding Solar Panel Mounts

Each side of the trailer will have a folding solar panel mount. These are constructed of an aluminium frame that is connected to the exterior portion of the cage near the roof. These frames are connected with hinges to that when the trailer is stationary they can be raised in order to generate more electricity. When the trailer needs to be moved, they can be folded down and secured so the trailer is safe to be taken onto roads.



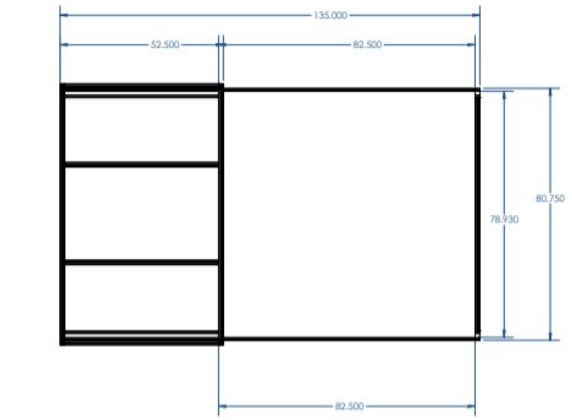
Solar Panel Mount Drawing

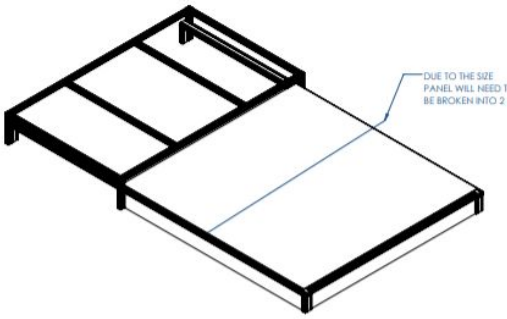
Motion

Roof

The opening roof system allows drones to take off and land through the roof of the trailer. The roof itself is mounted onto linear bearings that slide towards the front of the trailer. The driving motor is a ⅓ hp DC motor that transfers its power to the sprocket and chain mechanism that pushes and pulls the roof. Drawings for this system are below. [1]

REVISIONS			
REV.	DESCRIPTION	DATE	APPROVED





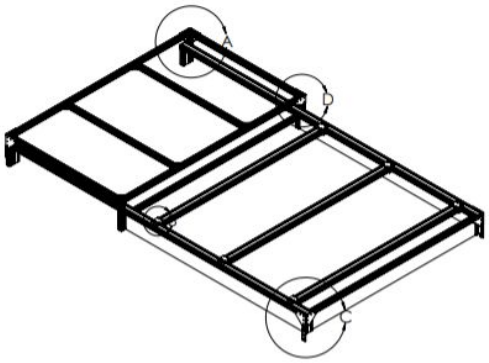
TBLOTB	DO NOT SCALE DRAWING DIMENSIONS ARE IN INCHES TOLERANCES UNLESS OTHERWISE NOTED: FRACTIONAL 1/32 ANGULAR: MACH 1° BEND 1° TWO PLACE DECIMAL .03 THREE PLACE DECIMAL .005	CUSTOMER: UNIVERSITY ACCOUNT DESCRIPTION: Trailer Cage WITH SLIDING DOOR DATE CREATED: 8/14/2018 QUOTE # REV: DATE LAST SAVED: 8/29/2018 MDRAW # PRINTED # DRAWN BY: hmagana SHEET SCALE: 1:24 SHEET: 1 OF 1	
THIRD ANGLE PROJECTION	SIZE: B Designer		

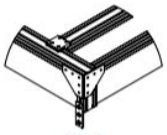
REVISIONS			
REV.	DESCRIPTION	DATE	APPROVED

DETAIL A
SCALE 1 : 12

DETAIL B
SCALE 1 : 12

DETAIL D
SCALE 1 : 12

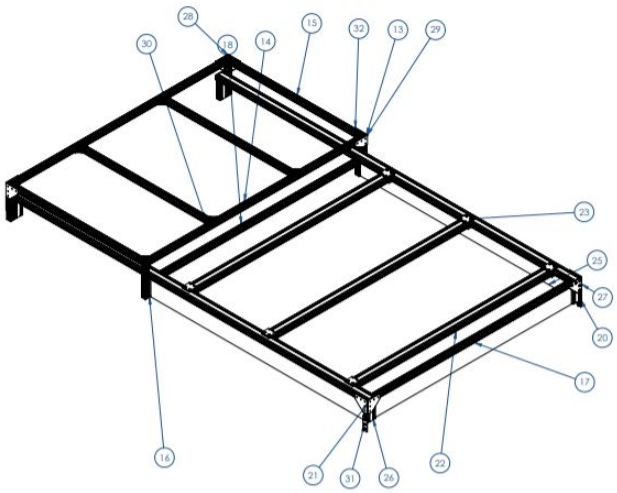




DETAIL C
SCALE 1 : 12

TBLOTB	DO NOT SCALE DRAWING DIMENSIONS ARE IN INCHES TOLERANCES UNLESS OTHERWISE NOTED: FRACTIONAL 1/32 ANGULAR: MACH 1° BEND 1° TWO PLACE DECIMAL .03 THREE PLACE DECIMAL .005	CUSTOMER: DESCRIPTION: Trailer Cage 2 DATE CREATED: 10/17/2018 QUOTE # REV: DATE LAST SAVED: 10/17/2018 MDRAW # PRINTED # DRAWN BY: hmagana SHEET SCALE: 1:24 SHEET: 1 OF 1	
THIRD ANGLE PROJECTION	SIZE: B Designer		

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	650006	BASE	3
2	650006	BASE	8
3	650006	BASE	4
4	650006	BASE	8
5	650006	BASE	4
6	650006	BASE	6
7	650006	BASE	2
8	650006	BASE	2
9	650006	BASE	4
10	650006	BASE	2
11	650006	BASE	4
12	650006	BASE	2
13	650006	BASE	4
14	650006	BASE	2
15	650006	BASE	4
16	650006	BASE	2
17	650006	BASE	1
18	650006	BASE	1
19	655478		8
20	650009	BASE	4
21	650009	BASE	2
22	650009	BASE	3
23	657156		6
24	panel top		1
25	655432		2
26	655432		1
27	653183		1
28	653183		9
29	653182		1
30	653165		12
31	653140		2
32	653182		1



REVISIONS

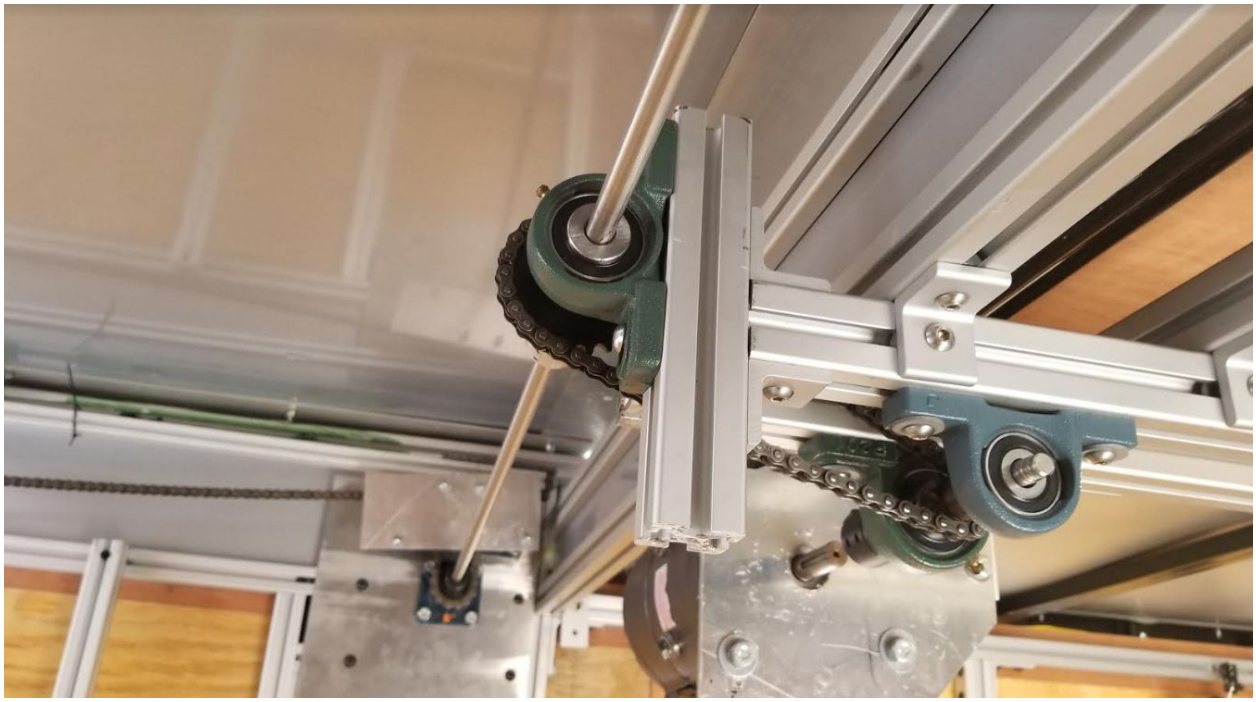
REV.	DESCRIPTION	DATE	APPROVED

DO NOT SCALE DRAWING
DIMENSIONS ARE IN INCHES
TOLERANCES
UNLESS OTHERWISE NOTED:
FRACTIONAL 1/32
ANGULAR: MACH ± BEND ±°
TWO PLACE DECIMAL .03
THREE PLACE DECIMAL .005
THIRD ANGLE PROJECTION
Designer
B

CUSTOMER:
DESCRIPTION: Trailer Cage
DATE CREATED: 10/17/2018
QUOTE #
REV:
DATE LAST SAVED: 10/17/2018
DRAWN #
PART(S) #
DRAWN BY: hmagana
SHEET SCALE: 1:24
SHEET: 1 OF 1

Roof Drawings

SPROCKET AND CHAIN MECHANISM



Sprocket Chain Mechanism

Doors

The door motion system allows for objects to move out of the back doors of the trailer. The system contains a 6 ft. long motor shaft that has a curved hinge mechanism on it. This hinge connects to a linear bearing on the doors that allow it to change position on the door as it turns. The hinge is attached to this bearing by a pin and cotter key so it may pivot. As the shaft turns, it pushes the door open or pulls it shut. Each door is driven by a 0.25 hp motor [2]. A drawing of this mechanism is below.

Door System

Lift

The lift system is the mechanism that allows the upper landing pad to become level with the trailer's roof. This system is driven by two pairs of Progressive Automation lifting columns [3]. These columns are given 5 V digital signals that tell it to raise or lower. A proximity switch will be used to sense when the lift is fully raised.

Extending Rail

The extending rail is the mechanism that allows the lower landing pad to be extended out of the trailer. This rail has wheels that slide in channels, allowing it to slide, while also being able to support the weight of the landing pad and drone. This mechanism is driven by a ⅛ hp DC motor [4].

Landing Pads

Pad Motor

The rotating pad is driven by the HT34-486 stepper motor. This bipolar phase stepper motor has a step angle of 1.8° [104]. This when paired with the drive system allows the motor to make fine rotation adjustments.

Pad Motor Driver

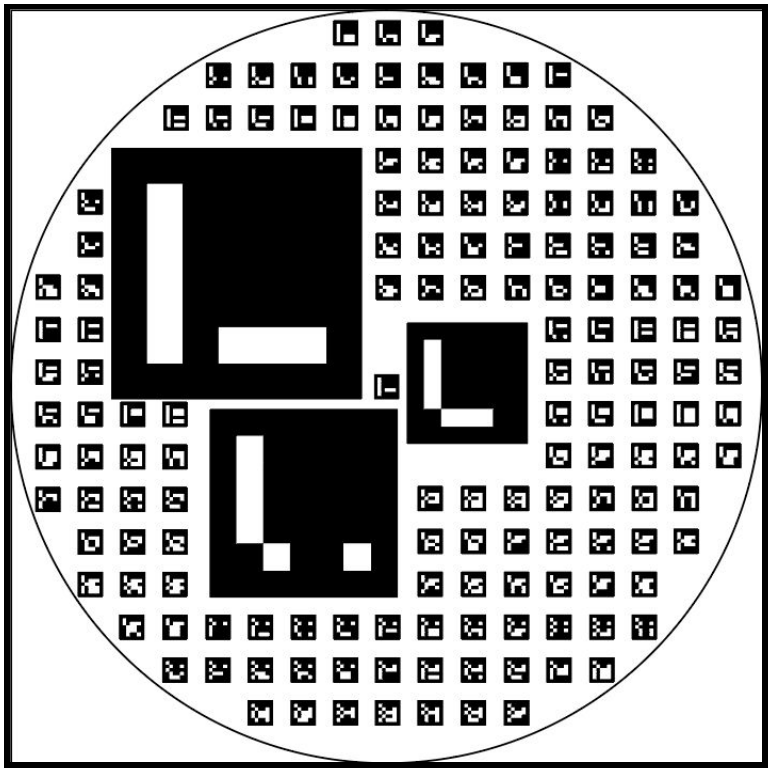
The landing pad motor driver is driven by the Si ST-10 motor driver [105]. This driver is programmed to accept commands from a man machine interface (MMI). The master computer runs a MMI emulator to send movement commands to the driver, which will also send feedback to the master computer.

Pad Motor Drive System

The landing pad drive system uses a simple timing belt pulley system to turn the motor. The motor turns the timing belt with a timing belt sprocket. This belt is tensioned onto the outside of the table. As the motor turns, the shaft sprocket and table create a large gear ratio, which allows the motor to turn the table very accurately with little torque.

Aruco Markers

Below is the design on the landing pad of the NEST.



Landing Pad - Aruco Markers

The design represents two different theories about how the drone will land--accurately in the center or with some error around the sides. Accordingly, the precise landing procedure is divided into two parts, a

landing script and a locating script. The large markers spiraling down to the center represent the landing part of the script. Those markers, and all the rest, represent the location part of the script.

Aruco Markers are 16 bits. OpenCV detects the corners of these bits and comprises them into IDs. The script can then tell where it is based on the ID that is returned by the script. Once the script has locked on to a marker, it can get positional data and orient itself accordingly. This data consists of both distance on the x, y, and z plane and gyroscopic data for yaw, pitch, and roll of the drone in relation to the pad.

Electrical

Batteries

The source used in the NEST system is four RB200 12V 200Ah batteries [5]. These batteries are charged by the NEST's solar panel array. A Relion battery indicator is also installed to monitor the battery bank. This will display the current and voltage readings [6].

Solar Panels

The NEST's generation system is a combination of 7 160 W solar panels [7]. Three panels are located on the roof and 2 on each of the folding mounts located on the side of the trailer.

Inverter

The NEST uses a 3000 W pure sine wave inverter to supply AC power to the computer, PLC, and upper landing pad lifts. The fuse kit that accompanies this uses 400A inverter fuses. [7]

Proximity Switches

The NEST uses inductive proximity switches to get feedback from the state of the machinery. These sensors switch from 12 V to 0 V when a piece of metal is in front of them. These are used in conjunction with steel tags in order to evaluate position [8].

Control

Master Computer

The master computer is the nerve center of the NEST. The master computer is responsible for all servers, control scripts, and data retrieval from the drone.

Programmed Logic Controller

The NEST uses a Siemens LOGO! PLC. This is a small PLC designed for small automation projects [9]. The PLC uses a mixture of proximity sensor and user inputs to safely drive motors. The PLC runs a web server that the master computer uses for control.

Battery Exchange System

Gantry

Robot Arm

Battery Accessories

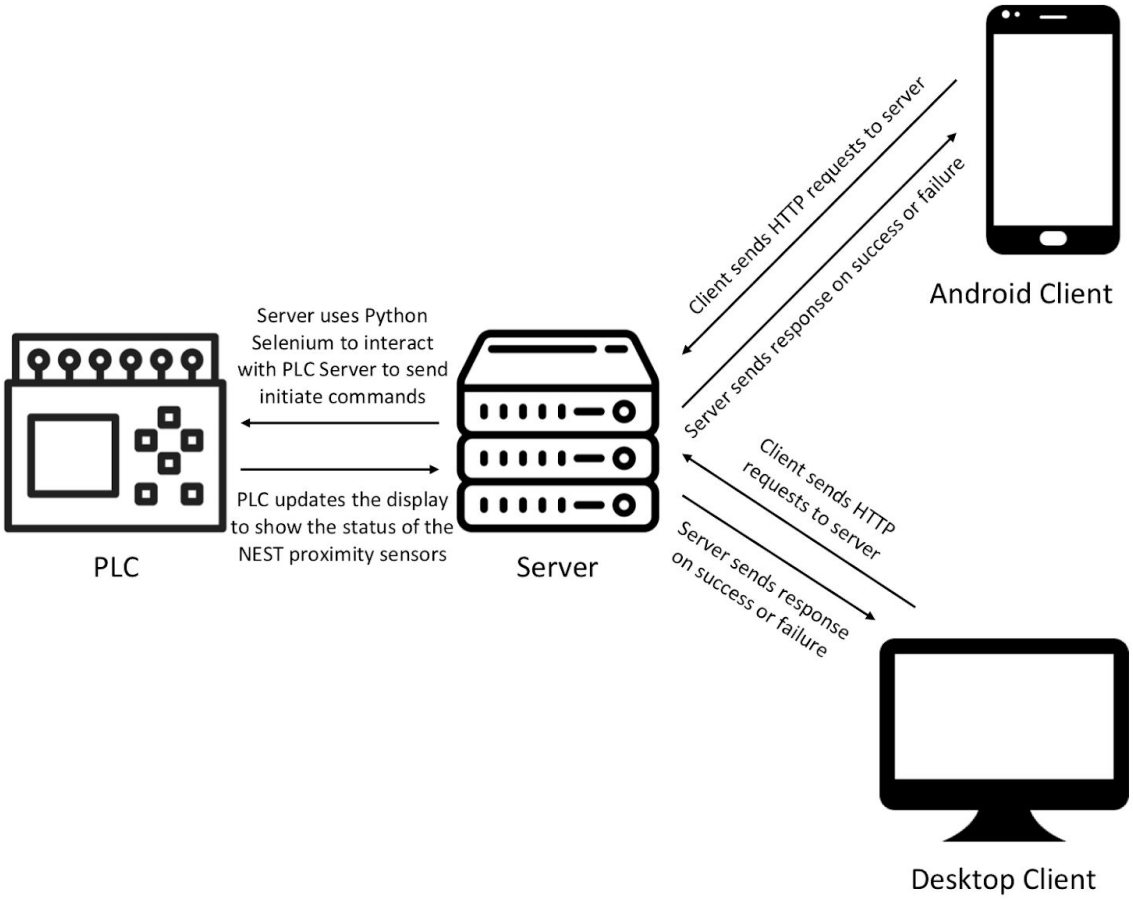
Charging System

Software

Software Overview

The NEST software package includes code for the server located on the master computer on the NEST, the desktop and Android clients, and the PLC which interfaces with the hardware. The NEST server is created with Python and Flask and uses Flask routing to carry out commands. The GUIs for both the

desktop client and the NEST server are created using PyQt5 and Flask. The Android application is created using Kotlin. These applications work together to carry out commands for the hardware. For these applications to work, they need to be on the same network. In order for the server to connect with the PLC, the PLC must be running its own web server on the same network. The clients send commands to the Flask server on the master computer, and the Flask server relays these commands to the PLC through the PLC's web server. The following chart shows the relationship between each piece of software.



Landing Procedure

PLC Code

The primary job of the PLC is to interface with the hardware components of the NEST in order to control motors, check proximity sensors, and execute commands. The PLC code is responsible for protecting the NEST and users from erroneous commands. The program in it controls the motors using information from

both the user and small proximity sensors placed on all moving parts. This assures that commands that must occur in sequence will only occur in the proper order.
The PLC runs a small web server. This web server is where the user can send commands to the PLC by setting certain command flags. The user can also monitor the status of certain proximity switches as well. This data is used to assure that both the PLC and machinery are working as intended.

[INSERT UPDATED FBD]

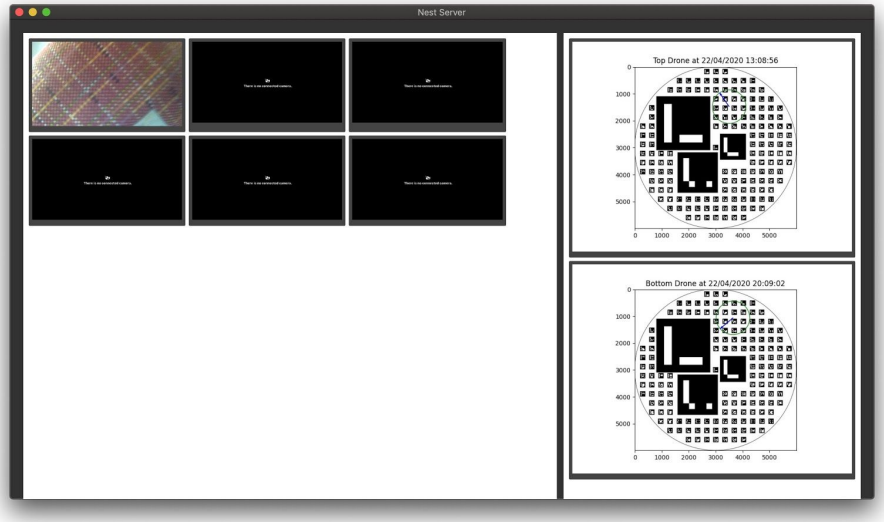
Servers

The NEST docking station features three local servers of varying types in order to give the user the most amount of information and control over both the drones and the docking station itself. All three servers are required to be on the same network

Python Flask Server on the Master PC

The master computer is responsible for processing requests and initiating a response from the PLC. It is built using Python 3.7+ and the Flask library. The server holds a machine object that represents the status of the NEST (ex. Doors: open, roof: closed, etc.). This object is used to determine whether or not a command should be sent to the PLC. To initiate a request, the clients navigate or perform an HTTP request to one of the flask routes. This server also handles video streaming through Flask. Before initiating the server, the user must verify that the strings in StringConstants.py are correct for IP and Port numbers. It is recommended that the user use their IPv4 address to keep the address relative and local. Additionally, the address of the PLC's web server must be added correctly for this application to work properly. The default strings that need to be modified are located in the StringConstants.py file in the root directory. See the documentation on Github to get started using this application. Github

The following image shows the user interface for the server.



PLC Web Server

The PLC has its own web server that it uses to display the status of each of the proximity sensors and also has input buttons to initiate commands on the PLC. The PLC has built in logic to determine whether or not a command is safe to execute. If the PLC determines that a command is not safe, then it will ignore the request. The Master PC interacts with this web server using Selenium. The PLC web server can be navigated to using the PLC IP address (It is currently set to http://192.168.99.3/). The user is required to login to use the PLC and to navigate to the custom PLC webpage. The PLC setup is more fully described above.

[INSERT UPDATED UI]

Drone Server

The drone server is a Telemetry style of server and is responsible for communicating tasks to the drones and receiving their collected data. This server is also responsible for scheduling missions. Github

Android Application

Software Docs

Overview

The android application acts as a client for the server. It sends and receives commands using a TCP connection and displays those commands in a log on the UI. It uses a WebView to display and switch between the different video streams from the server. A password override allows commands to be forced.

Built With

- Kotlin - Programming language
- AndroidX - Android libraries
- Volley - HTTP library
- Dokka - Documentation engine for Kotlin

Optional Tools

- kdoc-generator - Generates KDoc

Getting Started

Instructions for setting up the app on your machine.

Prerequisites

- Android Studio
- Kotlin Plugin
- Git
- Access to IMPRESS Github

Installation

1. Clone the repo
   ```
   git clone https://github.com/impress-msu/NestApp.git
   ```
2. Open project in Android Studio
3. Build and run the app on device or virtual device

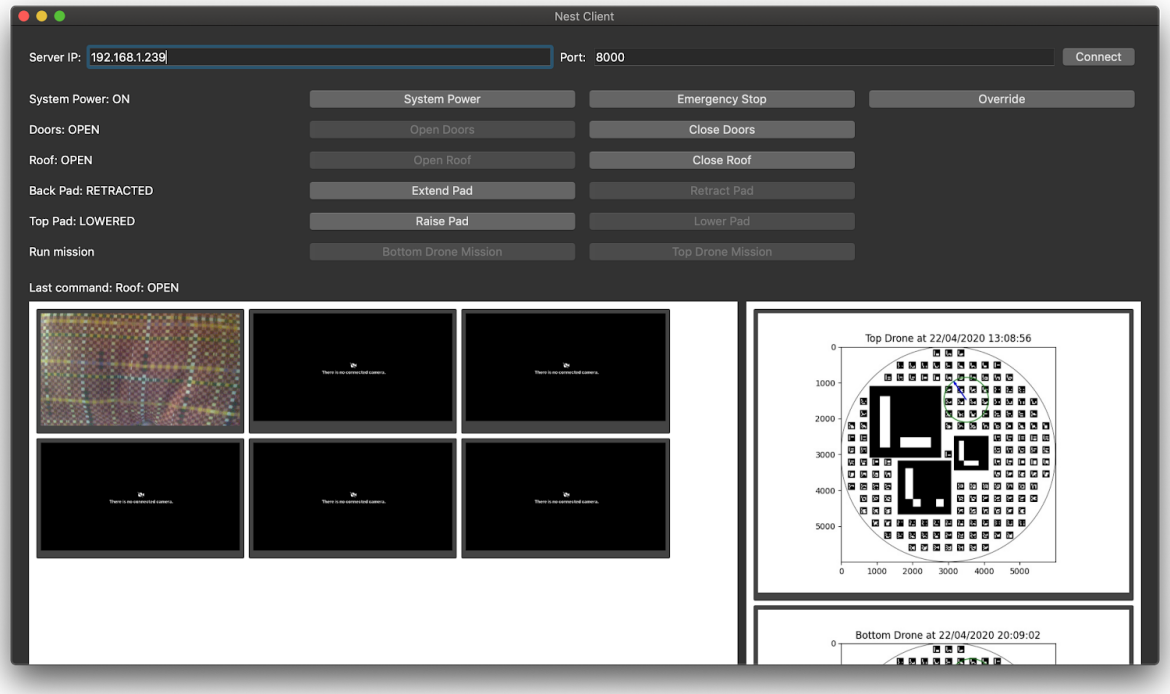
Contributing

1. Create a feature branch
   ```
   git checkout -b featureName
   ```
2. Commit your changes with message
   ```
   git commit -m “added this new feature”
   ```
3. Push to the branch
   ```
   git push origin featureName
   ```
4. Create a new pull request on the Github repo

Images

Windows Application

The Windows Application is a python executable for sending commands and receiving data from the Flask server on the master PC (See Above). It is important to clarify that this server is LOCAL ONLY.



The user is responsible for setting the server IP and port numbers and hitting connect. Upon a good connection, the user should be shown the video feeds and images. The user can then click buttons to send commands. The client polls the server every second to check for an updated status of the NEST. This ensures that each client has the same commands available and prevents erroneous commands.

Operating Procedures

On/Off Checklist

Turning the system on:

- ☐ Unlock back door
- ☐ Plug everything in
- ☐ Power on computer

- [ ] Check for any obstructions near moving parts (this includes the operator)
- [ ] Turn on breaker
- [ ] Check for red light on activated sensors
- [ ] Turn emergency stop

Turning the system off:

- [ ] Press emergency stop
- [ ] Check for red light on activated sensors
- [ ] Turn off breaker
- [ ] Power off computer
- [ ] Unplug everything
- [ ] Lock back door
- [ ] Double check that all previous steps have been completed
- [ ] Lock side door

Debugging

PLC State Table

State	Doors	Rail
Closed	**Close**	**Retracted**
Impossible	**Close**	**Cannot Extend**
Intermediate State	**Open**	**Retracted**
Takeoff/Land	**Open**	**Extended**

State	Roof	Lift
Closed	**Close**	**Lowered**
Impossible	**Close**	**Cannot Raise**
Intermediate State	**Open**	**Lowered**
Takeoff/Land	**Open**	**Raised**

Links

[1] Drive sprocket
https://www.grainger.com/product/TSUBAKI-Fixed-Bore-Roller-Chain-Sprocket-6L837
Idle sprocket
https://www.grainger.com/product/FENNER-DRIVES-Idler-Sprocket-3ZP37
Chain
https://www.grainger.com/product/DAYTON-Carbon-Steel-Roller-Chain-2YDW5
Bearing Hub
https://www.grainger.com/product/TRITAN-4-Bolt-Flange-Bearing-with-36UY09
Motor Shaft
https://www.grainger.com/product/KEYSHAFT-316-Stainless-Steel-Keyed-30F964
Shaft Coupler
https://www.grainger.com/product/RULAND-MANUFACTURING-1-Piece-Set-Screw-1-2-Bore-2AEK5
Motor
https://www.grainger.com/product/DAYTON-DC-Permanent-Magnet-Motor-6ML01

[2] Motor used for the door
https://www.groschopp.com/motor/87718/

[3] Progressive Automations
https://www.progressiveautomations.com/products/flt-06

[4] Motor used for rail
https://www.automationdirect.com/adc/shopping/catalog/motors/dc_gearmotors_(up_to_1-z-5hp)/right_angle_shaft/mtgr-p20-1k174

[104]
https://www.applied-motion.com/products/stepper-motors/ht34-486

[105]
https://www.applied-motion.com/products/stepper-drives/st10-si-nn

[5] Batteries
https://relionbattery.com/products/lithium/rb200

[6] Battery monitor
https://relionbattery.com/products/lithium/battery-indicator

[7] Solar Kit

https://invertersrus.com/product/gopower-solar-extreme-480/

[8]
https://www.grainger.com/product/AUTONICS-500-Hz-Inductive-Cylindrical-32W462

[9] PLC
https://new.siemens.com/global/en/products/automation/systems/industrial/plc/logo.html

[#] Github
https://github.com/lpjune/NestServerV2