UM1993
User manual

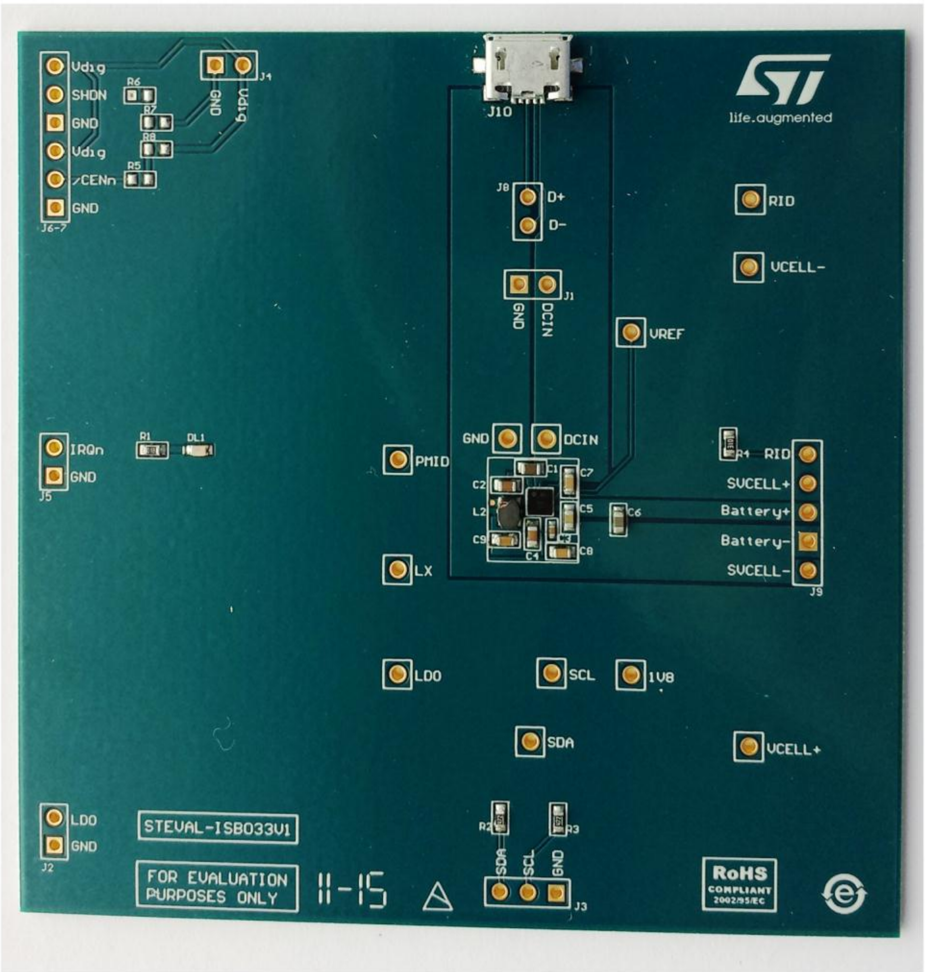
Switch mode single cell Li+ battery charger with OTG boost, voltage mode fuel gauge and LDO

Introduction

This user manual is for the STEVAL-ISB033V1 evaluation board based on the STBCFG01 high efficiency switching battery charger.

The objective of this document is to demonstrate the functionality of the STEVAL-ISB033V1 and how to use it.

Figure 1: STEVAL-ISB033V1 evaluation board



Contents

1 System overview 3

2 Schematic diagram 4

3 Input / Output connections 6

4 Connections 7

5 GUI description 8

5.1 How to start 8

5.1.1 Installation 8

5.2 General description 10

5.2.1 USB interface module 11

5.2.2 TAB overview 11

6 Application information 19

6.1 Operation description 19

6.1.1 Battery charger 19

6.1.2 Input current limit 20

6.1.3 Dynamic input current limit (DICL) 20

6.1.4 Automatic recharge 21

6.1.5 Battery detection 21

6.1.6 Battery overvoltage protection 21

7 Board layout 23

8 Revision history 25

1 System overview

The STEVAL-ISB033V1 is a product evaluation board based on the STBCFG01 high efficiency switching battery charger, integrating the necessary functions to charge single cell Li-Ion batteries, monitor the battery charge and generate 5.0 V to supply USB OTG bus powered devices.

The device also integrates an LDO regulator to support system boot under dead battery conditions.

The battery charger features a smart input current limit, whereby the maximum input current can be selected via I²C and, if the input voltage drops below a programmable threshold, the input current is reduced even if the selected maximum current limit has not been reached. The dynamic input current limit can be disabled.

An automatic input pre-bias load renders the device suitable for applications using voltage sources requiring a minimum external load for proper regulation.

The STBCFG01 also integrates a voltage mode fuel gauge to provide accurate charge status information without the current sensing resistor.

2 Schematic diagram

Figure 2: STEVAL-ISB033V1 circuit schematic

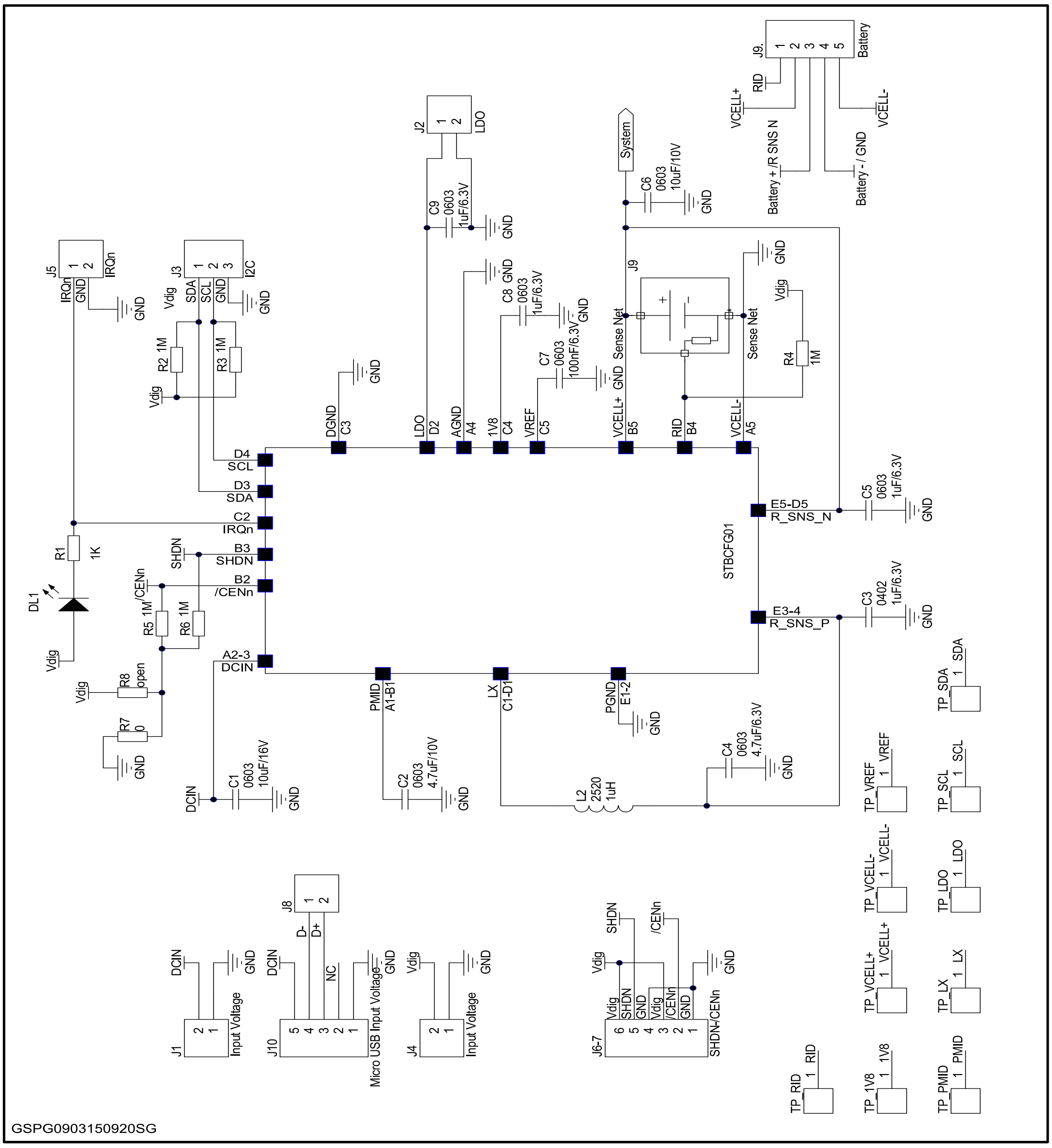


Table 1: List of external components

Component	Size	Value	Supplier	Part number
C_1	0603	10 µF, 16 V	MURATA	GRM188R61C106MA73D
C_2	0603	4.7 µF, 25 V	TDK	C1608X5R1E475K080AC
C_3	0402	1 µF, 16 V	MURATA	GRM155R61C105KA12D
C_4	0603	4.7 µF, 25 V	TDK	C1608X5R1E475K080AC
C_5	0603	1 µF, 16 V	MURATA	GRM185R61C105KE44
C_6	0603	10 µF, 16 V	MURATA	GRM188R61C106MA73D
C_7	0603	100 nF, 25 V	MURATA	GRM188R71H104KA93
C_8	0603	1 µF, 16 V	MURATA	GRM185R61C105KE44

Component	**Size**	**Value**	**Supplier**	**Part number**
C_9	0603	1 µF, 16 V	MURATA	GRM185R61C105KE44
L_2	2.5*2 mm	1 µH, 2.2 A	TDK	VLS252012ET-1R0N
R_1	0603	1 kΩ		
R_2	0603	1 MΩ		
R_3	0603	1 MΩ		
R_4	0603	1 MΩ		
R_5	0603	1 MΩ		
R_6	0603	1 MΩ		
R_7	0603	0 Ω		
R_8	0603	OPEN		

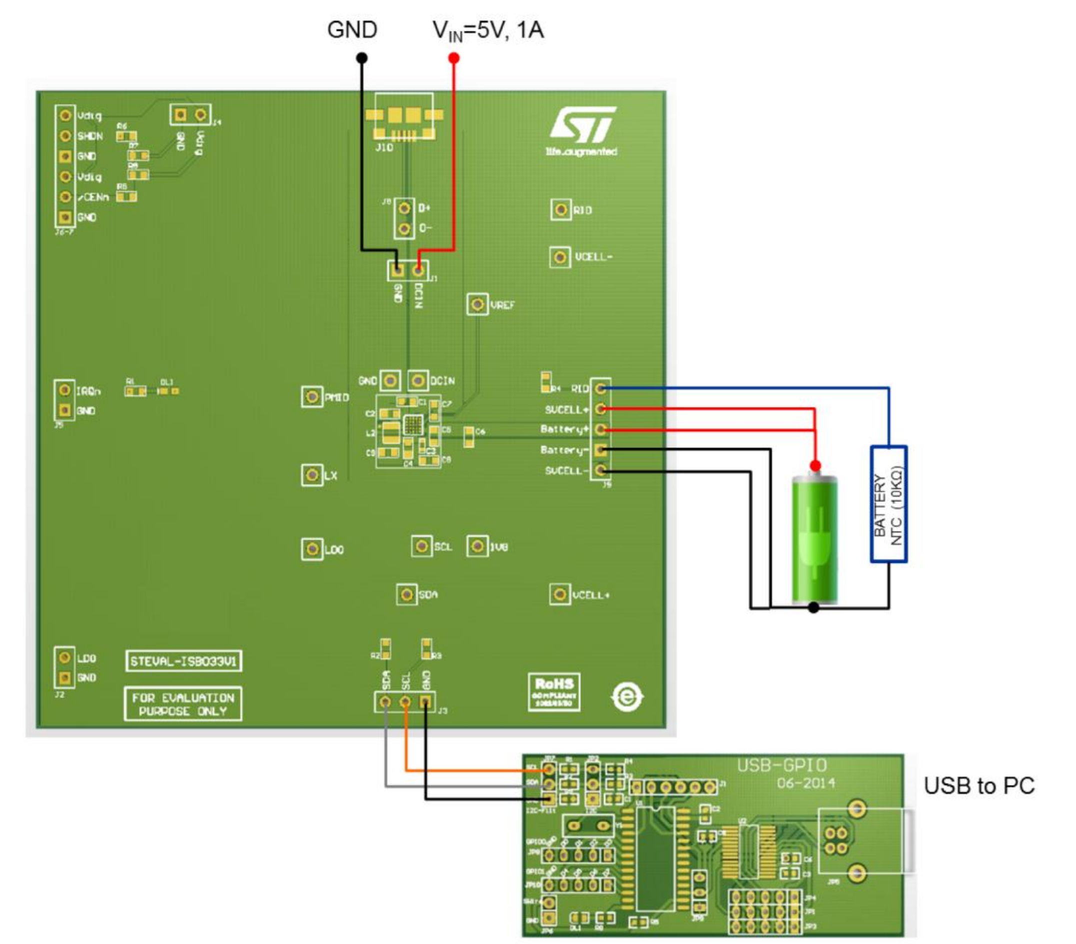
3 Input / Output connections

Table 2: Input and output connections

Reference designator	Name	Description
J1	Input Voltage	J1_1: GND power J1_2: DCIN power
J2	LDO	J2_1: LDO pin J2_2: GND
J3	I²C	J3_1: SDA pin J3_2: SCL pin J3_3: GND
J4	1.8 V	J4_1: GND J4_2: VDIG pin
J5	IRQn	J5_1: IRQn pin J5_2: GND
J6-7	SHDN-/CENn	J6_1: GND J6_2: CENn pin J6_3: VDIG J6_4: GND J6_5: SHDN pin J6_6: VDIG
J8	USB DATA (not used)	J8_1: D- J8_2: D+
J9	Battery	J9_1: RID pin J9_2: VCELL+ pin J9_3: R_SNS_N pin / Battery+ J9_4: GND / Battery- J9_5: VCELL- pin
J10	MICRO USB Input Voltage	J10_1: GND J10_2: NC J10_3: D+ (to J8) J10_4: D- (to J8) J10_5: DCIN pin

4 Connections

Figure 3: Input / Output connections



1. Connect a power supply source between DCIN (J1_2) and GND (J1_1) pins; otherwise plug the USB cable to a USB wall adapter.
2. Connect a battery between Battery+ (J9_3) and Battery- (J9_4) pins and connect the Battery sensing wire VCell+ (J9_2) and VCell- (J9_5) directly on the battery poles.
3. Connect the board to the ST USB-GPIO interface through the SDA (J3_1), SCL (J3_2) and GND (J3_3) pins.
4. Connect the ST USB-GPIO interface to the PC to drive the STBCFG01 with the STBCFG01 GUI described in *Section 5: "GUI description"*.

5 GUI description

The STBCFG01 GUI (Graphical User Interface) can be used to read, check, modify and write all the STBCFG01 registers and simplifies register table control, thanks to its user-friendly Windows®-based graphical interface.

All the STBCFG01 parameters are programmable through Slider, Check-LED and Text inputs in the GUI.

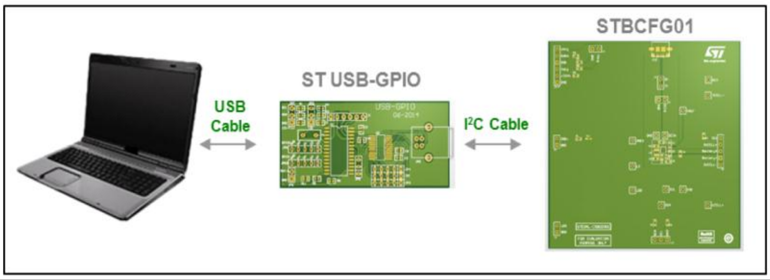
The STBCFG01 GUI is connected to the STBCFG01 application board through an STUSB-GPIO interface that provides the correct I²C protocol to drive the DUT (Device Under Test).

The evaluation kit consists of:

- ST USB-GPIO hardware
- Graphic User Interface setup file
- USB drivers
- User manual

5.1 How to start

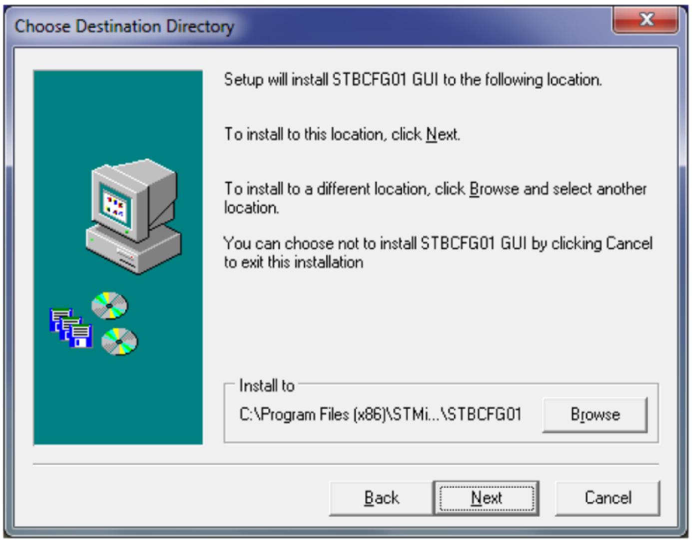
Figure 4: How to Start



5.1.1 Installation

Some software installation is required to run the STBCFG01 GUI.

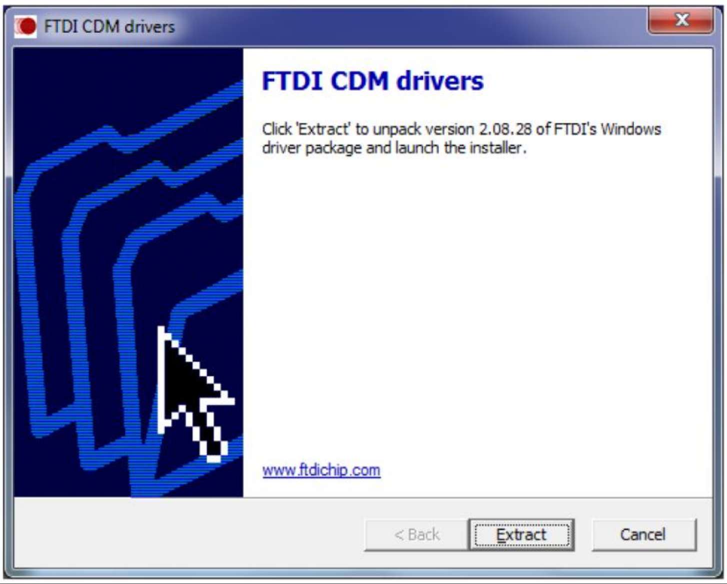
Figure 5: GUI installation



Run the installation package to start the guided software setup procedure.

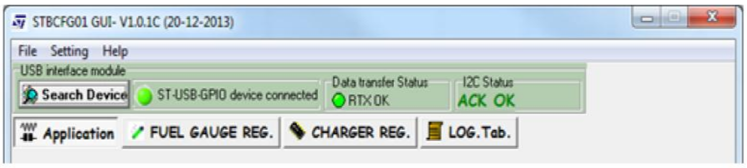
When the GUI installation finishes, the USB driver installation shown below commences automatically. Wait until this window closes to complete the installation.

Figure 6: USB driver installation



- Connect the ST USB-GPIO interface to the PC via the USB cable. The supply voltage is provided by the USB connection.
- Start the STBCFG01 GUI.
- At startup, the GUI should automatically connect the ST USB-GPIO interface; if it doesn't, press the "Search Device" button and wait few seconds until the "ST-USB-GPIO Device Connected" status indicator turns on, as shown in the figure below. This indicator can either be:
  - light green if the ST USB-GPIO interface is connected;
  - dark green if the ST USB-GPIO interface wasn't retrieved.

Figure 7: ST USB-GPIO interface connection



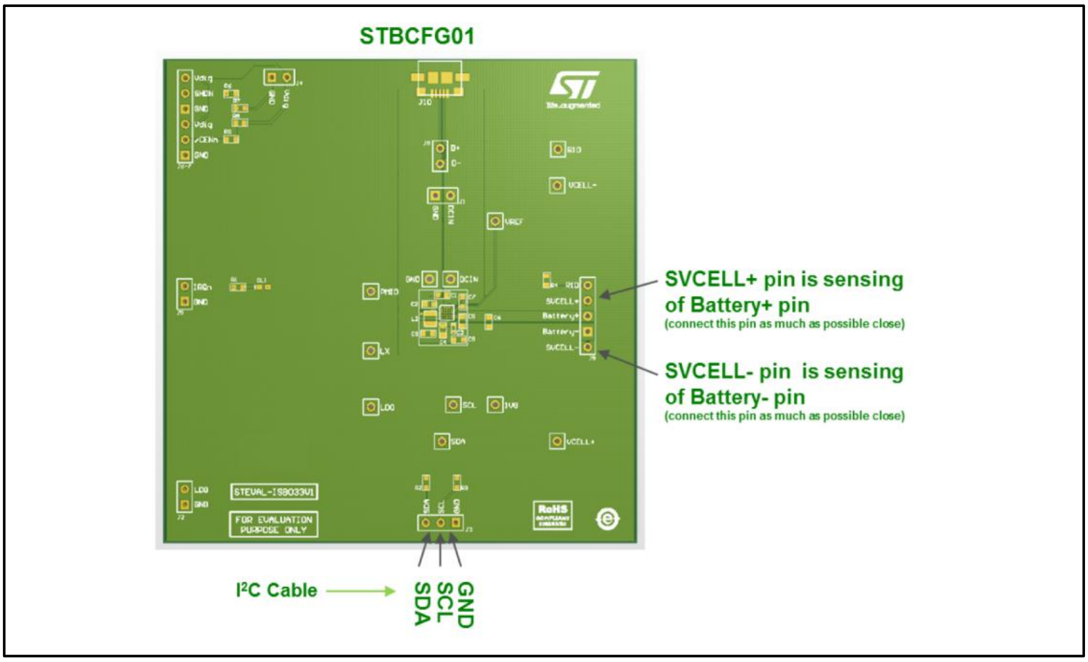
If the board isn't connected (status indicator is unlit), try the following:

- Check the USB cable connection between the PC and the ST USB-GPIO interface;
- Press the "Search Device" button on the GUI and wait a few seconds.

If the board is connected (status indicator lit), the GUI can be used to control the DUT as described as below.

- Connect the control board to the I²C pin using a short cable (max. length 30 cm) as shown below.

Figure 8: STBCFG01 demo board



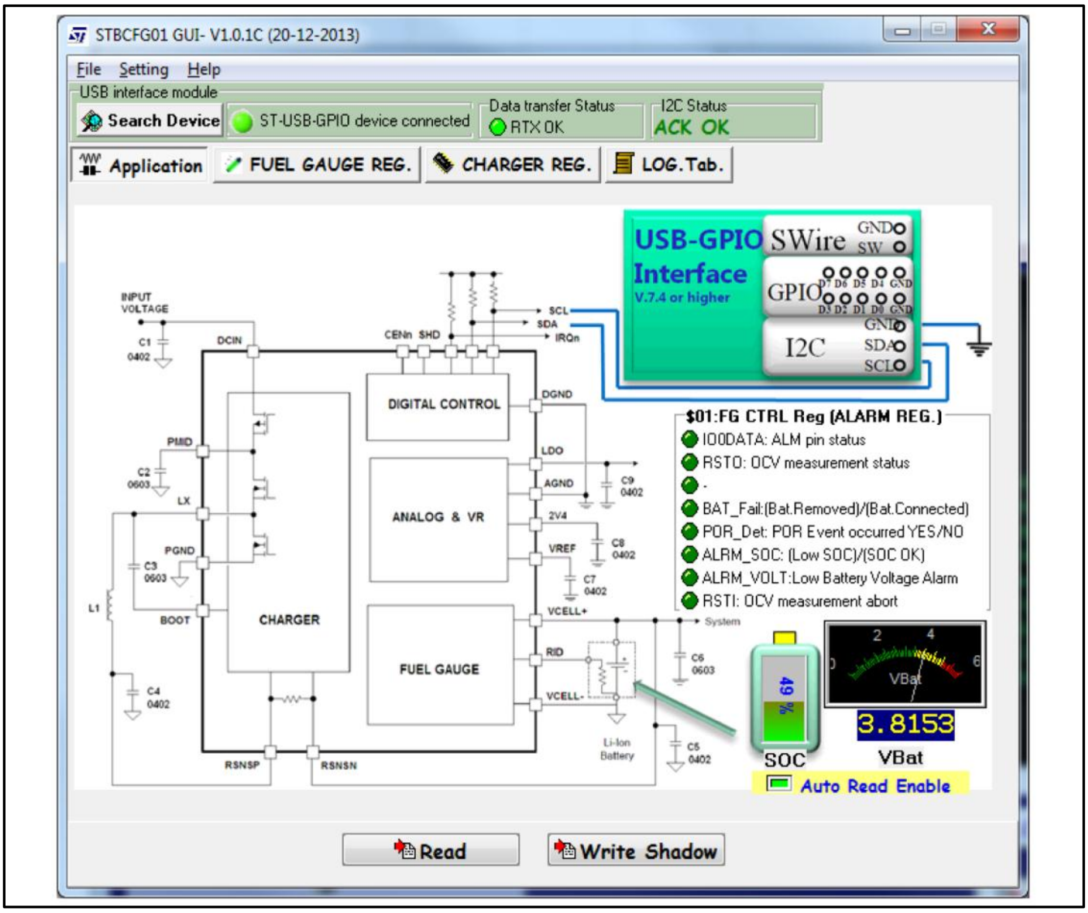
- The "Data Transfer Status" indicator is refreshed when read or write commands are sent to the DUT; it can be:
  - green (RTX OK) for correct data transfer between the GUI and the ST-USB-GPIO interface;
  - red (RTX Error) for incorrect data transfer between the GUI and the interface
- The "I²C Status" indicator shows the acknowledge status from the I²C bus; it is refreshed when read or write commands are sent to the DUT; it can either be:
  - green (ACK OK) to indicate correctly Acknowledged;
  - red (ACK Fail) to indicate not Acknowledged.

Do NOT disconnect the USB connection while the GUI is open as this causes a software error. Be sure to quit the GUI before disconnecting the USB cable.

5.2 General description

The GUI figured below has several tabs and sections.

Figure 9: GUI: main page



5.2.1 USB interface module

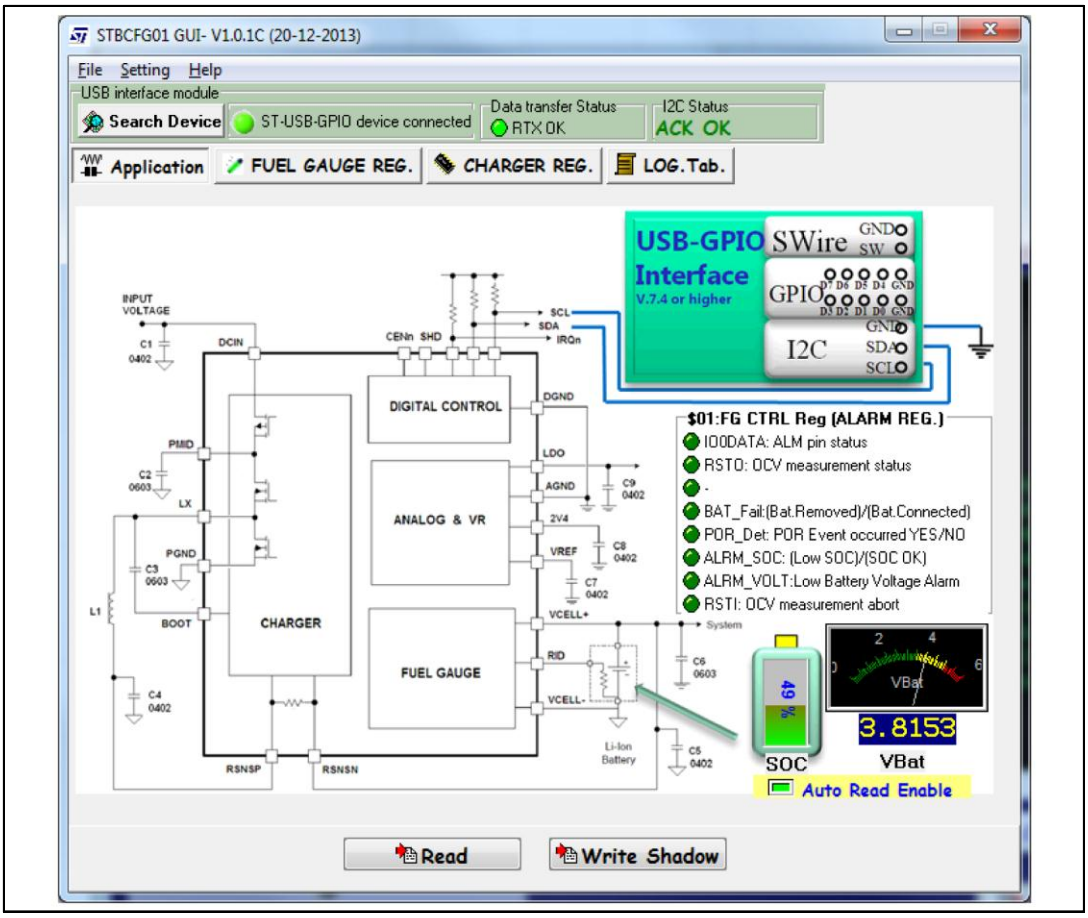
This block is used to connect and check the ST USB to GPIO operation.

5.2.2 TAB overview

The GUI offers several tabs which you can interact with to manipulate the registers:

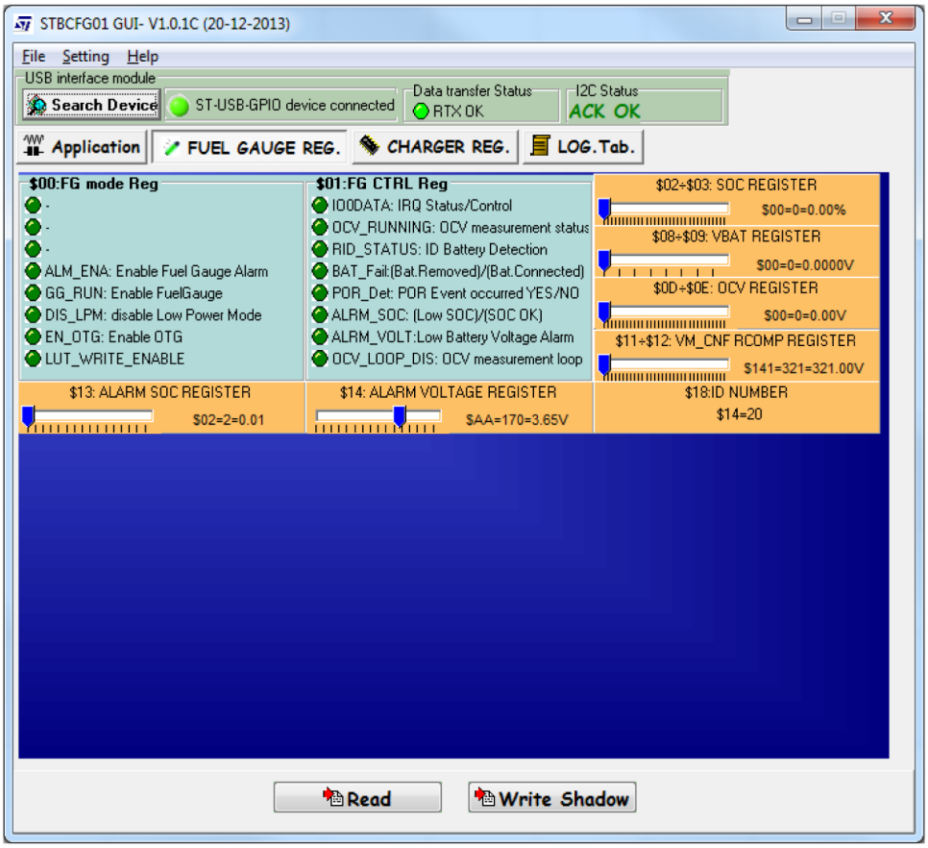
Application: shows the STBCFG01 Application Schematic. The "Auto Read Enable" allows the status of battery to be displayed, as shown below.

Figure 10: Auto read enabled



FUEL GAUGE REG.: shows the fuel gauge register tab, where you can enable several functions like OTG. You can also read the battery value and state of charge (SOC).

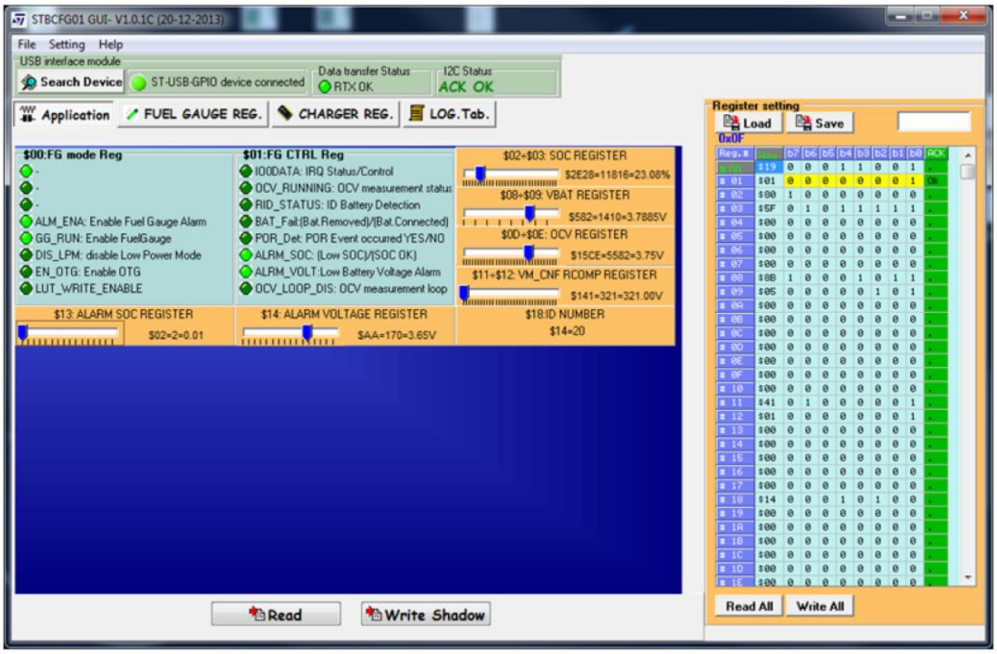
Figure 11: Fuel gauge register tab



Register content changes can be viewed in the register map, which is displayed by checking "Show Bit Table" in the settings, as shown below.

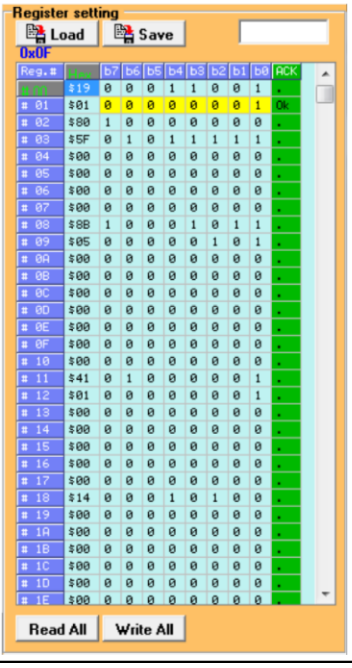
When you mouse over a control slider, the corresponding bits in the register table are highlighted yellow. This table is editable, with any changes made to the bits in the register reflected in the control sliders, and vice versa.

Figure 12: "Show Bit Table" set



The first column contains the register address and the second contains its value in hexadecimal format; columns b7….b0 show the same value in binary format, as shown below.

Figure 13: Register map detail

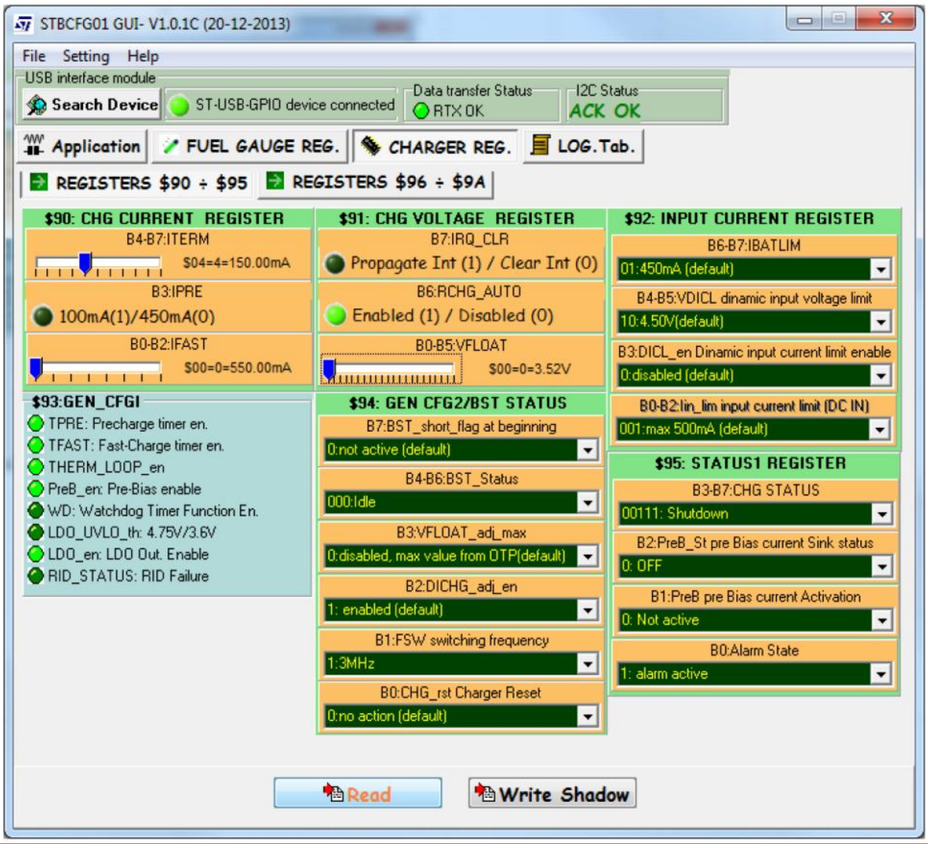


You can select one or more bits, click the right mouse button and select the write or read command. By selecting a specific register, you can write its value directly in the white text box.

The "$" character indicates hexadecimal notation.

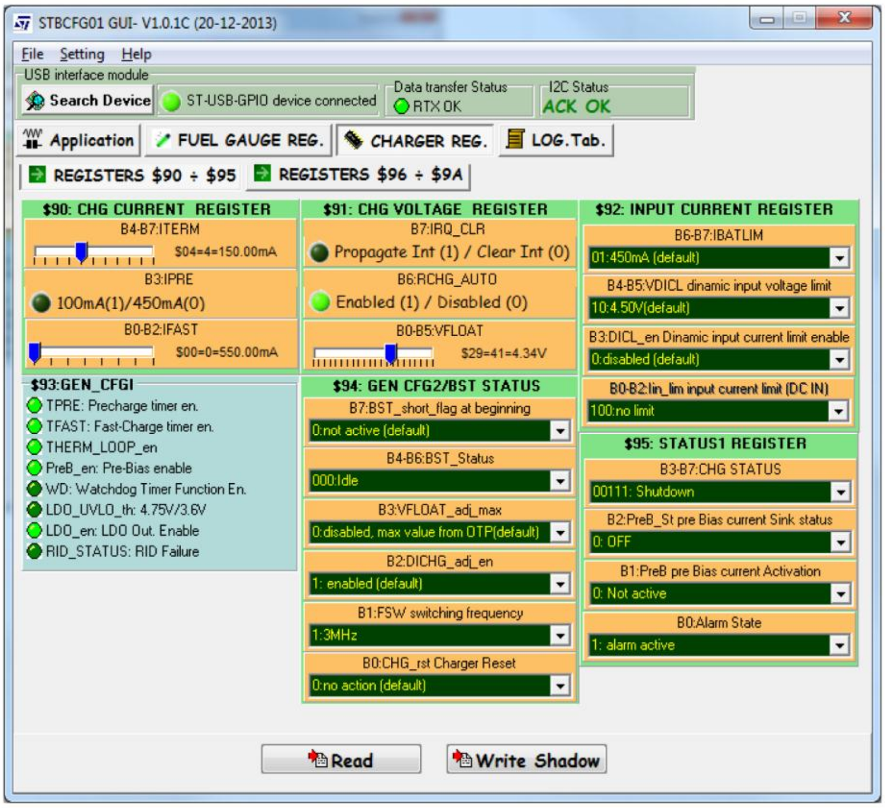
CHARGER REG.: shows the charger register, where the "REGISTER $90÷$95" button gives access to several protection functions and charge current and voltage manipulation.

Figure 14: Charger register tab



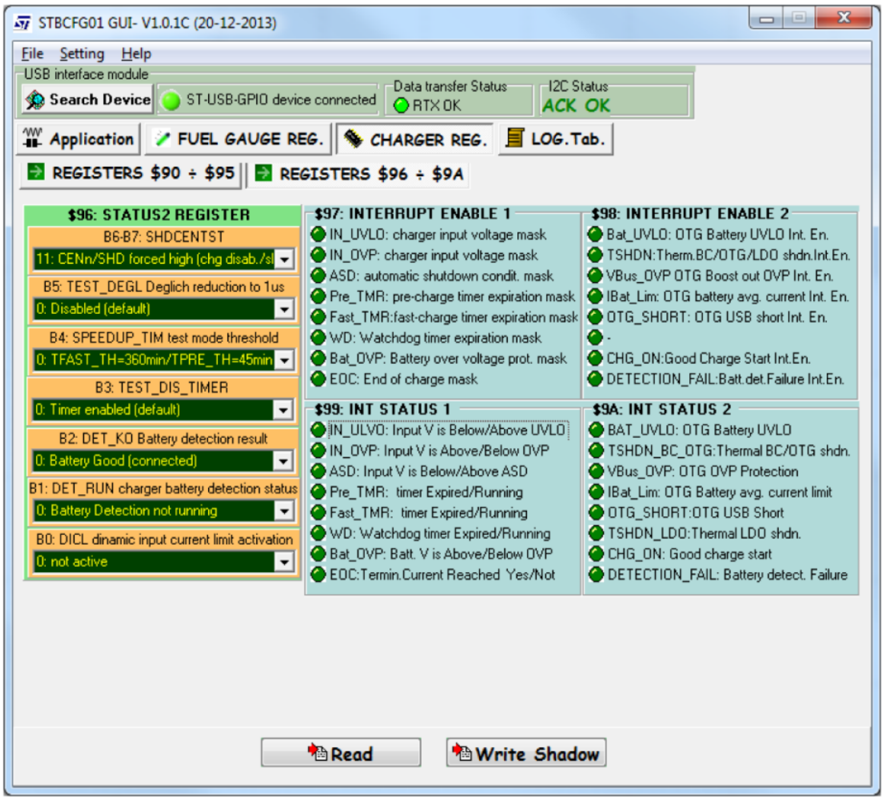
Initially, the registers are set to the default value (see datasheet). The registers must be set according to the type of battery used (see below).

Figure 15: Register setting



The "REGISTER $96÷$9A" allows you to enable and view INTERRUPT states (on/off). "INT STATUS" shows the active interrupts, while "INTERRUPT ENABLE" renders the specific interrupt visible on IRQ pins.

Figure 16: Interrupt status

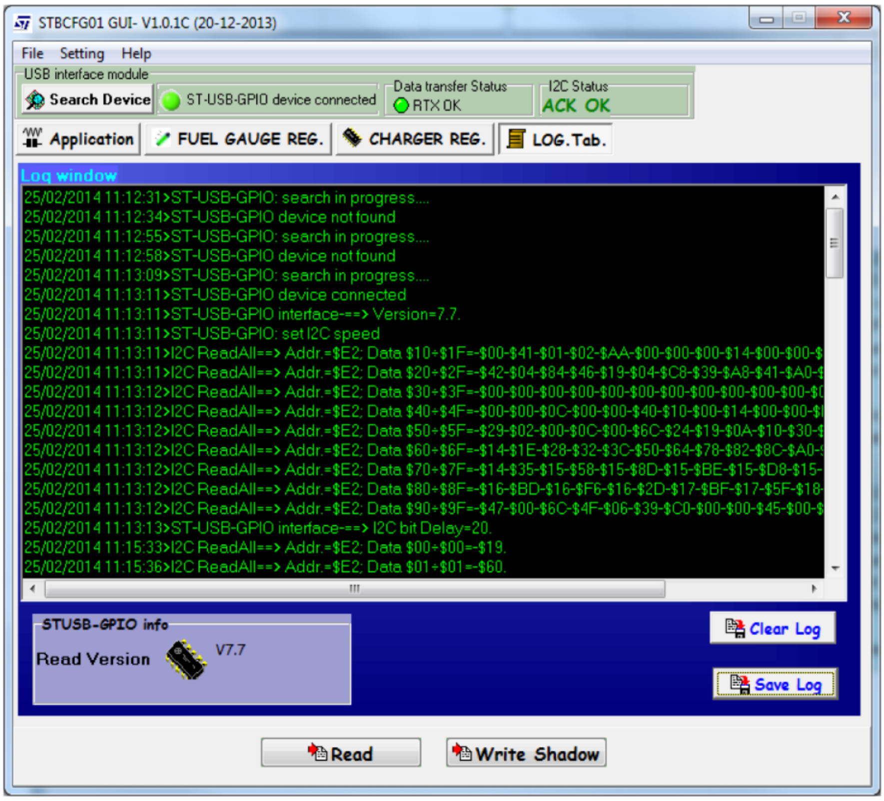


Log: the LOG tab opens a log of the events occurring during operation, very useful for debugging the D.U.T. and the GUI.

The "Save Log" button saves the content of the Log window, while the "Clear Log" button clears the window.

The ST-USB-GPIO info block provides information related to the USB-GPIO interface, such as the firmware version.

Figure 17: Log window



6 Application information

6.1 Operation description

6.1.1 Battery charger

The STBCFG01 integrates a high efficiency 1.2 A battery charger implementing the CC/CV charging algorithm for single cell Li-Ion battery powered applications. The switching frequency can be either 2 MHz or 3 MHz, according to platform noise requirements, and the inductor value is 1 µH. The charging current sensing resistor is integrated.

When the battery is heavily discharged (VCELL+ < V_{TRK}, V_{TRK} = 2 V) the device enters trickle charge mode and charges the battery in linear mode with a low current (I_{TRK} = 45 mA) up to the trickle charge threshold.

As soon as the battery voltage enters the pre-charge range (V_{TRK} < VCELL+ < V_{PRE}, V_{PRE} = 3 V), the device commences switch mode charging and increases the charging current up to the pre-charge current level (I_{PRE}) so the system voltage quickly rises to a level which allows the system to wake up.

The typical value for the pre-charge current is 450 mA, but this value can be decreased to 100 mA through the I²C compatible interface.

A 45 minute (typ.) safety timer is active during both trickle charge and pre-charge modes.

When the battery voltage rises above the pre-charge threshold, the STBCFG01 enters fast charge mode and increases the charging current up to the I_{FAST} value. The fast charge current can be set through the I²C compatible interface to between 550 mA and 1.2 A in 100 mA steps.

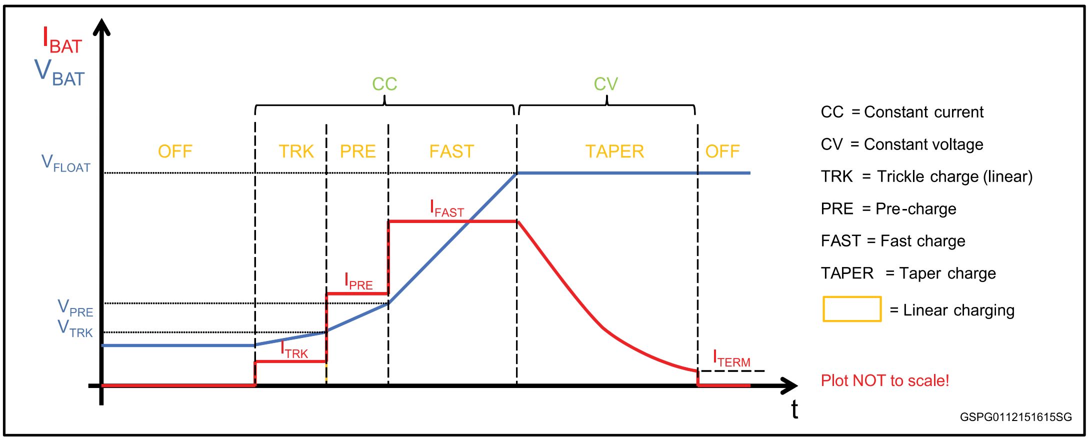
A soft-start function forces the battery current to change gradually when the charging current is altered.

The constant voltage mode is triggered when the battery voltage reaches the programmable floating voltage threshold (V_{FLOAT} = 3.60 V to 4.70 V in 20 mV steps). In this mode, the charging current tapers down to the termination current threshold, where the charging process stops. The termination current is programmable from 50 mA to 300 mA in 25 mA steps.

A 360 minute (typ.) safety timer is active when the fast charge starts.

The charger can be disabled pulling the charger enable input (CENn) high and is automatically stopped under automatic shutdown conditions [ASD, (V_{DCIN} - V_{BAT}) < V_{ASD}].

Figure 18: CC/CV charging cycle



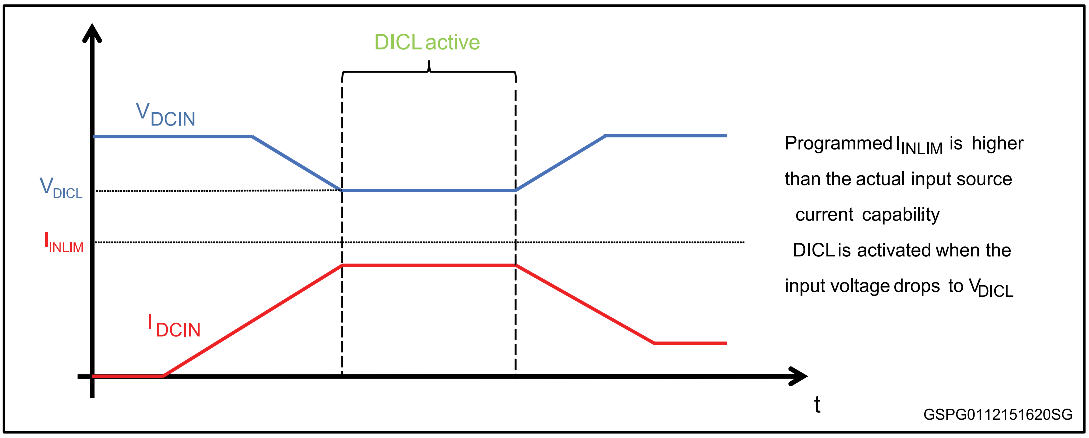
6.1.2 Input current limit

The STBCFG01 implements a programmable input current limitation to prevent the battery charger from exceeding the DCIN voltage source current capacity. The current limit can be set through I²C to 100 mA, 500 mA, 800 mA and 1.2 A maximum values. A "no limit" option is also available.

6.1.3 Dynamic input current limit (DICL)

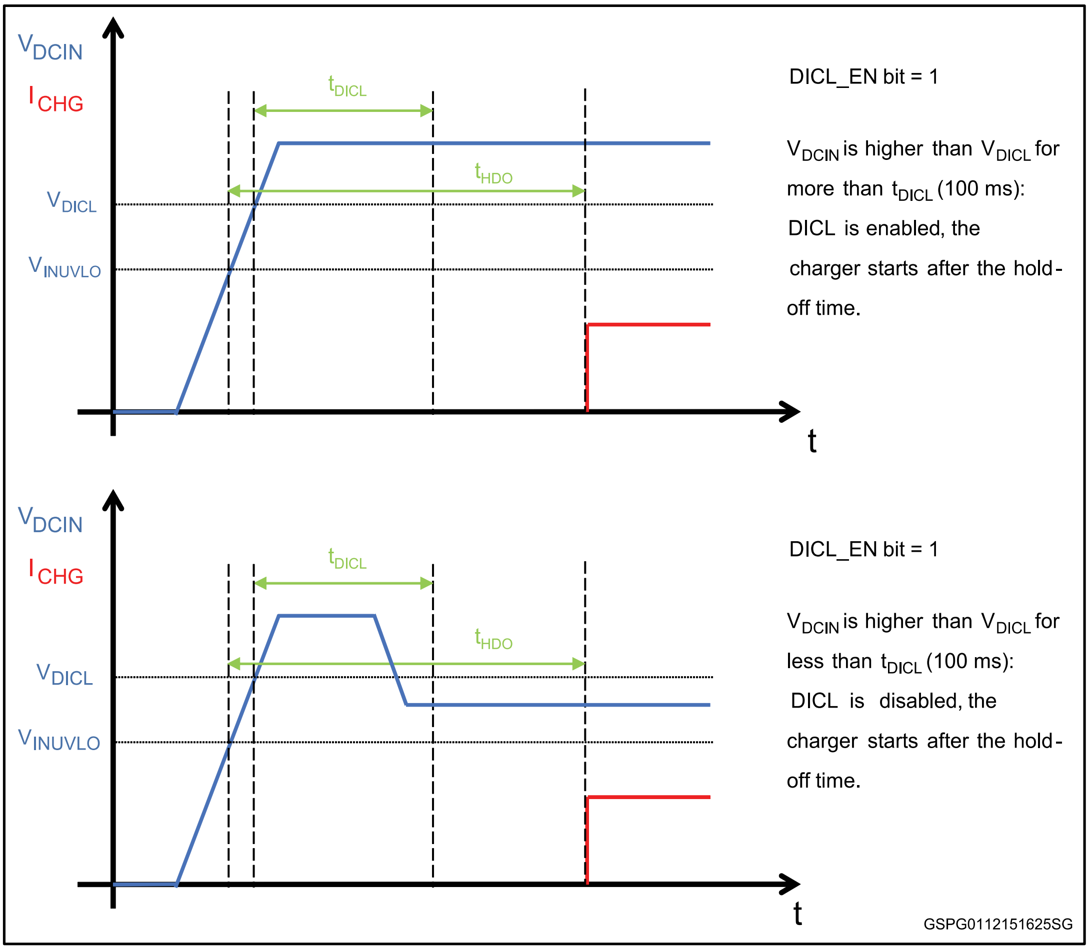
Regardless of the chosen input current limit, a dynamic input current limit loop can also be enabled through the I²C compatible interface (DICL_en bit). When DICL is active, an analog loop limits the input current when the input voltage drops to a programmable threshold (V_{DICL} = 4.0 V to 4.75 V in 250 mV steps).

Figure 19: DICL activation



When a valid input source is connected to DCIN, DICL is only enabled if the input voltage is higher than V_{DICL} for a deglitch time (t_{DICL}=100 ms). The user can check the enabling status of DICL function by reading DICL status bit. Once DICL is enabled, it can be disabled through the DICL_en bit.

Figure 20: DICL enable strategy



6.1.4 Automatic recharge

When the charging cycle is over, the device continues monitoring the battery voltage: if the voltage drops below the auto-recharge threshold ($V_{RCHG} = V_{FLOAT}$ - 120 mV), a new charging cycle begins to maintain the battery at maximum capacity. The automatic recharge function can be disabled through I²C.

6.1.5 Battery detection

The battery charger IC features a battery detection function to determine whether a battery is inserted before starting the charging cycle and to ensure that charging is interrupted when a battery is disconnected.

If a battery is not detected when the input voltage is valid, the device keeps running the detection sequence until a battery is inserted. The battery detection outcome is given by the DET_ok bit. If the battery detection function or RID comparator detects battery disconnection, a BAT_Fail interrupt is generated. See the battery detection algorithm flowchart for more details.

6.1.6 Battery overvoltage protection

If the battery voltage exceeds the battery overvoltage protection threshold ($V_{BATOVP}=V_{FLOAT}$+100 mV) for more than t_{BOVP}, the charging cycle stops and an alarm is generated.

The battery overvoltage protection is only active when the charger is enabled. The charger automatically restarts when the battery voltage falls below the battery overvoltage protection threshold.

7 Board layout

Figure 21: Assembly layer

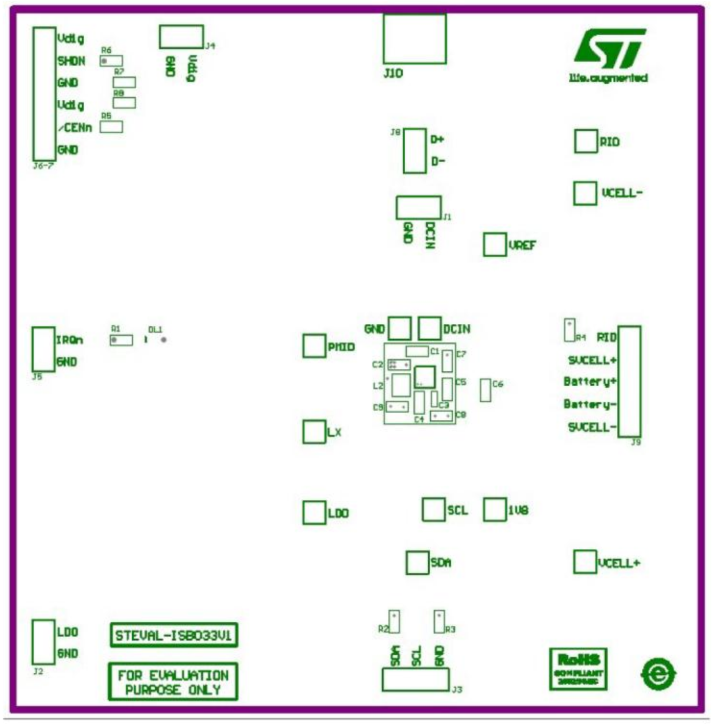


Figure 22: Top layer

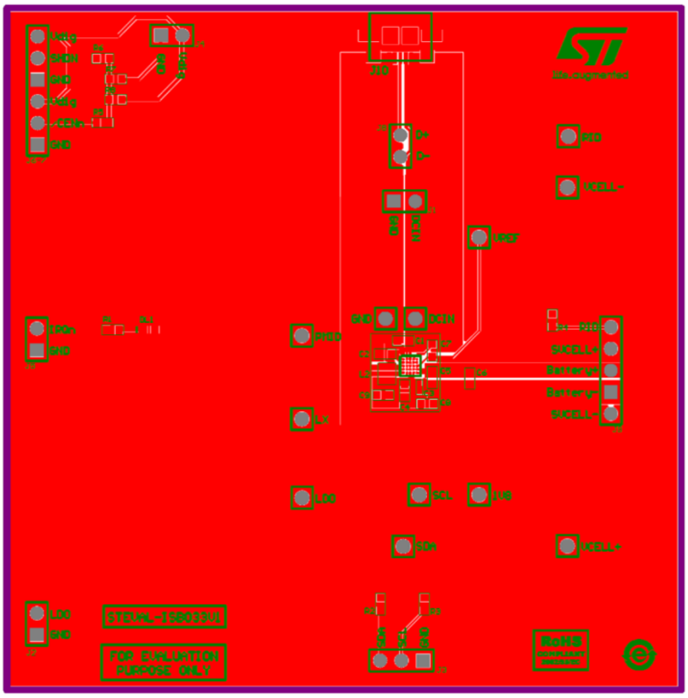
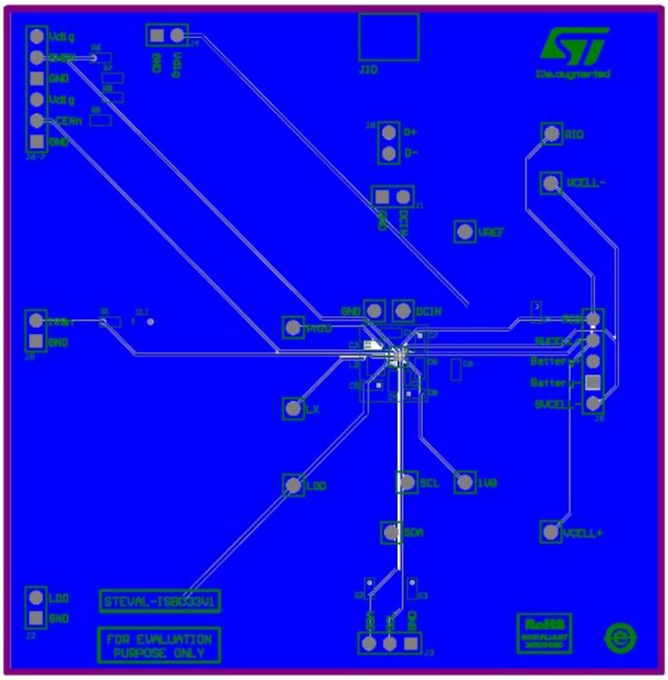


Figure 23: Bottom layer



8 Revision history

Table 3: Document revision history

Date	Revision	Changes
16-Dec-2015	1	Initial release.