BRIGHTEK

BRIGHTEK (EUROPE) LIMITED

Brighten Up The World With LED!

APPLICATION NOTES

ATTENTION

OBSERVE PRECAUTIONS FOR HANDLING ELECTROSTATIC SENSITIVE DEVICES

- ▶ EMC SMD with IC
- ▶ 3838 IC 0.85t
- ▶ Red/Green/Blue

Release Date: 18 February 2026 Version: A1.6

N0M67S34IC Application Note

Sleep Mode

3838 IC Integrated

3838 IC-Integrated

RoHS Compliant

AUTOMOTIVE AEC-Q100 AUTOMOTIVE AEC-Q102

FEATURES:

- **Package:** EMC EIA STD Package with Integrated IC
- **Forward Current:** max.60mA/channel
- **Forward Voltage (typ.):** +4.0~+5.5V
- **Luminous Intensity (typ.):** 4000mcd mixed white@63mA
- **Dominant Wavelength (typ.):** 625/525/460nm
- **Operating Temperature:** -40~+125°C
- **Storage Temperature:** -40~+125°C
- **IC Feature:**
  - ✓ Support tunable 8-16 bits PWM dimming for each RGB output to realize simple modulation of colour and brightness.
  - ✓ 6 bits global current adjustment.
  - ✓ Excellent high VDD voltage endurance up to 10V.
  - ✓ Built-in OTP for colour calibration at D65 to perform great colour consistency.
  - ✓ Built-in thermal detection and reporting.
  - ✓ CRC protected serial communication.
  - ✓ Built-in LED voltage detection, open/short/VDS detections and reporting function.
  - ✓ Watch-Dog function to prevent flicking caused by hot-plug.
  - ✓ Signal transmission frequency up to 5MHz.

APPLICATIONS:

- Automotive
- Telecommunication
- Indicator
- Home Appliance
- Decoration Lighting
- Full Colour LED Strip

1. INTRODUCTION:

This device supports 64 discrete levels of current regulation, allowing fine-tuned brightness control with a maximum luminous intensity of up to 4000 millicandelas (mcd). It utilizes 16-bit pulse-width modulation (PWM) for high-resolution dimming and smooth, precise colour transitions, facilitating complex and dynamic ambient lighting effects.

The integrated LED driver architecture supports daisy-chained serial communication, enabling scalable control of multiple LEDs connected in series. Data integrity is maintained through an embedded cyclic redundancy check (CRC) mechanism, which provides robust error detection and enhances communication reliability under noisy or long-distance transmission conditions.

Additionally, the system incorporates on-chip temperature sensing and internal voltage feedback circuits. These features allow for real-time monitoring of thermal and electrical parameters, ensuring stable LED performance and enabling closed-loop control for improved reliability and thermal management.

2.1 APPLICATION CIRCUIT:

Figure 1 presents the recommended LED connection circuit for practical implementation. In diagram (a), the inclusion of an RC filter, a bypass capacitor, and a TVS diode is advised to improve noise immunity and stabilize the power supply. Pin 4 (NC) of the LED should be left floating and must not be connected.

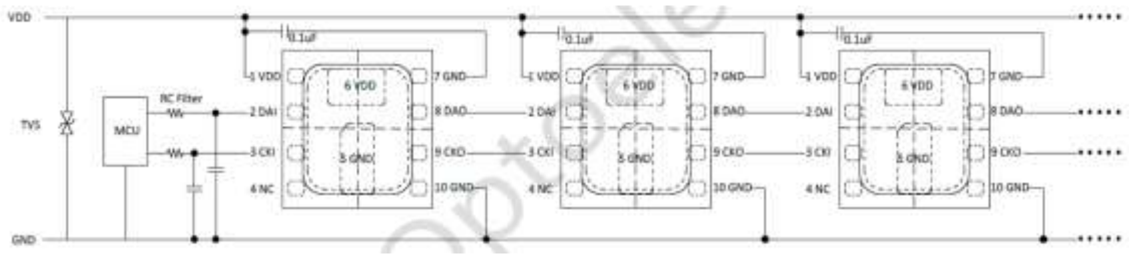


Figure 1. Diagram (a)

Diagrams (b-1) and (b-2) depict two possible data feedback configurations for the final LED in the chain, applicable when signal return functionality is required.

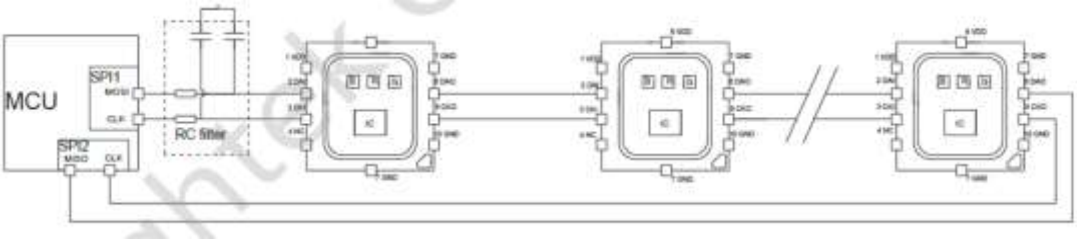


Diagram (b-1) High-speed data report configuration

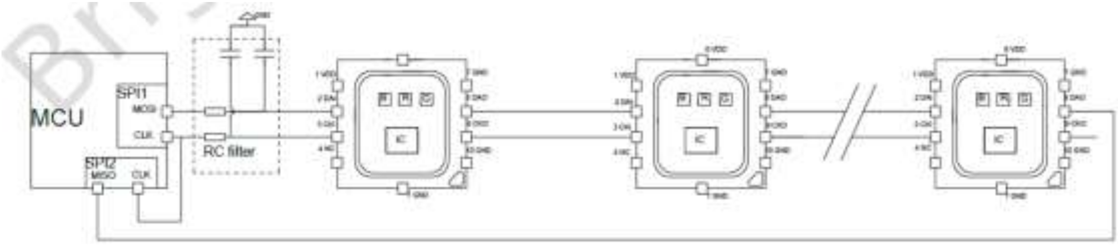


Diagram (b-2) Alternative Data Report Configuration

In the high-speed feedback configuration illustrated in Diagram b-1, the CKO and DAO pins of the last LED are routed to an alternate SPI2 interface or general-purpose I/O (GPIO) on the MCU. This setup enables bidirectional data communication at rates of up to 10 Mbps.

In the alternate feedback configuration shown in Diagram b-2, the CKO pin of the final LED is left unconnected, while the DAO pin is linked to the MCU's MISO pin or a suitable GPIO. In this mode, the return data is sampled based on the clock signal originating from the first LED in the chain, with the MOSI line held inactive. However, this configuration may result in signal phase misalignment due to cumulative propagation delays between LEDs, with a typical delay of ≤30.5ns per LED. Consequently, the MCU must account for the total phase shift across the chain when issuing return commands to ensure correct data interpretation.

2.2 POWER SUPPLY:

The recommended typical supply voltage for the EVO 3838 LED is 5V. When multiple LEDs are connected in series, it is important to minimize the length and maximize the width of the 5V power trace on the PCB to reduce voltage drop resulting from trace resistance.

The drive current for each LED should be configured according to the specific operating conditions. Refer to the LED’s datasheet for the maximum rated current, and select a power supply capable of delivering sufficient current based on the application’s brightness, color requirements, and overall power budget.

To ensure stable operation and eliminate potential color flickering, it is strongly recommended to place a 0.1µF ceramic bypass capacitor between VDD and GND for each LED. This capacitor should be positioned as close as possible to the LED power pins in the PCB layout to minimize parasitic inductance and optimize decoupling performance.

2.3 RC FILTER CIRCUIT:

An RC filter circuit is employed to attenuate high-frequency noise during signal transmission. The values of the resistor (R) and capacitor (C) should be selected based on the frequency of the interfering signal (Finterfering), as defined by Equation (1):

$$F_{\text{interfering}} = \frac{1}{2\pi RC} \tag{1}$$

If no significant interference is observed in the application environment, the capacitor may be omitted, and the resistor can be replaced with a 0Ω jumper.

When implementing the RC filter, it is recommended to use a resistor in the range of 100Ω to 300Ω to suppress noise without causing excessive signal attenuation. A typical capacitor value of 100pF to 1 nF is suggested, depending on the target filtering frequency.

It is important to ensure that the operating signal frequency of the MCU remains well below the cutoff frequency of the RC filter, so that the control signals are not inadvertently attenuated or distorted by the filtering circuit.

2.4 TVS SELECTION:

It is recommended to place a TVS (Transient Voltage Suppression) diode at the power input of the PCB to provide protection against high-voltage transients. The selected TVS diode should have a breakdown voltage below 8V and a clamping voltage under 9V to ensure effective suppression within the operating range.

2.5 MICROCONTROLLERS:

This product is compatible with signal input levels of both 3.3V and 5V, allowing direct interfacing with MCUs and FPGAs operating at either logic level.

The following MCU models have been successfully tested for compatibility and reliable control of the product:

1. NXP® S32K144, S32K116
2. STMicroelectronics® STM32F103
3. Microchip® PIC18F47Q10

Note: Compatibility is not restricted to the MCU models or brands listed above. Any microcontroller that meets the required communication interface and protocol specifications can be used with this product.

3.1 EXAMPLE OF BASIC OPERATING SEQUENCE:

The following sequence outlines a typical control procedure for managing a chain of LEDs using a controller to implement static or dynamic lighting effects:

I. Setup Phase

a) Power on the LED devices and allow a stabilization period of approximately 1ms.
b) Transmit initialization commands to each LED to configure essential parameters. This includes setting values in the Primary Register, such as global current levels and communication speed for data feedback.
c) Optionally, read each LED’s open/short detection status and internal temperature data.

II. Main Loop

a) Update the color output of selected LEDs as required to generate the desired lighting effect by sending color update commands.

III. Stabilization Loop (Optional)

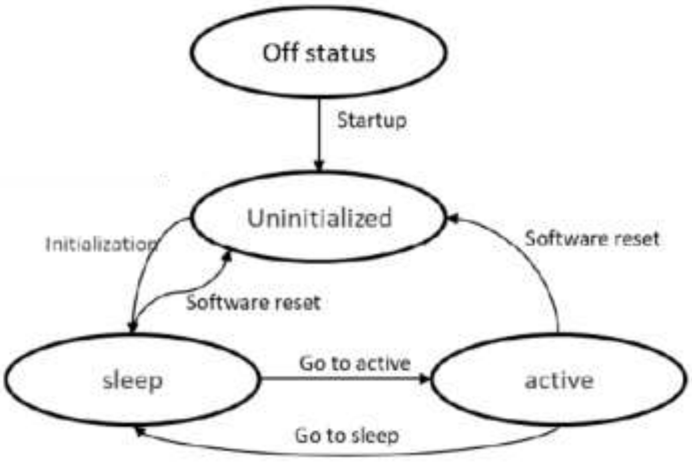
a) Periodically read the internal temperature of each LED.
b) Check for open/short conditions on each device to monitor operational integrity.

IV. Shutdown Phase

a) Send commands to place the LEDs into Sleep Mode.
b) Power down the LED chain by turning off the external power supply.

3.2 DEVICE STATUS:

Figure 2 illustrates the status transitions of the LED device.



Off status: LED device is powered off
Uninitialized: The device has been powered on but not yet initialized
Sleep: The device is in low-power standby mode
Active: The LED is actively illuminated and functioning normally

Figure 2. Device State Diagram

4.1 COMMUNICATION FORMAT:

This product utilises a four-line serial communication interface consisting of two input lines - CKI (Clock In) and DAI (Data In) - and two output lines - CKO (Clock Out) and DAO (Data Out). The communication interface supports data rates of up to 5MHz.

Figure 3 provides the waveform and timing diagram for the communication protocol. Each LED in the chain requires a 72-bit data frame, which may contain command or PWM data. The controller must transmit a total of 72bits per LED, scaled by the number of LEDs connected in series.

At the end of each transmission cycle, CKI and DAI must be held high for a minimum of 200μs to ensure proper data latching.

During data transmission, each LED samples incoming data (DAI) on the falling edge of CKI, with bits transmitted in MSB-first order. Each LED captures the first 72-bit frame addressed to it and shifts the remaining bits to the next device downstream via CKO and DAO.

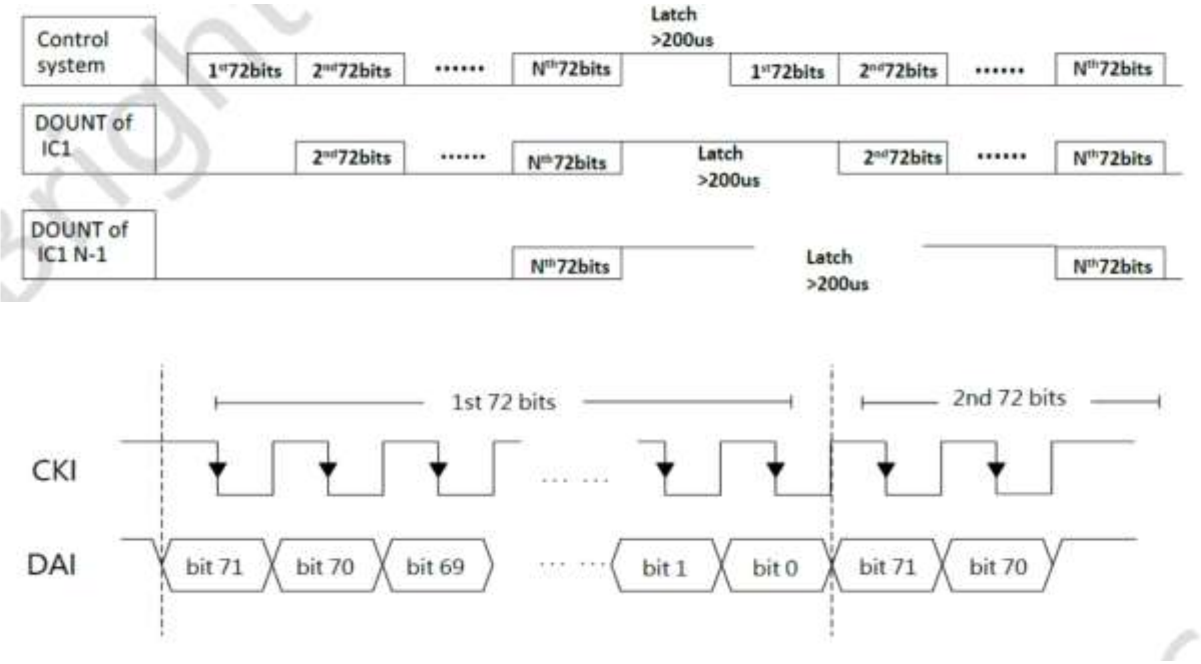


Figure 3. Communication Format

Figure 4 depicts the process of LED data output. While performing status feedback (readback), the MCU needs to keep sending clock signals via CKI. These clock signals pass through all the LEDs and are returned by the CKO pin of the final LED. At the same time, the DAO pin of the last LED outputs the corresponding return data, which is sampled on the rising edge of the clock signal.

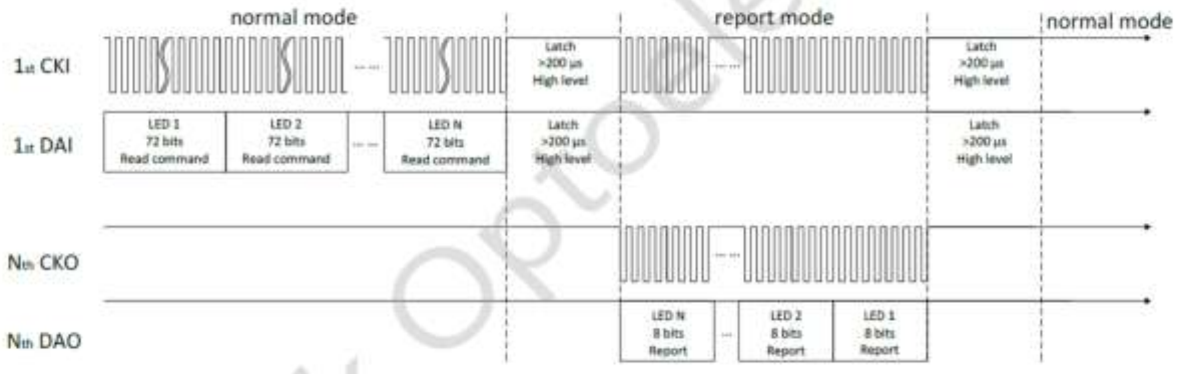


Figure 4. Communication of Report

Notes:

a) After the MCU sends the read command, CKI and DAI need to be set to high potential and wait for at least 200 μ s before starting to send the CLK signal When the MCU sends a CLK signal from the CKI port, CKO sends the same CLK signal as CKI. CKO rising edge collects DAO data.

b) The number of clock pulses required for a readback operation (CKI pin) equals the number of LEDs × 8 bits.

c) The MCU will receive the status feedback value of the last device first, and the status feedback value of the first device will be received by the MCU last.

4.2 REGISTER MAPPING:

Figure 5 and Table 1 detail the LED command packet format. LED functionality is configured by the controller through writes to internal registers. Each 72-bit command packet is structured as follows:

- Bits [71:67]: 5-bit CRC checksum for error detection
- Bits [66:64]: 3-bit command type field (CMD)
- Bits [63:0]: 64-bit register data payload (REG)

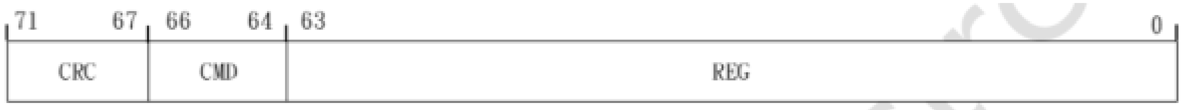
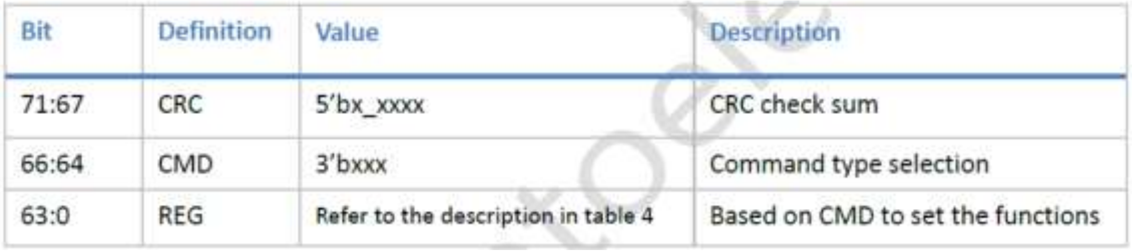


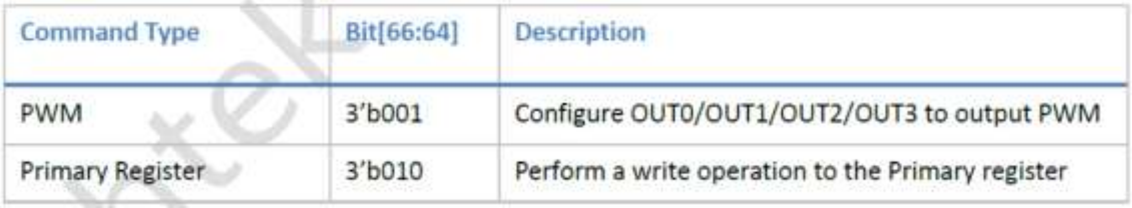
Figure 5. Command Composition Format



Bit	Definition	Value	Description
71:67	CRC	5'bx_xxxx	CRC check sum
66:64	CMD	3'bxxx	Command type selection
63:0	REG	Refer to the description in table 4	Based on CMD to set the functions

Table 1. Register Mapping

Table 2 lists the available CMD types, including PWM setting and Primary Register writing.



Command Type	Bit[66:64]	Description
PWM	3'b001	Configure OUT0/OUT1/OUT2/OUT3 to output PWM
Primary Register	3'b010	Perform a write operation to the Primary register

Table 2. CMD Type

Each register write command includes a 5-bit CRC code located at bits [71:67], which serves to validate the integrity of the transmitted data. Each CRC bit is derived by performing a bitwise XOR operation on a defined subset of bits within the command, combined with a specified initial value, as outlined in Table 3.

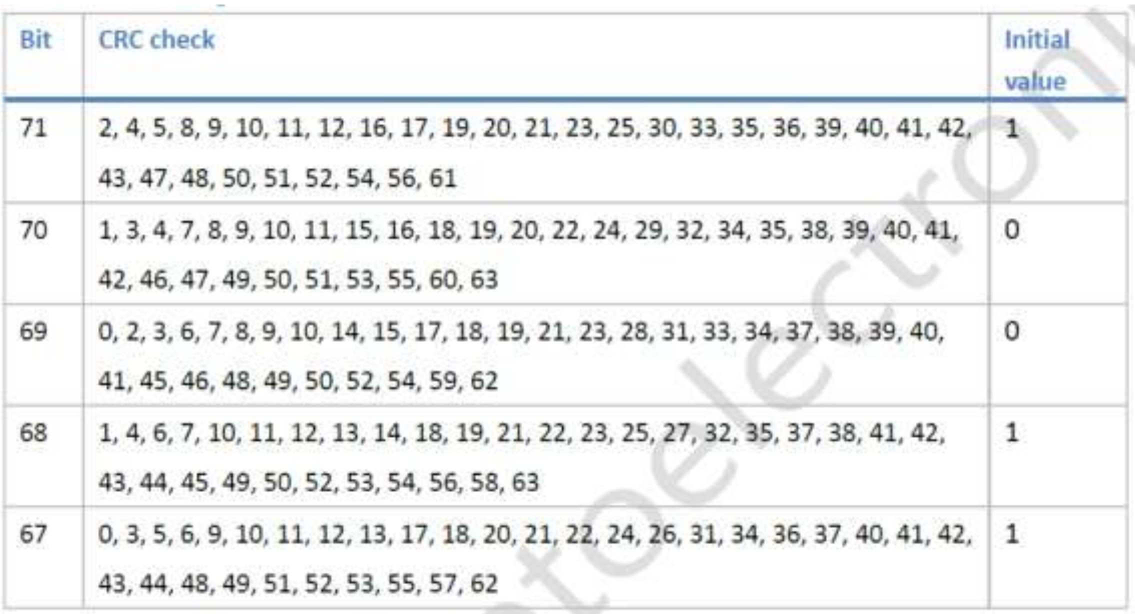
For instance:

- Bit[71] = Bit[2] ^ Bit[4] ^ … ^ Bit[61] ^ 1
- Bit[70] = Bit[1] ^ Bit[3] ^ … ^ Bit[63] ^ 0

Notes:

a) The bit positions referenced correspond to indices in the binary representation of the full 72-bit command.

b) The symbol "^" denotes a bitwise XOR operation. For example, "^1" signifies an XOR with the initial value 1.



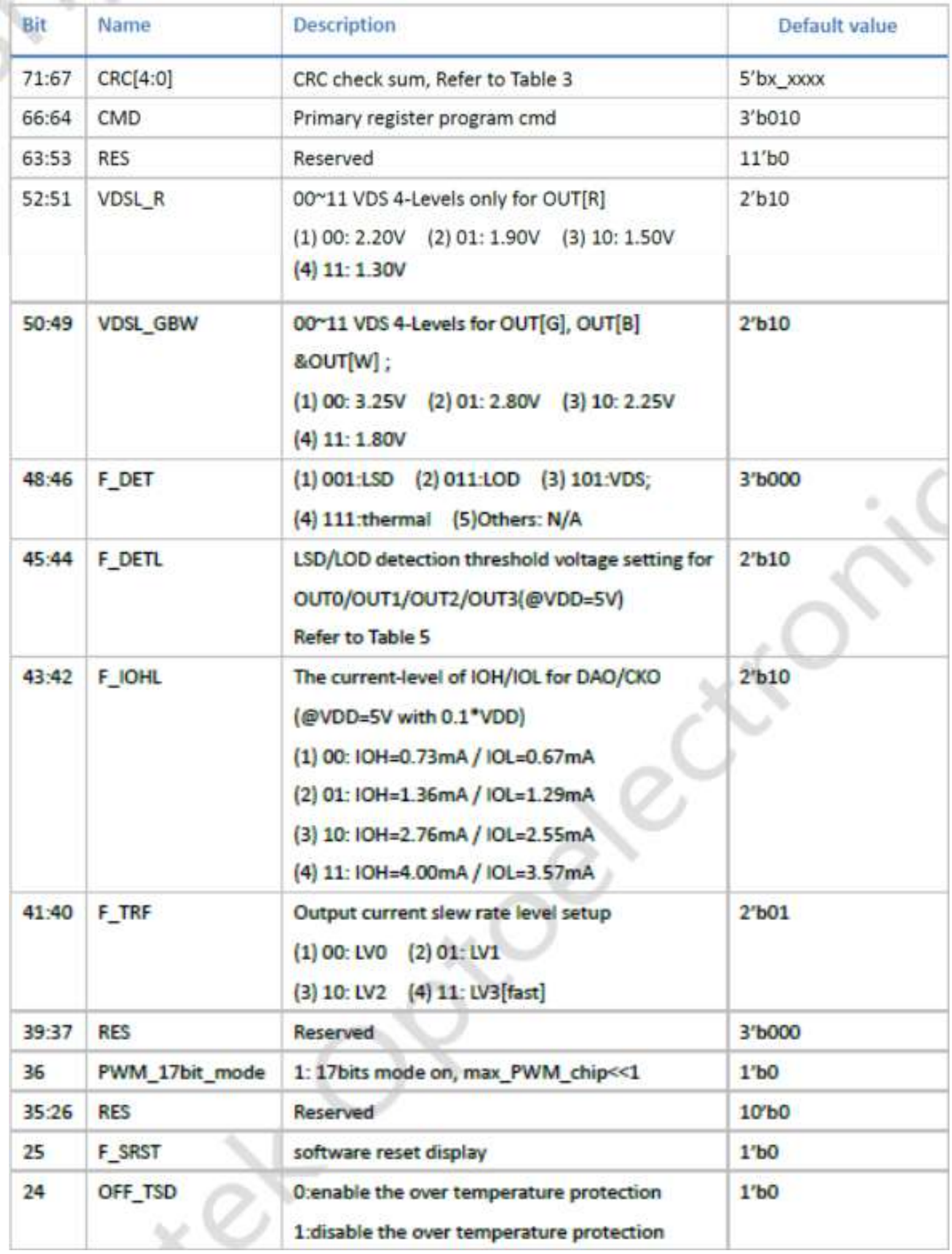
Bit	CRC check	Initial value
71	2, 4, 5, 8, 9, 10, 11, 12, 16, 17, 19, 20, 21, 23, 25, 30, 33, 35, 36, 39, 40, 41, 42, 43, 47, 48, 50, 51, 52, 54, 56, 61	1
70	1, 3, 4, 7, 8, 9, 10, 11, 15, 16, 18, 19, 20, 22, 24, 29, 32, 34, 35, 38, 39, 40, 41, 42, 46, 47, 49, 50, 51, 53, 55, 60, 63	0
69	0, 2, 3, 6, 7, 8, 9, 10, 14, 15, 17, 18, 19, 21, 23, 28, 31, 33, 34, 37, 38, 39, 40, 41, 45, 46, 48, 49, 50, 52, 54, 59, 62	0
68	1, 4, 6, 7, 10, 11, 12, 13, 14, 18, 19, 21, 22, 23, 25, 27, 32, 35, 37, 38, 41, 42, 43, 44, 45, 49, 50, 52, 53, 54, 56, 58, 63	1
67	0, 3, 5, 6, 9, 10, 11, 12, 13, 17, 18, 20, 21, 22, 24, 26, 31, 34, 36, 37, 40, 41, 42, 43, 44, 48, 49, 51, 52, 53, 55, 57, 62	1

Table 3. CRC Calculating Data

4.3.1 PRIMARY REGISTER FORMAT:

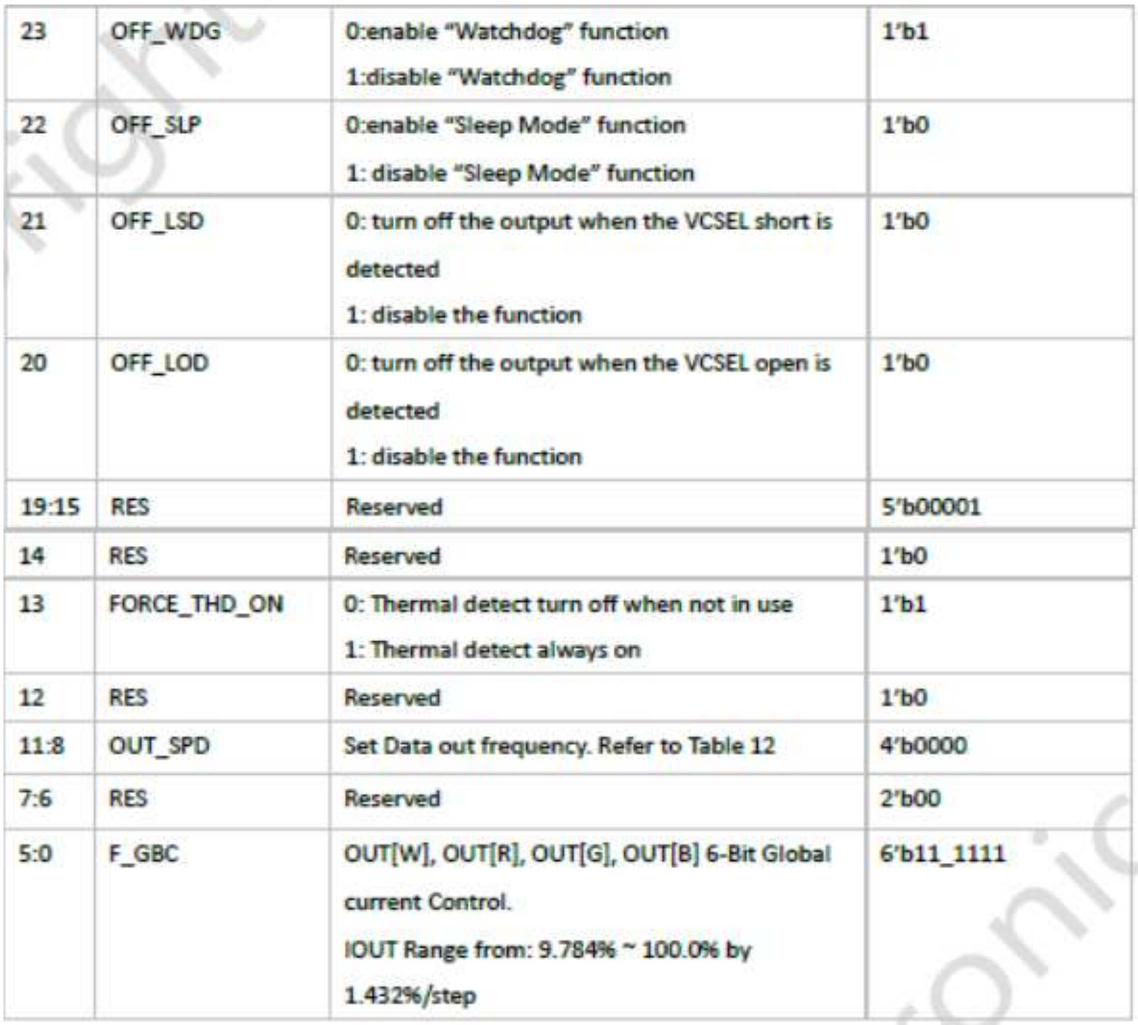
Table 4 specifies the format of the Primary Register. During device initialization, this register is commonly configured to define parameters such as brightness control, data return mode, and current scaling.

Additionally, it can be used to trigger internal status feedback. For instance, setting F_DET = 3’b111 enables the device to return temperature data.



Bit	Name	Description	Default value
71:67	CRC[4:0]	CRC check sum, Refer to Table 3	5'bx_xxxx
66:64	CMD	Primary register program cmd	3'b010
63:53	RES	Reserved	11'b0
52:51	VDSL_R	00~11 VDS 4-Levels only for OUT[R] (1) 00: 2.20V (2) 01: 1.90V (3) 10: 1.50V (4) 11: 1.30V	2'b10
50:49	VDSL_GBW	00~11 VDS 4-Levels for OUT[G], OUT[B] &OUT[W] ; (1) 00: 3.25V (2) 01: 2.80V (3) 10: 2.25V (4) 11: 1.80V	2'b10
48:46	F_DET	(1) 001:LSD (2) 011:LOD (3) 101:VDS; (4) 111:thermal (5)Others: N/A	3'b000
45:44	F_DETL	LSD/LOD detection threshold voltage setting for OUT0/OUT1/OUT2/OUT3(@VDD=5V) Refer to Table 5	2'b10
43:42	F_IOHL	The current-level of IOH/IOL for DAO/CKO (@VDD=5V with 0.1*VDD) (1) 00: IOH=0.73mA / IOL=0.67mA (2) 01: IOH=1.36mA / IOL=1.29mA (3) 10: IOH=2.76mA / IOL=2.55mA (4) 11: IOH=4.00mA / IOL=3.57mA	2'b10
41:40	F_TRF	Output current slew rate level setup (1) 00: LV0 (2) 01: LV1 (3) 10: LV2 (4) 11: LV3[fast]	2'b01
39:37	RES	Reserved	3'b000
36	PWM_17bit_mode	1: 17bits mode on, max_PWM_chip<<1	1'b0
35:26	RES	Reserved	10'b0
25	F_SRST	software reset display	1'b0
24	OFF_TSD	0:enable the over temperature protection 1:disable the over temperature protection	1'b0

Table 4. Primary Register (continue)



23	OFF_WDG	0:enable "Watchdog" function 1:disable "Watchdog" function	1'b1
22	OFF_SLP	0:enable "Sleep Mode" function 1: disable "Sleep Mode" function	1'b0
21	OFF_LSD	0: turn off the output when the VCSEL short is detected 1: disable the function	1'b0
20	OFF_LOD	0: turn off the output when the VCSEL open is detected 1: disable the function	1'b0
19:15	RES	Reserved	5'b00001
14	RES	Reserved	1'b0
13	FORCE_THD_ON	0: Thermal detect turn off when not in use 1: Thermal detect always on	1'b1
12	RES	Reserved	1'b0
11:8	OUT_SPD	Set Data out frequency. Refer to Table 12	4'b0000
7:6	RES	Reserved	2'b00
5:0	F_GBC	OUT[W], OUT[R], OUT[G], OUT[B] 6-Bit Global current Control. IOUT Range from: 9.784% ~ 100.0% by 1.432%/step	6'b11_1111

Table 4. Primary Register (continue)

4.3.2 LED OPEN/SHORT DETECTIONS:

Figure 6 shows the internal configuration of the LED diode connections. Each color channel (R, G, B) monitors its respective cathode output voltage, denoted as Vout[R], Vout[G], and Vout[B]. These voltages are compared against a defined threshold to detect open-circuit or short-circuit conditions.

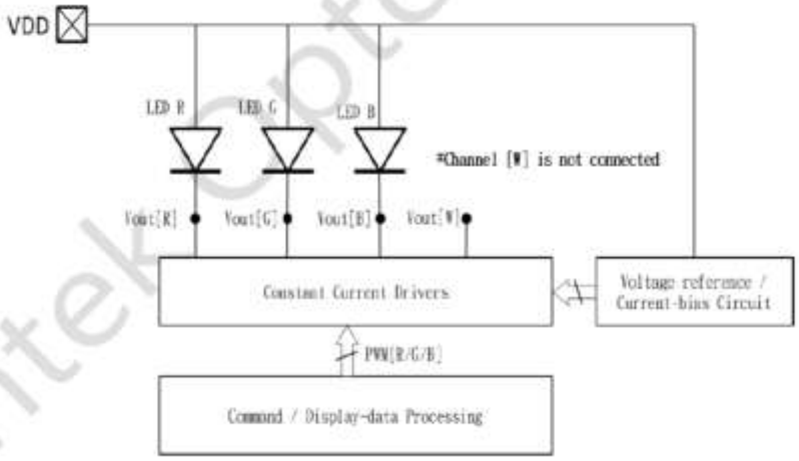
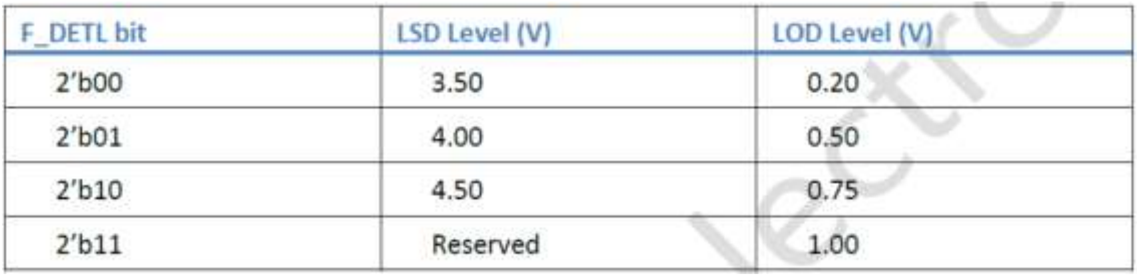

VDD
LED R
LED G
LED B
*Channel [W] is not connected
Vout[R]
Vout[G]
Vout[B]
Vout[W]
Constant Current Drivers
Voltage reference / Current-bias Circuit
PWM[R/G/B]
Command / Display-data Processing

Figure 6. LED R/G/B Connection Inside the Device

The detection mode is configured via the F_DET field (bits [48:46]) in the Primary Register, which allows selection between open detection (LOD) and short detection (LSD). The voltage threshold for comparison is set using the F_DETL field (bits [45:44]), with the corresponding levels defined in Table 5.

- Open detection (LOD): A fault is flagged if the sensed voltage is below the configured threshold.
- Short detection (LSD): A fault is flagged if the sensed voltage is above the configured threshold.



F_DETL bit	LSD Level (V)	LOD Level (V)
2'b00	3.50	0.20
2'b01	4.00	0.50
2'b10	4.50	0.75
2'b11	Reserved	1.00

Table 5.F_DETL

Figure 7 illustrates the open/short detection sequence. A complete detection cycle involves transmitting two Primary Register commands and receiving two corresponding return values. Table 6 defines the format of the 8-bit result:

- A return value of 1 indicates an abnormal condition (open or short).
- A return value of 0 indicates normal operation.

Note: As the white (W) channel is unused, LOD_W and LSD_W values can be disregarded.

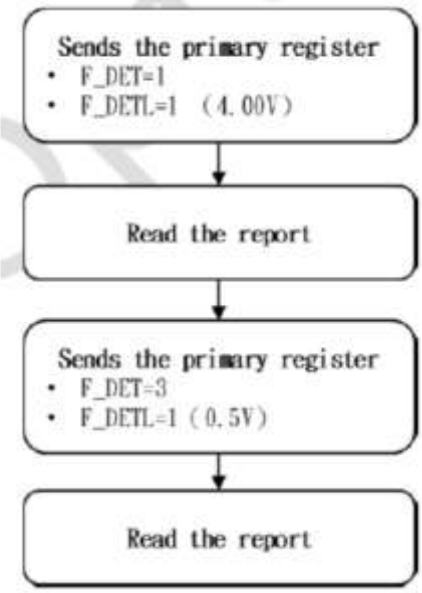
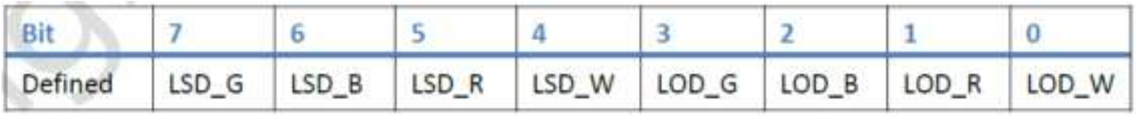


Figure 7. Open Short Circuit Detection Process



Bit	7	6	5	4	3	2	1	0
Defined	LSD_G	LSD_B	LSD_R	LSD_W	LOD_G	LOD_B	LOD_R	LOD_W

Table 6. The 8 Bits Report Data Format of LSD and LOD

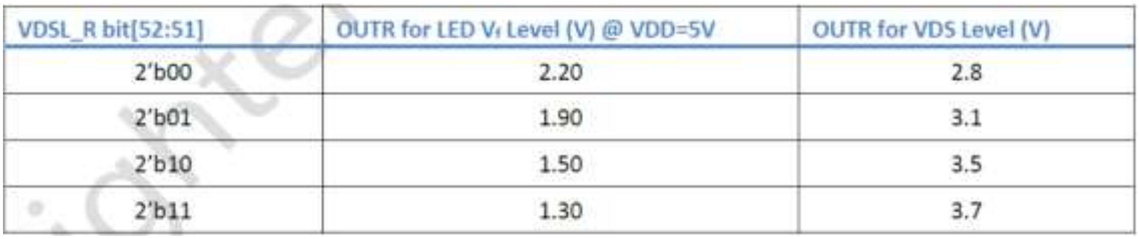
4.3.3 V_{DS} DETECTIONS:

VDS (Drain-to-Source Voltage) represents the voltage drop across the current driver’s drain and source terminals. The device monitors VDS and estimates the forward voltage (Vf) of the R/G/B LEDs using the following relation:

$$\mathrm{Vf} = \mathrm{VDD} - \mathrm{VDS} \qquad \ldots\ldots(2)$$

By comparing the measured VDS against a predefined threshold, the device determines whether an LED channel is operating within its expected range. If VDS for a given channel exceeds its threshold, the device returns a logic **'1'**, indicating that the corresponding Vf is *lower* than the expected minimum.

Table 7 defines the VDS reference levels for the red channel (VDSL_R, bits [52:51]) at VDD = 5V. Due to the inherently lower forward voltage of red LEDs, VDS tends to be higher, necessitating a separate reference range.

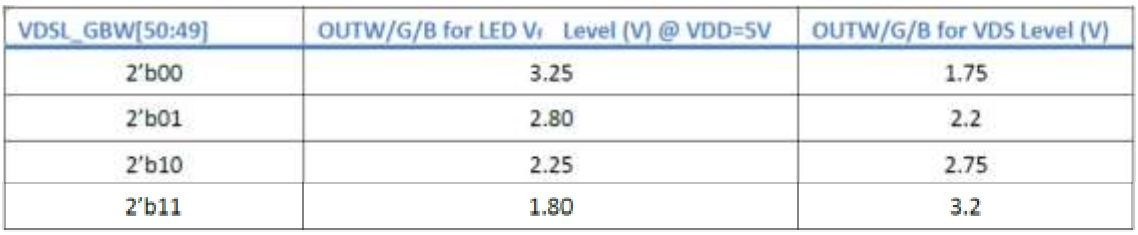


VDSL_R bit[52:51]	OUTR for LED V_f Level (V) @ VDD=5V	OUTR for VDS Level (V)
2'b00	2.20	2.8
2'b01	1.90	3.1
2'b10	1.50	3.5
2'b11	1.30	3.7

Table 7. The reference voltage-level of LED R

Table 8 provides the VDS reference levels for green, blue, and white channels (VDSL_GBW, bits [50:49]) under the same VDD condition.

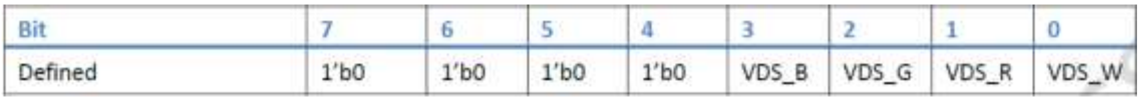
Note: While VDS threshold levels are fixed, the actual reference voltage levels vary with VDD, and this must be taken into account during evaluation.



VDSL_GBW[50:49]	OUTW/G/B for LED V_f Level (V) @ VDD=5V	OUTW/G/B for VDS Level (V)
2'b00	3.25	1.75
2'b01	2.80	2.2
2'b10	2.25	2.75
2'b11	1.80	3.2

Table 8. The reference voltage-level of LED G/B/W

Table 9 defines the return data format. A return value of 1 indicates that the measured VDS exceeds the configured threshold, while a return value of 0 indicates normal operation.



Bit	7	6	5	4	3	2	1	0
Defined	1'b0	1'b0	1'b0	1'b0	VDS_B	VDS_G	VDS_R	VDS_W

Table 9. The 8 Bits Report Data Format of V_{DS} Detection

Figure 8 illustrates the VDS detection process for the red channel. By sweeping through different VDSL_R settings and monitoring the returned results, the forward voltage (Vf) of the red LED can be estimated.

For example, at VDD = 5 V:

- If VDSL_R = 1 returns 1, it implies VDS > 3.1V and thus Vf < 1.9V.
- Further setting VDSL_R = 2 returns 0, indicating Vf lies between 1.5V and 1.9V.

This approach can be used in applications to assess LED Vf variation and binning consistency.

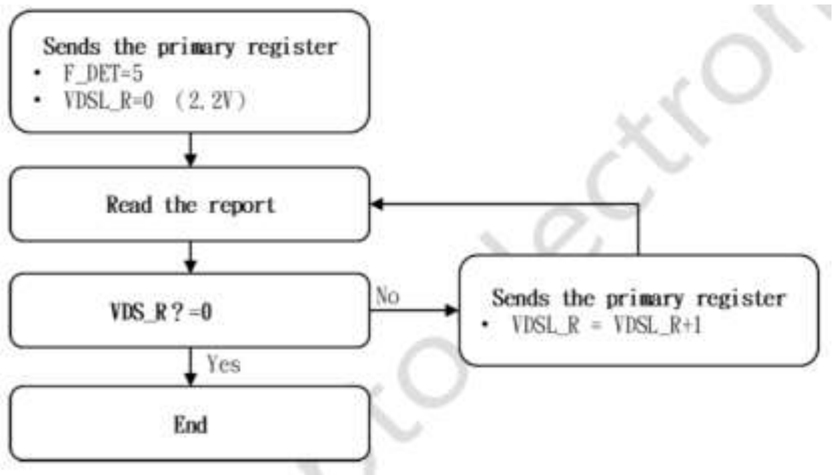


Figure 8. V_{DS} Detection Process for Channel R

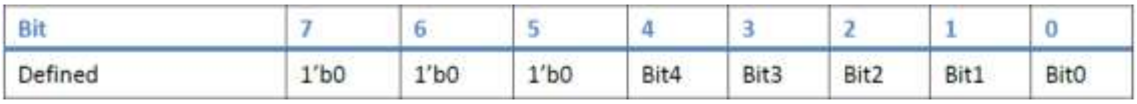
4.3.4 THERMAL DETECTION DATA REPORT:

The device supports thermal feedback functionality through an integrated temperature sensor that monitors the internal operating temperature. This data can be used to adaptively regulate the LED drive current in real time.

When the F_DET field in the Primary Register is configured to 0b111, the device outputs temperature data during the feedback cycle. The returned data is an 8-bit value, where only the lower 5 bits (Bits[4:0]) contain valid temperature information. The upper 3 bits (Bits[7:5]) are reserved.

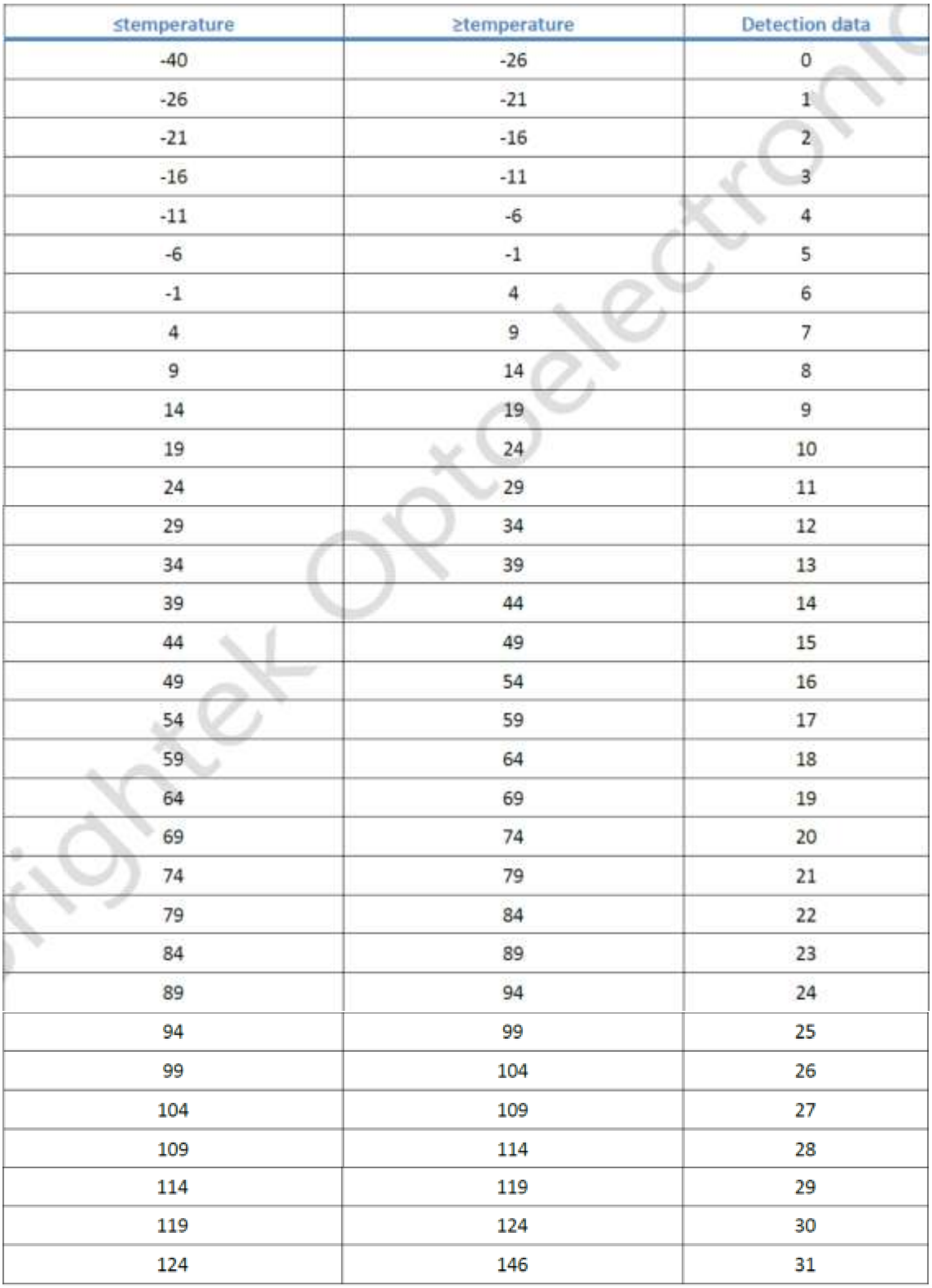
- Table 10 defines the 8-bit return data format for thermal detection.
- Table 11 provides a lookup table mapping the 5-bit temperature code to corresponding temperature ranges, covering 32 discrete levels from -40 °C to +146 °C.

Note: The temperature resolution is approximately 5°C per level, enabling coarse thermal monitoring of the LED package during operation.



Bit	7	6	5	4	3	2	1	0
Defined	1’b0	1’b0	1’b0	Bit4	Bit3	Bit2	Bit1	Bit0

Table 10. The 8 Bits Report Data Format of Thermal Detection



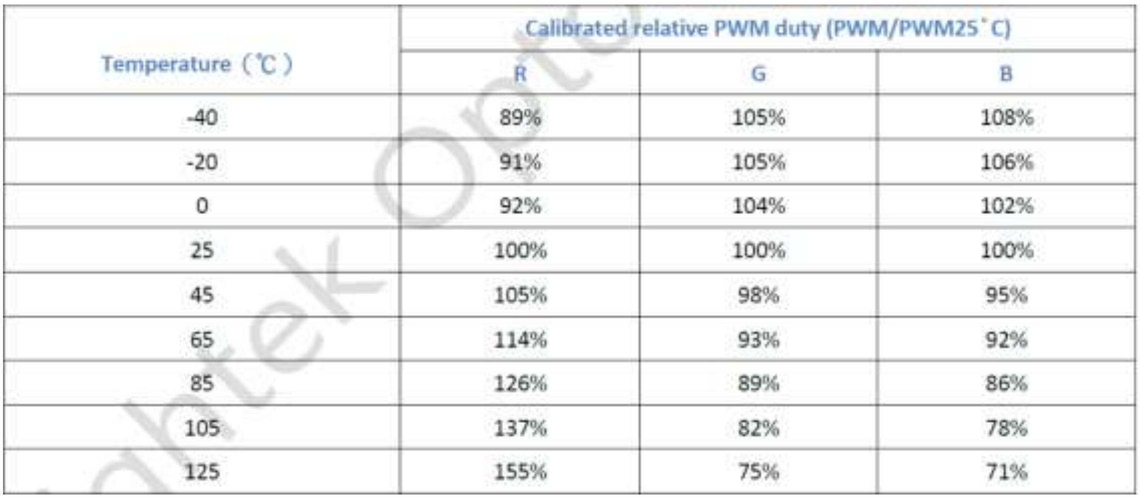
≤temperature	≥temperature	Detection data
-40	-26	0
-26	-21	1
-21	-16	2
-16	-11	3
-11	-6	4
-6	-1	5
-1	4	6
4	9	7
9	14	8
14	19	9
19	24	10
24	29	11
29	34	12
34	39	13
39	44	14
44	49	15
49	54	16
54	59	17
59	64	18
64	69	19
69	74	20
74	79	21
79	84	22
84	89	23
89	94	24
94	99	25
99	104	26
104	109	27
109	114	28
114	119	29
119	124	30
124	146	31

Table 11. Temperature Table

4.3.5 TEMPERATURE COMPENSATION SUGGESTION:

LED brightness and chromaticity are sensitive to temperature fluctuations. Variations in temperature can lead to shifts in luminous efficacy and color coordinates, making it challenging to maintain consistent optical output compared to standard room-temperature conditions.

Table 12 presents recommended compensation ratios for red, green, and blue LEDs across a range of temperatures, calibrated relative to the D65 white point. When ambient temperature data is available - e.g., via an external MCU - these ratios can be used to implement a real-time temperature compensation algorithm by adjusting the R/G/B PWM duty cycles accordingly, thereby minimizing color shift under varying thermal conditions.



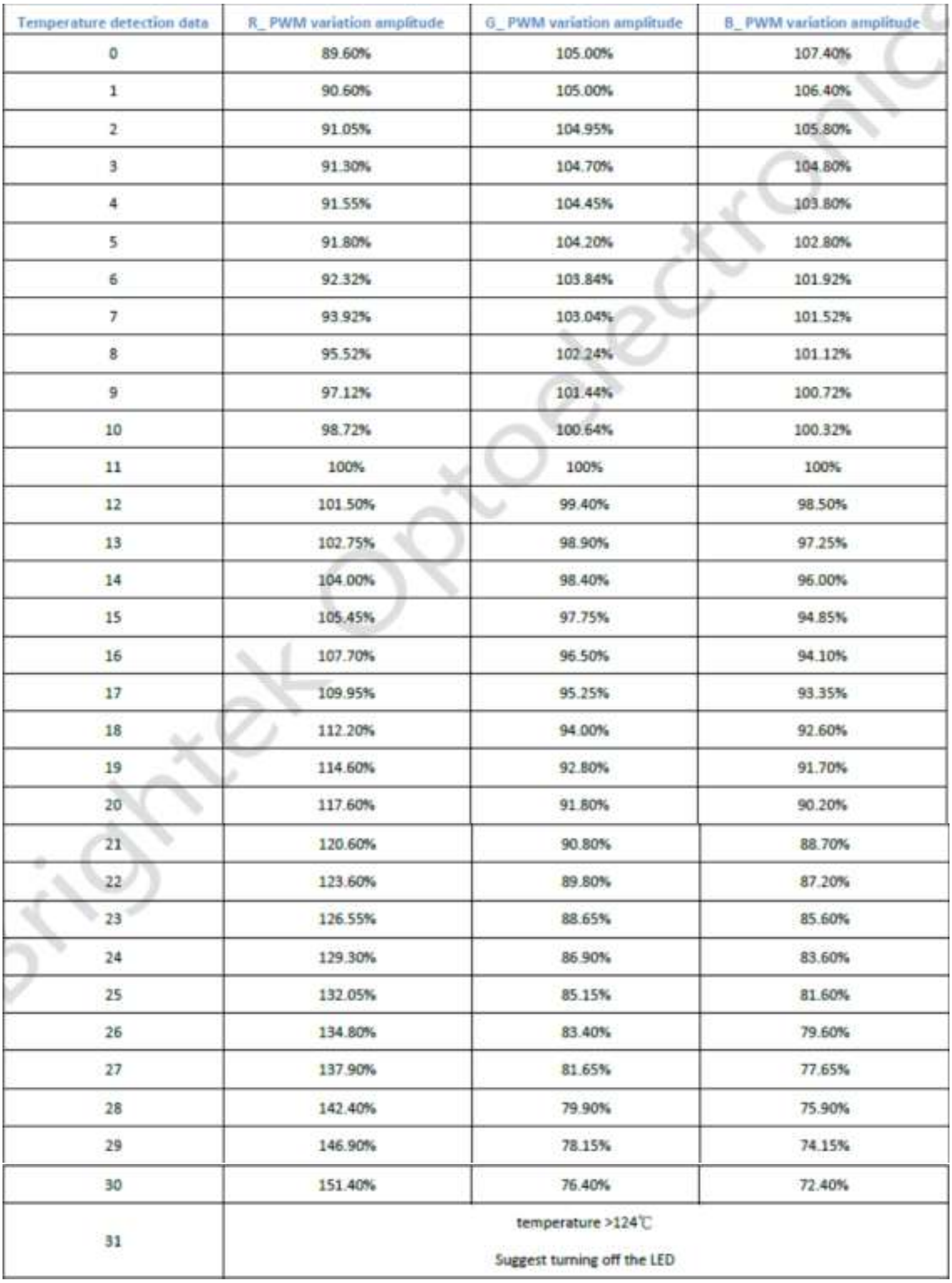
Temperature（℃）	Calibrated relative PWM duty (PWM/PWM25˚C)		
	R	G	B
-40	89%	105%	108%
-20	91%	105%	106%
0	92%	104%	102%
25	100%	100%	100%
45	105%	98%	95%
65	114%	93%	92%
85	126%	89%	86%
105	137%	82%	78%
125	155%	75%	71%

Table 12. Lookup Table of D65 White Light

Table 13 defines the PWM compensation ratios associated with each thermal detection code. Based on the returned temperature code, users can apply the corresponding ratio values for the red, green, and blue channels (denoted as RR, RG, and RB) to compute the temperature-compensated PWM values using the formula:

$$PWM_{R/G/B} = PWM_{Def} \times R_{R/G/B}$$

where PWM_{Def} is the default PWM value under nominal (room-temperature) conditions.



Temperature detection data	R_ PWM variation amplitude	G_ PWM variation amplitude	B_ PWM variation amplitude
0	89.60%	105.00%	107.40%
1	90.60%	105.00%	106.40%
2	91.05%	104.95%	105.80%
3	91.30%	104.70%	104.80%
4	91.55%	104.45%	103.80%
5	91.80%	104.20%	102.80%
6	92.32%	103.84%	101.92%
7	93.92%	103.04%	101.52%
8	95.52%	102.24%	101.12%
9	97.12%	101.44%	100.72%
10	98.72%	100.64%	100.32%
11	100%	100%	100%
12	101.50%	99.40%	98.50%
13	102.75%	98.90%	97.25%
14	104.00%	98.40%	96.00%
15	105.45%	97.75%	94.85%
16	107.70%	96.50%	94.10%
17	109.95%	95.25%	93.35%
18	112.20%	94.00%	92.60%
19	114.60%	92.80%	91.70%
20	117.60%	91.80%	90.20%
21	120.60%	90.80%	88.70%
22	123.60%	89.80%	87.20%
23	126.55%	88.65%	85.60%
24	129.30%	86.90%	83.60%
25	132.05%	85.15%	81.60%
26	134.80%	83.40%	79.60%
27	137.90%	81.65%	77.65%
28	142.40%	79.90%	75.90%
29	146.90%	78.15%	74.15%
30	151.40%	76.40%	72.40%
31	temperature >124℃ Suggest turning off the LED		

Table 13. Lookup Table of Temperature Detection Data and PWM Variation Amplitude (D65)

4.3.6 CKO/DAO FREQUENCY SELECTION:

This device allows configuration of the inter-LED data reporting frequency, enabling optimised data throughput tailored to specific application requirements, as illustrated in Diagram (c).

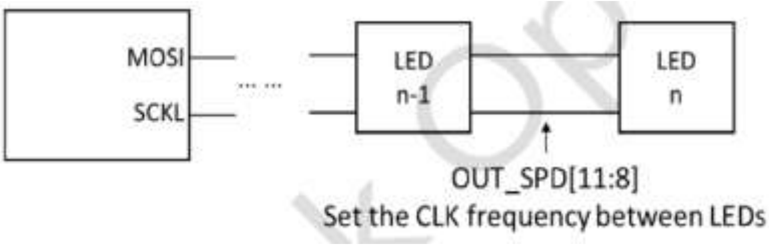


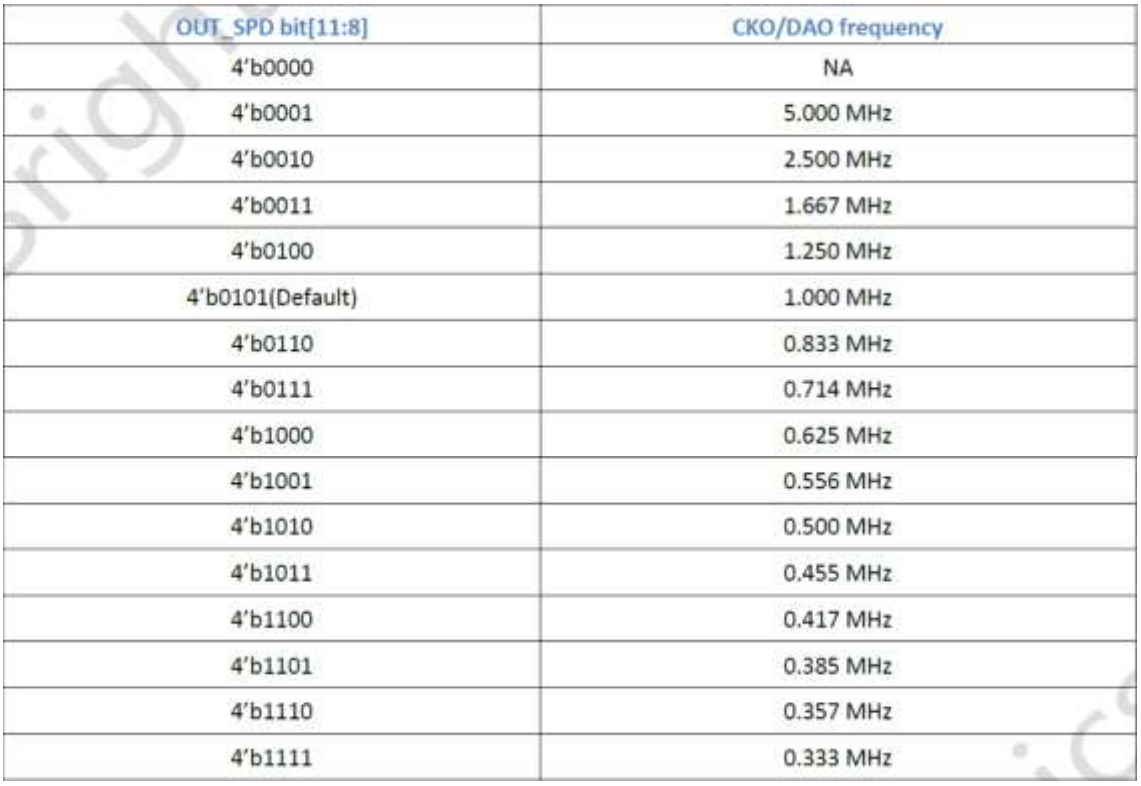
Diagram (C)

The CKO/DAO data transmission rate is primarily governed by the clock signal frequency provided by the MCU, but it is capped by the maximum limit specified in the OUT_SPD field (bits [11:8]) of the Primary Register.

For instance, if OUT_SPD = 4'b0101 (corresponding to 1 MHz) and the MCU supplies a 500 kHz clock during reporting, the effective data rate is 500kHz.

Table 14 defines the allowable frequencies for each OUT_SPD setting.

Note: To ensure reliable data acquisition and prevent timing mismatches, the MCU's report clock frequency must always remain below or equal to the value set in OUT_SPD.



OUT_SPD bit[11:8]	CKO/DAO frequency
4'b0000	NA
4'b0001	5.000 MHz
4'b0010	2.500 MHz
4'b0011	1.667 MHz
4'b0100	1.250 MHz
4'b0101(Default)	1.000 MHz
4'b0110	0.833 MHz
4'b0111	0.714 MHz
4'b1000	0.625 MHz
4'b1001	0.556 MHz
4'b1010	0.500 MHz
4'b1011	0.455 MHz
4'b1100	0.417 MHz
4'b1101	0.385 MHz
4'b1110	0.357 MHz
4'b1111	0.333 MHz

Table 14. CKO/DAO Frequency Table

4.3.7 SOFTWARE RESET:

During PWM operation, the device can be reset by setting F_SRST = 1 (bit [25]) in the Primary Register. Following the completion of the software reset, the user must retransmit the PWM data to restore normal display functionality.

4.3.8 WATCHDOG:

This device incorporates an internal watchdog mechanism to detect abnormal data transmission and automatically initiate error handling, preventing system lockup. In the absence of PWM activity, the device transitions into a low-power mode to reduce energy consumption.

The watchdog feature can be disabled by setting OFF_WDG = 1 (bit [23]) in the Primary Register.

4.3.9 SLEEP MODE:

When all PWM output channels are set to zero (black screen) and the CKI line remains high for more than 450 ms, the device automatically enters Sleep Mode to minimize power consumption.

Figure 9 illustrates the waveform associated with entering Sleep Mode.

While in Sleep Mode, the device can be awakened by applying an 8-bit clock signal on the CKI line. Upon wake-up, normal display operation resumes immediately.

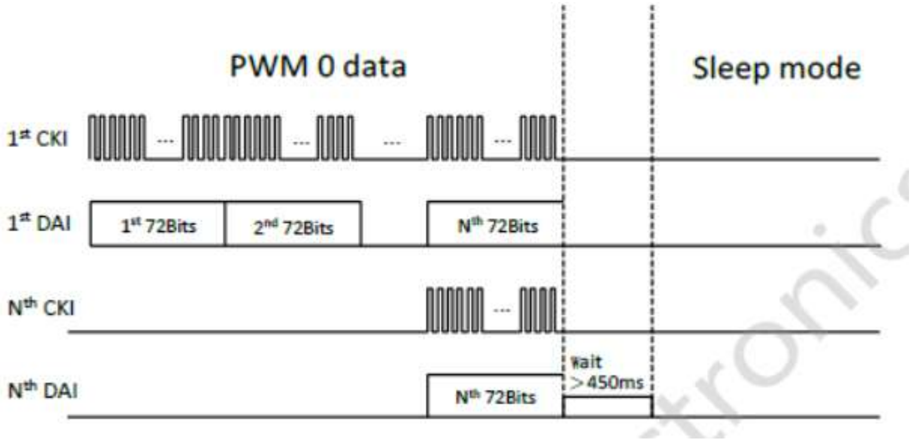


Figure 9. Enter Sleep Mode Waveform

Figure 10 depicts the waveform for exiting Sleep Mode.

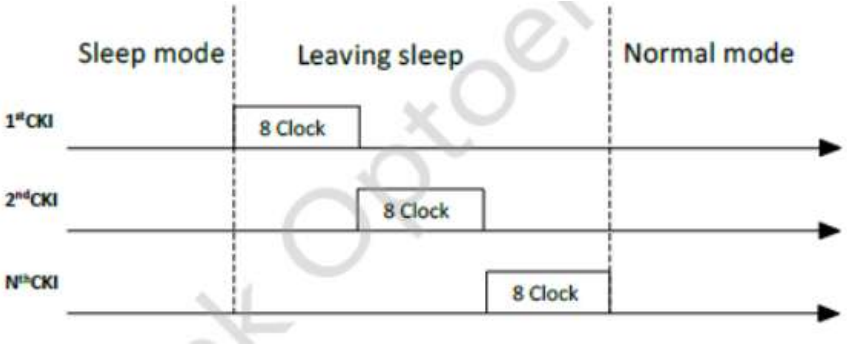


Figure 10. Leave Sleep Mode Waveform

Note: Sleep Mode is a low-power state intended for energy savings during periods without visual output. Recovery occurs instantly upon receiving a valid clock and data input.

4.4 PWM DATA FORMAT:

This product supports 16-bit PWM grayscale control, allowing direct assignment of PWM values for the R, G, B, and W channels via register programming. The PWM data format is detailed in Figure 11 and Table 15. Each PWM command consists of 72 bits, structured as follows:

- Bits [71:67]: CRC checksum
- Bits [66:64]: Command type, where 0b001 indicates a PWM command
- Bits [63:0]: PWM data payload. The PWM data payload contains four 16-bit segments in this order: PWM_W (white), PWM_R (red), PWM_B (blue), and PWM_G (green).

Each LED requires a dedicated 72-bit PWM command, and the total number of commands must correspond to the number of LEDs in the chain.

PWM values are transmitted most significant bit (MSB) first, with a valid range from 0 (fully OFF) to 65,535 (fully ON).

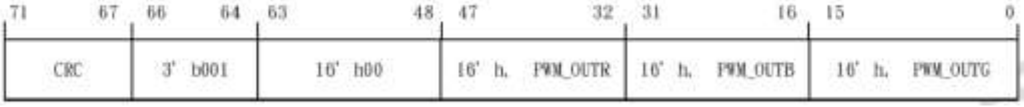
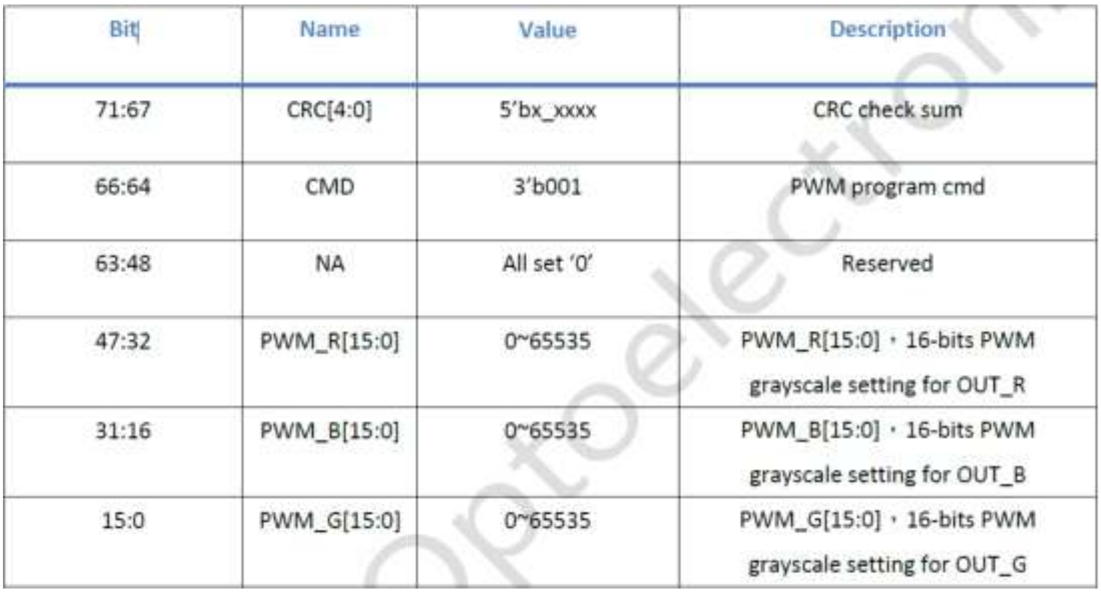


Figure 11. PWM Data Command Format



Bit	Name	Value	Description
71:67	CRC[4:0]	5'bx_xxxx	CRC check sum
66:64	CMD	3'b001	PWM program cmd
63:48	NA	All set '0'	Reserved
47:32	PWM_R[15:0]	0~65535	PWM_R[15:0]，16-bits PWM grayscale setting for OUT_R
31:16	PWM_B[15:0]	0~65535	PWM_B[15:0]，16-bits PWM grayscale setting for OUT_B
15:0	PWM_G[15:0]	0~65535	PWM_G[15:0]，16-bits PWM grayscale setting for OUT_G

Table 15. PWM Data Command Format Table

4.5 UNDER VOLTAGE LOCK OUT:

To guarantee safe and reliable operation, the device incorporates an **Under Voltage Lockout (UVLO)** protection circuit. If the supply voltage (**VDD**) falls below 3.4 V, the device disables PWM output and enters a lockout state. Normal operation automatically resumes once **VDD** exceeds 3.5V.

Note: This protection prevents erratic behavior or display flickering due to insufficient supply voltage.

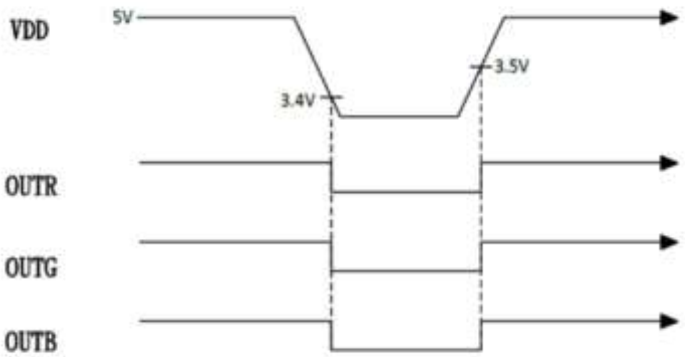
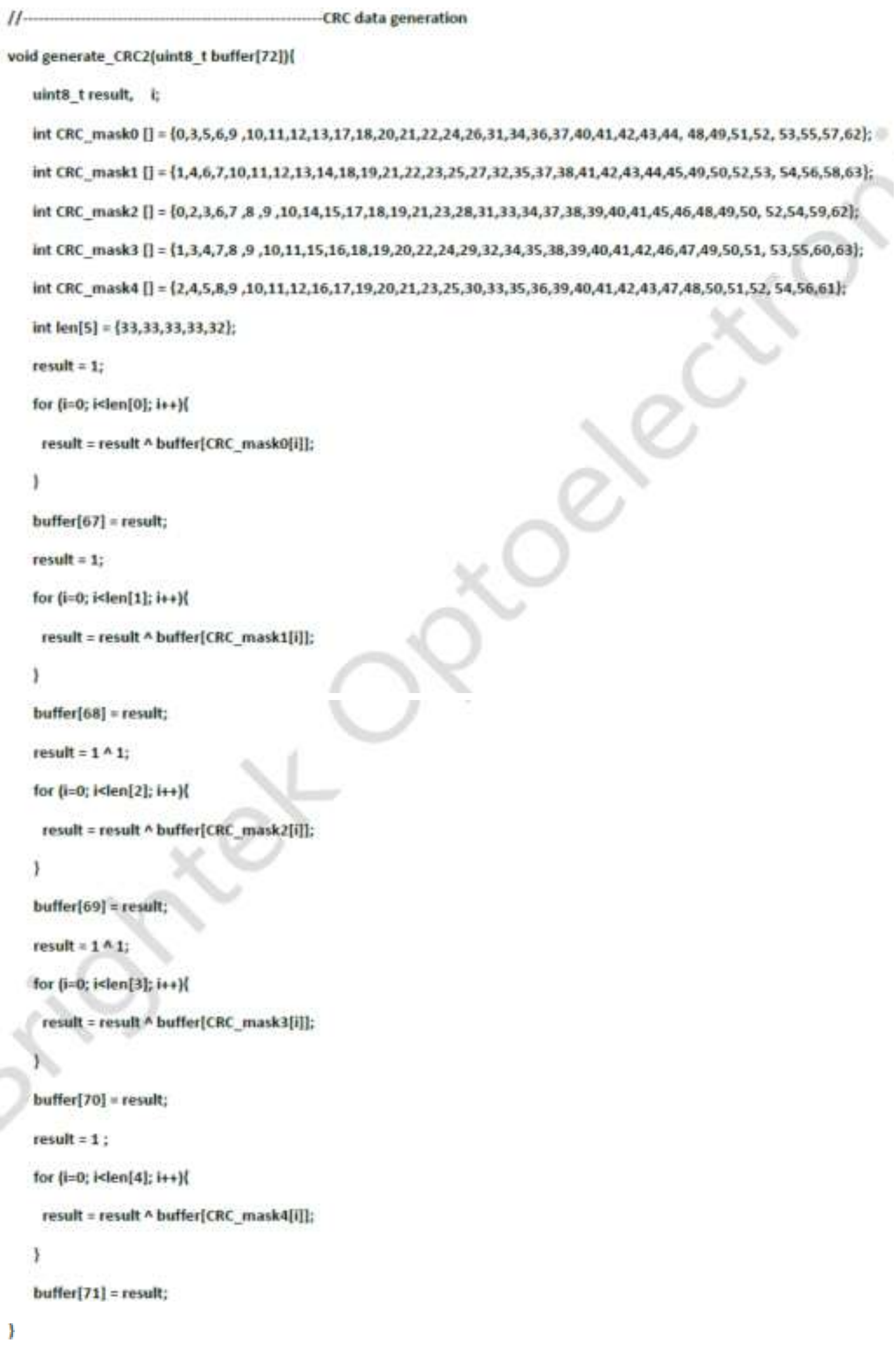


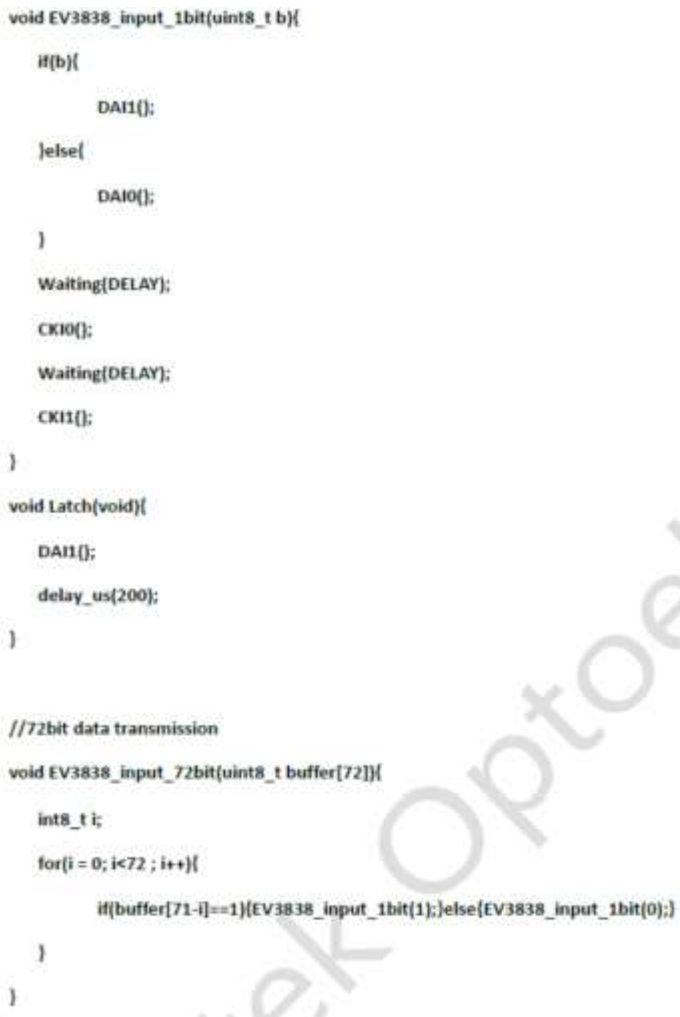
Figure 12. PWM Data Command Format Table

5.1 REVERENCE CODE GENERALIZATION - CRC Code:



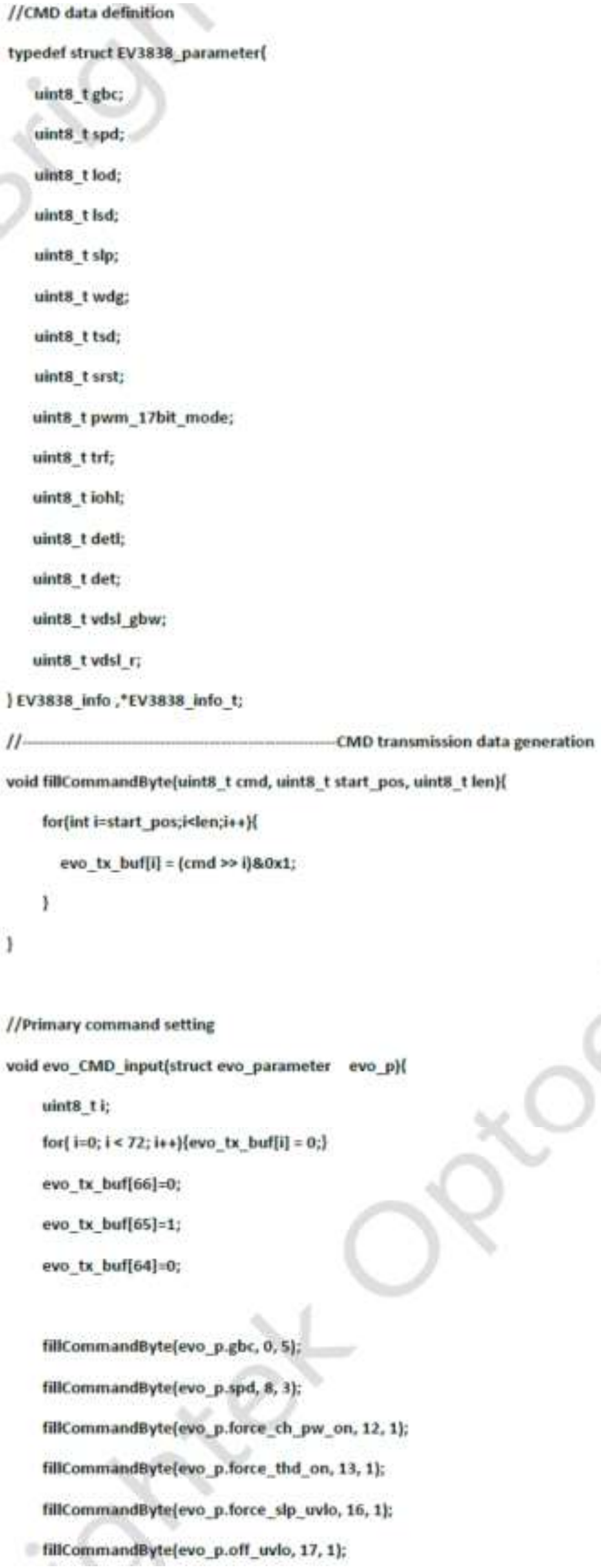
```
//---------------------------------------------------------CRC data generation
void generate_CRC2(uint8_t buffer[72]){
    uint8_t result,    i;
    int CRC_mask0 [] = {0,3,5,6,9 ,10,11,12,13,17,18,20,21,22,24,26,31,34,36,37,40,41,42,43,44, 48,49,51,52, 53,55,57,62};
    int CRC_mask1 [] = {1,4,6,7,10,11,12,13,14,18,19,21,22,23,25,27,32,35,37,38,41,42,43,44,45,49,50,52,53, 54,56,58,63};
    int CRC_mask2 [] = {0,2,3,6,7 ,8 ,9 ,10,14,15,17,18,19,21,23,28,31,33,34,37,38,39,40,41,45,46,48,49,50, 52,54,59,62};
    int CRC_mask3 [] = {1,3,4,7,8 ,9 ,10,11,15,16,18,19,20,22,24,29,32,34,35,38,39,40,41,42,46,47,49,50,51, 53,55,60,63};
    int CRC_mask4 [] = {2,4,5,8,9 ,10,11,12,16,17,19,20,21,23,25,30,33,35,36,39,40,41,42,43,47,48,50,51,52, 54,56,61};
    int len[5] = {33,33,33,33,32};
    result = 1;
    for (i=0; i<len[0]; i++){
      result = result ^ buffer[CRC_mask0[i]];
    }
    buffer[67] = result;
    result = 1;
    for (i=0; i<len[1]; i++){
      result = result ^ buffer[CRC_mask1[i]];
    }
    buffer[68] = result;
    result = 1 ^ 1;
    for (i=0; i<len[2]; i++){
      result = result ^ buffer[CRC_mask2[i]];
    }
    buffer[69] = result;
    result = 1 ^ 1;
    for (i=0; i<len[3]; i++){
      result = result ^ buffer[CRC_mask3[i]];
    }
    buffer[70] = result;
    result = 1 ;
    for (i=0; i<len[4]; i++){
      result = result ^ buffer[CRC_mask4[i]];
    }
    buffer[71] = result;
}
```

5.2 REVERENCE CODE GENERALIZATION - DATA SENDING:

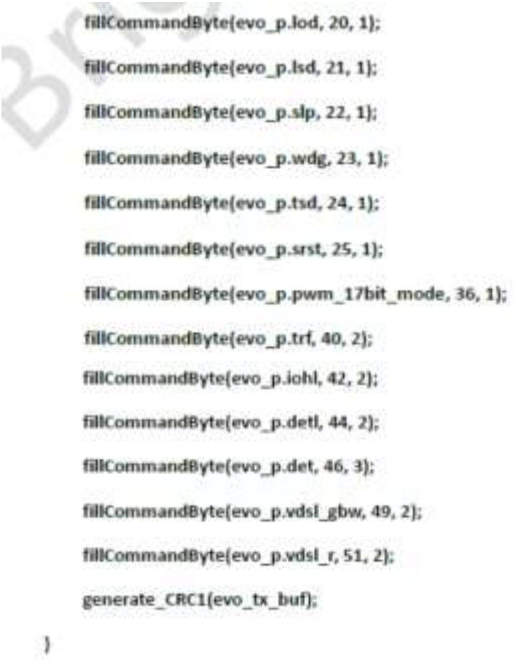


```
void EV3838_input_1bit(uint8_t b){
    if(b){
            DAI1();
    }else{
            DAI0();
    }
    Waiting(DELAY);
    CKI0();
    Waiting(DELAY);
    CKI1();
}
void Latch(void){
    DAI1();
    delay_us(200);
}

//72bit data transmission
void EV3838_input_72bit(uint8_t buffer[72]){
    int8_t i;
    for(i = 0; i<72 ; i++){
            if(buffer[71-i]==1){EV3838_input_1bit(1);}else{EV3838_input_1bit(0);}
    }
}
```

5.3 REVERENCE CODE GENERALIZATION - PRIMARY REGISTER COMMAND:

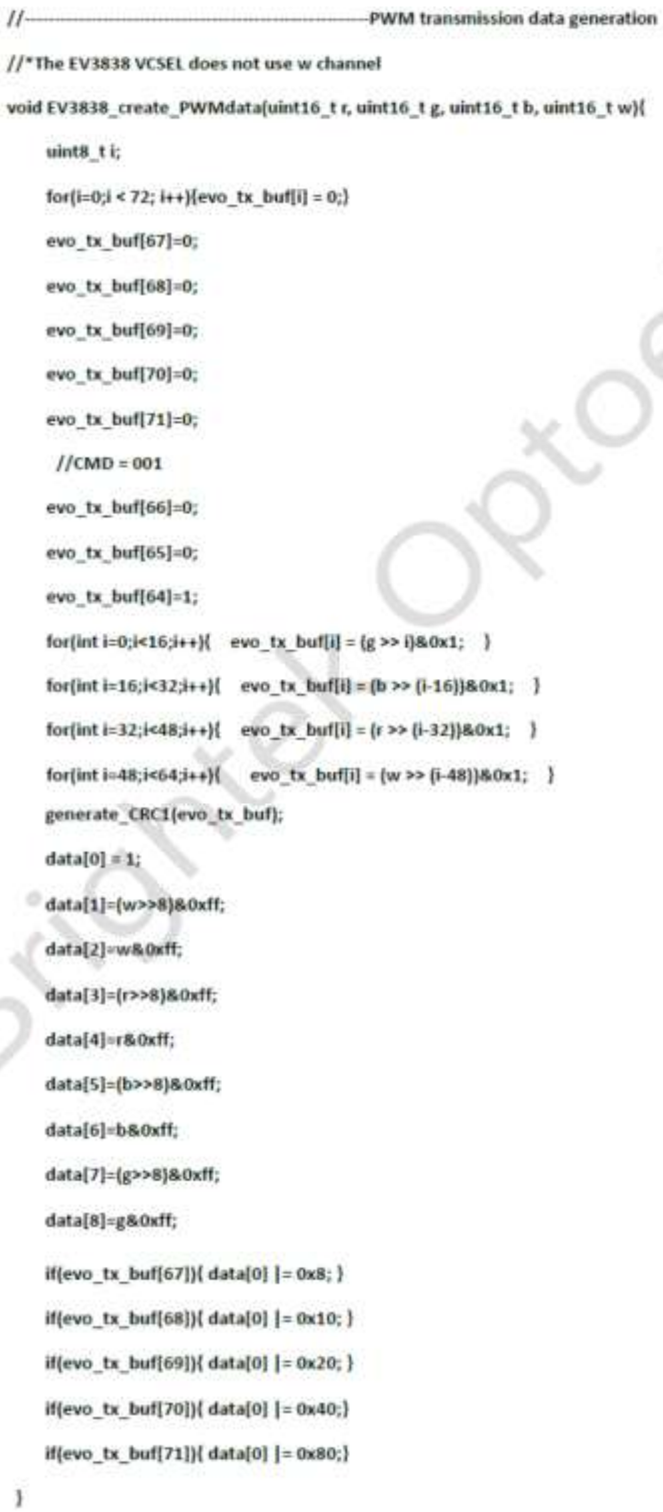


```
//CMD data definition
typedef struct EV3838_parameter{
    uint8_t gbc;
    uint8_t spd;
    uint8_t lod;
    uint8_t lsd;
    uint8_t slp;
    uint8_t wdg;
    uint8_t tsd;
    uint8_t srst;
    uint8_t pwm_17bit_mode;
    uint8_t trf;
    uint8_t iohl;
    uint8_t detl;
    uint8_t det;
    uint8_t vdsl_gbw;
    uint8_t vdsl_r;
} EV3838_info ,*EV3838_info_t;
//-----------------------------------------------CMD transmission data generation
void fillCommandByte(uint8_t cmd, uint8_t start_pos, uint8_t len){
     for(int i=start_pos;i<len;i++){
        evo_tx_buf[i] = (cmd >> i)&0x1;
     }
}


//Primary command setting
void evo_CMD_input(struct evo_parameter    evo_p){
     uint8_t i;
     for( i=0; i < 72; i++){evo_tx_buf[i] = 0;}
     evo_tx_buf[66]=0;
     evo_tx_buf[65]=1;
     evo_tx_buf[64]=0;


     fillCommandByte(evo_p.gbc, 0, 5);
     fillCommandByte(evo_p.spd, 8, 3);
     fillCommandByte(evo_p.force_ch_pw_on, 12, 1);
     fillCommandByte(evo_p.force_thd_on, 13, 1);
     fillCommandByte(evo_p.force_slp_uvlo, 16, 1);
     fillCommandByte(evo_p.off_uvlo, 17, 1);
```



```
    fillCommandByte(evo_p.lod, 20, 1);

    fillCommandByte(evo_p.lsd, 21, 1);

    fillCommandByte(evo_p.slp, 22, 1);

    fillCommandByte(evo_p.wdg, 23, 1);

    fillCommandByte(evo_p.tsd, 24, 1);

    fillCommandByte(evo_p.srst, 25, 1);

    fillCommandByte(evo_p.pwm_17bit_mode, 36, 1);

    fillCommandByte(evo_p.trf, 40, 2);

    fillCommandByte(evo_p.iohl, 42, 2);

    fillCommandByte(evo_p.detl, 44, 2);

    fillCommandByte(evo_p.det, 46, 3);

    fillCommandByte(evo_p.vdsl_gbw, 49, 2);

    fillCommandByte(evo_p.vdsl_r, 51, 2);

    generate_CRC1(evo_tx_buf);

}
```

5.4 REVERENCE CODE GENERALIZATION - PWM COMMAND:



```
//-------------------------------------------------------PWM transmission data generation

//*The EV3838 VCSEL does not use w channel

void EV3838_create_PWMdata(uint16_t r, uint16_t g, uint16_t b, uint16_t w){

    uint8_t i;

    for(i=0;i < 72; i++){evo_tx_buf[i] = 0;}

    evo_tx_buf[67]=0;

    evo_tx_buf[68]=0;

    evo_tx_buf[69]=0;

    evo_tx_buf[70]=0;

    evo_tx_buf[71]=0;

     //CMD = 001

    evo_tx_buf[66]=0;

    evo_tx_buf[65]=0;

    evo_tx_buf[64]=1;

    for(int i=0;i<16;i++){    evo_tx_buf[i] = (g >> i)&0x1;    }

    for(int i=16;i<32;i++){    evo_tx_buf[i] = (b >> (i-16))&0x1;    }

    for(int i=32;i<48;i++){    evo_tx_buf[i] = (r >> (i-32))&0x1;    }

    for(int i=48;i<64;i++){     evo_tx_buf[i] = (w >> (i-48))&0x1;    }
    generate_CRC1(evo_tx_buf);

    data[0] = 1;

    data[1]=(w>>8)&0xff;

    data[2]=w&0xff;

    data[3]=(r>>8)&0xff;

    data[4]=r&0xff;

    data[5]=(b>>8)&0xff;

    data[6]=b&0xff;

    data[7]=(g>>8)&0xff;

    data[8]=g&0xff;

    if(evo_tx_buf[67]){ data[0] |= 0x8; }

    if(evo_tx_buf[68]){ data[0] |= 0x10; }

    if(evo_tx_buf[69]){ data[0] |= 0x20; }

    if(evo_tx_buf[70]){ data[0] |= 0x40;}

    if(evo_tx_buf[71]){ data[0] |= 0x80;}

 }
```

6. Q&A:

Q1: How are VDSL_R and VDSL_GBW defined? Why is there also a setting for W channel, and how should it be applied?

A: The VDSL_R and VDSL_GBW reference voltages are defined based on the forward voltage characteristics of the R/G/B LEDs. Internally, the IC supports four channels: R, G, B, and W. However, in the EVO3838 product, only the R/G/B channels are utilized.

Q2: How should F_TRF (Bits 41:40) — Output current slew rate level be selected? What is the actual slew rate for each level?

A: F_TRF sets the output current slew rate. The four levels represent relative strength (weaker or stronger transition edges). A weaker slew rate helps reduce transient noise. It is recommended to use the default setting of 0x01 for general applications.

Q3: Does F_GBC (Bits 5:0) define the maximum IOUT as a percentage? How is it related to the 16-bit PWM brightness settings in the PWM data?

A: F_GBC is a global brightness control register for synchronously adjusting the DC current level of all R/G/B/W channels. It is a 6-bit value. In contrast, the 16-bit PWM values for each of the R/G/B/W channels are used to control the duty cycle, i.e., the relative brightness of each individual channel.

Q4: Can OTP programming of the IC be performed by the customer?

A: OTP programming is performed during the factory production stage and is not accessible to customers. All relevant OTP parameters are pre-calibrated to ensure consistency and reliability during usage, and no additional programming is required by the user.

REVISION RECORD:

Version	Date	Summary of Revision
A1.0	08/02/2025	First issued.
A1.1	07/03/2025	Updated thermal feedback register description – Table 11.
A1.2	26/03/2025	Corrected the LED pin assignment in the circuit diagram to align with the LED datasheet. Revised the description of OUT_SPD configuration in Section 4.3.6 for greater clarity.
A1.3	14/04/2025	Updated data feedback connection example and timing diagram.
A1.5	28/05/2025	P9: Modify the content of Figure 4 and add an explanation of the feedback command; P22: Adjust the data bits corresponding to G and B in Table 15.
A1.6	04/07/2025	Update PIN definition.