

## Getting started with the STEVAL-WBC86TX wireless power transmitter evaluation board for up to 5 W Qi-BPP applications

### Introduction

The [STEVAL-WBC86TX](#) evaluation board, based on the [STWBC86](#), is designed for wireless power transmitter applications and allows its users to quickly start their 5 W Qi BPP wireless charging projects.

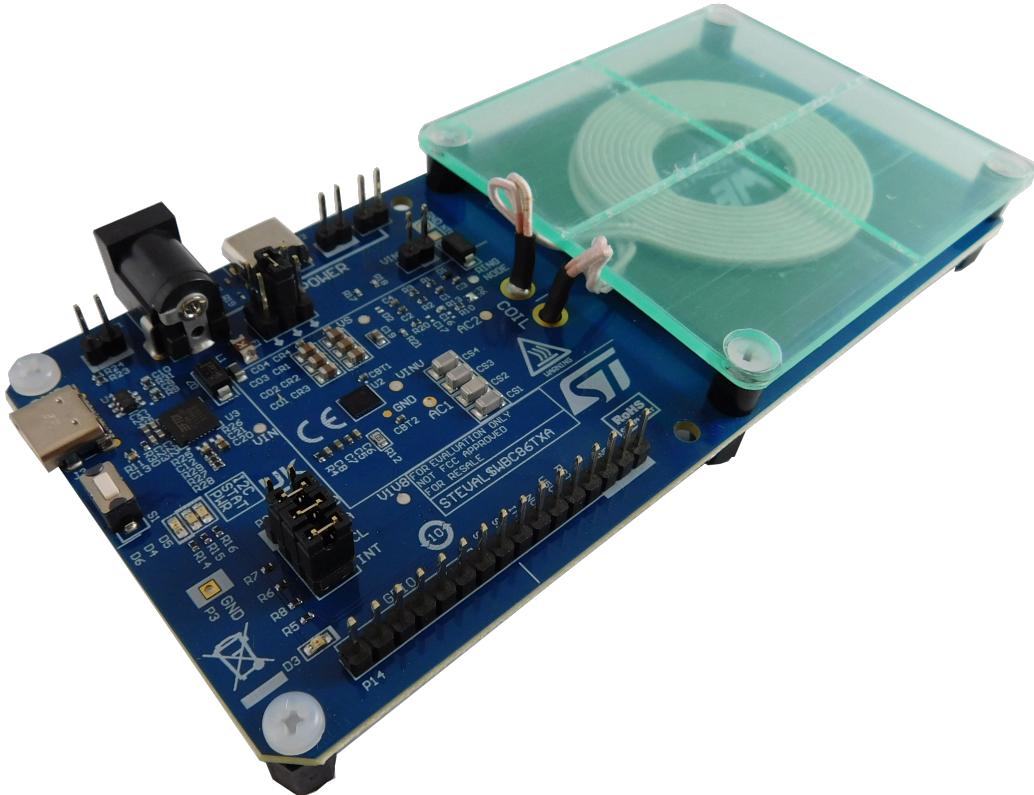
The STWBC86 wireless transmitter IC can deliver up to 15 W (coil dependent), however this reference design document is limited to only provide sufficient information to develop a project for up to 5 W charging compatible with Qi 1.2.4 Baseline Power Profile (BPP) power transfer by Wireless Power Consortium's inductive wireless power technology.

The integrated circuit requires only a few external components and can work with 5-20 V input voltage.

Using an on-board USB-to-I2C bridge, the user can monitor and control the [STWBC86](#) using the [STSW-WPSTUDIO](#) graphical user interface (GUI).

The [STEVAL-WBC86TX](#) includes several safety mechanisms providing overtemperature (OTP), overcurrent (OCP), and overvoltage (OVP) protections as well as foreign object detection (FOD) for reliable designs.

**Figure 1. STEVAL-WBC86TX board**



To get started with the STEVAL-WBC86TX, the following items are needed to use the reference design kit:

- Evaluation kit components:
  - STEVAL-WBC86TX board

- Additional hardware:
  - USB adapter 5 V / 3 A or power supply
  - 2 x USB Type-C® cables (one can be replaced with either 2.1 mm jack or pin cable)
  - Windows PC
- Software:
  - [STSW-WPSTUDIO](#) Wireless Power Studio PC GUI installation package
  - I2C drivers
- Application notes:
  - GUI guide: UM3164

Begin by installing both the I2C drivers and the STSW-WPSTUDIO GUI. Visit the [ST website](#) for additional information regarding the [STSW-WPSTUDIO](#) GUI.

Connect a 5 V power supply to power the board using either the USB Type-C®, jack, or pin cable. Using a jumper, select the chosen method of power delivery on header P1.

Using a USB Type-C® cable, connect the board to the PC (connector P4 on the board). This allows the user to communicate with the board - program it and monitor its function.

## 1 Reference design specifications

The target specifications of the STEVAL-WBC86TX evaluation board are as follows:

**Table 1. STEVAL-WBC86TX target specifications**

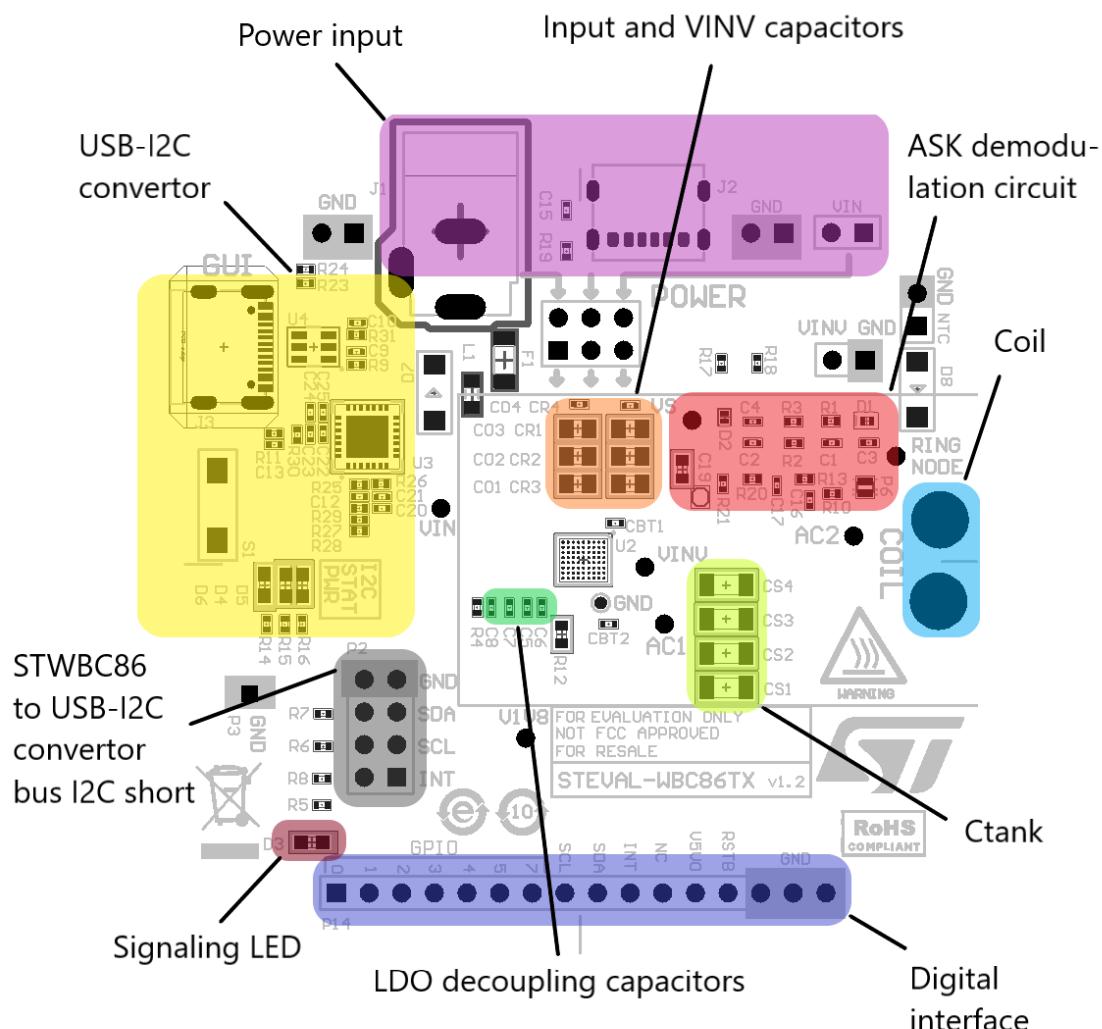
Parameter	Description
Qi compatibility	Qi 1.2.4 compatible
Tx application PCB area	40 mm x 24 mm
Tx coil specifications	Inductance 6.3 uH, DCR 20 mOhm, ACR 20 mOhm @ 100 kHz, dimensions 53.3 mm x 53.3 mm x 6 mm
Qi Tx topology	A11a
Input voltage (Vin)	5 V
Input current (Iin)	1.5 A
Host MCU	STM32 used as a reference, the reference I2C driver can be ported to any other MCU family
USB-to-I2C converter	FT260, embedded in the evaluation board
Efficiency	77.6 % (5 W operation) with STEVAL-WLC38RX 81 % (peak efficiency) with STEVAL-WLC38RX at 3 W
Applicable charging gap between Tx and Rx coils (z-distance)	3 – 13 mm (5 W output) with STEVAL-WLC38RX receiver, maximum 16 mm – stable communication without output enabled
Operational modes	Transmitter only

## 2 Overview of the board

The STEVAL-WBC86TX evaluation board is optimized for performance. The board features:

- STWBC86 wireless power transmitter chip with BPP 1.2.4 compatible firmware
- Very few external components, optimized BOM and PCB space
- On-chip high efficiency full bridge inverter
- 32-bit, 64 MHz Arm® Cortex® microcontroller with 8 KB SRAM
- 9-channel, 10-bit A/D converter
- On-chip thermal management and protections
- Foreign object detection (FOD) function
- I2C interface for communication with host system (optional)
- On-board USB-to-I2C converter
- Chip scale package (CSP), ROHS compliant

Figure 2. STEVAL-WBC86TX evaluation board features



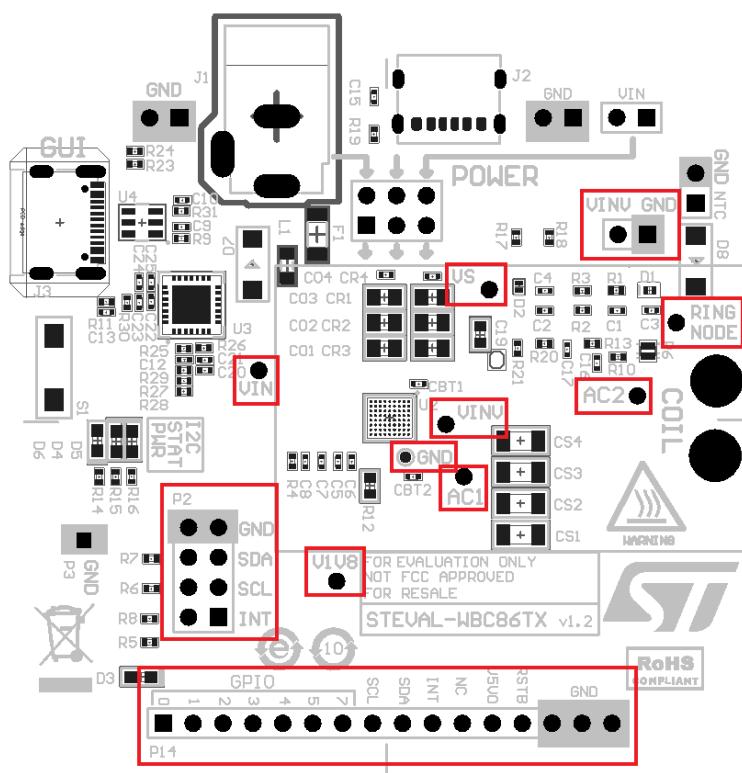
- Series resonant capacitors (Ctank) and the transmitting coil form a resonant circuit. This circuit is in charge of transmitting the power signal, so any components/tracks involved should be rated accordingly.

- CBT1 and CBT2 are bootstrapping capacitors, which ensure the proper functionality of the integrated inverter. This should be considered during PCB design, as these nets generate noise and should therefore be routed separately from sensitive circuits.
- ASK demodulation circuit - apart from transferring power, the power signal is also used for receiver to transmitter communication. The communication signal is extracted from the power signal using the ASK demodulation circuit and fed into the VS pin of STWBC86 for processing. For further details, refer to [Section 4.12.1 ASK communication](#).
- USB/I2C converter - provides a communication channel between a PC and STWBC86. LED D6 (red) indicates the I2C converter is powered, D4 (yellow) indicates that STWBC86 is connected to the GUI. LED D5 (green) indicates the I2C communication was initialized and is ready. Switch S1 resets the converter. Please note that header P2 connects the converter's I2C signals to the STWBC86 I2C signals. Short the corresponding pins with a jumper to establish a connection between the two ICs.
- Power input (USB Type-C® connector/jack/pin header) - 3 separate inputs can be used to power the board, but only one is used at a time. Therefore, it is necessary to select the input using a jumper on header P1.
- Red LED (D3) - connected to GPIO0, can be configured to signal various conditions (power ready, communication active etc.).

## 2.1 Test points

The [STEVAL-WBC86TX](#) features several connectors and test points to provide easy access to key signals.

**Figure 3. Connectors and test points**



**Table 2.** Connectors and test points

Connector / test point	Name	Description
VIN	VIN	Input voltage (power pins)
VINV	VINV	Inverter voltage pins
P2	I2C	SDA, SCL, INT, and RST signals for I2C communication
P14	Digital interface	I2C, GPIO, and RST signals
TP1	Ring node	Ring node
TP2	VIN	STWBC86 input voltage sensing
TP3	AC2	Resonant circuit terminal
TP4	AC1	Resonant circuit terminal
TP5	VINV	STWBC86 inverter voltage sensing
TP6	V1V8	STWBC86's 1.8 V LDO output
TP7	GND	Ground
TP8	VS	VS signal sensing

## 2.2

### Basic operating modes

The STWBC86 is designed to work in transmitter mode only. Once the board is powered up, the device automatically starts pinging (if enabled), which means it starts scanning its power transfer interface for a potential power receiver. Once a suitable receiver is found, the STWBC86 initializes power transfer. After the receiver is removed from the interface, the device returns to the pinging phase.

## 3 Graphical user interface (GUI)

The STWBC86 (and other STMicroelectronics wireless charging devices) can be configured using STMicroelectronics' STCHARGE Wireless Power Studio GUI. The GUI can also be used to control, monitor, and program the device.

For more information, please see the UM3164.

### 3.1 Connecting STWBC86 to PC GUI

Connect the board to a PC by plugging a USB Type-C® cable into the connector J3. Make sure the STWBC86 I<sub>2</sub>C pins are connected to the USB Type-C® connector. This can be done by shorting the appropriate signals (SDA, SCL, INT) on header P2. Power up the board and open the GUI on your PC. Click the Connection button in the top menu.

Up to two devices can be connected at a time - this allows the user to control both Rx and Tx at the same time). Select WBC86 as the Tx and click the Connect button on the right side of the window.

Figure 4. GUI connection

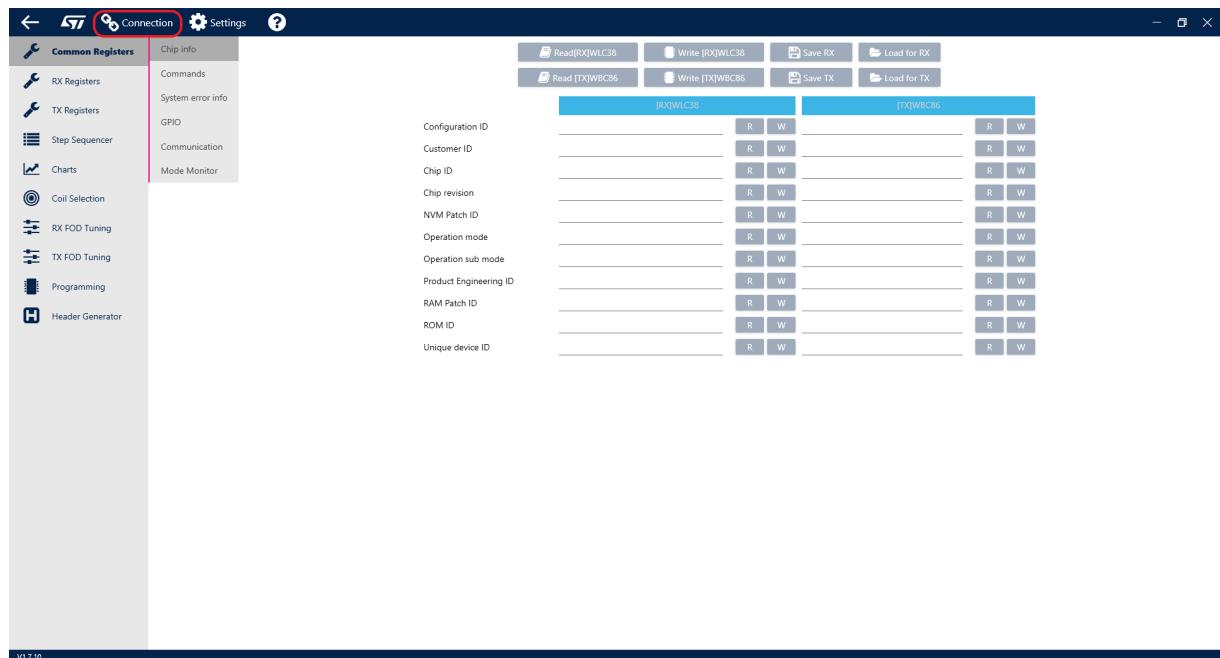
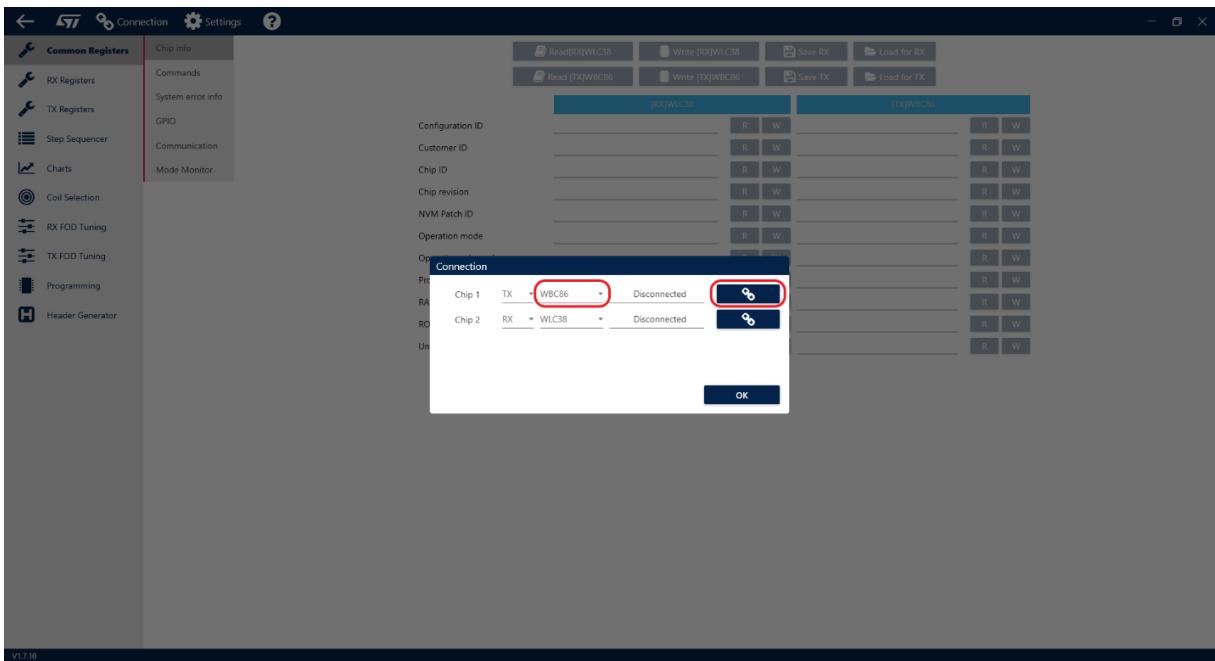


Figure 5. GUI device connection



### 3.2

## Patch and Configuration files

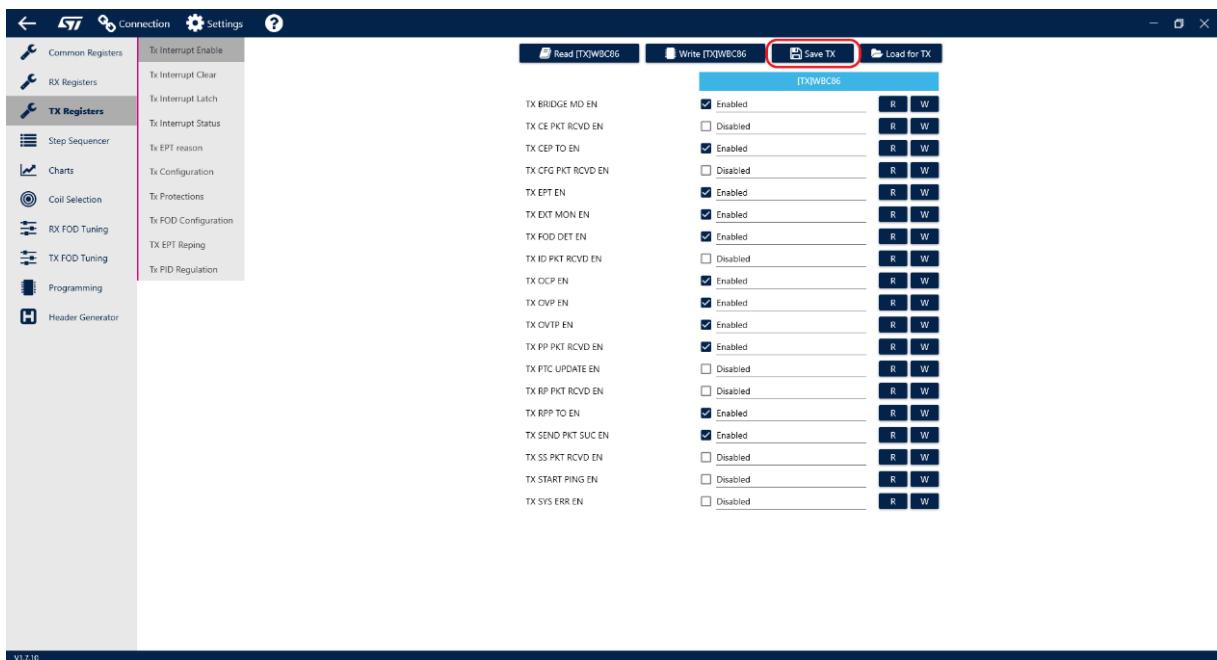
Firmware of the device can be updated using a Patch file (a binary file in .memh format). The latest version of the Patch can be found on this [\[ST website\]](#). Updating the firmware is not required but may improve performance of the board.

The device can be configured using a Configuration file, a binary file containing settings of all registers, which can be found in the GUI. The GUI can also be used to generate a custom Configuration file, making it easier to quickly change configuration of the board and/or transfer the configuration to another board.

### 3.3 Configuration file generation

Using the STSW-WPSTUDIO makes generating the Configuration file quite simple – the user can do so by clicking the “Save TX” button in the TX Registers tab or the Common Registers tab, entering a configuration ID number (used for version control) and pressing OK. The GUI then asks for a save destination. After choosing a location, the Configuration is saved as a .memh file in the selected folder.

**Figure 6. Generation of configuration file**



**Figure 7. Version of the configuration file**

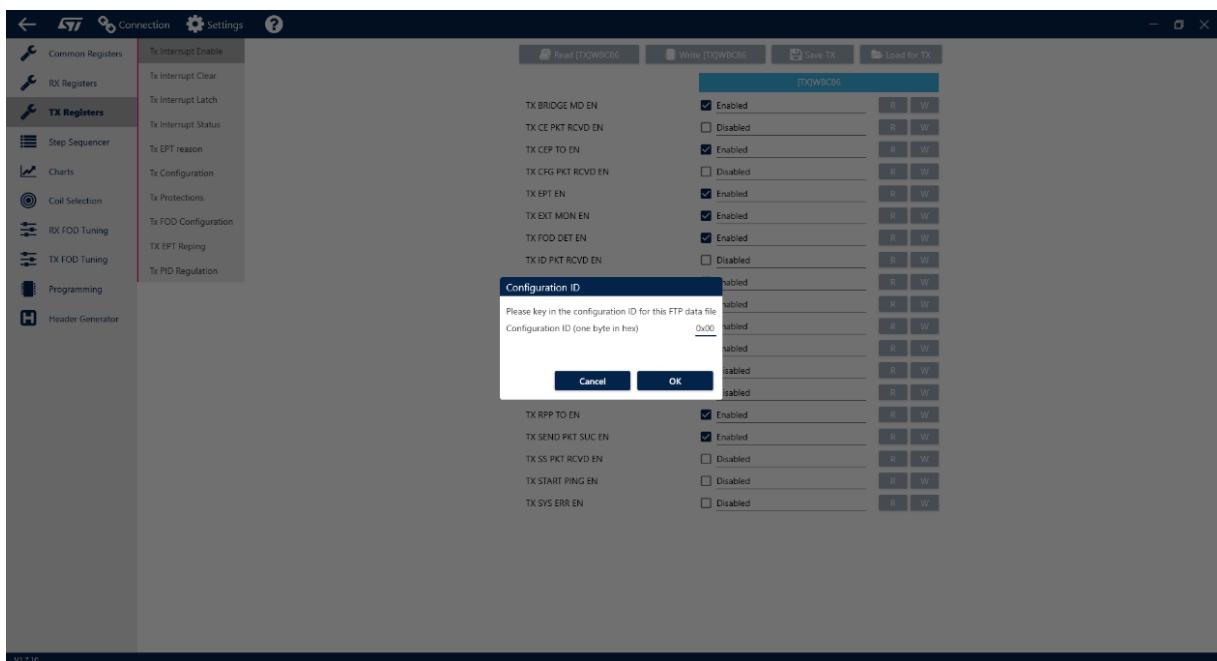
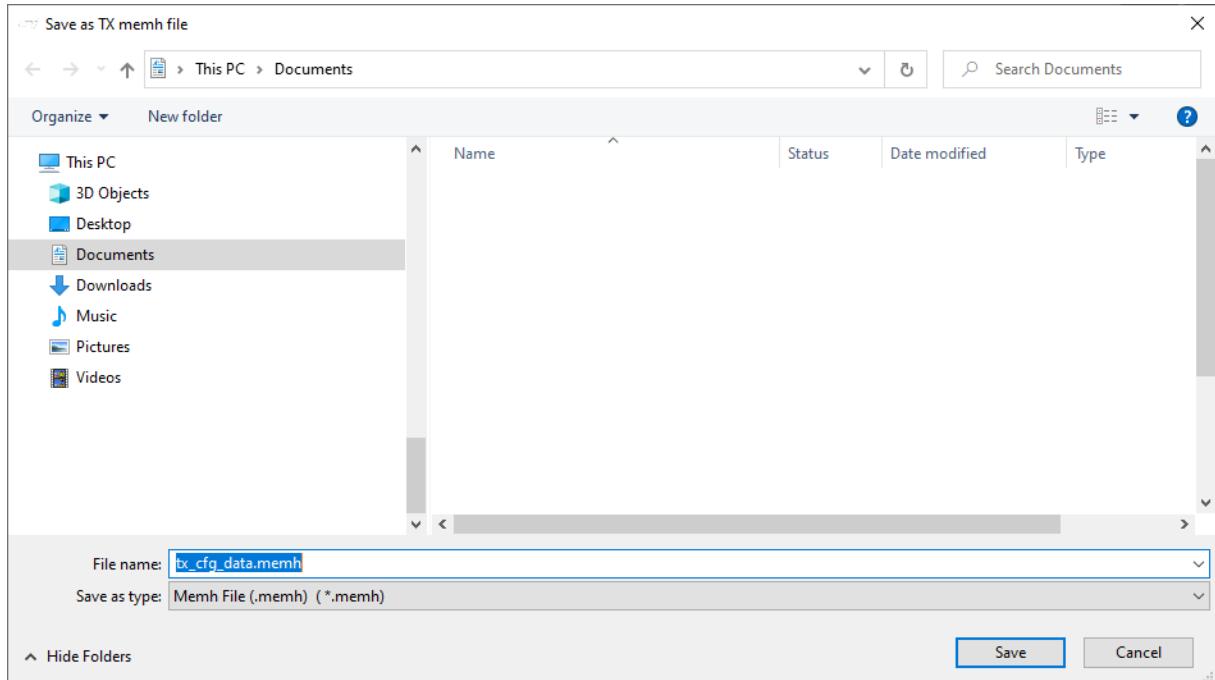


Figure 8. Saving of the configuration file



### 3.4

#### Header file

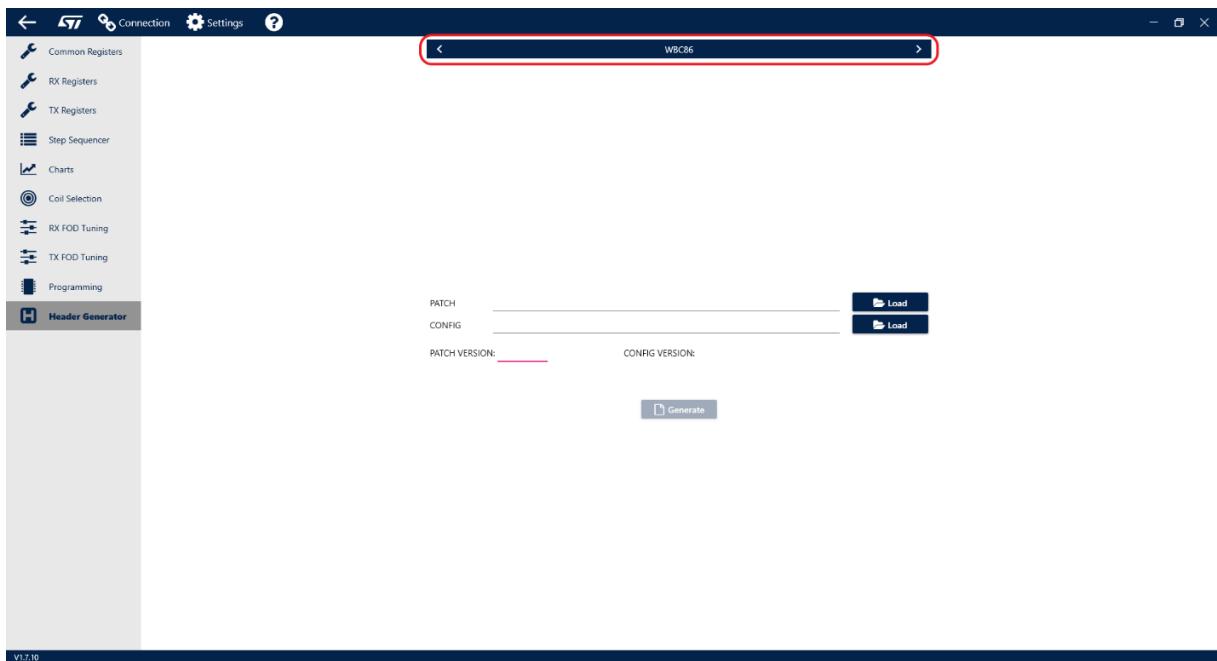
The GUI can also be used to generate a Header file, a binary .h file containing both Configuration and Patch files. The Header file makes programming the device using a host IC easier, as both Configuration and Patch can be loaded at once by simply including the Header file in the host code.

### 3.5

## Header file generation

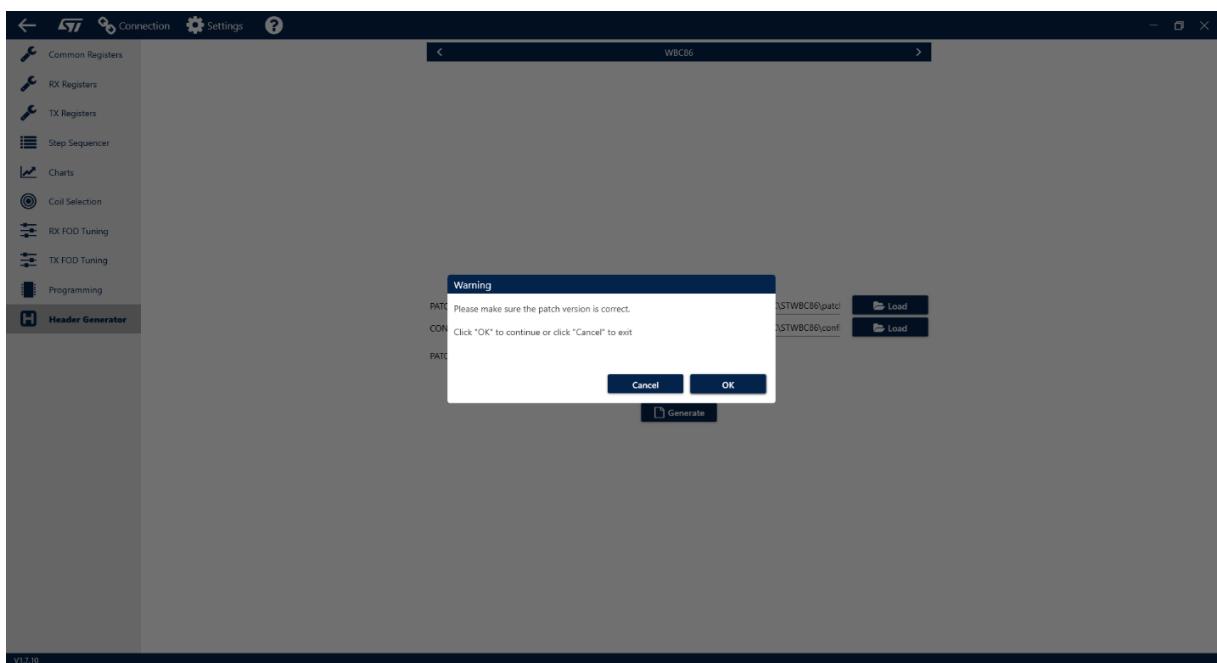
A custom Header file can be generated in the Header Generator tab. Start by selecting WBC86 in the top menu.

Figure 9. Header generator - chip selection



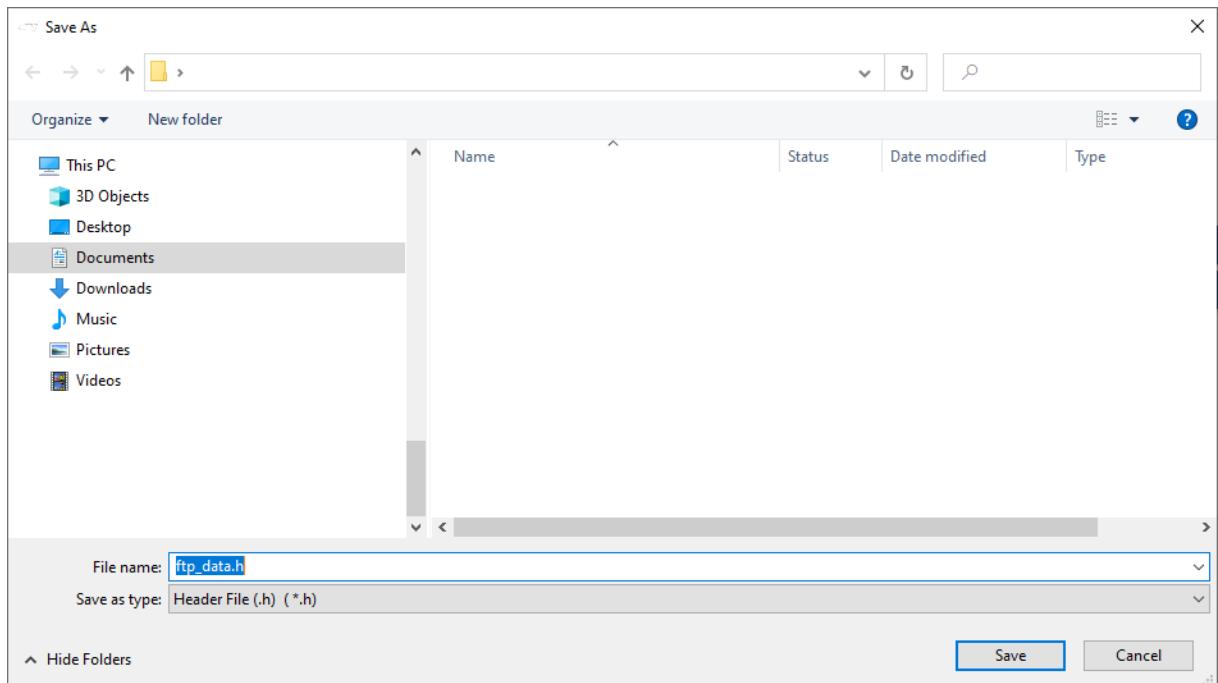
Continue by selecting the Patch and Configuration file and press Generate. A pop-up window appears, asking to confirm the correct Patch version has been selected.

Figure 10. Header generation - pop up window



After confirming the Patch version, the GUI asks for a save destination. After choosing a location, the Header file is saved as a .h file in the selected folder.

Figure 11. Header generation - save header file



### 3.6

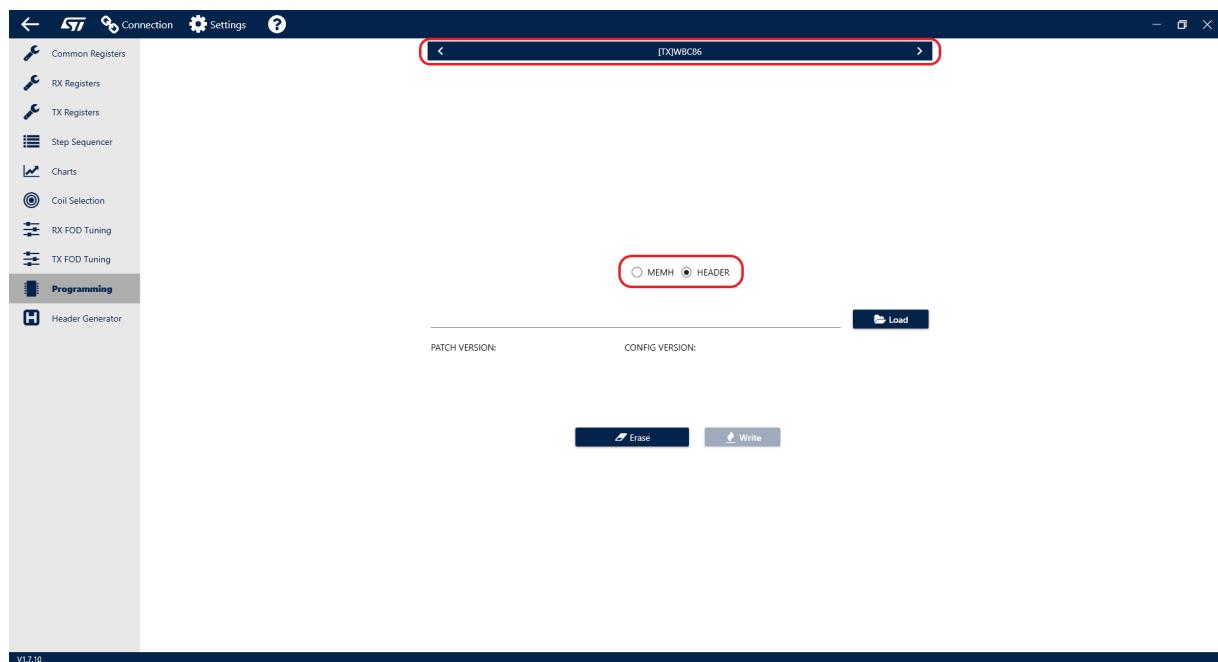
## Programming the device

The device can be programmed in three ways – by changing the register values directly in the **GUI**, by using a **Header** file, which loads both Configuration and Patch files at once, or by loading the two **memh** files separately using the GUI.

The Configuration and Patch files directly modify values stored in the NVM. Therefore, any changes written by Configuration and Patch files will be retained even after reset. On the contrary, changes made by the GUI (Write Tx button) are written into the I2C registers, which are cleared upon reset.

To load the **Header** file using the GUI, navigate to the Programming tab in the side menu. Select WBC86 in the top menu and “HEADER” in the toggle selector.

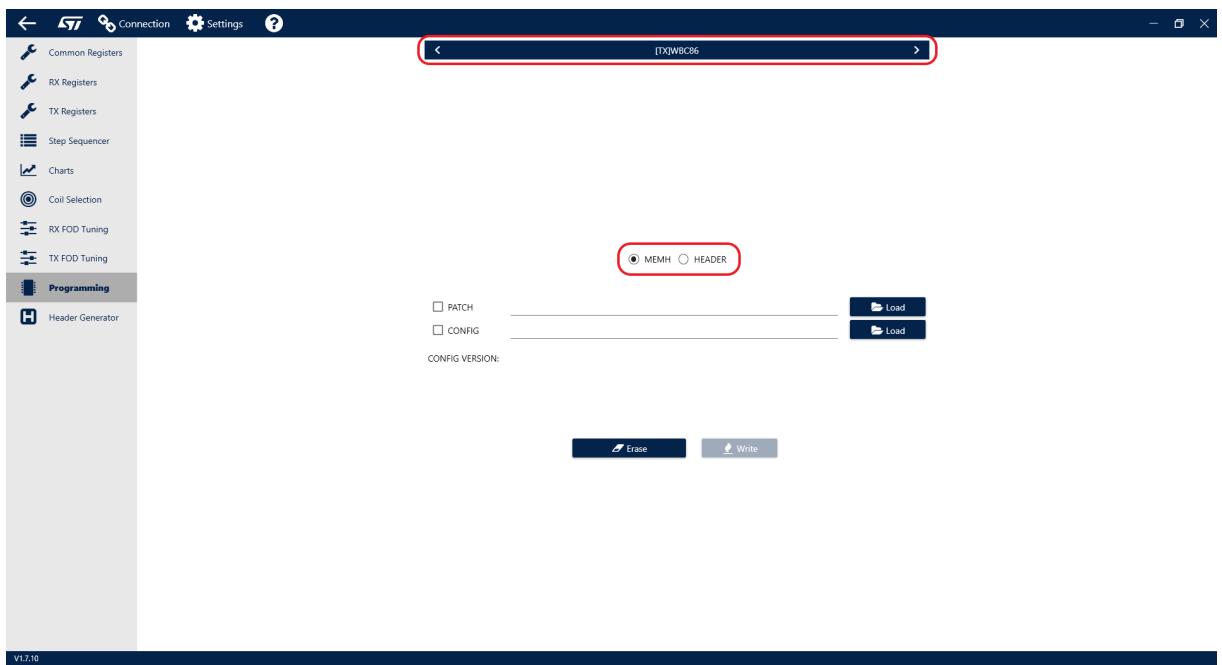
**Figure 12. Programming the device by header file**



Select the .h file that is to be written. The GUI automatically identifies the Patch and Configuration files included in the Header file. Press the “Write” button to load the Header file into the device.

To load the **memh** files (Patch and Configuration) using the GUI, navigate to the Programming tab in the side menu. Select WBC86 in the top menu and “MEMH” in the toggle selector.

Figure 13. Generating the header file by patch and configuration files

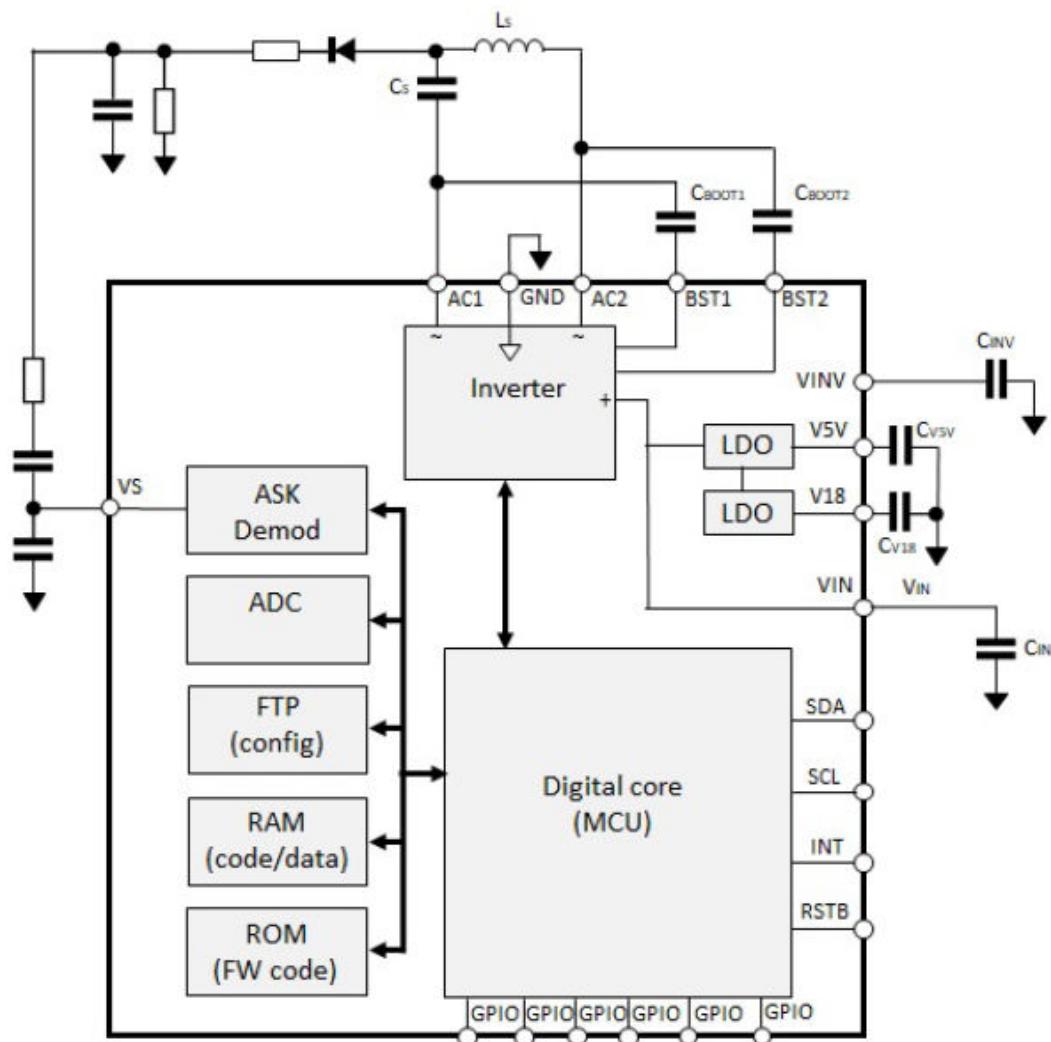


Select Patch and Configuration files that are to be written. Press the “Write” button to load the .memh files into the device.

## 4 Device description and operation

### 4.1 System block diagram

Figure 14. System block diagram

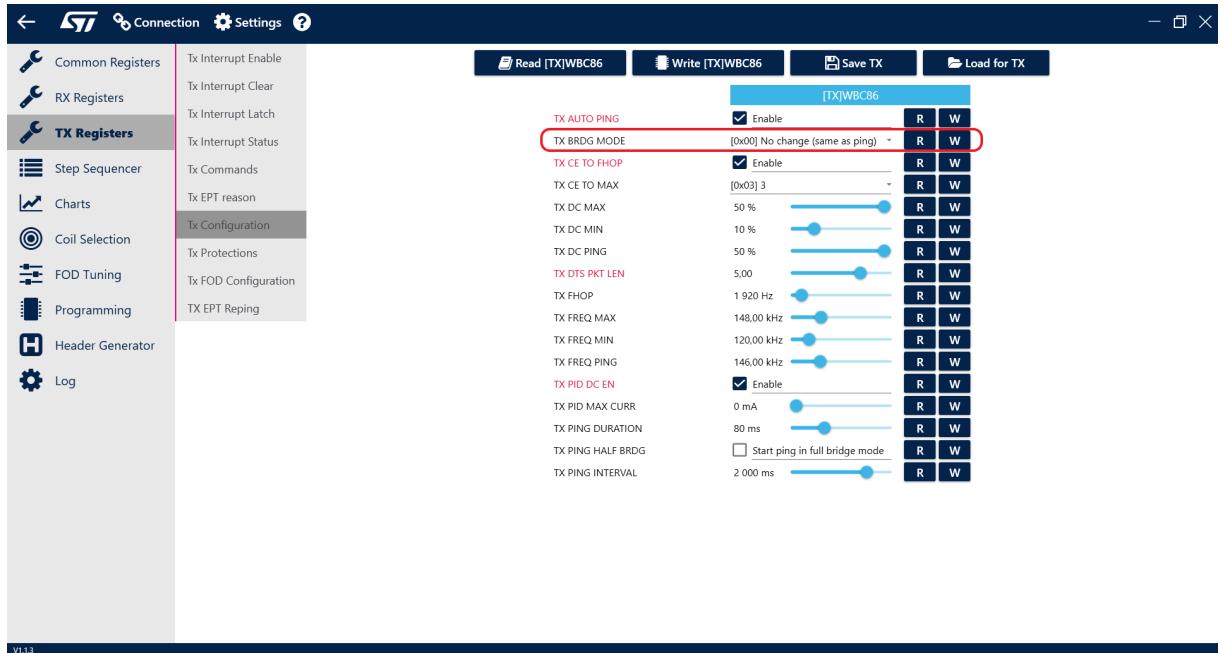


### 4.2 Integrated power inverter

The integrated power inverter is a key block in charge of converting the DC input into an AC power signal for the transmitting coil. The power inverter consists of four N-channel MOSFET transistors arranged into a H-bridge, conveniently driven by an internal control block, which also simultaneously monitors the key parameters of the board to optimize switching and charging the external bootstrap capacitors for the high-side switches.

Some applications may require driving the power inverter in half-bridge mode – for example delivering a very small amount of power might be difficult with some Tx/Rx coil combinations. The STWBC86 can be configured to operate in half-bridge mode using the GUI.

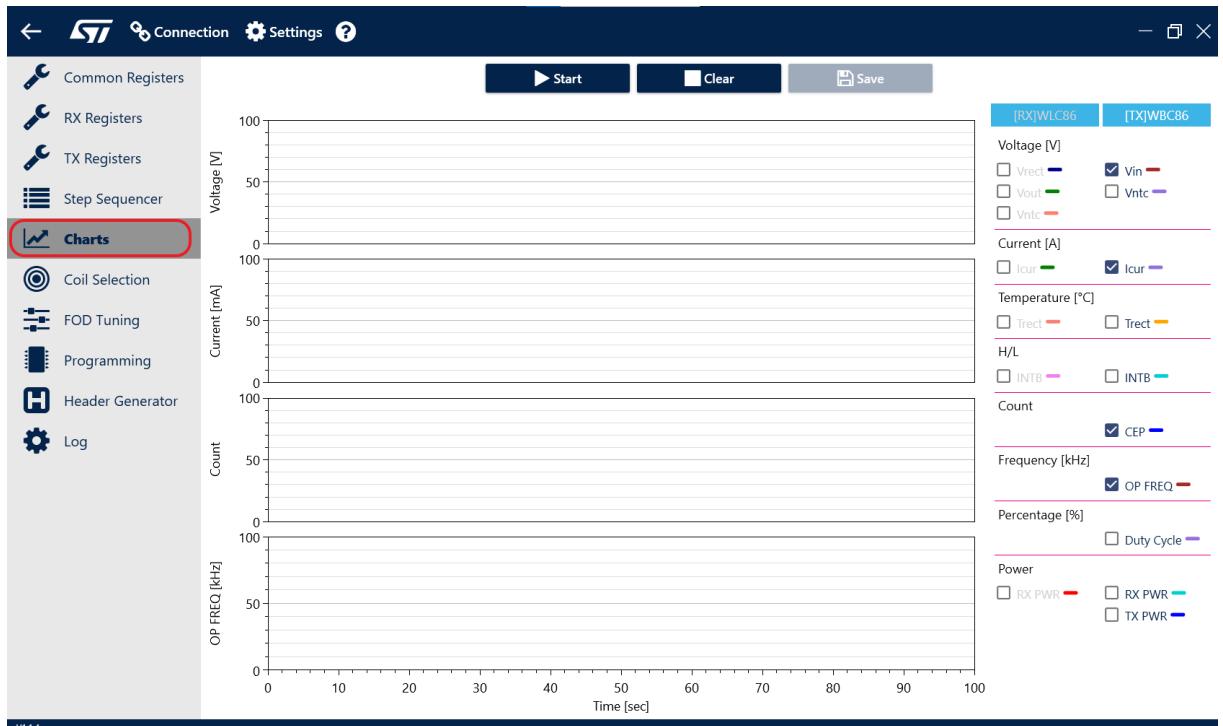
Figure 15. H-bridge mode settings



## 4.3 ADC

The STWBC86 allows the user to monitor key operational parameters using an internal ADC. Instantaneous values can be displayed in the Charts tab of the GUI. The GUI enables the user to monitor the input voltage, input current, device temperature, operating frequency, duty cycle, transmitted power, and more.

Figure 16. GUI charts monitoring



## 4.4

### LDOs

The device is equipped with 2 internal low dropout regulators (LDOs) – a 5 V and a 1.8 V one, with the latter deriving its power from the former.

The 1.8 V LDO powers the digital part of the IC, while the 5 V LDO powers the analog part of the IC but can also be used to power external low-power circuitry (such as LEDs). The maximum current externally drawn from this LDO should not exceed 10 mA.

External LDO capacitors should be placed as close to the IC as possible.

## 4.5

### Power-up sequence

Once power is applied to the input (and the device is not forced into reset), the power-up sequence of the device starts. After the internal main LDO reaches the target output voltage, the digital core of the device starts operating. Default device settings are used until the digital core is woken up after which the firmware loads settings saved in the Configuration file.

If both Patch and Configuration files are loaded into the device and the automatic start function is enabled, the device enters the digital ping phase of power transfer (see [Section 4.11 WPC Qi wireless power transfer](#)). If the automatic start function is disabled, the device does not proceed to the ping phase until the TX\_EN command is executed.

If Patch and/or Configuration files are not loaded, the device stays in a so-called DC mode. In this mode, the device is powered up and ready to be programmed. The device also enters this mode if either the Patch or Configuration, or both files, are corrupt.

## 4.6

### UVLO

The STWBC86 is also equipped with a UVLO function. The UVLO is triggered when the input voltage drops below 2.9 V. The inverter stops switching and the device is powered down. Normal operation is resumed as soon as the input voltage rises above 3 V.

## 4.7

### Chip reset

The device can be forced into reset by pulling the RSTB pin to ground. This can easily be done by a jumper on header P14. When the RSTB pin is released and allowed to be pulled up, the device resumes normal operation.

## 4.8

### Protections overview

The STEVAL-WBC86TX board uses both hardware and software protection to ensure safe voltage and current levels. The purpose of those protections is to avoid damage to either the board itself or even the potential receiver, caused by unexpected conditions – overvoltage and/or overcurrent. The temperature is also monitored, although only software protection is used for this purpose.

The software protections can be enabled/disabled in the GUI; the GUI can also be used to adjust thresholds for the respective protections.

The hardware protections are permanently set and cannot be disabled or adjusted in the GUI. The thresholds for the hardware protections are as follows:

- Overcurrent protection: 3 A (fuse)
- Overvoltage protection: 22 V (TVS diode)

The triggering of a software protection results in the transmitter terminating a power transfer and generating a corresponding interrupt (can be configured in the GUI).

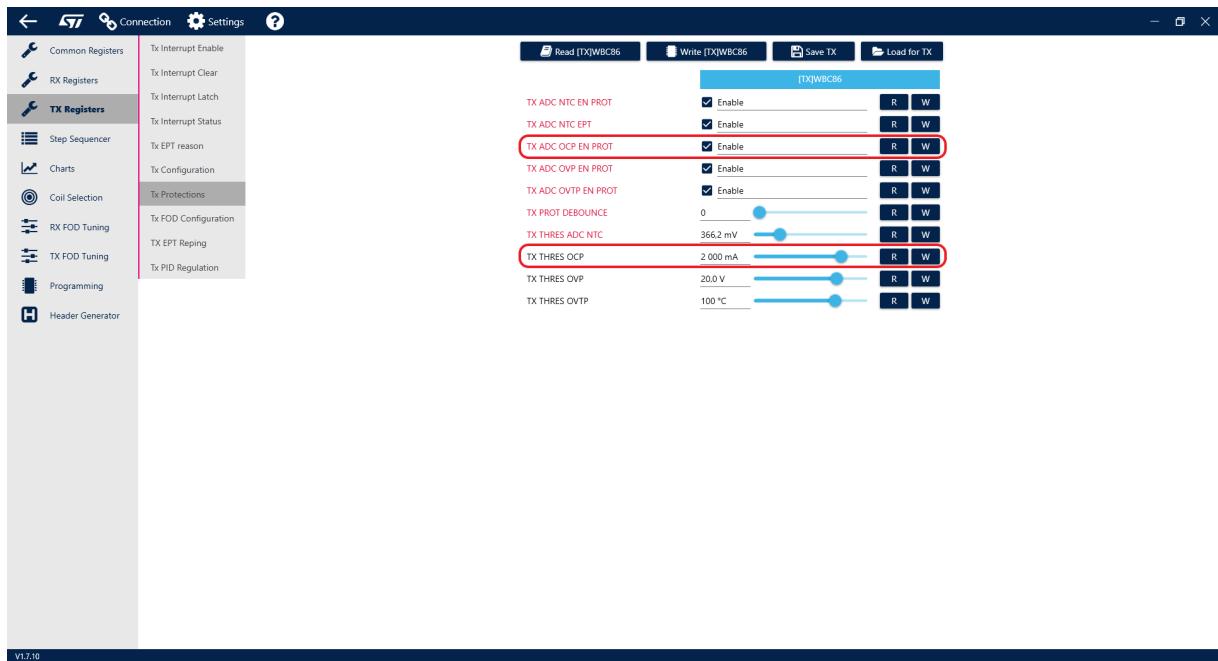
## 4.8.1

### Overcurrent protection (OCP)

A transmitter overload or a short on the output (transmitting coil) may lead to excessive input current values. To prevent damage to the transmitter caused by such currents, two separate protections (hardware and software) are implemented.

A fuse (F1) on the input track rated at 3 A serves as the hardware protection, while an ADC monitoring the input current serves as the software one. If the input current exceeds a set threshold, the power transmitter terminates the power transfer and generates an OCP interrupt. The threshold is configurable in the GUI and can be set in a range of 0 to 2500 mA.

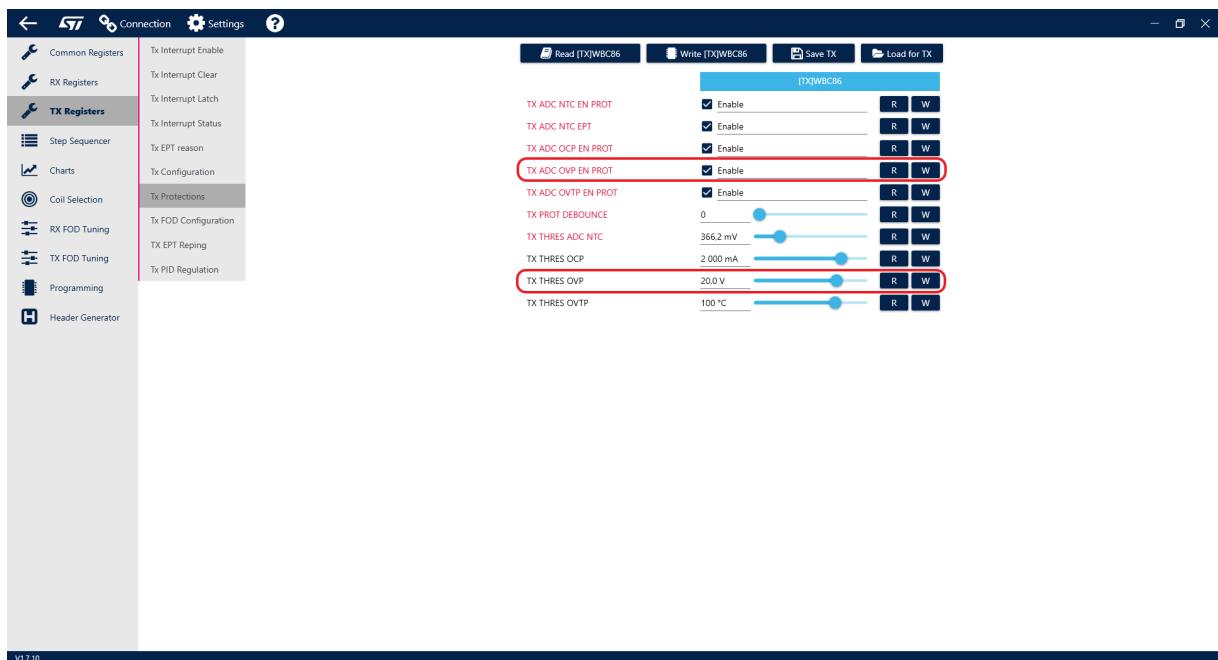
Figure 17. OCP settings



#### 4.8.2 Overvoltage protection (OVP)

Excessive input voltage may damage the board and/or device. For this reason, a TVS diode is placed at the input of the board. The IC is also equipped with an ADC dedicated to monitoring the input voltage level. The protection can be enabled/disabled in the GUI and the threshold can be set in a range of 0 to 25.5 V. Triggering OVP also leads to power transfer termination, as an increase in transmitter input voltage may also lead to an increased Rx VRECT voltage.

Figure 18. OVP settings

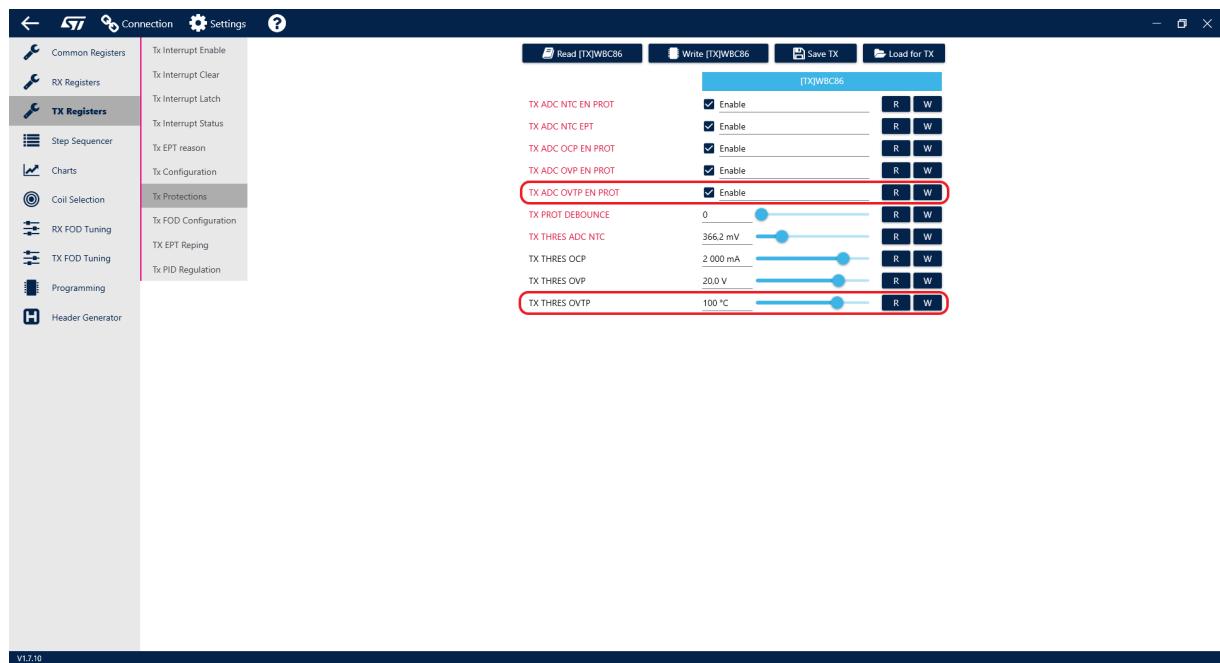


#### 4.8.3

#### Overtemperature protection (OVTP)

The temperature of the IC is continuously monitored by a temperature sensor. Excessive IC temperature may indicate that either the operating power is too high, or an internal fault occurred. It should be considered that PCB design may affect thermal performance as well. If the temperature exceeds a set threshold, power transfer is terminated, and an OVTP interrupt is generated by the transmitter. The threshold is configurable in the GUI and can be set in a range of 0 to 151 °C.

Figure 19. OVTP settings



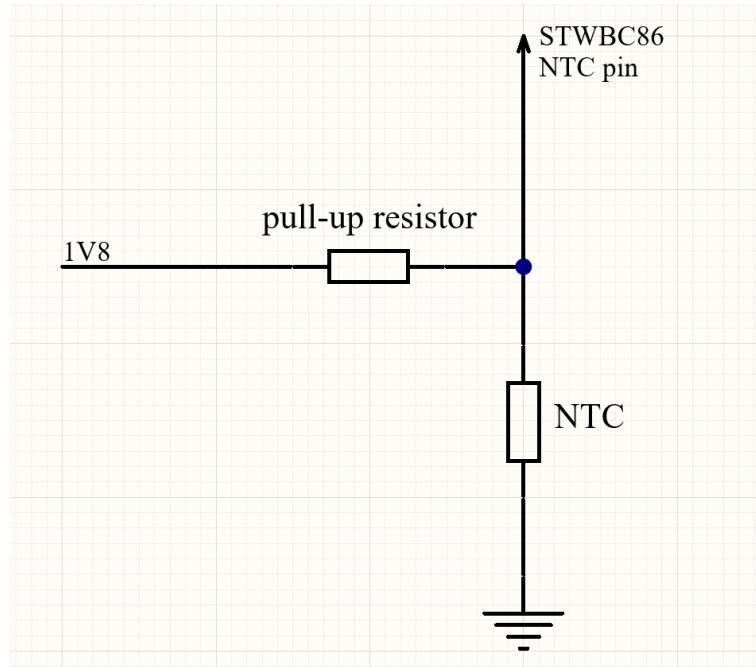
#### 4.8.4

#### NTC

An external NTC can be used to further monitor the operational temperature of the board. The user may choose a component/board region to be monitored by placing the NTC on it/nearby, although monitoring the transmitting coil is presumably the most common practice.

Together with a pull-up resistor, the NTC forms a voltage divider, which is connected to the STWBC86 NTC pin. The NTC pin is 1.98 V tolerant, although an internal ADC supports only up to 1.5 V NTC voltage reading. Therefore, it is recommended to design the voltage divider to reach the (low) NTC threshold at the highest allowed temperature, while leaving a large enough margin for accurate temperature monitoring (the NTC voltage should be kept below 1.5 V).

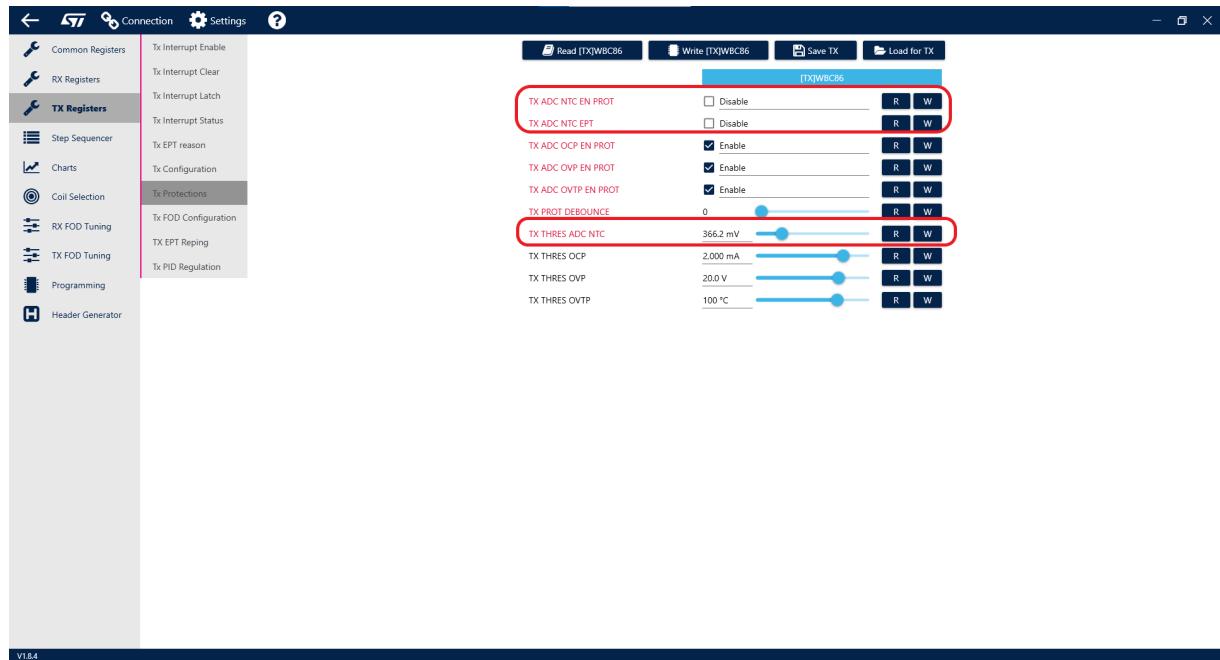
Figure 20. NTC connection on board



The user can choose if triggering this protection only generates a corresponding interrupt, or if the device also terminates the power transfer upon triggering. The interrupt must be enabled when power transfer termination is desired.

To set the NTC threshold using GUI, the threshold value must be set (in mV) based on the parameters of the resistor divider. The protection will be triggered when the divider output voltage drops below the set threshold.

Figure 21. NTC settings



## 4.9

## Foreign object detection (FOD)

A foreign object is any object placed either on or near the transmitting coil, which is not considered a valid wireless power receiver and is magnetically active. Presence of the magnetic field generated by the transmitting coil may cause eddy currents to form in the foreign object (such as coins, keys etc.), which in turn would heat the object to potentially dangerous temperatures. To avoid possible damage to the device or even injury to the user, the power transmitter must be able to detect the presence of a foreign object. This detection can be implemented in several different ways.

One of the most common ways is estimating the amount of power lost in the system. The power receiver indicates the total amount of received power from the power transmitter. This power consists of power available to the load and power loss, which occurs on the supporting circuitry.

A high power loss value could indicate the presence of a foreign object.

The STWBC86 uses precise AD converters and 32-bit arithmetic to monitor input power and can therefore estimate the amount of power lost beyond the power transmission interface using a mathematical model of the system.

### 4.9.1

### FOD tuning

The default FOD parameters are optimized for STEVAL-STWBC86TX. Any change to the topology might require adjustments to the FOD parameters too. The user can record these parameters using the GUI to properly trigger FOD when required, as well as avoid false triggers.

The most common use conditions include:

1. Aligned position without any FO.
2. Misaligned position without any FO. Parameters should be recorded in a few various misalignments, including the maximum expected one - usually 5mm misalignment in all x, y, and z axis respectively (excluding their combinations).
3. Aligned position with FO - reference foreign objects prescribed by the Qi specification, coins, keys etc. Each FO requires a separate measurement.
4. Misaligned position with FO.

The FOD parameters should be recorded for the whole range of Rx loads.

Before the actual tuning, please disable the EPT FOD function of STWBC86TX in the TX registers tab. Leaving the function enabled would result in termination of the power transfer during parameter logging, caused by the presence of FO. However, for the change to take an effect, you will have to generate and load a new CFG file with the feature disabled, as this setting cannot be changed while the device is running.

Figure 22. Disable FOD detection

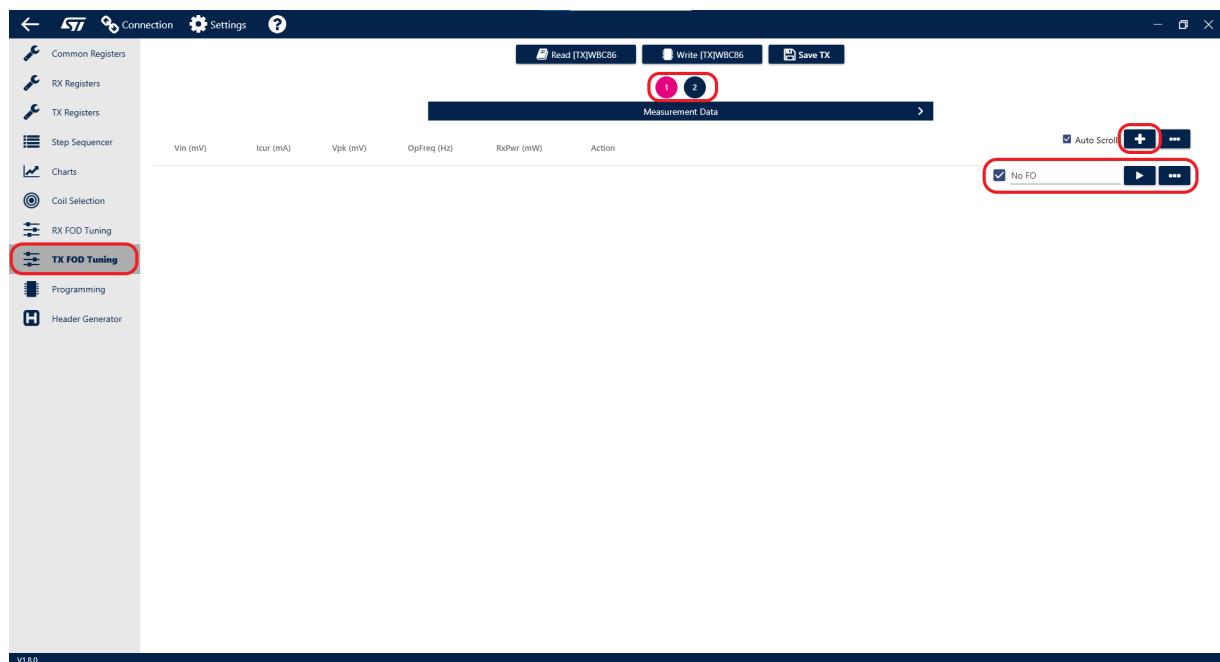


## Data collection

To collect accurate data, the ring node voltage divider must be set accordingly, please refer to chapter 6.2 Ring node voltage sensing for more information.

To start the tuning process, head to the TX FOD tuning tab of the GUI. To add a new data set, press the '+' button on the right side of the screen. The data set can be labelled using the name line below to help better distinguish it later.

Figure 23. Add a new dataset



Ticking the check box next to a data set will toggle its display in the plot on the next page of the tool. If you want to hide a particular data set, simply untick the corresponding check box.

Figure 24. Display dataset in plot



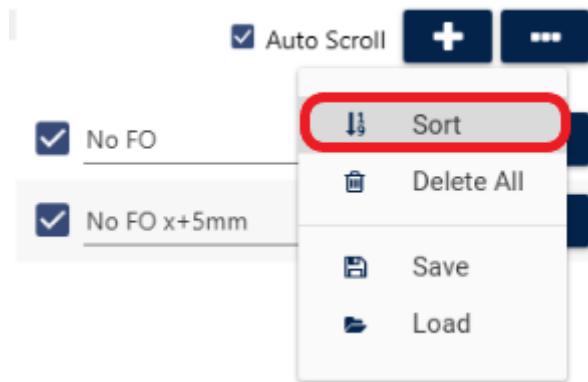
The newest data is always added to the bottom of the set. When enabled, the Auto Scroll function will automatically take you to the end of the list after logging a new set.

Figure 25. Enable auto scroll



Additional options can be found by clicking the '...' button next to the Auto scroll box.

Figure 26. Additional dataset options



The first button in the drop-down menu will sort the captured data by input current. Always sort the data after the capturing is finished. Otherwise, the load curve might not display correctly on the second page of the tool.

The second button will delete all data. The third will enable you to save the data in a .txt file format and the last lets you import any previously captured data. The loaded data will be appended to the end of the list.

Click the '...' button next to any data set to access its properties. Here, you can adjust the line style used in the plot and the display colour for the data set. A solid line is used by default for each data set but can be switched to a dashed line by ticking the line style box. This can be used to distinguish different conditions, such as sets with and without a foreign object.

Figure 27. Line properties - dashed/solid

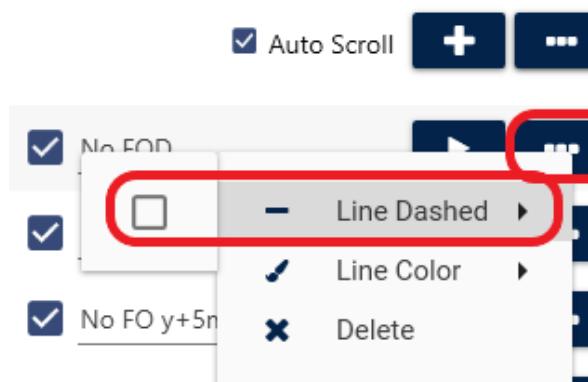
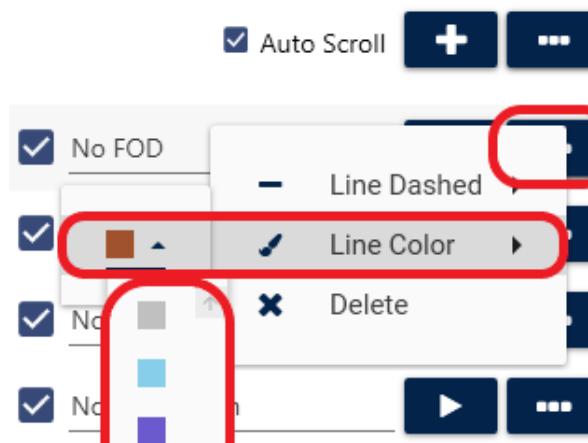


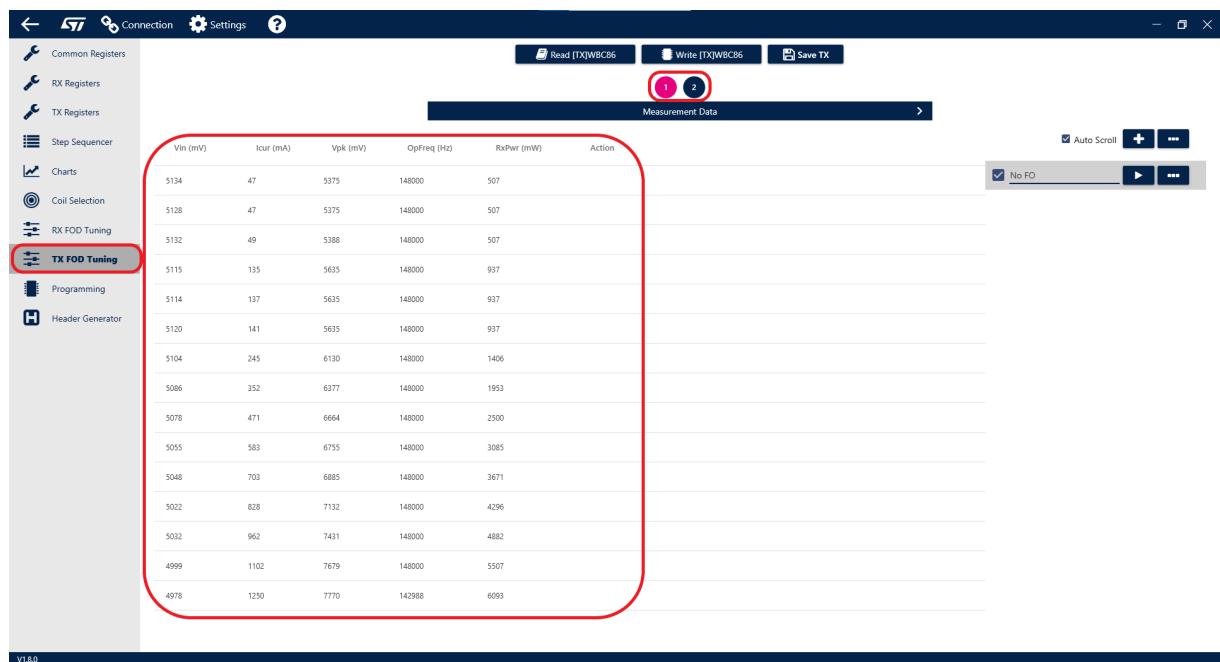
Figure 28. Line properties - color



Before capturing a new set of data, prepare the intended operating conditions (FO, misalignment) and allow power transfer to be established, then click the start button.

The GUI will then begin to periodically log the operating conditions and the logged data will be shown in a table in the middle of the GUI window. We recommend recording data for the whole load range in appropriate steps. It is also recommended to record at least three data points for each load, as the operating parameters may vary slightly – mostly because of noise.

Figure 29. Logged data



Vin (mV)	Icur (mA)	Vpk (mV)	OpFreq (Hz)	RxPwr (mW)	Action
\$134	47	5375	148000	507	
\$128	47	5375	148000	507	
\$132	49	5388	148000	507	
\$115	135	5635	148000	937	
\$114	137	5635	148000	937	
\$120	141	5635	148000	937	
\$104	245	6130	148000	1406	
\$086	352	6377	148000	1953	
\$078	471	6664	148000	2500	
\$055	583	6755	148000	3085	
\$048	703	6885	148000	3671	
\$022	828	7132	148000	4296	
\$032	962	7431	148000	4882	
4999	1102	7679	148000	5507	
4978	1250	7770	142988	6093	

One of the variables used to estimate foreign object presence is received power, reported by the RX via received power packet (labelled RxPwr in the table). This packet is usually only sent once every 1500 ms. It is therefore necessary to wait for about 1.5 seconds after changing the load to allow the RPP value to be updated before reading. We recommend using the following procedure:

1. Record a few samples on the first load step.
2. Pause the recording, change the load and wait at least 1.5 s.
3. Optional: check if the recorded samples are all acceptable.
4. Resume the recording.
5. Repeat until enough load steps are recorded.

Data points containing the old RPP value can be distinguished by comparing both the RxPwr and Icur values to the previous points. If a sample contains an RPP value similar to a previous point (load step), but the current is significantly different, it is most likely caused by desynchronization of the current and the received power reading. The desynchronized data points can also be spotted in the plot, as those points will usually be significantly higher/lower than the rest.

Any unsuitable data point can be deleted using the "X" button in the Action column.

**Figure 30. Delete a single data point**

Vin (mV)	Icur (mA)	Vpk (mV)	OffFreq (Hz)	RxPwr (mW)	Action
5129	48	5518	148000	507	
5143	49	5518	148000	507	
5114	133	5544	148000	507	
5116	136	5635	148000	937	
5122	137	5700	148000	937	
5122	137	5700	148000	937	
5121	138	5700	148000	937	
5107	243	6195	148000	1406	
5105	245	6169	148000	1406	
5107	245	6208	148000	1406	
5083	356	6442	148000	1992	
5083	361	6442	148000	1992	
5079	362	6416	148000	1992	
5080	473	6690	148000	2539	
5072	473	6716	148000	2539	
5079	473	6716	148000	2539	

**Figure 31. Spot wrong data point in plot**


### Parameter tuning

Plot of the logged data can be seen on the second page of the tuning tool, accessed by clicking the "2" button in the upper part of the window. Here you can see estimated power loss curves based on the data captured.

Figure 32. Plot logged data



The logged data are processed before being plotted, as the tool also takes into consideration a few physical properties of the circuit. Those properties are set in the top part of the parameter list on the right side of the window and include:

- TxSwitchLossCoef** selects if the switch loss should be included in the calculation.
- CoilResistance** refers to the STWBC86TX coil resistance.
- Cser** and **CserResistance** refer to the STWBC86TX series resonant capacitor capacitance and resistance values.

Values listed below are the tuneable parameters of the FOD algorithm.

**VcpPeakDrop** is a ring node voltage value modifier. **CoilCoef** is a scaling factor, which incorporates frequency dependent losses into the calculation. Adjust the **VcpPeakDrop** and **CoilCoef** values until the foreign object and non-foreign object curves are as separate as possible. **VcpPeakDrop** can be set in steps of 50 mV, **CoilCoef** can be set in steps of 1.

**PlossThreshold** sets a threshold for foreign object detection. If the calculated power loss is higher than the set threshold, FOD protection is triggered. **PlossThreshold** can be set in steps of 32 mW.

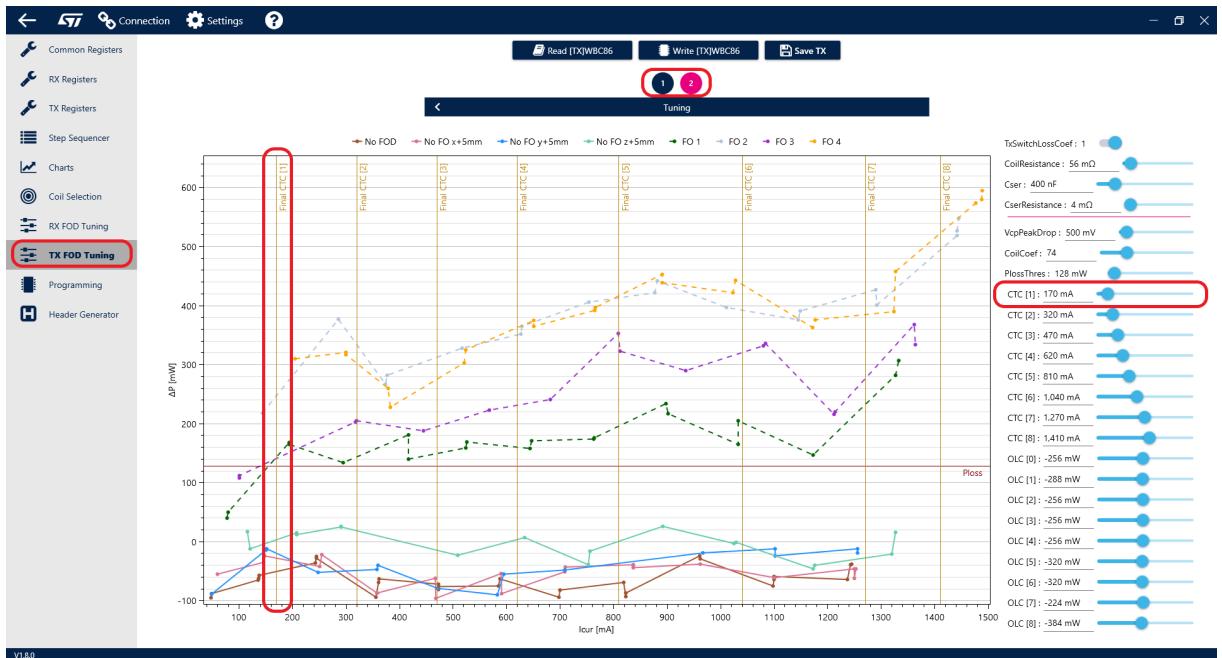
**Figure 33. Ploss threshold**



After adjusting all previously mentioned parameters, the curves can be further adjusted by setting power loss offsets for selected input current intervals. Begin by defining the intervals using the CTC values. Those values refer to the respective input current thresholds and are displayed as vertical lines in the plot. A unique offset can be applied to every interval using the corresponding OLC slider. Please note that the first OLC value corresponds to the interval ranging from zero input current to CTC [1], the second corresponds to interval "CTC [1] to CTC [2]", and so on, with the last OLC value corresponding to interval "CTC [8] and above".

Remember, that the offsets adjust the relative position of ALL curves to the PlossThreshold, they cannot be used to separate the FO and non-FO curves. Only VcpPeakDrop and CoilCoef can be used to separate the curves.

**Figure 34. CTC threshold**



The goal of FOD tuning is to establish a threshold which would both trigger FOD when foreign object is present and not trigger FOD when not. This would translate to a condition when all FO curves are above the set threshold and non-FO curves are below the threshold, with a sufficient margin (at least for the non-FO curves). The margin is necessary to ensure the power transfer in not terminated, even when there is a slight misalignment of the coils.

Under some condition, the power dissipated in the foreign object might be very low, which would cause the FO and non-FO curves to be very close to each other. This would make it almost impossible to separate the curves (and to detect the presence of foreign object) effectively. However, the main reason of FOD is preventing excessive heating of the FO. A low power dissipation would lead to only a minor temperature rise of the FO.

Certain value combinations may not be suitable. To help you avoid problems during operation, non-recommended coil coefficient, VcpPeakDrop and OLC values will be highlighted in red. Saving and uploading those values may cause false FO detection during operation.

**Figure 35. Unsuitable parameter values highlighted in red**



### Parameter loading

The new FOD parameters must be loaded into the chip when the tuning is finished. However, before loading the parameters, please remember to enable the EPT FOD function, you have disabled as per the instructions at the beginning of this guide.

To load the parameters into the device, begin by saving the parameters using the “Write TX” button. Continue by navigating to the TX Registers, Tx FOD Configuration and enable FOD. Optionally, adjust the FOD debounce value. Click the “Save TX” button to generate a new CFG file containing the new parameters. Finish by loading the CFG file into the device using the standard procedure described in section 3.6.

**Figure 36. Load tuned parameters**

**Figure 37. Enable FOD and set FOD debounce**

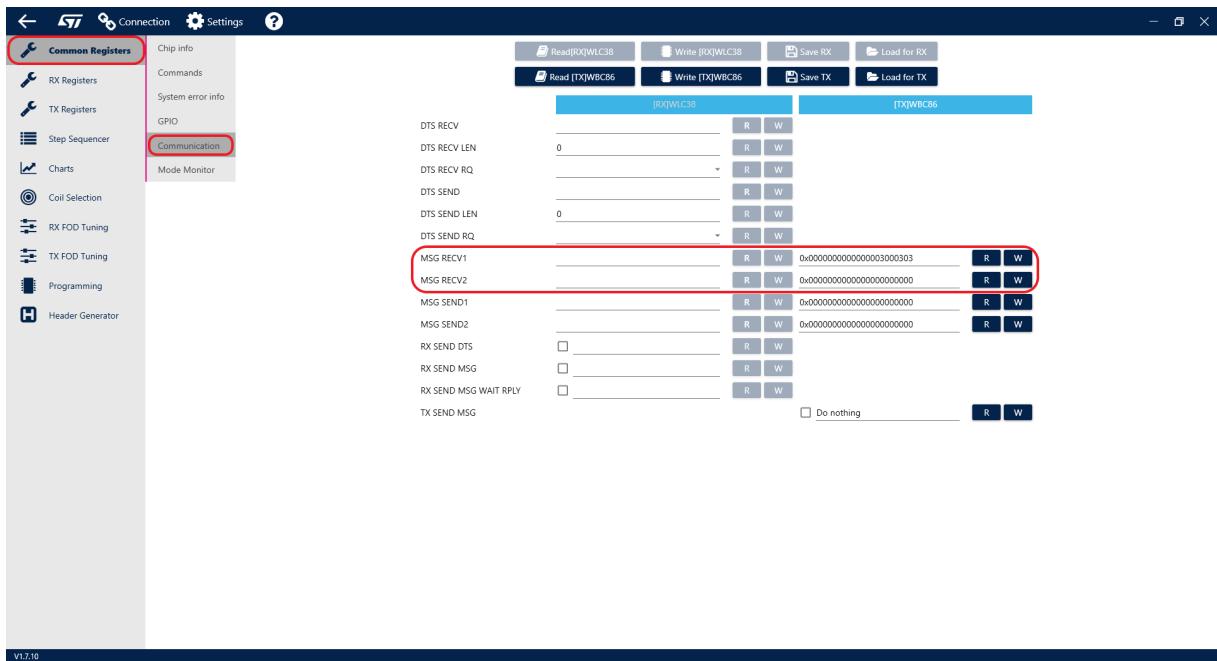

## 4.10 EPT reason

Power transfer may be terminated for several different reasons. Determining the exact cause might be difficult, as more than one fault condition might have been met. For this reason, the power transmitter is equipped with EPT reason registers. Those registers indicate the state of various parameters, which might be responsible for the termination. It should be noted that, as mentioned above, multiple bits (reasons) might be set to one after a power transfer termination occurs.

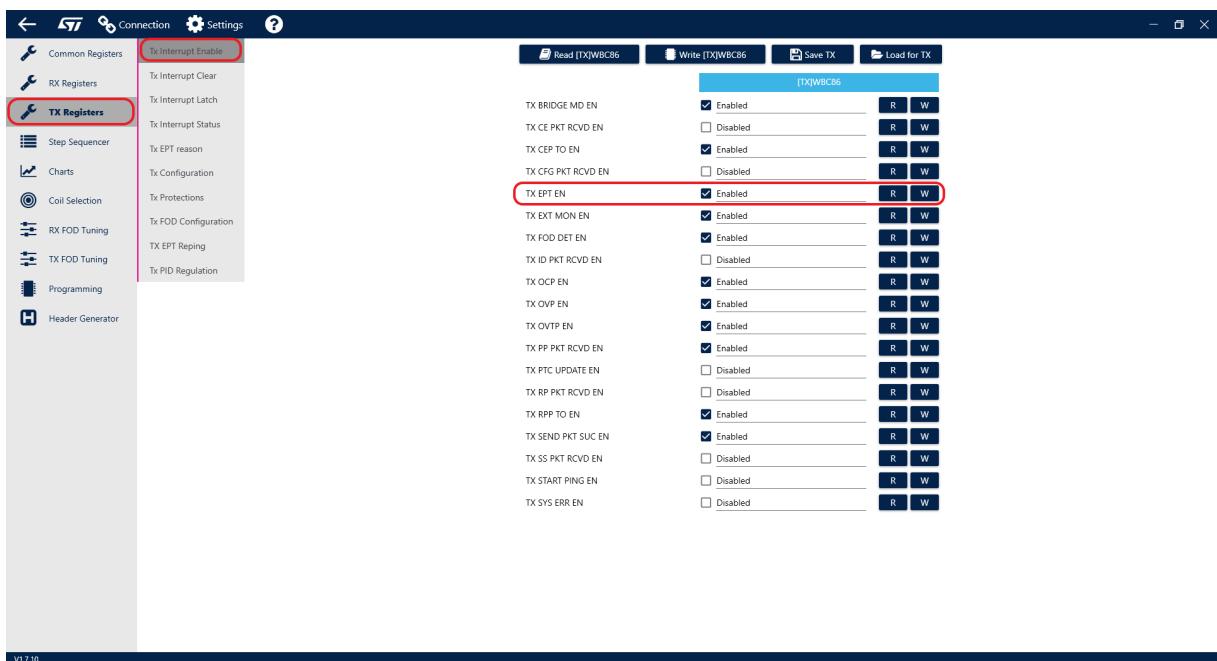
Figure 38. EPT reason

	[TX]WBC86
TX CEP TO	Did not appear R W
TX CFG	Did not appear R W
TX CFG CNT	Did not appear R W
TX EXT MON	Did not appear R W
TX FOD	Did not appear R W
TX HOST	Did not appear R W
TX ID ERR	Did not appear R W
TX NEGO ERR	Did not appear R W
TX NEGO TO	Did not appear R W
TX OCP	Did not appear R W
TX OVP	Did not appear R W
TX OVTP	Did not appear R W
TX PCH ERR	Did not appear R W
TX RPP TO	Did not appear R W
TX RX EPT	Did not appear R W
TX RX RST	Did not appear R W
TX SS ERR	Did not appear R W
TX SS TO	Did not appear R W
TX SYS ERR	Did not appear R W
TX XID ERR	Did not appear R W

The power receiver may generate an EPT request as well and send it to the transmitter. If the transmitter receives an EPT packet from the receiver, it terminates the power transfer immediately. The EPT packet that the receiver sends should also contain the reason for the request. The reason can be retrieved from the communication and decoded by the user/host. Contents of the packet can be found in the buffer (MSG RECV1&2), which can be found in the communication section of the Common Registers tab.

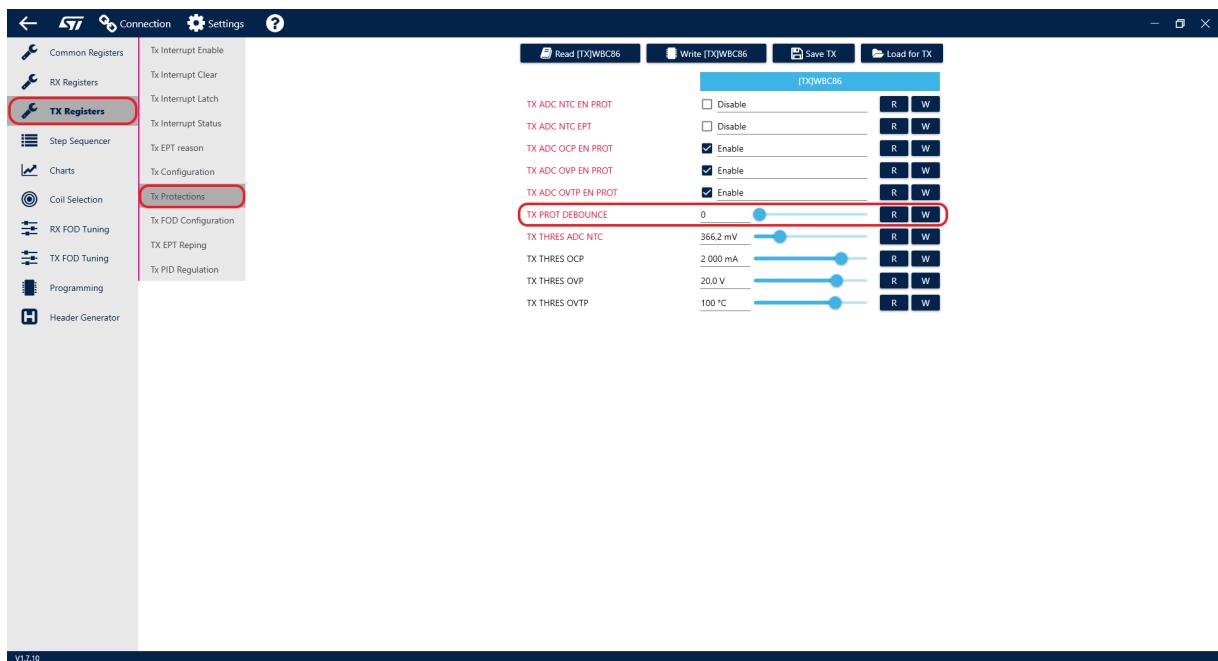
**Figure 39. Received message**


However, the values in the buffer are updated frequently, therefore reading the RX EPT reason requires using a host controller triggered to an RX EPT packet interrupt.

**Figure 40. EPT interrupt setting**


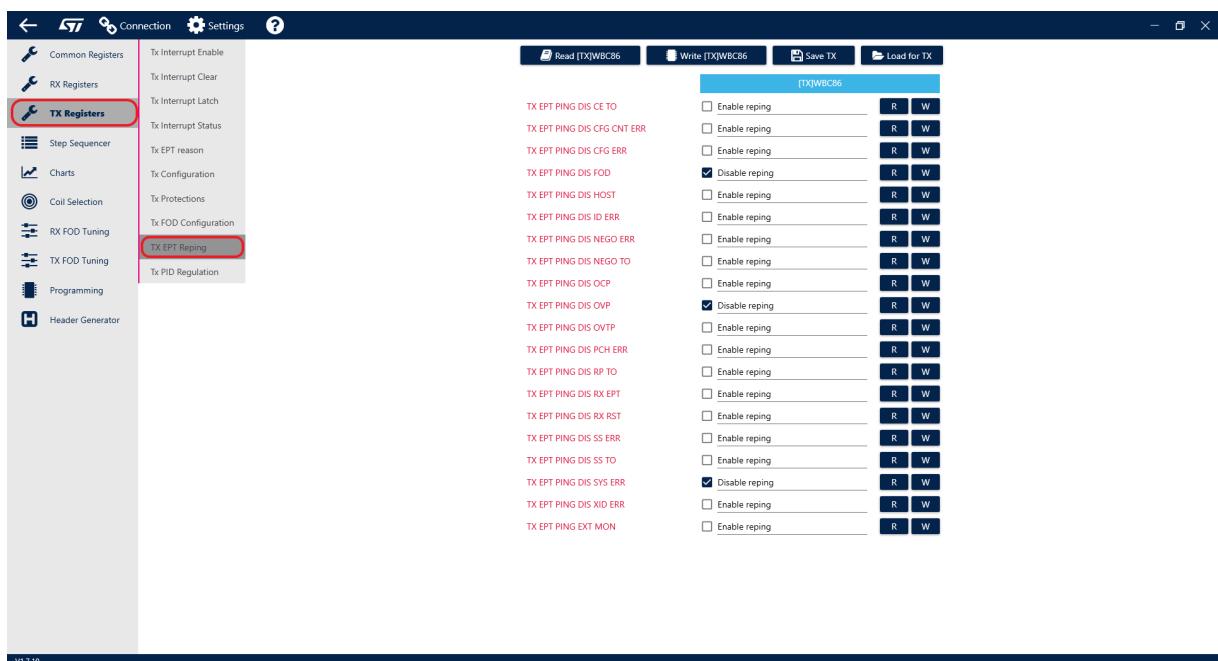
The GUI enables the user to set a debounce value for EPT conditions. This means that the device does not terminate the power transfer immediately upon registering an EPT condition, but rather after registering a number of conditions greater than the specified value. As an example: when the debounce is set to 2 and the device temperature exceeds the set overtemperature threshold, the device terminates power transfer only after **3** consecutive temperature measurements exceed the threshold.

**Figure 41. Protection debounce setting**



The GUI enables the user to select which (EPT) condition causes the transmitter to stop pinging even after the EPT condition no longer applies. When used, this feature keeps the device inactive until it is restarted (by a power cycle) by the user. In addition, entering the inactive state can be indicated by assigning a TX error (0x1E) function to one of the GPIO pins (see [Section 4.14 GPIOx and INTB pins](#)).

**Figure 42. EPT reping settings**



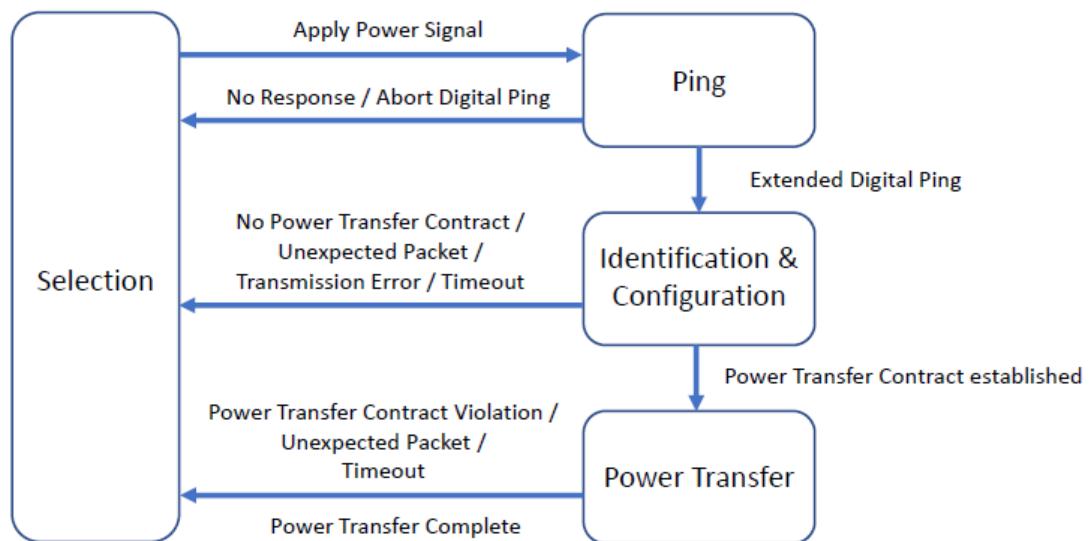
Checking a box causes the corresponding condition to block the device from pinging after it is triggered.

## 4.11

## WPC Qi wireless power transfer

The flowchart in the figure below shows the steps required to reach power transfer in the Baseline Power Profile (BPP) according to Qi 1.2.4.

Figure 43. Power transfer start up sequence



**Note:** For more details refer to: *The Qi Wireless Power Transfer System Power Class 0 Specification, Parts 1 and 2: Interface Definitions, Version 1.2.4 February 2018.*

- Digital ping: this phase is an interrogation session during which the potential power receiver is expected to reply through amplitude shift-keying (ASK) modulation as defined by the Qi specification. After a valid power receiver is detected, the transmitter proceeds to the Identification & Configuration phase.
- Identification & Configuration: this phase aims to identify the receiver and to gather information necessary for a stable and reliable power transfer, such as the maximum power or FSK communication parameters. After receiving the information, the power transmitter creates a Power transfer contract; basically a summary of the operational parameters. After the power receiver is identified and the power transfer is created, the power transmitter proceeds to the power transfer itself.
- Power Transfer: this is the final step. The power transmitter initially increases and subsequently modulates the transmitted power in response to control (feedback) data from the receiver. The receiver periodically sends information to the power transmitter, such as required power (CEP), received power (RPP), etc., which are used to maintain a closed control loop. If a critical event (for example, overvoltage, overcurrent or overtemperature) occurs, the power transmitter terminates the power transfer immediately.

#### 4.11.1

#### Wireless power interface

Wireless power interface is the area in which power transfer takes place. It consists of two parts – the transmitter (primary) power interface and the receiver (secondary) power interface. The main component of the interfaces is the transmitting/receiving coil. The user should avoid placing any objects which are magnetically active, yet are not required for the power transfer itself, on or near the wireless power interface in order to avoid possible damage or injury (see [Section 4.9 Foreign object detection \(FOD\)](#)).

Detection of a valid receiver on the primary wireless power interface is performed by a digital ping. A digital ping is a burst of power generated by the transmitter; its parameters are defined by the Qi specification. The target of this burst is to generate enough power for the receiver to establish communication between the transmitter and the receiver.

The Qi specification defines a specific way that the receiver is expected to answer the digital ping. The reason is that only a proper answer can inform the transmitter that a valid power receiver was placed on the wireless power interface.

#### 4.12

#### Bidirectional communication

The amount of power transmitted is (from a control standpoint) fully dependent on the operating conditions of the power transmitter (that is, its bridge voltage, operating frequency and duty cycle).

Since there is no direct electrical feedback from the load (system output) to the power transmitter, the power receiver must establish communication with the power transmitter to provide this feedback instead.

To match load requirements and prevent excessive power transmission, the power receiver communicates to the power transmitter the required power level using amplitude shift keying (ASK). This is done by the receiver modulating the amount of power that it draws from the power signal. The transmitter detects this as a modulation of the voltage across and/or current through the transmitting coil and adjusts its operating conditions accordingly.

The power transmitter can communicate to the receiver as well, this time using frequency shift keying (FSK). This is done by the power transmitter directly modulating the power signal. Communication in this direction enables proprietary communication.

STWBC86 supports proprietary packet FSK reply on proprietary ASK packets from Rx. This feature is requiring control from the external host MCU on Tx side and details can be found in dedicated Application Note about bidirectional communication between STWLC38 and STWBC86.

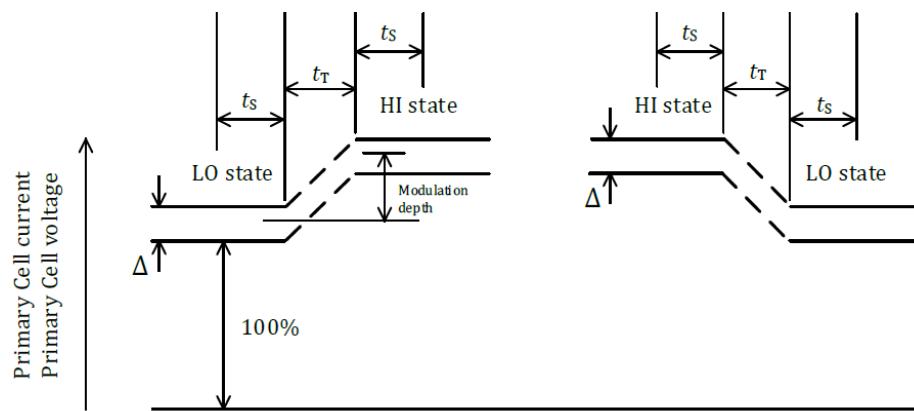
#### 4.12.1 ASK communication

A state (either high or low) is characterized by the amplitude being constant (with a certain variation  $\Delta$ ) for at least 150 ms.

If the power receiver and the power transmitter coils are properly aligned, then for all appropriate loads at least one of the following three conditions apply.

- The difference of the amplitude of the transmitter coil current in the high and low state is at least 15 mA
- The difference of the amplitude of the transmitter coil voltage in the high and low state is at least 200 mV
- The difference of the transmitter coil current in the high and low state is at least 15 mA. The transmitter coil current is measured at instants in time that correspond to one quarter of the cycle of the control signal driving the half-bridge inverter

**Figure 44. Amplitude modulation of the Power Signal**



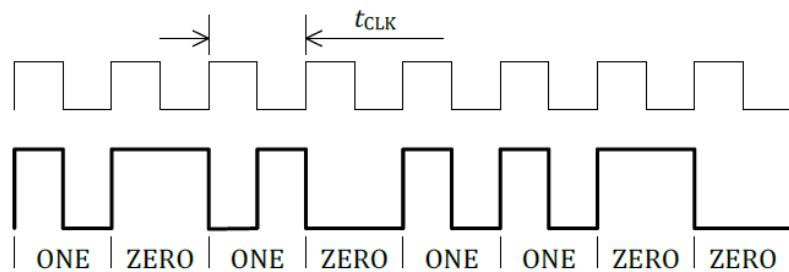
Parameter	Symbol	Value	Unit
Maximum transition time	$t_T$	100	$\mu s$
Minimum stable time	$t_s$	150	$\mu s$
Current amplitude variation	$\Delta$	8	mA
Voltage amplitude variation	$\Delta$	110	mV

**Note:** For more details refer to: *The Qi Wireless Power Transfer System Power Class 0 Specification, Parts 1 and 2: Interface Definitions, Version 1.2.4 February 2018.*

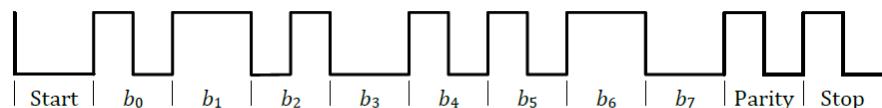
The power receiver uses a differential bi-phase encoding scheme to modulate the data bits onto the power signal. The power signal is then demodulated by the power transmitter's external demodulation circuit and the demodulated signal is fed into the VS pin of STWBC86 for processing.

The communication itself is carried out via packets. Each packet consists of 4 parts and each of those parts consists of several bits/bytes.

Each communication bit is synchronized to the power receiver's internal 2 kHz clock signal period. A ONE bit is encoded as two transitions of the power signal, while a ZERO bit is encoded as a single transition.

**Figure 45.** Example of a differential bi-phase encoding scheme

To transmit a single **byte** of data, the power receiver must send an 11-bit sequence. This sequence consists of a START bit (ZERO), the 8 data bits of the byte (LSB first), a parity bit and a STOP bit (ONE). The parity is odd, meaning an even number of ONE bits in the data byte results in the parity bit being equal to ONE, while an odd number of ONE bits in the data byte results in the parity bit being equal to ZERO. An example of such message can be seen in the image below:

**Figure 46.** Example of the asynchronous serial format

A **packet** consists of a preamble, header, message, and checksum. The preamble contains 11 to 25 ONE bits and enables the power transmitter to synchronize with the incoming data. The header, message, and checksum are sequences of three or more bytes. The header indicates the packet type while also implicitly providing the size of the message. The checksum consists of a single byte. It is calculated as an exclusive-OR of both the header and message bytes and enables the power transmitter to check for transmission errors.

For a more detailed explanation please refer to the Qi Specification, section *Power Receiver to Power Transmitter communications interface*.

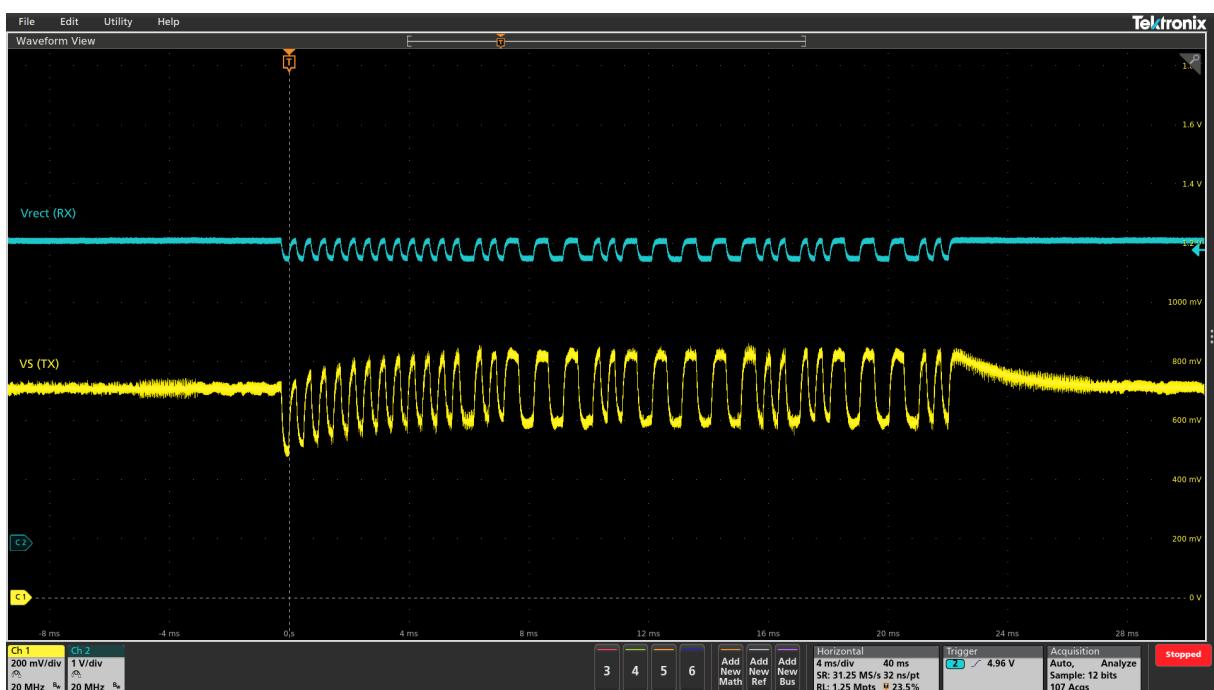
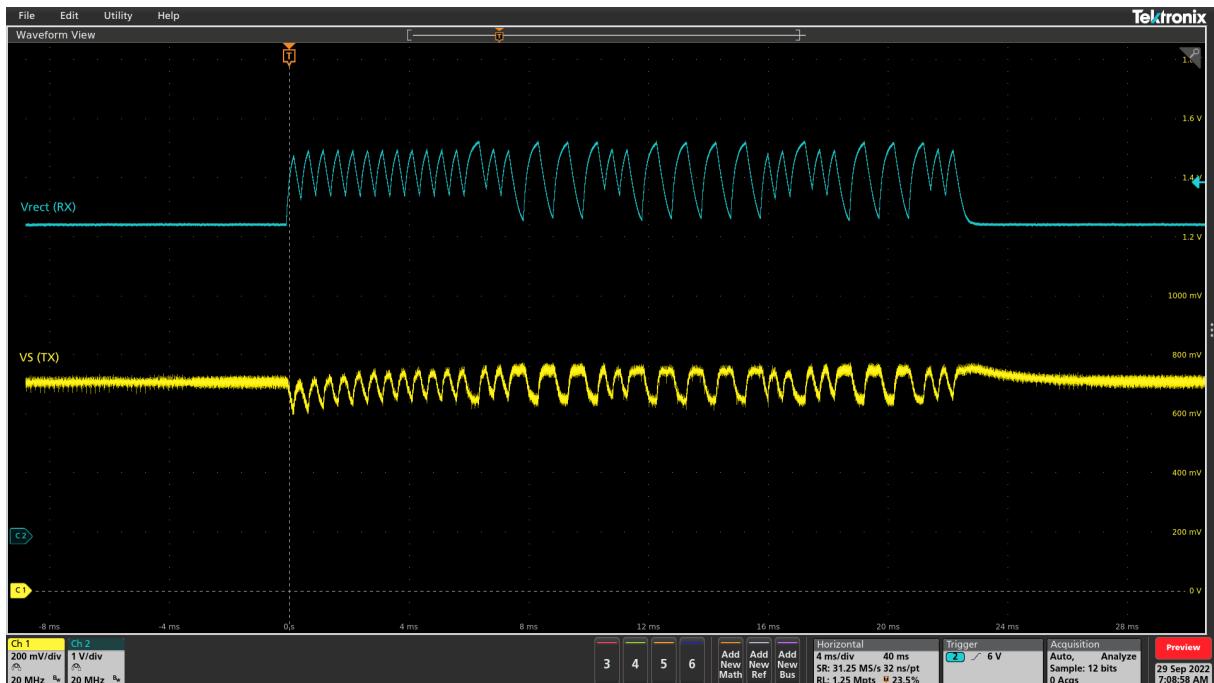
**Figure 47.** ASK communication example - high load current

Figure 48. ASK communication - no load



#### 4.12.2 FSK communication

The power transmitter modulates the power signal by switching between its normal operating frequency  $f_{op}$  and its modulated frequency  $f_{mod}$ . The difference between  $f_{op}$  and  $f_{mod}$  can be described by two parameters: **polarity** and **depth**.

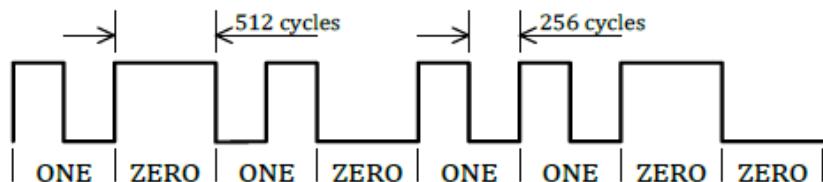
**Polarity** is given by the difference between  $f_{mod}$  and  $f_{op}$  – a positive polarity corresponds to a positive difference, while a negative polarity corresponds to a negative difference.

**Depth** describes the magnitude of difference between  $f_{op}$  and  $f_{mod}$ . Please note that a negative polarity results in a higher induced voltage on the power receiver coil and should therefore be used with care.

The FSK communication is executed via packets. Each packet consists of 3 parts – a header, a message, and a checksum. The header is a single byte that indicates the packet type, while also implicitly providing the size of the message part. The message size can range from 1 up to 27 bytes. The checksum is a single byte which provides a way to check for transmission errors. It is calculated as an exclusive-OR of the header bytes and the message bytes.

Each FSK **bit** is aligned to 512 periods of the power signal. A ONE bit is represented by two transitions over the 512-period slot - the frequency change occurs at the very start of the slot and after 256 power signal periods pass. A ZERO bit is represented by a single transition (at the start of the 512-period slot).

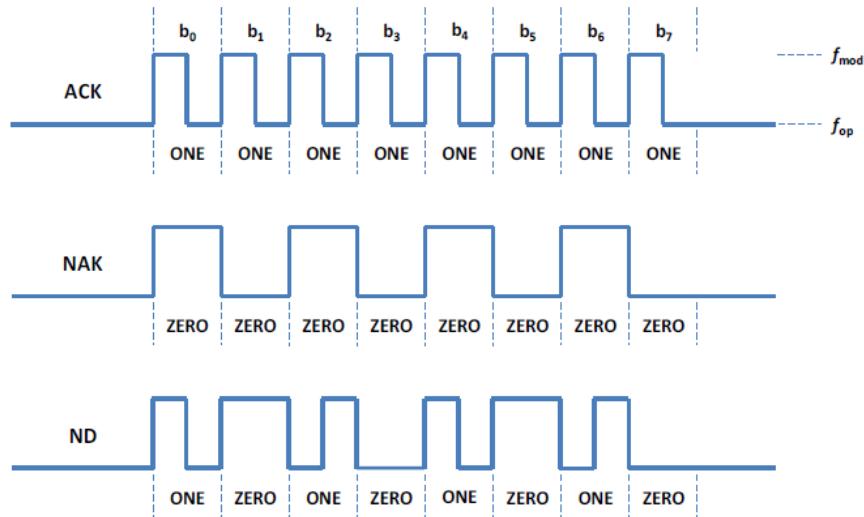
Figure 49. Example of differential bi-phase encoding



Each data **byte** is transferred as a sequence of 11-bits. The sequence consists of a start bit (ZERO), the data byte itself, a parity bit, and a stop bit (ONE). The parity used for byte encoding is even – the parity bit set to a ONE if the data byte contains an odd number of ONE bits. If the data byte contains an even number of ONE bits, the parity bit is set to a ZERO instead. This encoding scheme is used for all 3 parts of the FSK packet - the header, message, and checksum.

There is also an additional type of message that can be sent from the transmitter to the receiver – a response to a receiver's message. A response consists of 8 bits, so the receiver can use quite a simple logic to decode it. The response can be either Acknowledge (ACK), Not-Acknowledge (NACK) or Not-Defined (ND). ACK is encoded as a series of 8 ONE bits, NACK is encoded as 8 zero bits and ND is encoded as a series of 8 bits of alternating ZERO and ONE bits ('01010101').

Figure 50. Format of the three defined responses



For a more detailed explanation please refer to the Qi Specification, section Power Transmitter to Power Receiver communications interface.

#### 4.12.3

#### Most common Qi communication packets

- **Control error packet (CEP)** provides feedback from the power receiver to the power transmitter about the amount of power required by the load. CEP is a two's complement signed integer; its maximum range is -128 to +127. Its value is calculated by the receiver as a portion of the currently transmitted power, by which the current power level should be increased/decreased. A negative CEP indicates less power is required, while a positive CEP indicates more power is required. CEP equal to zero indicates no changes of the operation point are required.
- **Received power packet (RPP)** value is calculated by the power receiver as a portion of the maximum power value contained in the configuration packet and is used to inform the transmitter about the amount of power received by the receiver. This information is mainly used for foreign object detection.
- **Signal strength packet (SS)** indicates the degree of coupling between the transmitting and receiving coils. Its value is an unsigned integer, and it is calculated from the current power/voltage level as a portion of the maximum expected power/voltage.
- **End power transfer (EPT)** packet is generated by the receiver, and its purpose is to signal to the transmitter that power transfer shall be terminated. This request may have several causes, but the most common ones are:
  - The battery is fully charged,
  - A protection was triggered (overvoltage, overtemperature, etc.),
  - An internal fault in the receiver occurred,
  - No/insufficient response to control error packets is detected.
- **Proprietary packet (PP)** enables sending custom messages between the transmitter and the receiver. Qi standard defines different proprietary packet types, which can range in length from 1 to 20 bytes.
- **Charge status packet (CSP)** contains an unsigned integer which states the current charge status of the battery in the receiving device. CSP equal to 100 means the battery is fully charged, while CSP equal to 0 means the battery is completely discharged. If the receiving device does not contain a battery or it cannot provide charge status information, the CSP value should be set to 0xFF.

## 4.13 I2C interface

The STWBC86 can operate fully independent, that is, without being interfaced with a host system. In applications in which the STWBC86 is a part of peripherals managed by the host system, the two SDA and SCL pins could be connected to the existing I2C bus. The device works as an I2C slave and supports standard (100 kbps), fast (400 kbps), and fast mode plus (1 Mbps) data transfer modes. The STWBC86 has been assigned a 0x61 7-bit hardware address.

The pins are up to 3.3 V tolerant, and the pull-up resistors should be selected as a trade-off between communication speed (lower resistors lead to faster edges) and data integrity (the input logic levels must be guaranteed to preserve communication reliability). When the bus is idle, both SDA and SCL lines are pulled HIGH.

### 4.13.1 Data validity

The data on the SDA line remains stable during the high state of every SCL clock pulse. The high and low states of the SDA line only change when the SCL clock signal is low.

### 4.13.2 Start and stop conditions

Both the SDA and the SCL lines remain high when the I2C bus is not busy. A START condition is indicated by a high-to-low transition of the SDA line when SCL is HIGH, while the STOP condition is indicated by a low-to-high transition of the SDA line when SCL is HIGH. A STOP condition must be sent before each START condition.

### 4.13.3 Byte format

Every byte transferred over the SDA line contains 8 bits. Each byte received by STWBC86 is generally followed by an acknowledge (ACK) bit. The most significant bit (MSB) is transferred first. A single data bit is transferred during each clock pulse.

The device generates an ACK pulse (by pulling the SDA line low during the acknowledge clock pulse) to confirm a correct device address or data bytes reception.

### 4.13.4 Interface protocol

The interface protocol is composed of:

- A start condition (START)
- A device address + R/W bit (read =1 / write =0)
- A register H address byte
- A register L address byte
- A sequence of n data bytes (each data byte must be acknowledged by the receiver)
- A stop condition (STOP)

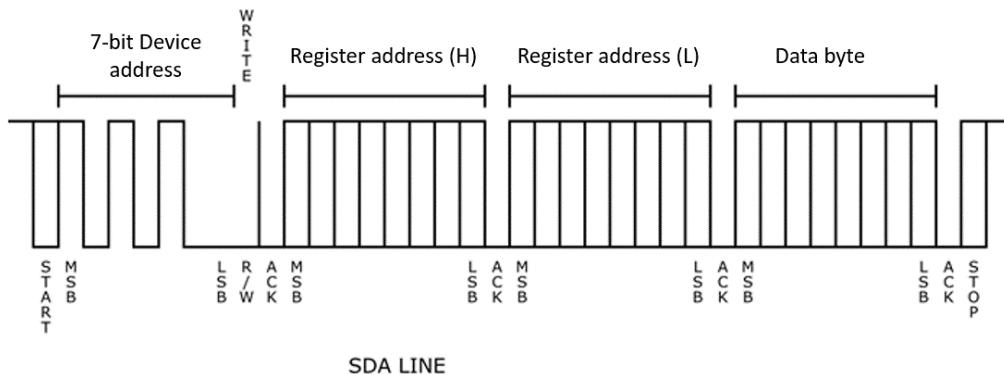
The register address byte determines the first register in which the read or write operation takes place. When a read or write operation is finished, the register address is automatically incremented.

Frequently used acronyms:

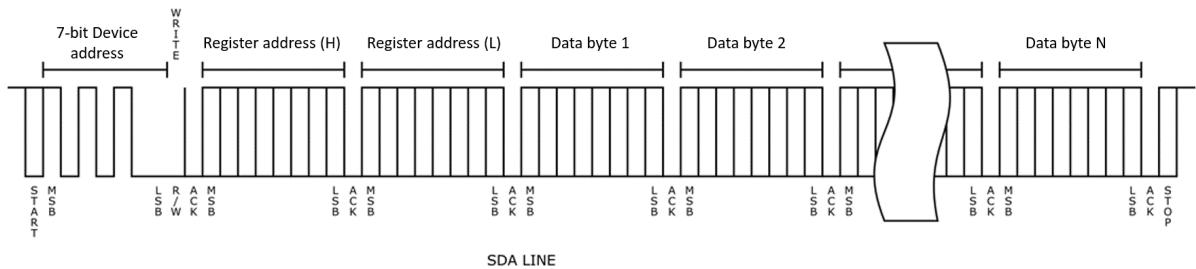
SAD: Slave Address; SUB AD: Subaddress; SR: Repeated start; R: Read; W: Write; SAK: Slave Acknowledge; MAK: Master Acknowledge; NMAK: No Master Acknowledge.

### 4.13.5 Writing to a single register

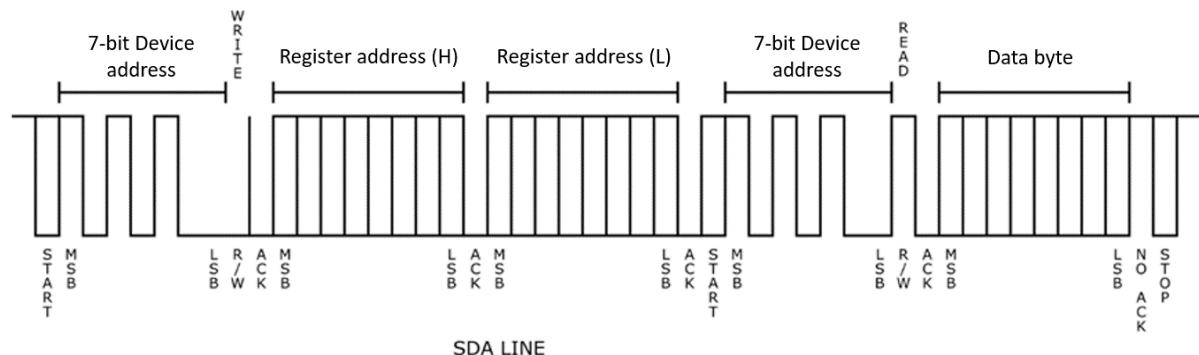
Writing to a single register begins with a START condition followed by the device address 0xC2 (7-bit device address plus R/W bit cleared), two bytes of the register pointer and the data byte to be written in the destination register. Each transmitted byte is acknowledged by the STWBC86 through an ACK pulse.

**Figure 51. Writing to a single register****4.13.6****Writing to multiple registers with incremental addressing**

The STWBC86 supports writing to multiple registers with auto-incremental addressing. When data is written into a register, the register pointer is automatically incremented, therefore transferring data to a set of subsequent registers (also known as page write) is a straightforward operation.

**Figure 52. Writing to multiple registers****4.13.7****Reading from a single register**

Reading from a single register begins with a START condition followed by the device address byte 0xC2 (7-bit device address plus R/W bit cleared) and two bytes of register pointer. A restart condition is then generated and the device address 0xC3 (7-bit device address plus R/W bit asserted) is sent, followed by data reading. The ACK pulse is generated by the STWBC86 at the end of each byte, but not for data bytes retrieved from the register. A STOP condition is finally generated to terminate the operation.

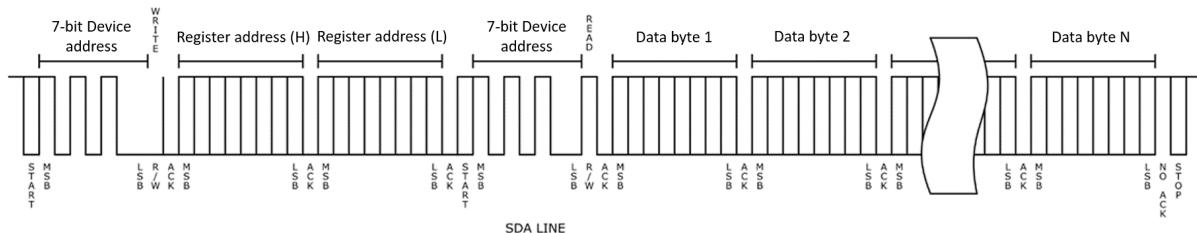
**Figure 53. Reading from a single register**

#### 4.13.8

#### Reading from multiple registers with incremental addressing

Similarly, to multiple bytes (page) writing, reading from subsequent registers relies on an auto-increment of the register: The master can extend data reading to the following registers by generating an ACK pulse at the end of each byte. Data reading starts immediately, and the stream is terminated by an NMAK at the end of the last data byte, followed by a STOP condition.

**Figure 54. Reading from multiple registers**



#### 4.14

#### GPIOx and INTB pins

GPIO0 through GPIO7 are programmable general-purpose I/O pins. These pins can be configured as inputs or outputs (push-pull or open-drain) and assigned various functions.

**Table 3. GPIO functions**

Code	I/O	Function
0x01	I	Pull-up
0x02	I	Pull-down
0x03	O	Open drain (Active high)
0x04	O	Open drain (Active low)
0x05	O	Interrupt (Open drain)
0x06	O	Firmware ready
0x1A	O	Power transfer on – high when in power transfer, low in standby and ping state
0x1E	O	Tx error – high in error state which causes Tx to stop pinging. Conditions which cause this error can be set in the GUI

The INTB (GPIO6) pin is an interrupt output line that can be assigned to any internal interrupt condition and used to inform the host system about a specific event.

#### 4.15

#### Interrupt registers

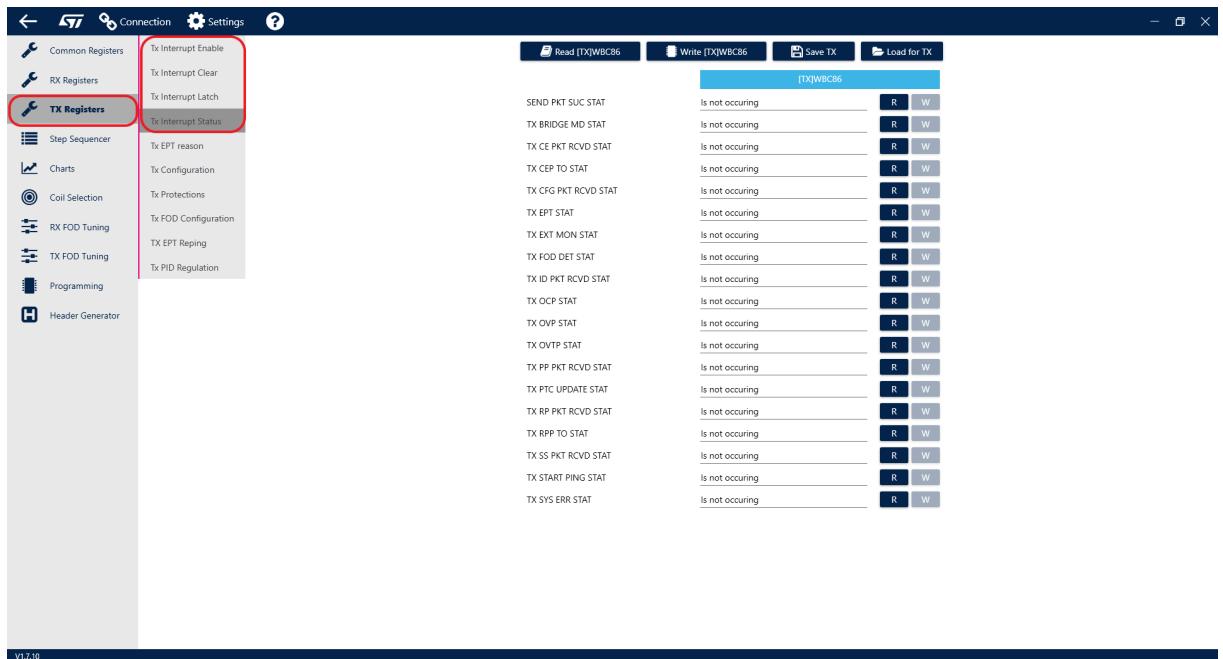
There are 4 bits (enable, clear, latch, and status) assigned to each interrupt sorted into separate tabs.

The Enable tab can be used to either enable the corresponding interrupt (write), or to check whether the interrupt is already enabled (read).

The Latch tab can be used to determine which interrupts have been triggered (read only). After being triggered, the bit remains set to 1 until cleared by writing a 1 into the corresponding clear register (write only).

The Status tab can be used to determine which interrupt is being triggered at the moment (read only). The status bit goes back to zero after the triggering condition is removed.

Figure 55. Interrupt registers



## 4.16 Frequency hopping (fhop)

One of the most common causes of ASK demodulation failure is noise produced by the adapter supplying power to the transmitter. The noise interferes with the communication, which may cause the decoding to fail, as the principal components of the noise are often close to the 2 kHz ASK communication frequency.

The STWBC86 features a frequency hopping (fhop) function that tries to improve the communication conditions by shifting the operating frequency by a set step whenever CE timeout occurs, as the new operating point may be less affected by the interference.

The fhop feature can be enabled and configured in the Tx Configuration section of the Tx Registers tab:

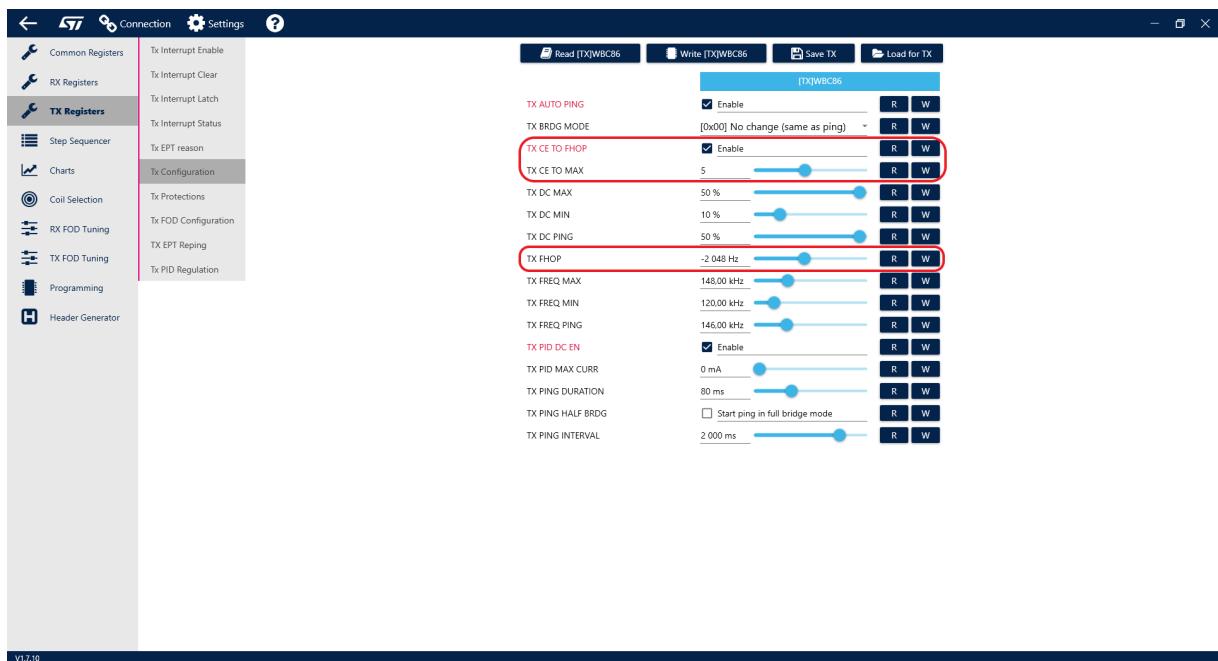
- TX CE TO MAX defines the number of CEP timeouts that the device must register before terminating the power transfer. When set to 1, the device terminates the power transfer immediately after registering a CEP timeout. This register must never be set to 0. This should be considered especially when the device is programmed by a host MCU! If the hopping feature is enabled, the device performs a frequency hop whenever CEP timeout occurs. The maximum number of hops should be set conservatively. A high value enables the transmitter to keep transmitting power for quite some time even after the receiver is removed.
- TX CE TO FHOP enables the hopping feature. However, the device always waits for the number of CE timeouts set in the TX CE TO MAX register even when hopping disabled!
- TX FHOP defines the hop size in Hz. A positive value specifies a hop to a higher frequency, while a negative value means a hop to a lower frequency. However, the frequency cannot be shifted beyond the set minimum and maximum frequency (frequency range). Therefore, when a CEP timeout occurs, the firmware first checks if the frequency hopping would shift the operating frequency outside of the range. If it determines that a shift beyond the boundary would occur, the firmware starts frequency hopping in the opposite direction instead. No other checks are performed, however.

It is recommended to set the TX FHOP\*TX CE TO MAX product to be at most half of the frequency range. This ensures that the target frequency always remains within the set boundaries.

Note:

*Operating frequency does not move beyond the boundary even if the target exceeds it.*

**Figure 56. FHOP settings**



The figure below shows an example of frequency hopping with the parameters defined above.

**Figure 57. FHOP example**



1. Receiver is placed on the transmitter; transmitter regulates based on the CEP value; the Rx is removed shortly after.
2. CE timeout occurred, as there was no communication from the receiver. The transmitter performed four frequency hops (-2048 Hz step).
3. CE timeout occurred for the fifth time; the Tx ended the power transfer.

The same would apply in case of ASK demodulation error. If the transmitter is not able to decode the set number of CE packets in the row, power transfer is terminated. However, if a CE packet is decoded successfully, the transmitter resumes normal operation.

## 4.17 PID tuning

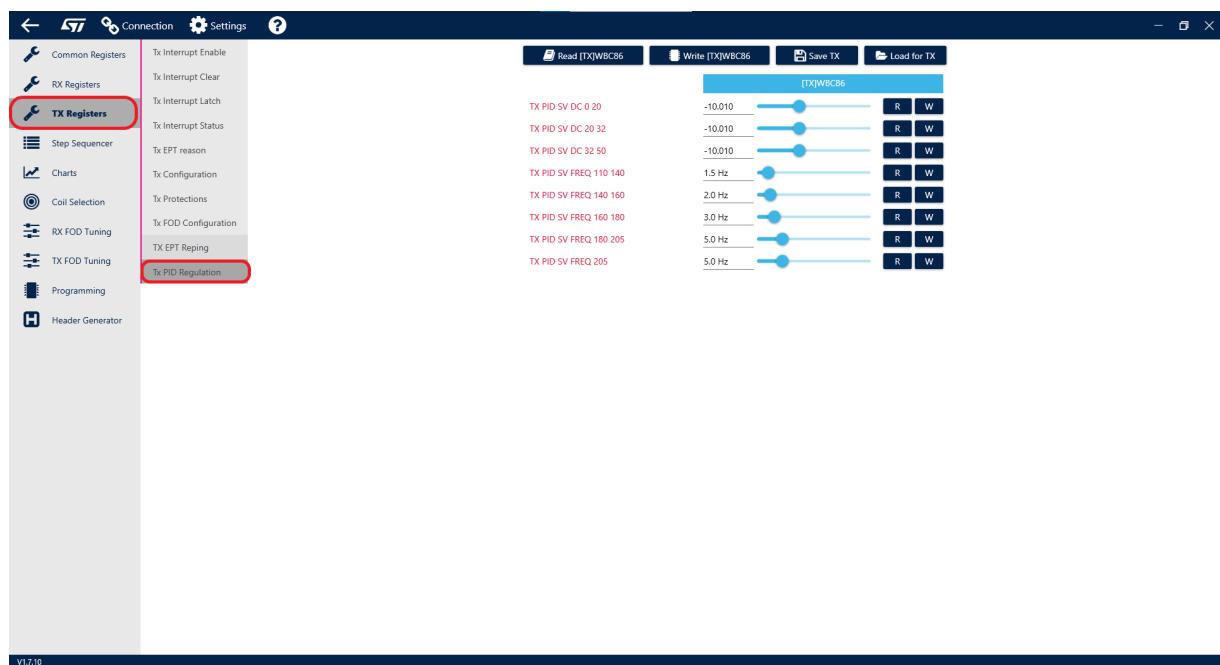
Power regulation of the power transmitter is controlled by a PID algorithm, as defined by the Qi specification. The default scaling factors for the respective parts are set according to the topology used – A11a in the case of STEVAL-WBC86TX.

Application specific requirements may demand adjustments in the topology, such as coil design or the input voltage (also defined by the topology specification). However, modifications to the circuit may affect the PID regulation performance, and consequently, may (or may not) require adjusting the PID parameters.

For example: for a 6 V input voltage, the PID regulation may be too aggressive to achieve a stable regulation loop. Decreasing the scaling factors may improve the performance by slowing down the loop but may adversely impact other aspects of the regulation. Therefore, thorough testing is necessary after any adjustments to the PID parameters!

The PID parameters can be found and adjusted in the Tx PID Regulation part of the TX Registers tab.

Figure 58. PID parameters



## 5 STEVAL-WBC86TX description and operation

### 5.1 STWBC86 default configuration

Table 4. Basic parameters

Parameter	Settings
Transmitter topology	A11a
TX bridge mode	Full sync
Minimum duty cycle	10 %
Maximum duty cycle	50 %
Minimum operating frequency	120 kHz
Maximum operating frequency	148 kHz
Ping duty cycle	50 %
Ping frequency	146 kHz
Ping duration	80 ms
Ping interval	2,000 ms
Overshoot protection (OCP)	2 A
Oversupply protection (OVP)	20 V
Overtemperature protection (OVTP)	100 °C
NTC protection	Disabled
Maximum CEP timeout count (for fhop feature)	3
Fhop step	1,920 kHz
Protection debounce	0
FOD debounce	4

- Enabled interrupts
  - Bridge mode change
  - CEP timeout
  - RPP timeout
  - Packet sent successfully
  - EPT from RX received
  - Foreign object detected
  - OCP triggered
  - OVP triggered
  - OVTP triggered
- Enabled features
  - Foreign object detection
  - Auto ping
  - Fhop
  - Duty cycle regulation
- EPT conditions after which the device does not start pinging automatically
  - Foreign object detected
  - OVP triggered
  - System error

- GPIOs
  - No functions assigned

## 5.2

### Typical performance characteristics

The following table shows charging performance of an STWBC86/STWLCL38 (TX/RX) setup at various load currents, with the temperature being measured after 5 minutes of continuous operation.

**Table 5. Charging performance**

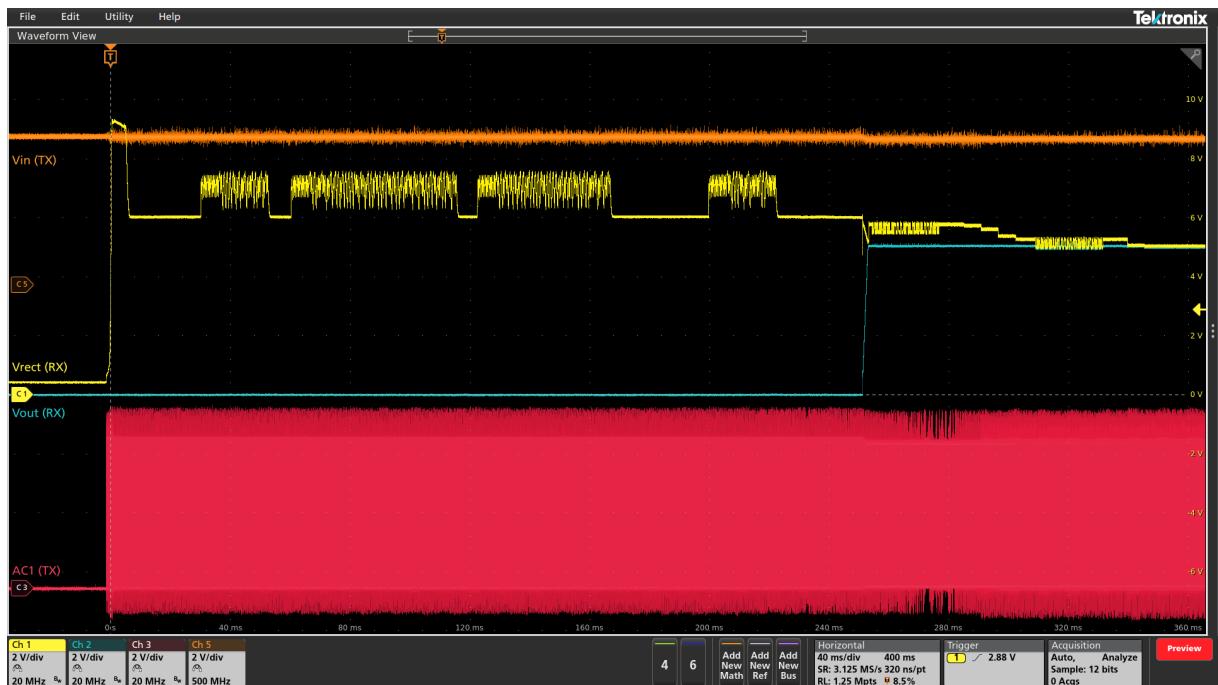
Iout [mA] (RX)	lin [mA] (TX)	Vout [V] (RX)	Vin [V] (TX)	Trect [°C] (RX)	Trect [°C] (TX)
200	248	5,01	5,015	30	31,7
500	597	5,01	4,965	33	31,5
700	854	5,01	4,951	35	31,5
1000	1294	5,01	4,885	41	38,7

#### 5.2.1

##### Power-up waveforms

The following figure shows power-up waveforms of an STWBC86 (TX) and STWLCL38 (RX) setup. The transmitter and receiver coils are aligned and a  $10 \Omega$  load is connected to the receiver output.

**Figure 59. Power up waveform**



**Table 6. Signal legend for power up waveform**

Color	Signal
Yellow	Vrect (RX)
Blue	Vout (RX)
Red	AC1 (TX)
Orange	Vin (TX)

## 5.2.2 Efficiency and spatial freedom in the XY plane

Efficiency is one of the most important metrics of wireless charging performance evaluation. Spatial freedom, size of the area in which a power receiver can be placed on the power transmitter, which still allows sufficient power to be transmitted, is another important metric.

Efficiency and spatial freedom of the STEVAL-WBC86TX were measured with STEVAL-WLC38RX as the receiver. The efficiency was measured from the transmitter DC input to the receiver DC output. The measurement does not include any power losses in the wall adapter or the USB Type-C® cable.

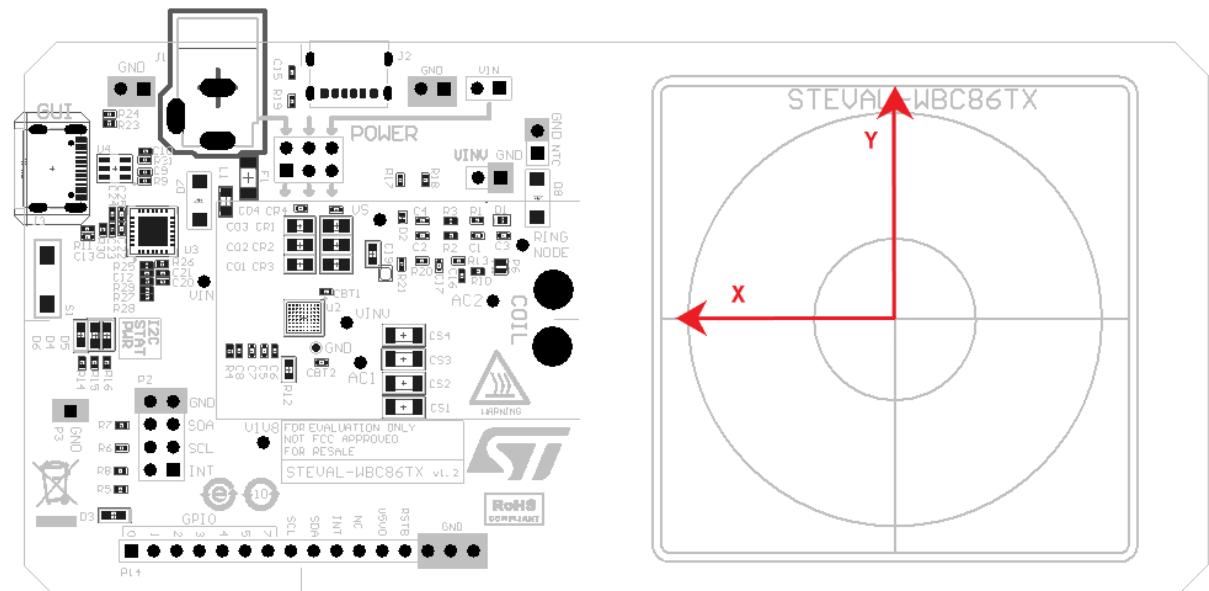
The test setup comprised of:

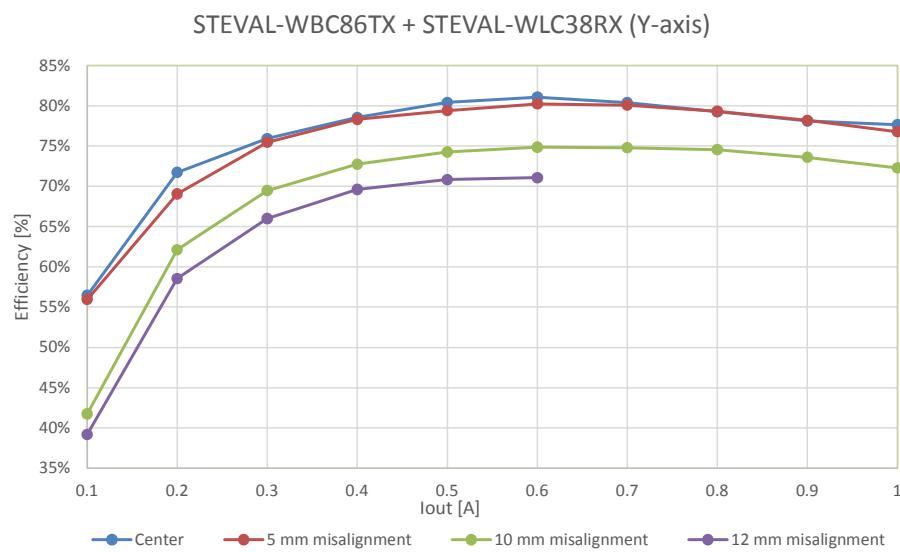
- 20 W PISEN wall adapter, model A829-120167C-EU1
- USB Type-C® power cable from Vention, model COT
- STEVAL-WBC86TX as transmitter
- STEVAL-WLC38RX as receiver
- Electronic load in CC mode, model BK Precision 8500.

The maximum efficiency achieved with this setup was 81%.

Efficiency curves for various misalignments in the X and Y axis are shown in the figures below:

Figure 60. X, Y axis for misalignment



**Figure 61. Efficiency curve in X axis****Figure 62. Efficiency curve in Y axis**

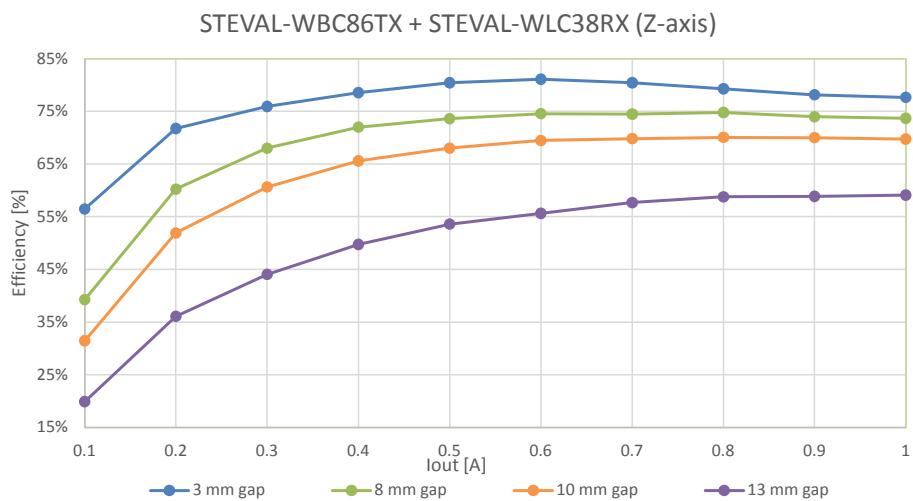
### 5.2.3

### Efficiency and spatial freedom in the Z-axis

Z-axis distance between the coils, also known as charging gap, is an additional parameter that significantly affects charging performance. Therefore, the STEVAL-WBC86TX was also tested at various charging gap distances.

Efficiency curves for various misalignments in the Z-axis are shown in the figure below:

Figure 63. Efficiency curve in Z axis

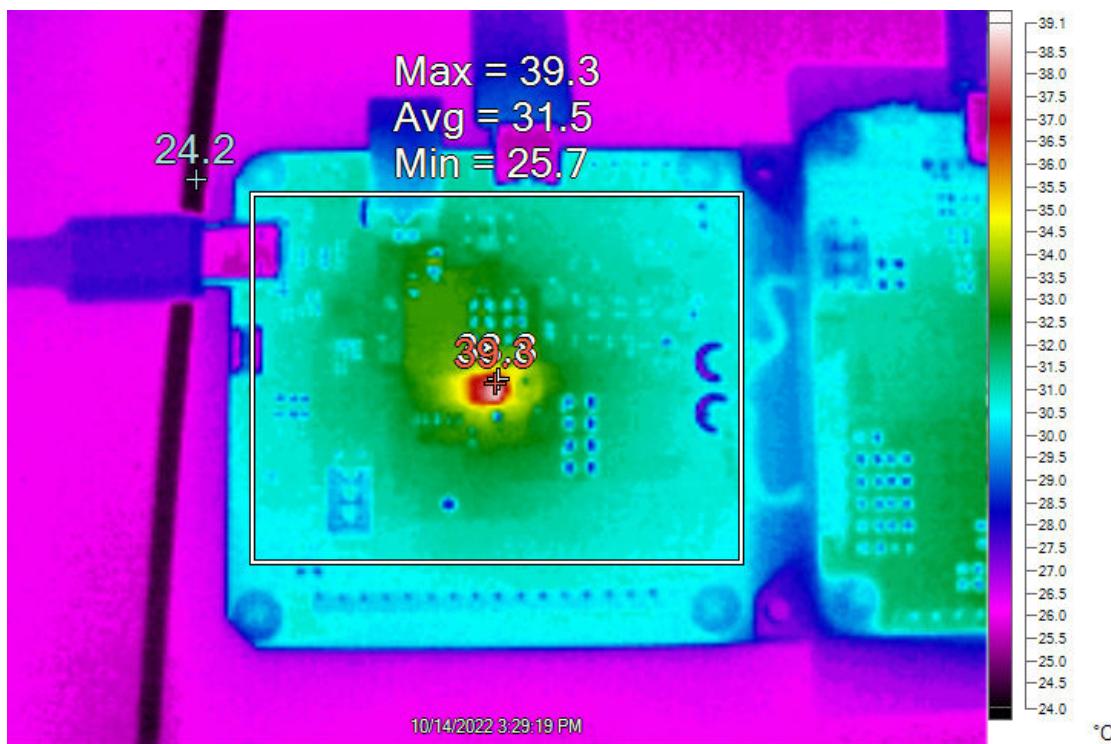


Z-distance of 3 mm is a typical value for most applications (2 mm on the TX side + 1 mm on the RX side). Clearly, the transmitter can deliver sufficient power even with 13 mm charging gap. However, efficiency is decreasing rapidly with increasing charging gap. Therefore, minimizing Z-distance whenever possible is recommended.

#### 5.2.4 Thermal performance

Thermal performance of the board with a 5 W load (5 V/1 A on the RX side) after 5 minutes of continuous operation.

Figure 64. Thermal performance



## 6 Designing a 5 W wireless power transmitter based on the STEVAL-WBC86TX evaluation board

The design should begin with external component selection, as those components have a significant impact on the board performance. Modifications of the external components are also harder to implement after the board is manufactured and assembled.

### 6.1 External component design

The STWBC86 requires only a few external components to work properly. The design should begin with coil and series resonant capacitor selection, as those are the crucial parts of the design and affect most of the device's functions.

#### 6.1.1 Coil selection

Coil selection is a complex problem, as there are several factors that influence the resulting performance. When selecting the coil, the designer should consider the coil's inductance, DCR, ACR, current rating, dimensions, maximum z-distance, and layout.

DCR, ACR (DC and AC resistance) should be kept as low as possible to minimize power loss caused by the current flowing through the coil.

Dimensions, maximum z-distance, and coil layout have to be considered to achieve the best coupling possible, as better coupling results in higher efficiency. Similar sized coils might offer a higher coupling factor, on the other hand a different sized RX/TX coil might offer greater freedom of positioning.

Using the default coil combination found in STMicroelectronics's EVKs should be a good starting point. For more information regarding coil selection, please refer to AN5961 or the Wireless Power Consortium's documentation, which describes coils used in the Qi certified topologies.

#### 6.1.2 Series resonant capacitor selection

Selection of the series capacitors is greatly influenced by the transmitting coil design. The value should be selected so that the resulting circuit has its resonant peak located at, or very close to, 100 kHz. The following formula can be used to calculate the series capacitance value:

$$f = \frac{1}{2 \cdot \pi \cdot \sqrt{L \cdot C}} \rightarrow C = \frac{1}{L \cdot (2 \cdot \pi \cdot f)^2}$$

Voltage rating of the capacitors should consider that during operation, voltage generated across the capacitors is usually much higher than the input/output voltage of the system. Therefore, the minimum recommended voltage rating of the series resonant capacitors is 50 V.

Using C0G dielectric capacitors for the resonant circuit design is recommended to minimize temperature and biasing voltage influence on the capacitance value as much as possible.

For more information regarding capacitor value selection, please refer to Wireless Power Consortium's documentation, which describes capacitors used in the Qi certified topologies.

#### 6.1.3 Vin and Vinv capacitors selection

Selection of the VIN and VINV capacitor's value is mostly a trade-off between load transition response time and capacitor cost and/or size. Capacitance of 30 uF for each of the nodes should be a good starting point. The voltage rating should respect the maximum input voltage. Ceramic X5R or X7R (preferred) dielectric capacitors are recommended.

Capacitor derating caused by temperature and DC bias should also be considered, as the capacitance decrease caused by those effects may be quite significant. Please refer to the capacitor datasheet for more information.

#### 6.1.4 Hardware input protections

Hardware input overvoltage and overcurrent protections are recommended to avoid potential damage to the device caused by unusual conditions or wrong operation of the board.

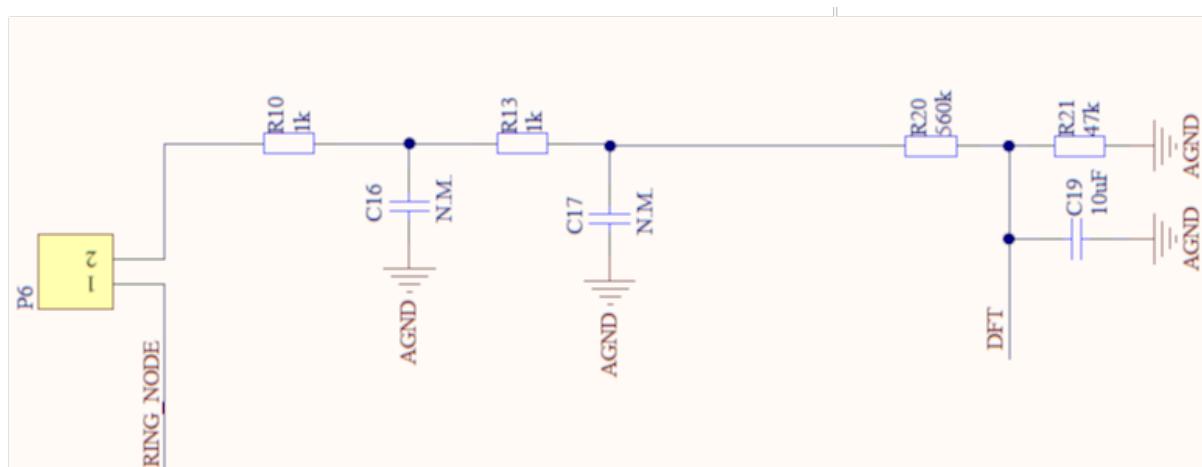
A TVS diode is a good choice for overvoltage protection. Its breakdown voltage should be higher than the input voltage but should keep the input voltage safely below 27 V (absolute maximum rating of the input pins).

A fuse can be used as a cheap and simple overcurrent protection. The maximum input current should be limited to 2 A.

## 6.2 Ring node voltage sensing

Ring node (the node between transmitting coil and series resonant capacitor) voltage measurement is used as an additional indicator in foreign object detection. However, the DFT pin, used for the ring node voltage sensing, is only 1.98 V tolerant. In addition, the internal ADC supports only up to a 1.5 V reading. A resistor divider is therefore recommended to keep the DFT voltage below 1.5 V.

**Figure 65. Ring node voltage sensing**



To accurately measure the ring node voltage, the firmware requires the user to provide an accurate division factor in the GUI. The corresponding setting can be found in the Tx FOD Configuration section of the TX Registers tab. The division factor is defined as the ring node voltage divided by the DFT pin voltage.

**Figure 66. Ring node divider configuration**



## 6.3 PCB layout guidelines

- Power tracks (AC1, AC2, VIN, VINV) and power ground tracks should be kept wide enough to sustain high current. Duplicating these tracks in inner layers and adding vias is advisable wherever possible to lower impedance as much as possible.

- AC1, AC2, BOOT1, and BOOT2 generate noise. Using shielding near these traces (by placing ground planes below) is recommended.
- Power ground carries the sum of ripple current and DC current from the inverter. Current return paths from LDO capacitors should be routed separately from these high current paths.
- AC1 and AC2 tracks should be routed close together to minimize the area of the resulting loop.
- Communication (I2C) and sensing signals should be routed far from noise generating nets (AC1, AC2, BOOT1, and BOOT2) to minimize the effects of interference induced from those high di/dt nets.
- BOOT1, BOOT2 capacitors should be placed as close to the device as possible.
- Input and inverter decoupling capacitors should be placed close to the device to minimize the area of high current loops.
- Auxiliary LDO capacitors should be placed as close to the device as possible.
- Thermal performance and grounding can be enhanced by dedicating one layer as a ground plane. No signal/power tracks should be routed on this layer to ensure ground integrity.

## 6.4 Reference code with STM32 development boards

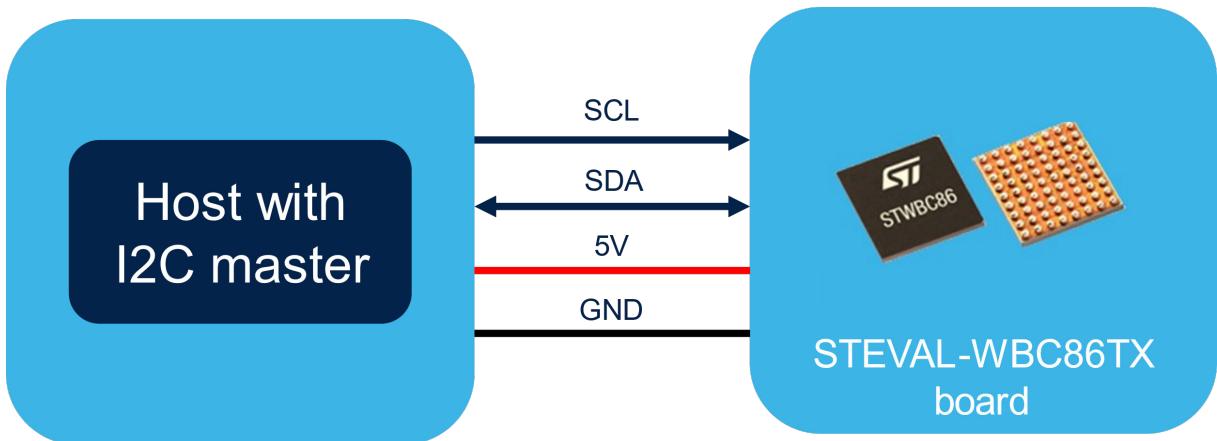
### 6.4.1 Hardware requirements

1. STM32 development board (for example, STM32 Nucleo-144)
2. STEVAL-WBC86TX

### 6.4.2 Hardware connections

1. Connect the STWBC86's I2C pins to the master I2C bus. These signals require pull-up resistors to work properly.
2. Power up STWBC86.

Figure 67. Pin connection between host and STWBC86 chip/evaluation board



STM32 Nucleo-144 board is used as an example.

Table 7. Pin connection between host (STM32) and STWBC86

STM32 Nucleo-144	STEVAL-WBC86TX board
5V (CN11.18)	VIN
GND (CN11.20)	GND
I2C1_SDA (PB9 -> CN12.5)	SDA
I2C1_SCL (PB8 -> CN12.3)	SCL

#### 6.4.3 Software requirements

1. Patch/Configuration data in header (.h) file format [STSW-WBC86FWBPP](#)
2. STWLC NVM programming reference code
3. [STM32CubeIDE](#)

#### 6.4.4 Source files

The source files included in the reference code are listed below.

1. *STWBC86.h*
  - Provides NVM programming API and holds the structure and register definitions of STWBC86.
2. *STWBC86.c*
  - Main file which describes the NVM related programming sequences and I2C Write/ReadWrite operations.
3. *nvm\_data.h*
  - Patch and Configuration data in Header file format to be programmed into the chip.

#### 6.4.5 Reference code porting procedure

1. Create a new project in the [STM32CubeIDE](#) with an STM32 development board. Visit the STMicroelectronics website for [STM32CubeIDE](#) documentation.
2. Enable the I2C and UART features of STM32.

**Figure 68. Connectivity**



3. Copy *STWBC86.h*, *STWBC86.c*, and *nvm\_data.h* into the main directory (*Core/Src*).
4. Add the following code into *Core/Src/main.c*.
  - a. Include the Header file

```
/* USER CODE BEGIN includes */#include "STWBC86.h"/* USER CODE END includes */
```
  - b. Configure I2C and UART master in **main()**

```
/* USER CODE BEGIN 2 */// WLC - initialize I2C and UART peripheralhi2c = &hi2c1;huart = &huart3;
```
  - c. Execute NVM programming

```
// WLC - display chip informationchar buff[PAGE_SIZE] = {0};chip_info_show(buff);pr_info(buff);// WLC - perform NVM programming/* USER CODE END 2 */pr_info(buff);nvm_program_show(buff);memset(buff, 0, PAGE_SIZE);
```

5. Add the following I<sup>2</sup>C error handling code below stm32f7xx\_it.c when using STM32 FW\_F7 V1.7.0 or lower.  
This handles the I<sup>2</sup>C NACK after performing a system reset.

```
/*
 * @brief this function handles I2C1 error interrupt.
 */
void I2C1_ER_IRQHandler(void)
{
    /* USER CODE BEGIN I2C1_ER_IRQn 0 */
    i2c_temp.Mode = hi2c1.Mode;
    i2c_temp.State = hi2c1.State;
    /* USER CODE END I2C1_ER_IRQn 0 */
    HAL_I2C_ER_IRQHandler(&hi2c1);
    /* USER CODE BEGIN I2C1_ER_IRQn 1 */
    hi2c1.Mode = i2c_temp.Mode;
    hi2c1.State = i2c_temp.State;
    /* USER CODE END I2C1_ER_IRQn 1 */
}
```

#### 6.4.6

#### API

1. chip\_info\_show

Name	int chip_info_show(char *buf)		
Description	Read chip information and return chip ID and version		
Parameters	char *	buf	Host allocated buffer to return chip information or error codes.
Return value	int		Length of the output buffer

2. nvm\_program\_show

Name	int nvm_program_show(char *buf)		
Description	Program NVM with provided Header file.		
Parameters	char *	buf	Host allocated buffer to return error codes.
Return value	int		Length of the output buffer

#### 6.4.7

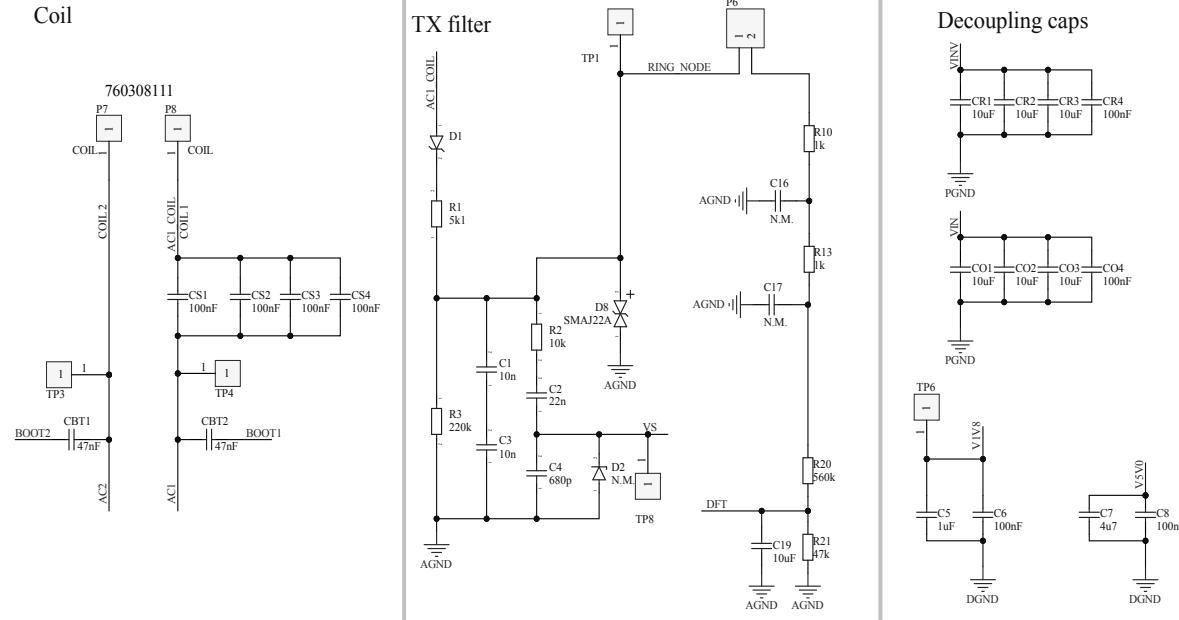
#### Error codes

Table 8. Error codes with descriptions

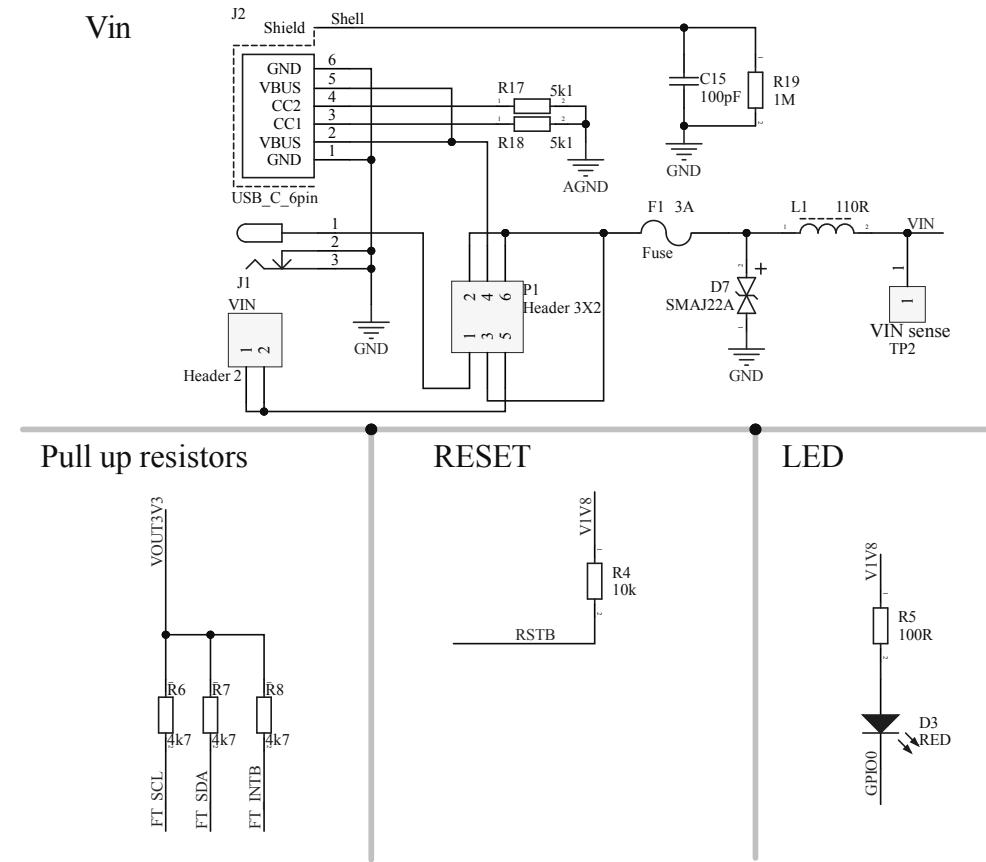
Error code	Value	Description
OK	0x00000000	Success operation.
E_BUS_R	0x80000001	Error during I2C read operation.
E_BUS_W	0x80000002	Error during I2C write operation.
E_BUS_WR	0x80000003	Error during I2C write read operation.
E_UNEXPECTED_OP_MODE	0x80000004	Invalid operating mode. NVM programming must be done in DC mode.
E_NVM_WRITE	0x80000005	Failed to program NVM.
E_INVALID_INPUT	0x80000006	Invalid input parameters.
E_MEMORY_ALLOC	0x80000007	Failed to allocate memory buffer.
E_UNEXPECTED_HW_REV	0x80000008	Invalid chip revision. Should match the provided nvm_data.h.
E_TIMEOUT	0x80000009	Timeout error when performing firmware operation.
E_NVM_DATA_MISMATCH	0x8000000A	Data mismatch with programmed data.
E_UNEXPECTED_CHIP_ID	0x8000000B	Invalid chip ID. Should match the provided nvm_data.h.

## Schematic diagrams

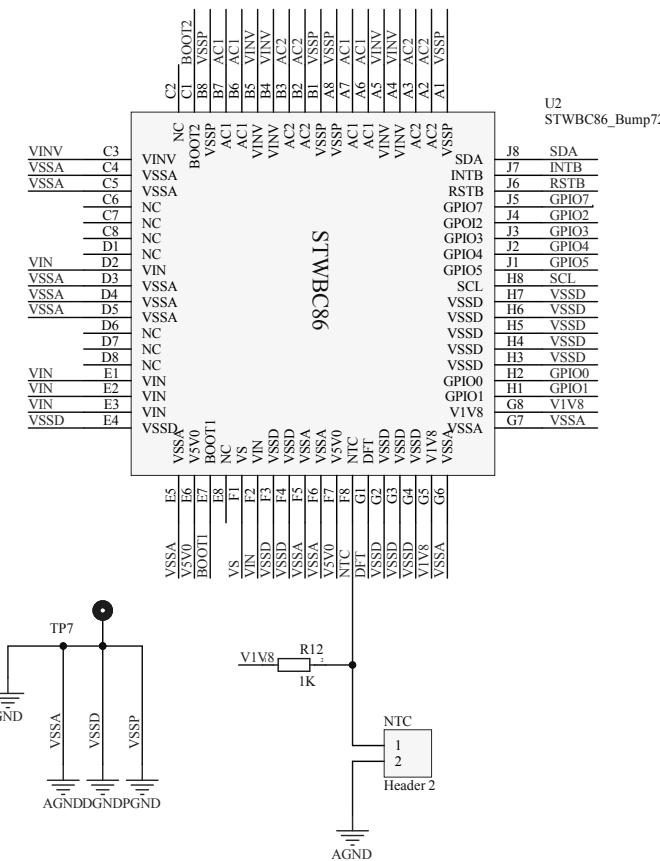
**Figure 69. STEVAL-WBC86TX circuit schematic (1 of 4)**



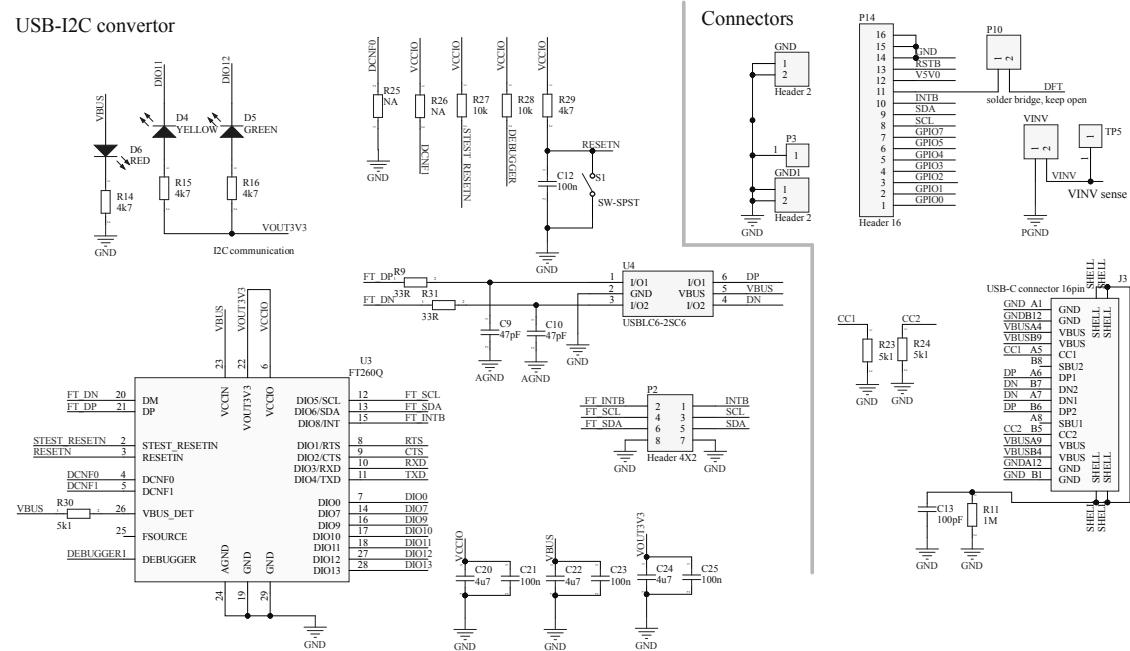
**Figure 70. STEVAL-WBC86TX circuit schematic (2 of 4)**



**Figure 71. STEVAL-WBC86TX circuit schematic (3 of 4)**



**Figure 72. STEVAL-WBC86TX circuit schematic (4 of 4)**



## 8 Bill of materials

**Table 9. STEVAL-WBC86TX bill of materials**

Item	Q.ty	Ref.	Value	Description	Manufacturer	Part number
1	4	C6, C8, CO4, CR4	100nF, C_0402, 50V, 10%,	Min 50V; X5R/X7R	Wurth	885012205092
2	2	C16, C17	N.M.		N.M.	N.M.
3	6	CO1, CO2, CO3, CR1, CR2, CR3	10uF, C_0805, 35V, 10%,	Min 35V; X5R/X7R	Murata	GRM21BR6YA106KE43 L
4	6	TP1, TP2, TP3, TP4, TP6, TP8	N.M., J_TESTPOINT_1.1mm,	PIN	N.M.	N.M.
5	5	R17, R18, R23, R24, R30	5k1, R_0402, 62.5m W, 5%,	Resistor	Yageo	RC0402JR-075K1L
6	4	C12, C21, C23, C25	100n, C_0402, 50V, 10%,	Min 6.3V; X5R/X7R	Wurth	885012205092
7	4	CS1, CS2, CS3, CS4	100nF, C_1206, 50V, 5%,	Min 50V; C0G	Murata	GRM31C5C1H104JA01 L
8	4	GND, GND1, VIN, VINV	HDR1X2,	Header 2	Harwin	M20-9730245
9	3	R4, R27, R28	10k, R_0402, 62.5m W, 5%,	Resistor	Yageo	RC0402JR-1310KL
10	7	R6, R7, R8, R14, R15, R16, R29	4k7, R_0402, 62.5m W, 5%,	Resistor	Yageo	RC0402JR-134K7L
11	2	C1, C3	10n, C_0402, 50V, 10%,	Min. 50V; X5R/X7R	Wurth	885012205067
12	2	C9, C10	47pF, C_0402, 50V, 5%,	Min. 6.3V; X5R/X7R	Wurth	885012005059
13	2	C13, C15	100pF, C_0402, 50V, 10%,	Min. 50V; X5R/X7R	Wurth	885012205055
14	1	C19	10uF, C_0603, 6.3V, 20%,	Min. 6.3V; X5R/X7R	Wurth	885012106006
15	4	C7, C20, C22, C24	4u7, C_0402, 6.3V, 20%,	Min. 6.3V; X5R/X7R	Wurth	885012105008
16	2	CBT1, CBT2	47nF, C_0402, 50V, 10%,	Min. 50V; X5R/X7R	Murata	GRT155R71H473KE01 D
17	2	D3, D6	D_0603_LED, 2V, 20mA,	RED	Wurth	150060RS75000
18	2	D7, D8	D_SMA (DO-214AC), 22V, 400W,	Diode	Littelfuse	SMAJ22A
19	2	P7, P8 connect coil 760308111 (item 27)	2mm_PIN,	COIL	N.M.	N.M.
20	2	R9, R31	33R, R_0402, 62.5m W, 5%,	Resistor	Yageo	RC0402FR-0733RL
21	2	R10, R13	1k, R_0402, 62.5m W, 5%,	Resistor	Yageo	RC0402JR-071KA
22	2	R11, R19	1M, R_0402, 62.5m W, 5%,	Resistor	Yageo	RC0402JR-7D1ML
23	1	R12	1k, R_0603	Resistor	Yageo	RC0603JR-071KL

Item	Q.ty	Ref.	Value	Description	Manufacturer	Part number
24	1	C2	22n, C_0402, 50V, 20%,	Min. 50V; X5R/X7R	Murata	GCM155R71H223MA55 D
25	1	C4	680p, C_0402, 50V, 10%,	Min. 50V; X5R/X7R	Wurth	885012205060
26	1	C5	1uF, C_0402, 10V, 20%,	Min. 16V; X5R/X7R	Wurth	885012105012
27	1	COIL connect to P7, P8 (item 19)			Wurth	760308111
28	1	D1	DIOMELF1006N	Diode	Diodes Incorporated	1N4448HLP-7
29	1	D2	N.M., D_SOD882	Diode	N.M.	N.M.
30	1	D4	D_0603_LED, 2V, 20mA	YELLOW	Wurth	150060YS55040
31	1	D5	D_0603_LED, 2V, 20mA	GREEN	Wurth	150060VS55040
32	1	F1	F_SMD_1206_DIN3216M 65V, 3A	Fuse	Bourns	SF-1206S300W-2
33	1	J1	J_PTH_POWERJACK_2.1M M, 18V, 1.5A	Barrel connector	Kubiconn	163-179PH-EX
34	1	J2	USB_C_UJC-HP-3-SMT-TR, 3A	USB_C_6pin	CUI devices	UJC-HP-3-SMT-TR
35	1	L1	L_SMD_0805_DIN2012M, 3A	Inductor	Wurth	742792025
36	1	NTC	N.M., HDR1X2	Header 2	N.M.	N.M.
37	1	P1	HDR2X3	Header 3X2	Harwin	M20-9720345
38	1	P2	HDR2X4	Header 4X2	Harwin	M20-9720445
39	1	P3	N.M., 2mm_PIN	PIN	N.M.	N.M.
40	1	J3	J_USB4105-GF-A-120, 20V, 5A	USB Type-C® connector 16-pin	Wurth	629722000214
41	1	P6 solder together	P_solderingoption_02	Solder bridge, keep close	N.M.	N.M.
42	1	P10 keep open	P_solderingoption_02	Solder bridge, keep open	N.M.	N.M.
43	1	P14	HDR1X16	Header 16	Harwin	M20-9991645
44	1	R1	5k1, R_0402, 62.5m W, 1%	Resistor 1%	Yageo	RC0402FR-075K1L
45	1	R2	10k, R_0402, 62.5m W, 1%	Resistor 1%	Yageo	RC0402FR-0710KL
46	1	R3	220k, R_0402, 62.5m W, 1%	Resistor 1%	Yageo	RC0402FR-07220KL
47	1	R5	100R, R_0402, 62.5m W, 5%	Resistor	Yageo	AR0402JR-07100RL
48	1	R20	560k, R_0402, 62.5m W, 1%	Resistor 1%	Yageo	RC0402FR-07560KL
49	1	R21	47k, R_0402, 62.5m W, 1%	Resistor 1%	Yageo	RC0402FR-0747KL
50	2	R25, R26	N.M., R_0402, 5%	Resistor	N.M.	N.M.
51	1	S1	SWITCH_P-DT2112C	SW-SPST	DIPTRONIC	P-DT2112C
52	1	TP5	N.M., J_TESTPOINT_1.1mm	PIN	N.M.	N.M.
53	1	TP7	N.M., P_SMD_TEST_12	Test Point	N.M.	N.M.
54	1	U2	B_72B_240um	STWBC86_Bump 72	STMicroelectronic s	STWBC86
55	1	U3	U_WQFN28	FT260Q	FTDI	FT260Q-T

Item	Q.ty	Ref.	Value	Description	Manufacturer	Part number
56	1	U4	V_SOT23-6L	USBLC6-2SC6, SOT23-6L	STMicroelectronics	USBLC6-2SC6
57	4	Jumpers short SDA, SCL, INT, and POWER coming from USB Type-C® (in the center of 2x3 header)		Short SDA, SCL, INT, and POWER coming from USB Type-C® (in the center of 2x3 header)	Wurth	60900213421
58	1	Both sides tape. Total length 32m, means 220 PCBs (dimension 12.7x13mm) , 1mm thickness			3M	5958FR(1/2"X36YD)
59	6	Standoff Material requirement UL94 - V0 or V1		Standoff, hex metric, M3	Wurth	970080365
60	4	Untreated standoff Material requirement UL94 - V0 or V1		Standoff, M3, untreated	Wurth	960060042
61	4	Screw with flat head M3 Material requirement UL94 - V0 or V1		Screw M3 with flat head	Essentra	50M030050I016
62	2	Screw M3 Material requirement UL94 - V0 or V1		Screw M3	Wurth	97790603111
63	1	Custom spacer STEVAL-WBC86TX rev A Material requirement UL94 - V0 or V1			Vendor for plexi	Vendor for plexi

## 9 STEVAL-WBC86TX PCB layout

Figure 73. Top layer

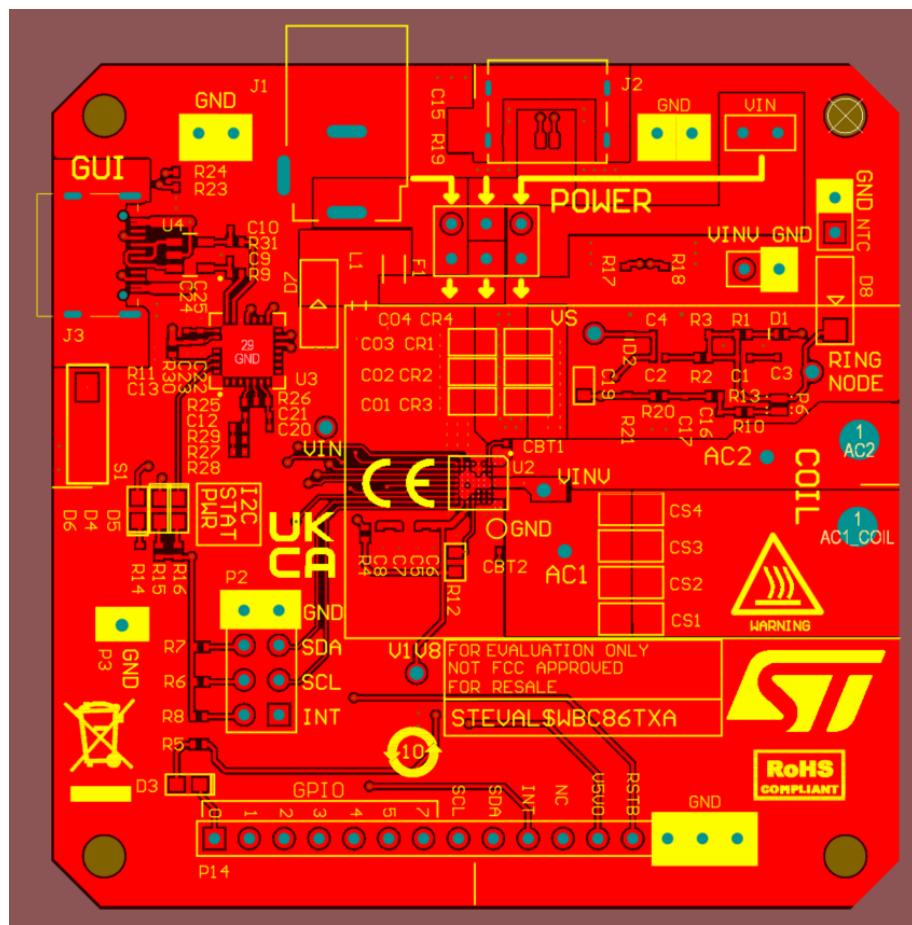


Figure 74. Inner1 layer

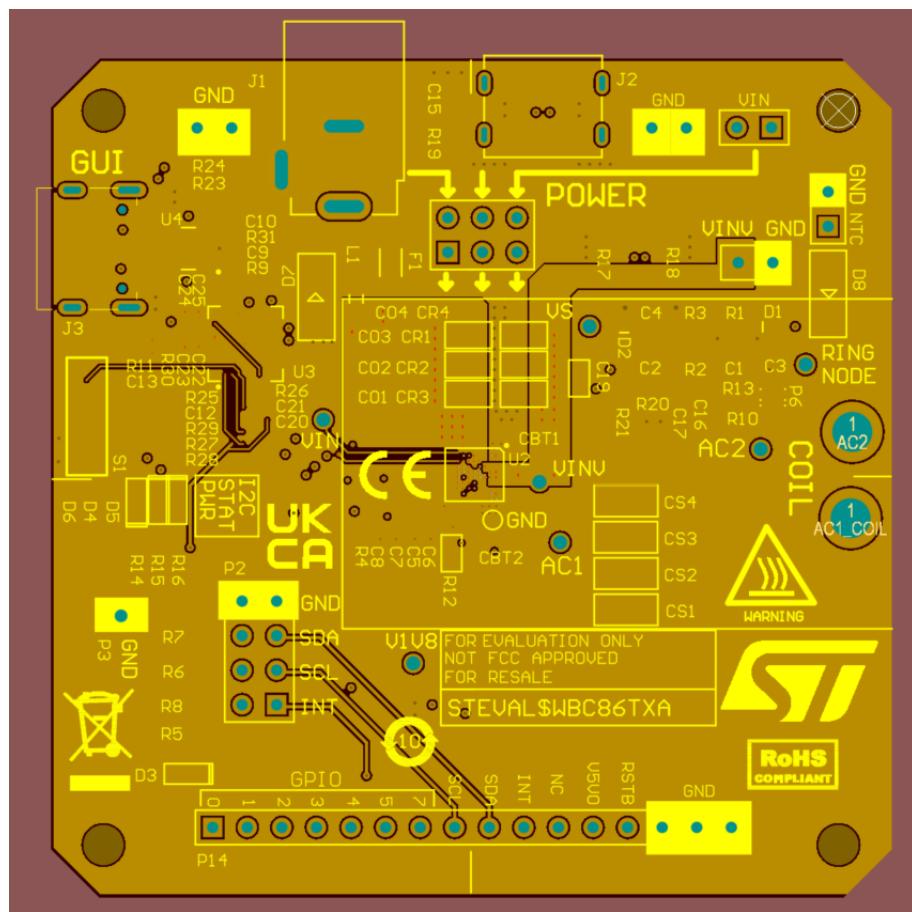


Figure 75. Inner2 layer

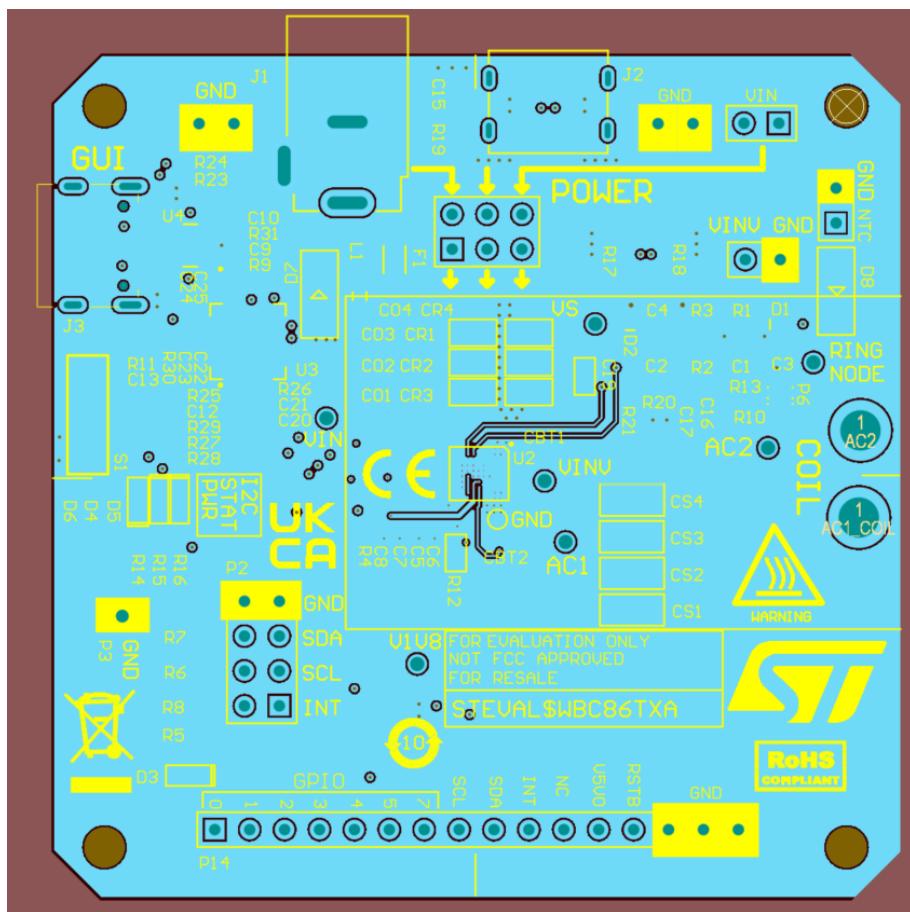
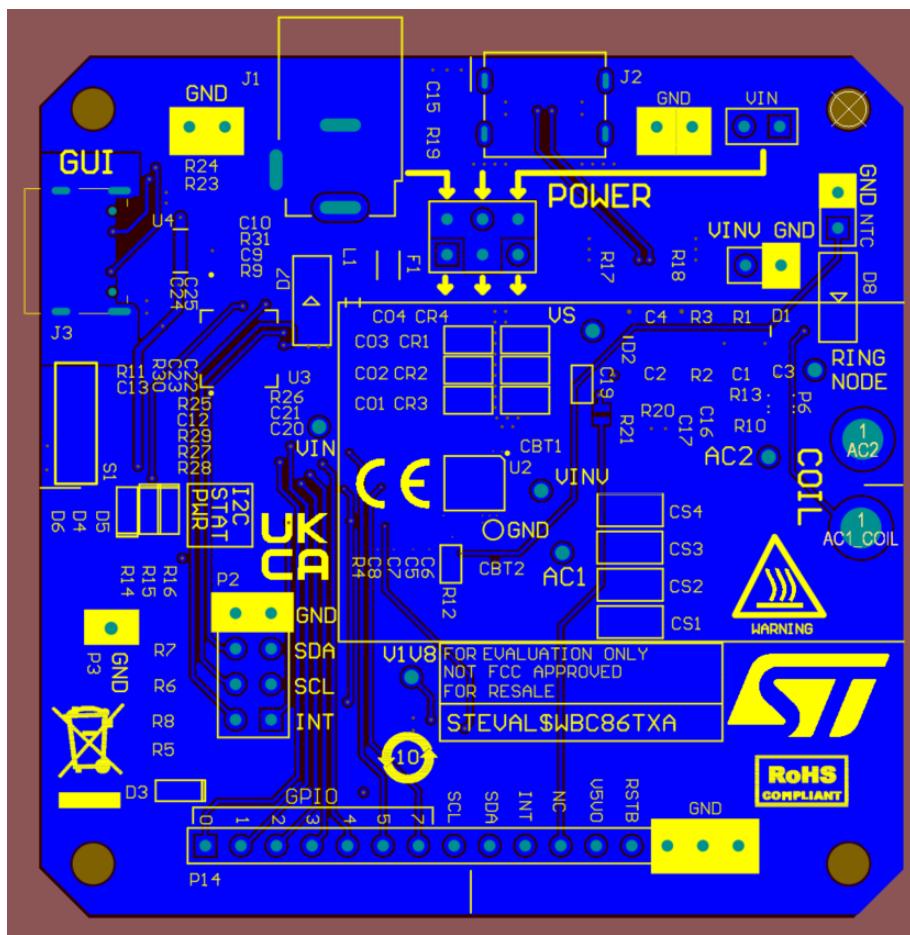


Figure 76. Bottom layer



## 10 Board versions

**Table 10. STEVAL-WBC86TX versions**

Finished good	Schematic diagrams	Bill of materials
STEVAL\$WBC86TXA <sup>(1)</sup>	STEVAL\$WBC86TXA schematic diagrams	STEVAL\$WBC86TXA bill of materials

1. This code identifies the STEVAL-WBC86TX evaluation board first version.

## 11 Regulatory compliance information

### Formal Product Notice Required by FCC:

For evaluation only; not FCC approved for resale

FCC NOTICE - This kit is designed to allow:

(1) Product developers to evaluate electronic components, circuitry, or software associated with the kit to determine

whether to incorporate such items in a finished product and

(2) Software developers to write software applications for use with the end product.

This kit is not a finished product and when assembled may not be resold or otherwise marketed unless all required FCC equipment authorizations are first obtained. Operation is subject to the condition that this product not cause harmful interference to licensed radio stations and that this product accept harmful interference. Unless the assembled kit is designed to operate under part 15, part 18 or part 95 of this chapter, the operator of the kit must operate under the authority of an FCC license holder or must secure an experimental authorization under part 5 of this chapter 3.1.2.

### Formal Product Notice Required by Industry Canada

For evaluation purposes only. This kit generates, uses, and can radiate radio frequency energy and has not been tested for compliance with the limits of computing devices pursuant to Industry Canada (IC) rules.

À des fins d'évaluation uniquement. Ce kit génère, utilise et peut émettre de l'énergie radiofréquence et n'a pas été testé pour sa conformité aux limites des appareils informatiques conformément aux règles d'Industrie Canada (IC).

### Notice for the European Union

The kit STEVAL-WBC86TX is in conformity with the essential requirements of the Directive 2014/53/EU (RED) and of the Directive 2015/863/EU (RoHS). Applied harmonized standards are listed in the EU Declaration of Conformity. Compliance to EMC standards in Class A (industrial intended use).

### Notice for the United Kingdom

The kit STEVAL-WBC86TX is in compliance with the UK Radio Equipment Regulations 2017 (UK SI 2017 No. 1206 and amendments) and with the Restriction of the Use of Certain Hazardous Substances in Electrical and Electronic Equipment Regulations 2012 (UK SI 2012 No. 3032 and amendments). Applied standards are listed in the UK Declaration of Conformity. Compliance to EMC standards in Class A (industrial intended use).

## Appendix A Thermal measurements

Thermal measurements on the Board tested at 25°C ambient temperature.

**Figure 77.** Temperature Before Power Transfer Maximum 25°C on STEVAL-WBC86TX

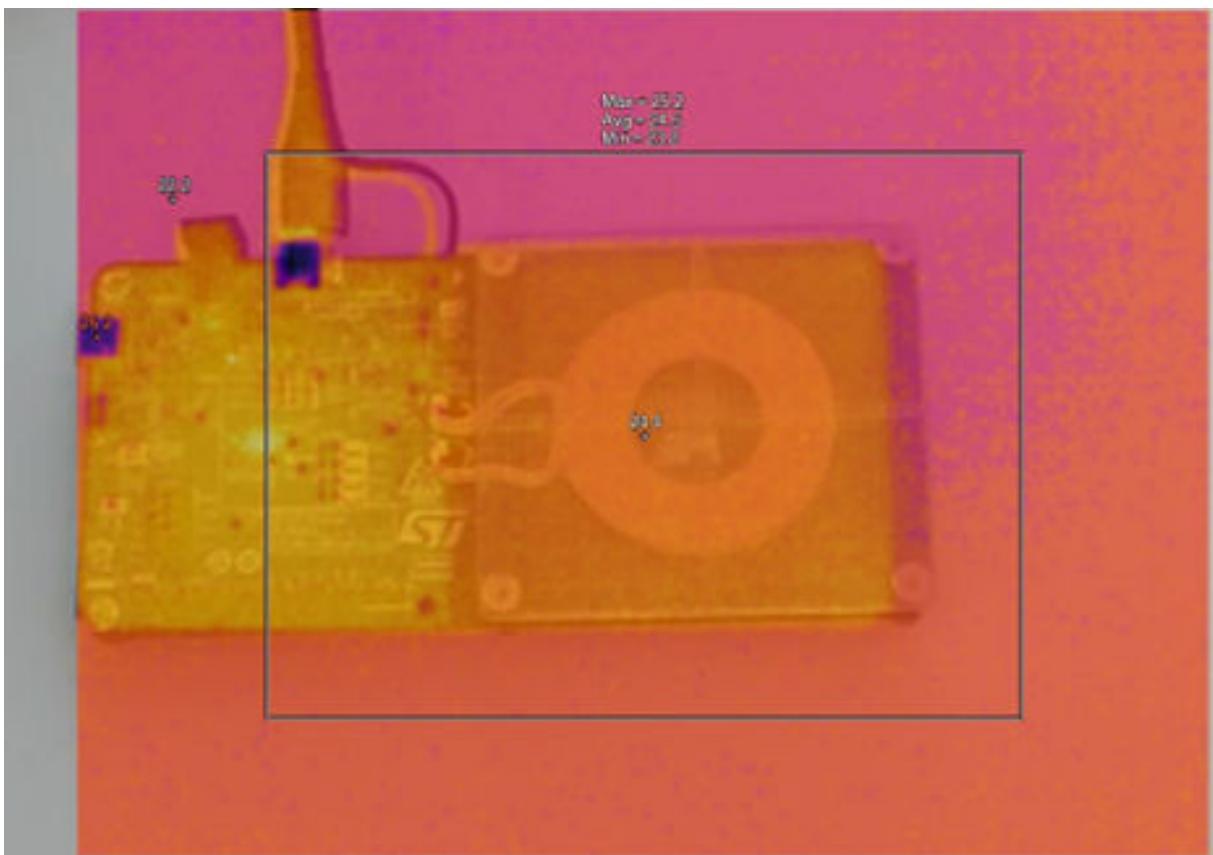


Figure 78. Maximum Temperature of the Board after 2hours (40.8°C) - 5W power transfer STEVAL-WLC38RX

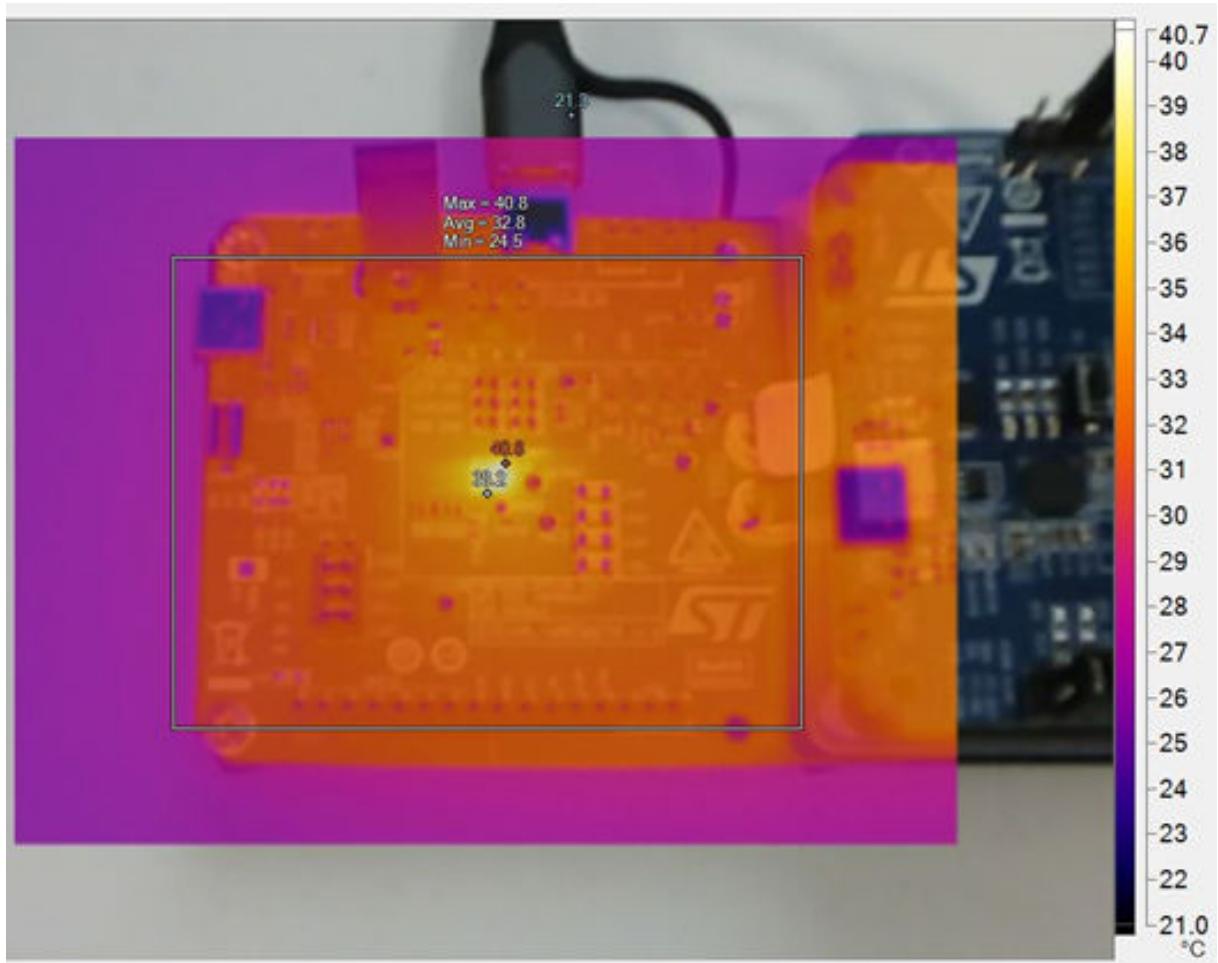
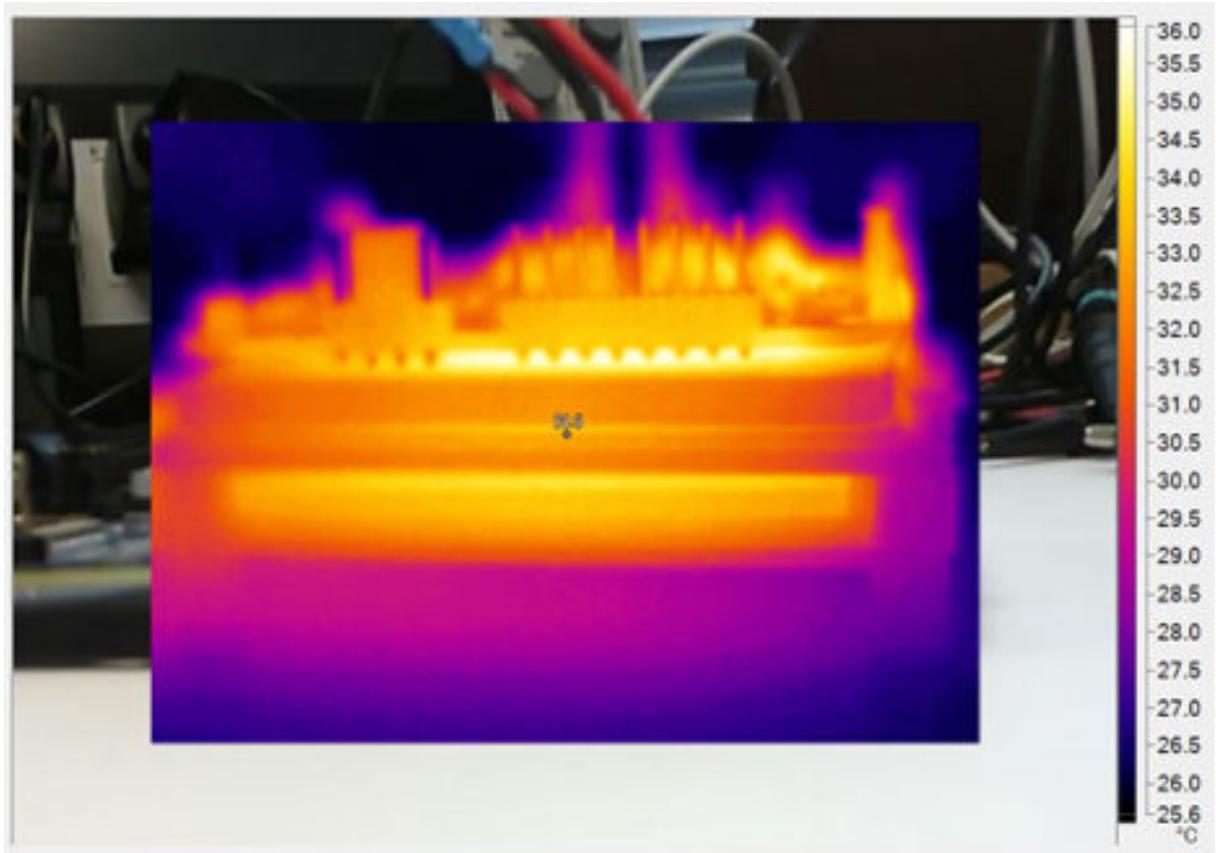


Figure 79. Maximum Temperature of the Board and Plastic case after 2hours (33.5°C) - 5W power transfer



Note: *The ambient temperature should not exceed 65°C*

## Revision history

**Table 11. Document revision history**

Date	Revision	Changes
20-Jul-2023	1	Initial release.

## Contents

<b>1</b>	<b>Reference design specifications</b>	<b>3</b>
<b>2</b>	<b>Overview of the board</b>	<b>4</b>
<b>2.1</b>	Test points	5
<b>2.2</b>	Basic operating modes	6
<b>3</b>	<b>Graphical user interface (GUI)</b>	<b>7</b>
<b>3.1</b>	Connecting STWBC86 to PC GUI	7
<b>3.2</b>	Patch and Configuration files	8
<b>3.3</b>	Configuration file generation	9
<b>3.4</b>	Header file	10
<b>3.5</b>	Header file generation	11
<b>3.6</b>	Programming the device	13
<b>4</b>	<b>Device description and operation</b>	<b>15</b>
<b>4.1</b>	System block diagram	15
<b>4.2</b>	Integrated power inverter	15
<b>4.3</b>	ADC	16
<b>4.4</b>	LDOs	17
<b>4.5</b>	Power-up sequence	17
<b>4.6</b>	UVLO	17
<b>4.7</b>	Chip reset	17
<b>4.8</b>	Protections overview	17
<b>4.8.1</b>	Overcurrent protection (OCP)	17
<b>4.8.2</b>	Ovvoltage protection (OVP)	18
<b>4.8.3</b>	Overtemperature protection (OVTP)	19
<b>4.8.4</b>	NTC	19
<b>4.9</b>	Foreign object detection (FOD)	21
<b>4.9.1</b>	FOD tuning	21
<b>4.10</b>	EPT reason	30
<b>4.11</b>	WPC Qi wireless power transfer	32
<b>4.11.1</b>	Wireless power interface	33
<b>4.12</b>	Bidirectional communication	33
<b>4.12.1</b>	ASK communication	35
<b>4.12.2</b>	FSK communication	37
<b>4.12.3</b>	Most common Qi communication packets	38
<b>4.13</b>	I2C interface	39
<b>4.13.1</b>	Data validity	39

<b>4.13.2</b>	Start and stop conditions . . . . .	39
<b>4.13.3</b>	Byte format . . . . .	39
<b>4.13.4</b>	Interface protocol . . . . .	39
<b>4.13.5</b>	Writing to a single register . . . . .	39
<b>4.13.6</b>	Writing to multiple registers with incremental addressing . . . . .	40
<b>4.13.7</b>	Reading from a single register . . . . .	40
<b>4.13.8</b>	Reading from multiple registers with incremental addressing . . . . .	41
<b>4.14</b>	GPIOx and INTB pins . . . . .	41
<b>4.15</b>	Interrupt registers . . . . .	41
<b>4.16</b>	Frequency hopping (fhop) . . . . .	42
<b>4.17</b>	PID tuning . . . . .	44
<b>5</b>	<b>STEVAL-WBC86TX description and operation . . . . .</b>	<b>45</b>
<b>5.1</b>	STWBC86 default configuration . . . . .	45
<b>5.2</b>	Typical performance characteristics . . . . .	46
<b>5.2.1</b>	Power-up waveforms . . . . .	46
<b>5.2.2</b>	Efficiency and spatial freedom in the XY plane . . . . .	47
<b>5.2.3</b>	Efficiency and spatial freedom in the Z-axis . . . . .	48
<b>5.2.4</b>	Thermal performance . . . . .	49
<b>6</b>	<b>Designing a 5 W wireless power transmitter based on the STEVAL-WBC86TX evaluation board . . . . .</b>	<b>50</b>
<b>6.1</b>	External component design . . . . .	50
<b>6.1.1</b>	Coil selection . . . . .	50
<b>6.1.2</b>	Series resonant capacitor selection . . . . .	50
<b>6.1.3</b>	Vin and Vinv capacitors selection . . . . .	50
<b>6.1.4</b>	Hardware input protections . . . . .	50
<b>6.2</b>	Ring node voltage sensing . . . . .	51
<b>6.3</b>	PCB layout guidelines . . . . .	51
<b>6.4</b>	Reference code with STM32 development boards . . . . .	52
<b>6.4.1</b>	Hardware requirements . . . . .	52
<b>6.4.2</b>	Hardware connections . . . . .	52
<b>6.4.3</b>	Software requirements . . . . .	53
<b>6.4.4</b>	Source files . . . . .	53
<b>6.4.5</b>	Reference code porting procedure . . . . .	53
<b>6.4.6</b>	API . . . . .	54
<b>6.4.7</b>	Error codes . . . . .	54
<b>7</b>	<b>Schematic diagrams . . . . .</b>	<b>55</b>
<b>8</b>	<b>Bill of materials . . . . .</b>	<b>59</b>

---

<b>9</b>	<b>STEVAL-WBC86TX PCB layout</b>	<b>62</b>
<b>10</b>	<b>Board versions</b>	<b>66</b>
<b>11</b>	<b>Regulatory compliance information</b>	<b>67</b>
<b>Appendix A</b>	<b>Thermal measurements</b>	<b>68</b>
<b>Revision history</b>		<b>71</b>
<b>List of tables</b>		<b>75</b>
<b>List of figures</b>		<b>76</b>

## List of tables

<b>Table 1.</b>	STEVAL-WBC86TX target specifications . . . . .	3
<b>Table 2.</b>	Connectors and test points . . . . .	6
<b>Table 3.</b>	GPIO functions . . . . .	41
<b>Table 4.</b>	Basic parameters . . . . .	45
<b>Table 5.</b>	Charging performance . . . . .	46
<b>Table 6.</b>	Signal legend for power up waveform . . . . .	46
<b>Table 7.</b>	Pin connection between host (STM32) and STWBC86 . . . . .	52
<b>Table 8.</b>	Error codes with descriptions . . . . .	54
<b>Table 9.</b>	STEVAL-WBC86TX bill of materials . . . . .	59
<b>Table 10.</b>	STEVAL-WBC86TX versions . . . . .	66
<b>Table 11.</b>	Document revision history . . . . .	71

## List of figures

<b>Figure 1.</b>	STEVAL-WBC86TX board . . . . .	1
<b>Figure 2.</b>	STEVAL-WBC86TX evaluation board features . . . . .	4
<b>Figure 3.</b>	Connectors and test points . . . . .	5
<b>Figure 4.</b>	GUI connection . . . . .	7
<b>Figure 5.</b>	GUI device connection . . . . .	8
<b>Figure 6.</b>	Generation of configuration file . . . . .	9
<b>Figure 7.</b>	Version of the configuration file . . . . .	9
<b>Figure 8.</b>	Saving of the configuration file . . . . .	10
<b>Figure 9.</b>	Header generator - chip selection . . . . .	11
<b>Figure 10.</b>	Header generation - pop up window . . . . .	11
<b>Figure 11.</b>	Header generation - save header file . . . . .	12
<b>Figure 12.</b>	Programming the device by header file . . . . .	13
<b>Figure 13.</b>	Generating the header file by patch and configuration files . . . . .	14
<b>Figure 14.</b>	System block diagram . . . . .	15
<b>Figure 15.</b>	H-bridge mode settings . . . . .	16
<b>Figure 16.</b>	GUI charts monitoring . . . . .	16
<b>Figure 17.</b>	OCP settings . . . . .	18
<b>Figure 18.</b>	OVP settings . . . . .	18
<b>Figure 19.</b>	OVTP settings . . . . .	19
<b>Figure 20.</b>	NTC connection on board . . . . .	20
<b>Figure 21.</b>	NTC settings . . . . .	20
<b>Figure 22.</b>	Disable FO detection . . . . .	21
<b>Figure 23.</b>	Add a new dataset . . . . .	22
<b>Figure 24.</b>	Display dataset in plot . . . . .	22
<b>Figure 25.</b>	Enable auto scroll . . . . .	22
<b>Figure 26.</b>	Additional dataset options . . . . .	23
<b>Figure 27.</b>	Line properties - dashed/solid . . . . .	23
<b>Figure 28.</b>	Line properties - color . . . . .	23
<b>Figure 29.</b>	Logged data . . . . .	24
<b>Figure 30.</b>	Delete a single data point . . . . .	25
<b>Figure 31.</b>	Spot wrong data point in plot . . . . .	25
<b>Figure 32.</b>	Plot logged data . . . . .	26
<b>Figure 33.</b>	Ploss threshold . . . . .	27
<b>Figure 34.</b>	CTC threshold . . . . .	27
<b>Figure 35.</b>	Unsuitable parameter values highlighted in red . . . . .	28
<b>Figure 36.</b>	Load tuned parameters . . . . .	29
<b>Figure 37.</b>	Enable FOD and set FOD debounce . . . . .	29
<b>Figure 38.</b>	EPT reason . . . . .	30
<b>Figure 39.</b>	Received message . . . . .	31
<b>Figure 40.</b>	EPT interrupt setting . . . . .	31
<b>Figure 41.</b>	Protection debounce setting . . . . .	32
<b>Figure 42.</b>	EPT reping settings . . . . .	32
<b>Figure 43.</b>	Power transfer start up sequence . . . . .	33
<b>Figure 44.</b>	Amplitude modulation of the Power Signal . . . . .	35
<b>Figure 45.</b>	Example of a differential bi-phase encoding scheme . . . . .	36
<b>Figure 46.</b>	Example of the asynchronous serial format . . . . .	36
<b>Figure 47.</b>	ASK communication example - high load current . . . . .	36
<b>Figure 48.</b>	ASK communication - no load . . . . .	37
<b>Figure 49.</b>	Example of differential bi-phase encoding . . . . .	37
<b>Figure 50.</b>	Format of the three defined responses . . . . .	38
<b>Figure 51.</b>	Writing to a single register . . . . .	40
<b>Figure 52.</b>	Writing to multiple registers . . . . .	40
<b>Figure 53.</b>	Reading from a single register . . . . .	40

<b>Figure 54.</b>	Reading from multiple registers . . . . .	41
<b>Figure 55.</b>	Interrupt registers . . . . .	42
<b>Figure 56.</b>	FHOP settings . . . . .	43
<b>Figure 57.</b>	FHOP example . . . . .	43
<b>Figure 58.</b>	PID parameters . . . . .	44
<b>Figure 59.</b>	Power up waveform . . . . .	46
<b>Figure 60.</b>	X, Y axis for misalignment . . . . .	47
<b>Figure 61.</b>	Efficiency curve in X axis . . . . .	48
<b>Figure 62.</b>	Efficiency curve in Y axis . . . . .	48
<b>Figure 63.</b>	Efficiency curve in Z axis . . . . .	49
<b>Figure 64.</b>	Thermal performance . . . . .	49
<b>Figure 65.</b>	Ring node voltage sensing . . . . .	51
<b>Figure 66.</b>	Ring node divider configuration . . . . .	51
<b>Figure 67.</b>	Pin connection between host and STWBC86 chip/evaluation board . . . . .	52
<b>Figure 68.</b>	Connectivity . . . . .	53
<b>Figure 69.</b>	STEVAL-WBC86TX circuit schematic (1 of 4) . . . . .	55
<b>Figure 70.</b>	STEVAL-WBC86TX circuit schematic (2 of 4) . . . . .	56
<b>Figure 71.</b>	STEVAL-WBC86TX circuit schematic (3 of 4) . . . . .	57
<b>Figure 72.</b>	STEVAL-WBC86TX circuit schematic (4 of 4) . . . . .	58
<b>Figure 73.</b>	Top layer . . . . .	62
<b>Figure 74.</b>	Inner1 layer . . . . .	63
<b>Figure 75.</b>	Inner2 layer . . . . .	64
<b>Figure 76.</b>	Bottom layer . . . . .	65
<b>Figure 77.</b>	Temperature Before Power Transfer Maximum 25°C on STEVAL-WBC86TX . . . . .	68
<b>Figure 78.</b>	Maximum Temperature of the Board after 2hours (40.8°C) - 5W power transfer STEVAL-WLC38RX . . . . .	69
<b>Figure 79.</b>	Maximum Temperature of the Board and Plastic case after 2hours (33.5°C) - 5W power transfer . . . . .	70

**IMPORTANT NOTICE – READ CAREFULLY**

STMicroelectronics NV and its subsidiaries ("ST") reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST's terms and conditions of sale in place at the time of order acknowledgment.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of purchasers' products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. For additional information about ST trademarks, refer to [www.st.com/trademarks](http://www.st.com/trademarks). All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2023 STMicroelectronics – All rights reserved