Pally for Doosan

User Manual

Version 1.0

Table of Contents

1 - Document Revision Information

2 - License

Disclaimer

3 - Requirements

3.1 - Software Requirements

Doosan software version 3.2.2 or higher is required.

3.2 - Hardware Requirements

H2017

3.2.1 - Supported external hardware

Lifting columns

- None at the moment, but will be supported soon

Grippers

- UniGripper CoLight Regular
- Schmalz FXCB with Foam or Suction cups
- Piab CPT series 300, 400, 500 mm with Foam or Suction cups
- Custom vacuum gripper with 24V digital signal
- Custom gripper
- Grippers with one or more individually controllable grip areas

3.3 - Compatibility notes

!	**Please read the following compatibility notes carefully before upgrading an existing Pally installation in production. Some changes in the recent versions may require new customer acceptance tests and/or modifications in the main robot program. Always make a full backup before upgrading.**

4 - Functional description

4.1 - The palletizer workflow

The typical use case for a palletizing robot is described as follows:

- The operator starts the robot and the palletizer software,
- The operator selects the product to be palletized,
- The robot starts palletizing on an empty pallet,
- When the pallet is full, the operator must replace it with an empty pallet
- Palletizing continues until the operator stops the program.

In addition to the above described default use case, the following scenarios are also available in Pally:

- Finish the current pallet immediately and begin a new empty pallet,
- Build a partial pallet, i.e. palletize fewer boxes and/or layers than usual,
- Continue an incomplete pallet, i.e. specify the starting layer/box position,

---

Note: Every pallet is built of one specific product type. Thus, it is not possible to palletize different products on the same pallet.

4.2 - Pallet patterns

Every product has its own pallet layout, depending on the box dimensions and the label position on the boxes. Products can be palletized using interlocking layers, which means that every second layer is mirrored or rotated in order to improve pallet stability. Other products have to be palletized so that the boxes are put exactly on the top of each other. The palletizer software can handle these requirements by defining the pallet pattern.

Note: A pallet pattern is the logical representation of a pallets layout, which can be dynamically loaded into the Pally software as a Pally program. The number of supported Pally programs are only limited by the available disk space.

4.3 - Pallet completion state

While pallet patterns define the expected layout of the pallet as a static model, the pallet completion state follows the actual status of the pallet currently being stacked. The pallet completion state keeps track of which exact box positions are already done, and which are waiting to be filled. The completion state is updated every time the robot has successfully moved a box from the pickup position and released it at the target position on the pallet.

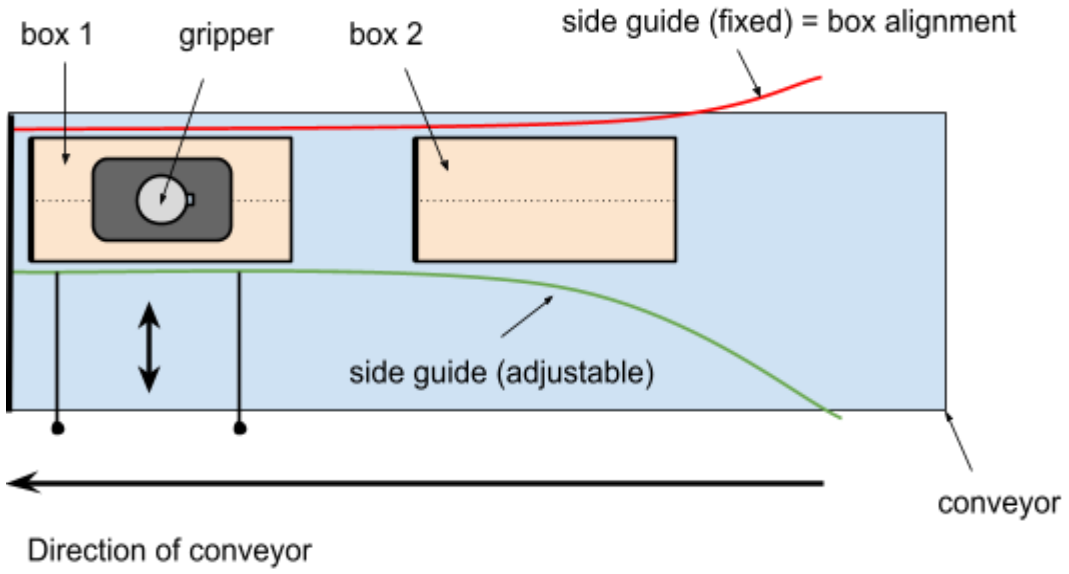
Note: The robot will not be aware of any boxes that are moved to, on or from the pallet by the operator.

4.4 - Pickup position

The pickup position is at the end of the conveyor belt, where the boxes stop and wait until the robot picks them up.

Note: The pickup position must be stationary, i. e. the conveyor should be fixed to the floor.

One side of the conveyor should have a fixed side-guide, which the boxes can be aligned to. The other side-guide should be adjustable for different product sizes where needed. Depending on the box weight and dimensions, the robot can lift up multiple boxes at the same time. When multi-grip is available, the robot will align the gripper equally between the boxes to maximize gripping stability.



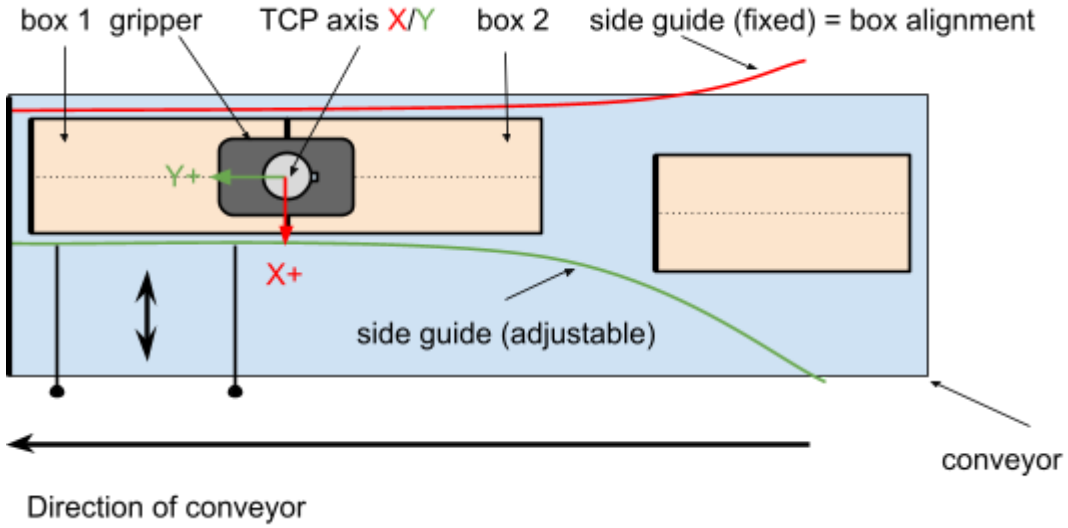
Pickup position with one product.

4.4.1 - Single-pick and multi-pick

Depending on the actual pallet pattern and gripper dimensions, it is possible to pick one or more boxes at the pickup position. To pick multiple boxes, the positions in the pattern file must follow each other by product length. The pick and place positions are automatically calculated for the given number of boxes.

The program will automatically retry with fewer boxes when the specified number of boxes cannot be palletized:

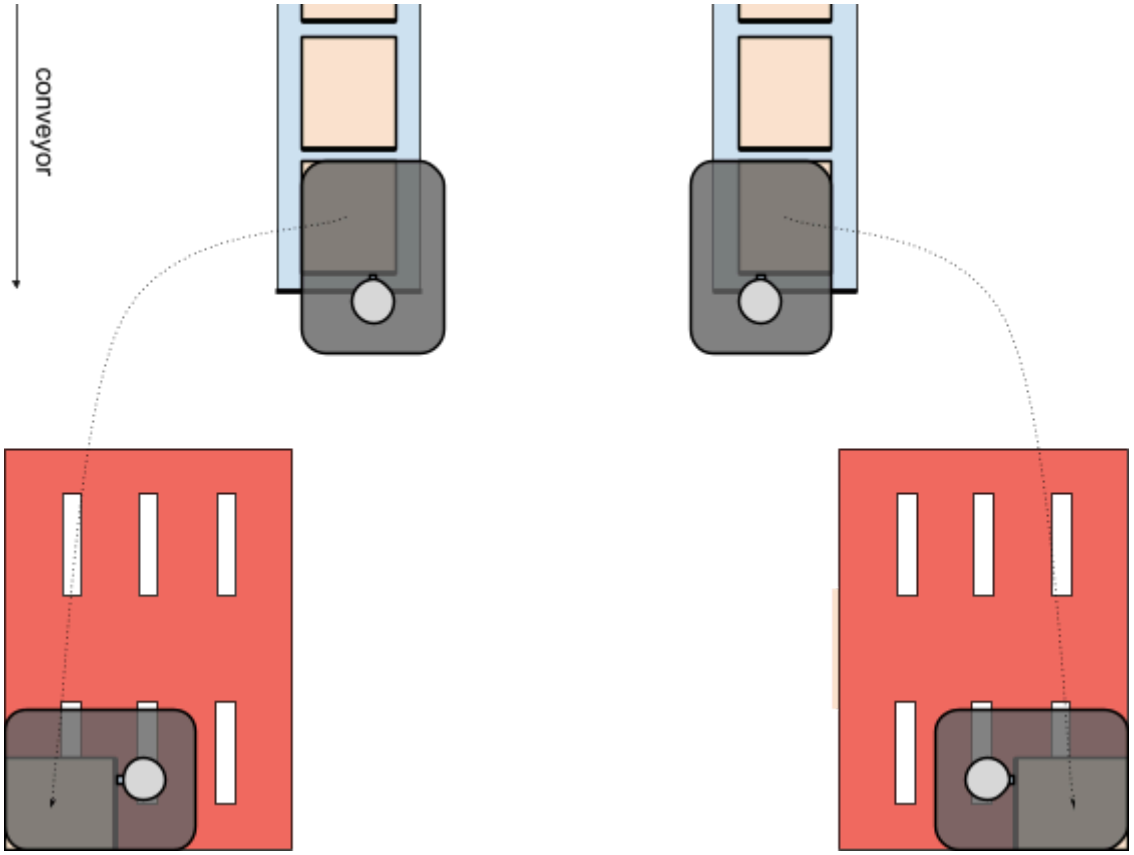
- when the gripper is too small to pick all boxes
- the pick position is out of reach
- the target position is out of reach
- collision-free path planning is not possible



Pickup position with two products. Notice that the tool coordinate system is aligned to the conveyor direction.

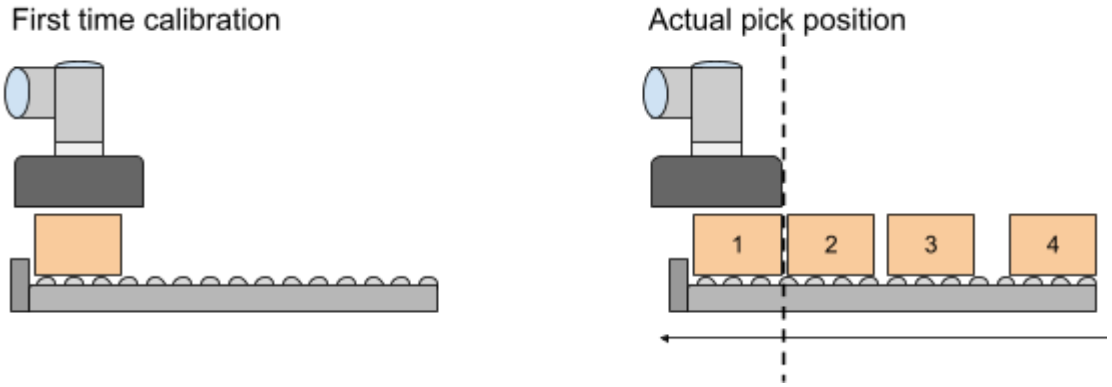
4.4.2 - Gripper alignment at the pickup position - offset grip

To avoid accidental multi-picking, the front edge of the gripper is default aligned with the rear edge of the (last) product being picked. This is especially important when boxes are smaller than the gripper. In this case the gripper is not aligned centered above the box, which is a common misunderstanding when using Pally for the first time.
If you use a gripper smaller than the product you can turn off the edge alignment to ensure the gripper always picks the product in the center. This can be particularly useful when using a mechanical gripper. You can do this by setting the variable rf_gripper_align_to_edges to false.



Edge alignment optimization for large grippers

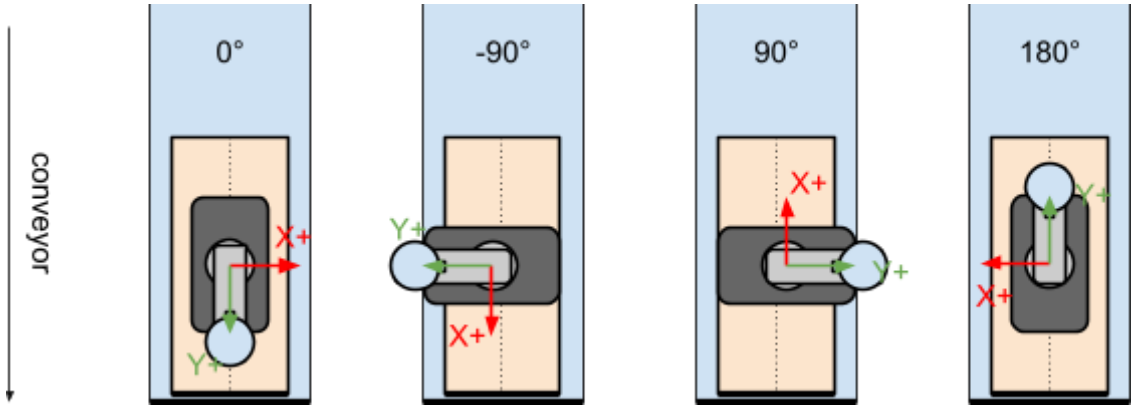
The first time calibration requires the gripper center aligned to the box center, as shown in the following diagram. In contrast to the calibration process, the real pick position will be calculated with respect to the edge alignment (when needed).



Gripper alignment at pickup

4.4.3 - Gripper rotation at the pickup position - gripper optimization

In order to get better reach or performance, the program can automatically recalculate and use the gripper rotated by +90, -90, 180 degrees at the pickup position. This may be necessary to reach the pallet corners or avoid collision with the base frame. The allowed rotations can be configured.



Gripper rotation at pickup

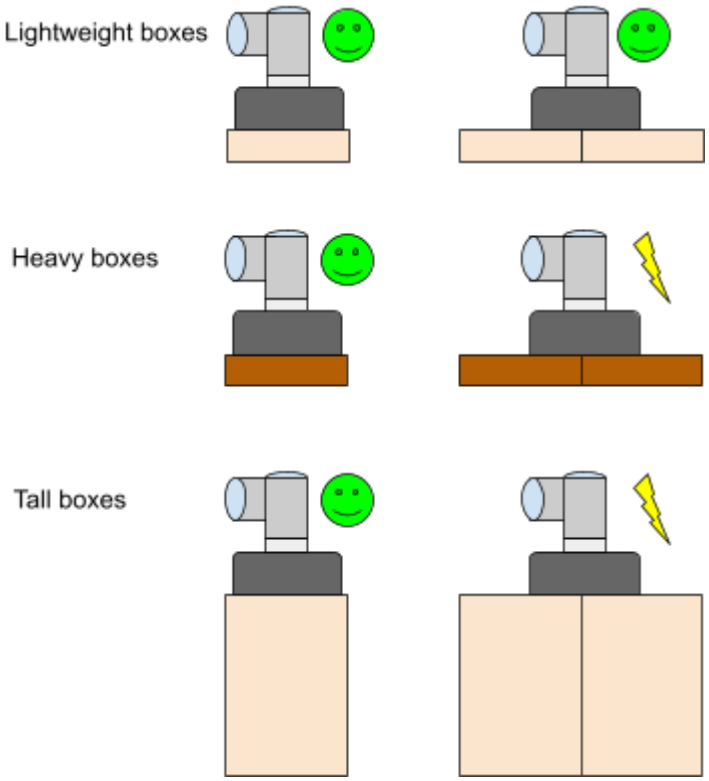
There are 3 default optimization modes to choose from: 4-ways, 2-ways, and no optimization, which are listed below.

None	only 0°
2-ways	0° and 180°
4-ways	0°, -90°, 90°, and 180°

The optimizer combines gripper rotation and gripper alignment in order to maximize reach on the pallet. Which edges of the (possibly rotated) gripper are aligned to which edges of the box(es) depend on the actual target rotation on the pallet.

4.4.4 - Grip quality estimation

Based on the actual product and gripper dimensions, payload weight, and the number of boxes being picked, the program estimates a reference value called 'grip quality' which is used by the optimizer to choose between possible gripper rotations, reduce the number of boxes to be picked in one robot cycle and reduce speed and acceleration.



Grip quality estimation

Grip quality in Pally is a numerical value that tells about the probability of losing boxes during transport. Usually, a higher grip quality implies a more reliable box transport between the pick and drop position. In comparison, a lower grip quality often means the gripper won't be able to hold the box.
Grip quality is not an absolute measure and has no unit. But it can be used by the optimizer to compare different picks.
Grip quality estimation has a significant impact on the following program behavior:

- Can the robot pick more or fewer boxes in one cycle?
- Which gripper rotation is best for the current pick?
- Which combination of the vacuum zones is best for the next pick?
- Can the robot move the box(es) with maximum speed and acceleration?

4.4.5 - Smart acceleration

When the estimated grip quality value drops below a minimum threshold, the program moves the boxes with reduced acceleration in order to minimize the risk of dropping boxes during transport. The parameters of this algorithm can be configured.

The program will try to choose a gripper rotation that has higher grip quality value. If the grip quality is still low, the program will reduce accelerations to avoid lost boxes. The grip quality minimum threshold is project and product specific, and provided as a tuning parameter.

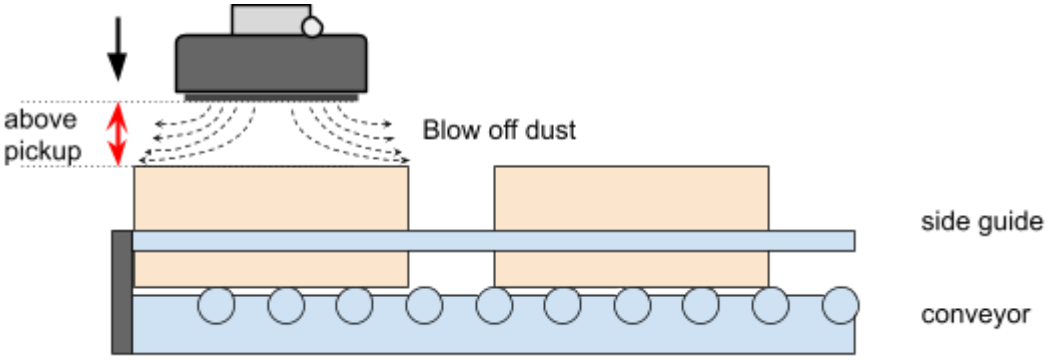
4.5 - Path planning

The path planning algorithm is an essential part of the palletizer application and is done with the online tool MyRobot.cloud. It calculates the optimal movement from the pickup position to the pallet position. These calculations are performed dynamically when the robot moves one or more boxes from the pickup to the pallet position.
After picking the products, the robot moves vertically up until the box is removed from between the side guides. Depending on the current pickup and target positions, this is followed by one or more waypoints in order to move and rotate the box(es) without collision to the robot base or the existing boxes on the pallet. The approach position is the last waypoint in the proximity of the final target position. The robot uses lower acceleration and speed from the approach to the final target position to improve the pallet accuracy.

Note: To ensure a smooth palletizer experience, some properties of the production line should be carefully measured.

4.5.1 - Above pickup position

The path starts with a position above the pickup point, to ensure the gripper moves vertically down against the box surface. At this point the program can automatically activate compressed air (if available) to clean the gripper foam and the box surface before picking.

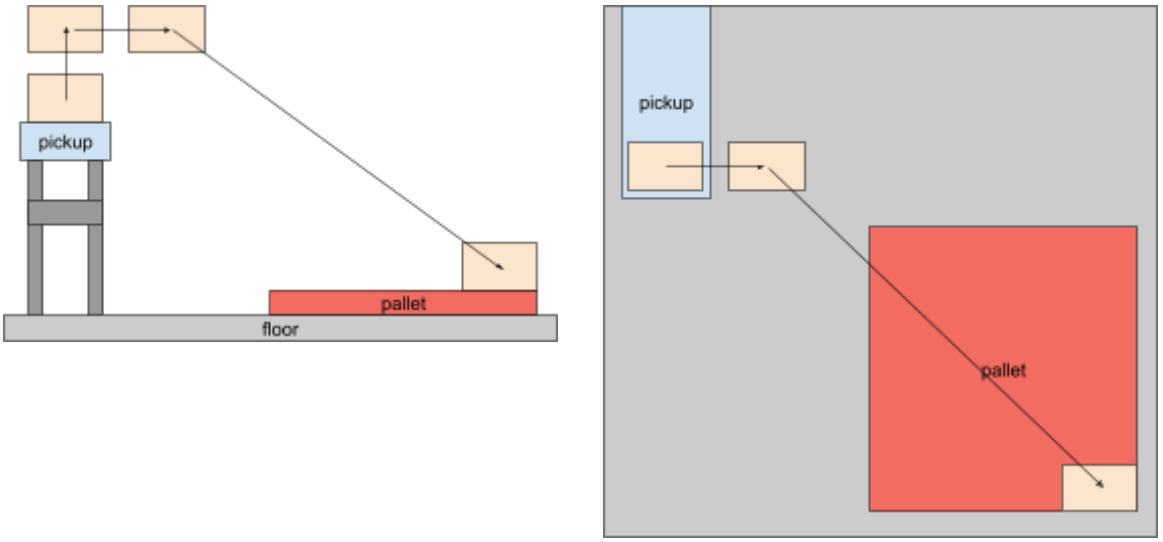


Gripper approaching box vertically from above

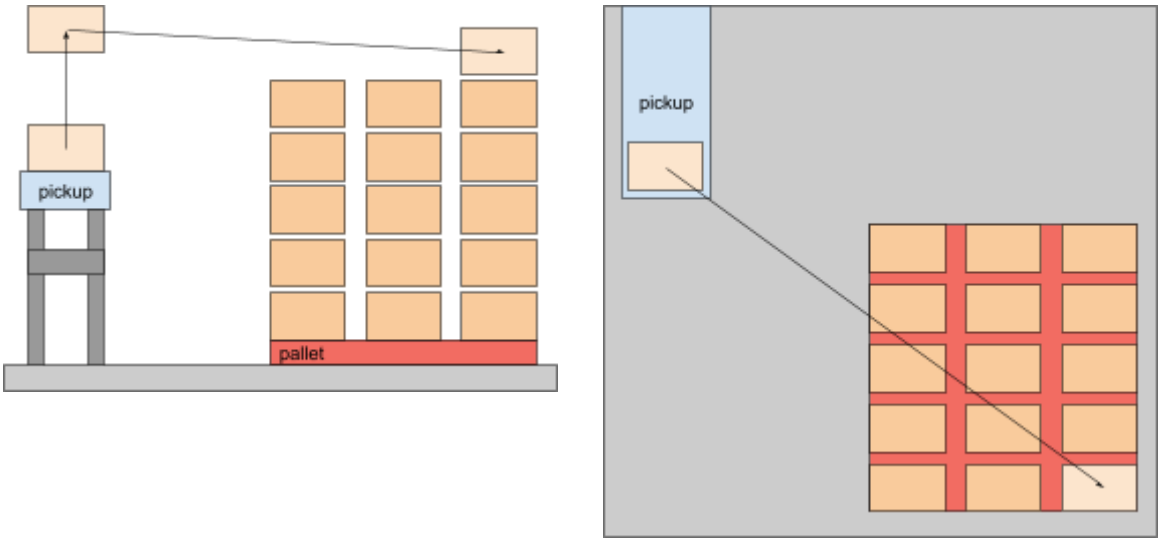
4.5.2 - The path planning algorithm

The purpose of the path planning algorithm is to move the box from the pick position to the pallet position without collisions and as fast as possible.
The path planning is done in the online tool MyRobot.cloud.
The path planning result depends on several settings that are introduced below.

First, the path planning algorithm will try to move the box on a linear path whenever possible. After the box is taken from the pickup position, the algorithm tries to find one waypoint - called *smart exit* position - from where the shortest, direct linear movement is possible.



The primary path planning algorithm on lower layers.



The primary path planning algorithm on higher layers.

It is possible to define a rectangular area where the path planning algorithm can search for an intermediate waypoint (smart exit) between box-free and approach position, to perform linear movements without collision to the robot base, see the green areas.

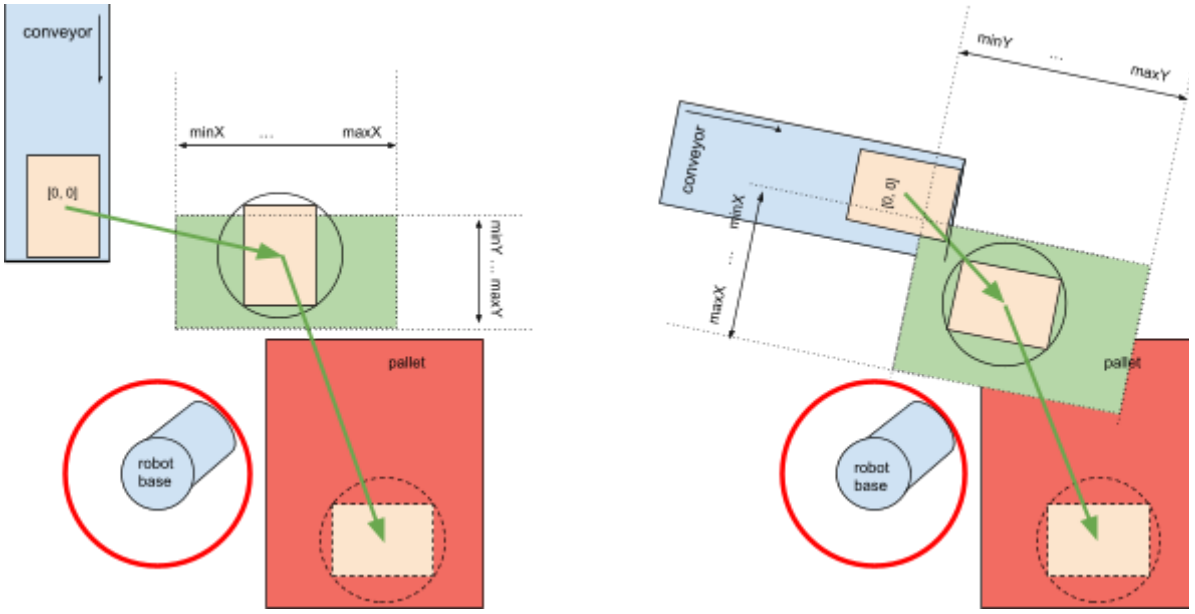
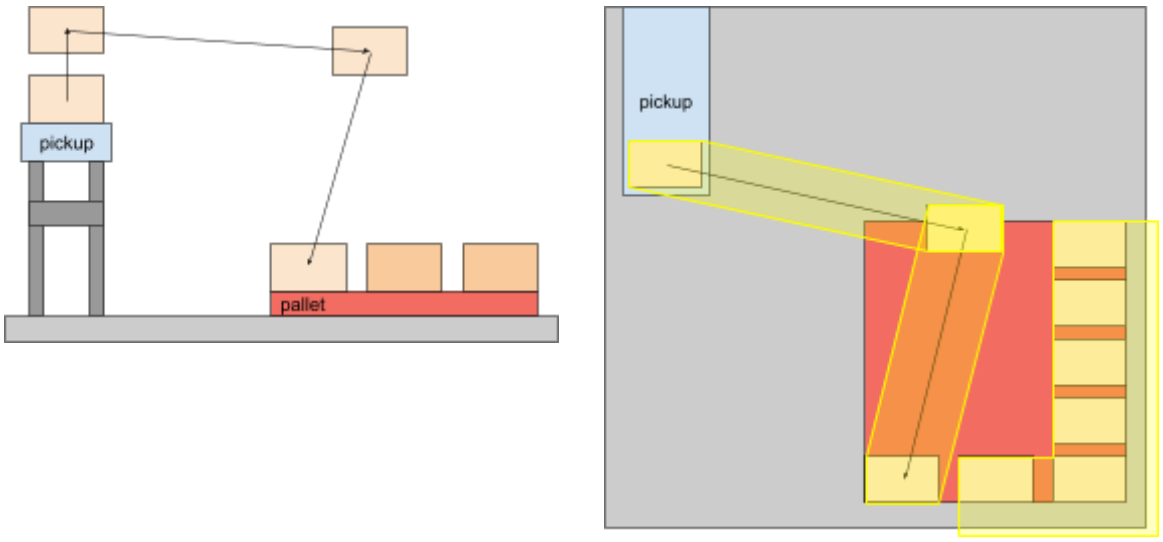


Illustration of the *smart exit* search area.

When it is not possible to find a *smart exit* position from where a direct linear movement is possible, one or more waypoints are being inserted. In this case the program evaluates the pallet completion state (position of the already palletized boxes) and finds a collision-free path to the target position. It is often necessary to move the box above all other boxes, and then lower vertically to the target position. This path is normally longer than a direct linear movement, and hence more time demanding.



Path planning with an extra waypoint. Boxes on the same layer define a collision area to be avoided.

When the target position is below the conveyor level, it is often necessary to keep the box as high as possible until the robot leaves the conveyor area. This behavior can be configured via the *high approach boost* settings: lower values let the box move closer to the top of the other boxes on the pallet, higher values keep the box closer to the *box free* height.

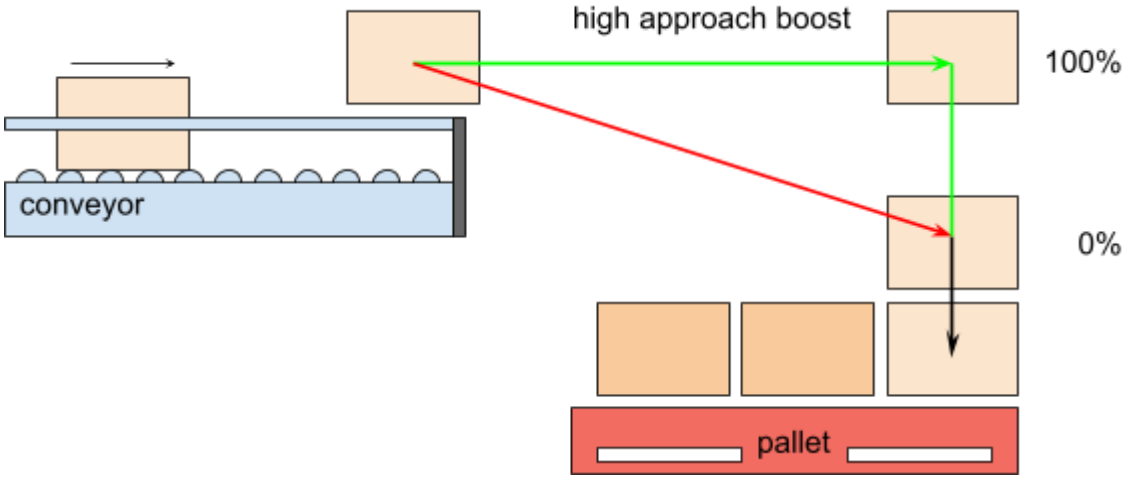


Illustration of the *High approach boost* parameters.

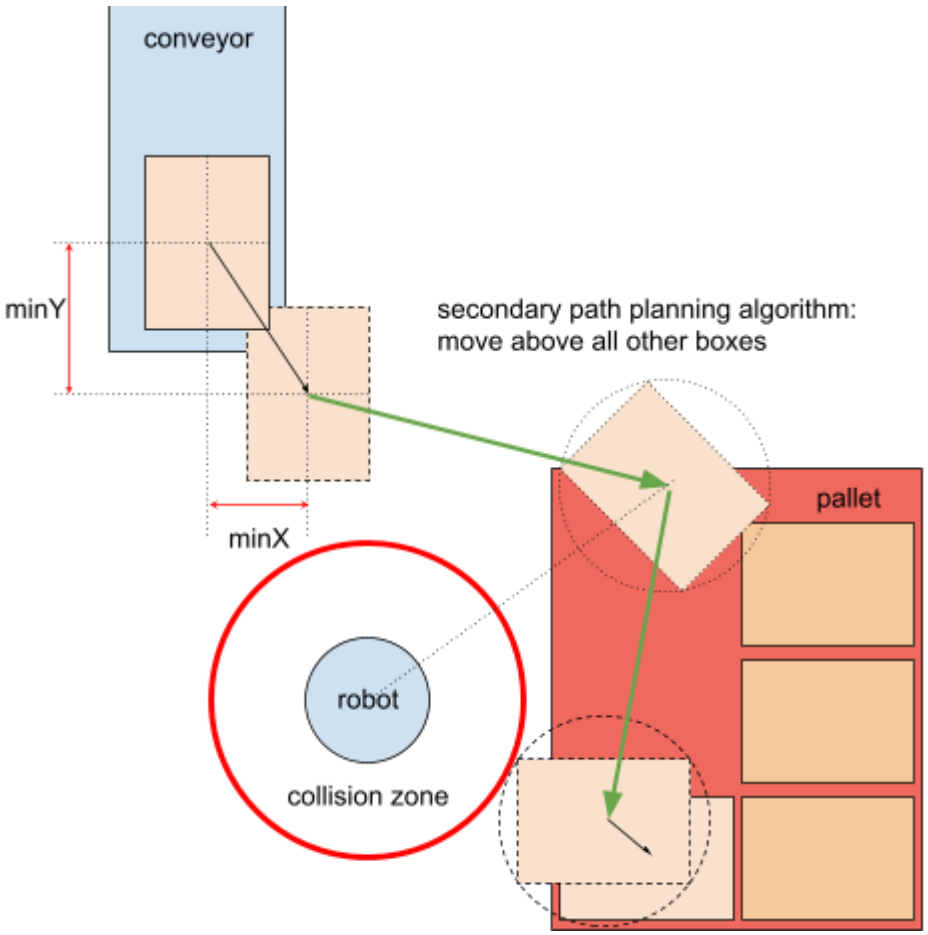
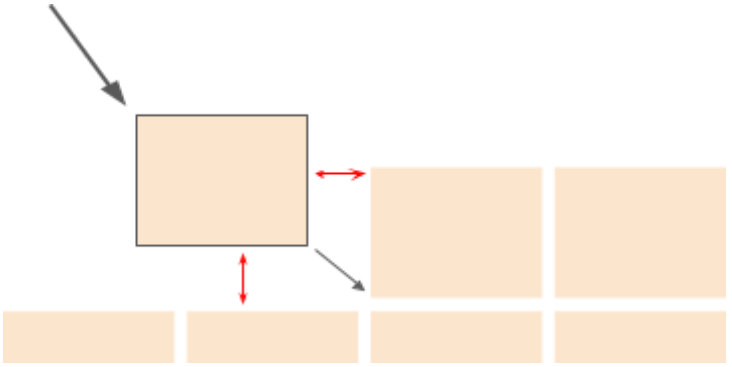


Illustration of when the High *approach* is used.

4.5.3 - Approaching the target position

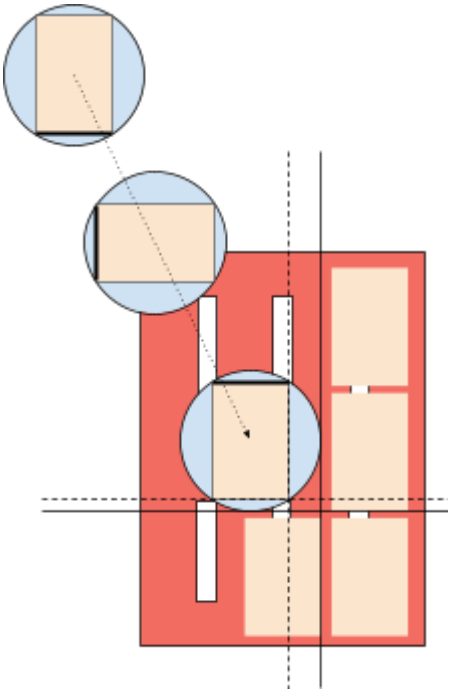
In order to improve precision, the last section of the movement is being performed with lower speed and acceleration.
Normally the robot moves the box into its final position by using the "Approach" distance.



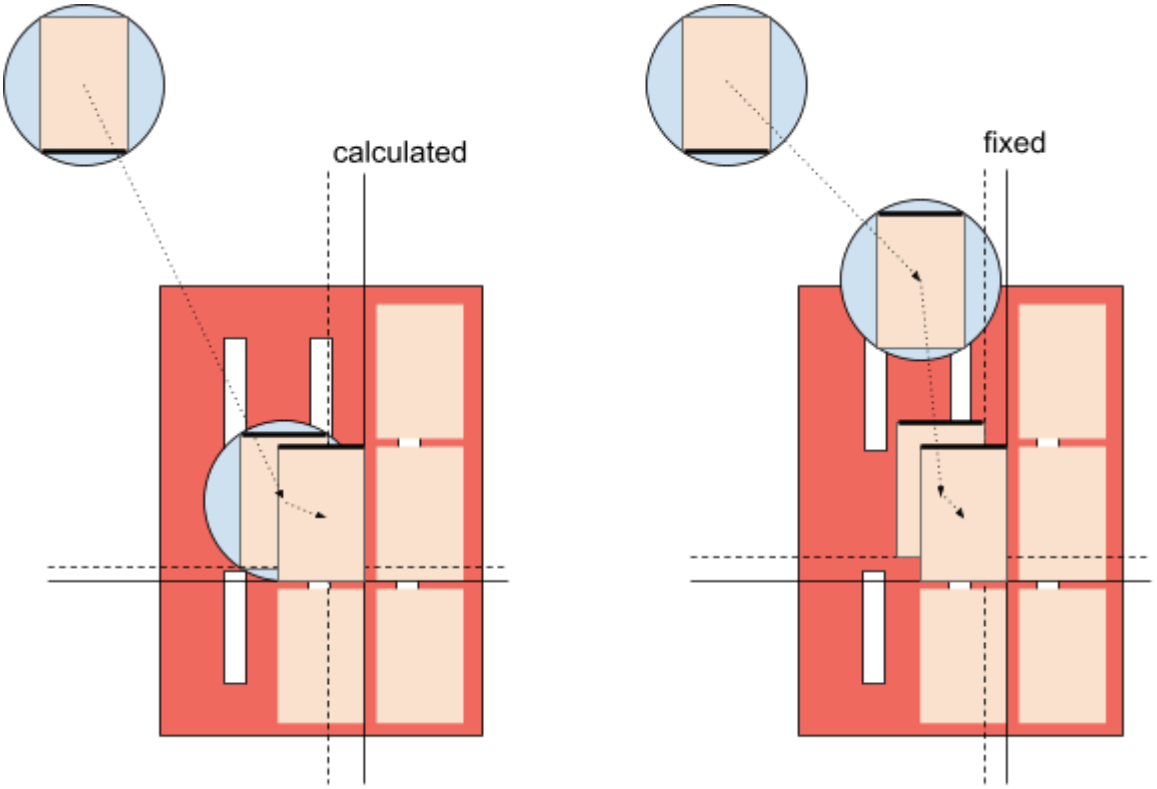
Approaching the target position with lower speed and acceleration

---

Note: To keep the necessary distance from the existing boxes on the pallet, the approach position may be recalculated dynamically if rotation occurs between the pickup and the target position. This default behavior can be changed by selecting 'fixed' approach.

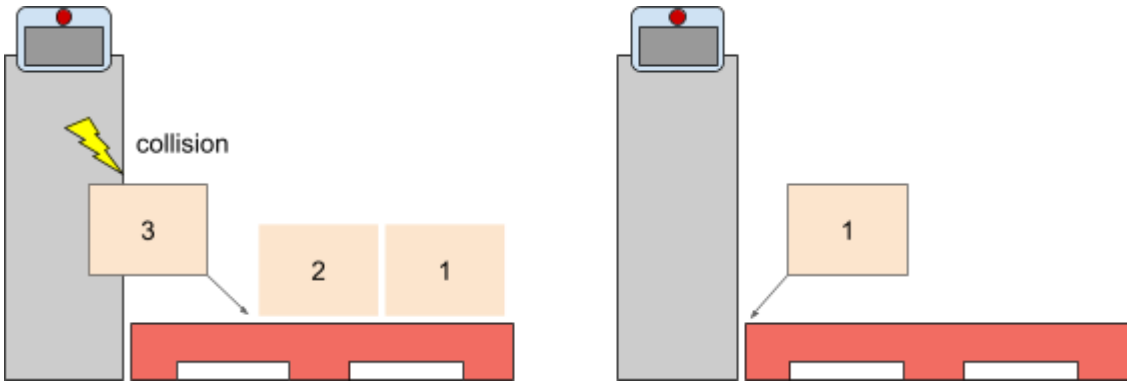


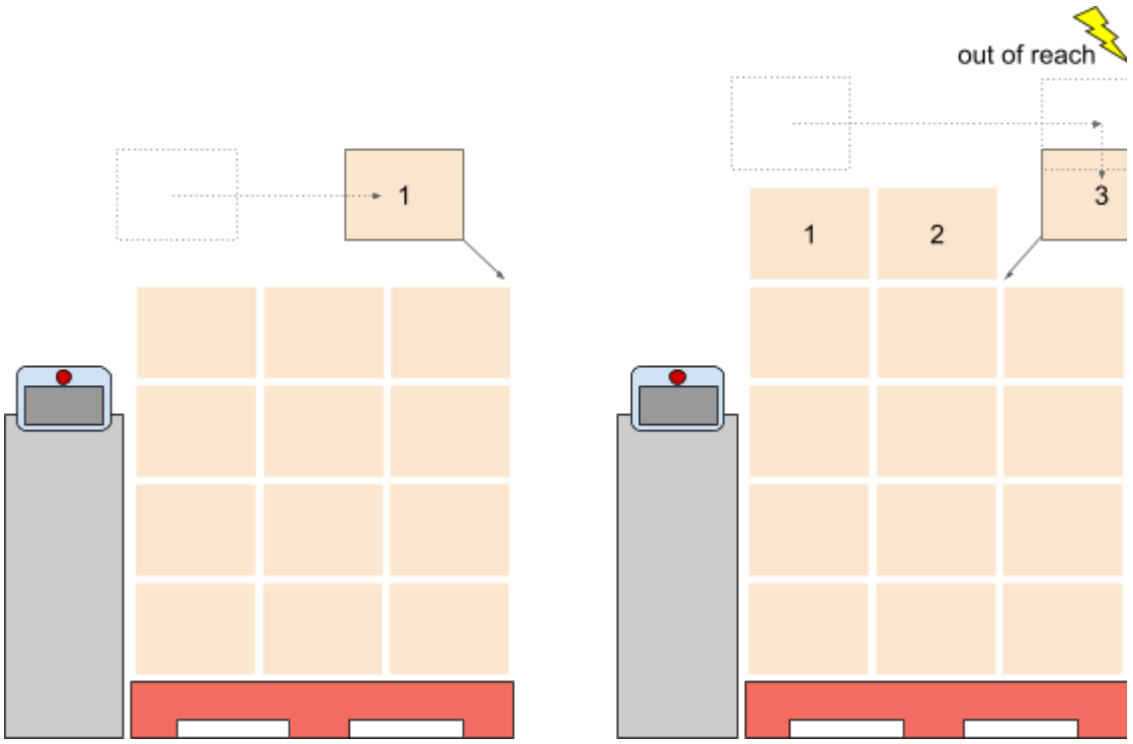
Dynamically calculated approach distance for rotating boxes.



Difference between dynamically calculated and fixed approach distance.
Please note the extra waypoint added to complete rotation before approaching.

When palletizing the lower layers, the robot may not have enough room between the mounting base and the box position to keep the approach distance - the box or the robot arm can collide to the base and get damaged.
To work around this problem, it is possible to change the approach direction to inverted approach and consequently the order of boxes in problematic layers.

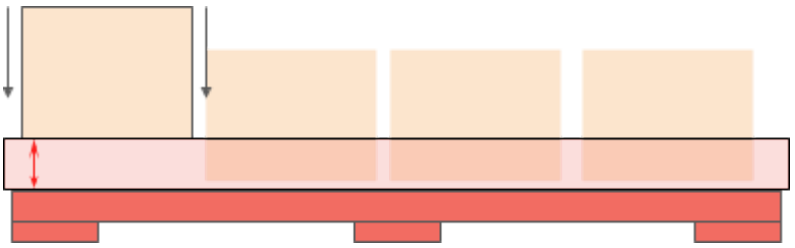




Pros and cons of normal and inverse approach. Notice the impacts on reach, collisions, box order, etc.

4.5.4 - Pallet lip

When pallets have an outer edge lip, the robot can enforce a vertical movement on the first layer to avoid collision with the pallet lip.



Vertical movement from above the pallet lip

4.5.5 - Return path

After releasing the box at the target position, the robot visits the path waypoints in reverse order, to return from the pallet position and pick the next box(es) from the conveyor.
In a typical setup, the robot waits above the pickup position until the required amount of boxes are available.
It is possible to modify the return path by removing one or more waypoints or implementing a user defined path.

4.5.6 - User-defined path

It is possible to override the default Pally path planning by implementing custom movement for one or more specific box positions, or even replace the default path planning in the entire project. It is also possible to implement custom code to sort out boxes from the conveyor that should not be palletized, based on a project-specific signal or quality check. Refer to the callbacks section for further details.

4.6 - Payload control and robot stabilization

Collaborative robots are generally very sensitive to the weight and center of gravity of the attached payload. The robot settings must always correspond to the actual payload in order to work properly. Changing the payload settings too early or too late will result in instant protective stops due to false collision alarms. Especially in palletizer applications where heavy boxes are picked and released, it is essential to manage payload settings as accurately as possible.
The program has built-in routines for monitoring errors in the robot position, which may occur due to sudden changes in the payload. During picking and releasing boxes with a vacuum gripper, the program can automatically adjust the delays between switching the vacuum valve(s) and updating the payload weight and center of gravity. This can significantly reduce the occurrence of unexpected protective stops at the pick and drop positions without affecting robot safety requirements.

Note: This function may introduce small delays when picking and releasing boxes. Some tuning parameters are provided for adjusting or disabling the function.

4.7 - Shim papers

Shim papers are considered as special layers with no products. These layers have their own height - i.e. the thickness of the shim paper itself. Pally does not include a default implementation for shim paper placement - this has to be implemented for each project by using the onSheet callback.

4.8 - Lifting column

Not supported yet

4.9 - Zones

Normally the robot completes a layer before starting the next layer. However, the pallet can be split into different regions - called zones - along the X, Y, and Z-axis. As a general rule, a zone cannot be started until the previous zone is finished.

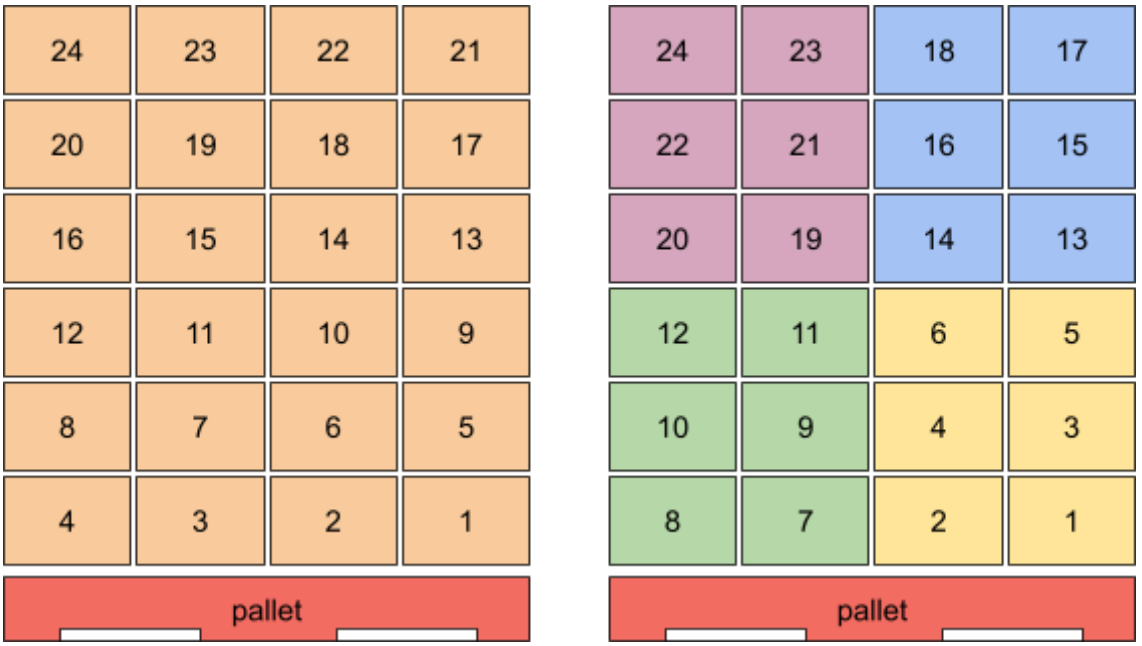
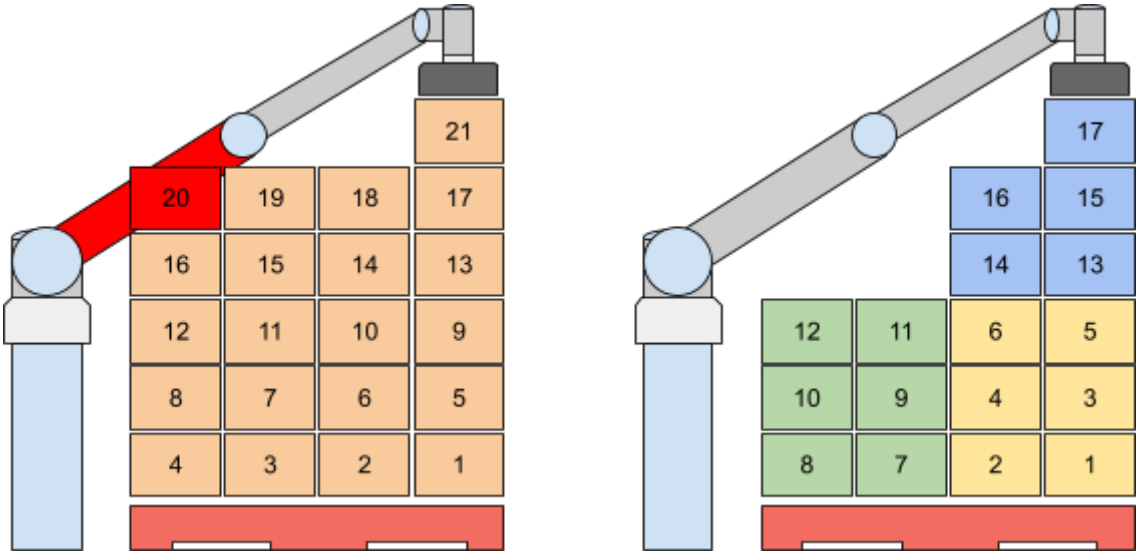


Illustration of how zones divide pallets into smaller regions.
Each color indicates a different zone.

Zones is one of the most powerful features and can be used to solve problems that could not have been done otherwise, e.g.:

- Avoid collision between upper arm and boxes nearest to the robot on the top layers,
- Take full control of the lifting column positioning,
- Control external machines, for example, start a stretch wrapper to wrap the pallet after N layers are completed.



Using zones to avoid collision by changing the palletizing order.

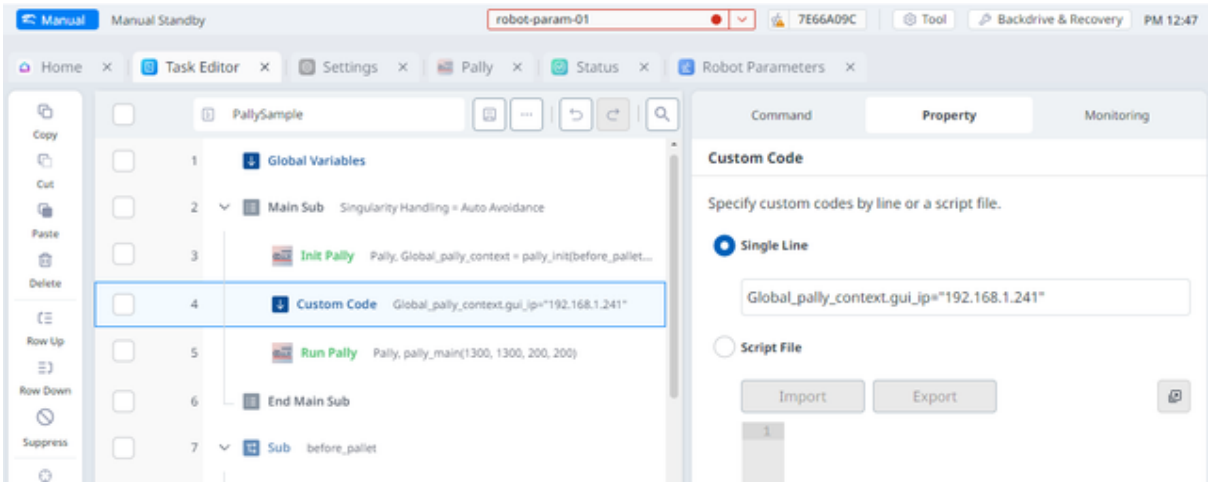
Zones require a special configuration file. This file will describe the conditions for splitting the pallet into smaller regions. For further information, we refer to the online Pally knowledge base. The zone configurations are transferred to the robot along with the regular pallet patterns.

4.10 - User extensions - Callbacks

It is possible to extend the functionality with user-defined commands, which are automatically executed when specific events occur.
You get access to this by defining a global variable for Pally and initialize it to **0.**

The Init Pally program node will return a variable called Global_pally_context which points to an object with configuration data inside the Pally module.

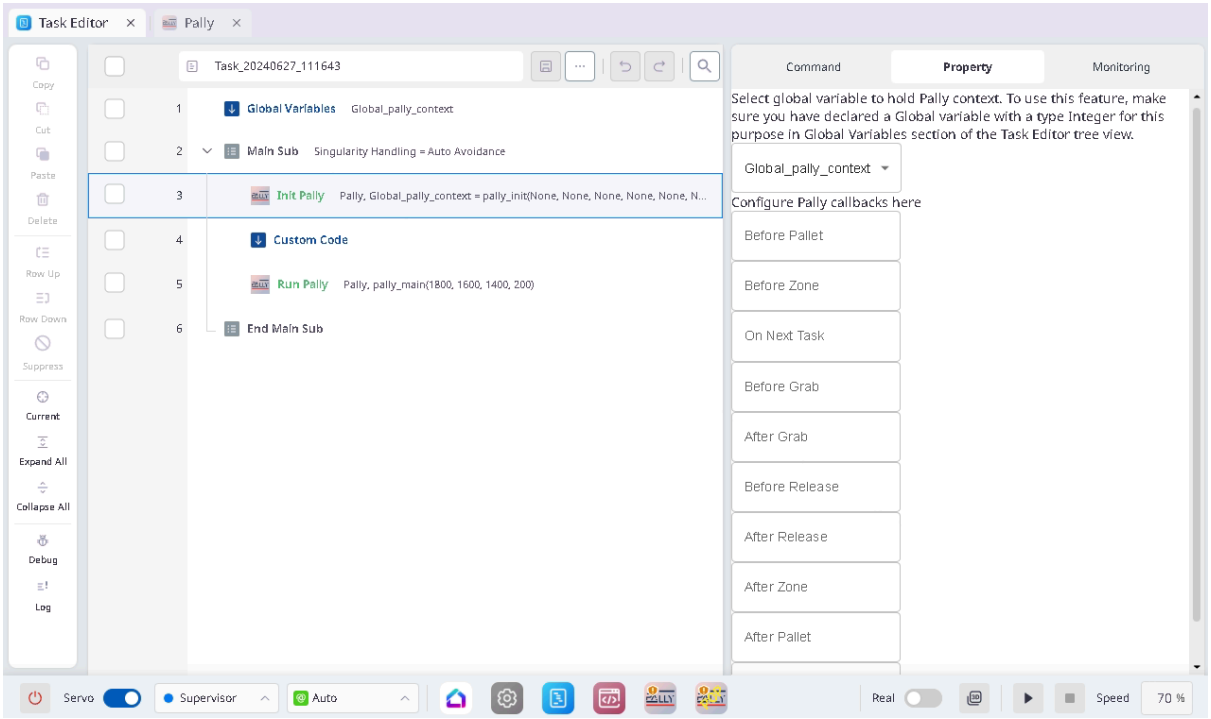
You can use Global_pally_context anywhere in the callback subroutines or below the Init Pally program node in the main program to change any Pally settings in runtime.



Currently the following events are available:

- **beforePallet**: Before starting a new empty pallet.
- **beforeZone**: Before entering a new zone.
- **onNextTask**: Before starting all calculations for the next box(es)
- **beforeGrab**: Before lifting up a box from the pickup position.
- **afterGrab**: After lifting up a box from the pickup position.
- **beforeRelease**: Before releasing a box on the target position.
- **afterRelease**: After releasing a box on the target position.
- **afterZone**: After leaving a completed zone.
- **onSheet**: When a shim paper needs to be inserted.
- **afterPallet**: After the pallet is complete.

Create your own subroutines that will be called at specific events during palletizing. Enter the name of each subroutine in the corresponding edit-box, or leave the edit box empty.



In Pally you can get access to the following variables via Global_pally_context:

- # tuning parameters
  - max_speed
  - max_acceleration
  - precise_speed
  - precise_acceleration
  - min_grip_quality
  - release_strategy
  - grip_delay
  - grip_release_delay
  - blend
  - product_selection_strategy
  - product_selection_predefined
  - force_palletizing_mode
  - trace_program_callbacks
  - set_workpiece_weight
  - set_weight_workaround
  - gripper_weight
  - gripper_tcp
  - gripper_cog
  - start_pos
  - start_jpos

- # callback context
    - Instance
    - MovePerformed
    - TaskCompleted
    - ZoneNr
    - LayerNr
    - ProductCount
    - ProductName
    - LayerAlt
    - LayerHeight
    - PalletNr
    - PalletCenter
    - MoveTarget
    - MountPosition

Example:
Global_pally_context.grip_release_delay=0.5

Example II:
movel(Global_pally_context.MoveTarget)
Global_pally_context.MovePerformed=True

Some typical examples of uses for user-defined commands are to insert a shim paper, start and stop the conveyor, sort out inappropriate boxes to a waste bin, perform custom motion in very special layouts, control light signals and signal external machines, etc.

4.11 - Manual and automated operation modes

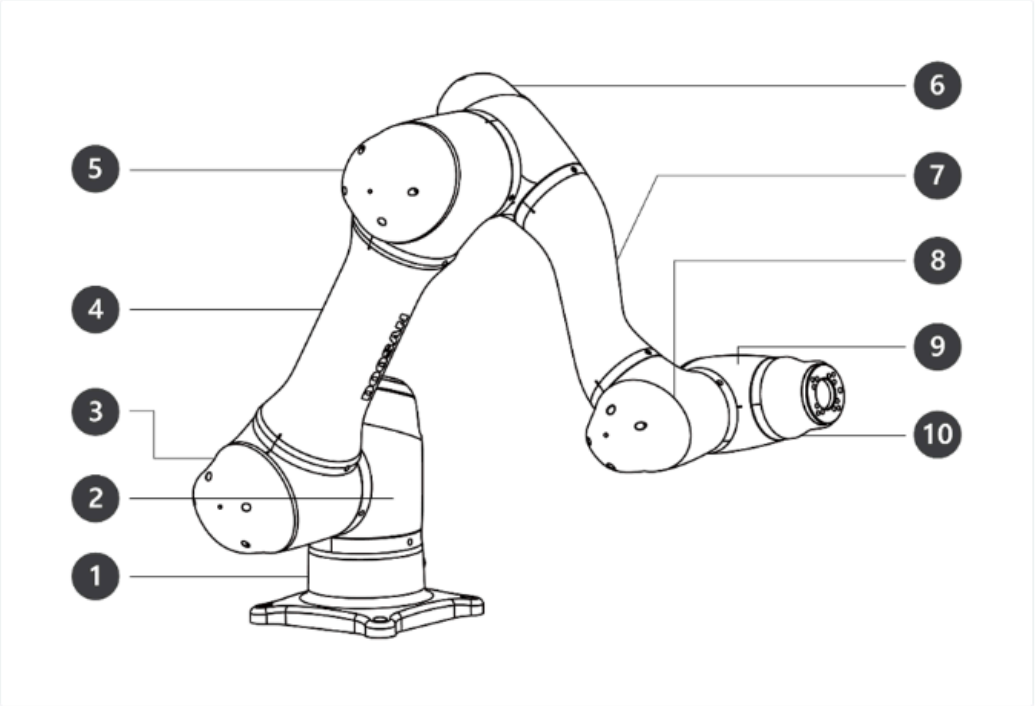
The palletizer program can be controlled manually by the operator, or automated via the corresponding variables and interfaces.
When using the program in manual mode, the operator can select the pattern to be palletized, optionally choose the number of boxes and define the starting position on the pallet.
In automated mode, it is possible to start the palletizer program with a predefined pattern, or let the operator select from a list of patterns. There are currently 3 different strategies to select patterns:

- Automatically select the (alphabetically) first pattern from the pattern storage folder
- Automatically select the pattern with a specific name, as defined in a special variable
- Manually select the pattern by the operator

5 - Physical Installation - Best Practice

5.1 - Robot installation

Names of Parts

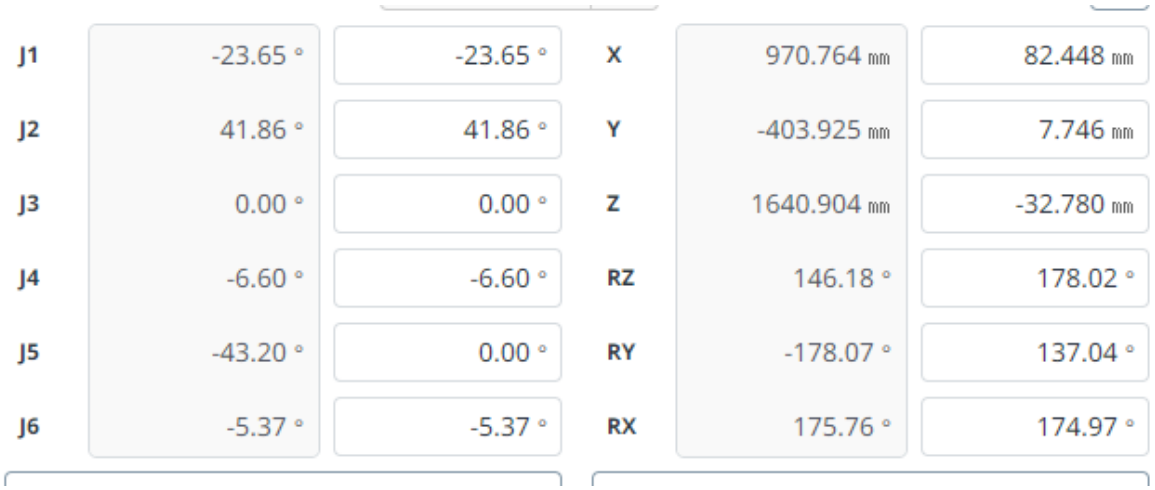


No.	Name	No.	Name
1	Base	6	J4
2	J1	7	Link2
3	J2	8	J5
4	Link1	9	J6
5	J3	10	Tool Flange

5.1.1 - Check joint positions first

The H2017 joints have a limited +-360 degrees position range, so it is essential to check whether the joint positions are not too close to their limits, before attaching the vacuum hose and other objects.

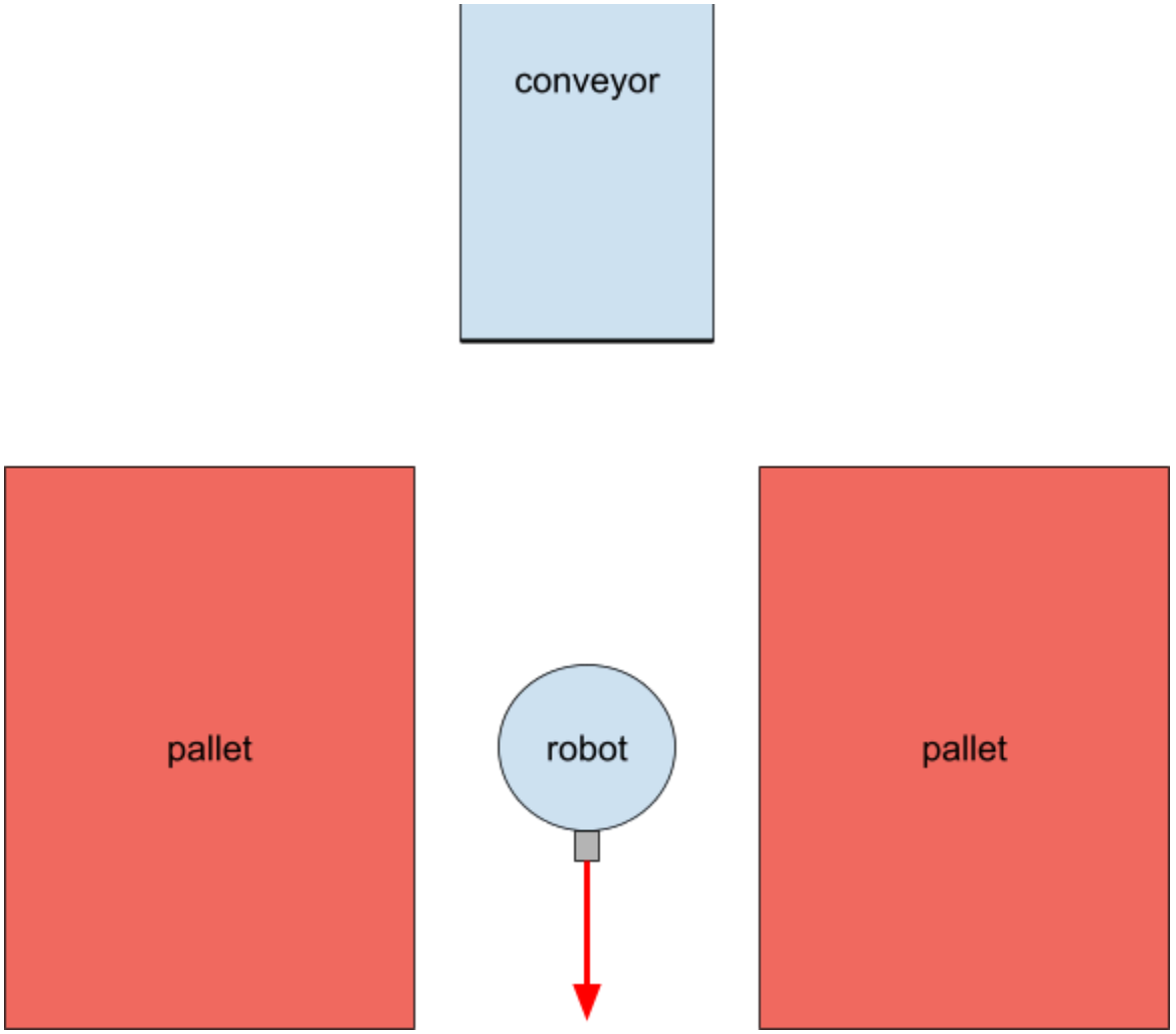
Note: The robot can be operated as "left-handed" or "right-handed". Left handed or right handed implies on which side the J4 is located, seen from the opposite side of the conveyor. For a right handed robot, the J4 is located to the right side of the base joint. For a left handed robot, the J4 is located to the left of the base joint. The examples in this document are made with a left-handed configuration.



Check the joint positions before mounting vacuum hoses and cables.

5.1.2 - Robot position on the base column

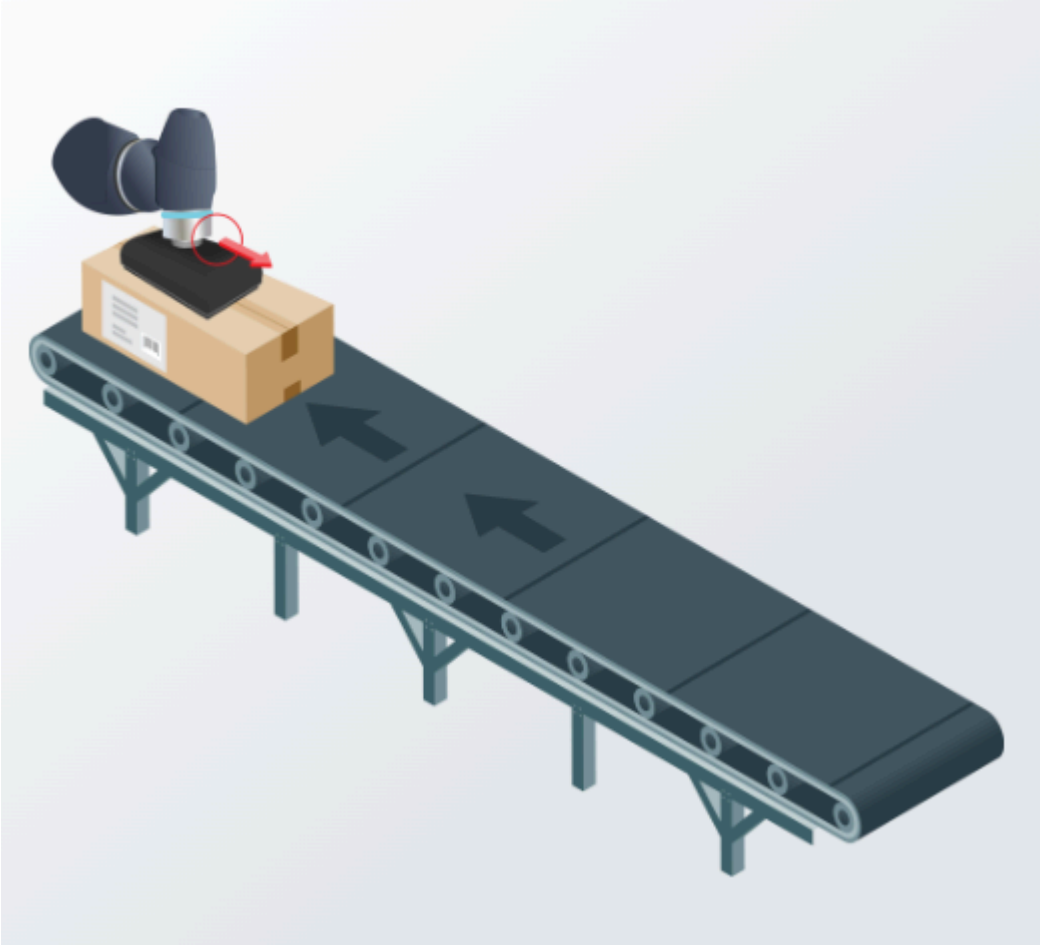
Before mounting the robot on the base column, make sure that the power connector on the Base points behind the palletizer cell. This is not mandatory, but recommended for best reachability both on the left and right pallet positions.



Check the position of the power connector.

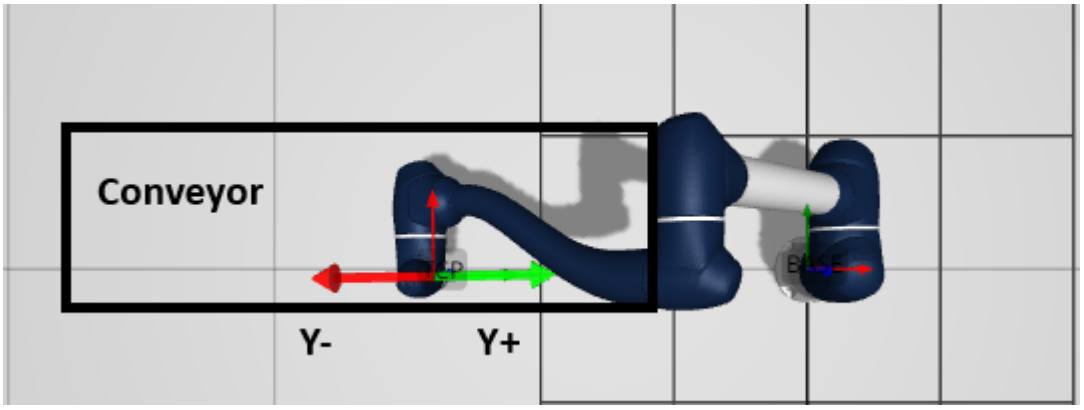
5.1.3 - Attach the gripper properly

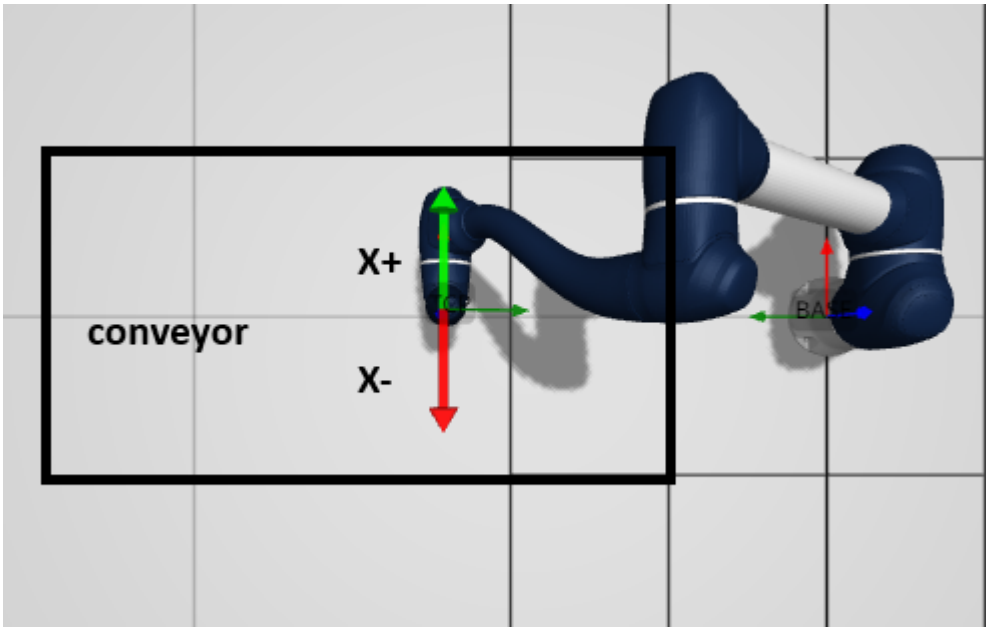
Attach the gripper when X2 connector is pointing down the conveyor.



Depending on the physical dimensions of your gripper, you might have to configure the robot tool center point under Installation / General / TCP as described below.

Note: The conveyor direction is calculated directly from the Y-axis of the tool coordinate system at the pickup position.





Make sure TCP is configured properly.

Pally supports a wide range of grippers, and some of them might have either position, rotation, or both offsets that must be taken into account when TCP is being configured. To verify the TCP settings are correct, perform the following steps:

- Move the robot to the pickup position.
- Now go to the Move tab, and select "Tool" in the drop-down called "Feature".
- Now press the Y- arrow button and the gripper should move parallel to the conveyor and in the direction where boxes arrive from, i.e towards box 2.
- Press the Y+ button, and the gripper should move towards the end of the conveyor.
- When the gripper is above the box on the pickup› position, press the RZ+ and RZ- buttons: the gripper should rotate around its own center point and not around the robot tool flange. Rotating by 180 degrees, the gripper should remain perfectly above the center of the box at the pickup position.

Note: Setting the TCP properly is essential for multi-pick and other path planning calculations.

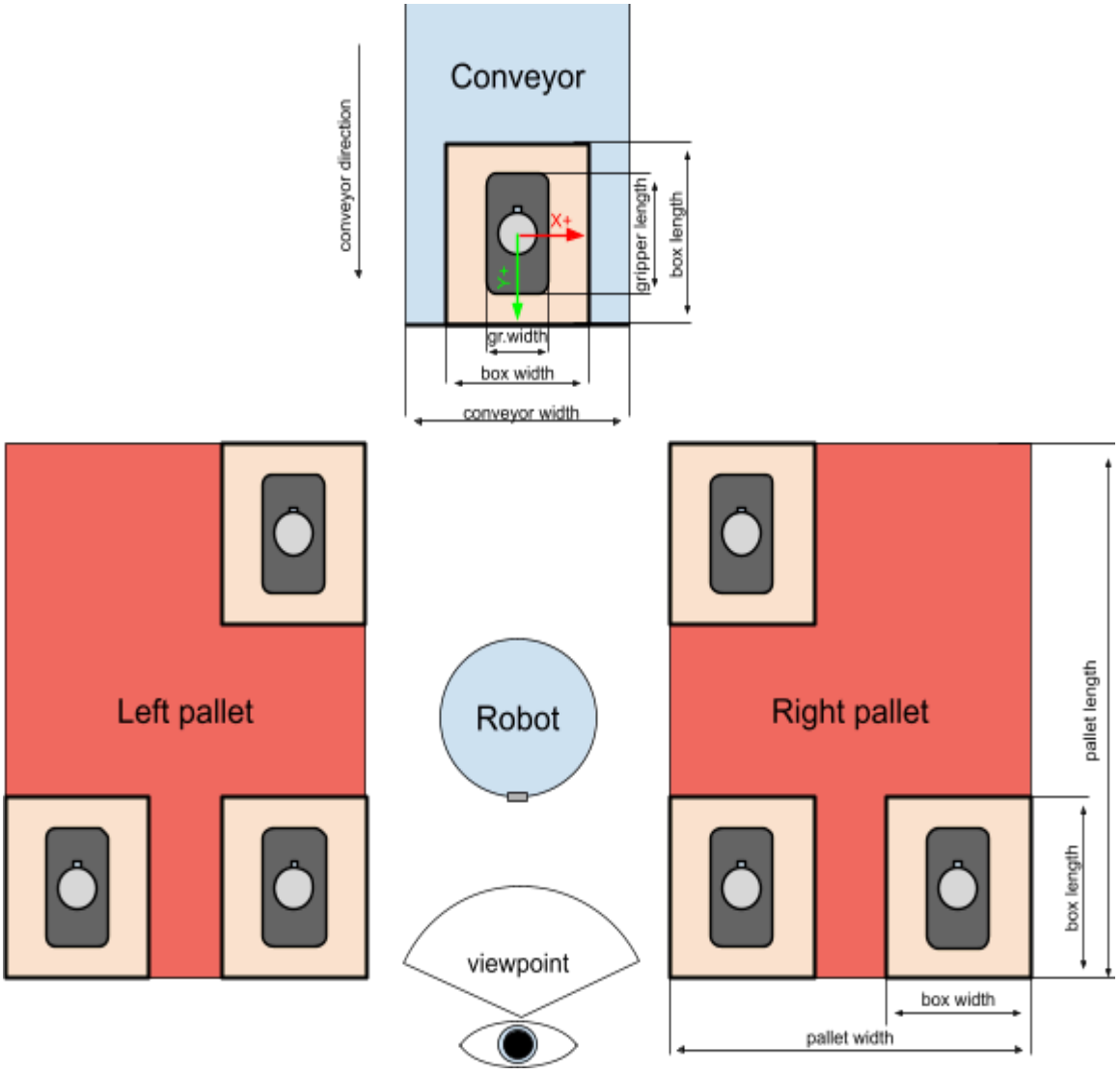
5.1.4 - Lifting column

Not supported with this version

5.2 - Layout

The palletizer cell consists of the robot, the pallets, and the pickup position. The conveyor and the pallet positions must be calibrated by moving the robot (with the gripper attached) into various calibration points.

Note: Always use the "width" and "length" values as shown below. Make sure that the width and length values are used consistently during the entire calibration.

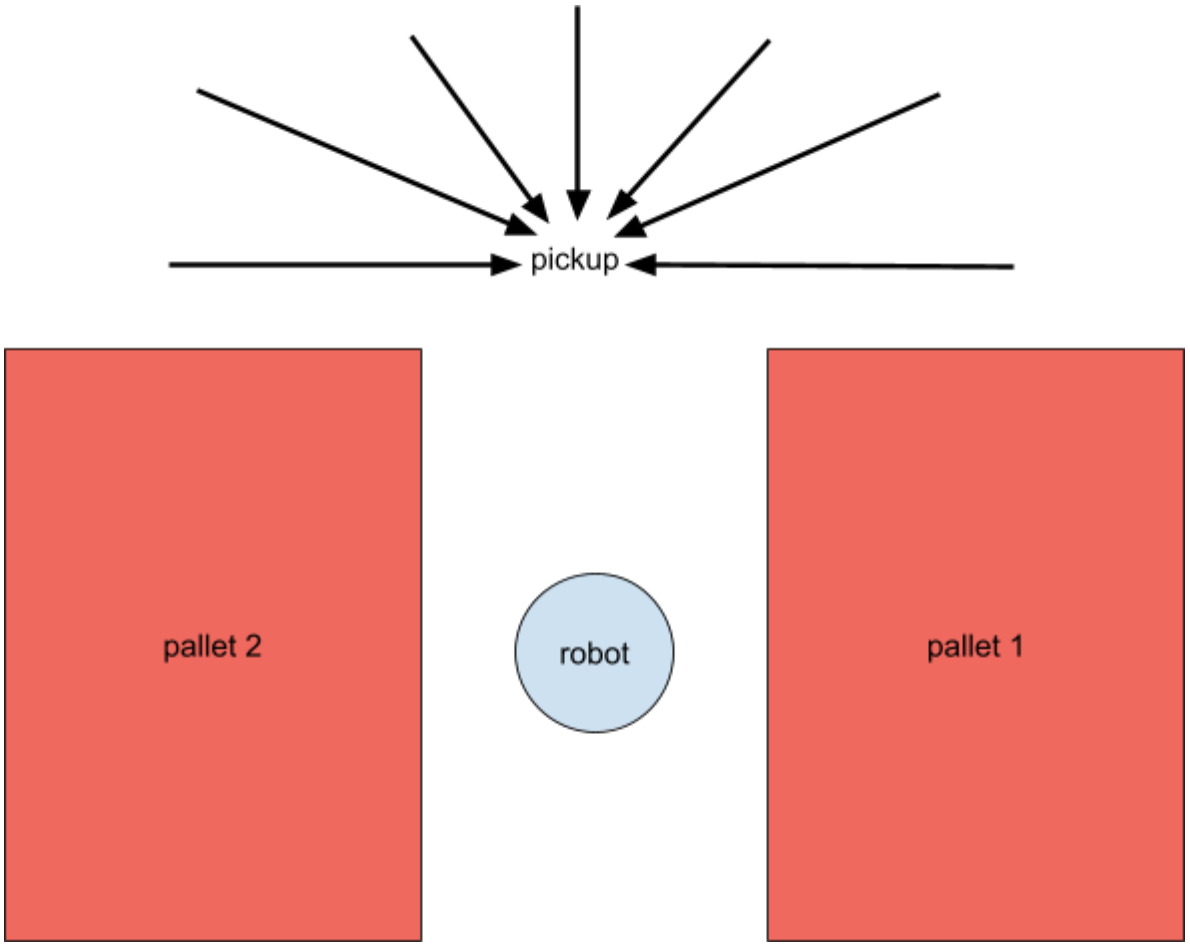


Calibration points for the pickup position and pallets.

5.2.1 - Supported pickup positions

Product arrival angle

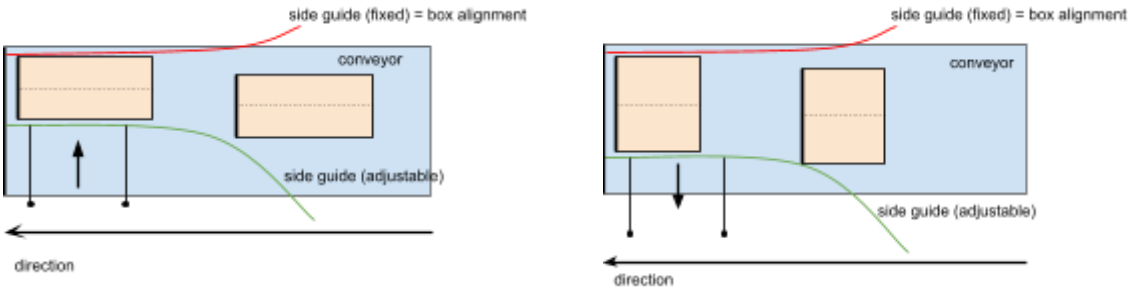
Pally supports several different pickup positions. The boxes may arrive at the pickup position from the side, from the front or from an angle. I.e., the boxes can arrive both sideways or straight towards the robot as illustrated by the arrows in. The end point for the pickup must be a fixed position.



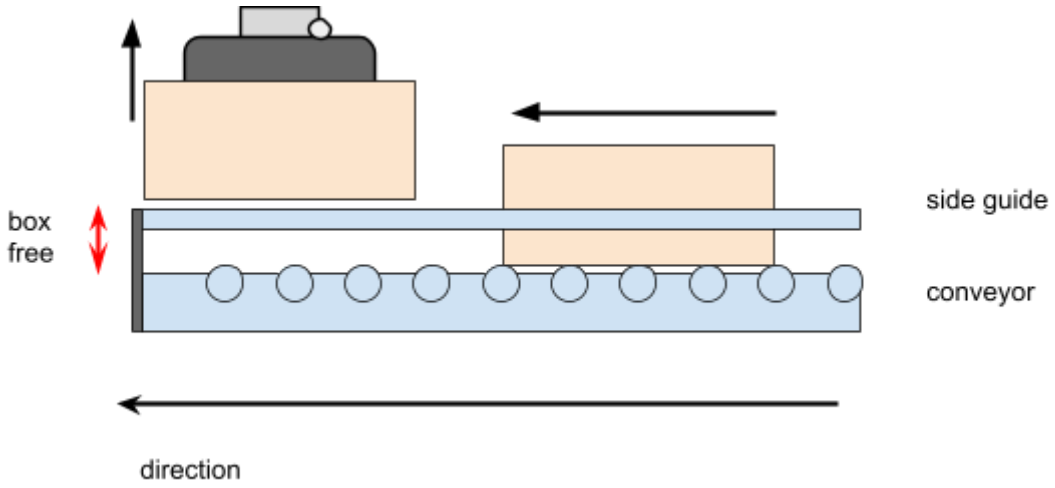
A product may arrive at the pickup position from a wide variety of angles, but should always be stationary on the same location when being picked up by the robot.

Multi-picking

Pally supports lifting and placing two or more products at the same time, as long as the total payload and the placement on the pallet allows it. When using multi-picking, the boxes must be aligned on the conveyor. Either guides or pushers are recommended for this purpose. Low guides are preferred, since these will give more room for the robot to optimize its paths for increased capacity and increased lifespan.



Use side guides or pushers to align products to the same reference point.



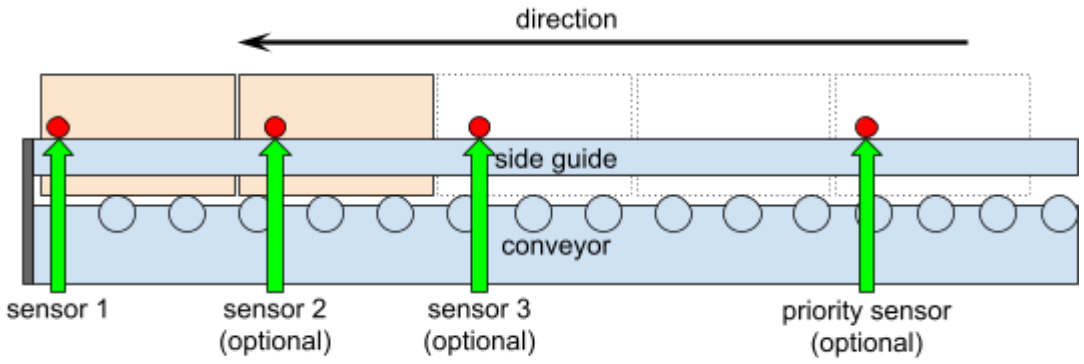
Shorter “box free” travel will improve speed and expand robot lifetime.

5.2.2 - Suggested sensor placements

Product presence sensors
Place the sensors as low as possible on the conveyor. Install one sensor at the very end of the conveyor to detect one box ready for pickup. Install the next sensor where the second box will be located.

Note: If palletizing many different products from the same line, an adjustable mounting of sensor 2 is recommended, so that it easily can be moved according to product size.

Priority sensors
It is possible to install a sensor close to where the line is full of products waiting to be palletized, to inform the robot that the buffer is getting close to full. This enables the robot to prioritize between lines. Multiple conveyors are not supported in Pally version 1.0.



Sensor placement.

Note: Maximum 8 product sensors per pickup position are supported.

5.3 - Dimensions

The recommended dimensions are specified in this chapter. The robot may palletize without issues using other dimensions as well, although significant deviations from the optimal dimensions may introduce unexpected failures.

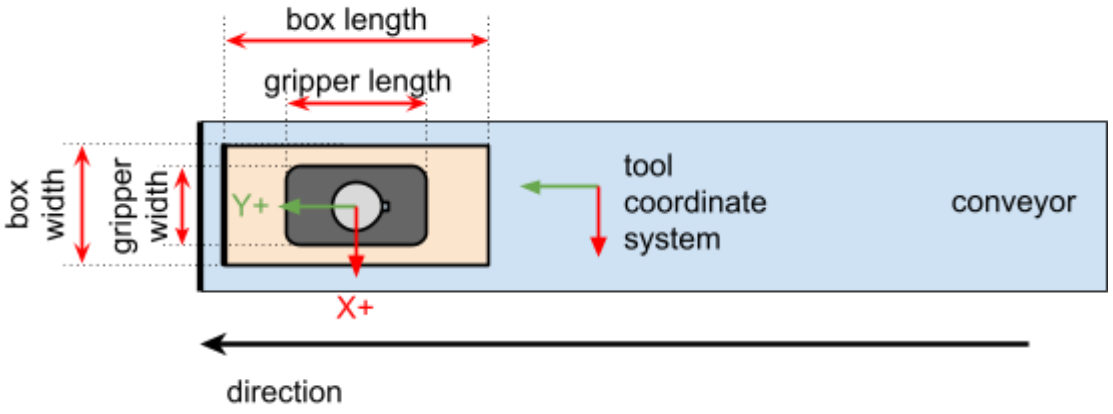
5.3.1 - Pallet

Pally is compatible with *any* pallets, as long as the selected robot has enough reach across the pallet. Please configure the pallet dimensions in the Pattern.

Pally can palletize on two separate pallets.

5.3.2 - Pickup from Conveyor

For maximum reach and flexibility it is recommended to place the robot so that the pickup position will be approximately at the following coordinates in respect to the bottom of the robot base.

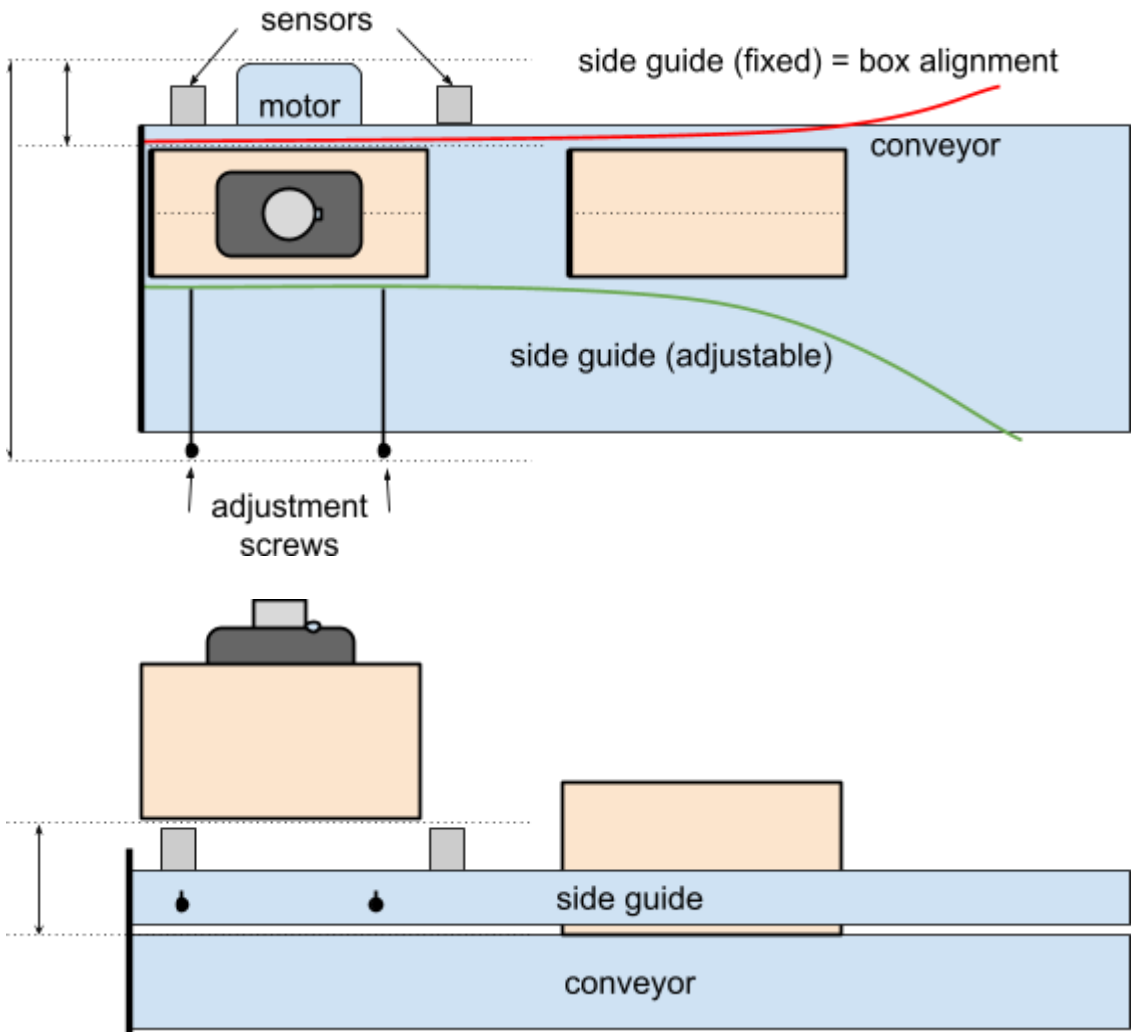


Box and gripper orientation relative to conveyor direction.

5.3.3 - Important parameters to be measured

The path planning algorithm requires the following values to be measured and entered:

- The total width of the conveyor, including any mounted objects. E.g. a motor.
- The position of the fixed guide. Left or right side of the conveyor, seen from the end.
- The fixed guide width, including any objects mounted on the conveyor. E.g. sensors.
- The minimum elevation needed to freely move the box above the conveyor. Measure this from the highest point, including any object which may be a hurdle.



Important dimensions to be measured.

5.4 - Gripper

Pally has built-in support for some *UniGripper*, *Schmalz* and *Piab* grippers, but custom grippers can be also specified. The weight of the gripper is part of the payload the robot must carry, i.e. a heavy gripper reduces the maximum weight of products to be eligible for palletizing. Keep this in mind when selecting a gripper.

Note: When installing a gripper, the tool center point (TCP) must be configured. Make sure the Y axis of the tool coordinate system is parallel to the conveyor transport direction. You ensure this by having the X2 connector pointing down the conveyor

---

5.5 - I/O Connections

All I/O ports are configurable in Pally, and may be used as needed. Typical configurations, such as a push button to confirm that a new, empty pallet is ready to use, and a light to indicate that the robot knows there is an empty pallet present, are supported.

Note: The program can be expanded to use more I/O in the callback features.

---

6 - Installing and Configuring hardware

Note: Install the URCap according to the software manual for your Doosan Robot before proceeding.

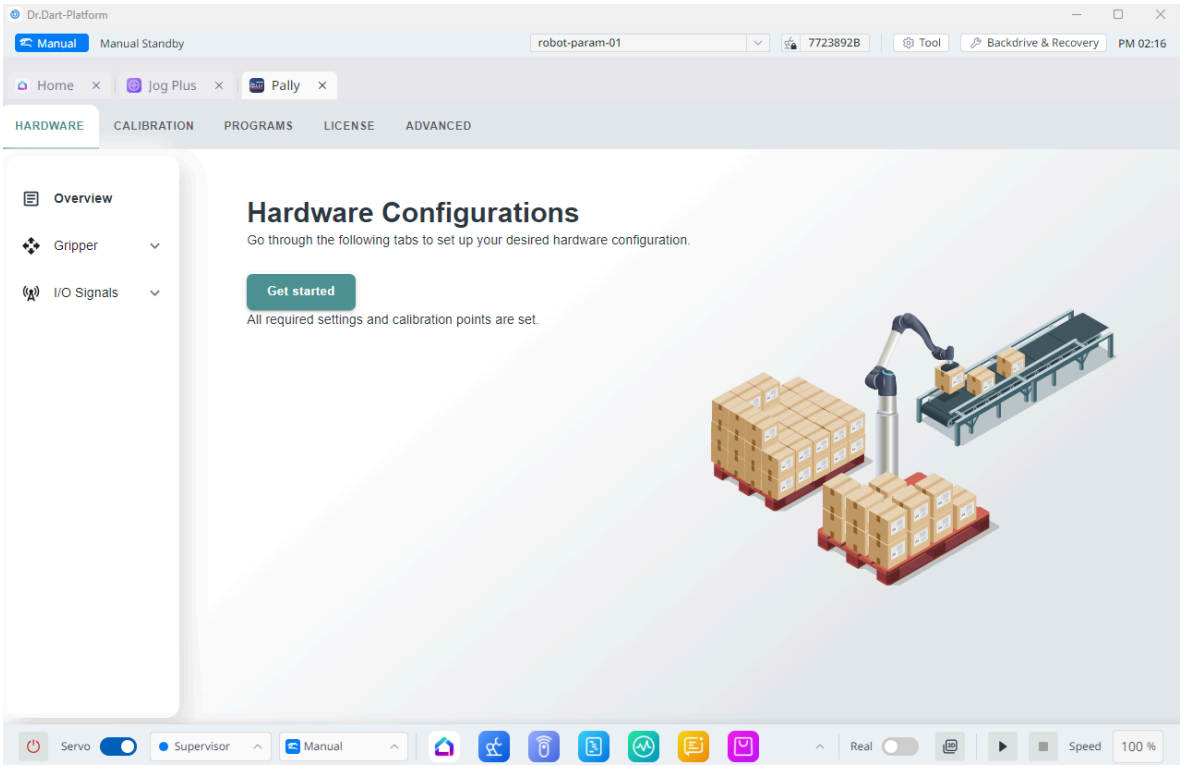
6.1 - Hardware

This chapter describes the various settings in the installation routine.

Note: This is a one time procedure, and it is important to follow the steps precisely.

6.1.1 - Overview

The Overview tab shows the completion status of the required installation settings. Verify that all items listed in this section are properly configured.



Overview of the installation status

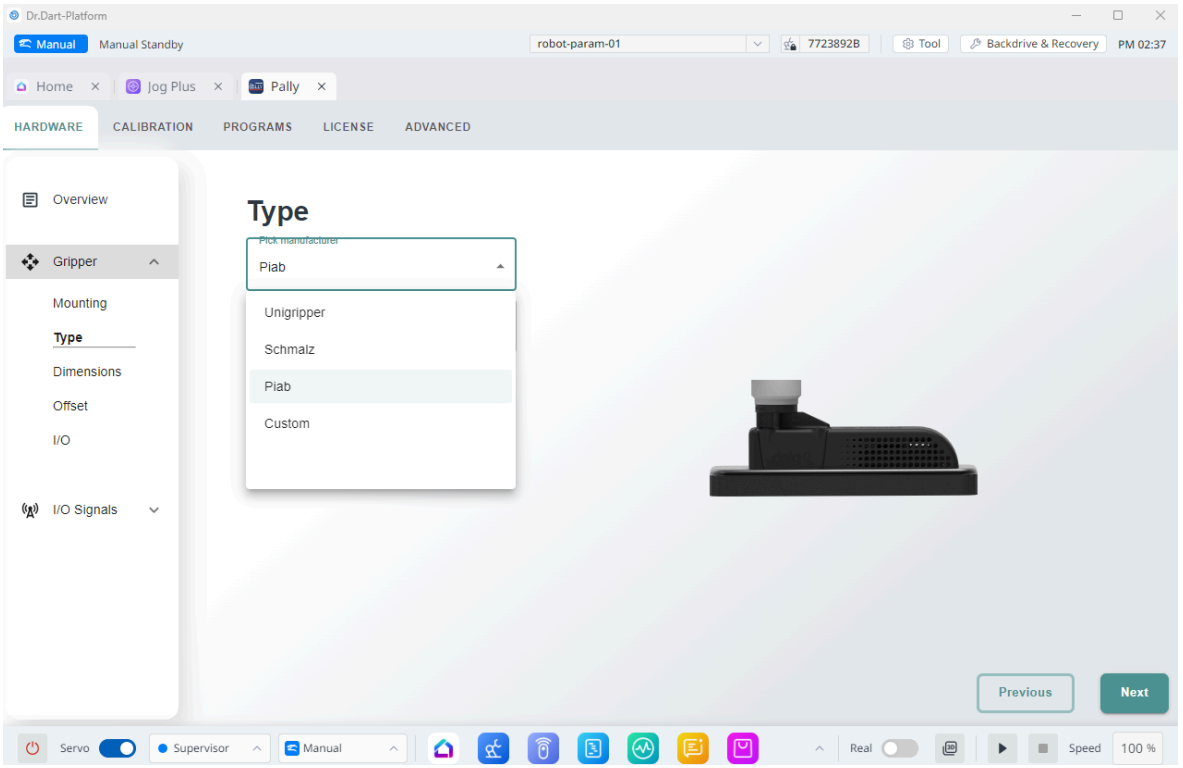
6.1.3 - Gripper

In this tab you can choose the gripper manufacturer and model, dimensions and weight, the IO channels connected to the gripper, and the mounting offset.
First choose the manufacturer of your gripper:

- Piab
- Schmalz
- UniGripper
- other

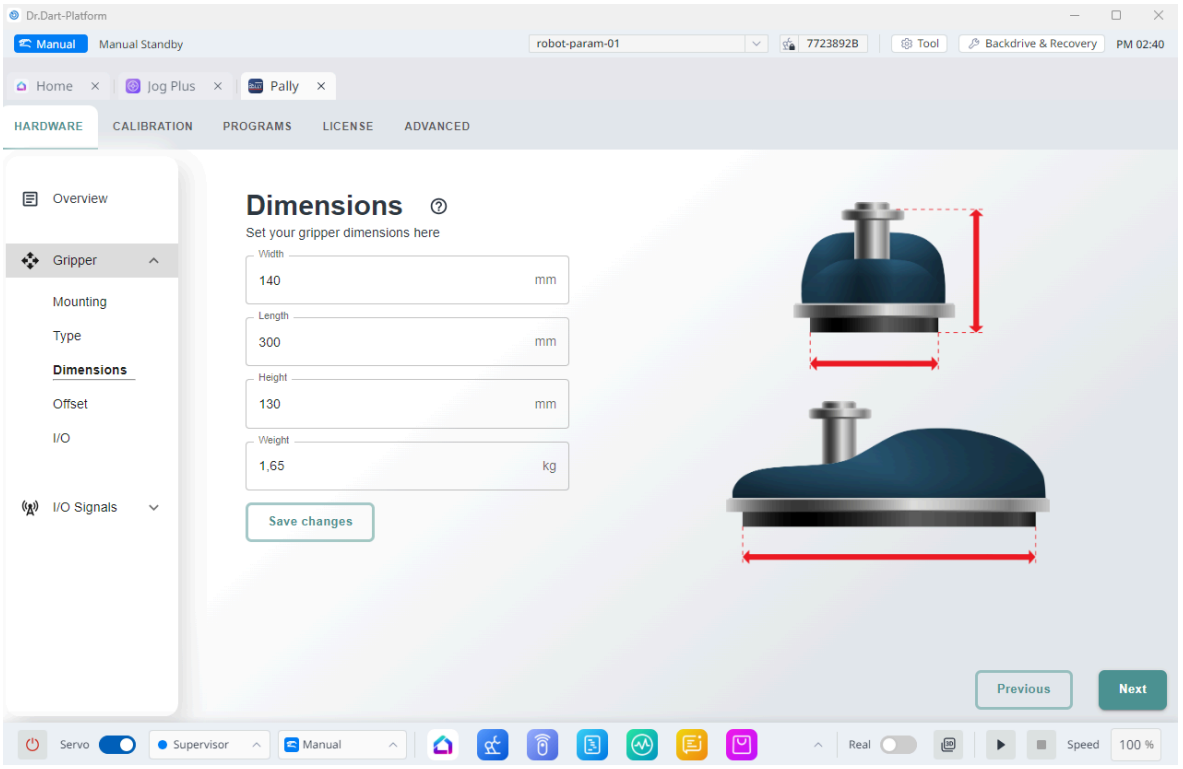
If you don't find the manufacturer in the drop-down list, select 'other'.

Choose the gripper model that is mounted on the robot. The dimensions, weight, TCP, and IO settings will be set automatically for supported gripper models.



Choose the gripper manufacturer and model

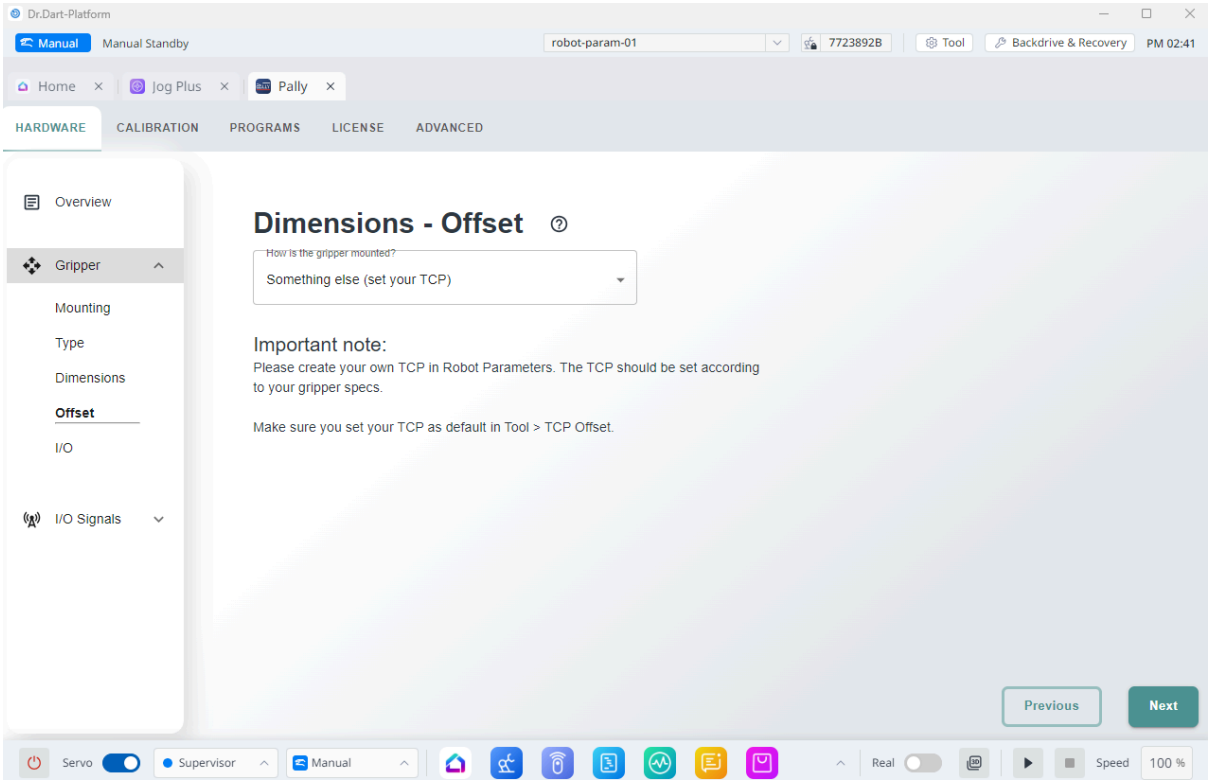
For specific gripper models, the dimensions, weight, TCP, and IO channels will be automatically preselected. Custom grippers require these values to be filled according to the gripper specifications. Please note: gripper length is the dimension which is parallel to the conveyor direction.



Gripper dimensions

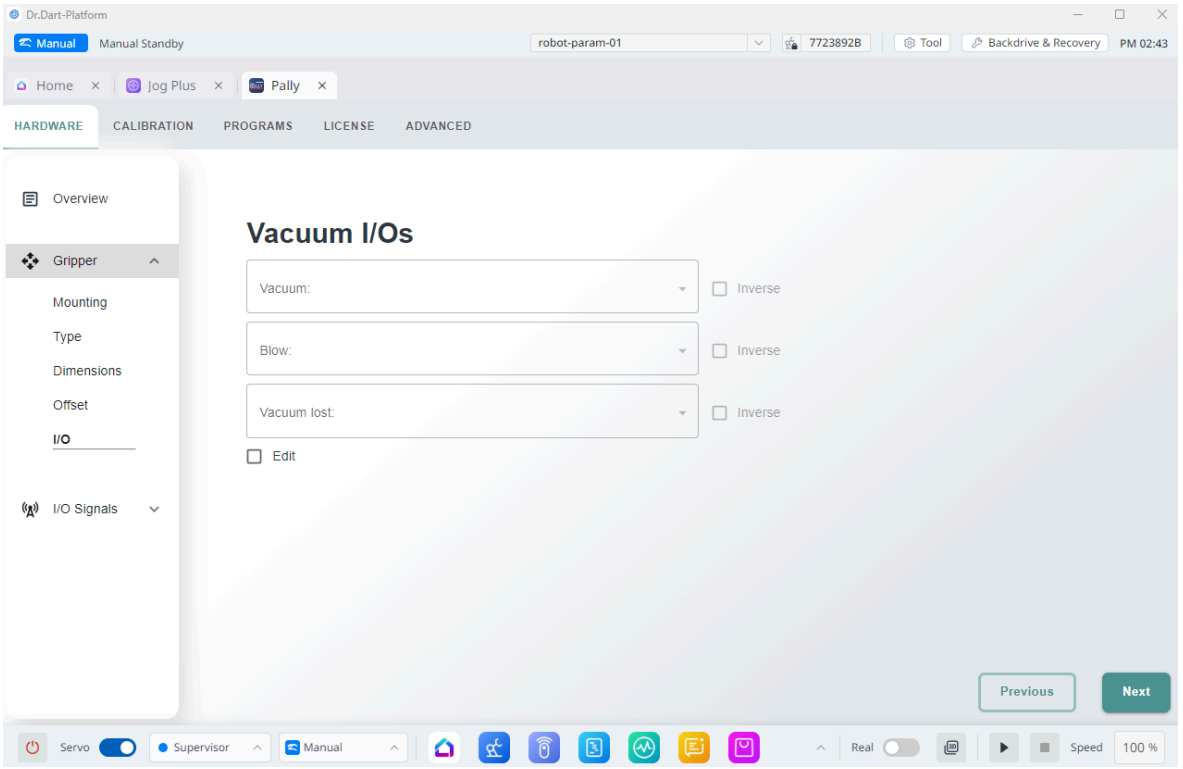
Some gripper models e.g. the Piab CPT supports more than one offset/mounting option. Select the correct offset that corresponds to the actual mounting. A TCP called "PallyTCP" will be automatically created; make sure to set it as default TCP in Tool.

If you use a tool changer or mount your gripper using a custom made offset plate, select the option "Something else (set your TCP)" and configure the robot TCP in Tool.



For grippers that support compressed air blow, select the "blow on" signal. By using this signal, compressed air will be used to blow dust off the box before picking, and it also makes the release process faster.

The 'Vacuum lost' signal can be used to detect false picks and lost boxes. The program evaluates the signal when picking the box and also before dropping the box on the pallet. To avoid empty holes in the pallet pattern, the last position is automatically repeated when loss of vacuum is detected. Please note that your gripper may require special configuration or extra hardware components to use this feature.



Select the gripper IO channels for vacuum, blow, and feedback.

6.1.3.1 - Supported gripper models

Piab CPT - Foam and Suction cups
Choose this option if a Piab CPT gripper has been mounted on the robot. The weight and dimensions of the gripper are selected automatically for the chosen model. Select the suction and blow signals according to the hardware connections.

Schmalz FXCB - Foam and Schmalz FXCB - Suction cups
Choose this option if a Schmalz FXCB Foam gripper or a Schmalz FXCB Suction cup gripper has been mounted on the robot. Once the type of gripper is set to one of these Schmalz grippers, the IO channels, weight, and dimensions are predefined. The grip and release functions are controlled by the Pally via the tool digital IO channels.

UniGripper Co/Light Regular, Mini and Maxi
Choose this option for standard mounted Unigripper Co/Light gripper only, as weight and dimensions are predefined. The grip and release functions are controlled by the Pally via the digital IO channels configured here.

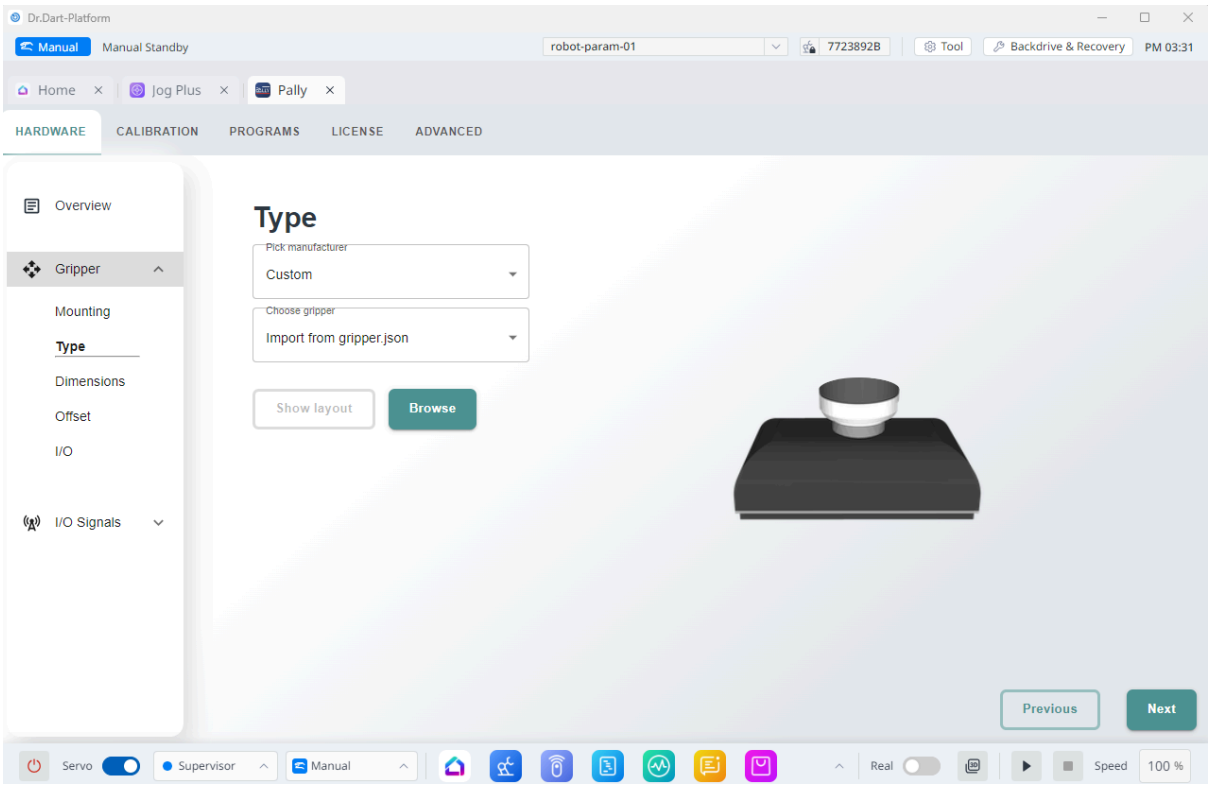
6.1.3.2 - Integrating other grippers

Custom Vacuum Gripper
Choose this option for other vacuum grippers that can be controlled by simple digital IO signals. Weight, dimensions and TCP of the gripper must be manually entered using this option. The grip and release functions are controlled by the Pally via the digital IO channels configured here.

Custom Gripper
Choose this option if the gripper has its own software to control grip and release functions. Weight and dimensions of the gripper must be manually entered using this option.

Note: For grippers in this category, it might be necessary to manually configure TCP (position, orientation, payload, center of gravity) according to the technical specifications of your gripper. Consult your gripper manufacturer for further details.

Import from gripper.json
You can upload your Gripper.json file after you have chosen the Import from gripper json in the dropdown menu.



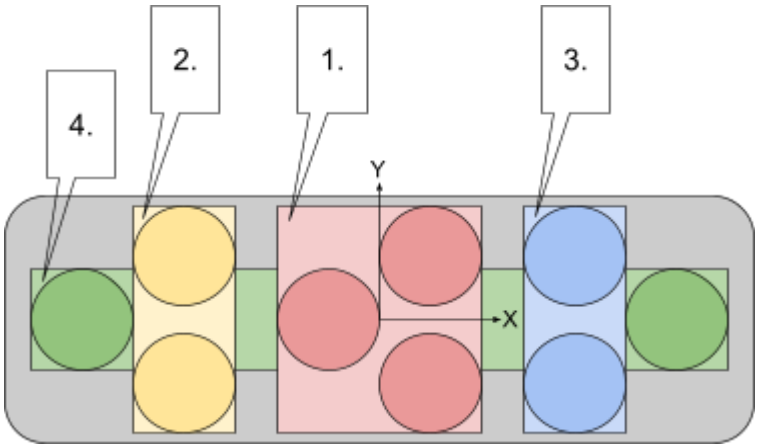
In order to have this functionality working, the position and the dimensions of the bounding rectangle should be defined for each vacuum zone as well as the digital output channels that

control the vacuum valves and optionally the compressed air used at release. The file has the following structure:

- name: name of the gripper
- description: a short description of the gripper
- picture: an optional picture in PNG format, encoded as a base64 string
- dimensions:
    - width, length, height: total dimensions of the entire gripper
    - weight: weight of the gripper including mounting bracket and screws
    - foamHeight: the height of the foam / suction cups that can be pressed down
- properties:
    - edgeAlignment: if true, the box edges will be aligned to the gripper zone edges, otherwise the gripper zone(s) will be centered above the box center
    - coveragePercent: defines how many percents of the vacuum zone (bounding rectangle) area must be covered in order to use a specific zone
- tcp: the tool center point (TCP) of the gripper as [x, y, z, rx, ry, rz]
- cog: the center of gravity (COG) of the empty gripper as [x, y, z]
- zones: definition of each individually controllable vacuum area
    - id: unique identifier (number) of the vacuum area, starting from 1
    - x, y: center point of the bounding rectangle relative to the gripper center
    - width, length: size of the bounding rectangle
    - grip: digital IO signal that controls the vacuum
        - type: standard, configurable, tool (robot IO type)
        - channel: the digital output channel (0-1 for tool 0-7 otherwise)
        - inverse: true if high level signal means no vacuum
    - release: digital signal that controls the compressed air (optional)
- configurations: list of valid combinations of vacuum areas that can be used together

(Lengths are specified in millimeters, weight in kilograms.)

The following diagram is an example of a 4-channel suction-cup gripper with its corresponding gripper.json file. The different colors indicate different groups of suction cups that can be controlled together.




```
{
  "name": "4-channel gripper",
  "description": "test gripper for multiple product sizes",
  "dimensions": {
    "width": 628,
    "length": 252,
    "height": 202,
    "weight": 1.9,
    "foamHeight": 15
  },
  "properties": {
    "coveragePercent": 51,
    "edgeAlignment": false
  },
  "tcp": [0, 0, 110, 0, 0, 90],
  "cog": [0, 0, 80],
  "zones": [
    {
      "id": 1,
      "x": 0,
      "y": 0,
      "width": 134,
      "length": 201,
      "grip": {
        "type": "standard",
        "channel": 0,
        "inverse": false
      },
      "release": {
        "type": "standard",
        "channel": 4
      }
```

```
    },
    {
      "id": 2,
      "x": -163,
      "y": 0,
      "width": 67,
      "length": 201,
      "grip": {
        "type": "standard",
        "channel": 1
      },
      "release": {
        "type": "standard",
        "channel": 5
      }
    },
    {
      "id": 3,
      "x": 163,
      "y": 0,
      "width": 67,
      "length": 201,
      "grip": {
        "type": "standard",
        "channel": 2
      },
      "release": {
        "type": "standard",
        "channel": 6
      }
    },
    {
      "id": 4,
      "x": 0,
      "y": 0,
      "width": 527,
      "length": 67,
      "grip": {
        "type": "standard",
        "channel": 3
      },
      "release": {
        "type": "standard",
        "channel": 7
      }
    }
  ],
  "configurations": [
```

```
      [1,2,3,4], [4], [1,2,3], [1,2], [2,3], [1], [2], [3]
    ]
}
```

Note: TCP settings are optional and may not be included in older gripper.json files. If your gripper is mounted with a position/rotation offset, make sure that the offset is properly set.

6.1.3.3 - Gripper visualization

When pressing the button "Show layout" under the gripper selection drop-down, Pally displays a simple diagram of the calculated gripper layout and mounting position relative to the robot tool flange. A schematic drawing of the robot tool flange and the tool connector is provided for a reference coordinate system.
For multi-zone grippers defined by a gripper.json file, each vacuum zone is shown in a different color.
The diagram shows the gripper and the robot tool flange from the underside.

6.1.4 - Input/Output

The assignment of input/output channels to each hardware unit can be configured here.

Inverted signals are supported; where an inverted signal means that the signal goes from high to low when a specific event occurs (e.g product present). On the contrary, the default signal (i.e not inverted signal) means the signal goes from low to high when a specific event occurs. To use an inverted signal, simply click on the ‘Inverse’ checkbox next to the dropdown for the signal value.

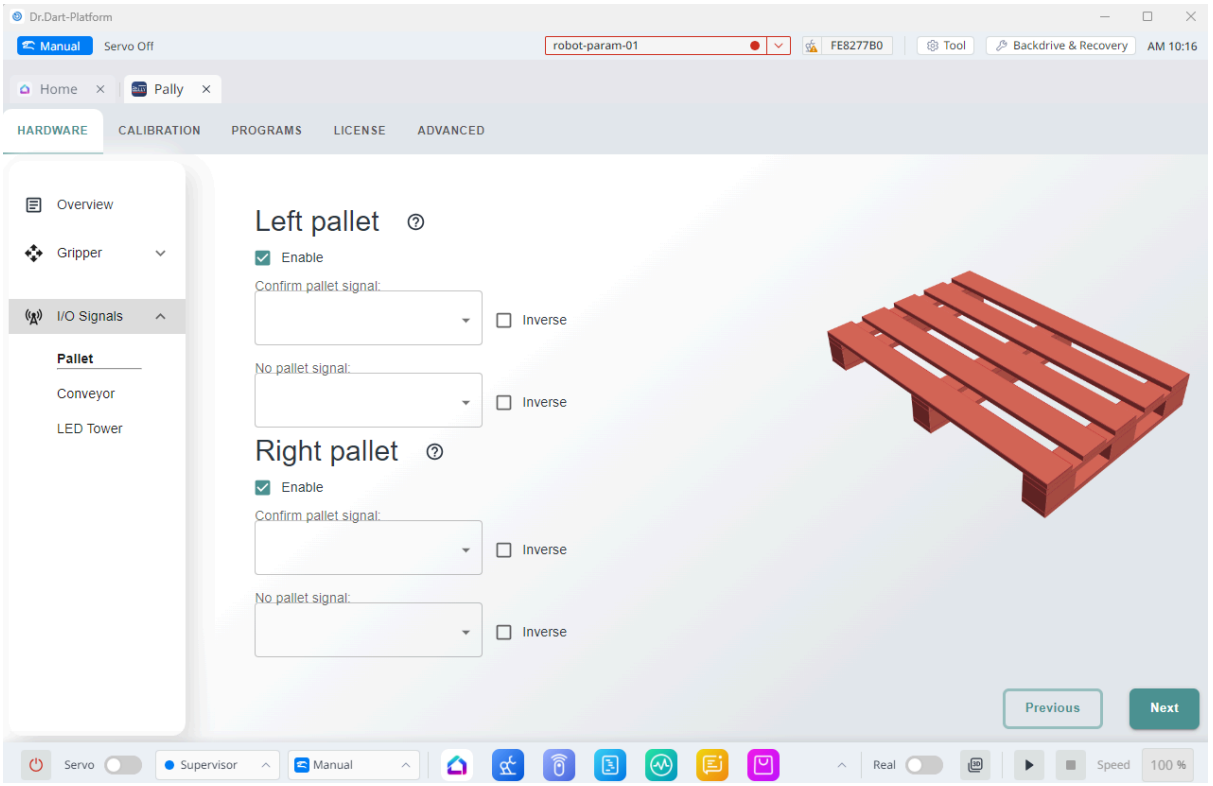
For input signals it is also possible to select the "Always LOW" and "Always HIGH" options. In this case, no physical input channel is being evaluated, and the result is constantly low or constantly high, respectively. For output signals it is possible to select "None". In this case the signal won't appear on any physical output channel.

It is very important to select different channels for each unit. For instance, do not use digital_input[0] for pallet presence AND product presence sensor, as this will result in unexpected behavior.

6.1.4.1 Pallet

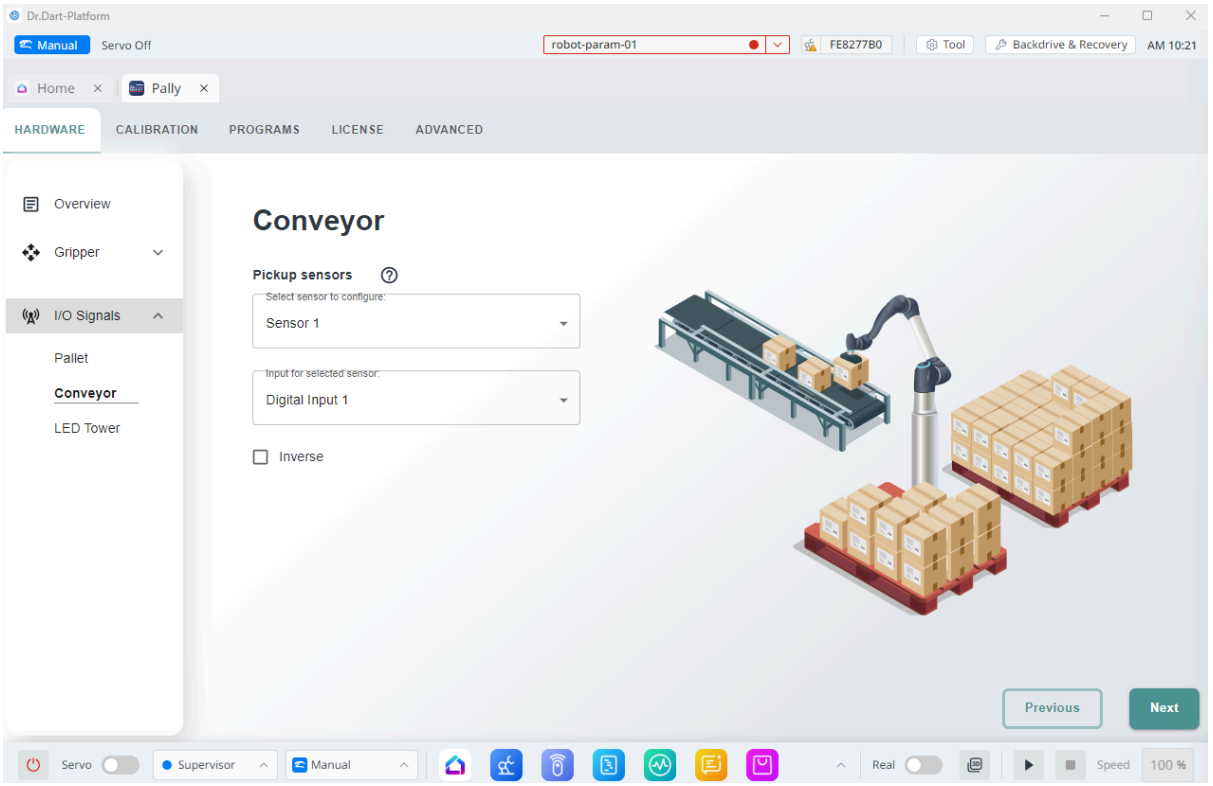
Confirm pallet signal
These input signals shall be connected to physical push buttons where the operator can confirm that the empty pallet is present and ready for palletizing. After a pallet has been finished, the program will not start palletizing the same pallet position again until the respective signal is received for at least 0.2 seconds.
Empty pallets can be also confirmed by using the Pally Operator Panel In this case the physical signals can be omitted.

No pallet signal
These output signals can be connected to physical light sources, for example, to illuminate the pallet present buttons. The light will flash with 1Hz while the program is waiting for the pallet present button to be pushed.
When using the Pally Operator Panel the pallet state is displayed on the teach pendant. In this case the physical signals can be omitted.



6.1.4.2 Conveyor

Choose what sensor you use and set corresponding I/O. Each product position needs a unique sensor. If you intend to be able to triple pick you need to have 3 sensors installed and set the I/O's.

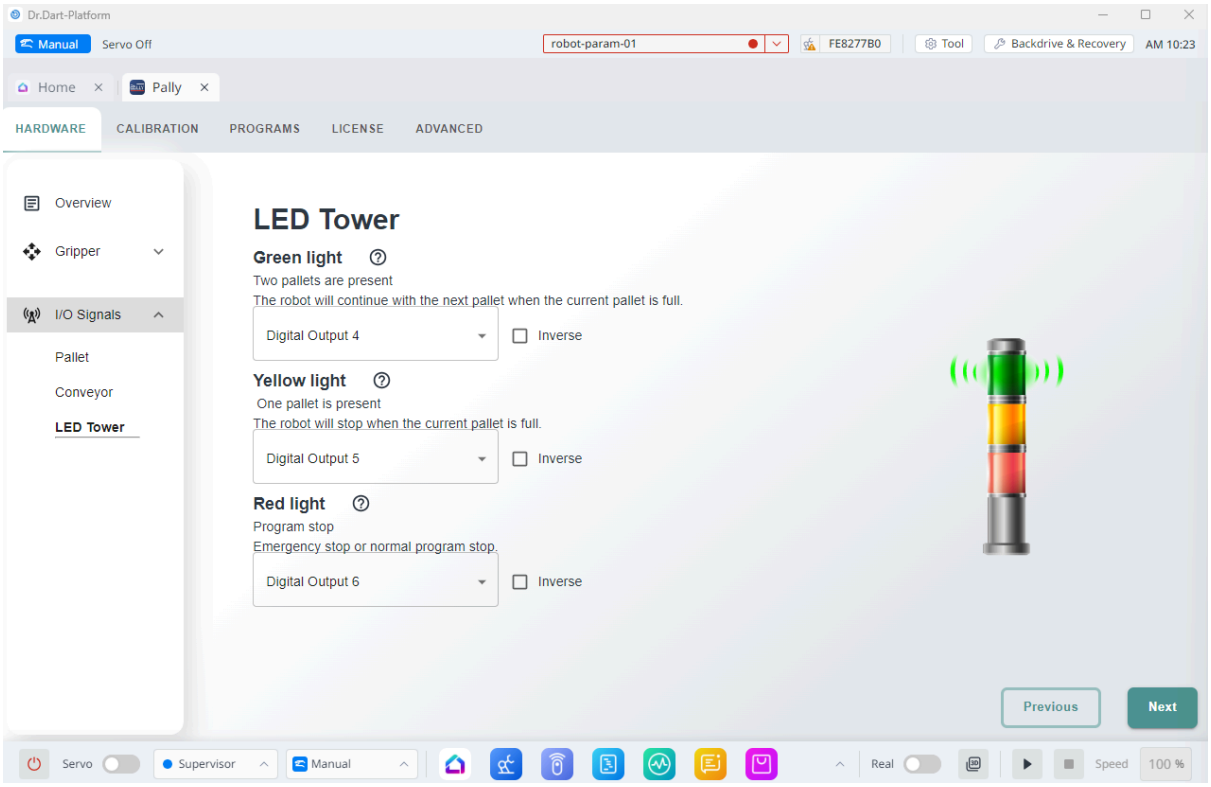


6.1.4.3 LED-tower

These output signals can be connected to light sources that indicate the status of the palletizer robot as follows.

Red	Program stop (emergency stop, normal program stop)
Yellow	Only one pallet is present, the robot will stop when the current pallet is full
Green	Both pallets are present, the robot will continue with the next when current pallet is full

Do not use the Green output in projects with only one pallet position, as it will never be ON. In this case use the Yellow output to indicate normal operation.



Note: It is recommended to give meaningful names to the input/output channels to represent their physical meaning in the I/O Setup menu in the installation tab.

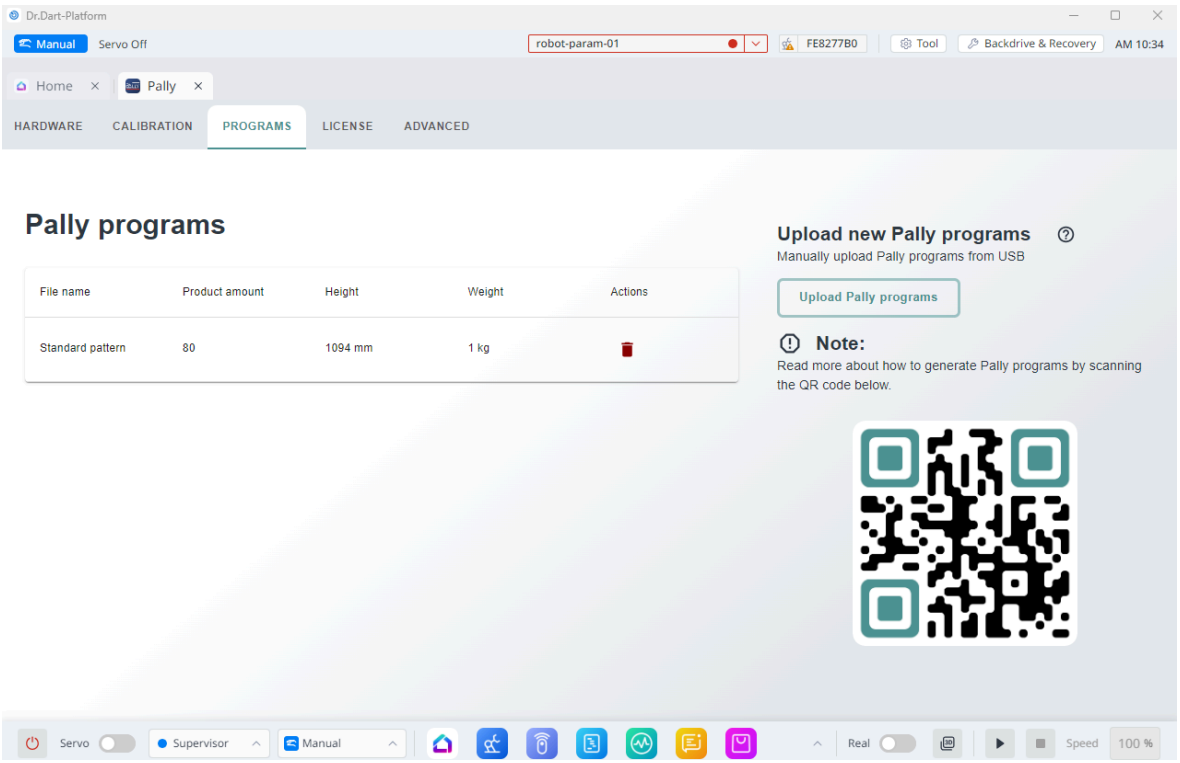
6.2 - Programs

Pally uses programs generated in [MyRobot.cloud | Palletizing Robot Management System | Rocketfarm] to control the path while palletizing patterns.
You can read more about how to generate programs in this article - [Pally knowledge base - Generate Pally Programs]

After generating the Pally Program you use a USB stick. Insert the USB stick and press the button “Upload Pally programs” on the Programs tab.

Note: When you import a program with the same name multiple times, the old data is overwritten in the database

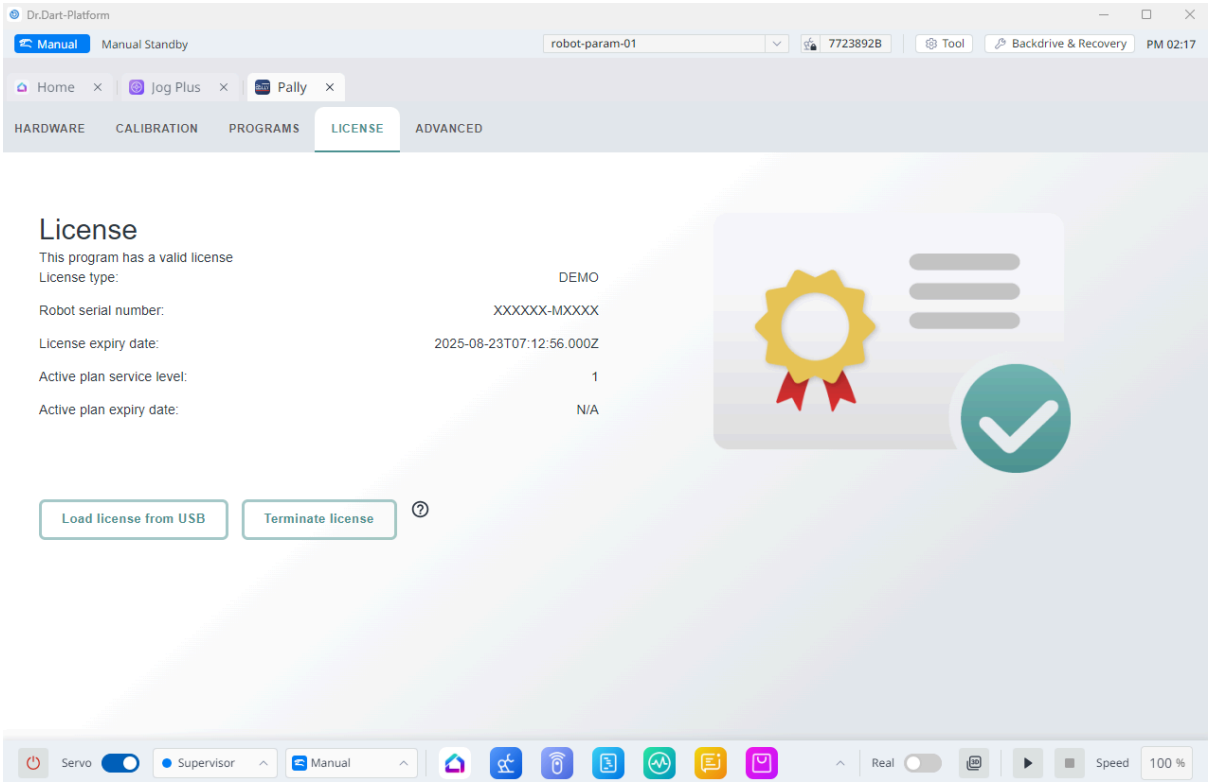
---



Pattern management.

6.3 - License

Using Pally requires a valid license. Licenses can be installed and removed on the License page.



6.3.1 - Installing a license

A valid license must be obtained before Pally can be used. This can be done as follows:

- Send a license request to license@rocketfarm.no with the serial number of the robot
- Rocketfarm provides you with a license file
- Navigate to the tab labeled 'License'
- Download the license file to the root folder of a USB stick.
- Insert the USB stick in the USB connector on the Teach pendant and click the 'Load *license from USB*'-button..

When a license cannot be installed, the program will show a detailed error message, like:

- The license is already expired and cannot be installed
- License has been terminated and cannot be reused
- The same license is already installed on the robot
- License is not suitable for the actual hardware (the request was generated on another robot / controller)
- Robot system clock is wrong (license issue date is in the future)
- Multiple license files present on the same USB

The license file are named using the following naming conventions, where *[serialNumber]* is the serial number of the robot:

- License file:
  no.rocketfarm.pal.[serialNumber].license

6.3.2 - Terminating a license

If the license is no longer needed, insert a USB disk and press the "Terminate license" button. The program will deactivate the license and generate an evidence file on the USB.

The URCap can be safely uninstalled after the evidence file has been generated and sent to Rocketfarm.

6.4 - Advanced

For better visibility, the Advanced screen has been divided into the following tabs:

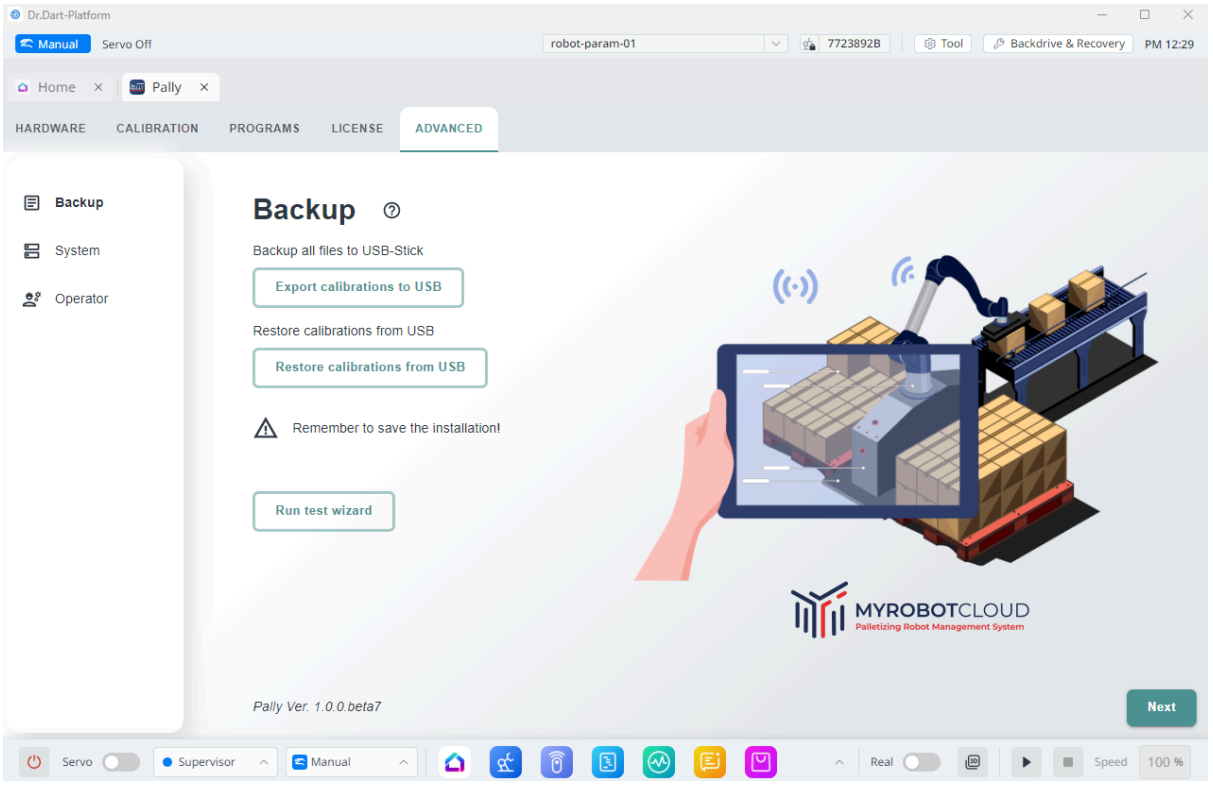
- Backup
- System
- Operator

6.4.1 - Backup

Export calibrations to USB will download a backup of the Pally program

Restore calibrations from USB can be used to restore an existing calibration

Run test Wizard can be used to test the current calibration and is highly recommended before exporting it and generating Pally programs with it.



6.4.2 - System

Here you can set the Log level. Recommended log level is Warning

Critical - Will log very serious errors that may prevent the program from continuing to run.

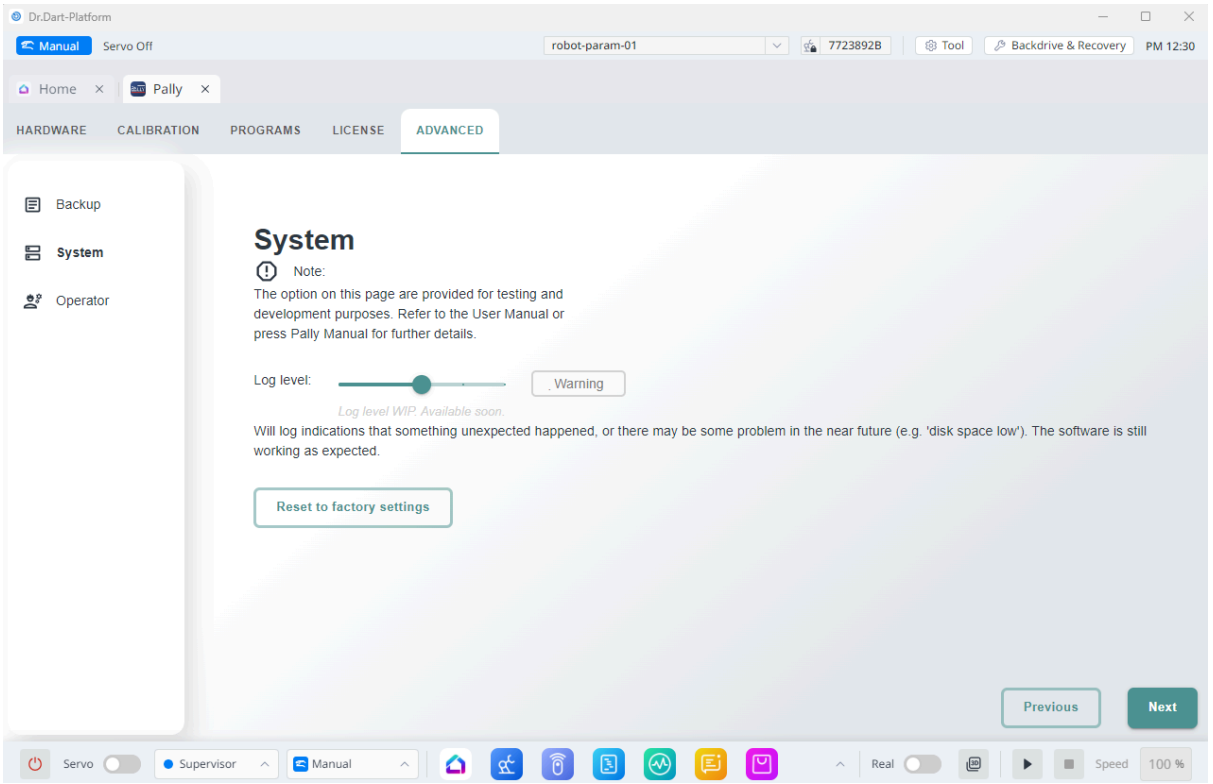
Error - Will log more serious problems, when the software is not able to perform some function.

Warning - Will log indications that something unexpected happened, or there may be some problem in the near future (e.g. 'disk space low'). The software is still working as expected.

Info - Will log confirmation that things are working as expected.

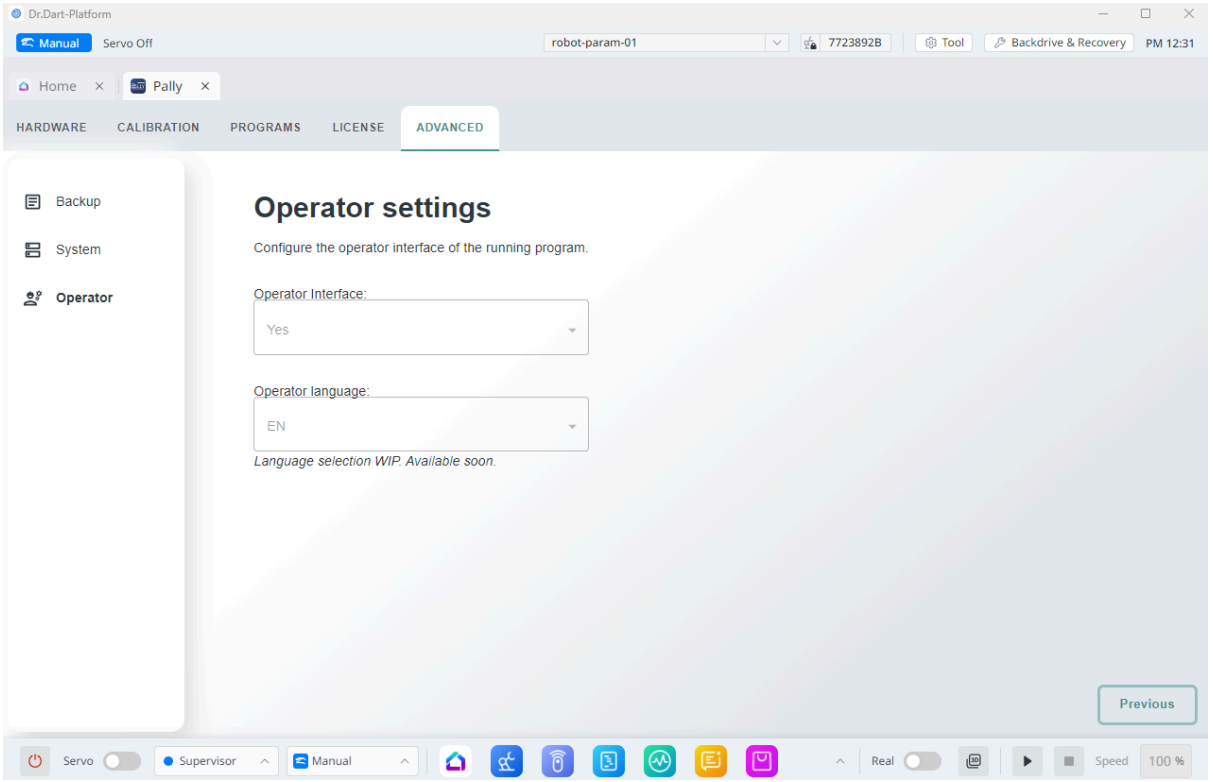
Debug - Will log detailed information, typically of interest only when diagnosing problems.

Reset to factory settings - all values in Pally will be set to empty, both hardware and calibration



6.4.3 - Operator

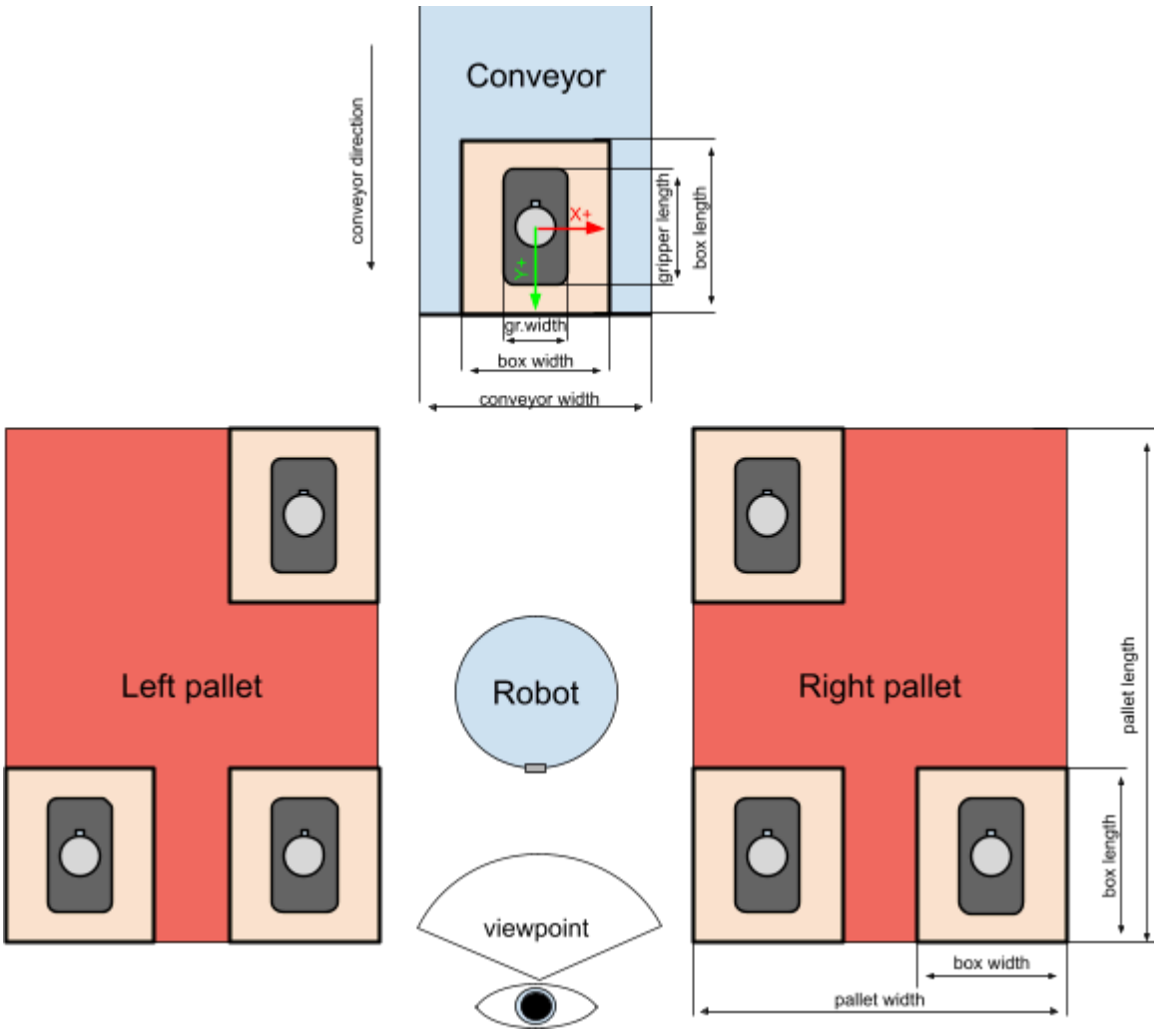
Turn on and off the Operator Panel as well as choose the language



7 - Calibration

Note: There are several points to be calibrated here. You can use the 'freedrive' mode on the robot, or the built-in '3d' view or the 'jog plus' screen. The 'teach...' buttons in Pally will not open any dialog but silently update the corresponding settings.

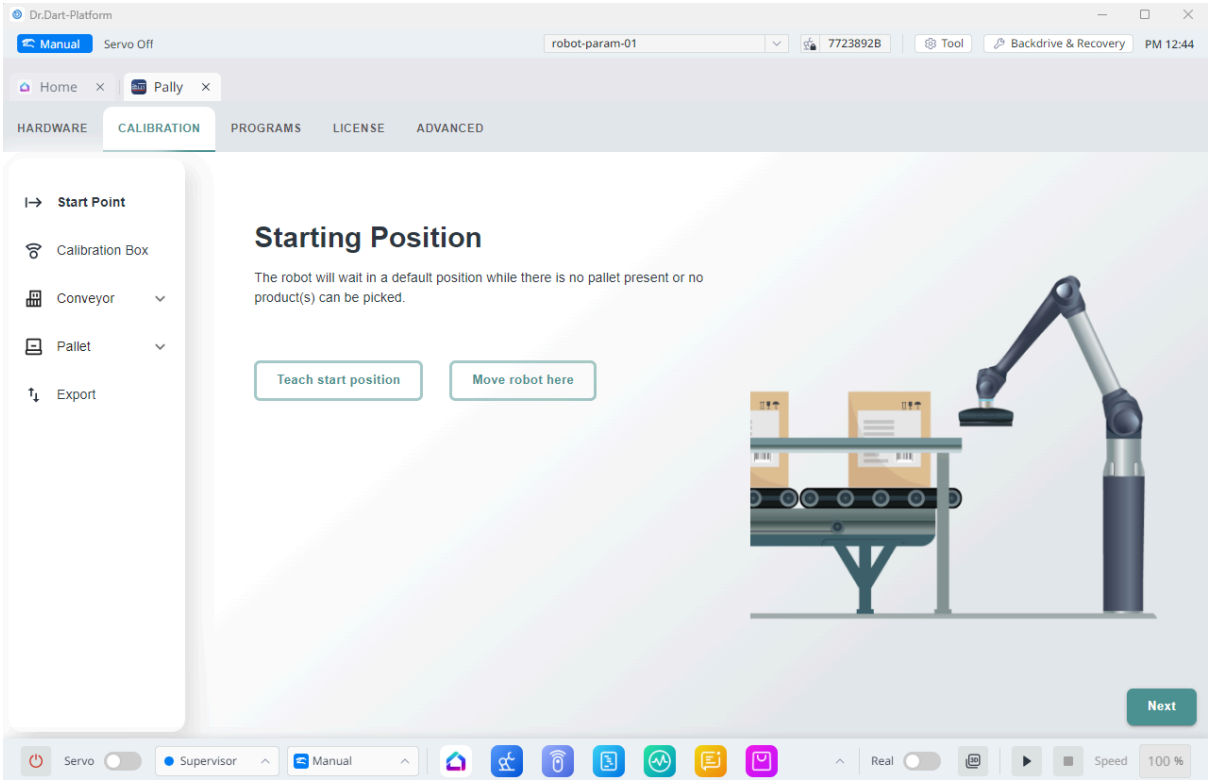
It is important to use width and length values consistently during the calibration. Make sure you use the width and length values as illustrated in the following diagram, and do not swap width and length in any position.



7.1 - Start Point

Starting position is essential for all waypoint calculations. Move the robot to a position where it is above the conveyor and there is a comfortable distance (approx. 30-40cm) between the gripper and the boxes.

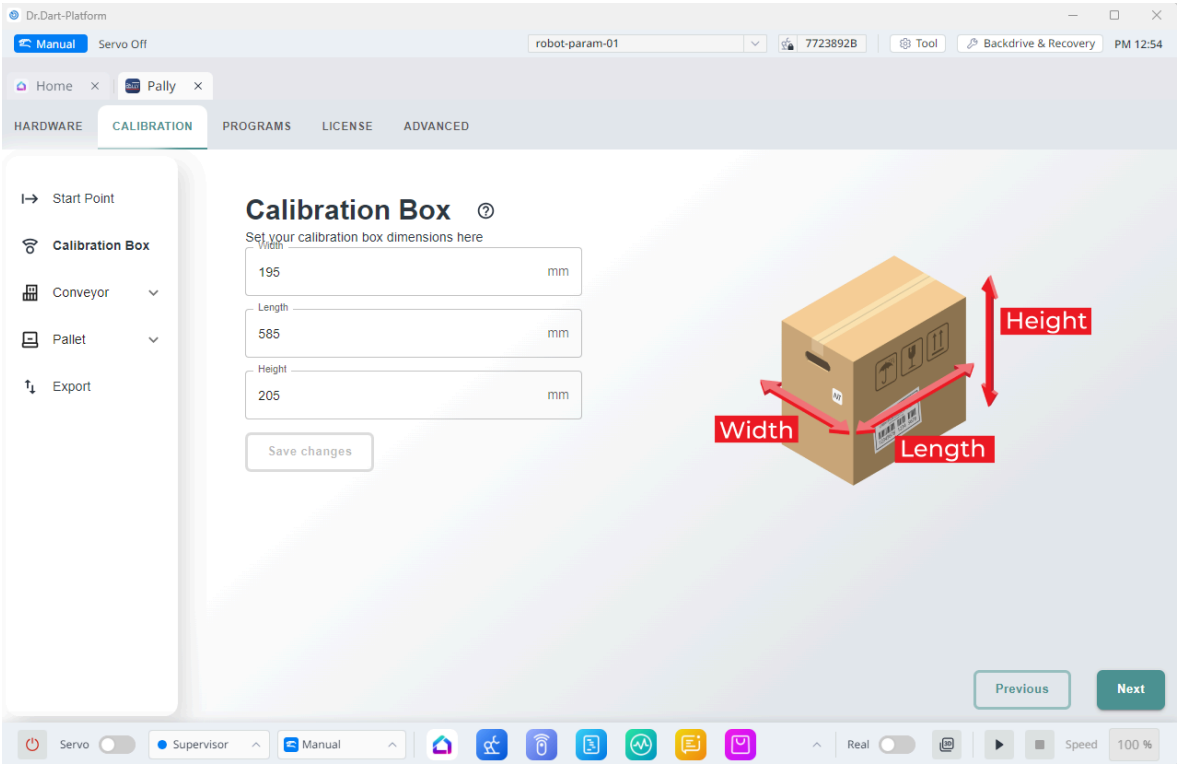
When you press the “Teach start position” button, it will store the actual robot position and joint angles without further notice.



7.2 Calibration Box

Enter the calibration box dimensions in mm and press Save changes.

Note: The calibration box does not have to be the same box as you intend to palletize later. It is more important that the box is stable and accurate. We recommend that the box is slightly bigger than your gripper. This makes it easier to place the gripper perfectly centered on the box, with equal distance to all sides.

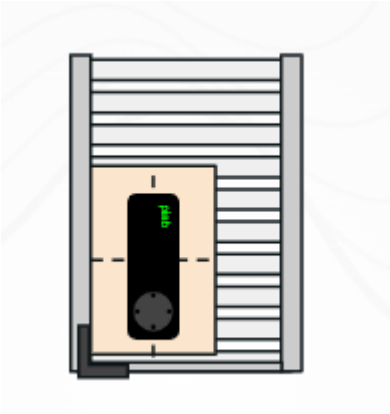


Calibration box

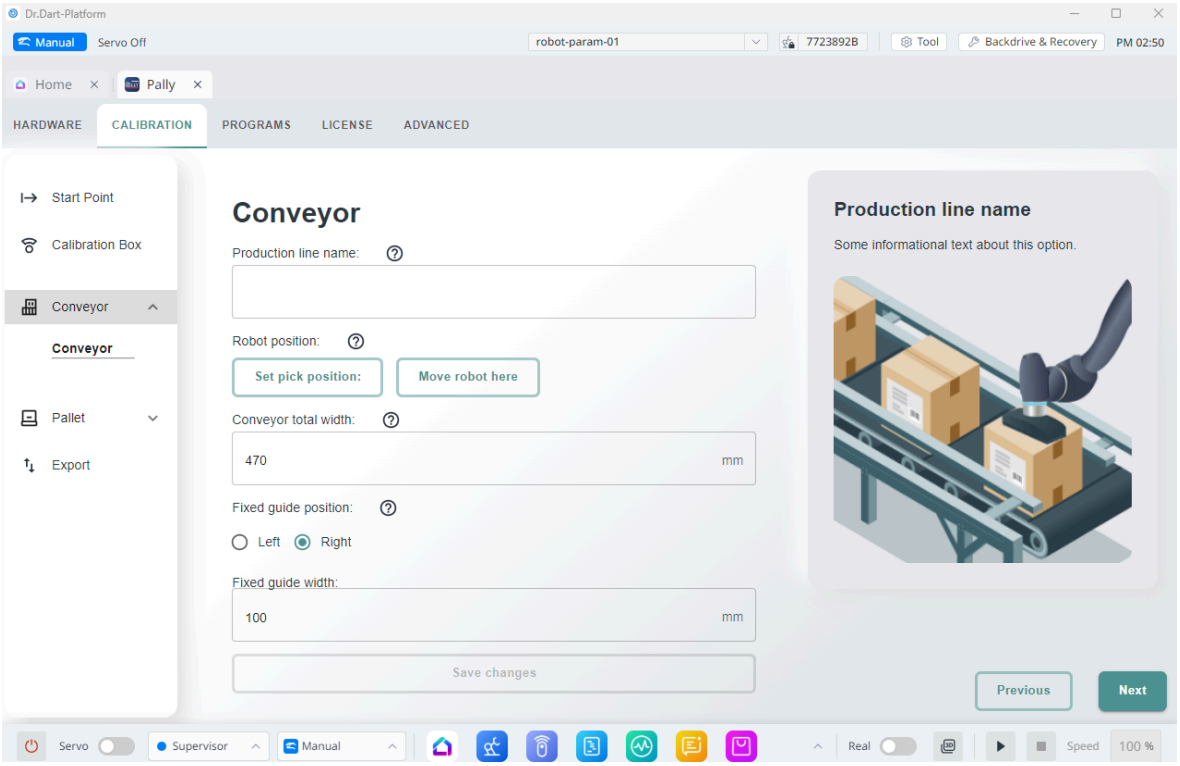
Note: The measurements must be accurate.

7.3 Conveyor

Align the gripper in the center of the calibration box, so that the orientation of the tool is perfectly perpendicular to the pickup position.

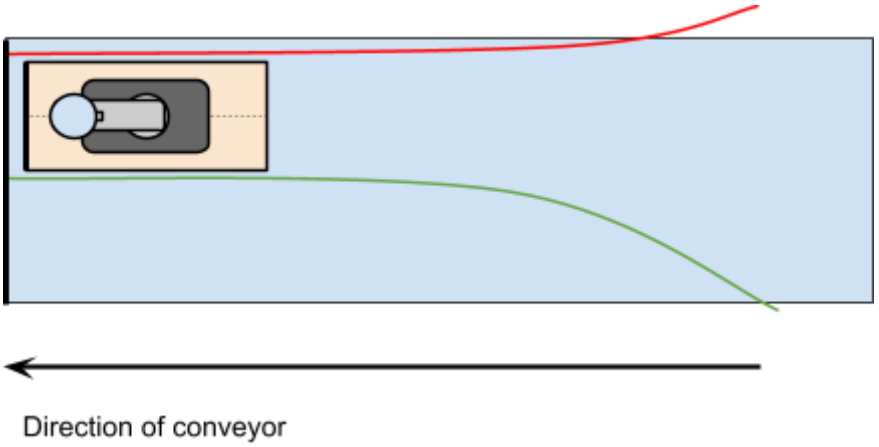


It is recommended that the calibration box is slightly larger than the gripper. This makes it easier to place the gripper in the center of the box.



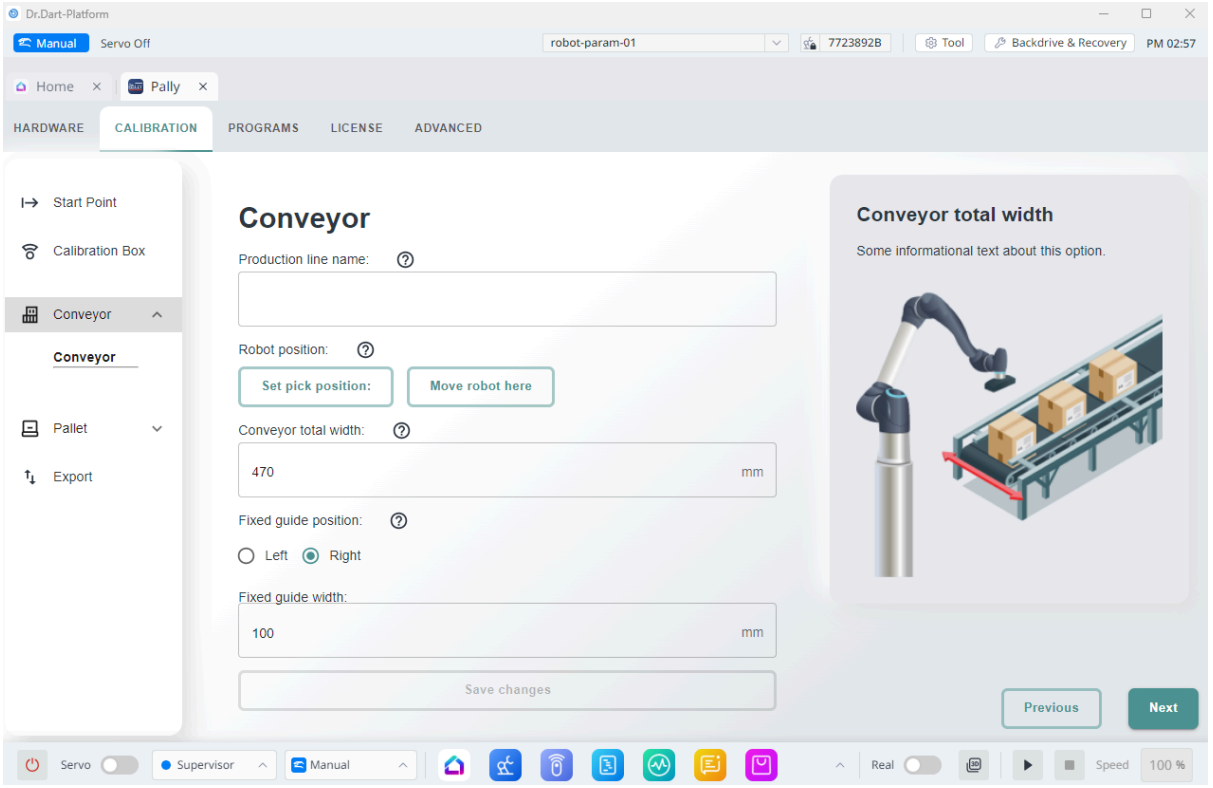
Calibrating the pickup position with a calibration box.

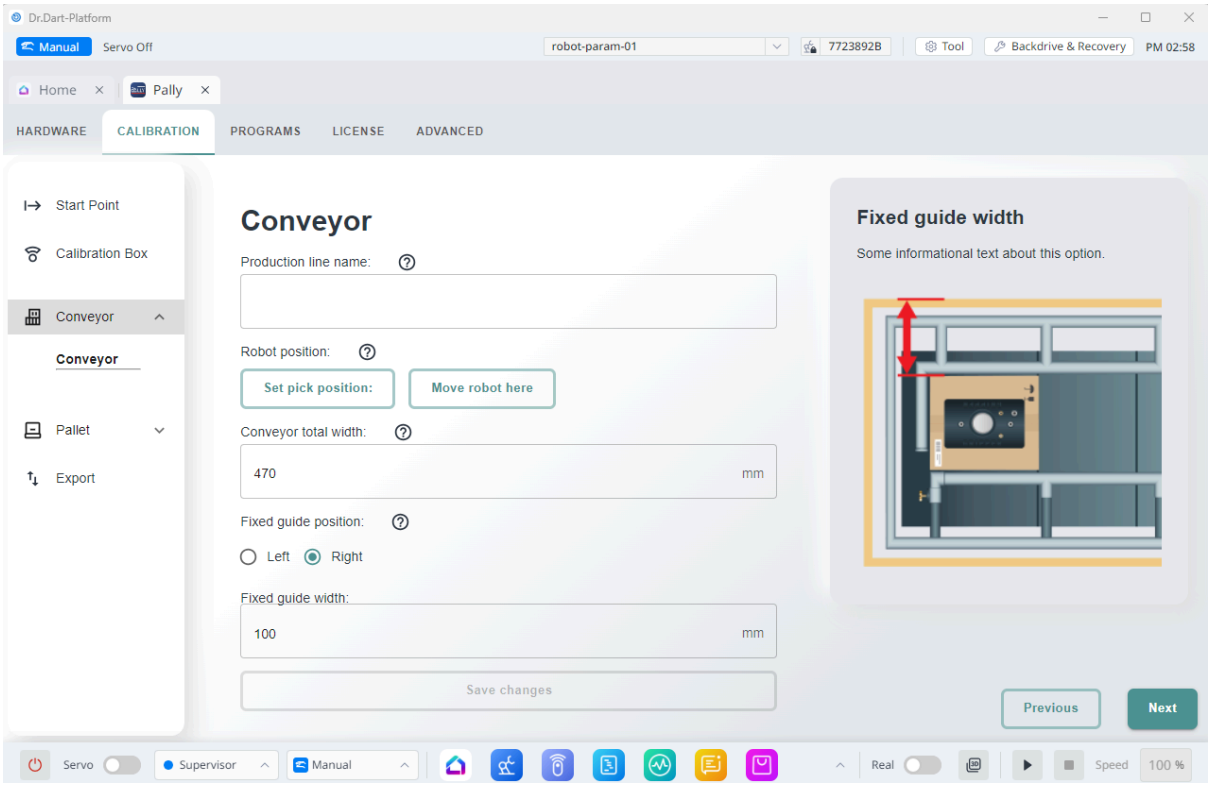
Some grippers may have offset by design, or can be mounted with an offset.
The correct calibration of the pickup position with an offset-mounted gripper is shown below.



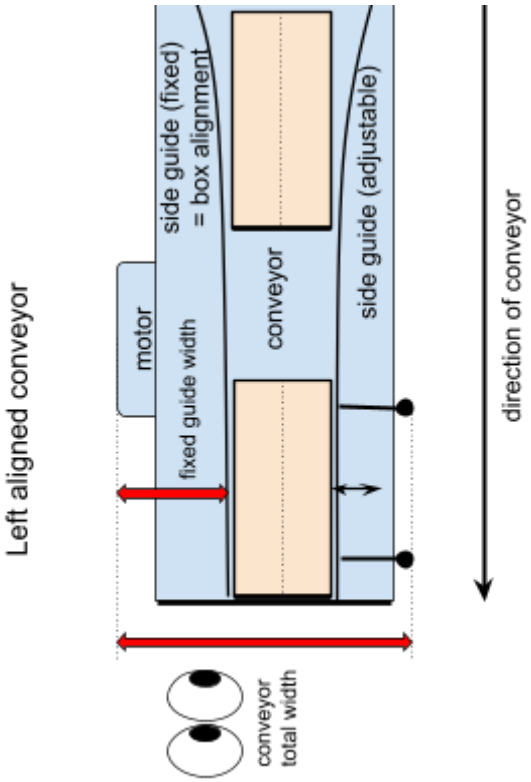
Calibration of the pickup position with an offset-mounted gripper

Enter the conveyor dimensions and select the fixed guide position.





Note: The fixed guide orientation is looking from the end of the conveyor and up the line.



The fixed guide orientation is looking from the end of the conveyor and up the line. In this example the fixed guide position is *left*.

7.4 - Pallet

The pallet center point and the floor inclination angle are calibrated by moving the robot above three corners of an empty pallet.

Note: A lifting column has to be at its zero position during calibration.

Note: It is necessary to calibrate the pallet positions again when the gripper dimensions on the *Installation* page are updated.

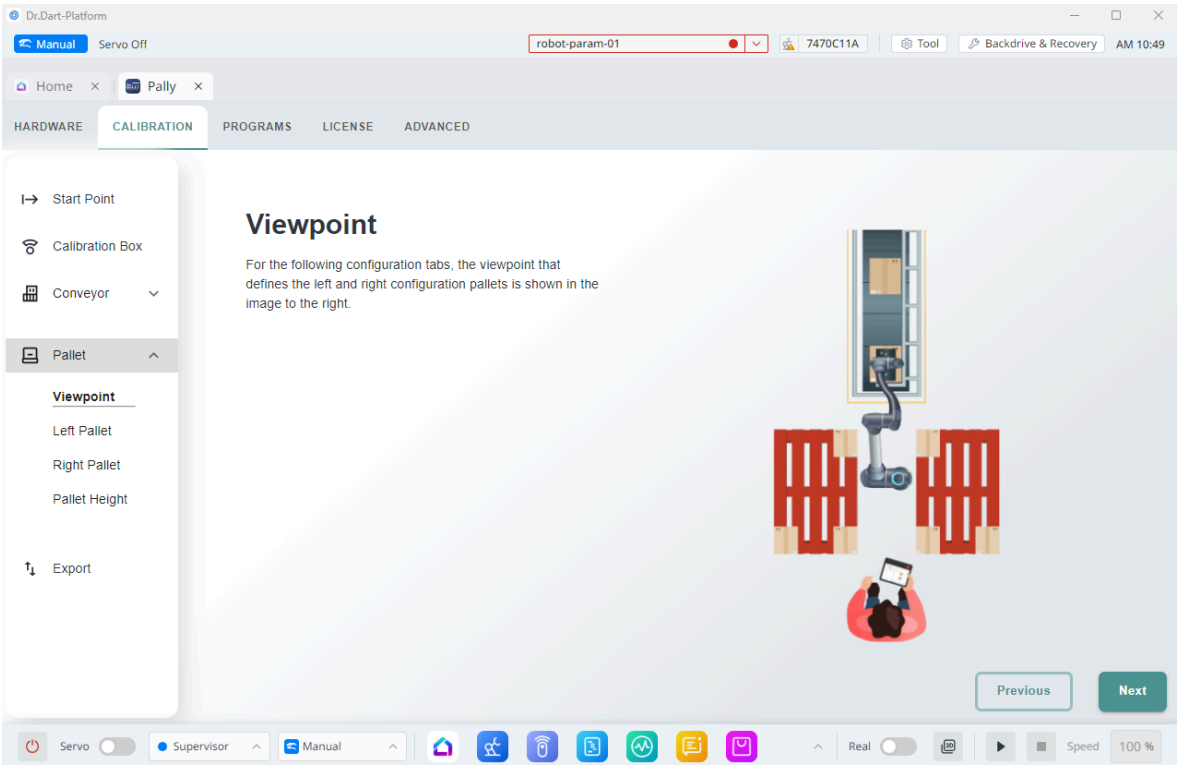


Illustration of the pallet calibration points and viewpoint conventions.

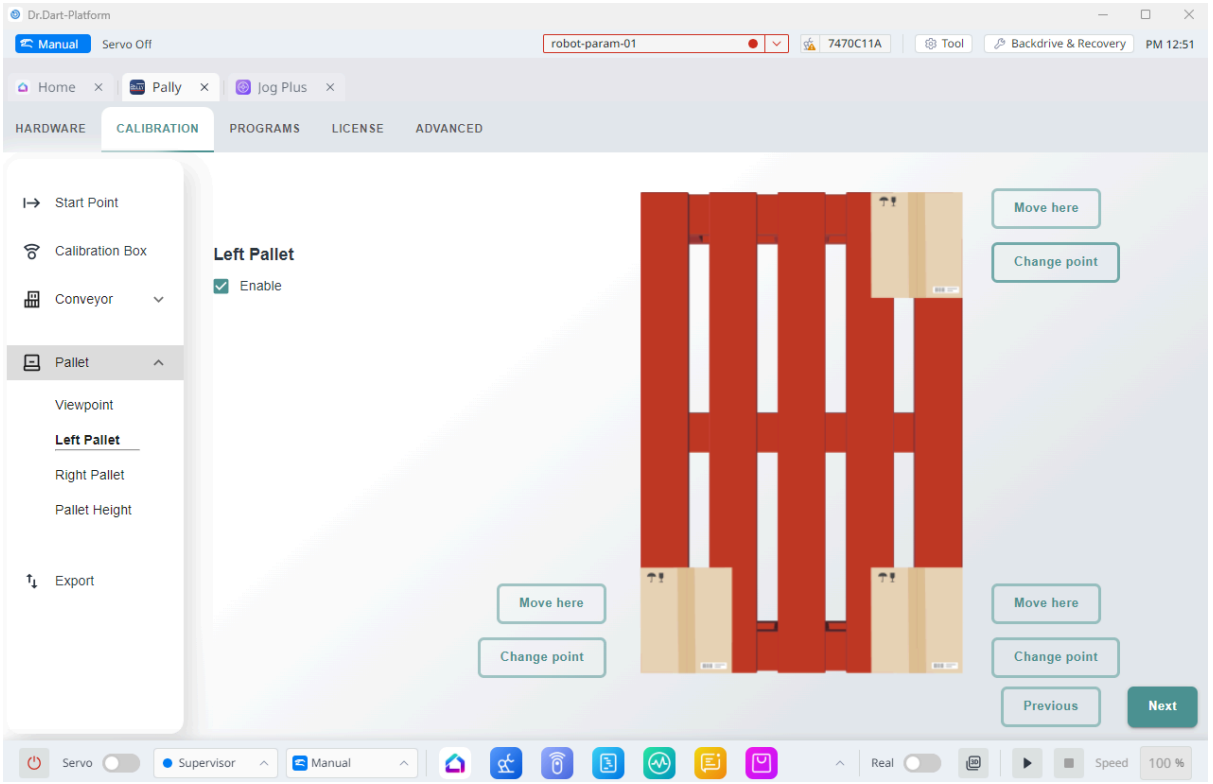
Left Pallet

Note: When calibrating a pallet, make sure the robot is not pushing the calibration box on the pallet, preferably keep a little space (ca. 20 mm) under the calibration box. This ensures that you do not get protective stops caused by the box being pressed down on the pallet when suction cups or foam decompress.

Each pallet requires 3 calibration points. We recommend that you activate suction on your gripper when you are done with pick up calibration. That way you ensure that the box is in the same position during pallet calibration, and this makes it easier to do a good calibration.

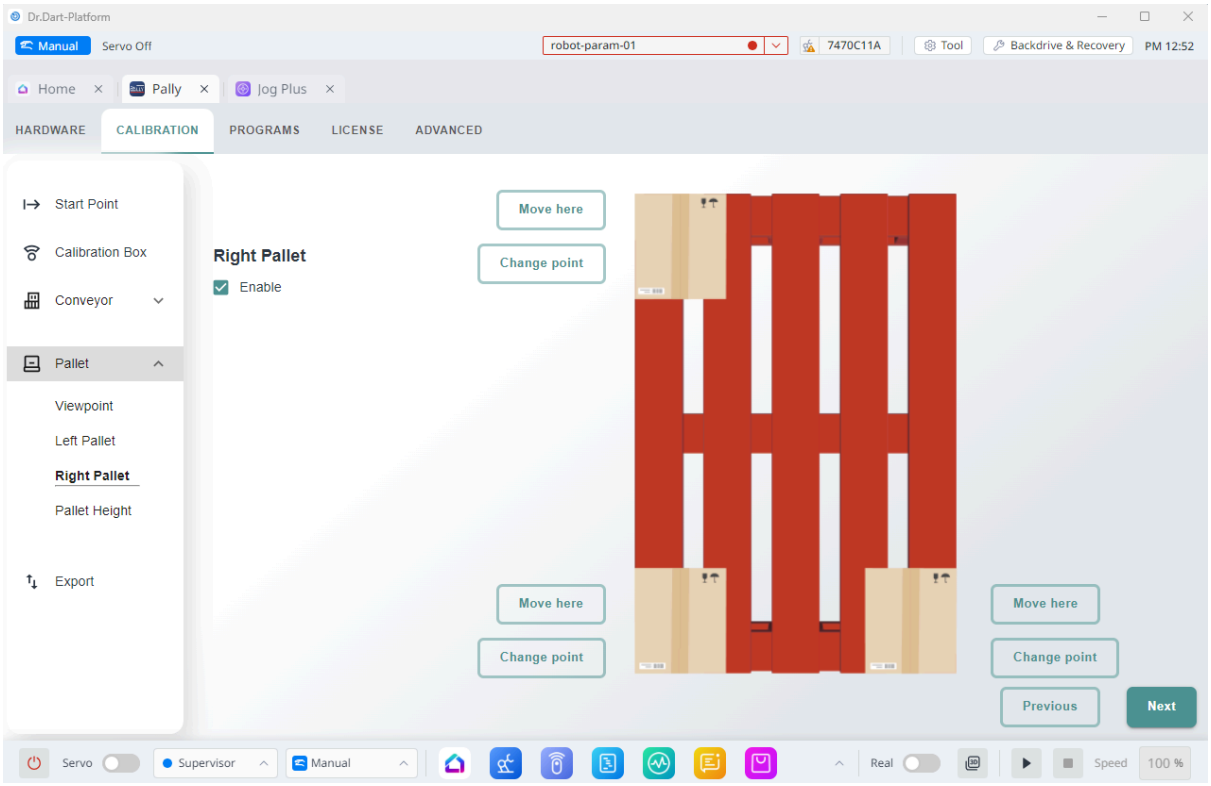
Keep the rotation of the gripper unchanged on all 3 positions

Make sure that the box is located perfectly in the corner before setting the first position. Then you can move using the tool view and move in straight Y or X direction to reach your next position. If the box drifts outward or inwards on the pallet it can indicate that the pick up position is not calibrated correctly.



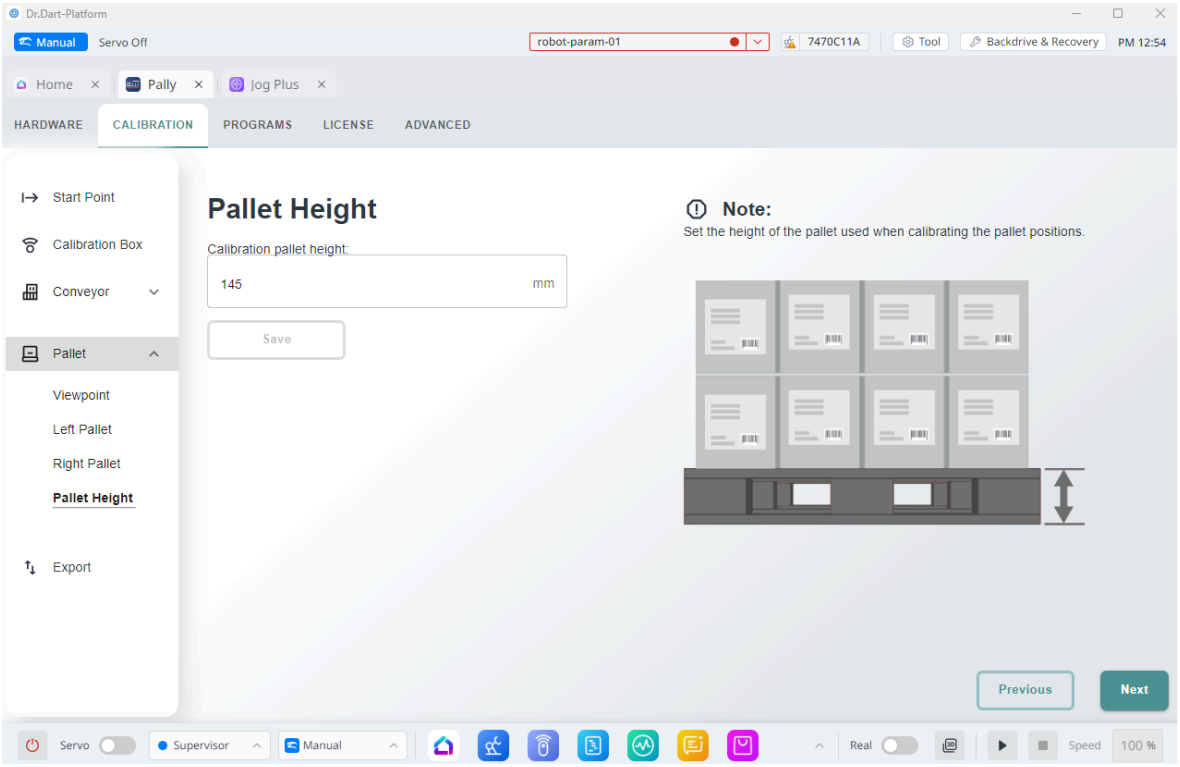
Right Pallet

Repeat the steps from the left pallet



Pallet Height

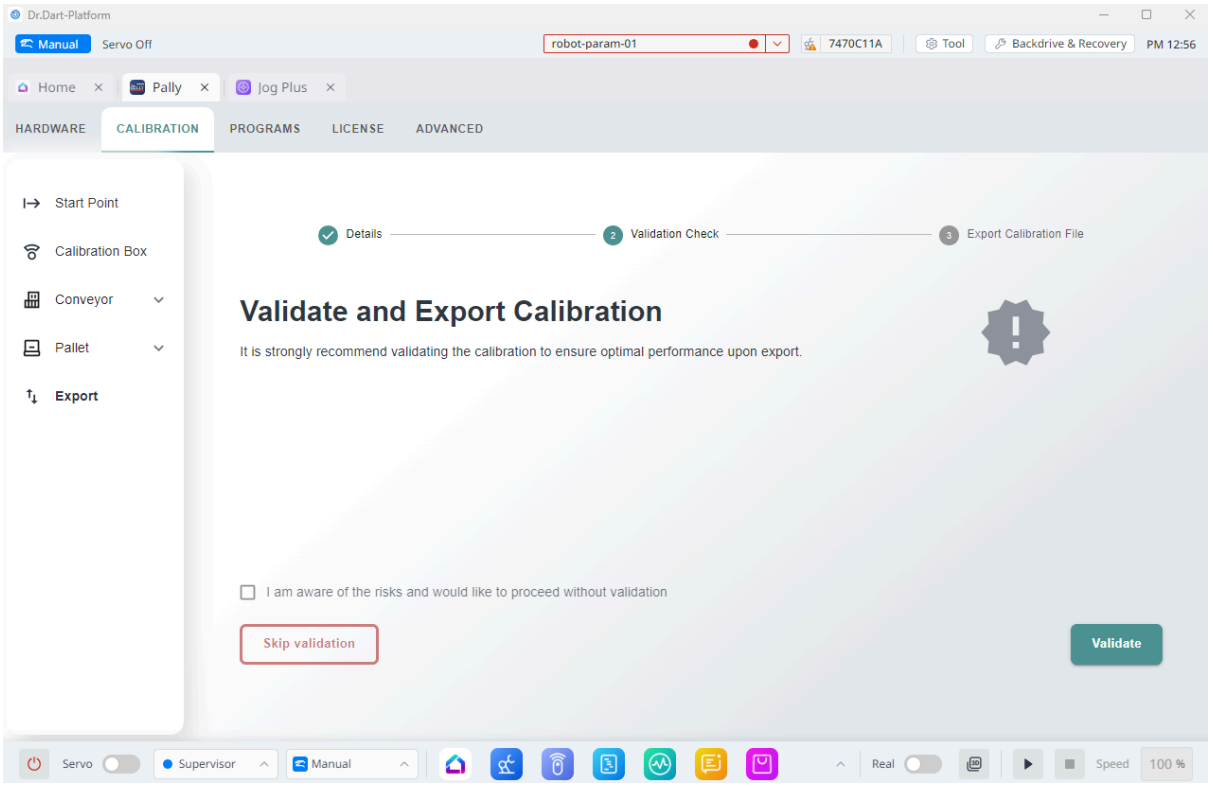
Measure and enter the height of the empty pallet that was used during calibration, if you want to palletize on pallets with different heights.



Height of the pallet used during calibration.

7.5 - Export

We recommend that you run a test of the calibration to ensure that the calibration is good. You find this Validate and Export Calibration in step 2 of the Export.



The calibration test wizard is a utility that can be used to verify the calibration points. The pick position on the conveyor, left and right pallet positions can be tested.

There are several different types of tests included, which are performed sequentially.

- The test wizard performs several calculation routines to identify some of the most typical mistakes usually made during calibration.
- It can move the robot along a path that is generated based on the current calibration data, and ask the user to confirm that the robot motion looks consistent to the physical conveyor and pallet placement.

Before each test step, the wizard explains the test that will be performed next. The user can start or skip the test and proceed to the next step by pressing the corresponding button.

When a test fails, a warning message appears with possible explanations of the failure. The user can stop the test wizard and fix the error, or proceed to the next step.

The test cases are divided into test groups. Currently there are 3 groups:

- Testing the pickup position
- Testing the right pallet position (in projects where available)
- Testing the left pallet position (in projects where available)

The test cases for the left and right pallet are identical.

Testing a pickup position contains the following test cases

- Moving parallel to the conveyor: this test can help detect calibration issues related to TCP settings.
- Moving to the pick position with 180 degrees rotation: this test can help detect calibration issues related to TCP settings.

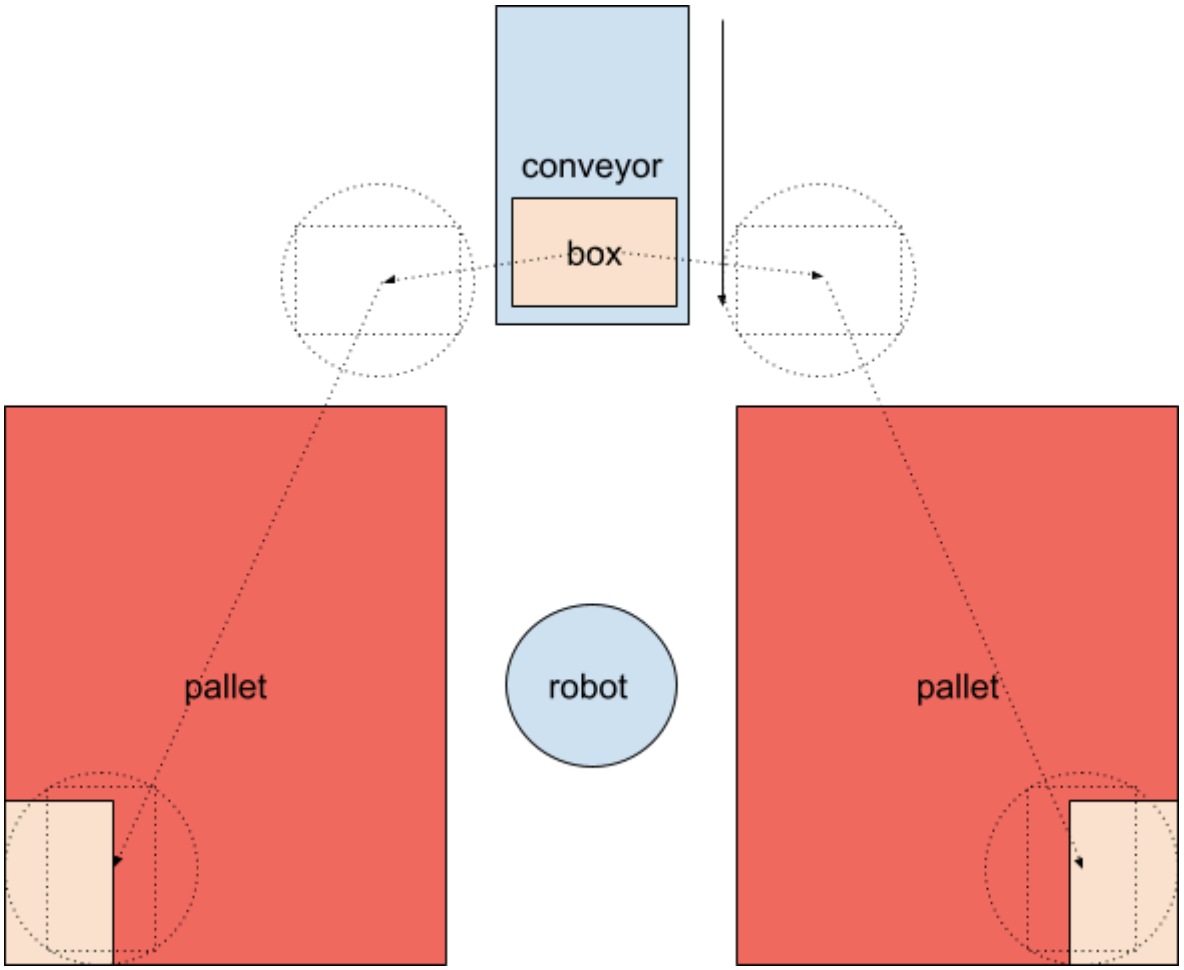
Testing a pallet position contains the following test cases

- Verification of the pallet size: this test can verify that the calibration box was used as required.
- Verification of the pallet shape: this test can verify that the correct pallet corners were used during calibration.
- Verification of the calibration points: this test can check that the robot position at each pallet corner is accurate enough.

7.5 - Examples: typical configuration scenarios

7.5.1 - Classic setup with one optimal pickup position

The recommended configuration is shown below. There is enough room around the pickup position where the robot can move the product towards the pallets. Usually no additional configuration is required.

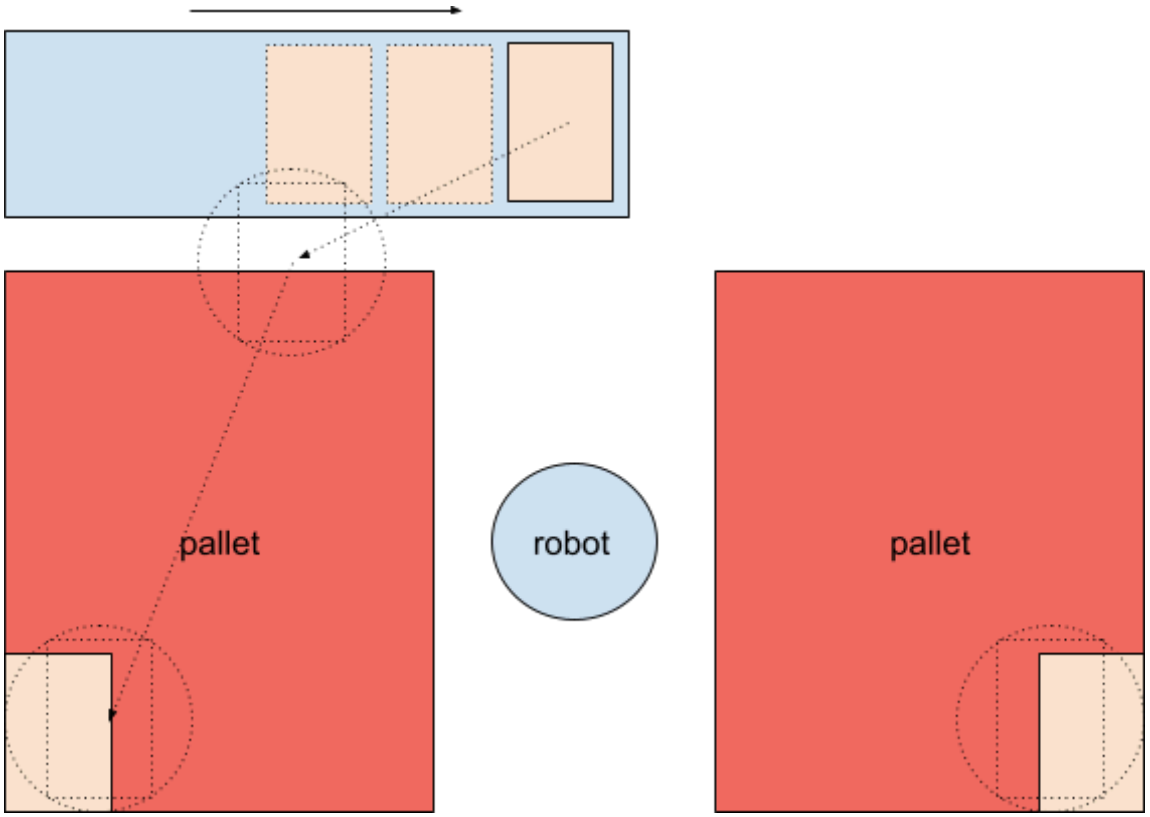


Classic setup with optimal pickup position.

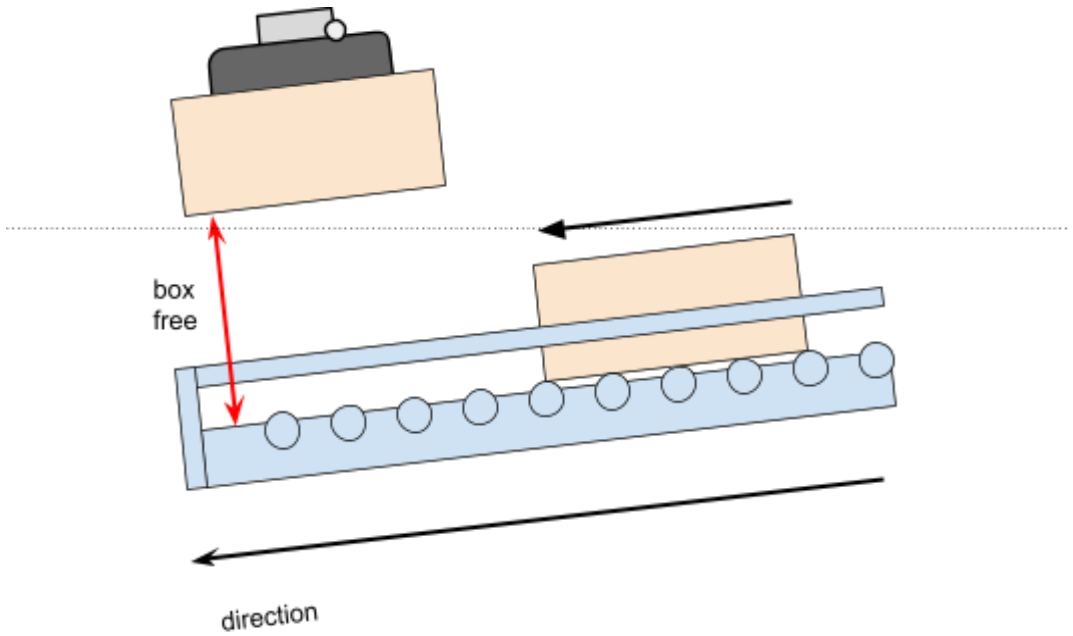
7.5.2 - One pickup position with products coming in from the side

Setups similar to illustration introduces a few points to be taken into consideration.

- Pickup position should not be in the pallet shadow, i.e. behind a pallet seen from the robot.
- Use limited smart exit search.
- Choose *box_free* high enough to move the box freely above the next boxes.



Pickup position sideways. The box will be moved above the other boxes on the conveyor.

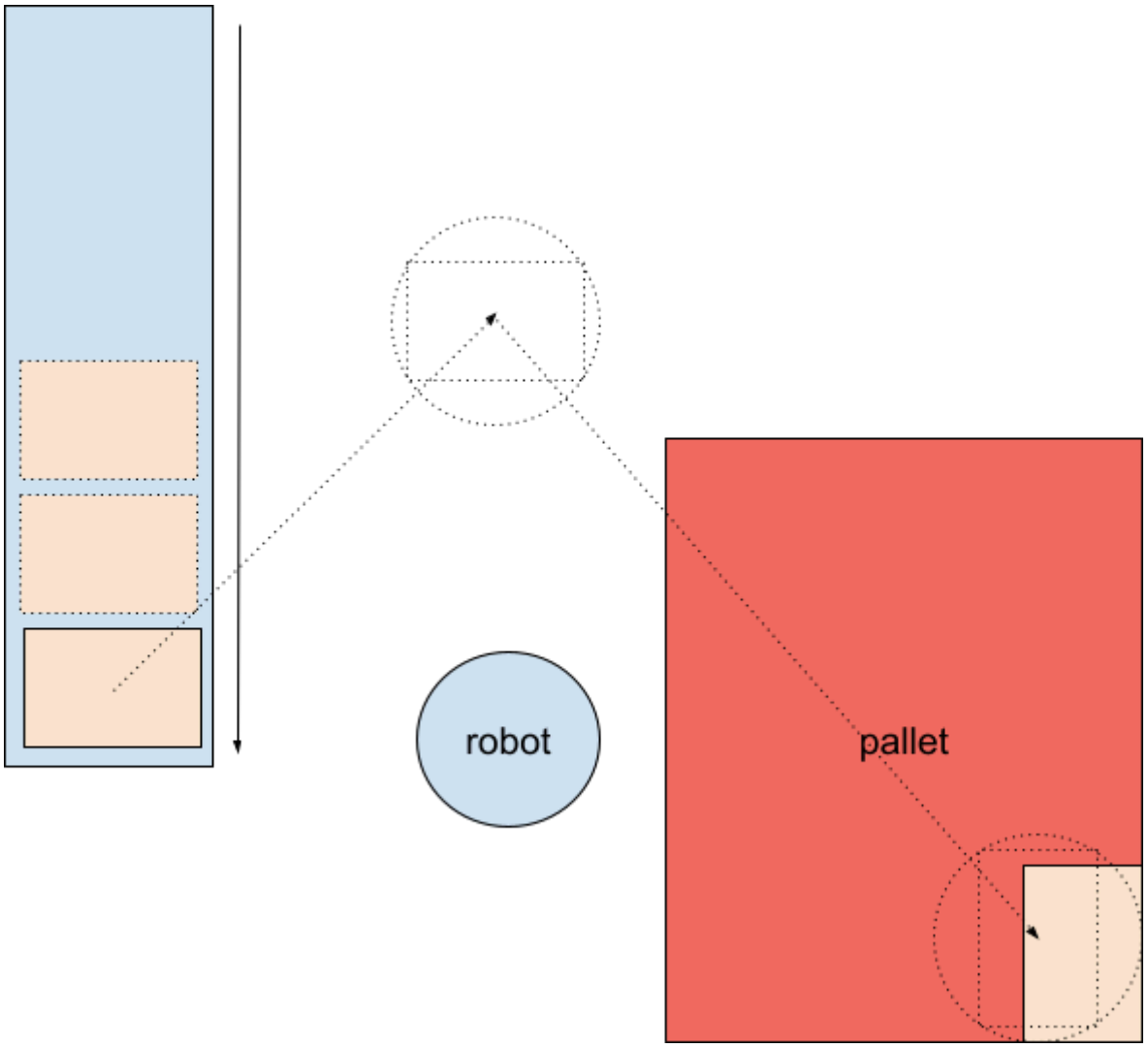


Choose a box free distance that is big enough to move the box above the others.

7.5.3 - One pallet position and one pickup position on the opposite side

Setups similar to illustration introduces a few points to be taken into consideration.

- Choose *box_free* high enough to move the box freely above the next boxes.
- Use relatively large values for *smart exit min_x* and *negative min_y*.
- Recommended settings (approximately):
    - smart exit search area X: 500 - 1000 mm
    - smart exit search area Y: -900 - -700 mm (negative)
    - box free: 350 mm (or more, depending on the box height)



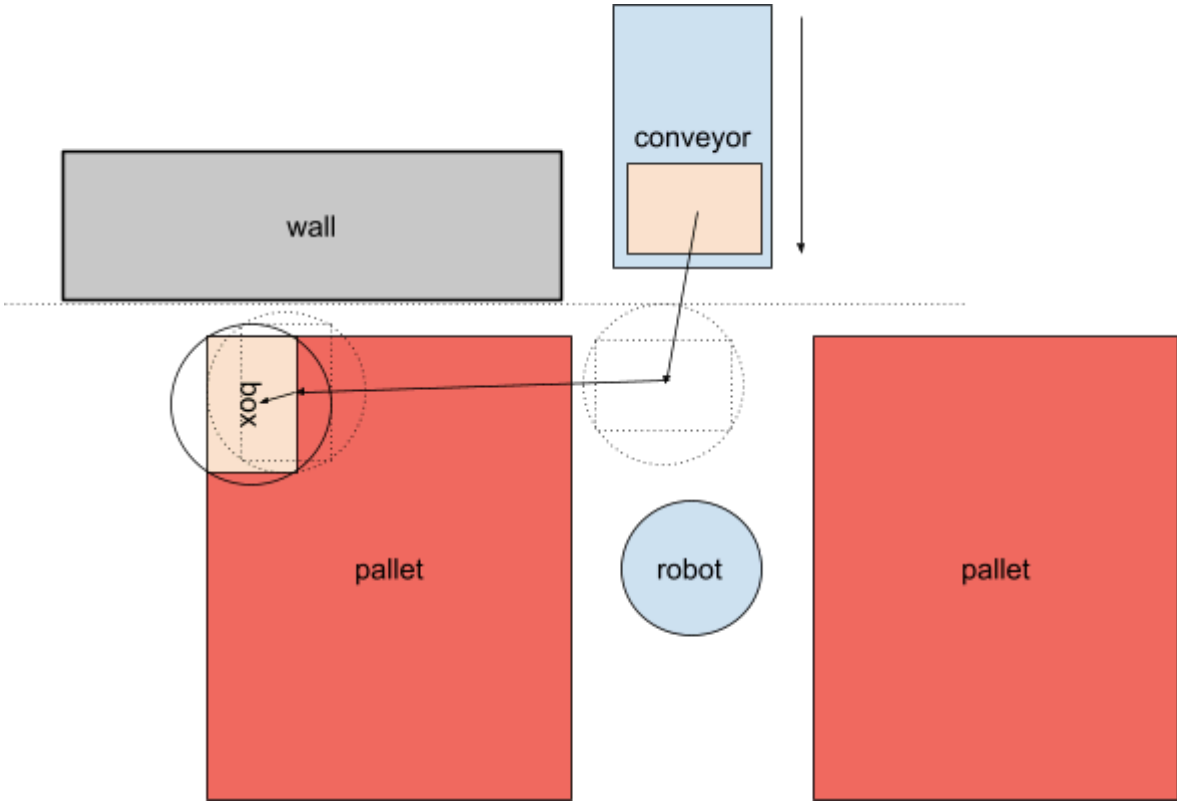
Pickup position and pallet position are on the opposite sides of the robot.

7.5.4 - Limited space, walls other object

Setups similar to illustration introduces a few points to be taken into consideration.

- Individually configure smart exit for left and right pallet.
- The smart exit search area may be limited to one single point or a very small area, to avoid collision with the wall.

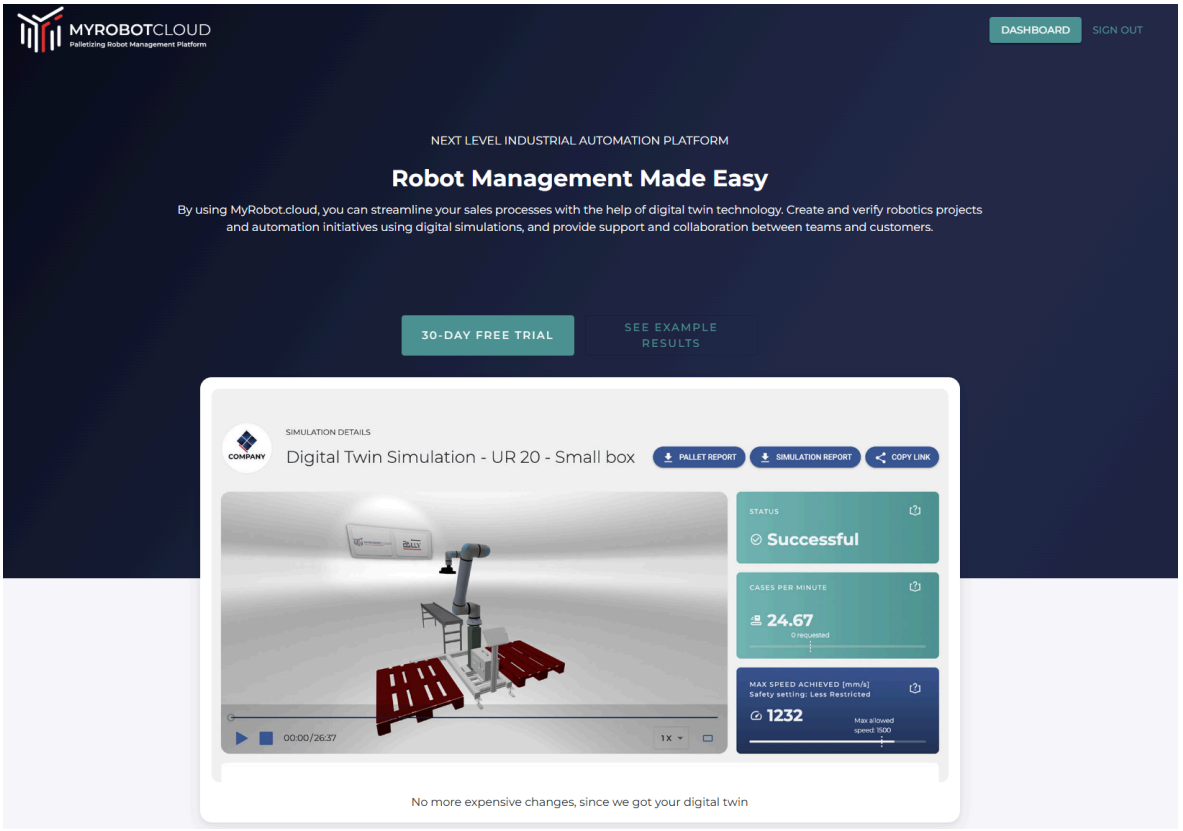
Note: Reserve enough distance between the wall and the pallet to perform box rotation.



Path planning with fixed *smart exit* position, to avoid collision with the wall.

8 - Patterns and Programs

To start defining patterns and creating programs, use a web browser and navigate to https://myrobot.cloud/
where you can create your patterns in a graphical editing environment. All patterns can be exported to a regular text-file and further edited in any text-editor.

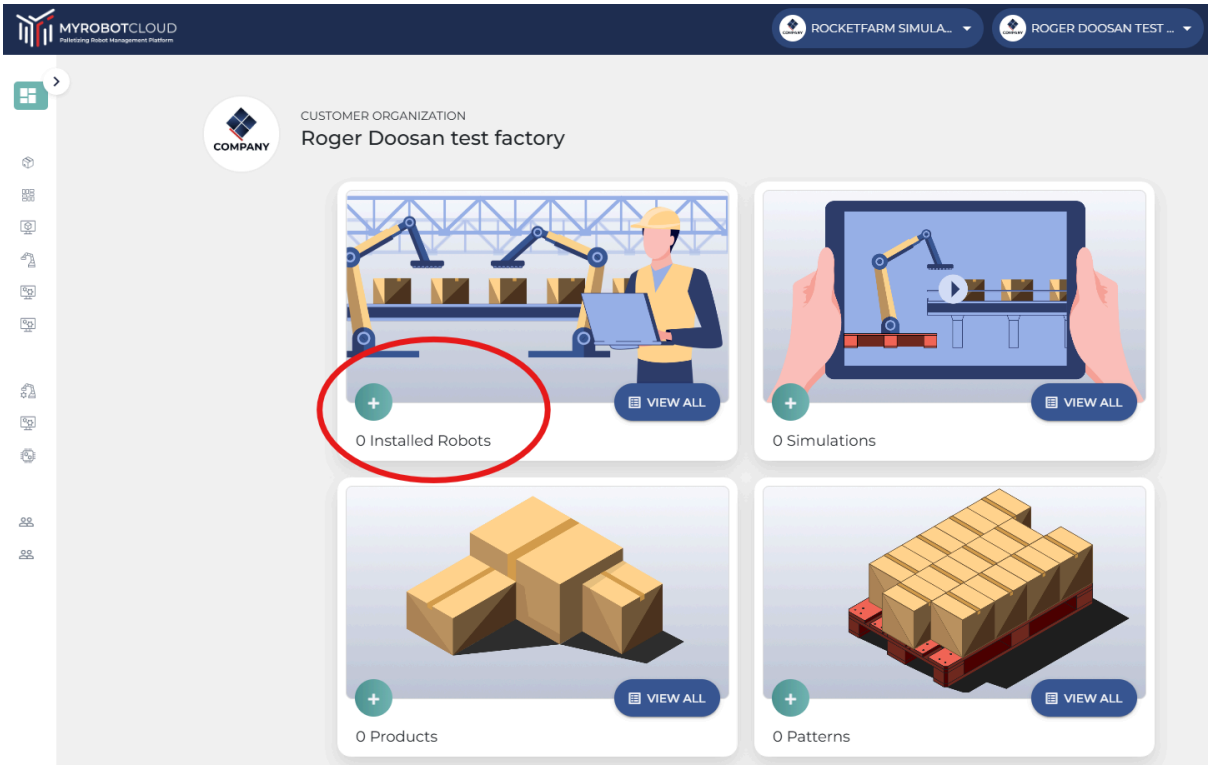


The graphical pattern editing environment at https://myrobot.cloud/

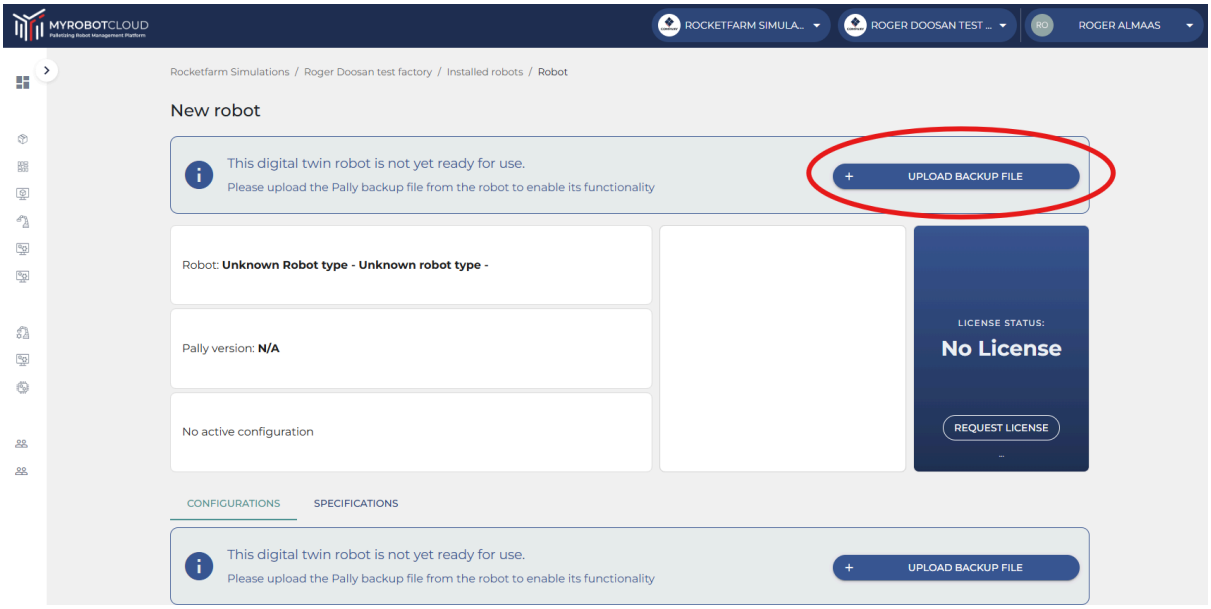
8.1 - Upload Calibration

Select customer to work with

On the customer organization dashboard, press + on *Installed Robots*



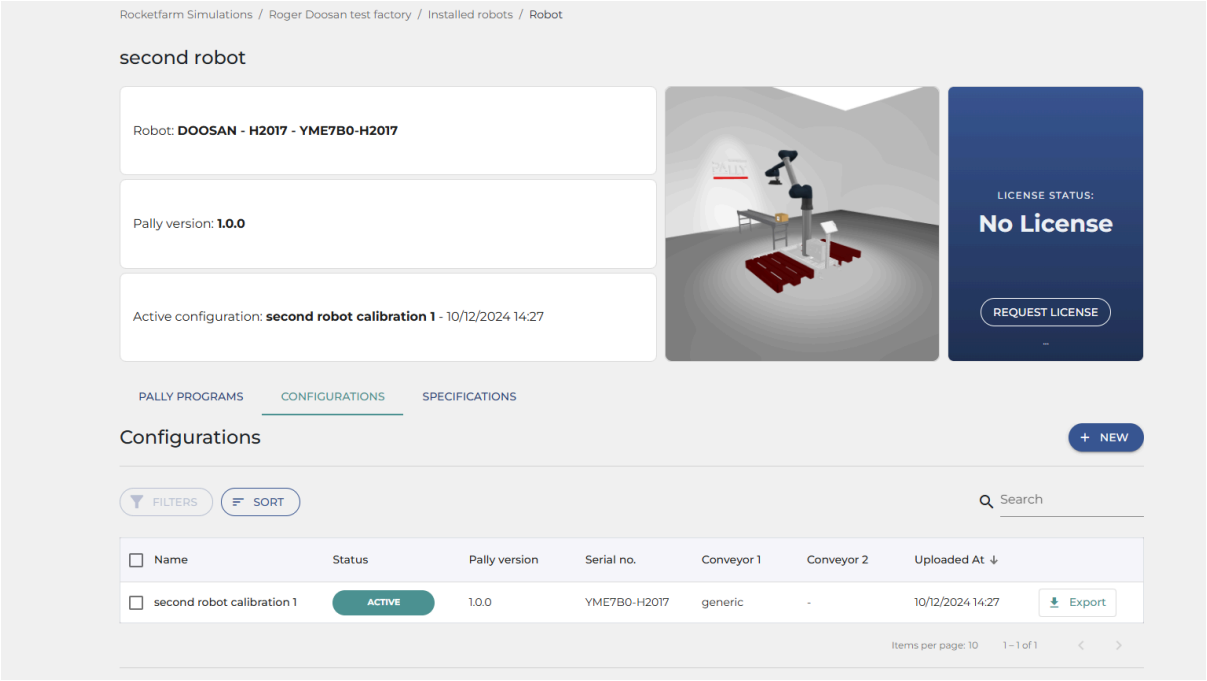
Select the new robot to add a new backup and then press Upload Backup file to import the calibration file from the Doosan robot



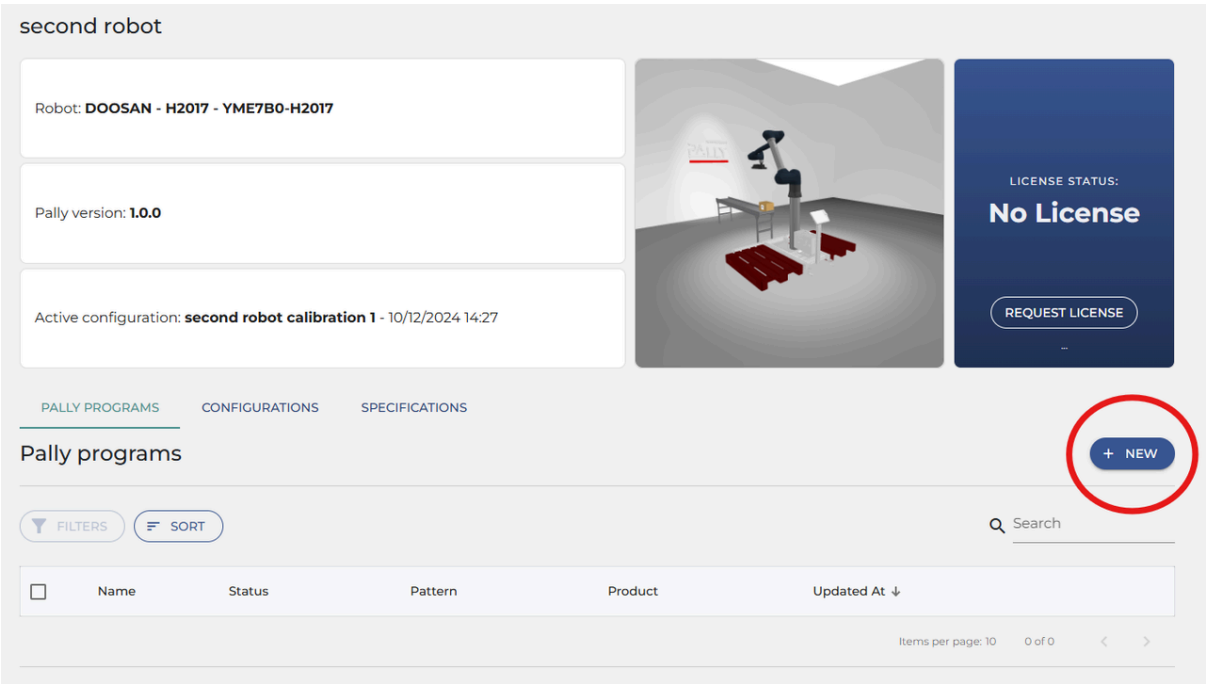
Enter a name for your robot and press *Upload Calibration File*

Give you backup a name and choose the generated calibration file from your robot

Once you've done this you will see your robot listed.

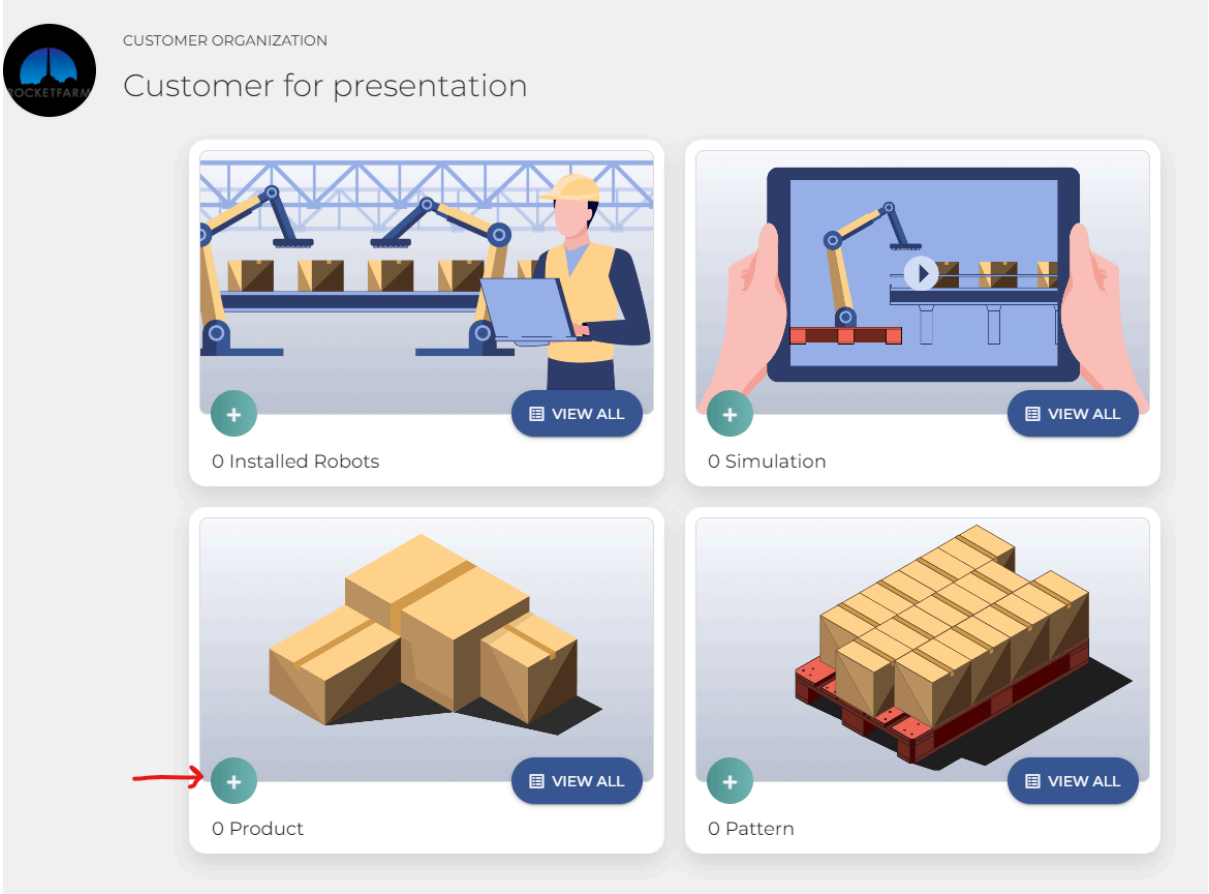


After activating the license you are ready for generating the Pally programs.



8.2 - Create products

You can start creating a new product by pressing + in the product frame on the customer dashboard.

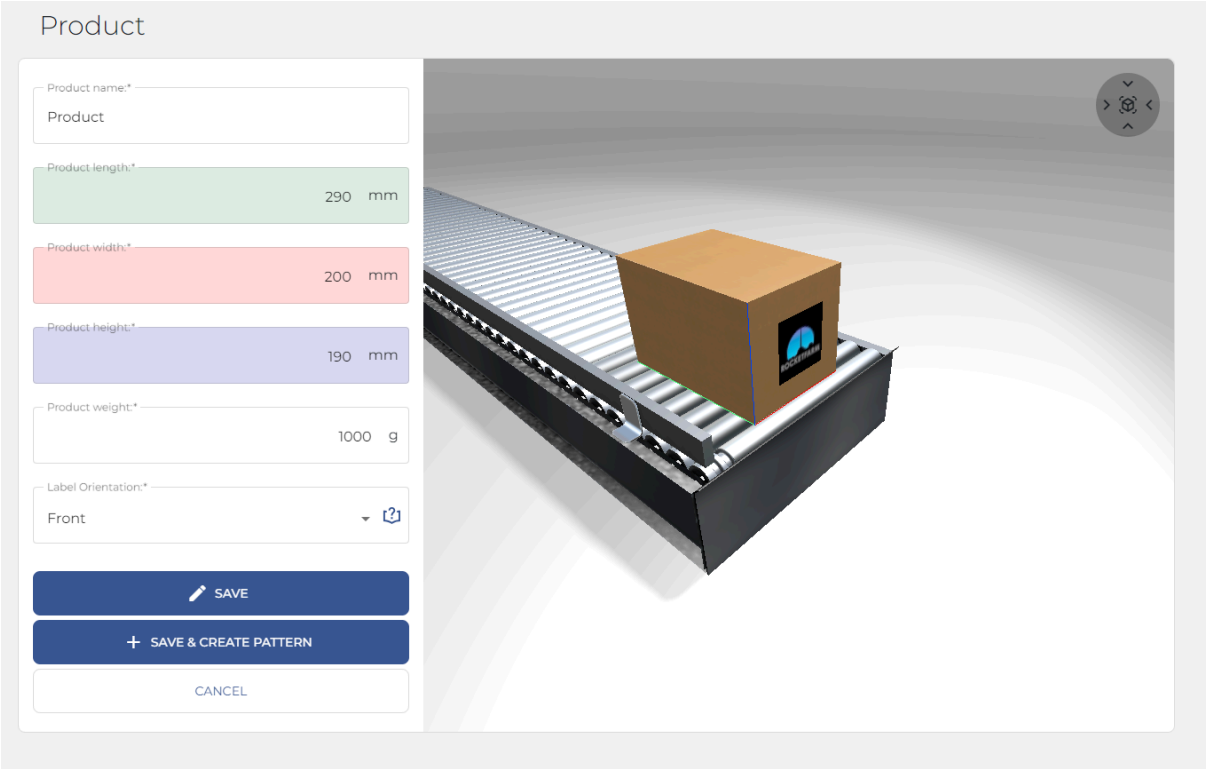


Make note of the orientation of the dimensions as it sits on the conveyor. It's critical that the dimensions you put in for the different sides of the box matches this orientation as the product is coming down the conveyor.

You can see the colour in the input field corresponds to the colour on the box.

Give the product a name

Use the input fields to enter the product dimensions; **Length, Width, Height and Weight** of the boxes in millimeters and grams.

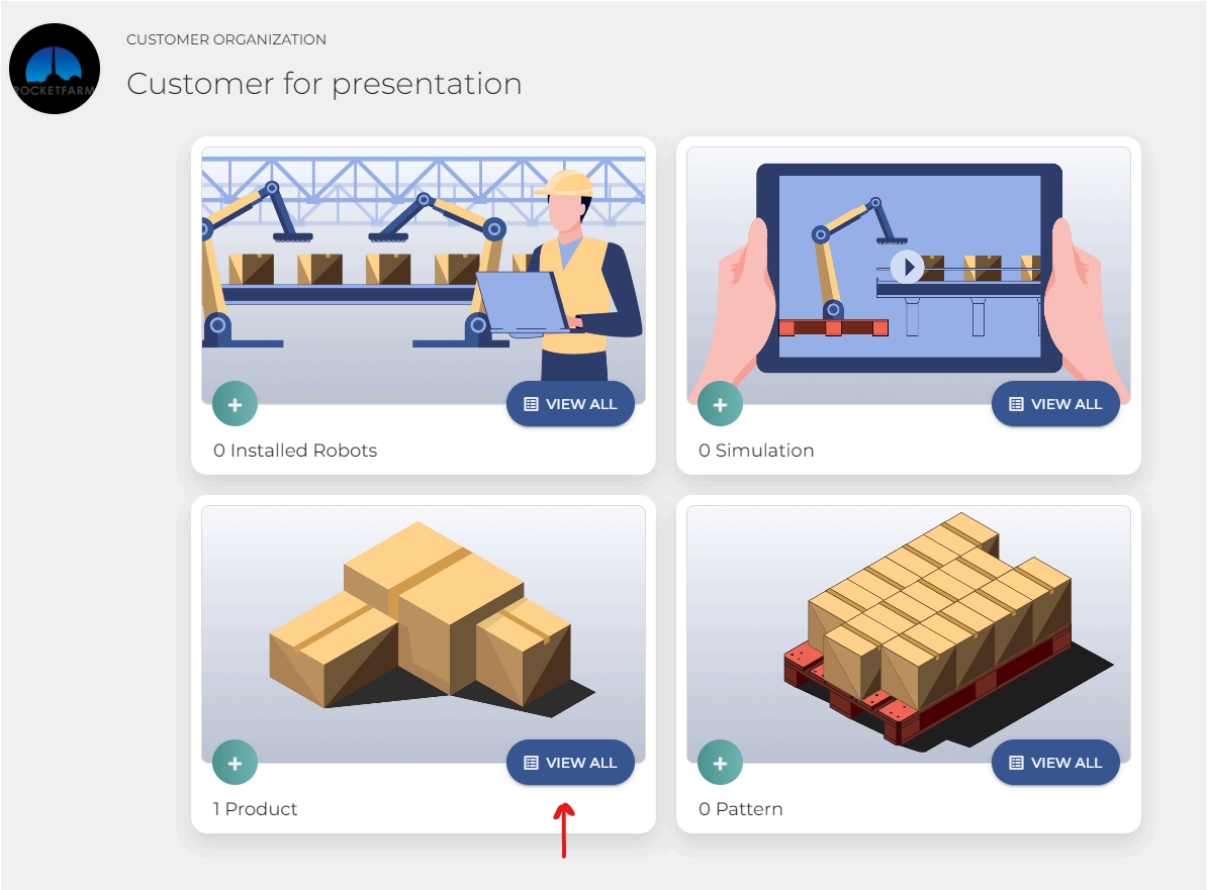


Label orientation: Here you can choose if the product has a label and in what direction the label is oriented. If you have added the logo of you customer, that logo will be shown as the label on the boxes. You can also choose to have no logo on the boxes, by choosing “None”

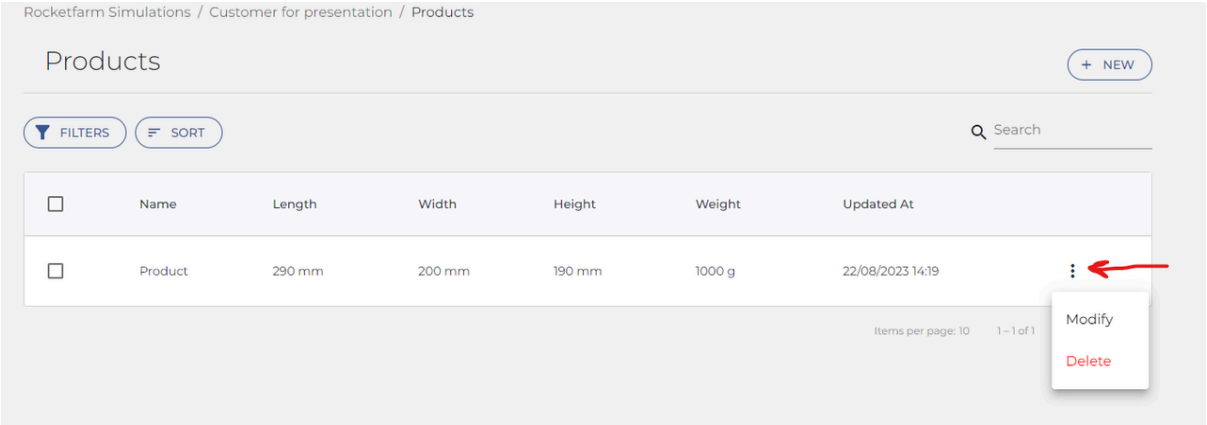
Click “Save”.

By creating a product you can easily reuse the product for multiple patterns without having to create a new product for each pattern.

You can now find your product in the product view

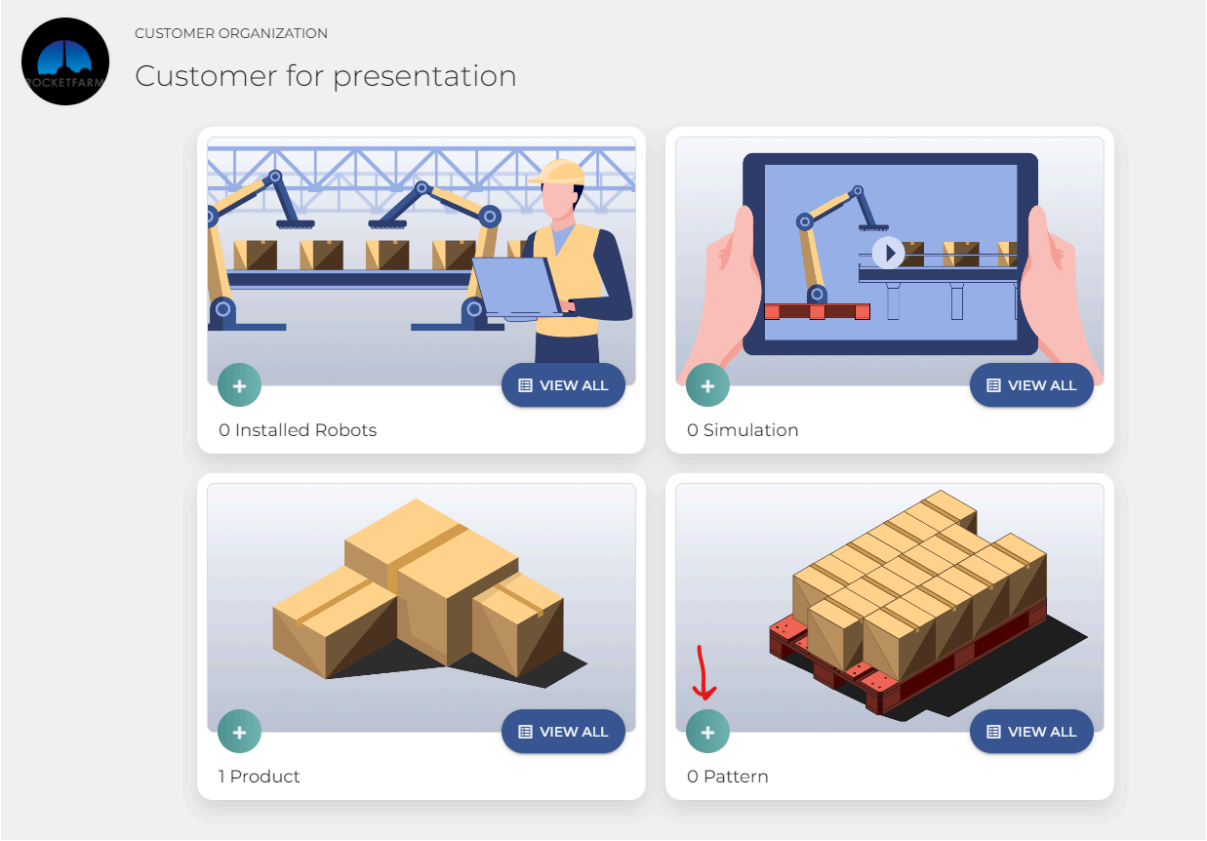


You can edit your product by pressing the 3 dots on the right of the pattern



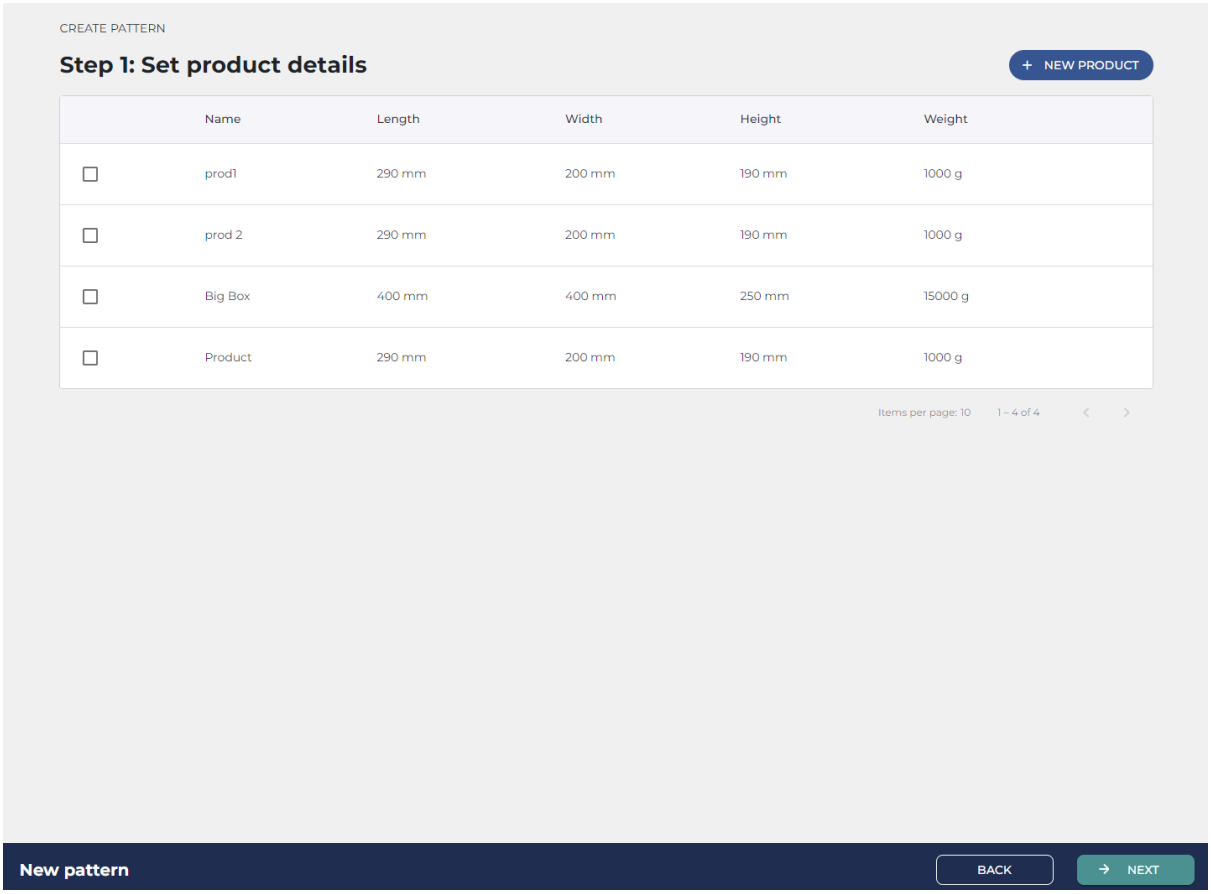
8.3 - Create Pattern

You can start creating a new pattern by pressing + in the pattern frame on the customer dashboard.

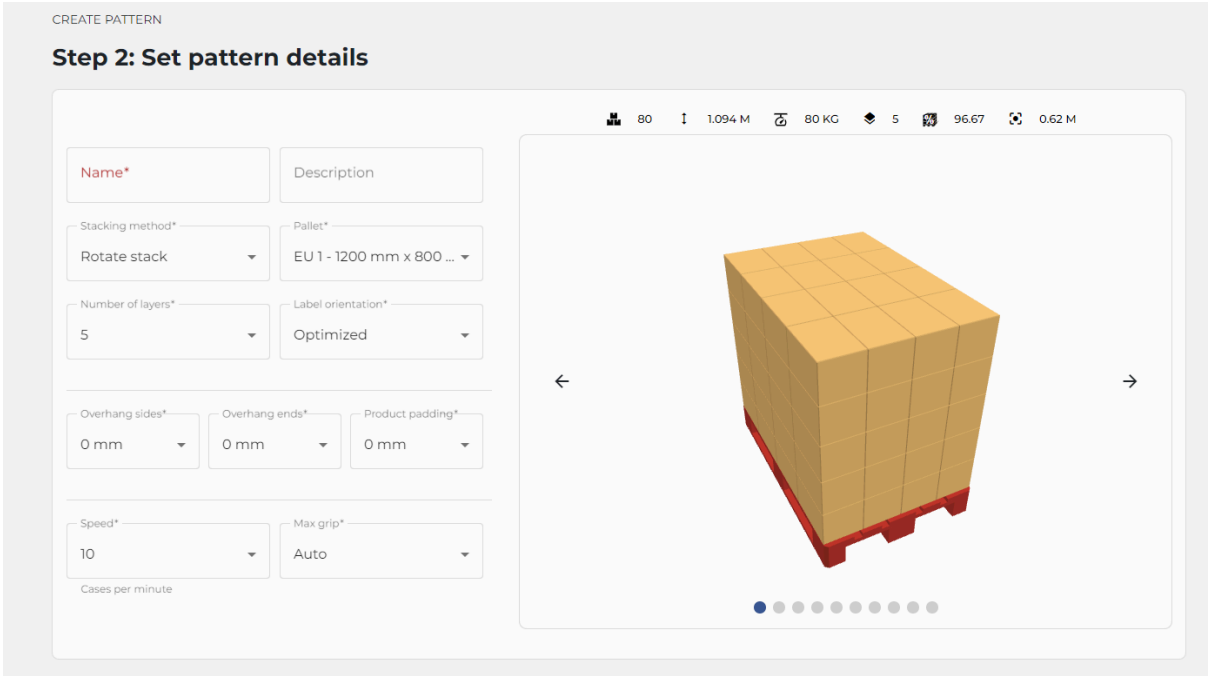


You can now choose if you want to create a new product or choose a product you have created previously.

Then you press “Next” in the bottom right corner



Now you can just go through the steps:

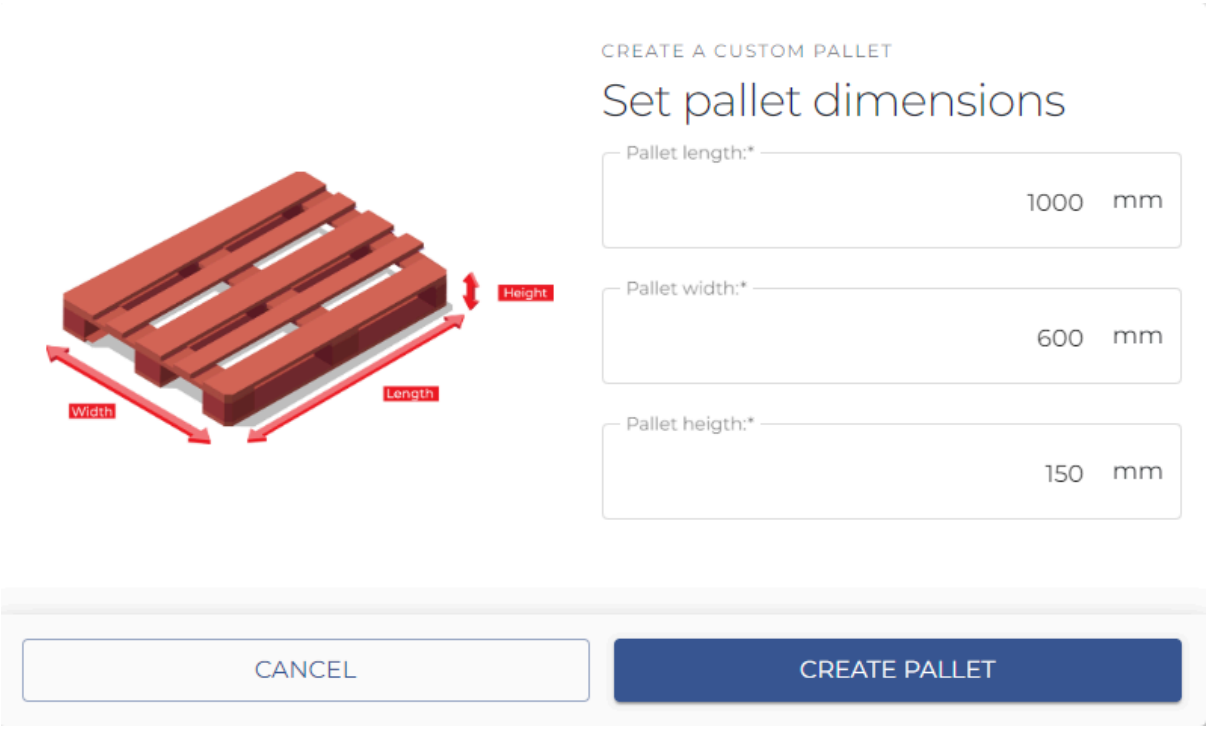


Name : Give the pattern you are about to create a suitable name

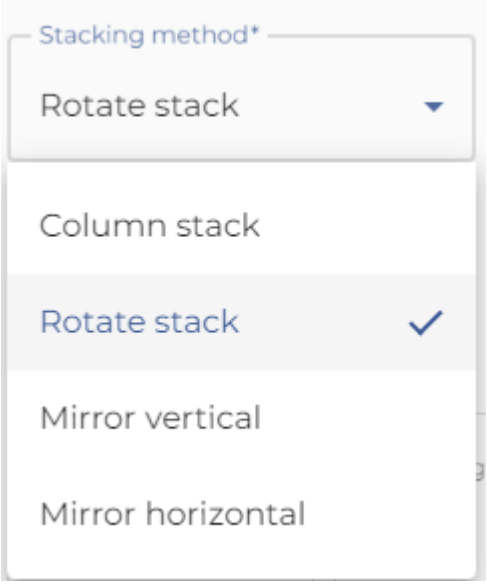
Description : Here you can add aditional description to your pattern

Pallet : Choose between different standard types of pallets; EU, US, AU, CHEP, etc.

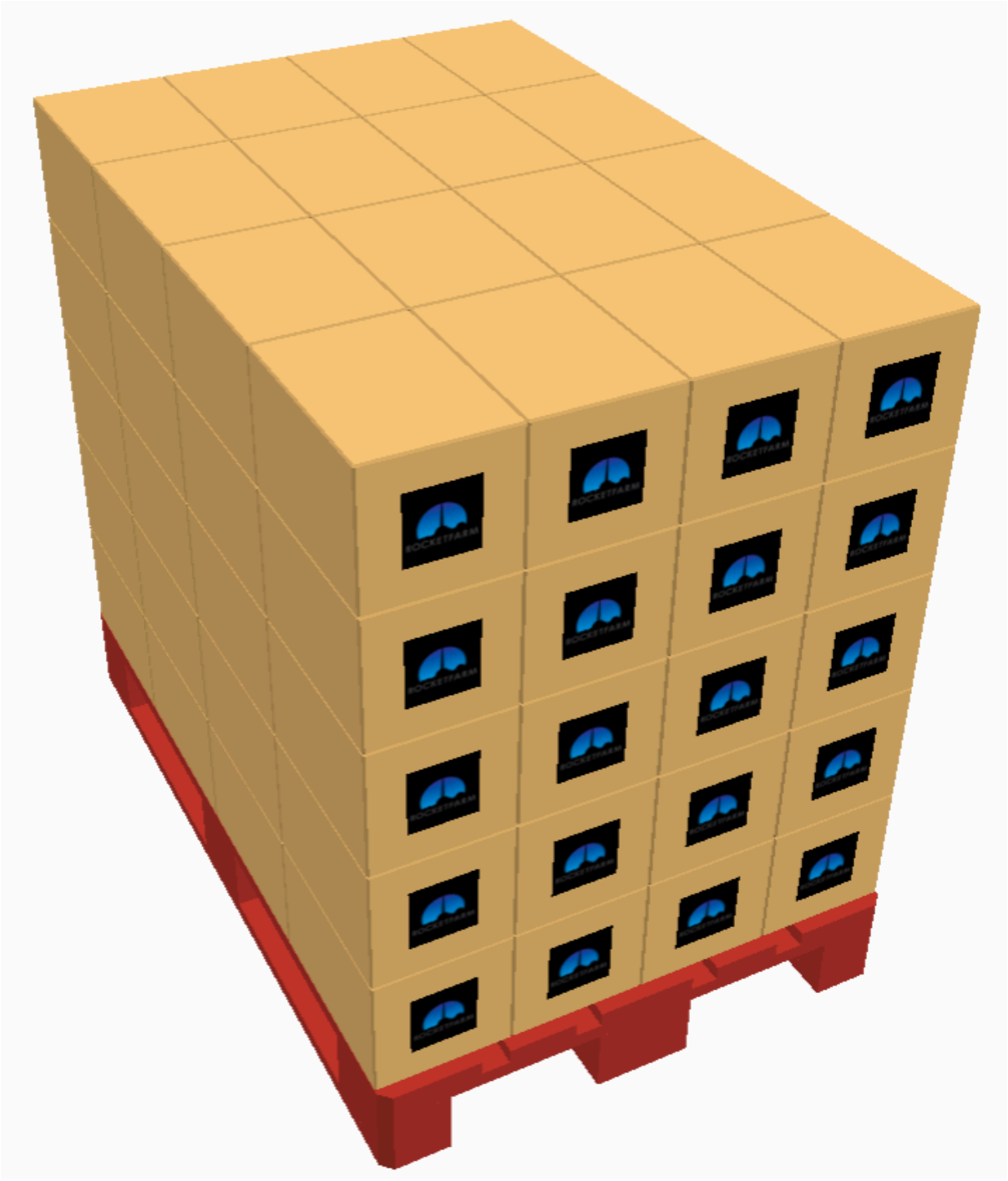
Custom Pallet? : Press here to create a custom pallet with your own choice of dimensions



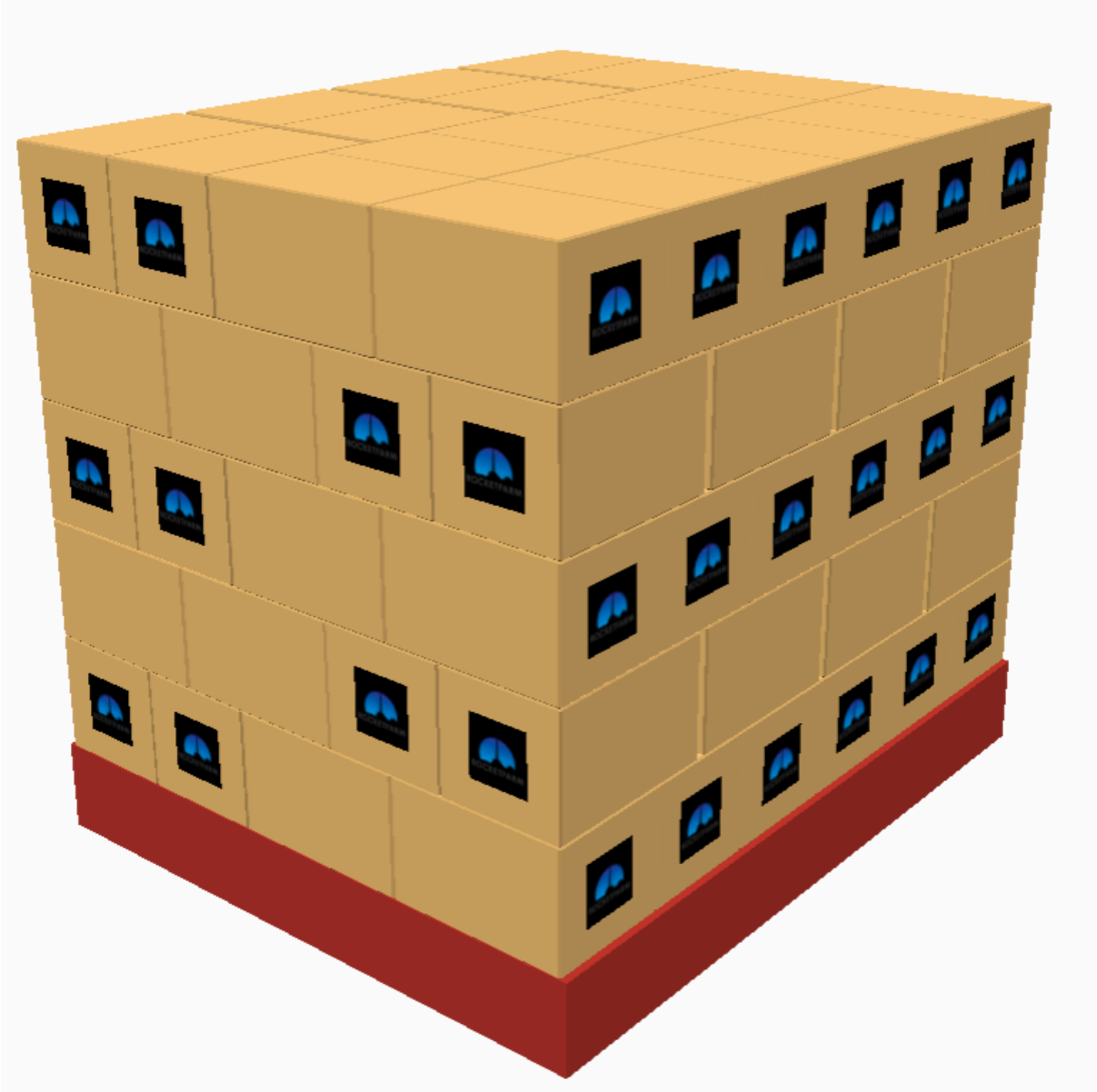
Stacking method : define how you want the boxes stacked on the pallet.



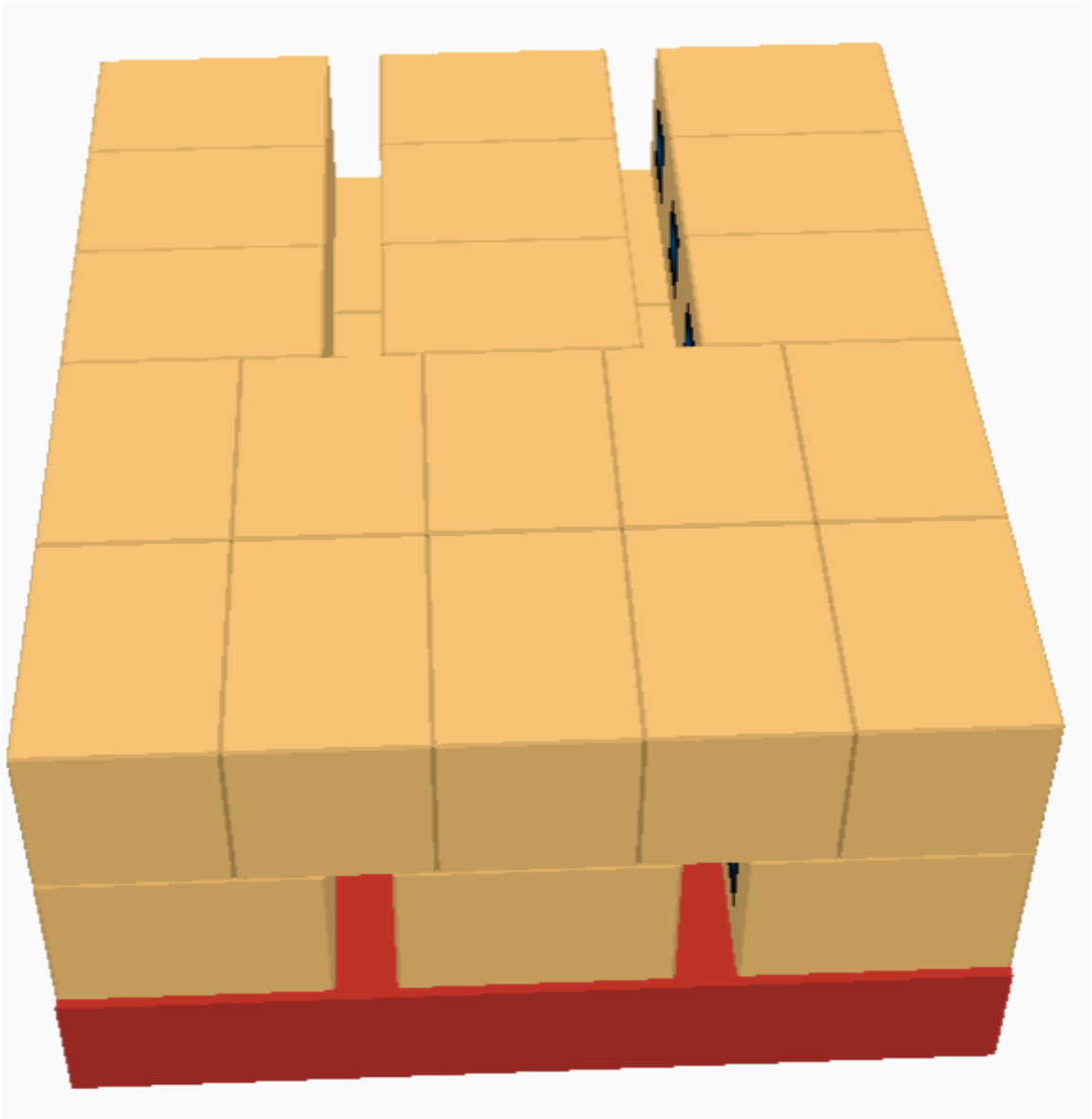
Column stack : stacks the boxes on top of each other in columns



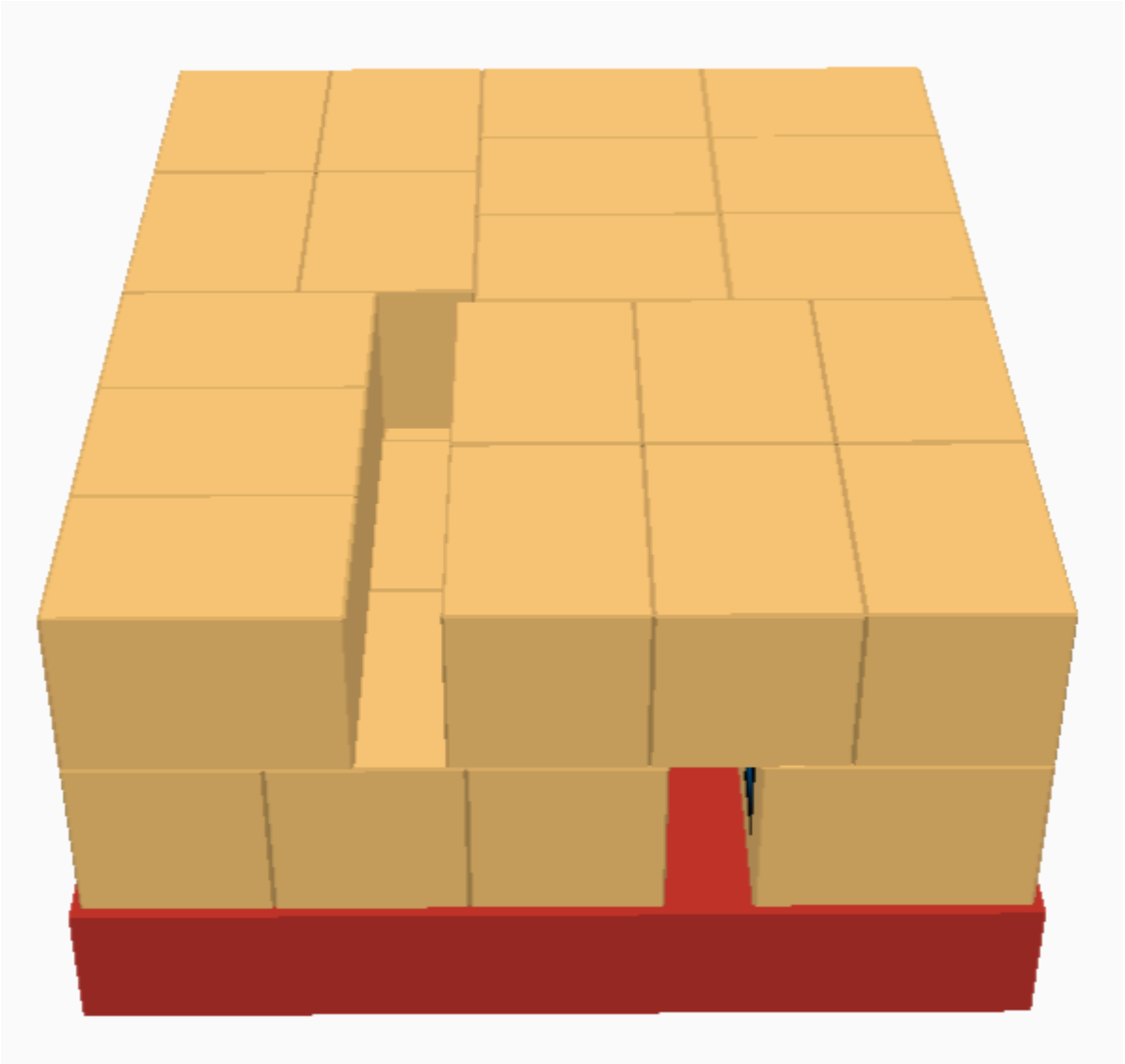
Rotate stack : stacks the boxes in an overlapping pattern by rotating the boxes from one layer to the next. This often creates a more stable pallet.



Mirror vertical : It mirrors the different layers on the long side of the pallet

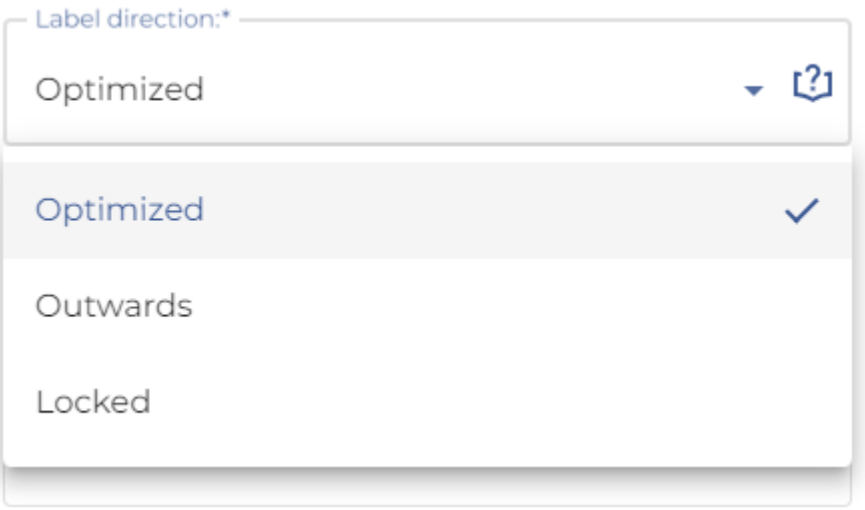


Mirror horizontal : It mirrors the different layers on the short side of the pallet



Number of layers : Choose how many layers of boxes should be stacked on the pallet

Label direction : Choose how Pally handles label orientation while placing the boxes on the pallet



Optimized : Pally ignores the label direction and places the boxes in the direction that is most efficient.

Outwards : Pally places all the outer boxes with the label facing outwards on the pallet.

Locked : Pally places all boxes with the label facing the same direction.

Using the Outwards or Locked label direction can in some patterns create a reach issue.

Overhang : Means that the placing of the box extends past the edge of the pallet. You can let the boxes extend past the edge on the side, ends or both. Make sure the value entered doesn't make

the box unstable when palletized. NOTE: These dimensions are in millimeters.

Product padding : This is the amount of space to be left between boxes in all directions horizontally. NOTE: This dimension is in millimeters

By adding product padding you remove the option to multipick boxes.

Max grip : This is the maximum amount of boxes to the cobot is allowed to pick simultaneously. When Auto is selected, Pally will automatically calculate how many boxes it can pick for each cycle. If the application requires limitation in the number of boxes handles in the same cycle you

can choose the appropriate number here.

Speed : This is where you set the number of required boxes per minute (cpm) for the application. This will affect if a simulation is successful or not. If the simulation palletize more boxes per minute than this number, a simulation is successful.

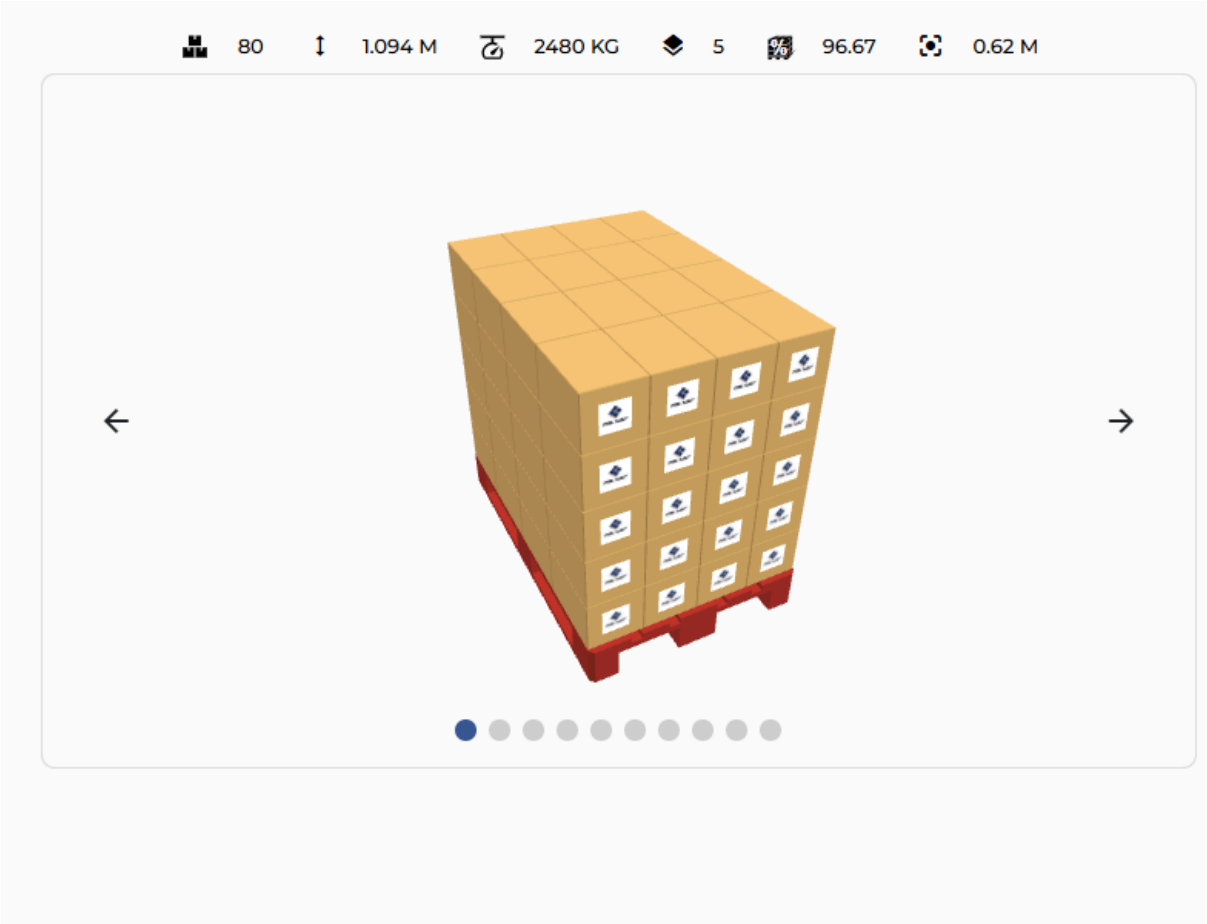
Based on the choices in the steps above, you can now let MRC suggest some base patterns - Pattern 1 to 10. you can change between the different patterns by using the arrows below the 3D model of the pattern.

Above the 3D model you can find useful information regarding the pallet.

The first number is total number of boxes, then you get total pallet height and total pallet weight.

Then you get number of layers and pallet cube efficiency. Pallet cube efficiency is a number that can tell you how efficient the pattern is compared to the pallet size.

Last you get information on how high the center of gravity is for the pallet.

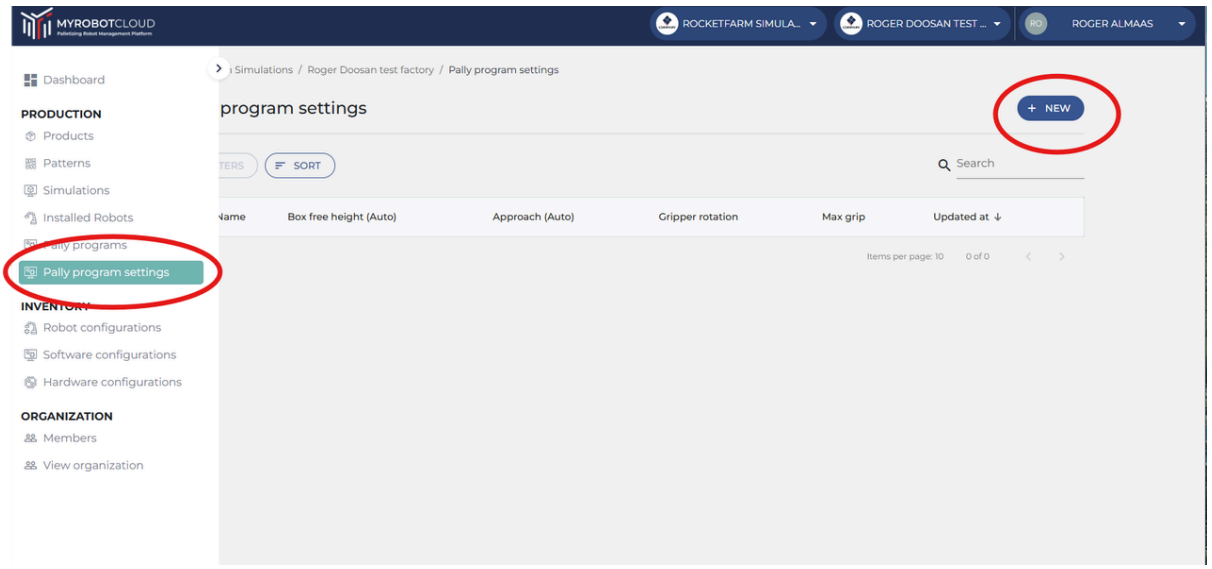


You can also choose to edit the pattern yourself, and build it completely as you like it. Just choose “Confirm & Customize” for advanced editing. Read more about this in MyRobot.cloud Knowledge base - https://rocketfarm.atlassian.net/wiki/spaces/MKB/overview?homepageId=1446118296

When you have created the pattern you like you press “Confirm”” and the pattern adds to the list of existing patterns for the customer.

8.4 - Pally Program Settings

at the customer organizations and go to Pally program settings on the left meny. Then you can add new Pally path strategies.



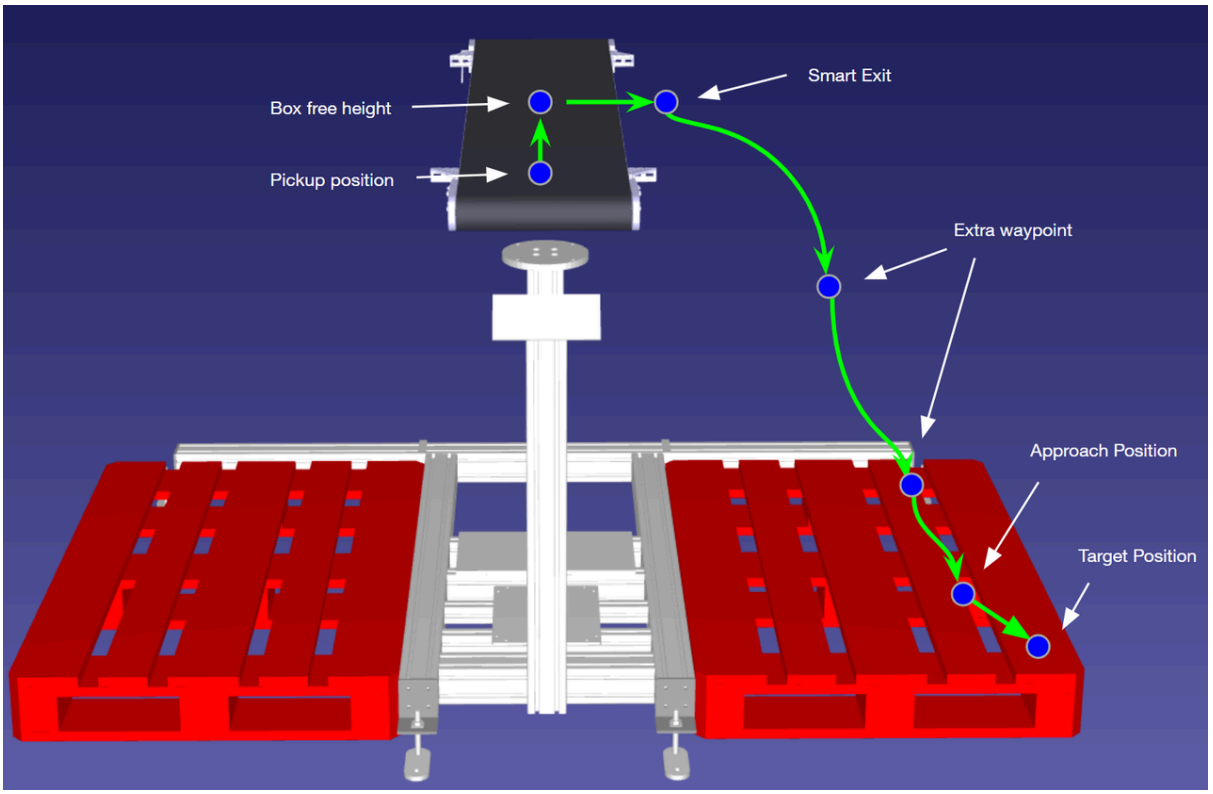
A new dialogue should open where you can set the path planning parameters.

Create Pally Program settings

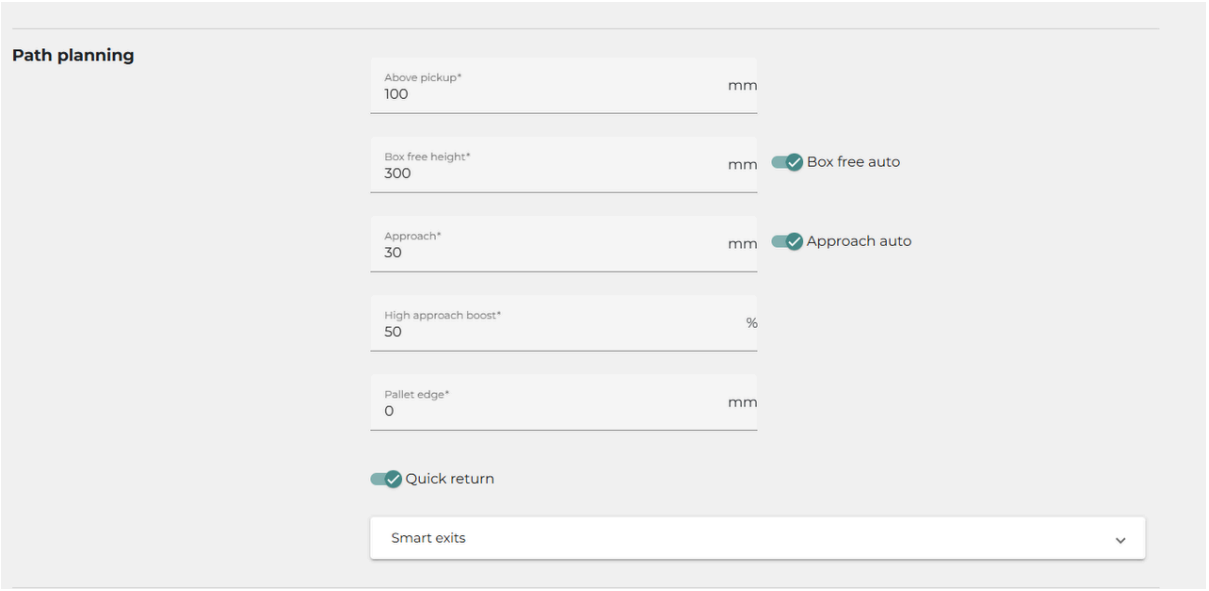
Name your strategy

8.4.1 Path planning

Visualization of the different waypoints for a box path.



Set your path planning values



Above pickup - Gripper distance from the top of the box, from where the robot will move down vertically until the gripper foam reaches the box surface. At this point the program automatically

activates compressed air (if available) to clean the gripper foam and the box surface before picking.

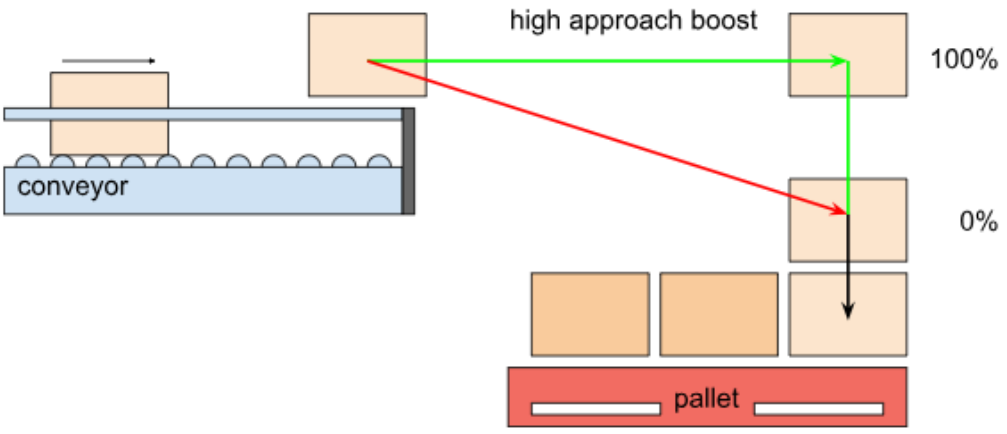
Box free height - The elevation required for moving the box above the side guides and sensors

Box free auto - The value is set to default enabled. Enabled to lift boxes above other boxes on the conveyor if they are higher than the height of the sensors and guides. Disabled to lift each box type to a constant height regardless of the box dimensions (e.g. when lifting over sensors and side guides only).

Approach - Defines the length of the last diagonal movement that is used to put the box on its final destination on the pallet with low speed and acceleration.

Approach auto - The actual approach distances may be recalculated by the program in run-time when rotation between the pick position and the target position is required. By disabling this, the program will finish all box rotations earlier, and approach the target position to the exact distance.

High approach boost - This value must be between 0% and 100%. It determines how high above the other boxes the alternative path planning algorithm moves the box. Lower values let the box move closer to the top of the other boxes on the pallet, higher values keep the box closer to the *box free* height. Only in effect when placing boxes below conveyor height.



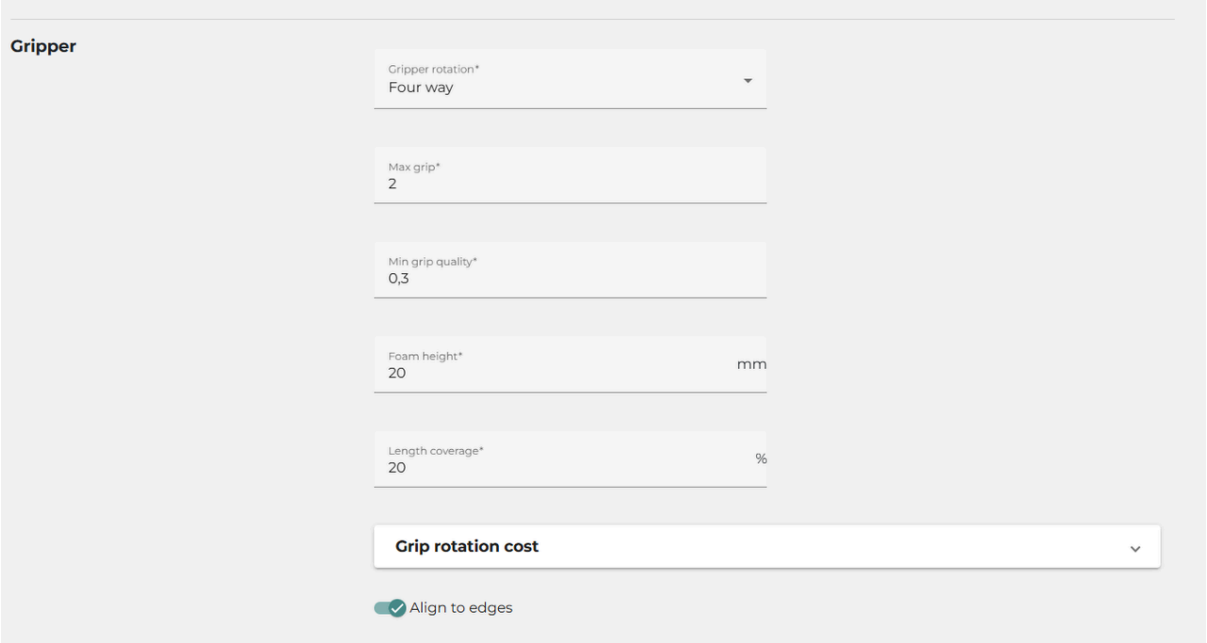
Pallet edge - Enforces a vertical approach with the specified height on the first layer of boxes. Set to 0 if there is none.

Smart exit - Defines a rectangular area where the path planning algorithm can search for an intermediate waypoint between box-free and approach position, to perform linear movements without collision to the robot base.

When entering the smart exit search values, consider the following coordinate system:
X=0, Y=0 is identical to, or straight above the *box-free* position. Positive Y will move the box along the conveyor direction, regardless of the current conveyor and gripper orientation. Positive X will move perpendicular to the conveyor towards the target pallet.

Note: The path planning algorithm will exclude smart-exit waypoints that collide with the robot shoulder, and - when moving below the conveyor level - it will also exclude waypoints that collide with the conveyor. When multiple waypoints within the given search area are possible, the waypoint with the shortest total path length will be chosen.

8.4.2 Gripper



Gripper rotation - Depending on this setting, Pally will evaluate several different gripper orientations at the pickup position and choose the one that fits best to a specific target position.This parameter has special importance when the gripper is offset-mounted. In this case the gripper orientation may have a huge impact on reach and performance.

One way: forward (the gripper direction at the calibrated pick position)

Two way: forward and backward

Four way: forward, backward, left, right

Max grip - you can set a limit to how many boxes you allow Pally to pick in one cycle.

Min grip quality - Based on the actual product and gripper dimensions, payload weight, and the number of boxes being picked, Pally will estimate a reference value called 'grip quality,' which is used by the optimizer to choose between possible gripper rotations and reduce the number of boxes to be picked in one robot cycle. It will also reduce acceleration while handling products.

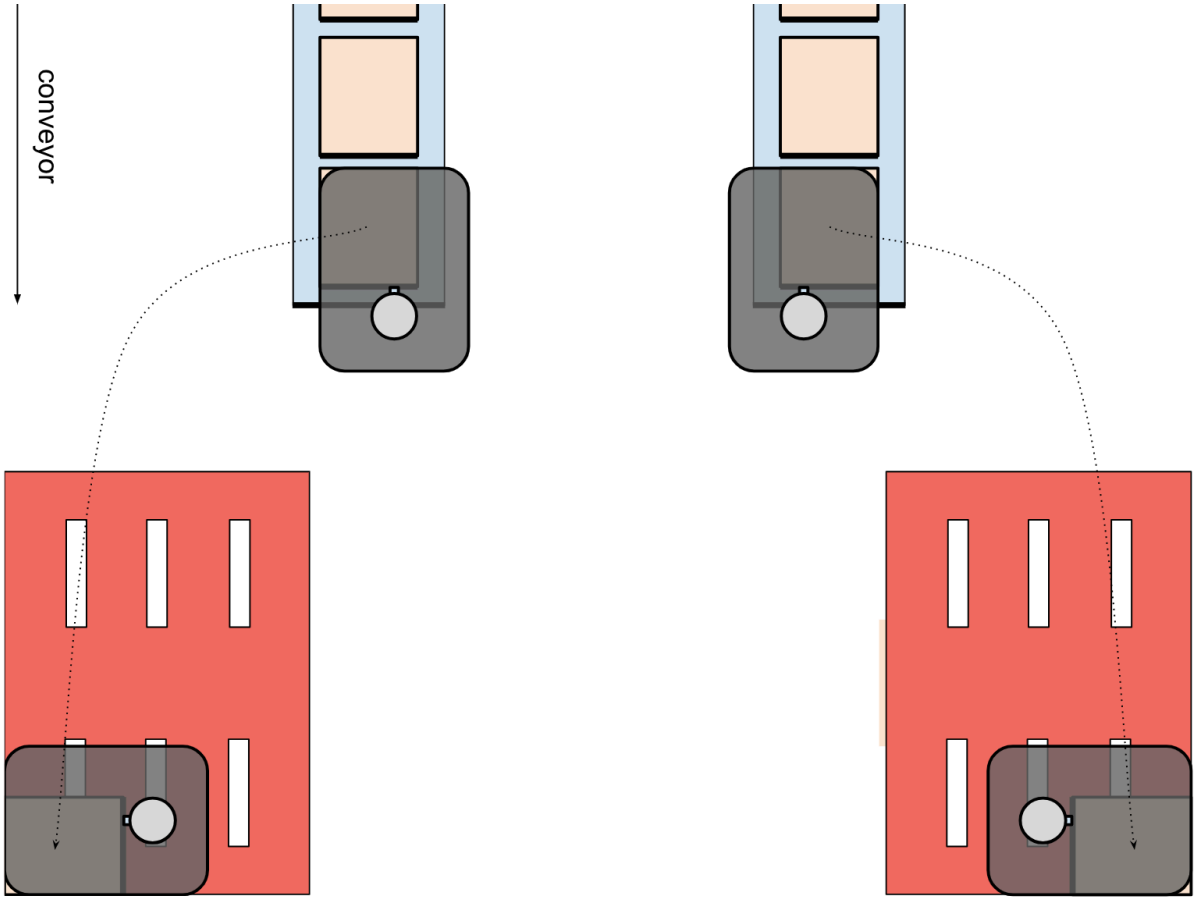
Grip quality in Pally is a numerical value that tells about the probability of losing boxes during transport. Usually, a higher grip quality implies a more reliable box transport between the pick and drop position. In comparison, a lower grip quality often means the gripper won't be able to hold the box. It is a floating value from 0 and up, default set to 0,3

Foam height - Gripper foam (suction cup) squeezing length in meters. Default value 0.02.

Length coverage - Minimum required fraction of the total length of each box covered by the gripper in multi-pick

Grip rotation cost - Additional cost of each specific gripper rotation; front, back, left and right. Default: [0, 30, 10, 10]

Align to edges - To avoid accidental multi-picking, the front edge of the gripper is always aligned with the rear edge of the (last) product being picked. This is especially important when boxes are smaller than the gripper. In this case the gripper is not aligned centered above the box, which is a common misunderstanding when using Pally for the first time.



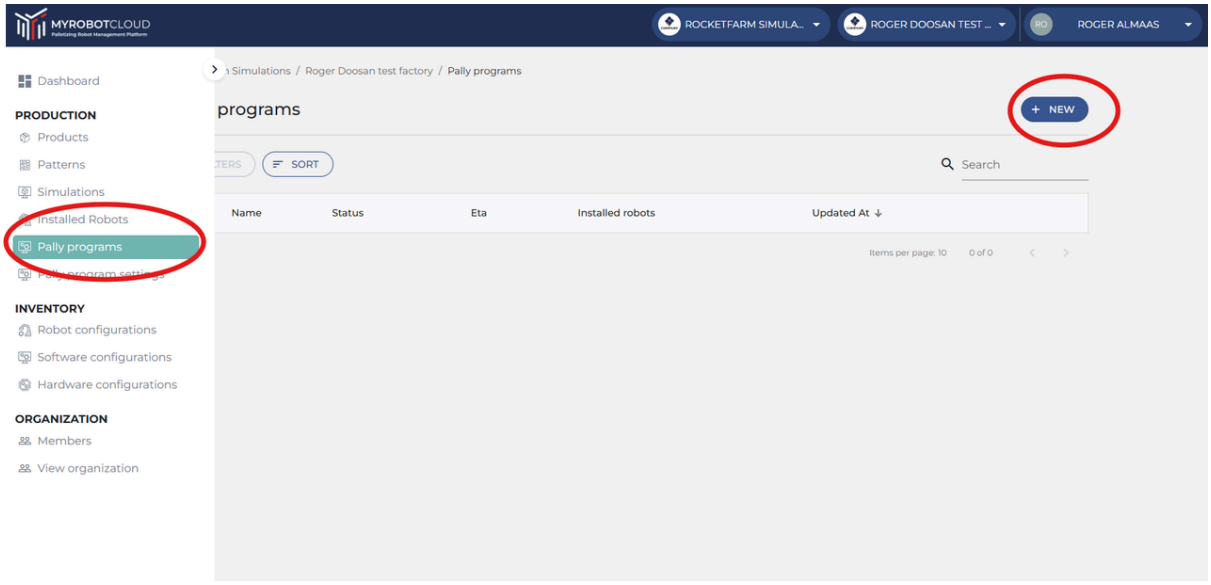
You can disable this, that will enforce the gripper centered above the box.

Once satisfied. Press → *Confirm*

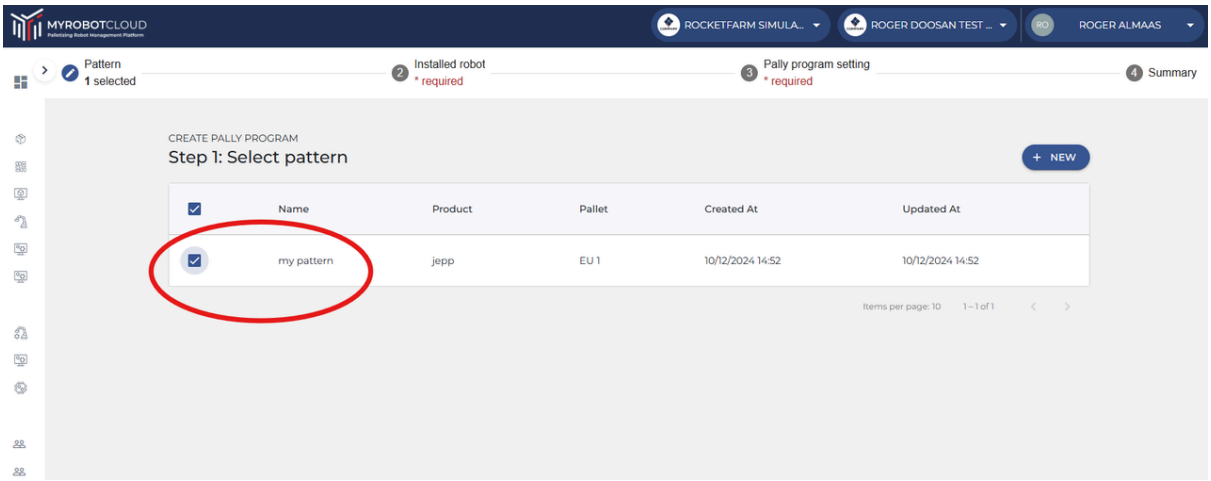
8.5 - Generate Pally Program

A Pally program is a json file. This json file contains all information the robot needs to palletize the pattern. It includes information about the product and the pallet. It includes all target positions and complete path planning with waypoints for each palletizing movement.

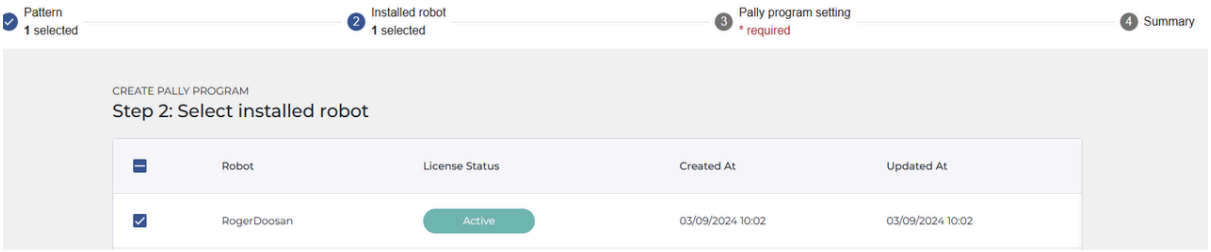
From the customer organization's dashboard go to Pally programs on the left menu.



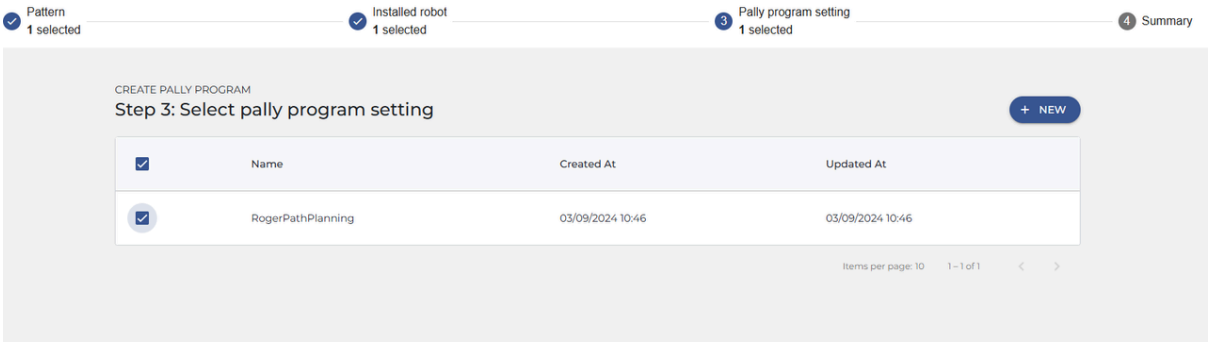
Select your pattern(s). You can create more than one program simultaneously



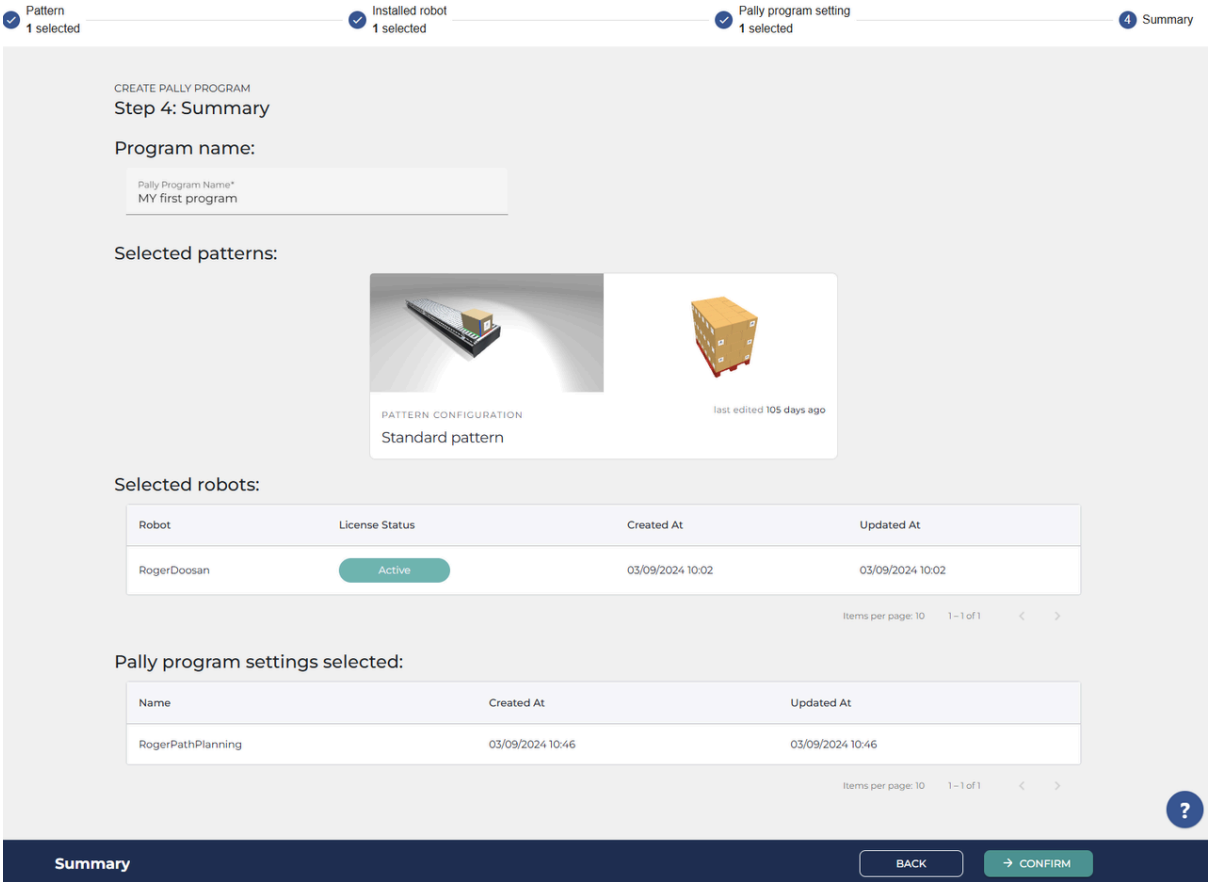
Select your Installed Robot



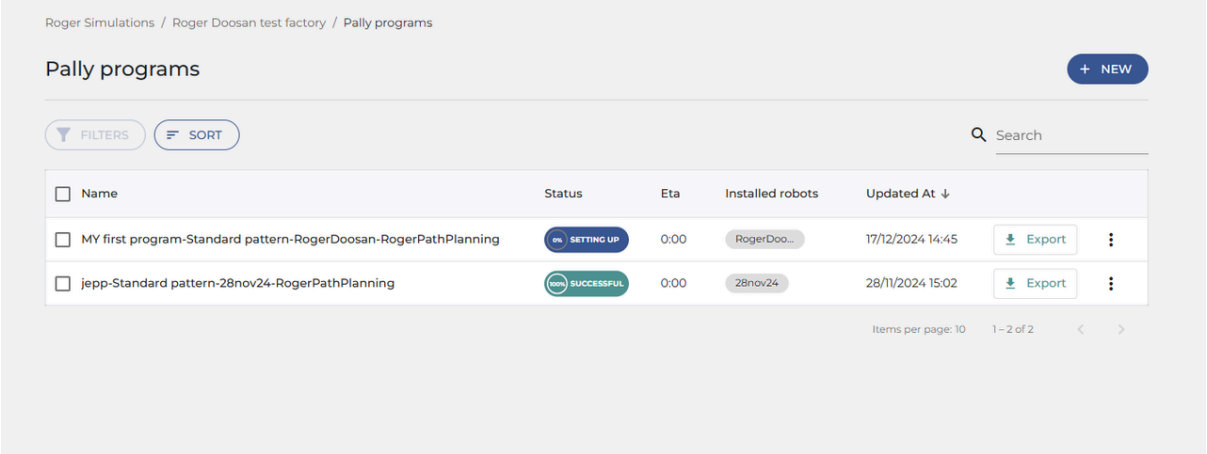
Select your Pally program setting



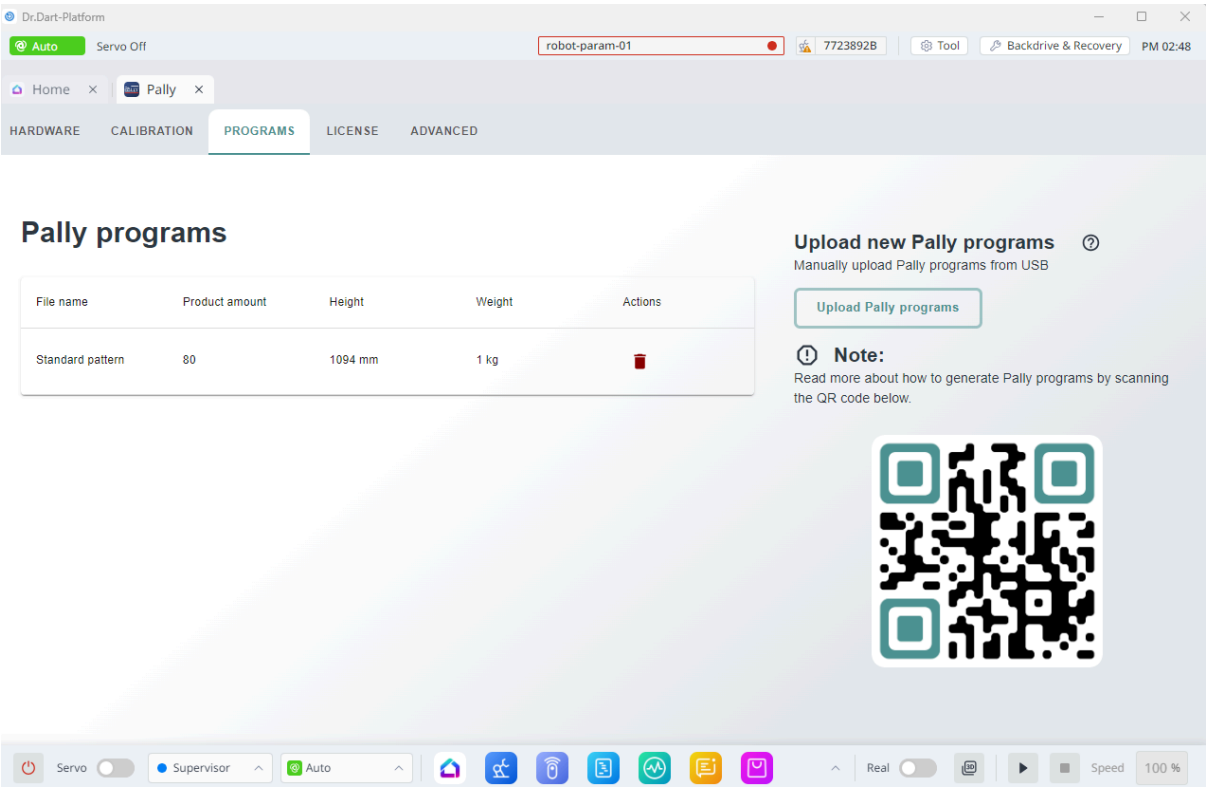
You will now get a summary and then you press Confirm



Once the waypoint generation is finished you can download the generated .json



Download the file to a USB drive and upload the file on the robot according to the Pally Manual **Upload Pally programs**

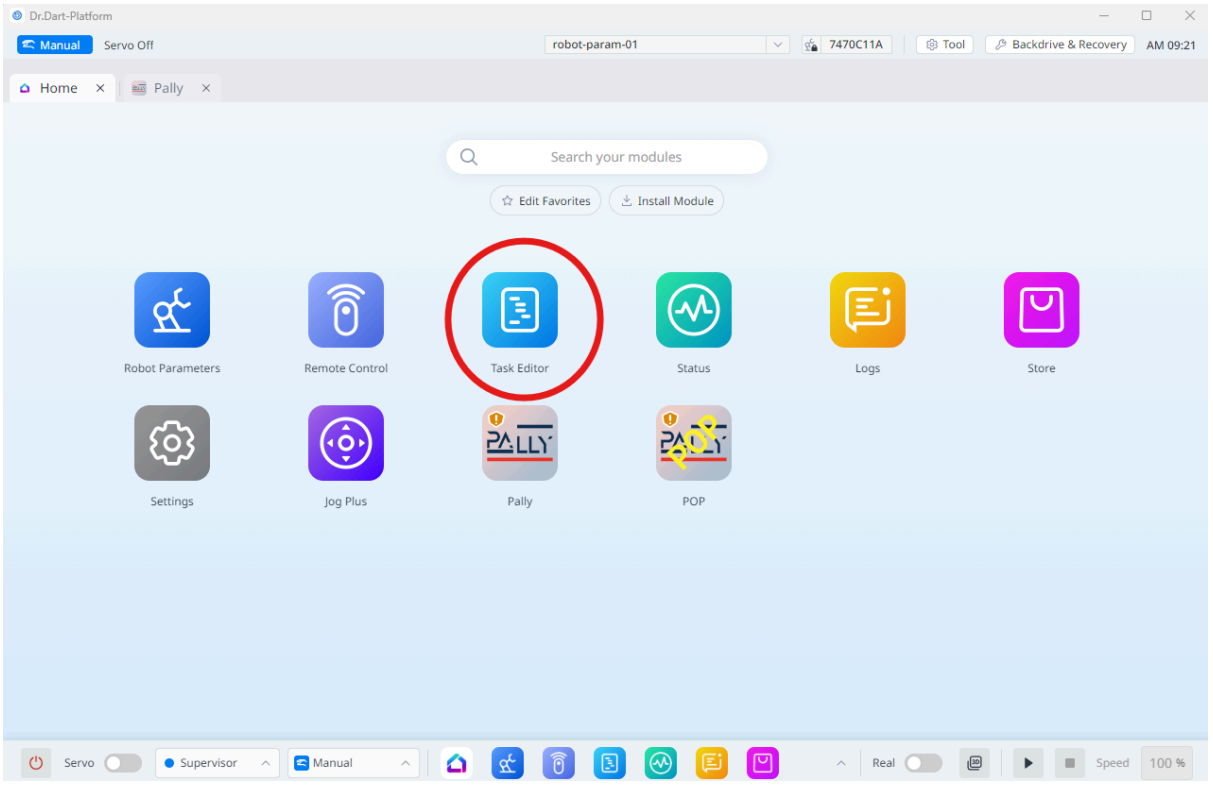


9 - Programming Pally

This chapter will give an introduction on how to use the program. It will describe how to create the main program, how to start the program, how to use the program in different modes, error recovery and shutdown of the robot.

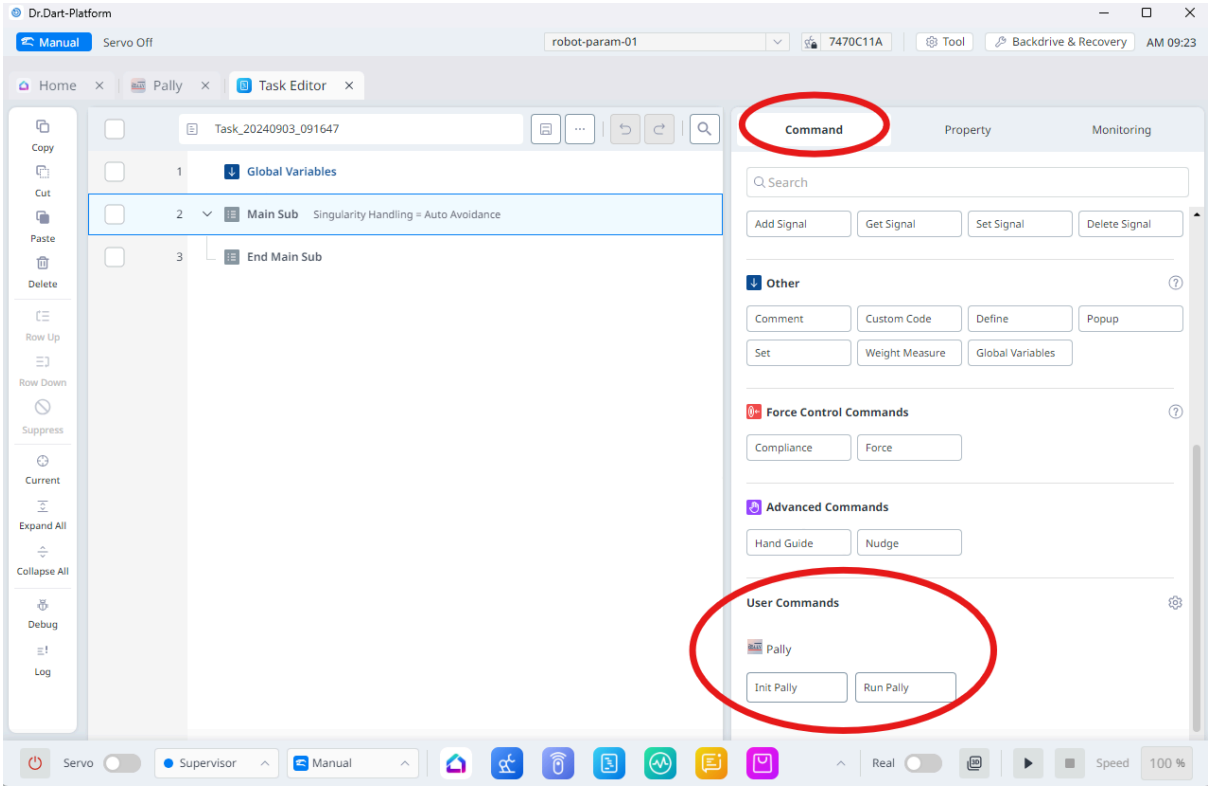
9.1 - Creating the main program

To make a palletizer program, you'll need to open the Task Editor.



From the Pally you have two different nodes:

- **Init Pally (optional)**
  - Get access to the Pally context (variables) and set up your callbacks
- **Run Pally (required)**
  - Configure the maximum speed and acceleration for Pally



9.2 - Init Pally program node

This node is optional. You can configure callbacks and get access to the Pally context via this node.

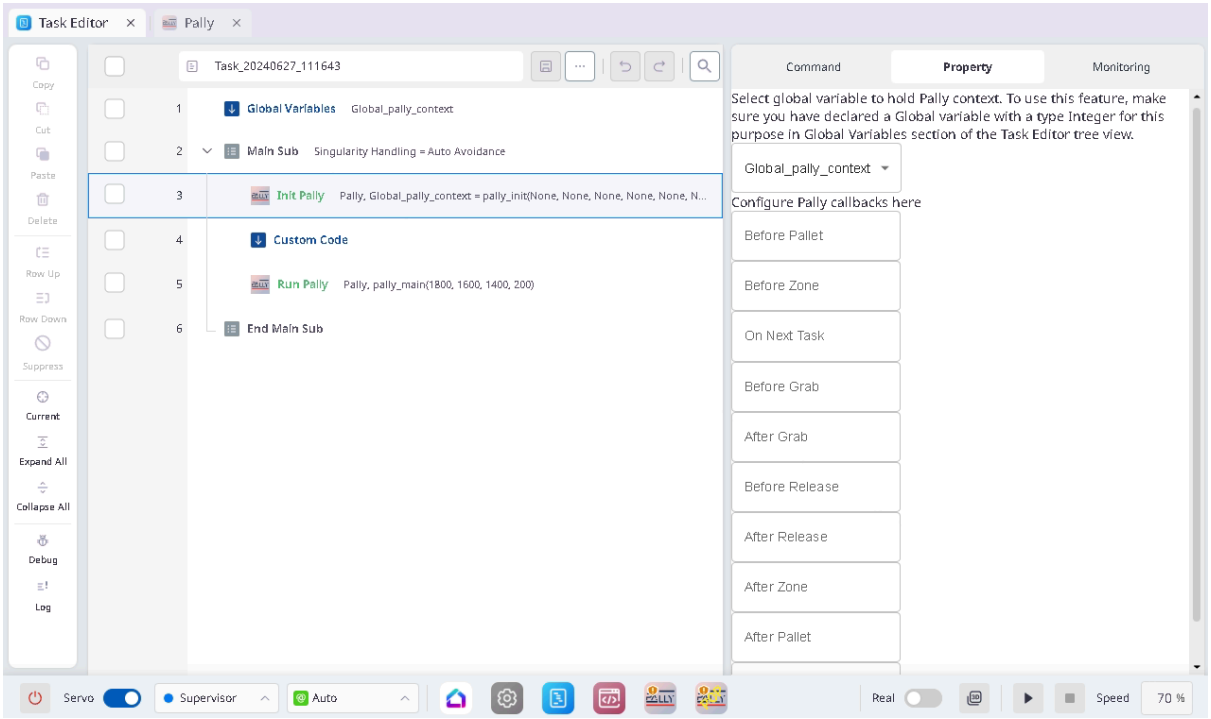
9.2.1 - Define Pally context global variable

This is required in order to access the Pally variables.

Define a global variable for Pally and initialize it to **0**. The examples here use Global_pally_context.

9.2.2 - Set up callbacks

Create your own subroutines that will be called at specific events during palletizing. Enter the name of each subroutine in the corresponding edit-box, or leave the edit box empty.



Configure Pally callbacks

Currently the following events are available:

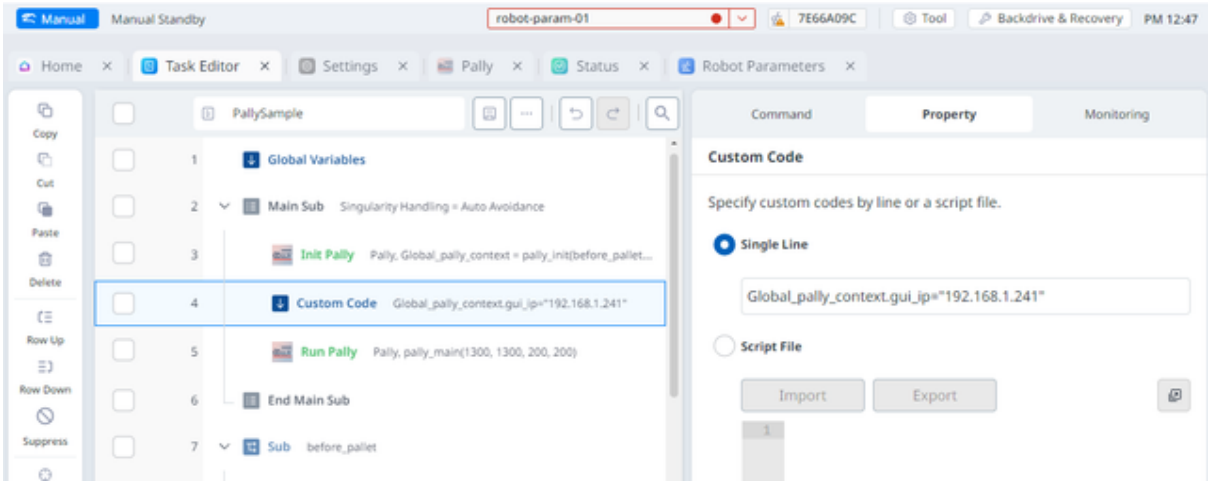
- **beforePallet**: Executed every time before a new pallet is started. This node is typically used to report the status to a production system.
- **beforeZone**: Executed before entering a new zone. Zones are mainly used with lifting columns.
- **onNextTask**: Executed before calculations for the next box position begin. The calculation parameters can be altered here. Set MovePerformed=True here in order to use a custom path towards the pick position.
- **beforeGrab**: Executed when the robot is ready for pick up at the pickup position. Typical uses for this node is to turn on a custom gripper or stop the conveyor. Commands that move the robot to the pick position should be inserted here when a custom path is in use.
- **afterGrab**: Executed when the robot has lifted up the box from the pickup position. Commands that move the robot from the pick position to the target position should be inserted here when a custom path is in use.
- **beforeRelease**: Executed when the robot is ready to drop the box on the pallet. Use this node to turn off a custom gripper.
- **afterRelease**: Executed when the robot has released the box. Use this node e.g. to report the progress to a production supervisor system. Commands that move the robot back from the target position should be inserted here when a custom path is in use.
- **afterZone**: Executed after leaving a completed. Zones are mainly used with lifting columns

- **onSheet**: Executed every time when a shim paper needs to be inserted. Commands that pick the shim paper and move it to the corresponding pallet should be inserted here.
- **afterPallet**: Executed when the robot has finished the pallet. Use this node e.g. to turn on a warning light, or show a popup message, so the operators can replace the pallet.

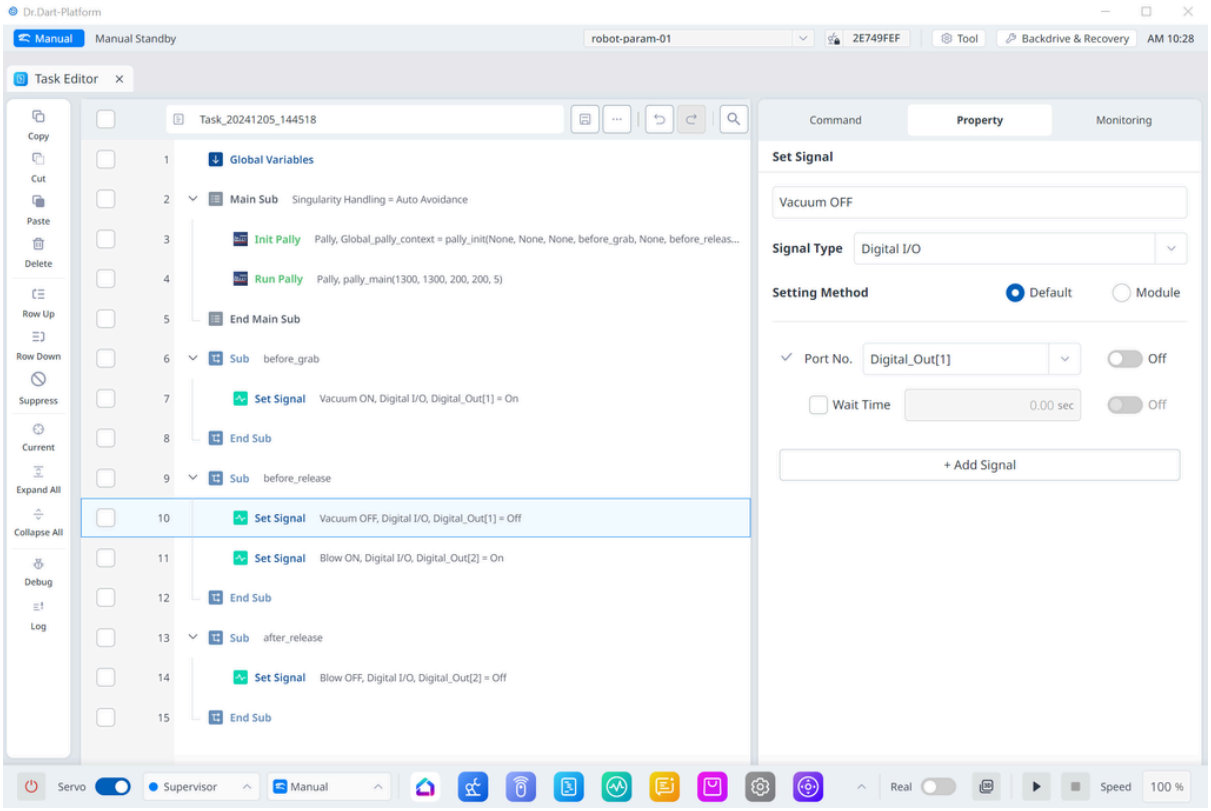
9.2.3 - Access the Pally context

The Init Pally program node will return a variable called Global_pally_context which points to an object with configuration data inside the Pally module.

You can use Global_pally_context anywhere in the callback subroutines or below the Init Pally program node in the main program to change any Pally settings in runtime.



Here is an example on how to add callbacks

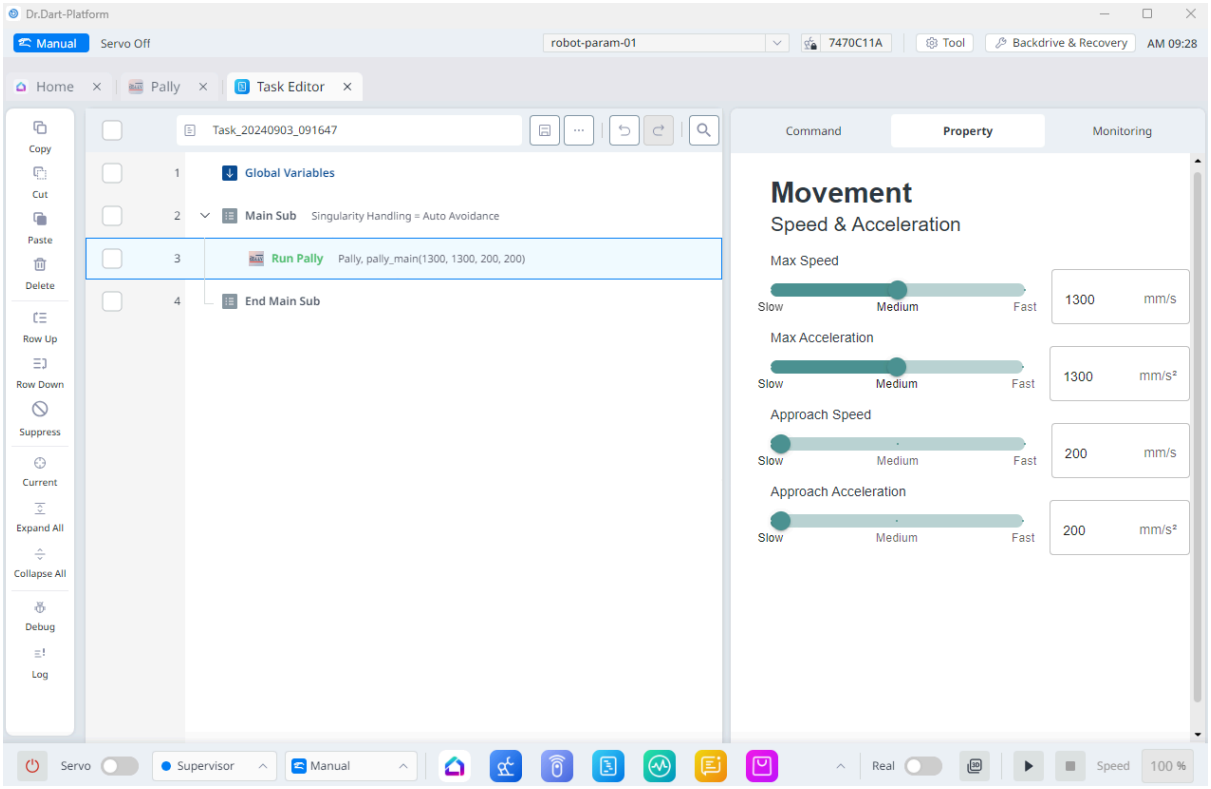


9.3 - Run Pally program node

This is the main program node that will control the robot and perform palletizing tasks.

The main program includes the product selection via operator input, and palletizing whole pallets.

Currently you can set the speed and acceleration along the path and at the approach position on this screen.



Maximum speed and maximum acceleration

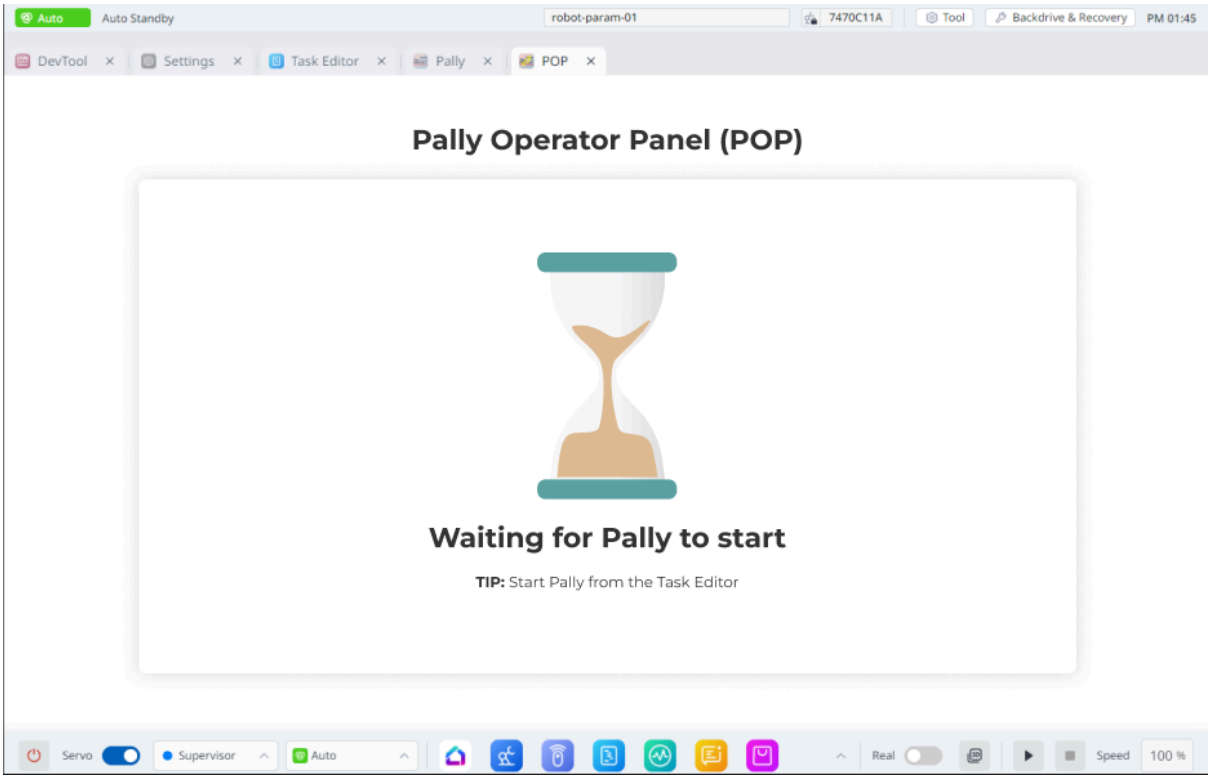
These parameters are used to limit the robot movement in general.

Approach speed and approach acceleration
These parameters are used to limit the robot movement when approaching the pallet position. This is especially important when the boxes on the pallet are very close to each other or the box dimensions have large variations, and quick movements would increase the occurrence of unexpected safety stops.

Once you have completed programming Pally, press the play button on the toolbar.

9.4 Pally Operator Panel

From the Main Menu, select the POP icon to open the Pally Operator Panel.

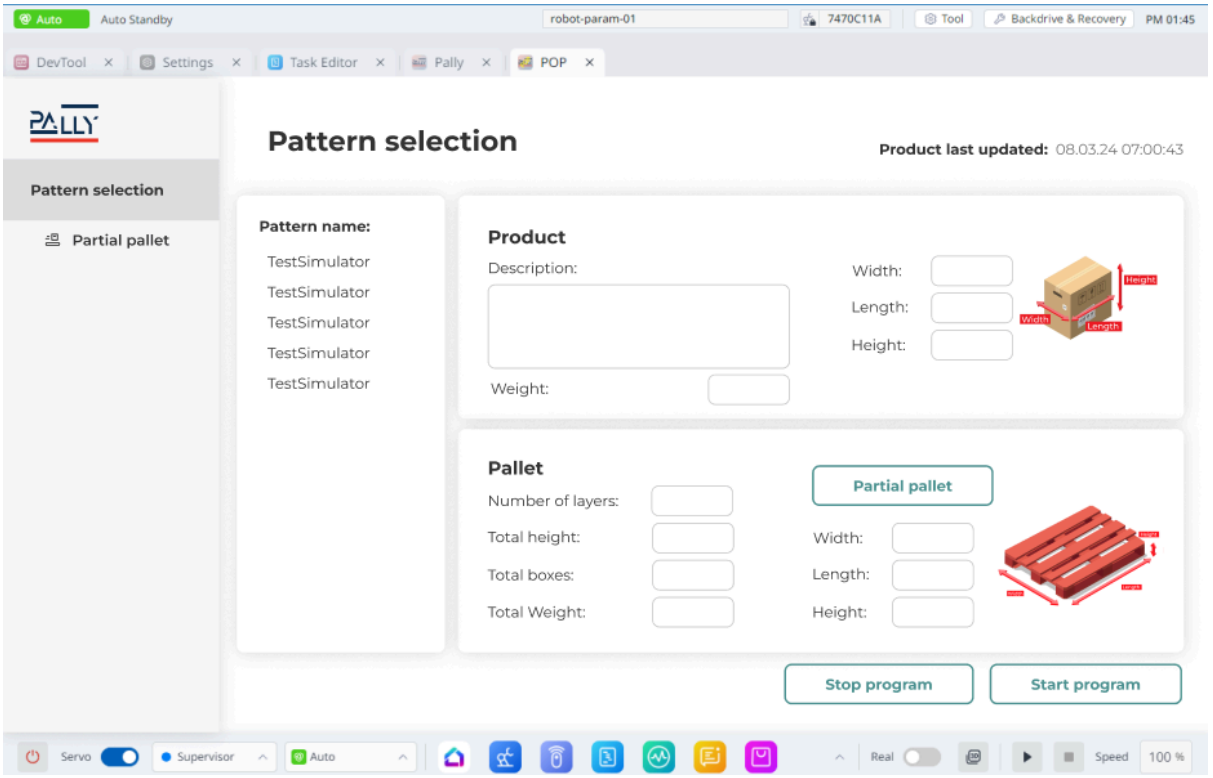


This is the Idle screen, where it waits for Pally to start.

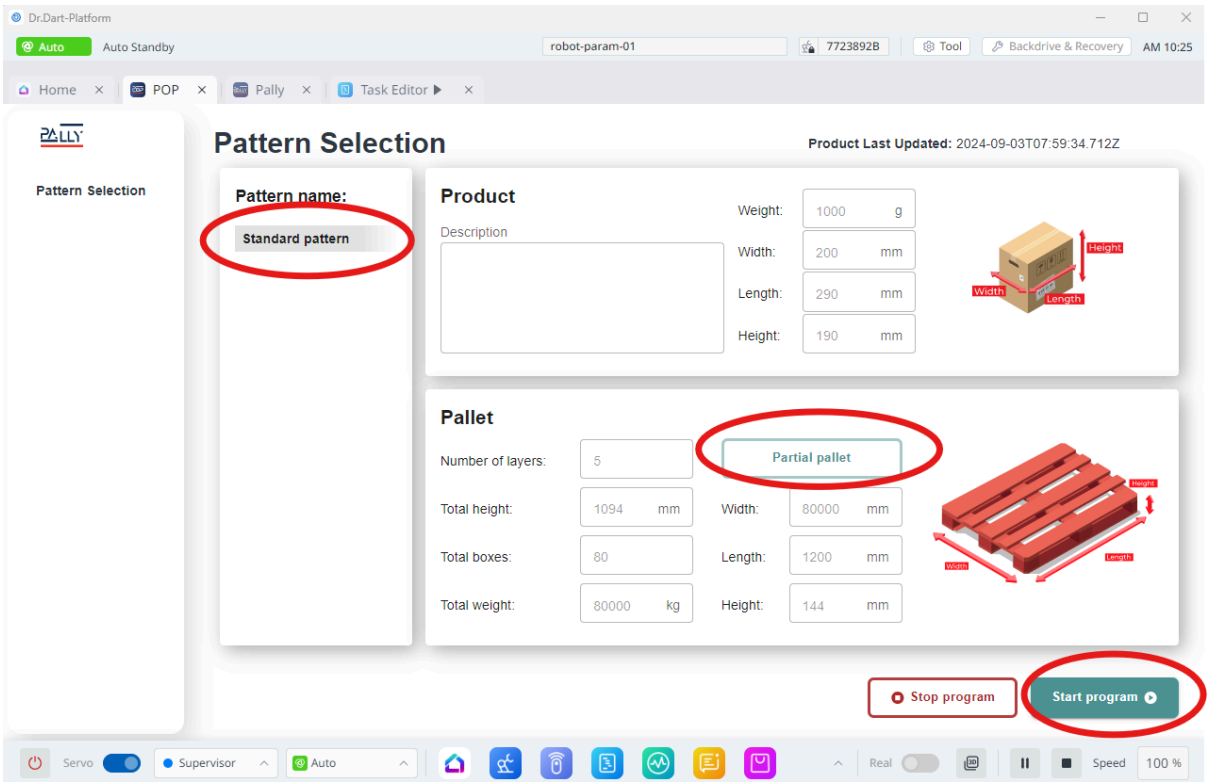
You can start the Pally program by pressing the Pally button in the footer to run the Pally palletizer program.

9.4.1 Pattern selection

When the program in Task Editor has started, the Product Selection screen appears.

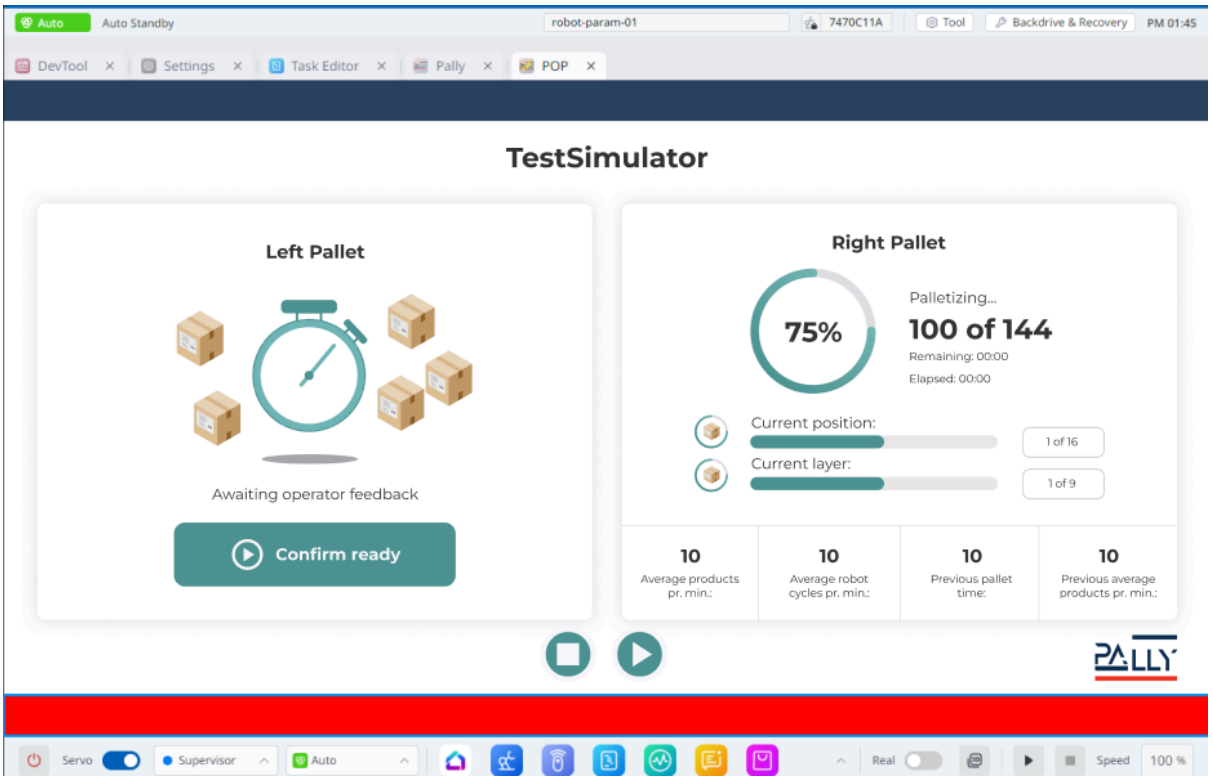


Choose your pattern and start the program or go to partial pallet



Confirm pallet

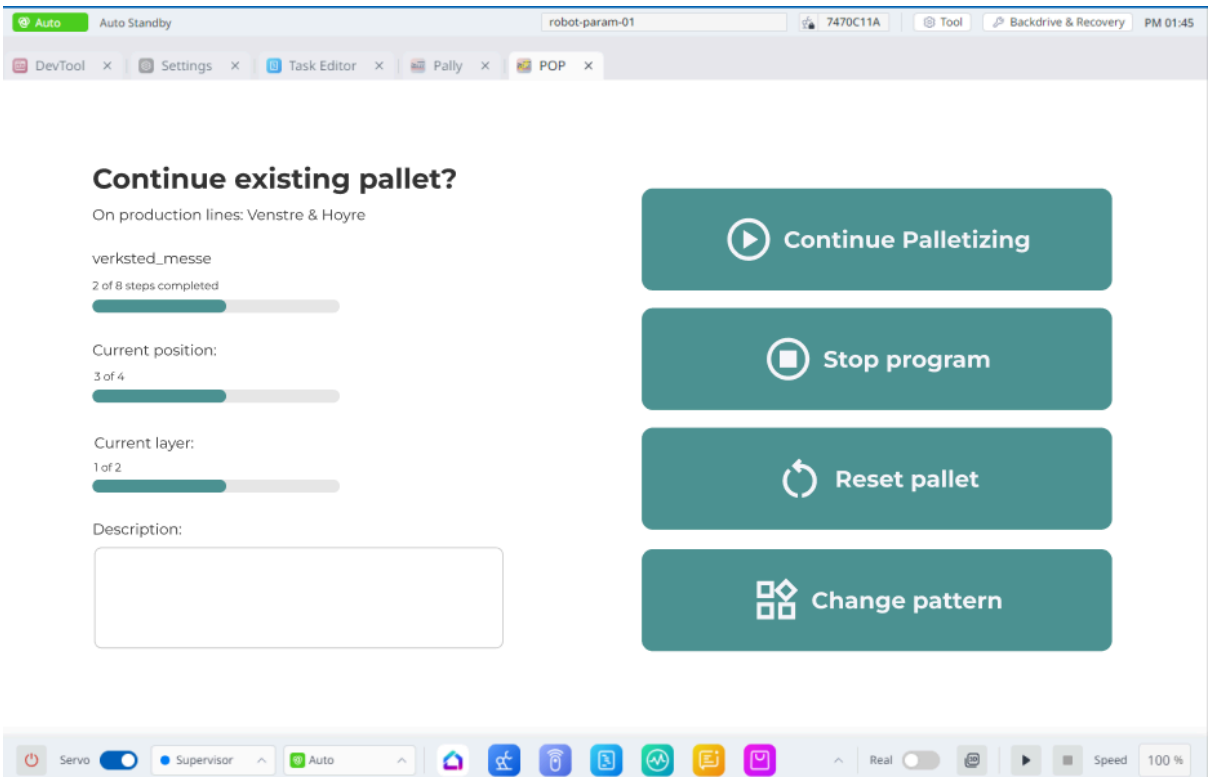
After selecting a pattern, the Pallet Confirmation (aka Runtime UI) appears. This screen is visible during palletizing.



Press “Confirm ready” to start palletizing that pallet.

After stopping and starting the program in Task Editor, the “Continue existing pallet” screen appears.

Continue existing pallet



Press “Continue” to continue the pallet that was in progress before the program stopped.

Press “Reset” to start a new empty pallet with the same pattern that was selected before the program stopped.

Press “Change pattern” to select another pattern and start a new empty pallet.

Press “Stop program” to terminate the program.

9.4.2 - Partial Pallet

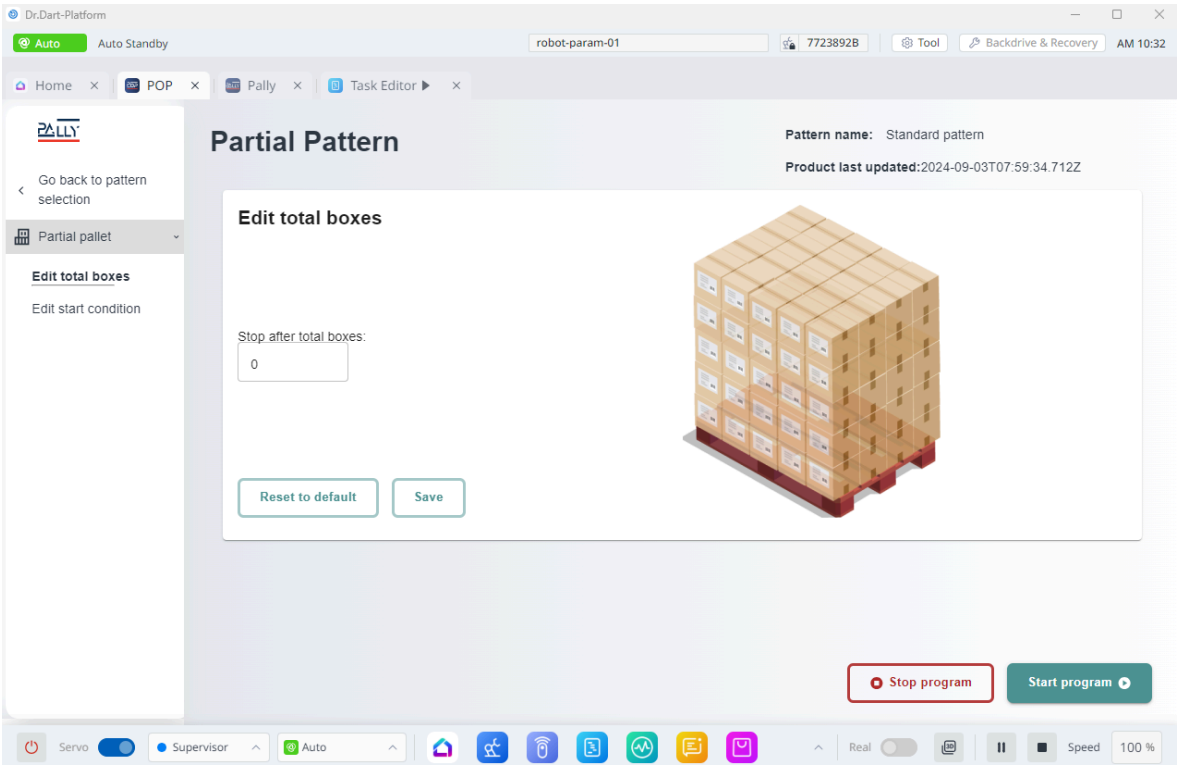
To alter the start conditions on a pallet or the stop conditions, use the partial pallet feature. Once a pattern is selected in the 'Pattern Selection'-dialog, a new dialog will open by clicking on the 'Partial Pallet' button.

In this dialog it is possible to edit two aspects of the selected pattern:

1. Edit the total boxes in the pattern
2. Edit the start conditions in the pattern
    a. Edit the start layer
    b. Edit the start box on the specified start layer

These two features are explained in more detail below.

9.4.2.1 Edit Total Boxes



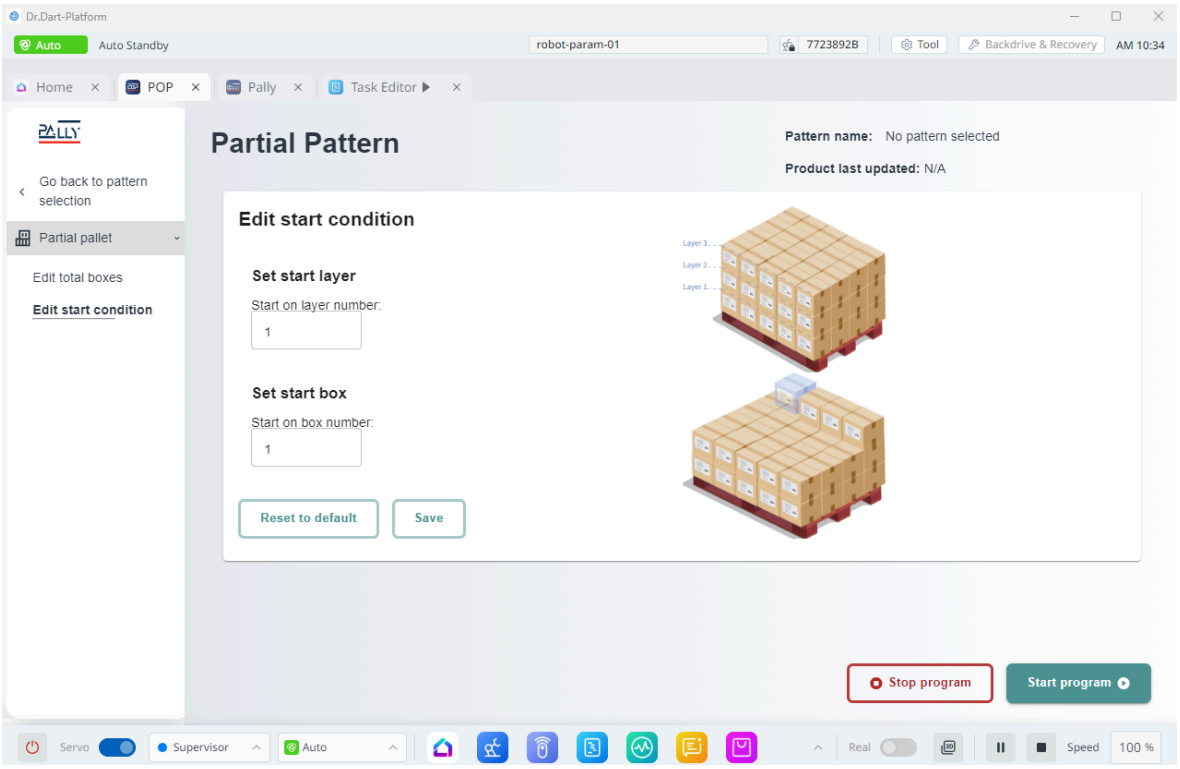
Edit the total boxes palletized on the first pallet

In order to palletize an incomplete pallet, type in the desired number of boxes in the input field under the ‘Stop after total boxes’-label. Make sure to input values in the range between 1 and the original amount of boxes in the selected pattern. Any total box number higher than the original amount of boxes will not be valid.

Note that the new value for the total boxes will only apply to the *first* pallet that is palletized, the remaining pallets will have the original amount of boxes in the selected pattern.

To reset the total boxes *and* the start conditions back to the default values for the selected pattern, press the ‘Reset to default’-button.

9.4.2.2 Edit Start Conditions



Edit the start conditions for the palletizer on the first pallet

In order to start palletizing from a different start condition than on the first box on the first layer, enter new values into the input fields below the ‘Start on box number’-label and the ‘Start on layer number’-label respectively.

The default values are 1 for both the new start layer and the new start box. Note that it is not possible to input values that are larger than the layer count for the selected pattern in the start layer input field. Be careful when entering a new start box, as there is no input validation on this

field yet. One must therefore make sure that the number inserted into the new start box field is less than or equal to the amount of boxes that are present in the specified start layer.

If the selected pattern contains shim paper, it is important to note that these shim paper layers do **not** count as a layer. Therefore, to specify a new start layer for a pattern with shim paper(s), disregard each shim paper and specify the starting layer according to the layers containing boxes.

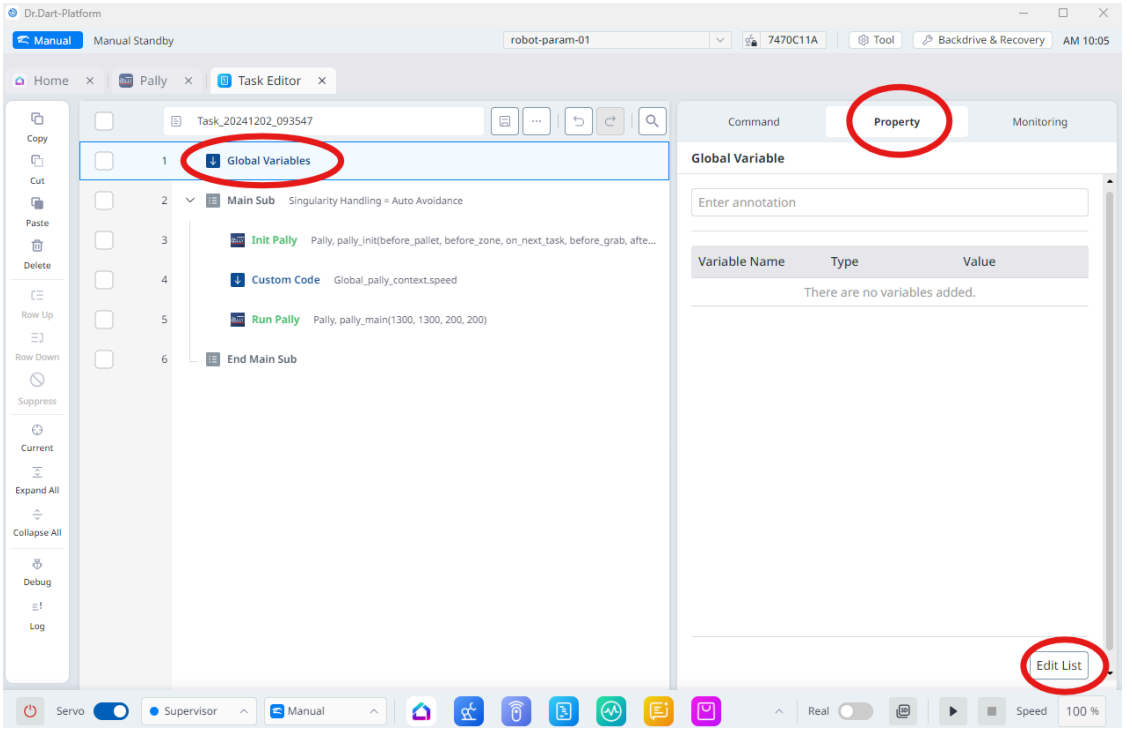
The new start box number signifies the box's number on the selected layer, such that if the chosen start layer is the second layer and one wishes to start on the third box on the second layer, then input the number '3' into the 'Start on box number'-field.

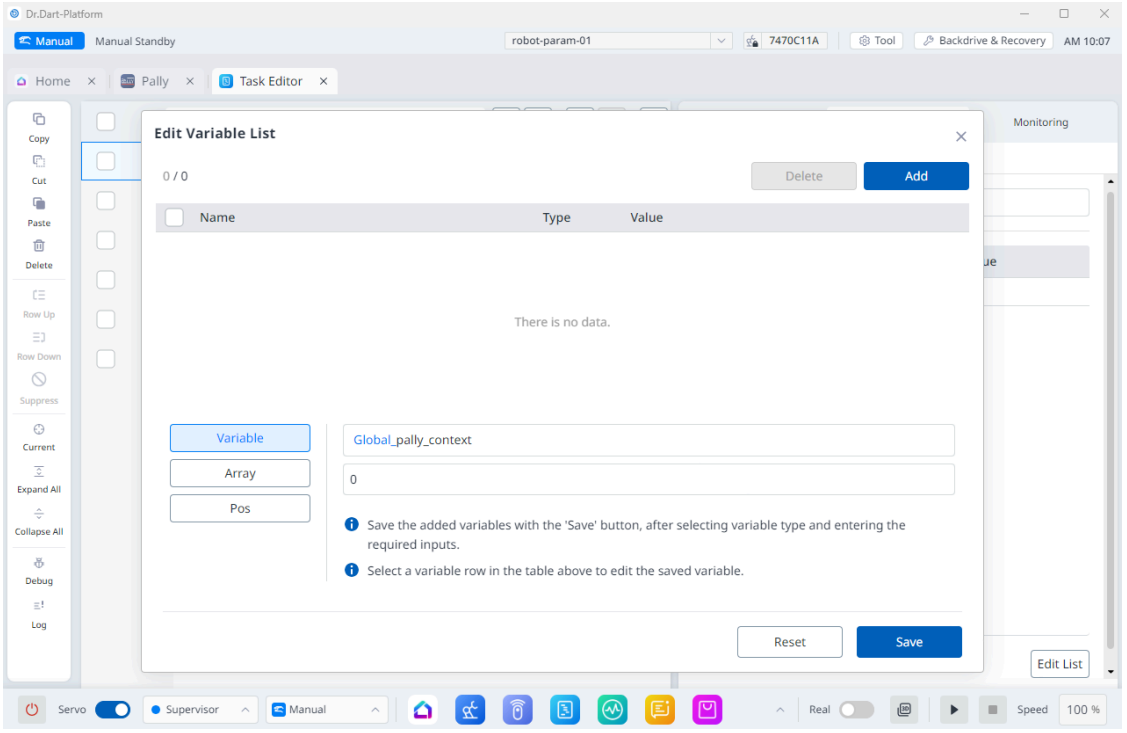
9.5 - Recovery from error

When the program stops and must be started again, remove any boxes from the gripper and turn off the vacuum. Use the freedrive button to move the robot into a position that is close enough to the default wait position, so the robot can start without collision. It may be necessary to set the empty gripper weight as current payload manually before activating the robot arm.

10 - Advanced configuration variables

All settings on the Pally graphical configuration panels are translated into variables, which - along with some additional variables - determine the final behavior of the palletizer program. To get better control of the program, or change some settings in runtime, it may be necessary to alter one or more of these configuration variables.
To access the Pally variables you need to define a global variable for Pally and initialize it to **0.** The examples here use Global_pally_context.





10.1 - List of available configuration variables

The following tables contain a short explanation of the variables with their data source. Related variables are categorized into separate tables.

10.1.1 - Speed and acceleration

Name	Description	Reference
max_acceleration	Maximum acceleration	mm/s2
max_speed	Maximum tool speed	mm/s
precise_acceleration	Approach acceleration	mm/s2
precise_speed	Approach tool speed	mm/s
min_grip_quality	Minimum required grip quality for moving with full acceleration	Floating value from 0.01 to 20 where low value equals higher acceleration

10.1.2 - Gripper

Name	Description	Reference
gripper_tcp	Gripper tcp	
gripper_cog	Gripper cog	
rf_gripper_height	Gripper height	Installation / Gripper / Dimensions
gripper_weight	Gripper weight in kg	Installation / Gripper / Dimensions
grip_delay	Delay after turning on the gripper at pickup	Default: 0.2
rf_max_grip	Maximum number of boxes in multi-pick	Default: number of product-sensors installed
grip_release_delay	Delay after turning off the gripper at target	Default: 0.2

10.1.3 - Automation

Name	Description	Reference
product_selection_strategy	Method to select product on startup: 0: GUI 1: First 2: Predefined	Default: 0
product_selection_predefined	Name of the pattern file - without the file extension ".json" - when product selection strategy is 2. In Dual Product mode: two pattern names for the primary and secondary conveyor should be given in the following format: "pattern1;:;pattern2" (Use separator ";:;" between names)	

10.1.4 - Other/Special

Name	Description	Reference
force_palletizing_mode	When setting this variable to True the program will activate "Palletizing Mode". When False, the program will not change the current mode. In Palletizing Mode, the joint angle of the lower arm is fixed. This will avoid singularity issues but limit the degrees of freedom.	
release_strategy	Bitmask that selects one or more waypoints to be removed from the return path Bit 0: remove box free Bit 1: remove target position Bit 2: use vertical exit path Bit 3: remove approach point Bit 4: remove everything after smart-exit point	Default: 7 (remove box free, remove target position, use vertical exit)
trace_program_callbacks	Variable for debugging.	When set to True, the Task Editor will trace and highlight the current program line inside Pally Callbacks that is being executed. This can make the execution of callbacks run slower. When set to False, the program lines will not be highlighted.
blend	Blend radius in linear movements as a fraction of the distance between subsequent waypoints.	Default: 0.3 Recommended values: 0 - 0.3
gui_ip	Variable for Dart simulator, to set the IP address of the PC that runs the Dart Platform.	
MovePerformed	A *boolean* value that Pally uses to check if it should move the robot on the path	True - Pally will NOT move the robot on the path created by Pally.

	created by Pally.	False - Pally will move the robot on the default calculated path. This is the default value.
TaskCompleted	A *boolean* value that controls whether the main Pally program should proceed to the next box position on the pallet, or repeat the same box position again.	True - Box has reached the target position on the pallet, the program can proceed to next box. False - Box has not reached the target position, the program should repeat the same position. The default value is True unless lost vacuum is reported by the “vacuum lost” signal.
ZoneNr	Indicates which part of the pallet is being palletized (when using zones)	
LayerNr	An *integer* representing the current layer number that is being palletized. The value is zero-based.	Any integer between 0 and the (total number of layers - 1) for the current pattern
ProductCount	An *integer* that is equal to the number of boxes that have been palletized *so far*.	Any value between 0 and the total number of boxes present in the pattern
ProductName	A *string* variable that is equal to the name of the current pattern file that is being palletized.	
LayerAlt	The altitude, measured from the top of the empty pallet	
LayerHeight	Height of the layer currently being processed.	

PalletNr	An *integer* representing the current pallet, either left or right pallet, that is being palletized.	1 = Right pallet 2 = Left pallet
PalletCenter	The PalletCenter variable contains a pose calculated from the 3 corner calibration points and refers to the center point on the empty pallet.	
MoveTarget	A *pose* variable that contains the robot's target pose	
MountPosition	A *pose* variable that contains the new mounting position compared to the mounting position where Pally was calibrated.	
set_workpiece_weight	Used for workaround regarding product weight for early version. Not in use	
set_weight_workaround	Used for workaround regarding product weight for early version. Not in use	Default false