32-channel LED driver with automotive CAN FD Light interface

Introduction

The L99LDLH32 is a monolithic 32-channels linear current regulator specifically designed for automotive exterior O/LED rear lighting applications. The output channels are in high side configuration, and for this reason, the L99LDLH32 is suitable to drive an OLED panel with a common cathode.

The L99LDLH32 integrates a dedicated CAN FD Light communication interface, allowing high-rate data transmission (at 1 Mbit/s) and using CAN FD structure for long frames. Besides the CAN FD Light physical layer, the device integrates the protocol handler and a precise oscillator, so no external devices required to facilitate communication with Commander ECU.

The L99LDLH32 can operate in Bus mode using CAN FD Light interface or stand-alone/fail-safe mode using internal few time programmable (FTP) memory registers.

Thanks to integrated PHY and protocol handler, the device is ready for zonal architecture and the domain-oriented topology, communicating directly with the zonal controller or body computer module (BCM).

This manual aims at guiding the user through the device's functionality, giving practical guidelines for a quick design and debug. The user should consider this document to hand on the device. It is intended to complement the datasheet, functional safety manual, and other applicative documents such as the FTP programming tool, tiny board (EV-L99LDLH32 evaluation board), and any future supplemental application notes.

1 General Items

1.1 Overview

The L99LDLH32 offers a high level of flexibility thanks to its programmability through CAN FD Light communication interface. This feature supports generic platform approaches, which require a software configurability of several parameters.

As the device potentially controls safety-critical functions such as taillight, break light and turn indicators, built-in features are integrated to support a high level of functional safety on the application level. They are supplied directly from a car battery. The L99LDLH32 guarantees up to 35 V output-driving capability, and features 32 regulated current sources. It is able to provide from 1 mA up to 15 mA programmable current to drive each pixel of the O/LED panels. The current can be individually programmed using an integrated 8-bit DAC.

Each O/LED pixel can be supplied through one pre-regulator connected between the battery and the integrated current source. The pre-regulated (and the battery) voltages are internally monitored by an 8-bit ADC; the results are stored in a dedicated result register. The brightness can be adjusted separately for each channel through an 8-bit PWM exponential dimming control. An 8-bit linear global dimming can be superimposed to each channel's individual PWM. A slow turn-on and turn-off time improves the system's low noise generation performances. To simplify light function handling, the 32 channels can be grouped in two different light function groups.

Each channel can be mapped or not mapped to the direct input and through the pre-regulated external supply voltage (VPRE_REG pin). This feature ensures the highest flexibility to control lighting functions composed of different O/LED types, currents, and string lengths (see Section 5 O(LED) supply voltage regulation). Thanks to the integrated high precision oscillators, the device generates all relevant timing functions such as the PWM frequency, the duty cycle, the trigger points for ADC conversion, the phase shift, etc. No external timers are required.

In the L99LDLH32, thanks to integrating an 8-bit ADC, a complete diagnostic is also available: open load detection, short to ground detection, single LED short detection, (O)LED temperature monitoring, protection through NTC, supply voltage monitoring, device temperature monitoring, etc. Diagnostic thresholds are freely configurable for each function group. Moreover, the user can configure the desired device reaction to detected faults, ensuring maximum flexibility for the application and sophisticated auto-recovery strategies (further details in Section 4 Diagnostic and protections). The device also features the gradual output delay, which avoids contemporary turning on all the channels used for a specific light function, reducing the inrush current. To improve EMI performances further, the device implements an internal clock dithering to have a spread spectrum noise reduction. L99LDLH32 is equipped with a thermal warning (TW) and outputs thermal shutdown (TSD). The device performs an automatic thermal derating based on external NTC measurement and device junction temperature (T_J). The supply voltage range is between VS_MIN and 40 V avoiding any additional load dump protection on the power supply stage.

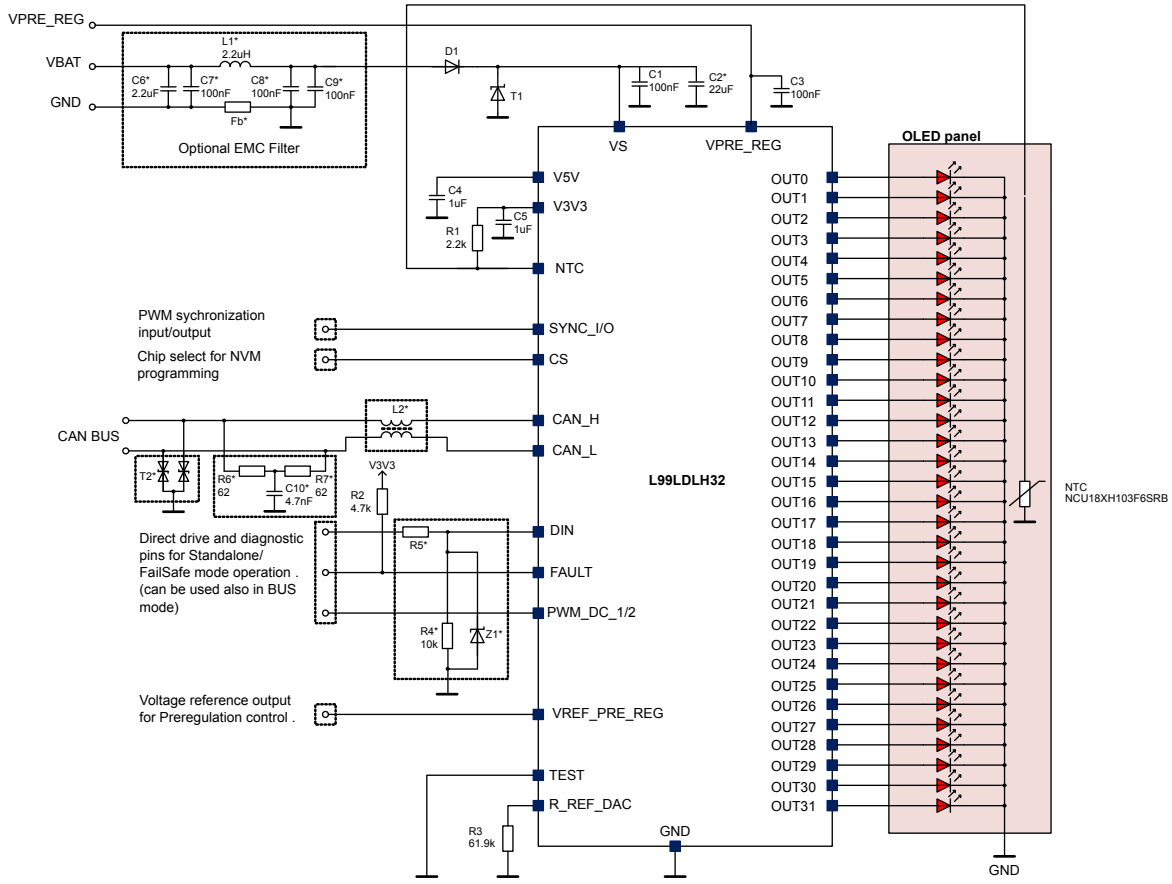
Besides the full (2x) AEC-Q100 qualification, the L99LDLH32 is also validated in the application. This either vs specific tests aimed at validating device features or vs. standard and norms like:

1. VW80000, ISO 16750-2: road vehicles - environmental conditions and testing for electrical and electronic equipment (limited only to some electrical tests)
   - long time over voltage
   - jumpstart
   - load dump test
   - slow ramp down and up of VBAT voltage
   - slow ramp down and fast ramp-up
   - reset behavior at voltage drop
   - engine cranking low voltage
   - loss of power connection
   - loss of ground connection
2. Conducted emission, according to IEC 61967-4 (150 Ω method)
3. Conducted susceptibility, according to IEC62132-4 (DPI method)
4. CISPR25, according to VW TL81000 (artificial network test class 5)
5. BCI, according to VW TL81000 (compliance with test severity L2)
6. ISO pulses (intended for the reference design): 1, 2a, 3a, 3b

1.2 Typical application schematic and description of external components

The figure below shows a typical application schematic for the L99LDLH32:

Figure 1. Typical application schematics



This paragraph comments on all the external components required by the L99LDLH32. Each external passives are reported with its label (refer to Figure 1. Typical application schematics), and the correspondent pin of the L99LDLH32 is highlighted within brackets. The optional component is labeled with "*" and enclosed by a dashed box.

- **C1*(Vs):*** using 100 nF (C1) ceramic capacitor mounted close to the device VBAT pin, and GND terminals capacitor helps suppress voltage transients originating from other drivers connected in parallel and sharing the same battery line. This capacitor is capable of suppressing only low energetic short-transient pulses.
- **C2 *(Vs):*** during an interruption of VBAT, the parallel of C1 and C2 (22uF) capacitors keeps the device active during short disconnection events.
- **C3 *(VPRE_REG):*** 100 nF is acting as a decoupling capacitor. The external preregulator has its output capacitor (in the case of dc-dc converter used for preregulation, the Cout of the dc-dc is also connected at this pin).
- ***(OUT0*-31):** used to connect the OLED (or LED) segment's cathode. If OLED panels require ESD protections, it is allowed to have i.e., capacitors between the OUTx and GND. In general, the L99LDLH32 does not require any output capacitors. It is essential to consider that the eventual capacitors placed on these pins should be chosen according to the set blanking time (as a rule of thumb, the higher the capacitance value required to sustain disconnection events, the higher blanking time should be set).

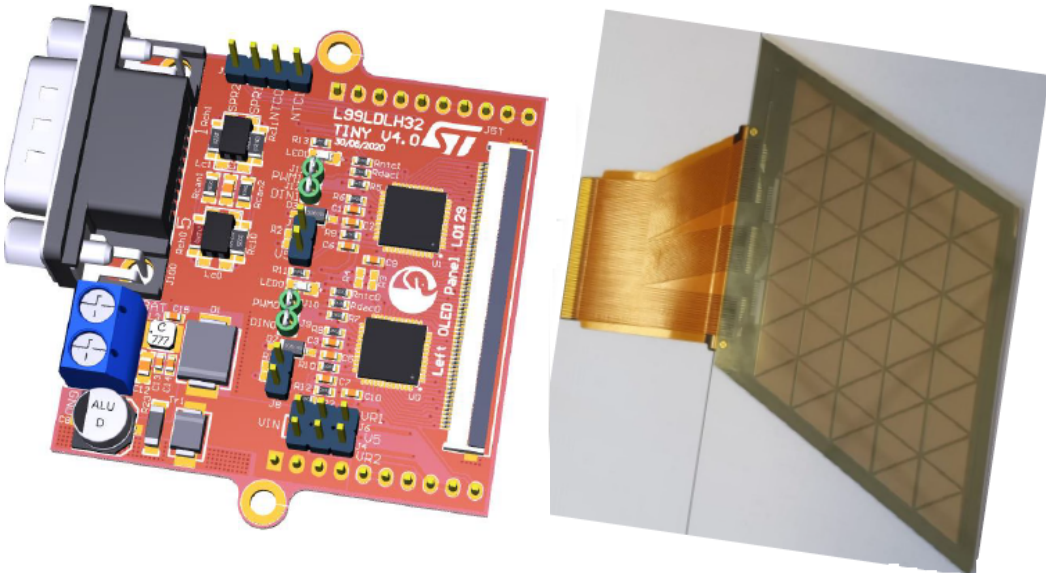
- **R3** ***(R_REF_DAC):*** this resistor is used to define the reference for the output current. It is recommended to use an E96 resistor (1% tolerance). Using 61.9 kΩ, the output current setting is in the specified range of the datasheet. In our OLED reference design, the output current ranges from 2.5 mA to 5 mA.
- ***(TEST):*** not used, connected to GND.
- **R2** ***(FAULT):*** the FAULT pin is a bidirectional I/O pin and it has an open-drain output stage, and pull-up is needed to define the logic high. If more devices are present, the FAULT pin can be paralleled. Our reference design has a 10 kΩ resistor for each device; another possibility is to have an individual resistor of 4.7 kΩ for both devices (or more). The PU resistor at the FAULT pin should be tied to 3.3 V not to exceed the abs max of the FAULT pin. To dimension, the resistor R2 refers to the parameter I_{FAULT} on table 81 of the L99LDLH32 datasheet.
- ***(CS)*** this pin is used as a chip select for the NVM programming; this pin should be pulled down to GND in regular operation.
- ***(Sync_IO):*** this pin is used to connect more devices to synchronize the PWM clock. On our tiny board, we have two L99LDLH32 connected through the SYNC_IO pins: one device acting as “Provider” providing PWM clock and the other one as “Consumer.” (refer to the chapter: Section 6 Premises on devices NVM management to define the provider and consumer).
- ***(NTC):*** L99LDLH32 has a derating function, and we can configure the starting temperature. With the suggested part number for the NTC and PU resistor R1 (2.2 kΩ), the derating function is compliant with the datasheet. We recommend using the following part number NCU18XH103F6SRB or, in case, find an alternative part with similar characteristics.
- **C4, C5** ***(V3V3, V5V)*****:** for both pins, are recommended 1 uF capacitors for the stability of the internal linear voltage regulators 3.3 V and 5 V. These capacitors also act as decoupling capacitors to sustain the fast transients generated by the internal logic of the IC.
- **D1** ***(Vs)*****:** a reverse polarity protection is required. In this case, we adopted a diode between the battery rail (VBAT) and the pin Vs. According to ISO 7637-2 / 2011, this diode also covers the test pulse 1 and 3a.
- **T1** ***(Vs)*****:** transil protection is required to be compliant with ISO7637-2 (2011) (positive pulses ISO 2a and 3b). To select the part number properly, the clamping voltage of the transil (37 V) should be lower than the abs max of the pin (40 V) and be the breakdown voltage should be higher than the 24 V required for the jump start.
- **(*DIN*):** if the L99LDLH32 is used only in Bus mode and the DIN pin is not used, connecting it to the ground is recommended. Alternatively, DIN can be directly connected to a BCM or Rear Light ECU. A further option is dimming the DIN by switching the battery line through a voltage divider (R4*, R5*). The PWM via battery line represents an optional scenario, and in this case, the couple of resistors should be dimensioned according to the maximum level of VBAT, required by the application, and the absolute maximum rating of the DIN pin (see V_{DIN} on table 71 of the DS12879). A trade-off should be met, dimensioning also according to the minimum required VBAT, being able to reach the logic high level of the DIN pin (DIN_H parameter, table 81 of the datasheet). To fulfill both conditions with a voltage divider (R4*, R5*), a zener diode Z1 in parallel to R4* is needed. Z1* should be selected according to the absolute maximum rating of DIN pin and the minimum logic level (DIN_H parameter).
- **T2* (*CAN_H, CAN_L*):** this component is intended as optional and recommended in case of connection of the L99LDLH32 directly with the BCM. In case of “traditional” approach with dedicated rear light ECU, this optional TVS for CAN bus is not necessary. The integrated CAN FD transceiver was not designed to sustain the ESD system level test according to IEC 61000-4-2. To increase the ESD system-level immunity, we recommend using the ESDCAN03 (available in SOT323 and DFN packages).
- **L2* (*CAN_H, CAN_L*):** *CAN common* mode choke [CMC] is usually used on the external CAN bus (car communication network) on the BCM side. It is an optional component related to the communication bus network. On our tiny board, we used the following part number: 1210CAN-104.
- ***R6*, R7*, C10* (CAN_H, CAN_L):*** Can FD Light termination is needed at the two ends of the line in our reference design we adopted two resistors (62 Ω each) and 4.7 nF between the common node and ground.
- ***C6*,7*,8*,9*, L1*, Fb* (Vs):*** the Pi filter connected between the battery line and ground is tied to the VS pin of the L99LDLH32 through the reverse polarity protection diode D1. This optional EMC filter was used during the evaluations allowing compliance with the conducted emission, susceptibility, and CISPR25 (test 2, 3, 4 in Section 2.1 On the initial state).

1.3 Schematic and layout of the tiny board

This paragraph shows the layout of the tiny board equipped with two L99LDLH32, driving an OLED panel with 60 segments. The figure below shows the final board and the OLED load board.

The reference design for the OLED tailight application, also foreseen a Chorus 1 M (not shown in Figure 2) acting as BCM or rear light ECU.

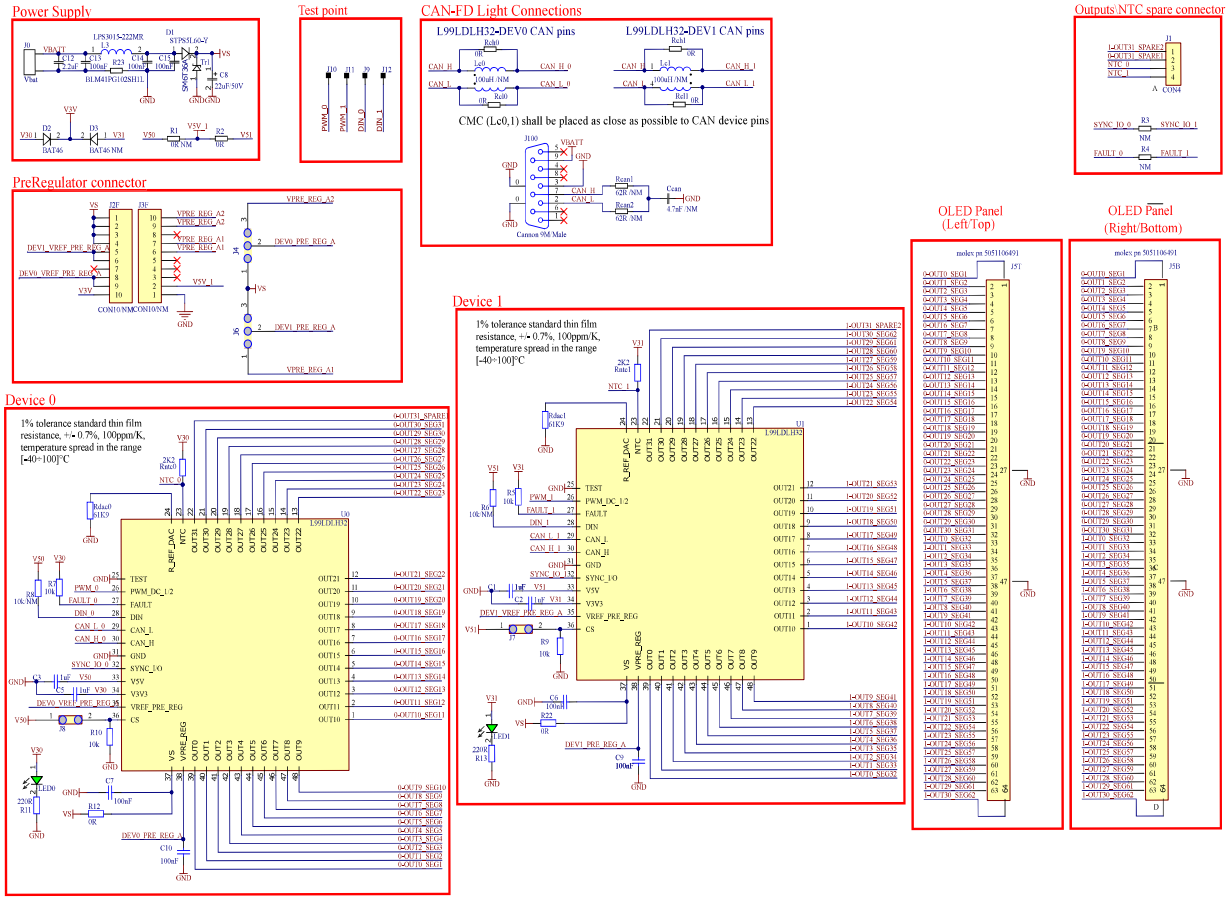
Figure 2. Tiny board with 2xL99LDLH32 and 60 segments OLED board load



This design is not intended to fit any particular form factor. This design aims to provide a compact solution, providing a way to test the board's performance. Several optimizations can be further done by the customer for final production. For example, the test point can be removed, and the capacitors' size, number, and value can be improved.

The figure below, unveil the schematic of the tiny board reference design:

Figure 3. Tiny board schematic



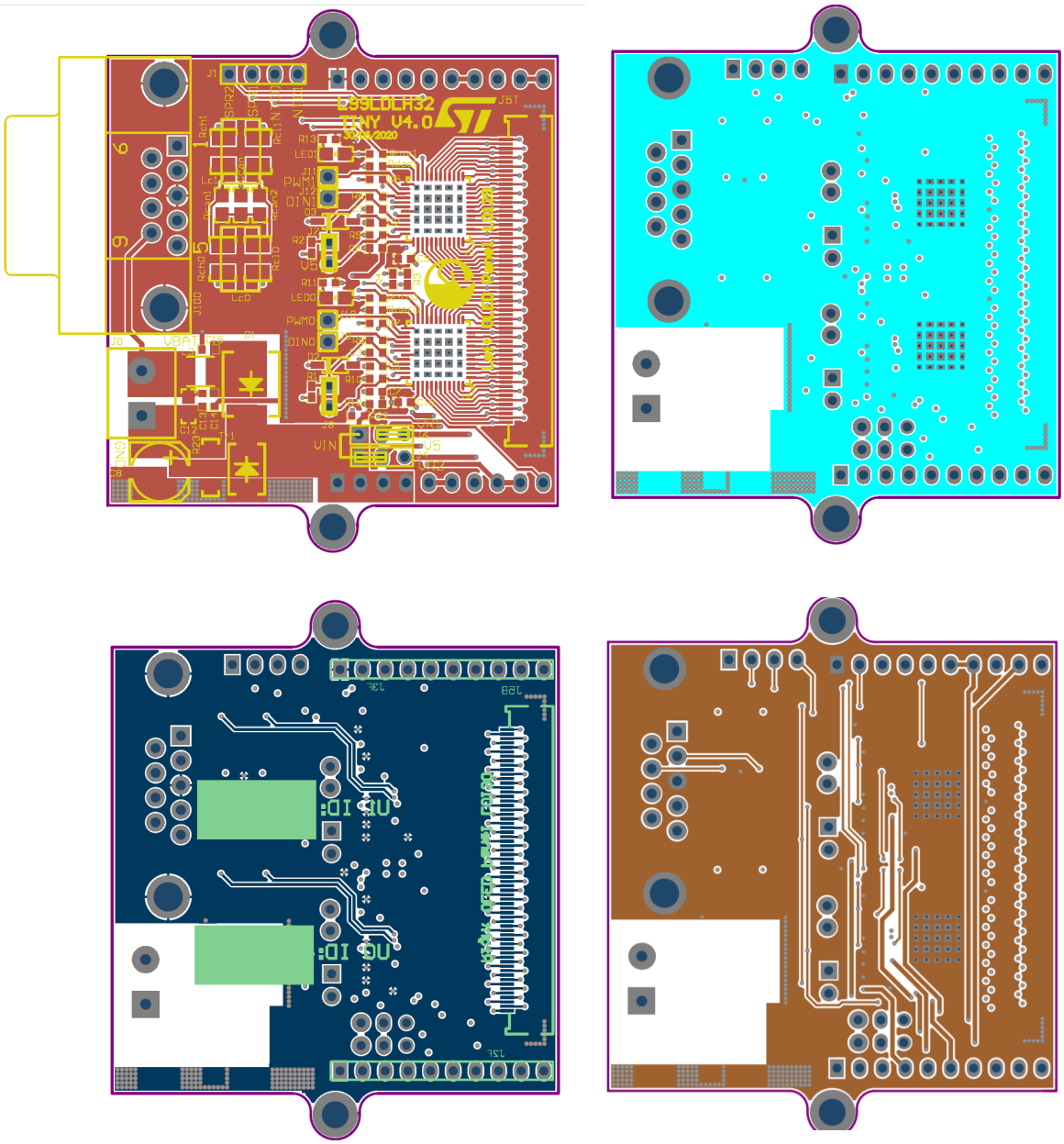
Even if the L99LDLH32 (QFN48L) package has an excellent thermal impedance, a good PCB layout can optimize the heat transfer, essential for the long-term reliability of the device.

The L99LDLH32 requires a few tricks for the layout:

- Place the capacitors close to the pin of the IC
- CMC (if required) should be placed as close as possible to the CAN_H, CAN_L pins
- It is not recommended to have a ground plane under the inductor and ferrite (for the optional EMC filter)

The tiny board is on 4 layer PCB (FR4); in the figure below, each layer is shown.

Figure 4. 4-layer PCB layout of the tiny board equipped with 2xL99LDLH32



In Figure 4, clockwise, starting from top left (red layer): top layer, signal layer 1, ground-bottom plane, and signal layer 2.

The Table 1 reports the mandatory and optional components required by L99LDLH32, according to the typical application schematic reported in Figure 1. All the designators reported in the Table 1 follow the schematic in Figure 1, and the list of components below is intended for one L99LDLH32.

Table 1. BOM

Quantity	Designator	Comment	PN
1	C2*	22 µF/50 V	EEEFK1H220P
1	C1, C3, C7*, C8*, C9*	100 µF	
1	C4, C5	1 µF	
1	C6*	2.2 µF	
1	C10	4.7 nF	
1	D1	DIODO_SMC	STPS5L60-Y
1	T1	DIODO_SMB	SM6T27A
1	T2*		ESDCAN03-2BWY
2	J5B, J5T	molex	5051106491
1	J100	Cannon 9M/Male	5504F1-09P-02A-03
1	L1*	LPS3015-222MR	LPS3015-222MR
1	L2*		1210CAN-104
1	Fb*	BLM41PG102SH1L	BLM41PG102SH1L
1	R1	2.2 kΩ	
1	R2	4.7 kΩ	
1	R3	61.9 kΩ (E96- 1% tol)	
2	R6*, R7*	62R	
1	NTC		NCU18XH103F6SRB

1.4 CISPR 25 characterization

The analysis concerning the conducted emission of the L99LDLH32 was done using the executed emission voltage method and average limits for conducted disturbances (CISPR25) specifications:

- peak detection
- average detection

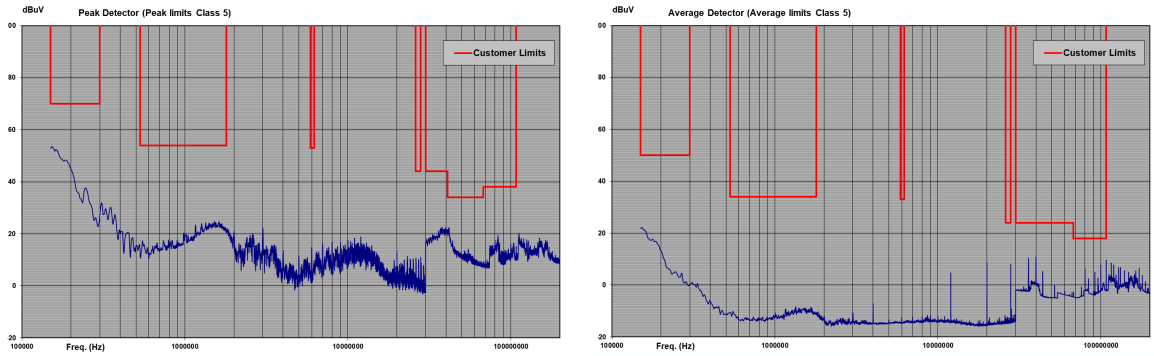
To be class 5 compliant, the peak detector should pass the peak limit for class 5 and then the average limit for class 5.

The test was conducted with VBAT = 12.5 V at room temperature. The hardware under test was the L99LDLH32 (v 4.0) Tiny board, the microcontroller Chorus 1 M Tiny board V1.0 and the load board composed of 32 OLED segments. In terms of SW settings, the operation was conducted in Bus mode, and the WD was served via CAN FD Light.

The test configuration is done according to Std CISPR25 limits (2016 edition) and VW TL-81000 AN (2018 edition).

The L99LDLH32 is compliant (with CLASS 5) with standard CISPR25 and VW TL-81000 limits. In the Figure 5 is reported the test result of the standard CISPR25 compliance.

Figure 5. CISPR25 characterization in bus mode

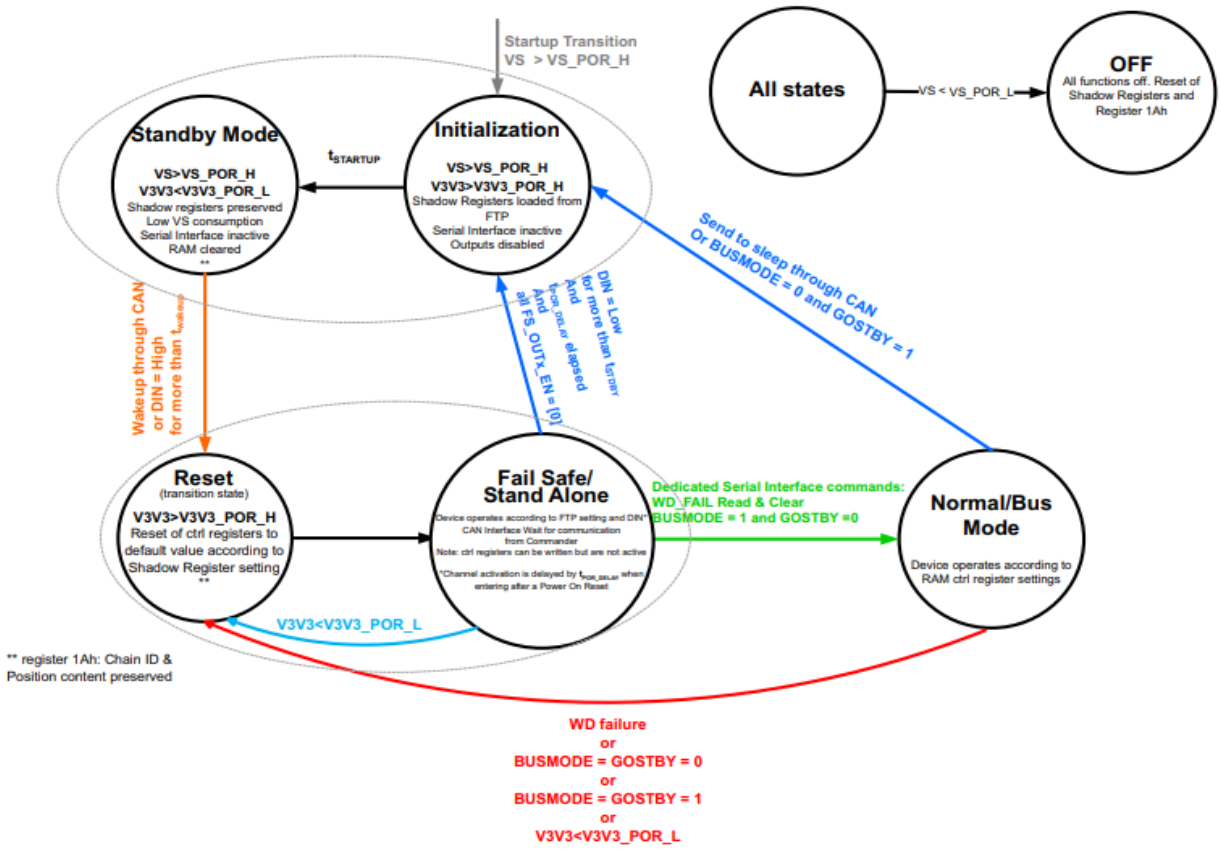


In Figure 5, on left-hand side, it is reported the characterization in the Bus mode peak detection; whilst on right-hand side, there is the characterization in the Bus mode average detection.

2 Device control

In order to operate the device L99LDLH32 in an application environment, it is necessary to have some knowledge about states into which the device can be driven and the purpose of them. The diagram below provides an overview of the state machine.

Figure 6. L99LDLH32 state machine



As soon as the VS pin of the device is driven above the VS_POR_H level, the device gets through the state machine entry point, and the logic becomes active.

From a user perspective, it is worth knowing that three states can be set on purpose, each having distinctive operating conditions:

- Standby mode: low consumption state, regulators off, no logic for control over light functions running
- Fail Safe mode: light function ON according to device NVM memory settings. Diagnostics available on device FAULT pins. This state is meant to allow device control of light function without MCU supervision
- Bus mode: this mode provides full access to device registers to program the broadest set of device operating conditions and diagnostic information retrieval. An MCU connected to the device CAN FD Light bus pins through a transceiver is mandatory in order to exchange messages (for example, commands from the CAN FD Light commander MCU side vs diagnostic information from device side). Permanence in Bus mode is granted if the device receives periodic watchdog messages from the MCU. Each one needs to toggle the previous value of the lower bit of register 19h within a programmable timeout (of 25, 50, 100, and 200 ms to reset the watchdog counter). Otherwise, the device performs a transition in Fail Safe mode (assuming that MCU is not providing proof of responsiveness, something in the system might have gone wrong).

Permanence in Bus mode is granted if the device receives periodic watchdog messages from the MCU. Each one needs to toggle the previous value of the lower bit of register 19h within a programmable timeout (of 25, 50, 100, and 200 ms to reset the watchdog counter). Otherwise, the device will perform a transition in Fail Safe mode (assuming that MCU is not providing proof of responsiveness, something in the system might have gone wrong).

Initialization state and reset state are pass-through states, in the sense that they are not permanent and the exit transition from these states is performed automatically by the device logic, no application command involved. The former state is where the transfer of NVM information into shadow registers takes place while entrance in the latter is triggered by voltage drops of the V3V3 regulator below its POR level. In this case shadow registers defaults are reloaded into control registers, then transition into Fail Safe state takes place.

2.1 On the initial state

The Figure 6 shows that power up starts the state machine entry point, which is the initialization state. The first noticeable point after supply is that depending on hardware settings, the device can then either settle in standby mode or Fail Safe mode. More in detail, after VS>VS_POR_H one of the two conditions below can happen:

- If DIN pin of the device is kept low and no CAN FD Light frames are sent, the device settles in standby mode as soon as VS>VS_POR_H and remains in such state for a time startup (needed for initialization phase, max 650 µs)
- If DIN pin is kept high or a determined wake up frame (WUP) sent through the CAN FD Light bus is recognized, then the device performs a transition in Fail Safe mode

For the second case, permanence in Fail Safe mode after transition from standby is granted only if the DIN pin is kept high or if one of the device channels is kept ON by means of the FS_OUT_ENx option programmed into the device NVM memory. If conditions for Fail Safe permanence are not being held, the device performs a transition back in standby (passing through the initialization state) within the time specified at the tPOR_DELAY field within NVM (delay ranging from 0 to 100 ms depending on tPOR_DELAY programmed value).

2.2 Transition between states and related commands

This section shows some examples of device transitions and how they are supposed to be performed.

For the end user, device transitions are classified in two categories depending on whether the transition is initiated by a CAN FD Light command or not.

State transitions outlined within the table below are initiated through CAN FD Light commands.

Table 2. State transitions (with CAN FD Light commands)

Transition type	Unicast frame payload content ([Opcode | Address], [Data byte]) (1)	Comments
Fail Safe to Bus mode	Sequence of two commands: [AEh], [00008000] and [18h],[00000001h]	Payload fitting within the CAN FD Light frame structure outlined below
Bus mode to Fail Safe	[18h], [00000000h] or [18h], [00000011h]	
Bus mode to Standby (through init state)	[18h], [00000010h]	Broadcast goto stdby possible as well (see Figure 14 and Figure 15)

1. *Colon between address and databyte stands for concatenation of bit strings.*

Each L99LDLH32 device replies with a CAN FD Light frame containing GSB to each of these commands.

The GSB byte content is set within the payload of the response and contains information related to the current device status (check Section 6.3 for more detail).

Below the state transitions that are started without CAN FD Light commands.

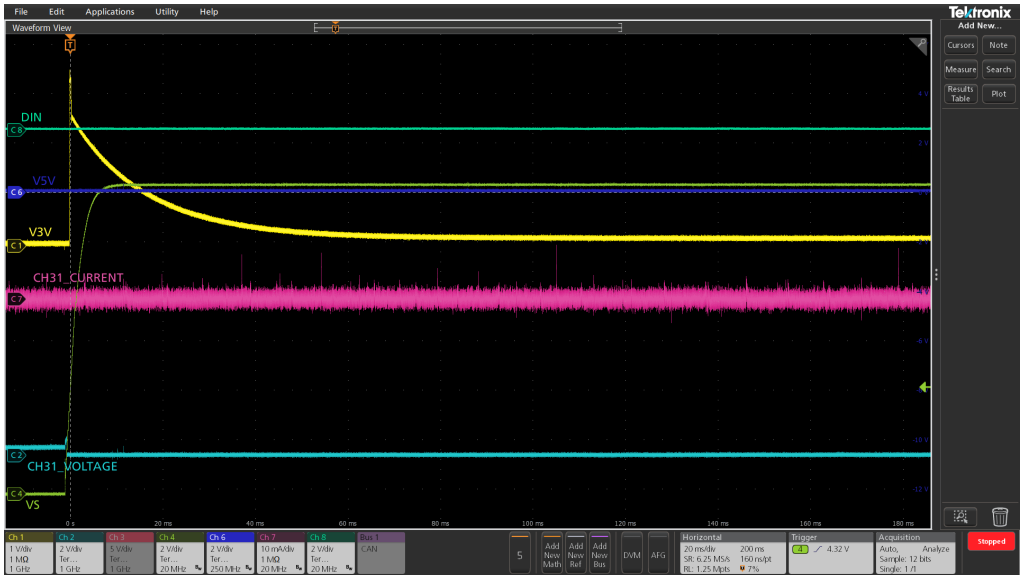
Table 3. State transitions (without CAN FD Light commands)

Transition type	Action for transition initiation	Comments
Standby to Fail Safe	CAN FD wake up pattern or DIN low to high transition	After transition, permanence in Fail Safe mode needs DIN high or FS_OUT_ENx (in NVM, row 1,2, 3, 4 column 9) settings on at least one channel
Fail Safe to standby (through init state)	DIN low and no output channel set with FS_OUT_ENx	DIN needs to be kept low for more than tSTDBY and transition starts after tPOR_DELAY
Bus mode to Fail Safe	Watchdog expiration	Watchdog expiration window can be set to 25, 50, 100 ,200 ms in device NVM

State machine transitions examples:

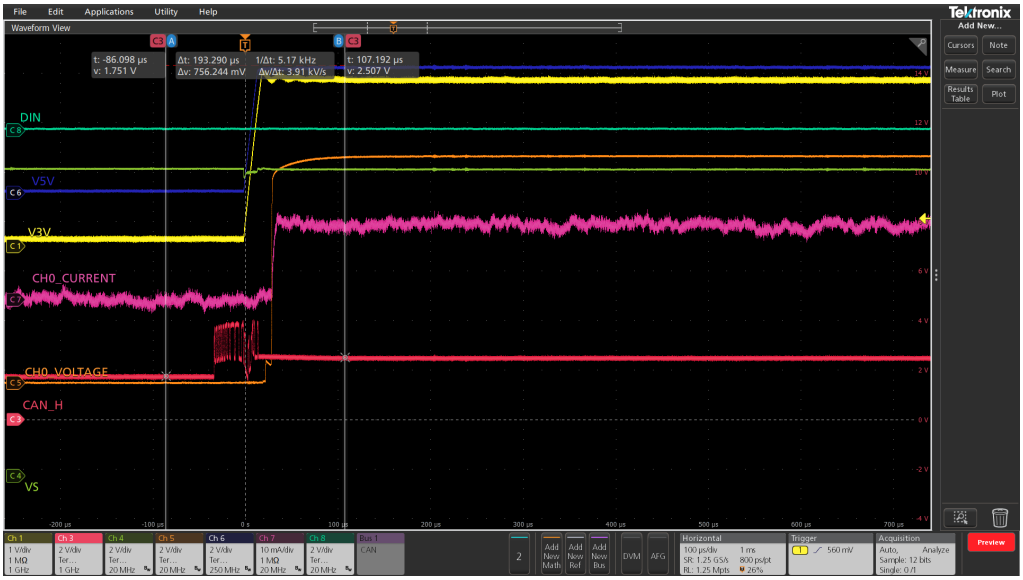
- Device startup: performs init → stdby transition after VS>VS_POR_H

Figure 7. Startup transition (DIN was kept low)



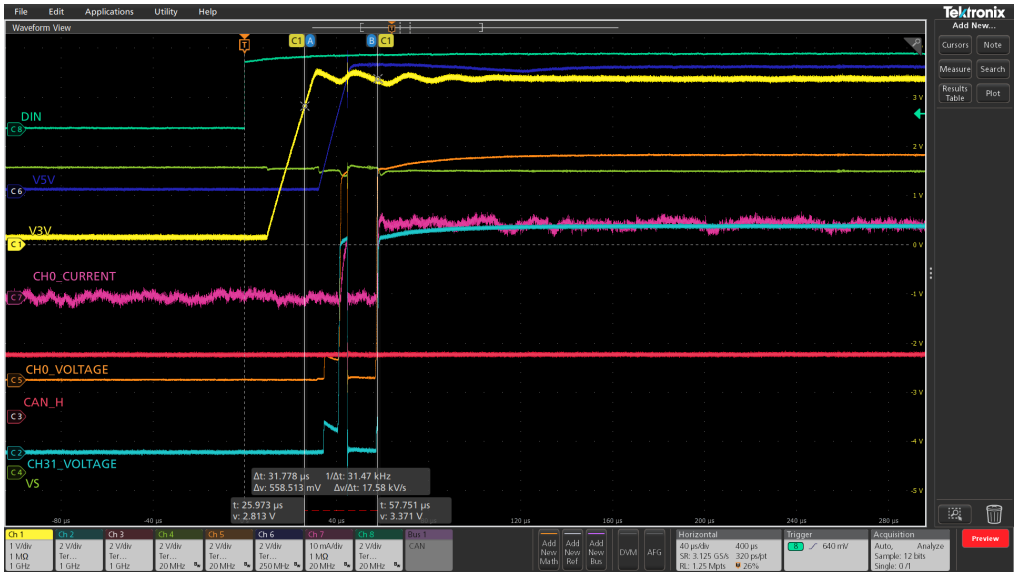
- Wake up by CAN FD frame : stdby -> Fail Safe

Figure 8. Device transition from stdby to Fail Safe after wakeup frame



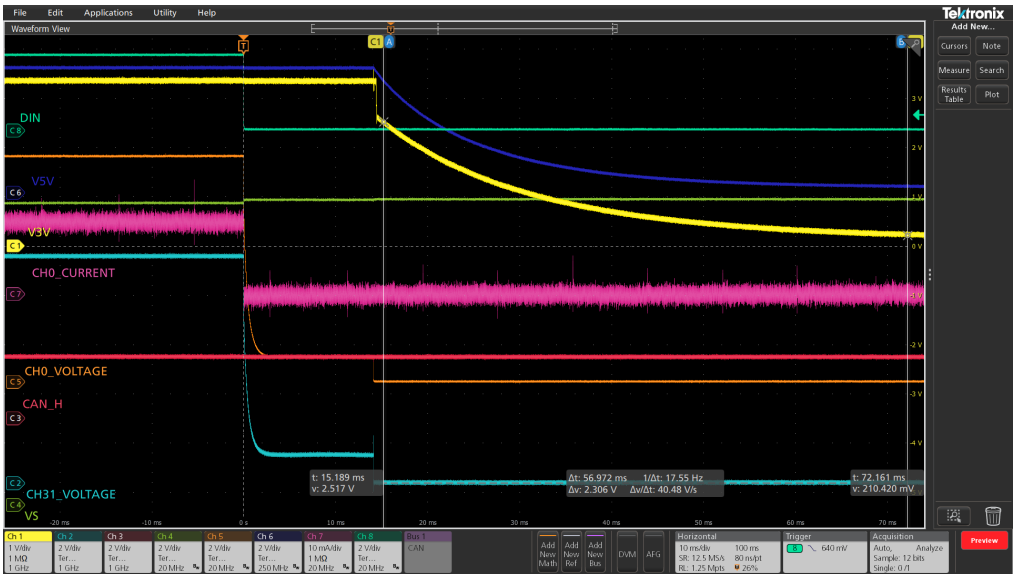
- Wake up through DIN : stdby → Fail Safe

Figure 9. DIN startup, 25 µs startup time



- Fail Safe → stdby due to DIN falling edge:

Figure 10. Transition to stdby after DIN high to low transition

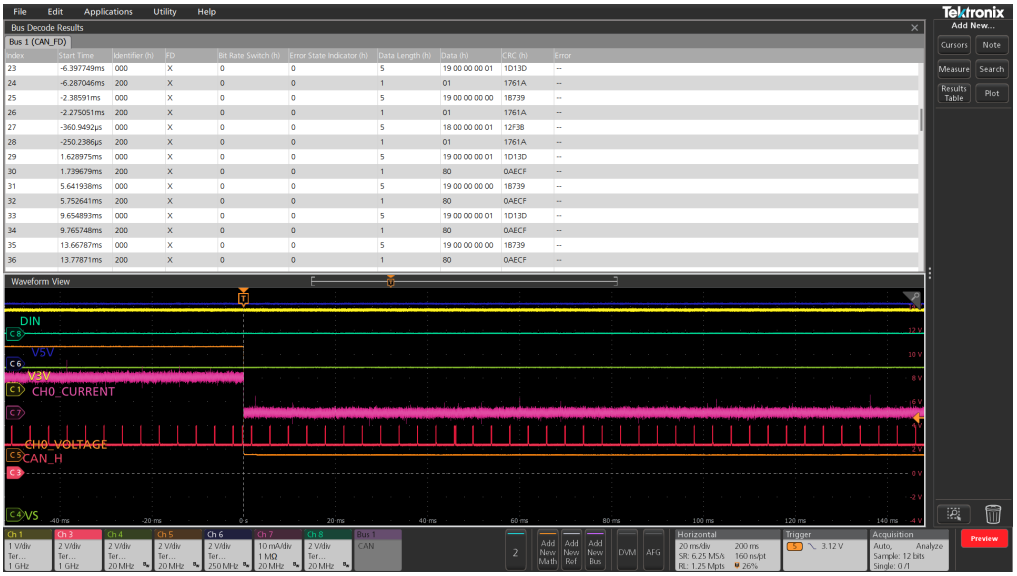


- Fail Safe mode to Bus mode through CAN FD light command:

In the example below:

- **Fail Safe :** channel0 is set permanently on with 15 mA current setting
- **Bus mode :** channel0 is off

Figure 11. Fail Safe to Bus mode transition (on CAN FD Light command frame n.27)



- Bus mode to Fail Safe (through reset) after CAN FD Light command

Figure 12. **Bus mode to Fail Safe transition (GO_STDBY=1 in register 18h)**

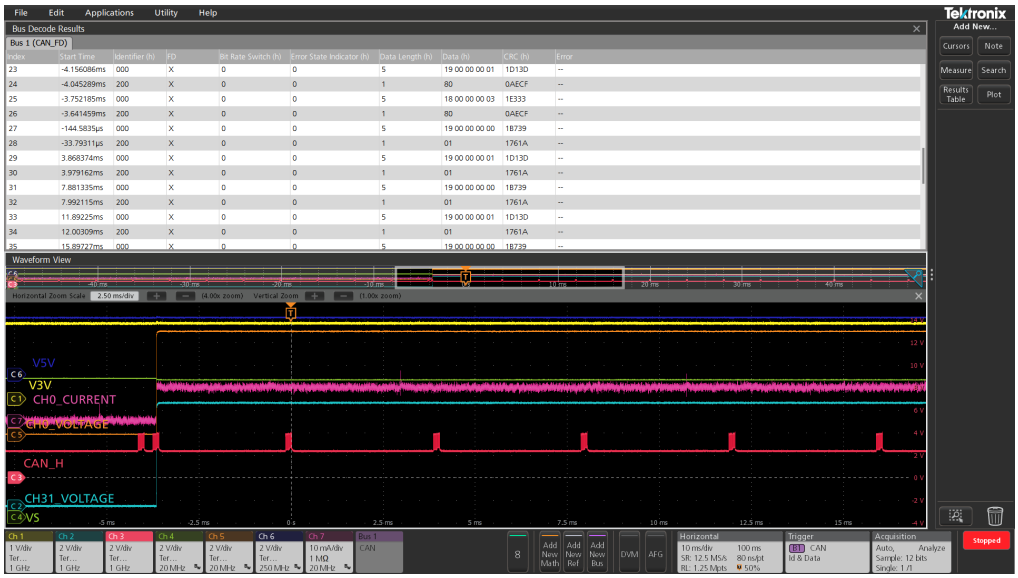
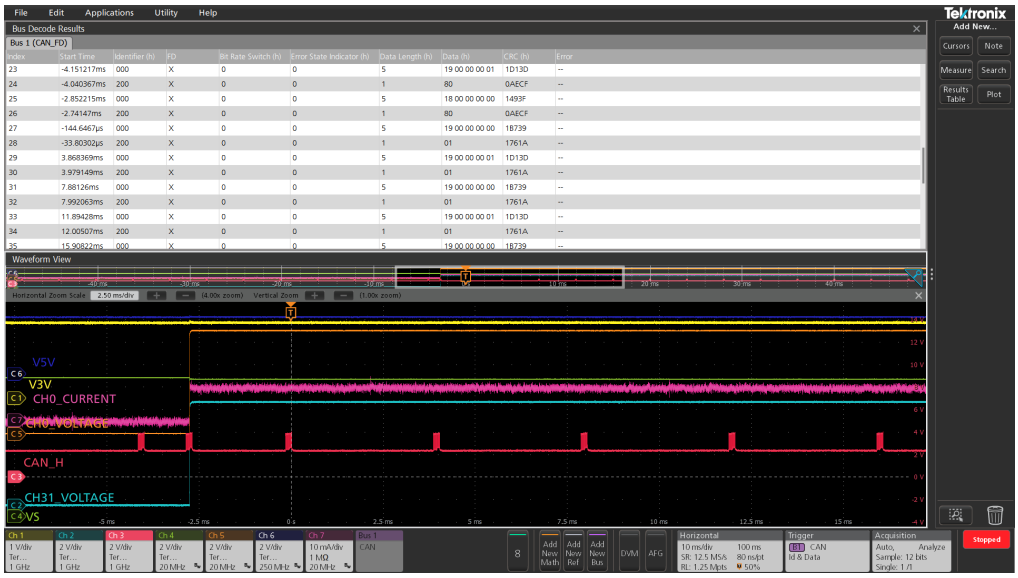


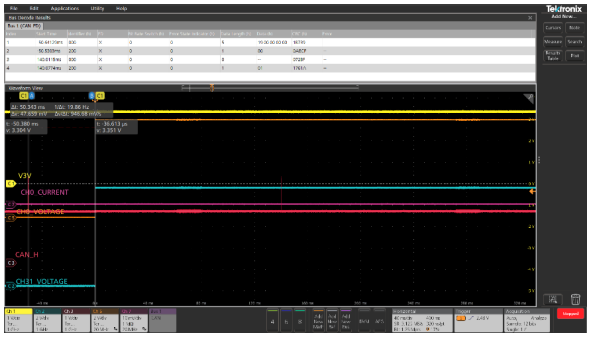
Figure 13. **Bus mode to Fail Safe transition (GO_STDBY=0 in register 18h)**



- Transition from Bus mode to Fail Safe (through reset) after watchdog expiration

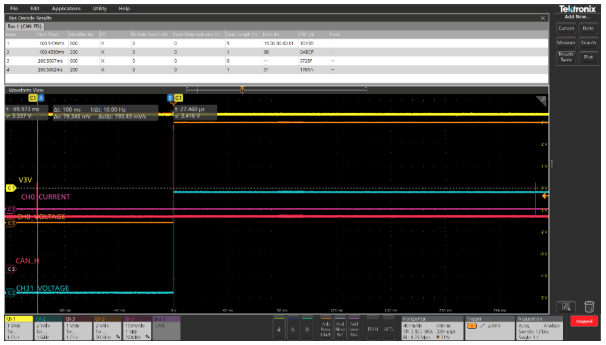
Below Bus mode to Fail Safe transition after tWD expiration (tWD = 50 µs and tWD = 100 µs):

Figure 14. WD settings = 00



*Channel voltages traced; current probe disconnected

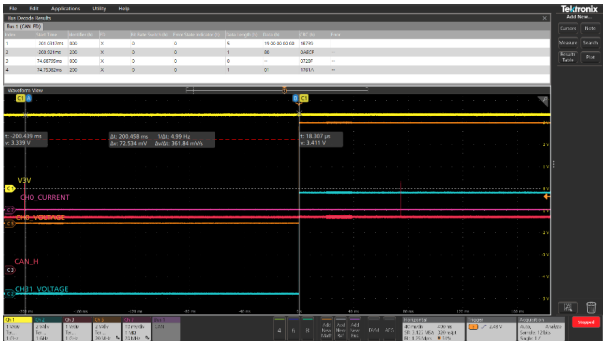
Figure 15. WD setting = 01



*Channel voltages traced; current probe disconnected

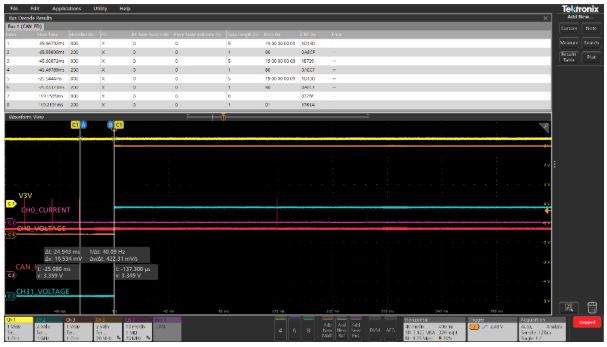
Below Bus mode to Fail Safe transition after tWD expiration (tWD = 200 µs and tWD = 25 µs):

Figure 16. WD settings = 10



*Channel voltages traced; current probe disconnected

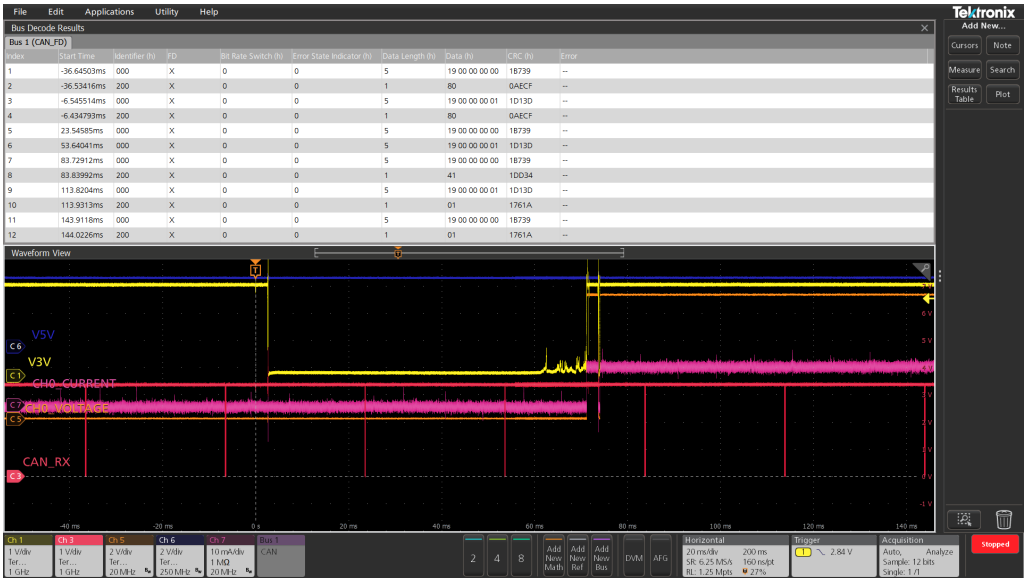
Figure 17. WD settings = 11



*Channel voltages traced; current probe disconnected

- Transition from Bus mode to Fail Safe (through reset) after V3V3 regulator drop

Figure 18. From Bus mode to Fail Safe after V3V3 drop (GSB = 80 h for Bus mode, GSB = 41,01 for Fail Safe)



- Bus mode to standby transition (through initialization) after CAN FD Light command

Figure 19. Unicast goto stdby

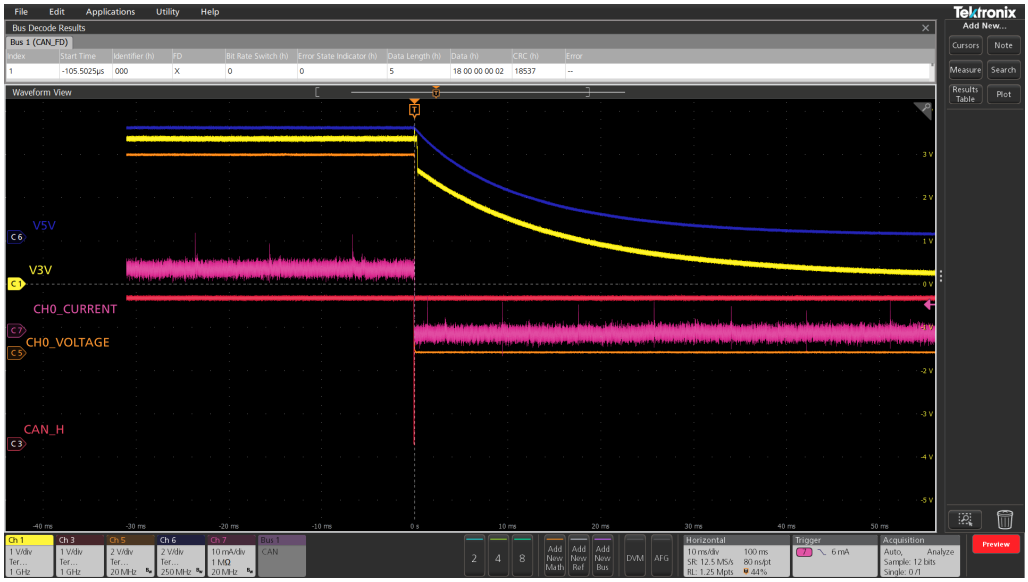
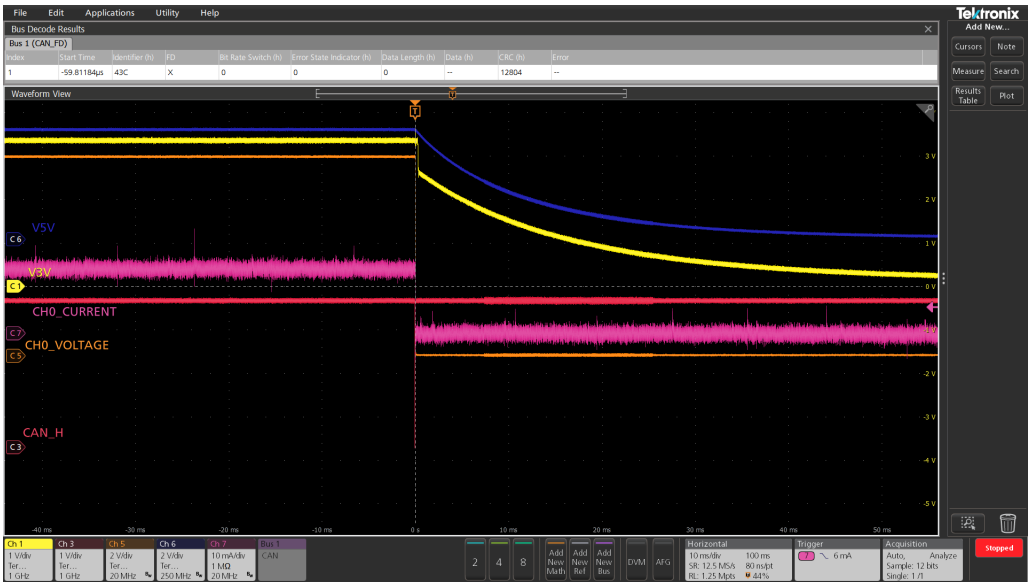


Figure 20. **Broadcast goto stdby**



3 LED brightness control

Led brightness depends on led current intensity and led dimming through PWM. Both of these parameters can be controlled writing the settings into the user registers when operating in Bus mode or in NVM for Fail Safe mode operation.

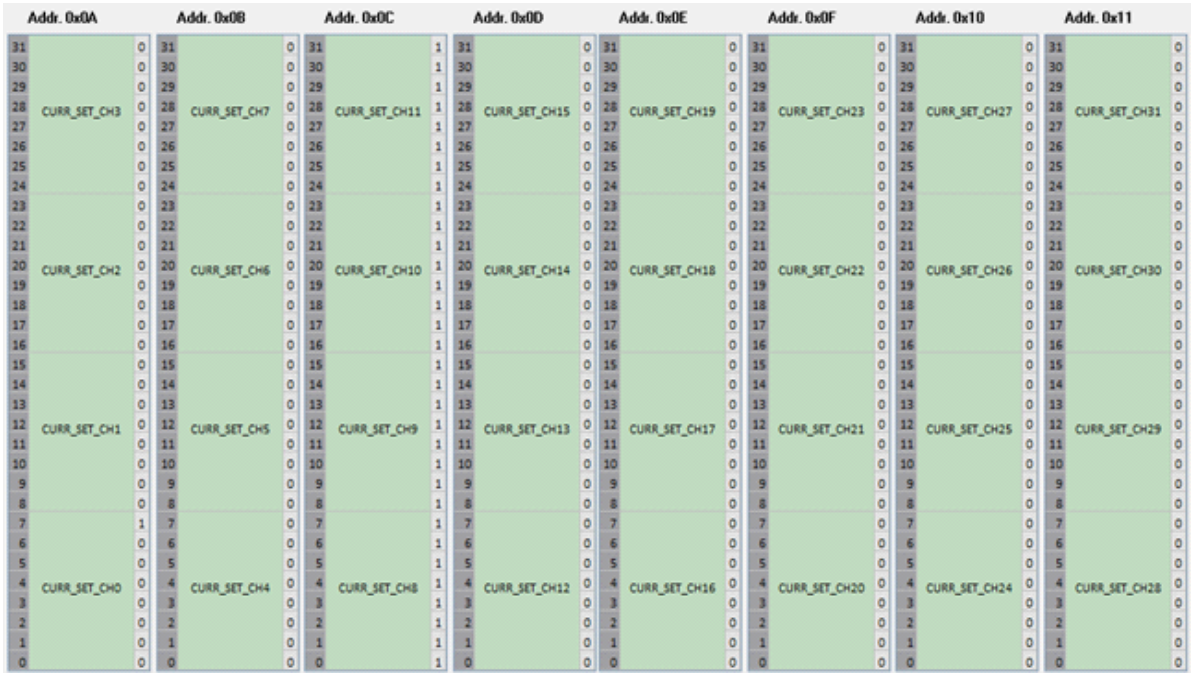
3.1 Single device led current intensity setting

Each channel can be set with a specific current level that is active during the ON phase of the duty cycle during PWM dimming. Such level can be set within the range from 1 mA to 15 mA with 8-bit granularity (for example, 256 possible levels).

3.1.1 LED current intensity in Bus mode

Current intensity settings in Bus mode operation are mapped on registers addresses (ranging from 0x0A to 0x11) with channel allocation as outlined below.

Figure 21. Channel current setting in Bus mode - register mapping



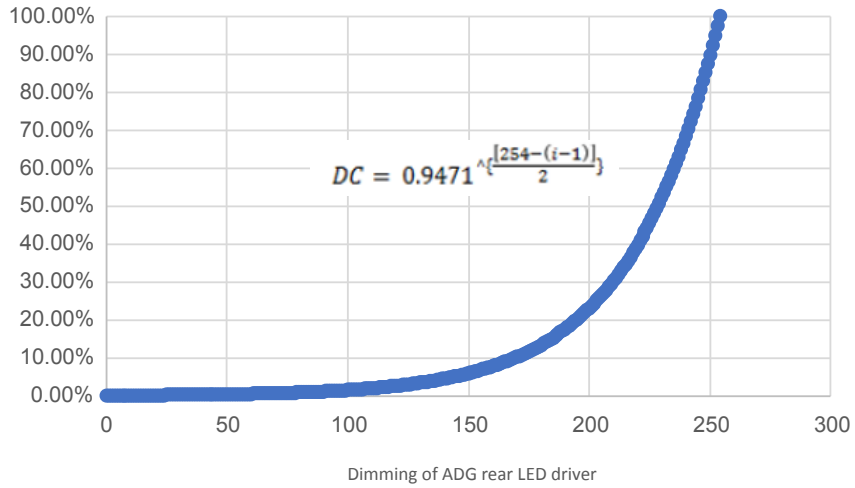
3.1.2 LED current intensity in Fail Safe mode

Current intensity for Fail Safe mode operation needs to be set within the internal NVM of the device, at rows from 1 up to 4, columns from 0 up to 7 (check the NVM fields layout).

3.2 Single device led pwm duty cycle setting

In L99LDLH32, each channel can be individually dimmed on an 8-bit exponential duty cycle to adjust LED brightness change to human eye light perception in a way that gets perceived as linear. The exponential law used to calculate the dimming steps is the following:

Figure 22. Duty cycle vs dimming step



Where "i", 1 ≤ i ≤ 255, α ≈ 0.9471, N = 254. For the step number "i" = 0, DC is set to "0". The function of duty cycle vs 8-bit PWM_DUTY_CHx setting is shown in the following graph. The duty cycle is linked to the tPWM_ON_OFF (reported in the DS) and the programmed PWM_FREQ. The minimum duty cycle is clamped to 0.1% for the frequency range 200 Hz – 400 Hz. (for PWM_FREQ>400 Hz, tPWM_ON_OFF is 2.5 µs).

PWM frequency options are outlined below:

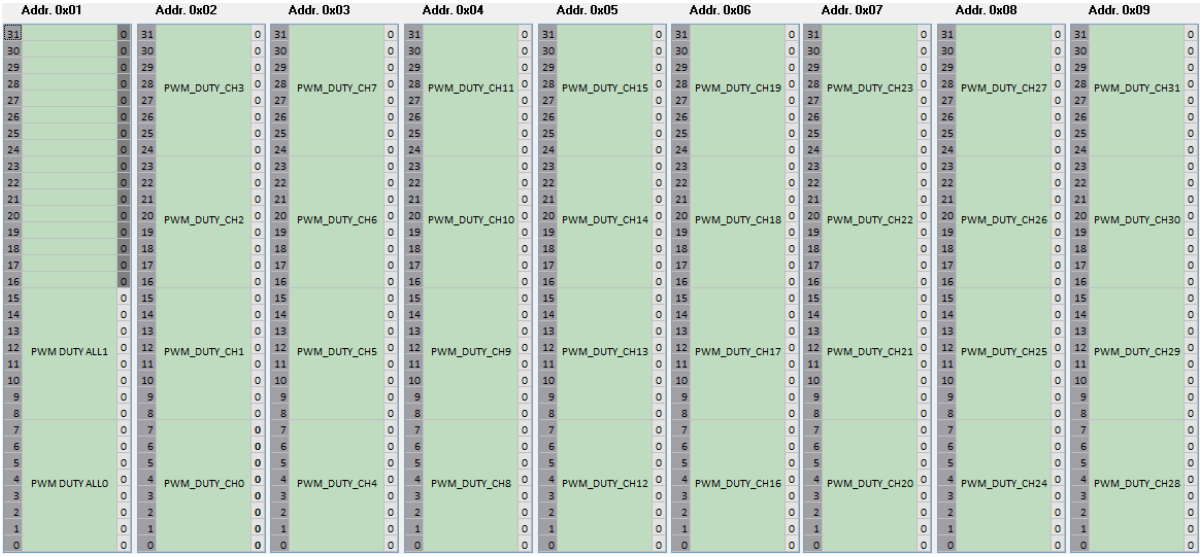
Table 4. PWM frequency range

PWM_FREQ_DINx [from 2 to 0]			PWM_FREQ [Hz]
bit 2	bit 1	bit 0	
0	0	0	200
0	0	1	300
0	1	0	400
0	1	1	500
1	0	0	700
1	0	1	1000
1	1	0	1200
1	1	1	1400

3.2.1 Duty cycle setting in Bus mode

When operating in Bus mode, duty cycle can be set channel by channel individually through registers in the range 02h to 09h:

Figure 23. Channel duty cycle setting in Bus mode - register mapping



At register address 01h, PWM_DUTY_ALL0 field allows us to set a duty cycle multiplier for all channels belonging to the DIN0 group (internally coded as group 01). This allows simultaneous dimming for all channels belonging to the same group.

Aside from programmed duty cycle control through the device logic, it is possible to drive the PWM duty cycle through a square wave (0 to 3V domain) applied to the DIN pin of the device. This applies only to channels mapped to the DIN0 group (mapping to group 01, which can be set through the DIN_MAP register) once the DIN0_ENABLE (bit 18 at register 17h) is set.

PWM frequency options outlined in the previous section are to be set at register address 15h (bits 8 to 13) when operating in Bus mode:

Figure 24. PWM frequency bits

The figures below report some examples:

Figure 25. Bus mode, Fpwm = 200 Hz, duty cycle = 49%, Current = 15 mA

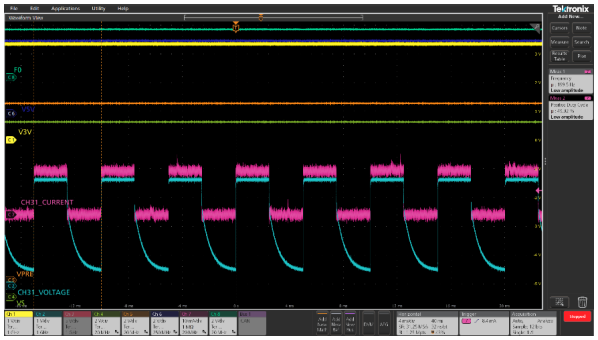


Figure 26. Bus mode, Fpwm = 700 Hz, duty cycle = 25%, Current = 15 mA

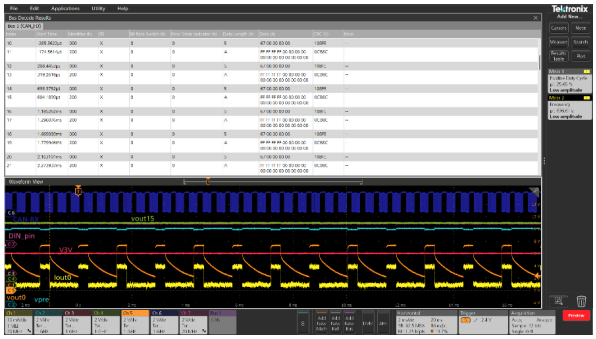


Figure 27. Bus mode, Fpwm = 700 Hz, duty cycle = 95%, Current = 15 mA

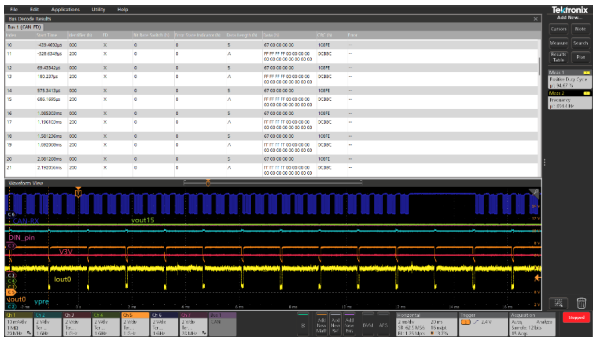
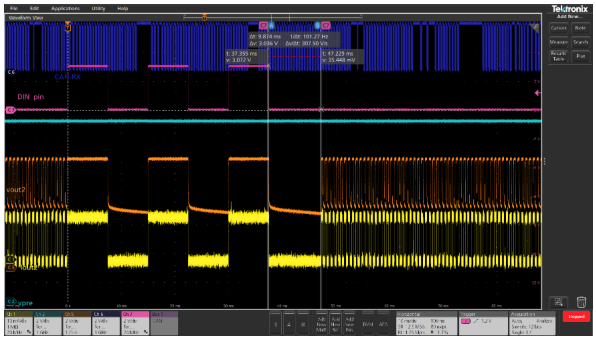


Figure 28. Bus mode, Fpwm = 1.4 kHz, duty cycle = 50%, Current = 15 mA, DIN_EN = 1



The case in the last example shows a condition in which DIN_EN is set in order to synchronize the leds PWM control with a 0.3 V square wave control signal applied to the DIN pin. Device control can be broken down in 3 windows:

- Initially DIN pin is low and the device operates PWM through the internal logic in accordance with the settings (Fpwm = 1.4 kHz, DC = 50%, current = 15 mA)
- Since DIN_EN = 1, as soon as DIN raises and starts toggling it takes over the output control. This works as long as the low of the square wave imposed through DIN is shorter than 10 ms. If DIN is held low for more than 10 ms after a high to low transition the logic releases DIN control and give it back to the internal PWM engine.

3.2.2 Duty cycle setting in Fail Safe

When operating in Fail Safe mode duty cycle settings need to be programmed in NVM before operation along with the PWM_FS_ALL_EN0=1 bit setting in NVM. These DC percentages are global for all channels mapped to DIN0 group (to be selected at rows from 1 to 4 and columns 10 and 11 of NVM, for NVM fields layout check related annex). The two duty cycle values set in NVM (PWM_DUTY_ALL0, PWM_DUTY_ALL_ALT0) can be selectively activated depending on whether the level of the PWM_DC_1/2 pin is low or high (0.3 V). Operation with programmed duty cycle values in Fail Safe mode require DIN0 pin to be high as enabler.

100% DC operation can be achieved equivalently by:

- Programming maximum duty cycle (FFh) within the previously mentioned NVM cells after indications above
- with PWM_FS_ALL_EN0=0 and DIN pin high for DIN_MAP01 channels (for example, channels belonging to the DIN0 group)
- Setting FS_OUT_ENABLE=1 regardless of channel/DIN group

3.2.3 Fail Safe dimming in provider and consumer configuration

L99LDLH32 has a built-in mechanism to establish a phase shift relationship among several devices during PWM operation. For this purpose, one of the L99LDLH32 devices (the so called “Provider”) is required to provide a common clock reference, through a scaled version of the 20 MHz oscillator (down to approximately 4.17 kHz) propagated to the SYNC_IO pin (“Provider mode” is active when the NVM bit called SYNC I/O_P/C, shorthand for provider and consumer setting, is set to zero. This is also the default value for newly deployed devices). Once the provider L99LDLH32 device is set, all other devices, subjected to the reference clock, are supposed to be connected to the SYNC_I/O pin and initialized as consumers (through NVM SYNC I/O_P/C=1). Each consumer device is supposed to set with the related PWM phase shift in NVM (PHASE_DEV bits in NVM) according to the following options:

Table 5. PWM phase shift (in consumer mode)

PHASE_xDEV [from 2 to 0]			PWM_PHASE_SHIFT [µs]
bit 2	bit 1	bit 0	
0	0	0	0
0	0	1	15
0	1	0	30
0	1	1	45
1	0	0	60
1	0	1	75
1	1	0	90
1	1	1	105

Note: Each of the consumer devices, connected to the provider, is subject to a boot sequence after power up. At the end, the phase relation towards the provider is different from 0 despite PHASE_DEV=0. Therefore, all other non-zero phase shift options are assumed to be relative to the default one after the startup. To have a known initial phase relation, the internal PWM counter can be reset by PWM_SYNC bit on each device (otherwise, the initial phase shift between devices is not known as there are some differences between devices, such us V3V3 startup).

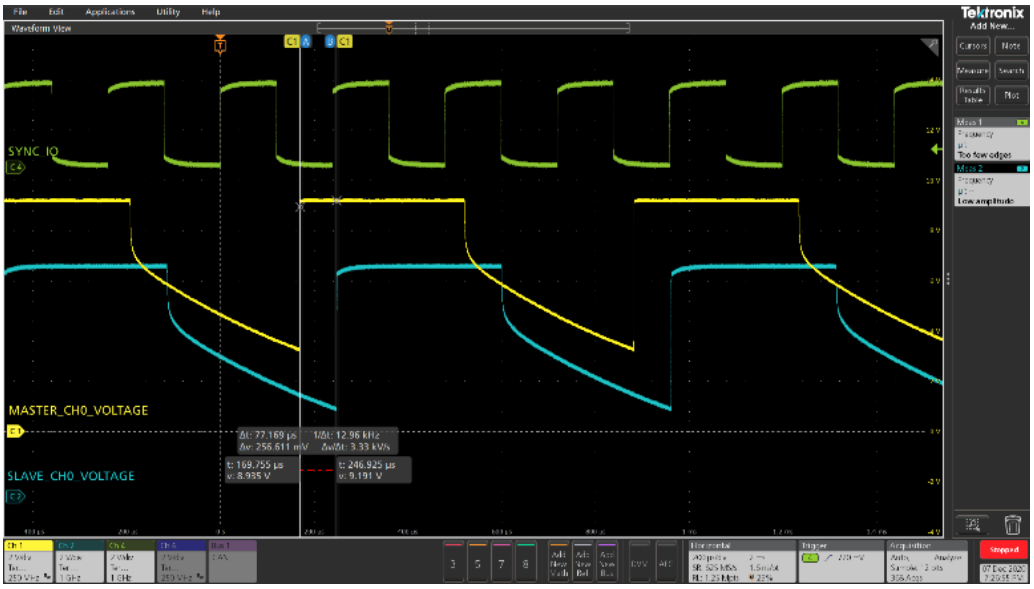
Example of Hardware setup:

- 1 board 2 devices active
- DEV1=Provider (ResponderID 4); SYNC I/O_P/C=0 (as by default)
- DEV0=Consumer (ResponderID 0); SYNC I/O_P/C=1 (in FTP)
- DEV0, DEV1 SYNC_IO pins connected
- DEV1_PreregA=DEV0_PreregA=DEV0=VS
- VS=12 V
- CH0, Dev0 and CH0, DEV1 running at 200 Hz, DC=49% and Current=15 mA for shift comparison

After SYNC I/O_P/C connection and apply phase shift options to the consumer device and observe output waveform shift increments according to specification.

Below starting point with SYNC_IO frequency provided by provider device and with both provider and consumer running PWM at 200 Hz and 0 us phase shift setting (intrinsic phase shift after devices switch on):

Figure 29. **SYNC_IO (at 200 Hz and 0 µs phase shift setting)**



Here below the focus on default time offset (77 µs) between provider and consumer device at consumer phase shift setting 000:

Figure 30. **Provider and consumer device (offset 77 µs)**

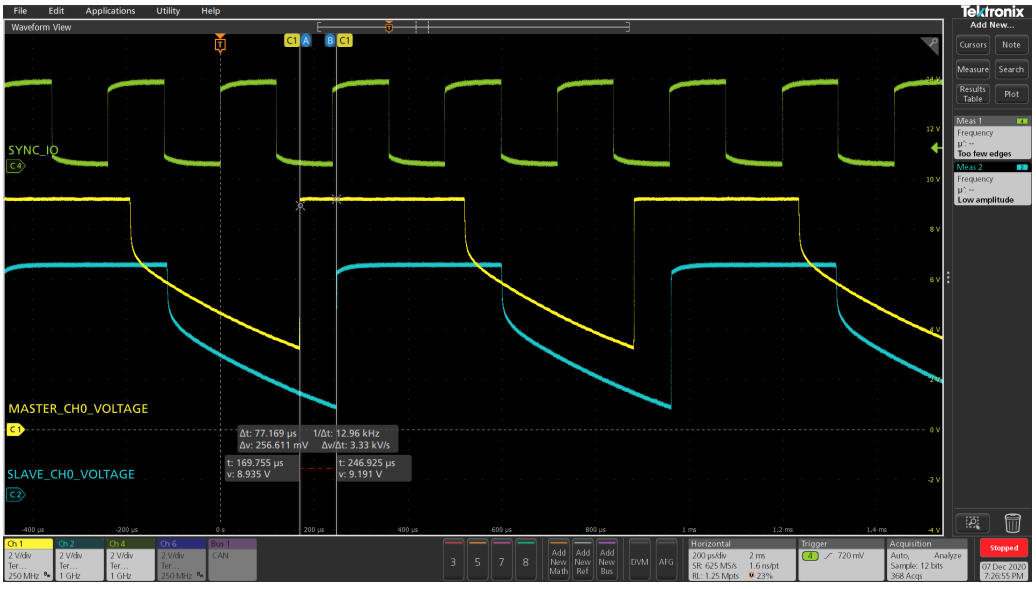


Figure 31. **Phase shift 000 (77 µs offset)**

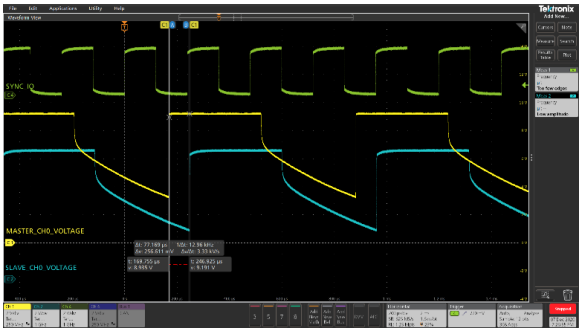


Figure 32. **Phase shift 001 (+15 µs specification target)**

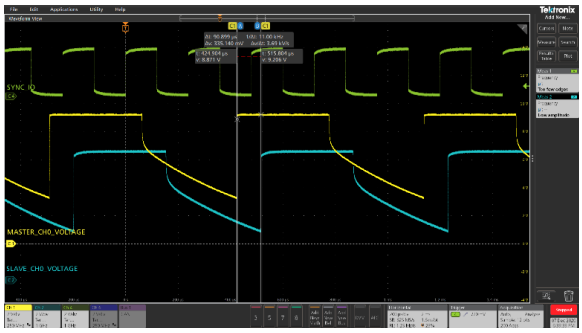


Figure 33. **Phase shift 010 (+30 µs specification target)**

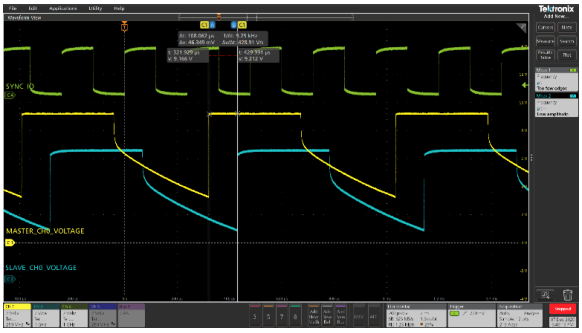


Figure 34. **Phase shift 011 (+45 µs specification target)**

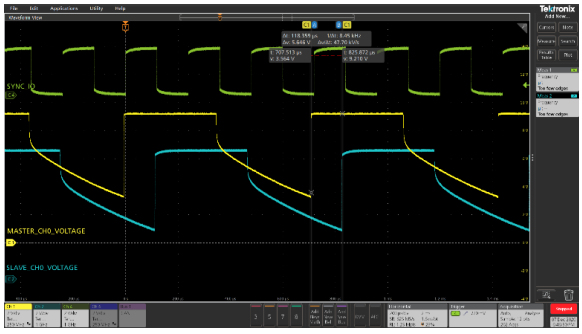


Figure 35. **Phase shift 100 (+60 µs specification target)**

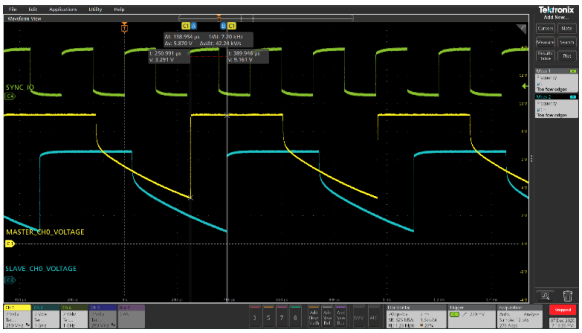


Figure 36. **Phase shift 101 (+75 µs specification target)**

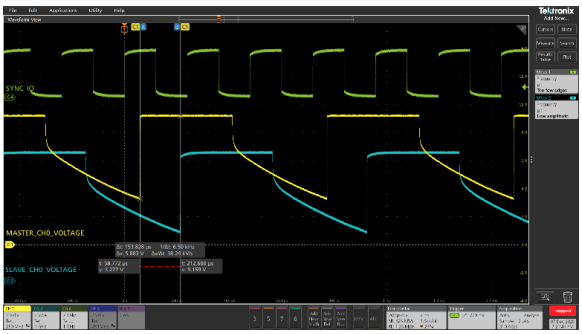


Figure 37. **Phase shift 110 (+90 µs specification target)**

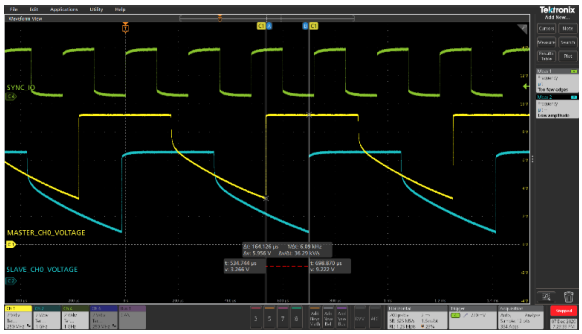
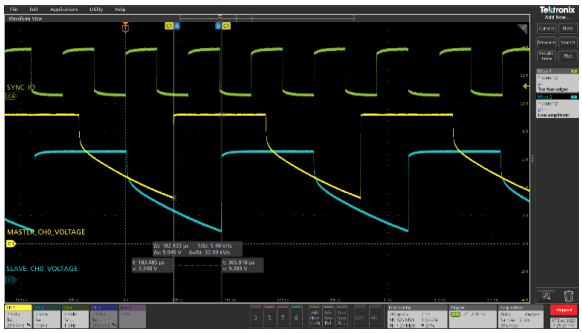


Figure 38. **Phase shift 111 (+105 µs specification target)**



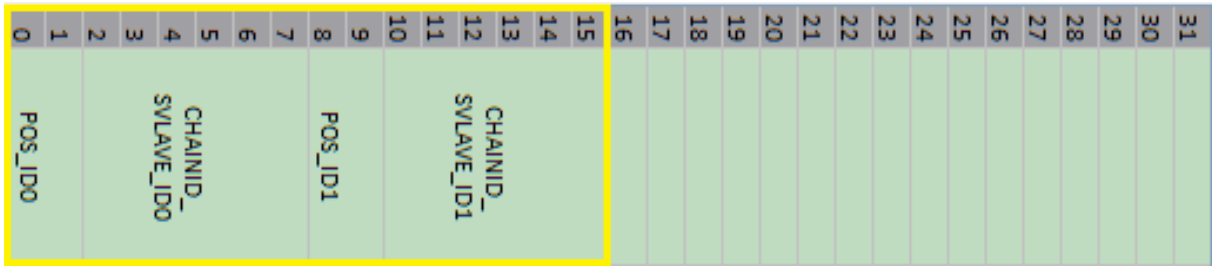
3.3 Broadcast setting of current level and duty cycle

In addition to the previously described current intensity and duty cycle setting, performed channel by channel on a single device (maximum of four simultaneous channel updates with a single frame). The L99LDLH32 Bus mode allows broadcast messages to update these parameters on multiple devices simultaneously (up to 16 channels per device and 4 devices in one shot). This feature requires every device to get preliminarily initialized with two parameters:

- **A chain identifier,** playing essentially the role of an address related to a group of up to 4 devices, to be later addressed by a single broadcast frame, for current intensity or duty cycle update. Resorting to the broader context of communication network theory for an analogy, this matches with the multicasting concept.
- **A position,** specific to each of the devices belonging to the same *Chain identifier*, and can take up to 4 possible values (from 0 to 3). This provides each single device with the pointer to the location in which to fetch the current and duty cycle update data (16 bytes) within the received broadcast frame.

Initialization of these parameters is done through an on-purpose message, which entails a device *ResponderID", Chain ID and position*. As an atomic unit required for a proper management of broadcast messages that is sent by the CAN FD Light commander, during device operation. Once the broadcast initialization message is correctly received by a given device, chain ID and position initialization data show up in register 1Ah (from bits 0 to 15):

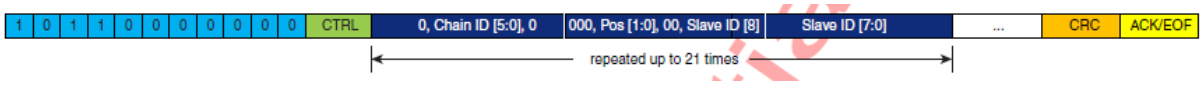
Figure 39. Register 1 ah



Warning:

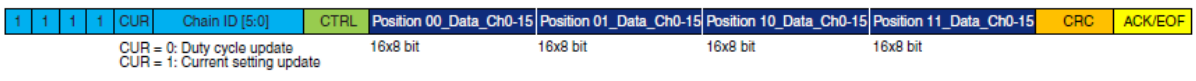
In the context of initialization for broadcast messages each device has a twofold ResponderID. The (main) ResponderID gets preassigned through the lower 9 bits of row 0 in NVM (check related annex for NVM fields content) and is valid to address all the CAN-FD light unicast communication frames, as well as broadcast frames addressing current and duty cycle update of the first 16 (from 0 to 15) channels of a given device. The secondary ResponderID is the main ResponderID incremented by one (ResponderID + 1, this doesn't need preliminary setting, it is implicitly adopted by device internal implementation) and will be used for broadcast frames intended to update current/duty cycle of channels with index from 16 to 31.

Figure 40. Broadcast init message



After this initialization scheme, at any time during device operation the CAN FD Light commander can send a broadcast message with the given chain identifier to address the related devices and the updates for the current and the duty cycle (specified within the 5th bit of the frame as shown below). The message payload is decoded by all devices matching the chain identifier and data are extracted by each device according to the position that was preassigned through the broadcast initialization message.

Figure 41. Broadcast frame structure



Examples

Test case 1. Two devices with ResponderID=0 and 4 respectively. Step by step the procedure below:

- Power-up
- Go to Bus mode
- Broadcast init with DLC=6; [(ChainID=1, Pos=0, ResponderID=1); (ChainID=1, Pos=1, ResponderID=4)]
- Device 0: status 1Ah readback
- Device 1: status 1Ah readback
- Broadcast PWM data sent (ChainID=0)
- Data acceptance verification
- Broadcast PWM data sent (ChainID=1)
- Data acceptance verification

Figure 42. Broadcast init (test case 1)

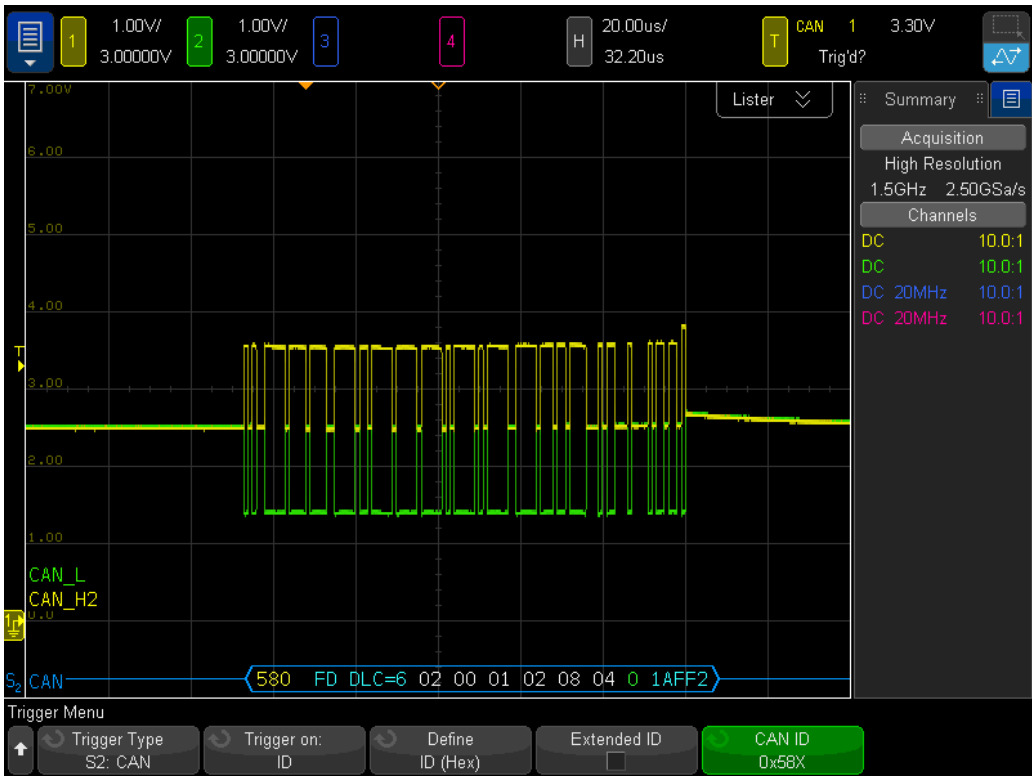


Figure 43. Broadcast init verification (test case 1)

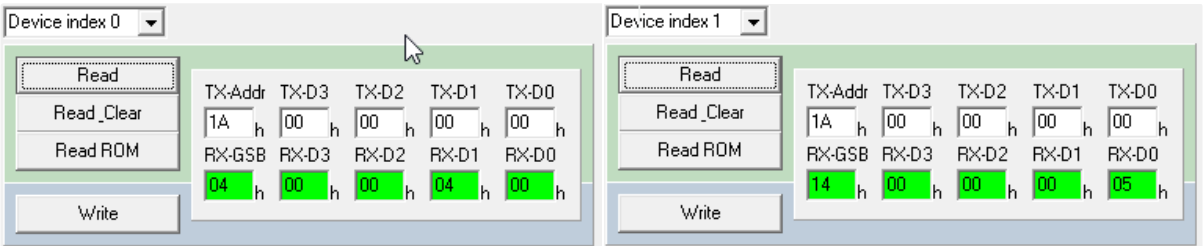


Figure 44. **Broadcast PWM data send to ChainID=0 (test case 1)**

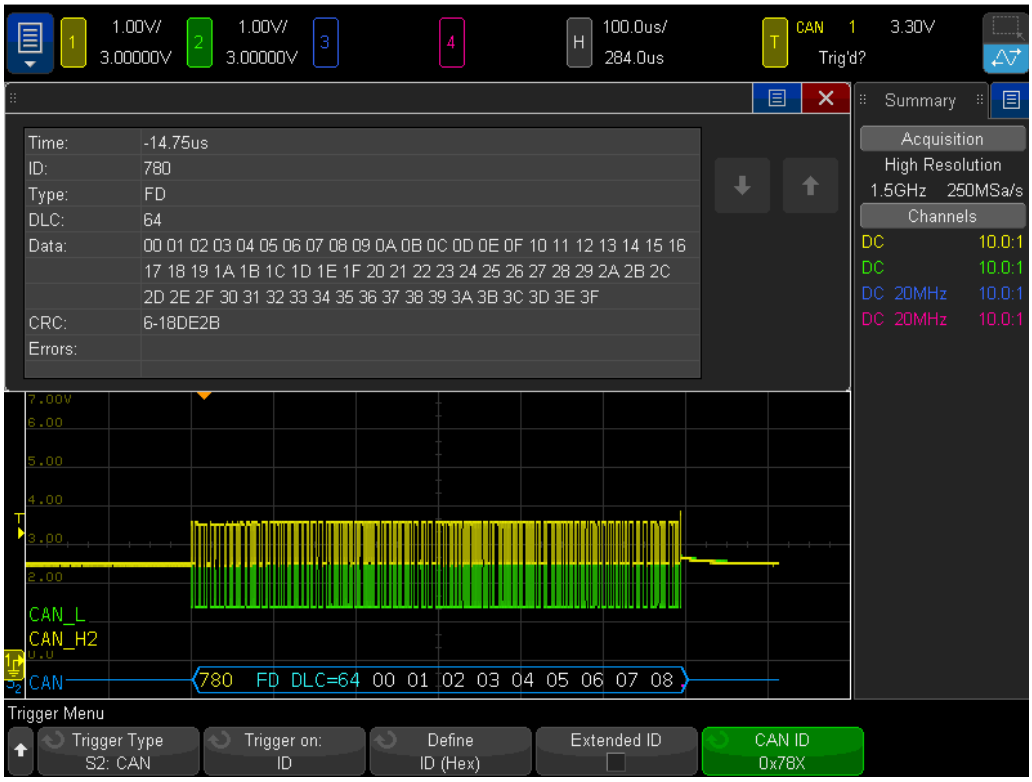


Figure 45. **Data acceptance verification (device 0 left, device 1 right) - ChainID=0 (test case 1)**

Control registers - Readback

Addr/name	D3	D2	D1	D0
01h-Control01:	00	00	00	00
02h-Control02:	03	02	01	00
03h-Control03:	07	06	05	04
04h-Control04:	0B	0A	09	08
05h-Control05:	0F	0E	0D	0C
06h-Control06:	00	00	00	00
07h-Control07:	00	00	00	00
08h-Control08:	00	00	00	00
09h-Control09:	00	00	00	00

Control registers - Readback

Addr/name	D3	D2	D1	D0
01h-Control01:	00	00	00	00
02h-Control02:	00	00	00	00
03h-Control03:	00	00	00	00
04h-Control04:	00	00	00	00
05h-Control05:	00	00	00	00
06h-Control06:	03	02	01	00
07h-Control07:	07	06	05	04
08h-Control08:	0B	0A	09	08
09h-Control09:	0F	0E	0D	0C

Figure 46. **Broadcast PWM data send to ChainID=1 (test case 1)**

Figure 47. **Data acceptance verification (device 0 left, device 1 right) - ChainID=1 (test case 1)**

Control registers - Readback

Addr/name	D3	D2	D1	D0
01h-Control01:	00	00	00	00
02h-Control02:	03	02	01	00
03h-Control03:	07	06	05	04
04h-Control04:	0B	0A	09	08
05h-Control05:	0F	0E	0D	0C
06h-Control06:	03	02	01	00
07h-Control07:	07	06	05	04
08h-Control08:	0B	0A	09	08
09h-Control09:	0F	0E	0D	0C

Control registers - Readback

Addr/name	D3	D2	D1	D0
01h-Control01:	00	00	00	00
02h-Control02:	13	12	11	10
03h-Control03:	17	16	15	14
04h-Control04:	1B	1A	19	18
05h-Control05:	1F	1E	1D	1C
06h-Control06:	03	02	01	00
07h-Control07:	07	06	05	04
08h-Control08:	0B	0A	09	08
09h-Control09:	0F	0E	0D	0C

Test case 2. Step by step the procedure below:

- Power-up
- Go to Bus mode
- Broadcast init with DLC=6; [(ChainID=0, Pos=0, ResponderID=0); (ChainID=1,Pos=0, ResponderID=4)]
- Broadcast init with DLC=6; [(ChainID=1, Pos=1, ResponderID=1); (ChainID=0,Pos=1,ResponderID=5)]
- Device 0: status 1Ah readback
- Device 1: status 1Ah readback
- Broadcast PWM data sent (ChainID=0)
- Data acceptance verification
- Broadcast PWM data sent (ChainID=1)
- Data acceptance verification

Figure 48. **Broadcast inits (test case 2)**

Figure 49. **Broadcast init verification (test case 2)**

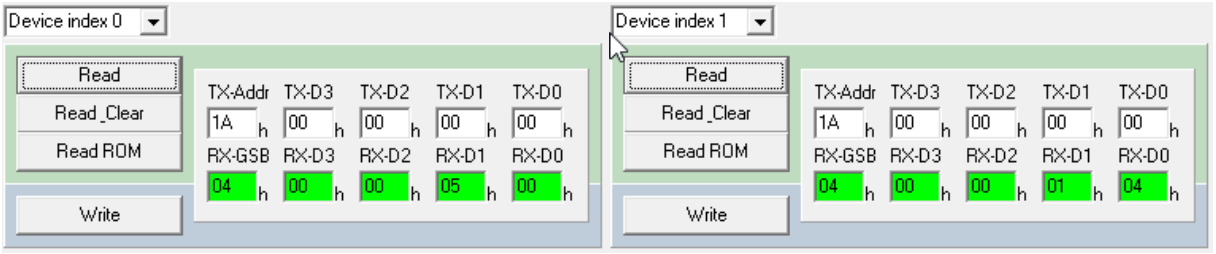
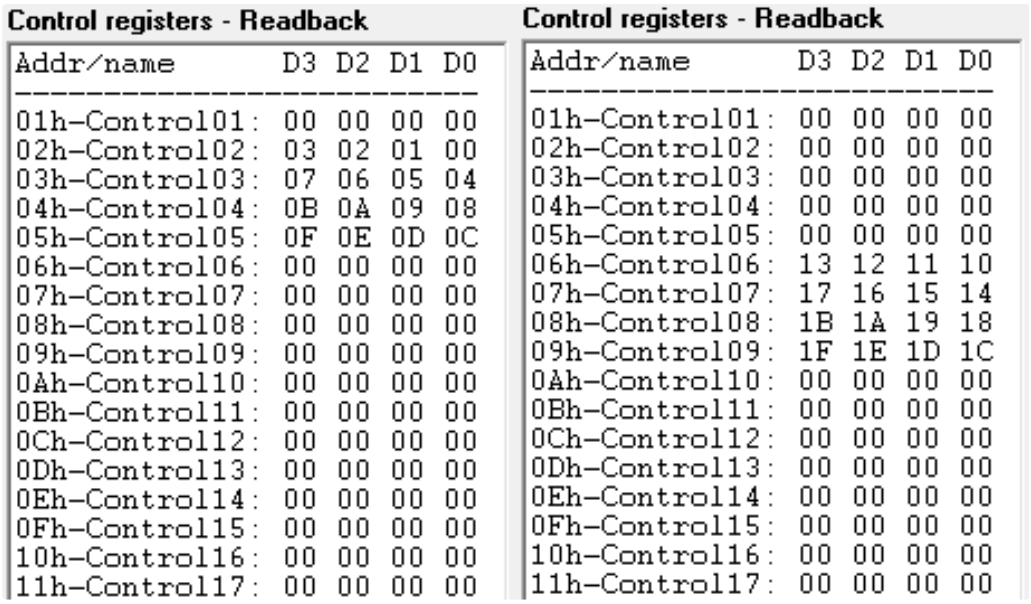


Figure 50. **Broadcast PWM data send to ChainID=0 (test case 2)**

Figure 51. **Data acceptance verification (device 0 left, device 1 right) - ChainID=0 (test case 2)**



Control registers - Readback

Addr/name	D3	D2	D1	D0
01h-Control01:	00	00	00	00
02h-Control02:	03	02	01	00
03h-Control03:	07	06	05	04
04h-Control04:	0B	0A	09	08
05h-Control05:	0F	0E	0D	0C
06h-Control06:	00	00	00	00
07h-Control07:	00	00	00	00
08h-Control08:	00	00	00	00
09h-Control09:	00	00	00	00
0Ah-Control10:	00	00	00	00
0Bh-Control11:	00	00	00	00
0Ch-Control12:	00	00	00	00
0Dh-Control13:	00	00	00	00
0Eh-Control14:	00	00	00	00
0Fh-Control15:	00	00	00	00
10h-Control16:	00	00	00	00
11h-Control17:	00	00	00	00

Control registers - Readback

Addr/name	D3	D2	D1	D0
01h-Control01:	00	00	00	00
02h-Control02:	00	00	00	00
03h-Control03:	00	00	00	00
04h-Control04:	00	00	00	00
05h-Control05:	00	00	00	00
06h-Control06:	13	12	11	10
07h-Control07:	17	16	15	14
08h-Control08:	1B	1A	19	18
09h-Control09:	1F	1E	1D	1C
0Ah-Control10:	00	00	00	00
0Bh-Control11:	00	00	00	00
0Ch-Control12:	00	00	00	00
0Dh-Control13:	00	00	00	00
0Eh-Control14:	00	00	00	00
0Fh-Control15:	00	00	00	00
10h-Control16:	00	00	00	00
11h-Control17:	00	00	00	00

Figure 52. **Broadcast PWM data send to ChainID=1 (test case 2)**

Figure 53. **Data acceptance verification (device 0 left, device 1 right) - ChainID=1 (test case 2)**

Control registers - Readback

Addr/name	D3	D2	D1	D0
01h-Control01:	00	00	00	00
02h-Control02:	03	02	01	00
03h-Control03:	07	06	05	04
04h-Control04:	0B	0A	09	08
05h-Control05:	0F	0E	0D	0C
06h-Control06:	13	12	11	10
07h-Control07:	17	16	15	14
08h-Control08:	1B	1A	19	18
09h-Control09:	1F	1E	1D	1C
0Ah-Control10:	00	00	00	00
0Bh-Control11:	00	00	00	00
0Ch-Control12:	00	00	00	00
0Dh-Control13:	00	00	00	00
0Eh-Control14:	00	00	00	00
0Fh-Control15:	00	00	00	00
10h-Control16:	00	00	00	00
11h-Control17:	00	00	00	00

Control registers - Readback

Addr/name	D3	D2	D1	D0
01h-Control01:	00	00	00	00
02h-Control02:	03	02	01	00
03h-Control03:	07	06	05	04
04h-Control04:	0B	0A	09	08
05h-Control05:	0F	0E	0D	0C
06h-Control06:	13	12	11	10
07h-Control07:	17	16	15	14
08h-Control08:	1B	1A	19	18
09h-Control09:	1F	1E	1D	1C
0Ah-Control10:	00	00	00	00
0Bh-Control11:	00	00	00	00
0Ch-Control12:	00	00	00	00
0Dh-Control13:	00	00	00	00
0Eh-Control14:	00	00	00	00
0Fh-Control15:	00	00	00	00
10h-Control16:	00	00	00	00
11h-Control17:	00	00	00	00

Figure 54. **Broadcast current data send to ChainID=0 (test case 2)**

Figure 55. **Data (current) acceptance verification (device 0 left, device 1 right) - ChainID=0 (test case 2)**

Control registers - Readback

Addr/name	D3	D2	D1	D0
01h-Control01:	00	00	00	00
02h-Control02:	03	02	01	00
03h-Control03:	07	06	05	04
04h-Control04:	0B	0A	09	08
05h-Control05:	0F	0E	0D	0C
06h-Control06:	13	12	11	10
07h-Control07:	17	16	15	14
08h-Control08:	1B	1A	19	18
09h-Control09:	1F	1E	1D	1C
0Ah-Control10:	03	02	01	00
0Bh-Control11:	07	06	05	04
0Ch-Control12:	0B	0A	09	08
0Dh-Control13:	0F	0E	0D	0C
0Eh-Control14:	00	00	00	00
0Fh-Control15:	00	00	00	00
10h-Control16:	00	00	00	00
11h-Control17:	00	00	00	00

Control registers - Readback

Addr/name	D3	D2	D1	D0
01h-Control01:	00	00	00	00
02h-Control02:	03	02	01	00
03h-Control03:	07	06	05	04
04h-Control04:	0B	0A	09	08
05h-Control05:	0F	0E	0D	0C
06h-Control06:	13	12	11	10
07h-Control07:	17	16	15	14
08h-Control08:	1B	1A	19	18
09h-Control09:	1F	1E	1D	1C
0Ah-Control10:	00	00	00	00
0Bh-Control11:	00	00	00	00
0Ch-Control12:	00	00	00	00
0Dh-Control13:	00	00	00	00
0Eh-Control14:	13	12	11	10
0Fh-Control15:	17	16	15	14
10h-Control16:	1B	1A	19	18
11h-Control17:	1F	1E	1D	1C

Figure 56. **Broadcast current data send to ChainID=1 (test case 2)**

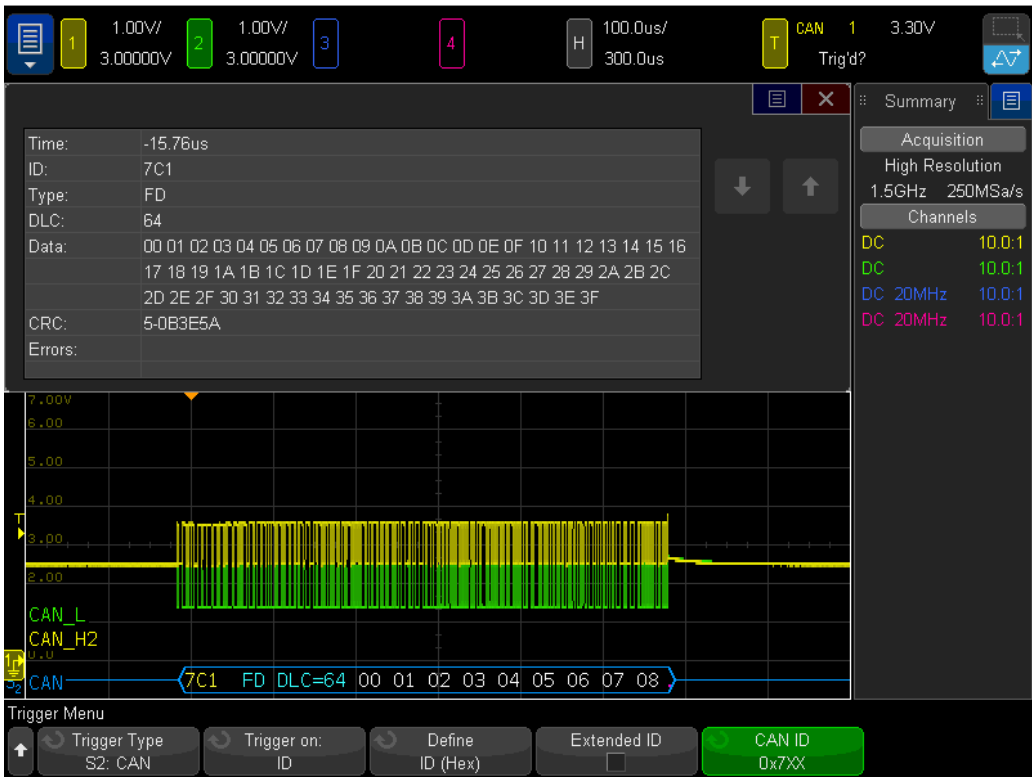
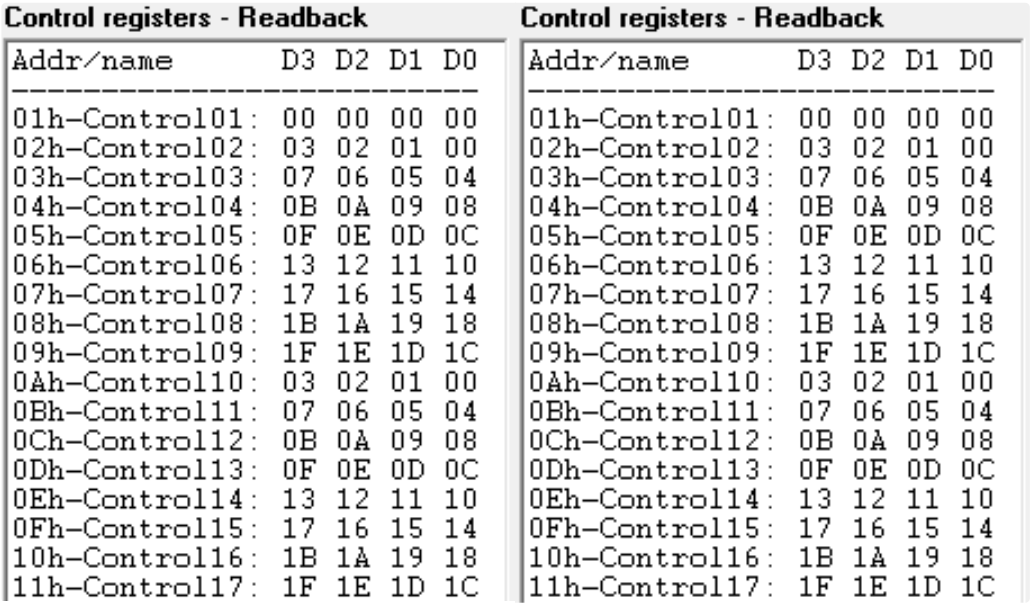


Figure 57. **Data (current) acceptance verification (device 0 left, device 1 right) - ChainID=1 (test case 2)**



Control registers - Readback

Addr/name	D3	D2	D1	D0
01h-Control01:	00	00	00	00
02h-Control02:	03	02	01	00
03h-Control03:	07	06	05	04
04h-Control04:	0B	0A	09	08
05h-Control05:	0F	0E	0D	0C
06h-Control06:	13	12	11	10
07h-Control07:	17	16	15	14
08h-Control08:	1B	1A	19	18
09h-Control09:	1F	1E	1D	1C
0Ah-Control10:	03	02	01	00
0Bh-Control11:	07	06	05	04
0Ch-Control12:	0B	0A	09	08
0Dh-Control13:	0F	0E	0D	0C
0Eh-Control14:	13	12	11	10
0Fh-Control15:	17	16	15	14
10h-Control16:	1B	1A	19	18
11h-Control17:	1F	1E	1D	1C

Control registers - Readback

Addr/name	D3	D2	D1	D0
01h-Control01:	00	00	00	00
02h-Control02:	03	02	01	00
03h-Control03:	07	06	05	04
04h-Control04:	0B	0A	09	08
05h-Control05:	0F	0E	0D	0C
06h-Control06:	13	12	11	10
07h-Control07:	17	16	15	14
08h-Control08:	1B	1A	19	18
09h-Control09:	1F	1E	1D	1C
0Ah-Control10:	03	02	01	00
0Bh-Control11:	07	06	05	04
0Ch-Control12:	0B	0A	09	08
0Dh-Control13:	0F	0E	0D	0C
0Eh-Control14:	13	12	11	10
0Fh-Control15:	17	16	15	14
10h-Control16:	1B	1A	19	18
11h-Control17:	1F	1E	1D	1C

3.4 Bus load management

An L99LDLH32 device operating in Bus mode is subjected to CAN FD Light frames containing commands as well as requests for device status read. Usually, the bus is a resource that gets shared among more devices; therefore, appropriate planning needs to be done upfront to achieve the proper tradeoff between bus load, frequency of commands, and frequency of status read requested by the CAN FD Light commander (MCU) to address application specifications. Constraints on the latter parameters can vary among applications and depend mostly on the overall set of resources present in the system as well as safety requirements. An example of bus allocation is outlined both for ASIL and QM applications below:

- Task schedule - Bus mode operation - 256 OLED pixel (8 x L99LDLH32 connected on the bus)
  - FTTI 400 ms
  - Duty cycle updated every 10 ms
  - Watchdog timeout 100 ms
  - Watchdog refresh 40 ms
  - Diagnostic read back period 80 ms
  - FuSa register readback period 200 ms
    - PWM read back
    - Chain ID read back

Once the device is woken up, broadcast initialization, current level settings are fixed and all preliminary consistency checks for safety are performed (if that is required by the application targets). The device can be sent in Bus mode and enter the operating loop with exchanges of CAN FD Light messages as outlined in the next picture (under the assumption of average bit stuffing overhead):

Figure 58. **Application scenario 1**

ASIL

Time [ms]	Frame Type	Frame Length	Comment
0	Broadcast n.1	0,63 ms	Duty cycle update on Device 0.0, 1.0, 2.0, 3.0
0,63	Broadcast n.2	0,63 ms	Duty cycle update on Device 0.1, 1.1, 2.1, 3.1
1,26	Broadcast n.3	0,63 ms	Duty cycle update on Device 4.0, 5.0, 6.0, 7.0
1,89	Broadcast n.4	0,63 ms	Duty cycle update on Device 4.1, 5.1, 6.1, 7.1
2,52	Burst read m.	0,11 ms	Output Status reading - Device x
2,62	Burst read s.	0,20 ms	
2,82	Burst read m.	0,11 ms	Status reading - Device x
2,93	Burst read s.	0,20 ms	
3,13	WD trig m.	0,11 ms	WD update Device x
3,24	WD trig s.	0,07 ms	
3,31	Sync frame	0,06 ms	
3,37	Single Read M	0,11 ms	Safety frame 1
3,48	Single Read S	0,11 ms	
3,58	Single Read M	0,11 ms	Safety frame 2
3,69	Single Read S	0,11 ms	
3,79	Sync frame	0,06 ms	
3,85	Single Read M	0,11 ms	Safety frame 3
3,96	Single Read S	0,11 ms	
4,07	Single Read M	0,11 ms	Safety frame 4
4,17	Single Read S	0,11 ms	
4,28	Sync frame	0,06 ms	
4,34	WD trig m.	0,11 ms	WD update Device y
4,44	WD trig s.	0,07 ms	
4,51	Sync frame	0,06 ms	
4,58	Single Clear M	0,11 ms	Status Clear Slot device x
4,68	Single Clear S	0,11 ms	
4,79	Sync frame	0,06 ms	
4,85			

For ASIL applications only

The recommended safety check frames are:

Figure 59. **Safety check frames**

Read back CHAIN_IDPOS - Device x
Read back PWM Setting Device x 0-3
Read back PWM Setting Device x 4-7
Read back PWM Setting Device x 8-11
Read back PWM Setting Device x 12-15
Read back PWM Setting Device x 16-19
Read back PWM Setting Device x 20-23
Read back PWM Setting Device x 24-27
Read back PWM Setting Device x 28-31
Read back PWM_DUTY_ALL Device x

Considering the task schedule period is 10 ms, the total bus load in this application example is 48.5%. In case of QM applications, the busload can be reduced as safety check frames can be eliminated:

Figure 60. **QM**

QM

Time [ms]	Frame Type	Frame Length	Comment
0	Broadcast n.1	0,63 ms	Duty cycle update on Device 0.0, 1.0, 2.0, 3.0
0,63	Broadcast n.2	0,63 ms	Duty cycle update on Device 0.1, 1.1, 2.1, 3.1
1,26	Broadcast n.3	0,63 ms	Duty cycle update on Device 4.0, 5.0, 6.0, 7.0
1,89	Broadcast n.4	0,63 ms	Duty cycle update on Device 4.1, 5.1, 6.1, 7.1
2,52	Burst read m.	0,11 ms	Output Status reading - Device x
2,62	Burst read s.	0,20 ms	
2,82	Burst read m.	0,11 ms	Status reading - Device x
2,93	Burst read s.	0,20 ms	
3,13	WD trig m.	0,11 ms	WD update Device x
3,24	WD trig s.	0,07 ms	
3,31	Sync frame	0,06 ms	
3,37	WD trig m.	0,11 ms	WD update Device y
3,48	WD trig s.	0,07 ms	
3,55	Sync frame	0,06 ms	
3,61	Single Clear M	0,11 ms	Status Clear Slot device x
3,71	Single Clear S	0,11 ms	
3,82	Sync frame	0,06 ms	
3,88			

The busload is reduced to 38.8%.
The following table provides a summary overview for theoretical busload for different use cases.

Table 6. Theoretical busload

Busload	Devices on the bus	
	5	8
ASIL	30%	48%
QM	25%	39%

4 Diagnostic and protections

4.1 Diagnostic overview

The L99LDLH32 provides the detailed diagnostic information related to output channels and other internal functions. A key part of the diagnostic subsystem is an integrated 8-bit ADC that measures the voltage at all outputs (in PWM on-state), V_S, V_{PRE_REG}, internal T_J, and external NTC sensor. These ADC results are then processed by a digital engine that evaluates and controls various fault conditions. The following diagnostic information is available for the user:

Complete overview of available diagnostic information:

- 8-bit ADC values (in registers 1Fh – 26h, 2Ch and 2Dh):
  - VLED_ON_CHx: output on-state voltage on channel x
  - VLEDON_LOW: highest on-state voltage of an active channel
  - VS: VS pin voltage
  - VPRE_REG: VPRE_REG pin voltage
  - Tj: chip temperature
  - NTC_ADC: NTC pin voltage
- Output channel related diagnostic flags (in registers 27h – 2Bh):
  - VLEDON_RFR_CHx: VLEDON ADC refresh status bit of channel x
  - OUT_STATUS_CHx: readback of channel x state
  - SHT_CHx: short circuit status bit of channel x
  - OL_CHx: open load status bit of channel x
  - OUT_SHT_GND_CHx: short to GND status bit of channel x
- Output channel related diagnostic flags - logic OR of all channels (register 2Eh):
  - OR_OUT_SHT_GND: logic OR of all OUT_SHT_GND_CHx bits
  - OR_OUT_STATUS: logic OR of all OUT_STATUS_CHx bits
  - OR_SHT: logic OR of all SHT_CHx bits
  - OR_OL: logic OR of all OL_CHx bits
- Device related diagnostic flags (in register address 2Eh):
  - PG_NOT_VPRE_REG: PG threshold is not reached
  - TSD: thermal shutdown
  - TW: thermal warning
  - NTC_DER_ACT: NTC derating active
  - NTC_FAULT: NTC fault
  - DIN_STATUS: readback of DIN pin
  - WD_STATUS: WD counter status
  - VS_UV: VS undervoltage
  - WD_FAIL: WD timeout indication
  - DAC_RES_FAULT: external DAC Reference Resistor Fault:
  - VREF_PRE_REG_MAX: reference voltage is set to maximum level

- Diagnostic flags in global sStatus byte:
  - GSBN: a NOR combination of the remaining bits of GSB
  - RSTB: device reset indication
  - FE2: functional error 2 → OR combination of:
    - SHT_CHx
    - OL_CHx
    - OUT_SHT_GND_CHx
  - FE1: functional error 1: = TSD
  - DE: device error 1 → OR combination of:
    - NTC_FAULT
    - PG_NOT_VPRE_REG
    - VS_UV
    - DAC_RES_FAULT
  - GW: Global warning → OR combination of:
    - TW
    - NTC_DER_ACT
  - FS: Fail Safe and standalone mode indication

4.2 Diagnostic feedback via CAN FD

All above listed diagnostic information can be read via CAN FD interface. There are several types of diagnostic commands described in this chapter.

4.2.1 Global status byte (GSB)

The global status byte (GSB) information is sent in each device response frame in first data byte 0, except the burst read command. This means that GSB value is obtained after any unicast access (for example after the watchdog refresh command). The GSB information can be fetched also by sending a unicast frame without data. Device responds with frame with one data byte.

Figure 61. GSB request command

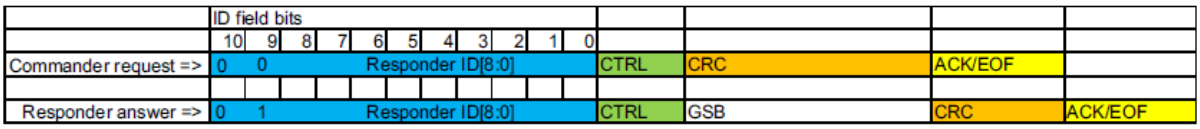
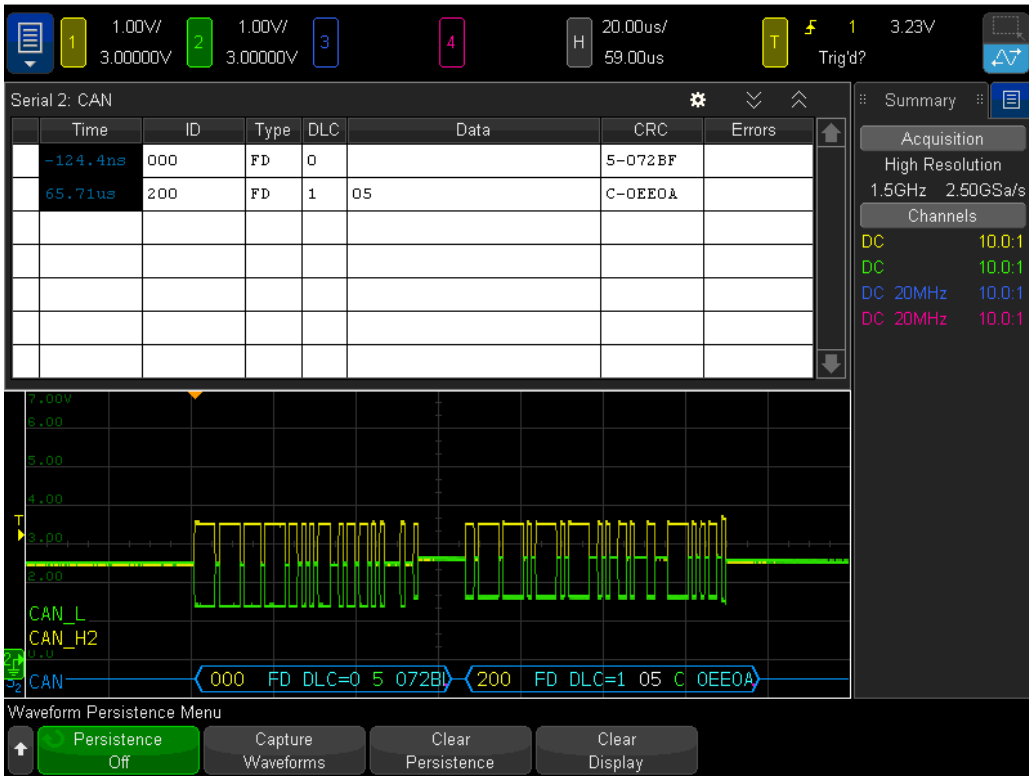


Figure 62. Example of unicast GSB request command with device response



The above screenshot shows the GSB unicast request to the device with ResponderID=000h. Device responds with GSB=05h, indicating that FS and DE bits are set.

Figure 63. Single RAM register write command with GSB response

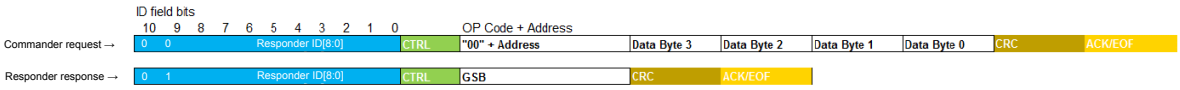
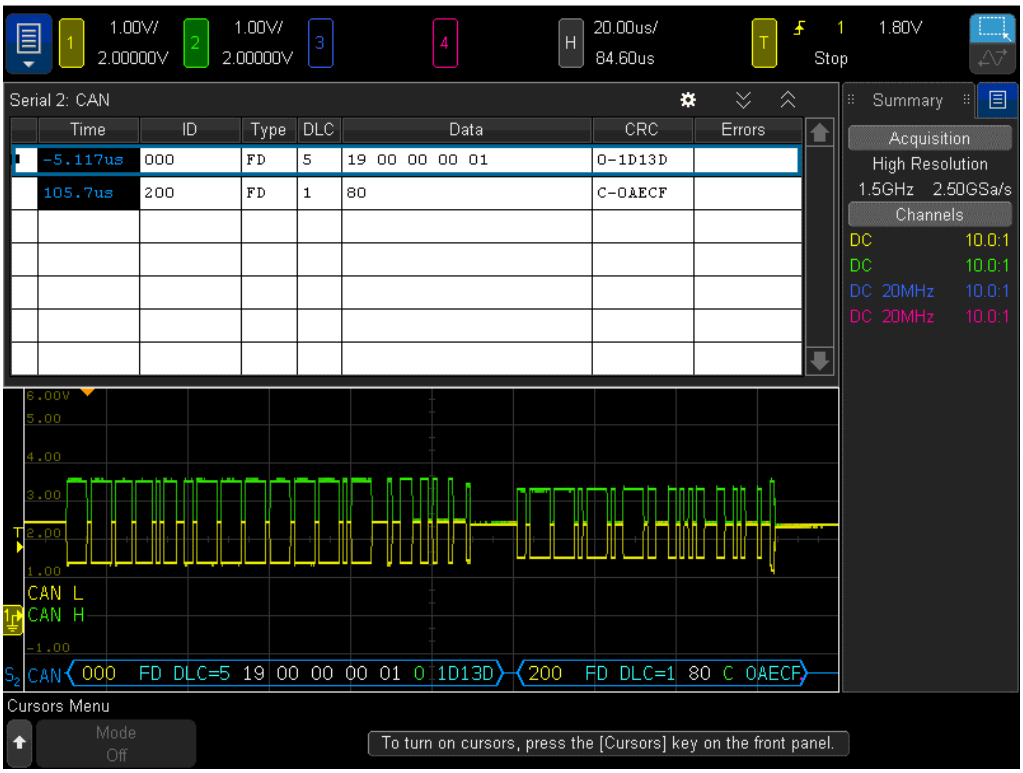


Figure 64. **Example of GSB response to watchdog refresh frame**



The above screenshot shows the device response to watchdog refresh command. In this case the GSB=80h which means no errors and no warnings are set. Same response is sent to any other single register write command as shown on Figure 63. Single RAM register write command with GSB response.

4.2.2 Single RAM register read

This command (OP Code “01”) is used to read diagnostic (or other) information from the single register on the specific address. Device responds with 5 data bytes - GSB + 4 byte register content.

Figure 65. **Single RAM register read command**

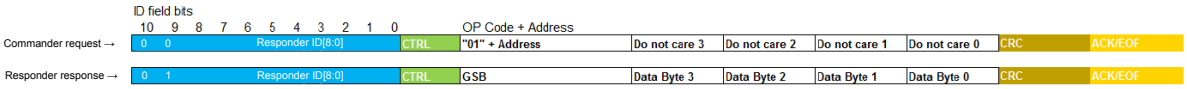
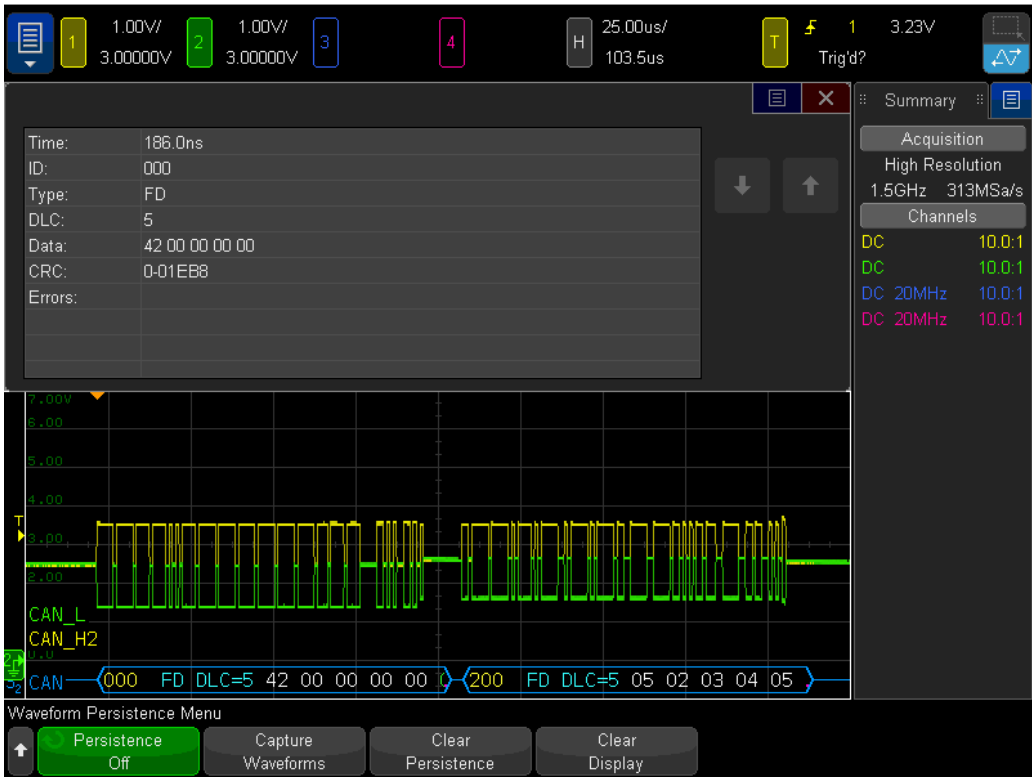


Figure 66. **Example of single read command**



This command can be used for reading any register in the RAM space. However, there are 4 specific registers dedicated for “burst read mode”. In case of accessing these addresses, the device responds in burst read mode (see next paragraph Section 4.2.3).

4.2.3 Burst read mode

A burst read mode is available when a single RAM Read request (OP Code “01”) is executed on a specific RAM addresses eligible to burst read mode (addresses 1Fh, 23h, 27h, and 2Ch). Then the responder answers with 16 bytes of data. The 4 least significant bytes correspond to the data stored in the register addressed by the commander request. The 12 following bytes correspond to the data stored on the next 3 consecutive addresses. Global status byte is not provided in this case.

Figure 67. **Read request on burst read mode RAM register**

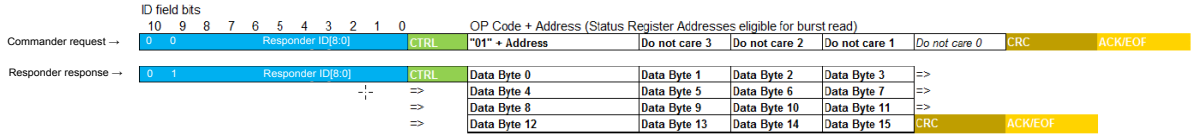
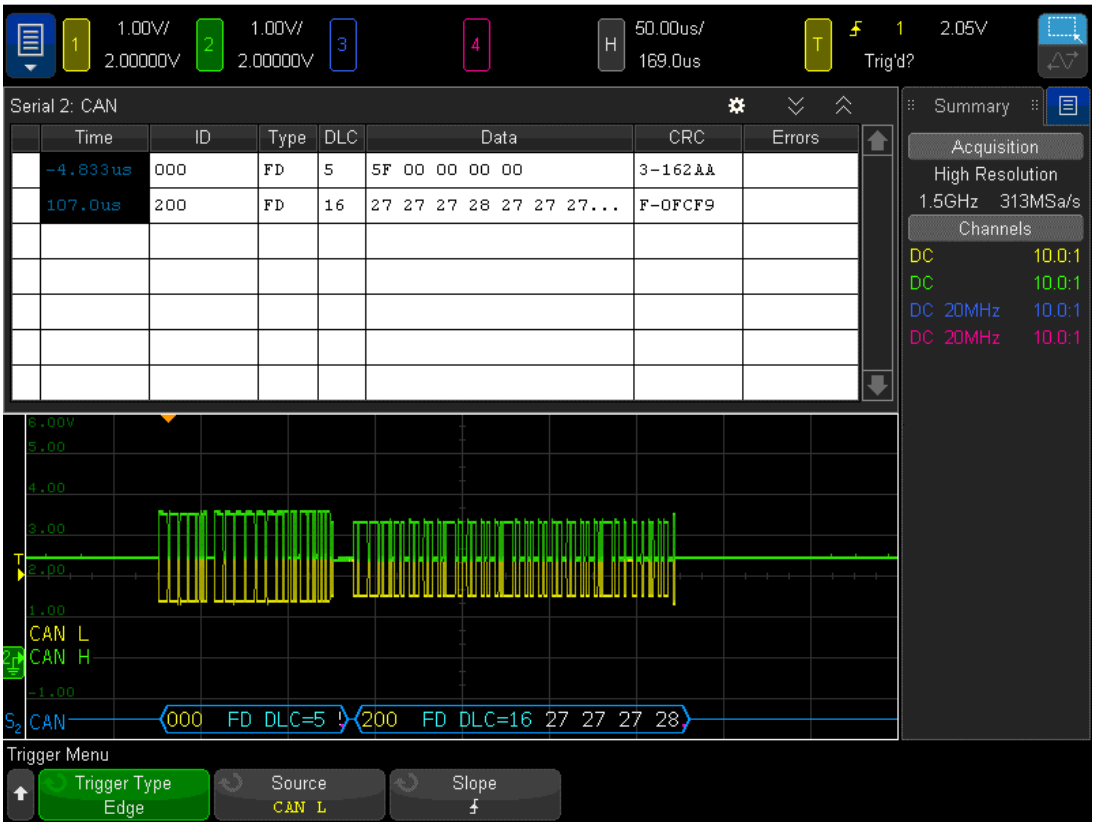


Figure 68. **Example burst read mode access on register 1Fh**

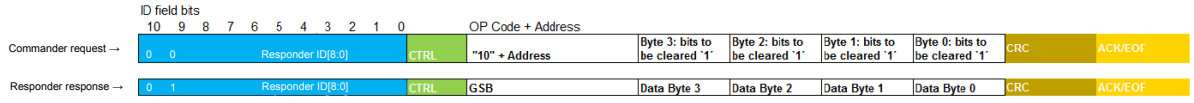


The above screenshot shows a device response to read command executed on address 1Fh, dedicated for burst mode. As seen in the register map, there is VLEDON ADC information for channels 0 to 15. In this example, the device responds with 16 data bytes (byte 0 = VLEDON_CH0, byte 1 = VLED_CH1 …) with values of 27h (→ ~6.1V) or 28h (→ ~6.3V).

4.2.4 Single RAM register clear

This command (OP code “10”) is used to clear diagnostic flags at a specific address. The command is applicable on addresses eligible for clearing (28h, 29h, 2Ah, and 2Eh). The flags to be cleared are selected using a mask defined by data byte 0-3. The device responds with 5 data bytes (GSB + 4 byte register content) immediately after the clear command is executed, therefore providing the already cleared register content. In case of a permanent fault, the corresponding flags are set again after a new fault validation time.

Figure 69. **Single RAM register clear command**



In the example below there is a sequence of read and clear command on register 29h (open load flags) sent during a permanent open load condition on CH0.

Figure 70. Example of read and clear command sequence on register 29h

The example in Figure 70 shows a permanent open load condition on ch.0. The response to a read command (with a timestamp of -1.985 ms) shows an open load on ch.0. The following clear command removes the fault so the ch.0 is reactivated. Since the open load condition is permanent, the channel is latched off again after the fault validation time of ~35 ms (8 PWM periods filter time).

4.3 Diagnostic feedback via Fault pin

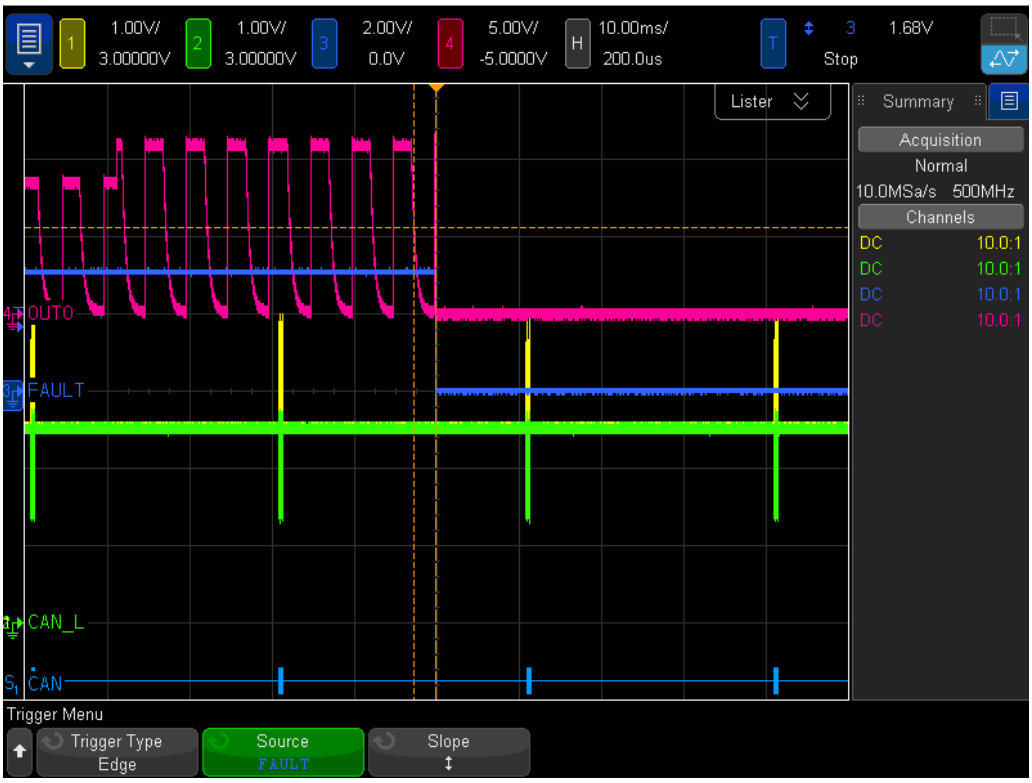
4.3.1 Diagnostic capability

Diagnostic feedback is also available on the FAULT pin. This is an open drain pin associated to channels mapped to the DIN group “01” (channels with the DIN_MAP_CHx bit set to “1”). It is active low and requires an external pull-up. It can indicate the following faults:

- Device related faults (always propagated to FAULT pin):
  - TSD thermal shutdown
  - VS_UV VS undervoltage
  - DAC_RES_FAULT: external DAC reference resistor fault
- Output channel related faults (propagation to FAULT pin configurable)
  - OL: open load on channels associated in DIN group (propagation enable by OL_EN bit)
  - SHT: short circuit on channels associated to DIN group (propagation enable by SHT_EN bit)
  - OUT_SHT_GND: Short to GND on channels associated to DIN group (propagation enable by SHT_EN bit)

The example below shows an OL fault condition on OUT0. The channel is in PWM mode (200 Hz/50% duty cycle), mapped in the DIN group using DIN_MAP_CH0=1. The OL fault propagation to FAULT pin is enabled using OL_EN=1. Device is configured in Bus mode:

Figure 71. Example of OL fault indication on FAULT pin



The OL event is visible on output voltage 0 signal (purple plot) as a sudden step to 12 V. Then, after the fault validation filter time, the output 0 is latched off and the FAULT pin (blue plot) is pulled low. Green and yellow plots are CAN_L and CAN_H signals (watchdog refresh activity every 30 ms visible in this case).

4.3.2 FAULT Bus capability

The fault pin has a bidirectional functionality. Several fault pins of several devices can be connected to a common fault bus associated to a particular light function. This allows an automatic deactivation of the complete light function in case of fault on the single O/LED string only, according to the selected fault reaction mode. The fault reaction mode setting is available in configuration register 1 (address 15h) and the setting is individual per function group:

Table 7. Fault reaction mode setting for function group “00” (non-mapped channels)

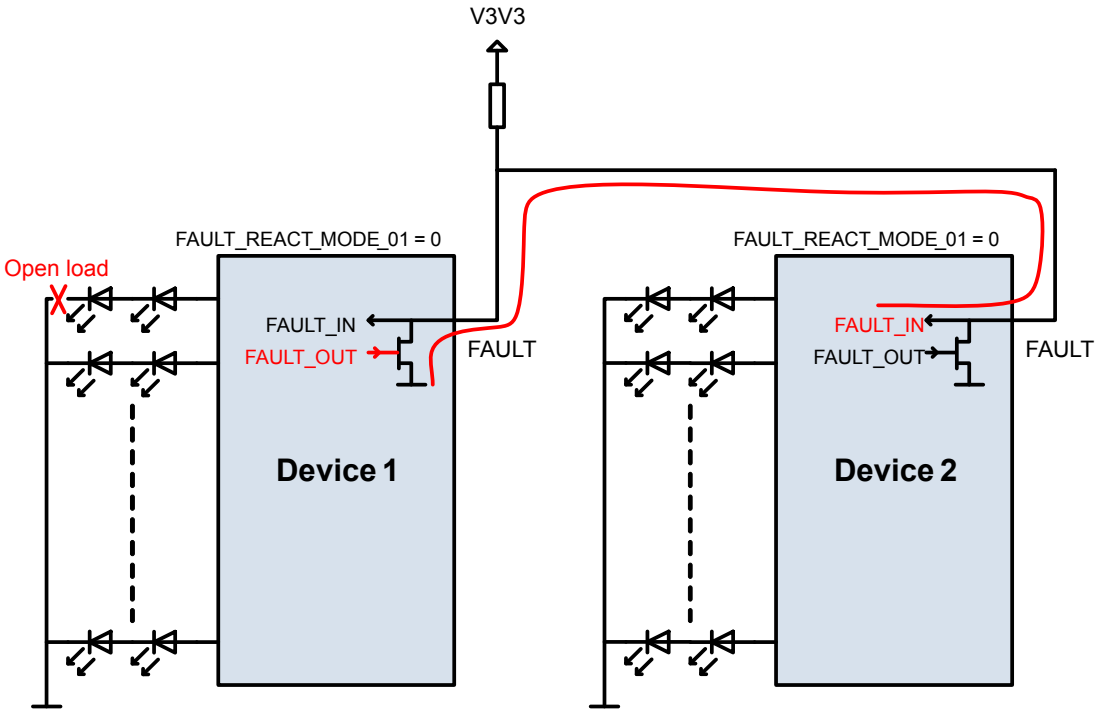
FAULT_REACT_MODE_00	Description
0	Fault on one string → all strings of the same function group within the same device deactivated
1	Fault on one string → no action on other strings

In the case of function group "00", the FAULT bus function is not available because this group is not linked to the FAULT pin.

Table 8. **Fault reaction mode setting for function group “01” (channel mapped to DIN)**

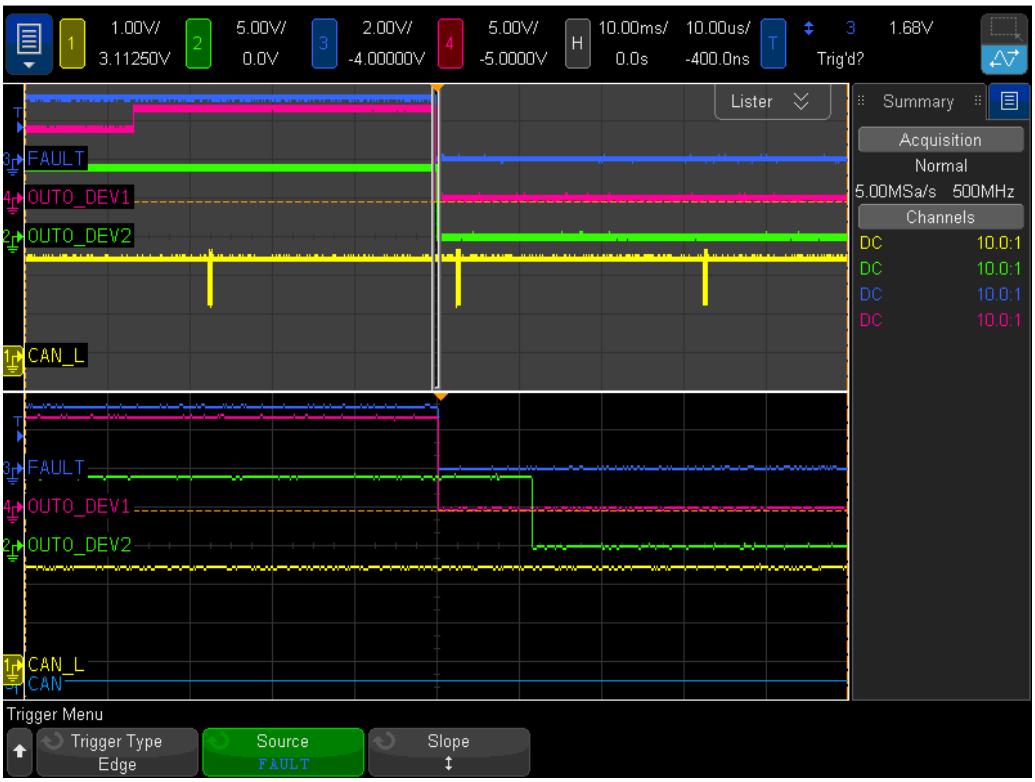
FAULT_REACT_MODE_01	Description
0	Fault on one string → all strings of the same fault bus deactivated
1	Fault on one string → no action on other strings

Figure 72. **Principle of the FAULT bus functionality**



The principle of the FAULT bus functionality is shown on the example of Figure 72, supported with measurement on Figure 73. In the case of open load condition applied on OUT0 of device 1 (see sudden increase of the output voltage on purple plot), the fault is detected after a certain validation time, and all channels mapped in DIN group (“01”) are then latched off. At the same time the FAULT pin goes low (see blue plot). The device 2, connected on same FAULT bus, detects this condition through its FAULT pin and turns off all outputs mapped in group “01” too. The delay between device 1 and device 2 channel turn off is few microseconds (seen time zoom on the bottom half of Figure 73).

Figure 73. **Example of OL fault propagation to another device**

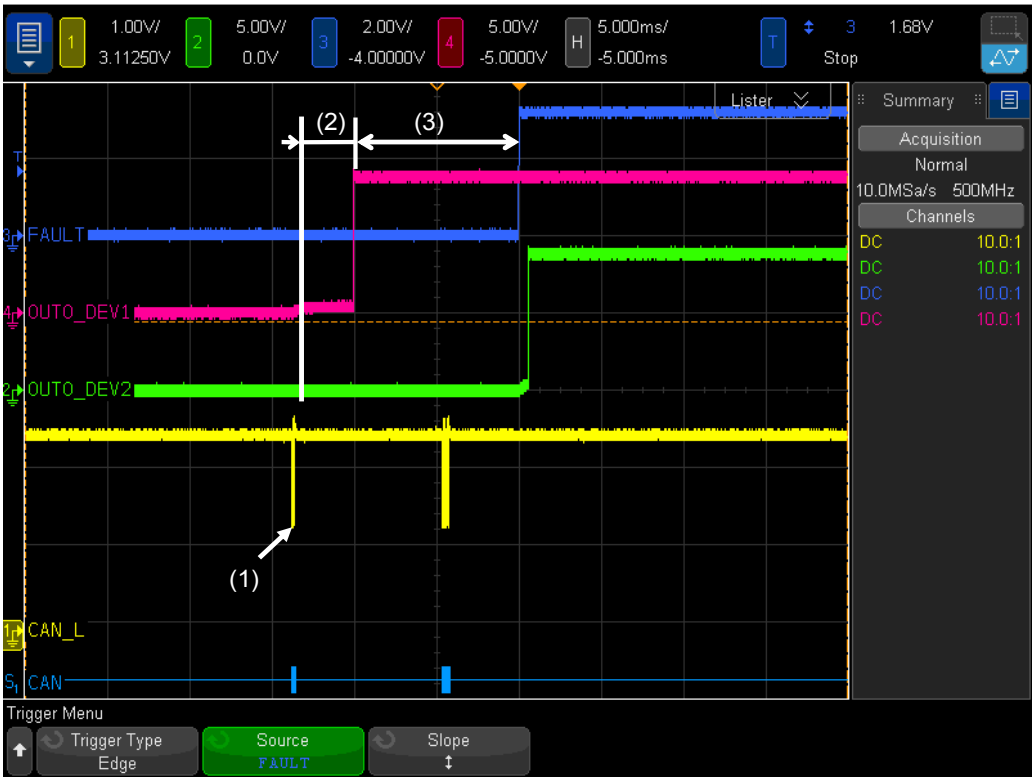


To achieve the behavior described in this example, the following settings must be used on both devices:

- OL_EN = 1: open load propagation to FAULT pin enable
- FAULT_REACT_MODE_01 = 0: fault on one string → all string of the same group deactivated

The next screenshot on Figure 74 shows the behavior after sending the open load-clear command while the open load condition on the output is not present anymore. After sending the clear command (1), the faulty output is activated at the next zero crossing of the PWM counter (end of the phase (2)). The slight voltage increase, evident between the command and the channel activation - phase (2), is caused by the release of the internal active pulldown. During phase (3), the device is checking whether there is still the open load fault on the output. If no fault is detected, all other channels in the same function group, as well as the fault pin are released (see rising edge on blue plot). The released Fault pin is detected also by the other device which allows automatic activation of all mapped channels. Otherwise, if the fault persists, the faulty channel is switched off again, while other channels in the group remain off and the fault pin remains low. This implementation prevents any flickering when trying to clear a permanent fault.

Figure 74. **FAULT bus functionality - clear command**



4.4 Diagnostic availability and validation strategy

4.4.1 Blanking time, ADC sampling

Output channel diagnostics is based on voltage monitoring using an internal ADC. Sampling of each channel is performed once per PWM period with a configurable blanking time after the PWM rising edge. In case the PWM on-time is lower than the blanking time, the diagnostic on this channel is disabled and the preregulator control signal is set to maximum (if the preregulation algorithm is enabled). In the case of 100% duty cycle (steady state), the sampling is performed in the same way, according to the internal PWM counter. Blanking time ensures stable output voltage during sampling, especially when combined with OLEDs with relatively high capacitance. There are 2 configuration bits per function group to set the desired blanking time:

Table 9. **Blanking time setting for function group “00” (non-mapped channels)**

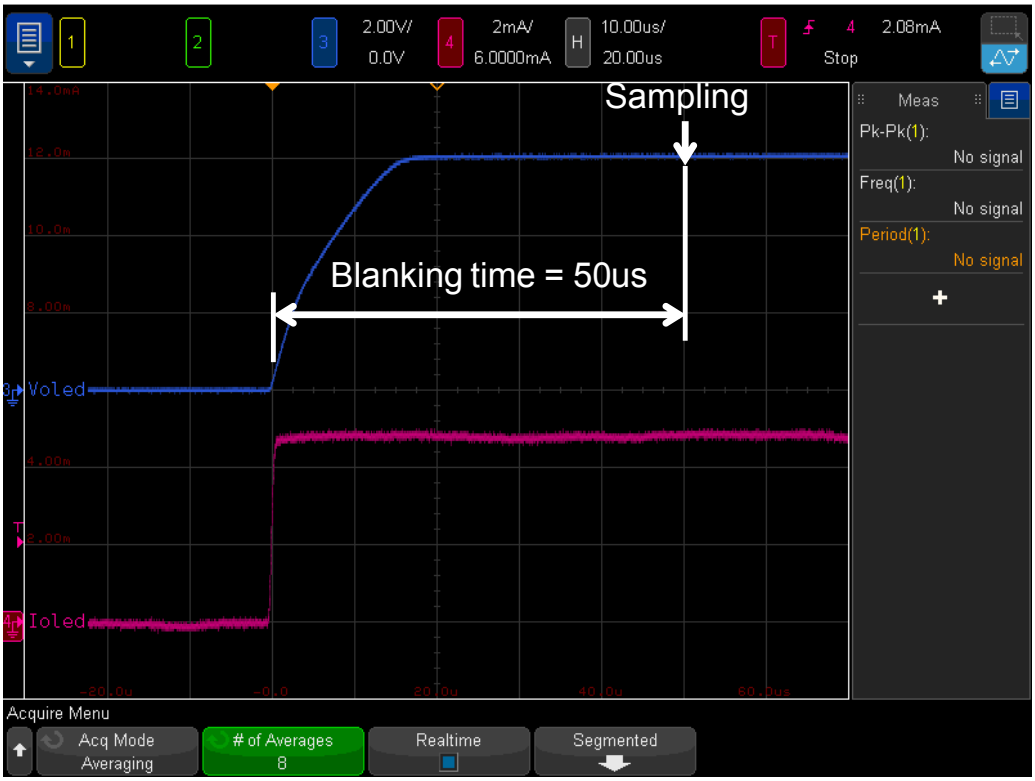
DIAG_BLANK_00 [from 1 to 0]	tDIAG_BLANK_x [µs]
00	50
01	100
10	150
11	200

Table 10. Blanking time setting for function group “01” (channels mapped to DIN)

DIAG_BLANK_01 [from 1 to 0]	tDIAG_BLANK_x [µs]
00	50
01	100
10	150
11	200

The setting of the blanking time in the application must be done according to load characteristic, to sample the already stabilized output voltage. Usually, the lowest possible value is preferred to allow diagnostic at lower PWM duty cycle values. The example in Figure 75 below shows the turn-on phase of a 0.5 cm^2 OLED pixel. As can be seen from the blue plot, forward voltage is reached in 15 µs. This means that the shortest available blanking time (50 µs) can be configured with sufficient margin for reliable diagnostic.

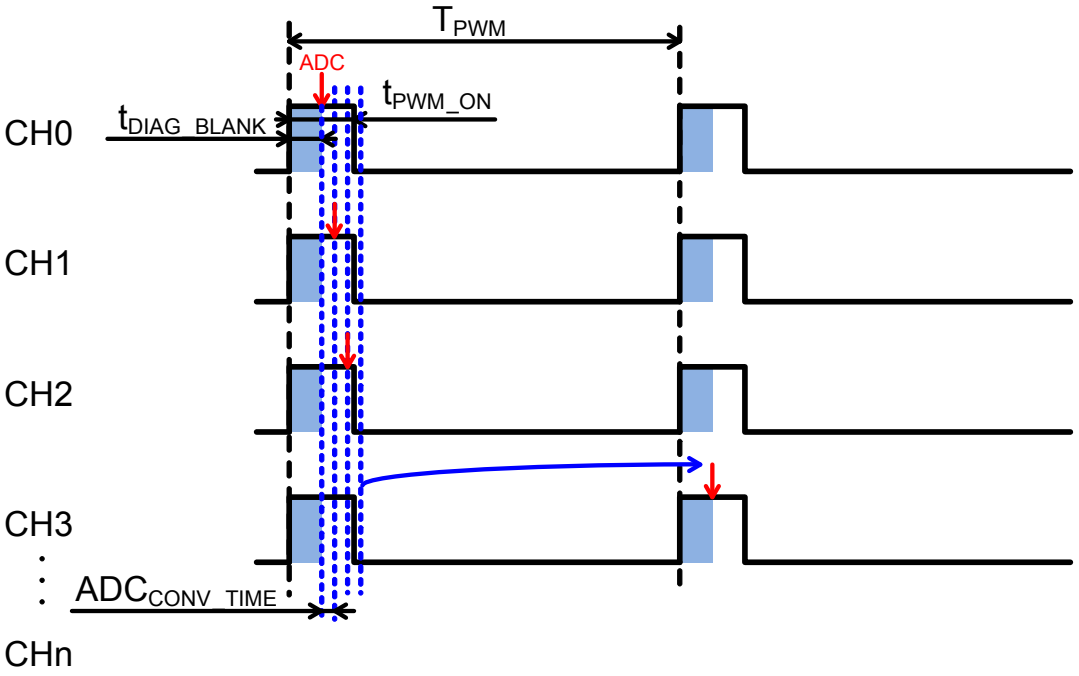
Figure 75. OLED pixel turn-on phase (5 mA, 0.5 cm^2)



As mentioned at the beginning of this chapter, the sampling of each channel is performed once per PWM period. Normally, diagnostic samples of all 32 channels are collected within one PWM period. In the case of low duty cycle, when the PWM on-time is close to the blanking time, it might happen that several periods are needed to complete diagnostic sampling of all channels. This case is depicted on Figure 76.

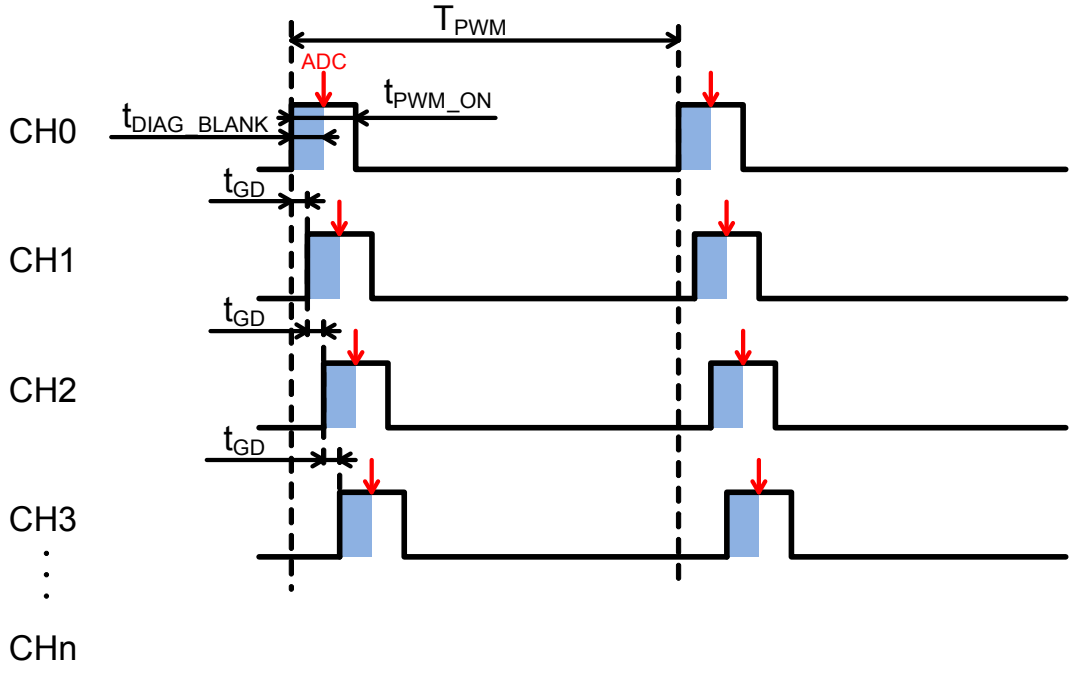
Since the gradual delay is disabled in this example, the request for sampling all 32 channels is coming at the same time. In this case the ADC conversion is performed sequentially, starting from the first channel. Due to a relatively short PWM on-time (very close to the blanking time) and considering ADC conversion time, it is possible to convert three channels only within a PWM period. In this case, the conversion of next channels continues in the next PWM periods. This means that 11 PWM periods are needed to complete the conversion on all channels (the diagnostic reaction time is 11 times higher).

Figure 76. **Diagnostic with gradual delay disabled at t_{PWM_ON} close to t_{DIAG_BLANK}**



In order to improve the diagnostic response time in the corner case explained above, the gradual delay can be enabled. Then the conversion of all channels can be completed within the single PWM period, even with such a short PWM on-time (see the figure below).

Figure 77. **Diagnostic with gradual delay enabled at t_{PWM_ON} close to t_{DIAG_BLANK}**

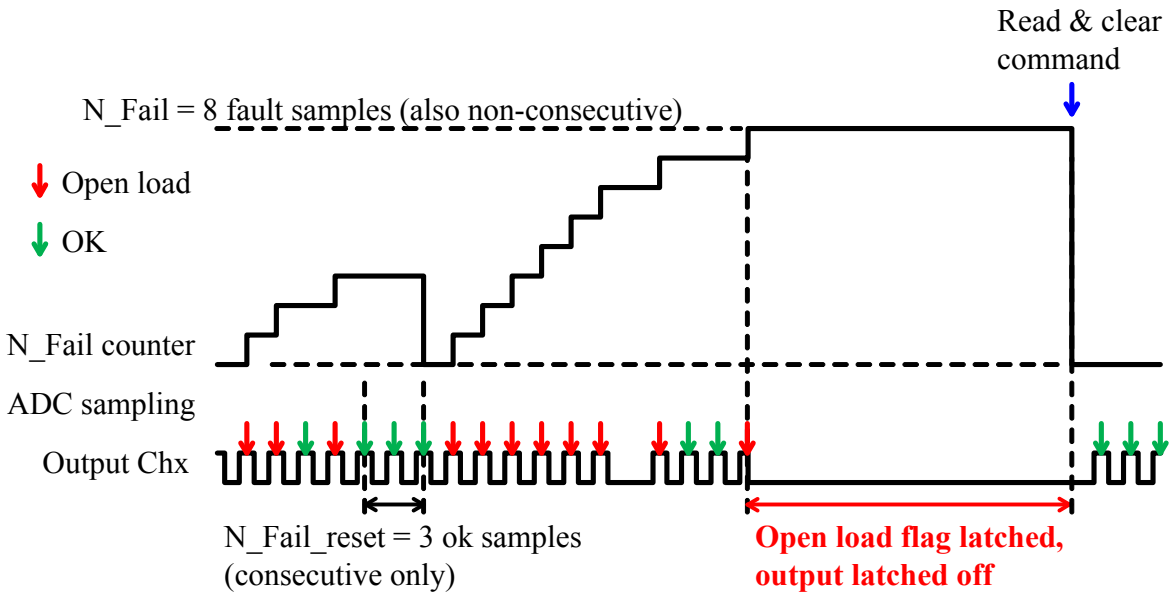


4.4.2 Fault validation strategy

To provide reliable diagnostic information, the ADC results are evaluated and filtered by device logic. In case the converted ADC value indicates any fault condition (open load, short-circuit or short-circuit to GND), a failure counter of related channels is incremented. The fault is validated after “N_Fails” number of detected failures (also non-consecutive). The failure counter is reset after “N_Fails_reset” number of consecutive NOT failures detection. The fault counter value is maintained also when the channel is switched off. This implementation guarantees reliable diagnostics also in the case of dynamic output control (for example, fast changing between on and off state). If the fault condition is present, it will always be detected, assuming that PWM on-time is higher than the blanking time. Once the fault is validated, the corresponding flag in the register is latched. Depending on the type of the fault and device configuration, also one or more corresponding channels can be latched-off. The fault flag (and corresponding output channel) can be unlatched by sending the read & clear command, which resets the N_Fail counter and activates the output. If the read & clear command is applied when the fault is not latched (N_Fail < 8), the N_Fail counter value is not touched (reset).

The concept of fault validation described above is illustrated in the example in Figure 78, which shows an intermittent open load condition. The example combines most of the possible N_Fail counter management scenarios (increment by faulty sample, keeping the actual value in case of sporadic ok sample or during off-state, reset in case of 3 consecutive ok samples, fault latch upon reaching the N_Fail threshold and reset by read & clear command). The N_Fail threshold for an open load is 8 (refer to datasheet).

Figure 78. Fault validation strategy - Intermittent open load example



This validation strategy is applied for any possible load fault. If the on time of any channel is shorter than the configured blanking time, the device automatically disables fault detection on this channel and sets the preregulator control to maximum, to avoid unreliable diagnostic.

All possible fault types are described in detail in the next chapters.

4.4.3 Power good

The power good function supervises the V_{PRE_REG} voltage (anode of the O/LED). The main purpose of this function is to inhibit channel diagnostics if the V_{PRE_REG} is below the $V_{PG\_TH\_VPRE\_REG}$ threshold, to prevent false diagnostic.

The $V_{PG\_TH\_VPRE\_REG}$ threshold is configured in the device configuration register #2 (the default value is configured in FTP). If the threshold is not reached, the PG_NOT_VPRE_REG flag is set. This flag is in device status register #3 and is also reflected in the DE bit of the global status byte.

When reading the PG_NOT_VPRE_REG bit in the application, it must be considered that the refresh of this bit also depends on the state of the preregulator control algorithm:

- Preregulator control algorithm disabled (VREF_PRE_REG bit of device configuration register #3 = 1)
  - the PG_NOT_VPRE_REG status bit is refreshed continuously (VREF_PRE pin voltage set permanently to maximum)
- Preregulator control algorithm enabled (VREF_PRE_REG bit of device configuration register #3 = 0)
  - the PG_NOT_VPRE_REG status bit is updated only in VREF_PRE = max. condition (for example, after first turn-on of a channel when the algorithm is temporarily requesting maximum VPRE_REG voltage in order to determine the O/LED forward voltage). Otherwise, the bit is frozen at its last value, since the VPRE_REG voltage is variable
  - whenever the PG_NOT_VPRE_REG is set, it is recommended to perform another reading, after sending two consecutive UNICAST WRITE frames to disable and enable the VREF_PRE_REG bit (in the device configuration register #3). This sequence temporarily creates VREF_MAX conditions so that the PG_NOT_VPRE_REG bit is refreshed

4.5 ADC values

The device allows the user to read all internal 8-bit ADC values in registers 1Fh - 26h, 2Ch, and 2Dh:

- VLED_ON_CHx: output on-state voltage on channel x
- VLEDON_LOW: highest on-state voltage of an active channel
- VS: VS pin voltage
- VPRE_REG: VPRE_REG pin voltage
- Tj: chip temperature
- NTC_ADC: NTC pin voltage

From a diagnostic and fault management perspective, it is not mandatory for the user to read these ADC values because the device processes them internally. However, it is useful for some specific cases (like specific diagnostic requirements, safety and plausibility checks, software preregulator control etc.).

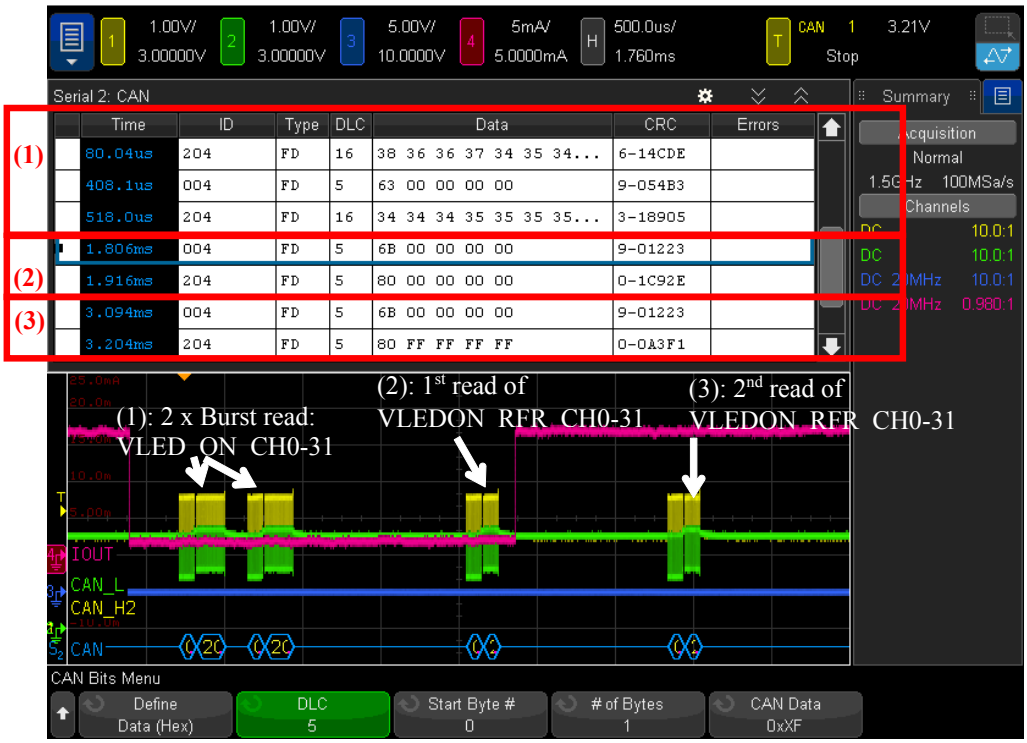
4.5.1 VLED_ON_CHx (address range from 1Fh to 26h)

VLED_ON_CHx is the 8-bit output on-state voltage available in the register area from address 1Fh to 26h. These values can be read via burst read commands (first 16 channels via burst read executed on address 1Fh, second 16 channels via burst read executed on address 23h). The measurement range is from 0 V to 40 V, so that 00h corresponds to 0 V and FFh to 40 V, giving a step of 157 mV.

VLED_ON_CHx values are refreshed according to the sampling scheme described in previous Section 4.4.1 Blanking time, ADC sampling. As there are conditions where the register value is not refreshed (for example, when the PWM on-time is lower than the blanking time, or if the output is switched off), it may be useful for the user to know if the value has been refreshed since the last read or not. This is indicated by VLEDON_RFR_CHx bits (1 bit per channel). The bit is automatically set after the refresh of the corresponding VLED_ON_CHx and cleared after reading the VLED_ON_CHx value by the user.

The example of the VLEDON_RFR_CHx bit functionality is shown on Figure 79. All outputs are activated at 200 Hz PWM at 50% duty cycle. The first 2 burst read commands (1) executed on address 1Fh and 23h (during PWM off state - see purple plot), provides VLED_ON values of all 32 channels sampled in the previous on state. Then, there is a single register read command (2) executed on address 2Bh (still within the same PWM off state), providing the VLEDON_RFR_CHx bit values. As seen from the device response, all bits are 0. This is expected, because the VLEDON values were not refreshed yet since the previous burst read. Second reading of the register 2Bh is performed ~1 ms after the next PWM rising edge (3). As seen from the device response, all bits are set 1. This means that all VLEDON ADC values in related registers were updated with new values.

Figure 79. **VLEDON_RFR_CHx bit functionality example**



4.5.2 VS, VPRE_REG, TJ (address 2Ch)

These values are available in the register address 2Ch, which can be read by a burst read command executed on this address.

VS is the 8-bit readback of the voltage on VS pin, respectively the VPRE_REG is the voltage on VPRE_REG pin. The measurement range of both values is from 0 V to 40 V, so that 00h corresponds to 0V and FFh to 40 V, giving a step of 157 mV.

TJ is the 8-bit value of junction temperature. Measurement range is from -40 °C to 175 °C with following mapping (a linear interpolation can be used for other values between these points):

- 6Bh: 175 °C
- 82h: 140 °C
- A6h: 85 °C
- CCh: 25 °C
- F7h: -40 °C

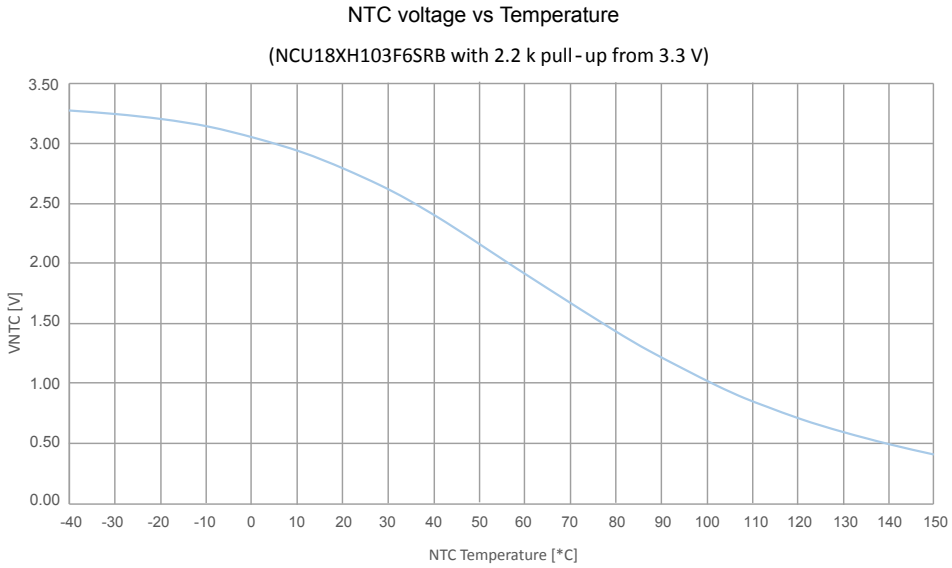
4.5.3 VLEDON_LOW, NTC (address 2Dh)

These values are available in the register address 2Dh, which can be read either by a single read command or together with other values by a burst read command executed on previous address 2Ch.

The VLEDON_LOW reflects the highest VLED_ON_CHx value of the active channel. All active channels are included in this evaluation, even those with the PWM on-time lower than the blanking time set. In this later case, the actual VLED_ON_CHx register values (last valid ADC conversions) are considered. The VLEDON_LOW reflects also channels in a temporary fault condition while they are active. A channel is excluded from evaluation when it is latched-off after fault validation is complete. If all device channels are switched off the VLEDON_LOW is set to 00h. Typically, the VLEDON_LOW can be used as an input value for software control of an external preregulator. The availability of this value significantly reduces the BUS load and software complexity if software control is preferred for any reason. Otherwise, the device has a built-in preregulator control algorithm providing the reference through VREF_PRE_REG pin, which operates fully autonomously without the need for SW intervention and additional BUS load (see detailed description in chapter related to the preregulation algorithm).

The NTC is an 8-bit readback of the voltage on the NTC input pin. The measurement range is from 0 V to 2.5 V, so that 00h corresponds to 0 V and FFh to 2.5 V, giving a step of ~10 mV. This voltage can be used for precise temperature measurement on the NTC, usually placed as close as possible to (O)LED. The transfer characteristic using the recommended NTC and pull-up is shown in the following graph:

Figure 80. NTC output voltage characteristic



Note: Since the ADC measurement range of the NTC pin is up to 2.5 V, the minimum temperature that can be measured is 35 °C, considering the recommended NTC and pull-up.

4.6 Output channel diagnostic

The L99LDLH32 features open load, short-circuit and short-circuit to ground detection for each output channel, both in Bus mode and Fail Safe/Stand Alone mode. The diagnostic is automatically disabled if the voltage at the V_{PRE_REG} pin (anode of the O/LED) is below the $V_{PG\_TH\_VPRE\_REG}$ threshold to prevent false diagnostic in case the anode supply voltage is below expected minimum (see Section 4.4.3).

4.6.1 Open load

The open load condition is recognized as soon as the voltage drop across the internal current source - read by ADC as a difference between VPRE_REG and VOUTx - is less than the specified open load threshold (VOL_TH). The open load fault is validated after N_Fail = 8 open load detections. This filtering prevents false open load detection, for example in the case of switching on an output with a higher LED forward voltage, when the preregulation algorithm and the external preregulator need some time to increase the VPRE_REG voltage. The fault validation strategy is described in detail in Section 4.4 .

When an open load fault is confirmed, the OL_CHx flag of the corresponding channel is set and the output is latched off. Depending on the fault reaction mode setting, other outputs in the same group may also be latched off. The OL_CHx flags are available in status register address 29h, which can be read either by a single read command or together with other flags by a burst read command executed on previous address 27h. The fault is also reported in the global status byte register in the functional error 2 (FE2) bit, as well as in the status register #3 (address 2Eh) in the OR_OL bit, providing an OR combination of all channels open load status.

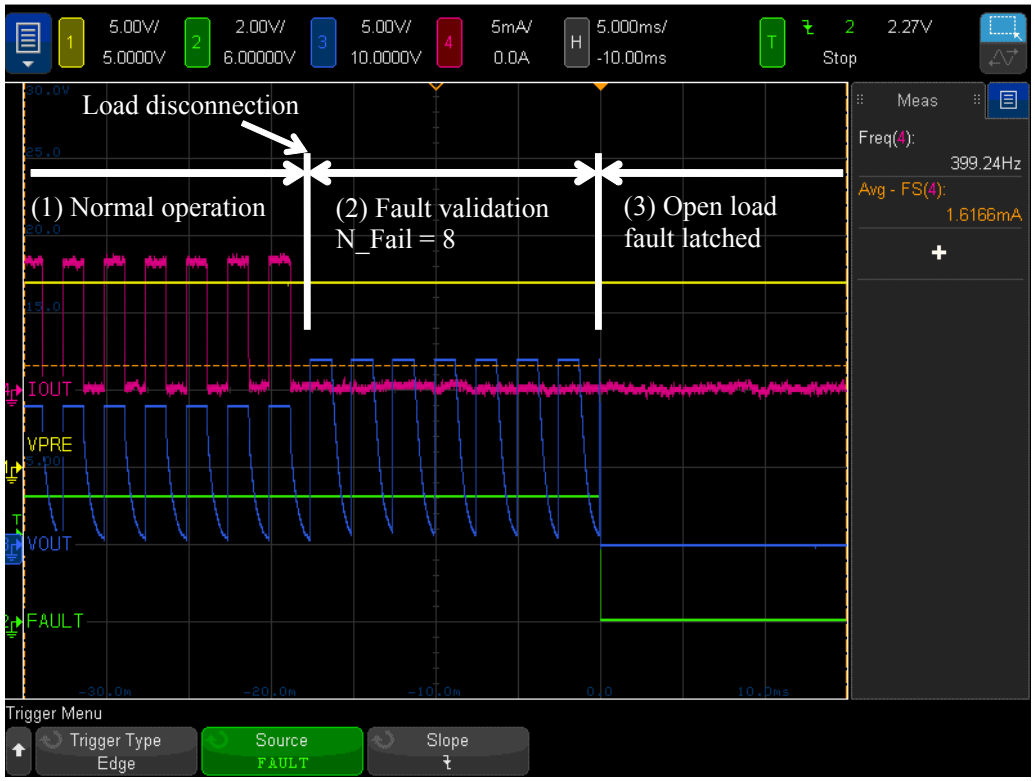
For all (and only) those channels mapped on DIN pin (DIN_MAP_CHx = 01), the fault can be indicated also via FAULT pin, depending on OL_EN configuration bit setting. If the bit OL_EN = 0, the open load fault is NOT propagated on the FAULT bus; while if OL_EN = 1, the open load fault is propagated to FAULT bus (more details about FAULT bus functionality can be seen in Section 4.3.2 FAULT Bus capability).

In Bus mode, the fault bit can be cleared, and channel output restarted by sending a read & clear command to address 29h. Selective bitwise read & clear is possible. In case of sending the read & clear command while the open load condition is still present, the fault bit is cleared, and channel output restarted. The output will then remain active until the fault is reconfirmed.

In Fail Safe and standalone mode, the fault is cleared (and channel restarted) either automatically or upon falling edge of the DIN pin, depending on channel configuration:

- Channels configured with FS_OUT_EN = 0 (controlled by DIN pin)
  - The channel is restarted upon the falling edge of DIN with t_{DIN_FALL} settling time. This means that the fault is not cleared during PWM dimming with PWM off-time shorter than t_{DIN_FALL} time
- Channels configured with FS_OUT_EN = 1 (permanently on)
  - The fault is cleared, and the channel is restarted automatically, after elapsing the $t_{AUTORESTART}$

Figure 81. Example of open load condition detection



An example of open load detection is shown in Figure 81. The device normally operates in PWM mode until the load is disconnected - phase (1). After the load is disconnected, the output current (red plot) remains zero, while the output voltage (blue plot) reaches the VPRE_REG voltage (yellow plot) every PWM on-state, phase (2). When the N_Fail=8 is reached (8^{th} PWM period), the open load fault is confirmed, and the channel is latched off , phase (3). In this case, the device is configured so that the fault is propagated to the FAULT pin (see falling edge on the green plot).

4.6.2 LED short circuit

The LED short-circuit condition is recognized as soon as the output voltage V_{OUTx}, read by the ADC, is lower than the configured short-circuit threshold. The short-circuit fault is validated after N_Fail = 14 short-circuit detections. The fault validation strategy is described in detail in Section 4.4 Diagnostic availability and validation strategy.

The short-circuit detection can be enabled or disabled through the SHT_DET_EN bit:

- SHT_DET_EN = 0: short-circuit detection disabled
- SHT_DET_EN = 1: short-circuit detection enabled

If the short-circuit is confirmed, the SHT_CHx flag of the corresponding channel is set. Depending on the setting of the SHT_OFF_x bit (x = group, either "00" or "01"), two different scenarios are possible:

- SHT_OFF_x = 0: faulty channel output kept active
- SHT_OFF_x = 1: faulty channel output latched off

Depending on the fault reaction mode setting, other outputs in the same group may also be latched off. The SHT_CHx flags are available in status register address 28h, which can be read either by a single read command or together with other flags by a burst read command executed on previous address 27h. The fault is also reported in the global status byte register in the functional error 2 (FE2) bit, as well as in the status register #3 (address 2Eh) in the OR_SHT bit, providing an OR combination of all channels' short-circuit status.

For all (and only) those channels mapped on the DIN pin (DIN_MAP_CHx = 01), the fault can be indicated also via the FAULT pin, depending on the SHT_EN and the SHT_OFF_01 configuration bit setting:

- SHT_EN = 0: the short-circuit fault is NOT propagated on the FAULT bus
- (SHT_EN = 1) & (SHT_OFF_01 = 1): the short-circuit fault is propagated to the FAULT bus - more details about FAULT bus functionality can be seen in Section 4.3.2 FAULT Bus capability

The reason why fault propagation is conditioned also by the SHT_OFF_01 bit is to prevent other channels from being switched off via the FAULT bus if the channels affected by the SHT fault need to remain on (when SHT_OFF_01 = 0 required).

In Bus mode, the fault bit can be cleared, and channel output restarted by sending a read & clear command to address 28h. Selective bitwise read & clear is possible. In the case of sending the read & clear command while the short-circuit condition is still present, the fault bit is cleared, and channel output restarted until the fault is reconfirmed again.

In Fail Safe and standalone mode, the fault is cleared (and the channel restarted) either automatically or upon falling edge of the DIN pin, depending on channel configuration:

- Channels configured with FS_OUT_EN = 0 (controlled by DIN pin)
  - The fault is cleared, and the channel is restarted upon the falling edge of DIN with t_{DIN_FALL} settling time. This means that the fault is not cleared during PWM dimming with PWM off-time shorter than t_{DIN_FALL} time
- Channels configured with FS_OUT_EN = 1 (permanently on)
  - The fault is cleared, and the channel is restarted automatically, after elapsing the $t_{AUTORESTART}$

The device has 2 groups of configurable SHT thresholds: the SHT_THR_VPRE_REG related to channels not mapped in DIN group (DIN_MAP_CHx = 0) and the SHT_VPRE_REG_DIN, related to channels mapped in DIN group (DIN_MAP_CHx = 1). The SHT_THR_VPRE_REG is configurable in configuration register #1 (address 15h) and the SHT_VPRE_REG_DIN in configuration register #2 (address 16h). The default values are configurable in FTP memory:

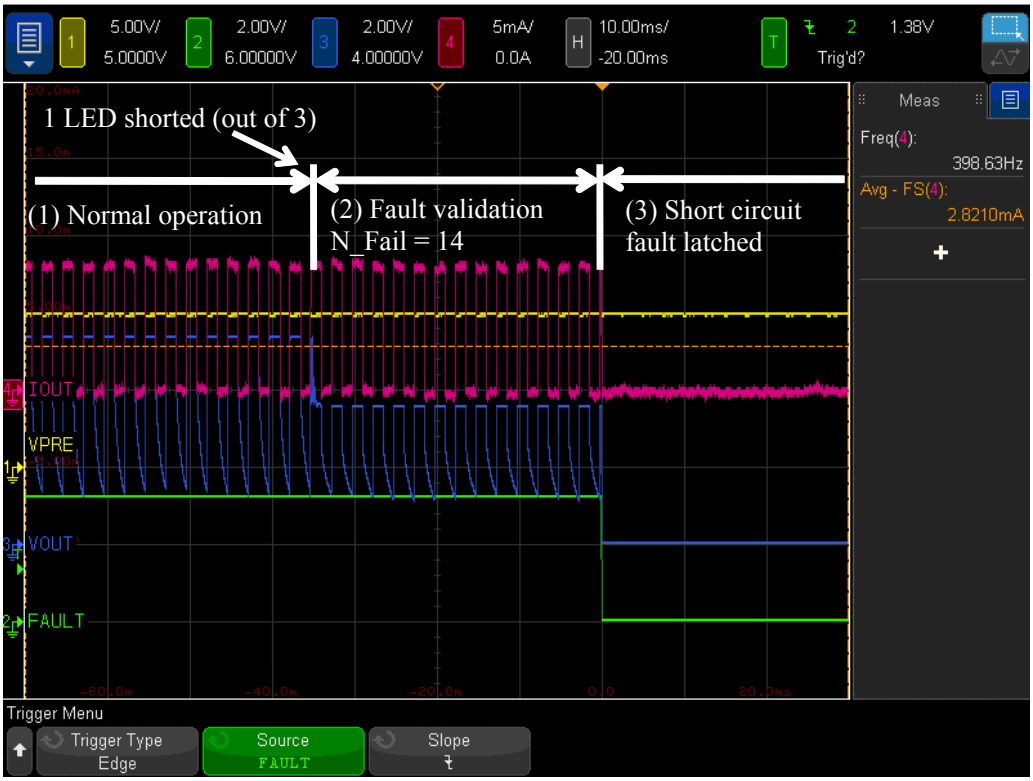
Table 11. SHT threshold setting for function group “00” (channels non-mapped to DIN)

SHT_THR_VPRE_REG (register 15h, bit 0-3)	Short-circuit threshold for group “00” channels adjustable in 314 mV steps
0h	0.31 V
1h	0.62 V
2h	0.94 V
…	…
Fh	5.02 V

Table 12. SHT threshold setting for function group “01” (channels mapped to DIN)

SHT_THR_VPRE_REG_DIN (register 16h, bit 16-19)	Short-circuit threshold for group “01” channels adjustable in 314 mV steps
0h	0.31 V
1h	0.62 V
2h	0.94 V
…	…
Fh	5.02 V

Figure 82. Example of short-circuit condition detection



An example of short-circuit detection is shown in the Figure 82 using the following setting: SHT_OFF_01=1, SHT_DET_EN=1, SHT_EN=1, SHT_THR_VPRE_REG_DIN =”1111”. In phase (1), the device normally operates in PWM mode until one LED in the string is shorted, causing the output voltage to drop below the configured threshold, phase (2). The short-circuit condition is then detected every PWM cycle until N_FAIL=14 is reached. After the short-circuit is confirmed, the SHT flag of the corresponding channel is set and the channel is latched off, phase (3). In this case, the device is configured so that the fault is propagated to FAULT pin (see falling edge on the green plot).

4.6.3 Short to GND

The short-circuit to GND condition is recognized when the output voltage V_{OUTx}, read by ADC, falls below a fixed threshold $V_{OUT\_SHT\_GND\_TH}$. The fault is validated after N_Fail = 14 short circuit to GND detections. The fault validation strategy is described in detail in Section 4.4 Diagnostic availability and validation strategy.

The short circuit to GND detection is always active, except the case when the VPRE_REG voltage is below the configured power good threshold. If the short circuit to GND condition is confirmed, the corresponding OUT_SHT_GND_CHx flag is set, and the channel is latched off. The faulty channel is disabled regardless the SHT_OFF_x bit setting, described in the previous chapter.

Depending on the fault reaction mode setting, other outputs in the same group may also be latched off. The OUT_SHT_GND_CHx flags are available in status register address 2Ah, which can be read either by a single read command or together with other flags by a burst read command executed on previous address 27h. The fault is also reported in the global status byte register in the functional error 2 (FE2) bit, as well as in the status register #3 (address 2Eh) in the OR_OUT_SHT_GND bit, providing an OR combination of all channels’ short circuit to GND status.

For all (and only) those channels mapped on DIN pin (DIN_MAP_CHx = 01), the fault can be indicated also via the FAULT pin, depending on the SHT_EN configuration bit setting:

- SHT_EN = 0: the short circuit to GND fault is NOT propagated on FAULT bus
- SHT_EN = 1: the short circuit to GND fault is propagated to FAULT bus - more details about FAULT bus functionality can be seen in Section 4.3.2 FAULT Bus capability

In Bus mode, the fault bit can be cleared, and channel output restarted by sending a read & clear command to address 2Ah. Selective bitwise read & clear is possible. In the case of sending the read & clear command while the short circuit to GND condition is still present, the fault bit is cleared, and channel output restarted until the fault is reconfirmed again.

In Fail Safe and standalone mode, the fault is cleared (and the channel restarted) automatically after elapsing the $t_{AUTORESTART}$.

4.6.4 Output status

The output status bits (1 bit per channel), available in status register address 27h, provide information about the actual state of output channels. This register (together with other diagnostic registers) can be accessed by the burst read command, executed on address 27h:

- OUT_STATUS_CHx = 0: channel gate OFF
- OUT_STATUS_CHx = 1: channel gate ON (regardless of the PWM OFF phase)

(where x = channel number):

These bits are intended as additional information to the other diagnostic flags, for safety reasons, as confirmation that the actual state of the channel is as expected. As an example, these bits can be used as a verification check of the FAULT bus propagation (described in detail in Section 4.3.2 FAULT Bus capability). In case some channels are disabled via the FAULT bus, due to a fault on another device, the OUT_STATUS_CHx information confirms that these channels have been disabled and the FAULT bus path is thus functioning as expected.

The information is also available in the status register #3 (address 2Eh) in the OR_OUT_STATUS bit, providing an OR combination of all channels output status.

4.7 DIN status

The DIN_STATUS bit, available in status register #3 (address 2Eh), reflects the logical status at the DIN pin:

DIN_STATUS = 0: DIN pin = low

DIN_STATUS = 1: DIN pin = high

This flag can be used, for example, as a plausibility check of direct drive path.

4.8 Watchdog fail and status flags

The WD_FAIL bit, available in the status register #3 (address 2Eh), indicates the watchdog expiration fail:

- WD_FAIL = 0: WD_TRIG bit toggled within WD_CONF time out
- WD_FAIL = 1: WD_TRIG bit not toggled within WD_CONF time out
  - device enters Fail Safe and standalone mode
  - the WD_FAIL bit is read and clear bit - after it is set, a Clear command is required to clear the fault. Cleared WD_FAIL bit is a prerequisite for entering the Bus mode

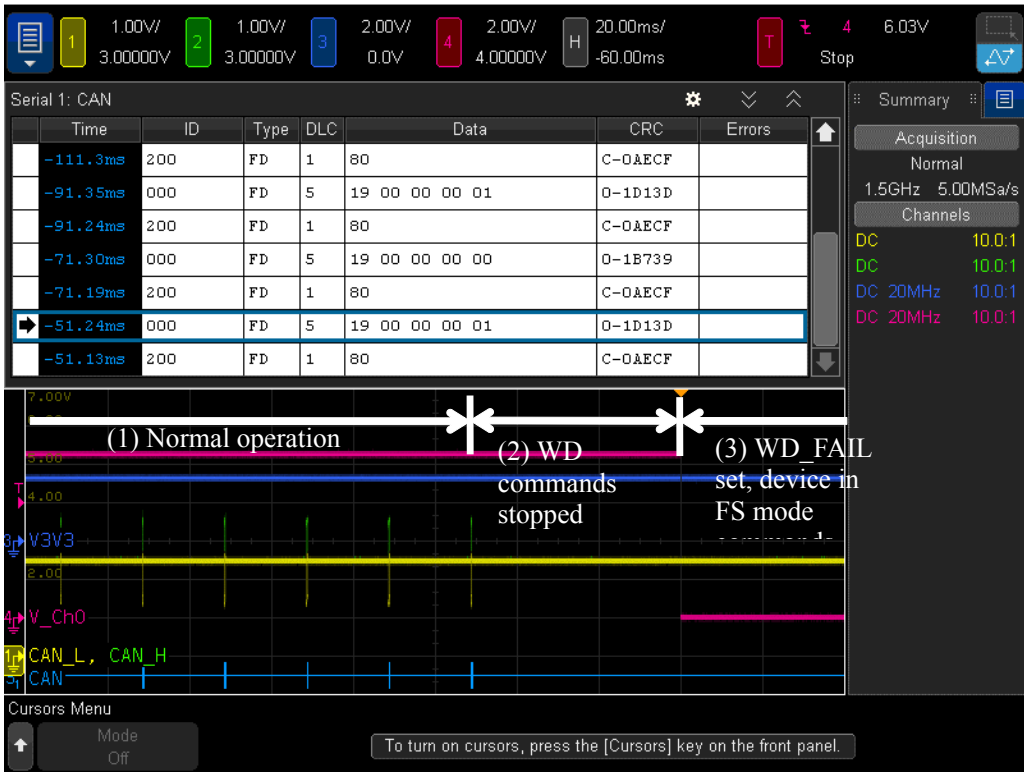
The WD_STATUS bits, available in the status register #3 too, indicates the actual position of the watchdog timeout counter:

- WD_STATUS = 00: 0% < watchdog timeout counter position < 24%
- WD_STATUS = 01: 24% < watchdog timeout counter position < 50%
- WD_STATUS = 10: 50% < watchdog timeout counter position < 74%
- WD_STATUS = 11: watchdog timeout counter position > 74%

This status information can be used for safety checks, to indicate that the counter is running, or to indicate a limit state when the counter is approaching the timeout. Normally, the watchdog should be toggled well before the adjusted timeout (for example, within 50-70% of the timeout period) to allow sufficient margin.

The example in the figure below shows the transition from Bus mode to Fail Safe mode due to the expiration of the WD timeout, after stopping sending WD commands. To better see the transition, Ch0 (red waveform) is configured ON in Bus mode and OFF in Fail Safe mode. In this example, a default timeout of 50 ms is configured. In phase (1), device is in Bus mode and watchdog is properly refreshed (see CAN commands every 20 ms). In phase (2), the sending of the watchdog commands is stopped. After expiring the 50 ms watchdog timeout, the WD_FAIL bit is set and device is switched in Fail Safe mode - see turn-off of the Ch0, phase (3).

Figure 83. Example of transition from Bus to Fail Safe mode due to WD fail



4.9 DAC_RES_FAULT

The DAC_RES_FAULT bit, available in the status register #3 (address 2Eh), indicates open or short-circuit fault on external 61.9 kΩ resistor on R_REF_DAC pin.

- DAC_RES_FAULT = 0: No fault on R_REF_DAC pin
- DAC_RES_FAULT = 1: Open or short-circuit fault on R_REF_DAC pin

The fault also propagates to the FAULT pin and is reflected in the global status byte register in the device error (DE) bit. In the case of DAC_RES_FAULT = 1, all outputs are disabled, regardless of the actual device setting.

The open and short-circuit thresholds are not part of the device specification. Evaluation performed on several samples showed that a fault is reported if the required value of the external resistance of 61.9 kΩ changes by more than +/- 16.5% typically.

4.10 NTC_FAULT

The NTC_FAULT bit, available in the status register #3 (address 2Eh), indicates short-circuit fault on NTC pin:

- NTC_FAULT = 0: no short-circuit fault detected on NTC pin ($V_{NTC} > V_{NTC\_SHT}$)
  - NTC derating activated when the $V_{NTC} < V_{NTC\_TH}$
- NTC_FAULT = 1: short-circuit fault detected on NTC pin ($V_{NTC} < V_{NTC\_SHT}$)
  - Maximum NTC derating applied (output current = 50% of the nominal value)

The fault is also reflected in the global status byte register - DE bit set.

4.11 VREF_PRE_REG_MAX

The VREF_PRE_REG_MAX bit, available in the status register #3 (address 2Eh), indicates a condition when the reference feedback voltage for the external preregulator is set to maximum.

- VREF_PRE_REG_MAX = 0: reference voltage is not set to maximum level
- VREF_PRE_REG_MAX = 1: reference voltage is set to maximum level

If the preregulation algorithm is disabled, the bit is always set because the reference voltage is permanently set to maximum.

4.12 FS bit

This bit is a part of the global status byte (bit n.0) and provides information about the current state of the device:

- FS = 0: device in Bus mode
- FS = 1: device in Fail Safe and standalone mode

In terms of application and diagnostic, it is a key information which should be monitored regularly.

4.13 Thermal warning - TW bit

The TW bit, available in the status register #3 (address 2Eh), is set when junction temperature T_J, read by the internal ADC, rises above the thermal warning temperature threshold (T_{TW}). Thermal warning is also reflected in the global status byte register-global warning (GW) bit set.

- $T_J < T_{TW} – T_{TW\_HST}$: TW = 0, T_J derating inactive
- $T_{TW} – T_{TW\_HST} < T_J < T_{TW}$: Hysteresis band of TW flag, T_J derating active
- $T_J > T_{TW}$: TW = 1, T_J derating active

As soon as the T_J drops below the thermal warning reset threshold (T_{TW}-T_{TW_HYS}), the TW bit is automatically cleared.

Even if a temperature warning is detected, the device as well as output channels are kept functional. To avoid further temperature increase, a T_J thermal derating, reducing the LED current, is automatically started. The derating concept is described more in detail in Section 4.16 LED current derating.

4.14 Thermal shutdown - TSD bit

The TSD bit, available in the status register #3 (address 2Eh), indicates a condition when the junction temperature is above the thermal shutdown temperature threshold (T_{TSD}). In this case, the TSD bit is set, and all the output channels are switched off. The fault pin is pulled low. The device logic and state machine remain active.

Thermal shutdown condition is also reflected in the global status byte register - functional error 1 (FE1) bit set.

As soon as the T_J drops below the thermal shutdown reset threshold (T_{TSD}-T_{TSD_HYS}), the output channels are automatically reactivated, the TSD bit is cleared and the FAUTL pin is released.

4.15 Vs undervoltage - V_{S_UV}

The V_{S_UV} bit, available in the status register #3 (address 2Eh), indicates a condition where the VS supply falls below the VS under-voltage threshold (VS_UV). In this case, all output channels are disabled regardless of the operating mode and the FAULT pin is set low. This feature is implemented to prevent wrong output current due to low battery voltage and to indicate that the VS supply voltage is below the normal operating range, so that some device parameters may deviate from the specified value.

This bit is reflected also in the global status byte register, where device error (DE) bit is set. As soon as the VS supply rises above the under-voltage reset threshold ($V_{S\_UV\_RES}$) the device is automatically reactivated, and the flag cleared.

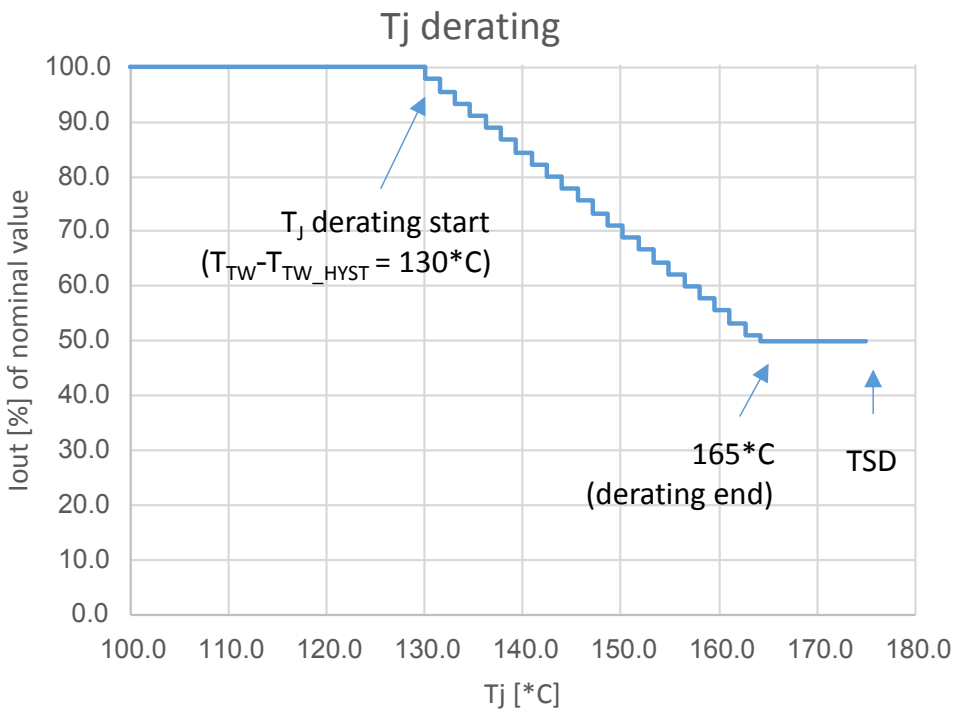
4.16 LED current derating

In case of temperature increasing, the device performs an automatic (O)LED current derating based on the measurement of external NTC voltage as well as on the internal device temperature (T_J). The (O)LED current is reduced proportionally to temperature increase. Both, the T_J and NTC monitoring are contemporarily active. The actual derating is determined by the parameter that would require the most severe derating.

4.16.1 Derating based on internal T_J

T_J based derating is active when device T_J, measured by internal ADC, is above the (T_{TW} – T_{TW_HYST}) derating start threshold. The (O)LED current is reduced proportionally to the temperature increase above the derating threshold. The maximum current reduction is 50% of the nominal LED current. This corresponds to T_J = 165 °C. As can be seen in Figure 84, the output current decreases in discrete steps, determined by the granularity of the T_J ADC reading of 10 mV. The active T_J derating is indicated by the TW flag, except a condition after start of the derating, when the T_J is still below the T_{TW} = 140 °C (inside the T_{TW_HYS} = 10 °C hysteresis band of the TW flag).

Figure 84. **T_J based derating curve - percentage of nominal current**



Note: The absolute minimum output current during the derating is equal to the minimum DAC current setting (1 mA). This means that if the nominal current (before derating starts) is lower than 2 mA, derating is always stop at 1 mA (earlier than 50% of the nominal current).

4.16.2 Derating based on external NTC

The NTC based derating algorithm monitors the voltage at the NTC pin. The derating is active when the NTC voltage is below a configurable VNTC_TH threshold. There are 8 VNTC_TH thresholds available. Using the recommended NTC part number and pull-up resistor value results in the following temperature thresholds:

Table 13. **NTC derating start - configurable VNTC_TH thresholds**

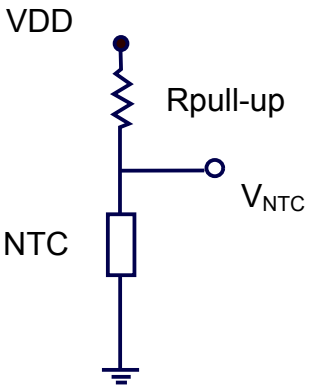
Code [-]	VNTC_TH [V]	VNTC_TH [°C] (With recommended NTC part number and pull-up)
000	2.034	55
001	1.908	60
010	1.783	65
011	1.660	70
100	1.423	80
101	1.207	90
110	1.106	95
111	1.013	100

V_{NTC_TH} is the NTC voltage corresponding to the derating start temperature Rpull-up 0 2.2 kΩ, at VDD = 3.3 V; NTC = 10 kΩ ± 1%, at 25 °C.

NTC part number: MURATA NCU18XH103F6SRB.

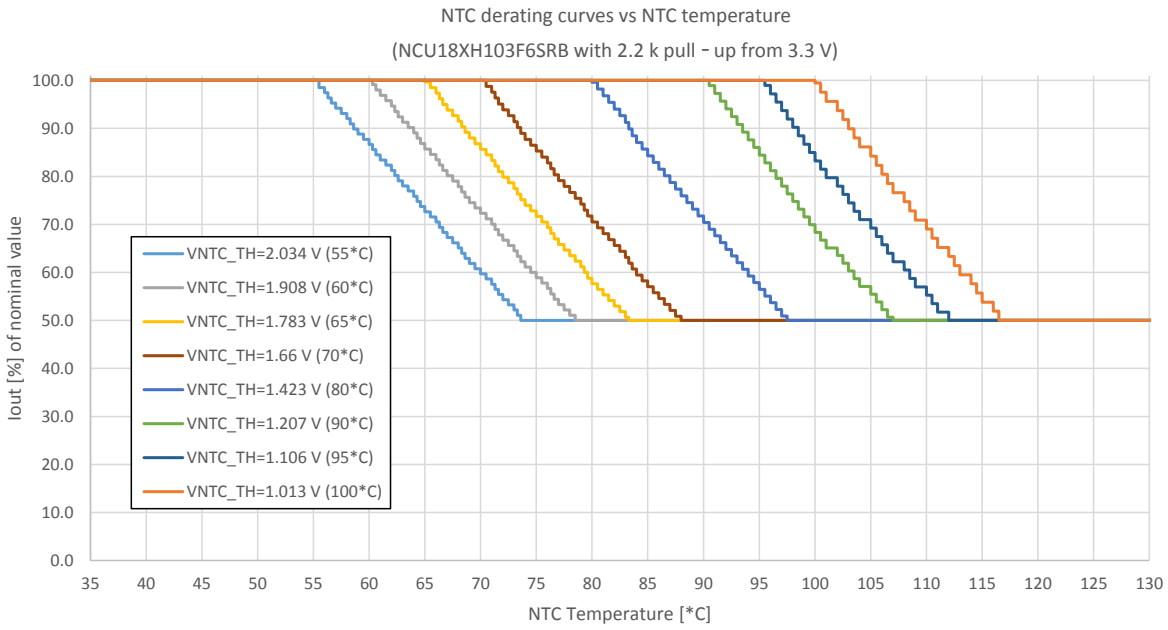
NTC chip type temperature characteristics: B-costant = 3380K ± 0.7%.

Figure 85. **Recommended NTC circuitry**



When the NTC voltage is below the configured threshold, (O)LED current is reduced proportionally to NTC voltage until reaching 50% of the nominal LED current. This corresponds to an NTC temperature approximately 20 °C higher than the one corresponding to V_{NTC_TH}. The resulting derating curves are shown in the following diagram:

Figure 86. **NTC derating curves vs NTC temperature**



As can be seen in Figure 86, the output current decreases in discrete steps, determined by the granularity of the NTC ADC voltage reading of 10 mV. Active NTC derating is indicated by NTC_DER_ACT bit in status register #3 (address 2Eh) and is also reflected in GW bit of the global status byte (the GW is a logic OR combination of TW and NTC_DER_ACT flag).

If necessary, the NTC temperature information can be obtained from ADC read-back of the NTC pin, according to the following NTC circuit characteristic:

Figure 87. V_NTC with total error

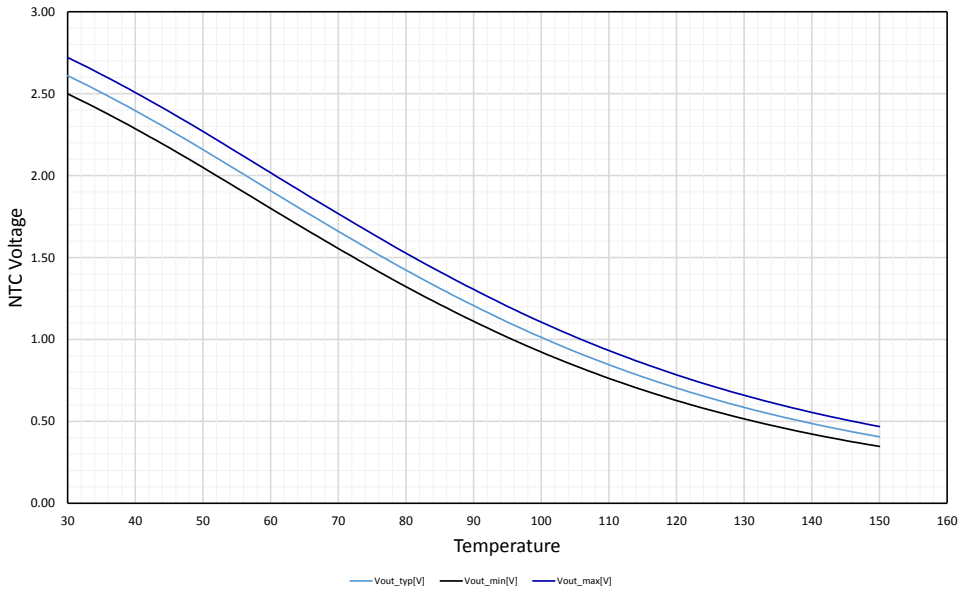
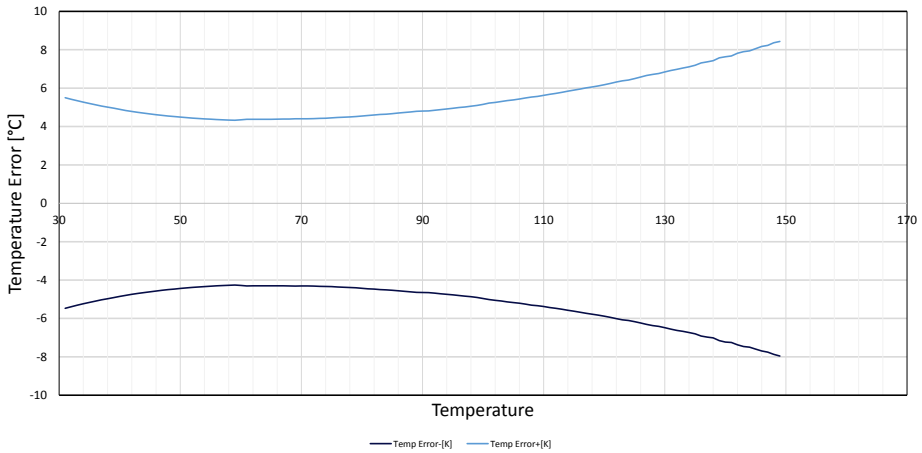


Figure 88. Total temperature error

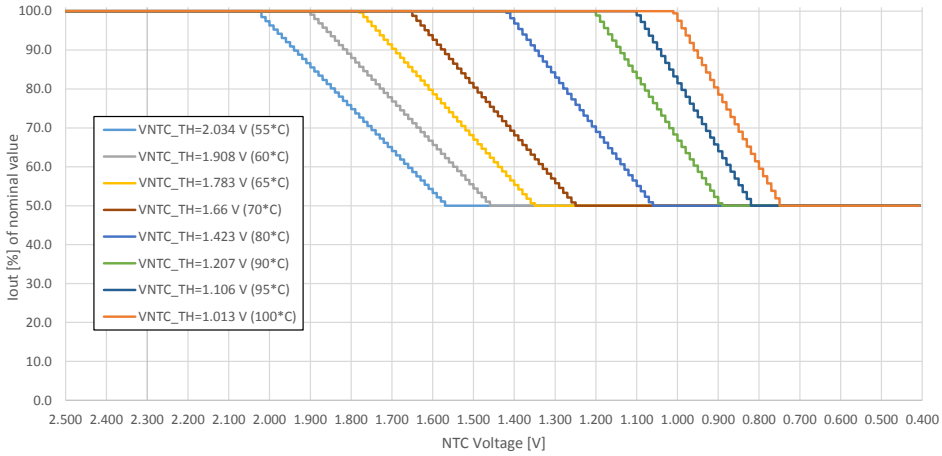


The measurement range of the ADC NTC pin is up to 2.5 V, so the minimum temperature that can be measured with this type of NTC circuit is 35 °C.

As mentioned in the previous text, the NTC derating is designed for a given NTC type and pull-up value (in this case the derating start/end temperature values correspond to the datasheet).

Derating algorithm can also work with different types of NTC/circuits. In this case, however, the temperature values given in the datasheet do not match and need to be determined based on the temperature/voltage characteristics of the NTC circuit used. The derating curves versus NTC voltage are shown in the figure below:

Figure 89. **NTC derating curves vs NTC voltage**



Note: The absolute minimum output current during the derating is equal to the minimum DAC current setting (1 mA). This means that if the nominal current (before derating starts) is lower than 2 mA, derating is always stop at 1 mA (earlier than 50% of the nominal current).

4.17 Application example

This example describes a possible diagnostic scenario in a typical application with 8 L99LDLH32 devices on the BUS (256 OLED pixels).

Requirements:

- PWM duty cycle updated every 25 ms
  - Individual duty cycle value for each channel
  - Same PWM frequency for all channels
- Watchdog timeout 200 ms
- Watchdog refresh 100 ms
- Diagnostic read back period 100 ms
  - All channel related digital diagnostic flags
  - All device related digital diagnostic flags
  - ADC values:
    - T_J, VPRE_REG, VS, NTC and VLEDLOW

CAN communication with the devices is realized by a fixed sequence of periodically sent commands to guarantee the required timing of duty cycle update and diagnostic. The shortest timeframe is a duty cycle update every 25 ms, which defines the main interval of sending the sequence (see Figure 90):

Figure 90. Schedule of diagnostic/control CAN commands sent every 25 ms

Time [ms]	Frame Type	Frame Length		Comment
0.00	Broadcast n.1	0.69	ms	Duty cycle update on Device 0.0, 1.0, 2.0, 3.0
0.69	Broadcast n.2	0.69	ms	Duty cycle update on Device 0.1, 1.1, 2.1, 3.1
1.37	Broadcast n.3	0.69	ms	Duty cycle update on Device 4.0, 5.0, 6.0, 7.0
2.06	Broadcast n.4	0.69	ms	Duty cycle update on Device 4.1, 5.1, 6.1, 7.1
2.75	Burst read m.	0.12	ms	Channel diagnostic digital flags reading - Device x (request) OUT_STATUS, SHT, OL, OUT_SHT_GND
2.86	Burst read s.	0.22	ms	Channel diagnostic digital flags reading - Device x (response) OUT_STATUS, SHT, OL, OUT_SHT_GND
3.08	Burst read m.	0.12	ms	ADC values + other digital flags - Device x (request) Tj, VPRE_REG, VS, NTC_ADC, VLEDLOW, other digital flags
3.20	Burst read s.	0.22	ms	ADC values + other digital flags - Device x (response) Tj, VPRE_REG, VS, NTC_ADC, VLEDLOW, other digital flags
3.42	WD trig m.	0.12	ms	Watchdog trigger - Device x (request)
3.53	WD trig s.	0.08	ms	Watchdog trigger - Device x (response)
3.61	Single clear m.	0.12	ms	Status clear slot - Device x (request)
3.72	Single clear s.	0.12	ms	Status clear slot - Device x (response)
3.84	Sync frame	0.07	ms	Synchronization frame
3.91	Burst read m.	0.12	ms	Channel diagnostic digital flags reading - Device y (request) OUT_STATUS, SHT, OL, OUT_SHT_GND
4.02	Burst read s.	0.22	ms	Channel diagnostic digital flags reading - Device y (response) OUT_STATUS, SHT, OL, OUT_SHT_GND
4.24	Burst read m.	0.12	ms	ADC values + other digital flags - Device y (request) Tj, VPRE_REG, VS, NTC_ADC, VLEDLOW, other digital flags
4.36	Burst read s.	0.22	ms	ADC values + other digital flags - Device y (response) Tj, VPRE_REG, VS, NTC_ADC, VLEDLOW, other digital flags
4.58	WD trig m.	0.12	ms	Watchdog trigger - Device y (request)
4.69	WD trig s.	0.08	ms	Watchdog trigger - Device y (response)
4.77	Single clear m.	0.12	ms	Status clear slot - Device y (request)
4.89	Single clear s.	0.12	ms	Status clear slot - Device y (response)
5.00	Sync frame	0.07	ms	Synchronization frame
5.07	Blank	19.93	ms	Remaining time in 25ms timeslot
Bus load		20%		

The duty cycle of all channels is updated by 4 broadcast commands (1 broadcast command updates a 64 channels) in each 25 ms timeslot. In each timeslot, the diagnostic flags of 2 devices are read (device x, device y), so it takes 4 timeslots (100 ms) to collect diagnostic information from all devices. Similarly, for the watchdog refresh, there are 2 devices updated per timeslot, which results in watchdog refresh every 100 ms. There are 2 clear commands present in each timeslot, to allow clearing faults if necessary. To simplify the software and keep the number of commands in the sequence fixed, these commands are always sent even if no fault needs to be cleared. In this case, clear commands are sent with a data mask of 00:00:00:00h, so it has no effect.

This command sequence timeslots results in a 20% BUS load. This means that there is sufficient margin to increase the diagnostic/PWM refresh rate and/or to add additional commands (for example, PWM duty cycle readback and chain configuration readback, for ASIL applications).

The above-described application example was implemented in a real environment, where 4 evaluation boards (each with 2 devices) were connected to one CAN bus. The Figure 91 shows a complete diagnostic cycle consisting of four 25 ms timeslots. The next figures show in detail the listing of the commands in each timeslot. The configured ResponderIDs are: 000h, 004h, 008h, 00Ch, 010h, 014h, 018h, and 01Ch.

Figure 91. **CAN bus screenshot - complete diagnostic cycle (four 25 ms timeslots)**

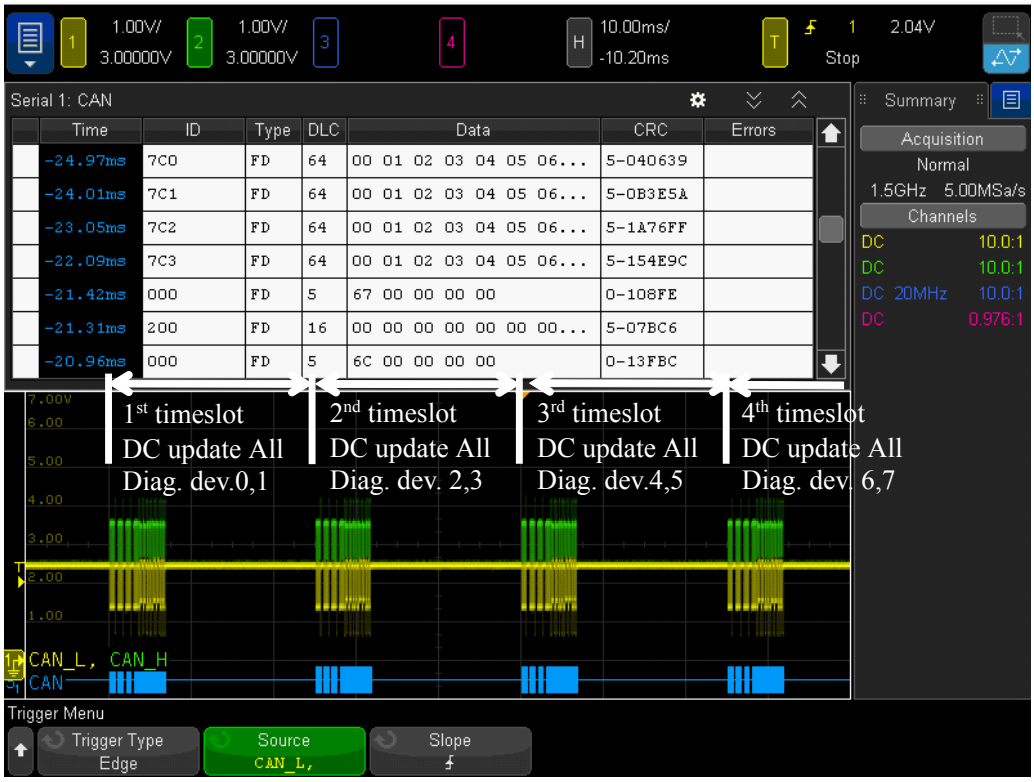


Figure 92. **CAN bus screenshot - 1st timeslot detail**

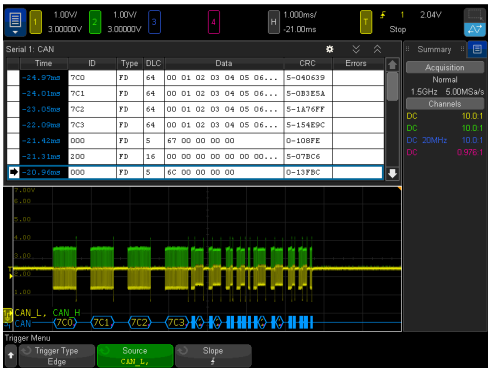


Figure 93. **1st timeslot - listing of commands**

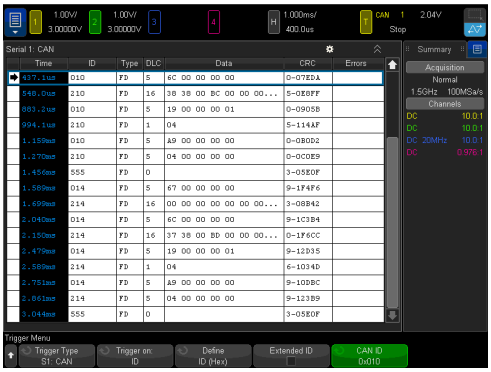


Figure 94. CAN bus screenshot - 2nd timeslot detail

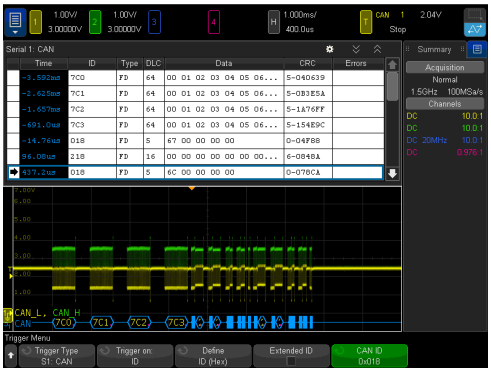


Figure 95. 2nd timeslot - listing of commands

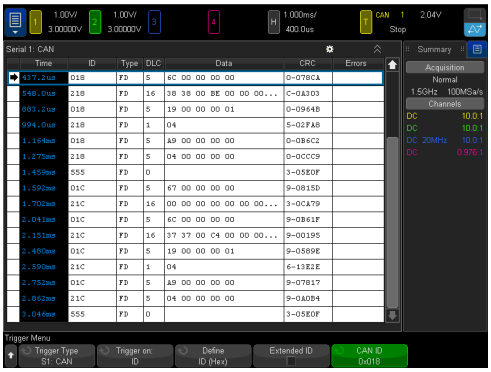


Figure 96. CAN bus screenshot - 3rd timeslot detail

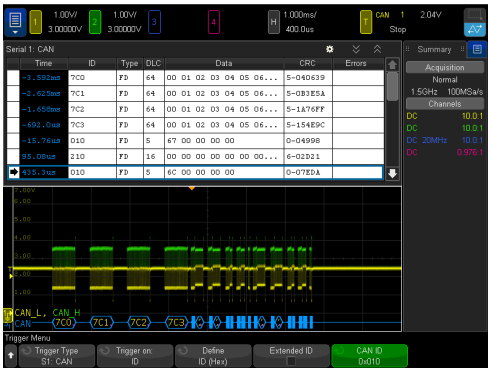


Figure 97. 3rd timeslot - listing of commands

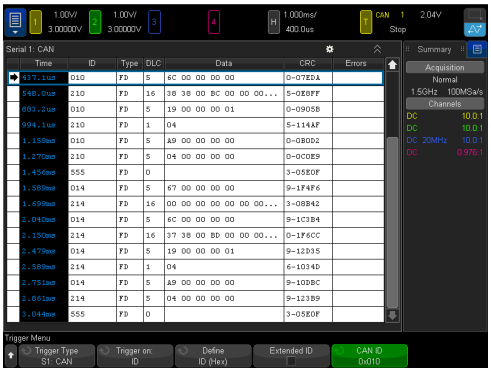


Figure 98. CAN bus screenshot - 4th timeslot detail

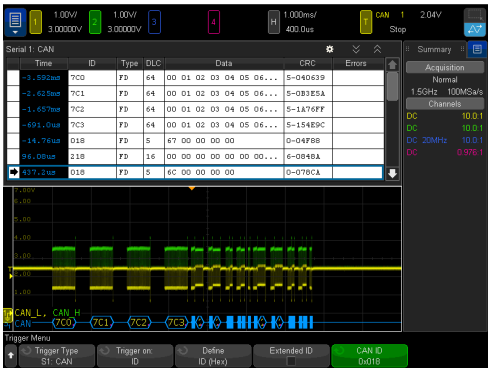
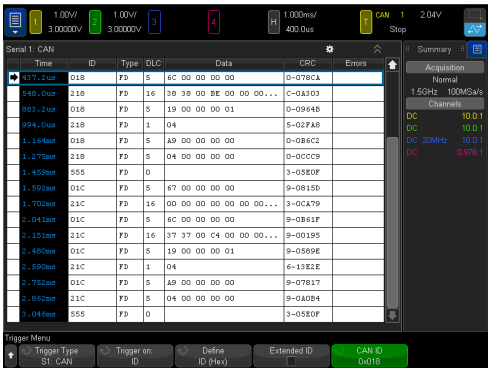


Figure 99. 4th timeslot - listing of commands



5 O(LED) supply voltage regulation

The use of multichannel linear current regulator is often limited by the total power dissipation. In the device, the application can take into account the reasonable efforts for thermal management and heat spreading. As a general rule, assuming that the die pads of the QFN7x7 48L are connected through thermal vias to a continuous copper plane, the total power dissipation allowed in a single device is about 2.0 - 2.5 W. In order to stay within this power dissipation limit, it is essential to check the voltage drop on each of the 32 linear current regulators. The output current regulator, to ensure the correct operation at all operating temperature and current conditions, requires a minimum voltage drop (VOUTx_DROP). This minimum VOUTx_DROP must be ensured through the application of a sufficiently high precontroller supply voltage. Due to the number of channels and the sum of the current to be regulated, for efficiency reason, it is assumed that the pre regulator is a DC/DC switching converter. In overtemperature, the supply voltage required for the LEDs (O) could be higher or lower than the battery voltage. In this case, a converter with step-up & down capability is required, for example a SEPIC topology. In order to avoid excessive power dissipation, as already explained above, the power supply voltage of the pre regulator must be neither too low nor too high. Things get even more complicated because the voltage drop across (O) LED varies with the type, the current, the temperature and the aging. This complex situation, dependent on many variables, can be solved thanks to the L99LDLH32.

5.1 VREF_PRE_REG feature description

The L99LDLH32 features one analog pin (VREF_PRE_REG) providing a voltage proportional to the maximum output voltage of active channels plus the required drop voltage across the current regulator required for proper regulation. This signal can be used in combination with an analogue discrete circuit to directly close the feedback loop with an external DC/DC converter. This feature can be enabled or disabled by the VREF_PRE_REG configuration bit in the RAM respectively FTP memory map.

The VREF_PRE_REG generation is based on an algorithm able to verify which string has the highest drop voltage, adding a minimum drop on the integrated linear switches to grant the proper current regulation.

Two different kinds of regulation algorithm can be selected through a specific configuration bit in the NVM area section accessible in the STMicroelectronics testing flow: an open loop control - and a closed loop regulation algorithm for providing the analog feedback. While the open loop control method is the best solution when several L99LDLH32 devices share the same DC/DC regulator, the closed loop regulation algorithm is able to compensate even for tolerances of external components and can further minimize power dissipation. It works when only one L99LDLH32 is connected to the DC/DC regulator.

Both methods are deeper described in the following chapters.

5.1.1 Open loop control algorithm

The generated analog feedback is proportional to the sum of the highest (O)LED string voltage and the target output voltage drop (~1.4 V).

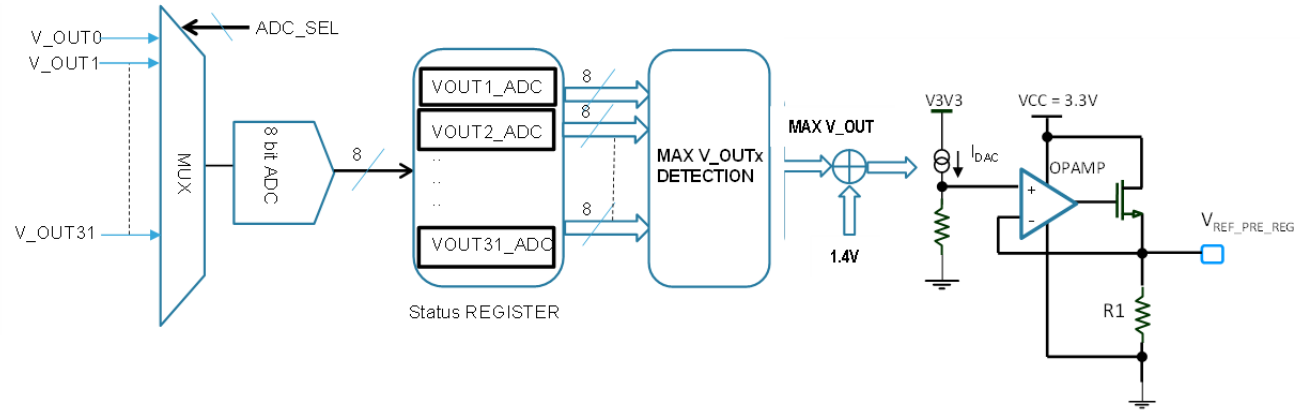
The result is expressed by the following equation:

Equation 1

$$V_{REF\_PRE\_REG} = K \times (V_{OUTMAX} + 1.4) \quad (1)$$

In L99LDLH32 the proportional “K” factor is equal to 0.05.

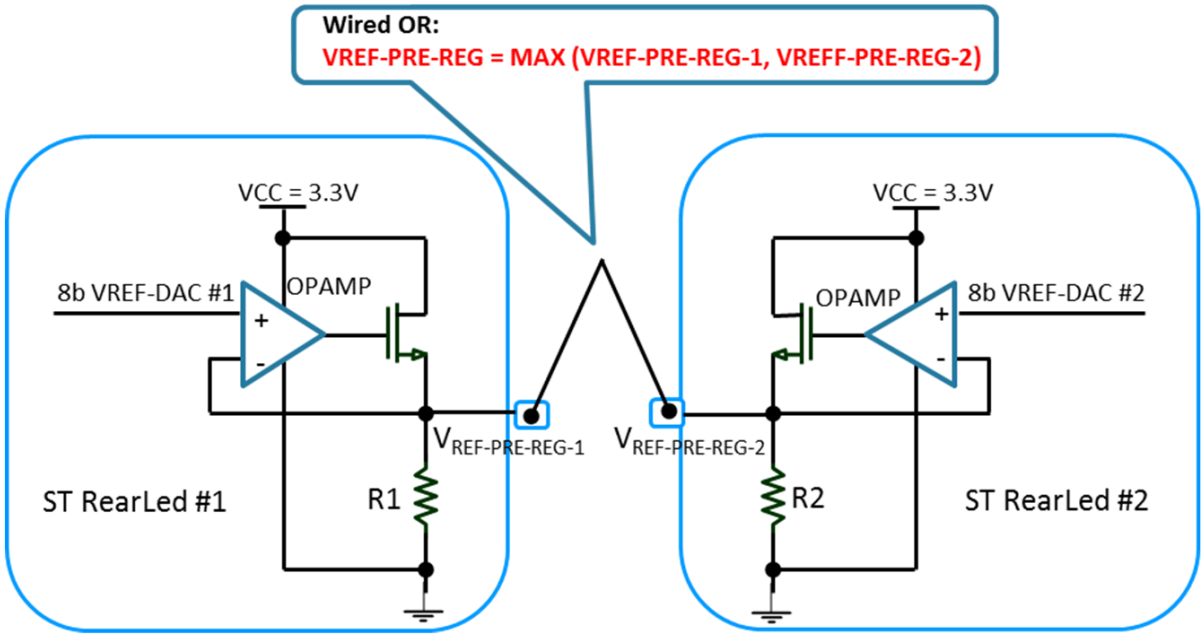
Figure 100. $V_{REF\_PRE\_REG}$ - generation concept



The output voltage of each of the maximum 32 active channels is sampled by the integrated ADC once during PWM on-phase every PWM period. The sampling is masked for all those channels with a PWM on-time shorter than t_DIAG_BLANK. The control block determines the maximum VOUT and adds the desired drop voltage (1.4 V). Finally, the VREF_PRE_REG voltage is calculated according to 1 and the output DAC is set accordingly.

As it can be seen from Figure 100. $V_{REF\_PRE\_REG}$ - generation concept, the VREF_PRE_REG output signal is buffered. For applications with a common DC/DC converter, this allows to wire-OR the VREF_PRE_REG output signal from several devices to provide only the highest reference voltage back to the DC/DC (see Figure 101. Wired-OR connection between two or more (O)LED drivers). The output with the highest VREF_PRE_REG, being applied also to the inverting pin of the other device with low reference, will force to depolarize the output MOS, disabling the buffer having the lower voltage.

Figure 101. **Wired-OR connection between two or more (O)LED drivers**



It is to be noted, regardless of the PWM frequency, that the VREF_PRE_REG signal is updated every 21 ms. This delay time has been introduced to make sure the VREF_PRE_REG signal is changed more slowly than the load regulation time constant of the DC/DC converter (bandwidth), thus avoiding interference of the DC/DC own regulator loop with this algorithm.

VREF_PRE_REG voltage operating range is between 0.0 V minimum and 2.0 V maximum.

Figure 102. $V_{REF\_PRE\_REG}$ - flow chart illustrates the working principle of the open loop algorithm. When the VREF_PRE_REG control bit in the device configuration #3 register is set, the algorithm is disabled and the maximum VREF_PRE_REG voltage is applied. If the VREF_PRE_REG control bit in the device configuration #3 register is cleared, the algorithm is enabled. Then, as long as all output channels are disabled and the algorithm is not running, a minimum VREF_PRE_REG voltage of 0.25 V is applied, in order:

1. not to interfere with running algorithm on other ICs sharing the same DC/DC voltage regulator
2. always ensure the DC/DC voltage regulator supplies a certain minimum voltage

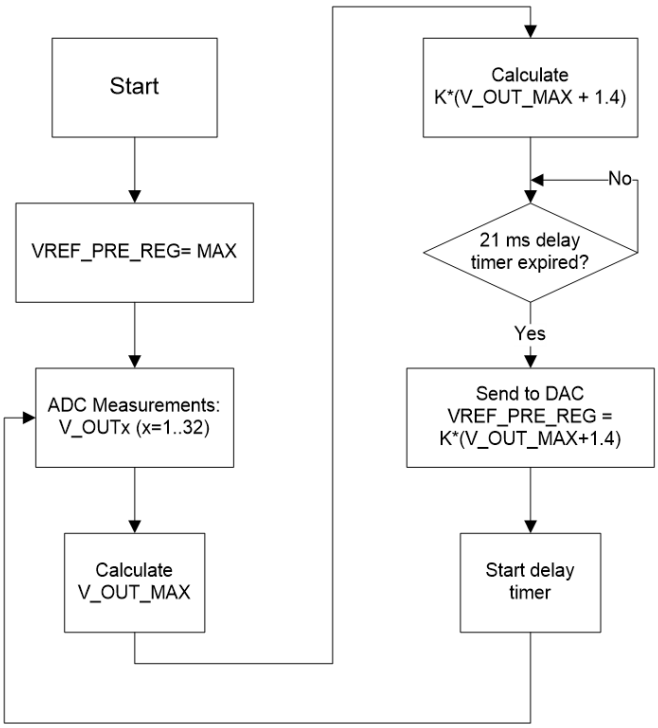
As soon as a channel is enabled, the VREF_PRE_REG voltage is set to its maximum value. This is done, because the forward voltage of the string connected to this channel cannot be known a priori, therefore the preregulator voltage is set to its maximum value to generate the maximum preregulator voltage to ensure a sufficient supply voltage is applied to the string.

A side remark: a condition when VREF_PRE_REG voltage is set to its maximum value can be detected by the application checking the VREF_PRE_REG_MAX status bit in the device status register #3.

Then, the output voltage of all active channels is measured and the maximum out of it is determined. Based on this value the device calculates VREF_PRE_REG voltage, checks if the 21 ms delay timer is expired and if this is the case, provides the calculated code to the DAC.

In case an open-load event occurs on any channel while the algorithm is running and while the device is validating the open-load condition, the algorithm restarts from the initial step, setting the VREF_PRE_REGx to the maximum value, trying to solve the open-load condition. Inactive channels and channels which are latched off due to a validated fault condition (open load, short-circuit, or short-circuit to GND) won´t be taken into account for the VOUT_MAX determination. If the on time during a PWM period of any channel is too short to allow ADC conversion (basically t_on < t_DIAG_BLANK + 3 µs), the VREF_PRE_REG voltage is set to its maximum value.

Figure 102. $V_{REF\_PRE\_REG}$ - flow chart



The device features a Power Good threshold supervision. This Power Good threshold supervises the VPRE_REG voltage (DC/DC regulator output voltage). When the Power Good threshold is reached, the diagnostic is enabled, otherwise it remains disabled. Only when the VPRE_REG voltage is sufficiently high, the L99LDLH32 output current regulators can source the configured load current. If the VPRE_REG voltage is too low, a false open load failure could be detected. This is avoided by the Power Good threshold supervision. The VREF_PRE_REG open loop control algorithm may lead to situations in which VPRE_REG voltage is temporarily set below the configured Power Good threshold. To avoid application malfunction in such scenario, the following steps need to be applied:

1. the device autonomously disables the supervision of the Power Good threshold while the VREF_PRE_REG open loop control algorithm is running and VREF_PRE_REG is not set to VREF_PRE_REG_MAX. The PG_NOT_VPRE_REG bit in the device status register #3 won´t be refreshed as long as the algorithm is running
2. whenever PG_NOT_VPRE_REG is set, the application shall temporarily deactivate the Pre-Regulator control algorithm, by two consecutive UNICAST WRITE frames to disable and enable the VREF_PRE_REG bit.

Figure 103. $V_{REF\_PRE\_REG}$ - operating example

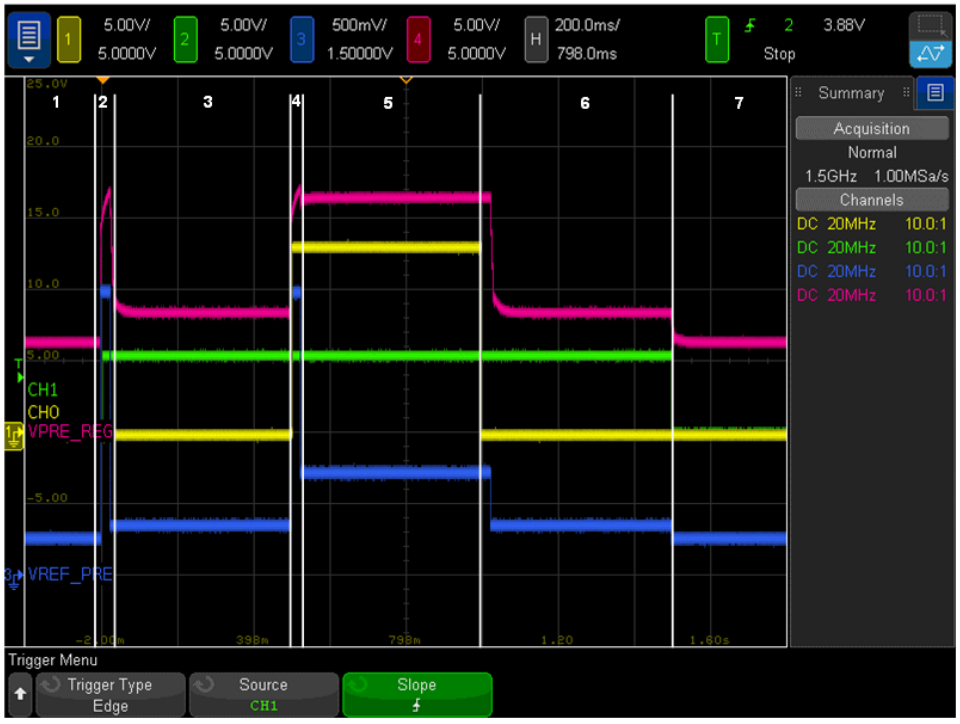
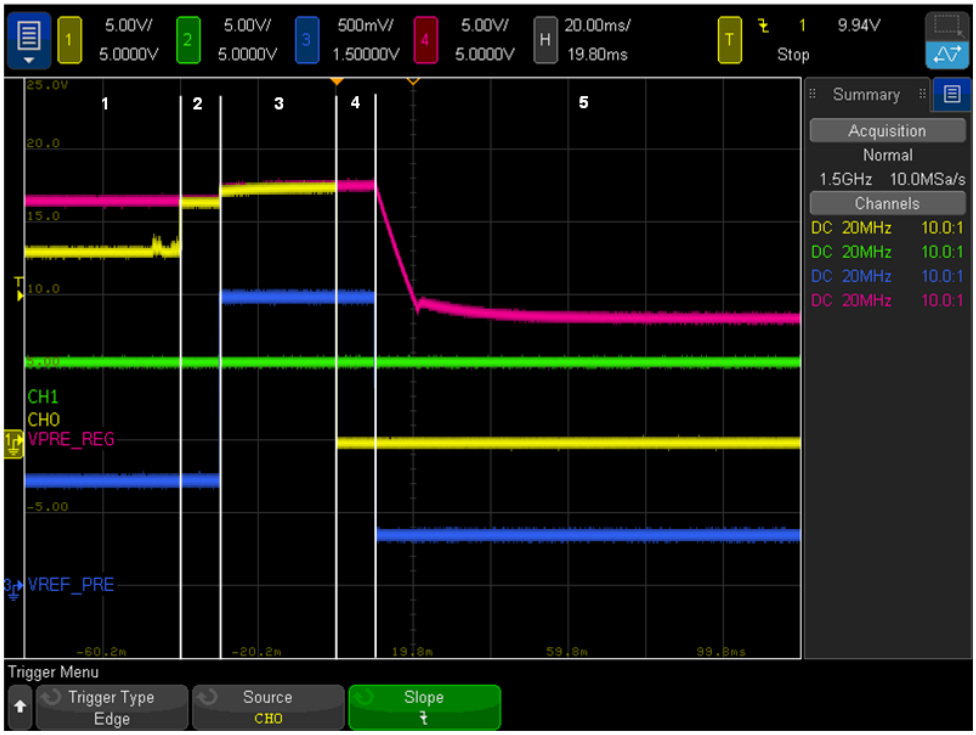


Figure 103. $V_{REF\_PRE\_REG}$ - operating example illustrates the working principle of the VREF_PRE_REG generation. In Phase 1 the open loop control algorithm is enabled, but all output channels are off. VREF_PRE_REG is set to its minimum voltage when algorithm is not running. Phase 2, channel 1 driving a short string is turned on. Consequently, VREF_PRE_REG is set to its maximum voltage, the DC/DC regulator tries to follow it. Phase 3, VOUT_MAX was determined and VREF_PRE_REG is set according to Equation 1. Phase 4, channel 0 driving a long string is activated and VREF_PRE_REG voltage is set again to the maximum voltage. Phase 5, VOUT_MAX (now ch0) was determined and VREF_PRE_REG is set accordingly to Equation 1. Phase 6, channel 0 is turned off, and VREF_PRE_REG is reduced to optimize the Power Dissipation considering the active channel 1. Phase 7, channel 1 as last active channel is turned off, VREF_PRE_REG is set to its minimum when algorithm is not running.

Figure 104. $V_{REF\_PRE\_REG}$ - open load

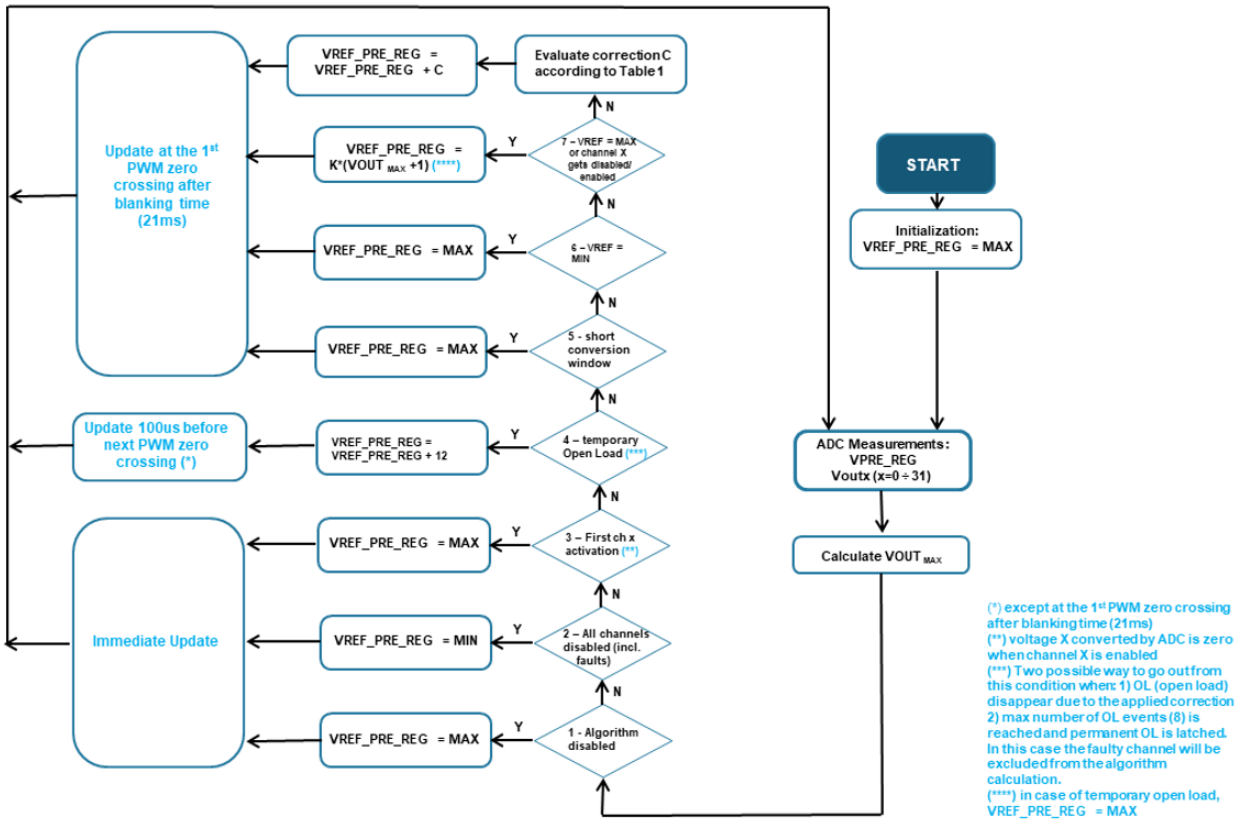


While actuator fault conditions like shorted (O)LEDs or short to GND can be easily managed by the control loop, in fact such fault conditions are “just” lead to a lower VOUT, an open load fault is more critical to handle. Figure 104. $V_{REF\_PRE\_REG}$ - open load depicts such situation: Phase 1: two channels, channel 1 with a short string and channel 0 with a long string are active with VREF_PRE_REG set to K*(VOUT_CH0 +1.4 V). Phase 2, an open load condition occurs at ch0, its output voltage is rising to the DC/DC regulator output voltage (VPRE_REG). Phase 3, the open loop control algorithm detected the open load condition and raised VREF_PRE_REG to its maximum voltage, VPRE_REG is increasing to its maximum voltage defined by the external circuit, while the output is maintained on as long as the temporary open load condition is validated/ debounced by the device. Phase 4, the open load validation is completed by the device, the open load is confirmed as permanent fault and the corresponding channel is latched off. Phase 5, the open loop control algorithm sets the VREF_PRE_REG voltage according to Equation 1, disregarding the latched channel(s).

5.1.2 Closed loop regulation algorithm

The closed loop regulation algorithm can be applied when only one L99LDLH32 device is supposed to control the DC/DC regulator output voltage. In comparison to the open loop control algorithm there are some fundamental differences which are explained in the following figure.

Figure 105. **Closed loop regulation flow chart**



The initial steps are quite the same as in the open loop case. To avoid potential false Open Load generation, the external pre-regulator voltage has to be higher than the minimum supply voltage required by the highest LED string.

Then, at the start of the algorithm, the VREF_PRE_REG voltage is initialized to its maximum value setting the external pre-regulator to the maximum value. After this first step, the internal ADC measures the $VOUT_X$ of all active channels.

The maximum (O)LED voltage drop is coincident with the VOUT_MAX, so the engine calculates the maximum value among the $VOUT_X$ (x = from 0 to 31).

The next three steps are again like in the open loop concept, except the VREF_PRE_REG voltage will be updated always immediately, not waiting the elapse of the 21 ms delay timer.

The first major difference is the temporary open load handling. As soon as a temporary open load condition is detected, the drop voltage between pre-regulator output voltage (VPRE_REG) and the L99LDLH32 channel output voltage (VOUTx) drop below the VOL_TH threshold. A temporary open load event is triggered internally in the device and the open-load validation procedure is starting. During each PWM period, the drop voltage is measured again and, as long as the temporary open load condition is confirmed, the VREF_PRE_REG voltage is increased every PWM period by 94 mV. The process stops as soon as the VPRE_REG voltage is increased sufficiently to resolve the open load condition, or the open load validation process is completed by confirming the open load fault as permanent. In the latter case the channel is latched off and VREF_PRE_REG is set to its maximum (see step 7 of the flow chart).

If the on-time of any channel during a PWM period is too short to allow the ADC sampling of the output voltage, basically when t_ON < t_DIAG_BLANK + 3 µs, VREF_PRE_REG is set to the maximum value (flow chart step 5). The open loop formula is applied if condition 7 of the flow chart (Figure 105. Closed loop regulation flow chart) is fulfilled (regulator running at its maximum value) but only if all the previous conditions at the highest priority are not satisfied.

After this step, and in case none of the conditions 1 to 7 are matched, the difference (VPRE_REG – $VOUT_{MAX}$) is evaluated to determine a correction factor “C” as follows:

Table 14. Correction factor “C” for closing the loop

VPRE_REG – $VOUT_{MAX}$	Correction (C)
$<V_{OL\_TH}$	+94.1 mV
< 0.63 V	+31.4 mV
< 1.25 V	+7.8 mV
≤ 1.57 V	0
≤ 1.88 V	-7.8 mV
≤ 2.198 V	-15.6 mV
> 2.198 V	-31.4 mV

The value of VREF_PRE_REG is updated adding or subtracting the “C” value at the first PWM duty cycle after 21 ms of blanking time. This correction factor acts closing the feedback loop. Regardless of the tolerances of the external circuit components, the closed loop algorithm moves the VREF_PRE_REG voltage accordingly to generate a voltage drop of about 1.4 V across the channel with VOUT_MAX.

The following figure depicts the operating principle of the closed loop regulation algorithm based on an application example turning on/off sequentially a short and a long load string. The different steps of actions are explained in Table 15. Stepwise explanation. To challenge the capability of the closed loop algorithm to compensate even a significant error on the gain factor of the external circuit, a positive error of approximately 5 V was intentionally introduced in the circuit.

Figure 106. $V_{REF\_PRE\_REG}$ - operating example closed loop algorithm

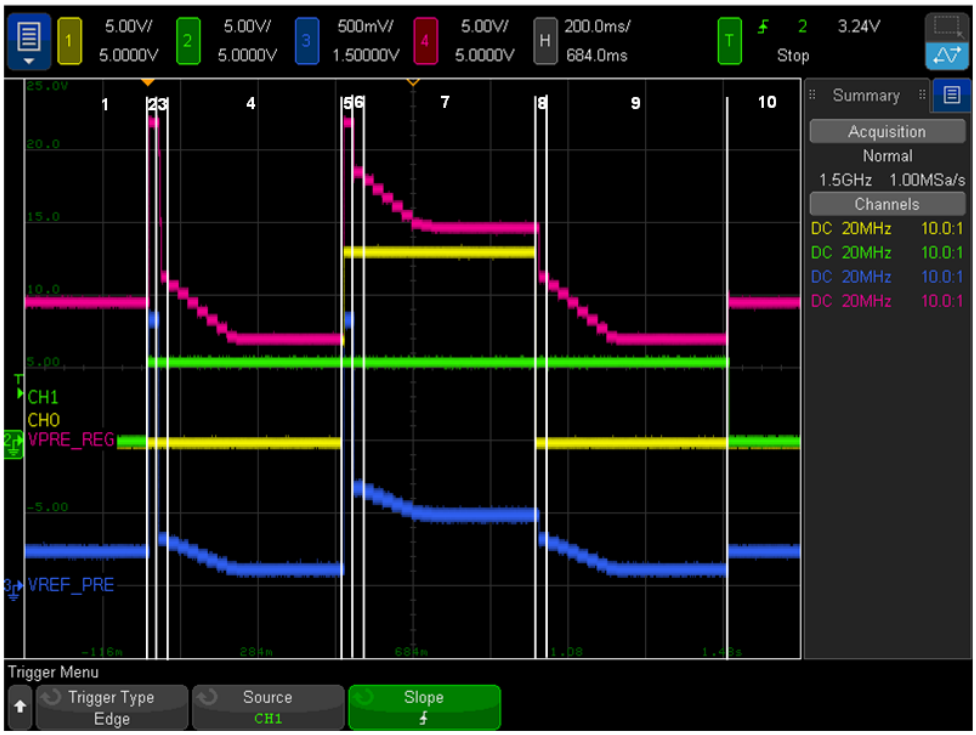


Table 15. Stepwise explanation

Step	Circuit behaviour
1	All channels off, VREF_PRE_REG set to minimum voltage when algorithm is not running according to condition 2 of the flow chart
2	Channel 1 short string turned on, VREF_PRE_REG set to maximum according to condition 3 of the flow chart
3	Open loop formula (Equation 1) applied according to condition 7 of the flow chart - ~ 7 V difference between VPRE_REG and VOUT_MAX
4	Closed loop adaptation with correction steps “C” applied. Circuit error is compensated by the loop. VPRE_REG – VOUT_MAX stepwise reduced to ~ 1.5 V
5	Channel 0 long string turned on, VREF_PRE_REG set to maximum according to condition 3 of the flow chart
6	Open loop formula (Equation 1) applied according to condition 7 of the flow chart - ~ 7 V difference between VPRE_REG and VOUT_MAX
7	Closed loop adaptation with correction steps “C” applied. Circuit error is compensated by the loop. VPRE_REG – VOUT_MAX stepwise reduced to ~ 1.5 V
8	Channel 0 long string is deactivated. Open loop formula (Equation 1) applied according to condition 7 of the flow chart - ~ 7 V difference between VPRE_REG and VOUT_MAX
9	Closed loop adaptation with correction steps “C” applied. Circuit error is compensated by the loop. VPRE_REG – VOUT_MAX stepwise reduced to ~ 1.5 V
10	Channel 1 short string is deactivated. All channels off. VREF_PRE_REG is set to minimum voltage when algorithm is not running according to condition 2 of the flow chart

How the closed loop control algorithm manages an open load fault case is illustrated in the following figure.

Figure 107. $V_{REF\_PRE\_REG}$ - operating example closed loop algorithm in open load

Table 16. **Stepwise explanation of previous figure**

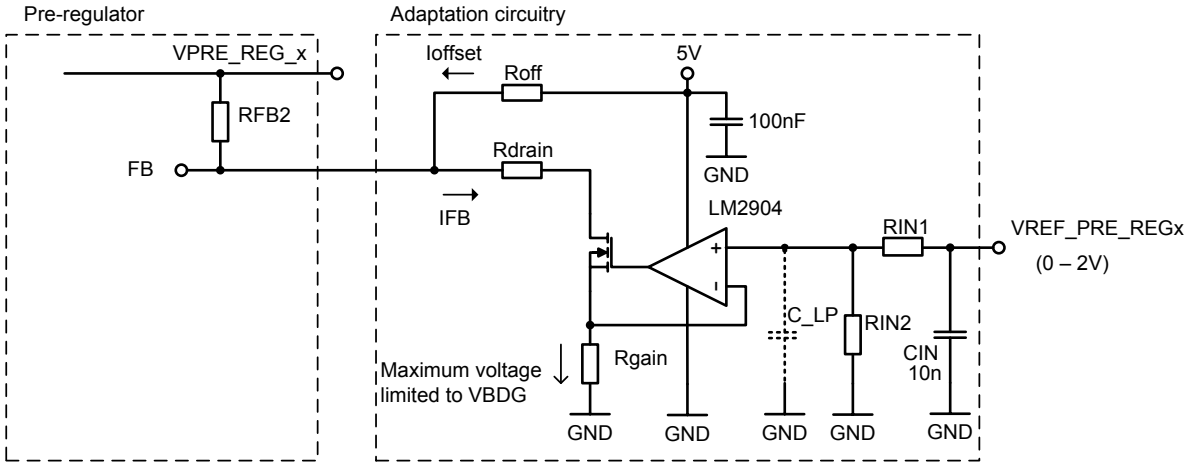
Step	Circuit behaviour
1	Channel 0 & 1 on, circuit in closed loop adaptation
2	Open Load on Channel 0. Open load validation procedure running in the device. VREF_PRE_REG is stepwise increased every PWM period
3	Open Load validated on Channel 0. Channel latched off. VREF_PRE_REG set to maximum
4	VREF_PRE_REG set according to open loop formula with Channel 1 as VOUT_MAX
5	Closed loop adaptation with correction steps “C” applied. Circuit error is compensated by the loop. VPRE_REG – VOUT_MAX stepwise reduced to ~ 1.5 V

5.2 HW design of feedback loop - circuit schematics

Most of the DC/DC regulators on the market do not offer an adjustable reference voltage accessible through a dedicated pin, but the majority of the devices offer a feedback pin. The voltage on this feedback pin is compared against a precise reference (in many cases the band gap voltage) and the error is amplified. If the feedback voltage is higher than the reference voltage, the DC/DC regulator tries to reduce its regulated output voltage and vice versa. Therefore, the VREF_PRE_REG voltage generated by the L99LDLH32 cannot be connected directly to the DC/DC regulator, but an external circuit is needed to convert the VREF_PRE_REG voltage into a feedback voltage.

5.2.1 Circuit schematics for open loop control algorithm

Figure 108. **Circuit schematics connecting $V_{REF\_PRE\_REG}$ signal(s) to feedback pin**



The adaptation circuitry works as voltage → current converter. The VREF_PRE_REG voltage, provided by the L99LDLH32 device(s), calculated as per Equation 1, is translated to a current sunk from the upper resistor of the feedback divider RFB2 (bottom resistor is not present) of the DC/DC (SEPIC) converter. The voltage → current conversion ratio is defined by the Rgain resistor and by the RIN1 and RIN2 divider on the input. This divider is required to keep the OPAmp non-inverting input voltage below the feedback voltage to allow proper regulation. To dimension the RIN divider properly, it has to be considered the internal impedance of the VREF_PRE_REG signal of ~ 700 Ω. The Rdrain limits the maximum possible feedback current and so limits the maximum SEPIC output voltage to the required level. The C_LP capacitor on the non-inverting input of the OPAmp is optional. Together with RIN1 it forms a low pass filter to slow down the feedback loop. Depending on the type of the DC/DC converter and its bandwidth it might be needed or not. As implementation guideline, the time constant of this low pass filter should not exceed 3 ms. The resistors should be dimensioned according to the following formulas:

Equation 2

$$R_{IN\_Ratio} = \frac{R_{IN2}}{R_{IN2} + R_{IN1} + 700\Omega} \quad (2)$$

Equation 3

$$R_{gain} = K \cdot R_{INRatio} \cdot R_{FB2} \quad (3)$$

Equation 4

$$R_{drain} = \frac{V_{FB} \cdot R_{FB2}}{V_{PRE\_REG\_MAX}} - R_{gain} \quad (4)$$

An additional offset current can be added by the R_{OFF} resistor in order to compensate the V_{FB} voltage shift of the FB node (so to align the 0V VREF_PRE_REG input to the 0V VPRE_REG output)

Equation 5

$$R_{OFF} = \frac{R_{FB2} \cdot (5V - V_{FB})}{V\_FB} \quad (5)$$

Having defined those values, the VPRE_REG voltage is calculated as

Equation 6

$$V_{PRE\_REG} = VREF_{PRE\_reg} \cdot \frac{R_{INRatio} \cdot R_{FB2}}{R_{gain}} + V_{FB} - I_{offset} \cdot R_{FB2} \quad (6)$$

5.2.1.1 *Application example (1)*

Figure 109. Calculation of resistor dimensioning for application example 1

External adaptation circuitry - component value calculator			
VOFFSUPPLY	5	V	Offset resistor supply voltage (for Roff calculation)
VBDG	0,7	V	Reference voltage of external preregulator
VPRE_REG max.	18	V	Maximum output voltage of the pre-regulator (for Rdrain calculation)
K	0,05	-	K factor (0.1 for L99LDLL16, 0.05 for L99LDLH32)
RFB2	43	kohm	Upper resistor of feedback divider
RIN1	27	kohm	Upper resistor of control voltage divider
RIN2	33	kohm	Lower resistor of control voltage divider
Vref_pre impedance	0,7	kohm	Internal resistance of Vref_pre output
RIN_ratio	0,543657331	-	Control voltage divider ratio (out/in)
Rgain	1,168863262	kohm	Current sink shunt resistor
Roff	264,1428571	kohm	Offset current definition resistor
Rdrain	0,50335896	kohm	Maximum VPRE_REG voltage definition resistor

As it can be seen from the above calculation example, the offset resistor calculation results in a quite high number. As it often happens in automotive design, resistors higher than 100 kΩ are not tolerated in circuit schematics. Therefore the offset resistor can be even skipped producing a small error.

Without this resistor, the SEPIC output voltage will be shifted by V_{FB} higher. A positive shift is anyway required, considering all the tolerances of components, since the circuitry must provide enough output voltage > (V_{OUT_MAX}+1.4 V) also in worst case combination of component tolerances. All these calculations are available in a separate calculator:

Figure 110. Tolerances for worst case calculation (example 1)

Vref pre reg Impedance	40	% tolerance
RFB2	1	% tolerance
Bandgap	1	% tolerance
Voff_supply	3	% tolerance
Roff	1	% tolerance
Rgain	1	% tolerance
Rdrain	1	% tolerance
RIN1	1	% tolerance
RIN2	1	% tolerance
K	7	% tolerance
OpAmp offset	2	mV

Figure 111. Worst case SEPIC output voltage spread considering all tolerances (example 1)

Vout (OLED)	8 V			
	min	typ	max	
Vref_pre (floating)	0,437 V	0,471 V	0,503 V	Vref_pre (floating)=(Vout+1.4)*k
V_Rgain	0,232 V	0,256 V	0,279 V	V_Rgain=Vref_pre*RIN2/(Vref_resistance+RIN1+RIN2)
IFB	0,209 mA	0,233 mA	0,256 mA	IFB=V_Rgain/Rgain
Ioffset	0,000 mA	0,000 mA	0,000 mA	Ioffset=(Voff_supply-Vbdg)/Roff
Vpre_reg	9,598 V	10,699 V	11,839 V	Vpre_reg=(IFB-Ioffset)*RFB2+VBDG
Vpre_reg_max	17,103 V	17,610 V	18,131 V	Vpre_reg_max=(VBDG/(Rgain+Rdrain)-(VOFFSUPPLY-VBDG)/Roff)*RFB_2+VBDG

	Vref_pre (floating) [V]			Vpre_reg [V]			Vpre_reg - Vout [V]			V_Rgain [V]			IFB [mA]		
Vout (OLED) [V]	min	typ	max	min	typ	max	min	typ	max	min	typ	max	min	typ	max
4	0,251	0,270	0,289	5,78	6,44	7,14	1,78	2,44	3,14	0,133	0,147	0,161	0,119	0,133	0,148
5	0,298	0,320	0,342	6,73	7,50	8,31	1,73	2,50	3,31	0,158	0,174	0,191	0,142	0,158	0,175
6	0,344	0,370	0,396	7,69	8,56	9,49	1,69	2,56	3,49	0,183	0,201	0,220	0,164	0,183	0,202
7	0,391	0,420	0,449	8,64	9,63	10,66	1,64	2,63	3,66	0,207	0,228	0,250	0,187	0,208	0,229
8	0,437	0,470	0,503	9,60	10,69	11,84	1,60	2,69	3,84	0,232	0,256	0,279	0,209	0,232	0,256
9	0,484	0,520	0,556	10,55	11,75	13,02	1,55	2,75	4,02	0,257	0,283	0,309	0,232	0,257	0,283
10	0,530	0,570	0,610	11,51	12,81	14,19	1,51	2,81	4,19	0,282	0,310	0,338	0,254	0,282	0,310
11	0,577	0,620	0,663	12,46	13,88	15,37	1,46	2,88	4,37	0,307	0,337	0,368	0,277	0,306	0,338
12	0,623	0,670	0,717	13,42	14,94	16,54	1,42	2,94	4,54	0,332	0,364	0,397	0,299	0,331	0,365
13	0,670	0,720	0,770	14,38	16,00	17,72	1,38	3,00	4,72	0,357	0,391	0,427	0,321	0,356	0,392
14	0,716	0,770	0,824	15,33	17,06	18,89	1,33	3,06	4,89	0,382	0,419	0,456	0,344	0,381	0,419
15	0,763	0,820	0,877	16,29	18,13	20,07	1,29	3,13	5,07	0,407	0,446	0,486	0,366	0,405	0,446
16	0,809	0,870	0,931	17,24	19,19	21,25	1,24	3,19	5,25	0,432	0,473	0,515	0,389	0,430	0,473
17	0,856	0,920	0,984	18,20	20,25	22,42	1,20	3,25	5,42	0,457	0,500	0,544	0,411	0,455	0,500
18	0,902	0,970	1,038	19,15	21,31	23,60	1,15	3,31	5,60	0,482	0,527	0,574	0,434	0,479	0,527
19	0,949	1,020	1,091	20,11	22,38	24,77	1,11	3,38	5,77	0,507	0,555	0,603	0,456	0,504	0,554
20	0,995	1,070	1,145	21,06	23,44	25,95	1,06	3,44	5,95	0,532	0,582	0,633	0,479	0,529	0,581

As seen in Figure 111. Worst case SEPIC output voltage spread considering all tolerances (example 1), the resistor values are calculated so that the worst case minimum SEPIC output voltage is ~1.4 V above V_{OUT} (OLED), to guarantee the proper current regulation. A typical SEPIC voltage is then about 2.5 V/3 V above the V_{OUT}.

5.2.1.2 *Experimental circuit verification with SEPIC converter evaluation board*

Figure 112. Circuit schematic with SEPIC topology

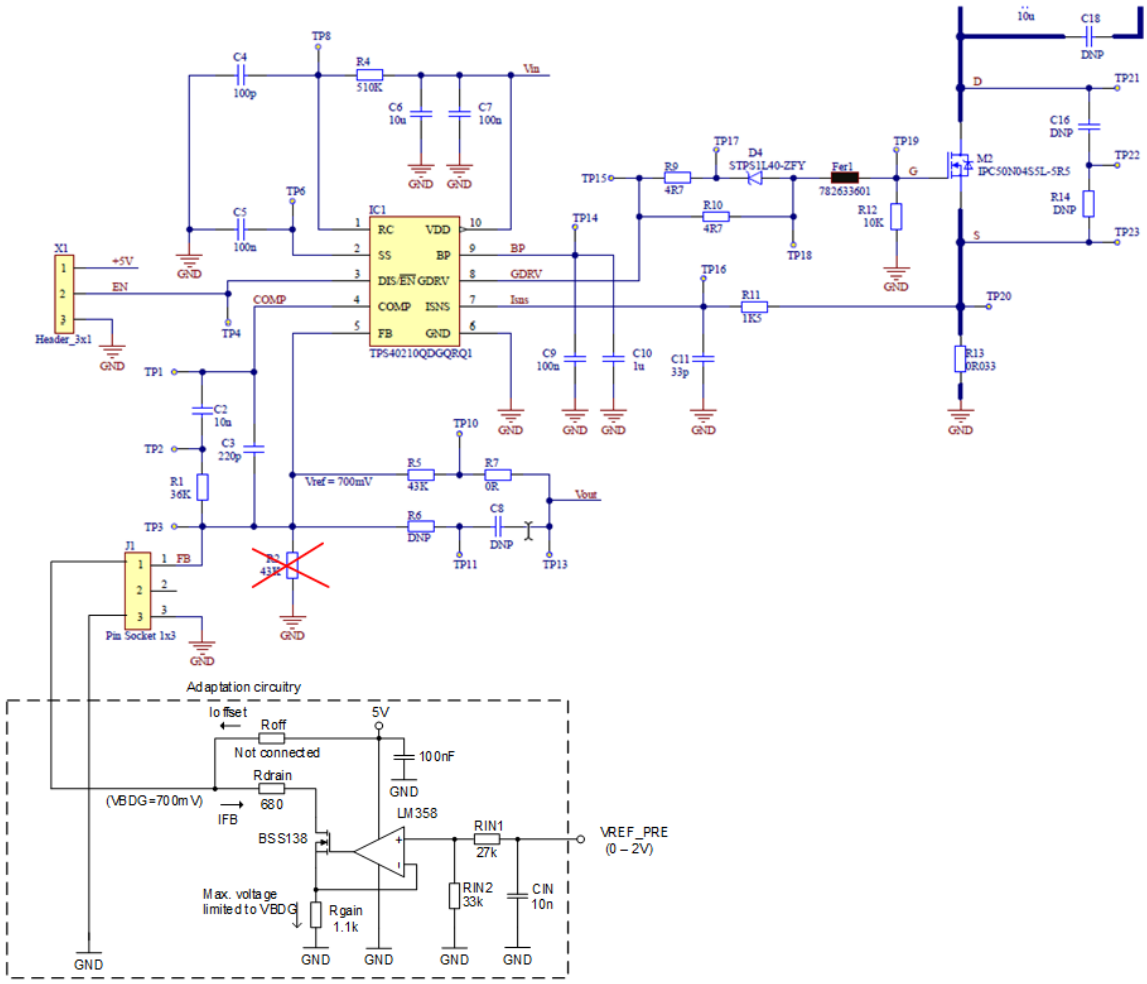
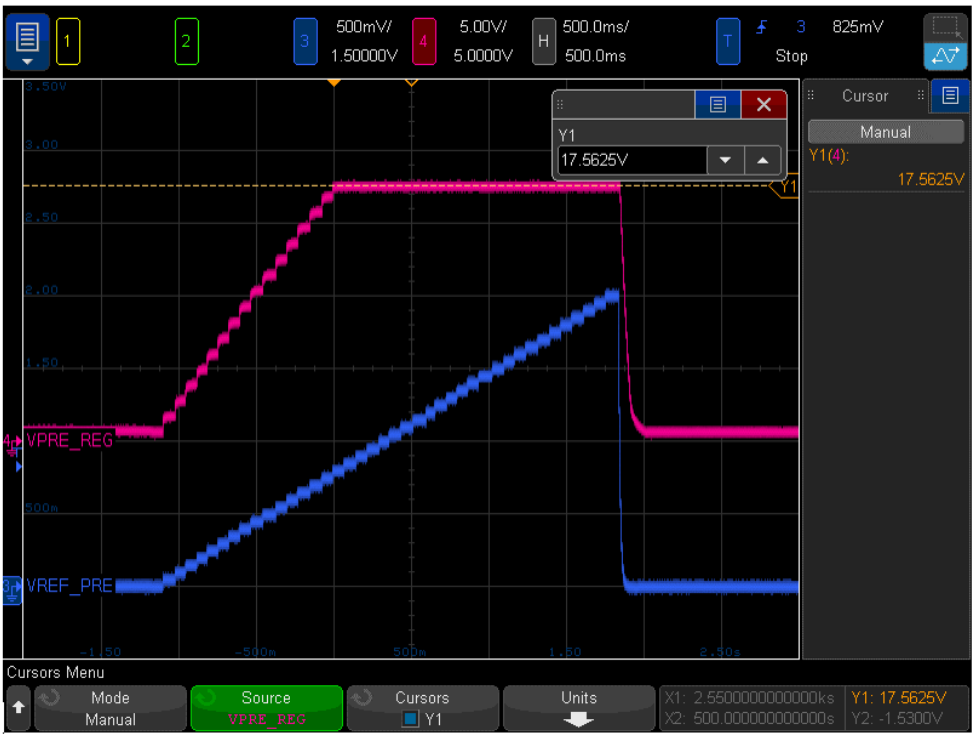


Table 17. V_{PRE_REG} vs VREF_PRE_REG measurement

VREF_PRE_REG [V]	Measured VPRE_REG [V]	Calculated VPRE_REG [V]
0.25	6.10	6.08
0.30	7.18	7.15
0.35	8.28	8.23
0.40	9.36	9.30
0.45	10.44	10.38
0.50	11.52	11.45
0.55	12.60	12.53
0.60	13.68	13.60
0.65	14.78	14.68
0.70	15.86	15.75
0.75	16.94	16.83
0.80	17.60	17.61

VREF_PRE_REG [V]	Measured VPRE_REG [V]	Calculated VPRE_REG [V]
0.85	17.60	17.61
0.90	17.60	17.61
0.95	17.60	17.61
1.00	17.60	17.61
1.05	17.60	17.61
1.10	17.60	17.61
1.15	17.60	17.61
1.20	17.60	17.61
1.25	17.60	17.61
1.30	17.60	17.61
1.35	17.60	17.61
1.40	17.60	17.61
1.45	17.60	17.61
1.50	17.60	17.61
1.55	17.60	17.61
1.60	17.60	17.61
1.65	17.60	17.61
1.70	17.60	17.61
1.75	17.60	17.61
1.80	17.60	17.61
1.85	17.60	17.61
1.90	17.60	17.61
1.95	17.60	17.61
2.00	17.60	17.61

Figure 113. V_{PRE_REG} vs VREF_PRE_REG measurements

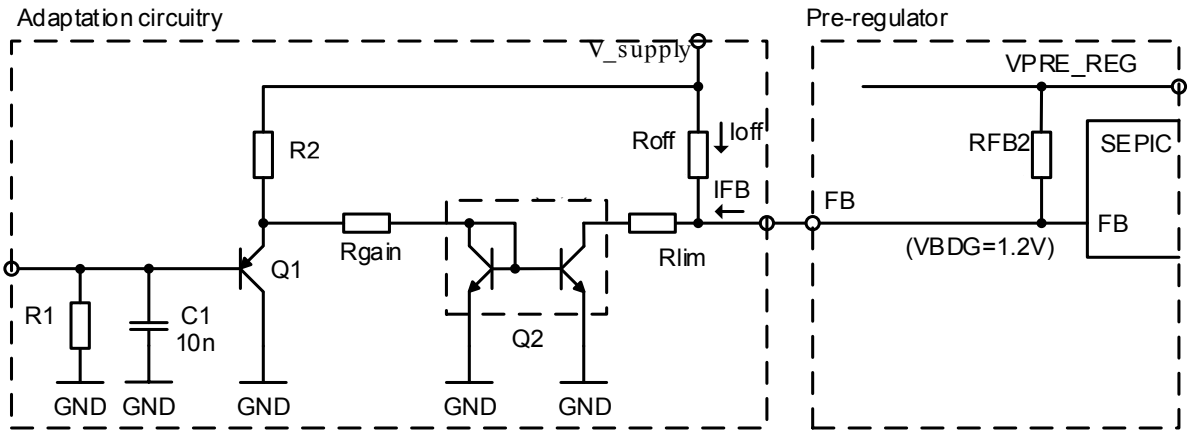


The measured SEPIC output voltage for a given VREF_PRE_REG input voltage perfectly corresponds to typical calculated values.

5.2.2 Circuit schematics for closed loop regulation algorithm

5.2.2.1 Low-cost solution with BJT devices

Figure 114. Circuit schematics connecting $V_{REF\_PRE\_REG}$ signal(s) to feedback pin (with BJT)



A low cost solution avoiding Op Amps has been targeted. The adaptation circuitry works as voltage → current converter. The VREF_PRE_REG input voltage is translated to current sunk from the upper resistor of the feedback divider RFB2 (bottom resistor is not present). The RFB2 value should not be selected too high in order to operate at higher current levels and so to minimize sensitivity to noise and leakages. An additional offset current is added by the Roff resistor in order to compensate the VFB voltage shift of the FB node (so to align the 0V VREF_PRE_REG input to the 0V VPRE_REG output).

The voltage → current conversion ratio is defined by the Rgain resistor. The Q1 transistor (emitter follower) acts as a level shifter shifting the VREF_PRE_REG voltage by one junction drop higher to be inside the operating range of the NPN current mirror and so to work in the whole VREF_PRE_REG voltage range. The current mirror is realized using a dual NPN transistor in one package which minimizes temperature difference between them. Even if the exact parameter matching is not guaranteed for this low cost type, it is still better than using individual transistors. The Rlim resistor in series with current mirror output is used for adjusting the maximum VPRE_REG voltage. The output current of the mirror is limited when the voltage drop across the RLIM reaches VBDG.

The resistors are to be dimensioned as follows:

Equation 7

$$R_{OFF} = \frac{R_{FB2} \cdot \left(V_{Supply} - V_{FB}\right)}{V_{FB}} \quad (7)$$

Equation 8

$$R_{lim} = \frac{\left(V_{FB} - V_{CE_{SAT}}(Q2)\right)}{\frac{V_{PRE\_REG_{MAX}} - V_{FB}}{R_{FB2}} + \frac{V_{supply} - V_{FB}}{R_{off}}} \quad (8)$$

V_CE_SAT is the saturation collector emitter drop voltage across the NPN current mirror. To ensure Q1 is working properly as a level shifter a certain minimum collector current I_CE_Q1_min is needed. The collector current flowing through Q1 is lowest when VREF_PRE_REG is set to its maximum, equal to 2 V.

Equation 9

$$\frac{\left(V_{supply} - 2V - V_{BE}(Q1)\right)}{R_2} - \frac{(2V + V_{BE}(Q1) - V_{BE}(Q2))}{R_{gain}} > I_{CE}min(Q1) \quad (9)$$

With

Equation 10

$$R_{gain} \cong K \cdot R_{FB2} = 0.05 \cdot R_{FB2} \quad (10)$$

The pull-down resistor R1 on the base of Q1 is needed to sink the base current of Q1 in all conditions while keeping the drop voltage below the VREF_PRE_REG voltage. This pull-down resistor is in parallel with the biasing resistor in the VREF_PRE_REG output stage (see Figure 101. Wired-OR connection between two or more (O)LED drivers). The maximum base current is expected when VREF_PRE_REG is at minimum voltage. It is to be noted that, the VREF_PRE_REG while the closed loop regulation is active, could drop even below 0.25 V. With 0.1 V as the bare minimum operating voltage for VREF_PRE_REG during closed loop operation, the dimension criteria for R1 becomes:

Equation 11

$$\frac{\left(V_{supply} - V_{BE}(Q1) - 0{,}1V\right)}{h_{FE}(Q1)^{*}R_2} \cdot \frac{R1 \cdot R_{bias}}{(R1 + R_{bias})} < 0{,}1V \quad (11)$$

R_{bias} is typically about 80 kΩ. From the above formula we can conclude as important design criteria to choose a pnp transistor with high gain h_{FE}.

On the other hand, as explained in the previous chapter, the VREF_PRE_REG voltage has an internal serial impedance of about 700 Ω, therefore another condition applies:

Equation 12

$$R1 \gg 700\Omega \quad (12)$$

in order to ensure a small error of VREF_PRE_REG at the base of Q1. A good value for R1 is in the range of 10/22 kΩ.

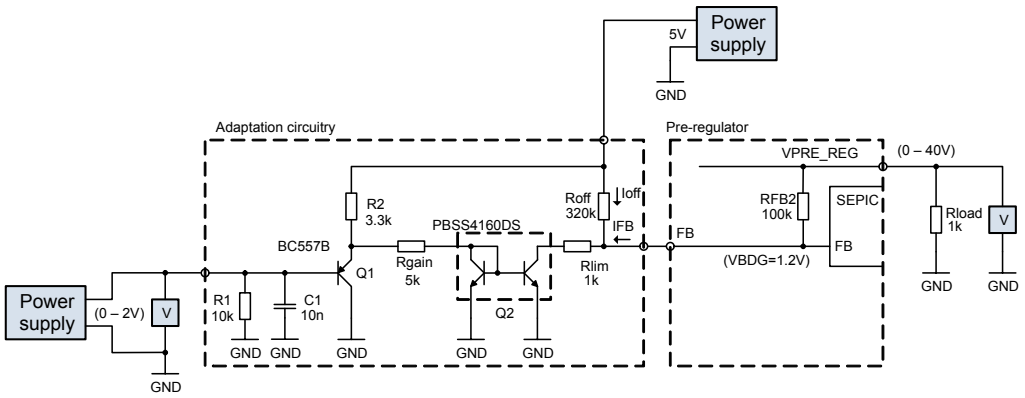
Having defined those values, the VPRE_REG voltage is calculated as

Equation 13

$$V_{PRE\_REG} = V_{REF\_PRE\_REG} \cdot \frac{R_{FB2}}{R_{gain}} + V_{FB} - I_{offset} \cdot R_{FB2} \quad (13)$$

5.2.2.1.1 Application example (2)

Figure 115. Circuit schematic connecting $V_{REF\_PRE\_REG}$ signal(s) to feedback pin



5.2.2.1.2 Experimental circuit verification with SEPIC converter evaluation board

This circuit was used to verify the linearity and possible offset between the VREF_PRE_REG input voltage and the VPRE_REG output voltage. External resistors were calculated according to the Equation 7, Equation 8 and Equation 10 Rlim was dimensioned to limit the maximum VPRE_REG at 20 V. Table 18. V_{PRE_REG} vs $V_{REF\_PRE\_REG}$ measurement (application example 2) reports the difference between the ideal Vout calculated per Equation 13 and Figure 116. VPRE_REG vs VREF_PRE_REG illustrates the results.

Table 18. V_{PRE_REG} vs $V_{REF\_PRE\_REG}$ measurement (application example 2)

VREF_PRE_REG [V]	VPRE_REG [V]	VPRE_REG_IDEAL [V]	Error [V]
0	3.3	0.0	3.3
0.1	4.9	2.0	2.9
0.2	6.6	4.0	2.6
0.3	8.3	6.0	2.3
0.4	10.1	8.0	2.1
0.5	11.8	10.0	1.8
0.6	13.6	12.0	1.6
0.7	15.4	14.0	1.4
0.8	17.1	16.0	1.1
0.9	18.8	18.0	0.8
1	20.3	20.0	0.3
1.1	21.0	20.5	0.5

VREF_PRE_REG [V]	VPRE_REG [V]	VPRE_REG_IDEAL [V]	Error [V]
1.2	21.3	21.0	0.3
1.3	21.5	21.0	0.5
1.4	21.6	21.0	0.6
1.5	21.7	21.0	0.7
1.6	21.8	21.0	0.8
1.7	21.8	21.0	0.8
1.8	21.9	21.0	0.9
1.9	21.9	21.0	0.9
2	22.0	21.0	1.0

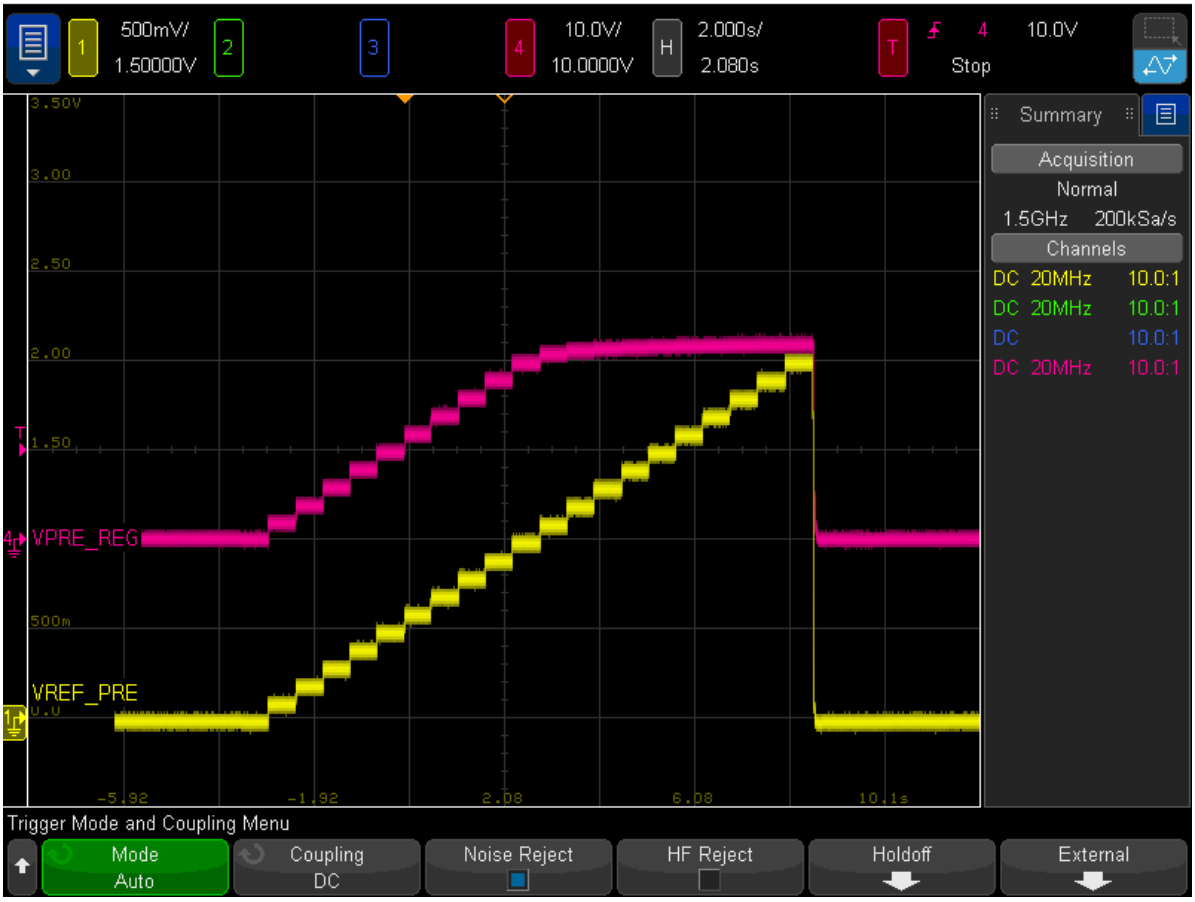
Figure 116. **VPRE_REG vs VREF_PRE_REG**

There is a certain offset and some nonlinearity vs the ideal characteristics. Besides the known tolerances of the external resistors, the feedback voltage, the supply voltage of the circuit, "hfe" matching error of "NPN" current mirror, the major part of the errors comes from the V_{BE} mismatch between Q1 and Q2. Therefore Q1 and Q2 should be selected so to have similar or even identical V_{BE} considering the range of collector operating current. The difference at a certain operating point can be compensated by adapting the external component dimension following the formulas from Equation 7, Equation 8 and Equation 10 as shown in the following table and figure where Rgain has been reduced to 4.5 kΩ and Roff to 76 kΩ. Since the V_{BE} has a non-negligible temperature dependency of ~ 2 mV/K, it is important to position Q1 and Q2 in close proximity on the PCB to ensure identical operating temperature.

Table 19. V_{PRE_REG} vs $V_{REF\_PRE\_REG}$ measurement trimmed circuit

VREF_PRE_REG [V]	VPRE_REG [V]	VPRE_REG_IDEAL [V]	Error [V]
0	0.2	0.0	0.2
0.1	2.0	2.0	0.0
0.2	3.9	4.0	-0.1
0.3	5.9	6.0	-0.1
0.4	7.9	8.0	-0.1
0.5	9.9	10.0	-0.1
0.6	11.9	12.0	-0.1
0.7	14.0	14.0	0.0
0.8	16.0	16.0	0.0
0.9	18.0	18.0	0.0
1	19.9	20.0	-0.1
1.1	20.9	21.0	-0.1
1.2	21.3	21.0	0.3
1.3	21.5	21.0	0.5
1.4	21.6	21.0	0.6
1.5	21.7	21.0	0.7
1.6	21.8	21.0	0.8
1.7	21.8	21.0	0.8
1.8	21.9	21.0	0.9
1.9	21.9	21.0	0.9
2	21.9	21.0	0.9

Figure 117. **VPRE_REG vs VREF_PRE_REG trimmed circuit**



The trimming of the circuit certainly can be applied for experimental trials only, not for volume production. Therefore, the dimensioning of the external components has to be done so that the circuit will work properly in the whole temperature range and across all component tolerances. The boundary conditions to be met are:

- The circuit must guarantee a positive error, this means considering all tolerances, the generated VPRE_REG can only be higher, never lower than the targeted value. This is important to avoid dropouts of the regulated output voltage at channel turn on/off which could lead to a short visible brightness fluctuation.
- The maximum positive error must be limited in order to guarantee at the lowest regulated output voltage a regulated VREF_PRE_REG which stays always above the voltage drop across R1, Q1 base resistor

A calculator tool is available to facilitate the component dimensioning task considering worst case tolerances.

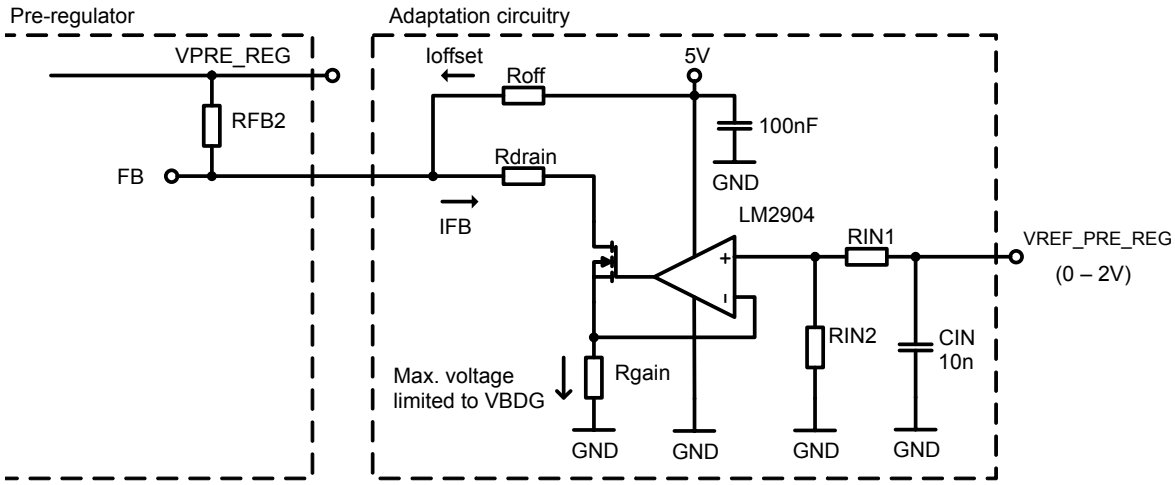
5.2.2.1.3 Further design and layout recommendations

- The NPN current mirror circuitry must be as close as possible to pre-regulator, to minimize the feedback pin connection length, as it is the most sensitive point of the regulator. This signal should not be filtered otherwise it degrades the regulation loop performance
- The whole circuitry should be close to the SEPIC (distance between PNP and NPN current mirror should be minimized)
- The RFB2 resistor value should be maximum 100 kΩ. With higher value the current mirror will operate at very low current levels so it becomes more sensitive to noise and eventual leakage currents
- The 10 nF on PNP input is mandatory. Without this capacitor the VREF_PRE_REG signal is quite noisy. In case of longer distance the VREF_PRE_REG connection (>10 cm), it is possible place close to the device VREF_PRE_REG output, additional 10 nF capacitor.

5.2.2.2 *Circuit schematics with Op Amp*

Of course, the same circuit as proposed for the open loop control regulation can be used also when closed loop regulation is selected. This circuit has the advantage of lower tolerances and therefore higher precision (see Section 5.2.1). The component dimensioning requirements are less critical than for the BJT circuit, but still important.

Figure 118. Circuit schematics connecting $V_{REF\_PRE\_REG}$ signal to feedback pin



The same boundary conditions to be met are:

- The circuit must guarantee a positive error, this means considering all tolerances, the generated VPRE_REG can only be higher, never lower than the targeted value. This is important to avoid dropouts of the regulated output voltage at channel turn on/off which could lead to a short visible brightness fluctuation.
- The maximum positive error must be limited to guarantee at the lowest regulated output voltage a regulated VREF_PRE_REG which stays always above 100 mV

5.2.2.2.1 Application example (3)

Figure 119. Calculation of resistor dimensioning for application example 3

VOFFSUPPLY	5	V	Offset resistor supply voltage (for Roff calculation)	
VBDG	1,2	V	Reference voltage of external preregulator	
VPRE_REG max.	20	V	Maximum output voltage of the pre-regulator (for Rdrain calculation)	
K	0,05	-	K factor (0.1 for L99LDLL16, 0.05 for L99LDLH32)	
RFB2	43	kohm	Upper resistor of feedback divider	
RIN1	27	kohm	Upper resistor of control voltage divider	
RIN2	33	kohm	Lower resistor of control voltage divider	
Vref_pre impedance	0,7	kohm	Internal resistance of Vref_pre output	
RIN_ratio	0,543657331	-	Control voltage divider ratio (out/in)	
Rgain	1,168863262	kohm	Current sink shunt resistor	
Roff	136,1666667	kohm	Offset current definition resistor	
Rdrain	1,411136738	kohm	Maximum VPRE_REG voltage definition resistor	

As it can be seen from the above calculation example, the offset resistor calculation results in a quite high number. As it often happens in automotive design, resistors higher than 100 kΩ are not tolerated in circuit schematics. Therefore the offset resistor can be even skipped producing a small error.

Without this resistor, the SEPIC output voltage will be shifted by V_{FB} higher. A positive shift is anyway required, as previously discussed. All these calculations are available in a separate calculator:

Figure 120. Tolerances for worst case calculation (example 3)

Vref pre reg Impedance	40	% tolerance
RFB2	1	% tolerance
Bandgap	1	% tolerance
Voff_supply	3	% tolerance
Roff	1	% tolerance
Rgain	1	% tolerance
Rdrain	1	% tolerance
RIN1	1	% tolerance
RIN2	1	% tolerance
K	7	% tolerance
OpAmp offset	2	mV

Figure 121. Worst case SEPIC output voltage spread considering all tolerances (example 3)

Vout (OLED)	4	V															
	min		typ		max												
Vref_pre (floating)	0,252	V	0,271	V	0,289	V		Vref_pre (floating)=(Vout+1.41)*k									
V_Rgain	0,133	V	0,147	V	0,162	V		V_Rgain=Vref_pre*RIN2/(Vref_resistance+RIN1+RIN2)									
IFB	0,120	mA	0,134	mA	0,148	mA		IFB=V_Rgain/Rgain									
Ioffset	0,000	mA	0,000	mA	0,000	mA		Ioffset=(Voff_supply-Vbdg)/Roff									
Vpre_reg	6,28	V	6,95	V	7,65	V		Vpre_reg=(IFB-Ioffset)*RFB2+VBDG									
Vpre_reg_max	20,45	V	21,05	V	21,66	V		Vpre_reg_max=(VBDG/(Rgain+Rdrain)-(VOFFSUPPLY-VBDG)/Roff)*RFB_2+VBDG									
	Vref_pre (floating) [V]			Vpre_reg [V]				Vpre_reg - Vout [V]			V_Rgain [V]			IFB [mA]		
Vout (OLED) [V]	min	typ	max	min	typ	max		min	typ	max	min	typ	max	min	typ	max	
4	0,252	0,271	0,289	6,28	6,95	7,65		2,28	2,95	3,65	0,132905	0,147059	0,161505	0,11963	0,13369	0,14830	
5	0,298	0,321	0,343	7,24	8,01	8,83		2,24	3,01	3,83	0,157845	0,174424	0,190988	0,142075	0,158401	0,17533	
6	0,345	0,371	0,396	8,19	9,07	10,00		2,19	3,07	4,00	0,182781	0,201425	0,220472	0,164521	0,183114	0,20245	
7	0,391	0,421	0,450	9,15	10,14	11,18		2,15	3,14	4,18	0,207715	0,228608	0,249953	0,186966	0,207823	0,22952	
8	0,438	0,471	0,503	10,10	11,20	12,36		2,10	3,20	4,36	0,232654	0,255791	0,279439	0,209411	0,232537	0,25660	
9	0,484	0,521	0,557	11,06	12,26	13,53		2,06	3,26	4,53	0,257593	0,282974	0,308922	0,231857	0,257249	0,28367	
10	0,531	0,571	0,610	12,01	13,32	14,71		2,01	3,32	4,71	0,28253	0,310157	0,338405	0,254302	0,28196	0,31074	
11	0,577	0,621	0,664	12,97	14,39	15,88		1,97	3,39	4,88	0,307467	0,337339	0,367885	0,276748	0,306672	0,33782	
12	0,624	0,671	0,717	13,92	15,45	17,06		1,92	3,45	5,06	0,332404	0,364521	0,397374	0,299193	0,331384	0,36489	
13	0,670	0,721	0,771	14,88	16,51	18,24		1,88	3,51	5,24	0,357341	0,391705	0,426856	0,321639	0,356096	0,3919	
14	0,717	0,771	0,824	15,84	17,57	19,41		1,84	3,57	5,41	0,382277	0,418888	0,456339	0,344084	0,380807	0,41904	
15	0,763	0,821	0,878	16,79	18,64	20,59		1,79	3,64	5,59	0,407214	0,446071	0,485824	0,366529	0,405519	0,44611	
16	0,810	0,871	0,931	17,75	19,70	21,76		1,75	3,70	5,76	0,432151	0,473254	0,515306	0,388975	0,430231	0,47319	
17	0,856	0,921	0,985	18,70	20,76	22,94		1,70	3,76	5,94	0,457088	0,500437	0,544789	0,41142	0,454942	0,50026	
18	0,903	0,971	1,038	19,66	21,83	24,11		1,66	3,83	6,11	0,482025	0,527619	0,574272	0,433866	0,479654	0,52733	
19	0,949	1,021	1,092	20,61	22,89	25,29		1,61	3,89	6,29	0,506962	0,554802	0,603756	0,456311	0,504366	0,55441	
20	0,996	1,071	1,145	21,57	23,95	26,47		1,57	3,95	6,47	0,531899	0,581985	0,633239	0,478757	0,529077	0,58148	
lowest Vref	0,183	V															

As seen in Figure 121, the resistor values are calculated so that the worst case minimum SEPIC output voltage is always > 1.4 V above V_{OUT} (OLED), to fulfill the first boundary design condition. The lowest possible VREF_PRE_REG voltage was calculated and confirmed to match with the requirements of the second boundary design condition. The dimensioning of the circuit can therefore be considered robust against all tolerances and temperature variations.

6 Premises on devices NVM management

Both L99LDLH32 and L99LDLL16 are equipped with an NVM allowing to program some parameters that are relevant for the devices' functionality. The memory layout along with device parameters is outlined below.

Figure 122. NVM chunk 0 (columns from 0 to 7)

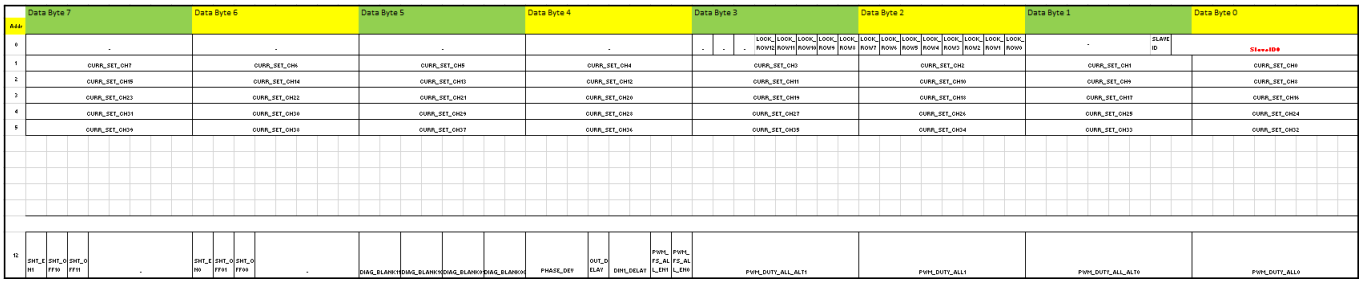
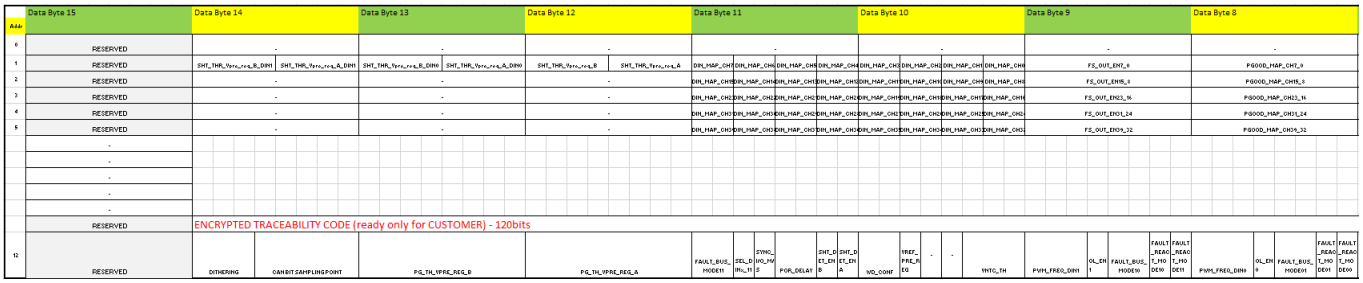


Figure 123. NVM chunk 1 (columns from 8 to 15)



In order to provide some context for a first-time reading on the subject, device parameters in NVM are influential in the following areas of the device functionality:

- Device ResponderID
- Per channel current intensity
- Per channel DIN_MAP group
- Per channel FS_OUT_ENABLE
- Settings for detection of open load, short, short to gnd, pwr good and NTC
- PWM parameters: operating frequency and duty cycle for Fail Safe mode operation
- Watchdog configuration
- Parameters for Commander/Responder operation
- Output gradual delay and dithering
- LOCK bits to write protect NVM partially or entirely

More specific details related to each single parameter can be found in the L99LDLH32 and L99LDLL16 devices datasheets.

In NVM programming mode it is possible to access the NVM in read and write on a limited set of addresses, each address (as outlined in Figure 122. NVM chunk 0 (columns from 0 to 7) and Figure 123. NVM chunk 1 (columns from 8 to 15), the first column on the left side) is related to a specific row of parameters. The scope for customer NVM access on each device (L99LDLH32, L99LDLL16) is specified below:

L99LDLL16 device read

- The customer can read rows: **0, 1, 2, 11 and 12 with response for user** (it can take up to 450 us per row)
- Reading any other different row is forbidden, otherwise the FTP enters in a lock state and further commands on FTP cannot be executed until next transition though initialization state

L99LDLL16 device write

- If the relative lock bit is 0, the customer can write rows: **0, 1, 2 and 12 with no response about when write operation has been completed** (just an ack for CAN system). Each write operation lasts about 12 ms per row
- Writing any other different row is forbidden. Otherwise the FTP enters in a lock state and further commands on FTP cannot be executed until next transition through Initialization state

L99LDLH32 device read

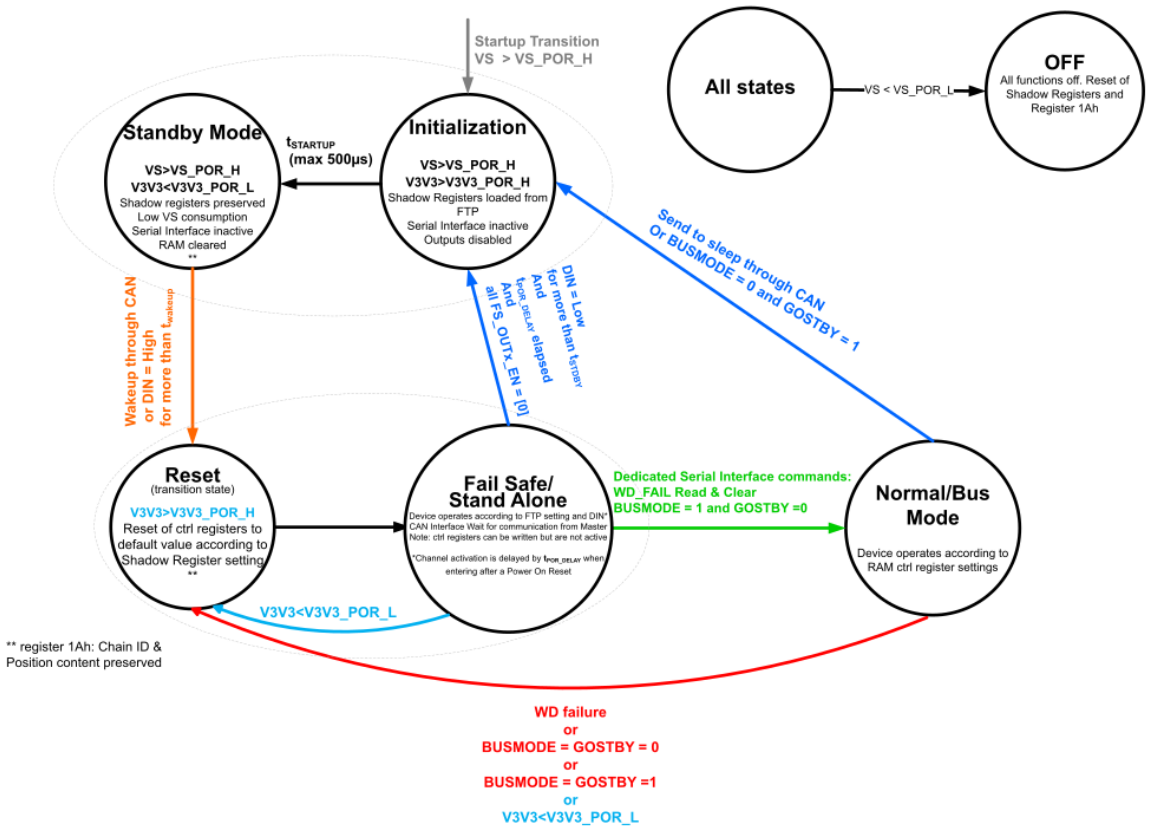
- The following rows can be read: **0, 1, 2, 3, 4, 11 and 12 with response for user** (it can take up to 450 us per row)
- Trying to read other different locations, the system does anything, and **no response is provided**

L99LDLH32 device write

- If the relative lock bit is 0, the customer can write rows: **0, 1, 2, 3, 4 and 12 with no response for user** (just an ack for CAN system)
- The total time required for a complete write cycle is equal to 9.046 ms. Writing must not be interrupted with any new frame while the actual writing process is still in execution
- After trying to write to other different locations or when the related lock bit is set to 1, the system will not take any action and **no response is provided**

Whether not explicitly highlighted in the sections that follow, for NVM programming actions to be effective the device needs to be either in Fail Safe mode or Bus mode. These two states are highlighted in the picture related to the devices state machine below:

Figure 124. Devices state machine



For NVM programming purposes it is recommendable to perform a transition to Fail Safe mode which is relatively straightforward by raising DIN pin level. Permanence in that state only needs DIN level to be kept high. No additional messaging effort through watchdog frames needed as in Bus mode.

6.1 Recommendations before starting first NVM programming

Before following up with a step by step procedure to write generic information within NVM rows, we start by providing some advice in order to prevent a first write attempt to fall into NVM areas which might unintentionally compromise the device use once forever.

NVM contains 12 LOCK bits all located in row0 of the NVM (row 0, data byte 2 and 3). Each one meant to write-protect the memory content of an NVM row in the range from 0 to 12 in a way that can be permanent. Therefore need to be used carefully.

It is possible to program all the NVM rows from ROW1 onwards (ROW1 included) and change their content. The corresponding lock bit located on ROW0 is set and a new NVM download occurs.

Once the system is restarted, if a given row (>0) is locked, no change can be done on it, but it is still possible to restore its content by reprogramming the corresponding lock bit on the ROW0.

LOCK_ROW0 bit must not be set to 1 before the NVM programming process conclusion. Otherwise, after the next NVM download, it will not be possible to modify any bits in ROW0 anymore, including those LOCK bits that could have already been set to lock other rows. In such a case, the NVM would be partially or entirely read-only from that point.

If the intention is to lock only the ResponderID it is possible by setting the bit LOCK_RESPONDER ID in ROW0 to 1. In such way, after the next NVM download ResponderID and LOCK_RESPONDER ID fields will not be writable anymore.

In FTP programming / reading mode, the following commands are forbidden:

1. Burst read
2. FTP command accessing nonexisting rows in customer mode (valid for L99LDLL16)
    a. L99LDLL16 customer mode: Any address different from 0, 1, 2, 11 and 12

In case such command is accidentally issued the device enters in a lock state and subsequent commands on NVM cannot be executed. If accessed through CAN, in lock state the device provides no answer, except the acknowledge of the CAN frame. To leave the lock state it is necessary to send the device in standby mode (execute a “go-to-sleep” command), and wake up the device after a minimum delay time of 0.5 ms.

Subsequently, the standard procedure to enter NVM programming mode (outlined in the next section) must be reapplied.

6.2 NVM single device programming using CAN FD interface

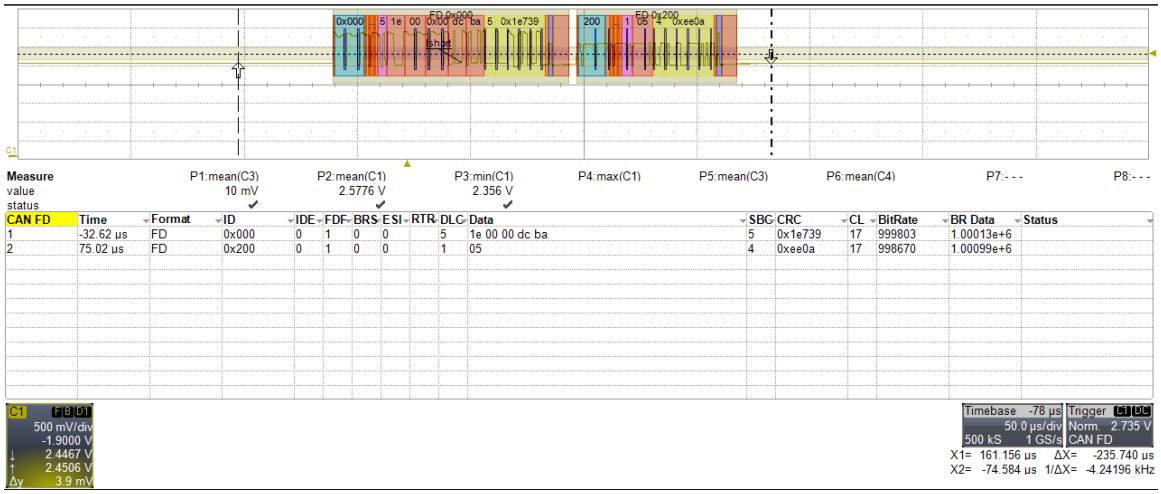
Step by step procedure to write/read generic NVM row:

1. **ENABLE**: set physical pin CS (to 5V) or SW bit CS_EN (in control register 24)

2. **NVM KEY WRITE:** write customer key (hex 0000DCBA) to address (hex 1E) through CAN using a unicast frame for single write (5 bytes) (hex 1E0000DCBA). This address is valid only for operations on FTP using CAN communication, it does not exist as physical address in RAM device

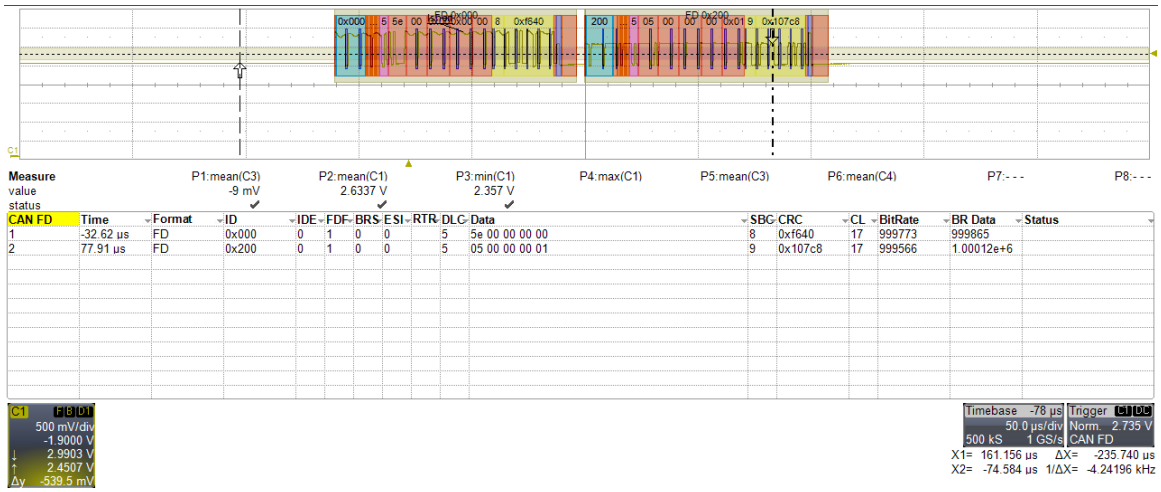
   In order to be sure that the previous operation was successful which means NVM access is granted, a new unicast read frame at the same address 1E is required (see next step).

Figure 125. Customer key write transaction captured by CAN FD decoder on oscilloscope (device in Fail Safe mode, ResponderID=0)



3. **NVM KEY STATUS:** In case the previous operation was successfully completed and the device grants NVM access, a status bit is set and can be read using a unicast read frame (5 bytes) at address 1E (hex 5E,00,00,00,00). Status bit to check for operation outcome is the last bit of the device reply to the frame, the status bit has to be 1 for confirmation of successful outcome

Figure 126. NVM key status



Now that the key has unlocked NVM, it can be accessed for write or reads using a predefined command format on CAN FD unicast frames payload data consisting of 16 bytes.

Commands are encoded within the payload of the CAN FD unicast frames following the syntax described bit by bit below starting from the command code + address byte:

[1, 0, 1, (W/R), a3, a2, a1,a0] + [15 bytes of NVM data related to the addressed row]

(a3, a2,a1,a0) are bits related to the FTP row address, (W/R)=1 for write and 0 for read and “+” just represents an operator for concatenation of bit sequences. Clearly the 15 bytes of data following opcode and address are meaningful in case of write operation otherwise a padding consisting in dummy “00h” bytes can be used.

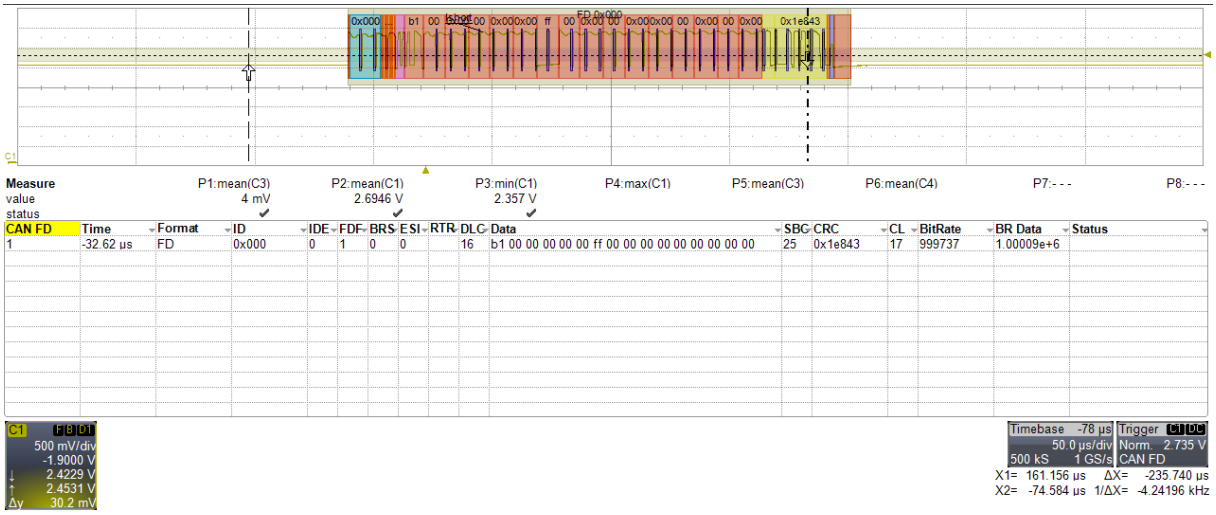
Write and Read transactions examples is described in the next two steps which unlike the previous ones come with no chronology constraint (for example, the user might read and/or write, only one or both depending on specific needs).

4. **WRITE:** Following the syntax outlined in the previous point, in order to set permanent ON in Fail Safe (FS_OUT_EN) for outputs ranging from 0 to 7, we know from the NVM map (refer to Figure 123. NVM chunk 1 (columns from 8 to 15)) that we need to set to FFh byte at row1, column(byte) 9 of NVM therefore the command will be encoded into the unicast payload frame as follows (here expressed in hexadecimal):

   [B10000000000FF000000000000000000]h

   (Write=1, Address=0001 coded within the first byte)

Figure 127. CAN FD frame setting FS_OUT_EN for all channels from 0 to 7

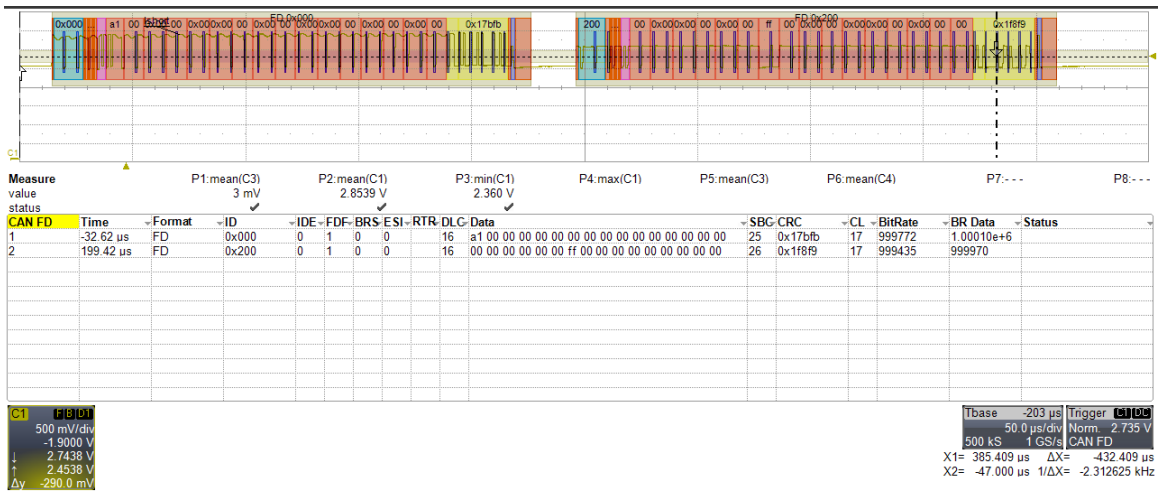


5. **READ**: Following the syntax outlined in point 3, as example of read back of FTP row 1 (written at point 4) through CAN FD frame, the content of the payload would be:

   [A10000000000FF000000000000000000]h

   (Write=0, Address=0001 coded within the first byte)

Figure 128. Readback of NVM row 1, previously written at point 4



6. **RELOAD NVM**: after write operations in NVM the new values are not immediately loaded into RAM. A transition through the initialization state is necessary, since only in this state the device reads the NVM memory and downloads its content into the RAM registers (as outlined in Figure 124). To that end, either power cycling the device or a transition to standby would allow NVM reload of the new content

6.2.1 Observations on Responder ID write

ResponderID NVM parameter is located at row0, bytes 0, 1 (check the Figure 122. NVM chunk 0 (columns from 0 to 7)) and can be written following the same general rules previously outlined. Specifically, supposing need to write ResponderID=04h here is what the payload of the command unicast frame would be:

[B00000000000000000000000000000004]h,

since (Write=1, Address=0000) for the first two bytes on the left-hand side.

In general, aside from devices on bench evaluation purposes, it is always recommendable to write-protect the ResponderID for final product evaluation, this can be accomplished through its related LOCK bit located starting from row0, column (byte) 1 of the NVM (refer to Figure 122. NVM chunk 0 (columns from 0 to 7)). As before, adding lock on ResponderID write would change the unicast command frame payload as follows:

[B00000000000000000000000000000024]h

At beginning it is recommended to write only the target ResponderID0 without setting any lock bits, then read the FTP Row0 in order to be sure that its content is correct and then rewrite again the Row0 with the target ResponderID0 and Lock_ResponderID set at 1.

After Row0 writing operations in FTP, to make the new content effective it is necessary to perform a transition through the initialization state (for example, power cycle or goto standby), since only in this state the device reads the FTP memory and download its content into the RAM registers.

6.3 NVM programming with more devices connected on the same CAN FD bus

When more than one device is connected to the same CAN FD bus, actions to identify the correct target device for NVM programming need to be taken. As outlined in the step-by-step procedure for single device programming in section 3, the target device CS pin will have to be risen to 5 V in order to ENABLE access, then all CAN FD Light unicast messages for NVM write/read operation need to be set with the target-device ResponderID, which must be a distinct identifier for each responder device to achieve consistent CAN FD Light bus operation. All other programming steps aside from this premise do not change from what was described in Section 3 for the single device case.

During initial system setup however, when NVM is blank, by default all devices would respond as ResponderIDs 0, therefore, for this case the following steps are recommended:

1. Address each single device in order to program its individual ResponderID first. CS is set high on the target device only, then the steps described in Section 3 include (2) KEY WRITE and (3) KEY STATUS frame will be sent but the answer from the reponder temporarily ignored. At this point, guidelines for ResponderID setting in 3.1 can be applied and NVM reloaded (pwr cycling or goto standby). At the next device transition to Fail Safe or Bus mode (according to state machine in Figure 124. Devices state machine) device is checked to respond according to newly assigned responder ID
2. Once all devices heading the same bus are programmed with a distinct responder ID, follow the guidelines provided at the heading of this section (referring to an initial bus state with distinct identifiers for distinct devices)

6.4 FTP Programming tool - description

The tool is intended to support a user-friendly programming and reading FTP (NVM) memory. It can handle up to 8 L99LDLH32 devices simultaneously, as there are 8 HW chip select lines available and eight configuration pages. However, the number of devices on the CAN bus is not limited to 8. In the case of a larger number of devices, the user can program several groups of devices step by step, by reconnecting the chip select lines manually or by selecting another group of responder IDs (if already programmed).

The graphical user interface (GUI) allows to read or write FTP data from / to all devices. Actual FTP configuration is visible in human readable form as well as in hexadecimal memory map form. Both forms can be freely edited while changes are propagated between them. The FTP configuration can be saved to or loaded from xml file.

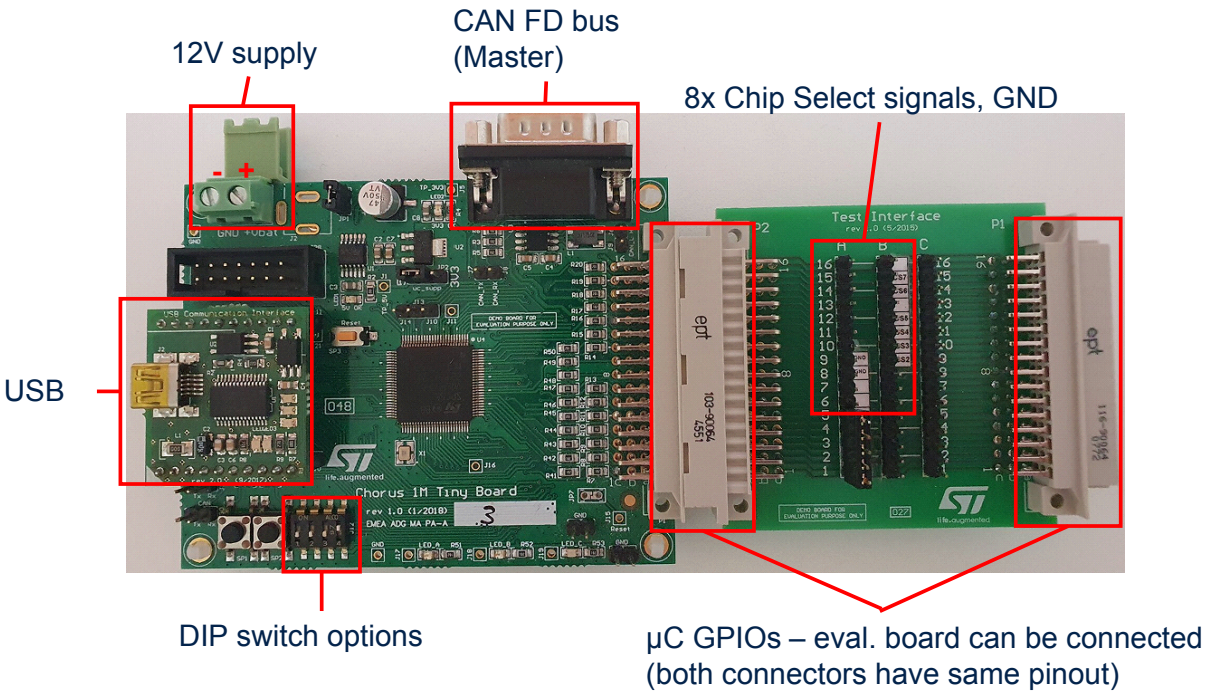
The tool supports fully automatic mode, loading and verifying the FTP configuration in all devices by one click, as well as interactive mode when the user can access each device / FTP row individually. Every FTP write command is automatically followed by FTP read command for verification. The verification status is reported in the “Log window”.

Both, HW and SW chip select modes are supported. In the case of HW chip select mode, it is possible to automatically scan connected responder IDs by checking whole address range for each chip select signal. This is useful, for example, in the case of connecting already programmed devices with unknown responder IDs.

6.4.1 Programming tool - hardware reference setup

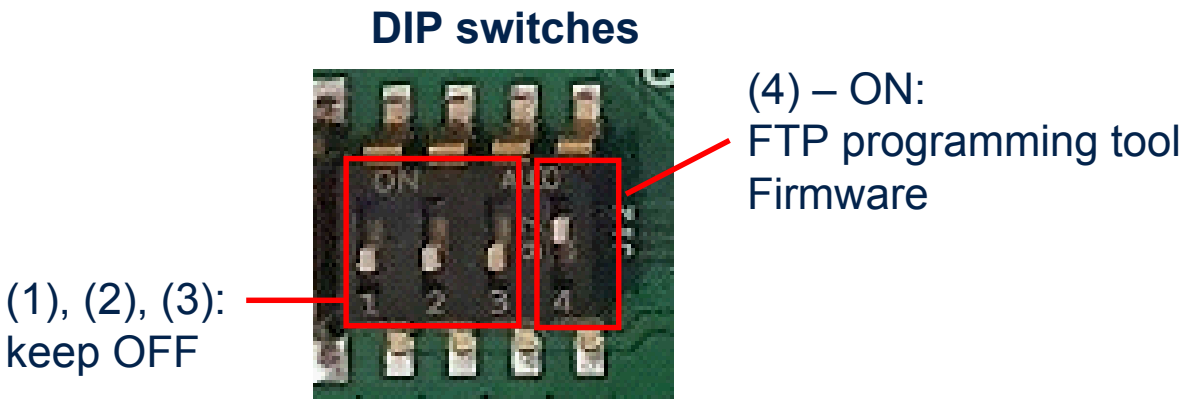
An overview of the hardware reference for the programming mode is reported in Figure 129.

Figure 129. HW setup - programming modes



The FTP programming tool firmware, can be properly selected through the below DIP switches:

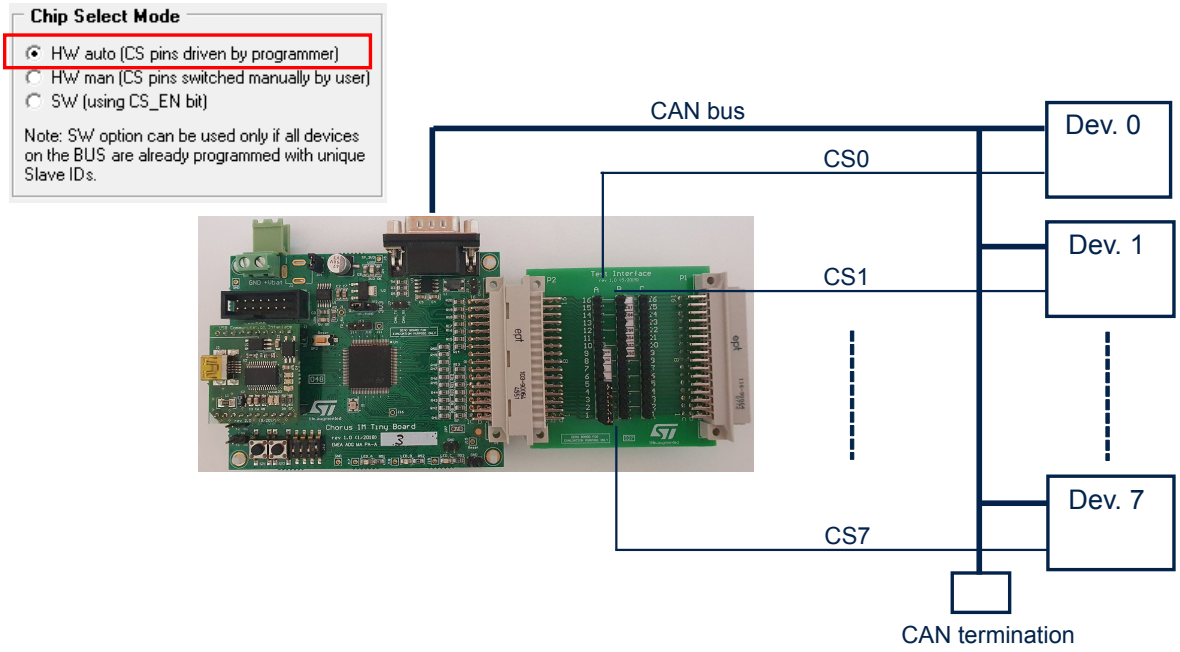
Figure 130. DIP switches



When FTP tool firmware is selected, 3 different programming modes of the chip select can be chosen:

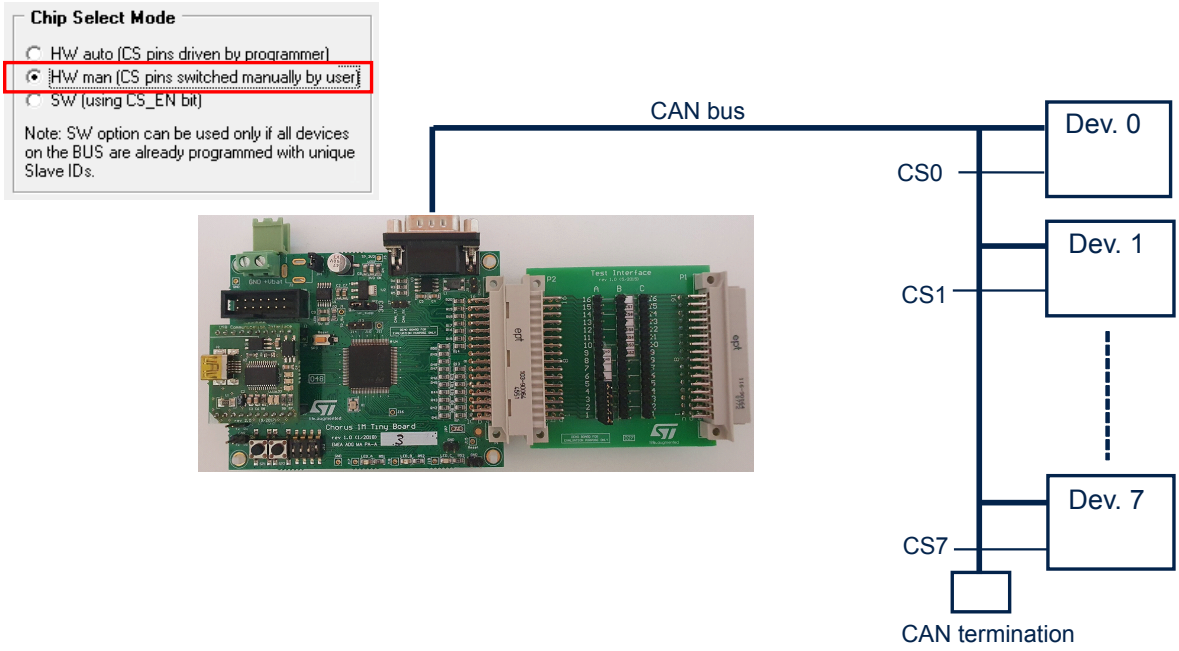
1. HW auto - chip select pins driven by the programmer
2. HW man - chip select pins manually switched by user
3. SW - playing with chip select enable bit
4. **Hardware automatic programming (HW auto):** when selected, it allows an automatic programming and verification of all connected devices (either factory default or already programmed) as well as automatic detection of all connected ResponderID addresses.

Figure 131. **Automatic (HW) programming mode**



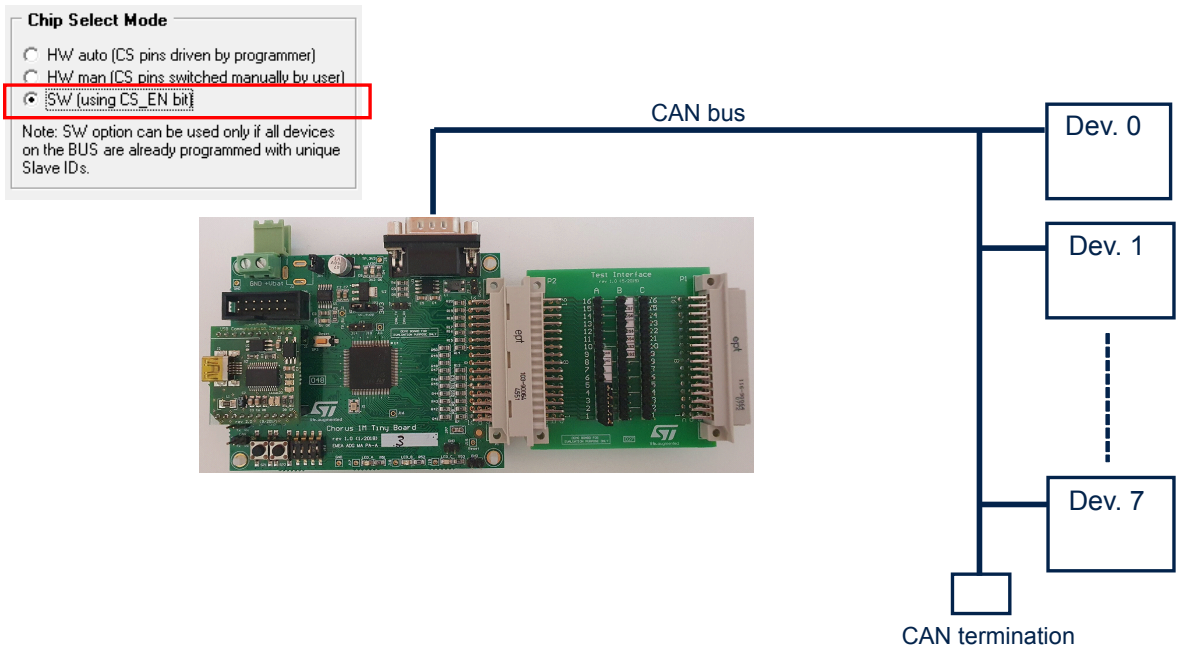
1. **Hardware manual programming (HW man):** when selected, it allows semiautomatic programming and verification of connected devices (either factory default or already programmed), while CSx pins need to be controlled manually by the user (for example, by jumper). Semiautomatic means that the user can access only a device with CSx pin pulled high. (“Program All”, “Read All”, “Autodetect All” functions are not possible).

Figure 132. **Manual (HW) programming mode**



1. **Software programming (SW)**: this mode allows automatic programming and verification of all connected devices without need of HW CS pin management. As a prerequisite, all devices must be already programmed with unique ResponderIDs.

Figure 133. **Software programming mode**



The programming/verification sequences work regardless the direct input (DIN) pin state.

However, according to DIN pin state, the device behaves differently after completing the programming:

- DIN = ‘0’ (low state)

Device goes automatically in standby after completing each FTP access (provided that also all FS_OUT_ENx bit are 0). This means that when the device is accessed again (wake-up), previously programmed FTP settings are already applied, so for example, the device is accessible on new programmed ResponderID if it was changed by the previous FTP write command.

- DIN = ‘1’ (high state)

Device stays active after FTP programming so programmed FTP settings are not applied yet. This is done after the next Power on Reset (supply reconnection). This means that if the new ResponderID is programmed the device is still accessible on a previous ResponderID until next power on reset (supply reconnection) event.

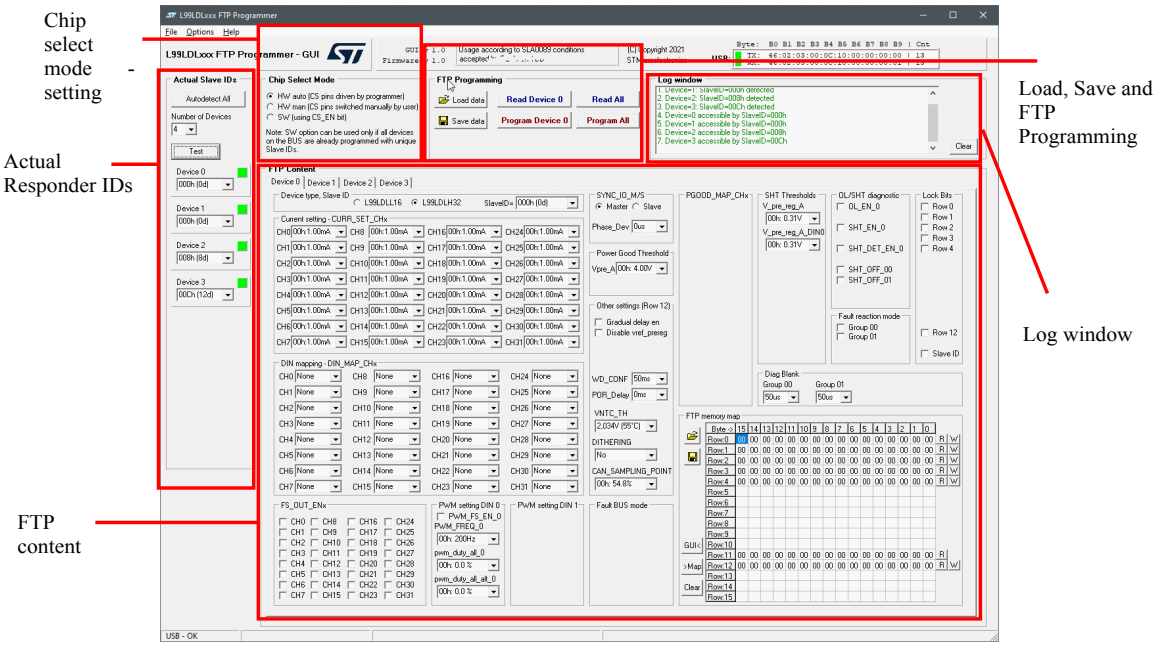
6.4.2 Programming tool - graphical user interface (GUI) description

The Figure 134, shows how the GUI looks like. There are 5 main sections:

- **Chip select mode setting** - already described in the previous chapter;
- **Actual ResponderIDs** - on this section, it is possible to:
  - Perform automatic detection of connected ResponderIDs by trying to access all possible 128 addresses for each active CS line. Function is available only in “HW auto” (all devices tested) or “HW man” (only selected CSx device checked) modes. Successfully detected device is marked by green color and detected address is shown. The detection result is also shown in the “Log window”;
  - Have the number of devices on the bus. It is set either manually or automatically after loading new FTP data from XML file (depending on the number of devices in the XML);
  - Test if manually selected (or auto-detected) ResponderIDs are accessible (green-device accessible, red-device not accessible). The test result is also shown in the log window;
  - Perform manual adjustment of ResponderID (=0 in case of factory default device). Address is found automatically upon pressing “Autodetect” button;
  - Read device availability status (updated upon pressing “Autodetect”, “Test” button or upon manual values selection):
    - Green = device accessible
    - Red = device not accessible
    - Yellow = availability status not checked yet
- **FTP content** - on this section, it is possible to:
  - Select: device ‘x’ (each device has its own tab), device type (L99LDLL16 or L99LDLH32), ResponderID to be programmed;
  - Read FTP data GUI in human readable form (changes automatically propagated in HEX memory map);
  - Load/Save data to XML file, only for selected device;
  - Refresh GUI according to both HEX and GUI data (normally done automatically)
  - Reset to factory default (all 0);
  - Have HEX memory (editable, changes automatically propagated in human readable form);
  - Read or write individual FTP rows (status shown in the log window)
- **Load, save, FTP programming** - on this section, it is possible to:
  - Load data: load FTP data from XML file. The file can contain data up to 8 devices. After loading the file, the number of devices is updated, and content is shown in human readable form as well as in the HEX form;
  - Savedata: save current FTP configuration of all devices;
  - Read Device x: read all customer FTP rows from Device ‘x’ (‘x’ = actual device selection). The read status is shown in the Log window;
  - Program Device x: write all customer FTP rows with actual data in GUI. Each write command is followed by read verification. The status is shown in the Log window;
  - Read All: read device x repeated for all devices. The status is shown in the Log window;
  - Programm All: Program Device x repeated for all devices. The status is shown in the Log window.

- **Log window** - on this section, there are:
  - The log, showing status of the following user actions:
    - Autodetect and test functions
    - Read device x - the status is reported for each FTP Read command (7 lines shown for L99LDLH32)
    - Program device x - the status is reported for each FTP Write/Read command (6 lines shown for L99LDLH32)
    - Read/program all - same as Read/Write device x repeated for each device
    - R and W buttons in HEX editor (individual row access)
  - The error warning, shown in case of any error in the log history
  - Clear button:
    - Entire log can be cleared by clear button (including the error warning);
    - The log is continuously filled (the number of lines is not limited)

Figure 134. Graphical user interface (GUI) - overview



7 Device in ASIL applications

7.1 Introduction

The L99LDLH32 provides many diagnostic and protection functions to perform various plausibility checks to meet the requirements of ASIL applications.

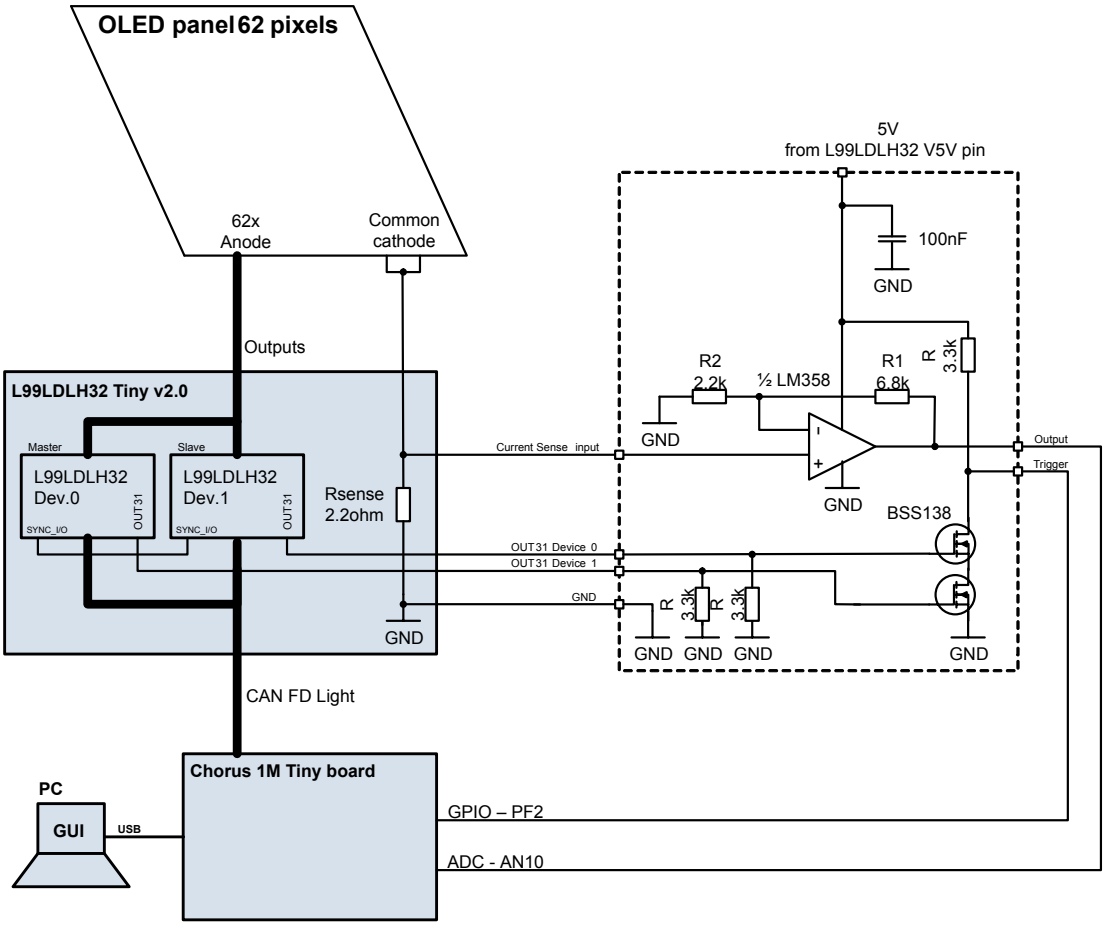
7.2 External current monitoring in common cathode

External current monitoring of each output channel would require many external components, given that the device has 32 channels. This would lead to very high complexity and cost. Fortunately, in most applications, all channels are used to control the same light function. Therefore, it is sufficient to monitor the sum of the current of all channels. This can be accomplished by adding a shunt resistor to the common cathode path. The main challenge is then to properly synchronize the voltage sensing with the PWM signal of the output channels, taking into account the variable PWM duty cycles of each channel and the fact that one OLED panel can be driven by several devices. In the following chapters, 2 possible solutions are described and they are supported by a real evaluation performed in the lab.

7.2.1 Synchronous current sensing

This current measurement method is based on measuring the sum of the current across the OLED panel in the common cathode GND path at the given time of a PWM period. The concept is explained using the example in Figure 135, with a 62 segment OLED panel driven by 2 L99LDLH32 devices. Since each device has 32 channels, 1 channel per device remains free and it is as a trigger signal for ADC synchronization.

Figure 135. **Synchronous current sensing - block diagram**



The OLED panel consists of 58 segments at 5 mA and 4 segments at 2.5 mA. This results in the total current of 300 mA when all segments are active. The current sense voltage, from 2.2 Ω sense resistor, is amplified by non-inverting amplifier with gain = (6.8 k+2.2 k)/2.2 k = 4.091. The output is directly connected to µC ADC input. The maximum expected output voltage, when all 62 OLED segments are activated, is: 300 mA * 2.2 Ω * 4.091 = 2.7 V. The measurement is performed by the microcontroller's ADC once per device's PWM period.

The ADC trigger signal is generated using unused device outputs 31, permanently configured to minimum current (1 mA) and duty cycle according to the desired sampling point (usually close to the beginning of the period). Both outputs are combined in one signal using 2 MOSFETs (=NAND function). The resulting trigger signal is low when both device outputs are high (PWM on-state). The ADC sampling is performed at the rising edge of the trigger signal (when one of the devices outputs 31 returns low at the end of its PWM on-time). This concept allows to detect a case when both devices are not properly synchronized. The trigger signal is not available (remains continuously high) if the PWM on-time pulses of the two outputs 31 do not overlap.

As can be seen from the previous text, the PWM signals of both devices must be synchronous. This is achieved by connecting the devices via SYNC_IO pins, configuring one device as master and the other as slave, and by synchronization the command sequence to ensure that the PWM signals of both devices are properly aligned.

7.2.1.1 *Application initialization example*

Device configuration:

- Device 0: configured as master SYNC I/O M/S = [0]
- Device 1: configured as slave SYNC I/O M/S = [1]
- SYNC I/O pins interconnected on PCB

- PHASE_DEV = 000 for device 0 and 1
- OUT_DELAY = [0] for device 0 and 1
- Device 0, Ch31: CURR_SET=00h (1 mA), DUTY=6Fh (2%)
- Device 1, Ch31: CURR_SET=00h (1 mA), DUTY=6Fh (2%)
  - Note: the duty cycle value of 2% for device 0 and device 1 is only an example. In other cases, a higher duty cycle value may be required depending on the timing accuracy of the synchronization sequence and the sampling point position requirement.

Synchronization sequence:

- Determine PWM period
  - considering adjusted PWM frequency (+/- 5% accuracy)
  - or directly measured by the microcontroller on the trigger signal (to allow this measurement, the duty cycle of the output 31 of one device needs to be temporarily set to 100%)
- Send PWM_SYNC=1 command to device 0
- Delay time (1 PWM period)
- Send PWM_SYNC=1 command to device 1

(1 PWM period delay between e/o master frame 1 and e/o master frame 2)

Note: in this example, the PWM period was determined by the measurement with resulting value of 2508 µs (configured PWM frequency was 400 Hz).

Figure 136. Synchronisation sequence overview

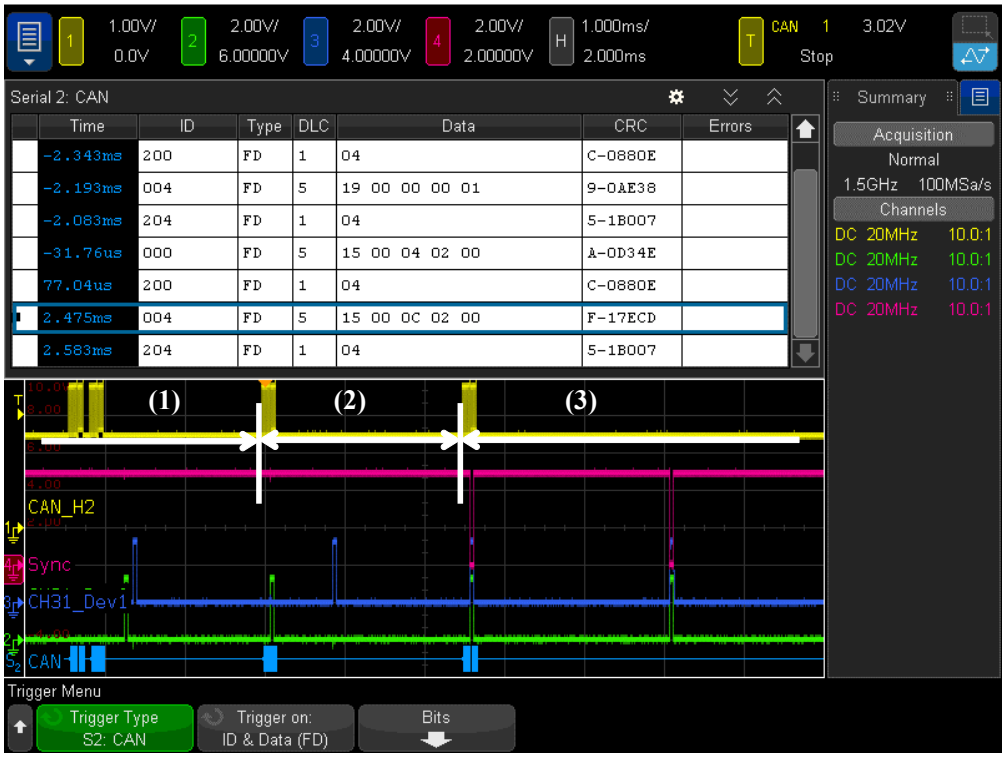


Figure 137. **Detail to 1st and 2nd PWM_SYNC=1 command**

An example of the synchronization sequence is shown in Figure 136 and Figure 137. The green plot shows the output 31 of device 0, the blue plot the output 31 of device 1, the purple plot the trigger signal and the yellow plot the CAN_H signal. In phase (1), a phase shift between both PWM signals can be seen. Because the phase shift is higher than the configured duty cycle, there is no overlap, so the resulting trigger signal remains high.
At the beginning of phase (2), the command PWM_SYNC=1 is sent to device 0. Immediately after receiving this command, the PWM counter is reset. The reset can be seen in detail in Figure 137 - the green plot. At the beginning of phase (3), exactly after the delay of 1 PWM period after the previous command was sent, another PWM_SYNC=1 command is sent to device 1. The reset can be seen in detail in Figure 137 - the blue plot. After this command, the PWMs of both devices are aligned and the trigger signal is therefore available (NAND function of both PWM signals) - see purple plot. The ADC sampling is then triggered by the rising edge of the trigger signal.

7.2.1.2 *Application Run-Mode example*

During the application run mode, the following steps need to be performed:

Application run mode:

1. Receive new PWM duty cycle pattern (for example, every 10 ms)
2. Determine minimum duty cycle per device
3. Calculate VEXP (expected sense voltage with all n for active channel)

   $V_{EXP} = Gain_{LA} * Rsense \sum_{n=0}^{n=61} CURR\_SET\_CH(n)$
4. Refresh PWM duty cycle, program ch31 with a duty cycle corresponding to the desired sampling point
5. GPI/O - PF2 low/ligh transition triggers next step 6
6. ADC - AN10 conversion → VSAMPLE
7. VEXP - VSAMPLE > error threshold?
   - No: no error
   - Yes: error handling routine
8. Wait for the next duty cycle update request x (go to step 1)

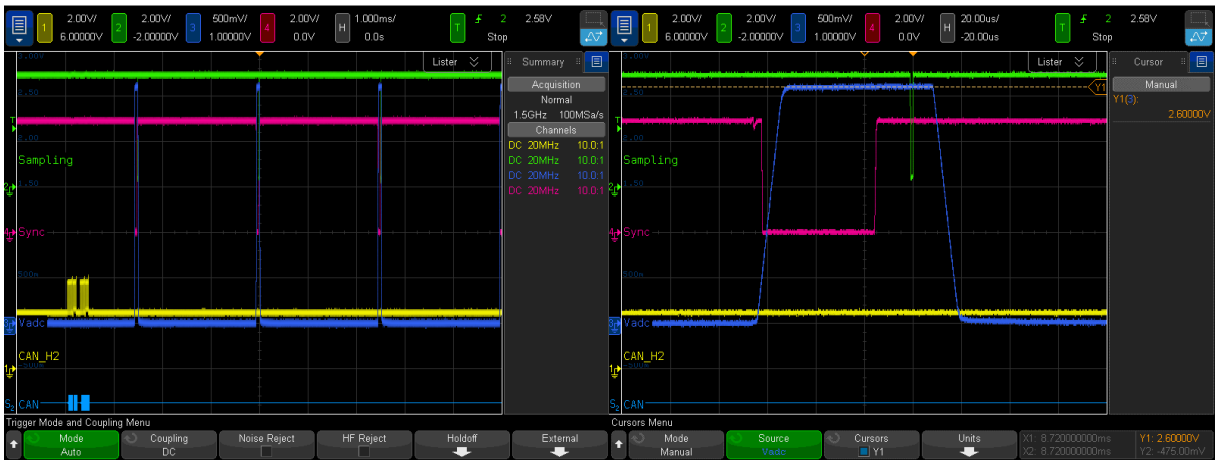
Prerequisite: minimum duty cycle device > PWM period spread + sampling delay

Note: In case no transition detected repeat "Application Initialization" sequence or increase duty cycle.

The example in Figure 138 shows a condition where all channels are active at 3% duty cycle. The blue plot is the sensing voltage, the purple plot is the ADC trigger signal, the green plot is the end of ADC conversion notification and yellow plot is the CAN_H signal. Note that there is approximately 15 µs delay between the ADC trigger condition (rising edge of the purple plot) and the end of the ADC conversion (falling edge of the green plot).

The measured sense voltage is 2.61 V (average of 10 consecutive measurements over 10 PWM periods, see Figure 138). This corresponds to the calculated expected voltage of 2.707 V with error of -3.6%. In addition to the measurement error itself, this error also includes the device's current source error because the measured value is compared to the target set point current (not the actual current).

Figure 138. **Example - all pixels ON at 3% without gradual delay**



Theoretical ADC voltage calculation		
Rsense	2.2	Ohm
Full pixel current setting (49h):	5.01	mA
Half pixel current setting (1Ch)	2.54	mA
Amplifier gain:	4.091	-
Number of 5mA pixels:	**58**	-
Number of 2.5mA pixels:	**4**	-
ADC voltage:	**2.707**	**V**

0: ADC = 0853h =>	2.602	V
1: ADC = 085Bh =>	2.612	V
2: ADC = 0863h =>	2.621	V
3: ADC = 085Ah =>	2.611	V
4: ADC = 0852h =>	2.601	V
5: ADC = 0863h =>	2.621	V
6: ADC = 085Ch =>	2.613	V
7: ADC = 085Bh =>	2.612	V
8: ADC = 0853h =>	2.602	V
9: ADC = 0852h =>	2.601	V
ADC average:	2.610	V
Theoretical value:	2.707	V
Error:	-3.6	%

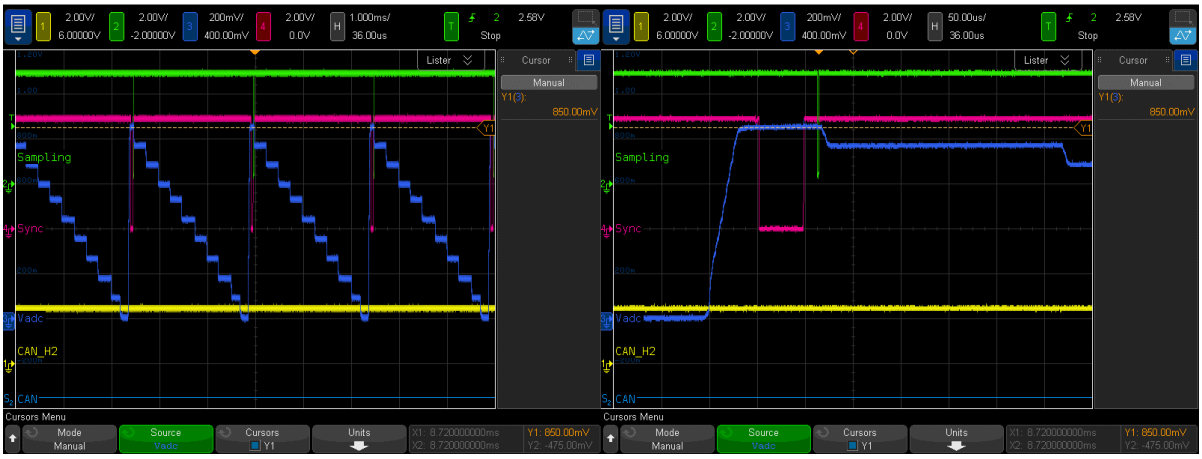
The other example in Figure 139 shows a condition where 20 channels are active, each with a different duty cycle value:

- Device 0:
  - Ch0=5%, Ch2=15%, Ch4=25%, Ch6=35%, Ch8=45%, Ch10=55%, Ch12=65%, Ch14=75%, Ch16=85%, Ch18=95%
  - Other channels OFF
- Device 1:
  - Ch0=5%, Ch2=15%, Ch4=25%, Ch6=35%, Ch8=45%, Ch10=55%, Ch12=65%, Ch14=75%, Ch16=85%, Ch18=95%

Other channels OFF

The measured voltage is 0.856 V (average of 10 consecutive measurements over 10 PWM periods - see Figure 139). This corresponds to the calculated expected voltage of 0.88 V with an error of -2.7%. In addition to the measurement error itself, this error also includes the device's current source error because the measured value is compared to the target set point current (not the actual current).

Figure 139. **Example - 20 pixels ON at different duty cycles with gradual delay**



					0: ADC = 02C3h =>	0.863	
					1: ADC = 02BBh =>	0.853	
					2: ADC = 02C3h =>	0.863	
					3: ADC = 02C3h =>	0.863	
					4: ADC = 02BBh =>	0.853	
					5: ADC = 02BBh =>	0.853	
Theoretical ADC voltage calculation					6: ADC = 02BBh =>	0.853	
Rsense			2.2	Ohm	7: ADC = 02B3h =>	0.844	
Full pixel current setting (49h):			5.01	mA	8: ADC = 02BBh =>	0.853	
Half pixel current setting (1Ch)			2.54	mA	9: ADC = 02C3h =>	0.863	
Amplifier gain:			4.091	-			
Number of 5mA pixels:			**19**	-	ADC average:	0.856	V
Number of 2.5mA pixels:			**1**	-	Theoretical value:	0.880	V
ADC voltage:			**0.880**	V	Error:	-2.7	%

Summary:

This method of current monitoring gives reasonably accurate results. Only a few factors contribute to the measurement error: tolerance of R_{SENSE}, R1, R2, OPAMP offset and ADC accuracy.

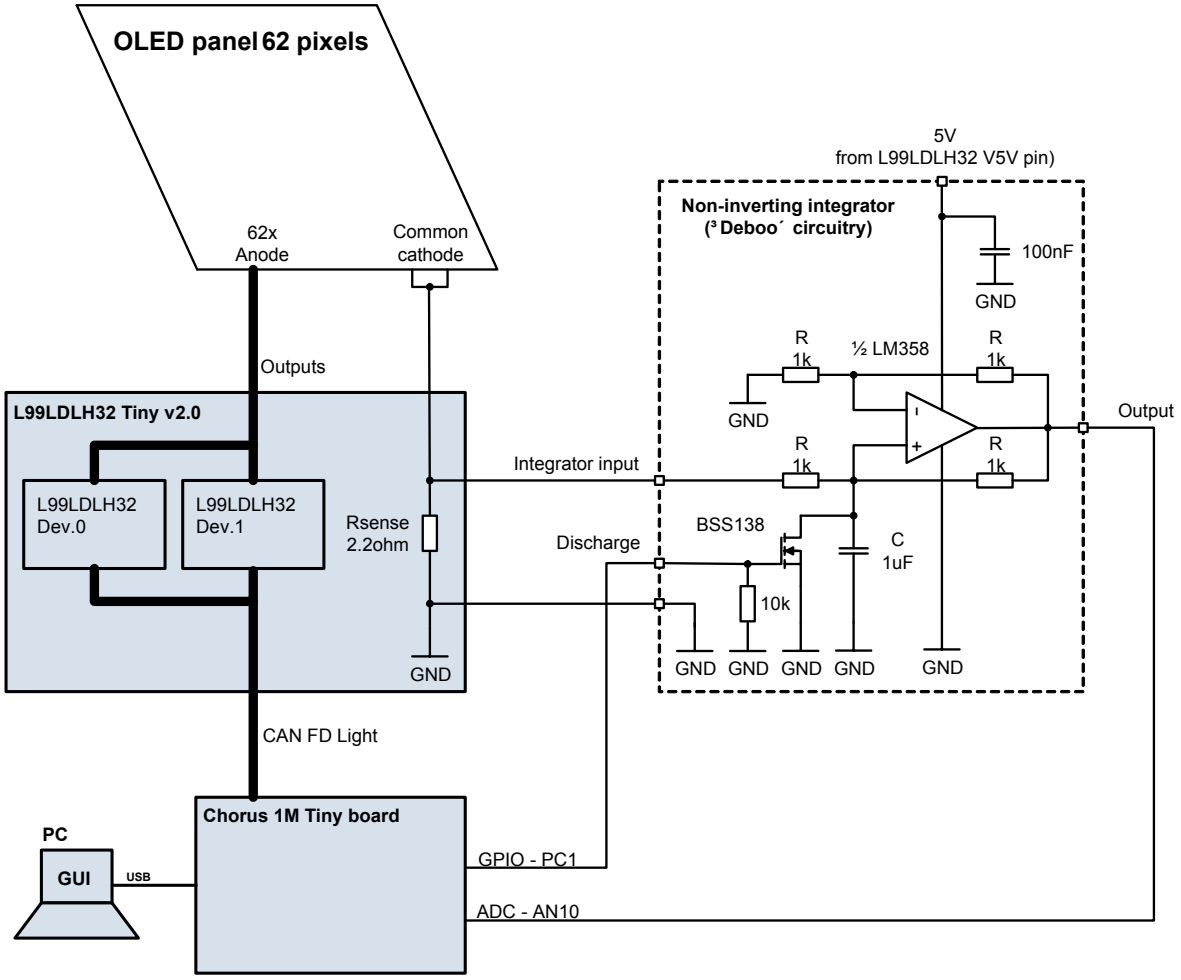
7.2.2 Asynchronous current sensing with integrator

This current measurement method is based on measuring the sum current across OLED panel in the common cathode GND path. The shunt resistor voltage is fed to a noninverting integrator with an integration period equal to the PWM period. The integrator output voltage is measured at the end of the integration window using the ADC input of the microcontroller. This allows the average value of the summing current to be measured over the PWM period regardless of the phase shift between the PWM period and the integration window. This concept does not occupy the output channel for ADC triggering, so all outputs can be dedicated to the light function.

7.2.2.1 Application example - HW implementation

The concept is explained using the example in Figure 140, with a 62 segment OLED panel driven by 2 L99LDLH32 devices. As in the previous method, the OLED panel consists of 58 segments at 5 mA and 4 segments at 2.5 mA. This results in the total current of 300 mA when all segments are active.

Figure 140. **Asynchronous current sensing with integrator - block diagram**



The standard simple inverting integrator circuit is not very suitable for this application as it provides a negative output voltage and requires a negative supply. Therefore, the noninverting solution shown in Figure 140 was used. The output voltage is given by the following formula:

$$V_{OUT} = \frac{2}{RC}\int v_{IN}(t)dt$$

For sufficient accuracy over the entire temperature range, a film capacitor (usually with 5% tolerance) should be used along with resistors with 1% tolerance. For OP Amp, it is important to select a type that can handle input and output voltages up to the GND supply terminal (not necessarily "rail-to-rail"). Many inexpensive general-purpose OP Amp's with pnp input stage meet this requirement, such as the LM358 (dual OP Amp) used in this evaluation. A single version of the TS321 in a SOT23-5 package is also available. The typical offset of this type is 2 mV, which can result in an error of 10 mV in the output voltage when considering the integration time of 2.5 ms and the values of the components used in this example (this corresponds to an error of 0.9 mA in the summed OLED current).

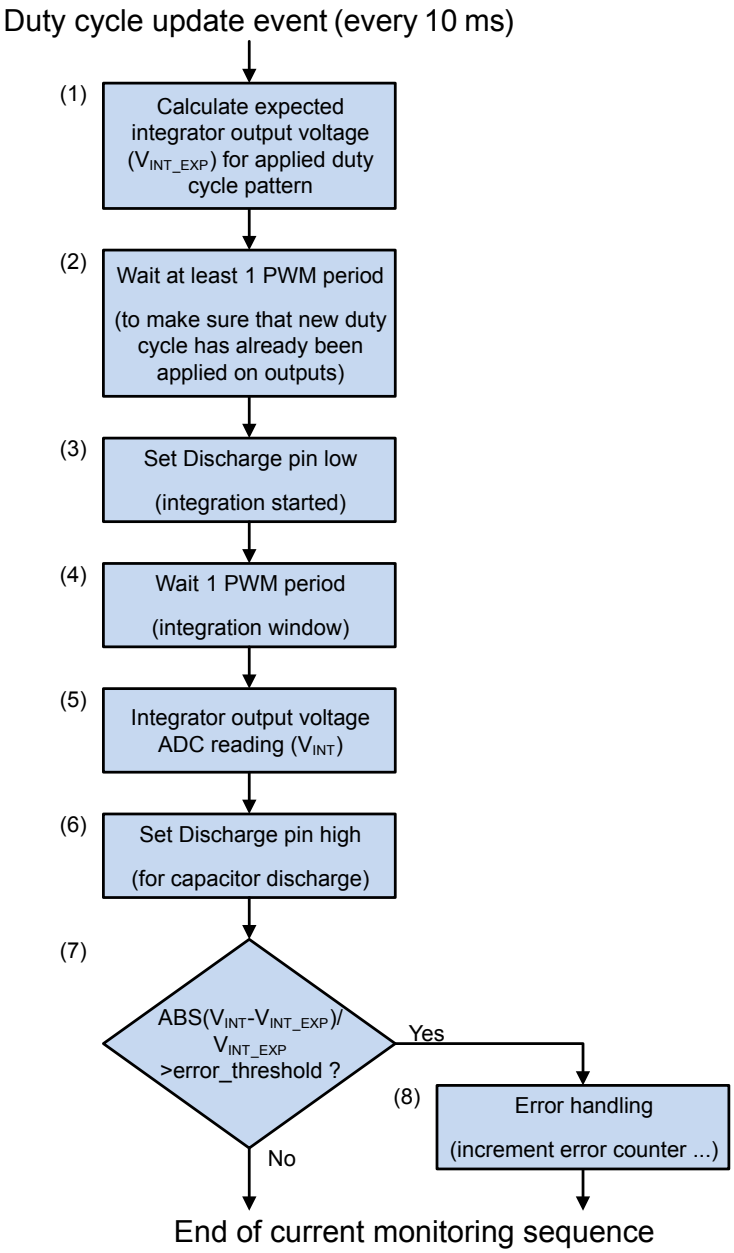
The value of the current sensing resistor is chosen as a compromise between accuracy, power dissipation and acceptable voltage drop. With a 2.2 Ω resistor used, the voltage drop is 661 mV and the power dissipation is 200 mW with all segments active with 100% duty cycle. This additional voltage drop must be considered when programming the diagnostic short circuit threshold.

The RC integrator values are chosen so that the maximum output voltage (all channels on at 100% duty cycle) is 3.307 V, knowing that the maximum operating output voltage of the OP Amp used is 3.5 V (1.5 V below the 5 V supply voltage). This also corresponds to the input range of the ADC (5 V full-scale).

7.2.2.2 *Application example - SW implementation*

During the application run mode, the following steps need to be performed:

Figure 141. Software flow chart



Step (1), the calculation of the expected output voltage of the integrator, is performed according to the following equation:

$$V_{INT\_EXP} = K1 + K2 * \sum_{n=0}^{n=61} CURR\_SET\_CH(n) * DUTY\_CH(n)$$

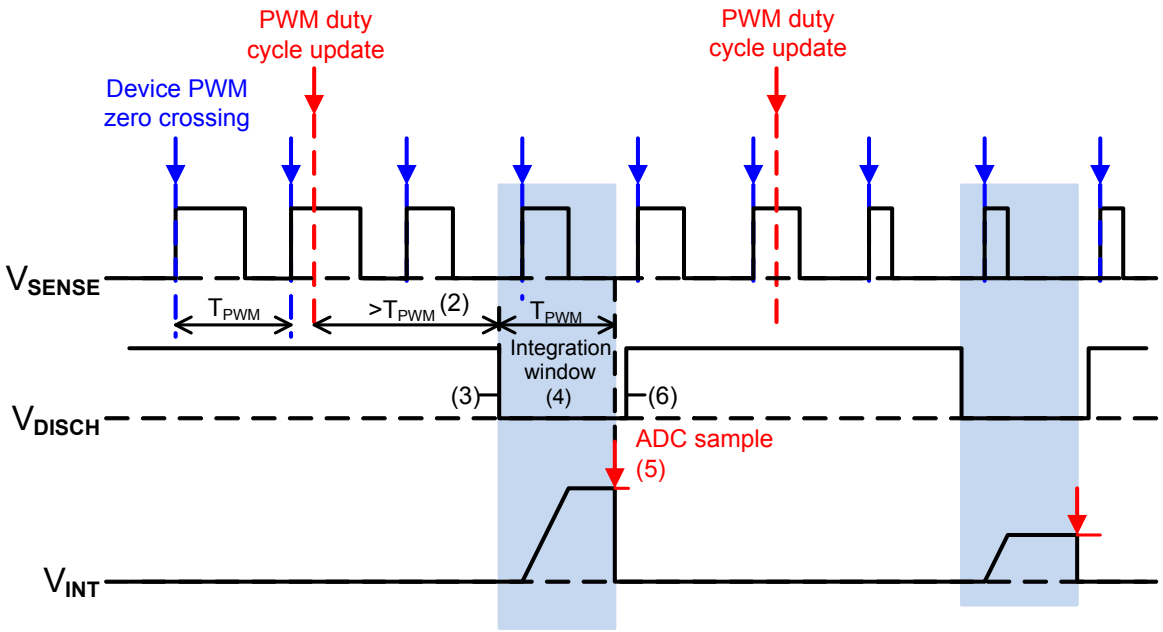
Where K1 and K2 are constants determined during EOL calibration:

- K1 is calibration constant eliminating offset error (mostly caused by Op Amp offset)
- K2 is calibration constant eliminating proportional error (mostly caused by error of nominal values of all passive components and current regulators).

Note: depending on number of possible duty cycle patterns, the SUM part of the formula could be pre-calculated in advance (compiled in the code) for every duty cycle pattern, to decrease the computation load of the microcontroller.

Steps (2) to (6) are explained in the Figure 142.

Figure 142. Principle of the integration



The principle is also demonstrated in the oscilloscope screenshots in Figure 143 and Figure 144. The PWM frequency is 400 Hz, duty cycle update period is 10 ms (2 broadcast commands for each update) and integration started ~4 ms after each duty cycle update. The red plot is the integrator discharge signal, the green plot is the R_{SENSE} voltage, the blue plot is the integrator output voltage (ADC input), and the yellow plot indicates duty cycle update events (ends of CAN broadcast commands).

Figure 143. **Duty cycle update and current measurement every 10 ms**

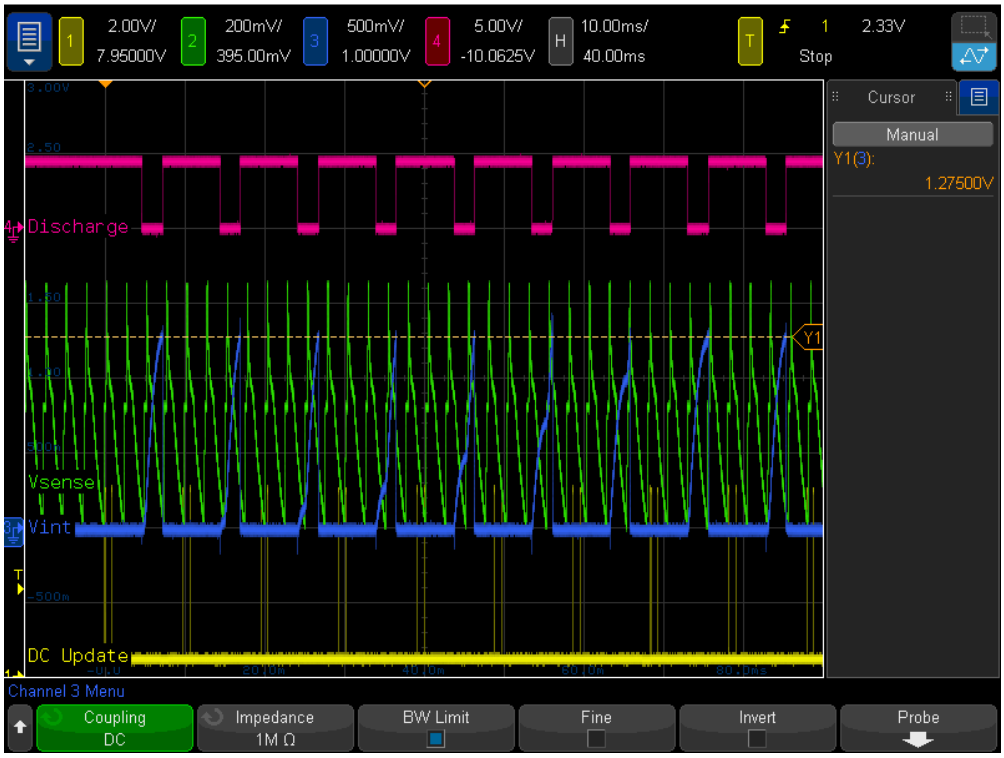
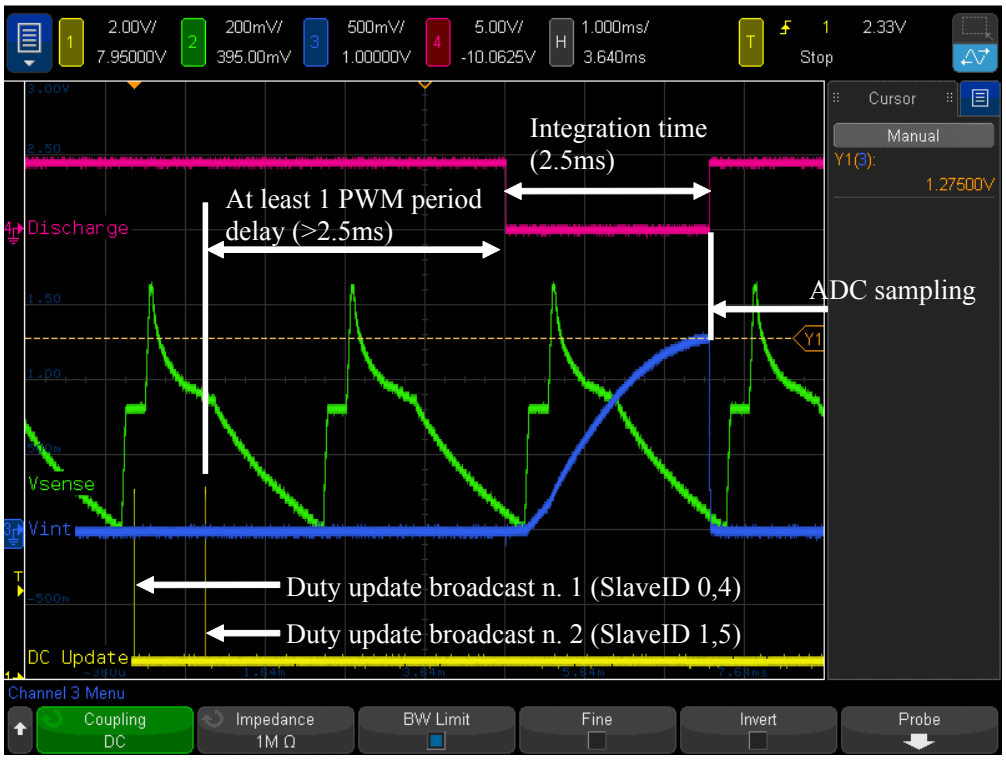


Figure 144. **Detail of duty cycle update and integration period**



The next example shows the case when all channels are active with 50.7% duty cycle and with gradual delay enabled. The measurement sequence is performed 10 times (every 10 ms). As can be seen from the integration voltage detail in Figure 145 and Figure 146, the voltage at the end of the integration period is always the same, regardless of the phase relation between the PWM and the integration window.

Figure 145. Example with all channels at 50.7% and gradual delay (1st pulse detail)

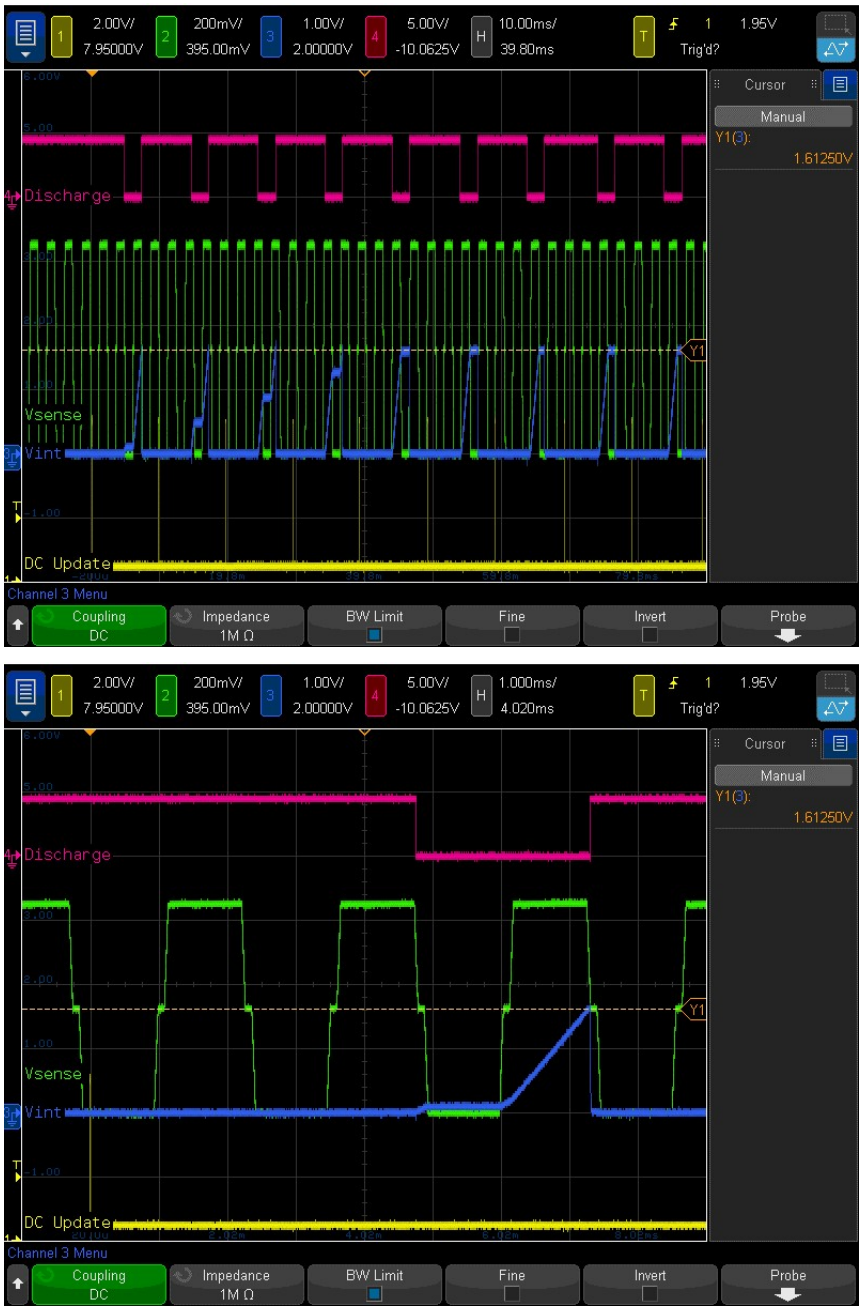


Figure 146. **Example with all channels at 50.7% and gradual delay (2nd, 3rd and 5th pulse)**

Vint (scope)	1.612	V	
Vint (adc samples):			
0: ADC = 0524h =>		1.607	V
1: ADC = 051Bh =>		1.596	V
2: ADC = 0523h =>		1.606	V
3: ADC = 0520h =>		1.602	V
4: ADC = 052Dh =>		1.618	V
5: ADC = 0529h =>		1.613	V
6: ADC = 0523h =>		1.606	V
7: ADC = 0527h =>		1.611	V
8: ADC = 0526h =>		1.609	V
9: ADC = 0520h =>		1.602	V
ADC - average:		1.607	V
Expected value:		1.676	V
Error:		-4.1	%

The measured voltage is 1.607 V (average of 10 consecutive measurements over 10 PWM periods). This corresponds to the calculated expected voltage of 1.676 V with an error of -4.1%. In addition to the measurement error itself, this error also includes the device's current source error because the measured value is compared to the target set point current (not the actual current).

8 Conclusions

The advantage of this method is that the measurement is independent from the phase relation between the current measurement and PWM signals. Important is the integration period, which must be equal to the PWM period. This leads to easier initialization (no synchronization sequence is required). The result of the measurement is the average current value over the PWM period, so the value directly reflects the brightness of the light function (the PWM duty cycle is also reflected in the result). In addition, it need any spare device outputs, so they can all be dedicated to the light function.

The disadvantage is the hardware complexity and higher sensitivity to components tolerances, requiring an EOL calibration. These latter points can be overcome by implementing the integrator in software if the appropriate microcontroller resources are available.

Revision history

Table 20. **Document revision history**

Date	Revision	Changes
24-Aug-2022	1	Initial release.

Contents

List of tables

List of figures