AN2507
Application note

High-Power Camera Flash LED Driver with I²C™

Introduction

This application note explains the design of a FLASH LED driver using the STCF03 device, which is a Buck-Boost current mode converter with an I²C interface. The schematic, functional description, recommendations for PCB Layout and external components selection are also discussed in this application note. This device is designed for driving a single LED with a forward voltage range from 2.7 to 5 V. A detailed functional description can be found below.

Package and demo board top view



Version B - Version C



Version A



Contents

1 Schematic description 6

1.1 Application schematic 6

2 Selection of external components 7

2.1 Input and output capacitor selection 7

2.2 Inductor selection 7

2.3 LED selection 7

2.4 R_{FL} selection 8

2.5 R_{TR} selection 8

2.6 NTC AND R_X resistor selection 8

3 PCB design 10

3.1 PCB design rules 10

3.2 PCB layout 10

3.2.1 A four-layer PCB with application area 45.1 mm^2 for BGA package, version B 10

3.2.2 A two-layer PCB with application area 72.4 mm^2 for QFN package . . . 13

3.2.3 A four-layer PCB with application area 45.1 mm^2 for BGA package, version C 14

4 Internal registers 16

4.1 Accessing the internal registers 16

5 Operation modes 17

5.1 SHUTDOWN mode 17

5.2 SHUTDOWN mode with the NTC-feature activated 17

5.3 READY mode and NTC 18

5.4 TORCH mode 18

5.5 FLASH mode 19

6 The STATUS register and the ATN pin 21

6.1 The STATUS register 21

7 Reading and writing to the STCF03 registers through the I^2C bus . . 22

7.1 Writing to a single register 22
7.2 Writing to multiple registers with incremental addressing 22
7.3 Reading from a single register 23
7.4 Reading from multiple registers with incremental addressing 23

8 Examples of register setup for each mode 24
8.1 Example 1: 600 mA FLASH with 700 ms duration 24
8.2 Example 2: 25 mA TORCH 25
8.3 Example 3: an Auxiliary LED running at 10 mA for 500 ms 26
8.4 Example 4: Red-eye reduction (multiple short flashes) 27
8.5 Example 5: A FLASH pulse longer than 1.5 s 28

9 Revision history 30

List of tables

Table 1. Recommended components . . . 9
Table 2. Accessibility of internal registers . . . 17
Table 3. COMMAND register data to enter SHUTDOWN mode (version B) . . . 17
Table 4. COMMAND register data to enter SHUTDOWN mode (version A and C) . . . 17
Table 5. COMMAND register data to enter SHUTDOWN mode with NTC activated (version A and C) 18
Table 6. COMMAND register data to enter READY mode . . . 18
Table 7. COMMAND register data to enter READY mode with NTC ON . . . 18
Table 8. COMMAND register data to enter TORCH mode . . . 19
Table 9. COMMAND register data to enter FLASH mode . . . 19
Table 10. STATUS register bits . . . 21
Table 11. Effect of the STATUS register bits on the operation of the device . . . 21
Table 12. TORCH mode and FLASH mode dimming registers settings . . . 24
Table 13. COMMAND register data to enter FLASH mode . . . 24
Table 14. DIMMING register data for the FLASH mode . . . 25
Table 15. I^2C data packet for activating the FLASH mode . . . 25
Table 16. COMMAND register data for the TORCH mode . . . 25
Table 17. DIMMING register data for the TORCH mode . . . 25
Table 18. I^2C data packet to activate TORCH mode . . . 26
Table 19. I^2C data packet for terminating the TORCH mode . . . 26
Table 20. COMMAND register data for the AUX_LED . . . 26
Table 21. COMMAND register data for the AUX_LED . . . 26
Table 22. I^2C data packet for activating the READY mode . . . 27
Table 23. I^2C data packet for activating the AUX_LED . . . 27
Table 24. COMMAND register data for FLASH mode . . . 27
Table 25. DIMMING register data for the FLASH mode . . . 27
Table 26. I^2C data packet for activating the FLASH mode . . . 28
Table 27. I^2C data packet for activating the FLASH mode . . . 28
Table 28. 1^{st} I^2C data packet to restart the FLASH mode . . . 29
Table 29. 2^{nd} I^2C data packet for restart of the FLASH mode . . . 29
Table 30. 3^{rd} I^2C data packet to restart the FLASH mode . . . 29
Table 31. Revision history . . . 30

List of figures

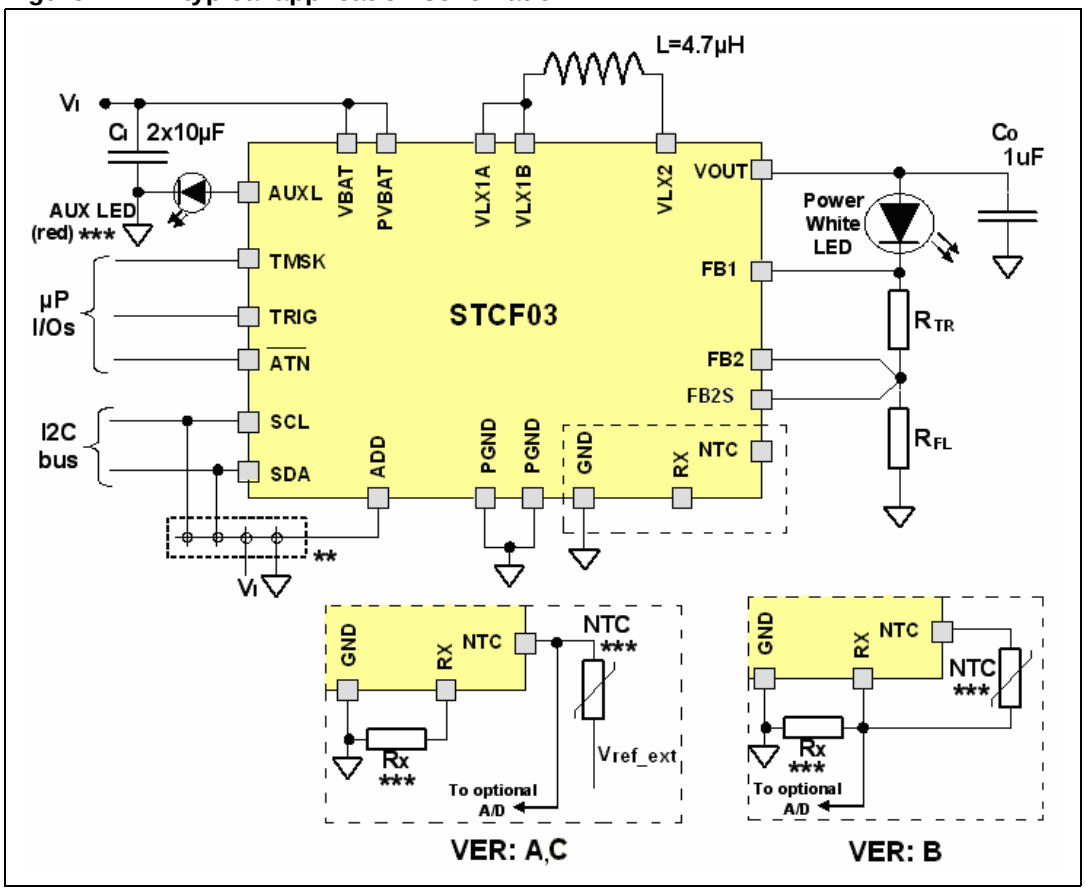
Figure 1. A typical application schematic . . . 6
Figure 2. NTC connection for versions with internal voltage reference . . . 9
Figure 3. Top layer . . . 10
Figure 4. Middle layer 1 . . . 11
Figure 5. Middle layer 2 . . . 11
Figure 6. Bottom layer . . . 12
Figure 7. Top overlay . . . 12
Figure 8. Top layer . . . 13
Figure 9. Bottom layer . . . 13
Figure 10. Top overlay . . . 14
Figure 11. Top layer . . . 14
Figure 12. Middle layer 1 . . . 15
Figure 13. Middle layer 2 . . . 15
Figure 14. Bottom layer . . . 16
Figure 15. Top overlay . . . 16
Figure 16. Splitting the FLASH pulse into several shorter pulses . . . 20
Figure 17. Writing to a single register . . . 22
Figure 18. Writing to multiple registers . . . 22
Figure 19. Reading from a single register . . . 23
Figure 20. Reading from multiple registers . . . 23
Figure 21. Multiple flashes handled by the TRIG pin . . . 28
Figure 22. I^2C bus packets timing for a FLASH lasting longer than FTIM max . . . 30

1 Schematic description

The FLASH LED driver STCF03 has a high operational frequency (1.8 MHz) which allows the usage of small-sized external components. The three versions (A, B and C) differ in the way the NTC feature is supported.

1.1 Application schematic

Figure 1. A typical application schematic



*Note: ** Connect to V_I, or GND or SDA or SCL to choose one of the 4 different I^2C Slave Addresses*

*Note: *** Optional components to support auxiliary functions*

- Version A: STCF03PNR - QFN package, external reference for NTC protection
- Version B: STCF03ITBR - BGA package, internal reference for NTC protection
- Version C: STCF03TBR - BGA package, external reference for NTC protection

2 Selection of external components

2.1 Input and output capacitor selection

It is recommended to use ceramic capacitors with low ESR as input and output capacitors. It is recommended to use 10 µF/6.3 V as a minimum value of input capacitor and 1 µF/ 6.3 V as an optimal value of output capacitor to achieve good stability of the device supplied from low input voltage (2.7 V) at maximum ratings of output power.

Note: see recommended components in Table 1.

2.2 Inductor selection

A thin shielded inductor with a low DC series resistance of winding is recommended for this application. To achieve a good efficiency in step-up mode, it is recommended to use an inductor with a DC series resistance $R_{DCL} = R_D/10$ [Ω, Ω, 1], where R_D is the dynamic resistance of the LED [Ω, Ω, 1].

For nominal operation, the peak inductor current can be calculated by the following formula:

Equation 1

$$I_{PEAK} = \left(\left(\frac{I_{OUT}}{n}\right) + \left(\frac{(V_{OUT} - V_{IN}) \bullet V_{IN}^2}{2 \bullet L \bullet F \bullet V_{OUT}^2}\right)\right) \bullet \frac{V_{OUT}}{V_{IN}}$$

Where:

- I_{PEAK}: Peak inductor current
- I_{OUT}: Current sourced at the VOUT-pin
- n: Efficiency of the STCF03
- V_{OUT}: Output voltage at the VOUT-pin
- V_{IN}: Input voltage at the VBAT-pin
- L: Inductance value of the inductor
- F: Switching frequency

Note: see recommended components in Table 1.

2.3 LED selection

All LEDs with a forward voltage range ranging from 2.7 V to 5 V are compatible with STCF03. The forward voltage spread of any selected LED must however lay within this range (2.7 V to 5 V). It is possible to set the level of the LED current in FLASH mode and TORCH mode by setting the values of the corresponding sensing resistors. The level of the LED current in FLASH mode can be set by changing the external FLASH resistor.

Note: see recommended components in Table 1.

2.4 R_{FL} selection

The value of the R_{FL} resistor can be calculated by the following equations:

$R_{FL} = V_{FB2}/I_{FLASH}$ where V_{FB2} = 226 mV and $P_{RFLASH} = R_{FL} * I_{FLASH}^2$, where P_{RFL} is the power dissipated on the R_{FL} resistor. It is recommended to use a thin metal film resistor with 0805 package size and 1% tolerance. The maximum LED current in FLASH mode for STCF03 is (800 mA) for a battery voltage ranging from 2.7 V to 5.5 V in VQFPN version.

2.5 R_{TR} selection

The value of the R_{TR} resistor can be calculated by following equations:

Equation 2

$$R_{TR} = \left(\frac{V_{REF} - I_{TORCH} \bullet R_{FL}}{I_{TORCH}}\right) \quad \text{and} \quad P_{RTORCH} = R_{TR} \bullet I_{TORCH}^2$$

where P_{RTORCH} is the power dissipated on the R_{TR} resistor. It is recommended to follow the equation R_{TR} = 6.66* R_{FL} to avoid any jump in the current dimming values.

It is recommended to use a thin metal film resistor with 1% or 5% tolerance. The maximum LED current in TORCH mode for SCTF03 is 200 mA for a battery voltage ranging from 2.7 V to 5.5 V

2.6 NTC AND R_X resistor selection

a) A, C versions without an internal reference voltage for the NTC feature.

STCF03 requires a negative thermistor (NTC) for sensing the LED temperature, as well as an R_X resistor and an external voltage reference in order to use the NTC feature. Please refer to the typical application schematic in *Figure 1* VER A,C for more details.

Once the NTC feature is activated, the internal switch connects the R_X resistor to the NTC, and this creates a voltage divider supplied by the external reference voltage connected to the NTC.

If the temperature of the NTC-thermistor rises due to the heat dissipated by the LED, the voltage on the NTC pin increases. When this voltage exceeds 0.56 V, the NTC_W bit in the STATUS register is set to High, and the ATN pin is set to Low to inform the microcontroller that the LED is becoming hot. The NTC_W bit is cleared by reading the STATUS register.

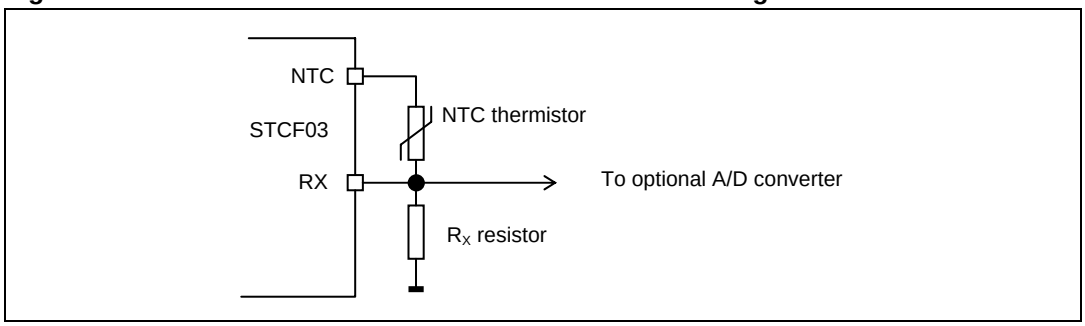
If the voltage on the NTC pin rises further and exceeds 1.2 V, the NTC_H bit in the STATUS register is set to High, and the ATN pin is set to Low to inform the microcontroller that the LED is too hot and the device goes automatically to the READY mode to avoid damaging the LED. This status is latched, until the microcontroller reads the STATUS register. Reading the STATUS register clears the NTC_H bit.

The selection of the NTC and R_X resistors values strongly depends on the power dissipated by the LED and all components surrounding the NTC-thermistor and on the cooling capabilities of each specific application. The R_X and the NTC values in *Table 1* below work well in the demo board presented in this application note. A real application may require a different type of NTC-thermistor to achieve optimal thermal protection.

The procedure to activate the NTC-feature is described in *Section 5.2*.

b) Versions with an internal reference voltage for the NTC. This version requires a different connection between the R_X and NTC resistors. See *Figure 2* below or *Figure 1* version B.

Figure 2. NTC connection for versions with internal voltage reference



Note: Versions with internal reference voltage do not support the SHUTDOWN+NTC mode, because the internal reference voltage is off in SHUTDOWN mode.

Table 1. Recommended components

Component	Manufacturer	Part number	Value	Size
C_I	TDK	C1608X5R0J106M	10 µF	0603
C_O	TDK	C1608X5R0J105M	1 µF	0603
L	TDK	VLF4012AT-4R7M1R1	4.7 µH	3.7 x 3.5 x 1.2 mm
NTC	Murata	NCP21WF104J03RA	100 kΩ	0805
R_{FL}	Tyco	RL73K1JR27JTD	0.27 Ω	0603
R_{TR}	Rohm	MCR01MZPJ6R20	1.8 Ω	0402
R_X	Rohm	MCR01MZPJ15K	15 kΩ	0402
LED	Luxeon LED	LXCL-PW1		

3 PCB design

3.1 PCB design rules

STCF03 is a powerful switching device where the PCB must be designed in line with switched supplies design rules. The power tracks (or wires in demo-board) must be as short as possible and wide enough, because of the high currents involved. It is recommended to use a 4 layers PCB to get the best performance. All external components must be placed as close as possible to STCF03. All high-energy switched loops should be as small as possible to reduce EMI. Most of LEDs need efficient cooling, which could be done by using a dedicated copper area on the PCB. Please refer to the selected LED's reference guide to design the heatsink. Place the R_{FL} resistor as close as possible to the PGND pins and the ground pin of the COUT capacitor. In case a modification of any PCB layer is required, it is highly recommended to use enough vias. Place the NTC resistor as close as possible to the LED for good temperature sensing. Direct connection between GND and PGND is necessary in order to achieve correct output current value. No LED current should flow through this track! Voltage sensing on the R_{FL} resistor must to done on a track from ball FB2 and directly connected to the R_{FL} resistor. Again, no current should flow through this track. Pin FB2S must be connected to the R_{FL} resistor pin. Vias connecting the STCF03 pins to the copper tracks (if used) must be 0.1 mm in diameter for BGA version. It is recommended to use the filled vias.

3.2 PCB layout

3.2.1 A four-layer PCB with application area 45.1 mm^2 for BGA package, version B

(for version C is layout exactly same except the NTC connection, see *Figure 1*)

Figure 3. Top layer

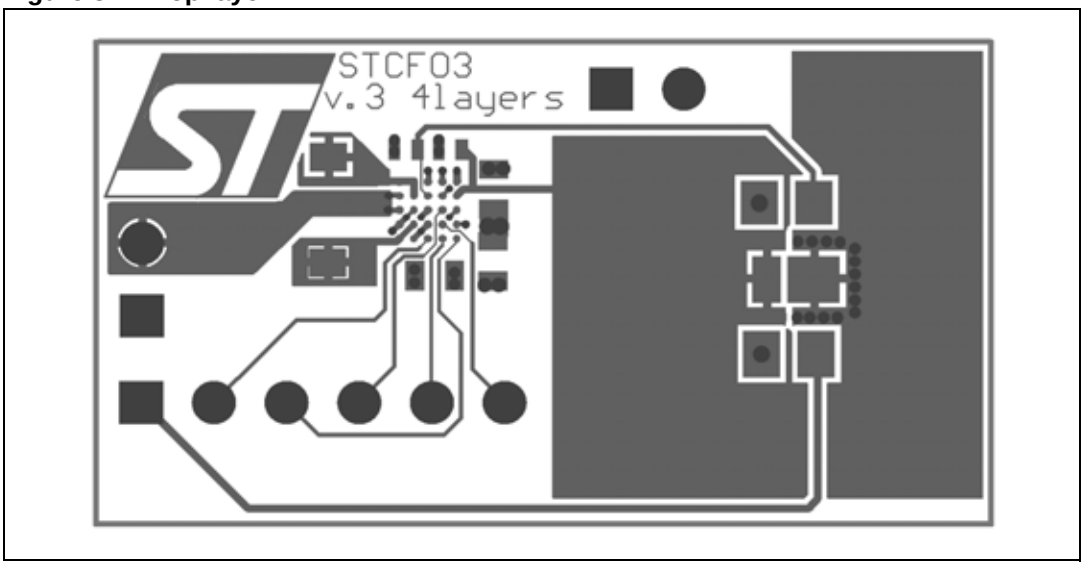


Figure 4. Middle layer 1

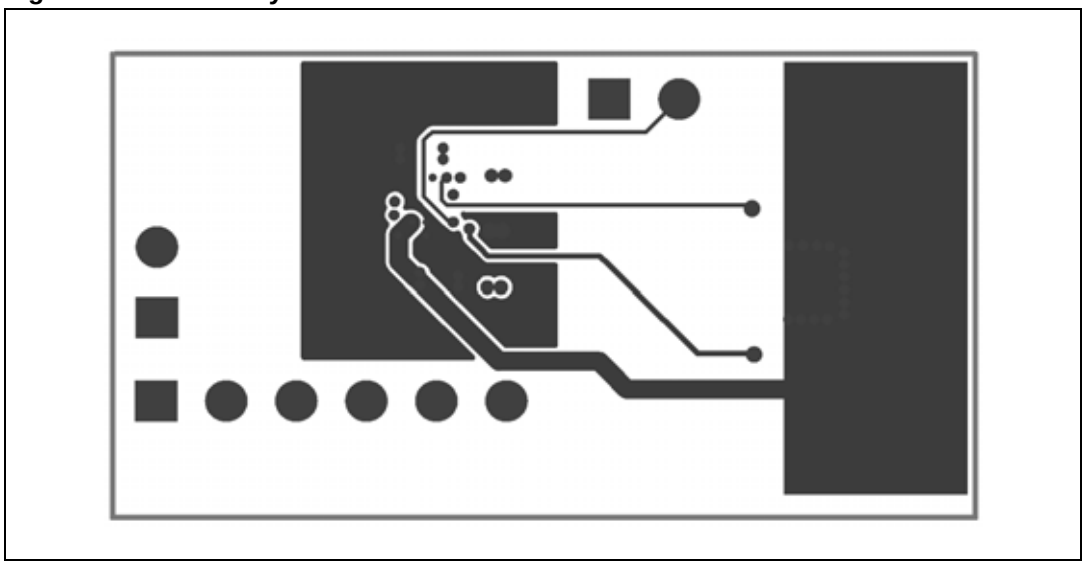


Figure 5. Middle layer 2

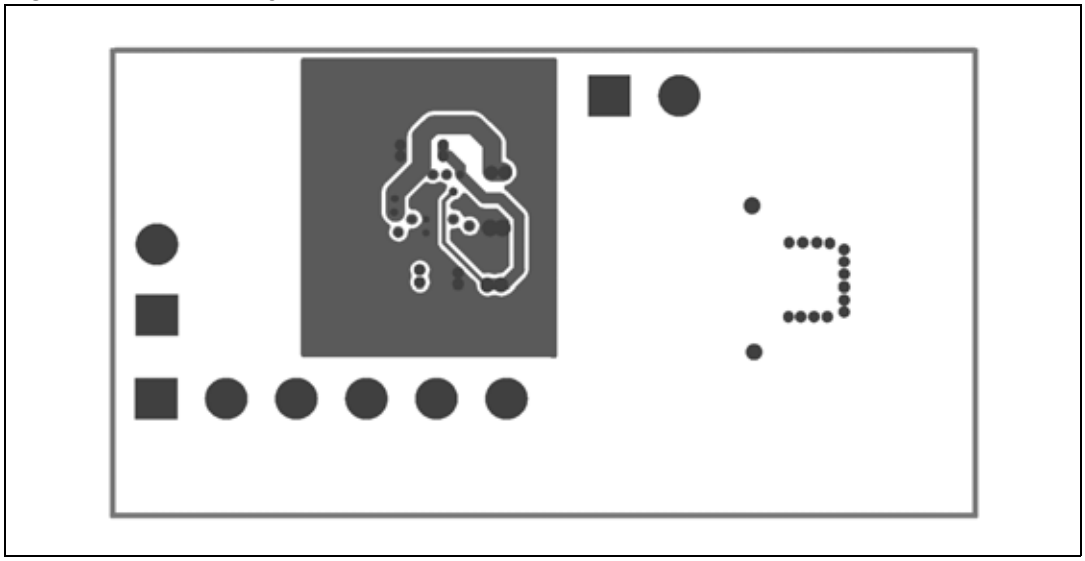


Figure 6. Bottom layer

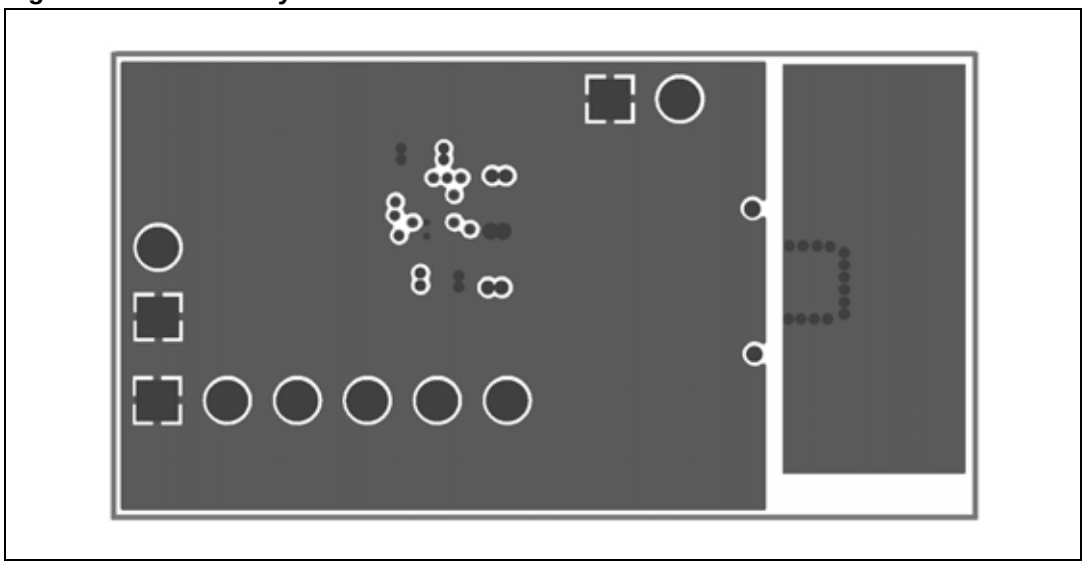
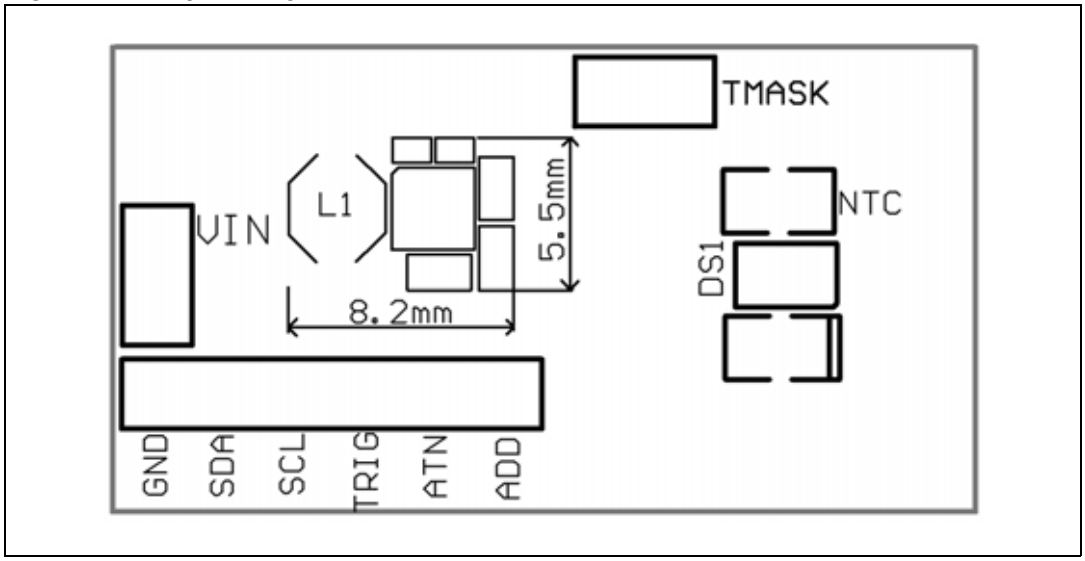


Figure 7. Top overlay



3.2.2 A two-layer PCB with application area 72.4 mm^2 for QFN package

Figure 8. Top layer

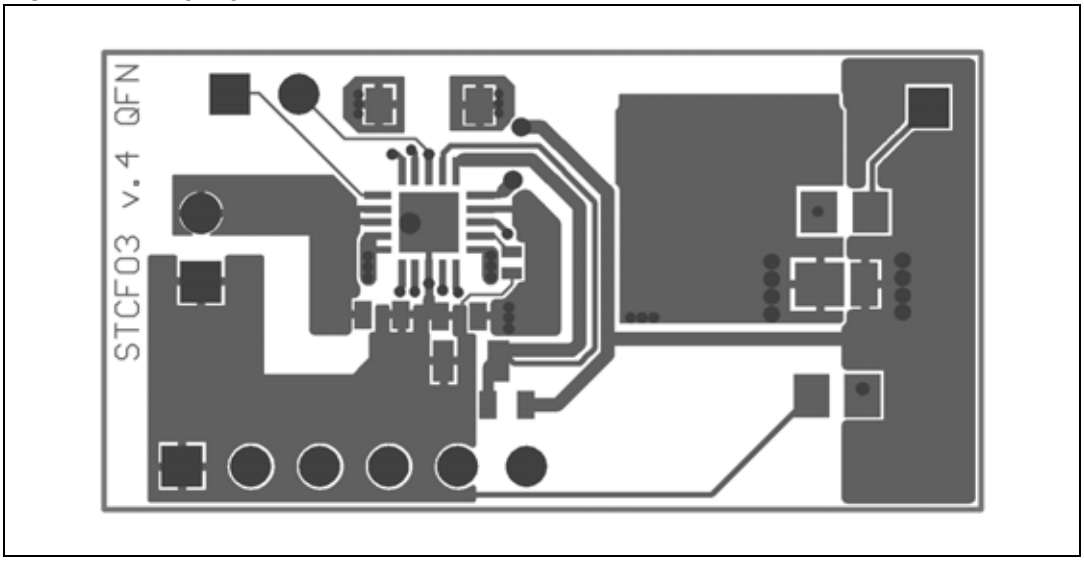


Figure 9. Bottom layer

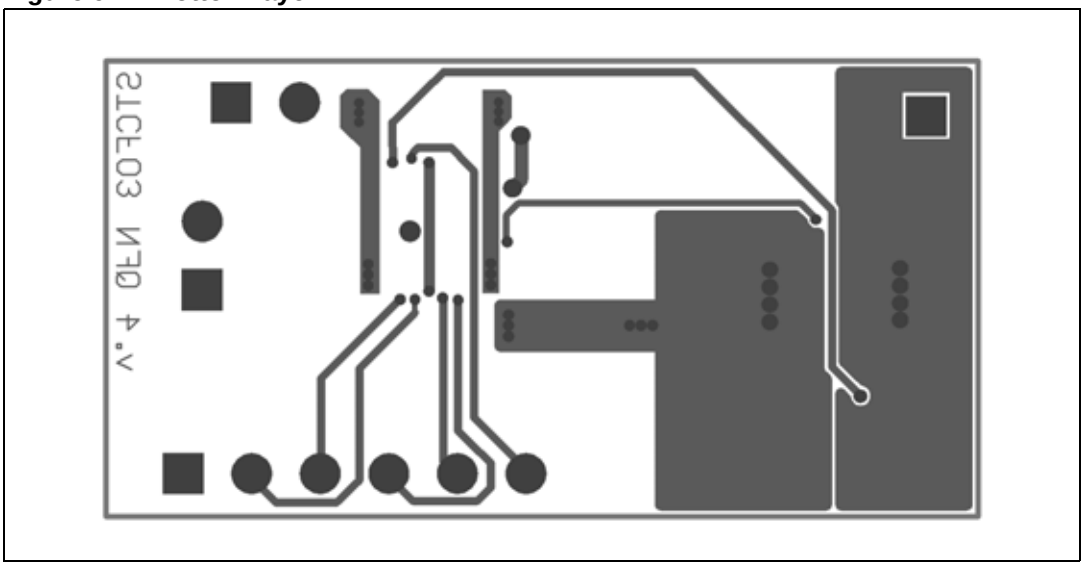
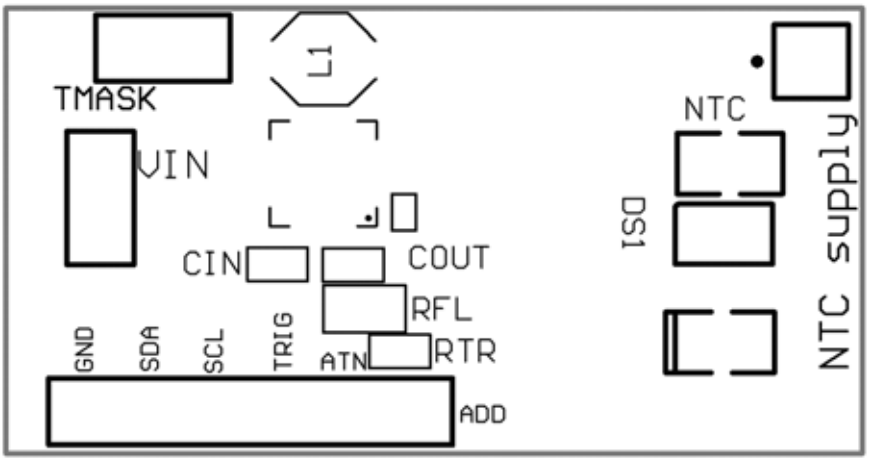


Figure 10. Top overlay



3.2.3 A four-layer PCB with application area 45.1 mm^2 for BGA package, version C

Figure 11. Top layer

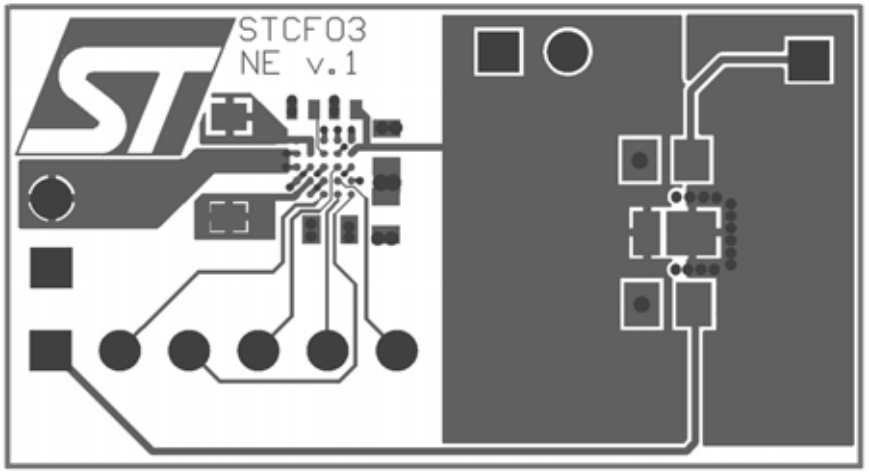


Figure 12. Middle layer 1

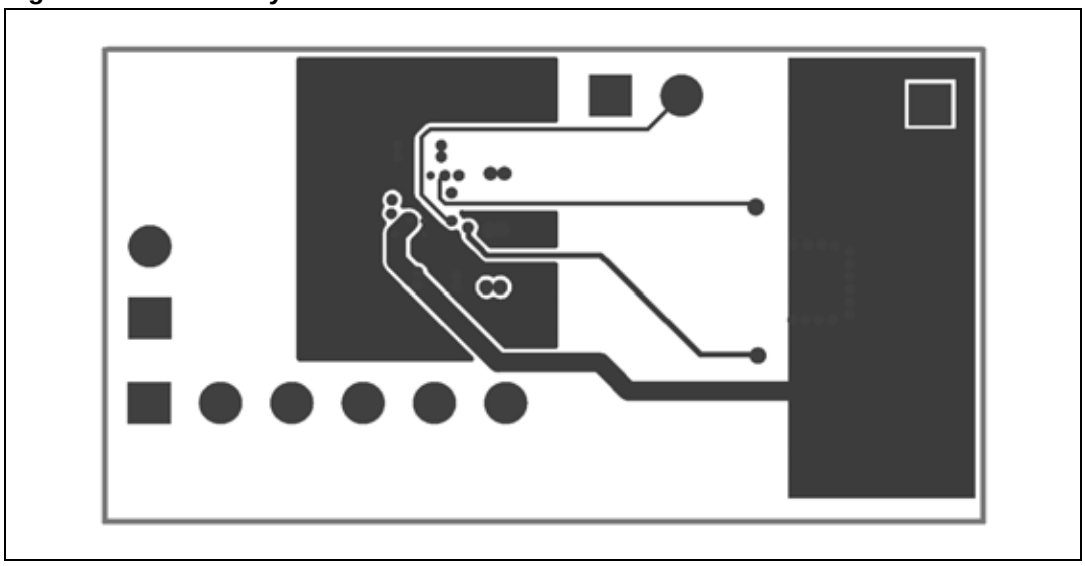


Figure 13. Middle layer 2

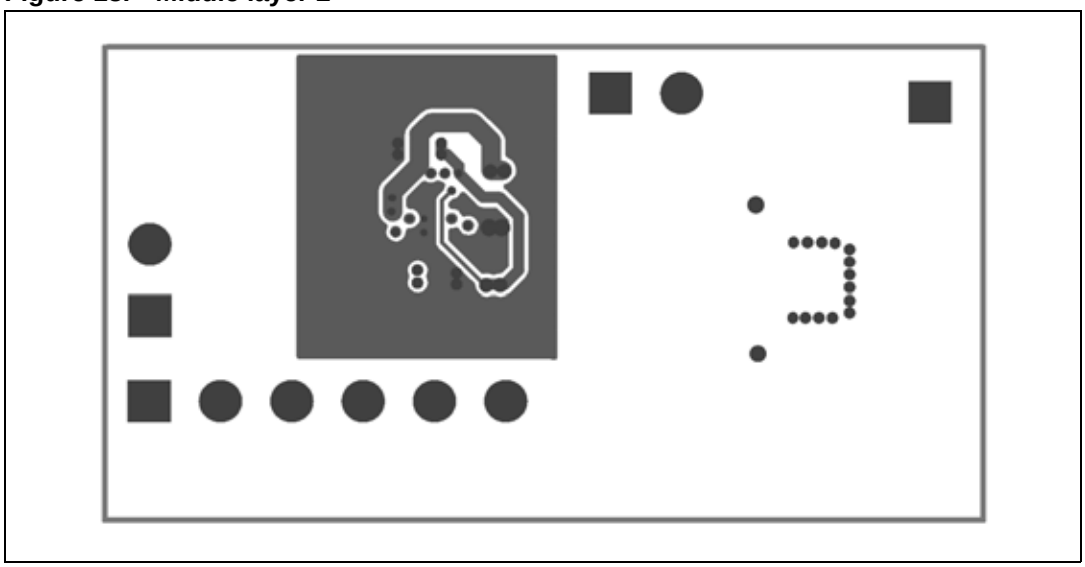


Figure 14. Bottom layer

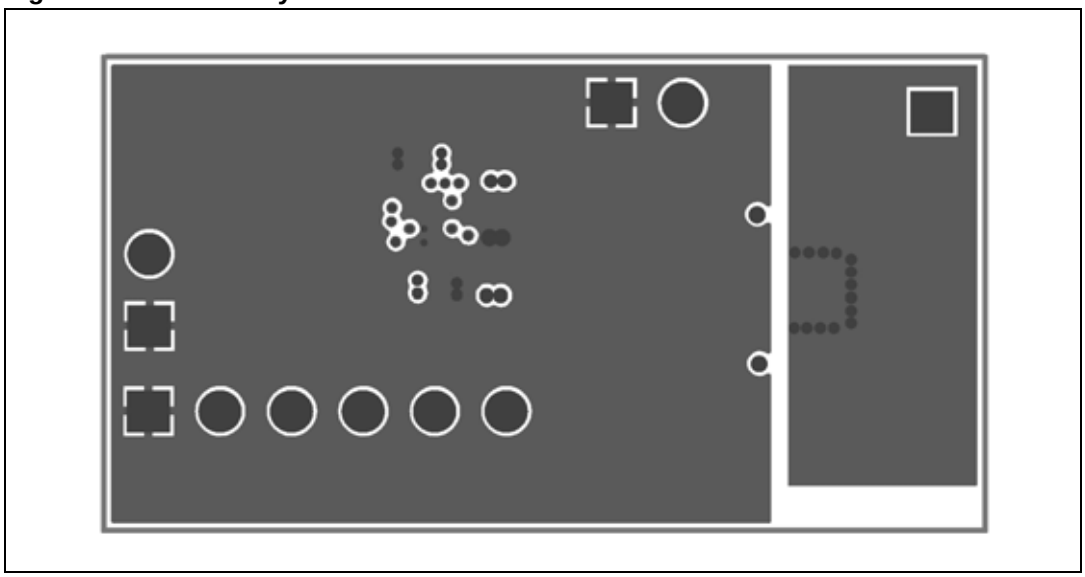
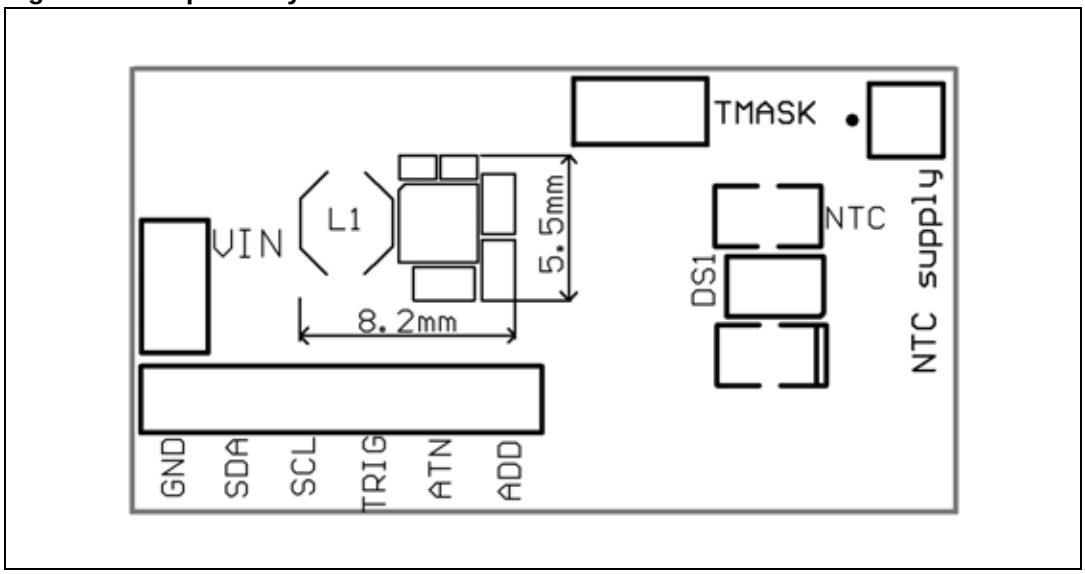


Figure 15. Top overlay



4 Internal registers

4.1 Accessing the internal registers

There are 4 internal registers in STCF03 (which are the COMMAND, DIMMING, AUX_LED and STATUS registers). The STATUS register is read-only. The COMMAND register can be accessed in any operation mode. All the other registers can be accessed in any mode, except in SHUTDOWN mode. When the device enters SHUTDOWN mode, the DIMMING, AUX_LED and STATUS registers are cleared. The COMMAND register value remains untouched when entering SHUTDOWN mode. *Table 2* shows the accessibility of each register in all operation modes.

Table 2. Accessibility of internal registers

Register	Address	Mode				SHUTDOWN value	PowerON reset value
		SHUTDOWN	READY	TORCH	FLASH		
COMMAND	00	Read / Write	Read / write	Read / write	Read / write	Untouched	Cleared
DIMMING	01	Inaccessible	Read / write	Read / write	Read / write	Cleared	Cleared
AUX_LED	02	Inaccessible	Read / write	Read / write	Read / write	Cleared	Cleared
STATUS	03	Inaccessible	Read only	Read only	Read only	Cleared	Cleared

5 Operation modes

5.1 SHUTDOWN mode

SHUTDOWN mode is entered after the Power-ON reset. This mode is mainly used to decrease the power consumption of the device. During this mode, only the I^2C interface is alive. The only thing which can be done in SHUTDOWN mode is to access the COMMAND register. Entering SHUTDOWN mode by writing to the COMMAND register aborts any running operation and clears the values of the DIMMING, AUX_LED and STATUS registers. The COMMAND register value is not affected by entering SHUTDOWN mode.

The following data must be written to the COMMAND register to enter SHUTDOWN mode.

Table 3. COMMAND register data to enter SHUTDOWN mode (version B)

	PWR_ON	TRIG_EN	TCH_ON	NTC_ON	FTIM_3	FTIM_2	FTIM_1	FTIM_0
CMD_REG	0	x	x	x	x	x	x	x
	MSB							LSB

Table 4. COMMAND register data to enter SHUTDOWN mode (version A and C)

	PWR_ON	TRIG_EN	TCH_ON	NTC_ON	FTIM_3	FTIM_2	FTIM_1	FTIM_0
CMD_REG	0	x	x	0	x	x	x	x
	MSB							LSB

5.2 SHUTDOWN mode with the NTC-feature activated

This mode is supported only in version A, which does not have any internal voltage reference for the NTC feature. When this operation mode is activated, the microcontroller can still monitor the NTC voltage through its A/D converter, while STCF03 remains in SHUTDOWN mode and therefore saves power.

The following data must be written to the COMMAND register to enter SHUTDOWN mode + NTC.

Table 5. COMMAND register data to enter SHUTDOWN mode with NTC activated (version A and C)

	PWR_ON	TRIG_EN	TCH_ON	NTC_ON	FTIM_3	FTIM_2	FTIM_1	FTIM_0
CMD_REG	0	x	x	1	x	x	x	x
	MSB							LSB

5.3 READY mode and NTC

The READY mode allows the user to access all the internal registers. The NTC feature can be activated in this mode and the temperature of the LED can be sensed by the A/D converter of the microcontroller. The following data must be written to the COMMAND register to enter READY mode.

Table 6. COMMAND register data to enter READY mode

	PWR_ON	TRIG_EN	TCH_ON	NTC_ON	FTIM_3	FTIM_2	FTIM_1	FTIM_0
CMD_REG	1	0	0	0	x	x	x	x
	MSB							LSB

The following data must be written to the COMMAND register to activate the NTC feature.

Table 7. COMMAND register data to enter READY mode with NTC ON

	PWR_ON	TRIG_EN	TCH_ON	NTC_ON	FTIM_3	FTIM_2	FTIM_1	FTIM_0
CMD_REG	1	0	0	1	x	x	x	x
	MSB							LSB

As soon as the NTC feature is activated, the internal switch connects the NTC resistor to the R_X resistor, thereby creating a voltage divider. The voltage on this divider can be, if desired, monitored by the A/D converter of the microcontroller. An external voltage reference must be connected to the NTC to use this feature (only in version A and C). The bits NTC_W and NTC_H of the STATUS register will not be properly set if there is no external reference voltage connected to the NTC (only in version A and C).

If the NTC feature is not going to be used, neither the negative thermistor, nor the external reference needs to be connected. In this case, it is recommended to ground the RX pin. As the NTC feature is automatically activated during the FLASH and TORCH mode, leaving the RX pin floating could lead to unwanted interruptions of the light due to non-defined voltages on the RX pin

5.4 TORCH mode

This mode is intended to be used for low light intensities. The LED current in the TORCH mode can be adjusted in a range from 15 mA up to 200 mA.

The TORCH mode is activated by writing the following data to the COMMAND register.

Table 8. COMMAND register data to enter TORCH mode

	PWR_ON	TRIG_EN	TCH_ON	NTC_ON	FTIM_3	FTIM_2	FTIM_1	FTIM_0
CMD_REG	1	0	1	x	x	x	x	x
	MSB							LSB

The DIMMING register value (TDIM) must be set as well, unless it has already been set during a previous operation. If TDIM register is not set, then the default output current value will be at the minimum.

There is no internal timer which controls the TORCH duration. Therefore, as soon as the TORCH mode is activated, it remains active until a new mode is entered by writing a new data to the COMMAND register.

If the TORCH mode was terminated by entering READY or FLASH mode, it can be restarted again by writing the corresponding data to the COMMAND register only, because entering any of the READY and FLASH modes does not influence the TDIM value. If the TORCH mode was terminated by entering into SHUTDOWN mode, then the TDIM value must be set again during the restart of the TORCH, because entering the SHUTDOWN mode clears the TDIM value.

As soon as the TORCH mode is activated, the NTC feature is automatically activated too in order to protect the LED against overheating. The NTC feature will be activated even if the NTC_ON bit in the COMMAND register is set to zero.

5.5 FLASH mode

This mode is intended to be used for high light intensities. The LED current in the FLASH mode can be adjusted up to 800 mA with the input voltage ranging from 3.3 V up to 5.5 V.

The FLASH mode is activated by writing the following data to the COMMAND register.

Table 9. COMMAND register data to enter FLASH mode

	PWR_ON	TRIG_EN	TCH_ON	NTC_ON	FTIM_3	FTIM_2	FTIM_1	FTIM_0
CMD_REG	1	1	x	x	x	x	x	x
	MSB							LSB

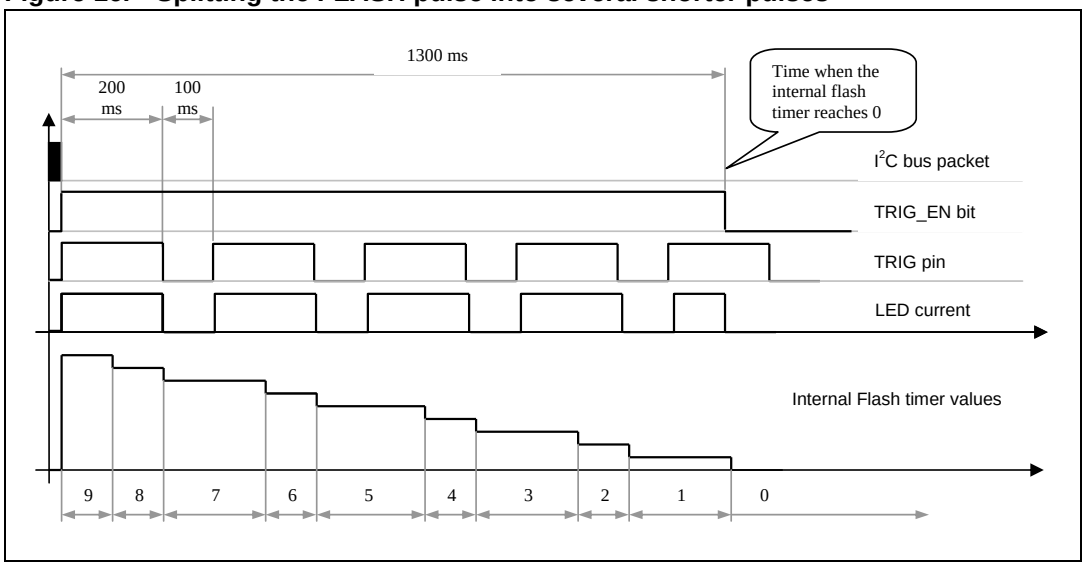
The DIMMING register value (FDIM) must be set as well, unless it has already been set during a previous operation.

The activation of the FLASH mode requires the TRIG pin to be High. The FLASH mode is active only when the TRIG_EN bit in the COMMAND register is set to 1 and the TRIG pin is High. This gives the user the possibility to choose between a soft and a hard triggering of the FLASH mode.

The soft triggering is done by writing data to the internal registers only, while the TRIG pin is permanently kept High, that is, by connecting it to VBAT. This saves one pin of the microcontroller, which can be used for a different purpose, but this way of triggering is less accurate than the hard one. The second disadvantage of this solution is that the FLASH duration can only be set in discrete steps of the internal timer (1 step = approx. 100 ms).

Hard-triggering of the FLASH mode requires the microcontroller to manage the TRIG pin. The COMMAND and the DIMMING registers are loaded with data before the TRIG pin is set to High. This allows the user to avoid the I^2C-bus latency. FLASH mode then starts as soon as the TRIG pin is set to High. It takes typically about 0.7 ms to ramp-up the LED current to the adjusted value. This time may vary according to the LED current value and the battery voltage. When the TRIG pin is kept High long enough, the internal timer reaches zero and the FLASH mode is over. As soon as the FLASH is timed out, the ATN pin is pulled down for 11µs to inform the microcontroller that the STATUS register was updated and that the flash is over. If the TRIG pin is set to Low before the internal timer reaches zero, the FLASH mode will be interrupted and can be restarted by setting the TRIG pin to High again. The internal timer is stopped while the TRIG pin is Low. This means that the user can split the FLASH into several pulses of a total length equal to the FTIM value. *Figure 16* below shows splitting of the FLASH into several shorter pulses. The cumulative length of all the pulses is determined by the FTIM value. *Figure 16* shows the case for FTIM = 9 (900 ms FLASH time). The cumulative time when the TRIG pin is High is 1000 ms (5 pulses 200 ms long). The last FLASH pulse will be100ms long only. The reason is that the internal FLASH timer reaches zero and the TRIG_EN bit is set to 0.

Figure 16. Splitting the FLASH pulse into several shorter pulses



Hard triggering allows therefore a smooth setting of the FLASH duration. The resolution is about 8.8µs. The minimum FLASH duration is limited by the ramp-up time of the LED current and the maximum is limited by the FTIM value. If it is necessary to make a FLASH pulse longer than the maximum allowed by FTIM, then it is necessary to reload the COMMAND register before the internal timer reaches zero (start a new FLASH before the previous one elapses). See *Section 8.5* - Example 5 for more details.

6 The STATUS register and the ATN pin

6.1 The STATUS register

Table 10. STATUS register bits

Bit name	N/A	F_RUN	LED_F	NTC_W	NTC_H	OT_F	N/A	VOUTOK_N
	MSB							LSB

A detailed description of each bit is stated in the datasheet.

Table 11. Effect of the STATUS register bits on the operation of the device

Bit name	F_RUN (STAT_REG)	LED_F (STAT_REG)	NTC_W (STAT_REG)	NTC_H (STAT_REG)	OT_F (STAT_REG)	VOUTOK_N (STAT_REG)
Default value	0	0	0	0	0	0
Latched (1)	NO	YES	YES	YES	YES	YES
Forces READY mode when set	NO	YES	NO	YES	YES	YES
Sets ATN LOW when set	NO	YES	YES	YES	YES	YES

1. YES means that the bit is set by internal signals and is reset to its default value by an I^2C-read operation of STAT_REG; NO means that the bit is set and reset by internal signals in real-time

When the status register is latched, reading and writing to the registers is still possible, but the bits TRIG_EN and TCH_ON in the COMMAND register and AUXL register cannot be changed, until the device is unlatched. It is necessary to read the STATUS register to unlatch the device.

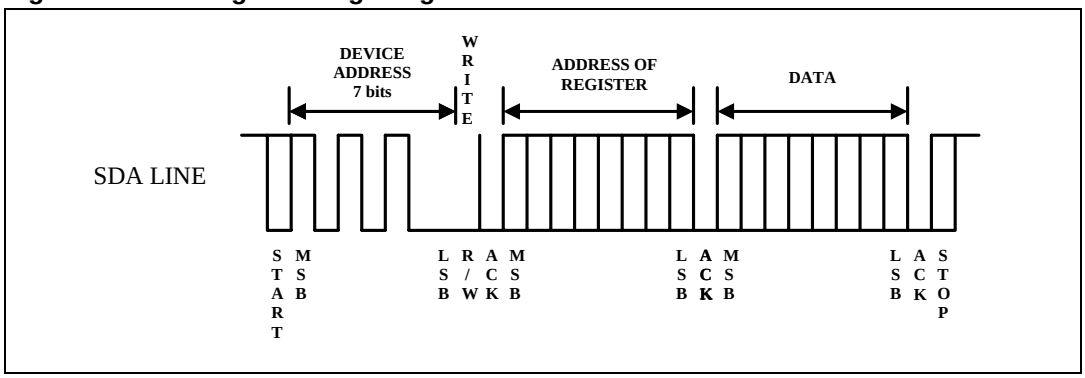
The ATN pin is also pulled down when the internal timer reaches zero in FLASH mode. In this case the ATN pin is pulled down for 11 µs only. It is recommended to connect the ATN pin to the interrupt input of the microcontroller. If it is not connected to the interrupt input, the ATN pin should be pulled fast enough not to miss the 11 µs pulse, that is, by a programming loop which is entered after start of the FLASH mode. This loop runs until the ATN pin gets Low. It is recommended to make a time-out of such a loop.

7 Reading and writing to the STCF03 registers through the I²C bus

7.1 Writing to a single register

Writing to a single register starts with a START-bit followed by the 7-bit device address of STCF03. The 8-th bit is the R/W bit, which is 0 in this case. R/W = 1 means a Reading operation. Then the master awaits an acknowledgement from STCF03. The 8-bit address of the desired register is sent afterwards to STCF03. It will also be followed by an acknowledge pulse. The last transmitted byte is the data that is going to be written into the register. It is followed again by an acknowledge pulse from STCF03. Then the master generates a STOP-bit and the communication is over. See *Figure 17* below.

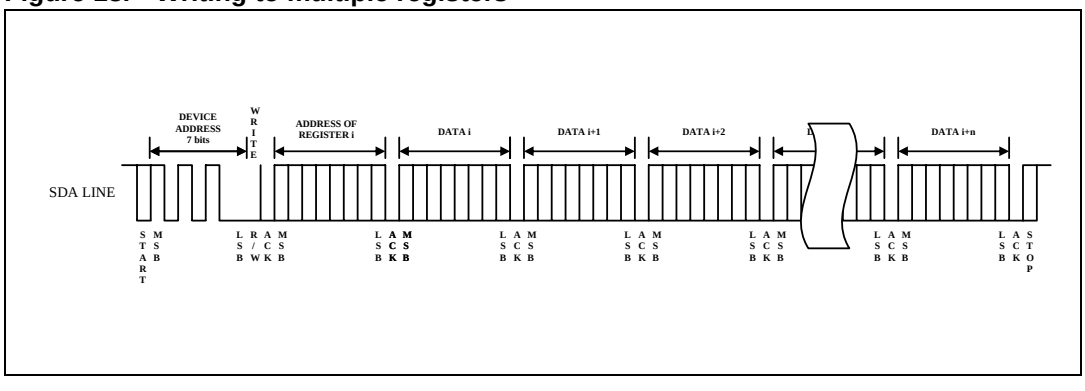
Figure 17. Writing to a single register



7.2 Writing to multiple registers with incremental addressing

It would be unpractical to send several times the device address and the address of the register when writing to multiple registers. STCF03 supports writing to multiple registers with incremental addressing. When data is written to a register, the register address is automatically incremented (by one), and therefore the next data can be sent without sending again the device address and the register address. See *Figure 18* below.

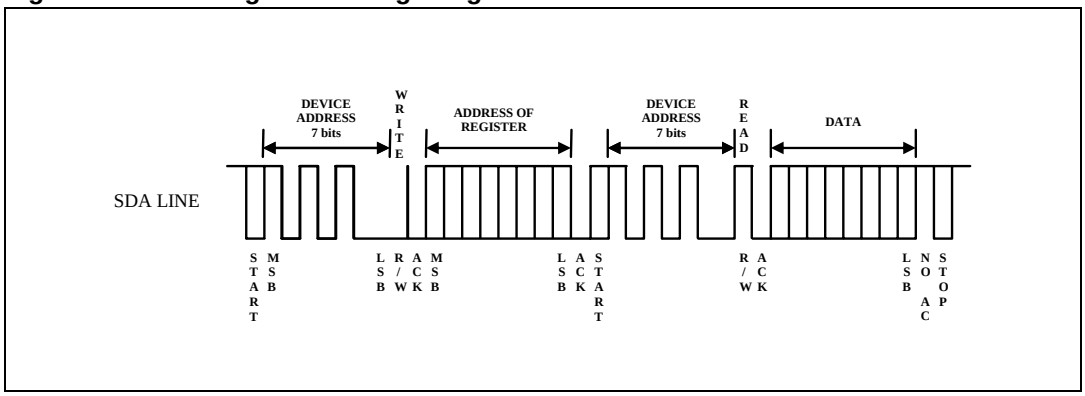
Figure 18. Writing to multiple registers



7.3 Reading from a single register

The reading operation starts with a START-bit followed by 7 bit device address of STCF03. The 8-th bit is the R/W bit, which is 0 in this case. STCF03 confirms the receiving of the address + R/W bit by an acknowledge pulse. The address of the register which should be read is sent after and confirmed by an acknowledge pulse from STCF03 again. Then the master generates a START-bit again and sends the device address followed by the R/W-bit, which is 1 now. STCF03 confirms the receiving of the address + R/W-bit by an acknowledge pulse, and starts to send data to the master. No acknowledge pulse from the master is required after receiving the data. Then the master generates a STOP-bit to terminate the communication. See the *Figure 19* below.

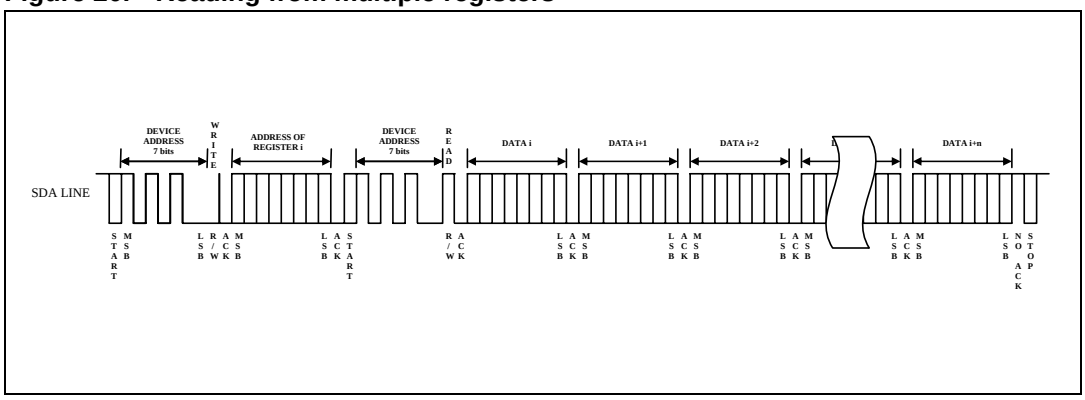
Figure 19. Reading from a single register



7.4 Reading from multiple registers with incremental addressing

Reading from multiple registers starts in the same way as reading from a single register. As soon as the first register is read, the register address is automatically incremented. If the master generates an acknowledge pulse after receiving the data from the first register, then reading from the next register can start immediately without having to send once more the device and the register addresses. The last acknowledge pulse before the STOP-bit is not required. See *Figure 20* below.

Figure 20. Reading from multiple registers



8 Examples of register setup for each mode

A device address 0x62 is used in all the example below. The STCF03 is configured to this device address, if the ADD pin is connected to VBAT pin. In the demoboard the device address is 0x60 because the ADD pin is connected to GND.

Table 12. TORCH mode and FLASH mode dimming registers settings

T_DIM (hex)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F								
F_DIM (hex)									0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
LED current [mA]	16	19	23	27	32	39	46	55	65	77	92	109	124	147	175	209	248	296	352	418	498	592	705	840
Internal step	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
V_{REF1} [mV]	33	40	47	56	67	80	95	113	134	160	190	227	33	40	47	56	67	79	95	113	134	160	190	227
Sense Resist.	R_{FL} + R_{TR}	R_{FL} + R_{TR}	R_{FL} + R_{TR}	R_{FL} + R_{TR}	R_{FL} + R_{TR}	R_{FL} + R_{TR}	R_{FL} + R_{TR}	R_{FL} + R_{TR}	R_{FL} + R_{TR}	R_{FL} + R_{TR}	R_{FL} + R_{TR}	R_{FL} + R_{TR}	R_{FL}	R_{FL}	R_{FL}	R_{FL}	R_{FL}	R_{FL}	R_{FL}	R_{FL}	R_{FL}	R_{FL}	R_{FL}	R_{FL}

Note: LED current values refer to R_{FL}= 0.27 Ohm, R_{TR} = 1.8 Ohm

8.1 Example 1: 600 mA FLASH with 700 ms duration

Let's suppose that R_{FL} = 0.27 Ω The targeted value of the FLASH current is 600 mA and the FLASH duration should be 700 ms.

The reference voltage must be set to 160 mV to achieve a 600 mA FLASH current with a 0.27 Ω sensing resistor. The value of FDIM (4 bits) must be set to 0xD to set up the reference voltage to 160 mV. (See *Table 12*)

The FLASH duration timer can be set to 100ms up to 1500 ms in 100ms increments. If the desired FLASH duration is 700 ms the value FTIM (4 bits) must be set to 0x7.

Bit PWR_ON of the command register must be set to 1.

Bit TRIG_EN of the command register must be set to 1.

Bit TCH_ON of the command register must be set to 0.

Bit NTC_ON of the command register can be set to any value, because NTC is automatically ON when the FLASH mode is active. Setting this bit to 0 will not switch off the NTC.

Table 13. COMMAND register data to enter FLASH mode

	PWR_ON	TRIG_EN	TCH_ON	NTC_ON	FTIM_3	FTIM_2	FTIM_1	FTIM_0
CMD_REG	1	1	x	x	0	1	1	1
	MSB							LSB

Table 14. DIMMING register data for the FLASH mode

DIM_REG	**TDIM_3**	**TDIM_2**	**TDIM_1**	**TDIM_0**	**FDIM_3**	**FDIM_2**	**FDIM_1**	**FTIM_0**
	0	0	0	0	1	1	0	1
	MSB							LSB

It is necessary to write 4 bytes to STCF03 to make a FLASH.

Table 15. I^2C data packet for activating the FLASH mode

Byte	Hex	Binary								Comment
1	62	0	1	1	0	0	0	1	0	Device address + R/W bit
2	00	0	0	0	0	0	0	0	0	Command register address
3	D7	1	1	0	1	0	1	1	1	Data of the command register
4	0D	0	0	0	0	1	1	0	1	Data of the dimming register

8.2 Example 2: 25 mA TORCH

Let's suppose that R_{FL} = 0.27 Ω, R_{TR} = 1.8 Ω and the targeted value of the TORCH current is 25 mA.

The reference voltage must be set to 56 mV to achieve 25 mA in TORCH mode with the resistor values mentioned above. The value of TDIM (4 bits) must be set to 0x3 to set up the reference voltage to 56 mV.

Bit PWR_ON of the command register must be set to 1.

Bit TRIG_EN of the command register must be set to 0.

Bit TCH_ON of the command register must be set to 0.

Bit NTC_ON of the command register can be set to any value, because NTC is automatically ON, when TORCH mode is active. Setting this bit to 0 does not switch off the NTC.

Table 16. COMMAND register data for the TORCH mode

CMD_REG	**PWR_ON**	**TRIG_EN**	**TCH_ON**	**NTC_ON**	**FTIM_3**	**FTIM_2**	**FTIM_1**	**FTIM_0**
	1	0	1	1	0	0	0	0
	MSB							LSB

Table 17. DIMMING register data for the TORCH mode

DIM_REG	**TDIM_3**	**TDIM_2**	**TDIM_1**	**TDIM_0**	**FDIM_3**	**FDIM_2**	**FDIM_1**	**FDIM_0**
	0	0	1	1	0	0	0	0
	MSB							LSB

It is necessary to write 4 bytes to the STCF03 to run the TORCH mode.

Table 18. I²C data packet to activate TORCH mode

Byte	Hex	Binary								Comment
1	62	0	1	1	0	0	0	1	0	Device address + R/W bit
2	00	0	0	0	0	0	0	0	0	Command register address
3	B0	1	0	1	1	0	0	0	0	Data of the command register
4	30	0	0	1	1	0	0	0	0	Data of the dimming register

The duration of the TORCH mode is "unlimited". TORCH mode is terminated by setting the TCH_ON bit in the COMMAND register to 0.

Termination of the TORCH mode can be done by writing the following data to STCF03.

Table 19. I²C data packet for terminating the TORCH mode

Byte	Hex	Binary								Comment
1	62	0	1	1	0	0	0	1	0	Device address + R/W bit
2	00	0	0	0	0	0	0	0	0	Command register address
3	80	1	0	0	0	0	0	0	0	Data of the command register

This puts the STCF03 into READY mode.

8.3 Example 3: an Auxiliary LED running at 10 mA for 500 ms

STCF03 must be into READY mode (both bits TRIG_EN and TCH_ON are 0) to activate the Auxiliary LED.

A 10 mA output current is reached when AUXI is set to 0x8.

AUXT must be set to 0x5 to have a 500 ms duration of the Auxiliary LED lighting.

Table 20. COMMAND register data for the AUX_LED

	PWR_ON	TRIG_EN	TCH_ON	NTC_ON	FTIM_3	FTIM_2	FTIM_1	FTIM_0
CMD_REG	1	0	0	0	0	0	0	0
	MSB							LSB

Table 21. COMMAND register data for the AUX_LED

	AUX_3	AUX_2	AUX_1	AUX_0	AUX_3	AUX_2	AUX_1	AUX_0
AUX_LED	1	0	0	0	0	1	0	1
	MSB							LSB

Writing the 3 bytes below to STCF03 puts it into READY mode. This can be skipped if it already is in READY mode.

Table 22. I²C data packet for activating the READY mode

Byte	Hex	Binary								Comment
1	62	0	1	1	0	0	0	1	0	Device address + R/W bit
2	00	0	0	0	0	0	0	0	0	Command register address
3	80	1	0	0	0	0	0	0	0	Data of the command register

Writing the following 3 bytes to STCF03 will activate the Auxiliary LED for the desired time.

Table 23. I²C data packet for activating the AUX_LED

Byte	Hex	Binary								Comment
1	62	0	1	1	0	0	0	1	0	Device address + R/W bit
2	02	0	0	0	0	0	0	1	0	Auxiliary LED register address
3	85	1	0	0	0	0	1	0	1	Data of the auxiliary LED register

8.4 Example 4: Red-eye reduction (multiple short flashes)

There are two ways to manage this task. The first one is to use hardware triggering of the flashes through the TRIG pin. This is the most suitable and recommended solution, as it reduces the usage of the I²C bus and the length of each FLASH pulse can be adjusted continuously. The second solution is to use the software triggering feature, which means a periodical reloading of the COMMAND register. This however increases traffic on the I²C bus and the flashes can only have length, adjustable in 100 ms increments only.

Let's suppose that R_{FL} = 0.27 Ω and the targeted value of the FLASH current is 600 mA. The task is to make 5 flashes of 200 ms duration with 100 ms pause between them. The setting of the reference voltage is identical to the one in *Section 8.1*. The FLASH timer (FTIM) is set to 0xF, which represents 1.5 s.

Table 24. COMMAND register data for FLASH mode

	PWR_ON	TRIG_EN	TCH_ON	NTC_ON	FTIM_3	FTIM_2	FTIM_1	FTIM_0
CMD_REG	1	1	0	1	1	1	1	1
	MSB							LSB

Table 25. DIMMING register data for the FLASH mode

	TDIM_3	TDIM_2	TDIM_1	TDIM_0	FDIM_3	FDIM_2	FDIM_1	FDIM_0
DIM_REG	0	0	0	0	1	1	0	1
	MSB							LSB

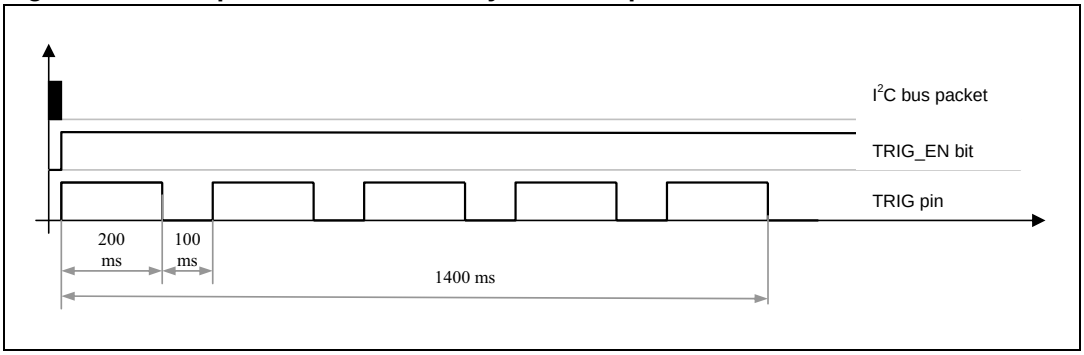
The data packet which has to be sent is in the table below.

Table 26. I²C data packet for activating the FLASH mode

Byte	Hex	Binary								Comment
1	62	0	1	1	0	0	0	1	0	Device address + R/W bit
2	00	0	0	0	0	0	0	0	0	Command register address
3	DF	1	1	0	1	1	1	1	1	Data of the command register
4	0D	0	0	0	0	1	1	0	1	Data of the dimming register

The picture below shows the TRIG pin and the I²C bus timings.

Figure 21. Multiple flashes handled by the TRIG pin



8.5 Example 5: A FLASH pulse longer than 1.5 s

Let's suppose that R_{FL} = 0.27 Ω and the targeted value of the FLASH current is 600 mA. The task is to make a single FLASH pulse with a 4 seconds duration.

It is necessary to reload FTIM in the COMMAND REGISTER before the internal FLASH timer reaches zero. This guarantees that the FLASH does go on and does not stop after 1.5 sec.

The first packet must contain also the DIMMING REGISTER data, if they are different from those which were used in the previous operation.

- Packet 1

Sets FLASH mode with 1.5 s duration and the proper dimming.

Table 27. I²C data packet for activating the FLASH mode

Byte	Hex	Binary								Comment
1	62	0	1	1	0	0	0	1	0	Device address + R/W bit
2	00	0	0	0	0	0	0	0	0	Command register address
3	DF	1	1	0	1	1	1	1	1	Data of the command register
4	0D	0	0	0	0	1	1	0	1	Data of the dimming register

- Packet 2

Sets FLASH mode with 1.5 s duration. Dimming is not set again as it is same as before.

Table 28. 1st I2C data packet to restart the FLASH mode

Byte	Hex	Binary								Comment
1	62	0	1	1	0	0	0	1	0	Device address + R/W bit
2	00	0	0	0	0	0	0	0	0	Command register address
3	DF	1	1	0	1	1	1	1	1	Data of the command register

- Packet 3

Sets FLASH mode with 1.5 s duration. Dimming remains untouched.

Table 29. 2nd I2C data packet for restart of the FLASH mode

Byte	Hex	Binary								Comment
1	62	0	1	1	0	0	0	1	0	Device address + R/W bit
2	00	0	0	0	0	0	0	0	0	Command register address
3	DF	1	1	0	1	1	1	1	1	Data of the command register

- Packet 4

Sets FLASH mode with 1 s duration. Dimming remains untouched.

Table 30. 3rd I2C data packet to restart the FLASH mode

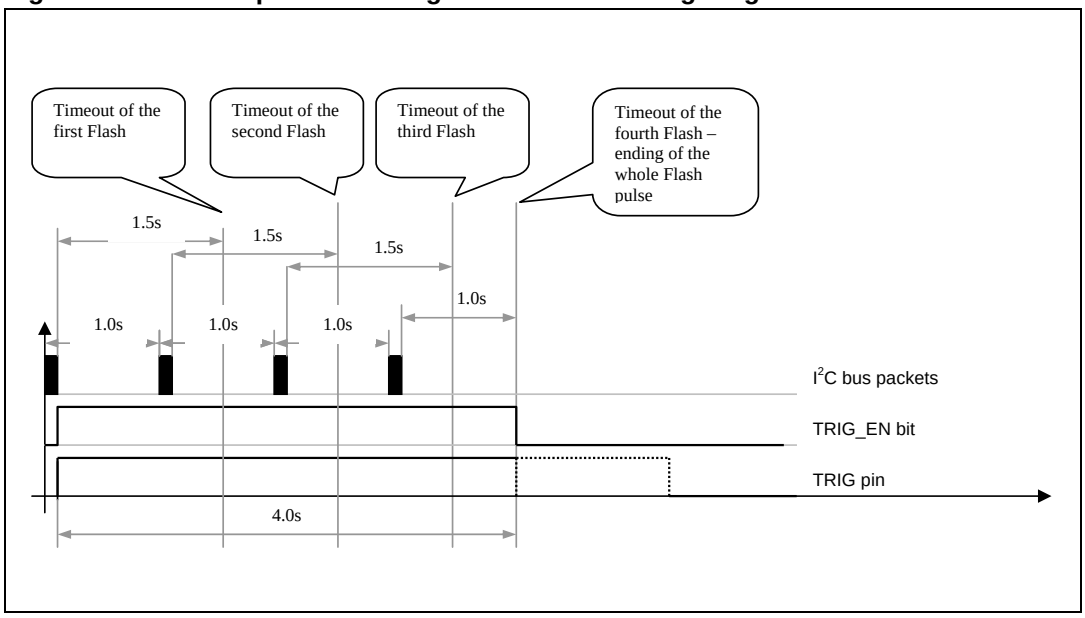
Byte	Hex	Binary								Comment
1	62	0	1	1	0	0	0	1	0	Device address + R/W bit
2	00	0	0	0	0	0	0	0	0	Command register address
3	DA	1	1	0	1	1	0	1	0	Data of the command register

Please refer to *Figure 22* for more details about the I2C-bus packets timing.

The solution described above is using a software termination of the FLASH pulse. (It is timed out by the internal timer.) The FLASH pulse could be also terminated by setting the TRIG pin to low after 4 seconds. In this case, the fourth packet could be the same as packets *Packet 2* and *Packet 3*, because the timing of the FLASH is done by the TRIG pin and it is not necessary to change the value of FTIM in the COMMAND REGISTER.

This way of periodical reloading of the COMMAND REGISTER can be used to achieve a continuous FLASH light. In this case, it is very strongly recommended to guarantee an efficient cooling of both the LED and the chip, otherwise the light can be interrupted by activation of the thermal protections.

Figure 22. I²C bus packets timing for a FLASH lasting longer than FTIM max



9 Revision history

Table 31. Revision history

Date	Revision	Changes
19-Apr-2007	1	Initial release

Please Read Carefully:

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST’s terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN ST’S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS EXPRESSLY APPROVED IN WRITING BY AN AUTHORIZED ST REPRESENTATIVE, ST PRODUCTS ARE NOT RECOMMENDED, AUTHORIZED OR WARRANTED FOR USE IN MILITARY, AIR CRAFT, SPACE, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS WHERE FAILURE OR MALFUNCTION MAY RESULT IN PERSONAL INJURY, DEATH, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. ST PRODUCTS WHICH ARE NOT SPECIFIED AS "AUTOMOTIVE GRADE" MAY ONLY BE USED IN AUTOMOTIVE APPLICATIONS AT USER’S OWN RISK.

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.

Information in this document supersedes and replaces all information previously supplied.

The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

© 2007 STMicroelectronics - All rights reserved

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com