LILYGO T-Embed User Guide

Contents

LILYGO®

T-Embed

T-Embed User Guide

About This Guide

The T-Embed User Guide is intended to help users set up the basic software development environment for developing applications using hardware based on the T-Embed. It provides a simple example that illustrates how to use Arduino, including the menu-based configuration wizard, compiling the Arduino and firmware download to the ESP32-S3 module.

Release Notes

Date 2023.03
Version V1.0
Release notes First release.

Table of Contents

6. ip
7. reboot

Product Usage Instructions

Introduction

T-Embed:
The T-Embed is a hardware platform used for developing applications. This guide provides instructions on how to set up the development environment for the T-Embed.

Arduino:
Arduino is an open-source electronics platform based on easy-to-use hardware and software. This guide provides instructions on how to use Arduino for developing applications using the T-Embed.

Preparation:
Before getting started with the T-Embed, make sure you have the necessary hardware and software components. These include:

- T-Embed hardware platform
- Arduino IDE software
- ESP32-S3 module
- USB cable for connecting the T-Embed to your computer

Get Started:
To get started with the T-Embed, follow these steps:

1. Connect the T-Embed to your computer using the USB cable.
2. Download and install the Arduino IDE software.
3. Open the Arduino IDE and select the T-Embed board from the Tools menu.
4. Load the example sketch provided with the T-Embed and compile it.

Configure:
To configure the T-Embed, follow these steps:

1. Select the menu-based configuration wizard from the Tools menu in the Arduino IDE.
2. Follow the prompts to configure the T-Embed for your specific application.

Connect:
To connect the T-Embed to other devices, follow these steps:

1. Connect the T-Embed to the other device using the appropriate interface (e.g. Wi-Fi, Bluetooth, etc.).
2. Use the SSC command reference provided in this guide to establish a connection between the T-Embed and the other device.

Test Demo:
To test a demo application on the T-Embed, follow these steps:

1. Load the demo sketch provided with the T-Embed and compile it.
2. Upload the sketch to the T-Embed using the Arduino IDE.
3. Follow the instructions provided with the demo to test its functionality.

Upload Sketch

Build and Flash:
To build and flash a sketch to the T-Embed, follow these steps:

1. Load the sketch in the Arduino IDE.
2. Compile the sketch.
3. Upload the sketch to the ESP32-S3 module using the Arduino IDE.

Monitor:
To monitor the output of a sketch running on the T-Embed, follow these steps:

1. Connect to the T-Embed using a serial monitor (e.g. PuTTY, CoolTerm, etc.).
2. Configure the serial monitor to use the appropriate port and baud rate.
3. View the output of the sketch in the serial monitor.

SSC Command Reference:
The SSC command reference provides a list of commands that can be used to communicate with the T-Embed.

These commands include:

- op
- sta
- ap
- mac
- dhcp
- ip
- reboot

Refer to the T-Embed User Guide for detailed information on how to use these commands.

About This Guide

This document is intended to help users set up the basic software development environment for developing applications using hardware based on the T-Embed. Through a simple example, this document illustrates how to use Arduino, including the menu based configuration wizard, compiling the Arduino and firmware download to the ESP32-S3 module.

Release Notes
Date Version Release notes
2023.03 V1.0 First release.

Introduction

T-Embed

T-Embed is a development board. It can work independently.
It consists of ESP32-S3 MCU supporting Wi-Fi + BLE communication protocol and motherboard PCB. And equipped with 1.9 inch ST7789V IPS LCD and 7 RGB lights
At the core of this module is the ESP32S3R8 chip.
ESP32-S3 integrates Wi-Fi (2.4 GHz band) and Bluetooth 5.0 solutions on a single chip, along with dual high performance cores and many other versatile peripherals. Powered by 40 nm technology, ESP32-S3 provides a robust, highly integrated platform to meet the continuous demands for efficient power usage, compact design, security, high performance, and reliability.
Xinyuan provides the basic hardware and software resources that empowers application developers to build their ideas around the ESP32-S3 series hardware. The software development framework provided by Xinyuan is intended for rapidly developing Internet-of-Things (IoT) applications, with Wi-Fi, Bluetooth, flexible power management and other advanced system features.
The RF frequency range is BLE: 2402-2480MHz 2.4G Wi-Fi: 2412-2462MHz
The T-Embed manufacturer is Shenzhen Xin Yuan Electronic Technology Co., Ltd.

Arduino

A set of cross-platform applications written in Java. The Arduino Software IDE is derived from the Processing programming language and the integrated development environment of the Wiring program. Users can develop applications in Windows/Linux/MacOS based on Arduino. It is recommended to use Windows 10. Windows OS has been used as an example in this document for illustration purposes.

Preparation

To develop applications for ESP32-S3 you need:

- PC loaded with either Windows, Linux or Mac operating system
- Toolchain to build the Application for ESP32-S3
- Arduino that essentially contains API for ESP32-S3 and scripts to operate the Toolchain
- The ESP32-S3 board itself and a USB cable to connect it to the PC

E- Label

T- Embed is a development board. Use of electronic label. The display method is as follows:

1. Connect USB or lithium battery for power supply, and the program starts to run.
2. The screen will display Label first, and the duration is 3 seconds.

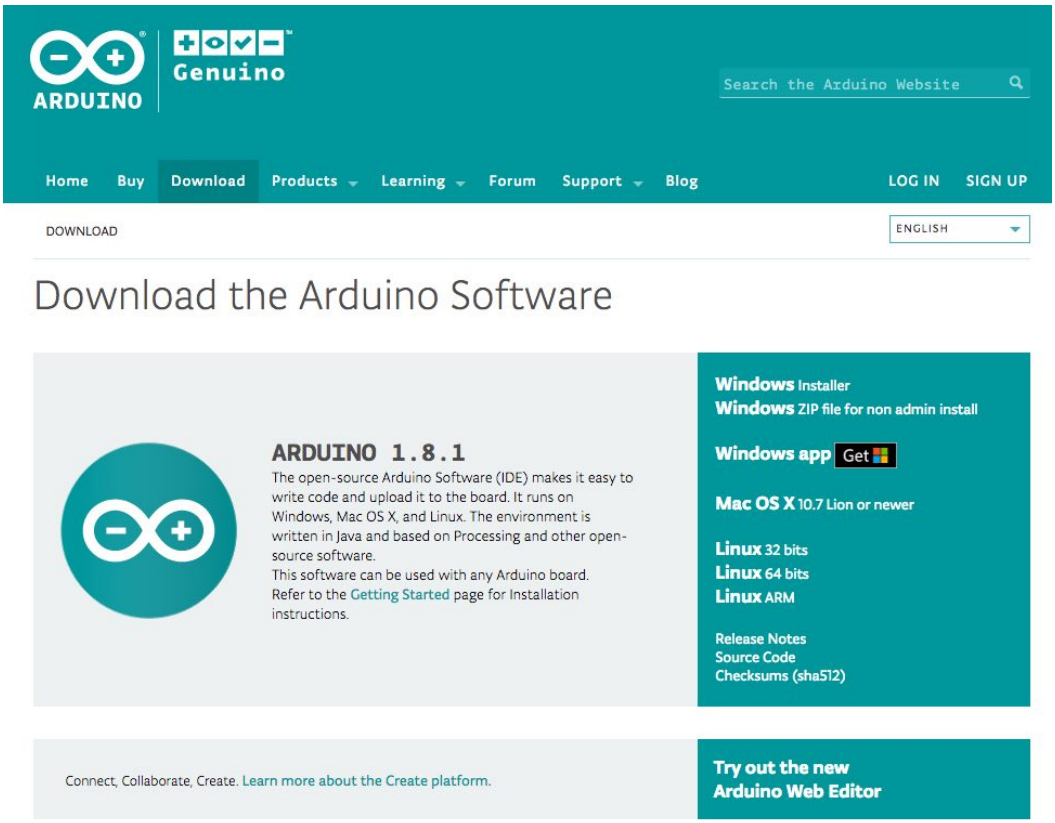
Get Started

Download the Arduino Software
The quickest how to install the Arduino Software (IDE) on Windows machines

Quick Start Guide
The website provides a quick start tutorial

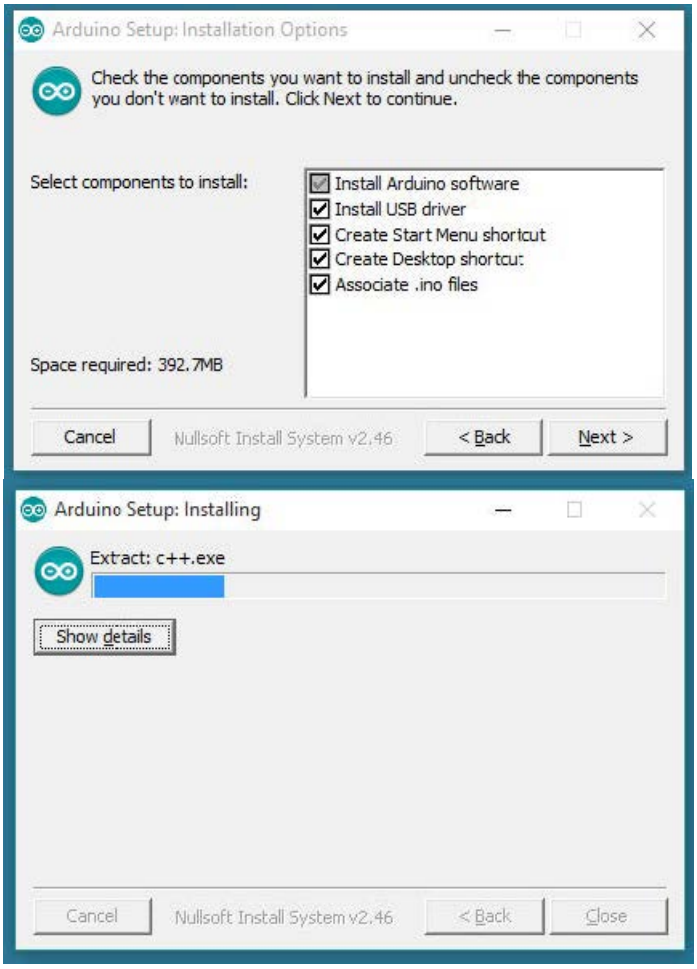
- **Windows:**
  https://www.arduino.cc/en/Guide/Windows
- **Linux:**
  https://www.arduino.cc/en/Guide/Linux
- **Mac OS X:**
  https://www.arduino.cc/en/Guide/MacOSX

Installation steps for Windows platform Arduino



Enter the download interface, select Windows installer to install directly

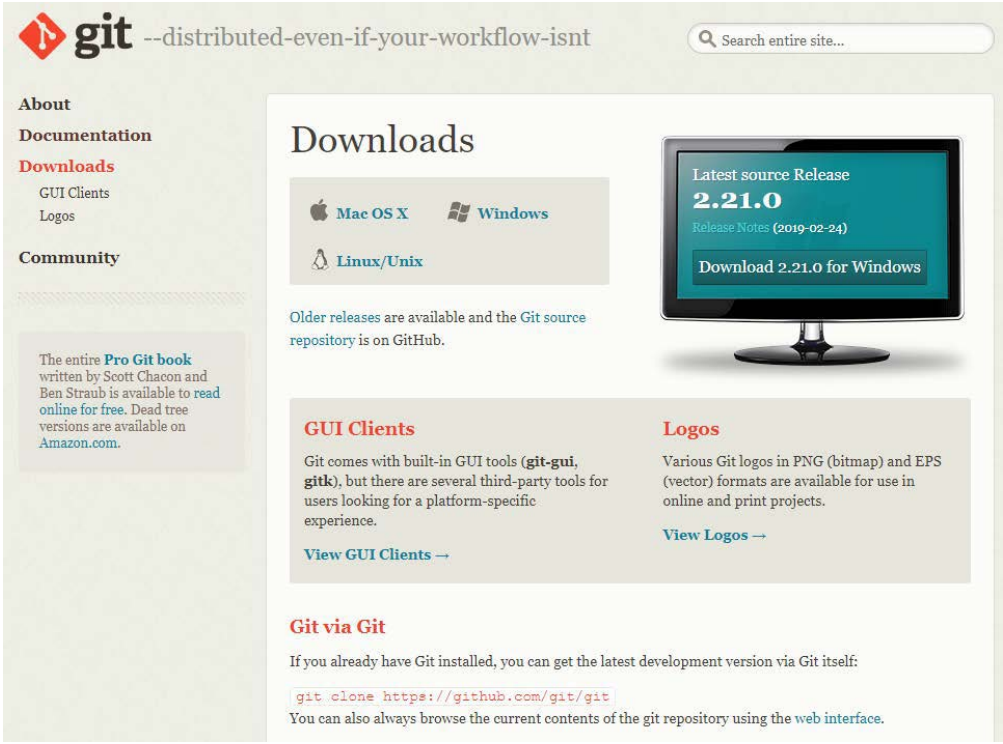
Install the Arduino Software



Wait for installation

Configure

Download Git
Download the installation package Git.exe



Pre-build configuration
Click Arduino icon, then right click and select “Open folder where “
Select hardware ->
Mouse ** Right click ** ->
Click Git Bash Here

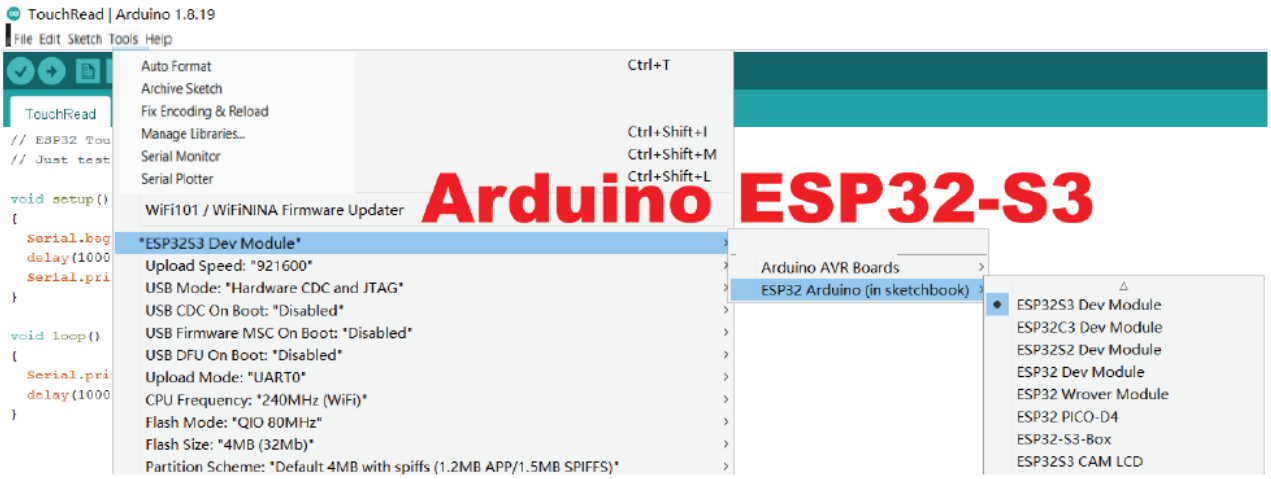
Cloning a remote repository

- mkdir espressif
- cd espressif
- git clone –recursive **https://github.com/espressif/arduino-esp32.git** esp32

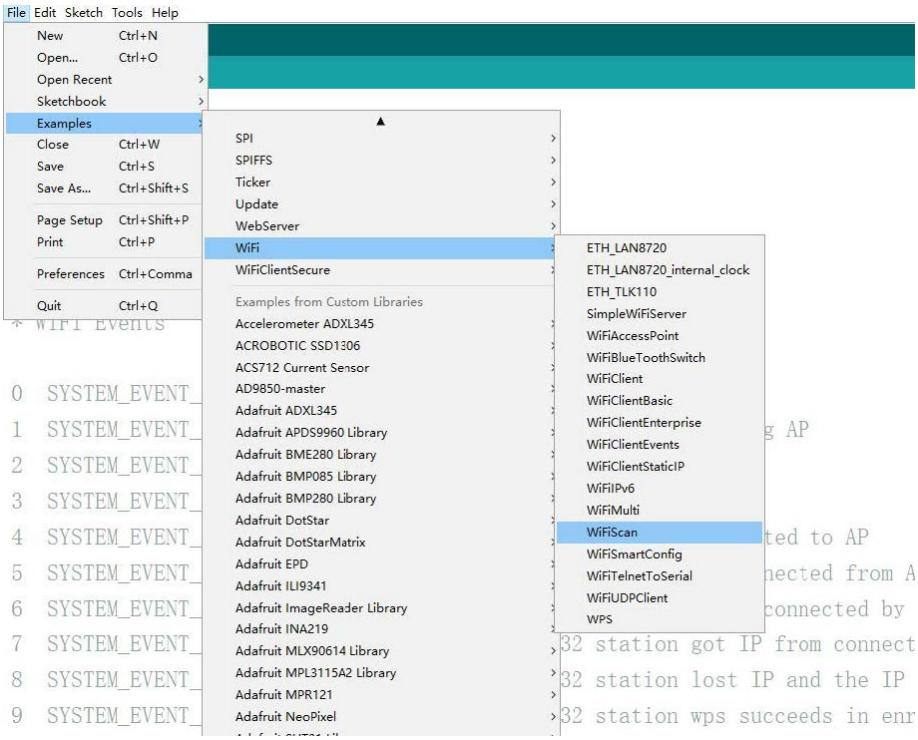
Connect
You are almost there. To be able to proceed further, connect ESP32-S3 board to PC, check under what serial port the board is visible and verify if serial communication works.

Test Demo
Select File->Example->WiFi->-WiFiScan

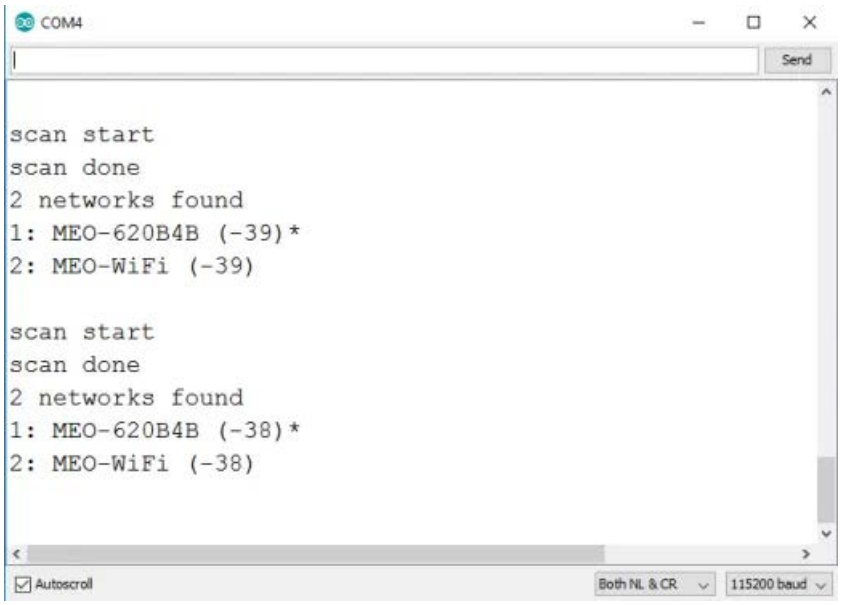
Upload Sketch
Select Board
Tools<<Board<<ESP32S3 Dev Module



Upload
Sketch << Upload



Serial Monitor
Tools << Serial Monitor



SSC Command Reference

Here lists some common Wi-Fi commands for you to test the module.

op
Description
op commands are used to set and query the Wi-Fi mode of the system.

Example

- op -Q
- op -S -o wmode

Parameter

Table 6-1. op Command Parameter

Parameter **Description**

-Q	Query Wi-Fi mode.
-S	Set Wi-Fi mode.
wmode	There are 3 Wi-Fi modes: • mode = 1: STA mode • mode = 2: AP mode • • mode = 3: STA+AP mode

sta
Description
sta commands are used to scan the STA network interface, connect or disconnect AP, and query the connecting status of STA network interface.

Example

- sta -S [-s ssid] [-b bssid] [-n channel] [-h] sta -Q
- sta -C [-s ssid] [-p password]
- sta -D

Parameter

Table 6-2. sta Command Parameter

Parameter	Description
-S scan	Scan Access Points.
-s ssid	Scan or connect Access Points with the ssid.
-b bssid	Scan the Access Points with the bssid.
-n channel	Scan the channel.
-h	Show scan results with hidden ssid Access Points.
-Q	Show STA connect stutus.
-D	Disconnected with current Access Points.

ap
Description
ap commands are used to set the parameter of AP network interface.

Example

- ap -S [-s ssid] [-p password] [-t encrypt] [-n channel] [-h] [-m max_sta]
- ap –Q
- ap –L

Parameter
Table 6-3. ap Command Parameter

Parameter **Description**

-S	Set AP mode.
-s ssid	Set AP ssid.
-p password	Set AP password.
-t encrypt	Set AP encrypt mode.
-h	Hide ssid.
-m max_sta	Set AP max connections.
-Q	Show AP parameters.
-L	Show MAC Address and IP Address of the connected station.

mac
Description
mac commands are used to query the MAC address of the network interface.

Examplemac -Q [-o mode]

Parameter
Table 6-4. mac Command Parameter

Parameter **Description**

-Q	Show MAC address.
-o mode	• mode = 1: MAC address in STA mode. • mode = 2: MAC address in AP mode.

dhcp
Description
dhcp commands are used to enable or disable dhcp server/client.

Example

- dchp -S [-o mode]
- dhcp -E [-o mode]
- dhcp -Q [-o mode]

Parameter
Table 6-5. dhcp Command Parameter

Parameter **Description**

-S	Start DHCP (Client/Server).
-E	End DHCP (Client/Server).
-Q	show DHCP status.
-o mode	• mode = 1 : DHCP client of STA interface. • mode = 2 : DHCP server of AP interface. • mode = 3 : both.

ip
Description
ip command are used to set and query the IP address of the network interface.

Example

- ip -Q [-o mode]
- ip -S [-i ip] [-o mode] [-m mask] [-g gateway]

Parameter
Table 6-6. ip Command Parameter

Parameter **Description**

-Q	Show IP address.
-o mode	• mode = 1 : IP address of interface STA. • mode = 2 : IP address of interface AP. • mode = 3 : both
-S	Set IP address.
-i ip	IP address.
-m mask	Subnet address mask.
-g gateway	Default gateway.

reboot
Description
reboot command is used to reboot the board.

Example
reboot

ram
ram command is used to query the size of the remaining heap in the system.

Example
ram

FCC Caution:
Any Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.
This device complies with part 15 of the FCC Rules.

Operation is subject to the following two conditions:

1. This device may not cause harmful interference, and
2. This device must accept any interference received, including interference that may cause undesired operation.

This transmitter must not be co-located or operating in conjunction with any other antenna or transmitter.

IMPORTANT NOTE:
Note: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

FCC Radiation Exposure Statement:
This equipment complies with FCC radiation exposure limits set forth for an uncontrolled environment .
This equipment should be installed and operated with minimum distance 20cm between the radiator& your body.

Documents / Resources

T-Embed User Guide LILYGO® V1.0 Copyright 2023	LILYGO T-Embed [pdf] User Guide 2ASYE-T-EMBED, 2ASYETEMBED, T-Embed

References

- GitHub - espressif/arduino-esp32: Arduino core for the ESP32
- Arduino IDE 1 Installation (Linux) | Arduino Documentation
- Arduino IDE 1 Installation (macOS) | Arduino Documentation
- Arduino IDE 1 Installation (Windows) | Arduino Documentation