miniATV goes WinterSIM
Test1 – Driving straight ahead

Objective:
Give both the physical miniATV and it's digital twin in WinterSIM the maximum speed steering command to drive for 10s straight ahead, note down the distance each miniATV drove and how much the physical miniATV diverged to the left or right.
All ROS topics are recorded and the data of the GNSS topics is used to compare the physical and digital miniATV.
Additionally, the physical miniATV's driven distance and deviation from the straight line will be measured with a measuring tape.
The battery level before each test run is also noted down to bring light to the relationship between charge level and power of the physical miniATV.
At the same time than measuring similarity in movement between the physical and digital miniATV, recording all the ROS topics gives insight into delays in the system caused by the software architecture. For the physical miniATV, the hardware could also add delays.
The recorded rosbags of all tests can later be used as a benchmark, for example to compare both miniATVs' behavior in different weathers.

Necessary tools, devices and settings:

- 50 m of measuring tape for measuring the distance the physical miniATV has driven forward (0.5 cm is the smallest increment on the tape)
- Tent pegs to fix the 50 m measuring tape to the ground in order to keep the measuring tape without slack
- A smaller measuring tape, 3 m should be enough, to measure the miniATV's deviation from the straight line.
- Semi-permanent marker for the start position and the line from which the deviation is measured
- A tripod and camera
- A timer with the ability to set marks for synchronizing the screen recording with the camera recording and the start of the test drive Python script
- If possible, the test is conducted in dry weather, both in the physical testing ground and the digital twin of the parking lot. Set the weather in WinterSIM to similar weather
- At least 10 m extension cable for supplying the "Mustaboxi" WiFi and the Laptop with electricity
- Chalk and duct tape for markings on the ground
- iperf3 installed on the Laptop and the miniATV for testing the connection between those two
- Laptop: WinterSIM, ROS1, carla_ros_bridge and multimaster_fkie
- miniATV: ROS1, multimaster_fkie and the files in https://github.com/Lapland-Robotics/miniATV/tree/master's_thesis_cs/atv_setup, especially the emergency stop

Illustration 1: The roll of 50 m measuring tape and the 3 m measuring tape used for the test

Testing ground:
A part of the Rantavitikka parking lot of the Lapland University of Applied Sciences (Link to Google Maps: https://goo.gl/maps/R1jgJubXY1oM7QNd9), that is only rarely used for the Lapland UAS's own cars and trailers and not by the students and employers. Even pedestrians rarely frequent this space. Still, this testing area is shared with other parties and their safety must be considered during the tests.

Ideally the testing area would be totally even. But since the parking lot is designed to conduct the water of the molten snow fallen during the long arctic winter, all surfaces are curved to prevent flooding on the parking lot and to support the drainage. Therefore, a test track was chosen that has minimal bumps and slope, but has a grid of tiles that helps measurement and adds visual marks to the video recording. The texture is not replicated in that area in the digital twin of the parking lot, but the rest of the parking lot area is a wild mix of different types of asphalt from different phases of the construction of the parking lot, renovations and patching of damage in the asphalt. In contrast to the tiles, changes in texture follow no pattern there. Since the asphalted parts of the parking lot are usually in active use, the less frequented area with the tiles was chosen.

Illustration 2: Google maps image of the Rantavitikka parking lot with highlighted testing area

Illustration 3: View to the east of an quite even area of the test ground

Important:

- 14 test runs are conducted, 7 for each available network (Phone Hotspot and “Mustaboxi” WiFi – connection speed is tested before for each network)
- Since the screen recording could slow down the laptop and influence the delays in ROS and WinterSIM, only 2 of 7 tests for each network option are recorded.
- The hardware of the miniATV may influence the physical part of the test in two ways: The level of the battery charge could influence the speed and acceleration of the physical miniATV. The physical miniATV never showed an obvious loss of power when the battery level sank, but the first version of the “Snower” with a similar type of motors, but much heavier and with tracks instead of wheels, showed a clear decline in power caused by the level of the battery. The other influencing part of the hardware is the steering, it is likely that a deviation to the left or right will become more pronounced the longer the physical miniATV is driving due to rapidly aging 3D-printed plastic gearwheels and transmissions which could only be repaired and maintained but not replaced by the time of the tests due to time constraints.
- No interference of the screen recording or the change to another network is expected to influence the physical miniATV’s deviation from the straight line or it’s speed and acceleration, which might occur over the time span of the test (see explanations above). Therefore both options for the networks are tested one after the other and not alternately.
- For this test, only a speed command is given, the steering angle command is set to 0
- If possible, test only in dry weather with clear sky to prevent slipping and improve the GNSS accuracy

Test setup:

- Choose an as even as possible area in the test ground for the test that is similar in the WinterSIM
- Mark the start position on the ground and transfer it's GNSS position to the WinterSIM
- Place the 50m measuring tape and secure it with the tent pegs
- Place the camera tripod and adjust the WinterSIM spectator accordingly
- Get a table for the laptop (for ergonomic working conditions), an extension cable for the devices and WiFi "Mustaboxi" and set that up near the middle of the test track where it doesn't disturb the driving but provides a good network connection to each end of the (ca. 20m) test track
- Connect the phone to a power bank

Test procedure:

In case of software of hardware failures:

Note down the failure and if sensible, try again to restart the current test run (for example if the test script or the WinterSIM didn't start). If the test run fails during the driving or close to the end, only restart that test run for very solid reasons (which should also be noted down).

Before the very 1st test run:

- This test procedure is to be published in the GitHub and not to be changed between test runs
- The miniATV battery has to be fully charged, calibrate the steering and get the miniATV to the test ground with the similar battery of the "Snower".
- Calibrate the steering such as to cause as less deviation as possible
- Check the connection speed between the Laptop and miniATV with iperf3 in both directions and safe the results

Before every test run:

- Reset both the physical and digital miniATV to it's starting position
- Check if the ground plate and the GNSS sensor are still aligned
- Note down the time and date of the next test run
- Directly before starting recordings and test script, note down the physical miniATV's battery level (this is fluctuating a bit in the second decimal, try using roughly the middle point of the fluctuation)

For recorded test runs (1st, 2nd, 8th, 9th):

1. Start the timer on the phone
2. Place a marker in the timer when starting the camera recordings
3. Start the WinterSIM and carla_ros_bridge
4. Place another marker when the screen recording is started
5. Place the last marker when the test script test2_straight_ahead.py is started
6. Make a screenshot of those 3 markers and note down the times in the datasheet

For not-recorded test run:

- Start the WinterSIM and carla_ros bridge
- Start the test script test2_straight_ahead.py

After each test run:

- Measure the distance driven and the deviation from the straight line
- Close the WinterSIM and all related windows to "clear the stage"

Between the 7th and 8th test run:

- Change the network from the phone hotspot to the “Mustaboxi” WiFi (might need a restart of the miniATV)
- Change the GNSS sensor’s network as well (has to be done manually)
- Test the connection between the Laptop and the miniATV with

After all 14 test runs:

- All collected data and notes is to be uploaded into the GitHub for safekeeping
- Passwords might be blurred out
- Notes and data collected on paper have to be transferred to suitable electronic documents for safekeeping and evaluation
- Assemble a short report with photos about how the test went

Required commands:

For the iperf3:

First start the server on the miniATV: ***iperf3 -s***
On the laptop go to the folder for this test’s log files:
Then on the laptop: ***iperf3 -c 192.168.1.106 --logfile test2_run1***
And to switch direction: ***iperf3 -c 192.168.1.106 -R --logfile test2_run3_R***

For starting the everything else:

(From the home directory of the Laptop)
To start the WinterSIM, spawn the digital miniATV in the digital testing ground, starting everything on the miniATV side and starting the multimaster_fkie:
bash wintersim_miniatv_start.sh
To start the actual test script that gives the steering commands:
rosrun carla_manual_control test2_straigt_ahead.py

Net-work:	# test run	Battery level [V]	Date & Time	Timer at start of video record-ing	Timer at start of screen record-ing	Timer at start of the Python test script	Distance driven ahead (two decimals) [m]	Devi ation Left/ Right ?	Deviation (one decimal) [cm]	Notes ? (See below)
Phone Hot-spot	Laptop – miniATV connection speed test conducted and saved? □									
	1									
	2									
	3			X	X	X				
	4			X	X	X				
	5			X	X	X				
	6			X	X	X				
	7			X	X	X				
Musta -boxi	Laptop – miniATV connection speed test conducted and saved? □									
	8									
	9									
	10			X	X	X				
	11			X	X	X				
	12			X	X	X				
	13			X	X	X				
	14			X	X	X				

Further notes (mention the # of the test run!):