WPRSWLIBUG

Wireless Power Receiver SW Library User's Guide

Rev. 3.0 — 4 July 2022 User Guide

Contents

1 Read Me First

This document describes the API of the Wireless Power Receiver library. The library enables users to evaluate the Qi wireless charging solution easily in customer applications.

This document describes the library interface and software features, and enables users to develop customized applications based on the WPR library.

2 Abbreviations

The following table provides the abbreviation descriptions in this document.

Table 1. Abbreviations

Abbreviation	Description
WPR	Wireless Power Receiver
DC	Direct Current
JTAG	Joint Test Action Group
SCI	Serial Communications Interface
EMI	Electro-Magnetic Interference
UART	Universal Asynchronous Receiver / Transmitter
TX	Transmitter
ASK	Amplitude-Shift Keying
FSK	Frequency-Shift Keying
BPP	Baseline Power Profile
EPP	Extended Power Profile
FOD	Foreign Object Detection
HW	Hardware
RX	Receiver

Table continues on the next page...

Table 1. Abbreviations (continued)

Abbreviation	Description
DDM	Digital Demodulation Module
ADC	Analog-to-Digital Converter
HAL	Hardware Abstraction Layer
NVM	Non-Volatile Memory
MCU	Microcontroller Unit
EPT	End Power Transfer
CEP	Control Error Packet

3 Overview

3.1 WPR software layers

The following figure shows the WPR library software layers.

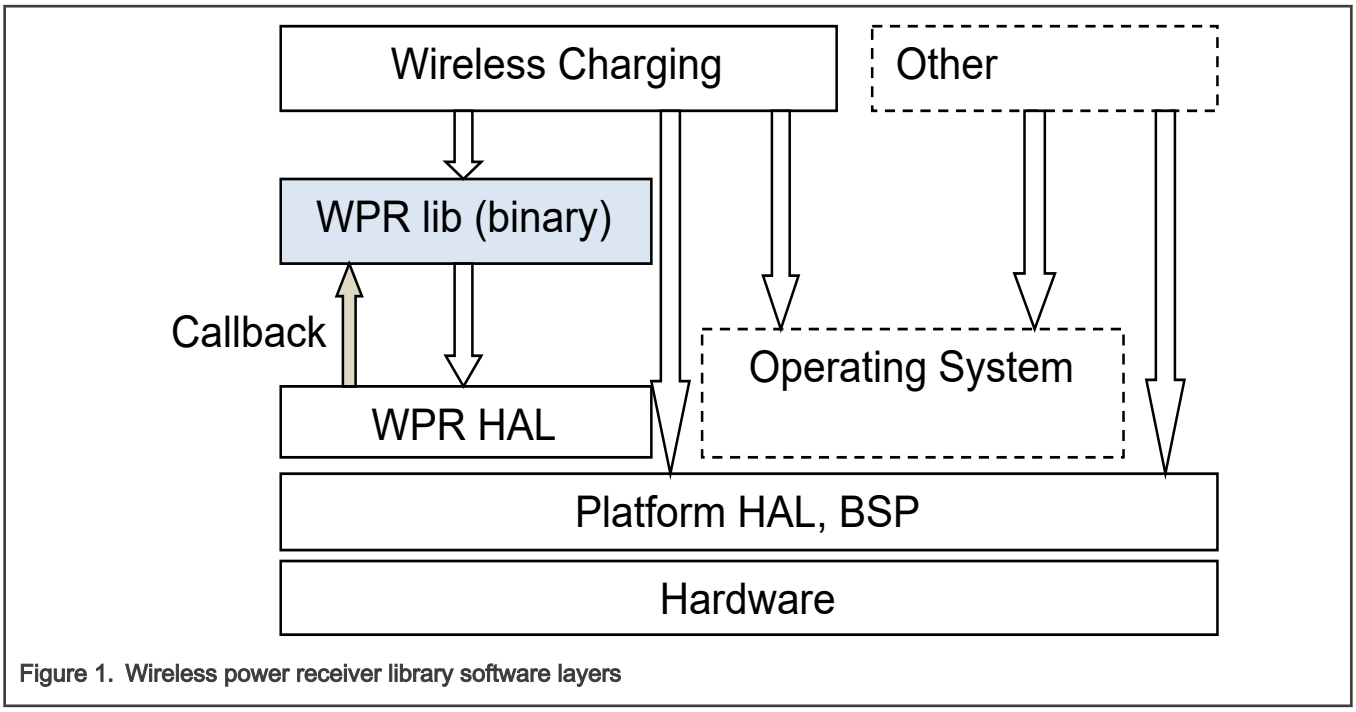

Figure 1. Wireless power receiver library software layers

The WPR library is provided as a binary format, while the application and Board Support Package (BSP) are in the source format.

The main modules in the WPR library include:

- BPP and EPP Qi state machine
- Voltage/Current sampling and protection
- Received power calculation and FOD support

The WPR library API and WPR Hardware Abstraction Layer (HAL) API are provided in the source format, with main functions like:

- WPR library API
  - Library version retrieval
  - Library initialization
  - Library main entry function
  - Callbacks such as Qi communication interrupt callback
- WPR HAL API
  - Control and communication related HAL
  - Timer-related HAL
  - Voltage/Current sensing HAL

3.1.1 WPR software dynamics

There are two timer interrupt service routines in the wireless power receiver software. The first one serves for system tick and Qi packet sending timing control. The second one is used for the FSK demodulation. If the FSK demodulation timer driver is not available, the library can also run as the BPP power receiver and can work with the BPP power transmitter. The following figure shows the software dynamics related to the library.

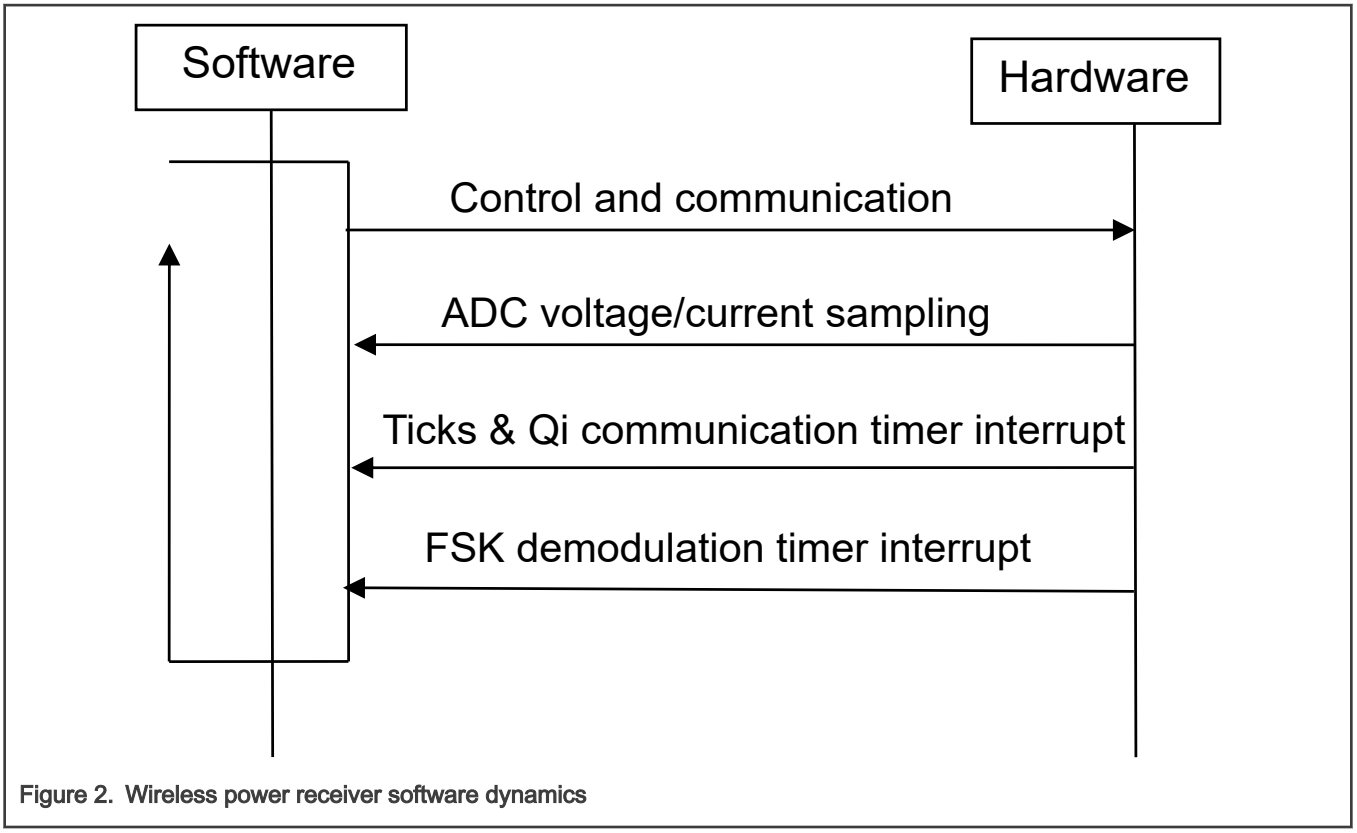

Figure 2. Wireless power receiver software dynamics

For one instance:

- The main loop performs all the Qi functions like state machine, voltage/current monitor, received power calculation, and so on. The frequency detection module chooses which state machine to run when the RX is powered on.
- The monitor function samples the rectified voltage, output voltage, output current, and temperature every 250 µs.
- The ticks and Qi communication timer needs to generate interrupt every 250 µs. This interrupt service routine includes system ticks, timeout count, control communication pin for sending the packet, and so on.

- The FSK demodulation timer is enabled when the RX needs to receive response from the transmitter. This FSK demodulation timer interrupt service routine is a part of the driver. The WPR library provides only a handle function to process the packet received by the FSK demodulation timer.

4 WPR Library API

The wireless power receiver software library is provided as a binary library. It includes APIs for the customer to build a wireless charging application. The APIs are divided into two major parts: application API and HAL API.

4.1 Error types

In the state machine, the charging state monitor checks the system state every 100 ms. If one of the values is abnormal and sustains for specified time, the system error code is generated and certain protection becomes active. The application can get this error through the API. The following table lists the system error codes and corresponding End Power Transfer codes. If the monitor detects some system errors and this state sustains for a certain period, RX sends the EPT packet to TX. Currently, the WPR code does not generate the `SYS_ERR_OUT_DISCONNECT` code.

Table 2. System error codes and End Power Transfer codes

System error code	Remark	EPT code
`SYS_ERR_NONE`	Normal	None
`SYS_ERR_OUT_DISCONNECT`	Output current is less than the limit.	Charge complete
`SYS_ERR_OUT_OVER_VOLTAGE`	Output voltage is higher than the limit.	Over voltage
`SYS_ERR_OUT_UNDER_VOLTAGE`	Output voltage is less than the limit.	Internal fault
`SYS_ERR_OUT_OVER_CURRENT`	Output current is higher than the limit.	Over current
`SYS_ERR_DCDC_NO_POWERGOOD`	Output of DCDC chip is abnormal.	Internal fault
`SYS_ERR_RECT_UNDER_VOLTAGE`	Rectified voltage is lower than the limit.	None
`SYS_ERR_RECT_OVER_VOLTAGE`	Rectified voltage is higher than the limit.	Over voltage
`SYS_ERR_TEMP_OVER_HEAT`	Temperature is higher than the limit.	Over temperature
`SYS_ERR_LDO_MOSFET_OVERLOAD`	The power dissipation on external MOSFET is too large.	Internal fault
`SYS_ERR_APP_TURN_OFF`	Application sends an EPT packet.	Charge complete

In the negotiation phase, any abnormal will result in an error code. Customer can debug the FSK demodulation function by using this error code. The following table lists the FSK demodulation error codes. If the negotiation fails, RX retries 3 times by default and then sends the EPT packet with the negotiation failure code (0x0A).

Table 3. FSK demodulation error codes definition and remark

Definition	Remark
#define FSK_ERR_NONE 0x00	Normal
#define FSK_ERR_ENTER_NEGO_ACK_TIME_OUT 0x01	No response from TX at the end of configuration packet

Table continues on the next page...

Table 3. FSK demodulation error codes definition and remark (continued)

Definition	Remark
#define FSK_ERR_TX_IDE_HEADER_MISMATCH 0x02	Request IDE packet failure
#define FSK_ERR_TX_CFG_HEADER_MISMATCH 0x03	Request CFG packet failure
#define FSK_ERR_MSG_HEADER_TIME_OUT 0x04	Timeout to receive the header of a packet
#define FSK_ERR_MSG_CHECKSUM_TIME_OUT 0x05	Timeout to receive the whole packet
#define FSK_ERR_RETRY_EXCEED_LIMIT 0x06	Retry times limit exceeded (three times)
#define FSK_ERR_TX_RESP_TIME_TOO_LONG 0x07	TX response time exceeds limit

4.2 Application API functions

4.2.1 qi_get_library_version

Prototype:

```
uint16_t qi_get_library_version(void)
```

Description:

Gets the WPR library version.

Return:

`uint16_t` type with value format `x.y.z`: 4-bit x, 4-bit y, and 8-bit z. For example, 0x3000 means version 3.0.0.

4.2.2 qi_system_init

Prototype:

```
void qi_system_init (void)
```

Description:

Initializes the WPR library. It will initialize the communication and control parameters, reset state machine internal states.

Return:

None.

4.2.3 qi_system_sample

Prototype:

```
void qi_system_sample (void)
```

Description:

Sample all the ADC channel by calling `system_sample_channel` HAL API function. This function needs to be called by application every 250 µs.

Return:

None.

4.2.4 qi_charging_state_machine

Prototype:

```
void qi_charging_state_machine (void)
```

Description:

Main entry function of the WPR library. Make sure this function will be called within 1 ms interval, which is required for timing check.

Return:

None.

4.2.5 qi_get_charging_phase

Prototype:

```
uint8_t qi_get_charging_phase(void)
```

Description:

Gets the current phase in the state machine.

Return:

Returns the charging phases in the state machine, including values listed in the following table.

Table 4. Charging phases in the state machine

Phase	Description	Remark
QI_PHASE_SELECTION	Selection phase	Default phase when startup.
QI_PHASE_PING	Ping phase	-
QI_PHASE_IDE_CFG	Identification & Configuration phase	-
QI_PHASE_NEGOTIATION	Negotiation/Renegotiation phase	-
QI_PHASE_PWR_TRANS	Power transfer/calibration phase	Calibration is a part of the power transfer phase.

4.2.6 qi_get_time_second

Prototype:

```
uint32_t qi_get_time_second(void)
```

Description:

Gets second ticks of the charging state machine.

Return:

Returns second ticks value.

4.2.7 qi_get_time_millisecond

Prototype:

```
uint32_t qi_get_time_millisecond(void)
```

Description:

Gets millisecond ticks of the charging state machine.

Return:

Returns millisecond ticks value

4.2.8 qi_get_transmit_status

Prototype:

```
uint8_t qi_get_transmit_status(void)
```

Description:

Gets packet transmitting status.

Return:

Returns packet transmitting status. There are three statuses in the state machine, listed in the following table.

Table 5. Packet transmitting status

Status	Description	Remark
PK_TX_STOP	No packet to transmit	Voltage/Current sampling in this time.
PK_TX_DELAY	Delay some time before entering next state or sending next packet	It is used to satisfy Qi timing.
PK_TX_RUN	Packet is transmitting	-

4.2.9 qi_send_charge_status_packet

Prototype:

```
void qi_send_charge_status_packet (uint8_t level)
```

Description:

Sends charge status to the power transmitter. The charge status will not be sent to TX in the library. Customer could call this function to send charge status.

Parameters:

`level`: this value indicates the charge level of the energy storage device, as a percentage of the fully charged level. For clarity, the value **0** means an empty energy storage device, and the value **100** means a fully-charged energy storage device.

Return:

None.

4.2.10 qi_send_end_power_packet

Prototype:

```
void qi_send_end_power_packet (uint8_t reason)
```

Description:

Sends end power transfer packet to the power transmitter when receiver has entered power transfer phase.

Parameters:

`reason`: this value indicates end power code, listed in the following table.

Return:

None.

Table 6. End Power Transfer values

EPT code	Description	Remark
EPC_UNKNOWN	No specific reason.	-
EPC_CHARGE_COMPLETE	The battery of Rx is fully charged.	-
EPC_INTERNAL_FAULT	The power good signal from DCDC chip is abnormal or output voltage exceeds the lower limit.	May be sent in library
EPC_OVER_TEMPERATURE	The temperature exceeds the upper limit.	May be sent in library
EPC_OVER_VOLTAGE	Output voltage or rectified voltage exceeds the upper limit.	May be sent in library
EPC_OVER_CURRENT	Output current exceeds the upper limit.	May be sent in library
EPC_BATTERY_FAILURE	The battery of Rx has some problems.	-
EPC_NO_RESPONSE	TX does not respond to Control Error Packets as expected.	May be sent in library
EPC_NEGO_FAILURE	RX cannot negotiate with TX successfully.	May be sent in library
EPC_RESTART_PWR_TRANS	RX receive NAK response from TX after sending received power packet with mode 0x00.	-

4.2.11 qi_send_renegotiation_packet

Prototype:

```
void qi_send_renegotiation_packet (uint8_t params_mask)
```

Description:

Sends the renegotiation packet to the power transmitter. This packet will not be sent to TX in the library. The user can call this function to send renegotiation packet.

Parameters:

`params_mask`: this parameter is a bit mask value. Three bits indicate whether three parameters need to be negotiated or not. These parameters should be prepared in the `g_nego_params` structure and then use this API. The following table lists the parameters that can be renegotiated.

Table 7. Bit mask for renegotiation packet

Bit mask	Description	Remark
NEGO_GUARANTEED_POWER_MASK	Guaranteed Power packet	-
NEGO_FSK_PARAMETERS_MASK	FSK Parameters packet	Include polarity and depth
NEGO_MAXIMUM_POWER_MASK	Maximum Power packet	-

Return:

None.

The result of this renegotiation request can be found in variable `renegotiation_status`. The following table lists the result.

Table 8. Result of renegotiation request

Bit mask	Description	Remark
RENEGO_STS_NONE	Initial state	-
RENEGO_STS_SEND_FAILURE	No response from TX	The result when sending renegotiation packet
RENEGO_STS_RESP_NAK	TX response NAK	
RENEGO_STS_RESP_ND	TX response ND	
RENEGO_STS_NEGO_SUCCESS	TX response ACK and renegotiation phase complete	The value does not indicate that TX accepts the renegotiation parameters

4.2.12 qi_send_proprietary_packet

Prototype:

```
void qi_send_proprietary_packet (uint8_t header, uint8_t *pkbuf)
```

Description:

Sends the proprietary packet to the power transmitter. The size of the array pkbuf is related to the header.

Parameters:

`header`: the header of proprietary packet.

`pkbuf`: the content of this proprietary packet.

Return:

None

4.2.13 fsk_normal_packet_byte_handle

Prototype:

```
void fsk_normal_packet_byte_handle (uint8_t data)
```

Description:

Processes the data received by the FSK demodulation timer. These data are a response for the General TX Request from transmitter. This function processes one byte at a time.

Parameters:

`data`: the packet data to be processed.

Return:

None

4.2.14 fsk_pattern_packet_byte_handle

Prototype:

```
void fsk_pattern_packet_byte_handle (uint8_t data)
```

Description:

Processes the data received by FSK demodulation timer. These data are response for the Specific TX Request from the transmitter. This function processes one byte at a time.

Parameters:

`data`: the packet data to be processed.

Return:

None

4.3 HAL API functions

4.3.1 output_enable

Prototype:

```
void output_enable(void)
```

Description:

Enables the output of the receiver. It enables the DC-DC chip for the DC-DC scheme board or enables the LDO function for the LDO scheme board.

Return:

None

4.3.2 output_disable

Prototype:

```
void output_disable(void)
```

Description:

Disables the output of the receiver. It disables the DC-DC chip for the DC-DC scheme board or disables the LDO function module for the LDO scheme board.

Return:

None

4.3.3 comm_bitIO_clear

Prototype:

```
void comm_bitIO_clear (void)
```

Description:

Resets the communication pin. When a packet is sent out, the RX resets the communication pin to the initial state.

Return:

None

4.3.4 comm_bitIO_invert

Prototype:

```
void comm_bitIO_invert (void)
```

Description:

Toggles the communication pin. The power receiver uses a differential bi-phase encoding scheme to modulate data bits onto the power signal.

Return:

None

4.3.5 system_sample_channel

Prototype:

```
uint16_t system_sample_channel (uint8_t sampleChannel)
```

Description:

According to the sampleChannel parameter, this function samples the specified ADC channel and calculates the physical quantity of expected sample value according to the schematic. The unit is mV, mA, or degree centigrade.

Parameters:

`sampleChannel`: indicates which type of value the library wants to get. The following table lists the channel types.

Table 9. ADC sample channel definition

Type	Description	Remark
SAMPLE_CH_VBUS	Rectified voltage sample channel	Unit: mV
SAMPLE_CH_OUT_VOLT	Output voltage sample channel	Unit: mV
SAMPLE_CH_OUT_CURR	Output current sample channel	Unit: mA
SAMPLE_CH_TEMP	Temperature sample channel	Unit: centigrade degree

Return:

Returns a physical quantity of the voltage, current or temperature according to the parameters. The unit is mV, mA, or degree centigrade.

4.3.6 check_dcdc_status

Prototype:

```
uint8_t check_dcdc_status(void)
```

Description:

Checks the DCDC output status. If the output of DCDC chip is disabled or shorted out, the PGood pin of the chip pulls down. The library checks this pin to protect the board. When getting a zero value for some time, the library sends the End Power Transfer packet to end the power transfer. This function is only used in the DCDC output scheme.

Return:

Returns zero for short-out or DCDC disabled, one when output is OK.

4.3.7 fsk_clock_enable

Prototype:

```
void fsk_clock_enable(uint8_t flag)
```

Description:

Enables or disables the FSK demodulation timer. If the flag is equal to one, it enables the FSK timer; otherwise, it disables the FSK timer. When a packet needs to be received from the transmitter, the RX enables the FSK timer first.

Parameters:

`flag`: vaule one for enabling the clock, value zero for disabling the clock.

Return:

None

4.3.8 fsk_set_hw_packet_type

Prototype:

```
void fsk_set_hw_packet_type( uint8_t type)
```

Description:

Sets the packet type for the FSK demodulation timer to receive: normal packet or pattern packet. The parameter type indicates the received packet type, as listed in the following table.

Table 10. Demodulation packet type

Type	Description	Remark
FSK_MESSAGE_NORMAL	FSK timer receives response of General TX Request	FSK received data is 11 bits
FSK_MESSAGE_PATTERN	FSK timer receives response of Specific TX Request	FSK received data is 8 bits

Return:

None

5 WPR Library Configurations

The WPR library has two structures to configure the control and communication in the state machine: one is the `qi_system_params` structure, and the other is the `negotiation_request_params` structure.

5.1 qi_system_params structure

```
typedef struct{
uint8_t WPC_revision;
uint16_t manufCode;
uint32_t devId;
uint8_t  extId[8];
uint8_t  maxPwr;
uint8_t received_power_send_rate;
uint8_t FSKPolarity;
uint8_t FSKDepth;
uint8_t FSKNego;
uint16_t rect_voltage_ping_startup;
uint16_t rect_voltage_buck_a;
uint16_t rect_voltage_buck_b;
uint16_t rect_voltage_buck_c;
uint16_t rect_voltage_buck_d;
uint16_t rect_voltage_buck_e;
uint16_t vrec_shift_load_ab;
uint16_t vrec_shift_load_bc;
uint16_t rect_voltage_ldo_a;
uint16_t rect_voltage_ldo_b;
uint16_t rect_voltage_ldo_c;
uint16_t rect_voltage_ldo_d;
uint16_t rect_voltage_ldo_e;
uint16_t rect_voltage_ldo_f;
uint16_t rect_voltage_ldo_g;
uint16_t out_voltage_expected;
uint16_t max_rec_voltage;
uint16_t max_out_voltage;
uint16_t min_out_voltage;
uint16_t max_out_current;
uint8_t  min_out_current;
uint16_t max_temperture;
uint8_t  fod_window_size;
uint8_t  fod_window_offset;
uint8_t  mcu_input_current;
uint8_t  current_sensor;
uint16_t rect_vol_loss;
uint16_t coil_rect_resisrance;
int16_t  vol_quadratic_coeff;
int16_t  vol_linear_coeff;
int16_t  vol_offset_mw;
int16_t  ldo_quadratic_coeff;
int16_t  ldo_linear_coeff;
int16_t  ldo_offset_mw;
int16_t  ploss_buck_quadratic;
int16_t  ploss_buck_linear;
int16_t  ploss_buck_offset;
int16_t  ploss_buck_no_load_linear;
int16_t  ploss_buck_no_load_offset;
int16_t fod_offset_5W;
int16_t fod_offset_10W;
int16_t fod_offset_15W;
```

```
uint8_t time_out_of_rec_over_vol;
uint8_t time_out_of_out_vol_out_of_range;
uint8_t time_out_of_buck_no_power_good;
uint8_t time_out_of_out_over_curr;
uint8_t time_out_of_out_disconnect;
uint8_t time_out_of_out_over_temper;
uint8_t  fsk_silent;
uint8_t  fsk_header_time_out;
uint8_t  fsk_message_time_out;
uint16_t fsk_packet_time_out;
}qi_system_params;
```

Table 11. Library parameters configuration description

Name	Description	Remark
WPC_revision	WPC version value that will be sent to TX	Format x.y.z: 4-bit x, 4-bit y, and 8-bit z
manufCode	Manufacturer code, FSL is 0x28	-
devId	Basic device identifier	-
extId[8]	Extended device identifier	-
received_power_send_rate	Number of control error packet to send received power packet	-
maxPwr	Maximum expected output power	Configuration packet parameters
FSKPolarity	FSK modulation polarity	
FSKDepth	FSK modulation depth	
FSKNego	Inform TX to enter negotiation phase	
rect_voltage_ping_startup	The minimum rectified voltage that RX enter PING phase	-
rect_voltage_dcdc_a	The expected voltage when output current is lower than 200 mA	DC-DC board charging with middle power TX
rect_voltage_dcdc_b	The expected voltage when output current is lower than 500 mA	
rect_voltage_dcdc_c	The expected voltage when output current is greater or equals 500 mA	
rect_voltage_dcdc_d	The expected voltage when output current is lower than 200 mA	DC-DC board charging with low power TX
rect_voltage_dcdc_e	The expected voltage when output current is greater or equals 200 mA	
vrec_shift_load_ab	Vrec shift from a to b on DCDC board	The unit of the load is mA
vrec_shift_load_bc	Vrec shift from b to c on DCDC board	

Table continues on the next page...

Table 11. Library parameters configuration description (continued)

Name	Description	Remark
rect_voltage_ldo_a	The expected voltage when output current is lower than 50 mA	LDO board charging with middle power TX. If TX is low power transmitter, only use last four levels.
rect_voltage_ldo_b	The expected voltage when output current is lower than 100 mA	
rect_voltage_ldo_c	The expected voltage when output current is lower than 200 mA	
rect_voltage_ldo_d	The expected voltage when output current is lower than 300 mA / lower than 50 mA	
rect_voltage_ldo_e	The expected voltage when output current is lower than 400 mA / lower than 100 mA	
rect_voltage_ldo_f	The expected voltage when output current is lower than 500 mA	
rect_voltage_ldo_g	The expected voltage when output current is higher or equals 500 mA	
out_voltage_expected	Expected output voltage	-
max_rec_voltage	Maximum value of rectified voltage	-
max_out_voltage	Maximum value of output voltage	-
min_out_voltage	Minimum value of output voltage	-
max_out_current	Maximum value of output current	-
min_out_current	Minimum value of output current	-
max_temperture	Maximum value of temperature	-
fod_window_size	Time window size of received power sampling	Configuration packet parameters
fod_window_offset	Time window offset of received power sampling	
mcu_input_current	Input current of the MCU (default 17mA)	Used for FOD calibration
current_sensor	Sampling output current sensor resistance value	
rect_vol_loss	The voltage drops before and after rectifier	
coil_rect_resisrance	Impedance estimation of the coil and rectifier	
vol_quadratic_coeff	Quadratic term coefficient	Received power calculation when in light mode
vol_linear_coeff	Linear term coefficient	

Table continues on the next page...

Table 11. Library parameters configuration description (continued)

Name	Description	Remark
vol_offset_mw	Constant term	
ldo _quadratic_coeff	Quadratic term coefficient	Quadratic equation fitting for LDO board
ldo _linear_coeff	Linear term coefficient	
ldo _offset_mw	Constant term	
ploss_dcdc_quadratic	Quadratic term coefficient	Please refer to relevant document
ploss_dcdc_linear	Linear term coefficient	
ploss_dcdc_offset	Constant term	
ploss_dcdc_no_load_linear	Power loss of DCDC board when no load	
ploss_dcdc_no_load_offset		
fod_offset_5W	Additional received power offset when load less than 5W	-
fod_offset_10W	Additional received power offset when load less than 10W and larger than 5 W	-
fod_offset_15W	Additional received power offset when load larger than 10W	-
time_out_of_rec_over_vol	Rectified voltage overvoltage timeout limit	Timeout limit that will activate protection
time_out_of_out_vol_out_of_range	Output voltage out of range timeout limit	
time_out_of_dcdc_no_power_good	Power good signal of DCDC chip abnormal timeout limit	
time_out_of_out_over_curr	Output over current timeout limit	
time_out_of_out_disconnect	Output disconnect timeout limit	
time_out_of_out_over_temper	Over temperature timeout limit	
fsk_silent	Delay time before state machine enter next state when a packet is received from TX	-
fsk_header_time_out	Timeout limit to receive header of TX packet	-
fsk_message_time_out	Timeout limit to receive whole pattern packet	-
fsk_packet_time_out	Timeout limit to receive whole general packet	-

5.2 negotiation_request_params structure

```
typedef struct{
uint8_t Qtpt_Reported_value;
```

```
    uint8_t Request_Guaranteed_Power_value;
    uint8_t Select_Received_Power_Packet_value;
    uint8_t Transmitter_Modulation_Depth_value;
    uint8_t Request_Maximum_Power_value;
}negotiation_request_params;
```

Table 12. Negotiation phase specific request parameters

Name	Description	Remark
`Qtpt_Reported_value`	Q-factor value	Used in Request FOD Test Result packet
`Request_Guaranteed_Power_value`	Request guaranteed power code	10 for 5W, 20 for 10W, 30 for 15W
`Select_Received_Power_Packet_value`	Received power packet type	0x04 for 8bits, 0x31 for 24bits
`Transmitter_Modulation_Depth_value`	Transmitter modulation depth	0x00, 0x01, 0x02, 0x03
`Request_Maximum_Power_value`	Request maximum power TX need to provide	10 for 5W, 20 for 10W, 30 for 15W

5.3 WPID Packet Parameters structure

```
typedef struct{
    uint8_t organization_unique_id[3];
    uint8_t ouid_crc[2];
    uint8_t serial_id[3];
    uint8_t sid_crc[2];     // it will be calculated and filled by library automatically.
}wpid_params;
```

Table 13. WPID packets parameters

Name	Description	Remark
`organization_unique_id`	Organization ID which dispatched by WPC	-
`ouid_crc`	Organization unique ID CRC code	It will be calculated and filled by library automatically
`serial_id`	Device serial ID	-
`sid_crc`	Device serial ID CRC code	It will be calculated and filled by library automatically

6 Key Features of the Library

The library has the following key features:

- Supports Wireless Power Consortium (WPC) Qi Version 1.2.4 specification.
- Supports FSK communication signals from the EPP Transmitter.
- Support Foreign Object Detection (FOD).

7 Revision History

The following table provides the revision history.

Table 14. Revision history

Revision number	Date	Substantive changes
GA 3.0	07/2022	Initial release.

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this data sheet expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, µVision, Versatile — are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved.

Airfast — is a trademark of NXP B.V.

Bluetooth — the Bluetooth wordmark and logos are registered trademarks owned by Bluetooth SIG, Inc. and any use of such marks by NXP Semiconductors is under license.

Cadence — the Cadence logo, and the other Cadence marks found at www.cadence.com/go/trademarks are trademarks or registered trademarks of Cadence Design Systems, Inc. All rights reserved worldwide.

CodeWarrior — is a trademark of NXP B.V.

ColdFire — is a trademark of NXP B.V.

ColdFire+ — is a trademark of NXP B.V.

EdgeLock — is a trademark of NXP B.V.

EdgeScale — is a trademark of NXP B.V.

EdgeVerse — is a trademark of NXP B.V.

eIQ — is a trademark of NXP B.V.

FeliCa — is a trademark of Sony Corporation.

Freescale — is a trademark of NXP B.V.

HITAG — is a trademark of NXP B.V.

ICODE and I-CODE — are trademarks of NXP B.V.

Immersiv3D — is a trademark of NXP B.V.

I2C-bus — logo is a trademark of NXP B.V.

Kinetis — is a trademark of NXP B.V.

Layerscape — is a trademark of NXP B.V.

Mantis — is a trademark of NXP B.V.

MIFARE — is a trademark of NXP B.V.

NTAG — is a trademark of NXP B.V.

Processor Expert — is a trademark of NXP B.V.

QorIQ — is a trademark of NXP B.V.

SafeAssure — is a trademark of NXP B.V.

SafeAssure — logo is a trademark of NXP B.V.

Synopsys — Portions Copyright © 2021 Synopsys, Inc. Used with permission. All rights reserved.

Tower — is a trademark of NXP B.V.

UCODE — is a trademark of NXP B.V.

VortiQa — is a trademark of NXP B.V.

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 4 July 2022
Document identifier: WPRSWLIBUG