LP5562 Four-Channel RGB- or White-LED Driver With Internal Program Memory and Independent Channel Control

1 Features

- Four Independently Programmable LED Outputs With 8-Bit Current Setting (From 0 mA to 25.5 mA With 100-μA Steps) and 8-Bit PWM Control
- Typical LED Output Saturation Voltage 60 mV and Current Matching 1%
- Flexible PWM Control for LED Outputs
- Automatic Power-Save Mode With External Clock
- Three Program Execution Engines With Flexible Instruction Set
- Autonomous Operation With Program Execution Engines
- SRAM Program Memory for Lighting Pattern Programs
- DSBGA 12-Pin Package, 0.4-mm Pitch

2 Applications

- Fun Lights
- Indicator Lights
- Keypad RGB Backlighting and Phone Cosmetics

3 Description

The LP5562 is a four-channel LED driver designed to produce variety of lighting effects. The device has a program memory for creating variety of lighting sequences. When the program memory has been loaded, the LP5562 can operate independently without processor control.

The LP5562 is able to automatically enter power save mode, when LED outputs are not active and thus lowering current consumption.

Four independent LED channels have accurate programmable current sinks, from 0 mA to 25.5 mA with 100-μA steps and flexible PWM control. Each channel can be configured into each of the three program execution engines. Program execution engines have program memory for creating desired lighting sequences with PWM control.

The LP5562 has four pin-selectable I^2C addresses. This allows connecting up to four parallel devices in one I^2C bus. The device requires only one small, low-cost ceramic capacitor.

Device Information(1)

PART NUMBER	PACKAGE	BODY SIZE (MAX)
LP5562	DSBGA (12)	1.648 mm × 1.248 mm

(1) For all available packages, see the orderable addendum at the end of the data sheet.

Typical Application Diagram

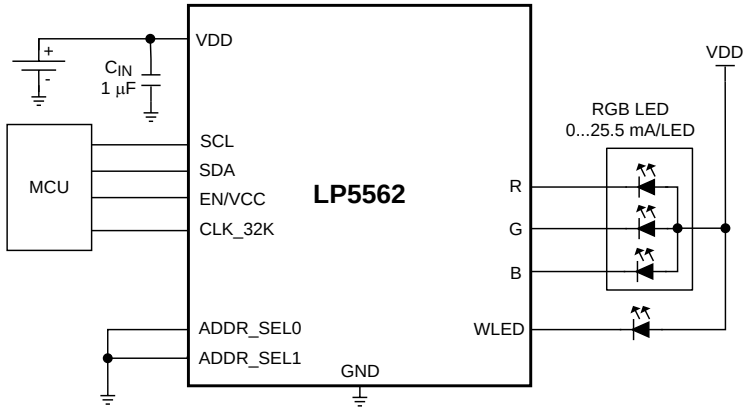


Table of Contents

4 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from Revision A (September 2015) to Revision B — **Page**

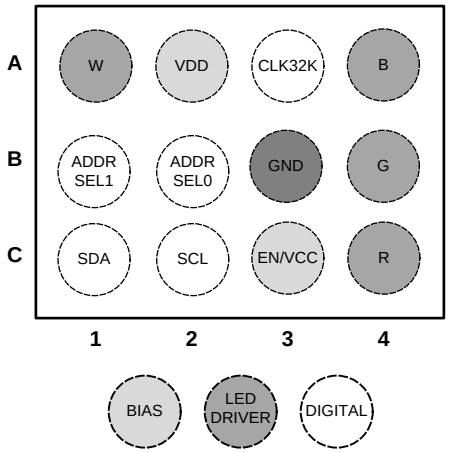
- Changed title for SEO/keyword improvement ... 1
- Changed $R_{\theta JA}$ from "68°C/W" to "85.9°C/W"; added additional required thermal information ... 4

Changes from Original (April 2013) to Revision A — **Page**

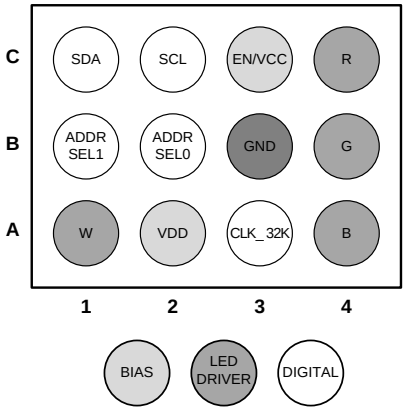
- Added *Pin Configuration and Functions* section, *ESD Ratings* table, *Feature Description* section, *Device Functional Modes, Application and Implementation* section, *Power Supply Recommendations* section, *Layout* section, *Device and Documentation Support* section, and *Mechanical, Packaging, and Orderable Information* section ... 1

5 Pin Configuration and Functions

YQE Package
12-Pin DSBGA
Top View



YQE Package
12-Pin DSBGA
Bottom View



Pin Functions

PIN		TYPE(1)	DESCRIPTION
NUMBER	NAME		
A1	W	A	LED driver current sink terminal
A2	VDD	P	Power supply
A3	CLK_32K	I	External 32-kHz clock input
A4	B	A	LED driver current sink terminal
B1	ADDR_SEL1	I	I^2C address selection pin
B2	ADDR_SEL0	I	I^2C address selection pin
B3	GND	G	Ground
B4	G	A	LED driver current sink terminal
C1	SDA	I/O	I^2C serial interface data input/output
C2	SCL	I	I^2C serial interface clock
C3	EN/VCC	P	Enable/Logic power supply
C4	R	A	LED driver current sink terminal

(1) A: Analog; G: Ground; P: Power pin; I: Input pin; I/O: Input/Output pin; O: Output pin.

6 Specifications

6.1 Absolute Maximum Ratings

over operating free-air temperature range (unless otherwise noted)(1)

	MIN	MAX	UNIT
V (V_{DD}, $V_{EN/VCC}$, R, G, B, W)	−0.3	6	V
Voltage on pins	−0.3	V_{DD} + 0.3 with 6 V maximum	V
Continuous power dissipation(2)	Internally limited		
Junction temperature, T_{J-MAX}		125	°C
Maximum lead temperature (soldering)	See(3)		
Storage temperature, T_{stg}	−65	150	°C

(1) Stresses beyond those listed under *Absolute Maximum Ratings* may cause permanent damage to the device. These are stress ratings only, which do not imply functional operation of the device at these or any other conditions beyond those indicated under *Recommended Operating Conditions*. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.
(2) Internal thermal shutdown circuitry protects the device from permanent damage. Thermal shutdown engages at T_J = 150°C (typical) and disengages at T_J = 130°C (typical).
(3) For detailed soldering specifications and information, refer to Texas Instruments *Application Note AN-1112 : DSBGA Wafer Level Chip Scale Package*.

6.2 ESD Ratings

			VALUE	UNIT
$V_{(ESD)}$	Electrostatic discharge	Human-body model (HBM), per ANSI/ESDA/JEDEC JS-001(1)	±1000	V
		Charged-device model (CDM), per JEDEC specification JESD22-C101(2)	±250	

(1) JEDEC document JEP155 states that 500-V HBM allows safe manufacturing with a standard ESD control process.
(2) JEDEC document JEP157 states that 250-V CDM allows safe manufacturing with a standard ESD control process.

6.3 Recommended Operating Conditions

over operating free-air temperature range (unless otherwise noted)

	MIN	NOM	MAX	UNIT
V_{DD}	2.7		5.5	V
$V_{EN/VCC}$	1.65		V_{DD}	V
Junction temperature, T_J	−40		125	°C
Ambient temperature, T_A(1)	−40		85	°C

(1) In applications where high power dissipation and/or poor package thermal resistance is present, the maximum ambient temperature may have to be derated. Maximum ambient temperature (T_{A-MAX}) is dependent on the maximum operating junction temperature ($T_{J-MAX-OP}$ = 125°C), the maximum power dissipation of the device in the application (P_{D-MAX}), and the junction-to ambient thermal resistance of the part/package in the application ($R_{\theta JA}$), as given by the equation: $T_{A-MAX} = T_{J-MAX-OP} – (R_{\theta JA} \times P_{D-MAX})$.

6.4 Thermal Information

THERMAL METRIC(1)		LP5562 YQE (DSBGA) 12 PINS	UNIT
$R_{\theta JA}$	Junction-to-ambient thermal resistance	85.9	°C/W
$R_{\theta JC(top)}$	Junction-to-case (top) thermal resistance	1.0	°C/W
$R_{\theta JB}$	Junction-to-board thermal resistance	15.3	°C/W
ψ_{JT}	Junction-to-top characterization parameter	0.6	°C/W
ψ_{JB}	Junction-to-board characterization parameter	15.4	°C/W

(1) For more information about traditional and new thermal metrics, see *Semiconductor and IC Package Thermal Metrics*.

6.5 Electrical Characteristics

Unless otherwise specified: limits for typical values are for T_A = 25°C and minimum and maximum limits apply over the operating ambient temperature range (−40°C < T_A < +85°C); V_{IN} = 3.6V, $V_{EN/VCC}$ = 1.8 V.(1)(2)(3)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
CURRENT CONSUMPTION AND OSCILLATOR ELECTRICAL CHARACTERISTICS						
I_{VDD}	Standby supply current	EN = 0 (pin), CHIP_EN = 0 (bit), external 32 kHz clock running or not running		0.2	2	µA
		EN = 1 (pin), CHIP_EN = 0 (bit), external 32 kHz clock not running		2		µA
		EN = 1 (pin), CHIP_EN = 0 (bit) External 32-kHz clock running		2.4		µA
	Normal mode supply current	LED drivers disabled		0.25		mA
		LED drivers enabled		1		mA
	Powersave mode supply current	External 32-kHz clock running		10		µA
		Internal oscillator running		0.25		mA
f_{OSC}	Internal oscillator frequency accuracy	T_A = 25°C	–4%		4%	
			–7%		7%	
LED DRIVER ELECTRICAL CHARACTERISTICS (R, G, B, W OUTPUTS)						
$I_{LEAKAGE}$	R, G, B, W pin leakage current	T_A = 25°C		0.1	1	µA
I_{MAX}	Maximum source current	Outputs R, G, B, W		25.5		mA
I_{OUT}	Accuracy of output current(4)	Output current set to 17.5 mA, V_{DD} = 3.6 V T_A = 25°C	–4%		4%	
		Output current set to 17.5 mA, V_{DD} = 3.6 V	–5%		5%	
I_{MATCH}	Matching(4)	Output current set to 17.5 mA, V_{DD} = 3.6V		1%	2%	
f_{LED}	LED PWM switching frequency	PWM_HF = 1		558		Hz
		PWM_HF = 0		256		
V_{SAT}	Saturation voltage(5)	Output current set to 17.5 mA T_A = 25°C		60	100	mV

(1) The electrical characteristics tables list ensured specifications under the listed recommended conditions except as otherwise modified or specified by the electrical characteristics test conditions and/or notes. Typical specifications are estimations only and are not verified by production testing.
(2) All voltages are with respect to the potential at the GND pins.
(3) Minimum and maximum limits are ensured by design, test, or statistical analysis. Typical numbers are not verified by production, but do represent the most likely norm.
(4) Output current accuracy is the difference between the actual value of the output current and programmed value of this current. Matching is the maximum difference from the average. For the constant current outputs on the part, the following are determined: the maximum output current (MAX), the minimum output current (MIN), and the average output current of all outputs (AVG). Two matching numbers are calculated: (MAX – AVG)/AVG and (AVG – MIN)/AVG. The largest number of the two (worst case) is considered the matching figure. Note that some manufacturers have different definitions in use.
(5) Saturation voltage is defined as the voltage when the LED current has dropped 10% from the set value.

6.6 Logic Interface Characteristics

Unless otherwise specified: limits for typical values are for T_A = 25°C and minimum and maximum limits apply over the operating ambient temperature range (−40°C < T_A < +85°C); V_{EN} = 1.65 V.

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
LOGIC INPUT EN						
V_{IL}	Input low level				0.5	V
V_{IH}	Input high level		1.2			V
I_I	Logic input current		–1		1	µA
t_{DELAY}	Input delay(1)			2		µs
LOGIC INPUT SCL, SDA, CLK_32K, ADDR_SEL0, ADDR_SEL1, V_{EN} = 1.8 V						
V_{IL}	Input low level				0.2 × V_{EN}	V
V_{IH}	Input high level		0.8 × V_{EN}			V
I_I	Input current		–1		1	µA
f_{CLK_32K}	Clock frequency			32		kHz
f_{SCL}	Clock frequency				400	kHz
LOGIC OUTPUT SDA						
V_{OL}	Output low level	I_{OUT} = 3 mA (pullup current)		0.3	0.5	V
I_L	Output leakage current				1	µA

(1) The I²C host should allow at least 1ms before sending data to the LP5562 after the rising edge of the enable line.

6.7 Recommended External Clock Source Conditions

over operating free-air temperature range (unless otherwise noted)(1)(2)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
LOGIC INPUT CLK_32K						
f_{CLK_32K}	Clock frequency			32.7		kHz
t_{CLKH}	High time		6			µs
t_{CLKL}	Low time		6			µs
t_r	Clock rise time	10% to 90%			2	µs
t_f	Clock fall time	90% to 10%			2	µs

(1) Specification is ensured by design and is not tested in production. V_{EN} = 1.65 V to V_{DD}.
(2) The ideal external clock signal for the LP5562 is a 0 V to V_{EN} 25% to 75% duty-cycle square wave. At frequencies above 32.7 kHz, program execution will be faster and at frequencies below 32.7 kHz program execution will be slower.

6.8 I²C Timing Requirements (SDA, SCL)

See(1)

		MIN	MAX	UNIT
f_{SCL}	Clock frequency		400	kHz
1	Hold time (repeated) START condition	0.6		µs
2	Clock low time	1.3		µs
3	Clock high time	600		ns
4	Setup time for a repeated START condition	600		ns
5	Data hold time	50		ns
6	Data setup time	100		ns
7	Rise time of SDA and SCL	20 + 0.1Cb	300	ns
8	Fall time of SDA and SCL	15 + 0.1Cb	300	ns
9	Set-up time for STOP condition	600		ns
10	Bus-free time between a STOP and a START condition	1.3		µs
Cb	Capacitive load for each bus line	10	200	pF

(1) Specification is ensured by design and is not tested in production. V_{EN} = 1.65 V to V_{DD}.

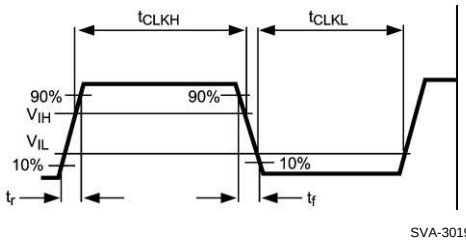


Figure 1. External Clock Timing

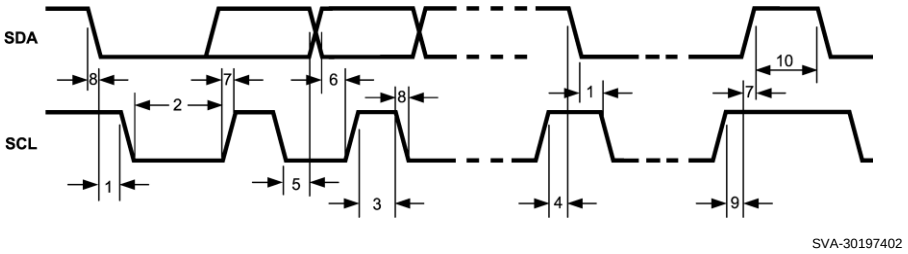


Figure 2. I²C Timing Parameters

6.9 Typical Characteristics: Current Consumption

Unless otherwise specified: V_{DD} = 3.6 V, V_{EN} = 3.3 V. Here are presented input current consumption measurements. Current consumption is measured during a LED blink program execution. Program code sets every LED output to full PWM value for 2 seconds and then PWM is set to 0 for 2 seconds. This is looped endlessly. 750 measurements are taken during one measurement cycle.

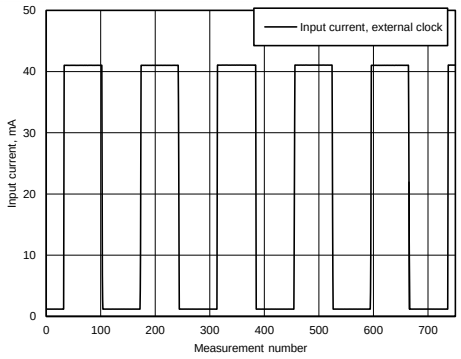


Figure 3. Input Current Consumption in Normal Mode With External Clock Running. 4 LEDs (RGBW) Set as Load. Every LED Driver Current Value Is Set to 10 mA.

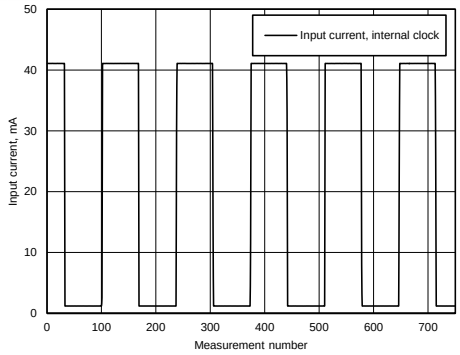


Figure 4. Input Current Consumption in Normal Mode With Internal Clock Running. 4 LEDs (RGBW) set as Load. Every LED Driver Current Value is Set to 10 mA .

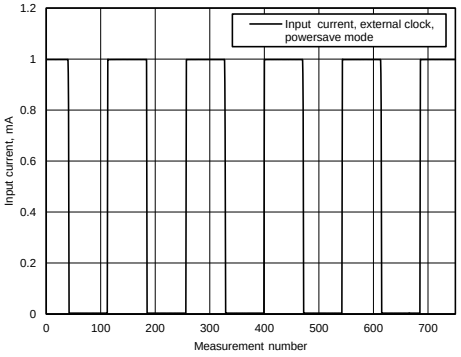


Figure 5. Input Current Consumption in Power Save Mode With External Clock Running, no LEDs as Load. All 4 LED Drivers are Enabled During Program Execution.

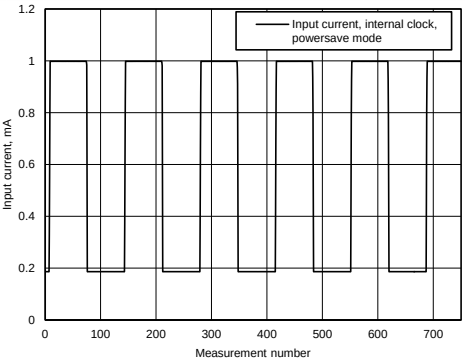


Figure 6. Input Current Consumption in Power Save Mode With Internal Clock Running, no LEDs as Load. All 4 LED Drivers are Enabled During Program Execution.

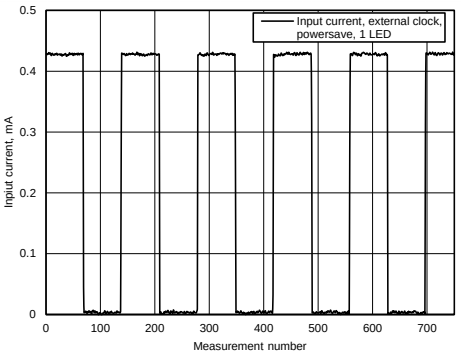


Figure 7. Input Current Consumption in Power Save Mode With External Clock Running, no LEDs as Load. Only 1 LED Driver is Enabled During Program Execution.

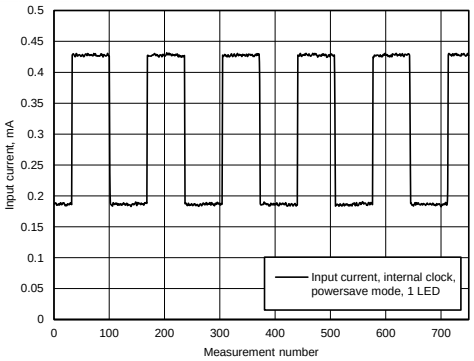


Figure 8. Input Current Consumption in Power Save Mode With Internal Clock Running, no LEDs as Load. Only 1 LED Driver is Enabled During Program Execution.

Typical Characteristics: Current Consumption (continued)

Unless otherwise specified: V_{DD} = 3.6 V, V_{EN} = 3.3 V. Here are presented input current consumption measurements. Current consumption is measured during a LED blink program execution. Program code sets every LED output to full PWM value for 2 seconds and then PWM is set to 0 for 2 seconds. This is looped endlessly. 750 measurements are taken during one measurement cycle.

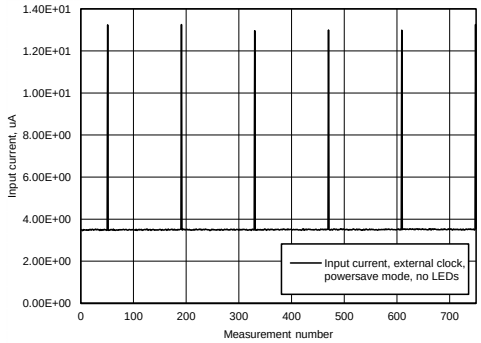


Figure 9. Input Current Consumption in Power Save Mode With External Clock Running, no LEDs as Load and no LED Drivers are Enabled During Program Execution.

6.10 Typical Characteristics: LED Output

LED driver typical performance images.

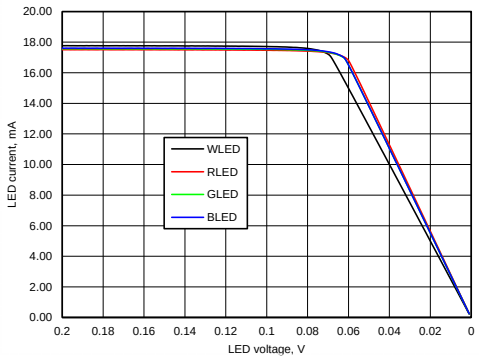


Figure 10. Every LED Driver Saturation Voltage, Current Setting 17.5 mA

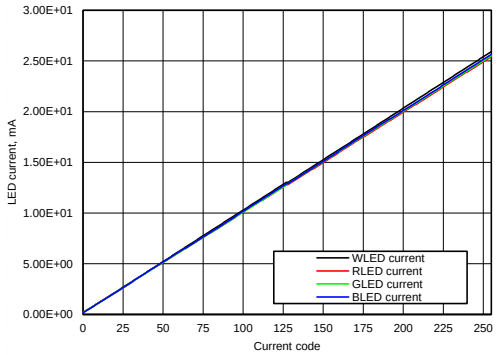


Figure 11. LED Driver Currents Compared to Current Setting Code

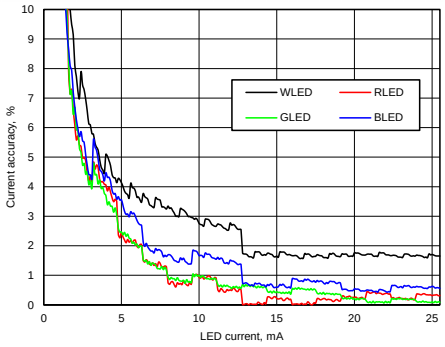


Figure 12. LED Driver Current Accuracy With Different Current Setting

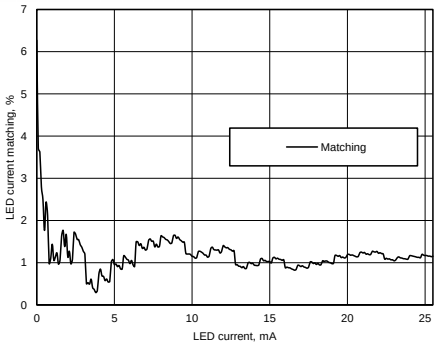


Figure 13. LED Driver Current Matching Between all LED Drivers With Different Current Setting

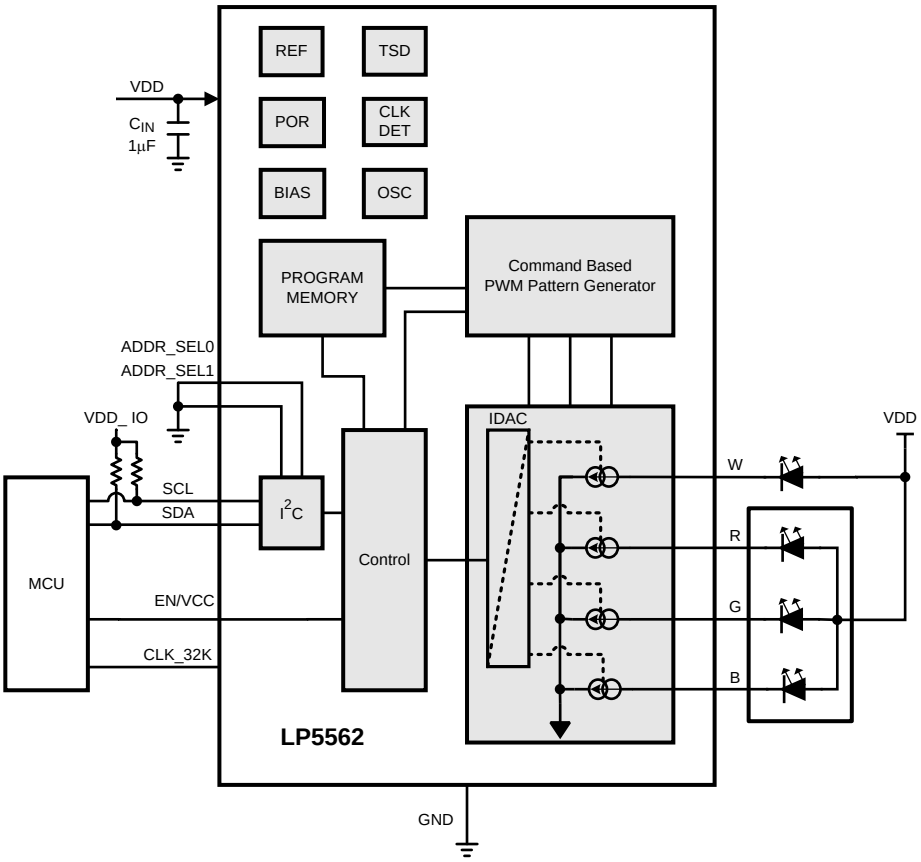
7 Detailed Description

7.1 Overview

The LP5562 is a RGBW LED driver for indicator LED and keypad lighting. The device has an internal program memory for creating a variety of lighting sequences. When the program memory has been loaded, the LP5562 can operate independently without processor control.

The device has 4 LED drivers that are constant current sinks with 8-bit current and 8-bit PWM control. The current sinks can be controlled via the three execution engines or direct PWM control. The execution engines have five different functions used to build lighting sequences: Ramp, Set PWM, Go to Start, Branch, or End Trigger. These control methods and functions are explained in detail in *Feature Description* and *Device Functional Modes*.

7.2 Functional Block Diagram



7.3 Feature Description

7.3.1 LED Drivers Operational Description

The LP5562 has 4 LED drivers that are constant current sinks with 8-bit current and 8-bit PWM control. Current is controlled from I^2C registers. PWM can be controlled with program execution engines or direct I^2C register writes.

7.3.1.1 LED Driver Current Control

LED driver output current can be programmed with I^2C register from 0 mA up to 25.5 mA. Current setting resolution is 100 µA (8-bit control).

Table 1. B_CURRENT Register (05h), G_CURRENT Register (06h), R_CURRENT Register (07h), W CURRENT Register (0Fh)

NAME	BIT	DESCRIPTION			
		CURRENT SETTING			
		bin	hex	dec	mA
CURRENT	7:0	0000 0000	00	0	0.0
		0000 0001	01	1	0.1
		0000 0010	02	2	0.2
		0000 0011	03	3	0.3
		0000 0100	04	4	0.4
		0000 0101	05	5	0.5
		0000 0110	06	6	0.6
		...	...	...	...
		1010 1111	**AF**	**175**	**17.5 (def)**
		...	...	...	...
		1111 1011	FB	251	25.1
		1111 1100	FC	252	25.2
		1111 1101	FS	253	25.3
		1111 1110	FE	254	25.4
		1111 1111	FF	255	25.5

7.3.1.2 Controlling LED Driver Output PWM

PWM can be controlled by either with program execution engines (1, 2 and 3) or via I^2C registers (02h for B, 03h for G, 04h for R and 0Eh for W).

Control of LED driver output PWM selection is managed with 2 bits for each LED output from register 70h. The Table 3 describes the selection options. With these bits for example all LED outputs can be controlled from one program execution engine.

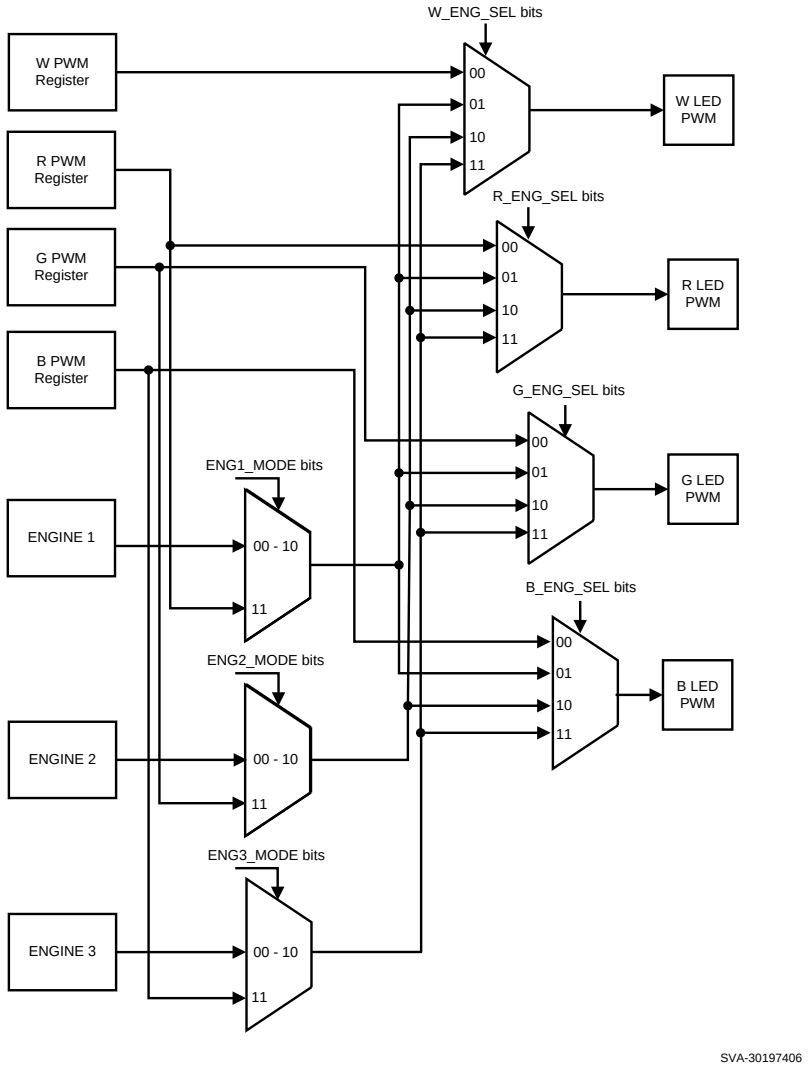


Figure 14. Controlling LED Outputs

The LED driver PWM control with 8-bit I²C register is defined in Table 2.

Table 2. LED Driver PWM Control Bits Register 70h

NAME	BIT	DESCRIPTION
PWM	7:0	LED PWM value during I²C control operation mode
		0000 0000 = 0% PWM
		1111 1111 = 100% PWM

If the LED driver outputs are controlled with engines, the engine adjusts the PWM according to the program code. However, when the engine *mode* bits are set to '11', the engine is set to *direct* mode. In *direct* mode the PWM controls of engines comes:

- Engine 1 PWM control comes from B PWM I²C register (02h)
- Engine 2 PWM control comes from G PWM I²C register (03h)
- Engine 3 PWM control comes from R PWM I²C register (04h)

When the engine *mode* bits are set to '11' along with the LED PWM Output selection bits, it is possible to control all LED outputs from one I²C register.

Table 3. LED PWM Output Selection Bits

B_ENG_SEL bits[1:0] G_ENG_SEL bits[3:2] R_ENG_SEL bits[5:4] W_ENG_SEL bits[7:6]	DESCRIPTION
00	Output is controlled via I^2C registers
01	ENG1_MODE and ENG1_EXEC register control LED output PWM instead of I^2C register
10	ENG2_MODE and ENG2_EXEC register control LED output PWM instead of I^2C register
11	ENG3_MODE and ENG3_EXEC register control LED output PWM instead of I^2C register

7.3.2 Direct I^2C Register PWM Control Example

- Device Start-up
  - Supply 3.6 V to VDD
  - Supply 1.8 V to EN
  - Wait 1 ms
  - Write to address 00h 0100 0000b (chip_en to '1')
  - Wait 500 μs (start-up delay)
- Use internal clock
  - Write to address 08h 0000 0001b (enable internal clock)
- Direct PWM control
  - Write to address 70h 0000 0000b (Configure all LED outputs to be controlled from I^2C registers)
- Write PWM values
  - Write to address 02h 1000 0000b (B driver PWM 50% duty cycle)
  - Write to address 03h 1100 0000b (G driver PWM 75% duty cycle)
  - Write to address 04h 1111 1111b (R driver PWM 100% duty cycle)

LEDs are turned on after the PWM values are written. Changes to the PWM value registers are reflected immediately to the LED brightness. Default LED current (17.5 mA) is used for LED outputs, if no other values are written.

PWM frequency is either 256 Hz or 558 Hz. Frequency is set with PWM_HF bit in register 08h. When PWM_HF is 0, the frequency is 256 Hz. When the PWM_HF bit is 1, the PWM frequency is 558 Hz. Brightness adjustment is either linear or logarithmic. This can be set with LOG_EN bit in register 00h. When LOG_EN = 0 linear adjustment scale is used and when LOG_EN = 1 logarithmic scale is used. By using logarithmic scale the visual effect seems linear to the eye. Register control bits are presented in following tables:

Table 4. ENABLE Register (00h)

NAME	BIT	DESCRIPTION
LOG_EN	7	Logarithmic PWM adjustment enable bit 0 = Linear adjustment 1 = Logarithmic adjustment

Table 5. CONFIG Register (08h)

NAME	BIT	DESCRIPTION
PWM_HF	6	PWM clock frequency 0 = 256 Hz 1 = 558 Hz

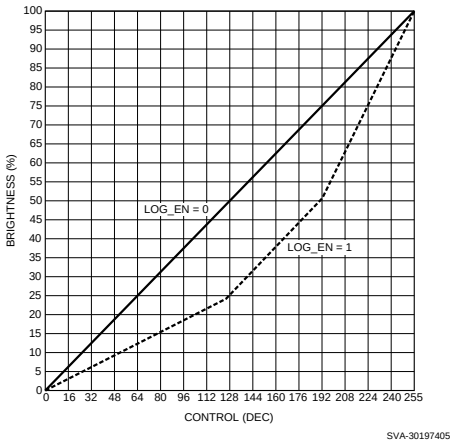


Figure 15. Logarithmic and Linear PWM Adjustment Curves

7.3.3 Program Execution Engines

Use of program execution engines is the other LED output PWM control method available in the LP5562. The device has 3 program execution engines. These engines create PWM controlled lighting patterns to the mapped LED outputs according to program codes developed by the user. Program coding is done using programming commands (see *Program Execution Engine Programming Commands*.) Programs are loaded into SRAM memory and engine control bits are used to run these programs autonomously. LED outputs can be mapped into these 3 engines with register 70h bit settings (see Table 3). The engines have different operation modes, program execution states, and program counters. Each engine has its own section of the SRAM memory.

7.3.3.1 Program Execution Engine States

Engine program execution is controlled from ENABLE register (00h). There are four different states for each engine, and these states are described in Table 6.

Table 6. ENABLE Register (00h)

NAME	BIT	DESCRIPTION
ENG1_EXEC	5:4	Engine 1 program execution **00b** = Hold: Wait until current command is finished then stop while EXEC mode is hold. PC can be read or written only in this mode. **01b** = Step: Execute instruction defined by current Engine 1 PC value, increment PC, and change ENG1_EXEC to 00b (Hold). **10b** = Run: Start at program counter value defined by current Engine 1 PC value. **11b** = Execute instruction defined by current Engine 1 PC value and change ENG1_EXEC to 00b (Hold).
ENG2_EXEC	3:2	Engine 2 program execution **00b** = Hold: Wait until current command is finished then stop while EXEC mode is hold. PC can be read or written only in this mode. **01b** = Step: Execute instruction defined by current Engine 2 PC value, increment PC, and change ENG2_EXEC to 00b (Hold). **10b** = Run: Start at program counter value defined by current Engine 2 PC value. **11b** = Execute instruction defined by current Engine 2 PC value and change ENG2_EXEC to 00b (Hold).
ENG3_EXEC	1:0	Engine 3 program execution **00b** = Hold: Wait until current command is finished then stop while EXEC mode is hold. PC can be read or written only in this mode. **01b** = Step: Execute instruction defined by current engine 3 PC value, increment PC, and change ENG3_EXEC to 00b (Hold). **10b** = Run: Start at program counter value defined by current engine 3 PC value. **11b** = Execute instruction defined by current engine 3 PC value and change ENG3_EXEC to 00b (Hold).

7.3.3.2 Program Execution Engine Operation Modes

Operation modes are defined in register address 01h. Each engine (1, 2, 3) operation mode can be configured separately. *Mode* registers are synchronized to a 32-kHz clock. Delay between consecutive I^2C writes to OP_MODE register (01h) need to be longer than 153 μs (typ).

Table 7. Operation Mode Register (OP_MODE (01h))

NAME	BIT	DESCRIPTION
ENG1_MODE	5:4	Engine 1 operation mode **00b** = Disabled, reset engine 1 PC **01b** = Load program to SRAM, reset engine 1 PC **10b** = Run program defined by ENG1_EXEC **11b** = Direct control from B PWM I^2C register, reset engine 1 PC
ENG2_MODE	3:2	Engine 2 operation mode **00b** = Disabled, reset engine 2 PC **01b** = Load program to SRAM, reset engine 2 PC **10b** = Run program defined by ENG2_EXEC **11b** = Direct control from G PWM I^2C register, reset engine 2 PC
ENG3_MODE	1:0	Engine 3 operation mode **00b** = Disabled, reset engine 3 PC **01b** = Load program to SRAM, reset engine 3 PC **10b** = Run program defined by ENG3_EXEC **11b** = Direct control from R PWM I^2C register, reset engine 3 PC

7.3.3.2.1 Operation Modes

- ***Disabled***
  - Each channel can be configured to disabled mode. For the current engine mapped LED output brightness will be 0 during this mode. Disabled mode resets respective engine's PC.
- ***Load program***
  - LP5562 can store 16 commands for each engine (1, 2, 3). Each command consists of 16 bits. Because one register has only 8 bits, one command requires two I^2C register addresses. In order to reduce program load time the LP5562 supports address auto increment. Register address is incremented after each 8 data bits. The whole program memory can be written in one I^2C write sequence. Program memory is defined in the LP5562 register table, from address 10h to address 2Fh for engine 1, from address 30h to address 4Fh for engine 2, and from address 50h to address 6Fh for engine 3. In order to access program memory at least one channel operation mode needs to be *load program*.
  - SRAM memory writes are allowed only to the channel in *load program* mode. All engines are in hold while one or several engines are in *load program* mode, and PWM values are frozen for the engines which are not in *load program*mode. Program execution continues when all engines are out of *load program* mode. *Load program* mode resets respective engine's Program Counter (PC).
- ***Run program***
  - *Run program* mode executes the commands defined in program memory for respective engine (1, 2, 3). Execution register bits in ENABLE register (00h) define how the program is executed. The program start position can be programmed to Program Counter register (see Table 8). By manually selecting the PC start value, user can write different lighting sequences to the SRAM memory, and select appropriate sequence with the PC register. If program counter runs to end (15), next command will be executed from program location 0. If internal clock is used in the *run program* mode, operation mode needs to be written disabled (00b) before disabling the chip (with CHIP_EN bit or EN pin) to ensure that the sequence starts from the correct program counter (PC) value when restarting the sequence. PC registers are synchronized to 32 kHz clock. Delay between consecutive I^2C writes to Program Counter (PC) registers (09h, 0Ah, 0Bh) need to be longer than 153μs (typ.).
  - Execution registers are synchronized to 32kHz clock. Delay between consecutive I^2C writes to ENABLE register (00h) need to be longer than 488μs (typ.).
  - Note that entering LOAD program or Direct Control Mode from RUN PROGRAM mode is not allowed. Engine execution mode should be set to Hold, and Operation Mode to disabled, when changing operation mode from RUN mode.
- ***Direct control***
  - In *Direct control* mode the engine PWM output is controlled by R, G and B PWM I^2C registers.

– When engine 1 is in *Direct control* mode, the engine 1 PWM output is controlled by B PWM I^2C register (02h).
– When engine 2 is in *Direct control* mode, the engine 2 PWM output is controlled by G PWM I^2C register (03h).
– When engine 3 is in *Direct control* mode, the engine 3 PWM output is controlled by R PWM I^2C register (04h).

7.3.3.3 Program Execution Engine Program Counter (PC)

Program execution engine Program Counter tells the current program code command, which engine is executing. By setting the program counter value before starting the engine execution, user can set the starting point of the program execution.

Table 8. Engine1 PC Register (09h), Engine2 PC Register (0Ah), Engine3 PC Register (0Bh)

NAME	BIT	DESCRIPTION
PC	3:0	Program counter value from 0 to 15d

7.3.3.4 Program Execution Engine Programming Commands

The LP5562 has three independent programmable engines (1, 2, 3). Trigger connections between engines are common for all engines. All engines have own program memory sections for storing LED lighting patterns. Brightness control and patterns are done with 8-bit PWM control (256 steps) to get accurate and smooth color control. Program execution is timed with a 32.7-kHz clock. This clock can be generated internally or an external 32-kHz clock can be connected to the CLK_32K pin. Using an external clock enables synchronization of LED timing to this clock rather than an internal clock. Selection of the clock is made with address 08H bits INT_CLK_EN and CLK_DET_EN. See *External Clock* for details. Supported commands are listed in Table 9.

Table 9. LED Controller Programming Commands(1)

Command	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RampWait	0	Pre-scale	Step time						Sign	Increment (number of steps)						
Set PWM	0	1	0						PWM Value							
Go to Start	0	0	0						0	0	0	0	0	0	0	0
Branch	1	0	1	Loop count						x	Step / command number					
End	1	1	0	Int	Reset	X										
Trigger	1	1	1	X	X	X	Wait for trigger on engines 1, 2, 3			X	X	X	Send trigger to engines 1,2, 3			X

(1) X means do not care whether 1 or 0.

7.3.3.4.1 Ramp/Wait

The ramp command generates a PWM ramp starting from current value. At each ramp step the output is incremented by one. Time for one step is defined with Prescale and Step time bits. Minimum time for one step is 0.49 ms and maximum time is 63 × 15.6 ms = 1 second/step, so it is possible to program very fast and also very slow ramps. Increment value defines how many steps are taken in one command. Number of actual steps is Increment + 1. Maximum value is 127d, which corresponds to half of full-scale (128 steps). If during ramp command PWM reaches minimum/maximum (0/255) ramp command will be executed to the end and PWM will stay at minimum/maximum. This enables the ramp command to be used as combined ramp and wait command in a single instruction.

The ramp command can be used as wait instruction when increment is zero.

Setting register 00h bit LOG_EN sets the scale as either linear to logarithmic. When LOG_EN = 0, linear scale is used, and when LOG_EN = 1, logarithmic scale is used. By using logarithmic scale the visual effect of the ramp command seems linear to the eye.

Table 10. Ramp/Wait Command

Ramp/Wait command															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Pre-scale	Step time						Sign	Increment						

Table 11. Ramp/Wait Command Bits

NAME	VALUE(d)	DESCRIPTION
Prescale	0	Divides master clock (32.768 Hz) by 16 = 2048 Hz, 0.49 ms cycle time
	1	Divides master clock (32.768 Hz) by 512 = 64 Hz, 15.6 ms cycle time
Step time	1-63	One ramp increment done in (step time) x (clock after prescale) Note: 0 means set PMW command.
Sign	0	Increase PWM output
	1	Decrease PWM output
Increment	0-127	The number of steps is Increment + 1. Note: 0 is a wait instruction.

For example, if following parameters are used for ramp:

- Prescale = 1 ≥ cycle time = 15.6 ms
- Step time = 2 ≥ time = 15.6 ms × 2 = 31.2 ms
- Sign = 0 ≥ rising ramp Increment = 4 ≥ 5 cycles

Ramp command will be: 0100 0010 0000 0100b = 4204h

If current PWM value is 3, and the first command is as described above, the next command is a ramp with otherwise same the parameters, but with Sign = 1 (Command = 4284h), the result will be like in the following figure:

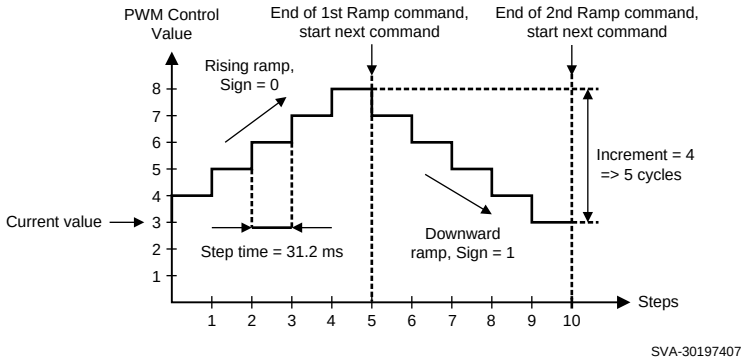


Figure 16. Example of 2 Sequential Ramp Commands

7.3.3.4.2 Set PWM

Set PWM output value from 0 to 255. Command takes sixteen 32-kHz clock cycles (= 488 μs). Setting register 00h bit LOG_EN sets the scale from linear to logarithmic.

Table 12. Set PWM Command Bits

Set PWM command															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	0	0	0	PWM value							

7.3.3.4.3 Go-to-Start

Go-to-start command resets the Program Counter register and continues executing program from the 00h location. Command takes sixteen 32-kHz clock cycles. Note that default value for all program memory registers is 0000h, which is Go-to-Start command.

Table 13. Go-to-Start Command Bits

Go-to-Start command															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

7.3.3.4.4 Branch

When branch command is executed, the 'step number' value is loaded to PC, and program execution continues from this location. Looping is done by the number defined in loop count parameter. Nested looping is supported (loop inside loop). The number of nested loops is not limited. Command takes sixteen 32-kHz clock cycles.

Table 14. Branch Command (1)

Branch command															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	1	Loop count						X	X	X	Step number			

(1) X means do not care whether 1 or 0

Table 15. Branch Command Bits

NAME	VALUE	DESCRIPTION
loop count	0-63	The number of loops to be done. 0 means infinite loop.
step number	0-15	The step number to be loaded to program counter.

7.3.3.4.5 End

End program execution resets the program counter and sets the corresponding EXEC register to 00b (hold). Command takes sixteen 32 kHz clock cycles.

Table 16. End Command (1)

End command															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	0	int	reset	X	X	X	X	X	X	X	X	X	X	X

(1) X means do not care whether 1 or 0.

Table 17. End Command Bits

NAME	VALUE	DESCRIPTION
int	0	No interrupt will be sent.
	1	Send interrupt by setting corresponding status register bit high to notify that program has ended. Interrupt can only be cleared by reading interrupt status register 0Ch.

Table 17. End Command Bits (continued)

NAME	VALUE	DESCRIPTION
reset	0	Keep the current PWM value.
	1	Set PWM value to 0.

7.3.3.4.6 Trigger

Wait or send triggers can be used to synchronize operation between different engines. The send-trigger command takes sixteen 32-kHz clock cycles; the wait-for-trigger command takes at least sixteen 32-kHz clock cycles. The receiving engine stores sent triggers. Received triggers are cleared by wait for trigger command if received triggers match to engines defined in the command. Engine waits until all defined triggers have been received.

Table 18. Trigger Command(1)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	X	X	X	wait trigger <2:0>			X	X	X	send trigger <2:0>			X
						ENG3	ENG2	ENG1				ENG3	ENG2	ENG1	

(1) X means do not care whether 1 or 0.

Table 19. Trigger Command Bits

NAME	VALUE(d)	DESCRIPTION
wait trigger<2:0>	0-7	Wait for trigger for the engine(s) defined. Several triggers can be defined in the same command. Bit 0 is engine 1, bit 1 is engine2, bit 2 is engine 3.
send trigger<2:0>	0-7	Send trigger for the engine(s) defined. Several triggers can be defined in the same command. Bit 0 is engine 1, bit 1 is engine2, bit 2 is engine 3.

7.3.3.5 Program Load and Execution Example

- Start up device and configure device to SRAM write mode
  - Supply 3.6V to VDD
  - Supply 1.8V to EN
  - Wait 1 ms
  - Generate 32 kHz clock to CLK_32K pin
  - Write to address 00h 0100 0000b (enable device)
  - Wait 500 μs (startup delay)
  - Write to address 01h 0001 0000b (configure engine 1 into 'Load program to SRAM' mode)
- Program load to SRAM
  - Write to address 10h 0000 0011b (1st ramp command 8MSB)
  - Write to address 11h 0111 1111b (1st ramp command 8 LSB)
  - Write to address 12h 0100 1101b (1st wait command 8 MSB)
  - Write to address 13h 0000 0000b (1st wait command 8 LSB)
  - Write to address 14h 0000 0011b (2nd ramp command 8 MSB)
  - Write to address 15h 1111 1111b (2nd ramp command 8 LSB)
  - Write to address 16h 0110 0000b (2nd wait command 8 MSB)
  - Write to address 17h 0000 0000b (2nd wait command 8 LSB)
- Enable Power Save and use external 32 kHz clock
  - Write to address 08h 0010 0000b (enable powersave, use external clock)
- Run program
  - Write to address 01h 0010 0000b (Configure LED controller operation mode to "Run program" in engine 1)
  - Write to address 00h 0110 0000b (Configure program execution mode from "Hold" to "Run" in engine 1)

The LP5562 will generate a 1100 ms long LED pattern which will be repeated infinitely. The LED pattern is illustrated in the figure below.

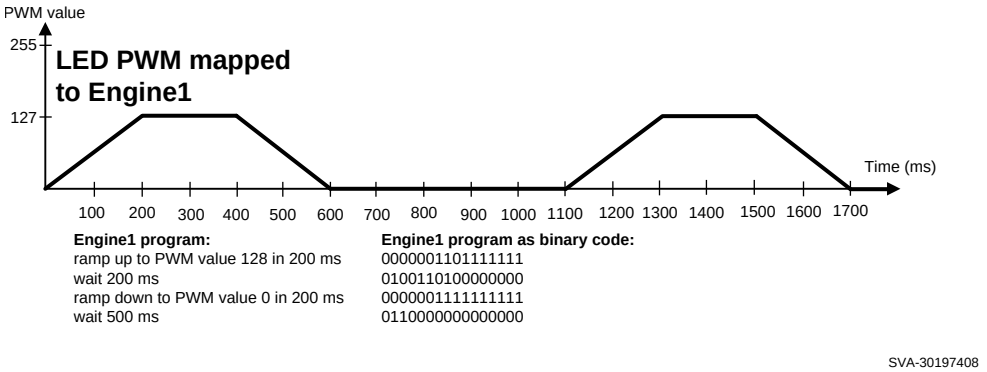


Figure 17. LED Lighting Pattern and Code for Program Load and Execution Example

7.3.4 Power-Save Mode

Automatic power save mode is enabled when the PS_EN bit in register address 08h is 1. Almost all analog blocks are powered down in power save, if an external clock is used. However, if an internal clock has been selected, only the LED drivers are disabled during power save since the digital part of the LED controller need to remain active. During program execution the LP5562 can enter power-save mode if there is no PWM activity in engine controlled outputs. To prevent short power-save sequences during program execution, the LP5562 has a command look-ahead filter. In each instruction cycle every engine commands are analyzed, and if there is sufficient time left with no PWM activity, the device will enter power save. In power save program execution continues uninterruptedly. When a command that requires PWM activity is executed, fast internal startup sequence will be started automatically. The following tables describe commands and conditions that can activate power save. All engines need to meet power-save conditions in order to enable power save.

Table 20. Engine Operation Mode and Power Save

ENGINE OPERATION MODE	POWER SAVE CONDITION
00b	Disabled mode enables power save
01b	Load program to SRAM mode prevents power save.
10b	Run program mode enables power save if there is no PWM activity and command look-ahead filter condition is met.
11b	Direct control mode enables power save if there is no PWM activity.

Table 21. Engine Commands and Power Save

COMMAND	POWER SAVE CONDITION
Wait	No PWM activity and current command wait time longer than 50 ms. If prescale = 1 then wait time needs to be longer than 80 ms.
Ramp	Ramp command PWM value reaches minimum 0 and current command execution time left more than 50 ms. If prescale = 1 then time left needs to be more than 80 ms.
Trigger	No PWM activity during wait for trigger command execution.
End	No PWM activity or Reset bit = 1.
Set PWM	Enables power save if PWM set to 0 and next command generates at least 50 ms wait.
Other commands	No effect to power save.

7.3.5 External Clock

The presence of an external clock can be detected by the LP5562. Program execution is clocked with an internal 32-kHz clock or with an external clock. Clocking is controlled with register address 08h bits, INT_CLK_EN, and CLK_DET_EN as seen in Table 22.

An external clock can be used if clock is present at the CLK_32K pin. The external clock frequency must be 32 kHz for the program execution PWM timing to be as specified. If higher or lower frequency is used, it will affect the program engine execution speed. If a clock frequency other than 32 kHz is used, the program execution timings must be scaled accordingly.

The LP5562 has automatic external clock detection. The external clock detector block only detects too low clock frequency (< 4 kHz), but it is recommended not to use external clock below 20 kHz. If external clock frequency is higher than specified, the external clock detector notifies that external clock is present. External clock status can be checked with read only bit EXT_CLK_USED in register address 0Ch, when the external clock detection is enabled (CLK_DET_EN bit = high). If EXT_CLK_USED = 1, then the external clock is detected and it is used for timing, if automatic clock selection is enabled.

If an external clock is stuck-at-zero or stuck-at-one, or the clock frequency is too low, the clock detector indicates that external clock is not present.

If an external clock is not used on the application, CLK_32K pin should be connected to GND to prevent floating of this pin and extra current consumption.

Table 22. CONFIG Register (08h)

NAME	BIT	DESCRIPTION
CLK_DET_EN, INT_CLK_EN	1:0	LED Controller clock source 00b = External clock source (CLK_32K) 01b = Internal clock 10b = Automatic selection 11b = Internal clock

7.3.6 Thermal Shutdown

If the LP5562 reaches thermal shutdown temperature (150°C typical) the device operation is disabled and the device state is in STARTUP mode, until no thermal shutdown event is present. Device will enter Normal mode when temperature drops below 130°C (typical) degrees.

Fault is cleared when thermal shutdown disappears.

7.3.7 Logic Interface Operational Description

The LP5562 features a flexible logic interface for connecting to processor and peripheral devices. Communication is done with the I^2C-compatible interface, and different logic input/output pins makes it possible to synchronize operation of several devices.

7.3.8 I/O Levels

I^2C interface, CLK_32K. ADDR_SEL0, and ADDR_SEL1 pins input levels are defined by voltage in EN pin. Using the EN pin as a voltage reference for logic inputs simplifies PCB routing and eliminates the need for a dedicated VIO pin. The following block diagram describes EN pin connections.

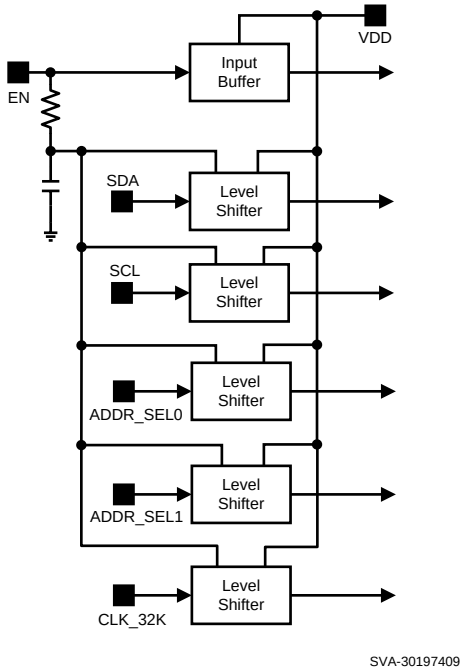


Figure 18. Using EN Pin as Digital I/O Voltage Reference

7.3.9 ADDR_SEL0, ADDR_SEL1 Pins

The ADDR_SEL0 and ADDR_SEL1 pins define the device I^2C address. Pins are referenced to EN pin signal level. See I^2C Addresses for I^2C address definitions.

7.3.10 CLK_32 Pin

The CLK_32K pin is used for connecting an external 32-kHz clock to LP5562. An external clock can be used to synchronize the sequence engines of several LP5562 devices. Using an external clock can also improve automatic power save mode efficiency, because an internal clock can be switched off automatically when device has entered power-save mode, and an external clock is present. Device can be used without the external clock. If external clock is not used on the application, the CLK_32K pin should be connected to GND to prevent floating of this pin and extra current consumption.

7.4 Device Functional Modes

RESET: In the *reset* mode all the internal registers are reset to the default values. Reset is done always if FFh is written to Reset Register (0Dh) or internal Power On Reset is activated. Power On Reset (POR) will activate when supply voltage is connected or when the supply voltage V_{DD} falls below 1.5 V (typical). Once V_{DD} rises above 1.9 V (typical), POR will inactivate and the chip will continue to the *standby* mode. CHIP_EN control bit is low after POR by default.

STANDBY: The *standby* mode is entered if the register bit CHIP_EN or EN pin is *low* and Reset is not active. This is the low power consumption mode, when all circuit functions are disabled. Registers can be written in this mode if EN pin is high. Control bits are effective after start up.

START-UP: When CHIP_EN bit is written high and EN pin is high, the *internal startup sequence* powers up all the needed internal blocks (VREF, Bias, Oscillator etc.). Start-up delay after setting EN pin *high* is 1 ms (typical). Start-up delay after setting chip_en bit to '1' is 500 μs (typical). If the device temperature rises too high, the Thermal Shutdown (TSD) disables the device operation and the device state is in *start-up* mode, until no thermal shutdown event is present.

NORMAL: During *normal* mode the user controls the device using the Control Registers. If EN pin is set low, the CHIP_EN bit is reset to 0.

Device Functional Modes (continued)

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

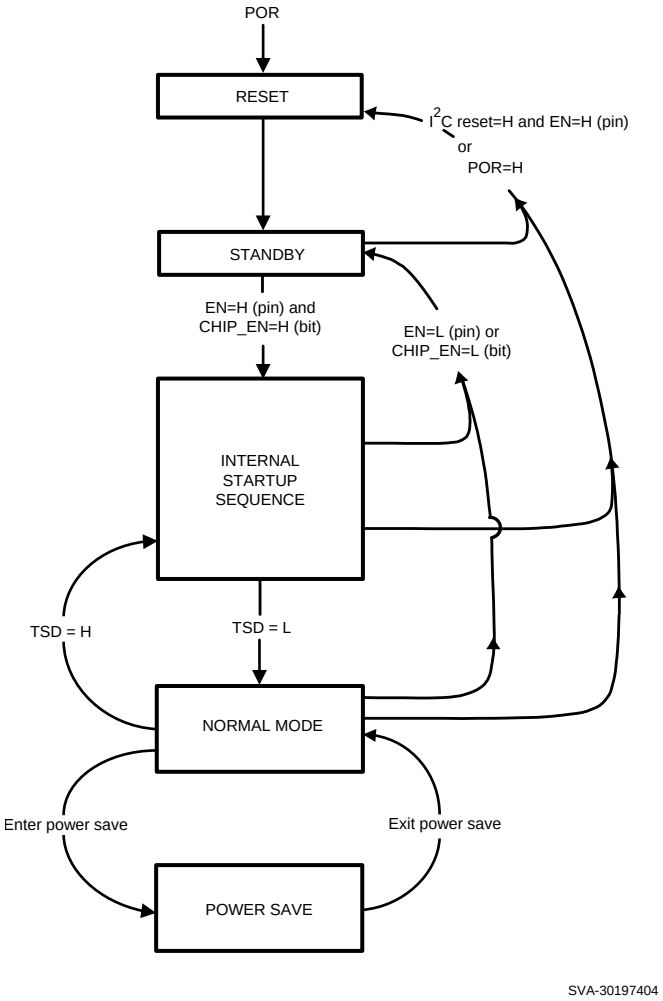


Figure 19. Modes of Operation

7.5 Programming

7.5.1 SRAM Memory

In the LP5562 there is a SRAM memory reserved for storing the LED lighting programs. Each engine has its own section of the memory so that engine 1 has registers 10h to 2Fh, engine 2 has registers 30h to 4Fh, and engine 3 has registers 50h to 6Fh. For each engine 16 engine commands (16-bit) can be stored. Each 16-bit command takes up two I^2C registers.

Table 23. SRAM Memory Registers

Address	Register	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
10h	Prog mem ENG1	COMMAND1_ENG1[15:8]							
11h	Prog mem ENG1	COMMAND1_ENG1[7:0]							
...									
2Eh	Prog mem ENG1	COMMAND16_ENG1[15:8]							
2Fh	Prog mem ENG1	COMMAND16_ENG1[7:0]							
30h	Prog mem ENG2	COMMAND1_ENG2[15:8]							
31h	Prog mem ENG2	COMMAND1_ENG2[7:0]							
...									
4Eh	Prog mem ENG2	COMMAND16_ENG2[15:8]							
4Fh	Prog mem ENG2	COMMAND16_ENG2[7:0]							
50h	Prog mem ENG3	COMMAND1_ENG3[15:8]							
51h	Prog mem ENG3	COMMAND1_ENG3[7:0]							
...									
6Eh	Prog mem ENG3	COMMAND16_ENG3[15:8]							
6Fh	Prog mem ENG3	COMMAND16_ENG3[7:0]							

When downloading a program to the SRAM engine modes need to be set to Load mode (see Table 6). While loading sequential I^2C writing can be used (repeated start see Figure 24). However, please note that sequential read of the SRAM is not possible.

7.5.2 I^2C-Compatible Serial Bus Interface

7.5.2.1 Interface Bus Overview

The I^2C compatible synchronous serial interface provides access to the programmable functions and registers on the device. This protocol uses a two-wire interface for bidirectional communications between the IC's connected to the bus. The two interface lines are the Serial Data Line (SDA), and the Serial Clock Line (SCL). These lines should be connected to a positive supply, via a pullup resistor and remain HIGH even when the bus is idle.

Every device on the bus is assigned a unique address and acts as either a Master or a Slave depending on whether it generates or receives the serial clock (SCL).

7.5.2.2 Data Transactions

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 permits a single data line to transfer both command/control information and data using the synchronous serial clock.

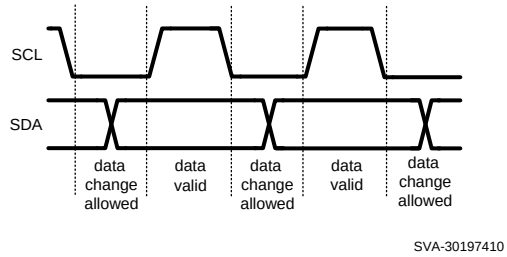


Figure 20. Data Validity

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. The following sections provide further details of this process.

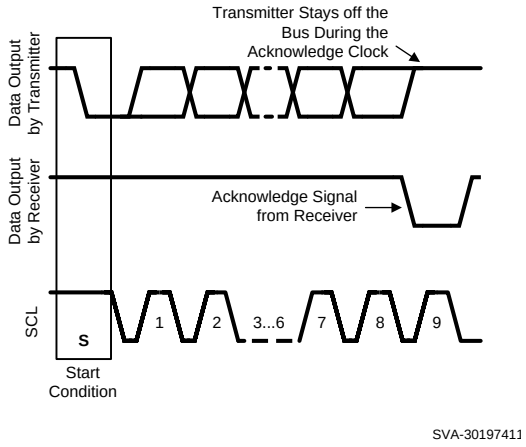


Figure 21. Acknowledge Signal

The Master device on the bus always generates the Start and Stop Conditions (control codes). After a Start Condition is generated, the bus is considered busy and it retains this status until a certain time after a Stop Condition is generated. A high-to-low transition of the data line (SDA) while the clock (SCL) is high indicates a Start Condition. A low-to-high transition of the SDA line while the SCL is high indicates a Stop Condition.

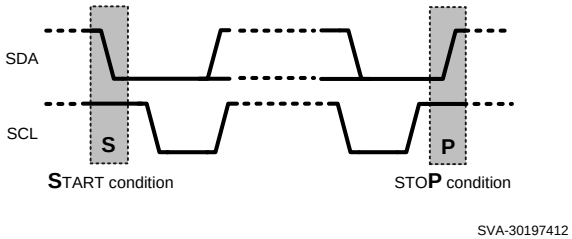


Figure 22. Start and Stop Conditions

In addition to the first Start Condition, a repeated Start Condition can be generated in the middle of a transaction. This allows another device to be accessed, or a register read cycle.

7.5.2.3 Acknowledge Cycle

The Acknowledge Cycle consists of two signals: the acknowledge clock pulse the master sends with each byte transferred, and the acknowledge signal sent by the receiving device.

The master generates the acknowledge clock pulse on the ninth clock pulse of the byte transfer. The transmitter releases the SDA line (permits it to go high) to allow the receiver to send the acknowledge signal. The receiver must pull down the SDA line during the acknowledge clock pulse and ensure that SDA remains low during the high period of the clock pulse, thus signaling the correct reception of the last data byte and its readiness to receive the next byte.

7.5.2.4 Acknowledge After Every Byte Rule

The master generates an acknowledge clock pulse after each byte transfer. The receiver sends an acknowledge signal after every byte received.

There is one exception to the “acknowledge after every byte” rule. When the master is the receiver, it must indicate to the transmitter an end of data by not-acknowledging (“negative acknowledge”) the last byte clocked out of the slave. This “negative acknowledge” still includes the acknowledge clock pulse (generated by the master), but the SDA line is not pulled down.

7.5.2.5 Addressing Transfer Formats

Each device on the bus has a unique slave address. The LP5562 operates as a slave device with the 7-bit address. LP5562 I^2C address is pin selectable from four different choices. If 8-bit address is used for programming, the 8th bit is 1 for read and 0 for write. Table 24 shows the 8-bit I^2C addresses.

Table 24. I^2C Addresses

ADDR_SEL [1:0]	I 2 C ADDRESS WRITE (8 bits)	I 2 C ADDRESS READ (8 bits)
00	0110 0000 = 60h	0110 0001 = 61h
01	0110 0010 = 62h	0110 0011 = 63h
10	0110 0100 = 64h	0110 0101 = 65h
11	0110 0110 = 66h	0110 0111 = 67h

Before any data is transmitted, the master transmits the address of the slave being addressed.

The slave device should send an acknowledge signal on the SDA line, once it recognizes its address.

The slave address is the first seven bits after a Start Condition. The direction of the data transfer (R/W) depends on the bit sent after the slave address — the eighth bit.

When the slave address is sent, each device in the system compares this slave address with its own. If there is a match, the device considers itself addressed and sends an acknowledge signal. Depending upon the state of the R/W bit (1:read, 0:write), the device acts as a transmitter or a receiver.

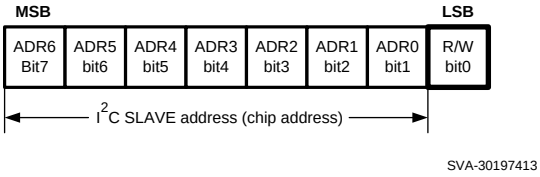


Figure 23. I^2C chip address

7.5.2.6 Control Register Write Cycle

- Master device generates start condition.
- Master device sends slave address (7 bits) and the data direction bit (r/w = 0).
- Slave device sends acknowledge signal if the slave address is correct.
- Master sends control register address (8 bits).
- Slave sends acknowledge signal.
- Master sends data byte to be written to the addressed register.

- Slave sends acknowledge signal.
- If master will send further data bytes the control register address will be incremented by one after acknowledge signal.
- Write cycle ends when the master creates stop condition.

7.5.2.7 Control Register Read Cycle

- Master device generates a start condition.
- Master device sends slave address (7 bits) and the data direction bit (r/w = 0).
- Slave device sends acknowledge signal if the slave address is correct.
- Master sends control register address (8 bits).
- Slave sends acknowledge signal.
- Master device generates repeated start condition.
- Master sends the slave address (7 bits) and the data direction bit (r/w = 1).
- Slave sends acknowledge signal if the slave address is correct.
- Slave sends data byte from addressed register.
- If the master device sends acknowledge signal, the control register address will be incremented by one. Slave device sends data byte from addressed register.
- Read cycle ends when the master does not generate acknowledge signal after data byte and generates stop condition.

Table 25. I²C Data Read/Write Flow(1)

	ADDRESS MODE
Data Read	<Start Condition>
	<Slave Address><r/w = 0>[Ack]
	<Register Addr.>[Ack]
	<Repeated Start Condition>
	<Slave Address><r/w = 1>[Ack]
	[Register Data]<Ack or NAck>
	... additional reads from subsequent register address possible
	<Stop Condition>
Data Write	<Start Condition>
	<Slave Address><r/w='0'>[Ack]
	<Register Addr.>[Ack]
	<Register Data>[Ack]
	... additional writes to subsequent register address possible
	<Stop Condition>

(1) <>Data from master [] Data from slave

7.5.2.8 Register Read/Write Format

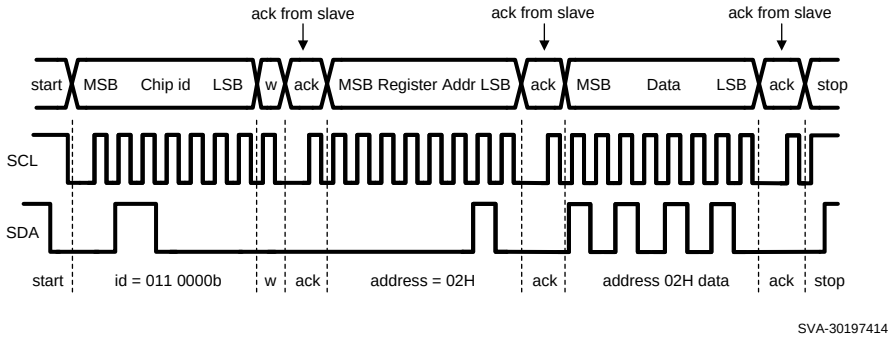


Figure 24. Register Write Format

When a *read* function is to be accomplished, a *write* function must precede the *read* function, as show in the Read Cycle waveform.

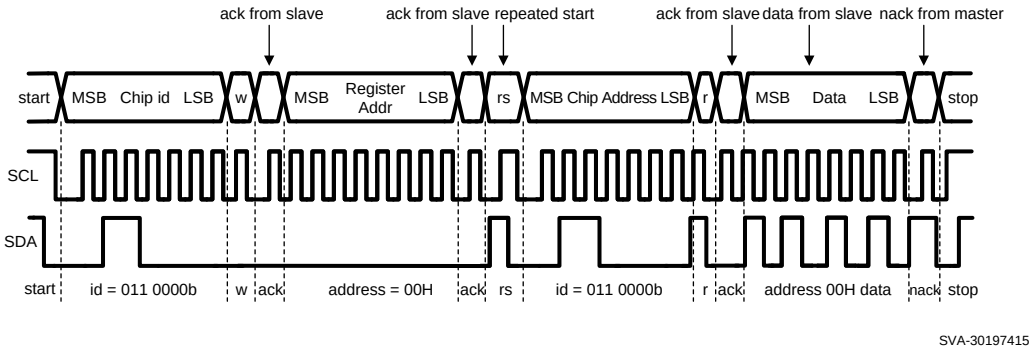


Figure 25. Register Read Format

w = write (SDA = 0)

r = read (SDA = 1)

ack = acknowledge (SDA pulled down by either master or slave)

rs = repeated start

id = 7-bit chip address

7.6 Register Maps

Table 26. LP5562 Control Register Names and Default Values

ADDR (HEX)	REGISTER	D7	D6	D5	D4	D3	D2	D1	D0	DEFAULT
00	ENABLE	LOG_EN	CHIP_EN	ENG1_EXEC[1:0]		ENG2_EXEC[1:0]		ENG3_EXEC[1:0]		0000 0000
01	OP MODE			ENG1_MODE[1:0]		ENG2_MODE[1:0]		ENG3_MODE[1:0]		0000 0000
02	B PWM	B_PWM[7:0]								0000 0000
03	G PWM	G_PWM[7:0]								0000 0000
04	R PWM	R_PWM[7:0]								0000 0000
05	B CURRENT	B_CURRENT[7:0]								1010 1111
06	G CURRENT	G_CURRENT[7:0]								1010 1111
07	R CURRENT	R_CURRENT[7:0]								1010 1111
08	CONFIG		PWM_HF	PS_EN				CLK_DET_EN	INT_CLK_E N	0000 0000
09	ENG1 PC					ENG1_PC[3:0]				0000 0000
0A	ENG2 PC					ENG2_PC[3:0]				0000 0000
0B	ENG3 PC					ENG3_PC[3:0]				0000 0000
0C	STATUS					EXT_CLK_US ED	ENG1_INT	ENG2_INT	ENG3_INT	0000 0000
0D	RESET	RESET[7:0]								0000 0000
0E	W PWM	W_PWM[7:0]								00000000
0F	W CURRENT	W_CURRENT[7:0]								10101111
70	LED MAP	W_ENG_SEL		R_ENG_SEL		G_ENG_SEL		B_ENG_SEL		00111001
10	PROG MEM ENG1	CMD_1_ENG1[15:8]								0000 0000
11	PROG MEM ENG1	CMD_1_ENG1[7:0]								0000 0000
...										
2E	PROG MEM ENG1	CMD_16_ENG1[15:8]								0000 0000
2F	PROG MEM ENG1	CMD_16_ENG1[7:0]								0000 0000
30	PROG MEM ENG2	CMD_1_ENG2[15:8]								0000 0000
31	PROG MEM ENG2	CMD_1_ENG2[7:0]								0000 0000
...										
4E	PROG MEM ENG2	CMD_16_ENG2[15:8]								0000 0000
4F	PROG MEM ENG2	CMD_16_ENG2[7:0]								0000 0000
50	PROG MEM ENG3	CMD_1_ENG3[15:8]								0000 0000
51	PROG MEM ENG3	CMD_1_ENG3[7:0]								0000 0000
...										
6E	PROG MEM ENG3	CMD_16_ENG3[15:8]								0000 0000
6F	PROG MEM ENG3	CMD_16_ENG3[7:0]								0000 0000

7.6.1 Enable Register (Enable) (Address = 00h) [reset = 00h]

EXEC registers are synchronized to the 32-kHz clock. Delay between consecutive I^2C writes to ENABLE register (00h) need to be longer than 488 μs (typical).

Figure 26. Enable Register

7	6	5	4	3	2	1	0
LOG_EN	CHIP_EN	ENG1_EXEC[1:0]		ENG2_EXEC[1:0]		ENG3_EXEC[1:0]	
R/W	R/W	R/W		R/W		R/W	

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 27. Enable Register Field Descriptions

Bit	Field	Type	Reset	Description
7	LOG_EN	R/W		Logarithmic PWM adjustment generation enable
6	CHIP_EN	R/W		Master chip enable. Enables device internal startup sequence. See for further information. Setting EN pin low resets the CHIP_EN state to 0. Allow 500 μs delay after setting chip_en bit to '1'
5:4	ENG1_EXEC	R/W		Engine 1 program execution. **00b** = Hold: Wait until current command is finished then stop while EXEC mode is hold. PC can be read or written only in this mode. **01b** = Step: Execute instruction defined by current engine 1 PC value, increment PC and change ENG1_EXEC to 00b (Hold) **10b** = Run: Start at program counter value defined by current engine 1 PC value **11b** = Execute instruction defined by current engine 1 PC value and change ENG1_EXEC to 00b (Hold)
3:2	ENG2_EXEC	R/W		Engine 2 program execution **00b** = Hold: Wait until current command is finished then stop while EXEC mode is hold. PC can be read or written only in this mode. **01b** = Step: Execute instruction defined by current engine 2 PC value, increment PC and change ENG2_EXEC to 00b (Hold) **10b** = Run: Start at program counter value defined by current engine 2 PC value **11b** = Execute instruction defined by current engine 2 PC value and change ENG2_EXEC to 00b (Hold)
ENG3_EXEC	1:0	R/W		Engine 3 program execution **00b** = Hold: Wait until current command is finished then stop while EXEC mode is hold. PC can be read or written only in this mode. **01b** = Step: Execute instruction defined by current engine 3 PC value, increment PC and change ENG3_EXEC to 00b (Hold) **10b** = Run: Start at program counter value defined by current engine 3 PC value **11b** = Execute instruction defined by current engine 3 PC value and change ENG3_EXEC to 00b (Hold)

7.6.2 Operation Mode Register (OP Mode) (address = 01h) [reset = 00h]

MODE registers are synchronized to 32-kHz clock. Delay between consecutive I^2C writes to OP_MODE register (01h) need to be longer than 153 μs (typ).

Figure 27. OP Mode Register

7	6	5	4	3	2	1	0
		ENG1_MODE[1:0]		ENG1_MODE[1:0]		ENG1_MODE[1:0]	
R/W				R/W		R/W	

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 28. OP Mode Register Field Descriptions

Bit	Field	Type	Reset	Description
5:4	ENG1_MODE	R/W		Engine 1 operation mode 00b = Disabled 01b = Load program to SRAM, reset engine 1 PC 10b = Run program defined by ENG1_EXEC 11b = Direct control
3:2	ENG2_MODE	R/W		Engine 2 operation mode 00b = Disabled 01b = Load program to SRAM, reset engine 2 PC 10b = Run program defined by ENG2_EXEC 11b = Direct control
1:0	ENG3_MODE	R/W		Engine 3 operation mode 00b = Disabled 01b = Load program to SRAM, reset engine 3 PC 10b = Run program defined by ENG3_EXEC 11b = Direct control

7.6.3 B LED Output PWM Control Register (B_PWM) (address = 02h) [reset = 00h]

Figure 28. B_PWM Control Register

7	6	5	4	3	2	1	0
B_PWM[7:0]							
R/W							

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 29. B_PWM Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	B_PWM	R/W		B LED output PWM value during direct control operation mode

7.6.4 G LED Output PWM Control Register (G_PWM) (address = 03h) [reset = 00h]

Figure 29. G_PWM Control Register

7	6	5	4	3	2	1	0
G_PWM[7:0]							
R/W							

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 30. G_PWM Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	G_PWM	R/W		G LED output PWM value during direct control operation mode

7.6.5 R LED Output PWM Control Register (R_PWM) (address = 04h) [reset = 00h]

Figure 30. R_PWM Register

7	6	5	4	3	2	1	0
R_PWM[7:0]							
R/W							

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 31. R_PWM Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	R_PWM	R/W		R LED output PWM value during direct control operation mode

7.6.6 B LED Output Current Control Register (B_CURRENT)(address = 05h) [reset = AFh]

Figure 31. B_CURRENT Control Register

7	6	5	4	3	2	1	0
B_CURRENT[7:0]							
R/W							

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 32. B_CURRENT Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	B_CURRENT	R/W		Current setting 0000 0000b = 0.0 mA 0000 0001b = 0.1 mA 0000 0010b = 0.2 mA 0000 0011b = 0.3 mA 0000 0100b = 0.4 mA 0000 0101b = 0.5 mA 0000 0110b = 0.6 mA ... **1010 1111b = 17.5 mA (default)** ... 1111 1011b = 25.1 mA 1111 1100b = 25.2 mA 1111 1101b = 25.3 mA 1111 1110b = 25.4 mA 1111 1111b = 25.5 mA

7.6.7 G LED Output Current Control Register (G_CURRENT)(address = 06h) [reset = AFh]

Figure 32. G_CURRENT Control Register

7	6	5	4	3	2	1	0
G_CURRENT[7:0]							
R/W							

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 33. G_CURRENT Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	G_CURRENT	R/W		Current setting 0000 0000b = 0.0 mA 0000 0001b = 0.1 mA 0000 0010b = 0.2 mA 0000 0011b = 0.3 mA 0000 0100b = 0.4 mA 0000 0101b = 0.5 mA 0000 0110b = 0.6 mA ... **1010 1111b = 17.5 mA (default)** ... 1111 1011b = 25.1 mA 1111 1100b = 25.2 mA 1111 1101b = 25.3 mA 1111 1110b = 25.4 mA 1111 1111b = 25.5 mA

7.6.8 R LED Output Current Control Register (R_CURRENT) (address = 07h) [reset = AFh]

Figure 33. R_CURRENT Control Register

7	6	5	4	3	2	1	0
R_CURRENT[7:0]							
R/W							

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 34. R_CURRENT Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	R_CURRENT	R/W		Current setting 0000 0000b = 0.0 mA 0000 0001b = 0.1 mA 0000 0010b = 0.2 mA 0000 0011b = 0.3 mA 0000 0100b = 0.4 mA 0000 0101b = 0.5 mA 0000 0110b = 0.6 mA ... **1010 1111b = 17.5 mA (default)** ... 1111 1011b = 25.1 mA 1111 1100b = 25.2 mA 1111 1101b = 25.3 mA 1111 1110b = 25.4 mA 1111 1111b = 25.5 mA

7.6.9 Configuration Control Register (CONFIG) (address = 08h) [reset = 00h]

Figure 34. CONFIG Register

7	6	5	4	3	2	1	0
	PWM_HF	PS_EN				CLK_DET_EN	INT_CLK_EN
	R/W	R/W				R/W	R/W

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 35. CONFIG Register Field Descriptions

Bit	Field	Type	Reset	Description
6	PWM_HF	R/W		PWM clock 0 = 256-Hz PWM clock used 1 = 558-Hz PWM clock used
5	PWRSAVE_EN	R/W		Power save mode enable
1:0	CLK_DET_EN, INT_CLK_EN	R/W		LED Controller clock source 00b = External clock source (CLK_32K) 01b = Internal clock 10b = Automatic selection 11b = Internal clock

7.6.10 Engine 1 Program Counter Value Register (Engine 1 PC) (address = 09h) [reset = 00h]

PC registers are synchronized to a 32-kHz clock. Delay between consecutive I^2C writes to PC registers needs to be longer than 153 μs (typical). PC register can be read or written only when EXEC mode is *hold*.

Figure 35. Engine 1 PC Value Register

7	6	5	4	3	2	1	0
				ENG1_PC[3:0]			
				R/W			

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 36. Engine 1 PC Register Field Descriptions

Bit	Field	Type	Reset	Description
3:0	ENG1_PC	R/W		Engine 1 program counter value

7.6.11 Engine 2 Program Counter Value Register (Engine 2 PC) (address = 0Ah) [reset = 00h]

PC registers are synchronized to 32-kHz clock. Delay between consecutive I^2C writes to PC registers needs to be longer than 153 μs (typical). PC register can be read or written only when EXEC mode is *hold*.

Figure 36. Engine 2 PC Value Register

7	6	5	4	3	2	1	0
				ENG2_PC[3:0]			
				R/W			

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 37. Engine 2 PC Register Field Descriptions

Bit	Field	Type	Reset	Description
3:0	ENG2_PC	R/W		Engine 2 program counter value

7.6.12 Engine 3 Program Counter Value Register (Engine 3 PC) (address = 0Ah) [reset = 00h]

PC registers are synchronized to 32 kHz clock. Delay between consecutive I^2C writes to PC registers needs to be longer than 153 μs (typ.). PC register can be read or written only when EXEC mode is *hold*.

Figure 37. Engine 3 PC Value Register

7	6	5	4	3	2	1	0
				ENG3_PC[3:0]			
				R/W			

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 38. Engine 3 PC Register Field Descriptions

Bit	Field	Type	Reset	Description
3:0	ENG3_PC	R/W		Engine 3 program counter value

7.6.13 STATUS/INTERRUPT Register (address = 0Ch) [reset = 00h]

Note: Register INT bits will be cleared when read operation to Status/Interrupt register occurs.

Figure 38. STATUS/INTERRUPT Register

7	6	5	4	3	2	1	0
				EXT_CLK USED	ENG1_INT	ENG2_INT	ENG3_INT
				R	R	R	R

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 39. STATUS/INTERRUPT Register Field Descriptions

Bit	Field	Type	Reset	Description
3	EXT_CLK USED	R		External clock state 0 = Internal clock used 1 = External 32kHz clock used
2	ENG1_INT	R		Interrupt from engine 1
1	ENG2_INT	R		Interrupt from engine 2
0	ENG3_INT	R		Interrupt from engine 3

7.6.14 RESET Register (address = 0Dh) [reset = 00h]

Figure 39. RESET Register

7	6	5	4	3	2	1	0
RESET[7:0]							
R/W							

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 40. RESET Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	RESET	R/W		Reset all register values when FFh is written.

7.6.15 WLED Output PWM Control Register (W_PWM) (address = 0Eh) [reset = 00h]

Figure 40. W_PWM Control Register

7	6	5	4	3	2	1	0
W_PWM[7:0]							
R/W							

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 41. W_PWM Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	W_PWM	R/W		W LED Output PWM value during direct control operation mode

7.6.16 W LED Output Current Control Register (W_CURRENT) (address = 0Fh) [reset = AFh]

Figure 41. W_CURRENT Control Register

7	6	5	4	3	2	1	0
W_CURRENT[7:0]							
R/W							

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 42. W_CURRENT Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	W_CURRENT	R/W		Current setting 0000 0000b = 0.0 mA 0000 0001b = 0.1 mA 0000 0010b = 0.2 mA 0000 0011b = 0.3 mA 0000 0100b = 0.4 mA 0000 0101b = 0.5 mA 0000 0110b = 0.6 mA ... **1010 1111b = 17.5 mA (default)** ... 1111 1011b = 25.1 mA 1111 1100b = 25.2 mA 1111 1101b = 25.3 mA 1111 1110b = 25.4 mA 1111 1111b = 25.5 mA

7.6.17 LED Mapping Register (LED Map) (address = 70h) [reset = 39h]

Figure 42. LED Map Register

7	6	5	4	3	2	1	0
W_ENG_SEL[1:0]		R_ENG_SEL[1:0]		G_ENG_SEL[1:0]		B_ENG_SEL[1:0]	
R/W		R/W		R/W		R/W	

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 43. LED Map Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	W_ENG_SEL	R/W		Selection from where W LED output PWM is controlled, 00 = I^2C register 0Eh, 01 = Engine 1, 10 = Engine 2, 11 = Engine 3
5:4	R_ENG_SEL	R/W		Selection from where R LED output PWM is controlled 00 = I^2C register 04h, 01 = Engine 1, 10 = Engine 2, 11 = Engine 3
3:2	G_ENG_SEL	R/W		Selection from where G LED output PWM is controlled 00 = I^2C register 03h, 01 = Engine 1, 10 = Engine 2, 11 = Engine 3
1:0	B_ENG_SEL	R/W		Selection from where B LED output PWM is controlled 00 = I^2C register 02h, 01 = Engine 1, 10 = Engine 2, 11 = Engine 3

7.6.18 Program Memory (address = 10h – 6Fh) [reset = 00h]

See chapter *SRAM Memory* for further information.

Figure 43. Program Execution Engine Commands

Command	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RampWait	0	Pre-scale	Step time						Sign	Increment						
Set PWM	0	1	0						PWM Value							
Go toStart	0	0	0						0	0	0	0	0	0	0	0
Branch	1	0	1	Loop Count						X	Step number					
End	1	1	0	Int	Reset	X										
Trigger	1	1	1	Wait for trigger on channels 5-0						Send trigger on channels 5-0						X

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

8 Application and Implementation

NOTE

Information in the following applications sections is not part of the TI component specification, and TI does not warrant its accuracy or completeness. TI's customers are responsible for determining suitability of components for their purposes. Customers should validate and test their design implementation to confirm system functionality.

8.1 Application Information

The LP5562 is designed for mobile applications with an input voltage V_{IN} between 2.7 V to 5.5 V to support a 1S lithium-ion battery source. A microcontroller or other I^2C host is required to initialize and configure the LP5562 after power-up. An internal of external 32-kHz clock is required. If multiple LP5562 devices are used to sequence multiple RGB LEDs, then the external 32-kHz clock input is required. The ADDRSEL0 and ADDRSEL1 pins can be used to allow unique sequencing of up to four LP5562 devices on the same I^2C bus.

The four LED current drivers can be configured up to 25.5-mA LED current each and are tolerant up to 6-V LED supply voltage.

8.2 Typical Application

Figure 44 shows the typical application for LP5562 supporting one RGB LED and one White LED with the four LED current outputs.

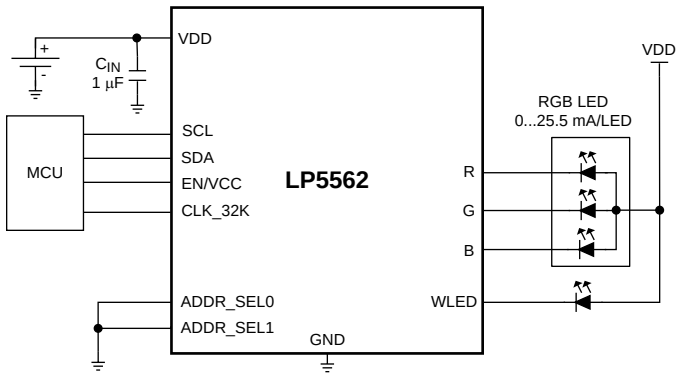


Figure 44. LP5562 Typical Application

8.2.1 Design Requirements

For typical LED-driver applications, use the parameters listed in Table 44.

Table 44. Design Parameters

DESIGN PARAMETER	EXAMPLE VALUE
Minimum input voltage	2.7 V
Maximum input voltage	5.5 V
EN/VCC logic level	1.8 V
R output current	24 mA
G output current	20 mA
B output current	22 mA
WLED output current	18 mA
PWM frequency	256 Hz
32-kHz clock source	External
Light engines 1, 2 and 3	Blink on/off
Power-save mode	Enabled

8.2.2 Detailed Design Procedure

Table 45. Recommended External Components

MODEL	TYPE	VENDOR	VOLTAGE RATING	SIZE INCH (MM)
1 µF for C_{IN}				
C1005X5R1A105K	Ceramic X5R	TDK	10 V	0402 (1005)
GRM155R61A105KE15D	Ceramic X5R	Murata	10 V	0402 (1005)
LEDs	User Defined			

8.2.2.1 Output Current Configuration

- For 22-mA “B” output current the register address 0x05 is set to 0xDC. B_CURRENT[7:0] = (22 mA / 0.1) = 220 decimal.
- For 22-mA “G” output current the register address 0x06 is set to 0xC8. G_CURRENT[7:0] = (20 mA / 0.1) = 200 decimal.
- For 24-mA “R” output current the register address 0x07 is set to 0xF0. R_CURRENT[7:0] = (24 mA / 0.1) = 240 decimal.
- For 18-mA “WLED” output current the register address 0x0F is set to 0xB4. W_CURRENT[7:0] = (18 mA / 0.1) = 180 decimal.

8.2.2.2 PWM Frequency Configuration

To set the PWM frequency to 256 Hz the PWM_HF bit must be written to '0'. It is located at register address 0x08 bit 6.

8.2.2.3 Clock Source Configuration

To set the clock source to external clock input the INT_CLK_EN and CLK_DET_EN bits are set to “00”. These are located at register address 0x08 bits 0 and 1.

8.2.2.4 Power-Save Mode Configuration

To enable the power save mode function the PWRSAVE_EN bit must be set to '1'. It is located at register address 0x08 bit 5.

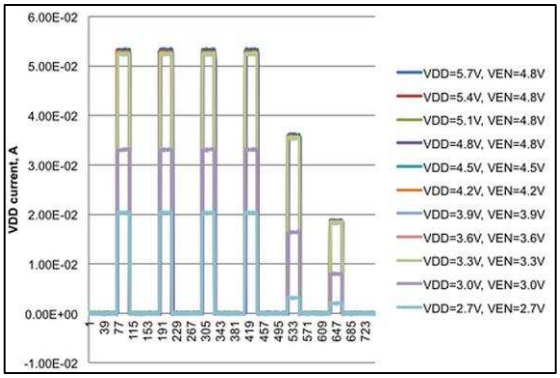
8.2.2.5 Light Engine Configuration

Sample code for 100% to 0% duty blinking LED function is shown in Table 46.

Table 46. LED Function Sample Codes

ADDRESS	VALUE	COMMENTS	
Engine 1 Section Start			
0	40FF	Loop1	set_pwm 255
1	4D00		wait 200
2	4000		set_pwm 0
3	6000		wait 500
4	A200		branch 4, loop1
5	D000		end, i
Engine 2 Section Start			
10	40FF	Loop2	set_pwm 255
11	4D00		wait 200
12	4000		set_pwm 0
13	6000		wait 500
14	A290		branch 5, loop2
15	D000		end, i
Engine 3 Section Start			
20	40FF	Loop3	set_pwm 255
21	4D00		wait 200
22	4000		set_pwm 0
23	6000		wait 500
24	A320		branch 6, loop3
25	D000		end, i

8.2.3 Application Curve



External Clock PS. Blink-Program

Figure 45. LED Driver Current Accuracy vs RGBW Current

9 Power Supply Recommendation

The LP5562 is intended to operate from a single cell lithium-ion battery or battery charger up to 5.5 V. The resistance of the input supply rail should be low enough so that the input current transient does not cause too high drop at the LP5562 VDD pin. If the input supply is also powering the LEDs and is connected by using long wires additional bulk capacitance may be required in addition to the ceramic bypass capacitors in the VDD / LED supply line.

10 Layout

10.1 Layout Guidelines

Figure 46 is a layout recommendation for the LP5562 used to demonstrate the principles of good layout. This example is for a 2-layer PCB and a single LP5562 device. This layout can be adapted to the actual application layout if/where possible. If multiple LP5562 devices are used in the application then not all the ADDRSEL pins will be tied to ground. In that case a PCB using HDI process is needed to route ADDRSEL0 and GND bumps. The VDD decoupling cap must be placed close to VDD bump, and the traces for W, R, G, and B bumps must be sized to carry 25.5 mA.

10.2 Layout Example

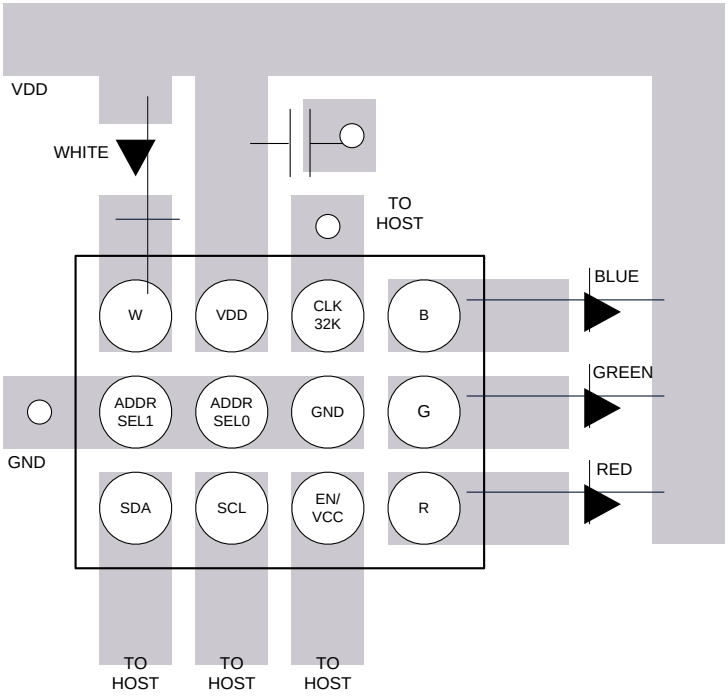


Figure 46. LP5562 Layout Example

11 Device and Documentation Support

11.1 Device Support

11.1.1 Third-Party Products Disclaimer

TI'S PUBLICATION OF INFORMATION REGARDING THIRD-PARTY PRODUCTS OR SERVICES DOES NOT CONSTITUTE AN ENDORSEMENT REGARDING THE SUITABILITY OF SUCH PRODUCTS OR SERVICES OR A WARRANTY, REPRESENTATION OR ENDORSEMENT OF SUCH PRODUCTS OR SERVICES, EITHER ALONE OR IN COMBINATION WITH ANY TI PRODUCT OR SERVICE.

11.2 Documentation Support

11.2.1 Related Documentation

For additional information, see the following:

AN-1112 DSBGA Wafer Level Chip Scale Package.

11.3 Community Resources

The following links connect to TI community resources. Linked contents are provided "AS IS" by the respective contributors. They do not constitute TI specifications and do not necessarily reflect TI's views; see TI's Terms of Use.

TI E2E™ Online Community ***TI's Engineer-to-Engineer (E2E) Community.*** Created to foster collaboration among engineers. At e2e.ti.com, you can ask questions, share knowledge, explore ideas and help solve problems with fellow engineers.

Design Support ***TI's Design Support*** Quickly find helpful E2E forums along with design support tools and contact information for technical support.

11.4 Trademarks

E2E is a trademark of Texas Instruments.
All other trademarks are the property of their respective owners.

11.5 Electrostatic Discharge Caution

This integrated circuit can be damaged by ESD. Texas Instruments recommends that all integrated circuits be handled with appropriate precautions. Failure to observe proper handling and installation procedures can cause damage.

ESD damage can range from subtle performance degradation to complete device failure. Precision integrated circuits may be more susceptible to damage because very small parametric changes could cause the device not to meet its published specifications.

11.6 Glossary

SLYZ022 — *TI Glossary*.

This glossary lists and explains terms, acronyms, and definitions.

12 Mechanical, Packaging, and Orderable Information

The following pages include mechanical, packaging, and orderable information. This information is the most current data available for the designated devices. This data is subject to change without notice and revision of this document. For browser-based versions of this data sheet, refer to the left-hand navigation.

PACKAGING INFORMATION

Orderable Device	Status (1)	Package Type	Package Drawing	Pins	Package Qty	Eco Plan (2)	Lead finish/ Ball material (6)	MSL Peak Temp (3)	Op Temp (°C)	Device Marking (4/5)	Samples
LP5562TME/NOPB	ACTIVE	DSBGA	YQE	12	250	RoHS & Green	SNAGCU	Level-1-260C-UNLIM	-40 to 85	D5	Samples
LP5562TMX/NOPB	ACTIVE	DSBGA	YQE	12	3000	RoHS & Green	SNAGCU	Level-1-260C-UNLIM	-40 to 85	D5	Samples

(1) The marketing status values are defined as follows:
ACTIVE: Product device recommended for new designs.
LIFEBUY: TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.
NRND: Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.
PREVIEW: Device has been announced but is not in production. Samples may or may not be available.
OBSOLETE: TI has discontinued the production of the device.

(2) **RoHS:** TI defines "RoHS" to mean semiconductor products that are compliant with the current EU RoHS requirements for all 10 RoHS substances, including the requirement that RoHS substance do not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, "RoHS" products are suitable for use in specified lead-free processes. TI may reference these types of products as "Pb-Free".
RoHS Exempt: TI defines "RoHS Exempt" to mean products that contain lead but are compliant with EU RoHS pursuant to a specific EU RoHS exemption.
Green: TI defines "Green" to mean the content of Chlorine (Cl) and Bromine (Br) based flame retardants meet JS709B low halogen requirements of <=1000ppm threshold. Antimony trioxide based flame retardants must also meet the <=1000ppm threshold requirement.

(3) MSL, Peak Temp. - The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

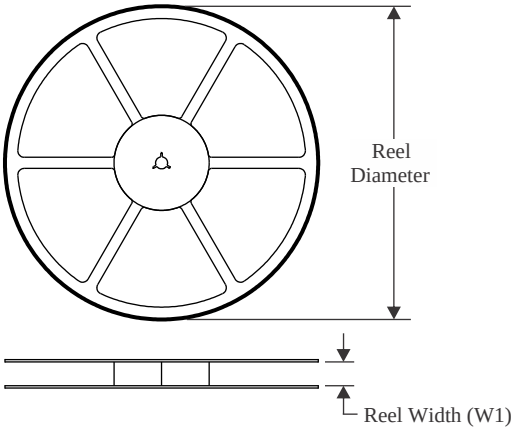
(4) There may be additional marking, which relates to the logo, the lot trace code information, or the environmental category on the device.

(5) Multiple Device Markings will be inside parentheses. Only one Device Marking contained in parentheses and separated by a "~" will appear on a device. If a line is indented then it is a continuation of the previous line and the two combined represent the entire Device Marking for that device.

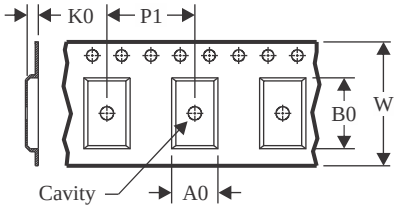
(6) Lead finish/Ball material - Orderable Devices may have multiple material finish options. Finish options are separated by a vertical ruled line. Lead finish/Ball material values may wrap to two lines if the finish value exceeds the maximum column width.

TAPE AND REEL INFORMATION

REEL DIMENSIONS

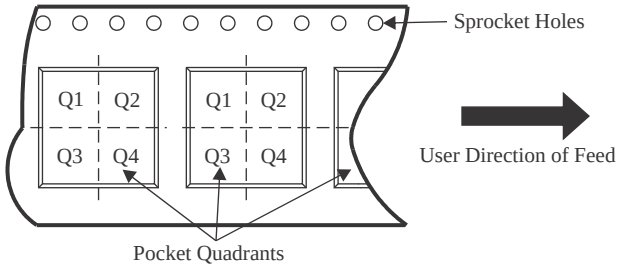


TAPE DIMENSIONS



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
P1	Pitch between successive cavity centers

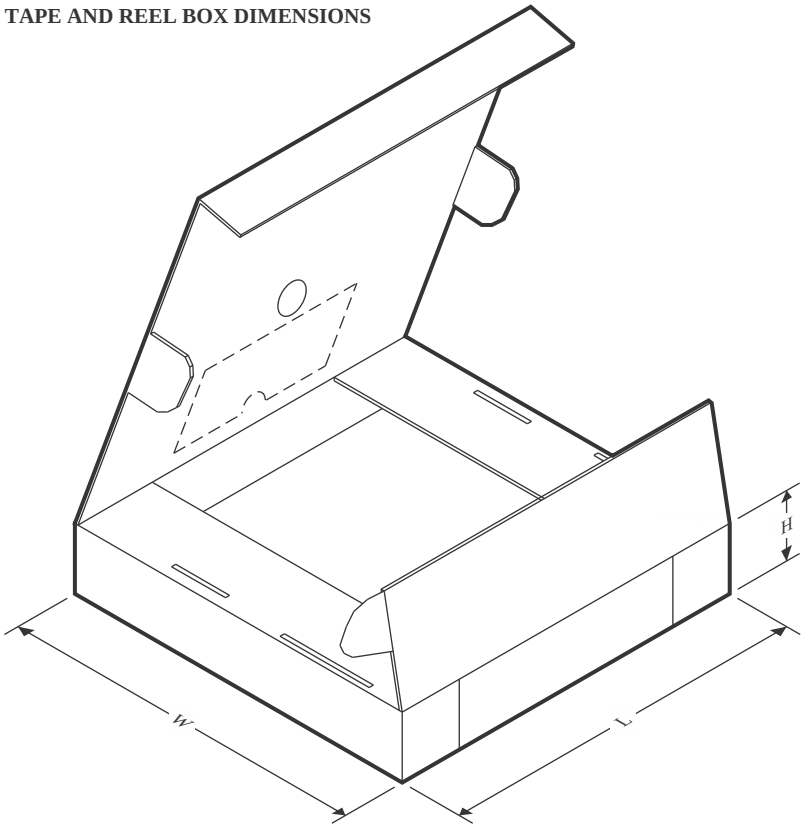
QUADRANT ASSIGNMENTS FOR PIN 1 ORIENTATION IN TAPE



*All dimensions are nominal

Device	Package Type	Package Drawing	Pins	SPQ	Reel Diameter (mm)	Reel Width W1 (mm)	A0 (mm)	B0 (mm)	K0 (mm)	P1 (mm)	W (mm)	Pin1 Quadrant
LP5562TME/NOPB	DSBGA	YQE	12	250	178.0	8.4	1.35	1.75	0.76	4.0	8.0	Q2
LP5562TMX/NOPB	DSBGA	YQE	12	3000	178.0	8.4	1.35	1.75	0.76	4.0	8.0	Q2

TAPE AND REEL BOX DIMENSIONS



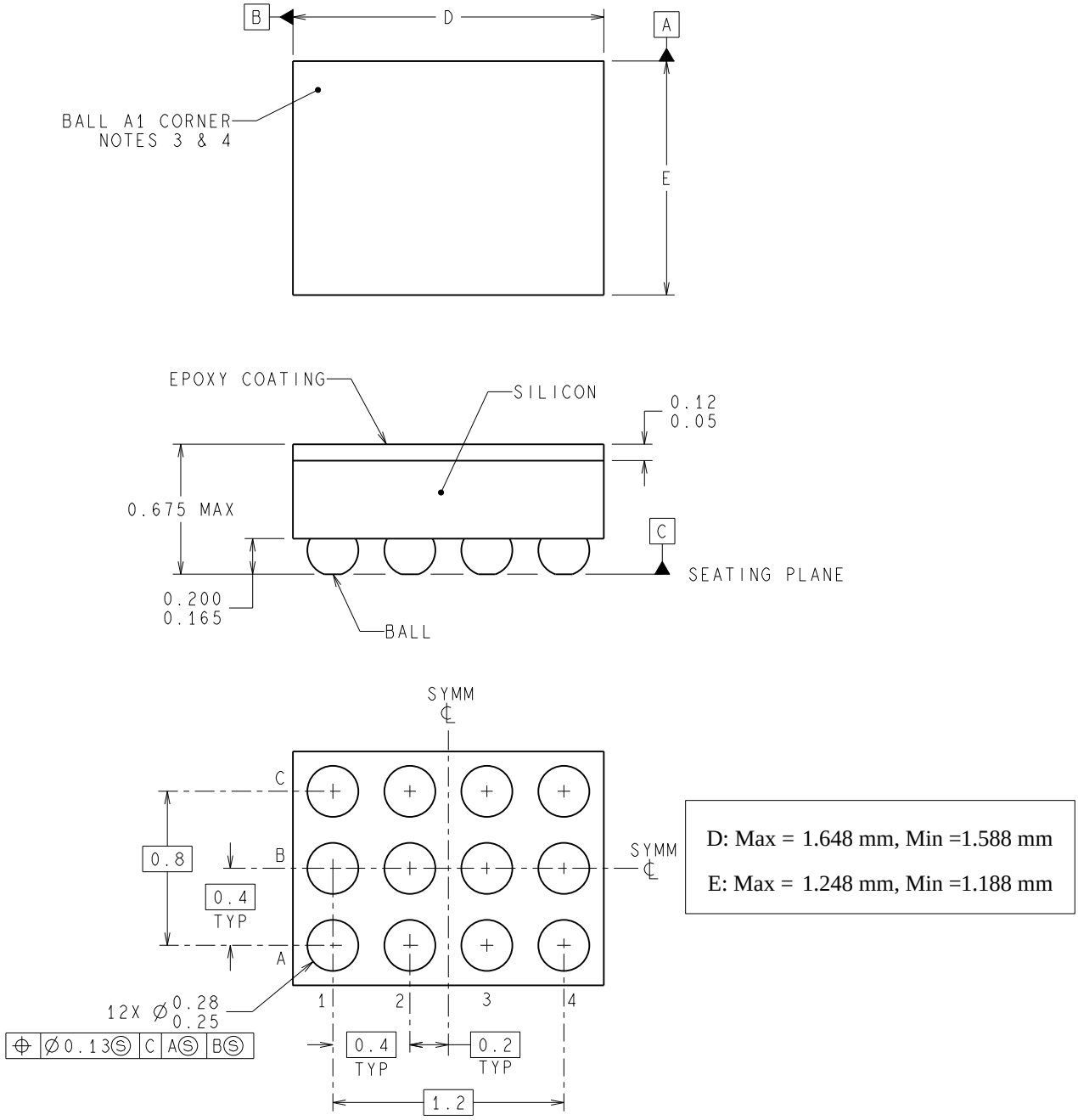
*All dimensions are nominal

Device	Package Type	Package Drawing	Pins	SPQ	Length (mm)	Width (mm)	Height (mm)
LP5562TME/NOPB	DSBGA	YQE	12	250	208.0	191.0	35.0
LP5562TMX/NOPB	DSBGA	YQE	12	3000	208.0	191.0	35.0

YQE0012

DSBGA - 0.675mm MAX HEIGHT

DIE SIZE BALL GRID ARRAY



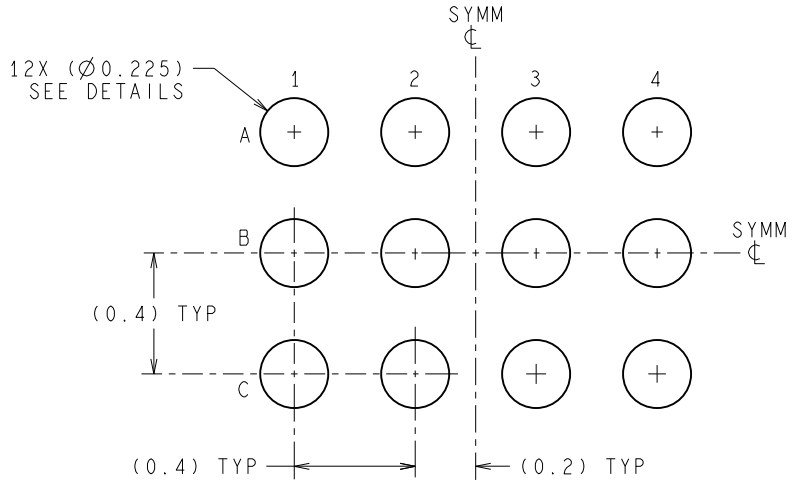
NOTES:

1. DIMENSIONS ARE IN MILLIMETERS. DIMENSIONS IN PARENTHESIS ARE FOR REFERENCE ONLY.
2. THIS DRAWING IS SUBJECT TO CHANGE WITHOUT NOTICE.
3. BALL A1 IS ESTABLISHED BY UPPER LEFT CORNER WITH RESPECT TO TEXT ORIENTATION.
4. COLUMNS NOT ORIENTED PER STANDARD CONFIGURATION.

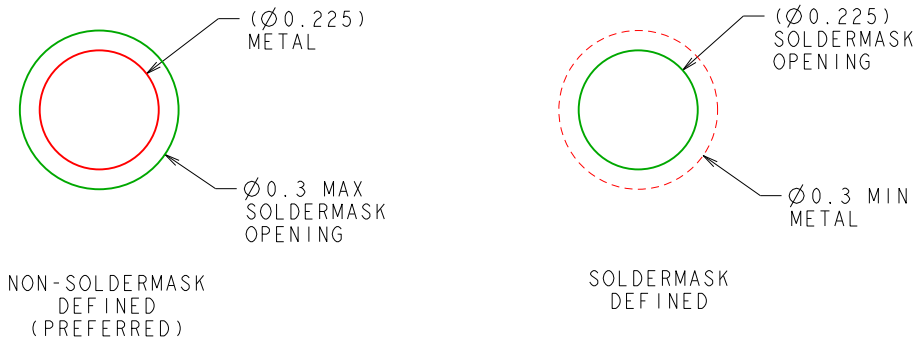
YQE0012

DSBGA - 0.675mm MAX HEIGHT

DIE SIZE BALL GRID ARRAY



LAND PATTERN EXAMPLE
SCALE:40X



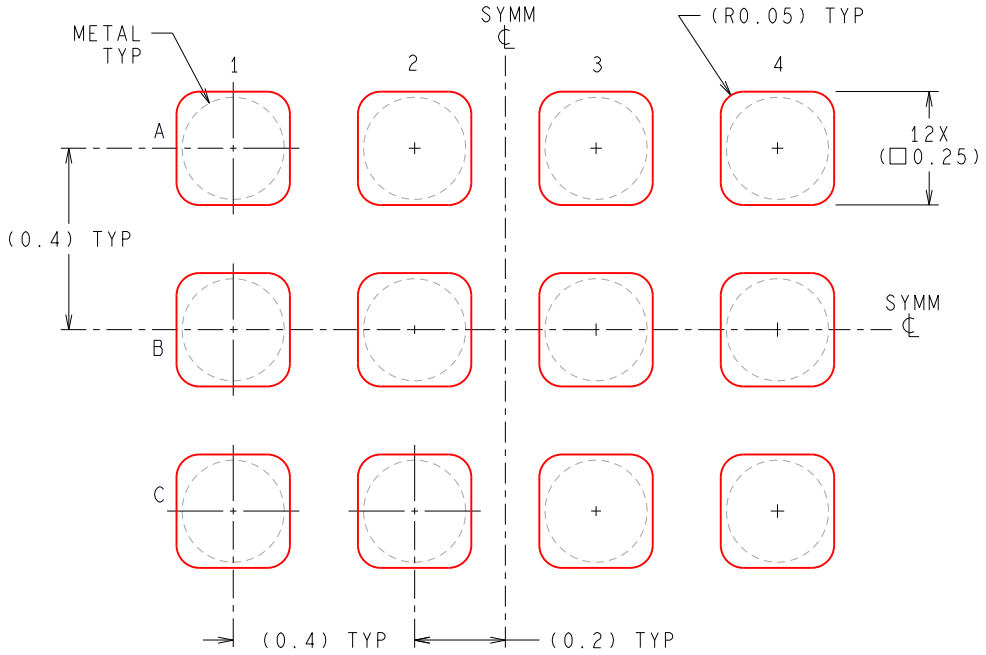
SOLDERMASK DETAILS
NOT TO SCALE

4218314/A 02/2013

YQE0012

DSBGA - 0.675mm MAX HEIGHT

DIE SIZE BALL GRID ARRAY



SOLDERPASTE EXAMPLE
BASED ON 0.1mm THICK STENCIL
SCALE:60X

4218314/A 02/2013

NOTES: (CONTINUED)

5. LASER CUT APERTURES WITH TRAPEZOIDAL WALLS AND ROUNDED CORNERS WILL OFFER BETTER PASTE RELEASE. REFER TO IPC-7525 FOR DESIGN CONSIDERATIONS.
6. NO SHARP EDGES ON OPENINGS - ELECTROPOLISH OPTIONAL TO ACCOMPLISH THIS.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265