

# ESP32-LyraTD-SYNA

## User Guide



Version 1.1  
Espressif Systems  
Copyright © 2019

# About This Guide

---

This document helps you to get started with the ESP32-LyraTD-SYNA development board.

## Release Notes

Date	Version	Release notes
2019.09	V1.0	Initial release.
2019.10	V1.1	Updated module figure in Chapter 1.1.

## Documentation Change Notification

Espressif provides email notifications to keep customers updated on changes to technical documentation. Please subscribe at <https://www.espressif.com/en/subscribe>.

## Certification

Download certificates for Espressif products from <https://www.espressif.com/en/certificates>.

# Table of Contents

---

<b>1. Overview .....</b>	<b>1</b>
1.1. ESP32-LyraTD-SYNA.....	1
1.2. Hardware Resources.....	1
<b>2. Configuring for AVS .....</b>	<b>3</b>
2.1. Software Preparation .....	3
2.2. Hardware Preparation .....	3
2.3. Flashing the Firmware .....	4
2.3.1. Driver Installation .....	4
2.3.2. Flashing ESP32.....	4
2.4. Network Configuration .....	7
2.4.1. Wi-Fi Networking.....	7
2.5. Interactive Function.....	10
2.5.1. Voice Recognition.....	10
2.5.2. Buttons .....	10
<b>A. Appendix - Schematics .....</b>	<b>11</b>



# 1.

# Overview

ESP32-LyraTD-SYNA is one of Espressif's Audio Development Board based on ESP32 MCU and Synaptics DSP. It is an Acoustic Echo Cancellation (AEC) solution, supporting voice recognition and voice wake-up. Audio files in the format of WAV, MP3, AAC, FLAC, OPUS and OGG can be decoded and encoded without quality loss. It also supports connection to Amazon's AVS (Alexa Voice Service), Google's Dialogflow and Google's GVA (Google Voice Assistant).

## 1.1. ESP32-LyraTD-SYNA

ESP32-LyraTD-SYNA is based on ESP32-WROVER-B, a BT/Wi-Fi combo module, and a digital signal processor (DSP) that features a two-microphone array for noise reduction and echo cancellation, as well as a Coder-Decoder (CODEC) that includes two pairs of DACs & ADCs with high sampling rates.

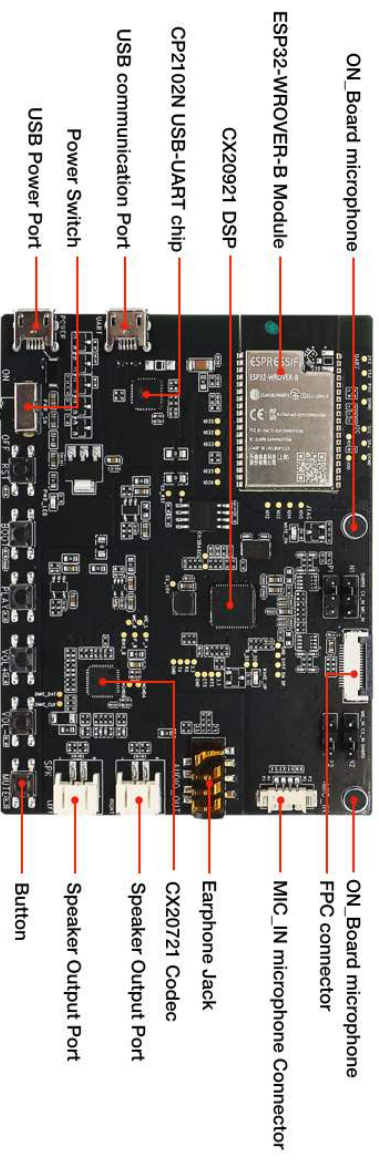


Figure 1-1. ESP32-LyraTD-SYNA

## 1.2. Hardware Resources

### 1. USB Power Port

Provides power supply to the whole system. It is recommended that the system be connected to a power adapter for sufficient current supply.

### 2. ESP32-WROVER-B module

This is a powerful, general-purpose, Wi-Fi+BT+BLE dual core MCU module, targeting a wide variety of applications ranging from low-power sensor networks to the most demanding tasks, such as voice encoding/decoding, music streaming and runs voice assistant client SDK.

### 3. CX20921 DSP

CX20921 is a dual core, far field voice input processor which supports Smart Source Pickup(SSP) noise suppression and various functionalities such as voice control, voice search, Voice over Internet Protocol (VOIP) and acoustic echo cancellation (AEC). This



audio DSP is used for wake-word detection. It captures audio data from two external microphones.

#### **4. CX20721 codec**

CX20721 is an audio Coder-Decoder which has integrated AudioSmart Class-D with 2.8W per channel.

#### **5. CP2102N USB-UART Chip**

An upgraded version of the CP2102 chip supports USB-to-UART conversion.

#### **⚠ Notice:**

*The IC design of this chip is a little different from CP2102.*

#### **6. USB Communication Port**

This is a normal USB communication port between PC and ESP32 module.

#### **7. ON-Board microphone**

There are two types of microphone connection, on-board and by MIC-IN connector.

#### **8. MIC-IN microphone connector**

Supports two external microphones.

#### **9. FPC connector**

This connection is reserved for connecting external sub board consisting of more devices, such as microphone array, buttons and LEDs etc.

#### **10. Earphone jack**

Used for plugging in earphones. Audio output is via codec.

#### **11. Speaker output port**

Supports two external speakers output via codec.

#### **12. Button**

The six buttons figure different functions, such as RST, BOOT, PLAY, VOL+/-, MUTE etc.



## 2. Configuring for AVS

---

### 2.1. Software Preparation

#### 1. Installing Flash Download Tool

Linux/macOS/Windows users can use the terminal to flash firmware. For more details, please check [ESP-IDF Programming Guide](#).

Windows users may also use a graphical tool to flash firmware onto the board using the [Flash Download Tool](#).

#### 2. Downloading Firmware

Please follow the build instructions of the Alexa SDK from the Github page: <https://github.com/espressif/esp-va-sdk>.

#### ⚠ Notice:

The GitHub repository uses Espressif's wake-word engine (MWE) for processing the 'Alexa' wake-word. This allows easy evaluations. For production use cases, please reach out to [sales@espressif.com](mailto:sales@espressif.com) or your Synaptics sales representative to obtain an Amazon certified wake-word engine.

#### 3. Debugging Tool

Use serial monitor/console that suits your development environment at baud rate of 115200.

#### 4. App for Provisioning

In order to run Alexa you need to provision LyraTD-SYNA with Alexa credentials and network settings (Wi-Fi SSID and passphrase).

- Android APK: <https://play.google.com/store/apps/details?id=com.espressif.provbleavs>
- iOS: <https://apps.apple.com/us/app/esp-alexaid1464127534>

### 2.2. Hardware Preparation

- 1 x PC (Linux Operating System is recommended).
- 1 x ESP32-LyraTD-SYNA.
- 2 x Micro USB cables (one is for the USB power supply).
- 1 x Speaker or 1 x earphone.



## 2.3. Flashing the Firmware

This document captures the process of flashing the firmware on Windows. Please refer to the [ESP-VA-SDK](#) repository on Github for setup on other platforms.

### 2.3.1. Driver Installation

Connect the board to a PC with a Micro-USB cable and install the USB-to-UART driver.

### 2.3.2. Flashing ESP32



Figure 2-1. Flash Download Tool

- In order to build examples of esp-va-sdk, please follow the instructions mentioned on the [GitHub page](#).
- Once you build the Alexa firmware, you will generate three artifacts: **bootloader.bin**, **partitions.bin** and **alexabin**.
- For Flash Download Tool users, follow the instructions given below.
  1. Run Flash Download Tool, and click on “**ESP32 Download Tool**” at the interface. See Figure 2-1 above.
  2. Then you will see the interface shown in Figure 2-2 below.

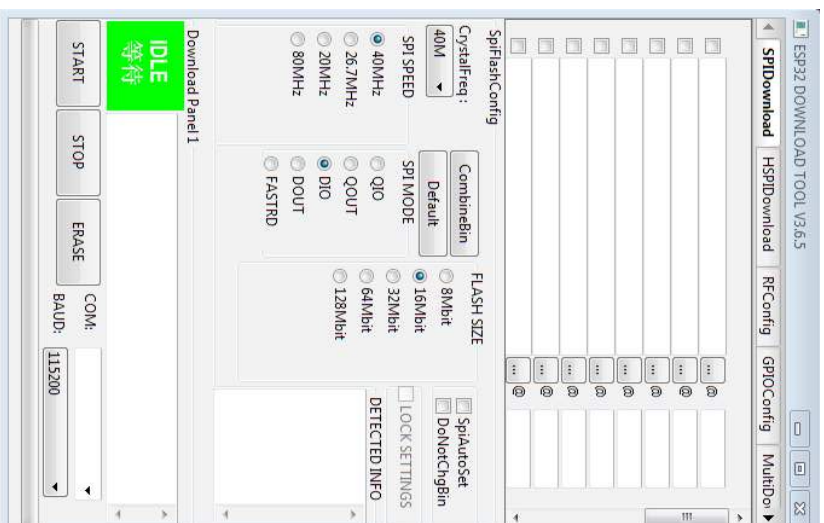


Figure 2-2: ESP32 Download Tool

3. Select the binary files (mentioned in table below) and set appropriate address in the tool as shown in Figure 2-3.

Binary Files	Address	Function
<i>bootloader.bin</i>	0x1 000	basic configuration
<i>partitions.bin</i>	0x8000	basic configuration
<i>alexax.bin</i>	0x1 0000	audio application





Figure 2-3. Flashing Configuration

4. The SPI Download configuration on the ESP32 Download Tool is shown in Figure 2-3. Select **“80 MHz”** for **“SPI SPEED”**, select **“DIO”** for **“SPI MODE”**, and select **“128 Mbit”** for **“FLASH SIZE”**.
5. Select the COM port in line with the one used by the PC; select **“1152000”** as a baud rate.
6. Click the **“START”** button on the ESP32 Flash Download Tool to start downloading the firmware.
7. Wait for the download to finish. See Figure 2-4.

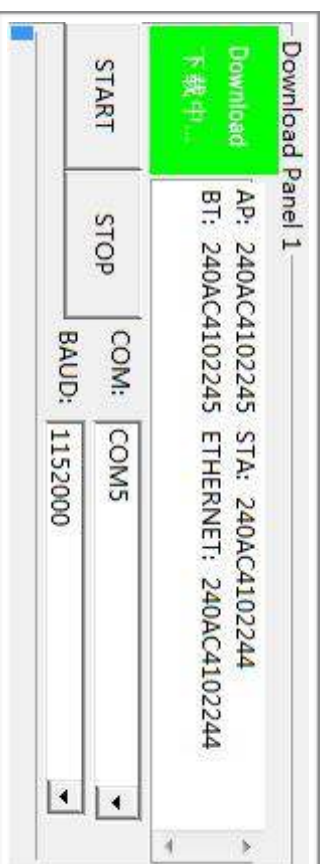


Figure 2-4. Downloading Firmware



**Note:**

Terminal users can follow the flashing instructions as available on the [GitHub page](#).

## 2.4. Network Configuration

### 2.4.1. Wi-Fi Networking

1. After successfully downloading the firmware, power on the board again by pressing the EN button.
2. To configure the network for the first time:
  - Turn on the bluetooth in your phone. Open Espressif's Alexa programming app and click on **Provision New Device**. Make sure your phone has internet connectivity.

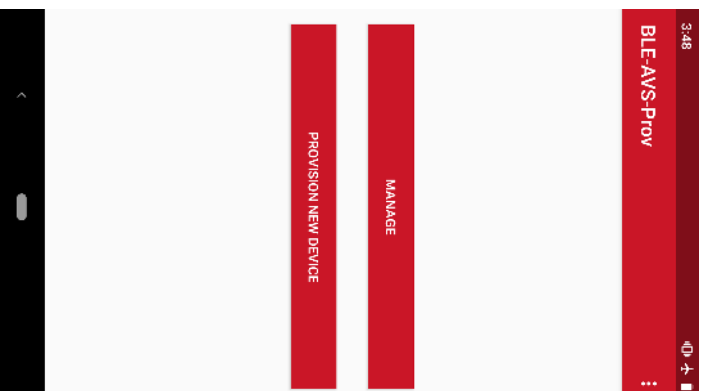


Figure 2-5. Provision New Device

- The provisioning happens over BLE (Bluetooth Low Energy). Android mandates applications to have Location permission for accessing BLE. Please provide the permission when the pop-up shows up.
- You will see the list of available devices that needs to be provisioned, namely **ESP-Alexa-WXYZ**. Select the device that you wish to provision.

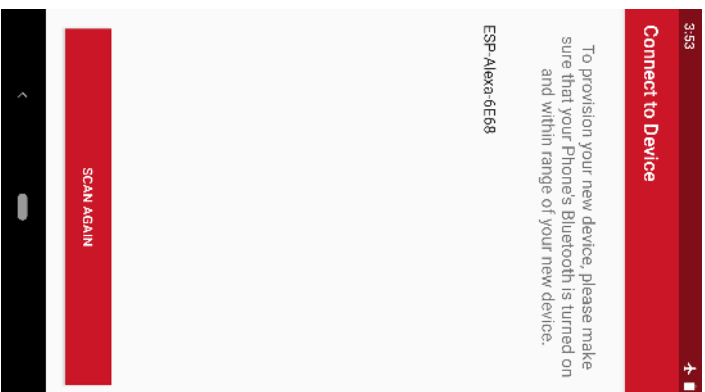


Figure 2-6. List of available devices

- Next step is to login to your amazon account.

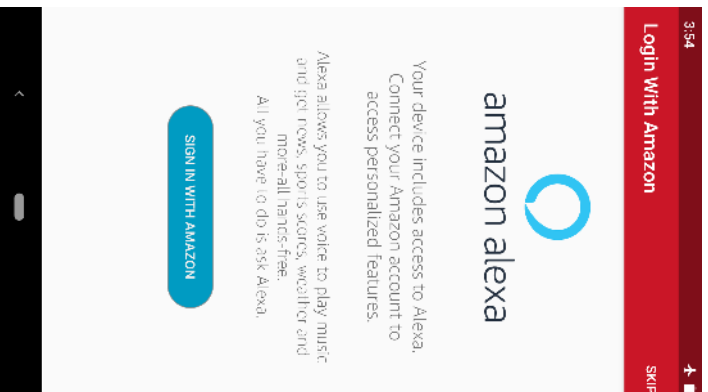


Figure 2-7. Login to your amazon

- After successful login and authorization, you will be shown a list of Wi-Fi networks in the vicinity. Choose the network that you wish the device to connect to. Once



selected, enter the passphrase of that Wi-Fi network and click on the **provision** button.

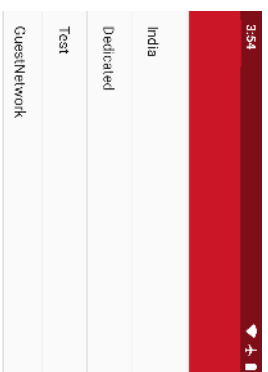


Figure 2-8. Available Wi-Fi networks

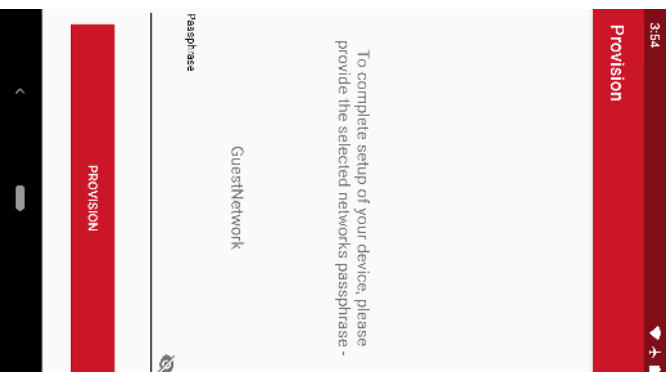


Figure 2-9. Enter your passphrase

- On successfully connecting to the Wi-Fi network, after provisioning, the DSP and Alexa will be initialized, and ready to take voice commands.



## 2.5. Interactive Function

### 2.5.1. Voice Recognition

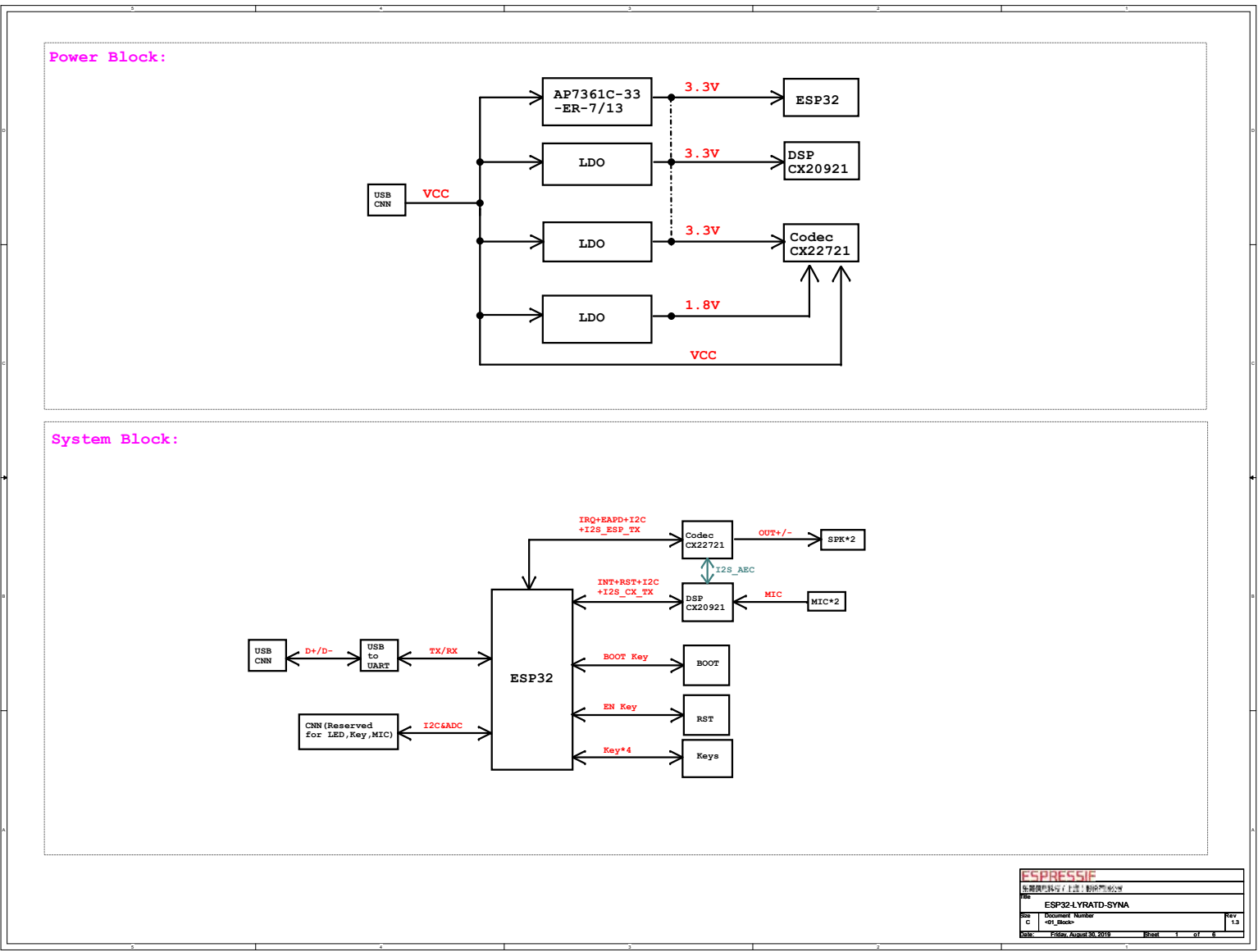
- Once all initialization are done, the device is ready. You can either use "**PLAY**" button on the board (Tap-To-Talk) or say "Alexa" to start a conversation. For Tap-to-Talk, press and release the button and speak.
- You can now ask any command like:
  - Tell me a joke.
  - How is the weather?
  - Will it rain today?
  - Sing a song.
  - Play TuneIn radio.
  - Set volume level to 7.

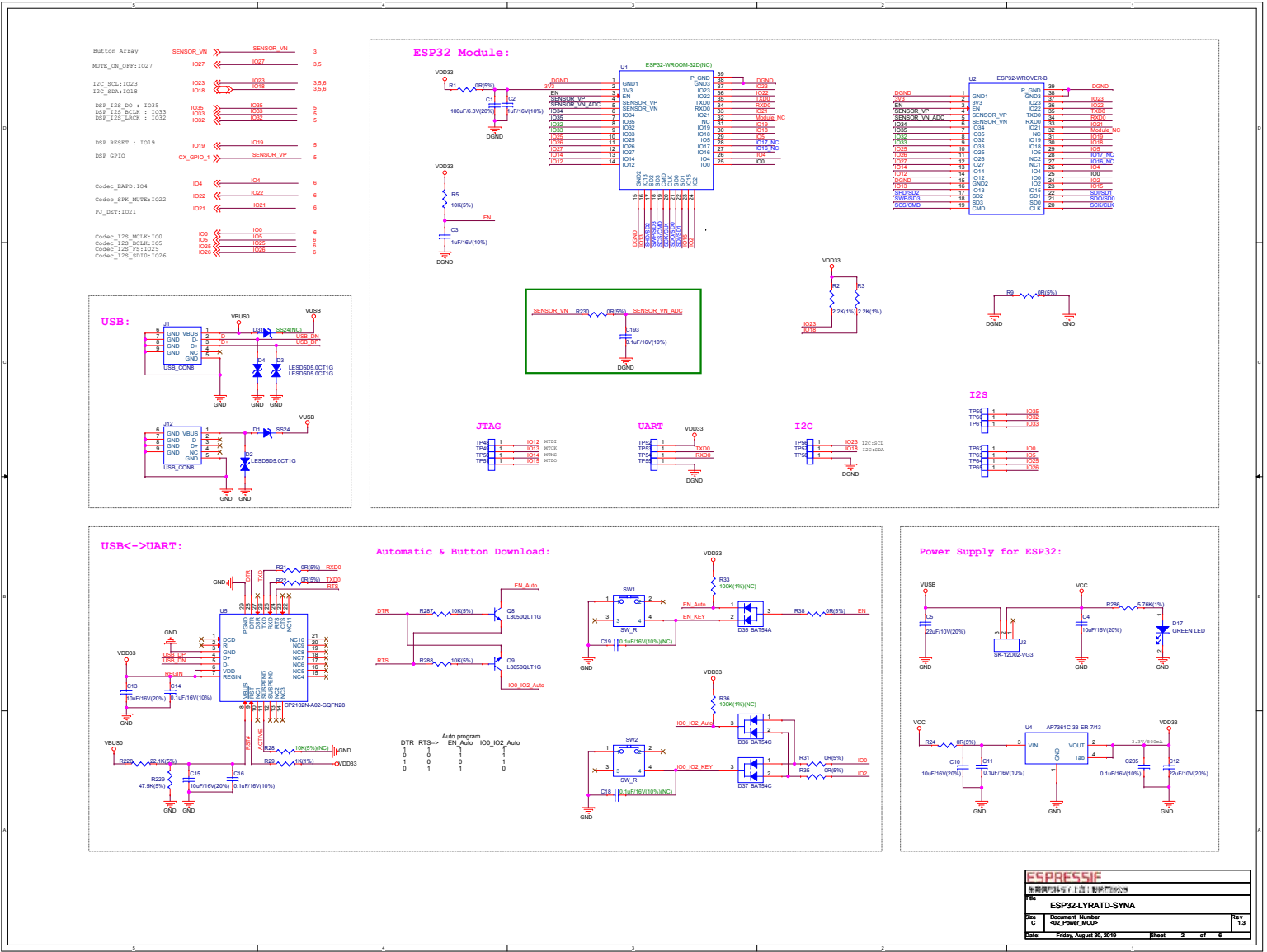
### 2.5.2. Buttons

- The "**PLAY**" button can be used as tap-to-talk. The same button can also be used to disable an Alarm/Timer.
- The "**Vol+**" and "**Vol-**" buttons can be used to adjust the volume of the sound. Press them repeatedly until you reach the desired sound volume.
- The "**Mute**" button can be used to mute/unmute the device.
- To reset to factory settings, press "**Vol+**" and "**Vol-**" button together for about 10 seconds.



# A. Appendix - Schematics





<b>ESPRESSIF</b>	
Espressif Systems / 乐鑫科技	
<b>ESP32-LYRATD-SYNA</b>	
Doc. No.	ESP32-lyratd-syna
Rev.	1.0
Date	Friday, August 30, 2019
Sheet	2 of 6

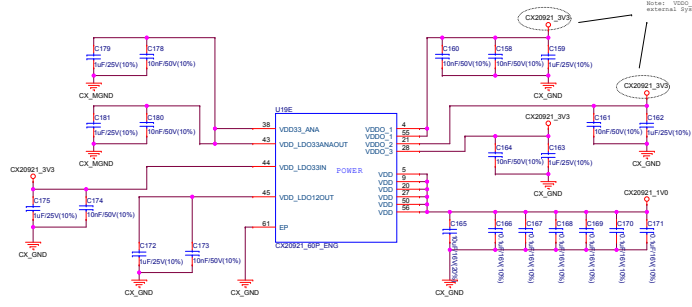
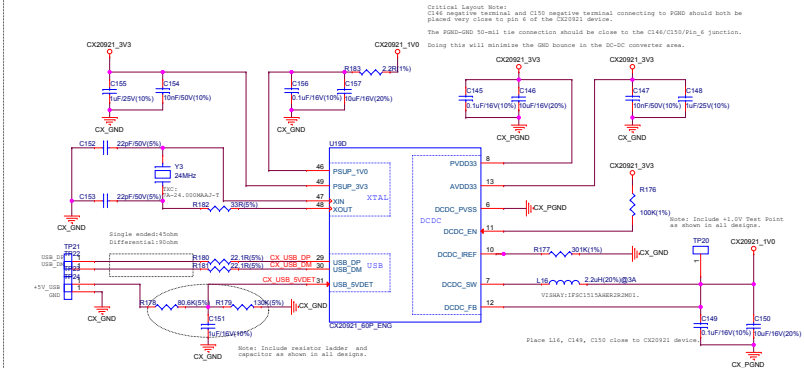
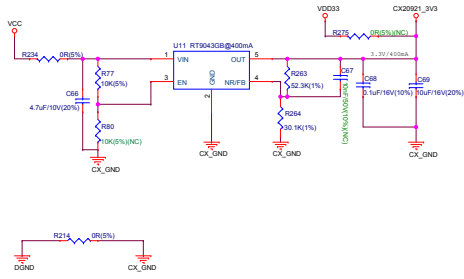






**Power for CX20921:**

From CX20921 datasheet P18.  
Power consumption max is 3.3x49mA=161.7mW.



VDD0\_1 - Input supply which internally powers I2S TX, I2C Slave, SPI Master, and RESET interfaces.  
VDD0\_2 - Input supply which internally powers I2S RX, GPIOs, UART and DMIC interfaces.  
VDD0\_3 - Input supply which internally powers the USB interface, must always connect to +1.0V only.

**DCDC/DIGITAL/MIC GROUNDS**

Short PGND to DGND directly with a trace.  
Short MGND to DGND directly with a trace.  
Do not use a resistor or ferrite between the grounds.



Create PGND plane, separate from normal top digital ground plane for internal DCDC.  
Tie MGND to GND with 50-mil tie from the GND to digital ground plane together at C146/C150/Pin\_6 junction using 50mil traces.

Mic section has its own plane.  
Tie MGND to GND with 50-mil tie from the GND to digital ground plane together at C146/C150/Pin\_6 junction using 50mil traces.

<b>ESPRESSIF</b>			
Espressif Systems (HK) Limited 乐鑫科技(香港)有限公司			
ESP32-LYRATD-SYNA			
Doc ID:	Document Number		Rev:
C	004_CX20921_Power		1.0
File:	Espressif August 2019	Sheet:	4 of 6







Espressif IoT Team

[www.espressif.com](http://www.espressif.com)

#### **Disclaimer and Copyright Notice**

Information in this document, including URL references, is subject to change without notice.

THIS DOCUMENT IS PROVIDED AS IS WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE.

All liability, including liability for infringement of any proprietary rights, relating to use of information in this document is disclaimed. No licenses express or implied, by estoppel or otherwise, to any intellectual property rights are granted herein.

The Wi-Fi Alliance Member logo is a trademark of the Wi-Fi Alliance. The Bluetooth logo is a registered trademark of Bluetooth SIG.

All trade names, trademarks and registered trademarks mentioned in this document are property of their respective owners, and are hereby acknowledged.

**Copyright © 2019 Espressif Inc. All rights reserved.**