AS1170 EVK – setup with ESP32 board

Application Note

Published by ams-OSRAM AG
Tobelbader Strasse 30,
8141 Premstaetten Austria
Phone +43 3136 500-0
ams-osram.com
© All rights reserved

AS1170 EVK – setup with ESP32 board

Application Note No. AN001076

Valid for:
AS1170

Abstract

This application note provides detailed instructions on setting up and running an Arduino code for an ESP32 microcontroller. The code demonstrates how to interface with I²C devices, use PWM for LED control, and read/write to registers.

Table of contents

1 Introduction 4

2 Preliminary setup 4

2.1 Installing the USB to UART driver 4

2.2 Setting up the Arduino IDE 5

3 Hardware setup 6

3.1 Components required 6

3.2 Circuit diagram 6

4 Software setup 7

4.1 Libraries required 7

4.2 Installing libraries 8

5 Code explanation 8

5.1 Constants and defines 8

5.2 Register structure and field map 8

5.3 Functions 9

5.4 Setup section 9

5.5 Loop section 9

6 Running the code 10

7 Troubleshooting 10

8 Summary / results 10

9 Revision information 10

1 Introduction

This application note provides detailed instructions on setting up and running an Arduino code for an ESP32 microcontroller. The code demonstrates how to interface with I²C devices, use PWM for LED control, and read/write to registers.

2 Preliminary setup

2.1 Installing the USB to UART driver

To communicate with the ESP32 S2 DevKit M1, you need to install the USB to UART driver. Follow these steps:

- Download the Driver: Visit the Silicon Labs website CP210x USB to UART Bridge VCP Drivers - Silicon Labs (silabs.com) and download the appropriate driver for your operating system.
- Install the Driver: Follow the installation instructions provided on the website to install the driver on your computer.

2.2 Setting up the Arduino IDE

To program the ESP32 S2 DevKit M1, you need to set up the Arduino IDE with the appropriate board support package. Follow these steps:

- Install the Arduino IDE: If you haven't already, download and install the Arduino IDE from the Arduino website.
- Open the Arduino IDE: Launch the Arduino IDE on your computer.
- Add the ESP32 Board Manager URL:
  - Go to File > Preferences.
  - In the Additional Board Manager URLs field, add the following URL:

    https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json
  - Click OK to save the changes.
- Install the ESP32 Board Package:
  - Go to Tools > Board > Boards Manager.
  - In the Boards Manager window, search for esp32.
  - Find the package titled esp32 by Espressif Systems and click Install.
  - Ensure you install version 2.0.17.
- Select the ESP32 S2 DevKit M1 Board:
  - Go to Tools > Board > esp32.
  - Select ESP32S2 Dev Module from the list.

3 Hardware setup

3.1 Components required

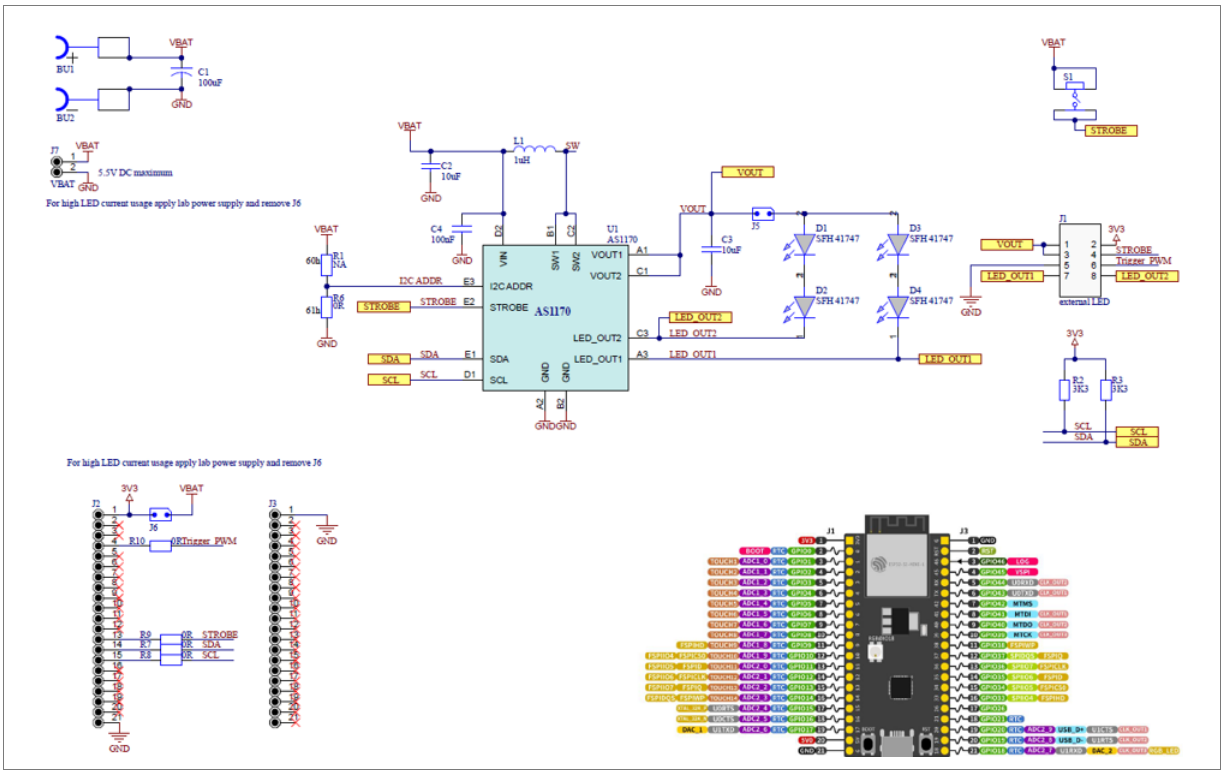
- ESP32 Development Board
- AS1170 EVK
- PU/PD resistors/jumpers
- Connecting Wires

3.2 Circuit diagram

Connect the components as follows:

- SDA (I²C Data): GPIO 14
- SCL (I²C Clock): GPIO 15
- PWM Output: GPIO 5

Figure 1: Circuit diagram



4 Software setup

4.1 Libraries required

Ensure the following libraries are installed in your Arduino IDE:

- Wire.h
- SD.h
- HardwareSerial.h
- Arduino.h
- Driver/ledc.h

4.2 Installing libraries

To install the libraries, go to Sketch > Include Library > Manage Libraries and search for the required libraries.

5 Code explanation

5.1 Constants and defines

- I²C Address: _slaveaddress is set to 0x30.
- Pins: SDA_PIN and SCL_PIN are set to GPIO 14 and 15 respectively.
- PWM Settings: Frequency is set to 5000 Hz, channel to 0, resolution to 8 bits, and output pin to GPIO 5

5.2 Register structure and field map

A Register structure is defined to hold register information including name, address, default value, and fields.

The following table resumes the whole register map and fields.

Figure 2: Register map and fields

Register definition	Addr	Default	Access	<b7>	<b6>	<b5>	<b4>	<b3>	<b2>	<b1>	<b0>
ChipID	0h	B4h	R	fixed_id	fixed_id	fixed_id	fixed_id	fixed_id	version	version	version
Current set LED1	1h	9Ch	R/W	led_current1	led_current1	led_current1	led_current1	led_current1	led_current1	led_current1	led_current1
Current set LED2	2h	9Ch	R/W	led_current2	led_current2	led_current2	led_current2	led_current2	led_current2	led_current2	led_current2
Low voltage	4h	2Ch	R/W	const_v_mode	vin_low_v_shutdown	vin_low_v	vin_low_v	vin_low_v	vin_low_v_run	vin_low_v_run	vin_low_v_run
Flash timer	5h	23h	R/W	flash_timeout	flash_timeout	flash_timeout	flash_timeout	flash_timeout	flash_timeout	flash_timeout	flash_timeout
Control	6h	00h	R/W				auto_strobe	out_on	reserved	mode_setting	mode_setting
Strobe signaling	7h	C0h	R/W	strobe_on	strobe_type						
Fault	8h	00h	R/Sc	fault_ovp	fault_led_short	fault_overtemp	fault_timeout	fault_tx mask	reserved	reserved	fault_uvlo
PWM and Indicator	9h	00h	R/W			load_balance_on	led_out2above1	led_out1above2	freq_switch_on	inct_pwm	inct_pwm
Minimum LED current	Eh	00h	R	led_current_min	led_current_min	led_current_min	led_current_min	led_current_min	led_current_min	led_current_min	led_current_min
Actual LED current	Fh	00h	R	led_current_actual	led_current_actual	led_current_actual	led_current_actual	led_current_actual	led_current_actual	led_current_actual	led_current_actual
Password register	80h	00h	W	password	password	password	password	password	password	password	password
Current boost	81h	00h	R/W								current boost

5.3 Functions

- readRegister: Reads a byte or specific bit from a register.
- writeRegister: Writes a byte to a register.
- loadRegisters: Loads register information from a CSV file.
- printRegisterValue: Prints the value of a register.
- pollserial: Polls the serial port for incoming data.

5.4 Setup section

- Initializes I²C communication.
- Sets up serial communication at 115200 baud.
- Configures the PWM channel and attaches it to the GPIO pin.
- Loads registers from a CSV file.
- Reads and prints initial register values.

5.5 Loop section

- Continuously polls the serial port for commands.
- Handles commands to read, write, and reset registers, and control PWM output.

6 Running the code

- Upload the Code: Connect your ESP32 to the computer and upload the code using the Arduino IDE.
- Open Serial Monitor: Open the Serial Monitor to view register values and send commands.
- Send Commands: Use the Serial Monitor to send commands such as read, write, and reset.

7 Troubleshooting

- No Output on Serial Monitor: Ensure the baud rate is set to 115200.
- I²C Communication Issues: Check the connections and ensure the correct I²C address is used.
- PWM Not Working: Verify the PWM pin and settings.

8 Summary / results

This guide provided a comprehensive overview of setting up and running an Arduino code for an ESP32. The code demonstrates I²C communication, PWM control, and register manipulation.

9 Revision information

Changes to current revision v1-00	Page
Initial production version	

- Page and figure numbers for the previous version may differ from page and figure numbers in the current revision.
- Correction of typographical errors is not explicitly mentioned.

ABOUT ams OSRAM Group (SIX: AMS)

The ams OSRAM Group (SIX: AMS) is a global leader in intelligent sensors and emitters. By adding intelligence to light and passion to innovation, we enrich people's lives. With over 110 years of combined history, our core is defined by imagination, deep engineering expertise and the ability to provide global industrial capacity in sensor and light technologies. Our around 20,000 employees worldwide focus on innovation across sensing, illumination and visualization to make journeys safer, medical diagnosis more accurate and daily moments in communication a richer experience. Headquartered in Premstaetten/Graz (Austria) with a co-headquarters in Munich (Germany), the group achieved EUR 3.6 billion revenues in 2023. Find out more about us on https://ams-osram.com

DISCLAIMER

PLEASE CAREFULLY READ THE BELOW TERMS AND CONDITIONS BEFORE USING THE INFORMATION SHOWN HEREIN. IF YOU DO NOT AGREE WITH ANY OF THESE TERMS AND CONDITIONS, DO NOT USE THE INFORMATION.

The information provided in this general information document was formulated using the utmost care; however, it is provided by ams-OSRAM AG or its Affiliates* on an "as is" basis. Thus, ams-OSRAM AG or its Affiliates* does not expressly or implicitly assume any warranty or liability whatsoever in relation to this information, including – but not limited to – warranties for correctness, completeness, marketability, fitness for any specific purpose, title, or non-infringement of rights. In no event shall ams-OSRAM AG or its Affiliates* be liable – regardless of the legal theory – for any direct, indirect, special, incidental, exemplary, consequential, or punitive damages arising from the use of this information. This limitation shall apply even if ams-OSRAM AG or its Affiliates* has been advised of possible damages. As some jurisdictions do not allow the exclusion of certain warranties or limitations of liabilities, the above limitations and exclusions might not apply. In such cases, the liability of ams-OSRAM AG or its Affiliates* is limited to the greatest extent permitted in law.

ams-OSRAM AG or its Affiliates* may change the provided information at any time without giving notice to users and is not obliged to provide any maintenance or support related to the provided information. The provided information is based on special conditions, which means that the possibility of changes cannot be precluded.

Any rights not expressly granted herein are reserved. Other than the right to use the information provided in this document, no other rights are granted nor shall any obligations requiring the granting of further rights be inferred. Any and all rights and licenses regarding patents and patent applications are expressly excluded.

It is prohibited to reproduce, transfer, distribute, or store all or part of the content of this document in any form without the prior written permission of ams-OSRAM AG or its Affiliates* unless required to do so in accordance with applicable law.

* ("Affiliate" means any existing or future entity: (i) directly or indirectly controlling a Party; (ii) under the same direct, indirect or joint ownership or control as a Party; or (iii) directly, indirectly or jointly owned or controlled by a Party. As used herein, the term "control" (including any variations thereof) means the power or authority, directly or indirectly, to direct or cause the direction of the management and policies of such Party or entity, whether through ownership of voting securities or other interests, by contract or otherwise.)

For further information on our products please see the Product Selector and scan this QR Code.

Published by ams-OSRAM AG
Tobelbader Strasse 30, 8141 Premstaetten, Austria
ams-osram.com © All Rights Reserved.

Published by ams-OSRAM AG
Tobelbader Strasse 30,
8141 Premstaetten Austria
Phone +43 3136 500-0
ams-osram.com
© All rights reserved