RT1650 Application Note

Contents

I^{2}C

The RT1650 provides I^{2}C interface to communicate with external host device. Besides OTP firmware programming and MTP setting programming can be approached through the I^{2}C interface, the external host can also communicate with the RT1650 to achieve more flexible applications. For example, the host can read the ADC information via the I^{2}C Interface. The example code please refer to the annex A. Table 1 shows the specification of the I^{2}C. Table 2 shows the RT1650 register definition. In addition, the I^{2}C is used to read the internal status and the power source is from the V$_{RECT}$. If the wireless function disable or in the adapter mode, the I^{2}C can't be accessed.

- I^{2}C Slave
  0100010X (in binary format)
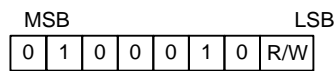 0x44 / 0x45 (hex format, include R/W bit)

MSB							LSB
0	1	0	0	0	1	0	R/W

Table 1. RT1650 I^{2}C specification

Symbol	Description	Min	Typ	Max	Unit
V$_{IL\_I^2C}$	I^{2}C Input logic low			0.6	V
V$_{IH\_I^2C}$	I^{2}C Input logic high	1.2			V
f$_{SCL}$	SCL frequency	10		400	kHz

Table 2. RT1650 register definition

Address	MSB	LSB	Name	Description
0x64	7	0	V$_{RECT}$	V$_{RECT}$ (4V~8V), unit = 15.68mV
0x66	7	0	V$_{OUT}$	V$_{OUT}$ (3V~6V), unit = 11.76mV
0x67	7	0	I$_{OUT}$	I$_{OUT}$ (0A~2A), unit = 7.84mA
0x78	7	0	last CE packet	last CE packet
0x79	7	0	last RP packet	last RP packet
0x7A	7	0	Received Power [7:0] (mW)	low byte of Received Power (mW)
0x7B	6	0	Received Power [14:8] (mW)	high byte of Received Power (mW)
0x7B	7	7	Received Power updating flag	0 : Received Power is valid 1 : Received Power is updating, not valid
0x10	7	7	V$_{OUT}$ enable	0 : V$_{OUT}$ is disable 1 : V$_{OUT}$ is enable
0x02	7	0	freq_cnt [7:0]	Frequency = 1000 / ((freq_cnt[13:0] * 0.11) / 128) kHz
0x03	5	0	freq_cnt [13:8]	
0x7C	3	0	WPC phase status	WPC status 0 : booting 1 : ping phase 2 : ID_CF phase 3 : Negotiation phase 4 : power transfer phase

e.g. : 1. If read the 0x7A data is 0xAA, 0x7B data is 0x21.

The received power is 0x21 * 256 + 0xAA = 8618mW.

2. If read the 0x7A data is 0x55, 0x7B data is 0x91.

This data should be ignore because the data is updating.

RT1650 will update the ADC status of the V_{RECT}, V_{OUT} and I_{OUT} before each CE packet and calculate the received power then updating the register before each RP packet. The time interval of each CE packet is 150ms and each RP packet is 1500ms. The time of the data updating is only few micro seconds. By the way, the RP function is using to detect the FOD for steady state.
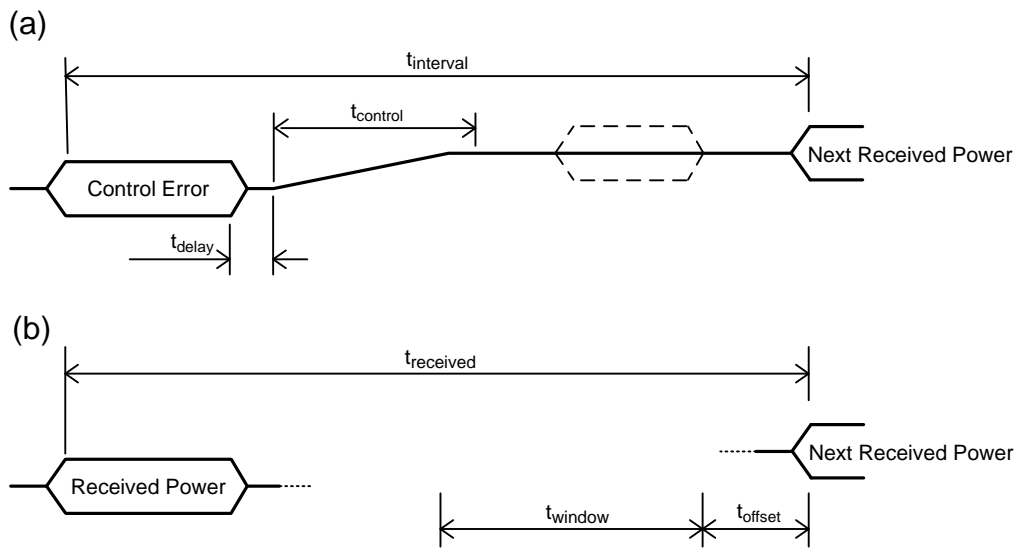

Figure 1. Power Receiver timing in the power transfer phase

Table 3. Power Receiver timing in the power transfer phase

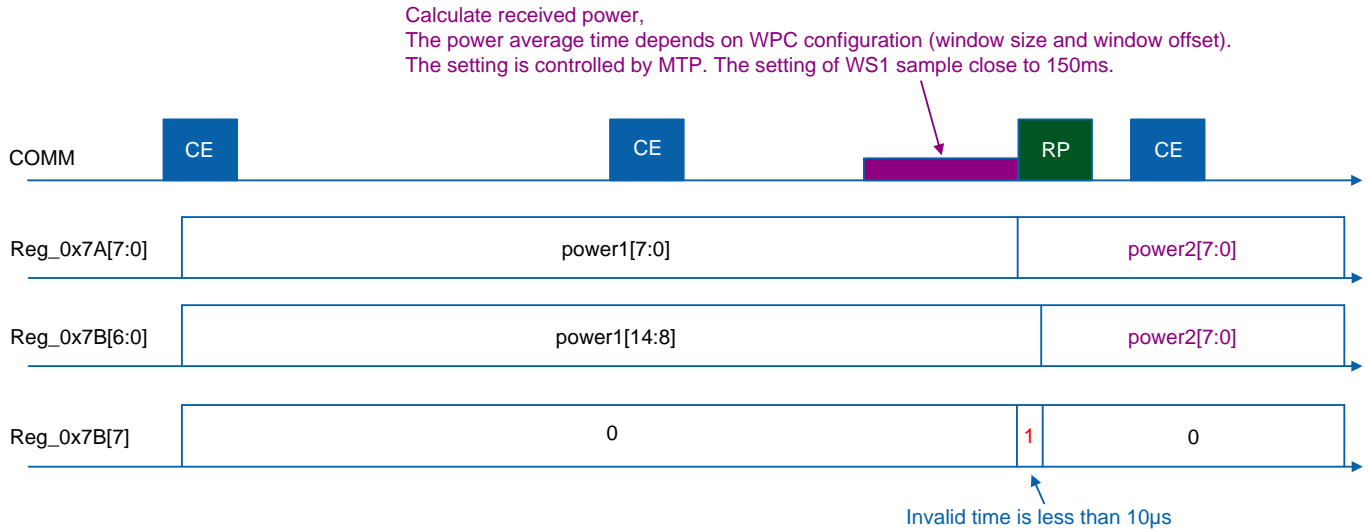| Parameter	Symbol	Minimum	Target	Maximum	Unit
Interval*	$t_{interval}$	—	250	350.0^{+0}	ms
Controller time	$t_{control}$	24.0_{-0}	25	N.A.	ms
Received Power Packet time	$t_{received}$	—	1500	4000.0	ms

Figure 2. Received Power Calculate timing

Power Transfer phases

Figure 3. shows the 4 power transfer phases for the WPC v1.1.

- **SELECTION :** As soon as the Power Transmitter applies a Power Signal, the Power Receiver shall enter the selection phase.
- **PING :** The power Receiver should send the Digital Ping Packet to power Transmitter then into next phase. If not, the system shall revert to the Selection phase. The power Receiver also can send the End Power transfer Packet to stop the power Transmitter.
- **IDENTIFICATION & CONFIGURATIOIN :** In this phase, the Power Receiver identifies the revision of the System Description Wireless Power Transfer the Power Receiver complies and configuration information such as the maximum power that the Power Receiver intends to provide at its output. The Power Transmitter uses this information to create a Power Transfer Contract.
- **POWER TRANSFER :** In this phase, the Power Transmitter continues to provide power to the Power Receiver. The power Receiver sends the Control Error Packet for adjusting the Primary Cell current. The Power Transmitter stops to provide power when the Received Power Packet is too low to trigger the FOD function or End Power Transfer Packet is sent from power Receiver.
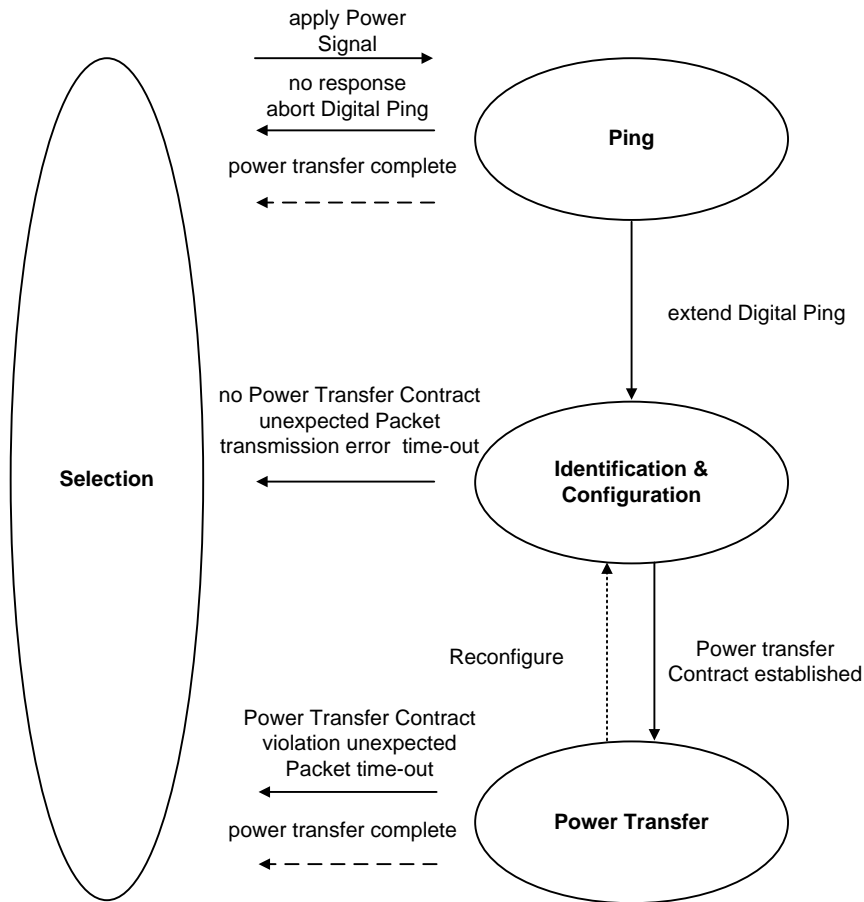
Figure 3. WPC v1.1 Low Power Transfer Phases

Mode Selection

The RT1650 provides 2 input pins for operating mode control. The V_{IH} of the Mode0 and Mode1 is 1.2V (min), V_{IL} is 0.6V (max), shown as Table 4. Table 5 shows an example of operating mode control for wireless power and external adapter power. In default mode, both MODE0 and MODE1 are low, the wireless power is enabled and the adapter power has a higher priority. The wireless power is the normally operation, shown as Figure 4. Once the adapter power is detected, the wireless power will be turned off and the ADEN will be pulled low to turn on the external switch for connecting the adapter power to system load, shown as Figure 5. When the MODE1 is pulled to high, the adapter power will be turned off by the external switch and enters wireless mode to allow wireless power operation only, shown as Figure 6. In adapter mode, the wireless power is turned off always and ADEN is pulled low to turn on external switch for adapter power, shown as Figure 7. In this mode, it allows an external charger operating in USB OTG mode to connect the OUT pin to power the USB at ADD pin, shown as Figure 8. If both MODE0 and MODE1 pins are pulled to high, the wireless power and adapter power are disabled, shown as Figure 9.

Table 4. RT1650 Mode0 and Mode1 specification

Symbol	Description	Min	Typ	Max	Unit
V_{IL_Mode}	Mode Input logic low			0.6	V
V_{IH_Mode}	Mode Input logic high	1.2			V

Table 5. Operation Mode Control

Mode	MODE0	MODE1	Wireless Power	Adapter Power	OTG
Default	0	0	ON	ON[(*)]	OFF
Wireless	0	1	ON	OFF	OFF
Adapter	1	0	OFF	ON	Allowed
Disable	1	1	OFF	OFF	OFF

(*)Note : If both adapter power and wireless power are present, adapter power is given higher priority.
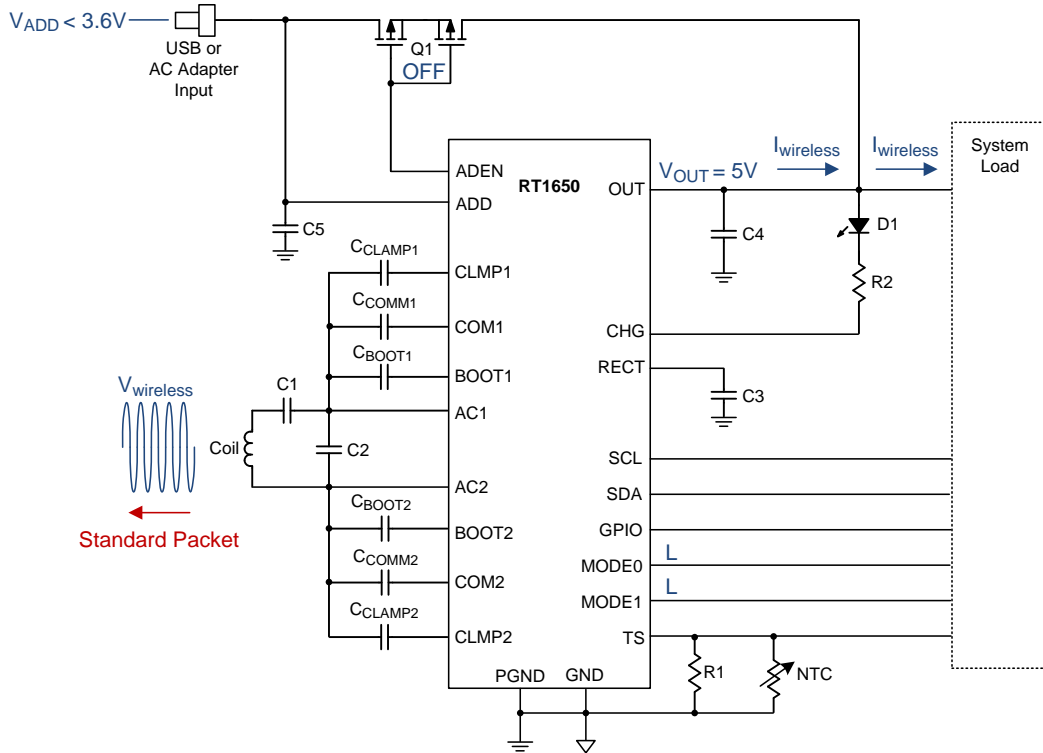

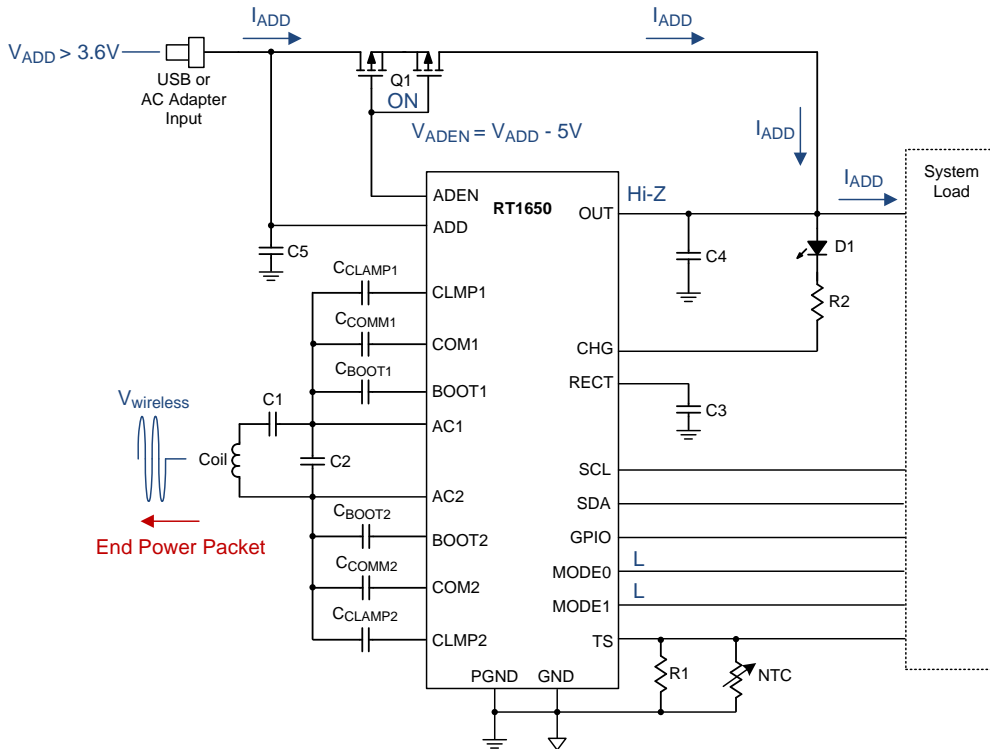Figure 4. Default Mode Wireless Power operation

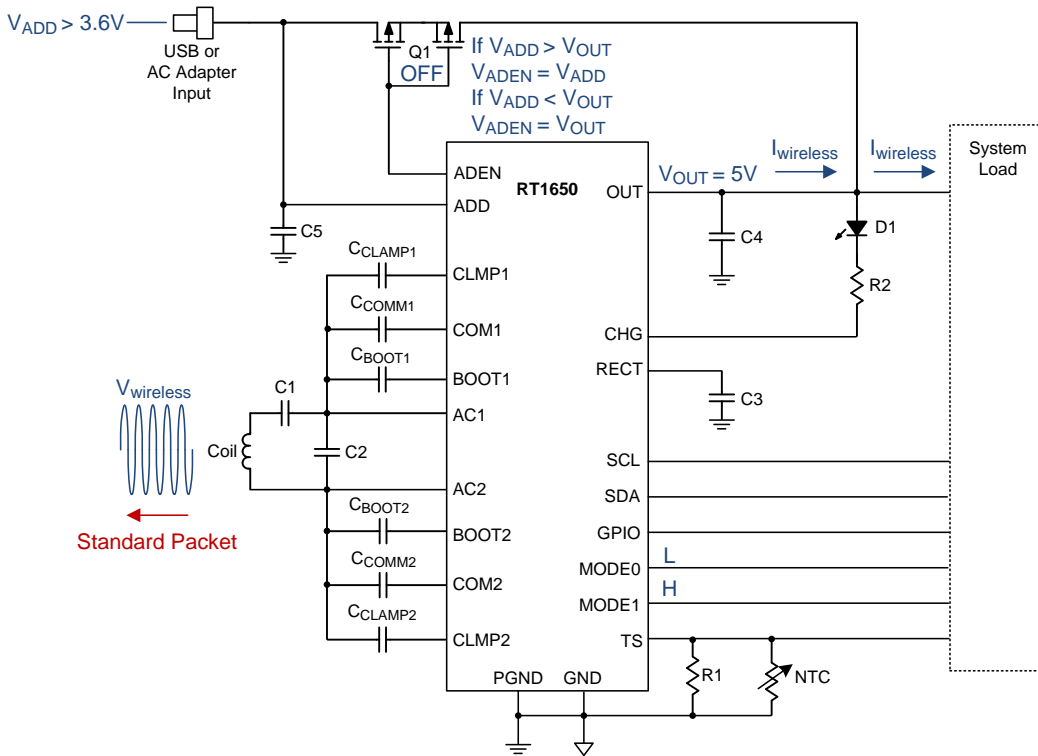Figure 5. Default Mode adapter power priority operation
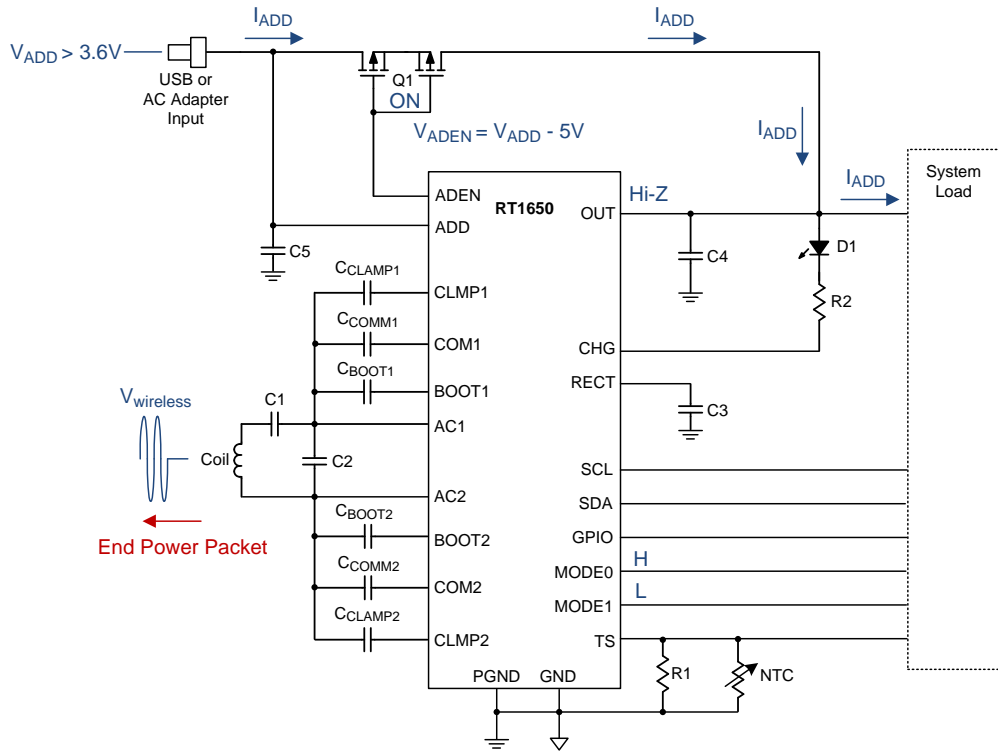

Figure 6. Wireless Power Mode operation
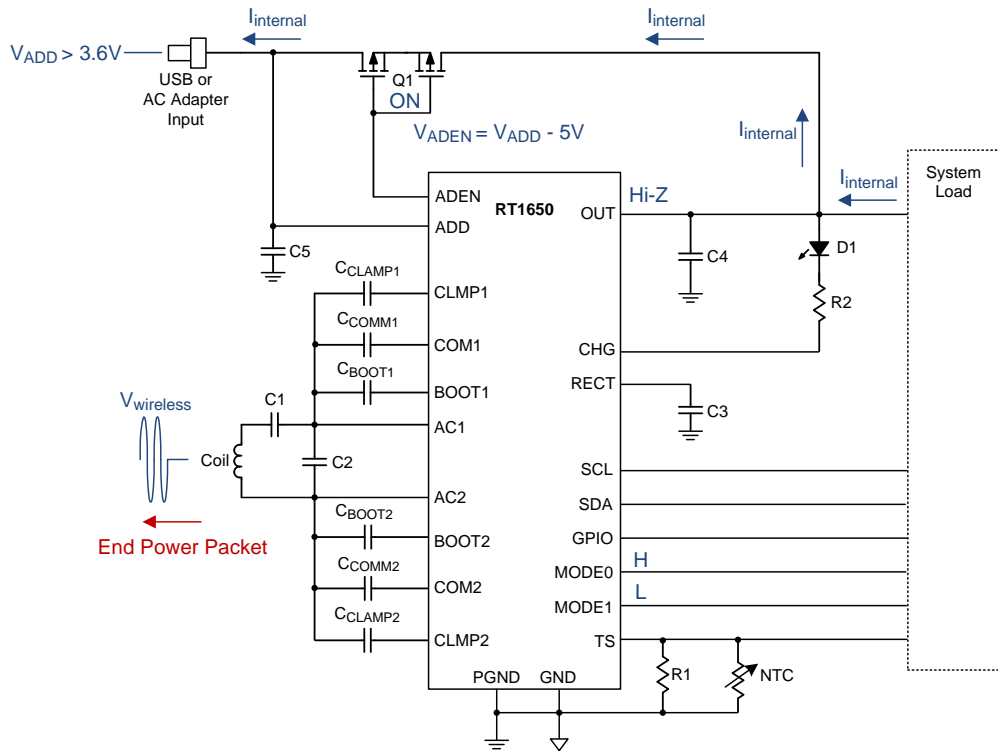
Figure 7. Adapter Mode operation
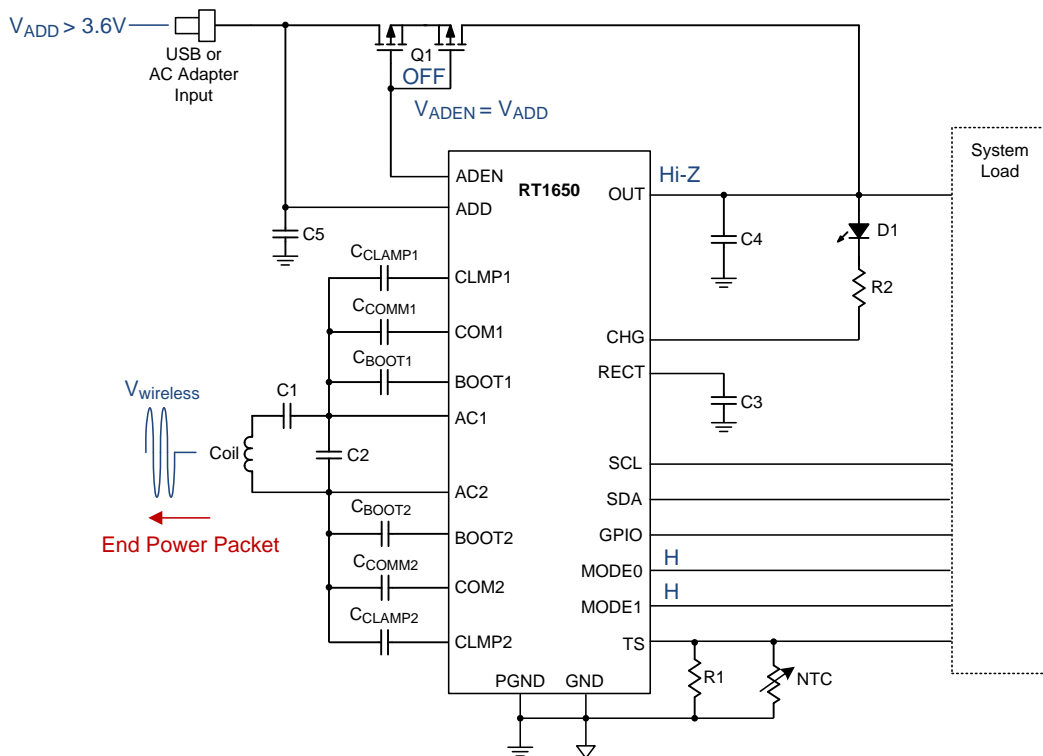

Figure 8. OTG Mode operation

Figure 9. Disable Mode operation

Thermal Management

The RT1650 provides an external device thermal management function with an external NTC thermistor and a resistor connected between TS pin and GND pin shown as Figure 10. User can use this function to control the temperature of the coil, battery or other device. An internal current source (60μA) is provided to the external NTC thermistor and generates a voltage at the TS pin. The TS voltage is detected and sent to the ADC converter for external device thermal manage control.
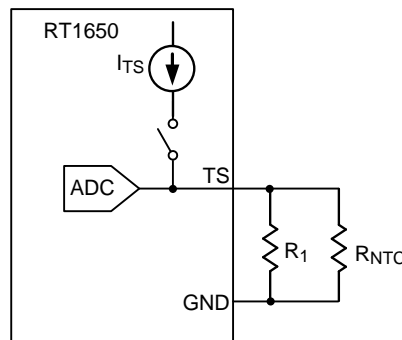

Figure 10. NTC Circuit for Device Temperature Detection and Thermoregulation

The thermal management function is shown as Figure 11. If the temperature is higher than Hot_temp or lower than Cold_temp threshold, the RT1650 will send the EPT to disable the power transfer. When the detected temperature increases and reaches the desired Regulation_temp, RT1650 will decrease the current limit to reduce the output current to regulate the temperature. When the detected temperature is lower than the Regulation_temp, the current limit will increase to the default value. This function is shown as Figure 12.
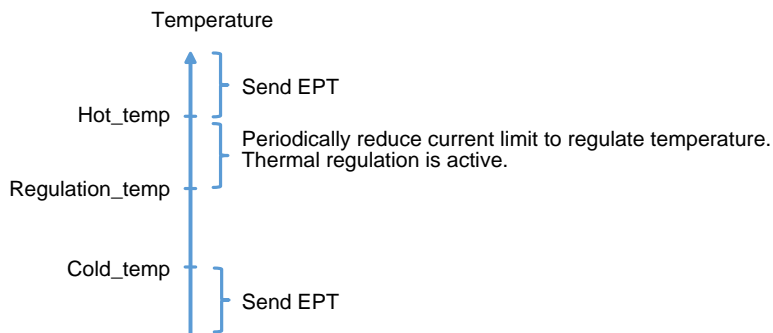
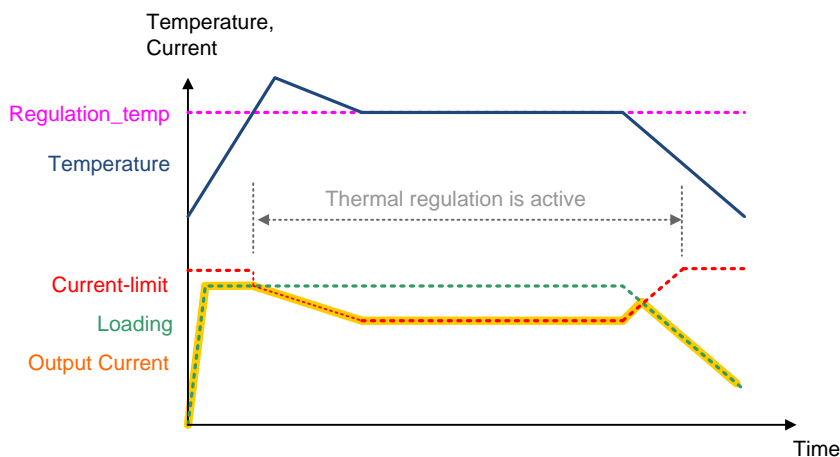Figure 11. Thermal management function

Figure 12. Thermoregulation Control

The thermal management is programmable by MTP of each temperature setting. Figure 13 is the Control Panel of this function. Please refer to the following description for these item.

- Thermal Regulation check box : enable or disable all the thermal management function.
- send EPT when HOT : Send the EPT to Tx when the temperature is higher than Hot_temp.
- send EPT when COLD : Send the EPT to Tx when the temperature is lower than Cold_temp.
- Regulation : Setting for the Regulation_temp (range is 0°C~155°C, step is 1°C)
- Hot : Setting for the Hot_temp (range is 0°C~155°C, step is 1°C)
- Cold : Setting for the Cold_temp (range is -40°C~155°C, step is 1°C)
- Step : The current limit reducing and rising step. The unit is 0.01mA/CE. The CE interval time is 150ms as default. e.g. If the value is 40, step is 40 x 0.01mA/CE = 0.4mA/CE.
- min current limit during : The minimum value of the thermoregulation. This value should be higher than 250mA.
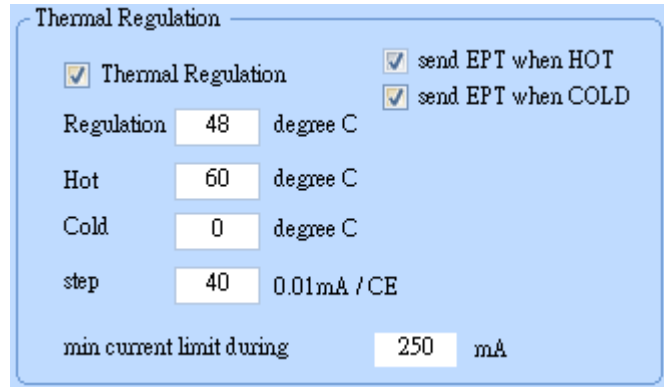
Figure 13. Thermal Regulation Control Panel

The NTC thermistor should be placed as close as possible to the device such as battery or mobile device. The recommended NTC thermistor is NCP15WF104F03RC (tolerance±1%, β = 4250K). The typical resistance of the NTC is 100kΩ at 25degree. The recommended resistance for R_1 is 33kΩ (±1%).

The value of the NTC thermistor at the desired temperature can be estimated by the following equation.

$$R_{NTC\_Reg} = R_O e^{\beta\left(\frac{1}{T_{Reg}} - \frac{1}{T_0}\right)}$$

$$R_{eg} = \frac{R_1 \cdot R_{NTC\_Reg}}{R_1 + R_{NTC\_Reg}}$$

where T_{Reg} is the desired regulation temperature in degree Kelvin. R_O is the nominal resistance at temperature T_0 and β is the temperature coefficient of the NTC thermistor. R_{eg} is the equivalent resistor of NTC thermistor in parallel with R_1.

Figure 14 shows the equivalent resistance of the thermistor in parallel with R_1 resistor varies with operating temperature. Figure 15 shows the VTS voltage with operating temperature. Customer can select the desire temperature and calculate the mapping data by the following equation.

Data = (VTS / 2) * 1024

If the thermal management function is not used (R_{NTC} = open), the resistor R_1 = 24kΩ must be connected between the TS and GND pins.
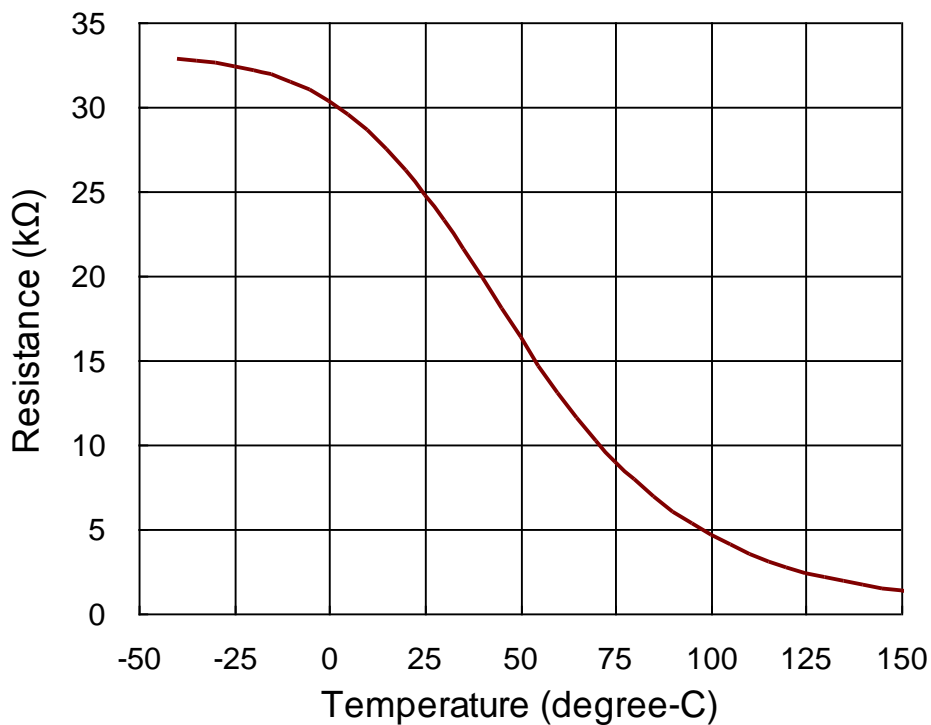
Figure 14. Equivalent Resistance for Temperature Sensing
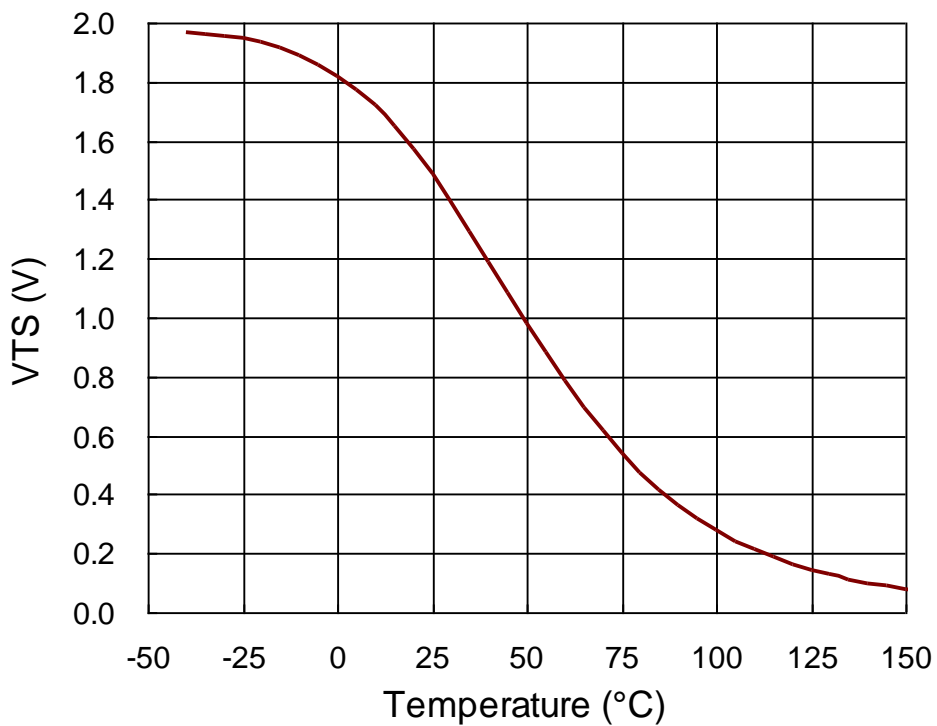

Figure 15. Thermal Sensing Voltage

GPIO

The RT1650 provides a programmable general purpose input/output (GPIO) pin. The GPIO can be used as an input or used as a status indicator for different application. Before use this GPIO, user should discuss its functions with RICHTEK and then RICHTEK code its function into firmware.

- GPIO can be programmed as an output port, be a status indicator. For example,
    - To control LED flashing when Rx position search
    - To indicate thermal regulation is active
    - To indicate battery is full or charging is complete
- GPIO can be programmed as input port, to connect external signal and inform MCU. For example,
    - if GPIO is high, MCU turn on V_{OUT}
    - If GPIO is low, MCU turn off V_{OUT}
- Option for GPIO
    - internal pull-up option (pull-up to 3.3V)
    - Internal pull-low option
    - GPIO can be push-pull or open-drain architecture when GPIO programmed as an output.

Table 6. RT1650 GPIO specification

Symbol	Description	min	typ	max
V_{IL}	input logic low voltage			0.8V
V_{IH}	input logic high voltage	2V		
V_{OL}	output low voltage			0.4V
V_{OH}	output high voltage when push-pull architecture	2.6V	3.3V	
V_{OH}	output high voltage when open-drain architecture		Hi-Z	

Table 7. RT1650 GPIO functions

	Description	H	L
Output	I^{2}C Status	Ready	Not ready
Output	V_{OUT} Status	On	Off
Output	Thermal Regulation Status	Active	Not active
Output	Battery Charge Complete Status	Active	Not active
Input	V_{OUT} Control	On	Off
Input	EPT Control (internal pull-high)	Normal operation	Send EPT

Received Power

The RT1650 is a WPC 1.1.1 compatible device. In order to enable a power transmitter to monitor the power loss across the interface as one of the possible methods to limit the temperature rise of foreign objects, the RT1650 reports its received power to the power transmitter. The received power equals the power that is available from the output of the power receiver plus any power that is lost in producing that output power (the power loss in the secondary coil and series resonant capacitor, the power loss in the shielding of the power receiver, the power loss in the rectifier). In WPC 1.1.1 specification, foreign object detection (FOD) is enforced. This means the RT1650 will send received power information with known accuracy to the transmitter. The received power is sensed as the Figure 16.
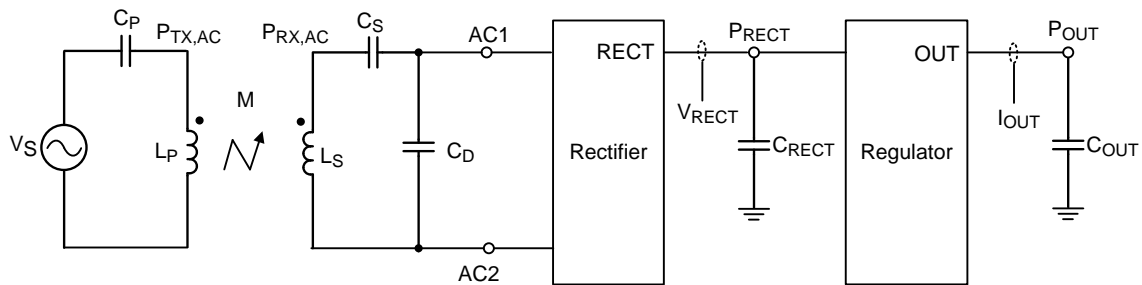

Figure 16. Received Power Sensed

The received power can calculate by the following formula.

$P_{RX,AC} = (V_{RECT} \times I_{OUT}) / Eff_{RECT} + P_{res\_loss} + P_{offset}$

    $P_{RX, AC}$ is the Received Power for RP packet

    V_{RECT} is the output voltage of rectifier from ADC

    I_{OUT} is the output current from ADC

    Eff_{RECT} is the efficiency of rectifier

    P_{offset} is the initial power offset for P_{TX} and P_{RX}

$P_{res\_loss} = k (R_S + R_{ESR}) \times I_{OUT}^2$

    k is a constant coefficient

    R_S is the AC resistance of Rx coil

    R_{ESR} is the AC resistance of series capacitor

$R_S = R_{X100} [1 + A (f / 100 - 1) + B (f / 100 - 1)^2]$

    R_{X100} is the Rx coil resistance at 100kHz

    F is the AC frequency from Tx

    A and B is the resistance matching coefficient

$R_{ESR} = R_{CS100} / (f / 100)$

    R_{CS100} is the capacitor resistance at 100kHz

To use the GUI for the FOD calibration, the customer should measure the resistance of the coil from 100kHz to 200kHz and select the R_{X100}, coefficient A and B to match the resister with frequency. The resister of the capacitor, R_{CS100}, could be measure or get from the datasheet. If the customer can get the WPC certification transmitter, the P_{offset} can be selected for initial calibration. E.g. For this coil, we can get the A = 0.56, B = 0.32, R_{x100} = 362mΩ, shown as Figure 17. R_{CS_100} = 150mΩ. Poffset = 100mW. Then use the GUI to set the parameter to MTP, shown as Figure 19. Please refer to the following description for these item.

- CoeffA : The coefficient A of the coil resistance matching. (range is 0~2.55, step is 0.01)
- CoeffB : The coefficient B of the coil resistance matching. (range is 0~2.55, step is 0.01)
- R_{x100} : The resistance of the coil at 100kHz. (range is 0~510mΩ, step is 2mΩ)
- R_{cs100} : The resistance of the Cs cap at 100kHz. (range is 0~255mΩ, step is 1mΩ)
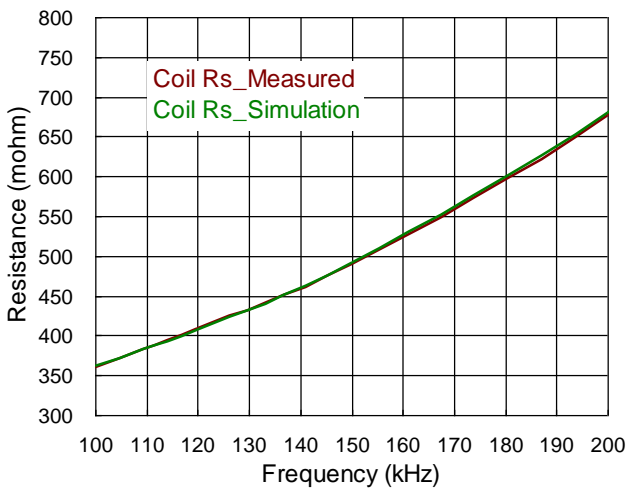- Power_offset : To compensate the power offset of the Tx and Rx. (range is 0~1.27W, step is 0.01W)
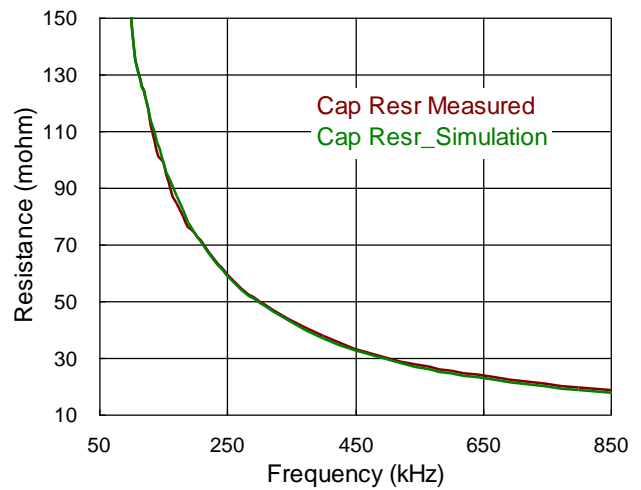

Figure 17. Coil resistance matching

Figure 18. Capacitor resistance matching

Figure 19. Received Power Control Panel

Foreign Object Detection

For the WPC 1.1.1 standard, a Power Receiver shall report its Received Power $P_{received}$ in a Received Power Packet such that $P_{received} - 250mW \leq P_{PR} \leq P_{received}$. This means that the reported Received Power should higher than the actual Received Power P_{PR}, by at most 250mW. RT1650 provide the rectifier efficiency and the resonant tank loss compensating to minimum the power offset between the transmitter and receiver and provide the power offset for FOD function. Figure 20 is the RT1650 FOD tuning flow. For the new model, customer should measure the parameter of the R_{x100}, CoeffA, CoeffB and R_{CS100} then setting to the MTP. First step is that measure the P_{Rx} and $P_{Rxtarget}$ by the FOD test jig, shown as Figure 21. This step can observe the power offset of the initial state and the received power behavior. The second step that we should adjust the power offset to keep the $0 \leq P_{RX}-P_{TX} \leq 250mW$ at no load, shown as Figure 22. The third step is that check the power offset for heavy load and adjust Rx100 to minimize the power difference at heavy load, shown as Figure 23. The fourth step is that check the received power again. If there is any over spec, we can modify the rectifier efficiency to optimize the power, shown as Figure 24.
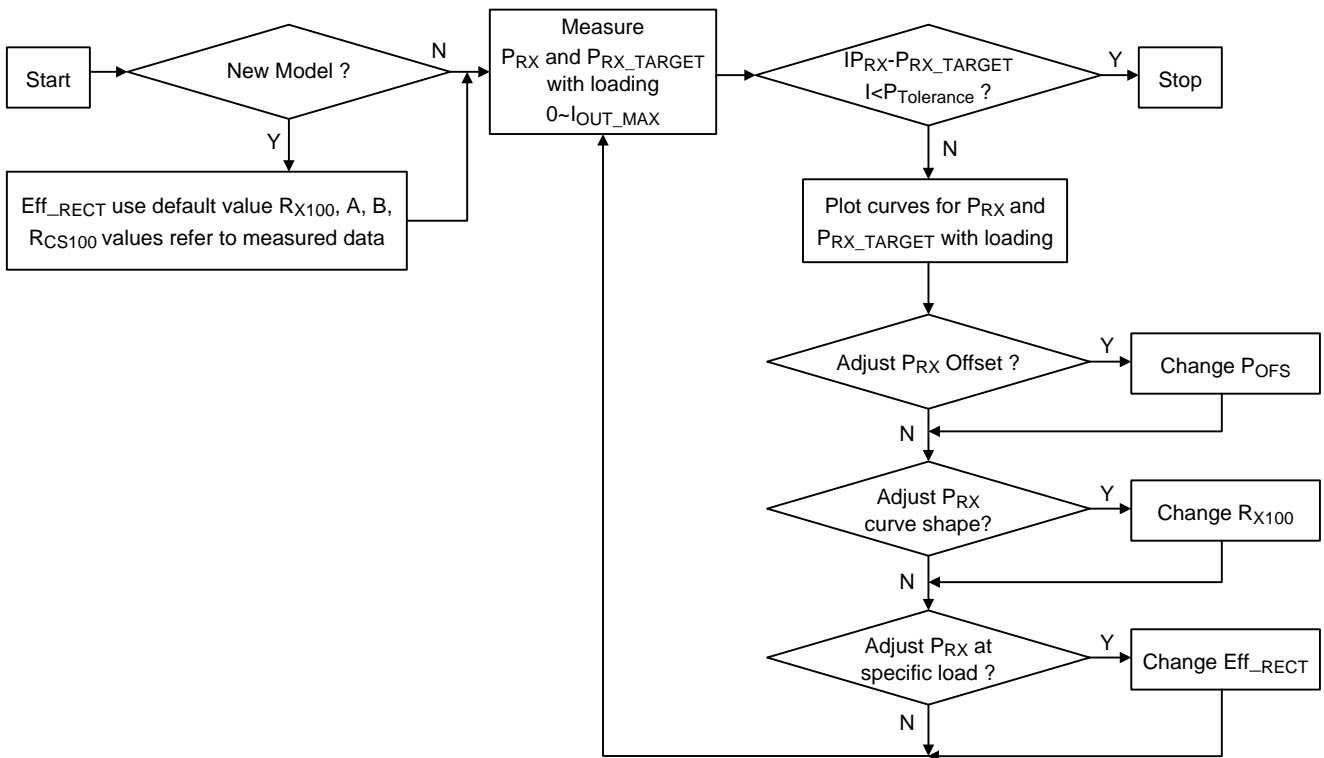

Figure 20. FOD Tuning Flow

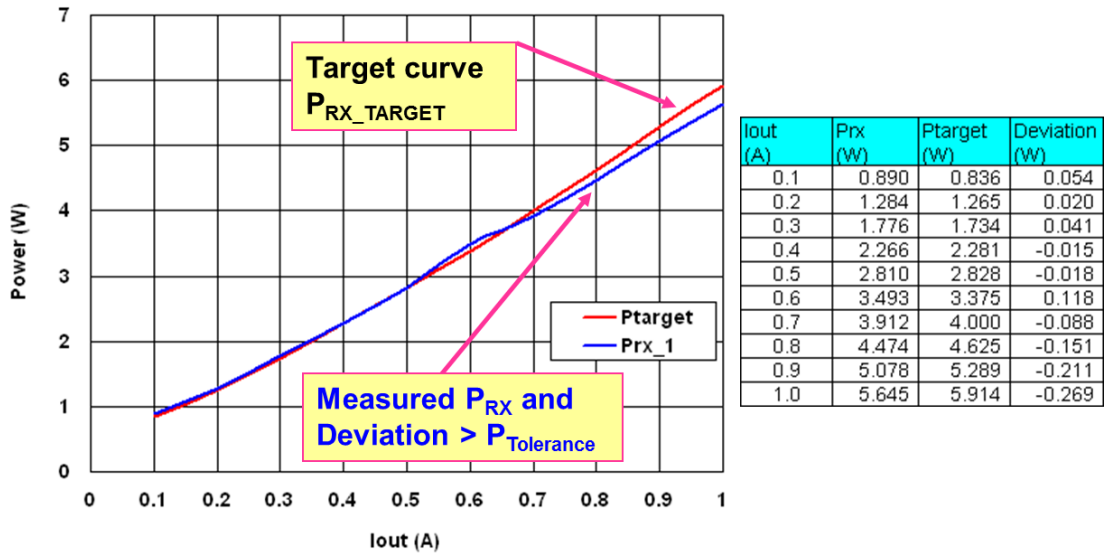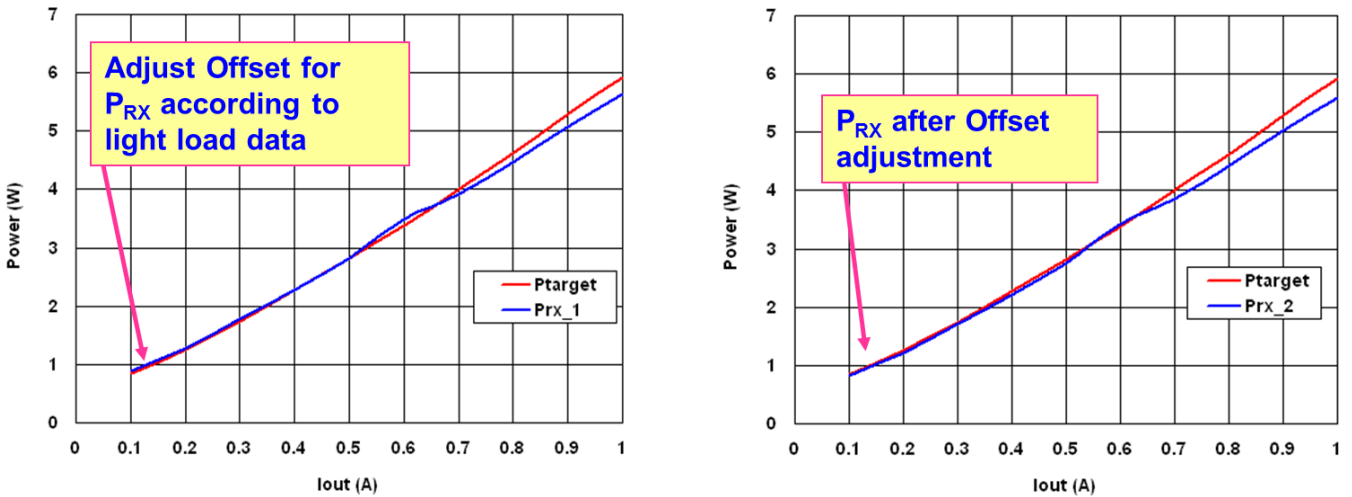| Iout (A)	Prx (W)	Ptarget (W)	Deviation (W)
0.1	0.890	0.836	0.054
0.2	1.284	1.265	0.020
0.3	1.776	1.734	0.041
0.4	2.266	2.281	-0.015
0.5	2.810	2.828	-0.018
0.6	3.493	3.375	0.118
0.7	3.912	4.000	-0.088
0.8	4.474	4.625	-0.151
0.9	5.078	5.289	-0.211
1.0	5.645	5.914	-0.269

Figure 21. The First Step of FOD Tuning Flow
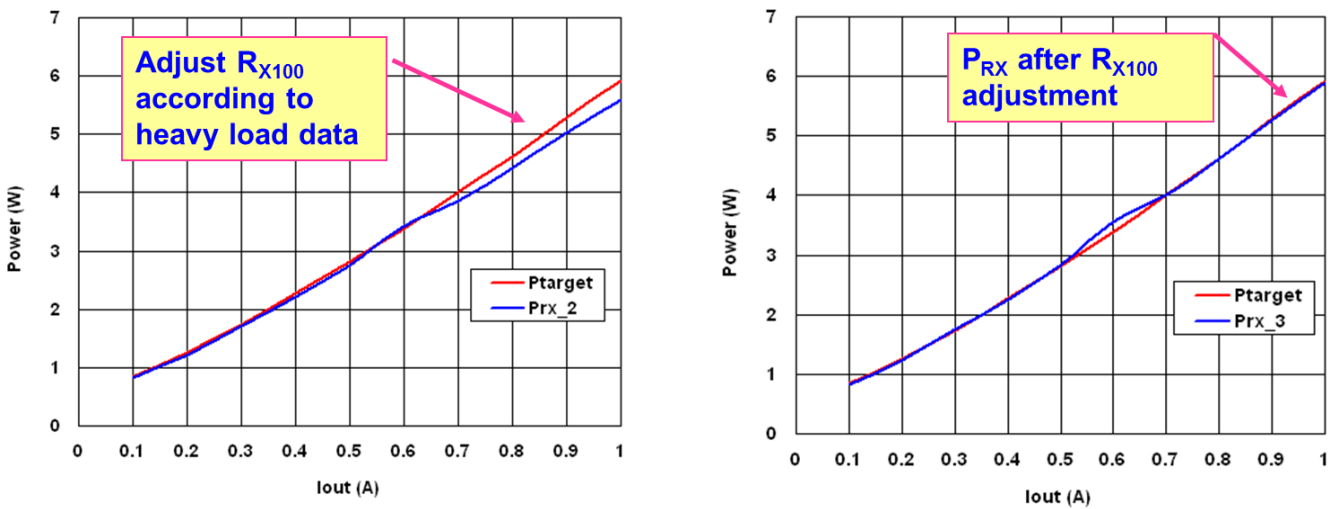

Figure 22. The Second Step of FOD Tuning Flow
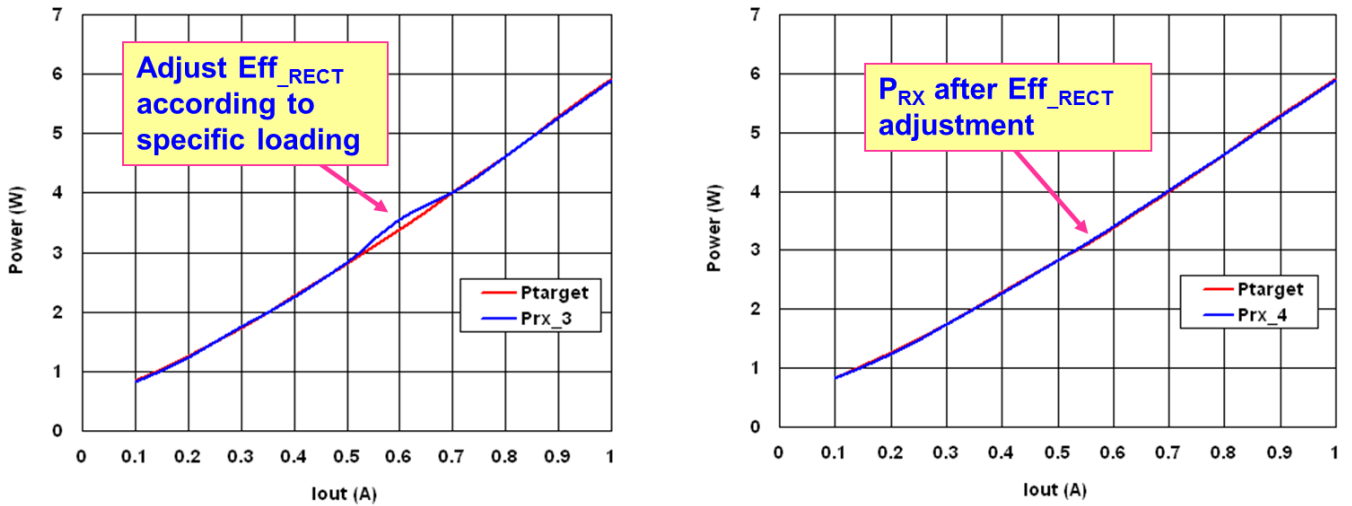

Figure 23. The Third Step of FOD Tuning Flow

Figure 24. The Fourth Step of FOD Tuning Flow

Battery Charge Complete Detection

The RT1650 supports battery charge complete detection function, shown as Figure 25. A programmable charge complete current threshold and a programmable charge complete detect time are provided. This function can be used to send the Charge Status packet to the transmitter for indicating a full charged status 100%. There are 3 operation modes when the charge complete status is detected, shown as Figure 26. Mode1 is to send a CS packet to transmitter only. In the Mode2, the RT1650 will send a CS packet and an EPT packet to transmitter. In the Mode3, the RT1650 will send a CS packet (0x05) then stop communication with the transmitter.
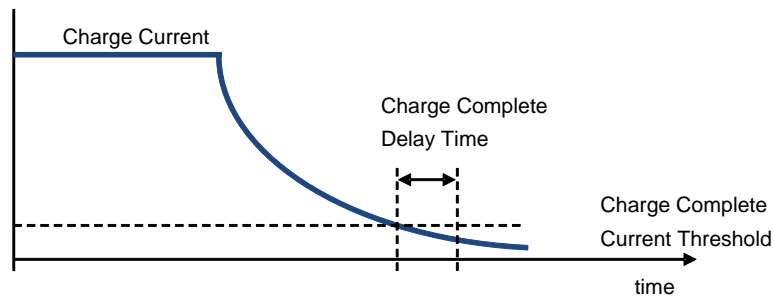

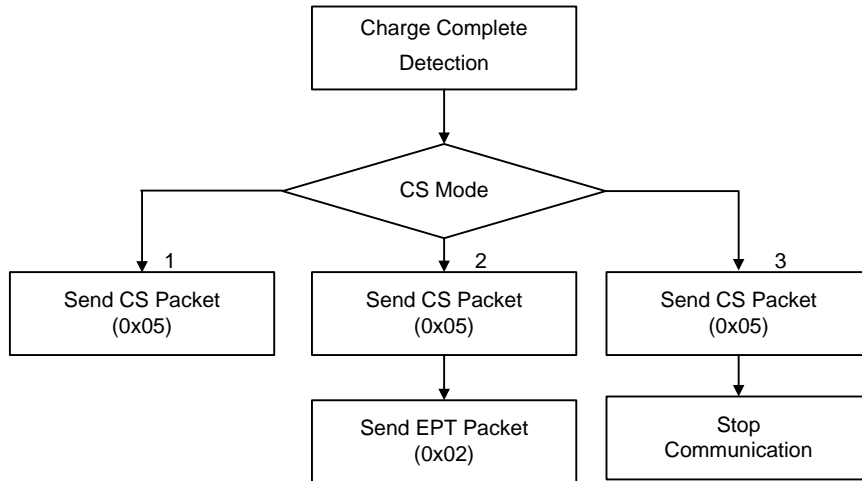Figure 25. The Fourth Step of FOD Tuning Flow

Figure 26. The Fourth Step of FOD Tuning Flow

The Charge Complete is programmable by MTP of each temperature setting. Figure 27 is the Control Panel of this function. Please refer to the following description for these item.

- enable Charge Complete : enable or disable the Battery Charge Complete Detection function.
- Complete mA : The Charge Complete detect threshold current. (range is 0~510mA, step is 2mA)
- Complete sec : The Charge Complete detect threshold current. (range is 0~3825sec, step is 15sec)
- send Charge Status 100 when charge complete : Send the CS packet when Charge Complete. Enable this function for Mode1, Mode2 and Mode3.
- send EPT when charge complete : Send the EPT packet after CS packet. Enable this function for Mode2.
- stop packet after charge complete : Stop the communication after CS packet. Enable this function for Mode3.
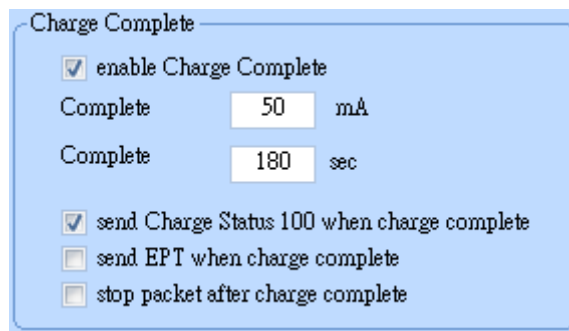

Figure 27. The Fourth Step of FOD Tuning Flow

MTP Program

For the MTP program, please contact to the RICHTEK to get the GUI, test Jig and the driver. The standard program step is as following description.

1. Disable the Tx or remove the coil of the Rx from Tx.
2. Supply 7V and 30mA source ability at least to the RECT pin to GND.
3. Connect the SDA, SCL and GND pins of the test jig "RT Bridgeboard" to PCB. The customer should install the driver "RTBridgeboardUtilitiesV130.exe" first.
4. Open the "RT1650_GUI_tool"
5. Fill in the parameter.
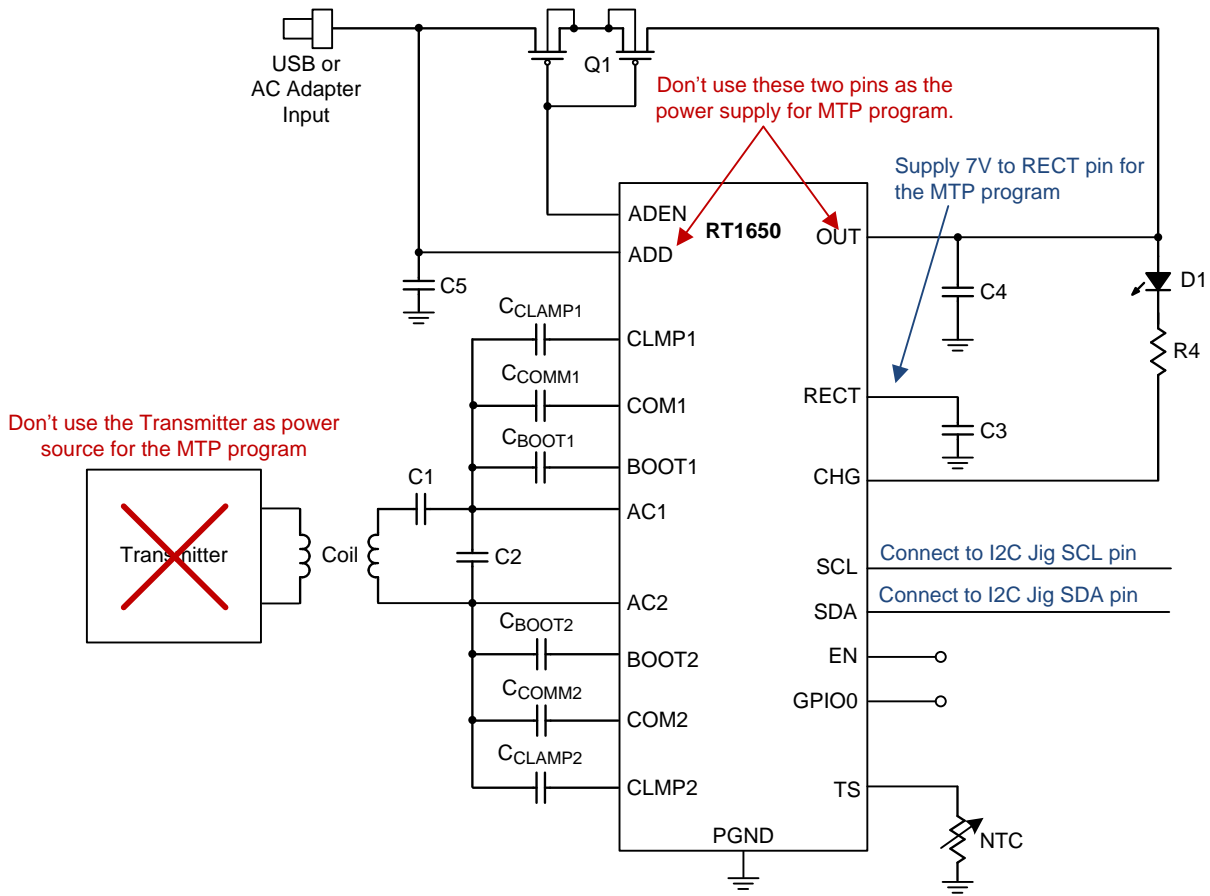

Figure 28. Power source at MTP program

Component Maximum Voltage Rating

The component value and the maximum voltage rating is as following suggestion, shown as Figure 29. These value is selected based on the WPC standard transmitter and 5V adapter application. The customer should be modify by the customer design and application.
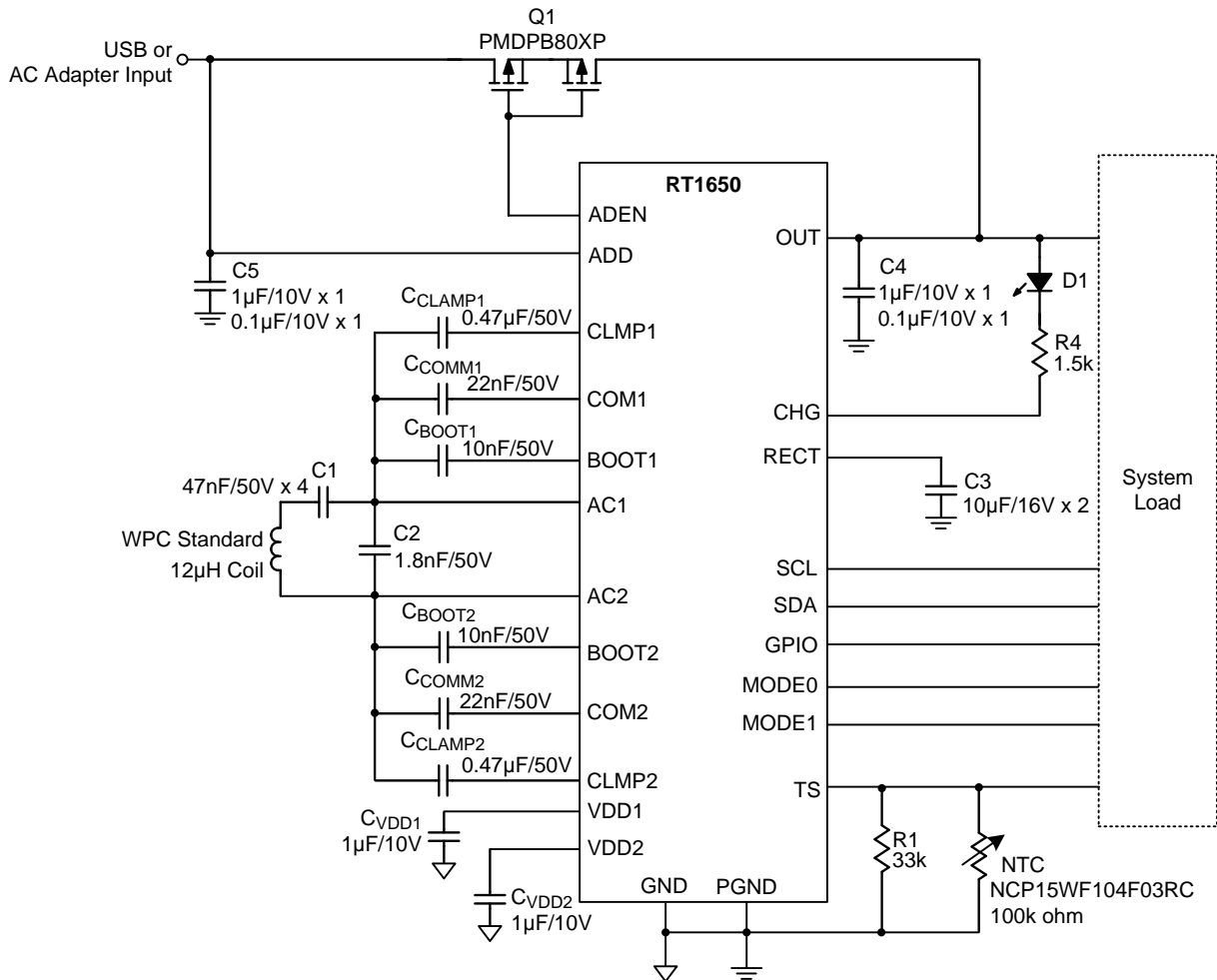

Figure 29. Application Circuit component value

Item	Part Reference	Value	Part Number
1	C1	47nF/50V/0805	GRM21B7U1H473JA01L_MURATA
2	C2	1.8nF/50V/0603/X7R	0603B182K500_WALSIN
3	C_{CLAMP1}, C_{CLAMP2}	0.47µF/50V/0603/X7R	C1608X7R1H474KT_TDK
4	C_{COMM1}, C_{COMM2}	22nF/50V/0603/X7R	0603B223K500_WALSIN
5	C_{BOOT1}, C_{BOOT2}	10nF/50V/0603/X7R	0603B103K500_WALSIN
6	C_{VDD1}, C_{VDD2}	1µF/10V/1005/X5R	C1005X5R1A105K050BB_TDK
7	C3	10µF/16V/0805/X5R	C2012X5R1C106KT_TDK
8	C4	1µF/10V/1005/X5R	C1005X5R1A105K050BB_TDK
		0.1µF/10V/0603/X5R	C0603X5R1A104K030BC_TDK
9	C5	1µF/10V/1005/X5R	C1005X5R1A105K050BB_TDK
		0.1µF/10V/0603/X5R	C0603X5R1A104K030BC_TDK

C1~C4 can use the normal X7R to replace. Part Number : 0603B473K500_WALSIN

Programmable Dynamic Rectifier Voltage Control

The RT1650 provides a programmable Dynamic Rectifier Voltage Control function to optimize the transient response and power efficiency for applications. Figure 30 show an example to summarize how the rectifier behavior is dynamically adjusted based the $V_{RECT\_SET1\sim4}$ and $I_{OUT\_TH1\sim3}$, which are available to be programmed by MTP. The RT1650 has the V_{RECT} tracking function for the higher efficiency application, shown as Figure 31. This function use the I_{OUT} to calculate the minimum drop-out voltage of the LDO to improve the system efficiency. This function also can tracking the rectifier voltage by the V_{OUT} when current limit. To avoid the V_{OUT} be clamped by the V_{RECT} when the current limit released, RT1650 provide the tracking threshold parameter for the tracking function working.
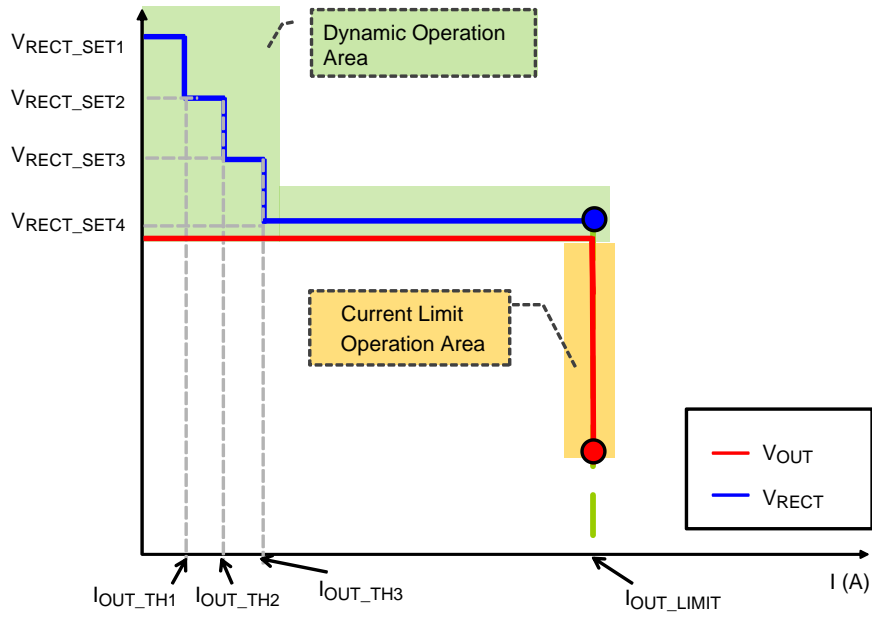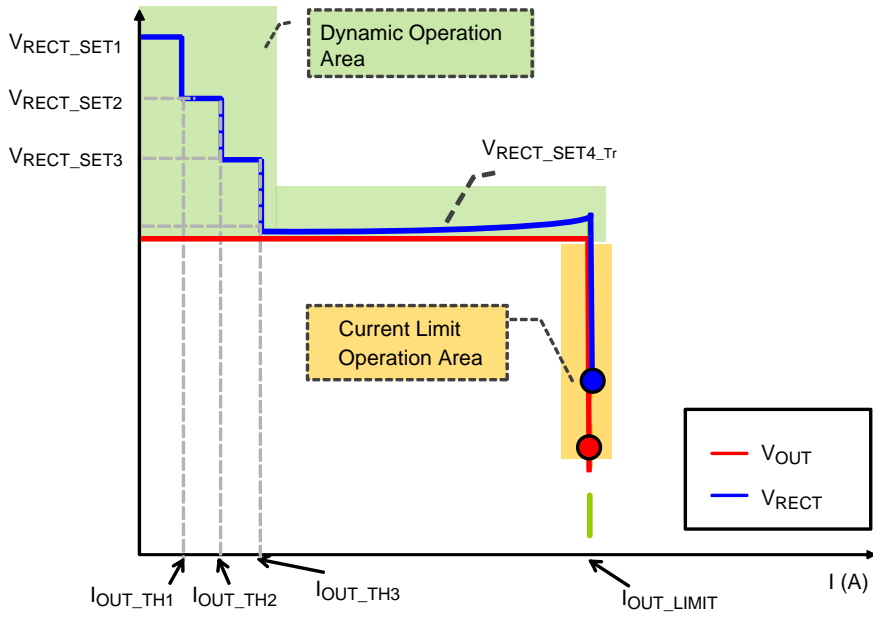
Figure 30. Dynamic Rectifier Voltage Control

Figure 31. V$_{RECT}$ Tracking control

Figure 32. VRECT Tracking control Panel

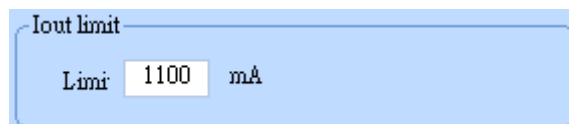The Dynamic Rectifier Voltage Control is programmable by MTP. Figure 32 is the Control Panel of this function. Please refer to the following description for these item.

- V_{RECT_1} : The rectifier voltage target of $I_{OUT} < I_{OUT\_TH1}$. (range is 5V~10V, step is 0.01V)
- V_{RECT_2} : The rectifier voltage target of $I_{OUT\_TH1} < I_{OUT} < I_{OUT\_TH2}$. (range is 5V~10V, step is 0.01V)
- V_{RECT_3} : The rectifier voltage target of $I_{OUT\_TH2} < I_{OUT} < I_{OUT\_TH3}$. (range is 5V~10V, step is 0.01V)
- V_{RECT_4} : The rectifier voltage target of $I_{OUT} > I_{OUT\_TH3}$. (range is 5V~10V, step is 0.01V)
- I_{OUT_TH1} : The threshold for the rectifier voltage change. (range is 0A~1A, step is 0.01A)
- I_{OUT_TH2} : The threshold for the rectifier voltage change. (range is 0A~1A, step is 0.01A)
- I_{OUT_TH3} : The threshold for the rectifier voltage change. (range is 0A~1A, step is 0.01A)
- $I_{OUT\_TH\_HYS}$: The hysteresis of the Dynamic Rectifier Voltage Control. (range is 0A~1A, step is 0.01A)
- enable V_{RECT} Tracking : Enable V_{RECT} Tracking function. Set the $V_{RECT\_4\_TR} = V_{OUT} + I_{OUT} * R + V_{offset}$.
- R : The equivalent resistor of the V_{RECT} Tracking function. (range is 0~1.275Ω, step is 5mΩ)
- V_{offset} : The offset voltage of the V_{RECT} Tracking function. (range is 0~2.55V, step is 0.010V)

Figure 33. Iout limit control Panel

The I_{out} limit Control is programmable by MTP. Figure 33 is the I_{out} limit Control Panel. Please refer to the following description for this item.

- Limit : The Iout limit threshold. (range is 200mA~1800mA, step is 10mA)

V_{OUT} disable for battery system

RT1650 have a detect function for loading is battery to avoid the reset fail. This function detects the T_x power then check the V_{RECT} and V_{OUT} status. If the T_x have no power, RT1650 will close the V_{OUT} and V_{RECT}.

Position Search

RT1650 provide the position search function for customer. This function adjusts the CHG pin frequency to control the LED flicker to let the user can know the best position for coupling. For this function:
1. Enable this function in MTP.
2. A LED and a 10kΩ resistor should be connected to the RECT pin and CHG pin.

CE packet interval

The communication of the WPC is the ASK modulation and the bit encoding scheme. If the check sum data send from Rx is different with the check sum value, Tx will ignore this packet. If the Tx can't receive the complete packet in 1500ms, Tx will time-out and shut down. For the real system, the load may changes in the communication and that may let the data wrong. RT1650 provide the CE interval control function. If the Iout change more than the threshold setting after the communication, RT1650 reduce the packet interval time to avoid the check sum error then time out.

Annex A

```
void CTEST2::OnBnClickedButtonFastInform()
{
    CString s;
    // WPC status
    //---------------------------------------------------------
    int WPC_status = i2c_rd( 0x44, 0x7C) & 0x0F;
    if (WPC_status == 0)
        s.Format( _T("WPC status = booting\n"));
    else if (WPC_status == 1)
        s.Format( _T("WPC status = ping phase\n"));
    else if (WPC_status == 2)
        s.Format( _T("WPC status = identification & configuration phase\n"));
    else if (WPC_status == 3)
        s.Format( _T("WPC status = negotiation phase\n"));
    else if (WPC_status == 4)
        s.Format( _T("WPC status = power transfer phase\n"));
    else
        s.Format( _T("WPC status = un-known\n"));
    //---------------------------------------------------------

    // Vrect
    //---------------------------------------------------------
    double Vrect = 4.0 + (double) i2c_rd( 0x44, 0x64) * (8-4)/255;
    if (Vrect == 8.0)
        s.AppendFormat( _T("Vrect >= 8.0 V\n"));
    else if (Vrect <= 4.0)
        s.AppendFormat( _T("Vrect <= 4.0 V\n"));
    else
        s.AppendFormat( _T("Vrect = %.2f V\n"), Vrect);
    //---------------------------------------------------------

    // Iout
    //---------------------------------------------------------
    double Iout = (double) i2c_rd( 0x44, 0x67) * (2000-0)/255;
    s.AppendFormat( _T("Iout = %.2f mA\n"), Iout);
    //---------------------------------------------------------

    // Vout
    //---------------------------------------------------------
    bool bVoutEn = i2c_rd( 0x44, 0x10) & 0x80;
    if (bVoutEn)
        s.AppendFormat( _T("Vout enable : ") );
    else
        s.AppendFormat( _T("Vout disable : ") );
```

```
double Vout = 3.0 + (double) i2c_rd( 0x44, 0x66) * (6-3)/255;
if (Vout == 6.0)
        s.AppendFormat( _T("Vout >= 6.0 V\n"));
else if (Vout <= 3.0)
        s.AppendFormat( _T("Vout <= 3.0 V\n"));
else
        s.AppendFormat( _T("Vout = %.2f V\n"), Vout);
//---------------------------------------------------------

// CE & RP
//---------------------------------------------------------
int reg_0x78 = i2c_rd( 0x44, 0x78);
int CE = (reg_0x78 & 0x7F) + (reg_0x78 & 0x80)*-1;
        s.AppendFormat( _T("CE packet = %d\n"), CE);
int RP = i2c_rd( 0x44, 0x79);
        s.AppendFormat( _T("RP packet = %d\n"), RP);
//---------------------------------------------------------

// Received Power
//---------------------------------------------------------
int power;
for(int i=0; i<5; i++)
{
        power = (i2c_rd( 0x44, 0x7B) << 8) + i2c_rd( 0x44, 0x7A);
        if (power < 0x7FFF)
                break;
}
s.AppendFormat( _T("Received Power = %d mW\n"), power);
//---------------------------------------------------------

// Frequency
//---------------------------------------------------------
int freq_cnt = ((i2c_rd( 0x44, 0x03) & 0x3F) << 8) + i2c_rd( 0x44, 0x02);
double freq;
if (freq_cnt != 0)
        freq = 1000 / ((freq_cnt * 0.11) / 128); // KHz
else
        freq = 0;
s.AppendFormat( _T("Frequency = %.2f KHz\n"), freq);
//---------------------------------------------------------
}
```

More Information

For marking information, contact our sales representative directly or through a Richtek distributor located in your area, otherwise visit our website for detail.

Next Steps	
Richtek Newsletter	Subscribe Richtek Newsletter

Richtek Technology Corporation

14F, No. 8, Tai Yuen 1st Street, Chupei City

Hsinchu, Taiwan, R.O.C.

Tel: 886-3-5526789