

Photoneo

Locator Studio + Conveyor belt

Integration Guide





Marker Pattern Origin

ΔMSD

Robot workspace Origin

Robot Base

Scanner

$T_y = 100\text{mm}$

$RZ = 32^\circ$

$T_x = 150\text{mm}$

$T_y = 100\text{mm}$

$RZ = 32^\circ$

$T_x = ?$

Target

ΔTSC

ΔL

ΔCTD

Conveyor belt direction

Target X = $T_x - \Delta MSD + TD$

Encoder

ΔMSD = Marker Space Displacement
 Target = $T[X,Y,Z,RX,RY,RZ]$
 $TD = \Delta TSC + \Delta L + \Delta CTD$

ΔTSC = trigger to scan completion [m]
 ΔL = localization [m]
 ΔCTD = Conveyor tracking distance(variable in time/speed) [m]

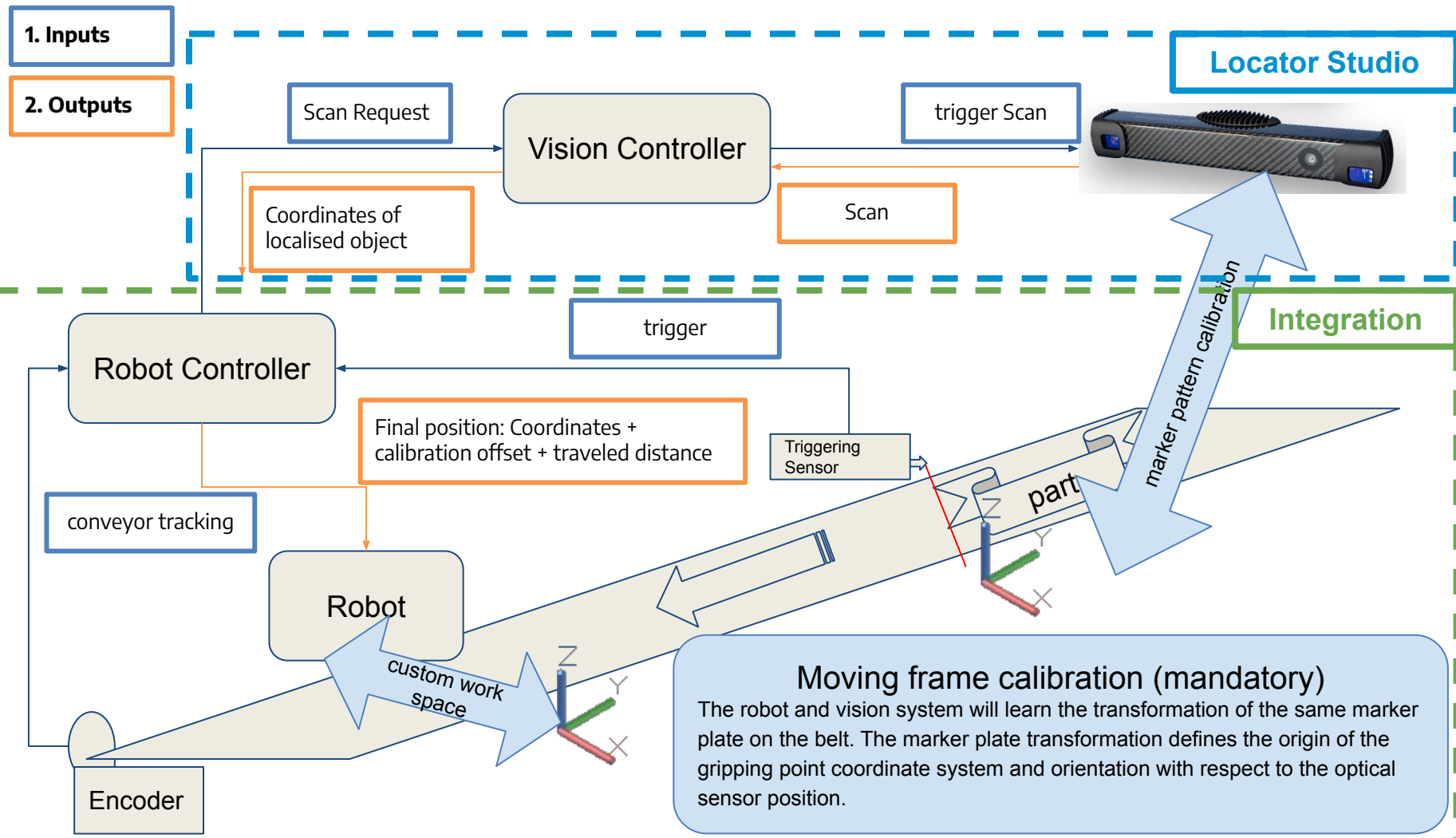
Pick pose calculation

1. Pick pose calculation components:

- Coordinates from localisation in Marker space (marker pattern calibration - compute transformation matrix from camera space to marker space)
- Traveled distance from acquisition to start of picking in conveyor tracking mode
 - possible triggers:
 1. SW - `pho_wait_for_req_completion()` (scan request)
 2. HW - HW trigger output
- Origin Custom Work space - taught in Robot
- Calibration distance - distance between position of marker pattern for camera calibration and position of marker pattern for custom work space calibration (linear transformation between Camera and custom WS origin)

2. Formula:

localised pose - calibration distance + traveled distance from scan



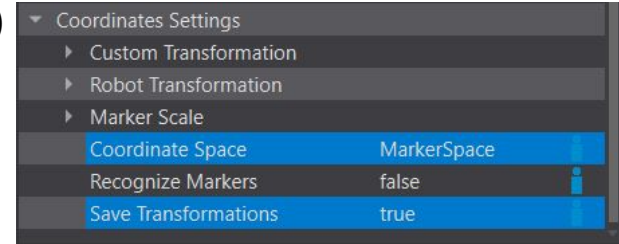
Calibration

1. Camera to Conveyor belt - common origin (Marker pattern calibration)

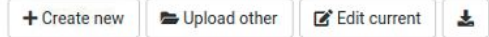
- Put calibration pattern respectively to conveyor belt
- Add triggering object for sensor in the origin of marker pattern
- Store calibration according marker pattern in PhoXi Control as marker space
- export phop and use it as custom scanning profile in VS

1. Robot to Conveyor belt - common origin

- move conveyor belt, without moving marker pattern according to conveyor belt, so marker pattern is in reach of robot
- calibrate custom coordinate space in robot
- track and save “calibration” encoder value
- synchronise robot with conveyor



Localization profile* ⓘ



Scanning profile type* ⓘ



Scanning profile* ⓘ

