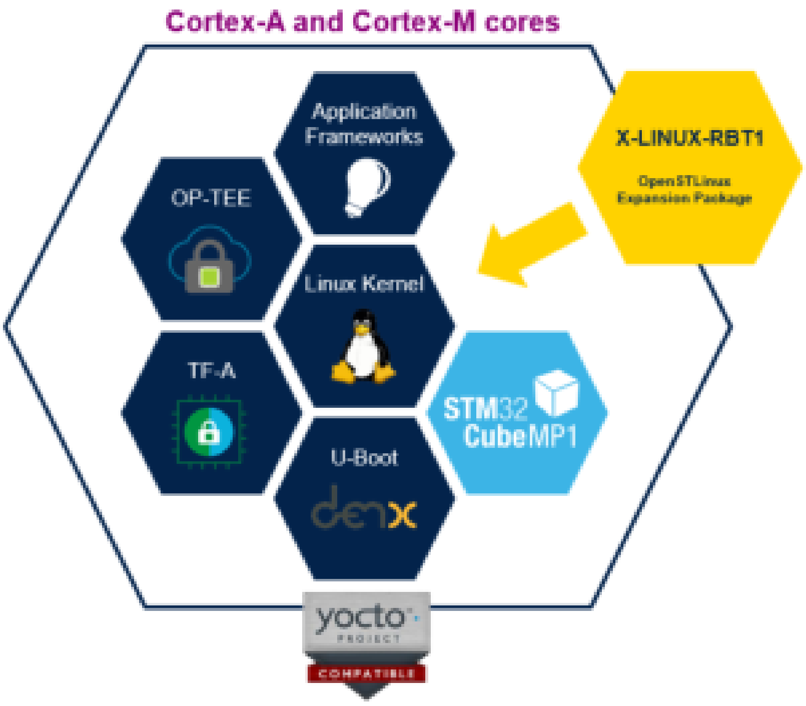
Getting started with X-LINUX-RBT1 MPU Software Package for X-STM32MP-RBT01 board

Introduction

The X-LINUX-RBT1 is a Linux-based expansion software package designed for robotics application development on STM32MP and other microprocessor platforms. It provides drivers, APIs, and applications tailored for the X-STM32MP-RBT01 board, which features the STSPIN948 motor driver. This package serves as a foundational tool for engineers to build complex robotics solutions.

Figure 1. **X-LINUX-RBT1 package**



1 Software features

The X-LINUX-RBT1 package includes a range of features for robotics application development:

1. Embedded web server with a web client for network based remote control.
2. Intuitive remote control web app with joystick interface.
3. Sensor fusion middleware for precise heading and orientation.
4. ToF-based (Time-of-Flight) obstacle detection.
5. Emergency stop functionality triggered by motor faults, user input, collisions, or topples.
6. Data logging capabilities for debugging and AI training.

2 X-LINUX-RBT1 architecture

The package is composed of multiple layers and modules.

2.1 Hardware drivers and APIs

- Kernel and device tree patches included in the package expose components like the ISM330DHCX (IMU), LPS22HH (pressure sensor), and IIS2MDC (magnetometer) via the Linux IIO subsystem.
- User-space Python drivers are provided for components like the STSPIN948 (motor driver) and VL53L5CX (ToF sensor), with low-level I2C, PWM, and GPIO configurations are handled via device tree patches.

2.2 Sensor algorithms

- Compute useful metrics from raw sensor data, such as altitude from pressure readings, distances from ToF sensor data, and orientation using sensor fusion.

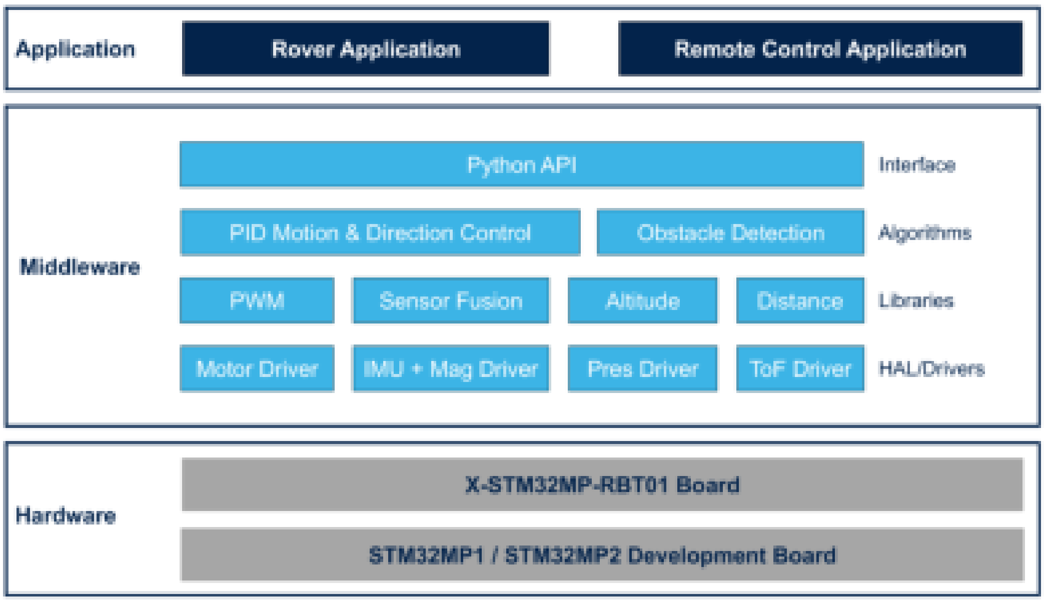
2.3 Robotics algorithms

- High-level algorithms tailored to robotics, including kinematics, obstacle detection, and path correction.

2.4 Applications

- Includes sample applications demonstrating practical use-cases, such as remote rover control, integrating all modules into cohesive robotics solutions.

Figure 2. **System components**

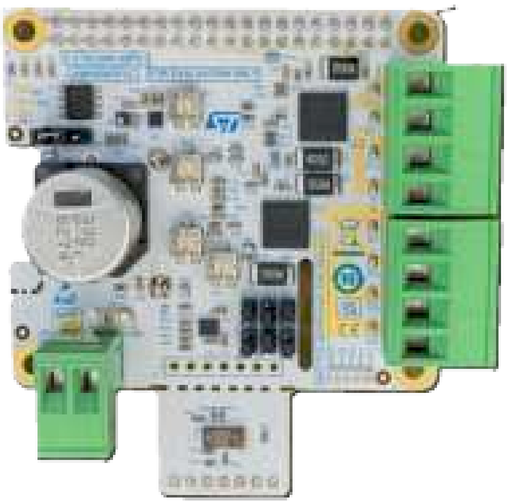


Engineers can develop custom applications leveraging the APIs and drivers provided in this package.

3 Hardware setup

The current package provides software support for the X-STM32MP-RBT01 expansion board.

Figure 3. X-STM32-RBT01 board



The key STMicroelectronics components available on this board are described below:

- STSPIN948: a 4.5 A dual full-bridge driver for brushed DC motors or bipolar stepper motors. Amplifiers for current sensing and adjustable slew-rate for EMI performance tweaking are other notable features.
- VL53L5CX: a state-of-the-art, Time-of-Flight (ToF) multizone ranging sensor.
- ISM330DHCX: The ISM330DHCX is a high-performance 3D digital accelerometer and gyroscope system-in-package designed for Industry 4.0 applications, offering superior stability, accuracy, and low noise. It includes embedded features such as a machine learning core, programmable FSM, FIFO, sensor hub, event decoding, and interrupt.
- LPS22HH: an ultra-compact piezoresistive absolute pressure sensor.
- IIS2MDC: a high-accuracy, ultra-low-power 3-axis digital magnetic sensor.

The X-STM32MP-RBT01 board can be plugged into the 40-pin connectors available on STM32MP discovery boards or Raspberry Pi, as shown below.

Figure 4. X-STM32MP-RBT01 with STM32MP157F-DK2

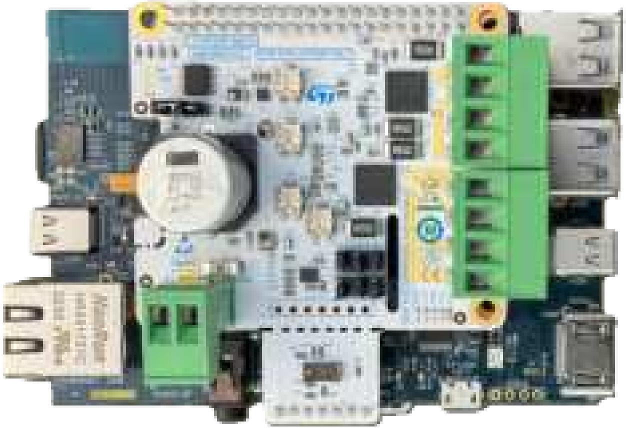
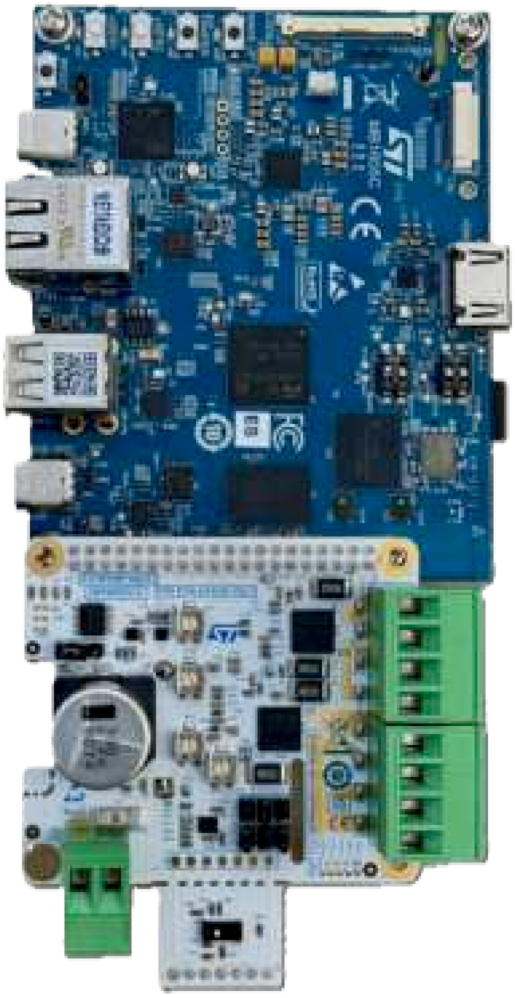


Figure 5. X-STM32MP-RBT01 with STM32MP257F-DK



3.1 Important setup notes

- Ensure correct board orientation when mounting on platforms like STM32MP Discovery kits or Raspberry Pi. For example, the board mounts “inward” on the STM32MP157F-DK2 and Raspberry Pi but “outward” on the STM32MP257F-DK board.
- Verify jumper settings for the STSPIN948 to operate in “Dual Independent Full Bridge Mode” as configured in the provided software. For other configurations, modify the motor driver code.
- Some GPIOs connected to the 40-pin headers are shared with other peripherals on STM32MP boards and may require the connection/disconnection of solder bridges. For example, in the STM32MP157F-DK2 board SB13, SB14, SB15, SB16 should be closed and SB01, SB02, SB03, and SB04 should be opened by desoldering the 0 Ω resistor. Refer to the specific board user manual for details.

3.2 Software setup

This section describes the software setup required for building, flashing, deploying, and running the application.

3.3 Recommended PC prerequisites

A Linux® PC running Ubuntu® 20.04 or higher is recommended. Developers can follow the link below for details: PC prerequisites.

Follow the instructions on the ST wiki page Image flashing to prepare a bootable SD card with the starter package.

Alternatively, a Windows or Mac computer can also be used; in that case, the following tools would be useful:

- STM32CubeProgrammer for flashing images.

- TeraTerm or PuTTY for console interface access.
- WinSCP for file transfer.

3.4 STM32MPU software prerequisites

The following Python packages are required for the X-LINUX-RBT1 software:

```
# Install required packages
```

apt-get install python3-gpiod

pip install smbus2 fastapi uvicorn websockets netifaces qrcode

3.5 Deploying the files to the MPU board

Transfer the binaries, Python scripts, and application resources to the STM32MP board from the development PC. Files can be transferred via a serial link, network connection, or external USB drive.

To connect to a WLAN, refer to How to Setup a WLAN Connection.

- For details on how to transfer the files over a network connection refer to How to Transfer a File Over a Network.
- For details on transfers using the serial link, for Linux hosts, refer to How to transfer a file over a serial console. For Windows hosts, refer to How to transfer files to Discovery kit using Tera Term.
- Alternatively, the user could transfer the files using an external USB drive.

To quickly evaluate the X-LINUX-RBT1 package, developers may copy the contents of the “application” folder contained in the package to the /usr/local/x-linux-rbt1 folder on the STM32MP board using any of the above methods. To ease this action, the deployment script present inside the “scripts” folder of the X-LINUX-RBT1 package can be used (if using a network connection to transfer the files).

```
# Go to the scripts folder
cd scripts
# Add execute permission to the deployment script
chmod +x deploy.sh
# Run the deployment script
./deploy.sh <MPU board IP>
```

4 Using the application

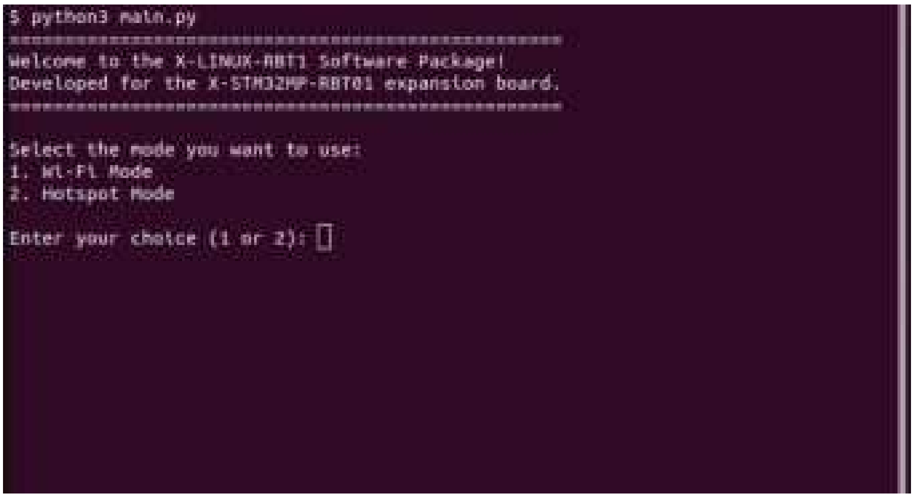
4.1 Launching the application

Once the files are deployed and the board is rebooted, the user can explore X-LINUX-RBT1, by accessing the terminal through ssh and running the application using the following command:

`python3 /usr/local/x-linux-rbt1/main.py`

This opens the command-line interface (CLI) of the application, where various network configuration options are displayed.

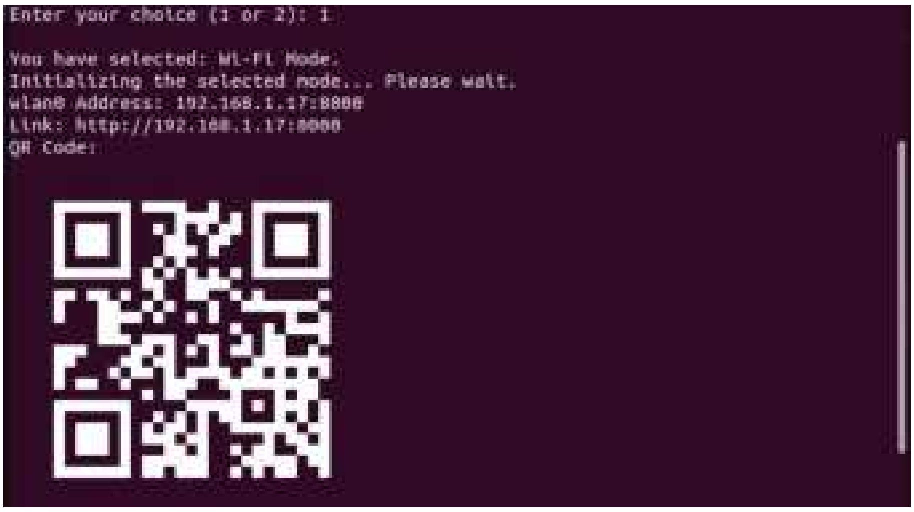
Figure 6. Application interface



```
$ python3 main.py
==================================================
Welcome to the X-LINUX-RBT1 Software Package!
Developed for the X-STM32MP-RBT01 expansion board.
==================================================

Select the mode you want to use:
1. Wi-Fi Mode
2. Hotspot Mode

Enter your choice (1 or 2): 
```

After initial configuration, a QR code is displayed which can be scanned on a mobile device to open the web-app to control the rover.

Figure 7. X-LINUX-RBT1 connection info screen



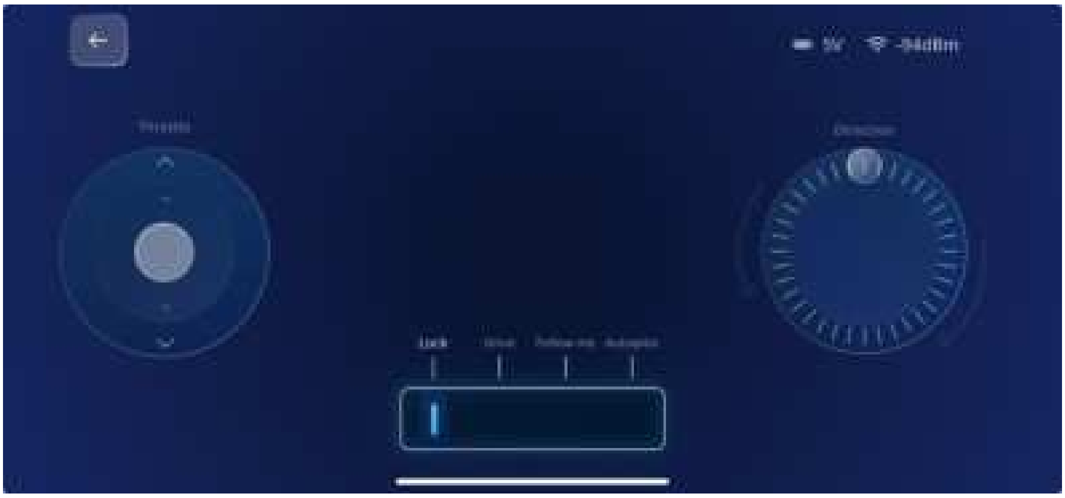
```
Enter your choice (1 or 2): 1

You have selected: Wi-Fi Mode.
Initializing the selected mode... Please wait.
wlan0 Address: 192.168.1.17:8000
Link: http://192.168.1.17:8000
QR Code:
```

After scanning the QR code on the mobile device, the Remote Control Interface opens (provided the device is connected to the same network as the board).

4.2 Remote control web app

To control the rover remotely, the **remote control web app** is hosted through the embedded web server module of the X-LINUX-RBT1 software. Once the application is running, a URL is provided for accessing the web app. The URL is also presented as a QR code to open the URL on a mobile device for user convenience.

Figure 8. **Remote interface**



4.2.1 Features of the remote control interface

- **Joystick-based control**:
- – **Left joystick**: Controls throttle for rover speed.
  - **Right joystick**:
    - The middle stick controls omni-directional movement when using mecanum wheels.
    - The outer dial adjusts rover heading or rotation.
- **Mode selection**:
  - **Remote control mode**: Manually control the rover.
  - **Autopilot mode**: Enables autonomous operations based on pre-configured algorithms.

5 Notes on compatibility

- Prebuilt binaries such as device tree blobs (DTBs) and kernel modules are platform-dependent and are provided for specific MPU boards. Customization may be necessary for other platforms.
- If kernel customization is needed, refer to How to Customize the Linux Kernel.
- On STM32MP boards, some GPIOs available on the 40-pin header are shared with other peripherals. To ensure exclusive access to the 40-pin header, certain solder bridges may need to be opened or closed. Refer to the user manual of the specific MPU board. Also, take a look at the board_pin_mapping.md file of the package for more information.

5.1 Compatibility information

The X-LINUX-RBT1 software package is validated for OpenSTLinux version 6.0. Running it on other ecosystem versions may require additional configuration. The software is tested on the following boards:

1. STM32MP257F-DK
2. STM32MP157F-DK2

6 Related information and documentation

Here are additional resources related to the X-LINUX-RBT1 package and its supported hardware:

- X-STM32MP-RBT01 Expansion Board
- STM32MP257F-DK Board
- STM32MP157F-DK2 Board
- STSPIN948 Motor Driver
- ISM330DHCX IMU
- LPS22HH Pressure Sensor
- VL53L5CX ToF Sensor
- IIS2MDC Magnetometer

Revision history

Table 1. Document revision history

Date	Revision	Changes
07-Apr-2025	1	Initial release.

Contents

List of figures