AntiaTech

Test Documentation

09.12.2022

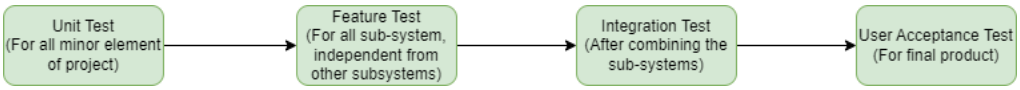
Table of Contents

AntiaTech Test Documentation

Introduction

This document provides possible test scenarios of the product. There are 4 different test scenario are defined; unit, feature, integration and user acceptance tests.

This demo document contains only the first three. Since the final product has not been completed, we don't have an acceptance test yet.



Definitions

Unit Test

By doing the unit test, we aim to test all minor elements of the system. These tests are not very comprehensive but very critical for our processes because any fault in a minor component in the system may have an important effect on other subsystems.

Feature Test

Feature test aims to test the combination of sub-system elements. These tests are executed for all different sub-systems, and the test inputs don't depend on the other sub-systems.

These tests are important in order to prevent any fault while integrating the sub-systems.

Integration Test

Integration test aims to test the behavior of a combination of all sub-systems. By doing these tests, we would like to ensure that our sub-systems are properly integrated and can work together.

User Acceptance Test

User Acceptance test aims to test if the final product meets the specifications of the project definition. This test will probably be executed at the end of the development progress.

Unit Tests

Check if DC Motors Work Continuously with Rated Current

Summary

Location & Date

Description & Aim

DC motors will probably work continuously during the game. Thus, we need to check if they can properly work continuously without interruption or hazard. In this test, we aims to assure that dc motors can work at least 10 minutes without interruption with rated current.

Participants

Preconditions & Environment Requirements

1. DC Motors which are connected with wheels
2. DC Supply
3. Digital Multimeter (Ground Truth: If multimeter & dc supply shows the same values)

Scenario

Step	Data	Expected Result	Actual Result	Error
Connect 1 of the dc motors to the supply	-	-		
Then, activate power supply with zero voltage & current	0V,0A	No motion observed		
Then, increase voltage limit	9V	No motion observed because supply is at CC mode		
Then, starts to increase current	0.10A	Motor starts to run with low speed		
Then, set current value to the rated current of dc motor	0.5A	Motor speed increases, Motor temperature increases		
Then, wait 10 minutes to observe if motor has been damaged or not	10 minutes	Motor should run continuously without interruption Motor temperature should not be high		

Check If Maximum Power Consumption of All Motor Units is In Proper Limits

Summary

Location & Date

Description & Aim

In the project, we are going to use batteries to power up all motor components. Since the batteries has some current limits, we need to check our maximum power ratings to run our system in proper limits.

In this test, all motors are run at their rated currents and maximum current rate will be observed. Then, it is optimized according to batteries capability.

Participants

Preconditions & Environment Requirements

1. DC Supply
2. Battery
3. Digital Multimeter *2 (Ground Truth: If multimeter & dc supply shows the same values)
4. DC Motors

Scenario

Step	Data	Expected Result	Actual Result	Error
Connect all dc motors at the same dc supply	-	-		
Then, activate power supply with zero voltage & current	0V,0A	No motion observed		
Then, increase voltage limit	9V	No motion observed because supply is at CC mode		
Then, starts to increase current	0.3A	Motors start to run with low speed		
Then, increate the current up to DC supply enters VC mode	-	Motors reach top speed. At 9V case, maximum current is observed. Check the 1 series 9V battery capability to run all motors at the same time.		

Check If Mechanical Components Fits Our Purpose After Printing Them

Summary

Location & Date

Description & Aim

Some mechanical components are printed in 3D printer. Since they are all plastic, their mechanical strength should be checked. Note that this test doesn't aim to broke printed material. This test only aims to be assure that if printed component is suitable for us or not.

Participants

Preconditions & Environment Requirements

1. Printed Barrel
2. Printed Wheels
3. Printed Floor
4. Printed Motor Bed
5. Printed Screw Bed
6. Other Printed Components
7. No ground truth for these tests

Scenario

Step	Data	Expected Result	Actual Result	Error
Check Barrel	Ball can pass inside it			
Check DC Motor Bed	Motor Fit In Bed			
Check Screw Bed is Capable to Carry all Balls etc.				
Check Wheels fit into Barrel and DC Motor Mill				
Check if Floor is capable to carry weight	Floor can carry all balls.	Up to 1.5KG		

Check if Servo Motor of Ball Thrower Can Work as Expected

Summary

Location & Date

Description & Aim

In the project, a servo motor is placed next to the ball thrower's barrel, and it supplies horizontal angle changes for the ball thrower. Aim of this test is to check if the servo motor is working properly in desired limits. Ground truth is protractor with right angle measurements.

Participants

Preconditions & Environment Requirements

- Arduino
- Servo Motor
- Protractor

Scenario

Step	Data	Expected Result	Actual Result	Error
Connect Servo Motor Directly to the Arduino	-	-	-	
Write a Code with 10° Angle Changes every time step in Arduino	-	-	-	
Compile and Upload the Code to Arduino	-	RX TX leds of the Arduino blinks	-	
Time Step 1	10°	Protractor angle shows 10°		
Repeat Previous Step 1 for 10 times	20°, 30°, 40°, 50°, 60°, 70°, 80°, 90°, 100°	Protractor angle shows 20°, 30°, 40°, 50°, 60°, 70°, 80°, 90°, 100°		
Write and Upload a Code with 5° Angle Changes every time step in Arduino				
Time Step 1	5°	Protractor angle shows 5°		
Repeat Previous Step 1 for 10 times	10°, 15°, 20°, 25°, 30°, 35°, 40°, 45°, 50°	Protractor angle shows 10°, 15°, 20°, 25°, 30°, 35°, 40°, 45°, 50°		
Write a Code with 100° Angle Changes in Total in Arduino				
Uploaded Code runs and changes the servo motor angle	100°	Protractor angle shows 100°		

Check if Step Motor of Ball Thrower Can Work as Expected

Summary

Location & Date

Description & Aim

In the project, a step motor is placed next to the ball thrower's tank, and it supplies vertical angle changes fort he ball thrower. Aim of this test is to check if the step motor is working properly in desired limits. Ground truth is protactor with right angle measurements.

Participants

Preconditions & Environment Requirements

- Power Supply and Motor Driver
- Step Motor
- Protractor

Scenario

Step	Data	Expected Result	Actual Result	Error
Connect Step Motor to Power Supply and Motor Driver	-	-	-	
Write a Code with 10° Angle Changes every time step in Arduino	-	-	-	
Give High to First Input of Motor Driver to Change Step Motor Angle		Protactor angle should show 0°		
Repeat Previous Step 3 times	45°, 90°, 135°	Protactor angle should show 45°, 90°, 135°		
Place the motor next to the system	-	Motor torque is high enough to turn the tank		

Check if Microphones Work Properly

Summary

Location & Date

Description & Aim

Three microphones are planned to be used in this project, two for recording the sound of the balls hitting the table and one for acquiring the player's commands. Since the performance expected from the microphones is not high, this standard unit test is designed to be applied to all microphones.

Participants

Preconditions & Environment Requirements

- A quiet place
- A person (two if possible, a male & a female) who will speak the commands (Ground truth: the command(s) the microphone should record is known)
- USB microphone
- Distance measurement tool (e.g., a tape measure)

Scenario

Step	Data	Expected Result	Actual Result	Error
Connect microphone to start recording sound	-	-	-	-
Arrange the distance from microphone to person who will speak the commands	40 cm	-	-	-
Record the command(s) the person speaks	10 times	all records include the command(s) in a clearly understandable way		
Repeat the second step	70 cm	-	-	-
Repeat the third step	10 times	at least 8 records include the command(s) in a clearly understandable way		
Repeat the second step	110 cm	-	-	-
Repeat the third step	10 times	at least 7 records include the command(s) in an understandable way		

Check if System Ball Tracking Can Work Continuously

Summary

Location & Date

Description & Aim

We have designed our system to be simple and user-friendly. So, we expect the system to be up and running at the moment it is powered. Camera will continuously monitor the table as the practices goes on. We need to see if the image processing module of our product works for long periods of time without any overheating or program crashing.

Participants

Preconditions & Environment Requirements

- Raspberry Pi
- An image sensor (a webcam, piCamera etc.)
- A table tennis set-up

Scenario

Step	Data	Expected Result	Actual Result	Error
Connect image sensor to the Raspberry Pi	-	-	-	
Connect power supply to Raspberry Pi	-	Image processing program starts running automatically	-	
Let the system run for a while	2 hours(runtime)	$-40° < T_{measured} ° < 85°$		
Let the system run for a while	2 hours(runtime)	No crashing		

Check if the Vibration Sensor Can Work Properly

Summary

Location & Date

Description & Aim

The two vibration sensors at the left and right edges of the table-tennis table are to sense the vibration when a ball hits the table. If they do not hit the right part of the table, sensors don't detect the vibration. Ground truth is ten certain ball hits on the table.

Participants

Preconditions & Environment Requirements

- Vibration sensor
- An Arduino with power supply
- A table tennis set-up and balls

Scenario

Step	Data	Expected Result	Actual Result	Error
Connect vibration sensor to Arduino	-	-	-	
Upload the code to Arduino for detecting vibration	-	-	-	
Place the vibration sensor and Arduino next to tennis table	-	-	-	
Power the Arduino and throw a ball to the table at 25 cm distance to set-up	0-10000 Hz	Serial monitor shows non-zero data between 0-10000 Hz		
Repeat the previous step 9 times	0-10000 Hz	Serial monitor shows non-zero data between 0-10000 Hz		
Throw a ball to the table at 50 cm distance to set-up cm distance to set-up	0-10000 Hz	Serial monitor shows non-zero data between 0-10000 Hz		
Repeat the previous step 9 times	0-10000 Hz	Serial monitor shows non-zero data between 0-10000 Hz		
Throw a ball to the table at 80 cm distance to set-up cm distance to set-up	0-10000 Hz	Serial monitor shows non-zero data between 0-10000 Hz		
Repeat the previous step 9 times	0-10000 Hz	Serial monitor shows non-zero data between 0-10000 Hz		

Feature Tests

Check if ball-thrower can throw ball with manually entered input speeds

Summary

Location & Date

Description & Aim

After combining minor components, we need to check if ball-thrower can throw balls with desired speed and desired direction.

Participants

Preconditions & Environment Requirements

- Power Supply
- Ball
- Ball-Thrower Body with controller elements

Scenario

Step	Data	Expected Result	Actual Result	Error
Check if all motor components are connected to input terminal	-	All motors should be connected to the related port of motor driver		
Then, power up controller but not dc motors	5V	Arduino and Controller Circuit leds should be activated		
Then, set PWM of motor controller inputA	20%	Only motors 1 should starts to run		
Then, push a ball to the barrel and check if it is thrown by whell		Ball, should be thrown with low speed		
Then, change the PWM of motor controller inputA	70%	• Motor should accelerate • Possible “zzz” noise can be occurred • Motor should run continously		
Then, change the PWM of motor controller inputA	0%	Motor 1 should decelerate and stop.		
Then, repeat the steps above for the motor 2				

Check if balls are sent to barrel with manually entered input speeds

Summary

Location & Date

Description & Aim

Balls should be pushed to the barrel in order to throw them. Thus, we build a push mechanism that contains a turning platform and 1 dc motor.

Participants

Preconditions & Environment Requirements

- Balls
- Box Mechanism which contains all dc motors, turning platform(floor) and controller

Scenario

Step	Data	Expected Result	Actual Result	Error
Check the motor is connected to the input terminal.		Motor terminals should be connected the terminals properly.		
Then, give power to the controller but not motors.		Arduino and Controller Circuit LEDs should be activated		
Then, set PWM of motor controller	30%	Balls should be pushed to the barrel slowly (1ball/1 sec)		
Then, change the PWM of motor controller	80%	Balls should be pushed to the barrel fastly (2 balls /3sec)		

Check if Barrel and Motors Are Connected Properly

Summary

Location & Date

Description & Aim

DC motors of ball thrower should be integrated to the barrel in order to proper throw mechanism. Thus, we need to do test this part carefully because in case of any error or mistake, the balls can't be accelerated properly.

Participants

Preconditions & Environment Requirements

1. Printed Barrel
2. Printed Motor Bed
3. DC Motor connected with a Wheel
4. DC Supply
5. Ball

Scenario

Step	Data	Expected Result	Actual Result	Error
Check if Motor Bed and Barrel is well connected		When someone tries to separate bed from barrel, it should not be separated		
Then, check if DC motor can fit into motor bed		DC motor should fit into the bed and there should not be any space.		
Then, put DC motor into the bed and connect its terminal to dc supply directly		Motor is ready to be run		
Then, activate dc supply and set the voltage	V = 9V	Motor should start to accelerate, and motor should be stationary inside the bed		

Check if Ball Thrower Changes Horizontal Angles Correctly

Summary

Location & Date

Description & Aim

After combining the servo motor with the rest of the ball thrower mechanism, horizontal angle changes of the overall system with the proper time limits will be checked. The ground truth is protractor measuring the angle differences, and chronometer measuring the time change.

Participants

Preconditions & Environment Requirements

- Tennis Table
- Balls
- DC Power Supply
- Ball Thrower
- Protractor
- Chronometer

Scenario

Step	Data	Expected Result	Actual Result	Error
Activate ball thrower	-	-	-	
Activate the mode with 10° angle changes for every time step, and minimize the mass of the ball thrower	-	-		
Time Step 1 - Ball thrower with minimum mass rotates horizontal 10°	10°	Protractor angle shows 10°		
Repeat the previous step 6 times	20°, 30°, 40°, 50°, 60°	Protractor angle shows 20°, 30°, 40°, 50°, 60°		
Activate the mode with 10° angle changes for every time step, and maximize the mass of the ball thrower				
Time Step 1 - Ball thrower with maximum mass rotates horizontal 10°	10°	Protractor angle shows 10°		
Repeat the previous step 6 times	20°, 30°, 40°, 50°, 60°	Protractor angle shows 20°, 30°, 40°, 50°, 60°		
Activate the mode with 10° angle changes for every 3 secs				
Ball thrower rotates horizontal 10° every 3 seconds	10°, 20°, 30°, 40°, 50°, 60°	Protractor angle changes every 3 seconds and show 10°, 20°, 30°, 40°, 50°, 60° in total 18 seconds		

Check if Ball Thrower Changes Vertical Angles Correctly

Summary

Location & Date

Description & Aim

After combining the step motor with the rest of the ball thrower mechanism, vertical angle changes of the overall system with the proper time limits will be checked. The ground truth is protractor measuring the angle differences, and chronometer measuring the time change.

Participants

Preconditions & Environment Requirements

1. Tennis Table
2. Balls
3. DC Power Supply
4. Ball Thrower
5. Protractor
6. Chronometer

Scenario

Step	Data	Expected Result	Actual Result	Error
Activate ball thrower	-	-	-	
Activate the mode with 10° angle changes for every time step, and minimize the mass of the ball thrower	-	-		
Time Step 1 - Ball thrower with minimum mass rotates vertical 10°	10°	Protractor angle shows 10°		
Repeat the previous step 6 times	20°, 30°, 40°, 50°, 60°	Protractor angle shows 20°, 30°, 40°, 50°, 60°		
Activate the mode with 10° angle changes for every time step, and maximize the mass of the ball thrower				
Time Step 1 - Ball thrower with maximum mass rotates vertical 10°	10°	Protractor angle shows 10°		
Repeat the previous step 6 times	20°, 30°, 40°, 50°, 60°	Protractor angle shows 20°, 30°, 40°, 50°, 60°		
Activate the mode with 10° angle changes for every 3 secs				
Ball thrower rotates vertical 10° every 3 seconds	10°, 20°, 30°, 40°, 50°, 60°	Protractor angle changes every 3 seconds and show 10°, 20°, 30°, 40°, 50°, 60° in total 18 seconds		

Check if System Can Detect the Ball Accurately

Summary

Location & Date

Description & Aim

The detection of the ball hitting the tennis table is done by vibration sensors and microphone. Ground truth is ten certain ball hits on the table for each unit, for each side and outside of the table (ball thrower side - user side - floor).

Participants

Preconditions & Environment Requirements

- Ball thrower
- A table tennis set-up

Scenario

Step	Data	Expected Result	Actual Result	Error
Activate ball thrower	-	-	-	
Throw a ball to the ball thrower side of table	0-10000 Hz	Ball thrower detects the throw		
Repeat the previous step 9 times	0-10000 Hz	• Ball thrower detects the throw 9 times • Vibration sensors detect the throw 9 times • Microphone set-up detects the throw 9 times		
Throw a ball to the user side of table	0-10000 Hz	Ball thrower does not detect the throw		
Repeat the previous step 9 times	0-10000 Hz	• Ball thrower does not detect any throw • Vibration sensors detect the throw 9 times • Microphone set-up detects the throw 9 times		
Throw a ball to the outside of table	0-10000 Hz	Ball thrower does not detect the throw		
Repeat the previous step 9 times	0-10000 Hz	• Ball thrower does not detect any throw • Vibration sensors detect the throw 9 times • Microphone set-up detects the throw 9 times		

Check if System Can Track Ball Accurately

Summary

Location & Date

Description & Aim

Camera will continuously monitor the table as the practices goes on. We need to see if the image processing module of our product works.

Participants

Preconditions & Environment Requirements

- Raspberry Pi
- An image sensor (a webcam, piCamera etc.)
- A table tennis set-up

Scenario

Step	Data	Expected Result	Actual Result	Error
Connect power supply to Raspberry Pi	-	-	-	
Connect image sensor to the Raspberry Pi	-	-	-	
Execute the ball tracking software	-	Program runs successfully		
Place the balls in predestined locations where we know the x and y locations (ground truth).	x and y coordinates of the pre-decided locations (in cm)			
See if the algorithm responses	-	Program prints any (x, y) location to the console		
Compare the generated results and ground truth.	x and y coordinates of the pre-decided locations (in cm)	exact (x±1, y±1) locations of the balls		

Check if Raspberry Pi Understands the Verbal Commands

Summary

Location & Date

Description & Aim

To check how successfully Raspberry Pi understands the verbal commands recorded via a USB microphone.

Participants

Preconditions & Environment Requirements

- A quiet place
- USB microphone
- Raspberry Pi 4 (4GB) (running speech recognition software)
- A person (two if possible, a male & a female) who will speak the commands (Ground truth: the command(s) Pi should understand is known)
- Distance measurement tool (e.g., a tape measure)

Scenario

Step	Data	Expected Result	Actual Result	Error
Connect USB microphone	-	-	-	-
Get Pi running the code	-	-	-	-
Arrange the distance from microphone to person who will speak the command	40 cm	-	-	-
Check whether Pi understands the command after the person speaks it	10 times	At least 9 times Raspberry Pi understood the command		
Repeat the previous two steps	70 cm 10 times	At least 8 times Raspberry Pi understood the command		
Repeat the previous two steps	110 cm 10 times	At least 7 times Raspberry Pi understood the command		
Repeat the previous four steps for different commands	-	-	-	-

Integration Test

Check if ball-thrower mechanism with all components can work properly.

Summary

Location & Date

Description & Aim

After constucting all mechanical components of the project, we need to assure that ball-thrower can take ball from box and throw them with desired speed and to desired direction.

Participants

Preconditions & Environment Requirements

1. All mechanical part is done (Barrel, box, other dc motor components etc.)
2. Balls

Scenario

Step	Data	Expected Result	Actual Result	Error
Check if all necessary components are connected to the power supply.		All required components are ready to power up.		
Then, activate one of the barrel motors	Speed of motor will be determined by a pot or Raspberry command	Motor 1 starts and accelerate		
Then, activate the ball pusher motor	Speed of motor will be determined by a pot or Raspberry command	Balls are started to be pushed to barrel and they should be thrown		
Then, activate the servo motors that change the motor direction horizontally	Direction of motor will be determined by a pot or Raspberry command	Barrel starts to turn through desired direction while they are currently throwing balls		
Then, stop the servo motor of horizontal motion and change the angle of barrel for vertical direction	Direction of motor will be determined by a pot or Raspberry command	Barrel angle changes according to user decision		

Check if immediate photo capture is possible by signaling

Summary

Location & Date

Description & Aim

We would like to capture exactly where the ball has landed on the table. To achieve this, we plan to sense when the ball has hit the table, be it vibration sensor or audio sensor, and capture the place of the ball with our image sensor. This test is to see if we can capture the exact place of the ball with little error when a correct signal is emitted.

Participants

Preconditions & Environment Requirements

- Raspberry Pi
- An image sensor (a webcam, piCamera etc.)
- A table tennis set-up
- A signal emitter with the correct timing
- Ball launcher shooting at the exact place at an interval (ground truth)

Scenario

Step	Data	Expected Result	Actual Result	Error
Connect power supply to Raspberry Pi	-	-	-	
Connect image sensor to the Raspberry Pi	-	-	-	
Execute the ball tracking software	-	Program runs successfully		
Start the ball launcher	-	-	-	
Compare the generated results and ground truth.	(x, y) coordinates (in cm) of the place where the balls repeatedly lands	(x±2cm, y±2cm)		