Nine-Channel LED Driver with Execution ENGINE and Charge Pump

Features

- Power Supply Voltage Range: 2.7V to 5.5V
- Operating Temperature Range: -40°C to 105°C
- 9 Independent Programmable LED Channels
- Each Channel Supports an 8-Bit Current Setting, with a Maximum Output Current of 25.5mA
- Each Channel Features 12-Bit PWM Control Resolution
- High-efficiency Adaptive Charge Pump (1x/1.5x) with Efficiency up to 94%
- Charge Pump Incorporates Soft Start, Overcurrent, and Short-Circuit Protection
- LED Driver Efficiency Reaches up to 93%
- Integrated Open-Circuit, Short-Circuit, and Voltage Error Detection
- Automatic Power-Saving Mode
- Three Independent Program Execution Engines
- Large SRAM Program Memory
- I²C interface
- Supports Cascading Applications
- Low-level external trigger function
- AW21209FOR package: FOWLP 2.26X2.26-25B
- AW21209QNR package: QFN 4X4-24L

Applications

- Ambient light
- Indicator light
- Electronic cigarette
- Bluetooth headphones
- Speaker
- Programmable current source
- VR/XR/AR

General Description

AW21209 is a programmable 9-channel LED driver specifically designed for mobile devices. It features an internal program memory that enables it to operate independently without the need for an external processor and generate the desired lighting effects. There are two products in total: AW21209FOR and AW21209QNR. Since the AW21209QNR has no GPO pins, all control logic related to GPO (such as the GPO bit in the IO_CTRL register and the mapping function of GPO in the LED mapping instruction) is not applicable.

AW21209 is equipped with an efficient charge pump that can drive LEDs across the entire voltage range of lithium-ion batteries and automatically selects the optimal charge pump gain to meet the forward voltage requirements of the LEDs, ensuring high efficiency over a wide operating voltage range. When the LEDs are not activated, the AW21209 automatically enters a power-saving mode, significantly reducing current consumption in idle states.

AW21209 features a control interface compatible with the I²C protocol and supports four different address selections, facilitating system integration. The TRIG pin can trigger the operation of light effects and synchronize multiple devices. The INT pin can send an interrupt signal to the processor after the lighting sequence is completed. The GPO pin can be used as a digital control pin for other devices.

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Typical Application Circuit

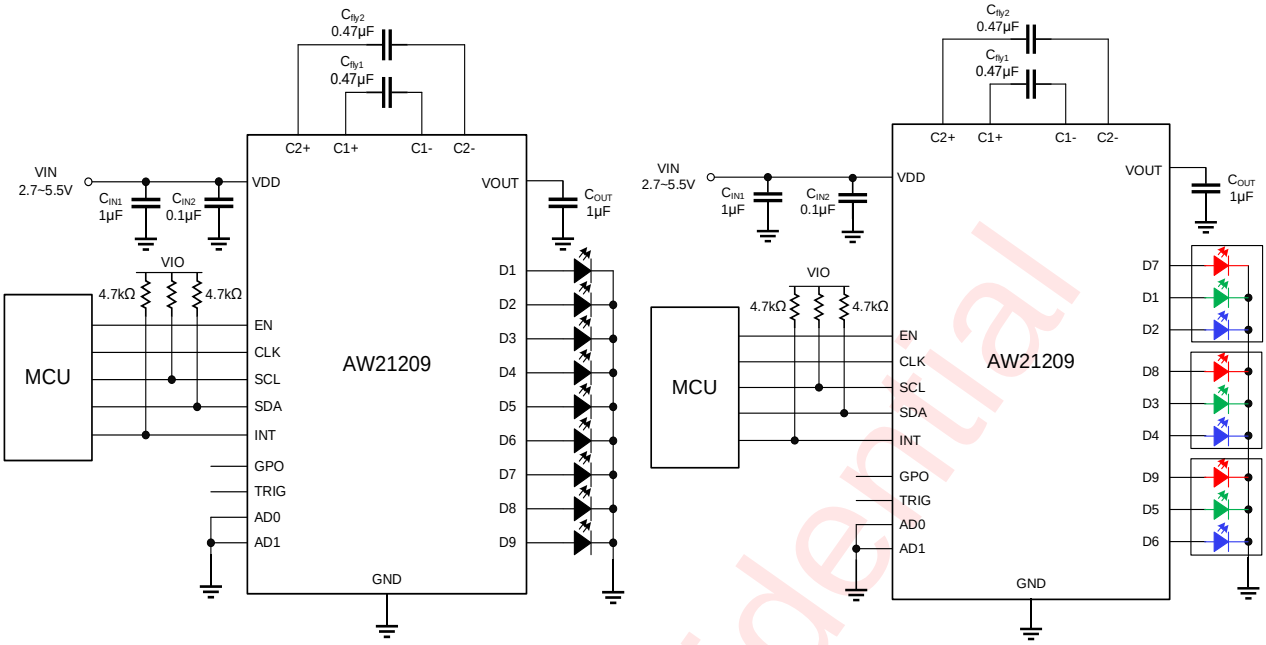
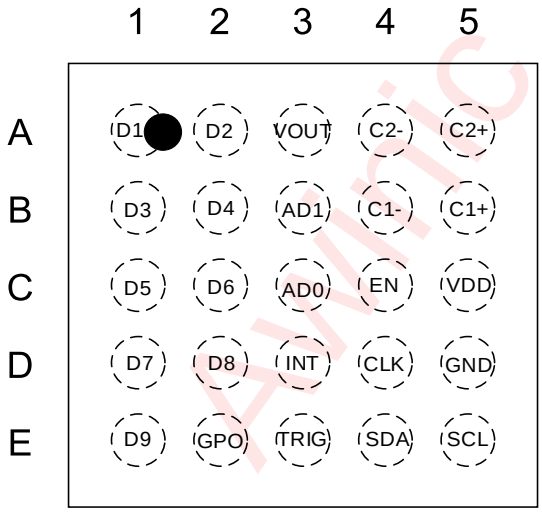


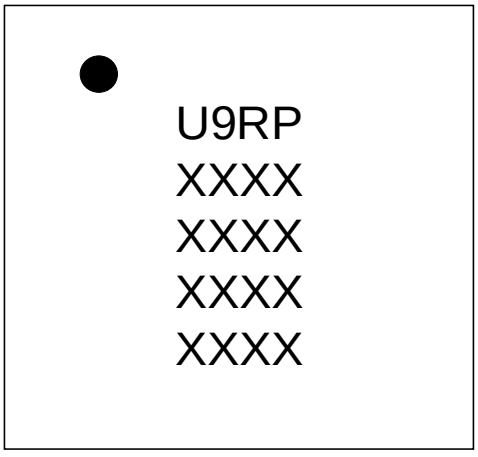
Figure 1 AW21209 Simple Application Circuit

Pin Configuration And Top Mark

AW21209FOR
(Top View)



AW21209FOR Marking
(Top View)

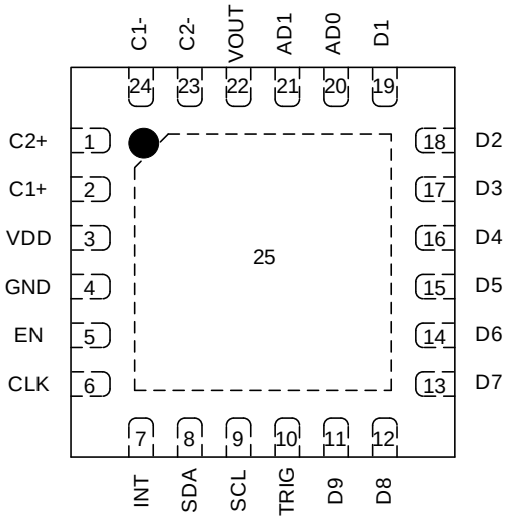


U9RP - AW21209FOR
XXXX/XXXX/XXXX/XXXX -
Production Tracing Code

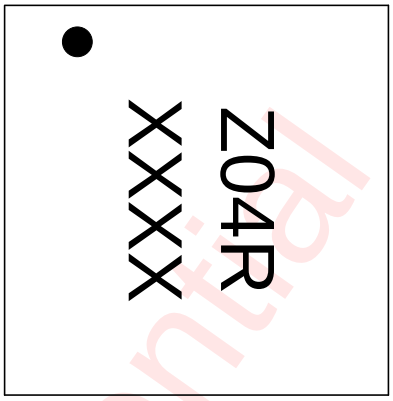
Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

AW21209QNR
(Top View)



AW21209QNR Marking
(Top View)



Z04R - AW21209QNR
XXXX - Production Tracing Code

Figure 2 Pin Configuration and Marking

Pin Definition

No. AW21209FOR	No. AW21209QNR	NAME	DESCRIPTION
A1	19	D1	Constant current source, connect to LED's anode.
A2	18	D2	Constant current source, connect to LED's anode.
A3	22	VOUT	Charge pump output voltage, connect to a 1μF capacitor.
A4	23	C2-	Switched cap flying cap connection. Connect a 0.47μF capacitor between this pin and C2+.
A5	1	C2+	Switched cap flying cap connection. Connect a 0.47μF capacitor between this pin and C2-.
B1	17	D3	Constant current source, connect to LED's anode.
B2	16	D4	Constant current source, connect to LED's anode.
B3	21	AD1	I^2C address setting, connects to GND or VDD for different device address of I^2C.
B4	24	C1-	Switched cap flying cap connection. Connect a 0.47μF capacitor between this pin and C1+.
B5	2	C1+	Switched cap flying cap connection. Connect a 0.47μF capacitor between this pin and C1-.
C1	15	D5	Constant current source, connect to LED's anode.
C2	14	D6	Constant current source, connect to LED's anode.
C3	20	AD0	I^2C address setting, connects to GND or VDD for different device address of I^2C.

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

C4	5	EN	Shutdown the chip when pulled low.
C5	3	VDD	Power supply: 2.7V~5.5V.
D1	13	D7	Constant current source, connect to LED's anode.
D2	12	D8	Constant current source, connect to LED's anode.
D3	7	INT	Interrupt, low active, leave unconnected if not used.
D4	6	CLK	External 32kHz clock input, connect to GND if not used.
D5	4	GND	Ground.
E1	11	D9	Constant current source, connect to LED's anode.
E2	-	GPO	General purpose output, leave unconnected if not used.
E3	10	TRIG	Trigger, connect to GND if not used.
E4	8	SDA	Serial data I/O for I^2C interface.
E5	9	SCL	Serial clock input for I^2C interface.
-	25	Thermal pad	Only for heat dissipation

Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Functional Block Diagram

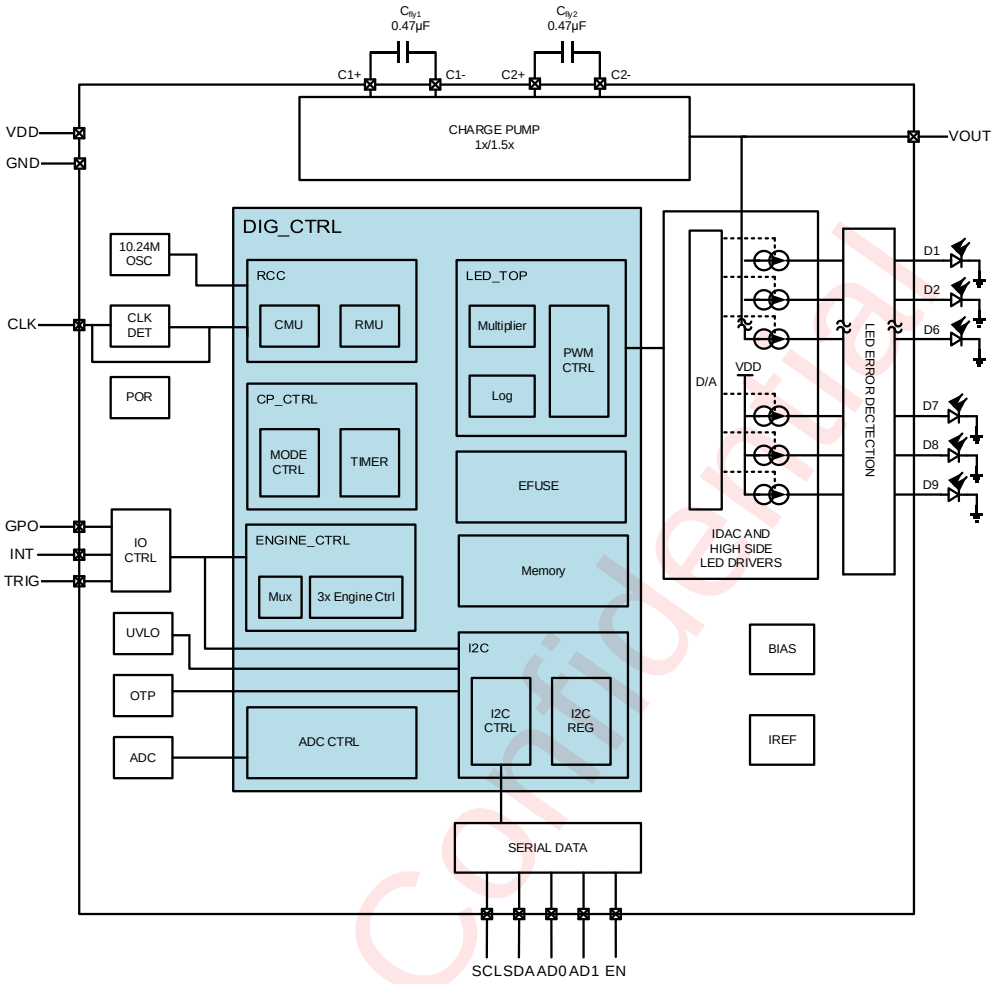


Figure 3 Functional Block Diagram

Ordering Information

Part Number	Temperature	Package	Marking	Moisture Sensitivity Level	Environmental Information	Delivery Form
AW21209FOR	-40℃～105℃	FOWLP 2.26X2.26-25B	U9RP	MSL1	ROHS+HF	3000 units/ Tape and Reel
AW21209QNR	-40℃～105℃	QFN 4X4-24L	Z04R	MSL1	ROHS+HF	4500 units/ Tape and Reel

Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Absolute Maximum Ratings(NOTE1)

PARAMETERS		RANGE
Supply voltage range VDD		-0.3V to 6V
Input voltage range	D1 to D9, C1-,C1+,C2-,C2+,VOUT, EN,CLK,INT,SDA,SCL,TRIG,ASEL0,ASEL1,	-0.3V to VDD
Junction-to-ambient thermal resistance θ_{JA} (AW21209FOR)		78.9°C/W
Junction-to-ambient thermal resistance θ_{JA} (AW21209QNR)		50°C/W
Maximum operating junction temperature T_{JMAX}		150℃
Storage temperature T_{STG}		-65°C to 150°C
Lead temperature (soldering 10 seconds)		260°C
ESD (Including CDM HBM)(NOTE 2)		
HBM		±4kV
CDM		±1.5kV
Latch-Up		
Test condition: JESD78F		+IT：200mA -IT：-200mA

NOTE1: Conditions out of those ranges listed in "absolute maximum ratings" may cause permanent damages to the device. In spite of the limits above, functional operation conditions of the device should within the ranges listed in "recommended operating conditions". Exposure to absolute-maximum-rated conditions for prolonged periods may affect device reliability.

NOTE2: The human body model is a 100pF capacitor discharged through a 1.5kΩ resistor into each pin. HBM test method: ESDA/JEDEC JS-001-2023(AW21209FOR), ESDA/JEDEC JS-001-2024(AW21209QNR), CDM test method: ESDA/JEDEC JS -002-2022

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Min.	Typ.	Max.	Unit
VDD	Input voltage	2.7	3.6	5.5	V
	Voltage on logic pins (input or output pins)	0		VDD	
I_{OUT}	Recommended charge pump load current	0		150	mA
T_A	Operating free-air temperature range	-40°	25	105	°C
C_{IN1}	Input Capacitance	0.5	1		μF
C_{IN2}	Input Capacitance		0.1		μF
C_{OUT}	Output Capacitance	0.5	1	4.7	μF
C_{fly1}	Charge pump capacitance	0.24	0.47	1	μF
C_{fly2}	Charge pump capacitance	0.24	0.47	1	μF

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Electrical Characteristics

Typical values apply for T_A=25℃, V_{DD}=3.6V ,V_{EN}=1.2V, C_{out}=1μF, C_{IN}=0.1μF, C_{fly1}= C_{fly2}=0.47μF.

PARAMETER		TEST CONDITION	MIN	TYP	MAX	UNIT
Power supply voltage and current						
I_{VDD}	Standby supply current	EN=L,CHIP_EN=0(bit),external 32-Khz clock running or not running		0.2	1	μA
		EN=H, CHIP_EN=0, external 32KHZ clock not running		1	2	μA
		EN=H, CHIP_EN=0, external 32KHZ clock running		1.4	3	μA
	Normal mode supply current	External 32KHz clock running, charge pump and current source outputs disabled		0.33	0.5	mA
		Charge pump in 1X mode, no load current source outputs disabled		0.55	0.8	mA
		Charge pump in 1.5X mode, no load, current source outputs disabled		1.8	2	mA
	Power save mode supply current	External 32Khz clock running		5	10	μA
		Internal oscillator running		0.3	0.5	mA
f_{osc}	Internal oscillator frequency accuracy		-4%		4%	
OTP	Over temperature protection threshold			150		℃
OTP_{HYS}	Over temperature protection hysteresis			20		℃
Charge Pump						
R_{out}	Charge pump output resistance	Gain=1.5X		3.5		Ω
		Gain=1X		1		
T_{ON}	Vout turn-on time	VDD=3.6V, Iout=60mA			120	μs
LED Driver						
$I_{Leakage}$	Leakage current (outputs D1 to D9)	PWM =0%		0.1	1	μA
I_{MAX}	Maximum source current	Outputs D1 to D9		25.5		mA
$I_{accuracy}$	Output current accuracy	Output current set to 17.5mA	-4		4	%
I_{MATCH}	Matching	Output current set to 17.5mA		1	2.5	%
f_{led}	LED switching frequency			312		Hz
V_{SAT}	Saturation voltage	Output current set to 17.5mA		50	100	mV
ADC						
LSB	Least significant bit			28.6		mV
E_{ABS}	Total unadjusted error	V_{IN_TEST}=0V to VDD		<±3		LSB

Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

PARAMETER		TEST CONDITION	MIN	TYP	MAX	UNIT
t_{conv}	Conversion time			2.7		ms
V_{IN_TEST}	DC voltage sample range		0		5	V
LOGIC INPUT EN						
V_{IL}	Input low level				0.24	V
V_{IH}	Input high level		0.96			V
I_L	Input current		-1		1	μA
T_{DELAY}	Input delay			2		μs
LOGIC IUTPUT SDA, TRIG, INT						
V_{IL}	Input low level				0.2* V_{EN}	V
V_{IH}	Input high level		0.8* V_{EN}			V
I_L	Input current		-1		1	μA
LOGIC OUTPUT SDA, TRIG, INT						
V_{OL}	Output low level	I_{OUT}=3mA(pullup current)		0.3		V
I_L	Output leakage current	V_{OUT}=2.8V			1	μA
LOGIC OUTPUT GPO						
V_{OL}	Output low level	I_{OUT}=3mA		0.3		V
V_{OH}	Output high level	I_{OUT}=-2mA		VDD-0.3		
I_L	Output leakage current	V_{OUT}=2.8V			1	μA

I²C INTERFACE TIMING REQUIREMENTS

Parameter		Fast Mode		Fast Mode Plus		Unit
		Min.	Max.	Min.	Max.	
F_{SCL}	Interface clock frequency	-	400	-	1000	kHz
$T_{HD:STA}$	(Repeat-start) START condition hold time	0.6	-	0.26	-	μs
T_{LOW}	Low level width of SCL	1.3	-	0.5	-	μs
T_{HIGH}	High level width of SCL	0.6	-	0.26	-	μs
$T_{SU:STA}$	(Repeat-start) START condition setup time	0.6	-	0.26	-	μs
$T_{HD:DAT}$	Data hold time	0	-	0	-	μs
$T_{SU:DAT}$	Data setup time	0.1	-	0.05	-	μs
T_R	Rising time of SDA and SCL	-	0.3	-	0.12	μs
T_F	Falling time of SDA and SCL	-	0.3	-	0.12	μs
$T_{SU:STO}$	STOP condition setup time	0.6	-	0.26	-	μs
T_{BUF}	Time between start and stop condition	1.3	-	0.5	-	μs

Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

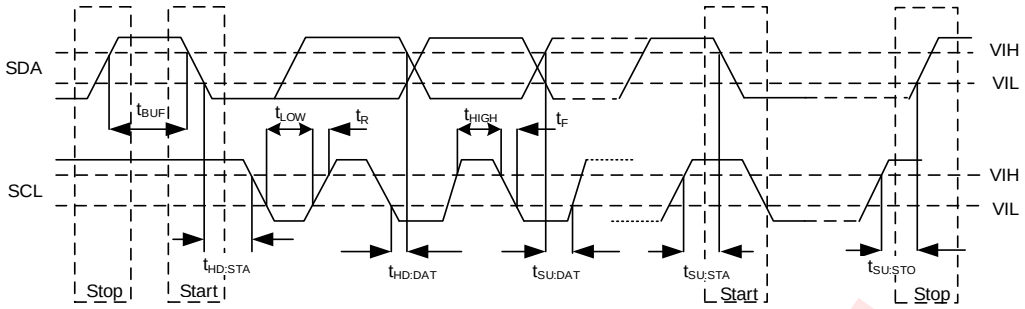


Figure 4 I²C Interface Timing

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Detailed Functional Description

Overview

AW21209 is a programmable 9-channel LED driver specifically designed for mobile devices. Device includes charge pump, high-side current sources, I²C interface, program execution engines and temperature compensation. Each channel supports an 8-bit current setting, with a maximum output current of 25.5mA, each channel features 12-bit pwm control resolution. Device startup timing is showed below.

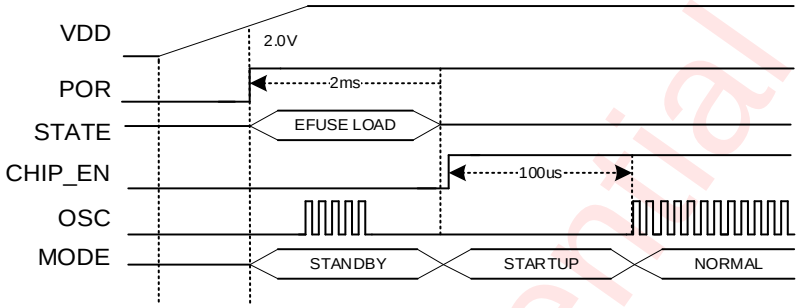


Figure 5 startup timing

Dimming

Analog Dimming

Analog dimming of AW21209 is controlled by individual 8-bit LED CURRENT register setting.

Table 1 DX_CURRENT Register (26H-2EH)

8-BIT CURRENT register		Iout
Binary	Decimal	mA
0000_0000	0	0.0
0000_0001	1	0.1
0000_0010	2	0.2
0000_0011	3	0.3
---	---	---
1010_1111	175	17.5
----	---	---
1111_1110	254	25.4
1111_1111	255	25.5

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

PWM Dimming

AW21209 has 9 LED ports, each of which is controlled by an independent PWM output register (16H - 1EH). When PWM is not FF, duty cycle is (PWM/256)*100%; When PWM is FF, duty cycle is 100%.

The system clock operates at 10.24 MHz and can be configured with the division factor and PWM mode through the PWM_CFG register (80H). The specific configuration is as follows:

Table 2 PWM frequency

PWM_DIV	0	1	2	3	4	5	6	7
pwmclk	10.24MHz	5.12MHz	2.56MHz	1.28MHz	640KHz	320KHz	160KHz	80KHz
9bit+3Dither	20KHz	10KHz	5KHz	2.5KHz	1.25KHz	625Hz	312.5Hz	156.25Hz
12bit	2.5KHZ	1.25KHz	625Hz	312.5Hz	156.25Hz	78.13Hz	39.06Hz	19.53Hz

1. Master Fader Modulation Function

AW21209 is equipped with master fader control, which allows the user to fade in or fade out multiple LEDs by writing to only one register. This is a useful function to minimize serial-bus traffic between the MCU and the AW21209. Users can choose whether to enable the master fader modulation according to their needs. Once enabled, each LED port has three master fader options to choose from.

2. Log Compensation Function

Human eyes are more sensitive to dim light. To make the brightness changes more linear to the human eye, device has added a log modulation function. This function can be enabled through the LOG_EN (06H - 0EH) signal to implement exponential modulation. The implementation is divided into eight segments for precise control.

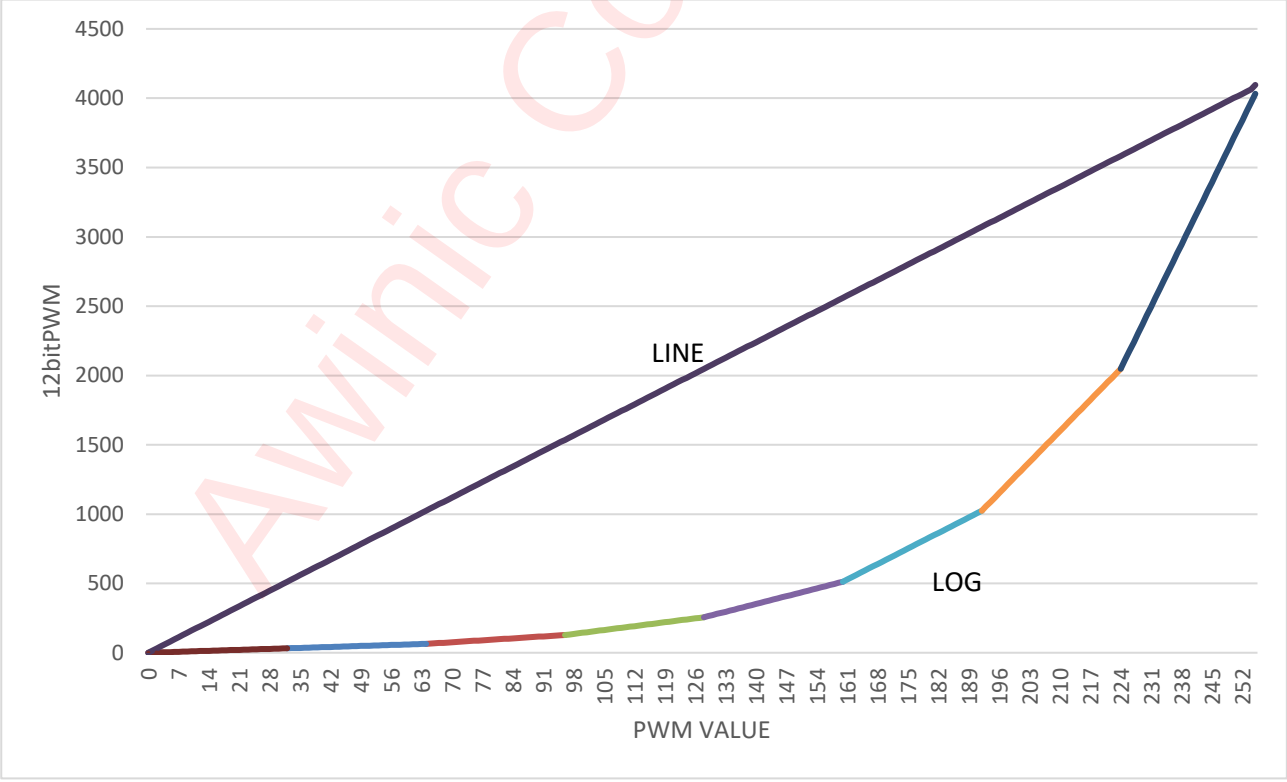


Figure 6 Linear and Logarithmic dimming curves

3. Temperature Compensation Function

Temperature variations can cause changes in LED brightness and color purity, leading to unexpected color

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

deviations. To address this, the AW21209 integrates an ADC (Analog-to-Digital Converter) to monitor temperature and regulate the PWM (Pulse Width Modulation) output duty cycle through temperature compensation, thereby achieving light compensation. Additionally, it supports external temperature sensors to capture temperature changes and implement thermal compensation effects. Users can select the slope of the temperature compensation, enabling brightness compensation in different scenarios. The temperature modulation range is from -39℃ to 89℃. To provide more margin for better brightness compensation, it is essential to ensure that the PWM output duty cycle does not exceed 50% at 25℃.The compensation calculation formula is as follows:

$$PWM_F = \frac{[PWM_S - (25\,℃ - T) * CorrectionFactor * PWM_S]}{2}$$

Among them, PWMs stands for the PWM configuration duty cycle register DX_PWM, T represents the temperature at the ADC testing point, and Correction Factor represents the temperature compensation slope (which can be freely configured by the).

Table 3 DX_TEMP_COMP Register (06H-0EH)

Name	Bits	Description
TEM_COMP	4:0	Temperature Compensation Slope Selection. Default Temperature: 25℃. Default: Disabled. 0_0000: Disable temperature compensation function; 1_1111：enable，1℃ adjust-1.5%； 1_1110：enable，1℃ adjust-1.4%； …… 1_0001：enable，1℃ adjust-0.1%； 1_0000：enable，1℃ adjust 0%； 0_0001：enable，1℃ adjust0.1%； …… 0_1110：enable，1℃ adjust1.4%； 0_1111：enable，1℃ adjust1.5%；

4. Data Flow

Each channel supports three types of modulation: master fader, exponential compensation, and temperature compensation. These three modulation modes can be freely combined as needed. For details on the three modulation enable switch configurations, please refer to the table below.

Table 4 Modulation compensation implementation

LOG_EN	FADER_EN	TEMP_EN	OUTPUT
0	0	0	pwm_value
1	0	0	Log(pwm_value)
0	1	0	Master_fader*pwm_value

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

1	1	0	Log(master_fader)*pwm_value
0	0	1	Temp(pwm_value)
1	0	1	Temp(Log(pwm_value))
0	1	1	Temp(master_fader*pwm_value)
1	1	1	Temp(Log(master_fader)*pwm_value)

Note:

1. pwm_value is DX_PWM register configuration value;
2. log(pwm_value) represents the value after applying logarithmic compensation to pwm_value;
3. temp(pwm_value) represents the value after applying temperature compensation to pwm_value;
4. master_fader * pwm_value refers to the modulation of pwm_value by master_fader, where the product retains the high 12 bits.

Clock

The clock is a core component for the normal operation of the chip. This chip provides two clock sources: a 10.24MHz high-speed clock and a 32kHz low-speed clock, which meet the requirements of different functional modules.

High-speed Clock

The 10.24MHz high-speed clock is generated by the internal oscillator (OSC) and provides a stable clock signal for the LED control module and the charge pump. After normal power-on, the internal OSC will automatically start and generate the 10.24MHz clock to ensure the normal operation of the chip's core functional modules. When the chip enters Power Save mode and the user configures the use of the external clock, the 10.24MHz clock will stop working to further reduce power consumption. It is important to note that when using the external clock and entering Power Save mode, the external clock must remain active; otherwise, the chip will not be able to wake up normally from Power Save mode.

Low-speed Clock

The low-speed clock is used for the instruction execution engine, which fetches and executes instructions stored in the internal SRAM to generate the desired lighting effects. The chip defaults to using the external clock on the CLK pin. Users can switch between the external clock and the internally derived 32.768kHz clock by configuring the CLK_SEL register.

After power-on and enabling the CHIP_EN bit in the CONFIG register, the chip requires a minimum of 32μs to detect the external clock. The AW21209 can detect external clock signals with frequencies above 5kHz.If the external clock stops or its frequency drops below 5kHz, the chip will determine that the external clock is invalid.

Switching between the internal and external clocks requires a delay of at least 153μs for the change to take effect. To maximize power savings in Power Save mode, users are recommended to configure the chip to use the external clock. When the chip is in Power Save mode and using the external clock, the 10.24MHz clock will be disabled. It is crucial that the external clock remains active during Power Save mode to ensure proper wake-up functionality. Failure to maintain the external clock during Power Save mode may result in the chip being unable to wake up correctly.

Note： The EXEC register, MODE register, PWM register, PC register and MASTER_FADER register are synchronized to the clock. Therefore, they do not support the I²C bus's auto-increment write operation. After configuring these registers via the I²C bus, a delay of four clock cycles is required for the settings to take effect. This delay ensures proper synchronization with the clock and guarantees stable operation of the chip. When

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

using an external clock for AW21209, if no clock is provided to the CLK pin before the CHIP_EN configuration, it takes about 200μs for clock detection/clock switching/clock domain synchronization.

Charge Pump

Overview

The AW21209 is equipped with a built-in charge pump that supports 1x and 1.5x modes. The 1.5x mode can stabilize the output voltage at 4.5V in a single lithium battery system through a switched capacitor charge pump and linear regulation technology. During the charging phase, VDD charges Cfly1 and Cfly2, and during the pumping phase, the stored charges in Cfly1 and Cfly2 are transferred to VOUT. Compared with traditional switched capacitor charge pumps, AW21209 can achieve precise voltage output by adjusting the on resistance of the transfer transistor, while also having higher conversion efficiency.

Output Resistance

The model of the charge pump during boosting is shown below.

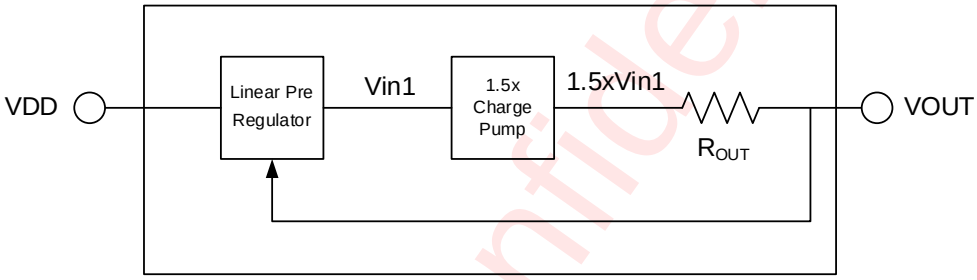


Figure 7 Charge Pump Output Resistance Model

The charge pump model is equipped with a linear pre adjustment block, a 1.5 x voltage multiplier charge pump, and an output resistor (R_{OUT}). The typical value of ROUT is 3.5 Ω, which varies depending on the switching frequency, input voltage, flying capacitor capacitance, switch internal resistances, and capacitor ESR. The system compensates for the voltage drop caused by different load currents on R_{OUT} by adjusting Vin1, thereby stabilizing VOUT at 4.5V. As load current increases, the regulator reduces its internal voltage differential, elevating Vin1 proportionally to preserve output stability. In this state, the system operates as a closed-loop regulator until reaching current saturation-defined when Vin1 equals input voltage and the regulator enters dropout condition. Beyond this threshold, operation transitions to open-loop charge pump behavior where output voltage decreases linearly with current according to the relationship :

$V_{OUT} = 1.5 \times VDD - I_{OUT} \times R_{OUT}$, maintaining predictable performance while sacrificing regulation capability.

LED Forward Voltage Monitoring

When the charge-pump automatic mode selection is enabled, voltages over LED drivers D1 to D6 are monitored. (Note: Power input for current source outputs D7, D8 and D9 are internally connected to the VDD pin.) Driver saturation monitor does not have a fixed voltage limit, since saturation voltage is a function of temperature and current.

Gain Change Hysteresis

The charge-pump gain control system employs digital filtering to suppress supply voltage disturbances (such as transient voltage fluctuations during GSM bursts) from inducing unwarranted gain adjustments, while incorporating hysteresis to eliminate periodic gain variations that may arise from LED driver interactions or voltage drops inherent to 1x charge-pump operation. This dual-protection architecture features user-adjustable hysteresis thresholds, with factory-programmable default settings, enabling flexible optimization where hysteresis parameters can be minimized to near-zero levels or calibrated to precise operational requirements across various application environments. LED forward voltage monitoring and gain control block diagram is

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

shown below.

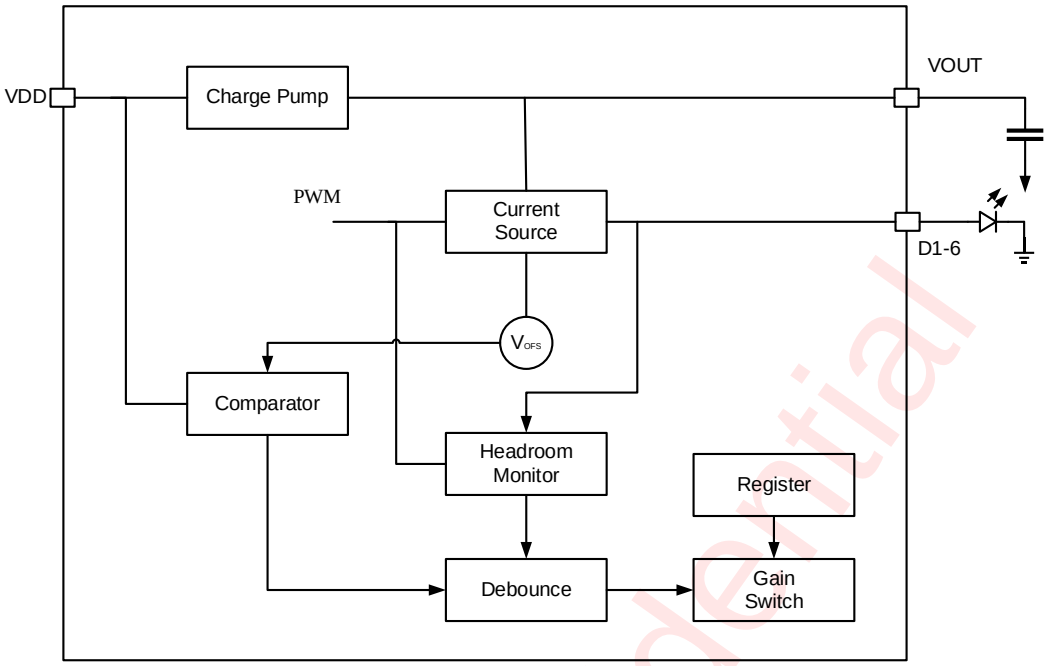


Figure 8 Forward Voltage Monitoring and Gain Control Block

Controlled charge pump

The charge pump operation is governed by two CP_MODE configuration bits (bits[3:2]) in the MISC register at address 36H. When both bits are set to low logic level (00), the charge pump remains disabled with its output voltage internally pulled down through a 300kΩ resistor (typical). Any non-zero combination of these bits activates the charge pump functionality, requiring a 100us startup period when transitioning from disabled state. The device supports forced bypass mode for direct battery voltage connection to current sources, while the 1.5x boost mode elevates output voltage to 4.5V nominal through charge pump operation. In automatic mode, charge-pump operation mode is determined by saturation of constant current drivers and internal timer as shown below.

In automatic gain change mode, the charge pump is switched to bypass mode (1×), when LEDs are inactive for over 50 ms.

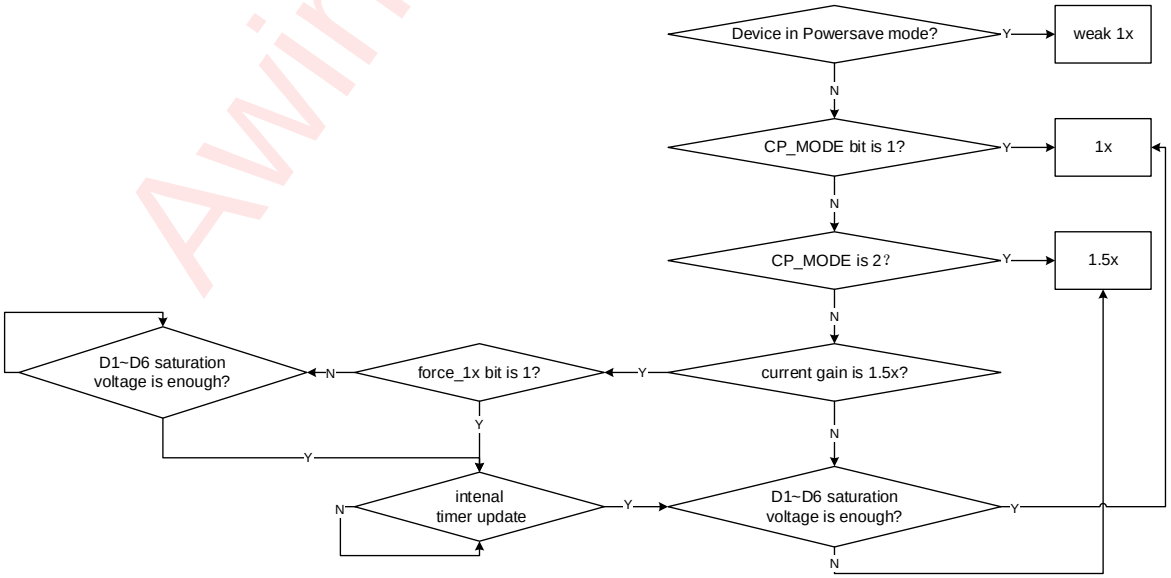


Figure 9 CP mode determination

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Power save Mode

Automatic Power save Mode

Automatic power-save mode is controlled by POWERSAVE_EN bit in register address 36H is 1.

In automatic power-save mode, the charge pump transitions to a low-current weak 1× mode configuration. System behavior varies depending on clock source selection: when in External Clock Mode, Charge pump protection circuitry maintains active state while All analog subsystems except protection circuits are powered down; when in Internal Clock Mode, Charge pump and LED driver circuits are disabled while digital control logic remains operational to maintain system continuity.

The architecture implements a current-limited passive keep-alive circuitry that maintains output voltage at battery potential during power conservation states. Automatic power-save activation occurs when all LED driver outputs exhibit PWM signal inactivity for a continuous 50ms interval.

To prevent transient power-state transitions during program execution, the system incorporates an instruction pipeline analyzer. This feature continuously monitors execution engines 1-3, initiating power-save entry only when sustained PWM inactivity (>50ms) is detected across all LED driver outputs. It should be noted that the engine judges 50ms by counting with the clock used for program execution. If an external clock of 64kHz is used, then the 50ms here will become 25ms.

In power save mode program execution continues uninterrupted. When an instruction that requires PWM activity is executed, a fast internal-startup sequence is started automatically. If the registers related to PWM (02H/03H/06~1EH/40H/48~4AH) are accessed, the device can exit the power-saving mode.

PWM Power save Mode

PWM cycle power-save mode is enabled when register 36 bit [2] PWM_PS_EN is set to 1. The PWM Power-Save Mode (PSM) optimizes energy efficiency by deactivating analog circuitry including the Charge Pump and LED analog circuits during the PWM cycle's off-phase, while still being operable during program execution.

Unlike the Automatic Power-Save Mode that engages during PWM inactivity, this advanced mode achieves energy conservation within active PWM cycles.

When implemented with a D9-output LED operating at 50% PWM duty cycle and 5mA drive current, the PWM-PSM maintains equivalent LED brightness while significantly reducing system input current: consumption decreases to approximately 50uA during LED-off phases and stabilizes around 200uA when charge-pump-powered outputs are active, demonstrating effective power management without compromising output performance.

Protection Features

Thermal Shutdown (TSD)

The device features an integrated thermal shutdown protection system designed to prevent overheating damage. This system activates the LED Thermal Shutdown protocol when the junction temperature reaches 150°C and automatically resumes normal operation once the temperature decreases to 130°C. Primary activation is typically caused by excessive power dissipation, which may result from elevated input voltages and/or output currents combined with inadequate thermal management. During protective operation, thermal cycling phenomena may occur, characterized by pulsed voltage output as the system alternates between shutdown (near-zero internal power dissipation), cooling recovery, and reactivation. These cycles continue until ambient temperature reduction, input voltage optimization, current load reduction, or enhanced thermal dissipation measures are implemented. To permanently resolve thermal cycling under specified operating conditions, improved heat sinking solutions are required to achieve appropriate thermal equilibrium in relation to the device's power dissipation profile.

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Power-On Reset (POR)

The device has internal comparators that monitor the voltages at V_{IN}. When V_{IN} is below 1.6V, reset is active and the device is in the POWERDOWN state.

Error Detection

The device has built-in LED error detection through ADC controller. Error detection does not only detect open and short circuit, but provides an opportunity to measure the VF of the LEDs. The test event is activated by a serial interface write to TEST_CTRL(41H), and the results can be read to TEST_ADC_DATA(42H) through the serial interface during the next cycle. This feature can also be addressed to measure the voltage on VDD and VOUT pin. Typical example usage includes monitoring battery voltage or using INT pin as a light sensor interface.

Multifunctional I/O

The device has 3 I/Os, includes TRIG/INT/GPO.

TRIG is only controlled by ENGING instructions, and the pad is open-drain, which means it needs a pull-up resistor.

INT is interrupt pad in default, but it can also change to a general-output pad. User need to set INT_CONF bit in IO_CTRL(3BH).

GPO is a general-output pad, and its output data is controlled by GPO bit in IO_CTRL(3BH), This feature is only applicable to AW21209FOR.

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Device Work Mode

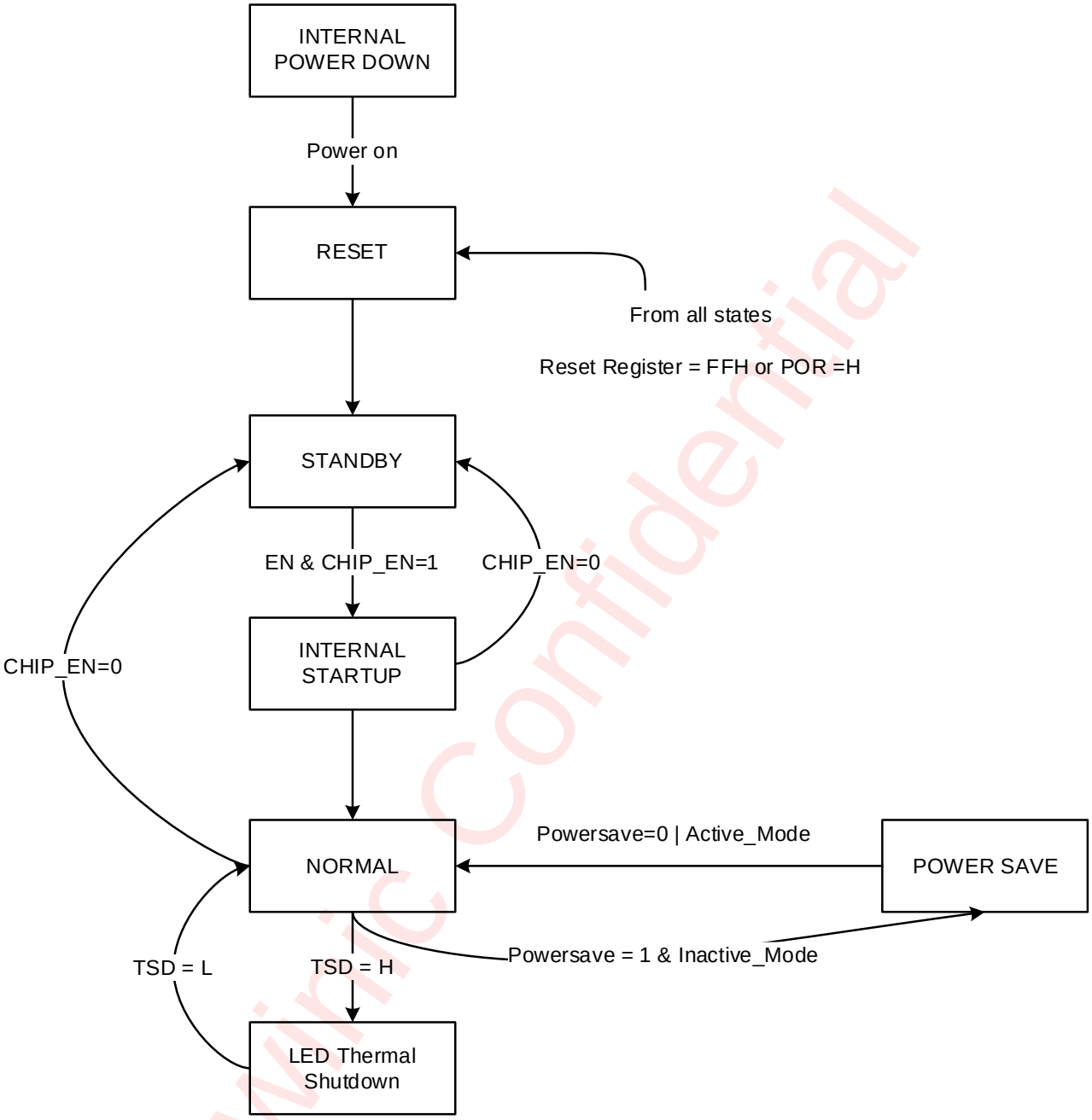


Figure 10 operating mode transition

RESET: In RESET mode, all internal registers are restored to their default states. The system enters RESET mode upon writing FFh to the Reset Register (3DH) or when the internal Power-On Reset (POR) circuit is activated, which occurs during initial chip power-up or when the supply voltage VDD drops below the typical threshold of 1.6V. Following VDD restoration above 2.0V (typ.), the POR function disengages, automatically transitioning the device into STANDBY mode. By default, the CHIP_EN control bit maintains a low logic level after any POR sequence completion.

STANDBY: The device enters STANDBY mode when either the CHIP_EN register bit or the EN pin is set to LOW while the Reset signal remains inactive, initiating a low-power state where all internal circuitry is deactivated. During this mode, most register settings remain writable provided the EN pin is subsequently asserted HIGH, enabling configuration updates that become operational immediately upon exiting standby (refer to the Register Details section for specific register behavior and configuration timing requirements).

STARTUP: When CHIP_EN bit is written high and EN pin is high, the INTERNAL STARTUP SEQUENCE

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

powers up all the needed internal blocks (VREF, bias, oscillator etc.). Startup delay is 100 μs.

NORMAL: During NORMAL mode the user controls the chip using the Control Registers.

POWER SAVE: In POWER-SAVE mode analog blocks are disabled to minimize power consumption. See Automatic Power-Save Mode section for further information.

LED Thermal Shutdown: In LED Thermal Shutdown mode, Charge Pump and LED output circuits will be disabled.

Device Program

After power-on, the chip will perform initialization. The user must wait for 2ms before configuring Bit 6 of the CONFIG register (CHIP_EN) to enable the chip. Only after the chip is enabled can the user configure other registers via the I²C bus. If a global soft reset is required, the user can configure the SOFT_RESET register with the value 0xFF to trigger the soft reset. After the soft reset, the chip will restore all registers to their default values. Similar to the power-on process, the user must wait for 2ms after the soft reset is completed, then reconfigure CHIP_EN before continuing to configure other registers via the I²C bus.

General I²C Operation

This device supports the I²C serial bus and data transmission protocol in fast mode at 400kHz and super-fast mode at 1000kHz. This device operates as a slave on the I²C bus. Connections to the bus are made via the open-drain I/O pins SCL and SDA. The pull-up resistor can be selected in the range of 1k~10kΩ and the typical value is 4.7kΩ. This device can support different high level (1.2V~5V) of this I²C interface.

Device Address

Table 5 I²C Device Address Configuration

AD1 Connection	AD0 Connection	Device Address
GND	GND	0x32
GND	VDD	0x33
VDD	GND	0x34
VDD	VDD	0x35

The I²C device address of AW21209 depends on the status of pins AD0 and AD1. Connecting pin AD0 or AD1 to GND or VCC will change the device address as showed in table above.

Data Validation

When SCL is high level, SDA level must be stable. SDA can be changed only when SCL is low level.

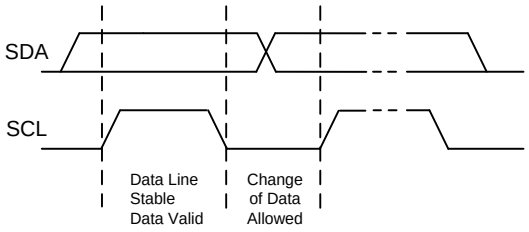


Figure 11 Data Validation Diagram

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

I²C Start/Stop

I²C start: SDA changes from high level to low level when SCL is high level.

I²C stop: SDA changes from low level to high level when SCL is high level.

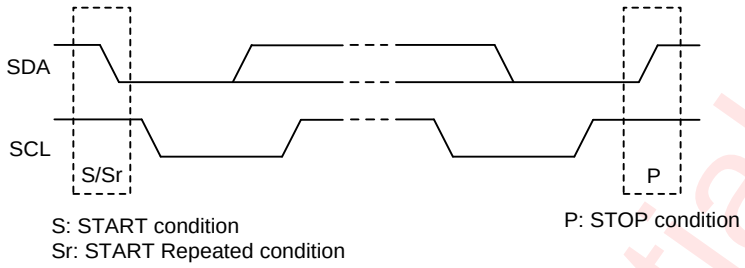


Figure 12 I²C Start/Stop Condition Timing

Acknowledge(ACK)

ACK means the successful transfer of I²C bus data. After master sends an 8-bit data, SDA must be released; SDA is pulled to GND by slave device when slave acknowledges.

When master reads, slave device sends 8-bit data, releases the SDA and waits for ACK from master. If ACK is send and I²C stop is not send by master, slave device sends the next data. If ACK is not send by master, slave device stops to send data and waits for I²C stop.

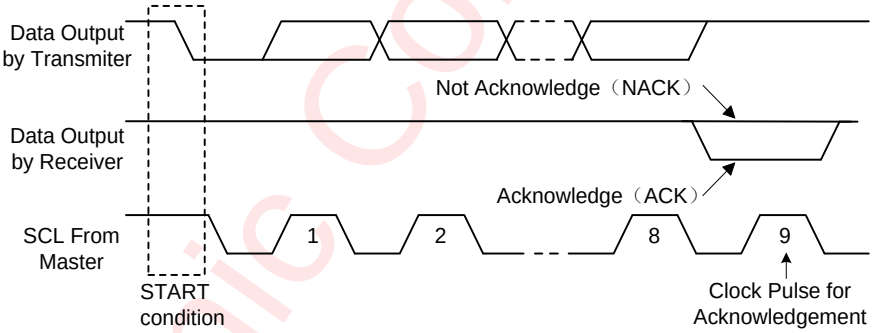


Figure 13 I²C ACK Timing

Write Cycle

One data bit is transferred during each clock pulse. Data is sampled during the high state of the serial clock (SCL). Consequently, throughout the clock's high period, the data should remain stable. Any changes on the SDA line during the high state of the SCL and in the middle of a transaction, aborts the current transaction. New data should be sent during the low SCL state. This protocol allows a single data line to transfer both command/control information and data using the synchronous serial clock.

Each data transaction is composed of a start condition, a number of byte transfers (set by the software) and a stop condition to terminate the transaction. Every byte written to the SDA bus must be 8 bits long and is transferred with the most significant bit first. After each byte, an Acknowledge signal must follow.

In a write process, the following steps should be followed:

a) Master device generates START condition. The "START" signal is generated by lowering the SDA signal while the SCL signal is high.

b) Master device sends slave address (7-bit) and the data direction bit (R/W = 0).

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

c) Slave device sends acknowledge signal if the slave address is correct.
d) Master sends control register address (8-bit)
e) Slave sends acknowledge signal
f) Master sends data byte to be written to the addressed register
g) Slave sends acknowledge signal
h) If master will send further data bytes, the control register address will be incremented by one after acknowledge signal (repeat step f and g)
i) Master generates STOP condition to indicate write cycle end

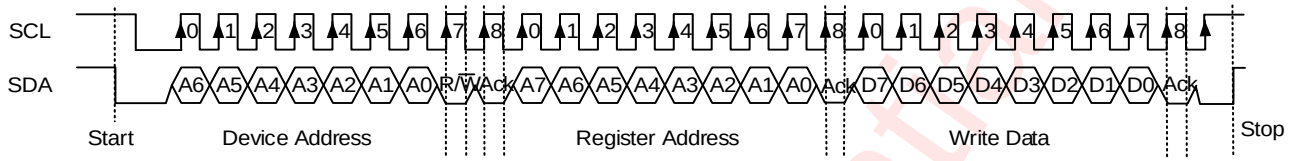


Figure 14 I²C Write Byte Cycle

Read Cycle

In a read cycle, the following steps should be followed:

a) Master device generates START condition
b) Master device sends slave address (7-bit) and the data direction bit (R/W = 0).
c) Slave device sends acknowledge signal if the slave address is correct.
d) Master sends control register address (8-bit)
e) Slave sends acknowledge signal
f) Master generates STOP condition followed with START condition or REPEAT START condition
g) Master device sends slave address (7-bit) and the data direction bit (R/W = 1).
h) Slave device sends acknowledge signal if the slave address is correct.
i) Slave sends data byte from addressed register.
j) If the master device sends acknowledge signal, the slave device will increase the control register address by one, then send the next data from the new addressed register.
k) If the master device generates STOP condition, the read cycle is ended.

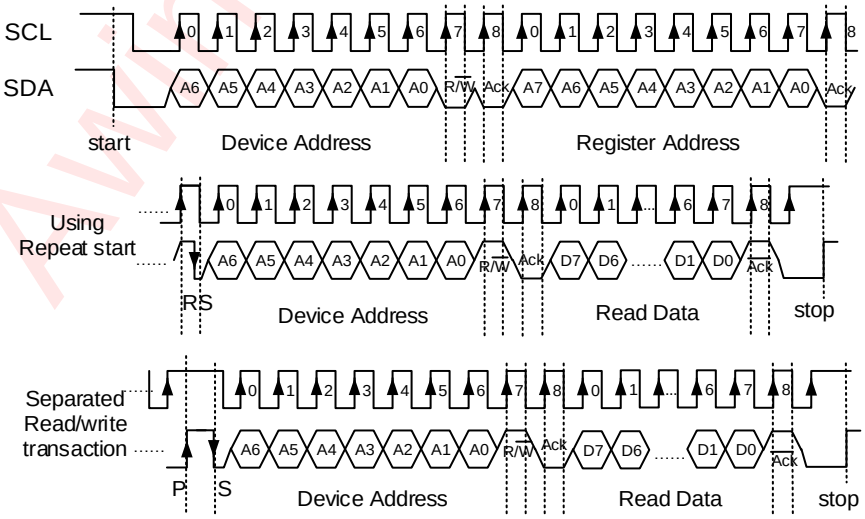


Figure 15 I²C Read Byte Cycle

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Execution Engine Programming

The AW21209 enables users to customize the dimming and flashing sequences of LEDs with flexibility. Each LED can be individually controlled through the serial bus, and LED drivers can also be grouped for programmed flashing patterns.

Equipped with three independent program execution engines, the AW21209 supports the creation of three distinct programmable LED banks that can be assigned to different LED drivers. Each bank can accommodate between 1 and 9 LED driver outputs.

SRAM MEMORY

The AW21209 has internal SRAM for the three LED engines. SRAM can contain up to 96 16-bit instructions (addresses 0 through 95). Memory allocation among the three LED engines is done dynamically, so that each LED engine has a separate start address and program counter (PC) that are set in the ENGINE x PROG START registers (addresses 4Ch, 4Dh, 4Eh) and ENGINE x PC registers (addresses 37h, 38h, 39h). This allows flexible memory allocation among the LED engines, and multiple engines can recall the same memory address. The PC uses relative memory addressing; when the PC is zero the engine is executing an instruction at its start address.

The register PROG MEM PAGE SEL(4Fh) select the SRAM memory page and all the SRAM memory is divided into 6 pages, each page address is from 50h to 6fh. Engines must be set to load the program mode (register 01h) before writing the SRAM.

PROGRAM EXECUTION ENGINE OPERATION MODES

Operation modes are defined in register ENGINE CTRL(01h). Bit[5:4] in this register define the engine1 operation mode, bit[3:2] define the engine2 operation mode and bit[1:0] define the engine3 operation mode. Each engine (1, 2, 3) operation mode can be configured separately.

• ***Disabled***

Each engine can be configured to disabled mode. In this mode, the engine program execution is stop and the program execution module of current engine is reset, but the content of SRAM is not affected. The PWM values are frozen and PC values are reset for current engines.

The reset time lasts approximately 1ms, so after switch to disabled mode, it is necessary to wait at least 1ms or check whether the engine busy bit(in register STATUS) is set before switching to another mode.

• ***Load program***

Writing to program memory is allowed only when one engine is in load program operation mode. All the three engines are stop running while one or more engines are in load program mode. PWM values are frozen and PC values are reset. Load program mode can be entered from the disabled mode only. Entering load program mode from the run program or halt mode is not allowed.

• ***Run program***

Run program mode executes the instructions stored in the program memory. Register CONFIG. define how the program is executed (hold, step, free run or execute once). Register ENGINE x PC and ENG x PROG START ADDR define where the program begin to execute.

• ***Halt***

Instruction execution aborts immediately, and engine operation halts. The PWM values and PC remains unchanged in this mode.

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

PROGRAM EXECUTION ENGINE MODES

Program execution modes are defined in register CONFIG (00h). Bit[5:4] in this register define the engine1 execution mode, bit[3:2] define the engine2 execution mode and bit[1:0] define the engine3 execution mode. Each engine (1, 2, 3) operation mode can be configured separately.

• ***Hold***

Hold causes the execution engine to finish the current instruction and then stop. Program counter (PC) can be read or written only in this mode.

• ***Step***

Execute the instruction at the location pointed by the PC, increment the PC by one and then reset current engine's program execution mode to hold.

• ***Free run***

Start program execution from the location pointed by the PC.

• ***Execute once***

Execute the instruction pointed by the current PC value and reset current engine's program execution mode to hold. The difference between step and execute once is that execute once does not increment the PC.

INSTRUCTION INTRODUCTION

For the three independent programmable execution engines of AW21209, users write their program blocks (consisting of many instructions) into the SRAM. Note that in order to access program memory the operation mode needs to be *load program*, at least for one of the three program execution engines. And then users can config operation mode to *run program,* these instructions will start to executed according to the program execution modes. The program execution can use the internal clock of 32.768kHz or the external clock. Next, these instructions will be introduced in detail.

LED Driver Instructions

LED Driver instruction set is listed in the following tables:

Table 6 AW21209 LED Driver Instructions

Inst	Bit [15]	Bit [14]	Bit [13]	Bit [12]	Bit [11]	Bit [10]	Bit [9]	Bit [8]	Bit [7]	Bit [6]	Bit [5]	Bit [4]	Bit [3]	Bit [2]	Bit [1]	Bit [0]
ramp(1)	0	prescale	step time					sign	number of increments							
ramp(2)	1	0	0	0	0	1	0	0	0	0	pre-scale	sign	step time		no. of increments	
set_pwm(1)	0	1	0	0	0	0	0	0	PWM value							
set_pwm(2)	1	0	0	0	0	1	0	0	0	1	1	0	0	0	PWM value	
wait	0	prescale	time					0	0	0	0	0	0	0	0	0

(1) This opcode is used with numerical operands.

(2) This opcode is used with variables.

Ramp

This instruction facilitates a smooth transition between one PWM value and another on the D1 to D9 outputs, effectively generating ramps with either a negative or positive slope. The AW21209 supports programming both extremely fast and extremely slow ramps.

The ramp instruction facilitates the generation of a PWM ramp, utilizing the effective PWM value as the starting point. At each ramp step, the output is incremented or decremented by one unit unless the number of increments is set to zero. The time span for a single ramp step is determined by the prescale bit [14] and the

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

step time bits [13:9]. Specifically, when the prescale is configured to 0, the cycle time is set to 0.49 ms, and when the prescale is configured to 1, the cycle time is set to 15.6 ms. Consequently, the minimum time span for a single step is 0.49 ms (prescale × step time span = 0.49 ms × 1), while the maximum time span is 15.6 ms × 31 = 484 ms.

The number of increments specifies the total steps taken during a single ramp instruction. The maximum increment value is 255, representing a progression from the zero value to the maximum value. If the PWM reaches its minimum or maximum value (0 or 255) during the ramp instruction, the instruction will continue to completion regardless of saturation. The ramp instruction governs the eight most-significant bits (MSB) of the PWM values, while the remaining bits are internally interpolated as intermediate ramp values to ensure a smoother transition. This functionality allows the ramp instruction to serve as both a ramp and a wait instruction. Note: The ramp instruction functions as a wait instruction when the increment bits [7:0] are set to zero.

Programming ramps using variables closely mirrors the process of programming ramps with numerical operands. The sole distinction lies in capturing the step time and number of increments from variable registers at the onset of instruction execution. Any updates to the variables after the initiation of instruction execution will not influence the ongoing process. Furthermore, during each ramp step, the output is incremented or decremented by one unit, provided the increment value is not zero. The duration of a single step is determined by the prescale and step time bits. Step time can be specified using variables A, B, C, or D. Variables A, B, and C are configured via the ld instruction, whereas Variable D serves as a global variable that can be set by writing to the VARIABLE register (address 3C). Additionally, the LED TEST ADC register (address 42) can also serve as a source for Variable D. It should be noted that Variable D is the sole local variable accessible for reading across the serial bus. Naturally, the variable stored at address 3Ch is also readable (and writable).

NAME	VALUE(d)	DESCRIPTION
prescale	0	Divides master clock (32.7 kHz) by 16 = 2048 Hz, 0.488 ms cycle time
	1	Divides master clock (32.7 kHz) by 512 = 64 Hz, 15.625 ms cycle time
sign	0	Increase PWM output
	1	Decrease PWM output
step time(1)	1-31	One ramp increment done in (step time) x (clock after prescale)
increments(1)	0-255	The number of increment/decrement cycles. Note: Value 0 takes the same time as increment by 1, but it is the wait instruction
step time(2)	0-3	One ramp increment done in (step time) × (prescale). Step time is loaded with the value (5 LSB bits) of the variable defined below. 0 = local variable A 1 = local variable B 2 = global variable C 3 = register address 3CH variable D value, or register address 42H value. The value of the variable should be from 00001b to 11111b (1d to 31d) for correct operation
increments(2)	0-3	The number of increment/decrement cycles. Value is taken from variable following defined: 0 = local variable A 1 = local variable B 2 = global variable C 3 = register address 3CH variable D value, or register address 42H value.

(1) Valid for numerical operands.

(2) Valid for variables.

Ramp Instruction Application Example

Suppose that the LED dimming is controlled according to the linear scale and effective PWM value at the moment t = 0 is 140d (approximately 55%), as shown in Figure 25, and goal is to reach a PWM value of 145d (approximately 57%) at the moment t = 937.5ms. The parameters for the RAMP instruction are:

- Prescale = 1 → 15.625 ms cycle time

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

- Step time = 12 → step time span is 12 × 15.625 ms = 187.5 ms
- Sign = 0 → increase PWM output
- # of increments = 5 → take 5 steps

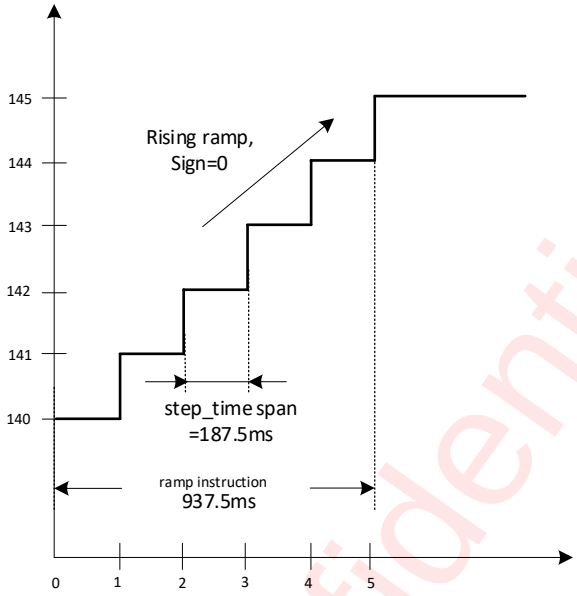


Figure 16 Example of ramp commands

Set_PWM

This instruction is designed to set the PWM value on outputs D1 through D9 without employing any ramping functionality. The PWM output value can be configured within the range of 0 to 255 using the PWM value bits [7:0]. Execution of this instruction requires sixteen clock cycles.

NAME	VALUE(d)	DESCRIPTION
PWM value (i)(1)	0-255	PWM output duty cycle 0 - 100%
variable (ii)(2)	0-3	0 = local variable A 1 = local variable B 2 = global variable C 3 = register address 3CH variable D value, or register address 42H value

(1) Valid for numerical operands.
(2) Valid for variables.

Wait

When a wait instruction is executed, the engine enters the wait state, and the PWM values on the outputs are held constant.

NAME	VALUE(d)	DESCRIPTION
prescale	0	Divide master clock (32.7 kHz) by 16 which means 0.488 ms cycle time.
	1	Divide master clock (32.7 kHz) by 512 which means 15.625 ms cycle time.
time	1-31	Total wait time is = (time) × (prescale). Maximum 484 ms, minimum 0.488 ms.

Clock

Clock frequency adjustment affects the instruction execution cycle, the ramp instruction ramp – up\ ramp – down time, and the wait instruction waiting time. An instruction is 16 bits, and its execution requires 16 clock cycles. The instruction execution can use an internal clock of 32.768 kHz, or an external clock of 32 kHz or 64 kHz.

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

clock	instruction execution cycle
internal clock of 32.768 kHz	16 * 1 / 32768Hz * 1000 = 0.488ms
external clock of 32 kHz	16 * 1 / 32000Hz * 1000 = 0.5ms
external clock of 64 kHz	16 * 1 / 64000Hz * 1000 = 0.25ms

Both the ramp and wait instructions have a prescale parameter, which is related to the clock cycle. Take the ramp instruction as an example.

clock	prescale	
internal clock of 32.768 kHz	0	Divides master clock (32.768kHz) by 16 = 2048 Hz, 0.488 ms cycle time
	1	Divides master clock (32.768kHz) by 512 = 64 Hz, 15.625 ms cycle time
external clock of 32 kHz	0	Divides master clock (32kHz) by 16 = 2000 Hz, 0.5 ms cycle time
	1	Divides master clock (32kHz) by 512 = 62.5 Hz, 16 ms cycle time
external clock of 64 kHz	0	Divides master clock (64kHz) by 16 = 4000 Hz, 0.25 ms cycle time
	1	Divides master clock (64kHz) by 512 = 125 Hz, 8 ms cycle time

LED Mapping Instructions

LED Mapping instruction set is listed in the following tables：

Table 7 AW21209 LED Mapping Instructions

Inst	Bit [15]	Bit [14]	Bit [13]	Bit [12]	Bit [11]	Bit [10]	Bit [9]	Bit [8]	Bit [7]	Bit [6]	Bit [5]	Bit [4]	Bit [3]	Bit [2]	Bit [1]	Bit [0]
mux_ld_start	1	0	0	1	1	1	1	0	0	SRAM address 0-95						
mux_map_start	1	0	0	1	1	1	0	0	0	SRAM address 0-95						
mux_ld_end	1	0	0	1	1	1	0	0	1	SRAM address 0-95						
mux_sel	1	0	0	1	1	1	0	1	0	LED select						
mux_clr	1	0	0	1	1	1	0	1	0	0	0	0	0	0	0	0
mux_map_next	1	0	0	1	1	1	0	1	1	0	0	0	0	0	0	0
mux_map_prev	1	0	0	1	1	1	0	1	1	1	0	0	0	0	0	0
mux_ld_next	1	0	0	1	1	1	0	1	1	0	0	0	0	0	0	1
mux_ld_prev	1	0	0	1	1	1	0	1	1	1	0	0	0	0	0	1
mux_ld_addr	1	0	0	1	1	1	1	1	0	SRAM address 0-95						
mux_map_addr	1	0	0	1	1	1	1	1	1	SRAM address 0-95						

These instructions define the engine-to-LED mapping. The mapping information is stored in a table, which is stored in the SRAM (program memory of the AW21209). AW21209 has three program execution engines which can be mapped to 9 LED drivers or to one GPO pin. One engine can control one or multiple.

LED drivers. There are totally eleven instructions for the engine-to-LED-driver control: mux_ld_start, mux_map_start, mux_ld_end,mux_sel, mux_clr, mux_map_next, mux_map_prev, mux_ld_next, mux_ld_prev, mux_ld_addr and mux_map_addr.

Mapping table in SRAM.

	Bit [15]	Bit [14]	Bit [13]	Bit [12]	Bit [11]	Bit [10]	Bit [9]	Bit [8]	Bit [7]	Bit [6]	Bit [5]	Bit [4]	Bit [3]	Bit [2]	Bit [1]	Bit [0]
mapping table in SRAM	GPO(only AW21209 FOR)							D9	D8	D7	D6	D5	D4	D3	D2	D1

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

MUX_LD_START; MUX_LD_END

The instructions mux_ld_start and mux_ld_end are utilized to specify the memory location of the mapping table.

NAME	VALUE(d)	DESCRIPTION
SRAM address	0-95	Mapping table start/end address

MUX_MAP_START

Mux_map_start specifies the starting address of the mapping table in memory, and simultaneously activates (maps) the first row of the table.

NAME	VALUE(d)	DESCRIPTION
SRAM address	0-95	Mapping table start address

MUX_SEL

The mux_sel instruction enables the connection of exactly one LED driver (or the GPO pin) to a program execution engine. To connect multiple LEDs to a single engine, the mapping table is utilized. Once the mapping for an LED has been released, the PWM register value continues to control the LED brightness. If the mapping for the GPO pin is released, serial bus control assumes responsibility for managing the GPO state.

NAME	VALUE(d)	DESCRIPTION
LED select	0-16	0 = no drivers selected 1 = LED1 selected 2 = LED2 selected ... 9 = LED9 selected 16 = GPO, This feature is only applicable to AW21209FOR

MUX_CLR

Mux_clr instruction clears the mapping between the engine and the driver. Once the mapping for an LED has been released, the PWM register continues to control the LED brightness. If the mapping for the GPO pin is released, serial bus control assumes responsibility for managing the GPO state.

MUX_MAP_NEXT

This instruction activates the next row in the mapping table each time it is invoked. For instance, if the second row is currently active, invoking the mux_map_next instruction will activate the third row. If the end address of the mapping table is reached, activation will roll back to the start address of the mapping table upon the next invocation of the mux_map_next instruction. The engine does not push a new PWM value to the LED driver output until either the set_pwm or ramp instruction is executed. Once the mapping for an LED has been released, the value stored in the PWM register continues to control the LED brightness. If the mapping for the GPO pin is released, serial bus control assumes responsibility for managing the GPO state.

MUX_LD_NEXT

Similar to the mux_map_next instruction, this instruction only updates the index pointer to point to the next row in the mapping table. However, no actual mapping is established, and the connection between the engine and the LED driver remains unchanged.

MUX_MAP_PREV

This instruction activates the previous row in the mapping table each time it is invoked. For instance, if the third row is currently active, invoking the mux_map_prev instruction will activate the second row. If the start address of the mapping table is reached, activation will roll to the end address of the mapping table upon the next invocation of the mux_map_prev instruction. The engine does not push a new PWM value to the LED driver output until either the set_pwm or ramp instruction is executed. Once the mapping for an LED has been released, the value stored in the PWM register continues to control the LED brightness. If the mapping for the GPO pin is released, serial bus control assumes responsibility for managing the GPO state.

MUX_LD_PREV

Similar to the mux_map_prev instruction, this instruction only updates the index pointer to point to the previous

AWINIC Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

row in the mapping table. However, no actual mapping is established, and the connection between the engine and the LED driver remains unchanged.

MUX_MAP_ADDR

The mux_map_addr instruction sets the index pointer to point to the row in the mapping table defined by bits [6:0] and activates the specified row. The engine does not push a new PWM value to the LED driver output until either the set_pwm or ramp instruction is executed. Once the mapping for an LED has been released, the value stored in the PWM register continues to control the LED brightness. If the mapping for the GPO pin is released, serial bus control assumes responsibility for managing the GPO state.

NAME	VALUE(d)	DESCRIPTION
SRAM address	0-95	Any SRAM address containing mapping data.

MUX_LD_ADDR

The Mux_ld_addr instruction sets the index pointer to point to the row in the mapping table defined by bits [6:0], but the specified row is not activated.

NAME	VALUE(d)	DESCRIPTION
SRAM address	0-95	Any SRAM address containing mapping data.

Branch Instructions

LED Branch instruction set is listed in the following tables:

Table 8 AW21209 Branch Instructions

Inst	Bit [15]	Bit [14]	Bit [13]	Bit [12]	Bit [11]	Bit [10]	Bit [9]	Bit [8]	Bit [7]	Bit [6]	Bit [5]	Bit [4]	Bit [3]	Bit [2]	Bit [1]	Bit [0]
rst	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
branch(1)	1	0	1	loop count						step number						
branch(2)	1	0	0	0	0	1	1	step number							loop count	
int	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0
end	1	1	0	int	reset	0	0	0	0	0	0	0	0	0	0	0
trigger	1	1	1	wait for trigger						send a trigger						0
				ext. trig	X(3)	X(3)	E3	E2	E1	ext. trig	X(3)	X(3)	E3	E2	E1	
trig_clear	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
jne	1	0	0	0	1	0	0	Number of instructions to be skipped if the operation returns true					variable 1		variable 2	
jl	1	0	0	0	1	0	1	Number of instructions to be skipped if the operation returns true					variable 1		variable 2	
jge	1	0	0	0	1	1	0	Number of instructions to be skipped if the operation returns true					variable 1		variable 2	
je	1	0	0	0	1	1	1	Number of instructions to be skipped if the operation returns true					variable 1		variable 2	

(1) This opcode is used with numerical operands.
(2) This opcode is used with variables.
(3) X means do not care.

BRANCH

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

The branch instruction is primarily designed to repeat a specific segment of program code multiple times. This instruction loads the step number value into the program counter. The loop count parameter specifies the number of repetitions for the instructions within the loop. The AW21209 supports nested looping, meaning loops can be embedded within other loops. The maximum number of nested loops supported is 8. Execution of this instruction requires sixteen clock cycles.

NAME	VALUE(d)	DESCRIPTION
loop count(1)	0-63	The number of loops to be done. 0 means an infinite loop.
step number	0-95	The step number to be loaded to program counter.
loop count(2)	0-3	Selects the variable for loop count value. Loop count is loaded with the value of the variable defined below. 0 = local variable A 1 = local variable B 2 = global variable C 3 = register address 3CH variable D value, or register address 42H value

(1) Valid for numerical operands.
(2) Valid for variables.

INT

An interrupt is sent to the processor by pulling the INT pin low and setting the corresponding status bit to a high state. The interrupt can be cleared by reading the interrupt status bits in the STATUS register located at address 3A.

RST

Rst instruction resets Program Counter register (address 37H, 38H, or 39H) and continues executing the program from the program start address defined in 4C-4E. Instruction takes sixteen clock cycles. Note that default value for all program memory registers is 0000H, which is the rst instruction.

END

End program execution. Instruction takes sixteen clock cycles.

NAME	VALUE(d)	DESCRIPTION
int	0	No interrupt is sent. PWM register values remain intact.
	1	Reset program counter value to 0 and send interrupt to processor by pulling the INT pin down and setting corresponding status bit high to notify that program has ended. PWM register values remains intact. Interrupt can be cleared by reading interrupt bits in STATUS register at address 3A.
reset	0	Reset program counter value to 0 and hold. PWM register values remains intact.
	1	Reset program counter value to 0 and hold. PWM register values of the non-mapped drivers remains. PWM register values of the mapped drivers is set to 0000 0000. On completion of int instruction with this bit set to 1 the master fader registers are set to zero as follows: Program execution engine 1 sets MASTER FADER 1 (48H) to zero, engine 2 sets MASTER FADER 2 (49H) to zero and engine 3 sets MASTER FADER 3 (4AH) to zero.

TRIGGER and TRIG_CLEAR

The mechanisms of waiting for or sending triggers can be employed, for instance, to synchronize the operations among program execution engines. A send trigger instruction requires sixteen clock cycles, while waiting for a trigger takes at least sixteen clock cycles. The receiving engine retains the triggers that have been transmitted, and received triggers are cleared upon execution of the wait-for-trigger instruction. The wait-for-trigger instruction continues to execute until all predefined triggers have been received. (Note: multiple triggers can be defined within a single instruction.)

External trigger input signal must stay low for at least two clock cycles to be executed. Trigger output signal is

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

three clock cycles long. External trigger signal is active low; that is, when trigger is sent/received the pin is pulled to GND. Send external trigger is masked; that is, the device that has sent the trigger does not recognize it. If send and wait external trigger are used on the same instruction, the send external trigger is executed first, then the wait external trigger.

The trig_clear instruction clears pending triggers for a single execution engine. Use this instruction in each execution engine at the beginning of program execution to clear any pending triggers. Pending triggers are always cleared whenever the engine mode is in the disabled state or load program to SRAM.

NAME	VALUE(d)	DESCRIPTION
wait for trigger	0-31	Wait for trigger from the engine(s). Several triggers can be defined in the same instruction. Bit[7] engages engine 1, bit [8] engine 2, bit [9] engine 3 and bit [12] is for external trigger I/O.Bits [10] and [11] are not in use.
send a trigger	0-31	Send a trigger to the engine(s). Several triggers can be defined in the same instruction. Bit [1] engages engine 1, bit [2] engine 2, bit [3] engine 3 and bit [6] is for external trigger I/O. Bits [4] and [5] are not in use.

AW21209 can trigger the engine to run the lighting effect through the change of the TRIG pin level from high to low. Regarding the time from external triggering to the lighting effect response, the following calculation method can be referred to: The theoretical maximum value of this time is 44 clock cycles plus one PWM cycle. By adjusting the frequency division (no division) and mode (9 bit + 3 dither), the PWM cycle can reach the minimum value. The minimum value of the PWM cycle is: 1/20000Hz×1000 = 0.05 milliseconds. When the instruction cycle is calculated at 64 kHz, the theoretical maximum value = 0.7375 milliseconds.

The AW21209 instruction set includes the following conditional jump instructions: jne (jump if not equal); jge (jump if greater or equal); jl (jump if less); je (jump if equal). If the condition is true, a certain number of instructions are skipped (that is, the program jumps forward to a location relative to the present location). If condition is false, the next instruction is executed.

NAME	VALUE(d)	DESCRIPTION
number of instructions to be skipped if the operation returns true.	0-31	The number of instructions to be skipped when the statement is true. Note: value 0 means redundant code.
variable 1	0-3	Defines the variable to be used in the test: 0 = local variable A 1 = local variable B 2 = global variable C 3 = register address 3CH variable, or register address 42H value.
variable 2	0-3	Defines the variable to be used in the test: 0 = local variable A 1 = local variable B 2 = global variable C 3 = register address 3CH variable, or register address 42H value.

Arithmetic Instructions

LED Arithmetic instruction set is listed in the following tables:

Table 9 AW21209 Data Transfer And Arithmetic Instructions

Inst	Bit [15]	Bit [14]	Bit [13]	Bit [12]	Bit [11]	Bit [10]	Bit [9]	Bit [8]	Bit [7]	Bit [6]	Bit [5]	Bit [4]	Bit [3]	Bit [2]	Bit [1]	Bit [0]
ld	1	0	0	1	Target variable		0	0	8-bit value							
add(1)	1	0	0	1	Target		0	1	8-bit value							

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

					variable								
add(2)	1	0	0	1	Target variable	1	1	0	0	0	0	variable 1	variable 2
sub(1)	1	0	0	1	Target variable	1	0	8-bit value					
sub(2)	1	0	0	1	Target variable	1	1	0	0	0	1	variable 1	variable 2

(1) This opcode is used with numerical operands.
(2) This opcode is used with variables.

LD

This instruction is used to assign a value to a variable, overwriting the previous value stored in that variable. Each engine has two local variables, designated as A and B, while variable C serves as a global variable.

NAME	VALUE(d)	DESCRIPTION
target variable	0-2	0 = variable A 1 = variable B 2 = variable C
8-bit value	0-255	Variable value

ADD

The operator either adds an 8-bit value to the current value of the target variable or adds the value of Variable 1 (A, B, C, or D) to the value of Variable 2 (A, B, C, or D), storing the result in the register of Variable A, B, or C. Variables overflow from 255 to 0.

NAME	VALUE(d)	DESCRIPTION
8-bit value(1)	0-255	The value to be added.
target variable	0-2	0 = variable A 1 = variable B 2 = variable C
variable 1(2)	0-3	0 = local variable A 1 = local variable B 2 = global variable C 3 = register address 3CH variable, or register address 42H value.
variable 2(2)	0-3	0 = local variable A 1 = local variable B 2 = global variable C 3 = register address 3CH variable, or register address 42H value.

(1) Valid for numerical operands.
(2) Valid for variables.

SUB

The SUB operator either subtracts an 8-bit value from the current value of the target variable or subtracts the value of Variable 2 (A, B, C, or D) from the value of Variable 1 (A, B, C, or D), storing the result in the register of the target variable (A, B, or C). Variables overflow from 0 to 255.

NAME	VALUE(d)	DESCRIPTION
8-bit value(1)	0-255	The value to be subed.
target variable	0-2	0 = variable A 1 = variable B 2 = variable C
variable 1(2)	0-3	0 = local variable A 1 = local variable B

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

		2 = global variable C 3 = register address 3CH variable, or register address 42H value.
variable 2(2)	0-3	0 = local variable A 1 = local variable B 2 = global variable C 3 = register address 3CH variable, or register address 42H value.

(1) Valid for numerical operands.
(2) Valid for variables.

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

CONTROLLING LED OUTPUTS

REGISTERS DIRECT CONTROL

Registers direct control is active by default.The ports of the chip are functionally independent of each other, and a single LED can be lit up separately according to the application. To light up D1, you can follow the steps below:

1) Power on the chip, pull up the EN pin, and delay for 2 ms.
2) Set the value of the CHIP_EN[Bit6] in the CONFIG(00H) register to 1, and the chip enters the NORMAL mode.
3) Set the value of CP_MODE [Bit4-Bit3] in the MISC(36H) register to 1, and use the 1X mode.
4) Set the value of CLK_SEL[Bit1-Bit0] in the MISC(36H) register to 0, and use the external clock. If no external clock is available, the internal clock can be selected.
5) Set the D1_CURRENT(26H) register to FFH, and the current is 25.5 mA.
6) Set the D1_PWM(16H) register to FFH, and the duty cycle is 100%.

ENGINE CONTROL

The priority of the program execution engine is higher than that of the direct control register. Therefore, if the user has set the PWM register to a certain value, this value will be automatically overridden when the program execution engine controls the driver.The steps to output light effects using the program execution engine are as follows:

1) Design the light effect code. The engine controls D1-D9 to achieve autonomous breathing. Engine 1 controls D1, D4, and D7; Engine 2 controls D2, D5, and D8; Engine 3 controls D3, D6, and D9. Please note that the light effect here is only for demonstration. AW21209 has a rich instruction set and can generate various required light effects.
2) Power on the chip, pull up the EN pin, and delay for 2ms.
3) Set the value of the CHIP_EN[Bit6] in the CONFIG(00H) register to 1. The chip enters the NORMAL mode.
4) Set the value of CP_MODE [Bit4-Bit3] in the MISC(36H) register to 1, and use the 1X mode.
5) Set the value of CLK_SEL[Bit1-Bit0] in the MISC(36H) register to 0, and use the external clock. If no external clock is available, the internal clock can be selected.
6) Set the values of the D1-D9 CURRENT(26H-2EH) registers to FFH, and the current is 25.5mA.
7) Configure the ENG1_PROG_START_ADDR(4CH), ENG2_PROG_START_ADDR(4DH), and ENG3_PROG_START_ADDR(4EH) registers.
8) Set the value of the ENG_CTRL(01H) register to 0x15, and the control mode of engines 123 is the loading mode.
9) Configure the light effect running program to SRAM.
10) Set the value of the CONFIG(00H) register to 0x6A, and the program execution mode of engines 123 is the free-running mode.
11) Set the value of the ENG_CTRL(01H) register to 0x2A, and the control mode of engines 123 is set to the running mode.

MASTER FADER CONTROL

AW21209 supports the master fader control function. Users only need to write to one register to perform fade-in or fade-out operations on multiple LEDs. The master fader control function has two modes: direct mode and engine mode.

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

DIRECT MODE

The steps for Master Fader1 to use direct mode control D1-D9 are as follows:

1) Power on the chip, pull up the EN pin, and delay for 2ms.
2) Set the value of CHIP_EN[Bit6] in the CONFIG(00H) register to 1, and the chip enters the NORMAL mode.
3) Set the value of CP_MODE [Bit4-Bit3] in the MISC(36H) register to 1, using the 1X mode.
4) Set the value of CLK_SEL[Bit1-Bit0] in the MISC(36H) register to 1, using the internal clock.
5) Set the D1-D9 MAPPING(06H-0EH) to 1, selecting Master Fader1.
6) Set the D1_D9 PWM(16H-1EH) register to FFH, with a duty cycle of 100%.
7) Set the Master Fader1(48H) register to FFH and observe the brightness.
8) Set the Master Fader1(48H) register to 80H and observe the brightness.
9) Set the Master Fader1(48H) register to 0H and observe the brightness.
10) In steps 7, 8, and 9, the brightness of the 9 LEDs gradually decreases to 0.

ENGINE MODE

The steps for Master Fader1 to use engine mode control D1-D9 are as follows:

1) Power on the chip, pull up the EN pin, and delay for 2ms.
2) Set the value of the CHIP_EN[Bit6] in the CONFIG(00H) register to 1,, and the chip enters the NORMAL mode.
3) Set the value of CP_MODE [Bit4-Bit3] in the MISC(36H) register to 1, using the 1X mode.
4) Set the value of CLK_SEL[Bit1-Bit0] in the MISC(36H) register to 1, using the internal clock.
5) Set the D1-D9 MAPPING(06H-0EH) to 0.
6) Set the D1-D9 RATIO_EN(02H-03H) to 1.
7) Set the D1-D9 PWM(16H-1EH) registers to FFH, with a duty cycle of 100%.
8) ENG1 selects D1-D9, set_pwm 0x10, this value will be passed to MASTER_FADER1.

The duty cycle output from ports D1 to D9 will be the product of the values in the D1-D9 PWM registers and the master fader register.The advantage of using the ENG mode lies in its ability to operate independently without the need for an external processor.

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Register Configuration

Register List

ADDR	NAME	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Default
0x00	CONFIG		CHIP_EN	ENG1_EXEC		ENG2_EXEC		ENG3_EXEC		0x00
0x01	ENG_CTRL			ENG1_MODE		ENG2_MODE		ENG3_MODE		0x00
0x02	RATIO1								D9_RATIO_EN	0x00
0x03	RATIO2	D8_RATIO_EN	D7_RATIO_EN	D6_RATIO_EN	D5_RATIO_EN	D4_RATIO_EN	D3_RATIO_EN	D2_RATIO_EN	D1_RATIO_EN	0x00
0x04	LED_ON1								D9_ON	0x01
0x05	LED_ON2	D8_ON	D7_ON	D6_ON	D5_ON	D4_ON	D3_ON	D2_ON	D1_ON	0xFF
0x06	LED1_CTRL	D1_MAPPING		D1_LOG_EN	D1_TEMP_COMP					0x00
0x07	LED2_CTRL	D2_MAPPING		D2_LOG_EN	D2_TEMP_COMP					0x00
0x08	LED3_CTRL	D3_MAPPING		D3_LOG_EN	D3_TEMP_COMP					0x00
0x09	LED4_CTRL	D4_MAPPING		D4_LOG_EN	D4_TEMP_COMP					0x00
0x0A	LED5_CTRL	D5_MAPPING		D5_LOG_EN	D5_TEMP_COMP					0x00
0x0B	LED6_CTRL	D6_MAPPING		D6_LOG_EN	D6_TEMP_COMP					0x00
0x0C	LED7_CTRL	D7_MAPPING		D7_LOG_EN	D7_TEMP_COMP					0x00
0x0D	LED8_CTRL	D8_MAPPING		D8_LOG_EN	D8_TEMP_COMP					0x00
0x0E	LED9_CTRL	D9_MAPPING		D9_LOG_EN	D9_TEMP_COMP					0x00
0x16	LED1_PWM	D1_PWM								0x00
0x17	LED2_PWM	D2_PWM								0x00
0x18	LED3_PWM	D3_PWM								0x00
0x19	LED4_PWM	D4_PWM								0x00
0x1A	LED5_PWM	D5_PWM								0x00

Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

ADDR	NAME	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Default
0x1B	LED6_PWM	D6_PWM								0x00
0x1C	LED7_PWM	D7_PWM								0x00
0x1D	LED8_PWM	D8_PWM								0x00
0x1E	LED9_PWM	D9_PWM								0x00
0x26	LED1_CURR	D1_CURRENT								0xAF
0x27	LED2_CURR	D2_CURRENT								0xAF
0x28	LED3_CURR	D3_CURRENT								0xAF
0x29	LED4_CURR	D4_CURRENT								0xAF
0x2A	LED5_CURR	D5_CURRENT								0xAF
0x2B	LED6_CURR	D6_CURRENT								0xAF
0x2C	LED7_CURR	D7_CURRENT								0xAF
0x2D	LED8_CURR	D8_CURRENT								0xAF
0x2E	LED9_CURR	D9_CURRENT								0xAF
0x36	MISC	VARIABLE_D_SEL	EN_AUTO_INCR	POWER SAVE_EN	CP_MODE		PWM_PS_EN	CLK_SEL		0x40
0x37	ENG1_PC_CFG		ENG1_PC							0x00
0x38	ENG2_PC_CFG		ENG2_PC							0x00
0x39	ENG3_PC_CFG		ENG3_PC							0x00
0x3A	STATUS	LEDTEST_MEAS_DONE	MASK_BUSY	START UP_BUSY	ENGINE_BUSY	EXT_CLK_DET	ENG1_INT	ENG2_INT	ENG3_INT	0x40
0x3B	IO_CTRL	EN_CLK_OUT					INT_CONF	GPO (only AW21209FOR)	INT_GPO	0x00
0x3C	VARIABLE_CFG	VARIABLE								0x00

Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

ADDR	NAME	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Default
0x3D	SOFT_RESET	RESET								0x00
0x3E	TEMP_CTRL	TEMP_MEAS_BUSY					EN_TEMP_SENSOR		SEL_EXT_TEMP	0x00
0x3F	TEMP_READ	TEMPERATURE_INTERNAL								0x19
0x40	TEMP_WRITE	TEMPERATURE_EXTERNAL								0x00
0x41	TEST_CTRL	EN_LED_TEST_ADC	EN_LED_TEST_INT		LED_TEST_SEL					0x00
0x42	TEST_ADC_DATA	TEST_DATA								0x00
0x45	ENG1_VARIABLE_A_CFG	ENG1_VARIABLE_A								0x00
0x46	ENG2_VARIABLE_A_CFG	ENG2_VARIABLE_A								0x00
0x47	ENG3_VARIABLE_A_CFG	ENG3_VARIABLE_A								0x00
0x48	MASTER_FADER1_CFG	MASTER_FADER1								0x00
0x49	MASTER_FADER2_CFG	MASTER_FADER2								0x00
0x4A	MASTER_FADER3_CFG	MASTER_FADER3								0x00
0x4C	ENG1_PROG_START_ADDR		ENG1_START_ADDR							0x00
0x4D	ENG2_PROG_START_ADDR		ENG2_START_ADDR							0x08
0x4E	ENG3_PROG_START_ADDR		ENG3_START_ADDR							0x10

Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

ADDR	NAME	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Default
0x4F	PROG_MEM_PAGE_SEL						SRAM_PAGE_SEL			0x00
0x50	PROG_MEMORY00	INSTRUCTION1_MSB								0x00
0x51	PROG_MEMORY01	INSTRUCTION1_LSB								0x00
0x52	PROG_MEMORY02	INSTRUCTION2_MSB								0x00
0x53	PROG_MEMORY03	INSTRUCTION2_LSB								0x00
0x54	PROG_MEMORY04	INSTRUCTION3_MSB								0x00
0x55	PROG_MEMORY05	INSTRUCTION3_LSB								0x00
0x56	PROG_MEMORY06	INSTRUCTION4_MSB								0x00
0x57	PROG_MEMORY07	INSTRUCTION4_LSB								0x00
0x58	PROG_MEMORY08	INSTRUCTION5_MSB								0x00
0x59	PROG_MEMORY09	INSTRUCTION5_LSB								0x00
0x5A	PROG_MEMORY0A	INSTRUCTION6_MSB								0x00
0x5B	PROG_MEMORY0B	INSTRUCTION6_LSB								0x00
0x5C	PROG_MEMORY0C	INSTRUCTION7_MSB								0x00
0x5D	PROG_MEMORY0D	INSTRUCTION7_LSB								0x00
0x5E	PROG_MEMORY0E	INSTRUCTION8_MSB								0x00
0x5F	PROG_MEMORY0F	INSTRUCTION8_LSB								0x00

Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

ADDR	NAME	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Default
0x60	PROG_MEMORY10	INSTRUCTION9_MSB								0x00
0x61	PROG_MEMORY11	INSTRUCTION9_LSB								0x00
0x62	PROG_MEMORY12	INSTRUCTION10_MSB								0x00
0x63	PROG_MEMORY13	INSTRUCTION10_LSB								0x00
0x64	PROG_MEMORY14	INSTRUCTION11_MSB								0x00
0x65	PROG_MEMORY15	INSTRUCTION11_LSB								0x00
0x66	PROG_MEMORY16	INSTRUCTION12_MSB								0x00
0x67	PROG_MEMORY17	INSTRUCTION12_LSB								0x00
0x68	PROG_MEMORY18	INSTRUCTION13_MSB								0x00
0x69	PROG_MEMORY19	INSTRUCTION13_LSB								0x00
0x6A	PROG_MEMORY1A	INSTRUCTION14_MSB								0x00
0x6B	PROG_MEMORY1B	INSTRUCTION14_LSB								0x00
0x6C	PROG_MEMORY1C	INSTRUCTION15_MSB								0x00
0x6D	PROG_MEMORY1D	INSTRUCTION15_LSB								0x00
0x6E	PROG_MEMORY1E	INSTRUCTION16_MSB								0x00
0x6F	PROG_MEMORY1F	INSTRUCTION16_LSB								0x00
0x70	ENG1_MAPPING1	GPO_MAPPING_ENG1(only AW21209FOR)							D9_MAPPING_ENG1	0x00

Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

ADDR	NAME	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Default
0x71	ENG1_MAPPING2	D8_MAPPING_ENG1	D7_MAPPING_ENG1	D6_MAPPING_ENG1	D5_MAPPING_ENG1	D4_MAPPING_ENG1	D3_MAPPING_ENG1	D2_MAPPING_ENG1	D1_MAPPING_ENG1	0x00
0x72	ENG2_MAPPING1	GPO_MAPPING_ENG2(only AW21209FOR)							D9_MAPPING_ENG2	0x00
0x73	ENG2_MAPPING2	D8_MAPPING_ENG2	D7_MAPPING_ENG2	D6_MAPPING_ENG2	D5_MAPPING_ENG2	D4_MAPPING_ENG2	D3_MAPPING_ENG2	D2_MAPPING_ENG2	D1_MAPPING_ENG2	0x00
0x74	ENG3_MAPPING1	GPO_MAPPING_ENG3(only AW21209FOR)							D9_MAPPING_ENG3	0x00
0x75	ENG3_MAPPING2	D8_MAPPING_ENG3	D7_MAPPING_ENG3	D6_MAPPING_ENG3	D5_MAPPING_ENG3	D4_MAPPING_ENG3	D3_MAPPING_ENG3	D2_MAPPING_ENG3	D1_MAPPING_ENG3	0x00
0x76	CP_CTRL	THRESHOLD		THSHOLD_LIMIT_RAISE_EN	TIMER		FORCE_1X			0x00
0x7F	ENHANCED_CFG	ENHANCED_PASSWORD								0x00
0x80	PWM_CFG					PWM_DIV			PWM_MODE	0x16
0xA0	CHIP_ID_R					CHIP_ID				0x00

Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Register Detailed Description

CONFIG: (Address 00h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7	Reserved	RW	Not used	0
6	CHIP_EN	RW	Chip work mode. In standby mode, All registers are NOT allowed to be accessed except CHIP_EN bit 0: Standby mode. 1: Normal mode	0x0
5:4	ENG1_EXEC	RW	Engine 1 program execution control. Execution register bits define how the program is executed. Hold mode: Hold causes the execution engine to finish the current instruction and then stop. Program counter (PC) can be read or written only in this mode. Step mode: Execute the instruction at the location pointed by the PC, increment the PC by one and then reset ENG1_EXEC bits to 00 (i.e. enter hold). Rree run mode: Start program execution from the location pointed by the PC. execute once mode: Execute the instruction pointed by the current PC value and reset ENG1_EXEC to 00 (that is, enter hold). The difference between step and execute once is that execute once does not increment the PC. b00: hold mode b01: step mode b10: free run mode b11: execute once mode	0x0
3:2	ENG2_EXEC	RW	Please refer to the description of ENG1_EXEC.	0x0
1:0	ENG3_EXEC	RW	Please refer to the description of ENG1_EXEC.	0x0

ENG_CTRL: (Address 01h)

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Bit	Symbol	R/W	Description	Default
7:6	Reserved	RW	Not used	0
5:4	ENG1_ MODE	RW	Engine 1 operation modes. Disable mode: Engines can be configured to disabled mode each one separately. Load program mode: Writing to program memory is allowed only when the engine is in load program operation mode and engine busy bit (reg 3A) is not set. Serial bus master should check the busy bit before writing to program memory or allow at least 1ms delay after entering to load mode before memory write, to ensure initialization. All the three engines are in hold while one or more engines are in load program mode. PWM values are frozen, also. Program execution continues when all the engines are out of load program mode. Load program mode resets the program counter of the respective engine. Load program mode can be entered from the disabled mode only. Entering load program mode from the run program mode or halt mode is not allowed. Run Program mode: Run program mode executes the instructions stored in the program memory. Execution register (ENG1_EXEC etc.) bits define how the program is executed (hold, step, free run or execute once). Program start address can be programmed to the PC register. The PC is reset to zero when the PC's upper limit value is reached. Halt mode: Instruction execution aborts immediately, and engine operation halts. b00: Disable mode b01: Load program mode b10: Run Program mode b11: Halt mode	0x0
3:2	ENG2_ MODE	RW	Please refer to the description of ENG1_MODE.	0x0
1:0	ENG3_ MODE	RW	Please refer to the description of ENG1_MODE.	0x0

RATIO1: (Address 02h)

Bit	Symbol	R/W	Description	Default
7:1	Reserved	RW	Not used	0
0	D9_RATIO_ EN	RW	Ratiometric dimming enable. 0: Disable 1: Enable	0x0

RATIO2: (Address 03h)

Bit	Symbol	R/W	Description	Default

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

7	D8_RATIO_EN	RW	Ratiometric dimming enable. 0: Disable 1: Enable	0x0
6	D7_RATIO_EN	RW	Ratiometric dimming enable. 0: Disable 1: Enable	0x0
5	D6_RATIO_EN	RW	Ratiometric dimming enable. 0: Disable 1: Enable	0x0
4	D5_RATIO_EN	RW	Ratiometric dimming enable. 0: Disable 1: Enable	0x0
3	D4_RATIO_EN	RW	Ratiometric dimming enable. 0: Disable 1: Enable	0x0
2	D3_RATIO_EN	RW	Ratiometric dimming enable. 0: Disable 1: Enable	0x0
1	D2_RATIO_EN	RW	Ratiometric dimming enable. 0: Disable 1: Enable	0x0
0	D1_RATIO_EN	RW	Ratiometric dimming enable. 0: Disable 1: Enable	0x0

LED_ON1: (Address 04h)

Bit	Symbol	R/W	Description	Default
7:1	Reserved	RW	Not used	0
0	D9_ON	RW	LED Output analog circuit enable. 0: Disable 1: Enable	0x1

LED_ON2: (Address 05h)

Bit	Symbol	R/W	Description	Default
7	D8_ON	RW	LED Output analog circuit enable. 0: Disable 1: Enable	0x1
6	D7_ON	RW	LED Output analog circuit enable. 0: Disable 1: Enable	0x1
5	D6_ON	RW	LED Output analog circuit enable. 0: Disable 1: Enable	0x1

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

4	D5_ON	RW	LED Output analog circuit enable. 0: Disable 1: Enable	0x1
3	D4_ON	RW	LED Output analog circuit enable. 0: Disable 1: Enable	0x1
2	D3_ON	RW	LED Output analog circuit enable. 0: Disable 1: Enable	0x1
1	D2_ON	RW	LED Output analog circuit enable. 0: Disable 1: Enable	0x1
0	D1_ON	RW	LED Output analog circuit enable. 0: Disable 1: Enable	0x1

LED1_CTRL: (Address 06h)

Bit	Symbol	R/W	Description	Default
7:6	D1_ MAPPING	RW	The duty cycle on D1 output is the D1 PWM register value (address 16H) multiplied with the value in the MASTER FADER register. b00: no mapping b01: MASTER FADER1 b10: MASTER FADER2 b11: MASTER FADER3	0x0
5	D1_LOG_EN	RW	0: linear adjustment 1: logarithmic adjustment	0x0

Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

4:0	D1_TEMP_COMP	RW	The reference temperature is 25°C (that is, the temperature at which the compensation has no effect) and the correction factor (slope) can be set in 0.1% /°C steps to any value between -1.5% /°C and +1.5%/°C, with a default to 0.0% /°C. The PWM duty cycle at temperature T (in centigrade) can be obtained as follows: $PWM_F = [PWM_S – (25 – T) \times$ correction factor $\times PWM_S] / 2$, where PWM_F is the final duty cycle at temperature T, PWM_S is the set PWM duty cycle (PWM duty cycle is set in registers 16H to 1EH) and the value of the correction factor is obtained from the table below. b00000: Not activated b11111: -1.5%/°C b11110: -1.4%/°C b11101: -1.3%/°C b11100: -1.2%/°C b11011: -1.1%/°C b11010: -1%/°C b11001: -0.9%/°C b11000: -0.8%/°C b10111: -0.7%/°C b10110: -0.6%/°C b10101: -0.5%/°C b10100: -0.4%/°C b10011: -0.3%/°C b10010: -0.2%/°C b10001: -0.1%/°C b10000: 0%/°C b00001: 0.1%/°C b00010: 0.2%/°C b00011: 0.3%/°C b00100: 0.4%/°C b00101: 0.5%/°C b00110: 0.6%/°C b00111: 0.7%/°C b01000: 0.8%/°C b01001: 0.9%/°C b01010: 1%/°C b01011: 1.1%/°C b01100: 1.2%/°C b01101: 1.3%/°C b01110: 1.4%/°C b01111: 1.5%/°C	0x0

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

LED2_CTRL: (Address 07h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:6	D2_ MAPPING	RW	The duty cycle on D2 output is the D2 PWM register value (address 17H) multiplied with the value in the MASTER FADER register. b00: no mapping b01: MASTER FADER1 b10: MASTER FADER2 b11: MASTER FADER3	0x0
5	D2_LOG_EN	RW	0: linear adjustment 1: logarithmic adjustment	0x0
4:0	D2_TEMP_ COMP	RW	Please refer to the description of D1_TEMP_COMP	0x0

LED3_CTRL: (Address 08h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:6	D3_ MAPPING	RW	The duty cycle on D3 output is the D3 PWM register value (address 18H) multiplied with the value in the MASTER FADER register. b00: no mapping b01: MASTER FADER1 b10: MASTER FADER2 b11: MASTER FADER3	0x0
5	D3_LOG_EN	RW	0: linear adjustment 1: logarithmic adjustment	0x0
4:0	D3_TEMP_ COMP	RW	Please refer to the description of D1_TEMP_COMP	0x0

LED4_CTRL: (Address 09h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:6	D4_ MAPPING	RW	The duty cycle on D4 output is the D4 PWM register value (address 19H) multiplied with the value in the MASTER FADER register. b00: no mapping b01: MASTER FADER1 b10: MASTER FADER2 b11: MASTER FADER3	0x0
5	D4_LOG_EN	RW	0: linear adjustment 1: logarithmic adjustment	0x0
4:0	D4_TEMP_ COMP	RW	Please refer to the description of D1_TEMP_COMP	0x0

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

LED5_CTRL: (Address 0Ah)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:6	D5_ MAPPING	RW	The duty cycle on D5 output is the D5 PWM register value (address 1AH) multiplied with the value in the MASTER FADER register. b00: no mapping b01: MASTER FADER1 b10: MASTER FADER2 b11: MASTER FADER3	0x0
5	D5_LOG_EN	RW	0: linear adjustment 1: logarithmic adjustment	0x0
4:0	D5_TEMP_ COMP	RW	Please refer to the description of D1_TEMP_COMP	0x0

LED6_CTRL: (Address 0Bh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:6	D6_ MAPPING	RW	The duty cycle on D6 output is the D6 PWM register value (address 1BH) multiplied with the value in the MASTER FADER register. b00: no mapping b01: MASTER FADER1 b10: MASTER FADER2 b11: MASTER FADER3	0x0
5	D6_LOG_EN	RW	0: linear adjustment 1: logarithmic adjustment	0x0
4:0	D6_TEMP_ COMP	RW	Please refer to the description of D1_TEMP_COMP	0x0

LED7_CTRL: (Address 0Ch)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:6	D7_ MAPPING	RW	The duty cycle on D7 output is the D7 PWM register value (address 1CH) multiplied with the value in the MASTER FADER register. b00: no mapping b01: MASTER FADER1 b10: MASTER FADER2 b11: MASTER FADER3	0x0
5	D7_LOG_EN	RW	0: linear adjustment 1: logarithmic adjustment	0x0
4:0	D7_TEMP_ COMP	RW	Please refer to the description of D1_TEMP_COMP	0x0

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

LED8_CTRL: (Address 0Dh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:6	D8_ MAPPING	RW	The duty cycle on D8 output is the D8 PWM register value (address 1DH) multiplied with the value in the MASTER FADER register. b00: no mapping b01: MASTER FADER1 b10: MASTER FADER2 b11: MASTER FADER3	0x0
5	D8_LOG_EN	RW	0: linear adjustment 1: logarithmic adjustment	0x0
4:0	D8_TEMP_ COMP	RW	Please refer to the description of D1_TEMP_COMP	0x0

LED9_CTRL: (Address 0Eh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:6	D9_ MAPPING	RW	The duty cycle on D9 output is the D9 PWM register value (address 1EH) multiplied with the value in the MASTER FADER register. b00: no mapping b01: MASTER FADER1 b10: MASTER FADER2 b11: MASTER FADER3	0x0
5	D9_LOG_EN	RW	0: linear adjustment 1: logarithmic adjustment	0x0
4:0	D9_TEMP_ COMP	RW	Please refer to the description of D1_TEMP_COMP	0x0

LED1_PWM: (Address 16h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D1_PWM	RW	LED pwm duty cycle. When PWM is not FF, duty cycle is (PWM/256)*100%; When PWM is FF, duty cycle is 100%.	0x0

LED2_PWM: (Address 17h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D2_PWM	RW	LED pwm duty cycle. When PWM is not FF, duty cycle is (PWM/256)*100%; When PWM is FF, duty cycle is 100%.	0x0

LED3_PWM: (Address 18h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D3_PWM	RW	LED pwm duty cycle. When PWM is not FF, duty cycle is (PWM/256)*100%; When PWM is FF, duty cycle is 100%.	0x0

Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

LED4_PWM: (Address 19h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D4_PWM	RW	LED pwm duty cycle. When PWM is not FF, duty cycle is (PWM/256)*100%; When PWM is FF, duty cycle is 100%.	0x0

LED5_PWM: (Address 1Ah)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D5_PWM	RW	LED pwm duty cycle. When PWM is not FF, duty cycle is (PWM/256)*100%; When PWM is FF, duty cycle is 100%.	0x0

LED6_PWM: (Address 1Bh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D6_PWM	RW	LED pwm duty cycle. When PWM is not FF, duty cycle is (PWM/256)*100%; When PWM is FF, duty cycle is 100%.	0x0

LED7_PWM: (Address 1Ch)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D7_PWM	RW	LED pwm duty cycle. When PWM is not FF, duty cycle is (PWM/256)*100%; When PWM is FF, duty cycle is 100%.	0x0

LED8_PWM: (Address 1Dh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D8_PWM	RW	LED pwm duty cycle. When PWM is not FF, duty cycle is (PWM/256)*100%; When PWM is FF, duty cycle is 100%.	0x0

LED9_PWM: (Address 1Eh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D9_PWM	RW	LED pwm duty cycle. When PWM is not FF, duty cycle is (PWM/256)*100%; When PWM is FF, duty cycle is 100%.	0x0

LED1_CURR: (Address 26h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D1_ CURRENT	RW	LED driver output current control register. The resolution is 8-bits and step size is 100 μA. 0x0 : 0.0mA 0x1 : 0.1mA … 0xFF: 25.5mA	0xAF

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

LED2_CURR: (Address 27h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D2_ CURRENT	RW	LED driver output current control register. The resolution is 8-bits and step size is 100 μA. 0x0 : 0.0mA 0x1 : 0.1mA … 0xFF: 25.5mA	0xAF

LED3_CURR: (Address 28h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D3_ CURRENT	RW	LED driver output current control register. The resolution is 8-bits and step size is 100 μA. 0x0 : 0.0mA 0x1 : 0.1mA … 0xFF: 25.5mA	0xAF

LED4_CURR: (Address 29h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D4_ CURRENT	RW	LED driver output current control register. The resolution is 8-bits and step size is 100 μA. 0x0 : 0.0mA 0x1 : 0.1mA … 0xFF: 25.5mA	0xAF

LED5_CURR: (Address 2Ah)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D5_ CURRENT	RW	LED driver output current control register. The resolution is 8-bits and step size is 100 μA. 0x0 : 0.0mA 0x1 : 0.1mA … 0xFF: 25.5mA	0xAF

LED6_CURR: (Address 2Bh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D6_ CURRENT	RW	LED driver output current control register. The resolution is 8-bits and step size is 100 μA. 0x0 : 0.0mA 0x1 : 0.1mA … 0xFF: 25.5mA	0xAF

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

LED7_CURR: (Address 2Ch)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D7_ CURRENT	RW	LED driver output current control register. The resolution is 8-bits and step size is 100 μA. 0x0 : 0.0mA 0x1 : 0.1mA … 0xFF: 25.5mA	0xAF

LED8_CURR: (Address 2Dh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D8_ CURRENT	RW	LED driver output current control register. The resolution is 8-bits and step size is 100 μA. 0x0 : 0.0mA 0x1 : 0.1mA … 0xFF: 25.5mA	0xAF

LED9_CURR: (Address 2Eh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	D9_ CURRENT	RW	LED driver output current control register. The resolution is 8-bits and step size is 100 μA. 0x0 : 0.0mA 0x1 : 0.1mA … 0xFF: 25.5mA	0xAF

MISC: (Address 36h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7	VARIABLE_ D_SEL	RW	Engine variable D source selection. 0: From 3CH register 1: From 42H register	0x0
6	EN_AUTO_ INCR	RW	The automatic increment feature of the serial bus address enables a quick memory write of successive registers within one transmission. 0: Disable 1: Enable	0x1
5	POWERSAV E_EN	RW	Automatic Power-Save mode enable. 0: Disabled 1: Enabled	0x0

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

4:3	CP_MODE	RW	Charge-pump operation mode. b00: off b01: forced 1x b10: forced 1.5× b11: automatic selection	0x0
2	PWM_PS_EN	RW	PWM power-save mode enable.	0x0
1:0	CLK_SEL	RW	Engine Clock Selection. b00: forced external clock b01: forced internal clock b10: automatic selection b11: forced internal clock	0x0

ENG1_PC_CFG: (Address 37h)

Bit	Symbol	R/W	Description	Default
7	Reserved	RW	Not used	0
6:0	ENG1_PC	RW	Program counter value for program execution engine 1.	0x0

ENG2_PC_CFG: (Address 38h)

Bit	Symbol	R/W	Description	Default
7	Reserved	RW	Not used	0
6:0	ENG2_PC	RW	Program counter value for program execution engine 2.	0x0

ENG3_PC_CFG: (Address 39h)

Bit	Symbol	R/W	Description	Default
7	Reserved	RW	Not used	0
6:0	ENG3_PC	RW	Program counter value for program execution engine 3.	0x0

STATUS: (Address 3Ah)

Bit	Symbol	R/W	Description	Default
7	LEDTEST_MEAS_DONE	RO	This bit indicates when the LED test is done, and the result is written to the TEST DATA register. Typically the conversion takes 2.7 milliseconds to complete.	0x0
6	MASK_BUSY	RW	Mask bit for interrupts generated by STARTUP_BUSY or ENGINE_BUSY. Reading the register 3A clears the status bits [5:4] and releases INT pin to high state. When this bit is 1, no external interrupt is generated from STARTUP_BUSY or ENGINE_BUSY event. 0: not masked 1: masked	0x1

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

5	STARTUP_BUSY	RO	A status bit which indicates that the device is running the internal start-up sequence. 0: Done 1: Busy Note: STARTUP_BUSY = 1 always when CHIP_EN bit is 0.	0x0
4	ENGINE_BUSY	RO	A status bit which indicates that a program execution engine is clearing internal registers. Serial bus master should not write or read program memory, or registers 00H, 37H to 39H or 4CH to 4EH, when this bit is set to 1. 0: Ready 1: Busy	0x0
3	EXT_CLK_DET	RO	0: external clock not detected 1: external clock detected	0x0
2	ENG1_INT	RO	Interrupt bits for program execution engine 1. This bit is set by END or INT instruction. Reading the interrupt bit clears the interrupt.	0x0
1	ENG2_INT	RO	Interrupt bits for program execution engine 2. This bit is set by END or INT instruction. Reading the interrupt bit clears the interrupt.	0x0
0	ENG3_INT	RO	Interrupt bits for program execution engine 3. This bit is set by END or INT instruction. Reading the interrupt bit clears the interrupt.	0x0

IO_CTRL: (Address 3Bh)

Bit	Symbol	R/W	Description	Default
7	EN_CLK_OUT	RW	PAD_CLK32K Output Enable. 0: Disable 1: Enable	0x0
6:3	Reserved	RW	Not used	0
2	INT_CONF	RW	PAD_INT used as General-Output-Function enable. If enabled, output data is controlled by INT_GPO bit. 0: Disable 1: Enable	0x0
1	GPO	RW	GPO output data.	0x0
0	INT_GPO	RW	PAD_INT output data.	0x0

VARIABLE_CFG: (Address 3Ch)

Bit	Symbol	R/W	Description	Default
7:0	VARIABLE	RW	These bits are used for storing a global 8-bit variable.	0x0

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

SOFT_RESET: (Address 3Dh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	RESET	WO	Writing 8'hFF into this register resets chip. Internal registers are reset to the default values. Reading RESET register returns 8'h00.	0x0

TEMP_CTRL: (Address 3Eh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7	TEMP_ MEAS_ BUSY	RO	Temperature measurement done or not. 0: Done 1: Ongoing	0x0
6:3	Reserved	RW	Not used	0
2	EN_TEMP_ SENSOR	RW	Temp sensor Enable. Every time when EN_TEMP_SENSOR is written high a new measurement period is started. The length of the measurement period depends on temperature. At 25°C a measurement takes 20 milliseconds. Temperature can be read from register 3F. 0: Disable 1: Enable	0x0
1	Reserved	RW	Not used	0
0	SEL_EXT_ TEMP	RW	Temperature compensation source select. 0: From 3FH register 1: From 40H register	0x0

TEMP_READ: (Address 3Fh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	TEMPERAT URE_ INTERNAL	RO	These bits are used for storing an 8-bit temperature reading acquired from the internal temperature sensor. This register is a read-only register. Temperature reading is stored in 8-bit 2's complement format. b11010111: -41°C b11011000: -40°C b11011001: -39°C ..... b01011001: 89°C	0x19

TEMP_WRITE: (Address 40h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	TEMPERAT URE_ EXTERNAL	RW	These bits are used for storing an 8-bit temperature reading acquired from an external sensor, if such a sensor is used. Temperature reading is stored in 8-bit 2's complement format, like in 3F register.	0x0

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

TEST_CTRL: (Address 41h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7	EN_LED_ TEST_ADC	RW	Writing this bit high starts single LED test conversion. LED test measurement cycle is 2.7 milliseconds.	0x0
6	EN_LED_ TEST_INT	RW	LED_TEST interrupt function enable. If this bit is 1, interrupt signal is sent to the INT pin when the LED test is accomplished. 0: Disable 1: Enable	0x0
5	Reserved	RW	Not used	0
4:0	LED_TEST_ SEL	RW	b00000: D1 b00001: D2 b00010: D3 b00011: D4 b00100: D5 b00101: D6 b00110: D7 b00111: D8 b01000: D9 b01111: VOUT b10000: VDD Others: Reserved	0x0

TEST_ADC_DATA: (Address 42h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	TEST_DATA	RO	LED test ADC's least significant bit corresponds to 28.6 mV. The measured voltage V (typical) is calculated as follows: V = (RESULT(DEC) × 28.6) mV.	0x0

ENG1_VARIABLE_A_CFG: (Address 45h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	ENG1_ VARIABLE_ A	RO	These bits are used for Engine 1 local variable.	0x0

ENG2_VARIABLE_A_CFG: (Address 46h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	ENG2_ VARIABLE_ A	RO	These bits are used for Engine 2 local variable.	0x0

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

ENG3_VARIABLE_A_CFG: (Address 47h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	ENG3_ VARIABLE_ A	RO	These bits are used for Engine 3 local variable.	0x0

MASTER_FADER1_CFG: (Address 48h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	MASTER_ FADER1	RW	An 8-bit register to control all the LED-drivers mapped to MASTER FADER1. Master fader allows the user to control dimming of multiple LEDS with a single serial bus write.	0x0

MASTER_FADER2_CFG: (Address 49h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	MASTER_ FADER2	RW	An 8-bit register to control all the LED-drivers mapped to MASTER FADER2. Master fader allows the user to control dimming of multiple LEDS with a single serial bus write.	0x0

MASTER_FADER3_CFG: (Address 4Ah)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	MASTER_ FADER3	RW	An 8-bit register to control all the LED-drivers mapped to MASTER FADER3. Master fader allows the user to control dimming of multiple LEDS with a single serial bus write.	0x0

ENG1_PROG_START_ADDR: (Address 4Ch)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7	Reserved	RW	Not used	0
6:0	ENG1_ START_ ADDR	RW	Engine 1 program start address.	0x0

ENG2_PROG_START_ADDR: (Address 4Dh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7	Reserved	RW	Not used	0
6:0	ENG2_ START_ ADDR	RW	Engine 2 program start address.	0x8

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

ENG3_PROG_START_ADDR: (Address 4Eh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7	Reserved	RW	Not used	0
6:0	ENG3_ START_ ADDR	RW	Engine 3 program start address.	0x10

PROG_MEM_PAGE_SEL: (Address 4Fh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:3	Reserved	RW	Not used	0
2:0	SRAM_ PAGE_ SEL	RW	These bits select the program memory page. The program memory is divided into six pages of 16 instructions; 0: page 0 1: page 1 2: page 2 … 5: page 5	0x0

PROG_MEMORY00: (Address 50h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	INSTRUCTI ON1_MSB	RW	Instructions[15:8].	0x0

.....

PROG_MEMORY1F: (Address 6Fh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	INSTRUCTI ON16_LSB	RW	Instructions[7:0].	0x0

ENG1_MAPPING1: (Address 70h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7	GPO_ MAPPING_ ENG1	RO	GPO pin mapped to ENG1 or not. 0: non-mapped 1: mapped	0x0
6:4	Reserved	RO	Not used	0
3	MF3_ MAPPING_ ENG1	RO	MasterFader3 pin mapped to ENG1 or not. 0: non-mapped 1: mapped	0x0
2	MF2_ MAPPING_ ENG1	RO	MasterFader2 pin mapped to ENG1 or not. 0: non-mapped 1: mapped	0x0

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

1	MF1_ MAPPING_ ENG1	RO	MasterFader1 pin mapped to ENG1 or not. 0: non-mapped 1: mapped	0x0
0	D9_ MAPPING_ ENG1	RO	LED9 pin mapped to ENG1 or not. 0: non-mapped 1: mapped	0x0

ENG1_MAPPING2: (Address 71h)

Bit	Symbol	R/W	Description	Default
7	D8_ MAPPING_ ENG1	RO	LED8 pin mapped to ENG1 or not. 0: non-mapped 1: mapped	0x0
6	D7_ MAPPING_ ENG1	RO	LED7 pin mapped to ENG1 or not. 0: non-mapped 1: mapped	0x0
5	D6_ MAPPING_ ENG1	RO	LED6 pin mapped to ENG1 or not. 0: non-mapped 1: mapped	0x0
4	D5_ MAPPING_ ENG1	RO	LED5 pin mapped to ENG1 or not. 0: non-mapped 1: mapped	0x0
3	D4_ MAPPING_ ENG1	RO	LED4 pin mapped to ENG1 or not. 0: non-mapped 1: mapped	0x0
2	D3_ MAPPING_ ENG1	RO	LED3 pin mapped to ENG1 or not. 0: non-mapped 1: mapped	0x0
1	D2_ MAPPING_ ENG1	RO	LED2 pin mapped to ENG1 or not. 0: non-mapped 1: mapped	0x0
0	D1_ MAPPING_ ENG1	RO	LED1 pin mapped to ENG1 or not. 0: non-mapped 1: mapped	0x0

ENG2_MAPPING1: (Address 72h)

Bit	Symbol	R/W	Description	Default
7	GPO_ MAPPING_ ENG2	RO	GPO pin mapped to ENG2 or not. 0: non-mapped 1: mapped	0x0
6:4	Reserved	RO	Not used	0

Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

3	MF3_ MAPPING_ ENG2	RO	MasterFader3 pin mapped to ENG2 or not. 0: non-mapped 1: mapped	0x0
2	MF2_ MAPPING_ ENG2	RO	MasterFader2 pin mapped to ENG2 or not. 0: non-mapped 1: mapped	0x0
1	MF1_ MAPPING_ ENG2	RO	MasterFader1 pin mapped to ENG2 or not. 0: non-mapped 1: mapped	0x0
0	D9_ MAPPING_ ENG2	RO	LED9 pin mapped to ENG2 or not. 0: non-mapped 1: mapped	0x0

ENG2_MAPPING2: (Address 73h)

Bit	Symbol	R/W	Description	Default
7	D8_ MAPPING_ ENG2	RO	LED8 pin mapped to ENG2 or not. 0: non-mapped 1: mapped	0x0
6	D7_ MAPPING_ ENG2	RO	LED7 pin mapped to ENG2 or not. 0: non-mapped 1: mapped	0x0
5	D6_ MAPPING_ ENG2	RO	LED6 pin mapped to ENG2 or not. 0: non-mapped 1: mapped	0x0
4	D5_ MAPPING_ ENG2	RO	LED5 pin mapped to ENG2 or not. 0: non-mapped 1: mapped	0x0
3	D4_ MAPPING_ ENG2	RO	LED4 pin mapped to ENG2 or not. 0: non-mapped 1: mapped	0x0
2	D3_ MAPPING_ ENG2	RO	LED3 pin mapped to ENG2 or not. 0: non-mapped 1: mapped	0x0
1	D2_ MAPPING_ ENG2	RO	LED2 pin mapped to ENG2 or not. 0: non-mapped 1: mapped	0x0
0	D1_ MAPPING_ ENG2	RO	LED1 pin mapped to ENG2 or not. 0: non-mapped 1: mapped	0x0

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

ENG3_MAPPING1: (Address 74h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7	GPO_ MAPPING_ ENG3	RO	GPO pin mapped to ENG3 or not. 0: non-mapped 1: mapped	0x0
6:4	Reserved	RO	Not used	0
3	MF3_ MAPPING_ ENG3	RO	MasterFader3 pin mapped to ENG3 or not. 0: non-mapped 1: mapped	0x0
2	MF2_ MAPPING_ ENG3	RO	MasterFader2 pin mapped to ENG3 or not. 0: non-mapped 1: mapped	0x0
1	MF1_ MAPPING_ ENG3	RO	MasterFader1 pin mapped to ENG3 or not. 0: non-mapped 1: mapped	0x0
0	D9_ MAPPING_ ENG3	RO	LED9 pin mapped to ENG3 or not. 0: non-mapped 1: mapped	0x0

ENG3_MAPPING2: (Address 75h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7	D8_ MAPPING_ ENG3	RO	LED8 pin mapped to ENG3 or not. 0: non-mapped 1: mapped	0x0
6	D7_ MAPPING_ ENG3	RO	LED7 pin mapped to ENG3 or not. 0: non-mapped 1: mapped	0x0
5	D6_ MAPPING_ ENG3	RO	LED6 pin mapped to ENG3 or not. 0: non-mapped 1: mapped	0x0
4	D5_ MAPPING_ ENG3	RO	LED5 pin mapped to ENG3 or not. 0: non-mapped 1: mapped	0x0
3	D4_ MAPPING_ ENG3	RO	LED4 pin mapped to ENG3 or not. 0: non-mapped 1: mapped	0x0
2	D3_ MAPPING_ ENG3	RO	LED3 pin mapped to ENG3 or not. 0: non-mapped 1: mapped	0x0

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

1	D2_ MAPPING_ ENG3	RO	LED2 pin mapped to ENG3 or not. 0: non-mapped 1: mapped	0x0
0	D1_ MAPPING_ ENG3	RO	LED1 pin mapped to ENG3 or not. 0: non-mapped 1: mapped	0x0

CP_CTRL: (Address 76h)

Bit	Symbol	R/W	Description	Default
7:6	THRESHOLD	RW	Threshold voltage (typical) pre-setting. Bits set the threshold voltage at which the charge-pump gain changes from 1.5× to 1×. The threshold voltage is defined as the voltage difference between highest voltage output (D1 to D6) and input voltage VDD: VTHRESHOLD = VDD – MAX(voltage on D1 to D6). If VTHRESHOLD is larger than the set value (100 mV to 400 mV), the charge pump is in 1× mode. b00: 400mV b01: 300mV b10: 200mV b11: 100mV	0x0
5	THSHOLD_ LIMIT_ RAISE_EN	RW	If this bit is 1, real threshold is [7:6] + 400mv; if not, real threshold is same as [7:6] defined.	0x0
4:3	TIMER	RW	A forced mode change from 1.5× to 1× is attempted at the interval specified with these bits. Mode change is allowed if there is enough voltage over the LED drivers to ensure proper operation. Set FORCE_1x to 1 (see following 76H - Bit [2] FORCE_1x) to activate this feature. When this bit is 11, charge pump switches gain from 1× mode to 1.5× mode only; the gain reset back to 1× is enabled under certain conditions, for example in the powersave mode. b00: 5ms b01: 10ms b10: 50ms b11: infinite	0x0
2	FORCE_1X	RW	Activates forced mode change. In forced mode, charge pump mode change from 1.5× to 1× is attempted at the constant interval specified with the TIMER bits.	0x0
1:0	Reserved	RW	Not used	0

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

ENHANCED_CFG: (Address 7Fh)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:0	ENHANCED_PASSWORD	WO	Write 0x23 to enable 0x80 register access.	0x0

PWM_CFG: (Address 80h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:4	Reserved	RW	Not used	0x1
3:1	PWM_DIV	RO	PWM clock division b000: 1 b001: 1/2 b010: 1/4 b011: 1/8 b100: 1/16 b101: 1/32 b110: 1/64 b111: 1/128	0x3
0	PWM_MODE	RW	0: 12bit 1: 9bit + 3Dither	0x0

CHIP_ID_R: (Address A0h)				
Bit	**Symbol**	**R/W**	**Description**	**Default**
7:4	Reserved	RO	Not used	0
3:0	CHIP_ID	RO	CHIP_ID. 0: AW21209FOR 8: AW21209QNR	0x00\ 0x08

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Application Information

The AW21209 series supports up to 9 LED channels or 3 RGB channels in parallel. It is recommended to use a combination of 1μF and 0.1μF X5R or X7R capacitors for the input capacitor. For the charge pump capacitor, C_{out} is recommended to be 1μF, and C_{fly1} and C_{fly2} are recommended to be 0.47μF. Since the charge pump is only connected to D1-6, it is suggested to connect the red LED to D7-9 when using RGB. The SCL, SDA, and INT pins are default open-drain outputs and need to be connected to a pull-up resistor to the communication level, which is determined by the voltage on the EN pin. The I²C address of the AW21209 can be selected through AD0 and AD1. It is recommended to connect the AD0 and AD1 pins to GND, resulting in a 7-bit address of 0x32.

Typical Application

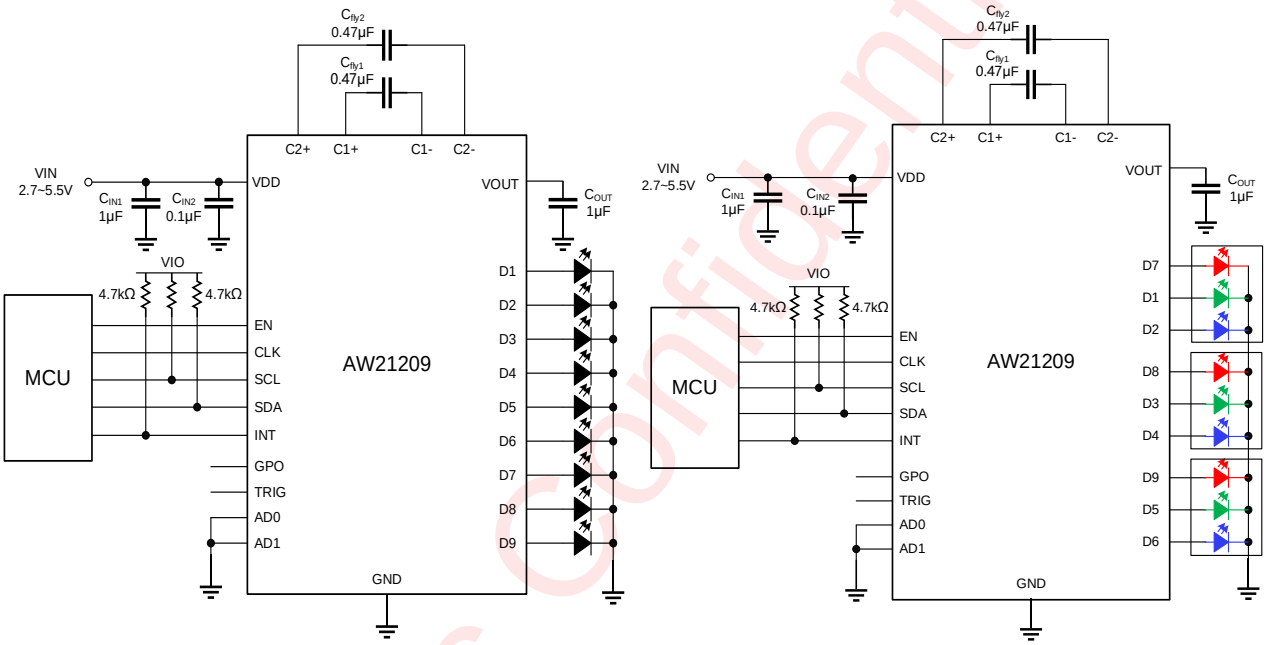


Figure 17 AW21209 Application Circuit

PCB Layout Consideration

1. Place the input capacitors (C_{IN1}, C_{IN2}) close to the corresponding pins of the chip, and position the small-sized high-frequency filter capacitors closer to the pins than the large-sized low-frequency filter capacitors.

2. Place the charge pump capacitors (C_{out}, C_{fly1}, C_{fly2}) as close as possible to the corresponding pins of the chip to ensure that the traces are as short and thick as possible.

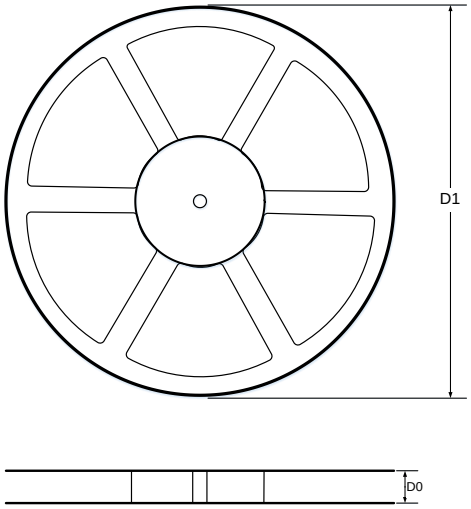
Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

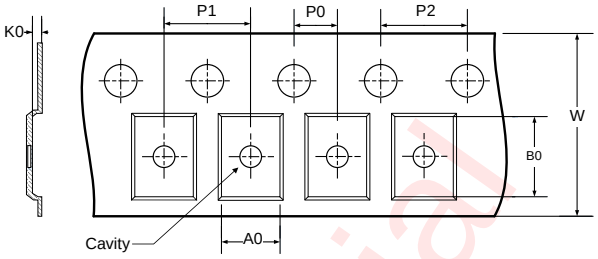
Tape And Reel Information

AW21209FOR:

REEL DIMENSIONS

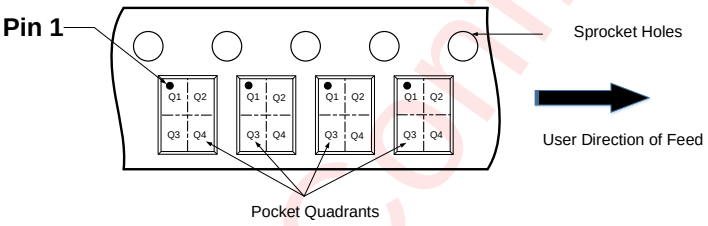


TAPE DIMENSIONS



A0：Dimension designed to accommodate the component width
B0：Dimension designed to accommodate the component length
K0：Dimension designed to accommodate the component thickness
W：Overall width of the carrier tape
P0：Pitch between successive cavity centers and sprocket hole
P1：Pitch between successive cavity centers
P2：Pitch between sprocket hole
D1：Reel Diameter
D0：Reel Width

QUADRANT ASSIGNMENTS FOR PIN 1 ORIENTATION IN TAPE



Note：The above picture is for reference only. Please refer to the value in the table below for the actual size

DIMENSIONS AND PIN1 ORIENTATION

D1 (mm)	D0 (mm)	A0 (mm)	B0 (mm)	K0 (mm)	P0 (mm)	P1 (mm)	P2 (mm)	W (mm)	Pin1 Quadrant
178.0	9.00	2.52	2.52	0.75	2.00	4.00	4.00	8.00	Q1

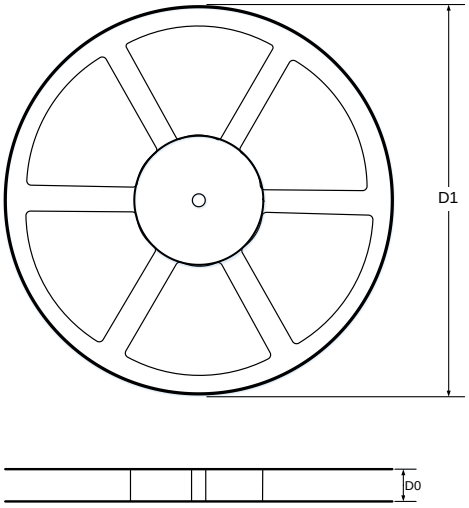
All dimensions are nominal

Awinic Confidential

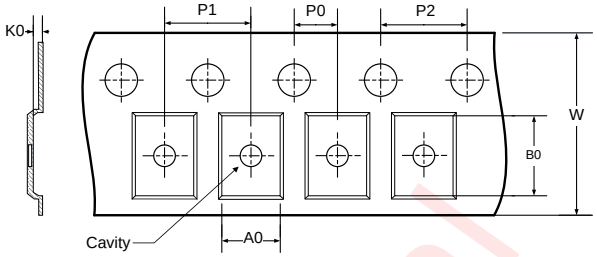
 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

AW21209QNR:

REEL DIMENSIONS

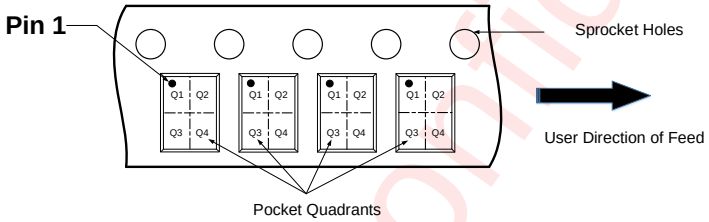


TAPE DIMENSIONS



A0：Dimension designed to accommodate the component width
B0：Dimension designed to accommodate the component length
K0：Dimension designed to accommodate the component thickness
W：Overall width of the carrier tape
P0：Pitch between successive cavity centers and sprocket hole
P1：Pitch between successive cavity centers
P2：Pitch between sprocket hole
D1：Reel Diameter
D0：Reel Width

QUADRANT ASSIGNMENTS FOR PIN 1 ORIENTATION IN TAPE



Note：The above picture is for reference only. Please refer to the value in the table below for the actual size

DIMENSIONS AND PIN1 ORIENTATION

D1 (mm)	D0 (mm)	A0 (mm)	B0 (mm)	K0 (mm)	P0 (mm)	P1 (mm)	P2 (mm)	W (mm)	Pin1 Quadrant
330.0	12.40	4.35	4.35	1.10	2.00	8.00	4.00	12.00	Q1

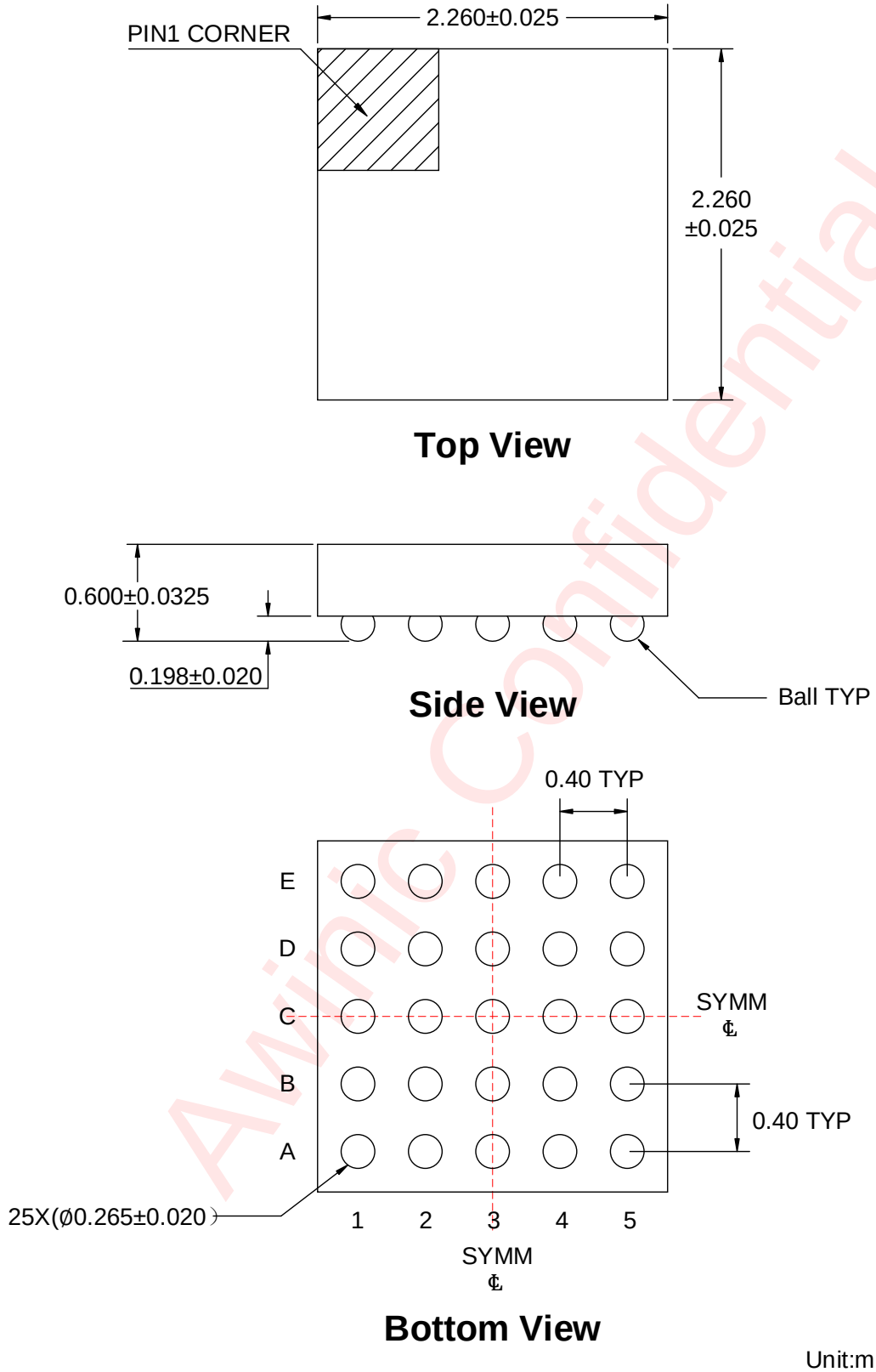
All dimensions are nominal

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Package Description

AW21209FOR:

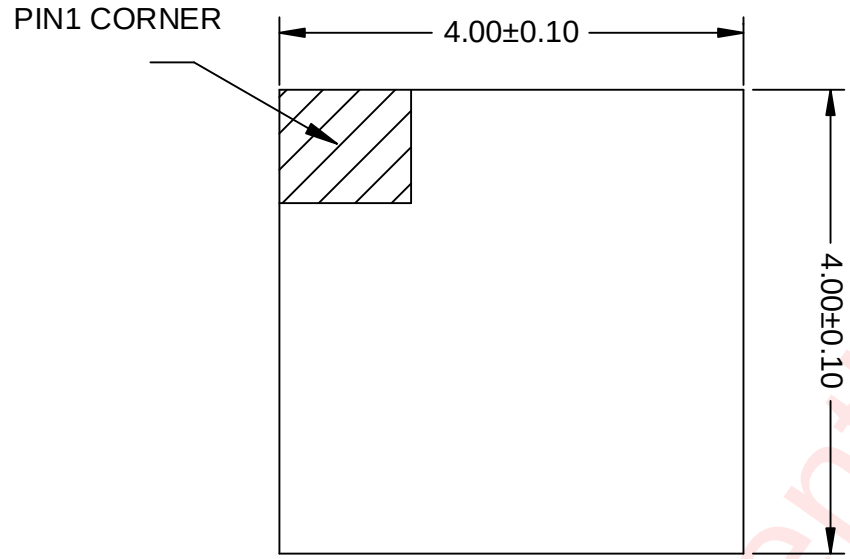


Unit:mm

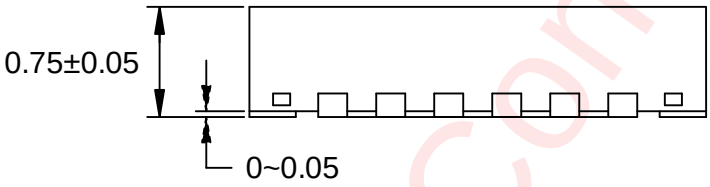
Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

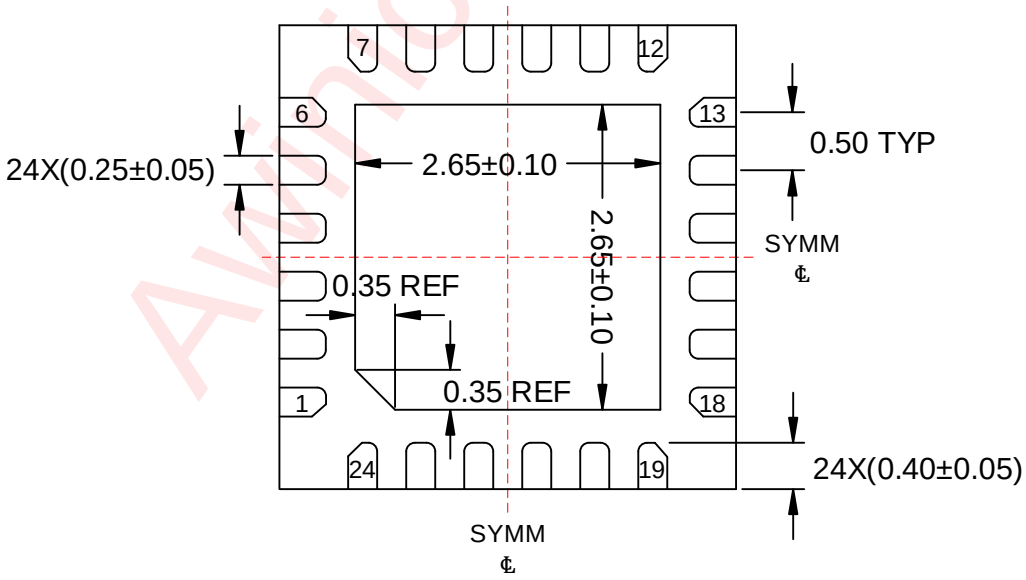
AW21209QNR:



Top View



Side View



Bottom View

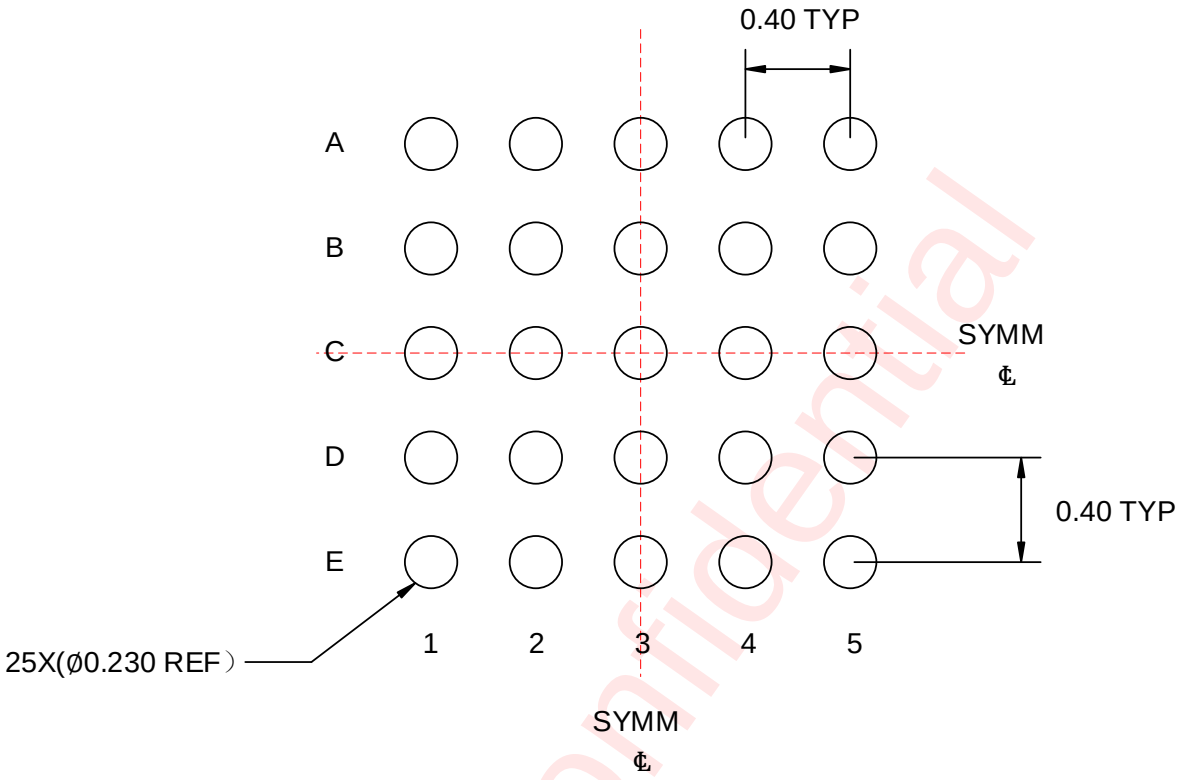
Unit:mm

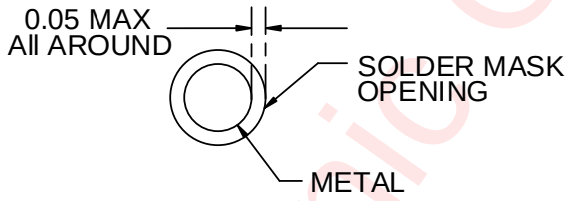
Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

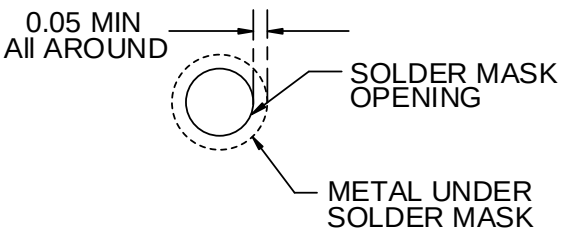
Land Pattern Data

AW21209FOR:





NON-SOLDER MASK DEFINED



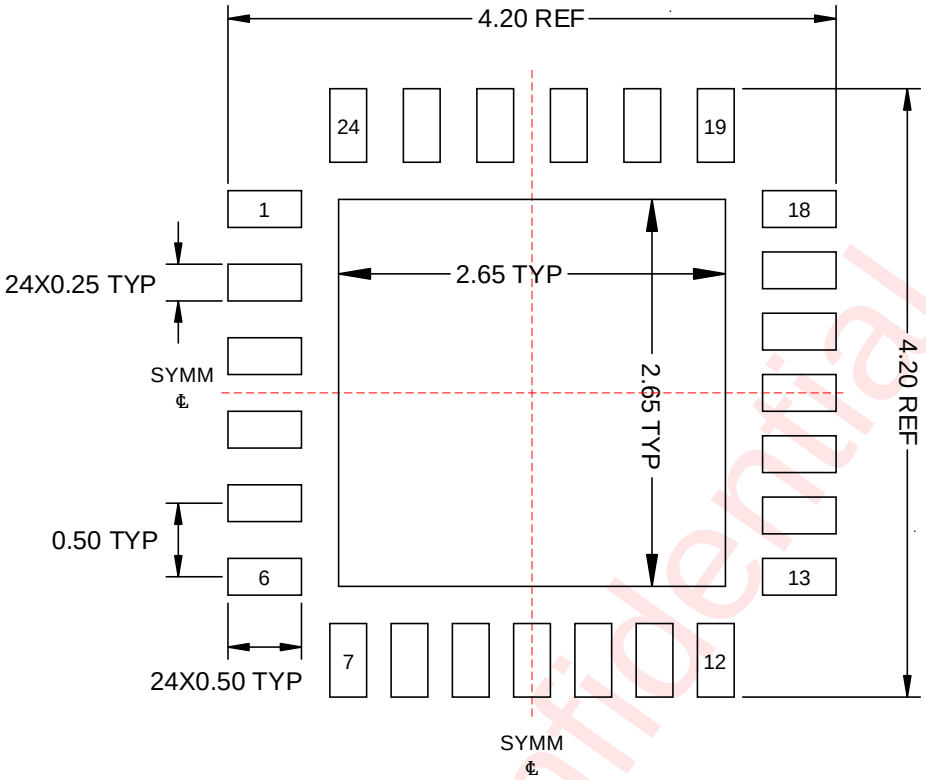
SOLDER MASK DEFINED

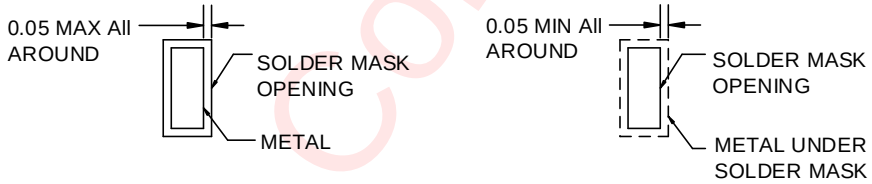
Unit: mm

Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

AW21209QNR:





NON SOLDER MASK DEFINED

SOLDER MASK DEFINED

Unit：mm

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Revision History

Version	Date	Change Record
V1.0	Mar 2025	Officially released
V1.1	Jul 2025	1.Update function block diagram (P4) 2.Add led mapping instructions (P26) 3.Update controlling led outputs (P32)
V1.2	Jul 2025	1.Update function block diagram (P4) 2.Add PWM frequency (P11) 3.Update the description of the clock chapter (P13) 4.Add ENHANCED_CFG and PWM_CFG register (P40\P62)
V1.3	Jul 2025	Add AW21209QNR

Awinic Confidential

© 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved

Disclaimer

All trademarks are the property of their respective owners. Information in this document is believed to be accurate and reliable. However, Shanghai AWINIC Technology Co., Ltd (AWINIC Technology) does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information.

AWINIC Technology reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. Customers shall obtain the latest relevant information before placing orders and shall verify that such information is current and complete. This document supersedes and replaces all information supplied prior to the publication hereof.

AWINIC Technology products are not designed, authorized or warranted to be suitable for use in medical, military, aircraft, space or life support equipment, nor in applications where failure or malfunction of an AWINIC Technology product can reasonably be expected to result in personal injury, death or severe property or environmental damage. AWINIC Technology accepts no liability for inclusion and/or use of AWINIC Technology products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications that are described herein for any of these products are for illustrative purposes only. AWINIC Technology makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

All products are sold subject to the general terms and conditions of commercial sale supplied at the time of order acknowledgement.

Nothing in this document may be interpreted or construed as an offer to sell products that is open for acceptance or the grant, conveyance or implication of any license under any copyrights, patents or other industrial or intellectual property rights.

Reproduction of AWINIC information in AWINIC data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. AWINIC is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of AWINIC components or services with statements different from or beyond the parameters stated by AWINIC for that component or service voids all express and any implied warranties for the associated AWINIC component or service and is an unfair and deceptive business practice. AWINIC is not responsible or liable for any such statements.

Awinic Confidential

 © 2025 Shanghai Awinic Technology Co., Ltd. All Rights Reserved