PLATFORM PILOT

12. Jul 2023

DATA SHEET

The Neobotix PlatformPilot is an all-in-one software solution for the realization of your automation requirements. Serving as the pilot for the robot platform, it transforms mobile robots into Autonomous Mobile Robots (AMR) and navigates them efficiently and securely to their intended destinations.

In addition, it seamlessly integrates actuator control and sensor data processing, offering navigation and localization capabilities across diverse transport systems. Furthermore, it includes numerous additional modules designed to automate complex day-to-day industrial processes.

SYSTEM COMPONENTS

PLATFORM PILOT CORE	The control kernel, running on the AMR's on-board computer.
PLATFORM PILOT GUI	Cross-platform app for setup and built-in web interface for a quick and easy monitoring.
PLATFORM PILOT EDITOR	Intuitive and powerful programming tool, integrated into the Web-GUI.
PLATFORM PILOT HELP	Integrated help, online documentation, video tutorials, and more.
PLATFORM PILOT TOWER	Manages traffic of several robots in the same area.

HARDWARE REQUIREMENTS

	MINIMUM	RECOMMENDED
PlatformPilot Core/GUI	DualCore CPU, 2 GB RAM, 2 GB SSD	Intel i7 or similar, OpenCL or CUDA supported platform, 8 GB RAM, 8 GB SSD
Web-GUI	Any device with a modern internet browser	Any device with a modern internet browser

SOFTWARE REQUIREMENTS

PlatformPilot Core/GUI	Microsoft Windows 10 or newer, Ubuntu Linux 20.04 or newer
Web-GUI	Any OS with a modern internet browser

SUPPORTED ROBOTS

ROX & EMROX	Full support for all models
Neobotix Legacy AMR	Support for all standard models after 2016 (with RelayBoard V2.2) For older or customised robots please contact Neobotix
Self-built AMR	Customised versions and interfaces are possible, please contact Neobotix
3rd Party AMR	Please contact Neobotix
Kinematics	Differential drive (centred and three-wheeler), Omnidirectional (Mecanum, Neobotix OmniDriveModules, ArgoDrives)

DATA SHEET

MAPPING

Initial Mapping	Fully automatic SLAM (simultaneous localisation and mapping) with loop closure while robot is steered by remote control, manual map cleaning and defining of areas with easy graphical tools, approx. 10 minutes per 1.000 m^2
Map Updates	Manual erasing of outdated areas, automatic remapping equivalent to initial mapping
Map Elements	Landmarks, forbidden areas
Resolution	Adjustable (0,1 to 10 cm / pixel), recommended resolution is 3 cm / pixel
Maximum Size	Unlimited
Layer / Area Handling	Maps can be loaded automatically by command from script or application controller
Map Transfer	Maps can be shared by all robots with identical scanner height, e. g. all ROX & EMROX

Roadmaps

Roadmap Creation	Manually by graphical drag-and-drop tools
Roadmap Elements	Stations: Named destinations for automatic mode Nodes: Junctions between edges, used purely for curve and path definition Edges: Straight paths between stations and / or nodes, support many parameters
Path Parameters	Max. speed, orientation, max. deviation (distance and angle), and many more
Roadmap Handling	Any roadmap can be loaded with any map, e. g. for different tasks or days, last used roadmap is loaded automatically, new roadmaps can be loaded by command from script or application controller
Roadmap Transfer	Roadmaps can be shared by all robots, preferably by robots of same footprint

INTERFACES

VNX	Full access to all features, please see programmers manual for details
OPC UA	Supports both client and server operation, please see programmers manual for details
HTTP REST API	Provides access to almost any functionality of the PlatformPilot via the HTTP protocol.
JSON-RPC	JSON-RPC protocol version 2.0, please see programmers manual for details

APPLICATION PROGRAMMING

PlatformPilot-Editor	Blockly-based graphical "no-code" programming, provides access to all basic features and functions of the AMR plus all common loops, logic, variable handling etc.
Lua Scripts	Easy to learn and use script language, PlatformPilot-Core interpretes and executes scripts, please see here for details
ROS / ROS 2	Integrate PlatformPilot into a ROS(2) environment, such that it is possible to control the platform via ROS as well as visualize all data in RViz.
Robot Arms	The coordination of robot arms and other external devices with the mobile platform has to be implemented in an external script, e. g. Lua.

DATA SHEET

Sensors

Sick Laser Scanners	S300 Expert, microScan3 EFI-Pro, nanoScan3 EFI-Pro, other models on request
Other Laser Scanners	On request
Other	Intel RealSense 3D, Sick safeVisionary2

LOCALISATION

Average Accuracy	Up to ±15 mm (depending on sensor setup)
Environment	Any environment with static landmarks, moving through areas without landmarks detectable by laser scan is possible under certain conditions, please contact Neobotix for details
Initial Position	Last known position is automatically loaded at startup or set by command
Localisation Time	Initial localisation after startup or setting a new position needs approximately 30 seconds, during operation localisation is permanently updated with new data
Home Position	A virtual home station can be set up for quick position updates, e. g. after moving the robot while switched off

DOCKING

Average Accuracy	± 5 mm when using Neobotix docking landmark
Search Area	Adjustable, default is 1,4 m^2 in 2 m distance from target station
Average Docking Time	10 seconds
Predefined Functions	Automatic charging, dock to work station

TRAFFIC MANAGEMENT

Number of Robots	Unlimited
Number of Workspaces	One workspace per Tower process
Supported Maps	One map for all robots within the workspace
Supported Roadmaps	One roadmap for all robots within the workspace
Route Planning	Based on priorities of robots (fixed) and tasks (dynamic)
Collision Avoidance	Plans routes, pauses and detours to ensure smooth operation, can move idle robots aside
Deadlock Avoidance	Detects, resolves and prevents deadlocks