REALTEK

RTL9607C
SINGLE-CHIP PON

GPIO
Application Note
(CONFIDENTIAL: Development Partners Only)

Rev. 1.0.0
26th May 2017

Realtek Semiconductor Corp.
No. 2, Innovation Road II, Hsinchu Science Park, Hsinchu 300, Taiwan
Tel.: +886-3-578-0211 Fax: +886-3-577-6047
www.realtek.com

COPYRIGHT

TRADEMARKS

DISCLAIMER

Realtek provides this document “as is”, without warranty of any kind, neither expressed nor implied, including, but not limited to, the particular purpose. Realtek may make improvements and/or changes in this document or in the product described in this document at any time. This document could include technical inaccuracies or typographical errors.

USING THIS DOCUMENT

Though every effort has been made to assure that this document is current and accurate, more information may have become available subsequent to the production of this guide. In that event, please contact your Realtek representative for additional information that may help in the development process.

CONFIDENTIALITY

This document is confidential and should not be provided to a third-party without the permission of Realtek Semiconductor Corporation.

REVISION HISTORY

Revision	Release Date	Summary
1.0.0	201/05/26	First Release

Table of Contents

1. OVERVIEW 1
2. API 7
2.1. SET GPIO MODE 7
2.2. ENABLE GPIO PIN 7
2.3. GET GPIO INPUT VALUE 8
2.4. GET GPIO OUTPUT VALUE 8
2.5. GET GPIO INTERRUPT STATUS VALUE 8
2.6. CLEAN GPIO INTERRUPT STATUS 9
2.7. SET GPIO INTERRUPT MODE 9
2.8. GET GPIO INTERRUPT MODE 10
2.9. REGISTER ISR HANDLER 10
2.10. UNREGISTER ISR HANDLER 11

List of Tables

TABLE 1. REGISTER SET (BASE: 0XB800_3300)..... 1
TABLE 2. GPIO PORT A, B, C, D DIRECTION REGISTER (PABCD_DIR) (0XB800_3308)..... 1
TABLE 3. GPIO PORT A, B, C, D DATA REGISTER (PABCD_DAT) (0XB800_330C)..... 2
TABLE 4. PORT A, B,C,D INTERRUPT STATUS REGISTER (PABCD_ISR) (0XB800_3310)..... 2
TABLE 5. PORT A, B INTERRUPT MASK REGISTER (PAB_IMR) (0XB800_3314)..... 2
TABLE 6. PORT C, D INTERRUPT MASK REGISTER (PCD_IMR) (0XB800_3318)..... 3
TABLE 7. PORT E, F, G, H DIRECTION REGISTER (PEFGH_DIR) (0XB800_3324)..... 3
TABLE 8. PORT E, F, G, H DATA REGISTER (PEFGH_DAT) (0XB800_3328)..... 3
TABLE 9. PORT E, F, G, H INTERRUPT STATUS REGISTER (0XB800_332C)..... 4
TABLE 10. PORT E, F INTERRUPT MASK REGISTER (PEF_IMR) (0XB800_3330)..... 4
TABLE 11. PORT C, D INTERRUPT MASK REGISTER (PCD_IMR) (0XB800_3334)..... 4
TABLE 12. PORT A, B, C, AND D INTERRUPT ENABLING REGISTER FOR CPU0 (PABCD_C0_IER) (0XB800_3338)..... 5
TABLE 13. PORT A, B, C, AND D INTERRUPT ENABLING REGISTER FOR CPU1 (PABCD_C1_IER) (0XB800_333C)..... 5
TABLE 14. GPIO PORT J, K, M, N DIRECTION REGISTER (PJKMN_DIR) (B800_3348)..... 5
TABLE 15. PORT J, K, M, N DATA REGISTER (PJKMN_DAT) (B800_334C)..... 5
TABLE 16. PORT J, K, M, N INTERRUPT STATUS REGISTER (PJKMN_ISR) (B800_3350)..... 6
TABLE 17. PORT J, K INTERRUPT MODE REG. (PJK_IMR) (B800_3354)..... 6
TABLE 18. PORT M, N INTERRUPT MODE REG. (PMN_IMR) (B800_3358)..... 6

1. Overview

This application note introduces how to control RTL9607C GPIO function. RTL9607C provide GPIO function and there are 66 General Purpose Input/Output (GPIO) pins (pin 0-10, 12-42, 44-69). Each GPIO pin may be configured as an input or output pin. The GPIO Control register may be used to control the signals of GPIO pins. Because the GPIO pins are shared with some I/O pins and no hardware multiplex to maintain the mutual exclusion of utilization, it's required to have a global proper configuration of the GPIO pin utilization. All the GPIO control register are defined as following table.

Table 1. Register Set (Base: 0xB800_3300)

Offset	Name	Description	Field
0x08	PABCD_DIR	Port A, B, C, D direction register.	MISC
0x0C	PABCD_DAT	Port A, B, C,D data register.	MISC
0x10	PABCD_ISR	Port A, B, C,D interrupt status register.	MISC
0x14	PAB_IMR	Port A, B interrupt mask register.	MISC
0x18	PCD_IMR	Port C, D interrupt mask register.	MISC
0x24	PEFGH_DIR	Port E, F, G, H direction register.	MISC
0x28	PEFGH_DAT	Port E, F, G, H data register.	MISC
0x2C	PEFGH_ISR	Port E, F, G, H interrupt status register.	MISC
0x30	PEF_IMR	Port E, F interrupt mask register.	MISC
0x34	PGH_IMR	Port G, H interrupt mask register.	MISC
0x38	PABCD_C0_IER	Port A~D Int. Enabling Reg. for CPU0	MISC
0x3C	PABCD_C1_IER	Port A~D Int. Enabling Reg. for CPU1	MISC
0x48	PJKMN_DIR	Port J, K, M, N direction register.	MISC
0x4C	PJKMN_DAT	Port J, K, M, N data register.	MISC
0x50	PJKMN_ISR	Port J, K, M, N interrupt status register.	MISC
0x54	PJK_IMR	Port J, K, interrupt mask register.	MISC
0x58	PMN_IMR	PortM, N interrupt mask register.	MISC

Table 2. GPIO Port A, B, C, D Direction Register (PABCD_DIR) (0xB800_3308)

Bit	Bit Name	Description	R/W	InitVal
n.31-n.24	DRC_D[7:0]	Pin direction configuration of Port D 0=Configured as input pin 1=Configured as output pin	R/W	00H
n.23-n.16	DRC_C[7:0]	Pin direction configuration of Port C. 0=Configured as input pin 1=Configured as output pin	R/W	00H
n.15-n.8	DRC_B[7:0]	Pin direction configuration of Port B. 0=Configured as input pin 1=Configured as output pin	R/W	00H
n.7-n.0	DRC_A[7:0]	Pin direction configuration of Port A. 0=Configured as input pin 1=Configured as output pin	R/W	00H

Table 3. GPIO Port A, B, C, D Data Register (PABCD_DAT) (0xB800_330C)

Bit	Bit Name	Description	R/W	InitVal
n.31-n.24	PD_D[7:0]	Pin data of Port D. 0 : Data=0 1 : Data=1	R, R/W	00H
n.23-n.16	PD_C[7:0]	Pin data of Port C. 0 : Data=0 1 : Data=1	R, R/W	00H
n.15-n.8	PD_B[7:0]	Pin data of Port B. 0 : Data=0 1 : Data=1	R, R/W	00H
n.7-n.0	PD_A[7:0]	Pin data of Port A. 0 : Data=0 1 : Data=1	R, R/W	00H

Table 4. Port A, B,C,D Interrupt Status Register (PABCD_ISR) (0xB800_3310)

Bit	Bit Name	Description	R/W	InitVal
n.31-n.24	IPS_D[7:0]	Interrupt pending status of port D. Write '1' to clear the interrupt	R/W	00H
n.23-n.16	IPS_C[7:0]	Interrupt pending status of port C. Write '1' to clear the interrupt	R/W	00H
n.15-n.8	IPS_B[7:0]	Interrupt pending status of port B. Write '1' to clear the interrupt	R/W	00H
n.7-n.0	IPS_A[7:0]	Interrupt pending status of port A. Write '1' to clear the interrupt	R/W	00H

Table 5. Port A, B Interrupt Mask Register (PAB_IMR) (0xB800_3314)

Bit	Bit Name	Description	R/W	InitVal
n.31- n.30	PB7_IM[1:0]	PortB.7 interrupt mode.	R/W	00B
n.29- n.28	PB6_IM[1:0]	PortB.6 interrupt mode.	R/W	00B
n.27- n.26	PB5_IM[1:0]	PortB.5 interrupt mode.	R/W	00B
n.25- n.24	PB4_IM[1:0]	PortB.4 interrupt mode.	R/W	00B
n.23- n.22	PB3_IM[1:0]	PortB.3 interrupt mode.	R/W	00B
n.21- n.20	PB2_IM[1:0]	PortB.2 interrupt mode.	R/W	00B
n.19- n.18	PB1_IM[1:0]	PortB.1 interrupt mode.	R/W	00B
n.17- n.16	PB0_IM[1:0]	PortB.0 interrupt mode.	R/W	00B
n.15- n.14	PA7_IM[1:0]	PortA.7 interrupt mode.	R/W	00B
n.13-n.12	PA6_IM[1:0]	PortA.6 interrupt mode.	R/W	00B
n.11- n.10	PA5_IM[1:0]	PortA.5 interrupt mode.	R/W	00B
n.9-n.8	PA4_IM[1:0]	PortA.4 interrupt mode.	R/W	00B
n.7-n.6	PA3_IM[1:0]	PortA.3 interrupt mode.	R/W	00B
n.5-n.4	PA2_IM[1:0]	PortA.2 interrupt mode.	R/W	00B
n.3-n.2	PA1_IM[1:0]	PortA.1 interrupt mode.	R/W	00B
n.1-n.0	PA0_IM[1:0]	PortA.0 interrupt mode. 00=Disable interrupt 01=Enable falling edge interrupt 10=Enable rising edge interrupt 11=Enable both falling or rising edge interrupt	R/W	00B

Table 6. Port C, D Interrupt Mask Register (PCD_IMR) (0xB800_3318)

Bit	Bit Name	Description	R/W	InitVal
n.31- n.30	PD7_IM[1:0]	PortD.7 interrupt mode.	R/W	00B
n.29- n.28	PD6_IM[1:0]	PortD.6 interrupt mode.	R/W	00B
n.27- n.26	PD5_IM[1:0]	PortD.5 interrupt mode.	R/W	00B
n.25- n.24	PD4_IM[1:0]	PortD.4 interrupt mode.	R/W	00B
n.23- n.22	PD3_IM[1:0]	PortD.3 interrupt mode.	R/W	00B
n.21- n.20	PD2_IM[1:0]	PortD.2 interrupt mode.	R/W	00B
n.19- n.18	PD1_IM[1:0]	PortD.1 interrupt mode.	R/W	00B
n.17- n.16	PD0_IM[1:0]	PortC.0 interrupt mode.	R/W	00B
n.15- n.14	PC7_IM[1:0]	PortC.7 interrupt mode.	R/W	00B
n.13-n.12	PC6_IM[1:0]	PortC.6 interrupt mode.	R/W	00B
n.11- n.10	PC5_IM[1:0]	PortC.5 interrupt mode.	R/W	00B
n.9-n.8	PC4_IM[1:0]	PortC.4 interrupt mode.	R/W	00B
n.7-n.6	PC3_IM[1:0]	PortC.3 interrupt mode.	R/W	00B
n.5-n.4	PC2_IM[1:0]	PortC.2 interrupt mode.	R/W	00B
n.3-n.2	PC1_IM[1:0]	PortC.1 interrupt mode.	R/W	00B
n.1-n.0	PC0_IM[1:0]	PortC.0 interrupt mode. 00=Disable interrupt 01=Enable falling edge interrupt 10=Enable rising edge interrupt 11=Enable both falling or rising edge interrupt	R/W	00B

Table 7. Port E, F, G, H Direction Register (PEFGH_DIR) (0xB800_3324)

Bit	Bit Name	Description	R/W	InitVal
n.31-n.24	DRC_H[7:0]	Pin direction configuration of Port H. 0=Configured as input pin 1=Configured as output pin	R/W	00H
n.23-n.16	DRC_G[7:0]	Pin direction configuration of Port G. 0=Configured as input pin 1=Configured as output pin	R/W	00H
n.15-n.8	DRC_F[7:0]	Pin direction configuration of Port F. 0=Configured as input pin 1=Configured as output pin	R/W	00H
n.7-n.0	DRC_E[7:0]	Pin direction configuration of Port E. 0=Configured as input pin 1=Configured as output pin	R/W	00H

Table 8. Port E, F, G, H Data Register (PEFGH_DAT) (0xB800_3328)

Bit	Bit Name	Description	R/W	InitVal
n.31-n.24	PD_H[7:0]	Pin data of Port H. 0 : Data=0 1 : Data=1	R, R/W	00H
n.23-n.16	PD_G[7:0]	Pin data of Port G. 0 : Data=0 1 : Data=1	R, R/W	00H
n.15-n.8	PD_F[7:0]	Pin data of Port F. 0 : Data=0 1 : Data=1	R, R/W	00H

n.7-n.0	PD_E[7:0]	Pin data of Port E. 0 : Data=0 1 : Data=1	R, R/W	00H

Table 9. Port E, F, G, H interrupt status register (0xB800_332C)

Bit	Bit Name	Description	R/W	InitVal
n.31-n.24	IPS_H[7:0]	Interrupt pending status of port H. Write '1' to clear the interrupt	R/W	00H
n.23-n.16	IPS_G[7:0]	Interrupt pending status of port G. Write '1' to clear the interrupt	R/W	00H
n.15-n.8	IPS_F[7:0]	Interrupt pending status of port F. Write '1' to clear the interrupt	R/W	00H
n.7-n.0	IPS_E[7:0]	Interrupt pending status of port E. Write '1' to clear the interrupt	R/W	00H

Table 10. Port E, F Interrupt Mask Register (PEF_IMR) (0xB800_3330)

Bit	Bit Name	Description	R/W	InitVal
n.31- n.30	PF7_IM[1:0]	PortF.7 interrupt mode.	R/W	00B
n.29- n.28	PF6_IM[1:0]	PortF.6 interrupt mode.	R/W	00B
n.27- n.26	PF5_IM[1:0]	PortF.5 interrupt mode.	R/W	00B
n.25- n.24	PF4_IM[1:0]	PortF.4 interrupt mode.	R/W	00B
n.23- n.22	PF3_IM[1:0]	PortF.3 interrupt mode.	R/W	00B
n.21- n.20	PF2_IM[1:0]	PortF.2 interrupt mode.	R/W	00B
n.19- n.18	PF1_IM[1:0]	PortF.1 interrupt mode.	R/W	00B
n.17- n.16	PF0_IM[1:0]	PortF.0 interrupt mode.	R/W	00B
n.15- n.14	PE7_IM[1:0]	PortE.7 interrupt mode.	R/W	00B
n.13-n.12	PE6_IM[1:0]	PortE.6 interrupt mode.	R/W	00B
n.11- n.10	PE5_IM[1:0]	PortE.5 interrupt mode.	R/W	00B
n.9-n.8	PE4_IM[1:0]	PortE.4 interrupt mode.	R/W	00B
n.7-n.6	PE3_IM[1:0]	PortE.3 interrupt mode.	R/W	00B
n.5-n.4	PE2_IM[1:0]	PortE.2 interrupt mode.	R/W	00B
n.3-n.2	PE1_IM[1:0]	PortE.1 interrupt mode.	R/W	00B
n.1-n.0	PE0_IM[1:0]	PortE.0 interrupt mode. 00=Disable interrupt 01=Enable falling edge interrupt 10=Enable rising edge interrupt 11=Enable both falling or rising edge interrupt	R/W	00B

Table 11. Port C, D interrupt Mask Register (PCD_IMR) (0xB800_3334)

Bit	Bit Name	Description	R/W	InitVal
n.31- n.30	PH7_IM[1:0]	PortH.7 interrupt mode.	R/W	00B
n.29- n.28	PH6_IM[1:0]	PortH.6 interrupt mode.	R/W	00B
n.27- n.26	PH5_IM[1:0]	PortH.5 interrupt mode.	R/W	00B
n.25- n.24	PH4_IM[1:0]	PortH.4 interrupt mode.	R/W	00B
n.23- n.22	PH3_IM[1:0]	PortH.3 interrupt mode.	R/W	00B
n.21- n.20	PH2_IM[1:0]	PortH.2 interrupt mode.	R/W	00B
n.19- n.18	PH1_IM[1:0]	PortH.1 interrupt mode.	R/W	00B
n.17- n.16	PH0_IM[1:0]	PortH.0 interrupt mode.	R/W	00B
n.15- n.14	PG7_IM[1:0]	PortG.7 interrupt mode.	R/W	00B
n.13-n.12	PG6_IM[1:0]	PortG.6 interrupt mode.	R/W	00B

Bit	Bit Name	Description	R/W	InitVal
n.11- n.10	PG5_IM[1:0]	PortG.5 interrupt mode.	R/W	00B
n.9-n.8	PG4_IM[1:0]	PortG.4 interrupt mode.	R/W	00B
n.7-n.6	PG3_IM[1:0]	PortG.3 interrupt mode.	R/W	00B
n.5-n.4	PG2_IM[1:0]	PortG.2 interrupt mode.	R/W	00B
n.3-n.2	PG1_IM[1:0]	PortG.1 interrupt mode.	R/W	00B
n.1-n.0	PG0_IM[1:0]	PortG.0 interrupt mode. 00=Disable interrupt 01=Enable falling edge interrupt 10=Enable rising edge interrupt 11=Enable both falling or rising edge interrupt	R/W	00B

Table 12. Port A, B, C, and D Interrupt Enabling Register for CPU0 (PABCD_C0_IER) (0xB800_3338)

Bit	Bit Name	Description	R/W	InitVal
n.31-n.24	PD_IE	Port D (GPIO#31 ~ GPIO#24) interrupt enabling bits. 0: Disable 1: Enable	R/W	00B
n.23-n.16	PC_IE	Port C (GPIO#23 ~ GPIO#16) interrupt enabling bits. 0: Disable 1: Enable	R/W	00B
n.15-n.8	PB_IE	Port B (GPIO#15 ~ GPIO#8) interrupt enabling bits. 0: Disable 1: Enable	R/W	00B
n.7-n.0	PA_IE	Port A (GPIO#7 ~ GPIO#0) interrupt enabling bits. 0: Disable 1: Enable	R/W	00B

Table 13. Port A, B, C, and D Interrupt Enabling Register for CPU1 (PABCD_C1_IER) (0xB800_333C)

Bit	Bit Name	Description	R/W	InitVal
n.31-n.24	PD_IE	Similar to PABCD_C0_IER.PD_IE, except for CPU 1.	R/W	00B
n.23-n.16	PC_IE	Similar to PABCD_C0_IER.PC_IE, except for CPU 1.	R/W	00B
n.15-n.8	PB_IE	Similar to PABCD_C0_IER.PB_IE, except for CPU 1.	R/W	00B
n.7-n.0	PA_IE	Similar to PABCD_C0_IER.PA_IE, except for CPU 1.	R/W	00B

Table 14. GPIO Port J, K, M, N Direction Register (PJKMN_DIR) (B800_3348)

Bit	Bit Name	Description	R/W	InitVal
n.31-n.24	DRC_N	Pin direction configuration of Port N. 0=Configured as input pin 1=Configured as output pin	R/W	00H
n.23-n.16	DRC_M	Similar to DRC_N, except for port M.	R/W	00H
n.15-n.8	DRC_K	Similar to DRC_N, except for port K.	R/W	00H
n.7-n.0	DRC_J	Similar to DRC_N, except for port J.	R/W	00H

Table 15. Port J, K, M, N Data Register (PJKMN_DAT) (B800_334C)

Bit	Bit Name	Description	R/W	InitVal
n.31-n.24	PD_N	Pin data of Port N. 0 : Data=0 1 : Data=1	R, R/W	00H

n.23-n.16	PD_M	Similar to PD_N, except for port M.	R, R/W	00H
n.15-n.8	PD_K	Similar to PD_N, except for port K.	R, R/W	00H
n.7-n.0	PD_J	Similar to PD_N, except for port J.	R, R/W	00H

Table 16. Port J, K, M, N Interrupt Status Register (PJKMN_ISR) (B800_3350)

Bit	Bit Name	Description	R/W	InitVal
n.31-n.24	IPS_N	Interrupt pending status of port N. Write '1' to clear the interrupt	R/W	00H
n.23-n.16	IPS_M	Similar to IPS_N, except for port M.	R/W	00H
n.15-n.8	IPS_K	Similar to IPS_N, except for port K.	R/W	00H
n.7-n.0	IPS_J	Similar to IPS_N, except for port J.	R/W	00H

Table 17. Port J, K Interrupt Mode Reg. (PJK_IMR) (B800_3354)

Bit	Bit Name	Description	R/W	InitVal
n.31- n.30	PK7_IM	PortK.7 interrupt mode: 00=Disable interrupt 01=Enable falling edge interrupt 10=Enable rising edge interrupt 11=Enable both falling or rising edge interrupt	R/W	00B
n.29- n.28	PK6_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.27- n.26	PK5_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.25- n.24	PK4_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.23- n.22	PK3_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.21- n.20	PK2_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.19- n.18	PK1_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.17- n.16	PK0_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.15- n.14	PJ7_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.13-n.12	PJ6_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.11- n.10	PJ5_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.9-n.8	PJ4_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.7-n.6	PJ3_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.5-n.4	PJ2_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.3-n.2	PJ1_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.1-n.0	PJ0_IM	Similar to PK7_IM, except for a different pin.	R/W	00B

Table 18. Port M, N Interrupt Mode Reg. (PMN_IMR) (B800_3358)

Bit	Bit Name	Description	R/W	InitVal
n.31- n.30	PN7_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.29- n.28	PN6_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.27- n.26	PN5_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.25- n.24	PN4_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.23- n.22	PN3_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.21- n.20	PN2_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.19- n.18	PN1_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.17- n.16	PN0_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.15- n.14	PM7_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.13-n.12	PM6_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.11- n.10	PM5_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.9-n.8	PM4_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.7-n.6	PM3_IM	Similar to PK7_IM, except for a different pin.	R/W	00B

Bit	Bit Name	Description	R/W	InitVal
n.5-n.4	PM2_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.3-n.2	PM1_IM	Similar to PK7_IM, except for a different pin.	R/W	00B
n.1-n.0	PM0_IM	Similar to PK7_IM, except for a different pin.	R/W	00B

2. API

Realtek API provides a series of interface to let users setup the GPIO without writing register and table directly. This section will discuss these APIs.

2.1. Set GPIO Mode

The ***rtk_gpio_mode_set*** API will set GPIO mode. The GPIO mode would be GPIO_INPUT/ GPIO_OUTPUT.

Example:

```
/*
   Set GPIO10 to output mode
*/
int32 ret ;
if((ret= rtk_gpio_mode_set (10,GPIO_OUTPUT)) != RT_ERR_OK)
{
    return ret;
}
```

2.2. Enable GPIO PIN

The ***rtk_gpio_state_set*** API would enable GPIO function for given GPIO id.

Example:

```
/* Enable GPIO10 */
int32 ret;

if((ret= rtk_gpio_state_set (10,ENABLED)) != RT_ERR_OK)
    return ret;
```

2.3. Get GPIO Input Value

The ***rtk_gpio_databit_get*** API would get the gpio pin input value.

Example:

```
/*
  get input value from GPIO pin 10
*/

unit32 data;
int32 ret;

if((ret= rtk_gpio_databit_get (10, &data)) != RT_ERR_OK)
    return ret;
```

2.4. Get GPIO Output Value

The ***rtk_gpio_databit_set*** API would set the gpio pin output value.

Example:

```
/*
  Set GPIO pin 10 output value to 1
*/

unit32 data;
int32 ret;

Data =1;
if((ret= rtk_gpio_databit_set (10, data)) != RT_ERR_OK)
    return ret;
```

2.5. Get GPIO Interrupt Status Value

The ***rtk_gpio_intrStatus_get*** API would get the GPIO pin interrupt status value.

Example:

```
/*
```

```
  Get GPIO pin 10 interrupt status value
*/

rtk_enable_t state;
int32 ret;

if((ret= rtk_gpio_intrStatus_get (10, &state)) != RT_ERR_OK)
    return ret;
```

2.6. Clean GPIO Interrupt Status

The ***rtk_gpio_intrStatus_clean*** API would clean gpio interrupt status.

Example:

```
/*
  Clean GPIO pin 10 interrupt status
*/

int32 ret;

if((ret= rtk_gpio_intrStatus_clean (10)) != RT_ERR_OK)
    return ret;
```

2.7. Set GPIO Interrupt Mode

The ***rtk_gpio_intr_set*** API would set GPIO interrupt mode. The GPIO mode would be GPIO_INTR_DISABLE/GPIO_INTR_ENABLE_FALLING_EDGE/GPIO_INTR_ENABLE_RISING_EDGE/ GPIO_INTR_ENABLE_BOTH_EDGE.

Example:

```
/*
  Set GPIO pin 10 interrupt GPIO_INTR_ENABLE_FALLING_EDGE mode
*/

int32 ret;

if((ret= rtk_gpio_intr_set (10, GPIO_INTR_ENABLE_FALLING_EDGE)) !=
RT_ERR_OK)
```

```
    return ret;
```

2.8. Get GPIO Interrupt Mode

The ***rtk_gpio_intr_get*** API would get GPIO interrupt mode. The GPIO mode would be GPIO_INTR_DISABLE/GPIO_INTR_ENABLE_FALLING_EDGE/GPIO_INTR_ENABLE_RISING_EDGE/ GPIO_INTR_ENABLE_BOTH_EDGE.

Example:

```
/*
  Get GPIO pin 10 interrupt mode
*/

rtk_gpio_intrMode_t intrMode;
int32 ret;

if((ret= rtk_gpio_intr_get (10, &intrMode)) != RT_ERR_OK)
    return ret;
```

2.9. Register ISR Handler

The ***rtk_irq_gpioISR_register*** API would register GPIO pin interrupt service routine.

Example:

```
/*
  Register GPIO pin 10 interrupt service routine
*/
void isr(void)
{
…
}

int32 ret;

if((ret= rtk_irq_gpioISR_register (10, isr)) != RT_ERR_OK)
    return ret;
```

2.10. Unregister ISR Handler

The ***rtk_irq_gpioISR_unregister*** API would unregister GPIO pin interrupt service routine.

Example:

```
/*
  Unregister GPIO pin 10 interrupt service routine
*/

int32 ret;

if((ret= rtk_irq_gpioISR_unregister (10)) != RT_ERR_OK)
    return ret;
```

Realtek Semiconductor Corp.
Headquarters
No. 2, Innovation Road II, Hsinchu Science Park,
Hsinchu 300, Taiwan, R.O.C.
Tel: 886-3-5780211 Fax: 886-3-5776047
www.realtek.com

全文阅读已结束，下载本文需要使用

200 积分

下载此文档

阅读了该文档的用户还阅读了这些文档

RTL9607C_LED_Apr

RTL9607C_RTK_Por

RTL9607C_L2_Table

RTL9607C_Storm_Fi

EngNote_RTL9607C note_V01(2019111

Application_Nc

发表评论

验证码： 换一张

匿名评论

提交