UNIVERSITÀ DEGLI STUDI DI BRESCIA

LABORATORIO DI MISURE MECCANICHE E TERMICHE
DIPARTIMENTO DI INGEGNERIA MECCANICA E INDUSTRIALE

Manual
Instrumented Crutches

Summary

mmtlab.unibs.it

Via Branze 38
25123 Brescia
Italy

Hardware overview

The hardware configuration for the use of the instrumented crutches includes the following devices:

- **Coordinator**: is a computer that acts as the crutch coordinator. It provides a user interface and manages the data storage.
- **Left-crutch/ Right-crutch**: instrumented crutch equipped with a processing unit and sensorized for the force and inertial measurements. It also provides the step phases detection.
- **Synchronizer** (optional): a processing unit that allows trigger signals to be sent to external devices (e.g., visual systems) via dry contact and BNC terminals.

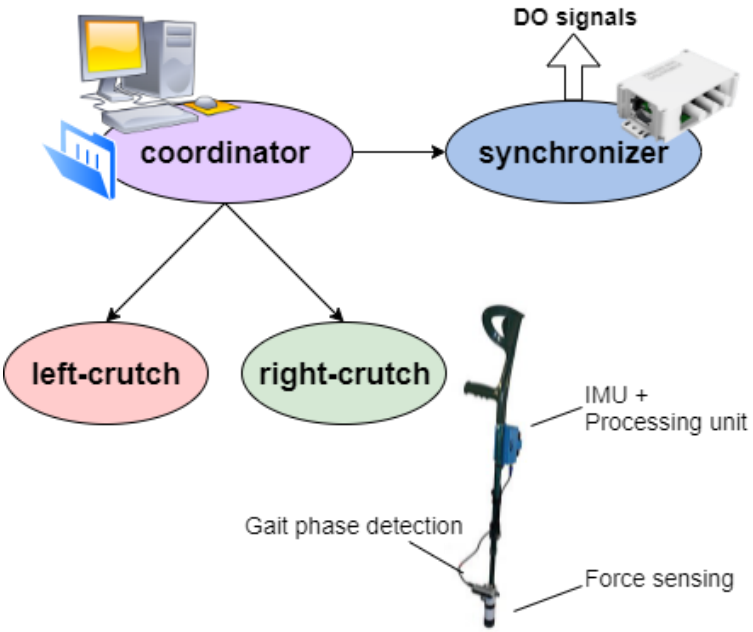


Figure 1 - Diagram of the devices for the use of the instrumented crutches.

Instrumented crutches

Each instrumented crutch is equipped with a raspberry to acquire data from sensors and send them to the coordinator via ROS network and Wi-Fi connection. A power bank of 10Ah supplies the raspberry allowing wireless using for an estimated time of 8 hours.

Force

A strain gauge is mounted on the bottom part of the crutch close to the tip. The strain gauge is composed by three full bridges connecting in parallel and mounted with 120 degrees from each other. The generated voltage is measured by a load cell amplifier (Sparkfun Load Cell Amp HX711 in Figure 2). The strain gauge allows to measure the force applied along the axis of the crutch and the data are sent to the coordinator who will save them to csv files in the format as described in *The saved csv files* use of the ";" as a separator.

Data format - Force.

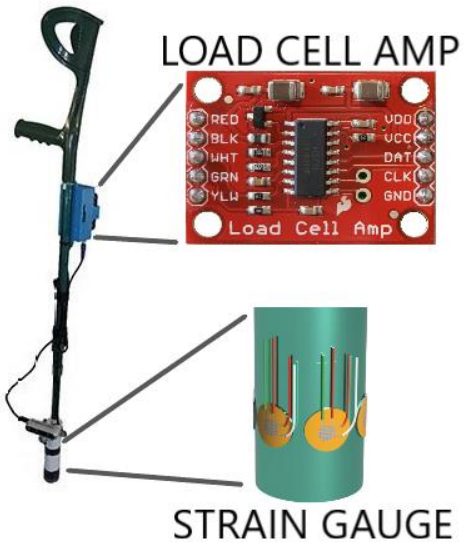


Figure 2 – HX7111 load cell amplifier with three Wheatstone bridges connected in parallel for force measurement.

IMU

Inertial measurements are obtained by an 9DoF IMU (Steval-MK1159V1 in Figure 3) equipped with accelerometer, gyroscope and magnetometer. The IMU is fixed on the raspberry near the handle of the crutch. The data is sent to the coordinator who saves them according to the data format indicated in the section *Data format - IMU*.

Figure 3 - 9DoF IMU Steval MK1159V1.

Step phase

The step phases are obtained by depth images captured by the Realsense D435i mounted on the bottom part of the crutch as shown in Figure 4. The ground plane is identified from the depth images and the distances between the plane and the points of the foot are computed. The acquisition time and the percentiles 5, 25, 50, 75, 95 of the distance distribution are sent to the coordinator who saves them in the format indicated in

the paragraph *Data format - Realsense*. The percentiles are then used as features to obtain the step phase and the reliability using the classification algorithm Random Forest Modified[1].

Figure 4 - Realsense D435i for depth image acquisition.

Coordinator

The coordinator provides a user interface to control the instrumented crutches, collect and save data. Depending on the working mode, the coordinator allows to handle the acquisition start command by itself or an external device. In ROS-mode the start and stop commands are sent by an external device through the ROS network. While working with the Stand-alone modality the commands are sent directly with the available user interface. The user interface usage is explained in detail in the section *Use of the user interface*.

Synchronizer (optional)

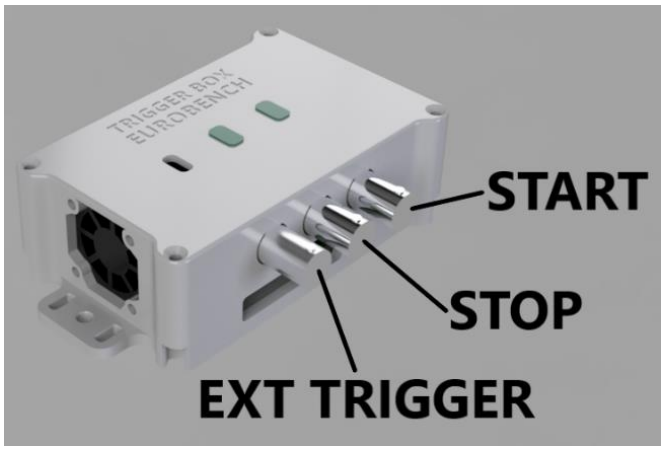
The synchronizer shown in Figure 5 allows to send trigger commands at the start and at the end of the acquisitions. It also provides a digital input to detected trigger signals and publish the timestamp on a topic of the ROS network. It is not necessary for the operation of the instrumented crutches, but it is useful for devices that cannot interface on the ROS network and allow to manage the acquisition through trigger signals (e.g., some vision systems). The device is equipped with a raspberry pi B+ connected on the same ROS network of crutches and waits for the start command. Digital outputs and inputs are available via BNC female connectors or clean contact terminals.

The available trigger signals and messages, shown in Figure 5, are:

- Start: a pulse from 0 to 1 of duration 150ms when "True" is received in the message of the topic "/start". A new message is published on the topic "/start_timestamp" as soon as the trigger Start output is raised.
- Stop: a pulse from 0 to 1 of duration 150ms when "False" is received in the message of the topic "/start".
- Ext trigger: the timestamp of the device is publish on the topic “ext_trigger_timestamp” when a rising edge is detected on the input signal.

[1] Pasinetti, S., Fornaser, A., Lancini, M., De Cecco, M., & Sansoni, G. (2020). Assisted Gait Phase Estimation through an Embedded Depth Camera Using Modified Random Forest Algorithm Classification. *IEEE Sensors Journal*, *20*(6), 3343–3355. https://doi.org/10.1109/JSEN.2019.2957667

a) Trigger box



b) Trigger signals

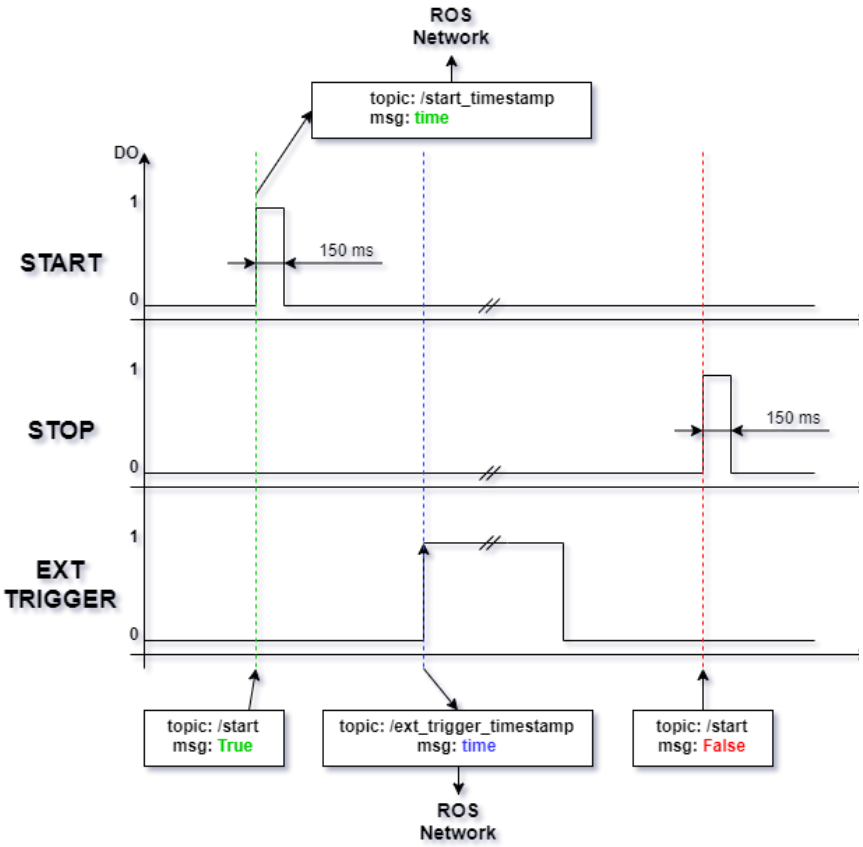


Figure 5 - a) Synchronizer box; b) Trigger signals on digital outputs and inputs and message published on the ROS network.

ROS network

ROS allows to exchange messages between nodes sending them on topics. Nodes are graphically represented by oval shapes and topics by rectangular shapes. As shown in Figure 6 the *publisher* node publishes the message on the topic and another node called *subscriber* reads the message from the topic. In addition to the standard representation of the ROS network elements, the launch of a node from another node is represented by an arrow with a dashed outline and starts from the node that launches (launcher node) and points to the launched node (launched node).

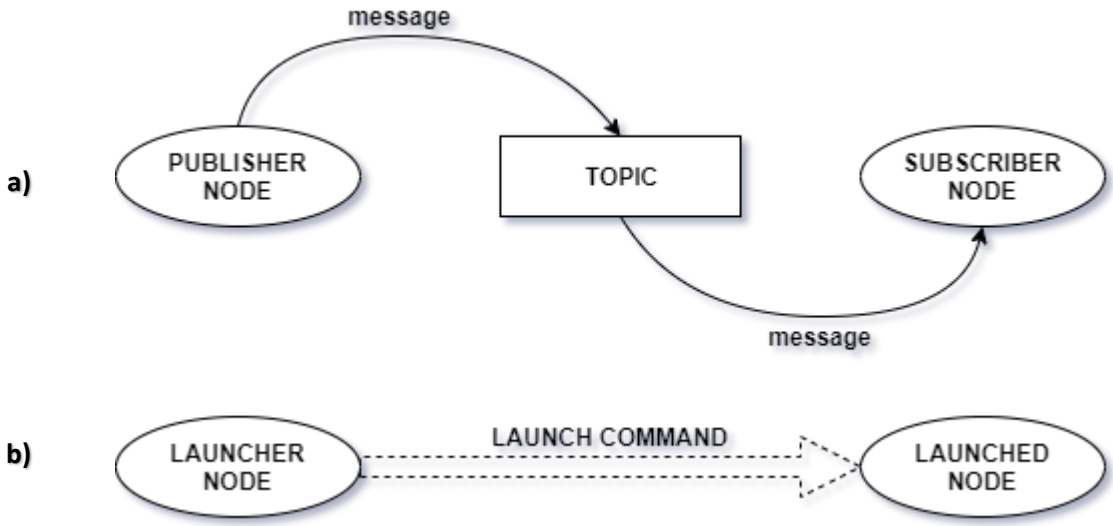


Figure 6 - a) Legend of the main elements of ROS; b) Custom legend to represent the launch command used in the instrumented crutches software.

The ROS environment requires a master that can be run on the coordinator or a different device belonging to the same network. In Stand-alone mode the coordinator performs the function of master, while in ROS-mode the master is executed on another device.

Each node and topic of the instrumented crutches is identified by the MAC address of the raspberry mounted on the crutch allowing to uniquely locate the nodes and the topics of the crutch. In the following diagrams for generality and simplicity they will be indicated with "node_name_MAC" or "topic_name_MAC". For example, the raspberry of the left crutch has MAC b8:27:eb:81:73:56 and it generates the nodes "node_name_b8_27_eb_81_73_56" and the topics "topic_name_b8_27_eb_81_73_56".

The */master* node in the green box in Figure 7 manages the *start* of acquisitions on the topic */start* and defines the test identifier on the topic */test_ID*. In Stand-alone mode the /master node can exist on the network, but the coordinator does not subscribe to the /start and */test_ID* topics because the acquisitions are managed by the user interface.

When the **BULLET** executable is started, the /crutch_coordinator node is created on the coordinator to manage the instrumented crutches. At the connection command with the instrumented crutch, the coordinator starts the node */crutch_controller_MAC* on the crutch, which publishes its status on the topic /status_crutch_controller_MAC. The */crutch_controller_MAC* node subscribes to the /nodes_enabling topic which provides the enabling commands for the onboard devices.

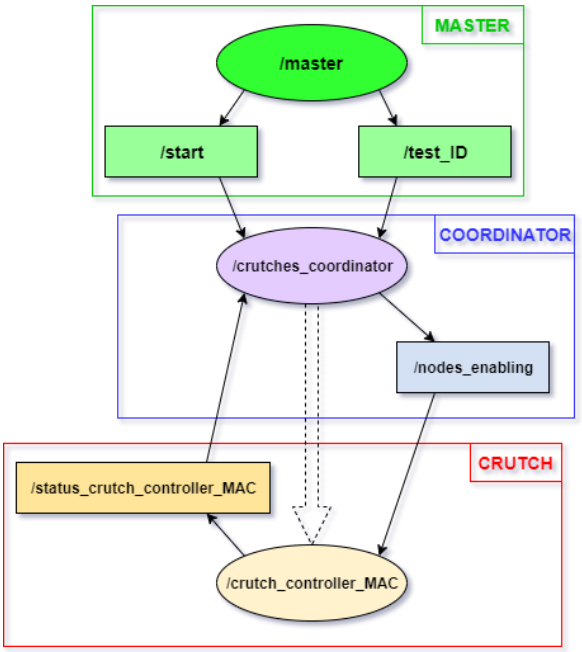


Figure 7 – The ROS network after the connection command is sent.

Below is the list of all the devices that can be enabled:

- the load cell node: /loadcell_MAC
- the IMU node: */IMU_MAC*
- the Realsense node: */camera_MAC.*

The */nodes_acquiring topics* are created to control the star of the acquisitions and */offset* to enable force offset. Each node on the instrumented crutch subscribes to the topic /nodes_enabling to close itself on the disable command.

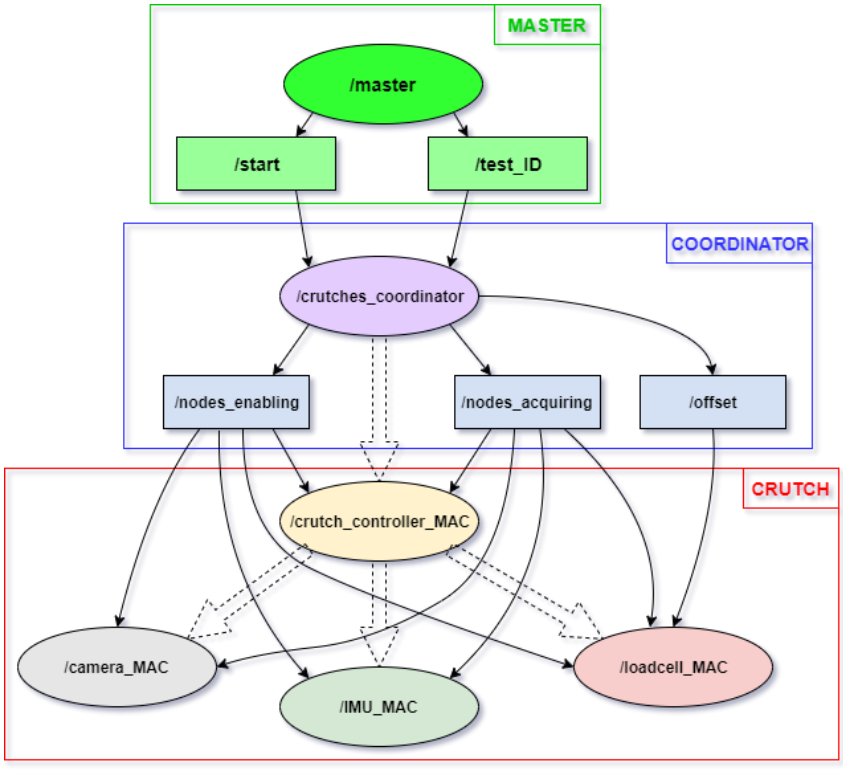


Figure 8 - ROS network at the launch of loadcell, IMU and camera nodes.

When the coordinator enables the acquisition by sending the message on the topic /nodes_acquiring the devices start to collect the data and to publish them on the topic */IMU_readings_MAC* and /loadcell_readings_MAC. The node */camera_MAC* does not immediately publish messages because it saves images locally on the raspberry and analyses them at the end of the acquisition. The devices also publish their states on the topics:

- */status_IMU_MAC*
- */status_loadcell_MAC*
- */status_camera_MAC*.

The node */camera_MAC* starts analysing all the images saved and publishes the results on the topic /step_features_MAC. The coordinator is updated of the progress on the topic /status_analyze_MAC.

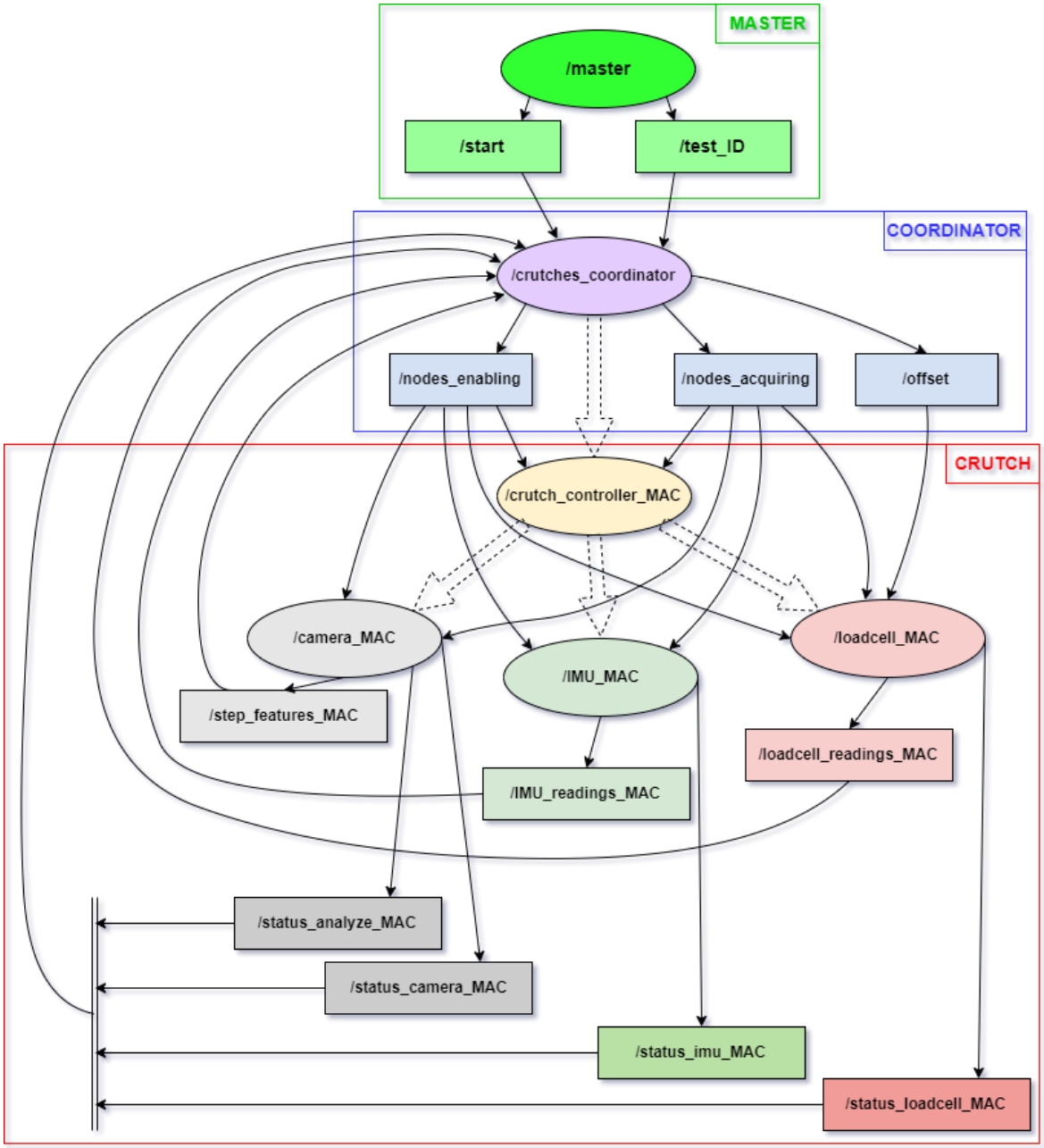


Figure 9 - ROS network during analysis of camera depth images.

Data format

The csv files of the data are named as:

subject_X_cond_Y_run_Z_[type]. csv

where:

- X is the subject number
- Y is the condition number
- Z is the test number
- [type] is the saved data type

The saved csv files use of the ";" as a separator.

Data format - Force

- Subject_X_cond_Y_run_Z_wrench_LeftCrutch.csv
- Subject_X_cond_Y_run_Z_wrench_RightCrutch.csv

Format:

time	force
Value	Value

Example:

time; force
0; 1.67
0.02; 3.5
0.04; 3.95
…….

Data format - IMU

- Subject_X_cond_Y_run_Z_imu_LeftCrutch.csv
- Subject_X_cond_Y_run_Z_imu_RightCrutch.csv

format:

time	acc_x	acc_y	acc_z	gyro_x	gyro_y	gyro_z	mag_x	mag_y	mag_z
value	value	value	value	value	value	value	value	value	value

Example:

time; acc_x; acc_y; acc_z; gyro_x; gyro_y; gyro_z; mag_x; mag_y; mag_z
0.03; 1.67; 9.81; 3.4; 1.2; 5.74; 4.36; 0.87; 0.64; 0.12
…….

Data format - Realsense

- Subject_X_cond_Y_run_Z_stepFeature_LeftCrutch.csv
- Subject_X_cond_Y_run_Z_stepFeature_RightCrutch.csv

format:

time	perc5	perc25	perc50	perc75	perc95
value	value	value	value	value	value

Example:

```
time; perc5; perc25; perc50; perc75; perc95
0.008; 12.54; 548.44; 695.61; 845.15; 742.23
…….
```

Data format - Information

- Subject_X_cond_Y_run_Z_info_LeftCrutch.csv
- Subject_X_cond_Y_run_Z_info_RightCrutch.csv

format:

mac	hostname	loadcell_enable	camera_enable	imu_enable	force_offset	hole	length	start_timestamp_ros
value	value	value	value	value	value	value	value	value

Example:

```
mac; hostname; loadcell_enable; camera_enable; imu_enable; force_offset; hole; length; start_timestamp_ros
b8:27:eb:81:73:56; left-crutch; True; False; True; 4864.65; 7; 0.870; 169751486544456
```

- **mac**: MAC address of the raspberry
- **hostname**: the hostname assigned to the raspberry
- **loadcell_enable, camera_enable, imu_enable**: enabled devices
- **force_offset**: the force offset
- **hole**: the hole to which the lower part of the crutch is fixed with the upper one
- **length**: crutch length from handle to tip (ground contact)
- **start_timestamp_ros**: value of the time at the start of the acquisition published by the synchronizer if present

Use of the user interface

The user interface (UI) that controls both crutches is installed on the Coordinator.

1. Run **roscore** on the ROS master.
2. Double click on **Instrumented Crutches** to open the UI.
3. Choose the **working mode** from the dialog window at opening and click **OK**.

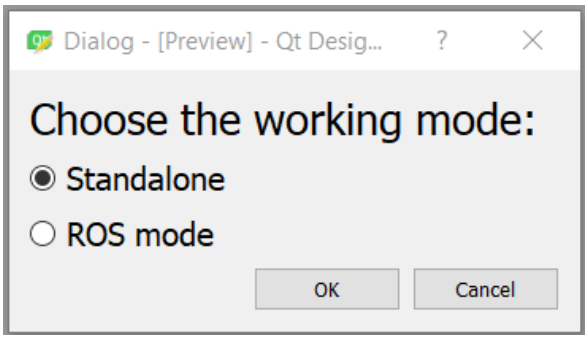


Figure 10 – Dialog window at open to choose the working mode.

Working mode:

a) **Standalone**: the start acquisition button is visible and enabled. The numbers of the subject, the condition and the run are editable.

a) **ROS-mode**: the start acquisition button is disable and not visible. The start command is sent on the "/start" topic and the numbers of the subject, the condition and the run are read from the "/test_ID" topic.

Standalone **ROS-mode**

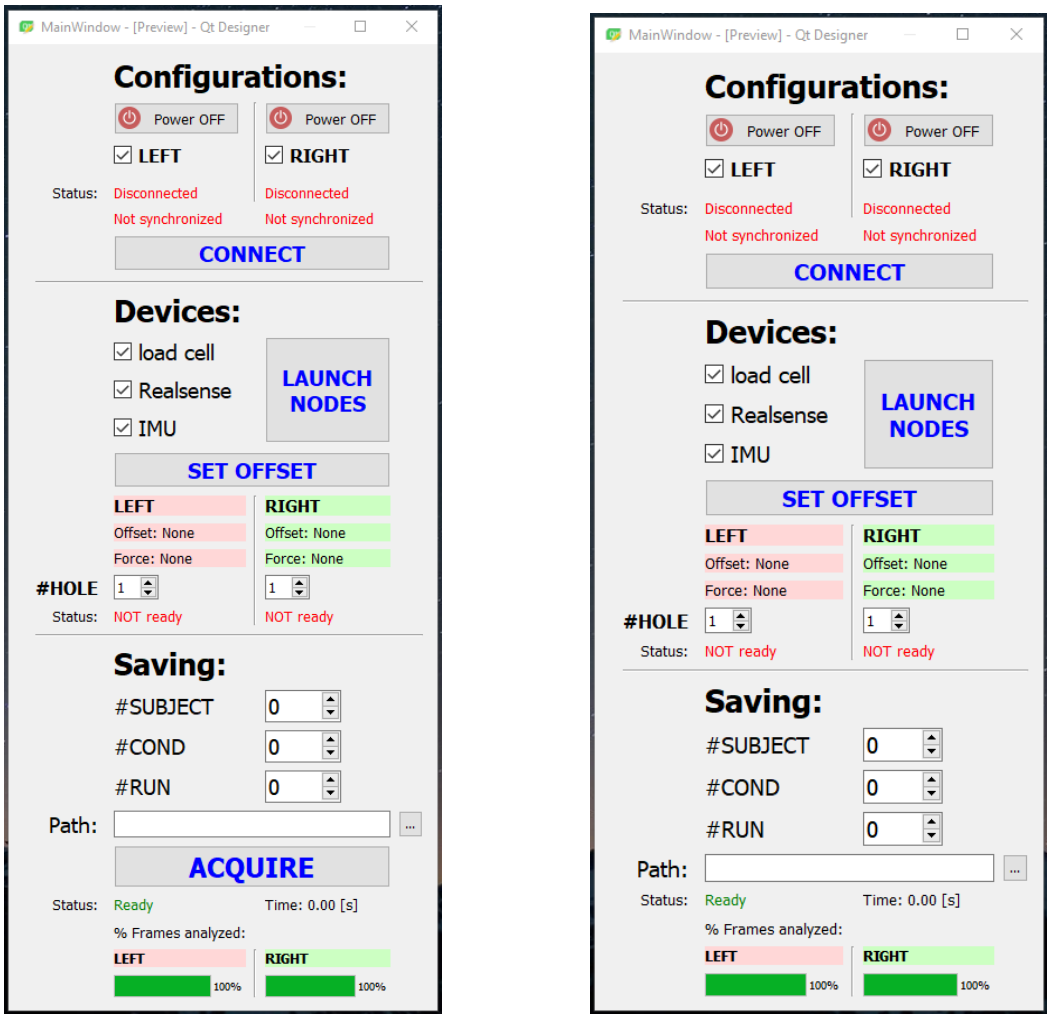


Figure 11 – UI to control the instrumented crutches depending on the working mode.

4. Turn on the instrumented crutches connecting the USB cable to the device.
5. **Wait** at least **2 minutes** to allow the date synchronization of the instrumented crutches with their ntp server.

6. Check the crutch to be enabled in the **Configurations** panel and click **CONNECT**.

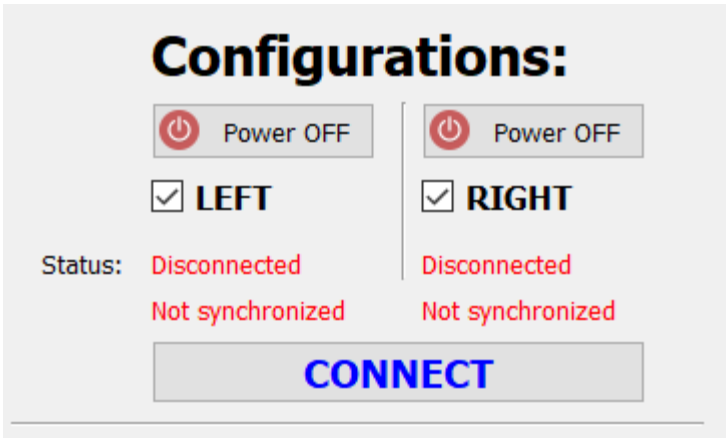


Figure 12 – Configurations panel.

Status:

a) **Disconnected**/**Connected**: indicates the connection status.
b) **Not synchronized**/**Synchronized**: indicates if the device has updated its date and time successfully.

7. Check the devices to be enabled from the **Devices** panel and click **LAUNCH NODES**.

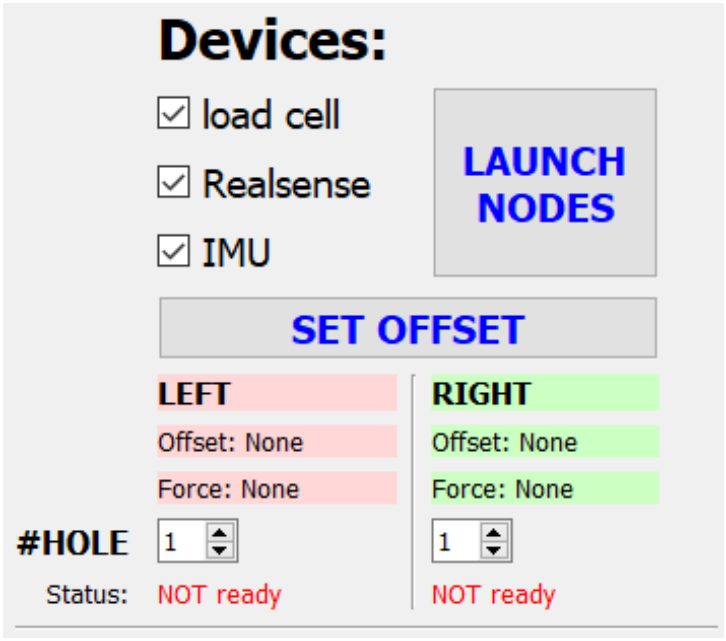


Figure 13 – Devices panel.

Status:

c) **NOT ready**/**Ready**: indicates the devices status. Wait until the label indicates "Ready" before starting the acquisitions.

8. If the load cell is enabled, click on **SET OFFSET** button to reset the force offset:
    a. **Offset** indicates the actual force offset of the load cell.
    b. **Force** shows the first 10 samples acquired after the SET OFFSET command. A value close to zero indicates a successful offset procedure (if not click again SET OFFSET).

9. Select the **hole number** of the crutches that connects the upper and lower part together. This value will be saved in "_info_CrutchLeft.csv" and/or "_info_CrutchRight.csv".
10. Configure the test proprieties in the **Saving** panel:
    a. **#SUBJECT**: subject number.
    b. **#COND**: condition number.
    c. **#RUN**: test number.
    d. **Path**: folder where the csv files will be saved.

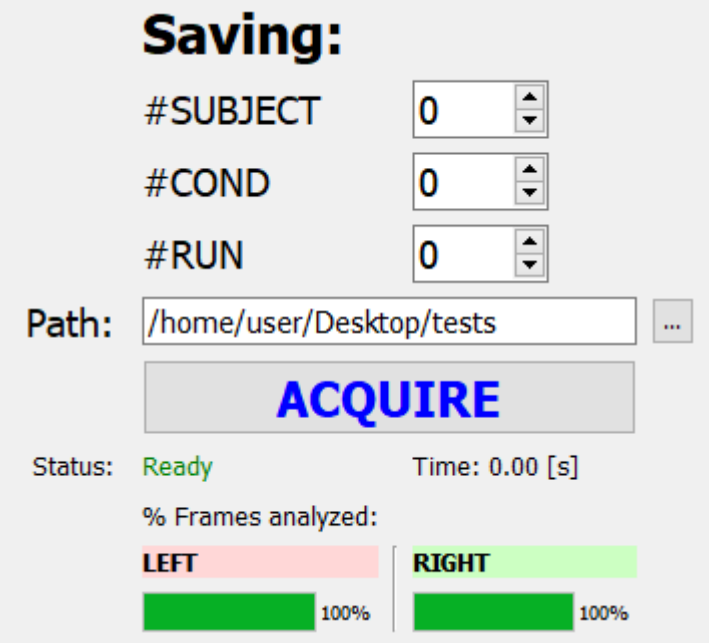


Figure 14 – Saving panel.

Status:

d) **Ready**/**Acquiring…**: indicates the acquisition status. Wait until the label indicates "Ready" before starting the **next acquisitions**.
e) **Time**: time passed from the start command.
f) **% Frames analysed**: if the Realsense device is enabled the progress bar indicates the current analysing status.

11. Click **ACQUIRE** button to start the acquisition.
12. Click **STOP** button to end the acquisition (the STOP button will appear after clicking ACQUIRE).