Click here to ask an associate for production status of specific part numbers.

12-Switch Matrix Manager for Automotive Lighting

MAX20092

General Description

The MAX20092 12-switch matrix manager IC for automotive lighting applications includes a 12-switch array for bypassing individual LEDs in a single- or dual-string application. It features 12 individually controlled n-channel MOSFET switches rated for 10V with on-resistance of 0.100Ω. A single current source can be used to power all the LEDs connected in series. Individual LEDs can be dimmed by turning on and off the bypass switches across each LED, and can also be configured in 2 strings with 6 switches in series per string and 4 strings with 3 switches per string. A separate current source powers each string. Each switch can be connected across 1 or 2 LEDs in series. The IC also includes an internal charge pump that provides power for the gate drive of each of the LED bypass switches. The low on-resistance of the switches minimizes conduction loss and power dissipation.

The IC features a serial peripheral interface (SPI) for serial communication. The MAX20092 is a slave device that uses the SPI to communicate with an external microcontroller (µC), which is the master device. Each of the 12 switches can be independently programmed to bypass the LEDs across each of the switches in the string. Each switch can be turned fully on, fully off, or dimmed with or without fade-transition mode. The PWM frequency can be set by an internal oscillator or set to an external clock source. The IC features open-LED protection as well as open- and short-LED fault reporting through the SPI. The MAX20092 is available in a 32-pin (5mm x 5mm) side-wettable TQFN (SWTQFN) package with a thermally enhanced exposed pad.

Applications

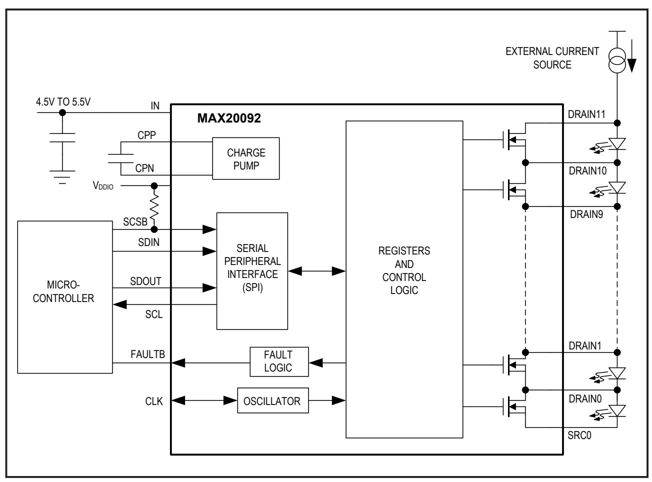
- Automotive Front-Light Systems
- Automotive Tail-Light Systems
- Automotive Matrix-Lighting Systems

Benefits and Features

- Automotive Ready: AEC-Q100 Qualified
- Flexible Configuration Allows the Use of the Same Device in Different Applications
    - Single-, Dual-, and Quad-String Configurations
    - Up to 12 Switches in Series in Single-String Configurations
    - Up to 6 Switches in Series in Dual-String Configurations
    - Up to 3 Switches in Series in Quad-String Configurations
    - Up to 2 LEDs per Switch
- Optimal PWM Dimming Arrangement Provides Excellent Dimming Performance
    - Programmable 12-Bit PWM Dimming
    - Fade Transition Between PWM Dimming States
    - Internal Clock Generator or External Clock for PWM Dimming
- Protection Features and Package Improve Reliability
    - Open-LED Protection
    - Programmable Open-LED and Shorted-LED Threshold
    - Open- and Shorted-LED Fault Reporting
    - Thermally Enhanced 32-Pin SWTQFN Package

Ordering Information appears at end of data sheet.

Simplified Block Diagram



19-100310; Rev 1; 11/19

Absolute Maximum Ratings

IN, V_{DDIO} to GND ....-0.3V to +6V
CPP to GND ....-0.3V to +70V
CPN to GND ....-0.3V to +65V
CPP to CPN ....-0.3V to +16V
RGRADE to GND ....-0.3V to V_{IN} + 0.3V
CPP to any DRAIN_ Pin.......................................-0.3V to +70V
DRAIN11, DRAIN8, DRAIN5, and DRAIN2 to GND....-0.3V to +56V
SRC0, SRC3, SRC6, SRC9 to GND....................-0.3V to +56V
DRAINx to DRAIN(x-1).......................................-0.3V to +14.0V
SDOUT, SCL, EN, CLK, ADDR2, ADDR1, ADDR0 SDIN,
SCSB, FAULTB to GND .... -0.3V to V_{DDIO} + 0.3V

Continuous Power Dissipation (Single-Layer Board)
(T_A = +70°C, derate 21.3mW/°C above +70°C).....1702.1mW
Continuous Power Dissipation (Multilayer Board)
(T_A = +70°C, derate 33.2mWºC above +70°C).........2658mW
Operating Temperature Range......................... -40°C to +125°C
Junction Temperature...+150°C
Storage Temperature Range .... -40°C to +150°C
Soldering Temperature (reflow).......................................+260°C

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Package Information

32-Pin (5mm x 5mm) SWTQFN

PACKAGE CODE	T3255Y+10C
Outline Number	21-100210
Land Pattern Number	90-100092
Thermal Resistance, Single-Layer Board:	
Junction to Ambient (θ_{JA})	47°C/W
Junction to Case (θ_{JC})	1.7°C/W
Thermal Resistance, Four-Layer Board:	
Junction to Ambient (θ_{JA})	30°C/W
Junction to Case (θ_{JC})	1.6°C/W

For the latest package outline information and land patterns (footprints), go to **www.maximintegrated.com/packages**. Note that a "+", "#", or "-" in the package code indicates RoHS status only. Package drawings may show a different suffix character, but the drawing pertains to the package regardless of RoHS status.

Package thermal resistances were obtained using the method described in JEDEC specification JESD51-7, using a four-layer board. For detailed information on package thermal considerations, refer to **www.maximintegrated.com/thermal-tutorial**.

Electrical Characteristics

(Input voltage and enable = V_{IN} = V_{EN} = V_{DDIO} = 5V, T_A = T_J = -40ºC to +125ºC, unless otherwise noted. Typical values are at T_A = +25ºC.) (Note 1)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
SUPPLY VOLTAGES						
Operating Supply Voltage Range	V_{IN}		4.5		+5.5	V
Input Operating Bias Current	I_{IN-Q}	No switching, SDIN, SDOUT, and SCL are idle and SCSB is high		2.3		mA
Input POR Threshold	V_{IN-POR}	V_{IN} rising	3.91	4.2	4.475	V
Input POR Hysteresis	POR_{VIN_HYST}			100		mV
Charge-Pump Operating Voltage	V_{CPP}				65	V
CLK INTERNAL OSCILLATOR OR EXTERNAL SYNC CLOCK SOURCE						
LED PWM-Dimming-Frequency Accuracy	f_{OSC}	Internal oscillator	-10		+10	%
Minimum LED PWM-Dimming Frequency		DIV[1:0] = 0x3, internal oscillator	225	250	275	Hz
CLK External Sync Input, Low Threshold	V_{IL_CLK}	2.2V < V_{DDIO} < 5.5V			0.3 x V_{DDIO}	V
		1.7V < V_{DDIO} < 2.2V			0.2 x V_{DDIO}	
CLK External Sync Input, High Threshold	V_{OH_CLK}	2.2V < V_{DDIO} < 5.5V	0.7 x V_{DDIO}			V
	V_{IH_CLK}	1.7V < V_{DDIO} < 2.2V	0.82 x V_{DDIO}			
CLK External Sync Input, Clock Frequency	f_{CLK}		0.30		10.0	MHz
CLK External Sync Input, Clock Pulse-Width High	t_{CLKH}		30			ns
CLK External Sync Input, Clock Pulse-Width Low	t_{CLKL}		30			ns
LED MATRIX SWITCHES						
Single-Switch On-Resistance	$R_{DS(ON)}$	(Note 2)		0.070		Ω
On-Resistance with Series Switches 11–9 On	$R_{DS11-9(ON)}$	All series switches from 11–9 are on (Note 2)		0.2	0.45	Ω
On-Resistance with Series Switches 8–6 On	$R_{DS8-6(ON)}$	Series switches from 8–6 are all on (Note 2)		0.2	0.45	Ω
On-Resistance with Series Switches 5–3 On	$R_{DS5-3(ON)}$	All series switches from 5–3 are on (Note 2)		0.2	0.45	Ω
On-Resistance with Series Switches 2–0 On	$R_{DS2-0(ON)}$	All series switches from 2–0 are on (Note 2)		0.2	0.45	Ω
Switch Leakage Current	$I_{DS(OFF)}$	V_{DS} = 10V			5	µA
Open-LED Threshold	V_{OTH}	Rising VDS	9.0	9.45	10.45	V

Electrical Characteristics (continued)

(Input voltage and enable = V_{IN} = V_{EN} = V_{DDIO} = 5V, T_A = T_J = -40ºC to +125ºC, unless otherwise noted. Typical values are at T_A = +25ºC.) (Note 1)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Programmable Shorted-LED Threshold	V_{STH}	V_{TH} = 0V, rising V_{DS}	0.80	1	1.2	V
		V_{TH} = 1, rising V_{DS}	4.3	4.7	5.1	
Open-Trace Detect Minimum LED Current	I_{LED_MIN}				230	mA
LED MATRIX SWITCHES—LED SWITCH TIMING						
LED Switch Open/Close Slew Accuracy	t_{SLEW_ERR}		-33		+33	%
LED Switch Open/Close Delay Variation	t_{SWDEL}	Variation of slew rate on any switch from the average value	-13.5		+13.5	%
LED Slew-Rate Setting 0	SR_LED_0	0 to 6V step, 10–90% rise/fall time, LED_SLEW[2:0] = 0x0		0.36		µs
LED Slew-Rate Setting 1	SR_LED_1	0 to 6V step, 10–90% rise/fall time, LED_SLEW[2:0] = 0x1		0.73		µs
LED Slew-Rate Setting 2	SR_LED_2	0 to 6V step, 10–90% rise/fall time, LED_SLEW[2:0] = 0x2		1.11		µs
LED Slew-Rate Setting 3	SR_LED_3	0 to 6V step, 10–90% rise/fall time, LED_SLEW[2:0] = 0x3		1.8		µs
LED Slew-Rate Setting 4	SR_LED_4	0 to 6V step, 10–90% rise/fall time, LED_SLEW[2:0] = 0x4		2.45		µs
LED Slew-Rate Setting 5	SR_LED_5	0 to 6V step, 10–90% rise/fall time, LED_SLEW[2:0] = 0x5		3.6		µs
LED Slew-Rate Setting 6	SR_LED_6	0 to 6V step, 10–90% rise/fall time, LED_SLEW[2:0] = 0x6		4.8		µs
LED Slew-Rate Setting 7	SR_LED_7	0 to 6V step, 10–90% rise/fall time, LED_SLEW[2:0] = 0x7		7.2		µs
SPI ELECTRICAL CHARACTERISTICS—POWER REQUIREMENTS						
I/O Supply Voltage	V_{DDIO}		1.7		5.5	V
VDDIO UVLO Rising Threshold	$UVLO_{VDDIO}$				1.65	V
VDDIO UVLO Hysteresis				50		mV
Static I/O Supply Current	I_{DDIO}	Static inputs, all outputs unloaded (Note 3)			10	µA
SPI ELECTRICAL CHARACTERISTICS—DIGITAL INPUT CHARACTERISTICS (ADDR0–ADDR2, SCLK, SDIN, SCSB, EN)						
Input High Voltage	V_{IH}	2.2V < V_{DDIO} < 5.5V	0.7 x V_{DDIO}			V
		1.7V < V_{DDIO} < 2.2V	0.82 x V_{DDIO}			
Input Low Voltage	V_{IL}	2.2V < V_{DDIO} < 5.5V			0.3 x V_{DDIO}	V
		1.7V < V_{DDIO} < 2.2V			0.2 x V_{DDIO}	
Input Leakage Current	I_{IN}	Input driven to 0V or V_{DDIO} (Note 4)			±1	µA

Electrical Characteristics (continued)

(Input voltage and enable = V_{IN} = V_{EN} = V_{DDIO} = 5V, T_A = T_J = -40ºC to +125ºC, unless otherwise noted. Typical values are at T_A = +25ºC.) (Note 1)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Internal Safety Impedance	R_{PD}	SDI, SCLK pulldown to GND (Note 5)	40	100	160	kΩ
	I_{RPU_SCSB}	SFT_SCSB pullup resistance to V_{DDIO}	40	100	160	
Input Capacitance	C_{IN}			10		pF
Hysteresis Voltage	V_H			0.15		V
SPI ELECTRICAL CHARACTERISTICS—DIGITAL OUTPUT CHARACTERISTICS (SDOUT, CLK)						
Output High Voltage	V_{OH}	V_{DDIO} > 2.5V, I_{SOURCE} = 5mA	V_{DDIO} - 0.4			V
		V_{DDIO} > 1.7V, I_{SOURCE} = 2mA	V_{DDIO} - 0.4			
Output Low Voltage	V_{OL}	V_{DDIO} > 2.5V, I_{SINK} = 5mA			0.4	V
		V_{DDIO} > 1.7V, I_{SINK} = 2mA			0.4	
Output Three-State Leakage	I_{OZ}	Output voltage between 0V and V_{DDIO}			±1	µA
Output Three-State Capacitance	C_{OZ}			10		pF
SPI TIMING CHARACTERISTICS						
SCLK Frequency	f_{SCLK}		128		4000	kHz
SPI SCSB Maximum Pulse Width	t_{MAX_SCSB}	The maximum duration of an SPI packet; longer than this and the part times out the SPI packet and declares an SPI_ERR; internal clock frequency depends on PWM_IN_SEL register settings			2048	Clock Cycles
SCLK Pulse Width High	t_{CH}		87.5			ns
SCLK Pulse Width Low	t_{CL}	(Note 6)	87.5			ns
SCSB Fall to SCLK Rise Setup Time	t_{CSS0}	To first SCLK rising edge (Note 6)	50			ns
SCSB Fall to SCLK Rise Hold Time	t_{CSH0}	Applies to inactive rising edge preceding the first rising edge	50			ns
SCSB Rise to SCLK Rise Hold Time	t_{CSH1}	Applies to 32nd rising edge	50			ns
SCSB Rise to SCLK Rise	t_{CSA}	Applies to 32nd rising edge, guarantees aborted (unqualified) sequence	25			ns
	t_{CSQ}	Applies to 33rd rising edge, guarantees qualified sequence	25			
SCSB Pulse Width High	t_{CSPW}		100			ns
SDI to SCLK Rise Setup Time	t_{DS}		20			ns
SDI to SCLK Rise Hold Time	t_{DH}		20			ns
SCLK Fall to SDO Transition	t_{DOT}	C_{LOAD} = 20pF			80	ns
SCLK Fall to SDO Hold	t_{DOH}	C_{LOAD} = 0pF	2			ns
SCSB Fall to SDO Transition	t_{DOE}	C_{LOAD} = 20pF			50	ns
SCSB Rise to SDO High Impedance	t_{DOZ}	Disable time			50	ns

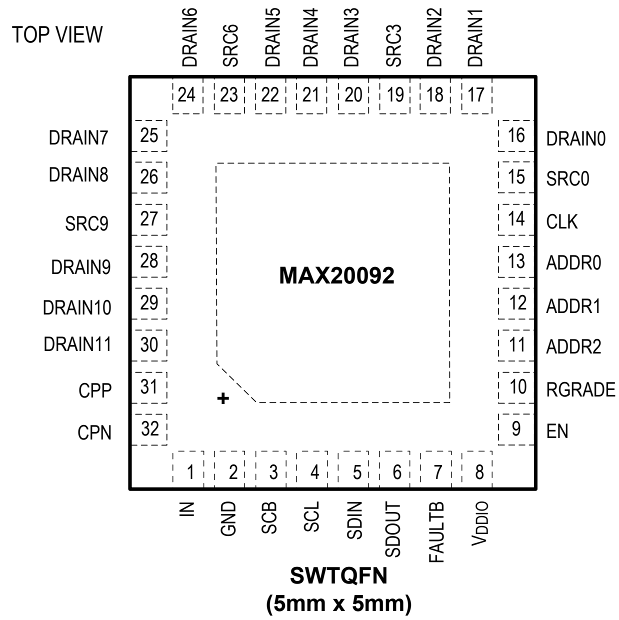
Electrical Characteristics (continued)

(Input voltage and enable = V_{IN} = V_{EN} = V_{DDIO} = 5V, T_A = T_J = -40ºC to +125ºC, unless otherwise noted. Typical values are at T_A = +25ºC.) (Note 1)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
FAULT STATUS OUTPUT						
FAULTB Output Low Voltage	V_{FAULTB}	I_{FAULTB} = 3mA		0.04	0.4	V
FAULTB Output High Leakage Current	I_{FAULTB}	V_{FAULTB} = 5V			1.0	µA
GRADE SELECTION						
RGRADE Output Current	I_{RGRADE}		46.2	50.0	53.5	µA
RGRADE_0 Range		RGRADE[2:0] = 0b000			8.1	kΩ
RGRADE_1 Range		RGRADE[2:0] = 0b001	10.1		12.1	kΩ
RGRADE_2 Range		RGRADE[2:0] = 0b010	14.8		16.2	kΩ
RGRADE_3 Range		RGRADE[2:0] = 0b011	19.6		22.3	kΩ
RGRADE_4 Range		RGRADE[2:0] = 0b100	26.5		30.5	kΩ
RGRADE_5 Range		RGRADE[2:0] = 0b101	36.0		42.6	kΩ
RGRADE_6 Range		RGRADE[2:0] = 0b110	50.6		58.0	kΩ
RGRADE_7 Range		RGRADE[2:0] = 0b111	70.0			kΩ
CHARGE PUMP						
Charge-Pump Frequency	f_{CPP}			8.192		MHz
Charge-Pump Output Voltage	V_o	V_{CPP} - V_{CPN}, I_{CPP} = 250µA	7.4		9	V
Charge-Pump Power-Good Threshold	V_{CPP_OK}	Rising threshold	5.68	6.9	7.85	V
THERMAL SHUTDOWN						
Thermal-Warning Threshold	TH_WARN	Rising temperature		150		ºC
Thermal Shutdown	TH_SHDN	Rising temperature		160		ºC
Thermal-Shutdown Hysteresis	HYS_SHDN			20		ºC

Note 1: Limits are 100% tested at T_A = +25°C. Limits over the operating temperature range and relevant supply voltage range are guaranteed by design and characterization.
Note 2: Excludes bond-wire resistance. Typical package bond-wire impedance = 15mΩ per pin.
Note 3: Static logic inputs with V_{IL} = GND and V_{IH} = V_{DDIO} (Note 1). SCSB = V_{IH} (if pullup active).
Note 4: No internal safety pullup/pulldown impedances active, input buffers only.
Note 5: Internal safety pullup/pulldown impedances available, with enable function.
Note 6: Applications must afford time for the device to drive data on the SDO bus and meet the µC setup time prior to the µC latching in the result on the following SCLK RE. In practice, this is determined by loading and µC characteristics, and the relevant t_{DOT}/t_{DOE}.

Pin Configuration



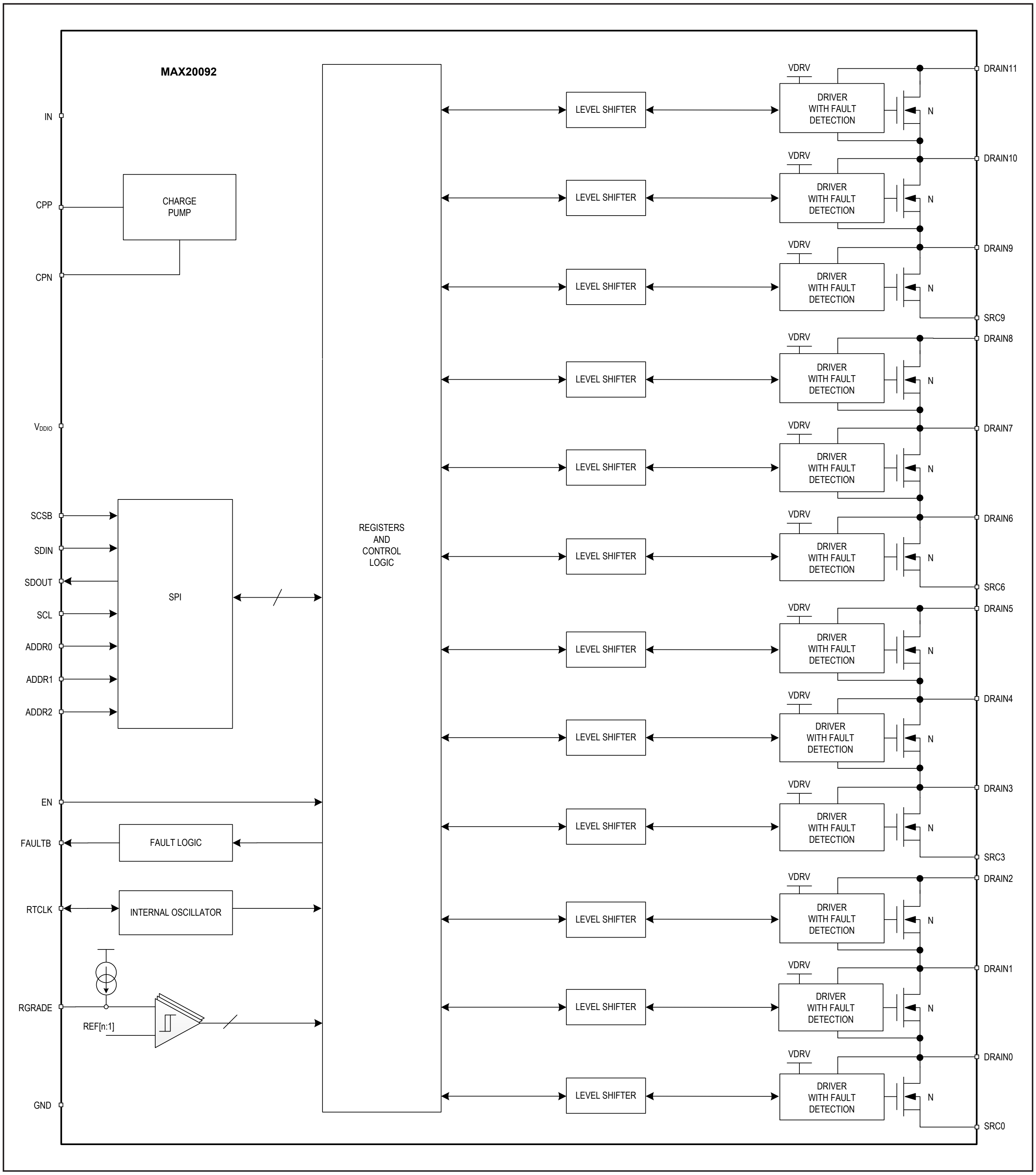
Pin Description

PIN	NAME	FUNCTION
1	IN	5V Power-Supply Input. Bypass with a 0.1μF (min) ceramic capacitor.
2	GND	Ground Connection
3	SCSB	Chip-Select Pin for SPI. This pin is pulled low to enable the SPI.
4	SCL	Clock Input Pin for SPI
5	SDIN	Data Input Pin for SPI
6	SDOUT	Data Output Pin for SPI
7	FAULTB	Fault Output for Fault-Condition Report. The FAULTB pin is asserted (pulled low) to indicate that a fault condition has been detected, as defined by the STAT_GEN register. The FAULTB pin is deasserted (released to high) after the part sends its fault response address successfully, or the fault condition is cleared by an SPI write command.
8	VDDIO	Digital Interface Power-Supply Input
9	EN	Enable Pin. Drive EN to V_{DDIO} to enable the device; drive it to GND to reset the registers and stop switching.
10	RGRADE	Grade Selection Pin. This pin sets the REV_ID bits 5 to 7 in the NO_OP (0x00) register. Connect a 69.8kΩ to 0Ω resistor between RGRADE and GND, or drive with a voltage source between GND and IN capable of sinking 50μA to provide part grading information that can be read back from register NO_OP (0x00).
11	ADDR2	SPI Interface Address Selection. Connect to V_{DDIO}, GND, or SDI.
12	ADDR1	SPI Interface Address Selection. Connect to V_{DDIO}, GND, or SDI.
13	ADDR0	SPI Interface Address Selection. Connect to V_{DDIO}, GND, or SDI.

Pin Description (continued)

PIN	NAME	FUNCTION
14	CLK	External PWM Clock Input and Internal Oscillator Output Pin
15	SRC0	Source of Internal Switch 0
16	DRAIN0	Drain of Switch 0. Connect to anode of LED0 and cathode of LED1.
17	DRAIN1	Drain of Switch 1. Connect to anode of LED1 and cathode of LED2.
18	DRAIN2	Drain of Switch 2. Connect to anode of LED2 and cathode of LED3 (if LED3 in series with LED2).
19	SRC3	Source of Internal Switch 3
20	DRAIN3	Drain of Switch 3. Connect to anode of LED3 and cathode of LED4.
21	DRAIN4	Drain of Switch 4. Connect to anode of LED4 and cathode of LED5.
22	DRAIN5	Drain of Switch 5. Connect to anode of LED5. If LED6 in series with LED5, also connect to cathode of LED6.
23	SRC6	Source of Internal Switch 6
24	DRAIN6	Drain of Switch 6. Connect this pin to anode of LED6.
25	DRAIN7	Drain of Switch 7. Connect to anode of LED7 and cathode of LED8.
26	DRAIN8	Drain of Switch 8. Connect to anode of LED8 and (if LED9 in series with LED8) cathode of LED9.
27	SRC9	Source of Internal Switch 9
28	DRAIN9	Drain of Switch 9. Connect to anode of LED9 and cathode of LED10.
29	DRAIN10	Drain of Switch 10. Connect to anode of LED10 and cathode of LED11.
30	DRAIN11	Drain of Switch 11. Connect to anode of LED11.
31	CPP	Charge-Pump Positive Terminal. Connect a 0.1μF ceramic capacitor from this pin to CPN.
32	CPN	Charge-Pump Negative Voltage. Connect a 0.1μF ceramic capacitor from this pin to CPP.
—	EP	Exposed Pad. Connect EP to GND.

Functional Diagram



Detailed Description

The MAX20092 12-switch matrix manager IC for automotive lighting includes a 12-switch array for bypassing individual LEDs in a single- or dual-string application. The IC features 12 individually controlled n-channel MOSFET switches rated for 10V with an on-resistance of 0.100Ω. A single current source can be used to power all the LEDs connected in series. Individual LEDs can be dimmed by turning on and off the bypass switches across each LED. The IC can also be configured in two strings with 6 switches in series per string or 4 strings with 3 switches per string. A separate current source powers each string. Each switch can be connected across one or two LEDs in series. The IC also includes an internal charge pump that provides the gate drive for each of the LED bypass switches. The low on-resistance of the switches minimizes conduction loss and power dissipation.

The IC features a serial peripheral interface (SPI) for serial communication. The IC is a slave device and uses the SPI to communicate with an external microcontroller which is the master. Each of the 12 switches can be independently programmed to bypass the LEDs across each of the switches in the string. Each switch can be turned fully on, fully off, or dimmed with or without a fade transition mode. The PWM frequency can be set by an internal oscillator or can also be set to an external clock source. The IC features open-LED protection, as well as open- and short-LED fault reporting through SPI. The IC is available in a 5mm x 5mm 32-pin side-wettable TQFN (SWTQFN) package with a thermally enhanced exposed pad. The SPI is capable of operating from 1.8V to 5.5V and is driven from the V_{DDIO} supply.

Each switch has an individual driver, overvoltage-protection circuit, and a diagnostic circuit referenced to the source of that switch. This configuration allows for fully dynamic operation with the switches above it and below it. The IC monitors overvoltage conditions on each switch and automatically protects them in the event of an open-LED connection. The open-LED fault can be programmed at one of the two threshold levels: 4.5V or 9.0V. The short-LED fault can also be programmed at one of two threshold levels: 1.0V or 4.4V. When a shorted-LED fault is detected in a channel, the channel switch continues with what it is doing. The IC detects open-LED conditions as well as shorted-LED conditions and reports them through the fault-reporting network. The IC also detects and reports a thermal-warning condition. The FAULTB signal pulls low if thermal shutdown is activated.

Power-On Reset

Once the IC is powered, an internal power-on reset (POR) signal sets all the registers to their default states. All 12 switches are in the on state upon a POR (all LEDs are off). The LEDs remain off until a command is received by the SPI. To ensure reliable operation, the IN supply voltage (V_{IN}) must be greater than $V_{IN\text{-}POR}$. If V_{IN} falls below $V_{IN\text{-}POR}$, the registers reset to their default state. The IN voltage must be greater than $V_{IN\text{-}POR}$ and EN = 1 for SPI operation.

Enable Function (EN)

When the EN pin is low, all registers are reset to their default values and all the bypass switches are turned on.

When the EN pin is brought high, the IC is enabled and the SPI registers can be written to, as described in the *SPI Transactions* section. The bypass switches remain in their default on state until the SPI is used to enable LED dimming.

Internal Switches

Each switch connected between DRAINn and DRAINn-1 has a typical on-resistance of 0.10Ω. This measurement includes the on-resistance of the internal switch and the resistance of the bond wires to the DRAINn and DRAINn-1 pads. Each bypass switch, when driven to an off state, allows the string current to flow through the corresponding parallel-connected LED, turning the LEDs on. Driving the bypass switch to an on state shunts the current through the bypass switch and turns the LEDs off. Each bypass switch can have one or two LEDs in series across it.

LED Fault Detection and Protection

The IC is able to detect a shorted LED, open LED, and open trace between the device and the LED. To detect and report an LED fault, several conditions must be met. First, the LED switch must be operating, so SW_GO_EN and the EN pin should both be high. LED-open and LED-short detection requires the switch be open, so the duty cycle must be greater than zero. Conversely, open-fault detection requires the switch to be closed, so PWM duty cycle must be less than 100%. In general, it takes up to one dimming cycle to make sure these conditions have been met after a fault condition is applied. This period depends on the PWM dimming frequency.

LED Open-Fault Detection and Protection

An open-LED fault is triggered when the voltage between the individual LED switch DRAIN node and switch SOURCE node exceeds V_{OTH} and is reported in register STAT_OPEN_LED (0x06). The switch is closed when an open-LED detection occurs and remains closed until the next PWM dimming open-switch request occurs. By default, the open fault results in the FAULTB pin being driven low; however, open faults can be masked by writing 0b1 to the MSK_OPEN_LED bit in the CNFG_MSK (0x04) register. If an open-LED fault is detected multiple times, it is recommended that the OPEN_OVRD (0x08) register be updated to force the corresponding LED switch to remain closed continuously to provide a bypass for the faulty LED.

LED Short Detection

A short-LED fault is triggered when the voltage between the switch DRAIN node and the switch SOURCE node is below V_{STH} for an open switch condition, and is reported in the STAT_SHRT (0x09) register. The LED short comparator is sampled at the end of each LED pulse to avoid false detects during the beginning of the pulse. No action is taken with the switch in response to detecting a short-LED fault, with it continuing to operate as programmed. The short fault, by default, results in FAULTB being driven low; however, short faults can be masked by writing 0b1 to MSK_SHRT in the CFG_MSK (0x04) register.

Open-Trace Detection

An open-trace fault is triggered when the current through the closed switch is less than I_{LED_MIN} and is reported in the STAT_OPEN_TRACE (0x07) register. No action is taken with the channel switch in response to detecting an open trace, with it continuing to operate as programmed. By default, the open-trace fault results in the FAULTB pin being driven low; however, open-trace faults can be masked by writing 0b1 to MSK_OPEN_TRACE in the CFG_MSK (0x04) register.

The open-trace fault is sampled before the rising PWM edge, which is the edge turning the switch off. If there is an open trace to the drain side of the switch, the switch above it has to be open during the PWM rising edge of the switch that is detecting the condition. For example, if there is an open trace on the drain side of N1 (see Figure 1), then N2 has to be open when N1 is turning on and similarly, for detecting an open-trace condition on the drain of N0 (N1 must be open when N0 is turning on). For the top-most-switch N2 drain connection, and the bottom-most-switch source connection, there is no constraint to detect the open-trace condition.

Thermal Shutdown

The IC features an on-chip temperature-protection circuit to prevent the device from overheating.

When the die temperature rises above the thermal-warning threshold (+150ºC), the TH_WARN bit is set, causing the FAULTB pin to be asserted if unmasked, but no action is taken with the switches. If asserted, the FAULTB pin remains asserted until the die temperature drops below the thermal-warning threshold, and a read of the STAT_GEN (0x05) register has occurred. To clear the TH_WARN bit, the die temperature must be below the thermal-warning threshold.

When the die temperature rises above the thermal-shutdown threshold (+160ºC), the TH_SHDN bit is set, causing the FAULTB pin to be asserted and all switches to either be closed (LEDs turned off) or opened (LEDs turned on), depending on the value of the CNFG_MSK (0x04) register. Switches remain static and the FAULTB pin remains asserted until the die temperature drops below the thermal-warning threshold, and a read of the STAT_GEN register has occurred.

When the device recovers from thermal shutdown, it resumes operation from where it was before the thermal shutdown. The TH_WARN and TH_SHDN bits are cleared on read.

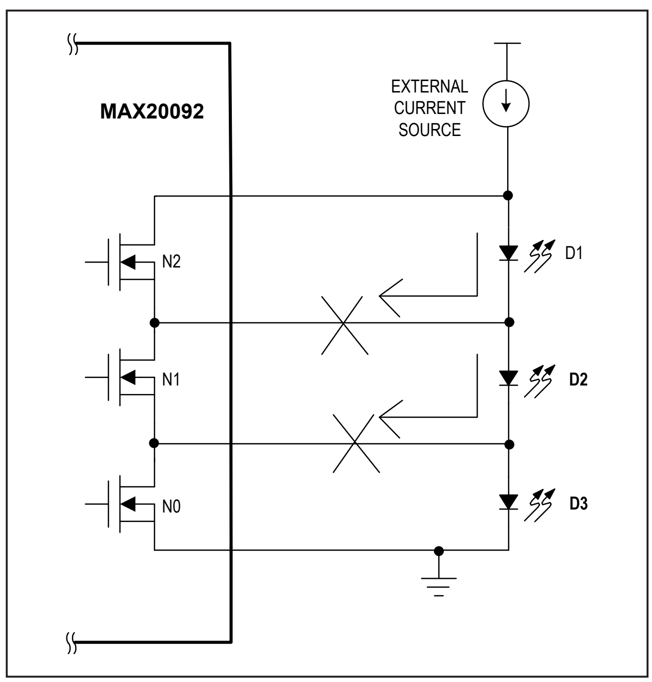


Figure 1. Open-Trace Detection

FAULTB Pin Operation

The FAULTB pin is an active-low, open-drain output. The IC asserts a fault if any of the bits are set in the SPI STAT_GEN register (0x05). See the STAT_GEN register (0x05) in the *Register Map* section and the subsequent register table for more details on the definition of each bit. Once all the bits have been cleared by reading the appropriate SPI registers, the FAULTB pin is released. **Note:** If the conditions causing the fault bit(s) to be asserted persist, the bits are immediately set again. The FAULTB pin remains low.

PWM Clock and Synchronous Operation with Multiple Devices

The PWM clock for the IC can be selected from the internal oscillator or from an external clock source driving the CLK pin. The CLK pin is bidirectional, allowing a single device to be the master clock, providing a common clock source to multiple devices. The PWM clock source and CLK pin direction are configured through PWM_CLK[1:0] in the CNFG_GEN (0x03) register. The default value is internal oscillator with the CLK pin disabled. For synchronous operation with multiple devices, use the PWM_CLK_SEL bits in the CNFG_GEN (0x03) register to set the master with internal oscillator and CLK pin output, and the slave devices with external oscillator and CLK pin input.PWM dimming frequency is programmable by setting the value of the DIV[1:0] bits in the CNFG_GEN (0x03) register, which sets the divide ratio for both the internal (8.192MHz) and external clock sources. When disabled, the CLK pin is high impedance with a 100kΩ pulldown resistor.

Parallel Operation for Higher Current Applications

The switches in the IC can handle current up 1.6A (max); however, for applications that require higher currents, the switches can be configured in parallel. For example, if the current capability needs to be doubled with 6 LEDs, connect switch SW0 in parallel to SW6, SW1 in parallel to SW7, and so on. Make sure the switches connected in parallel have the same phase shift and PWM dimming duty cycles.

PWM Dimming

The IC provides 12-bit programmable dimming on each individual switch. An internal 12-bit counter (COUNT) is generated according to the clock settings. The switch turns off when COUNT is equal to the delay set by the corresponding PSFT register and stays off until the COUNT exceeds the sum of PSFT and PWM duty-control registers. In this way, the duty cycle and relative phase shift of the individual switches can be set independently (see Figure 2).

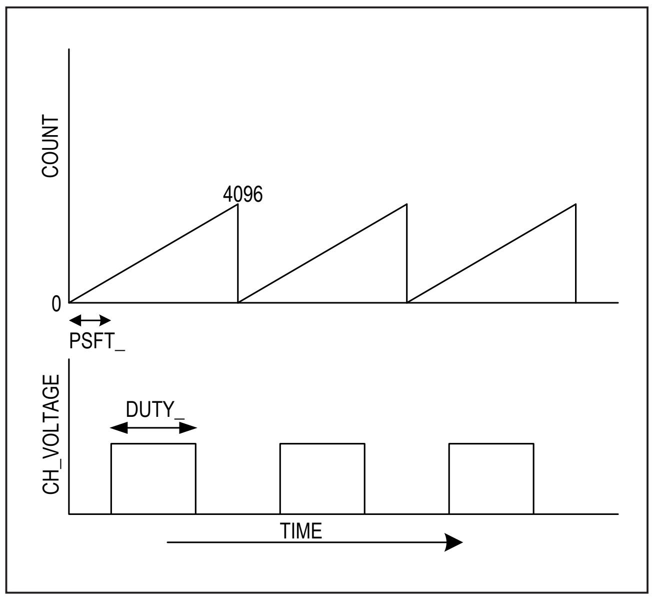


Figure 2. PWM Dimming

Dimming With and Without Fade

Each switch of the IC can be independently programmed to perform dimming without fade transition, or dimming with fade transition. For dimming without fade transition, the dimming changes from the initial value to the target value in one dimming cycle. For dimming with fade transition, the dimming changes transitionally step by step, starting from the initial value to the target value in multiple dimming cycles, following a predetermined exponential curve.

To enable dimming with fade transition, set the FADE bit to 1 and the DUTY bits to the target value for the specific switches. Each transitional step value is calculated using 12 bits according to the following formula:

$$DUTYnext = DUTYnow \times CF$$

where DUTY is the duty cycle, and CF the constant factor.

CF = 1.0625 and CF = 0.9375 for an up transition and down transition, respectively.

DUTYnext continues to be updated according to the formula until DUTYnext reaches the target value. The transition period is defined by the TDIM_ register for the switch. The number of transitional steps depends on the distance between the initial value and the target value. The maximum number of transitional steps from 1(/4095) to 4095(/4095) is 115 steps. See Figure 3 for the up-transition curve. The number of transitional steps depends on the distance between the initial value and the target value. The maximum number of transitional steps from 4095(/4095) to 1(/4095) is 111 steps. See Figure 4 for the down-transition curve.

Duty-cycle steps smaller than CF update in 1 step.

Each step runs TDIM_ PWM dimming cycles, and each dimming cycle consists of 4096 clock cycles; therefore Tstep = TDIM_ x 4096.

Grade Selection

The IC provides eight levels of detection between 0 and 4V on the RGRADE pin, which can be read back as part of the REV_ID in the NO_OP (0x00) register. The RGRADE pin sources 50μA, allowing the use of an external resistor between RGRADE and GND to set the voltage level. See Table 1 for threshold levels and recommended resistor values.

Table 1. RGRADE Recommended Values

RGRADE[2:0]	RESISTOR VALUE R_{RGRADE} (KΩ, 1%)	RGRADE VOLTAGE (V)
0b000	0 (Short to GND)	0.0
0b001	11.3	0.565
0b010	15.8	0.790
0b011	21.0	1.05
0b100	28.7	1.435
0b101	40.2	2.01
0b110	54.9	2.745
0b111	75.0	3.75

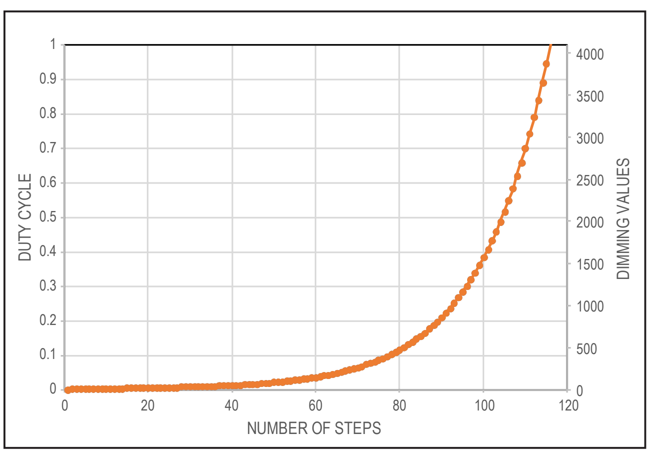


Figure 3. Up-Transition Curve

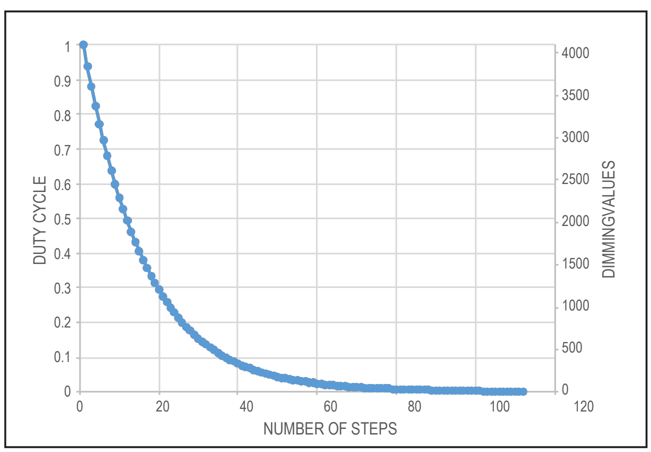


Figure 4. Down-Transition Curve

4-Wire Serial Interface

General Description

The IC recognizes transactions during which SCSB is low. The serial peripheral interface (SPI) latches SDI input data on SCLK's rising edge and executes commands on SCSB's rising edge, subject to qualification criteria. SDO is active when SCSB is low, and is held in three-state when SCSB is high. Star connections of up to 26 devices are supported. Internal pullup and pulldown resistors are included on all input pins to enhance safety in the event of a broken wire/trace. SPI_ERR and CRC checks are offered for interface verification.

The following features are supported by the SPI:

- 32-Bit SPI Frame
- SCSB Active-Low Device Selection
- SCSB Rising-Edge Transaction Execution
- SDI Data Latched on SCLK Rising Edge
- 7-Bit Register Addressing with 13-Bit Data
- 128 Write/Read Accessible Registers (00–7F\h)
- Embedded CRC3 Checking
- SPI_ERR Interface Error-Indicator Bit
- Configurable Internal Pullup/Pulldown Terminations on All Input Pins
- Star Connections of Up to 26 Devices Supported on Single SCSB Line

Overview

The MAX20092 interface is SPI, QSPI, Microwire, and DSP compatible. The operation and timing criteria of the SPI is shown in Figure 5. The device is programmed by a 32-cycle SPI instruction, framed by a low interval on SCSB. The start of the transaction is defined by the SCLK rising edge following the SCSB falling edge (subject to t_{CSH0} and t_{CSS0} timing criteria). Transactions including a number of SCLK rising edges not equal to 32 are not qualified for execution (also based on t_{CSA}, t_{CSH1}, and t_{CSQ} timing criteria). Qualified transactions are executed on the rising edge of SCSB. To abort a command sequence, the rise of SCSB must precede a qualified 32nd rising edge of SCLK (meeting the t_{CSA} timing requirement).If SCLK stops toggling for more that 256µs while SCSB remains low, the device declares a SPI timeout error and asserts CLK_ERR.

The SDI content of the SPI transaction consists of a leading read/writeB (R/WB) bit followed by device ID, address, input data information, and CRC. Data is latched into the registers on SCLK rising edges, subject to setup and hold criteria (t_{DS}, t_{DH}).

SDO is actively driven during intervals where SCSB is low. SDO is initially driven by the device when SCSB falls (t_{DOE} timing applies), presenting the MSB of the output data (the SPI_ERR bit for all transactions). Following the initial SCLK rising edge, SDO is updated in response to SCLK falling edges, conforming to hold- and transition-time criteria ($t_{DOH,}$ t_{DOT}), allowing the µC to latch the data on SCLK rising edges. When SCSB is high, the SDOUT line is high impedance, allowing other devices to access the SDO bus.

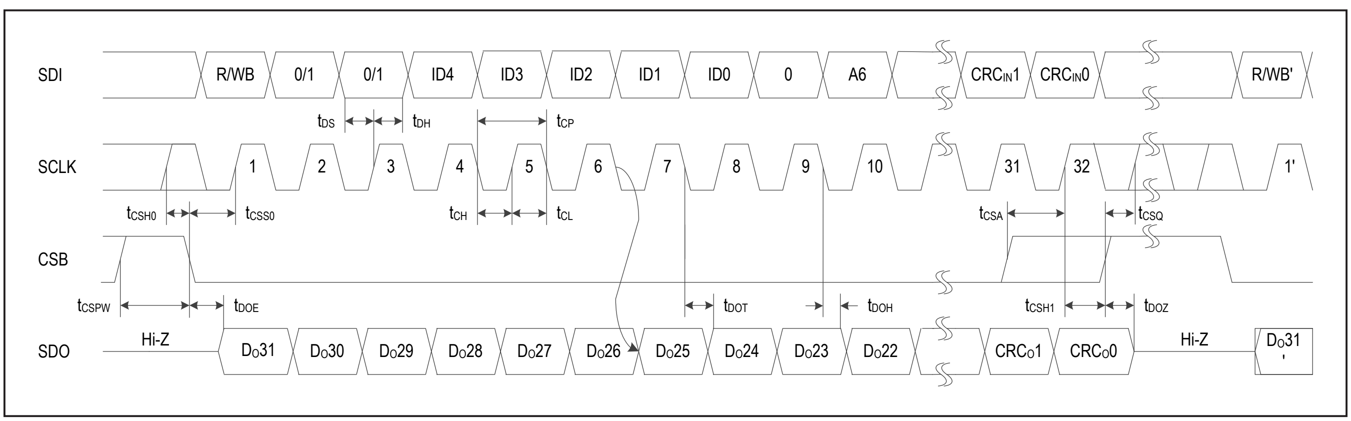


Figure 5. SPI Timing Diagram

Device Connections

The SPI ensures compatible operation with standard microcontrollers (μCs) from a variety of manufacturers. The μC always operates as the master, and is able to initiate read and write transactions to individual slave devices (using standard connections), or groups of slave devices in a star configuration. The device(s) always operate in the slave role when connected to a μC and cannot initiate a SPI transaction.

The SCLK line should be driven by the master and hooked up to all slave devices. Only the slave devices (or group of slave devices) with its SCSB line low will accept SCLK. SPI transactions to the slave devices are defined by SCLK rising edges. The MAX20092 can therefore support SPI formats with (CPOL=0, CPHA=0) or (CPOL=1, CPHA=1), see Figure 6 for alignment examples. The SDI line should be hooked up to a master-out/slave-in (MOSI) port. A single SDI line can be routed to all SPI slave devices sharing the interface, but only the slave device (or group of slave devices) with the SCSB line low will accept SDI data. The μC should update SDIN in response to SCLK falling edges so the slave can latch data in on SCLK rising edges. The SDO line should be hooked up to a master-in/slave-out (MISO) port. A single SDO line can be routed to all slave SPI devices sharing the interface, but only the slave device (or group of slave devices) with its SCSB line low can access and drive the shared SDO bus. The slave updates SDO in response to SCLK falling edges, so the μC can latch data in on SCLK rising edges.

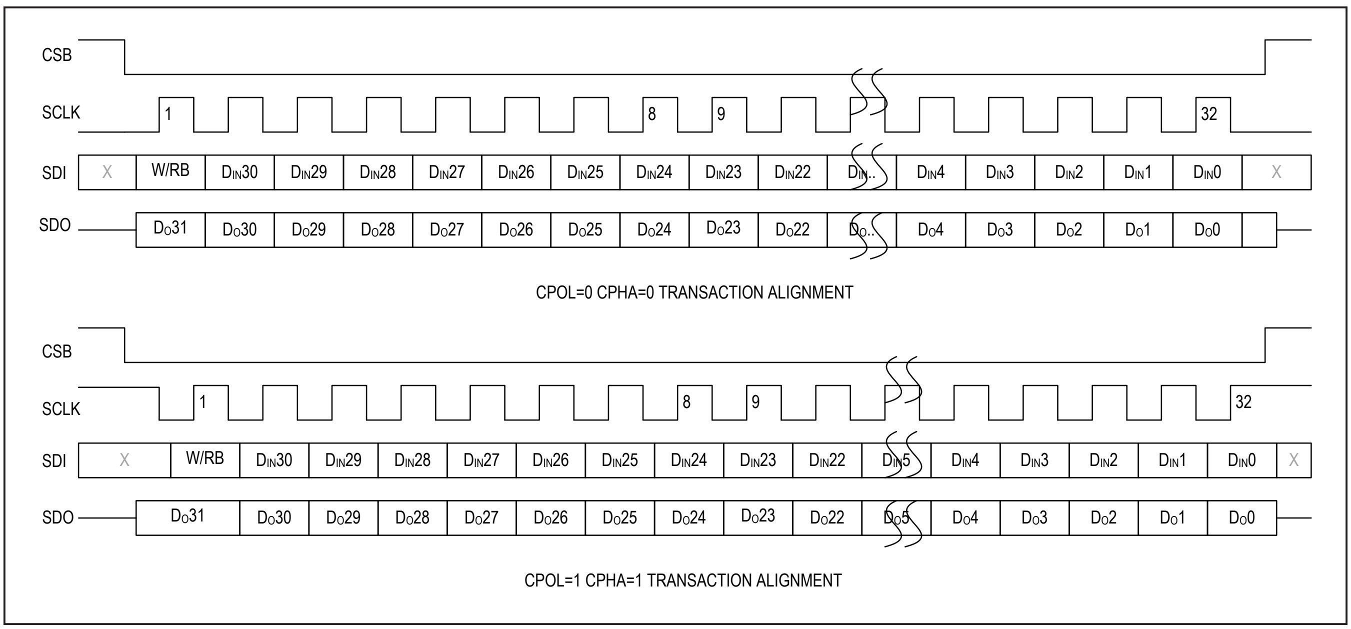


Figure 6. SPI Transaction Format

Star Device Connections

The IC's SPI allows multiple devices to share the interface, with the active device for the transaction being selected by a predetermined device ID number. The SCSB, SCLK, SDI, and SDO lines are common to all devices. Transaction-qualification criteria remains in effect, and in write mode the device executes the instructions present in the 32 bits of a qualified transaction. In read mode, the device returns the requested data through SDO during the read-mode transaction.

An example of a standard Star connection is shown in Figure 7.

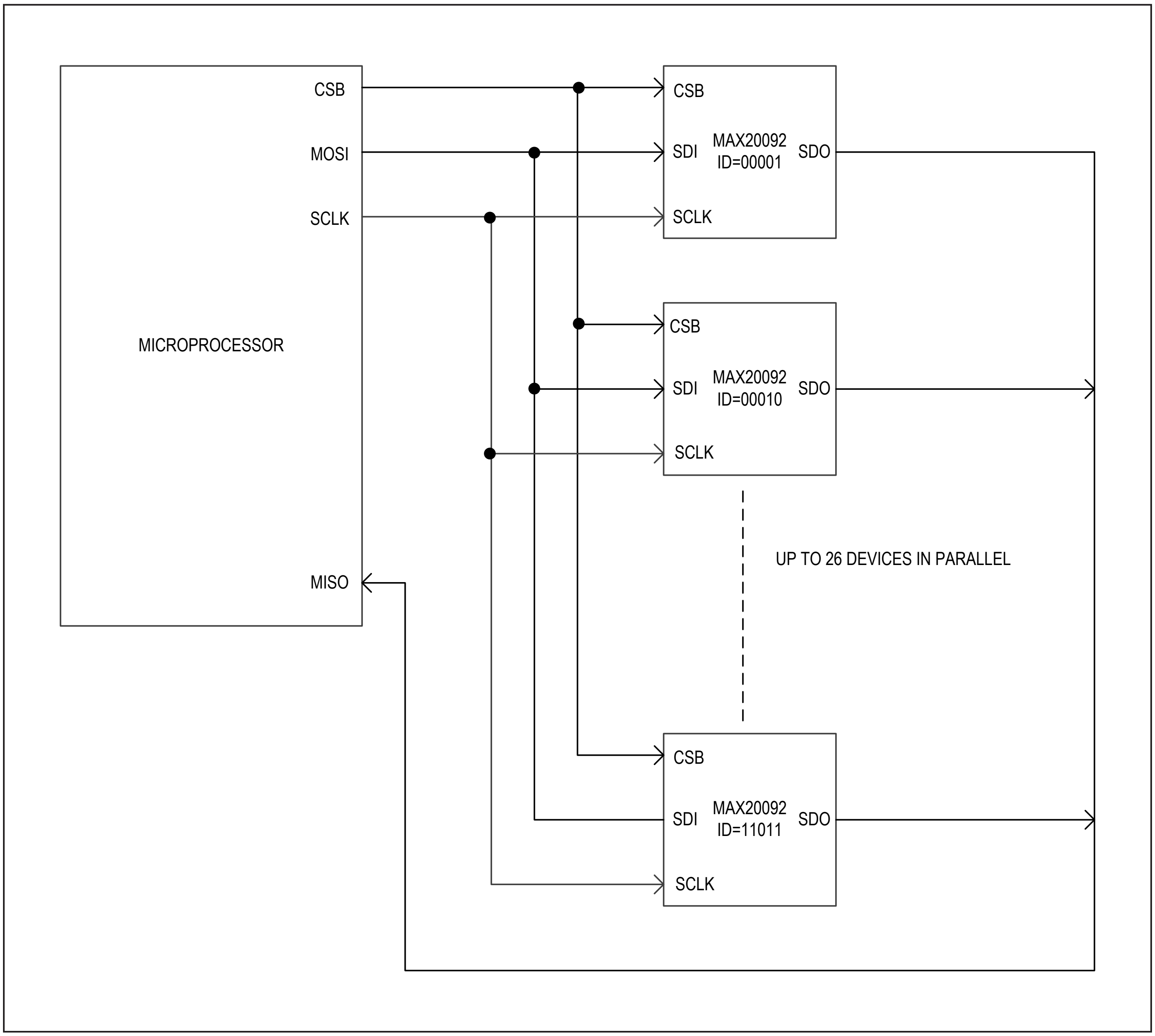


Figure 7. Star Connections

Device Identification Number

Each device in the star connection is given a unique device ID number through hard-wiring pins ADDR2, ADDR1, and ADDR0. Each pin can be connected to three different values (V_{DD}, GND, and SDI), for a total of 26 valid combinations. The combination of {GND, GND, GND} is reserved for a general-call ID and should not be used.

Table 2. Device ID Mapping

ADDR2	ADDR1	ADDR0	DEVICE_ID[4:0]
GND	GND	GND	Invalid, reserved for general-call ID.
GND	GND	SDI	00001
GND	GND	V_{DD}	00010
GND	SDI	GND	00011
GND	SDI	SDI	00100
GND	SDI	V_{DD}	00101
GND	V_{DD}	GND	00110
GND	V_{DD}	SDI	00111
GND	V_{DD}	V_{DD}	01000
SDI	GND	GND	01001
SDI	GND	SDI	01010
SDI	GND	V_{DD}	01011
SDI	SDI	GND	01100
SDI	SDI	SDI	01101
SDI	SDI	V_{DD}	01110
SDI	V_{DD}	GND	01111
SDI	V_{DD}	SDI	10000
SDI	V_{DD}	V_{DD}	10001
V_{DD}	GND	GND	10010
V_{DD}	GND	SDI	10011
V_{DD}	GND	V_{DD}	10100
V_{DD}	SDI	GND	10101
V_{DD}	SDI	SDI	10110
V_{DD}	SDI	V_{DD}	10111
V_{DD}	V_{DD}	GND	11000
V_{DD}	V_{DD}	SDI	11001
V_{DD}	V_{DD}	V_{DD}	11010

Individual Call, Global Call, Cluster-Call Command

For the first 8 bits of the transaction, the master sends:

{Read/write bit, CMD1, CMD0, ID4, ID3, ID2, ID1, ID0}

Where CMD[1:0] is the Call-Type command. This field allows the master to access:

- A single device (individual call)
- Multiple devices (cluster call), or
- All devices (global call) on the shared SCSB line

Safety Pulldown Resistors

To guard against broken SPI connections, the IC includes internal safety terminations on all interface input ports. SCLK and SDI have internal pulldowns to GND. SCSB and RESETB (if present) have internal pullups to V_{DDIO}. All safety resistors are 100kΩ nominal.

The internal safety resistors can be individually enabled or disabled using SPI configuration bits (SFT_CLK, SFT_SDI, SFT_SCSB) with a high state indicating the safety termination is enabled/engaged and a low state indicating it is disengaged. This allows the user to eliminate loading currents when the safety resistors are not needed.

SPI Transactions

Write-Mode Transactions

A properly constructed write-mode transaction is made up of 32-bit data frames. Each SDI data frame from the master (or the previous device in the chain) contains a R/WB bit, a 2-bit individual call command, a 5-bit device identification number, a 7-bit address, 13 bits of input data or instructions, and a 3-bit CRC. During a write-mode transaction, the device outputs data on the SDO line; both transaction log and repeated transaction data are transferred through SDO. The device only accepts and executes qualified SPI transactions based on the last 32 bits of data received. Details of write-mode transactions are explained below and summarized in Figure 8.

Write Bit - R/WB = 0 (DIN31):

Write-mode transactions are identified by R/WB = 0 in the MSB position of a 32-bit data frame.

Individual Call Command - CMD[1:0] = 01 (DIN[30:29]):

Individual write transactions are identified by CMD[1:0] = 01 in the DIN[30:29] positions of the frame.

Table 3. Call Type Command

CMD1	CMD0	CALL TYPE	DESCRIPTION
0	0	Invalid	—
0	1	Individual	One device accessed at a time
1	0	Global or Cluster	All or multiple devices accessed at the same time
1	1	Invalid	—

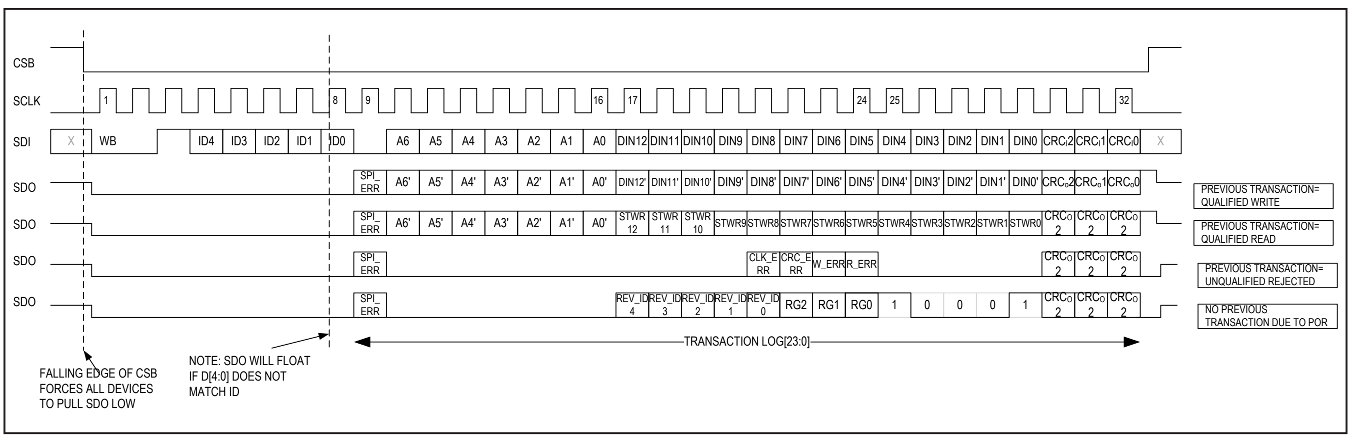


Figure 8. Write-Mode Transaction

Device ID - ID[4:0] (DIN[28:24]):

Note: After the 8th bit has been received (rising 8th SCLK edge), the selected slave continues to drive the SDO line with the contents of the transaction log for the remaining 24 bits. The other slaves release the SDO line such that only one slave is driving.

Address - A[6:0] (DIN[22:16]):

Write-mode transactions allow new information to be written to internal configuration registers within the device. The configuration register address to be written is indicated by A[6:0] within the data frame. In this format, up to 128 register addresses are supported (0\h thru 7F\h) for write-mode access.

Note: DIN23 must always be 0.

Input Data - DIN[12:0] (DIN[15:3]):

The next 13 bits of data in the 32-bit data frame represent data that will be written to the requested register, or describe internal operations to be executed.

SDI CRC Bit - CRCi[2:0] (DIN[2:0]):

Write-mode transactions are protected by a 3-bit cyclic redundancy check (CRC) on the SDI data frame. CRC is provided by the master on the last 3 bits on SDI.

CRC is calculated by applying the DIN[31:3] message (i.e., Address + Input Data) on polynomial 0x5 ($x^3 + x^1 + x^0$) with a start value of binary 000.

The receiving device (slave) will calculate its own CRC using the same polynomial and starting value. The slave only accepts/executes the command if its own CRC matches the last 3 bits on SDI.

Output Data - DO[31:0] and LOG[23:0]:

During write-mode transactions, the device outputs data through the SDO line.

In write mode, the device pulls down SDO immediately following the falling edge of SCSB. The first 8 bits are always 0. The remaining 24 bits are the contents of the internal transaction log register (LOG[23:0]).

See the *Internal Transaction Log* section for a detailed explanation of content.

If further SCLK cycles are provided, SDO outputs 0 for the remainder of the transaction frame and CLK_ERR is set.

Note: This method also provides the μC an opportunity to check the SPI integrity, since the transaction log content of the previously qualified/executed transaction will be relayed back to the μC through SDO during each complete single or extended transaction.

Write-Mode Qualification Check (SPI_ERR):

To qualify for write-mode execution, the following conditions must be met:

- SPI transaction must be exactly 32 bits in length (no CLK_ERR recorded)
- SDI data frame CRC check must pass (no CRC_ERR recorded)
- A[6:0] must select a valid write-accessible register or command (no W_ERR recorded)

If the SPI transaction is qualified, the instruction is executed, any requested internal register contents are updated, and the Internal Transaction Log is updated to indicate the successful transaction.

If the SPI write transaction is not qualified, the instruction is not executed, the device's internal SPI_ERR indicator and appropriate SPI diagnostic bit are set, and the Internal Transaction Log is updated to indicate the failed transaction. The SPI_ERR bit is returned in response to later read and write-mode transactions, notifying the μC that the SPI interface may be compromised.

Read-Mode Transactions:

A properly constructed read-mode transaction is made up of 32-bit data frames. Each SDI data frame from the master contains a R/WB bit, a 2-bit Individual Call command, a 5-bit Device Identification number, a 7-bit address, 13 bits of data set to all zeros (000\h), and a 3-bit CRC.

During a read-mode transaction, the IC outputs data on the SDO line; the content of the SDO data frame is described in detail below. The MAX20092 only accepts qualified SPI transactions, based on the last 16 bits of data received. Details of read-mode transactions are explained below and summarized in Figure 9.

Read Bit - R/WB = 1 (DIN31):

Read-mode transactions are identified by R/WB = 1 in the MSB position of a 32-bit data frame.

Individual Call Command - CMD[1:0] = 01 (DIN[30:29]):

Individual write transactions are identified by CMD[1:0] = 01 in the DIN[30:29] positions of the frame.

Device ID - ID[4:0] (DIN[28:24]):

Note: After the 8th bit has been received (rising 8th SCLK edge), the selected slave continues to drive the SDO line with the contents of the transaction log for the remaining 24 bits. The other slaves release the SDO line such that only one slave is driving.

Address - A[6:0] (DIN[22:16]):

Read-mode transactions allow new information to be read from internal registers within the device. The register address to be read back is indicated by A[6:0] within the data frame. In this format, up to 128 register addresses are supported (00\h thru 7F\h) for read-mode access. **Note:** DIN23 must always be 0.

Input Data - DIN[12:0] (DIN[15:3]):

The 13-bit input data in a read mode must be set to zero (000\h).

SDI CRC Bit - CRCi[2:0] (DIN[2:0]):

Read-mode transactions are protected by a 3-bit cyclic redundancy check (CRC) on the SDI data frame. CRC is provided by the master on the last 3 bits on SDI.

CRC is calculated by applying the DIN[31:3] message (i.e., Address + Input Data) on polynomial 0x5 (x^3+ x^1+ x^0) with a start value of binary 000.

The receiving device (slave) calculates its own CRC using the same polynomial and starting value. The slave only accepts/executes the command if its own CRC matches the last 3 bits on SDI.

Output Data - Current Status[6:0]+Data Requested[13:0]:

In a read operation, the device pulls down SDO immediately following the falling edge of SCSB. **Note:** The first 8 bits of the transaction are always 0.

The device then relays the SPI_ERR status, up to 7 bits of general-status data (STR[6:0]), the 13 bits of data requested by A[6:0], and a calculated 3-bit CRC in direct response to an incoming read-mode transaction.

If further SCLK cycles are provided, SDO outputs 0 for the remainder of the transaction frame and CLK_ERR is set.

SDO CRC Bits - CRCo[2:0] (DO[2:0]):

Read-mode transactions are protected by a 3-bit cyclic redundancy check (CRC) on the SDO data frame. CRC is provided by the slave on the last 3 bits on SDO.

CRC is calculated by applying the message DO[23:3] (i.e., ST + Data Requested) on polynomial 0x5 (x^3+ x^1+ x^0) with a start value of binary 000.

The receiving device (master) calculates its own CRC using the same polynomial and starting value. The master only accepts/executes the command if its own CRC matches the last 3 bits on SDO.

Read-Mode Qualification Check (SPI_ERR):

To qualify for read-mode execution, the following conditions must be met:

- SPI transaction must be exactly 32 bits in length (no CLK_ERR recorded)
- SDI data frame CRC check must pass (no CRC_ERR recorded)
- DIN[12:0] must be all zeros (no R_ERR recorded)
- Individual call command CMD==01 received (no R_ERR recorder)

If the SPI read transaction is qualified, any clear-on-read internal register contents are updated, and the Internal Transaction Log updated with the content requested by the successful transaction.

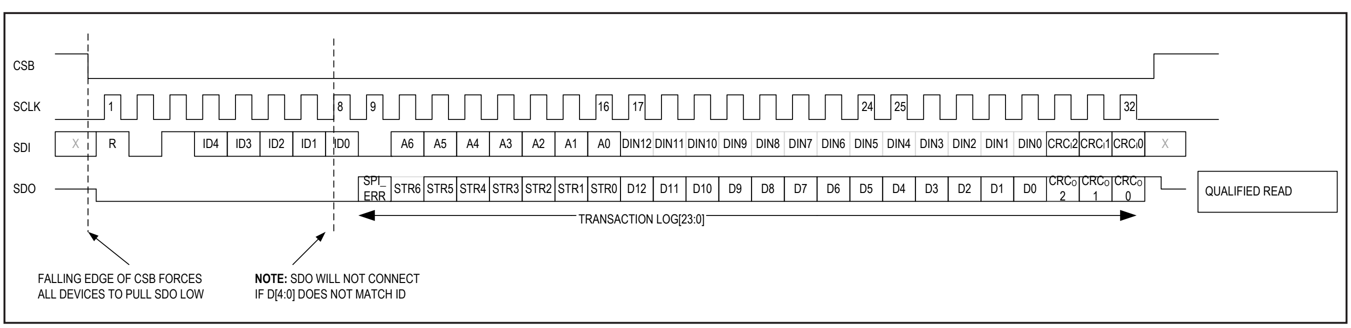


Figure 9. Read-Mode Transaction

If the SPI read transaction is not qualified, clear-on-read internal register contents are not updated, the device's internal SPI_ERR indicator and appropriate SPI diagnostic bit are set, and the Internal Transaction Log updated to indicate the failed transaction. This SPI_ERR bit is returned in response to later read- and write-mode transactions, notifying the µC that the SPI may be compromised.

Note: The output data is always driven onto SDO, regardless of whether the read transaction is qualified or not.

General-Call Transactions:

A general-call transaction is an extension of the write transaction (see Figure 10).

In large LED arrays where multiple MAX20092 devices are used, execute the same command across all devices to save time. For example, the master must have the ability set all LED1's PWM duty cycle to 50% for all 26 devices with a single 32-bit transaction.

Note: Only write transactions are legal for general-call transactions.

Write Bit - R/WB = 0 (DIN31):

Only write-mode transactions are legal. Write-mode transactions are identified by R/WB = 0 in the MSB position of a 32-bit data frame.

General-Call Command - CMD[1:0] = 10 (DIN[30:29]):

General-call transactions are identified by CMD[1:0] = 10 in the DIN[30:29] positions of the frame.

Device ID - ID[4:0] = 00000 (DIN[28:24])

Note: After the 8th bit has been received (8th rising SCLK edge), all slave devices pull down SDO for the remainder of the transaction.

Address - A[6:0] (DIN[22:16]):

General-call transactions allow new information to be written to internal configuration registers within the device. The configuration register address to be written is indicated by A[6:0] within the data frame. In this format, up to 128 register addresses are supported (0\h thru 7F\h) for write-mode access.**Note:** DIN23 must always be 0.

Input Data - DIN[12:0] (DIN[15:3]):

The next 13 bits of data in the 32-bit data frame represent data that will be written to the requested register, or describe the internal operations to be executed.

SDI CRC Bit - CRCi[2:0] (DIN[2:0]):

General-call transactions are protected by a 3-bit cyclic redundancy check (CRC) on the SDI data frame. CRC is provided by the master on the last 3 bits on SDI.

It is calculated by applying the message DIN[31:3] (i.e., Address + Input Data) on polynomial 0x5 ($x^3 + x^1 + x^0$) with a start value of binary 000

The receiving device (slave) calculates its own CRC using the same polynomial and starting value. The slave only accepts/executes the command if its own CRC matches the last 3 bits on SDI.

Output Data - DO[31:0]

During general-call transactions, the MAX20092 pulls down SDO following the falling edge of SCSB. The first 8 bits are always 0. After the 8th bit has been received, the SDO continues to be pulled down for the remainder of the transaction.

General-Call Qualification Check (SPI_ERR):

To qualify for general-call execution, the following conditions must be met:

- SPI transaction must be exactly 32 bits in length (no CLK_ERR recorded)
- SDI data frame CRC check must pass (no CRC_ERR recorded)
- A[6:0] must select a valid write-accessible register or command (no RW_ERR recorded)

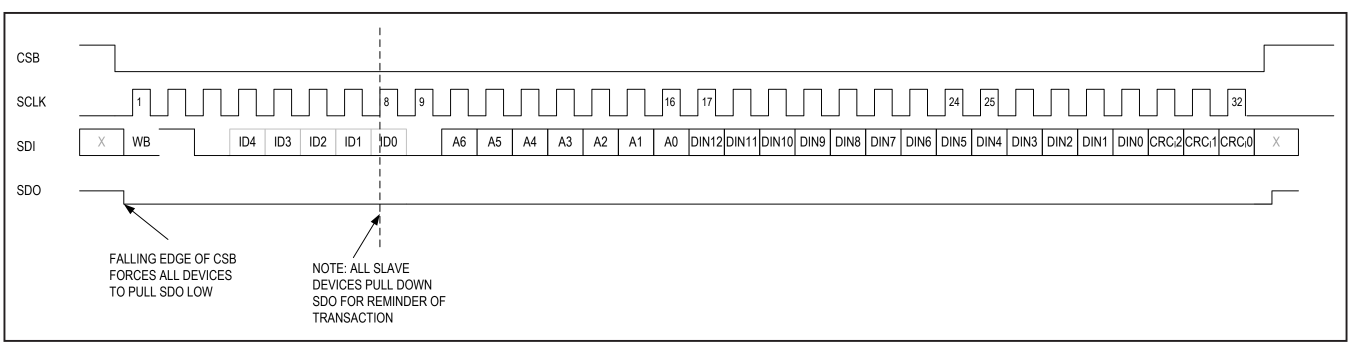


Figure 10. General-Call Transaction

If the SPI transaction is qualified, the instruction is executed, any requested internal register contents are updated, and the Internal Transaction Log updated to indicate the successful transaction.

If the SPI write transaction is not qualified, the instruction is not executed, the device's internal SPI_ERR indicator and appropriate SPI diagnostic bit are set, and the Internal Transaction Log updated to indicate the failed transaction. The SPI_ERR bit is returned in response to later read- and write-mode transactions, notifying the µC that the SPI interface may be compromised.

Cluster-Call Transactions

A cluster-call transaction is an extension of the general-call transaction (see Figure 11).

Multiple devices can be grouped into the same Cluster ID (configurable through a user-accessible register CID). This transaction allows the master to dim or brighten a cluster of LEDs with a single 32-bit transaction, but not affect LEDs assigned to a different Cluster ID.

Note: Only write transactions are legal for cluster-call transactions.

Cluster-Call Usage Example

In the cluster-call example (see Figure 12), each string is vertically connected to one MAX20092 device. There are eight MAX20092 devices shown horizontally. Cluster 1 (CID=1) is assigned to the LEDs in the left headlight and cluster 2 (CID=2) is assigned to the LEDs in the right headlight. When an oncoming car is approaching on the right, the master sends a cluster transaction to CID=2 to reduce the duty cycle of all LEDs in the right cluster.

Write Bit - R/WB = 0 (DIN31):

Only write-mode transactions are legal. Write-mode transactions are identified by R/WB = 0 in the MSB position of a 32-bit data frame.

Cluster-Call Command - CMD[1:0] = 10 (DIN[30:29]):

Cluster-call transactions are identified by CMD[1:0] = 10 in the DIN[30:29] positions of the frame.

Cluster ID - CID[4:0] (DIN[28:24]):

The master provides a 5-bit Cluster ID following the cluster-call command. Slaves assigned to the Cluster ID continue to process the remainder of the command. **Note:** After the 8th bit has been received, all slave devices assigned to a predetermined cluster pull down SDO for the remainder of the transaction. Slave devices not assigned to cluster SDO are high impedance for the remainder of the transaction.

Address - A[6:0] (DIN[22:16]):

Cluster-call transactions allow new information to be written to internal configuration registers within the device. The configuration register address to write to is indicated by A[6:0] within the data frame. In this format, up to 128 register addresses are supported (0\h thru 7F\h) for write-mode access. **Note:** DIN23 must always be 0.

Input Data - DIN[12:0] (DIN[15:3]):

The next 13 bits of data in the 32-bit data frame represent data that will be written to the requested register, or describes internal operations that must be executed.

SDI CRC Bit - CRCi[2:0] (DIN[2:0]):

Cluster-call transactions are protected by a 3-bit cyclic redundancy check (CRC) on the SDI data frame. CRC is provided by the master on the last 3 bits on SDI.

CRC is calculated by applying the DIN[31:3] message (i.e., Address + Input Data) on polynomial 0x5 ($x^3 + x^1 + x^0$) with a start value of binary 000.

The receiving device (slave) calculates its own CRC using the same polynomial and starting value. The slave only accepts/executes the command if its own CRC matches the last 3 bits on SDI.

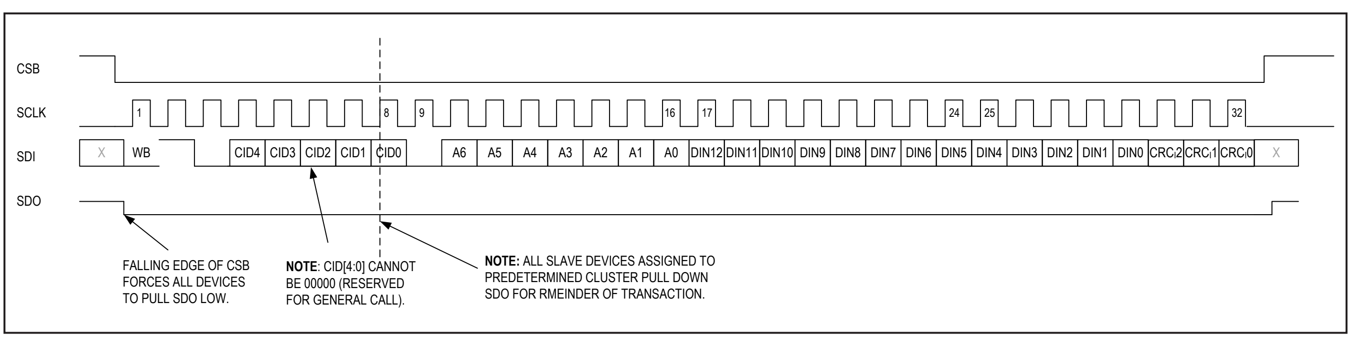


Figure 11. Cluster-Call Transaction

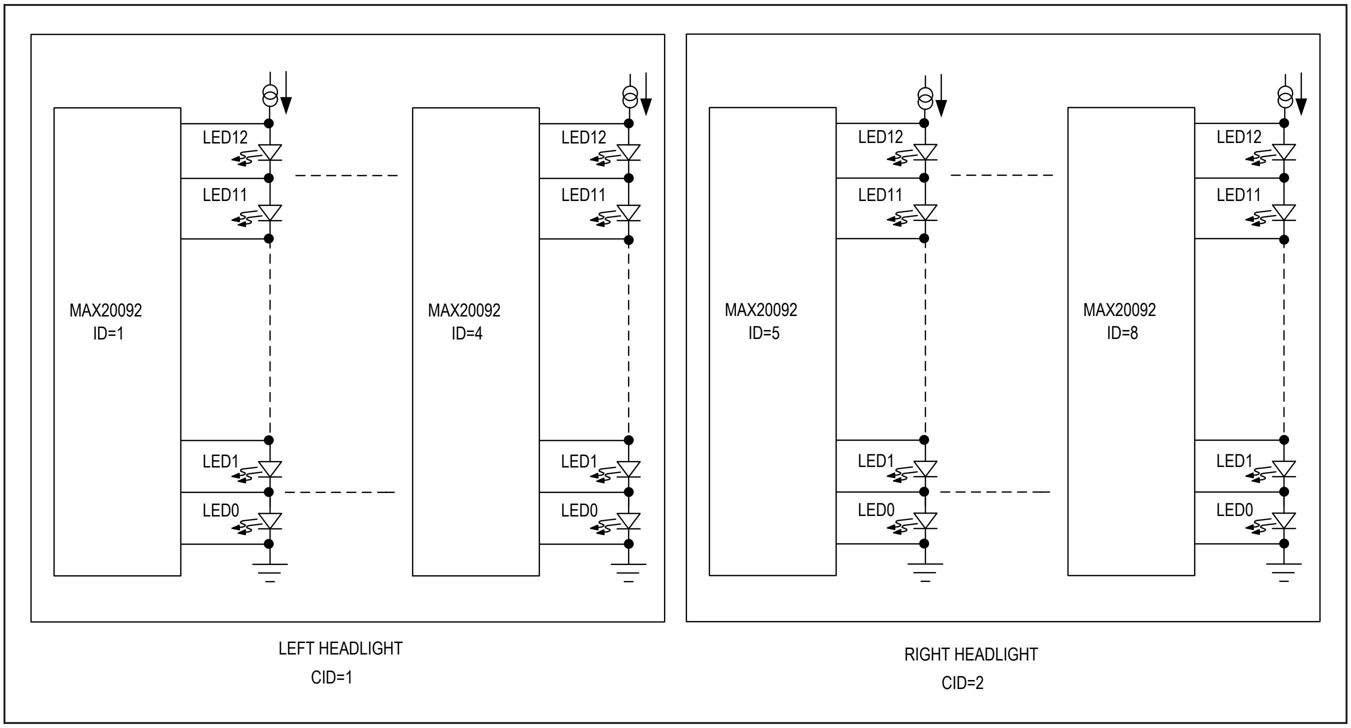


Figure 12. Cluster-Call Example

Output Data - DO[31:0]:

During cluster-call transactions, the device pulls down SDO following the falling edge of SCSB. The first 8 bits are always 0. After the 8th bit is received, all slave devices assigned to a predetermined cluster pull down SDO for the remainder of the transaction. Slave devices not assigned to a cluster do not connect SDO for the remainder of the transaction.

Cluster-Call Qualification Check (SPI_ERR):

To qualify for cluster call execution, the following conditions must be met:

- SPI transaction must be exactly 32 bits in length (no CLK_ERR recorded)
- SDI data frame CRC check must pass (no CRC_ERR recorded)
- A[6:0] must select a valid write-accessible register or command (no RW_ERR recorded)

If the SPI transaction is qualified, the instruction is executed, any requested internal register contents are updated, and the Internal Transaction Log updated to indicate the successful transaction.

If the SPI write transaction is not qualified, the instruction is not executed, the device's internal SPI_ERR indicator and appropriate SPI diagnostic bit are set, and the Internal Transaction Log updated to indicate the failed transaction. The SPI_ERR bit is returned in response to later read- and write-mode transactions, notifying the µC that the SPI may be compromised.

SPI_ERR and SPI Diagnostic Bits

If the MAX20092 is provided with an unqualified transaction, the SPI_ERR bit is set to 1, allowing the master to observe the bit and made aware of the problem during every subsequent transaction. In addition to the SPI_ERR bit, detailed SPI diagnostic bits are available to help diagnose the interface:

- CLK_ERR: Issued for read- or write-mode transactions not exactly 32 bits in length or a SPI timeout occurs
- CRC_ERR: Issued for read- or write-mode transactions that fail CRC checks
- W_ERR: Issued for write-mode transactions to invalid addresses or CMD[1:0] equals 00\b or 11\b
- R_ERR: Issued for read-mode transactions where DIN[12:0] was not 0000\h or CMD[1:0] does not equal 01\b or read-mode transactions to invalid addresses

If multiple errors occur during a single transaction, only the first error is reported, in the order of precedence given above. This helps with identification of the root cause (e.g., a malformed transaction 33 SCLK cycles in length would fail the clock check, but may also fail CRC and address checks since the data is also likely misaligned as a result. In such cases, only the CLK_ERR SPI diagnostic bit is set).All SPI diagnostic bits are clear-on-read. Once asserted, they continue to read back as high until the content is cleared by reading back the SPI configuration register with a qualified read-mode transaction. The IC keeps a cumulative list of all SPI failure types observed during failed transactions.

Note: Transactions processed after any SPI diagnostic bit is set, and remain high, are qualified and executed/accepted/logged normally, and their qualification can be determined by their inclusion in the Internal Transaction Log. Only transactions that individually fail qualification checks are shown as unqualified in the resulting Internal Transaction Log.

Internal Transaction Log

The Internal Transaction Log (LOG[23:0]) is updated on the SCSB rising edge after each SPI transaction. The contents of the log are determined by the read/write mode of the transaction and whether the transaction was qualified and executed. Details of the Internal Transaction Log content are explained in the following sections and summarized in Table 4.

Table 4. Internal Transaction Log Contents

TRANSACTION TYPE	READ MODE	WRITE MODE + INITIAL TRANSACTION	WRITE MODE + PREV TRANS = UNQUALIFIED/ REJECTEDTRANSACTION	WRITE MODE + PREV TRANS = QUALIFIED WRITE MODE (INCLUDES CLUSTER CALL + GENERAL CALL)	WRITE MODE + PREV TRANS = QUALIFIED READ MODE
LOG[23]	SPI_ERR	SPI_ERR=0	SPI_ERR=1	SPI_ERR	SPI_ERR
LOG[22]	STR[6]	0	0	A'[6]	A'[6]
LOG[21]	STR[5]	0	0	A'[5]	A'[5]
LOG[20]	STR[4]	0	0	A'[4]	A'[4]
LOG[19]	STR[3]	0	0	A'[3]	A'[3]
LOG[18]	STR[2]	0	0	A'[2]	A'[2]
LOG[17]	STR[1]	0	0	A'[1]	A'[1]
LOG[16]	STR[0]	0	0	A'[0]	A'[0]
LOG[15]	D[12]	REV_ID[4]	0	DIN'[12]	STWR[12]
LOG[14]	D[11]	REV_ID[3]	0	DIN'[11]	STWR[11]
LOG[13]	D[10]	REV_ID[2]	0	DIN'[10]	STWR[10]
LOG[12]	D[9]	REV_ID[1]	0	DIN'[9]	STWR[9]
LOG[11]	D[8]	REV_ID[0]	CLK_ERR	DIN'[8]	STWR[8]
LOG[10]	D[7]	RGRADE[2]	CRC_ERR	DIN'[7]	STWR[7]
LOG[9]	D[6]	RGRADE[1]	W_ERR	DIN'[6]	STWR[6]
LOG[8]	D[5]	RGRADE[0]	R_ERR	DIN'[5]	STWR[5]
LOG[7]	D[4]	1	0	DIN'[4]	STWR[4]
LOG[6]	D[3]	0	0	DIN'[3]	STWR[3]
LOG[5]	D[2]	0	0	DIN'[2]	STWR[2]
LOG[4]	D[1]	0	0	DIN'[1]	STWR[1]
LOG[3]	D[0]	1	0	DIN'[0]	STWR[0]
LOG[2]	CRCo[2]	CRCo[2]	CRCo[2]	CRCo[2]	CRCo[2]
LOG[1]	CRCo[1]	CRCo[1]	CRCo[1]	CRCo[1]	CRCo[1]
LOG[0]	CRCo[0]	CRCo[0]	CRCo[0]	CRCo[0]	CRCo[0]

Read-Mode Transaction Log

If the completed transaction is qualified in read mode, the device will provide the current SPI_ERR status, the current device 7-bit status STR, 13-bit requested register data, and a 3-bit CRCo as LOG[23:0]. This allows the µC to frequently check the SPI integrity, response to the last transaction, and device status with minimal communication overhead in standard connections during every write-mode command issued. **Note:** The CRC and status information provided is fetched during the subsequent write-mode transaction (rather than being latched at the time of read-mode execution), providing the most current device information available.

STR[0] = CP_RDY_N

STR[1] = STAT_OPEN_LED[0] or STAT_OPEN_TRACE[0] or STAT_SHORT[0] or
STAT_OPEN_LED[1] or STAT_OPEN_TRACE[1] or STAT_SHORT[1] or
STAT_OPEN_LED[2] or STAT_OPEN_TRACE[2] or STAT_SHORT[2]

STR[2] = STAT_OPEN_LED[3] or STAT_OPEN_TRACE[3] or STAT_SHORT[3] or
STAT_OPEN_LED[4] or STAT_OPEN_TRACE[4] or STAT_SHORT[4] or
STAT_OPEN_LED[5] or STAT_OPEN_TRACE[5] or STAT_SHORT[5]

STR[3] = STAT_OPEN_LED[6] or STAT_OPEN_TRACE[6] or STAT_SHORT[6] or
STAT_OPEN_LED[7] or STAT_OPEN_TRACE[7] or STAT_SHORT[7] or
STAT_OPEN_LED[8] or STAT_OPEN_TRACE[8] or STAT_SHORT[8]

STR[4] = STAT_OPEN_LED[9] or STAT_OPEN_TRACE[9] or STAT_SHORT[9] or
STAT_OPEN_LED[10] or STAT_OPEN_TRACE[10] or STAT_SHORT[10] or
STAT_OPEN_LED[11] or STAT_OPEN_TRACE[11] or STAT_SHORT[11]

STR[5] = TH_SHDN or TH_WARN

STR[6] = 0

CRCo is calculated by applying the message LOG[23:3] on polynomial 0x5 ($x^3 + x^1 + x^0$) with a start value of binary 000.

Write Mode + Initial Transaction Log

If there was no previous transaction due to a power-cycling event, the device returns the current SPI_ERR status = 0 as LOG[23], zeros in the address space, and a calculated 3-bit CRC bit as LOG[2:0]. The data bits return the device REV_ID[4:0], along with 11\h, allowing the µC to confirm integrity of the SDO data path and perform a CRC check on the initial SPI transaction following a POR event. CRCo is calculated by applying the message LOG[23:3] on polynomial 0x5 ($x^3 + x^1 + x^0$) with a start value of binary 000.

Write Mode + Previous Transaction = Unqualified Transaction Log

If the previous transaction was unqualified and rejected, the device stores the current SPI_ERR status = 1 as LOG[23], a calculated 3-bit CRC as LOG[2:0]; the current/cumulative SPI diagnostic error and hardware reset status are returned as LOG[11:8] and all remaining LOG bits are set to zero. This allows the µC to be aware of the transaction failure during the following transaction. CRCo is calculated by applying the message LOG[23:3] on polynomial 0x5 ($x^3 + x^1 + x^0$) with a start value of binary 000.

Write Mode + Previous Transaction = Qualified Write Transaction Log

If the completed transaction is qualified/executed in write mode upon execution, the device will store the current SPI_ERR status, the executed/previous A[6:0] and DIN[12:0] content as LOG[23:0]. This allows the μC to frequently check the SPI integrity and response to the last transaction with minimal communication overhead during every write-mode command issued. **Note:** This is the previously executed transaction data, not the internal content of any registers modified as a result of the transaction; use a read-mode transaction if an explicit verification of internal register content is desired. **Note:** Qualified general call and cluster-call commands are also considered qualified write transactions for the purposes of the transaction log.

CRCo is calculated by applying the message LOG[23:3] on polynomial 0x5 ($x^3+ x^1+ x^0$) with a start value of binary 000.

Write Mode + Previous Transaction = Qualified Read Transaction Log

If the completed transaction is qualified/executed in write mode, upon execution, the device will store the current SPI_ERR status, the executed/previous A[6:0], current device status (up to 13 bits) STR, and 3-bit CRCo content as LOG[23:0]. This allows the μC to frequently check SPI interface integrity and response to the last transaction with minimal communication overhead during every write mode command issued. Note this is the previously executed transaction data, not the internal content of any registers modified as a result of the transaction – use a read mode transaction if an explicit verification of internal register content is desired.

STWR[0] = STAT_OPEN_LED[0] or STAT_OPEN_TRACE[0] or STAT_SHORT[0]

STWR[1] = STAT_OPEN_LED[1] or STAT_OPEN_TRACE[1] or STAT_SHORT[1]

STWR[2] = STAT_OPEN_LED[2] or STAT_OPEN_TRACE[2] or STAT_SHORT[2]

STWR[3] = STAT_OPEN_LED[3] or STAT_OPEN_TRACE[3] or STAT_SHORT[3]

STWR[4] = STAT_OPEN_LED[4] or STAT_OPEN_TRACE[4] or STAT_SHORT[4]

STWR[5] = STAT_OPEN_LED[5] or STAT_OPEN_TRACE[5] or STAT_SHORT[5]

STWR[6] = STAT_OPEN_LED[6] or STAT_OPEN_TRACE[6] or STAT_SHORT[6]

STWR[7] = STAT_OPEN_LED[7] or STAT_OPEN_TRACE[7] or STAT_SHORT[7]

STWR[8] = STAT_OPEN_LED[8] or STAT_OPEN_TRACE[8] or STAT_SHORT[8]

STWR[9] = STAT_OPEN_LED[9] or STAT_OPEN_TRACE[9] or STAT_SHORT[9]

STWR[10] = STAT_OPEN_LED[10] or STAT_OPEN_TRACE[10] or STAT_SHORT[12]

STWR[11] = STAT_OPEN_LED[11] or STAT_OPEN_TRACE[11] or STAT_SHORT[11]

STWR[12] = TH_SHDN or TH_WARN or CP_RDY_N

CRCo is calculated by applying the message LOG[23:3] on polynomial 0x5 ($x^3+ x^1+ x^0$) with a start value of binary 000.

Register Map

ADDRESS	NAME	MSB							LSB
USER COMMANDS									
0x00	NO_OP[15:8]				REV_ID[4:0]				
	NO_OP[7:0]	RGRADE[2:0]			SDO_TEST[4:0]				
0x01	SW_GO[15:8]				–	–	–	–	–
	SW_GO[7:0]	–	–	–	–	–	–	–	SW_GO_EN
0x02	CNFG_SPI[15:8]				CID[4:0]				
	CNFG_SPI[7:0]	CLK_ERR	CRC_ERR	W_ERR	R_ERR	–	SFT_SCSB	SFT_CLK	SFT_SDI
0x03	CNFG_GEN[15:8]				–	–	–	–	VTH
	CNFG_GEN[7:0]	–	LED_SLEW[2:0]			DIV[1:0]		PWM_CLK_SEL[1:0]	
0x04	CNFG_MSK[15:8]				–	–	–	–	–
	CNFG_MSK[7:0]	–	MSK_SPI_ERR	MSK_OPEN_TRACE	MSK_OPEN_LED	MSK_SHORT_LED	MSK_CP_RDY_N	TH_SHDN_ACT	MSK_TH_WARN
0x05	STAT_GEN[15:8]				–	–	–	CONFIG_NOT_DONE	RGRADE_NOT_READY
	STAT_GEN[7:0]	EXT_CLK_ERR	SPI_ERR	OPEN_TRACE	OPEN_LED	SHORT_LED	CP_RDY_N	TH_SHDN	TH_WARN
0x06	STAT_OPEN_LED[15:8]				–	OPEN_LED_STAT[11:8]			
	STAT_OPEN_LED[7:0]	OPEN_LED_STAT[7:0]							
0x07	STAT_OPEN_TRACE[15:8]				–	OPEN_TRACE_STAT[11:8]			
	STAT_OPEN_TRACE[7:0]	OPEN_TRACE_STAT[7:0]							
0x08	OPEN_OVRD[15:8]				–	OPEN_LED_OVR[11:8]			
	OPEN_OVRD[7:0]	OPEN_LED_OVR[7:0]							
0x09	STAT_SHORT_LED[15:8]				–	SHORT_LED_STAT[11:8]			
	STAT_SHORT_LED[7:0]	SHORT_LED_STAT[7:0]							
0x0A	CNFG_GROUPA[15:8]				–	GROUPA_SEL[11:8]			
	CNFG_GROUPA[7:0]	GROUPA_SEL[7:0]							
0x0B	CNFG_GROUPB[15:8]				–	GROUPB_SEL[11:8]			
	CNFG_GROUPB[7:0]	GROUPB_SEL[7:0]							
0x0C	CNFG_GROUPC[15:8]				–	GROUPC_SEL[11:8]			
	CNFG_GROUPC[7:0]	GROUPC_SEL[7:0]							
0x0D	CNFG_GROUPD[15:8]				–	GROUPD_SEL[11:8]			
	CNFG_GROUPD[7:0]	GROUPD_SEL[7:0]							
0x0E	CNFG_MSK_LED[15:8]				–	CNFG_MSK_LED[11:8]			
	CNFG_MSK_LED[7:0]	CNFG_MSK_LED[7:0]							

Register Map (continued)

ADDRESS	NAME	MSB							LSB
USER COMMANDS									
0x10	PSFT_GRP[15:8]				–	–	–	–	–
	PSFT_GRP[7:0]	PSFT_GROUP[3:0]				PSFT[3:0]			
0x11	PSFT_2_1_0[15:8]				–	PSFT_2[3:0]			
	PSFT_2_1_0[7:0]	PSFT_1[3:0]				PSFT_0[3:0]			
0x12	PSFT_5_4_3[15:8]				–	PSFT_5[3:0]			
	PSFT_5_4_3[7:0]	PSFT_4[3:0]				PSFT_3[3:0]			
0x13	PSFT_8_7_6[15:8]				–	PSFT_8[3:0]			
	PSFT_8_7_6[7:0]	PSFT_7[3:0]				PSFT_6[3:0]			
0x14	PSFT_11_10_9[15:8]				–	PSFT_11[3:0]			
	PSFT_11_10_9[7:0]	PSFT_10[3:0]				PSFT_9[3:0]			
0x18	TDIM_GRP[15:8]				–	–	–	–	–
	TDIM_GRP[7:0]	TDIM_GROUP[3:0]				–	TDIM[2:0]		
0x19	TDIM_2_1_0[15:8]				–	–	TDIM_2[2:0]		
	TDIM_2_1_0[7:0]	–	TDIM_1[2:0]			–	TDIM_0[2:0]		
0x1A	TDIM_5_4_3[15:8]				–	–	TDIM_5[2:0]		
	TDIM_5_4_3[7:0]	–	TDIM_4[2:0]			–	TDIM_3[2:0]		
0x1B	TDIM_8_7_6[15:8]				–	–	TDIM_8[2:0]		
	TDIM_8_7_6[7:0]	–	TDIM_7[2:0]			–	TDIM_6[2:0]		
0x1C	TDIM_11_10_9[15:8]				–	–	TDIM_11[2:0]		
	TDIM_11_10_9[7:0]	–	TDIM_10[2:0]			–	TDIM_9[2:0]		
0x20	PWM_GRPA_DUTY[15:8]				FADE_A	DUTY_A[11:8]			
	PWM_GRPA_DUTY[7:0]	DUTY_A[7:0]							
0x21	PWM_GRPB_DUTY[15:8]				FADE_B	DUTY_B[11:8]			
	PWM_GRPB_DUTY[7:0]	DUTY_B[7:0]							
0x22	PWM_GRPC_DUTY[15:8]				FADE_C	DUTY_C[11:8]			
	PWM_GRPC_DUTY[7:0]	DUTY_C[7:0]							
0x23	PWM_GRPD_DUTY[15:8]				FADE_D	DUTY_D[11:8]			
	PWM_GRPD_DUTY[7:0]	DUTY_D[7:0]							
0x24	PWM0[15:8]				FADE_0	DUTY_0[11:8]			
	PWM0[7:0]	DUTY_0[7:0]							
0x25	PWM1[15:8]				FADE_1	DUTY_1[11:8]			
	PWM1[7:0]	DUTY_1[7:0]							
0x26	PWM2[15:8]				FADE_2	DUTY_2[11:8]			
	PWM2[7:0]	DUTY_2[7:0]							
0x27	PWM3[15:8]				FADE_3	DUTY_3[11:8]			
	PWM3[7:0]	DUTY_3[7:0]							
0x28	PWM4[15:8]				FADE_4	DUTY_4[11:8]			
	PWM4[7:0]	DUTY_4[7:0]							

Register Map (continued)

ADDRESS	NAME	MSB							LSB
USER COMMANDS									
0x29	PWM5[15:8]				FADE_5	DUTY_5[11:8]			
	PWM5[7:0]	DUTY_5[7:0]							
0x2A	PWM6[15:8]				FADE_6	DUTY_6[11:8]			
	PWM6[7:0]	DUTY_6[7:0]							
0x2B	PWM7[15:8]				FADE_7	DUTY_7[11:8]			
	PWM7[7:0]	DUTY_7[7:0]							
0x2C	PWM8[15:8]				FADE_8	DUTY_8[11:8]			
	PWM8[7:0]	DUTY_8[7:0]							
0x2D	PWM9[15:8]				FADE_9	DUTY_9[11:8]			
	PWM9[7:0]	DUTY_9[7:0]							
0x2E	PWM10[15:8]				FADE_10	DUTY_10[11:8]			
	PWM10[7:0]	DUTY_10[7:0]							
0x2F	PWM11[15:8]				FADE_11	DUTY_11[11:8]			
	PWM11[7:0]	DUTY_11[7:0]							

Register Details

NO_OP = 0x00

NO_OP is a read-only register that reads the content of RGRADE, revision ID, and test pattern.

BIT				12	11	10	9	8
Field				REV_ID[4:0]				
Reset				0x01				
Access Type				Read Only				
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	RGRADE[2:0]			SDO_TEST[4:0]				
Reset	0x0			0x11				
Access Type	Read Only			Read Only				

BITFIELD	BITS	DESCRIPTION	DECODE
REV_ID	12:8	Revision Information: Reads back 5-bit hardware revision ID.	
RGRADE	7:5	Measured resistor connected to RGRADE pin. For proper operation, 1% resistor tolerance is required.	0b000: 0 - Short to GND 0b001: 10.7kΩ 0b010: 14.7kΩ 0b011: 20.0kΩ 0b100: 28.0kΩ 0b101: 38.3kΩ 0b110: 53.6kΩ 0b111: 73.2kΩ
SDO_TEST	4:0	Test Pattern: 11\h is always returned in this location for interface checking.	

SW_GO = 0x01

SW_GO is a read/write register that enables the PWM signals.

BIT				12	11	10	9	8
Field				–	–	–	–	–
Reset				–	–	–	–	–
Access Type				–	–	–	–	–
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	–	–	–	–	–	–	–	SW_GO_EN
Reset	–	–	–	–	–	–	–	0x0
Access Type	–	–	–	–	–	–	–	Write, Read

BITFIELD	BITS	DESCRIPTION	DECODE
SW_GO_EN	0	A software sync can occur by enabling SW_GO_EN with a general call command. If SW_GO_EN = 0, all LED switches are closed and all PWM counters reset to 0. If SW_GO_EN = 1, all LED switches operate according to their programmed values and all PWM counters start counting from 0. Recommendation: SW_GO_EN command should be issued after CP_RDY_N transitions low, ensuring CLK is present and CPP voltage is valid.	0x0: All LED switches are closed, and all PWM counters are reset to 0. 0x1: All LED switches operate according to their programmed values, and all PWM counters start counting from 0.

CNFG_SPI = 0x02

CNFG_SPI is a read/write access register that controls how the SPI is configured: the cluster ID assignment and whether the internal safety terminations are engaged.

In read mode, four interface status bits are added. The SPI error indicator bits (_ERR) show what type(s) of SPI transaction errors have occurred for the MAX20092 since CNFG_SPI was last read. Once read back, their status is returned to zero (clear-on-read). SPI_ERR is the combination of the three error indicator bits:

SPI_ERR = (CLK_ERR or PAR_ERR or RW_ERROR)

BIT					12	11	10	9	8
Field					CID[4:0]				
Reset					0x01				
Access Type					Write, Read				
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
Field	CLK_ERR	CRC_ERR	W_ERR	R_ERR	–	SFT_SCSB	SFT_CLK	SFT_SDI	
Reset	0b0	0b0	0b0	0b0	–	0b1	0b1	0b1	
Access Type	Read Clears All	Read Clears All	Read Clears All	Read Clears All	–	Write, Read	Write, Read	Write, Read	

BITFIELD	BITS	DESCRIPTION	DECODE
CID	12:8	Cluster Identification: During a cluster call transaction, the SPI accepts the transaction if the received CID[4:0] matches the contents of this register.	
CLK_ERR	7	SPI Clock Error Indicator (SPI_ERR Term: Read only, clear-on-read).	0x0: Normal operation. 0x1: Clock Error: At least one SPI transaction, because clock count does not equal 32.
CRC_ERR	6	CRC Error Indicator (SPI_ERR Term: Read only, clear-on-read)	0x0: Normal operation. 0x1: CRC Error: At least one SPI transaction rejected due to a failed CRC check.
W_ERR	5	Write Error Indicator (SPI_ERR Term: Read only, clear-on-read)	0x0: Normal operation. 0x1: Write Error: At least one SPI transaction rejected for writing to an unsupported address, or reading with DIN[12:0] ≠ 0000\h).
R_ERR	4	Read Error Indicator (SPI_ERR Term: Read only, clear-on-read)	
SFT_SCSB	2	SCSB Safety Pullup Enable	0x0: Pullup disabled. 0x1: Pullup enabled (100kΩ connection to V_{DDIO}, default).
SFT_CLK	1	SCLK Safety Pulldown Enable	0x0: Pulldown disabled. 0x1: Pulldown enabled (100kΩ connection to GND, default).
SFT_SDI	0	SDI Safety Pulldown Enable	0x0: Pulldown disabled. 0x1: Pulldown enabled (100kΩ connection to GND, default).

CNFG_GEN = 0x03

CNFG_GEN is a read/write access register that controls the dimming clock divider ratio, the slew rate of the LED switches, the threshold used for the short-LED fault-detection function, and the functionality of the CLK pin.

BIT				12	11	10	9	8
Field				–	–	–	–	VTH
Reset				–	–	–	–	0b0
Access Type				–	–	–	–	Write, Read
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	–	LED_SLEW[2:0]			DIV[1:0]		PWM_CLK_SEL[1:0]	
Reset	–	0x0			0x0		0x0	
Access Type	–	Write, Read			Write, Read		Write, Read	

BITFIELD	BITS	DESCRIPTION	DECODE
VTH	8	Selects LED Open-Fault and LED Short-Fault Thresholds: This bit should be programmed once prior to setting SW_GO_EN high.	0x0: V_{OTH} = 9V (typ), V_{STH} = 1V (typ) 0x1: V_{OTH} = 9V (typ) V_{STH} = 4.4V (typ)
LED_SLEW	6:4	Slew Control for the Internal LED Gate Driver: Updating the LED gate-driver slews can take time, up to a full PWM dimming cycle; therefore, consecutive writes to LED_SLEW should be at least 1 dimming cycle apart in time. In most cases, this register is set only once, prior to setting SW_GO_EN.	µΩµµ
DIV	3:2	PWM Dimming-Frequency Select	0x0: f_{OSC}/4096 0x0: External clock frequency divided by 4096. 0x1: f_{OSC}/8192 0x1: External clock frequency divided by 8192. 0x2: f_{OSC}/16,384 0x2: External clock frequency divided by 16,384. 0x3: f_{OSC}/32,768 0x3: External clock frequency divided by 32,768
PWM_CLK_SEL	1:0	Determines internal/external PWM clock and direction of CLK pin.	0x0: Internal OSC, CLK pin disabled (default). 0x1: Internal OSC, CLK pin output. 0x2: External OSC, CLK pin input. 0x3: External OSC, CLK pin input.

CNFG_MSK = 0x04

CNFG_MSK is a read/write access register that controls the masking of fault conditions from the FAULTB pin.

BIT				12	11	10	9	8
Field				–	–	–	–	–
Reset				–	–	–	–	–
Access Type				–	–	–	–	–
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	–	MSK_SPI_ERR	MSK_OPEN_TRACE	MSK_OPEN_LED	MSK_SHORT_LED	MSK_CP_RDY_N	TH_SHDN_ACT	MSK_TH_WARN
Reset	–	0b0	0b0	0b0	0b0	0b0	0b0	0b0
Access Type	–	Write, Read	Write, Read	Write, Read	Write, Read	Write, Read	Write, Read	Write, Read

BITFIELD	BITS	DESCRIPTION	DECODE
MSK_SPI_ERR	6	Masks SPI_ERR to FAULTB.	0x0: SPI_ERR being set high asserts the FAULTB pin. 0x1: SPI_ERR bit does not assert the FAULTB pin.
MSK_OPEN_TRACE	5	Masks all open-trace detections to FAULTB.	0x0: Any OPEN_TRACE__ detects assert the FAULTB pin. 0x1: Any OPEN_TRACE__ detects do not assert the FAULTB pin.
MSK_OPEN_LED	4	Masks all open-LED detections to FAULTB.	0x0: Any OPEN_LED__ detects assert the FAULTB pin. 0x1: Any OPEN_LED__ detects do not assert the FAULTB pin.
MSK_SHORT_LED	3	Masks all STAT_SHORT_LED detections to FAULTB.	0x0: Any STAT_SHORT_LED bits set high assert the FAULTB pin. 0x1: Any STAT_SHORT_LED bits set high do not assert the FAULTB pin.
MSK_CP_RDY_N	2	Mask CP_RDY_N to FAULTB.	0x0: CP_RDY_N asserts the FAULTB pin. 0x1: CP_RDY_N does not assert the FAULTB pin.
TH_SHDN_ACT	1	Thermal-Shutdown Action: This bit selects whether to open or close the LED switches when a TH_SHDN is high.	0x0: Closes all LED switches. 0x1: Opens all LED switches.
MSK_TH_WARN	0	Mask-Thermal Warning to FAULTB.	0x0: TH_WARN asserts the FAULTB pin. 0x1: TH_WARN does not assert the FAULTB pin.

STAT_GEN = 0x05

STAT_GEN is a read-only access register that provides general operations and warnings. FAULTB is asserted whenever any of these bits is high, unless the corresponding MASK bit is set.

BIT				12	11	10	9	8
Field				–	–	–	CONFIG_ NOT_DONE	RGRADE_ NOT_READY
Reset				–	–	–		
Access Type				–	–	–	Read Only	Read Only
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	EXT_CLK_ERR	SPI_ERR	OPEN_TRACE	OPEN_LED	SHORT_LED	CP_RDY_N	TH_SHDN	TH_WARN
Reset							0b0	0b0
Access Type	Read Only	Read Only	Read Only	Read Only	Read Only	Read Only	Read Clears All	Read Clears All

BITFIELD	BITS	DESCRIPTION	DECODE
CONFIG_ NOT_DONE	9	This bit indicates that the SPI interface has not completed programming the LED switch configuration, triggered by writing CNFG_GEN. The master should ensure this bit is low before attempting to program CNFG_GEN. This bit does not assert the FAULTB pin.	0x0: Configuration complete; ready for new CNFG_GEN command. 0x1: Configuration not complete.
RGRADE_ NOT_READY	8	This signal indicates that the RGRADE read operation is not complete. When the signal goes low, the read is complete and RGRADE[2:0] in register 0x0 is valid. This signal does not assert the FAULTB pin.	0x0: RGRADE[2:0] is valid. 0x1: RGRADE[2:0] is not valid.
EXT_CLK_ ERR	7	EXT_CLK_ERR is asserted when the part is configured to use an external clock (PWM_CLK_SEL = x2 or x3) and the external clock is slower than the minimum operating frequency.	
SPI_ERR	6	SPI_ERR is asserted if any of the error bits in SNFG_SPI are set.	0x0: SPI is operating normally. 0x1: At least 1 of R_ERR, W_ERR, CRC_ ERR, or CLK_ERR has been asserted.
OPEN_ TRACE	5	OPEN_TRACE is asserted if any OPEN_TRACE_STAT bit is high.	0x0: All LED drivers operating normally. 0x1: At least one LED driver has open-trace detected.
OPEN_LED	4	OPEN_LED is asserted if any OPEN_LED_STAT bit is high.	0x0: All LED drivers operating normally. 0x1: At least one LED driver has open detected.
SHORT_LED	3	SHORT_LED is asserted if any SHORT_LED_STAT bit is high.	0x0: All LED drivers operating normally. 0x1: At least one LED driver has short detected.
CP_RDY_N	2	CP_RDY_N is a read-only bit that indicates the charge-pump voltage is below the operating threshold.	0x0: CP operating normally. 0x1: CP is below VCPP_OK threshold
TH_SHDN	1	Thermal Shutdown. Latched, clear on read if condition has been resolved.	
TH_WARN	0	Thermal Warning. Latched, clear-on-read if condition has been resolved.	0x0: Normal operation 0x1: Device has exceeded the thermal-warning threshold.

STAT_OPEN_LED = 0x06

STAT_OPEN is a read-only access register that provides open-detect information on the 12 LED output drivers.

BIT				12	11	10	9	8
Field				–	OPEN_LED_STAT[11:8]			
Reset				–				
Access Type				–	Read Clears All			
BIT	7	6	5	4	3	2	1	0
Field	OPEN_LED_STAT[7:0]							
Reset								
Access Type	Read Clears All							

BITFIELD	BITS	DESCRIPTION	DECODE
OPEN_LED_STAT	11:0	Indicates that an open-LED condition has been detected.	0x0: Normal 0x1: Open LED

STAT_OPEN_TRACE = 0x07

STAT_OPEN_TRACE is a read-only register that provides open-trace fault information on the 12 LED output drivers.

BIT				12	11	10	9	8
Field				–	OPEN_TRACE_STAT[11:8]			
Reset				–				
Access Type				–	Read Clears All			
BIT	7	6	5	4	3	2	1	0
Field	OPEN_TRACE_STAT[7:0]							
Reset								
Access Type	Read Clears All							

BITFIELD	BITS	DESCRIPTION	DECODE
OPEN_TRACE_STAT	11:0	Indicates an open-trace has been detected.	0x0: Normal 0x1: Open trace

OPEN_OVRD = 0x08

OPEN_OVRD is a read/write register that overrides the LED switching control signals. When this feature is disabled, the LED switch operates normally. When this feature is enabled, the LED switch is always forced to a closed position (i.e., the LED duty cycle is zero, regardless of the DUTY or TDIM settings).The intent is to allow the µP to manually force the switch to stay closed after it has determined the particular LED is permanently opened. This further suppresses FAULTB signals from the switch(es) since LED faults are only detected when the switch opens.

BIT				12	11	10	9	8
Field				–	OPEN_LED_OVR[11:8]			
Reset				–				
Access Type				–	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	OPEN_LED_OVR[7:0]							
Reset								
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
OPEN_LED_OVR	11:0	Open-LED Override: When this bit is set, the corresponding LED switch is always held closed.	0x0: Normal 0x1: LED switch is always closed.

STAT_SHORT_LED = 0x09

STAT_SHORT_LED is a read-only access register that provides short-detect information on the 12 LED output drivers.

BIT				12	11	10	9	8
Field				–	SHORT_LED_STAT[11:8]			
Reset				–				
Access Type				–	Read Clears All			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	SHORT_LED_STAT[7:0]							
Reset								
Access Type	Read Clears All							

BITFIELD	BITS	DESCRIPTION
SHORT_LED_STAT	11:0	Indicates an LED short has been detected.

CNFG_GROUPA = 0x0A

CNFG_GRPA is a read/write register that allows the user to assign particular LED drivers to a this group. LED drivers assigned to this group respond to qualified transactions on the following registers:

- PSFT_GRP (if PSFT_GROUP==0001)
- TDIM_GROUP (if TDIM_GROUP=0001)
- PWM_GRPA_DUTY

BIT				12	11	10	9	8
Field				–	GROUPA_SEL[11:8]			
Reset				–				
Access Type				–	Write, Read			
BIT	7	6	5	4	3	2	1	0
Field	GROUPA_SEL[7:0]							
Reset								
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
GROUPA_SEL	11:0	Set high if assigning a register to GroupA.	0x0: Not assigned 0x1: Assigned

CNFG_GROUPB = 0x0B

CNFG_GRPB is a read/write register that allows the user to assign particular LED drivers to a this group. LED drivers assigned to this group respond to qualified transactions on the following registers:

- PSFT_GRP (if PSFT_GROUP==00010)
- TDIM_GROUP (if TDIM_GROUP=0010)
- PWM_GRPB_DUTY

BIT				12	11	10	9	8
Field				–	GROUPB_SEL[11:8]			
Reset				–				
Access Type				–	Write, Read			
BIT	7	6	5	4	3	2	1	0
Field	GROUPB_SEL[7:0]							
Reset								
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
GROUPB_SEL	11:0	Set high if assigning a register to GroupB.	0x0: Not assigned 0x1: Assigned

CNFG_GROUPC = 0x0C

CNFG_GRPC is a read/write register that allows the user to assign particular LED drivers to a this group. LED drivers assigned to this group respond to qualified transactions on the following registers:

- PSFT_GRP (if PSFT_GROUP==0100)
- TDIM_GROUP (if TDIM_GROUP=0100)
- PWM_GRPc_DUTY

BIT				12	11	10	9	8
Field				–	GROUPC_SEL[11:8]			
Reset				–				
Access Type				–	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	GROUPC_SEL[7:0]							
Reset								
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
GROUPC_SEL	11:0	Set high if assigning a register to GroupC.	0x0: Not assigned 0x1: Assigned

CNFG_GROUPD = 0x0D

CNFG_GRPD is a read/write register that allows the user to assign particular LED drivers to a this group. LED drivers assigned to this group respond to qualified transactions on the following registers:

- PSFT_GRP (if PSFT_GROUP==1000)
- TDIM_GROUP (if TDIM_GROUP=1000)
- PWM_GRPD_DUTY

BIT				12	11	10	9	8
Field				–	GROUPD_SEL[11:8]			
Reset				–				
Access Type				–	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	GROUPD_SEL[7:0]							
Reset								
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
GROUPD_SEL	11:0	Set high if assigning a register to GroupD.	0x0: Not assigned 0x1: Assigned

CNFG_MSK_LED = 0x0E

CNFG_MSK_LED prevents LED faults from asserting the FAULTB pin. This allows the μP to instruct the part to ignore faults from a particular LED when that LED is deliberately not populated in the application.

BIT				12	11	10	9	8
Field				–	CNFG_MSK_LED[11:8]			
Reset				–				
Access Type				–	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	CNFG_MSK_LED[7:0]							
Reset								
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION
CNFG_MSK_LED	11:0	Set bit(s) high to mask OPEN_LED, SHORT_LED, and OPEN_TRACE faults from those LEDs asserting FAULTB.

PSFT_GRP = 0x10

PSFT_GRP is a read/write register that allows the user to assign the same phase shift to one or more LED drivers.

The contents of PSFT are written to the desired group specified by PSFT_GROUP.Example:If PSFT_GROUP == Group A, PSFT == 0001, and LED11, LED9, and LED6 are assigned to Group A (through CNFG_GRPA), then PSFT_11, PSFT_9, and PSFT_6 will contain 0001 after the transaction is executed.

BIT				12	11	10	9	8
Field				–	–	–	–	–
Reset				–	–	–	–	–
Access Type				–	–	–	–	–
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	PSFT_GROUP[3:0]				PSFT[3:0]			
Reset	0x1				0x0			
Access Type	Write, Read				Write, Read			

BITFIELD	BITS	DESCRIPTION	DECODE
PSFT_GROUP	7:4	Group Select: bit 0: Group A selected bit 1: Group B selected bit 2: Group C selected bit 3: Group D selected Multiple groups can be selected at a time. Note: 0000 is not a valid selection, the transaction is not executed and the 4-bit value is unchanged.	

BITFIELD	BITS	DESCRIPTION	DECODE
PSFT	3:0	Phase Select	0x0: 0º 0x1: 30º 0x2: 60º 0x3: 90º 0x4: 120º 0x5: 150º 0x6: 180º 0x7: 210º 0x8: 240º 0x9: 270º 0xA: 300º 0xB: 330º 0xC: 0º 0xD: 0º 0xE: 0º 0xF: —

PSFT_2_1_0 = 0x11

PSFT_2_1_0 is a read/write register that controls the phase shift for LED drivers 0, 1, and 2.

BIT				12	11	10	9	8
Field				–	PSFT_2[3:0]			
Reset				–	0x0			
Access Type				–	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	PSFT_1[3:0]				PSFT_0[3:0]			
Reset	0x0				0x0			
Access Type	Write, Read				Write, Read			

BITFIELD	BITS	DESCRIPTION	DECODE
PSFT_2	11:8	LED 2 Phase Select	0x0: 0º 0x1: 30º 0x2: 60º 0x3: 90º 0x4: 120º 0x5: 150º 0x6: 180º 0x7: 210º 0x8: 240º 0x9: 270º 0xA: 300º 0xB: 330º 0xC: 0º 0xD: 0º 0xE: 0º 0xF

BITFIELD	BITS	DESCRIPTION	DECODE
PSFT_1	7:4	LED 1 Phase Select	0x0: 0º 0x1: 30º 0x2: 60º 0x3: 90º 0x4: 120º 0x5: 150º 0x6: 180º 0x7: 210º 0x8: 240º 0x9: 270º 0xA: 300º 0xB: 330º 0xC: 0º 0xD: 0º 0xE: 0º 0xF: —
PSFT_0	3:0	LED 0 Phase Select	0x0: 0º 0x1: 30º 0x2: 60º 0x3: 90º 0x4: 120º 0x5: 150º 0x6: 180º 0x7: 210º 0x8: 240º 0x9: 270º 0xA: 300º 0xB: 330º 0xC: 0º 0xD: 0º 0xE: 0º 0xF: —

PSFT_5_4_3 = 0x12

PSFT_5_4_3 is a read/write register that controls the phase shift for LED drivers 3, 4, and 5.

BIT				12	11	10	9	8
Field				–	PSFT_5[3:0]			
Reset				–	0x0			
Access Type				–	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	PSFT_4[3:0]				PSFT_3[3:0]			
Reset	0x0				0x0			
Access Type	Write, Read				Write, Read			

BITFIELD	BITS	DESCRIPTION	DECODE
PSFT_5	11:8	LED 5 Phase Select	0x0: 0º 0x1: 30º 0x2: 60º 0x3: 90º 0x4: 120º 0x5: 150º 0x6: 180º 0x7: 210º 0x8: 240º 0x9: 270º 0xA: 300º 0xB: 330º 0xC: 0º 0xD: 0º 0xE: 0º 0xF: —
PSFT_4	7:4	LED 4 Phase Select	0x0: 0º 0x1: 30º 0x2: 60º 0x3: 90º 0x4: 120º 0x5: 150º 0x6: 180º 0x7: 210º 0x8: 240º 0x9: 270º 0xA: 300º 0xB: 330º 0xC: 0º 0xD: 0º 0xE: 0º 0xF: —
PSFT_3	3:0	LED 3 Phase Select	0x0: 0º 0x1: 30º 0x2: 60º 0x3: 90º 0x4: 120º 0x5: 150º 0x6: 180º 0x7: 210º 0x8: 240º 0x9: 270º 0xA: 300º 0xB: 330º 0xC: 0º 0xD: 0º 0xE: 0º 0xF: —

PSFT_8_7_6 = 0x13

PSFT_8_7_6 is a read/write register that controls the phase shift for LED drivers 6, 7, and 8.

BIT				12	11	10	9	8
Field				–	PSFT_8[3:0]			
Reset				–	0x0			
Access Type				–	Write, Read			
BIT	7	6	5	4	3	2	1	0
Field	PSFT_7[3:0]				PSFT_6[3:0]			
Reset	0x0				0x0			
Access Type	Write, Read				Write, Read			

BITFIELD	BITS	DESCRIPTION	DECODE
PSFT_8	11:8	LED 8 Phase Select	0x0: 0º 0x1: 30º 0x2: 60º 0x3: 90º 0x4: 120º 0x5: 150º 0x6: 180º 0x7: 210º 0x8: 240º 0x9: 270º 0xA: 300º 0xB: 330º 0xC: 0º 0xD: 0º 0xE: 0º 0xF: —
PSFT_7	7:4	LED 7 Phase Select	0x0: 0º 0x1: 30º 0x2: 60º 0x3: 90º 0x4: 120º 0x5: 150º 0x6: 180º 0x7: 210º 0x8: 240º 0x9: 270º 0xA: 300º 0xB: 330º 0xC: 0º 0xD: 0º 0xE: 0º 0xF: —

BITFIELD	BITS	DESCRIPTION	DECODE
PSFT_6	3:0	LED 6 Phase Select	0x0: 0º 0x1: 30º 0x2: 60º 0x3: 90º 0x4: 120º 0x5: 150º 0x6: 180º 0x7: 210º 0x8: 240º 0x9: 270º 0xA: 300º 0xB: 330º 0xC: 0º 0xD: 0º 0xE: 0º 0xF: —

PSFT_11_10_9 = 0x14

PSFT_11_10_9 is a read/write register that controls the phase shift for LED drivers 9, 10, and 11.

BIT				12	11	10	9	8
Field				–	PSFT_11[3:0]			
Reset				–	0x0			
Access Type				–	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	PSFT_10[3:0]				PSFT_9[3:0]			
Reset	0x0				0x0			
Access Type	Write, Read				Write, Read			

BITFIELD	BITS	DESCRIPTION	DECODE
PSFT_11	11:8	LED 11 Phase Select	0x0: 0º 0x1: 30º 0x2: 60º 0x3: 90º 0x4: 120º 0x5: 150º 0x6: 180º 0x7: 210º 0x8: 240º 0x9: 270º 0xA: 300º 0xB: 330º 0xC: 0º 0xD: 0º 0xE: 0º 0xF: —

BITFIELD	BITS	DESCRIPTION	DECODE
PSFT_10	7:4	LED 10 Phase Select	0x0: 0º 0x1: 30º 0x2: 60º 0x3: 90º 0x4: 120º 0x5: 150º 0x6: 180º 0x7: 210º 0x8: 240º 0x9: 270º 0xA: 300º 0xB: 330º 0xC: 0º 0xD: 0º 0xE: 0º 0xF: —
PSFT_9	3:0	LED 9 Phase Select	0x0: 0º 0x1: 30º 0x2: 60º 0x3: 90º 0x4: 120º 0x5: 150º 0x6: 180º 0x7: 210º 0x8: 240º 0x9: 270º 0xA: 300º 0xB: 330º 0xC: 0º 0xD: 0º 0xE: 0º 0xF: —

TDIM_GRP = 0x18

TDIM_GRP is a read/write register that allows the user to assign the same dimming period to one or more LED drivers. The contents of TDIM are written to the desired group specified by TDIM_GROUP.Example:If TDIM_GROUP == Group A, PSFT == 001, and LED12, LED9, and LED6 are assigned to Group A (through CNFG_GRPA), then TDIM_12, TDIM_9, and TDIM_6 will contain 001 after the transaction is executed.

BIT				12	11	10	9	8
Field				–	–	–	–	–
Reset				–	–	–	–	–
Access Type				–	–	–	–	–
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	TDIM_GROUP[3:0]				–	TDIM[2:0]		
Reset	0x1				–	0x0		
Access Type	Write, Read				–	Write, Read		

BITFIELD	BITS	DESCRIPTION	DECODE
TDIM_GROUP	7:4	Group Select: Bit 0: Group A selected Bit 1: Group B selected Bit 2: Group C selected Bit 3: Group D selected Multiple groups can be selected at a time. Note: 0000 is not a valid selection, the transaction is not executed and the 4-bit value is unchanged.	
TDIM	2:0	Dimming Period Select	0x0: Update PWM duty cycle every 1 PWM period 0x1: Update PWM duty cycle every 2 PWM periods 0x2: Update PWM duty cycle every 4 PWM periods 0x3: Update PWM duty cycle every 8 PWM periods 0x4: Update PWM duty cycle every 16 PWM periods 0x5: Update PWM duty cycle every 32 PWM periods 0x6: Update PWM duty cycle every 32 PWM periods 0x7: Update PWM duty cycle every 32 PWM periods1 PWM duty period = 4096 clock cycles

TDIM_2_1_0 = 0x19

TDIM_2_1_0 is a read/write register that controls the dimming period for LED drivers 2, 1, and 0.

BIT				12	11	10	9	8
Field				–	–	TDIM_2[2:0]		
Reset				–	–	0x0		
Access Type				–	–	Write, Read		
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	–	TDIM_1[2:0]			–	TDIM_0[2:0]		
Reset	–	0x0			–	0x0		
Access Type	–	Write, Read			–	Write, Read		

BITFIELD	BITS	DESCRIPTION	DECODE
TDIM_2	10:8	LED 2 Dimming Period Select	0x0: Update PWM duty cycle every 1 PWM period. 0x1: Update PWM duty cycle every 2 PWM periods. 0x2: Update PWM duty cycle every 4 PWM periods. 0x3: Update PWM duty cycle every 8 PWM periods. 0x4: Update PWM duty cycle every 16 PWM periods. 0x5: Update PWM duty cycle every 32 PWM periods. 0x6: Update PWM duty cycle every 32 PWM periods. 0x7: Update PWM duty cycle every 32 PWM periods. 1 PWM duty period = 4096 clock cycles
TDIM_1	6:4	LED 1 Dimming Period Select	0x0: Update PWM duty cycle every 1 PWM period. 0x1: Update PWM duty cycle every 2 PWM periods. 0x2: Update PWM duty cycle every 4 PWM periods. 0x3: Update PWM duty cycle every 8 PWM periods. 0x4: Update PWM duty cycle every 16 PWM periods. 0x5: Update PWM duty cycle every 32 PWM periods. 0x6: Update PWM duty cycle every 32 PWM periods. 0x7: Update PWM duty cycle every 32 PWM periods. 1 PWM duty period = 4096 clock cycles
TDIM_0	2:0	LED 0 Dimming Period Select	0x0: Update PWM duty cycle every 1 PWM period. 0x1: Update PWM duty cycle every 2 PWM periods. 0x2: Update PWM duty cycle every 4 PWM periods. 0x3: Update PWM duty cycle every 8 PWM periods. 0x4: Update PWM duty cycle every 16 PWM periods. 0x5: Update PWM duty cycle every 32 PWM periods. 0x6: Update PWM duty cycle every 32 PWM periods. 0x7: Update PWM duty cycle every 32 PWM periods. 1 PWM duty period = 4096 clock cycles

TDIM_5_4_3 = 0x1A

TDIM_5_4_3 is a read/write register that controls the dimming period for LED drivers 5, 4, and 3.

BIT				12	11	10	9	8
Field				–	–	TDIM_5[2:0]		
Reset				–	–	0x0		
Access Type				–	–	Write, Read		
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	–	TDIM_4[2:0]			–	TDIM_3[2:0]		
Reset	–	0x0			–	0x0		
Access Type	–	Write, Read			–	Write, Read		

BITFIELD	BITS	DESCRIPTION	DECODE
TDIM_5	10:8	LED 5 Dimming Period Select	0x0: Update PWM duty cycle every 1 PWM period. 0x1: Update PWM duty cycle every 2 PWM periods. 0x2: Update PWM duty cycle every 4 PWM periods. 0x3: Update PWM duty cycle every 8 PWM periods. 0x4: Update PWM duty cycle every 16 PWM periods. 0x5: Update PWM duty cycle every 32 PWM periods. 0x6: Update PWM duty cycle every 32 PWM periods. 0x7: Update PWM duty cycle every 32 PWM periods. 1 PWM duty period = 4096 clock cycles
TDIM_4	6:4	LED 4 Dimming Period Select	0x0: Update PWM duty cycle every 1 PWM period. 0x1: Update PWM duty cycle every 2 PWM periods. 0x2: Update PWM duty cycle every 4 PWM periods. 0x3: Update PWM duty cycle every 8 PWM periods. 0x4: Update PWM duty cycle every 16 PWM periods. 0x5: Update PWM duty cycle every 32 PWM periods. 0x6: Update PWM duty cycle every 32 PWM periods. 0x7: Update PWM duty cycle every 32 PWM periods. 1 PWM duty period = 4096 clock cycles
TDIM_3	2:0	LED 3 Dimming Period Select	0x0: Update PWM duty cycle every 1 PWM period. 0x1: Update PWM duty cycle every 2 PWM periods. 0x2: Update PWM duty cycle every 4 PWM periods. 0x3: Update PWM duty cycle every 8 PWM periods. 0x4: Update PWM duty cycle every 16 PWM periods. 0x5: Update PWM duty cycle every 32 PWM periods. 0x6: Update PWM duty cycle every 32 PWM periods. 0x7: Update PWM duty cycle every 32 PWM periods. 1 PWM duty period = 4096 clock cycles

TDIM_8_7_6 = 0x1B

TDIM_8_7_6 is a read/write register that controls the dimming period for LED drivers 8, 7, and 6.

BIT				12	11	10	9	8
Field				–	–	TDIM_8[2:0]		
Reset				–	–	0x0		
Access Type				–	–	Write, Read		
BIT	7	6	5	4	3	2	1	0
Field	–	TDIM_7[2:0]			–	TDIM_6[2:0]		
Reset	–	0x0			–	0x0		
Access Type	–	Write, Read			–	Write, Read		

BITFIELD	BITS	DESCRIPTION	DECODE
TDIM_8	10:8	LED 8 Dimming Period Select	0x0: Update PWM duty cycle every 1 PWM period. 0x1: Update PWM duty cycle every 2 PWM periods. 0x2: Update PWM duty cycle every 4 PWM periods. 0x3: Update PWM duty cycle every 8 PWM periods. 0x4: Update PWM duty cycle every 16 PWM periods. 0x5: Update PWM duty cycle every 32 PWM periods. 0x6: Update PWM duty cycle every 32 PWM periods. 0x7: Update PWM duty cycle every 32 PWM periods. 1 PWM duty period = 4096 clock cycles
TDIM_7	6:4	LED 7 Dimming Period Select	0x0: Update PWM duty cycle every 1 PWM period. 0x1: Update PWM duty cycle every 2 PWM periods. 0x2: Update PWM duty cycle every 4 PWM periods. 0x3: Update PWM duty cycle every 8 PWM periods. 0x4: Update PWM duty cycle every 16 PWM periods. 0x5: Update PWM duty cycle every 32 PWM periods. 0x6: Update PWM duty cycle every 32 PWM periods. 0x7: Update PWM duty cycle every 32 PWM periods. 1 PWM duty period = 4096 clock cycles
TDIM_6	2:0	LED 6 Dimming Period Select	0x0: Update PWM duty cycle every 1 PWM period. 0x1: Update PWM duty cycle every 2 PWM periods. 0x2: Update PWM duty cycle every 4 PWM periods. 0x3: Update PWM duty cycle every 8 PWM periods. 0x4: Update PWM duty cycle every 16 PWM periods. 0x5: Update PWM duty cycle every 32 PWM periods. 0x6: Update PWM duty cycle every 32 PWM periods. 0x7: Update PWM duty cycle every 32 PWM periods. 1 PWM duty period = 4096 clock cycles

TDIM_11_10_9 = 0x1C

TDIM_11_10_9 is a read/write register that controls the dimming period for LED drivers 11, 10, and 9.

BIT				12	11	10	9	8
Field				–	–	TDIM_11[2:0]		
Reset				–	–	0x0		
Access Type				–	–	Write, Read		
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	–	TDIM_10[2:0]			–	TDIM_9[2:0]		
Reset	–	0x0			–	0x0		
Access Type	–	Write, Read			–	Write, Read		

BITFIELD	BITS	DESCRIPTION	DECODE
TDIM_11	10:8	LED 11 Dimming Period Select	0x0: Update PWM duty cycle every 1 PWM period. 0x1: Update PWM duty cycle every 2 PWM periods. 0x2: Update PWM duty cycle every 4 PWM periods. 0x3: Update PWM duty cycle every 8 PWM periods. 0x4: Update PWM duty cycle every 16 PWM periods. 0x5: Update PWM duty cycle every 32 PWM periods. 0x6: Update PWM duty cycle every 32 PWM periods. 0x7: Update PWM duty cycle every 32 PWM periods. 1 PWM duty period = 4096 clock cycles
TDIM_10	6:4	LED 10 Dimming Period Select	0x0: Update PWM duty cycle every 1 PWM period. 0x1: Update PWM duty cycle every 2 PWM periods. 0x2: Update PWM duty cycle every 4 PWM periods. 0x3: Update PWM duty cycle every 8 PWM periods. 0x4: Update PWM duty cycle every 16 PWM periods. 0x5: Update PWM duty cycle every 32 PWM periods. 0x6: Update PWM duty cycle every 32 PWM periods. 0x7: Update PWM duty cycle every 32 PWM periods. 1 PWM duty period = 4096 clock cycles
TDIM_9	2:0	LED 9 Dimming Period Select	0x0: Update PWM duty cycle every 1 PWM period. 0x1: Update PWM duty cycle every 2 PWM periods.. 0x2: Update PWM duty cycle every 4 PWM periods. 0x3: Update PWM duty cycle every 8 PWM periods. 0x4: Update PWM duty cycle every 16 PWM periods. 0x5: Update PWM duty cycle every 32 PWM periods. 0x6: Update PWM duty cycle every 32 PWM periods. 0x7: Update PWM duty cycle every 32 PWM periods. 1 PWM duty period = 4096 clock cycles

PWM_GRPA_DUTY = 0x20

PWM_GRPA_DUTY is a read/write register that allows the user to assign the same duty cycle and enable/disable PWM dimming to one or more LED drivers.

The contents of DUTY_A are written to LEDs assigned to Group A.

Example: If DUTY_A == 0x0AA and LED11, LED8, and LED5 are assigned to Group A (through CNFG_GRPA), then DUTY_11, DUTY_8, and DUTY_5 will contain 0x0AA after the transaction is executed.

BIT				12	11	10	9	8
Field				FADE_A	DUTY_A[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	DUTY_A[7:0]							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_A	12	Group A PWM Dimming Enable	0x0: Disabled 0x1: Enabled
DUTY_A	11:0	Group A Duty-Cycle Selection: 0x000 = Off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

PWM_GRPB_DUTY = 0x21

PWM_GRPB_DUTY is a read/write register that allows the user to assign the same duty cycle and enable/disable PWM dimming to one or more LED drivers.

The contents of DUTY_B are written to LEDs assigned to Group B.

Example: If DUTY_B == 0x0AA and LED11, LED9, and LED6 are assigned to Group B (through CNFG_GRPB), then DUTY_11, DUTY_9, and DUTY_6 will contain 0x0AA after the transaction is executed.

BIT				12	11	10	9	8
Field				FADE_B	DUTY_B[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	DUTY_B[7:0]							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_B	12	Group B PWM Dimming Enable	0x0: Enabled 0x1: Disabled
DUTY_B	11:0	Group B Duty-Cycle Selection: 0x000 = Off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

PWM_GRPC_DUTY = 0x22

PWM_GRPC_DUTY is a read/write register that allows the user to assign the same duty cycle and enable/disable PWM dimming to one or more LED drivers.

The contents of DUTY_C are written to LEDs assigned to Group B.Example:

If DUTY_C == 0x0AA and LED11, LED9, and LED6 are assigned to Group C (through CNFG_GRPC), then DUTY_11, DUTY_9, and DUTY_6 will contain 0x0AA after the transaction is executed.

BIT				12	11	10	9	8
Field				FADE_C	DUTY_C[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	DUTY_C[7:0]							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_C	12	Group C PWM Dimming Enable	0x0: Enabled 0x1: Disabled
DUTY_C	11:0	Group C Duty-Cycle Selection: 0x000 = Off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

PWM_GRPD_DUTY = 0x23

PWM_GRPD_DUTY is a read/write register that allows the user to assign the same duty cycle and enable/disable PWM dimming to one or more LED drivers.

The contents of DUTY_D are written to LEDs assigned to Group D.Example: If DUTY_D == 0x0AA and LED11, LED9, and LED6 are assigned to Group B (through CNFG_GRPD), then DUTY_11, DUTY_9, and DUTY_6 will contain 0x0AA after the transaction is executed.

BIT				12	11	10	9	8
Field				FADE_D	DUTY_D[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	DUTY_D[7:0]							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_D	12	Group D PWM Dimming Enable	0x0: Enabled 0x1: Disabled
DUTY_D	11:0	Group D Duty-Cycle Selection: 0x000 = Off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

PWM0 = 0x24

PWM0 is a read/write register that configures the LED0 duty cycle and enables/disables PWM dimming.

BIT				12	11	10	9	8
Field				FADE_0	DUTY_0[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	DUTY_0[7:0]							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_0	12	LED0 PWM Dimming Enable	0x0: Enabled 0x1: Disabled
DUTY_0	11:0	LED0 Duty-Cycle Selection: 0x000 = Off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

PWM1 = 0x25

PWM1 is a read/write register that configures the LED1 duty cycle and enables/disables PWM dimming.

BIT				12	11	10	9	8
Field				FADE_1	DUTY_1[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	DUTY_1[7:0]							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_1	12	LED1 PWM Dimming Enable	0x0: Enabled 0x1: Disabled
DUTY_1	11:0	LED1 Duty-Cycle Selection: 0x000 = Off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

PWM2 = 0x26

PWM2 is a read/write register that configures the LED2 duty cycle and enables/disables PWM dimming.

BIT				12	11	10	9	8
Field				FADE_2	DUTY_2[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	DUTY_2[7:0]							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_2	12	LED2 PWM Dimming Enable	0x0: Enabled 0x1: Disabled
DUTY_2	11:0	LED2 Duty-Cycle Selection: 0x000 = Off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

PWM3 = 0x27

PWM3 is a read/write register that configures the LED3 duty cycle and enables/disables PWM dimming.

BIT				12	11	10	9	8
Field				FADE_3	DUTY_3[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	DUTY_3[7:0]							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_3	12	LED3 PWM Dimming Enable	0x0: Enabled 0x1: Disabled
DUTY_3	11:0	LED3 Duty-Cycle Selection: 0x000 = Off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

PWM4 = 0x28

PWM4 is a read/write register that configures the LED4 duty cycle and enables/disables PWM dimming.

BIT				12	11	10	9	8
Field				FADE_4	DUTY_4[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FIELD	**DUTY_4[7:0]**							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_4	12	LED4 PWM Dimming Enable	0x0: Enabled 0x1: Disabled
DUTY_4	11:0	LED4 Duty-Cycle Selection: 0x000= off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

PWM5 = 0x29

PWM5 is a read/write register that configures the LED5 duty cycle and enables/disables PWM dimming.

BIT				12	11	10	9	8
Field				FADE_5	DUTY_5[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	7	6	5	4	3	2	1	0
Field	DUTY_5[7:0]							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_5	12	LED5 PWM Dimming Enable	0x0: Enabled 0x1: Disabled
DUTY_5	11:0	LED5 Duty-Cycle Selection: 0x000 = Off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

PWM6 = 0x2A

PWM6 is a read/write register that configures the LED6 duty cycle and enables/disables PWM dimming.

BIT				12	11	10	9	8
Field				FADE_6	DUTY_6[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	7	6	5	4	3	2	1	0
Field	DUTY_6[7:0]							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_6	12	LED6 PWM Dimming Enable	0x0: Enabled 0x1: Disabled
DUTY_6	11:0	LED 6 Duty-Cycle Selection: 0x000 = Off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

PWM7 = 0x2B

PWM7 is a read/write register that configures the LED7 duty cycle and enables/disables PWM dimming.

BIT				12	11	10	9	8
Field				FADE_7	DUTY_7[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	DUTY_7[7:0]							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_7	12	LED7 PWM Dimming Enable	0x0: Enabled 0x1: Disabled
DUTY_7	11:0	LED7 Duty-Cycle Selection: 0x000 = Off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

PWM8 = 0x2C

PWM8 is a read/write register that configures the LED8 duty cycle and enables/disables PWM dimming.

BIT				12	11	10	9	8
Field				FADE_8	DUTY_8[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	DUTY_8[7:0]							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_8	12	LED8 PWM Dimming Enable	0x0: Enabled 0x1: Disabled
DUTY_8	11:0	LED8 Duty-Cycle Selection: 0x000 = Off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

PWM9 = 0x2D

PWM9 is a read/write register that configures the LED9 duty cycle and enables/disables PWM dimming.

BIT				12	11	10	9	8
Field				FADE_9	DUTY_9[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	DUTY_9[7:0]							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_9	12	LED9 PWM Dimming Enable	0x0: Enabled 0x1: Disabled
DUTY_9	11:0	LED9 Duty-Cycle Selection: 0x000 = Off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

PWM10 = 0x2E

PWM10 is a read/write register that configures the LED10 duty cycle and enables/disables PWM dimming.

BIT				12	11	10	9	8
Field				FADE_10	DUTY_10[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	DUTY_10[7:0]							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_10	12	LED10 PWM Dimming Enable	0x0: Enabled 0x1: Disabled
DUTY_10	11:0	LED10 Duty-Cycle Selection: 0x000 = Off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

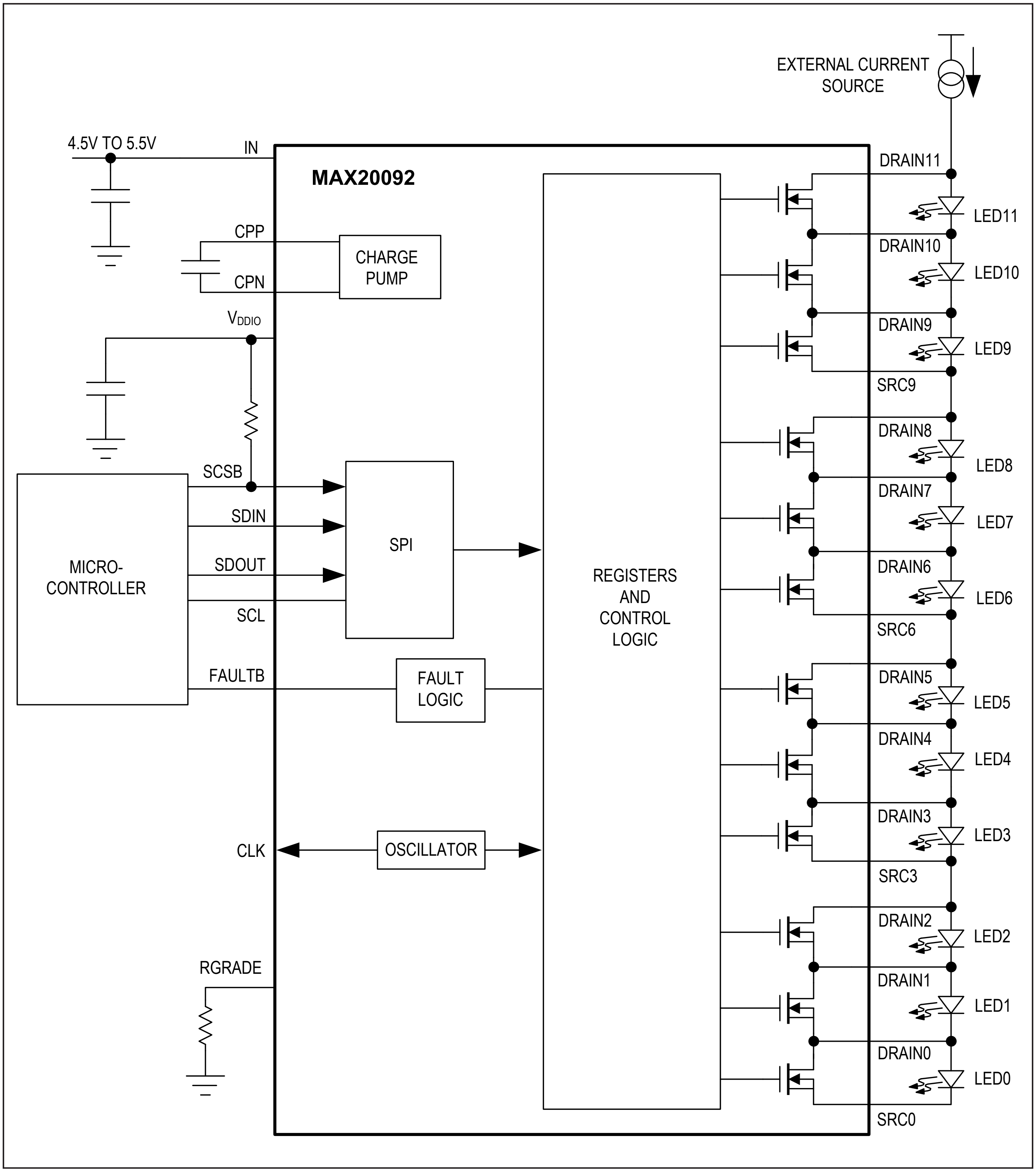
PWM11 = 0x2F

PWM11 is a read/write register that configures the LED11 duty cycle and enables/disables PWM dimming.

BIT				12	11	10	9	8
Field				FADE_11	DUTY_11[11:8]			
Reset				0b0	0x000			
Access Type				Write, Read	Write, Read			
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	DUTY_11[7:0]							
Reset	0x000							
Access Type	Write, Read							

BITFIELD	BITS	DESCRIPTION	DECODE
FADE_11	12	LED11 PWM Dimming Enable	0x0: Enabled 0x1: Disabled
DUTY_11	11:0	LED11 Duty-Cycle Selection: 0x000 = Off 0x001 = 1/4095 duty cycle ... 0xfff = 100% duty cycle	

Typical Application Circuits



Ordering Information

PART	TEMP RANGE	PIN-PACKAGE
MAX20092ATJ/VY+	-40°C to +125°C	32 SWTQFN-EP*
MAX20092ATJ/VY+T	-40°C to +125°C	32 SWTQFN-EP*

/V Denotes an automotive-qualified part.
+Denotes a lead(Pb)-free/RoHS-compliant package.
SW = Side-wettable TQFN package.
**EP = Exposed pad.*
T = Tape-and-reel package.