

P9412

Wireless Power TRx WattShare™ Solution for Wireless Charging for Wireless Power Charging

Description

The P9412 CSP TRx evaluation board can be used to demonstrate the features and performance of the P9412 Wireless Power Transceiver solution for Mobile Device with a custom Rx only or TRx coils. The intuitive top-level placement of components, layout, and controls simplify the design-in process, optimizes the user experience, and emphasizes the impressive level of integration and abundance of useful features that this device offers.

The device is powered by a Renesas Proprietary TRx coil that can be copied when used with Renesas products (Worldwide patent pending). The P9412 operates in RX mode or TX mode depending on the setting and the firmware loaded to its Multiple-Time Programmable memory (MTP).

Features

- P9412 CSP TRx WattShare™ Wireless power Solution
- Delivers over 30W output power as a Receiver
- Delivers over 6W output power as Transmitter
- Integrated high-performance capacitor divider
- XY Position Sensing Technology
- I2C Connector
- Multiple-Time Programmable Memory (MTP)
- Fully assembled with test points and coil fixture
- 6-layer PCB with 1oz. copper

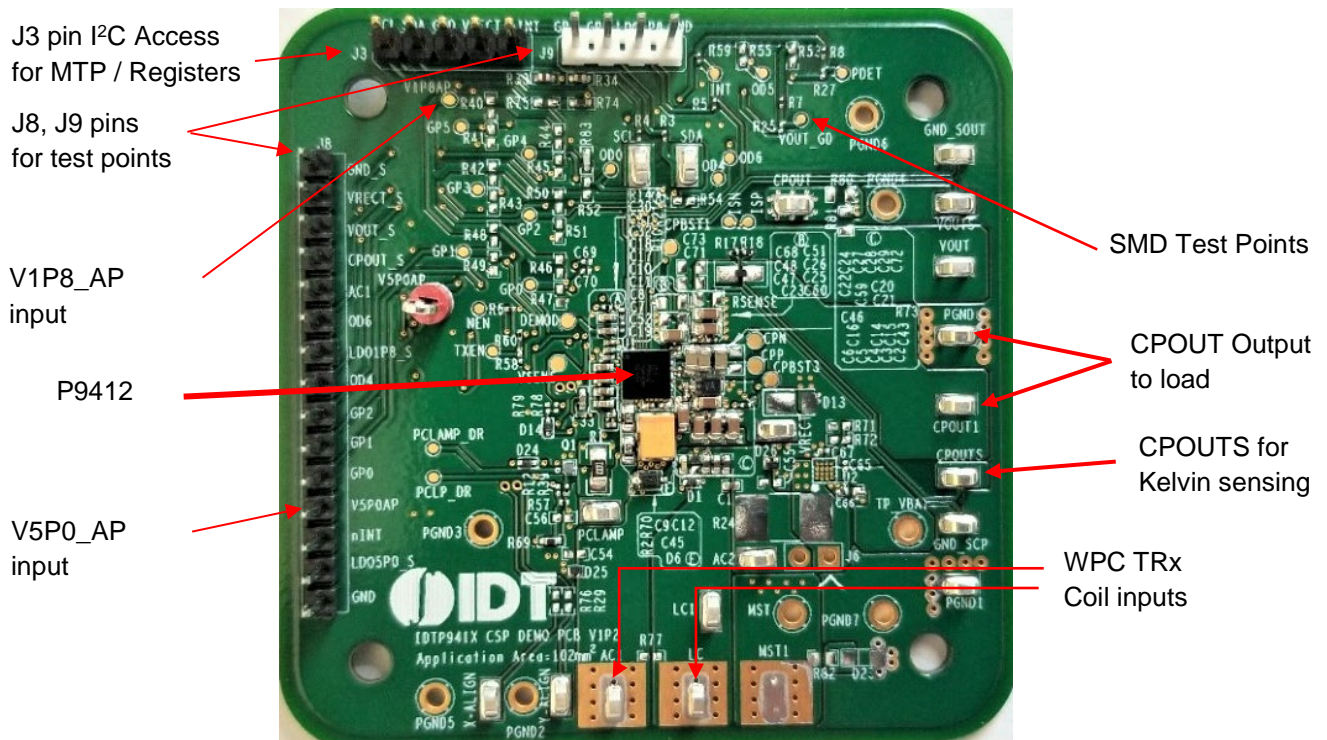


Figure 1. P9412 CSP Demo Board v1.3

Contents

| | |
|---|-----------|
| 1. Usage Guide | 7 |
| 1.1 Quick-Start Guide for Rx and Tx Mode Operation | 7 |
| 1.2 Using the Windows GUI | 8 |
| 1.2.1. Using the GUI to Program the P9412 | 8 |
| 1.2.2. Using the GUI to Read / Write to Registers | 12 |
| 1.2.3. Using the GUI to Run in High Power Capacitor Divider Mode..... | 15 |
| 1.2.4. Using the GUI to Run in TX Mode | 18 |
| 1.3 I ² C Function | 21 |
| 1.3.1. I2C Read back of MTP contents | 22 |
| 2. Registers | 23 |
| 2.1.1. <i>Identification and Revision Registers</i> | 23 |
| 2.1.1.1. Chip ID Register, Chip_ID_L (0x00), Chip_ID_H (0x01)..... | 23 |
| 2.1.1.2. Chip Revision and Font Register, Chip_Rev (0x02) | 23 |
| 2.1.1.3. Customer ID Register, Customer ID (0x03)..... | 23 |
| 2.1.1.4. Firmware Major Rev. Registers, FW_Major_Rev_L (0x04), FW_Major_Rev_H (0x05)..... | 23 |
| 2.1.1.5. Firmware Minor Rev. Registers, Minor_Rev_L (0x06), FW_Minor_Rev_H (0x07) | 23 |
| 2.1.1.6. Firmware Date/Time Registers, FW_Date_Code (0x 08~13), FW_Timer_Code (0x 14~1B) | 23 |
| 2.1.1.7. Configuration Major Revision Registers, CFG_Major_Rev (0x1C)..... | 24 |
| 2.1.1.8. Configuration Minor Revision Registers, CFG_Minor_Rev (0x1D)..... | 24 |
| 2.1.1.9. Reference Design Version Register, RefDesignVer (0x4A)..... | 24 |
| 2.1.2. <i>Status and Interrupt Registers</i> | 25 |
| 2.1.2.1. Status Registers, Status_L (0x34), Status_H (0x35) | 25 |
| 2.1.2.2. Interrupt Registers, INT_L (0x36), INT_H (0x37)..... | 25 |
| 2.1.2.3. Interrupt Enable Registers, INT_Enable_L (0x38), INT_Enable_H (0x39) | 26 |
| 2.1.2.4. Interrupt Clear Registers, INT_Clear_L (0x3A), INT_Clear_H (0x3B) | 27 |
| 2.1.2.5. TRX System Operating Mode Register, Sys_Op_Mode (0x4C) | 27 |
| 2.1.2.6. Over-Current Status Register, OC_Status (0xB5) | 27 |
| 2.1.3. <i>Battery Status and Power Transfer Registers</i> | 28 |
| 2.1.3.1. Charge Status Register, CHG_Status (0x3E)..... | 28 |
| 2.1.3.2. End of Power Transfer Register, EPT_Code (0x3F) | 28 |
| 2.1.4. <i>Operation Parameters Registers</i> | 28 |
| 2.1.4.1. Vout Set Register, Vout_Set_L (0x6C), Vout_Set_H (0x6D) | 28 |
| 2.1.4.2. TRX Vout Voltage Registers, Vout_L (0x42), Vout_H (0x43) | 29 |
| 2.1.4.3. Iout Limit Set Register (0x3D) | 29 |
| 2.1.4.4. TRX Iout / Iin Value Registers, Iout / Iin_L (0x44), Iout / Iin_H (0x45) | 29 |
| 2.1.4.5. TRX Vrect Voltage Registers, Vrect_L (0x40), Vrect_H (0x41) | 29 |
| 2.1.4.6. TRX Die Temperature Registers, DieTemp_L (0x46), DieTemp_H (0x47)..... | 29 |
| 2.1.4.7. TRX AC Frequency Registers, AC_Freq_L (0x48), AC_Freq_H (0x49)..... | 29 |
| 2.1.5. <i>Command and Communication Registers</i> | 30 |
| 2.1.5.1. System Command Register, SYS_CMND_L (0x4E), SYS_CMND_H (0x4F)..... | 30 |
| 2.1.5.2. Config Table Command Register, Config_CMND (0xCA) | 30 |
| 2.1.5.3. TRX Header Register (PropPkt Send), TRX_Header_Out (0x50)..... | 32 |
| 2.1.5.4. TRX Data Value1 Register (PropPkt Send), TRX_Data_VALUE1_Out (0x51)..... | 33 |
| 2.1.5.5. TRX Data Value 2~5 Reg. (PropPkt Send), TRX_Data_Value2_5_Out (0x52, 0x53, 0x54, 0x55).. | 33 |
| 2.1.5.6. TRX Header Register (PropPkt Received), TRX_Header_In (0x58) | 34 |

| | | |
|-----------|---|----|
| 2.1.5.7. | TRX Data Value 1~2 Register (PropPkt Received), TRX_Data_Value1_2_In (0x59, 0x5A) | 34 |
| 2.1.5.8. | Com Channel Send Size Register, CC_Send_Size_L (0x140), CC_Send_Size_H (0x141) | 34 |
| 2.1.5.9. | Com Channel Send Index Register, CC_Send_Index_L (0x142), CC_Send_Index_H (0x143) | 34 |
| 2.1.5.10. | Com Channel Receive Size Register, CC_Recv_Size_L (0x144), CC_Recv_Size_H (0x145) | 35 |
| 2.1.5.11. | Com Channel Receive Index Reg., CC_Recv_Index_L (0x146), CC_Recv_Index_H (0x147)..... | 35 |
| 2.1.5.12. | Com Channel Status Register, CC_Status (0x148)..... | 35 |
| 2.1.5.13. | Pending Packets Register, Pend_Pkts (0x149)..... | 35 |
| 2.1.5.14. | ADT Packet Time Out Register, ADT_Timeout_PKT (0x150) | 35 |
| 2.1.5.15. | ADT Stream Time Out Register, ADT_Timeout_STR (0x151) | 35 |
| 2.1.5.16. | ADT Error Code Register, ADT_Error_Code (0x14D) | 36 |
| 2.1.5.17. | ADT Buffer Registers, (0x0800 ~ 0x0FFF) | 36 |
| 2.1.5.18. | Frequency Shift Keyed modulation (FSK) Transmitter to Receiver Communication | 37 |
| 2.1.5.19. | FSK Communication Protocol | 37 |
| 2.1.6. | <i>HW Control and Monitor Registers</i> | 37 |
| 2.1.6.1. | Ping Frequency Register, PingFreq_L (0x6A), PingFreq_H (0x6B) | 37 |
| 2.1.6.2. | HW Flag Register, HW_Flag (0x81) | 38 |
| 2.1.6.3. | Over Voltage Protection Register, OV_Set (0xB3) | 38 |
| 2.1.6.4. | RX Mode Communication Modulation FET Register, CMFET_L (0xF4), CMFET_H (0xF5) | 38 |
| 2.1.6.5. | RX Mode AFC Communication Modulation FET Register, AFC_CMFET (0xB2) | 38 |
| 2.1.6.6. | RX Mode High Vout Communication Modulation FET Register, HiVout_CMFET (0x11B)..... | 39 |
| 2.1.6.7. | Align X Register, AlignX (0xB0)..... | 39 |
| 2.1.6.8. | Align Y Register, AlignY (0xB1)..... | 39 |
| 2.1.6.9. | Align adc Offset Registers, AlignAdcOffX (0x164), AlignAdcOffY (0x165) | 40 |
| 2.1.6.10. | Align Slope1 Registers, AlignSlope1X (0x166), AlignSlope1Y (0x167) | 40 |
| 2.1.6.11. | Align Slope2 Registers, AlignSlope2X (0x168), AlignSlope2Y (0x169) | 40 |
| 2.1.6.12. | Align Offset Registers, AlignOffX (0x16A), AlignOffY (0x16B)..... | 41 |
| 2.1.6.13. | Align Threshold Registers, AlignThreshX (0x16C), AlignThreshY (0x16D) | 41 |
| 2.1.7. | <i>Vrect Control Registers</i> | 41 |
| 2.1.7.1. | Target_Vrect Register, Vrect_Target_L(0x90), VrectTarget_H (0x91) | 41 |
| 2.1.7.2. | Vrect Knee Register, PwrKnee (0x92)..... | 41 |
| 2.1.7.3. | Vrect Correction Factor Register, VrCorrFactor (0x93) | 42 |
| 2.1.7.4. | Vrect Maximum Correction Register, VrMaxCorr_L (0x94), VrMaxCorr_H (0x95) | 42 |
| 2.1.7.5. | Vrect Minimum Correction Register, VrMinCorr_L (0x96), VrMinCorr_H (0x97) | 42 |
| 2.1.7.6. | Vrect Adjust Register, VRectAdj (0x5E) | 42 |
| 2.1.8. | <i>Capacitor Divider Registers</i> | 42 |
| 2.1.8.1. | Capacitor Divider Mode Status Register, CDMoSts (0x100) | 42 |
| 2.1.8.2. | Capacitor Divider Mode Request Register, CDMoReq (0x101)..... | 42 |
| 2.1.8.3. | TRX CPout Voltage Registers, VCPout_L (0x10C), VCPout_H (0x10D)..... | 43 |
| 2.1.8.4. | Capacitor Divider Vout Threshold Reg., CD_Vout_Thd_L (0x10A), CD_Vout_Thd_H (0x10B) | 43 |
| 2.1.8.5. | Capacitor Divider Frequency Set Register, CD_Freq_L (0x108), CD_Freq_H (0x109)..... | 43 |
| 2.1.9. | <i>Foreign Object Detection Registers</i> | 43 |
| 2.1.9.1. | RX FOD Adjustable Parameters Registers, (0x70 ~ 0x7F) | 43 |
| 2.1.9.2. | TX FOD Threshold Registers, TX_FOD_Thrsh_L (0xD4), TX_FOD_Thrsh_H (0xD5)..... | 44 |
| 2.1.9.3. | TX FOD Gain Register, TX_FOD_Gain (0xD1) | 44 |
| 2.1.9.4. | TX FOD Offset Registers, TX_FOD_Offset_L (0xD2), TX_FOD_Offset_H (0xD3) | 44 |
| 2.1.9.5. | TX FOD Offset Option Register, FOD_OffsetOpt (0xA3)..... | 44 |
| 2.1.10. | <i>WPC Basic and Extended Protocol Registers</i> | 45 |
| 2.1.10.1. | EPP Q-Factor Register, EPP_Q_Factor (0x83)..... | 45 |

| | | |
|------------|--|----|
| 2.1.10.2. | EPP Q-Factor A Register, EPP_Q_Fact_A (0x138) | 46 |
| 2.1.10.3. | EPP Q-Factor B Register, EPP_Q_Fact_B (0x139) | 46 |
| 2.1.10.4. | EPP TX Guaranteed Power Register, EPP_TXGuarPwr (0x84) | 46 |
| 2.1.10.5. | EPP TX Potential Power Register, EPP_TXPotentPwr (0x85) | 46 |
| 2.1.10.6. | EPP TX Capability Flag Register, EPP_TXCapaFlag (0x86) | 46 |
| 2.1.10.7. | EPP Renegotiation Status Register, EPP_RN_Sts (0x87) | 46 |
| 2.1.10.8. | EPP Current RPP Header Register, MPCur_RPP (0x88) | 47 |
| 2.1.10.9. | EPP Current Negotiated Power Register, MPCur_NegPwr (0x89)..... | 47 |
| 2.1.10.10. | EPP Current Maximum Power Register, MPCur_MaxPwr (0x8A)..... | 47 |
| 2.1.10.11. | EPP Current FSK Modulation Register, MPCur_FSK (0x8B) | 47 |
| 2.1.10.12. | EPP Request RPP Header Register, MPReq_RPP (0x8C)..... | 47 |
| 2.1.10.13. | EPP Request Re-Negotiated Power Register, MPReq_NegPwr (0x8D) | 47 |
| 2.1.10.14. | EPP Request Maximum Power Register, MPReq_MaxPwr (0x8E)..... | 47 |
| 2.1.10.15. | EPP Request FSK Modulation Register, MPReq_FSK (0x8F) | 48 |
| 2.1.10.16. | WPC Spec Revision Register, WPC_SpecRev (0xB9)..... | 48 |
| 2.1.10.17. | EPP Rx Manufacturer Code Reg., MpRxManufCode_L (0xBA), MpRxManufCode_H (0xBB).... | 48 |
| 2.1.10.18. | WPC Identification Register, WPC_ID (0xE0 ~ 0xE3) | 48 |
| 2.1.10.19. | Extended Identification Packet Register, WPC_ExtID (0xE4 ~ 0xEB)..... | 49 |
| 2.1.10.20. | Signal Strength Packet Register, SSPValue (0xB4) | 49 |
| 2.1.10.21. | Control Error Packet Register, CEPValue (0x5F) | 49 |
| 2.1.10.22. | RX Power Register, Rx_Pwr_L (0xCE), Rx_Pwr_H (0xCF) | 49 |
| 2.1.10.23. | WPC Manufacturer ID Register, WPC_ManufID_L (0x15E), WPC_ManufID_H (0x15F) | 50 |
| 2.1.11. | <i>Proprietary High-Power Protocol Registers</i> | 50 |
| 2.1.11.1. | Proprietary Tx Potential Power Register, PropTxPotenPwr (0xC4)..... | 53 |
| 2.1.11.2. | Proprietary Requested Power Register, PropReqPwr (0xC5) | 53 |
| 2.1.11.3. | Proprietary Current Power Register, PropCurrPwr (0xC6) | 53 |
| 2.1.11.4. | Proprietary Negotiated Power Step Size Register, PropModePwrStep (0xC7) | 53 |
| 2.1.11.5. | Proprietary Mode Status Register, PropModeStatus (0xC8) | 53 |
| 2.1.11.6. | Proprietary Mode Error Register, PropErrStatus (0xC9)..... | 53 |
| 2.1.12. | <i>TX Mode Registers</i> | 54 |
| 2.1.12.1. | TX Status Registers, Status_L (0x34), Status_H (0x35)..... | 54 |
| 2.1.12.2. | TX Interrupt Registers, INT_L (0x36), INT_H (0x37) | 54 |
| 2.1.12.3. | TX Interrupt Enable Registers, INT_Enable_L (0x38), INT_Enable_H (0x39)..... | 55 |
| 2.1.12.4. | TX Interrupt Clear Registers, INT_Clear_L (0x3A), INT_Clear_H (0x3B)..... | 56 |
| 2.1.12.5. | System TX Command Register, TX_CMND (0x4D) | 56 |
| 2.1.12.6. | TX Mode System Command Register, TxSysCmnd_L (0x4E), TxSysCmnd_H (0x4F)..... | 57 |
| 2.1.12.7. | TX Mode CEP Threshold Register, TxCepThrshVal (0xD6)..... | 57 |
| 2.1.12.8. | TX Mode CEP Threshold Count Limit Register, TxCepThrshCntLmt (0xD7) | 57 |
| 2.1.12.9. | TX Mode API Current Limit Registers, I_API_Limit_L (0x56), I_API_Limit_H (0x57) | 58 |
| 2.1.12.10. | TX Mode API Current Hysteresis Register, I_API_Hys (0x82) | 59 |
| 2.1.12.11. | TX Mode API Current Limit Offset Registers, I_API_Offset_L (0xF6), I_API_Offset_H (0xF7) ... | 59 |
| 2.1.12.12. | TX Ping Frequency (Period) Register, PingFreqPer_L (0x90), PingFreqPer_H (0x91)..... | 59 |
| 2.1.12.13. | TX Ping Duty Cycle Register, PingDC (0x92)..... | 59 |
| 2.1.12.14. | TX Minimum Duty Cycle Register, MinDC (0x93)..... | 59 |
| 2.1.12.15. | TX Minimum Frequency (Period) Register, MinFreqPer_L (0x94), MinFreqPer_H (0x95) | 60 |
| 2.1.12.16. | TX Maximum Frequency (Period) Register, MaxFreqPer_L (0x96), MaxFreqPer_H (0x97) | 60 |
| 2.1.12.17. | TX Mode Operating Period Register, TxPeriod_L (0xA4), TxPeriod_H (0xA5) | 60 |
| 2.1.12.18. | TX Mode Operating Duty Cycle Register, TxDuty (0xA6)..... | 60 |

| | | |
|------------|---|-----------|
| 2.1.12.19. | TX Mode Over-Voltage Protection Register, Tx_OVP_L (0x9E), TX_OVP_H (0x9F)..... | 60 |
| 2.1.12.20. | TX Mode Over-Current Protection Register, TX_OCP_L (0xA0), TX_OCP_H (0xA1)..... | 61 |
| 2.1.12.21. | TX Conflict Threshold Register, TxConfThresh (0x134) | 61 |
| 2.1.12.22. | TX Conflict Count Register, TxConfCnt (0xAF) | 61 |
| 2.1.12.23. | TX Power Register, TX_Power_L (0xAC), TX_Power_H (0xAD) | 61 |
| 2.1.12.24. | TX Dead Time Register, TX_DeadTime (0x9C) | 61 |
| 2.1.12.25. | TX Control Delay Register, TX_ControlDelay (0xAE)..... | 62 |
| 2.1.13. | <i>Commonly Used Registers</i> | 62 |
| 2.1.13.1. | Monitoring, Status, Interrupt, and Operating Mode Registers..... | 62 |
| 2.1.13.2. | Settings and Command Registers..... | 62 |
| 3. | Schematic Diagram | 63 |
| 4. | Bill of Materials | 64 |
| 5. | Board Layout | 66 |
| 5.1 | FTDI Dongle | 72 |
| 6. | Ordering Information | 74 |
| 7. | Revision History | 74 |

Figures

| | | |
|------------|---|----|
| Figure 1. | P9412 CSP Demo Board v1.3..... | 1 |
| Figure 2. | Capacitor Divider Mode Transition Sequence..... | 7 |
| Figure 3. | GUI Software | 8 |
| Figure 4. | FTDI USB-I2C Bridge Connected to I2C Terminal J3 of P9412 CSP TRx Demo Board..... | 8 |
| Figure 5. | Initial Screen of P9412 GUI..... | 9 |
| Figure 6. | USB-Bridge is Detected, P9412 is Not Connected | 9 |
| Figure 7. | P9412 MTP Programming using I ² C Slave Device Address 0x3C | 10 |
| Figure 8. | MTP Programming Successful..... | 10 |
| Figure 9. | GUI Detects an Unprogrammed Part and Issues a Warning | 11 |
| Figure 10. | Firmware Program Verification Success | 11 |
| Figure 11. | Verify FW Revision and Date Code..... | 12 |
| Figure 12. | I2C Read / Write Registers..... | 13 |
| Figure 13. | Basic 1 Tab, after a “Read 1 time” Operation | 13 |
| Figure 14. | VO _{UT} Adjustments Can be made in Three Ways..... | 14 |
| Figure 15. | Check Cap Divider Mode | 15 |
| Figure 16. | Enable Cap Divider Mode Operation..... | 16 |
| Figure 17. | P9412 Successfully Enters Cap Div. Mode and is Ready to Transition to High-Power Operation..... | 16 |
| Figure 18. | P9412 Transition to High Power Operation..... | 17 |
| Figure 19. | Using the GUI to Manually Increase the RX V _{out} Voltage..... | 17 |
| Figure 20. | Using the GUI to Manually Decrease the RX V _{out} Voltage | 18 |
| Figure 21. | Basic Registers Initial Read Back; Before TX Mode Entry..... | 19 |
| Figure 22. | Tx Mode Entry, Without an Rx Placed on the P9412 Coil; V _{rect} and V _{out} Voltage Read Back ~7V ... | 19 |
| Figure 23. | Tx Mode, With an Rx Placed on the P9412 Coil; Use the GUI to Monitor the Status..... | 20 |
| Figure 24. | Exit Tx Mode, With an Rx Placed on the P9412 Coil; Use the GUI to Send Exit Command..... | 20 |
| Figure 25. | Verify Exit TX Mode; Use the GUI to Read Mode | 21 |
| Figure 26. | I2C Read Protocol using P9412 | 21 |

Figure 27. I2C Write Protocol using P941222

Figure 28. Example of differential bi-phase encoding.37

Figure 29. Example of asynchronous serial byte format.37

Figure 30. Proprietary High Power Protocol Flow Chart52

Figure 31. P9412 DEMO Board v1.3 Schematic.....63

Figure 32. Top and Top Silkscreen Layer66

Figure 33. Inner1 GND Layer67

Figure 34. Inner2 POWER/Signal/GND Layer.....68

Figure 35. Inner3 POWER/ GND Layer.....69

Figure 36. Inner4 POWER/Signal/GND Layer.....70

Figure 37. Bottom Layer71

Figure 38. FTDI Dongle for Programming the Firmware into P9412 MTP72

Figure 39. FTDI Dongle Schematic for Programming the Firmware into P9412 MTP and Reading Registers73

Tables

Table 1. Bill of Materials64

1. Usage Guide

The P9412 CSP TRx Demo board is designed to demonstrate the performance and functionality of the P9412 wireless transceiver in a lab bench test environment. For complex or electrically sensitive situations, it is recommended to use the reference layout to integrate this design into the final system in order to eliminate hardware limitations or signal degradation introduced by long leads.

With no computer interface, the demo board can function in its pre-programmed Rx mode of operation together with a WPC compatible Tx transmitter such as the P9235A, P9236A, P9247 Tx EVKIT, or the P9260 Automotive Tx.

1.1 Quick-Start Guide for Rx and Tx Mode Operation

The P9412, when used as a receiver (Rx), has three Capacitor Divider operating modes:

- Disable: Vout is on, but CPout is off
- Bypass: CPout = Vout
- Cap Divider: CPout = Vout/2

Any mode change between Bypass and Cap divider mode must go through Disable (see Figure 2). Note that the firmware handles the transition through the Disable mode automatically and the information is being provided for reference. CPout must not be loaded during mode transitions between Bypass and Cap Divider.

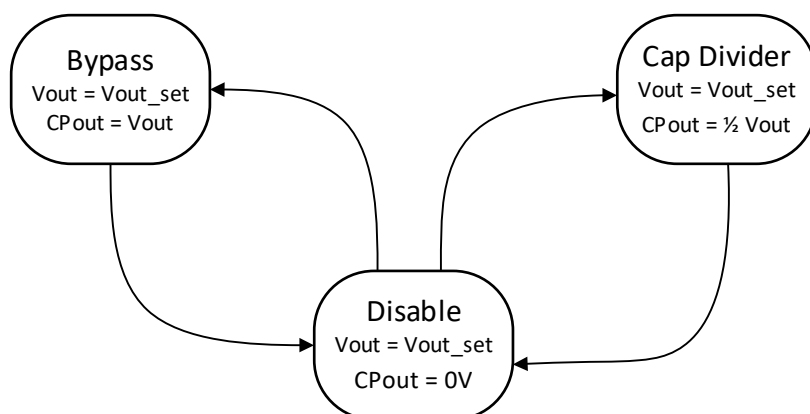


Figure 2. Capacitor Divider Mode Transition Sequence

The default configuration for the EVK demo board is for LDO1P8 to be connected to the SCL and SDA I2C lines only. To apply 1.8VDC from the target Application PCB, wire 1.8VDC to the V1P8_AP input test point and add a GND reference wire (can be shared with VBAT GND wire). In order for the P9412 Demo PCB to properly operate, the V1P8_AP pin should be powered by a 1.8VDC source at all times.

With no load applied to the VOUT or CPOUT pins, place the P9412 Rx coil with the windings facing down (toward the Tx coil) centered above a valid Tx unit with a 1mm non-metallic spacer placed between the Tx interface and the Rx coil. The charging pad can be a pre-powered WPC Tx, or can be powered after placing the Rx coil into position.

Verify that power is available at the Rx output by measuring the voltage at the CPOUT1 test point to ground (GND). The P9412 will start up in Bypass mode (i.e., Cap. Div. is not running and VOUT is transferred directly to the CPOUT node) and can deliver up to 1.5A at 12V (if the Tx can support this much power transfer). A heavy load (loads > 1A) should not be applied until the Cap Divider Bypass Mode has been verified (CPOUT = VOUT). Additionally, startup into heavy loads can cause the TX to fail to connect due to FOD or OC. If higher output power is desired, the integrated capacitor divider must be set into Cap Divider mode where up to 3A load can be

connected between CPOUT1 and the GND terminals. For more information, see “Using the GUI to run in high power Capacitor Divider mode”.

The I2C connector J3 can be used to program the P9412 or R/W to any valid Read/Write registers. The included USB-I2C Bridge (FT4222 Dongle) and GUI for Windows PC allow reading, writing to registers, and programming new firmware into MTP.

GUI software is shown in Figure 3. Use this revision or higher (latest revision available).



Figure 3. GUI Software

1.2 Using the Windows GUI

Connect the Bridge to PC via USB connector. Attach the Bridge to the I²C terminal J3 on the P9412 EVAL board as shown on Figure 44. Line up the GND (black) of the Dongle with the pin 3 of J3 connector.

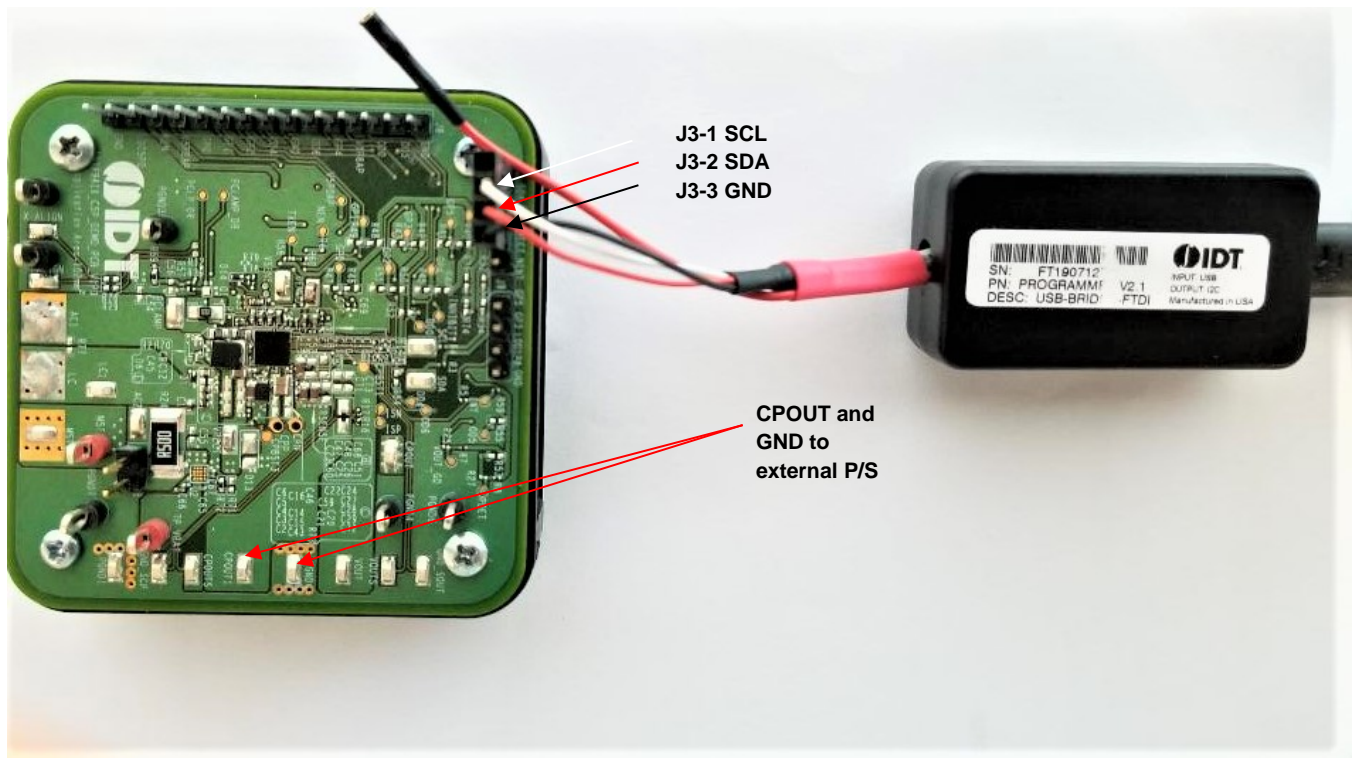


Figure 4. FTDI USB-I2C Bridge Connected to I2C Terminal J3 of P9412 CSP TRx Demo Board

1.2.1. Using the GUI to Program the P9412

To program the device, remove the P9412 demo board coil from the Tx, connect GND, SDA, SCL to the FTDI USB-I2C Bridge and power CPOUT from an external Power Supply set to 5V to 8V (see Figure 4).

Note: Field MTP updates are possible using the host AP using a similar method and I2C. Contact the factory for details.

1. Open the GUI program – P9412_Demo_GUI_v0.10r007_.exe or higher. The initial screen of the GUI is shown on Figure 5.

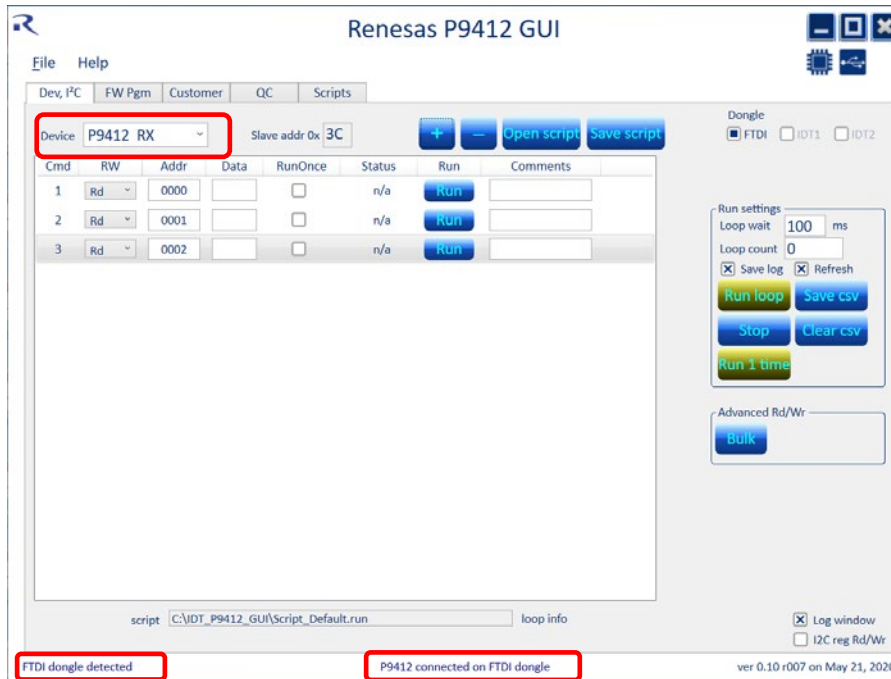


Figure 5. Initial Screen of P9412 GUI

Note: Make sure the external power supply is turned on and the USB-I2C Bridge is connected to the PC. Verify that the Device selected is the P9412 at the top of screen. Check the “FTDI dongle detected” and “P9412 connected on FTDI dongle” messages are shown at the bottom of the screen. If you do not see these messages, unplug the USB cable at the PC side, plug it in again and then check all connections.

If the message indicates that the “FTDI dongle detected” but the P9412 is not connected, see Figure 6 – it may be necessary to select the P9412 again using the “Device” pull-down menu and clicking on “P9412 RX”.

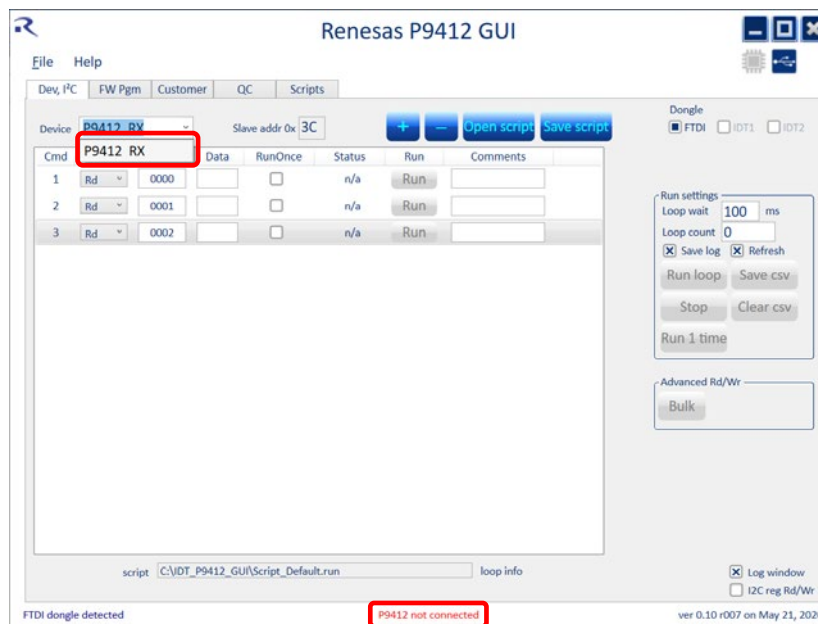


Figure 6. USB-Bridge is Detected, P9412 is Not Connected

- To update the MTP program, select the “FW Pgm” tab. Then make sure the proper file type is selected, either HEX or BIN box is checked and Press the “Load File” button, a pop-up window will appear. Navigate to the current P9412 FW *.* file and Open the file. See Figure 7, which indicates the FW file was read successfully.

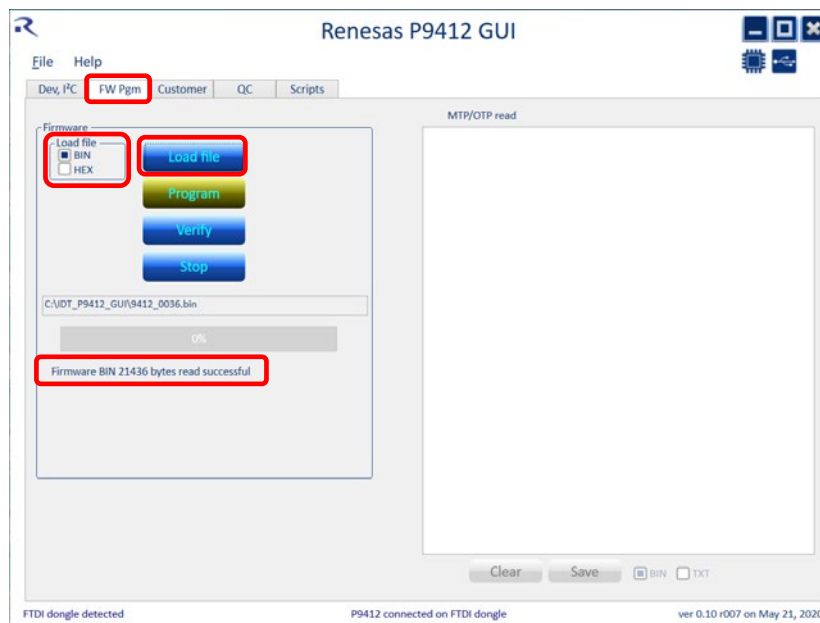


Figure 7. P9412 MTP Programming using I²C Slave Device Address 0x3C

- After loading the file, press the “Program” button, the MTP will be programmed and the GUI will indicate if successful or if there were errors. See Figure 8, which indicates that the programming was successful. If there are any errors during programming, attempt to program again.

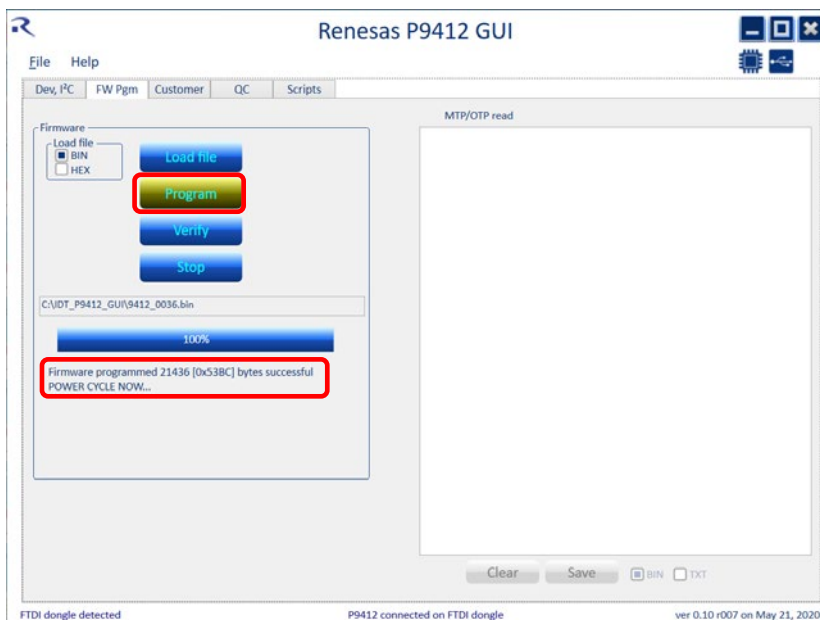


Figure 8. MTP Programming Successful

- a. If an un-programmed part is detected, the GUI will issue a warning message (see Figure 9). Follow the instructions: Turn off the external P/S and connect to Vrect or Vout, wait for 20 seconds, then turn on the P/S and try to program again by “clicking” on the OK button. Failure to connect the external P/S to Vrect or Vout when programming a blank IC can result in damage to the part.

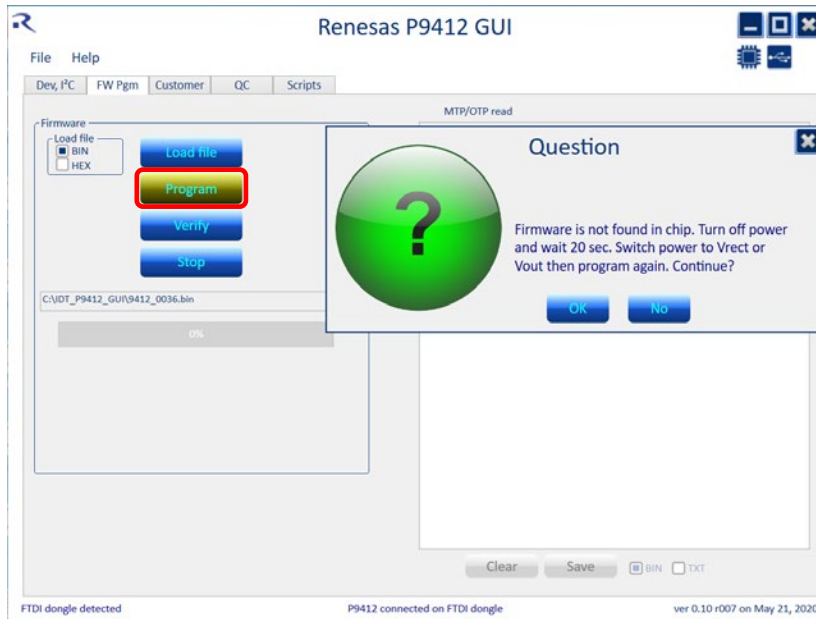


Figure 9. GUI Detects an Unprogrammed Part and Issues a Warning

- 4. If the firmware was programmed successfully, power cycle the external P/S (turn P/S off then on). Then press the “Verify” button. A total match should be indicated for successful programming (see Figure 10). If there are any errors, attempt to program again.

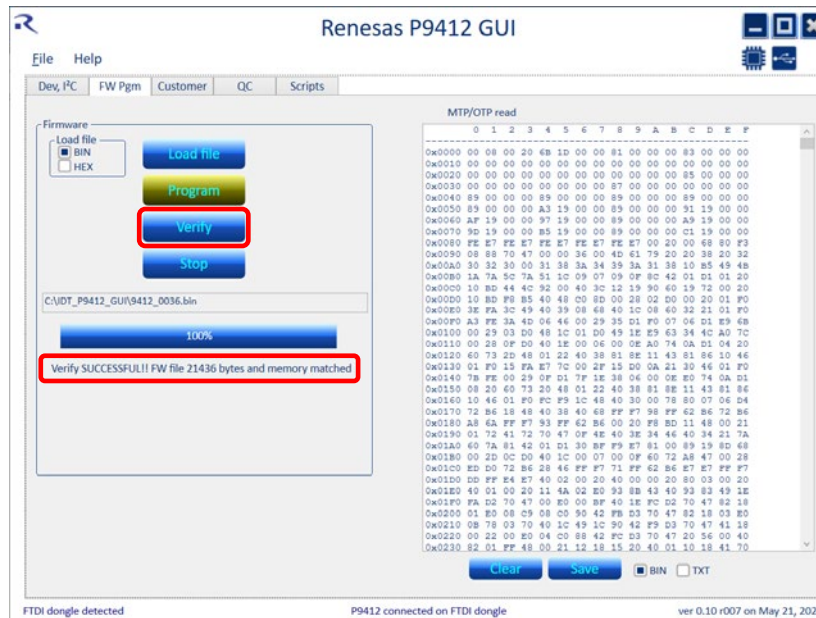


Figure 10. Firmware Program Verification Success

5. After programming and verification success.
 - a. Verify programming, it is recommended to select the “Basic 1” tab and press the “Read 1 time” button to check that the firmware revision and date code are correct.
 - b. Turn off the external Power Supply and either set up the P9412 for TRx mode (see “Using the GUI to run in TX mode” section), or remove the external power supply from CPOUT and place the P9412 on a compatible Tx.

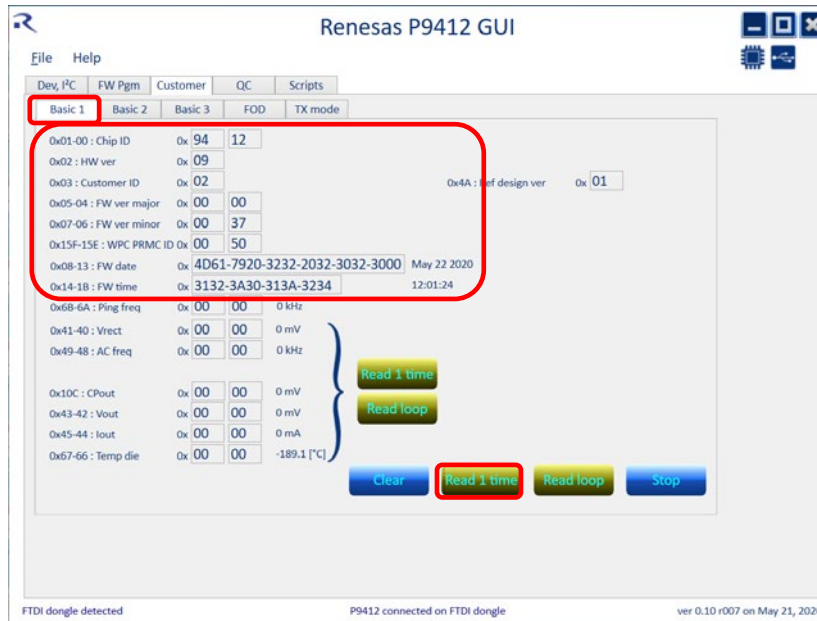


Figure 11. Verify FW Revision and Date Code

1.2.2. Using the GUI to Read / Write to Registers

Access to the P9412 status and control settings are done by reading and writing the I²C registers. The registers are described in “Registers.” To read and write to the registers, use one of the following procedures:

1. Place the P9412 on a compatible Tx. The P9412 is configured as a BPP Rx and will connect and initially start in Bypass mode, where the Capacitor Divider is not running and VOUT is transferred directly to the CPOUT node. The voltage at CPOUT and VOUT should be 5V.

2. To read and write specific registers, select the “Dev, I2C” tab and click the “I2C reg Rd/Wr” box. The I2C Rd/Wr block will stay accessible when different tabs are selected (see Figure 12).
 - a. For example, to check the Rx mode CPout voltage, read the 16bit - code of I²C register 0x10C. First enter 010C in the “Addr” field, select 2 “Bytes”, then press the “Read” button.

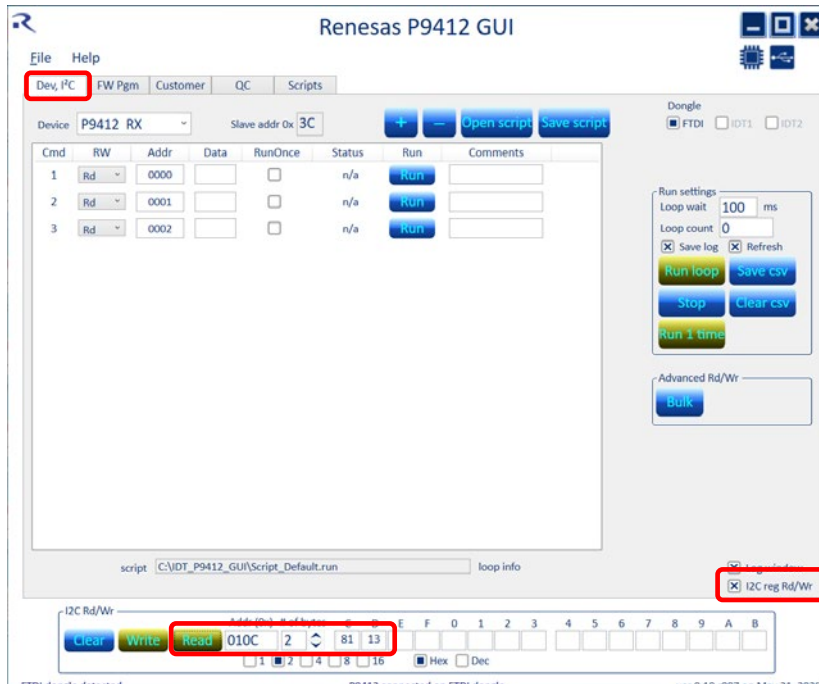


Figure 12. I2C Read / Write Registers

3. For bulk reading of common operating registers, select the Basic 1 tab and select one of the “Read 1 time” buttons depending on what section the register of interest is in. The register values within the bracketed section can be constantly monitored by selecting “Read loop”, and the monitoring can be stopped by selecting “Stop”. Similarly, all the registers on this tab can be monitored continuously by selecting the “Read loop” button at the bottom.

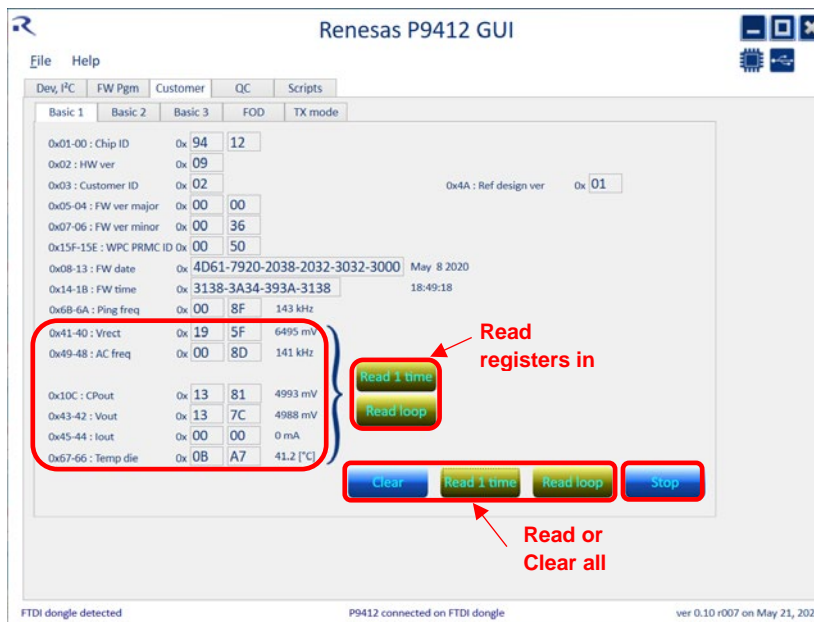


Figure 13. Basic 1 Tab, after a “Read 1 time” Operation

- Reciprocally, register 0x6C (valid for FW37 and above) is used to set the value of VOUT, as shown in Figure 14. Any register can be read (or written assuming it is a writeable register) from the I2C reg Rd/Wr tool at any time. Address is the starting address and the number of bytes should be specified prior to clicking “Read” for read, or “Write” for write operations.

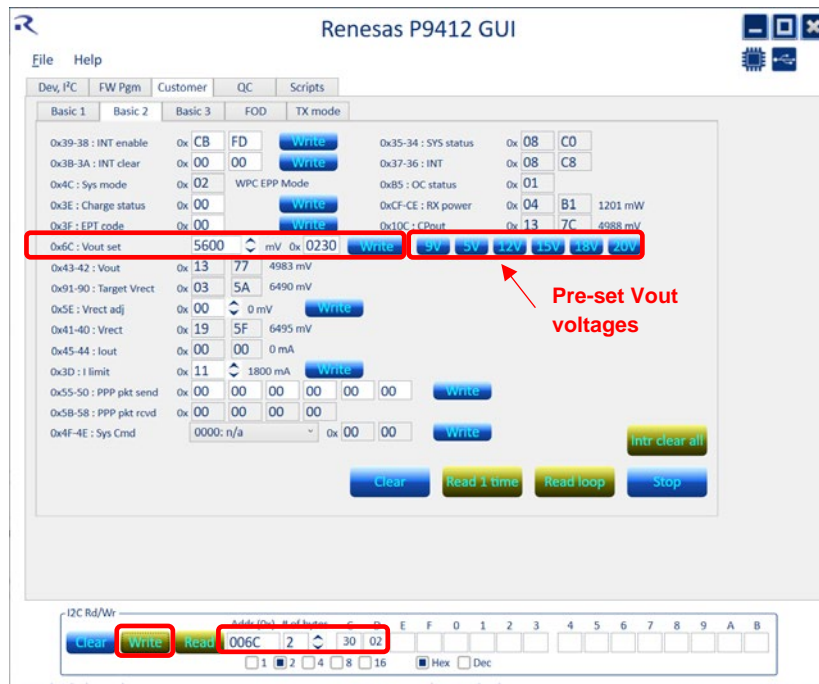


Figure 14. VOUT Adjustments Can be made in Three Ways

- Vout_Set = Value in 0.01V entered into 0x6C: From the “Basic 2” tab, Vout Set field, enter or select the voltage using the up/down arrows and then clicking Write, or
- Vout_Set = code_value (decimal value converted to hex) * 0.01 (V); 0x0230 corresponds to 5.6V and clicking Write in “I2C Rd/Wr” block, or
- Click on one of the pre-set values. When clicking a pre-set button it is not necessary to click the Write button.

Vout_Set in 40 mV step, 3.52 V ~ 20V range.

Note: The P9412 is configured as a BPP/EPP Rx and will connect and initially start in Bypass mode, CPOUT ≈ VOUT. In Bypass mode, the VOUT or CPOUT voltage is limited to a 12V maximum, and a standard BPP Tx is normally limited to 5W (typically CPOUT = 5V, 1A load on the Rx). In order to attain higher output power, an EPP TX or a proprietary TX, and running the P9412 in Cap Divider mode is required. For instructions on entering Cap Divider mode, see “Using the GUI to run in high power Capacitor Divider mode”.

1.2.3. Using the GUI to Run in High Power Capacitor Divider Mode

When preparing to deliver higher power or to use the integrated Capacitor Divider (CD), the following steps should be taken to allow the Cap Divider the opportunity to soft-start safely without causing wireless connection interruptions or cause excessive current to flow during start-up:

1. Reduce the load on CPOUT to 0A.
2. Set the Cap Divider mode to “Cap Div”.
3. Monitor the CD mode Status register for Cap Div Operation.
 - a. Or, check the INT register for the CD_MODECHANGE_INT (used to notify the AP).
4. Increase the CPOUT voltage and proprietary Tx input voltage as required (during Cap Div mode, CPOUT ≈ VOUT / 2).

Note: VOUT is only allowed to be set above 12V in Cap Div mode. Also, VOUT must be below 12V when exiting Cap Div mode. CPout must not be loaded during any Cap Divider mode transitions. When in Cap Div mode, change the CPOUT voltage by changing Vout_set to 2x the CPOUT target voltage.

To transition from low power to high power operation, follow the procedure outlined. Low power operation is with the Capacitor Divider in Bypass mode. Check the Cap Divider mode by using the “Basic 3” tab, and reading the Cap Div mode (see Figure 15).

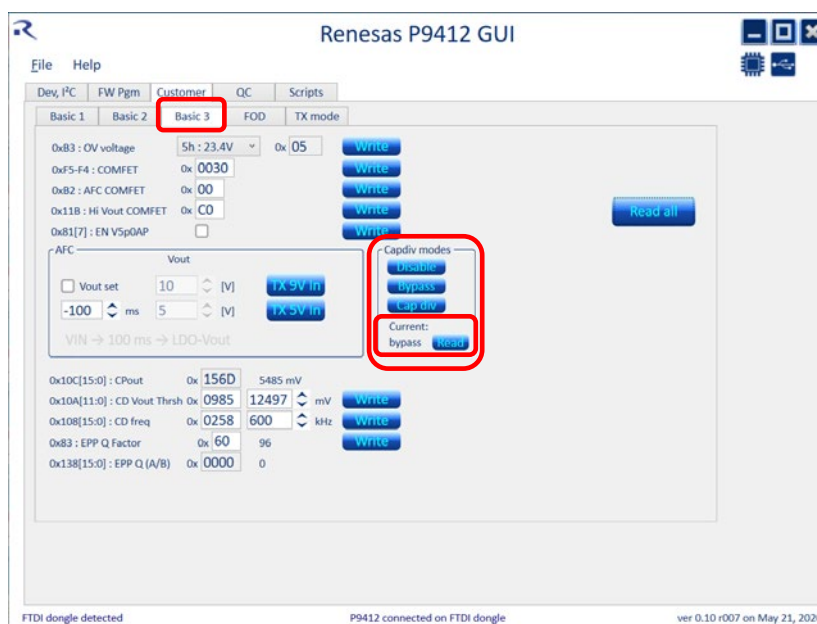


Figure 15. Check Cap Divider Mode

Check the VOUT voltage by measuring the VOUT test point or by using the GUI (“Basic 2” tab, Monitoring: Vout, and Vout_set). Once it is determined that the P9412 is in Bypass mode and the VOUT voltage is set to less than 12V, reduce the output load to 0mA.

Referring to Figure 16: select the QC tab, click the CD box, and then press the “Set Mode” button [step 1]. This is in preparation to set the P9247 QC TX to a higher input voltage and higher power transfer operation. By clicking the “Set Mode” button, the GUI is requesting a transition from Bypass mode to Cap Divider mode operation (see Figure 2). Verify Cap Divider operation by checking the voltage at VOUT and at CPOUT. The voltage at CPOUT should be half of the voltage at VOUT. Note that the Iout check field is indicating “Add load” [step 2], this is normal and prevents the transition to high power operation (or the increase of RX Vout and TX Vin voltages) without first adding a 200mA load to CPOUT. The 200mA load is recommended to make the output

voltage transition smoother and to reduce the coupled spikes that occur when the TX input power supply is being adjusted.

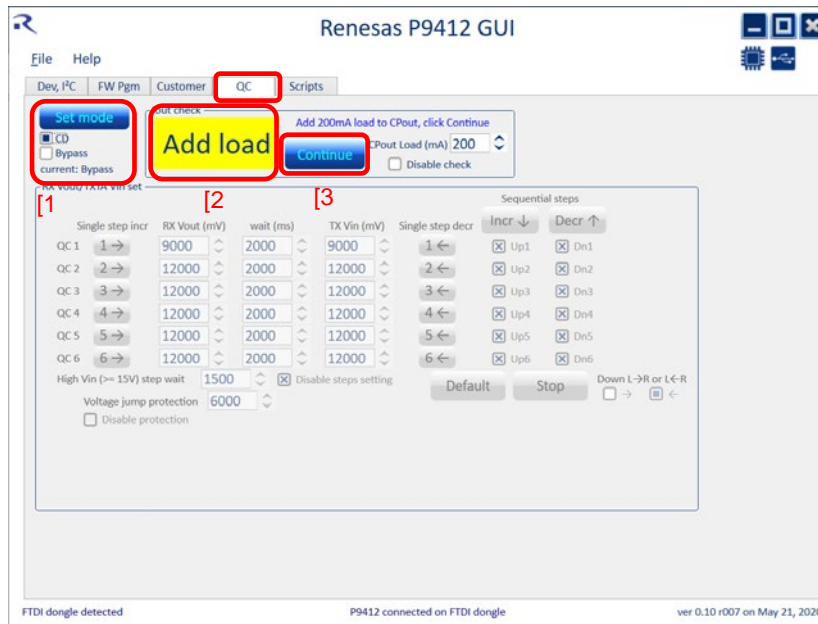


Figure 16. Enable Cap Divider Mode Operation

Add a 200mA load and click the “Continue” button [step 3]. By clicking the “Continue” button while the P9412 is placed on the P9247 QC TX, a “Load OK” message should be displayed as shown in Figure 17. This indicates that the GUI is now able to change the P9412’s VOUT voltage and send a command to the P9247 to change the TX input voltage after the time interval given. Note that if the RX Vout voltages indicate a maximum of 12V, it is necessary to press the “Default” button. The sequence the GUI follows for increasing the RX Vout and TX VIN voltages is: 1. Increase Vout_set to the level indicated; 2. Wait for the designated interval for the RX Vout to stabilize (Note: the wait period should not be less than 100ms for best results); 3. Increase the TX Vin voltage to increase the power transferred.

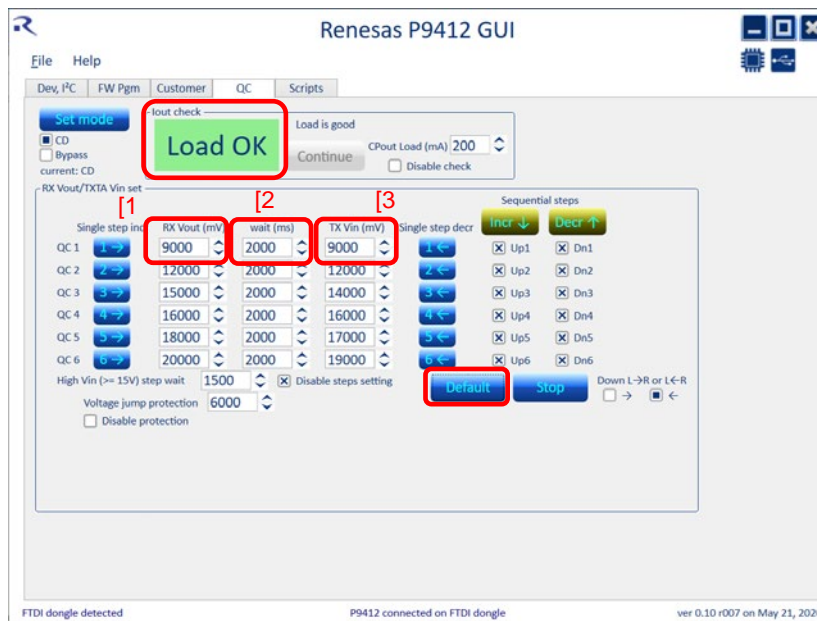


Figure 17. P9412 Successfully Enters Cap Div. Mode and is Ready to Transition to High-Power Operation

To increase the RX VOUT voltage and TX input voltage make sure all the Up and Down step boxes are checked. Then press the “Incr↓” button. The arrow direction indicates the sequence of the steps from Up1 to Up6. The step-up sequence may take several seconds to complete. Similarly, the system can be set back to 9Vout operation by clicking the “Decr↑” button.

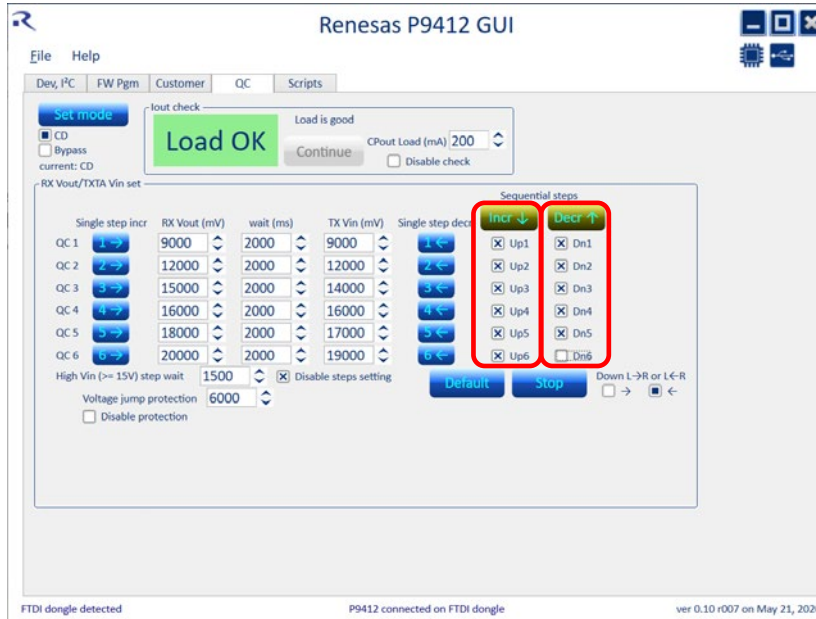


Figure 18. P9412 Transition to High Power Operation

In addition to using the Incr↓ and Decr↑ buttons it is possible to manually control the sequence using the “Single step incr” buttons. For example, to increase the RX Vout from 9V to 12V, press the “2→” button. The sequence the GUI follows for increasing the RX Vout voltage is: 1. Increase Vout_set to the 12V; 2. Wait for the 2 second interval for RX Vout to stabilize; 3. Increase the TX Vin voltage to 12V to increase the power transferred.

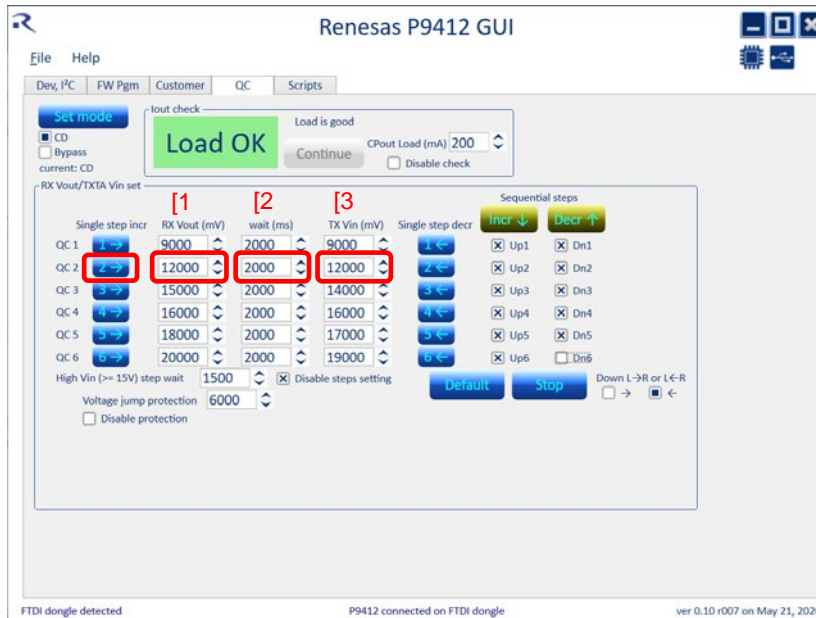


Figure 19. Using the GUI to Manually Increase the RX Vout Voltage

Similarly, the RX VOUT voltage can be manually decreased by using the “Single step decr” buttons. For example, to decrease the RX Vout from 18V to 16V, press the “4←” button. The sequence the GUI follows for decreasing the RX Vout voltage is: 1. Decrease TX Vin voltage from 17V to 16V to lower the power transferred; 2. Wait for the 2 second interval for TX Vin and RX Vout to stabilize; 3. Decrease the RX Vout voltage to 16V.

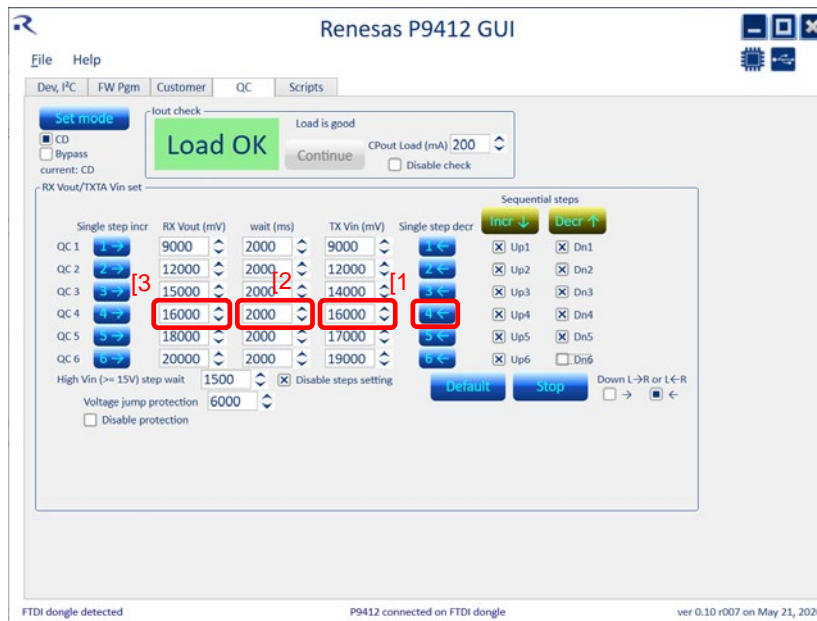


Figure 20. Using the GUI to Manually Decrease the RX Vout Voltage

1.2.4. Using the GUI to Run in TX Mode

Use the following procedure to enter TX mode operation:

1. Remove the P9412 from the TX pad. Remove load from the CPOUT pin.
2. Connect an external 7V Power supply to CPOUT and GND.
3. Turn on the external power supply.

- Select the “Basic 1” Tab and check that the GUI is connected to the P9412 demo board by pressing the “Clear all” and then the “Read 1 time” button on the “Basic 1” tab. Check that the firmware revision and date code are reading correctly and that the Vrect, Vout, and VCPout voltages are all ~0V and that Iout ~0mA.

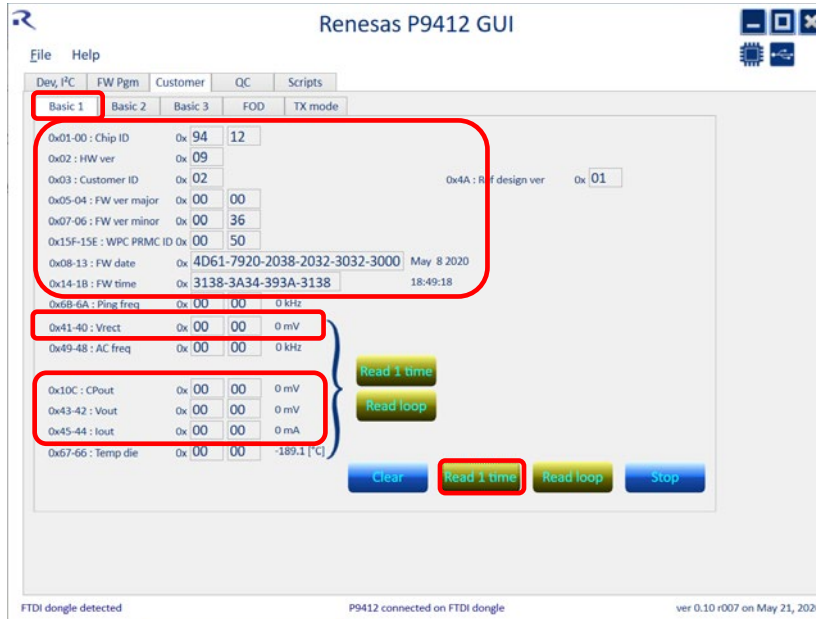


Figure 21. Basic Registers Initial Read Back; Before TX Mode Entry

If the registers look correct then change to the TX mode tab and press the “TX mode” button. Then press the “RD all” button. The Vrect, Vout, and VCPout voltages should be near the external power supply voltage applied to CPOUT as shown. The Iout current should be low since the RX is not placed on the P9412 coil. Verify “TX mode” has been entered.

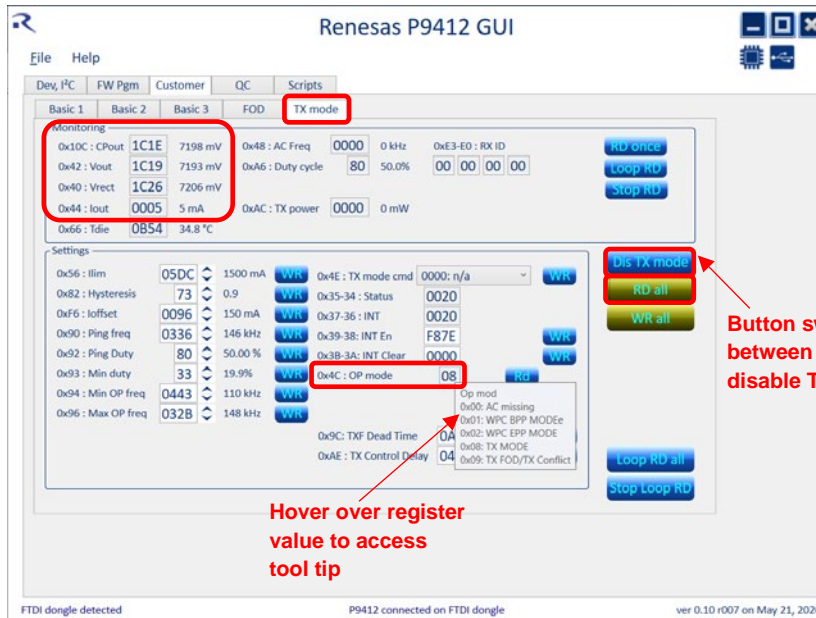


Figure 22. Tx Mode Entry, Without an Rx Placed on the P9412 Coil; Vrect and Vout Voltage Read Back ~7V

- Place the Rx on the P9412’s coil and check the output voltage of the Rx. If a connection is established, there should be ~5V present on the Rx output. Continue to monitor the P9412 Vrect, Vout, Iout, and RX freq registers.

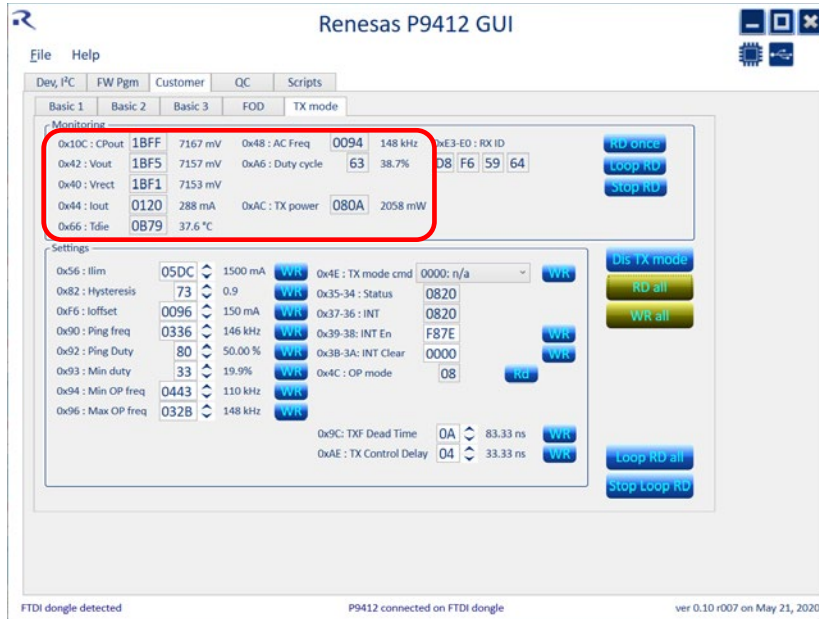


Figure 23. Tx Mode, With an Rx Placed on the P9412 Coil; Use the GUI to Monitor the Status

To exit TX mode operation, either use the “TX mode cmd” register to write a 0x02 to register 0x4F or the “Dis TX mode” button. Verify by reading the OP mode register 0x4C – it should read 0x00 or AC missing (see Figures 24 and 25).

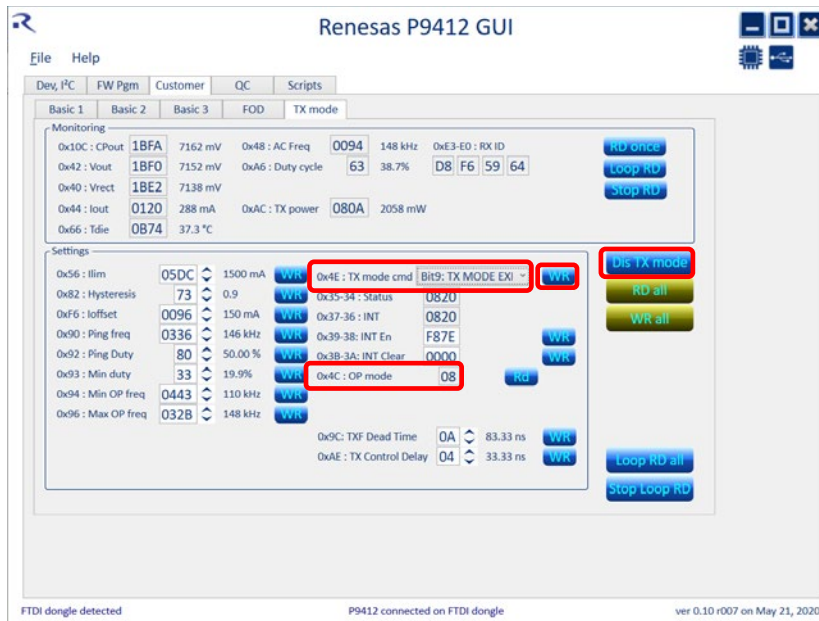


Figure 24. Exit Tx Mode, With an Rx Placed on the P9412 Coil; Use the GUI to Send Exit Command

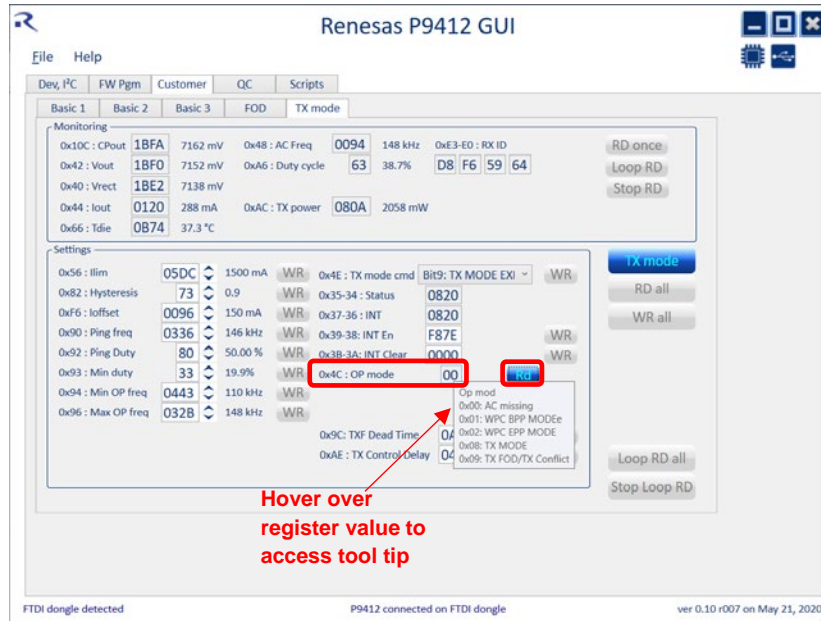


Figure 25. Verify Exit TX Mode; Use the GUI to Read Mode

1.3 I²C Function

The P9412 uses standard I²C slave implementation protocol to communicate with a host Application Processor (AP) or other I²C peripherals. The communication protocol is implemented using 8 bits for data and 16 bits for addresses. The P9412 registers are written using address 0x78 (Write) and read using address 0x79 (Read). The default slave address of the P9412 device is 0x3Ch.

When writing to the P9412, care should be taken to only write to registers marked exclusively as Read/Write (“RW”). Registers marked as Read Only (“R”) should never be attempted to be written to. Likewise, register locations marked “Reserved”, should not be written to. When writing to a RW register that contains a combination of RW fields and reserved fields, a read-modify-write should be performed to the intended bit/field only. All other bits/field, including reserved bits/field should NOT be modified.

Standard Single I²C Read

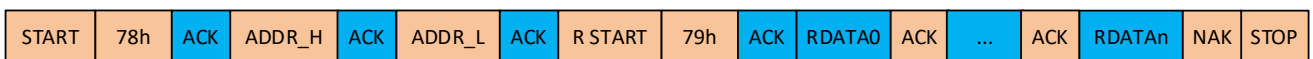


Transferred from master to slave



Transferred from slave to master

Standard Multiple I²C Read



Transferred from master to slave




Transferred from slave to master


Note: Support up to 255 bytes in one block

Figure 26. I²C Read Protocol using P9412

Standard Single I2C Write





 Transferred from master to slave

 Transferred from slave to master

Standard Multiple I2C Write



 Transferred from master to slave

 Transferred from slave to master

Note: Support up to 255 bytes in one block

Figure 27. I2C Write Protocol using P9412

The P9412 TRx device operates in Tx or Rx mode depending on the firmware (FW) loaded into memory by the AP, stored in Multiple-Time Programmable (MTP) registers, or updated in SRAM. Some registers are defined and implemented for Rx mode only, some registers are for Tx mode only, and some registers are common to both operating modes (TRx).

Additionally, the P9412 implements back channel communication following the Frequency Shift Keyed modulation interface (FSK). Details regarding FSK, timing, and encoding scheme can be found in the “FSK Communication”.

An External Power Supply set to 1.8V should be connected to the V1P8_AP test point.

1.3.1. I2C Read back of MTP contents

In order to verify the contents of the P9412 in case the GUI is not available or cannot be used, the following commands can be used to read-back the memory contents. Do not attempt to write to memory this way. Refer to the *P9412 MTP Programming Guide* for details regarding writing to MTP using an AP, and great care should be taken to follow the detailed guide if updating MTP using an AP in a production environment or at production volumes.

Complete the following steps to read MTP:

1. Write I2C Byte (**0x4810, 0x1**).
 - a. Once this is executed, I2C will only read MTP memory section where the FW binary is written. So to go back to regular I2C operation, the AP will need to do a power-cycle.
 - b. MTP memory is write-protected so I2C will only do reads, it cannot write to the memory during this setting.
 - c. To confirm that I2C is reading from MTP, the AP can check the first 4 bytes that should read: (0x00 0x08 0x00 0x20)
2. Read and dump whole FW memory into a binary/text file.
3. Power-cycle the P9412.

2. Registers

The following tables comprise the list of address locations, field names, available operations (R, W, or RW), default values, and functional descriptions of all internally accessible registers contained within the P9412 (FW Minor Revision 0038).

2.1.1. Identification and Revision Registers

2.1.1.1. Chip ID Register, Chip_ID_L (0x00), Chip_ID_H (0x01)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--------------------------|
| 0x00 [7:0] | Chip_ID_L | R | 0x12 | Chip ID low byte |
| 0x01 [7:0] | Chip_ID_H | R | 0x94 | Chip ID high byte |

2.1.1.2. Chip Revision and Font Register, Chip_Rev (0x02)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|------------------------------------|
| 0x02 [7:0] | Chip_Rev | R | 09 | Chip revision. P9412 = 09 = Rev D. |

2.1.1.3. Customer ID Register, Customer ID (0x03)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|------------------------------|
| 0x03 [7:0] | Customer_ID | R | TBD | Read FW customization number |

2.1.1.4. Firmware Major Rev. Registers, FW_Major_Rev_L (0x04), FW_Major_Rev_H (0x05)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x04 [7:0] | FW_Major_Rev_L | R/W | TBD | Major revision of firmware in low byte |
| 0x05 [7:0] | FW_Major_Rev_H | R/W | TBD | Major revision of firmware in high byte |

2.1.1.5. Firmware Minor Rev. Registers, Minor_Rev_L (0x06), FW_Minor_Rev_H (0x07)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x06 [7:0] | FW_Minor_Rev_L | R/W | TBD | Minor revision of firmware in low byte |
| 0x07 [7:0] | FW_Minor_Rev_H | R/W | TBD | Minor revision of firmware in high byte |

2.1.1.6. Firmware Date/Time Registers, FW_Date_Code (0x 08~13), FW_Timer_Code (0x 14~1B)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description | | | | | | | | | | | | | | | | | | |
|-----------------|----------------------|-------|---------------|---|----------|-------|-------|-------|----|---|-------|----|---|-------|----|---|-------|----|-------|-------|----|-------|
| 0x 08 [7:0] | FW_Date_Code [7:0] | R/W | TBD | Date Code of firmware in Flash or MTP Data Encoding Format : Ascii code E.g.) May 8 2020(18:49:18) | | | | | | | | | | | | | | | | | | |
| 0x 09 [7:0] | FW_Date_Code [15:8] | R/W | TBD | | | | | | | | | | | | | | | | | | | |
| 0x 0A [7:0] | FW_Date_Code [23:16] | R/W | TBD | | | | | | | | | | | | | | | | | | | |
| 0x 0B [7:0] | FW_Date_Code [31:24] | R/W | TBD | | | | | | | | | | | | | | | | | | | |
| 0x 0C [7:0] | FW_Date_Code [39:32] | R/W | TBD | | | | | | | | | | | | | | | | | | | |
| 0x 0D [7:0] | FW_Date_Code [47:40] | R/W | TBD | | | | | | | | | | | | | | | | | | | |
| 0x 0E [7:0] | FW_Date_Code [55:48] | R/W | TBD | | | | | | | | | | | | | | | | | | | |
| 0x 0F [7:0] | FW_Date_Code [63:56] | R/W | TBD | | | | | | | | | | | | | | | | | | | |
| | | | | <table border="1"> <thead> <tr> <th>Register</th> <th>Value</th> <th>ASCII</th> </tr> </thead> <tbody> <tr> <td>0x 08</td> <td>4D</td> <td>M</td> </tr> <tr> <td>0x 09</td> <td>61</td> <td>a</td> </tr> <tr> <td>0x 0A</td> <td>79</td> <td>y</td> </tr> <tr> <td>0x 0B</td> <td>20</td> <td>Space</td> </tr> <tr> <td>0x 0C</td> <td>20</td> <td>Space</td> </tr> </tbody> </table> | Register | Value | ASCII | 0x 08 | 4D | M | 0x 09 | 61 | a | 0x 0A | 79 | y | 0x 0B | 20 | Space | 0x 0C | 20 | Space |
| Register | Value | ASCII | | | | | | | | | | | | | | | | | | | | |
| 0x 08 | 4D | M | | | | | | | | | | | | | | | | | | | | |
| 0x 09 | 61 | a | | | | | | | | | | | | | | | | | | | | |
| 0x 0A | 79 | y | | | | | | | | | | | | | | | | | | | | |
| 0x 0B | 20 | Space | | | | | | | | | | | | | | | | | | | | |
| 0x 0C | 20 | Space | | | | | | | | | | | | | | | | | | | | |

| Address and bit | Register Field Name | R/W | Default Value | Function and Description | | |
|-----------------|-----------------------|-----|---------------|---|--------------|--------------|
| 0x 10 [7:0] | FW_Date_Code [71:64] | R/W | TBD | 0x 0D | 38 | 8 |
| 0x 11 [7:0] | FW_Date_Code [79:72] | R/W | TBD | 0x 0E | 20 | Space |
| 0x 12 [7:0] | FW_Date_Code [87:80] | R/W | TBD | 0x 0F | 32 | 2 |
| 0x 13 [7:0] | FW_Date_Code [95:88] | R/W | TBD | 0x 10 | 30 | 0 |
| | | | | 0x 11 | 32 | 2 |
| | | | | 0x 12 | 30 | 0 |
| | | | | 0x 13 | 00 | null |
| 0x 14 [7:0] | FW_Timer_Code [7:0] | R/W | TBD | Time Code of firmware in Flash or MTP | | |
| 0x 15 [7:0] | FW_Timer_Code [13:8] | R/W | TBD | Data Encoding Format : Ascii code E.g.) May 8 2020(18:49:18) | | |
| 0x 16 [7:0] | FW_Timer_Code [23:16] | R/W | TBD | Register | Value | ASCII |
| 0x 17 [7:0] | FW_Timer_Code [15:8] | R/W | TBD | 0x 14 | 31 | 1 |
| 0x 18 [7:0] | FW_Timer_Code [31:24] | R/W | TBD | 0x 15 | 38 | 8 |
| 0x 19 [7:0] | FW_Timer_Code [39:32] | R/W | TBD | 0x 16 | 3A | : |
| 0x 1A [7:0] | FW_Timer_Code [47:40] | R/W | TBD | 0x 17 | 34 | 4 |
| 0x 1B [7:0] | FW_Timer_Code [55:48] | R/W | TBD | 0x 18 | 39 | 9 |
| | | | | 0x 19 | 3A | : |
| | | | | 0x 1A | 31 | 1 |
| | | | | 0x 1B | 38 | 8 |

2.1.1.7. Configuration Major Revision Registers, CFG_Major_Rev (0x1C)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---------------------------------------|
| 0x1C [7:0] | CFG_Major_Rev | R/W | TBD | Major revision of configuration table |

2.1.1.8. Configuration Minor Revision Registers, CFG_Minor_Rev (0x1D)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---------------------------------------|
| 0x1D [7:0] | CFG_Minor_Rev | R/W | TBD | Minor revision of configuration table |

2.1.1.9. Reference Design Version Register, RefDesignVer (0x4A)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x4A [7:0] | RefDesignVer | R | - | Reference design version number. If V1p8_AP is detected at startup then reference design version 2 is determined. 0x01 = Ref Design Version 1: SW Inhibit and GPIO Q factor functions are disabled 0x02 = Ref Design Version 2: SW Inhibit and GPIO Q factor functions are enabled |

2.1.2. Status and Interrupt Registers

2.1.2.1. Status Registers, Status_L (0x34), Status_H (0x35)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x34 [7] | STAT_VOUT | R | 0 | Set when Vout is ON. Cleared when Vout is OFF. Interrupt event is generated on SET and CLR events. |
| 0x34 [6] | STAT_VRECT | R | 0 | Indicates AC power is applied. The flag is set before the Configuration Packet. It is cleared on system reset or when power is removed. Interrupt event is generated on SET event. |
| 0x34 [5] | MODE_CHANGE | R | 0 | No function attached. Refer to "TRX System Operating Mode Register, Sys_Op_Mode (0x4C)". |
| 0x34 [4] | OVER_VOLTAGE | R | 0 | Set if Overvoltage Protection circuit is enabled. Cleared otherwise. Interrupt event is generated on SET and CLR events. |
| 0x34 [3] | OVER_CURR | R | 0 | Set if Overcurrent Protection circuit is enabled. Cleared otherwise. Interrupt event is generated on SET and CLR events. For more information, see Over-Current Status Register, OC_Status (0xB5). |
| 0x34 [2] | OVER_TEMP | R | 0 | Set if Internal temperature exceeds 130°C. Cleared otherwise. Interrupt event is generated on SET and CLR events. |
| 0x34 [1] | Reserved | R | 0 | Reserved |
| 0x34 [0] | ADT Error | R | 0 | Set if ADT Error condition exists, Cleared if error condition doesn't exist. Interrupt event is generated on SET event. ADT= Auxiliary Data Transport |
| 0x35 [7] | Data Received | R | 0 | "1" indicates TX data is received when in RX mode or RX data received when in TX mode. "0" indicates no data is received. |
| 0x35 [6] | CD_ERROR | R | 0 | Set if an error condition occurs while operating in capacitor divider mode. Possible error condition(s): Failure in exiting capacitor divider mode (2:1) because voltage on Vout is too high (above 12V) This bit is cleared together with the corresponding interrupt flag. |
| 0x35 [5] | Reserved | R | 0 | Reserved |
| 0x35 [4] | PropModeStat | R | 0 | No function attached. Refer to Proprietary Mode Status Register, PropModeStatus (0xC8) and Proprietary Mode Error Register, PropErrStatus (0xC9). |
| 0x35 [3] | CD_MODECHANGE | R | 0 | No function attached. See Capacitor Divider Mode Status Register, CDMoSts (0x100). |
| 0x35 [2] | AC Missing Detect | R | 0 | "1" indicates valid AC signal is not present, "0" indicates AC signals exist. Interrupt only generated after power up from battery (external source different from AC power). |
| 0x35 [1] | ADT Received | R | 0 | "1" indicates TX ADT is received, "0" indicates no TX ADT is received. ADT= Auxiliary Data Transport |
| 0x35 [0] | ADT Sent | R | 0 | "1" indicates RX ADT is sent, "0" indicates not all RX ADT has been sent. |

2.1.2.2. Interrupt Registers, INT_L (0x36), INT_H (0x37)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x36 [7] | STAT_VOUT_INT | R | 0 | "1" indicates a pending interrupt for VOUT state change (off to on or on to off). |
| 0x36 [6] | STAT_VRECT_INT | R | 0 | AC power applied and stable interrupt. |
| 0x36 [5] | MODECHANGE_INT | R | 0 | "1" indicates a pending interrupt for Mode Change. Read current mode from System Mode Register |
| 0x36 [4] | OVER_VOLT_INT | R | 0 | "1" indicates a pending interrupt for Over Voltage event. |
| 0x36 [3] | OVER_CURR_INT | R | 0 | "1" indicates a pending interrupt for Over Current event. For more information, see Over-Current Status Register, OC_Status (0xB5). |
| 0x36 [2] | OVER_TEMP_INT | R | 0 | "1" indicates a pending interrupt for Over Temperature event. |
| 0x36 [1] | Reserved | R | 0 | Reserved |
| 0x36 [0] | ADT_Error_INT | R | 0 | "1" indicates a pending interrupt for ADT Error event. |
| 0x37 [7] | Data Received_INT | R | 0 | "1" indicates a pending interrupt for TX data received when in RX mode or RX data received when in TX mode. (No data received state change to data received state). When in RX mode this interrupt is set on any defined header. |

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x37 [6] | CD_ERROR_INT | R | 0 | "1" indicates a pending interrupt for Capacitor Divider Error event |
| 0x37 [5] | Reserved | R | 0 | Reserved |
| 0x37 [4] | PropModeStat_INT | R | 0 | "1" indicates a pending interrupt for proprietary mode entry. Refer to System Mode register (0x4C) and Proprietary Mode Error Register (0xC9). If proprietary mode is enabled, "1" indicates a pending interrupt when the negotiation process is complete. Refer to Proprietary Mode Status Register, PropModeStatus (0xC8) and Proprietary Mode Error Register, PropErrStatus (0xC9). |
| 0x37 [3] | CD_MODECHANGE_INT | R | 0 | "1" indicates a pending interrupt for Capacitor Divider Mode Changed event |
| 0x37 [2] | AC Missing_INT | R | 0 | "1" indicates a pending interrupt that valid AC does not exist, "0" indicates AC signals exist. Interrupt only generated after power up from battery (external source different from AC power). |
| 0x37 [1] | ADT Received_INT | R | 0 | "1" indicates a pending interrupt for TX ADT Received. (No ADT received state change to ADT received state). |
| 0x37 [0] | ADT Sent_INT | R | 0 | "1" indicates a pending interrupt for RX ADT Sent. (No ADT sent state change to ADT sent state). |

2.1.2.3. Interrupt Enable Registers, INT_Enable_L (0x38), INT_Enable_H (0x39)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x38 [7] | STAT_VOUT_EN | RW | 1 | VOUT state change interrupt enable. Default value is "1". AP writes "0" to disable the interrupt. |
| 0x38 [6] | STAT_VRECT_EN | RW | 1 | AC power applied and stable interrupt enable. Default value is "1". |
| 0x38 [5] | MODECHANGE_EN | RW | 1 | Mode Changed interrupt enable. Default value is "1". AP writes "0" to disable the interrupt. |
| 0x38 [4] | OVER_VOLT_EN | RW | 1 | Overvoltage condition ON/OFF interrupt enable. Default value is "1" |
| 0x38 [3] | OVER_CURR_EN | RW | 1 | Overcurrent condition ON/OFF interrupt enable. Default value is "1" |
| 0x38 [2] | OVER_TEMP_EN | RW | 1 | Over-temperature condition ON/OFF interrupt enable. Default value is "1" |
| 0x38 [1] | Reserved | R | 0 | Reserved |
| 0x38 [0] | ADT_Error_EN | RW | 1 | ADT Error interrupt enable. Default value is "1". AP writes "0" to disable the interrupt |
| 0x39 [7] | Data Received_EN | RW | 1 | TX Data Received interrupt enable. Default value is "1". AP writes "0" to disable the interrupt |
| 0x39 [6] | CD_ERROR_EN | RW | 1 | Capacitor Divider Mode Changed interrupt enable. Default value is "1". AP writes "0" to disable the interrupt. |
| 0x39 [5] | Reserved | R | 0 | Reserved |
| 0x39 [4] | PropModeStat_EN | R/W | 1 | Proprietary Mode Status interrupt enable. Default value is "1". AP writes "0" to disable the interrupt. |
| 0x39 [3] | CD_MODECHANGE_EN | RW | 1 | Capacitor Divider Mode Changed interrupt enable. Default value is "1". AP writes "0" to disable the interrupt. |
| 0x39 [2] | AC Missing_EN | R/W | 0 | AP writes "1" is to enable the interrupt from the Interrupt Registers' corresponding bit, "0" is to disable the interrupt. Interrupt only generated after power up from battery (external source different from AC power). |
| 0x39 [1] | ADT Received_EN | RW | 1 | Tx ADT Received interrupt enable. Default value is "1". AP writes "0" to disable the interrupt |
| 0x39 [0] | ADT Sent_EN | RW | 1 | Rx ADT Sent interrupt enable. Default value is "1". AP writes "0" to disable the interrupt |

2.1.2.4. Interrupt Clear Registers, INT_Clear_L (0x3A), INT_Clear_H (0x3B)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x3A [7] | STAT_VOUT_CLR | W | 0 | VOUT state change interrupt flag clear. AP writes "1" to clear the corresponding Interrupt Registers' bit and this bit is self-cleared to "0" (by MCU) afterwards. |
| 0x3A [6] | STAT_VRECT_CLR | W | 0 | AC power applied and stable interrupt flag clear |
| 0x3A [5] | MODECHANGE_CLR | W | 0 | Mode Changed interrupt flag clear |
| 0x3A [4] | OVER_VOLT_CLR | W | 0 | Overvoltage condition ON/OFF interrupt flag clear |
| 0x3A [3] | OVER_CURR_CLR | W | 0 | Overcurrent condition ON/OFF interrupt flag clear |
| 0x3A [2] | OVER_TEMP_CLR | W | 0 | Over-temperature condition ON/OFF interrupt flag clear |
| 0x3A [1] | Reserved | W | 0 | Reserved |
| 0x3A [0] | ADT_Error_CLR | W | 0 | ADT Error interrupt flag clear. AP writes "1" to clear the corresponding Interrupt Registers' bit and this bit is self-cleared to "0" (by MCU) afterwards. |
| 0x3B [7] | Data Received_CLR | W | 0 | Tx data received interrupt flag clear |
| 0x3B [6] | CD_ERROR_CLR | W | 0 | Capacitor Divider Error interrupt flag clear. |
| 0x3B [5] | Reserved | W | 0 | Reserved |
| 0x3B [4] | PropModeStat_CLR | W | 0 | Proprietary Mode Status interrupt flag clear. |
| 0x3B [3] | CD_MODECHANGE_CLR | W | 0 | Capacitor Divider Mode Changed interrupt flag clear. |
| 0x3B [2] | AC Missing_CLR | W | 0 | AC Missing interrupt flag clear. Interrupt only generated after power up from battery (external source different from AC power). |
| 0x3B [1] | ADT Received_CLR | W | 0 | Tx ADT received interrupt flag clear. |
| 0x3B [0] | ADT Sent_CLR | W | 0 | Rx ADT sent interrupt flag clear. |

Set bits in this register to clear corresponding interrupt flags. The register is self-cleared. Writing to this register does not invoke the clear by itself. The user must set BIT 5 in System Command Register (0x4E) to trigger the interrupt clear event (see System Command Register, SYS_CMND_L (0x4E), SYS_CMND_H (0x4F)).

2.1.2.5. TRX System Operating Mode Register, Sys_Op_Mode (0x4C)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x4C [7:4] | Reserved | R | 0 | Reserved |
| 0x4C [3:0] | Sys_Op_Mode | R | 0000 | 0000 = AC Missing 0001 = WPC Basic Protocol 0010 = WPC Extended Protocol 0011 = Renesas Proprietary Protocol 1000 = TX Mode 1001 = TX FOD (Stop power transfer) / TX Conflict (Stop ping) |

This register is cleared at entry to AC Missing State (DC power only), and will read back 0x0. This is the state when power is provided by the user to Vrect, Vout, or CPout and no AC signal is detected on the rectifier inputs. For Capacitor Divider mode status, see Capacitor Divider Mode Status Register, CDMoSts (0x100). For Tx mode status see TX Status Registers, Status_L (0x34), Status_H (0x35).

2.1.2.6. Over-Current Status Register, OC_Status (0xB5)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0xB5 [7:5] | Reserved | R | 0 | Reserved |
| 0xB5 [4] | CDIV_OC | R | 0 | "1" indicates an Over-Current condition in the Cap Divider circuit block. Provides additional information on cause of the OC status or interrupt bit being set. Bit is cleared when OC_INT is cleared. |
| 0xB5 [3:1] | Reserved | R | 0 | Reserved |
| 0xB5 [0] | MLDO_OC | R | 0 | "1" indicates an Over-Current condition in the Main LDO circuit block. Provides additional information on cause of the OC status or interrupt bit being set. Bit is cleared when OC_INT is cleared. |

2.1.3. Battery Status and Power Transfer Registers

2.1.3.1. Charge Status Register, CHG_Status (0x3E)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x3E [7:0] | CHG_Status | R/W | 0x00 | <p>The AP writes this register with the value intended to be sent as payload to the Charge Status Packet as defined below. The FW does not verify or modify the value in any way.</p> <p>0x0 = Reserved 0x1 = Charge status packet sent with parameter = 1 (1%) 0x2: Charge status packet send with parameter = 2 (2%) 0x64 = Charge status packet send with parameter = 100 (100%) 0x65 ~ 0xFE = Reserved 0xFF = No Battery Charge Device or Not Providing Charge Status Packet</p> |

2.1.3.2. End of Power Transfer Register, EPT_Code (0x3F)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x3F [7:0] | EPT_Code | R/W | 0x00 | <p>The AP writes this register with the value intended to be sent as payload to the End of Power Transfer Packet as defined below. The FW does not verify or modify the value in any way.</p> <p>0 = WPC mode, unknown EPT should be sent. 1 = WPC mode, End of Charge EPT packet should be sent. 2 = WPC mode, Internal Fault EPTpacket should be sent. 3 = WPC mode, Over Temperature EPTpacket should be sent. 4 = WPC mode, Over Voltage EPT packet should be sent. 5 = WPC mode, Over Current EPTpacket should be sent. 6 = WPC mode, Battery Failure EPTpacket should be sent. 7 = WPC mode, Reconfiguration EPT packet should be sent. 8 = WPC mode, No Response EPT packet should be sent. 9 ~ 254 = Reserved</p> |

2.1.4. Operation Parameters Registers

Note: See *Vrect Control Registers*, *Capacitor Divider Registers*, and *TX Mode Registers*.

2.1.4.1. Vout Set Register, Vout_Set_L (0x6C), Vout_Set_H (0x6D)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x6C [7:0] | Vout_Set_L [7:0] | RW | 0xF4 | <p>8 LSB of output voltage setting of the main LDO in 10mV units. Firmware increments in 40mV steps. Vout_Set range is from 3.52V to 20V. For EPP mode operation, it is recommended to increase Vout_Set after a connection is established.</p> |
| 0x6D [7:0] | Vout_Set_H [15:8] | RW | 0x01 | <p>8 MSB of output voltage setting of the main LDO in 10mV units. Firmware increments in 40mV steps. Default value: 0x1F4 = 5V. <i>Example: To set Vout to 5.120V, write 0x200 (512 in decimal).</i></p> |

Note: The Vout_Set registers at address 0x6C, applies to FW37 and above. GUI version v0.10r007 or higher is compatible with this register address as well as the earlier Vout_Set register address, 0x3C.

2.1.4.2. TRX Vout Voltage Registers, Vout_L (0x42), Vout_H (0x43)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x42 [7:0] | Vout_L [7:0] | R | - | 8 LSB of current main LDO Vout Voltage value. The AP may read this register to get current Vout level in mV. |
| 0x43 [7:0] | Vout_H [15:8] | R | - | 8 MSB of current main LDO Vout Voltage value. <i>Example: If Vout = 0x1388h => 5000 = 5000mV = 5V</i> |

2.1.4.3. Iout Limit Set Register (0x3D)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x3D [7:0] | ILim | RW | 0x11 | Set main LDO current limit. Current Limit = 100mA*(ILim+1) The maximum value of this register is 0x12 (18 in decimal) corresponding to a value of 1.9A |

2.1.4.4. TRX Iout / Iin Value Registers, Iout / Iiin_L (0x44), Iout / Iin_H (0x45)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x44 [7:0] | Iout / Iiin_L [7:0] | R | - | 8 LSB of RX Iout / TX Iin current value. The AP may read this register to get current Iout / Iin level in mA. |
| 0x45 [7:0] | Iout / Iin_H [15:8] | R | - | 8 MSB of Iout / Iin current value. <i>Example: 0x3B6h => 950 950mA = 0.95A</i> |

2.1.4.5. TRX Vrect Voltage Registers, Vrect_L (0x40), Vrect_H (0x41)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x40 [7:0] | Vrect_L [7:0] | R | - | 8 LSB of current Vrect Voltage value. The AP may read this register to get current Vrect level in mV. |
| 0x41 [7:0] | Vrect_H [15:8] | R | - | 8 MSB of current Vrect Voltage value. <i>Example: If Vrect = 0x1828h => 6184 = 6184mV = 6.184V</i> |

2.1.4.6. TRX Die Temperature Registers, DieTemp_L (0x46), DieTemp_H (0x47)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x46 [7:0] | DieTemp_L [7:0] | R | - | 8 LSB of current Die Temperature value. The AP may read this register to get current die temperature in degrees C. |
| 0x47 [7:0] | DieTemp_H [15:8] | R | - | 8 MSB of I current Die Temperature value. <i>Example: 0x0037h => 55 = 55C</i> |

2.1.4.7. TRX AC Frequency Registers, AC_Freq_L (0x48), AC_Freq_H (0x49)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x48 [7:0] | AC_Freq_L [7:0] | R | - | 8 LSB of current AC frequency value. The AP may read this register to get current frequency of the AC signal in kHz. |
| 0x49 [7:0] | AC_Freq_H [15:8] | R | - | 8 MSB of I current AC frequency value. <i>Example: 0x0087h => 135 = 135kHz</i> |

2.1.5. Command and Communication Registers

2.1.5.1. System Command Register, SYS_CMND_L (0x4E), SYS_CMND_H (0x4F)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x4E [7] | EPP RENEGOTIATE | RW | 0 | Initiate Renegotiation Request in EPP mode. The AP must configure the new requested parameters before setting this bit. |
| 0x4E [6] | CD_CMND | RW | 0 | Initiate capacitor divider command. To request a mode change set the desired mode in the "Capacitor Divider Mode Request Register, CDMoDeReq (0x101)". |
| 0x4E [5] | CLR_INT | RW | 0 | If AP sets this bit to "1" then MFC-IC MCU clears the interrupt corresponding to the bit(s) which has a value of "1" in Interrupt Clear Registers and. then MFC's MCU sets the bit(s) in Interrupt Clear Registers to "0" |
| 0x4E [4] | SEND_CSP | RW | 0 | If AP sets this bit to "1" then MFC-IC MCU sends the Charge Status packet (defined in the Battery Charge Status Register) to TX and then MFC's MCU sets this bit to "0" |
| 0x4E [3] | SEND_EPT | RW | 0 | If AP sets this bit to "1" then MFC-IC MCU sends the End of Power packet (defined in the End of Power Transfer Register) to TX and then MFC's MCU sets this bit to "0" |
| 0x4E [2] | CT_CMND | RW | 0 | Execute the config table command set with register 0xCA. See Config Table Command Register, Config_CMND (0xCA) and TRX Header Register (PropPkt Send), TRX_Header_Out (0x50). |
| 0x4E [1] | LDO_TGL | RW | 0 | If AP sets this bit to "1" then MFC-IC MCU toggles LDO output once (from on to off, or from off to on), and then MFC's MCU sets this bit to "0". The result can be read from the System Status Register. Only toggles when Cap Divider is disabled. |
| 0x4E [0] | SEND_PPP | RW | 0 | If AP sets this bit to "1" then MFC-IC MCU sends the Proprietary Packet (defined in the Proprietary Packet Registers) to TX and then MFC's MCU sets this bit to "0". |
| 0x4F [7:3] | Reserved | R | 0 | Reserved |
| 0x4F [2] | SEND_ADT | RW | 0 | If AP sets this bit to "1" then MFC-IC MCU sends the ADT (defined in the Communication Channel Registers) to TX and then MFC's MCU sets this bit to "0". |
| 0x4F [1] | PROPPWRREQ | RW | 0 | Request power from TX based on PropReqPwr (0xC5). |
| 0x4F [0] | PROPMODEEN | RW | 0 | Enable Proprietary Mode (TX power capability and authentication). |

1. The AP sets any of the bits in this register to initiate the corresponding process. The register is self-cleared when the command is read by the FW and the process loaded in the execution queue. For TX mode commands see System TX Command Register, TX_CMND (0x4D) and TX Mode System Command Register, TxSysCmnd_L (0x4E), TxSysCmnd_H (0x4F).

2.1.5.2. Config Table Command Register, Config_CMND (0xCA)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0xCA [7:0] | Config_CMND | RW | 0x0 | Executes config table command once CT_CMND bit in System Command Register (0x4E) is set. 1: Get the default config table – available at address TBD 2: Get the custom config table – available at address TBD (Work In Progress) 3: Save the custom config table (Work In Progress) |

The **Communication Channel** is designed to exchange data between the Tx and Rx. The Com Channel supports Proprietary Packets and ADT Communication Messages as payload. From the user point of view there should not be any difference related to the direction of the message, except for the speed.

The **Proprietary Packet** follows the WPC specification for its form. The FW does not check the content of the packet, so the AP may load any header and data, including the capability to simulate packets already defined by WPC for a special function. The AP's needed actions to send a Proprietary Packet are described below:

1. Load the Proprietary Packet in the registers specified in TRX Header Register (PropPkt Send), TRX_Header_Out (0x50) and TRX Data Value 2~5 Reg. (PropPkt Send), TRX_Data_Value2_5_Out (0x52, 0x53, 0x54, 0x55), header first, followed by the packet data. The header and packet data follow WPC spec. The checksum is not needed – the FW will calculate it;
2. Set System Command register (0x4E [0]) to start the send process;
3. Check the Pending Pkts register (0x149 [0]) to indicate process completion;

Steps needed to be executed by the AP when Proprietary Packet is received:

1. Receive System interrupt, Data Received_INT (0x37 [9]);
2. Read the byte in the TRX Header Register (0x58). This is the proprietary packet header. Decode the packet header;
3. Read the two bytes from the TRX Data Value 1~2 registers (0x59 and 0x5A) to complete the packet. Alternatively, in step2, the AP may read all three bytes and execute step 3 only if needed;
4. Clear the Data Received_INT interrupt. This step is needed even if the interrupt is not enabled and the packet received event was recognized by polling the System Status register (0x34). No new packet will be accepted otherwise.

The **Communication Message** is transferred by a series of packets generated and handled by the FW state machine. The payload data is loaded/read to/from the communication data buffer, 2K bytes long. The steps needed to be executed by the AP to send a Communication message are similar to these in the Proprietary Packet:

1. Verify the Com Channel is not in use (Com Channel Status Register: receive busy 0x148 [1] and send busy 0x148 [0] are cleared);
2. Load the message in the ADT Data Buffer (0x0800);
3. The Com Channel Send Size Register (0x140) must be written with the size of the message in bytes (1 to 2K);
4. Set System Command register, SEND_ADT (0x4F [1]) to start the send process;
5. Wait for the ADT_Sent Interrupt (0x37 [0]) to indicate process completion and clear the interrupt. If there is a communication error, an ADT_Error_interrupt (0x36 [0]) will be set. ADT error codes can be read at the ADT Error Code register (0x14D [4;0])

Steps needed to be executed by the AP when Com Message is received:

1. Receive ADT Received interrupt (0x37 [1]);
2. Read the Com Channel Received Size register (0x144) to find the message size;
3. Read the message size number of bytes from the ADT Data buffer registers (0x0800);
4. Clear the ADT_Received interrupt;

The AP may monitor the data transfer progress by checking periodically index registers: Com Channel Send Index register (0x142) and the Com Channel Receive Index register (0x146) and verify the indexes are changing. The AP may also implement a Time Out function. It is possible the required time to send a message is

unusually longer if power level needs to be adjusted frequently. In any case, the AP may want to interrupt the Communication process at any time.

2.1.5.3. TRX Header Register (PropPkt Send), TRX_Header_Out (0x50)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--|-------------------------|--------------|---------------|--|---------|--------------|--------------|-------------------|--|--|------|-----------------|---|------|--------------------|---|---|--|--|------|------------------------|---|------|---------------|---|------|----------------|---|------|-------------------------|---|-----------------------------|--|--|------|--------------------|---|------|---------------|---|------|----------------|---|------|---------------|---|--|--|--|------|-------------|---|------|-------------|---|------|-------------|---|------|-------------|---|------|-------------|---|------|-------------|---|------|-------------|---|------|-------------|---|------|-------------|---|------|-------------|---|------|-------------|----|------|-------------|----|------|-------------|----|
| 0x50 [7:0] RX Mode | RX Header Out | RW | 0x00 | <p>0x02: end power transfer; 0x05: for charge status packet; 0x18 ~ 0xE2: proprietary packet; The rest values are reserved.</p> <table border="1"> <thead> <tr> <th>Header'</th> <th>Packet Types</th> <th>Message Size</th> </tr> </thead> <tbody> <tr> <td colspan="3"><i>ping phase</i></td> </tr> <tr> <td>0x01</td> <td>Signal Strength</td> <td>1</td> </tr> <tr> <td>0x02</td> <td>End Power Transfer</td> <td>1</td> </tr> <tr> <td colspan="3"><i>identification & configuration phase</i></td> </tr> <tr> <td>0x06</td> <td>Power Control Hold-off</td> <td>1</td> </tr> <tr> <td>0x51</td> <td>Configuration</td> <td>5</td> </tr> <tr> <td>0x71</td> <td>Identification</td> <td>7</td> </tr> <tr> <td>0x81</td> <td>Extended Identification</td> <td>8</td> </tr> <tr> <td colspan="3"><i>power transfer phase</i></td> </tr> <tr> <td>0x02</td> <td>End Power Transfer</td> <td>1</td> </tr> <tr> <td>0x03</td> <td>Control Error</td> <td>1</td> </tr> <tr> <td>0x04</td> <td>Received Power</td> <td>1</td> </tr> <tr> <td>0x05</td> <td>Charge Status</td> <td>1</td> </tr> <tr> <td colspan="3"><i>identification & configuration / power transfer phase</i></td> </tr> <tr> <td>0x18</td> <td>Proprietary</td> <td>1</td> </tr> <tr> <td>0x19</td> <td>Proprietary</td> <td>1</td> </tr> <tr> <td>0x28</td> <td>Proprietary</td> <td>2</td> </tr> <tr> <td>0x29</td> <td>Proprietary</td> <td>2</td> </tr> <tr> <td>0x38</td> <td>Proprietary</td> <td>3</td> </tr> <tr> <td>0x48</td> <td>Proprietary</td> <td>4</td> </tr> <tr> <td>0x58</td> <td>Proprietary</td> <td>5</td> </tr> <tr> <td>0x68</td> <td>Proprietary</td> <td>6</td> </tr> <tr> <td>0x78</td> <td>Proprietary</td> <td>7</td> </tr> <tr> <td>0x84</td> <td>Proprietary</td> <td>8</td> </tr> <tr> <td>0xA4</td> <td>Proprietary</td> <td>12</td> </tr> <tr> <td>0xC4</td> <td>Proprietary</td> <td>16</td> </tr> <tr> <td>0xE2</td> <td>Proprietary</td> <td>20</td> </tr> </tbody> </table> <p>*Header values not listed in this table correspond to reserved Packet types</p> | Header' | Packet Types | Message Size | <i>ping phase</i> | | | 0x01 | Signal Strength | 1 | 0x02 | End Power Transfer | 1 | <i>identification & configuration phase</i> | | | 0x06 | Power Control Hold-off | 1 | 0x51 | Configuration | 5 | 0x71 | Identification | 7 | 0x81 | Extended Identification | 8 | <i>power transfer phase</i> | | | 0x02 | End Power Transfer | 1 | 0x03 | Control Error | 1 | 0x04 | Received Power | 1 | 0x05 | Charge Status | 1 | <i>identification & configuration / power transfer phase</i> | | | 0x18 | Proprietary | 1 | 0x19 | Proprietary | 1 | 0x28 | Proprietary | 2 | 0x29 | Proprietary | 2 | 0x38 | Proprietary | 3 | 0x48 | Proprietary | 4 | 0x58 | Proprietary | 5 | 0x68 | Proprietary | 6 | 0x78 | Proprietary | 7 | 0x84 | Proprietary | 8 | 0xA4 | Proprietary | 12 | 0xC4 | Proprietary | 16 | 0xE2 | Proprietary | 20 |
| Header' | Packet Types | Message Size | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <i>ping phase</i> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x01 | Signal Strength | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x02 | End Power Transfer | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <i>identification & configuration phase</i> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x06 | Power Control Hold-off | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x51 | Configuration | 5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x71 | Identification | 7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x81 | Extended Identification | 8 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <i>power transfer phase</i> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x02 | End Power Transfer | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x03 | Control Error | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x04 | Received Power | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x05 | Charge Status | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <i>identification & configuration / power transfer phase</i> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x18 | Proprietary | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x19 | Proprietary | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x28 | Proprietary | 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x29 | Proprietary | 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x38 | Proprietary | 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x48 | Proprietary | 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x58 | Proprietary | 5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x68 | Proprietary | 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x78 | Proprietary | 7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x84 | Proprietary | 8 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0xA4 | Proprietary | 12 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0xC4 | Proprietary | 16 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0xE2 | Proprietary | 20 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x50 [7:0] TX Mode | TX Header Out | RW | 0x00 | <p>0x01 = TX-ID 0x02~0xFF = Reserved</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Note: The contents of these registers is based on the most recently written value (for example, if a PPP packet is sent, the values stored in these registers will match those written by the P9412 FW or the AP, unless they are over-written (updated) or power is cycled).

This register's functionality depends on the operation mode (TX or RX Mode).

2.1.5.4. TRX Data Value1 Register (PropPkt Send), TRX_Data_VALUE1_Out (0x51)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------------|---------------------|-----|---------------|--|
| 0x51 [7:0] RX Mode | RX Data_Value1 Out | RW | 0x00 | 0x00 = Unknown 0x01 = Request_TX-ID 0x05 = Charge Status 0x06 = AFC_SET 0x07 = AFC_Debounce 0x08 = S-ID Tag 0x09 = S-ID Token 0x0A = TX Standby 0x0B = LED Control * Corresponding RX Data_Value : LED Enable 0x00, LED Disable 0xFF 0x0C = Request AFC_TX * Corresponding RX Data_Value : 0x00 0x0D = Cooling Control * Corresponding RX Data_Value : ON 0x00, OFF 0xFF 0x0F = Battery SOC 0x18 = Power Hold 0x10-0xFF = Reserved (unless listed) |
| 0x51 [7:0] TX Mode | TX Data_Value1 Out | RW | 0x00 | Bit [7:0] of TX Data_Value |

Note: This register's functionality depends on the operation mode (TX or RX Mode).

2.1.5.5. TRX Data Value 2~5 Reg. (PropPkt Send), TRX_Data_Value2_5_Out (0x52, 0x53, 0x54, 0x55)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------------|----------------------------|-----|---------------|------------------------------|
| 0x52 [7:0] RX Mode | RX Data_Value2 Out [7:0] | RW | 0x0 | Bit [7:0] of RX Data_Value |
| 0x53 [7:0] RX Mode | RX Data_Value3 Out [15:8] | RW | 0x0 | Bit [15:8] of RX Data_Value |
| 0x54 [7:0] RX Mode | RX Data_Value4 Out [23:16] | RW | 0x0 | Bit [23:16] of RX Data_Value |
| 0x55 [7:0] RX Mode | RX Data_Value5 Out [31:24] | RW | 0x0 | Bit [31:24] of RX Data_Value |
| 0x52 [7:0] TX Mode | TX Data_Value2 Out [7:0] | RW | 0x0 | Bit [7:0] of TX Data_Value |
| 0x53 [7:0] TX Mode | TX Data_Value3 Out [15:8] | RW | 0x0 | Bit [15:8] of TX Data_Value |

Note: This register's functionality depends on the operation mode (TX or RX Mode).

2.1.5.6. TRX Header Register (PropPkt Received), TRX_Header_In (0x58)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------------|---------------------|-----|---------------|--|
| 0x58 [7:0] RX Mode | TX Header In | R | 0x00 | 0x00 = Unknown 0x01 = TX-ID 0x02 = AFC_TX 0x03 = ACK 0x04 = NAK 0x05 = Charge Stop 0x18 = Power Hold 0x06-0xFF = Reserved (unless listed) |
| 0x58 [7:0] TX Mode | RX Header In | R | 0x00 | 0x18 = Proprietary Packet Header 0x28 = Proprietary Packet Header (TX_ID Request) |

Note: This register's functionality depends on the operation mode (TX or RX Mode).

In TX mode, every time a Proprietary Packet 0x28 0x01 0x00 (TX_ID request) is received – for phone to phone, an interrupt is set and TX_ID FSK response is not automatically sent.

2.1.5.7. TRX Data Value 1~2 Register (PropPkt Received), TRX_Data_Value1_2_In (0x59, 0x5A)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------------|--------------------------|-----|---------------|--|
| 0x59 [7:0] RX Mode | TX Data_Value1 In [7:0] | R | 0x0 | Bit [7:0] of TX Data_Value |
| 0x5A [7:0] RX Mode | TX Data_Value2 In [15:8] | R | 0x0 | Bit [15:8] of TX Data_Value |
| 0x59 [7:0] TX Mode | RX Data_Value1 In [7:0] | R | 0x0 | Bit [7:0] of RX Data_Value 0x01: TX_ID Request |
| 0x5A [7:0] TX Mode | RX Data_Value2 In [15:8] | R | 0x0 | Bit [15:8] of RX Data_Value 0x00: TX_ID Request |

Note: This register's functionality depends on the operation mode (TX or RX Mode).

2.1.5.8. Com Channel Send Size Register, CC_Send_Size_L (0x140), CC_Send_Size_H (0x141)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x140 [7:0] | ccWrSize [7:0] | RW | 0x00 | 8 LSB of Com Channel Send Size of the message to be sent. Maximum value is 2K. |
| 0x141 [7:0] | ccWrSize [15:8] | RW | 0x00 | 8 MSB of Com Channel Send Size. |

2.1.5.9. Com Channel Send Index Register, CC_Send_Index_L (0x142), CC_Send_Index_H (0x143)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x142 [7:0] | ccWrIndx [7:0] | R | 0x00 | 8 LSB of Current index of the message being transmitted. |
| 0x143 [7:0] | ccWrIndx [15:8] | R | 0x00 | 8 MSB of Current index of the message being transmitted. |

2.1.5.10. Com Channel Receive Size Register, CC_Recv_Size_L (0x144), CC_Recv_Size_H (0x145)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x144 [7:0] | ccRdSize [7:0] | RW | 0x00 | 8 LSB of Com Channel Receive Size of the message to be received. Maximum value is 2044. |
| 0x145 [7:0] | ccRdSize [15:8] | RW | 0x00 | 8 MSB of Com Channel Send Size. |

2.1.5.11. Com Channel Receive Index Reg., CC_Recv_Index_L (0x146), CC_Recv_Index_H (0x147)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x146 [7:0] | ccRdIndx [7:0] | R | 0x00 | 8 LSB of Current index of the message being received. |
| 0x147 [7:0] | ccRdIndx [15:8] | R | 0x00 | 8. MSB of Current index of the message being received. |

2.1.5.12. Com Channel Status Register, CC_Status (0x148)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x148 [7:2] | Reserved | R | 0x0 | Reserved |
| 0x148 [1] | ADT_Rcv_Busy | R | 0x0 | "1" indicates that the Com Channel is busy with ADT receive |
| 0x148 [0] | ADT_Send_Busy | R | 0x0 | "1" indicates that the Com Channel is busy with ADT send |

2.1.5.13. Pending Packets Register, Pend_Pkts (0x149)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x149 [7:6] | Reserved | R | 0x0 | Reserved |
| 0x149 [5] | Pend_DSR | R | 0x0 | "1" indicates Data Set Ready packets are pending to be sent |
| 0x149 [4] | Pend_ADT | R | 0x0 | "1" indicates ADT packets are pending to be sent |
| 0x149 [3] | Pend_Reneg | R | 0x0 | "1" indicates Renegotiation packets are pending to be sent |
| 0x149 [2] | Pend_Charge | R | 0x0 | "1" indicates Charge status packets are pending to be sent |
| 0x149 [1] | Pend_AckPkt | R | 0x0 | "1" indicates ACK is pending to be sent |
| 0x149 [0] | Pend_Prop | R | 0x0 | "1" indicates Proprietary packets are pending to be sent |

2.1.5.14. ADT Packet Time Out Register, ADT_Timeout_PKT (0x150)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|-----------------------|-----|---------------|--|
| 0x150 [7:0] | ADT_Timeout_PKT [7:0] | R/W | 0x00 | ADT Timeout for sending a single packet. 00: Disabled, 01: 50ms, FF: 12750 ms |

2.1.5.15. ADT Stream Time Out Register, ADT_Timeout_STR (0x151)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|-----------------------|-----|---------------|---|
| 0x151 [7:0] | ADT_Timeout_STR [7:0] | R/W | 0x00 | ADT Timeout for sending a full message stream. 00: Disabled, 01: 500ms, FF: 127,500 ms |

2.1.5.16. ADT Error Code Register, ADT_Error_Code (0x14D)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|-------------------------------------|
| 0x14D [7:5] | Reserved | R/W | 0x00 | Reserved |
| 0x14D [4] | ADT RCVD OVFLW | R/W | 0 | "1" indicates Received ADT OverFlow |
| 0x14D [3] | ADT STR TO | R/W | 0 | "1" indicates ADT Stream TimeOut |
| 0x14D [2] | ADT PKT TO | R/W | 0 | "1" indicates ADT Packet TimeOut |
| 0x14D [1] | ADT BUSY ERR | R/W | 0 | "1" indicates ADT Busy Error |
| 0x14D [0] | ADT FAULT | R/W | 0 | "1" indicates ADT Fault |

2.1.5.17. ADT Buffer Registers, (0x0800 ~ 0x0FFF)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x0800 [7:3] | ADT Type | R/W | | End: 0x00; General Purpose: 0x01; Authentication: 0x02; Reset: 0x05 |
| 0x0800 [2:0] | ADT Message Size | R/W | | MSB of the ADT message size in bytes |
| 0x0801 [7:0] | ADT Message Size | R/W | | LSB of the ADT message size in bytes |
| 0x0802 [7:0] | ADT Parameters | R/W | | This register is always "0" |
| 0x0803 [7:0] | ADT Parameters | R/W | | This register is always "0" |
| 0x0804~0x0FFF | ADT Parameters | R/W | | ADT Message Data |

Examples of ADT message buffer:

1-byte message:

| Address | Value |
|---------|-------|
| 0x0800 | 0x10 |
| 0x0801 | 0x01 |
| 0x0802 | 0x00 |
| 0x0803 | 0x00 |
| 0x0804 | Data0 |

100-byte message:

| Address | Value |
|---------|-------|
| 0x0800 | 0x10 |
| 0x0801 | 0x64 |
| 0x0802 | 0x00 |
| 0x0803 | 0x00 |
| 0x0804 | Data0 |
| ... | ... |

2044-byte message:

| Address | Value |
|---------|----------|
| 0x0800 | 0x17 |
| 0x0801 | 0xFB |
| 0x0802 | 0x00 |
| 0x0803 | 0x00 |
| 0x0804 | Data0 |
| ... | ... |
| 0x0FFF | Data2043 |

ADT Reset message:

| Address | Value |
|---------|-------|
| 0x0800 | 0x28 |
| 0x0801 | 0x00 |
| 0x0802 | 0x00 |
| 0x0803 | 0x00 |

Note: TX Data Command and Value is transmitted in a packet format from TX to RX via a proprietary back channel, using FSK (less than 1% positive frequency deviation) modulation at the end (several ms later) of Control Error Packet, Received Power Packet or Charge Status Packet. The frequency deviation is calculated using the following formula:

$$F_m = 60000 / ((60000/F) - 3) \text{ (kHz)} - \text{Equation 1}$$

where, F_{mod} is the changed frequency in period to the PWM power transfer signal (kHz),

F_{op} is the base operating frequency of power transfer based on coupling and Rx loading condition (kHz)

60,000 is the trimmed frequency of the internal oscillator responsible for counting the period of the power transfer signal (FCLOCK_60). The counter divides this clock by 3 prior to implementing frequency adjustments.

Upon receiving such a packet, the MFC-IC will send an acknowledgement packet with WPC Proprietary Packet format and 0x18 as header and 0xFF as payload. MFC-IC does not require the check-sum of Tx FSK modulation packet.

2.1.5.18. Frequency Shift Keyed modulation (FSK) Transmitter to Receiver Communication

The MFC-IC implements FSK communication when used in conjunction with WPC compliant transmitters such as the P9235S. The FSK communication protocol allows the transmitter to send data to the receiver using the power transfer link in the form of modulating the power transfer signal. This modulation shall appear in the form of a change in the base operating frequency (f_{op}) to the modulated operating frequency (f_{mod}) in periods of 256 consecutive cycles. Equation 1 should be used to compute the modulated frequency based on any given operating frequency. The MFC-IC will only implement positive FSK Polarity adjustments, in other words, the modulated frequency will always be higher than the operating frequency during FSK communication.

2.1.5.19. FSK Communication Protocol

The FSK Byte encoding scheme and Packet Structure is similar to that defined in WPC specifications. The FSK communication will use a differential bi-phase encoding scheme to modulate data bits into the power transfer signal. The start bit will consist of 512 consecutive f_{mod} cycles (or a logic '0'). A logic '1' value will be sent by sending 256 consecutive f_{op} cycles followed by 256 f_{mod} cycles or vice versa, and a logic '0' is sent by sending 512 consecutive f_{mod} or f_{op} cycles.

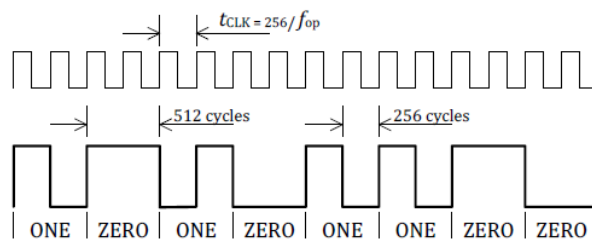


Figure 28. Example of differential bi-phase encoding.

Each byte will comply with the following start, data, Parity, stop asynchronous serial format structure:

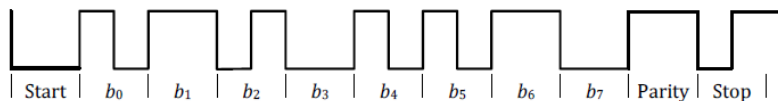


Figure 29. Example of asynchronous serial byte format.

Finally, the packet of each message will be composed of a single byte Header (0x2 in this case) and a single byte payload (0x0 or 0x1 in this case to indicate 5 V or 9 V adaptor).

2.1.6. HW Control and Monitor Registers

For Capacitor Divider HW registers, see *Capacitor Divider Registers*.

2.1.6.1. Ping Frequency Register, PingFreq_L (0x6A), PingFreq_H (0x6B)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x6A [7:0] | PingFreq_L [7:0] | R | - | 8 LSB of the Tx frequency during the ping in kHz. |
| 0x6B [7:0] | PingFreq_H [15:8] | R | - | 8 MSB of the Tx frequency during the ping in kHz. Ping Frequency = 0x008F => 143 = 143 kHz |

2.1.6.2. HW Flag Register, HW_Flag (0x81)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x81 [7] | VP5p0AP_EN[7] | RW | 0 | V5p0AP switch control. AP writes a "1" to enable the V5p0AP switch to power the P9412 from an external 5V supply. The P9412 is normally powered from Vrect. |
| 0x81 [6:5] | Reserved | R | 0x0 | Reserved |
| 0x81 [4] | ALIGN_EN | RW | 0 | XY Alignment enable. AP writes "1" to enable alignment FW. X align input on GP2 and Y align input on GP5. |
| 0x81 [3:0] | Reserved | R | 0x0 | Reserved |

2.1.6.3. Over Voltage Protection Register, OV_Set (0xB3)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0xB3[7:3] | Reserved | R | -0x00 | Reserved |
| 0xB3 [2:0] | OV_Set | RW | 0x05 0x05 | Set Overvoltage Protection level. The HW enables an additional DC Load when Vrect reaches the set level. The possible combinations are: 0h = 18.0V 1h = 21.3V 2h = 16.8V 3h = 14.7V 4h = 13.0V 5h = 23.4V 6h = 24.7V 7h = 26.0V BPP default value: 23.4V, and EPP default value: 23.4V. |

2.1.6.4. RX Mode Communication Modulation FET Register, CMFET_L (0xF4), CMFET_H (0xF5)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--------------------------|
| 0xF4 [7] | CMA | RW | 0x0 | 0 = Enable, 1 = Disable. |
| 0xF4 [6] | CMB | RW | 0x0 | 0 = Enable, 1 = Disable. |
| 0xF4 [5] | CM1 | RW | 0x1 | 0 = Enable, 1 = Disable. |
| 0xF4 [4] | CM2 | RW | 0x1 | 0 = Enable, 1 = Disable. |
| 0xF4 [3:0] | Reserved | RW | 0x0 | Reserved |
| 0xF5 [7:0] | Reserved | RW | 0x0 | Reserved |

1. Register ADDR 0xF4 applies when $V_{OUT} \leq 8.5V$ OR $8.5V < V_{OUT} < 12V$ AND $I_{OUT} \geq 320mA$ OR $12V < V_{OUT} < 17.5V$.

2.1.6.5. RX Mode AFC Communication Modulation FET Register, AFC_CMFET (0xB2)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--------------------------|
| 0xB2 [7] | CMA | RW | 0x0 | 0 = Enable, 1 = Disable. |
| 0xB2 [6] | CMB | RW | 0x0 | 0 = Enable, 1 = Disable. |
| 0xB2 [5] | CM1 | RW | 0x0 | 0 = Enable, 1 = Disable. |
| 0xB2 [4] | CM2 | RW | 0x0 | 0 = Enable, 1 = Disable. |
| 0xB2 [3:0] | Reserved | RW | 0x0 | Reserved |

1. Register ADDR 0xB2 applies when $8.5V < V_{OUT} < 12V$ AND $I_{OUT} < 320mA$.

2.1.6.6. RX Mode High Vout Communication Modulation FET Register, HiVout_CMFET (0x11B)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--------------------------|
| 0x11B [7] | CMA | RW | 0x1 | 0 = Enable, 1 = Disable. |
| 0x11B [6] | CMB | RW | 0x1 | 0 = Enable, 1 = Disable. |
| 0x11B [5] | CM1 | RW | 0x0 | 0 = Enable, 1 = Disable. |
| 0x11B [4] | CM2 | RW | 0x0 | 0 = Enable, 1 = Disable. |
| 0x11B [3:0] | Reserved | RW | 0x0 | Reserved |

1. Register ADDR 0x11B applies when VOUT > 17.5V.

2.1.6.7. Align X Register, AlignX (0xB0)

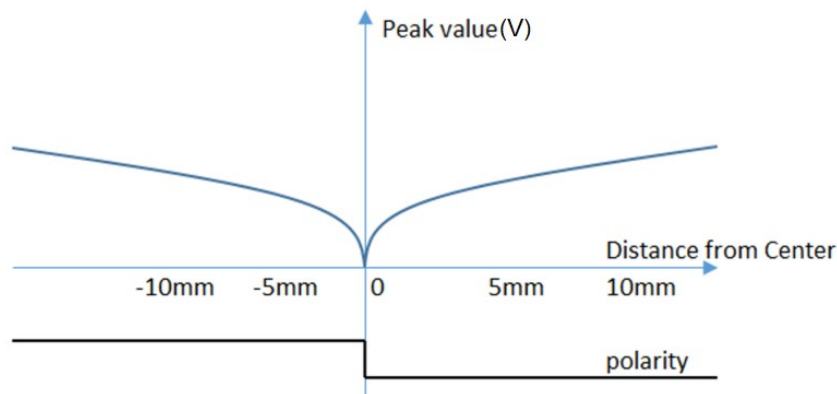
| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0xB0 [7:0] | AlignX | R | | 8-bit signed integer representing the X position of the alignment coil connected to the GPIO2 input. |

2.1.6.8. Align Y Register, AlignY (0xB1)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0xB1 [7:0] | AlignY | R | | 8-bit signed integer representing the Y position of the alignment coil connected to the GPIO5 input. |

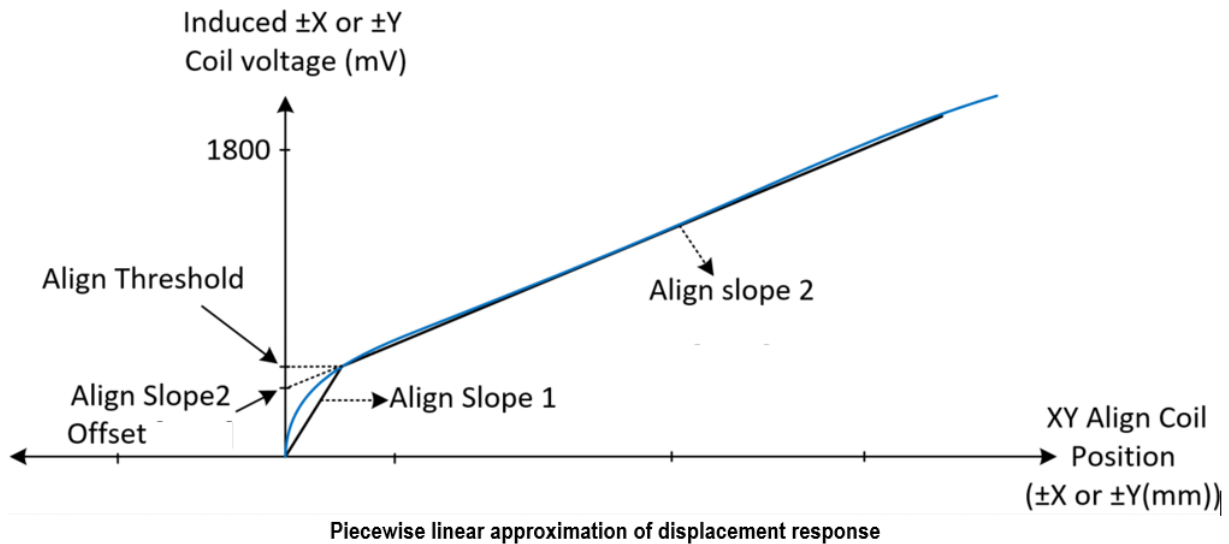
The alignment registers can be used to detect the current position of the Rx device's Rx Coil relative to the center of the Tx coil magnetic field source. To do this requires a special RX coil that includes XY position sensing coils. The position sensing coils are connected to the P9412's GP0 and GP1 pins which monitors the resulting voltage. When the coil is centered the voltage induced on the position coils will be 0V.

The induced voltage from one or both pairs of the XY coils it is observed to generally follow a voltage to distance relationship when the phone Rx coil center is moved out from center alignment along either the X or Y axis of the device.



Typical Voltage to Displacement Response of XY alignment coils

As seen by the graph, near the center alignment there is typically a region where the voltage to displacement slope of the curve is much steeper than the response of the induced voltage to distance of the XY coils at locations further from the aligned position. To account for these differences a piecewise linear approximation is used to correct for this non-linear response as shown below.



Where:

- Align Threshold is used to select between Slope1 and Slope2 for position calculation.
- Align Slope1 and Align Slope2 are determined based on characterization of XY coils under test
- Align Slope2 Offset is necessary to complete the slope-intercept form of a line equation used for Slope2.

2.1.6.9. Align adc Offset Registers, AlignAdcOffX (0x164), AlignAdcOffY (0x165)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x164 [7:0] | AlignAdcOffX | R | 0xDD | Signed 8 bit integer representing the ADC offset for X alignment input signal applied to GP2 |
| 0x165 [7:0] | AlignAdcOffY | R | 0xDD | Signed 8 bit integer representing the ADC offset for Y alignment input signal applied to GP5 |

2.1.6.10. Align Slope1 Registers, AlignSlope1X (0x166), AlignSlope1Y (0x167)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x166 [7:0] | AlignSlope1X | RW | 0x2B | First slope value of the 2-piece linear approximation for the X alignment value. Q5 Format. Slope1X = AlignSlope1 / 32. |
| 0x167 [7:0] | AlignSlope1Y | RW | 0x2B | First slope value of the 2-piece linear approximation for the Y alignment value. Q5 Format. Slope1Y = AlignSlope1 / 32. |

2.1.6.11. Align Slope2 Registers, AlignSlope2X (0x168), AlignSlope2Y (0x169)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x168 [7:0] | AlignSlope2X | RW | 0x45 | Second slope value of the 2-piece linear approximation for the X alignment value. Q5 Format. Slope2X = AlignSlope2 / 32. |
| 0x169 [7:0] | AlignSlope2Y | RW | 0x45 | Second slope value of the 2-piece linear approximation for the Y alignment value. Q5 Format. Slope2Y = AlignSlope2 / 32. |

2.1.6.12. Align Offset Registers, AlignOffX (0x16A), AlignOffY (0x16B)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x16A [7:0] | AlignOffX | RW | 0xE8 | Signed 8 bit integer representing the second line offset value of the 2-piece linear approximation for the X alignment value. |
| 0x16B [7:0] | AlignOffY | RW | 0xE8 | Signed 8 bit integer representing the second line offset value of the 2-piece linear approximation for the Y alignment value. |

2.1.6.13. Align Threshold Registers, AlignThreshX (0x16C), AlignThreshY (0x16D)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x16C [7:0] | AlignThreshX | RW | 0x1E | ADC threshold value for the slope change for X alignment. |
| 0x16D [7:0] | AlignThreshY | RW | 0x1E | ADC threshold value for the slope change for Y alignment. |

2.1.7. Vrect Control Registers

These registers define the behavior of Vrect Target above Vout. While Vout can be set by the AP directly, the Window between Vout and Vrect depends on the output power. Following algorithm is used to calculate Vrect Target:

$OutputPower(0.1W) = Vout * Iout * 10; PowerDifference = PwrKnee - OutputPower; (if PwrKnee > OutputPower)$

$PowerDifference = 0; (if PwrKnee < OutputPower)$

$Window (adc codes) = PowerDifference^2 * VrCorrFactor / 32;$

$Window = Window + VrMinCorr;$

$if (Window > VrMaxCorr) \quad Window = VrMaxCorr;$

$Window = Window + VRectAdj;$

$VrectTarget(adc codes) = Vout(adc codes) + Window;$

2.1.7.1. Target_Vrect Register, Vrect_Target_L(0x90), VrectTarget_H (0x91)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|----------------------|-----|---------------|---|
| 0x90 [7:0] | VrectTarget_L [7:0] | R | 0x5B | 8 LSB of current value of VrectTarget in ADC codes. Default value: 0x35B. |
| 0x91 [7:0] | VrectTarget_H [15:8] | R | 0x03 | 8 MSB of current value of VrectTarget in ADC codes. The ADC code to Voltage conversion formula is: $Target Vrect (V) = VrectTarget [15:0] * 30.975(V) / 4095$ |

2.1.7.2. Vrect Knee Register, PwrKnee (0x92)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x92 [7:0] | PwrKnee | RW | 0x19 | Threshold in units of 0.1W output power at which minimal window is applied. Default value: 0x19. |

2.1.7.3. Vrect Correction Factor Register, VrCorrFactor (0x93)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x93 [7:0] | VrCorrFactor | RW | 0x0C | Coefficient used in the Vrect Target calculation algorithm. Default value: 0x0C. |

2.1.7.4. Vrect Maximum Correction Register, VrMaxCorr_L (0x94), VrMaxCorr_H (0x95)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x94 [7:0] | VrMaxCorr_L [7:0] | RW | 0x9E | 8 LSB of maximum width of the window in ADC codes. |
| 0x95 [7:0] | VrMaxCorr_H [15:8] | RW | 0x00 | 8 MSB of maximum width of the window in ADC codes. Default vale: 0x009E. |

2.1.7.5. Vrect Minimum Correction Register, VrMinCorr_L (0x96), VrMinCorr_H (0x97)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x96 [7:0] | VrMinCorr_L [7:0] | RW | 0x09 | 8 LSB of minimum width of the window in ADC codes. |
| 0x97 [7:0] | VrMinCorr_H [15:8] | RW | 0x00 | 8 MSB of minimum width of the window in ADC codes. Default value: 0x0009. |

2.1.7.6. Vrect Adjust Register, VRectAdj (0x5E)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x5E [7:0] | VRectAdj | RW | 0x00 | 8-bit signed integer representing Vrect final adjustment in ADC codes (-128 to +127). The adjustment is applied during the final step of the VrectTarget calculation, thus overwriting Minimum and Maximum boundaries. $Vrect_Adj (V) = VRectAdj [7:0] * 30.975 (V) / 4095$ $Vrect (V) = Target Vrect + Vrect_Adj$ |

2.1.8. Capacitor Divider Registers

2.1.8.1. Capacitor Divider Mode Status Register, CDMoDeSts (0x100)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x100 [7:2] | Reserved | R | - | Reserved |
| 0x100 [1:0] | CD Mode Status | R | 0 | Indicates the firmware is running in Capacitor Divider mode. 0x0 Cap Divider in Disable mode 0x1 Cap divider in Bypass mode. CPout voltage is same as VOUT. 0x2 Cap Divider in Cap Div mode. CPout voltage is half of VOUT. |

2.1.8.2. Capacitor Divider Mode Request Register, CDMoDeReq (0x101)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x101 [7:2] | Reserved | R | - | Reserved |
| 0x101 [1:0] | CD Mode Req | RW | 0 | Set this bit to request a Cap Divider mode change. 0x0 Request Disable mode 0x1 Request Bypass mode. CPout voltage is same as VOUT. 0x2 Request Cap Div mode. CPout voltage is half of VOUT. |

2.1.8.3. TRX CPout Voltage Registers, VCPout_L (0x10C), VCPout_H (0x10D)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x10C [7:0] | VCPout_L [7:0] | R | - | 8 LSB of current VCPout voltage value. The AP may read this register to get current VCPout level in mV. |
| 0x10D [7:0] | VCPout_H [15:8] | R | - | 8 MSB of current VCPout voltage value. <i>Example: If VCPout = 0x1388h => 5000 = 5000mV = 5V</i> |

2.1.8.4. Capacitor Divider Vout Threshold Reg., CD_Vout_Thd_L (0x10A), CD_Vout_Thd_H (0x10B)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|----------------------|-----|---------------|--|
| 0x10A [7:0] | CD_Vout_Thd_L [7:0] | RW | 0x85 | 8 LSB of CD_Vout_Thd. Pause capacitor divider mode transitions when Vout is above this threshold. Value is in ADC codes. |
| 0x10B [7:4] | Reserved | R | 0x0 | Reserved |
| 0x10B [3:0] | CD_Vout_Thd_H [11:8] | RW | 0x09 | 4 MSB of CD_Vout_Thd. Pause capacitor divider mode transitions when Vout is above this threshold. Value is in ADC codes. The ADC code to Voltage conversion formula is: $Vout(V) = Vout(adc\ count) * 21.0(V) / 4095 = 2437 * 21 / 4095 = 12.5V$ |

2.1.8.5. Capacitor Divider Frequency Set Register, CD_Freq_L (0x108), CD_Freq_H (0x109)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x108 [7:0] | CD_Freq_L [7:0] | RW | 0x58 | 8 LSB of CD_Freq. Set frequency of the capacitor divider in kHz units. |
| 0x109 [7:0] | CD_Freq_H [15:8] | RW | 0x02 | 8 MSB of CD_Freq. Set frequency of the capacitor divider in kHz units. Default value: 600kHz. <i>Example: 0x0258h => 600 = 600kHz</i> |

2.1.9. Foreign Object Detection Registers

The FOD Registers are divided into 8 pairs. Each pair has one byte for gain setting, and one byte for offset setting. The first 6 pairs control the Received Power calculation for 6 power sectors during Power Transfer phase. The seventh pair calibrates the internal DC Load, and the eighth pair is used during EPP Calibration Phase 1. The set values of the FOD Registers are found with the help of Renesas developed calibration procedure using the nok9 tester.

The FW initializes the FOD Registers from one of two sets default values, one for BPP mode, and one for EPP mode. The correct set is loaded at completion of the ID & Configuration Phase.

2.1.9.1. RX FOD Adjustable Parameters Registers, (0x70 ~ 0x7F)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|-----------------------------------|
| 0x70 [7:0] | FOD_0_A | R/W | 0xC0 | FOD adjustable parameters, gain |
| 0x71 [7:0] | FOD_0_B | R/W | 0x16 | FOD adjustable parameters, offset |
| 0x72 [7:0] | FOD_1_A | R/W | 0xAC | FOD adjustable parameters |
| 0x73 [7:0] | FOD_1_B | R/W | 0X1A | FOD adjustable parameters |
| 0x74 [7:0] | FOD_2_A | R/W | 0X98 | FOD adjustable parameters |
| 0x75 [7:0] | FOD_2_B | R/W | 0X14 | FOD adjustable parameters |
| 0x76 [7:0] | FOD_3_A | R/W | 0X94 | FOD adjustable parameters |
| 0x77 [7:0] | FOD_3_B | R/W | 0X12 | FOD adjustable parameters |
| 0x78 [7:0] | FOD_4_A | R/W | 0X92 | FOD adjustable parameters |
| 0x79 [7:0] | FOD_4_B | R/W | 0X08 | FOD adjustable parameters |
| 0x7A[7:0] | FOD_5_A | R/W | 0X92 | FOD adjustable parameters |
| 0x7B[7:0] | FOD_5_B | R/W | 0X08 | FOD adjustable parameters |

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|----------------------------|
| 0x7C[7:0] | FOD_6_A | R/W | 0X14 | FOD calibration parameters |
| 0x7D[7:0] | FOD_6_B | R/W | 0X00 | FOD calibration parameters |
| 0x7E[7:0] | FOD_7_A | R/W | 0X01 | FOD calibration parameters |
| 0x7F[7:0] | FOD_7_B | R/W | 0X50 | FOD calibration parameters |

For TX mode of operation, the FOD parameter is a single value and the threshold is set by the difference of the TX power (power transmitted) and the RX power (power received). Therefore the $TxFodThreshold = TxPower - RxPower$. Adjustments to the $TxFodThreshold$ can be made using the $TxFodGain$ and $TxFodOffset$ registers.

2.1.9.2. TX FOD Threshold Registers, TX_FOD_Thrsh_L (0xD4), TX_FOD_Thrsh_H (0xD5)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0xD4[7:0] | TxFodThrsh_L [7:0] | R/W | 0xDC | 8 LSB of TX FOD Threshold. Default value TBD |
| 0xD5[7:0] | TxFodThrsh_H [15:8] | R/W | 0x05 | 8 MSB of TX FOD Threshold. $TxFodThrsh = 0x05DC = 1500 \text{ mW}$ |

2.1.9.3. TX FOD Gain Register, TX_FOD_Gain (0xD1)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0xD1[7:0] | TxFodGain [7:0] | R/W | 0x13 | TX FOD Gain is used for the power calculation of the TX FOD Default value 0x13 $TxFodGain = 0x13 = 19 \times 1/100 = 0.19$ |

2.1.9.4. TX FOD Offset Registers, TX_FOD_Offset_L (0xD2), TX_FOD_Offset_H (0xD3)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|----------------------|-----|---------------|--|
| 0xD2[7:0] | TxFodOffset_L [7:0] | R/W | 0x64 | 8 LSB of the TX FOD offset. The TX FOD Offset is used in the power calculation of the TX FOD. |
| 0xD3[7:0] | TxFodOffset_H [15:8] | R/W | 0x00 | 8 MSB of the TX FOD offset. The offset is signed and needs to be converted in 2's compliment. Default is 100mW. <i>Example: If 100mW then $100\text{mW} \Rightarrow 0x0064$, $-100\text{mW} \Rightarrow 0xFF9C$</i> |

2.1.9.5. TX FOD Offset Option Register, FOD_OffsetOpt (0xA3)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0xA3 [7:0] | FOD_OffsetOpt [7:0] | R/W | 0x00 | FOD_OffsetOpt adjusts the $TxFodThrsh$ for phone to phone charging. The AP writes to this register to update this value as required. The byte is signed and needs to be converted in 2's compliment. The step size is 40 mW: $FOD_OffsetOpt \text{ (mW)} = 40 \times FOD_OffsetOpt [7:0]$ in decimal. Examples: $FOD_OffsetOpt = 0x50 = 40 \times 80 = + 3200\text{mW}$; $Offset = 0xFE = 40 \times -2 = - 80\text{mW}$; $Offset = 0x7F = \text{Disable FOD}$ |

2.1.10. WPC Basic and Extended Protocol Registers

After the Rx has been placed on an EPP TX, the P9412 must report its Q-Factor to the Tx so the Tx may check for an Open FOD alarm event. The Tx is responsible for measuring the current quality factor of the Tx coil before connecting to the Rx device, and the P9412 will report the typical Q observed by the device. This will be compared to the Tx Q-factor measured value, and if within the Tx allowed range, charging will commence. If it is outside the allowable range the Tx will not power the Rx due to the suspected presence of a Foreign Object.

This feature applies to the EPP wireless power protocol and every phone or end product that is likely to need some tuning to the reported Quality Factor value. This can be programmed in one of two ways using:

- Two resistor dividers from LDO1P8 to IO0 (Q factor A) and IO1 (Q factor A) read-back by ADC, or
- Firmware register [0x83] programmed at the factory, which is selected by connecting GPIO0 to GND

The simplest and recommended method is to set the value using external resistors read by the ADC at startup to set the reported Q by hardware population option (recommended), or the value can be stored in a register to be sent to the Tx (requires custom firmware to be programmed at the factory). If the value is stored in a register, each device must be able to update the MTP register on the manufacturing floor to support device receiver type or model variations. When using the internal register option with IO0 tied to GND, the IO1 input should also be biased to GND or LDO1P8 to conserve power and avoid a floating input. Since the result of IO1 is not used in this case (register will always be set to 0), it is permissible to leave the pin floating as well. The GPIO0 and GPIO1 pins are the inputs used to select the firmware register or hardware selection for the reported Q-factor value. The external resistor dividers should be biased by the LDO1P8 power supply to avoid incorrect ADC voltage from being read in the case of a dead battery. The following table defines the reported Q-factor as a function of voltage on IO0 and IO1. When selecting resistors, the values should be selected such that the typical voltage is centered within each range and the total resistance should be > 50kΩ to conserve power.

| GP0 Volt (mV) | Q factor A | GP1 Volt (mV) | Q factor B * |
|-------------------|-------------|-------------------|--------------|
| GP0 < 300 | 0 (Default) | GP1 < 200 | 0 |
| 300 < GP0 < 550 | 30 | 200 < GP1 < 400 | 1 |
| 550 < GP0 < 800 | 40 | 400 < GP1 < 600 | 2 |
| 800 < GP0 < 1050 | 50 | 600 < GP1 < 800 | 3 |
| 1050 < GP0 < 1300 | 60 | 800 < GP1 < 1000 | 4 |
| 1300 < GP0 < 1550 | 70 | 1000 < GP1 < 1200 | 5 |
| 1550 < GP0 < 1800 | 80 | 1200 < GP1 < 1400 | 6 |
| GP0 > 1800 | 90 | 1400 < GP1 < 1600 | 7 |
| | | 1600 < GP1 < 1800 | 8 |
| | | GP1 > 1800 | 9 |

* if Q factor A is 0, Q factor B will be set to 0 as well.

2.1.10.1. EPP Q-Factor Register, EPP_Q_Factor (0x83)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x83 [7:0] | EPP_Q_Factor | RW | 0x60 | <p>If Q Factor A is non-zero, Q Factor = Q Factor A + Q Factor B. See EPP Q-Factor A Register, EPP_Q_Fact_A (0x138) and EPP Q-Factor B Register, EPP_Q_Fact_B (0x139)</p> <p>If Q Factor A is zero, the default value will be used.</p> <p>The AP can overwrite the Q Factor value by writing to this register within 300ms after Vrect ON interrupt is received.</p> |

2.1.10.2. EPP Q-Factor A Register, EPP_Q_Fact_A (0x138)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x138 [7:0] | EPP_Q_Fact_A | R | 0x00 | Q-Factor value using GPIO0 input, 10's unit. See Table below |

1. This register is populated only for Reference Design Version 2 (see Reference Design Version Register, RefDesignVer (0x4A)).

2.1.10.3. EPP Q-Factor B Register, EPP_Q_Fact_B (0x139)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x139 [7:0] | EPP_Q_Fact_B | R | 0x00 | Q-Factor value using GPIO1 input, 1's unit. See Table below |

1. This register is populated only for Reference Design Version 2 (see Reference Design Version Register, RefDesignVer (0x4A)).

2.1.10.4. EPP TX Guaranteed Power Register, EPP_TXGuarPwr (0x84)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x84 [7:0] | EPP_TXGuarPwr | R | 0x00 | Tx Guaranteed Power Value as reported in the Capabilities packet (Header 0x31). Units of 0.5W. WPC spec 5.3.3.3, B ₀ |

2.1.10.5. EPP TX Potential Power Register, EPP_TXPotentPwr (0x85)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x85 [7:0] | EPP_TXPotentPwr | R | 0x00 | Tx Potential Power Value as reported in the Capabilities packet (Header 0x31). Units of 0.5W. WPC spec 5.3.3.3, B ₁ |

2.1.10.6. EPP TX Capability Flag Register, EPP_TXCapaFlag (0x86)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x86 [7:0] | EPP_TXCapaFlag | R | 0x00 | Tx Capabilities packet flags as reported in the Capabilities packet (Header 0x31). See WPC spec 5.3.3.3, B ₂ |

| | b ₇ | b ₆ | b ₅ | b ₄ | b ₃ | b ₂ | b ₁ | b ₀ |
|----------------|----------------|----------------|------------------------|----------------|----------------|----------------|----------------|----------------|
| B ₀ | Power Class | | Guaranteed Power Value | | | | | |
| B ₁ | Reserved | | Potential Power Value | | | | | |
| B ₂ | Reserved | | | | | | WPID | Not Res Sens |

2.1.10.7. EPP Renegotiation Status Register, EPP_RN_Sts (0x87)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x87 [7:3] | Reserved | R | 0x00 | Reserved |
| 0x87 [2] | RN_ERROR | R | 0x0 | Re-Negotiation phase completed with error or NAK. |
| 0x87 [1] | RN_DONE | R | 0x0 | Re-Negotiation phase completed with ACK. |
| 0x87 [0] | RN_CapaREQ | R | 0x0 | Request to send General Request Capabilities Packet during Re-Negotiation phase. This bit is self-cleared upon exit from the phase. |

2.1.10.8. EPP Current RPP Header Register, MPCur_RPP (0x88)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x88 [7:0] | MPCur_RPP | R | 0x04 0x31 | Current value of the Received Power Packet header. BPP default value: 0x04; EPP default value: 0x31. |

2.1.10.9. EPP Current Negotiated Power Register, MPCur_NegPwr (0x89)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x89 [7:0] | MPCur_NegPwr | R | 0x0A 0x0C | Current value of the negotiated Guaranteed Power Value (Negotiated Power as a result of the power negotiation). Units of 0.5W. BPP default value: 0x0A; EPP default value: 0x0C. |

2.1.10.10. EPP Current Maximum Power Register, MPCur_MaxPwr (0x8A)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x8A [7:0] | MPCur_MaxPwr | R | 0x0A 0x0C | Current value of the negotiated Maximum Power. Units of 0.5W. BPP default value: 0x0A; EPP default value: 0x0C. |

2.1.10.11. EPP Current FSK Modulation Register, MPCur_FSK (0x8B)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x8B [7:3] | Reserved | R | 0x00 | Reserved |
| 0x8B [2] | FSK_POLARITY | R | 0x0 | Current value of the negotiated FSK modulation polarity. |
| 0x8B [1:0] | FSK_DEPTH | R | 0x0 | Current value of the negotiated FSK modulation depth. |

2.1.10.12. EPP Request RPP Header Register, MPReq_RPP (0x8C)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x8C [7:0] | MPReq_RPP | RW | 0x31 | Requested value to Re-Negotiate Received Power Packet header. This register is provided here for completeness. It is not used. |

2.1.10.13. EPP Request Re-Negotiated Power Register, MPReq_NegPwr (0x8D)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x8D [7:0] | MPReq_NegPwr | RW | 0x0C | Requested value to Re-Negotiate Guaranteed Power Value (Negotiated Power as a result of the power negotiation). Units of 0.5W. EPP default value: 0x0C. |

2.1.10.14. EPP Request Maximum Power Register, MPReq_MaxPwr (0x8E)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x8E [7:0] | MPReq_MaxPwr | RW | 0x0C | Requested value to Re-Negotiate Maximum Power. Units of 0.5W. EPP default value: 0x0C. |

2.1.10.15. EPP Request FSK Modulation Register, MPReq_FSK (0x8F)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x8F [7:3] | Reserved | R | 0x00 | Reserved |
| 0x8F [2] | FSK_POLARITY | RW | 0x0 | Requested value to Re-Negotiate FSK modulation polarity. |
| 0x8F [1:0] | FSK_DEPTH | RW | 0x0 | Requested value to Re-Negotiate FSK modulation depth. |

2.1.10.16. WPC Spec Revision Register, WPC_SpecRev (0xB9)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0xB9 [7:0] | WPC_SpecRev | R | 0x00 | WPC Spec revision value. Value is updated once ID packet is received in TX mode or in RX mode during EPP negotiation. It is not updated in BPP RX mode. WPC_SpecRev = 0x12 = Spec revision 1.2. |

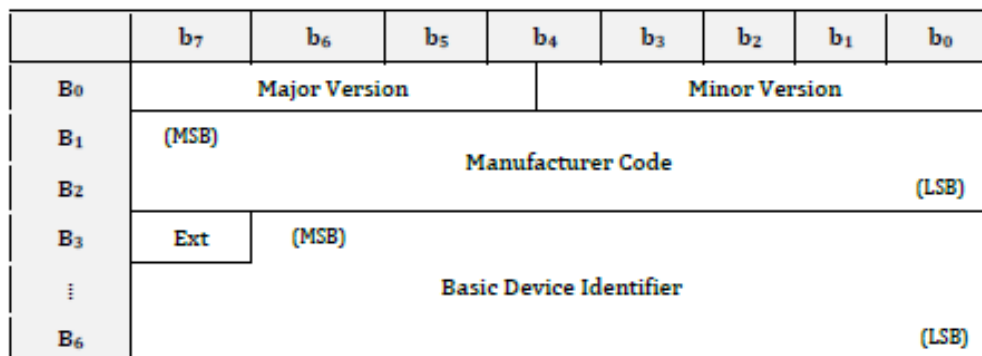
2.1.10.17. EPP Rx Manufacturer Code Reg., MpRxManufCode_L (0xBA), MpRxManufCode_H (0xBB)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|------------------------|-----|---------------|---|
| 0xBA [7:0] | MpRxManufCode_L [7:0] | R | 0x00 | 8 LSB of Rx WPC Manufacturer code. The code is assigned by WPC to each company producing compliant products. |
| 0xBB [7:0] | MpRxManufCode_H [15:8] | R | 0x00 | 8 MSB of Rx WPC Manufacturer code. Value is updated once ID packet is received in TX mode or in RX mode during EPP negotiation. It is not updated in BPP RX mode. |

2.1.10.18. WPC Identification Register, WPC_ID (0xE0 ~ 0xE3)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0xE0 [7:0] | WPC_ID[0] | R | - | b4-b0 = Minor Version of WPC spec for power receiver b7-b5 : Major version of WPC spec for power receiver |
| 0xE1 [7:0] | WPC_ID[1] | R | - | Manufacturer ID (MSB) |
| 0xE2 [7:0] | WPC_ID[2] | R | - | Manufacturer ID (LSB) |
| 0xE3 [7:0] | WPC_ID[3] | R | - | MSB of Device ID, if b7 is 0 then no Extended Device Identifier is used if b7 is 1 then Extended Device Identifier is used |

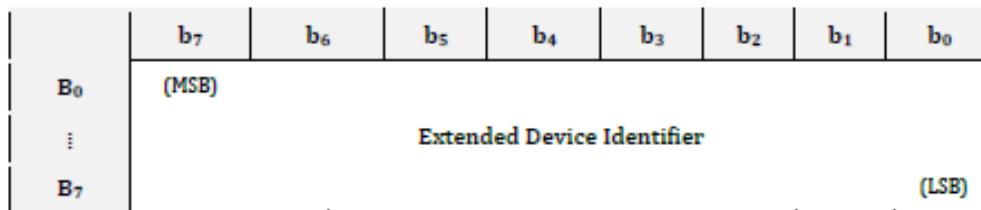
1. This register displays the information of the Rx Identification packet (0x71) read during Identification phase. Valid in TX mode only.



2.1.10.19. Extended Identification Packet Register, WPC_ExtID (0xE4 ~ 0xEB)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|-----------------------------------|
| 0xE4 [7:0] | WPC_ExtID[0] | R | - | MSB of Extended Device Identifier |
| 0xE5 [7:0] | WPC_ExtID[1] | R | - | |
| 0xE6 [7:0] | WPC_ExtID[2] | R | - | |
| 0xE7 [7:0] | WPC_ExtID[3] | R | - | |
| 0xE8 [7:0] | WPC_ExtID[4] | R | - | |
| 0xE9 [7:0] | WPC_ExtID[5] | R | - | |
| 0xEA [7:0] | WPC_ExtID[6] | R | - | |
| 0xEB [7:0] | WPC_ExtID[7] | R | - | LSB of Extended Device Identifier |

1. This register displays the information of the Rx Identification packet (0x81) read during Identification phase



2.1.10.20. Signal Strength Packet Register, SSPValue (0xB4)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0xB4 [7:0] | SSPValue | R | 0x00 | The value of the signal strength packet. This is the first packet sent to the transmitter in the ping phase. Value is valid in both RX and TX modes. SS = SSPValue / 255. If SSPValue = 0x8F => 143 / 255 = 56.1% |

2.1.10.21. Control Error Packet Register, CEPValue (0x5F)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x5F [7:0] | CEPValue | R | 0x00 | Control Error packet value. Value is valid in both RX and TX modes. |

2.1.10.22. RX Power Register, Rx_Pwr_L (0xCE), Rx_Pwr_H (0xCF)

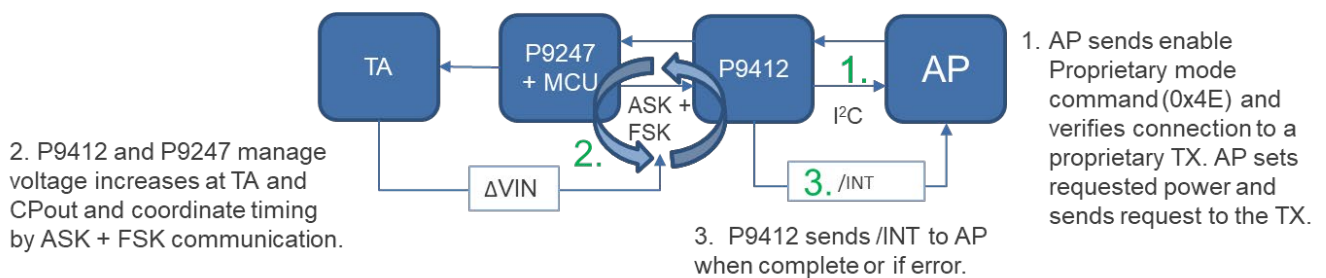
| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0xCE [7:0] | Rx_Pwr_L [7:0] | R | 0x00 | 8 LSB of current unsigned integer value contained in this field indicating the average amount of power that the Rx receives from the Tx in mW. Valid in RX mode only. |
| 0xCF [7:0] | Rx_Pwr_H [15:8] | R | 0x00 | 8 MSB of current unsigned integer value contained in this field indicating the average amount of power that the Rx receives from the Tx in mW. Valid in RX mode only. |

2.1.10.23. WPC Manufacturer ID Register, WPC_ManufID_L (0x15E), WPC_ManufID_H (0x15F)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|----------------------|-----|---------------|---|
| 0x15E [7:0] | WPC_ManufID_L [7:0] | R | 0x50 | 8 LSB of WPC Manufacturer code. The code is assigned by WPC to each company producing compliant products. |
| 0x15F [7:0] | WPC_ManufID_H [15:8] | R | 0x00 | 8 MSB of WPC Manufacturer code. The code is assigned by WPC to each company producing compliant products. Valid in RX and TX modes. WPC_ManufID default is 0x50 = PRMC for Renesas. The AP must update this value within 300ms after Vrect ON interrupt for the new value to take effect. |

2.1.11. Proprietary High-Power Protocol Registers

The P9412 can run in a high-power mode when coupled to a Renesas Proprietary TX (P9247). A simplified flow chart is shown below. The AP is required to start the high power operation by sending a command to transition to cap divider mode (if not already operating in CD mode). Once verified that the P9412 is in CD mode, the AP sends a command to enable the Renesas proprietary mode. The AP will receive an interrupt when the TX responds and determine if the P9412 is connected to a Renesas proprietary TX. After confirmation, the AP sends a request for high power to the TX. The TX and RX begin power negotiation, and the P9412 will send an interrupt when the negotiation is complete.



The detailed sequence of steps is as follows:

1. Clear all interrupts
 - a. Write 0xFFFF to register 0x3A
 - b. Write 0x20 to register 0x4E
2. Cap Divider mode should be enabled.
 - a. Read CD mode status register 0x100, it should read 0x02 if in Cap Divider mode.
 - b. If not, Write 0x02 to register 0x101
 - c. Write 0x40 to register 0x4E
 - d. Wait for CD_MODECHANGE_INT (0x37[3])
 - e. Clear CD_MODECHANGE interrupt
3. Enable Proprietary mode
 - a. Write 0x01 to register 0x4F
4. Wait for PropModeStatus interrupt, register 0x37[4].
 - a. Read the System Mode register 0x4C, it should read 0x03 if Proprietary mode is enabled.
 - b. If there was an error during Proprietary mode enable, Proprietary Error Status register (0xC9) is updated with the error code.
 - c. Clear the PropModeStatus interrupt
5. Read TX potential power register (0xC4) to see TX max power capability. Value is in 0.5W units.
6. Request power. Supported range is between 7.5W to 30W.
 - a. Write requested power to register 0xC5 (in 0.5W units)
 - b. Write 0x02 to register 0x4F
7. Wait for PropModeStatus interrupt, register 0x37[4].
 - a. AP read the final achieved power at register 0xC6 (in 0.5W units)
 - b. If there are any errors from Status registers – PropModeStatus (0xC8) and PropErrStatus (0xC9)
 - c. Clear the PropModeStatus interrupt

Note: If any more power changes required, start from step 5. The first 4 steps are required only once for enabling the proprietary mode.

A detailed flow chart is provided below.

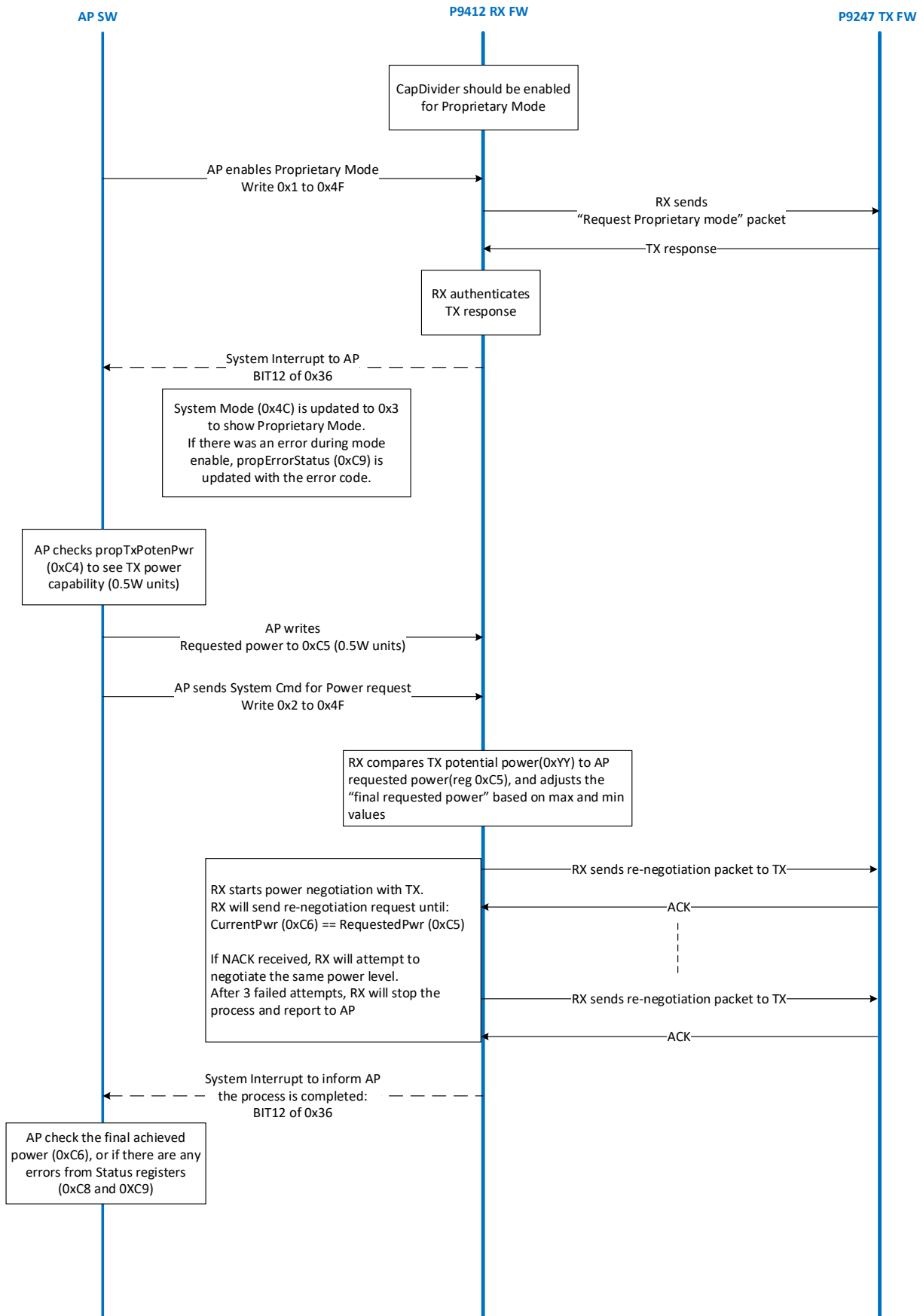


Figure 30. Proprietary High Power Protocol Flow Chart

2.1.11.1. Proprietary Tx Potential Power Register, PropTxPotenPwr (0xC4)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0xC4 [7:0] | PropTxPotenPwr | R | 0x00 | TX Potential Power value in 0.5W units. This is based on the TX response of "Proprietary Mode Request" packet |

2.1.11.2. Proprietary Requested Power Register, PropReqPwr (0xC5)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0xC5 [7:0] | PropReqPwr | RW | 0x60 | AP requested power value in 0.5W units. Maximum value is 60 (30W), minimum value is 15 (7.5W). |

2.1.11.3. Proprietary Current Power Register, PropCurrPwr (0xC6)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0xC6 [7:0] | PropCurrPwr | R | 0x12 | Current power value in 0.5W units. Updated on each negotiation stage |

2.1.11.4. Proprietary Negotiated Power Step Size Register, PropModePwrStep (0xC7)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0xC7 [7:0] | PropModePwrStep | RW | 0x00 | Power ramp-up step per negotiation stage. In 0.5W units. |

2.1.11.5. Proprietary Mode Status Register, PropModeStatus (0xC8)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0xC8 [7:0] | PropModeStatus | R | 0x00 | BIT 0: PROPMODETXAUTH - TX authentication is successful and proprietary mode is enabled. BIT 1: PROPMODENEGINIT - Power re-negotiation is initialized. BIT 2: PROPMODEPWRNEG - Power re-negotiation is in progress. BIT 3: PROPMODEDONE - Power re-negotiation is done. BIT 4: PROPMODEERR - Error during proprietary mode operation. Check propErrorStatus (0xC9) for error details. |

2.1.11.6. Proprietary Mode Error Register, PropErrStatus (0xC9)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0xC9 [7:0] | PropErrStatus | R | 0x00 | BIT 0: PROPERRTXNACK - TX sent NACK for requested power after 3 tries. Check propCurrentPwr (0xC6) for achieved power level. BIT 1: PROPERRAUTH - Error during TX authentication. Either TX does not support Renesas Prop Protocol, or the authentication is failed. BIT 2: PROPERRNEGINPRG - Negotiation is in progress. New Power request is sent from AP during ongoing power negotiation. It does not stop the ongoing negotiation. BIT 3: PROPERRINIT - Proprietary mode initialization error. Currently returned when TX does not support EPP. BIT 4: PROPERRCAPDIV - Cap Divider is not enabled. BIT 5: PROPERRTXPWR - TX potential power is not enabled. Returned when AP makes a power request before enabling proprietary mode and authentication. |

2.1.12. TX Mode Registers

2.1.12.1. TX Status Registers, Status_L (0x34), Status_H (0x35)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x34 [7] | Reserved | R | 0 | Reserved |
| 0x34 [6] | CEP OVERDRIVE | R | 0 | "1" indicates a CEP Over Drive condition exists. See TX Mode CEP Threshold Register, TxCepThrshVal (0xD6) and TX Mode CEP Threshold Count Limit Register, TxCepThrshCntLmt (0xD7). |
| 0x34 [5] | MODE_CHANGE | R | 0 | No function attached. Refer to "TRX System Operating Mode Register, Sys_Op_Mode (0x4C)". |
| 0x34 [4] | OVER_VOLT | R | 0 | Set if Overvoltage Protection circuit is enabled. Cleared otherwise. Interrupt event is generated on SET and CLR events. |
| 0x34 [3] | OVER_CURR | R | 0 | Set if Overcurrent Protection circuit is enabled. Cleared otherwise. Interrupt event is generated on SET and CLR events. |
| 0x34 [2] | OVER_TEMP | R | 0 | Set if Internal temperature exceeds 130°C. Cleared otherwise. Interrupt event is generated on SET and CLR events. |
| 0x34 [1] | TX_CONFLICT | R | 0 | Set if a foreign TX is detected in TX mode. See TX Conflict Threshold Register, TxConfThrsh (0x134) and TX Conflict Count Register, TxConfCnt (0xAF). |
| 0x34 [0] | Reserved | R | 0 | Reserved |
| 0x35 [7] | Data Received | R | 0 | "1" indicates TX data is received when in RX mode or RX data received when in TX mode. "0" indicates no data is received. |
| 0x35 [6] | CD_ERROR | R | 0 | "1" indicates a Capacitor Divider error condition exists while operating in TX mode. Possible error condition(s): Failure to turn on Capacitor Divider bypass FETs when entering TX mode This bit is cleared together with the corresponding interrupt flag. |
| 0x35 [5] | RX Not Detected | R | 0 | "1" indicates RX is not detected in 180 seconds after the first ping in TX mode, "0" indicates RX is detected in 180 seconds after the first ping in TX mode. |
| 0x35 [4] | TX FOD | R | 0 | "1" indicates TX FOD condition exists, "0" indicates no such a condition exists. |
| 0x35 [3] | RX Connected | R | 0 | "1" indicates RX and TX is connected (power transfer), "0" indicates TX and RX connection has not been established. |
| 0x35 [2] | AC Missing Detect | R | 0 | "1" indicates valid AC signal is not present, "0" indicates AC signals exist. Interrupt only generated after power up from battery (external source different from AC power). |
| 0x35 [1] | Reserved | R | 0 | Reserved |
| 0x35 [0] | Reserved | R | 0 | Reserved |

2.1.12.2. TX Interrupt Registers, INT_L (0x36), INT_H (0x37)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x36 [7] | Reserved | R | 0 | No function attached. |
| 0x36 [6] | CEP_OVERDRV_INT | R | 0 | "1" indicates a pending interrupt for CEP Over Drive. See TX Mode CEP Threshold Register, TxCepThrshVal (0xD6) and TX Mode CEP Threshold Count Limit Register, TxCepThrshCntLmt (0xD7). |
| 0x36 [5] | MODECHANGE_INT | R | 0 | "1" indicates a pending interrupt for Mode Change. Read current mode from "TRX System Operating Mode Register, Sys_Op_Mode (0x4C)". |

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x36 [4] | OVER_VOLT_INT | R | 0 | "1" indicates a pending interrupt for Over Voltage event. |
| 0x36 [3] | OVER_CURR_INT | R | 0 | "1" indicates a pending interrupt for Over Current event. |
| 0x36 [2] | OVER_TEMP_INT | R | 0 | "1" indicates a pending interrupt for Over Temperature event. |
| 0x36 [1] | TX CONFLICT_INT | R | 0 | "1" indicates a pending interrupt when a foreign TX is detected. See TX Conflict Threshold Register, TxConfThrs (0x134) and TX Conflict Count Register, TxConfCnt (0xAF). |
| 0x36 [0] | Reserved | R | 0 | Reserved |
| 0x37 [7] | Data Received_INT | R | 0 | "1" indicates a pending interrupt for TX data received when in RX mode or RX data received when in TX mode. (No data received state change to data received state). When in TX mode this interrupt is set only when PPP has header equal to 0x18, 0x28, and 0x05. |
| 0x37 [6] | CD_ERROR_INT | R | 0 | "1" indicates a pending interrupt due to a Capacitor Divider Error when entering TX mode. |
| 0x37 [5] | RX Not Detected_INT | R | 0 | "1" indicates a pending interrupt for RX is not detected in 180 seconds after the first ping in TX mode. |
| 0x37 [4] | TX FOD_INT | R | 0 | "1" indicates a pending interrupt for TX FOD event. |
| 0x37 [3] | RX Connected_INT | R | 0 | "1" indicates a pending interrupt for TX mode connection state change (connected to disconnected or vice versa). |
| 0x37 [2] | AC Missing_INT | R | 0 | "1" indicates a pending interrupt that valid AC does not exist, "0" indicates AC signals exist. Interrupt only generated after power up from battery (external source different from AC power). |
| 0x37 [1] | Reserved | R | 0 | Reserved |
| 0x37 [0] | Reserved | R | 0 | Reserved |

2.1.12.3. TX Interrupt Enable Registers, INT_Enable_L (0x38), INT_Enable_H (0x39)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x38 [7] | Reserved | R | 0 | Reserved |
| 0x38 [6] | CEP_OVERDRV_EN | RW | 1 | CEP_OVERDRV interrupt enable. Default value is "1" AP writes "0" to disable the interrupt. |
| 0x38 [5] | MODECHANGE_EN | RW | 1 | Mode Changed interrupt enable. Default value is "1". AP writes "0" to disable the interrupt. |
| 0x38 [4] | OVER_VOLT_EN | RW | 1 | Overvoltage condition ON/OFF interrupt enable. Default value is "1" |
| 0x38 [3] | OVER_CURR_EN | RW | 1 | Overcurrent condition ON/OFF interrupt enable. Default value is "1" |
| 0x38 [2] | OVER_TEMP_EN | RW | 1 | Over-temperature condition ON/OFF interrupt enable. Default value is "1" |
| 0x38 [1] | TX CONFLICT_EN | RW | 1 | Default value is "1" in TX mode. When powered-up in other modes, default value is "0". |
| 0x38 [0] | Reserved | R | 0 | Reserved |
| 0x39 [7] | Data Received_EN | RW | 1 | TX Data Received interrupt enable. Default value is "1". AP writes "0" to disable the interrupt |
| 0x39 [6] | CD_ERROR_EN | RW | 1 | Capacitor Divider Mode Changed interrupt enable. Default value is "1". AP writes "0" to disable the interrupt. |
| 0x39 [5] | RX Not Detected_EN | RW | 1 | Default value is "1" in TX mode. When powered-up in other modes, default value is "0". |
| 0x39 [4] | TX FOD_EN | RW | 1 | Default value is "1" in TX mode. When powered-up in other modes, default value is "0". |
| 0x39 [3] | RX Connected_EN | RW | 1 | Capacitor Divider Mode Changed interrupt enable. Default value is "1" in TX mode. When powered-up in other modes, default value is "0". |

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x39 [2] | AC Missing_EN | R/W | 0 | AP writes "1" is to enable the interrupt from the Interrupt Registers' corresponding bit, "0" is to disable the interrupt. Interrupt only generated after power up from battery (external source different from AC power). |
| 0x39 [1] | Reserved | R | 0 | Reserved |
| 0x39 [0] | Reserved | R | 0 | Reserved |

2.1.12.4. TX Interrupt Clear Registers, INT_Clear_L (0x3A), INT_Clear_H (0x3B)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x3A [7] | Reserved | R | 0 | Reserved |
| 0x3A [6] | CEP_OVERDRV_CLR | RW | 0 | AP writes "1" to clear the corresponding Interrupt Registers' bit and this bit is self-cleared to "0" (by MCU) afterwards. |
| 0x3A [5] | MODECHANGE_CLR | RW | 0 | Mode Changed interrupt flag clear |
| 0x3A [4] | OVER_VOLT_CLR | RW | 0 | Overvoltage condition ON/OFF interrupt flag clear |
| 0x3A [3] | OVER_CURR_CLR | RW | 0 | Overcurrent condition ON/OFF interrupt flag clear |
| 0x3A [2] | OVER_TEMP_CLR | RW | 0 | Over-temperature condition ON/OFF interrupt flag clear |
| 0x3A [1] | TX CONFLICT_CLR | RW | 0 | AP writes "1" to clear the corresponding Interrupt Registers' bit and this bit is self-cleared to "0" (by MCU) afterwards |
| 0x3A [0] | Reserved | R | 0 | ADT Error interrupt flag clear. AP writes "1" to clear the corresponding Interrupt Registers' bit and this bit is self-cleared to "0" (by MCU) afterwards. |
| 0x3B [7] | Data Received_CLR | RW | 0 | Tx data received interrupt flag clear |
| 0x3B [6] | CD_ERROR_CLR | RW | 0 | Capacitor Divider Error interrupt flag clear. |
| 0x3B [5] | RX Not Detect_CLR | RW | 0 | AP writes "1" to clear the corresponding Interrupt Registers' bit and this bit is self-cleared to "0" (by MCU) afterwards. |
| 0x3B [4] | TX FOD_CLR | RW | 0 | AP writes "1" to clear the corresponding Interrupt Registers' bit and this bit is self-cleared to "0" (by MCU) afterwards. |
| 0x3B [3] | RX Connected_CLR | RW | 0 | AP writes "1" to clear the corresponding Interrupt Registers' bit and this bit is self-cleared to "0" (by MCU) afterwards. |
| 0x3B [2] | AC Missing_CLR | RW | 0 | AC Missing interrupt flag clear. Interrupt only generated after power up from battery (external source different from AC power). |
| 0x3B [1] | Reserved | R | 0 | Reserved |
| 0x3B [0] | Reserved | R | 0 | Reserved |

Set bits in this register to clear corresponding interrupt flags. The register is self-cleared. Writing to this register does not invoke the clear by itself. The user must set BIT 5 in System Command Register (0x4E) to trigger the interrupt clear event (see TX Mode System Command Register, TxSysCmnd_L (0x4E), TxSysCmnd_H (0x4F)).

2.1.12.5. System TX Command Register, TX_CMND (0x4D)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x4D [7] | TXModeEn | W | 0x0 | Enable TX mode when in WPC mode (AC missing state) |
| 0x4D [6:0] | Reserved | R | 0x0 | Reserved |

The AP sets any of the bits in this register to initiate TX commands. The register is self-cleared when the command is read by the FW and the process loaded in the execution queue. The register is usually read within 1ms after it was modified and immediately cleared after that. It should be understood the clearing event does not mean command was already executed.

2.1.12.6. TX Mode System Command Register, TxSysCmnd_L (0x4E), TxSysCmnd_H (0x4F)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x4E [7:6] | Reserved | R | 0 | Reserved |
| 0x4E [5] | TX_CLR_INT | RW | 0 | If AP sets this bit to "1" then MFC-IC MCU clears the interrupt corresponding to the bit(s) which has a value of "1" in Interrupt Clear Registers and also sets the bit(s) in Interrupt Clear Registers to "0", as well as sets this bit to "0" |
| 0x4E [4:2] | Reserved | R | 0 | Reserved |
| 0x4E [1] | SEND_ACK | RW | 0 | If AP sets this bit to "1" then MFC-IC MCU sends the ACK (defined in the WPC spec) to RX and then MFC's MCU sets this bit to "0" |
| 0x4E [0] | TX SEND PPP | RW | 0 | If AP sets this bit to "1" then MFC-IC MCU sends the Proprietary Packet (defined in the Proprietary Packet Registers) to RX and then MFC's MCU sets this bit to "0" |
| 0x4F [7:2] | Reserved | R | 0 | Reserved |
| 0x4F [1] | TXMODE_EXIT | RW | 0 | Set to exit TX Mode |
| 0x4F [0] | Reserved | R | 0 | Reserved |

The AP sets any of the bits in this register to initiate the corresponding process. The register is self-cleared when the command is read by the FW and the process loaded in the execution queue. The register is usually read within 1ms after it was modified and immediately cleared after that. It should be understood the clearing event does not mean command was already executed.

2.1.12.7. TX Mode CEP Threshold Register, TxCepThrshVal (0xD6)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0xD6 [7:0] | TxCepThrshVal | R/W | 0x7F | 2's compliment signed integer representing values -128 ~ +127. The value sets the threshold for incrementing the CEP counter. Setting range is 0x00 to 0x7F. If the CEP value is greater or equal to the TxCepThrshVal for TxCepThrshCntLmt consecutive packets, a TX Mode CEP OVERDRV interrupt (0x36 [6]) is generated. Default value is TxCepThrshVal = 0x7F = 127 |

2.1.12.8. TX Mode CEP Threshold Count Limit Register, TxCepThrshCntLmt (0xD7)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0xD7 [7:0] | TxCepThrshCntLmt | R/W | 0x7F | The value sets the limit for the number of consecutive CEP packets with values that meet or exceed the threshold value. Setting range is 0x00 to 0xFF. If the CEP value is greater or equal to the TxCepThrshVal for TxCepThrshCntLmt consecutive packets, a TX Mode CEP OVERDRV interrupt (0x36 [6]) is generated. Default value is TxCepThrshCntLmt = 0x7F = 127 |

1. Need to set register 0xD7 to 0x00h(reset) before setting real max counter number. (Step1 0xD7h=0x00h, Step2 0xD7h=0x7Fh (127 count)).

2.1.12.9. TX Mode API Current Limit Registers, I_API_Limit_L (0x56), I_API_Limit_H (0x57)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x56 [7:0] | I_API_Limit [7:0] | RW | 0xDC | 8 LSB of the TX mode input current limit. The AP writes to this register to update this value as required. Default is 1500mA. |
| 0x57 [7:0] | I_API_Limit [15:8] | RW | 0x05 | 8 MSB of the TX mode input current limit. The AP writes to this register to update this value as required. Default is 1500mA. <i>Example: If 1.5A = 1500mA, 1500 => 0x5DCh</i> |

Note: API current limit is not a hardware limit control and input currents that are above the API current limit can result due to:

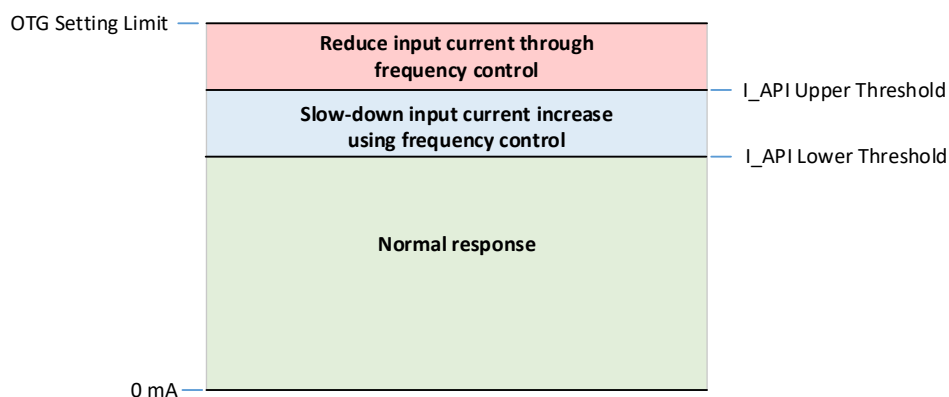
1. Large load transients that start below the hysteresis level and end above the current limit setting (I_API_Limit – I_API_Offset) causing a current overshoot response.
2. The TX mode frequency is at its maximum limit (register 0x96) and the duty is at its minimum limit (register 0x93), then the firmware is unable to reduce the input current further.

When operating in TX mode, the FW monitors the ADC TX input current and determines how the TX responds to increasing input current. The ADC TX input current is divided into three regions defined by the I_API_Limit (register 0x56), I_API_Offset (register 0xF6), and I_API_Hys (register 0x82) parameters. These parameters define an upper and lower threshold level:

$$I_API \text{ Upper Threshold} = I_API_Limit (0x56) - I_API_Offset (0xF6)$$

$$I_API \text{ Lower Threshold} = I_API \text{ Upper Threshold} \times I_API_Hys (0x82) / 128$$

I_API Current Limit



If the ADC Input current is below the I_API Lower Threshold, the FW will operate in the normal TX mode and has a normal response to increasing input current. If the ADC Input current is above the I_API Lower threshold but below the I_API Upper threshold then the FW has a slow response to increasing input current to minimize overshoot. If the ADC Input current is above the I_API Upper threshold then the FW increases the frequency until the ADC Input current drops below the I_API Upper threshold.

In the large transient case #1 above, the ADC Input current starts out in “normal response” region. So when a large load step is applied to the RX, the TX will respond with a large input current increase. When the RX sends the next CEP, the ADC Input current may still be in the “normal response” region and the TX will respond with another large input current increase. On the following RX CEP, the ADC Input current may have skipped the “slow-down input current” region and enters the “reduce input current” region. By this time the TX input current has overshoot the I_API Upper threshold and the FW will now begin to increase the frequency until the ADC Input current drops below the I_API Upper threshold.

In the maximum frequency case #2 above, the same sequence of events of case #1 occurs. However, when the TX is in the “reduce input current” region it is unable to increase the frequency beyond the maximum frequency

limit set by TX Mode Max Operating Frequency Register (0x96) and reduce the duty below the minimum duty set by the TX Mode Minimum Duty Setting Register (0x93). In this case the FW is unable to force the TX input current below the I_API Upper threshold level.

2.1.12.10. TX Mode API Current Hysteresis Register, I_API_Hys (0x82)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x82 [7:0] | I_API_Hys [7:0] | RW | 0x73 | TX mode input current limit hysteresis. The AP writes to this register to update this value as required. Default is 90%. Example: I_API_Hys = 0x73 => 115/128 => 0.90, so hysteresis level is I_API_Limit x I_API_Hys, or 1500 x 0.90 = 1350mA |

2.1.12.11. TX Mode API Current Limit Offset Registers, I_API_Offset_L (0xF6), I_API_Offset_H (0xF7)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0xF6 [7:0] | I_API_Offset [7:0] | RW | 0x96 | 8 LSB of the TX mode input current limit offset. The AP writes to this register to update this value as required. Default is 150mA. |
| 0xF7 [7:0] | I_API_Offset [15:8] | RW | 0x00 | 8 MSB of the TX mode input current limit offset. The offset is signed and needs to be converted in 2's compliment. The AP writes to this register to update this value as required. Default is 150mA. Example: If 150mA 150 => 0x0096, -150mA => 0xFF6A |

2.1.12.12. TX Ping Frequency (Period) Register, PingFreqPer_L (0x90), PingFreqPer_H (0x91)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|----------------------|-----|---------------|--|
| 0x90 [7:0] | PingFreqPer_L [7:0] | RW | 0x36 | Ping frequency in number of 120MHz clock cycles Default value of 146 kHz |
| 0x91 [7:0] | PingFreqPer_H [15:8] | RW | 0x03 | Ping frequency in number of 120MHz clock cycles $PingFreqPer (cnts) = 120 \times 10^3 (kHz) / Ping Freq (kHz) = 120,000 / 146 = 821 \text{ dec} = 0x0336$ |

2.1.12.13. TX Ping Duty Cycle Register, PingDC (0x92)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x92 [7:0] | PingDC [7:0] | RW | 0x80 | TX Ping Duty Cycle. Default value of 50% $PingDC / 256 = \text{Duty Cycle (e.g. } 128 = 50\%)$ |

2.1.12.14. TX Minimum Duty Cycle Register, MinDC (0x93)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x93 [7:0] | MinDC [7:0] | RW | 0x4C | TX Minimum Duty Cycle. Default value of 30% $MinDty / 256 = \text{Duty Cycle (e.g. } 51 = 20\%)$ |

2.1.12.15. TX Minimum Frequency (Period) Register, MinFreqPer_L (0x94), MinFreqPer_H (0x95)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x94 [7:0] | MinFreqPer_L [7:0] | RW | 0x43 | TX mode minimum allowable frequency in number of 120MHz clock cycles Default value of 110 kHz |
| 0x95 [7:0] | MinFreqPer_H [15:8] | RW | 0x04 | TX mode minimum allowable frequency in number of 120MHz clock cycles MinFreqPer (cnts) = 120×10^3 (kHz) / Min Freq (kHz) = 120,000 / 110 = 1090 dec = 0x0443 |

2.1.12.16. TX Maximum Frequency (Period) Register, MaxFreqPer_L (0x96), MaxFreqPer_H (0x97)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x96 [7:0] | MaxFreqPer_L [7:0] | RW | 0x2B | TX mode maximum allowable frequency in number of 120MHz clock cycles Default value of 148 kHz |
| 0x97 [7:0] | MaxFreqPer_H [15:8] | RW | 0x03 | TX mode maximum allowable frequency in number of 120MHz clock cycles MaxFreqPer (cnts) = 120×10^3 (kHz) / Max Freq (kHz) = 120,000 / 148 = 811 dec = 0x032B |

2.1.12.17. TX Mode Operating Period Register, TxPeriod_L (0xA4), TxPeriod_H (0xA5)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0xA4 [7:0] | TxPeriod_L [7:0] | R | - | TX mode operating period in number of 120MHz clock cycles |
| 0xA5 [7:0] | TxPeriod_H [15:8] | R | - | TX mode operating period in number of 120MHz clock cycles TX mode operating freq (kHz) = 120×10^3 (kHz) / TxPeriod (cnts) Example: Tx operating freq = 120,000 / 923 = 130 kHz |

2.1.12.18. TX Mode Operating Duty Cycle Register, TxDuty (0xA6)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0xA6 [7:0] | TxDuty [7:0] | R | - | TX mode operating Duty Cycle. Tx operating Duty Cycle = TxDuty / 256 Example: Tx operating duty = 128 / 256 = 50% |

2.1.12.19. TX Mode Over-Voltage Protection Register, Tx_OVP_L (0x9E), TX_OVP_H (0x9F)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x9E [7:0] | Tx_OVP_L [7:0] | R | 0x10 | 8 LSB of current unsigned integer value contained in this field indicating the Tx Mode Over-voltage protection threshold. |
| 0x9F [7:0] | Tx_OVP_H [15:8] | R | 0x27 | 8 MSB of current unsigned integer value contained in this field indicating the Tx Mode Over-voltage protection threshold. Tx_OVP = 0x2710 = 10,000 dec mV = 10V. |

2.1.12.20. TX Mode Over-Current Protection Register, TX_OCP_L (0xA0), TX_OCP_H (0xA1)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0xA0 [7:0] | Tx_OCP_L [7:0] | RW | 0x34 | 8 LSB of current unsigned integer value contained in this field indicating the Tx Mode Over-current protection threshold. |
| 0xA1 [7:0] | Tx_OCP_H [15:8] | RW | 0x08 | 8 MSB of current unsigned integer value contained in this field indicating the Tx Mode Over-current protection threshold. Tx_OCP = 0x0834 = 2,100 dec mA = 2.1A. |

2.1.12.21. TX Conflict Threshold Register, TxConfThrsh (0x134)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0x134 [7:0] | TxConfThrsh [7:0] | RW | 0xA0 | 8 LSB of current unsigned integer value contained in this field indicating the Tx Conflict threshold. When a phone, in TX Mode, is placed on or near a foreign TX, an AC voltage is generated on the DEMOD pin. If the DEMOD voltage is greater or equal to the TxConfThrsh value for TxConfCnt consecutive times, a TX Conflict interrupt is generated. |
| 0x135 [7:0] | TxConfThrsh [15:8] | RW | 0x04 | 8 MSB of current unsigned integer value contained in this field indicating the Tx Conflict threshold. TxConfThrsh = 0x04A0 = 1184 dec => $1184 * 2.1 / 4095 = 0.607$ V. |

2.1.12.22. TX Conflict Count Register, TxConfCnt (0xAF)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|--|
| 0xAF [7:0] | TxConfCnt [7:0] | RW | 0x10 | TX Conflict count sets the limit for the number of consecutive times the TX Conflict threshold is exceeded before an interrupt is triggered and the TX pings are stopped |

2.1.12.23. TX Power Register, TX_Power_L (0xAC), TX_Power_H (0xAD)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0xAC [7:0] | TX_Power_L [7:0] | RW | 0x00 | 8 LSB of current unsigned integer value contained in this field indicating the TX Power in mW. |
| 0xAD [7:0] | TX_Power_H [15:8] | RW | 0x00 | 8 MSB of current unsigned integer value contained in this field indicating the TX Power in mW. TX_Power = 0x1388 = 5000 dec mW = 5W. |

2.1.12.24. TX Dead Time Register, TX_DeadTime (0x9C)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|---------------------|-----|---------------|---|
| 0x9C [7:0] | TX_DeadTime [7:0] | RW | 0x0A | Sets the dead time of the rectifier switches in TX mode. Default value is 0x0A= 83.3 ns $TX_DeadTime (s) = TX_DeadTime (decimal) / 120MHz = 10 / 120M = 83.3ns$ |

Note: TX dead time register is only populated in TX mode.

2.1.12.25. TX Control Delay Register, TX_ControlDelay (0xAE)

| Address and bit | Register Field Name | R/W | Default Value | Function and Description |
|-----------------|-----------------------|-----|---------------|--|
| 0xAE [7:0] | TX_ControlDelay [7:0] | RW | 0x04 | Sets the delay time of the rectifier switches in TX mode. Default value is 0x04 = 33.3 ns $TX_ControlDelay (s) = TX_ControlDelay (decimal) / 120MHz = 4 / 120M = 33.3ns$ |

Note: TX Control Delay register is only populated in TX mode.

2.1.13. Commonly Used Registers

2.1.13.1. Monitoring, Status, Interrupt, and Operating Mode Registers

| Parameter | Register Address | R/W | Operating Modes | Reference Table No. |
|-------------------|------------------|-----|-----------------|------------------------------|
| Vout | 0x42 (2 bytes) | R | Rx, TRx | 2.1.4.2 |
| Iout / Iin | 0x44 (2 bytes) | R | Rx, TRx | 2.1.4.4 |
| CPout | 0x10C (2 bytes) | R | Rx, TRx | 2.1.8.3 |
| Vrect | 0x40 (2 bytes) | R | Rx, TRx | 2.1.4.5 |
| DieTemp | 0x46 (2 bytes) | R | Rx, TRx | 2.1.4.6 |
| AC Frequency | 0x48 (2 bytes) | R | Rx, TRx | 2.1.4.7 |
| Duty Cycle | 0xA6 (1 byte) | R | TRx | 2.1.12.18 |
| System Status | 0x34 (2 bytes) | R | Rx, TRx | 2.1.2.1 (Rx), 2.1.12.1 (TRx) |
| Interrupts | 0x36 (2 bytes) | R | Rx, TRx | 2.1.2.2 (Rx), 0 (TRx) |
| Interrupt Enables | 0x38 (2 bytes) | R | Rx, TRx | 2.1.2.3 (Rx), 2.1.12.3 (TRx) |
| Interrupt Clear | 0x3A (2 bytes) | R | Rx, TRx | 2.1.2.4 (Rx), 2.1.12.4 (TRx) |
| System Op Modes | 0x4C (1 byte) | R | Rx, TRx | 2.1.2.5 |
| CD Mode Status | 0x100 (1 byte) | R | Rx | 2.1.8.1 |

2.1.13.2. Settings and Command Registers

| Parameter | Register Address | R/W | Operating Modes | Reference Table No. |
|-------------------------|----------------------|-----|-----------------|---------------------|
| Vout_Set | 0x6C (2 bytes) | RW | Rx, | 2.1.4.1 |
| ILim | 0x42 (1 byte) | RW | Rx | 2.1.4.3 |
| Vrect_Adj | 0x5E (signed 1 byte) | RW | Rx | 2.1.7.6 |
| System Commands | 0x4E (2 bytes) | RW | Rx, | 2.1.5.1 |
| CD Mode Request* | 0x101 (1 byte) | RW | Rx | 2.1.4.6 |
| Tx Command | 0x4D (1 byte) | RW | TRx | 2.1.12.5 |
| TX Mode System Commands | 0x4E (2 bytes) | RW | TRx | 2.1.12.6 |

* For AP control of CD mode transitions from Bypass to Cap Divider mode or from Cap Divider mode to Bypass. The following procedure should be used:

1. Read the CD Mode Status register 0x100. If the value is 0x01 then operating in Bypass mode.
2. To transition from Bypass to Cap Divider mode, enter a value of 0x02 to the CD Mode Request register (0x101) to request Cap Divider mode operation.
3. Execute the change by writing 0x40 to the System Command Register (0x4E).
4. Verify the CD mode change by reading the CD Mode Status register 0x100.

3. Schematic Diagram

IDTP9412 CSP DEMO PCB V1.3 ADVANCED INFORMATION SUBJECT TO CHANGE

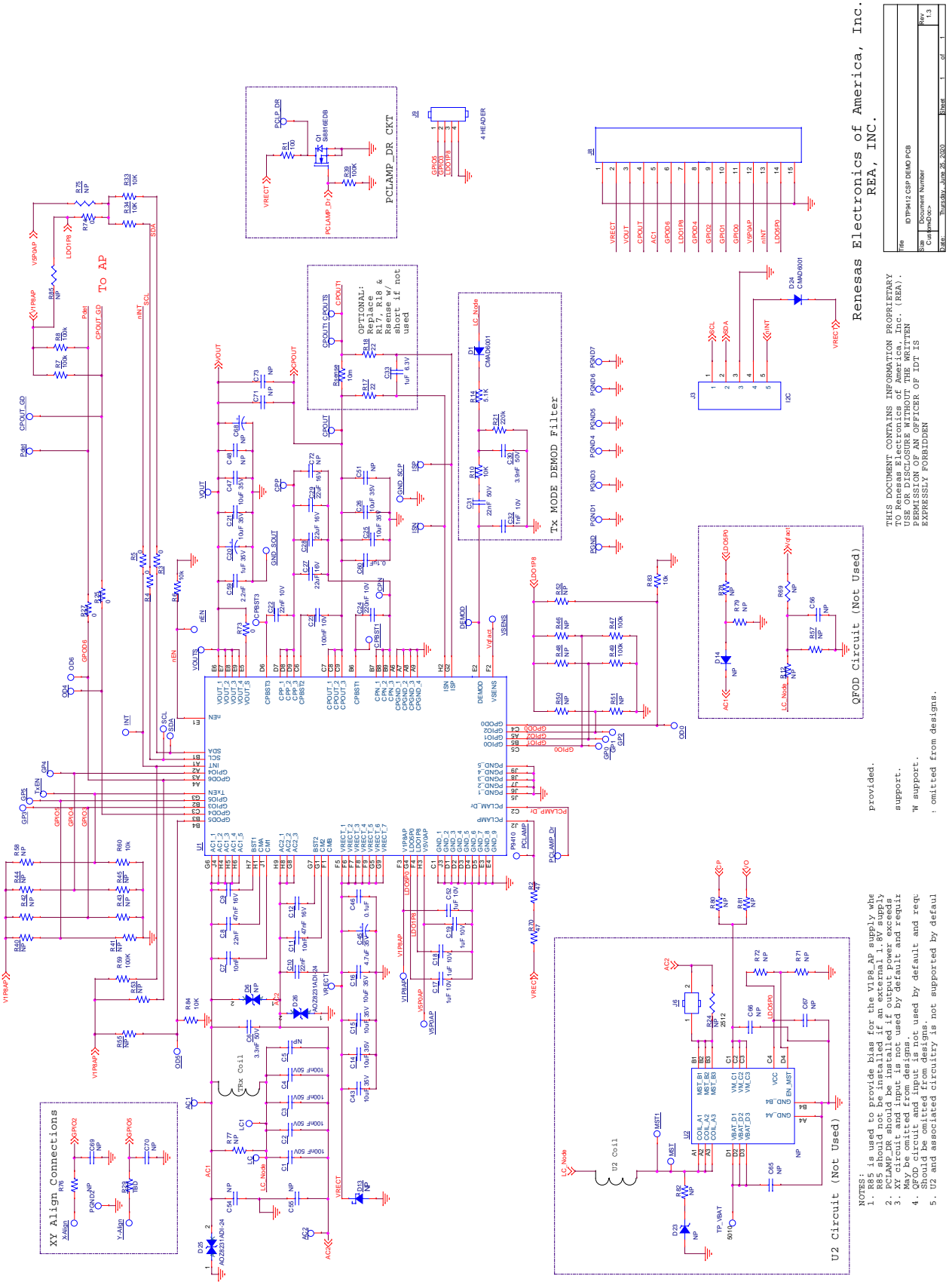


Figure 31. P9412 DEMO Board v1.3 Schematic

NOTES:

1. R85 is used to provide bias for the VI8AP supply when the VI8AP supply is not used.
2. PCLAMP_DR should be installed if output power exceeds 100mW.
3. XY circuit and input is not used by default and requires support.
4. May be omitted from designs. It is used by default and requires support.
5. U2 and associated circuitry is not supported by default.

THIS DOCUMENT CONTAINS INFORMATION PROPRIETARY TO Renesas Electronics of America, Inc. (RE.A). USE OR DISCLOSURE WITHOUT THE WRITTEN EXPRESSLY FORBIDDEN.

Revised: 2008.05.26.2008 Rev. 1.3

Revised: 2008.05.26.2008 Rev. 1.3

4. Bill of Materials

Table 1. Bill of Materials

| Item | Quantity | Reference | Part | PCB Footprint | Part Number |
|------|----------|---|---------------|------------------------|--------------------|
| 1 | 19 | PGND1,LC1,CPOUT1, AC1, AC2, Y-Align, X-Align, VRECT,VOUTS,VOUT,SDA,SCL,PGND, PCLAMP,LC,GND_SOUT,GND_SCP, CPOUTS,CPOUT | TP | test_pt_sm_135 x70 | |
| 2 | 6 | CPBST1,CPBST3, CPP,CPN, VSENS,DEMODO | NP | tp_sm_45D | SMD_Pad_Only |
| 3 | 4 | C1,C2,C3,C4 | 100nF 50V | 0402 | GRM155R61H104KE19D |
| 4 | 1 | C5 | NP | 0402 | GRM155R61H104KE19D |
| 5 | 1 | C6 | 3.3nF 50V | 0402 | CL05B332KB5NNNC |
| 6 | 2 | C7, C11 | 10nF | 0402 | CL05B103KB5NNNC |
| 7 | 2 | C8,C10 | 22nF | 0402 | GCM155R71H223KA55D |
| 8 | 2 | C9,C12 | 47nF 16V | 0201 | GRM033R61C473KE84D |
| 9 | 8 | C14,C15,C16,C21,C25,C26, C43,C47 | 10uF 35V | 0603 | GRM188R6YA106MA73D |
| 10 | 4 | C17,C18,C19,C52 | 1uF 10V | 0201 | GRM033R61A105ME15D |
| 11 | 1 | C20 | 1uF 35V | Cap_pol_2p0x1 p25mm | T58W9105M035C0500 |
| 12 | 1 | C22 | 22nF 10V | 0201 | CL03A223KP3NNNC |
| 13 | 1 | C23 | 100nF 10V | 0201 | CL03A104MP3NNNC |
| 14 | 1 | C24 | 220nF 10V | 0201 | CL03A224KP3NNNC |
| 15 | 3 | C27,C28, C29 | 22uF 16V | 0603 | CL10A226M07JZNC |
| 16 | 1 | C30 | 3.9nF, 50V | 0402 | CL05B392JB5NNNC |
| 17 | 1 | C31 | 22nF 50V | 0402 | GCM155R71H223KA55D |
| 18 | 1 | C32 | 1nF 10V | 0201 | 0201ZD102KAT2A |
| 19 | 1 | C33 | 1uF 6.3V | 0402 | CL05A105MQ5NNNC |
| 20 | 1 | C45 | 4.7uF 35V | Cap_pol_3p5x2 p8mm | TCNL475M035R0300 |
| 21 | 2 | C46,C60 | 0.1uF | 0201 | GRM033R6YA104KE14D |
| 22 | 4 | C48,C51,C71,C73 | NP | 0603 | GRM188R6YA106MA73D |
| 23 | 17 | R40,R41,R42,R43,R44,R45, R46,R48,R50,R51,R52,C54, C55,C56,R71,R76,R77 | NP | 0402 | NP |
| 24 | 1 | C59 | 2.2nF | 0201 | GRM033R71E222KA12J |
| 25 | 2 | C65,C67 | NP | 0201 | GRM033R61A105ME15D |
| 26 | 1 | C66 | NP | 0402 | CL05A105KA5NQNC |
| 27 | 1 | C68 | NP | Cap_pol_2p0x1 p25mm | T529P475M025AAE300 |
| 28 | 2 | C69,C70 | NP | 0201 | 0201ZD102KAT2A |
| 29 | 1 | C72 | NP | 0603 | CL10A226M07JZNC |
| 30 | 2 | D1,D24 | CMAD6001 | sod923 | CMAD6001 TR |
| 31 | 1 | D6 | NP | SOD323 | SD24C-01FTG |
| 32 | 1 | D13 | NP | POWER DI123 | DFLT24A |
| 33 | 1 | D14 | NP | sod923 | CMAD6001 TR |
| 34 | 1 | D23 | NP | SMini2-F5-B | DZ2J360M0L |
| 35 | 2 | D25,D26 | AOZ8231ADI-24 | dfn1006_ 2ld_diode | AOZ8231ADI-24 |

| Item | Quantity | Reference | Part | PCB Footprint | Part Number |
|------|----------|--|-----------|-----------------------------|------------------|
| 36 | 17 | GP1,GP2,GP3,OD4,GP4,OD5, GP5,OD6,OD0,CPOUT_GD, nEN,VOUT_GD, TxEN, Pdet, INT, PCLP_DR, PCLAMP_Dr, GP0 | NP | TP_SM_30CIR | SMD_Pad_Only |
| 37 | 2 | ISP,ISN | NP | TP_SM_30CIR | SMD_Pad_Only |
| 38 | 1 | J3 | I2C | sip5 | 901200765 |
| 39 | 1 | J6 | NP | jumper2pin01in | 68000-102HLF |
| 40 | 1 | J8 | Header15 | header_1x15_0 p1Pitch60p42d | TSW-115-14-T-S |
| 41 | 1 | J9 | 4 HEADER | sip-4 | 3-644456-4 |
| 42 | 2 | TP_VBAT,MST | NP | test_pt90_65d | |
| 43 | 1 | MST1 | NP | test_pt_sm_135 x70 | |
| 44 | 2 | PGND2,PGND4 | NP | test_pt90_65d | |
| 45 | 4 | PGND3,PGND5,PGND6,PGND7 | TP | test_pt90_65d | |
| 46 | 2 | V1P8AP,V5P0AP | TP | test_pt90_65d | |
| 47 | 1 | Q1 | Si8816EDB | BGA-4 | SI8816EDB-T2-E1 |
| 48 | 1 | Rsense | 10m | 0402_0603_0805 | LVT04R0100FER |
| 49 | 1 | R1 | 100 | 0805 | CRGH0805F100R |
| 50 | 2 | R2,R70 | 47 | 0603 | CRGP0603F47R |
| 51 | 6 | R3,R4,R5,R25,R27,R73 | 0 | 0201 | ERJ-1GN0R00C |
| 52 | 2 | R6,R60 | 10k | 0201 | RC0603J103CS |
| 53 | 4 | R7,R8,R39,R59 | 100K | 0201 | RC0201JR-07100KL |
| 54 | 1 | R10 | 10K | 0201 | RMCF0201FT10K0 |
| 55 | 3 | R12,R78,R79 | NP | 0201 | RMCF0201FT10K0 |
| 56 | 1 | R14 | 5.1K | 0201 | RC0201JR-075K1L |
| 57 | 2 | R17,R18 | 22 | 0201 | CRCW020122R0FNED |
| 58 | 1 | R21 | 220K | 0201 | AC0201FR-07220KL |
| 59 | 1 | R24 | NP | 2512 | CSRN2512FKR500 |
| 60 | 1 | R29 | TBD | 0402 | NP |
| 61 | 3 | R33,R34, R84 | 10K | 0402 | TRR01MZPF1002 |
| 62 | 2 | R47,R49 | 100K | 0402 | RC0402FR-07100KL |
| 63 | 1 | R53 | NP | 0402 | RC0402JR-07100KL |
| 64 | 1 | R55 | NP | 0402 | RC0603J103CS |
| 65 | 1 | R57 | NP | 0201 | RC0201FR-071ML |
| 66 | 1 | R58 | NP | 0201 | |
| 67 | 3 | R69,R75,R85 | NP | 0402 | RMCF0402ZT0R00 |
| 68 | 1 | R72 | NP | 0402 | RC0201JR-075K1L |
| 69 | 1 | R74 | 0 | 0402 | RMCF0402ZT0R00 |
| 70 | 2 | R80,R81 | NP | 0603 | RC1608J000CS |
| 71 | 1 | R82 | NP | 0402_0603 | RC1608J000CS |
| 72 | 1 | R83 | 10K | 0402 | RC0603J103CS |
| 73 | 1 | U1 | P9412 | DSBGA81LD_9 x9_0p4mm | P9412-1AWQI8 |

1. Recommended capacitor temperature/dielectric and voltage ratings for WPC resonance capacitors is 50 V with low ESR capacitors. Furthermore, COG/NPO-type capacitor values stay constant with voltage while X7R and X5R capacitor values derate over the working voltage range at 40% to over 80%.

5. Board Layout

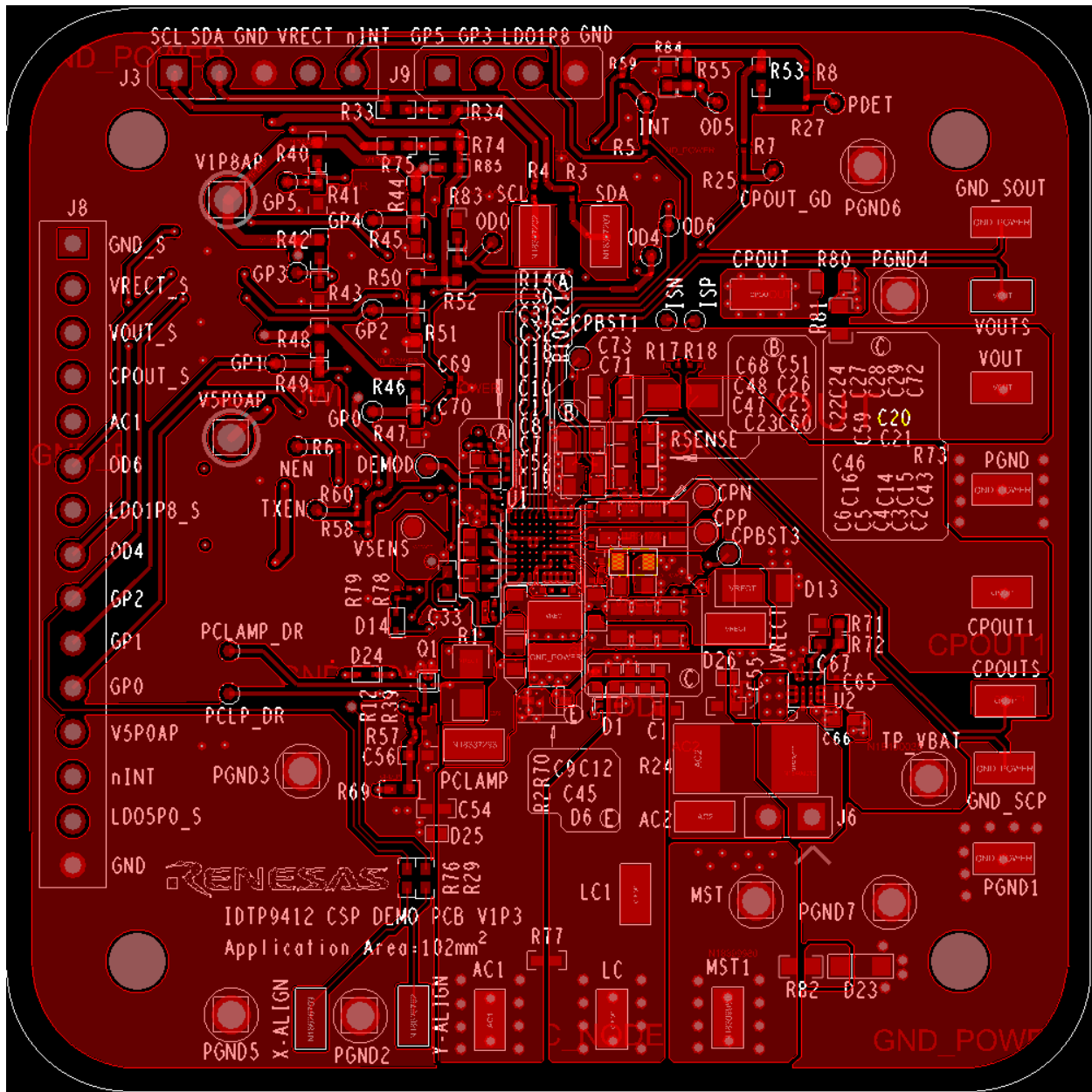


Figure 32. Top and Top Silkscreen Layer

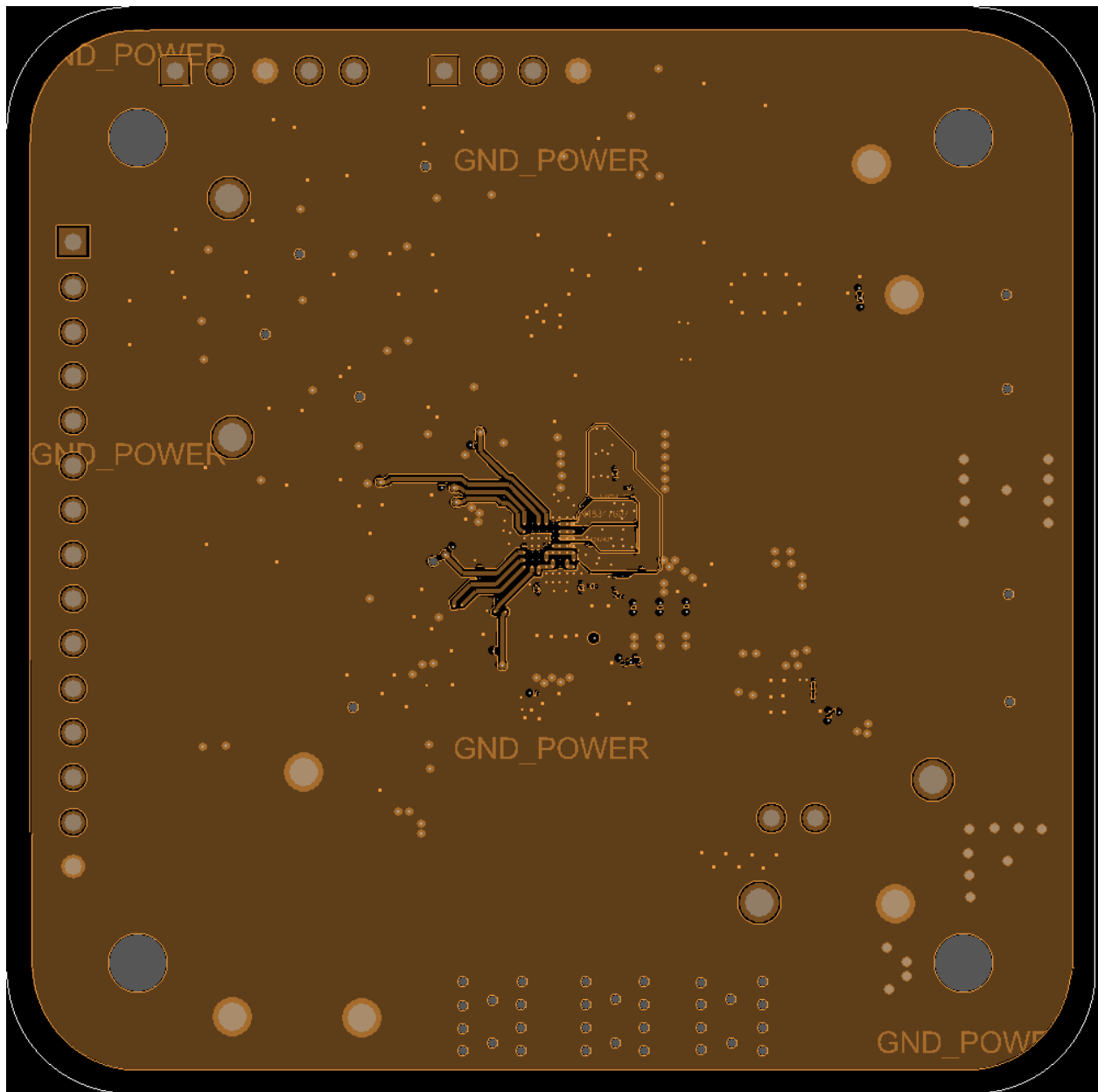


Figure 33. Inner1 GND Layer

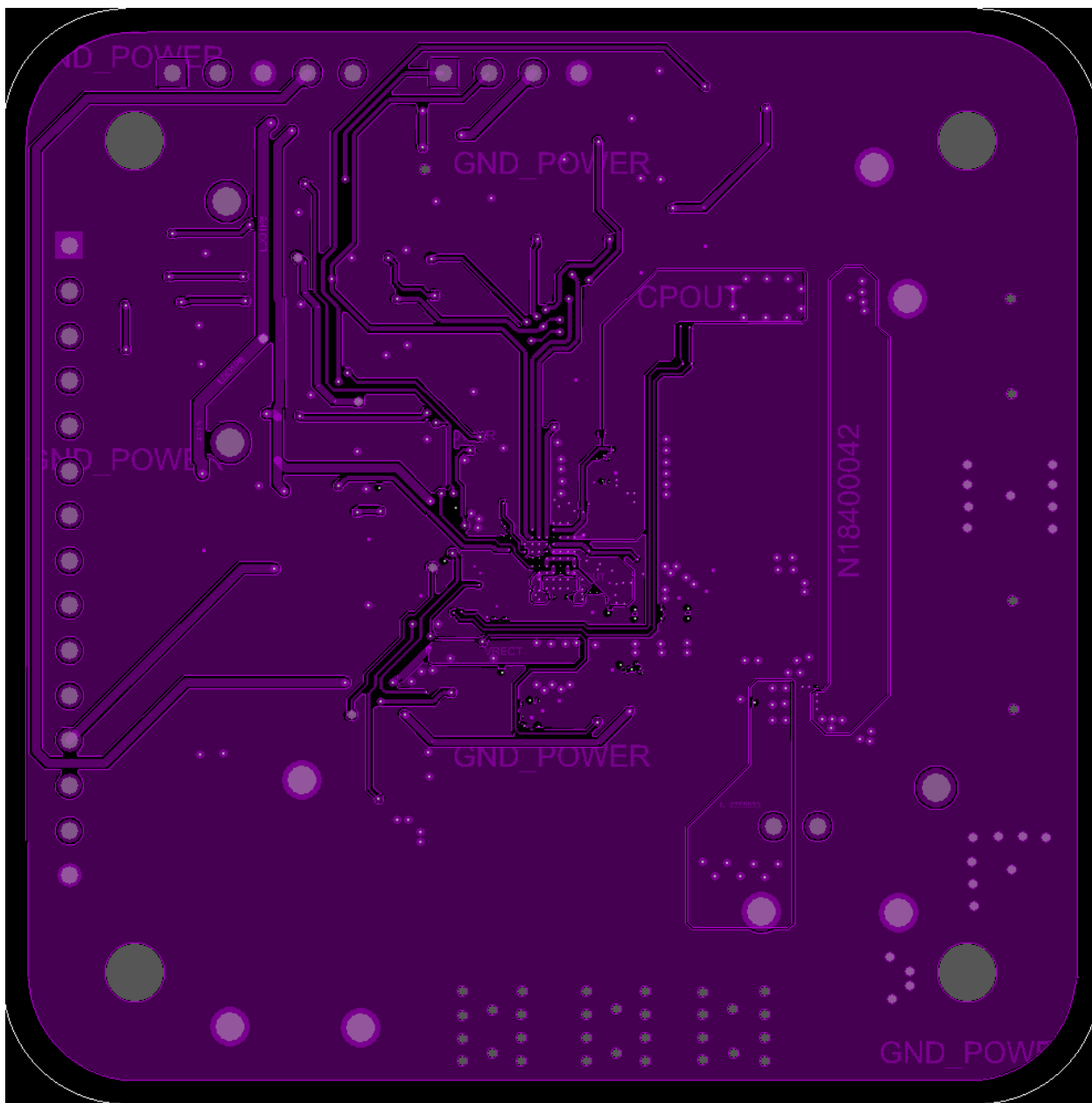


Figure 34. Inner2 POWER/Signal/GND Layer

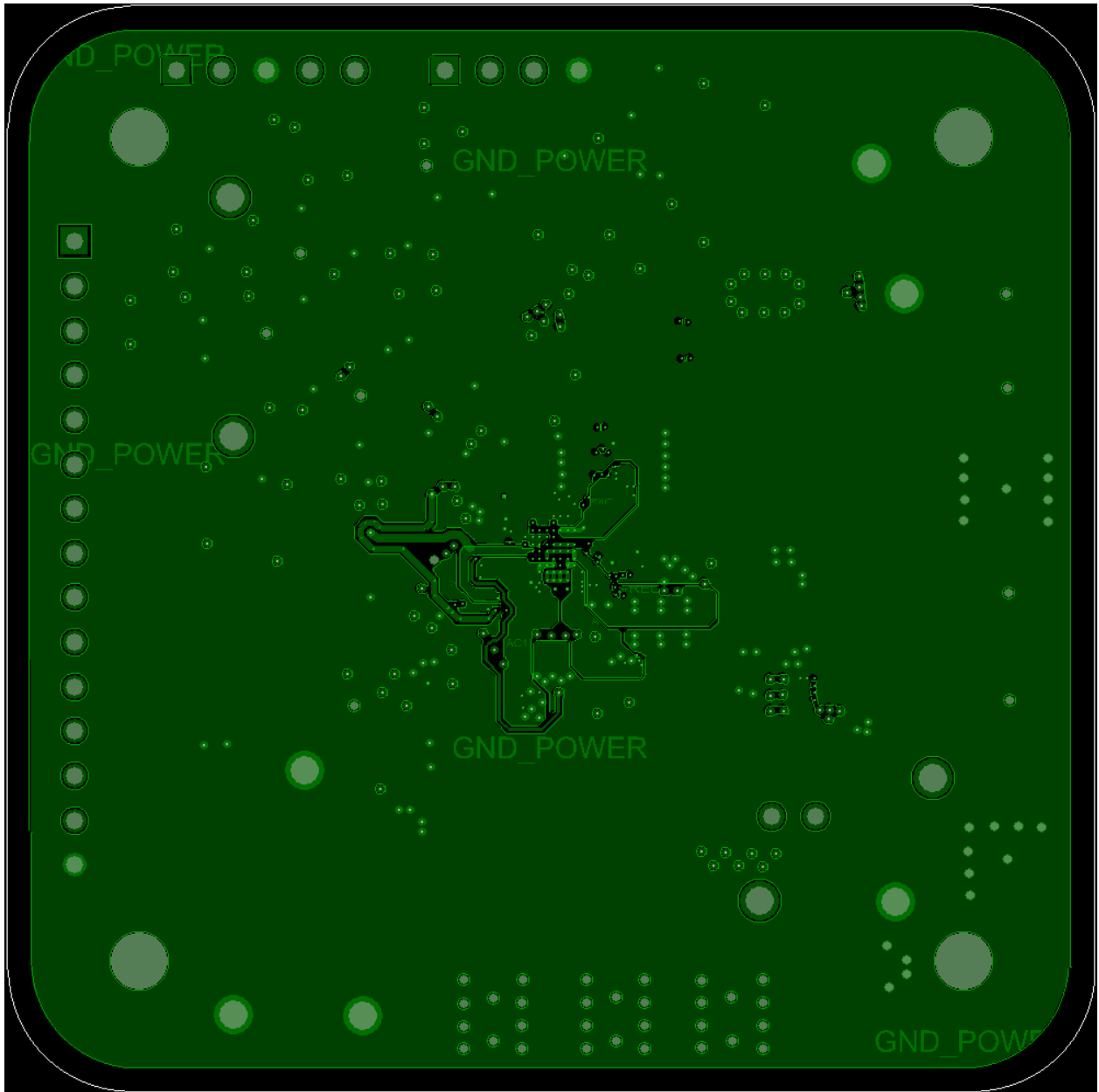


Figure 35. Inner3 POWER/ GND Layer

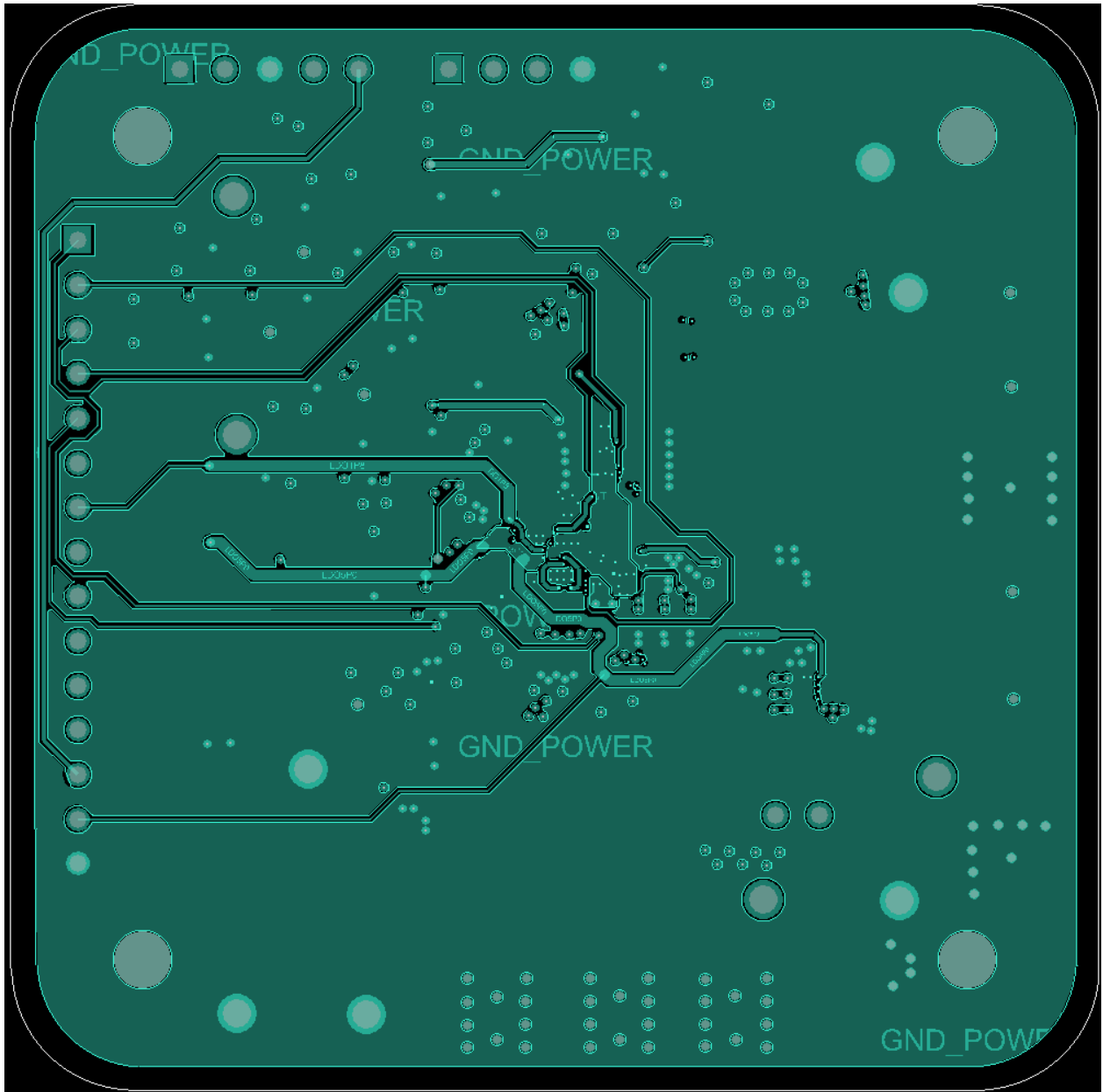


Figure 36. Inner4 POWER/Signal/GND Layer

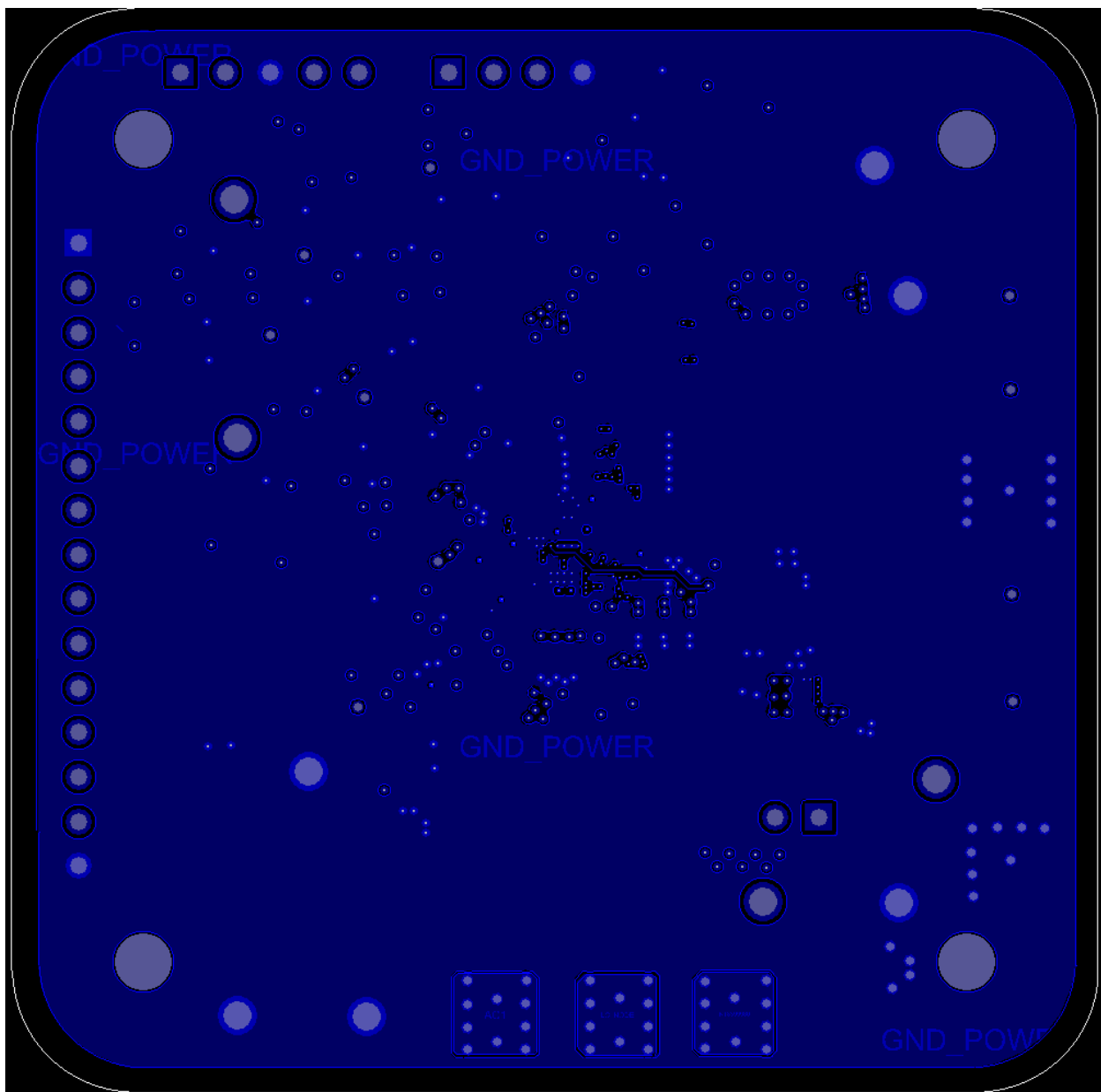


Figure 37. Bottom Layer

5.1 FTDI Dongle

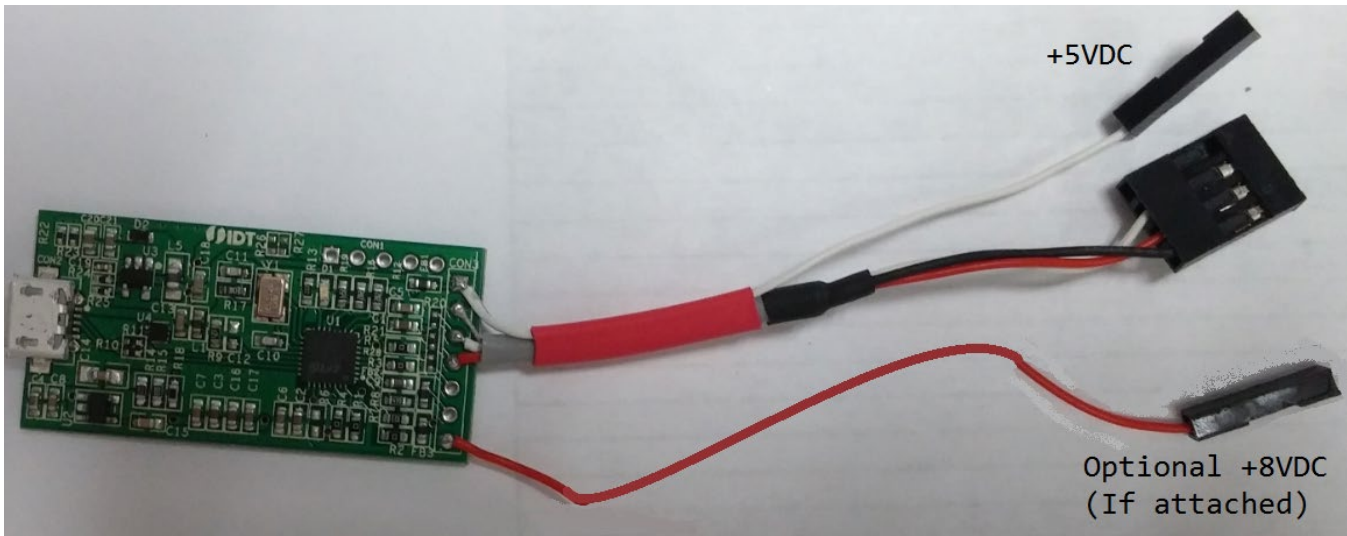


Figure 38. FTDI Dongle for Programming the Firmware into P9412 MTP

Caution: The +5V and +8V DC power supplies are intended for I2C rail bias only and current consumption must be limited to less than 50mA. These power supplies should not be loaded except for programming or register polling to prevent damage to the dongle.

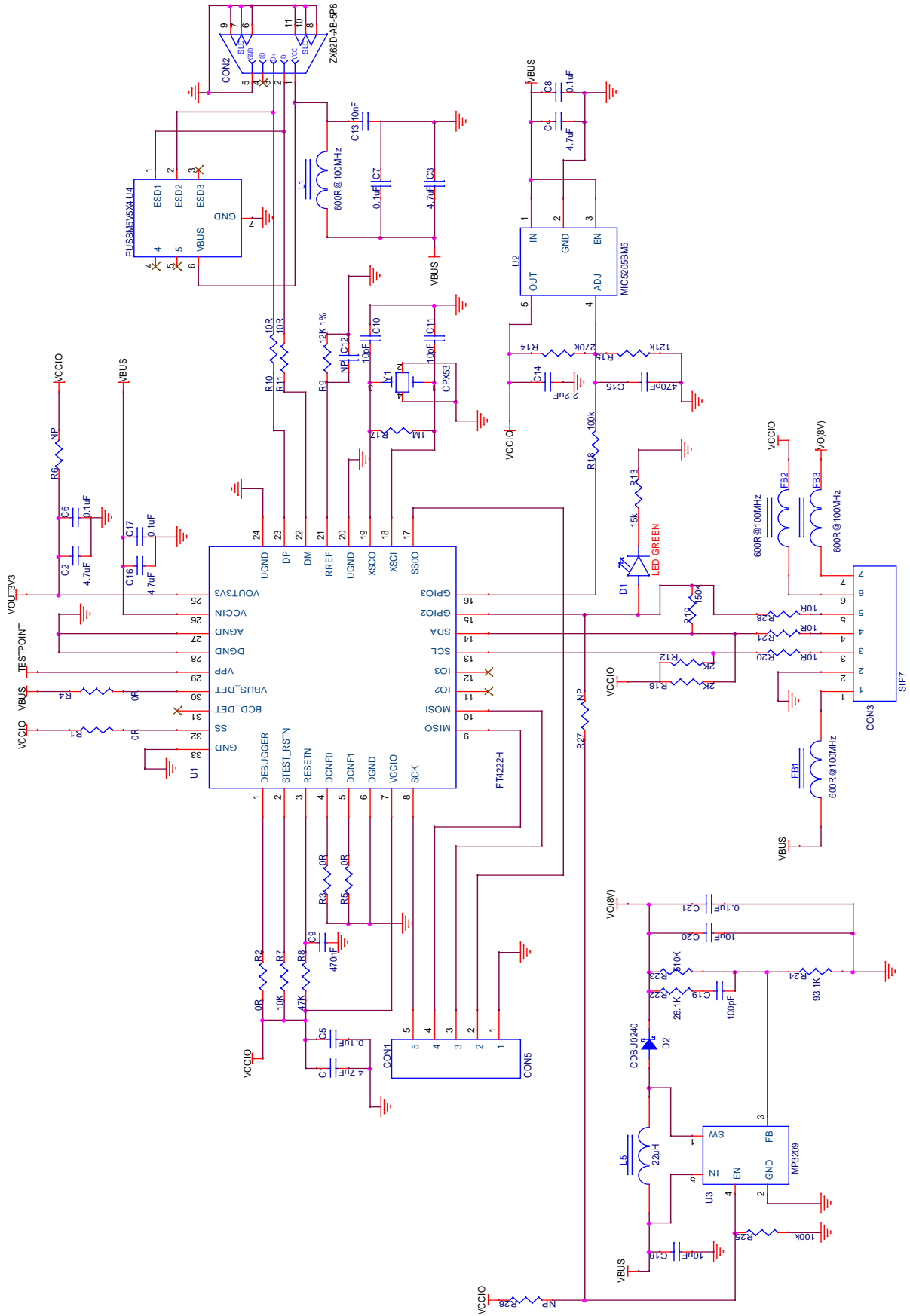


Figure 39. FTDI Dongle Schematic for Programming the Firmware into P9412 MTP and Reading Registers

6. Ordering Information

| Part Number | Temperature Range (°C) |
|-------------|------------------------|
| P9412-EVK | 0°C to +85°C |

7. Revision History

| Revision | Date | Description |
|----------|----------|---|
| 1.7 | Jul.8.20 | <ul style="list-style-type: none">• Added Proprietary High Power Protocol• Removed raw ADC registers |

IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD-PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers who are designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only to develop an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third-party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising from your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Disclaimer Rev.1.01 Jan 2024)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit www.renesas.com/contact-us/.